

# ReCentering Psych Stats: Psychometrics

Lynette H. Bikos, PhD, ABPP (she/her)

12 May 2024



# Contents

<b>1</b>	<b>Introduction</b>	<b>25</b>
1.1	What to expect in each chapter . . . . .	25
1.2	Strategies for Accessing and Using this OER . . . . .	26
1.3	If You are New to R . . . . .	26
1.3.1	Base R . . . . .	26
1.3.2	R Studio . . . . .	27
1.3.3	R Hygiene . . . . .	27
1.3.3.1	Everything is documented in the .rmd file . . . . .	28
1.3.3.2	File organization . . . . .	28
1.3.3.3	Chunks . . . . .	28
1.3.3.4	Packages . . . . .	29
1.3.3.5	Knitting . . . . .	29
1.3.4	tRoubleshooting in R maRkdown . . . . .	29
1.3.5	stRategies for success . . . . .	30
1.3.6	Resources for getting staRted . . . . .	30
<b>2</b>	<b>Questionnaire Construction: The Fundamentals</b>	<b>33</b>
2.1	Navigating this Lesson . . . . .	33
2.1.1	Learning Objectives . . . . .	33
2.1.2	Planning for Practice . . . . .	34
2.1.3	Readings & Resources . . . . .	34
2.2	Components of the Questionnaire . . . . .	34
2.3	What Improves (or Threatens) Response Rates and Bias? . . . . .	36
2.3.1	Should Likert-type scales include a midpoint? . . . . .	36
2.3.2	Should <i>continuous rating scales</i> be used in surveys? . . . . .	37
2.3.3	Should Likert-type response options use an ascending or descending order? .	38

2.3.4	Should surveys include negatively worded items? . . . . .	39
2.4	Construct-specific guidance . . . . .	40
2.5	Surveying in the Online Environment . . . . .	41
2.6	In my Surveys . . . . .	42
2.6.1	Demographics and Background Information . . . . .	42
2.6.2	Survey Order . . . . .	45
2.6.3	Forced Responses . . . . .	45
2.7	Practice Problems . . . . .	45
<b>3</b>	<b>Be a QualTRIXter</b>	<b>47</b>
3.1	Navigating this Lesson . . . . .	47
3.1.1	Learning Objectives . . . . .	47
3.1.2	Planning for Practice . . . . .	47
3.1.3	Readings & Resources . . . . .	48
3.1.4	Packages . . . . .	48
3.2	Research Vignette . . . . .	48
3.3	Qualtrics Essentials . . . . .	49
3.4	Qual-TRIX . . . . .	50
3.5	Even moRe, particularly relevant to iRb . . . . .	51
3.6	intRavenous Qualtrics . . . . .	52
3.6.1	The Codebook . . . . .	54
3.6.2	Using data from an exported Qualtrics .csv file . . . . .	55
3.6.3	Tweaking Data Format . . . . .	55
3.7	Practice Problems . . . . .	55
<b>4</b>	<b>Psychometric Validity: Basic Concepts</b>	<b>57</b>
4.1	Navigating this Lesson . . . . .	57
4.1.1	Learning Objectives . . . . .	57
4.1.2	Planning for Practice . . . . .	58
4.1.3	Readings & Resources . . . . .	58
4.1.4	Packages . . . . .	59
4.2	Research Vignette . . . . .	59
4.3	Fundamentals of Validity . . . . .	63
4.4	Validity Criteria . . . . .	64

<b>CONTENTS</b>	<b>5</b>
4.4.1 Content Validity . . . . .	64
4.4.2 Face Validity: The “Un”validity . . . . .	66
4.4.3 Criterion-Related Validity . . . . .	66
4.4.4 Construct Validity . . . . .	67
4.4.5 Internal Consistency . . . . .	67
4.4.6 Structural Validity . . . . .	68
4.4.6.1 Exploratory Factor Analysis . . . . .	68
4.4.6.2 Confirmatory Factor Analysis . . . . .	68
4.4.7 Experimental Interventions . . . . .	68
4.4.8 Convergent and Discriminant Validity . . . . .	69
4.4.8.1 Determining Statistically Significant Differences Between Correlations	72
4.4.8.2 Multitrait-Multimethod Matrix . . . . .	75
4.4.9 Incremental Validity . . . . .	76
4.4.10 Considering the Individual and Social Consequences of Testing . . . . .	79
4.5 Factors Affecting Validity Coefficients . . . . .	80
4.6 Practice Problems . . . . .	82
4.6.1 Problem #1: Play around with this simulation. . . . .	82
4.6.2 Problem #2: Conduct the reliability analysis selecting different variables. . .	82
4.6.3 Problem #3: Try something entirely new. . . . .	83
4.6.4 Grading Rubric . . . . .	83
4.7 Homeworked Example . . . . .	83
4.7.1 Check and, if needed, format data . . . . .	84
4.7.2 Create a correlation matrix that includes the instrument-of-interest and the variables that will have varying degrees of relation . . . . .	85
4.7.3 With convergent and discriminant validity in mind, interpret the validity coefficients; this should include an assessment about whether the correlation coefficients (at least two different pairings) are statistically significantly different from each other. . . . .	86
4.7.4 With at least three variables, evaluate the degree to which the instrument demonstrates incremental validity (this should involve two regression equations and their statistical comparison) . . . . .	90
<b>5 Reliability</b>	<b>93</b>
5.1 Navigating this Lesson . . . . .	93
5.1.1 Learning Objectives . . . . .	93
5.1.2 Planning for Practice . . . . .	93

5.1.3	Readings & Resources . . . . .	94
5.1.4	Packages . . . . .	95
5.2	Defining Reliability . . . . .	95
5.2.1	Begins with Classical Test Theory (CTT) . . . . .	95
5.2.2	Why are we concerned with reliability? Error! . . . . .	95
5.2.3	The Reliability Coefficient . . . . .	96
5.3	Research Vignette . . . . .	97
5.4	A Parade of Reliability Coefficients . . . . .	102
5.4.1	Reliability Options for a Single Administration . . . . .	102
5.4.1.1	Split-half reliability . . . . .	102
5.4.1.2	Alpha coefficients . . . . .	106
5.4.1.3	Omega . . . . .	110
5.4.1.4	Some summary statements about reliability from single administrations . . . . .	116
5.4.2	Reliability Options for Two or more Administrations . . . . .	116
5.4.2.1	Test-retest of total scores . . . . .	116
5.4.2.2	Test-retest recap . . . . .	119
5.4.3	Interrater Reliability . . . . .	120
5.4.3.1	Cohen's kappa . . . . .	120
5.4.3.2	Intraclass correlation (ICC) . . . . .	122
5.5	What do we do with these coefficients? . . . . .	123
5.5.1	Corrections for attenuation . . . . .	123
5.5.2	Predicting true scores (and their CIs) . . . . .	123
5.5.3	How do I keep it all straight? . . . . .	124
5.6	Practice Problems . . . . .	124
5.6.1	Problem #1: Play around with this simulation. . . . .	124
5.6.2	Problem #2: Use the data from the live ReCentering Psych Stats survey. . . . .	125
5.6.3	Problem #3: Try something entirely new. . . . .	126
5.6.4	Grading Rubric . . . . .	126
5.7	Homeworked Example . . . . .	126
5.7.1	Check and, if needed, format the data . . . . .	127
5.7.2	Calculate and report the alpha coefficient for a total scale score and subscales (if the scale has them) . . . . .	128
5.7.3	Subscale alphas . . . . .	130

5.7.4 Calculate and report t and h . . . . .	133
5.7.5 With these two determine what proportion of the variance is due to all the factors, error, and g. . . . .	137
5.7.6 Calculate total and subscale scores. . . . .	137
5.7.7 Describe other reliability estimates that would be appropriate for the measure you are evaluating. . . . .	137
<b>6 Item Analysis for Educational Achievement Tests (Exams)</b>	<b>139</b>
6.1 Navigating this Lesson . . . . .	139
6.1.1 Learning Objectives . . . . .	139
6.1.2 Planning for Practice . . . . .	140
6.1.3 Readings & Resources . . . . .	140
6.1.4 Packages . . . . .	140
6.2 Research Vignette . . . . .	140
6.3 Item Analysis in the Educational/Achievement Context . . . . .	140
6.3.1 And now a quiz! Please take it. . . . .	141
6.4 Item Difficulty . . . . .	143
6.4.1 Percent passing . . . . .	143
6.4.2 Several factors prevent .50 from being the ideal difficulty level . . . . .	143
6.5 Item Discrimination . . . . .	144
6.5.1 Index of Discrimination . . . . .	145
6.5.2 Application of Item Difficulty and Discrimination . . . . .	145
6.6 In the <i>psych</i> Package . . . . .	150
6.6.1 A Mini-Introduction to IRT . . . . .	155
6.7 Closing Thoughts on Developing Measures in the Education/Achievement Context .	163
6.8 Practice Problems . . . . .	164
<b>7 Item Analysis for Likert Type Scale Construction</b>	<b>165</b>
7.1 Navigating this Lesson . . . . .	165
7.1.1 Learning Objectives . . . . .	165
7.1.2 Planning for Practice . . . . .	166
7.1.3 Readings & Resources . . . . .	166
7.1.4 Packages . . . . .	167
7.2 Introducing Item Analysis for Survey Development . . . . .	167
7.2.1 Workflow for Item Analysis . . . . .	167

7.3	Research Vignette . . . . .	168
7.4	Step I: Corrected item-total correlations . . . . .	172
7.4.1	Data Prep . . . . .	173
7.4.2	Calculating Item-Total Correlation Coefficients . . . . .	175
7.5	Step II: Correlating Items with Other Scale Totals . . . . .	179
7.6	Step III: Interpreting and Writing up the Results . . . . .	181
7.7	A Conversation with Dr. Szymanski . . . . .	182
7.8	Practice Problems . . . . .	183
7.8.1	Problem #1: Play around with this simulation. . . . .	183
7.8.2	Problem #2: Use raw data from the ReCentering Psych Stats survey on Qualtrics. . . . .	183
7.8.3	Problem #3: Try something entirely new. . . . .	184
7.8.4	Grading Rubric . . . . .	184
7.9	Bonus Reel: . . . . .	185
7.10	Homeworked Example . . . . .	186
7.10.1	Check and, if needed, format and score data . . . . .	187
7.10.2	Report alpha coefficients and average inter-item correlations for the total and subscales . . . . .	188
7.10.3	Produce and interpret corrected item-total correlations for total and subscales, separately . . . . .	188
7.10.3.1	All Course Evaluation Items . . . . .	188
7.10.3.2	Valued-by-Student Subscale . . . . .	190
7.10.3.3	Traditional Pedagogy Items . . . . .	191
7.10.3.4	Socially Responsive Pedagogy Items . . . . .	193
7.10.4	Produce and interpret correlations between the individual items of a given subscale and the subscale scores of all other subscales . . . . .	194
7.10.4.1	Valued-by-the-Student Items . . . . .	194
7.10.5	Traditional Pedagogy Items . . . . .	197
7.10.5.1	Socially Responsive Pedagogy Items . . . . .	200
7.10.6	APA style results section with table . . . . .	202
7.10.7	Explanation to grader . . . . .	202

<b>8 Principal Components Analysis</b>	<b>205</b>
8.1 Navigating this Lesson . . . . .	205
8.1.1 Learning Objectives . . . . .	205
8.1.2 Planning for Practice . . . . .	206
8.1.3 Readings & Resources . . . . .	206
8.1.4 Packages . . . . .	207
8.2 Exploratory Principal Components Analysis . . . . .	207
8.2.1 Some Framing Ideas (in very lay terms) . . . . .	207
8.3 PCA Workflow . . . . .	208
8.4 Research Vignette . . . . .	210
8.5 Working the Vignette . . . . .	213
8.5.1 Three Diagnostic Tests to Evaluate the Appropriateness of the Data for Component-or-Factor Analysis . . . . .	215
8.5.1.1 Is my sample adequate for PCA? . . . . .	215
8.5.1.2 Are the correlations among the variables large enough to be analyzed? . . . . .	217
8.5.1.3 Is there multicollinearity or singularity in my data? . . . . .	217
8.5.1.4 APA Style Summary So Far . . . . .	218
8.5.2 Principal Components Analysis . . . . .	218
8.5.3 Specifying the Number of Components . . . . .	223
8.5.3.1 Quick recap of how to evaluate the # of components we extracted . . . . .	232
8.5.4 Component Rotation . . . . .	233
8.5.4.1 Orthogonal rotation . . . . .	233
8.5.4.2 Oblique rotation . . . . .	238
8.5.5 Component Scores . . . . .	244
8.6 APA Style Results . . . . .	246
8.7 Back to the FutuRe: The relationship between PCA and item analysis . . . . .	247
8.7.1 Calculating and Extracting Item-Total Correlation Coefficients . . . . .	248
8.7.1.1 Corrected item-total correlations from the <i>psych::alpha()</i> . . . . .	248
8.7.1.2 Correlating items with other subscale totals . . . . .	254
8.7.1.3 Extracting values, binding them together, and joining the files . . . . .	258
8.7.1.4 Interpreting the result . . . . .	261
8.8 Practice Problems . . . . .	262
8.8.1 Problem #1: Play around with this simulation. . . . .	262
8.8.2 Problem #2: Conduct a PCA with another simulated set of data in the OER. . . . .	263

8.8.3	Problem #3: Try something entirely new. . . . .	263
8.8.4	Grading Rubric . . . . .	263
8.9	Homeworked Example . . . . .	264
8.9.1	Check and, if needed, format data . . . . .	264
8.9.2	Conduct and interpret the three diagnostic tests to determine if PCA is appropriate as an analysis (KMO, Bartlett's, determinant) . . . . .	265
8.9.2.1	KMO . . . . .	265
8.9.2.2	Bartlett's . . . . .	266
8.9.2.3	Determinant . . . . .	266
8.9.3	Determine how many components to extract (e.g., scree plot, eigenvalues, theory) . . . . .	267
8.9.4	Conduct an orthogonal extraction and rotation with a minimum of two different factor extractions . . . . .	269
8.9.5	Conduct an oblique extraction and rotation with a minimum of two different factor extractions . . . . .	275
8.9.6	Determine which factor solution (e.g., orthogonal or oblique; which number of factors) you will suggest . . . . .	281
8.9.7	APA style results section with table and figure of one of the solutions . . . . .	281
8.9.8	Explanation to grader . . . . .	284
<b>9</b>	<b>Principal Axis Factoring</b>	<b>285</b>
9.1	Navigating this Lesson . . . . .	285
9.1.1	Learning Objectives . . . . .	285
9.1.2	Planning for Practice . . . . .	286
9.1.3	Readings & Resources . . . . .	286
9.1.4	Packages . . . . .	287
9.2	Exploratory Factor Analysis (with a quick contrast to PCA) . . . . .	287
9.3	PAF Workflow . . . . .	288
9.4	Research Vignette . . . . .	290
9.5	Working the Vignette . . . . .	293
9.5.1	Data Prep . . . . .	293
9.5.1.1	Three Diagnostic Tests to Evaluate the Appropriateness of the Data for Factor (or Component))Analysis . . . . .	295
9.5.1.2	APA Style Summary So Far . . . . .	297
9.5.2	Principal Axis Factoring (PAF) . . . . .	298
9.5.2.1	Specifying the number of factors . . . . .	303

9.5.2.2	Quick recap of how to evaluate the # of factors we extracted . . . . .	313
9.5.3	Factor Rotation . . . . .	313
9.5.3.1	Orthogonal rotation . . . . .	314
9.5.3.2	Oblique rotation . . . . .	320
9.5.4	Factor Scores . . . . .	325
9.6	APA Style Results . . . . .	328
9.6.1	Comparing FA and PCA . . . . .	329
9.7	Going Back to the Future: What, then, is Omega? . . . . .	330
9.8	Comparing PFA to Item Analysis and PCA . . . . .	336
9.9	Practice Problems . . . . .	338
9.9.1	Problem #1: Play around with this simulation. . . . .	339
9.9.2	Problem #2: Conduct a PAF with another simulated set of data in the OER.	339
9.9.3	Problem #3: Try something entirely new. . . . .	339
9.9.4	Grading Rubric . . . . .	339
9.10	Homeworked Example . . . . .	340
9.10.1	Check and, if needed, format data . . . . .	341
9.10.2	Conduct and interpret the three diagnostic tests to determine if PAF is appropriate as an analysis (KMO, Bartlett's, determinant) . . . . .	341
9.10.2.1	KMO . . . . .	341
9.10.2.2	Bartlett's . . . . .	342
9.10.2.3	Determinant . . . . .	343
9.10.3	Determine how many components to extract (e.g., scree plot, eigenvalues, theory) . . . . .	343
9.10.4	Conduct an orthogonal rotation with a minimum of two different factor extractions . . . . .	346
9.10.5	Conduct an oblique rotation with a minimum of two different factor extractions	352
9.10.6	Determine which factor solution (e.g., orthogonal or oblique; which number of factors) you will suggest . . . . .	358
9.10.7	APA style results section with table and figure of one of the solutions . . . . .	359
9.10.8	Explanation to grader . . . . .	361
<b>10 CFA: First Order Models</b>		<b>365</b>
10.1	Navigating this Lesson . . . . .	365
10.1.1	Learning Objectives . . . . .	365
10.1.2	Planning for Practice . . . . .	365

10.1.3 Readings & Resources . . . . .	366
10.1.4 Packages . . . . .	367
10.2 Two Broad Categories of Factor Analysis: Exploratory and Confirmatory . . . . .	367
10.2.1 Common to Both Exploratory and Confirmatory Approaches . . . . .	367
10.2.2 Differences between EFA and CFA . . . . .	368
10.2.3 On the relationship between EFA and CFA . . . . .	369
10.3 Exploring a Standard CFA Model . . . . .	370
10.3.1 Model Identification for CFA . . . . .	371
10.3.2 Selecting Indicators/Items for a Reflective Measurement . . . . .	371
10.4 CFA Workflow . . . . .	372
10.4.1 CFA in <i>lavaan</i> Requires Fluency with the Syntax . . . . .	373
10.4.2 Differing Factor Structures . . . . .	374
10.5 Research Vignette . . . . .	376
10.5.1 Modeling the GRMSAAW as Unidimensional . . . . .	383
10.5.1.1 Model Test <i>User</i> Model: . . . . .	388
10.5.1.2 Model Test <i>Baseline</i> Model . . . . .	389
10.5.1.3 Incremental Fit Indices (User versus Baseline Models) . . . . .	389
10.5.1.4 Loglikelihood and Information Criteria . . . . .	390
10.5.1.5 Root Mean Square Error of Approximation . . . . .	390
10.5.1.6 Standardized Root Mean Square Residual . . . . .	390
10.5.1.7 APA Style Results from the Unidimensional model . . . . .	401
10.5.2 Modeling the GRMSAAW as a First-Order, 4-factor model . . . . .	402
10.5.2.1 Specifying and Running the Model . . . . .	402
10.5.2.2 Interpretation . . . . .	408
10.6 Model Comparison . . . . .	411
10.6.1 APA Results Section (so far...) . . . . .	413
10.7 A concluding thought . . . . .	414
10.8 Practice Problems . . . . .	415
10.8.1 Problem #1: Play around with this simulation. . . . .	415
10.8.2 Problem #2: Use simulated data from other lessons. . . . .	416
10.8.3 Problem #3: Try something entirely new. . . . .	416
10.8.4 Grading Rubric . . . . .	416
10.9 Homeworked Example . . . . .	416
10.9.1 Prepare data for CFA (items only df, reverse-scored) . . . . .	417

10.9.2 Specify and run a unidimensional model . . . . .	418
10.9.3 Specify and run a single-order model with correlated factors . . . . .	422
10.9.4 Narrate adequacy of fit with $\chi^2$ , CFI, RMSEA, SRMR (write a mini-results section) . . . . .	426
10.9.5 Compare model fit with $\chi^2\Delta$ , AIC, BIC . . . . .	427
10.9.6 APA style results with table(s) and figure . . . . .	427
10.9.7 Explanation to grader . . . . .	428
<b>11 CFA: Hierarchical and Nested Models</b>	<b>429</b>
11.1 Navigating this Lesson . . . . .	429
11.1.1 Learning Objectives . . . . .	429
11.1.2 Planning for Practice . . . . .	430
11.1.3 Packages . . . . .	431
11.2 CFA Workflow . . . . .	431
11.3 Another Look at Varying Factor Structures . . . . .	432
11.4 Revisiting Model Identification . . . . .	434
11.4.1 Identification Status . . . . .	434
11.4.2 Identification in Practice . . . . .	435
11.5 Research Vignette . . . . .	436
11.6 A Quick <i>lavaan</i> Syntax Recap . . . . .	443
11.7 Comparing and Tweaking Multidimensional First-Order Models . . . . .	444
11.8 An Uncorrelated Factors Model . . . . .	444
11.8.1 Specifying the Model . . . . .	444
11.8.2 Interpreting the Output . . . . .	450
11.8.3 Partial Write-up . . . . .	450
11.9 A Correlated Factors Model . . . . .	450
11.9.1 Nested Models . . . . .	450
11.9.2 Interpreting the Output . . . . .	456
11.9.3 Partial Write-up . . . . .	456
11.10 Model Respecification . . . . .	458
11.10.1 Respecifying with Correlated Errors . . . . .	459
11.10.1.1 Interpreting the Output . . . . .	463
11.10.2 Respecifying with Crossloadings . . . . .	465
11.10.2.1 Interpreting the Output . . . . .	469

11.11	Modeling the GRMSAAW as a Second-Order Structure . . . . .	473
11.11.1	Interpreting the Output . . . . .	478
11.11.2	Partial Write-up . . . . .	479
11.12	Modeling the GRMSAAW as a Bifactor Model . . . . .	480
11.12.1	Interpreting the Output . . . . .	488
11.12.2	Partial Write-up . . . . .	488
11.13	Another Look at Omega . . . . .	490
11.13.1	$\Omega_h$ for Bifactor Models . . . . .	490
11.13.2	$\omega_h$ for Second Order Models . . . . .	491
11.13.3	Partial Write-up . . . . .	492
11.14	Preparing an Overall APA Style Results Section . . . . .	492
11.15	A Conversation with Dr. Keum . . . . .	494
11.16	Practice Problems . . . . .	494
11.16.1	Problem #1: Play around with this simulation. . . . .	495
11.16.2	Problem #2: Use simulated data from other lessons. . . . .	495
11.16.3	Problem #3: Try something entirely new. . . . .	495
11.16.4	Grading Rubric . . . . .	495
11.17	Homeworked Example . . . . .	496
11.17.1	Prepare data for CFA (items only df, reverse-scored) . . . . .	497
11.17.2	Specify and run a unidimensional model . . . . .	497
11.17.3	Narrate adequacy of fit with $\chi^2$ , CFI, RMSEA, SRMR (write a mini-results section) . . . . .	501
11.17.4	Specify and run a single-order model with correlated factors . . . . .	501
11.17.5	Narrate adequacy of fit with $\chi^2$ , CFI, RMSEA, SRMR (write a mini-results section) . . . . .	506
11.17.6	Specify and run a second-order model . . . . .	506
11.17.7	Narrate adequacy of fit with $\chi^2$ , CFI, RMSEA, SRMR (write a mini-results section) . . . . .	510
11.17.8	Specify and run a bifactor model . . . . .	510
11.17.9	Narrate adequacy of fit with $\chi^2$ , CFI, RMSEA, SRMR (write a mini-results section) . . . . .	525
11.17.10	Compare model fit with $\chi^2\Delta$ , AIC, BIC . . . . .	525
11.17.11	Calculate omega hierarchical ( $\omega_h$ ) and omega-hierarchical-subscales ( $\omega_{h-ss}$ ) . . . . .	526
11.17.12	APA style results with table(s) and figure . . . . .	527
11.17.13	Explanation to grader . . . . .	528

CONTENTS	15
----------	----

<b>12 Invariance Testing</b>	<b>529</b>
12.1 Navigating this Lesson . . . . .	529
12.1.1 Learning Objectives . . . . .	529
12.1.2 Planning for Practice . . . . .	530
12.1.3 Readings & Resources . . . . .	530
12.1.4 Packages . . . . .	531
12.2 Invariance Testing (aka Multiple-Samples SEM or Multiple-Group CFA [MG-CFA])	531
12.2.1 Introducing the Topic and the Terminology . . . . .	531
12.2.2 Evaluation Strategies . . . . .	533
12.2.2.1 Free baseline approach . . . . .	533
12.2.2.2 Constrained baseline approach . . . . .	533
12.2.3 Invariance Testing Workflow . . . . .	535
12.2.4 Successive Gradations of Measurement Invariance . . . . .	535
12.2.4.1 Configural invariance . . . . .	535
12.2.4.2 Weak invariance . . . . .	536
12.2.4.3 Strong invariance** . . . . .	536
12.2.4.4 Strict invariance . . . . .	538
12.2.5 Tests for Model Comparison . . . . .	538
12.2.6 Partial measurement invariance . . . . .	539
12.3 Research Vignette . . . . .	539
12.4 Whole-Group and Baseline Analyses . . . . .	543
12.4.1 Whole Group CFA . . . . .	543
12.4.2 Interpreting the Output . . . . .	547
12.4.3 Partial Write-up . . . . .	547
12.4.4 Baseline Model when Severity = Mild . . . . .	548
12.4.5 Baseline Model when Severity = Severe . . . . .	552
12.5 Configural Invariance . . . . .	555
12.5.1 Interpreting the Output . . . . .	562
12.5.2 Partial Write-up . . . . .	563
12.6 Weak Invariance . . . . .	563
12.6.1 Interpreting the Output . . . . .	569
12.6.2 Partial Write-up . . . . .	569
12.7 Strong Invariance . . . . .	571
12.7.1 Interpreting the Output . . . . .	579

12.7.2 Partial Write-up . . . . .	580
12.8 Strict Invariance . . . . .	581
12.8.1 Interpreting the Output . . . . .	588
12.8.2 Partial Write-up . . . . .	589
12.9 Partial Measurement Invariance . . . . .	589
12.10 APA Style Write-up of the Results . . . . .	590
12.10.1 Measurement Invariance Across Disability Severity . . . . .	590
12.11 Practice Problems . . . . .	591
12.11.1 Problem #1: Play around with this simulation. . . . .	592
12.11.2 Problem #2: Adapt one of the simulated data sets. . . . .	592
12.11.3 Problem #3: Try something entirely new. . . . .	592
12.11.4 Grading Rubric . . . . .	592
12.12 Homeworked Example . . . . .	593
<b>13 Hybrid Models</b>	<b>629</b>
13.1 Navigating this Lesson . . . . .	630
13.1.1 Learning Objectives . . . . .	630
13.1.2 Planning for Practice . . . . .	630
13.1.3 Readings & Resources . . . . .	630
13.1.4 Packages . . . . .	631
13.2 Introducing the Statistic . . . . .	631
13.3 Research Vignette . . . . .	633
13.4 Importing and Preparing the Data . . . . .	634
13.4.1 Analyzing and Managing Missingness . . . . .	638
13.4.2 Assessing the Distributional Characteristics of the Data . . . . .	643
13.4.3 Preliminary Analyses . . . . .	646
13.4.3.1 Internal Consistency Coefficients . . . . .	646
13.4.3.2 Means, SDs, r-matrix . . . . .	649
13.4.4 Summary of Data Preparation . . . . .	650
13.5 The Measurement Model: Specification and Evaluation . . . . .	651
13.5.0.1 Managing missing data with FIML . . . . .	652
13.5.0.2 Interpreting the Output . . . . .	656
13.6 The Structural Model: Specification and Evaluation . . . . .	660
13.6.1 Model Identification . . . . .	661

13.6.1.1	Model identification for the overall SEM . . . . .	662
13.6.1.2	Model identification for the structural portion of the model . . . . .	663
13.6.2	Specifying and Evaluating the Structural Model . . . . .	664
13.6.3	APA Style Write-up of the Results . . . . .	675
13.6.3.1	Preliminary Analyses . . . . .	675
13.6.3.2	Primary Analyses . . . . .	675
13.7	Practice Problems . . . . .	675
13.7.1	Problem #1: Download a fresh sample . . . . .	676
13.7.2	Problem #2: Swap one or more of the variables . . . . .	676
13.7.3	Problem #3: Try something entirely new. . . . .	676
13.7.4	Grading Rubric . . . . .	676
13.8	Homeworked Example . . . . .	676
13.8.0.1	Write up of preliminary analyses . . . . .	689
13.8.1	APA style results with table(s) and figure. . . . .	701
<b>14</b>	<b>Additional Simulations</b>	<b>703</b>
14.1	iBelong Scale . . . . .	704
14.2	Identity Threat . . . . .	709
14.3	Anti-Racism Behavioral Inventory . . . . .	715



# BOOK COVER

This open education resource (OER) is available in two formats:

- Formatted as an [html book](#) via GitHub Pages
- As a [PDF](#)

All materials used in creating this OER are available at its [GitHub repo](#).

As a perpetually-in-progress, open education resource, feedback is always welcome. This IRB-approved (SPU IRB #202102010R, no expiration) [Qualtrics-hosted survey](#) includes formal rating scales, open-ended text boxes, and a portal for uploading attachments (e.g., marked up PDFs). You are welcome to complete only the portions that are relevant to you.

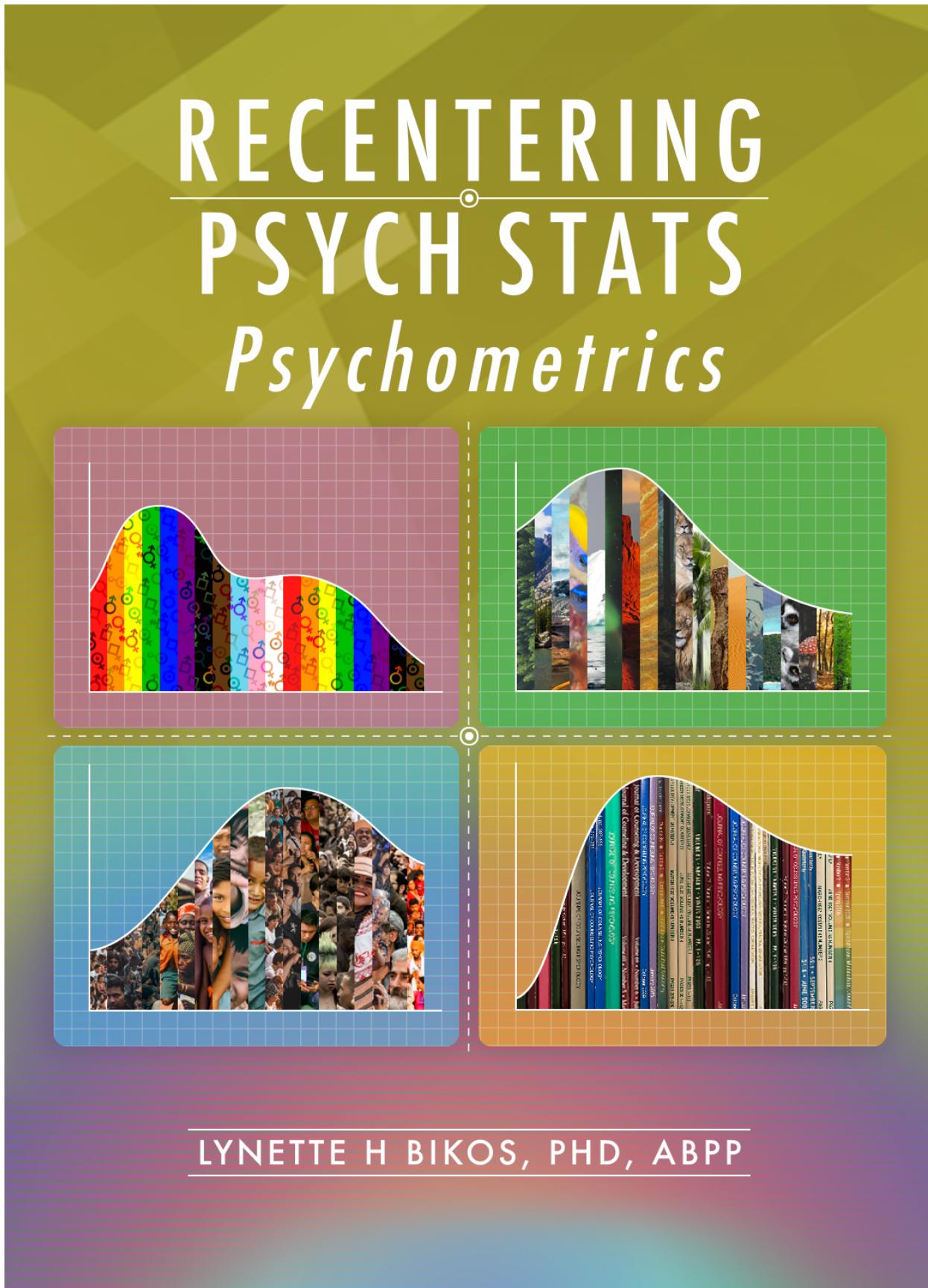


Figure 1: An image of the book cover. It includes four quadrants of non-normal distributions representing gender, race/ethnicity, sustainability/global concerns, and journal articles.

# PREFACE

If you are viewing this document, you should know that this is a book-in-progress. Early drafts are released for the purpose teaching my classes and gaining formative feedback from a host of stakeholders. The document was last updated on 12 May 2024. Emerging volumes on other statistics are posted on the [ReCentering Psych Stats](#) page at my research team's website.

## [Screencasted Lecture Link](#)

To *center* a variable in regression means to set its value at zero and interpret all other values in relation to this reference point. Regarding race and gender, researchers often center male and White at zero. Further, it is typical that research vignettes in statistics textbooks are similarly seated in a White, Western (frequently U.S.), heteronormative, framework. The purpose of this project is to create a set of open educational resources (OER) appropriate for doctoral and post-doctoral training that contribute to a socially responsive pedagogy – that is, it contributes to justice, equity, diversity, and inclusion.

Statistics training in doctoral programs are frequently taught with fee-for-use programs (e.g., SPSS/AMOS, SAS, MPlus) that may not be readily available to the post-doctoral professional. In recent years, there has been an increase and improvement in R packages (e.g., *psych*, *lavaan*) used for in analyses common to psychological research. Correspondingly, many graduate programs are transitioning to statistics training in R (free and open source). This is a challenge for post-doctoral psychologists who were trained with other software. This OER will offer statistics training with R and be freely available (specifically in a GitHub repository and posted through GitHub Pages) under a Creative Commons Attribution-NonCommercial-ShareAlike license [CC BY-NC-SA 4.0].

Training models for doctoral programs in health service psychology are commonly scholar-practitioner, scientist-practitioner, or clinical-scientist. An emerging model, the *scientist-practitioner-advocacy* training model, incorporates social justice advocacy so that graduates are equipped to recognize and address the sociocultural context of oppression and unjust distribution of resources and opportunities [Mallinckrodt et al., 2014]. In statistics textbooks, the use of research vignettes engages the learner around a tangible scenario for identifying independent variables, dependent variables, covariates, and potential mechanisms of change. Many students recall examples in Field's [2012] popular statistics text: Viagra to teach one-way ANOVA, beer goggles for two-way ANOVA, and bushtucker for repeated measures. What if the research vignettes were more socially responsive?

In this OER, research vignettes will be from recently published articles where:

- the author's identity is from a group where scholarship is historically marginalized (e.g., BIPOC, LGBTQ+, LMIC [low-middle income countries]),

- the research is responsive to issues of justice, equity, inclusion, diversity,
- the lesson's statistic is used in the article, and
- there is sufficient information in the article to simulate the data for the chapter example(s) and practice problem(s); or it is publicly available.

In training for multicultural competence, the saying, “A fish doesn’t know that it’s wet” is often used to convey the notion that we are often unaware of our own cultural characteristics. In recent months and years, there has been an increased awakening to the institutional and systemic racism that our systems are perpetuating. Queuing from the water metaphor, I am hopeful that a text that is recentered in the ways I have described can contribute to *changing the water* in higher education and in the profession of psychology.

## Copyright with Open Access

This book is published under a [Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License](#). This means that this book can be reused, remixed, retained, revised, and redistributed (including commercially) as long as appropriate credit is given to the authors. If you remix, or modify the original version of this open textbook, you must redistribute all versions of this open textbook under the same license: CC BY-SA 4.0.

A [GitHub open-source repository](#) contains all of the text and source code for the book, including data and images.

# ACKNOWLEDGEMENTS

As a doctoral student at the University of Kansas (1992-1996), I learned that “a foreign language” was a graduation requirement. *Please note that as one who studies the intersections of global, vocational, and sustainable psychology, I regret that I do not have language skills beyond English.* This could have been met with credit from high school, but my rural, mid-Missouri high school did not offer such classes. This requirement would have typically been met with courses taken during an undergraduate program – but my non-teaching degree in the University of Missouri’s School of Education was exempt from this. The requirement could have also been met with a computer language (FORTRAN, C++) – but I did not have any of those either. There was a tiny footnote on my doctoral degree plan that indicated that a 2-credit course, “SPSS for Windows” would substitute for the language requirement. Given that it was taught by my one of my favorite professors, I readily signed up. As it turns out, Samuel B. Green, PhD, was using the course to draft chapters in the textbook [[Green and Salkind, 2017](#)] that has been so helpful for so many. Unfortunately, Drs. Green (1947 - 2018) and Salkind (1947 - 2017) are no longer with us. I have worn out numerous versions of their text. Another favorite text of mine has been Dr. Barbara Byrne’s [[2016b](#)], “Structural Equation Modeling with AMOS.” I loved the way she worked through each problem and paired it with a published journal article, so that the user could see how the statistical evaluation fit within the larger project/article. I took my tea-stained text with me to a workshop she taught at APA and was proud of the signature she added to it. Dr. Byrne created SEM texts for a number of statistical programs (e.g., LISREL, EQS, MPlus). As I was learning R, I wrote Dr. Byrne, asking if she had an edition teaching SEM/CFA with R. She promptly wrote back, saying that she did not have the bandwidth to learn a new statistics package. We lost Dr. Byrne in December 2020. I am so grateful to these role models for their contributions to my statistical training. I am also grateful for the doctoral students who have taken my courses and are continuing to provide input for how to improve the materials.

The inspiration for training materials that re\*center statistics and research methods came from the [Academics for Black Survival and Wellness Initiative](#). This project, co-founded by Della V. Mosley, Ph.D., and Pearis L. Bellamy, M.S., made clear the necessity and urgency for change in higher education and the profession of psychology.

At very practical levels, I am indebted to SPU’s Library, and more specifically, SPU’s Education, Technology, and Media Department. Assistant Dean for Instructional Design and Emerging Technologies, R. John Robertson, MSc, MCS, has offered unlimited consultation, support, and connection. Senior Instructional Designer in Graphics & Illustrations, Dominic Wilkinson, designed the logo and book cover. Psychology and Scholarly Communications Librarian, Kristin Hoffman, MLIS, has provided consultation on topics ranging from OERS to citations. I am also indebted to Associate Vice President, Teaching and Learning at Kwantlen Polytechnic University, Rajiv Jhangiani, PhD. Dr. Jhangiani’s text [[2019](#)] was the first OER I ever used, and I was grateful for

his encouraging conversation.

Throughout the ReCentering Psych Stats series, students (both paid and volunteer) have served as editorial assistants in copyediting the text and correcting the text captions. In the Psychometrics volume, these have included Layla Hakim and Kelly Ng.

Financial support for this project has been provided the following:

- *Call to Action on Equity, Inclusion, Diversity, Justice, and Social Responsivity Request for Proposals* grant from the Association of Psychology Postdoctoral and Internship Centers (2021-2022).
- *Diversity Seed Grant*, Office of Inclusive Excellence and Advisory Council for Diversity and Reconciliation (ACDR), Seattle Pacific University.
- *ETM Open Textbook & OER Development Funding*, Office of Education, Technology, & Media, Seattle Pacific University.

# Chapter 1

## Introduction

[Screencasted Lecture Link](#)

### 1.1 What to expect in each chapter

This textbook is intended as *applied*, in that a primary goal is to help the scientist-practitioner-advocate use a variety of statistics in research problems and *writing them up* for a program evaluation, dissertation, or journal article. In support of that goal, I try to provide just enough conceptual information so that the researcher can select the appropriate statistic (i.e., distinguishing between when ANOVA is appropriate and when regression is appropriate) and assign variables to their proper role (e.g., covariate, moderator, mediator).

This conceptual approach does include occasional, step-by-step, *hand-calculations* (only we calculate them arithmetically in R) to provide a *visceral feeling* of what is happening within the statistical algorithm that may be invisible to the researcher. Additionally, the conceptual review includes a review of the assumptions about the characteristics of the data and research design that are required for the statistic. Statistics can be daunting, so I have worked hard to establish a *workflow* through each analysis. When possible, I include a flowchart that is referenced frequently in each chapter and assists the researcher keep track of their place in the many steps and choices that accompany even the simplest of analyses.

As with many statistics texts, each chapter includes a *research vignette*. Somewhat unique to this resource is that the vignettes are selected from recently published articles. Each vignette is chosen with the intent to meet as many of the following criteria as possible:

- the statistic that is the focus of the chapter was properly used in the article,
- the author's identity is from a group where scholarship is historically marginalized (e.g., BIPOC, LGBTQ+, LMIC [low middle income countries]),
- the research has a justice, equity, inclusion, diversity, and social responsibility focus and will contribute positively to a social justice pedagogy, and
- the data is available in a repository or there is sufficient information in the article to simulate the data for the chapter example(s) and practice problem(s).

In each chapter we employ *R* packages that will efficiently calculate the statistic and the dashboard of metrics (e.g., effect sizes, confidence intervals) that are typically reported in psychological science.

## 1.2 Strategies for Accessing and Using this OER

There are a number of ways you can access this resource. You may wish to try several strategies and then select which works best for you. I demonstrate these in the screencast that accompanies this chapter.

1. Simply follow along in the .html formatted document that is available on via GitHub Pages, and then
  - open a fresh .rmd file of your own, copying (or retyping) the script and running it
2. Locate the original documents at the [GitHub repository](#). You can:
  - open them to simply take note of the “behind the scenes” script
  - copy/download individual documents that are of interest to you
  - fork a copy of the entire project to your own GitHub site and further download it (in its entirety) to your personal workspace. The [GitHub Desktop app](#) makes this easy!
3. Listen to the accompanying lectures (I think sound best when the speed is 1.75). The lectures are being recorded in Panopto and should include the closed captioning.
4. Provide feedback to me! If you fork a copy to your own GitHub repository, you can:
  - open up an editing tool and mark up the document with your edits,
  - start a discussion by leaving comments/questions, and then
  - sending them back to me by committing and saving. I get an e-mail notifying me of this action. I can then review (accepting or rejecting) them and, if a discussion is appropriate, reply back to you.

## 1.3 If You are New to R

R can be overwhelming. Jumping right into advanced statistics might not be the easiest way to start. However, in these chapters, I provide complete code for every step of the process, starting with uploading the data. To help explain what R script is doing, I sometimes write it in the chapter text; sometimes leave hashtags-comments in the chunks; and, particularly in the accompanying screen casted lectures, try to take time to narrate what the R script is doing.

I've found that, somewhere on the internet, there's almost always a solution to what I'm trying to do. I am frequently stuck and stumped and have spent hours searching the internet for even the tiniest of things. When you watch my videos, you may notice that in my R studio, there is a “scRiptuRe” file. I takes notes on the solutions and scripts here – using keywords that are meaningful to me so that when I need to repeat the task, I can hopefully search my own prior solutions and find a fix or a hint.

### 1.3.1 Base R

The base program is free and is available here: <https://www.r-project.org/>

Because R is already on my machine (and because the instructions are sufficient), I will not walk through the installation, but I will point out a few things.

- Follow the instructions for your operating system (Mac, Windows, Linux)
- The “cran” (I think “cranium”) is the *Comprehensive R Archive Network*. In order for R to run on your computer, you have to choose a location. Because proximity is somewhat related to processing speed, select one that is geographically “close to you.”
- You will see the results of this download on your desktop (or elsewhere if you chose to not have it appear there) but you won’t ever use R through this platform.

### 1.3.2 R Studio

*R Studio* is the desktop application I work in R. It’s a separate download. Choose the free, desktop, option that is appropriate for your operating system: <https://www.rstudio.com/products/RStudio/>

- The **upper right window** includes several tabs:
  - Environment: it lists the *objects* that are available to you (e.g., dataframes)
- The **lower right window** has a number of helpful tab:
  - Files: Displays the file structure in your computer’s environment. Make it a practice to (a) organize your work in small folders and (b) navigating to that small folder that is holding your project when you are working on it.
  - Packages: Lists the packages that have been installed. If you navigate to it, you can see if it is “on.” You can also access information about the package (e.g., available functions, examples of script used with the package) in this menu. This information opens in the Help window.
  - Viewer and Plots are helpful, later, when we can simultaneously look at our output and still work on our script.
- The **primary window** is where we work:
  - R Studio runs in the background (i.e., in the console). Occasionally, I can find useful troubleshooting information here.
  - More commonly, I open my R Markdown document so that it takes the whole screen and I work directly, right here.
- **R Markdown** is the way that many analysts write *script*, conduct analyses, and even write up results. These are saved as .rmd files.
  - In R Studio, open an R Markdown document through File/New File/R Markdown
  - Specify the details of your document (title, author, desired output)
  - In a separate step, SAVE this document [File/Save] into a NEW FILE FOLDER that will contain anything else you need for your project (e.g., the data).
  - *Packages* are at the heart of working in R. Installing and activating packages require writing script.

### 1.3.3 R Hygiene

Many initial problems in R can be solved with good R hygiene. Here are some suggestions for basic practices. It can be tempting to “skip this.” However, in the first few weeks of class, these are the solutions I am presenting to my students.

### 1.3.3.1 Everything is documented in the .rmd file

Although others do it differently, everything is in my .rmd file. That is, for uploading data and opening packages I write the code in my .rmd file. Why? Because when I read about what I did hours or years later, I have a permanent record of very critical things like (a) where my data is located, (b) what version I was using, and (c) what package was associated with the functions.

### 1.3.3.2 File organization

File organization is a critical key to this:

- Create a project file folder.
- Put the data file in it.
- Open an R Markdown file.
- Save it in the same file folder.
- When your data and .rmd files are in the same folder (not your desktop, but a shared folder), they can be connected.

### 1.3.3.3 Chunks

The R Markdown document is an incredible tool for integrating text, tables, and analyses. This entire OER is written in R Markdown. A central feature of this is “chunks.”

The easiest way to insert a chunk is to use the INSERT/R command at the top of this editor box. You can also insert a chunk with the keyboard shortcut: CTRL/ALT/i

“Chunks” start and end with those three tic marks and will show up in a shaded box, like this:

```
# hashtags let me write comments to remind myself what I did here I
# am simply demonstrating arithmetic (but I would normally be running
# code)
2021 - 1966
```

```
## [1] 55
```

Each chunk must open and close. If one or more of your tic marks get deleted, your chunk won’t be read as such, and your script will not run. The only thing in the chunks should be script for running R; you can hashtag-out script, so it won’t run.

Although unnecessary, you can add a brief title for the chunk in the opening row, after the “r.” These create something of a table of contents of all the chunks – making it easier to find what you did. You can access them in the “Chunks” tab at the bottom left of R Studio. If you wish to knit a document, you cannot have identical chunk titles.

You can put almost anything you want in the space outside of tics. Syntax for simple formatting in the text areas (e.g., using italics, making headings, bold, etc.) is found here: [https://rmarkdown.rstudio.com/authoring\\_basics.html](https://rmarkdown.rstudio.com/authoring_basics.html)

### 1.3.3.4 Packages

As scientist-practitioners (and not coders), we will rely on *packages* to do our work for us. At first you may feel overwhelmed about the large number of packages that are available. Soon, though, you will become accustomed to the ones most applicable to our work (e.g., psych, tidyverse, lavaan, apaTables).

Researchers treat packages differently. In these lectures, I list all the packages we will use in an opening chunk that asks R to check to see if the package is installed, and if not, installs it.

```
if (!require(psych)) {
  install.packages("psych")
}
```

```
## Loading required package: psych
```

To make a package operable, you need to open it through the library. This process must be repeated each time you restart R. I don't open the package (through the "library(package\_name)") command until it is time to use it. Especially for new users, I think it's important to connect the functions with the specific packages.

```
# install.packages ('psych')
library(psych)
```

If you type in your own "install.packages" code, hashtag it out once it's been installed. It is problematic to continue to re-run this code.

### 1.3.3.5 Knitting

An incredible feature of R Markdown is its capacity to *knit* to HTML, PowerPoint, or word. If you access the .rmd files for this OER, you can use annotate or revise them to suit your purposes. If you redistribute them, though, please honor the Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License with a citation.

## 1.3.4 tRoubleshooting in R maRkdown

Hiccups are normal. Here are some ideas that I have found useful in getting unstuck.

- In an R script, you must have everything in order – Every. Single. Time.
  - All the packages must be in your library and activated; if you restart R, you need to reload each package.
  - If you open an .rmd file and want a boxplot, you cannot just scroll down to that script. You need to run any *prerequisite* script (like loading the package, importing data, putting the data in the global environment, etc.)
  - Do you feel lost? clear your global environment (broom) and start at the top of the R script. Frequent, fresh starts are good.

- Your .rmd file and your data need to be stored in the same file folder. These should be separate for separate projects, no matter how small.
- Type any warnings you get into a search engine. Odds are, you'll get some decent hints in a manner of seconds. Especially at first, these are common errors:
  - The package isn't loaded (if you restarted R, you need to reload your packages)
  - The .rmd file has been saved yet, or isn't saved in the same folder as the data
  - Errors of punctuation or spelling
- Restart R (it's quick – not like restarting your computer)
- If you receive an error indicating that a function isn't working or recognized, and you have loaded the package, type the name of the package in front of the function with two colons (e.g., psych::describe(df)). If multiple packages are loaded with functions that have the same name, R can get confused.

### 1.3.5 stRategies for success

- Engage with R, but don't let it overwhelm you.
  - The *mechanical is also the conceptual*. Especially when it is *simpler*, do try to retype the script into your own .rmd file and run it. Track down the errors you are making and fix them.
  - If this stresses you out, move to simply copying the code into the .rmd file and running it. If you continue to have errors, you may have violated one of the best practices above (Is the package loaded? Are the data and .rmd files in the same place? Is all the prerequisite script run?).
  - Still overwhelmed? Keep moving forward by downloading a copy of the .rmd file that accompanies any given chapter and just “run it along” with the lecture. Spend your mental power trying to understand what each piece does. Then select a practice problem that is appropriate for your next level of growth.
- Copy script that works elsewhere and replace it with your datafile, variables, etc.
- The learning curve is steep, but not impossible. Gladwell[2008] reminds us that it takes about 10,000 hours to get GREAT at something (2,000 to get reasonably competent). Practice. Practice. Practice.
- Updates to R, R Studio, and the packages are NECESSARY, but can also be problematic. It could very well be that updates cause programs/script to fail (e.g., “X has been deprecated for version X.XX”). Moreover, this very well could have happened between my distribution of these resources and your attempt to use it. My personal practice is to update R, R Studio, and the packages a week or two before each academic term.
- Embrace your downward dog. Also, walk away, then come back.

### 1.3.6 Resources for getting staRted

R for Data Science: <https://r4ds.had.co.nz/>

R Cookbook: <http://shop.oreilly.com/product/9780596809164.do>

R Markdown homepage with tutorials: <https://rmarkdown.rstudio.com/index.html>

R has cheat sheets for everything, here's one for R Markdown: <https://www.rstudio.com/wp-content/uploads/2015/02/rmarkdown-cheatsheet.pdf>

R Markdown Reference guide: <https://www.rstudio.com/wp-content/uploads/2015/03/rmarkdown-reference.pdf>

Using R Markdown for writing reproducible scientific papers: <https://libscie.github.io/rmarkdown-workshop/handout.html>

LaTeX equation editor: <https://www.codecogs.com/latex/eqneditor.php>



# Chapter 2

## Questionnaire Construction: The Fundamentals

### [Screencasted Lecture Link](#)

The focus of this chapter is on the technical issues of constructing a survey. I found this lesson to be more of a struggle to prepare than I expected. Why? There is a great deal of lore about what increases response rates and participation. Yet, research over the years, has both supported, contradicted, or not addressed these claims. One example is where to include “sensitive items.” Historically, textbook authors have recommended that these should come last so that respondents would be engaged in the process and be more willing to complete the survey [Krathwohl, 2009, Rowley, 2014]. Yet, research has shown that this has not held up in employee groups [Roberson and Sundstrom, 1990] nor among members of the National Association of Social Workers [Robert G. Green et al., 2000].

Given these contradictions, this lecture starts with the overall structure of a survey. The core of the lecture focuses on recent, evidence-based support for item-level decisions. I briefly discuss construct-specific guidance and discuss specific considerations for the on-line environment. I then close by addressing some of the decisions that I routinely make in survey construction and provide my rationale for why. Because this lesson occurs at the beginning of a text on psychometrics – this “skips over and around” reliability and validity. These important issues will be addressed in subsequent lessons.

### 2.1 Navigating this Lesson

There is just under one hour of lecture.

While the majority of R objects and data you will need are created within the R script that sources the chapter, there are a few resources that cannot be created from within the R framework. Additionally, sometimes links fail. All original materials are provided at the [GitHub site](#) that hosts the book. More detailed guidelines for ways to access all these materials are provided in the OER’s [introduction](#)

### 2.1.1 Learning Objectives

Focusing on this week's materials, make sure you can:

- Outline the overall structure/components of a questionnaire,
- Articulate test construction myths (e.g., location of sensitive items, “requirement” to have reverse scored items) and their evidence-based solutions (when they have them)
- List elements to consider when the questionnaire is administered online

### 2.1.2 Planning for Practice

This is a two-part lesson on questionnaire construction. After the second lesson, a detailed suggestion for practice will be provided that lists criteria for creating and piloting a survey of your own.

### 2.1.3 Readings & Resources

In preparing this chapter, I drew heavily from the following resource(s). Other resources are cited (when possible, linked) in the text with complete citations in the reference list.

- Chyung, S. Y., Roberts, K., Swanson, I., & Hankinson, A. (2017). Evidence-Based Survey Design: The Use of a Midpoint on the Likert Scale. *Performance Improvement*, 56(10), 15–23. <https://doi.org/10.1002/pfi.21727>
- Chyung, S. Y., Barkin, J. R., & Shamsy, J. A. (2018a). Evidence-Based Survey Design: The Use of Negatively Worded Items in Surveys. *Performance Improvement*, 57(3), 16–25. <https://doi.org/10.1002/pfi.21749>
- Chung, S. Y., Kennedy, M., & Campbell, I (2018b). Evidence-based survey design: The use of ascending or descending order of Likert-type response options. *Performance Improvement*, 57(9), 9-16. <https://doi.org/10.1002/pfi.21800>
- Chyung, S. Y., Swanson, I., Roberts, K., & Hankinson, A. (2018c). Evidence-Based Survey Design: The Use of Continuous Rating Scales in Surveys. *Performance Improvement*, 57(5), 38–48. <https://doi.org/10.1002/pfi.21763>
  - Finding the Chyung et al. series was like finding a pot of gold! They provide empirical support for guiding choices about survey construction. And they are current! If you don't have time to read them in detail, I recommend you scan them and archive them for future reference.

## 2.2 Components of the Questionnaire

Let's start by examining the components of a questionnaire and the general guidelines for their construction[[Colton and Covert, 2015](#), [Pershing and Pershing, 2001](#)]:

### Title

- reflect the content of the instrument

- be concisely worded
- be written in language easily understood by the respondents
- should not be offensive or off-putting
- should be formatted clearly at the top/beginning of the document

### **Introductory Statement**

- include a brief summary of the instrument's purpose
- contain an appropriate statement concerning the confidentiality of the respondent's information (informed consent)
- be motivating such that respondents are inspired/willing to complete the items
- specify the approximate amount of time required to complete the instrument

### **Directions**

- complete, unambiguous, concise
- written at a language level appropriate to the respondents
- tell the respondents how to return the instrument once they have completed it (surprisingly, in Qualtrics, this is also important; submission requires hitting that last little “→”)

### **Items**

- discussed throughout this textbook

### **Closing Statement**

- thank the participants for their participation
- remind participants that their information is valuable and perhaps remind about
  - next steps or follow-up
  - confidentiality

### **Overall Structure/Look**

- should be coherent with an easy-to-follow layout
- professional appearance
  - not crowded, plenty of white space
  - avoiding a “slick look”
  - numbering and headings to provide a sense of progress
  - breaks between every 4-6 questions (or shading alternate items)
  - in a sense, inviting and “easy on the eye”

Pershing and Pershing [2001] reviewed 50 *reactionnaires* (i.e., like “questionnaires” specifically intended to collect reactions) that were used by training evaluators at a “prestigious medical school.” Their purpose was to determine the degree to which the survey design adhered to the recommendations. The results suggested that:

- 72% did not include an introductory statement; an additional 16% were “minimal”
- 78% had no closing statement
- 30% had no directions; another 54% of directions were “minimal”
- 8% were professional in appearance

In summary, the formatting of the reactionnaires were not designed in a way that would maximize respondent engagement. In turn, we might expect this to threaten the psychometric reliability and validity.

## 2.3 What Improves (or Threatens) Response Rates and Bias?

When we design survey instruments based on our own preference rather than research-based evidence, we may get less than optimal data. Chyung et al. [2018b] reviewed the five steps [Schwarz and Oyserman, 2001] that survey respondents engage when answering structured, closed-ended survey items.

1. Interpreting the question.
2. Retrieving information from their memory.
3. Integrating the retrieved information with the item prompt.
4. Selecting one of the given response options.
5. Editing the answer for reasons of social desirability.

Chyung and colleagues appear to be starting such a systematic review. What follows are their evidence-based evaluations regarding some of the most common questions about questionnaire construction.

### 2.3.1 Should Likert-type scales include a midpoint?

Likert-type scales, named after Rensis Likert, include a set of questions or statements that can be responded to with a consistent set of response options. The response options are frequently scaled with intensity options ranging from 2 to 11; four and five point “agreement” scales str shown below. There are many variants of response options (i.e., frequency, like me, degree of stressfulness). Regarding the issue of a *midpoint* (“neutral” or “neither disagree nor agree”), Chyung et al. [2017] reviewed the literature. Examining their article, we can see variants of Likert-style scaling for a scale of agreement. They look something like this:

---

Type						
No midpoint (4 pt.)	Strongly Disagree	Disagree	<i>skipped</i>	Agree	Strongly Agree	
Midpoint (5 pt.)	Strongly Disagree	Disagree	Neither Disagree nor Agree	Agree	Strongly Agree	

---

Chyung and colleagues quickly suggest that the question is not “Should I use a midpoint?” but rather “When should I use a midpoint?”

The article is more detailed, but essentially, a midpoint is appropriate when:

- the measurement scale is interval (instead of ordinal; this is a good statistical property to have)
- the question content is such that the midpoint is a *true* midpoint and not a point for hedging or avoiding

If a true midpoint is impossible, then consider adding an option such as “I don’t know” or “It depends.” If the “I don’t know” option is used, the researcher needs a plan for recoding the data as missing and having a plan for managing missingness

### 2.3.2 Should *continuous rating scales* be used in surveys?

First, let’s consider the distinction between *discrete*, *continuous*, and *numerical* scales. Figure 4 in the Chyung, Swanson, Roberts, and Hankinson [2018a] article illustrate the major differences and some variations.

- **Discrete** scales are Likert-type scales that range between 2 and 11 *discrete* options. Classically, respondents pick *words* (e.g., pain rated as *no pain*, *mild*, *moderate*, *severe*, *extreme*, *worst pain possible*).
  - Six-point discrete rating scales result in a collection of six *ordered values*.
  - The measurement scale for discrete scales is *ordinal*.
  - Ordinal scales should be analyzed with non-parametric statistical procedures, however parametric approaches can be used if the data are normally distributed and there is a mid-point.
- **Continuous** scales allow respondents to indicate a response anywhere within a given range – usually by marking a place on a horizontal line on a continuum of a minimum of 100 points. There are no discrete categories defined by words or numbers.
  - Continuous scales result in precise numbers (e.g., 26 or 26.8 if the scale is 0 to 100).
  - The measurement scale for continuous scales is *interval*.
  - Interval scales can be evaluated with parametric statistics.
  - *Visual analog scales (VAS; aka graphic rating scales, GRS)* are another variant of continuous rating scales if they allow the participants to make “anywhere on the line.” Some VAS scales have verbal descriptors to guide the marking; some have numbers (hence, *numerical response scales*). In Qualtrics there is a slider option that serves this function.

Which is better? The mixed results are summarized in Chyung et al’s [2018a] Table 1. With a focus on the types of research I encounter in my program, here is my take-away:

- Continuous scales provide better data (i.e., more precise/full information, more likely to be normally distributed, better reliability) for statistical analysis.
  - *Caveat:* If the response scale on a Likert scale is increased to 11, there is a better chance to have normally distributed responses.
  - *Caveat:* When “simple descriptive statistics” are desired (e.g., histograms, frequency distributions) the discrete scale may be the best choice.

- Discrete and continuous options (including sliders) are easy to use, except in the case where respondents complete the surveys on mobile devices.
  - *Caveat:* There has been more missing data with sliders (compared to radio buttons).
  - *Caveat:* Respondents are more likely to change their responses on sliders. If this means there is greater accuracy or more careful responding, this is desirable.
- In both circumstances adding “don’t know,” “prefer not to respond,” or “not applicable” may improve the validity of the responses.
  - *Caveat:* When using one of these response options, plan ahead for coding as missing and managing the missingness.

### 2.3.3 Should Likert-type response options use an ascending or descending order?

Let’s first look at the difference between ascending and descending order [Chyung et al., 2018b]:

Type						
Ascending	Strongly Disagree	Disagree	Neither Disagree nor Agree	Agree	Strongly Agree	
Descending	Strongly Agree	Agree	Neither Agree nor Disagree	Disagree	Strongly Disagree	

In the consideration of the choice between ascending/descending, we are concerned with *response-order effects*. Let’s first examine these conceptually/theoretically.

**Recency effect** is the tendency of survey respondents to select the options that they see at the end of the response-option list. This is expected when options are presented orally (e.g., during interviews, people tend to choose from the last-offered options).

**Primacy effect** is the survey respondents’ tendency to select the options that are presented at the beginning of the response-option list. This is expected when options are presented visually. For example, people tend to choose among the first-presented categories in self-administered written survey questionnaires.

- *Left-sided selection bias* occurs when respondents read text from left-to-right and are more inclined to select options from the left.
- *Satisficing theory* occurs when individuals seek solutions that are “simply satisfactory” so as to minimize psychological costs. Thus, respondents may
  - select the first option that seems “reasonable enough”,
  - select the “I don’t know” response, or
  - randomly select one of the options.
- *Acquiescence bias* is the tendency for respondents to agree with the statement provided—aka yea-saying bias (e.g., being polite).

- Closely related is *social-desirability bias*, the tendency for respondents to select among the options they think are more socially acceptable or desirable (instead of true responses).
- In surveys, this generally is selecting *agree* or *strongly agree*.

Considering these response biases together, Chyung et al. suggest that when the response options are presented in descending order (*Strongly agree, Agree, Neutral, Disagree, Strongly disagree*), respondents would (theoretically) see a positive option immediately on the left side of the response scale and perceive it to be socially desirable and satisfactory. As a result, they may to select it without having to spend more time to choose a more accurate response. After reviewing 13 studies, Chyung et al. observed that many studies (paper and web based, with children and adults, in English and other language):

- Revealed response-order effects in self-administered surveys, especially the primacy effect, associated with left-side selection bias, acquiescence bias, and satisficing.
- Showed more positive average scores from descending-ordered scales.

Recommendations:

- Present response scales in ascending order.
  - When a number line is used, lower and negative numbers should be on the left.
- When using descended order scales:
  - keep respondents motivated to complete items accurately,
  - present half items with descended-ordered scales and the other half with ascended-ordered scales,
  - assign half of participants with descended-ordered scales; half with ascended-ordered scales, and
  - present response options vertically rather than horizontally.

#### 2.3.4 Should surveys include negatively worded items?

In examining this question, Chyung et al. [Chyung et al., 2018c] made a distinction between (see Table 1 in the article):

- **Statement format** with a consistent response scale (e.g., strongly disagree to strongly agree).
- **Question format** with variable response scales that are tailored to individual survey questions.
  - A challenge with this format is the difficulty in calculating an average score of data obtained from multiple survey items.

The advent of negatively worded items began with Rensis Likert in 1932. He was an American social psychologist who, in attempt to mitigate acquiescence/yea-saying biases, recommended designing one half of survey items to be associated with agreement and the other half with disagreement.

Although Likert recommended “straightforward statements,” incorporating negative words can become quickly complicated. Table 2 in the Chyung paper shows that there are four ways of wording survey statements:

**Reverse-coding**, which is necessary when including negatively worded items in a scale, assumes that agreeing to a positively worded statement and disagreeing to its negatively worded counterpart are the same. Tables 3 and 4 in the Chyung et al., manuscript [2018c] show how this assumption may be faulty. A review of the negatively-worded-item literature suggested the following:

- Scales with all positively worded items yielded greater accuracy when compared with all negatively worded items or mixed worded items.
- Scores on positively and negatively worded items are not the same (e.g., strongly disagreeing to a positively worded statement is different from strongly agreeing to a negatively worded statement)
- Positively worded items produce higher means than negatively worded items. This may be due to
  - carelessness and fatigue in reading items,
  - the cognitive processing of positive and negative items may be different.
- A *method factor* has shown itself where exploratory approaches to factor analysis have produced separate factors with the negatively worded (or otherwise ambiguous) items creating their own factor. This results in a threat to construct validity and reliability.

Chyung, Barkin, and Ramsey [2018c] noted that respondent performance declines approximately 12 minutes after starting a survey. It appears that respondents increasingly fail to notice negatively worded statements even when there are efforts to draw their attention to them via bolding, underlining, or capitalizing the negated element (e.g., **not**). Thus, when negatively worded items are used, they should probably be presented early in the protocol.

Chyung et al [2018c] also cautioned about a response set bias that can occur when using all positively worded items. They recommended making design choices that enhance bias-free and accurate responding based on the research design.

- For example, attributes to be measured in some constructs (e.g., depression, anxiety) are, themselves, negative and so a negatively worded item may be most clear and appropriate.
- The inclusion (and subsequent analysis) of negatively phrased items may help *detect* acquiescence bias.
- Table 5 in the Chyung et al [2018c] manuscript provides some guidelines that are more nuanced when negative items must be included. For example,
  - Ensure that negatively worded items are true polar opposites and symmetrical (so they can be analyzed with the positively worded items).
  - Group negative items together (and forewarn/format so they are recognized as such).
  - Administer the survey when respondents are not fatigued.
  - Analyze the effect of the negatively worded items.

## 2.4 Construct-specific guidance

Across disciplines and constructs, there may be localized guidance. One domain-specific example is *self-efficacy*. In this case, construct-specific guidance addresses both the (a) content of the items and (b) formatting of the scales. Regarding content, even though there are some *general self-efficacy scales* Bandura's original definition suggests that scales and their items should be task specific (i.e., career decision-making self-efficacy, math self-efficacy). Further, Bandura [2006], recommended the following for self-efficacy scales:

1. Phrase items as “can do” rather than “will do.”
2. Maintain consistency with the self-efficacy construct definition (e.g., domain specific, a focus on capability rather than self-worth).
3. Include items that reflect gradations of challenge.
4. Ask individuals to rate their current (as opposed to future) operative capabilities.
5. Use 100-point continuous scaling.

## 2.5 Surveying in the Online Environment

Nearly a decade ago, a survey of human subjects review boards suggested that 94% of the IRB applications reviewed involved online or Web-based surveys [Buchanan and Hvizdak, 2009]. Thus, it is important to understand the online environment. A first set of considerations involve data security, identity, and permission (implicit and explicit).

The **IP address** as well as **longitude/latitude** has been a contentious issue for a number of years [Buchanan and Hvizdak, 2009]. EU data protection laws consider IP addresses as personally identifiable data; in the U.S., IP addresses typically fall outside the definition of “personal information.” In Qualtrics, the default is to collect and download the IP address (the “anonymize response” option can prevent this data from being collected). On the one hand it is helpful to know geographically “from where” participants are responding; on the other, some consider its capture to be a violation of privacy. Relatedly, **paradata** and **metadata** are data such as typing speed, changed answers, response times, and time spent on the survey. For both geolocation and paradata/metadata, a strong consideration is **fully informed consent** [Schober and Conrad, 2007]. Is it ethical to capture this information without explicitly saying so? A best practice is to provide complete descriptions of what data is being collected and provide a rationale (in the IRB application, if not directly in the informed consent) for why it is necessary. Survey tools like Qualtrics have strong options for anonymizing data (i.e., permanently deleting identifying information). If this option is being used, this information should be included in the informed consent (and, perhaps, recruiting materials).

The specific **survey tool** being used should be evaluated. Buchanan and Hvizdak [2009] argued that until each tool is vetted and its privacy policies and data security policies are understood, we cannot be certain how security, consent, and privacy are operationalized within the individual tools. For example, it is possible that tool creators *could* gather respondent data and repurpose it for their own marketing, for sale to other researchers, and so forth.

Online and web-based protocols increase our reach geographically and cross-culturally. A first impression might be that the online environment increases **access** [Schober and Conrad, 2007]. We should probably think twice about this presumption. Consider the decades of psychological research

based on White, college-educated, males. Does the online environment create another strata of privileged research with technology that may not be accessible in terms of both internet/technology as well as capacity/fluency with the tool? Additionally, does the way that the survey is promoted result in invitations that only occur within certain segments of the internet? If so, the results may not be representative. On the other hand, what are the risks of not adopting new technologies before everyone has them. Another consideration is cultural and language translation. These issues are addressed more completely in the lesson on [invariance testing](#).

When paper/pencil measures were administered in face-to-face settings (individually or in auditoriums of students) there was some degree of a **standardized protocol**. This is lost when surveys are administered online. Further, we cannot guarantee *who* is taking the survey. Increasingly, when surveys are offered through fee-based programs like mTurk and Prolific, bots have been trained to take the surveys and receive the incentive. Survey programs like Qualtrics now offer additional packages to help with security and bot-prevention. It is probably also wise to add attention-check items (e.g., “This is an attention check. Please answer ‘3’”). Another option is to include a final question that asks the respondent to ensure the integrity of the response. An example last item might be, “To what degree is the statement below true of you: *I read each question/item and provided answers that were true for me.*” The scaling for this item was a 100-point slider with “Untrue,” “Neither true or untrue,” and “True”.

When respondents are remote, what happens if they have a **negative reaction to the survey**? In a face-to-face context, debriefings can occur, and referrals can be made. IRB committees are likely to consider the degree to which surveys may be upsetting and require resources for referral and assistance in the case of an adverse event.

**Security of test items** might also be concerning. It is inappropriate to use proprietary items without the permission of its author. If the security of items is important (e.g., SAT/GRE, intelligence test items, inkblots) because they are central to administration, how can they be protected in the virtual environment?

Consequently, when students in our programs write doctoral dissertations they are to include the following in their Method section.

- Describe how informed consent will be obtained in the online environment.
- Describe the level of identification that is collected. If the claim of “anonymous” or “de-identified” indicate whether/not this includes capturing the IP address; some researchers believe that capturing a computer’s IP address threatens anonymity.
- Describe the steps to be taken to ensure that respondents met the inclusion/exclusion criteria of the study.
- Anticipate and describe how the online (e.g., uncontrolled, public, distractions) setting might affect responses.
- Particularly if the survey contained sensitive materials, describe how respondents might access resources for debriefing or referral.
- Identify the permissions (from original authors or copyright holders) granted to reformat and post (on the internet) existing surveys. If items are considered to be secure (e.g., those on the MMPI or WAIS), identify steps taken to protect them.

## 2.6 In my Surveys

Because there isn't empirical data on every decision that we make in survey construction, I thought it might be useful for me to address some of the decisions that I find myself making in the online surveys I use in my own research.

### 2.6.1 Demographics and Background Information

A core value that I hope to reflect in the *ReCentering Psych Stats* series is to promote socially and culturally responsive research. Correspondingly, the information we collect from participants should ensure that they feel that their identities are authentically reflected in the survey. Naively, when I first considered how to capture race/ethnicity in my surveys, I looked to the categories used in the U.S. Census. Immediately, I learned that this is problematic. Rather than elaborating here, I invite you to take a listen to NPR's [Code Switch](#) podcast. Two of the episodes review how the assessment of race and ethnicity has evolved and explain why it is problematic: [Census Watch 2020](#) and [The U.S. Census and Our Sense of Us](#). As made clear in the Code Switch podcasts, the assessment of race and ethnicity in the U.S. Census *erases* people when their identities are not included.

My last few surveys have captured race/ethnicity data differently. Each time, I engage in several practices that (I hope) will continue to shape the item in a socially and culturally responsive way. Systematically, I:

- conduct a quick internet search to see if there is an emerging best practice (even though I may have also searched weeks or months prior),
- consider who the intended research population is in relationship to the topic of investigation,
- look to recently published, similar, research to see what other researchers are doing, and
- ask for a collegial, formative review from individuals who hold marginalized identities, whose data will be requested in the survey.

When I engage in research, I try to balance the need to quantify (with discrete categories) who is participating in the survey and inviting respondents to state (in their own words) their identity. This is consistent with my view that variables like race, ethnicity, and gender identity are socially constructed. In addition to this particular worldview, Parent [2013] has suggested that the worst possible kind of missing data pattern (MNAR – missing not at random) may be caused when items are *unanswerable* to particular person. Therefore, it is essential that all individuals recognize themselves in the items that assess demographic variables.

A recent survey of mine was directed toward community members (including students, alumni, staff, faculty) of my predominantly White, liberal arts, Christian institution. After reviewing recently published research articles and consulting with a handful of individuals, I chose to include the following categories – *each with a text write-in box* so that individuals could select the category(ies) that fit best and have the opportunity to refine it(them). I am excited to review this data because such responses may inform my next survey. The categories included:

- Asian or Pacific Islander
- Black or African American

- Hispanic or Latino
- Native American or Alaskan Native
- White or Caucasian
- Biracial or multiracial
- An international/global identity that does not fit in the U.S. categorization of race/ethnicity
- A race/ethnicity not listed above

Respondents could select multiple categories. Additionally, they could write in an identity that better aligned with their self-understanding.

RaceEthnicity ★

Which of the following best describes you?

*Owing to the limitations of surveys, we have provided categories typically used in survey construction. Please select the category(ies) that is(are) closest to the manner in which you identify. You are welcome to refine the response in the text box.*

Asian or Pacific Islander  
 Black or African American  
 Hispanic or Latino  
 Native American or Alaskan Native  
 White or Caucasian  
 Biracial or multiracial  
 An international/global identity that does not fit in the U.S. categorization of race/ethnicity  
 A race/ethnicity not listed above

Figure 2.1: Image of a survey item inquiring about race/ethnicity from a survey. Each option has an option for the respondent to clarify.

The option to select multiple boxes results in some extra coding when preparing the data for analysis. I am taking approach that we will *listen* to the data and decide, based on the results, how to report the findings in a way that will efficiently fit into an APA style empirical paper and honor the respondents.

The population of interest for this particular study are those who are engaged in protest activities regarding hiring practices and policies that result in discrimination to members of the LGBTQIA+ community. This means that questions of gender identity, pronouns, and relationship to the LGBTQIA+ community are important to the research and need to be asked sensitively and with great security of the data.

Regarding gender identity, I used a similar approach, allowing individuals to select multiple categories and offering write-in boxes for each. The categories included:

- female
- male
- nonbinary
- trans woman
- trans man
- another identity (with a write-in box)
- prefer not to say

Additionally, I invited individuals to identify their pronouns. Again, write-in boxes were offered with each option.

- they/them/theirs
- she/her/hers
- he/him/his
- they/she/he
- neo pronouns (e.g., xe/xem/xyr, ze/hir/hirs, ey/em/eir)
- another identity (with a write-in box)

Finally, we wanted individuals to indicate their relationship to the LGBTQIA+ community. We asked them to select all that apply. Only the “something else” box had a write-in option:

- Member
- Exploring/Questioning
- Ally
- Not related
- Something else (with a write-in box)

I expect that my future surveys may inquire about these variables differently. If you have found a different way to ask, please consider e-mailing me. I would love to provide different options and give credit to contributors.

### 2.6.2 Survey Order

Historically, demographic information has been first or last in the survey. Although some research has reported no differences in response rates when demographic and sensitive data are at the beginning or end [Krathwohl, 2009, Rowley, 2014], I am inclined to open the survey with questionnaire items that are closely related to the topic listed on the recruitment materials and end the survey with the demographic information. Why? It makes sense to me that if someone has responded positively to the survey topic, they expect to answer questions (right away) about that topic.

In between that opening survey and closing demographic items, I consider if there are any *order effects* that would engage in undue *priming* of responses. If there are such concerns, I think through the order to minimize these biasing effects. If there are no such concerns, I put my surveys in blocks and then ask my survey program to randomly present the blocks. This serves two purposes:

- counterbalancing possible order effects, and
- distributing missingness for individuals who do not complete the survey.

### 2.6.3 Forced Responses

Programs like Qualtrics are able to engage in a variety of *content validation* procedures. If these are in place, they may require the person to enter a properly formatted response (e.g., phone number, e-mail address, numerical response between 0 and 100) before responding. These are

extremely helpful tools in collecting data that will be closest-to-being-ready-for-analysis. These same procedures can *force* or *request* a response.

*Requiring* a response is tempting. However, doing so violates IRB requirements that allow a person to skip or “quit at any time without penalty.” They may also anger a person such that they stop responding. Some researchers get around this by *requiring* the response but including a “Not applicable” or “Prefer to not answer” column. Because I worry that (a) the respondent may confuse that option with one extreme of the scale and/or (b) my research team and I will forget to code it as missing data, I prefer the *request* response alternative.

In Qualtrics in particular, I turn on the “Request response” feature for each of the questions. If an item is skipped, a simple warning is displayed that invites the respondent to review the page of answers to see if they would like to answer the question. If not, they can simply move forward.

## 2.7 Practice Problems

In each of these lessons I provide suggestions for practice that allow you to select one or more problems that are graded in difficulty. With each of these options I encourage you to:

This is a two-part lesson on questionnaire construction. After the [second lesson](#), a detailed suggestion for practice will be provided that lists criteria for creating and piloting a survey of your own.

# Chapter 3

## Be a QualTRIXter

### [Screencasted Lecture Link](#)

The focus of this lecture is on the technical and mechanical tools available in Qualtrics (and likely other survey platforms) to increase the effectiveness of your survey.

### 3.1 Navigating this Lesson

This lecture is just under one hour. Plan for another 30 minutes for *intRavenous qualtRics* practice.

While the majority of R objects and data you will need are created within the R script that sources the chapter, occasionally there are some that cannot be created from within the R framework. Additionally, sometimes links fail. All original materials are provided at the [GitHub site](#) that hosts the book. More detailed guidelines for ways to access all these materials are provided in the OER's [introduction](#)

#### 3.1.1 Learning Objectives

Focusing on this week's materials, make sure you can:

- Utilize basic Qualtrics tools (e.g., question type, use of headers) so that surveys are present materials clearly to the respondent.
- Incorporate more advanced tools (e.g., display logic, randomization) that may increase the respondent's ability to complete the survey and provide accurate responses.
- Provide a rationale for survey options that protect (or possibly reveal) an individual's identity.

#### 3.1.2 Planning for Practice

This is the second of a two-part lesson on questionnaire construction. At the end of this lesson is a detailed suggestion for practice that lists criteria for creating and piloting a survey of your own. There are four essential criteria for your survey:

- Adhere to the evidence-based practices identified in the lesson on [questionnaire construction](#).

- Utilize four techniques (in the context of Qualtrics, I term these *qualTRIXter skills*) that increase the flow, effectiveness, and appearance of your survey.
- Pilot and consider feedback provided by those who took the survey.
- Import the data into the R environment.

### 3.1.3 Readings & Resources

In preparing this chapter, I drew heavily from the tutorials available at the [Qualtrics support site](#). I have tried to link them throughout the presentation. It is likely they could change at any time and/or they might not work on your particular browser.

### 3.1.4 Packages

The packages used in this lesson are embedded in this code. When the hashtags are removed, the script below will (a) check to see if the following packages are installed on your computer and, if not (b) install them.

```
# will install the package if not already installed
# if(!require(qualtrics)){install.packages('qualtrics')}
```

## 3.2 Research Vignette

I will demonstrate the qual”TRIX” by using a Qualtrics account hosted at Seattle Pacific University. The only surveys in this account are for the *Recentering Psych Stats* chapters and lessons. All surveys are designed to not capture personally identifying information and not collecting IP addresses nor longitude/latitude. I use this survey in several lessons in this OER. If you haven’t taken the survey yet, [I invite you to do so, now](#).

As a teaching activity for the ReCentering Psych Stats OER, the topic of the survey was selected to be consistent with the overall theme of OER. Specifically, the purpose of this study is to understand the campus climate for students whose identities make them vulnerable to bias and discrimination. These include students who are Black, non-Black students of color, LGBTQ+ students, international students, and students with disabilities.

After consulting with a diverse group of stakeholders and subject matter experts (and revising the response options numerous times) I have attempted to center anti-Black racism in the U.S. [Mosley et al., 2021, 2020, Singh, 2020]. In fact, the display logic does not present the race items when the course is offered outside the U.S. There are only five options for race: *biracial/multiracial*, *Black*, *non-Black person(s) of color*, *White*, and *I did not notice* (intended to capture a color-blind response). One unintended negative consequence of this design is that the response options could contribute to *colorism* [Adames et al., 2021, Capielo Rosario et al., 2019]. Another possibility is that the limited options may erase, or make invisible, other identities. At the time that I wrote up the first description of this survey, the murder of six Asian American women in Atlanta had just occurred. The Center for the Study of Hate and Extremism has documented that while overall hate crimes dropped by 7% in 2020, anti-Asian hate crimes reported to the police in America’s largest cities increased by 149% [noa, a]. These incidents have occurred not only in cities, but in

our neighborhoods and on our campuses [Kim, 2021b,a, noa, b]. While this survey is intended to assess campus climate as a function of race, it unfortunately does not distinguish between many identities that experience marginalization.

Although the dataset should provide the opportunity to test a number of statistical models, one working hypothesis that framed the study is that there will be a greater sense of belonging and less bias and discrimination when there is similar representation (of identities that are often marginalized) in the instructional faculty and student body. Termed, “structural diversity” [Lewis and Shah, 2019] this is likely an oversimplification. In fact, an increase in diverse representation without attention to interacting factors can increase hostility on campus [Hurtado, 2007]. Thus, the task of rating of a single course relates to the larger campus along the dimensions of belonging and bias/discrimination. For example, if a single class has higher ratings on issues of inclusivity, diversity, and respect, we would expect that sentiment to be echoed in the broader institution.

The survey design has notable limitations You will likely notice that we ask about demographic characteristics of the instructional staff and classmates in the course rated, but we do not ask about the demographic characteristics of the respondent. In making this decision, we likely lose important information. For example, Iacobino and James [2016] have noted that White students perceive campus more favorably than Black student counterparts.

The decision to not collect demographic details about the respondent was about protecting their (your) identity. As you will see, you have the opportunity to download and analyze the data. If a faculty member asked an entire class to take the survey, the date stamp and a handful of demographic identifiers could very likely identify a student. In certain circumstances, this might be risky in that private information (i.e., gender nonconformity, disclosure of a disability) along with course evaluation data and a date stamp could identify the respondent.

Further, the items that ask respondents to *guess* the identities of the instructional staff and classmates are limited, and contrary to best practices in survey construction that recommend providing the option of a “write-in” a response.

In parallel, the items asking respondents to identify characteristics of the instructional staff along dimensions of gender, international status, and disability are “large buckets” and do not include “write-in” options. Similarly, there was no intent to cause harm by erasing or making invisible individuals whose identities are better defined by different descriptors. Further, no write-in items were allowed. This was also intentional to prevent potential harm caused by people who could leave inappropriate, racist, or otherwise harmful comments.

As I review Qualtrics essentials and trix, I will their use (if used) in the ReCentering Psych Stats survey.

### 3.3 Qualtrics Essentials

Qualtrics is a powerful program and I find that many of the surveys we distribute don’t capitalize on the features Qualtrics has to offer. Qualtrics has detailed tutorials and instructions that are well worth the investment of a weekend to review them.

In this lecture I will point you to the elements that I think are critical to constructing online surveys. Because Qualtrics tutorials are (a) clear and thorough and (b) frequently updated, I will (a) point you to the tutorials that are available at the time of this lecture prep, (b) tell you why I think they are appropriate, and (c) show you how we have used them in some of our own surveys.

Even if you think you know what you are doing, start here (and then always take the time to “look around” at all the options on each window):

**Survey Basic Overview:** Qualtrics’ [Survey Basic Overview](#) tutorial is a great place to start. From there, you can follow all kinds of leads, looking for things you want to do with your survey – and getting ideas for what will improve it.

**Blocks** are the basic organizational tool in Qualtrics surveys. Blocks have two purposes: (a) grouping items shown on “one page,” and (b) specifying ordering and/or random selection/presentation in the survey flow.

**Question types:** Take time to look at all the options. You might be surprised to learn that there is a better choice than you might have imagined.

Let’s take a look at super basic/helpful question types:

- **Text/graphic:** These are the types you should use for providing information (e.g., informed consent) to the participants or displaying a logo or graphic stimulus.
- **Matrix table:** The matrix table is a more efficient way to use the Likert-style items (than multiple choice). There is some controversy about whether not to use matrix tables vs. multiple choice dropdowns. As both a survey developer and a respondent, I prefer the matrix table.
  - Make sure to select a reasonable amount of header repetitions. This allows the respondent the maximum opportunity to see the column descriptors (and avoid guessing/remembering) while they are responding.
- **Slider :** The slider is designed for obtaining truly continuous data on a 1 to 100 scale. This range can be adapted to any interval you choose, and you can add anchors to the scale. If the scale you are using is already published, and has not been psychometrically evaluated for slider use, you should probably stick with the format recommended in the publication. But if you are writing test items, consider this option.
- **Text Entry Questions:** Text boxes have multiple options for answer length.
- **Validation:** Content validation allows the user to permit certain types of information and specify their formats (e.g., numbers, e-mail addresses, dates). There is art to balancing between being overly restricting and ensuring that the data is entered in the most clear and consistent way possible with honoring the uniqueness of each respondent. Another validation option I frequently use is one that asks individuals if they intended to leave something blank. This is tool that helps prevent missingness without forcing an individual to respond to an item that (a) might not be clear to them, (b) might not be appropriate for them, and/or (c) might result in an answer that is untrue for their unique circumstance.

## 3.4 Qual-TRIX

**Collaborating** with other Qualtrics users in your institution is easy! Scroll down to “Collaborating Inside Your Organization” and follow the instructions for adding individuals to your survey (you must “own” the survey; your collaborators will not be able to add others).

The ability to **schedule survey distributions** is like having your very own assistant! If you have a roster (contact list) you can schedule distributions, reminders, and thank you's. Qualtrics will keep track of who responds and send reminders to the non-responders. Here are resources for

- [E-mail overview](#)
- [E-mail distribution management](#)
- [Directories](#)

**Personalizing** invitations and surveys. **Piped text** is a way to personalize invitations and/or “carry forward” prior responses into new questions.

**Randomization of blocks** (or a subset of blocks) can be used for several purposes such as: (a) using random selection to display one or more blocks to respondents – as in a random clinical trial, (b) to randomly display a percentage of blocks or items to shorten the survey in a planned missing design, and (c) randomly display some or all of the blocks of the survey to all respondents so that when respondents experience test fatigue, when they quit responding, “the last items/surveys” aren’t always the same ones. This functions to distribute missingness across surveys.

**Randomization of items** within a block can be used for similar purposes. You can also use this to display only some of the items (e.g., planned missingness).

**File upload from respondents** is an additional package that requires the institution to pay a higher fee. If available, this allows respondents to upload some sort of file (photo, PowerPoint, .pdf). We use it for poster contests at professional contests (where students upload their poster for online judging in advance of the conference). A colleague of mine uses this function to collect application elements (i.e., resumes, cover letters, reference letters) to a fellowship program.

- As researchers, we can also upload files (e.g., hardcopy of informed consent, documents to be reviewed) for use by the respondent.

**Display, Skip, and/or Branch Logic** can be used to help display to respondents *only* the items that pertain to them. There are multiple approaches to doing this. Using a display logic approach may feel a bit *backward* where the logic is applied *from* the landing spot. We did this extensively in a study that involved two language versions and three age options.

Two other approaches for these issues are [skip logic](#) and [branch logic](#)

## 3.5 Even moRe, particularly relevant to iRb

We can use Qualtrics tools for purposes beyond collecting and downloading data. These tools are especially useful when I think about IRB applications and ethics related to data collection.

**Exporting to Word:** Helpful for your IRB application (and perhaps in a cloud so that a team can use track changes to edit), it is super simple to export the survey to Microsoft Word. Additionally, you can specify options for including question numbers, recode values, logic, and so forth. This works well to create a codebook for your research team.

**Anonymizing responses:** Another step toward an anonymous response is to withhold the IP address, latitude/longitude, and any contact information (e.g., e-mail, name) that you may have uploaded in an e-mail distribution directory. This is accomplished in the Survey Options menu. Do be careful – while anonymizing responses is an ethical, best practice, the deleted information cannot be recovered.

**Prevent ballot box stuffing:** Want to make sure that respondents only answer once? In the same Survey Options window, you can prevent ballot box stuffing. This is helpful when surveys are distributed with an *anonymous link*. The tool prevents more than one survey from the same IP address.

Other security options include

- Password protection
- HTTP Referrer verification

Look also at:

- **Progress bar** to provide participants hope (or despair) for “how much longer.”
- **Survey termination** to connect custom endings and thank-you notes.
- **Partial completion** to specify how long the respondent has to complete the survey (after opening it) and whether it is recorded or deleted if it is not completed.
  - Related to this, back on the *Data & Analysis* tab, you can see both the numbers of [recorded responses and responses in progress](#). You also have options to manually determine how you want to include/exclude the responses in progress.
  - Failure of the respondent to click the final “->” submit and progress symbol is often the reason that surveys that are > 90% complete aren’t counted as “complete.” What to do? Options: (a) don’t say “Thanks and goodbye” on a page that has any items, and (b) provide instructions to look for the “->” symbol to continue.

Finally, **PREVIEW PREVIEW PREVIEW!** There is no better way check your work than with previews.

## 3.6 intRavenous Qualtrics

Access credentials for the institutional account, individual user’s account, and survey are essential for getting the survey items and/or results to export into R. The Qualtrics website provides a tutorial for [generating an API token](#).

We need two pieces of information: the **root\_url** and an **API token**.

- Log into your respective qualtrics.com account.
- Select Account Settings
- Choose “Qualtrics IDs” from the username dropdown

We need the **root\_url**. This is the first part of the web address for the Qualtrics account. For our institution it is: spupsych.az1.qualtrics.com

The API token is in the box labeled, “API.” If it is empty, select, “Generate Token.” If you do not have this option, locate the *brand administrator* for your Qualtrics account. They will need to set up your account so that you have API privileges.

*BE CAREFUL WITH THE API TOKEN* This is the key to your Qualtrics accounts. If you leave it in an .rmd file that you forward to someone else, this key and the base URL gives access to every survey in your account. If you share it, you could be releasing survey data to others that would violate confidentiality promises in an IRB application.

If you mistakenly give out your API token, you can generate a new one within your Qualtrics account and re-protect all its contents.

You do need to change the API key/token if you want to download data from a different Qualtrics account. If your list of surveys generates the wrong set of surveys, restart R, make sure you have the correct API token and try again.

```
# only have to run this ONCE to draw from the same Qualtrics
# account...but will need to get different #token if you are changing
# between accounts.
qualtrics::qualtrics_api_credentials(api_key = "oEwd9qu9xJ0f3RoE9iiCZKSs2sfNuSbvy8LnFYxo",
  base_url = "spupsych.az1.qualtrics.com", overwrite = TRUE, install = TRUE)
readRenvironment("~/.Renviron")
```

*all\_surveys()* generates a dataframe containing information about all the surveys stored on your Qualtrics account.

```
surveys <- qualtrics::all_surveys()
# View this as an object (found in the right: Environment). Get
# survey id # for the next command If this is showing you the WRONG
# list of surveys, you are pulling from the wrong Qualtrics account
# (i.e., maybe this one instead of your own). Go back and change your
# API token (it saves your old one). Changing the API likely requires
# a restart of R.
surveys
```

To retrieve the survey, use the *fetch\_survey()* function.

```
# obtained with the survey ID
#'surveyID' should be the ID from above
#'verbose' prints messages to the R console
#'label', when TRUE, imports data as text responses; if FALSE prints the data as numerical responses
#'convert', when TRUE, attempts to convert certain question types to the 'proper' data type in
#'force_request', when TRUE, always downloads the survey from the API instead of from a temporary file
#'import_id', when TRUE includes the unique Qualtrics-assigned ID;
# since I have provided labels, I want false
```

```
# Out of the blue, I started getting an error, that R couldn't find
# function 'fetch_survey.' After trying a million things, adding
# qualRics:: to the front of it solved the problem
QTRX_df <- qualRics::fetch_survey(surveyID = "SV_b2cClqA1LGQ6nLU", time_zone = NULL,
  verbose = FALSE, label = FALSE, convert = FALSE, force_request = TRUE,
  import_id = FALSE)

# useLocalTime = TRUE,
```

The optional script below will let you save the simulated data to your computing environment as either a .csv file (think “Excel lite”) or .rds object (preserves any formatting you might do).

```
# write the simulated data as a .csv write.table(QTRX_df,
# file='QTRX_df.csv', sep=',', col.names=TRUE, row.names=FALSE) bring
# back the simulated dat from a .csv file QTRX_df <- read.csv
# ('QTRX_df.csv', header = TRUE)

# to save the df as an .rds (think 'R object') file on your computer;
# it should save in the same file as the .rmd file you are working
# with saveRDS(QTRX_df, 'QTRX_df.rds') bring back the simulated dat
# from an .rds file QTRX_df <- readRDS('QTRX_df.rds')
```

### 3.6.1 The Codebook

In order to prepare data from a survey, it is critical to know about its content, scoring directions for scales/subscales, and its design. As I demonstrated above, we can export a **codebook**, that is, a Word (or PDF) version of the survey with all the coding. In Qualtrics the protocol is: Survey/Tools/ImportExport/Export Survey to Word. Then select all the options you want (especially “Show Coded Values”). A tutorial provided by Qualtrics can be found [here](#). This same process can be used to print the PDF example I used above.

I recommend providing custom variable names and recode values directly in Qualtrics before exporting them into R (and before exporting the codebook). A Qualtrics tutorial for this is provided [here](#). In general, consider these qualities when creating variable names:

- Brevity: historically, SPSS variable names could be a maximum of 8 characters.
- Intuitive: although variables can be renamed in R (e.g., for use in charts and tables), it is helpful when the name imported from Qualtrics provides some indication of what the variable is.
- Systematic: start items in a scale with the same stem, followed by the item number – ITEM1, ITEM2, ITEM3.
- Do not include special characters or spaces in variable names; this is problematic for R.
- Do not start variable names with numerals; this is problematic for R.

More complete information about data preparation is covered in chapters in the [ReCentering Psych Stats: Multivariate Modeling](#) text.

### 3.6.2 Using data from an exported Qualtrics .csv file

It is also possible to download the Qualtrics data in a variety of formats (e.g., CSV, Excel, SPSS). Since my R and Qualtrics history began by using files with the CSV extension (think “Excel” lite), that is my preference.

In Qualtrics, these are the steps to download the data: Projects/YOURsurvey/Data & Analysis/Export & Import/Export data/CSV/Use numeric values. In order to import this data into R, it is critical that to save this file in the same folder as the .rmd file that you will use with the data.

R is sensitive to characters used filenames. As downloaded, my Qualtrics .csv file had a long name with spaces and symbols that are not allowed. Therefore, I gave it a simple, sensible, filename, “ReC\_Download210319.csv”. An idiosyncrasy of mine is to datestamp filenames. I use two-digit representations of the year, month, and date so that if the letters preceding the date are the same, the files would alphabetize automatically.

```
# QTRX_csv <- qualtrics::read_survey('ReC_Download210319.csv',
# strip_html = TRUE, import_id = FALSE, time_zone=NULL, legacy =
# FALSE)
```

Although minor tweaking may be required, the same script above should be applicable to this version of the data.

### 3.6.3 Tweaking Data Format

Two general approaches:

1. Inside Qualtrics: Use the [recode values](#) option (found in the item’s gearbox, to the left of the block) to specify variable names and recode values. These should be preserved on the download.
2. In the R script: In another lecture I demonstrate how to change the formats of data (character, string), selecting only the variables in which we are interested (e.g., excluding the meta-data), and renaming variables sensibly.

Both work! You can choose your preference. When you are working with a team, map out an explicit process with your collaborators.

## 3.7 Practice Problems

The suggestion for practice is to develop a questionnaire, format it, pilot it, and download it. Essentially you will be

- Formatting a survey on Qualtrics using all the best practices identified in the lecture. These include:

- Having an introductory statement (to include statement of confidentiality), directions for each sub-survey (if more than one), and a closing statement.
- Selecting the most appropriate question type for the items. For example, matrix instead of multiple choice.
- Within the question type, using the appropriate options for proper formatting (e.g., the anchors in a matrix should be topically consistent and equal interval).
- The survey should include minimum of 3 of the qualTRIXter skills (identified in lecture). Choose from:
  - establishing collaboration
  - scheduling e-mail distribution and follow-up
  - personalizing the survey in some way
  - randomization of blocks or items
  - integrating display, skip, or branch logic (e.g., having males and females take a different route)
  - exporting the survey to Word
  - recoding variables in the item controls
  - anonymizing the responses
  - preventing ballot box stuffing
  - including a progress bar
  - creating a custom ending, e-mail, or thank-you note
  - something else that YOU discovered that isn't in the lecture
- Piloting it, getting their feedback, and identifying what problems are (and how you might fix them).
  - with 3 folks from your research team, cohort, or this class
  - with 3 additional folks who aren't quite as “research savvy”
  - collect their feedback (ideally in a text-item directly on the survey itself) and write a brief summary (3 paragraphs max) of their impressions and how you might improve the survey
- Import the Qualtrics data directly R
  - preferably, directly from Qualtrics with the API token, base URL, and survey ID
  - alternatively (for the same # of points) from the exported CSV file *via the qualtrics package* (required)

Assignment Component	Points Possible	Points Earned
1. Qualtrics survey best practices	5	
2. QualTRIXter skills (at least 3)	5	
3. Minimum of 6 pilot respondents	5	
4. Summary of pilot feedback	5	
5. Import of Qualtrics data into R	5	
6. Explanation to grader	5	
<b>Totals</b>	<b>20</b>	

# Chapter 4

## Psychometric Validity: Basic Concepts

### [Screencasted Lecture Link](#)

The focus of this lecture is to provide an introduction to validity. This includes understanding some of the concerns of validity, different aspects of validity, and factors as they affect validity coefficients.

### 4.1 Navigating this Lesson

There is just over one hour of lecture. If you work through the materials with me, plan for an additional hour.

While the majority of R objects and data you will need are created within the R script that sources the chapter, occasionally there are some that cannot be created from within the R framework. Additionally, sometimes links fail. All original materials are provided at the [GitHub site](#) that hosts the book. More detailed guidelines for ways to access all these materials are provided in the OER's [introduction](#)

#### 4.1.1 Learning Objectives

Focusing on this week's materials, make sure you can:

- Distinguish between different types of validity based on short descriptions.
- Compute and interpret validity coefficients.
- Evaluate the incremental validity of an instrument-of-interest.
- Define and interpret the standard error of estimate.
- Develop a rationale that defends importance of establishing the validity of a measuring instrument.

### 4.1.2 Planning for Practice

In each of these lessons I provide suggestions for practice that allow you to select one or more problems that are graded in difficulty. The least complex is to change the random seed in the research and rework the problem demonstrated in the lesson. The results *should* map onto the ones obtained in the lecture.

The second option involves utilizing one of the simulated datasets available in this OER. The [last lesson](#) in the OER contains three simulations that could be used for all of the statistics-based practice suggestions. At this very first lesson which involves statistics, I strongly encourage you to select one of these sets and use it for all of the statistics-based homework. Working a single instrument “all the way through” a psychometric evaluation is a great way to (a) understand the psychometric evaluation workflow and (b) get to know an instrument.

Alternatively, Lewis and Neville’s [2015] Gendered Racial Microaggressions Scale for Black Women will be used in the lessons on exploratory factor analysis; Keum et al.’s Gendered Racial Microaggressions Scale for Asian American Women [Keum et al., 2018] will be used in the lessons on confirmatory factor analysis; and Conover et al.’s [2017] Ableist Microaggressions Scale is used in the lesson on invariance testing. Any of these would be suitable for the PCA and PAF homework assignments.

As a third option, you are welcome to use data to which you have access and is suitable for validity testing. In any case, please select a scale that has item-level data for which there is a theorized total scale score as well as two or more subscales (three subscales is ideal). With each of these options, plan to:

- Create a correlation matrix that includes the instrument-of-interest and the variables that will have varying degrees of relation\*.
- With convergent and discriminant validity in mind, interpret the validity coefficients; this should include an assessment about whether the correlation coefficients (at least two different pairings) are statistically significantly different from each other.
- With at least three variables, evaluate the degree to which the instrument demonstrates incremental validity

### 4.1.3 Readings & Resources

In preparing this chapter, I drew heavily from the following resource(s). Other resources are cited (when possible, linked) in the text with complete citations in the reference list.

- Jhangiani, R. S., Chiang, I.-C. A., Cuttler, C., & Leighton, D. C. (2019). Reliability and Validity. In *Research Methods in Psychology*. <https://doi.org/10.17605/OSF.IO/HF7DQ>
- Clark, L. A. & Watson, D. (1995). Constructing validity: Basic issues in objective scale development. *Psychological Assessment*, 7, 309-319.
  - In this manuscript, Clark and Watson (1995) create a beautiful blend of theoretical issues and practical suggestions for creating measures that evidence construct validity. From the practical perspective, the authors first guide potential scale constructors through the literature review and creating an item pool (including tips on writing items). The

authors address structural validity by first beginning with strategies for constructing the test. In this section, the authors revisit the issue of dimensionality (i.e., alpha vs. factor analysis). Finally, the authors look at initial data collection (addressing sample size) and psychometric evaluation.

#### 4.1.4 Packages

The packages used in this lesson are embedded in this code. When the hashtags are removed, the script below will (a) check to see if the following packages are installed on your computer and, if not (b) install them.

```
# will install the package if not already installed  
# if(!require(tidyverse)){install.packages('tidyverse')}  
# if(!require(MASS)){install.packages('MASS')}  
# if(!require(psych)){install.packages('psych')}
```

## 4.2 Research Vignette

This lesson provides descriptions of numerous pathways for establishing an instrument's validity. In fact, best practices involving numerous demonstrations of validity. Across several lessons, we will rework several of the correlational analyses reported in the research vignette. For this lesson in particular, the research vignette allows demonstrations of convergent/discriminant validity and incremental validity.

The research vignette for this lesson is the development and psychometric evaluation of the Perceptions of the LGBTQ College Campus Climate Scale [Szymanski and Bissonette, 2020]. The scale is six items with responses rated on a 7-point Likert scale ranging from 1 (*strongly disagree*) to 7 (*strongly agree*). Higher scores indicate more negative perceptions of the LGBTQ campus climate. Szymanski and Bissonette have suggested that the psychometric evaluation supports using the scale in its entirety or as subscales composed of the following items:

- College Response to LGBTQ students:
  - My university/college is cold and uncaring toward LGBTQ students.
  - My university/college is unresponsive to the needs of LGBTQ students.
  - My university/college provides a supportive environment for LGBTQ students.  
[un]supportive; must be reverse-scored
- LGBTQ Stigma:
  - Negative attitudes toward LGBTQ persons are openly expressed on my university/college campus.
  - Heterosexism, homophobia, biphobia, transphobia, and cissexism are visible on my university/college campus.
  - LGBTQ students are harassed on my university/college campus.

Because data is collected at the item level (and I want this resource to be as practical as possible), I have simulated the data for each of the scales utilized in the research vignette at the item level. Simulating the data involved using factor loadings, means, standard deviations, and correlations between the scales. Because the simulation will produce “out-of-bounds” values, the code below re-scales the scores into the range of the Likert-type scaling and rounds them to whole values.

Five additional scales were reported in the Szymanski and Bissonette article [2020]. Unfortunately, I could not locate factor loadings for all of them; and in two cases, I used estimates from a more recent psychometric analysis. When the individual item and their factor loadings were known, I assigned names based on item content (e.g., “lo\_energy”) rather than using item numbers (e.g., “PHQ4”). When I am doing psychometric analyses, I prefer item-level names so that I can quickly see (without having to look up the item content) how the items are behaving. While the focus of this series of chapters is on the LGBTQ Campus Climate scale, this simulated data might be useful to you in one or more of the suggestions for practice (e.g., examining the psychometric characteristics of one or the other scales). The scales, their original citation, and information about how I simulated data for each are listed below.

- **Sexual Orientation-Based Campus Victimization Scale** [Herek, 1993] is a 9-item item scale with Likert scaling ranging from 0 (*never*) to 3 (*two or more times*). Because I was not able to locate factor loadings from a psychometric evaluation, I simulated the data by specifying a 0.8 as a standardized factor loading for each of the items.
- **College Satisfaction Scale** [Helm et al., 1998] is a 5-item scale with Likert scaling ranging from 1 (*strongly disagree*) to 7 (*strongly agree*). Higher scores represent greater college satisfaction. Because I was not able to locate factor loadings from a psychometric evaluation, I simulated the data by specifying a 0.8 as a standardized factor loading for each of the items.
- **Institutional and Goals Commitment** [Pascarella and Terenzini, 1980] is a 6-item subscale from a 35-item measure assessing academic/social integration and institutional/goal commitment (5 subscales total). The measure had with Likert scaling ranging from 1 (*strongly disagree*) to 5 (*strongly agree*). Higher scores on the institutional and goals commitment subscale indicate greater intentions to persist in college. Data were simulated using factor loadings in the source article.
- **GAD-7** [Spitzer et al., 2006] is a 7-item scale with Likert scaling ranging from 0 (*not at all*) to 3 (*nearly every day*). Higher scores indicate more anxiety. I simulated data by estimating factor loadings from Brattmyr et al. [2022].
- **PHQ-9** [Kroenke et al., 2001] is a 9-item scale with Likert scaling ranging from 0 (*not at all*) to 3 (*nearly every day*). Higher scores indicate higher levels of depression. I simulated data by estimating factor loadings from Brattmyr et al. [2022].

#Entering the intercorrelations, means, and standard deviations from the journal article

```
Szymanski_generating_model <- '
  #measurement model
  CollegeResponse  =~ .88*cold + .73*unresponsive + .73*supportive
  Stigma =~ .86*negative + .76*heterosexism + .71*harassed
  Victimization =~ .8*Vic1 + .8*Vic2 + .8*Vic3 + .8*Vic4 + .8*Vic5 + .8*Vic6 + .8*Vic7 +
  CollSat =~ .8*Sat1 + .8*Sat2 + .8*Sat3 + .8*Sat4 + .8*Sat5
  Persistence =~ .69*graduation_importance + .63*right_decision + .62*will_register + .58*will_study +
  Anxiety =~ .851*nervous + .887*worry_control + .894*much_worry + 674*cant_relax + .484*will_study
```

```

Depression =~ .798*anhedonia + .425*down + .591*sleep + .913*lo_energy + .441*appetite

#Means
CollegeResponse ~ 2.71*1
Stigma ~3.61*1
Victimization ~ 0.11*1
CollSat ~ 5.61*1
Persistence ~ 4.41*1
Anxiety ~ 1.45*1
Depression ~1.29*1

#Correlations
CollegeResponse ~~ .58*Stigma
CollegeResponse ~~ -.25*Victimization
CollegeResponse ~~ -.59*CollSat
CollegeResponse ~~ -.29*Persistence
CollegeResponse ~~ .17*Anxiety
CollegeResponse ~~ .18*Depression

Stigma ~~ .37*Victimization
Stigma ~~ -.41*CollSat
Stigma ~~ -.19*Persistence
Stigma ~~ .27*Anxiety
Stigma ~~ .24*Depression

Victimization ~~ -.22*CollSat
Victimization ~~ -.04*Persistence
Victimization ~~ .23*Anxiety
Victimization ~~ .21*Depression

CollSat ~~ .53*Persistence
CollSat ~~ -.29*Anxiety
CollSat ~~ -.32*Depression

Persistence ~~ -.22*Anxiety
Persistence ~~ -.26*Depression

Anxiety ~~ .76*Depression
'

set.seed(240218)
dfSzy <- lavaan::simulateData(model = Szymanski_generating_model,
                               model.type = "sem",
                               meanstructure = T,
                               sample.nobs=646,
                               standardized=FALSE)

```

```

#used to retrieve column indices used in the rescaling script below
col_index <- as.data.frame(colnames(dfSzy))

#The code below loops through each column of the dataframe and assigns the scaling accordingly
#Rows 1 thru 6 are the Perceptions of LGBTQ Campus Climate Scale
#Rows 7 thru 15 are the Sexual Orientation-Based Campus Victimization Scale
#Rows 16 thru 20 are the College Satisfaction Scale
#Rows 21 thru 26 are the Institutional and Goals Commitment Scale
#Rows 27 thru 33 are the GAD7
#Rows 34 thru 42 are the PHQ9

for(i in 1:ncol(dfSzy)){
  if(i >= 1 & i <= 6){
    dfSzy[,i] <- scales::rescale(dfSzy[,i], c(1, 7))
  }
  if(i >= 7 & i <= 15){
    dfSzy[,i] <- scales::rescale(dfSzy[,i], c(0, 3))
  }
  if(i >= 16 & i <= 20){
    dfSzy[,i] <- scales::rescale(dfSzy[,i], c(1, 7))
  }
  if(i >= 21 & i <= 26){
    dfSzy[,i] <- scales::rescale(dfSzy[,i], c(1, 5))
  }
  if(i >= 27 & i <= 33){
    dfSzy[,i] <- scales::rescale(dfSzy[,i], c(0, 3))
  }
  if(i >= 34 & i <= 42){
    dfSzy[,i] <- scales::rescale(dfSzy[,i], c(0, 3))
  }
}

#rounding to integers so that the data resembles that which was collected
library(tidyverse)
dfSzy <- dfSzy %>% round(0)

#quick check of my work
#psych::describe(dfSzy)

#Reversing the supportive item on the Perceptions of LGBTQ Campus Climate Scale so that the exercises
dfSzy <- dfSzy %>%
  dplyr::mutate(supportiveNR = 8 - supportive)

#Reversing three items on the Institutional and Goals Commitments scale so that the exercises
dfSzy <- dfSzy %>%

```

```
dplyr::mutate(not_graduateNR = 8 - not_graduate)%>%
dplyr::mutate(undecidedNR = 8 - undecided)%>%
dplyr::mutate(grades_unimportantNR = 8 - grades_unimportant)

dfSzy <- dplyr::select(dfSzy, -c(supportive, not_graduate, undecided, grades_unimportant))
```

The optional script below will let you save the simulated data to your computing environment as either an .rds object (preserves any formatting you might do) or a.csv file (think “Excel lite”).

```
# to save the df as an .rds (think 'R object') file on your computer;
# it should save in the same file as the .rmd file you are working
# with saveRDS(dfSzy, 'SzyDF.rds') bring back the simulated dat from
# an .rds file dfSzy <- readRDS('SzyDF.rds')

# write the simulated data as a .csv write.table(dfSzy,
# file='SzyDF.csv', sep=',', col.names=TRUE, row.names=FALSE) bring
# back the simulated dat from a .csv file
dfSzy <- read.csv("SzyDF.csv", header = TRUE)
```

As we move into the lecture, allow me to provide a content advisory. Individuals who hold LGBTQIA+ identities are frequently the recipients of discrimination and harassment. If you are curious about why these items are considered to be stigmatizing or non-responsive, please do not ask a member of the LGBTQIA+ community to explain it to you; it is not their job to educate others on discrimination, harassment, and microaggressions. Rather, please read the article in its entirety. Additionally, resources such as [The Trevor Project](#), [GLSEN](#), and [Campus Pride](#) are credible sources of information for learning more.

## 4.3 Fundamentals of Validity

**Validity** (the classic definition) is the ability of a test to measure what it purports to measure. Supporting that definition are these notions:

- Validity is extent of matching, congruence, or “goodness of fit” between the operational definition and concept it is supposed to measure.
- An instrument is said to be valid if it taps the concept it claims to measure.
- Validity is the appropriateness of the interpretation of the results of an assessment procedure for a given group of individuals, not to the procedure itself.
- Validity is a matter of degree; it does not exist on an all-or-none basis.
- Validity is always specific to some particular use or interpretation.
- Validity is a unitary concept.
- Validity involves an overall evaluative judgment.

Over the years (and, perhaps within each construct), validity has somewhat of an *evolutionary* path from a focus on content, to prediction, to theory and hypothesis testing.

When the focus is on **content**, we are concerned with the:

- Assessment of what individuals had learned in specific content areas.
- Relevance of its content (i.e., we compare the content to the content domain).

When the focus is on **prediction**, we are concerned with:

- How different persons respond in a given situation (now or later).
- The correlation coefficient between test scores (predictor) and the assessment of a criterion (performance in a situation)

A focus on **theory and hypothesis testing** adds:

- A strengthened theoretical orientation.
- A close linkage between psychological theory and verification through empirical and experimental hypothesis testing.
- An emphasis on constructs in describing and understanding human behavior.

**Constructs** are broad categories, derived from the common features shared by directly observable behavioral variables. They are theoretical entities and not directly observable. **Construct validity** is at the heart of psychometric evaluation. We define **construct validity** as the fundamental and all-inclusive validity concept, insofar as it specifies what the test measures. Content and predictive validation procedures are among the many sources of information that contribute to the understanding of the constructs assessed by a test.

## 4.4 Validity Criteria

We have just stated that validity is an overall, evaluative judgment. Within that umbrella are different criteria by which we judge the validity of a measure. We casually refer to them as *types*, but each speaks to that unitary concept.

### 4.4.1 Content Validity

Content validity is concerned with the representativeness of the domain being assessed. Content validation procedures may differ depending on whether the test is in the educational/achievement context or if it is more of an attitude/behavioral survey.

In the educational/achievement context, content validation seeks to ensure the items on an exam are appropriate for the content domain being assessed.

A **table of specifications** is a two-way chart which indicates the instructionally relevant learning tasks to be measured. Percentages in the table indicate the relative degree of emphasis that each content area.

Let's imagine that I was creating a table of specifications for items on a quiz for this very chapter. The columns represent the types of outcomes we might expect. The American Psychological Association often talks about *KSAs* (knowledge, skills, attitudes), so I will utilize those as a framework. You'll notice that the number of items and percentages do not align mathematically. Because,

in the exam, I would likely weight application items (e.g., “work the problem”) more highly than knowledge items (e.g., multiple choice, true/false), the relative weighting may differ.

### Table of Specifications

Learning Objectives	Knowledge	Skills	Attitudes	% of test
Distinguish between different types of validity based on short descriptions.	6 items			30%
Compute and interpret validity coefficients.		2 items		15%
Evaluate the incremental validity of an instrument-of-interest.		1 item		20%
Define and interpret the standard error of estimate.	1 item			15%
Develop a rationale that defends importance of establishing the validity of a measuring instrument.			1 item	20%
<b>TOTALS</b>	<b>7 items</b>	<b>3 items</b>	<b>1 item</b>	<b>100%</b>

**Subject matter experts** (SMEs) are individuals chosen to evaluate items based on their degree of knowledge of the subject being assessed. If SMES are used, the researcher should:

- Report how many SMEs and list their professional qualifications.
- Report any directions the SMEs were given; if they were used to evaluate items, report the extent of agreement.

Empirical procedures for enhancing content validity of educational assessments may include:

- Comparing item-level and total scores with grades; lower grades should get lower scores.
- Analyzing individual errors.
- Observing student work methods (have the students “think aloud” in front of an examiner).
- Evaluating the role of speed, noting how many do not complete the test in the time allowed.
- Correlating the scores with a reading comprehension test (if the exam is highly correlated, then it may be a test of reading and not another subject). Alternatively, if it is a reading comprehension test, give the student the questions (without the passage) to see how well they answered the questions on the basis of prior knowledge.

For surveys and tests outside of educational settings, content validation procedures ask, “Does the test cover a representative sample of the specified skills and knowledge?” and “Is test performance reasonably free from the influence of irrelevant variables?” Naturally, SMEs might be used.

An example of content validation from Industrial-Organizational Psychology is the job analysis which precedes the development of test for employee selection and classification. Not all tests require content analysis. In aptitude and personality tests we are probably more interested in other types of validity evaluation.

#### 4.4.2 Face Validity: The “Un”validity

Face validity is concerned with the question, “How does an assessment look on the ‘face of it’?” Let’s imagine that on a qualification exam for electricians, a math item asks the electrician to estimate the amount of yarn needed to complete a project. The item may be more *face valid* if the calculation was with wire. Thus, face validity can often be improved by reformulating test items in terms that appear relevant and plausible for the context.

Face validity should never be regarded as a substitute for objectively determined validity. In contrast, it should not be assumed that when a (valid and reliable) test has been modified to increase its face validity, that its objective validity and reliability is unaltered. That is, it should be reevaluated.

#### 4.4.3 Criterion-Related Validity

Criterion-related validity has to do with the test’s ability to *predict* an outcome (the criterion). If the criterion is something that occurs simultaneously, it is an assessment of **concurrent validity**; if it is in the future, it is an assessment of **predictive validity**.

A **criterion** is the “thing” that the test should, theoretically, be able to *predict*. That prediction could be occurring at the same time (*concurrent validity*) or at a future time (*predictive validity*). Regardless, the estimate of the criteria must be independent of the survey/assessment being evaluated. The table below provides examples of types of tests and concurrent and predictive validity criteria.

Type of Test	Concurrent Criteria Example	Predictive Criteria Example
A shorter (or cheaper) standardized achievement test	school grades, existing standardized tests	subsequent graduation/college admissions, cumulative GPA
Employee selection tests	decision made by a search committee	subsequent retention or promotion of the selected employee
Assessment of depression severity (shorter or cheaper)	diagnosis from a mental health professional; correlation with an established measure	inpatient hospitalization or act of self-harm

**Contrasted groups** is a specific type of criterion-related validity. Clearly differentiated groups (e.g., salesclerks versus executives; engineers versus musicians) are chosen to see if exam performance or profiles differ in predictable ways.

**Criterion contamination** occurs when test scores, themselves, are used to make decisions about the criteria. To prevent this:

- No person who participates in the assignment of criterion ratings can have any knowledge of the examinee’s test scores.
- The test scores must be kept strictly confidential.

There are a number of issues related to criterion-related validity.

- Is the criterion choice appropriate?
  - Criterion validity is only as good as the validity of the criterion to which one is making a comparison.
  - In the 1980s and 1990s there was more attention in this area; that is critics questioned the quality of the criterion being used.
- To what degree can the results of criterion-related validity be generalized?
  - Most tests are developed (intentionally) for a local context, setting, or audience. Consequently, in the local context, the criterion-prediction sample is usually too small (i.e., 50 cases or less).
  - Those who want to generalize the test to a broader population should evaluate the test in relation to the new purpose.
- Is there a role for meta-analysis?
  - Repeated validation studies of our tests, on different samples, results in a number of small-scale studies, each with their own validity coefficients.
  - We can use meta-analytic procedures in reporting the results of validity coefficients when they are used for establishing criterion validity.

#### 4.4.4 Construct Validity

**Construct validity** was introduced in 1954 in the first edition of APA's testing standards and is defined as the extent to which the test may be said to measure a theoretical construct or trait. The overarching focus is on the role of *psychological theory* in test construction and the ability to formulate hypotheses that can be supported (or not) in the evaluation process. Construct validity is established by the accumulation of information from a variety of sources.

There are a number of sources that can be used to support construct validity.

#### 4.4.5 Internal Consistency

In the next **chapter**, you will learn that **internal consistency** is generally considered to be an index of reliability. In the context of criterion-related validity, a goal is to ensure that the criterion is the total score on the test itself. To that end, some of the following could also support this aspect of validity:

- Comparing high and low scorers. Items that fail to show a significantly greater proportion of "passes" in the upper than the lower group are considered invalid and are modified or eliminated.
- Computing a biserial correlation between the item and total score.
- Correlating the subtest score with the total score. Any subtest whose correlation with the total score is too low is eliminated.

Although some take issue with this notion, the degree of *homogeneity* (the degree to which items assess the same thing) has some bearing on construct validity. There is a tradeoff between items that measure a narrow slice of the construct definition (internal consistency estimates are likely to be higher) and those that sample the construct definition more broadly (internal consistency estimates are likely to be lower).

Admittedly, the contribution of internal consistency data is limited. In absence of external data, it tells us little about WHAT the test measures.

#### 4.4.6 Structural Validity

##### 4.4.6.1 Exploratory Factor Analysis

**Exploratory factor analysis** (EFA) is used to simplify the description of behavior by reducing the number of categories (factors or dimensions) to fewer than the number of items. In our research vignette the 6-item Perceptions of Campus Climate Scale will be represented by two factors [Szymanski and Bissonette, 2020] In instrument development, techniques like *principal components analysis* or *principal axis factoring* are used to identify clusters (latent factors) among items. We frequently treat these as scales and subscales.

Imagine the use of 20 tests to 300 people. There would be 190 correlations.

- Irrespective of content, we can probably summarize the intercorrelations of tests with 5-6 factors.
- When the clustering of tests includes vocabulary, analogies, opposites, and sentence completions, we might suggest a “verbal comprehension factor.”
- Factorial validity is the correlation of the test with whatever is common to a group of tests or other indices of behavior. If our single test has a correlation of .66 with the factor on which it loads, then the “factorial validity of the new test as a measure of the common trait is .66.”

When EFA is utilized, the items are “fed” into an iterative process that analyzes the relations and “reveals” (or suggests – we are the ones who interpret the data) how many factors (think scales/subscales) and which items comprise them.

##### 4.4.6.2 Confirmatory Factor Analysis

**Confirmatory factor analysis** (CFA) involves specifying, a priori, a proposed relationship of items, scales, and subscales and then testing its *goodness of fit*. In CFA (a form of structural equation modeling [SEM]), the latent variables (usually the higher order scales and total scale score) are positioned to *cause* the responses on the indicators/items.

Subsequent lessons provide examples of both EFA and CFA approaches to psychometrics.

#### 4.4.7 Experimental Interventions

Construct validity is also supported by hypothesis testing and experimentation. If we expect that the construct assessed by the instrument is malleable (e.g., depression) and that an intervention

could change it, then a random clinical trial that evaluated the effectiveness of an intervention (and it worked – depression scores declined) would simultaneously provide support for the intervention as well as the instrument.

#### 4.4.8 Convergent and Discriminant Validity

In a psychometric evaluation, we will often administer our instrument-of-interest along with a battery of instruments that are more-and-less related. **Convergent validity** is supported when there are *moderately high* correlations between our tests and the instruments with which we expect moderately high correlations. In contrast, **discriminant validity** is established by low and/or non-significant correlations between our instrument-of-interest and instruments that should be unrelated. For example, we want a low and non-significant correlation between a quantitative reasoning test and scores on a reading comprehension test. Why? Because if the correlation is too high, the test cannot discriminate between reading comprehension and math.

There are no strict cut-offs to establish convergence or discrimination. We can even ask, “Could a correlation intended to support convergence be too high?” It is possible! Unless the instrument-of-interest offers advantages such as brevity or cost, then correlations that fall into the ranges of multicollinearity or singularity can indicate unnecessary duplication or redundancy.

In our research vignette, Szymanski and Bissonnette [2020] conducted a correlation matrix that reports the bivariate relations between the LGBTQ Campus Climate full-scale as well as the College Response and Stigma subscales with measures that assess (a) LGBTQ victimization, (b) satisfaction with college, (c) persistence attitudes, and (d) anxiety, and (e) depression.

In order to produce this correlation matrix, we must first score each of the scales. In the items we prepared, the Perceptions of LGBTQ Campus Climate scale had one reverse-scored item. Similarly, the Institutional Goals and Commitments Scale had three reversed items. A first step in scoring is reversing these items.

The naming conventions that researchers use vary. I added an “NR” (for “needs reversing”) to the original items so that I would remember to reverse-score them. I also am careful to reverse-score items into new variables. Otherwise, we risk getting confused about whether/not items are in their original or reversed formats.

Reverse-scoring the item is easily accomplished by subtracting the variable from “one plus” the scaling. Because both of these scales were on a 7-point scale, we will subtract the “NR” variables from 8.

```
# Reverse scoring the single item from the LGBTQ Campus Climate Scale
dfSzy <- dfSzy %>%
  dplyr::mutate(unsupported = 8 - supportiveNR)

# Reversing three items on the Institutional and Goals Commitments
# scale

dfSzy <- dfSzy %>%
  dplyr::mutate(not_graduate = 8 - not_graduateNR) %>%
  dplyr::mutate(undecided = 8 - undecidedNR) %>%
  dplyr::mutate(grades_unimportant = 8 - grades_unimportantNR)
```

Next, we create scale and/or subscale scores. The `sjstats::mean_n()` function allows us to specify how many items (whole number) or what percentage of items should be present in order to get the mean. It is customary to require 75-80% of items to be present for scoring. Three-item variables might allow one missing (i.e., 66%). In the code below, I first make lists of the variables that belong in each scale and subscale, then I create the new variables.

```
# Making the list of variables
LGBTQ_Climate <- c("cold", "unresponsive", "unsupportive", "negative",
  "heterosexism", "harassed")
CollResponse <- c("cold", "unresponsive", "unsupportive")
Stigma <- c("negative", "heterosexism", "harassed")
Victimization <- c("Vic1", "Vic2", "Vic3", "Vic4", "Vic5", "Vic6", "Vic7",
  "Vic8", "Vic9")
CampSat <- c("Sat1", "Sat2", "Sat3", "Sat4", "Sat5")
Persist <- c("graduation_importance", "right_decision", "will_register",
  "not_graduate", "undecided", "grades_unimportant")
GAD7 <- c("nervous", "worry_control", "much_worry", "cant_relax", "restless",
  "irritable", "afraid")
PHQ9 <- c("anhedonia", "down", "sleep", "lo_energy", "appetite", "selfworth",
  "concentration", "too_slowfast", "s_ideation")

# Creating the new variables
dfSzy$LGBTQclimate <- sjstats::mean_n(dfSzy[, LGBTQ_Climate], 0.75)
dfSzy$CollegeRx <- sjstats::mean_n(dfSzy[, CollResponse], 0.66)
dfSzy$Stigma <- sjstats::mean_n(dfSzy[, Stigma], 0.66)
dfSzy$Victimization <- sjstats::mean_n(dfSzy[, Victimization], 0.8)
dfSzy$CampusSat <- sjstats::mean_n(dfSzy[, CampSat], 0.75)
dfSzy$Persistence <- sjstats::mean_n(dfSzy[, Persist], 0.8)
dfSzy$Anxiety <- sjstats::mean_n(dfSzy[, GAD7], 0.75)
dfSzy$Depression <- sjstats::mean_n(dfSzy[, PHQ9], 0.8)

# If the scoring code above does not work for you, try the format
# below which involves inserting two periods in front of the variable
# list. One example is provided. dfLewis$Belonging <-
# sjstats::mean_n(dfLewis[, ..Belonging_vars], 0.80)
```

A correlation matrix of the scaled scores allows us to compare our scale(s) of interest to others within its nomological net.

```
apaTables::apa.cor.table(dfSzy[c("LGBTQclimate", "CollegeRx", "Stigma",
  "Victimization", "CampusSat", "Persistence", "Anxiety", "Depression")],
  filename = "SzyCor.doc", table.number = 1, show.sig.stars = TRUE, landscape = TRUE)
```

Table 1

Means, standard deviations, and correlations with confidence intervals

.15**	-.23**	-.10**	.54**
[.08, .23]	[-.30, -.15]	[-.18, -.03]	[.48, .59]

Note. M and SD are used to represent mean and standard deviation, respectively.

Values in square brackets indicate the 95% confidence interval.

The confidence interval is a plausible range of population correlations that could have caused the sample correlation (Cumming, 2014).

\* indicates  $p < .05$ . \*\* indicates  $p < .01$ .

Examination of these values follow some expected patterns. First, the LGBTQ climate score (i.e., the total scale score) is highly correlated with each of its subscales (College Response  $r = .83$ ,  $p < 0.01$ ; Stigma  $r = .83$ ,  $p = 0.01$ ). These strong correlations are somewhat misleading because half of the items on the total scale are the same items on each of the subscales. The correlation between the two subscales is still statistically significant, but much lower ( $r = 0.37$ ,  $p = 0.01$ ).

Convergent and discriminant validity are of interest when we compare the LGBTQ Climate total scale score and the College Response and Stigma subscales with the additional measures. Regarding the total LGBTQ Climate score, a very strong correlation was observed with campus satisfaction ( $r = -0.49$ ,  $p < 0.01$ ); less strong correlations were observed with and persistence ( $r = -0.21$ ,  $p < 0.01$ ), anxiety ( $r = 0.17$ ,  $p < 0.01$ ), and depression ( $r = 0.18$ ,  $p < 0.01$ ). Recalling that higher scores on the LGBTQ Campus Climate score indicate a negative climate, we see that as the LGBTQ campus climate becomes increasingly stigmatizing and nonresponsive, students experience lower overall campus satisfaction and are less likely to persist at that institution. The correlation between LGBTQ Campus Climate and victimization was non-significant ( $r = 0.01$ ,  $p > 0.05$ ).

In assessing patterns of convergent and discriminant validity, the researcher would also take the time to map out the subscales (i.e., College Response, Stigma) with the additional measures.

#### 4.4.8.1 Determining Statistically Significant Differences Between Correlations

Without a formal test, it is inappropriate for researchers to declare that one correlation is stronger than another. The package *cocor* allows the comparisons of *dependent* (i.e., all respondents are from the same sample) and *independent* (i.e., correlations are compared between two different samples) where the correlations themselves can be *overlapping* (i.e., with one common variable) or *non-overlapping* (i.e., the variables in both correlations are different).

Because all of the correlations we computed are within the same sample, they are *dependent*. When assessing convergent and discriminant validity it is common to ask if the correlations between the additional measures are different between the subscales of the focal measure. Results could support the notion that the subscales are related and yet different. An example of this might be comparing the correlations between victimization with the College Response subscale ( $r = -.17$ ,  $p < 0.01$ ) and victimization with Stigma subscale ( $r = 0.20$ ,  $p < 0.01$ ). This test would be *overlapping* because the victimization variable is in common. Here's the code:

```
cocor::cocor(formula = ~CollegeRx + Victimization | Stigma + Victimization,
             data = dfSzy)
```

Results of a comparison of two overlapping correlations based on dependent groups

Comparison between r.jk (Victimization, CollegeRx) = -0.1741 and r.jh (Victimization, Stigma) =  
Difference: r.jk - r.jh = -0.3756  
Related correlation: r.kh = 0.3734  
Data: dfSzy: j = Victimization, k = CollegeRx, h = Stigma  
Group size: n = 646  
Null hypothesis: r.jk is equal to r.jh  
Alternative hypothesis: r.jk is not equal to r.jh (two-sided)  
Alpha: 0.05

pearson1898: Pearson and Filon's z (1898)  
z = -8.9455, p-value = 0.0000  
Null hypothesis rejected

hotelling1940: Hotelling's t (1940)  
t = -9.0342, df = 643, p-value = 0.0000  
Null hypothesis rejected

williams1959: Williams' t (1959)  
t = -9.0340, df = 643, p-value = 0.0000  
Null hypothesis rejected

olkin1967: Olkin's z (1967)  
z = -8.9455, p-value = 0.0000  
Null hypothesis rejected

dunn1969: Dunn and Clark's z (1969)  
z = -8.7137, p-value = 0.0000  
Null hypothesis rejected

hendrickson1970: Hendrickson, Stanley, and Hills' (1970) modification of Williams' t (1959)  
t = -9.0341, df = 643, p-value = 0.0000  
Null hypothesis rejected

steiger1980: Steiger's (1980) modification of Dunn and Clark's z (1969) using average correlation  
z = -8.6124, p-value = 0.0000  
Null hypothesis rejected

meng1992: Meng, Rosenthal, and Rubin's z (1992)  
z = -8.5080, p-value = 0.0000  
Null hypothesis rejected  
95% confidence interval for r.jk - r.jh: -0.4678 -0.2926  
Null hypothesis rejected (Interval does not include 0)

hittner2003: Hittner, May, and Silver's (2003) modification of Dunn and Clark's z (1969) using  
z = -8.6124, p-value = 0.0000

Null hypothesis rejected

```

zou2007: Zou's (2007) confidence interval
 95% confidence interval for r.jk - r.jh: -0.4568 -0.2921
 Null hypothesis rejected (Interval does not include 0)

```

Fisher's z-test ( $z = -8.6124$ ,  $p < 0.001$ ) tells us that the correlations are statistically significantly different from each other; Zhou's confidence intervals provide the CI for the size of the difference between the two correlations. That is, the difference could be as large as -0.4568 or as small as -0.2921.

Another type of correlation comparison is with a the total and/or subscales, looking at the relative magnitude of their correlation with different variables. For example, we might wish to ask if the LGBTQ Campus Climate total scale is different degrees of correlation with anxiety ( $r = 0.17$ ,  $p < 0.01$ ) and depression ( $r = .18$ ,  $p < 0.01$ ).

```

cocor::cocor(formula = ~LGBTQclimate + Anxiety | LGBTQclimate + Depression,
  data = dfSzy)

```

Results of a comparison of two overlapping correlations based on dependent groups

```

Comparison between r.jk (LGBTQclimate, Anxiety) = 0.1724 and r.jh (LGBTQclimate, Depression) =
Difference: r.jk - r.jh = -0.0071
Related correlation: r.kh = 0.5387
Data: dfSzy: j = LGBTQclimate, k = Anxiety, h = Depression
Group size: n = 646
Null hypothesis: r.jk is equal to r.jh
Alternative hypothesis: r.jk is not equal to r.jh (two-sided)
Alpha: 0.05

```

```

pearson1898: Pearson and Filon's z (1898)
  z = -0.1914, p-value = 0.8482
  Null hypothesis retained

```

```

hotelling1940: Hotelling's t (1940)
  t = -0.1911, df = 643, p-value = 0.8485
  Null hypothesis retained

```

```

williams1959: Williams' t (1959)
  t = -0.1909, df = 643, p-value = 0.8487
  Null hypothesis retained

```

```

olkin1967: Olkin's z (1967)
  z = -0.1914, p-value = 0.8482
  Null hypothesis retained

```

```
dunn1969: Dunn and Clark's z (1969)
```

$z = -0.1909$ , p-value = 0.8486  
 Null hypothesis retained

hendrickson1970: Hendrickson, Stanley, and Hills' (1970) modification of Williams' t (1959)  
 $t = -0.1911$ , df = 643, p-value = 0.8485  
 Null hypothesis retained

steiger1980: Steiger's (1980) modification of Dunn and Clark's z (1969) using average correlation  
 $z = -0.1909$ , p-value = 0.8486  
 Null hypothesis retained

meng1992: Meng, Rosenthal, and Rubin's z (1992)  
 $z = -0.1909$ , p-value = 0.8486  
 Null hypothesis retained  
 95% confidence interval for  $r_{jk} - r_{jh}$ : -0.0825 0.0678  
 Null hypothesis retained (Interval includes 0)

hittner2003: Hittner, May, and Silver's (2003) modification of Dunn and Clark's z (1969) using average correlation  
 $z = -0.1909$ , p-value = 0.8486  
 Null hypothesis retained

zou2007: Zou's (2007) confidence interval  
 95% confidence interval for  $r_{jk} - r_{jh}$ : -0.0798 0.0656  
 Null hypothesis retained (Interval includes 0)

Fisher's z-test ( $z = -0.1909$ ,  $p = 0.8486$ ) tells us that the correlations are not statistically significantly different from each other; Zhou's confidence intervals indicate that the differences range between -0.0798 and 0.0656. Because this interval crosses zero, we know that the difference could be zero, or reversed in direction.

#### 4.4.8.2 Multitrait-Multimethod Matrix

The **multitrait-multimethod matrix** is a systematic experimental design for the dual approach of convergent and discriminant validation, which requires the assessment of two or more traits (classically, math, English, and reading scores) by two or more methods (self, parent, and teacher). Conducting a web-based image search on this term will show a matrix of alpha coefficients and correlation coefficients that are interpreted in relationship to each other. Roughly:

- alpha coefficients (internal consistency) should be the highest,
- validity coefficients (correlations of the same trait assessed by different methods) should be higher than correlations between different traits measured by different methods,
- validity coefficients (correlations of the same trait assessed by different methods) should be higher than different traits measured by the same method.

#### 4.4.9 Incremental Validity

**Incremental validity** is the increase in predictive validity attributable to the test. It indicates the contribution the test makes to the selection of individuals who will meet the minimum standards in criterion performance. There are different ways to assess this – one of the most common is to first enter known predictors and then see if the instrument-of-interest continues to account variance over-and-above those that are entered.

In the Szymanski and Bissonette [2020] psychometric evaluation, the negative relations with satisfaction with college and intention to persist in college as well as positive relations with both anxiety and depression persisted even after controlling for LGBTQ victimization experiences.

I will demonstrate this procedure, predicting the contribution that the LGBTQ Campus Climate total scale score has on predicting intention to persist in college, over and above LGBTQ victimization.

The process is to use hierarchical linear regression. Two models are built. In the first mode (“PfV” stands [in my mind] for “Persistence from Victimization”), persistence is predicted from victimization. The second model adds the LGBTQ Campus Climate Scale. I asked for summaries of each model. Then the *anova()* function compares the model.

```
PfV <- lm(Persistence ~ Victimization, data = dfSzy)
PfVC <- lm(Persistence ~ Victimization + LGBTQclimate, data = dfSzy)
summary(PfV)
```

Call:

```
lm(formula = Persistence ~ Victimization, data = dfSzy)
```

Residuals:

Min	1Q	Median	3Q	Max
-1.19281	-0.34150	-0.02281	0.29669	1.32226

Coefficients:

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	3.09906	0.07947	38.997	<0.0000000000000002 ***
Victimization	-0.04566	0.05023	-0.909	0.364
---				
Signif. codes:	0 '***'	0.001 '**'	0.01 '*'	0.05 '.'
	0.1	' '	1	

Residual standard error: 0.4224 on 644 degrees of freedom

Multiple R-squared: 0.001281, Adjusted R-squared: -0.0002694

F-statistic: 0.8263 on 1 and 644 DF, p-value: 0.3637

From the PfV model we learn that victimization has a non-significant effect on intentions to persist in college ( $B = -0.046$ ,  $p = 0.364$ ). Further, the  $R^2$  is quite small (0.001).

```
summary(PfVC)
```

Call:

```
lm(formula = Persistence ~ Victimization + LGBTQclimate, data = dfSzy)
```

Residuals:

Min	1Q	Median	3Q	Max
-1.1696	-0.2842	0.0094	0.2569	1.3571

Coefficients:

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	3.64440	0.12795	28.483 < 0.0000000000000002	***
Victimization	-0.04183	0.04919	-0.850	0.395
LGBTQclimate	-0.13788	0.02568	-5.369	0.000000111 ***
---				
Signif. codes:	0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1			

Residual standard error: 0.4135 on 643 degrees of freedom

Multiple R-squared: 0.04413, Adjusted R-squared: 0.04116

F-statistic: 14.84 on 2 and 643 DF, p-value: 0.0000004991

In the PfVC model, we see that the LGBTQ Campus Climate full scale score has a significant impact on intentions to persist. Specifically, for each additional point higher on the LGBTQ climate score, intentions to persist decrease by .14 points ( $p < 0.001$ ). Together, the model accounts for 4% of the variance, representing a  $\Delta R^2$  of 4%.

```
# calculating R2 change
```

```
0.04413 - 0.001281
```

```
[1] 0.042849
```

```
anova(PfV, PfVC)
```

Analysis of Variance Table

Model 1: Persistence ~ Victimization

Model 2: Persistence ~ Victimization + LGBTQclimate

Res.Df	RSS	Df	Sum of Sq	F	Pr(>F)
1	644	114.88			
2	643	109.95	1	4.929	28.824 0.0000001108 ***
---					

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

We see that there is a statistically significant difference between the models  $F(1, 643) = 28.824, p < 0.001$ .

Let's try another model. With anxiety as our dependent variable, the code below asks if LGBTQ Campus Climate accounts for a proportion of the variance over-and-above victimization.

```
AfV <- lm(Anxiety ~ Victimization, data = dfSzy)
AfVC <- lm(Anxiety ~ Victimization + LGBTQclimate, data = dfSzy)
summary(AfV)
```

Call:

```
lm(formula = Anxiety ~ Victimization, data = dfSzy)
```

Residuals:

Min	1Q	Median	3Q	Max
-1.05943	-0.30935	-0.01528	0.31306	1.24148

Coefficients:

	Estimate	Std. Error	t value	Pr(> t )							
(Intercept)	1.21756	0.07131	17.073	< 0.0000000000000002 ***							
Victimization	0.17427	0.04508	3.866	0.000122 ***							
---											
Signif. codes:	0	'***'	0.001	'**'	0.01	'*'	0.05	'..'	0.1	' '	1

Residual standard error: 0.379 on 644 degrees of freedom

Multiple R-squared: 0.02268, Adjusted R-squared: 0.02116

F-statistic: 14.95 on 1 and 644 DF, p-value: 0.0001219

```
summary(AfVC)
```

Call:

```
lm(formula = Anxiety ~ Victimization + LGBTQclimate, data = dfSzy)
```

Residuals:

Min	1Q	Median	3Q	Max
-1.00814	-0.29249	-0.02563	0.29877	1.20705

Coefficients:

	Estimate	Std. Error	t value	Pr(> t )							
(Intercept)	0.81071	0.11561	7.012	0.0000000000595 ***							
Victimization	0.17141	0.04444	3.857	0.000126 ***							
LGBTQclimate	0.10287	0.02320	4.433	0.00001092659570 ***							
---											
Signif. codes:	0	'***'	0.001	'**'	0.01	'*'	0.05	'..'	0.1	' '	1

Residual standard error: 0.3736 on 643 degrees of freedom

Multiple R-squared: 0.05166, Adjusted R-squared: 0.04872

F-statistic: 17.52 on 2 and 643 DF, p-value: 0.0000000392

```
anova(AfV, AfVC)
```

#### Analysis of Variance Table

```
Model 1: Anxiety ~ Victimization
Model 2: Anxiety ~ Victimization + LGBTQclimate
Res.Df   RSS Df Sum of Sq    F    Pr(>F)
1     644 92.513
2     643 89.770  1    2.7436 19.651 0.00001093 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

This model is a little more exciting in that our first model (AfV) is statistically significant ( $B = 0.174, p < 0.001$ ). That is, victimization has a statistically significant effect on anxiety, accounting for 2% of the variance. In the second model, the LGBTQ Campus Climate total scale score is also significant ( $B = 0.103, p < 0.001$ ), and accounts for an additional 3% of variance ( $\Delta R^2 = 0.029$ ). There is a statistically significant difference between models ( $F[1, 643] = 19.651, p < .001$ ).

```
# calculating change in R2
0.05166 - 0.02268
```

```
[1] 0.02898
```

#### 4.4.10 Considering the Individual and Social Consequences of Testing

Messick [Messick, 2000] and others recommend that the “consequences of testing” be included in the concept of test validity. Messick’s point was to consider the unintended consequences of specific uses. That is, their use may be detrimental to individuals or to members of certain ethnic or other populations with diverse experiential backgrounds. Examples of inappropriate use have included:

- The California Psychological Inventory (CPI) being used as a screening tool for employment as a security job. Two of its items inquired about same-sex activities and the employer was using this to screen out gay men. Applicants were able to demonstrate, in court, a consistent rejection of gay applicants.
- While this is not a psychological test, urine samples are often collected as drug screening tools. In reality, urine can reveal a number of things, such as pregnancy.

The issue begs “conflicting goals.” In this case, the problem was not caused by the test but rather by its misuse. Studying the “consequences” of testing is one that is not necessarily answerable by empirical data/statistical analysis. It requires critical observation, human judgment, and systematic debate.

## 4.5 Factors Affecting Validity Coefficients

Keeping in mind that a *validity coefficient* is merely the correlation between the test and some criteria, the same elements that impact the magnitude and significance of a correlation coefficient will similarly affect a validity coefficient.

**Nature of the group.** A test that has high validity in predicting a particular criterion in one population, may have little or no validity in predicting the same criterion in another population. If a test is designed for use in diverse populations, information about the population generalizability should be reported in the technical manuals.

**Sample heterogeneity.** Other things being equal, if there is a linear relationship between X and Y, it will have a greater magnitude when the sample is heterogeneous.

**Pre-selection.** Just like internal and external validity in a research design can be threatened by selection issues, pre-selection can also impact the validity coefficients of a measure. For example, if we are evaluating a new test for job selection, we may select a group of newly hired employees. We plan to collect some measure of job performance at a later date. Our results may be limited by the criteria used to select the employees. Were they volunteers? Were they only those hired? Were they ALL of the applicants?

**Validity coefficients may change over time.** Consider the relationship between the college boards and grade point average at Yale University. Fifty years ago,  $r_{xy} = .72$ ; today  $r_{xy} = .52$ . Why? The nature of the student body has become more diverse (50 years ago, the student body was predominantly White, high SES, and male).

The **form of the relationship** matters. The Pearson R assumes the relationship between the predictor and criterion variables is linear, uniform, and homoscedastic (equal variability throughout the range of a bivariate distribution). When the variability is unequal throughout the range of the distribution the relationship is heteroscedastic.

There could also be other factors involved in the relationship between the instrument and the criterion:

- curvilinearity
- an undetected mechanism, such as a moderator

Finally, what is our threshold for acceptability?

- Consider statistical significance – but also its limitations (e.g., power, Type I error, Type II error)
- Consider the magnitude of the correlation; and also  $R^2$  (the proportion of variance accounted for)
- Consider error:
  - The standard error of the estimate shows the margin of error to be expected in the individuals predicted criterion score as the result of the imperfect validity of the instrument.

$$SE_{est} = SD_y \sqrt{1 - r_{xy}^2}$$

Where:

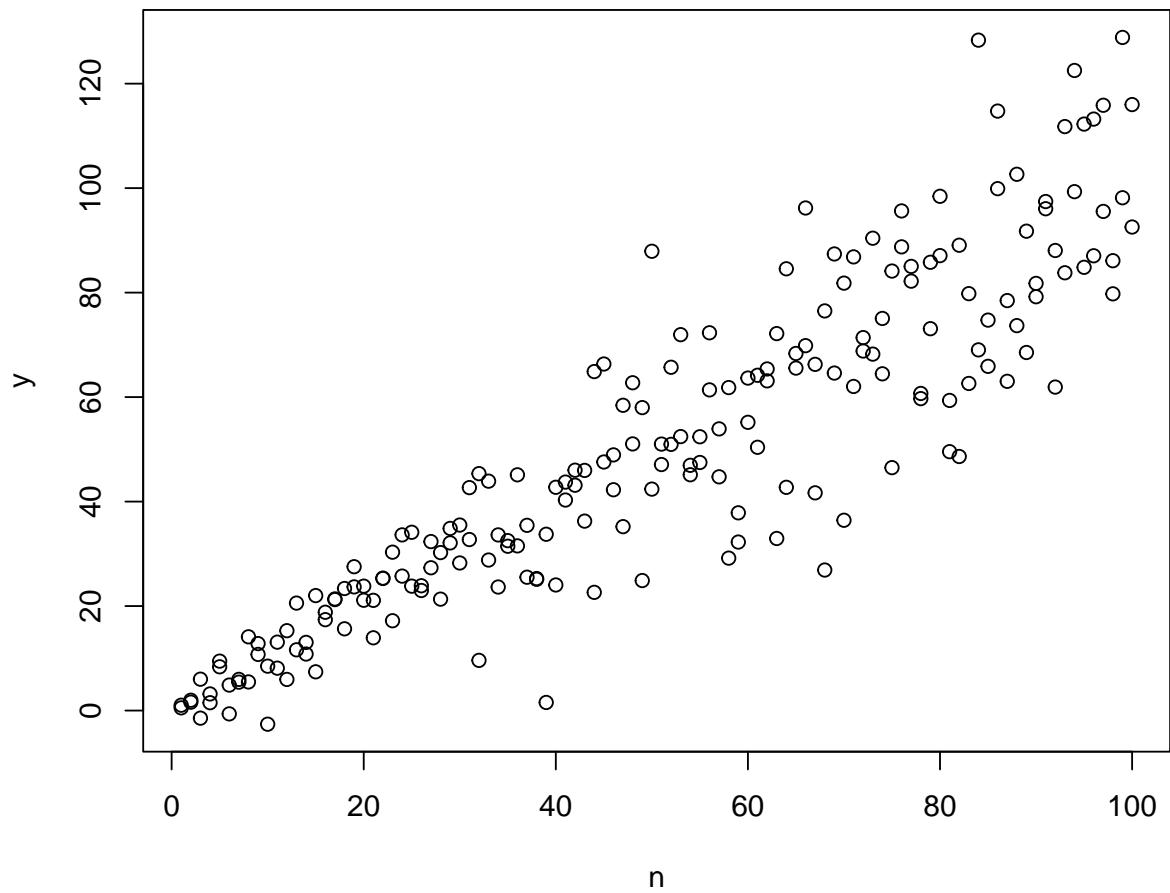


Figure 4.1: Illustration of heteroscedasticity

- $r_{xy}^2$  is the square of the validity coefficient, and
- $SD_y$  is the standard deviation of the criterion scores.

If the validity were perfect ( $r_{xy}^2 = 1.00$ ), the error of estimate would be 0.00. If the validity were zero, the error of estimate would equal  $SD_y$ .

Interpreting  $SE_{est}$ :

- If  $r_{xy} = .80$ , then  $\sqrt{1 - r_{xy}^2} = .60$ .
- Error is 60% as large as it would be by chance. Stated another way, predicting an individual's criterion performance has a margin of error that is 40% smaller than it would be by chance.

To obtain the  $SE_{est}$ , we merely multiply by the  $SD_y$ . This puts error in the metric of the criterion variable.

Your Turn:

- If  $r_{xy} = .25$ , then  $\sqrt{1 - r_{xy}^2} = ??$

Make a statement about chance. Make a statement about margin of error.

## 4.6 Practice Problems

In each of these lessons, I provide suggestions for practice that allow you to select one or more problems that are graded in difficulty. With each of these options, I encourage you to interpret examine aspects of the construct validity through the creation and interpretation of validity coefficients. Ideally, you will examine both convergent/discriminant validity as well as incremental validity.

### 4.6.1 Problem #1: Play around with this simulation.

Copy the script for the simulation and then change (at least) one thing in the simulation to see how it impacts the results.

If calculating is new to you, perhaps you just change the number in “set.seed(210907)” from 210907 to something else. Your results should parallel those obtained in the lecture, making it easier for you to check your work as you go.

### 4.6.2 Problem #2: Conduct the reliability analysis selecting different variables.

The Szymanski and Bissonette [2020] article conducted a handful of incremental validity assessments. Select different outcome variables (e.g., depression) and/or use the subscales as the instrument-of-interest.

#### 4.6.3 Problem #3: Try something entirely new.

Using data for which you have permission and access (e.g., IRB approved data you have collected or from your lab; data you simulate from a published article; data from an open science repository; data from other chapters in this OER), create validity coefficients and use three variables to estimate the incremental validity of the instrument-of-interest.

#### 4.6.4 Grading Rubric

Assignment Component	Points Possible	Points Earned
1. Check and, if needed, format and score the data.	5	_____
2. Create a correlation matrix that includes the instrument-of-interest and the variables that will have varying degrees of relation.	5	_____
3. With convergent and discriminant validity in mind, interpret the validity coefficients; this should include an assessment about whether the correlation coefficients (at least two different pairings) are statistically significantly different from each other.	5	_____
4. With at least three variables, evaluate the degree to which the instrument demonstrates incremental validity (this should involve two regression equations and their statistical comparison).	5	_____
5. Explanation to grader.	5	_____
<b>Totals</b>	<b>25</b>	_____

## 4.7 Homeworked Example

### Screencast Link

For more information about the data used in this homeworked example, please refer to the description and codebook located at the end of the [introduction](#) in first volume of ReCentering Psych Stats.

As a brief review, this data is part of an IRB-approved study, with consent to use in teaching demonstrations and to be made available to the general public via the open science framework. Hence, it is appropriate to use in this context. You will notice there are student- and teacher- IDs. These numbers are not actual student and teacher IDs, rather they were further re-identified so that they could not be connected to actual people.

Because this is an actual dataset, if you wish to work the problem along with me, you will need to download the [ReC.rds](#) data file from the Worked\_Examples folder in the ReC\_Psychometrics project on the GitHub.

The course evaluation items can be divided into three subscales:

- **Valued by the student** includes the items: ValObjectives, IncrUnderstanding, IncrInterest
- **Traditional pedagogy** includes the items: ClearResponsibilities, EffectiveAnswers, Feedback, ClearOrganization, ClearPresentation
- **Socially responsive pedagogy** includes the items: InclusvClassrm, EquitableEval, MultPerspectives, DEIIntegration

In this homework focused on validity we will score the total scale and subscales, create a correlation matrix of our scales with a different scale (or item), formally test to see if correlation coefficients are statistically significantly different from each other, conduct a test of incremental validity.

#### 4.7.1 Check and, if needed, format data

```
big <- readRDS("ReC.rds")
```

Let's check the structure...

```
str(big)
```

```
Classes 'data.table' and 'data.frame': 310 obs. of 33 variables:
 $ deID           : int 1 2 3 4 5 6 7 8 9 10 ...
 $ CourseID       : int 57085635 57085635 57085635 57085635 57085635 57085635 57085635 ...
 $ Dept           : chr "CPY" "CPY" "CPY" "CPY" ...
 $ Course          : Factor w/ 3 levels "Psychometrics",...: 2 2 2 2 2 2 2 2 2 2 ...
 $ StatsPkg        : Factor w/ 2 levels "SPSS","R": 2 2 2 2 2 2 2 2 2 2 ...
 $ Centering       : Factor w/ 2 levels "Pre","Re": 2 2 2 2 2 2 2 2 2 2 ...
 $ Year            : int 2021 2021 2021 2021 2021 2021 2021 2021 2021 2021 ...
 $ Quarter         : chr "Fall" "Fall" "Fall" "Fall" ...
 $ IncrInterest    : int 5 3 4 2 4 3 5 3 2 5 ...
 $ IncrUnderstanding: int 2 3 4 3 4 4 5 2 4 5 ...
 $ ValObjectives   : int 5 5 4 4 5 5 5 5 4 5 ...
 $ ApprAssignments : int 5 4 4 4 5 3 5 3 3 5 ...
 $ EffectiveAnswers: int 5 3 5 3 5 3 4 3 2 3 ...
 $ Respectful      : int 5 5 4 5 5 4 5 4 5 5 ...
 $ ClearResponsibilities: int 5 5 4 4 5 4 5 4 4 5 ...
 $ Feedback         : int 5 3 4 2 5 NA 5 4 4 5 ...
 $ OvInstructor     : int 5 4 4 3 5 3 5 4 3 5 ...
 $ MultPerspectives : int 5 5 4 5 5 4 5 5 5 5 ...
 $ OvCourse          : int 3 4 4 3 5 3 5 3 2 5 ...
 $ InclusvClassrm   : int 5 5 5 5 5 4 5 5 5 4 5 ...
 $ DEIIntegration   : int 5 5 5 5 5 4 5 5 5 5 ...
 $ ClearPresentation: int 4 4 4 2 5 3 4 4 4 5 ...
 $ ApprWorkload     : int 5 5 3 4 4 2 5 4 4 5 ...
 $ MyContribution   : int 4 4 4 4 5 4 4 3 4 5 ...
```

```
$ InspiredInterest      : int  5 3 4 3 5 3 5 4 4 5 ...
$ Faith                 : int  5 NA 4 2 NA NA 4 4 4 NA ...
$ EquitableEval         : int  5 5 3 5 5 3 5 5 3 5 ...
$ SPFC.Decolonize.Opt.Out: chr  "" "" "" ...
$ ProgramYear           : Factor w/ 3 levels "Second","Transition",...: 3 3 3 3 3 3 3 3 3 ...
$ ClearOrganization     : int  3 4 3 4 4 4 5 4 4 5 ...
$ RegPrepare             : int  5 4 4 4 4 3 4 4 4 5 ...
$ EffectiveLearning      : int  2 4 3 4 4 2 5 3 2 5 ...
$ AccessibleInstructor   : int  5 4 4 4 5 4 5 4 5 5 ...
- attr(*, ".internal.selfref")=<externalptr>
```

We will need to create the three subscales. The codebook above, lists which variables go in each subscale score.

```
# Making the list of variables
ValuedVars <- c("ValObjectives", "IncrUnderstanding", "IncrInterest")
TradPedVars <- c("ClearResponsibilities", "EffectiveAnswers", "Feedback",
  "ClearOrganization", "ClearPresentation")
SRPedVars <- c("InclusvClassrm", "EquitableEval", "MultPerspectives", "DEIintegration")
Total <- c("ValObjectives", "IncrUnderstanding", "IncrInterest", "ClearResponsibilities",
  "EffectiveAnswers", "Feedback", "ClearOrganization", "ClearPresentation",
  "InclusvClassrm", "EquitableEval", "MultPerspectives", "DEIintegration")

# Creating the new variables
big$Valued <- sjstats::mean_n(big[, ValuedVars], 0.66)
big$TradPed <- sjstats::mean_n(big[, TradPedVars], 0.75)
big$SRPed <- sjstats::mean_n(big[, SRPedVars], 0.75)
big$Total <- sjstats::mean_n(big[, Total], 0.8)

# If the scoring code above does not work for you, try the format
# below which involves inserting two periods in front of the variable
# list. One example is provided. dfLewis$Belonging <-
# sjstats::mean_n(dfLewis[, ..Belonging_vars], 0.80)
```

#### 4.7.2 Create a correlation matrix that includes the instrument-of-interest and the variables that will have varying degrees of relation

Unfortunately, data from the course evals don't include any outside scales. However, I didn't include the "Overall Instructor" (OvInstructor) in any of the items, so we *could* think of it as a way to look at convergent and discriminant validity.

```
apaTables::apa.cor.table(big[c("Valued", "TradPed", "SRPed", "OvInstructor")],
  filename = "ReC_cortable.doc", table.number = 1, show.sig.stars = TRUE,
  landscape = TRUE)
```

Table 1

Means, standard deviations, and correlations with confidence intervals

Variable	M	SD	1	2	3
1. Valued	4.25	0.68			
2. TradPed	4.25	0.76	.70** [.63, .75]		
3. SRPed	4.52	0.58	.56** [.48, .64]	.71** [.65, .76]	
4. OvInstructor	4.37	0.94	.63** [.56, .70]	.80** [.76, .84]	.67** [.60, .73]

Note. M and SD are used to represent mean and standard deviation, respectively.

Values in square brackets indicate the 95% confidence interval.

The confidence interval is a plausible range of population correlations that could have caused the sample correlation (Cumming, 2014).

\* indicates  $p < .05$ . \*\* indicates  $p < .01$ .

All the correlations are strong and positive. However, look at the correlation between Overall Instructor and SCRPed!

#### 4.7.3 With convergent and discriminant validity in mind, interpret the validity coefficients; this should include an assessment about whether the correlation coefficients (at least two different pairings) are statistically significantly different from each other.

We need to see if these correlations are statistically significantly different from each other. I am interested in knowing if the correlations between Overall Instructor and each of the three course dimensions (Valued [ $r = 0.63, p < 0.01$ ], TradPed [ $r = 0.80, p < 0.01$ ], SRPed [ $r = 0.67, p < 0.01$ ]) are statistically significantly different from each other.

```
cocor::cocor(formula = ~Valued + OvInstructor | TradPed + OvInstructor,
  data = big)
```

Results of a comparison of two overlapping correlations based on dependent groups

Comparison between r.jk (OvInstructor, Valued) = 0.6344 and r.jh (OvInstructor, TradPed) = 0.7760

```
Difference: r.jk - r.jh = -0.1652
Related correlation: r.kh = 0.697
Data: big: j = OvInstructor, k = Valued, h = TradPed
Group size: n = 307
Null hypothesis: r.jk is equal to r.jh
Alternative hypothesis: r.jk is not equal to r.jh (two-sided)
Alpha: 0.05

pearson1898: Pearson and Filon's z (1898)
z = -5.4939, p-value = 0.0000
Null hypothesis rejected

hotelling1940: Hotelling's t (1940)
t = -6.2651, df = 304, p-value = 0.0000
Null hypothesis rejected

williams1959: Williams' t (1959)
t = -6.1447, df = 304, p-value = 0.0000
Null hypothesis rejected

olkin1967: Olkin's z (1967)
z = -5.4939, p-value = 0.0000
Null hypothesis rejected

dunn1969: Dunn and Clark's z (1969)
z = -5.9983, p-value = 0.0000
Null hypothesis rejected

hendrickson1970: Hendrickson, Stanley, and Hills' (1970) modification of Williams' t (1959)
t = -6.2651, df = 304, p-value = 0.0000
Null hypothesis rejected

steiger1980: Steiger's (1980) modification of Dunn and Clark's z (1969) using average correlation
z = -5.9444, p-value = 0.0000
Null hypothesis rejected

meng1992: Meng, Rosenthal, and Rubin's z (1992)
z = -5.9182, p-value = 0.0000
Null hypothesis rejected
95% confidence interval for r.jk - r.jh: -0.4644 -0.2333
Null hypothesis rejected (Interval does not include 0)

hittner2003: Hittner, May, and Silver's (2003) modification of Dunn and Clark's z (1969) using
z = -5.8868, p-value = 0.0000
Null hypothesis rejected

zou2007: Zou's (2007) confidence interval
95% confidence interval for r.jk - r.jh: -0.2282 -0.1089
```

Null hypothesis rejected (Interval does not include 0)

Fisher's z-test ( $z = -5.887$ ,  $p < 0.001$ ) indicates that the correlation of overall instructor with the valued subscale ( $r = 0.63$ ) is lower than its correlation with the traditional pedagogy subscale ( $r = 0.80$ ).

```
cocor::cocor(formula = ~TradPed + OvInstructor | SRPed + OvInstructor,
  data = big)
```

Results of a comparison of two overlapping correlations based on dependent groups

```
Comparison between r.jk (OvInstructor, TradPed) = 0.7962 and r.jh (OvInstructor, SRPed) = 0.679
Difference: r.jk - r.jh = 0.1211
Related correlation: r.kh = 0.7091
Data: big: j = OvInstructor, k = TradPed, h = SRPed
Group size: n = 298
Null hypothesis: r.jk is equal to r.jh
Alternative hypothesis: r.jk is not equal to r.jh (two-sided)
Alpha: 0.05
```

```
pearson1898: Pearson and Filon's z (1898)
z = 4.2785, p-value = 0.0000
Null hypothesis rejected
```

```
hotelling1940: Hotelling's t (1940)
t = 4.6684, df = 295, p-value = 0.0000
Null hypothesis rejected
```

```
williams1959: Williams' t (1959)
t = 4.5800, df = 295, p-value = 0.0000
Null hypothesis rejected
```

```
olkin1967: Olkin's z (1967)
z = 4.2785, p-value = 0.0000
Null hypothesis rejected
```

```
dunn1969: Dunn and Clark's z (1969)
z = 4.5174, p-value = 0.0000
Null hypothesis rejected
```

```
hendrickson1970: Hendrickson, Stanley, and Hills' (1970) modification of Williams' t (1959)
t = 4.6684, df = 295, p-value = 0.0000
Null hypothesis rejected
```

```
steiger1980: Steiger's (1980) modification of Dunn and Clark's z (1969) using average correlation
z = 4.4945, p-value = 0.0000
```

Null hypothesis rejected

```
meng1992: Meng, Rosenthal, and Rubin's z (1992)
z = 4.4834, p-value = 0.0000
Null hypothesis rejected
95% confidence interval for r.jk - r.jh: 0.1510 0.3855
Null hypothesis rejected (Interval does not include 0)
```

```
hittner2003: Hittner, May, and Silver's (2003) modification of Dunn and Clark's z (1969) using
z = 4.4678, p-value = 0.0000
Null hypothesis rejected
```

```
zou2007: Zou's (2007) confidence interval
95% confidence interval for r.jk - r.jh: 0.0676 0.1802
Null hypothesis rejected (Interval does not include 0)
```

Fisher's z-test ( $z = 4.4678$ ,  $p < 0.001$ ) indicates that the correlation of overall instructor with the traditional pedagogy subscale ( $r = 0.80$ ) is higher than its correlation with the socially responsive pedagogy subscale ( $r = 0.67$ ).

```
cocor::cocor(formula = ~Valued + OvInstructor | SRPed + OvInstructor, data = big)
```

Results of a comparison of two overlapping correlations based on dependent groups

```
Comparison between r.jk (OvInstructor, Valued) = 0.6338 and r.jh (OvInstructor, SRPed) = 0.6717
Difference: r.jk - r.jh = -0.0379
Related correlation: r.kh = 0.5624
Data: big: j = OvInstructor, k = Valued, h = SRPed
Group size: n = 299
Null hypothesis: r.jk is equal to r.jh
Alternative hypothesis: r.jk is not equal to r.jh (two-sided)
Alpha: 0.05
```

```
pearson1898: Pearson and Filon's z (1898)
z = -1.0091, p-value = 0.3129
Null hypothesis retained
```

```
hotelling1940: Hotelling's t (1940)
t = -1.0355, df = 296, p-value = 0.3013
Null hypothesis retained
```

```
williams1959: Williams' t (1959)
t = -1.0071, df = 296, p-value = 0.3147
Null hypothesis retained
```

```
olkin1967: Olkin's z (1967)
```

$z = -1.0091$ , p-value = 0.3129  
 Null hypothesis retained

dunn1969: Dunn and Clark's z (1969)  
 $z = -1.0062$ , p-value = 0.3143  
 Null hypothesis retained

hendrickson1970: Hendrickson, Stanley, and Hills' (1970) modification of Williams' t (1959)  
 $t = -1.0355$ , df = 296, p-value = 0.3013  
 Null hypothesis retained

steiger1980: Steiger's (1980) modification of Dunn and Clark's z (1969) using average correlation  
 $z = -1.0060$ , p-value = 0.3144  
 Null hypothesis retained

meng1992: Meng, Rosenthal, and Rubin's z (1992)  
 $z = -1.0058$ , p-value = 0.3145  
 Null hypothesis retained  
 95% confidence interval for r.jk - r.jh: -0.1948 0.0627  
 Null hypothesis retained (Interval includes 0)

hittner2003: Hittner, May, and Silver's (2003) modification of Dunn and Clark's z (1969) using average correlation  
 $z = -1.0058$ , p-value = 0.3145  
 Null hypothesis retained

zou2007: Zou's (2007) confidence interval  
 95% confidence interval for r.jk - r.jh: -0.1129 0.0360  
 Null hypothesis retained (Interval includes 0)

Fisher's z-test ( $z = -1.006$ ,  $p = 0.315$ ) indicates that the correlation of overall instructor with the valued subscale ( $r = 0.4$ ) is not statistically significantly different than its correlation with the socially responsive pedagogy subscale ( $r = 0.67$ ).

#### 4.7.4 With at least three variables, evaluate the degree to which the instrument demonstrates incremental validity (this should involve two regression equations and their statistical comparison)

Playing around with these variables, let's presume our outcome of interest is the student's *valuation of the class* (i.e., the "Valued by the Student variable") and we usually predict it through traditional pedagogy. What does SRPed contribute over-and-above?

*Please understand, that we would normally have a more robust dataset with other indicators – maybe predicting students' grades?*

*Also, we are completely ignoring the multi-level nature of this data. The published manuscript takes a multi-level approach to analyzing the data and my lessons on multi-level modeling address this as well.*

```
big <- na.omit(big) #included b/c there was uneven missingness and the subsequent comparison
Step1 <- lm(Valued ~ TradPed, data = big)
Step2 <- lm(Valued ~ TradPed + SRPed, data = big)
summary(Step1)
```

Call:

```
lm(formula = Valued ~ TradPed, data = big)
```

Residuals:

Min	1Q	Median	3Q	Max
-1.43330	-0.25471	0.04673	0.25388	1.79522

Coefficients:

	Estimate	Std. Error	t value	Pr(> t )							
(Intercept)	1.67482	0.18581	9.014 <0.0000000000000002	***							
TradPed	0.61426	0.04191	14.656 <0.0000000000000002	***							
---											
Signif. codes:	0	'***'	0.001	'**'	0.01	'*'	0.05	'. '	0.1	' '	1

Residual standard error: 0.4274 on 213 degrees of freedom

Multiple R-squared: 0.5021, Adjusted R-squared: 0.4998

F-statistic: 214.8 on 1 and 213 DF, p-value: < 0.0000000000000002

```
summary(Step2)
```

Call:

```
lm(formula = Valued ~ TradPed + SRPed, data = big)
```

Residuals:

Min	1Q	Median	3Q	Max
-1.39671	-0.22675	0.03228	0.24841	1.71917

Coefficients:

	Estimate	Std. Error	t value	Pr(> t )							
(Intercept)	1.44912	0.26349	5.500	0.000000109 ***							
TradPed	0.56933	0.05602	10.162 < 0.0000000000000002	***							
SRPed	0.09116	0.07554	1.207	0.229							
---											
Signif. codes:	0	'***'	0.001	'**'	0.01	'*'	0.05	'. '	0.1	' '	1

Residual standard error: 0.427 on 212 degrees of freedom

Multiple R-squared: 0.5055, Adjusted R-squared: 0.5009

F-statistic: 108.4 on 2 and 212 DF, p-value: < 0.0000000000000002

In the first step we see that traditional pedagogy had a statistically significant effect on the valued dimension  $B = 0.614, p < 0.001$ . This model accounted for 50% of variance.

In the second step, socially responsive pedagogy was not a statistically significant predictor, over and above traditional pedagogy  $B = 0.091, p = 0.228$ . This model accounted for 51% of variance.

We can formally compare these two models with an the *anova()* function in base R.

```
anova(Step1, Step2)
```

#### Analysis of Variance Table

```
Model 1: Valued ~ TradPed
Model 2: Valued ~ TradPed + SRPed
  Res.Df   RSS Df Sum of Sq    F Pr(>F)
1     213 38.918
2     212 38.652  1   0.26554 1.4564 0.2288
```

We see that socially responsive pedagogy adds only a non-significant proportion of variance over traditional pedagogy ( $F[1, 212] = 38.652, p = 0.229$ ).

# Chapter 5

## Reliability

### [Screencasted Lecture Link](#)

The focus of this lecture is the assessment of reliability. We start by defining *classical test theory* and examining several forms of reliability. While the majority of our time is spent considering estimates of internal consistency, we also review retest reliability and inter-rater reliability.

### 5.1 Navigating this Lesson

There is one hour and twenty minutes of lecture. If you work through the materials with me, plan for an additional hour.

While the majority of R objects and data you will need are created within the R script that sources the chapter, occasionally there are some that cannot be created from within the R framework. Additionally, sometimes links fail. All original materials are provided at the [GitHub site](#) that hosts the book. More detailed guidelines for ways to access all these materials are provided in the OER's [introduction](#)

#### 5.1.1 Learning Objectives

Focusing on this week's materials, make sure you can:

- Define “reliability.”
- Identify broad classes of reliability.
- Interpret reliability coefficients.
- Describe the strengths and limitations of the alpha coefficient.

#### 5.1.2 Planning for Practice

In each of these lessons I provide suggestions for practice that allow you to select one or more problems that are graded in difficulty. The least complex is to change the random seed in the research and rework the problem demonstrated in the lesson. The results *should* map onto the ones obtained in the lecture.

The second option involves utilizing one of the simulated datasets available in this OER. The **last lesson** in the OER contains three simulations that could be used for all of the statistics-based practice suggestions. Especially if you started with one of these examples in an earlier lesson, I highly recommend you continue with that.

Alternatively, Lewis and Neville's [2015] Gendered Racial Microaggressions Scale for Black Women will be used in the lessons on exploratory factor analysis; Keum et al.'s Gendered Racial Microaggressions Scale for Asian American Women [Keum et al., 2018] will be used in the lessons on confirmatory factor analysis; and Conover et al.'s [2017] Ableist Microaggressions Scale is used in the lesson on invariance testing. Any of these would be suitable for the PCA and PAF homework assignments.

As a third option, you are welcome to use data to which you have access and is suitable for PCA. In any case, please select a scale that has item-level data for which there is a theorized total scale score as well as two or more subscales (three subscales is ideal). With each of these options, plan to:

- Format (i.e., rescore, if necessary) a dataset so that it is possible to calculate estimates of internal consistency
- Calculate and report the alpha coefficient for the total scale scores and subscales (if the scale has them)
- Calculate and report  $\omega_t$  and  $\omega_h$ . With these two determine what proportion of the variance is due to all the factors, error, and  $g$ .
- Calculate total and subscale scores.
- Describe other reliability estimates that would be appropriate for the measure you are evaluating.

### 5.1.3 Readings & Resources

In preparing this chapter, I drew heavily from the following resource(s). Other resources are cited (and linked, when possible) in the text with complete citations in the reference list.

- Jhangiani, R. S., Chiang, I.-C. A., Cuttler, C., & Leighton, D. C. (2019). Reliability and Validity. In *Research Methods in Psychology*. <https://doi.org/10.17605/OSF.IO/HF7DQ>
- Revelle, W., & Condon, D. M. (2019a). Reliability from to : A tutorial. Psychological Assessment. <https://doi.org/10.1037/pas0000754>
  - A full-text preprint is available [here](#).
- Revelle, W., & Condon, D. M. (2019b). Reliability from to : A tutorial. Online supplement. Psychological Assessment. <https://doi.org/10.1037/pas0000754>
- Revelle, William. (n.d.). Reliability. In An introduction to psychometric theory with applications in R. Retrieved from <http://www.personality-project.org/dev/r/book/#chapter7>
  - All three documents provide a practical integration of conceptual and mechanical.
- Szymanski, D. M., & Bissonette, D. (2020). Perceptions of the LGBTQ College Campus Climate Scale: Development and psychometric evaluation. *Journal of Homosexuality*, 67(10), 1412–1428. <https://doi.org/10.1080/00918369.2019.1591788>
  - The research vignette for this lesson.

### 5.1.4 Packages

The packages used in this lesson are embedded in this code. When the hashtags are removed, the script below will (a) check to see if the following packages are installed on your computer and, if not (b) install them.

```
# will install the package if not already installed
# if(!require(psych)){install.packages('psych')}
# if(!require(tidyverse)){install.packages('tidyverse')}
# if(!require(MASS)){install.packages('MASS')}
# if(!require(sjstats)){install.packages('sjstats')}
# if(!require(apaTables)){install.packages('apaTables')}
# if(!require(qualtrics)){install.packages('qualtrics')}
```

## 5.2 Defining Reliability

### 5.2.1 Begins with Classical Test Theory (CTT)

CTT is based on Spearman's (1904) *true-score model* where:

- an observed score consists of two components – a true component and an error component
- $X = T + E$ 
  - $X$  = the fallible, observed/manifest score, obtained under ideal or perfect conditions of measurement (these conditions never exist);
  - $T$  = the true/latent score (that will likely remain unknown); and
  - $E$  = random error
- In CTT, we assume that the traits measured are constant and the errors random.
  - Therefore, the mean of measurement errors for any individual (upon numerous repeated testing) would be zero.
- That said, in CTT, the true score would be equal to the mean of the observed scores over an indefinite number of repeated measures.
  - Caveat: this is based on the assumption that when individuals are repeatedly measured, their true scores remain unchanged.
- In classic test theory, true score can be estimated over multiple trials. However, if errors are systematically biased, the true score will remain unknown.

### 5.2.2 Why are we concerned with reliability? Error!

Measurements are imperfect and every observation has some unknown amount of error associated with it. There are two components in error:

- **random/unsystematic:** varies in unpredictable and inconsistent ways upon repeated measurements; sources are unknown

- **systematic:** recurs upon repeated measurements reflecting situational or individual effects that, theoretically, could be specified.

Correlations are attenuated (i.e., smaller than) from the true correlation if the observations contain error. Knowing the reliability of an instruments allows us to:

- estimate the degree to which measured at one time and place with one instrument predict scores at another time and/or place and perhaps measured with a different instrument
- estimate the consistency of scores
- estimate “...the degree to which test scores are free from errors of measurement” (APA, 1985, p. 19)

Figure 7.1a in [Revelle’s chapter](#) illustrates the *attenuation* of the correlation between the variables  $p$  and  $q$  as a function of reliability.

- circles (latent variables [LV]) represent the *true score*
- observed/measured/manifest variables are represented by squares, and each has an associated error; not illustrated are the *random* and *systematic* components of error
- a true score is composed of a measured variable and its error
- the relationship between the true scores would be stronger than the one between the measured variables
- moving to 7.1b in Revelle’s chapter, the correlation between LV  $p$  and the observed ‘’ can be estimated from the correlation of  $p'$  with a parallel test (this is the reliability piece)

Figure 7.2 in Revelle’s Chapter 7 [] illustrates the conceptual effect of reliability on the estimation of a true score. The figure is meant to demonstrate that when error variances are small and reliability is greater, the variance of the true scores more closely approximates that of observed scores.

### 5.2.3 The Reliability Coefficient

The symbol for reliability,  $r_{xx}$ , sums up the big-picture definition that reliability is the correlation of a measure with itself. There are a number of ways to think about it:

- a “theoretical validity” of a measure because it refers to a relationship between observed scores and scores on a latent variable or construct,
- represents the fraction of an observed score variance that is not error,
- ranges from 0-1
  - 1, when all observed variance is due to true-score variance; there are no random errors,
  - 0, when all observed variance is due to random errors of measurement,
- represents the squared correlation between observed scores and true scores,
- the ratio between true-score variance and observed-score variance (for a formulaic rendition see [[Pedhazur and Schmelkin, 1991](#)]),

$$r_{xt}^2 = r_{xx} = \frac{\sigma_2^t}{\sigma_2^x}$$

where  $r_{xt}^2$  is the proportion of variance between observed scores ( $t + e$ ) and true scores ( $t$ ); its square root is the correlation

$r_{xx}$  is the reliability of a measure

$\sigma_2^t$  is the variance of true scores

$\sigma_2^x$  is the variance of observed scores

- The reliability coefficient is interpreted as the proportion of systematic variance in the observed score.
  - .8 means that 80% of the variance of the observed scores is systematic;
  - .2 (e.g., 1.00 - .8) is the proportion of variance due to random errors;
  - the reliability coefficient is population specific.

To restate the first portion of the formula: although reliability is expressed as a correlation between observed scores, it is also the ratio of reliable variance to total variance.

### 5.3 Research Vignette

The research vignette for this lesson is the development and psychometric evaluation of the Perceptions of the LGBTQ College Campus Climate Scale [Szymanski and Bissonette, 2020]. The scale is six items with responses rated on a 7-point Likert scale ranging from 1 (*strongly disagree*) to 7 (*strongly agree*). Higher scores indicate more negative perceptions of the LGBTQ campus climate. Szymanski and Bissonette [2020] have suggested that the psychometric evaluation supports using the scale in its entirety or as subscales. Each item is listed below with its variable name in parentheses:

- College response to LGBTQ students:
  - My university/college is cold and uncaring toward LGBTQ students. (cold)
  - My university/college is unresponsive to the needs of LGBTQ students. (unresponsive)
  - My university/college provides a supportive environment for LGBTQ students. (unsupportive)
    - \* this item must be reverse-scored
- LGBTQ Stigma:
  - Negative attitudes toward LGBTQ persons are openly expressed on my university/college campus. (negative)
  - Heterosexism, homophobia, biphobia, transphobia, and cissexism are visible on my university/college campus. (heterosexism)
  - LGBTQ students are harassed on my university/college campus. (harassed)

A preprint of the article is available at ResearchGate. Below is the script for simulating item-level data from the factor loadings, means, and sample size presented in the published article.

Because data is collected at the item level (and I want this resource to be as practical as possible, I have simulated the data for each of the scales at the item level).

Simulating the data involved using factor loadings, means, and correlations between the scales. Because the simulation will produce “out-of-bounds” values, the code below rescales the scores into the range of the Likert-type scaling and rounds them to whole values.

Five additional scales were reported in the Szymanski and Bissonette article [2020]. Unfortunately, I could not locate factor loadings for all of them; and in two cases, I used estimates from a more recent psychometric analysis. When the individual item and their factor loadings are known, I assigned names based on item content (e.g., “lo\_energy”) rather than using item numbers (e.g., “PHQ4”). When I am doing psychometric analyses, I prefer item-level names so that I can quickly see (without having to look up the item names) how the items are behaving. While the focus of this series of chapters is on the LGBTQ Campus Climate scale, this simulated data might be useful to you in one or more of the suggestions for practice (e.g., examining the psychometric characteristics of one or the other scales). The scales, their original citation, and information about how I simulated data for each are listed below.

- **Sexual Orientation-Based Campus Victimization Scale** [Herek, 1993] is a 9-item item scale with Likert scaling ranging from 0 (*never*) to 3 (*two or more times*). Because I was not able to locate factor loadings from a psychometric evaluation, I simulated the data by specifying a 0.8 as a standardized factor loading for each of the items.
- **College Satisfaction Scale** [Helm et al., 1998] is a 5-item scale with Likert scaling ranging from 1 (*strongly disagree*) to 7 (*strongly agree*). Higher scores represent greater college satisfaction. Because I was not able to locate factor loadings from a psychometric evaluation, I simulated the data by specifying a 0.8 as a standardized factor loading for each of the items.
- **Institutional and Goals Commitment** [Pascarella and Terenzini, 1980] is a 6-item subscale from a 35-item measure assessing academic/social integration and institutional/goal commitment (5 subscales total). The measure had with Likert scaling ranging from 1 (*strongly disagree*) to 5 (*strongly agree*). Higher scores on the institutional and goals commitment subscale indicate greater intentions to persist in college. Data were simulated using factor loadings in the source article.
- **GAD-7** [Spitzer et al., 2006] is a 7-item scale with Likert scaling ranging from 0 (*not at all*) to 3 (*nearly every day*). Higher scores indicate more anxiety. I simulated data by estimating factor loadings from Brattmyr et al. [2022].
- **PHQ-9** [Kroenke et al., 2001] is a 9-item scale with Likert scaling ranging from 0 (*not at all*) to 3 (*nearly every day*). Higher scores indicate higher levels of depression. I simulated data by estimating factor loadings from Brattmyr et al. [2022].

```
#Entering the intercorrelations, means, and standard deviations from the journal article
```

```
Szymanski_generating_model <- '
  #measurement model
  CollegeResponse =~ .88*cold + .73*unresponsive + .73*supportive
  Stigma =~ .86*negative + .76*heterosexism + .71*harassed
  Victimization =~ .8*Vic1 + .8*Vic2 + .8*Vic3 + .8*Vic4 + .8*Vic5 + .8*Vic6 + .8*Vic7 + .8*Vic8 + .8*Vic9 + .8*Vic10 + .8*Vic11 + .8*Vic12 + .8*Vic13 + .8*Vic14 + .8*Vic15 + .8*Vic16 + .8*Vic17 + .8*Vic18 + .8*Vic19 + .8*Vic20 + .8*Vic21 + .8*Vic22 + .8*Vic23 + .8*Vic24 + .8*Vic25 + .8*Vic26 + .8*Vic27 + .8*Vic28 + .8*Vic29 + .8*Vic30 + .8*Vic31 + .8*Vic32 + .8*Vic33 + .8*Vic34 + .8*Vic35 + .8*Vic36 + .8*Vic37 + .8*Vic38 + .8*Vic39 + .8*Vic40 + .8*Vic41 + .8*Vic42 + .8*Vic43 + .8*Vic44 + .8*Vic45 + .8*Vic46 + .8*Vic47 + .8*Vic48 + .8*Vic49 + .8*Vic50 + .8*Vic51 + .8*Vic52 + .8*Vic53 + .8*Vic54 + .8*Vic55 + .8*Vic56 + .8*Vic57 + .8*Vic58 + .8*Vic59 + .8*Vic60 + .8*Vic61 + .8*Vic62 + .8*Vic63 + .8*Vic64 + .8*Vic65 + .8*Vic66 + .8*Vic67 + .8*Vic68 + .8*Vic69 + .8*Vic70 + .8*Vic71 + .8*Vic72 + .8*Vic73 + .8*Vic74 + .8*Vic75 + .8*Vic76 + .8*Vic77 + .8*Vic78 + .8*Vic79 + .8*Vic80 + .8*Vic81 + .8*Vic82 + .8*Vic83 + .8*Vic84 + .8*Vic85 + .8*Vic86 + .8*Vic87 + .8*Vic88 + .8*Vic89 + .8*Vic90 + .8*Vic91 + .8*Vic92 + .8*Vic93 + .8*Vic94 + .8*Vic95 + .8*Vic96 + .8*Vic97 + .8*Vic98 + .8*Vic99 + .8*Vic100 + .8*Vic101 + .8*Vic102 + .8*Vic103 + .8*Vic104 + .8*Vic105 + .8*Vic106 + .8*Vic107 + .8*Vic108 + .8*Vic109 + .8*Vic110 + .8*Vic111 + .8*Vic112 + .8*Vic113 + .8*Vic114 + .8*Vic115 + .8*Vic116 + .8*Vic117 + .8*Vic118 + .8*Vic119 + .8*Vic120 + .8*Vic121 + .8*Vic122 + .8*Vic123 + .8*Vic124 + .8*Vic125 + .8*Vic126 + .8*Vic127 + .8*Vic128 + .8*Vic129 + .8*Vic130 + .8*Vic131 + .8*Vic132 + .8*Vic133 + .8*Vic134 + .8*Vic135 + .8*Vic136 + .8*Vic137 + .8*Vic138 + .8*Vic139 + .8*Vic140 + .8*Vic141 + .8*Vic142 + .8*Vic143 + .8*Vic144 + .8*Vic145 + .8*Vic146 + .8*Vic147 + .8*Vic148 + .8*Vic149 + .8*Vic150 + .8*Vic151 + .8*Vic152 + .8*Vic153 + .8*Vic154 + .8*Vic155 + .8*Vic156 + .8*Vic157 + .8*Vic158 + .8*Vic159 + .8*Vic160 + .8*Vic161 + .8*Vic162 + .8*Vic163 + .8*Vic164 + .8*Vic165 + .8*Vic166 + .8*Vic167 + .8*Vic168 + .8*Vic169 + .8*Vic170 + .8*Vic171 + .8*Vic172 + .8*Vic173 + .8*Vic174 + .8*Vic175 + .8*Vic176 + .8*Vic177 + .8*Vic178 + .8*Vic179 + .8*Vic180 + .8*Vic181 + .8*Vic182 + .8*Vic183 + .8*Vic184 + .8*Vic185 + .8*Vic186 + .8*Vic187 + .8*Vic188 + .8*Vic189 + .8*Vic190 + .8*Vic191 + .8*Vic192 + .8*Vic193 + .8*Vic194 + .8*Vic195 + .8*Vic196 + .8*Vic197 + .8*Vic198 + .8*Vic199 + .8*Vic200 + .8*Vic201 + .8*Vic202 + .8*Vic203 + .8*Vic204 + .8*Vic205 + .8*Vic206 + .8*Vic207 + .8*Vic208 + .8*Vic209 + .8*Vic210 + .8*Vic211 + .8*Vic212 + .8*Vic213 + .8*Vic214 + .8*Vic215 + .8*Vic216 + .8*Vic217 + .8*Vic218 + .8*Vic219 + .8*Vic220 + .8*Vic221 + .8*Vic222 + .8*Vic223 + .8*Vic224 + .8*Vic225 + .8*Vic226 + .8*Vic227 + .8*Vic228 + .8*Vic229 + .8*Vic230 + .8*Vic231 + .8*Vic232 + .8*Vic233 + .8*Vic234 + .8*Vic235 + .8*Vic236 + .8*Vic237 + .8*Vic238 + .8*Vic239 + .8*Vic240 + .8*Vic241 + .8*Vic242 + .8*Vic243 + .8*Vic244 + .8*Vic245 + .8*Vic246 + .8*Vic247 + .8*Vic248 + .8*Vic249 + .8*Vic250 + .8*Vic251 + .8*Vic252 + .8*Vic253 + .8*Vic254 + .8*Vic255 + .8*Vic256 + .8*Vic257 + .8*Vic258 + .8*Vic259 + .8*Vic260 + .8*Vic261 + .8*Vic262 + .8*Vic263 + .8*Vic264 + .8*Vic265 + .8*Vic266 + .8*Vic267 + .8*Vic268 + .8*Vic269 + .8*Vic270 + .8*Vic271 + .8*Vic272 + .8*Vic273 + .8*Vic274 + .8*Vic275 + .8*Vic276 + .8*Vic277 + .8*Vic278 + .8*Vic279 + .8*Vic280 + .8*Vic281 + .8*Vic282 + .8*Vic283 + .8*Vic284 + .8*Vic285 + .8*Vic286 + .8*Vic287 + .8*Vic288 + .8*Vic289 + .8*Vic290 + .8*Vic291 + .8*Vic292 + .8*Vic293 + .8*Vic294 + .8*Vic295 + .8*Vic296 + .8*Vic297 + .8*Vic298 + .8*Vic299 + .8*Vic300 + .8*Vic301 + .8*Vic302 + .8*Vic303 + .8*Vic304 + .8*Vic305 + .8*Vic306 + .8*Vic307 + .8*Vic308 + .8*Vic309 + .8*Vic310 + .8*Vic311 + .8*Vic312 + .8*Vic313 + .8*Vic314 + .8*Vic315 + .8*Vic316 + .8*Vic317 + .8*Vic318 + .8*Vic319 + .8*Vic320 + .8*Vic321 + .8*Vic322 + .8*Vic323 + .8*Vic324 + .8*Vic325 + .8*Vic326 + .8*Vic327 + .8*Vic328 + .8*Vic329 + .8*Vic330 + .8*Vic331 + .8*Vic332 + .8*Vic333 + .8*Vic334 + .8*Vic335 + .8*Vic336 + .8*Vic337 + .8*Vic338 + .8*Vic339 + .8*Vic340 + .8*Vic341 + .8*Vic342 + .8*Vic343 + .8*Vic344 + .8*Vic345 + .8*Vic346 + .8*Vic347 + .8*Vic348 + .8*Vic349 + .8*Vic350 + .8*Vic351 + .8*Vic352 + .8*Vic353 + .8*Vic354 + .8*Vic355 + .8*Vic356 + .8*Vic357 + .8*Vic358 + .8*Vic359 + .8*Vic360 + .8*Vic361 + .8*Vic362 + .8*Vic363 + .8*Vic364 + .8*Vic365 + .8*Vic366 + .8*Vic367 + .8*Vic368 + .8*Vic369 + .8*Vic370 + .8*Vic371 + .8*Vic372 + .8*Vic373 + .8*Vic374 + .8*Vic375 + .8*Vic376 + .8*Vic377 + .8*Vic378 + .8*Vic379 + .8*Vic380 + .8*Vic381 + .8*Vic382 + .8*Vic383 + .8*Vic384 + .8*Vic385 + .8*Vic386 + .8*Vic387 + .8*Vic388 + .8*Vic389 + .8*Vic390 + .8*Vic391 + .8*Vic392 + .8*Vic393 + .8*Vic394 + .8*Vic395 + .8*Vic396 + .8*Vic397 + .8*Vic398 + .8*Vic399 + .8*Vic400 + .8*Vic401 + .8*Vic402 + .8*Vic403 + .8*Vic404 + .8*Vic405 + .8*Vic406 + .8*Vic407 + .8*Vic408 + .8*Vic409 + .8*Vic410 + .8*Vic411 + .8*Vic412 + .8*Vic413 + .8*Vic414 + .8*Vic415 + .8*Vic416 + .8*Vic417 + .8*Vic418 + .8*Vic419 + .8*Vic420 + .8*Vic421 + .8*Vic422 + .8*Vic423 + .8*Vic424 + .8*Vic425 + .8*Vic426 + .8*Vic427 + .8*Vic428 + .8*Vic429 + .8*Vic430 + .8*Vic431 + .8*Vic432 + .8*Vic433 + .8*Vic434 + .8*Vic435 + .8*Vic436 + .8*Vic437 + .8*Vic438 + .8*Vic439 + .8*Vic440 + .8*Vic441 + .8*Vic442 + .8*Vic443 + .8*Vic444 + .8*Vic445 + .8*Vic446 + .8*Vic447 + .8*Vic448 + .8*Vic449 + .8*Vic450 + .8*Vic451 + .8*Vic452 + .8*Vic453 + .8*Vic454 + .8*Vic455 + .8*Vic456 + .8*Vic457 + .8*Vic458 + .8*Vic459 + .8*Vic460 + .8*Vic461 + .8*Vic462 + .8*Vic463 + .8*Vic464 + .8*Vic465 + .8*Vic466 + .8*Vic467 + .8*Vic468 + .8*Vic469 + .8*Vic470 + .8*Vic471 + .8*Vic472 + .8*Vic473 + .8*Vic474 + .8*Vic475 + .8*Vic476 + .8*Vic477 + .8*Vic478 + .8*Vic479 + .8*Vic480 + .8*Vic481 + .8*Vic482 + .8*Vic483 + .8*Vic484 + .8*Vic485 + .8*Vic486 + .8*Vic487 + .8*Vic488 + .8*Vic489 + .8*Vic490 + .8*Vic491 + .8*Vic492 + .8*Vic493 + .8*Vic494 + .8*Vic495 + .8*Vic496 + .8*Vic497 + .8*Vic498 + .8*Vic499 + .8*Vic500 + .8*Vic501 + .8*Vic502 + .8*Vic503 + .8*Vic504 + .8*Vic505 + .8*Vic506 + .8*Vic507 + .8*Vic508 + .8*Vic509 + .8*Vic510 + .8*Vic511 + .8*Vic512 + .8*Vic513 + .8*Vic514 + .8*Vic515 + .8*Vic516 + .8*Vic517 + .8*Vic518 + .8*Vic519 + .8*Vic520 + .8*Vic521 + .8*Vic522 + .8*Vic523 + .8*Vic524 + .8*Vic525 + .8*Vic526 + .8*Vic527 + .8*Vic528 + .8*Vic529 + .8*Vic530 + .8*Vic531 + .8*Vic532 + .8*Vic533 + .8*Vic534 + .8*Vic535 + .8*Vic536 + .8*Vic537 + .8*Vic538 + .8*Vic539 + .8*Vic540 + .8*Vic541 + .8*Vic542 + .8*Vic543 + .8*Vic544 + .8*Vic545 + .8*Vic546 + .8*Vic547 + .8*Vic548 + .8*Vic549 + .8*Vic550 + .8*Vic551 + .8*Vic552 + .8*Vic553 + .8*Vic554 + .8*Vic555 + .8*Vic556 + .8*Vic557 + .8*Vic558 + .8*Vic559 + .8*Vic560 + .8*Vic561 + .8*Vic562 + .8*Vic563 + .8*Vic564 + .8*Vic565 + .8*Vic566 + .8*Vic567 + .8*Vic568 + .8*Vic569 + .8*Vic570 + .8*Vic571 + .8*Vic572 + .8*Vic573 + .8*Vic574 + .8*Vic575 + .8*Vic576 + .8*Vic577 + .8*Vic578 + .8*Vic579 + .8*Vic580 + .8*Vic581 + .8*Vic582 + .8*Vic583 + .8*Vic584 + .8*Vic585 + .8*Vic586 + .8*Vic587 + .8*Vic588 + .8*Vic589 + .8*Vic590 + .8*Vic591 + .8*Vic592 + .8*Vic593 + .8*Vic594 + .8*Vic595 + .8*Vic596 + .8*Vic597 + .8*Vic598 + .8*Vic599 + .8*Vic600 + .8*Vic601 + .8*Vic602 + .8*Vic603 + .8*Vic604 + .8*Vic605 + .8*Vic606 + .8*Vic607 + .8*Vic608 + .8*Vic609 + .8*Vic610 + .8*Vic611 + .8*Vic612 + .8*Vic613 + .8*Vic614 + .8*Vic615 + .8*Vic616 + .8*Vic617 + .8*Vic618 + .8*Vic619 + .8*Vic620 + .8*Vic621 + .8*Vic622 + .8*Vic623 + .8*Vic624 + .8*Vic625 + .8*Vic626 + .8*Vic627 + .8*Vic628 + .8*Vic629 + .8*Vic630 + .8*Vic631 + .8*Vic632 + .8*Vic633 + .8*Vic634 + .8*Vic635 + .8*Vic636 + .8*Vic637 + .8*Vic638 + .8*Vic639 + .8*Vic640 + .8*Vic641 + .8*Vic642 + .8*Vic643 + .8*Vic644 + .8*Vic645 + .8*Vic646 + .8*Vic647 + .8*Vic648 + .8*Vic649 + .8*Vic650 + .8*Vic651 + .8*Vic652 + .8*Vic653 + .8*Vic654 + .8*Vic655 + .8*Vic656 + .8*Vic657 + .8*Vic658 + .8*Vic659 + .8*Vic660 + .8*Vic661 + .8*Vic662 + .8*Vic663 + .8*Vic664 + .8*Vic665 + .8*Vic666 + .8*Vic667 + .8*Vic668 + .8*Vic669 + .8*Vic670 + .8*Vic671 + .8*Vic672 + .8*Vic673 + .8*Vic674 + .8*Vic675 + .8*Vic676 + .8*Vic677 + .8*Vic678 + .8*Vic679 + .8*Vic680 + .8*Vic681 + .8*Vic682 + .8*Vic683 + .8*Vic684 + .8*Vic685 + .8*Vic686 + .8*Vic687 + .8*Vic688 + .8*Vic689 + .8*Vic690 + .8*Vic691 + .8*Vic692 + .8*Vic693 + .8*Vic694 + .8*Vic695 + .8*Vic696 + .8*Vic697 + .8*Vic698 + .8*Vic699 + .8*Vic700 + .8*Vic701 + .8*Vic702 + .8*Vic703 + .8*Vic704 + .8*Vic705 + .8*Vic706 + .8*Vic707 + .8*Vic708 + .8*Vic709 + .8*Vic710 + .8*Vic711 + .8*Vic712 + .8*Vic713 + .8*Vic714 + .8*Vic715 + .8*Vic716 + .8*Vic717 + .8*Vic718 + .8*Vic719 + .8*Vic720 + .8*Vic721 + .8*Vic722 + .8*Vic723 + .8*Vic724 + .8*Vic725 + .8*Vic726 + .8*Vic727 + .8*Vic728 + .8*Vic729 + .8*Vic730 + .8*Vic731 + .8*Vic732 + .8*Vic733 + .8*Vic734 + .8*Vic735 + .8*Vic736 + .8*Vic737 + .8*Vic738 + .8*Vic739 + .8*Vic740 + .8*Vic741 + .8*Vic742 + .8*Vic743 + .8*Vic744 + .8*Vic745 + .8*Vic746 + .8*Vic747 + .8*Vic748 + .8*Vic749 + .8*Vic750 + .8*Vic751 + .8*Vic752 + .8*Vic753 + .8*Vic754 + .8*Vic755 + .8*Vic756 + .8*Vic757 + .8*Vic758 + .8*Vic759 + .8*Vic760 + .8*Vic761 + .8*Vic762 + .8*Vic763 + .8*Vic764 + .8*Vic765 + .8*Vic766 + .8*Vic767 + .8*Vic768 + .8*Vic769 + .8*Vic770 + .8*Vic771 + .8*Vic772 + .8*Vic773 + .8*Vic774 + .8*Vic775 + .8*Vic776 + .8*Vic777 + .8*Vic778 + .8*Vic779 + .8*Vic780 + .8*Vic781 + .8*Vic782 + .8*Vic783 + .8*Vic784 + .8*Vic785 + .8*Vic786 + .8*Vic787 + .8*Vic788 + .8*Vic789 + .8*Vic790 + .8*Vic791 + .8*Vic792 + .8*Vic793 + .8*Vic794 + .8*Vic795 + .8*Vic796 + .8*Vic797 + .8*Vic798 + .8*Vic799 + .8*Vic800 + .8*Vic801 + .8*Vic802 + .8*Vic803 + .8*Vic804 + .8*Vic805 + .8*Vic806 + .8*Vic807 + .8*Vic808 + .8*Vic809 + .8*Vic810 + .8*Vic811 + .8*Vic812 + .8*Vic813 + .8*Vic814 + .8*Vic815 + .8*Vic816 + .8*Vic817 + .8*Vic818 + .8*Vic819 + .8*Vic820 + .8*Vic821 + .8*Vic822 + .8*Vic823 + .8*Vic824 + .8*Vic825 + .8*Vic826 + .8*Vic827 + .8*Vic828 + .8*Vic829 + .8*Vic830 + .8*Vic831 + .8*Vic832 + .8*Vic833 + .8*Vic834 + .8*Vic835 + .8*Vic836 + .8*Vic837 + .8*Vic838 + .8*Vic839 + .8*Vic840 + .8*Vic841 + .8*Vic842 + .8*Vic843 + .8*Vic844 + .8*Vic845 + .8*Vic846 + .8*Vic847 + .8*Vic848 + .8*Vic849 + .8*Vic850 + .8*Vic851 + .8*Vic852 + .8*Vic853 + .8*Vic854 + .8*Vic855 + .8*Vic856 + .8*Vic857 + .8*Vic858 + .8*Vic859 + .8*Vic860 + .8*Vic861 + .8*Vic862 + .8*Vic863 + .8*Vic864 + .8*Vic865 + .8*Vic866 + .8*Vic867 + .8*Vic868 + .8*Vic869 + .8*Vic870 + .8*Vic871 + .8*Vic872 + .8*Vic873 + .8*Vic874 + .8*Vic875 + .8*Vic876 + .8*Vic877 + .8*Vic878 + .8*Vic879 + .8*Vic880 + .8*Vic881 + .8*Vic882 + .8*Vic883 + .8*Vic884 + .8*Vic885 + .8*Vic886 + .8*Vic887 + .8*Vic888 + .8*Vic889 + .8*Vic890 + .8*Vic891 + .8*Vic892 + .8*Vic893 + .8*Vic894 + .8*Vic895 + .8*Vic896 + .8*Vic897 + .8*Vic898 + .8*Vic899 + .8*Vic900 + .8*Vic901 + .8*Vic902 + .8*Vic903 + .8*Vic904 + .8*Vic905 + .8*Vic906 + .8*Vic907 + .8*Vic908 + .8*Vic909 + .8*Vic910 + .8*Vic911 + .8*Vic912 + .8*Vic913 + .8*Vic914 + .8*Vic915 + .8*Vic916 + .8*Vic917 + .8*Vic918 + .8*Vic919 + .8*Vic920 + .8*Vic921 + .8*Vic922 + .8*Vic923 + .8*Vic924 + .8*Vic925 + .8*Vic926 + .8*Vic927 + .8*Vic928 + .8*Vic929 + .8*Vic930 + .8*Vic931 + .8*Vic932 + .8*Vic933 + .8*Vic934 + .8*Vic935 + .8*Vic936 + .8*Vic937 + .8*Vic938 + .8*Vic939 + .8*Vic940 + .8*Vic941 + .8*Vic942 + .8*Vic943 + .8*Vic944 + .8*Vic945 + .8*Vic946 + .8*Vic947 + .8*Vic948 + .8*Vic949 + .8*Vic950 + .8*Vic951 + .8*Vic952 + .8*Vic953 + .8*Vic954 + .8*Vic955 + .8*Vic956 + .8*Vic957 + .8*Vic958 + .8*Vic959 + .8*Vic960 + .8*Vic961 + .8*Vic962 + .8*Vic963 + .8*Vic964 + .8*Vic965 + .8*Vic966 + .8*Vic967 + .8*Vic968 + .8*Vic969 + .8*Vic970 + .8*Vic971 + .8*Vic972 + .8*Vic973 + .8*Vic974 + .8*Vic975 + .8*Vic976 + .8*Vic977 + .8*Vic978 + .8*Vic979 + .8*Vic980 + .8*Vic981 + .8*Vic982 + .8*Vic983 + .8*Vic984 + .8*Vic985 + .8*Vic986 + .8*Vic987 + .8*Vic988 + .8*Vic989 + .8*Vic990 + .8*Vic991 + .8*Vic992 + .8*Vic993 + .8*Vic994 + .8*Vic995 + .8*Vic996 + .8*Vic997 + .8*Vic998 + .8*Vic999 + .8*Vic1000 + .8*Vic1001 + .8*Vic1002 + .8*Vic1003 + .8*Vic1004 + .8*Vic1005 + .8*Vic1006 + .8*Vic1007 + .8*Vic1008 + .8*Vic1009 + .8*Vic1010 + .8*Vic1011 + .8*Vic1012 + .8*Vic1013 + .8*Vic1014 + .8*Vic1015 + .8*Vic1016 + .8*Vic1017 + .8*Vic1018 + .8*Vic1019 + .8*Vic1020 + .8*Vic1021 + .8*Vic1022 + .8*Vic1023 + .8*Vic1024 + .8*Vic1025 + .8*Vic1026 + .8*Vic1027 + .8*Vic1028 + .8*Vic1029 + .8*Vic1030 + .8*Vic1031 + .8*Vic1032 + .8*Vic1033 + .8*Vic1034 + .8*Vic1035 + .8*Vic1036 + .8*Vic1037 + .8*Vic1038 + .8*Vic1039 + .8*Vic1040 + .8*Vic1041 + .8*Vic1042 + .8*Vic1043 + .8*Vic1044 + .8*Vic1045 + .8*Vic1046 + .8*Vic1047 + .8*Vic1048 + .8*Vic1049 + .8*Vic1050 + .8*Vic1051 + .8*Vic1052 + .8*Vic1053 + .8*Vic1054 + .8*Vic1055 + .8*Vic1056 + .8*Vic1057 + .8*Vic1058 + .8*Vic1059 + .8*Vic1060 + .8*Vic1061 + .8*Vic1062 + .8*Vic1063 + .8*Vic1064 + .8*Vic1065 + .8*Vic1066 + .8*Vic1067 + .8*Vic1068 + .8*Vic1069 + .8*Vic1070 + .8*Vic1071 + .8*Vic1072 + .8*Vic1073 + .8*Vic1074 + .8*Vic1075 + .8*Vic1076 + .8*Vic1077 + .8*Vic1078 + .8*Vic1079 + .8*Vic1080 + .8*Vic1081 + .8*Vic1082 + .8*Vic1083 + .8*Vic1084 + .8*Vic1085 + .8*Vic1086 + .8*Vic1087 + .8*Vic1088 + .8*Vic1089 + .8*Vic1090 + .8*Vic1091 + .8*Vic1092 + .8*Vic1093 + .8*Vic1094 + .8*Vic1095 + .8*Vic1096 + .8*Vic1097 + .8*Vic1098 + .8*Vic1099 + .8*Vic1100 + .8*Vic1101 + .8*Vic1102 + .8*Vic1103 + .8*Vic1104 + .8*Vic1105 + .8*Vic1106 + .8*Vic1107 + .8*Vic1108 + .8*Vic1109 + .8*Vic1110 + .8*Vic1111 + .8*Vic1112 + .8*Vic1113 + .8*Vic1114 + .8*Vic1115 + .8*Vic1116 + .8*Vic1117 + .8*Vic1118 + .8*Vic1119 + .8*Vic1120 + .8*Vic1121 + .8*Vic1122 + .8*Vic1123 + .8*Vic1124 + .8*Vic1125 + .8*Vic1126 + .8*Vic1127 + .
```

```

CollSat =~ .8*Sat1 + .8*Sat2 + .8*Sat3 + .8*Sat4 + .8*Sat5
Persistence =~ .69*graduation_importance + .63*right_decision + .62*will_register + .55*will_study + .55*will_research + .55*will_teach
Anxiety =~ .851*nervous + .887*worry_control + .894*much_worry + .674*cant_relax + .484*restless
Depression =~ .798*anhedonia + .425*down + .591*sleep + .913*lo_energy + .441*appetite
Harm =~ .798*anhedonia + .425*down + .591*sleep + .913*lo_energy + .441*appetite

#Means
CollegeResponse ~ 2.71*1
Stigma ~3.61*1
Victimization ~ 0.11*1
CollSat ~ 5.61*1
Persistence ~ 4.41*1
Anxiety ~ 1.45*1
Depression ~1.29*1

#Correlations
CollegeResponse ~~ .58*Stigma
CollegeResponse ~~ -.25*Victimization
CollegeResponse ~~ -.59*CollSat
CollegeResponse ~~ -.29*Persistence
CollegeResponse ~~ .17*Anxiety
CollegeResponse ~~ .18*Depression

Stigma ~~ .37*Victimization
Stigma ~~ -.41*CollSat
Stigma ~~ -.19*Persistence
Stigma ~~ .27*Anxiety
Stigma ~~ .24*Depression

Victimization ~~ -.22*CollSat
Victimization ~~ -.04*Persistence
Victimization ~~ .23*Anxiety
Victimization ~~ .21*Depression

CollSat ~~ .53*Persistence
CollSat ~~ -.29*Anxiety
CollSat ~~ -.32*Depression

Persistence ~~ -.22*Anxiety
Persistence ~~ -.26*Depression

Anxiety ~~ .76*Depression
'

set.seed(240218)
dfSzy <- lavaan::simulateData(model = Szymanski_generating_model,
                               model.type = "sem",

```

```

        meanstructure = T,
        sample.nobs=646,
        standardized=FALSE)

#used to retrieve column indices used in the rescaling script below
col_index <- as.data.frame(colnames(dfSzy))

#The code below loops through each column of the dataframe and assigns the scaling accordingly
#Rows 1 thru 6 are the Perceptions of LGBTQ Campus Climate Scale
#Rows 7 thru 15 are the Sexual Orientation-Based Campus Victimization Scale
#Rows 16 thru 20 are the College Satisfaction Scale
#Rows 21 thru 26 are the Institutional and Goals Commitment Scale
#Rows 27 thru 33 are the GAD7
#Rows 34 thru 42 are the PHQ9

for(i in 1:ncol(dfSzy)){
  if(i >= 1 & i <= 6){
    dfSzy[,i] <- scales::rescale(dfSzy[,i], c(1, 7))
  }
  if(i >= 7 & i <= 15){
    dfSzy[,i] <- scales::rescale(dfSzy[,i], c(0, 3))
  }
  if(i >= 16 & i <= 20){
    dfSzy[,i] <- scales::rescale(dfSzy[,i], c(1, 7))
  }
  if(i >= 21 & i <= 26){
    dfSzy[,i] <- scales::rescale(dfSzy[,i], c(1, 5))
  }
  if(i >= 27 & i <= 33){
    dfSzy[,i] <- scales::rescale(dfSzy[,i], c(0, 3))
  }
  if(i >= 34 & i <= 42){
    dfSzy[,i] <- scales::rescale(dfSzy[,i], c(0, 3))
  }
}

#rounding to integers so that the data resembles that which was collected
library(tidyverse)
dfSzy <- dfSzy %>% round(0)

#quick check of my work
#psych::describe(dfSzy)

#Reversing the supportive item on the Perceptions of LGBTQ Campus Climate Scale so that the ex

dfSzy <- dfSzy %>%
  dplyr::mutate(supportiveNR = 8 - supportive)

```

```
#Reversing three items on the Institutional and Goals Commitments scale so that the exercises work

dfSzy <- dfSzy %>%
  dplyr::mutate(not_graduateNR = 8 - not_graduate)%>%
  dplyr::mutate(undecidedNR = 8 - undecided)%>%
  dplyr::mutate(grades_unimportantNR = 8 - grades_unimportant)

dfSzy <- dplyr::select(dfSzy, -c(supportive, not_graduate, undecided, grades_unimportant))
```

The optional script below will let you save the simulated data to your computing environment as either an .rds object (preserves any formatting you might do) or a.csv file (think “Excel lite”).

```
# to save the df as an .rds (think 'R object') file on your computer;
# it should save in the same file as the .rmd file you are working
# with saveRDS(dfSzy, 'SzyDF.rds') bring back the simulated dat from
# an .rds file dfSzy <- readRDS('SzyDF.rds')
```

```
# write the simulated data as a .csv write.table(dfSzy,
# file='SzyDF.csv', sep=',', col.names=TRUE, row.names=FALSE) bring
# back the simulated dat from a .csv file dfSzy <-
# read.csv('SzyDF.csv', header = TRUE)
```

If we look at the information about this particular scale, we recognize that the *supportive* item is scaled in the opposite direction of the rest of the items. That is, a higher score on *supportive* would indicate a positive perception of the campus climate for LGBTQ individuals, whereas higher scores on the remaining items indicate a more negative perception. Before moving forward, we must reverse score this item.

In this psychometrics example I have given my variables one-word names that represent each item. Many researchers (including myself) will often give variable names that are alpha numerical: LGBTQ1, LGBTQ2, LGBTQn. In the psychometric evaluations, the one-word names may be useful shortcuts as one begins to understand the inter-item relations.

In reverse-scoring the *supportive* item, I will rename it “*unsupportive*” as an indication of its reversed direction.

```
library(tidyverse)

dfSzy <- dfSzy %>%
  dplyr::mutate(unsupportive = 8 - supportiveNR)  #when reverse-coding, subtract the variable value from 8

# When unhashtagged, this code provides item-level descriptive
# statistics psych::describe(dfSzy)
```

Next, I will create dfs that each contain the items of the total and subscales. These will be useful in the reliability estimates that follow.

Note that I am adding “T1” (time 1) to the end of the variable names. Later in the lesson when we evaluate test-retest reliability, we will simulate and add a “T2.”

```
LGBTQT1 <- dplyr::select(dfSzy, cold, unresponsive, unsupportive, negative,
  heterosexism, harassed)
ResponseT1 <- dplyr::select(dfSzy, cold, unresponsive, unsupportive)
StigmaT1 <- dplyr::select(dfSzy, negative, heterosexism, harassed)
```

As we move into the lecture, allow me to provide a content advisory. Individuals who hold LGBTQIA+ identities are frequently the recipients of discrimination and harassment. If you are curious about why these items are considered to be stigmatizing or non-responsive, please do not ask a member of the LGBTQIA+ community to explain it to you; it is not their job to educate others on discrimination, harassment, and microaggressions. Rather, please read the article in its entirety. Additionally, resources such as [The Trevor Project](#), [GLSEN](#), and [Campus Pride](#) are credible sources of information for learning more.

## 5.4 A Parade of Reliability Coefficients

While I cluster the reliability coefficients into large groups, please understand that these are somewhat overlapping.

Table 1 in Revelle and Condon’s [\[2019b\]](#) article provides a summary of the type of reliability tested, the findings, and the function used in the *psych* package.

### 5.4.1 Reliability Options for a Single Administration

If reliability is defined as the correlation between a test and a test just like it, how do we estimate the reliability of a single test, given only one time [\[Revelle and Condon, 2019b\]](#)? It may help to keep in mind that reliability is the ratio of true score variance to test score variance (or  $1 - \text{the ratio of error variance}$ ). Thus, the goal is to estimate the amount of error variance in the test. In this case we can investigate:

- a correlation between two random parts of the test
- internal consistency
- the internal structure of the test

#### 5.4.1.1 Split-half reliability

*Split-half reliability* is splitting a test into two random halves, correlating the two halves, and adjusting the correlation with the *Spearman-Brown* prophecy formula. Abundant formulaic detail in Revelle’s Chapter 7/Reliability [] .

An important question to split-half is “How to split?” Revelle terms it a “combinatorially difficult problem.” There are 126 possible splits for a 10-item scale; 6,345 possible splits for a 16-item scale; and over 4.5 billion for a 36-item scale! The *psych* package’s *splitHalf()* function will try all possible splits for scales of up to 16 items, then sample 10,000 splits for scales longer than that.

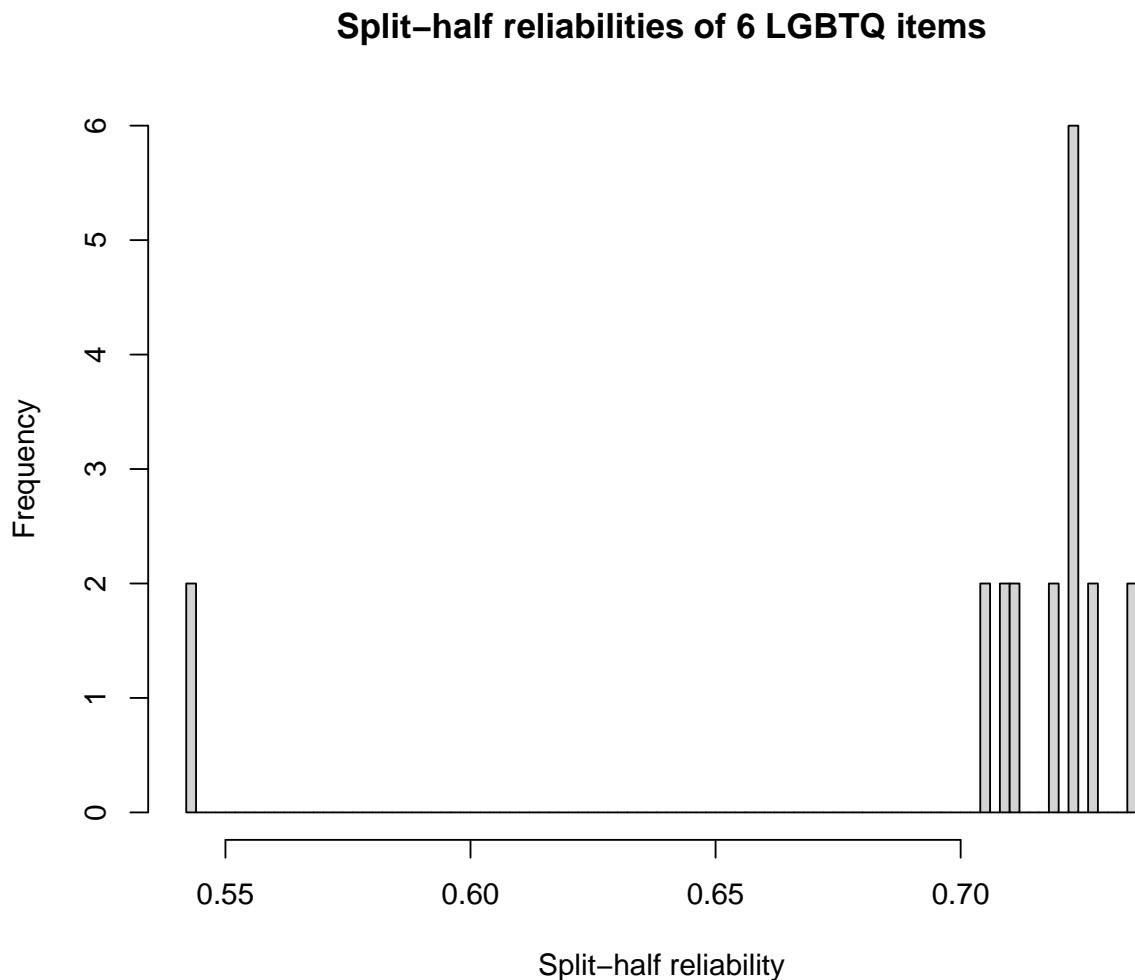
```
split <- psych::splitHalf(LGBTQT1, raw = TRUE, brute = TRUE)
split #show the results of the analysis
```

Split half reliabilities

Call: psych::splitHalf(r = LGBTQT1, raw = TRUE, brute = TRUE)

```
Maximum split half reliability (lambda 4) = 0.73
Guttman lambda 6                      = 0.68
Average split half reliability        = 0.7
Guttman lambda 3 (alpha)              = 0.7
Guttman lambda 2                      = 0.71
Minimum split half reliability (beta) = 0.54
Average interitem r = 0.28 with median = 0.25
                                         2.5% 50% 97.5%
Quantiles of split half reliability   = 0.54 0.72 0.73
```

```
hist(split$raw, breaks = 101, xlab = "Split-half reliability", main = "Split-half reliabilities")
```



Results of the split-half can provide some indication of whether not the scale is unidimensional.

In this case, the maximum reliability coefficient is 0.73, the average 0.70, and the lowest is 0.54. Similarly, we can examine the quantiles: 0.54, 0.72, 0.73.

The split-half output also includes the classic Cronbach's (1951) alpha coefficient (0.70; aka Guttman lambda 3) and average interitem correlations (0.25). The figure plots the frequencies of the reliability coefficient values.

While I did not find guidelines on what constitutes a “high enough lower bound” to establish homogeneity, Revelle suggested that a scale with 0.85, 0.80, and 0.65 had “strong evidence for a relatively homogeneous scale.” When the values were 0.81, 0.73, 0.42, Revelle indicated that there was “strong evidence for non-homogeneity” [Revelle and Condon, 2019a, p. 11]. In making this declaration, Revelle was also looking at the strength of the inter-item correlation and for a rather tight, bell-shaped, distribution at the higher ( $> 0.73$ ) end of the figure.

What happens when we examine the split-half estimates of the subscales? With only three items, there’s not much of a split and so the associated histogram will not be helpful.

```
splitRx <- psych::splitHalf(ResponseT1, raw = TRUE, brute = TRUE)
splitRx #show the results of the analysis
```

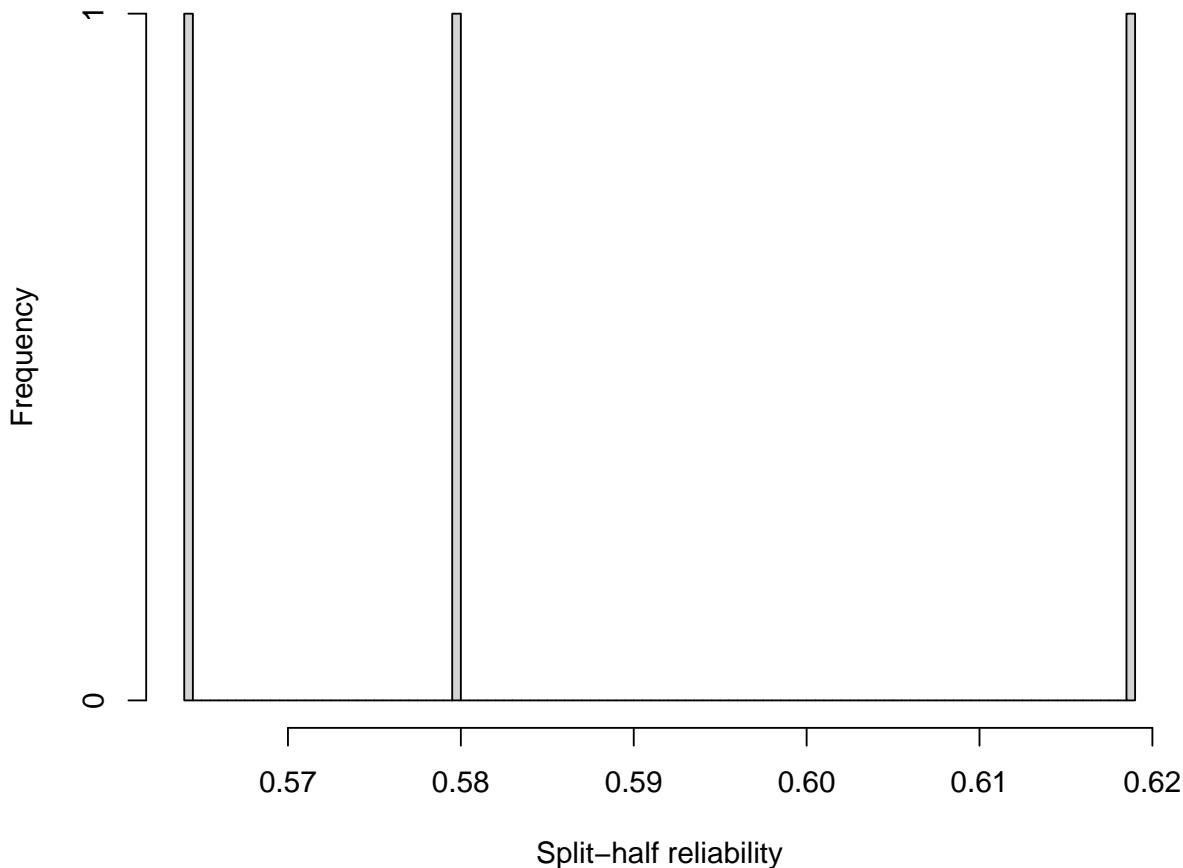
Split half reliabilities

Call: psych::splitHalf(r = ResponseT1, raw = TRUE, brute = TRUE)

```
Maximum split half reliability (lambda 4) = 0.62
Guttman lambda 6                      = 0.57
Average split half reliability        = 0.59
Guttman lambda 3 (alpha)              = 0.66
Guttman lambda 2                      = 0.66
Minimum split half reliability (beta) = 0.56
Average interitem r = 0.39 with median = 0.4
                                         2.5% 50% 97.5%
Quantiles of split half reliability   = 0.57 0.58 0.62
```

```
hist(splitRx$raw, breaks = 101, xlab = "Split-half reliability", main = "Split-half reliability")
```

### Split-half reliabilities of 3 items of the College Response subscale



The alpha is 0.66. The range of splits for max, ave, and low are 0.62, 0.59, and 0.55 and the quantiles are 0.57, 0.58, 0.62. The inter-item correlations have an average of 0.40.

Let's look at the split-half reliability coefficients for the Stigma subscale.

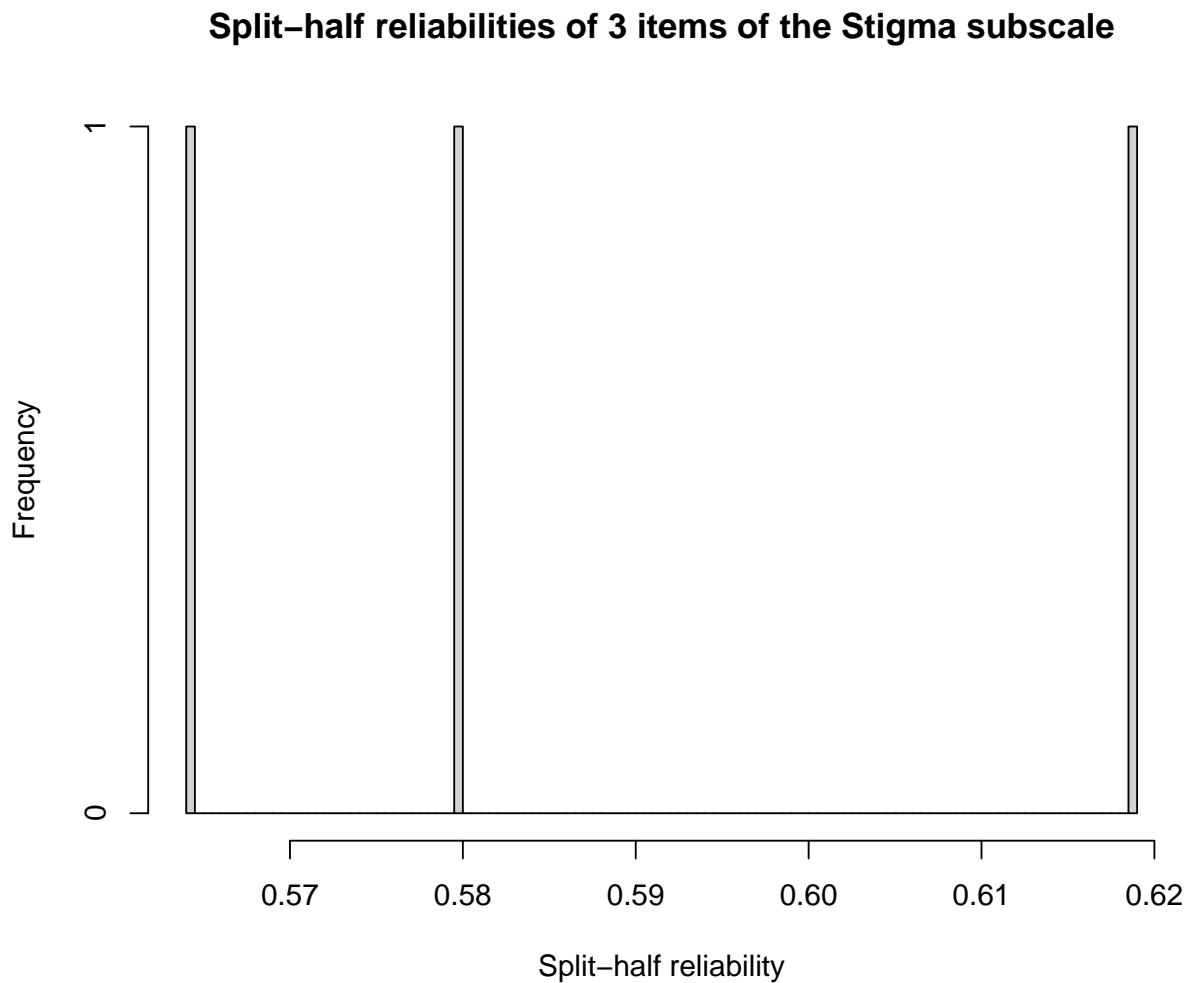
```
splitSt <- psych::splitHalf(StigmaT1, raw = TRUE, brute = TRUE)
splitSt #show the results of the analysis
```

```
Split half reliabilities
Call: psych::splitHalf(r = StigmaT1, raw = TRUE, brute = TRUE)

Maximum split half reliability (lambda 4) =  0.56
Guttman lambda 6                      =  0.53
Average split half reliability        =  0.56
Guttman lambda 3 (alpha)              =  0.63
Guttman lambda 2                      =  0.63
Minimum split half reliability (beta) =  0.55
Average interitem r =  0.36  with median =  0.36
```

	2.5% 50% 97.5%
Quantiles of split half reliability	= 0.55 0.56 0.56

```
hist(splitRx$raw, breaks = 101, xlab = "Split-half reliability", main = "Split-half reliabiliti
```



The alpha of this subscale is lower than the total scale score ( $\alpha = 0.60$ ). The maximum, average, and minimum split-half reliabilities were 0.56, 0.56, and 0.55; quantiles were at 0.55, 0.56, and 0.56. The average interitem correlation was 0.36.

Because the alpha coefficient can be defined as the “average of all possible split-half coefficients” for the groups tested, it is common for researchers to not provide split-half results in their papers – this is true for our research vignette. I continue to teach the split-half because it can be a stepping stone in the conceptualization of internal consistency as an estimate of reliability.

#### 5.4.1.2 Alpha coefficients

The most common methods to assess internal consistency are the *KR20* (for dichotomous items) and  $\alpha$  (for Likert scaling); alpha has an alias,  $\lambda_3$  (i.e., the Guttman lambda 3).

Alpha and the Guttman 3 (used for scales with Likert-type scaling) may be thought of as:

- a function of the number of items and the average correlation between the items
- the correlation of a test with a non-existent test just like it
- average of all possible split-half coefficients for the groups tested

Although the *psych* package has an incredible and thorough *alpha()* function, Revelle is not a fan of alpha. In fact, his alpha function reports a 95% CI around alpha as well as bootstrapped alpha results.

Let's grab alpha coefficients for our total and subscales.

```
psych::alpha(LGBTQT1)
```

Reliability analysis

Call: psych::alpha(x = LGBTQT1)

	raw_alpha	std.alpha	G6(smc)	average_r	S/N	ase	mean	sd	median_r	
	0.7	0.7	0.68		0.28	2.4	0.018	4	0.63	0.25

95% confidence boundaries

lower alpha upper

Feldt	0.66	0.7	0.74
-------	------	-----	------

Duhachek	0.66	0.7	0.74
----------	------	-----	------

Reliability if an item is dropped:

	raw_alpha	std.alpha	G6(smc)	average_r	S/N	alpha	se	var.r	med.r
cold	0.64	0.64	0.61		0.27	1.8	0.022	0.0066	0.22
unresponsive	0.66	0.66	0.63		0.28	2.0	0.021	0.0073	0.25
unsupportedive	0.67	0.67	0.63		0.29	2.0	0.021	0.0058	0.25
negative	0.66	0.66	0.63		0.28	2.0	0.021	0.0084	0.25
heterosexism	0.66	0.66	0.63		0.28	2.0	0.021	0.0087	0.25
harassed	0.67	0.67	0.64		0.29	2.0	0.021	0.0078	0.25

Item statistics

	n	raw.r	std.r	r.cor	r.drop	mean	sd
cold	646	0.68	0.68	0.59	0.49	4.1	1.03
unresponsive	646	0.63	0.63	0.51	0.43	4.3	0.99
unsupportedive	646	0.62	0.62	0.51	0.42	3.7	0.98
negative	646	0.64	0.63	0.51	0.42	4.0	1.04
heterosexism	646	0.61	0.63	0.51	0.43	4.0	0.90
harassed	646	0.63	0.61	0.49	0.41	3.9	1.07

Non missing response frequency for each item

	1	2	3	4	5	6	7	miss
cold	0.00	0.04	0.22	0.40	0.23	0.09	0.00	0

unresponsive	0.00	0.03	0.17	0.37	0.33	0.09	0.01	0
unsupportive	0.01	0.07	0.35	0.37	0.17	0.02	0.01	0
negative	0.01	0.07	0.23	0.39	0.24	0.05	0.00	0
heterosexism	0.00	0.03	0.24	0.43	0.26	0.03	0.00	0
harassed	0.01	0.07	0.27	0.37	0.22	0.05	0.01	0

The second screen of output shows the information we are interested in:

- **raw\_alpha**, .70 is based on the covariances
- **std.alpha**, .70 is based on correlations
- **average\_r**, .28 is the average inter-item correlation (i.e., all possible pairwise combinations of items)

```
psych::alpha(ResponseT1)
```

Reliability analysis

Call: psych::alpha(x = ResponseT1)

	raw_alpha	std.alpha	G6(smc)	average_r	S/N	ase	mean	sd	median_r
	0.66	0.66	0.57	0.39	1.9	0.023	4	0.77	0.4

95% confidence boundaries

lower alpha upper

Feldt	0.61	0.66	0.70
Duhachek	0.62	0.66	0.71

Reliability if an item is dropped:

	raw_alpha	std.alpha	G6(smc)	average_r	S/N	alpha	se	var.r	med.r
cold	0.52	0.52	0.35	0.35	1.1	0.038	NA	0.35	
unresponsive	0.60	0.60	0.42	0.42	1.5	0.032	NA	0.42	
unsupported	0.58	0.58	0.40	0.40	1.4	0.033	NA	0.40	

Item statistics

	n	raw.r	std.r	r.cor	r.drop	mean	sd
cold	646	0.80	0.79	0.62	0.50	4.1	1.03
unresponsive	646	0.76	0.76	0.55	0.45	4.3	0.99
unsupported	646	0.76	0.77	0.57	0.46	3.7	0.98

Non missing response frequency for each item

	1	2	3	4	5	6	7	miss
cold	0.00	0.04	0.22	0.40	0.23	0.09	0.00	0
unresponsive	0.00	0.03	0.17	0.37	0.33	0.09	0.01	0
unsupported	0.01	0.07	0.35	0.37	0.17	0.02	0.01	0

In the case of the College Response subscale:

- **raw\_alpha**, .66 is based on the covariances
- **std.apha**, .66 is based on correlations
- **average\_r**, .39 is the average interitem correlation

```
psych::alpha(StigmaT1)
```

Reliability analysis

Call: psych::alpha(x = StigmaT1)

	raw_alpha	std.alpha	G6(smc)	average_r	S/N	ase	mean	sd	median_r
	0.62	0.63	0.53	0.36	1.7	0.025	4	0.76	0.36

95% confidence boundaries

lower alpha upper

Feldt 0.57 0.62 0.67

Duhachek 0.57 0.62 0.67

Reliability if an item is dropped:

	raw_alpha	std.alpha	G6(smc)	average_r	S/N	alpha	se	var.r	med.r
negative	0.52	0.52	0.35	0.35	1.1	0.038	NA	0.35	
heterosexism	0.53	0.53	0.36	0.36	1.1	0.037	NA	0.36	
harassed	0.53	0.54	0.37	0.37	1.2	0.036	NA	0.37	

Item statistics

	n	raw.r	std.r	r.cor	r.drop	mean	sd
negative	646	0.77	0.76	0.56	0.44	4.0	1.0
heterosexism	646	0.73	0.76	0.55	0.44	4.0	0.9
harassed	646	0.77	0.75	0.54	0.43	3.9	1.1

Non missing response frequency for each item

	1	2	3	4	5	6	7	miss
negative	0.01	0.07	0.23	0.39	0.24	0.05	0.00	0
heterosexism	0.00	0.03	0.24	0.43	0.26	0.03	0.00	0
harassed	0.01	0.07	0.27	0.37	0.22	0.05	0.01	0

In the case of the Stigma subscale:

- **raw\_alpha**, .62 is based on the covariances
- **std.apha**, .63 is based on correlations
- **average\_r**, .36 is the average interitem correlation

The documentation for this package is incredible. Scrolling down through the description of the *alpha()* function provides a description of these different statistics.

Especially useful are item-level statistics:

- **r.drop** is the corrected item-total correlation (in the next lesson) for this item against the remaining items in the scale
- **mean** and **sd** are the mean and standard deviation of each item across all individuals.

The popularity of alpha emerged when tools available for calculation were less sophisticated; since then, we have learned that alpha can be misleading:

- Alpha inflates, somewhat artificially, even when inter-item correlations are low.
  - A 14-item scale will have an alpha of at least .70, even if it has two orthogonal (i.e., unrelated) scales [Cortina, 1993].
- Alpha assumes a unidimensional factor structure.
- The same alpha can be obtained for dramatically different underlying factor structures (see graphs in Revelle's Chapter 7).

The proper use of alpha requires the following:

- *Tau equivalence*, that is, equal covariances with the latent score represented by the test.
- *Unidimensionality*, equal factor loadings on the single factor of the test.

When either of these is violated, alpha underestimates reliability and overestimates the fraction of test variance that is associated with the general variance in the test.

Alpha and the split half are *internal consistency* estimates. Moving to *model-based* techniques allows us to take into consideration the factor structure of the scale. In the original article [Szymanski and Bissonette, 2020], results were as follows:

#### 5.4.1.3 Omega

Assessing reliability with *omega* ( $\omega$ ) statistics falls into a larger realm of *composite reliability* where reliability is assessed from a ratio of the variability explained by the items compared with the total variance of the entire scale [McNeish, 2018]. Members of the omega family of reliability estimates come from factor exploratory (i.e., EFA) and confirmatory (i.e., CFA; structural equation modeling [SEM]) factor analytic approaches. This lesson precedes the lessons on CFA and SEM. Therefore, my explanations and demonstrations will be somewhat brief. I intend to revisit omega output in the CFA and SEM lessons and encourage you to review this section now, then return to this section again after learning more about CFA and SEM.

In the context of *psychometrics*, it may be useful (albeit an oversimplification) to think of factors as scales/subscales where  $g$  refers to the amount of variance in the *general* factor (or total scale score) and subscales to be items that have something in common that is separate from what is  $g$ .

Model-based estimates examine the correlations or covariances of the items and decompose the test variance into that which is:

- common to all items (**g**, a general factor),
- specific to some items (**f**, orthogonal group factors), and

- unique to each item (confounding **s** specific, and **e** error variance)

$\omega$  is something of a shapeshifter. In the *psych* package:

- $\omega_t$  represents the total reliability of the test ( $\omega_t$ )
  - In the *psych* package, this is calculated from a bifactor model where there is one general *g* factor (i.e., each item loads on the single general factor), one or more group factors (*f*), and item-specific factors.
- $\omega_h$  extracts a higher-order factor from the correlation matrix of lower-level factors, then applies the Schmid and Leiman [1957] transformation to find the general loadings on the original items. Stated another way, it is a measure of the general factor saturation (*g*; the amount of variance attributable to one common factor). The subscript “*h*” acknowledges the hierarchical nature of the approach.
  - the  $\omega_h$  approach is exploratory and defined if there are three or more group factors (with only two group factors, the default is to assume they are equally important, hence the factor loadings of those subscales will be equal)
  - Najera Catalan [Najera Catalan, 2019] suggests that  $\omega_h$  is the best measure of reliability when dealing with multiple dimensions.
- $\omega_g$  is an estimate that uses a bifactor solution via the SEM package *lavaan* and tends to be a larger (because it forces all the cross loadings of lower-level factors to be 0)
  - the  $\omega_g$  is confirmatory, requiring the specification of which variables load on each group factor

Two commands in *psych* get us the results:

- *omega()* reports only the EFA solution
- *omegaSem()* reports both EFA and CFA solutions
  - We will use the *omegaSem()* function

Note that in our specification, we indicate there are two factors. We do not tell it what items belong to what factors (think, *subscals*). One test will be to see if the items align with their respective factors.

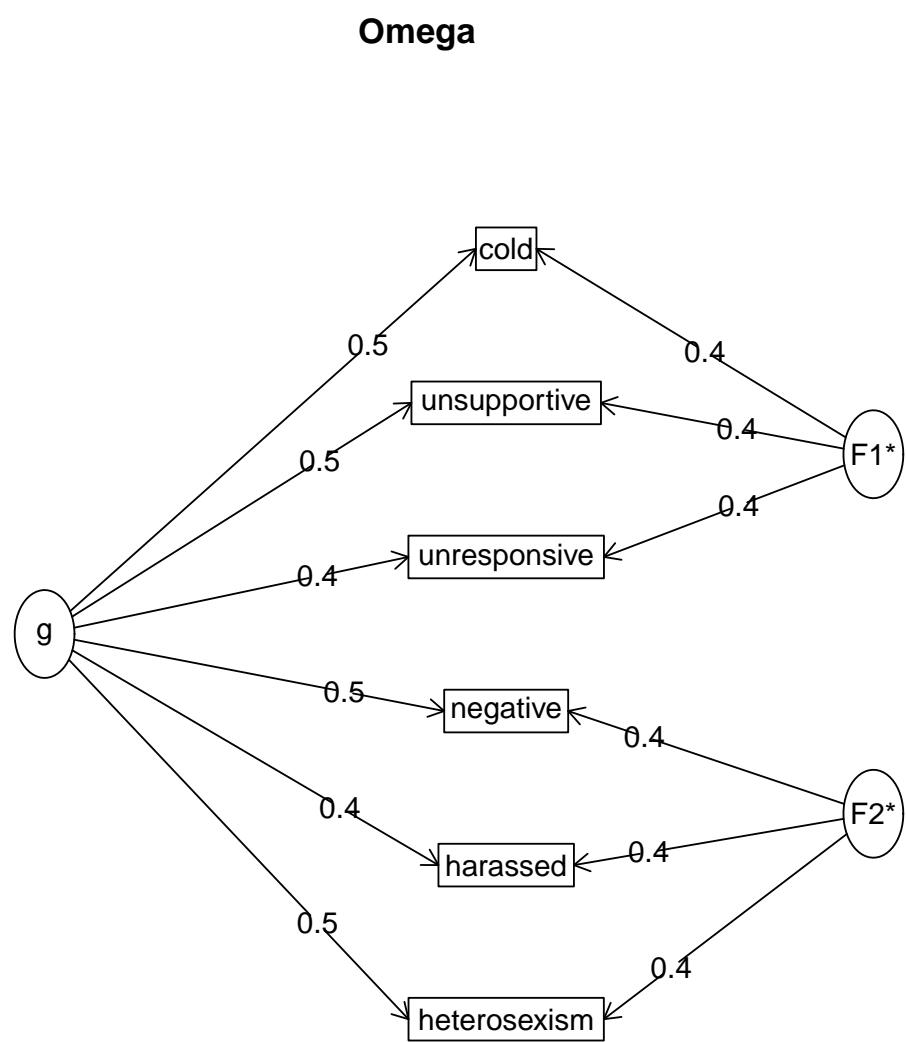
```
psych::omegaSem(LGBTQT1, nfactors = 2)
```

```
Loading required namespace: GPArotation
```

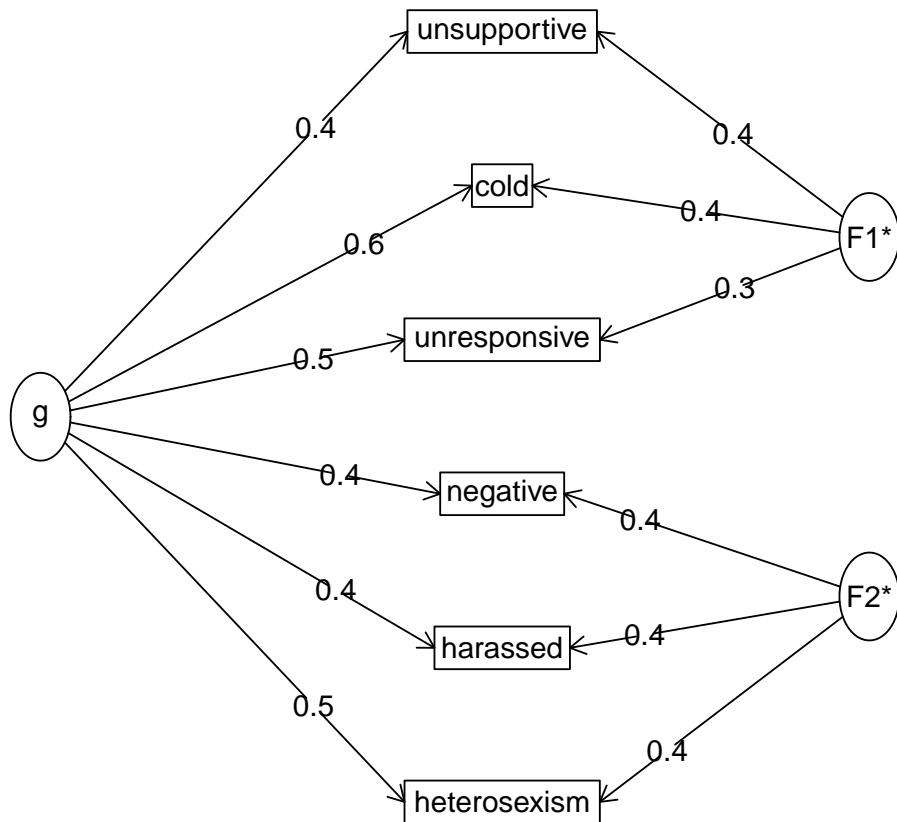
```
Three factors are required for identification -- general factor loadings set to be equal.
Proceed with caution.
```

```
Think about redoing the analysis with alternative values of the 'option' setting.
```

```
Warning in lav_model_vcov(lavmodel = lavmodel, lavsamplestats = lavsamplestats, : lavaan WARNING
  Could not compute standard errors! The information matrix could
  not be inverted. This may be a symptom that the model is not
  identified.
```



### Omega from SEM



```

Call: psych::omegaSem(m = LGBTQT1, nfactors = 2)
Omega
Call: omegah(m = m, nfactors = nfactors, fm = fm, key = key, flip = flip,
  digits = digits, title = title, sl = sl, labels = labels,
  plot = plot, n.obs = n.obs, rotate = rotate, Phi = Phi, option = option,
  covar = covar)
Alpha:          0.7
G.6:           0.68
Omega Hierarchical: 0.54
Omega H asymptotic: 0.73
Omega Total      0.74
  
```

Schmid Leiman Factor loadings greater than 0.2

	g	F1*	F2*	h2	u2	p2
cold	0.53	0.45		0.49	0.51	0.59
unresponsive	0.45	0.37		0.34	0.66	0.60

unsupportive	0.45	0.41	0.37	0.63	0.55	
negative	0.46		0.40	0.37	0.63	0.58
heterosexism	0.46		0.39	0.36	0.64	0.59
harassed	0.44		0.39	0.35	0.65	0.56

With Sums of squares of:

g	F1*	F2*
1.31	0.51	0.46

general/max 2.59 max/min = 1.1

mean percent general = 0.58 with sd = 0.02 and cv of 0.03

Explained Common Variance of the general factor = 0.58

The degrees of freedom are 4 and the fit is 0

The number of observations was 646 with Chi Square = 2.59 with prob < 0.63

The root mean square of the residuals is 0.01

The df corrected root mean square of the residuals is 0.02

RMSEA index = 0 and the 10 % confidence intervals are 0 0.049

BIC = -23.3

Compare this with the adequacy of just a general factor and no group factors

The degrees of freedom for just the general factor are 9 and the fit is 0.18

The number of observations was 646 with Chi Square = 115.31 with prob < 0.0000000000000000

The root mean square of the residuals is 0.1

The df corrected root mean square of the residuals is 0.13

RMSEA index = 0.135 and the 10 % confidence intervals are 0.114 0.158

BIC = 57.08

Measures of factor score adequacy

	g	F1*	F2*
Correlation of scores with factors	0.74	0.57	0.56
Multiple R square of scores with factors	0.54	0.33	0.31
Minimum correlation of factor score estimates	0.09	-0.34	-0.38

Total, General and Subset omega for each subset

	g	F1*	F2*
Omega total for total scores and subscales	0.74	0.66	0.63
Omega general for total scores and subscales	0.54	0.38	0.36
Omega group for total scores and subscales	0.20	0.28	0.27

The following analyses were done using the lavaan package

Omega Hierarchical from a confirmatory model using sem = 0.54

Omega Total from a confirmatory model using sem = 0.74

With loadings of

	g	F1*	F2*	h2	u2	p2	
cold		0.56	0.42		0.49	0.51	0.64

unresponsive	0.47	0.33	0.34	0.66	0.65
unsupportive	0.44	0.42	0.38	0.62	0.51
negative	0.45		0.42	0.37	0.63
heterosexism	0.46		0.39	0.36	0.64
harassed	0.43		0.40	0.34	0.66
					0.54

With sum of squared loadings of:

g	F1*	F2*
1.32	0.46	0.49

The degrees of freedom of the confirmatory model are 3 and the fit is 2.658827 with p = 0  
 general/max 2.71 max/min = 1.05  
 mean percent general = 0.58 with sd = 0.06 and cv of 0.1  
 Explained Common Variance of the general factor = 0.58

Measures of factor score adequacy

	g	F1*	F2*
Correlation of scores with factors	0.74	0.55	0.58
Multiple R square of scores with factors	0.55	0.31	0.33
Minimum correlation of factor score estimates	0.10	-0.39	-0.34

Total, General and Subset omega for each subset

	g	F1*	F2*
Omega total for total scores and subscales	0.74	0.66	0.63
Omega general for total scores and subscales	0.54	0.41	0.34
Omega group for total scores and subscales	0.20	0.26	0.28

To get the standard sem fit statistics, ask for summary on the fitted object

There is a ton of output! How do we make sense of it?

First, our items aligned perfectly with their respective factors (subsubscales). That is, it would be problematic if the items switched factors.

Second, we can interpret our results. Like alpha, the omegas range from 0 to 1, where values closer to 1 represent good reliability [Najera Catalan, 2019]. For unidimensional measures,  $\omega_t$  values above 0.80 indicate satisfactory reliability. For multidimensional measures with well-defined dimensions, we strive for  $\omega_h$  values above 0.65 (and  $\omega_t > 0.8$ ). These recommendations are based on a Monte Carlo study that examined a host of reliability indicators and how their values corresponded with accurate predictions of poverty status. With this in mind, let's examine the output related to our simulated research vignette.

Let's start with the output in the lower portion where the values are "from a confirmatory model using sem."

Omega is a reliability estimate for factor analysis that represents the proportion of variance in the LGBTQ scale attributable to common variance rather than error. The omega for the total reliability of the test ( $\omega_t$ ; which included the general factors and the subscale factors) was .74, meaning that 74% of the variance in the total scale is due to the factors and 26% (100% - 74%) is attributable to error.

Omega hierarchical ( $\omega_h$ ) estimates are the proportion of variance in the LGBTQ score attributable to the general factor, which in effect treats the subscales as error.  $\omega_h$  for the the LGBTQ total scale was .54. A quick calculation with  $\omega_h$  (.54) and  $\omega_t$  (.74;  $.54/.74 = .72$ ) lets us know that that 73% of the reliable variance in the LGBTQ total scale is attributable to the general factor.

```
.54/.74
```

```
[1] 0.7297297
```

Amongst the output is the Cronbach's alpha coefficient (.70). Szymanski and Bissonette [2020] did not report omega results; this may be because there were only two subfactors and/or they did not feel like a bifactor analysis would be appropriate. You might notice the lavaan warning indicating that three factors are needed in order to identify the CFA model. There is a longer explanation about factor identification. Stay tuned for CFA models.

#### 5.4.1.4 Some summary statements about reliability from single administrations

- With the exception of the worst split-half reliability and  $\omega_g$  or  $\omega_h$ , all of the reliability estimates are functions of test length and will tend asymptotically towards 1 as the number of items increases.
- The omega output provides a great deal more information about reliability than a simple alpha.
  - Figure 7.5 in [Revelle's chapter](#) shows four different structural representations of measures that have equal alphas (all .72)
- $\omega_{(h)}$ ,  $\beta$ , and the worst split-half reliability are estimates of the amount of general factor variance in the test scores
- In the case of low general factor saturation, the EFA based  $\omega_{(h)}$  is positively biased, so the CFA-based estimate,  $\omega_{(g)}$ , should be used.
- $\omega_{(t)}$  is the model-based estimate of the greatest lower bound of the total reliability of the test; so is the best split-half reliability.

Revelle and Condon's [2019a] recommendations to researchers:

- Report at least two coefficients (e.g.,  $\omega_{(h)}$  and  $\omega_{(t)}$ ) and discuss why each is appropriate for the inference that is being made.
- Report more than “just alpha” unless you can demonstrate that the measure is tau equivalent and unidimensional.

#### 5.4.2 Reliability Options for Two or more Administrations

##### 5.4.2.1 Test-retest of total scores

The purpose of test-retest reliability is to understand the stability of the measure over time. With two time points, T1 and T2, the test-retest correlation is an unknown mixture of trait, state, and specific variance, and is a function of the length of time between two measures.

- With two time points we cannot distinguish between trait and state effects, that said
  - we would expect a high degree of stability if the retest were (relatively) immediate
- With three time points we can leverage some SEM tools to distinguish between trait and state components
- A large test-retest correlation over a long period of time indicates temporal stability. Temporal stability is:
  - expected if we are assessing something trait like (e.g., cognitive ability, personality trait)
  - not expected if we are assessing something state like (e.g., emotional state, mood)
  - not expected if there was an intervention (or condition) and the T1 and T2 administrations are part of a pre- and post-test design.

There are some *methodological* concerns about test-retest reliability. For example, owing to memory and learning effects, the average response time to a second administration of identical items takes about 80% the time compared to the first administration.

Szymanski and Bissonette [2020] did not assess retest reliability. We can, though, imagine how this might work. Let's imagine that both waves were taken in the same academic term, approximately two weeks apart.

With both sets of data we need to create scores for the total scale score and the two subscales. We would also need to join the two datasets into a single dataframe.

To demonstrate the retest reliability, I simulated a new dataset with total and subscale scores for our variables for Time 1 and Time 2. This next script is simply that simulation (i.e., you can skip over it). If this were your data, you would have item-level data and need to calculate total and subscale scores (as we did above).

```
SimCor_mu <- c(3.13, 2.68, 3.58, 3.16, 2.66, 2.76)
SimCor_sd <- c(0.82, 1.04, 1.26, 0.83, 1.05, 0.99)
simCor <- matrix(c(1, 0.64, 0.77, 0.44, 0.33, 0.29, 0.64, 1, 0.53, 0.35,
  0.46, 0.34, 0.77, 0.53, 1, 0.27, 0.4, 0.47, 0.44, 0.35, 0.27, 1, 0.63,
  0.62, 0.33, 0.46, 0.4, 0.63, 1, 0.57, 0.29, 0.34, 0.47, 0.62, 0.57,
  1), ncol = 6)
scovMat <- SimCor_sd %*% t(SimCor_sd) * simCor
set.seed(210829)
retest_df <- MASS::mvrnorm(n = 646, mu = SimCor_mu, Sigma = scovMat, empirical = TRUE)
colnames(retest_df) <- c("TotalT1", "ResponseT1", "StigmaT1", "TotalT2",
  "ResponseT2", "StigmaT2")
retest_df <- as.data.frame(retest_df) #converts to a df so we can use in R
library(dplyr)
retest_df <- retest_df %>%
  dplyr::mutate(ID = row_number()) #add ID to each row
retest_df <- retest_df %>%
  dplyr::select(ID, everything()) #moving the ID number to the first column; requires
```

Examining our df, we can see the ID variable and the three sets of scores for each wave of analysis. Now we simply ask for their correlations. There are a number of ways to do this – the *apaTables* package can do the calculations and pop it into a manuscript-ready table.

We won't want the ID variable to be in the table.

```
retest_df2 <- retest_df %>%
  dplyr::select(c(-ID))

apaTables::apa.cor.table(data = retest_df2, landscape = TRUE, table.number = 1,
  filename = "Table_1_Retest.doc")
```

Table 1

Means, standard deviations, and correlations with confidence intervals

Variable	M	SD	1	2	3	4
1. TotalT1	3.13	0.82				
2. ResponseT1	2.68	1.04	.64** [.59, .68]			
3. StigmaT1	3.58	1.26	.77** [.74, .80]	.53** [.47, .58]		
4. TotalT2	3.16	0.83	.44** [.38, .50]	.35** [.28, .42]	.27** [.20, .34]	
5. ResponseT2	2.66	1.05	.33** [.26, .40]	.46** [.40, .52]	.40** [.33, .46]	.63** [.58, .67]
6. StigmaT2	2.76	0.99	.29** [.22, .36]	.34** [.27, .41]	.47** [.41, .53]	.62** [.57, .67]

.57\*\*  
 [.52, .62]

Note. M and SD are used to represent mean and standard deviation, respectively.  
 Values in square brackets indicate the 95% confidence interval.

The confidence interval is a plausible range of population correlations that could have caused the sample correlation (Cumming, 2014).

\* indicates  $p < .05$ . \*\* indicates  $p < .01$ .

As expected in this simulation,

- the strongest correlations are within each scale at their respective time, that is:
  - the T1 variables correlate with each other;
  - the T2 variables correlate with each other.
- the next strongest correlations are with the same scale/subscale configuration across time, for example
  - TotalT1 with TotalT2 ( $r = .44, p < 0.01$ )
  - ResponseT1 with ResponseT2 ( $r = .46, p < 0.01$ )
  - StigmaT1 with StigmaT2 ( $r = .47, p < 0.01$ )
- the lowest correlations are different scales at T1 and T2
  - ResponseT1 with StigmaT2 ( $r = .29$ )

The range of retest correlations (e.g., .44 to .47 with  $p < 0.01$ ) are sufficient to be confident in test-retest reliability.

#### 5.4.2.2 Test-retest recap

Here are some summary notions for retest reliability:

- Increases in the time interval will lower the reliability coefficient.
- An experimental intervention that is designed to impact the retest assessment will lower the reliability coefficient.
- State measures will have lower retest coefficients than trait measures.
- The three phenomena above all interact with each other.

Revelle and Condon's [2019a; 2019b] materials elaborate on this further. Their Table 1 is especially helpful. In addition to the myriad of vignettes used to illustrate issues with state, trait, items, whole scale, and so forth, there are demonstrations for duplicated items, assessing for consistency, and parallel/alternate forms.

If you are asking, "Hey, is parallel/alternate forms really a variant of test retest?" Great question! In fact, split-half could be seen as test-retest! Once you get in the weeds, the distinctions become less clear.

### 5.4.3 Interrater Reliability

#### 5.4.3.1 Cohen's kappa

Cohen's kappa coefficient is used to calculate proportions of agreement corrected for chance. This type of analysis occurs in research designs where there is some kind of (usually) categorical designation of a response. I don't have an outside research vignette for this. In the past, I was involved in research where members of the research team coded counselor utterances according to Hill's *helping skills* system designed by Clara Hill [Hill, 2020]. In the helping skills system, 15 different helping skills are divided into three larger groups that generally reflect the counseling trajectory: *exploration, insight, action*. One of our analyses coded counselor utterances into these three categories. Let's look at a fabricated (not based on any real data) simulation where four raters each evaluated 12 counselor utterances (that represent the arch of a nonsensically speedy counseling session).

```
Rater1 <- c("exploration", "exploration", "exploration", "exploration",
  "exploration", "exploration", "insight", "insight", "action", "action",
  "action", "action")
Rater2 <- c("exploration", "exploration", "exploration", "insight", "exploration",
  "insight", "exploration", "exploration", "exploration", "action", "exploration",
  "action")
Rater3 <- c("exploration", "insight", "exploration", "exploration", "exploration",
  "exploration", "exploration", "insight", "insight", "insight", "action",
  "action")
Rater4 <- c("exploration", "exploration", "exploration", "exploration",
  "exploration", "exploration", "exploration", "exploration", "exploration",
  "action", "action", "action")
ratings <- data.frame(Rater1, Rater2, Rater3, Rater4)
```

Historically, kappa could only be calculated for 2 raters at a time. Presently, though, it appears there can be any number of raters and the average agreement is reported.

Let's take a look at the data, then run the analysis, and interpret the results.

```
psych::cohen.kappa(ratings)
```

Cohen Kappa (below the diagonal) and Weighted Kappa (above the diagonal)  
For confidence intervals and detail print with all=TRUE

	Rater1	Rater2	Rater3	Rater4
Rater1	1.00	0.40	0.21	0.62
Rater2	0.14	1.00	0.00	0.57
Rater3	0.48	0.00	1.00	0.30
Rater4	0.54	0.45	0.43	1.00

Average Cohen kappa for all raters 0.34  
Average weighted kappa for all raters 0.35

Kappa can range from -1.00 to 1.00.

- $K = .00$  indicates that the observed agreement is exactly equal to the agreement that could be observed by chance.
- Negative kappa indicates that observed kappa is less than the expected chance agreement.
- $K = 1.00$  equals perfect agreement between judges.

There are commonly understood concerns about using kappa:

- Research teams typically set an expected standard (e.g., .85) and train raters until kappa is achieved.
  - In lengthy projects, rating agreement is rechecked periodically; if necessary there is retraining.
- Obtaining an acceptable kappa becomes difficult as the number of categories increases.
  - An example is Hill's *Helping Skills System* when all 15 categories; we chose to use the three categories (into which the 15 categories are subsumed).
- It is also difficult to obtain an adequate kappa when *infrequent* categories (e.g., "insight") exist.

Our kappa of .35 indicates that this rating team has a 35% chance of agreement, corrected for by chance. This is substantially below the standard. Let's imagine that the team spends time with their dictionaries, examines common errors, and makes some decision rules.

Here's the resimulation of the "improved" agreement.

```
Rater1b <- c("exploration", "exploration", "exploration", "exploration",
           "exploration", "exploration", "insight", "insight", "insight", "action",
           "action", "action")
Rater2b <- c("exploration", "exploration", "exploration", "exploration",
           "exploration", "insight", "insight", "insight", "exploration", "action",
           "action", "action")
Rater3b <- c("exploration", "exploration", "exploration", "exploration",
           "exploration", "exploration", "exploration", "insight", "insight",
           "insight", "action", "action")
Rater4b <- c("exploration", "exploration", "exploration", "exploration",
           "exploration", "exploration", "exploration", "exploration", "insight",
           "action", "action", "action")
after_training <- data.frame(Rater1b, Rater2b, Rater3b, Rater4b)
```

Now run it again.

```
psych::cohen.kappa(after_training)
```

```
Warning in cohen.kappa1(x1, w = w, n.obs = n.obs, alpha = alpha, levels =
levels): upper or lower confidence interval exceed abs(1) and set to +/- 1.
```

Warning in cohen.kappa1(x1, w = w, n.obs = n.obs, alpha = alpha, levels = levels): upper or lower confidence interval exceed abs(1) and set to +/- 1.

Warning in cohen.kappa1(x1, w = w, n.obs = n.obs, alpha = alpha, levels = levels): upper or lower confidence interval exceed abs(1) and set to +/- 1.

Warning in cohen.kappa1(x1, w = w, n.obs = n.obs, alpha = alpha, levels = levels): upper or lower confidence interval exceed abs(1) and set to +/- 1.

Cohen Kappa (below the diagonal) and Weighted Kappa (above the diagonal)

For confidence intervals and detail print with all=TRUE

	Rater1b	Rater2b	Rater3b	Rater4b
Rater1b	1.00	0.83	0.55	0.80
Rater2b	0.73	1.00	0.36	0.60
Rater3b	0.72	0.45	1.00	0.46
Rater4b	0.71	0.43	0.70	1.00

Average Cohen kappa for all raters 0.62

Average weighted kappa for all raters 0.6

We observe improved scores, but this team needs more training if we aspire to a kappa of 0.85!

### 5.4.3.2 Intraclass correlation (ICC)

Another option for interrater reliability is the intraclass correlation (ICC). This is the same ICC we use in multilevel modeling! The ICC is used when we have numerical ratings.

In our fabricated vignette below, five raters are evaluating the campus climate for LGBTQIA+ individuals for 10 units/departments on a college campus. Using the ICC can help us determine the degree of leniency and variability within judges.

Below is a simulation of the data (you can ignore this)...

```
Rater1 <- c(1, 1, 1, 4, 2, 3, 1, 3, 3, 5)
Rater2 <- c(1, 1, 2, 1, 4, 4, 4, 4, 5, 5)
Rater3 <- c(3, 3, 3, 2, 3, 3, 6, 4, 4, 5)
Rater4 <- c(3, 5, 4, 2, 3, 6, 6, 6, 5, 5)
Rater5 <- c(2, 3, 3, 3, 4, 4, 4, 4, 5, 5)
ICC_df <- data.frame(Rater1, Rater2, Rater3, Rater4, Rater5)
```

```
#If the code below will not run remove the hashtags from the two lines of code below to install
```

```
#tools::package_dependencies("Matrix", which = "LinkingTo", reverse = TRUE) [[1L]]
#install.packages("lme4", type = "source")
```

We can use the *psych::ICC* function to obtain the ICC values.

```
# psych::ICC(ICC_df [1:10,1:5], lmer = TRUE) #find the ICCs for the
# 10 campus units and 5 judges
psych::ICC(ICC_df, missing = TRUE, alpha = 0.05, lmer = TRUE, check.keys = FALSE)
```

```
Call: psych::ICC(x = ICC_df, missing = TRUE, alpha = 0.05, lmer = TRUE,
check.keys = FALSE)
```

#### Intraclass correlation coefficients

	type	ICC	F	df1	df2	p	lower bound	upper bound
Single_raters_absolute	ICC1	0.34	3.5	9	40	0.00259	0.082	0.70
Single_random_raters	ICC2	0.37	5.4	9	36	0.00011	0.118	0.71
Single_fixed_raters	ICC3	0.47	5.4	9	36	0.00011	0.188	0.78
Average_raters_absolute	ICC1k	0.72	3.5	9	40	0.00259	0.308	0.92
Average_random_raters	ICC2k	0.74	5.4	9	36	0.00011	0.400	0.92
Average_fixed_raters	ICC3k	0.81	5.4	9	36	0.00011	0.537	0.95

Number of subjects = 10      Number of Judges = 5

See the help file for a discussion of the other 4 McGraw and Wong estimates,

In the output, reliability for a single judge  $ICC_1$  is the ratio of person variance to total variance. Reliability for multiple judges  $ICC_{1k}$  adjusts the residual variance by the number of judges.

The ICC function reports six reliability coefficients: 3 for the case of single judges and 3 for the case of multiple judges. It also reports the results in terms of a traditional ANOVA as well as a mixed effects linear model. Additionally, confidence intervals are reported.

Like most correlation coefficients, the ICC ranges from 0 to 1.

- An ICC close to 1 indicates high similarity between values from the same group.
- An ICC close to zero means that values from the same group are not similar.

## 5.5 What do we do with these coefficients?

### 5.5.1 Corrections for attenuation

Circa 1904, Spearman created the reliability coefficient out of a need to adjust observed correlations between related constructs for the error of measurement in each construct. This is only appropriate if the measure is seen as the expected value of a single underlying construct. However, “under the hood,” SEM programs model the pattern of observed correlations in terms of a measurement (reliability) model as well as a structural (validity) model.

### 5.5.2 Predicting true scores (and their CIs)

True scores remain unknown and so the reliability coefficient is used in a couple of ways to estimate the true score (and the confidence interval [CI] around that true score).

Take a quick look at the formula for predicting a true score and observe that the reliability coefficient is used within. It generally serves to nudge the observed score a bit closer to the mean:  $T' = (1 - r_{xx})\bar{X} + r_{xx}X$

The CI around that true score includes some estimate of standard error:  $CI_{95} = T' + / - z_{cv}(s_e)$ . Two estimates are commonly used. One is the standard error of estimate  $s_e = s_x \sqrt{r_{xx}(1 - r_{xx})}$  (i.e., the standard deviation of predicted true scores for a given observed score). Another is the standard error of measurement ( $s_m = s_x \sqrt{1 - r_{xx}}$ ) (i.e., an estimate of the amount of variation to be expected in test scores; aka, the standard deviation of the errors of measurement).

*I can hear you asking* What is the difference between  $s_e$  and  $s_m$ ?

- Because  $r_{xx}$  is almost always a fraction,  $s_e$  is smaller than  $s_m$ .
- When the reliability is high, the two standard errors are fairly similar to each other.
- Using  $s_m$  will result in wider confidence intervals.

### 5.5.3 How do I keep it all straight?

Table 1 in Revelle and Condon's [Revelle and Condon, 2019a] article helps us connect the type of reliability we are seeking with the statistic(s) and the R function within the *psych* package.

## 5.6 Practice Problems

In each of these lessons I provide suggestions for practice that allow you to select one or more problems that are graded in difficulty. The practice problems are the start of a larger project that spans multiple lessons. Therefore, if possible, please use a dataset that has item-level data for which there is a theorized total scale score as well as two or more subscales. With each of these options I encourage you to:

- Format (i.e., rescore if necessary) a dataset so that it is possible to calculate estimates of internal consistency
- Calculate and report the alpha coefficient for a total scale scores and subscales (if the scale has them)
- Calculate and report  $\omega_t$  and  $\omega_h$ . With these two determine what proportion of the variance is due to all the factors, error, and  $g$ .
- Calculate total and subscale scores.
- Describe other reliability estimates that would be appropriate for the measure you are evaluating.

### 5.6.1 Problem #1: Play around with this simulation.

If evaluating internal consistency is new to you, copy the script for the simulation and then change (at least) one thing in the simulation to see how it impacts the results. Perhaps you just change the number in "set.seed(210827)" from 210827 to something else. Your results should parallel those obtained in the lecture, making it easier for you to check your work as you go.

### 5.6.2 Problem #2: Use the data from the live ReCentering Psych Stats survey.

The script below pulls live data directly from the ReCentering Psych Stats survey on Qualtrics. As described in the [Scrubbing and Scoring chapters](#) of the ReCentering Psych Stats Multivariate Modeling volume, the Perceptions of the LGBTQ College Campus Climate Scale [Szymanski and Bissonette, 2020] was included (LGBTQ) and further adapted to assess perceptions of campus climate for Black students (BLst), non-Black students of color (nBSoC), international students (INTst), and students disabilities (wDIS). Consider conducting the analyses on one of these scales or merging them together.

```
library(tidyverse)
# only have to run this ONCE to draw from the same Qualtrics
# account...but will need to get different token if you are changing
# between accounts
library(qualtRics)
# qualtrics_api_credentials(api_key =
# 'mUgPMYsSYkiWpMFkwHale1QE5HNmh5LRUaA8d9PDg', base_url =
# 'spupsych.az1.qualtrics.com', overwrite = TRUE, install = TRUE)
QTRX_df <- qualtRics::fetch_survey(surveyID = "SV_b2cClqA1LGQ6nLU", time_zone = NULL,
verbose = FALSE, label = FALSE, convert = FALSE, force_request = TRUE,
import_id = FALSE)
climate_df <- QTRX_df %>%
  select("Blst_1", "Blst_2", "Blst_3", "Blst_4", "Blst_5", "Blst_6",
         "nBSoC_1", "nBSoC_2", "nBSoC_3", "nBSoC_4", "nBSoC_5", "nBSoC_6",
         "INTst_1", "INTst_2", "INTst_3", "INTst_4", "INTst_5", "INTst_6",
         "wDIS_1", "wDIS_2", "wDIS_3", "wDIS_4", "wDIS_5", "wDIS_6", "LGBTQ_1",
         "LGBTQ_2", "LGBTQ_3", "LGBTQ_4", "LGBTQ_5", "LGBTQ_6")
# Item numbers are supported with the following items: _1 'My campus
# unit provides a supportive environment for ___ students' _2
# '_____ is visible in my campus unit' _3 'Negative attitudes
# toward persons who are ___ are openly expressed in my campus
# unit.' _4 'My campus unit is unresponsive to the needs of ___
# students.' _5 'Students who are_____ are harassed in my campus
# unit.' _6 'My campus unit is cold and uncaring toward ___
# students.'
# Item 1 on each subscale should be reverse coded. The College
# Response scale is composed of items 1, 4, 6, The Stigma scale is
# composed of items 2,3, 5
```

The optional script below will let you save the simulated data to your computing environment as either a .csv file (think “Excel lite”) or .rds object (preserves any formatting you might do).

```
# write the simulated data as a .csv write.table(climate_df,
# file='climate_df.csv', sep=',', col.names=TRUE, row.names=FALSE)
# bring back the simulated dat from a .csv file climate_df <-
# read.csv ('climate_df.csv', header = TRUE)
```

```
# to save the df as an .rds (think 'R object') file on your computer;
# it should save in the same file as the .rmd file you are working
# with saveRDS(climate_df, 'climate_df.rds') bring back the simulated
# dat from an .rds file climate_df <- readRDS('climate_df.rds')
```

### 5.6.3 Problem #3: Try something entirely new.

Complete the same steps using data for which you have permission and access. This might be data of your own, from your lab, simulated from an article, or located on an open repository.

### 5.6.4 Grading Rubric

Assignment Component	Points Possible	Points Earned
1. Check and, if needed, format and score data	5	_____
2. Calculate and report the alpha coefficient for a total scale scores and subscales (if the scale has them)	5	_____
3. Calculate and report $\omega_t$ and $\omega_h$ . With these two determine what proportion of the variance is due to all the factors, error, and $g$ .	5	_____
4. Calculate total and subscale scores.	5	_____
5. Describe other reliability estimates that would be appropriate for the measure you are evaluating.	5	_____
6. Explanation to grader	5	_____
<b>Totals</b>	<b>30</b>	_____

## 5.7 Homeworked Example

### Screencast Link

For more information about the data used in this homeworked example, please refer to the description and codebook located at the end of the [introduction](#) in first volume of ReCentering Psych Stats.

As a brief review, this data is part of an IRB-approved study, with consent to use in teaching demonstrations and to be made available to the general public via the open science framework. Hence, it is appropriate to use in this context. You will notice there are student- and teacher- IDs. These numbers are not actual student and teacher IDs, rather they were further re-identified so that they could not be connected to actual people.

Because this is an actual dataset, if you wish to work the problem along with me, you will need to download the [ReC.rds](#) data file from the Worked\_Examples folder in the ReC\_Psychometrics project on the GitHub.

The course evaluation items can be divided into three subscales:

- **Valued by the student** includes the items: ValObjectives, IncrUnderstanding, IncrInterest
- **Traditional pedagogy** includes the items: ClearResponsibilities, EffectiveAnswers, Feedback, ClearOrganization, ClearPresentation
- **Socially responsive pedagogy** includes the items: InclusvClassrm, EquitableEval, MultPerspectives, DEIntegration

In this homework focused on reliability we will report alpha coefficients for total scale score and subscale scores. We'll also calculate omega total and omega hierarchical and determine what proportion of variance is due to all the factors, error, and  $g$ . Finally, we'll calculate total and subscale scores.

### 5.7.1 Check and, if needed, format the data

```
big <- readRDS("ReC.rds")
```

Let's check the structure...

```
str(big)
```

```
Classes 'data.table' and 'data.frame': 310 obs. of 33 variables:
 $ deID                  : int 1 2 3 4 5 6 7 8 9 10 ...
 $ CourseID               : int 57085635 57085635 57085635 57085635 57085635 57085635 57085635 57085635 ...
 $ Dept                   : chr "CPY" "CPY" "CPY" "CPY" ...
 $ Course                 : Factor w/ 3 levels "Psychometrics",...: 2 2 2 2 2 2 2 2 2 2 ...
 $ StatsPkg                : Factor w/ 2 levels "SPSS","R": 2 2 2 2 2 2 2 2 2 2 ...
 $ Centering               : Factor w/ 2 levels "Pre","Re": 2 2 2 2 2 2 2 2 2 2 ...
 $ Year                    : int 2021 2021 2021 2021 2021 2021 2021 2021 2021 2021 ...
 $ Quarter                 : chr "Fall" "Fall" "Fall" "Fall" ...
 $ IncrInterest             : int 5 3 4 2 4 3 5 3 2 5 ...
 $ IncrUnderstanding         : int 2 3 4 3 4 4 5 2 4 5 ...
 $ ValObjectives            : int 5 5 4 4 5 5 5 5 4 5 ...
 $ ApprAssignments           : int 5 4 4 4 5 3 5 3 3 5 ...
 $ EffectiveAnswers          : int 5 3 5 3 5 3 4 3 2 3 ...
 $ Respectful                : int 5 5 4 5 5 4 5 4 5 5 ...
 $ ClearResponsibilities     : int 5 5 4 4 5 4 5 4 4 5 ...
 $ Feedback                  : int 5 3 4 2 5 NA 5 4 4 5 ...
 $ OvInstructor              : int 5 4 4 3 5 3 5 4 3 5 ...
 $ MultPerspectives           : int 5 5 4 5 5 4 5 5 5 5 ...
 $ OvCourse                  : int 3 4 4 3 5 3 5 3 2 5 ...
 $ InclusvClassrm            : int 5 5 5 5 5 4 5 5 5 4 5 ...
 $ DEIntegration              : int 5 5 5 5 5 4 5 5 5 5 5 ...
 $ ClearPresentation           : int 4 4 4 2 5 3 4 4 4 5 ...
 $ ApprWorkload               : int 5 5 3 4 4 2 5 4 4 5 ...
 $ MyContribution              : int 4 4 4 4 5 4 4 3 4 5 ...
 $ InspiredInterest            : int 5 3 4 3 5 3 5 4 4 5 ...
```

```
$ Faith : int 5 NA 4 2 NA NA 4 4 4 NA ...
$ EquitableEval : int 5 5 3 5 5 3 5 5 3 5 ...
$ SPFC.Decolonize.Opt.Out: chr "" "" "" ...
$ ProgramYear : Factor w/ 3 levels "Second","Transition",...: 3 3 3 3 3 3 3 3 3 ...
$ ClearOrganization : int 3 4 3 4 4 4 5 4 4 5 ...
$ RegPrepare : int 5 4 4 4 4 3 4 4 4 5 ...
$ EffectiveLearning : int 2 4 3 4 4 2 5 3 2 5 ...
$ AccessibleInstructor : int 5 4 4 4 5 4 5 4 5 5 ...
- attr(*, ".internal.selfref")=<externalptr>
```

Let's create a df with the items only.

```
library(tidyverse)
items <- big %>%
  dplyr::select(ValObjectives, IncrUnderstanding, IncrInterest, ClearResponsibilities,
    EffectiveAnswers, Feedback, ClearOrganization, ClearPresentation,
    MultPerspectives, InclusvClassrm, DEIintegration, EquitableEval)
```

### 5.7.2 Calculate and report the alpha coefficient for a total scale score and subscales (if the scale has them)

```
psych::alpha(items)
```

```
Reliability analysis
Call: psych::alpha(x = items)
```

	raw_alpha	std.alpha	G6(smc)	average_r	S/N	ase	mean	sd	median_r
	0.92	0.92	0.93	0.49	11	0.0065	4.3	0.61	0.48

```
95% confidence boundaries
      lower alpha upper
Feldt     0.90   0.92   0.93
Duhachek  0.91   0.92   0.93
```

Reliability if an item is dropped:

	raw_alpha	std.alpha	G6(smc)	average_r	S/N	alpha	se	var.r
ValObjectives	0.92	0.92	0.93	0.51	11.3	0.0067	0.016	
IncrUnderstanding	0.91	0.91	0.92	0.49	10.6	0.0070	0.016	
IncrInterest	0.91	0.91	0.92	0.49	10.4	0.0070	0.018	
ClearResponsibilities	0.91	0.91	0.92	0.48	10.0	0.0073	0.015	
EffectiveAnswers	0.91	0.91	0.92	0.48	10.0	0.0074	0.016	
Feedback	0.91	0.91	0.92	0.48	10.3	0.0071	0.018	
ClearOrganization	0.91	0.91	0.92	0.48	10.2	0.0073	0.016	
ClearPresentation	0.91	0.91	0.92	0.47	9.7	0.0076	0.015	

MultPerspectives	0.91	0.91	0.92	0.48	10.0	0.0073	0.017
InclusvClassrm	0.91	0.91	0.92	0.49	10.6	0.0069	0.018
DEIIntegration	0.92	0.92	0.93	0.52	11.8	0.0063	0.011
EquitableEval	0.91	0.91	0.93	0.49	10.5	0.0070	0.018
med.r							
ValObjectives	0.53						
IncrUnderstanding	0.50						
IncrInterest	0.48						
ClearResponsibilities	0.48						
EffectiveAnswers	0.48						
Feedback	0.48						
ClearOrganization	0.48						
ClearPresentation	0.47						
MultPerspectives	0.47						
InclusvClassrm	0.52						
DEIIntegration	0.53						
EquitableEval	0.48						

## Item statistics

	n	raw.r	std.r	r.cor	r.drop	mean	sd
ValObjectives	309	0.59	0.61	0.55	0.53	4.5	0.61
IncrUnderstanding	309	0.71	0.70	0.67	0.64	4.3	0.82
IncrInterest	308	0.75	0.73	0.71	0.68	3.9	0.99
ClearResponsibilities	307	0.80	0.80	0.79	0.75	4.4	0.82
EffectiveAnswers	308	0.80	0.79	0.78	0.75	4.4	0.83
Feedback	304	0.75	0.75	0.72	0.69	4.2	0.88
ClearOrganization	309	0.79	0.77	0.75	0.72	4.0	1.08
ClearPresentation	309	0.85	0.84	0.83	0.80	4.2	0.92
MultPerspectives	305	0.79	0.80	0.78	0.75	4.4	0.84
InclusvClassrm	301	0.68	0.70	0.67	0.62	4.6	0.68
DEIIntegration	273	0.51	0.53	0.49	0.42	4.5	0.74
EquitableEval	308	0.70	0.72	0.69	0.66	4.6	0.63

## Non missing response frequency for each item

	1	2	3	4	5	miss
ValObjectives	0.00	0.01	0.03	0.39	0.57	0.00
IncrUnderstanding	0.01	0.04	0.07	0.44	0.45	0.00
IncrInterest	0.02	0.09	0.14	0.44	0.31	0.01
ClearResponsibilities	0.01	0.02	0.07	0.31	0.59	0.01
EffectiveAnswers	0.01	0.02	0.08	0.36	0.53	0.01
Feedback	0.01	0.05	0.10	0.39	0.46	0.02
ClearOrganization	0.04	0.07	0.10	0.41	0.38	0.00
ClearPresentation	0.02	0.05	0.07	0.40	0.46	0.00
MultPerspectives	0.02	0.02	0.08	0.33	0.56	0.02
InclusvClassrm	0.01	0.01	0.05	0.23	0.70	0.03
DEIIntegration	0.00	0.01	0.10	0.22	0.67	0.12
EquitableEval	0.00	0.01	0.03	0.32	0.63	0.01

Total scale score alpha is 0.92

### 5.7.3 Subscale alphas

In the lecture, I created baby dfs of the subscales and ran the alpha on those; another option is to use concatenated lists of variables (i.e., variable vectors). Later, we can also use these to score the subscales.

```
ValuedVars <- c("ValObjectives", "IncrUnderstanding", "IncrInterest")
TradPedVars <- c("ClearResponsibilities", "EffectiveAnswers", "Feedback",
                 "ClearOrganization", "ClearPresentation")
SRPedVars <- c("InclusvClassrm", "EquitableEval", "MultPerspectives", "DEIintegration")

psych::alpha(items[, ValuedVars])
```

```
Reliability analysis
Call: psych::alpha(x = items[, ValuedVars])

  raw_alpha std.alpha G6(smc) average_r S/N   ase mean    sd median_r
 0.77      0.77      0.71      0.53 3.4 0.02  4.2 0.68      0.48

  95% confidence boundaries
        lower alpha upper
Feldt     0.72  0.77  0.81
Duhachek 0.73  0.77  0.81

Reliability if an item is dropped:
  raw_alpha std.alpha G6(smc) average_r S/N   alpha se var.r
ValObjectives       0.80      0.81      0.68      0.68 4.3  0.022  NA
IncrUnderstanding    0.60      0.65      0.48      0.48 1.8  0.040  NA
IncrInterest        0.59      0.61      0.44      0.44 1.6  0.044  NA
                    med.r
ValObjectives       0.68
IncrUnderstanding    0.48
IncrInterest         0.44

Item statistics
  n raw.r std.r r.cor r.drop mean    sd
ValObjectives     309  0.71  0.77  0.55   0.50  4.5 0.61
IncrUnderstanding 309  0.86  0.85  0.76   0.68  4.3 0.82
IncrInterest      308  0.90  0.87  0.79   0.70  3.9 0.99

Non missing response frequency for each item
  1   2   3   4   5 miss
ValObjectives     0.00 0.01 0.03 0.39 0.57 0.00
```

```
IncrUnderstanding 0.01 0.04 0.07 0.44 0.45 0.00
IncrInterest      0.02 0.09 0.14 0.44 0.31 0.01
```

Alpha for the Valued-by-Me dimension is .77

```
psych::alpha(items[, TradPedVars])
```

```
Reliability analysis
Call: psych::alpha(x = items[, TradPedVars])
```

	raw_alpha	std.alpha	G6(smc)	average_r	S/N	ase	mean	sd	median_r
	0.89	0.9	0.88	0.64	8.8	0.0094	4.3	0.76	0.65

	95% confidence boundaries		
	lower	alpha	upper
Feldt	0.87	0.89	0.91
Duhachek	0.88	0.89	0.91

Reliability if an item is dropped:

	raw_alpha	std.alpha	G6(smc)	average_r	S/N	alpha	se	var.r	med.r
ClearResponsibilities	0.86	0.86	0.84	0.62	6.4	0.013	0.0054		0.59
EffectiveAnswers	0.87	0.87	0.84	0.63	6.8	0.012	0.0045		0.65
Feedback	0.89	0.89	0.87	0.68	8.4	0.010	0.0016		0.69
ClearOrganization	0.88	0.88	0.85	0.64	7.2	0.012	0.0044		0.66
ClearPresentation	0.86	0.87	0.83	0.62	6.5	0.013	0.0030		0.62

Item statistics

	n	raw.r	std.r	r.cor	r.drop	mean	sd
ClearResponsibilities	307	0.87	0.87	0.84	0.79	4.4	0.82
EffectiveAnswers	308	0.84	0.85	0.81	0.76	4.4	0.83
Feedback	304	0.78	0.79	0.70	0.66	4.2	0.88
ClearOrganization	309	0.85	0.83	0.78	0.74	4.0	1.08
ClearPresentation	309	0.87	0.87	0.83	0.78	4.2	0.92

Non missing response frequency for each item

	1	2	3	4	5	miss
ClearResponsibilities	0.01	0.02	0.07	0.31	0.59	0.01
EffectiveAnswers	0.01	0.02	0.08	0.36	0.53	0.01
Feedback	0.01	0.05	0.10	0.39	0.46	0.02
ClearOrganization	0.04	0.07	0.10	0.41	0.38	0.00
ClearPresentation	0.02	0.05	0.07	0.40	0.46	0.00

Alpha for Traditional Pedagogy dimension is .90

```
psych::alpha(items[, SRPedVars])
```

Reliability analysis

Call: psych::alpha(x = items[, SRPedVars])

	raw_alpha	std.alpha	G6(smc)	average_r	S/N	ase	mean	sd	median_r
	0.81	0.81	0.78	0.52	4.3	0.017	4.5	0.58	0.54

95% confidence boundaries

lower alpha upper

Feldt 0.77 0.81 0.84

Duhachek 0.77 0.81 0.84

Reliability if an item is dropped:

	raw_alpha	std.alpha	G6(smc)	average_r	S/N	alpha	se	var.r	med.r
InclusvClassrm	0.74	0.74	0.67	0.49	2.9	0.025	0.0120		
EquitableEval	0.78	0.79	0.73	0.56	3.9	0.021	0.0034		
MultPerspectives	0.73	0.74	0.67	0.49	2.8	0.026	0.0153		
DEIIntegration	0.78	0.78	0.71	0.54	3.6	0.021	0.0044		
InclusvClassrm	0.50								
EquitableEval	0.57								
MultPerspectives	0.47								
DEIIntegration	0.57								

Item statistics

	n	raw.r	std.r	r.cor	r.drop	mean	sd
InclusvClassrm	301	0.82	0.83	0.76	0.67	4.6	0.68
EquitableEval	308	0.75	0.76	0.64	0.58	4.6	0.63
MultPerspectives	305	0.85	0.83	0.76	0.68	4.4	0.84
DEIIntegration	273	0.78	0.78	0.67	0.59	4.5	0.74

Non missing response frequency for each item

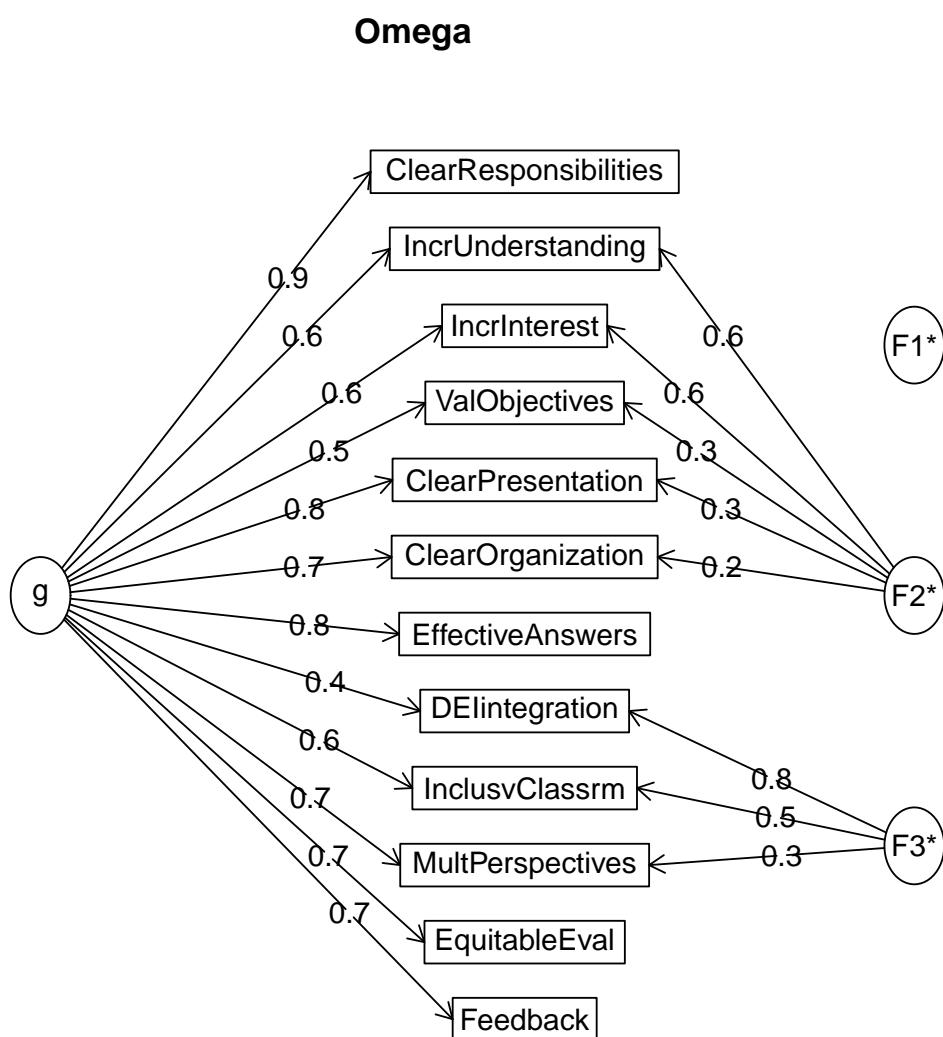
	1	2	3	4	5	miss
InclusvClassrm	0.01	0.01	0.05	0.23	0.70	0.03
EquitableEval	0.00	0.01	0.03	0.32	0.63	0.01
MultPerspectives	0.02	0.02	0.08	0.33	0.56	0.02
DEIIntegration	0.00	0.01	0.10	0.22	0.67	0.12

Alpha for the SCR Pedagogy dimension is .81

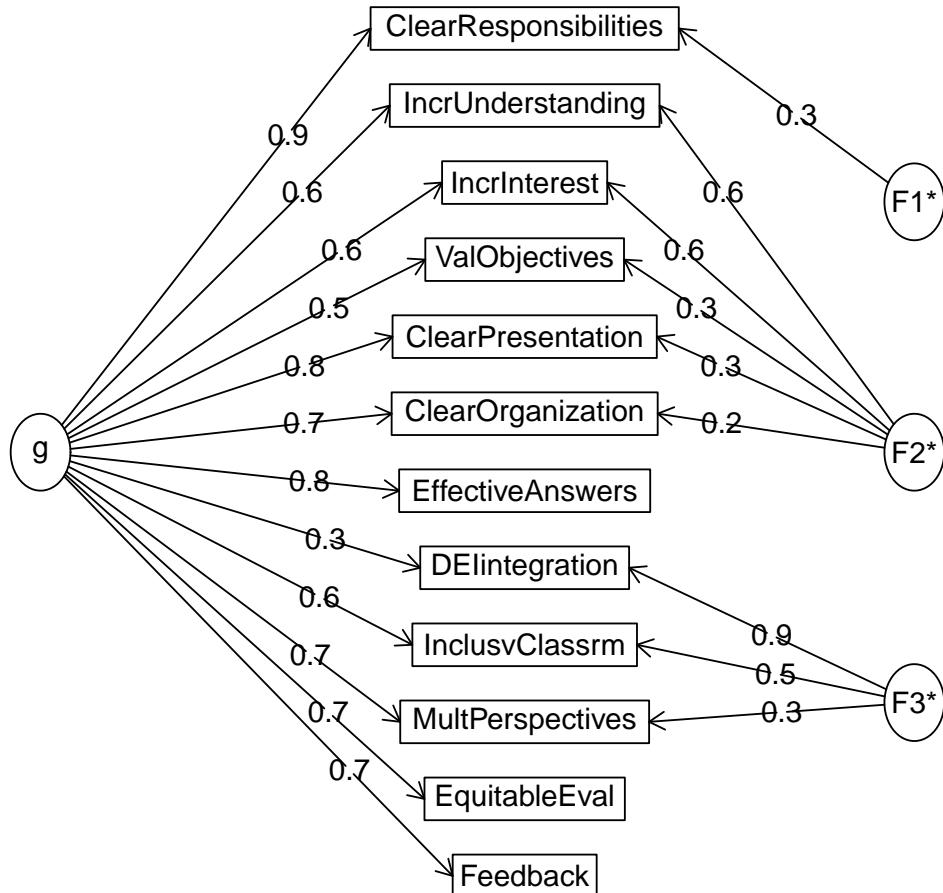
### 5.7.4 Calculate and report t and h

```
psych::omegaSem(items, nfactors = 3)
```

Warning in lav\_model\_vcov(lavmodel = lavmodel, lavsamplestats = lavsamplestats, : lavaan WARNING:  
 Could not compute standard errors! The information matrix could  
 not be inverted. This may be a symptom that the model is not  
 identified.



### Omega from SEM



```

Call: psych::omegaSem(m = items, nfactors = 3)
Omega
Call: omegah(m = m, nfactors = nfactors, fm = fm, key = key, flip = flip,
  digits = digits, title = title, sl = sl, labels = labels,
  plot = plot, n.obs = n.obs, rotate = rotate, Phi = Phi, option = option,
  covar = covar)
Alpha:          0.92
G.6:           0.93
Omega Hierarchical: 0.83
Omega H asymptotic: 0.88
Omega Total      0.94

```

	Schmid Leiman Factor loadings greater than 0.2						
	<i>g</i>	$F_{1*}$	$F_{2*}$	$F_{3*}$	<i>h</i> <sup>2</sup>	<i>u</i> <sup>2</sup>	<i>p</i> <sup>2</sup>
ValObjectives	0.47		0.32		0.33	0.67	0.67
IncrUnderstanding	0.57		0.60		0.69	0.31	0.47

IncrInterest	0.58	0.57	0.67	0.33	0.49
ClearResponsibilities	0.87		0.78	0.22	0.98
EffectiveAnswers	0.79		0.65	0.35	0.97
Feedback	0.73		0.56	0.44	0.96
ClearOrganization	0.75	0.23	0.62	0.38	0.90
ClearPresentation	0.81	0.30	0.74	0.26	0.88
MultPerspectives	0.75		0.30	0.65	0.35
InclusvClassrm	0.56		0.48	0.57	0.43
DEIIntegration	0.37		0.82	0.80	0.20
EquitableEval	0.70		0.52	0.48	0.94

With Sums of squares of:

	$g$	$F1^*$	$F2^*$	$F3^*$
5.51	0.04	0.98	1.06	

general/max 5.19 max/min = 26.42  
 mean percent general = 0.74 with sd = 0.26 and cv of 0.36  
 Explained Common Variance of the general factor = 0.73

The degrees of freedom are 33 and the fit is 0.25  
The number of observations was 310 with Chi Square = 76.65 with prob < 0.0000025  
The root mean square of the residuals is 0.02  
The df corrected root mean square of the residuals is 0.03  
RMSEA index = 0.065 and the 10 % confidence intervals are 0.046 0.085  
BIC = -112.66

RMSEA index = 0.147 and the 10 % confidence intervals are 0.134 0.16  
BIC = 105.22

## Measures of factor score adequacy

	g	F1*	F2*	F3*
Correlation of scores with factors	0.95	0.13	0.81	0.89
Multiple R square of scores with factors	0.91	0.02	0.66	0.79
Minimum correlation of factor score estimates	0.82	-0.96	0.32	0.58

Total, General and Subset omega for each subset

	g	F1*	F2*	F3*
Omega total for total scores and subscales	0.94	0.77	0.90	0.87
Omega general for total scores and subscales	0.83	0.76	0.69	0.64
Omega group for total scores and subscales	0.11	0.01	0.20	0.23

The following analyses were done using the `lavaan` package

Omega Hierarchical from a confirmatory model using sem = 0.82

Omega Total from a confirmatory model using sem = 0.94

With loadings of

	g	F1*	F2*	F3*	h2	u2	p2
ValObjectives	0.47		0.32		0.33	0.67	0.67
IncrUnderstanding	0.56		0.62		0.70	0.30	0.45
IncrInterest	0.57		0.56		0.64	0.36	0.51
ClearResponsibilities	0.86	0.32			0.84	0.16	0.88
EffectiveAnswers	0.80				0.65	0.35	0.98
Feedback	0.73				0.55	0.45	0.97
ClearOrganization	0.75		0.22		0.61	0.39	0.92
ClearPresentation	0.82		0.27		0.74	0.26	0.91
MultPerspectives	0.74			0.30	0.64	0.36	0.86
InclusvClassrm	0.56			0.49	0.56	0.44	0.56
DEIIntegration	0.33			0.87	0.87	0.13	0.13
EquitableEval	0.69				0.51	0.49	0.93

With sum of squared loadings of:

g	F1*	F2*	F3*
5.46	0.10	0.94	1.14

The degrees of freedom of the confirmatory model are 42 and the fit is 110.6184 with p = 0  
general/max 4.8 max/min = 10.87

mean percent general = 0.73 with sd = 0.27 and cv of 0.37

Explained Common Variance of the general factor = 0.71

Measures of factor score adequacy

	g	F1*	F2*	F3*
Correlation of scores with factors	0.95	0.57	0.82	0.95
Multiple R square of scores with factors	0.91	0.32	0.67	0.90
Minimum correlation of factor score estimates	0.82	-0.36	0.35	0.80

Total, General and Subset omega for each subset

	g	F1*	F2*	F3*
Omega total for total scores and subscales	0.94	0.84	0.89	0.87
Omega general for total scores and subscales	0.82	0.74	0.70	0.62
Omega group for total scores and subscales	0.11	0.10	0.20	0.26

To get the standard sem fit statistics, ask for summary on the fitted object

I'm reporting the values below the statement, "The following analyses were done using the lavaan package":

Omega total = .94 (omega total values > .80 are an indicator of good reliability). Interpretation: 94% of the variance in the total scale is due to the factors and the balance (6%) is due to error.

Omega hierarchical estimates the proportion of variance in the overall course evaluation score attributable to the general factors (thus treating the subscales as error). Omega h for the overall

course evaluation score was .82

**5.7.5 With these two determine what proportion of the variance is due to all the factors, error, and g.**

```
.82/.94
```

```
[1] 0.8723404
```

A quick calculation with omega h (.82) and omega total (.94) lets us know that 87% of the reliable variance in the overall course evaluation score is attributable to the general factor.

**5.7.6 Calculate total and subscale scores.**

This code uses the variable vectors I created above.

```
items$Valued <- sjstats::mean_n(items[, ValuedVars], 0.75)
items$TradPed <- sjstats::mean_n(items[, TradPedVars], 0.75)
items$SCRPed <- sjstats::mean_n(items[, SCRPedVars], 0.75)
items$Total <- sjstats::mean_n(items, 0.75)
```

```
scores <- items %>%
  dplyr::select(Valued, TradPed, SCRPed, Total)

psych::describe(scores)
```

	vars	n	mean	sd	median	trimmed	mad	min	max	range	skew	kurtosis
Valued	1	309	4.25	0.68	4.33	4.32	0.50	1.67	5	3.33	-0.90	0.57
TradPed	2	307	4.25	0.76	4.40	4.37	0.59	1.00	5	4.00	-1.42	2.48
SCRPed	3	299	4.52	0.58	4.75	4.61	0.37	2.25	5	2.75	-1.25	1.33
Total	4	308	4.34	0.60	4.41	4.41	0.62	1.83	5	3.17	-1.07	1.12
	se											
Valued		0.04										
TradPed		0.04										
SCRPed		0.03										
Total		0.03										

**5.7.7 Describe other reliability estimates that would be appropriate for the measure you are evaluating.**

These scales are for the purposes of course evaluations. In their development, it might be helpful to give it at the end of a single course and then again a few weeks later to determine test-retest reliability.

# Chapter 6

## Item Analysis for Educational Achievement Tests (Exams)

### [Screencasted Lecture Link](#)

In this lecture I walk through some procedures for analyzing the quality of multiple choice (including true/false) exam items. We look at item difficulty and item discrimination. We also look at item coverage as it relates to the learning objectives for an educational endeavor.

### 6.1 Navigating this Lesson

There is about one hour of lecture. If you work through the materials with me it would be plan for an additional 30 minutes.

While the majority of R objects and data you will need are created within the R script that sources the chapter, occasionally there are some that cannot be created from within the R framework. Additionally, sometimes links fail. All original materials are provided at the [GitHub site](#) that hosts the book. More detailed guidelines for ways to access all these materials are provided in the OER's [introduction](#)

#### 6.1.1 Learning Objectives

Focusing on this week's materials, make sure you can:

- Provide a rationale for why having a *test bank* might be a good idea.
- Describe the effects of skewness on the interpretation of exam results.
- Evaluate the quality of a multiple-choice item on the basis of item difficulty, correlation, and discrimination.
- Discuss the challenges of identifying an *ideal* difficulty level for test items. Further elaborate how guessing, speeded tests, interitem correlations, and the purposes of the test influence the *ideal difficulty*.

### 6.1.2 Planning for Practice

Practice suggestions for this lesson encourage you to think about the exams in your life: those you might be taking; those you might be writing or proctoring.

### 6.1.3 Readings & Resources

Classic psychometric texts tend to not cover item analysis for achievement tests and/or they skip over these fundamentals and move straight to item response theory/Rasch modeling (IRT). After scouring the internet, I landed on these two resources as concise, accessible, summaries.

- Understanding item analysis. Office of Educational Assessment, University of Washington. Retrieved September 20, 2019. Retrieved from <https://www.washington.edu/assessment/scanning-scoring/scoring/reports/item-analysis/>
  - It is common for excellent instructions/descriptions to accompany the scoring software used by institutions. UW appears to use ScorePak, and this resource provides both conceptual and interpretive information.
- Revelle, W. (2017). An overview of the psych package. Retrieved from <http://personality-project.org/r/overview.pdf>
  - Pages 85-85 provide a vignette for conducting item analysis on multiple choice items.

### 6.1.4 Packages

The packages used in this lesson are embedded in this code. When the hashtags are removed, the script below will (a) check to see if the following packages are installed on your computer and, if not (b) install them.

```
# will install the package if not already installed
# if(!require(psych)){install.packages('psych')}
```

## 6.2 Research Vignette

This lesson's research vignette is from my own class. Especially in the early years of my teaching, I gave high(er) stakes mid-term and final exams. There were usually 40 (or so) multiple choice or true/false items, 2-3 applied problems or short essays, and 1 longer essay. Today's vignette is an array of exam items from a statistics exam that demonstrate the desirable and undesirable elements we want in objective items.

## 6.3 Item Analysis in the Educational/Achievement Context

Multiple choice, true/false, and other *objectively* formatted/scored items are part-n-parcel to educational/achievement assessment. But how do we know if the items are performing the way they

should? This lecture focuses on item analysis in the context of multiple choice and true/false items. Using these practices can help you identify what selection of items you'd like for your exams. These can be critical tools in helping you improve your ability to assess student performance. In-so-doing, we walk through a bit of "what we used to do," to current common practices, to a glimpse of our future. We owe much of this to rapid advances in technology.

*Test banks* are instructor-created resources for developing/storing/protecting items for use in future exams. We create test banks when we carefully distribute/collect/protect items "that work" (from statistical perspective). Why would we want to do this?

- Once a test is "out" it's out. Instructors can presume that resourceful students are using it to study; yet all students won't have equal access to it.
- Developing "good" items takes a good deal of time; does the instructor want to redo this each term?
- Should we be piloting *all new items* on students each term and then having the debates about whether the item should be rescored?
  - Better is to introduce a proportion of new items each year and evaluate them for inclusion in the test bank; EPPP, SAT, GRE do this.
- A challenge is providing students appropriate study tools – old exams are favorites of students (but maybe there are other ways – worksheets, Jeopardy).

The conceptual portions of this lecture, particularly the interpretation of the difficulty and discrimination statistics are based in Anastasi's work [[Anastasi and Urbina, 1997](#)]

### 6.3.1 And now a quiz! Please take it.

Let's start with some items from an early version of the exam I gave when I taught CPY7020/Statistical Methods.

**Item 5** A grouping variable such as men or women that uses dummy coding of 1 and 0 to categorize the groups is an example of \_\_\_\_\_ scaling.

- a) Nominal
- b) Ordinal
- c) Interval
- d) Ratio

**Item 11** The term "grade inflation" has frequently been applied to describe the distribution of grades in graduate school. Which of the following best describes this distribution.

- a) negatively skewed
- b) uniform/rectangular
- c) positively skewed and leptokurtic

- d) uniform and platykurtic

**Item 19** All distributions of Z-scores will have the identical

- a) Mean
- b) Variance
- c) Standard deviation
- d) All of the above

**Item 21** The most appropriate score for comparing scores across two or more distributions (e.g., exam scores in math and art classes) is the:

- a) mean
- b) percentile rank
- c) raw score
- d) z-score

**Item 37** Of the following, what statement best describes  $r^2 = .49$

- a) strong positive correlation
- b) strong positive or negative correlation
- c) weak positive or negative correlation
- d) weak negative correlation

**Item 38** When there are no ties among ranks, what is the relationship between the Spearman rho ( $\rho$ ) and the Pearson ( $r$ )?

- a)  $\rho = r$
- ii)  $\rho > r$
- a)  $\rho < r$
- b) no relationship

## 6.4 Item Difficulty

### 6.4.1 Percent passing

**Item difficulty index** is the proportion of test takers who answer an item correctly. It is calculated by dividing the number of people who passed the item (e.g., 55) by the total number of people (e.g., 100).

- If 55% pass an item, we write  $p = .55$
- The easier the item, the larger the percentage will be.

What is an ideal pass rate (and this “ideal” is the *statistical ideal* mostly for norm-referenced tests like the ACT, SAT, GRE)?

- The closer the difficulty of an item approaches 1.00 or 0, the less differential information about test takers it contributes.
  - If, out of 100 people, 50 pass an item and 50 fail ( $p = .50$ )...we have 50 X 50 or 2,500 paired comparisons or differential bits of information.
- How much information would we have for an item passed by:
  - 70% of the people ( $70 * 30 = ???$ )
  - 90% of the people ( $90 * 10 = ???$ )
- For maximum differentiation, one would choose all items at the .50 level (but hold up...)

### 6.4.2 Several factors prevent .50 from being the ideal difficulty level

**Speeded tests** complicate the interpretation of item difficulty because items are usually of equivalent difficulty and there are so many that no one could complete them all. Thus, later items should be considered to be more difficult – but item difficulty is probably not the best assessment of item/scale quality.

**Guessing** the correct answer in true/false and multiple-choice contexts interfere with the goal of  $p = .50$ . In a 1952 issue of *Psychometrika*, Lord provided this guide for optimal  $p$  values based on the number of choices in the objective context:

Optimal  $p$  values

Number of Choices	Optimal Mean Difficulty Level
2 (T/F)	0.85
3	0.77
4	0.74
5	0.69
Constructed response essay	0.5

The purpose of the testing changes the ideal difficulty level.

- If the test is *norm-referenced* (ACT, SAT, GRE), .50 is very useful.
- If the test is mastery oriented,  $p$  values may be as high as 0.90 since student performance is a function of repeated attempts with feedback.

**Item intercorrelations** impacts interpretation of item difficulty.

- The more homogeneous the test, the higher these intercorrelations will be. If all items were perfectly intercorrelated and all were of the .50 difficulty level:
  - the same 50 persons out of 100 would pass each item, that is,
  - half of the test takers would obtain perfect scores, the other half zero scores
- It is best to select items with a moderate spread of difficulty but whose AVERAGE difficulty level is .50
- The percentage of persons passing an item expresses the item difficulty in terms of which statistical scale of measurement? Is it nominal, ordinal, interval, or ratio?
  - Because of this issue, we can correctly indicate the rank order or relative difficulty of the items
  - However, we cannot infer that the difference in difficulty between Items 1 and 2 is equal to the difference between Items 2 and 3.
- We can make an *equal-interval inference* with the table of normal curve frequencies (i.e., translating the proportion to z-scores). Z-scores would be used as the units if an equal interval inference was required in the analysis. For example,
  - $p = .84$  is equal to  $-1\text{ SD}$
  - $p = .16$  is equal to  $+1\text{ SD}$

*Seem a little upside down? Recall that we are calculating the percent passing and starting the count “from the top.” So a relatively easy item where 84% passed, would have an standard deviation of -1.*

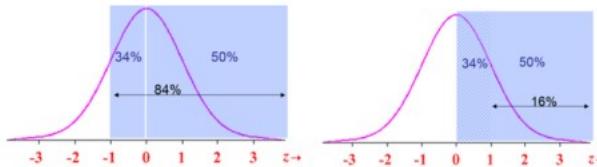


Figure 6.1: Image of graphs where  $p = .84$  and  $p = .16$

## 6.5 Item Discrimination

The degree to which an item differentiates correctly among test takers in the behavior that the test is designed to measure.

- The *criterion* can be internal or external to the test itself.

- Under some conditions, the two approaches lead to opposite results because (a) items chosen to maximize the validity of the test tend to be the ones rejected on the basis of internal consistency, and (b) rejecting items with low correlations with the total score tends to homogenize the test (we are more likely to keep items with the highest average intercorrelations).
- *Internal* criteria maximize internal consistency or homogeneity of the test.
  - Example: achievement test, where criteria is total score itself.
- *External* criteria maximize the validity of an external criterion.
  - Example: a different assessment of the same ability being assessed.

### 6.5.1 Index of Discrimination

- Compare the proportion of cases that pass an item in contrasting criterion groups
  - upper (U) and lower (L) criterion groups are selected from the extremes of the distribution
  - traditionally these groups are created from the 27% from each of those sides of the distribution
- This *index of discrimination (D)* can be expressed as a difference of raw frequencies (U - L), or (more conventionally) as the difference of percentages of those who scored it correctly in the upper 27% and lower 27% groups
  - when all members of the U group and none of the members of the L group pass, D = 100
  - when all members of the L group and none of the members of the U group pass, D = 0
  - optimum point at which these two conditions reach balance is with the upper and lower 27%

---

#### Optimal Discrimination

---

Difficulty	Discrimination
0.40 and larger	Excellent
0.30 - 0.39	Good
0.11 - 0.29	Fair
0.00 -0.10	Poor
Negative values	Mis-keyed or other major flaw

### 6.5.2 Application of Item Difficulty and Discrimination

Earlier I asked you to “take the quiz.” To keep it engaging, I encourage you to look at your own answers and compare them to “what happened” from in this actual exam administration. I will

KEY		A	A	D	D	B	A		
	ID	Missed	Score	Item5	Item11	Item19	Item21	Item37	Item38
	1	3	38	A	A	D	B	B	A
	10	3	38	A	C	D	D	B	D
	11	3	38	A	A	D	D	C	D
	8	4	37	A	A	D	D	C	D
	5	5	36	A	A	D	B	C	
	2	6	35	A	A	D	D	B	D
	7	7	34	A	D	D	B	B	D
	6	8	33	A	C	A	B	A	D
	9	10	31	A	C	D	B	C	A
	4	12	29	A	A	D	D	C	D
	12	12	29	A	C	D	D	A	A
	3	16	25	A	D	D	D	A	D

Figure 6.2: Image of scores and responses of 6 items from 12 students.

demonstrate how to evaluate my exam items with these indices of difficulty and discrimination. I have intentionally selected items with a variety of desirable (and undesirable) characteristics.

**Item 5** A grouping variable such as men or women that uses dummy coding of 1 and 0 to categorize the groups is an example of \_\_\_\_\_ scaling.

- a) Nominal
- b) Ordinal
- c) Interval
- d) Ratio

If we wanted to hand-calculate the index of discrimination for Item #5, we find that 3 people (100%) in the upper group selected the correct answer and 3 people (100%) in the lower group selected the correct answer:  $3 - 3 = 0$ . If you prefer percentages:  $100\% - 100\% = 0\%$ . This means there is no discrimination in performance of the upper and lower performing groupings.

Older scoring systems (e.g., Scantron) used to provide this information.

Considering what we have learned already, Item #5 is:

- too easy
- does not discriminate between upper and lower performance
- *Yes, there is more data on here, but we will save it for the next level of review...just a few moments.*

**Item 11** The term “grade inflation” has frequently been applied to describe the distribution of grades in graduate school. Which of the following best describes this distribution.

- a) negatively skewed
- b) uniform/rectangular

<b>Item 5</b>	Answer: A Point Biserial: 0.00	Correct Responses: Total Group: Upper 27% of Group: Lower 27% of Group:
Distractor Analysis - * indicates correct answer:		
	*A      B      C      D      E	
Pt-Biser.:	0.00      0.00      0.00      0.00      0.00	
Frequency%:	100.00      0.00      0.00      0.00      0.00	

Figure 6.3: Scantron image of item analysis for exam item #5

- c) positively skewed and leptokurtic
- d) uniform and platykurtic

For Item #11, 2 people (~66%) from the upper group selected the correct answer, 1 person (~33%) from the lower group selected the correct answer. Thus, the U-L was +1 (+33%) and the item is working in the proper direction.

<b>Item 11</b>	Answer: A Point Biserial: 0.44	Correct Responses: Total Group: 50.00% Upper 27% of Group: 66.67% Lower 27% of Group: 33.33%
Distractor Analysis - * indicates correct answer:		
	*A      B      C      D      E	
Pt-Biser.:	0.44      0.00      -0.15      -0.39      0.00	
Frequency%:	50.00      0.00      33.33      16.67      0.00	

Figure 6.4: Scantron image of item analysis for exam item #11

Considering what we have learned already, Item #11 is:

- difficult (50% overall selected the correct item)
- does discriminate between upper and lower performance, with more individuals in the upper groups selecting the correct answer than in the lower group

**Item 19** All distributions of Z-scores will have the identical

- a) Mean
- b) Variance
- c) Standard deviation
- d) All of the above

<b>Item 19</b>	Answer: D Point Biserial: -0.06	Correct Responses: Total Group: 91.87% Upper 27% of Group: 100.00% Lower 27% of Group: 100.00%
Distractor Analysis - * indicates correct answer:		
A	B	C
Pt-Biser.: 0.06	0.00	0.00
Frequency%: 8.33	0.00	0.00
D		
E		
Pt-Biser.: -0.06		
Frequency%: 0.00	91.87	0.00
** Review item(s): A		

Figure 6.5: Scantron image of item analysis for exam item #19

Hand calculation: Upper = 3 (100%), Lower = 3 (100%). Difference = 0.

Considering what we have learned already, Item #19 is:

- somewhat easy (92% overall selected the correct item)
- using the U - L discrimination index, it does not discriminate between upper and lower performance

**Item 21** The most appropriate score for comparing scores across two or more distributions (e.g., exam scores in math and art classes) is the:

- a) mean
- b) percentile rank
- c) raw score
- d) z-score

Hand calculation: Upper = 2 (66%), Lower = 3 (100%). Difference = -33%. This item is upside down. This is different than the Scantron snip below because uppers and lowers were likely calculated on exam total that included subjectively scored items (essays; and I no longer have that data).

<b>Item 21</b>	Answer: D Point Biserial: -0.13	Correct Responses: Total Group: 58.33% Upper 27% of Group: 66.67% Lower 27% of Group: 66.67%
Distractor Analysis - * indicates correct answer:		
A	B	C
0.00	0.13	0.00
-0.13	0.00	
D		
E		
Pt-Biser.: 0.00	41.67	0.00
Frequency%: 0.00	58.33	0.00
** Review item(s): B		

Figure 6.6: Scantron image of item analysis for exam item #21

Considering what we have learned already, Item #21 is:

- somewhat difficult (58% overall selected the correct item)
- on the basis of the hand-calculations it does not discriminate between uppers and lowers

**Item 37** Of the following, what statement best describes  $r^2 = .49$

- a) strong positive correlation

- b) strong positive or negative correlation
- c) weak positive or negative correlation
- d) weak negative correlation

Hand calculation: Upper = 2 (66%), Lower = 0 (0%). Difference = 66%.

<b>Item 37</b>	Answer: B Point Biserial: 0.48	Correct Responses: Total Group: 33.33% Upper 27% of Group: 66.67% Lower 27% of Group: 0.00%		
Distractor Analysis - * indicates correct answer:				
A	*B	C	D	E
Pt-Biser.: -0.52	0.48	0.00	0.00	0.00

Figure 6.7: Scantron image of item analysis for exam item #37

Considering what we have learned already, Item #37 is:

- very difficult (33% overall selected the correct item)
- on the basis of the hand-calculations, this completely discriminates the uppers from the lowers)

**Item 38** When there are no ties among ranks, what is the relationship between the Spearman rho ( $\rho$ ) and the Pearson r ( $r$ )?

- a)  $\rho = r$
- ii)  $\rho > r$
- a)  $\rho < r$
- b) no relationship

Hand calculation: Upper = 1 (33%), Lower = 1 (33%). Difference = 0%.

<b>Item 38</b>	Answer: A Point Biserial: -0.21	Correct Responses: Total Group: 25.00% Upper 27% of Group: 33.33% Lower 27% of Group: 33.33%		
Distractor Analysis - * indicates correct answer:				
'A	B	C	D	E
Pt-Biser.: -0.19	0.00	0.00	0.19	0.00

Figure 6.8: Scantron image of item analysis for exam item #38

Considering what we have learned already, Item #21 is:

- very difficult (25% overall selected the correct item)
- on the basis of the hand-calculations, this does not discriminate the uppers from the lowers

## 6.6 In the *psych* Package

Using the *score.multiple.choice()* function in the *psych* package. Documentation is pp. 85-86 in <http://personality-project.org/r/overview.pdf>

A multiple-choice exam presumes that there is one correct response. We start with a dataset that records the students' responses. It *appears* that the psych package requires these responses to be numerical (rather than A, B, C, D).

```
# For portability of the lesson, I hand-entered the exam score data.
# Variables are items (not students), so the entry is the 41 items
# for the 12 students
Item1 <- c(1, 1, 4, 1, 1, 1, 1, 1, 1, 1, 1)
Item2 <- c(4, 4, 1, 4, 4, 4, 4, 4, 4, 4, 4)
Item3 <- c(1, 1, 4, 1, 1, 1, 1, 3, 1, 1, 1)
Item4 <- c(2, 3, 2, 2, 2, 2, 2, 2, 2, 2, 2)
Item5 <- c(1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1)
Item6 <- c(1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1)
Item7 <- c(3, 3, 4, 4, 3, 3, 3, 3, 3, 3, 2)
Item8 <- c(1, 2, 2, 4, 2, 2, 2, 1, 4, 2, 2)
Item9 <- c(1, 1, 4, 4, 1, 4, 1, 1, 1, 1, 4)
Item10 <- c(3, 3, 3, 2, 3, 2, 3, 2, 2, 3, 3)
Item11 <- c(1, 1, 4, 1, 1, 3, 4, 1, 3, 3, 1, 3)
Item12 <- c(2, 1, 2, 4, 2, 2, 2, 2, 2, 2, 2)
Item13 <- c(2, 2, 3, 3, 2, 2, 2, 2, 2, 2, 1)
Item14 <- c(2, 2, 2, 2, 2, 3, 2, 2, 2, 2, 2)
Item15 <- c(2, 1, 1, 3, 2, 4, 2, 2, 2, 2, 4, 2)
Item16 <- c(2, 2, 2, 4, 4, 2, 2, 2, 4, 2, 2, 1)
Item17 <- c(1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1)
Item18 <- c(3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3)
Item19 <- c(4, 4, 4, 4, 1, 4, 4, 4, 4, 4, 4)
Item20 <- c(1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1)
Item21 <- c(2, 4, 4, 4, 2, 2, 2, 4, 2, 4, 4, 4)
Item22 <- c(3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 1)
Item23 <- c(3, 3, 2, 3, 3, 3, 3, 2, 3, 3, 2)
Item24 <- c(3, 3, 1, 3, 3, 2, 3, 3, 3, 3, 1)
Item25 <- c(2, 2, 2, 3, 2, 2, 2, 2, 2, 2, 2)
Item26 <- c(4, 4, 4, 4, 4, 4, 4, 1, 4, 4, 1)
Item27 <- c(4, 4, 1, 4, 4, 4, 4, 4, 4, 4, 4)
Item28 <- c(1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1)
Item29 <- c(1, 3, 1, 1, 1, 1, 1, 1, 1, 1, 1)
Item30 <- c(2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2)
Item31 <- c(1, 1, 1, 2, 1, 1, 3, 1, 1, 1, 1, 2)
Item32 <- c(1, 1, 3, 1, 1, 1, 3, 1, 1, 1, 1, 1)
Item33 <- c(3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3)
Item34 <- c(3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3)
Item35 <- c(3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3)
Item36 <- c(2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2)
```

```

Item37 <- c(2, 2, 1, 3, 3, 1, 2, 3, 3, 2, 3, 1)
Item38 <- c(1, 4, 4, 4, NA, 4, 4, 4, 1, 4, 4, 1)
Item39 <- c(3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3)
Item40 <- c(3, 3, 4, 3, 3, 3, 3, 3, 3, 3, 3, 3)
Item41 <- c(2, 1, 2, 2, 4, 4, 2, 2, 4, 4, 2)

exam <- data.frame(Item1, Item2, Item3, Item4, Item5, Item6, Item7, Item8,
  Item9, Item10, Item11, Item12, Item13, Item14, Item15, Item16, Item17,
  Item18, Item19, Item20, Item21, Item22, Item23, Item24, Item25, Item26,
  Item27, Item28, Item29, Item30, Item31, Item32, Item33, Item34, Item35,
  Item36, Item37, Item38, Item39, Item40, Item41)

```

The optional script below will let you save the simulated data to your computing environment as either a .csv file (think “Excel lite”) or .rds object (preserves any formatting you might do).

```

# write the simulated data as a .csv write.table(exam,
# file='exam.csv', sep=',', col.names=TRUE, row.names=FALSE) bring
# back the simulated dat from a .csv file exam <- read.csv
# ('exam.csv', header = TRUE)

```

```

# to save the df as an .rds (think 'R object') file on your computer;
# it should save in the same file as the .rmd file you are working
# with saverDS(exam, 'exam.rds') bring back the simulated dat from an
# .rds file exam <- readRDS('exam.rds')

```

We create a key of the correct answers.

```
exam.keys <- c(1,4,1,2,1,1,3,2,1,3,1,2,2,2,2,2,1,3,4,1,4,3,3,3,1,4,4,1,1,2,1,1,3,3,3,2,2,1,3,
```

We then insert that key into the *psych* package’s *score.multiple.choice()* function.

```
results <- psych::score.multiple.choice(exam.keys, exam, score = TRUE,
  short = FALSE, skew = TRUE)
```

Warning in cor(items, scores, use = "pairwise"): the standard deviation is zero

```
results
```

Call: NULL

```
(Unstandardized) Alpha:
[1] 0.73
```

```
Average item correlation:
[1] 0.06
```

## item statistics

	key	1	2	3	4	miss	r	n	mean	sd	skew	kurtosis	se
Item1	1	0.92	0.00	0.00	0.08	0.00	0.65	12	0.92	0.29	-2.65	5.48	0.08
Item2	4	0.08	0.00	0.00	0.92	0.00	0.65	12	0.92	0.29	-2.65	5.48	0.08
Item3	1	0.83	0.00	0.08	0.08	0.00	0.34	12	0.83	0.39	-1.57	0.53	0.11
Item4	2	0.00	0.92	0.08	0.00	0.00	-0.11	12	0.92	0.29	-2.65	5.48	0.08
Item5	1	1.00	0.00	0.00	0.00	0.00	NA	12	1.00	0.00	NaN	NaN	0.00
Item6	1	1.00	0.00	0.00	0.00	0.00	NA	12	1.00	0.00	NaN	NaN	0.00
Item7	3	0.00	0.08	0.67	0.25	0.00	0.72	12	0.67	0.49	-0.62	-1.74	0.14
Item8	2	0.17	0.67	0.00	0.17	0.00	-0.13	12	0.67	0.49	-0.62	-1.74	0.14
Item9	1	0.67	0.00	0.00	0.33	0.00	0.81	12	0.67	0.49	-0.62	-1.74	0.14
Item10	3	0.00	0.33	0.67	0.00	0.00	0.18	12	0.67	0.49	-0.62	-1.74	0.14
Item11	1	0.50	0.00	0.33	0.17	0.00	0.42	12	0.50	0.52	0.00	-2.16	0.15
Item12	2	0.08	0.83	0.00	0.08	0.00	0.17	12	0.83	0.39	-1.57	0.53	0.11
Item13	2	0.08	0.75	0.17	0.00	0.00	0.85	12	0.75	0.45	-1.01	-1.04	0.13
Item14	2	0.00	0.92	0.08	0.00	0.00	-0.04	12	0.92	0.29	-2.65	5.48	0.08
Item15	2	0.17	0.58	0.08	0.17	0.00	0.32	12	0.58	0.51	-0.30	-2.06	0.15
Item16	2	0.08	0.67	0.00	0.25	0.00	0.40	12	0.67	0.49	-0.62	-1.74	0.14
Item17	1	1.00	0.00	0.00	0.00	0.00	NA	12	1.00	0.00	NaN	NaN	0.00
Item18	3	0.00	0.00	1.00	0.00	0.00	NA	12	1.00	0.00	NaN	NaN	0.00
Item19	4	0.08	0.00	0.00	0.92	0.00	0.04	12	0.92	0.29	-2.65	5.48	0.08
Item20	1	1.00	0.00	0.00	0.00	0.00	NA	12	1.00	0.00	NaN	NaN	0.00
Item21	4	0.00	0.42	0.00	0.58	0.00	-0.19	12	0.58	0.51	-0.30	-2.06	0.15
Item22	3	0.08	0.00	0.92	0.00	0.00	0.34	12	0.92	0.29	-2.65	5.48	0.08
Item23	3	0.00	0.25	0.75	0.00	0.00	0.71	12	0.75	0.45	-1.01	-1.04	0.13
Item24	3	0.17	0.08	0.75	0.00	0.00	0.61	12	0.75	0.45	-1.01	-1.04	0.13
Item25	1	0.00	0.92	0.08	0.00	0.00	NA	12	0.00	0.00	NaN	NaN	0.00
Item26	4	0.17	0.00	0.00	0.83	0.00	0.34	12	0.83	0.39	-1.57	0.53	0.11
Item27	4	0.08	0.00	0.00	0.92	0.00	0.65	12	0.92	0.29	-2.65	5.48	0.08
Item28	1	1.00	0.00	0.00	0.00	0.00	NA	12	1.00	0.00	NaN	NaN	0.00
Item29	1	0.92	0.00	0.08	0.00	0.00	-0.11	12	0.92	0.29	-2.65	5.48	0.08
Item30	2	0.00	1.00	0.00	0.00	0.00	NA	12	1.00	0.00	NaN	NaN	0.00
Item31	1	0.75	0.17	0.08	0.00	0.00	0.41	12	0.75	0.45	-1.01	-1.04	0.13
Item32	1	0.83	0.00	0.17	0.00	0.00	0.45	12	0.83	0.39	-1.57	0.53	0.11
Item33	3	0.00	0.00	1.00	0.00	0.00	NA	12	1.00	0.00	NaN	NaN	0.00
Item34	3	0.00	0.00	1.00	0.00	0.00	NA	12	1.00	0.00	NaN	NaN	0.00
Item35	3	0.00	0.00	1.00	0.00	0.00	NA	12	1.00	0.00	NaN	NaN	0.00
Item36	2	0.00	1.00	0.00	0.00	0.00	NA	12	1.00	0.00	NaN	NaN	0.00
Item37	2	0.25	0.33	0.42	0.00	0.00	0.49	12	0.33	0.49	0.62	-1.74	0.14
Item38	1	0.27	0.00	0.00	0.73	0.08	-0.07	11	0.27	0.47	0.88	-1.31	0.14
Item39	3	0.00	0.00	1.00	0.00	0.00	NA	12	1.00	0.00	NaN	NaN	0.00
Item40	3	0.00	0.00	0.92	0.08	0.00	0.65	12	0.92	0.29	-2.65	5.48	0.08
Item41	4	0.08	0.58	0.00	0.33	0.00	0.40	12	0.33	0.49	0.62	-1.74	0.14

```
# short=FALSE allows us to produce scores; we will use these later in
# some IRT analyses names(results)
```

The first screen of output provides an alpha. In this context, *alpha* should tell us the consistency of getting answers right or wrong. Technically, the alpha is reduced to a KR-20 (Kuder Richardson 20). We interpret it the same. Alpha is directly affected by:

- *interitem correlations* among the items – a large number of positive correlations between items increases alpha
- test length – more items produce higher reliability (all things else equal)
- test content – the more diverse/broad, the lower the reliability coefficient

In the context of the classroom, reliabilities above .70 are probably adequate and above .80 are good. Reliabilities below .60 suggest that items should be investigated, and additional measures (tests, homework assignments) should be included in assigning grades.

Focus instead on the second screen of output.

**key** indicates which answer was correct.

**1, 2, 3, 4** (there would be as many as there are options in the multiple-choice exam) provide a *distractor analysis* by indicating the percentage of time that answer was chosen. For item 1, option 1 was correct, and it was chosen 92% of the time. No individuals chose options 2 or 3. Option 4 was chosen 8% of the time.

**miss** indicates how many times the item was skipped.

*r* is a point-biserial correlation with a dichotomous correct/incorrect correlated with the continuously scaled total scale score. Positively scored items let us know that the item is working in the proper direction; the students who got the item correct, did better on the overall total score and vice versa.

- One of the best indicators of an item's ability to *discriminate* (hence, **item discrimination**) among the criterion assessed on the test.
- It is important to investigate those with values close to zero (no relation between item performance with overall test performance) and those with negative values (meaning that those who had the correct answer on the item were those who scored lower on the exam).

*n* tells us how many participants completed the item (this would necessarily be the inverse of “miss”).

*mean* repeats the proportion of individuals who scored correctly; it would be the same as the percentage in the item keyed as the correct one. This is an indication of **item difficulty**.

*sd* gives an indication of the variability around that mean

It is important to look at the *r* and *mean* columns, together to understand the degree of difficulty and how well each item is discriminating between performance levels.

*skew* can provide an indication of ceiling and floor effects.

If a score has a significant negative skew (long tail to the left), then there may be a piling up of items at the upper end of the scale. This would indicate an *insufficient ceiling* and make it more difficult to discriminate among differences among the higher performers.

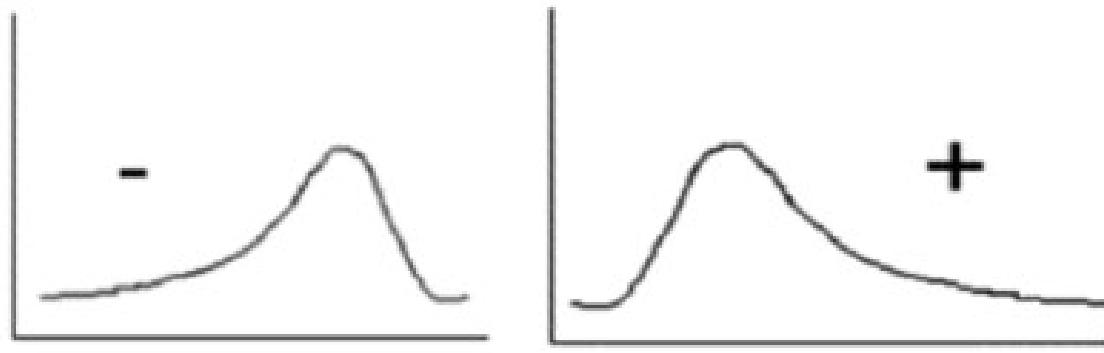


Figure 6.9: Image of two graphs illustrating positive and negative skew

If a score has a significant positive skew (long tail to the right), then there may be a piling up of items at the low end, indicating an *insufficient floor*. That is, it lacks the ability to discriminate between poorer performers.

How do you tell what is significant?

A general rule of thumb says that anything greater or less than the absolute value of 1.0 is significantly skewed. A formal z-test can be conducted this way:  $z_{skewness} = \frac{S-0}{SE_{skewness}}$

In our exam dataset, -2.65 is the most extremely negatively skewed item and its  $se = 0.08$ .

```
-2.65/0.08
```

```
[1] -33.125
```

Considering that anything greater than  $\pm 1.96$  is statistically significant, it is safe to say that this item has an insufficient ceiling.

What about the items with -0.30 ( $se = 0.15$ )?

```
-.30/.15
```

```
[1] -2
```

This is not as extreme (and recall my  $N = 12$ , so I should probably look up a critical  $t$  value), but there is still some question about whether my exam items can discriminate among high performers.

Please note, because these distributions are *dichotomous* (correct/incorrect) they will never be normally distributed, but, like the difficulty index, they give another glimpse of the ability to discriminate.

Before we look at the specific exam items and their output from the scoring function, let me introduce you to the features of the psych package that draw from *item response theory* (IRT).

### 6.6.1 A Mini-Introduction to IRT

To recap – at the instructional level, the combination of percent passing (mean) and point-biserial correlation (discrimination index) is status quo for evaluating/improving the items.

The *psych* package draws from its IRT capacity to conduct distractor analysis. IRT models individual responses to items by estimating individual ability (*theta*) and item difficulty (*diff*) parameters.

In these graphs, theta is on the X axis. Theta is the standard unit of the IRT model that represents the level of the domain being measured. Like a z-score, a theta unit of “1” is the SD of the calibrated sample.

The pattern of responses to multiple choice ability items can show that some items have poor distractors. This may be done by using the *irt.responses* function. A good distractor is one that is negatively related to ability.

As we look at each of the exam items, we will look at the *psych* input from the scoring function as well as use the *results* objects to create the IRT graphs.

**Item 5** A grouping variable such as men or women that uses dummy coding of 1 and 0 to categorize the groups is an example of \_\_\_\_\_ scaling.

- a) Nominal
- b) Ordinal
- c) Interval
- d) Ratio

Mean = 1.0 (much too easy),  $r = \text{NA}$ , Distractors: 1.00 0.00 0.00 0.00, skew = -2.65

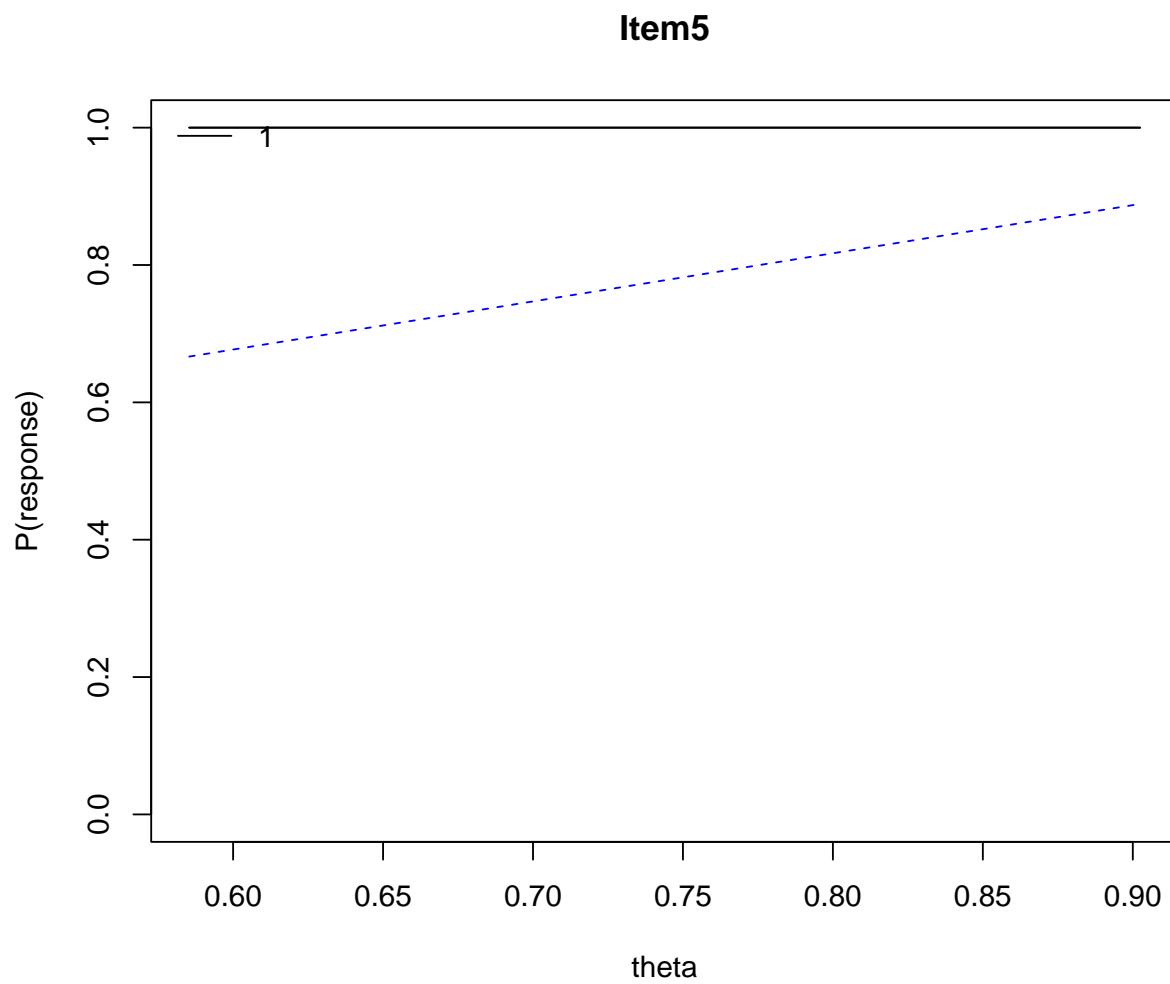
```
# irt.responses(scores$scores, exam[5], breaks = 2)
psych::irt.responses(results$scores, exam[5], breaks = 2)
```

Number of categories should be increased in order to count frequencies.

Warning in rbind(items, dummy): number of columns of result is not a multiple of vector length (arg 1)

Number of categories should be increased in order to count frequencies.

Warning in rbind(items, dummy): number of columns of result is not a multiple of vector length (arg 1)



With Item #5, 100% responded correctly (the flat, solid line at the top); there is not much to see.

**Item 11** The term “grade inflation” has frequently been applied to describe the distribution of grades in graduate school. Which of the following best describes this distribution.

- a) negatively skewed
- b) uniform/rectangular
- c) positively skewed and leptokurtic
- d) uniform and platykurtic

Mean = .50,  $r = .42$ , Distractors: 0.50 0.00 0.33 0.17, skew = 0.00

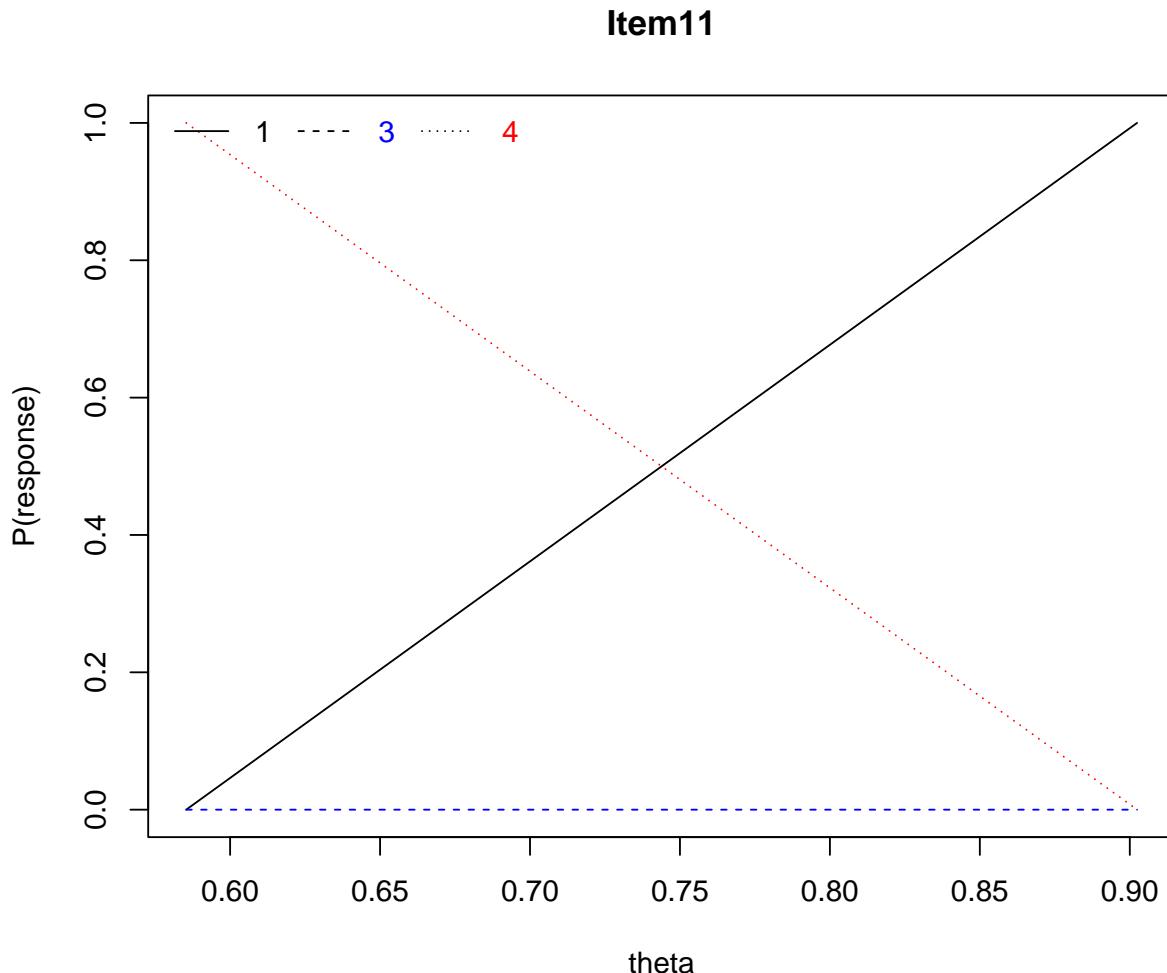
```
psych::irt.responses(results$scores, exam[11], breaks = 2)
```

Number of categories should be increased in order to count frequencies.

Warning in rbind(items, dummy): number of columns of result is not a multiple of vector length (arg 1)

Number of categories should be increased in order to count frequencies.

Warning in rbind(items, dummy): number of columns of result is not a multiple of vector length (arg 1)



With Item #11, there is a positive relationship between 1/A (correct answer) and ability (theta), no relationship between 3/C and ability, and a negative relationship between 4/D and ability (indicating that 4/D is a good distractor). These map onto each of the point-biserial correlations associated with the distractors in the Scantron output.

**Item 19** All distributions of Z-scores will have the identical

- a) Mean
- b) Variance

- c) Standard deviation
- d) All of the above

Mean = .92,  $r = .04$ , Distractors: 0.08 0.00 0.00 0.92 , skew = -2.65

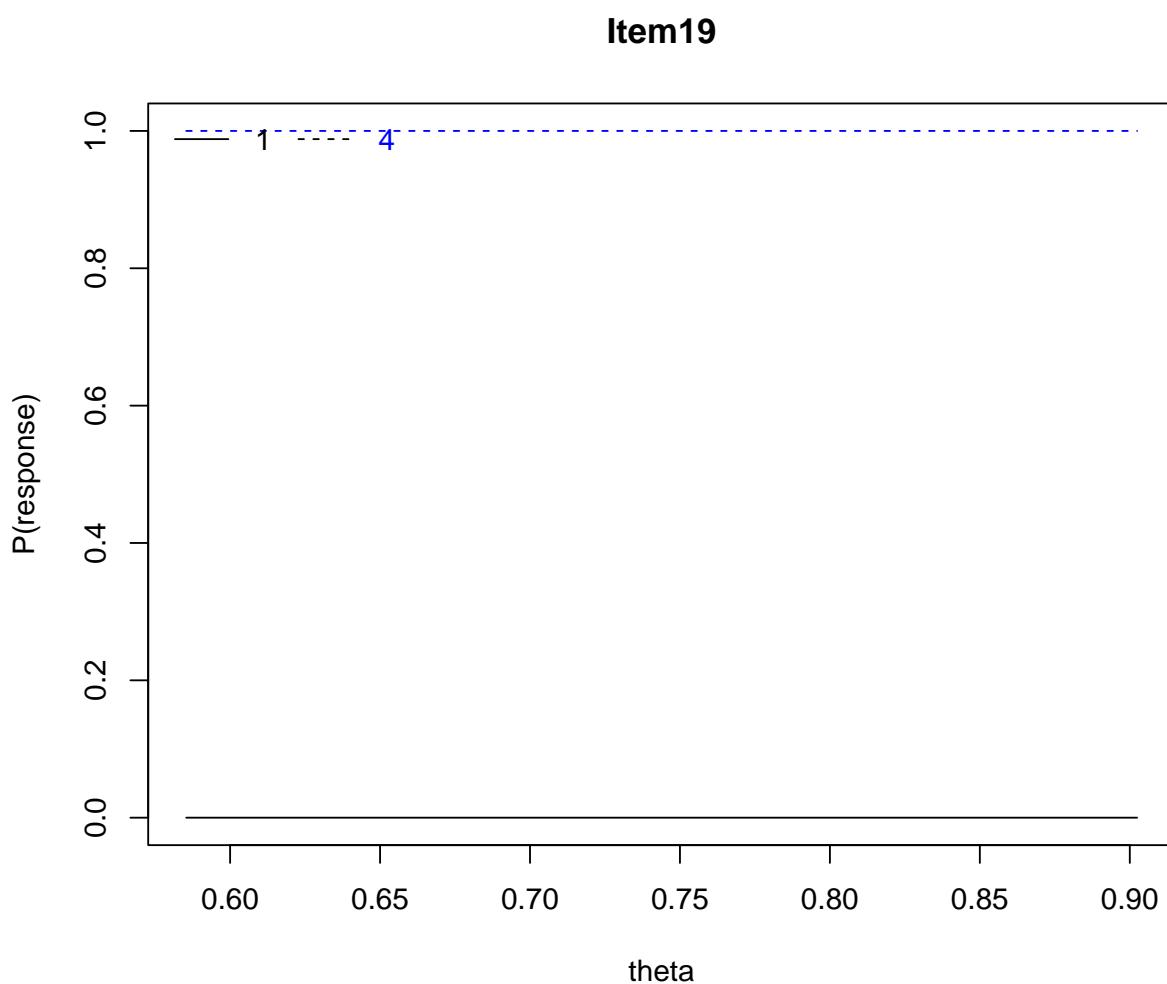
```
psych::irt.responses(results$scores, exam[19], breaks = 2)
```

Number of categories should be increased in order to count frequencies.

Warning in rbind(items, dummy): number of columns of result is not a multiple of vector length (arg 1)

Number of categories should be increased in order to count frequencies.

Warning in rbind(items, dummy): number of columns of result is not a multiple of vector length (arg 1)



Item #19 shows rather flat (no relationship) relations with ability for the correct item and the lone distractor.

**Item 21** The most appropriate score for comparing scores across two or more distributions (e.g., exam scores in math and art classes) is the:

- a) mean
- b) percentile rank
- c) raw score
- d) z-score

Mean = .58,  $r = -.19$ , Distractors: 0.00 0.42 0.00 0.58, skew = -0.30

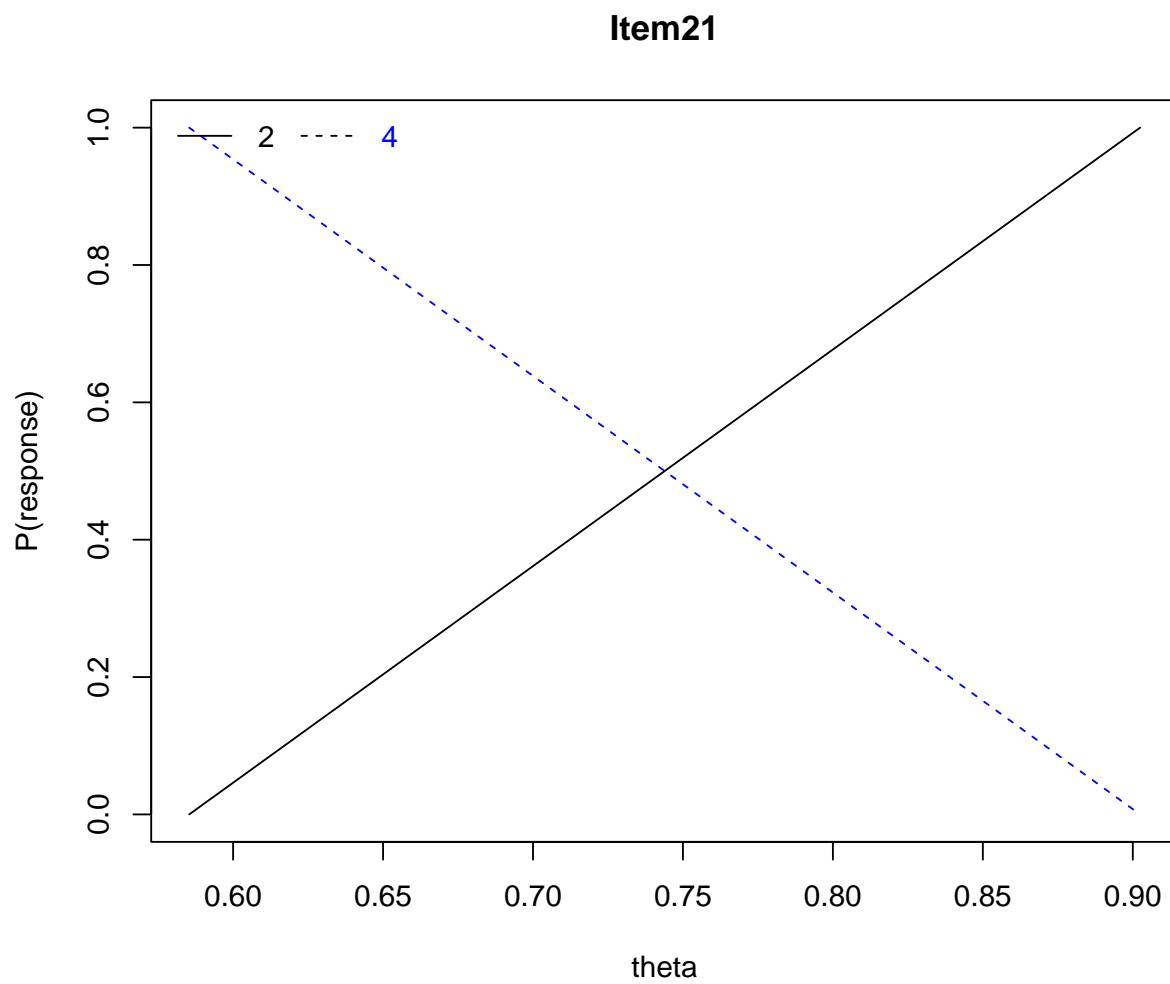
```
psych::irt.responses(results$scores, exam[21], breaks = 2)
```

Number of categories should be increased in order to count frequencies.

Warning in rbind(items, dummy): number of columns of result is not a multiple of vector length (arg 1)

Number of categories should be increased in order to count frequencies.

Warning in rbind(items, dummy): number of columns of result is not a multiple of vector length (arg 1)



For Item #21, a positive relationship between the WRONG answer (2/B) and ability (theta) and a negative relationship between 4/D (incorrect answer) and ability. This makes sense as the point biserial for the overall item was 0-.13.

**Item 37** Of the following, what statement best describes  $r^2 = .49$

- a) strong positive correlation
- b) strong positive or negative correlation
- c) weak positive or negative correlation
- d) weak negative correlation

Mean = .33,  $r = .49$ , Distractors: 0.25 0.33 0.42 0.00, skew = .62

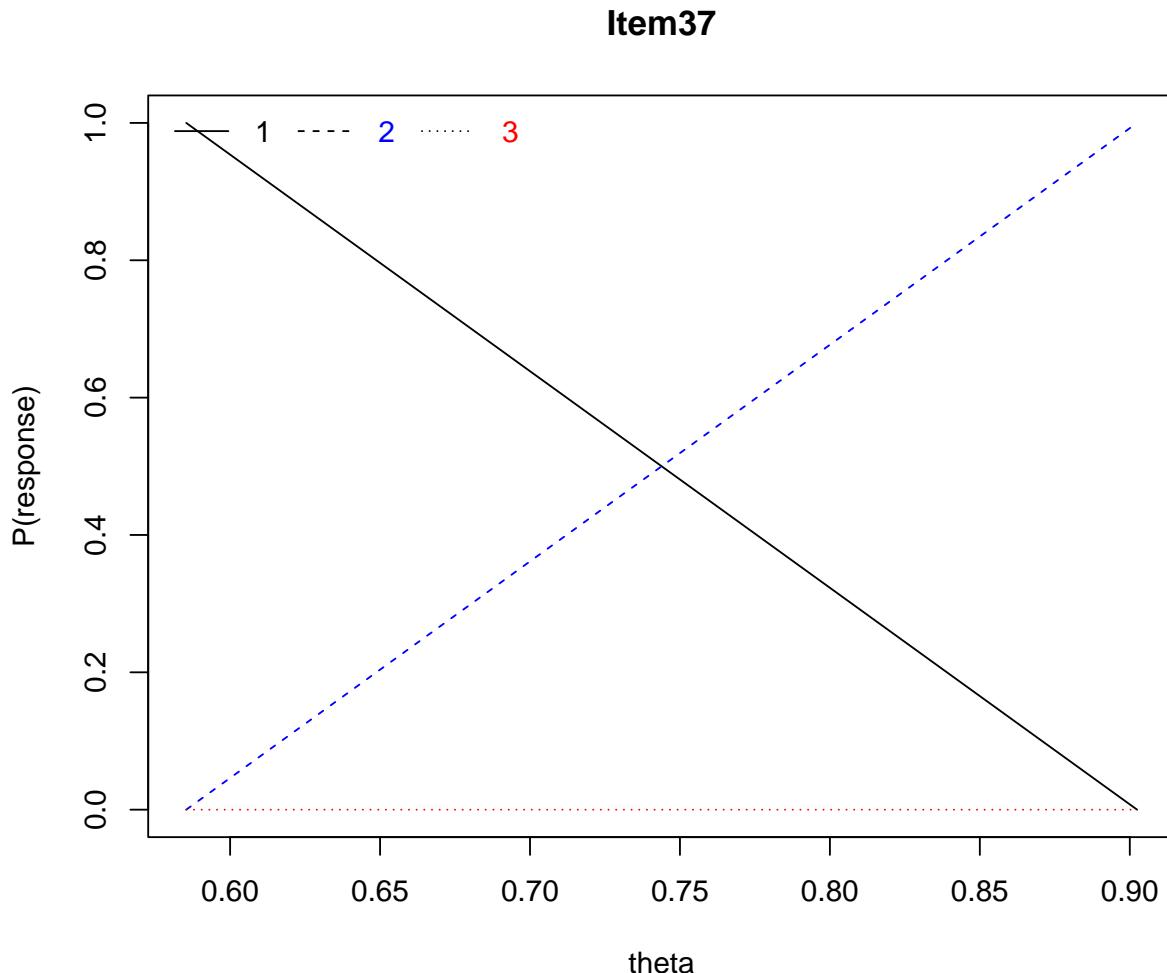
```
psych::irt.responses(results$scores, exam[37], breaks = 2)
```

Number of categories should be increased in order to count frequencies.

Warning in rbind(items, dummy): number of columns of result is not a multiple of vector length (arg 1)

Number of categories should be increased in order to count frequencies.

Warning in rbind(items, dummy): number of columns of result is not a multiple of vector length (arg 1)



For Item #37, a negative relation between endorsing 1/A and ability (a good distractor). No relationship with ability for endorsing 3/C. A positive relation with ability for those endorsing 2/B (correct answer).

**Item 38** When there are no ties among ranks, what is the relationship between the Spearman rho ( $\rho$ ) and the Pearson r ( $r$ )?

- a)  $\rho = r$
- ii)  $\rho > r$

- a)  $\rho < r$
- b) no relationship

Mean = .27,  $r = -.07$ , Distractors: 0.27 0.00 0.00 0.73, skew = .68

*Notice anything else that's funky about Item #38?*

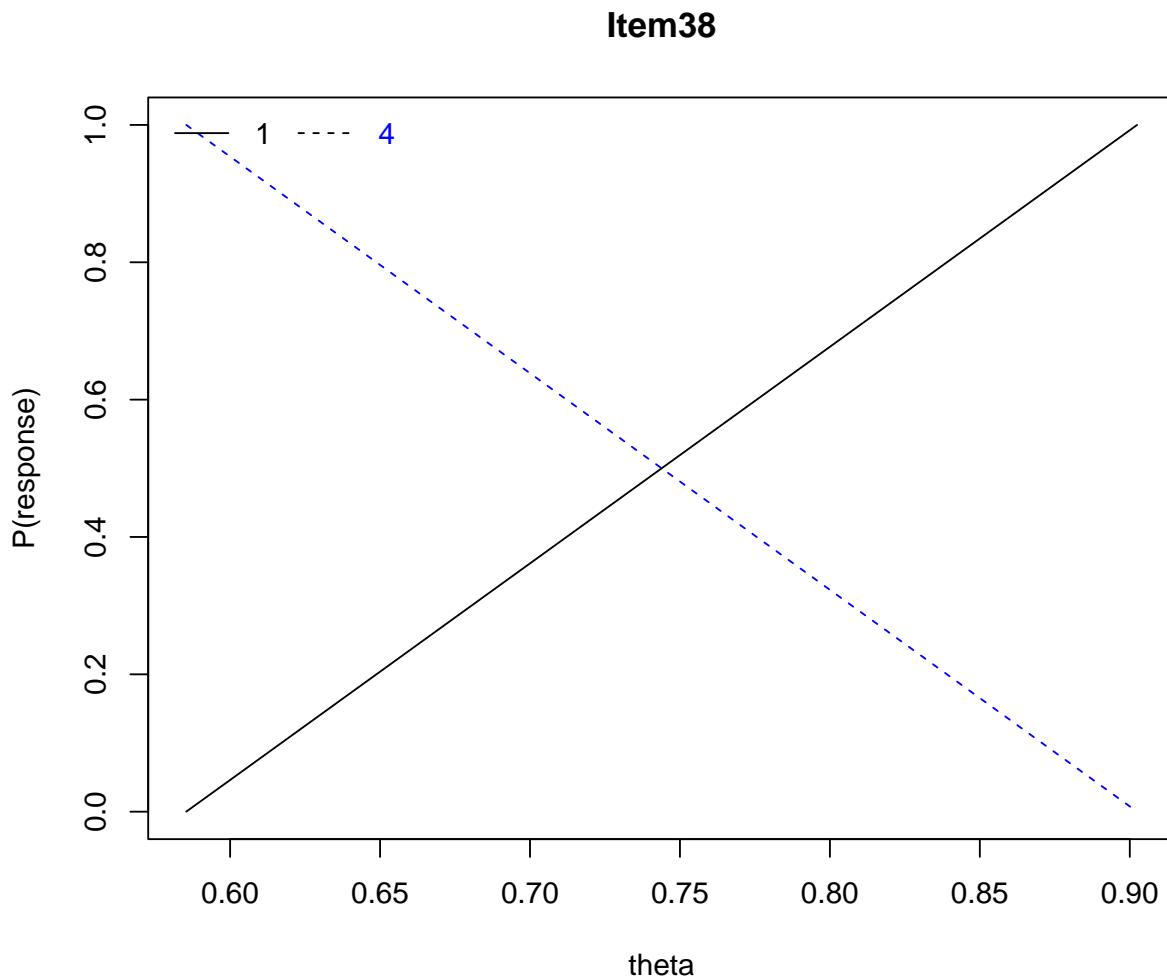
```
psych::irt.responses(results$scores, exam[38], breaks = 2)
```

Number of categories should be increased in order to count frequencies.

Warning in rbind(items, dummy): number of columns of result is not a multiple of vector length (arg 1)

Number of categories should be increased in order to count frequencies.

Warning in rbind(items, dummy): number of columns of result is not a multiple of vector length (arg 1)



## 6.7. CLOSING THOUGHTS ON DEVELOPING MEASURES IN THE EDUCATION/ACHIEVEMENT CONTEXT

For Item #38, there is a positive relationship with ability for endorsing 1/A (correct answer) and a negative relationship with ability for 4/D (incorrect answer).

Regarding overall test characteristics

3340 Final Exam Detailed Item Analysis Report On Final Exam Version A Version A				
Course #:		Instructor:	Bikos	
Course Title:		Description:		
Day/Time:		Term/Year:		
Total Possible Points:	56.00	Median Score:	49.00	Mean Score: 46.83
Student in this group:	12	Mode Score:	53.00	Variance: 37.14
Standard Deviation:	6.09	Highest Score:	53.00	Lowest Score: 35.00
Measure of Skewness:	-0.64	Degree of Kurtosis:	-1.07	Reliability Coefficient (KR20): 0.77

## 6.7 Closing Thoughts on Developing Measures in the Education/Achievement Context

Item analysis tends to be an assessment of *reliability*. However, in the context of educational assessment and achievement exams, there are also *validity* issues.

**Content validity** is concerned with whether or not the scale adequately represents the entirety of the *domain* to be assessed.

In educational and achievement contexts, this is often accomplished with a *table of specifications*. I introduced this in the [Validity lesson](#). As a refresher, I will include another example – imagining that I am going to write a quiz or short exam based on the learning objectives of this, single, lesson. There are a number of different ways to organize the types of knowledge that is being assessed. Since the American Psychological Association (and others) work in “KSAs” (knowledge, skills, attitudes) in their accreditation standards, I will use those.

In creating a table of specifications, we start with the learning objectives. Then we decide what type of items to write and what type of performance level they satisfy. This helps us ensure that all learning objectives are proportionately covered, using a variety of assessment approaches. Otherwise, we might be tempted to include the items that come easily to us or that are from our favorite topics. Personally, I find that when I work on the exam, and am informed by the learning objectives and table of specifications, I find myself tinkering with all three. I am inclined to believe that this results in an ever-increasingly-improved pedagogy.

### Table of Specifications

Learning Objectives	Knowledge	Skills	Attitudes	% of test
Provide a rationale for why having a <i>test bank</i> might be a good idea.			1 item	30%
Describe the effects of skewness on the interpretation of exam results.	2 items			10%
Evaluate the quality of a multiple-choice item on the basis of item difficulty, correlation, and discrimination.		5 items		25%

Learning Objectives	Knowledge	Skills	Attitudes	% of test
Discuss the challenges of identifying an <i>ideal</i> difficulty level for test items. Further elaborate how guessing, speeded tests, interitem correlations, and the purposes of the test influence the <i>ideal difficulty</i> .	2 items		1 item	35%
TOTALS	4 items	5 items	2 items	100%

There are a variety of free resources that help with this process. Below are some that I find helpful:

- [Bloom's Taxonomy Verbs](#) is freely available from Fractus Learning.
- [The Bloom's Taxonomy Verbs Poster for Teachers](#) is a great starter.
- If you have “writer’s block” for writing objectives, here is a [learning outcome generator](#) that may help get you started.
- From APA’s Education Directorate, [Guidance for Writing Behavioral Learning Objectives](#). The APA Guidance really emphasizes key components of well-written behavioral leaning objectives.
  - Objectives should be **observable and measurable**. Write them with action verbs (i.e., recite, compare, define) that Objectives measurable behaviors. The APA CE office disallows the use of “understand” as an action verb,
  - Statements should clearly describe what the learner will know or be able to do **as a result** of having participated.
  - Objectives should focus on the learner and learning (as opposed to what the trainer is doing or leading),
  - Objectives should be appropriate in breadth (not too few or too many).

**Takeaway message:** Together, mapping out exam coverage in a table of specifications PLUS item analysis (difficulty/discrimination) can be powerful tools in educational assessment.

## 6.8 Practice Problems

For this particular lesson, I think some of the most meaningful practice comes from multiple choice and true/false exams that occur in your life. If you are in a class, see if your instructor is willing to share item analysis information that they have received. Learning management systems like Canvas, automatically calculate these.

If you are an instructor, calculate and review item analysis data on your own items. Think about how you might improve items between exams and consider how the dificulty and discrimination capacity of the item changes.

# Chapter 7

## Item Analysis for Likert Type Scale Construction

### [Screencasted Lecture Link](#)

The focus of this lecture is on item analysis for surveys. We use information about alpha coefficients and item-total correlations (within and across subscales) to help assess what we might consider to be *within-scale convergent and discriminant validity* (although we tend to think of it as an assessment of reliability).

### 7.1 Navigating this Lesson

There is about 45 minutes of lecture. If you work through the materials with me it would be plan for an additional hour.

While the majority of R objects and data you will need are created within the R script that sources the chapter, occasionally there are some that cannot be created from within the R framework. Additionally, sometimes links fail. All original materials are provided at the [GitHub site](#) that hosts the book. More detailed guidelines for ways to access all these materials are provided in the OER's [introduction](#)

#### 7.1.1 Learning Objectives

Focusing on this week's materials, make sure you can:

- Define the corrected item-total correlation and compare it to an item-total correlation.
- List the preliminary steps essential for scale construction, beginning with item development.
- Name the type(s; e.g., reliability, validity) of psychometric evaluation that item analytic procedures assess.
- Identify threats to the interpretation of item-total correlations and alpha coefficients.
- Make decisions about item retention, deletion, and revision that balances statistical output with construct definitions.

### 7.1.2 Planning for Practice

In each of these lessons I provide suggestions for practice that allow you to select one or more problems that are graded in difficulty. The least complex is to change the random seed in the research and rework the problem demonstrated in the lesson. The results *should* map onto the ones obtained in the lecture.

The second option involves utilizing one of the simulated datasets available in this OER. The **last lesson** in the OER contains three simulations that could be used for all of the statistics-based practice suggestions. Especially if you started with one of these examples in an earlier lesson, I highly recommend you continue with that.

Alternatively, Lewis and Neville's [2015] Gendered Racial Microaggressions Scale for Black Women will be used in the lessons on exploratory factor analysis; Keum et al.'s Gendered Racial Microaggressions Scale for Asian American Women [Keum et al., 2018] will be used in the lessons on confirmatory factor analysis; and Conover et al.'s [2017] Ableist Microaggressions Scale is used in the lesson on invariance testing. Any of these would be suitable for the PCA and PAF homework assignments.

As a third option, you are welcome to use data to which you have access and is suitable for PCA. In any case, please plan to:

- Produce alpha coefficients, average inter-item correlations, and corrected item-total correlations for the total and subscales, separately.
- Produce correlations between the individual items of one subscale and the subscale scores of all other scales.
- Draft an APA style results section with an accompanying table.

In my example there were only two subscales. If you have more, you will need to compare each subscale with all the others. For example, if you had three subscales: A, B, C, you would need to compare A/B, B/C, and A/C.

### 7.1.3 Readings & Resources

In preparing this chapter, I drew heavily from the following resource(s). Other resources are cited (when possible, linked) in the text with complete citations in the reference list.

- Green & Salkind (2017). Lesson 38: Item analysis using the reliability Procedure. In S.B. Green and N.J. Salkind's, "Using SPSS for Windows and Macintosh: Analyzing and understanding data (8th ed). New York: Pearson.
  - Even though the operation of the chapter uses SPSS, the narration of the "what" and "why" of item analysis is clear and concise. Further, I have not found another chapter (not even in psychometrics texts) that addresses this as completely.
- Szymanski, D. M., & Bissonette, D. (2020). Perceptions of the LGBTQ College Campus Climate Scale: Development and psychometric evaluation. *Journal of Homosexuality*, 67(10), 1412–1428. <https://doi.org/10.1080/00918369.2019.1591788>
  - The research vignette for this lesson.

### 7.1.4 Packages

The packages used in this lesson are embedded in this code. When the hashtags are removed, the script below will (a) check to see if the following packages are installed on your computer and, if not (b) install them.

```
# will install the package if not already installed
# if(!require(tidyverse)){install.packages('tidyverse')}
# if(!require(MASS)){install.packages('MASS')}
# if(!require(psych)){install.packages('psych')}
# if(!require(apaTables)){install.packages('apaTables')}
# if(!require(sjstats)){install.packages('sjstats')}
# if(!require(qualtRics)){install.packages('qualtRics')}
```

## 7.2 Introducing Item Analysis for Survey Development

Item analysis can be used to help determine which items to include and exclude from a scale or subscale. The goal is to select a set of items that yields a summary score (total or mean) that is strongly related to the construct identified and defined in the scale.

- Item analysis is somewhat limiting because we usually cannot relate our items to a direct (external) measure of a construct to select our items.
- Instead, we *trust* (term used lightly) that the items we have chosen, together, represent the construct and we make decisions about the relative strength of each item's correlation to the total score.
- This makes it imperative that we look to both statistics and our construct definition (e.g., how well does each item map onto the construct definition)

If this is initial scale development, the researchers are wise to write more items than needed so that there is flexibility in selecting items with optimal functioning. Szymanski and Bissonette [2020] do this. Their article narrates how they began with 36 items, narrowed it to 24, and – on the basis of subject matter expertise and peer review – further narrowed it to 10. The reduction of additional items happened on the basis of exploratory factor analysis.

### 7.2.1 Workflow for Item Analysis

Step I: Calculate corrected item-total correlations. This involves:

- Reverse-scoring items that are negatively worded.
- Ensuring proper formatting of variables (i.e., numerical and integer formats).
- Evaluating the corrected item-total correlations (“r.drop” in the *psych::alpha* function)
- Consider deleting items with low item-total correlations.
  - Consider the how deleting items might create too narrow of a construct definition. If so, hesitate before deleting.

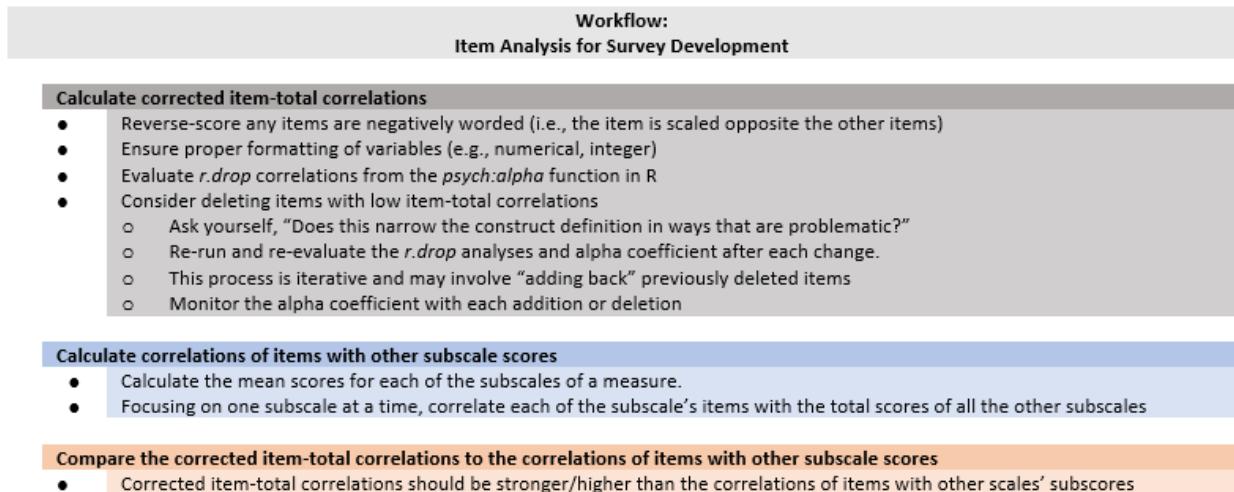


Figure 7.1: Image of workflow for item analysis for survey development.

- Re-run and re-evaluate the *r.drop* values and alpha coefficients after each change.
- This is an iterative process and may involve “adding back” previously deleted items.
- Calculate correlations of items with other subscale scores.
  - Calculate the mean scores for each of the subscales of a measure.
  - Focusing on one subscale at a time, correlate each of the subscale's items with the total score of all the other subscales.
- Compare the corrected item-total correlations to the correlations of items with other subscale scores.
  - The corrected item-total correlations should be stronger/higher than the correlations of items with other subscales.

### 7.3 Research Vignette

The research vignette for this lesson is the development and psychometric evaluation of the Perceptions of the LGBTQ College Campus Climate Scale [Szymanski and Bissonette, 2020]. The scale is six items with responses rated on a 7-point Likert scale ranging from 1 (*strongly disagree*) to 7 (*strongly agree*). Higher scores indicate more negative perceptions of the LGBTQ campus climate. Szymanski and Bissonette [2020] have suggested that the psychometric evaluation supports using the scale in its entirety or as subscales. Each item is listed below with its variable name in parentheses:

- College response to LGBTQ students:
  - My university/college is cold and uncaring toward LGBTQ students. (cold)
  - My university/college is unresponsive to the needs of LGBTQ students. (unresponsive)
  - My university/college provides a supportive environment for LGBTQ students. (unsupportive)

\* this item must be reverse-scored

- LGBTQ Stigma:
  - Negative attitudes toward LGBTQ persons are openly expressed on my university/college campus. (negative)
  - Heterosexism, homophobia, biphobia, transphobia, and cissexism are visible on my university/college campus. (heterosexism)
  - LGBTQ students are harassed on my university/college campus. (harassed)

A preprint of the article is available at ResearchGate. Below is the script for simulating item-level data from the factor loadings, means, and sample size presented in the published article.

Because data is collected at the item level (and I want this resource to be as practical as possible, I have simulated the data for each of the scales at the item level.

Simulating the data involved using factor loadings, means, and correlations between the scales. Because the simulation will produce “out-of-bounds” values, the code below rescales the scores into the range of the Likert-type scaling and rounds them to whole values.

Five additional scales were reported in the Szymanski and Bissonette article [2020]. Unfortunately, I could not locate factor loadings for all of them; and in two cases, I used estimates from a more recent psychometric analysis. When the individual item and their factor loadings are known, I assigned names based on item content (e.g., “lo\_energy”) rather than using item numbers (e.g., “PHQ4”). When I am doing psychometric analyses, I prefer item-level names so that I can quickly see (without having to look up the item names) how the items are behaving. While the focus of this series of chapters is on the LGBTQ Campus Climate scale, this simulated data might be useful to you in one or more of the suggestions for practice (e.g., examining the psychometric characteristics of one or the other scales). The scales, their original citation, and information about how I simulated data for each are listed below.

- **Sexual Orientation-Based Campus Victimization Scale** [Herek, 1993] is a 9-item item scale with Likert scaling ranging from 0 (*never*) to 3 (*two or more times*). Because I was not able to locate factor loadings from a psychometric evaluation, I simulated the data by specifying a 0.8 as a standardized factor loading for each of the items.
- **College Satisfaction Scale** [Helm et al., 1998] is a 5-item scale with Likert scaling ranging from 1 (*strongly disagree*) to 7 (*strongly agree*). Higher scores represent greater college satisfaction. Because I was not able to locate factor loadings from a psychometric evaluation, I simulated the data by specifying a 0.8 as a standardized factor loading for each of the items.
- **Institutional and Goals Commitment** [Pascarella and Terenzini, 1980] is a 6-item subscale from a 35-item measure assessing academic/social integration and institutional/goal commitment (5 subscales total). The measure had with Likert scaling ranging from 1 (*strongly disagree*) to 5 (*strongly agree*). Higher scores on the institutional and goals commitment subscale indicate greater intentions to persist in college. Data were simulated using factor loadings in the source article.
- **GAD-7** [Spitzer et al., 2006] is a 7-item scale with Likert scaling ranging from 0 (*not at all*) to 3 (*nearly every day*). Higher scores indicate more anxiety. I simulated data by estimating factor loadings from Brattmyr et al. [2022].
- **PHQ-9** [Kroenke et al., 2001] is a 9-item scale with Likert scaling ranging from 0 (*not at all*) to 3 (*nearly every day*). Higher scores indicate higher levels of depression. I simulated data by estimating factor loadings from Brattmyr et al. [2022].

#Entering the intercorrelations, means, and standard deviations from the journal article

```
Szymanski_generating_model <- '
#measurement model
CollegeResponse =~ .88*cold + .73*unresponsive + .73*supportive
Stigma =~ .86*negative + .76*heterosexism + .71*harassed
Victimization =~ .8*Vic1 + .8*Vic2 + .8*Vic3 + .8*Vic4 + .8*Vic5 + .8*Vic6 + .8*Vic7 +
CollSat =~ .8*Sat1 + .8*Sat2 + .8*Sat3 + .8*Sat4 + .8*Sat5
Persistence =~ .69*graduation_importance + .63*right_decision + .62*will_register + .55*
Anxiety =~ .851*nervous + .887*worry_control + .894*much_worry + 674*cant_relax + .484*
Depression =~ .798*anhedonia + .425*down + .591*sleep + .913*lo_energy + .441*appetite

#Means
CollegeResponse ~ 2.71*1
Stigma ~3.61*1
Victimization ~ 0.11*1
CollSat ~ 5.61*1
Persistence ~ 4.41*1
Anxiety ~ 1.45*1
Depression ~1.29*1

#Correlations
CollegeResponse ~~ .58*Stigma
CollegeResponse ~~ -.25*Victimization
CollegeResponse ~~ -.59*CollSat
CollegeResponse ~~ -.29*Persistence
CollegeResponse ~~ .17*Anxiety
CollegeResponse ~~ .18*Depression

Stigma ~~ .37*Victimization
Stigma ~~ -.41*CollSat
Stigma ~~ -.19*Persistence
Stigma ~~ .27*Anxiety
Stigma ~~ .24*Depression

Victimization ~~ -.22*CollSat
Victimization ~~ -.04*Persistence
Victimization ~~ .23*Anxiety
Victimization ~~ .21*Depression

CollSat ~~ .53*Persistence
CollSat ~~ -.29*Anxiety
CollSat ~~ -.32*Depression

Persistence ~~ -.22*Anxiety
Persistence ~~ -.26*Depression
```

```

Anxiety ~ ~ .76*Depression
'

set.seed(240218)
dfSzy <- lavaan::simulateData(model = Szymanski_generating_model,
                               model.type = "sem",
                               meanstructure = T,
                               sample.nobs=646,
                               standardized=FALSE)

#used to retrieve column indices used in the rescaling script below
col_index <- as.data.frame(colnames(dfSzy))

#The code below loops through each column of the dataframe and assigns the scaling accordingly
#Rows 1 thru 6 are the Perceptions of LGBTQ Campus Climate Scale
#Rows 7 thru 15 are the Sexual Orientation-Based Campus Victimization Scale
#Rows 16 thru 20 are the College Satisfaction Scale
#Rows 21 thru 26 are the Institutional and Goals Commitment Scale
#Rows 27 thru 33 are the GAD7
#Rows 34 thru 42 are the PHQ9

for(i in 1:ncol(dfSzy)){
  if(i >= 1 & i <= 6){
    dfSzy[,i] <- scales::rescale(dfSzy[,i], c(1, 7))
  }
  if(i >= 7 & i <= 15){
    dfSzy[,i] <- scales::rescale(dfSzy[,i], c(0, 3))
  }
  if(i >= 16 & i <= 20){
    dfSzy[,i] <- scales::rescale(dfSzy[,i], c(1, 7))
  }
  if(i >= 21 & i <= 26){
    dfSzy[,i] <- scales::rescale(dfSzy[,i], c(1, 5))
  }
  if(i >= 27 & i <= 33){
    dfSzy[,i] <- scales::rescale(dfSzy[,i], c(0, 3))
  }
  if(i >= 34 & i <= 42){
    dfSzy[,i] <- scales::rescale(dfSzy[,i], c(0, 3))
  }
}

#rounding to integers so that the data resembles that which was collected
library(tidyverse)
dfSzy <- dfSzy %>% round(0)

#quick check of my work

```

```
#psych::describe(dfSzy)

#Reversing the supportive item on the Perceptions of LGBTQ Campus Climate Scale so that the exercises

dfSzy <- dfSzy %>%
  dplyr::mutate(supportiveNR = 8 - supportive)

#Reversing three items on the Institutional and Goals Commitments scale so that the exercises

dfSzy <- dfSzy %>%
  dplyr::mutate(not_graduateNR = 8 - not_graduate)%>%
  dplyr::mutate(undecidedNR = 8 - undecided)%>%
  dplyr::mutate(grades_unimportantNR = 8 - grades_unimportant)

dfSzy <- dplyr::select(dfSzy, -c(supportive, not_graduate, undecided, grades_unimportant))
```

The optional script below will let you save the simulated data to your computing environment as either an .rds object (preserves any formatting you might do) or a.csv file (think “Excel lite”).

```
# to save the df as an .rds (think 'R object') file on your computer;
# it should save in the same file as the .rmd file you are working
# with saveRDS(dfSzy, 'SzyDF.rds') bring back the simulated dat from
# an .rds file dfSzy <- readRDS('SzyDF.rds')

# write the simulated data as a .csv write.table(dfSzy,
# file='SzyDF.csv', sep=',', col.names=TRUE, row.names=FALSE) bring
# back the simulated dat from a .csv file dfSzy <-
# read.csv('SzyDF.csv', header = TRUE)
```

Although Szymanski and Bissonette report inter-item correlations, it does not appear that they used item analysis to guide their selection of items. In fact, it is not necessary to do so. I teach item analysis because I think it provides a conceptual grounding for future lessons on exploratory and confirmatory factor analysis.

## 7.4 Step I: Corrected item-total correlations

You might think of corrected item-total correlations as form a *within-scale of convergent validity*.

- If needed, transform any items (i.e., reverse-coding) and calculate a total score.
- Calculate *corrected item-total correlations* by correlating each item to the total score *excluding* the item being evaluated.
  - to the degree that the item total represents the construct of interest, the items should be strongly correlated with the corrected total score.
- Make decisions about items and scales. For items that have low or negative correlations

Calculate corrected item-total correlations
<ul style="list-style-type: none"> <li>● Reverse-score any items are negatively worded (i.e., the item is scaled opposite the other items)</li> <li>● Ensure proper formatting of variables (e.g., numerical, integer)</li> <li>● Evaluate <i>r.drop</i> correlations from the <i>psych:alpha</i> function in R</li> <li>● Consider deleting items with low item-total correlations <ul style="list-style-type: none"> <li>○ Ask yourself, "Does this narrow the construct definition in ways that are problematic?"</li> <li>○ Re-run and re-evaluate the <i>r.drop</i> analyses and alpha coefficient after each change.</li> <li>○ This process is iterative and may involve "adding back" previously deleted items</li> <li>○ Monitor the alpha coefficient with each addition or deletion</li> </ul> </li> </ul>

Figure 7.2: Image of the first step in the workflow for item analysis for survey development.

- consider deletion,
- consider revision (requires new data collection).
- Each time an item is deleted, the item analysis needs to be repeated because it changes the total-scale score.
  - In fact, it's a very iterative process. At times, researchers "add back" a previously deleted item (once others are deleted) because with each deletion/addition the statistical construct definition is evolving.
- In multidimensional scales, if the total-scale score is ever used, researchers should conduct item analyses separately for both the total- and the sub- scale scores.

There are reasons to not "blindly follow the results of an item analysis" [Green and Salkind, 2017].

- **Method factors** (aka *method effects*) are common *methods* that are irrelevant to the characteristics or traits being measured – yet when analyzed they share variance. Examples of these include negatively word items and common phrasing such as "My supervisor tells me" versus "I receive feedback" [Chyung et al., 2018c].
- Adequacy of construct representation. That is, how broad is the construct and to what degree do the items represent the entire construct? Threats to the adequacy of the construct representation include:
  - Writing items on a particular, narrow, aspect of the construct, ignoring others.
  - Retaining items that are strongly correlated while deleting those that whose correlations are less strong (although they represent a different aspect of the construct).

This means we should think carefully and simultaneously about:

- statistical properties of the item and overall scale,
- construct definition,
- scale structure (unidimensional? multidimensional? hierarchical?).

### 7.4.1 Data Prep

Let's do the operational work to get all the pieces we need:

1. Reverse-code the *supportive* variable.

2. From the raw data calculate
  - total-scale score,
  - college response subscale,
  - stigma subscale.
3. The result is dataset with the item-level data and the three mean scores (total, college response, stigma).

When we review the information about this scale, we learn that the *supportive* item is scaled in the opposite direction of the rest of the items. That is, a higher score on *supportive* would indicate a positive perception of the campus climate for LGBTQ individuals whereas higher scores on the remaining items indicate a more negative perception. Before moving forward, we must reverse score this item.

In doing this, I will briefly note that in this case I have given my variables one-word names that represent each item. Many researchers (including myself) will often give variable names that are alpha numerical: LGBTQ1, LGBTQ2, LGBTQn. Either is acceptable. In the psychometrics case, I find the one-word names to be useful shortcuts as I begin to understand the inter-item relations.

In reverse-scoring the *supportive* item, I will rename it “*unsupportive*” as an indication of its reversed direction.

```
dfSzy <- dfSzy %>%
  dplyr::mutate(unsupportive = 8 - supportiveNR)  #scaling 1 to 7; so we subtract from 8

# psych::describe(dfSzy)
```

Next, we score the items. In our simulation, we have no missing data. Using an available information approach (AIA; [Parent, 2013]) where it is common to allow 20-25% missingness, we might allow the total-scale score to calculate if there is one variable missing. I am inclined to also score the subscales if there is one missing; thus, I set the threshold at 66%. The *mean\_n()* function in the *sjstats* packages is especially helpful for this.

```
LGBTQvars <- c("cold", "unresponsive", "negative", "heterosexism", "harassed",
  "unsupportive")
ResponseVars <- c("cold", "unresponsive", "unsupportive")
Stigmavars <- c("negative", "heterosexism", "harassed")

dfSzy$Total <- sjstats::mean_n(dfSzy[, LGBTQvars], 0.8)  #will create the mean for each individual
dfSzy$Response <- sjstats::mean_n(dfSzy[, ResponseVars], 0.66)  #will create the mean for each individual
dfSzy$Stigma <- sjstats::mean_n(dfSzy[, Stigmavars], 0.66)  #will create the mean for each individual

# If the scoring code above does not work for you, try the format
# below which involves inserting two periods in front of the variable
# list. One example is provided. dfLewis$Belonging <-
# sjstats::mean_n(dfLewis[., .Belonging_vars], 0.80)
```

While we are at it, let's just create tiny dfs with just our variables of interest.

```
LGBTQ <- dplyr::select(dfSzy, cold, unresponsive, unsupportive, negative,
  heterosexism, harassed)
Response <- dplyr::select(dfSzy, cold, unresponsive, unsupportive)
Stigma <- dplyr::select(dfSzy, negative, heterosexism, harassed)
```

#### 7.4.2 Calculating Item-Total Correlation Coefficients

Let's first ask, "Is there support for this instrument as a unidimensional measure?" To do that, we get an alpha for the whole scale score.

The easiest way to do this is apply the *alpha()* function to a tiny df with the variables in that particular scale or subscale. Any variables should be pre-reversed.

```
LGBTQalpha <- psych::alpha(LGBTQ) #Although unnecessary, I have saved the output as objects b
```

Reliability analysis  
Call: psych::alpha(x = LGBTQ)

	raw_alpha	std.alpha	G6(smc)	average_r	S/N	ase	mean	sd	median_r
	0.7	0.7	0.68	0.28	2.4	0.018	4	0.63	0.25

95% confidence boundaries  
lower alpha upper  
Feldt 0.66 0.7 0.74  
Duhachek 0.66 0.7 0.74

Reliability if an item is dropped:

	raw_alpha	std.alpha	G6(smc)	average_r	S/N	alpha	se	var.r	med.r
cold	0.64	0.64	0.61	0.27	1.8	0.022	0.0066	0.22	
unresponsive	0.66	0.66	0.63	0.28	2.0	0.021	0.0073	0.25	
unsupportive	0.67	0.67	0.63	0.29	2.0	0.021	0.0058	0.25	
negative	0.66	0.66	0.63	0.28	2.0	0.021	0.0084	0.25	
heterosexism	0.66	0.66	0.63	0.28	2.0	0.021	0.0087	0.25	
harassed	0.67	0.67	0.64	0.29	2.0	0.021	0.0078	0.25	

Item statistics

	n	raw.r	std.r	r.cor	r.drop	mean	sd
cold	646	0.68	0.68	0.59	0.49	4.1	1.03
unresponsive	646	0.63	0.63	0.51	0.43	4.3	0.99
unsupportive	646	0.62	0.62	0.51	0.42	3.7	0.98
negative	646	0.64	0.63	0.51	0.42	4.0	1.04
heterosexism	646	0.61	0.63	0.51	0.43	4.0	0.90
harassed	646	0.63	0.61	0.49	0.41	3.9	1.07

		Non missing response frequency for each item							
		1	2	3	4	5	6	7	miss
cold		0.00	0.04	0.22	0.40	0.23	0.09	0.00	0
unresponsive		0.00	0.03	0.17	0.37	0.33	0.09	0.01	0
unsupportive		0.01	0.07	0.35	0.37	0.17	0.02	0.01	0
negative		0.01	0.07	0.23	0.39	0.24	0.05	0.00	0
heterosexism		0.00	0.03	0.24	0.43	0.26	0.03	0.00	0
harassed		0.01	0.07	0.27	0.37	0.22	0.05	0.01	0

Examining our list, the overall alpha is 0.70. Further, the average inter-item correlation (*average\_r*) is .28.

*And just hold up a minute, I thought you told us alpha was bad!*

- While it is less than ideal, we still use it all the time:
  - keeping in mind its relative value (does it increase/decrease, holding other things [like sample size] constant) and
  - examining alpha alternatives (such as we obtained from the omega output)
- Why alpha in this context? Its information about *consistency* is essential. In evaluating a scale's reliability, we do want to know if items (unidimensionally or across subscales) are responding consistently high/middle/low.

We take note of two columns:

- *r.cor* is the correlation between the item and the total-scale score with the row-item included. When our focus is on the contribution of a specific item, this information is not helpful since this column gets “extra credit” for the redundancy of the duplicated item.
- *r.drop* is the corrected item-total correlation. This is the better choice because it excludes the row-item being evaluated (eliminating the redundancy) prior to conducting the correlation.
  - Looking at the two columns, notice that the *r.drop* correlations are lower. This is the more honest correlation of the item with the *other* items.
  - In item analysis, we look for items that have relatively high (assessing redundancy or duplication) or items and relatively low (indicating they are unlike the other items) values.

If we thought an item was problematic, we could eliminate it and rerun the analysis. Because we are looking at a list of items that “made the cut,” we don’t have any items that are concerningly high or low. For demonstration purposes, though, the corrected item-total correlation (*r.drop*) of the *harassed* variable was the lowest (0.40). Let’s re-run the analysis excluding this item.

```
minus_harassed <- dplyr::select(dfSzy, cold, unresponsive,
                                 unsupportive, negative, heterosexism)
```

```
psych::alpha(minus_harassed)
```

Reliability analysis

Call: psych::alpha(x = minus\_harassed)

raw_alpha	std.alpha	G6(smc)	average_r	S/N	ase	mean	sd	median_r
0.67	0.67	0.64	0.29	2	0.021	4	0.65	0.25

95% confidence boundaries

	lower	alpha	upper
Feldt	0.63	0.67	0.71
Duhachek	0.63	0.67	0.71

Reliability if an item is dropped:

	raw_alpha	std.alpha	G6(smc)	average_r	S/N	alpha	se	var.r	med.r
cold	0.58	0.58	0.53	0.26	1.4	0.027	0.0060	0.22	
unresponsive	0.62	0.62	0.56	0.29	1.6	0.025	0.0079	0.25	
unsupportive	0.61	0.61	0.56	0.28	1.6	0.025	0.0072	0.25	
negative	0.65	0.64	0.58	0.31	1.8	0.022	0.0093	0.30	
heterosexism	0.64	0.64	0.58	0.31	1.8	0.023	0.0097	0.30	

Item statistics

	n	raw.r	std.r	r.cor	r.drop	mean	sd
cold	646	0.72	0.71	0.61	0.50	4.1	1.03
unresponsive	646	0.66	0.66	0.53	0.43	4.3	0.99
unsupportedive	646	0.67	0.67	0.55	0.45	3.7	0.98
negative	646	0.63	0.62	0.46	0.37	4.0	1.04
heterosexism	646	0.60	0.62	0.47	0.38	4.0	0.90

Non missing response frequency for each item

	1	2	3	4	5	6	7	miss
cold	0.00	0.04	0.22	0.40	0.23	0.09	0.00	0
unresponsive	0.00	0.03	0.17	0.37	0.33	0.09	0.01	0
unsupportedive	0.01	0.07	0.35	0.37	0.17	0.02	0.01	0
negative	0.01	0.07	0.23	0.39	0.24	0.05	0.00	0
heterosexism	0.00	0.03	0.24	0.43	0.26	0.03	0.00	0

The alpha ( $\alpha = 0.67$ ) decreases; the overall inter-item correlations increase, slightly (*average\_r*; 0.29). This decrease in alpha is an example of how sample size can affect the result.

Examining item-level statistics, we do see greater variability (0.37 to 0.50) in the corrected item-total correlations (*r.drop*). What might this mean?

- The item we dropped (*harassed*) may be clustering with *negative* and *heterosexism* in a sub-ordinate factor (think subscale).
- Although item analysis is more of a tool in assessing reliability, the statistical information that *harassed* provided may broaden the construct definition (definitions are a concern of *validity*) of perceptions of campus climate such that it is necessary to ground/anchor *negative* and *heterosexism*.

Tentative conclusion: there is evidence that this is not a unidimensional measure. Let's move on to inspect similar data for each of the subscales. We'll start with the College Response subscale.

```
RESPalpha <- psych::alpha(Response)
RESPalpha
```

#### Reliability analysis

Call: psych::alpha(x = Response)

	raw_alpha	std.alpha	G6(smc)	average_r	S/N	ase	mean	sd	median_r
	0.66	0.66	0.57	0.39	1.9	0.023	4	0.77	0.4

#### 95% confidence boundaries

	lower	alpha	upper
Feldt	0.61	0.66	0.70
Duhachek	0.62	0.66	0.71

#### Reliability if an item is dropped:

	raw_alpha	std.alpha	G6(smc)	average_r	S/N	alpha	se	var.r	med.r
cold	0.52	0.52	0.35	0.35	1.1	0.038	NA	0.35	
unresponsive	0.60	0.60	0.42	0.42	1.5	0.032	NA	0.42	
unsupportive	0.58	0.58	0.40	0.40	1.4	0.033	NA	0.40	

#### Item statistics

	n	raw.r	std.r	r.cor	r.drop	mean	sd
cold	646	0.80	0.79	0.62	0.50	4.1	1.03
unresponsive	646	0.76	0.76	0.55	0.45	4.3	0.99
unsupportive	646	0.76	0.77	0.57	0.46	3.7	0.98

#### Non missing response frequency for each item

	1	2	3	4	5	6	7	miss
cold	0.00	0.04	0.22	0.40	0.23	0.09	0.00	0
unresponsive	0.00	0.03	0.17	0.37	0.33	0.09	0.01	0
unsupportive	0.01	0.07	0.35	0.37	0.17	0.02	0.01	0

The alpha for the College Response subscale is 0.66; this is a bit lower than the alpha for the total scale score ( $\alpha = 0.70$ ). The average inter-item correlation (*average\_r*) is higher somewhat higher than the that of the total scale score (0.39 versus 0.28).

Examining the corrected item-total correlations (*r.drop*) indicates strong correlations between the row-item with the remaining variables (0.45 to 0.50).

Let's examine at the Stigma subscale.

```
STIGalpha <- psych::alpha(Stigma)
STIGalpha
```

```
Reliability analysis
Call: psych::alpha(x = Stigma)
```

raw_alpha	std.alpha	G6(smc)	average_r	S/N	ase	mean	sd	median_r	
0.62	0.63	0.53		0.36	1.7	0.025	4	0.76	0.36

```
95% confidence boundaries
      lower alpha upper
Feldt      0.57  0.62  0.67
Duhachek   0.57  0.62  0.67
```

Reliability if an item is dropped:

	raw_alpha	std.alpha	G6(smc)	average_r	S/N	alpha	se	var.r	med.r
negative	0.52	0.52	0.35		0.35	1.1	0.038	NA	0.35
heterosexism	0.53	0.53	0.36		0.36	1.1	0.037	NA	0.36
harassed	0.53	0.54	0.37		0.37	1.2	0.036	NA	0.37

Item statistics

	n	raw.r	std.r	r.cor	r.drop	mean	sd
negative	646	0.77	0.76	0.56	0.44	4.0	1.0
heterosexism	646	0.73	0.76	0.55	0.44	4.0	0.9
harassed	646	0.77	0.75	0.54	0.43	3.9	1.1

Non missing response frequency for each item

	1	2	3	4	5	6	7	miss
negative	0.01	0.07	0.23	0.39	0.24	0.05	0.00	0
heterosexism	0.00	0.03	0.24	0.43	0.26	0.03	0.00	0
harassed	0.01	0.07	0.27	0.37	0.22	0.05	0.01	0

The alpha for the Stigma subscale is 0.63; this is a bit lower than the alpha for the total-scale ( $\alpha = 0.70$ ). In contrast, the inter-item correlation (*average\_r*) is a bit higher than the same for the total scale score (0.36 versus 0.28).

Examining the corrected item-total correlations (*r.drop*) indicates a strong correlation between the row-item with the remaining variables (0.43 to 0.44).

In addition to needing strong inter-item correlations (which we just assessed) we want the individual items to correlate more strongly with themselves than with the other scale. Let's do that next.

## 7.5 Step II: Correlating Items with Other Scale Totals

You might think of this step as analyzing a within-scale version of discriminant validity. That is, we do not want individual items from one scale to correlate more highly with subscale scores of other scales, than it does with its own.

- Calculate scale scores for each of the subscales of a measure.

**Calculate correlations of items with other subscale scores**

- Calculate the mean scores for each of the subscales of a measure.
- Focusing on one subscale at a time, correlate each of the subscale's items with the total scores of all the other subscales

Figure 7.3: Image of the second step in the workflow for item analysis for survey development.

- Focusing on one subscale at a time, correlate each of the subscale's items with the total scores of all the other subscales.
- Comparing to the results of Step I's corrected item-total process, each item should have stronger correlations with its own items (i.e., the corrected item-total correlations) than with the other subscale total scores.

In this first analysis, we will correlate the individual *items* from the College Response subscale to the Stigma subscale *score*.

```
apaTables::apa.cor.table(dfSzy[c("cold", "unresponsive", "unsupportive",
  "Stigma")])
```

Means, standard deviations, and correlations with confidence intervals

Variable	M	SD	1	2	3
1. cold	4.11	1.03			
2. unresponsive	4.31	0.99	.40**		
			[.34, .47]		
3. unsupportive	3.69	0.98	.42**	.35**	
			[.36, .49]	[.28, .42]	
4. Stigma	3.96	0.76	.33**	.28**	.26**
			[.26, .39]	[.21, .35]	[.18, .33]

Note. M and SD are used to represent mean and standard deviation, respectively. Values in square brackets indicate the 95% confidence interval.

The confidence interval is a plausible range of population correlations that could have caused the sample correlation (Cumming, 2014).

\* indicates  $p < .05$ . \*\* indicates  $p < .01$ .

We want the corrected item-total correlations of the College Response scale (0.45 to 0.50; retrieved from the *r.drop* column above) to be higher than their correlations with the Stigma scale (0.28 to 0.33) with all three items). These items follow the pattern.

Let's examine the individual items from the Stigma scale with the College Response subscale score.

```
apaTables::apa.cor.table(dfSzy[c("negative", "heterosexism", "harassed",
  "Response")])
```

Means, standard deviations, and correlations with confidence intervals

Variable	M	SD	1	2	3
1. negative	3.96	1.04			
2. heterosexism	4.00	0.90	.37**		
			[.30, .43]		
3. harassed	3.91	1.07	.36**	.35**	
			[.29, .42]	[.28, .42]	
4. Response	4.04	0.77	.29**	.29**	.27**
			[.22, .36]	[.22, .36]	[.20, .34]

Note. M and SD are used to represent mean and standard deviation, respectively.

Values in square brackets indicate the 95% confidence interval.

The confidence interval is a plausible range of population correlations that could have caused the sample correlation (Cumming, 2014).

\* indicates  $p < .05$ . \*\* indicates  $p < .01$ .

Similarly, the corrected item-total correlations (i.e.,  $r.drop$ ) from the Stigma subscale (0.43 to 0.44) are stronger than their correlation with the College Response subscale (0.27 to 0.29).

## 7.6 Step III: Interpreting and Writing up the Results

Now it's time to make sense of the results. Here's a reminder from the workflow:

- Compare the corrected item-total correlations to the correlations of items with other subscale scores
  - Corrected item-total correlations should be stronger/higher than the correlations of items with other scales' subscores

Figure 7.4: Image of the third step in the workflow for item analysis for survey development.

Tabling these results can be really useful to present them effectively. As is customary in APA style tables, when the item is in bold, the value represents its relationship with its own factor. These values come from the corrected item-total ( $r.drop$ ) values where the item is singled out and correlated with the remaining items in its subscale.

---

 Item-Total Correlations of Items with their Own and Other Subscales
 

---

Variables	College Response	Stigma
cold	.50	0.33
unresponsive	.45	0.28
unsupportive	.46	0.26
negative	0.29	<b>0.44</b>
heterosexism	0.29	<b>0.44</b>
harassed	0.27	<b>0.43</b>

Although I pitched this type of item-analysis as *reliability*, to some degree it assesses within-scale **convergent and discriminant validity** because we can see the item relates more strongly to members of its own scale (higher correlation coefficients indicate *convergence*) than to the subscale scores of the other scales. When this pattern occurs, we can argue that the items *discriminate* well.

## Results

Item analyses were conducted on the six items hypothesized to assess perceptions of campus climate for members of the LGBTQ community. To assess the within-scale convergent and discriminant validity of the College Response and Stigma subscales, each item was correlated with its own scale (with the item removed) and with the other subscale (see Table 1). In all cases, items were more highly correlated with their own scale than with the other scale. Coefficient alphas were 0.66, 0.63, and 0.70 for the College Response, Stigma, and total-scale scores, respectively. We concluded that the within-scale convergent and discriminant validity of this measure is strong.

For your consideration: You are at your dissertation defense. For one of your measures, the Cronbach's alpha is .45. A committee member asks, "So why was the alpha coefficient so low?" On the basis of what you have learned in this lesson, how do you respond?

## 7.7 A Conversation with Dr. Szymanski

Doctoral students Julian Williams (Industrial-Organizational Psychology), Jaylee York (Clinical Psychology), and I were able to interview the first author (Dawn Szymanski, PhD) about the article [[Szymanski and Bissonette, 2020](#)] and what it means. Here's a direct [link](#) to that interview.

Among other things, we asked:

- How were you able to create such an efficient (6 items) survey?
- What were the decisions around a potential third factor of *visibility*?
- What would you say to senior leadership on a college campus (where there hiring policies that discriminate against LGBTQIA+ applicants) who will acknowledge the research that indicates that the existence of such policies are associated with reduced well-being for members of the LGBTQIA+ community but who insists that their campus is different?
- How would you like to see the article used?

## 7.8 Practice Problems

In each of these lessons I provide suggestions for practice that allow you to select one or more problems that are graded in difficulty. For this lesson, please locate item-level data for a scale that has the potential for at least two subscales and a total-scale score. Ideally, you selected such data for practice from the prior lesson. Then, please examine the following:

- produce alpha coefficients, average inter-item correlations, and corrected item-total correlations for the total and subscales, separately
- produce correlations between the individual items of one subscale and the subscale scores of all other scales
- draft an APA style results section with an accompanying table.

In my example, there were only two subscales. If you have more, you will need to compare each subscale with all the others. For example, if you had three subscales: A, B, C, you would need to compare A/B, B/C, and A/C.

### 7.8.1 Problem #1: Play around with this simulation.

Copy the script for the simulation and then change (at least) one thing in the simulation to see how it impacts the results.

If item analysis is new to you, copy the script for the simulation and then change (at least) one thing in the simulation to see how it impacts the results. Perhaps you just change the number in “set.seed(210827)” from 210827 to something else. Your results should parallel those obtained in the lecture, making it easier for you to check your work as you go.

### 7.8.2 Problem #2: Use raw data from the ReCentering Psych Stats survey on Qualtrics.

The script below pulls live data directly from the ReCentering Psych Stats survey on Qualtrics. As described in the [Scrubbing and Scoring chapters](#) of the ReCentering Psych Stats Multivariate Modeling volume, the Perceptions of the LGBTQ College Campus Climate Scale [Szymanski and Bissonette, 2020] was included (LGBTQ) and further adapted to assess perceptions of campus climate for Black students (BLst), non-Black students of color (nBSoC), international students (INTst), and students with disabilities (wDIS). Consider conducting the analyses on one of these scales or merging them together and imagining subscales according to identity/group (LGBTQ, Black, non-Black, disability, international) or College Response and Stigma across the different groups.

```
library(tidyverse)
# only have to run this ONCE to draw from the same Qualtrics
# account...but will need to get different token if you are changing
# between accounts
library(qualtRics)
# qualtrics_api_credentials(api_key =
```

```

# 'mUgPMYsYkiWpMFkwHale1QE5HNmh5LRUaA8d9PDg', base_url =
# 'spupsych.az1.qualtrics.com', overwrite = TRUE, install = TRUE)
QTRX_df <- qualtrics::fetch_survey(surveyID = "SV_b2cClqA1LGQ6nLU", time_zone = NULL,
    verbose = FALSE, label = FALSE, convert = FALSE, force_request = TRUE,
    import_id = FALSE)
climate_df <- QTRX_df %>%
  select("Blst_1", "Blst_2", "Blst_3", "Blst_4", "Blst_5", "Blst_6",
    "nBSoC_1", "nBSoC_2", "nBSoC_3", "nBSoC_4", "nBSoC_5", "nBSoC_6",
    "INTst_1", "INTst_2", "INTst_3", "INTst_4", "INTst_5", "INTst_6",
    "wDIS_1", "wDIS_2", "wDIS_3", "wDIS_4", "wDIS_5", "wDIS_6", "LGBTQ_1",
    "LGBTQ_2", "LGBTQ_3", "LGBTQ_4", "LGBTQ_5", "LGBTQ_6")
# Item numbers are supported with the following items: _1 'My campus
# unit provides a supportive environment for ___ students' _2
# '_____ is visible in my campus unit' _3 'Negative attitudes
# toward persons who are ___ are openly expressed in my campus
# unit.' _4 'My campus unit is unresponsive to the needs of ___
# students.' _5 'Students who are___ are harassed in my campus
# unit.' _6 'My campus unit is cold and uncaring toward ___
# students.'

# Item 1 on each subscale should be reverse coded. The College
# Response scale is composed of items 1, 4, 6, The Stigma scale is
# composed of items 2,3, 5

```

The optional script below will let you save the simulated data to your computing environment as either a .csv file (think “Excel lite”) or .rds object (preserves any formatting you might do).

```

# write the simulated data as a .csv write.table(climate_df,
# file='climate_df.csv', sep=',', col.names=TRUE, row.names=FALSE)
# bring back the simulated dat from a .csv file climate_df <-
# read.csv ('climate_df.csv', header = TRUE)

```

```

# to save the df as an .rds (think 'R object') file on your computer;
# it should save in the same file as the .rmd file you are working
# with saveRDS(climate_df, 'climate_df.rds') bring back the simulated
# dat from an .rds file climate_df <- readRDS('climate_df.rds')

```

### 7.8.3 Problem #3: Try something entirely new.

Complete the same steps using data for which you have permission and access. This might be data of your own, from your lab, simulated from an article, or located on an open repository.

### 7.8.4 Grading Rubric

Using the lecture and workflow (chart) as a guide, please work through all the steps listed in the proposed assignment/grading rubric.

Assignment Component	Points Possible	Points Earned
1. Check and, if needed, format and score data	5	_____
2. Report alpha coefficients and average inter-item correlations for the total and subscales	5	_____
3. Produce and interpret corrected item-total correlations for total and subscales, separately	5	_____
4. Produce and interpret correlations between the individual items of a given subscale and the subscale scores of all other subscales	5	_____
5. APA style results section with table	5	_____
6. Explanation to grader	5	_____
<b>Totals</b>	<b>30</b>	_____

## 7.9 Bonus Reel:

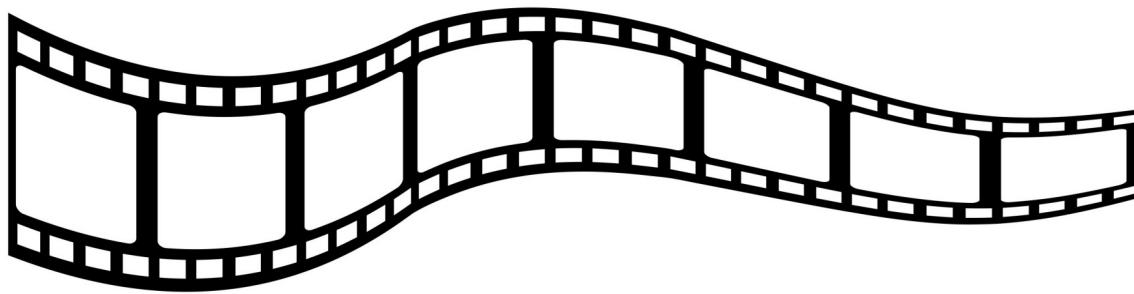


Figure 7.5: Image of a filmstrip

For our interpretation and results, I created the table by manually typing in the results. Since there were only two subscales, this was easy. However, it can be a very useful skill (and prevent typing errors) by leveraging R's capabilities to build a table.

The script below

- Creates a correlation matrix of the items of each scale and correlates them with the “other” subscale, separately for both subscales.
- Extracts the r.drop from each subscale
- Joins (adds more variables) the analyses across the corrected item-total and item-other subscale analyses
- Binds (adds more cases) the two sets of items together

```
Resp_othR <- psych::corr.test(dfSzy[c("negative", "heterosexism", "harassed",
  "Response")]) #Run the correlation of the subscale and the items that are *not* on the su
Resp_othR <- as.data.frame(Resp_othR$r) #extracts the 'r' matrix and makes it a df
```

```

Resp_othR$Items <- c("negative", "heterosexism", "harassed", "Response") #Assigning names to
Resp_othR <- Resp_othR[!Resp_othR$Items == "Response", ] #Removing the subscale score as a a
Resp_othR[, "Stigma"] <- NA #We need a column for this to bind the items later
Resp_othR <- dplyr::select(Resp_othR, Items, Response, Stigma) #All we need is the item name
RESPAlpha <- as.data.frame(RESPAlpha$item.stats) #Grabbing the alpha objet we created earlier
RESPAlpha$Items <- c("cold", "unresponsive", "unsupportive")

Stig_othR <- psych::corr.test(dfSzy[c("cold", "unresponsive", "unsupportive",
    "Stigma")]) #Run the correlation of the subscale and the items that are *not* on the subscale
Stig_othR <- as.data.frame(Stig_othR$r) #extracts the 'r' matrix and makes it a df
Stig_othR$Items <- c("cold", "unresponsive", "unsupportive", "Stigma") #Assigning names to the items
Stig_othR <- Stig_othR[!Stig_othR$Items == "Stigma", ] #Removing the subscale score as a a row
Stig_othR[, "Response"] <- NA #We need a column for this to bind the items later
Stig_othR <- dplyr::select(Stig_othR, Items, Response, Stigma) #All we need is the item name
STIGAlpha <- as.data.frame(STIGAlpha$item.stats) #Grabbing the alpha objet we created earlier
STIGAlpha$Items <- c("negative", "heterosexism", "harassed")

# Combining these four dfs
ResponseStats <- full_join(RESPAlpha, Stig_othR, by = "Items")
ResponseStats$Response <- ResponseStats$r.drop
ResponseStats <- dplyr::select(ResponseStats, Items, Response, Stigma)
StigmaStats <- full_join(STIGAlpha, Resp_othR, by = "Items")
StigmaStats$Stigma <- StigmaStats$r.drop
StigmaStats <- dplyr::select(StigmaStats, Items, Response, Stigma)
ItemAnalyses <- rbind(ResponseStats, StigmaStats)
ItemAnalyses

```

	Items	Response	Stigma
1	cold	0.5041391	0.3258401
2	unresponsive	0.4490390	0.2783617
3	unsupportive	0.4644473	0.2581391
4	negative	0.2874243	0.4401670
5	heterosexism	0.2925477	0.4365623
6	harassed	0.2704033	0.4291150

```

# Writing them to a .csv file allows post-r formatting
write.csv(ItemAnalyses, file = "LGBTQ_Climate_ItemAnalyses.csv", sep = ",",
          row.names = TRUE, col.names = TRUE)

```

## 7.10 Homeworked Example

[Screencast Link](#)

For more information about the data used in this homeworked example, please refer to the description and codebook located at the end of the [introduction](#) in first volume of ReCentering Psych Stats.

As a brief review, this data is part of an IRB-approved study, with consent to use in teaching demonstrations and to be made available to the general public via the open science framework. Hence, it is appropriate to use in this context. You will notice there are student- and teacher- IDs. These numbers are not actual student and teacher IDs, rather they were further re-identified so that they could not be connected to actual people.

Because this is an actual dataset, if you wish to work the problem along with me, you will need to download the [ReC.rds](#) data file from the Worked\_Examples folder in the ReC\_Psychometrics project on the GitHub.

The course evaluation items can be divided into three subscales:

- **Valued by the student** includes the items: ValObjectives, IncrUnderstanding, IncrInterest
- **Traditional pedagogy** includes the items: ClearResponsibilities, EffectiveAnswers, Feedback, ClearOrganization, ClearPresentation
- **Socially responsive pedagogy** includes the items: InclusvClassrm, EquitableEval, MultiPerspectives, DEIIntegration

In this homework focused on validity we will score the total scale and subscales, create a correlation matrix of our scales with a different scale (or item), formally test to see if correlation coefficients are statistically significantly different from each other, conduct a test of incremental validity.

### 7.10.1 Check and, if needed, format and score data

```
big <- readRDS("ReC.rds")
```

With the next code I will create an item-level df with only the items used in the three scales.

```
library(tidyverse)
items <- big %>%
  dplyr::select (ValObjectives, IncrUnderstanding, IncrInterest, ClearResponsibilities, EffectiveAnswers, Feedback, ClearOrganization, ClearPresentation)
```

Next I check the structure of the data.

```
str(items)
```

```
Classes 'data.table' and 'data.frame': 310 obs. of 12 variables:
 $ ValObjectives      : int  5 5 4 4 5 5 5 5 4 5 ...
 $ IncrUnderstanding   : int  2 3 4 3 4 4 5 2 4 5 ...
 $ IncrInterest        : int  5 3 4 2 4 3 5 3 2 5 ...
 $ ClearResponsibilities: int  5 5 4 4 5 4 5 4 4 5 ...
 $ EffectiveAnswers    : int  5 3 5 3 5 3 4 3 2 3 ...
 $ Feedback             : int  5 3 4 2 5 NA 5 4 4 5 ...
 $ ClearOrganization    : int  3 4 3 4 4 4 5 4 4 5 ...
 $ ClearPresentation    : int  4 4 4 2 5 3 4 4 4 5 ...
```

```
$ MultPerspectives      : int  5 5 4 5 5 4 5 5 5 5 ...
$ InclusvClassrm       : int  5 5 5 5 5 4 5 5 4 5 ...
$ DEIintegration        : int  5 5 5 5 5 4 5 5 5 5 ...
$ EquitableEval         : int  5 5 3 5 5 3 5 5 3 5 ...
- attr(*, ".internal.selfref")=<externalptr>
```

### 7.10.2 Report alpha coefficients and average inter-item correlations for the total and subscales

This task is completed in the next section.

### 7.10.3 Produce and interpret corrected item-total correlations for total and subscales, separately

In the lecture, I created baby dfs of the subscales and ran the alphas on those; another option is to create lists of variables (i.e., variable vectors) and use that instead. We can later use those same variable vectors to score the items.

#### 7.10.3.1 All Course Evaluation Items

First, I will calculate the alpha coefficient and inter-item correlations for the total score representing the course evaluation items.

```
psych::alpha(items)
```

```
Reliability analysis
Call: psych::alpha(x = items)
```

	raw_alpha	std.alpha	G6(smc)	average_r	S/N	ase	mean	sd	median_r
	0.92	0.92	0.93	0.49	11	0.0065	4.3	0.61	0.48

```
95% confidence boundaries
      lower alpha upper
Feldt      0.90  0.92  0.93
Duhachek   0.91  0.92  0.93
```

Reliability if an item is dropped:

	raw_alpha	std.alpha	G6(smc)	average_r	S/N	alpha	se	var.r
ValObjectives	0.92	0.92	0.93	0.51	11.3	0.0067	0.016	
IncrUnderstanding	0.91	0.91	0.92	0.49	10.6	0.0070	0.016	
IncrInterest	0.91	0.91	0.92	0.49	10.4	0.0070	0.018	
ClearResponsibilities	0.91	0.91	0.92	0.48	10.0	0.0073	0.015	
EffectiveAnswers	0.91	0.91	0.92	0.48	10.0	0.0074	0.016	
Feedback	0.91	0.91	0.92	0.48	10.3	0.0071	0.018	

ClearOrganization	0.91	0.91	0.92	0.48	10.2	0.0073	0.016
ClearPresentation	0.91	0.91	0.92	0.47	9.7	0.0076	0.015
MultPerspectives	0.91	0.91	0.92	0.48	10.0	0.0073	0.017
InclusvClassrm	0.91	0.91	0.92	0.49	10.6	0.0069	0.018
DEIIntegration	0.92	0.92	0.93	0.52	11.8	0.0063	0.011
EquitableEval	0.91	0.91	0.93	0.49	10.5	0.0070	0.018
med.r							
ValObjectives	0.53						
IncrUnderstanding	0.50						
IncrInterest	0.48						
ClearResponsibilities	0.48						
EffectiveAnswers	0.48						
Feedback	0.48						
ClearOrganization	0.48						
ClearPresentation	0.47						
MultPerspectives	0.47						
InclusvClassrm	0.52						
DEIIntegration	0.53						
EquitableEval	0.48						

## Item statistics

	n	raw.r	std.r	r.cor	r.drop	mean	sd
ValObjectives	309	0.59	0.61	0.55	0.53	4.5	0.61
IncrUnderstanding	309	0.71	0.70	0.67	0.64	4.3	0.82
IncrInterest	308	0.75	0.73	0.71	0.68	3.9	0.99
ClearResponsibilities	307	0.80	0.80	0.79	0.75	4.4	0.82
EffectiveAnswers	308	0.80	0.79	0.78	0.75	4.4	0.83
Feedback	304	0.75	0.75	0.72	0.69	4.2	0.88
ClearOrganization	309	0.79	0.77	0.75	0.72	4.0	1.08
ClearPresentation	309	0.85	0.84	0.83	0.80	4.2	0.92
MultPerspectives	305	0.79	0.80	0.78	0.75	4.4	0.84
InclusvClassrm	301	0.68	0.70	0.67	0.62	4.6	0.68
DEIIntegration	273	0.51	0.53	0.49	0.42	4.5	0.74
EquitableEval	308	0.70	0.72	0.69	0.66	4.6	0.63

## Non missing response frequency for each item

	1	2	3	4	5	miss
ValObjectives	0.00	0.01	0.03	0.39	0.57	0.00
IncrUnderstanding	0.01	0.04	0.07	0.44	0.45	0.00
IncrInterest	0.02	0.09	0.14	0.44	0.31	0.01
ClearResponsibilities	0.01	0.02	0.07	0.31	0.59	0.01
EffectiveAnswers	0.01	0.02	0.08	0.36	0.53	0.01
Feedback	0.01	0.05	0.10	0.39	0.46	0.02
ClearOrganization	0.04	0.07	0.10	0.41	0.38	0.00
ClearPresentation	0.02	0.05	0.07	0.40	0.46	0.00
MultPerspectives	0.02	0.02	0.08	0.33	0.56	0.02
InclusvClassrm	0.01	0.01	0.05	0.23	0.70	0.03
DEIIntegration	0.00	0.01	0.10	0.22	0.67	0.12

```
EquitableEval      0.00 0.01 0.03 0.32 0.63 0.01
```

At the total scale level,  $\alpha = 0.92$ ; the average inter-item correlations are 0.487; and the corrected item-total correlations ( $r.drop$ ) range from 0.42 to 0.80.

To obtain this data at the subscale level I will first create variable vectors.

```
Valued_vars <- c('ValObjectives', 'IncrUnderstanding', 'IncrInterest')
TradPed_vars <- c('ClearResponsibilities', 'EffectiveAnswers', 'Feedback', 'ClearOrganization')
SCRPed_vars <- c('MultPerspectives', 'InclusvClassrm', 'DEIintegration', 'EquitableEval')
```

I can insert these variable vectors into the *psych::alpha()* function to obtain the information.

### 7.10.3.2 Valued-by-Student Subscale

```
psych::alpha(items[, Valued_vars])
```

Reliability analysis

Call: psych::alpha(x = items[, Valued\_vars])

	raw_alpha	std.alpha	G6(smc)	average_r	S/N	ase	mean	sd	median_r
	0.77	0.77	0.71	0.53	3.4	0.02	4.2	0.68	0.48

95% confidence boundaries

lower alpha upper

Feldt 0.72 0.77 0.81

Duhachek 0.73 0.77 0.81

Reliability if an item is dropped:

	raw_alpha	std.alpha	G6(smc)	average_r	S/N	alpha	se	var.r
ValObjectives	0.80	0.81	0.68	0.68	4.3	0.022	NA	
IncrUnderstanding	0.60	0.65	0.48	0.48	1.8	0.040	NA	
IncrInterest	0.59	0.61	0.44	0.44	1.6	0.044	NA	
								med.r
ValObjectives	0.68							
IncrUnderstanding	0.48							
IncrInterest	0.44							

Item statistics

	n	raw.r	std.r	r.cor	r.drop	mean	sd
ValObjectives	309	0.71	0.77	0.55	0.50	4.5	0.61
IncrUnderstanding	309	0.86	0.85	0.76	0.68	4.3	0.82
IncrInterest	308	0.90	0.87	0.79	0.70	3.9	0.99

Non missing response frequency for each item

	1	2	3	4	5	miss
ValObjectives	0.00	0.01	0.03	0.39	0.57	0.00
IncrUnderstanding	0.01	0.04	0.07	0.44	0.45	0.00
IncrInterest	0.02	0.09	0.14	0.44	0.31	0.01

The alpha for Valued-by-the-Student subscale is 0.77. I will start building my homework table as I go along by adding the corrected item-total correlations (i.e., the *r.drop* column).

---

Item-Total Correlations of Items with their Own and Other Subscales

---

Variables	Valued	TradPed	SCRPed
ValObjectives	.50		
IncrUnderstanding	.68		
IncrInterest	.70		
ClearResponsibilities			
EffectiveAnswers			
Feedback			
ClearOrganization			
ClearPresentation			
MultPerspectives			
DEIntegration			
EquitableEval			
Feedback			

---

### 7.10.3.3 Traditional Pedagogy Items

Next I will produce the output for the Traditional Pedagogy subscale.

```
psych::alpha(items[,TradPed_vars])
```

```
Reliability analysis
```

```
Call: psych::alpha(x = items[, TradPed_vars])
```

```
raw_alpha std.alpha G6(smc) average_r S/N      ase mean    sd median_r
 0.89        0.9       0.88       0.64 8.8 0.0094   4.3 0.76       0.65
```

```
95% confidence boundaries
```

```
lower alpha upper
```

```
Feldt     0.87  0.89  0.91
```

```
Duhachek 0.88  0.89  0.91
```

```
Reliability if an item is dropped:
```

```
raw_alpha std.alpha G6(smc) average_r S/N alpha se  var.r
```

ClearResponsibilities	0.86	0.86	0.84	0.62	6.4	0.013	0.0054
EffectiveAnswers	0.87	0.87	0.84	0.63	6.8	0.012	0.0045
Feedback	0.89	0.89	0.87	0.68	8.4	0.010	0.0016
ClearOrganization	0.88	0.88	0.85	0.64	7.2	0.012	0.0044
ClearPresentation	0.86	0.87	0.83	0.62	6.5	0.013	0.0030

med.r

ClearResponsibilities	0.59
EffectiveAnswers	0.65
Feedback	0.69
ClearOrganization	0.66
ClearPresentation	0.62

#### Item statistics

	n	raw.r	std.r	r.cor	r.drop	mean	sd
ClearResponsibilities	307	0.87	0.87	0.84	0.79	4.4	0.82
EffectiveAnswers	308	0.84	0.85	0.81	0.76	4.4	0.83
Feedback	304	0.78	0.79	0.70	0.66	4.2	0.88
ClearOrganization	309	0.85	0.83	0.78	0.74	4.0	1.08
ClearPresentation	309	0.87	0.87	0.83	0.78	4.2	0.92

#### Non missing response frequency for each item

	1	2	3	4	5	miss
ClearResponsibilities	0.01	0.02	0.07	0.31	0.59	0.01
EffectiveAnswers	0.01	0.02	0.08	0.36	0.53	0.01
Feedback	0.01	0.05	0.10	0.39	0.46	0.02
ClearOrganization	0.04	0.07	0.10	0.41	0.38	0.00
ClearPresentation	0.02	0.05	0.07	0.40	0.46	0.00

Traditional Pedagogy  $\alpha = 0.90$ . I retrieve the corrected item-total correlations from the *r.drop* column.

---

#### Item-Total Correlations of Items with their Own and Other Subscales

---

Variables	Valued	TradPed	SCRPed
ValObjectives	<b>.50</b>		
IncrUnderstanding	<b>.67</b>		
IncrInterest	<b>.70</b>		
ClearResponsibilities		<b>.79</b>	
EffectiveAnswers		<b>.76</b>	
Feedback		<b>.66</b>	
ClearOrganization		<b>.74</b>	
ClearPresentation		<b>.78</b>	
MultPerspectives			
DEIntegration			
EquitableEval			

Variables	Valued	TradPed	SCRPed
Feedback			

#### 7.10.3.4 Socially Responsive Pedagogy Items

```
psych::alpha(items[, SCRPed_vars])
```

Reliability analysis

Call: psych::alpha(x = items[, SCRPed\_vars])

raw_alpha	std.alpha	G6(smc)	average_r	S/N	ase	mean	sd	median_r
0.81	0.81	0.78	0.52	4.3	0.017	4.5	0.58	0.54

95% confidence boundaries

lower alpha upper

Feldt	0.77	0.81	0.84
-------	------	------	------

Duhachek	0.77	0.81	0.84
----------	------	------	------

Reliability if an item is dropped:

	raw_alpha	std.alpha	G6(smc)	average_r	S/N	alpha	se	var.r
MultPerspectives	0.73	0.74	0.67	0.49	2.8	0.026	0.0153	
InclusvClassrm	0.74	0.74	0.67	0.49	2.9	0.025	0.0120	
DEIIntegration	0.78	0.78	0.71	0.54	3.6	0.021	0.0044	
EquitableEval	0.78	0.79	0.73	0.56	3.9	0.021	0.0034	

med.r

MultPerspectives	0.47
------------------	------

InclusvClassrm	0.50
----------------	------

DEIIntegration	0.57
----------------	------

EquitableEval	0.57
---------------	------

Item statistics

	n	raw.r	std.r	r.cor	r.drop	mean	sd
MultPerspectives	305	0.85	0.83	0.76	0.68	4.4	0.84
InclusvClassrm	301	0.82	0.83	0.76	0.67	4.6	0.68
DEIIntegration	273	0.78	0.78	0.67	0.59	4.5	0.74
EquitableEval	308	0.75	0.76	0.64	0.58	4.6	0.63

Non missing response frequency for each item

1	2	3	4	5	miss
---	---	---	---	---	------

MultPerspectives	0.02	0.02	0.08	0.33	0.56	0.02
------------------	------	------	------	------	------	------

InclusvClassrm	0.01	0.01	0.05	0.23	0.70	0.03
----------------	------	------	------	------	------	------

DEIIntegration	0.00	0.01	0.10	0.22	0.67	0.12
----------------	------	------	------	------	------	------

EquitableEval	0.00	0.01	0.03	0.32	0.63	0.01
---------------	------	------	------	------	------	------

Alpha for the SCR Pedagogy dimension is .81

Item-Total Correlations of Items with their Own and Other Subscales

Variables	Valued	TradPed	SCRPed
ValObjectives	.50		
IncrUnderstanding	.67		
IncrInterest	.70		
ClearResponsibilities		.79	
EffectiveAnswers		.76	
Feedback		.66	
ClearOrganization		.74	
ClearPresentation		.78	
MultPerspectives			.68
DEIntegration			.67
EquitableEval			.59
Feedback			.58

#### 7.10.4 Produce and interpret correlations between the individual items of a given subscale and the subscale scores of all other subscales

To do this we need to have subscale scores. Conveniently, I can use the variable vectors created in an earlier step. I will score each of the scales if 75% of the items are non-missing.

```
items$Valued <- sjstats::mean_n(items[,Valued_vars], .75)
items$TradPed <- sjstats::mean_n(items[,TradPed_vars], .75)
items$SCRPed <- sjstats::mean_n(items[,SCRPed_vars], .75)
items$Total <- sjstats::mean_n(items, .75)
```

For each subscale, we can produce an apaTables::apa.cor.table for the items-and-OtherScales. Since there are 3 subscales, this is going to get spicy.

##### 7.10.4.1 Valued-by-the-Student Items

First with the traditional pedagogy total scale score.

```
apaTables::apa.cor.table(items[c('ValObjectives', 'IncrUnderstanding', 'IncrInterest', 'TradPed'))
```

Means, standard deviations, and correlations with confidence intervals

Variable	M	SD	1	2	3
1. ValObjectives	4.52	0.61			
2. IncrUnderstanding	4.28	0.82	.44**		
			[.34, .52]		
3. IncrInterest	3.94	0.99	.48**	.68**	
			[.39, .56]	[.62, .74]	
4. TradPed	4.25	0.76	.50**	.62**	.61**
			[.41, .58]	[.55, .68]	[.54, .68]

Note. M and SD are used to represent mean and standard deviation, respectively. Values in square brackets indicate the 95% confidence interval.

The confidence interval is a plausible range of population correlations that could have caused the sample correlation (Cumming, 2014).

\* indicates  $p < .05$ . \*\* indicates  $p < .01$ .

After each analysis, I can update my table with the data representing the correlations with the individual items of one scale with the total scale score included in the correlation matrix. Our hope is that the corrected item-total correlations that we collected above will be stronger than the individual items' correlations with the other scales' total scores.

---

#### Item-Total Correlations of Items with their Own and Other Subscales

---

Variables	Valued	TradPed	SCRPed
ValObjectives	.50	.50	
IncrUnderstanding	.67	.62	
IncrInterest	.70	.61	
ClearResponsibilities		.79	
EffectiveAnswers		.76	
Feedback		.66	
ClearOrganization		.74	
ClearPresentation		.78	
MultPerspectives			.68
DEIntegration			.67
EquitableEval			.59
Feedback			.58

Next, the Valued-by-the-Student Items with socially responsive pedagogy total scale score.

```
apaTables::apa.cor.table(items[c('ValObjectives', 'IncrUnderstanding', 'IncrInterest', 'SCRPed
```

Means, standard deviations, and correlations with confidence intervals

Variable	M	SD	1	2	3
1. ValObjectives	4.52	0.61			
2. IncrUnderstanding	4.28	0.82	.44**		
			[.34, .52]		
3. IncrInterest	3.94	0.99	.48**	.68**	
			[.39, .56]	[.62, .74]	
4. SCRPed	4.52	0.58	.41**	.44**	.54**
			[.32, .50]	[.35, .53]	[.45, .61]

Note. M and SD are used to represent mean and standard deviation, respectively.

Values in square brackets indicate the 95% confidence interval.

The confidence interval is a plausible range of population correlations that could have caused the sample correlation (Cumming, 2014).

\* indicates  $p < .05$ . \*\* indicates  $p < .01$ .

---

#### Item-Total Correlations of Items with their Own and Other Subscales

---

Variables	Valued	TradPed	SCRPed
ValObjectives	.50	.50	.41
IncrUnderstanding	.67	.62	.44
IncrInterest	.70	.61	.54
ClearResponsibilities		.79	
EffectiveAnswers		.76	
Feedback		.66	
ClearOrganization		.74	
ClearPresentation		.78	
MultPerspectives			.68
DEIntegration			.67
EquitableEval			.59
Feedback			.58

### 7.10.5 Traditional Pedagogy Items

First, the traditional pedagogy items with the valued-by-the-student total scale score.

```
apaTables::apa.cor.table(items[c('ClearResponsibilities', 'EffectiveAnswers', 'Feedback', 'ClearOrganization', 'ClearPresentation', 'Valued')])
```

Means, standard deviations, and correlations with confidence intervals

Variable	M	SD	1	2	3
1. ClearResponsibilities	4.44	0.82			
2. EffectiveAnswers	4.36	0.83	.69** [.63, .74]		
3. Feedback	4.24	0.88	.63** [.56, .70]	.58** [.50, .65]	
4. ClearOrganization	4.01	1.08	.67** [.61, .73]	.60** [.52, .67]	.55** [.46, .62]
5. ClearPresentation	4.24	0.92	.69** [.62, .74]	.72** [.66, .77]	.55** [.47, .62]
6. Valued	4.25	0.68	.52** [.43, .60]	.59** [.51, .66]	.49** [.40, .57]
4		5			

.70\*\*  
[.63, .75]

.63\*\* .69\*\*  
[.55, .69] [.63, .75]

Note. M and SD are used to represent mean and standard deviation, respectively. Values in square brackets indicate the 95% confidence interval. The confidence interval is a plausible range of population correlations that could have caused the sample correlation (Cumming, 2014). \* indicates  $p < .05$ . \*\* indicates  $p < .01$ .

---

Item-Total Correlations of Items with their Own and Other Subscales

---

Variables	Valued	TradPed	SCRPed
ValObjectives	<b>.50</b>	.50	.41
IncrUnderstanding	<b>.67</b>	.62	.44
IncrInterest	<b>.70</b>	.61	.54
ClearResponsibilities	.52	<b>.79</b>	
EffectiveAnswers	.59	<b>.76</b>	
Feedback	.49	<b>.66</b>	
ClearOrganization	.63	<b>.74</b>	
ClearPresentation	.69	<b>.78</b>	
MultPerspectives			<b>.68</b>
DEIntegration			<b>.67</b>
EquitableEval			<b>.59</b>
Feedback			<b>.58</b>

---

Next, the traditional pedagogy items with the socially responsive pedagogy total scale score.

```
apaTables::apa.cor.table(items[c('ClearResponsibilities', 'EffectiveAnswers', 'Feedback', 'ClearOrganization', 'MultPerspectives', 'DEIntegration', 'EquitableEval', 'IncrInterest', 'IncrUnderstanding', 'ClearPresentation', 'ValObjectives')])
```

Means, standard deviations, and correlations with confidence intervals

Variable	M	SD	1	2	3
1. ClearResponsibilities	4.44	0.82			
2. EffectiveAnswers	4.36	0.83	.69**		
			[.63, .74]		
3. Feedback	4.24	0.88	.63**	.58**	
			[.56, .70]	[.50, .65]	
4. ClearOrganization	4.01	1.08	.67**	.60**	.55**
			[.61, .73]	[.52, .67]	[.46, .62]

5. ClearPresentation	4.24	0.92	.69**	.72**	.55**
			[.62, .74]	[.66, .77]	[.47, .62]
6. SCRPed	4.52	0.58	.64**	.60**	.62**
			[.56, .70]	[.52, .67]	[.55, .69]

4                5

.70\*\*  
 [.63, .75]

.51\*\*        .62\*\*  
 [.43, .59]    [.54, .68]

Note. M and SD are used to represent mean and standard deviation, respectively.  
 Values in square brackets indicate the 95% confidence interval.

The confidence interval is a plausible range of population correlations  
 that could have caused the sample correlation (Cumming, 2014).

\* indicates  $p < .05$ . \*\* indicates  $p < .01$ .

---

 Item-Total Correlations of Items with their Own and Other Subscales
 

---

Variables	Valued	TradPed	SCRPed
ValObjectives	.50	.50	.41
IncrUnderstanding	.67	.62	.44
IncrInterest	.70	.61	.54
ClearResponsibilities	.52	.79	.64
EffectiveAnswers	.59	.76	.60
Feedback	.49	.66	.62
ClearOrganization	.63	.74	.51
ClearPresentation	.69	.78	.62
MultPerspectives			.68

Variables	Valued	TradPed	SCRPed
DEIintegration			.67
EquitableEval			.59
Feedback			.58

### 7.10.5.1 Socially Responsive Pedagogy Items

First, the socially responsive pedagogy items with the valued-by-the-student total scale score.

```
apaTables::apa.cor.table(items[c('MultPerspectives', 'InclusvClassrm', 'DEIintegration', 'Equitab'))
```

Means, standard deviations, and correlations with confidence intervals

Variable	M	SD	1	2	3	4
1. MultPerspectives	4.39	0.84				
2. InclusvClassrm	4.61	0.68	.57**			
			[.49, .64]			
3. DEIintegration	4.53	0.74	.50**	.62**		
			[.41, .59]	[.54, .69]		
4. EquitableEval	4.57	0.63	.59**	.47**	.37**	
			[.51, .66]	[.37, .55]	[.27, .47]	
5. Valued	4.25	0.68	.54**	.48**	.29**	.46**
			[.45, .61]	[.38, .56]	[.18, .40]	[.37, .54]

Note. M and SD are used to represent mean and standard deviation, respectively.  
Values in square brackets indicate the 95% confidence interval.

The confidence interval is a plausible range of population correlations  
that could have caused the sample correlation (Cumming, 2014).

\* indicates  $p < .05$ . \*\* indicates  $p < .01$ .

---

Item-Total Correlations of Items with their Own and Other Subscales

---

Variables	Valued	TradPed	SCRPed
ValObjectives	.50	.50	.41
IncrUnderstanding	.67	.62	.44
IncrInterest	.70	.61	.54
ClearResponsibilities	.52	.79	.64
EffectiveAnswers	.59	.76	.60
Feedback	.49	.66	.62
ClearOrganization	.63	.74	.51
ClearPresentation	.69	.78	.62
MultPerspectives	.54		.68
DEIintegration	.48		.67
EquitableEval	.29		.59
Feedback	.46		.58

Next, the socially responsive pedagogy items with the traditional pedagogy total scale score.

```
apaTables::apa.cor.table(items[c('MultPerspectives', 'InclusvClassrm', 'DEIintegration', 'Equita
```

Means, standard deviations, and correlations with confidence intervals

Variable	M	SD	1	2	3	4
1. MultPerspectives	4.39	0.84				
2. InclusvClassrm	4.61	0.68	.57**			
				[.49, .64]		
3. DEIintegration	4.53	0.74	.50**	.62**		
				[.41, .59]	[.54, .69]	
4. EquitableEval	4.57	0.63	.59**	.47**	.37**	
				[.51, .66]	[.37, .55]	[.27, .47]
5. TradPed	4.25	0.76	.71**	.54**	.34**	.65**
				[.64, .76]	[.46, .62]	[.23, .44]
						[.58, .71]

Note. M and SD are used to represent mean and standard deviation, respectively.  
Values in square brackets indicate the 95% confidence interval.

The confidence interval is a plausible range of population correlations  
that could have caused the sample correlation (Cumming, 2014).

\* indicates  $p < .05$ . \*\* indicates  $p < .01$ .

---

 Item-Total Correlations of Items with their Own and Other Subscales
 

---

Variables	Valued	TradPed	SCRPed
ValObjectives	.50	.50	.41
IncrUnderstanding	.67	.62	.44
IncrInterest	.70	.61	.54
ClearResponsibilities	.52	.79	.64
EffectiveAnswers	.59	.76	.60
Feedback	.49	.66	.62
ClearOrganization	.63	.74	.51
ClearPresentation	.69	.78	.62
MultPerspectives	.54	.71	.68
InclusvClassrm	.48	.54	.67
DEIintegration	.29	.34	.59
EquitableEval	.46	.65	.58

#### 7.10.6 APA style results section with table

- Brief description of each step
- Brief instructions for interpreting results
- Presentation of results

Item analyses were conducted on the 12 course evaluation items that were selected to represent the Valued-by-the-Student, Traditional Pedagogy, and Socially Responsive Pedagogy subscales. To assess the within-scale convergent and discriminant validity of these three subscales, each item was correlated with its own scale (with the item removed) and with the other course evaluation scales (see Table 1). For the Valued and Traditional Pedagogy dimensions, items were more highly correlated with their own scale than with the other scale. For the SCRPed subscale, two items (multiple perspectives, equitable evaluations) had higher correlations with Traditional Pedagogy than with their hypothesized subscale (Socially Responsive Pedagogy). Coefficient alphas were .77, .90, .81, and .92 for the Valued-by-the-Student, Traditional Pedagogy, Socially Responsive, and total-scale scores, respectively. We conclude that more work is needed to create distinct and stable subscales.

#### 7.10.7 Explanation to grader

# **EXPLORATORY *FACTOR* ANALYSIS**

The next two lessons are devoted to exploratory *factor* analysis. The two approaches are principal components analysis (PCA) and principal axis factoring (PAF). In truth, only PAF is considered *factor* analysis. I will explain why in the lesson.

These approaches are loosely termed *exploratory* because the statistical process (not the researcher) produces the factor (think scale or subscale) and identifies which items belong to it. This is contrasted with *confirmatory* approaches (which use structural equation modeling) where the researcher assigns items to factors and analyzes the goodness of fit.



# Chapter 8

# Principal Components Analysis

## [Screencasted Lecture Link](#)

In this lesson on principal components analysis (PCA) I provide an introduction to the exploratory factor analysis (EFA) arena. We will review the theoretical and technical aspects of PCA, we will work through a research vignette, and then consider the relationship of PCA to item analysis and reliability coefficients.

Please note, although PCA is frequently grouped into EFA techniques, it is *exploratory*, but it is not *factor analysis*. We'll discuss the difference in the lecture.

## 8.1 Navigating this Lesson

There are about two hours of lecture. If you work through the materials with me, I would be plan for an additional hour-and-a-half.

While the majority of R objects and data you will need are created within the R script that sources the chapter, occasionally there are some that cannot be created from within the R framework. Additionally, sometimes links fail. All original materials are provided at the [GitHub site](#) that hosts the book. More detailed guidelines for ways to access all these materials are provided in the OER's [introduction](#)

### 8.1.1 Learning Objectives

Focusing on this week's materials, make sure you can:

- Distinguish between PCA and PAF on several levels:
  - which path diagram represents each best, and
  - keywords associated with each: factor loadings, linear components, describe versus explain.
- Recognize/define an identity matrix – what test would you use to diagnose it?
- Recognize/define multicollinearity and singularity – what test would you use to diagnose it?

- Describe the pattern of “loadings” (i.e., the relative weights of an item on its own scale compared to other scales) that supports the structure of the instrument.
- Compare the results from item analysis and PCA.

### 8.1.2 Planning for Practice

In each of these lessons I provide suggestions for practice that allow you to select one or more problems that are graded in difficulty. The least complex is to change the random seed in the research and rework the problem demonstrated in the lesson. The results *should* map onto the ones obtained in the lecture.

The second option involves utilizing one of the simulated datasets available in this OER. The [last lesson](#) in the OER contains three simulations that could be used for all of the statistics-based practice suggestions. Especially if you started with one of these examples in an earlier lesson, I highly recommend you continue with that.

Alternatively, Keum et al.’s Gendered Racial Microaggressions Scale for Asian American Women [2018] will be used in the lessons on confirmatory factor analysis and Conover et al.’s [2017] Ableist Microaggressions Scale is used in the lesson on invariance testing. Both of these would be suitable for the PCA and PAF homework assignments.

As a third option, you are welcome to use data to which you have access and is suitable for PCA. These could include other vignettes from this OER, other simulated data, or your own data (presuming you have permission to use it). In any case, please plan to:

- Properly format and prepare the data.
- Conduct diagnostic tests to determine the suitability of the data for PCA.
- Conducting tests to guide the decisions about number of components to extract.
- Conducting orthogonal and oblique extractions (at least two each with different numbers of components).
- Selecting one solution and preparing an APA style results section (with table and figure).
- Compare your results in light of any other psychometrics lessons where you have used this data.

### 8.1.3 Readings & Resources

In preparing this chapter, I drew heavily from the following resource(s). Other resources are cited (when possible, linked) in the text with complete citations in the reference list.

- Revelle, William. (n.d.). Chapter 6: Constructs, components, and factor models. In *An introduction to psychometric theory with applications in R*. Retrieved from <https://personality-project.org/r/book/#chapter6>
  - pp. 145 to 150 (we’ll continue with the rest in the next lecture). Stop at “6.2 Exploratory Factor Analysis.”
  - A simultaneously theoretical review of psychometric theory while working with R and data to understand the concepts.
- Revelle, W. (2019). *How To: Use the psych package for Factor Analysis and data reduction*.

- pp. 13 through 24 provide technical information about what we are doing
- Dekay, Nicole (2021). Quick Reference Guide: The statistics for psychometrics <https://www.humanalysts.com/quick-reference-guide-the-statistics-for-psychometrics>
- Lewis, J. A., & Neville, H. A. (2015). Construction and initial validation of the Gendered Racial Microaggressions Scale for Black Women. *Journal of Counseling Psychology*, 62(2), 289–302. <https://doi.org/10.1037/cou0000062>
  - Our research vignette for this lesson.

#### 8.1.4 Packages

The packages used in this lesson are embedded in this code. When the hashtags are removed, the script below will (a) check to see if the following packages are installed on your computer and, if not (b) install them.

```
# will install the package if not already installed
# if(!require(psych)){install.packages('psych')}
# if(!require(tidyverse)){install.packages('tidyverse')}
# if(!require(MASS)){install.packages('MASS')}
# if(!require(sjstats)){install.packages('sjstats')}
# if(!require(apaTables)){install.packages('apaTables')}
# if(!require(qualtrics)){install.packages('qualtrics')}
```

## 8.2 Exploratory Principal Components Analysis

The psychometric version of *parsimony* is seen in our attempt to *describe* (components) or to *explain* (factors) in the relationships between many observed variables in terms of a more limited set of components, latent factors, or dimensions.

That is, we are trying to:

- understand the structure of a set of variables,
- construct a questionnaire to measure an underlying latent variable, and
- reduce a data set to a more manageable size (e.g., representing bundles of items as subscale scores) while retaining as much of the information as possible

### 8.2.1 Some Framing Ideas (in very lay terms)

*Exploratory* versus *confirmatory* factor analysis.

- Both exploratory and confirmatory approaches to components/factor analysis are used in scale construction. Think of “scales” as being interchangeable with “factors” and “components.”
  - That said, “factors” and “components” are not interchangeable terms.

- **Exploratory:** Even though we may have an a priori model in mind, we *explore* the structure of the items by using diagnostics (KMO, Barlett's, determinant), factor extraction, and rotation to determine the number of scales (i.e., components or factors) that exist within the raw data or correlation matrix. The algorithms (including matrix algebra) determine the relationship of each item to its respective scales (i.e., components or factors).
- **Confirmatory:** Starting with an a priori theory, we specify the structure (i.e., number and levels of factors) and which items belong to factors. We use structural equation modeling as the framework. We evaluate the quality of the model with a number of fit indices.

Within the *exploratory* category we will focus on two further distinctions (there are even more). The first is principal components analysis (PCA). The second is principal axis factoring (PAF). PAF is one of the approaches that is commonly termed “exploratory factor analysis” (EFA). In this first lesson we focus on the differences between PCA and EFA.

- **Option #1/Component model:** PCA approximates the correlation matrix in terms of the product of components where each is a weighted linear sum of the variables. In the figure below, note how the arrows in the components analysis (a *path* model) point from variables to the component. Perhaps an oversimplification, think of each of these as a predictor variable contributing to an outcome.
- **Option #2/Factor model:** EFA (and in the next lesson, PAF/principal axis factoring) approximates the correlation matrix by the product of the two factors; this approach presumes that the factors are the causes (rather than as consequences). In the figure below, note how the arrows in the factor analysis model (a *structural* model) point from latent variable (or factor) to the observed variables (items). Factor analysis has been termed *causal modeling* because the latent variables are theorized to cause the responses to the individual items. There are other popular approaches, including parallel analysis (which is what the authors used in this lesson’s research vignette).

Well-crafted figures provide important clues to the analyses. In structural models, rectangles and squares indicate the presence of *observed* (also called *manifest*) variables. These are variables that have a column in the dataset. In our particular case, they are the responses to the 25 items in the GRMS.

Circles or ovals represent latent variables or factors. These were never raw data, but are composed of the relations of variables that were collected. They are more complex than mean or sum scores. Rather, they represent what the variables that represent them share in common.

Our focus today is on the principal component analysis (PCA) approach to scale construction.

### 8.3 PCA Workflow

Below is a screenshot of the workflow. The original document is located in the [GitHub site](#) that hosts the ReCentering Psych Stats: Psychometrics OER.

Steps in the process include:

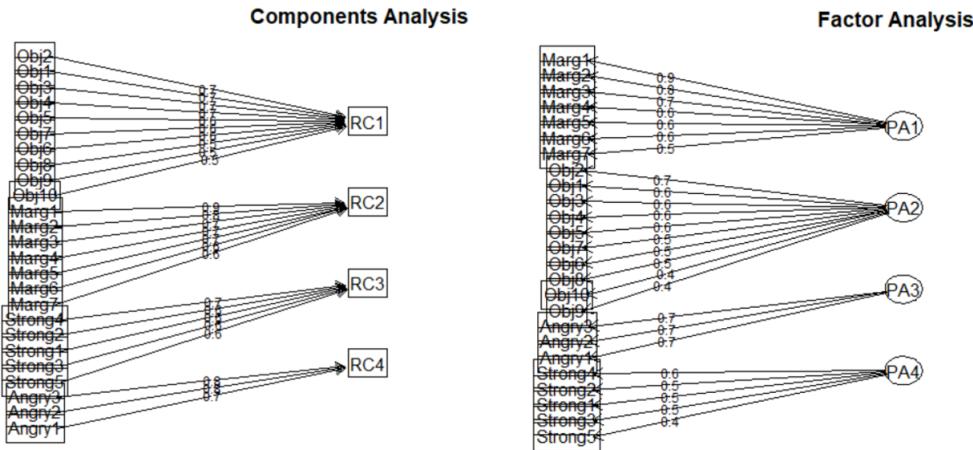


Figure 8.1: Comparison of path models for PCA and EFA for our research vignette

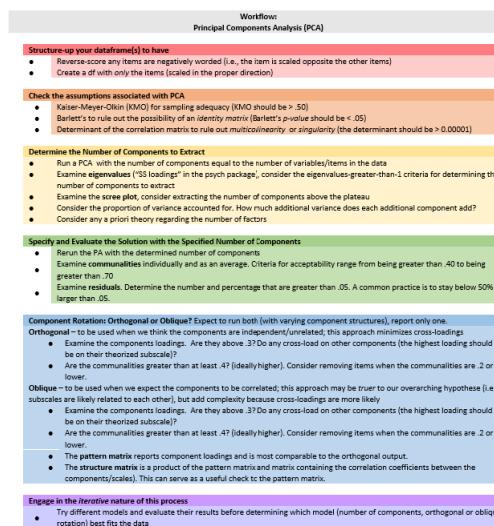


Figure 8.2: Image of the workflow for PCA

- Creating an *items only* dataframe where any items are scaled in the same direction (e.g., negatively worded items are reverse scored).
- Conducting tests that assess the statistical assumptions of PCA to ensure that the data is appropriate for PCA.
- Determining the number of components (think “subsubscales”) to extract.
- Conducting the component extraction – this process will likely occur iteratively,
  - exploring orthogonal (uncorrelated/independent) and oblique (correlated) components, and
  - changing the number of components to extract

Because the intended audience for the ReCentering Psych Stats OER is the scientist-practitioner-advocate, this lesson focuses on the workflow and decisions. As you might guess, the details of PCA can be quite complex. Some important notions to consider that may not be obvious from lesson, are these:

- The values of component loadings are directly related to the correlation (similarly, the covariance) matrix between the items.
  - Although I do not explain this in detail, nearly every analytic step attempts to convey this notion by presenting equivalent analytic options using the raw data and correlation matrix.
- PCA is about *dimension reduction* – our goal is fewer components (i.e., subscales) than there are items.
  - In this lesson’s vignette there are 25 items on the scale, and we will end up with 4 subscales.
- Principal component analysis is *exploratory*, but it is not “factor analysis.”
- Matrix algebra (e.g., using the transpose of a matrix, multiplying matrices together) plays a critical role in the analytic solution.

## 8.4 Research Vignette

This lesson’s research vignette emerges from Lewis and Neville’s Gendered Racial Microaggressions Scale for Black Women [2015]. The article reports on two separate studies that comprised the development, refinement, and psychometric evaluation of two parallel versions (stress appraisal, frequency) of the scale. Below, I simulate data from the final construction of the stress appraisal version as the basis of the lecture. Items were on a 6-point Likert scale ranging from 0 (*not at all stressful*) to 5 (*extremely stressful*).

Lewis and Neville [2015] reported support for a total scale score (25 items) and four subscales. Below, I list the four subscales along with the items and their abbreviation. At the outset, let me provide a content advisory. For those who hold this particular identity (or related identities) the content in the items may be upsetting. In other lessons, I often provide a variable name that gives an indication of the primary content of the item. In the case of the GRMS, I will simply provide an abbreviation of the subscale name and its respective item number. This will allow us to easily inspect the alignment of the item with its intended factor, and hopefully minimize discomfort.

If you are not a member of this particular identity, I encourage you to learn about these microaggressions by reading the article in its entirety. Please do not ask members of this group to explain why these microaggressions are harmful or ask if they have encountered them. The four factors, number of items, and sample item are as follows:

- Assumptions of Beauty and Sexual Objectification (10 items)
  - Unattractive because of size of butt (Obj1)
  - Negative comments about size of facial features (Obj2)
  - Imitated the way they think Black women speak (Obj3)
  - Someone made me feel unattractive (Obj4)
  - Negative comment about skin tone (Obj5)
  - Someone assumed I speak a certain way (Obj6)
  - Objectified me based on physical features(Obj7)
  - Someone assumed I have a certain body type (Obj8; stress only)
  - Made a sexually inappropriate comment (Obj9)
  - Negative comments about my hair when natural (Obj10)
  - Assumed I was sexually promiscuous (frequency only; not used in this simulation)
- Silenced and Marginalized (7 items)
  - I have felt unheard (Marg1)
  - My comments have been ignored (Marg2)
  - Someone challenged my authority (Marg3)
  - I have been disrespected in workplace (Marg4)
  - Someone has tried to “put me in my place” (Marg5)
  - Felt excluded from networking opportunities (Marg6)
  - Assumed I did not have much to contribute to the conversation (Marg7)
- Strong Black Woman Stereotype (5 items)
  - Someone assumed I was sassy and straightforward (Str1; stress only)
  - I have been told that I am too independent (Str2)
  - Someone made me feel exotic as a Black woman (Str2; stress only)
  - I have been told that I am too assertive
  - Assumed to be a strong Black woman
- Angry Black Woman Stereotype (3 items)
  - Someone has told me to calm down (Ang1)
  - Perceived to be “angry Black woman” (Ang2)
  - Someone accused me of being angry when speaking calm (Ang3)

Three additional scales were reported in the Lewis and Neville article [2015]. Because (a) the focus of this lesson is on exploratory factor analytic approaches and, therefore, only requires item-level data for the scale, and (b) the article does not include correlations between the subscales/scales of all involved measures, I only simulated item-level data for the GRMS items.

Below, I walk through the data simulation. This is not an essential portion of the lesson, but I will lecture it in case you are interested. None of the items are negatively worded (relative to the other items), so there is no need to reverse-score any items.

Simulating the data involved using factor loadings, means, standard deviations, and correlations between the scales. Because the simulation will produce “out-of-bounds” values, the code below rescales the scores into the range of the Likert-type scaling and rounds them to whole values.

```
# Entering the intercorrelations means and standard deviations from
# the journal article

LewisGRMS_generating_model <- "
  #measurement model
  Objectification =~ .69*Obj1 + .69*Obj2 + .60*Obj3 + .59*Obj4 + .55*Obj5 + .55*Obj6 + .
  Marginalized =~ .93*Marg1 + .81*Marg2 + .69*Marg3 + .67*Marg4 + .61*Marg5 + .58*Marg6 + .
  Strong =~ .59*Str1 + .55*Str2 + .54*Str3 + .54*Str4 + .51*Str5
  Angry =~ .70*Ang1 + .69*Ang2 + .68*Ang3

  #Means
  Objectification ~ 1.85*1
  Marginalized ~ 2.67*1
  Strong ~ 1.61*1
  Angry ~ 2.29*1

  #Correlations
  Objectification ~~ .63*Marginalized
  Objectification ~~ .66*Strong
  Objectification ~~ .51*Angry

  Marginalized ~~ .59*Strong
  Marginalized ~~ .62*Angry

  Strong ~~ .61*Angry

  "

set.seed(240311)
dfGRMS <- lavaan:::simulateData(model = LewisGRMS_generating_model, model.type = "sem",
  meanstructure = T, sample.nobs = 259, standardized = FALSE)

# used to retrieve column indices used in the rescaling script below
col_index <- as.data.frame(colnames(dfGRMS))

# The code below loops through each column of the dataframe and
# assigns the scaling accordingly Rows 1 thru 26 are the GRMS items

for (i in 1:ncol(dfGRMS)) {
  if (i >= 1 & i <= 25) {
    dfGRMS[, i] <- scales::rescale(dfGRMS[, i], c(0, 5))
  }
}
```

```
# rounding to integers so that the data resembles that which was
# collected
library(tidyverse)
dfGRMS <- dfGRMS %>%
  round(0)

# quick check of my work psych::describe(dfGRMS)
```

The optional script below will let you save the simulated data to your computing environment as either an .rds object (preserves any formatting you might do) or a .csv file (think “Excel lite”).

An .rds file preserves all formatting to variables prior to the export and re-import. For the purpose of this chapter, you don’t need to do either. That is, you can re-simulate the data each time you work the problem.

```
# to save the df as an .rds (think 'R object') file on your computer;
# it should save in the same file as the .rmd file you are working
# with saveRDS(dfGRMS, 'dfGRMS.rds') bring back the simulated dat
# from an .rds file dfGRMS <- readRDS('dfGRMS.rds')
```

If you save the .csv file and bring it back in, you will lose any formatting (e.g., ordered factors will be interpreted as character variables).

```
# write the simulated data as a .csv write.table(dfGRMS,
# file='dfGRMS.csv', sep=',', col.names=TRUE, row.names=FALSE) bring
# back the simulated dat from a .csv file dfGRMS <- read.csv
# ('dfGRMS.csv', header = TRUE)
```

Before moving on, I want to acknowledge that (at their first drafting), I try to select research vignettes that have been published within the prior 5 years. With a publication date of 2015, this article clearly falls outside that range. I have continued to include it because (a) the scholarship is superior – especially as the measure captures an intersectional identity, (b) the article has been a model for research that follows (e.g., Keum et al’s [2018] Gendered Racial Microaggression Scale for Asian American Women), and (c) there is often a time lag between the initial publication of a psychometric scale and its use. A key reason I have retained the GRMS as a psychometrics research vignette is that in [ReCentering Psych Stats: Multivariate Modeling](#), GRMS scales are used in a couple of more recently published research vignettes.

## 8.5 Working the Vignette

Below we will create a correlation matrix of our items. Whether we are conducting PCA or PAF, the *dimension-reduction* we are looking for clusters of correlated items in the *R*-matrix. Essentially, these are [\[Field, 2012\]](#):

- Statistical entities that can be plotted as classification axes where coordinates of variables along each axis represent the strength of the relationship between that variable to each component (later, factor).

- Mathematical equations, resembling regression equations, where each variable is represented according to its relative weight.

PCA in particular establishes which linear components exist within the data and how a particular variable might contribute to that component.

Below is the correlation matrix of our items. I have saved it as an object so that I can show you that PCA (and later, PAF) can also be conducted with just correlation data. It would be quite a daunting exercise to visually inspect this and manually cluster the correlations of items.

```
GRMSmatrix <- cor(dfGRMS) #correlation matrix created and saved as object
round(GRMSmatrix, 2)
```

	Obj1	Obj2	Obj3	Obj4	Obj5	Obj6	Obj7	Obj8	Obj9	Obj10	Marg1	Marg2	Marg3
Obj1	1.00	0.35	0.25	0.27	0.28	0.25	0.28	0.35	0.15	0.24	0.19	0.25	0.17
Obj2	0.35	1.00	0.31	0.25	0.27	0.23	0.31	0.28	0.26	0.24	0.22	0.21	0.25
Obj3	0.25	0.31	1.00	0.24	0.28	0.28	0.20	0.25	0.21	0.22	0.17	0.23	0.17
Obj4	0.27	0.25	0.24	1.00	0.39	0.23	0.28	0.30	0.26	0.28	0.22	0.18	0.14
Obj5	0.28	0.27	0.28	0.39	1.00	0.15	0.18	0.29	0.25	0.20	0.17	0.20	0.23
Obj6	0.25	0.23	0.28	0.23	0.15	1.00	0.20	0.14	0.21	0.12	0.10	0.14	0.05
Obj7	0.28	0.31	0.20	0.28	0.18	0.20	1.00	0.31	0.19	0.28	0.30	0.21	0.20
Obj8	0.35	0.28	0.25	0.30	0.29	0.14	0.31	1.00	0.19	0.23	0.27	0.14	0.14
Obj9	0.15	0.26	0.21	0.26	0.25	0.21	0.19	0.19	1.00	0.20	0.10	0.12	0.21
Obj10	0.24	0.24	0.22	0.28	0.20	0.12	0.28	0.23	0.20	1.00	0.09	0.12	0.17
Marg1	0.19	0.22	0.17	0.22	0.17	0.10	0.30	0.27	0.10	0.09	1.00	0.43	0.41
Marg2	0.25	0.21	0.23	0.18	0.20	0.14	0.21	0.14	0.12	0.12	0.43	1.00	0.35
Marg3	0.17	0.25	0.17	0.14	0.23	0.05	0.20	0.14	0.21	0.17	0.41	0.35	1.00
Marg4	0.19	0.18	0.24	0.26	0.20	0.10	0.25	0.24	0.07	0.12	0.38	0.23	0.32
Marg5	0.17	0.22	0.21	0.27	0.25	0.16	0.23	0.19	0.19	0.11	0.41	0.40	0.25
Marg6	0.18	0.27	0.16	0.23	0.22	0.26	0.28	0.26	0.15	0.26	0.35	0.27	0.25
Marg7	0.13	0.19	0.14	0.19	0.06	0.17	0.16	0.14	0.10	0.11	0.31	0.33	0.20
Str1	0.22	0.18	0.14	0.06	0.23	0.07	0.25	0.17	0.19	0.10	0.19	0.25	0.20
Str2	0.19	0.18	0.19	0.19	0.12	0.15	0.13	0.06	0.18	0.19	0.12	0.18	0.17
Str3	0.10	0.09	0.09	0.08	0.11	0.09	0.19	0.05	0.12	0.10	0.13	0.18	0.10
Str4	0.09	0.14	0.18	0.15	0.12	0.08	0.07	0.13	0.05	0.02	0.08	0.12	0.08
Str5	0.20	0.15	0.15	0.08	0.19	0.11	0.15	0.04	0.07	0.09	0.10	0.23	0.12
Ang1	0.06	0.07	0.07	0.09	0.12	0.04	0.15	0.07	0.17	0.06	0.16	0.23	0.18
Ang2	0.06	0.15	0.08	0.06	0.09	0.20	0.13	-0.03	0.00	0.14	0.17	0.19	0.19
Ang3	0.21	0.13	0.11	0.14	0.11	0.16	0.23	0.07	0.06	0.08	0.28	0.28	0.11
Marg4	Marg5	Marg6	Marg7	Str1	Str2	Str3	Str4	Str5	Ang1	Ang2	Ang3		
Obj1	0.19	0.17	0.18	0.13	0.22	0.19	0.10	0.09	0.20	0.06	0.06	0.21	
Obj2	0.18	0.22	0.27	0.19	0.18	0.18	0.09	0.14	0.15	0.07	0.15	0.13	
Obj3	0.24	0.21	0.16	0.14	0.14	0.19	0.09	0.18	0.15	0.07	0.08	0.11	
Obj4	0.26	0.27	0.23	0.19	0.06	0.19	0.08	0.15	0.08	0.09	0.06	0.14	
Obj5	0.20	0.25	0.22	0.06	0.23	0.12	0.11	0.12	0.19	0.12	0.09	0.11	
Obj6	0.10	0.16	0.26	0.17	0.07	0.15	0.09	0.08	0.11	0.04	0.20	0.16	
Obj7	0.25	0.23	0.28	0.16	0.25	0.13	0.19	0.07	0.15	0.15	0.13	0.23	
Obj8	0.24	0.19	0.26	0.14	0.17	0.06	0.05	0.13	0.04	0.07	-0.03	0.07	

Obj9	0.07	0.19	0.15	0.10	0.19	0.18	0.12	0.05	0.07	0.17	0.00	0.06
Obj10	0.12	0.11	0.26	0.11	0.10	0.19	0.10	0.02	0.09	0.06	0.14	0.08
Marg1	0.38	0.41	0.35	0.31	0.19	0.12	0.13	0.08	0.10	0.16	0.17	0.28
Marg2	0.23	0.40	0.27	0.33	0.25	0.18	0.18	0.12	0.23	0.23	0.19	0.28
Marg3	0.32	0.25	0.25	0.20	0.20	0.17	0.10	0.08	0.12	0.18	0.19	0.11
Marg4	1.00	0.30	0.26	0.16	0.10	0.21	0.05	0.06	0.03	0.12	0.22	0.17
Marg5	0.30	1.00	0.29	0.28	0.16	0.13	0.16	0.14	0.18	0.12	0.14	0.21
Marg6	0.26	0.29	1.00	0.20	0.13	0.18	0.15	0.13	0.08	0.11	0.21	0.12
Marg7	0.16	0.28	0.20	1.00	0.14	0.05	0.04	0.02	0.12	0.17	0.13	0.09
Str1	0.10	0.16	0.13	0.14	1.00	0.21	0.30	0.23	0.23	0.18	0.05	0.10
Str2	0.21	0.13	0.18	0.05	0.21	1.00	0.20	0.20	0.12	0.16	0.12	0.16
Str3	0.05	0.16	0.15	0.04	0.30	0.20	1.00	0.27	0.18	0.20	0.07	0.15
Str4	0.06	0.14	0.13	0.02	0.23	0.20	0.27	1.00	0.12	0.15	0.03	0.02
Str5	0.03	0.18	0.08	0.12	0.23	0.12	0.18	0.12	1.00	0.22	0.15	0.11
Ang1	0.12	0.12	0.11	0.17	0.18	0.16	0.20	0.15	0.22	1.00	0.24	0.23
Ang2	0.22	0.14	0.21	0.13	0.05	0.12	0.07	0.03	0.15	0.24	1.00	0.25
Ang3	0.17	0.21	0.12	0.09	0.10	0.16	0.15	0.02	0.11	0.23	0.25	1.00

This correlation matrix is so big that you might wish to write code so that you can examine it in sections.

```
# round(GRMSmatrix[,1:8], 2) round(GRMSmatrix[,9:16], 2)
# round(GRMSmatrix[,17:25], 2)
```

With component and factor analytic procedures we can analyze the data with either raw data or correlation matrix. Using the correlation matrix helps us perceive how this is a *structural* analysis. That is, we are trying to see if our more parsimonious extraction (i.e., our *dimension reduction*) reproduces this original correlation matrix. In each of the analyses I will include code for running the analyses with raw data and the *r*-matrix.

### 8.5.1 Three Diagnostic Tests to Evaluate the Appropriateness of the Data for Component-or-Factor Analysis

Below is a snip from the workflow to remind us of where we are in the steps to PCA.

Check the assumptions associated with PCA  
 • Kaiser-Meyer-O'Brien (KMO) for sampling adequacy (KMO should be > .50)  
 • Bartlett's to rule out the possibility of an identity matrix (Bartlett's p-value should be < .05)  
 • Determinant of the correlation matrix to rule out multicollinearity or singularity (the determinant should be > 0.00001)

Figure 8.3: Image of an excerpt from the workflow

#### 8.5.1.1 Is my sample adequate for PCA?

There have been a number of generic guidelines (some supported with empirical evidence, some not) about “how big” the sample size should be:

- 10-15 participants per variable

- 10 times as many participants as variables (Nunnally, 1978)
- 5 and 10 participants per variable up to 300 (Kass & Tinsley, 1979)
- 300 (Tabachnick & Fidell, 2007)
- 1000 = excellent, 300 = good, 100 = poor (Comrey & Lee, 1992)

Of course it is more complicated. Monte Carlo studies have shown that:

- if factor loadings are large (~.6), the solution is reliable regardless of size
- if communalities are large (~.6), relatively small samples (~100) are sufficient, but when they are lower (well below .5), then larger samples (>500 are indicated).

The **Kaiser-Meyer-Olkin** index (KMO) is an index of *sampling adequacy* that can be used with the actual sample to let us know if the sample size is sufficient relative to the statistical characteristics of the data. If the KMO is below the recommendations, we should probably collect more data to see if it can achieve a satisfactory value.

Kaiser's 1974 recommendations were:

- bare minimum of .5
- values between .5 and .7 as mediocre
- values between .7 and .8 as good
- values above .9 are superb

Revelle has included a KMO test in the psych package. The function can use either raw or matrix data. Either way, the only variables in the matrix should be the items of interest. This means that everything else (e.g., total or subscale scores, ID numbers) should be removed.

```
psych::KMO(dfGRMS)
```

```
Kaiser-Meyer-Olkin factor adequacy
Call: psych::KMO(r = dfGRMS)
Overall MSA =  0.85
MSA for each item =
   Obj1  Obj2  Obj3  Obj4  Obj5  Obj6  Obj7  Obj8  Obj9  Obj10 Marg1 Marg2 Marg3
   0.87  0.91  0.88  0.85  0.85  0.80  0.90  0.85  0.81  0.85  0.86  0.89  0.86
  Marg4 Marg5 Marg6 Marg7 Str1 Str2 Str3 Str4 Str5 Ang1 Ang2 Ang3
   0.86  0.90  0.89  0.84  0.83  0.85  0.82  0.74  0.84  0.78  0.76  0.81
```

```
# psych::KMO(GRMSmatrix)
```

We examine the KMO values for both the overall matrix and the individual items.

At the matrix level, our  $KMO = .85$ , which falls into Kaiser's definition of *good*. You can locate this value as the "Overall MSA."

At the item level, the KMO should be  $> .50$ . Variables with values below .50 should be evaluated for exclusion from the analysis (or run the analysis with and without the variable and compare the

difference). Because removing and adding variables impacts the KMO, be sure to re-evaluate the sampling adequacy if changes are made to the items (and/or sample size).

At the item level, our KMO values range between .74 (Str4) and .91 (Obj2).

Considering both item and matrix levels, we conclude that the sample size and the data are adequate for component (or factor) analysis.

#### 8.5.1.2 Are the correlations among the variables large enough to be analyzed?

**Bartlett's test** lets us know if a matrix is an *identity matrix*. In an identity matrix all correlation coefficients (everything on the off-diagonal) would be 0.0 (and everything on the diagonal would be 1.0).

A significant Barlett's (i.e.,  $p < .05$ ) tells that the  $R$ -matrix is not an identity matrix. That is, there are some relationships between variables that can be analyzed.

The `cortest.bartlett()` function is in the *psych* package and can be run either from the raw data or R matrix formats.

```
psych::cortest.bartlett(dfGRMS) #from the raw data
```

R was not square, finding R from data

```
$chisq  
[1] 1217.508
```

```
$df  
[1] 300
```

```
# raw data produces the warning 'R was not square, finding R from  
# data.' This means nothing other than we fed it raw data and the  
# function is creating a matrix from which to do the analysis.
```

```
# psych::cortest.bartlett(GRMSmatrix, n = 259) #if using the matrix,  
# must specify sample
```

Our Bartlett's test is significant:  $\chi^2(300) = 1217.508, p < .001$ . This means that our sample correlation matrix is statistically significantly different than an identity matrix and, therefore, supports a component-or-factor analytic approach for investigating the data.

#### 8.5.1.3 Is there multicollinearity or singularity in my data?

The **determinant of the correlation matrix** should be greater than 0.00001 (that would be 4 zeros, then the 1). If it is smaller than 0.00001 then we may have an issue with *multicollinearity* (i.e., variables that are too highly correlated) or *singularity* (variables that are perfectly correlated).

The determinant function we use comes from base R. It is easiest to compute when the correlation matrix is the object. However, it is also possible to specify the command to work with the raw data.

```
det(GRMSmatrix)
```

```
[1] 0.007499909
```

```
# det(cor(dfGRMS))#if using the raw data
```

With a value of 0.0075, our determinant is greater than the 0.00001 requirement. If it were not, then we could identify problematic variables (i.e., those correlating too highly with others; those not correlating sufficiently with others) and re-run the diagnostic statistics.

#### 8.5.1.4 APA Style Summary So Far

Data screening was conducted to determine the suitability of the data for principal components analyses. The Kaiser-Meyer-Olkin measure of sampling adequacy (KMO; Kaiser, 1970) represents the ratio of the squared correlation between variables to the squared partial correlation between variables. KMO ranges from 0.00 to 1.00; values closer to 1.00 indicate that the patterns of correlations are relatively compact, and that component analysis should yield distinct and reliable components (Field, 2012). In our dataset, the KMO value was .85, indicating acceptable sampling adequacy. The Bartlett's Test of Sphericity examines whether the population correlation matrix resembles an identity matrix (Field, 2012). When the  $p$  value for the Bartlett's test is  $< .05$ , we are fairly certain we have clusters of correlated variables. In our dataset,  $\chi^2(300) = 1217.508, p < .001$ , indicating the correlations between items are sufficiently large enough for principal components analysis. The determinant of the correlation matrix alerts us to any issues of multicollinearity or singularity and should be larger than 0.00001. Our determinant was 0.0075, supporting the suitability of our data for analysis.

#### 8.5.2 Principal Components Analysis

Below is a snip from the workflow to remind us where we are in the steps to PCA.

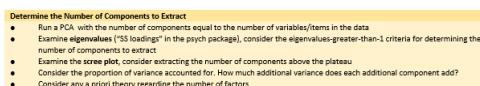


Figure 8.4: Image of an excerpt from the workflow

We can use the *principal()* function from the *psych* package with raw or matrix data.

We start by creating a principal components model that has the same number of components as there are variables in the data. This allows us to inspect the component's eigenvalues and make decisions about which to extract.

- Note, this is different than actual *factor* analysis where you *must* extract fewer factors than variables (e.g., extracting 18 [an arbitrary number] instead of 25).

```
# All of the code sets below are functionally identical. They simply
# swap out using the df or r-matrix, and whether I specify the number
# of factors or write code to instruct R to calculate it.

# pca1 <- psych::principal(GRMSmatrix, nfactors=25, rotate = 'none')
# #using the matrix form of the data and specifying the # factors

# pca1 <- psych::principal(GRMSmatrix,
# nfactors=length(GRMSmatrix[,1]), rotate = 'none') #using the matrix
# form of the data and letting the length function automatically
# calculate the # factors as a function of how many columns in the
# matrix

# pca1 <- psych::principal(dfGRMS, nfactors=25, rotate='none') #using
# raw data and specifying # factors

pca1 <- psych::principal(dfGRMS, nfactors = length(dfGRMS), rotate = "none") # using raw data
pca1
```

### Principal Components Analysis

Call: psych::principal(r = dfGRMS, nfactors = length(dfGRMS), rotate = "none")  
 Standardized loadings (pattern matrix) based upon correlation matrix

	PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8	PC9	PC10	PC11	PC12
Obj1	0.52	-0.29	0.06	0.08	-0.23	-0.12	-0.33	-0.15	-0.19	-0.18	0.04	0.09
Obj2	0.55	-0.28	0.00	0.07	-0.11	0.00	0.03	0.09	-0.28	-0.12	0.14	-0.16
Obj3	0.49	-0.26	0.06	0.06	-0.04	0.32	0.06	-0.24	-0.17	-0.13	0.15	0.01
Obj4	0.52	-0.35	-0.09	0.02	0.08	0.09	0.04	-0.15	0.40	0.18	-0.08	0.28
Obj5	0.51	-0.29	0.09	-0.12	-0.08	-0.04	0.14	-0.42	0.11	0.27	-0.23	-0.20
Obj6	0.40	-0.20	0.01	0.47	-0.14	0.44	-0.03	0.22	0.05	-0.12	0.00	-0.24
Obj7	0.56	-0.11	-0.03	0.06	0.02	-0.33	-0.28	0.26	0.07	0.01	0.04	-0.06
Obj8	0.48	-0.40	-0.13	-0.25	0.00	-0.14	-0.26	0.01	0.07	0.14	0.33	-0.05
Obj9	0.40	-0.28	0.16	-0.04	-0.07	-0.11	0.53	0.12	0.33	-0.29	0.00	-0.23
Obj10	0.42	-0.32	0.00	0.23	0.12	-0.35	0.14	0.21	-0.13	0.19	-0.18	0.39
Marg1	0.58	0.31	-0.36	-0.26	0.06	-0.02	-0.13	0.05	0.04	-0.06	-0.01	-0.07
Marg2	0.58	0.38	-0.08	-0.13	-0.21	0.09	-0.02	-0.03	-0.04	-0.13	-0.14	0.14
Marg3	0.51	0.23	-0.18	-0.23	0.11	-0.18	0.36	-0.08	-0.28	-0.10	-0.02	-0.16
Marg4	0.50	0.10	-0.35	-0.07	0.38	0.01	-0.05	-0.28	-0.10	-0.01	0.11	0.02
Marg5	0.56	0.19	-0.19	-0.20	-0.08	0.27	-0.01	-0.03	0.20	0.03	-0.31	0.01
Marg6	0.54	-0.01	-0.19	0.05	0.24	0.09	0.00	0.37	-0.09	0.27	-0.16	-0.17
Marg7	0.41	0.21	-0.27	-0.08	-0.38	0.20	0.16	0.34	0.04	0.01	0.25	0.36
Str1	0.43	0.12	0.45	-0.30	-0.11	-0.19	-0.06	0.11	-0.16	-0.13	0.00	-0.12
Str2	0.40	0.03	0.31	0.15	0.43	0.07	0.15	-0.05	-0.13	-0.37	-0.07	0.37
Str3	0.33	0.22	0.54	-0.08	0.17	0.00	-0.18	0.26	0.15	0.04	-0.22	-0.07
Str4	0.29	0.03	0.48	-0.23	0.31	0.41	-0.14	-0.01	-0.02	0.21	0.26	0.02



Str1	-0.17	1	0.000000000000000122	9.1
Str2	0.05	1	0.000000000000000133	8.8
Str3	-0.01	1	0.000000000000000000	7.1
Str4	0.13	1	0.000000000000000056	8.4
Str5	0.00	1	0.000000000000000100	9.2
Ang1	-0.04	1	0.000000000000000067	8.7
Ang2	-0.01	1	0.000000000000000022	6.5
Ang3	-0.08	1	0.000000000000000056	10.3

	PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8	PC9	PC10	PC11
SS loadings	5.37	1.71	1.54	1.22	1.08	1.04	1.02	0.98	0.95	0.90	0.84
Proportion Var	0.21	0.07	0.06	0.05	0.04	0.04	0.04	0.04	0.04	0.04	0.03
Cumulative Var	0.21	0.28	0.34	0.39	0.44	0.48	0.52	0.56	0.60	0.63	0.67
Proportion Explained	0.21	0.07	0.06	0.05	0.04	0.04	0.04	0.04	0.04	0.04	0.03
Cumulative Proportion	0.21	0.28	0.34	0.39	0.44	0.48	0.52	0.56	0.60	0.63	0.67
	PC12	PC13	PC14	PC15	PC16	PC17	PC18	PC19	PC20	PC21	PC22
SS loadings	0.83	0.75	0.73	0.69	0.68	0.64	0.61	0.58	0.55	0.51	0.48
Proportion Var	0.03	0.03	0.03	0.03	0.03	0.03	0.02	0.02	0.02	0.02	0.02
Cumulative Var	0.70	0.73	0.76	0.79	0.81	0.84	0.86	0.89	0.91	0.93	0.95
Proportion Explained	0.03	0.03	0.03	0.03	0.03	0.03	0.02	0.02	0.02	0.02	0.02
Cumulative Proportion	0.70	0.73	0.76	0.79	0.81	0.84	0.86	0.89	0.91	0.93	0.95
	PC23	PC24	PC25								
SS loadings	0.45	0.45	0.41								
Proportion Var	0.02	0.02	0.02								
Cumulative Var	0.97	0.98	1.00								
Proportion Explained	0.02	0.02	0.02								
Cumulative Proportion	0.97	0.98	1.00								

Mean item complexity = 7.9

Test of the hypothesis that 25 components are sufficient.

The root mean square of the residuals (RMSR) is 0  
with the empirical chi square 0 with prob < NA

Fit based upon off diagonal values = 1

The total variance for a particular variable will have two components: some of it will be share with other variables (common variance,  $h^2$ ) and some of it will be specific to that measure (unique variance,  $u^2$ ). Random variance is also specific to one item, but not reliably so.

We can examine this most easily by examining the matrix (second screen).

The columns PC1 thru PC25 are the (uninteresting at this point) unrotated loadings. PC stands for “principal component.” Although these don’t align with the specific items, at this point in the procedure, there are as many components as variables.

**Communalities** are represented as  $h^2$ . These are the proportions of common variance present in the variables. A variable that has no specific (or random) variance would have a communality of 1.0. If a variable shares none of its variance with any other variable its communality would be 0.0.

Because we extracted the same number components as variables, they all equal 1.0. That is, we have explained all the variance in each variable. When we specify fewer components, the value of the communalities will decrease.

\*\*Uniquenesses\* are represented as  $u^2$ . These are the amount of unique variance for each variable. They are calculated as  $1 - h^2$  (or 1 minus the communality). Technically (at this point in the analysis where we have an equal number of components as items), they should all be zero, but the *psych* package is very “quantsy” and decimals are reported to the 15th and 16th decimal places! (hence the  $u^2$  for Q1 is -0.0000000000000006661338).

The final column, *com*, represents *item complexity*. This is an indication of how well an item reflects a single construct. If it is 1.0 then the item loads only on one component, if it is 2.0, it loads evenly on two components, and so forth. For now, we can ignore this. *I mostly wanted to reassure you that “com” is not “communality”;  $h^2$  is communality.*

Let’s switch to the first screen of output.

**Eigenvalues** are displayed in the row called *SS loadings* (i.e., the sum of squared loadings). They represent the variance explained by the particular linear component. PC1 explains 5.37 units of variance (out of a possible 25, the total of components). As a proportion, this is  $5.37/25 = 0.21$  (reported in the *Proportion Var* row).

```
5.37/25
```

```
[1] 0.2148
```

Note:

- *Cumulative Var* is helpful in determining how many components we would like to retain to balance parsimony (where the goal is frequently “as few as possible”) with the amount of variance we want to explain.
- The eigenvalues are in descending order. If we were to use the *eigenvalue > 1.0* (i.e., “Kaiser’s”) criteria to determine how many components to extract, we would select 7. Joliffe’s criteria was 0.7 (thus, we would select 14 components). Eigenvalues are only one criteria, let’s look at the scree plot.

*Scree plot:* We can gain another view of how many components to extract by creating a scree plot.

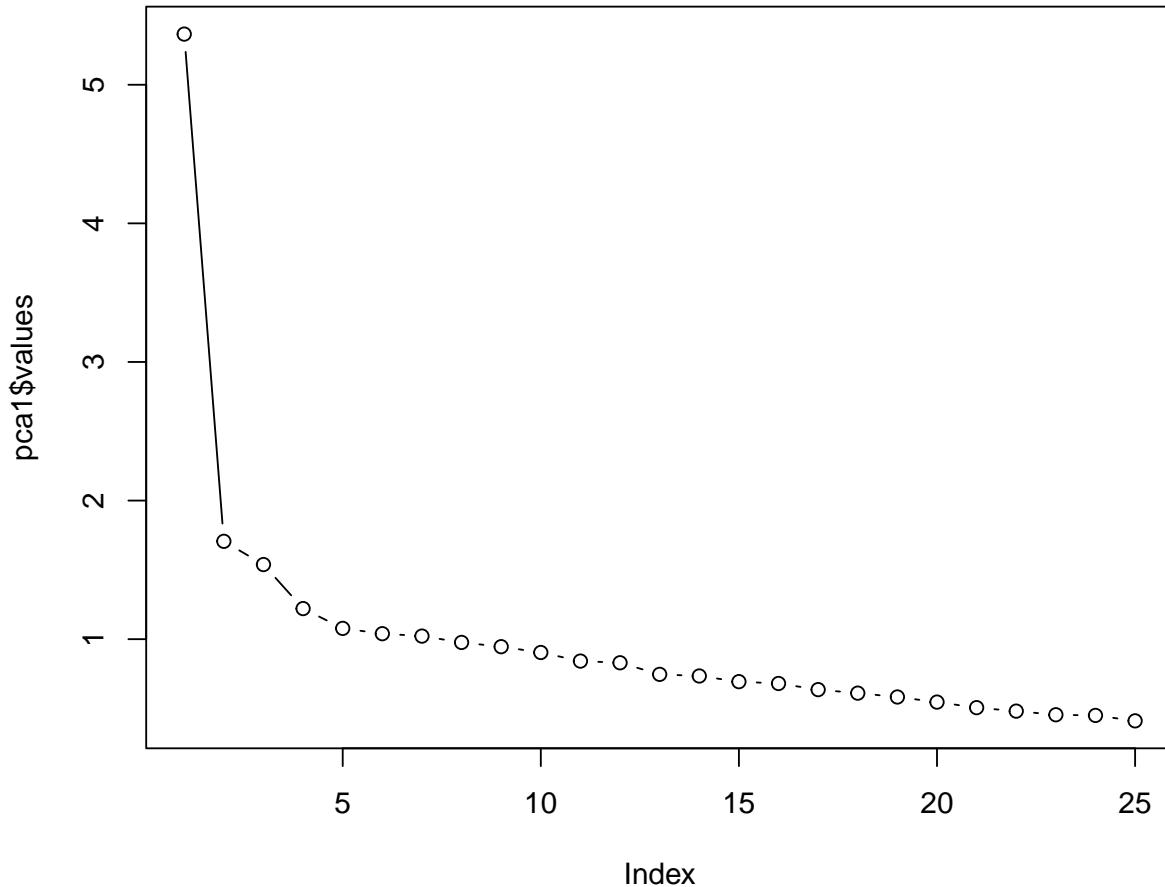
Eigenvalues are stored in the *pca1* object’s variable, “values”. We can see all the values captured by this object with the *names()* function:

```
names(pca1)
```

```
[1] "values"      "rotation"     "n.obs"       "communality"   "loadings"
[6] "fit"         "fit.off"      "fn"          "Call"        "uniquenesses"
[11] "complexity"  "valid"        "chi"         "EPVAL"       "R2"
[16] "objective"   "residual"    "rms"         "factors"     "dof"
[21] "null.dof"    "null.model"  "criteria"    "STATISTIC"   "PVAL"
[26] "weights"     "r.scores"    "Vaccounted" "Structure"   "scores"
```

Plotting the *eigenvalues* produces a scree plot. We can use this to further gauge the number of factors we should extract.

```
plot(pca1$values, type = "b") #type = 'b' gives us 'both' lines and points; type = 'l' gives
```



We look for the point of *inflection*. That is, where the baseline levels out into a plateau. It seems to me that there is only one clear component above the plateau. However, we see that components #5 and 5 flatten out, and then there is another drop. So, it could be 1, 2, or 4.

### 8.5.3 Specifying the Number of Components

Below is a snip from the workflow to remind us of where we are in the steps to PCA.

Having determined the number of components, we re-run the analysis with this specification. Especially when researchers may not have a clear theoretical structure that guides the process, researchers may do this iteratively with varying numbers of factors. Lewis and Neville [Lewis and Neville, 2015] examined solutions with 2, 3, 4, and 5 factors. Further, they used a parallel *factor* analysis, whereas we used a principal components analysis).

**Specify and Evaluate the Solution with the Specified Number of Components**

- Run the PA with the determined number of components
- Examine communalities individually and as an average. Criteria for acceptability range from being greater than .40 to being greater than .70
- Examine residuals. Determine the number and percentage that are greater than .05. A common practice is to stay below 50% larger than .05.

Figure 8.5: Image of an excerpt from the workflow

```
# pca2 <- psych::principal(GRMSmatrix, nfactors=4, rotate='none')
pca2 <- psych::principal(dfGRMS, nfactors = 4, rotate = "none") #can copy prior script, but can't run it
pca2
```

### Principal Components Analysis

Call: psych::principal(r = dfGRMS, nfactors = 4, rotate = "none")  
 Standardized loadings (pattern matrix) based upon correlation matrix

	PC1	PC2	PC3	PC4	h2	u2	com
Obj1	0.52	-0.29	0.06	0.08	0.37	0.63	1.7
Obj2	0.55	-0.28	0.00	0.07	0.39	0.61	1.5
Obj3	0.49	-0.26	0.06	0.06	0.32	0.68	1.6
Obj4	0.52	-0.35	-0.09	0.02	0.40	0.60	1.8
Obj5	0.51	-0.29	0.09	-0.12	0.36	0.64	1.8
Obj6	0.40	-0.20	0.01	0.47	0.42	0.58	2.3
Obj7	0.56	-0.11	-0.03	0.06	0.32	0.68	1.1
Obj8	0.48	-0.40	-0.13	-0.25	0.47	0.53	2.6
Obj9	0.40	-0.28	0.16	-0.04	0.27	0.73	2.2
Obj10	0.42	-0.32	0.00	0.23	0.33	0.67	2.5
Marg1	0.58	0.31	-0.36	-0.26	0.63	0.37	2.7
Marg2	0.58	0.38	-0.08	-0.13	0.50	0.50	1.9
Marg3	0.51	0.23	-0.18	-0.23	0.40	0.60	2.1
Marg4	0.50	0.10	-0.35	-0.07	0.39	0.61	2.0
Marg5	0.56	0.19	-0.19	-0.20	0.43	0.57	1.8
Marg6	0.54	-0.01	-0.19	0.05	0.33	0.67	1.2
Marg7	0.41	0.21	-0.27	-0.08	0.29	0.71	2.4
Str1	0.43	0.12	0.45	-0.30	0.50	0.50	2.9
Str2	0.40	0.03	0.31	0.15	0.28	0.72	2.2
Str3	0.33	0.22	0.54	-0.08	0.45	0.55	2.1
Str4	0.29	0.03	0.48	-0.23	0.37	0.63	2.1
Str5	0.34	0.20	0.37	0.10	0.30	0.70	2.7
Ang1	0.35	0.40	0.27	0.14	0.37	0.63	3.0
Ang2	0.33	0.37	-0.10	0.57	0.57	0.43	2.5
Ang3	0.38	0.31	-0.04	0.37	0.38	0.62	2.9

	PC1	PC2	PC3	PC4
SS loadings	5.37	1.71	1.54	1.22
Proportion Var	0.21	0.07	0.06	0.05
Cumulative Var	0.21	0.28	0.34	0.39
Proportion Explained	0.55	0.17	0.16	0.12
Cumulative Proportion	0.55	0.72	0.88	1.00

Mean item complexity = 2.1

Test of the hypothesis that 4 components are sufficient.

The root mean square of the residuals (RMSR) is 0.06

Fit based upon off diagonal values = 0.89

Our eigenvalues/SS loadings remain the same. With 4 components, we explain 39% of the variance (we can see this in the “Cumulative Var” row.

*Communality* is the proportion of common variance within a variable. Principal components analysis assumes that all variance is common. Before extraction, all variance was set at 1.0, therefore, changing from 25 to 4 components will change this value ( $h^2$ ) as well as its associated *uniqueness* ( $u^2$ ), which is calculated as “1.0 minus the communality.”

The *communalities* ( $h^2$ ) and *uniquenesses* ( $u^2$ ) have changed.

Now we see that 37% of the variance associate with Obj1 is common/shared (the  $h^2$  value).

Recall that we could represent this scale with all 25 items as components, but we want a more *parsimonious* explanation. By respecifying a smaller number of components, we lose some information. That is, the retained components (now 4) cannot explain all of the variance present in the data (as we saw, it explains about 39%, cumulatively). The amount of variance explained in each variable is represented by the communalities after extraction.

We can examine the communalities through the lens of Kaiser's criterion (the eigenvalue  $> 1$  criteria) to see if we think that "four" was a good number of components to extract.

Kaiser's criterion is believed to be accurate if:

- when there are fewer than 30 variables (we had 25) and, after extraction, the communalities are greater than .70
    - looking at our data, none are  $> .70$ , so, this does not support extracting four components
  - when the sample size is greater than 250 (ours was 259) and the average communality is  $> .60$ 
    - we can extract the communalities from our object and calculate the mean the average communality

Using the `names()` function again, we see that “communality” is available. Thus, we can easily calculate their mean. To get this value let’s first examine the possible contents of the object we created from this PCA analysis by asking for its names.

```
names(pca2)
```

```
[1] "values"      "rotation"     "n.obs"       "communality"   "loadings"  
[6] "fit"         "fit.off"      "fn"          "Call"        "uniquenesses"  
[11] "complexity"  "valid"        "chi"         "EPVAL"       "R2"
```

```
[16] "objective"      "residual"       "rms"           "factors"        "dof"
[21] "null.dof"       "null.model"     "criteria"       "STATISTIC"      "PVAL"
[26] "weights"         "r.scores"       "Vaccounted"    "Structure"      "scores"
```

We see that it includes communalities. Thus, we can easily calculate their mean.

```
mean(pca2$communality)
```

```
[1] 0.3932201
```

We see that the average communality is 0.39. These two criteria would suggest that we may not have the best solution. That said (in our defense):

- We used the scree plot as a guide and there was support for four dimensions.
- We have an adequate sample size and that was supported with the KMO.
- Are the number of components consistent with theory? We have not yet inspected the component loadings. This will provide us with more information.

We could do several things:

- re-run with a different number of components (recall Lewis and Neville [2015] ran models with 2, 3, 4, and 5 factors)
- conduct more diagnostics
  - reproduced correlation matrix
  - the difference between the reproduced correlation matrix and the correlation matrix in the data

The *factor.model()* function in *psych* produces the *reproduced correlation matrix* by using the *loadings* from our extracted object. Conceptually, this matrix is the correlations that should be produced if we did not have the raw data, but we only had the component loadings. We could do fancy matrix algebra and produce these.

The questions, though, is: How close did we get? How different is the *reproduced correlation matrix* from *GRMSmatrix* – the *R*-matrix produced from our raw data.

```
# produces the reproduced correlation matrix
round(psych::factor.model(pca2$loadings), 3)
```

	Obj1	Obj2	Obj3	Obj4	Obj5	Obj6	Obj7	Obj8	Obj9	Obj10	Marg1
Obj1	0.368	0.375	0.343	0.372	0.346	0.308	0.324	0.340	0.298	0.329	0.171
Obj2	0.375	0.387	0.350	0.388	0.352	0.313	0.341	0.358	0.297	0.335	0.217
Obj3	0.343	0.350	0.321	0.346	0.326	0.281	0.305	0.320	0.280	0.303	0.169
Obj4	0.372	0.388	0.346	0.404	0.356	0.291	0.333	0.397	0.292	0.335	0.224
Obj5	0.346	0.352	0.326	0.356	0.365	0.208	0.303	0.377	0.305	0.277	0.205
Obj6	0.308	0.313	0.281	0.291	0.208	0.424	0.274	0.157	0.202	0.338	0.044



Obj3	0.089	0.092	0.129
Obj4	0.020	0.065	0.107
Obj5	0.068	-0.017	0.059
Obj6	0.123	0.321	0.265
Obj7	0.148	0.180	0.205
Obj8	-0.065	-0.119	-0.027
Obj9	0.066	-0.009	0.050
Obj10	0.045	0.145	0.145
Marg1	0.197	0.196	0.238
Marg2	0.311	0.262	0.293
Marg3	0.188	0.138	0.187
Marg4	0.110	0.198	0.211
Marg5	0.191	0.157	0.206
Marg6	0.141	0.219	0.230
Marg7	0.143	0.192	0.201
Str1	0.277	-0.034	0.073
Str2	0.253	0.195	0.207
Str3	0.333	0.085	0.143
Str4	0.208	-0.074	0.016
Str5	0.309	0.205	0.216
Ang1	0.372	0.312	0.299
Ang2	0.312	0.575	0.454
Ang3	0.299	0.454	0.383

We're not really interested in this matrix. We just need it to compare it to the *GRMSmatrix* to produce the residuals. We do that next.

**Residuals** are the difference between the reproduced (i.e., those created from our component loadings) and *R*-matrix produced by the raw data.

If we look at the  $r_{Obj1Obj2}$  in our original correlation matrix (theoretically from the raw data [although we simulated data]), the value is 0.35. The reproduced correlation that we just calculated for this pair is 0.375. The difference is -0.025.

```
0.35 - 0.375
```

```
[1] -0.025
```

By using the *factor.residuals()* function R will calculate the residuals for each pair.

```
round(psych::factor.residuals(GRMSmatrix, pca2$loadings), 3)
```

	Obj1	Obj2	Obj3	Obj4	Obj5	Obj6	Obj7	Obj8	Obj9	Obj10
Obj1	0.632	-0.020	-0.093	-0.101	-0.063	-0.057	-0.044	0.006	-0.145	-0.088
Obj2	-0.020	0.613	-0.038	-0.140	-0.085	-0.079	-0.029	-0.076	-0.035	-0.090
Obj3	-0.093	-0.038	0.679	-0.105	-0.043	-0.002	-0.103	-0.072	-0.070	-0.088
Obj4	-0.101	-0.140	-0.105	0.596	0.031	-0.060	-0.048	-0.100	-0.036	-0.055

Obj5	-0.063	-0.085	-0.043	0.031	0.635	-0.055	-0.126	-0.086	-0.053	-0.072
Obj6	-0.057	-0.079	-0.002	-0.060	-0.055	0.576	-0.075	-0.015	0.004	-0.217
Obj7	-0.044	-0.029	-0.103	-0.048	-0.126	-0.075	0.675	0.009	-0.054	0.004
Obj8	0.006	-0.076	-0.072	-0.100	-0.086	-0.015	0.009	0.531	-0.101	-0.043
Obj9	-0.145	-0.035	-0.070	-0.036	-0.053	0.004	-0.054	-0.101	0.732	-0.049
Obj10	-0.088	-0.090	-0.088	-0.055	-0.072	-0.217	0.004	-0.043	-0.049	0.674
Marg1	0.014	-0.002	0.000	-0.005	-0.039	0.058	0.011	0.004	0.000	0.009
Marg2	0.068	0.000	0.055	-0.002	0.002	0.045	-0.061	-0.034	-0.004	0.031
Marg3	-0.002	0.042	0.005	-0.064	0.020	0.005	-0.052	-0.095	0.091	0.077
Marg4	-0.015	-0.061	0.048	0.000	-0.002	-0.041	-0.021	-0.020	-0.051	-0.038
Marg5	-0.039	-0.025	0.007	0.026	0.014	0.072	-0.053	-0.081	0.037	-0.020
Marg6	-0.099	-0.036	-0.102	-0.076	-0.030	0.024	-0.025	-0.014	-0.039	0.018
Marg7	-0.001	0.028	0.015	0.029	-0.076	0.093	-0.052	-0.025	0.037	0.023
Str1	0.026	0.005	-0.046	-0.071	-0.029	0.057	0.056	-0.006	-0.034	0.031
Str2	-0.041	-0.038	-0.027	0.016	-0.088	-0.075	-0.083	-0.040	-0.014	-0.002
Str3	-0.033	-0.025	-0.040	0.036	-0.052	0.042	0.052	0.026	-0.046	0.053
Str4	-0.056	0.008	0.036	0.061	-0.087	0.077	-0.055	0.009	-0.141	-0.034
Str5	0.049	0.006	0.009	0.002	0.054	-0.035	-0.013	0.027	-0.072	-0.009
Ang1	-0.025	-0.021	-0.016	0.072	0.050	-0.083	0.003	0.138	0.103	0.018
Ang2	-0.043	0.026	-0.009	-0.005	0.104	-0.124	-0.050	0.093	0.005	-0.007
Ang3	0.068	-0.027	-0.017	0.033	0.046	-0.110	0.021	0.092	0.012	-0.062
	Marg1	Marg2	Marg3	Marg4	Marg5	Marg6	Marg7	Str1	Str2	Str3
Obj1	0.014	0.068	-0.002	-0.015	-0.039	-0.099	-0.001	0.026	-0.041	-0.033
Obj2	-0.002	0.000	0.042	-0.061	-0.025	-0.036	0.028	0.005	-0.038	-0.025
Obj3	0.000	0.055	0.005	0.048	0.007	-0.102	0.015	-0.046	-0.027	-0.040
Obj4	-0.005	-0.002	-0.064	0.000	0.026	-0.076	0.029	-0.071	0.016	0.036
Obj5	-0.039	0.002	0.020	-0.002	0.014	-0.030	-0.076	-0.029	-0.088	-0.052
Obj6	0.058	0.045	0.005	-0.041	0.072	0.024	0.093	0.057	-0.075	0.042
Obj7	0.011	-0.061	-0.052	-0.021	-0.053	-0.025	-0.052	0.056	-0.083	0.052
Obj8	0.004	-0.034	-0.095	-0.020	-0.081	-0.014	-0.025	-0.006	-0.040	0.026
Obj9	0.000	-0.004	0.091	-0.051	0.037	-0.039	0.037	-0.034	-0.014	-0.046
Obj10	0.009	0.031	0.077	-0.038	-0.020	0.018	0.023	0.031	-0.002	0.053
Marg1	0.367	-0.094	-0.086	-0.089	-0.098	-0.020	-0.110	-0.012	0.026	0.043
Marg2	-0.094	0.499	-0.078	-0.134	-0.034	-0.053	-0.022	-0.041	-0.016	-0.057
Marg3	-0.086	-0.078	0.602	-0.033	-0.158	-0.044	-0.117	-0.031	0.051	-0.043
Marg4	-0.089	-0.134	-0.033	0.614	-0.079	-0.073	-0.159	0.015	0.128	0.048
Marg5	-0.098	-0.034	-0.158	-0.079	0.575	-0.038	-0.052	-0.079	-0.009	0.019
Marg6	-0.020	-0.053	-0.044	-0.073	-0.038	0.671	-0.066	-0.003	0.018	0.075
Marg7	-0.110	-0.022	-0.117	-0.159	-0.052	-0.066	0.714	0.037	-0.024	-0.002
Str1	-0.012	-0.041	-0.031	0.015	-0.079	-0.003	0.037	0.505	-0.054	-0.133
Str2	0.026	-0.016	0.051	0.128	-0.009	0.018	-0.024	-0.054	0.724	-0.091
Str3	0.043	-0.057	-0.043	0.048	0.019	0.075	-0.002	-0.133	-0.091	0.551
Str4	0.023	-0.048	-0.039	0.073	0.021	0.080	0.005	-0.187	-0.026	-0.109
Str5	-0.001	0.003	-0.005	-0.020	0.040	-0.035	0.049	-0.073	-0.148	-0.161
Ang1	-0.034	-0.079	-0.006	0.008	-0.076	-0.036	0.028	-0.096	-0.093	-0.134
Ang2	-0.026	-0.077	0.054	0.024	-0.015	-0.006	-0.059	0.084	-0.074	-0.011
Ang3	0.041	-0.015	-0.077	-0.038	0.003	-0.114	-0.115	0.029	-0.052	0.004
	Str4	Str5	Ang1	Ang2	Ang3					

Obj1	-0.056	0.049	-0.025	-0.043	0.068
Obj2	0.008	0.006	-0.021	0.026	-0.027
Obj3	0.036	0.009	-0.016	-0.009	-0.017
Obj4	0.061	0.002	0.072	-0.005	0.033
Obj5	-0.087	0.054	0.050	0.104	0.046
Obj6	0.077	-0.035	-0.083	-0.124	-0.110
Obj7	-0.055	-0.013	0.003	-0.050	0.021
Obj8	0.009	0.027	0.138	0.093	0.092
Obj9	-0.141	-0.072	0.103	0.005	0.012
Obj10	-0.034	-0.009	0.018	-0.007	-0.062
Marg1	0.023	-0.001	-0.034	-0.026	0.041
Marg2	-0.048	0.003	-0.079	-0.077	-0.015
Marg3	-0.039	-0.005	-0.006	0.054	-0.077
Marg4	0.073	-0.020	0.008	0.024	-0.038
Marg5	0.021	0.040	-0.076	-0.015	0.003
Marg6	0.080	-0.035	-0.036	-0.006	-0.114
Marg7	0.005	0.049	0.028	-0.059	-0.115
Str1	-0.187	-0.073	-0.096	0.084	0.029
Str2	-0.026	-0.148	-0.093	-0.074	-0.052
Str3	-0.109	-0.161	-0.134	-0.011	0.004
Str4	0.634	-0.136	-0.054	0.105	0.006
Str5	-0.136	0.702	-0.085	-0.051	-0.110
Ang1	-0.054	-0.085	0.628	-0.072	-0.067
Ang2	0.105	-0.051	-0.072	0.425	-0.204
Ang3	0.006	-0.110	-0.067	-0.204	0.617

Their calculated difference (-0.20) is quite close to our hand calculation (-0.25). There are several strategies to evaluate this matrix:

- See how large the residuals are compared to the original correlations.
  - The worst possible model would occur if we extracted no components and would be the size of the original correlations.
  - If the correlations were small to start with, we expect small residuals.
  - If the correlations were large to start with, the residuals will be relatively larger (this is not terribly problematic).
- Comparing residuals requires squaring them first (because residuals can be both positive and negative).
  - The sum of the squared residuals divided by the sum of the squared correlations is an estimate of model fit. Subtracting this from 1.0 means that it ranges from 0 to 1. Values > .95 are an indication of good fit.

Analyzing the residuals means we need to extract only the upper right of the triangle of the matrix into an object. We can do this in steps.

```
# first extract the residuals
pca2_resids <- psych::factor.residuals(GRMSmatrix, pca2$loadings)
# the object has the residuals in a single column
pca2_resids <- as.matrix(pca2_resids[upper.tri(pca2_resids)])
# display the first 6 rows of the residuals
head(pca2_resids)
```

```
[,1]
[1,] -0.02024211
[2,] -0.09293107
[3,] -0.03803285
[4,] -0.10123779
[5,] -0.14040791
[6,] -0.10510587
```

One criteria of residual analysis is to see how many residuals there are that are greater than an absolute value of 0.05. The result will be a single column with TRUE if it is  $> |0.05|$  and false if it is smaller. The sum function will tell us how many TRUE responses are in the matrix. Further, we can write script to obtain the proportion of total number of residuals.

```
large.resid <- abs(pca2_resids) > 0.05
# large.resid
sum(large.resid)
```

```
[1] 129
```

```
round(sum(large.resid)/nrow(pca2_resids), 3)
```

```
[1] 0.43
```

We learn that there are 129 residuals greater than the absolute value of 0.05. This represents 43% of the total number of residuals.

There are no hard rules about what proportion of residuals can be greater than 0.05. A common practice is to stay below 50% [Field, 2012].

Another approach to analyzing residuals is to look at their mean. Because of the +/- valences, we need to square them (to eliminate the negative), take the average, then take the square root.

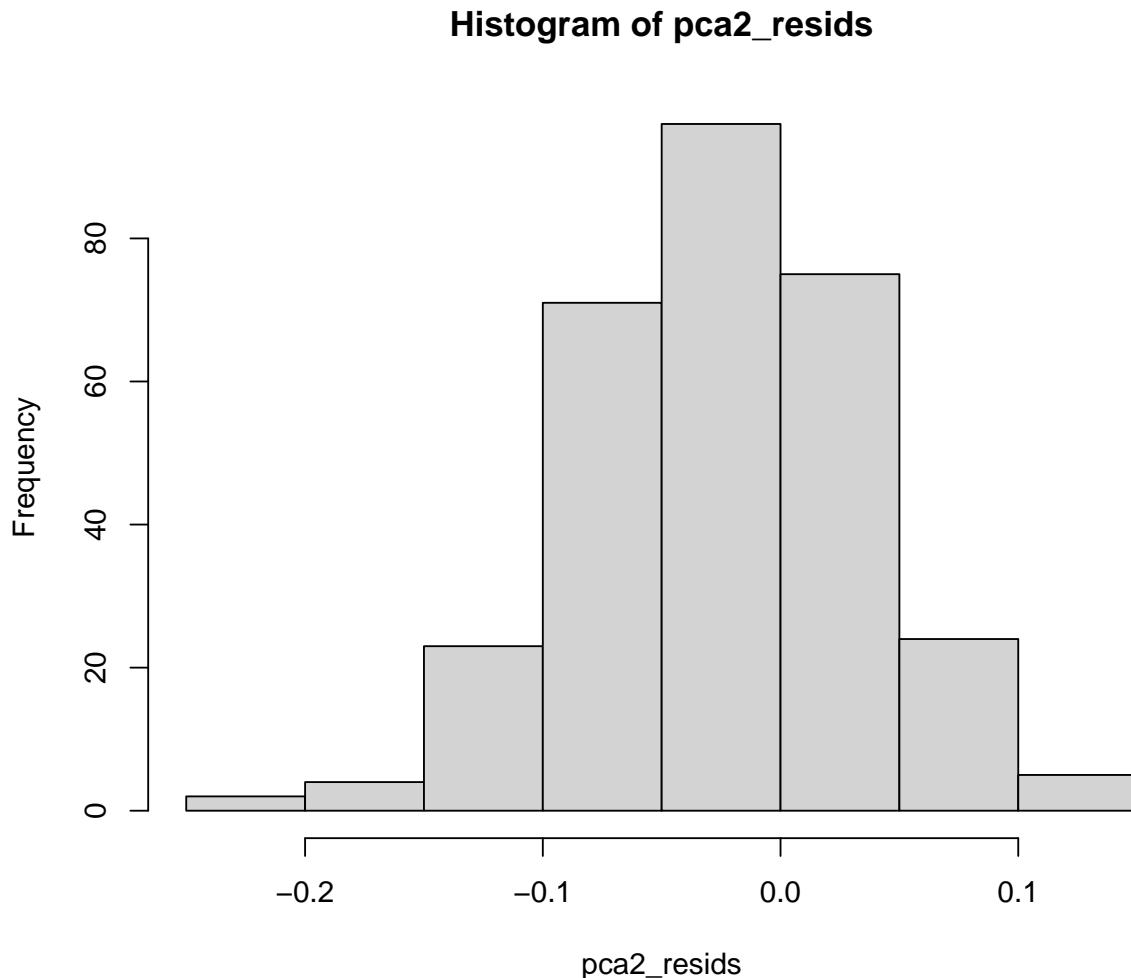
```
round(sqrt(mean(pca2_resids^2)), 3)
```

```
[1] 0.064
```

While there are no clear guidelines to interpret these, one recommendation is to consider extracting more components if the value is higher than 0.08 [Field, 2012]. Our value of 0.064 is  $< 0.08$ .

Finally, we expect our residuals to be normally distributed. A histogram can help us inspect the distribution.

```
hist(pca2_resids)
```



This looks reasonably normal to me, and I do not see an indication of outliers.

#### 8.5.3.1 Quick recap of how to evaluate the # of components we extracted

- If fewer than 30 variables, the eigenvalue  $> 1$  (Kaiser's) criteria is fine, so long as communalities are all  $> .70$ .
- If sample size  $> 250$  and the average communalities are  $.6$  or greater, this is acceptable.
- When  $N > 200$ , the scree plot can be used.
- Regarding residuals:
  - Fewer than 50% should have absolute values  $> 0.05$ .
  - Model fit should be  $> 0.90$ .

### 8.5.4 Component Rotation

Below is a snip from the workflow to remind us of where we are in the steps to PCA.

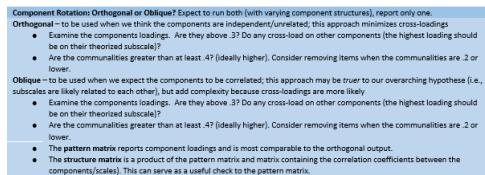


Figure 8.6: Image of an excerpt from the workflow

Rotation improves the interpretation of the components by maximizing the loading on each variable on one of the extracted components while minimizing the loading on all other components. Rotation works by changing the absolute values of the variables while keeping their differential values constant.

There are two big choices, and we need to make them on theoretical grounds:

- Orthogonal rotation if you think that the components are independent/unrelated.
  - Varimax is the most common orthogonal rotation.
- Oblique rotation if you think that the components are related correlated.
  - Oblimin and promax are common oblique rotations.

Which to do?

- Orthogonal is sometimes considered to be “easier” because it minimizes cross-loadings, but
- Can you think of a measure where the subscales would *not* be correlated?

#### 8.5.4.1 Orthogonal rotation

```

# pcaORTH <- psych::principal(GRMSmatrix, nfactors = 4, rotate =
# 'varimax')
pcaORTH <- psych::principal(dfGRMS, nfactors = 4, rotate = "varimax")
pcaORTH

```

Principal Components Analysis

Call: psych::principal(r = dfGRMS, nfactors = 4, rotate = "varimax")  
Standardized loadings (pattern matrix) based upon correlation matrix

	RC1	RC2	RC3	RC4	h2	u2	com
Obj1	0.57	0.12	0.13	0.08	0.37	0.63	1.2
Obj2	0.58	0.18	0.10	0.09	0.39	0.61	1.3
Obj3	0.53	0.13	0.14	0.07	0.32	0.68	1.3
Obj4	0.60	0.20	0.00	0.01	0.40	0.60	1.2

Obj5	0.53	0.18	0.20	-0.10	0.36	0.64	1.6
Obj6	0.49	-0.04	-0.02	0.43	0.42	0.58	2.0
Obj7	0.46	0.28	0.13	0.15	0.32	0.68	2.1
Obj8	0.57	0.27	0.00	-0.26	0.47	0.53	1.9
Obj9	0.47	0.05	0.21	-0.05	0.27	0.73	1.4
Obj10	0.54	0.02	0.00	0.17	0.33	0.67	1.2
Marg1	0.11	0.78	0.06	0.05	0.63	0.37	1.1
Marg2	0.09	0.61	0.28	0.18	0.50	0.50	1.7
Marg3	0.13	0.60	0.16	0.02	0.40	0.60	1.2
Marg4	0.24	0.56	-0.07	0.11	0.39	0.61	1.5
Marg5	0.20	0.60	0.15	0.04	0.43	0.57	1.4
Marg6	0.37	0.40	0.03	0.18	0.33	0.67	2.4
Marg7	0.10	0.51	0.00	0.12	0.29	0.71	1.2
Str1	0.16	0.19	0.65	-0.12	0.50	0.50	1.4
Str2	0.27	0.04	0.38	0.24	0.28	0.72	2.6
Str3	0.05	0.05	0.66	0.09	0.45	0.55	1.1
Str4	0.14	0.02	0.57	-0.12	0.37	0.63	1.2
Str5	0.10	0.06	0.47	0.25	0.30	0.70	1.7
Ang1	-0.04	0.20	0.44	0.37	0.37	0.63	2.4
Ang2	0.03	0.19	0.01	0.73	0.57	0.43	1.1
Ang3	0.09	0.25	0.12	0.55	0.38	0.62	1.5

	RC1	RC2	RC3	RC4
SS loadings	3.31	2.92	2.05	1.56
Proportion Var	0.13	0.12	0.08	0.06
Cumulative Var	0.13	0.25	0.33	0.39
Proportion Explained	0.34	0.30	0.21	0.16
Cumulative Proportion	0.34	0.63	0.84	1.00

Mean item complexity = 1.5

Test of the hypothesis that 4 components are sufficient.

The root mean square of the residuals (RMSR) is 0.06

Fit based upon off diagonal values = 0.89

Essentially, we have the same information as before, except that loadings are calculated after rotation (which adjusts the absolute values of the component loadings while keeping their differential values constant). Our communality and uniqueness values remain the same. The eigenvalues (SS loadings) should even out, but the proportion of variance explained and cumulative variance will remain the same (39%).

The `print.psych()` function facilitates interpretation and prioritizes the information about which we care most:

- “cut” will display loadings above .3
    - if some items load on no factors

- if some items have cross-loadings (and their relative weights)
- “sort” will reorder the loadings to make it clearer (to the best of its ability...in the case of ties) to which component/scale it belongs

```
pca_table <- psych::print.psych(pcaORTH, cut = 0.3, sort = TRUE)
```

### Principal Components Analysis

Call: psych::principal(r = dfGRMS, nfactors = 4, rotate = "varimax")  
 Standardized loadings (pattern matrix) based upon correlation matrix

	item	RC1	RC2	RC3	RC4	h2	u2	com
Obj4	4	0.60				0.40	0.60	1.2
Obj2	2	0.58				0.39	0.61	1.3
Obj1	1	0.57				0.37	0.63	1.2
Obj8	8	0.57				0.47	0.53	1.9
Obj10	10	0.54				0.33	0.67	1.2
Obj5	5	0.53				0.36	0.64	1.6
Obj3	3	0.53				0.32	0.68	1.3
Obj6	6	0.49			0.43	0.42	0.58	2.0
Obj9	9	0.47				0.27	0.73	1.4
Obj7	7	0.46				0.32	0.68	2.1
Marg1	11		0.78			0.63	0.37	1.1
Marg2	12		0.61			0.50	0.50	1.7
Marg5	15		0.60			0.43	0.57	1.4
Marg3	13		0.60			0.40	0.60	1.2
Marg4	14		0.56			0.39	0.61	1.5
Marg7	17		0.51			0.29	0.71	1.2
Marg6	16	0.37	0.40			0.33	0.67	2.4
Str3	20			0.66		0.45	0.55	1.1
Str1	18			0.65		0.50	0.50	1.4
Str4	21			0.57		0.37	0.63	1.2
Str5	22			0.47		0.30	0.70	1.7
Ang1	23			0.44	0.37	0.37	0.63	2.4
Str2	19			0.38		0.28	0.72	2.6
Ang2	24					0.73	0.57	0.43
Ang3	25					0.55	0.38	0.62
								1.5

	RC1	RC2	RC3	RC4
SS loadings	3.31	2.92	2.05	1.56
Proportion Var	0.13	0.12	0.08	0.06
Cumulative Var	0.13	0.25	0.33	0.39
Proportion Explained	0.34	0.30	0.21	0.16
Cumulative Proportion	0.34	0.63	0.84	1.00

Mean item complexity = 1.5

Test of the hypothesis that 4 components are sufficient.

The root mean square of the residuals (RMSR) is 0.06

In the unrotated solution, most variables loaded on the first component. After rotation, there are four clear components/scales. Further, there is clear (or at least reasonable) component/scale membership for each item. This table lists all factor loadings that are greater than 0.30. When an item has multiple factor loadings listed, we inspect it for “cross-loading.” We observe cross-loadings with the following items: Obj6, Marg6, Ang1.

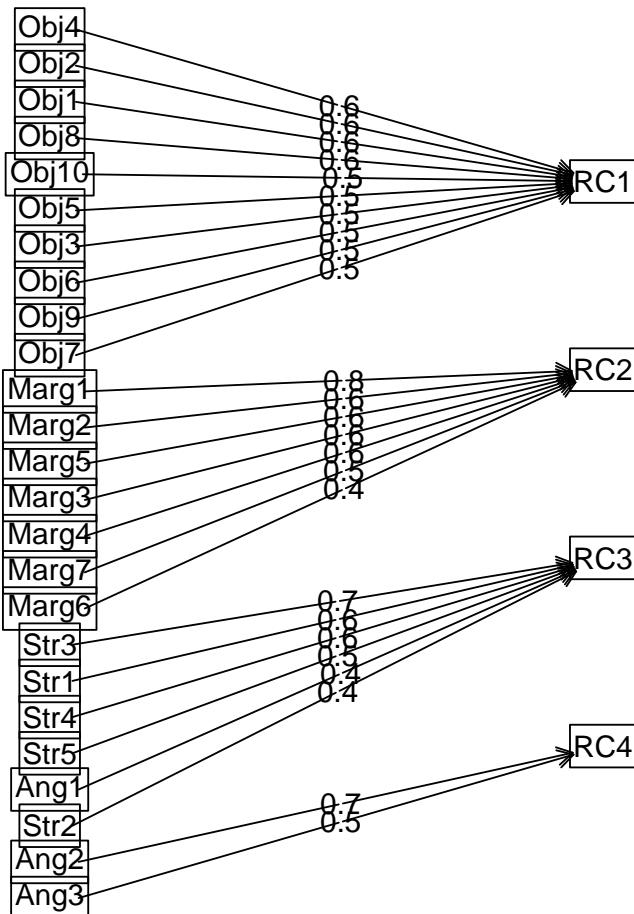
If this were a new scale and we had not yet established ideas for subscales, the next step would be to examine the items, themselves, and try to name the scales/components. If our scale construction included a priori/planned subscales, this is where we hope that the items fall where they were hypothesized to do so. Our simulated data worked pretty well, and with the exception of one item (i.e., Ang1) replicated the four scales that Lewis and Neville [2015] reported in the article.

- Assumptions of Beauty and Sexual Objectification
  - Silenced and Marginalized
  - Strong Woman Stereotype
  - Angry Woman Stereotype

We can also create a figure of the result.

```
psych::fa.diagram(pcaORTH)
```

## Components Analysis



We can extract the component loadings and write them to a table. This can be useful in preparing an APA style table for a manuscript or presentation.

```
names(pcaORTH)
```

```
[1] "values"      "rotation"     "n.obs"        "communality"   "loadings"
[6] "fit"         "fit.off"       "fn"           "Call"          "uniquenesses"
[11] "complexity"  "valid"         "chi"          "EPVAL"         "R2"
[16] "objective"   "residual"      "rms"          "factors"       "dof"
[21] "null.dof"    "null.model"    "criteria"     "STATISTIC"    "PVAL"
[26] "weights"     "r.scores"      "rot.mat"      "Vaccounted"   "Structure"
[31] "scores"
```

```
pcaORTH_table <- round(pcaORTH$loadings, 3)
write.table(pcaORTH_table, file = "pcaORTH_table.csv", sep = ",",
            row.names = FALSE)
pcaORTH_table
```

Loadings:

	RC1	RC2	RC3	RC4
Obj1	0.574	0.121	0.131	
Obj2	0.581	0.181		
Obj3	0.531	0.125	0.137	
Obj4	0.604	0.200		
Obj5	0.532	0.176	0.202	
Obj6	0.489		0.428	
Obj7	0.457	0.278	0.128	0.150
Obj8	0.572	0.273		-0.260
Obj9	0.468		0.210	
Obj10	0.545			0.170
Marg1	0.111	0.784		
Marg2		0.615	0.284	0.185
Marg3	0.134	0.596	0.155	
Marg4	0.237	0.559		0.113
Marg5	0.201	0.601	0.146	
Marg6	0.368	0.403		0.176
Marg7	0.102	0.511		0.121
Str1	0.157	0.192	0.648	-0.117
Str2	0.271		0.381	0.238
Str3			0.660	
Str4	0.142		0.574	-0.124
Str5	0.104		0.470	0.250
Ang1		0.196	0.444	0.367
Ang2		0.195		0.732
Ang3		0.245	0.118	0.549

	RC1	RC2	RC3	RC4
SS loadings	3.310	2.916	2.045	1.559
Proportion Var	0.132	0.117	0.082	0.062
Cumulative Var	0.132	0.249	0.331	0.393

#### 8.5.4.2 Oblique rotation

Whereas the orthogonal rotation sought to maximize the independence/unrelatedness of the components, an oblique rotation will allow them to be correlated. Researchers often explore both solutions but then only report one.

```
# pcaOBL <- psych::principal(GRMSmatrix, nfactors = 4, rotate =
# 'oblimin')
pcaOBL <- psych::principal(dfGRMS, nfactors = 4, rotate = "oblimin")
```

Loading required namespace: GPArotation

```
pcaOBL
```

### Principal Components Analysis

Call: psych::principal(r = dfGRMS, nfactors = 4, rotate = "oblimin")

Standardized loadings (pattern matrix) based upon correlation matrix

	TC2	TC1	TC3	TC4	h2	u2	com
Obj1	0.57	0.01	0.09	0.03	0.37	0.63	1.1
Obj2	0.57	0.08	0.05	0.04	0.39	0.61	1.1
Obj3	0.53	0.02	0.10	0.02	0.32	0.68	1.1
Obj4	0.60	0.11	-0.05	-0.04	0.40	0.60	1.1
Obj5	0.51	0.09	0.17	-0.15	0.36	0.64	1.5
Obj6	0.53	-0.17	-0.05	0.40	0.42	0.58	2.1
Obj7	0.42	0.19	0.07	0.10	0.32	0.68	1.6
Obj8	0.55	0.22	-0.05	-0.32	0.47	0.53	2.0
Obj9	0.46	-0.04	0.19	-0.09	0.27	0.73	1.4
Obj10	0.58	-0.09	-0.04	0.13	0.33	0.67	1.2
Marg1	-0.03	0.81	-0.03	0.01	0.63	0.37	1.0
Marg2	-0.04	0.60	0.21	0.14	0.50	0.50	1.4
Marg3	0.02	0.60	0.08	-0.02	0.40	0.60	1.0
Marg4	0.16	0.55	-0.16	0.07	0.39	0.61	1.4
Marg5	0.09	0.59	0.07	0.00	0.43	0.57	1.1
Marg6	0.32	0.35	-0.04	0.13	0.33	0.67	2.3
Marg7	0.02	0.52	-0.06	0.09	0.29	0.71	1.1
Str1	0.06	0.13	0.64	-0.15	0.50	0.50	1.2
Str2	0.24	-0.07	0.37	0.21	0.28	0.72	2.5
Str3	-0.02	-0.03	0.67	0.07	0.45	0.55	1.0
Str4	0.09	-0.04	0.59	-0.15	0.37	0.63	1.2
Str5	0.05	-0.02	0.46	0.23	0.30	0.70	1.5
Ang1	-0.12	0.14	0.42	0.36	0.37	0.63	2.4
Ang2	0.01	0.13	-0.04	0.73	0.57	0.43	1.1
Ang3	0.05	0.18	0.07	0.54	0.38	0.62	1.3

	TC2	TC1	TC3	TC4
SS loadings	3.34	2.90	2.05	1.55
Proportion Var	0.13	0.12	0.08	0.06
Cumulative Var	0.13	0.25	0.33	0.39
Proportion Explained	0.34	0.29	0.21	0.16
Cumulative Proportion	0.34	0.63	0.84	1.00

With component correlations of

TC2	TC1	TC3	TC4
TC2	1.00	0.35	0.20
TC1	0.35	1.00	0.24
TC3	0.20	0.24	1.00
TC4	0.10	0.16	0.09
			1.00

Mean item complexity = 1.4

Test of the hypothesis that 4 components are sufficient.

The root mean square of the residuals (RMSR) is 0.061  
with the empirical chi square 639.9 with prob < 0

Fit based upon off diagonal values = 0.89

We can make it a little easier to interpret by removing all factor loadings below .30.

```
psych::print.psych(pcaOBL, cut = 0.3, sort = TRUE)
```

## Principal Components Analysis

```
Call: psych::principal(r = dfGRMS, nfactors = 4, rotate = "oblimin")
```

Standardized loadings (pattern matrix) based upon correlation matrix

item	TC2	TC1	TC3	TC4	h2	u2	com
Obj4	4	0.60			0.40	0.60	1.1
Obj10	10	0.58			0.33	0.67	1.2
Obj1	1	0.57			0.37	0.63	1.1
Obj2	2	0.57			0.39	0.61	1.1
Obj8	8	0.55		-0.32	0.47	0.53	2.0
Obj6	6	0.53		0.40	0.42	0.58	2.1
Obj3	3	0.53			0.32	0.68	1.1
Obj5	5	0.51			0.36	0.64	1.5
Obj9	9	0.46			0.27	0.73	1.4
Obj7	7	0.42			0.32	0.68	1.6
Marg1	11		0.81		0.63	0.37	1.0
Marg3	13		0.60		0.40	0.60	1.0
Marg2	12		0.60		0.50	0.50	1.4
Marg5	15		0.59		0.43	0.57	1.1
Marg4	14		0.55		0.39	0.61	1.4
Marg7	17		0.52		0.29	0.71	1.1
Marg6	16	0.32	0.35		0.33	0.67	2.3
Str3	20			0.67	0.45	0.55	1.0
Str1	18			0.64	0.50	0.50	1.2
Str4	21			0.59	0.37	0.63	1.2
Str5	22			0.46	0.30	0.70	1.5
Ang1	23			0.42	0.36	0.37	0.63
Str2	19			0.37		0.28	0.72
Ang2	24				0.73	0.57	0.43
Ang3	25				0.54	0.38	0.62

	TC2	TC1	TC3	TC4
SS loadings	3.34	2.90	2.05	1.55
Proportion Var	0.13	0.12	0.08	0.06
Cumulative Var	0.13	0.25	0.33	0.39
Proportion Explained	0.34	0.29	0.21	0.16
Cumulative Proportion	0.34	0.63	0.84	1.00

With component correlations of

	TC2	TC1	TC3	TC4
TC2	1.00	0.35	0.20	0.10
TC1	0.35	1.00	0.24	0.16
TC3	0.20	0.24	1.00	0.09
TC4	0.10	0.16	0.09	1.00

Mean item complexity = 1.4

Test of the hypothesis that 4 components are sufficient.

The root mean square of the residuals (RMSR) is 0.06

Fit based upon off diagonal values = 0.89

The solution from the oblique rotation was similar to the orthogonal one. Note, though, that because our specification included “sort=TRUE” that the relative weights wiggled around and so the items are listed in a little different order.

Let's create a table and write it to a file.

```
pcaOBL_table <- round(pcaOBL$loadings, 3)
write.table(pcaOBL_table, file = "pcaOBL_table.csv", sep = ",", col.names = TRUE,
            row.names = FALSE)
pcaOBL_table
```

## Loadings:

	TC2	TC1	TC3	TC4
Obj1	0.574			
Obj2	0.574			
Obj3	0.526			
Obj4	0.604	0.108		
Obj5	0.509		0.166	-0.149
Obj6	0.533	-0.170		0.401
Obj7	0.424	0.192		0.103
Obj8	0.553	0.219		-0.318
Obj9	0.464		0.190	
Obj10	0.576			0.132
Marg1		0.810		
Marg2		0.600	0.211	0.145
Marg3		0.600		
Marg4	0.163	0.554	-0.155	
Marg5		0.592		
Marg6	0.319	0.349		0.131
Marg7		0.518		
Str1		0.129	0.644	-0.152

```

Str2  0.245      0.366  0.212
Str3           0.670
Str4           0.589 -0.148
Str5           0.463  0.234
Ang1 -0.116  0.140  0.423  0.356
Ang2           0.131      0.729
Ang3           0.183      0.535

          TC2   TC1   TC3   TC4
SS loadings 3.107 2.668 1.930 1.471
Proportion Var 0.124 0.107 0.077 0.059
Cumulative Var 0.124 0.231 0.308 0.367

```

The same four components/scales have emerged, but they are in different order.

The oblique rotation allows us to see the correlation between the components/scales. This was not available in the orthogonal rotation because the assumption of the orthogonal/varimax rotation is that the scales/components are uncorrelated; hence in the analysis they were fixed to 0.0.

We can see that all the scales have low to moderate (i.e., 0.09 to 0.35) correlations with each other.

Of course, there is always a little complexity. In oblique rotations, there is a distinction between the *pattern* matrix (which reports component loadings and is comparable to the matrix we interpreted for the orthogonal rotation) and the *structure* matrix (takes into account the relationship between the components/scales – it is a product of the pattern matrix and the matrix containing the correlation coefficients between the components/scales). Most interpret the pattern matrix because it is simpler; however, it could be that values in the pattern matrix are suppressed because of relations between the components. Therefore, the structure matrix can be a useful check and some editors will request it.

Obtaining the structure matrix requires two steps. First, we multiply the factor loadings with the phi matrix.

```

# names(pca0BL)
pca0BL$loadings %*% pca0BL$Phi

```

	TC2	TC1	TC3	TC4
Obj1	0.59854109	0.2370854	0.21076193	0.10026056
Obj2	0.61370763	0.2936877	0.18613777	0.11300577
Obj3	0.55625886	0.2340806	0.21133177	0.08838322
Obj4	0.62653372	0.2994708	0.09212284	0.03271529
Obj5	0.55782211	0.2813705	0.27627677	-0.06671857
Obj6	0.50436458	0.0686600	0.04959170	0.42362055
Obj7	0.51583784	0.3746086	0.21323217	0.18475919
Obj8	0.58631883	0.3483430	0.08569629	-0.22925474
Obj9	0.47739175	0.1500764	0.26407686	-0.03124475
Obj10	0.55033661	0.1232499	0.06974955	0.17330360
Marg1	0.25202377	0.7945638	0.16105543	0.13171661
Marg2	0.22669221	0.6607073	0.36285145	0.25811600

```
Marg3 0.24653097 0.6246027 0.23339896 0.08632620
Marg4 0.33257030 0.5843225 0.01980084 0.16383648
Marg5 0.31382840 0.6413860 0.23355689 0.10932345
Marg6 0.44535049 0.4708617 0.11768477 0.21681782
Marg7 0.19733852 0.5242792 0.07531732 0.17048225
Str1 0.22295271 0.2846102 0.67447904 -0.06303143
Str2 0.31665124 0.1416729 0.41881452 0.26069989
Str3 0.11375939 0.1383416 0.66583221 0.12771942
Str4 0.17266253 0.1046982 0.58122385 -0.09110045
Str5 0.16426964 0.1477531 0.49084330 0.27949111
Ang1 0.05479887 0.2604643 0.46720892 0.40679247
Ang2 0.12294975 0.2431195 0.06259901 0.74708242
Ang3 0.17778840 0.3025083 0.17023961 0.57585722
```

Next, we can use Field's [2012] function to produce the matrix.

```
# Field's function to produce the structure matrix
factor.structure <- function(fa, cut = 0.2, decimals = 2) {
  structure.matrix <- psych::fa.sort(fa$loadings %*% fa$Phi)
  structure.matrix <- data.frame(ifelse(abs(structure.matrix) < cut,
                                         "", round(structure.matrix, decimals)))
  return(structure.matrix)
}

factor.structure(pcaOBL, cut = 0.3)
```

	TC2	TC1	TC3	TC4
Obj4	0.63			
Obj2	0.61			
Obj1	0.6			
Obj8	0.59	0.35		
Obj5	0.56			
Obj3	0.56			
Obj10	0.55			
Obj7	0.52	0.37		
Obj6	0.5		0.42	
Obj9	0.48			
Marg1		0.79		
Marg2		0.66	0.36	
Marg5	0.31	0.64		
Marg3		0.62		
Marg4	0.33	0.58		
Marg7		0.52		
Marg6	0.45	0.47		
Str1			0.67	
Str3			0.67	
Str4			0.58	

Str5		0.49
Ang1		0.47 0.41
Str2	0.32	0.42
Ang2		0.75
Ang3	0.3	0.58

Although some of the relative values changed, our items were stable regarding their component membership.

### 8.5.5 Component Scores

Component *scores* (PC scores) can be created for each case (row) on each component (column). These can be used to assess the relative standing of one person on the construct/variable to another. We can also use them in regression (in place of means or sums) when groups of predictors correlate so highly that there is multicollinearity.

Computation involves multiplying an individual's item-level responses by the component loadings we obtained through the PCA process. The results will be one score per component for each row/case.

```
pcaOBL <- psych::principal(dfGRMS, nfactors = 4, rotate = "oblimin", scores = TRUE)
head(pcaOBL$scores, 10) #shows us only the first 10 (of N = 2571)
```

	TC2	TC1	TC3	TC4
[1,]	-0.6458500	-1.5519979	0.73544246	-0.76005053
[2,]	0.5116784	-1.0097579	0.74427743	0.29538913
[3,]	0.5825763	0.1327654	-0.03596272	1.40125523
[4,]	-1.1296840	-1.0820066	0.46203018	-2.08203840
[5,]	-0.3066491	0.7020903	-1.15731968	-0.08696116
[6,]	-0.6165563	0.5443947	-0.37332546	1.06336132
[7,]	0.3100935	0.7005788	-1.12949607	-0.99349900
[8,]	-1.7404874	-0.6389406	0.51456794	1.23689663
[9,]	-0.4801252	-1.0772976	-0.71350564	0.77814794
[10,]	0.3986271	-0.5906088	-0.71507356	0.80831256

```
dfGRMS <- cbind(dfGRMS, pcaOBL$scores) #adds them to our raw dataset
```

To bring this full circle, we can see the correlation of the component scores; the pattern maps onto what we saw previously.

```
psych::corr.test(dfGRMS[c("TC1", "TC4", "TC3", "TC2")])
```

```
Call:psych::corr.test(x = dfGRMS[c("TC1", "TC4", "TC3", "TC2")])
Correlation matrix
      TC1  TC4  TC3  TC2
TC1  1.00 0.16 0.24 0.35
```

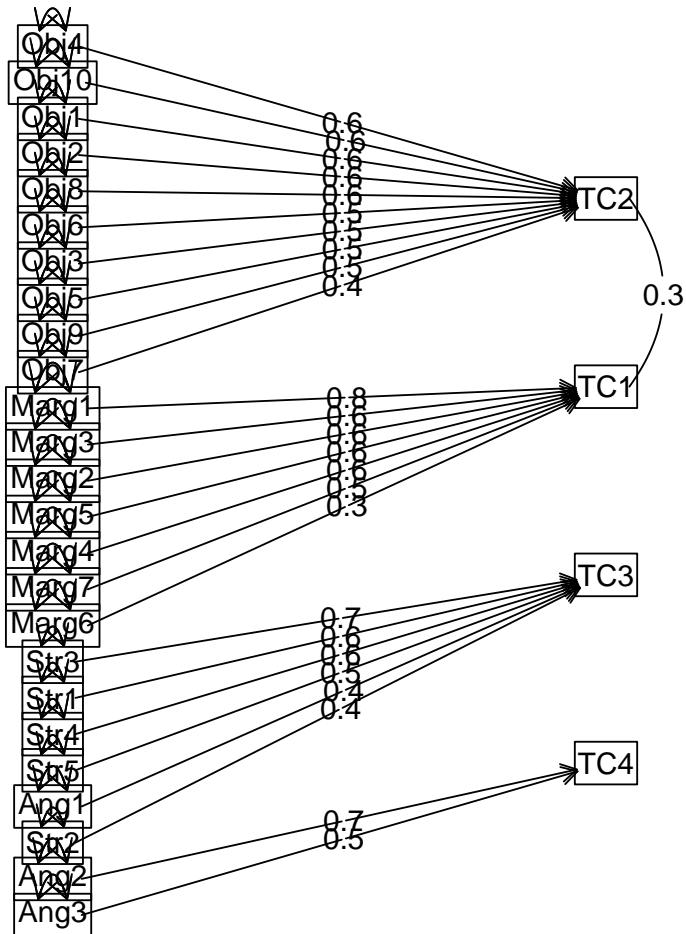
```
TC4 0.16 1.00 0.09 0.10
TC3 0.24 0.09 1.00 0.20
TC2 0.35 0.10 0.20 1.00
Sample Size
[1] 259
Probability values (Entries above the diagonal are adjusted for multiple tests.)
   TC1  TC4  TC3  TC2
TC1 0.00 0.03 0.00 0.00
TC4 0.01 0.00 0.19 0.19
TC3 0.00 0.13 0.00 0.00
TC2 0.00 0.10 0.00 0.00
```

To see confidence intervals of the correlations, print with the short=FALSE option

And now for a figure of the oblique rotation. Note that figure includes semi-circles between TC1/TC2 and TC1/TC4. These represent significant correlation coefficients between the components that are named. In contrast, the orthogonal rotation required the components to be uncorrelated.

```
psych::fa.diagram(pca0BL, error = TRUE, side = 3)
```

## Components Analysis



## 8.6 APA Style Results

### Results

The dimensionality of the 25 items from the Gendered Racial Microaggressions Scale for Black Women was analyzed using principal components analysis. Data screening were conducted to determine the suitability of the data for principal components analyses. The Kaiser-Meyer-Olkin measure of sampling adequacy (KMO; Kaiser, 1970) represents the ratio of the squared correlation between variables to the squared partial correlation between variables. KMO ranges from 0.00 to 1.00; values closer to 1.00 indicate that the patterns of correlations are relatively compact, and that component analysis should yield distinct and reliable components (Field, 2012). In our dataset, the KMO value was .85, indicating acceptable sampling adequacy. The Bartlett's Test of Sphericity examines whether the population correlation matrix resembles an identity matrix (Field, 2012). When the  $p$  value for the Bartlett's test is  $< .05$ , we are fairly certain we have clusters of correlated variables. In our dataset,  $\chi^2(300) = 1217.508, p < .001$ ,

indicating the correlations between items are sufficiently large enough for principal components analysis. The determinant of the correlation matrix alerts us to any issues of multicollinearity or singularity and should be larger than 0.00001. Our determinant was 0.0075, supporting the suitability of our data for analysis.

Four criteria were used to determine the number of components to extract: a priori theory, the scree test, the eigenvalue-greater-than-one criteria, and the interpretability of the solution. Kaiser's eigenvalue-greater-than-one criteria suggested seven components, and, in combination explained 39% of the variance. The inflexion in the scree plot suggested retaining between one and four components. Considering the a priori theory obtained from the original psychometric article [Lewis and Neville, 2015], four components were extracted. We investigated each with orthogonal (varimax) and oblique (oblimin) procedures. Given the low-to-moderate correlations (ranging from 0.09 to 0.35) and the clear component loadings, we determined that an oblique solution was most appropriate.

The rotated solution, as shown in Table 1 and Figure 1, yielded four interpretable components, each listed with the proportion of variance accounted for: assumptions of beauty and sexual objectification (13%), silenced and marginalized (12%), strong woman stereotype (8%), and angry woman stereotype (6%).

Regarding the Table 1, I would include a table with all the values, bolding those with component membership. This is easy, though, because it is how the table was exported when we wrote it to a .csv file.

## 8.7 Back to the FutuRe: The relationship between PCA and item analysis

Earlier in the ReCentering Psych Stats OER I included the lesson on [item analysis](#) because I find it to be a useful stepping stone into principal components and principal factor analyses.

Nearing the end of the lesson on PCA, we can ask, “How do the results we obtained from PCA compare to those found in item analysis?” To answer these questions, we will (a) calculate corrected item-total correlation coefficients, (c) calculate correlations between each item and the mean scores from the remaining scales, (c) calculate component loadings for a PCA with an orthogonal rotation, and (d) calculate component loadings for a PCA with an oblique rotation. I will teach the last step – assembling them into a single table – in R. The code is complicated, and many might choose to do it in a spreadsheet, outside of the R environment. After assembly, though, we can compare the results.

First, we score the total and subscales using the dataset we simulated above (dfGRMS).

```
library(tidyverse)
GRMSVars <- c("Obj1", "Obj2", "Obj3", "Obj4", "Obj5", "Obj6", "Obj7", "Obj8",
              "Obj9", "Obj10", "Marg1", "Marg2", "Marg3", "Marg4", "Marg5", "Marg6",
              "Marg7", "Str1", "Str2", "Str3", "Str4", "Str5", "Ang1", "Ang2", "Ang3")
```

```

ObjectifiedVars <- c("Obj1", "Obj2", "Obj3", "Obj4", "Obj5", "Obj6", "Obj7",
  "Obj8", "Obj9", "Obj10")
MarginalizedVars <- c("Marg1", "Marg2", "Marg3", "Marg4", "Marg5", "Marg6",
  "Marg7")
StrongVars <- c("Str1", "Str2", "Str3", "Str4", "Str5")
AngryVars <- c("Ang1", "Ang2", "Ang3")

dfGRMS$GRMStot <- sjstats::mean_n(dfGRMS[, GRMSVars], 0.8) #will create the mean for each individual
dfGRMS$Objectified <- sjstats::mean_n(dfGRMS[, ObjectifiedVars], 0.8) #will create the mean for each item
dfGRMS$Marginalized <- sjstats::mean_n(dfGRMS[, MarginalizedVars], 0.8) #will create the mean for each category
dfGRMS$Strong <- sjstats::mean_n(dfGRMS[, StrongVars], 0.8) #will create the mean for each item
dfGRMS$Angry <- sjstats::mean_n(dfGRMS[, AngryVars], 0.8) #will create the mean for each individual

```

While we are at it, let's just create tiny dfs with just our variables of interest.

```

GRMStotal <- dplyr::select(dfGRMS, Obj1:Ang3)
Objectification <- dplyr::select(dfGRMS, Obj1:Obj10)
Marginalization <- dplyr::select(dfGRMS, Marg1:Marg7)
Strong <- dplyr::select(dfGRMS, Str1:Str5)
Angry <- dplyr::select(dfGRMS, Ang1:Ang3)

```

## 8.7.1 Calculating and Extracting Item-Total Correlation Coefficients

### 8.7.1.1 Corrected item-total correlations from the *psych::alpha()*

Let's first ask, “Is there support for this instrument as a unidimensional measure?” To do that, we get an alpha for the whole scale score.

```

GRMSalpha <- psych::alpha(GRMStotal) #creating an object from this analysis so I can extract
GRMSalpha

```

Reliability analysis

Call: psych::alpha(x = GRMStotal)

raw_alpha	std.alpha	G6(smc)	average_r	S/N	ase	mean	sd	median_r
0.84	0.84	0.86	0.18	5.3	0.014	2.5	0.44	0.18

95% confidence boundaries

lower alpha upper

Feldt 0.81 0.84 0.87

Duhachek 0.81 0.84 0.87

Reliability if an item is dropped:

	raw_alpha	std.alpha	G6(smc)	average_r	S/N	alpha	se	var.r	med.r
Obj1	0.83	0.83	0.85	0.17	5.0	0.015	0.0066	0.17	

Obj2	0.83	0.83	0.85	0.17	5.0	0.015	0.0066	0.17
Obj3	0.83	0.84	0.85	0.17	5.1	0.015	0.0068	0.17
Obj4	0.83	0.83	0.85	0.17	5.0	0.015	0.0065	0.17
Obj5	0.83	0.83	0.85	0.17	5.0	0.015	0.0066	0.17
Obj6	0.84	0.84	0.85	0.18	5.2	0.015	0.0067	0.18
Obj7	0.83	0.83	0.85	0.17	5.0	0.015	0.0066	0.17
Obj8	0.84	0.84	0.85	0.18	5.1	0.015	0.0063	0.18
Obj9	0.84	0.84	0.85	0.18	5.2	0.015	0.0067	0.18
Obj10	0.84	0.84	0.85	0.18	5.2	0.015	0.0067	0.18
Marg1	0.83	0.83	0.85	0.17	5.0	0.015	0.0059	0.17
Marg2	0.83	0.83	0.85	0.17	5.0	0.015	0.0063	0.17
Marg3	0.83	0.83	0.85	0.17	5.1	0.015	0.0065	0.17
Marg4	0.83	0.84	0.85	0.17	5.1	0.015	0.0064	0.17
Marg5	0.83	0.83	0.85	0.17	5.0	0.015	0.0065	0.17
Marg6	0.83	0.83	0.85	0.17	5.0	0.015	0.0066	0.17
Marg7	0.84	0.84	0.85	0.18	5.2	0.015	0.0066	0.18
Str1	0.84	0.84	0.85	0.18	5.1	0.015	0.0068	0.17
Str2	0.84	0.84	0.85	0.18	5.2	0.015	0.0070	0.18
Str3	0.84	0.84	0.86	0.18	5.2	0.014	0.0066	0.18
Str4	0.84	0.84	0.86	0.18	5.3	0.014	0.0064	0.18
Str5	0.84	0.84	0.86	0.18	5.2	0.014	0.0067	0.18
Ang1	0.84	0.84	0.85	0.18	5.2	0.014	0.0067	0.18
Ang2	0.84	0.84	0.86	0.18	5.3	0.014	0.0064	0.18
Ang3	0.84	0.84	0.85	0.18	5.2	0.015	0.0067	0.18

## Item statistics

	n	raw.r	std.r	r.cor	r.drop	mean	sd	
Obj1	259	0.50	0.51	0.48	0.44	2.8	0.87	
Obj2	259	0.53	0.54	0.51	0.47	2.6	0.91	
Obj3	259	0.49	0.49	0.46	0.41	2.6	0.99	
Obj4	259	0.51	0.51	0.49	0.44	2.6	1.05	
Obj5	259	0.50	0.50	0.47	0.43	2.3	0.94	
Obj6	259	0.41	0.41	0.37	0.34	2.4	0.93	
Obj7	259	0.55	0.54	0.52	0.47	2.5	1.20	
Obj8	259	0.45	0.46	0.43	0.38	2.5	0.90	
Obj9	259	0.41	0.41	0.37	0.34	2.7	0.97	
Obj10	259	0.41	0.42	0.38	0.35	2.2	0.83	
Marg1	259	0.55	0.55	0.54	0.49	2.7	0.91	
Marg2	259	0.56	0.56	0.55	0.50	2.8	0.89	
Marg3	259	0.50	0.50	0.47	0.42	2.4	1.05	
Marg4	259	0.47	0.48	0.45	0.41	2.5	0.87	
Marg5	259	0.54	0.54	0.52	0.47	2.3	0.92	
Marg6	259	0.52	0.53	0.50	0.46	2.7	0.94	
Marg7	259	0.41	0.41	0.36	0.33	2.5	1.04	
Str1	259	0.44	0.45	0.41	0.37	2.7	0.93	
Str2	259	0.42	0.42	0.38	0.34	2.6	0.91	
Str3	259	0.37	0.37	0.32	0.29	2.3	0.98	
Str4	259	0.31	0.32	0.27	0.24	2.2	0.80	

```
Str5  259  0.38  0.37  0.32   0.29  2.5 1.05
Ang1  259  0.39  0.38  0.34   0.31  2.6 1.04
Ang2  259  0.36  0.35  0.31   0.28  2.3 1.05
Ang3  259  0.40  0.40  0.36   0.32  2.4 0.91
```

Non missing response frequency for each item

	0	1	2	3	4	5	miss
Obj1	0.01	0.07	0.27	0.49	0.16	0.01	0
Obj2	0.01	0.07	0.36	0.41	0.12	0.03	0
Obj3	0.02	0.11	0.32	0.39	0.14	0.02	0
Obj4	0.01	0.14	0.32	0.33	0.16	0.04	0
Obj5	0.02	0.16	0.39	0.33	0.09	0.01	0
Obj6	0.01	0.15	0.37	0.36	0.09	0.01	0
Obj7	0.04	0.16	0.31	0.29	0.13	0.06	0
Obj8	0.02	0.11	0.36	0.42	0.09	0.01	0
Obj9	0.01	0.10	0.33	0.37	0.17	0.02	0
Obj10	0.02	0.15	0.47	0.31	0.04	0.00	0
Marg1	0.00	0.07	0.34	0.39	0.17	0.02	0
Marg2	0.00	0.07	0.29	0.44	0.18	0.01	0
Marg3	0.03	0.16	0.32	0.34	0.12	0.02	0
Marg4	0.01	0.11	0.39	0.39	0.10	0.00	0
Marg5	0.01	0.18	0.39	0.33	0.08	0.01	0
Marg6	0.01	0.09	0.31	0.40	0.18	0.02	0
Marg7	0.02	0.16	0.28	0.39	0.13	0.03	0
Str1	0.01	0.06	0.36	0.37	0.17	0.03	0
Str2	0.01	0.08	0.36	0.40	0.14	0.02	0
Str3	0.02	0.17	0.42	0.27	0.12	0.01	0
Str4	0.01	0.14	0.52	0.28	0.05	0.00	0
Str5	0.02	0.15	0.31	0.32	0.20	0.00	0
Ang1	0.02	0.14	0.31	0.36	0.15	0.03	0
Ang2	0.03	0.22	0.32	0.33	0.08	0.02	0
Ang3	0.02	0.13	0.39	0.38	0.07	0.01	0

Alpha for the total scale score is 0.84.

And now each of the subscales:

```
ObjAlpha <- psych::alpha(Objetification) #creating an object from this analysis so I can ext...
```

Reliability analysis

Call: psych::alpha(x = Objetification)

raw_alpha	std.alpha	G6(smc)	average_r	S/N	ase	mean	sd	median_r
0.76	0.77	0.76	0.25	3.3	0.022	2.5	0.55	0.25

95% confidence boundaries

	lower	alpha	upper
Feldt	0.72	0.76	0.81
Duhachek	0.72	0.76	0.81

Reliability if an item is dropped:

	raw_alpha	std.alpha	G6(smc)	average_r	S/N	alpha	se	var.r	med.r
Obj1	0.74	0.74	0.73	0.24	2.9	0.024	0.0030	0.25	
Obj2	0.74	0.74	0.73	0.24	2.8	0.024	0.0032	0.24	
Obj3	0.75	0.75	0.73	0.25	3.0	0.024	0.0036	0.25	
Obj4	0.74	0.74	0.72	0.24	2.9	0.024	0.0032	0.25	
Obj5	0.74	0.75	0.73	0.25	2.9	0.024	0.0028	0.25	
Obj6	0.76	0.76	0.74	0.26	3.2	0.023	0.0026	0.26	
Obj7	0.75	0.75	0.73	0.25	3.0	0.023	0.0032	0.25	
Obj8	0.74	0.74	0.73	0.24	2.9	0.024	0.0030	0.25	
Obj9	0.75	0.76	0.74	0.26	3.1	0.023	0.0033	0.26	
Obj10	0.75	0.75	0.74	0.25	3.1	0.023	0.0032	0.25	

Item statistics

	n	raw.r	std.r	r.cor	r.drop	mean	sd
Obj1	259	0.59	0.60	0.54	0.47	2.8	0.87
Obj2	259	0.61	0.62	0.56	0.49	2.6	0.91
Obj3	259	0.57	0.57	0.49	0.43	2.6	0.99
Obj4	259	0.63	0.61	0.56	0.49	2.6	1.05
Obj5	259	0.58	0.58	0.51	0.44	2.3	0.94
Obj6	259	0.49	0.50	0.40	0.35	2.4	0.93
Obj7	259	0.60	0.57	0.50	0.43	2.5	1.20
Obj8	259	0.58	0.59	0.52	0.45	2.5	0.90
Obj9	259	0.52	0.51	0.42	0.37	2.7	0.97
Obj10	259	0.52	0.53	0.44	0.39	2.2	0.83

Non missing response frequency for each item

	0	1	2	3	4	5	miss
Obj1	0.01	0.07	0.27	0.49	0.16	0.01	0
Obj2	0.01	0.07	0.36	0.41	0.12	0.03	0
Obj3	0.02	0.11	0.32	0.39	0.14	0.02	0
Obj4	0.01	0.14	0.32	0.33	0.16	0.04	0
Obj5	0.02	0.16	0.39	0.33	0.09	0.01	0
Obj6	0.01	0.15	0.37	0.36	0.09	0.01	0
Obj7	0.04	0.16	0.31	0.29	0.13	0.06	0
Obj8	0.02	0.11	0.36	0.42	0.09	0.01	0
Obj9	0.01	0.10	0.33	0.37	0.17	0.02	0
Obj10	0.02	0.15	0.47	0.31	0.04	0.00	0

Alpha for the Assumptions of Beauty and Sexual Objectification scale is 0.77.

```
MargAlpha <- psych::alpha(Marginalization) #creating an object from this analysis so I can ex
```

Reliability analysis

Call: psych::alpha(x = Marginalization)

raw_alpha	std.alpha	G6(smc)	average_r	S/N	ase	mean	sd	median_r
0.75	0.75	0.73		0.3	3	0.024	2.6	0.6
								0.3

95% confidence boundaries

	lower alpha	upper
Feldt	0.7	0.75
Duhachek	0.7	0.75
	0.79	0.80

Reliability if an item is dropped:

	raw_alpha	std.alpha	G6(smc)	average_r	S/N	alpha	se	var.r	med.r
Marg1	0.69	0.69	0.66		0.27	2.3	0.030	0.0038	0.27
Marg2	0.71	0.71	0.68		0.29	2.5	0.028	0.0054	0.29
Marg3	0.72	0.73	0.70		0.31	2.6	0.027	0.0059	0.30
Marg4	0.73	0.73	0.70		0.31	2.8	0.026	0.0055	0.31
Marg5	0.71	0.72	0.69		0.30	2.5	0.028	0.0062	0.31
Marg6	0.73	0.74	0.71		0.32	2.8	0.026	0.0064	0.32
Marg7	0.74	0.74	0.72		0.33	2.9	0.025	0.0045	0.32

Item statistics

	n	raw.r	std.r	r.cor	r.drop	mean	sd
Marg1	259	0.73	0.74	0.69	0.60	2.7	0.91
Marg2	259	0.67	0.68	0.60	0.52	2.8	0.89
Marg3	259	0.64	0.63	0.53	0.46	2.4	1.05
Marg4	259	0.58	0.60	0.49	0.42	2.5	0.87
Marg5	259	0.65	0.66	0.58	0.50	2.3	0.92
Marg6	259	0.59	0.59	0.47	0.41	2.7	0.94
Marg7	259	0.58	0.56	0.43	0.37	2.5	1.04

Non missing response frequency for each item

	0	1	2	3	4	5	miss
Marg1	0.00	0.07	0.34	0.39	0.17	0.02	0
Marg2	0.00	0.07	0.29	0.44	0.18	0.01	0
Marg3	0.03	0.16	0.32	0.34	0.12	0.02	0
Marg4	0.01	0.11	0.39	0.39	0.10	0.00	0
Marg5	0.01	0.18	0.39	0.33	0.08	0.01	0
Marg6	0.01	0.09	0.31	0.40	0.18	0.02	0
Marg7	0.02	0.16	0.28	0.39	0.13	0.03	0

Alpha for the Silenced and Marginalized Scale is 0.75.

```
StrongAlpha <- psych::alpha(Strong) #creating an object from this analysis so I can extract at
```

Reliability analysis

Call: psych::alpha(x = Strong)

raw_alpha	std.alpha	G6(smc)	average_r	S/N	ase	mean	sd	median_r	
0.56	0.57	0.52		0.21	1.3	0.043	2.5	0.56	0.21

95% confidence boundaries

	lower	alpha	upper
--	-------	-------	-------

Feldt	0.47	0.56	0.64
Duhachek	0.48	0.56	0.64

Reliability if an item is dropped:

	raw_alpha	std.alpha	G6(smc)	average_r	S/N	alpha	se	var.r	med.r
Str1	0.46	0.47	0.40		0.18	0.89	0.054	0.0032	0.19
Str2	0.53	0.53	0.47		0.22	1.13	0.048	0.0042	0.23
Str3	0.47	0.48	0.41		0.19	0.91	0.053	0.0027	0.21
Str4	0.51	0.51	0.45		0.21	1.05	0.050	0.0035	0.21
Str5	0.55	0.55	0.48		0.24	1.23	0.045	0.0017	0.22

Item statistics

	n	raw.r	std.r	r.cor	r.drop	mean	sd
Str1	259	0.65	0.65	0.52	0.39	2.7	0.93
Str2	259	0.57	0.58	0.38	0.28	2.6	0.91
Str3	259	0.65	0.64	0.51	0.38	2.3	0.98
Str4	259	0.56	0.60	0.43	0.32	2.2	0.80
Str5	259	0.58	0.55	0.33	0.25	2.5	1.05

Non missing response frequency for each item

	0	1	2	3	4	5	miss
Str1	0.01	0.06	0.36	0.37	0.17	0.03	0
Str2	0.01	0.08	0.36	0.40	0.14	0.02	0
Str3	0.02	0.17	0.42	0.27	0.12	0.01	0
Str4	0.01	0.14	0.52	0.28	0.05	0.00	0
Str5	0.02	0.15	0.31	0.32	0.20	0.00	0

Alpha for the Strong Black Woman Stereotype is 0.57.

```
AngryAlpha <- psych::alpha(Angry) #creating an object from this analysis so I can extract and
AngryAlpha
```

Reliability analysis

Call: psych::alpha(x = Angry)

raw_alpha	std.alpha	G6(smc)	average_r	S/N	ase	mean	sd	median_r	
0.49	0.49	0.39		0.24	0.95	0.055	2.4	0.7	0.24

```

95% confidence boundaries
      lower alpha upper
Feldt     0.37  0.49  0.59
Duhachek  0.38  0.49  0.59

Reliability if an item is dropped:
    raw_alpha std.alpha G6(smc) average_r  S/N alpha se var.r med.r
Ang1      0.40      0.40    0.25      0.25 0.67    0.074    NA  0.25
Ang2      0.37      0.38    0.23      0.23 0.60    0.077    NA  0.23
Ang3      0.39      0.39    0.24      0.24 0.63    0.076    NA  0.24

Item statistics
      n raw.r std.r r.cor r.drop mean   sd
Ang1 259  0.71  0.70  0.43   0.30  2.6 1.04
Ang2 259  0.72  0.71  0.45   0.31  2.3 1.05
Ang3 259  0.67  0.70  0.44   0.31  2.4 0.91

Non missing response frequency for each item
      0   1   2   3   4   5 miss
Ang1 0.02 0.14 0.31 0.36 0.15 0.03   0
Ang2 0.03 0.22 0.32 0.33 0.08 0.02   0
Ang3 0.02 0.13 0.39 0.38 0.07 0.01   0

```

Alpha for the Angry Black Woman Stereotypes is 0.49.

### 8.7.1.2 Correlating items with other subscale totals

```

Obj_othR <- psych::corr.test(dfGRMS[c("Obj1", "Obj2", "Obj3", "Obj4", "Obj5",
                                         "Obj6", "Obj7", "Obj8", "Obj9", "Obj10", "Marginalized", "Strong",
                                         "Angry")])
Obj_othR

```

```

Call:psych::corr.test(x = dfGRMS[c("Obj1", "Obj2", "Obj3", "Obj4",
                                    "Obj5", "Obj6", "Obj7", "Obj8", "Obj9", "Obj10", "Marginalized",
                                    "Strong", "Angry")])
Correlation matrix
          Obj1 Obj2 Obj3 Obj4 Obj5 Obj6 Obj7 Obj8 Obj9 Obj10 Marginalized
Obj1      1.00 0.35 0.25 0.27 0.28 0.25 0.28 0.35 0.15 0.24      0.28
Obj2      0.35 1.00 0.31 0.25 0.27 0.23 0.31 0.28 0.26 0.24      0.35
Obj3      0.25 0.31 1.00 0.24 0.28 0.28 0.20 0.25 0.21 0.22      0.30
Obj4      0.27 0.25 0.24 1.00 0.39 0.23 0.28 0.30 0.26 0.28      0.33
Obj5      0.28 0.27 0.28 0.39 1.00 0.15 0.18 0.29 0.25 0.20      0.30
Obj6      0.25 0.23 0.28 0.23 0.15 1.00 0.20 0.14 0.21 0.12      0.22
Obj7      0.28 0.31 0.20 0.28 0.18 0.20 1.00 0.31 0.19 0.28      0.36
Obj8      0.35 0.28 0.25 0.30 0.29 0.14 0.31 1.00 0.19 0.23      0.31

```

Probability values (Entries above the diagonal are adjusted for multiple tests.)

```
Strong      0.00  0.00
Angry       0.00  0.00
```

To see confidence intervals of the correlations, print with the short=FALSE option

```
Marg_othR <- psych::corr.test(dfGRMS[c("Marg1", "Marg2", "Marg3", "Marg4",
                                         "Marg5", "Marg6", "Marg7", "Objectified", "Strong", "Angry")])
Marg_othR
```

```
Call:psych::corr.test(x = dfGRMS[c("Marg1", "Marg2", "Marg3", "Marg4",
                                   "Marg5", "Marg6", "Marg7", "Objectified", "Strong", "Angry")])
```

Correlation matrix

	Marg1	Marg2	Marg3	Marg4	Marg5	Marg6	Marg7	Objectified	Strong	Angry
Marg1	1.00	0.43	0.41	0.38	0.41	0.35	0.31	0.33	0.21	0.28
Marg2	0.43	1.00	0.35	0.23	0.40	0.27	0.33	0.32	0.32	0.33
Marg3	0.41	0.35	1.00	0.32	0.25	0.25	0.20	0.30	0.22	0.23
Marg4	0.38	0.23	0.32	1.00	0.30	0.26	0.16	0.33	0.15	0.24
Marg5	0.41	0.40	0.25	0.30	1.00	0.29	0.28	0.36	0.26	0.22
Marg6	0.35	0.27	0.25	0.26	0.29	1.00	0.20	0.40	0.22	0.21
Marg7	0.31	0.33	0.20	0.16	0.28	0.20	1.00	0.25	0.13	0.19
Objectified	0.33	0.32	0.30	0.33	0.36	0.40	0.25	1.00	0.38	0.26
Strong	0.21	0.32	0.22	0.15	0.26	0.22	0.13	0.38	1.00	0.30
Angry	0.28	0.33	0.23	0.24	0.22	0.21	0.19	0.26	0.30	1.00

Sample Size

```
[1] 259
```

Probability values (Entries above the diagonal are adjusted for multiple tests.)

	Marg1	Marg2	Marg3	Marg4	Marg5	Marg6	Marg7	Objectified	Strong	Angry
Marg1	0	0	0	0.00	0	0	0.00	0	0.01	0.00
Marg2	0	0	0	0.00	0	0	0.00	0	0.00	0.00
Marg3	0	0	0	0.00	0	0	0.01	0	0.00	0.00
Marg4	0	0	0	0.00	0	0	0.02	0	0.03	0.00
Marg5	0	0	0	0.00	0	0	0.00	0	0.00	0.00
Marg6	0	0	0	0.00	0	0	0.01	0	0.00	0.01
Marg7	0	0	0	0.01	0	0	0.00	0	0.04	0.01
Objectified	0	0	0	0.00	0	0	0.00	0	0.00	0.00
Strong	0	0	0	0.02	0	0	0.04	0	0.00	0.00
Angry	0	0	0	0.00	0	0	0.00	0	0.00	0.00

To see confidence intervals of the correlations, print with the short=FALSE option

```
Str_othR <- psych::corr.test(dfGRMS[c("Str1", "Str2", "Str3", "Str4", "Str5",
                                         "Objectified", "Marginalized", "Angry")])
Str_othR
```

```
Call:psych::corr.test(x = dfGRMS[c("Str1", "Str2", "Str3", "Str4",
                                   "Str5", "Objectified", "Marginalized", "Angry")])
```

Correlation matrix

	Str1	Str2	Str3	Str4	Str5	Objectified	Marginalized	Angry
Str1	1.00	0.21	0.30	0.23	0.23	0.28	0.26	0.16
Str2	0.21	1.00	0.20	0.20	0.12	0.28	0.24	0.21
Str3	0.30	0.20	1.00	0.27	0.18	0.18	0.18	0.20
Str4	0.23	0.20	0.27	1.00	0.12	0.18	0.14	0.10
Str5	0.23	0.12	0.18	0.12	1.00	0.21	0.20	0.23
Objectified	0.28	0.28	0.18	0.18	0.21	1.00	0.51	0.26
Marginalized	0.26	0.24	0.18	0.14	0.20	0.51	1.00	0.38
Angry	0.16	0.21	0.20	0.10	0.23	0.26	0.38	1.00

Sample Size

[1] 259

Probability values (Entries above the diagonal are adjusted for multiple tests.)

	Str1	Str2	Str3	Str4	Str5	Objectified	Marginalized	Angry
Str1	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.05
Str2	0.00	0.00	0.02	0.01	0.15	0.00	0.00	0.01
Str3	0.00	0.00	0.00	0.00	0.03	0.03	0.03	0.02
Str4	0.00	0.00	0.00	0.00	0.15	0.03	0.10	0.15
Str5	0.00	0.05	0.00	0.06	0.00	0.01	0.02	0.00
Objectified	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Marginalized	0.00	0.00	0.00	0.02	0.00	0.00	0.00	0.00
Angry	0.01	0.00	0.00	0.11	0.00	0.00	0.00	0.00

To see confidence intervals of the correlations, print with the short=FALSE option

```
Ang_othR <- psych::corr.test(dfGRMS[c("Ang1", "Ang2", "Ang3", "Objectified",
                                         "Marginalized", "Strong")])
Ang_othR
```

```
Call:psych::corr.test(x = dfGRMS[c("Ang1", "Ang2", "Ang3", "Objectified",
                                    "Marginalized", "Strong")])
```

Correlation matrix

	Ang1	Ang2	Ang3	Objectified	Marginalized	Strong
Ang1	1.00	0.24	0.23	0.16	0.25	0.31
Ang2	0.24	1.00	0.25	0.15	0.28	0.15
Ang3	0.23	0.25	1.00	0.23	0.28	0.18
Objectified	0.16	0.15	0.23	1.00	0.51	0.38
Marginalized	0.25	0.28	0.28	0.51	1.00	0.34
Strong	0.31	0.15	0.18	0.38	0.34	1.00

Sample Size

[1] 259

Probability values (Entries above the diagonal are adjusted for multiple tests.)

	Ang1	Ang2	Ang3	Objectified	Marginalized	Strong
Ang1	0.00	0.00	0	0.02	0	0.00
Ang2	0.00	0.00	0	0.03	0	0.03
Ang3	0.00	0.00	0	0.00	0	0.01
Objectified	0.01	0.01	0	0.00	0	0.00

Marginalized	0.00	0.00	0	0.00	0	0.00
Strong	0.00	0.02	0	0.00	0	0.00

To see confidence intervals of the correlations, print with the short=FALSE option

### 8.7.1.3 Extracting values, binding them together, and joining the files

```
# names(Obj_other) Extracting the item-level statistics from the
# alpha object
Obj_othR <- as.data.frame(Obj_othR$r) #Makes the item-total(other) correlation matrix a df
# Adding variable names so we don't get lost
Obj_othR$Items <- c("Obj1", "Obj2", "Obj3", "Obj4", "Obj5", "Obj6", "Obj7",
"Obj8", "Obj9", "Obj10", "Marginalized", "Strong", "Angry")
# deleting the ROWS with the total scale scores (the columns)
Obj_othR <- Obj_othR[!Obj_othR$Items == "Marginalized", ]
Obj_othR <- Obj_othR[!Obj_othR$Items == "Strong", ]
Obj_othR <- Obj_othR[!Obj_othR$Items == "Angry", ]
Obj_othR[, "Objectified"] <- NA #We need a column for this to bind the items, later.
Obj_othR <- dplyr::select(Obj_othR, Items, Objectified, Marginalized, Strong,
                         Angry) #Putting items in order
# Item Corrected Total Correlations
ObjAlpha <- as.data.frame(ObjAlpha$item.stats) #Grabbing the alpha object we created earlier
ObjAlpha$Items <- c("Obj1", "Obj2", "Obj3", "Obj4", "Obj5", "Obj6", "Obj7",
"Obj8", "Obj9", "Obj10")
# Joining the two and selecting the vars of interest
ObjStats <- full_join(ObjAlpha, Obj_othR, by = "Items")
ObjStats$Objectified <- ObjStats$r.drop #Copy the item-corrected total (r.drop) into the Objje
ObjStats <- dplyr::select(ObjStats, Items, Objectified, Marginalized, Strong,
                         Angry)
# rm(ObjAlpha, Obj_othR) #It's messay, dropping all the
# no-longer-necessary objects from the Global Environment

# Extracting the item-level statistics from the alpha object
Marg_othR <- as.data.frame(Marg_othR$r) #Makes the item-total(other) correlation matrix a df
# Adding variable names so we don't get lost
Marg_othR$Items <- c("Marg1", "Marg2", "Marg3", "Marg4", "Marg5", "Marg6",
"Marg7", "Objectified", "Strong", "Angry")
# deleting the rows with the total scale scores
Marg_othR <- Marg_othR[!Marg_othR$Items == "Objectified", ]
Marg_othR <- Marg_othR[!Marg_othR$Items == "Strong", ]
Marg_othR <- Marg_othR[!Marg_othR$Items == "Angry", ]
Marg_othR[, "Marginalized"] <- NA #We need a column for this to bind the items, later.
Marg_othR <- dplyr::select(Marg_othR, Items, Objectified, Marginalized,
                           Strong, Angry)
# Item Corrected Total Correlations
```

```

MargAlpha <- as.data.frame(MargAlpha$item.stats) #Grabbing the alpha objet we created earlier
MargAlpha$Items <- c("Marg1", "Marg2", "Marg3", "Marg4", "Marg5", "Marg6",
  "Marg7")
# Joining the two and selecting the vars of interest
MargStats <- full_join(MargAlpha, Marg_othR, by = "Items")
MargStats$Marginalized <- MargStats$r.drop #Copy the item-corrected total (r.drop) into the Marginalized column
MargStats <- dplyr::select(MargStats, Items, Objectified, Marginalized,
  Strong, Angry)
# rm(MargAlpha, Marg_othR) #It's messay, dropping all the no-longer-necessary objects from the Global Environment

Str_othR <- as.data.frame(Str_othR$r) #Makes the item-total(other) correlation matrix a df
# Adding variable names so we don't get lost
Str_othR$Items <- c("Strong1", "Strong2", "Strong3", "Strong4", "Strong5",
  "Objectified", "Marginalized", "Angry")
# deleting the rows with the total scale scores
Str_othR <- Str_othR[!Str_othR$Items == "Objectified", ]
Str_othR <- Str_othR[!Str_othR$Items == "Marginalized", ]
Str_othR <- Str_othR[!Str_othR$Items == "Angry", ]
Str_othR[, "Strong"] <- NA
Str_othR <- dplyr::select(Str_othR, Items, Objectified, Marginalized, Strong,
  Angry)
# Item Corrected Total Correlations
StrongAlpha <- as.data.frame(StrongAlpha$item.stats) #Grabbing the alpha objet we created earlier
StrongAlpha$Items <- c("Strong1", "Strong2", "Strong3", "Strong4", "Strong5")
# Joining the two and selecting the vars of interest
StrStats <- full_join(StrongAlpha, Str_othR, by = "Items")
StrStats$Strong <- StrStats$r.drop #Copy the item-corrected total (r.drop) into the Strong variable
StrStats <- dplyr::select(StrStats, Items, Objectified, Marginalized, Strong,
  Angry)
rm(StrongAlpha, Str_othR) #It's messay, dropping all the no-longer-necessary objects from the Global Environment

Ang_othR <- as.data.frame(Ang_othR$r) #Makes the item-total(other) correlation matrix a df
# Adding variable names so we don't get lost
Ang_othR$Items <- c("Angry1", "Angry2", "Angry3", "Objectified", "Marginalized",
  "Strong")
# deleting the rows with the total scale scores
Ang_othR <- Ang_othR[!Ang_othR$Items == "Objectified", ]
Ang_othR <- Ang_othR[!Ang_othR$Items == "Marginalized", ]
Ang_othR <- Ang_othR[!Ang_othR$Items == "Strong", ]
Ang_othR[, "Angry"] <- NA
Ang_othR <- dplyr::select(Ang_othR, Items, Objectified, Marginalized, Strong,
  Angry)
# Item Corrected Total Correlations
AngryAlpha <- as.data.frame(AngryAlpha$item.stats) #Grabbing the alpha objet we created earlier
AngryAlpha$Items <- c("Angry1", "Angry2", "Angry3")
# Joining the two and selecting the vars of interest

```

```

AngStats <- full_join(AngryAlpha, Ang_othR, by = "Items")
AngStats$Angry <- AngStats$r.drop #Copy the item-corrected total (r.drop) into the Angry variable
AngStats <- dplyr::select(AngStats, Items, Objectified, Marginalized, Strong,
                         Angry)
rm(AngryAlpha, Ang_othR) #It's messay, dropping all the no-longer-necessary objects from the environment

# Adding all the variables into a single table
ItemAnalysis <- rbind(ObjStats, MargStats, StrStats, AngStats)

# Preparing and adding the r.drop for total scale score
TotAlpha <- as.data.frame(GRMSalpha$item.stats)
TotAlpha$Items <- c("Obj1", "Obj2", "Obj3", "Obj4", "Obj5", "Obj6", "Obj7",
                    "Obj8", "Obj9", "Obj10", "Marg1", "Marg2", "Marg3", "Marg4", "Marg5",
                    "Marg6", "Marg7", "Strong1", "Strong2", "Strong3", "Strong4", "Strong5",
                    "Angry1", "Angry2", "Angry3")
TotAlpha <- dplyr::select(TotAlpha, Items, r.drop) #deleting the rows with the total scale scores

# Adding the r.drop for the total scale score
ItemAnalysis <- full_join(TotAlpha, ItemAnalysis, by = "Items")

# Adding the values from the orthogonal rotation I had to add
# 'unclass' to the loadings to render them into a df
pcaORTH_loadings <- data.frame(unclass(pcaORTH$loadings))
# Item names for joining (and to make sure we know which variable is
# which)
pcaORTH_loadings$Items <- c("Obj1", "Obj2", "Obj3", "Obj4", "Obj5", "Obj6",
                             "Obj7", "Obj8", "Obj9", "Obj10", "Marg1", "Marg2", "Marg3", "Marg4",
                             "Marg5", "Marg6", "Marg7", "Strong1", "Strong2", "Strong3", "Strong4",
                             "Strong5", "Angry1", "Angry2", "Angry3")

# Deleting those lower rows
pcaORTH_loadings <- rename(pcaORTH_loadings, objORTH = RC1, margORTH = RC2,
                           strORTH = RC3, angORTH2 = RC4)

# Joining with the Item Stats
Comparisons <- full_join(ItemAnalysis, pcaORTH_loadings, by = "Items") #I had to add 'unclass'

# Adding the oblique loadings
pcaOBLQ_loadings <- data.frame(unclass(pcaOBL$loadings)) #I had to add 'unclass' to the loadings
pcaOBLQ_loadings$Items <- c("Obj1", "Obj2", "Obj3", "Obj4", "Obj5", "Obj6",
                            "Obj7", "Obj8", "Obj9", "Obj10", "Marg1", "Marg2", "Marg3", "Marg4",
                            "Marg5", "Marg6", "Marg7", "Strong1", "Strong2", "Strong3", "Strong4",
                            "Strong5", "Angry1", "Angry2", "Angry3") #Item names for joining (and to make sure we know which)
# Deleting those lower rows
pcaOBLQ_loadings <-
# pcaOBLQ_loadings[!pcaORTH_loadings$Items == 'GRMSTot',]
# pcaOBLQ_loadings <- pcaOBLQ_loadings[!pcaORTH_loadings$Items ==

```

```

# 'Objectified',] pcaOBLQ_loadings <-
# pcaOBLQ_loadings[!pcaORTH_loadings$Items == 'Marginalized',]
# pcaOBLQ_loadings <- pcaOBLQ_loadings[!pcaORTH_loadings$Items ==
# 'Strong',] pcaOBLQ_loadings <-
# pcaOBLQ_loadings[!pcaORTH_loadings$Items == 'Angry',]
pcaOBLQ_loadings <- rename(pcaOBLQ_loadings, margOBLQ = TC1, objOBLQ = TC2,
    strOBLQ = TC3, angOBLQ = TC4)

# Joining with the Item Stats
Comparisons <- full_join(Comparisons, pcaOBLQ_loadings, by = "Items") #I had to add 'unclass'

write.csv(Comparisons, file = "GRMS_Comparisons.csv", sep = ",", row.names = FALSE,
    col.names = TRUE) #Writes the table to a .csv file where you can open it with Excel and f

Warning in write.csv(Comparisons, file = "GRMS_Comparisons.csv", sep = ",", :
attempt to set 'col.names' ignored

Warning in write.csv(Comparisons, file = "GRMS_Comparisons.csv", sep = ",", :
attempt to set 'sep' ignored

saveRDS(Comparisons, "GRMS_Comparisons.rds") #Writes the file as an .rds so that if anything

```

#### 8.7.1.4 Interpreting the result

The result of this work is a table that includes:

- **r.drop** Corrected item total (entire GRMS) coefficients
- **Item-total correlations** of the items correlated with their own subscale (bold; correlation does not include the item being correlated) and the other subscales
- **PCA: Orthogonal rotation** factor loadings of the four-scales with a rotation that maximizes the independents (uncorrelatedness) of the scales
- **PCA: Oblique rotation** factor loadings of the four-scales with a rotation that permits correlation between subscales

We expect to see similar results across the item-analysis, PCA orthogonal, and PCA oblique solutions. Our biggest interest is in whether items change scale membership and/or have cross-loadings. Overall, we are looking for items that *load* higher on their own scales than they do on other scales.

- When there are a number of cross-loadings, it means that the item will not discriminate well (think within-in scale discriminant validity).
- If there are a number of cross-loadings, there will likely be stronger correlations between subscales (indicating that an oblique rotation is/was an appropriate choice).
- Low/no cross-loadings, supports the choices of an orthogonal (uncorrelated) solution.
- Within-scale convergent validity is supported when an item has a strong, positive loading on its own scale and low/zero loadings on the other scales..

Items	r.drop	Item Analysis				Principal Components Analysis: Orthogonal Rotation				Principal Components Analysis: Oblique Rotation			
		Objectified	Marginalized	Strong	Angry	objORTH	margORTH	strORTH	angORTH2	objOBLQ	margOBLQ	strOBLQ	angOBLQ
Obj1	0.44	<b>0.47</b>	0.28	0.27	0.15	<b>0.57</b>	0.12	0.13	0.08	<b>0.57</b>	0.01	0.09	0.03
Obj2	0.47	<b>0.49</b>	0.35	0.24	0.16	<b>0.58</b>	0.18	0.10	0.09	<b>0.57</b>	0.08	0.05	0.04
Obj3	0.41	<b>0.43</b>	0.30	0.25	0.13	<b>0.53</b>	0.13	0.14	0.07	<b>0.53</b>	0.02	0.10	0.02
Obj4	0.44	<b>0.49</b>	0.33	0.18	0.14	<b>0.60</b>	0.20	0.00	0.01	<b>0.60</b>	0.11	-0.05	-0.04
Obj5	0.43	<b>0.44</b>	0.30	0.26	0.15	<b>0.53</b>	0.18	0.20	-0.10	<b>0.51</b>	0.09	0.17	-0.15
Obj6	0.34	<b>0.35</b>	0.22	0.17	0.18	<b>0.49</b>	-0.04	-0.02	0.43	<b>0.53</b>	-0.17	-0.05	0.40
Obj7	0.47	<b>0.43</b>	0.36	0.26	0.24	<b>0.46</b>	0.28	0.13	0.15	<b>0.42</b>	0.19	0.07	0.10
Obj8	0.38	<b>0.45</b>	0.31	0.14	0.05	<b>0.57</b>	0.27	0.00	-0.26	<b>0.55</b>	0.22	-0.05	-0.32
Obj9	0.34	<b>0.37</b>	0.22	0.20	0.11	<b>0.47</b>	0.05	0.21	-0.05	<b>0.46</b>	-0.04	0.19	-0.09
Obj10	0.35	<b>0.39</b>	0.22	0.17	0.14	<b>0.54</b>	0.02	0.00	0.17	<b>0.58</b>	-0.09	-0.04	0.13
Marg1	0.49	0.33	<b>0.60</b>	0.21	0.28	0.11	<b>0.78</b>	0.06	0.05	-0.03	<b>0.81</b>	-0.03	0.01
Marg2	0.50	0.32	<b>0.52</b>	0.32	0.33	0.09	<b>0.61</b>	0.28	0.18	-0.04	<b>0.60</b>	0.21	0.14
Marg3	0.42	0.30	<b>0.46</b>	0.22	0.23	0.13	<b>0.60</b>	0.16	0.02	0.02	<b>0.60</b>	0.08	-0.02
Marg4	0.41	0.33	<b>0.42</b>	0.15	0.24	0.24	<b>0.56</b>	-0.07	0.11	<b>0.16</b>	<b>0.55</b>	-0.16	0.07
Marg5	0.47	0.36	<b>0.50</b>	0.26	0.22	0.20	<b>0.60</b>	0.15	0.04	0.09	<b>0.59</b>	0.07	0.00
Marg6	0.46	0.40	<b>0.41</b>	0.22	0.21	0.37	<b>0.40</b>	0.03	0.18	<b>0.32</b>	<b>0.35</b>	-0.04	0.13
Marg7	0.33	0.25	<b>0.37</b>	0.13	0.19	0.10	<b>0.51</b>	0.00	0.12	0.02	<b>0.52</b>	-0.06	0.09
Strong1	0.37	0.28	0.26	<b>0.39</b>	0.16	0.16	0.19	<b>0.65</b>	-0.12	0.06	0.13	<b>0.64</b>	-0.15
Strong2	0.34	0.28	0.24	<b>0.28</b>	0.21	0.27	0.04	<b>0.38</b>	0.24	0.24	-0.07	<b>0.37</b>	0.21
Strong3	0.29	0.18	0.18	<b>0.38</b>	0.20	0.05	0.05	<b>0.66</b>	0.09	-0.02	-0.03	<b>0.67</b>	0.07
Strong4	0.24	0.18	0.14	<b>0.32</b>	0.10	0.14	0.02	<b>0.57</b>	-0.12	0.09	-0.04	<b>0.59</b>	-0.15
Strong5	0.29	0.21	0.20	<b>0.25</b>	0.23	0.10	0.06	<b>0.47</b>	0.25	0.05	-0.02	<b>0.46</b>	0.23
Angry1	0.31	0.16	0.25	<b>0.31</b>	<b>0.30</b>	-0.04	0.20	<b>0.44</b>	<b>0.37</b>	-0.12	0.14	<b>0.42</b>	<b>0.36</b>
Angry2	0.28	0.15	0.28	0.15	<b>0.31</b>	0.03	0.19	0.01	<b>0.73</b>	0.01	0.13	-0.04	0.73
Angry3	0.32	0.23	0.28	0.18	<b>0.31</b>	0.09	0.25	0.12	<b>0.55</b>	0.05	0.18	0.07	<b>0.54</b>

Figure 8.7: Image of a table of values from the item analysis and PCA solutions with orthogonal and oblique rotations

Our simulation from the Lewis and Neville's [2015] GRMS produced slightly different results from their original data. Specifically, our "Angry1" item cross-loaded on both the Strong Angry subscales with slightly stronger loadings on the Strong (incorrect) subscale. The items behaved much better in the original article.

## 8.8 Practice Problems

In each of these lessons I provide suggestions for practice that allow you to select one or more problems that are graded in difficulty. In psychometrics, I strongly recommend that you have started with a dataset that has a minimum of three subscales and use it for all of the assignments in the OER. In any case, please plan to:

- Properly format and prepare the data.
- Conduct diagnostic tests to determine the suitability of the data for PCA.
- Conduct tests to guide the decisions about number of components to extract.
- Conduct orthogonal and oblique rotations (at least two each with different numbers of extracted components).
- Select one solution and preparing an APA style results section (with table and figure).

### 8.8.1 Problem #1: Play around with this simulation.

Copy the script for the simulation and then change (at least) one thing in the simulation to see how it impacts the results. If PCA is new to you, perhaps you just change the number in "set.seed(240311)" from 240311 to something else. Your results should *parallel* those obtained in the lecture, making

it easier for you to check your work as you go. Don't be surprised if the factor loadings wiggle around a little. Do try to make sense of them.

### 8.8.2 Problem #2: Conduct a PCA with another simulated set of data in the OER.

The second option involves utilizing one of the simulated datasets available in this OER. The [last lesson](#) in the OER contains three simulations that could be used for all of the statistics-based practice suggestions. Especially if you started with one of these examples in an earlier lesson, I highly recommend you continue with that.

Alternatively, Keum et al.'s Gendered Racial Microaggressions Scale for Asian American Women [Keum et al., 2018] will be used in the lessons on confirmatory factor analysis and Conover et al.'s [Conover et al., 2017] Ableist Microaggressions Scale is used in the lesson on invariance testing. Both of these would be suitable for the PCA and PAF homework assignments.

### 8.8.3 Problem #3: Try something entirely new.

Using data for which you have permission and access (e.g., IRB approved data you have collected or from your lab; data you simulate from a published article; data from the ReCentering Psych Stats survey described in the [Qualtrics lesson](#), or data from an open science repository), complete a PCA analysis. The data should allow for at least three factors/subscales.

### 8.8.4 Grading Rubric

Using the lecture and workflow (chart) as a guide, please work through all the steps listed in the proposed assignment/grading rubric.

Assignment Component	Points Possible	Points Earned
1. Check and, if needed, format data	5	_____
2. Conduct and interpret the three diagnostic tests to determine if PCA is appropriate as an analysis (KMO, Bartlett's, determinant).	5	_____
3. Determine how many components to extract (e.g., scree plot, eigenvalues, theory).	5	_____
4. Conduct an orthogonal rotation with a minimum of two different numbers of component extractions.	5	_____
5. Conduct an oblique rotation with a minimum of two different numbers of component extractions.	5	_____
6. Determine which factor solution (e.g., orthogonal or oblique; with which number of components) you will suggest.	5	_____
7. APA style results section with table and figure of one of the solutions.	5	_____
8. Explanation to grader	5	_____

Assignment Component	Points Possible	Points Earned
<b>Totals</b>	40	_____

## 8.9 Homeworked Example

[Screencast Link](#)

For more information about the data used in this homeworked example, please refer to the description and codebook located at the end of the [introduction](#) in first volume of ReCentering Psych Stats.

As a brief review, this data is part of an IRB-approved study, with consent to use in teaching demonstrations and to be made available to the general public via the open science framework. Hence, it is appropriate to use in this context. You will notice there are student- and teacher- IDs. These numbers are not actual student and teacher IDs, rather they were further re-identified so that they could not be connected to actual people.

Because this is an actual dataset, if you wish to work the problem along with me, you will need to download the [ReC.rds](#) data file from the Worked\_Examples folder in the ReC\_Psychometrics project on the GitHub.

The course evaluation items can be divided into three subscales:

- **Valued by the student** includes the items: ValObjectives, IncrUnderstanding, IncrInterest
- **Traditional pedagogy** includes the items: ClearResponsibilities, EffectiveAnswers, Feedback, ClearOrganization, ClearPresentation
- **Socially responsive pedagogy** includes the items: InclusvClassrm, EquitableEval, MultPerspectives, DEIIntegration

In this homewoRked example I will conduct a principal components analysis. My hope is that the results will support my solution of three dimensions: valued-by-the-student, traditional pedagogy, socially responsive pedagogy.

### 8.9.1 Check and, if needed, format data

```
big <- readRDS("ReC.rds")
```

With the next code I will create an item-level df with only the items used in the three scales.

```
library(tidyverse)
items <- big %>%
  dplyr::select(ValObjectives, IncrUnderstanding, IncrInterest, ClearResponsibilities,
               EffectiveAnswers, Feedback, ClearOrganization, ClearPresentation,
               MultPerspectives, InclusvClassrm, DEIIntegration, EquitableEval)
```

Some of the analyses require non-missing data in the df.

```
items <- na.omit(items)
```

Let's check the structure of the data.

```
str(items)
```

```
Classes 'data.table' and 'data.frame': 267 obs. of 12 variables:
 $ ValObjectives      : int 5 5 4 4 5 5 5 4 5 3 ...
 $ IncrUnderstanding   : int 2 3 4 3 4 5 2 4 5 4 ...
 $ IncrInterest        : int 5 3 4 2 4 5 3 2 5 1 ...
 $ ClearResponsibilities: int 5 5 4 4 5 5 4 4 5 3 ...
 $ EffectiveAnswers    : int 5 3 5 3 5 4 3 2 3 3 ...
 $ Feedback            : int 5 3 4 2 5 5 4 4 5 2 ...
 $ ClearOrganization    : int 3 4 3 4 4 5 4 4 5 2 ...
 $ ClearPresentation    : int 4 4 4 2 5 4 4 4 5 2 ...
 $ MultPerspectives     : int 5 5 4 5 5 5 5 5 5 1 ...
 $ InclusvClassrm       : int 5 5 5 5 5 5 5 4 5 3 ...
 $ DEIintegration       : int 5 5 5 5 5 5 5 5 5 2 ...
 $ EquitableEval        : int 5 5 3 5 5 5 5 3 5 3 ...
 - attr(*, ".internal.selfref")=<externalptr>
 - attr(*, "na.action")= 'omit' Named int [1:43] 6 20 106 109 112 113 114 117 122 128 ...
 ..- attr(*, "names")= chr [1:43] "6" "20" "106" "109" ...
```

## 8.9.2 Conduct and interpret the three diagnostic tests to determine if PCA is appropriate as an analysis (KMO, Bartlett's, determinant)

### 8.9.2.1 KMO

The Kaiser-Meyer-Olkin (KMO) index is an index of *sampling adequacy* to let us know if the sample size is sufficient relative to the statistical characteristics of the data.

General criteria (1974, Kaiser):

- bare minimum of .5
- values between .5 and .7 as mediocre
- values between .7 and .8 are good
- values above .9 are superb

```
psych::KMO(items)
```

```
Kaiser-Meyer-Olkin factor adequacy
Call: psych::KMO(r = items)
Overall MSA =  0.91
MSA for each item =
```

	ValObjectives	IncrUnderstanding	IncrInterest
	0.94	0.89	0.89
	ClearResponsibilities	EffectiveAnswers	Feedback
	0.91	0.93	0.94
	ClearOrganization	ClearPresentation	MultPerspectives
	0.94	0.91	0.93
	InclusvClassrm	DEIIntegration	EquitableEval
	0.86	0.78	0.95

With a KMO of 0.91, the data seems appropriate to continue with the PCA.

### 8.9.2.2 Bartlett's

Barlett's test lets us know if the matrix is an *identity matrix* (i.e., where elements on the off-diagonal would be 0.0 and elements on the diagonal would be 1.0). Stated another way – items only correlate with “themselves” and not other variables.

When  $p < 0.05$  the matrix is not an identity matrix. That is, there are some relationships between variables that can be analyzed.

```
psych::cortest.bartlett(items)
```

```
R was not square, finding R from data
```

```
$chisq
[1] 1897.769
```

```
$p.value
[1] 0
```

```
$df
[1] 66
```

The Barlett's test,  $\chi^2(66) = 1897.77, p < 0.001$ , indicating that the correlation matrix is not an identity matrix and, on that dimension, is suitable for analysis.

### 8.9.2.3 Determinant

Multicollinearity or singularity is diagnosed by the determinant. The determinant should be greater than 0.00001. If smaller, then there may be an issue with multicollinearity (variables that are too highly correlated) or singularity (variables that are perfectly correlated).

```
items <- na.omit(items)
det(cor(items))
```

```
[1] 0.0006985496
```

The value of the determinant is 0.0007; greater than 0.00001. We are not concerned with multicollinearity or singularity.

Summary from data screening:

Data screening were conducted to determine the suitability of the data for this analyses. The Kaiser-Meyer-Olkin measure of sampling adequacy (KMO; Kaiser, 1970) represents the ratio of the squared correlation between variables to the squared partial correlation between variables. KMO ranges from 0.00 to 1.00; values closer to 1.00 indicate that the patterns of correlations are relatively compact, and that component analysis should yield distinct and reliable components (Field, 2012). In our dataset, the KMO value was 0.91, indicating acceptable sampling adequacy. The Bartlett's Test of Sphericity examines whether the population correlation matrix resembles an identity matrix (Field, 2012). When the  $p$  value for the Bartlett's test is  $< .05$ , we are fairly certain we have clusters of correlated variables. In our dataset,  $\chi^2(66) = 1897.77, p < 0.001$  indicating the correlations between items are sufficiently large enough for principal components analysis. The determinant of the correlation matrix alerts us to any issues of multicollinearity or singularity and should be larger than 0.00001. Our determinant was 0.0007 and, again, indicated that our data was suitable for the analysis.

### 8.9.3 Determine how many components to extract (e.g., scree plot, eigenvalues, theory)

Step #1: creating a principal components model with the same number of components as items

```
pca1 <- psych::principal(items, nfactors = length(items), rotate = "none") # using raw data at
```

```
Principal Components Analysis
Call: psych::principal(r = items, nfactors = length(items), rotate = "none")
Standardized loadings (pattern matrix) based upon correlation matrix
          PC1    PC2    PC3    PC4    PC5    PC6    PC7    PC8    PC9
ValObjectives  0.57 -0.13  0.42  0.68 -0.08  0.03  0.03 -0.06 -0.04
IncrUnderstanding  0.68 -0.37  0.39 -0.28  0.08  0.09 -0.12  0.25 -0.04
IncrInterest   0.73 -0.19  0.41 -0.17  0.32 -0.05  0.06 -0.12  0.03
ClearResponsibilities  0.81 -0.08 -0.37  0.04 -0.20  0.02 -0.03  0.09 -0.10
EffectiveAnswers  0.80 -0.16 -0.20 -0.09 -0.03  0.05  0.48 -0.05  0.07
Feedback        0.77  0.06 -0.28  0.11  0.32 -0.29  0.01  0.09 -0.31
ClearOrganization  0.79 -0.27 -0.12  0.04 -0.22 -0.22 -0.20  0.19  0.20
ClearPresentation  0.85 -0.21  0.00 -0.11 -0.24  0.01  0.05 -0.18  0.04
MultPerspectives  0.79  0.26 -0.11 -0.05  0.13 -0.14 -0.23 -0.35  0.17
InclusvClassrm   0.66  0.50  0.24 -0.20 -0.30  0.11 -0.06 -0.07 -0.28
DEIIntegration   0.50  0.75  0.19  0.03  0.02 -0.12  0.14  0.25  0.21
EquitableEval     0.73  0.13 -0.27  0.13  0.22  0.52 -0.12  0.07  0.07
                           PC10   PC11   PC12   h2                      u2 com
ValObjectives      0.05  0.02  0.04  1 -0.00000000000000022 2.9
IncrUnderstanding  0.27  0.00  0.07  1  0.000000000000000389 3.7
```

IncrInterest	-0.28	0.15	-0.09	1	0.000000000000000111	3.0
ClearResponsibilities	0.05	0.36	-0.11	1	0.000000000000000089	2.2
EffectiveAnswers	0.03	-0.01	0.19	1	0.000000000000000078	2.1
Feedback	0.02	-0.14	-0.02	1	0.000000000000000089	2.6
ClearOrganization	-0.23	-0.09	0.11	1	0.000000000000000033	2.5
ClearPresentation	0.08	-0.21	-0.28	1	0.000000000000000122	1.9
MultPerspectives	0.18	0.06	0.12	1	0.000000000000000100	2.4
InclusvClassrm	-0.13	-0.03	0.12	1	0.000000000000000056	3.7
DEIIntegration	0.06	0.01	-0.11	1	0.000000000000000033	2.6
EquitableEval	-0.08	-0.09	-0.02	1	0.000000000000000011	2.7

	PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8	PC9	PC10	PC11
SS loadings	6.38	1.23	0.95	0.67	0.52	0.47	0.39	0.37	0.32	0.27	0.24
Proportion Var	0.53	0.10	0.08	0.06	0.04	0.04	0.03	0.03	0.03	0.02	0.02
Cumulative Var	0.53	0.63	0.71	0.77	0.81	0.85	0.88	0.91	0.94	0.96	0.98
Proportion Explained	0.53	0.10	0.08	0.06	0.04	0.04	0.03	0.03	0.03	0.02	0.02
Cumulative Proportion	0.53	0.63	0.71	0.77	0.81	0.85	0.88	0.91	0.94	0.96	0.98
	PC12										
SS loadings	0.20										
Proportion Var	0.02										
Cumulative Var	1.00										
Proportion Explained	0.02										
Cumulative Proportion	1.00										

Mean item complexity = 2.7

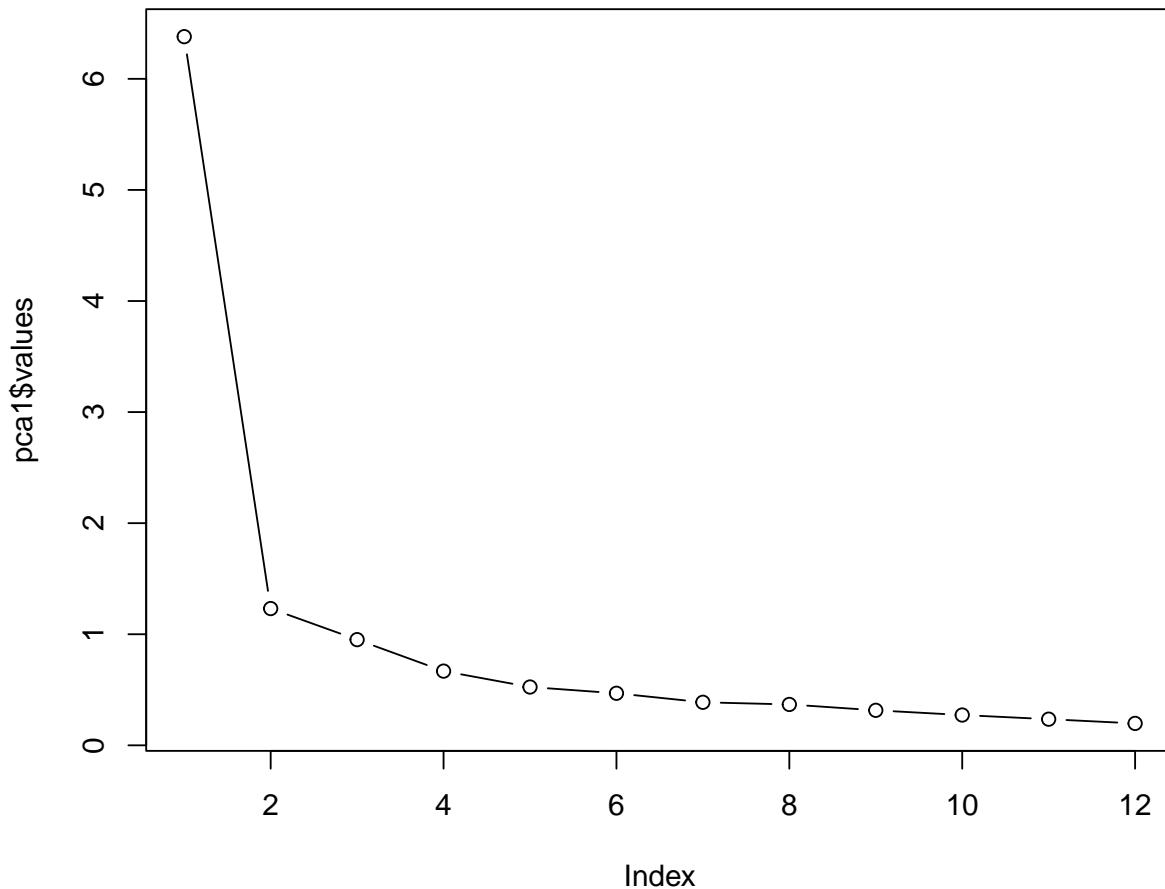
Test of the hypothesis that 12 components are sufficient.

The root mean square of the residuals (RMSR) is 0  
with the empirical chi square 0 with prob < NA

Fit based upon off diagonal values = 1

The eigenvalue-greater-than-one criteria suggests 2 factors (but the third component has an SS-loading of .95 – it's close to three).

```
plot(pca1$values, type = "b")
```



The scree plot looks like one factor.

Ugh.

- I want 3 factors (we could think of this as a priori theory); would account for 71% of variance.
- Eigenvalues-greater-than-one criteria suggests two; could account for 63% of variance.
- Scree plot suggests 1 (would account for 53% of variance)

*Note:* The lecture has more on evaluating communalities and uniquenesses and how this information can also inform the number of components we want to extract. Because it is easy to get lost (very lost) I will skip over this for now. If you were to create a measure and use PCA as an exploratory approach to understanding the dimensionality of an instrument, you would likely want to investigate further and report on these.

#### 8.9.4 Conduct an orthogonal extraction and rotation with a minimum of two different factor extractions

An orthogonal two factor solution

```
pcaORTH2f <- psych::principal(items, nfactors = 2, rotate = "varimax")
pcaORTH2f
```

Principal Components Analysis  
 Call: psych::principal(r = items, nfactors = 2, rotate = "varimax")  
 Standardized loadings (pattern matrix) based upon correlation matrix

	RC1	RC2	h2	u2	com
ValObjectives	0.55	0.19	0.34	0.66	1.2
IncrUnderstanding	0.77	0.05	0.59	0.41	1.0
IncrInterest	0.72	0.23	0.57	0.43	1.2
ClearResponsibilities	0.73	0.36	0.66	0.34	1.5
EffectiveAnswers	0.76	0.29	0.67	0.33	1.3
Feedback	0.62	0.46	0.59	0.41	1.8
ClearOrganization	0.81	0.18	0.69	0.31	1.1
ClearPresentation	0.83	0.27	0.76	0.24	1.2
MultPerspectives	0.53	0.64	0.70	0.30	1.9
InclusvClassrm	0.29	0.77	0.68	0.32	1.3
DEIIntegration	0.03	0.90	0.80	0.20	1.0
EquitableEval	0.55	0.50	0.55	0.45	2.0

	RC1	RC2
SS loadings	4.93	2.68
Proportion Var	0.41	0.22
Cumulative Var	0.41	0.63
Proportion Explained	0.65	0.35
Cumulative Proportion	0.65	1.00

Mean item complexity = 1.4  
 Test of the hypothesis that 2 components are sufficient.

The root mean square of the residuals (RMSR) is 0.07  
 with the empirical chi square 170.34 with prob < 0.0000000000000045

Fit based upon off diagonal values = 0.98

Sorting the scores into a table can help see the results more clearly. The “cut = #” command will not show the factor scores for factor loading < .30. I would do this “to see”, but I would include all the values in an APA style table.

```
pca_tableORT2f <- psych::print.psych(pcaORTH2f, cut = 0.3, sort = TRUE)
```

Principal Components Analysis  
 Call: psych::principal(r = items, nfactors = 2, rotate = "varimax")  
 Standardized loadings (pattern matrix) based upon correlation matrix

	item	RC1	RC2	h2	u2	com
ClearPresentation	8	0.83		0.76	0.24	1.2

ClearOrganization	7	0.81	0.69	0.31	1.1
IncrUnderstanding	2	0.77	0.59	0.41	1.0
EffectiveAnswers	5	0.76	0.67	0.33	1.3
ClearResponsibilities	4	0.73	0.36	0.66	0.34
IncrInterest	3	0.72	0.57	0.43	1.2
Feedback	6	0.62	0.46	0.59	0.41
ValObjectives	1	0.55	0.34	0.66	1.2
EquitableEval	12	0.55	0.50	0.55	0.45
DEIIntegration	11		0.90	0.80	0.20
InclusvClassrm	10		0.77	0.68	0.32
MultPerspectives	9	0.53	0.64	0.70	0.30

	RC1	RC2
SS loadings	4.93	2.68
Proportion Var	0.41	0.22
Cumulative Var	0.41	0.63
Proportion Explained	0.65	0.35
Cumulative Proportion	0.65	1.00

Mean item complexity = 1.4

Test of the hypothesis that 2 components are sufficient.

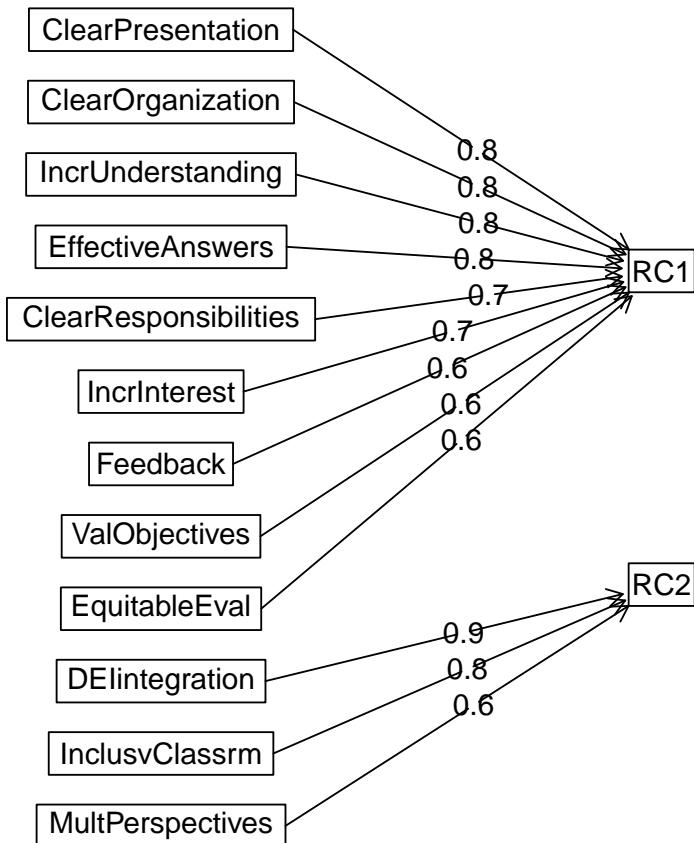
The root mean square of the residuals (RMSR) is 0.07  
 with the empirical chi square 170.34 with prob < 0.00000000000000045

Fit based upon off diagonal values = 0.98

F1: Includes everything else. F2: Includes the SCR items (although MultPerspectives cross-loads onto F1; Similarly, EquitableEval is on F1)

```
psych::fa.diagram(pcaORTH2f)
```

## Components Analysis



Plotting these figures from the program can facilitate conceptual understanding of what is going on – and can be a “check” to your work.

In the lecture I made a “biggish deal” about PCA being *components* (not *factor*) analysis. Although the two approaches can lead to similar results/conclusions, there are some significant differences “under the hood.” PCA can be thought of more as regression where the items predict the component. Consequently, the arrows go *from* the item *to* the component. Starting with the next lesson, the arrows will go from the factor to the item – because the factors (or latent variables) are assumed to predict the scores on the items (i.e., “depression” would predict how someone rates items that assess hopelessness, sleep, anhedonia, and so forth).

### An orthogonal three factor solution

```
pcaORTH3f <- psych::principal(items, nfactors = 3, rotate = "varimax")
pcaORTH3f
```

```
Principal Components Analysis
Call: psych::principal(r = items, nfactors = 3, rotate = "varimax")
Standardized loadings (pattern matrix) based upon correlation matrix
```

	RC1	RC3	RC2	h2	u2	com
ValObjectives	0.16	0.67	0.21	0.52	0.48	1.3
IncrUnderstanding	0.29	0.81	0.04	0.75	0.25	1.3
IncrInterest	0.30	0.78	0.23	0.74	0.26	1.5
ClearResponsibilities	0.85	0.22	0.14	0.80	0.20	1.2
EffectiveAnswers	0.75	0.37	0.12	0.71	0.29	1.5
Feedback	0.75	0.19	0.28	0.67	0.33	1.4
ClearOrganization	0.70	0.47	0.03	0.71	0.29	1.8
ClearPresentation	0.65	0.56	0.14	0.76	0.24	2.1
MultPerspectives	0.63	0.24	0.51	0.71	0.29	2.2
InclusvClassrm	0.26	0.30	0.76	0.74	0.26	1.6
DEIIntegration	0.15	0.07	0.90	0.84	0.16	1.1
EquitableEval	0.70	0.15	0.33	0.62	0.38	1.5

	RC1	RC3	RC2
SS loadings	3.93	2.64	1.99
Proportion Var	0.33	0.22	0.17
Cumulative Var	0.33	0.55	0.71
Proportion Explained	0.46	0.31	0.23
Cumulative Proportion	0.46	0.77	1.00

Mean item complexity = 1.5

Test of the hypothesis that 3 components are sufficient.

The root mean square of the residuals (RMSR) is 0.06  
with the empirical chi square 115.69 with prob < 0.000000000041

Fit based upon off diagonal values = 0.99

```
pca_tableORT3f <- psych::print.psych(pcaORTH3f, cut = 0.3, sort = TRUE)
```

```
Principal Components Analysis
Call: psych::principal(r = items, nfactors = 3, rotate = "varimax")
Standardized loadings (pattern matrix) based upon correlation matrix
          item  RC1   RC3   RC2   h2   u2 com
ClearResponsibilities  4  0.85           0.80 0.20 1.2
EffectiveAnswers       5  0.75  0.37           0.71 0.29 1.5
Feedback                6  0.75           0.67 0.33 1.4
EquitableEval          12 0.70           0.33 0.62 0.38 1.5
ClearOrganization       7  0.70  0.47           0.71 0.29 1.8
ClearPresentation        8  0.65  0.56           0.76 0.24 2.1
MultPerspectives        9  0.63           0.51 0.71 0.29 2.2
IncrUnderstanding        2    0.81           0.75 0.25 1.3
IncrInterest             3    0.78           0.74 0.26 1.5
ValObjectives            1    0.67           0.52 0.48 1.3
DEIIntegration           11           0.90 0.84 0.16 1.1
InclusvClassrm          10           0.30 0.76 0.74 0.26 1.6
```

	RC1	RC3	RC2
SS loadings	3.93	2.64	1.99
Proportion Var	0.33	0.22	0.17
Cumulative Var	0.33	0.55	0.71
Proportion Explained	0.46	0.31	0.23
Cumulative Proportion	0.46	0.77	1.00

Mean item complexity = 1.5

Test of the hypothesis that 3 components are sufficient.

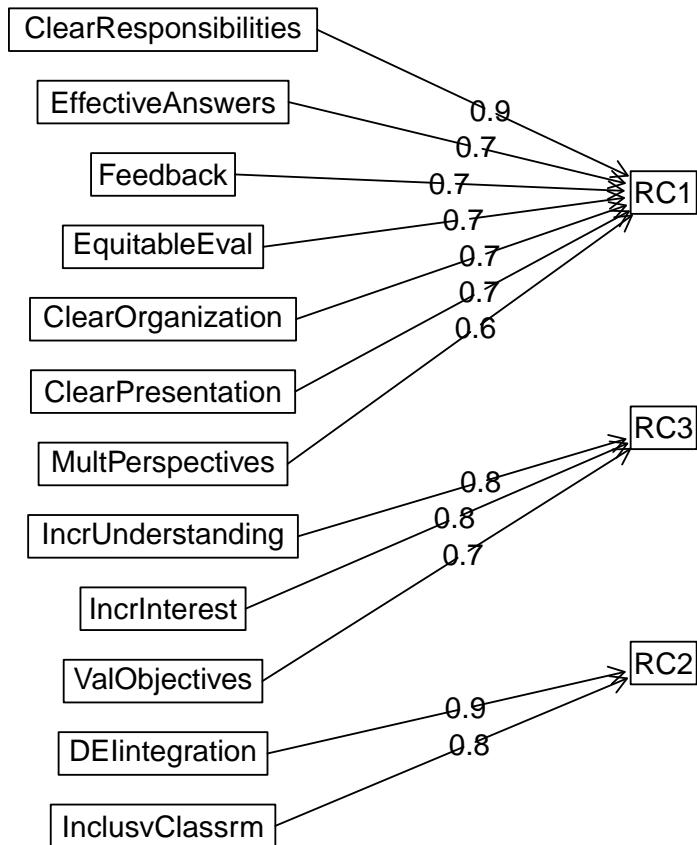
The root mean square of the residuals (RMSR) is 0.06  
with the empirical chi square 115.69 with prob < 0.000000000041

Fit based upon off diagonal values = 0.99

F1: Traditional Pedagogy F2: Valued-by-Me F3: SCRPed—except Equitable Eval \* MultPerspectives are on TradPed; MultPerspectives cross-load

```
psych::fa.diagram(pcaORTH3f)
```

## Components Analysis



The three-factor solution gets really close to my goals of (a) traditional pedagogy, (b) valued by the student, and (c) socially responsive pedagogy. The trouble is that I would prefer “multiple perspectives” to load with the socially responsive pedagogy factor.

### 8.9.5 Conduct an oblique extraction and rotation with a minimum of two different factor extractions

#### An oblique two factor solution

```
pca0BL2f <- psych::principal(items, nfactors = 2, rotate = "oblimin")
pca0BL2f
```

```
Principal Components Analysis
Call: psych::principal(r = items, nfactors = 2, rotate = "oblimin")
Standardized loadings (pattern matrix) based upon correlation matrix
          TC1   TC2   h2   u2 com
ValObjectives    0.58  0.01  0.34  0.66  1.0
```

IncrUnderstanding	0.84	-0.21	0.59	0.41	1.1
IncrInterest	0.76	0.00	0.57	0.43	1.0
ClearResponsibilities	0.75	0.13	0.66	0.34	1.1
EffectiveAnswers	0.80	0.05	0.67	0.33	1.0
Feedback	0.61	0.27	0.59	0.41	1.4
ClearOrganization	0.86	-0.08	0.69	0.31	1.0
ClearPresentation	0.87	0.00	0.76	0.24	1.0
MultPerspectives	0.50	0.49	0.70	0.30	2.0
InclusvClassrm	0.21	0.71	0.68	0.32	1.2
DEIIntegration	-0.10	0.93	0.80	0.20	1.0
EquitableEval	0.53	0.34	0.55	0.45	1.7

	TC1	TC2
SS loadings	5.50	2.11
Proportion Var	0.46	0.18
Cumulative Var	0.46	0.63
Proportion Explained	0.72	0.28
Cumulative Proportion	0.72	1.00

With component correlations of

TC1	TC2
TC1	1.00 0.43
TC2	0.43 1.00

Mean item complexity = 1.2

Test of the hypothesis that 2 components are sufficient.

The root mean square of the residuals (RMSR) is 0.07  
with the empirical chi square 170.34 with prob < 0.0000000000000045

Fit based upon off diagonal values = 0.98

```
pca_table0BL2f <- psych::print.psych(pca0BL2f, cut = 0.3, sort = TRUE)
```

#### Principal Components Analysis

Call: psych::principal(r = items, nfactors = 2, rotate = "oblimin")  
Standardized loadings (pattern matrix) based upon correlation matrix

	item	TC1	TC2	h2	u2	com
ClearPresentation	8	0.87		0.76	0.24	1.0
ClearOrganization	7	0.86		0.69	0.31	1.0
IncrUnderstanding	2	0.84		0.59	0.41	1.1
EffectiveAnswers	5	0.80		0.67	0.33	1.0
IncrInterest	3	0.76		0.57	0.43	1.0
ClearResponsibilities	4	0.75		0.66	0.34	1.1
Feedback	6	0.61		0.59	0.41	1.4
ValObjectives	1	0.58		0.34	0.66	1.0
EquitableEval	12	0.53	0.34	0.55	0.45	1.7

MultPerspectives	9	0.50	0.49	0.70	0.30	2.0
DEIintegration	11		0.93	0.80	0.20	1.0
InclusvClassrm	10		0.71	0.68	0.32	1.2

	TC1	TC2
SS loadings	5.50	2.11
Proportion Var	0.46	0.18
Cumulative Var	0.46	0.63
Proportion Explained	0.72	0.28
Cumulative Proportion	0.72	1.00

With component correlations of

TC1	TC2
TC1	1.00 0.43
TC2	0.43 1.00

Mean item complexity = 1.2

Test of the hypothesis that 2 components are sufficient.

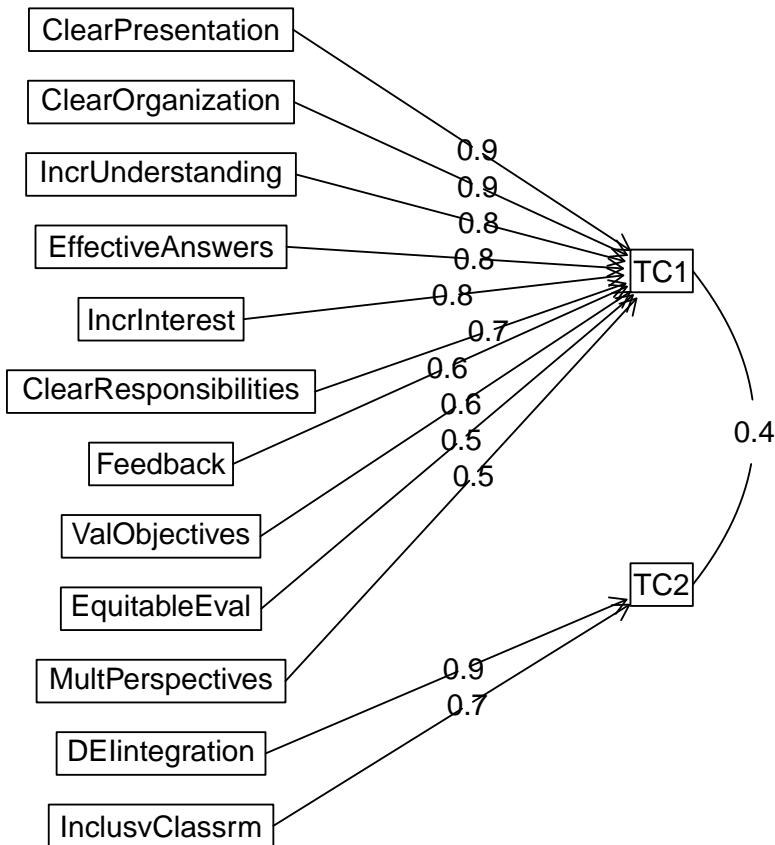
The root mean square of the residuals (RMSR) is 0.07  
 with the empirical chi square 170.34 with prob < 0.00000000000000045

Fit based upon off diagonal values = 0.98

Fairly similar results to the orthogonal variation of this – with EquitableEval and MultPerspectives cross-loading, with stronger loadings on the TradPed/Valued dimension.

```
psych::fa.diagram(pca0BL2f)
```

## Components Analysis



The curved line and value between TC1 and TC2 illustrates that in the oblique solution the components are allowed to correlate. There was no such path on the orthogonal figures. This is because the rotation required the components to be uncorrelated.

### An oblique three factor solution

```
pca0BL3f <- psych::principal(items, nfactors = 3, rotate = "oblimin")
pca0BL3f
```

```
Principal Components Analysis
Call: psych::principal(r = items, nfactors = 3, rotate = "oblimin")
Standardized loadings (pattern matrix) based upon correlation matrix
          TC1    TC3    TC2    h2   u2 com
ValObjectives -0.08  0.71  0.15  0.52  0.48 1.1
IncrUnderstanding  0.06  0.84 -0.07  0.75  0.25 1.0
IncrInterest     0.05  0.79  0.13  0.74  0.26 1.1
ClearResponsibilities  0.95 -0.06 -0.05  0.80  0.20 1.0
EffectiveAnswers   0.76  0.15 -0.06  0.71  0.29 1.1
```

Feedback	0.80	-0.06	0.12	0.67	0.33	1.1
ClearOrganization	0.68	0.29	-0.15	0.71	0.29	1.5
ClearPresentation	0.57	0.41	-0.02	0.76	0.24	1.8
MultPerspectives	0.59	0.03	0.39	0.71	0.29	1.7
InclusvClassrm	0.08	0.22	0.73	0.74	0.26	1.2
DEIIntegration	0.00	-0.02	0.92	0.84	0.16	1.0
EquitableEval	0.75	-0.10	0.18	0.62	0.38	1.2

	TC1	TC3	TC2
SS loadings	4.23	2.50	1.83
Proportion Var	0.35	0.21	0.15
Cumulative Var	0.35	0.56	0.71
Proportion Explained	0.49	0.29	0.21
Cumulative Proportion	0.49	0.79	1.00

With component correlations of

	TC1	TC3	TC2
TC1	1.00	0.58	0.39
TC3	0.58	1.00	0.25
TC2	0.39	0.25	1.00

Mean item complexity = 1.2

Test of the hypothesis that 3 components are sufficient.

The root mean square of the residuals (RMSR) is 0.06  
with the empirical chi square 115.69 with prob < 0.000000000041

Fit based upon off diagonal values = 0.99

```
pca_table0BL3f <- psych::print.psych(pca0BL3f, cut = 0.3, sort = TRUE)
```

#### Principal Components Analysis

Call: psych::principal(r = items, nfactors = 3, rotate = "oblimin")  
Standardized loadings (pattern matrix) based upon correlation matrix

	item	TC1	TC3	TC2	h2	u2	com
ClearResponsibilities	4	0.95			0.80	0.20	1.0
Feedback	6	0.80			0.67	0.33	1.1
EffectiveAnswers	5	0.76			0.71	0.29	1.1
EquitableEval	12	0.75			0.62	0.38	1.2
ClearOrganization	7	0.68			0.71	0.29	1.5
MultPerspectives	9	0.59		0.39	0.71	0.29	1.7
ClearPresentation	8	0.57	0.41		0.76	0.24	1.8
IncrUnderstanding	2		0.84		0.75	0.25	1.0
IncrInterest	3		0.79		0.74	0.26	1.1
ValObjectives	1		0.71		0.52	0.48	1.1
DEIIntegration	11			0.92	0.84	0.16	1.0
InclusvClassrm	10			0.73	0.74	0.26	1.2

	TC1	TC3	TC2
SS loadings	4.23	2.50	1.83
Proportion Var	0.35	0.21	0.15
Cumulative Var	0.35	0.56	0.71
Proportion Explained	0.49	0.29	0.21
Cumulative Proportion	0.49	0.79	1.00

With component correlations of

	TC1	TC3	TC2
TC1	1.00	0.58	0.39
TC3	0.58	1.00	0.25
TC2	0.39	0.25	1.00

Mean item complexity = 1.2

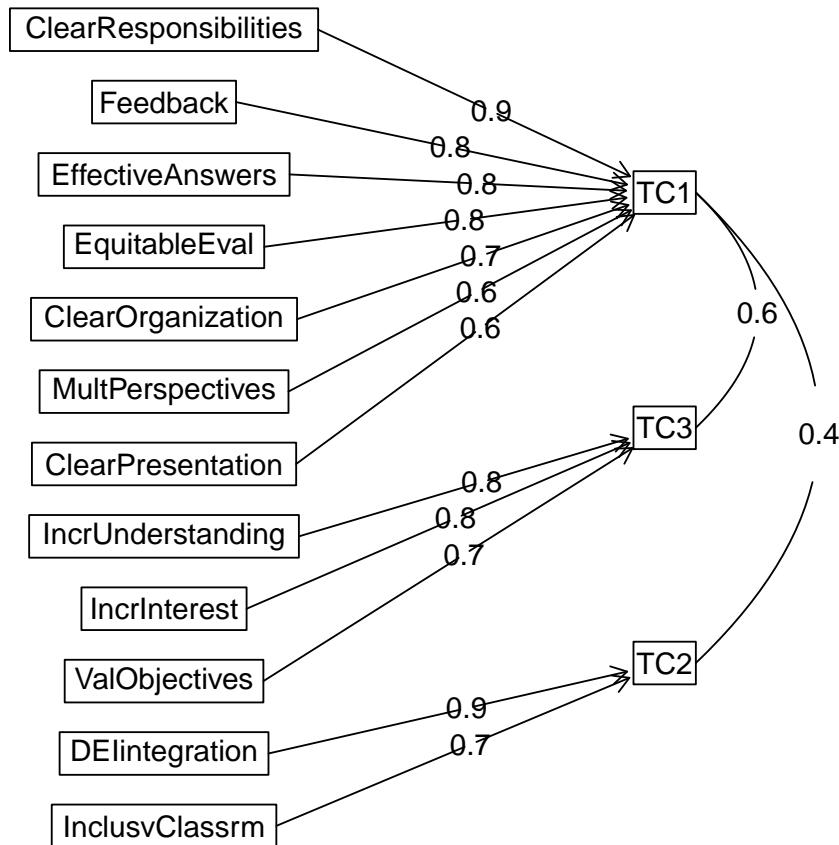
Test of the hypothesis that 3 components are sufficient.

The root mean square of the residuals (RMSR) is 0.06  
with the empirical chi square 115.69 with prob < 0.000000000041

Fit based upon off diagonal values = 0.99

```
psych::fa.diagram(pca0BL3f)
```

## Components Analysis



The results are quite similar to the orthogonal solution.

### 8.9.6 Determine which factor solution (e.g., orthogonal or oblique; which number of factors) you will suggest

From the oblique output we see that the correlations between the three subscales range from 0.25 to 0.58. These are high. Therefore, I will choose a 3-component, oblique, solution.

### 8.9.7 APA style results section with table and figure of one of the solutions

The dimensionality of the 12 course evaluation items was analyzed using principal components analysis. First, data were screened to determine the suitability of the data for this principal components analysis. The Kaiser-Meyer-Olkin measure of sampling adequacy (KMO; Kaiser, 1970) represents the ratio of the squared correlation between variables to the squared partial correlation between variables. KMO ranges from 0.00 to 1.00; values closer to 1.00 indicate that the patterns of correlations are relatively compact, and that component analysis should yield distinct and reliable components

(Field, 2012). In our dataset, the KMO value was 0.91, indicating acceptable sampling adequacy. The Barlett's Test of Sphericity examines whether the population correlation matrix resembles an identity matrix (Field, 2012). When the  $p$  value for the Bartlett's test is  $< .05$ , we are fairly certain we have clusters of correlated variables. In our dataset,  $\chi^2(66) = 1897.77, p < 0.001$  indicating the correlations between items are sufficiently large enough for principal components analysis. The determinant of the correlation matrix alerts us to any issues of multicollinearity or singularity and should be larger than 0.00001. Our determinant was 0.0007 and, again, indicated that our data was suitable for the analysis.

Four criteria were used to determine the number of components to extract: a priori theory, the scree test, the eigenvalue-greater-than-one criteria, and the interpretability of the solution. Kaiser's eigenvalue-greater-than-one criteria suggested two components, and, in combination explained 63% of the variance. The inflexion in the scree plot justified retaining one component. A priorily, we researchers were expecting three components – which would explain 71% of the variance. Correspondingly, we investigated two and three component solutions with orthogonal (varimax) and oblique (oblimin) procedures. Given the significant correlations (ranging from .25 to .58) and the correspondence of items loading on the a priorili hypothesized components, we determined that an oblique, three-component, solution was most appropriate.

The rotated solution, as shown in Table 1 and Figure 1, yielded three interpretable components, each listed with the proportion of variance accounted for: traditional pedagogy (35%), valued-by-me (21%), and socially and culturally responsive pedagogy (15%).

Regarding the Table 1, I would include a table with ALL the values, bolding those with component membership. This is easy, though, because we can export it to a .csv file and

```
pca0BL3fb <- psych::principal(items, nfactors = 3, rotate = "oblimin")
pca_table0BL3fb <- psych::print.psych(pca0BL3fb, sort = TRUE)
```

	item	TC1	TC3	TC2	h2	u2	com
ClearResponsibilities	4	0.95	-0.06	-0.05	0.80	0.20	1.0
Feedback	6	0.80	-0.06	0.12	0.67	0.33	1.1
EffectiveAnswers	5	0.76	0.15	-0.06	0.71	0.29	1.1
EquitableEval	12	0.75	-0.10	0.18	0.62	0.38	1.2
ClearOrganization	7	0.68	0.29	-0.15	0.71	0.29	1.5
MultPerspectives	9	0.59	0.03	0.39	0.71	0.29	1.7
ClearPresentation	8	0.57	0.41	-0.02	0.76	0.24	1.8
IncrUnderstanding	2	0.06	0.84	-0.07	0.75	0.25	1.0
IncrInterest	3	0.05	0.79	0.13	0.74	0.26	1.1
ValObjectives	1	-0.08	0.71	0.15	0.52	0.48	1.1
DEIIntegration	11	0.00	-0.02	0.92	0.84	0.16	1.0

```
InclusvClassrm      10  0.08  0.22  0.73 0.74 0.26 1.2
```

	TC1	TC3	TC2
--	-----	-----	-----

SS loadings	4.23	2.50	1.83
Proportion Var	0.35	0.21	0.15
Cumulative Var	0.35	0.56	0.71
Proportion Explained	0.49	0.29	0.21
Cumulative Proportion	0.49	0.79	1.00

With component correlations of

	TC1	TC3	TC2
TC1	1.00	0.58	0.39
TC3	0.58	1.00	0.25
TC2	0.39	0.25	1.00

Mean item complexity = 1.2

Test of the hypothesis that 3 components are sufficient.

The root mean square of the residuals (RMSR) is 0.06  
with the empirical chi square 115.69 with prob < 0.000000000041

Fit based upon off diagonal values = 0.99

```
pca0BL3fb_table <- round(pca0BL3fb$loadings, 3)
write.table(pca0BL3fb_table, file = "pca0BL3f_table.csv", sep = ",",
            row.names = TRUE)
pca0BL3fb_table
```

Loadings:

	TC1	TC3	TC2
ValObjectives	0.712	0.151	
IncrUnderstanding	0.844		
IncrInterest	0.787	0.132	
ClearResponsibilities	0.947		
EffectiveAnswers	0.764	0.154	
Feedback	0.800		0.119
ClearOrganization	0.685	0.293	-0.149
ClearPresentation	0.574	0.413	
MultPerspectives	0.593		0.391
InclusvClassrm		0.218	0.730
DEIIntegration			0.921
EquitableEval	0.751		0.184

	TC1	TC3	TC2
--	-----	-----	-----

SS loadings	3.854	2.185	1.655
Proportion Var	0.321	0.182	0.138
Cumulative Var	0.321	0.503	0.641

**8.9.8 Explanation to grader**

# Chapter 9

# Principal Axis Factoring

## [Screencasted Lecture Link](#)

This is the second lesson of *exploratory* principal components analysis (PCA) and factor analysis (EFA/PAF). This time the focus is on actual *factor analysis*. There are numerous approaches to this process (e.g., principal components analysis, parallel analyses). In this lesson I will demonstrate principal axis factoring (PAF).

## 9.1 Navigating this Lesson

There is about an hour-and-a-half of lecture. If you work through the materials with me it would be plan for an additional two hours.

While the majority of R objects and data you will need are created within the R script that sources the chapter, occasionally there are some that cannot be created from within the R framework. Additionally, sometimes links fail. All original materials are provided at the [GitHub site](#) that hosts the book. More detailed guidelines for ways to access all these materials are provided in the OER's [introduction](#)

### 9.1.1 Learning Objectives

Focusing on this lesson's materials, make sure you can:

- Distinguish between PCA and PAF on several levels:
  - recognize PCA and PAF from a path diagram
  - define keywords associated with each: factor loadings, linear components, describe v. explain.
- Recognize/define an identity matrix – what test would you use to diagnose it?
- Recognize/define multicollinearity and singularity – what test would you use to diagnose it?
- Describe the desired pattern of “loadings” (i.e., the relative weights of an item on its own scale compared to other scales)
- Compare the results from item analysis, PCA, PAF, and omega.

### 9.1.2 Planning for Practice

In each of these lessons I provide suggestions for practice that allow you to select one or more problems that are graded in difficulty. The least complex is to change the random seed in the research and rework the problem demonstrated in the lesson. The results *should* map onto the ones obtained in the lecture.

The second option involves utilizing one of the simulated datasets available in this OER. The **last lesson** in the OER contains three simulations that could be used for all of the statistics-based practice suggestions. Especially if you started with one of these examples in an earlier lesson, I highly recommend you continue with that.

Alternatively, Keum et al.'s Gendered Racial Microaggressions Scale for Asian American Women [2018] will be used in the lessons on confirmatory factor analysis and Conover et al.'s [2017] Ableist Microaggressions Scale is used in the lesson on invariance testing. Both of these would be suitable for the PCA and PAF homework assignments.

As a third option, you are welcome to use data to which you have access and is suitable for PCA. These could include other vignettes from this OER, other simulated data, or your own data (presuming you have permission to use it). In any case, please plan to:

- Properly format and prepare the data.
- Conduct diagnostic tests to determine the suitability of the data for PCA.
- Conducting tests to guide the decisions about number of factors to extract.
- Conducting orthogonal and oblique extractions (at least two each with different numbers of factors).
- Selecting one solution and preparing an APA style results section (with table and figure).
- Compare your results in light of any other psychometrics lessons where you have used this data (especially the **item analysis** and **PCA** lessons).

### 9.1.3 Readings & Resources

In preparing this chapter, I drew heavily from the following resource(s). Other resources are cited (when possible, linked) in the text with complete citations in the reference list.

- Revelle, William. (n.d.). Chapter 6: Constructs, components, and factor models. In *An introduction to psychometric theory with applications in R*. Retrieved from <https://personality-project.org/r/book/#chapter6>
  - pp. 150 to 167. Stop at “Non-Simple Structure Solutions: The Simplex and Circumplex.”
  - A simultaneously theoretical review of psychometric theory while working with R and data to understand the concepts.
- Revelle, W. (2019). *How To: Use the psych package for Factor Analysis and data reduction*.
  - Treat as reference. Pages 13 through 24 provide technical information about what we are doing.

### 9.1.4 Packages

The packages used in this lesson are embedded in this code. When the hashtags are removed, the script below will (a) check to see if the following packages are installed on your computer and, if not (b) install them.

```
# will install the package if not already installed
# if(!require(psych)){install.packages('psych')}
# if(!require(tidyverse)){install.packages('tidyverse')}
# if(!require(MASS)){install.packages('MASS')}
# if(!require(sjstats)){install.packages('sjstats')}
# if(!require(apaTables)){install.packages('apaTables')}
# if(!require(qualtrics)){install.packages('qualtrics')}
```

## 9.2 Exploratory Factor Analysis (with a quick contrast to PCA)

Whereas principal components analysis (PCA) is a regression analysis technique, principal factor analysis is a latent variable model [?].

Exploratory factor analysis has a rich history. In 1904, Spearman used it for a single factor. In 1947, Thurstone generalized it to multiple factors. Factor analysis is frequently used and controversial.

Factor analysis and principal components are commonly confused:

**Principal components** is

- linear sums of variables,
- solved with an eigenvalue or singular decomposition,
- represented by an  $n * n$  matrix in terms of the first  $k$  components and attempts to reproduce all of the  $R$  matrix, and
- paths which point from the items to a total scale score – all represented as observed/manifest (square) variables.

**Factor analysis** is

- linear sums of unknown factors,
- estimated as best fitting solutions, normally through iterative procedures, and
- controversial. Because:
  - At the *structural* level (i.e., covariance or correlation matrix), there are normally more observed variables than parameters to estimate them and the procedure seeks to find the best fitting solution using ordinary least squares, weighted least squares, or maximum likelihood.
  - At the *data* level, although scores can be estimated, the model is indeterminate.
  - This leads some to argue for using principal components; however, fans of factor analysis suggest that it is useful for constructing and evaluating theories.

- an attempt to model only the *common* part of the matrix, which means all of the off-diagonal elements and the common part of the diagonal (the *communalities*); the *uniquenesses* are the non-common (leftover) part
  - Stated another way, the factor model partitions the correlation or covariance matrix into
    - \* *common factors*,  $FF'$ , and
    - \* that which is *unique*,  $U^2$  (the diagonal matrix of *uniquenesses*)
- paths which point from the latent variable (LV) representing the factor (oval) to the items (squares) illustrating that the factor/LV “causes” the item’s score

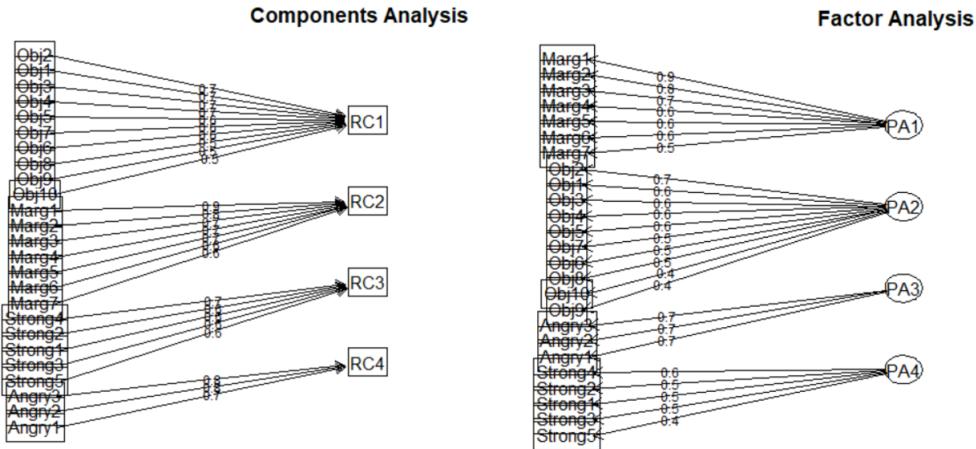


Figure 9.1: Comparison of path models for PCA and EFA

Our focus today is on the PAF approach to scale construction. By utilizing the same research vignette as in the [PCA lesson](#), we can identify similarities in differences in the approach, results, and interpretation. Let’s first take a look at the workflow for PAF.

### 9.3 PAF Workflow

Below is a screenshot of the workflow. The original document is located in the [GitHub site](#) that hosts the ReCentering Psych Stats: Psychometrics OER. You may find it refreshing that, with the exception of the change from “components” to “factors,” the workflow for PCA and PAF are quite similar.

Steps in the process include:

- Creating an items only dataframe where all items are scaled in the same direction (i.e., negatively worded items are reverse scored).
- Conducting tests that assess the statistical assumptions of PAF to ensure that the data is appropriate for PAF.
- Determining the number of factors (think “subscale”) to extract.
- Conducting the factor extraction – this process will likely occur iteratively,

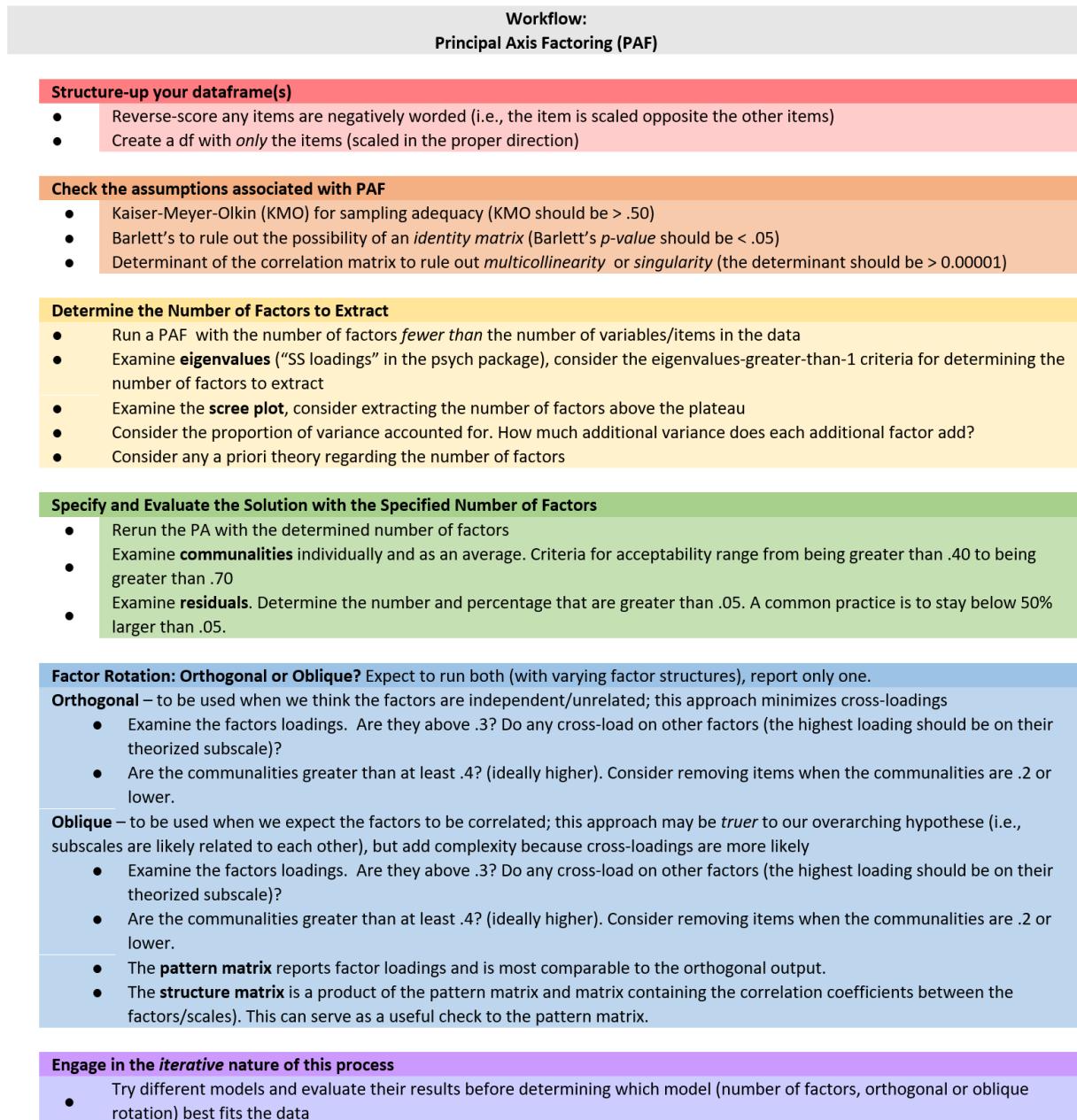


Figure 9.2: Image of the workflow for PAF

- exploring orthogonal (uncorrelated/independent) and oblique (correlated) factors, and
- changing the number of factors to extract

Because the intended audience for the ReCentering Psych Stats OER is the scientist-practitioner-advocate, this lesson focuses on the workflow and decisions. As you might guess, the details of PAF can be quite complex. Some important notions to consider that may not be obvious from lesson, are these:

- The values of factor loadings are directly related to the correlation matrix.
  - Although I do not explain this in detail, nearly every analytic step attempts to convey this notion by presenting equivalent analytic options using the raw data and correlation matrix.
- PAF (like PCA and related EFA procedures) is about *dimension reduction* – our goal is fewer factors (think subscales) than there are items.
  - In this lesson’s vignette there are 25 items on the scale, and we will have 4 subscales.
- As a latent variable procedure, PAF is both *exploratory* and *factor analysis*. This is in contrast to our prior [PCA lesson](#). Recall that PCA is a regression-based model and therefore not “factor analysis.”
- Matrix algebra (e.g., using the transpose of a matrix, multiplying matrices together) plays a critical role in the analytic solution.

## 9.4 Research Vignette

This lesson’s research vignette emerges from Lewis and Neville’s Gendered Racial Microaggressions Scale for Black Women [2015]. The article reports on two separate studies that comprised the development, refinement, and psychometric evaluation of two parallel versions (stress appraisal, frequency) of the scale. Below, I simulate data from the final construction of the stress appraisal version as the basis of the lecture. Items were on a 6-point Likert scale ranging from 0 (*not at all stressful*) to 5 (*extremely stressful*).

Lewis and Neville [2015] reported support for a total scale score (25 items) and four subscales. Below, I list the four subscales along with the items and their abbreviation. At the outset, let me provide a content advisory. For those who hold this particular identity (or related identities) the content in the items may be upsetting. In other lessons, I often provide a variable name that gives an indication of the primary content of the item. In the case of the GRMS, I will simply provide an abbreviation of the subscale name and its respective item number. This will allow us to easily inspect the alignment of the item with its intended factor, and hopefully minimize discomfort.

If you are not a member of this particular identity, I encourage you to learn about these microaggressions by reading the article in its entirety. Please do not ask members of this group to explain why these microaggressions are harmful or ask if they have encountered them. The four factors, number of items, and sample item are as follows:

- Assumptions of Beauty and Sexual Objectification (10 items)

- Unattractive because of size of butt (Obj1)
- Negative comments about size of facial features (Obj2)
- Imitated the way they think Black women speak (Obj3)
- Someone made me feel unattractive (Obj4)
- Negative comment about skin tone (Obj5)
- Someone assumed I speak a certain way (Obj6)
- Objectified me based on physical features(Obj7)
- Someone assumed I have a certain body type (Obj8; stress only)
- Made a sexually inappropriate comment (Obj9)
- Negative comments about my hair when natural (Obj10)
- Assumed I was sexually promiscuous (frequency only; not used in this simulation)
- Silenced and Marginalized (7 items)
  - I have felt unheard (Marg1)
  - My comments have been ignored (Marg2)
  - Someone challenged my authority (Marg3)
  - I have been disrespected in workplace (Marg4)
  - Someone has tried to “put me in my place” (Marg5)
  - Felt excluded from networking opportunities (Marg6)
  - Assumed I did not have much to contribute to the conversation (Marg7)
- Strong Black Woman Stereotype (5 items)
  - Someone assumed I was sassy and straightforward (Str1; stress only)
  - I have been told that I am too independent (Str2)
  - Someone made me feel exotic as a Black woman (Str2; stress only)
  - I have been told that I am too assertive
  - Assumed to be a strong Black woman
- Angry Black Woman Stereotype (3 items)
  - Someone has told me to calm down (Ang1)
  - Perceived to be “angry Black woman” (Ang2)
  - Someone accused me of being angry when speaking calm (Ang3)

Three additional scales were reported in the Lewis and Neville article [2015]. Because (a) the focus of this lesson is on exploratory factor analytic approaches and, therefore, only requires item-level data for the scale, and (b) the article does not include correlations between the subscales/scales of all involved measures, I only simulated item-level data for the GRMS items.

Below, I walk through the data simulation. This is not an essential portion of the lesson, but I will lecture it in case you are interested. None of the items are negatively worded (relative to the other items), so there is no need to reverse-score any items.

Simulating the data involved using factor loadings, means, standard deviations, and correlations between the scales. Because the simulation will produce “out-of-bounds” values, the code below rescales the scores into the range of the Likert-type scaling and rounds them to whole values.

```
# Entering the intercorrelations means and standard deviations from
# the journal article
```

```

LewisGRMS_generating_model <- "
  #measurement model
  Objectification =~ .69*Obj1 + .69*Obj2 + .60*Obj3 + .59*Obj4 + .55*Obj5 + .55*Obj6 + .
  Marginalized =~ .93*Marg1 + .81*Marg2 + .69*Marg3 + .67*Marg4 + .61*Marg5 + .58*Marg6 +
  Strong =~ .59*Str1 + .55*Str2 + .54*Str3 + .54*Str4 + .51*Str5
  Angry =~ .70*Ang1 + .69*Ang2 + .68*Ang3

  #Means
  Objectification ~ 1.85*1
  Marginalized ~ 2.67*1
  Strong ~ 1.61*1
  Angry ~ 2.29*1

  #Correlations
  Objectification ~~ .63*Marginalized
  Objectification ~~ .66*Strong
  Objectification ~~ .51*Angry

  Marginalized ~~ .59*Strong
  Marginalized ~~ .62*Angry

  Strong ~~ .61*Angry

  ""

set.seed(240311)
dfGRMS <- lavaan::simulateData(model = LewisGRMS_generating_model, model.type = "sem",
  meanstructure = T, sample.nobs = 259, standardized = FALSE)

# used to retrieve column indices used in the rescaling script below
col_index <- as.data.frame(colnames(dfGRMS))

# The code below loops through each column of the dataframe and
# assigns the scaling accordingly Rows 1 thru 26 are the GRMS items

for (i in 1:ncol(dfGRMS)) {
  if (i >= 1 & i <= 25) {
    dfGRMS[, i] <- scales::rescale(dfGRMS[, i], c(0, 5))
  }
}

# rounding to integers so that the data resembles that which was
# collected
library(tidyverse)
dfGRMS <- dfGRMS %>%
  round(0)

```

```
# quick check psych::describe(dfGRMS)
```

The optional script below will let you save the simulated data to your computing environment as either an .rds object (preserves any formatting you might do) or a .csv file (think “Excel lite”).

An .rds file preserves all formatting to variables prior to the export and re-import. For the purpose of this chapter, you don’t need to do either. That is, you can re-simulate the data each time you work the problem.

```
# to save the df as an .rds (think 'R object') file on your computer;
# it should save in the same file as the .rmd file you are working
# with saveRDS(dfGRMS, 'dfGRMS.rds') bring back the simulated dat
# from an .rds file dfGRMS <- readRDS('dfGRMS.rds')
```

If you save the .csv file and bring it back in, you will lose any formatting (e.g., ordered factors will be interpreted as character variables).

```
# write the simulated data as a .csv write.table(dfGRMS,
# file='dfGRMS.csv', sep=',', col.names=TRUE, row.names=FALSE) bring
# back the simulated dat from a .csv file dfGRMS <- read.csv
# ('dfGRMS.csv', header = TRUE)
```

Before moving on, I want to acknowledge that (at their first drafting), I try to select research vignettes that have been published within the prior 5 years. With a publication date of 2015, this article clearly falls outside that range. I have continued to include it because (a) the scholarship is superior – especially as the measure captures an intersectional identity, (b) the article has been a model for research that follows (e.g., Keum et al’s [2018] Gendered Racial Microaggression Scale for Asian American Women), and (c) there is often a time lag between the initial publication of a psychometric scale and its use. A key reason I have retained the GRMS as a psychometrics research vignette is that in [ReCentering Psych Stats: Multivariate Modeling](#), GRMS scales are used in a couple of more recently published research vignettes.

## 9.5 Working the Vignette

It may be useful to recall how we might understand factors in the psychometric sense:

- clusters of correlated items in an  $R$ -matrix
- statistical entities that can be plotted as classification axes where coordinates of variables along each axis represent the strength of the relationship between that variable to each factor.
- mathematical equations, resembling regression equations, where each variable is represented according to its relative weight

### 9.5.1 Data Prep

Since the first step is data preparation, let’s start by:

- reverse coding any items that are phrased in the opposite direction
- creating a *df* (as an object) that only contains the items in their properly scored direction (i.e., you might need to replace the original item with the reverse-coded item); there should be no other variables (e.g., ID, demographic variables, other scales) in this df
  - because the GRMS has no items like this we can skip these two steps

Our example today requires no reverse coding and the dataset I simulated only has item-level data (with no ID and no other variables). This means we are ready to start the PAF process.

Let's take a look at (and make an object of) the correlation matrix.

```
GRMSr <- cor(dfGRMS) #correlation matrix (with the negatively scored item already reversed) c
round(GRMSr, 2)
```

	Obj1	Obj2	Obj3	Obj4	Obj5	Obj6	Obj7	Obj8	Obj9	Obj10	Marg1	Marg2	Marg3
Obj1	1.00	0.35	0.25	0.27	0.28	0.25	0.28	0.35	0.15	0.24	0.19	0.25	0.17
Obj2	0.35	1.00	0.31	0.25	0.27	0.23	0.31	0.28	0.26	0.24	0.22	0.21	0.25
Obj3	0.25	0.31	1.00	0.24	0.28	0.28	0.20	0.25	0.21	0.22	0.17	0.23	0.17
Obj4	0.27	0.25	0.24	1.00	0.39	0.23	0.28	0.30	0.26	0.28	0.22	0.18	0.14
Obj5	0.28	0.27	0.28	0.39	1.00	0.15	0.18	0.29	0.25	0.20	0.17	0.20	0.23
Obj6	0.25	0.23	0.28	0.23	0.15	1.00	0.20	0.14	0.21	0.12	0.10	0.14	0.05
Obj7	0.28	0.31	0.20	0.28	0.18	0.20	1.00	0.31	0.19	0.28	0.30	0.21	0.20
Obj8	0.35	0.28	0.25	0.30	0.29	0.14	0.31	1.00	0.19	0.23	0.27	0.14	0.14
Obj9	0.15	0.26	0.21	0.26	0.25	0.21	0.19	0.19	1.00	0.20	0.10	0.12	0.21
Obj10	0.24	0.24	0.22	0.28	0.20	0.12	0.28	0.23	0.20	1.00	0.09	0.12	0.17
Marg1	0.19	0.22	0.17	0.22	0.17	0.10	0.30	0.27	0.10	0.09	1.00	0.43	0.41
Marg2	0.25	0.21	0.23	0.18	0.20	0.14	0.21	0.14	0.12	0.12	0.43	1.00	0.35
Marg3	0.17	0.25	0.17	0.14	0.23	0.05	0.20	0.14	0.21	0.17	0.41	0.35	1.00
Marg4	0.19	0.18	0.24	0.26	0.20	0.10	0.25	0.24	0.07	0.12	0.38	0.23	0.32
Marg5	0.17	0.22	0.21	0.27	0.25	0.16	0.23	0.19	0.19	0.11	0.41	0.40	0.25
Marg6	0.18	0.27	0.16	0.23	0.22	0.26	0.28	0.26	0.15	0.26	0.35	0.27	0.25
Marg7	0.13	0.19	0.14	0.19	0.06	0.17	0.16	0.14	0.10	0.11	0.31	0.33	0.20
Str1	0.22	0.18	0.14	0.06	0.23	0.07	0.25	0.17	0.19	0.10	0.19	0.25	0.20
Str2	0.19	0.18	0.19	0.19	0.12	0.15	0.13	0.06	0.18	0.19	0.12	0.18	0.17
Str3	0.10	0.09	0.09	0.08	0.11	0.09	0.19	0.05	0.12	0.10	0.13	0.18	0.10
Str4	0.09	0.14	0.18	0.15	0.12	0.08	0.07	0.13	0.05	0.02	0.08	0.12	0.08
Str5	0.20	0.15	0.15	0.08	0.19	0.11	0.15	0.04	0.07	0.09	0.10	0.23	0.12
Ang1	0.06	0.07	0.07	0.09	0.12	0.04	0.15	0.07	0.17	0.06	0.16	0.23	0.18
Ang2	0.06	0.15	0.08	0.06	0.09	0.20	0.13	-0.03	0.00	0.14	0.17	0.19	0.19
Ang3	0.21	0.13	0.11	0.14	0.11	0.16	0.23	0.07	0.06	0.08	0.28	0.28	0.11
	Marg4	Marg5	Marg6	Marg7	Str1	Str2	Str3	Str4	Str5	Ang1	Ang2	Ang3	
Obj1	0.19	0.17	0.18	0.13	0.22	0.19	0.10	0.09	0.20	0.06	0.06	0.21	
Obj2	0.18	0.22	0.27	0.19	0.18	0.18	0.09	0.14	0.15	0.07	0.15	0.13	
Obj3	0.24	0.21	0.16	0.14	0.14	0.19	0.09	0.18	0.15	0.07	0.08	0.11	
Obj4	0.26	0.27	0.23	0.19	0.06	0.19	0.08	0.15	0.08	0.09	0.06	0.14	
Obj5	0.20	0.25	0.22	0.06	0.23	0.12	0.11	0.12	0.19	0.12	0.09	0.11	
Obj6	0.10	0.16	0.26	0.17	0.07	0.15	0.09	0.08	0.11	0.04	0.20	0.16	

Obj7	0.25	0.23	0.28	0.16	0.25	0.13	0.19	0.07	0.15	0.15	0.15	0.13	0.23
Obj8	0.24	0.19	0.26	0.14	0.17	0.06	0.05	0.13	0.04	0.07	-0.03	0.07	
Obj9	0.07	0.19	0.15	0.10	0.19	0.18	0.12	0.05	0.07	0.17	0.00	0.06	
Obj10	0.12	0.11	0.26	0.11	0.10	0.19	0.10	0.02	0.09	0.06	0.14	0.08	
Marg1	0.38	0.41	0.35	0.31	0.19	0.12	0.13	0.08	0.10	0.16	0.17	0.28	
Marg2	0.23	0.40	0.27	0.33	0.25	0.18	0.18	0.12	0.23	0.23	0.19	0.28	
Marg3	0.32	0.25	0.25	0.20	0.20	0.17	0.10	0.08	0.12	0.18	0.19	0.11	
Marg4	1.00	0.30	0.26	0.16	0.10	0.21	0.05	0.06	0.03	0.12	0.22	0.17	
Marg5	0.30	1.00	0.29	0.28	0.16	0.13	0.16	0.14	0.18	0.12	0.14	0.21	
Marg6	0.26	0.29	1.00	0.20	0.13	0.18	0.15	0.13	0.08	0.11	0.21	0.12	
Marg7	0.16	0.28	0.20	1.00	0.14	0.05	0.04	0.02	0.12	0.17	0.13	0.09	
Str1	0.10	0.16	0.13	0.14	1.00	0.21	0.30	0.23	0.23	0.18	0.05	0.10	
Str2	0.21	0.13	0.18	0.05	0.21	1.00	0.20	0.20	0.12	0.16	0.12	0.16	
Str3	0.05	0.16	0.15	0.04	0.30	0.20	1.00	0.27	0.18	0.20	0.07	0.15	
Str4	0.06	0.14	0.13	0.02	0.23	0.20	0.27	1.00	0.12	0.15	0.03	0.02	
Str5	0.03	0.18	0.08	0.12	0.23	0.12	0.18	0.12	1.00	0.22	0.15	0.11	
Ang1	0.12	0.12	0.11	0.17	0.18	0.16	0.20	0.15	0.22	1.00	0.24	0.23	
Ang2	0.22	0.14	0.21	0.13	0.05	0.12	0.07	0.03	0.15	0.24	1.00	0.25	
Ang3	0.17	0.21	0.12	0.09	0.10	0.16	0.15	0.02	0.11	0.23	0.25	1.00	

In case you want to examine it in sections (easier to view):

```
# round(GRMSr[,1:8], 2) round(GRMSr[,9:16], 2) round(GRMSr[,17:25],  
# 2)
```

As with PCA, we can analyze the data with either raw data or correlation matrix. I will do both to demonstrate (a) that it's possible and to (b) continue emphasizing that this is a *structural* analysis. That is, we are trying to see if our more parsimonious extraction *reproduces* this original correlation matrix.

### 9.5.1.1 Three Diagnostic Tests to Evaluate the Appropriateness of the Data for Factor (or Component) Analysis

Here's a snip of our location in the PAF workflow.

#### Check the assumptions associated with PAF

- Kaiser-Meyer-Olkin (KMO) for sampling adequacy (KMO should be > .50)
- Barlett's to rule out the possibility of an *identity matrix* (Barlett's *p*-value should be < .05)
- Determinant of the correlation matrix to rule out *multicollinearity* or *singularity* (the determinant should be > 0.00001)

Figure 9.3: Image of our location in the PAF workflow

**9.5.1.1.1 Is my sample adequate for PAF?** We return to the **KMO** (Kaiser-Meyer-Olkin), an index of *sampling adequacy* that can be used with the actual sample to let us know if the sample size is sufficient (or if we should collect more data).

Kaiser's 1974 recommendations were:

- bare minimum of .5
- values between .5 and .7 are mediocre
- values between .7 and .8 are good
- values above .9 are superb

We use the *KMO()* function from the *psych* package with either raw or matrix dat.

```
psych::KMO(dfGRMS)
```

```
Kaiser-Meyer-Olkin factor adequacy
Call: psych::KMO(r = dfGRMS)
Overall MSA =  0.85
MSA for each item =
  Obj1  Obj2  Obj3  Obj4  Obj5  Obj6  Obj7  Obj8  Obj9  Obj10 Marg1 Marg2 Marg3
  0.87  0.91  0.88  0.85  0.85  0.80  0.90  0.85  0.81  0.85  0.86  0.89  0.86
Marg4 Marg5 Marg6 Marg7 Str1 Str2 Str3 Str4 Str5 Ang1 Ang2 Ang3
  0.86  0.90  0.89  0.84  0.83  0.85  0.82  0.74  0.84  0.78  0.76  0.81
```

```
# psych::KMO(GRMSr) #for the KMO function, do not specify sample size
# if using the matrix form of the data
```

We examine the KMO values for both the overall matrix and the individual items.

At the matrix level, our *KMO* = .85, which falls in between Kaiser's definitions of *good* and *superb*.

At the item level, the KMO should be > .50. Variables with values below .5 should be evaluated for exclusion from the analysis (or run the analysis with and without the variable and compare the difference). Because removing/adding variables impacts the KMO, be sure to re-evaluate.

At the item level, our KMO values range between .74 (Str4) and .91 (Obj2).

Considering both item- and matrix- levels, we conclude that the sample size and the data are adequate for factor (or component) analysis.

**9.5.1.1.2 Are the correlations among the variables big enough to be analyzed?**  
**Bartlett's** lets us know if a matrix is an *identity matrix*. In an identity matrix all correlation coefficients (everything on the off-diagonal) would be 0.0 (and everything on the diagonal would be 1.0).

A significant Barlett's (i.e.,  $p < .05$ ) tells that the *R*-matrix is not an identity matrix. That is, there are some relationships between variables that can be analyzed.

The *cortest.bartlett()* function in the *psych* package and can be run either from the raw data or R matrix formats.

```
psych::cortest.bartlett(dfGRMS)  #from the raw data
```

R was not square, finding R from data

Our Bartlett's test is significant:  $\chi^2(300) = 1217.508, p < .001$ . This supports a factor (or component) analytic approach for investigating the data.

**9.5.1.1.3 Is there multicollinearity or singularity in my data?** The determinant of the correlation matrix should be greater than 0.00001 (that would be 4 zeros before the 1). If it is smaller than 0.00001 then we may have an issue with *multicollinearity* (i.e., variables that are too highly correlated) or *singularity* (variables that are perfectly correlated).

The determinant function comes from base R. It is easiest to compute when the correlation matrix is the object. However, it is also possible to specify the command to work with the raw data.

```
# det(GRMSr)
det(cor(dfGRMS)) #if using the raw data
```

With a value of 0.0075, our determinant is greater than the 0.00001 requirement. If it were not, then we could identify problematic variables (i.e., those correlating too highly with others and those not correlating sufficiently with others) and re-run the diagnostic statistics.

### 9.5.1.2 APA Style Summary So Far

Data screening were conducted to determine the suitability of the data for principal axis factoring. The Kaiser-Meyer-Olkin measure of sampling adequacy (KMO; Kaiser, 1970) represents the ratio of the squared correlation between variables to the squared partial correlation between variables. KMO ranges from 0.00 to 1.00; values closer to 1.00 indicate that the patterns of correlations are relatively compact, and that factor analysis should yield distinct and reliable factors (Field, 2012). In our dataset, the KMO value was .85, indicating acceptable sampling adequacy. The Barlett's Test of Sphericity examines whether the population correlation matrix resembles an identity matrix (Field,

2012). When the  $p$  value for the Bartlett's test is  $< .05$ , we are fairly certain we have clusters of correlated variables. In our dataset,  $\chi^2(300) = 1217.508, p < .001$ , indicating the correlations between items are sufficiently large enough for principal axis factoring. The determinant of the correlation matrix alerts us to any issues of multicollinearity or singularity and should be larger than 0.00001. Our determinant was 0.0075, supporting the suitability of our data for analysis.

*Note:* If this looks familiar, it is! The same diagnostics are used in PAF and **PCA**.

### 9.5.2 Principal Axis Factoring (PAF)

Here's a snip of our location in the PAF workflow.

#### Determine the Number of Factors to Extract

- Run a PAF with the number of factors *fewer than* the number of variables/items in the data
- Examine **eigenvalues** ("SS loadings" in the *psych* package), consider the eigenvalues-greater-than-1 criteria for determining the number of factors to extract
- Examine the **scree plot**, consider extracting the number of factors above the plateau
- Consider the proportion of variance accounted for. How much additional variance does each additional factor add?
- Consider any *a priori* theory regarding the number of factors

Figure 9.4: Image of our location in the PAF workflow

We can use the *fa()* function, specifying *fm = "pa"* from the *psych* package with raw or matrix data.

One difference from PCA is that factor analysis will not (cannot) calculate as many factors as there are items. This means that we should select a reasonable number, like 20 (since there are 25 items). However, I received a number of errors/warnings and 13 is the first number that would run. I also received the warning, "maximum iteration exceeded." Therefore, I increased "max.iter" to 100.

Our goal is to begin to get an idea of the cumulative variance explained, and number of factors to extract. If we think there are four factors, we simply need to specify more than four factors on the *nfactors = ##* command. As long as that number is less than the total number of items, it does not matter what that number is.

```
# grmsPAF1 <- psych::fa(GRMSr, nfactors=10, fm = 'pa', max.iter =
# 100, rotate='none')# using the matrix data and specifying the # of
# factors.

grmsPAF1 <- psych::fa(dfGRMS, nfactors = 13, fm = "pa", max.iter = 100,
rotate = "none") # using raw data and specifying the max number of factors

# I received the warning 'maximum iteration exceeded'. It gave
# output, but it's best if we don't get that warning, so I increased
# it to 100.

grmsPAF1 #this object holds a great deal of information
```

```

Factor Analysis using method = pa
Call: psych::fa(r = dfGRMS, nfactors = 13, rotate = "none", max.iter = 100,
  fm = "pa")
Standardized loadings (pattern matrix) based upon correlation matrix
    PA1   PA2   PA3   PA4   PA5   PA6   PA7   PA8   PA9   PA10  PA11  PA12
Obj1  0.49 -0.20 -0.01  0.13 -0.01  0.11 -0.17 -0.16  0.01 -0.09  0.21  0.04
Obj2  0.51 -0.19 -0.03  0.13  0.07 -0.02 -0.04  0.00  0.06 -0.08  0.09 -0.01
Obj3  0.45 -0.17 -0.01  0.05  0.03  0.09  0.00  0.12 -0.04 -0.04  0.14  0.10
Obj4  0.49 -0.25 -0.11  0.03 -0.06  0.04  0.04  0.03 -0.09  0.18 -0.09  0.19
Obj5  0.54 -0.46  0.06 -0.50 -0.46  0.01 -0.09  0.08  0.04 -0.01 -0.06 -0.07
Obj6  0.38 -0.15  0.00  0.39 -0.08  0.07 -0.06  0.36 -0.08  0.02  0.09 -0.18
Obj7  0.51 -0.04 -0.04  0.16  0.06  0.03 -0.02 -0.22  0.05  0.02 -0.05 -0.12
Obj8  0.46 -0.29 -0.17 -0.02  0.15  0.05 -0.08 -0.25 -0.03  0.24  0.09 -0.11
Obj9  0.40 -0.30  0.16  0.09  0.06 -0.44  0.35  0.01 -0.27 -0.09 -0.03 -0.04
Obj10 0.39 -0.23 -0.04  0.21  0.00 -0.01  0.12 -0.15  0.33 -0.04 -0.19  0.20
Marg1 0.56  0.33 -0.24 -0.12  0.11 -0.12 -0.15 -0.06 -0.07  0.05 -0.11 -0.07
Marg2 0.55  0.30  0.03 -0.07  0.02 -0.14 -0.21  0.06  0.00 -0.07  0.05  0.15
Marg3 0.48  0.18 -0.09 -0.17  0.08 -0.19  0.11  0.01  0.12 -0.17 -0.01  0.00
Marg4 0.51  0.21 -0.47 -0.20  0.03  0.32  0.38  0.00 -0.08 -0.08  0.14 -0.02
Marg5 0.52  0.15 -0.09 -0.10  0.04 -0.10 -0.13  0.14 -0.13  0.03 -0.08  0.09
Marg6 0.51  0.02 -0.13  0.08  0.09  0.00 -0.01  0.12  0.15  0.08 -0.24 -0.16
Marg7 0.37  0.17 -0.10  0.05  0.08 -0.22 -0.12  0.12  0.06  0.13  0.12  0.08
Str1  0.40  0.02  0.34 -0.14  0.18  0.01 -0.07 -0.14  0.02 -0.18  0.07 -0.14
Str2  0.36  0.01  0.17  0.07  0.07  0.14  0.17  0.04 -0.01 -0.14 -0.05  0.11
Str3  0.30  0.10  0.38 -0.02  0.13  0.13 -0.01 -0.03 -0.03 -0.05 -0.17 -0.07
Str4  0.27 -0.03  0.36 -0.13  0.28  0.32  0.00  0.16 -0.10  0.16 -0.09  0.08
Str5  0.30  0.07  0.27 -0.02 -0.07  0.01 -0.11  0.04  0.13 -0.06  0.16  0.04
Ang1  0.34  0.32  0.38 -0.03 -0.17 -0.10  0.28 -0.11  0.09  0.34  0.16 -0.01
Ang2  0.30  0.28  0.00  0.15 -0.24  0.08  0.12  0.16  0.22 -0.02 -0.03 -0.08
Ang3  0.38  0.32  0.06  0.25 -0.39  0.12 -0.10 -0.24 -0.29 -0.07 -0.12  0.03
    PA13   h2     u2 com
Obj1 -0.05  0.42  0.5768 2.8
Obj2 -0.15  0.36  0.6397 1.9
Obj3 -0.07  0.30  0.7008 2.1
Obj4  0.11  0.43  0.5749 2.7
Obj5  0.00  1.00  0.0031 4.1
Obj6  0.05  0.51  0.4897 4.2
Obj7  0.12  0.38  0.6219 1.9
Obj8 -0.07  0.50  0.4962 4.2
Obj9 -0.01  0.69  0.3115 5.1
Obj10 0.05  0.48  0.5243 5.2
Marg1 -0.07  0.58  0.4213 2.9
Marg2  0.01  0.50  0.5037 2.4
Marg3 -0.20  0.44  0.5594 3.2
Marg4  0.12  0.86  0.1379 4.9
Marg5  0.12  0.41  0.5908 2.2
Marg6 -0.03  0.42  0.5816 2.5
Marg7  0.10  0.31  0.6869 4.2

```

Str1	0.11	0.43	0.5733	4.2
Str2	-0.02	0.25	0.7472	3.3
Str3	0.15	0.34	0.6630	3.7
Str4	-0.17	0.51	0.4923	6.2
Str5	0.10	0.25	0.7537	4.0
Ang1	-0.02	0.64	0.3585	6.3
Ang2	-0.07	0.35	0.6509	5.8
Ang3	-0.09	0.66	0.3388	6.2

Mean item complexity = 3.8

Test of the hypothesis that 13 factors are sufficient.

df null model = 300 with the objective function = 4.89 with Chi Square = 1217.51  
df of the model are 53 and the objective function was 0.1

The root mean square of the residuals (RMSR) is 0.01

The df corrected root mean square of the residuals is 0.03

The harmonic n.obs is 259 with the empirical chi square 17.64 with prob < 1  
The total n.obs was 259 with Likelihood Chi Square = 24.56 with prob < 1

Tucker Lewis Index of factoring reliability = 1.184

RMSEA index = 0 and the 90 % confidence intervals are 0 0

BIC = -269.95

Fit based upon off diagonal values = 1

## Measures of factor score adequacy

	PA1	PA2	PA3	PA4	PA5	PA6
Correlation of (regression) scores with factors	0.96	0.89	0.85	0.87	0.85	0.79
Multiple R square of scores with factors	0.92	0.79	0.72	0.75	0.72	0.62
Minimum correlation of possible factor scores	0.83	0.58	0.44	0.51	0.44	0.25
	PA7	PA8	PA9	PA10	PA11	
Correlation of (regression) scores with factors	0.80	0.71	0.71	0.67	0.64	
Multiple R square of scores with factors	0.64	0.50	0.50	0.45	0.42	
Minimum correlation of possible factor scores	0.28	0.01	0.00	-0.09	-0.17	
	PA12	PA13				

Correlation of (regression) scores with factors	0.58	0.56
Multiple R square of scores with factors	0.34	0.31
Minimum correlation of possible factor scores	-0.33	-0.38

The total variance for a particular variable will have two factors: some variance will be shared with other variables (common variance) and some variance will be specific to that measure (unique variance). Random variance is also specific to one item, but not reliably so. We can examine this most easily by examining the matrix (second screen).

The columns PA1 thru PA10 are the (uninteresting at this point) unrotated loadings. These are the loading from each factor to each variable. PA stands for “principal axis.”

Scrolling to the far right we are interested in:

**Communalities** are represented as  $h^2$ . These are the proportions of common variance present in the variables. A variable that has no specific (or random) variance would have a communality of 1.0. If a variable shares none of its variance with any other variable, its communality would be 0.0. As a point of comparison, in PCA these started as 1.0 because we extracted the same number of components as items. In PAF, because we must extract fewer factors than items, these will have unique values.

\*\*Uniquenesses\* are represented as  $u^2$ . These are the amount of unique variance for each variable. They are calculated as  $1 - h^2$  (or 1 minus the communality).

The final column, *com* represents *item complexity*. This is an indication of how well an item reflects a single construct. If it is 1.0 then the item loads only on one factor, if it is 2.0, it loads evenly on two factors, and so forth. For now, we can ignore this. *I mostly wanted to reassure you that “com” is not “communality” –  $h^2$  is communality.*

Let’s switch to the first screen of output.

**Eigenvalues** are displayed in the row called, *SS loadings* (i.e., the sum of squared loadings). They represent the variance explained by the particular linear factor. PA1 explains 4.86 units of variance (out of a possible 25; the # of potential factors). As a proportion, this is  $4.86/25 = 0.1944$  (reported in the *Proportion Var* row). We inspect the eigenvalues to see how many are  $> 1.0$  (Kaiser’s eigenvalue  $> 1$  criteria). We see there are three that meet Kaiser’s criteria and four that meet Joliffe’s criteria (eigenvalues  $> .70$ ).

4.86/25

[1] 0.1944

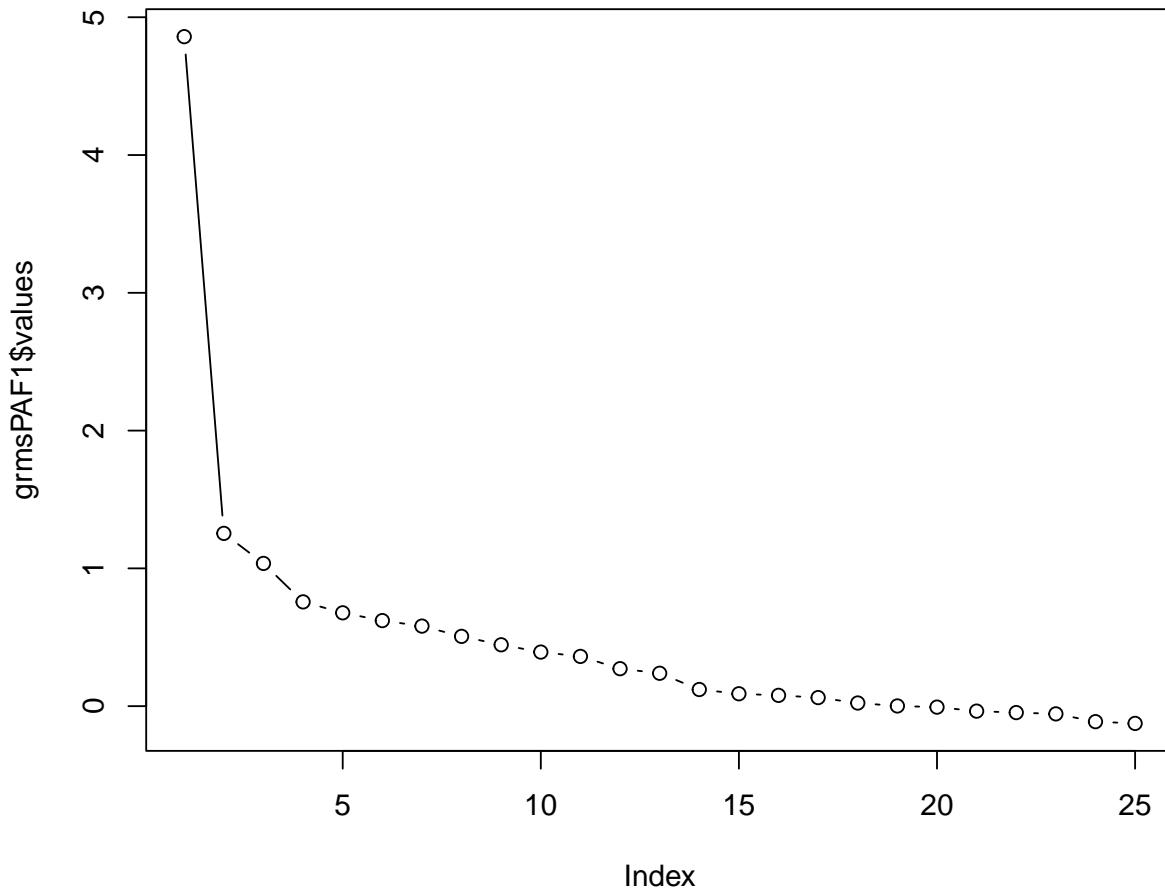
**Cumulative Var** is helpful to determine how many factors we’d like to retain to balance parsimony (few as possible) with the amount of variance we want to explain. Directly related to the eigenvalues, we can see how including each additional factor accounts for a greater proportion of variance.

**Scree plots** can help us visualize the relationship between the eigenvalues. A rule of thumb is to select the number of factors that is associated with the number of dots before the flattening of the curve. Eigenvalues are stored in the *grmsPAF1* object’s variable, “values”. We can see all the values captured by this object with the *names()* function:

```
names(grmsPAF1)
```

[1] "residual"	"dof"	"chi"
[4] "nh"	"rms"	"EPVAL"
[7] "crms"	"EBIC"	"ESABIC"
[10] "fit"	"fit.off"	"sd"
[13] "factors"	"complexity"	"n.obs"
[16] "objective"	"criteria"	"STATISTIC"
[19] "PVAL"	"Call"	"null.model"
[22] "null.dof"	"null.chisq"	"TLI"
[25] "F0"	"RMSEA"	"BIC"
[28] "SABIC"	"r.scores"	"R2"
[31] "valid"	"score.cor"	"weights"
[34] "rotation"	"hyperplane"	"communality"
[37] "communalities"	"uniquenesses"	"values"
[40] "e.values"	"loadings"	"model"
[43] "fm"	"Structure"	"communality.iterations"
[46] "method"	"scores"	"R2.scores"
[49] "r"	"np.obs"	"fn"
[52] "Vaccounted"		

```
plot(grmsPAF1$values, type = "b") #type = 'b' gives us 'both' lines and points; type = 'l' g
```



We look for the point of *inflection*. That is, where the baseline levels out into a plateau. I can see inflections after 1, 2, 3, and 4.

#### 9.5.2.1 Specifying the number of factors

Here's a snip of our location in the PAF workflow.

##### Specify and Evaluate the Solution with the Specified Number of Factors

- Rerun the PA with the determined number of factors
- Examine **communalities** individually and as an average. Criteria for acceptability range from being greater than .40 to being greater than .70
- Examine **residuals**. Determine the number and percentage that are greater than .05. A common practice is to stay below 50% larger than .05.

Figure 9.5: Image of our location in the PAF workflow

Having determined the number of factors, we must rerun the analysis with this specification. Especially when researchers may not have a clear theoretical structure that guides the process, re-

searchers may do this iteratively with varying numbers of factors. Lewis and Neville [Lewis and Neville, 2015] examined solutions with 2, 3, 4, and 5 factors (they conducted a parallel *factor* analysis; in contrast this lesson demonstrates principal axis factoring).

```
# grmsPAF2 <- psych::fa(GRMSr, nfactors=4, fm = 'pa', rotate='none')
grmsPAF2 <- psych::fa(dfGRMS, nfactors = 4, fm = "pa", rotate = "none") #can copy prior script
grmsPAF2
```

Factor Analysis using method = pa  
 Call: psych::fa(r = dfGRMS, nfactors = 4, rotate = "none", fm = "pa")  
 Standardized loadings (pattern matrix) based upon correlation matrix

	PA1	PA2	PA3	PA4	h2	u2	com
Obj1	0.48	-0.24	0.02	0.02	0.29	0.71	1.5
Obj2	0.52	-0.22	-0.02	0.06	0.32	0.68	1.4
Obj3	0.45	-0.20	0.02	0.05	0.25	0.75	1.4
Obj4	0.49	-0.27	-0.12	0.03	0.33	0.67	1.7
Obj5	0.47	-0.23	0.03	-0.05	0.28	0.72	1.5
Obj6	0.37	-0.15	0.00	0.26	0.22	0.78	2.2
Obj7	0.51	-0.09	-0.02	0.02	0.27	0.73	1.1
Obj8	0.45	-0.31	-0.16	-0.16	0.35	0.65	2.4
Obj9	0.37	-0.21	0.08	-0.02	0.19	0.81	1.7
Obj10	0.38	-0.23	-0.02	0.14	0.22	0.78	2.0
Marg1	0.58	0.36	-0.31	-0.23	0.62	0.38	2.6
Marg2	0.55	0.31	0.00	-0.09	0.41	0.59	1.6
Marg3	0.48	0.19	-0.09	-0.09	0.28	0.72	1.5
Marg4	0.46	0.11	-0.24	0.01	0.28	0.72	1.6
Marg5	0.53	0.16	-0.11	-0.11	0.33	0.67	1.4
Marg6	0.50	0.01	-0.12	0.06	0.27	0.73	1.2
Marg7	0.37	0.16	-0.13	-0.03	0.18	0.82	1.6
Str1	0.40	0.04	0.38	-0.24	0.36	0.64	2.6
Str2	0.36	-0.01	0.21	0.08	0.18	0.82	1.7
Str3	0.30	0.10	0.41	-0.08	0.28	0.72	2.1
Str4	0.26	-0.03	0.29	-0.11	0.17	0.83	2.3
Str5	0.31	0.09	0.27	0.04	0.18	0.82	2.2
Ang1	0.32	0.24	0.24	0.08	0.22	0.78	3.0
Ang2	0.31	0.31	0.01	0.49	0.43	0.57	2.5
Ang3	0.35	0.21	0.03	0.17	0.19	0.81	2.1

	PA1	PA2	PA3	PA4
SS loadings	4.67	1.03	0.83	0.57
Proportion Var	0.19	0.04	0.03	0.02
Cumulative Var	0.19	0.23	0.26	0.28
Proportion Explained	0.66	0.15	0.12	0.08
Cumulative Proportion	0.66	0.80	0.92	1.00

Mean item complexity = 1.9

Test of the hypothesis that 4 factors are sufficient.

```
df null model = 300 with the objective function = 4.89 with Chi Square = 1217.51
df of the model are 206 and the objective function was 0.77
```

The root mean square of the residuals (RMSR) is 0.04  
The df corrected root mean square of the residuals is 0.04

The harmonic n.obs is 259 with the empirical chi square 202.41 with prob < 0.56  
The total n.obs was 259 with Likelihood Chi Square = 189.19 with prob < 0.79

Tucker Lewis Index of factoring reliability = 1.027  
RMSEA index = 0 and the 90 % confidence intervals are 0 0.018  
BIC = -955.52  
Fit based upon off diagonal values = 0.97  
Measures of factor score adequacy

	PA1	PA2	PA3	PA4
Correlation of (regression) scores with factors	0.93	0.79	0.74	0.70
Multiple R square of scores with factors	0.87	0.62	0.55	0.48
Minimum correlation of possible factor scores	0.75	0.24	0.11	-0.03

Our eigenvalues/SS loadings wiggle around a bit from the initial run. With four factors, we now, cumulatively, explain 28% of the variance.

*Communality* is the proportion of common variance within a variable. Changing from 13 to 4 factors changed these values ( $h^2$ ) as well as their associated *uniquenesses* ( $u^2$ ), which are calculated as “1.0 minus the communality.”

Now we see that 29% of the variance associated with Obj1 is common/shared (the  $h^2$  value).

As a reminder of what we are doing, recall that we are looking for a more *parsimonious* explanation than 25 items on the GRMS. By respecifying a smaller number of factors, we lose some information. That is, the retained factors (now 4) cannot explain all of the variance present in the data (as we saw, it explains about 28%, cumulatively). The amount of variance explained in each variable is represented by the communalities after extraction.

We can also inspect the communalities through the lens of Kaiser’s criterion (the eigenvalue  $> 1$  criteria) to see if we think that four was a good number of factors to extract.

Kaiser’s criterion is believed to be accurate:

- when there are fewer than 30 variables (we had 25) and, after extraction, the communalities are greater than .70
  - looking at our data, none of the communalities is  $> .70$ , so, this does not support extracting four factors
- When the sample size is greater than 250 (ours was 259) and the average communality is  $> .60$ 
  - calculated below, ours was .28.

Using the *names()* function again, we see that “communality” is available for manipulation.

```
names(grmsPAF2)
```

[1] "residual"	"dof"	"chi"
[4] "nh"	"rms"	"EPVAL"
[7] "crms"	"EBIC"	"ESABIC"
[10] "fit"	"fit.off"	"sd"
[13] "factors"	"complexity"	"n.obs"
[16] "objective"	"criteria"	"STATISTIC"
[19] "PVAL"	"Call"	"null.model"
[22] "null.dof"	"null.chisq"	"TLI"
[25] "F0"	"RMSEA"	"BIC"
[28] "SABIC"	"r.scores"	"R2"
[31] "valid"	"score.cor"	"weights"
[34] "rotation"	"hyperplane"	"communality"
[37] "communalities"	"uniquenesses"	"values"
[40] "e.values"	"loadings"	"model"
[43] "fm"	"Structure"	"communality.iterations"
[46] "method"	"scores"	"R2.scores"
[49] "r"	"np.obs"	"fn"
[52] "Vaccounted"		

We can use this value to calculate their mean.

```
mean(grmsPAF2$communality)
```

```
[1] 0.2840516
```

```
# sum(grmsPAF2$communality) #
```

We see that our average communality is 0.28. These two criteria suggest that we may not have the best solution. That said (in our defense):

- We used the scree plot as a guide, and it was very clear.
- We have an adequate sample size and that was supported with the KMO.
- Are the number of factors consistent with theory? We have not yet inspected the factor loadings. This will provide us with more information.

We could do several things:

- rerun with a different number of factors (recall Lewis and Neville [2015] ran models with 2, 3, 4, and 5 factors)
- conduct more diagnostics tests
  - reproduced correlation matrix

- the difference between the reproduced correlation matrix and the correlation matrix from the data

The *factor.model()* function in *psych* produces the *reproduced correlation matrix* by using the *loadings* in our extracted object. Conceptually, this matrix is the correlations that should be produced if we did not have the raw data, but we only had the factor loadings.

The questions, though, is: How close did we get? How different is the *reproduced correlation matrix* from *GRMSmatrix* – the *R*-matrix produced from our raw data.

```
round(psych::factor.model(grmsPAF2$loadings), 3) #produces the reproduced correlation matrix
```

	Obj1	Obj2	Obj3	Obj4	Obj5	Obj6	Obj7	Obj8	Obj9	Obj10	Marg1	Marg2
Obj1	0.293	0.304	0.270	0.300	0.284	0.219	0.270	0.287	0.229	0.242	0.187	0.193
Obj2	0.304	0.319	0.282	0.317	0.291	0.239	0.286	0.294	0.232	0.256	0.215	0.212
Obj3	0.270	0.282	0.250	0.277	0.260	0.209	0.252	0.258	0.209	0.225	0.178	0.185
Obj4	0.300	0.317	0.277	0.326	0.288	0.229	0.279	0.317	0.225	0.255	0.219	0.186
Obj5	0.284	0.291	0.260	0.288	0.280	0.196	0.261	0.287	0.226	0.225	0.193	0.194
Obj6	0.219	0.239	0.209	0.229	0.196	0.225	0.207	0.169	0.161	0.210	0.097	0.131
Obj7	0.270	0.286	0.252	0.279	0.261	0.207	0.273	0.260	0.205	0.218	0.271	0.256
Obj8	0.287	0.294	0.258	0.317	0.287	0.169	0.260	0.353	0.221	0.223	0.242	0.170
Obj9	0.229	0.232	0.209	0.225	0.226	0.161	0.205	0.221	0.186	0.183	0.118	0.140
Obj10	0.242	0.256	0.225	0.255	0.225	0.210	0.218	0.223	0.183	0.217	0.113	0.126
Marg1	0.187	0.215	0.178	0.219	0.193	0.097	0.271	0.242	0.118	0.113	0.619	0.455
Marg2	0.193	0.212	0.185	0.186	0.194	0.131	0.256	0.170	0.140	0.126	0.455	0.411
Marg3	0.182	0.201	0.173	0.191	0.182	0.121	0.228	0.187	0.129	0.126	0.393	0.329
Marg4	0.195	0.221	0.186	0.226	0.185	0.155	0.234	0.212	0.127	0.158	0.378	0.287
Marg5	0.213	0.233	0.201	0.225	0.213	0.140	0.257	0.225	0.153	0.151	0.424	0.350
Marg6	0.239	0.263	0.227	0.259	0.227	0.197	0.261	0.233	0.169	0.199	0.322	0.276
Marg7	0.140	0.159	0.134	0.154	0.136	0.105	0.180	0.145	0.093	0.104	0.320	0.257
Str1	0.188	0.177	0.171	0.136	0.206	0.082	0.191	0.148	0.177	0.103	0.187	0.258
Str2	0.184	0.190	0.174	0.160	0.178	0.155	0.184	0.122	0.151	0.147	0.124	0.189
Str3	0.128	0.119	0.120	0.072	0.137	0.075	0.136	0.053	0.126	0.071	0.105	0.207
Str4	0.135	0.126	0.122	0.097	0.144	0.070	0.126	0.097	0.128	0.082	0.075	0.145
Str5	0.134	0.136	0.128	0.097	0.132	0.111	0.145	0.061	0.116	0.097	0.115	0.193
Ang1	0.101	0.110	0.103	0.066	0.098	0.101	0.138	0.016	0.083	0.070	0.176	0.242
Ang2	0.087	0.123	0.102	0.085	0.052	0.193	0.143	-0.037	0.038	0.114	0.177	0.223
Ang3	0.125	0.146	0.126	0.120	0.111	0.141	0.165	0.063	0.084	0.109	0.231	0.243
	Marg3	Marg4	Marg5	Marg6	Marg7	Str1	Str2	Str3	Str4	Str5	Ang1	Ang2
Obj1	0.182	0.195	0.213	0.239	0.140	0.188	0.184	0.128	0.135	0.134	0.101	0.087
Obj2	0.201	0.221	0.233	0.263	0.159	0.177	0.190	0.119	0.126	0.136	0.110	0.123
Obj3	0.173	0.186	0.201	0.227	0.134	0.171	0.174	0.120	0.122	0.128	0.103	0.102
Obj4	0.191	0.226	0.225	0.259	0.154	0.136	0.160	0.072	0.097	0.097	0.066	0.085
Obj5	0.182	0.185	0.213	0.227	0.136	0.206	0.178	0.137	0.144	0.132	0.098	0.052
Obj6	0.121	0.155	0.140	0.197	0.105	0.082	0.155	0.075	0.070	0.111	0.101	0.193
Obj7	0.228	0.234	0.257	0.261	0.180	0.191	0.184	0.136	0.126	0.145	0.138	0.143
Obj8	0.187	0.212	0.225	0.233	0.145	0.148	0.122	0.053	0.097	0.061	0.016	-0.037
Obj9	0.129	0.127	0.153	0.169	0.093	0.177	0.151	0.126	0.128	0.116	0.083	0.038

Obj10	0.126	0.158	0.151	0.199	0.104	0.103	0.147	0.071	0.082	0.097	0.070	0.114
Marg1	0.393	0.378	0.424	0.322	0.320	0.187	0.124	0.105	0.075	0.115	0.176	0.177
Marg2	0.329	0.287	0.350	0.276	0.257	0.258	0.189	0.207	0.145	0.193	0.242	0.223
Marg3	0.277	0.260	0.300	0.247	0.220	0.185	0.143	0.132	0.100	0.133	0.165	0.161
Marg4	0.260	0.281	0.286	0.264	0.219	0.098	0.117	0.052	0.045	0.087	0.116	0.182
Marg5	0.300	0.286	0.326	0.274	0.238	0.201	0.156	0.137	0.110	0.140	0.168	0.160
Marg6	0.247	0.264	0.274	0.271	0.203	0.142	0.160	0.097	0.085	0.123	0.136	0.189
Marg7	0.220	0.219	0.238	0.203	0.181	0.113	0.103	0.077	0.056	0.091	0.122	0.151
Str1	0.185	0.098	0.201	0.142	0.113	0.363	0.206	0.300	0.241	0.221	0.209	0.025
Str2	0.143	0.117	0.156	0.160	0.103	0.206	0.179	0.186	0.146	0.170	0.167	0.146
Str3	0.132	0.052	0.137	0.097	0.077	0.300	0.186	0.276	0.205	0.210	0.212	0.087
Str4	0.100	0.045	0.110	0.085	0.056	0.241	0.146	0.205	0.167	0.152	0.136	0.018
Str5	0.133	0.087	0.140	0.123	0.091	0.221	0.170	0.210	0.152	0.178	0.187	0.145
Ang1	0.165	0.116	0.168	0.136	0.122	0.209	0.167	0.212	0.136	0.187	0.223	0.215
Ang2	0.161	0.182	0.160	0.189	0.151	0.025	0.146	0.087	0.018	0.145	0.215	0.431
Ang3	0.187	0.180	0.196	0.186	0.155	0.120	0.142	0.123	0.073	0.140	0.181	0.256
	Ang3											
Obj1	0.125											
Obj2	0.146											
Obj3	0.126											
Obj4	0.120											
Obj5	0.111											
Obj6	0.141											
Obj7	0.165											
Obj8	0.063											
Obj9	0.084											
Obj10	0.109											
Marg1	0.231											
Marg2	0.243											
Marg3	0.187											
Marg4	0.180											
Marg5	0.196											
Marg6	0.186											
Marg7	0.155											
Str1	0.120											
Str2	0.142											
Str3	0.123											
Str4	0.073											
Str5	0.140											
Ang1	0.181											
Ang2	0.256											
Ang3	0.195											

We're not really interested in this matrix. We just need it to compare it to the *GRMSmatrix* to produce the residuals. We do that next.

**Residuals** are the difference between the reproduced (i.e., those created from our factor loadings) and *R*-matrix produced by the raw data.

If we look at the  $r_{Obj1Obj2}$  in our original correlation matrix (theoretically from the raw data [although we simulated data]), the value is 0.35. The reproduced correlation for this pair is 0.304. The difference is 0.046. The residuals table below shows 0.051 (rounding error).

```
.35 - .304
```

```
[1] 0.046
```

By using the `factor.residuals()` function we can calculate the residuals. Here we will see this difference calculated for us, for all the elements in the matrix.

```
round(psych::factor.residuals(GRMSr, grmsPAF2$loadings), 3)
```

	Obj1	Obj2	Obj3	Obj4	Obj5	Obj6	Obj7	Obj8	Obj9	Obj10
Obj1	0.707	0.051	-0.020	-0.030	-0.002	0.031	0.011	0.059	-0.076	-0.001
Obj2	0.051	0.681	0.030	-0.069	-0.023	-0.005	0.026	-0.012	0.030	-0.011
Obj3	-0.020	0.030	0.750	-0.036	0.023	0.070	-0.051	-0.009	0.000	-0.010
Obj4	-0.030	-0.069	-0.036	0.674	0.099	0.003	0.006	-0.020	0.031	0.024
Obj5	-0.002	-0.023	0.023	0.099	0.720	-0.043	-0.084	0.004	0.026	-0.021
Obj6	0.031	-0.005	0.070	0.003	-0.043	0.775	-0.008	-0.028	0.046	-0.089
Obj7	0.011	0.026	-0.051	0.006	-0.084	-0.008	0.727	0.047	-0.012	0.065
Obj8	0.059	-0.012	-0.009	-0.020	0.004	-0.028	0.047	0.647	-0.028	0.007
Obj9	-0.076	0.030	0.000	0.031	0.026	0.046	-0.012	-0.028	0.814	0.016
Obj10	-0.001	-0.011	-0.010	0.024	-0.021	-0.089	0.065	0.007	0.016	0.783
Marg1	-0.002	0.001	-0.009	0.001	-0.027	0.005	0.025	0.025	-0.017	-0.019
Marg2	0.053	-0.005	0.045	-0.009	0.003	0.007	-0.042	-0.034	-0.023	-0.003
Marg3	-0.013	0.045	0.001	-0.054	0.044	-0.069	-0.030	-0.048	0.086	0.039
Marg4	-0.007	-0.041	0.056	0.030	0.013	-0.054	0.017	0.026	-0.060	-0.036
Marg5	-0.042	-0.014	0.011	0.043	0.039	0.021	-0.026	-0.038	0.036	-0.042
Marg6	-0.061	0.006	-0.067	-0.031	-0.002	0.066	0.024	0.027	-0.019	0.057
Marg7	-0.010	0.032	0.008	0.039	-0.079	0.069	-0.024	-0.004	0.011	0.005
Str1	0.027	0.008	-0.029	-0.074	0.025	-0.016	0.055	0.018	0.014	-0.002
Str2	0.004	-0.006	0.015	0.030	-0.061	-0.003	-0.049	-0.059	0.032	0.040
Str3	-0.030	-0.031	-0.029	0.008	-0.028	0.020	0.052	-0.007	-0.010	0.028
Str4	-0.041	0.015	0.061	0.054	-0.023	0.013	-0.056	0.031	-0.075	-0.060
Str5	0.064	0.009	0.024	-0.017	0.058	0.002	0.003	-0.024	-0.051	-0.006
Ang1	-0.038	-0.042	-0.030	0.026	0.019	-0.061	0.013	0.057	0.086	-0.008
Ang2	-0.028	0.022	-0.019	-0.025	0.035	0.003	-0.012	0.010	-0.042	0.024
Ang3	0.081	-0.019	-0.015	0.020	-0.006	0.015	0.061	0.002	-0.022	-0.025
	Marg1	Marg2	Marg3	Marg4	Marg5	Marg6	Marg7	Str1	Str2	Str3
Obj1	-0.002	0.053	-0.013	-0.007	-0.042	-0.061	-0.010	0.027	0.004	-0.030
Obj2	0.001	-0.005	0.045	-0.041	-0.014	0.006	0.032	0.008	-0.006	-0.031
Obj3	-0.009	0.045	0.001	0.056	0.011	-0.067	0.008	-0.029	0.015	-0.029
Obj4	0.001	-0.009	-0.054	0.030	0.043	-0.031	0.039	-0.074	0.030	0.008
Obj5	-0.027	0.003	0.044	0.013	0.039	-0.002	-0.079	0.025	-0.061	-0.028
Obj6	0.005	0.007	-0.069	-0.054	0.021	0.066	0.069	-0.016	-0.003	0.020

Obj7	0.025	-0.042	-0.030	0.017	-0.026	0.024	-0.024	0.055	-0.049	0.052
Obj8	0.025	-0.034	-0.048	0.026	-0.038	0.027	-0.004	0.018	-0.059	-0.007
Obj9	-0.017	-0.023	0.086	-0.060	0.036	-0.019	0.011	0.014	0.032	-0.010
Obj10	-0.019	-0.003	0.039	-0.036	-0.042	0.057	0.005	-0.002	0.040	0.028
Marg1	0.381	-0.030	0.013	-0.002	-0.017	0.026	-0.011	0.003	-0.004	0.026
Marg2	-0.030	0.589	0.019	-0.057	0.053	-0.009	0.069	-0.005	-0.007	-0.027
Marg3	0.013	0.019	0.723	0.063	-0.049	0.007	-0.016	0.018	0.030	-0.036
Marg4	-0.002	-0.057	0.063	0.719	0.013	-0.006	-0.055	0.001	0.094	-0.004
Marg5	-0.017	0.053	-0.049	0.013	0.674	0.016	0.044	-0.043	-0.023	0.020
Marg6	0.026	-0.009	0.007	-0.006	0.016	0.729	-0.005	-0.012	0.024	0.050
Marg7	-0.011	0.069	-0.016	-0.055	0.044	-0.005	0.819	0.027	-0.051	-0.036
Str1	0.003	-0.005	0.018	0.001	-0.043	-0.012	0.027	0.637	0.009	0.001
Str2	-0.004	-0.007	0.030	0.094	-0.023	0.024	-0.051	0.009	0.821	0.012
Str3	0.026	-0.027	-0.036	-0.004	0.020	0.050	-0.036	0.001	0.012	0.724
Str4	0.009	-0.027	-0.021	0.018	0.031	0.048	-0.039	-0.015	0.057	0.063
Str5	-0.015	0.035	-0.010	-0.054	0.037	-0.040	0.032	0.010	-0.048	-0.029
Ang1	-0.014	-0.010	0.016	0.002	-0.053	-0.031	0.049	-0.028	-0.007	-0.013
Ang2	-0.007	-0.038	0.031	0.041	-0.018	0.024	-0.018	0.025	-0.025	-0.012
Ang3	0.048	0.036	-0.077	-0.007	0.012	-0.069	-0.069	-0.018	0.013	0.024
	Str4	Str5	Ang1	Ang2	Ang3					
Obj1	-0.041	0.064	-0.038	-0.028	0.081					
Obj2	0.015	0.009	-0.042	0.022	-0.019					
Obj3	0.061	0.024	-0.030	-0.019	-0.015					
Obj4	0.054	-0.017	0.026	-0.025	0.020					
Obj5	-0.023	0.058	0.019	0.035	-0.006					
Obj6	0.013	0.002	-0.061	0.003	0.015					
Obj7	-0.056	0.003	0.013	-0.012	0.061					
Obj8	0.031	-0.024	0.057	0.010	0.002					
Obj9	-0.075	-0.051	0.086	-0.042	-0.022					
Obj10	-0.060	-0.006	-0.008	0.024	-0.025					
Marg1	0.009	-0.015	-0.014	-0.007	0.048					
Marg2	-0.027	0.035	-0.010	-0.038	0.036					
Marg3	-0.021	-0.010	0.016	0.031	-0.077					
Marg4	0.018	-0.054	0.002	0.041	-0.007					
Marg5	0.031	0.037	-0.053	-0.018	0.012					
Marg6	0.048	-0.040	-0.031	0.024	-0.069					
Marg7	-0.039	0.032	0.049	-0.018	-0.069					
Str1	-0.015	0.010	-0.028	0.025	-0.018					
Str2	0.057	-0.048	-0.007	-0.025	0.013					
Str3	0.063	-0.029	-0.013	-0.012	0.024					
Str4	0.833	-0.034	0.017	0.013	-0.051					
Str5	-0.034	0.822	0.036	0.009	-0.034					
Ang1	0.017	0.036	0.777	0.025	0.051					
Ang2	0.013	0.009	0.025	0.569	-0.006					
Ang3	-0.051	-0.034	0.051	-0.006	0.805					

There are several strategies to evaluate this matrix.

- Compare the size of the residuals to the original correlations.
  - The worst possible model would occur if we extracted no factors, and the residuals are the size of the original correlations.
  - If the correlations were small to start with, we expect small residuals.
  - If the correlations were large to start with, the residuals will be relatively larger (this is not terribly problematic).
- Comparing residuals requires squaring them first (because residuals can be both positive and negative)
  - The sum of the squared residuals divided by the sum of the squared correlations is an estimate of model fit.
    - \* Subtracting this from 1.0 means that it ranges from 0 to 1.
    - \* Values  $> .95$  are an indication of good fit.

Analyzing the residuals means we need to extract only the upper right of the triangle them into an object. We can do this in steps.

```
grmsPAF2_resids <- psych::factor.residuals(GRMSr, grmsPAF2$loadings) #first extract the residuals
grmsPAF2_resids <- as.matrix(grmsPAF2_resids[upper.tri(grmsPAF2_resids)]) #the object has the upper triangle
head(grmsPAF2_resids)
```

```
[,1]
[1,]  0.05128890
[2,] -0.01969873
[3,]  0.03041703
[4,] -0.02971380
[5,] -0.06927144
[6,] -0.03556489
```

```
# unhashtag for a view of all the residuals grmsPAF2_resids
```

One criteria of residual analysis is to see how many residuals there are that are greater than an absolute value of 0.05. The result will be a single column with TRUE if it is  $> |0.05|$  and false if it is smaller. The sum function will tell us how many TRUE responses are in the matrix. Further, we can write script to obtain the proportion of total number of residuals.

```
large.resid <- abs(grmsPAF2_resids) > 0.05
# large.resid
sum(large.resid)
```

```
[1] 57
```

```
round(sum(large.resid)/nrow(grmsPAF2_resids), 3)
```

```
[1] 0.19
```

We learn that there are 57 residuals greater than the absolute value of 0.05. This represents 19% of the total number of residuals.

There are no hard rules about what proportion of residuals can be greater than 0.05. Field recommends that it stay below 50% [Field, 2012].

Another approach to analyzing residuals is to look at their mean. Because of the +/- valences, we need to square them (to eliminate the negative), take the average, then take the square root.

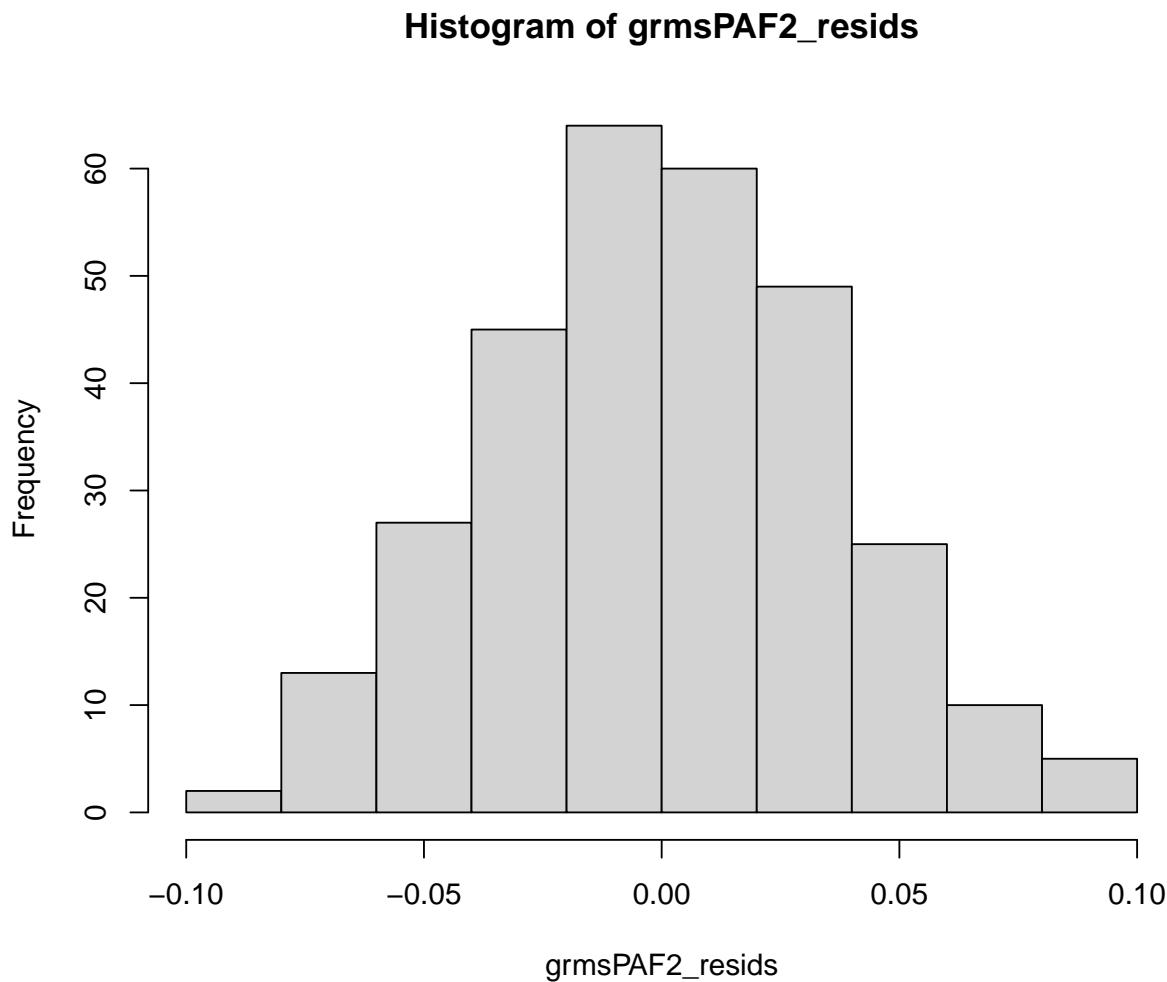
```
round(sqrt(mean(grmsPAF2_resids^2)), 3)
```

```
[1] 0.036
```

While there are no clear guidelines to interpret these, one recommendation is to consider extracting more factors if the value is higher than 0.08 [Field, 2012].

Finally, we expect our residuals to be normally distributed. A histogram can help us inspect the distribution.

```
hist(grmsPAF2_resids)
```



Not bad! It looks reasonably normal. No outliers.

### 9.5.2.2 Quick recap of how to evaluate the # of factors we extracted

- If fewer than 30 variables, the eigenvalue  $> 1$  (Kaiser's) criteria is fine, so long as communalities are all  $> .70$ .
- If sample size  $> 250$  and the average communalities are  $.6$  or greater, this is acceptable
- When  $N > 200$ , the scree plot can be used.
- Regarding residuals:
  - Fewer than 50% should have absolute values  $> 0.05$ .
  - Model fit should be  $> 0.90$ .

### 9.5.3 Factor Rotation

Here's a snip of our location in the PAF workflow.

**Factor Rotation: Orthogonal or Oblique?** Expect to run both (with varying factor structures), report only one.

**Orthogonal** – to be used when we think the factors are independent/unrelated; this approach minimizes cross-loadings

- Examine the factors loadings. Are they above .3? Do any cross-load on other factors (the highest loading should be on their theorized subscale)?
- Are the communalities greater than at least .4? (ideally higher). Consider removing items when the communalities are .2 or lower.

**Oblique** – to be used when we expect the factors to be correlated; this approach may be *true* to our overarching hypothesis (i.e., subscales are likely related to each other), but add complexity because cross-loadings are more likely

- Examine the factors loadings. Are they above .3? Do any cross-load on other factors (the highest loading should be on their theorized subscale)?
- Are the communalities greater than at least .4? (ideally higher). Consider removing items when the communalities are .2 or lower.
- The **pattern matrix** reports factor loadings and is most comparable to the orthogonal output.
- The **structure matrix** is a product of the pattern matrix and matrix containing the correlation coefficients between the factors/scales). This can serve as a useful check to the pattern matrix.

Figure 9.6: Image of our location in the PAF workflow

The original solution of a principal components or principal axis factor analysis is a set of vectors that best account for the observed covariance or correlation matrix. Each additional component or factor accounts for progressively less and less variance. The solution is efficient (yay) but difficult to interpret (boo).

Thanks to Thurstone's five rules toward a simple structure (circa 1947), interpretation of a matrix is facilitated by *rotation* (multiplying a matrix by a matrix of orthogonal vectors that preserve the communalities of each variable). Both the original matrix and the solution will be orthogonal.

*Parsimony* becomes a statistical consideration (an equation, in fact) and goal and is maximized when each variable has a 1.0 loading on one factor and the rest are zero.

Different rotation strategies emphasize different goals related to parsimony:

*Quartimax* seeks to maximize the notion of variable parsimony (each variable is associated with one factor) and permits the rotation toward a general factor (ignoring smaller factors). *Varimax* maximizes the variance of squared loadings taken over items instead of over factors and *avoids* a general factor.

Rotation improves the interpretation of the factor by maximizing the loading on each variable on one of the extracted factors while minimizing the loading on all other factors. Rotation works by changing the absolute values of the variables while keeping their differential values constant.

There are two big choices (to be made on theoretical grounds):

- Orthogonal rotation if you think that the factors are independent/unrelated.
  - varimax is the most common orthogonal rotation
- Oblique rotation if you think that the factors are related/correlated.
  - oblimin and promax are common oblique rotations

### 9.5.3.1 Orthogonal rotation

```
# grmsPAF2ORTH <- psych::fa(GRMSr, nfactors = 4, fm = 'pa', rotate =
# 'varimax')
grmsPAF2ORTH <- psych::fa(dfGRMS, nfactors = 4, fm = "pa", rotate = "varimax")
grmsPAF2ORTH
```

Factor Analysis using method = pa  
 Call: psych::fa(r = dfGRMS, nfactors = 4, rotate = "varimax", fm = "pa")  
 Standardized loadings (pattern matrix) based upon correlation matrix

	PA1	PA2	PA3	PA4	h2	u2	com
Obj1	0.49	0.14	0.16	0.04	0.29	0.71	1.4
Obj2	0.51	0.18	0.13	0.09	0.32	0.68	1.4
Obj3	0.45	0.14	0.15	0.07	0.25	0.75	1.5
Obj4	0.54	0.19	0.04	0.03	0.33	0.67	1.3
Obj5	0.47	0.15	0.19	-0.02	0.28	0.72	1.6
Obj6	0.39	0.05	0.06	0.26	0.22	0.78	1.8
Obj7	0.41	0.27	0.16	0.10	0.27	0.73	2.2
Obj8	0.52	0.23	0.03	-0.18	0.35	0.65	1.7
Obj9	0.38	0.07	0.19	-0.01	0.19	0.81	1.5
Obj10	0.44	0.06	0.06	0.12	0.22	0.78	1.2
Marg1	0.13	0.77	0.08	0.02	0.62	0.38	1.1
Marg2	0.13	0.53	0.30	0.15	0.41	0.59	1.9
Marg3	0.18	0.46	0.16	0.08	0.28	0.72	1.6
Marg4	0.26	0.45	-0.01	0.13	0.28	0.72	1.8
Marg5	0.23	0.49	0.16	0.07	0.33	0.67	1.7
Marg6	0.35	0.35	0.08	0.16	0.27	0.73	2.5
Marg7	0.14	0.38	0.07	0.10	0.18	0.82	1.5
Str1	0.16	0.16	0.55	-0.08	0.36	0.64	1.4
Str2	0.24	0.09	0.30	0.16	0.18	0.82	2.7
Str3	0.07	0.07	0.51	0.06	0.28	0.72	1.1
Str4	0.14	0.04	0.38	-0.03	0.17	0.83	1.3
Str5	0.11	0.09	0.36	0.16	0.18	0.82	1.8
Ang1	0.02	0.18	0.35	0.25	0.22	0.78	2.4
Ang2	0.05	0.20	0.06	0.62	0.43	0.57	1.2
Ang3	0.10	0.26	0.15	0.31	0.19	0.81	2.7

	PA1	PA2	PA3	PA4
SS loadings	2.60	2.26	1.41	0.83
Proportion Var	0.10	0.09	0.06	0.03
Cumulative Var	0.10	0.19	0.25	0.28
Proportion Explained	0.37	0.32	0.20	0.12
Cumulative Proportion	0.37	0.68	0.88	1.00

Mean item complexity = 1.7

Test of the hypothesis that 4 factors are sufficient.

df null model = 300 with the objective function = 4.89 with Chi Square = 1217.51  
 df of the model are 206 and the objective function was 0.77

The root mean square of the residuals (RMSR) is 0.04

The df corrected root mean square of the residuals is 0.04

The harmonic n.obs is 259 with the empirical chi square 202.41 with prob < 0.56  
The total n.obs was 259 with Likelihood Chi Square = 189.19 with prob < 0.79

Tucker Lewis Index of factoring reliability = 1.027

RMSEA index = 0 and the 90 % confidence intervals are 0 0.018

BIC = -955.52

Fit based upon off diagonal values = 0.97

Measures of factor score adequacy

	PA1	PA2	PA3	PA4
Correlation of (regression) scores with factors	0.84	0.85	0.76	0.72
Multiple R square of scores with factors	0.71	0.72	0.58	0.52
Minimum correlation of possible factor scores	0.42	0.45	0.17	0.03

Essentially, we have the same information as before, except that loadings are calculated after rotation (which adjusts the absolute values of the factor loadings while keeping their differential values constant). Our communality and uniqueness values remain the same. The eigenvalues (SS loadings) should even out, but the proportion of variance explained and cumulative variance (28%) will remain the same.

The *print.psych()* function facilitates interpretation and prioritizes the information about which we care most:

- *cut* displays loadings above .3, this allows us to see
  - if some items load on no factors
  - if some items have cross-loadings (and their relative weights)
- *sort* reorders the loadings to make it clearer (considering ties, to the best of its ability) to which factor/scale it belongs

```
grmsPAF2_table <- psych::print.psych(grmsPAF20RTH, cut = 0.3, sort = TRUE)
```

Factor Analysis using method = pa

Call: psych::fa(r = dfGRMS, nfactors = 4, rotate = "varimax", fm = "pa")

Standardized loadings (pattern matrix) based upon correlation matrix

item	PA1	PA2	PA3	PA4	h2	u2	com
Obj4	4 0.54				0.33	0.67	1.3
Obj8	8 0.52				0.35	0.65	1.7
Obj2	2 0.51				0.32	0.68	1.4
Obj1	1 0.49				0.29	0.71	1.4
Obj5	5 0.47				0.28	0.72	1.6
Obj3	3 0.45				0.25	0.75	1.5
Obj10	10 0.44				0.22	0.78	1.2
Obj7	7 0.41				0.27	0.73	2.2

Obj6	6	0.39		0.22	0.78	1.8
Obj9	9	0.38		0.19	0.81	1.5
Marg1	11		0.77	0.62	0.38	1.1
Marg2	12		0.53	0.41	0.59	1.9
Marg5	15		0.49	0.33	0.67	1.7
Marg3	13		0.46	0.28	0.72	1.6
Marg4	14		0.45	0.28	0.72	1.8
Marg7	17		0.38	0.18	0.82	1.5
Marg6	16	0.35	0.35	0.27	0.73	2.5
Str1	18		0.55	0.36	0.64	1.4
Str3	20		0.51	0.28	0.72	1.1
Str4	21		0.38	0.17	0.83	1.3
Str5	22		0.36	0.18	0.82	1.8
Ang1	23		0.35	0.22	0.78	2.4
Str2	19		0.30	0.18	0.82	2.7
Ang2	24			0.62	0.43	0.57
Ang3	25			0.31	0.19	0.81

	PA1	PA2	PA3	PA4
SS loadings	2.60	2.26	1.41	0.83
Proportion Var	0.10	0.09	0.06	0.03
Cumulative Var	0.10	0.19	0.25	0.28
Proportion Explained	0.37	0.32	0.20	0.12
Cumulative Proportion	0.37	0.68	0.88	1.00

Mean item complexity = 1.7

Test of the hypothesis that 4 factors are sufficient.

df null model = 300 with the objective function = 4.89 with Chi Square = 1217.51  
 df of the model are 206 and the objective function was 0.77

The root mean square of the residuals (RMSR) is 0.04

The df corrected root mean square of the residuals is 0.04

The harmonic n.obs is 259 with the empirical chi square 202.41 with prob < 0.56  
 The total n.obs was 259 with Likelihood Chi Square = 189.19 with prob < 0.79

Tucker Lewis Index of factoring reliability = 1.027

RMSEA index = 0 and the 90 % confidence intervals are 0 0.018

BIC = -955.52

Fit based upon off diagonal values = 0.97

Measures of factor score adequacy

	PA1	PA2	PA3	PA4
Correlation of (regression) scores with factors	0.84	0.85	0.76	0.72
Multiple R square of scores with factors	0.71	0.72	0.58	0.52
Minimum correlation of possible factor scores	0.42	0.45	0.17	0.03

In the unrotated solution, most variables loaded on the first factor. After rotation, there are four

clear factors/scales. Further, there is clear (or at least reasonable) factor/scale membership for each item and few cross-loadings. As with the PAC in the previous lesson, Ang1 is not clearly loading on the Angry scale.

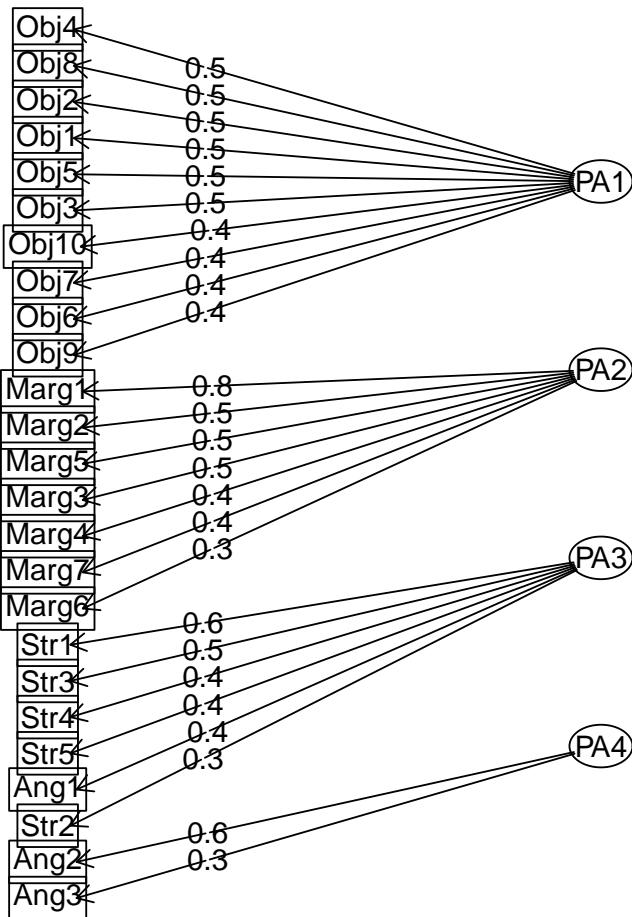
If this were a new scale and we had not yet established ideas for subscales, the next step would be to look back at the items, themselves, and try to name the scales/factors. If our scale construction included a priori/planned subscales, we would hope the items would where they were hypothesized to do so. As we noted with the Ang1 item, our simulated data nearly replicated the item membership onto the four scales that Lewis and Neville [[Lewis and Neville, 2015](#)] reported in the article.

- Assumptions of Beauty and Sexual Objectification
- Silenced and Marginalized
- Strong Woman Stereotype
- Angry Woman Stereotype

We can also create a figure of the result. Note the direction of the arrows from the factor (latent variable) to the items in PAF – in PCA the arrows went from item to component.

```
psych::fa.diagram(grmsPAF2ORTH)
```

## Factor Analysis



We can extract the factor loadings and write them to a table. This can be useful in preparing an APA style table for a manuscript or presentation.

```
# names(grmsPAF2ORTH)
pafORTH_table <- round(grmsPAF2ORTH$loadings, 3)
write.table(pafORTH_table, file = "pafORTH_table.csv", sep = ",",
           col.names = TRUE,
           row.names = FALSE)
pafORTH_table
```

Loadings:

	PA1	PA2	PA3	PA4
Obj1	0.495	0.144	0.160	
Obj2	0.513	0.179	0.131	
Obj3	0.452	0.140	0.147	
Obj4	0.536	0.191		
Obj5	0.469	0.154	0.189	

```

Obj6  0.391          0.257
Obj7  0.408  0.265  0.164
Obj8  0.517  0.232          -0.177
Obj9  0.382          0.187
Obj10 0.442          0.122
Marg1 0.126  0.772
Marg2 0.126  0.534  0.296  0.151
Marg3 0.175  0.462  0.161
Marg4 0.256  0.446          0.130
Marg5 0.230  0.493  0.162
Marg6 0.345  0.348          0.157
Marg7 0.142  0.381          0.104
Str1  0.160  0.161  0.553
Str2  0.236          0.301  0.160
Str3
Str4  0.141          0.381
Str5  0.115          0.362  0.161
Ang1
Ang2
Ang3  0.104  0.258  0.154  0.307

          PA1    PA2    PA3    PA4
SS loadings   2.600  2.257  1.414  0.830
Proportion Var 0.104  0.090  0.057  0.033
Cumulative Var 0.104  0.194  0.251  0.284

```

### 9.5.3.2 Oblique rotation

Whereas the orthogonal rotation sought to maximize the independence/unrelatedness of the factors, an oblique rotation will allow them to be correlated. Researchers often explore both solutions but only report one.

```

# grmsPAF2obl <- psych::fa(GRMSr, nfactors = 4, fm = "pa", rotate =
# 'oblimin')
grmsPAF2obl <- psych::fa(dfGRMS, nfactors = 4, fm = "pa", rotate = "oblimin")

```

Loading required namespace: GPArotation

```
grmsPAF2obl
```

```

Factor Analysis using method = pa
Call: psych::fa(r = dfGRMS, nfactors = 4, rotate = "oblimin", fm = "pa")
Standardized loadings (pattern matrix) based upon correlation matrix
          PA1    PA2    PA3    PA4    h2    u2 com
Obj1    0.51   0.00   0.08   0.00  0.29  0.71  1.1
Obj2    0.53   0.02   0.04   0.05  0.32  0.68  1.0

```

Obj3	0.46	0.00	0.07	0.04	0.25	0.75	1.1
Obj4	0.56	0.05	-0.05	0.00	0.33	0.67	1.0
Obj5	0.47	0.02	0.12	-0.06	0.28	0.72	1.2
Obj6	0.43	-0.11	-0.02	0.25	0.22	0.78	1.8
Obj7	0.38	0.14	0.08	0.06	0.27	0.73	1.4
Obj8	0.53	0.15	-0.05	-0.22	0.35	0.65	1.5
Obj9	0.39	-0.05	0.14	-0.04	0.19	0.81	1.3
Obj10	0.48	-0.08	-0.02	0.10	0.22	0.78	1.2
Marg1	-0.03	0.81	-0.03	-0.02	0.62	0.38	1.0
Marg2	-0.01	0.49	0.22	0.12	0.41	0.59	1.5
Marg3	0.08	0.43	0.09	0.05	0.28	0.72	1.2
Marg4	0.20	0.40	-0.11	0.11	0.28	0.72	1.8
Marg5	0.14	0.45	0.08	0.03	0.33	0.67	1.3
Marg6	0.31	0.25	-0.02	0.13	0.27	0.73	2.3
Marg7	0.07	0.36	0.00	0.08	0.18	0.82	1.2
Str1	0.07	0.08	0.56	-0.12	0.36	0.64	1.2
Str2	0.21	-0.04	0.27	0.14	0.18	0.82	2.5
Str3	-0.01	-0.02	0.53	0.04	0.28	0.72	1.0
Str4	0.10	-0.04	0.39	-0.06	0.17	0.83	1.2
Str5	0.06	-0.01	0.35	0.14	0.18	0.82	1.4
Ang1	-0.07	0.10	0.33	0.24	0.22	0.78	2.1
Ang2	0.01	0.07	-0.02	0.64	0.43	0.57	1.0
Ang3	0.04	0.18	0.09	0.30	0.19	0.81	1.9

	PA1	PA2	PA3	PA4
SS loadings	2.79	2.03	1.37	0.91
Proportion Var	0.11	0.08	0.05	0.04
Cumulative Var	0.11	0.19	0.25	0.28
Proportion Explained	0.39	0.29	0.19	0.13
Cumulative Proportion	0.39	0.68	0.87	1.00

With factor correlations of

	PA1	PA2	PA3	PA4
PA1	1.00	0.46	0.34	0.17
PA2	0.46	1.00	0.32	0.27
PA3	0.34	0.32	1.00	0.19
PA4	0.17	0.27	0.19	1.00

Mean item complexity = 1.4

Test of the hypothesis that 4 factors are sufficient.

df null model = 300 with the objective function = 4.89 with Chi Square = 1217.51  
 df of the model are 206 and the objective function was 0.77

The root mean square of the residuals (RMSR) is 0.04

The df corrected root mean square of the residuals is 0.04

The harmonic n.obs is 259 with the empirical chi square 202.41 with prob < 0.56

The total n.obs was 259 with Likelihood Chi Square = 189.19 with prob < 0.79

Tucker Lewis Index of factoring reliability = 1.027

RMSEA index = 0 and the 90 % confidence intervals are 0 0.018

BIC = -955.52

Fit based upon off diagonal values = 0.97

Measures of factor score adequacy

	PA1	PA2	PA3	PA4
Correlation of (regression) scores with factors	0.89	0.89	0.80	0.75
Multiple R square of scores with factors	0.79	0.79	0.65	0.57
Minimum correlation of possible factor scores	0.59	0.58	0.30	0.14

We can make it a little easier to interpret by removing all factor loadings below .30.

```
psych::print.psych(grmsPAF2obl, cut = 0.3, sort = TRUE)
```

Factor Analysis using method = pa

Call: psych::fa(r = dfGRMS, nfactors = 4, rotate = "oblimin", fm = "pa")

Standardized loadings (pattern matrix) based upon correlation matrix

item	PA1	PA2	PA3	PA4	h2	u2	com
Obj4	4	0.56			0.33	0.67	1.0
Obj8	8	0.53			0.35	0.65	1.5
Obj2	2	0.53			0.32	0.68	1.0
Obj1	1	0.51			0.29	0.71	1.1
Obj10	10	0.48			0.22	0.78	1.2
Obj5	5	0.47			0.28	0.72	1.2
Obj3	3	0.46			0.25	0.75	1.1
Obj6	6	0.43			0.22	0.78	1.8
Obj9	9	0.39			0.19	0.81	1.3
Obj7	7	0.38			0.27	0.73	1.4
Marg6	16	0.31			0.27	0.73	2.3
Marg1	11		0.81		0.62	0.38	1.0
Marg2	12		0.49		0.41	0.59	1.5
Marg5	15		0.45		0.33	0.67	1.3
Marg3	13		0.43		0.28	0.72	1.2
Marg4	14		0.40		0.28	0.72	1.8
Marg7	17		0.36		0.18	0.82	1.2
Str1	18		0.56		0.36	0.64	1.2
Str3	20		0.53		0.28	0.72	1.0
Str4	21		0.39		0.17	0.83	1.2
Str5	22		0.35		0.18	0.82	1.4
Ang1	23		0.33		0.22	0.78	2.1
Str2	19				0.18	0.82	2.5
Ang2	24			0.64	0.43	0.57	1.0
Ang3	25				0.19	0.81	1.9

PA1 PA2 PA3 PA4

SS loadings	2.79	2.03	1.37	0.91
Proportion Var	0.11	0.08	0.05	0.04
Cumulative Var	0.11	0.19	0.25	0.28
Proportion Explained	0.39	0.29	0.19	0.13
Cumulative Proportion	0.39	0.68	0.87	1.00

With factor correlations of  
PA1 PA2 PA3 PA4  
PA1 1.00 0.46 0.34 0.17  
PA2 0.46 1.00 0.32 0.27  
PA3 0.34 0.32 1.00 0.19  
PA4 0.17 0.27 0.19 1.00

Mean item complexity = 1.4

Test of the hypothesis that 4 factors are sufficient.

df null model = 300 with the objective function = 4.89 with Chi Square = 1217.51  
df of the model are 206 and the objective function was 0.77

The root mean square of the residuals (RMSR) is 0.04

The df corrected root mean square of the residuals is 0.04

The harmonic n.obs is 259 with the empirical chi square 202.41 with prob < 0.56  
The total n.obs was 259 with Likelihood Chi Square = 189.19 with prob < 0.79

Tucker Lewis Index of factoring reliability = 1.027

RMSEA index = 0 and the 90 % confidence intervals are 0 0.018

BIC = -955.52

Fit based upon off diagonal values = 0.97

Measures of factor score adequacy

	PA1	PA2	PA3	PA4
Correlation of (regression) scores with factors	0.89	0.89	0.80	0.75
Multiple R square of scores with factors	0.79	0.79	0.65	0.57
Minimum correlation of possible factor scores	0.59	0.58	0.30	0.14

In this rotation, the Angry scale falls apart. As before, Ang1 is loading onto Strong. Additionally, Ang3 has factor loadings that fall below .30, and therefore do not appear in this table. Additionally, because our specification included “sort=TRUE”, the relative weights wiggled around and so the items are listed in a different order than in the orthogonal rotation.

The oblique rotation allows us to see the correlation between the factors/scales. This was not available in the orthogonal rotation because the assumption of the orthogonal/varimax rotation is that the scales/factors are uncorrelated; hence in the analysis they were fixed to 0.0. The correlations from our simulated data range from .19 to .46.

Of course there is always a little complexity. In oblique rotations, there is a distinction between the *pattern* matrix (which reports factor loadings and is comparable to the matrix we interpreted for the orthogonal rotation) and the *structure* matrix (takes into account the relationship between the factors/scales – it is a product of the pattern matrix and the matrix containing the correlation

coefficients between the factors/scales). Most interpret the pattern matrix because it is simpler; however, it could be that values in the pattern matrix are suppressed because of relations between the factors. Therefore, the structure matrix can be a useful check and some editors will request it.

Obtaining the structure matrix requires two steps. First, multiply the factor loadings with the phi matrix.

```
grmsPAF2obl$loadings %*% grmsPAF2obl$Phi
```

	PA1	PA2	PA3	PA4
Obj1	0.5351558	0.2587471	0.2578729	0.10672133
Obj2	0.5596959	0.2939415	0.2397101	0.15920737
Obj3	0.4931574	0.2456837	0.2386994	0.13178696
Obj4	0.5676937	0.2946510	0.1536862	0.09619880
Obj5	0.5143469	0.2639663	0.2797209	0.05320390
Obj6	0.4083011	0.1433380	0.1363673	0.28476763
Obj7	0.4891359	0.3627793	0.2697523	0.18376868
Obj8	0.5462750	0.3181931	0.1337799	-0.09863944
Obj9	0.4098436	0.1673842	0.2524221	0.04306615
Obj10	0.4526040	0.1614974	0.1403757	0.15874466
Marg1	0.3385023	0.7854134	0.2197285	0.18936979
Marg2	0.3164379	0.5904981	0.3994360	0.29125182
Marg3	0.3190254	0.5084002	0.2613340	0.19797897
Marg4	0.3686089	0.4872840	0.1082789	0.22714662
Marg5	0.3780702	0.5477661	0.2747713	0.19177240
Marg6	0.4417130	0.4228182	0.1911285	0.24703507
Marg7	0.2525475	0.4115888	0.1520707	0.19123240
Str1	0.2752766	0.2583504	0.5843472	0.02598261
Str2	0.3031342	0.1787399	0.3515298	0.21686734
Str3	0.1645811	0.1543190	0.5235449	0.13423477
Str4	0.1999438	0.1116093	0.3967457	0.02531912
Str5	0.1976930	0.1656974	0.3929920	0.21880162
Ang1	0.1344340	0.2417716	0.3896315	0.32188168
Ang2	0.1467619	0.2423578	0.1284210	0.65279494
Ang3	0.2088530	0.3073000	0.2225652	0.37115411

Next, use Field's [2012] function to produce the matrix.

```
# Field's function to produce the structure matrix
factor.structure <- function(fa, cut = 0.2, decimals = 2) {
  structure.matrix <- psych::fa.sort(fa$loadings %*% fa$Phi)
  structure.matrix <- data.frame(ifelse(abs(structure.matrix) < cut,
                                         "", round(structure.matrix, decimals)))
  return(structure.matrix)
}

factor.structure(grmsPAF2obl, cut = 0.3)
```

	PA1	PA2	PA3	PA4
Obj4	0.57			
Obj2	0.56			
Obj8	0.55	0.32		
Obj1	0.54			
Obj5	0.51			
Obj3	0.49			
Obj7	0.49	0.36		
Obj10	0.45			
Marg6	0.44	0.42		
Obj9	0.41			
Obj6	0.41			
Marg1	0.34	0.79		
Marg2	0.32	0.59	0.4	
Marg5	0.38	0.55		
Marg3	0.32	0.51		
Marg4	0.37	0.49		
Marg7		0.41		
Str1		0.58		
Str3		0.52		
Str4		0.4		
Str5		0.39		
Ang1		0.39	0.32	
Str2	0.3	0.35		
Ang2			0.65	
Ang3	0.31		0.37	

Here we see some instability. Marg6 had cross-loadings with two scales and “hopped” membership onto Objectification. All three Ang items are showing factor loadings on their own scale. However, Ang1 is still loading on Strong and Ang3 has some cross-loading.

#### 9.5.4 Factor Scores

Factor *scores* (PA scores) can be created for each case (row) on each factor (column). These can be used to assess the relative standing of one person on the construct/variable to another. We can also use them in regression (in place of means or sums) when groups of predictors correlate so highly that there is multicollinearity.

Computation involves multiplying an individual’s item-level response by the factor loadings we obtained through the PAF process. The results will be one score per factor for each row/case.

```
# in all of this, don't forget to be specifying the dataset that has
# the reverse-coded item replaced
grmsPAF2obl <- psych::fa(dfGRMS, nfactors = 4, fm = "pa", rotate = "oblimin",
  scores = TRUE)
head(grmsPAF2obl$scores, 10) #shows us only the first 10 (of N = 2571)
```

```

          PA1        PA2        PA3        PA4
[1,] -0.8113951 -1.2471438  0.12524275 -0.5260185
[2,]  0.2984396 -0.7314827  0.50376631  0.2511270
[3,]  0.3834222  0.4035418  0.18143350  1.0737292
[4,] -0.9982416 -1.1583980 -0.01836819 -1.4226140
[5,] -0.1985534  0.4507445 -0.78705628  0.1813746
[6,] -0.4233586  0.3917061 -0.17730679  0.7556499
[7,]  0.2528621  0.4465398 -0.78836577 -0.4751612
[8,] -1.3823984 -0.5908492  0.36410712  0.5085523
[9,] -0.5534479 -1.2460939 -0.65935346  0.4168911
[10,]  0.1212895 -0.4771782 -0.60054957  0.1698606

```

```
dfGRMS <- cbind(dfGRMS, grmsPAF2obl$scores) #adds them to our raw dataset
```

To bring this full circle, we can see the correlation of the factor scores; the pattern maps onto what we saw previously in the correlations between factors in the oblique rotation.

```
psych::corr.test(dfGRMS[c("PA1", "PA2", "PA3", "PA4")])
```

```

Call:psych::corr.test(x = dfGRMS[c("PA1", "PA2", "PA3", "PA4")])
Correlation matrix
  PA1  PA2  PA3  PA4
PA1  1.00  0.58  0.48  0.28
PA2  0.58  1.00  0.45  0.41
PA3  0.48  0.45  1.00  0.33
PA4  0.28  0.41  0.33  1.00
Sample Size
[1] 259
Probability values (Entries above the diagonal are adjusted for multiple tests.)
  PA1  PA2  PA3  PA4
PA1  0    0    0    0
PA2  0    0    0    0
PA3  0    0    0    0
PA4  0    0    0    0

```

To see confidence intervals of the correlations, print with the short=FALSE option

We can extract the factor loadings and write them to a table. This can be useful in preparing an APA style table for a manuscript or presentation.

```

# names(grmsPAF2obl)
pafOBL_table <- round(grmsPAF2obl$loadings, 3)
write.table(pafOBL_table, file = "pafOBL_table.csv", sep = ",", col.names = TRUE,
            row.names = FALSE)
pafOBL_table

```

## Loadings:

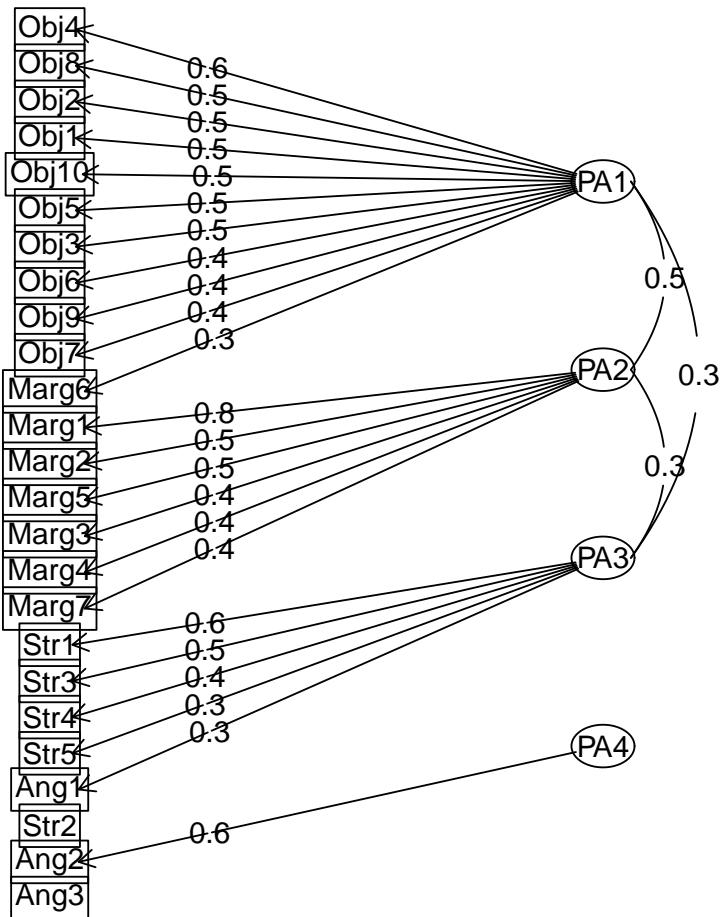
	PA1	PA2	PA3	PA4
Obj1	0.508			
Obj2	0.526			
Obj3	0.463			
Obj4	0.563			
Obj5	0.473		0.123	
Obj6	0.426	-0.114		0.246
Obj7	0.385	0.141		
Obj8	0.535	0.146		-0.220
Obj9	0.391		0.142	
Obj10	0.479			0.102
Marg1		0.811		
Marg2		0.492	0.223	0.117
Marg3		0.429		
Marg4	0.202	0.400	-0.108	0.106
Marg5	0.136	0.451		
Marg6	0.310	0.250		0.130
Marg7		0.356		
Str1			0.558	-0.116
Str2	0.206		0.266	0.140
Str3			0.526	
Str4			0.388	
Str5			0.348	0.144
Ang1		0.103	0.334	0.241
Ang2				0.635
Ang3		0.178		0.298

	PA1	PA2	PA3	PA4
SS loadings	2.523	1.762	1.190	0.794
Proportion Var	0.101	0.070	0.048	0.032
Cumulative Var	0.101	0.171	0.219	0.251

We can also obtain a figure of this PAF with oblique rotation.

```
psych::fa.diagram(grmsPAF2obl)
```

## Factor Analysis



## 9.6 APA Style Results

### Results

The dimensionality of the 25 items from the Gendered Racial Microaggressions Scale for Black Women was analyzed using principal axis factoring. First, data screening were conducted to determine the suitability of the data for principal axis factoring. The Kaiser-Meyer-Olkin measure of sampling adequacy (KMO; Kaiser, 1970) represents the ratio of the squared correlation between variables to the squared partial correlation between variables. KMO ranges from 0.00 to 1.00; values closer to 1.00 indicate that the patterns of correlations are relatively compact, and that factor analysis should yield distinct and reliable factors (Field, 2012). In our dataset, the KMO value was .85, indicating acceptable sampling adequacy. The Bartlett's Test of Sphericity examines whether the population correlation matrix resembles an identity matrix (Field, 2012). When the  $p$  value for the Bartlett's test is  $< .05$ , we are fairly certain we have clusters of correlated variables. In our dataset,  $\chi^2(300) = 1217.508, p < .001$ , indicating the

correlations between items are sufficiently large enough for principal axis factoring. The determinant of the correlation matrix alerts us to any issues of multicollinearity or singularity and should be larger than 0.00001. Our determinant was 0.0075, supporting the suitability of our data for analysis.

Four criteria were used to determine the number of factors to rotate: a priori theory, the scree test, the Eigenvalue-greater-than-one criteria, and the interpretability of the solution. Kaiser's eigenvalue-greater-than-one criteria suggested two factors, and, in combination explained 28% of the variance. The scree plot showed an inflexion that justified retaining between one and four factors. A priori theory based on Lewis and Neville's [2015] psychometric evaluation, suggested four factors. Based on the convergence of these decisions, four factors were extracted. We investigated each with orthogonal (varimax) and oblique (oblimin) procedures. Given the correspondence of the orthogonal solution with the original research, we selected this as our final model.

The rotated solution, as shown in Table 1 and Figure 1, yielded four interpretable factors, each listed with the proportion of variance accounted for: assumptions of beauty and sexual objectification (10%), silenced and marginalized (9%), strong woman stereotype (6%), and angry woman stereotype (3%).

Regarding the Table 1, I would include a table with all the values, bolding those with factor membership. This will be easy because we exported all those values to a .csv file.

### 9.6.1 Comparing FA and PCA

- FA derives a mathematical solution from which factors are estimated.
  - Only FA can estimate underlying factors, but it relies on the various assumptions to be met.
- PCA decomposes the original data into a set of linear variates.
  - This limits its concern to establishing which linear components exist within the data and how a particular variable might contribute to that component.
- Generally, FA and PCA result in similar solutions.
  - When there are 30 or more variables and communalities are  $> .7$  for all variables, different solutions are unlikely (Stevens, 2002).
  - When there are  $< 20$  variables and low communalities ( $< .4$ ) different solutions are likely to emerge.
  - Both are inferential statistics.
- Critics of PCA suggest
  - “At best it is a common factor analysis with some error added and at worst an unrecognizable hodgepodge of things from which nothing can be determined” (Cliff, 1987, p. 349).
  - PCA should never be described as FA and the resulting components should not be treated as reverently as true, latent variable, *factors*.

- To most of us (i.e., scientist-practitioners), the difference is largely from the algorithm used to derive the solutions. This is true for Field [Field, 2012] also, who uses the terms interchangeably. My take: use whichever you like, just be precise in the language describing what you did.

## 9.7 Going Back to the Future: What, then, is Omega?

Now that we've had an introduction to factor analysis, let's revisit the  $\omega$  grouping of reliability estimates. In the context of *psychometrics*, it may be useful to think of factors as scales/subscales where  $g$  refers to the amount of variance in the *general* factor (or total scale score) and subscales to be items that have something in common that is separate from what is  $g$ .

Model-based estimates examine the correlations or covariances of the items and decompose the test variance into that which is:

- common to all items (**g**, a general factor),
- specific to some items (**f**, orthogonal group factors), and
- unique to each item (confounding **s** specific, and **e** error variance)

In the *psych* package

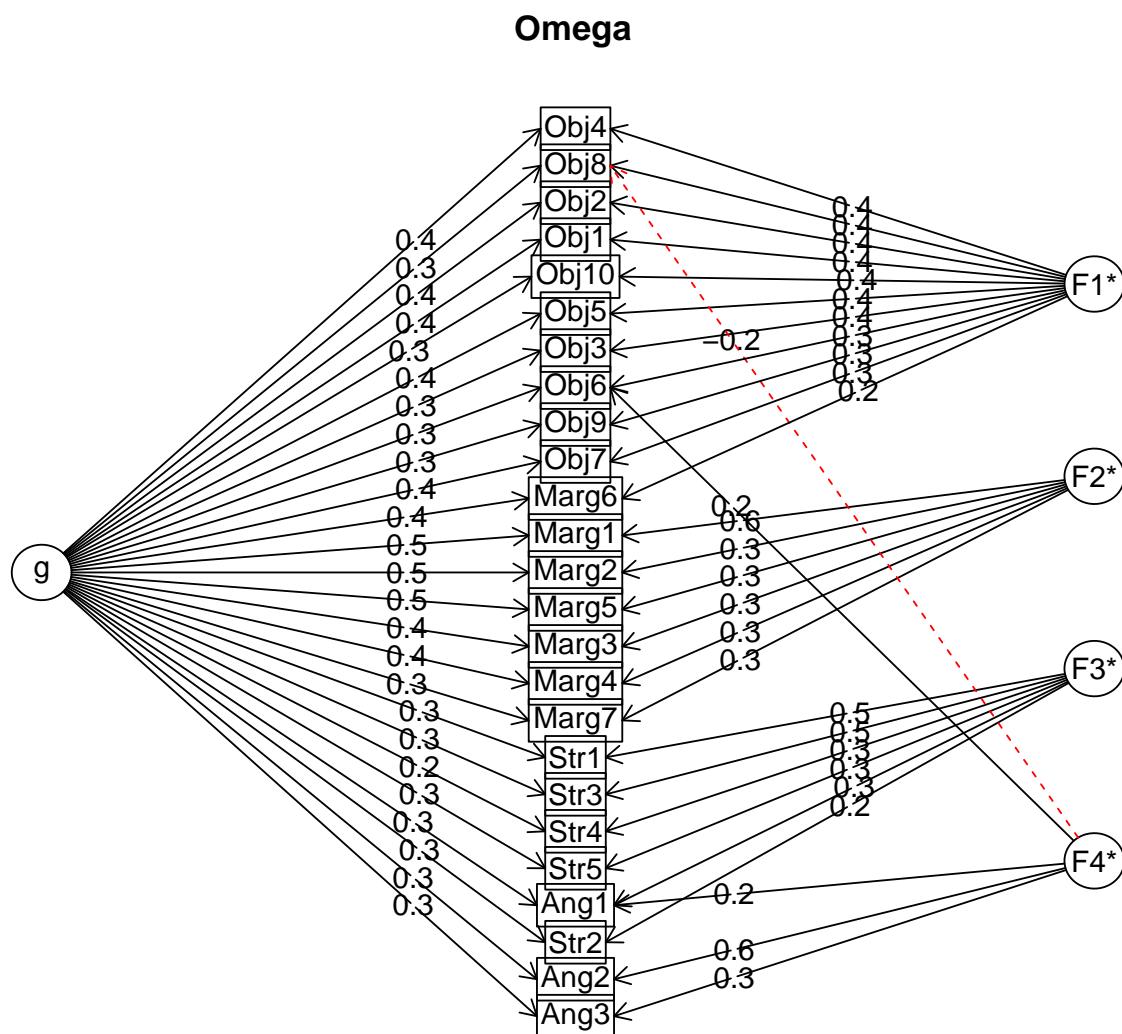
- $\omega_t$  represents the total reliability of the test ( $\omega_t$ ).
  - In the *psych* package, this is calculated from a bifactor model where there is one general  $g$  factor (i.e., each item loads on the single general factor), one or more group factors ( $f$ ), and an item-specific factor ( $s$ ).
- $\omega_h$  extracts a higher order factor from the correlation matrix of lower-level factors, then applies the Schmid and Leiman (1957) transformation to find the general loadings on the original items. Stated another way, it is a measure of the general factor saturation ( $g$ ; the amount of variance attributable to one common factor). The subscript “h” acknowledges the hierarchical nature of the approach.
  - the  $\omega_h$  approach is exploratory and defined if there are three or more group factors (with only two group factors, the default is to assume they are equally important, hence the factor loadings of those subscales will be equal)
  - Najera Catalan [Najera Catalan, 2019] suggests that  $\omega_h$  is the best measure of reliability when dealing with multiple dimensions.
- $\omega_g$  is an estimate that uses a bifactor solution via the SEM package *lavaan* and tends to be a larger (because it forces all the cross-loadings of lower-level factors to be 0)
  - $\omega_g$  is confirmatory, requiring the specification of which variables load on each group factor
- *psych::omegaSem()* reports both EFA and CFA solutions
  - We will use the *psych::omegaSem()* function.

Note that in our specification, we indicate there are two factors. We do not tell it (anywhere!) what items belong to what factors (think, *subsubscales*). One test will be to see if the items align with their respective factors.

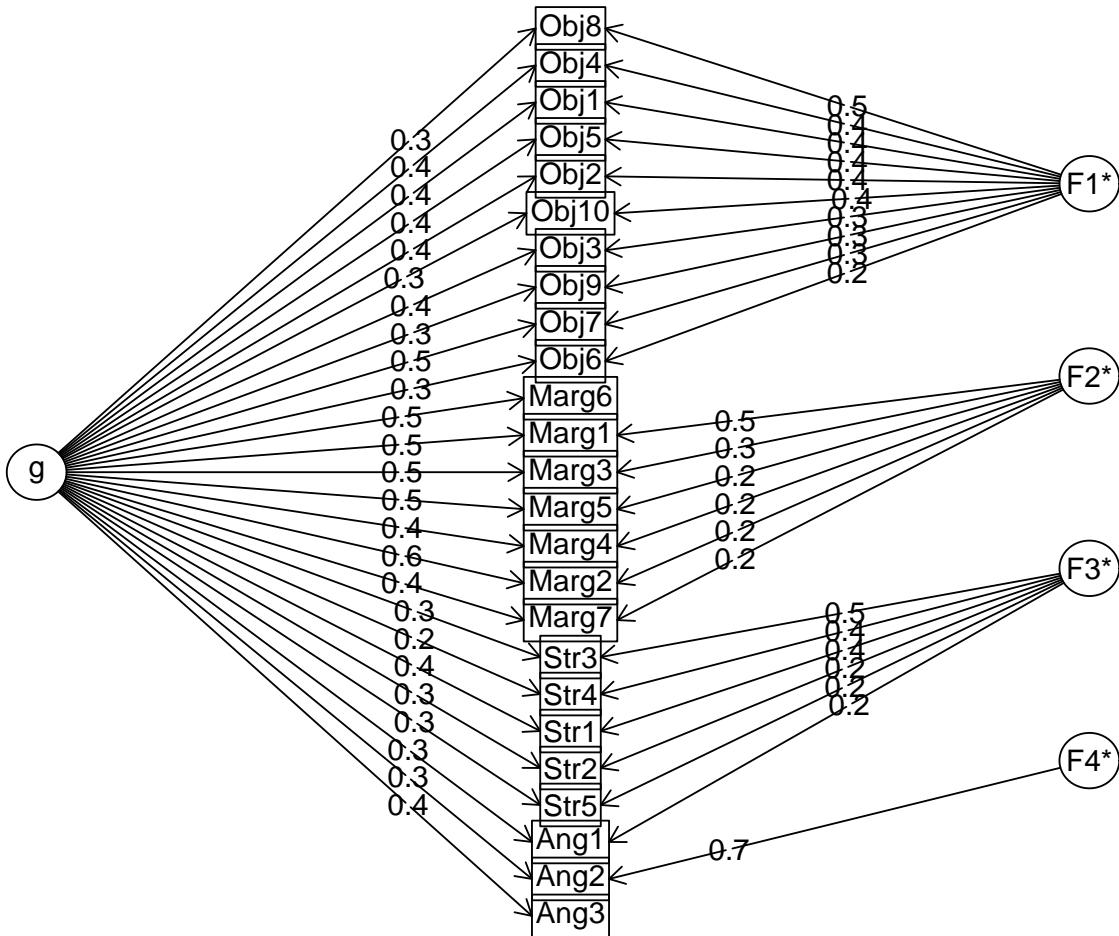
```
# Because we added the factor scores to our df (and now it has more
# variables than just our items), I will estimate omegaSem with the
# correlation matrix; I will need to tell it the n.obs

psych::omegaSem(GRMSr, nfactors = 4, n.obs = 259)
```

Warning in lav\_model\_vcov(lavmodel = lavmodel, lavsamplestats = lavsamplestats, : lavaan WARNING: Could not compute standard errors! The information matrix could not be inverted. This may be a symptom that the model is not identified.



# Omega from SEM



```
Call: psych::omegaSem(m = GRMSr, nfactors = 4, n.obs = 259)
Omega
Call: omegah(m = m, nfactors = nfactors, fm = fm, key = key, flip = flip,
  digits = digits, title = title, sl = sl, labels = labels,
  plot = plot, n.obs = n.obs, rotate = rotate, Phi = Phi, option = option,
  covar = covar)
Alpha:                 0.84
G.6:                  0.86
Omega Hierarchical:   0.57
Omega H asymptotic:  0.66
Omega Total:          0.86
```

	Schmid	Leiman	Factor loadings greater than 0.2					
	g	F1*	F2*	F3*	F4*	h2	u2	p2
Obj1	0.37	0.39				0.29	0.71	0.46
Obj2	0.40	0.40				0.32	0.68	0.49

Obj3	0.35	0.35		0.25	0.75	0.48	
Obj4	0.37	0.43		0.33	0.67	0.42	
Obj5	0.36	0.36		0.28	0.72	0.47	
Obj6	0.27	0.32		0.23	0.22	0.78	0.33
Obj7	0.41	0.29		0.27	0.73	0.62	
Obj8	0.35	0.41		-0.20	0.35	0.65	0.34
Obj9	0.28	0.30		0.19	0.81	0.41	
Obj10	0.28	0.37		0.22	0.78	0.36	
Marg1	0.53		0.58	0.62	0.38	0.46	
Marg2	0.49		0.35	0.41	0.59	0.59	
Marg3	0.42		0.30	0.28	0.72	0.62	
Marg4	0.39		0.28	0.28	0.72	0.55	
Marg5	0.46		0.32	0.33	0.67	0.64	
Marg6	0.41	0.24		0.27	0.73	0.62	
Marg7	0.33		0.25	0.18	0.82	0.59	
Str1	0.34		0.48	0.36	0.64	0.31	
Str2	0.29		0.23	0.18	0.82	0.46	
Str3	0.25		0.46	0.28	0.72	0.23	
Str4	0.21		0.34	0.17	0.83	0.25	
Str5	0.25		0.30	0.18	0.82	0.36	
Ang1	0.28		0.29	0.22	0.22	0.78	0.34
Ang2	0.27			0.61	0.44	0.56	0.16
Ang3	0.30			0.28	0.19	0.81	0.47

With Sums of squares of:

g	F1*	F2*	F3*	F4*
3.14	1.47	0.89	0.90	0.71

general/max 2.14 max/min = 2.07

mean percent general = 0.44 with sd = 0.13 and cv of 0.3

Explained Common Variance of the general factor = 0.44

The degrees of freedom are 206 and the fit is 0.77

The number of observations was 259 with Chi Square = 189.17 with prob < 0.79

The root mean square of the residuals is 0.04

The df corrected root mean square of the residuals is 0.04

RMSEA index = 0 and the 10 % confidence intervals are 0 0.018

BIC = -955.53

Compare this with the adequacy of just a general factor and no group factors

The degrees of freedom for just the general factor are 275 and the fit is 1.79

The number of observations was 259 with Chi Square = 445.32 with prob < 0.00000000034

The root mean square of the residuals is 0.09

The df corrected root mean square of the residuals is 0.09

RMSEA index = 0.049 and the 10 % confidence intervals are 0.041 0.057

BIC = -1082.81

## Measures of factor score adequacy

	g	F1*	F2*	F3*	F4*
Correlation of scores with factors	0.77	0.72	0.67	0.70	0.70
Multiple R square of scores with factors	0.60	0.52	0.44	0.49	0.49
Minimum correlation of factor score estimates	0.20	0.03	-0.11	-0.02	-0.01

## Total, General and Subset omega for each subset

	g	F1*	F2*	F3*	F4*
Omega total for total scores and subscales	0.86	0.78	0.72	0.59	0.44
Omega general for total scores and subscales	0.57	0.39	0.44	0.22	0.13
Omega group for total scores and subscales	0.19	0.39	0.28	0.37	0.31

The following analyses were done using the lavaan package

Omega Hierarchical from a confirmatory model using sem = 0.7

Omega Total from a confirmatory model using sem = 0.86

With loadings of

	g	F1*	F2*	F3*	F4*	h2	u2	p2
Obj1	0.38	0.39				0.30	0.70	0.48
Obj2	0.42	0.37				0.31	0.69	0.57
Obj3	0.37	0.32				0.24	0.76	0.57
Obj4	0.37	0.42				0.31	0.69	0.44
Obj5	0.36	0.38				0.28	0.72	0.46
Obj6	0.31	0.25				0.16	0.84	0.60
Obj7	0.46	0.26				0.28	0.72	0.76
Obj8	0.31	0.45				0.30	0.70	0.32
Obj9	0.28	0.31				0.17	0.83	0.46
Obj10	0.28	0.35				0.21	0.79	0.37
Marg1	0.54		0.54			0.58	0.42	0.50
Marg2	0.58		0.22			0.39	0.61	0.86
Marg3	0.46		0.28			0.29	0.71	0.73
Marg4	0.45		0.23			0.25	0.75	0.81
Marg5	0.53		0.23			0.33	0.67	0.85
Marg6	0.49					0.26	0.74	0.92
Marg7	0.37		0.21			0.18	0.82	0.76
Str1	0.36		0.40			0.29	0.71	0.45
Str2	0.34		0.23			0.17	0.83	0.68
Str3	0.27		0.50			0.32	0.68	0.23
Str4	0.20		0.41			0.21	0.79	0.19
Str5	0.31		0.22			0.14	0.86	0.69
Ang1	0.33		0.22			0.16	0.84	0.68
Ang2	0.35			0.69	0.60	0.40	0.20	
Ang3	0.39				0.18	0.82	0.85	

With sum of squared loadings of:

g	F1*	F2*	F3*	F4*
3.83	1.28	0.56	0.72	0.50

The degrees of freedom of the confirmatory model are 250 and the fit is 263.6354 with p = general/max 2.98 max/min = 2.55  
mean percent general = 0.58 with sd = 0.22 and cv of 0.37  
Explained Common Variance of the general factor = 0.56

Measures of factor score adequacy

	g	F1*	F2*	F3*	F4*
Correlation of scores with factors		0.86	0.75	0.65	0.69 0.77
Multiple R square of scores with factors		0.75	0.56	0.42	0.48 0.59
Minimum correlation of factor score estimates	0.49	0.13	-0.16	-0.05	0.19

Total, General and Subset omega for each subset

	g	F1*	F2*	F3*	F4*
Omega total for total scores and subscales	0.86	0.78	0.74	0.60	0.51
Omega general for total scores and subscales	0.70	0.43	0.56	0.28	0.22
Omega group for total scores and subscales	0.16	0.35	0.19	0.33	0.29

To get the standard sem fit statistics, ask for summary on the fitted object

There's a ton of output! How do we make sense of it?

First, excepting for the Angry scale (noted before), our items aligned reasonably with their respective factors (subsubscales).

Second, we can interpret our results. Like alpha, the omegas range from 0 to 1, where values closer to 1 represent good reliability [Najera Catalan, 2019]. For unidimensional measures,  $\omega_t$  values above 0.80 seem to be an indicator of good reliability. For multidimensional measures with well-defined dimensions we strive for  $\omega_h$  values above 0.65 (and  $\omega_t > 0.8$ ). These recommendations are based on a Monte Carlo study that examined a host of reliability indicators and how their values corresponded with accurate predictions of poverty status. With this in mind, let's examine the output related to our simulated research vignette.

Let's examine the output in the lower portion where the values are "from a confirmatory model using sem."

Omega is a reliability estimate for factor analysis that represents the proportion of variance in the GRMS scale attributable to common variance (rather than error). The omega for the total reliability of the test ( $\omega_t$ ; which included the general factors and the subscale factors) was .86, meaning that 86% of the variance in the total scale is due to the factors and 14% (100% - 86%) is attributable to error.

Omega hierarchical ( $\omega_h$ ) estimates are the proportion of variance in the GRMS score attributable to the general factor, which in effect treats the subscales as error.  $\omega_h$  for the the GRMS total scale was .70 A quick calculation with  $\omega_h$  (.70) and  $\omega_t$  (.86;  $.70/.86 = .81$ ) lets us know that that 81% of the reliable variance in the GRMS total scale is attributable to the general factor.

.70/.86

[1] 0.8139535

Amongst the output is the Cronbach's alpha coefficient (.84). Lewis and Neville [2015] did not report omega results. They reported an alpha of .92 for the version of the GRMS that assessed stress appraisal.

## 9.8 Comparing PFA to Item Analysis and PCA

In the lesson on PCA, we began a table that compared our item analysis (item corrected-total correlations with item-other scale correlations) and PCA results (both orthogonal and oblique). Let's now add our PAF results (both orthogonal and oblique).

In the prior lecture, I saved the file as both .rds and .csv objects. I will bring back in the .rds object and add to it.

```
GRMScomps <- readRDS("GRMS_Comparisons.rds")
grmsPAF2ORTH
```

```
Factor Analysis using method = pa
Call: psych::fa(r = dfGRMS, nfactors = 4, rotate = "varimax", fm = "pa")
Standardized loadings (pattern matrix) based upon correlation matrix
    PA1   PA2   PA3   PA4   h2   u2 com
Obj1  0.49  0.14  0.16  0.04  0.29  0.71  1.4
Obj2  0.51  0.18  0.13  0.09  0.32  0.68  1.4
Obj3  0.45  0.14  0.15  0.07  0.25  0.75  1.5
Obj4  0.54  0.19  0.04  0.03  0.33  0.67  1.3
Obj5  0.47  0.15  0.19 -0.02  0.28  0.72  1.6
Obj6  0.39  0.05  0.06  0.26  0.22  0.78  1.8
Obj7  0.41  0.27  0.16  0.10  0.27  0.73  2.2
Obj8  0.52  0.23  0.03 -0.18  0.35  0.65  1.7
Obj9  0.38  0.07  0.19 -0.01  0.19  0.81  1.5
Obj10 0.44  0.06  0.06  0.12  0.22  0.78  1.2
Marg1 0.13  0.77  0.08  0.02  0.62  0.38  1.1
Marg2 0.13  0.53  0.30  0.15  0.41  0.59  1.9
Marg3 0.18  0.46  0.16  0.08  0.28  0.72  1.6
Marg4 0.26  0.45 -0.01  0.13  0.28  0.72  1.8
Marg5 0.23  0.49  0.16  0.07  0.33  0.67  1.7
Marg6 0.35  0.35  0.08  0.16  0.27  0.73  2.5
Marg7 0.14  0.38  0.07  0.10  0.18  0.82  1.5
Str1  0.16  0.16  0.55 -0.08  0.36  0.64  1.4
Str2  0.24  0.09  0.30  0.16  0.18  0.82  2.7
Str3  0.07  0.07  0.51  0.06  0.28  0.72  1.1
Str4  0.14  0.04  0.38 -0.03  0.17  0.83  1.3
Str5  0.11  0.09  0.36  0.16  0.18  0.82  1.8
Ang1  0.02  0.18  0.35  0.25  0.22  0.78  2.4
Ang2  0.05  0.20  0.06  0.62  0.43  0.57  1.2
Ang3  0.10  0.26  0.15  0.31  0.19  0.81  2.7
```

	PA1	PA2	PA3	PA4
--	-----	-----	-----	-----

SS loadings	2.60	2.26	1.41	0.83
Proportion Var	0.10	0.09	0.06	0.03
Cumulative Var	0.10	0.19	0.25	0.28
Proportion Explained	0.37	0.32	0.20	0.12
Cumulative Proportion	0.37	0.68	0.88	1.00

Mean item complexity = 1.7

Test of the hypothesis that 4 factors are sufficient.

df null model = 300 with the objective function = 4.89 with Chi Square = 1217.51  
 df of the model are 206 and the objective function was 0.77

The root mean square of the residuals (RMSR) is 0.04

The df corrected root mean square of the residuals is 0.04

The harmonic n.obs is 259 with the empirical chi square 202.41 with prob < 0.56  
 The total n.obs was 259 with Likelihood Chi Square = 189.19 with prob < 0.79

Tucker Lewis Index of factoring reliability = 1.027

RMSEA index = 0 and the 90 % confidence intervals are 0 0.018

BIC = -955.52

Fit based upon off diagonal values = 0.97

Measures of factor score adequacy

	PA1	PA2	PA3	PA4
Correlation of (regression) scores with factors	0.84	0.85	0.76	0.72
Multiple R square of scores with factors	0.71	0.72	0.58	0.52
Minimum correlation of possible factor scores	0.42	0.45	0.17	0.03

```
# names(grmsPAF2ORTH) I had to add 'unclass' to the loadings to
# render them into a df
pafORTH_loadings <- data.frame(unclass(grmsPAF2ORTH$loadings))
pafORTH_loadings$Items <- c("Obj1", "Obj2", "Obj3", "Obj4", "Obj5", "Obj6",
  "Obj7", "Obj8", "Obj9", "Obj10", "Marg1", "Marg2", "Marg3", "Marg4",
  "Marg5", "Marg6", "Marg7", "Strong1", "Strong2", "Strong3", "Strong4",
  "Strong5", "Angry1", "Angry2", "Angry3") #Item names for joining (and to make sure we know
pafORTH_loadings <- dplyr::rename(pafORTH_loadings, PAF_OR_Obj = PA1, PAF_OR_Mar = PA2,
  PAF_OR_Str = PA3, PAF_OR_Ang = PA4)
# I had to add 'unclass' to the loadings to render them into a df
GRMScomps <- dplyr::full_join(GRMScomps, pafORTH_loadings, by = "Items")

# Now adding the PAF oblique loadings
pafOBLQ_loadings <- data.frame(unclass(grmsPAF2obl$loadings)) #I had to add 'unclass' to the
pafOBLQ_loadings$Items <- c("Obj1", "Obj2", "Obj3", "Obj4", "Obj5", "Obj6",
  "Obj7", "Obj8", "Obj9", "Obj10", "Marg1", "Marg2", "Marg3", "Marg4",
  "Marg5", "Marg6", "Marg7", "Strong1", "Strong2", "Strong3", "Strong4",
  "Strong5", "Angry1", "Angry2", "Angry3")
```

```

# Item names for joining (and to make sure we know which variable is
# which)
pafOBLQ_loadings <- dplyr::rename(pafOBLQ_loadings, PAF_OB_Obj = PA1, PAF_OB_Mar = PA2,
                                    PAF_OB_Str = PA3, PAF_OB_Ang = PA4)

# I had to add 'unclass' to the loadings to render them into a df
GRMScomps <- dplyr::full_join(GRMScomps, pafOBLQ_loadings, by = "Items")

# Writes the table to a .csv file where you can open it with Excel
# and format )
write.csv(GRMScomps, file = "GRMS_Comps.csv", sep = ",", row.names = FALSE,
          col.names = TRUE)

```

Warning in write.csv(GRMScomps, file = "GRMS\_Comps.csv", sep = ",", row.names = FALSE, : attempt to set 'col.names' ignored

Warning in write.csv(GRMScomps, file = "GRMS\_Comps.csv", sep = ",", row.names = FALSE, : attempt to set 'sep' ignored

Below we can see the consistency across item analysis, PCA (orthogonal and oblique), and PAF (orthogonal and oblique) comparisons. The results were consistent across the analyses, pointing only to problems with the Angry scale. Please note that the problems were with my simulated data and not the original data.

Items	r.drop	Item Analysis			Principal Components Analysis: Orthogonal Rotation			Principal Components Analysis: Oblique Rotation			Principal Axis Factoring: Orthogonal Rotation			Principal Axis Factoring: Oblique Rotation							
		Objectified	Marginalized	Strong	Angry	objORTH	marginORTH	strORTH	angORTH2	objOBLO	marginOBLO	strOBLO	angOBLO	PAF OR Obj	PAF OR Mar	PAF OR Str	PAF OR Ang	PAF OB Obj	PAF OB Mar	PAF OB Str	PAF OB Ang
Obj1	0.44	<b>0.47</b>	0.28	0.27	0.15	<b>0.57</b>	0.12	0.13	0.08	<b>0.57</b>	0.01	0.09	0.03	<b>0.49</b>	0.14	0.16	0.04	<b>0.51</b>	0.00	0.08	0.00
Obj2	0.47	<b>0.49</b>	0.35	0.24	0.16	<b>0.58</b>	0.18	0.10	0.09	<b>0.57</b>	0.08	0.05	0.04	<b>0.51</b>	0.18	0.13	0.09	<b>0.53</b>	0.02	0.04	0.05
Obj3	0.41	<b>0.43</b>	0.30	0.25	0.13	<b>0.53</b>	0.13	0.14	0.07	<b>0.53</b>	0.02	0.10	0.02	<b>0.45</b>	0.14	0.15	0.07	<b>0.46</b>	0.00	0.07	0.04
Obj4	0.44	<b>0.49</b>	0.33	0.18	0.14	<b>0.60</b>	0.20	0.00	0.01	<b>0.60</b>	0.11	-0.05	-0.04	<b>0.54</b>	0.19	0.04	0.03	<b>0.56</b>	0.05	-0.05	0.00
Obj5	0.43	<b>0.44</b>	0.30	0.26	0.15	<b>0.53</b>	0.18	0.20	-0.10	<b>0.51</b>	0.09	0.17	-0.15	<b>0.47</b>	0.15	0.19	-0.02	<b>0.47</b>	0.02	0.12	-0.06
Obj6	0.34	<b>0.35</b>	0.22	0.17	0.18	<b>0.49</b>	-0.04	-0.02	0.43	<b>0.53</b>	-0.17	-0.05	0.40	<b>0.39</b>	0.05	0.06	0.26	<b>0.43</b>	-0.11	-0.02	0.25
Obj7	0.47	<b>0.43</b>	0.36	0.26	0.24	<b>0.46</b>	0.28	0.13	0.15	<b>0.42</b>	0.19	0.07	0.10	<b>0.41</b>	0.27	0.16	0.10	<b>0.38</b>	0.14	0.08	0.06
Obj8	0.38	<b>0.45</b>	0.31	0.14	0.05	<b>0.57</b>	0.27	0.00	-0.26	<b>0.55</b>	0.22	-0.05	-0.32	<b>0.52</b>	0.23	0.03	-0.18	<b>0.53</b>	0.15	-0.05	-0.22
Obj9	0.34	<b>0.37</b>	0.22	0.20	0.11	<b>0.47</b>	0.05	0.21	-0.05	<b>0.46</b>	-0.04	0.19	-0.09	<b>0.38</b>	0.07	0.19	-0.01	<b>0.39</b>	-0.05	0.14	-0.04
Obj10	0.35	<b>0.39</b>	0.22	0.17	0.14	<b>0.54</b>	0.02	0.00	0.17	<b>0.58</b>	-0.09	-0.04	0.13	<b>0.44</b>	0.06	0.06	0.12	<b>0.48</b>	-0.08	-0.02	0.10
Marg1	0.49	0.33	<b>0.60</b>	0.21	0.28	0.11	<b>0.78</b>	0.06	0.05	-0.03	<b>0.81</b>	-0.03	0.01	0.13	<b>0.77</b>	0.08	0.02	-0.03	<b>0.81</b>	-0.03	-0.02
Marg2	0.50	0.32	<b>0.52</b>	0.32	0.33	0.09	<b>0.61</b>	0.28	0.18	-0.04	<b>0.60</b>	0.21	0.14	0.13	<b>0.53</b>	0.30	0.15	-0.01	<b>0.49</b>	0.22	0.12
Marg3	0.42	0.30	<b>0.46</b>	0.22	0.23	0.13	<b>0.60</b>	0.16	0.02	0.02	<b>0.60</b>	0.08	-0.02	0.18	<b>0.46</b>	0.16	0.08	<b>0.43</b>	0.09	0.05	
Marg4	0.41	0.33	<b>0.42</b>	0.15	0.24	0.24	<b>0.56</b>	-0.07	0.11	0.16	<b>0.55</b>	-0.16	0.07	0.26	<b>0.45</b>	-0.01	0.13	<b>0.40</b>	-0.11	0.11	
Marg5	0.47	0.36	<b>0.50</b>	0.26	0.22	0.20	<b>0.60</b>	0.15	0.04	0.09	<b>0.59</b>	0.07	0.00	0.23	<b>0.49</b>	0.16	0.07	0.14	<b>0.45</b>	0.08	0.03
Marg6	0.46	0.40	<b>0.41</b>	0.22	0.21	0.37	<b>0.40</b>	0.03	0.18	0.32	<b>0.35</b>	-0.04	0.13	0.35	<b>0.35</b>	0.08	0.16	0.31	<b>0.25</b>	-0.02	0.13
Marg7	0.33	0.25	<b>0.37</b>	0.13	0.19	0.10	<b>0.51</b>	0.00	0.12	0.02	<b>0.52</b>	-0.06	0.09	0.14	<b>0.38</b>	0.19	0.10	<b>0.36</b>	0.20	0.08	
Strong1	0.37	0.28	<b>0.39</b>	0.26	0.16	0.16	<b>0.49</b>	0.65	-0.12	0.06	<b>0.64</b>	-0.15	0.16	0.16	<b>0.55</b>	-0.08	0.07	<b>0.56</b>	-0.12		
Strong2	0.24	0.28	<b>0.28</b>	0.24	0.17	0.07	<b>0.64</b>	0.38	0.24	0.24	<b>0.67</b>	0.21	0.24	0.09	<b>0.50</b>	0.16	0.21	<b>0.54</b>	0.24	0.14	
Strong3	0.29	0.18	<b>0.38</b>	0.28	0.05	0.05	<b>0.66</b>	0.69	-0.02	0.08	<b>0.67</b>	0.07	0.07	0.51	<b>0.66</b>	-0.01	-0.02	<b>0.53</b>	0.04		
Strong4	0.24	0.18	<b>0.14</b>	0.32	0.10	0.14	<b>0.62</b>	0.57	-0.12	0.09	<b>0.64</b>	0.15	0.14	0.04	<b>0.38</b>	-0.03	0.10	<b>0.39</b>	-0.08		
Strong5	0.29	0.21	<b>0.20</b>	0.25	0.23	0.10	<b>0.66</b>	0.47	0.25	0.05	<b>0.62</b>	0.23	0.11	0.09	<b>0.36</b>	0.16	0.08	-0.01	<b>0.35</b>	0.14	
Angry1	0.31	0.16	<b>0.25</b>	<b>0.31</b>	0.04	0.20	<b>0.44</b>	<b>0.37</b>	-0.12	0.14	<b>0.42</b>	<b>0.36</b>	0.02	0.18	<b>0.35</b>	<b>0.25</b>	-0.07	0.10	<b>0.33</b>	<b>0.34</b>	
Angry2	0.28	0.15	0.28	0.15	<b>0.31</b>	0.03	0.19	0.01	<b>0.73</b>	0.01	0.13	-0.04	<b>0.73</b>	0.05	0.20	0.06	<b>0.62</b>	0.01	0.07	-0.02	<b>0.64</b>
Angry3	0.32	0.23	0.28	0.18	<b>0.31</b>	0.09	0.25	0.12	<b>0.56</b>	0.05	0.18	0.07	<b>0.54</b>	0.10	0.26	0.15	<b>0.31</b>	0.04	0.18	0.09	<b>0.30</b>

Figure 9.7: Comparison of path models for PCA and EFA

## 9.9 Practice Problems

In each of these lessons I provide suggestions for practice that allow you to select one or more problems that are graded in difficulty. In psychometrics, I strongly recommend that you have started with a dataset that has a minimum of three subscales and use it for all of the assignments in the OER. In any case, please plan to:

- Properly format and prepare the data.

- Conduct diagnostic tests to determine the suitability of the data for PAF
- Conduct tests to guide the decisions about number of factors to extract.
- Conduct orthogonal and oblique rotations (at least two each with different numbers of factor).
- Select one solution and preparing an APA style results section (with table and figure).
- Compare your results in light of any other psychometrics lessons where you have used this data (especially the **item analysis** and **PCA** lessons).

### 9.9.1 Problem #1: Play around with this simulation.

Copy the script for the simulation and then change (at least) one thing in the simulation to see how it impacts the results. If PAF is new to you, perhaps you just change the number in “`set.seed(240311)`” from 240311 to something else. Your results should *parallel* those obtained in the lecture, making it easier for you to check your work as you go. Don’t be surprised if the factor loadings wiggle around a little. Do try to make sense of them.

### 9.9.2 Problem #2: Conduct a PAF with another simulated set of data in the OER.

The second option involves utilizing one of the simulated datasets available in this OER. The [last lesson](#) in the OER contains three simulations that could be used for all of the statistics-based practice suggestions. Especially if you started with one of these examples in an earlier lesson, I highly recommend you continue with that.

Alternatively, Keum et al.’s Gendered Racial Microaggressions Scale for Asian American Women [[Keum et al., 2018](#)] will be used in the lessons on confirmatory factor analysis and Conover et al.’s [[Conover et al., 2017](#)] Ableist Microaggressions Scale is used in the lesson on invariance testing. Both of these would be suitable for the PCA and PAF homework assignments.

### 9.9.3 Problem #3: Try something entirely new.

Using data for which you have permission and access (e.g., IRB approved data you have collected or from your lab; data you simulate from a published article; data from the ReCentering Psych Stats survey described in the [Qualtrics lesson](#), or data from an open science repository), complete a PAF analysis. The data should allow for at least three factors/subscales.

### 9.9.4 Grading Rubric

Using the lecture and workflow (chart) as a guide, please work through all the steps listed in the proposed assignment/grading rubric.

Assignment Component	Points Possible	Points Earned
1. Check and, if needed, format data	5	_____
2. Conduct and interpret the three diagnostic tests to determine if PAF is appropriate as an analysis (KMO, Bartlett’s determinant).	5	_____

Assignment Component	Points Possible	Points Earned
3. Determine how many factors to extract (e.g., scree plot, eigenvalues, theory).	5	_____
4. Conduct an orthogonal rotation with a minimum of two different numbers of factor extractions.	5	_____
5. Conduct an oblique rotation with a minimum of two different numbers of factor extractions.	5	_____
6. Determine which factor solution (e.g., orthogonal or oblique; which number of factors) you will suggest.	5	_____
7. APA style results section with table and figure of one of the solutions.	5	_____
8. Explanation to grader	5	_____
<b>Totals</b>	<b>40</b>	_____

## 9.10 Homeworked Example

### Screencast Link

For more information about the data used in this homeworked example, please refer to the description and codebook located at the end of the [introduction](#) in first volume of ReCentering Psych Stats.

As a brief review, this data is part of an IRB-approved study, with consent to use in teaching demonstrations and to be made available to the general public via the open science framework. Hence, it is appropriate to use in this context. You will notice there are student- and teacher- IDs. These numbers are not actual student and teacher IDs, rather they were further re-identified so that they could not be connected to actual people.

Because this is an actual dataset, if you wish to work the problem along with me, you will need to download the [ReC.rds](#) data file from the Worked\_Examples folder in the ReC\_Psychometrics project on the GitHub.

The course evaluation items can be divided into three subscales:

- **Valued by the student** includes the items: ValObjectives, IncrUnderstanding, IncrInterest
- **Traditional pedagogy** includes the items: ClearResponsibilities, EffectiveAnswers, Feedback, ClearOrganization, ClearPresentation
- **Socially responsive pedagogy** includes the items: InclusvClassrm, EquitableEval, Mult-Perspectives, DEIntegration

In this homewoRked example I will use principal axis factoring in an exploratory factor analysis. My hope is that the results will support my solution of three dimensions: valued-by-the-student, traditional pedagogy, socially responsive pedagogy.

### 9.10.1 Check and, if needed, format data

```
big <- readRDS("ReC.rds")
```

With the next code I will create an item-level df with only the items used in the three scales.

```
library(tidyverse)
items <- big %>%
  dplyr::select(ValObjectives, IncrUnderstanding, IncrInterest, ClearResponsibilities,
    EffectiveAnswers, Feedback, ClearOrganization, ClearPresentation,
    MultPerspectives, InclusvClassrm, DEIintegration, EquitableEval)
```

Some of the analyses require non-missing data in the df.

```
items <- na.omit(items)
```

Let's check the structure of the data.

```
str(items)
```

```
Classes 'data.table' and 'data.frame': 267 obs. of 12 variables:
 $ ValObjectives      : int  5 5 4 4 5 5 5 4 5 3 ...
 $ IncrUnderstanding   : int  2 3 4 3 4 5 2 4 5 4 ...
 $ IncrInterest        : int  5 3 4 2 4 5 3 2 5 1 ...
 $ ClearResponsibilities: int  5 5 4 4 5 5 4 4 5 3 ...
 $ EffectiveAnswers    : int  5 3 5 3 5 4 3 2 3 3 ...
 $ Feedback             : int  5 3 4 2 5 5 4 4 5 2 ...
 $ ClearOrganization    : int  3 4 3 4 4 5 4 4 5 2 ...
 $ ClearPresentation    : int  4 4 4 2 5 4 4 4 5 2 ...
 $ MultPerspectives     : int  5 5 4 5 5 5 5 5 5 1 ...
 $ InclusvClassrm       : int  5 5 5 5 5 5 5 4 5 3 ...
 $ DEIintegration       : int  5 5 5 5 5 5 5 5 5 2 ...
 $ EquitableEval         : int  5 5 3 5 5 5 5 3 5 3 ...
 - attr(*, ".internal.selfref")=<externalptr>
 - attr(*, "na.action")= 'omit' Named int [1:43] 6 20 106 109 112 113 114 117 122 128 ...
 ..- attr(*, "names")= chr [1:43] "6" "20" "106" "109" ...
```

### 9.10.2 Conduct and interpret the three diagnostic tests to determine if PAF is appropriate as an analysis (KMO, Bartlett's, determinant)

#### 9.10.2.1 KMO

The Kaiser-Meyer-Olkin (KMO) index is an index of *sampling adequacy* to let us know if the sample size is sufficient relative to the statistical characteristics of the data.

General criteria (1974, Kaiser):

- bare minimum of .5
- values between .5 and .7 as mediocre
- values above .9 are superb

```
psych::KMO(items)
```

Kaiser-Meyer-Olkin factor adequacy  
 Call: psych::KMO(r = items)  
 Overall MSA = 0.91  
 MSA for each item =

	ValObjectives	IncrUnderstanding	IncrInterest
	0.94	0.89	0.89
ClearResponsibilities		EffectiveAnswers	Feedback
	0.91	0.93	0.94
ClearOrganization		ClearPresentation	MultPerspectives
	0.94	0.91	0.93
InclusvClassrm		DEIntegration	EquitableEval
	0.86	0.78	0.95

With a KMO of 0.91, the data seems appropriate to continue with the PCA.

### 9.10.2.2 Bartlett's

Bartlett's test lets us know if the matrix is an *identity matrix* (i.e., where elements on the off-diagonal would be 0.0 and elements on the diagonal would be 1.0). Stated another way – items only correlate with “themselves” and not other variables.

When  $p < 0.05$  the matrix is not an identity matrix. That is, there are some relationships between variables that can be analyzed.

```
psych::cortest.bartlett(items)
```

R was not square, finding R from data

```
$chisq
[1] 1897.769

$p.value
[1] 0

$df
[1] 66
```

The Barlett's test,  $\chi^2(66) = 1897.77, p < 0.001$ , indicating that the correlation matrix is not an identity matrix and, on that dimension, is suitable for analysis.

### 9.10.2.3 Determinant

Multicollinearity or singularity is diagnosed by the determinant. The determinant should be greater than 0.00001. If smaller, then there may be an issue with multicollinearity (variables that are too highly correlated) or singularity (variables that are perfectly correlated).

```
items <- na.omit(items)
det(cor(items))
```

```
[1] 0.0006985496
```

The value of the determinant is 0.0007; greater than 0.00001. We are not concerned with multicollinearity or singularity.

Summary from data screening:

Data screening were conducted to determine the suitability of the data for principal axis factoring. The Kaiser-Meyer-Olkin measure of sampling adequacy (KMO; Kaiser, 1970) represents the ratio of the squared correlation between variables to the squared partial correlation between variables. KMO ranges from 0.00 to 1.00; values closer to 1.00 indicate that the patterns of correlations are relatively compact, and that component analysis should yield distinct and reliable components (Field, 2012). In our dataset, the KMO value was 0.91, indicating acceptable sampling adequacy. The Bartlett's Test of Sphericity examines whether the population correlation matrix resembles an identity matrix (Field, 2012). When the  $p$  value for the Bartlett's test is  $< .05$ , we are fairly certain we have clusters of correlated variables. In our dataset,  $\chi^2(66) = 1897.77, p < 0.001$  indicating the correlations between items are sufficiently large enough for principal components analysis. The determinant of the correlation matrix alerts us to any issues of multicollinearity or singularity and should be larger than 0.00001. Our determinant was 0.0007 and, again, indicated that our data was suitable for the analysis.

### 9.10.3 Determine how many components to extract (e.g., scree plot, eigenvalues, theory)

Specify fewer factors than number of items (12). For me it wouldn't run unless it was 6 or fewer.

```
paf1 <- psych::fa(items, nfactors = 6, fm = "pa", max.iter = 100, rotate = "none") # using rat
```

```
maximum iteration exceeded
```

```
paf1
```

```
Factor Analysis using method = pa
Call: psych::fa(r = items, nfactors = 6, rotate = "none", max.iter = 100,
fm = "pa")
```

Standardized loadings (pattern matrix) based upon correlation matrix

	PA1	PA2	PA3	PA4	PA5	PA6	h2	u2	com
ValObjectives	0.52	-0.07	0.16	0.05	0.05	0.10	0.31	0.688	1.3
IncrUnderstanding	0.65	-0.28	0.31	0.03	-0.04	-0.01	0.60	0.400	1.9
IncrInterest	0.74	-0.18	0.50	-0.18	0.13	-0.02	0.87	0.127	2.1
ClearResponsibilities	0.80	-0.09	-0.34	0.10	0.03	-0.06	0.77	0.226	1.4
EffectiveAnswers	0.78	-0.14	-0.13	0.05	0.09	-0.17	0.68	0.315	1.3
Feedback	0.74	0.05	-0.20	-0.12	0.22	0.05	0.65	0.347	1.4
ClearOrganization	0.78	-0.28	-0.13	0.12	-0.07	0.32	0.83	0.175	1.8
ClearPresentation	0.84	-0.21	0.01	0.12	-0.20	-0.11	0.82	0.182	1.3
MultPerspectives	0.81	0.30	-0.12	-0.40	-0.23	0.03	0.97	0.029	2.0
InclusvClassrm	0.64	0.41	0.18	0.21	-0.14	-0.08	0.67	0.327	2.3
DEIIntegration	0.49	0.66	0.13	0.14	0.11	0.10	0.74	0.265	2.2
EquitableEval	0.69	0.09	-0.17	-0.02	0.13	-0.09	0.54	0.462	1.3

	PA1	PA2	PA3	PA4	PA5	PA6
SS loadings	6.11	0.97	0.65	0.31	0.22	0.19
Proportion Var	0.51	0.08	0.05	0.03	0.02	0.02
Cumulative Var	0.51	0.59	0.64	0.67	0.69	0.70
Proportion Explained	0.72	0.11	0.08	0.04	0.03	0.02
Cumulative Proportion	0.72	0.84	0.91	0.95	0.98	1.00

Mean item complexity = 1.7

Test of the hypothesis that 6 factors are sufficient.

df null model = 66 with the objective function = 7.27 with Chi Square = 1897.77  
 df of the model are 9 and the objective function was 0.06

The root mean square of the residuals (RMSR) is 0.01

The df corrected root mean square of the residuals is 0.02

The harmonic n.obs is 267 with the empirical chi square 2.97 with prob < 0.97  
 The total n.obs was 267 with Likelihood Chi Square = 14.81 with prob < 0.096

Tucker Lewis Index of factoring reliability = 0.976

RMSEA index = 0.049 and the 90 % confidence intervals are 0 0.093

BIC = -35.47

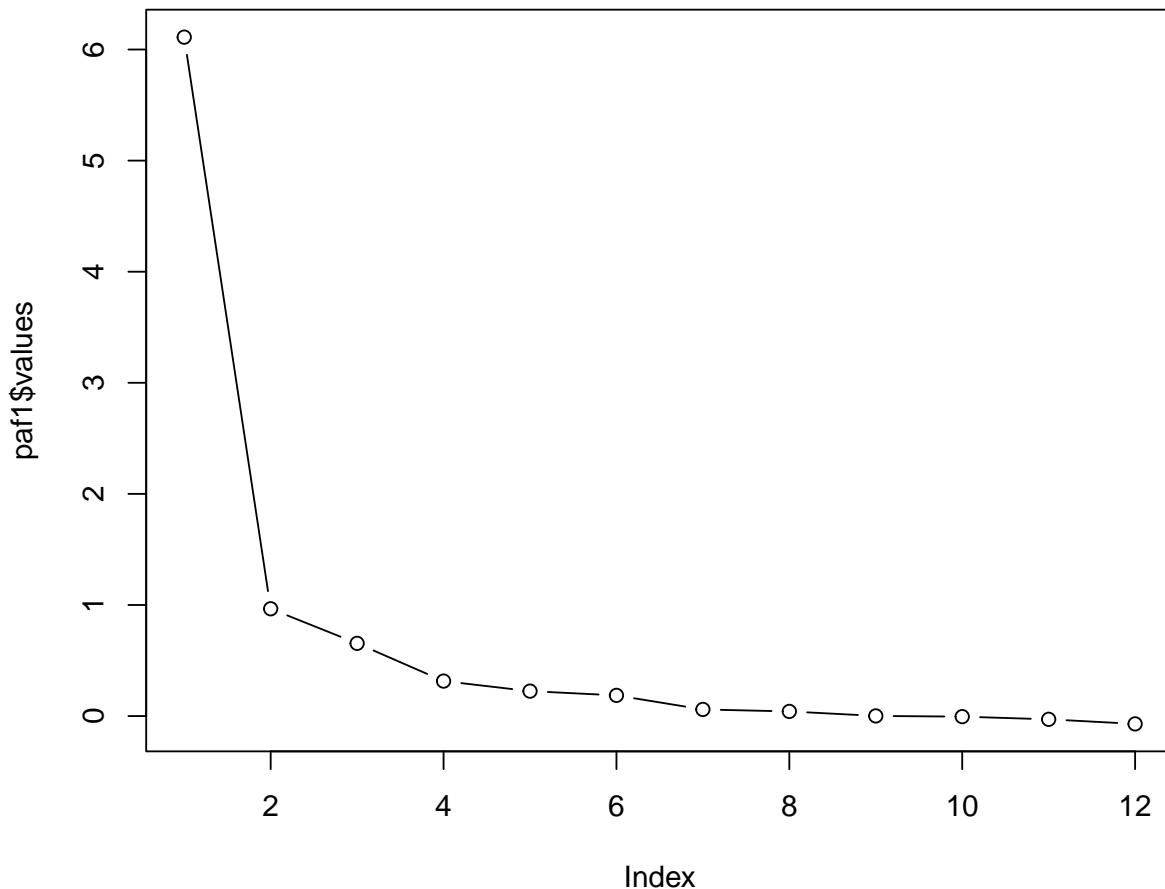
Fit based upon off diagonal values = 1

Measures of factor score adequacy

	PA1	PA2	PA3	PA4	PA5
Correlation of (regression) scores with factors	0.98	0.90	0.88	0.86	0.73
Multiple R square of scores with factors	0.97	0.81	0.77	0.74	0.54
Minimum correlation of possible factor scores	0.93	0.63	0.54	0.48	0.07
				PA6	
Correlation of (regression) scores with factors				0.68	
Multiple R square of scores with factors				0.46	
Minimum correlation of possible factor scores				-0.08	

The eigenvalue-greater-than-one criteria suggests 1 factor (but the second factor has an SSloading of .97).

```
plot(paf1$values, type = "b")
```



The scree plot looks like one factor.

Ugh.

- I want 3 factors (we could think of this as a priori theory); would account for 64% of variance.
- Two could account for 59% of variance.
- Eigenvalues-greater-than-one criteria and scree plot suggests 1 or 3 (would account for 64% of variance)

*Note:* The lecture has more on evaluating communalities and uniquenesses and how this information can also inform the number of components we want to extract. Because it is easy to get lost (very lost) I will skip over this for now. If you were to create a measure and use PAF as an exploratory approach to understanding the dimensionality of an instrument, you would likely want to investigate further and report on these.

#### 9.10.4 Conduct an orthogonal rotation with a minimum of two different factor extractions

##### An orthogonal two factor solution

```
pafORTH2f <- psych::fa(items, nfactors = 2, rotate = "varimax")
pafORTH2f
```

Factor Analysis using method = minres  
 Call: psych::fa(r = items, nfactors = 2, rotate = "varimax")  
 Standardized loadings (pattern matrix) based upon correlation matrix

	MR1	MR2	h2	u2	com
ValObjectives	0.48	0.21	0.27	0.73	1.4
IncrUnderstanding	0.67	0.12	0.46	0.54	1.1
IncrInterest	0.66	0.25	0.50	0.50	1.3
ClearResponsibilities	0.74	0.29	0.63	0.37	1.3
EffectiveAnswers	0.76	0.26	0.64	0.36	1.2
Feedback	0.63	0.38	0.54	0.46	1.7
ClearOrganization	0.79	0.17	0.65	0.35	1.1
ClearPresentation	0.83	0.23	0.75	0.25	1.2
MultPerspectives	0.58	0.55	0.64	0.36	2.0
InclusvClassrm	0.36	0.64	0.54	0.46	1.6
DEIIntegration	0.08	0.86	0.75	0.25	1.0
EquitableEval	0.57	0.40	0.49	0.51	1.8

	MR1	MR2
SS loadings	4.74	2.12
Proportion Var	0.39	0.18
Cumulative Var	0.39	0.57
Proportion Explained	0.69	0.31
Cumulative Proportion	0.69	1.00

Mean item complexity = 1.4

Test of the hypothesis that 2 factors are sufficient.

df null model = 66 with the objective function = 7.27 with Chi Square = 1897.77  
 df of the model are 43 and the objective function was 0.75

The root mean square of the residuals (RMSR) is 0.05

The df corrected root mean square of the residuals is 0.06

The harmonic n.obs is 267 with the empirical chi square 90.7 with prob < 0.000029  
 The total n.obs was 267 with Likelihood Chi Square = 194.26 with prob < 0.0000000000000000

Tucker Lewis Index of factoring reliability = 0.873

RMSEA index = 0.115 and the 90 % confidence intervals are 0.099 0.132

BIC = -46

Fit based upon off diagonal values = 0.99

Measures of factor score adequacy

	MR1	MR2
Correlation of (regression) scores with factors	0.94	0.9
Multiple R square of scores with factors	0.89	0.8
Minimum correlation of possible factor scores	0.78	0.6

Sorting the scores into a table can help see the results more clearly. The “cut = #” command will not show the factor scores for factor loading < .30. I would do this “to see”, but I would include all the values in an APA style table.

```
paf_tableORT2f <- psych::print.psych(pafORT2f, cut = 0.3, sort = TRUE)
```

Factor Analysis using method = minres

Call: psych::fa(r = items, nfactors = 2, rotate = "varimax")

Standardized loadings (pattern matrix) based upon correlation matrix

	item	MR1	MR2	h2	u2	com
ClearPresentation	8	0.83		0.75	0.25	1.2
ClearOrganization	7	0.79		0.65	0.35	1.1
EffectiveAnswers	5	0.76		0.64	0.36	1.2
ClearResponsibilities	4	0.74		0.63	0.37	1.3
IncrUnderstanding	2	0.67		0.46	0.54	1.1
IncrInterest	3	0.66		0.50	0.50	1.3
Feedback	6	0.63	0.38	0.54	0.46	1.7
MultPerspectives	9	0.58	0.55	0.64	0.36	2.0
EquitableEval	12	0.57	0.40	0.49	0.51	1.8
ValObjectives	1	0.48		0.27	0.73	1.4
DEIIntegration	11		0.86	0.75	0.25	1.0
InclusvClassrm	10	0.36	0.64	0.54	0.46	1.6

                  MR1    MR2

SS loadings	4.74	2.12
Proportion Var	0.39	0.18
Cumulative Var	0.39	0.57
Proportion Explained	0.69	0.31
Cumulative Proportion	0.69	1.00

Mean item complexity = 1.4

Test of the hypothesis that 2 factors are sufficient.

df null model = 66 with the objective function = 7.27 with Chi Square = 1897.77  
 df of the model are 43 and the objective function was 0.75

The root mean square of the residuals (RMSR) is 0.05

The df corrected root mean square of the residuals is 0.06

The harmonic n.obs is 267 with the empirical chi square 90.7 with prob < 0.000029

The total n.obs was 267 with Likelihood Chi Square = 194.26 with prob < 0.0000000000000000

Tucker Lewis Index of factoring reliability = 0.873

RMSEA index = 0.115 and the 90 % confidence intervals are 0.099 0.132

BIC = -46

Fit based upon off diagonal values = 0.99

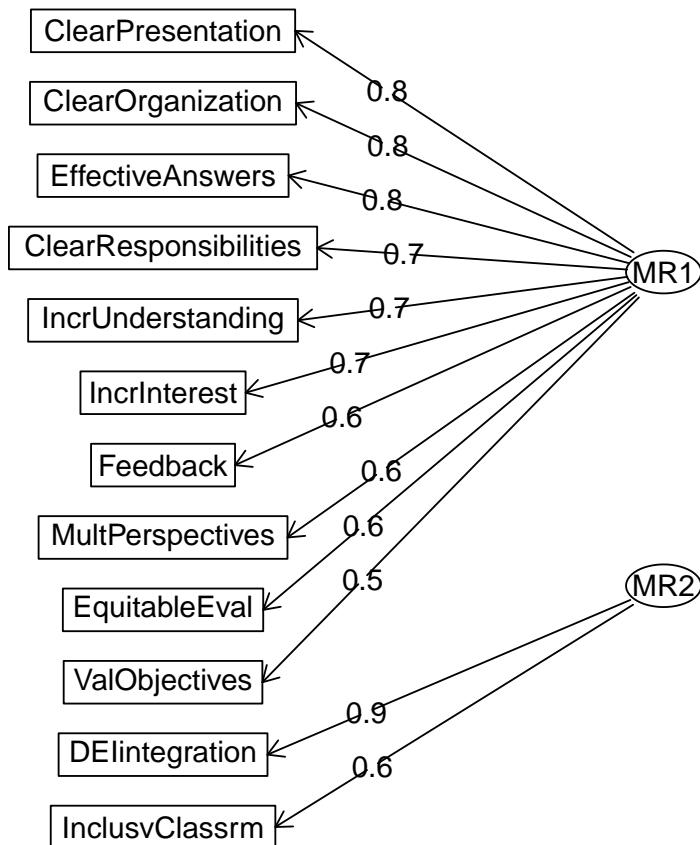
Measures of factor score adequacy

	MR1	MR2
Correlation of (regression) scores with factors	0.94	0.9
Multiple R square of scores with factors	0.89	0.8
Minimum correlation of possible factor scores	0.78	0.6

F1: Includes everything else. F2: Includes 2 SCR items – DEIintegration, InclsrvClssrm Also: EquitableEval MultPerspectives have high cross-loadings, but end up on the first factor

```
psych::fa.diagram(pafORTH2f)
```

## Factor Analysis



Plotting these figures from the program can facilitate conceptual understanding of what is going

on – and can be a “check” to your work.

In the lecture I made a “biggish deal” about PCA being *components* analysis and PAF being *factor* analysis. Although the two approaches can lead to similar results/conclusions, there are some significant differences “under the hood.” PCA can be thought of more as regression where the items predict the component. Consequently, the arrows went *from* the item *to* the component.

In PAF, the arrows will go from the factor to the item – because the factors (or latent variables) are assumed to predict the scores on the items (i.e., “depression” would predict how someone rates items that assess hopelessness, sleep, anhedonia, and so forth).

### An orthogonal three factor solution

```
pafORTH3f <- psych::fa(items, nfactors = 3, rotate = "varimax")
pafORTH3f
```

```
Factor Analysis using method = minres
Call: psych::fa(r = items, nfactors = 3, rotate = "varimax")
Standardized loadings (pattern matrix) based upon correlation matrix
          MR1   MR3   MR2   h2   u2 com
Val0bjectives    0.27  0.43  0.20  0.30  0.70  2.1
IncrUnderstanding 0.27  0.76  0.09  0.66  0.34  1.3
IncrInterest      0.27  0.75  0.25  0.69  0.31  1.5
ClearResponsibilities 0.84  0.24  0.17  0.79  0.21  1.2
EffectiveAnswers    0.67  0.41  0.18  0.64  0.36  1.8
Feedback           0.65  0.26  0.30  0.58  0.42  1.8
ClearOrganization    0.65  0.47  0.10  0.64  0.36  1.9
ClearPresentation    0.62  0.57  0.18  0.74  0.26  2.2
MultPerspectives     0.57  0.29  0.49  0.65  0.35  2.5
InclusvClassrm      0.28  0.30  0.63  0.56  0.44  1.9
DEIIntegration       0.14  0.07  0.85  0.75  0.25  1.1
EquitableEval        0.60  0.23  0.33  0.52  0.48  1.9

          MR1   MR3   MR2
SS loadings    3.37  2.39  1.77
Proportion Var  0.28  0.20  0.15
Cumulative Var  0.28  0.48  0.63
Proportion Explained 0.45  0.32  0.23
Cumulative Proportion 0.45  0.77  1.00
```

Mean item complexity = 1.8

Test of the hypothesis that 3 factors are sufficient.

```
df null model = 66 with the objective function = 7.27 with Chi Square = 1897.77
df of the model are 33 and the objective function was 0.29
```

The root mean square of the residuals (RMSR) is 0.02

The df corrected root mean square of the residuals is 0.03

The harmonic n.obs is 267 with the empirical chi square 19.55 with prob < 0.97  
 The total n.obs was 267 with Likelihood Chi Square = 75.2 with prob < 0.0000039

Tucker Lewis Index of factoring reliability = 0.954  
 RMSEA index = 0.069 and the 90 % confidence intervals are 0.049 0.09  
 BIC = -109.18  
 Fit based upon off diagonal values = 1  
 Measures of factor score adequacy

	MR1	MR3	MR2
Correlation of (regression) scores with factors	0.90	0.87	0.89
Multiple R square of scores with factors	0.82	0.76	0.79
Minimum correlation of possible factor scores	0.64	0.52	0.58

```
paf_table0R3f <- psych::print.psych(paf0RTH3f, cut = 0.3, sort = TRUE)
```

Factor Analysis using method = minres  
 Call: psych::fa(r = items, nfactors = 3, rotate = "varimax")  
 Standardized loadings (pattern matrix) based upon correlation matrix

	item	MR1	MR3	MR2	h2	u2	com
ClearResponsibilities	4	0.84		0.79	0.21	1.2	
EffectiveAnswers	5	0.67	0.41		0.64	0.36	1.8
Feedback	6	0.65		0.30	0.58	0.42	1.8
ClearOrganization	7	0.65	0.47		0.64	0.36	1.9
ClearPresentation	8	0.62	0.57		0.74	0.26	2.2
EquitableEval	12	0.60		0.33	0.52	0.48	1.9
MultPerspectives	9	0.57		0.49	0.65	0.35	2.5
IncrUnderstanding	2		0.76		0.66	0.34	1.3
IncrInterest	3		0.75		0.69	0.31	1.5
ValObjectives	1		0.43		0.30	0.70	2.1
DEIIntegration	11			0.85	0.75	0.25	1.1
InclusvClassrm	10			0.63	0.56	0.44	1.9

	MR1	MR3	MR2
SS loadings	3.37	2.39	1.77
Proportion Var	0.28	0.20	0.15
Cumulative Var	0.28	0.48	0.63
Proportion Explained	0.45	0.32	0.23
Cumulative Proportion	0.45	0.77	1.00

Mean item complexity = 1.8

Test of the hypothesis that 3 factors are sufficient.

df null model = 66 with the objective function = 7.27 with Chi Square = 1897.77  
 df of the model are 33 and the objective function was 0.29

The root mean square of the residuals (RMSR) is 0.02

The df corrected root mean square of the residuals is 0.03

The harmonic n.obs is 267 with the empirical chi square 19.55 with prob < 0.97  
The total n.obs was 267 with Likelihood Chi Square = 75.2 with prob < 0.000039

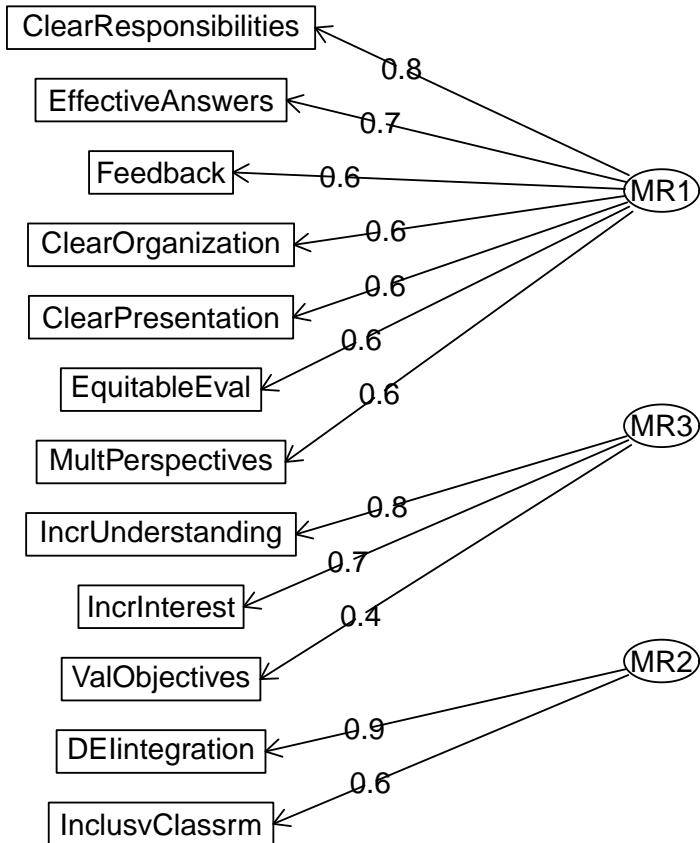
Tucker Lewis Index of factoring reliability = 0.954  
RMSEA index = 0.069 and the 90 % confidence intervals are 0.049 0.09  
BIC = -109.18  
Fit based upon off diagonal values = 1  
Measures of factor score adequacy

	MR1	MR3	MR2
Correlation of (regression) scores with factors	0.90	0.87	0.89
Multiple R square of scores with factors	0.82	0.76	0.79
Minimum correlation of possible factor scores	0.64	0.52	0.58

F1: Traditional Pedagogy...+MultPerspectives F2: Valued-by-the-Student F3: SCRPed—the 2 items; Note: EquitableEval and MultPerspectivs have some cross-loading with first factor

```
psych::fa.diagram(pafORTH3f)
```

## Factor Analysis



### 9.10.5 Conduct an oblique rotation with a minimum of two different factor extractions

#### An oblique two factor solution

```
pafOBL2f <- psych::fa(items, nfactors = 2, rotate = "oblimin")
pafOBL2f
```

```
Factor Analysis using method = minres
Call: psych::fa(r = items, nfactors = 2, rotate = "oblimin")
Standardized loadings (pattern matrix) based upon correlation matrix
          MR1   MR2   h2   u2 com
ValObjectives      0.50  0.05  0.27  0.73 1.0
IncrUnderstanding    0.73 -0.12  0.46  0.54 1.1
IncrInterest        0.69  0.03  0.50  0.50 1.0
ClearResponsibilities 0.77  0.04  0.63  0.37 1.0
```

EffectiveAnswers	0.80	0.00	0.64	0.36	1.0
Feedback	0.64	0.18	0.54	0.46	1.2
ClearOrganization	0.85	-0.11	0.65	0.35	1.0
ClearPresentation	0.89	-0.05	0.75	0.25	1.0
MultPerspectives	0.55	0.38	0.64	0.36	1.8
InclusvClassrm	0.30	0.55	0.54	0.46	1.5
DEIntegration	-0.05	0.89	0.75	0.25	1.0
EquitableEval	0.57	0.22	0.49	0.51	1.3

	MR1	MR2
SS loadings	5.32	1.54
Proportion Var	0.44	0.13
Cumulative Var	0.44	0.57
Proportion Explained	0.78	0.22
Cumulative Proportion	0.78	1.00

With factor correlations of

	MR1	MR2
MR1	1.00	0.45
MR2	0.45	1.00

Mean item complexity = 1.2

Test of the hypothesis that 2 factors are sufficient.

df null model = 66 with the objective function = 7.27 with Chi Square = 1897.77  
df of the model are 43 and the objective function was 0.75

The root mean square of the residuals (RMSR) is 0.05

The df corrected root mean square of the residuals is 0.06

The harmonic n.obs is 267 with the empirical chi square 90.7 with prob < 0.000029  
The total n.obs was 267 with Likelihood Chi Square = 194.26 with prob < 0.0000000000000000

Tucker Lewis Index of factoring reliability = 0.873

RMSEA index = 0.115 and the 90 % confidence intervals are 0.099 0.132

BIC = -46

Fit based upon off diagonal values = 0.99

### Measures of factor score adequacy

MR.1 MR.2

Correlation of (regression) scores with factors 0.97 0.91

Multiple R square of scores with factors 0.93 0.83

Minimum correlation of possible factor scores 0.86 0.65

```
paf tableOBL2f <- psych::print.psych(pafOBL2f, cut = 0.3, sort = TRUE)
```

```
Factor Analysis using method = minres  
Call: psych::fa(r = items, nfactors = 2, rotate = "oblimin")
```

Standardized loadings (pattern matrix) based upon correlation matrix

	item	MR1	MR2	h2	u2	com
ClearPresentation	8	0.89		0.75	0.25	1.0
ClearOrganization	7	0.85		0.65	0.35	1.0
EffectiveAnswers	5	0.80		0.64	0.36	1.0
ClearResponsibilities	4	0.77		0.63	0.37	1.0
IncrUnderstanding	2	0.73		0.46	0.54	1.1
IncrInterest	3	0.69		0.50	0.50	1.0
Feedback	6	0.64		0.54	0.46	1.2
EquitableEval	12	0.57		0.49	0.51	1.3
MultPerspectives	9	0.55	0.38	0.64	0.36	1.8
ValObjectives	1	0.50		0.27	0.73	1.0
DEIIntegration	11		0.89	0.75	0.25	1.0
InclusvClassrm	10		0.55	0.54	0.46	1.5

	MR1	MR2
SS loadings	5.32	1.54
Proportion Var	0.44	0.13
Cumulative Var	0.44	0.57
Proportion Explained	0.78	0.22
Cumulative Proportion	0.78	1.00

With factor correlations of

	MR1	MR2
MR1	1.00	0.45
MR2	0.45	1.00

Mean item complexity = 1.2

Test of the hypothesis that 2 factors are sufficient.

df null model = 66 with the objective function = 7.27 with Chi Square = 1897.77  
df of the model are 43 and the objective function was 0.75

The root mean square of the residuals (RMSR) is 0.05

The df corrected root mean square of the residuals is 0.06

The harmonic n.obs is 267 with the empirical chi square 90.7 with prob < 0.000029  
The total n.obs was 267 with Likelihood Chi Square = 194.26 with prob < 0.0000000000000000

Tucker Lewis Index of factoring reliability = 0.873

RMSEA index = 0.115 and the 90 % confidence intervals are 0.099 0.132

BIC = -46

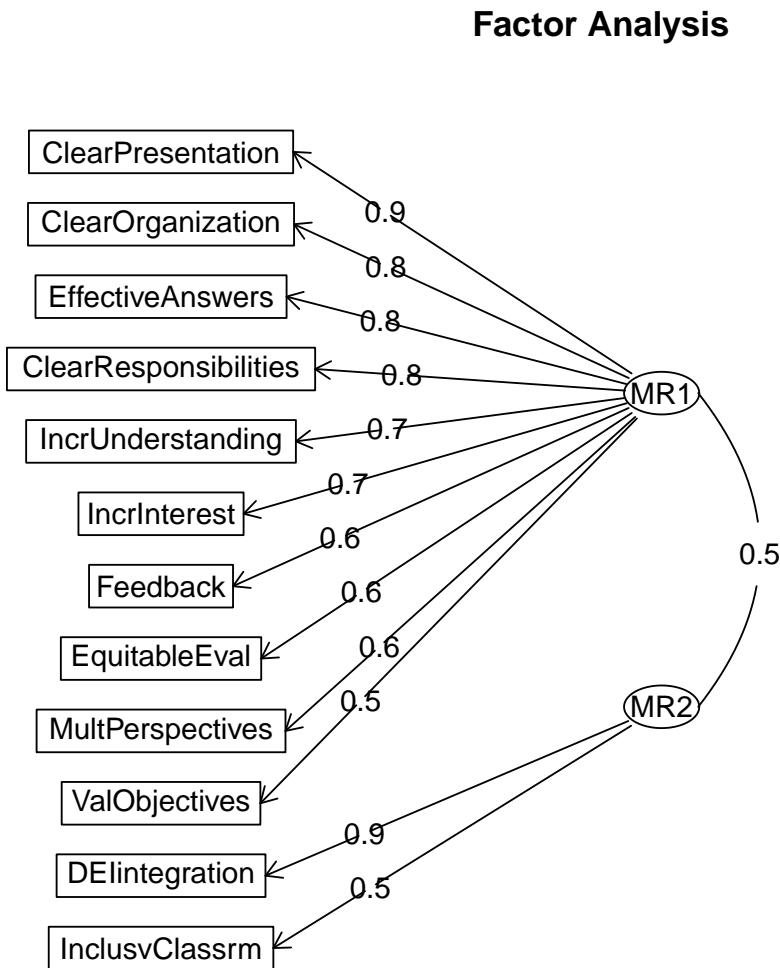
Fit based upon off diagonal values = 0.99

### Measures of factor score adequacy

	MR1	MR2
Correlation of (regression) scores with factors	0.97	0.91
Multiple R square of scores with factors	0.93	0.83
Minimum correlation of possible factor scores	0.86	0.65

Curiously, there are fewer cross-loadings. F1 has everything except the 2 SCR items which are on F2.

```
psych::fa.diagram(pafOBL2f)
```



With the curved line and value between MR1 and MR2, this figure helps make “allowance” for components to correlate, clear. There was no such path on the orthogonal figures. This is because the rotation required the factors to be uncorrelated.

### An oblique three factor solution

```
pafOBL3f <- psych::fa(items, nfactors = 3, rotate = "oblimin")
pafOBL3f
```

```
Factor Analysis using method = minres
Call: psych::fa(r = items, nfactors = 3, rotate = "oblimin")
Standardized loadings (pattern matrix) based upon correlation matrix
          MR1    MR3    MR2    h2    u2  com

```

ValObjectives	0.14	0.40	0.10	0.30	0.70	1.4
IncrUnderstanding	0.03	0.81	-0.05	0.66	0.34	1.0
IncrInterest	0.00	0.78	0.13	0.69	0.31	1.1
ClearResponsibilities	0.97	-0.11	-0.04	0.79	0.21	1.0
EffectiveAnswers	0.68	0.18	-0.01	0.64	0.36	1.1
Feedback	0.69	0.00	0.15	0.58	0.42	1.1
ClearOrganization	0.64	0.27	-0.10	0.64	0.36	1.4
ClearPresentation	0.54	0.41	-0.02	0.74	0.26	1.9
MultPerspectives	0.53	0.07	0.36	0.65	0.35	1.8
InclusvClassrm	0.12	0.21	0.58	0.56	0.44	1.3
DEIIntegration	-0.01	-0.02	0.88	0.75	0.25	1.0
EquitableEval	0.63	-0.01	0.19	0.52	0.48	1.2

	MR1	MR3	MR2
SS loadings	3.80	2.18	1.56
Proportion Var	0.32	0.18	0.13
Cumulative Var	0.32	0.50	0.63
Proportion Explained	0.50	0.29	0.21
Cumulative Proportion	0.50	0.79	1.00

With factor correlations of

MR1	MR3	MR2
MR1	1.00	0.65
MR3	0.65	1.00
MR2	0.43	0.31

Mean item complexity = 1.3

Test of the hypothesis that 3 factors are sufficient.

df null model = 66 with the objective function = 7.27 with Chi Square = 1897.77  
 df of the model are 33 and the objective function was 0.29

The root mean square of the residuals (RMSR) is 0.02

The df corrected root mean square of the residuals is 0.03

The harmonic n.obs is 267 with the empirical chi square 19.55 with prob < 0.97  
 The total n.obs was 267 with Likelihood Chi Square = 75.2 with prob < 0.000039

Tucker Lewis Index of factoring reliability = 0.954

RMSEA index = 0.069 and the 90 % confidence intervals are 0.049 0.09

BIC = -109.18

Fit based upon off diagonal values = 1

Measures of factor score adequacy

	MR1	MR3	MR2
Correlation of (regression) scores with factors	0.96	0.93	0.91
Multiple R square of scores with factors	0.92	0.86	0.83
Minimum correlation of possible factor scores	0.84	0.71	0.65

```
paf_tableOBL3f <- psych::print.psych(pafOBL3f, cut = 0.3, sort = TRUE)
```

Factor Analysis using method = minres  
 Call: psych::fa(r = items, nfactors = 3, rotate = "oblimin")  
 Standardized loadings (pattern matrix) based upon correlation matrix

	item	MR1	MR3	MR2	h2	u2	com
ClearResponsibilities	4	0.97			0.79	0.21	1.0
Feedback	6	0.69			0.58	0.42	1.1
EffectiveAnswers	5	0.68			0.64	0.36	1.1
ClearOrganization	7	0.64			0.64	0.36	1.4
EquitableEval	12	0.63			0.52	0.48	1.2
ClearPresentation	8	0.54	0.41		0.74	0.26	1.9
MultPerspectives	9	0.53		0.36	0.65	0.35	1.8
IncrUnderstanding	2		0.81		0.66	0.34	1.0
IncrInterest	3		0.78		0.69	0.31	1.1
ValObjectives	1		0.40		0.30	0.70	1.4
DEIIntegration	11			0.88	0.75	0.25	1.0
InclusvClassrm	10			0.58	0.56	0.44	1.3

	MR1	MR3	MR2
SS loadings	3.80	2.18	1.56
Proportion Var	0.32	0.18	0.13
Cumulative Var	0.32	0.50	0.63
Proportion Explained	0.50	0.29	0.21
Cumulative Proportion	0.50	0.79	1.00

With factor correlations of

MR1	MR3	MR2	
MR1	1.00	0.65	0.43
MR3	0.65	1.00	0.31
MR2	0.43	0.31	1.00

Mean item complexity = 1.3  
 Test of the hypothesis that 3 factors are sufficient.

df null model = 66 with the objective function = 7.27 with Chi Square = 1897.77  
 df of the model are 33 and the objective function was 0.29

The root mean square of the residuals (RMSR) is 0.02  
 The df corrected root mean square of the residuals is 0.03

The harmonic n.obs is 267 with the empirical chi square 19.55 with prob < 0.97  
 The total n.obs was 267 with Likelihood Chi Square = 75.2 with prob < 0.000039

Tucker Lewis Index of factoring reliability = 0.954  
 RMSEA index = 0.069 and the 90 % confidence intervals are 0.049 0.09  
 BIC = -109.18

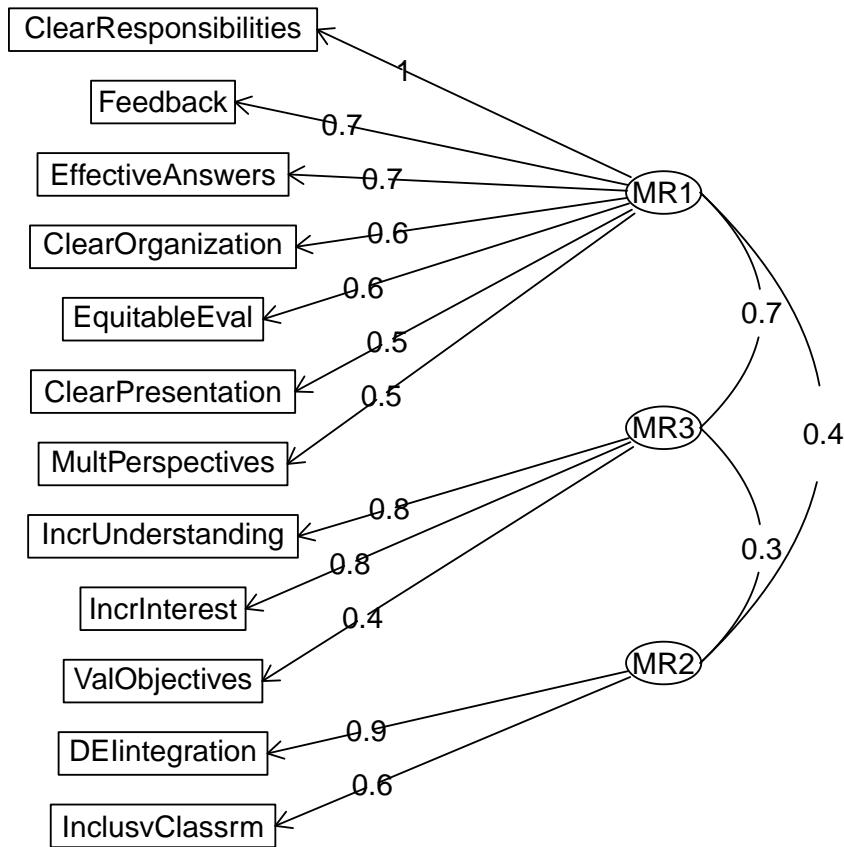
Fit based upon off diagonal values = 1

Measures of factor score adequacy

	MR1	MR3	MR2
Correlation of (regression) scores with factors	0.96	0.93	0.91
Multiple R square of scores with factors	0.92	0.86	0.83
Minimum correlation of possible factor scores	0.84	0.71	0.65

```
psych::fa.diagram(paf0BL3f)
```

## Factor Analysis



Again, pretty similar.

### 9.10.6 Determine which factor solution (e.g., orthogonal or oblique; which number of factors) you will suggest

From the oblique output we see that the correlations between the three subscales range from 0.25 to 0.58. These are high. Therefore, I will choose a 3-factor, oblique, solution.

### 9.10.7 APA style results section with table and figure of one of the solutions

The dimensionality of the 12 course evaluation items was analyzed using principal axis factoring (PAF). First, data were screened to determine the suitability of the data for this analyses. Data screening were conducted to determine the suitability of the data for this analyses. The Kaiser-Meyer-Olkin measure of sampling adequacy (KMO; Kaiser, 1970) represents the ratio of the squared correlation between variables to the squared partial correlation between variables. KMO ranges from 0.00 to 1.00; values closer to 1.00 indicate that the patterns of correlations are relatively compact, and that component analysis should yield distinct and reliable components (Field, 2012). In our dataset, the KMO value was 0.91, indicating acceptable sampling adequacy. The Bartlett's Test of Sphericity examines whether the population correlation matrix resembles an identity matrix (Field, 2012). When the  $p$  value for the Bartlett's test is  $< .05$ , we are fairly certain we have clusters of correlated variables. In our dataset,  $\chi^2(66) = 1897.77, p < 0.001$  indicating the correlations between items are sufficiently large enough for principal components analysis. The determinant of the correlation matrix alerts us to any issues of multicollinearity or singularity and should be larger than 0.00001. Our determinant was 0.0007 and, again, indicated that our data was suitable for the analysis.

Four criteria were used to determine the number of components to extract: a priori theory, the scree test, the eigenvalue-greater-than-one criteria, and the interpretability of the solution. Kaiser's eigenvalue-greater-than-one criteria suggested one component and explained 51% of the variance. The inflexion in the scree plot justified retaining one to three components. A priorili, we expected three factors – which would explain 63% of the variance. Correspondingly, we investigated two and three factor solutions with orthogonal (varimax) and oblique (oblimin) procedures. Given the significant correlations (ranging from .31 to .65) and the correspondence of items loading on the a priorili hypothesized components, we determined that an oblique, three-factor, solution was most appropriate.

The rotated solution, as shown in Table 1 and Figure 1, yielded three interpretable factors, each listed with the proportion of variance accounted for: traditional pedagogy (32%), valued-by-me (18%), and socially and culturally responsive pedagogy (13%).

Regarding the Table 1, I would include a table with ALL the values, bolding those with component membership. This is easy, though, because we can export it to a .csv file and

```
pafOBL3fb <- psych::fa(items, nfactors = 3, rotate = "oblimin")
paf_tableOBL3fb <- psych::print.psych(pafOBL3fb, sort = TRUE)
```

Factor Analysis using method = minres

```

Call: psych::fa(r = items, nfactors = 3, rotate = "oblimin")
Standardized loadings (pattern matrix) based upon correlation matrix
    item   MR1   MR3   MR2   h2   u2 com
ClearResponsibilities   4  0.97 -0.11 -0.04  0.79  0.21 1.0
Feedback                 6  0.69  0.00  0.15  0.58  0.42 1.1
EffectiveAnswers         5  0.68  0.18 -0.01  0.64  0.36 1.1
ClearOrganization        7  0.64  0.27 -0.10  0.64  0.36 1.4
EquitableEval            12 0.63 -0.01  0.19  0.52  0.48 1.2
ClearPresentation         8  0.54  0.41 -0.02  0.74  0.26 1.9
MultPerspectives          9  0.53  0.07  0.36  0.65  0.35 1.8
IncrUnderstanding        2  0.03  0.81 -0.05  0.66  0.34 1.0
IncrInterest              3  0.00  0.78  0.13  0.69  0.31 1.1
ValObjectives             1  0.14  0.40  0.10  0.30  0.70 1.4
DEIIntegration            11 -0.01 -0.02  0.88  0.75  0.25 1.0
InclusvClassrm           10  0.12  0.21  0.58  0.56  0.44 1.3

```

	MR1	MR3	MR2
SS loadings	3.80	2.18	1.56
Proportion Var	0.32	0.18	0.13
Cumulative Var	0.32	0.50	0.63
Proportion Explained	0.50	0.29	0.21
Cumulative Proportion	0.50	0.79	1.00

With factor correlations of

MR1	MR3	MR2
MR1 1.00	0.65	0.43
MR3 0.65	1.00	0.31
MR2 0.43	0.31	1.00

Mean item complexity = 1.3

Test of the hypothesis that 3 factors are sufficient.

df null model = 66 with the objective function = 7.27 with Chi Square = 1897.77  
 df of the model are 33 and the objective function was 0.29

The root mean square of the residuals (RMSR) is 0.02

The df corrected root mean square of the residuals is 0.03

The harmonic n.obs is 267 with the empirical chi square 19.55 with prob < 0.97  
 The total n.obs was 267 with Likelihood Chi Square = 75.2 with prob < 0.000039

Tucker Lewis Index of factoring reliability = 0.954

RMSEA index = 0.069 and the 90 % confidence intervals are 0.049 0.09

BIC = -109.18

Fit based upon off diagonal values = 1

Measures of factor score adequacy

	MR1	MR3	MR2
Correlation of (regression) scores with factors	0.96	0.93	0.91

Multiple R square of scores with factors	0.92	0.86	0.83
Minimum correlation of possible factor scores	0.84	0.71	0.65

```
pafOBL3fb_table <- round(pafOBL3fb$loadings, 3)
write.table(pafOBL3fb_table, file = "pafOBL3f_table.csv", sep = ",",
            row.names = FALSE)
pafOBL3fb_table
```

Loadings:

	MR1	MR3	MR2
ValObjectives	0.140	0.398	0.103
IncrUnderstanding		0.810	
IncrInterest		0.784	0.128
ClearResponsibilities	0.971	-0.108	
EffectiveAnswers	0.676	0.182	
Feedback	0.686		0.146
ClearOrganization	0.640	0.267	
ClearPresentation	0.543	0.405	
MultPerspectives	0.527		0.363
InclusvClassrm	0.121	0.207	0.580
DEIIntegration			0.880
EquitableEval	0.629		0.185

	MR1	MR3	MR2
SS loadings	3.283	1.758	1.339
Proportion Var	0.274	0.146	0.112
Cumulative Var	0.274	0.420	0.532

#### 9.10.8 Explanation to grader



# **Confirmatory Factor Analysis**



# Chapter 10

## CFA: First Order Models

### [Screencasted Lecture Link](#)

This is the first in our series on confirmatory factor analysis (CFA). In this lesson we will compare CFA to principal axis factoring (PAF) and principal components analysis (PCA). We will specify, run, and interpret first order models that are unidimensional and multidimensional. We will compare models to each other and identify key issues in model specification.

### 10.1 Navigating this Lesson

This lesson is just over two hours. I would add another two hours to work through and digest the materials.

While the majority of R objects and data you will need are created within the R script that sources the chapter, occasionally there are some that cannot be created from within the R framework. Additionally, sometimes links fail. All original materials are provided at the [GitHub site](#) that hosts the book. More detailed guidelines for ways to access all these materials are provided in the OER's [introduction](#).

#### 10.1.1 Learning Objectives

Focusing on this week's materials, make sure you can:

- Compare and contrast EFA and CFA.
- Identify the components of item-level variance in CFA.
- Specify CFA measurement models.
- Interpret fit indices (e.g., Chi-square, CFI, RMSEA).
- Interpret statistics used to compare two CFA models.

#### 10.1.2 Planning for Practice

In each of these lessons I provide suggestions for practice that allow you to select one or more problems that are graded in difficulty. The least complex is to change the random seed in the

research and rework the problem demonstrated in the lesson. The results *should* map onto the ones obtained in the lecture.

The second option involves utilizing one of the simulated datasets available in this OER. The [last lesson](#) in the OER contains three simulations that could be used for all of the statistics-based practice suggestions. Especially if you started with one of these examples in an earlier lesson, I highly recommend you continue with that.

Alternatively, Lewis and Neville's [2015] Gendered Racial Microaggressions Scale for Black Women was used in the lessons for exploratory factor analysis and Conover et al.'s [2017] Ableist Microaggressions Scale is used in the lesson on invariance testing. Both of these would be suitable for the CFA homework assignments.

As a third option, you are welcome to use data to which you have access and is suitable for CFA. This could include other simulated data, data found on an open access repository, data from the ReCentering Psych Stats survey described in the [Qualtrics lesson](#), or your own data (presuming you have permission to use it).

The suggestion for practice spans this chapter and the [next](#). From this assignment, you should plan to:

- Prepare the data frame for CFA.
- Specify and run unidimensional and single order (with correlated factors) models.
  - In the next chapter, you will add the specification, evaluation, and write-up of second-order and bifactor models.
- Narrate the adequacy of fit with  $\chi^2$ , CFI, RMSEA, SRMR
  - Write a mini-results section for each
- Compare model fit with  $\chi^2\Delta$ , AIC, and BIC.
- Write an APA style results sections with table(s) and figures.

### 10.1.3 Readings & Resources

In preparing this chapter, I drew heavily from the following resource(s). Other resources are cited (when possible, linked) in the text with complete citations in the reference list.

Byrne, B. M. (2016). Structural equation modeling with AMOS: Basic concepts, applications, and programming (3rd ed.). Routledge. <http://ebookcentral.proquest.com/lib/spu/detail.action?docID=4556523>

- Chapter 1, Structural Equation Modeling: The basics
- Chapter 3, Application 1: Testing the Factorial Validity of a Theoretical Construct (First-Order CFA Model)
- Chapter 4, Application 2: Testing the Factorial Validity of a Measurement Scale (First-Order CFA Model)

Dekay, Nicole (2021). Quick Reference Guide: The statistics for psychometrics <https://www.humananalysts.com/quick-reference-guide-the-statistics-for-psychometrics>

Kline, R. (2016). Principles and practice of structural equation modeling (Fourth ed., Methodology in the social sciences). New York: The Guilford Press.

- Chapter 9: Specification and Identification of Confirmatory Factor Analysis Models
- Chapter 13: Analysis of Confirmatory Factor Analysis Models
- Chapter 12: Global Fit Testing

Rosseel, Y. (2019). The *lavaan* tutorial. Belgium: Department of Data Analysis, Ghent University. <http://lavaan.ugent.be/tutorial/tutorial.pdf>

- “The model syntax” pp. 3 - 4
- “A first example: confirmatory factor analysis (CFA)” pp. 4-8.

#### 10.1.4 Packages

The packages used in this lesson are embedded in this code. When the hashtags are removed, the script below will (a) check to see if the following packages are installed on your computer and, if not (b) install them.

```
# will install the package if not already installed
# if(!require(lavaan)){install.packages('lavaan')}
# if(!require(lavaanPlot)){install.packages('lavaanPlot')}
# if(!require(psych)){install.packages('psych')}
# if(!require(semTable)){install.packages('semTable')}
```

## 10.2 Two Broad Categories of Factor Analysis: Exploratory and Confirmatory

Kline [2016] described confirmatory factor analysis as “exactly half that of SEM – the other half comes from regression analysis” (p. 189).

### 10.2.1 Common to Both Exploratory and Confirmatory Approaches

In both exploratory and confirmatory approaches, the variance of each indicator/item is divided into **common** and **unique** variance. When we assume that variance is 1.0, the common variance becomes the communality. If we have 8 items, we will have 8 communalities, and this represents the common variance explained by the factors or components.

- **Common variance** is shared among the indicators and serves as a basis for observed covariances among them that depart, meaningfully, from zero. We generally assume that:
  - Common variance is due to the factors.
  - There will be fewer factors than the number of indicators/items. After all, there is no point in retaining as many factors [explanatory entities] as there are entities to be explained [indicators/items].

- The proportion of total variance that is shared is the **communality** (estimated by  $h^2$ ); if  $h^2 = .70$ , then 70% of the total indicator variance is common and potentially explained by the factors.
- **Unique variance** consists of
  - **specific variance**
    - \* systematic variance that is not explained by any factor in the model,
  - **random measurement error**,
  - **method variance**, which is not represented in the figure, but could be another source of unique variance.
- In factor analysis, summing the communalities represents the total common variance (a portion of the total variance), but not the total variance.

Factor analysis, then, aligns well with classic test theory and classic approaches to understanding reliability (observed score = true score + error). The inclusion of error is illustrated well in the classic illustrations of CFA and SEM where each item/indicator includes common variance (from the factor) and error variance.

Recall that in principal components analysis (PCA is not factor analysis) one of the key distinctions is that all variance is common variance (there is no unique variance). Total common variance is equal to the total variance explained, which in turn is equal to the total variance.

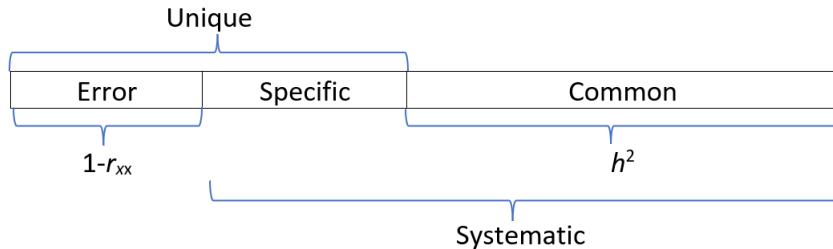


Figure 10.1: Figure illustrating the unique and common variance associated with a factor

### 10.2.2 Differences between EFA and CFA

Below are contrasts between *exploratory* and *confirmatory* factor analysis.

- **A priori specification of the number of factors**
  - EFA requires no a priori specification; prior to extraction an EFA program will extract as many factors as indicators. Typically, in subsequent analyses, the researchers specify how many factors to extract.
  - CFA requires researchers to specify the exact number of factors.
- **The degree of “exact correspondence” between indicators/items and factors/scales**

- EFA is an **unrestricted measurement model** That is, indicators/items depend on (theoretically, measure) all factors. The direct effects from factors to indicators are *pattern coefficients*. Kline [2016] says that most refer to these as *factor loadings* or just *loadings* but because he believes these terms are ambiguous, he refers to the direct effects as *pattern coefficients*. We assign them to factors based on their highest loadings (and hopefully no cross-loadings). Depending on whether we select an orthogonal or oblique relationship, correlations between factors will be permitted or suppressed.
- CFA is a **restricted measurement model**. The researcher specifies the factor(s) on which each indicator/item(s) depends (recall, the causal direction in CFA is from factor to indicators/items.)
- **Identification status:** The *identification* of a model has to do with whether it is theoretically possible for a statistics package to derive a unique set of model parameter estimates. Identification is related to model *degrees of freedom*; we will later explore under-, just-, and over-identified models. For now:
  - EFA models with multiple factors are *unidentified* because they will have more free parameters than observations. Thus, there is no unique set of statistical estimates for the multifactor EFA model, consequently this requires the rotation phase in EFA.
  - CFA models must be identified before they can be analyzed so there is only one unique set of parameter estimates. Correspondingly, there is no rotation phase in CFA.
- **Sharing variances**
  - In EFA the specific variance of each indicator is not shared with that of any other indicator.
  - In CFA, the researchers can specify if variance is shared between certain pairs of indicators (i.e., error covariances).

### 10.2.3 On the relationship between EFA and CFA

Kline [2016] admonishes us to not overinterpret the labels “exploratory” and “confirmatory”. Why?

- EFA requires no a priori hypotheses about the relationship between indicators/items and factors, but researchers often expect to specify a predetermined number of factors.
- CFA is not strictly confirmatory. After initial runs, many researchers modify models and hypotheses.

CFA is not a verification or confirmation of EFA results for the same data and number of factors. Kline [2016] does not recommend that researchers follow a model retained from EFA. Why?

- It is possible that the CFA model will be rejected. Oftentimes this is because the secondary coefficients (i.e., non-primary pattern coefficients) accounted for a significant proportion of variance in the model. When they are constrained to 0.0 in the CFA model, the model fit will suffer.
- If the CFA model is retained, then it is possible that both EFA and CFA capitalized on chance variation. Thus, if verification via CFA is desired, it should be evaluated through a replication sample.

### 10.3 Exploring a Standard CFA Model

The research vignette for today is a fairly standard CFA model.

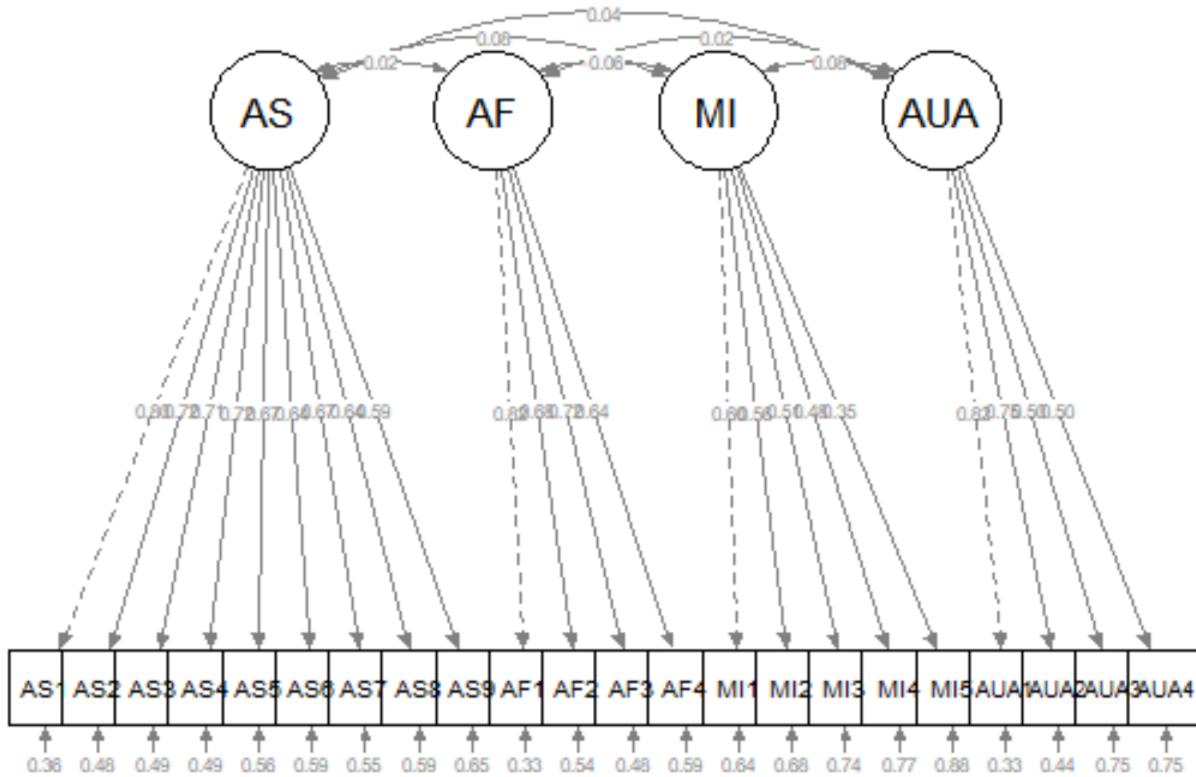


Figure 10.2: Image of the GRMSAAW represented as a standard CFA model

The image represents represents the hypothesis that  $AS_1 - AS_9$ ,  $AF_1 - AF_4$ ,  $MI_1 - MI_5$ , and  $AUA_1 - AUA_4$  measure, respectively, the AS, AF, MI, and AUA factors, which are assumed to covary. Specifically,in this model:

1. Each indicator is continuous with two causes:  $AS \rightarrow AS_1 \leftarrow E_1$ 
  - a single factor that the indicator is supposed to measure, and
  - all unique sources of influence represented by the error term
2. The error terms are independent of each other and of the factors
3. All associations are linear and the factors covary.
  - Hence, the symbol for an unanalyzed association is a solid line.
4. Each item has a single *pattern coefficient* (i.e., often more casually termed as a “factor loading”)
  - All other potential pattern coefficients are set to “0.00.” These are *hard hypotheses* and are specified by their absence (i.e., not specified in the code or in the diagram).

5. *Structure coefficients* are the Pearson correlations between factors and continuous indicators. They reflect any source of association, causal or non-causal. Sometimes the association is an undirected, back-door path. There is no pattern coefficient for  $AS_2 \leftrightarrow AF$ , but there is a connection from  $AS_2$  to  $AF$  via the  $AS \leftrightarrow AF$  covariance.
6. *Scaling constants* (aka *unit loading identification [ULI] constraints*) are necessary to scale the factors in a metric related to that of the explained (common) variance of the corresponding indicator, or *reference (marker) variable*. In the figure these are the dashed-line paths from  $AS \rightarrow AS_1$ ,  $AF \rightarrow AF_1$ ,  $MI \rightarrow MI_1$  and  $AUA \rightarrow AUA_1$ .
  - Selecting the reference marker variable is usually arbitrary and selected by the computer program as the first (or last) variable in the code/path. So long as all the indicator variables of the same factor have equally reliable scores, this works satisfactorily.
  - Additional scaling constants are found for each of the errors and indicators.

### 10.3.1 Model Identification for CFA

SEM, in general, requires that all models be *identified*. Measurement models analyzed in CFA share this requirement, but identification is more straightforward than in other models.

Standard CFA models are sufficiently identified when:

1. A single factor model has at least three indicators.
2. In a model with two or more factors, each factor has two or more indicators. There are some caveats and arguments:
  - Some recommend at least three to five indicators per factor to prevent technical problems with statistical identification.
  - In a recent SEM workshop, Todd Little indicated that optimal fit will occur when factors are *just-identified* with three items per factor.
    - Of course, three factors may be insufficient to represent the construct definition.

Identification becomes much more complicated than this, but for today's models this instruction is sufficient.

### 10.3.2 Selecting Indicators/Items for a Reflective Measurement

*Reflective measurement* is another term to describe the circumstance where latent variables are assumed to cause observed variables. Observed variables in reflective measurement are called *effect (reflective) indicators*.

- At least three for a unidimensional model; at least two per factor for a multidimensional model (but more is safer).
- The items/indicators should have reasonable internal consistency and correlate with each other.
- If the scale is multidimensional (i.e., with subscales) items should correlate more highly with other items in their factors than with items on other factors.

- Negative correlations reduce the reliability of factor measurement, so they should be reverse coded prior to analysis.
- Do not be tempted to specify a factor with indicators that do not measure something. A common mistake is to create a “background” factor and include indicators such as gender, ethnicity, and level of education. *Just what is the predicted relationship between gender and ethnicity?*

## 10.4 CFA Workflow

Below is a screenshot of a CFA workflow. The original document is located in the [GitHub site](#) that hosts the ReCentering Psych Stats: Psychometrics OER.

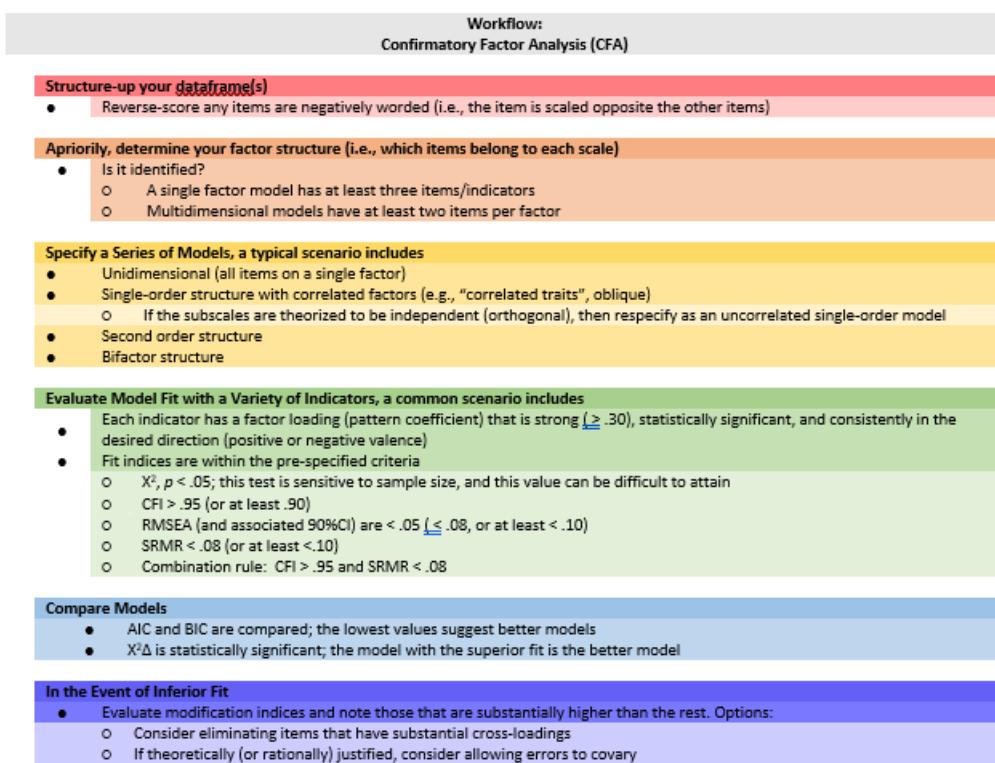


Figure 10.3: Image of a workflow for specifying and evaluating a confirmatory factor analytic model

Because the intended audience for the ReCentering Psych Stats OER is the scientist-practitioner-advocate, this lesson focuses on the workflow and decisions in straightforward CFA models. As you might guess, the details of CFA can be quite complex and require more investigation and decision-making in models that pose more complexity or empirical challenges. The following are the general steps in a CFA.

- Creating an items-only dataframe where any items are scaled in the same direction (e.g., negatively worded items are reverse-scored).
- Determining a factor structure that is *identified*.
  - A single factor (unidimensional) model has at least three items/indicators.

- Multidimensional models have at least two items per factor.
- Specify a series of models, these typically include:
  - a unidimensional model (all items on a single factor),
  - a single order structure with correlated factors,
  - a second order structure,
  - a bifactor structure.
- Evaluate model fit with a variety of indicators, including:
  - factor loadings,
  - fit indices.
- Compare models.
- In the event of poor model fit, investigate modification indices and consider respecification by:
  - eliminating items,
  - changing factor membership,
  - allowing errors to covary.

#### 10.4.1 CFA in *lavaan* Requires Fluency with the Syntax

- It's really just regression
  - tilda ( $\sim$ , *is regressed on*) is regression operator
  - place DV (y) on left of operator
  - place IVs, separate by + on the right
- f is a latent variable (LV)
- Example:  $y \sim f1 + f2 + x1 + x2$
- LVs must be *defined* by their manifest or latent indicators.
  - the special operator ( $=\sim$ , *is measured/defined by*) is used for this
  - Example:  $f1 =\sim y1 + y2 + y3$
- Variances and covariances are specified with a double tilde operator ( $\sim\sim$ , *is correlated with*)
  - Example of variance:  $y1 \sim\sim y1$  (the relationship with itself)
  - Example of covariance:  $y1 \sim\sim y2$  (relationship with another variable)
  - Example of covariance of a factor:  $f1 \sim\sim f2$

\*Intercepts ( $\sim 1$ ) for observed and LVs are simple, intercept-only regression formulas + Example of variable intercept:  $y1 \sim 1$  + Example of factor intercept:  $f1 \sim 1$

A complete lavaan model is a combination of these formula types, enclosed between single quotation marks. Readability of model syntax is improved by:

- splitting formulas over multiple lines

- using blank lines within single quote
- labeling with the hashtag

```
myModel <- '# regressions y1 + y2 ~ f1 + f2 + x1 + x2 f1 ~ f2 + f3 f2 ~ f3 + x1 + x2
```

```
# latent variable definitions
f1 =~ y1 + y2 + y3
f2 =~ y4 + y5 + y6
f3 =~ y7 + y8 + y9 + y10

# variances and covariances
y1 ~~ y1
y2 ~~ y2
f1 ~~ f2

# intercepts
y1 ~ 1
fa ~ 1
```

#### 10.4.2 Differing Factor Structures

All models worked in this lesson are *first-order* (or single-order) models; in the next lesson we extend to hierarchical and bifactor models. To provide an advanced cognitive organizer, let's take a look across the models.

Models A and B are first-order models. Note that all factors are on a single plane.

- Model A is unidimensional, each item is influenced by a single common factor and a term that includes systematic and random error. Note that there is only one *systematic* source of variance for each item, and it is from a single source.
- Model B is often referred to as a “correlated traits” model. Here, the larger construct is separated into distinct-yet-correlated elements. The variance of each item is assumed to be a weighted linear function of two or more common factors.
- Models C is a second-order factor structure. Rather than merely being correlated, factors are related because they share a common cause. In this model, the second order factor *explains* why three or more traits are correlated. Note that here is no direct relationship between the item and the target construct. Rather, the relationship between the second-order factor and each item is mediated through the primary factor (yes, an indirect effect!).
- Model D is a bifactor structure. Here each item loads on a general factor. This general factor (bottom row) reflects what is common among the items and represents the individual differences on the target dimension that a researcher is most interested in. Group factors (top row) are now specified as *orthogonal*. The group factors represent common factors measured by the items that explain item response variation not accounted for by the general factor. In some research scenarios, the group factors are termed “nuisance” dimensions. That is, that which they have in common interferes with measuring the primary target of interest.

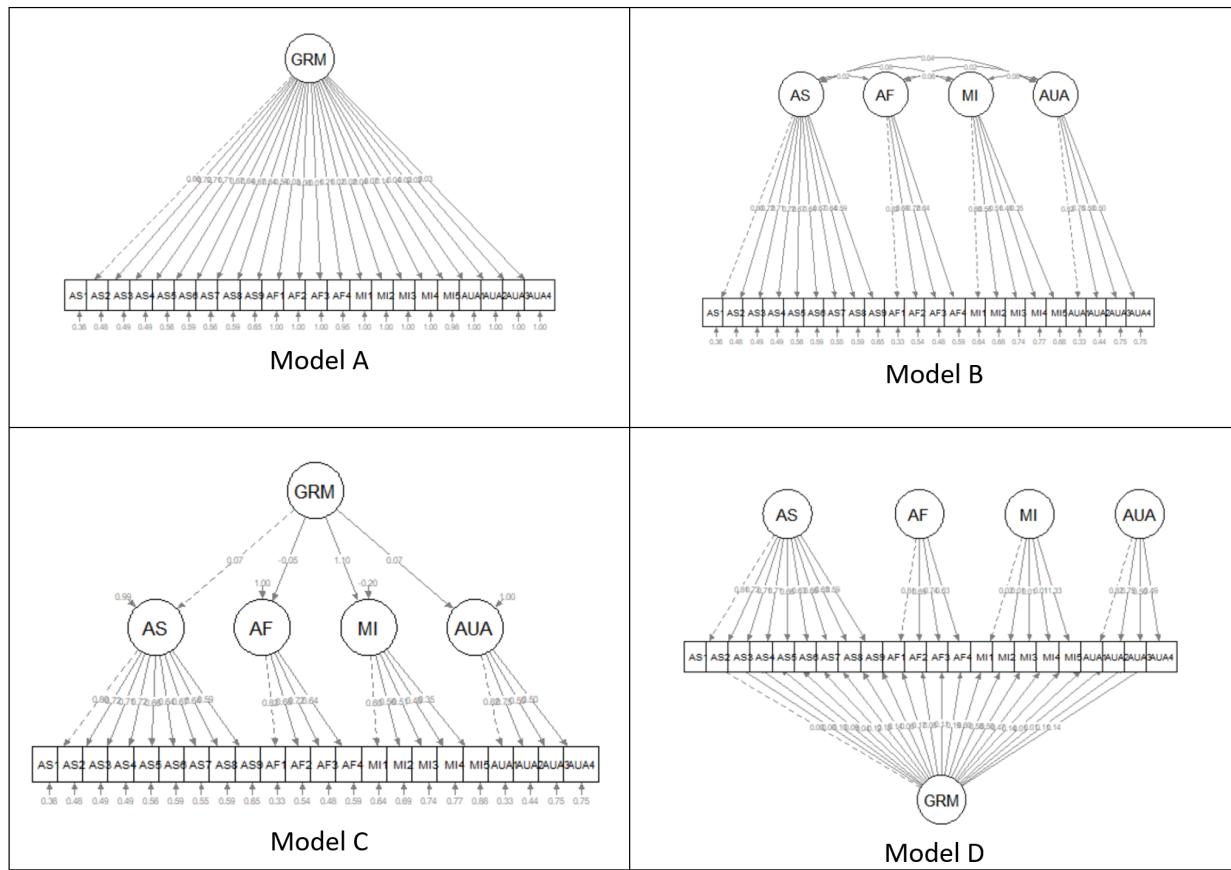


Figure 10.4: Image of first order (uncorrelated and correlated, second order, and bifactor structures)

## 10.5 Research Vignette

This lesson's research vignette emerges from Keum et al's Gendered Racial Microaggressions Scale for Asian American Women (GRMSAAW; [Keum et al., 2018]). The article reports on two separate studies that comprised the development, refinement, and psychometric evaluation of two, parallel, versions (stress appraisal, frequency) of scale. I simulated data from the final construction of the frequency version as the basis of the lecture. If the scale looks somewhat familiar it is because the authors used the Gendered Racial Microaggressions Scale for Black Women [Lewis and Neville, 2015] as a model.

Keum et al. [2018] reported support for a total scale score (22 items) and four subscales. Below, I list the four subscales, their number of items, and a single example item. At the outset, let me provide a content advisory For those who hold this particular identity (or related identities) the content in the items may be upsetting. In other lessons, I often provide a variable name that gives an indication of the primary content of the item. In the case of the GRMSAAW, I will simply provide an abbreviation of the subscale name and its respective item number. This will allow us to easily inspect the alignment of the item with its intended factor, and hopefully minimize discomfort. If you are not a member of this particular identity, I encourage you to learn about these microaggressions by reading the article in its entirety. Please do not ask members of this group to explain why these microaggressions are harmful or ask if they have encountered them.

There are 22 items on the GRMSAAW scale. Using the same item stems, the authors created two scales. One assesses frequency of the event, the second assesses the degree of stressfulness. I simulated data from the stressfulness scale. Its Likert style scaling included: 0 (*not at all stressful*), 1(*slightly stressful*), 2(*somewhat stressful*), 3(*moderately stressful*), 4(*very stressful*), and 5(*extremely stressful*).

The four factors, number of items, and sample item are as follows:

- Ascribed Submissiveness (9 items)
  - Others expect me to be submissive. (AS1)
  - Others have been surprised when I disagree with them. (AS2)
  - Others take my silence as a sign of compliance. (AS3)
  - Others have been surprised when I do things independent of my family. (AS4)
  - Others have implied that AAW seem content for being a subordinate. (AS5)
  - Others treat me as if I will always comply with their requests. (AS6)
  - Others expect me to sacrifice my own needs to take care of others (e.g., family, partner) because I am an AAW. (AS7)
  - Others have hinted that AAW are not assertive enough to be leaders. (AS8)
  - Others have hinted that AAW seem to have no desire for leadership. (AS9)
- Asian Fetishism (4 items)
  - Others express sexual interest in me because of my Asian appearance. (AF1)
  - Others take sexual interest in AAW to fulfill their fantasy. (AF2)
  - Others take romantic interest in AAW just because they never had sex with an AAW before. (AF3)
  - Others have treated me as if I am always open to sexual advances. (AF4)
- Media Invalidation (5 items)

- I see non-Asian women being casted to play female Asian characters.(MI1)
  - I rarely see AAW playing the lead role in the media. (MI2)
  - I rarely see AAW in the media. (MI3)
  - I see AAW playing the same type of characters (e.g., Kung Fu woman, sidekick, mistress, tiger mom) in the media. (MI4)
  - I see AAW characters being portrayed as emotionally distant (e.g., cold-hearted, lack of empathy) in the media. (MI5)
- Assumptions of Universal Appearance (4 items)
    - Others have talked about AAW as if they all have the same facial features (e.g., eye shape, skin tone). (AUA1)
    - Others have suggested that all AAW look alike.(AUA2)
    - Others have talked about AAW as if they all have the same body type (e.g., petite, tiny, small-chested). (AUA3)
    - Others have pointed out physical traits in AAW that do not look ‘Asian’.

Four additional scales were reported in the Keum et al. article [Keum et al., 2018]. Fortunately, I was able to find factor loadings from the original psychometric article or subsequent publications. For multidimensional scales, I assign variable names according to the scale to which the item belongs (e.g., Env42). In contrast, when subscales or short unidimensional scales were used, I assigned variable names based on item content (e.g., “blue”). In my own work, I prefer item-level names so that I can quickly see (without having to look up the item names) how the items are behaving. The scales, their original citation, and information about how I simulated data for each are listed below.

- **Racial Microaggressions Scale** (RMAS; [Torres-Harding et al., 2012]) is a 32-item scale with Likert scaling ranging from 0 (*never*) to 3 (*often/frequent*). Higher scores represent greater frequency of perceived microaggressions. I simulated data at the subscale level. The RMAS has six subscales, but only four (Invisibility, Low-Achieving/Undesirable Culture, Foreigner/Not Belonging, and Environmental Invalidation) were used in the study. Data were simulated using factor loadings (from the four factors) in the source article.
- **Schedule of Sexist Events** (SSE; [Klonoff and Landrine, 1995]) is a 20-item scale that with Likert scaling ranging from 1 (*the event has never happened to me*) to 6 (*the event happened almost all [i.e., more than 70%] of the time*). Higher scores represent greater frequency of everyday sexist events. I simulated data the subscale level. Within two larger scales (recent events, lifetime events), there are three subscales: Sexist Degradation and Its Consequences, Unfair/Sexist Events at Work/School, and Unfair Treatment in Distant and Close Relationships. Data were simulated using factor loadings from the source article.
- **PHQ-9** [Kroenke et al., 2001] is a 9-item scale with Likert scaling ranging from 0 (*not at all*) to 3 (*nearly every day*). Higher scores indicate higher levels of depression. I simulated data by estimating factor loadings from Brattmyr et al. [2022].
- **Internalized Racism in Asian American Scale** (IRAAS [Choi et al., 2017]) is a 14-item scale with Likert scaling ranging from 1 (*strongly disagree*) to 6 (*strongly agree*). Higher scores indicate greater internalized racism. Data were simulated using the factor loadings from the bifactor model in the source article.

As you consider homework options, there is sufficient simulated data to use the RMAS, SSE, or IRAAS.

Below, I walk through the data simulation. This is not an essential portion of the lesson, but I will lecture it in case you are interested. None of the items are negatively worded (relative to the other items), so there is no need to reverse-score any items.

Simulating the data involved using factor loadings, means, standard deviations, and correlations between the scales. Because the simulation will produce “out-of-bounds” values, the code below rescales the scores into the range of the Likert-type scaling and rounds them to whole values.

```
#Entering the intercorrelations, means, and standard deviations from the journal article

Keum_GRMS_generating_model <- '
  #measurement model
  General =~ .50*AS1 + .44*AS2 + .50*AS3 + .33*AS4 + .58*AS5 + .49*AS6 + .51*AS7 + .53*AS8
  AS =~ .68*AS1 + .65*AS2 + .53*AS3 + .55*AS4 + .54*AS5 + .55*AS6 + .42*AS7 + .47*AS8 + .49*AS9
  AF =~ .63*AF1 + .45*AF2 + .56*AF3 + .54*AF4
  AUA =~ .55*AUA1 + .55*AUA2 + .31*AUA3 + .31*AUA4
  MI =~ .27*MI1 + .53*MI2 + .57*MI3 + .29*MI4 + .09*MI5
  RMAS_FOR =~ .66*FOR1 + .90*FOR2 + .63*FOR4
  RMAS_LOW =~ .64*LOW22 + .54*LOW23 + .49*LOW28 + .63*LOW29 + .58*LOW30 + .67*LOW32 + .67*LOW33
  RMAS_INV =~ .66*INV33 + .70*INV39 + .79*INV40 + .71*INV41 + .71*INV47 + .61*INV49 + .61*INV50
  RMAS_ENV =~ .71*ENV42 + .70*ENV43 + .74*ENV44 + .57*ENV45 + .54*ENV46

  SSEL_Deg =~ .77*LDeg18 + .73*LDeg19 + .71*LDeg21 + .71*LDeg15 + .67*LDeg16 + .67*LDeg17
  SSEL_dRel =~ .69*LdRel4 + .68*LdRel6 + .64*LdRel7 + .64*LdRel5 + .63*LdRel11 + .49*LdRel12
  SSEL_cRel =~ .73*LcRel11 + .68*LcRel9 + .66*LcRel23
  SSEL_Work =~ .73*LWork17 + .10*LWork10 + .64*LWork2

  SSER_Deg =~ .72*RDeg15 + .71*RDeg21 + .69*RDeg18 + .68*RDeg16 + .68*RDeg13 + .65*RDeg12
  SSER_dRel =~ .74*RDeg4 + .67*RDeg6 + .64*RDeg5 + .54*RDeg7 + .51*RDeg1
  SSER_cRel =~ .69*RcRel9 + .59*RcRel11 + .53*RcRel23
  SSER_Work =~ .72*RWork10 + .67*RWork2 + .62*RWork17 + .51*RWork3

  SSE_Lifetime =~ SSEL_Deg + SSEL_dRel + SSEL_cRel + SSEL_Work
  SSE_Recent =~ SSER_Deg + SSER_dRel + SSEL_cRel + SSER_Work

  PHQ9 =~ .798*anhedonia + .425*down + .591*sleep + .913*lo_energy + .441*appetite +
    .441*anorexia + .441*restlessness + .441*fatigue + .441*headache + .441*nausea + .441*constipation +
    .441*urinary_symptoms + .441*sexual_symptoms + .441*depression + .441*suicidality + .441*other_symptoms

  gIRAAS =~ .51*SN1 + .69*SN2 + .63*SN3 + .65*SN4 + .67*WS5 + .60*WS6 + .74*WS7 + .44*WS8
  SelfNegativity =~ .60*SN1 + .50*SN2 + .63*SN3 + .43*SN4
  WeakStereotypes =~ .38*WS5 + .22*WS6 + .10*WS7 + .77*WS8 + .34*WS9 + .14*WS10
  AppearanceBias =~ .38*AB11 + .28*AB12 + .50*AB13 + .18*AB14

  #Means
  #Keum et al reported total scale scores, I divided those totals by the number of items
  AS ~ 3.25*1
  AF ~ 3.34*1
  AUA ~ 4.52
```

```
MI ~ 5.77*1
General ~ 3.81*1
RMAS_FOR ~ 3.05*1
RMAS_LOW ~ 2.6*1
RMAS_INV ~ 2.105*1
RMAS_ENV ~ 3.126*1
SSEL_Deg ~ 2.55*1
SSEL_dRel ~ 1.96*1
SSEL_cRel ~ 3.10*1
SSEL_Work ~ 1.66*1
SSER_Deg ~ 2.02*1
SSER_dRel ~ 1.592*1
SSER_cRel ~ 1.777*1
SSER_Work ~ 1.3925*1
SSER_Lifetime ~ 2.8245*1
SSER_Recent ~ 2.4875*1
PHQ9 ~ 1.836*1
gIRAAS ~ 2.246*1
```

```
#Correlations
AS ~~ .00*AF
AS ~~ .00*AUA
AS ~~ .00*MI
AS ~~ .00*General
AS ~~ .28*RMAS_FOR
AS ~~ .24*RMAS_LOW
AS ~~ .46*RMAS_INV
AS ~~ .16*RMAS_ENV
AS ~~ .40*SSE_Lifetime
AS ~~ .28*SSE_Recent
AS ~~ .15*PHQ9
AS ~~ .13*gIRAAS
```

```
AF ~~ .00*AUA
AF ~~ .00*MI
AF ~~ .00*General
AF ~~ .02*RMAS_FOR
AF ~~ .05*RMAS_LOW
AF ~~ .11*RMAS_INV
AF ~~ .07*RMAS_ENV
AF ~~ .34*SSE_Lifetime
AF ~~ .27*SSE_Recent
AF ~~ -.04*PHQ9
AF ~~ .21*gIRAAS
```

```
AUA ~~ .00*MI
AUA ~~ .00*General
```

```

AUA ~~ .18*RMAS_FOR
AUA ~~ .20*RMAS_LOW
AUA ~~ .01*RMAS_INV
AUA ~~ -.04*RMAS_ENV
AUA ~~ .02*SSE_Lifetime
AUA ~~ .92*SSE_Recent
AUA ~~ .02*PHQ9
AUA ~~ .17*gIRAAS

MI ~~ .00*General
MI ~~ -.02*RMAS_FOR
MI ~~ .08*RMAS_LOW
MI ~~ .31*RMAS_INV
MI ~~ .36*RMAS_ENV
MI ~~ .15*SSE_Lifetime
MI ~~ .08*SSE_Recent
MI ~~ -.05*PHQ9
MI ~~ -.03*gIRAAS

General ~~ .34*RMAS_FOR
General ~~ .63*RMAS_LOW
General ~~ .44*RMAS_INV
General ~~ .45*RMAS_ENV
General ~~ .54*SSE_Lifetime
General ~~ .46*SSE_Recent
General ~~ .31*PHQ9
General ~~ -.06*gIRAAS

RMAS_FOR ~~ .57*RMAS_LOW
RMAS_FOR ~~ .56*RMAS_INV
RMAS_FOR ~~ .37*RMAS_ENV
RMAS_FOR ~~ .33*SSE_Lifetime
RMAS_FOR ~~ .25*SSE_Recent
RMAS_FOR ~~ .10*PHQ9
RMAS_FOR ~~ .02*gIRAAS

RMAS_LOW ~~ .69*RMAS_INV
RMAS_LOW ~~ .48*RMAS_ENV
RMAS_LOW ~~ .67*SSE_Lifetime
RMAS_LOW ~~ .57*SSE_Recent
RMAS_LOW ~~ .30*PHQ9
RMAS_LOW ~~ .16*gIRAAS

RMAS_INV ~~ .59*RMAS_ENV
RMAS_INV ~~ .63*SSE_Lifetime
RMAS_INV ~~ .52*SSE_Recent

```

```

RMAS_INV ~~ .32*PHQ9
RMAS_INV ~~ .23*gIRAAS

RMAS_ENV ~~ .46*SSE_Lifetime
RMAS_ENV ~~ .31*SSE_Recent
RMAS_ENV ~~ .11*PHQ9
RMAS_ENV ~~ .07*gIRAAS

SSE_Lifetime ~~ .83*SSE_Recent
SSE_Lifetime ~~ .30*PHQ9
SSE_Lifetime ~~ .14*gIRAAS

SSE_Recent ~~ .30*PHQ9
SSE_Recent ~~ .20*gIRAAS

PHQ9 ~~ .18*gIRAAS

#Correlations between SES scales from the Klonoff and Landrine article
#Note that in the article the factor orders were reversed
SSEL_Deg ~~ .64*SSEL_dRel
SSEL_Deg ~~ .61*SSEL_cRel
SSEL_Deg ~~ .50*SSEL_Work
SSEL_dRel ~~ .57*SSEL_cRel
SSEL_dRel ~~ .57*SSEL_Work
SSEL_cRel ~~ .47*SSEL_Work

SSER_Deg ~ .54*SSER_dRel
SSER_Deg ~ .54*SSER_Work
SSER_Deg ~ .59*SSER_cRel
SSER_dRel ~ .56*SSER_Work
SSER_dRel ~ .46*SSER_cRel
SSER_Work ~ .43*SSER_cRel

SSE_Lifetime ~ .75*SSE_Recent

'

set.seed(240311)
dfGRMSAAW <- lavaan::simulateData(model = Keum_GRMS_generating_model,
                                     model.type = "sem",
                                     meanstructure = T,
                                     sample.nobs=304,
                                     standardized=FALSE)

#used to retrieve column indices used in the rescaling script below
col_index <- as.data.frame(colnames(dfGRMSAAW))

```

```

#The code below loops through each column of the dataframe and assigns the scaling accordingly
#Rows 1 thru 22 are the GRMS items
#Rows 23 thru 47 are the RMAS
#Rows 48 thru 87 are the SSE
#Rows 88 thru 96 are the PHQ9
#Rows 97 thru 110 are the IRAAS
#Rows 111 thru 112 are scale scores for SSE

for(i in 1:ncol(dfGRMSAAW)){
  if(i >= 1 & i <= 22){
    dfGRMSAAW[,i] <- scales::rescale(dfGRMSAAW[,i], c(0, 5))
  }
  if(i >= 23 & i <= 47){
    dfGRMSAAW[,i] <- scales::rescale(dfGRMSAAW[,i], c(0, 3))
  }
  if(i >= 48 & i <= 87){
    dfGRMSAAW[,i] <- scales::rescale(dfGRMSAAW[,i], c(1, 6))
  }
  if(i >= 88 & i <= 96){
    dfGRMSAAW[,i] <- scales::rescale(dfGRMSAAW[,i], c(0, 3))
  }
  if(i >= 97 & i <= 110){
    dfGRMSAAW[,i] <- scales::rescale(dfGRMSAAW[,i], c(1, 6))
  }
}

#rounding to integers so that the data resembles that which was collected
library(tidyverse)
dfGRMSAAW <- dfGRMSAAW %>% round(0)

#quick check of my work
#psych::describe(dfGRMSAAW)

```

The optional script below will let you save the simulated data to your computing environment as either an .rds object or a .csv file.

An .rds file preserves all formatting to variables prior to the export and re-import. For the purpose of this chapter, you don't need to do either. That is, you can re-simulate the data each time you work the problem.

```

# to save the df as an .rds (think 'R object') file on your computer;
# it should save in the same file as the .rmd file you are working
# with saveRDS(dfGRMSAAW, 'dfGRMSAAW.rds') bring back the simulated
# dat from an .rds file dfGRMSAAW <- readRDS('dfGRMSAAW.rds')

```

If you save the .csv file (think "Excel lite") and bring it back in, you will lose any formatting (e.g., ordered factors will be interpreted as character variables).

```
# write the simulated data as a .csv write.table(dfGRMSAAW,
# file='dfGRMSAAW.csv', sep=',', col.names=TRUE, row.names=FALSE)
# bring back the simulated dat from a .csv file dfGRMSAAW <- read.csv
# ('dfGRMSAAW.csv', header = TRUE)
```

### 10.5.1 Modeling the GRMSAAW as Unidimensional

Let's start simply, taking the GRMSAAW data and seeing about its fit as a unidimensional instrument. In fact, even when measures are presumed to be multi-dimensional, it is common to begin with a unidimensional assessment. Here's why:

- Operationally, it's a check to see that data, script, and so forth. are all working.
- If you can't reject a single-factor model (e.g., if there is a strong support for such), then it makes little sense to evaluate models with more factors [Kline, 2016].

Considerations for the *lavaan* code include:

- GRMSAAW is a latent variable and can be named anything. We know this because it is followed by: =~
- All the items follow and are “added” with the plus sign
  - Don't let this fool you...the assumption behind SEM/CFA is that the LV *causes* the score on the item/indicator. Recall, item/indicator scores are influenced by the LV and error.
- The entire model is enclosed in tic marks (‘ and ’)

```
grmsAAWmod1 <- "GRMSAAW =~ AS1 + AS2 + AS3 + AS4 + AS5 + AS6 + AS7 + AS8 + AS9 + AF1 + AF2 + AF3"
```

The object representing the model is then included in the *lavaan::cfa()* along with the dataset.

We can ask for a summary of the object representing the results.

```
set.seed(240311)
grmsAAWfit <- lavaan::cfa(grmsAAWmod1, data = dfGRMSAAW)
lavaan::summary(grmsAAWfit, fit.measures = TRUE, standardized = TRUE,
rsquare = TRUE)
```

*lavaan* 0.6.17 ended normally after 29 iterations

Estimator	ML
Optimization method	NLMINB
Number of model parameters	44
Number of observations	304
Model Test User Model:	

Test statistic	444.451
Degrees of freedom	209
P-value (Chi-square)	0.000

Model Test Baseline Model:

Test statistic	1439.317
Degrees of freedom	231
P-value	0.000

User Model versus Baseline Model:

Comparative Fit Index (CFI)	0.805
Tucker-Lewis Index (TLI)	0.785

Loglikelihood and Information Criteria:

Loglikelihood user model (H0)	-8387.014
Loglikelihood unrestricted model (H1)	-8164.789
Akaike (AIC)	16862.028
Bayesian (BIC)	17025.577
Sample-size adjusted Bayesian (SABIC)	16886.032

Root Mean Square Error of Approximation:

RMSEA	0.061
90 Percent confidence interval - lower	0.053
90 Percent confidence interval - upper	0.069
P-value H_0: RMSEA <= 0.050	0.012
P-value H_0: RMSEA >= 0.080	0.000

Standardized Root Mean Square Residual:

SRMR	0.067
------	-------

Parameter Estimates:

Standard errors	Standard
Information	Expected
Information saturated (h1) model	Structured

Latent Variables:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
GRMSAAW =~						
AS1	1.000				0.491	0.535
AS2	1.069	0.152	7.012	0.000	0.525	0.520

AS3	1.024	0.143	7.151	0.000	0.503	0.535
AS4	0.909	0.137	6.649	0.000	0.446	0.482
AS5	1.177	0.154	7.634	0.000	0.578	0.590
AS6	0.721	0.108	6.658	0.000	0.354	0.483
AS7	0.914	0.137	6.693	0.000	0.449	0.487
AS8	0.927	0.137	6.765	0.000	0.455	0.494
AS9	0.735	0.117	6.262	0.000	0.361	0.445
AF1	0.675	0.125	5.410	0.000	0.332	0.370
AF2	0.975	0.144	6.755	0.000	0.479	0.493
AF3	0.555	0.120	4.637	0.000	0.272	0.308
AF4	0.851	0.141	6.042	0.000	0.418	0.425
MI1	0.744	0.120	6.182	0.000	0.365	0.438
MI2	0.641	0.122	5.252	0.000	0.315	0.357
MI3	0.860	0.146	5.907	0.000	0.422	0.413
MI4	0.601	0.130	4.614	0.000	0.295	0.307
MI5	0.655	0.122	5.356	0.000	0.322	0.365
AUA1	0.825	0.144	5.740	0.000	0.405	0.398
AUA2	0.878	0.132	6.659	0.000	0.431	0.483
AUA3	0.714	0.118	6.058	0.000	0.350	0.426
AUA4	1.060	0.146	7.262	0.000	0.520	0.547

Variances:

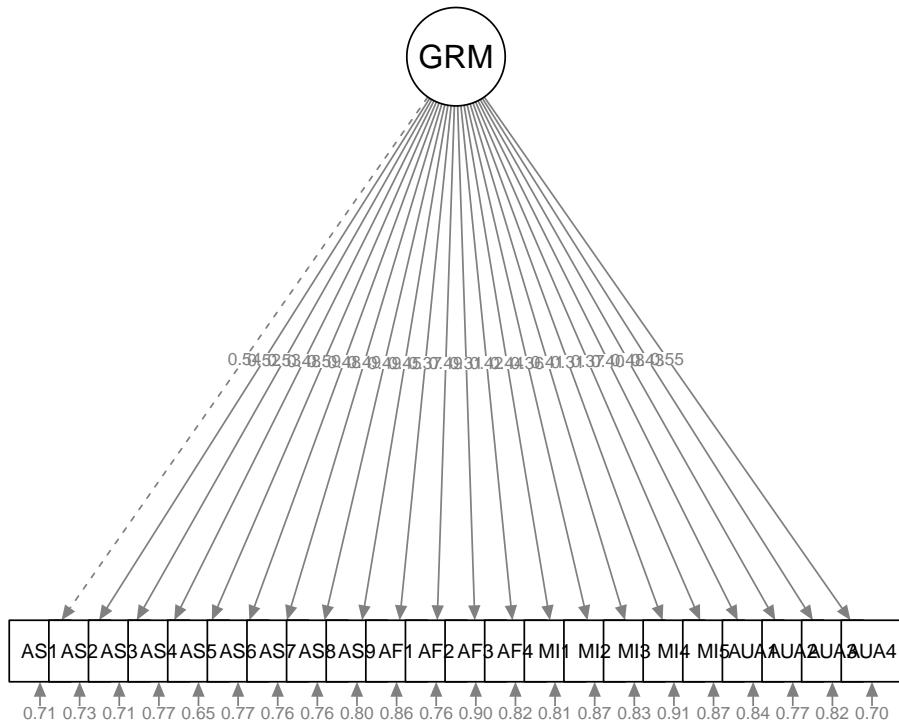
	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
.AS1	0.600	0.052	11.500	0.000	0.600	0.713
.AS2	0.744	0.064	11.565	0.000	0.744	0.730
.AS3	0.631	0.055	11.503	0.000	0.631	0.714
.AS4	0.657	0.056	11.704	0.000	0.657	0.767
.AS5	0.624	0.056	11.225	0.000	0.624	0.651
.AS6	0.412	0.035	11.701	0.000	0.412	0.766
.AS7	0.648	0.055	11.689	0.000	0.648	0.763
.AS8	0.641	0.055	11.663	0.000	0.641	0.756
.AS9	0.528	0.045	11.820	0.000	0.528	0.802
.AF1	0.693	0.058	12.002	0.000	0.693	0.863
.AF2	0.714	0.061	11.666	0.000	0.714	0.757
.AF3	0.707	0.058	12.113	0.000	0.707	0.905
.AF4	0.794	0.067	11.875	0.000	0.794	0.820
.MI1	0.564	0.048	11.841	0.000	0.564	0.809
.MI2	0.678	0.056	12.028	0.000	0.678	0.873
.MI3	0.870	0.073	11.906	0.000	0.870	0.830
.MI4	0.839	0.069	12.115	0.000	0.839	0.906
.MI5	0.672	0.056	12.011	0.000	0.672	0.866
.AUA1	0.872	0.073	11.941	0.000	0.872	0.842
.AUA2	0.610	0.052	11.700	0.000	0.610	0.766
.AUA3	0.553	0.047	11.871	0.000	0.553	0.818
.AUA4	0.634	0.055	11.448	0.000	0.634	0.701
GRMSAAW	0.241	0.051	4.694	0.000	1.000	1.000

R-Square:

	Estimate
AS1	0.287
AS2	0.270
AS3	0.286
AS4	0.233
AS5	0.349
AS6	0.234
AS7	0.237
AS8	0.244
AS9	0.198
AF1	0.137
AF2	0.243
AF3	0.095
AF4	0.180
MI1	0.191
MI2	0.127
MI3	0.170
MI4	0.094
MI5	0.134
AUA1	0.158
AUA2	0.234
AUA3	0.182
AUA4	0.299

I find it helpful to immediately plot what we did. A quick look alerts me to errors.

```
semPlot::semPaths(grmsAAW1fit, layout = "tree", style = "lisrel", what = "col",
whatLabels = "stand")
```



#### #### Interpreting the Output

With a quick look at the plot, let's work through the results. Rosseel's (2019) *lavaan* tutorial is a useful resource in walking through the output.

The *header* is the first few lines of the information. It contains:

- the *lavaan* version number (0.6-17 that I'm using on 4/12/24),
- maximum likelihood (ML) was used as the estimator,
- confirmation that the specification converged normally after 28 iterations,
- indication that 304 cases were used in this analysis (this would be less if some were skipped because of missing data),
- results including the model user test statistic, df, and corresponding p value:  $\chi^2(209) = 444.451, p < .001$ .

Model Test User Model:

**Fit statistics** are included in the second section. They are only shown when the argument “fit.measures = TRUE” is in the script. Standardized values are not the default, they require the argument, “standardized = TRUE”. We'll come back to these shortly...

*Parameter estimates* is the last section.

For now, we are interested in the “Latent Variables” section.

- *Estimate* contains the estimated or fixed parameter value for each model parameter;
- *Std. err* is the standard error for each estimated parameter;
- *Z-value* is the Wald statistic (the parameter divided by its SE)
- $P(>/z|)$  is the p value for testing the null hypothesis that the parameter equals zero in the population
- *Std.lv* standardizes only the LVs
- *Std.all* both latent and observed variables are standardized; this is considered the “completely standardized solution”

Note that item AS1 might seem incomplete – there is only a 1.000 and a value for the Std.lv. Recall we used this to scale the single factor by fixing its value to 1.000. Coefficients that are fixed to 1.0 to scale a factor have no standard errors and therefore no significance test.

The SE and associated  $p$  values are associated with the unstandardized estimates. Intuitively, it is easiest for me to understand the relative magnitude of the pattern coefficients by looking at the *Std.all* column. We can see that the items associated with what we will soon define as the AS factor are all strong and positive. The remaining items have variable loadings with many of the being quite low, non-significant, and even negatively balanced.

Let’s examine to the middle set metrics which assess *global fit*.

CFA falls into a *modeling* approach to evaluating results. While it provides some flexibility (we get away from the strict, NHST approach of  $p < .05$ ) there is greater interpretive ambiguity.

Fit statistics tend to be clustered together based on their approach to summarizing the *goodness* or *badness* of fit.

#### 10.5.1.1 Model Test *User Model*:

The chi-square statistic that evaluates the *exact-fit hypothesis* that there is no difference between the covariances predicted by the model, given the parameter estimates, and the population covariance matrix. Rejecting the hypothesis says that,

- the data contain covariance information that speak against the model, and
- the researcher should explain model-data discrepancies that exceed those expected by sampling error.

Traditional interpretation of the chi-square is an *accept-support test* where the null hypothesis represents the researchers’ belief that the model is correct. This means that the absence of statistical significance ( $p > .05$ ) that supports the model. This is backwards from our usual *reject-support test* approach.

The  $\chi^2$  is frequently criticized:

- *Accept-support test* approaches are logically weaker because the failure to disprove an assertion (the exact-fit hypothesis) does not prove that the assertion is true.

- Too small a sample size (low power) makes it more likely that the model will be retained.
- CFA/SEM requires large samples and so the  $\chi^2$  is frequently statistically significant; this frequently results in rejection of the researchers' model.

Kline [2016] recommends that we treat the  $\chi^2$  like a smoke alarm – if the alarm sounds, there may or may not be a fire (a serious model-data discrepancy), but we should treat the alarm seriously and further inspect issues of fit.

For our unidimensional GRMSAAW CFA  $\chi^2(209) = 444.451, p < .001$ , the significant  $p$  value is not what we want because it says that our specified model is different than the covariances in the model.

### 10.5.1.2 Model Test *Baseline Model*

This model is the *independence* model. That is, there is complete independence of all variables in the model (i.e., in which all correlations among variables are zero). This is the most restricted model. It is typical for chi-square values to be quite high (as it is in our example: 1439.317). On its own, this model is not useful to us. It is used, though, in comparisons of *incremental fit*.

### 10.5.1.3 Incremental Fit Indices (User versus Baseline Models)

Incremental fit indices ask the question, how much better is the fit of our specified model to the data than the baseline model (where it is assumed no relations between the variables).

The Comparative Fit Index (CFI) and Tucker-Lewis Index (TLI) are *goodness of fit* statistics, ranging from 0 to 1.0 where 1.0 is best.

**CFI:** compares the amount of departure from close fit for the researcher's model against that of the independence/baseline (null) model.

$$CFI = 1 - \frac{\hat{\Delta}_M}{\hat{\Delta}_B}$$

Where there is no departure from close fit, then CFI will equal 1.0. We interpret the value of the CFI as a percent of how much better the researcher's model is than the baseline model. While 81% sounds like an improvement – Hu and Bentler (1999) stated that “acceptable fit” is achieved when the  $CFI \geq .95$  and  $SRMR \leq .08$ ; the **combination rule**. It is important to note that later simulation studies have not supported those thresholds.

**TLI:** aka the **non-normed fit index (NNFI)** controls for  $df_M$  from the researcher's model and  $df_B$  from the baseline model. As such, it imposes a greater relative penalty for model complexity than the CFI. The TLI is a bit unstable in that the values can exceed 1.0.

Because the two measures are so related, only one should be reported (I typically see the CFI).

For our unidimensional GRMSAAW CFA, CFI = .805 and TLI = .785. While these predict around 81% better than the baseline/independence model, it does not come close to the standard of  $\geq .95$ .

#### 10.5.1.4 Loglikelihood and Information Criteria

The **Akaike Information Criterion (AIC)** and the **Bayesian Information Criterion (BIC)** utilize an information theory approach to data analysis by combining statistical estimation and model selection into a single framework. The BIC augments the AIC by taking sample size into consideration.

The AIC and BIC are usually used to select among competing nonhierarchical models and are only used in comparison with each other. Thus, our current values of 16862.028 (AIC) and 17025.577 (BIC) are meaningless on their own. The model with the smallest values of the predictive fit indices is chosen as the one that is most likely to replicate. It means that this model has relatively better fit and fewer free parameters than competing models.

Later in the lesson we will return to these values to compare a correlated, four-factor solution with this unidimensional model.

#### 10.5.1.5 Root Mean Square Error of Approximation

The RMSEA is an absolute fit index scaled as a *badness-of-fit* statistic where a value of 0.00 is the best fit. The RMSEA favors models with more degrees of freedom and larger sample sizes. A unique aspect of the RMSEA is its 90% confidence interval.

While there is chatter/controversy about what constitutes an acceptable value, there is general consensus that  $RMSEA \geq .10$  points to serious problems. An  $RMSEA \leq .05$  is desired. In evaluating the RMSEA, we need to monitor the upper bound of the confidence interval to see that it isn't sneaking into the danger zone.

For our unidimensional GRMSAAW CFA,  $RMSEA = 0.061$ , 90% CI(0.053, 0.069). This value is within the range of acceptability.

#### 10.5.1.6 Standardized Root Mean Square Residual

The SRMR is an absolute fit index that is a *badness-of-fit* statistic (i.e., perfect model fit is when the value = 0.00 and increasingly higher values indicate the “badness”).

The SRMR is a standardized version of the **root mean square residual (RMR)**, which is a measure of the mean absolute covariance residual. Standardizing the value facilitates interpretation.

Poor fit is indicated when  $SRMR \geq .10$ .

Recall, Hu and Bentler's **combination rule** (which is somewhat contested) suggested that the SRMR be interpreted along with the CFI such that:  $CFI \geq .95$  and  $SRMR \leq .08$ .

For our unidimensional GRMSAAW CFA,  $SRMR = 0.067$ .

Inspecting the residuals (we look for relatively large values) may help understand the source of poor fit, so let's do that.

```
lavaan::fitted(grmsAAW1fit)
```

```
$cov
```

	AS1	AS2	AS3	AS4	AS5	AS6	AS7	AS8	AS9	AF1	AF2	AF3
AS1	0.841											
AS2	0.258	1.020										
AS3	0.247	0.264	0.883									
AS4	0.219	0.234	0.224	0.856								
AS5	0.284	0.303	0.290	0.258	0.958							
AS6	0.174	0.186	0.178	0.158	0.205	0.537						
AS7	0.220	0.235	0.225	0.200	0.259	0.159	0.849					
AS8	0.223	0.239	0.229	0.203	0.263	0.161	0.204	0.849				
AS9	0.177	0.189	0.181	0.161	0.209	0.128	0.162	0.164	0.658			
AF1	0.163	0.174	0.167	0.148	0.192	0.117	0.149	0.151	0.120	0.803		
AF2	0.235	0.251	0.241	0.214	0.277	0.170	0.215	0.218	0.173	0.159	0.943	
AF3	0.134	0.143	0.137	0.122	0.157	0.096	0.122	0.124	0.098	0.090	0.130	0.781
AF4	0.205	0.219	0.210	0.187	0.242	0.148	0.188	0.190	0.151	0.139	0.200	0.114
MI1	0.179	0.192	0.184	0.163	0.211	0.129	0.164	0.166	0.132	0.121	0.175	0.100
MI2	0.154	0.165	0.158	0.140	0.182	0.111	0.141	0.143	0.114	0.104	0.151	0.086
MI3	0.207	0.222	0.212	0.189	0.244	0.150	0.190	0.192	0.152	0.140	0.202	0.115
MI4	0.145	0.155	0.148	0.132	0.170	0.104	0.132	0.134	0.106	0.098	0.141	0.080
MI5	0.158	0.169	0.162	0.144	0.186	0.114	0.144	0.146	0.116	0.107	0.154	0.088
AUA1	0.199	0.213	0.204	0.181	0.234	0.143	0.182	0.184	0.146	0.134	0.194	0.110
AUA2	0.212	0.226	0.217	0.192	0.249	0.153	0.193	0.196	0.156	0.143	0.207	0.118
AUA3	0.172	0.184	0.176	0.156	0.202	0.124	0.157	0.159	0.126	0.116	0.168	0.095
AUA4	0.255	0.273	0.261	0.232	0.301	0.184	0.233	0.237	0.188	0.173	0.249	0.142
	AF4	MI1	MI2	MI3	MI4	MI5	AUA1	AUA2	AUA3	AUA4		
AS1												
AS2												
AS3												
AS4												
AS5												
AS6												
AS7												
AS8												
AS9												
AF1												
AF2												
AF3												
AF4	0.968											
MI1	0.153	0.697										
MI2	0.132	0.115	0.777									
MI3	0.177	0.154	0.133	1.048								
MI4	0.123	0.108	0.093	0.125	0.926							
MI5	0.134	0.118	0.101	0.136	0.095	0.775						
AUA1	0.169	0.148	0.127	0.171	0.119	0.130	1.036					
AUA2	0.180	0.158	0.136	0.182	0.127	0.139	0.175	0.796				
AUA3	0.146	0.128	0.110	0.148	0.103	0.113	0.142	0.151	0.676			
AUA4	0.217	0.190	0.164	0.220	0.153	0.167	0.211	0.224	0.182	0.904		

```
# lavaan::residuals(grmsAAW1fit, type = 'raw')
# lavaan::residuals(grmsAAW1fit, type = 'standardized')

# will hashtag out for knitted file
lavaan::residuals(grmsAAW1fit, type = "cor")
```

```
$type
[1] "cor.bollen"

$cov
      AS1     AS2     AS3     AS4     AS5     AS6     AS7     AS8     AS9     AF1
AS1  0.000
AS2  0.101  0.000
AS3  0.059  0.096  0.000
AS4  0.073  0.122  0.062  0.000
AS5  0.137  0.055 -0.021  0.060  0.000
AS6  0.024  0.096  0.000 -0.043  0.047  0.000
AS7 -0.022  0.042  0.052  0.057  0.027 -0.036  0.000
AS8  0.003  0.091  0.037  0.023  0.086  0.042  0.055  0.000
AS9  0.080  0.109  0.017  0.067  0.072  0.095  0.013  0.064  0.000
AF1 -0.053 -0.088  0.062 -0.007 -0.057 -0.078 -0.070 -0.008 -0.050  0.000
AF2 -0.040 -0.134 -0.057  0.022 -0.053 -0.064 -0.050  0.008 -0.006  0.154
AF3 -0.027 -0.117  0.041 -0.088 -0.042 -0.074 -0.038 -0.018  0.015  0.222
AF4 -0.093 -0.059 -0.006  0.007 -0.016 -0.063 -0.015 -0.066 -0.079  0.203
MI1 -0.092 -0.091 -0.074 -0.069 -0.096  0.015 -0.015 -0.066 -0.100  0.082
MI2 -0.117 -0.001 -0.063 -0.049 -0.035 -0.005  0.040 -0.067 -0.089 -0.084
MI3 -0.080 -0.025 -0.030 -0.065 -0.045  0.013 -0.088  0.034  0.007 -0.007
MI4 -0.103 -0.111 -0.018 -0.153 -0.088 -0.014  0.070 -0.068 -0.026 -0.012
MI5  0.006 -0.046 -0.042  0.021 -0.028 -0.023  0.003 -0.046 -0.097 -0.015
AUA1 -0.075 -0.114 -0.017 -0.129 -0.057 -0.008 -0.016 -0.015 -0.119 -0.118
AUA2 -0.025 -0.059 -0.068 -0.101 -0.031  0.015 -0.040 -0.055 -0.074 -0.005
AUA3 -0.003 -0.028 -0.019 -0.060 -0.075  0.031 -0.049 -0.125 -0.022 -0.007
AUA4 -0.006 -0.078 -0.042  0.000 -0.013 -0.061  0.025 -0.065 -0.073  0.031
      AF2     AF3     AF4     MI1     MI2     MI3     MI4     MI5     AUA1     AUA2
AS1
AS2
AS3
AS4
AS5
AS6
AS7
AS8
AS9
AF1
AF2  0.000
AF3  0.056  0.000
AF4  0.138  0.088  0.000
```

```

MI1  0.133  0.019  0.064  0.000
MI2 -0.070 -0.006  0.063  0.115  0.000
MI3  0.016 -0.002 -0.002  0.130  0.223  0.000
MI4  0.043  0.017  0.025  0.141  0.111  0.157  0.000
MI5  0.035  0.081  0.012 -0.030  0.041 -0.015  0.119  0.000
AUA1 0.025  0.024 -0.024  0.084  0.113  0.059  0.045  0.022  0.000
AUA2 0.006 -0.006  0.022  0.029  0.055 -0.013  0.003  0.071  0.184  0.000
AUA3 0.029  0.011 -0.064 -0.008  0.030 -0.016  0.078  0.057  0.141  0.115
AUA4 0.005  0.005  0.017  0.072 -0.016 -0.029  0.017  0.002  0.100  0.098
      AUA3   AUA4

AS1
AS2
AS3
AS4
AS5
AS6
AS7
AS8
AS9
AF1
AF2
AF3
AF4
MI1
MI2
MI3
MI4
MI5
AUA1
AUA2
AUA3  0.000
AUA4  0.092  0.000

```

```
lavaan:::modindices(grmsAAW1fit)
```

	lhs	op	rhs	mi	epc	sepc.lv	sepc.all	sepc.nox
46	AS1	~~	AS2	6.859	0.108	0.108	0.162	0.162
47	AS1	~~	AS3	2.401	0.059	0.059	0.096	0.096
48	AS1	~~	AS4	3.353	0.070	0.070	0.112	0.112
49	AS1	~~	AS5	14.871	0.148	0.148	0.243	0.243
50	AS1	~~	AS6	0.381	0.019	0.019	0.038	0.038
51	AS1	~~	AS7	0.306	-0.021	-0.021	-0.034	-0.034
52	AS1	~~	AS8	0.005	0.003	0.003	0.004	0.004
53	AS1	~~	AS9	3.823	0.067	0.067	0.119	0.119
54	AS1	~~	AF1	1.511	-0.048	-0.048	-0.074	-0.074
55	AS1	~~	AF2	1.035	-0.041	-0.041	-0.062	-0.062
56	AS1	~~	AF3	0.374	-0.024	-0.024	-0.037	-0.037

57	AS1	~~	AF4	5.081	-0.094	-0.094	-0.137	-0.137
58	AS1	~~	MI1	5.049	-0.080	-0.080	-0.137	-0.137
59	AS1	~~	MI2	7.368	-0.104	-0.104	-0.164	-0.164
60	AS1	~~	MI3	3.666	-0.084	-0.084	-0.116	-0.116
61	AS1	~~	MI4	5.429	-0.099	-0.099	-0.140	-0.140
62	AS1	~~	MI5	0.018	0.005	0.005	0.008	0.008
63	AS1	~~	AUA1	3.179	-0.078	-0.078	-0.108	-0.108
64	AS1	~~	AUA2	0.407	-0.024	-0.024	-0.039	-0.039
65	AS1	~~	AUA3	0.004	-0.002	-0.002	-0.004	-0.004
66	AS1	~~	AUA4	0.027	-0.006	-0.006	-0.010	-0.010
67	AS2	~~	AS3	6.186	0.105	0.105	0.154	0.154
68	AS2	~~	AS4	9.103	0.129	0.129	0.184	0.184
69	AS2	~~	AS5	2.292	0.065	0.065	0.095	0.095
70	AS2	~~	AS6	5.695	0.081	0.081	0.146	0.146
71	AS2	~~	AS7	1.072	0.044	0.044	0.063	0.063
72	AS2	~~	AS8	5.239	0.097	0.097	0.140	0.140
73	AS2	~~	AS9	6.981	0.101	0.101	0.160	0.160
74	AS2	~~	AF1	4.097	-0.088	-0.088	-0.122	-0.122
75	AS2	~~	AF2	11.242	-0.150	-0.150	-0.205	-0.205
76	AS2	~~	AF3	6.878	-0.114	-0.114	-0.157	-0.157
77	AS2	~~	AF4	1.941	-0.065	-0.065	-0.084	-0.084
78	AS2	~~	MI1	4.814	-0.086	-0.086	-0.133	-0.133
79	AS2	~~	MI2	0.001	-0.001	-0.001	-0.002	-0.002
80	AS2	~~	MI3	0.335	-0.028	-0.028	-0.035	-0.035
81	AS2	~~	MI4	6.183	-0.118	-0.118	-0.149	-0.149
82	AS2	~~	MI5	1.123	-0.045	-0.045	-0.064	-0.064
83	AS2	~~	AUA1	7.163	-0.130	-0.130	-0.162	-0.162
84	AS2	~~	AUA2	2.155	-0.060	-0.060	-0.090	-0.090
85	AS2	~~	AUA3	0.445	-0.026	-0.026	-0.040	-0.040
86	AS2	~~	AUA4	4.164	-0.087	-0.087	-0.126	-0.126
87	AS3	~~	AS4	2.403	0.061	0.061	0.095	0.095
88	AS3	~~	AS5	0.360	-0.024	-0.024	-0.038	-0.038
89	AS3	~~	AS6	0.000	0.000	0.000	0.000	0.000
90	AS3	~~	AS7	1.752	0.052	0.052	0.081	0.081
91	AS3	~~	AS8	0.886	0.037	0.037	0.058	0.058
92	AS3	~~	AS9	0.164	0.014	0.014	0.025	0.025
93	AS3	~~	AF1	2.082	0.058	0.058	0.087	0.087
94	AS3	~~	AF2	2.114	-0.060	-0.060	-0.089	-0.089
95	AS3	~~	AF3	0.850	0.037	0.037	0.055	0.055
96	AS3	~~	AF4	0.024	-0.007	-0.007	-0.009	-0.009
97	AS3	~~	MI1	3.213	-0.065	-0.065	-0.109	-0.109
98	AS3	~~	MI2	2.144	-0.058	-0.058	-0.088	-0.088
99	AS3	~~	MI3	0.511	-0.032	-0.032	-0.043	-0.043
100	AS3	~~	MI4	0.158	-0.017	-0.017	-0.024	-0.024
101	AS3	~~	MI5	0.978	-0.039	-0.039	-0.060	-0.060
102	AS3	~~	AUA1	0.165	-0.018	-0.018	-0.025	-0.025
103	AS3	~~	AUA2	2.950	-0.065	-0.065	-0.105	-0.105
104	AS3	~~	AUA3	0.212	-0.017	-0.017	-0.028	-0.028

105	AS3	~~	AUA4	1.248	-0.044	-0.044	-0.069	-0.069
106	AS4	~~	AS5	2.576	0.064	0.064	0.100	0.100
107	AS4	~~	AS6	1.089	-0.033	-0.033	-0.063	-0.063
108	AS4	~~	AS7	1.916	0.055	0.055	0.084	0.084
109	AS4	~~	AS8	0.319	0.022	0.022	0.034	0.034
110	AS4	~~	AS9	2.469	0.056	0.056	0.095	0.095
111	AS4	~~	AF1	0.024	-0.006	-0.006	-0.009	-0.009
112	AS4	~~	AF2	0.287	0.022	0.022	0.033	0.033
113	AS4	~~	AF3	3.687	-0.078	-0.078	-0.114	-0.114
114	AS4	~~	AF4	0.026	0.007	0.007	0.010	0.010
115	AS4	~~	MI1	2.555	-0.059	-0.059	-0.096	-0.096
116	AS4	~~	MI2	1.178	-0.043	-0.043	-0.065	-0.065
117	AS4	~~	MI3	2.225	-0.068	-0.068	-0.090	-0.090
118	AS4	~~	MI4	10.991	-0.146	-0.146	-0.197	-0.197
119	AS4	~~	MI5	0.225	0.019	0.019	0.028	0.028
120	AS4	~~	AUA1	8.523	-0.133	-0.133	-0.175	-0.175
121	AS4	~~	AUA2	5.898	-0.093	-0.093	-0.147	-0.147
122	AS4	~~	AUA3	1.901	-0.050	-0.050	-0.083	-0.083
123	AS4	~~	AUA4	0.000	0.000	0.000	-0.001	-0.001
124	AS5	~~	AS6	1.613	0.040	0.040	0.079	0.079
125	AS5	~~	AS7	0.511	0.028	0.028	0.045	0.045
126	AS5	~~	AS8	5.348	0.091	0.091	0.144	0.144
127	AS5	~~	AS9	3.532	0.067	0.067	0.116	0.116
128	AS5	~~	AF1	2.026	-0.057	-0.057	-0.087	-0.087
129	AS5	~~	AF2	2.047	-0.060	-0.060	-0.089	-0.089
130	AS5	~~	AF3	1.001	-0.040	-0.040	-0.061	-0.061
131	AS5	~~	AF4	0.164	-0.018	-0.018	-0.025	-0.025
132	AS5	~~	MI1	6.141	-0.091	-0.091	-0.153	-0.153
133	AS5	~~	MI2	0.728	-0.034	-0.034	-0.052	-0.052
134	AS5	~~	MI3	1.311	-0.052	-0.052	-0.070	-0.070
135	AS5	~~	MI4	4.499	-0.093	-0.093	-0.129	-0.129
136	AS5	~~	MI5	0.496	-0.028	-0.028	-0.043	-0.043
137	AS5	~~	AUA1	2.084	-0.065	-0.065	-0.089	-0.089
138	AS5	~~	AUA2	0.674	-0.032	-0.032	-0.051	-0.051
139	AS5	~~	AUA3	3.657	-0.069	-0.069	-0.118	-0.118
140	AS5	~~	AUA4	0.130	-0.014	-0.014	-0.023	-0.023
141	AS6	~~	AS7	0.738	-0.027	-0.027	-0.052	-0.052
142	AS6	~~	AS8	1.052	0.032	0.032	0.062	0.062
143	AS6	~~	AS9	4.961	0.063	0.063	0.134	0.134
144	AS6	~~	AF1	3.044	-0.056	-0.056	-0.104	-0.104
145	AS6	~~	AF2	2.437	-0.051	-0.051	-0.095	-0.095
146	AS6	~~	AF3	2.553	-0.051	-0.051	-0.095	-0.095
147	AS6	~~	AF4	2.090	-0.050	-0.050	-0.087	-0.087
148	AS6	~~	MI1	0.127	0.010	0.010	0.022	0.022
149	AS6	~~	MI2	0.011	-0.003	-0.003	-0.006	-0.006
150	AS6	~~	MI3	0.095	0.011	0.011	0.019	0.019
151	AS6	~~	MI4	0.092	-0.011	-0.011	-0.018	-0.018
152	AS6	~~	MI5	0.272	-0.016	-0.016	-0.031	-0.031

153	AS6	~~	AUA1	0.035	-0.007	-0.007	-0.011	-0.011
154	AS6	~~	AUA2	0.133	0.011	0.011	0.022	0.022
155	AS6	~~	AUA3	0.506	0.020	0.020	0.043	0.043
156	AS6	~~	AUA4	2.410	-0.049	-0.049	-0.095	-0.095
157	AS7	~~	AS8	1.769	0.052	0.052	0.081	0.081
158	AS7	~~	AS9	0.095	0.011	0.011	0.019	0.019
159	AS7	~~	AF1	2.433	-0.063	-0.063	-0.093	-0.093
160	AS7	~~	AF2	1.471	-0.050	-0.050	-0.074	-0.074
161	AS7	~~	AF3	0.701	-0.034	-0.034	-0.050	-0.050
162	AS7	~~	AF4	0.125	-0.015	-0.015	-0.021	-0.021
163	AS7	~~	MI1	0.126	-0.013	-0.013	-0.021	-0.021
164	AS7	~~	MI2	0.806	0.036	0.036	0.054	0.054
165	AS7	~~	MI3	4.071	-0.091	-0.091	-0.121	-0.121
166	AS7	~~	MI4	2.347	0.067	0.067	0.091	0.091
167	AS7	~~	MI5	0.003	0.002	0.002	0.003	0.003
168	AS7	~~	AUA1	0.138	-0.017	-0.017	-0.022	-0.022
169	AS7	~~	AUA2	0.942	-0.037	-0.037	-0.059	-0.059
170	AS7	~~	AUA3	1.301	-0.041	-0.041	-0.069	-0.069
171	AS7	~~	AUA4	0.400	0.025	0.025	0.039	0.039
172	AS8	~~	AS9	2.294	0.053	0.053	0.092	0.092
173	AS8	~~	AF1	0.035	-0.007	-0.007	-0.011	-0.011
174	AS8	~~	AF2	0.039	0.008	0.008	0.012	0.012
175	AS8	~~	AF3	0.147	-0.015	-0.015	-0.023	-0.023
176	AS8	~~	AF4	2.355	-0.066	-0.066	-0.093	-0.093
177	AS8	~~	MI1	2.370	-0.056	-0.056	-0.093	-0.093
178	AS8	~~	MI2	2.233	-0.059	-0.059	-0.089	-0.089
179	AS8	~~	MI3	0.615	0.035	0.035	0.047	0.047
180	AS8	~~	MI4	2.210	-0.065	-0.065	-0.089	-0.089
181	AS8	~~	MI5	1.053	-0.040	-0.040	-0.061	-0.061
182	AS8	~~	AUA1	0.115	-0.015	-0.015	-0.020	-0.020
183	AS8	~~	AUA2	1.772	-0.051	-0.051	-0.081	-0.081
184	AS8	~~	AUA3	8.486	-0.105	-0.105	-0.176	-0.176
185	AS8	~~	AUA4	2.816	-0.066	-0.066	-0.103	-0.103
186	AS9	~~	AF1	1.186	-0.039	-0.039	-0.065	-0.065
187	AS9	~~	AF2	0.018	-0.005	-0.005	-0.008	-0.008
188	AS9	~~	AF3	0.106	0.012	0.012	0.019	0.019
189	AS9	~~	AF4	3.138	-0.069	-0.069	-0.106	-0.106
190	AS9	~~	MI1	5.119	-0.074	-0.074	-0.136	-0.136
191	AS9	~~	MI2	3.713	-0.068	-0.068	-0.114	-0.114
192	AS9	~~	MI3	0.027	0.007	0.007	0.010	0.010
193	AS9	~~	MI4	0.304	-0.022	-0.022	-0.033	-0.033
194	AS9	~~	MI5	4.392	-0.074	-0.074	-0.125	-0.125
195	AS9	~~	AUA1	6.887	-0.106	-0.106	-0.157	-0.157
196	AS9	~~	AUA2	3.012	-0.059	-0.059	-0.105	-0.105
197	AS9	~~	AUA3	0.239	-0.016	-0.016	-0.029	-0.029
198	AS9	~~	AUA4	3.230	-0.063	-0.063	-0.110	-0.110
199	AF1	~~	AF2	12.031	0.146	0.146	0.208	0.208
200	AF1	~~	AF3	20.113	0.184	0.184	0.263	0.263

201	AF1	~~	AF4	18.999	0.192	0.192	0.259	0.259
202	AF1	~~	MI1	3.127	0.066	0.066	0.105	0.105
203	AF1	~~	MI2	2.997	-0.070	-0.070	-0.102	-0.102
204	AF1	~~	MI3	0.024	-0.007	-0.007	-0.009	-0.009
205	AF1	~~	MI4	0.057	-0.011	-0.011	-0.014	-0.014
206	AF1	~~	MI5	0.092	-0.012	-0.012	-0.018	-0.018
207	AF1	~~	AUA1	6.189	-0.114	-0.114	-0.147	-0.147
208	AF1	~~	AUA2	0.014	-0.005	-0.005	-0.007	-0.007
209	AF1	~~	AUA3	0.023	-0.006	-0.006	-0.009	-0.009
210	AF1	~~	AUA4	0.540	0.029	0.029	0.044	0.044
211	AF2	~~	AF3	1.501	0.052	0.052	0.073	0.073
212	AF2	~~	AF4	10.383	0.146	0.146	0.194	0.194
213	AF2	~~	MI1	9.780	0.120	0.120	0.189	0.189
214	AF2	~~	MI2	2.427	-0.065	-0.065	-0.093	-0.093
215	AF2	~~	MI3	0.143	0.018	0.018	0.023	0.023
216	AF2	~~	MI4	0.879	0.043	0.043	0.056	0.056
217	AF2	~~	MI5	0.604	0.032	0.032	0.047	0.047
218	AF2	~~	AUA1	0.339	0.028	0.028	0.035	0.035
219	AF2	~~	AUA2	0.020	0.006	0.006	0.009	0.009
220	AF2	~~	AUA3	0.453	0.026	0.026	0.041	0.041
221	AF2	~~	AUA4	0.019	0.006	0.006	0.009	0.009
222	AF3	~~	AF4	3.396	0.081	0.081	0.109	0.109
223	AF3	~~	MI1	0.156	0.015	0.015	0.023	0.023
224	AF3	~~	MI2	0.016	-0.005	-0.005	-0.007	-0.007
225	AF3	~~	MI3	0.002	-0.002	-0.002	-0.003	-0.003
226	AF3	~~	MI4	0.109	0.015	0.015	0.019	0.019
227	AF3	~~	MI5	2.635	0.066	0.066	0.095	0.095
228	AF3	~~	AUA1	0.238	0.023	0.023	0.029	0.029
229	AF3	~~	AUA2	0.018	-0.005	-0.005	-0.008	-0.008
230	AF3	~~	AUA3	0.049	0.008	0.008	0.013	0.013
231	AF3	~~	AUA4	0.012	0.004	0.004	0.007	0.007
232	AF4	~~	MI1	2.068	0.058	0.058	0.086	0.086
233	AF4	~~	MI2	1.819	0.059	0.059	0.080	0.080
234	AF4	~~	MI3	0.002	-0.002	-0.002	-0.003	-0.003
235	AF4	~~	MI4	0.261	0.025	0.025	0.030	0.030
236	AF4	~~	MI5	0.061	0.011	0.011	0.015	0.015
237	AF4	~~	AUA1	0.275	-0.026	-0.026	-0.031	-0.031
238	AF4	~~	AUA2	0.267	0.022	0.022	0.031	0.031
239	AF4	~~	AUA3	2.030	-0.056	-0.056	-0.085	-0.085
240	AF4	~~	AUA4	0.164	0.018	0.018	0.025	0.025
241	MI1	~~	MI2	6.068	0.090	0.090	0.146	0.146
242	MI1	~~	MI3	8.331	0.121	0.121	0.172	0.172
243	MI1	~~	MI4	8.702	0.120	0.120	0.174	0.174
244	MI1	~~	MI5	0.432	-0.024	-0.024	-0.039	-0.039
245	MI1	~~	AUA1	3.380	0.077	0.077	0.110	0.110
246	MI1	~~	AUA2	0.457	0.024	0.024	0.041	0.041
247	MI1	~~	AUA3	0.033	-0.006	-0.006	-0.011	-0.011
248	MI1	~~	AUA4	3.144	0.065	0.065	0.108	0.108

249	MI2	~~	MI3	22.148	0.214	0.214	0.278	0.278
250	MI2	~~	MI4	4.983	0.099	0.099	0.131	0.131
251	MI2	~~	MI5	0.711	0.034	0.034	0.050	0.050
252	MI2	~~	AUA1	5.645	0.108	0.108	0.140	0.140
253	MI2	~~	AUA2	1.484	0.047	0.047	0.073	0.073
254	MI2	~~	AUA3	0.417	0.023	0.023	0.038	0.038
255	MI2	~~	AUA4	0.141	-0.015	-0.015	-0.023	-0.023
256	MI3	~~	MI4	10.514	0.163	0.163	0.191	0.191
257	MI3	~~	MI5	0.095	-0.014	-0.014	-0.018	-0.018
258	MI3	~~	AUA1	1.624	0.066	0.066	0.076	0.076
259	MI3	~~	AUA2	0.088	-0.013	-0.013	-0.018	-0.018
260	MI3	~~	AUA3	0.119	-0.014	-0.014	-0.021	-0.021
261	MI3	~~	AUA4	0.506	-0.032	-0.032	-0.043	-0.043
262	MI4	~~	MI5	5.730	0.105	0.105	0.140	0.140
263	MI4	~~	AUA1	0.866	0.047	0.047	0.055	0.055
264	MI4	~~	AUA2	0.003	0.002	0.002	0.003	0.003
265	MI4	~~	AUA3	2.620	0.065	0.065	0.096	0.096
266	MI4	~~	AUA4	0.156	0.017	0.017	0.024	0.024
267	MI5	~~	AUA1	0.207	0.021	0.021	0.027	0.027
268	MI5	~~	AUA2	2.520	0.061	0.061	0.095	0.095
269	MI5	~~	AUA3	1.464	0.044	0.044	0.072	0.072
270	MI5	~~	AUA4	0.002	0.002	0.002	0.003	0.003
271	AUA1	~~	AUA2	17.432	0.183	0.183	0.250	0.250
272	AUA1	~~	AUA3	9.421	0.127	0.127	0.183	0.183
273	AUA1	~~	AUA4	5.782	0.109	0.109	0.146	0.146
274	AUA2	~~	AUA3	7.002	0.092	0.092	0.159	0.159
275	AUA2	~~	AUA4	6.210	0.095	0.095	0.153	0.153
276	AUA3	~~	AUA4	5.069	0.081	0.081	0.137	0.137

Kline recommends evaluating the “cor” residuals. In our output, these seem to be the “cor.bollen” and are near the bottom. He recommends that residuals > .10 may be possible sources for misfit. He also indicated that patterns may be helpful (is there an item that has consistently high residuals).

Kline also cautions that there is no dependable or trustworthy connection between the size of the residual and the type or degree of model misspecification.

My first read of our results is that the items in the AS# factor were well-defined. I suspect that a multi-factor solution will improve the fit.

The *tidySEM* package has some useful tools to export the results to .csv files. This first set of code exports the fit indices.

```
UniDFitStats <- tidySEM::table_fit(grmsAAW1fit)
```

```
Registered S3 method overwritten by 'tidySEM':
method           from
predict.MxModel OpenMx
```

## UniDFitStats

```

      Name Parameters      fmin      chisq   df pvalue baseline.chisq
1 grmsAAW1fit        44 0.7310043 444.4506 209      0     1439.317
      baseline.df baseline.pvalue      cfi      tli      nnfi      rfi      nfi
1          231           0 0.8051418 0.7846304 0.7846304 0.6587029 0.6912074
      pnfi      ifi      rni      LL unrestricted.logl      aic      bic
1 0.6253781 0.8086262 0.8051418 -8387.014           -8164.789 16862.03 17025.58
      n      bic2      rmsea rmsea.ci.lower rmsea.ci.upper rmsea.ci.level
1 304 16886.03 0.06087514      0.05302463      0.06871732           0.9
      rmsea.pvalue rmsea.close.h0 rmsea.notclose.pvalue rmsea.notclose.h0
1 0.01213836      0.05      0.00001944378      0.08
      rmr rmr_nomean      srmr srmr_bentler srmr_bentler_nomean      crmr
1 0.05740799 0.05740799 0.06699187      0.06699187           0.06699187 0.07010942
      crmr_nomean srmr_mplus srmr_mplus_nomean      cn_05      cn_01      gfi      agfi
1 0.07010942 0.06699187           0.06699187 167.7071 178.4826 0.8602203 0.830793
      pgfi      mfi      ecvi
1 0.7106168 0.6789184 1.751482

```

The second set of code exports the parameter estimates.

```

UniD_paramEsts <- tidySEM::table_results(grmsAAW1fit, digits=3, columns = NULL)
UniD_paramEsts

```

	lhs op	rhs	est	se	pval	confint	est_sig	est_std	se_std
1	GRMSAAW =~	AS1	1.000	0.000	<NA>	[1.000, 1.000]	1.000	0.535	0.046
2	GRMSAAW =~	AS2	1.069	0.152	0.000	[0.770, 1.368]	1.069***	0.520	0.047
3	GRMSAAW =~	AS3	1.024	0.143	0.000	[0.743, 1.304]	1.024***	0.535	0.046
4	GRMSAAW =~	AS4	0.909	0.137	0.000	[0.641, 1.177]	0.909***	0.482	0.049
5	GRMSAAW =~	AS5	1.177	0.154	0.000	[0.875, 1.479]	1.177***	0.590	0.043
6	GRMSAAW =~	AS6	0.721	0.108	0.000	[0.509, 0.934]	0.721***	0.483	0.049
7	GRMSAAW =~	AS7	0.914	0.137	0.000	[0.646, 1.181]	0.914***	0.487	0.049
8	GRMSAAW =~	AS8	0.927	0.137	0.000	[0.658, 1.196]	0.927***	0.494	0.048
9	GRMSAAW =~	AS9	0.735	0.117	0.000	[0.505, 0.965]	0.735***	0.445	0.051
10	GRMSAAW =~	AF1	0.675	0.125	0.000	[0.431, 0.920]	0.675***	0.370	0.054
11	GRMSAAW =~	AF2	0.975	0.144	0.000	[0.692, 1.258]	0.975***	0.493	0.048
12	GRMSAAW =~	AF3	0.555	0.120	0.000	[0.320, 0.790]	0.555***	0.308	0.057
13	GRMSAAW =~	AF4	0.851	0.141	0.000	[0.575, 1.128]	0.851***	0.425	0.052
14	GRMSAAW =~	MI1	0.744	0.120	0.000	[0.508, 0.980]	0.744***	0.438	0.051
15	GRMSAAW =~	MI2	0.641	0.122	0.000	[0.402, 0.880]	0.641***	0.357	0.055
16	GRMSAAW =~	MI3	0.860	0.146	0.000	[0.575, 1.146]	0.860***	0.413	0.052
17	GRMSAAW =~	MI4	0.601	0.130	0.000	[0.346, 0.856]	0.601***	0.307	0.057
18	GRMSAAW =~	MI5	0.655	0.122	0.000	[0.415, 0.895]	0.655***	0.365	0.054
19	GRMSAAW =~	AUA1	0.825	0.144	0.000	[0.543, 1.107]	0.825***	0.398	0.053
20	GRMSAAW =~	AUA2	0.878	0.132	0.000	[0.620, 1.137]	0.878***	0.483	0.049
21	GRMSAAW =~	AUA3	0.714	0.118	0.000	[0.483, 0.944]	0.714***	0.426	0.052

22	GRMSAAW	=~	AUA4	1.060	0.146	0.000	[0.774, 1.346]	1.060***	0.547	0.045
23	AS1	~~	AS1	0.600	0.052	0.000	[0.497, 0.702]	0.600***	0.713	0.049
24	AS2	~~	AS2	0.744	0.064	0.000	[0.618, 0.870]	0.744***	0.730	0.049
25	AS3	~~	AS3	0.631	0.055	0.000	[0.523, 0.738]	0.631***	0.714	0.049
26	AS4	~~	AS4	0.657	0.056	0.000	[0.547, 0.767]	0.657***	0.767	0.047
27	AS5	~~	AS5	0.624	0.056	0.000	[0.515, 0.733]	0.624***	0.651	0.050
28	AS6	~~	AS6	0.412	0.035	0.000	[0.343, 0.480]	0.412***	0.766	0.047
29	AS7	~~	AS7	0.648	0.055	0.000	[0.539, 0.757]	0.648***	0.763	0.047
30	AS8	~~	AS8	0.641	0.055	0.000	[0.534, 0.749]	0.641***	0.756	0.048
31	AS9	~~	AS9	0.528	0.045	0.000	[0.440, 0.615]	0.528***	0.802	0.045
32	AF1	~~	AF1	0.693	0.058	0.000	[0.580, 0.807]	0.693***	0.863	0.040
33	AF2	~~	AF2	0.714	0.061	0.000	[0.594, 0.834]	0.714***	0.757	0.048
34	AF3	~~	AF3	0.707	0.058	0.000	[0.593, 0.821]	0.707***	0.905	0.035
35	AF4	~~	AF4	0.794	0.067	0.000	[0.663, 0.925]	0.794***	0.820	0.044
36	MI1	~~	MI1	0.564	0.048	0.000	[0.470, 0.657]	0.564***	0.809	0.045
37	MI2	~~	MI2	0.678	0.056	0.000	[0.568, 0.789]	0.678***	0.873	0.039
38	MI3	~~	MI3	0.870	0.073	0.000	[0.727, 1.013]	0.870***	0.830	0.043
39	MI4	~~	MI4	0.839	0.069	0.000	[0.703, 0.975]	0.839***	0.906	0.035
40	MI5	~~	MI5	0.672	0.056	0.000	[0.562, 0.781]	0.672***	0.866	0.040
41	AUA1	~~	AUA1	0.872	0.073	0.000	[0.729, 1.015]	0.872***	0.842	0.042
42	AUA2	~~	AUA2	0.610	0.052	0.000	[0.508, 0.712]	0.610***	0.766	0.047
43	AUA3	~~	AUA3	0.553	0.047	0.000	[0.462, 0.644]	0.553***	0.818	0.044
44	AUA4	~~	AUA4	0.634	0.055	0.000	[0.525, 0.742]	0.634***	0.701	0.049
45	GRMSAAW	~~	GRMSAAW	0.241	0.051	0.000	[0.140, 0.342]	0.241***	1.000	0.000
				pval_std	confint_std	est_sig_std		label		
1				0.000	[0.445, 0.626]	0.535***		GRMSAAW.BY.AS1		
2				0.000	[0.428, 0.612]	0.520***		GRMSAAW.BY.AS2		
3				0.000	[0.445, 0.625]	0.535***		GRMSAAW.BY.AS3		
4				0.000	[0.387, 0.578]	0.482***		GRMSAAW.BY.AS4		
5				0.000	[0.507, 0.674]	0.590***		GRMSAAW.BY.AS5		
6				0.000	[0.387, 0.579]	0.483***		GRMSAAW.BY.AS6		
7				0.000	[0.391, 0.582]	0.487***		GRMSAAW.BY.AS7		
8				0.000	[0.399, 0.589]	0.494***		GRMSAAW.BY.AS8		
9				0.000	[0.345, 0.545]	0.445***		GRMSAAW.BY.AS9		
10				0.000	[0.264, 0.476]	0.370***		GRMSAAW.BY.AF1		
11				0.000	[0.398, 0.588]	0.493***		GRMSAAW.BY.AF2		
12				0.000	[0.197, 0.419]	0.308***		GRMSAAW.BY.AF3		
13				0.000	[0.323, 0.526]	0.425***		GRMSAAW.BY.AF4		
14				0.000	[0.337, 0.538]	0.438***		GRMSAAW.BY.MI1		
15				0.000	[0.249, 0.464]	0.357***		GRMSAAW.BY.MI2		
16				0.000	[0.310, 0.515]	0.413***		GRMSAAW.BY.MI3		
17				0.000	[0.195, 0.418]	0.307***		GRMSAAW.BY.MI4		
18				0.000	[0.259, 0.472]	0.365***		GRMSAAW.BY.MI5		
19				0.000	[0.294, 0.502]	0.398***		GRMSAAW.BY.AUA1		
20				0.000	[0.388, 0.579]	0.483***		GRMSAAW.BY.AUA2		
21				0.000	[0.325, 0.528]	0.426***		GRMSAAW.BY.AUA3		
22				0.000	[0.458, 0.636]	0.547***		GRMSAAW.BY.AUA4		
23				0.000	[0.617, 0.810]	0.713***		Variances.AS1		

24	0.000 [0.634, 0.825]	0.730***	Variances.AS2
25	0.000 [0.618, 0.810]	0.714***	Variances.AS3
26	0.000 [0.675, 0.860]	0.767***	Variances.AS4
27	0.000 [0.553, 0.750]	0.651***	Variances.AS5
28	0.000 [0.674, 0.859]	0.766***	Variances.AS6
29	0.000 [0.670, 0.856]	0.763***	Variances.AS7
30	0.000 [0.662, 0.849]	0.756***	Variances.AS8
31	0.000 [0.713, 0.891]	0.802***	Variances.AS9
32	0.000 [0.784, 0.942]	0.863***	Variances.AF1
33	0.000 [0.663, 0.850]	0.757***	Variances.AF2
34	0.000 [0.837, 0.973]	0.905***	Variances.AF3
35	0.000 [0.733, 0.906]	0.820***	Variances.AF4
36	0.000 [0.721, 0.896]	0.809***	Variances.MI1
37	0.000 [0.796, 0.949]	0.873***	Variances.MI2
38	0.000 [0.745, 0.914]	0.830***	Variances.MI3
39	0.000 [0.838, 0.974]	0.906***	Variances.MI4
40	0.000 [0.788, 0.944]	0.866***	Variances.MI5
41	0.000 [0.759, 0.924]	0.842***	Variances.AUA1
42	0.000 [0.674, 0.859]	0.766***	Variances.AUA2
43	0.000 [0.732, 0.905]	0.818***	Variances.AUA3
44	0.000 [0.604, 0.798]	0.701***	Variances.AUA4
45	<NA> [1.000, 1.000]	1.000	Variances.GRMSAAW

We can write each of these to a .csv file that will be stored in the same folder as your .rmd file.

```
write.csv(UniDFitStats, file = "UnidimensionalFitStats.csv")
write.csv(UniD_paramEsts, file = "UnidimensionalParamEsts.csv")
```

*Troubleshooting* If, while working with this function you get the error, “Error in file(file, ifelse(append,”a”, “w”)) : cannot open the connection” it is because the .csv file that received your table is still open. R is just trying to write over it. A similar error happens when knitting or updating any spreadsheet or word document.

### 10.5.1.7 APA Style Results from the Unidimensional model

Writing up an APA style results section for a CFA involves describing the statistics that are being used and then presenting the results.

**Model testing.** To evaluate the models we, we used confirmatory factor analysis (CFA) in the R package, *lavaan* (v.0-6.9) with maximum likelihood estimation. Our sample size was 304. We selected fit criteria for their capacity to assess different aspects of the statistical analysis. As is common among SEM researchers, we reported the Chi-square goodness of fit ( $\chi^2$ ). This evaluates the discrepancy between the unrestricted sample matrix and the restricted covariance matrix. Although the associated *p* value indicates adequate fit when the value is non-significant, it is widely recognized that large sample size can result in a statistically significant *p* value [Byrne, 2016b]. The comparative fit

index (CFI) is an incremental index, comparing the hypothesized model at least .90 and perhaps higher than .95 [Kline, 2016]. The root mean square error of approximation (RMSEA) takes into account the error of approximation in the population and expresses it per degree of freedom. As such, the fit indicator considers the complexity of the model. Ideal values are equal to or less than .05, values less than .08 represent reasonable fit, and values between .08 and .10 represent mediocre fit. The standardized root mean residual (SRMR) is a standardized measure of the mean absolute covariance residual – the overall difference between the observed and predicted correlations. Values greater than .10 may indicate poor fit and inspection of residuals is then advised. Kline [2016] advised researchers to be cautious when using these criteria as strict cut-offs. Elements such as sample size and model complexity should be considered when evaluating fit.

Our first model was unidimensional where each of the 24 items loaded onto a single factor representing overall, gendered racial microaggressions towards Asian American women. The Chi-square index was statistically significant ( $\chi^2(209) = 444.451, p < .001$ ) indicating likely misfit. The CFI value of .81 indicated poor fit. In contrast, the RMSEA = 0.061, 90% CI(0.053, 0.069) and SRMR = 0.067 both fell within the ranges of acceptability. The AIC and BIC values were 16862.028 and 17025.577, respectively, and will become useful in comparing subsequent models.

## 10.5.2 Modeling the GRMSAAW as a First-Order, 4-factor model

### 10.5.2.1 Specifying and Running the Model

As we know from the article, the GRMSAAW has four subscales. Therefore, let's respecify it as a first-order, four-factor model, allowing the factors to correlate.

**Model identification** is always a consideration. In a multi-dimensional model, each factor requires a minimum of two items/indicators. Our shortest scales are the AF and AUA scales, each with 4 items, so we should be identified.

We will be using the *cfa()* function in lavaan. When we do this, it does three things by default:

1. The factor loading of the first indicator of a latent variable is fixed to 1.0; this fixes the scale of the LV
  2. Residual variances are added automatically.
  3. All exogenous LVs are correlated.
- If you are specifying an orthogonal model you will want to switch off the default behavior by including the statement: `auto.cov.lv.x=FALSE`

```
grmsAAW4mod <- "AS =~ AS1 + AS2 + AS3 + AS4 + AS5 + AS6 + AS7 + AS8 + AS9
                    AF =~ AF1 + AF2 + AF3 + AF4
                    MI =~ MI1 + MI2 + MI3 + MI4 + MI5
                    AUA =~ AUA1 + AUA2 + AUA3 + AUA4"
```

```

# This code is identical to the one we ran above -- in this code
# below, we are just clearly specifying the covariances -- but the
# default of lavaan is to correlate latent variables when the cfa()
# function is used.

grmsAAW4mod <- "AS =~ AS1 + AS2 + AS3 + AS4 + AS5 + AS6 + AS7 + AS8 + AS9
                  AF =~ AF1 + AF2 + AF3 + AF4
                  MI =~ MI1 + MI2 + MI3 + MI4 + MI5
                  AUA =~ AUA1 + AUA2 + AUA3 + AUA4

#covariances in our oblique model
  AS ~~ AF
  AS ~~ MI
  AS ~~ AUA
  AF ~~ MI
  AF ~~ AUA
  MI ~~ AUA
  "
}

set.seed(240311)
grmsAAW4fit <- lavaan::cfa(grmsAAW4mod, data = dfGRMSAAW)
lavaan::summary(grmsAAW4fit, fit.measures = TRUE, standardized = TRUE,
                rsquare = TRUE)

```

lavaan 0.6.17 ended normally after 42 iterations

Estimator	ML
Optimization method	NLMINB
Number of model parameters	50
Number of observations	304

Model Test User Model:

Test statistic	232.453
Degrees of freedom	203
P-value (Chi-square)	0.076

Model Test Baseline Model:

Test statistic	1439.317
Degrees of freedom	231
P-value	0.000

User Model versus Baseline Model:

Comparative Fit Index (CFI)	0.976
-----------------------------	-------

Tucker-Lewis Index (TLI) 0.972

Loglikelihood and Information Criteria:

Loglikelihood user model (H0)	-8281.015
Loglikelihood unrestricted model (H1)	-8164.789

Akaike (AIC)	16662.030
Bayesian (BIC)	16847.882
Sample-size adjusted Bayesian (SABIC)	16689.307

Root Mean Square Error of Approximation:

RMSEA	0.022
90 Percent confidence interval - lower	0.000
90 Percent confidence interval - upper	0.034
P-value H_0: RMSEA <= 0.050	1.000
P-value H_0: RMSEA >= 0.080	0.000

Standardized Root Mean Square Residual:

SRMR	0.047
------	-------

Parameter Estimates:

Standard errors	Standard
Information	Expected
Information saturated (h1) model	Structured

Latent Variables:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
AS =~						
AS1	1.000				0.550	0.600
AS2	1.132	0.136	8.330	0.000	0.623	0.617
AS3	0.958	0.123	7.769	0.000	0.527	0.561
AS4	0.901	0.120	7.504	0.000	0.496	0.536
AS5	1.152	0.134	8.620	0.000	0.634	0.647
AS6	0.669	0.094	7.133	0.000	0.368	0.503
AS7	0.829	0.118	7.043	0.000	0.456	0.495
AS8	0.905	0.120	7.551	0.000	0.498	0.540
AS9	0.757	0.104	7.256	0.000	0.417	0.514
AF =~						
AF1	1.000				0.505	0.563
AF2	1.195	0.174	6.862	0.000	0.603	0.621
AF3	0.738	0.137	5.395	0.000	0.373	0.422
AF4	1.138	0.171	6.665	0.000	0.575	0.584
MI =~						
MI1	1.000				0.482	0.577

MI2	0.917	0.148	6.216	0.000	0.442	0.501
MI3	1.169	0.177	6.602	0.000	0.563	0.550
MI4	0.921	0.157	5.865	0.000	0.444	0.461
MI5	0.688	0.137	5.018	0.000	0.332	0.377
AUA =~						
AUA1	1.000				0.553	0.543
AUA2	0.981	0.140	7.016	0.000	0.543	0.608
AUA3	0.785	0.122	6.457	0.000	0.434	0.528
AUA4	1.083	0.152	7.140	0.000	0.599	0.630

## Covariances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
AS ~~						
AF	0.148	0.030	4.951	0.000	0.533	0.533
MI	0.136	0.028	4.889	0.000	0.513	0.513
AUA	0.181	0.034	5.257	0.000	0.595	0.595
AF ~~						
MI	0.154	0.031	5.010	0.000	0.632	0.632
AUA	0.164	0.034	4.805	0.000	0.588	0.588
MI ~~						
AUA	0.189	0.036	5.303	0.000	0.709	0.709

## Variances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
.AS1	0.538	0.050	10.833	0.000	0.538	0.640
.AS2	0.632	0.059	10.699	0.000	0.632	0.620
.AS3	0.605	0.054	11.111	0.000	0.605	0.685
.AS4	0.610	0.054	11.260	0.000	0.610	0.713
.AS5	0.557	0.053	10.408	0.000	0.557	0.581
.AS6	0.401	0.035	11.433	0.000	0.401	0.747
.AS7	0.641	0.056	11.470	0.000	0.641	0.755
.AS8	0.601	0.053	11.235	0.000	0.601	0.708
.AS9	0.484	0.043	11.379	0.000	0.484	0.736
.AF1	0.548	0.055	9.928	0.000	0.548	0.683
.AF2	0.579	0.064	9.062	0.000	0.579	0.614
.AF3	0.642	0.057	11.230	0.000	0.642	0.822
.AF4	0.638	0.066	9.651	0.000	0.638	0.659
.MI1	0.465	0.047	9.823	0.000	0.465	0.667
.MI2	0.582	0.055	10.664	0.000	0.582	0.749
.MI3	0.731	0.072	10.158	0.000	0.731	0.697
.MI4	0.729	0.066	10.994	0.000	0.729	0.787
.MI5	0.665	0.058	11.519	0.000	0.665	0.858
.AUA1	0.730	0.069	10.535	0.000	0.730	0.705
.AUA2	0.501	0.051	9.787	0.000	0.501	0.630
.AUA3	0.487	0.046	10.675	0.000	0.487	0.721
.AUA4	0.546	0.058	9.475	0.000	0.546	0.603
AS	0.303	0.058	5.264	0.000	1.000	1.000
AF	0.255	0.058	4.412	0.000	1.000	1.000

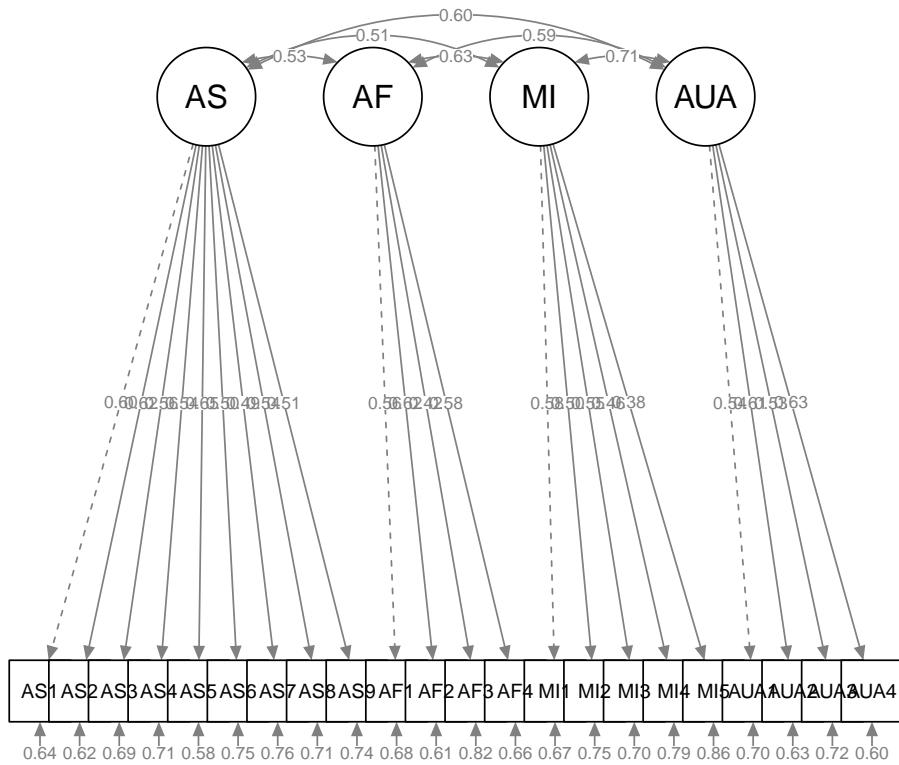
MI	0.232	0.051	4.559	0.000	1.000	1.000
AUA	0.306	0.070	4.391	0.000	1.000	1.000

R-Square:

	Estimate
AS1	0.360
AS2	0.380
AS3	0.315
AS4	0.287
AS5	0.419
AS6	0.253
AS7	0.245
AS8	0.292
AS9	0.264
AF1	0.317
AF2	0.386
AF3	0.178
AF4	0.341
MI1	0.333
MI2	0.251
MI3	0.303
MI4	0.213
MI5	0.142
AUA1	0.295
AUA2	0.370
AUA3	0.279
AUA4	0.397

I'm inclined to immediately create a figure. This permits me to inspect for modeling errors "at a glance."

```
semPlot::semPaths(grmsAAW4fit, layout = "tree", style = "lisrel", what = "col",
whatLabels = "stand")
```



Among my first steps are also to write the code to export the results. Because this involved correlated factors, I will produce an additional table. First, I create the objects that hold the results.

```
Coor4FitStats <- tidySEM:::table_fit(grmsAAW4fit)
Coor4_paramEsts <- tidySEM:::table_results(grmsAAW4fit, digits=3, columns = NULL)
Coor4Corrs <- tidySEM:::table_cors(grmsAAW4fit, digits=3)
#to see each of the tables, remove the hashtag
#Coor4FitStats
#Coor4_paramEsts
#Coor4Corrs
```

Next, I export them.

```
write.csv(Coor4FitStats, file = "Coor4FitStats.csv")
write.csv(Coor4_paramEsts, file = "Coor4_paramEstsParamEsts.csv")
write.csv(Coor4Corrs, file = "Coor4Corrs.csv")
```

### 10.5.2.2 Interpretation

Let's interpret the results.

Our model converged, normally, with 42 iterations. The estimator was the lavaan default, maximum likelihood (ML). All 304 cases were used in the analysis.

I mapped our pattern coefficients into the GRMSAAW tables. Most pattern coefficients are strong, significant, and stably connected to their respective factor. The lowest factor loading was 0.332 (MI5).

A multidimensional factor structure also includes correlations/covariances between factors. We can see that the correlation (look at the Std.all column) shows the following correlations (all are statistically significant):

AS & AF: 0.533 AS & MI: 0.513 AS & AUA: 0.595

AF & MI: 0.632 AF & AUA: 0.588 MI & AUA: 0.709

For our multi-dimensional GRMSAAW4 CFA  $\chi^2(203) = 232.453, p = 0.076$ , this non-significant  $p$  value is exactly what we want because it says that our specified model is not statistically significantly different than the covariances in the model. That is, our more parsimonious model is a reasonable explanation (simplification).

The CFI and TLI compare user (the 4-dimensional model we specified) and baseline (where no relations would exist between variables) models. These values will always be close together because the only difference is that the TLI imposes a penalty for any model complexity. The CFI seems to be more commonly reported and its value is 0.976. This means our model performed 98% better than a model with no relations. It well-exceeds the traditional cutoffs of .90 and the stricter cutoff of .95. The TLI imposes a greater relative penalty for model complexity, consequently it is a smidge lower at 0.972.

The RMSEA one of the *badness of fit*, absolute fit index, statistics where a value of 0.00 is the best fit. Our RMSEA = 0.022 (90%CI[.000, 0.034]). As a quick reminder, there is general consensus that  $RMSEA \leq .05$  is desired and an  $RMSEA \geq .10$  points to serious problems. We watch the upper bound of the confidence interval to see that it isn't sneaking into the danger zone.

The SRMR is another absolute, *badness of fit* index (i.e., perfect model fit is when the value = 0.00 and increasingly higher values indicate the “badness”). The SRMR is a measure of the mean absolute covariance residual. Standardizing the value facilitates interpretation. Poor fit is indicated when  $SRMR \geq .10$ . The GRMSAAW SRMR = 0.047.

Recall, Hu and Bentler's **combination rule** (which is somewhat contested) suggested that the SRMR be interpreted along with the CFI such that:  $CFI \geq .95$  and  $SRMR \leq .08$ .

For our unidimensional GRMSAAW CFA, the CFI = 0.976 and the SRMR = 0.047. Our results fell within that acceptable range!

The AIC and BIC utilize an information theory approach to data analysis by combining statistical estimation and model selection into a single framework. The BIC augments the AIC by taking sample size into consideration. We can compare the values from our current model to the former one. The model with the smallest value of the predictive fit index is chosen as the one that is most likely to replicate. It means that this model has relatively better fit and fewer free parameters than competing models. We will do that in the next section.

Before moving to model comparison, it is a good practice for locating sources of misfit (we look for relatively large values) is to inspect the residuals (in the “cor.bollen” section), so let’s do that.

```
lavaan::fitted(grmsAAW4fit)
```

```
$cov
      AS1   AS2   AS3   AS4   AS5   AS6   AS7   AS8   AS9   AF1   AF2   AF3
AS1  0.841
AS2  0.343 1.020
AS3  0.290 0.328 0.883
AS4  0.273 0.309 0.261 0.856
AS5  0.349 0.395 0.334 0.314 0.958
AS6  0.203 0.229 0.194 0.183 0.233 0.537
AS7  0.251 0.284 0.240 0.226 0.289 0.168 0.849
AS8  0.274 0.310 0.262 0.247 0.315 0.183 0.227 0.849
AS9  0.229 0.259 0.220 0.207 0.264 0.153 0.190 0.207 0.658
AF1  0.148 0.168 0.142 0.133 0.171 0.099 0.123 0.134 0.112 0.803
AF2  0.177 0.200 0.169 0.159 0.204 0.118 0.147 0.160 0.134 0.305 0.943
AF3  0.109 0.124 0.105 0.098 0.126 0.073 0.091 0.099 0.083 0.188 0.225 0.781
AF4  0.168 0.191 0.161 0.152 0.194 0.113 0.140 0.152 0.127 0.290 0.347 0.214
MI1  0.136 0.154 0.130 0.123 0.157 0.091 0.113 0.123 0.103 0.154 0.184 0.114
MI2  0.125 0.141 0.120 0.112 0.144 0.084 0.103 0.113 0.094 0.141 0.169 0.104
MI3  0.159 0.180 0.152 0.143 0.183 0.106 0.132 0.144 0.120 0.180 0.215 0.133
MI4  0.125 0.142 0.120 0.113 0.144 0.084 0.104 0.113 0.095 0.142 0.169 0.105
MI5  0.094 0.106 0.090 0.084 0.108 0.063 0.078 0.085 0.071 0.106 0.126 0.078
AUA1 0.181 0.205 0.174 0.163 0.209 0.121 0.150 0.164 0.137 0.164 0.196 0.121
AUA2 0.178 0.201 0.170 0.160 0.205 0.119 0.147 0.161 0.135 0.161 0.193 0.119
AUA3 0.142 0.161 0.136 0.128 0.164 0.095 0.118 0.129 0.108 0.129 0.154 0.095
AUA4 0.196 0.222 0.188 0.177 0.226 0.131 0.163 0.178 0.149 0.178 0.213 0.131
      AF4   MI1   MI2   MI3   MI4   MI5   AUA1   AUA2   AUA3   AUA4
AS1
AS2
AS3
AS4
AS5
AS6
AS7
AS8
AS9
AF1
AF2
AF3
AF4  0.968
MI1  0.175 0.697
MI2  0.160 0.213 0.777
MI3  0.205 0.271 0.249 1.048
MI4  0.161 0.214 0.196 0.250 0.926
```

```

MI5  0.120 0.160 0.147 0.187 0.147 0.775
AUA1 0.187 0.189 0.173 0.221 0.174 0.130 1.036
AUA2 0.183 0.185 0.170 0.217 0.171 0.128 0.300 0.796
AUA3 0.147 0.148 0.136 0.173 0.137 0.102 0.240 0.236 0.676
AUA4 0.202 0.205 0.188 0.239 0.189 0.141 0.331 0.325 0.260 0.904

```

```

# lavaan::residuals(grmsAAW4fit, type = 'raw')
# lavaan::residuals(grmsAAW4fit, type = 'standardized')
lavaan::residuals(grmsAAW4fit, type = "cor")

```

```

$type
[1] "cor.bollen"

$cov
      AS1     AS2     AS3     AS4     AS5     AS6     AS7     AS8     AS9     AF1
AS1  0.000
AS2  0.009  0.000
AS3  0.009  0.028  0.000
AS4  0.009  0.042  0.019  0.000
AS5  0.065 -0.038 -0.069 -0.002  0.000
AS6 -0.018  0.037 -0.024 -0.080  0.007  0.000
AS7 -0.058 -0.010  0.035  0.027 -0.006 -0.049  0.000
AS8 -0.057  0.015 -0.002 -0.028  0.027  0.009  0.028  0.000
AS9  0.010  0.024 -0.033  0.007  0.003  0.052 -0.024  0.007  0.000
AF1 -0.035 -0.081  0.091  0.011 -0.033 -0.050 -0.038  0.012 -0.040  0.000
AF2  0.025 -0.082  0.021  0.083  0.024  0.008  0.026  0.073  0.044 -0.014
AF3  0.003 -0.095  0.079 -0.060 -0.005 -0.038  0.000  0.013  0.037  0.099
AF4 -0.052 -0.029  0.046  0.045  0.034 -0.014  0.038 -0.024 -0.050  0.031
MI1 -0.036 -0.046 -0.006 -0.016 -0.029  0.078  0.051 -0.009 -0.057  0.038
MI2 -0.080  0.025 -0.016 -0.015  0.010  0.038  0.087 -0.029 -0.063 -0.130
MI3 -0.028  0.016  0.032 -0.017  0.016  0.071 -0.027  0.085  0.046 -0.051
MI4 -0.081 -0.098  0.014 -0.132 -0.060  0.015  0.103 -0.044 -0.011 -0.063
MI5  0.086  0.025  0.045  0.094  0.062  0.056  0.085  0.031 -0.033 -0.014
AUA1 -0.056 -0.107  0.014 -0.110 -0.032  0.021  0.017  0.007 -0.108 -0.151
AUA2  0.016 -0.031 -0.013 -0.062  0.020  0.067  0.016 -0.012 -0.045 -0.028
AUA3  0.037  0.000  0.033 -0.023 -0.027  0.079  0.003 -0.084  0.006 -0.024
AUA4  0.062 -0.024  0.040  0.063  0.067  0.015  0.105  0.002 -0.022  0.025
          AF2     AF3     AF4     MI1     MI2     MI3     MI4     MI5     AUA1    AUA2
AS1
AS2
AS3
AS4
AS5
AS6
AS7
AS8
AS9

```

```

AF1
AF2  0.000
AF3 -0.054  0.000
AF4 -0.015 -0.027  0.000
MI1  0.122  0.000  0.037  0.000
MI2 -0.090 -0.030  0.030 -0.018  0.000
MI3  0.004 -0.022 -0.030 -0.007  0.094  0.000
MI4  0.013 -0.012 -0.016  0.008 -0.010  0.030  0.000
MI5  0.067  0.093  0.028 -0.088 -0.017 -0.071  0.057  0.000
AUA1 0.023  0.012 -0.042  0.036  0.062  0.011 -0.010  0.022  0.000
AUA2 0.022 -0.008  0.019 -0.008  0.011 -0.051 -0.048  0.085  0.046  0.000
AUA3 0.046  0.011 -0.065 -0.038 -0.005 -0.046  0.036  0.071  0.023 -0.001
AUA4 0.045  0.017  0.033  0.054 -0.045 -0.049 -0.021  0.034 -0.024 -0.021
      AUA3  AUA4

AS1
AS2
AS3
AS4
AS5
AS6
AS7
AS8
AS9
AF1
AF2
AF3
AF4
MI1
MI2
MI3
MI4
MI5
AUA1
AUA2
AUA3  0.000
AUA4 -0.007  0.000

```

```
# lavaan::modindices(grmsAAW4fit)
```

## 10.6 Model Comparison

We evaluated two models (i.e., a unidimensional model and four-factor correlated model), which one is better? While, we have the narrative comparison (and would create a table with the comparisons) where the correlated traits/factor fit values ( $CFI = 0.976$ ,  $RMSEA = 0.022$  (90%CI[.000, .034]), and  $SRMR = .058$ ) outperformed the unidimensional ones ( $CFI = 0.81$ ,  $RMSEA = 0.061$

(90%CI[0.053, 0.069]), and SRMR = 0.067). We can formally compare them with statistical comparisons.

Easy are AIC and BIC comparisons where “smaller value wins.”

AIC GRMSAAW1: 16862.028 AIC GRMSAAW4: 16662.030

BIC GRMSAAW1: 17025.577 BIC GRMSAAW4: 16847.882

In both cases, the smaller values are for the more complex, 4-dimensional model. The interpretation is that the model with the smaller AIC/BIC values is most likely to replicate.

Additionally, the **chi-square difference test**,  $\chi^2_D$  can be used to compare nested models. Single-factor CFA models are nested under any other CFA model with two or more factors *for the same indicators*. This is because a one-factor model is a restricted version of any model with multiple factors. Our unidimensional GRMSAAW was nested under the 4-factor GRMSAAW model.

To calculate the chi-square difference test, we first grab the chi-square test values:

GRMSAAW1:  $\chi^2(209) = 444.451, p < .001$  GRMSAAW4:  $\chi^2(203) = 232.453, p = 0.076$

Given both sets of results we calculate:  $\chi^2(6) = 211.998, p < .05$  and determine that the two models are statistically significantly different. Given that the fit statistics are better for the single-order, correlated, four-factor model, we prefer that one.

How did I do that?

- Subtract the df
- Subtract the chi-square values
- Use a chi-square difference table to look up the chi-square critical value for a 6 df test
  - <https://www.itl.nist.gov/div898/handbook/eda/section3/eda3674.htm>, or
  - use this code to look it up `qchisq(p, df, lower.tail=FALSE)`
  - the critical value for our test is 12.592
- We conclude that the two models are statistically significantly different; our 4-factor model is preferred.

```
209 - 203 #subtract df
```

```
[1] 6
```

```
444.451 - 232.453 #subtract chi-square values
```

```
[1] 211.998
```

```
qchisq(0.05, 6, lower.tail = FALSE)
```

```
[1] 12.59159
```

Of course, there is a function for something this easy:

```
lavaan:::lavTestLRT(grmsAAW1fit, grmsAAW4fit)
```

#### Chi-Squared Difference Test

	Df	AIC	BIC	Chisq	Chisq diff	RMSEA	Df	diff			
grmsAAW4fit	203	16662	16848	232.45							
grmsAAW1fit	209	16862	17026	444.45		212	0.33606	6			
				Pr(>Chisq)							
grmsAAW4fit											
grmsAAW1fit				< 0.0000000000000022	***						
---											
Signif. codes:	0	'***'	0.001	'**'	0.01	'*'	0.05	'. '	0.1	' '	1

And we get the same result:  $\chi^2(6) = 212, p < .001$

#### 10.6.1 APA Results Section (so far...)

**Model testing.** To evaluate the models we, we used confirmatory factor analysis (CFA) in the R package, *lavaan* (v.0-6.9) with maximum likelihood estimation. Our sample size was 304. We selected fit criteria for their capacity to assess different aspects of the statistical analysis. As is common among SEM researchers, we reported the Chi-square goodness of fit ( $\chi^2$ ). This evaluates the discrepancy between the unrestricted sample matrix and the restricted covariance matrix. Although the associated  $p$  value indicates adequate fit when the value is non-significant, it is widely recognized that large sample size can result in a statistically significant  $p$  value [Byrne, 2016b]. The comparative fit index (CFI) is an incremental index, comparing the hypothesized model to the independent/baseline model. Adequate fit is determined when CFI values are at least .90 and perhaps higher than .95 [Kline, 2016]. The root mean square error of approximation (RMSEA) takes into account the error of approximation in the population and expresses it per degree of freedom. As such, the fit indicator considers the complexity of the model. Ideal values are equal to or less than .05, values less than .08 represent reasonable fit, and values between .08 and .10 represent mediocre fit. The standardized root mean residual (SRMR) is a standardized measure of the mean absolute covariance residual – the overall difference between the observed and predicted correlations. Values greater than .10 may indicate poor fit and inspection of residuals is then advised. Kline [2016] advised researchers to be cautious when using these criteria as strict cut-offs. Elements such as sample size and model complexity should be considered when evaluating fit. Table 1 provides a side-by-side comparison of the resulting parameter estimates and fit statistics; Figures 1 and 2 provide a graphic representation of the models tested.

Our first model was unidimensional where each of the 24 items loaded onto a single factor representing overall, gendered racial microaggressions towards Asian American women. The Chi-square index was statistically significant ( $\chi^2(209) = 444.451, p < .001$ ) indicating likely misfit. The CFI value of .81 indicated poor fit. In contrast, the RMSEA = 0.061, 90% CI(0.053, 0.069) and SRMR = 0.067 both fell within the ranges

of acceptability. The AIC and BIC values were 16862.028 and 17025.577, respectively, and will become useful in comparing subsequent models.

Our second model was a single-order, correlated traits model where each of the 22 items loaded onto one of four factors. Standardized pattern coefficients ranged between .37 and .60 on the AF factor, between .37 and .63 on the AS factor, between .33 and .56 on the MI factor, and between .43 and .60 on the AUA factor. The Chi-square index was statistically significant ( $\chi^2(203) = 232.453, p = 0.076$ ) indicating reasonable fit. The CFI value of 0.972 exceeded the recommendation of .95. The RMSEA = MSEA = 0.022 (90%CI[.000, 0.034]) was satisfactory. The SRMR value of 0.047 remained below the warning criteria of .10. The AIC and BIC values were 16662.030 and 16847.882, respectively.

The Chi-square difference test ( $\chi^2(6) = 211.998, p < .001$ ) was statistically significant and AIC and BIC values of the multidimensional value were lowest. Thus, we conclude the multidimensional model (i.e., the first-order, correlated factors model) is superior and acceptable for use in preliminary research and evaluation.

*We will continue to create, evaluate, and compare models in the next lesson.*

## 10.7 A concluding thought

Much like the children's game *Don't Break the Ice* we start with a full, saturated, matrix of sample data where every indicator/item is allowed to correlate/covary with every other.

As researchers, we specify a more parsimonious model where we fix some relations to zero and allow others to relate. In our GRMSAAW example, we allowed

- the AF items to relate via their relationship to the AF factor;
- the AS items to relate via their relationship to the AS factor;
- the MI items to relate via their relationship to the MI factor; and
- the AUA items to relate via their relationship to the AUA factor.
- we did not allow any of the items on any given factor to relate to the items on any other factor; these are *hard hypotheses* where we fix the relation to zero.

Our goal (especially via the chi-square test) is that we account for as much variance as possible through the specified relations that remain. Harkening to the *Don't Break the Ice* metaphor, we want the ice matrix to remain stable with as many ice cubes deleted as possible.



Source: <https://www.flickr.com/photos/arfsb/4407495674>

## 10.8 Practice Problems

In each of these lessons I provide suggestions for practice that allow you to select one or more problems that are graded in difficulty. In psychometrics, I strongly recommend that you have started with a dataset that has a minimum of three subscales and use it for all of the assignments in the OER. In any case, please plan to:

- Prepare the data frame for CFA.
- Specify and run unidimensional and single order (with correlated factors) models.
  - In the next chapter, you will add the specification, evaluation, and write-up of second-order and bifactor models.
- Narrate the adequacy of fit with  $\chi^2$ , CFI, RMSEA, SRMR
  - Write a mini-results section for each
- Compare model fit with  $\chi^2\Delta$ , AIC, and BIC.
- Write an APA style results sections with table(s) and figures.

### 10.8.1 Problem #1: Play around with this simulation.

The least complex is to change the random seed in the research and rework the problem demonstrated in the lesson. The results *should* map onto the ones obtained in the lecture.

### 10.8.2 Problem #2: Use simulated data from other lessons.

The second option involves utilizing one of the simulated datasets available in this OER. The [last lesson](#) in the OER contains three simulations that could be used for all of the statistics-based practice suggestions. Especially if you started with one of these examples in an earlier lesson, I highly recommend you continue with that.

Alternatively, Lewis and Neville's [2015] Gendered Racial Microaggressions Scale for Black Women was used in the lessons for exploratory factor analysis and Conover et al.'s [2017] Ableist Microaggressions Scale is used in the lesson on invariance testing. Both of these would be suitable for the CFA homework assignments.

### 10.8.3 Problem #3: Try something entirely new.

As a third option, you are welcome to use data to which you have access and is suitable for CFA. This could include other simulated data, data found on an open access repository, data from the ReCentering Psych Stats survey described in the [Qualtrics lesson](#), or your own data (presuming you have permission to use it).

### 10.8.4 Grading Rubric

Using the lecture and workflow (chart) as a guide, please work through all the steps listed in the proposed assignment/grading rubric.

Assignment Component	Points Possible	Points Earned
1. Prepare data for CFA (items only df, reverse-scored)	5	_____
2. Specify and run a unidimensional model	5	_____
3. Narrate adequacy of fit with $\chi^2$ , CFI, RMSEA, SRMR (write a mini-results section)	5	_____
4. Specify and run a single-order model with correlated factors	5	_____
5. Narrate adequacy of fit with $\chi^2$ , CFI, RMSEA, SRMR (write a mini-results section)	5	_____
6. Compare model fit with $\chi^2\Delta$ , AIC, BIC	5	_____
7. APA style results with table(s) and figure	5	_____
8. Explanation to grader	5	_____
<b>Totals</b>	<b>40</b>	_____

## 10.9 Homeworked Example

### Screencast Link

For more information about the data used in this homeworked example, please refer to the description and codebook located at the end of the [introduction](#) in first volume of ReCentering Psych Stats.

As a brief review, this data is part of an IRB-approved study, with consent to use in teaching demonstrations and to be made available to the general public via the open science framework. Hence, it is appropriate to use in this context. You will notice there are student- and teacher- IDs. These numbers are not actual student and teacher IDs, rather they were further re-identified so that they could not be connected to actual people.

Because this is an actual dataset, if you wish to work the problem along with me, you will need to download the [ReC.rds](#) data file from the Worked\_Examples folder in the ReC\_Psychometrics project on the GitHub.

The course evaluation items can be divided into three subscales:

- **Valued by the student** includes the items: ValObjectives, IncrUnderstanding, IncrInterest
- **Traditional pedagogy** includes the items: ClearResponsibilities, EffectiveAnswers, Feedback, ClearOrganization, ClearPresentation
- **Socially responsive pedagogy** includes the items: InclusvClassrm, EquitableEval, MultPerspectives, DEIIntegration

In this Homeworked Example I will conduct and compare single order, unidimensional and correlated traits CFA models. My hope is that the results will support my solution of three dimensions: valued-by-the-student, traditional pedagogy, socially responsive pedagogy. This is the first part of a larger two-part suggestion for practice. These steps will be repeated in the next lesson's homeworked example. While somewhat redundant, I am hopeful that the second set will provide a fairly complete set of code for someone who is analyzing their own data from the beginning.

### 10.9.1 Prepare data for CFA (items only df, reverse-scored)

We can upload the data from the .rds file. The file should be in the same folder as the .rmd file. I've named the df object that holds the data "big."

```
big <- readRDS("ReC.rds")
```

For the demonstration of CFA first order models, I will just pull in the items that I believe go onto the three factors.

```
library(tidyverse)
items <- big %>%
  dplyr::select(ValObjectives, IncrUnderstanding, IncrInterest, ClearResponsibilities,
               EffectiveAnswers, Feedback, ClearOrganization, ClearPresentation,
               MultPerspectives, InclusvClassrm, DEIIntegration, EquitableEval)
```

Let's quickly check the structure. The variables should be numeric or integer.

```
str(items)
```

```
Classes 'data.table' and 'data.frame': 310 obs. of 12 variables:
```

```
$ ValObjectives      : int  5 5 4 4 5 5 5 5 5 4 5 ...
$ IncrUnderstanding : int  2 3 4 3 4 4 5 2 4 5 ...
$ IncrInterest       : int  5 3 4 2 4 3 5 3 2 5 ...
$ ClearResponsibilities: int  5 5 4 4 5 4 5 4 4 5 ...
$ EffectiveAnswers   : int  5 3 5 3 5 3 4 3 2 3 ...
$ Feedback           : int  5 3 4 2 5 NA 5 4 4 5 ...
$ ClearOrganization  : int  3 4 3 4 4 4 5 4 4 5 ...
$ ClearPresentation  : int  4 4 4 2 5 3 4 4 4 5 ...
$ MultPerspectives    : int  5 5 4 5 5 4 5 5 5 5 ...
$ InclusvClassrm     : int  5 5 5 5 5 4 5 5 4 5 ...
$ DEIIntegration      : int  5 5 5 5 5 4 5 5 5 5 ...
$ EquitableEval       : int  5 5 3 5 5 3 5 5 3 5 ...
- attr(*, ".internal.selfref")=<externalptr>
```

### 10.9.2 Specify and run a unidimensional model

First we map the relations we want to analyze.

```
uniD <- "CourseEvals =~ ValObjectives + IncrUnderstanding + IncrInterest + ClearResponsibilities"
```

We analyze the relations by naming that object in our *lavaan* code.

```
set.seed(240311)
uniDfit <- lavaan::cfa(uniD, data = items)
lavaan::summary(uniDfit, fit.measures = TRUE, standardized = TRUE, rsquare = TRUE)
```

*lavaan* 0.6.17 ended normally after 32 iterations

Estimator	ML
Optimization method	NLMINB
Number of model parameters	24
Used	Total
Number of observations	267      310

Model Test User Model:

Test statistic	344.973
Degrees of freedom	54
P-value (Chi-square)	0.000

Model Test Baseline Model:

Test statistic	1940.157
Degrees of freedom	66
P-value	0.000

User Model versus Baseline Model:

Comparative Fit Index (CFI)	0.845
Tucker-Lewis Index (TLI)	0.810

Loglikelihood and Information Criteria:

Loglikelihood user model (H0)	-3038.064
Loglikelihood unrestricted model (H1)	-2865.578
Akaike (AIC)	6124.129
Bayesian (BIC)	6210.223
Sample-size adjusted Bayesian (SABIC)	6134.129

Root Mean Square Error of Approximation:

RMSEA	0.142
90 Percent confidence interval - lower	0.128
90 Percent confidence interval - upper	0.157
P-value H_0: RMSEA <= 0.050	0.000
P-value H_0: RMSEA >= 0.080	1.000

Standardized Root Mean Square Residual:

SRMR	0.074
------	-------

Parameter Estimates:

Standard errors	Standard
Information	Expected
Information saturated (h1) model	Structured

Latent Variables:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
CourseEvals =~						
ValObjectives	1.000				0.309	0.515
IncrUndrstndng	1.715	0.223	7.684	0.000	0.530	0.642
IncrInterest	2.142	0.269	7.971	0.000	0.662	0.685
ClearRspnsblts	2.065	0.239	8.652	0.000	0.638	0.808
EffectivAnswrs	2.105	0.244	8.617	0.000	0.650	0.800
Feedback	2.143	0.259	8.285	0.000	0.662	0.738
ClearOrganiztn	2.678	0.314	8.516	0.000	0.828	0.780
ClearPresenttn	2.521	0.285	8.832	0.000	0.779	0.846
MultPerspectvs	2.067	0.246	8.392	0.000	0.639	0.757
InclusvClassrm	1.246	0.170	7.324	0.000	0.385	0.592
DEIIntegration	1.015	0.174	5.820	0.000	0.314	0.424
EquitableEval	1.435	0.179	8.027	0.000	0.443	0.694

Variances:

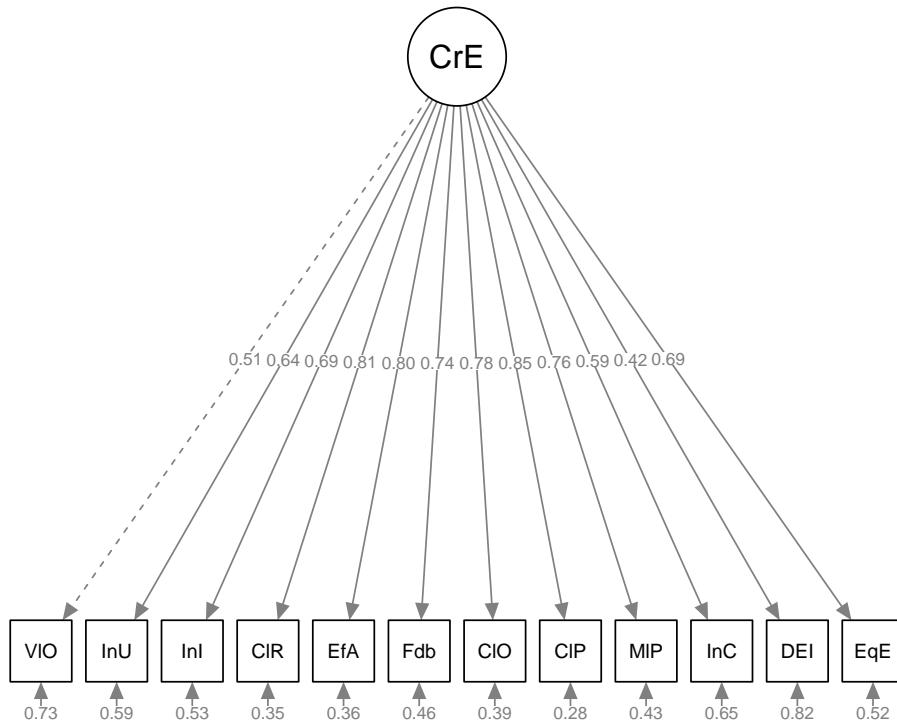
	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
.ValObjectives	0.265	0.024	11.254	0.000	0.265	0.735
.IncrUnderstndng	0.401	0.037	10.970	0.000	0.401	0.588
.IncrInterest	0.494	0.046	10.815	0.000	0.494	0.530
.ClearRspnsblts	0.217	0.022	9.983	0.000	0.217	0.348
.EffectivAnswnrs	0.237	0.024	10.060	0.000	0.237	0.359
.Feedback	0.367	0.035	10.555	0.000	0.367	0.455
.ClearOrganiztn	0.439	0.043	10.250	0.000	0.439	0.391
.ClearPresenttn	0.242	0.026	9.446	0.000	0.242	0.285
.MultPerspectvs	0.304	0.029	10.431	0.000	0.304	0.427
.InclusvClassrm	0.275	0.025	11.104	0.000	0.275	0.649
.DEIIntegration	0.449	0.040	11.372	0.000	0.449	0.820
.EquitableEval	0.211	0.020	10.777	0.000	0.211	0.518
CourseEvals	0.096	0.022	4.381	0.000	1.000	1.000

R-Square:

	Estimate
ValObjectives	0.265
IncrUnderstndng	0.412
IncrInterest	0.470
ClearRspnsblts	0.652
EffectivAnswnrs	0.641
Feedback	0.545
ClearOrganiztn	0.609
ClearPresenttn	0.715
MultPerspectvs	0.573
InclusvClassrm	0.351
DEIIntegration	0.180
EquitableEval	0.482

Let's plot the results to see if the figure resembles what we intended to specify.

```
semPlot::semPaths(uniDfit, layout = "tree", style = "lisrel", what = "col",
  whatLabels = "stand")
```



## Narrate adequacy of fit with  $\chi^2$ , CFI, RMSEA, SRMR (write a mini-results section)

**Model testing.** To evaluate the models we, we used confirmatory factor analysis (CFA) in the R package, lavaan (v.0-6.17) with maximum likelihood estimation. Our sample size was 267. We selected fit criteria for their capacity to assess different aspects of the statistical analysis. As is common among SEM researchers, we reported the Chi-square goodness of fit ( $\chi^2$ ). This evaluates the discrepancy between the unrestricted sample matrix and the restricted covariance matrix. Although the associated  $p$ -value indicates adequate fit when the value is non-significant, it is widely recognized that a large sample size can result in a statistically significant  $p$  value (Byrne, 2016b). The comparative fit index (CFI) is an incremental index, comparing the hypothesized model to an independent/baseline model. Acceptable fit occurs when values are at least .90 and perhaps higher than .95 (Kline, 2016). The root mean square error of approximation (RMSEA) takes into account the error of approximation in the population and expresses it per degree of freedom. As such, the fit indicator considers the complexity of the model. Ideal values are equal to or less than .05, values less than .08 represent reasonable fit, and values between .08 and .10 represent mediocre fit. The standardized root mean residual (SRMR) is a standardized measure of the mean absolute covariance residual –

the overall difference between the observed and predicted correlations. Values greater than .10 may indicate poor fit and inspection of residuals is then advised. Kline (2016) advised researchers to be cautious when using these criteria as strict cut-offs. Elements such as sample size and model complexity should be considered when evaluating fit.

Our first model was unidimensional where each of the 12 items loaded onto a single factor representing overall course evaluations. The Chi-square index was statistically significant ( $\chi^2(54) = 344.97, p < .001$ ) indicating likely misfit. The CFI value of .85 indicated poor fit. The RMSEA = .14 (90% CI [.13, .16]) suggested serious problems. The SRMR value of .07 was below the warning criteria of .10. The AIC and BIC values were 6124.13 and 6134.13, respectively, and will become useful in comparing subsequent models.

The *tidySEM* package has some useful tools to export the results to .csv files. This first set of code exports the fit indices.

```
uniDfitStats <- tidySEM::table_fit(uniDfit)
uniDfit_paramEsts <- tidySEM::table_results(uniDfit, digits = 3, columns = NULL)
# uniDfitStats uniDfit_paramEsts
```

We can write each of these to a .csv file that will be stored in the same folder as your .rmd file.

```
write.csv(uniDfitStats, file = "uniDfitStats.csv")
write.csv(uniDfit_paramEsts, file = "uniDfit_paramEsts.csv")
```

### 10.9.3 Specify and run a single-order model with correlated factors

First we map the relations we want to analyze.

```
corrF <- 'TradPed =~ ClearResponsibilities + EffectiveAnswers + Feedback + ClearOrganization
           Valued =~ ValObjectives + IncrUnderstanding + IncrInterest
           SCRPed =~ MultPerspectives + InclusvClassrm + DEIIntegration + EquitableEval

           TradPed~~Valued
           TradPed~~SCRPed
           Valued~~SCRPed
           '
```

Next we run the analysis.

```
set.seed(240311)
corrF_fit <- lavaan::cfa(corrF, data = items)
lavaan::summary(corrF_fit, fit.measures = TRUE, standardized = TRUE, rsquare = TRUE)
```

lavaan 0.6.17 ended normally after 42 iterations

Estimator	ML
Optimization method	NLMINB
Number of model parameters	27
	Used      Total
Number of observations	267      310

Model Test User Model:

Test statistic	224.795
Degrees of freedom	51
P-value (Chi-square)	0.000

Model Test Baseline Model:

Test statistic	1940.157
Degrees of freedom	66
P-value	0.000

User Model versus Baseline Model:

Comparative Fit Index (CFI)	0.907
Tucker-Lewis Index (TLI)	0.880

Loglikelihood and Information Criteria:

Loglikelihood user model (H0)	-2977.975
Loglikelihood unrestricted model (H1)	-2865.578
Akaike (AIC)	6009.951
Bayesian (BIC)	6106.807
Sample-size adjusted Bayesian (SABIC)	6021.201

Root Mean Square Error of Approximation:

RMSEA	0.113
90 Percent confidence interval - lower	0.098
90 Percent confidence interval - upper	0.128
P-value H_0: RMSEA <= 0.050	0.000
P-value H_0: RMSEA >= 0.080	1.000

Standardized Root Mean Square Residual:

SRMR	0.061
------	-------

Parameter Estimates:

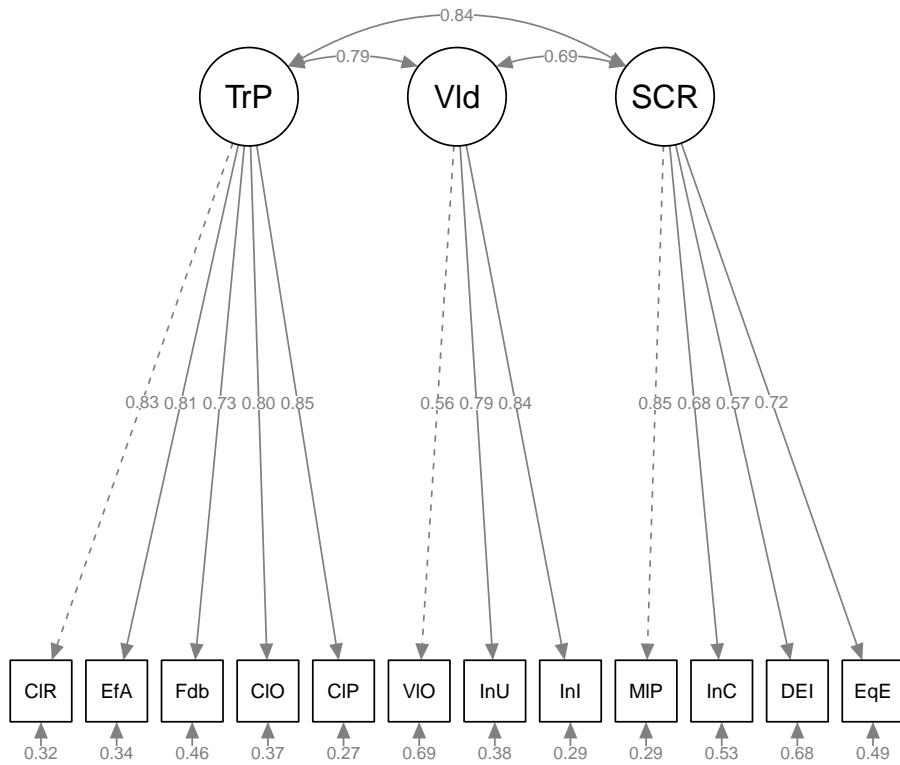
	Standard errors	Standard				
	Information	Expected				
	Information saturated (h1) model	Structured				
<b>Latent Variables:</b>						
	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
TradPed =~						
ClearRspnsblts	1.000				0.652	0.826
EffectivAnswrs	1.015	0.065	15.606	0.000	0.662	0.815
Feedback	1.010	0.075	13.481	0.000	0.659	0.735
ClearOrganiztn	1.295	0.086	15.106	0.000	0.845	0.797
ClearPresenttn	1.204	0.072	16.680	0.000	0.785	0.853
Valued =~						
ValObjectives	1.000				0.334	0.557
IncrUndrstndng	1.942	0.223	8.717	0.000	0.649	0.786
IncrInterest	2.438	0.273	8.932	0.000	0.815	0.844
SCRPed =~						
MultPerspectvs	1.000				0.713	0.846
InclusvClassrm	0.622	0.053	11.672	0.000	0.444	0.682
DEIIntegration	0.589	0.063	9.365	0.000	0.420	0.567
EquitableEval	0.642	0.052	12.410	0.000	0.458	0.717
<b>Covariances:</b>						
	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
TradPed ~~						
Valued	0.171	0.026	6.640	0.000	0.785	0.785
SCRPed	0.391	0.045	8.677	0.000	0.841	0.841
Valued ~~						
SCRPed	0.164	0.026	6.254	0.000	0.688	0.688
<b>Variances:</b>						
	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
.ClearRspnsblts	0.199	0.021	9.456	0.000	0.199	0.319
.EffectivAnswrs	0.222	0.023	9.618	0.000	0.222	0.336
.Feedback	0.371	0.036	10.415	0.000	0.371	0.460
.ClearOrganiztn	0.410	0.042	9.855	0.000	0.410	0.365
.ClearPresenttn	0.232	0.026	8.939	0.000	0.232	0.273
.ValObjectives	0.248	0.023	10.650	0.000	0.248	0.690
.IncrUndrstndng	0.260	0.032	8.041	0.000	0.260	0.382
.IncrInterest	0.268	0.043	6.308	0.000	0.268	0.288
.MultPerspectvs	0.203	0.029	7.052	0.000	0.203	0.285
.InclusvClassrm	0.226	0.023	10.028	0.000	0.226	0.534
.DEIIntegration	0.371	0.035	10.734	0.000	0.371	0.678
.EquitableEval	0.198	0.020	9.685	0.000	0.198	0.486
TradPed	0.426	0.053	8.085	0.000	1.000	1.000
Valued	0.112	0.024	4.595	0.000	1.000	1.000
SCRPed	0.509	0.063	8.039	0.000	1.000	1.000

R-Square:

	Estimate
ClearRspnsblts	0.681
EffectivAnswrs	0.664
Feedback	0.540
ClearOrganiztn	0.635
ClearPresenttn	0.727
ValObjectives	0.310
IncrUndrstndng	0.618
IncrInterest	0.712
MultPerspectvs	0.715
InclusvClassrm	0.466
DEIIntegration	0.322
EquitableEval	0.514

Plotting the results. Does it look like what we intended to specify?

```
semPlot::semPaths(corrF_fit, layout = "tree", style = "lisrel", what = "col",
whatLabels = "stand")
```



#### 10.9.4 Narrate adequacy of fit with $\chi^2$ , CFI, RMSEA, SRMR (write a mini-results section)

Our second model was a single-order, correlated traits model where each of the 12 items loaded onto one of four factors. Standardized pattern coefficients ranged between .74 and .85 on the TradPed factor, between .56 and .84 on the Valued factor, and between .57 and .85 on the SCRPed factor. The Chi-square index was statistically significant ( $\chi^2(51) = 224.795, p < 0.001$ ) indicating some degree of misfit. The CFI value of .91 fell below the recommendation of .95. The RMSEA = .113 (90% CI [.098, .128]) was higher than recommended. The SRMR value of .061 remained below the warning criteria of .10. The AIC and BIC values were 6009.95 and 6021.20, respectively.

Code for saving the results as a .csv file.

```
corrFitStats <- tidySEM::table_fit(corrF_fit)
corrF_paramEsts <- tidySEM::table_results(corrF_fit, digits = 3, columns = NULL)
```

```
corrFCorrs <- tidySEM::table_cors(corrF_fit, digits = 3)
# to see each of the tables, remove the hashtable corrFitStats
# corrF_paramEsts corrFCorrs
```

Next, I export them.

```
write.csv(corrFitStats, file = "corrFitStats.csv")
write.csv(corrF_paramEsts, file = "corrF_paramEsts.csv")
write.csv(corrFCorrs, file = "corrFCorrs.csv")
```

### 10.9.5 Compare model fit with $\chi^2\Delta$ , AIC, BIC

```
set.seed(240311)
lavaan::lavTestLRT(uniDfit, corrF_fit)
```

Chi-Squared Difference Test

	Df	AIC	BIC	Chisq	Chisq diff	RMSEA	Df diff				
corrF_fit	51	6010.0	6106.8	224.79							
uniDfit	54	6124.1	6210.2	344.97	120.18	0.38248	3				
				Pr(>Chisq)							
corrF_fit											
uniDfit				< 0.0000000000000022	***						
---											
Signif. codes:	0	'***'	0.001	'**'	0.01	'*'	0.05	'. '	0.1	' '	1

The Chi-square difference test ( $\chi^2(3) = 120.18, p < 0.001$ ) was statistically significant indicating that the two models were statistically significantly different. The AIC and BIC values of the correlated factors model were the lowest. Thus, we conclude the first-order, correlated factors model is superior and acceptable for use in preliminary research and evaluation.

### 10.9.6 APA style results with table(s) and figure

Because we have written mini-results throughout, we can assemble them into a full results section. Keep in mind that most CFA models will continue testing multidimensional models. Thus, the entire analysis continues in the next lesson and associated practice problem.

**Model testing.** To evaluate the models we, we used confirmatory factor analysis (CFA) in the R package, lavaan (v.0-6.17) with maximum likelihood estimation. Our sample size was 267. We selected fit criteria for their capacity to assess different aspects of the statistical analysis. As is common among SEM researchers, we reported the Chi-square goodness of fit (2). This evaluates the discrepancy between the unrestricted

sample matrix and the restricted covariance matrix. Although the associated  $p$ -value indicates adequate fit when the value is non-significant, it is widely recognized that a large sample size can result in a statistically significant  $p$  value (Byrne, 2016b). The comparative fit index (CFI) is an incremental index, comparing the hypothesized model to an independent/baseline model. Acceptable fit occurs when values are at least .90 and perhaps higher than .95 (Kline, 2016). The root mean square error of approximation (RMSEA) takes into account the error of approximation in the population and expresses it per degree of freedom. As such, the fit indicator considers the complexity of the model. Ideal values are equal to or less than .05, values less than .08 represent reasonable fit, and values between .08 and .10 represent mediocre fit. The standardized root mean residual (SRMR) is a standardized measure of the mean absolute covariance residual – the overall difference between the observed and predicted correlations. Values greater than .10 may indicate poor fit and inspection of residuals is then advised. Kline (2016) advised researchers to be cautious when using these criteria as strict cut-offs. Elements such as sample size and model complexity should be considered when evaluating fit.

Our first model was unidimensional where each of the 12 items loaded onto a single factor representing overall course evaluations. The Chi-square index was statistically significant ( $\chi^2(54) = 344.97, p < .001$ ) indicating likely misfit. The CFI value of .85 indicated poor fit. The RMSEA = .14 (90% CI [.13, .16]) suggested serious problems. The SRMR value of .07 was below below the warning criteria of .10. The AIC and BIC values were 6124.13 and 6134.13, respectively, and will become useful in comparing subsequent models.

Our second model was a single-order, correlated traits model where each of the 12 items loaded onto one of four factors. Standardized pattern coefficients ranged between .74 and .85 on the TradPed factor, between .56 and .84 on the Valued factor, and between .57 and .85 on the SCRPed factor. The Chi-square index was statistically significant ( $\chi^2(51) = 224.795, p < 0.001$  indicating some degree of misfit. The CFI value of .91 fell below the recommendation of .95. The RMSEA = .113 (90% CI [.098, .128]) was higher than recommended. The SRMR value of .061 remained below the warning criteria of .10. The AIC and BIC values were 6009.95 and 6021.20, respectively.

The Chi-square difference test ( $\chi^2(3) = 120.18, p < 0.001$ ) was statistically significant indicating that the two models were statistically significantly different. The AIC and BIC values of the the correlated factors model were the lowest. Thus, we conclude the first-order, correlated factors model is superior and acceptable for use in preliminary research and evaluation.

### 10.9.7 Explanation to grader

# Chapter 11

## CFA: Hierarchical and Nested Models

### [Screencasted Lecture Link](#)

This is the second lecture in our series on confirmatory factor analysis (CFA). In this lesson we will compare first-order structures (with correlated uncorrelated factors) to second-order and bifactor structures. Modification indices will allow us to tweak each model's fit. We will also determine and track the identification status of models, including nested/nesting models and examining issues of equivalent models.

### 11.1 Navigating this Lesson

The lecture is just under two hours. I would add another two-to-three hours to work through and digest the materials.

While the majority of R objects and data you will need are created within the R script that sources the chapter, occasionally there are some that cannot be created from within the R framework. Additionally, sometimes links fail. All original materials are provided at the [GitHub site](#) that hosts the book. More detailed guidelines for ways to access all these materials are provided in the OER's [introduction](#).

#### 11.1.1 Learning Objectives

Focusing on this week's materials, make sure you can:

- Specify single order (correlated and uncorrelated), second order, and bifactor models.
- Interpret model adequacy and fit.
- Compare models on the basis of statistical criteria.
- Determine which (among models) is the nested model.
- Memorize which model (nested or nesting) will have better fit (without looking at the results).
- Determine whether or not models (or alterations to their specification) remain statistically identified.

### 11.1.2 Planning for Practice

In each of these lessons I provide suggestions for practice that allow you to select one or more problems that are graded in difficulty. The least complex is to change the random seed and rework the problem demonstrated in the lesson. The results *should* map onto the ones obtained in the lecture.

The second option involves utilizing one of the simulated datasets available in this OER. The [last lesson](#) in the OER contains three simulations that could be used for all of the statistics-based practice suggestions. Especially if you started with one of these examples in an earlier lesson, I highly recommend you continue with that.

Alternatively, Lewis and Neville's [2015] Gendered Racial Microaggressions Scale for Black Women was used in the lessons for exploratory factor analysis and Conover et al.'s [2017] Ableist Microaggressions Scale is used in the lesson on invariance testing. Both of these would be suitable for the CFA homework assignments.

As a third option, you are welcome to use data to which you have access and is suitable for CFA. This could include other simulated data, data found on an open access repository, data from the ReCentering Psych Stats survey described in the [Qualtrics lesson](#), or your own data (presuming you have permission to use it).

The suggestion for practice spans the [prior chapter](#) and this one . For this combination assignment, you should plan to:

- Prepare the data frame for CFA.
- Specify and run unidimensional, single order (with correlated factors), second-order, and bifactor models.
- Narrate the adequacy of fit with  $\chi^2$ , CFI, RMSEA, SRMR.
  - Write a mini-results section for each.
- Compare model fit with  $\chi^2\Delta$ , AIC, and BIC.
- Write an APA style results sections with table(s) and figures.

In preparing this chapter, I drew heavily from the following resource(s). Other resources are cited (when possible, linked) in the text with complete citations in the reference list.

- Byrne, B. M. (2016). Application 3: Testing the Factorial Validity of Scores from a Measurement Scale (Second-Order CFA model). Chapter 5. In Structural Equation Modeling with AMOS: Basic Concepts, Applications, and Programming, Third Edition. Taylor & Francis Group. <http://ebookcentral.proquest.com/lib/spu/detail.action?docID=4556523>
- Dekay, Nicole (2021). Quick Reference Guide: The statistics for psychometrics <https://www.humanalysts.com/quick-reference-guide-the-statistics-for-psychometrics>
- Flora, D. B. (2020). Your Coefficient Alpha Is Probably Wrong, but Which Coefficient Omega Is Right? A Tutorial on Using R to Obtain Better Reliability Estimates. Advances in Methods and Practices in Psychological Science, 3(4), 484–501. <https://doi.org/10.1177/2515245920951747>

- Kline, R. (2016). Principles and practice of structural equation modeling (Fourth ed., Methodology in the social sciences). New York: The Guilford Press.
  - Chapter 6: Specification of Observed-Variable (Path Models)
  - Chapter 7: Identification of Observed-Variable (Path) Models \*
  - Chapter 9: Specification and Identification of Confirmatory Factor Analysis Models
  - Chapter 13: Analysis of Confirmatory Factor Analysis Models
- Rosseel, Y. (2019). The *lavaan* tutorial. Belgium: Department of Data Analysis, Ghent University. <http://lavaan.ugent.be/tutorial/tutorial.pdf>

### 11.1.3 Packages

The packages used in this lesson are embedded in this code. When the hashtags are removed, the script below will (a) check to see if the following packages are installed on your computer and, if not (b) install them.

```
#will install the package if not already installed
#if(!require(lavaan)){install.packages("lavaan")}
#if(!require(semPlot)){install.packages("semPlot")}
#if(!require(psych)){install.packages("psych")}
#if(!require(semTable)){install.packages("semTable")}
#if(!require(semTools)){install.packages("semTools")}
```

## 11.2 CFA Workflow

Below is a screenshot of a CFA workflow. The original document is in the [GitHub site](#) that hosts the ReCentering Psych Stats: Psychometrics OER.

Because the intended audience for the ReCentering Psych Stats OER is the scientist-practitioner-advocate, this lesson focuses on the typical workflow and associated decisions. As you might guess, the details of CFA can be quite complex and require more investigation and decision-making in models that pose more complexity or empirical challenges. The following are the general steps in a CFA.

- Creating an items only dataframe where any items are scaled in the same direction (e.g., negatively worded items are reverse-scored).
- Determining a factor structure that is *identified*, that is
  - a single factor (unidimensional) model has at least three items/indicators
  - a multidimensional model with at least two items per factor
- Specify a series of models, these typically include
  - a unidimensional model (all items on a single factor)
  - a single order structure with correlated factors
  - a second order structure
  - a bifactor structure

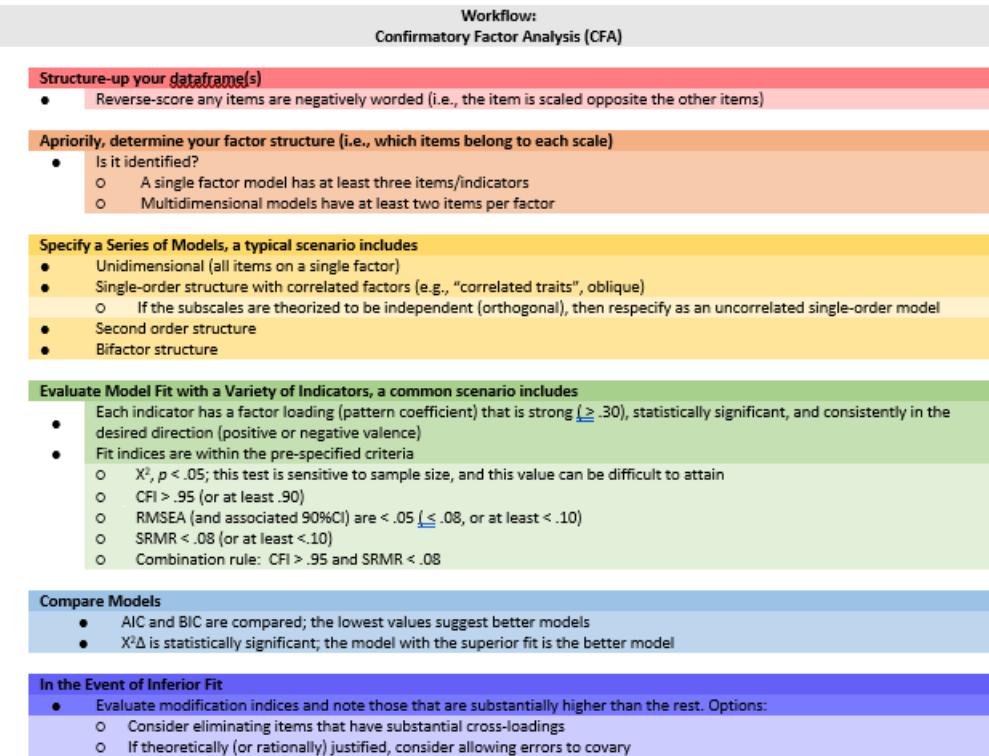


Figure 11.1: Image of a workflow for specifying and evaluating a confirmatory factor analytic model

- Evaluate model fit with a variety of indicators
  - factor loadings
  - fit indices
- Compare models
- In the event of poor model fit, investigate modification indices and consider respecification
  - eliminating items
  - changing factor membership
  - allowing errors to covary

### 11.3 Another Look at Varying Factor Structures

In this lecture we move into second-order and bifactor models, let's look again factor structures, considering unidimensional, first-order, and second-order variations.

Models A and B are **first-order models**. Note that all factors are on a single plane.

- Model A is unidimensional. Each item is influenced by a single common factor, and defined by a single term that includes systematic and random error. Note that there is only one *systematic* source of variance for each item AND it is from a single source: F1.

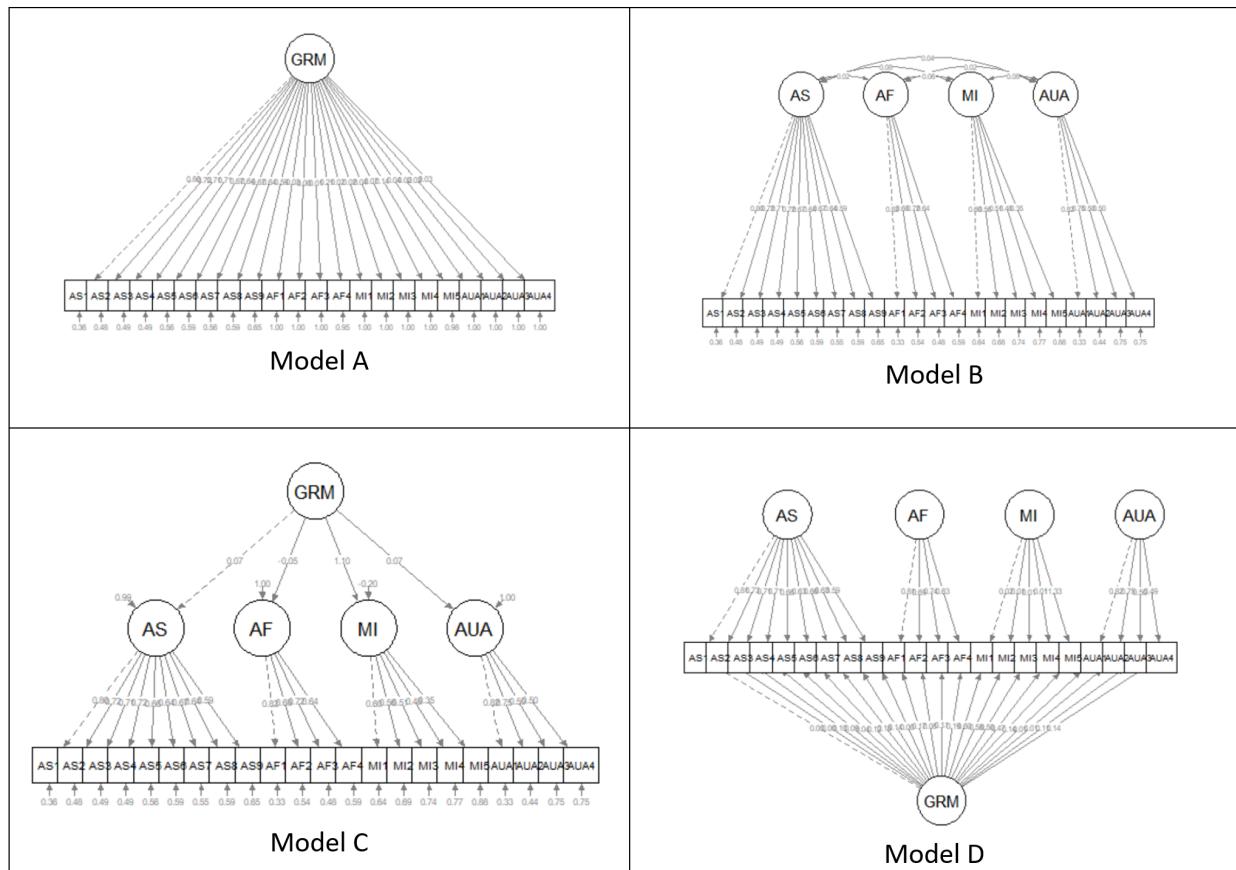


Figure 11.2: Image of first-order, second-order, and bifactor factor structures

- Model B is often referred to as a “correlated traits” model. Here, the larger construct is separated into distinct-yet-correlated elements. The variance of each item is assumed to be a weighted linear function of two or more common factors.
- Model C is a second-order factor structure. Rather than merely being correlated, factors are related because they share a common cause. In this model, the second-order factor *explains* why three or more traits are correlated. Note that there is no direct relationship between the item and the target construct. Rather, the relationship between the second-order factor and each item is mediated through the primary factor (yes, an indirect effect!).
- Model D is a bifactor structure. Here, each item loads on a general factor. This general factor (bottom row) reflects what is common among the items and represents the individual differences on the target dimension that a researcher is most interested in. Group factors (top row) are now specified as *orthogonal*. The group factors represent common factors measured by the items that explain item response variation not accounted for by the general factor. In some research scenarios, the group factors are termed “nuisance” dimensions. That is, that which they have in common interferes with measuring the primary target of interest.

## 11.4 Revisiting Model Identification

Model identification means it is *theoretically possible* for a statistical analysis to derive a unique estimate of every model parameter.

*Theoretical* emphasizes that identification is a property of the model and not the data; that is, it doesn’t matter if the sample size is 100 or 10,000.

CFA has the same general requirements for identification as other forms of SEM:

1. Every latent variable (including errors) must be scaled; and
2. Model degrees of freedom must be at least zero ( $df_M \leq 0$ ) (aka “the counting rule”; this means that there must be at least as many observations as there are free parameters)

### 11.4.1 Identification Status

**Underidentified** (or underdetermined) models violate the counting rule because there are more free parameters than observations.

For example, solve this equation:

$$a + b = 6.$$

There are an infinite number of solutions:  $4 + 2$ ,  $3 + 3$ ,  $2.5 + 3.5$ ...and so on to  $\infty$ . When the computing algorithm tries to solve this problem, it will fail to converge.

The parallel scenario in an SEM/CFA model with more free parameters than observations would have *negative df*.

**Just-identified** (or just-determined) models have a single unique solution,

$$(df_M = 0)$$

For example, for this set of equations:

$$\begin{aligned} a + b &= 6 \\ 2a + b &= 10 \end{aligned}$$

The only answer is:  $a = 4, b = 2$

**Overidentified** (or overdetermined) models have more observations than free parameters. That is:

$$df_M > 0$$

For example, solve for this set of equations:

$$\begin{aligned} a + b &= 6 \\ 2a + b &= 10 \\ 3a + b &= 12 \end{aligned}$$

There is no single solution that satisfies all three formulas, but there is a way to find a unique solution. We can impose a statistical criterion that leads to the *overidentified/overdetermined* circumstance with more observations than free parameters. For example, we could impose the *least squares criterion* (from regression, but with no intercept/constant in the prediction equation). The constraint (instruction) would read:

*Find values of  $a$  and  $b$  that yield total scores such that the sum of squared differences between the observations (6, 10, 12) and these total scores is as small as possible (and also unique).*

In this case, answers are  $a = 3.00, b = 3.33$  and the solutions are 6.44, 9.33, 12.33. While the solution doesn't perfectly reproduce the data, it facilitates model testing.

**The bad news** is that SEM/CFA computer tools are generally not helpful in determining whether a model is identified or not. Why? Computers are great at numerical processing, but not symbolic processing (needed for determining identification status). This means that we, the researchers, must learn the *identification heuristics* to determine the model's degree of identification.

*Need a break already?* My favorite scene during **The Imitation Game** parallels issues of identification, iterations, and convergence. The Turing machine runs and runs until its users can feed it proper start values so that it finally converges on a solution.

Kenny [Kenny, 2012] provides some helpful guidelines in determining model identification with the calculation of *knowns* and *unknowns*. In a standard CFA/SEM specification, *knowns* are the number of covariances between all the variables in the model,  $(k(k+1))/2$ , where  $k$  is the number of variables in the model. *Unknowns* are the *free parameters* that must be calculated. These include paths; covariances between exogenous variables, between disturbances (error terms), and between exogenous variables and disturbances (error terms); variances of the exogenous variables; and disturbances (error terms) of the endogenous variables (minus the number of linear constraints).

- \* If  $\$knowns \lt unknowns$  then the model is \*under-identified\*
- \* If  $\$knowns = unknowns$  then the model is \*just-identified\*
- \* If  $\$knowns \gt unknowns$  then the model is \*overidentified\*

### 11.4.2 Identification in Practice

It is essential that every latent variable (including errors) must be scaled such that the degrees of freedom for the model be greater than or equal to zero ( $df_M \geq 0$ ).

Operationally, in a standard CFA model:

1. A single factor model needs at least 3 indicators for the single factor.
2. Factor models with more than one factor require at least two or more indicators per factor.
  - For purposes of identification, more is better with 3-5 being recommended.

(Among other things) *nonstandard* models occur when:

- errors are allowed to correlate/covary
- *complex indicators* are defined by more than one factor

We will return to these as we encounter them later in today's lecture. Essentially, we will need to "subtract" 1 df for every parameter we "free" to covary. This is because we then need to estimate it and it becomes "unknown."

**Empirical underidentification** is also a threat. This means, the model fails to converge because of the characteristics of the data. For example, perhaps we specified model on the cusp of identification: 2 factors, correlated, with 2 indicators each. If in fact, the data did not support the correlation between the two factors...because of the "just barely" identified circumstance, you may receive an "empirically underidentified" solution.

Today we are going to specify second-order and bifactor models. As we do each, we will address these issues of model identification.

## 11.5 Research Vignette

This lesson's research vignette emerges from Keum et al's Gendered Racial Microaggressions Scale for Asian American Women (GRMSAAW; [Keum et al., 2018]). The article reports on two separate studies that comprised the development, refinement, and psychometric evaluation of two, parallel, versions (stress appraisal, frequency) of scale. I simulated data from the final construction of the frequency version as the basis of the lecture. If the scale looks somewhat familiar it is because the authors used the Gendered Racial Microaggressions Scale for Black Women [Lewis and Neville, 2015] as a model.

Keum et al. [2018] reported support for a total scale score (22 items) and four subscales. Below, I list the four subscales, their number of items, and a single example item. At the outset, let me provide a content advisory For those who hold this particular identity (or related identities) the content in the items may be upsetting. In other lessons, I often provide a variable name that gives an indication of the primary content of the item. In the case of the GRMSAAW, I will simply provide an abbreviation of the subscale name and its respective item number. This will allow us to easily inspect the alignment of the item with its intended factor, and hopefully minimize discomfort. If you are not a member of this particular identity, I encourage you to learn about these microaggressions by reading the article in its entirety. Please do not ask members of this group to explain why these microaggressions are harmful or ask if they have encountered them.

There are 22 items on the GRMSAAW scale. Using the same item stems, the authors created two scales. One assesses frequency of the event, the second assesses the degree of stressfulness. I simulated data from the stressfulness scale. Its Likert style scaling included: 0 (*not at all stressful*), 1(*slightly stressful*), 2(*somewhat stressful*), 3(*moderately stressful*), 4(*very stressful*), and 5(*extremely stressful*).

The four factors, number of items, and sample item are as follows:

- Ascribed Submissiveness (9 items)
  - Others expect me to be submissive. (AS1)
  - Others have been surprised when I disagree with them. (AS2)
  - Others take my silence as a sign of compliance. (AS3)
  - Others have been surprised when I do things independent of my family. (AS4)
  - Others have implied that AAW seem content for being a subordinate. (AS5)
  - Others treat me as if I will always comply with their requests. (AS6)
  - Others expect me to sacrifice my own needs to take care of others (e.g., family, partner) because I am an AAW. (AS7)
  - Others have hinted that AAW are not assertive enough to be leaders. (AS8)
  - Others have hinted that AAW seem to have no desire for leadership. (AS9)
- Asian Fetishism (4 items)
  - Others express sexual interest in me because of my Asian appearance. (AF1)
  - Others take sexual interest in AAW to fulfill their fantasy. (AF2)
  - Others take romantic interest in AAW just because they never had sex with an AAW before. (AF3)
  - Others have treated me as if I am always open to sexual advances. (AF4)
- Media Invalidation (5 items)
  - I see non-Asian women being casted to play female Asian characters.(MI1)
  - I rarely see AAW playing the lead role in the media. (MI2)
  - I rarely see AAW in the media. (MI3)
  - I see AAW playing the same type of characters (e.g., Kung Fu woman, sidekick, mistress, tiger mom) in the media. (MI4)
  - I see AAW characters being portrayed as emotionally distant (e.g., cold-hearted, lack of empathy) in the media. (MI5)
- Assumptions of Universal Appearance (4 items)
  - Others have talked about AAW as if they all have the same facial features (e.g., eye shape, skin tone). (AUA1)
  - Others have suggested that all AAW look alike.(AUA2)
  - Others have talked about AAW as if they all have the same body type (e.g., petite, tiny, small-chested). (AUA3)
  - Others have pointed out physical traits in AAW that do not look ‘Asian’

Four additional scales were reported in the Keum et al. article [Keum et al., 2018]. Fortunately, I was able to find factor loadings from the original psychometric article or subsequent publications. For multidimensional scales, I assign variable names according to the scale to which the item belongs (e.g., Env42). In contrast, when subscales or short unidimensional scales were used, I assigned variable names based on item content (e.g., “blue”). In my own work, I prefer item-level names so that I can quickly see (without having to look up the item names) how the items are behaving. The scales, their original citation, and information about how I simulated data for each are listed below.

- **Racial Microaggressions Scale** (RMAS; [Torres-Harding et al., 2012]) is a 32-item scale with Likert scaling ranging from 0 (*never*) to 3 (*often/frequent*). Higher scores represent greater frequency of perceived microaggressions. I simulated data at the subscale level. The RMAS has six subscales, but only four (Invisibility, Low-Achieving/Undesirable Culture, Foreigner/Not Belonging, and Environmental Invalidation) were used in the study. Data were simulated using factor loadings (from the four factors) in the source article.
- **Schedule of Sexist Events** (SSE; [Klonoff and Landrine, 1995]) is a 20-item scale that with Likert scaling ranging from 1 (*the event has never happened to me*) to 6 (*the event happened almost all [i.e., more than 70%] of the time*). Higher scores represent greater frequency of everyday sexist events. I simulated data the subscale level. Within two larger scales (recent events, lifetime events), there are three subscales: Sexist Degradation and Its Consequences, Unfair/Sexist Events at Work/School, and Unfair Treatment in Distant and Close Relationships. Data were simulated using factor loadings from the source article.
- **PHQ-9** [Kroenke et al., 2001] is a 9-item scale with Likert scaling ranging from 0 (*not at all*) to 3 (*nearly every day*). Higher scores indicate higher levels of depression. I simulated data by estimating factor loadings from Brattmyr et al. [2022].
- **Internalized Racism in Asian American Scale** (IRAAS [Choi et al., 2017]) is a 14-item scale with Likert scaling ranging from 1 (*strongly disagree*) to 6 (*strongly agree*). Higher scores indicate greater internalized racism. Data were simulated using the factor loadings from the bifactor model in the source article.

As you consider homework options, there is sufficient simulated data to use the RMAS, SSE, or IRAAS.

Below, I walk through the data simulation. This is not an essential portion of the lesson, but I will lecture it in case you are interested. None of the items are negatively worded (relative to the other items), so there is no need to reverse-score any items.

Simulating the data involved using factor loadings, means, standard deviations, and correlations between the scales. Because the simulation will produce “out-of-bounds” values, the code below rescales the scores into the range of the Likert-type scaling and rounds them to whole values.

*#Entering the intercorrelations, means, and standard deviations from the journal article*

```
Keum_GRMS_generating_model <- '
  #measurement model
  General =~ .50*AS1 + .44*AS2 + .50*AS3 + .33*AS4 + .58*AS5 + .49*AS6 + .51*AS7 + .53*AS8
  AS =~ .68*AS1 + .65*AS2 + .53*AS3 + .55*AS4 + .54*AS5 + .55*AS6 + .42*AS7 + .47*AS8 + .40*AS9
  AF =~ .63*AF1 + .45*AF2 + .56*AF3 + .54*AF4
  AUA =~ .55*AUA1 + .55*AUA2 + .31*AUA3 + .31*AUA4
  MI =~ .27*MI1 + .53*MI2 + .57*MI3 + .29*MI4 + .09*MI5
  RMAS_FOR =~ .66*FOR1 + .90*FOR2 + .63*FOR4
  RMAS_LOW =~ .64*LOW22 + .54*LOW23 + .49*LOW28 + .63*LOW29 + .58*LOW30 + .67*LOW32 + .68*LOW33
  RMAS_INV =~ .66*INV33 + .70*INV39 + .79*INV40 + .71*INV41 + .71*INV47 + .61*INV49 + .62*INV50
  RMAS_ENV =~ .71*ENV42 + .70*ENV43 + .74*ENV44 + .57*ENV45 + .54*ENV46

  SSEL_Deg =~ .77*LDeg18 + .73*LDeg19 + .71*LDeg21 + .71*LDeg15 + .67*LDeg16 + .67*LDeg17
  SSEL_dRel =~ .69*LdRel4 + .68*LdRel6 + .64*LdRel7 + .64*LdRel5 + .63*LdRel11 + .49*LdRel12
  SSEL_cRel =~ .73*LcRel11 + .68*LcRel9 + .66*LcRel23
```

```

SSEL_Work =~ .73*LWork17 + .10*LWork10 + .64*LWork2

SSER_Deg =~ .72*RDeg15 + .71*RDeg21 + .69*RDeg18 + .68*RDeg16 + .68*RDeg13 + .65*RDeg1
SSER_dRel =~ .74*RDeg4 + .67*RDeg6 + .64*RDeg5 + .54*RDeg7 + .51*RDeg1
SSER_cRel =~ .69*RcRel9 + .59*RcRel11 + .53*RcRel23
SSER_Work =~ .72*RWork10 + .67*RWork2 + .62*RWork17 + .51*RWork3

SSE_Lifetime =~ SSEL_Deg + SSEL_dRel + SSEL_cRel + SSEL_Work
SSE_Recent =~ SSER_Deg + SSER_dRel + SSER_cRel + SSER_Work

PHQ9 =~ .798*anhedonia + .425*down + .591*sleep + .913*lo_energy + .441*appetite +
gIRAAS =~ .51*SN1 + .69*SN2 + .63*SN3 + .65*SN4 + .67*WS5 + .60*WS6 + .74*WS7 + .44*WS8
SelfNegativity =~ .60*SN1 + .50*SN2 + .63*SN3 + .43*SN4
WeakStereotypes =~ .38*WS5 + .22*WS6 + .10*WS7 + .77*WS8 + .34*WS9 + .14*WS10
AppearanceBias =~ .38*AB11 + .28*AB12 + .50*AB13 + .18*AB14

#Means
#Keum et al reported total scale scores, I divided those totals by the number of items
AS ~ 3.25*1
AF ~ 3.34*1
AUA ~ 4.52
MI ~ 5.77*1
General ~ 3.81*1
RMAS_FOR ~ 3.05*1
RMAS_LOW ~ 2.6*1
RMAS_INV ~ 2.105*1
RMAS_ENV ~ 3.126*1
SSEL_Deg ~ 2.55*1
SSEL_dRel ~ 1.96*1
SSEL_cRel ~ 3.10*1
SSEL_Work ~ 1.66*1
SSER_Deg ~ 2.02*1
SSER_dRel ~ 1.592*1
SSER_cRel ~ 1.777*1
SSER_Work ~ 1.3925*1
SSER_Lifetime ~ 2.8245*1
SSER_Recent ~ 2.4875*1
PHQ9 ~ 1.836*1
gIRAAS ~ 2.246*1

#Correlations
AS ~~ .00*AF
AS ~~ .00*AUA
AS ~~ .00*MI
AS ~~ .00*General

```

```

AS ~~ .28*RMAS_FOR
AS ~~ .24*RMAS_LOW
AS ~~ .46*RMAS_INV
AS ~~ .16*RMAS_ENV
AS ~~ .40*SSE_Lifetime
AS ~~ .28*SSE_Recent
AS ~~ .15*PHQ9
AS ~~ .13*gIRAAS

AF ~~ .00*AUA
AF ~~ .00*MI
AF ~~ .00*General
AF ~~ .02*RMAS_FOR
AF ~~ .05*RMAS_LOW
AF ~~ .11*RMAS_INV
AF ~~ .07*RMAS_ENV
AF ~~ .34*SSE_Lifetime
AF ~~ .27*SSE_Recent
AF ~~ -.04*PHQ9
AF ~~ .21*gIRAAS

AUA ~~ .00*MI
AUA ~~ .00*General
AUA ~~ .18*RMAS_FOR
AUA ~~ .20*RMAS_LOW
AUA ~~ .01*RMAS_INV
AUA ~~ -.04*RMAS_ENV
AUA ~~ .02*SSE_Lifetime
AUA ~~ .92*SSE_Recent
AUA ~~ .02*PHQ9
AUA ~~ .17*gIRAAS

MI ~~ .00*General
MI ~~ -.02*RMAS_FOR
MI ~~ .08*RMAS_LOW
MI ~~ .31*RMAS_INV
MI ~~ .36*RMAS_ENV
MI ~~ .15*SSE_Lifetime
MI ~~ .08*SSE_Recent
MI ~~ -.05*PHQ9
MI ~~ -.03*gIRAAS

General ~~ .34*RMAS_FOR
General ~~ .63*RMAS_LOW
General ~~ .44*RMAS_INV
General ~~ .45*RMAS_ENV

```

```

General ~~ .54*SSE_Lifetime
General ~~ .46*SSE_Recent
General ~~ .31*PHQ9
General ~~ -.06*gIRAAS

RMAS_FOR ~~ .57*RMAS_LOW
RMAS_FOR ~~ .56*RMAS_INV
RMAS_FOR ~~ .37*RMAS_ENV
RMAS_FOR ~~ .33*SSE_Lifetime
RMAS_FOR ~~ .25*SSE_Recent
RMAS_FOR ~~ .10*PHQ9
RMAS_FOR ~~ .02*gIRAAS

RMAS_LOW ~~ .69*RMAS_INV
RMAS_LOW ~~ .48*RMAS_ENV
RMAS_LOW ~~ .67*SSE_Lifetime
RMAS_LOW ~~ .57*SSE_Recent
RMAS_LOW ~~ .30*PHQ9
RMAS_LOW ~~ .16*gIRAAS

RMAS_INV ~~ .59*RMAS_ENV
RMAS_INV ~~ .63*SSE_Lifetime
RMAS_INV ~~ .52*SSE_Recent
RMAS_INV ~~ .32*PHQ9
RMAS_INV ~~ .23*gIRAAS

RMAS_ENV ~~ .46*SSE_Lifetime
RMAS_ENV ~~ .31*SSE_Recent
RMAS_ENV ~~ .11*PHQ9
RMAS_ENV ~~ .07*gIRAAS

SSE_Lifetime ~~ .83*SSE_Recent
SSE_Lifetime ~~ .30*PHQ9
SSE_Lifetime ~~ .14*gIRAAS

SSE_Recent ~~ .30*PHQ9
SSE_Recent ~~ .20*gIRAAS

PHQ9 ~~ .18*gIRAAS

#Correlations between SES scales from the Klonoff and Landrine article
#Note that in the article the factor orders were reversed
SSEL_Deg ~~ .64*SSEL_dRel
SSEL_Deg ~~ .61*SSEL_cRel
SSEL_Deg ~~ .50*SSEL_Work
SSEL_dRel ~~ .57*SSEL_cRel

```

```

SSEL_dRel ~~ .57*SSEL_Work
SSEL_cRel ~~ .47*SSEL_Work

SSER_Deg ~ .54*SSER_dRel
SSER_Deg ~ .54*SSER_Work
SSER_Deg ~ .59*SSER_cRel
SSER_dRel ~ .56*SSER_Work
SSER_dRel ~ .46*SSER_cRel
SSER_Work ~ .43*SSER_cRel

SSE_Lifetime ~ .75*SSE_Recent

'

set.seed(240311)
dfGRMSAAW <- lavaan::simulateData(model = Keum_GRMS_generating_model,
                                    model.type = "sem",
                                    meanstructure = T,
                                    sample.nobs=304,
                                    standardized=FALSE)

#used to retrieve column indices used in the rescaling script below
col_index <- as.data.frame(colnames(dfGRMSAAW))

#The code below loops through each column of the dataframe and assigns the scaling accordingly
#Rows 1 thru 22 are the GRMS items
#Rows 23 thru 47 are the RMAS
#Rows 48 thru 87 are the SSE
#Rows 88 thru 96 are the PHQ9
#Rows 97 thru 110 are the IRAAS
#Rows 111 thru 112 are scale scores for SSE

for(i in 1:ncol(dfGRMSAAW)){
  if(i >= 1 & i <= 22){
    dfGRMSAAW[,i] <- scales::rescale(dfGRMSAAW[,i], c(0, 5))
  }
  if(i >= 23 & i <= 47){
    dfGRMSAAW[,i] <- scales::rescale(dfGRMSAAW[,i], c(0, 3))
  }
  if(i >= 48 & i <= 87){
    dfGRMSAAW[,i] <- scales::rescale(dfGRMSAAW[,i], c(1, 6))
  }
  if(i >= 88 & i <= 96){
    dfGRMSAAW[,i] <- scales::rescale(dfGRMSAAW[,i], c(0, 3))
  }
  if(i >= 97 & i <= 110){
    dfGRMSAAW[,i] <- scales::rescale(dfGRMSAAW[,i], c(1, 6))
  }
}

```

```

    }
}

#rounding to integers so that the data resembles that which was collected
library(tidyverse)
dfGRMSAAW <- dfGRMSAAW %>% round(0)

#quick check of my work
#psych::describe(dfGRMSAAW)

```

The optional script below will let you save the simulated data to your computing environment as either an .rds object or a .csv file.

An .rds file preserves all formatting to variables prior to the export and re-import. For the purpose of this chapter, you don't need to do either. That is, you can re-simulate the data each time you work the problem.

```

# to save the df as an .rds (think 'R object') file on your computer;
# it should save in the same file as the .rmd file you are working
# with saveRDS(dfGRMSAAW, 'dfGRMSAAW.rds') bring back the simulated
# dat from an .rds file dfGRMSAAW <- readRDS('dfGRMSAAW.rds')

```

If you save the .csv file (think "Excel lite") and bring it back in, you will lose any formatting (e.g., ordered factors will be interpreted as character variables).

```

# write the simulated data as a .csv write.table(dfGRMSAAW,
# file='dfGRMSAAW.csv', sep=',', col.names=TRUE, row.names=FALSE)
# bring back the simulated dat from a .csv file dfGRMSAAW <- read.csv
# ('dfGRMSAAW.csv', header = TRUE)

```

## 11.6 A Quick *lavaan* Syntax Recap

- It's really just regression
  - tilda (~, *is regressed on*) is the regression operator
  - place DV (y) on left side of the regression operator
  - place IVs, separated by +, on the right of the regression operator
- f is a latent variable (LV)
- Example:  $y \sim f_1 + f_2 + x_1 + x_2$
- LVs must be *defined* by their manifest or latent indicators.
  - the special operator (=<~, *is measured/defined by*) is used for this
  - Example:  $f_1 =\sim y_1 + y_2 + y_3$

- Variances and covariances are specified with a double tilde operator ( $\sim\sim$ , *is correlated with*)
  - Example of variance:  $y1 \sim\sim y1$  (variable's relationship with itself)
  - Example of covariance:  $y1 \sim\sim y2$  (relationship with another variable)
  - Example of covariance of a factor:  $f1 \sim\sim f2$
- Intercepts ( $\sim 1$ ) for observed variables and LVs are simple, intercept-only regression formulas.
  - Example of variable intercept:  $y1 \sim 1$
  - Example of factor intercept:  $f1 \sim 1$
- A complete lavaan model is a combination of these formula types, enclosed between single quotation marks. Readability of model syntax is improved by:
  - splitting formulas over multiple lines
  - using blank lines within single quote
  - labeling with the hashtag

```
CFAmode1 <- '
f1 =~ y1 + y2 + y3
f2 =~ y4 + y5 + y6
f3 =~ y7 + y8 + y9 + y10
,'
```

Behind the scenes the *cfa()* function:

- fixes the factor loading of the first indicator of an LV to 1 (setting the scale)
- automatically adds residual variances (required)
- correlates all exogenous LVs; to turn these off add the following statement to the *cfa()* function statement: *orthogonal = TRUE*

## 11.7 Comparing and Tweaking Multidimensional First-Order Models

In the prior lesson we examined unidimensional and multidimensional variants of the GRMSAAW. Our work determined that the first-order structure that included four correlated factors was superior to a unidimensional measure. Starting with the multidimensional model (four factors), let's specify both correlated and uncorrelated options and compare them. We'll choose the best and see if we can further "tweak" it into acceptable fit.

## 11.8 An Uncorrelated Factors Model

### 11.8.1 Specifying the Model

In the absence of a more complex (e.g., second-order) structure, *lavaan's cfa()* function automatically correlates first-order factors. However, the more parsimonious model is one with uncorrelated

factors. We'll run it first. To do so, we need to turn off the default so that factors will be uncorrelated. This is accomplished in the *cfa()* function script with *orthogonal = TRUE*.

In the first step we specify the equations in our model.

```
grmsAAW4mod <- "AS =~ AS1 + AS2 + AS3 + AS4 + AS5 + AS6 + AS7 + AS8 + AS9
                  AF =~ AF1 + AF2 + AF3 + AF4
                  MI =~ MI1 + MI2 + MI3 + MI4 + MI5
                  AUA =~ AUA1 + AUA2 + AUA3 + AUA4"
grmsAAW4mod
```

```
[1] "AS =~ AS1 + AS2 + AS3 + AS4 + AS5 + AS6 + AS7 + AS8 + AS9\n" AF =~ AF1 + AF2 +
```

The next code will run the model. This is where we insert *orthogonal = TRUE*.

```
# next, use the cfa function to apply the model to the data
set.seed(240311)
uncorrF <- lavaan:::cfa(grmsAAW4mod, data = dfGRMSAAW, orthogonal = TRUE)
lavaan:::summary(uncorrF, fit.measures = TRUE, standardized = TRUE, rsquare = TRUE)
```

*lavaan* 0.6.17 ended normally after 44 iterations

Estimator	ML
Optimization method	NLMINB
Number of model parameters	44
Number of observations	304

Model Test User Model:

Test statistic	461.102
Degrees of freedom	209
P-value (Chi-square)	0.000

Model Test Baseline Model:

Test statistic	1439.317
Degrees of freedom	231
P-value	0.000

User Model versus Baseline Model:

Comparative Fit Index (CFI)	0.791
Tucker-Lewis Index (TLI)	0.769

Loglikelihood and Information Criteria:

Loglikelihood user model (H0)	-8395.340
Loglikelihood unrestricted model (H1)	-8164.789

Akaike (AIC)	16878.679
Bayesian (BIC)	17042.229
Sample-size adjusted Bayesian (SABIC)	16902.683

Root Mean Square Error of Approximation:

RMSEA	0.063
90 Percent confidence interval - lower	0.055
90 Percent confidence interval - upper	0.071
P-value H_0: RMSEA <= 0.050	0.003
P-value H_0: RMSEA >= 0.080	0.000

Standardized Root Mean Square Residual:

SRMR	0.151
------	-------

Parameter Estimates:

Standard errors	Standard
Information	Expected
Information saturated (h1) model	Structured

Latent Variables:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
AS =~						
AS1	1.000				0.552	0.603
AS2	1.169	0.139	8.425	0.000	0.646	0.639
AS3	0.931	0.124	7.534	0.000	0.515	0.548
AS4	0.904	0.121	7.451	0.000	0.499	0.540
AS5	1.138	0.135	8.453	0.000	0.629	0.642
AS6	0.653	0.094	6.935	0.000	0.361	0.493
AS7	0.800	0.118	6.785	0.000	0.442	0.479
AS8	0.899	0.121	7.443	0.000	0.496	0.539
AS9	0.770	0.106	7.291	0.000	0.426	0.525
AF =~						
AF1	1.000				0.591	0.659
AF2	0.881	0.151	5.851	0.000	0.521	0.536
AF3	0.665	0.126	5.287	0.000	0.393	0.445
AF4	0.943	0.159	5.950	0.000	0.557	0.566
MI =~						
MI1	1.000				0.426	0.511
MI2	1.136	0.209	5.447	0.000	0.484	0.549
MI3	1.475	0.264	5.589	0.000	0.629	0.614
MI4	1.089	0.212	5.139	0.000	0.464	0.483
MI5	0.614	0.166	3.703	0.000	0.262	0.297

AUA =~

AUA1	1.000			0.590	0.579
AUA2	0.950	0.148	6.401	0.000	0.560
AUA3	0.745	0.124	6.027	0.000	0.439
AUA4	0.928	0.149	6.243	0.000	0.547

Covariances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
AS ~~						
AF	0.000				0.000	0.000
MI	0.000				0.000	0.000
AUA	0.000				0.000	0.000
AF ~~						
MI	0.000				0.000	0.000
AUA	0.000				0.000	0.000
MI ~~						
AUA	0.000				0.000	0.000

Variances:

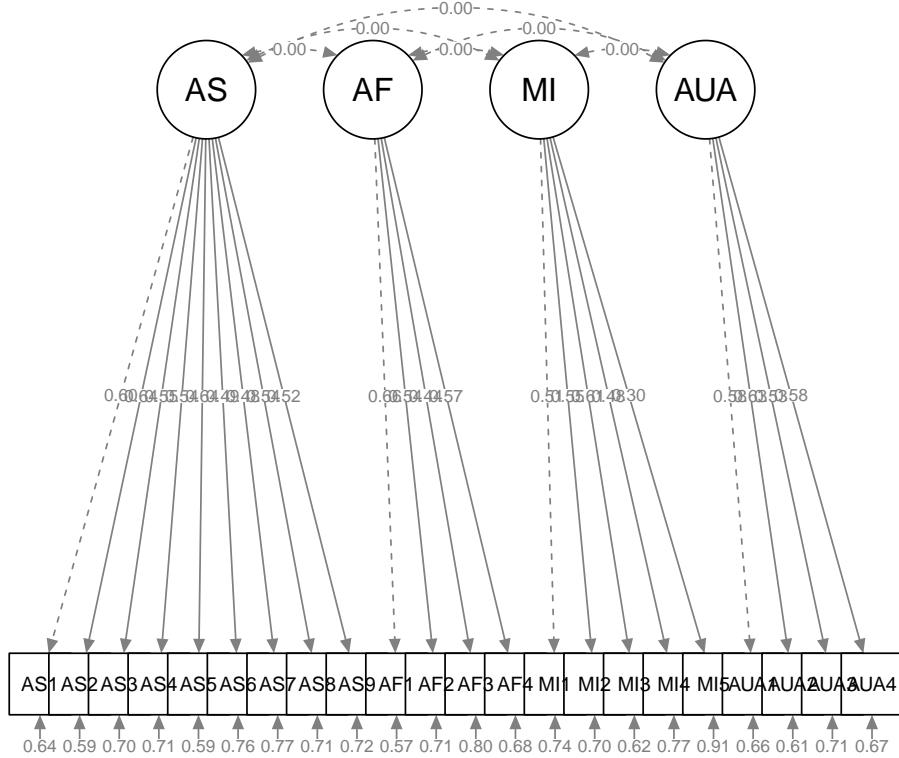
	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
.AS1	0.535	0.050	10.643	0.000	0.535	0.637
.AS2	0.603	0.059	10.283	0.000	0.603	0.591
.AS3	0.618	0.056	11.065	0.000	0.618	0.700
.AS4	0.607	0.055	11.117	0.000	0.607	0.709
.AS5	0.563	0.055	10.248	0.000	0.563	0.587
.AS6	0.407	0.036	11.386	0.000	0.407	0.757
.AS7	0.654	0.057	11.451	0.000	0.654	0.770
.AS8	0.602	0.054	11.122	0.000	0.602	0.710
.AS9	0.477	0.043	11.209	0.000	0.477	0.725
.AF1	0.454	0.065	6.992	0.000	0.454	0.565
.AF2	0.672	0.071	9.516	0.000	0.672	0.713
.AF3	0.627	0.059	10.659	0.000	0.627	0.802
.AF4	0.658	0.073	8.991	0.000	0.658	0.679
.MI1	0.516	0.053	9.801	0.000	0.516	0.739
.MI2	0.543	0.059	9.205	0.000	0.543	0.698
.MI3	0.652	0.082	7.959	0.000	0.652	0.622
.MI4	0.710	0.070	10.164	0.000	0.710	0.767
.MI5	0.706	0.061	11.654	0.000	0.706	0.912
.AUA1	0.689	0.075	9.140	0.000	0.689	0.664
.AUA2	0.482	0.059	8.205	0.000	0.482	0.606
.AUA3	0.483	0.049	9.827	0.000	0.483	0.714
.AUA4	0.605	0.066	9.201	0.000	0.605	0.669
AS	0.305	0.058	5.229	0.000	1.000	1.000
AF	0.349	0.076	4.605	0.000	1.000	1.000
MI	0.182	0.050	3.659	0.000	1.000	1.000
AUA	0.348	0.081	4.312	0.000	1.000	1.000

R-Square:

	Estimate
AS1	0.363
AS2	0.409
AS3	0.300
AS4	0.291
AS5	0.413
AS6	0.243
AS7	0.230
AS8	0.290
AS9	0.275
AF1	0.435
AF2	0.287
AF3	0.198
AF4	0.321
MI1	0.261
MI2	0.302
MI3	0.378
MI4	0.233
MI5	0.088
AUA1	0.336
AUA2	0.394
AUA3	0.286
AUA4	0.331

Producing a figure can be useful to represent what we did to others as well as checking our own work. That is, “Did we think we did what we intended?” When the *what* = “*col*”, *whatLabels* = “*stand*” combination is shown, paths that are “fixed” are represented by dashed lines. Below, we expect to see each of the four factors predicting only the items associated with their factor. One item for each factor (the first on the left) should be specified as the indicator variable (and represented with a dashed line). Additionally, the factors/latent variables should not be freed to covary (i.e., an uncorrelated traits or orthogonal model). Because they are “fixed” to be 0.00, they will be represented with dashed (not solid) curves with double-headed arrows.

```
semPlot::semPaths(uncorrF, layout = "tree", style = "lisrel", what = "col",
  whatLabels = "stand")
```



Among my first steps are also to write the code to export the results. The *tidySEM* package has useful functions to export the fit statistics, parameter estimates, and correlations among the latent variables (i.e., factors).

```
UncorrFitStats <- tidySEM::table_fit(uncorrF)
```

```
Registered S3 method overwritten by 'tidySEM':
  method      from
  predict.MxModel OpenMx
```

```
Uncorr_paramEsts <- tidySEM::table_results(uncorrF, digits=3, columns = NULL)
UncorrCorrs <- tidySEM::table_corrs(uncorrF, digits=3)
#to see each of the tables, remove the hashtag
#Uncorr_FitStats
#Uncorr_paramEsts
#UncorrCorrs
```

Next, I export them.

```
write.csv(UncorrFitStats, file = "UncorrFitStats.csv")
write.csv(Uncorr_paramEsts, file = "Uncorr_paramEsts.csv")
write.csv(UncorrCorrs, file = "UncorrCorrs.csv")
```

### 11.8.2 Interpreting the Output

Criteria	Our Results	Criteria met?
Factor loadings significant, strong, proper valence	AS: .59 to .80; AF: .64 to .82; MI: .35 to .62; AUA: .49 to .82	Yes
Non-significant chi-square	$\chi^2(209) = 461.102, p < 0.001$	No
$CFI \geq .90$ or $.95$	CFI = 0.791	No
$RMSEA \leq .05$ (but definitely < .10)	RMSEA = 0.063, 90%CI(0.055, 0.071)	Yes (with caution)
$SRMR \leq .08$ (but definitely < .10)	SRMS = 0.151	No
Combination rule: $CFI \geq .95$ and $SRMR \leq .08$	CFI = 0.791, SRS = 0.151	No

### 11.8.3 Partial Write-up

**Uncorrelated factors model.** Our model where factors were fixed to remain uncorrelated demonstrated less than adequate fit to the data:  $\chi^2(209) = 461.102, p < 0.001$ , CFI = 0.791, RMSEA = 0.063, 90%CI(0.055, 0.071), SRMR = 0.151. Factor loadings ranged from .59 to .80 for the AS scale, .64 to .82 for the AF scale, .35 to .62 for the MI scale, and .49 to .82 for the fear of AUA scale.

Our fit is not satisfactory. We can expect the correlated factors model should have a better fit. Instead of “tweaking” this one, let’s move onto the correlated factors model.

## 11.9 A Correlated Factors Model

Let’s revisit the statement I just made: *...the correlated factors model should have a better fit.* Why did I make this statement? It’s all about degrees of freedom and whether the model is the nested or nesting model.

### 11.9.1 Nested Models

When we specify (i.e., draw) models in SEM/CFA, we often think that the paths (single headed arrows/paths, double-headed arrows/covariances) between the parameters are our hypotheses. They are, but they are *soft hypotheses* in that we are *freeing* the elements to covary. The *hard hypotheses* (i.e., no paths, no covariances) states that the parameters are unrelated. We are trying to explain

the covariance matrix (where all parameters are freed to covary) with the fewest paths possible. That is, we are freeing the relations between our hypothesized parameters and restricting all others to be zero.

Two models are **nested** (aka **hierarchical**) if one is a proper subset of the other. The **nesting** model is the one with the most parameters freed to covary. That is, it has more paths/covariances drawn on it. Almost always, the **nesting model** (i.e., most sticks, fewer degrees of freedom) will have better fit than the **nested** model (i.e., fewer sticks, more degrees of freedom).

In our example, *uncorrF* has four uncorrelated factors and its degrees of freedom was 209.

Our new model will *add covariances* (making it the *nesting model* with presumed better fit) to all possible combinations of the four factors (we end up with 6 covariance paths). Freeing these additional factors to covary in the *corrF* model (recall they were fixed to 0.0 in the *uncorrF* model) leads to a model with 203 degrees of freedom. The degrees of freedom are lower because the algorithm now needs to estimate 6 additional covariances/parameters (i.e.,  $209 - 6 = 203$ ).

Model fit (generally) improves when paths/covariances are added (and degrees of freedom decreases). The model with the *most paths* (I think of “sticks” in a nest) and the *fewest df* is the *nesting model* and it (almost) always has superior fit.

Let's try. We continue to use the model of equations we specified for the orthogonal, uncorrelated traits, model.

```
# in our 4-factor models we can use the same baseM, the difference
# here is that we deleted 'orthogonal = TRUE' uncorrF <-
# lavaan:::cfa(grmsAAW4mod, data = dfGRMSAAW, orthogonal = TRUE) #for
# comparison, this was the uncorrelated model
set.seed(240311)
corrF <- lavaan:::cfa(grmsAAW4mod, data = dfGRMSAAW)
lavaan:::summary(corrF, fit.measures = TRUE, standardized = TRUE, rsquare = TRUE)
```

*lavaan* 0.6.17 ended normally after 42 iterations

Estimator	ML
Optimization method	NLMINB
Number of model parameters	50
Number of observations	304

Model Test User Model:

Test statistic	232.453
Degrees of freedom	203
P-value (Chi-square)	0.076

Model Test Baseline Model:

Test statistic	1439.317
Degrees of freedom	231

P-value	0.000
---------	-------

User Model versus Baseline Model:

Comparative Fit Index (CFI)	0.976
Tucker-Lewis Index (TLI)	0.972

Loglikelihood and Information Criteria:

Loglikelihood user model (H0)	-8281.015
Loglikelihood unrestricted model (H1)	-8164.789
Akaike (AIC)	16662.030
Bayesian (BIC)	16847.882
Sample-size adjusted Bayesian (SABIC)	16689.307

Root Mean Square Error of Approximation:

RMSEA	0.022
90 Percent confidence interval - lower	0.000
90 Percent confidence interval - upper	0.034
P-value H_0: RMSEA <= 0.050	1.000
P-value H_0: RMSEA >= 0.080	0.000

Standardized Root Mean Square Residual:

SRMR	0.047
------	-------

Parameter Estimates:

Standard errors	Standard
Information	Expected
Information saturated (h1) model	Structured

Latent Variables:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
AS =~						
AS1	1.000				0.550	0.600
AS2	1.132	0.136	8.330	0.000	0.623	0.617
AS3	0.958	0.123	7.769	0.000	0.527	0.561
AS4	0.901	0.120	7.504	0.000	0.496	0.536
AS5	1.152	0.134	8.620	0.000	0.634	0.647
AS6	0.669	0.094	7.133	0.000	0.368	0.503
AS7	0.829	0.118	7.043	0.000	0.456	0.495
AS8	0.905	0.120	7.551	0.000	0.498	0.540
AS9	0.757	0.104	7.256	0.000	0.417	0.514
AF =~						
AF1	1.000				0.505	0.563

AF2	1.195	0.174	6.862	0.000	0.603	0.621
AF3	0.738	0.137	5.395	0.000	0.373	0.422
AF4	1.138	0.171	6.665	0.000	0.575	0.584
MI ==						
MI1	1.000				0.482	0.577
MI2	0.917	0.148	6.216	0.000	0.442	0.501
MI3	1.169	0.177	6.602	0.000	0.563	0.550
MI4	0.921	0.157	5.865	0.000	0.444	0.461
MI5	0.688	0.137	5.018	0.000	0.332	0.377
AUA ==						
AUA1	1.000				0.553	0.543
AUA2	0.981	0.140	7.016	0.000	0.543	0.608
AUA3	0.785	0.122	6.457	0.000	0.434	0.528
AUA4	1.083	0.152	7.140	0.000	0.599	0.630

## Covariances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
AS ~~						
AF	0.148	0.030	4.951	0.000	0.533	0.533
MI	0.136	0.028	4.889	0.000	0.513	0.513
AUA	0.181	0.034	5.257	0.000	0.595	0.595
AF ~~						
MI	0.154	0.031	5.010	0.000	0.632	0.632
AUA	0.164	0.034	4.805	0.000	0.588	0.588
MI ~~						
AUA	0.189	0.036	5.303	0.000	0.709	0.709

## Variances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
.AS1	0.538	0.050	10.833	0.000	0.538	0.640
.AS2	0.632	0.059	10.699	0.000	0.632	0.620
.AS3	0.605	0.054	11.111	0.000	0.605	0.685
.AS4	0.610	0.054	11.260	0.000	0.610	0.713
.AS5	0.557	0.053	10.408	0.000	0.557	0.581
.AS6	0.401	0.035	11.433	0.000	0.401	0.747
.AS7	0.641	0.056	11.470	0.000	0.641	0.755
.AS8	0.601	0.053	11.235	0.000	0.601	0.708
.AS9	0.484	0.043	11.379	0.000	0.484	0.736
.AF1	0.548	0.055	9.928	0.000	0.548	0.683
.AF2	0.579	0.064	9.062	0.000	0.579	0.614
.AF3	0.642	0.057	11.230	0.000	0.642	0.822
.AF4	0.638	0.066	9.651	0.000	0.638	0.659
.MI1	0.465	0.047	9.823	0.000	0.465	0.667
.MI2	0.582	0.055	10.664	0.000	0.582	0.749
.MI3	0.731	0.072	10.158	0.000	0.731	0.697
.MI4	0.729	0.066	10.994	0.000	0.729	0.787
.MI5	0.665	0.058	11.519	0.000	0.665	0.858
.AUA1	0.730	0.069	10.535	0.000	0.730	0.705

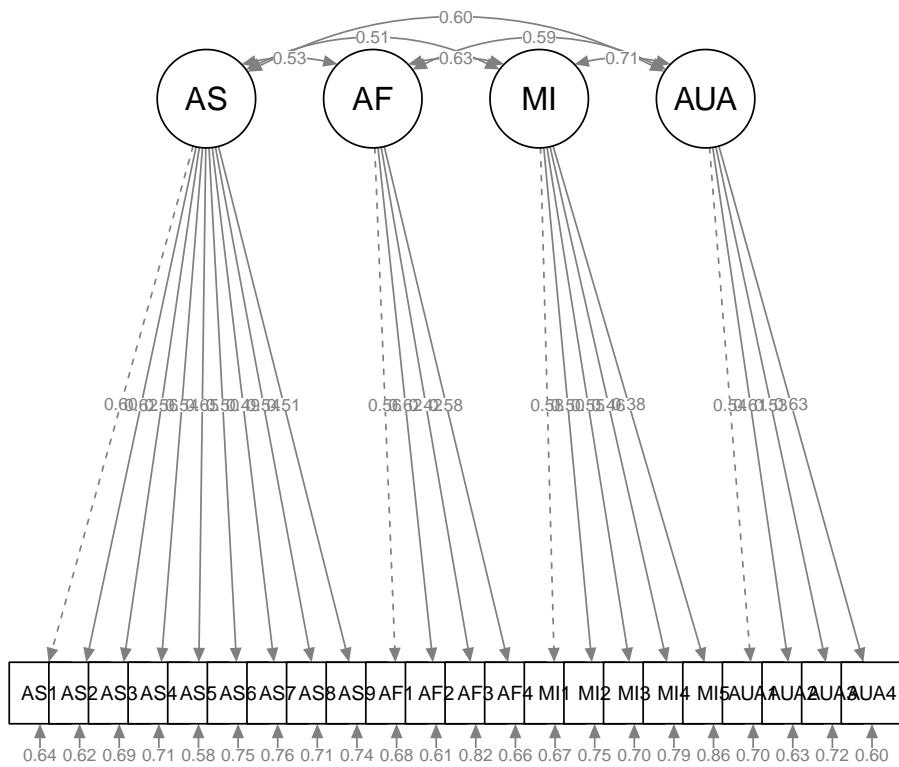
.AUA2	0.501	0.051	9.787	0.000	0.501	0.630
.AUA3	0.487	0.046	10.675	0.000	0.487	0.721
.AUA4	0.546	0.058	9.475	0.000	0.546	0.603
AS	0.303	0.058	5.264	0.000	1.000	1.000
AF	0.255	0.058	4.412	0.000	1.000	1.000
MI	0.232	0.051	4.559	0.000	1.000	1.000
AUA	0.306	0.070	4.391	0.000	1.000	1.000

R-Square:

	Estimate
AS1	0.360
AS2	0.380
AS3	0.315
AS4	0.287
AS5	0.419
AS6	0.253
AS7	0.245
AS8	0.292
AS9	0.264
AF1	0.317
AF2	0.386
AF3	0.178
AF4	0.341
MI1	0.333
MI2	0.251
MI3	0.303
MI4	0.213
MI5	0.142
AUA1	0.295
AUA2	0.370
AUA3	0.279
AUA4	0.397

As we plot this model we expect to see each of the four factors predicting only the items associated with their factor, one item for each factor (the first on the left) specified as the indicator variable, and double-headed arrows between the factors/latent variables, indicating that they are free to covary (i.e., a correlated traits model).

```
semPlot::semPaths(corrF, layout = "tree", style = "lisrel", what = "col",
whatLabels = "stand")
```



Among my first steps are also to write the code to export the results. The *tidySEM* package has useful functions to export the fit statistics, parameter estimates, and correlations among the latent variables (i.e., factors).

```
CorrFitStats <- tidySEM::table_fit(corrF)
Corr_paramEsts <- tidySEM::table_results(corrF, digits=3, columns = NULL)
CorrCorrs <- tidySEM::table_cors(corrF, digits=3)
#to see each of the tables, remove the hashtag
#CorrFitStats
#Corr_paramEsts
#CorrCorrs
```

Next, I export them.

```
write.csv(CorrFitStats, file = "CorrFitStats.csv")
write.csv(Corr_paramEsts, file = "Corr_paramEsts.csv")
write.csv(CorrCorrs, file = "CorrCorrs.csv")
```

### 11.9.2 Interpreting the Output

Criteria	Our Results	Criteria met?
Factor loadings significant, strong, proper valence	AS: .50 to .65; AF: .42 to .62; MI: .46 to .58; AUA: .54 to .63	Yes
Non-significant chi-square	$\chi^2(203) = 232.453, p = 0.076$	Yes
$CFI \geq .95$	$CFI = 0.976$	Yes
$RMSEA \leq .05$ (but definitely < .10)	$RMSEA = 0.022, 90\%CI(0.000, 0.034)$	Yes
$SRMR \leq .08$ (but definitely < .10)	$SRMR = 0.047$	Yes
Combination rule: $CFI \geq .95$ and $SRMR \leq .08$	$CFI = 0.976, SRS = 0.047$	Yes

### 11.9.3 Partial Write-up

**Correlated factors model.** We evaluated a single-order, correlated factors model where each of the 22 items loaded onto one of four factors and the factors were free to correlate. Standardized pattern coefficients ranged between .37 and .60 on the AF factor, between .37 and .63 on the AS factor, between .33 and .56 on the MI factor, and between .43 and .60 on the AUA factor. The Chi-square index was not statistically significant ( $\chi^2(203) = 232.453, p = 0.076$ ) indicating reasonable fit. The CFI value of 0.972 exceeded the recommendation of .95. The RMSEA = 0.022 (90%CI[.000, 0.034]) was satisfactory. The SRMR value of 0.047 remained below the warning criteria of .10. The AIC and BIC values were 16662.030 and 16847.882, respectively.

Recall that we can formally compare these models with the  $\chi^2_D$ , AIC, and BIC.

```
lavaan:::lavTestLRT(uncorrF, corrF)
```

Chi-Squared Difference Test

	Df	AIC	BIC	Chisq	Chisq diff	RMSEA	Df diff	Pr(>Chisq)			
corrF	203	16662	16848	232.45							
uncorrF	209	16879	17042	461.10	228.65	0.34938	6	< 0.0000000000000022			
corrF											
uncorrF	***										
---											
Signif. codes:	0	'***'	0.001	'**'	0.01	'*'	0.05	'. '	0.1	' '	1

The AIC and BIC are flexible to compare nested and non-nested models. Models with the lower values are superior. Consistent with what we expect (i.e., the nesting model [the model with the

most parameters and fewest degrees of freedom] should be superior), the AIC and BIC favor the correlated factors model.

The  $\chi^2_D$  can only be used for nested models (where items/indicators are identical – the only difference is the presence/absence of parameters). If it is statistically significant, the better model is the one with the lower chi-square value (and better fit indices). In this particular comparison there is a statistically significant difference favoring the correlated factors model ( $\chi^2(6) = 228.65, p < 0.001$ ).

To recap the highlights of nesting, the *nesting* model will usually have the best fit. The *nesting* model has:

- the most free parameters
  - (the most “sticks”)
- the fewest degrees of freedom

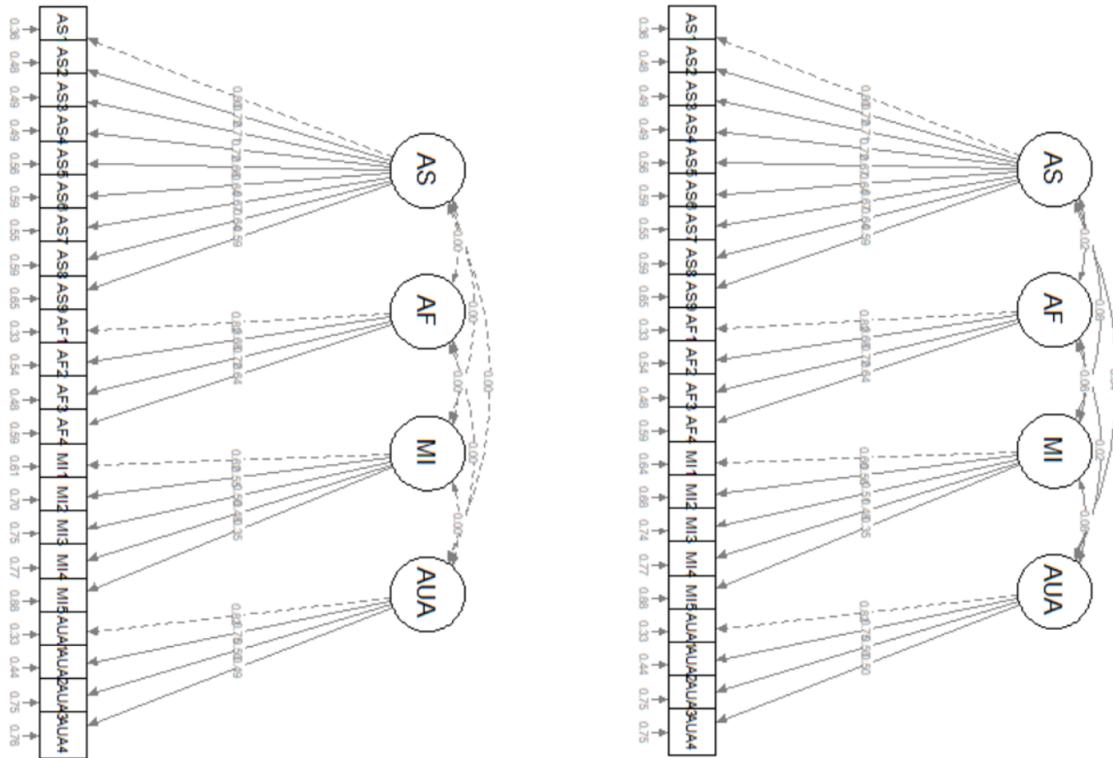


Figure 11.3: Side by side comparison of uncorrelated and correlated models

Examining the two models we compared side-by-side (uncorrelated on left; correlated on right), we note the additional “sticks” (i.e., the covariances that were freed) in the correlated factors model (i.e., the nesting model) and can guess (without analyzing the data) that it because it has (a) more sticks and (b) fewer degrees of freedom, it will have (c) better fit.

How to keep them straight: “the nested is within (or ‘sits in’ or ‘fits in’) the nesting model.” I also think, “The nested sits within but the nesting will win.” Just keep saying these until they “stick” (bad pun intended).

## 11.10 Model Respecification

Our correlated factors model has excellent fit, but this is not always the case. One way to improve model fit is to add parameters to simpler models – this is called **model building**. This can only occur for models that are **overidentified** (i.e., they have positive degrees of freedom).

In the CFA/psychometric case, a just-identified model is one that has at least 3 items per scale for a unidimensional factor structure and at least 2 items per scale in a multidimensional factor structure.

As we “free” each parameter (i.e., add paths or covariances), we correspondingly decrease the df. So, we must be diligent when engaging in model building.

In the CFA/psychometric case, *freeing parameters* usually means one of two things.

- Allowing cross-loadings.
  - This would mean that an item belongs to two factors/scales. While this might be theoretically defensible, items that belong to more than one scale cause scoring difficulties when the scale is put into practice.
- Allowing the error variances of indicators to correlate.
  - This would mean that there is something in common about the two items that is not explained/caused by the items’ relationship(s) with their respective factor(s). There are a variety of reasons this could occur, perhaps they have a content element that is in common, but different than the factor to which they belong. Methods factors (e.g., reverse scored items) can also contribute to items being correlated.

We use **modification indices** as a guide to determine if an error covariance is worth freeing. Modification indices tell you the degree to which your chi-square value will drop if the relationship between the two parameters is freed to relate (either a path or a covariance). Generally, a 1 degree of freedom change in a model will be a statistically significant difference if the chi-square value drops by 4 points. This is purely a statistical test that you have to then discern:

- if allowing the two elements to relate is theoretically defensible; and/or
- if there is truly something reasonably in common between the elements that is different from the theorized relations with the factors

Although many psychometricians frown on this, I think it, minimally, makes good diagnostic sense to take a look. The code below extracts the modification indices (MIs) from the object (*corrF*) that holds the *lavaan* output. Only MIs with a value greater than 4.0 are shown and they are sorted in descending order. We only ask for MIs greater than 4.0 because a 1 degree-of-freedom Chi-square difference test requires a difference of 3.841 (rounds to 4.0) to be statistically significant at  $p < 0.05$ .

```
lavaan:::modindices(corrF, sort = TRUE, minimum.value = 4)
```

	lhs	op	rhs	mi	epc	sepc.lv	sepc.all	sepc.nox
282	AF1	~~	AUA1	8.812	-0.124	-0.124	-0.196	-0.196
324	MI2	~~	MI3	8.166	0.135	0.135	0.206	0.206
275	AF1	~~	AF3	7.785	0.115	0.115	0.195	0.195
69	AF	=~	AS2	7.738	-0.435	-0.220	-0.218	-0.218
77	AF	=~	MI1	7.221	0.517	0.261	0.313	0.313
120	AUA	=~	MI5	6.740	0.537	0.297	0.337	0.337
95	MI	=~	AF1	6.221	-0.539	-0.260	-0.290	-0.290
241	AS7	~~	MI4	5.912	0.102	0.102	0.150	0.150
319	MI1	~~	MI5	5.849	-0.092	-0.092	-0.166	-0.166
288	AF2	~~	MI1	5.755	0.088	0.088	0.169	0.169
78	AF	=~	MI2	5.748	-0.473	-0.239	-0.271	-0.271
193	AS4	~~	MI4	5.479	-0.097	-0.097	-0.145	-0.145
289	AF2	~~	MI2	5.191	-0.090	-0.090	-0.155	-0.155
163	AS3	~~	AS5	5.059	-0.089	-0.089	-0.153	-0.153
124	AS1	~~	AS5	5.050	0.085	0.085	0.156	0.156
168	AS3	~~	AF1	4.910	0.082	0.082	0.143	0.143
112	AUA	=~	AF1	4.850	-0.372	-0.206	-0.230	-0.230
113	AUA	=~	AF2	4.735	0.416	0.230	0.237	0.237
278	AF1	~~	MI2	4.601	-0.080	-0.080	-0.141	-0.141
63	AS	=~	MI5	4.516	0.277	0.152	0.173	0.173
64	AS	=~	AUA1	4.492	-0.349	-0.192	-0.188	-0.188
67	AS	=~	AUA4	4.463	0.334	0.184	0.193	0.193
259	AS8	~~	AUA3	4.441	-0.072	-0.072	-0.133	-0.133
182	AS4	~~	AS6	4.380	-0.065	-0.065	-0.132	-0.132
92	MI	=~	AS7	4.127	0.312	0.151	0.163	0.163
96	MI	=~	AF2	4.072	0.494	0.238	0.245	0.245

### 11.10.1 Respecifying with Correlated Errors

When we inspect the modification indices output, we are:

- inspecting (and perhaps acting on) at the highest *mi* value, one at a time,
- seeing if that value seems a substantially higher than the next highest value.

In CFA models, freeing the errors of the items to covary means that there is something in common between the items that is not explained by their relationship to the factor (or, factors, if they are assigned to different factors). It is important to consider (theoretically, rationally) what might be shared between the items. It could be content; it could be a methods factor (e.g., reverse-scored items).

In our dataset, allowing the AF1 error to correlate with the AUA1 error will reduce the  $\chi^2$  by 8.812 points. Generally, a 1 degree of freedom change in a model will be a statistically significant difference if the chi-square value drops by 4 points, so we can expect this to make a statistically significant difference.

Next, we must inspect the relationship to see if we could justify connecting them through a path or covariance. The items in question are:

- Others express sexual interest in me because of my Asian appearance. (AF1)
- Others have talked about AAW as if they all have the same facial features (e.g., eye shape, skin tone). (AUA1)

The MI value greater than 4.0 suggests that there *may* be something in common between these two items that is not explained by their relationship with their respective factors (which are allowed to correlate). To allow these variables to covary we simply update the object which holds our map/model of equations and specify a covariance between the variables (“AF1 ~~ AUA1”) =.” What’s actually happening is that we are allowing the *errors* of these variables to correlate, hence this is frequently referred to as “correlated errors” or “error covariances.”

Here’s where the research team has the opportunity to reexamine these variables, their hypothesized relationship with the factor, and consider alternatives. In the specific case of these variables, the two factors are Asian Fetishism (AF) and Assumption of Universal Appearance (AUA). One of my first thoughts is that, like the items on the AUA factor, AF1 references appearance or physical features. Thus, I can see why the items would be correlated. I do note that an MI of 8 is relatively low. The typical approach is to look at the sorted MIs and start with the “outrageously high” ones. Once the MIs even out, we stop respecifying.

Were this my data, I would be happy with these results and not respecify the model. However, because this is a teaching lesson, I will demonstrate the respecification and evaluation.

```
ModInd_M1 <- "AS =~ AS1 + AS2 + AS3 + AS4 + AS5 + AS6 + AS7 + AS8 + AS9
                AF =~ AF1 + AF2 + AF3 + AF4
                MI =~ MI1 + MI2 + MI3 + MI4 + MI5
                AUA =~ AUA1 + AUA2 + AUA3 + AUA4

                #adding the correlated errors
                AF1 ~~ AUA1
"

```

We’ll give our respecified model a new object name and run it. Because we have added a path (allowing the cross-loading), this becomes the nesting model (it has the most paths and the fewest degrees of freedom).

```
set.seed(240311)
ModInd_M1f <- lavaan::cfa(ModInd_M1, data = dfGRMSAAW)
lavaan::summary(ModInd_M1f, fit.measures = TRUE, standardized = TRUE)
```

`lavaan 0.6.17 ended normally after 42 iterations`

Estimator	ML
Optimization method	NLMINB
Number of model parameters	51
Number of observations	304

Model Test User Model:

Test statistic	223.332
Degrees of freedom	202
P-value (Chi-square)	0.145

Model Test Baseline Model:

Test statistic	1439.317
Degrees of freedom	231
P-value	0.000

User Model versus Baseline Model:

Comparative Fit Index (CFI)	0.982
Tucker-Lewis Index (TLI)	0.980

Loglikelihood and Information Criteria:

Loglikelihood user model (H0)	-8276.455
Loglikelihood unrestricted model (H1)	-8164.789
Akaike (AIC)	16654.910
Bayesian (BIC)	16844.478
Sample-size adjusted Bayesian (SABIC)	16682.732

Root Mean Square Error of Approximation:

RMSEA	0.019
90 Percent confidence interval - lower	0.000
90 Percent confidence interval - upper	0.032
P-value H_0: RMSEA <= 0.050	1.000
P-value H_0: RMSEA >= 0.080	0.000

Standardized Root Mean Square Residual:

SRMR	0.046
------	-------

Parameter Estimates:

Standard errors	Standard
Information	Expected
Information saturated (h1) model	Structured

Latent Variables:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
AS =~						
AS1	1.000				0.550	0.600
AS2	1.133	0.136	8.327	0.000	0.623	0.617

AS3	0.959	0.123	7.771	0.000	0.528	0.561
AS4	0.901	0.120	7.493	0.000	0.495	0.535
AS5	1.152	0.134	8.611	0.000	0.634	0.647
AS6	0.670	0.094	7.132	0.000	0.368	0.503
AS7	0.829	0.118	7.038	0.000	0.456	0.495
AS8	0.906	0.120	7.553	0.000	0.498	0.541
AS9	0.757	0.104	7.248	0.000	0.416	0.513
AF ==~						
AF1	1.000				0.508	0.568
AF2	1.183	0.171	6.915	0.000	0.601	0.619
AF3	0.740	0.135	5.469	0.000	0.376	0.425
AF4	1.118	0.167	6.686	0.000	0.568	0.577
MI ==~						
MI1	1.000				0.483	0.578
MI2	0.915	0.147	6.217	0.000	0.442	0.501
MI3	1.170	0.177	6.617	0.000	0.565	0.552
MI4	0.918	0.157	5.860	0.000	0.443	0.460
MI5	0.683	0.137	5.000	0.000	0.330	0.375
AUA ==~						
AUA1	1.000				0.551	0.543
AUA2	0.982	0.140	7.023	0.000	0.541	0.606
AUA3	0.785	0.121	6.459	0.000	0.432	0.526
AUA4	1.087	0.152	7.161	0.000	0.599	0.630

## Covariances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
.AF1 ==~						
.AUA1	-0.126	0.042	-2.996	0.003	-0.126	-0.200
AS ==~						
AF	0.147	0.030	4.933	0.000	0.526	0.526
MI	0.136	0.028	4.891	0.000	0.513	0.513
AUA	0.180	0.034	5.251	0.000	0.593	0.593
AF ==~						
MI	0.156	0.031	5.047	0.000	0.635	0.635
AUA	0.173	0.033	5.217	0.000	0.618	0.618
MI ==~						
AUA	0.187	0.035	5.295	0.000	0.704	0.704

## Variances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
.AS1	0.538	0.050	10.832	0.000	0.538	0.640
.AS2	0.632	0.059	10.693	0.000	0.632	0.620
.AS3	0.605	0.054	11.105	0.000	0.605	0.685
.AS4	0.611	0.054	11.261	0.000	0.611	0.713
.AS5	0.557	0.054	10.408	0.000	0.557	0.581
.AS6	0.401	0.035	11.430	0.000	0.401	0.747
.AS7	0.642	0.056	11.469	0.000	0.642	0.755
.AS8	0.600	0.053	11.229	0.000	0.600	0.707

.AS9	0.485	0.043	11.379	0.000	0.485	0.737
.AF1	0.543	0.055	9.879	0.000	0.543	0.678
.AF2	0.582	0.063	9.180	0.000	0.582	0.617
.AF3	0.640	0.057	11.237	0.000	0.640	0.819
.AF4	0.646	0.066	9.808	0.000	0.646	0.667
.MI1	0.464	0.047	9.807	0.000	0.464	0.666
.MI2	0.582	0.055	10.667	0.000	0.582	0.749
.MI3	0.729	0.072	10.140	0.000	0.729	0.696
.MI4	0.730	0.066	11.000	0.000	0.730	0.788
.MI5	0.666	0.058	11.528	0.000	0.666	0.860
.AUA1	0.726	0.069	10.534	0.000	0.726	0.705
.AUA2	0.503	0.051	9.864	0.000	0.503	0.632
.AUA3	0.489	0.046	10.727	0.000	0.489	0.723
.AUA4	0.545	0.057	9.527	0.000	0.545	0.603
AS	0.302	0.057	5.260	0.000	1.000	1.000
AF	0.258	0.058	4.466	0.000	1.000	1.000
MI	0.233	0.051	4.567	0.000	1.000	1.000
AUA	0.304	0.069	4.402	0.000	1.000	1.000

### 11.10.1.1 Interpreting the Output

Criteria	Our Results	Criteria met?
Factor loadings significant, strong, proper valence	AS: AS: .50 to .65; AF: .43 to .62; MI: .38 to .58; AUA: .54 to .63	The cross-loaded item is really low (-0.20)
Non-significant chi-square	$\chi^2(202) = 223.332, p = 0.145$	Yes
$CFI \geq .95$	CFI = 0.982	Yes
$RMSEA \leq .05$ (but definitely < .10)	RMSEA = 0.019, 90%CI(0.000, 0.032)	Yes
$SRMR \leq .08$ (but definitely < .10)	SRMR = 0.047	Yes
Combination rule: $CFI \geq .95$ and $SRMR \leq .08$	CFI = 0.982, SRMR = 0.047	
Yes		

We can formally test the difference in models:

```
lavaan::lavTestLRT(corrF, ModInd_M1f)
```

#### Chi-Squared Difference Test

	Df	AIC	BIC	Chisq	Chisq diff	RMSEA	Df	diff	Pr(>Chisq)
ModInd_M1f	202	16655	16845	223.33					

```

corrF      203 16662 16848 232.45      9.1206 0.16344      1  0.002527 **  

---  

Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

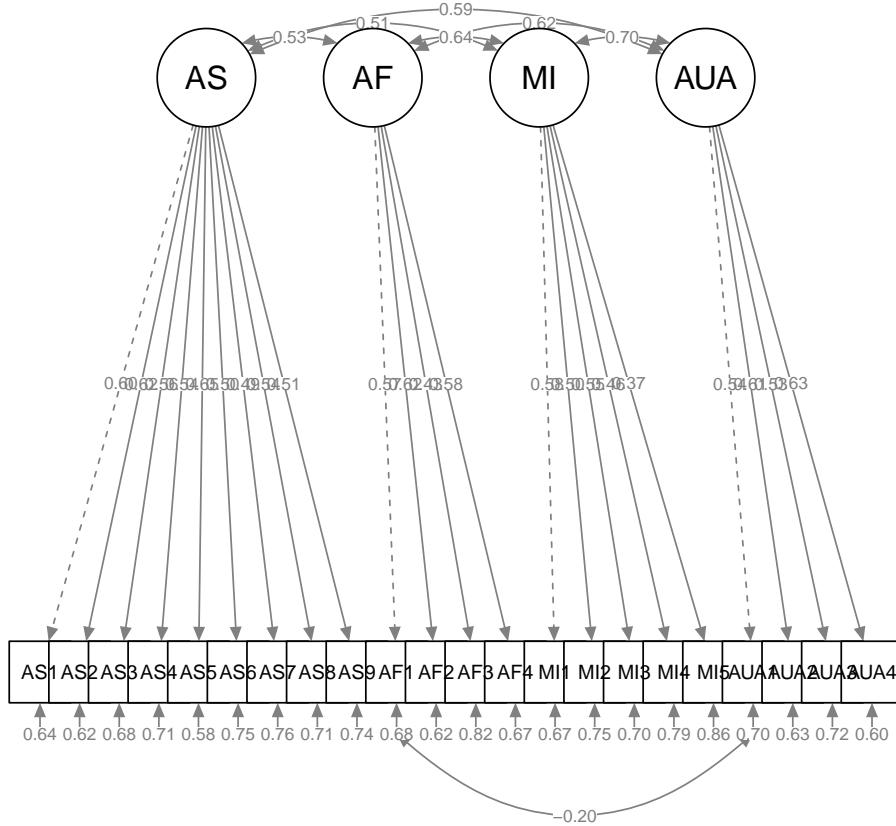
```

We see that the difference between ModInd\_M1f and corrF is statistically significant and that the AIC and BIC are very slightly lower (more favorable) for the respecified model. Because our fit indices were already strong, the correlated error is low, and it “makes a mess” of scoring and interpretation, we will not retain this model and I will not write it up. However, we can learn some things from it:

- The correlated error is statistically significant and negative.
- As predicted, freeing one parameter improved model fit. The respecified model with the additional parameter is the *nesting model*.
- Just because there is statistical support for freeing a parameter, there must be strong rationale for doing so.

The plot below shows the added covariance between the errors for AF1 and AUA1.

```
semPlot::semPaths(ModInd_M1f, layout = "tree", style = "lisrel", what = "col",
whatLabels = "stand")
```



### 11.10.2 Respecifying with Crossloadings

Another route to improving model fit is to allow items to load on more than one factor (i.e., crossload). Let's return to those original modification indices from the `corrF` specification.

```
lavaan:::modindices(corrF, sort = TRUE, minimum.value = 4)
```

lhs	op	rhs	mi	epc	sepc.lv	sepc.all	sepc.nox
282	AF1	~~ AUA1	8.812	-0.124	-0.124	-0.196	-0.196
324	MI2	~~ MI3	8.166	0.135	0.135	0.206	0.206
275	AF1	~~ AF3	7.785	0.115	0.115	0.195	0.195
69	AF	=~ AS2	7.738	-0.435	-0.220	-0.218	-0.218
77	AF	=~ MI1	7.221	0.517	0.261	0.313	0.313
120	AUA	=~ MI5	6.740	0.537	0.297	0.337	0.337
95	MI	=~ AF1	6.221	-0.539	-0.260	-0.290	-0.290
241	AS7	~~ MI4	5.912	0.102	0.102	0.150	0.150

```

319 MI1 ~~ MI5 5.849 -0.092 -0.092 -0.166 -0.166
288 AF2 ~~ MI1 5.755 0.088 0.088 0.169 0.169
78 AF == MI2 5.748 -0.473 -0.239 -0.271 -0.271
193 AS4 ~~ MI4 5.479 -0.097 -0.097 -0.145 -0.145
289 AF2 ~~ MI2 5.191 -0.090 -0.090 -0.155 -0.155
163 AS3 ~~ AS5 5.059 -0.089 -0.089 -0.153 -0.153
124 AS1 ~~ AS5 5.050 0.085 0.085 0.156 0.156
168 AS3 ~~ AF1 4.910 0.082 0.082 0.143 0.143
112 AUA == AF1 4.850 -0.372 -0.206 -0.230 -0.230
113 AUA == AF2 4.735 0.416 0.230 0.237 0.237
278 AF1 ~~ MI2 4.601 -0.080 -0.080 -0.141 -0.141
63 AS == MI5 4.516 0.277 0.152 0.173 0.173
64 AS == AUA1 4.492 -0.349 -0.192 -0.188 -0.188
67 AS == AUA4 4.463 0.334 0.184 0.193 0.193
259 AS8 ~~ AUA3 4.441 -0.072 -0.072 -0.133 -0.133
182 AS4 ~~ AS6 4.380 -0.065 -0.065 -0.132 -0.132
92 MI == AS7 4.127 0.312 0.151 0.163 0.163
96 MI == AF2 4.072 0.494 0.238 0.245 0.245

```

The highest “factor to item” (as opposed to “item to item”) modification index is  $AF == AS2$ . If we draw a path from AF to AS2, the overall chi-square will be reduced by 7.738 points. Recall, a 1 degree of freedom change in a model will be a statistically significant difference if the chi-square value drops by 4 points, so we can expect this to make a statistically significant difference.

In CFA models, allowing an item to load on more than one factor suggests that the item has something in common with both factors. Theoretically, this could make a great deal of sense. On, though, this creates confusion about scoring measures and interpreting them.

The item in question, AS2 reads, “Others have been surprised when I disagree with them.” Presently, it is assigned to the Ascribed Submissiveness factor. The MI is suggesting that it also be assigned to the Asian Fetishism scale.

In the context of this instrument whose CFA properties are already strong, I find it difficult to justify allowing these errors to covary, but I want to demonstrate the technique. We respecify it by adding AF1 to the MI factor.

```

ModInd_M2 <- "AS =~ AS1 + AS2 + AS3 + AS4 + AS5 + AS6 + AS7 + AS8 + AS9
                  AF =~ AF1 + AF2 + AF3 + AF4 + AS2
                  MI =~ MI1 + MI2 + MI3 + MI4 + MI5
                  AUA =~ AUA1 + AUA2 + AUA3 + AUA4
                  "

```

```

set.seed(240311)
ModInd_M2f <- lavaan:::cfa(ModInd_M2, data = dfGRMSAAW)
lavaan:::summary(ModInd_M2f, fit.measures = TRUE, standardized = TRUE)

```

`lavaan 0.6.17 ended normally after 45 iterations`

`Estimator`

`ML`

Optimization method	NLMINB
Number of model parameters	51
Number of observations	304
Model Test User Model:	
Test statistic	224.190
Degrees of freedom	202
P-value (Chi-square)	0.136
Model Test Baseline Model:	
Test statistic	1439.317
Degrees of freedom	231
P-value	0.000
User Model versus Baseline Model:	
Comparative Fit Index (CFI)	0.982
Tucker-Lewis Index (TLI)	0.979
Loglikelihood and Information Criteria:	
Loglikelihood user model (H0)	-8276.884
Loglikelihood unrestricted model (H1)	-8164.789
Akaike (AIC)	16655.768
Bayesian (BIC)	16845.336
Sample-size adjusted Bayesian (SABIC)	16683.590
Root Mean Square Error of Approximation:	
RMSEA	0.019
90 Percent confidence interval - lower	0.000
90 Percent confidence interval - upper	0.032
P-value H_0: RMSEA <= 0.050	1.000
P-value H_0: RMSEA >= 0.080	0.000
Standardized Root Mean Square Residual:	
SRMR	0.045
Parameter Estimates:	
Standard errors	Standard
Information	Expected
Information saturated (h1) model	Structured

## Latent Variables:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
AS ==						
AS1	1.000				0.544	0.593
AS2	1.437	0.194	7.424	0.000	0.781	0.774
AS3	0.974	0.125	7.806	0.000	0.530	0.564
AS4	0.913	0.121	7.514	0.000	0.496	0.536
AS5	1.149	0.135	8.536	0.000	0.625	0.638
AS6	0.675	0.095	7.124	0.000	0.367	0.501
AS7	0.835	0.119	7.032	0.000	0.454	0.493
AS8	0.913	0.121	7.542	0.000	0.496	0.539
AS9	0.760	0.105	7.221	0.000	0.413	0.510
AF ==						
AF1	1.000				0.501	0.559
AF2	1.222	0.176	6.943	0.000	0.613	0.631
AF3	0.752	0.138	5.456	0.000	0.377	0.426
AF4	1.123	0.170	6.622	0.000	0.563	0.572
AS2	-0.480	0.181	-2.648	0.008	-0.240	-0.238
MI ==						
MI1	1.000				0.481	0.577
MI2	0.916	0.148	6.200	0.000	0.441	0.500
MI3	1.171	0.177	6.597	0.000	0.564	0.551
MI4	0.924	0.157	5.872	0.000	0.445	0.463
MI5	0.690	0.137	5.021	0.000	0.332	0.377
AUA ==						
AUA1	1.000				0.553	0.543
AUA2	0.981	0.140	7.018	0.000	0.543	0.608
AUA3	0.784	0.121	6.459	0.000	0.434	0.528
AUA4	1.083	0.152	7.146	0.000	0.599	0.630

## Covariances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
AS ~~						
AF	0.156	0.031	5.092	0.000	0.572	0.572
MI	0.140	0.028	4.997	0.000	0.534	0.534
AUA	0.182	0.034	5.302	0.000	0.607	0.607
AF ~~						
MI	0.152	0.030	5.003	0.000	0.629	0.629
AUA	0.165	0.034	4.847	0.000	0.597	0.597
MI ~~						
AUA	0.189	0.036	5.301	0.000	0.709	0.709

## Variances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
.AS1	0.545	0.050	10.974	0.000	0.545	0.648
.AS2	0.566	0.063	9.029	0.000	0.566	0.555
.AS3	0.602	0.054	11.167	0.000	0.602	0.682

.AS4	0.610	0.054	11.324	0.000	0.610	0.712
.AS5	0.568	0.054	10.607	0.000	0.568	0.593
.AS6	0.402	0.035	11.496	0.000	0.402	0.749
.AS7	0.643	0.056	11.531	0.000	0.643	0.757
.AS8	0.602	0.053	11.310	0.000	0.602	0.710
.AS9	0.487	0.043	11.456	0.000	0.487	0.740
.AF1	0.552	0.055	10.047	0.000	0.552	0.687
.AF2	0.568	0.063	8.987	0.000	0.568	0.602
.AF3	0.639	0.057	11.232	0.000	0.639	0.818
.AF4	0.652	0.066	9.885	0.000	0.652	0.673
.MI1	0.466	0.047	9.826	0.000	0.466	0.668
.MI2	0.583	0.055	10.673	0.000	0.583	0.750
.MI3	0.730	0.072	10.152	0.000	0.730	0.697
.MI4	0.728	0.066	10.982	0.000	0.728	0.786
.MI5	0.665	0.058	11.515	0.000	0.665	0.858
.AUA1	0.730	0.069	10.536	0.000	0.730	0.705
.AUA2	0.502	0.051	9.795	0.000	0.502	0.630
.AUA3	0.487	0.046	10.679	0.000	0.487	0.721
.AUA4	0.545	0.058	9.475	0.000	0.545	0.603
AS	0.296	0.057	5.220	0.000	1.000	1.000
AF	0.251	0.057	4.403	0.000	1.000	1.000
MI	0.232	0.051	4.553	0.000	1.000	1.000
AUA	0.306	0.070	4.394	0.000	1.000	1.000

### 11.10.2.1 Interpreting the Output

Criteria	Our Results	Criteria met?
Factor loadings significant, strong, proper valence	AS:49. to .77; AF: -.24 to .63; MI: .38 to .58; AUA: .53 to .63	No! When added to AF, AS1 is in the wrong direction
Non-significant chi-square	$\chi^2(202) = 224.190, p = 0.136$	Yes
$CFI \geq 0.982$	CFI = 0.980	Yes
$RMSEA \leq .05$ (but definitely < .10)	RMSEA = 0.019, 90%CI(0.000, 0.032)	Yes
$SRMR \leq .08$ (but definitely < .10)	SRMR = 0.045	Yes
Combination rule: $CFI \geq .95$ and $SRMR \leq .08$	CFI = 0.982, SRMR = 0.045	Yes

We can formally test the difference in models.

```
lavaan::lavTestLRT(corrF, ModInd_M2f)
```

## Chi-Squared Difference Test

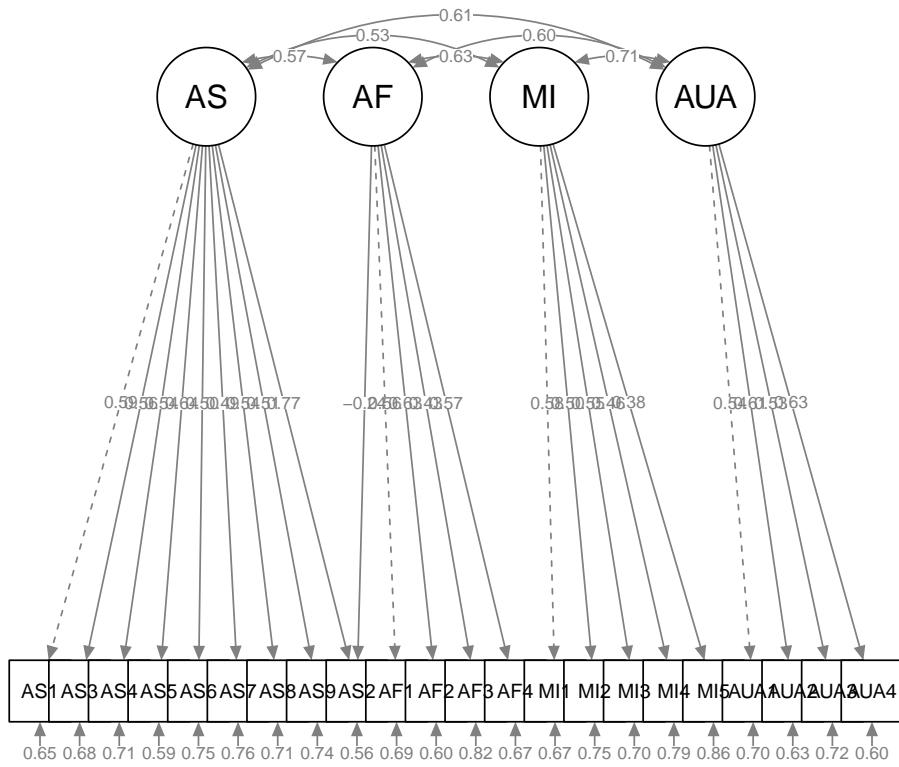
	Df	AIC	BIC	Chisq	Chisq diff	RMSEA	Df	diff	Pr(>Chisq)
ModInd_M2f	202	16656	16845	224.19					
corrF	203	16662	16848	232.45	8.2622	0.15456	1	0.004048	**
	---								
Signif. codes:	0	'***'	0.001	'**'	0.01	'*'	0.05	'. '	0.1 ' ' 1

AIC and BIC are able to compare nested and non-nested models. Models with the lower values are superior. They both favor the model that allows AF1 to crossload on MI.

The  $\chi^2_D$  can only be used for nested models (where items/indicators are identical – the only difference is the presence/absence of parameters). If it is statistically significant, the better model is the one with the lower chi-square value. This, too, favors the correlated factors model,  $\chi^2(1) = 8.262, p = 0.004$ .

Diagramming this model helps further clarify how we have specified this crossloading. It's maybe tough to see, but AF1 now has arrows pointing from the AF and MI factors.

```
semPlot::semPaths(ModInd_M2f, layout = "tree", style = "lisrel", what = "col",
whatLabels = "stand")
```



After each step, we should look again for modification indices.

```
lavaan:::modindices(ModInd_M2f, sort = TRUE, minimum.value = 4)
```

lhs	op	rhs	mi	epc	sepc.lv	sepc.all	sepc.nox
282	AF1	~~ AUA1	9.276	-0.127	-0.127	-0.200	-0.200
324	MI2	~~ MI3	8.250	0.135	0.135	0.207	0.207
275	AF1	~~ AF3	7.525	0.113	0.113	0.190	0.190
77	AF	=~ MI1	7.359	0.520	0.261	0.312	0.312
120	AUA	=~ MI5	6.739	0.538	0.298	0.338	0.338
78	AF	=~ MI2	6.647	-0.506	-0.254	-0.288	-0.288
124	AS1	~~ AS5	6.220	0.094	0.094	0.169	0.169
319	MI1	~~ MI5	5.868	-0.092	-0.092	-0.166	-0.166
241	AS7	~~ MI4	5.828	0.101	0.101	0.148	0.148
288	AF2	~~ MI1	5.707	0.087	0.087	0.169	0.169
193	AS4	~~ MI4	5.463	-0.096	-0.096	-0.145	-0.145
95	MI	=~ AF1	5.441	-0.492	-0.237	-0.264	-0.264
65	AS	=~ AUA1	4.882	-0.372	-0.202	-0.199	-0.199

289	AF2	~~	MI2	4.780	-0.086	-0.086	-0.149	-0.149
112	AUA	=~	AF1	4.739	-0.367	-0.203	-0.226	-0.226
64	AS	=~	MI5	4.561	0.286	0.156	0.177	0.177
259	AS8	~~	AUA3	4.480	-0.072	-0.072	-0.133	-0.133
168	AS3	~~	AF1	4.472	0.079	0.079	0.136	0.136
68	AS	=~	AUA4	4.463	0.342	0.186	0.195	0.195
163	AS3	~~	AS5	4.382	-0.082	-0.082	-0.140	-0.140
182	AS4	~~	AS6	4.229	-0.064	-0.064	-0.129	-0.129
278	AF1	~~	MI2	4.045	-0.075	-0.075	-0.132	-0.132

Not surprisingly, these values continue to be quite low, and I would not propose that we make any of the modifications (not even the ones I have just demonstrated).

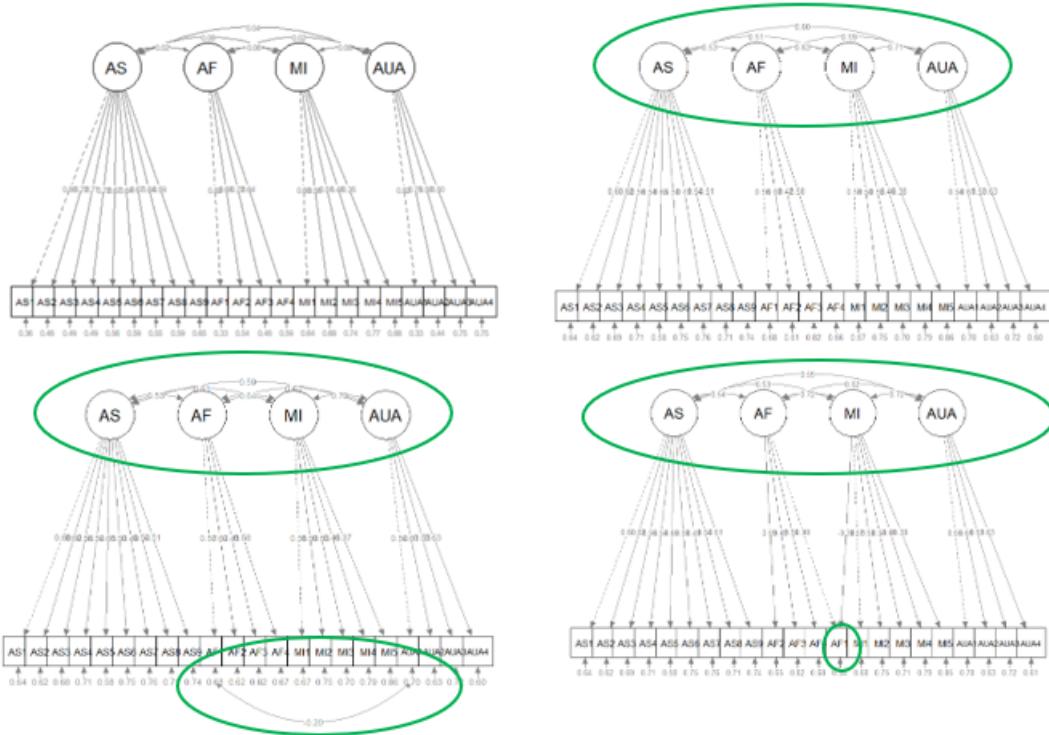


Figure 11.4: Side by side comparison of correlated, uncorrelated models, error covarying, and cross-loading models

Looking at the models side-by-side, we can continue to think about the nested-to-nesting continuum. The *uncorrF* (upper left) model is *nested* (fewer specified parameters, higher degrees of freedom) in the *corrF* model (upper right). Our initial comparison was of these two models. We expected *corrF* to have superior fit, and it did!

We then compared the *corrF* model to the two models below. In these comparisons *corrF* was nested in each of the lower models which had one parameter freed (the error covariance on the lower left; the cross-loading on the lower right). As is common, each of these nesting models (more parameters, fewer degrees of freedom) had better fit. However, because the additions were not theoretically justifiable (and the fit for *corrF* was satisfactory), we did not retain these respecifications.

Think back to the don't break the ice analogy – freeing all those parameters gets closer to the just-identified circumstance where all the relations in the sample covariance matrix are allowed to relate to each other (none are set to 0.0 or knocked out of the ice frame).



Source: <https://www.flickr.com/photos/arfsb/4407495674>

## 11.11 Modeling the GRMSAAW as a Second-Order Structure

Another approach to model building is to explore alternative factor structures. Let's investigate a second-order model.

A **second-order model** represents the hypothesis that a second-order factor,  $g$ , causes each of the identified **first-order factors**. Note that:

- The first-order factors have indicators, but the general factor has none; that is, the second-order factor is measured only indirectly through the indicators of the first-order factors.
- The specification of  $g$  as a common cause of the lower order factors implies that any additional association between the first-order factors is spurious.
- There must be at least three first-order factors or their disturbance variances may be under-identified;
  - each first-order factor should have at least two indicators; more is better
- There are two options for scaling  $g$ ;
  - fixing the direct effect of  $g$  on one factor (usually the first or last) to 1.0; or

- fixing the variance of  $g$  to 1.0 (standardizing it); this leaves all direct effects of  $g$  on the first-order factors as free parameters.

In our second-order model, we will add an the overall GRMS factor as our  $g$  below the four existing factors.

```
secondM <- "AS =~ AS1 + AS2 + AS3 + AS4 + AS5 + AS6 + AS7 + AS8 + AS9
            AF =~ AF1 + AF2 + AF3 + AF4
            MI =~ MI1 + MI2 + MI3 + MI4 + MI5
            AUA =~ AUA1 + AUA2 + AUA3 + AUA4
            GRMS =~ AS + AF + MI + AUA"
```

Next, we extract the results from the `secondM` object with the `lavaan::cfa()` function.

```
set.seed(240311)
secondF <- lavaan::cfa(secondM, data = dfGRMSAAW)
lavaan::summary(secondF, fit.measures = TRUE, standardized = TRUE, rsquare = TRUE)
```

`lavaan 0.6.17 ended normally after 41 iterations`

Estimator	ML
Optimization method	NLMINB
Number of model parameters	48
Number of observations	304

Model Test User Model:

Test statistic	234.741
Degrees of freedom	205
P-value (Chi-square)	0.076

Model Test Baseline Model:

Test statistic	1439.317
Degrees of freedom	231
P-value	0.000

User Model versus Baseline Model:

Comparative Fit Index (CFI)	0.975
Tucker-Lewis Index (TLI)	0.972

Loglikelihood and Information Criteria:

Loglikelihood user model (H0)	-8282.159
Loglikelihood unrestricted model (H1)	-8164.789

Akaike (AIC)	16660.319
Bayesian (BIC)	16838.736
Sample-size adjusted Bayesian (SABIC)	16686.504

Root Mean Square Error of Approximation:

RMSEA	0.022
90 Percent confidence interval - lower	0.000
90 Percent confidence interval - upper	0.034
P-value H_0: RMSEA <= 0.050	1.000
P-value H_0: RMSEA >= 0.080	0.000

Standardized Root Mean Square Residual:

SRMR	0.047
------	-------

Parameter Estimates:

Standard errors	Standard
Information	Expected
Information saturated (h1) model	Structured

Latent Variables:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
AS =~						
AS1	1.000				0.549	0.598
AS2	1.138	0.137	8.326	0.000	0.625	0.618
AS3	0.959	0.124	7.740	0.000	0.526	0.560
AS4	0.902	0.121	7.475	0.000	0.495	0.535
AS5	1.154	0.134	8.589	0.000	0.633	0.647
AS6	0.674	0.094	7.142	0.000	0.370	0.505
AS7	0.833	0.118	7.043	0.000	0.457	0.496
AS8	0.908	0.120	7.539	0.000	0.498	0.541
AS9	0.759	0.105	7.242	0.000	0.417	0.514
AF =~						
AF1	1.000				0.506	0.564
AF2	1.193	0.174	6.847	0.000	0.603	0.621
AF3	0.740	0.137	5.402	0.000	0.374	0.423
AF4	1.134	0.171	6.644	0.000	0.573	0.582
MI =~						
MI1	1.000				0.476	0.570
MI2	0.937	0.151	6.197	0.000	0.446	0.505
MI3	1.201	0.182	6.590	0.000	0.571	0.558
MI4	0.928	0.160	5.798	0.000	0.442	0.459
MI5	0.698	0.140	4.992	0.000	0.332	0.377
AUA =~						
AUA1	1.000				0.550	0.541

AUA2	0.988	0.141	6.988	0.000	0.544	0.609
AUA3	0.788	0.123	6.427	0.000	0.434	0.528
AUA4	1.091	0.153	7.112	0.000	0.600	0.631
GRMS ==						
AS	1.000				0.683	0.683
AF	1.001	0.184	5.450	0.000	0.742	0.742
MI	1.033	0.183	5.657	0.000	0.814	0.814
AUA	1.244	0.222	5.594	0.000	0.847	0.847

Variances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
.AS1	0.540	0.050	10.844	0.000	0.540	0.642
.AS2	0.630	0.059	10.679	0.000	0.630	0.618
.AS3	0.606	0.055	11.116	0.000	0.606	0.687
.AS4	0.611	0.054	11.264	0.000	0.611	0.714
.AS5	0.558	0.054	10.413	0.000	0.558	0.582
.AS6	0.400	0.035	11.421	0.000	0.400	0.745
.AS7	0.640	0.056	11.463	0.000	0.640	0.754
.AS8	0.600	0.053	11.230	0.000	0.600	0.708
.AS9	0.484	0.043	11.377	0.000	0.484	0.736
.AF1	0.548	0.055	9.902	0.000	0.548	0.682
.AF2	0.579	0.064	9.040	0.000	0.579	0.614
.AF3	0.641	0.057	11.212	0.000	0.641	0.821
.AF4	0.640	0.066	9.652	0.000	0.640	0.661
.MI1	0.471	0.048	9.902	0.000	0.471	0.676
.MI2	0.579	0.055	10.611	0.000	0.579	0.744
.MI3	0.722	0.072	10.051	0.000	0.722	0.689
.MI4	0.731	0.066	11.000	0.000	0.731	0.789
.MI5	0.665	0.058	11.510	0.000	0.665	0.858
.AUA1	0.733	0.070	10.552	0.000	0.733	0.708
.AUA2	0.500	0.051	9.760	0.000	0.500	0.629
.AUA3	0.488	0.046	10.671	0.000	0.488	0.722
.AUA4	0.544	0.058	9.439	0.000	0.544	0.601
.AS	0.160	0.036	4.499	0.000	0.533	0.533
.AF	0.115	0.034	3.342	0.001	0.449	0.449
.MI	0.076	0.028	2.750	0.006	0.337	0.337
.AUA	0.085	0.034	2.505	0.012	0.282	0.282
GRMS	0.140	0.036	3.877	0.000	1.000	1.000

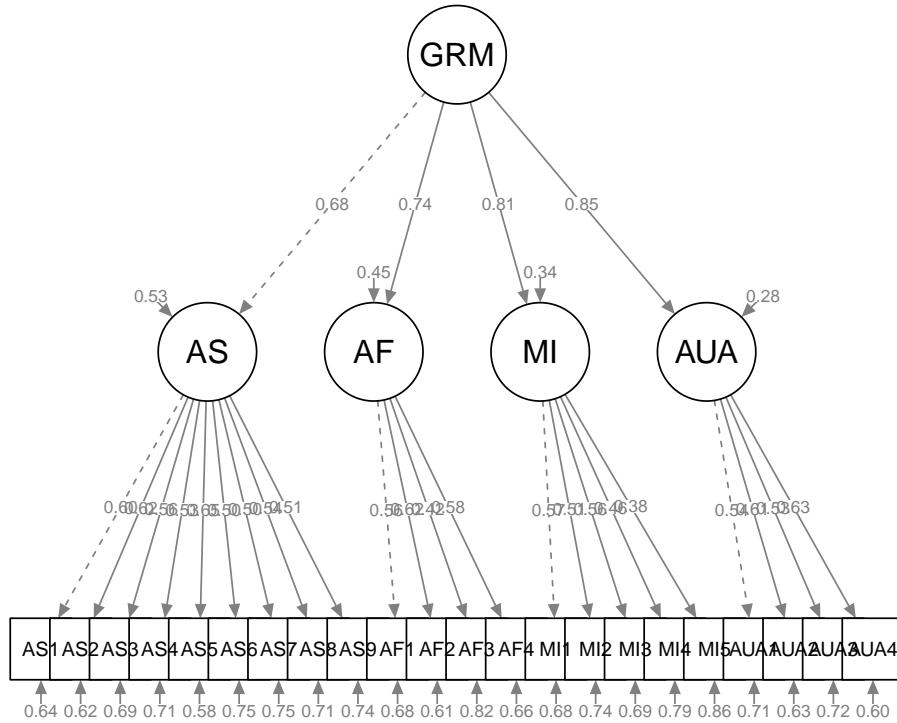
R-Square:

	Estimate
AS1	0.358
AS2	0.382
AS3	0.313
AS4	0.286
AS5	0.418
AS6	0.255
AS7	0.246

AS8	0.292
AS9	0.264
AF1	0.318
AF2	0.386
AF3	0.179
AF4	0.339
MI1	0.324
MI2	0.256
MI3	0.311
MI4	0.211
MI5	0.142
AUA1	0.292
AUA2	0.371
AUA3	0.278
AUA4	0.399
AS	0.467
AF	0.551
MI	0.663
AUA	0.718

As we plot this model we expect to see a “second level” factor predicting each of the “first order” factors. The indicator was set on GRM  $\rightarrow$  AS. Each of the four factors predicts only the items associated with their factor with one item for each factor (the first on the left) specified as the indicator variable.

```
semPlot::semPaths(secondF, layout = "tree", style = "lisrel", what = "col",
  whatLabels = "stand")
```



Again, among my first steps are also to write the code to export the results. The *tidySEM* package has useful functions to export the fit statistics, parameter estimates, and correlations among the latent variables (i.e., factors).

```
secondFFitStats <- tidySEM:::table_fit(secondF)
secondF_paramEsts <- tidySEM:::table_results(secondF, digits=3, columns = NULL)
#because of the second order, there are no correlations among the LVs to request
#to see each of the tables, remove the hashtag
#secondFFitStats
#secondF_paramEsts
```

Next, I export them.

```
write.csv(secondFFitStats, file = "secondFFitStats.csv")
write.csv(secondF_paramEsts, file = "secondF_paramEsts.csv")
```

### 11.11.1 Interpreting the Output

Criteria	Our Results	Criteria met?
Factor loadings significant, strong, proper valence	AS: .50 to .65; AF: .42 to .62; MI: .38 to .57; AUA: .53 to .63; GRMS: .68 to .85	Yes
Non-significant chi-square	$\chi^2(205) = 234.741, p = 0.076$	Yes
$CFI \geq .95$	$CFI = 0.975$	Yes
$RMSEA \leq .05$ (but definitely < .10)	$RMSEA = 0.022, 90\%CI(0.000, 0.034)$	Yes
$SRMR \leq .08$ (but definitely < .10)	$SRMR = 0.047$	Yes
Combination rule: $CFI \geq .95$ and $SRMR \leq .08$	$CFI = 0.975, SRS = 0.047$	Yes

### 11.11.2 Partial Write-up

**Second-order factor model.** Our next model represented a second order structure. Specifically, four first-order factors loaded onto a second factor model and demonstrated adequate fit to the data:  $\chi^2(205) = 234.741, p = 0.076, CFI = 0.975, RMSEA = 0.022, 90$ . Factor loadings ranged from .50 to .65 for the AS scale, .42 to .62 for the AF scale, .38 to .57 for the MI scale, .53 to .63 for the AUA scale, and .68 to .85 for the GRMS total scale.

Determining if models are nested vs. hierarchically arranged can be confusing, especially when it comes to adding in second-order structures. That is, replacing the six correlations (in the correlated factors model) with the second-order factor (fixing the first of the first-order factors to 1.0, so adding only 3 paths to be estimated) is not a clear fixing or freeing of paths. We need to know if they are so that we know if it is appropriate to apply/interpret the  $\chi^2_D$  difference test.

Luckily, the Muthen's (creators of Mplus) have a [discussion](#) post devoted to this and it appears that our correlated factors model is the nesting model for the second-order structure. If there is a statistically significant difference in models, then the correlated factors model is superior.

```
lavaan:::lavTestLRT(uncorrF, corrF, secondF)
```

#### Chi-Squared Difference Test

	Df	AIC	BIC	Chisq	Chisq diff	RMSEA	Df diff	Pr(>Chisq)
corrF	203	16662	16848	232.45				
secondF	205	16660	16839	234.74	2.289	0.02178	2	0.3185
uncorrF	209	16879	17042	461.10	226.361	0.42762	4	<0.0000000000000002

```
corrF
secondF
uncorrF ***
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Although there is not a statistically significant chi-square difference ( $\chi^2(2) = .37923, p = .827$ ) between the correlated factors and second order structure, the AIC and BIC favor (by a very small amount) the second-order model.

If our model fit was poor, we would want to inspect the modification indices and see if it would be justifiable to allow error covariances.

```
lavaan:::modindices(secondF, sort = TRUE, minimum.value = 4)
```

	lhs	op	rhs	mi	epc	sepc.lv	sepc.all	sepc.nox
303	AF1	~~	AUA1	9.880	-0.131	-0.131	-0.207	-0.207
137	GRMS	=~	MI5	9.635	1.388	0.520	0.591	0.591
76	AF	=~	MI1	8.672	0.519	0.262	0.314	0.314
296	AF1	~~	AF3	7.683	0.115	0.115	0.194	0.194
345	MI2	~~	MI3	7.464	0.129	0.129	0.200	0.200
68	AF	=~	AS2	7.396	-0.410	-0.207	-0.205	-0.205
119	AUA	=~	MI5	6.946	0.525	0.289	0.328	0.328
129	GRMS	=~	AF1	6.925	-0.953	-0.357	-0.399	-0.399
130	GRMS	=~	AF2	6.303	1.048	0.393	0.405	0.405
111	AUA	=~	AF1	6.125	-0.450	-0.248	-0.277	-0.277
309	AF2	~~	MI1	5.928	0.088	0.088	0.169	0.169
262	AS7	~~	MI4	5.764	0.101	0.101	0.148	0.148
214	AS4	~~	MI4	5.745	-0.099	-0.099	-0.149	-0.149
121	GRMS	=~	AS2	5.427	-0.614	-0.230	-0.228	-0.228
340	MI1	~~	MI5	5.391	-0.089	-0.089	-0.158	-0.158
145	AS1	~~	AS5	5.270	0.087	0.087	0.159	0.159
94	MI	=~	AF1	5.171	-0.467	-0.222	-0.248	-0.248
189	AS3	~~	AF1	5.066	0.084	0.084	0.145	0.145
184	AS3	~~	AS5	4.864	-0.087	-0.087	-0.150	-0.150
310	AF2	~~	MI2	4.801	-0.086	-0.086	-0.148	-0.148
77	AF	=~	MI2	4.730	-0.400	-0.202	-0.230	-0.230
66	AS	=~	AUA4	4.512	0.316	0.173	0.182	0.182
203	AS4	~~	AS6	4.441	-0.066	-0.066	-0.133	-0.133
280	AS8	~~	AUA3	4.375	-0.071	-0.071	-0.131	-0.131
95	MI	=~	AF2	4.349	0.479	0.228	0.235	0.235
299	AF1	~~	MI2	4.174	-0.075	-0.075	-0.134	-0.134
69	AF	=~	AS3	4.137	0.294	0.148	0.158	0.158
80	AF	=~	MI5	4.006	0.367	0.185	0.211	0.211

The same AF1  $\sim\!\sim$  AUA1 relationship is showing as the item that has a larger modification index than the others. As we saw earlier, freeing it to covary would improve the fit. However, our more parsimonious models (correlated factors, second-order) have excellent fit. Therefore, we will not respecify at this time.

## 11.12 Modeling the GRMSAAW as a Bifactor Model

**Bifactor models** are also known as **nested-factor** and **general-specific** models. Like the second-order model, they involve several specific, correlated constructs that make up a more general

construct of interest. The big difference:

- $g$  in the bifactor model directly affects the indicators but is *orthogonal/unrelated* to the specific factors
- bifactor models where  $g$  covaries with the specific factors may not be identified
- bifactor models partition variance into three nonoverlapping sources:
  - specific factors
  - the general factor ( $g$ )
  - error

Second-order and bifactor models make very different assumptions about whether  $g$  is unrelated to the other factors (bifactor model) or covaries with/mediates those other factors (second-order model).

Take note that the base factor structure for the bifactor model is identical to the second-order structure. The difference is that the script fixes the relations between  $g$  and each of the factors to 0.0; and the relations between each of the factors to each other as 0.0.

```
bifacM <- "
GRMS =~ AS1 + AS2 + AS3 + AS4 + AS5 + AS6 + AS7 + AS8 + AS9 + AF1 + AF2 + AF3 + AF4
AS =~ AS1 + AS2 + AS3 + AS4 + AS5 + AS6 + AS7 + AS8 + AS9
AF =~ AF1 + AF2 + AF3 + AF4
MI =~ MI1 + MI2 + MI3 + MI4 + MI5
AUA =~ AUA1 + AUA2 + AUA3 + AUA4

#fixes the relations between g and each of the factors to 0.0
GRMS ~~ 0*AS
GRMS ~~ 0*AF
GRMS ~~ 0*MI
GRMS ~~ 0*AUA

#fixes the relations (covariances) between each of the factors to 0.0
AS ~~ 0*AF
AS ~~ 0*MI
AS ~~ 0*AUA
AF ~~ 0*MI
AF ~~ 0*AUA
MI ~~ 0*AUA
"
```

```
# On the first run I received a warning; it is not uncommon to add
# the statement 'check.gradient=FALSE' to force a solution. Then it
# is important to closely inspect the results to see if things look
# ok. If you get really stuck it is possible to change optimizers
# through control statements
set.seed(240311)
bifacF <- lavaan:::cfa(bifacM, data = dfGRMSAAW, check.gradient = FALSE)
lavaan:::summary(bifacF, fit.measures = TRUE, standardized = TRUE, rsquare = TRUE)
```

lavaan 0.6.17 ended normally after 100 iterations

Estimator	ML
Optimization method	NLMINB
Number of model parameters	66
Number of observations	304

Model Test User Model:

Test statistic	191.208
Degrees of freedom	187
P-value (Chi-square)	0.401

Model Test Baseline Model:

Test statistic	1439.317
Degrees of freedom	231
P-value	0.000

User Model versus Baseline Model:

Comparative Fit Index (CFI)	0.997
Tucker-Lewis Index (TLI)	0.996

Loglikelihood and Information Criteria:

Loglikelihood user model (H0)	-8260.393
Loglikelihood unrestricted model (H1)	-8164.789
Akaike (AIC)	16652.786
Bayesian (BIC)	16898.110
Sample-size adjusted Bayesian (SABIC)	16688.791

Root Mean Square Error of Approximation:

RMSEA	0.009
90 Percent confidence interval - lower	0.000
90 Percent confidence interval - upper	0.027
P-value H_0: RMSEA <= 0.050	1.000
P-value H_0: RMSEA >= 0.080	0.000

Standardized Root Mean Square Residual:

SRMR	0.039
------	-------

Parameter Estimates:

	Standard errors	Standard				
	Information	Expected				
	Information saturated (h1) model	Structured				
<b>Latent Variables:</b>						
	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
<b>GRMS =~</b>						
AS1	1.000				0.363	0.396
AS2	0.907	0.192	4.719	0.000	0.329	0.326
AS3	1.089	0.209	5.204	0.000	0.395	0.421
AS4	0.917	0.191	4.789	0.000	0.333	0.359
AS5	1.231	0.219	5.616	0.000	0.447	0.456
AS6	0.779	0.160	4.880	0.000	0.283	0.386
AS7	1.077	0.213	5.060	0.000	0.391	0.424
AS8	0.936	0.193	4.847	0.000	0.340	0.369
AS9	0.650	0.157	4.149	0.000	0.236	0.291
AF1	0.877	0.212	4.136	0.000	0.318	0.355
AF2	1.432	0.278	5.149	0.000	0.520	0.535
AF3	0.765	0.200	3.829	0.000	0.278	0.314
AF4	1.215	0.256	4.744	0.000	0.441	0.448
MI1	1.154	0.231	5.001	0.000	0.419	0.501
MI2	0.884	0.211	4.199	0.000	0.321	0.364
MI3	1.109	0.253	4.380	0.000	0.402	0.393
MI4	0.956	0.228	4.196	0.000	0.347	0.361
MI5	0.992	0.220	4.504	0.000	0.360	0.409
AUA1	1.211	0.264	4.580	0.000	0.439	0.432
AUA2	1.235	0.250	4.947	0.000	0.448	0.502
AUA3	0.979	0.213	4.596	0.000	0.355	0.432
AUA4	1.509	0.288	5.240	0.000	0.547	0.576
<b>AS =~</b>						
AS1	1.000				0.416	0.453
AS2	1.398	0.252	5.548	0.000	0.581	0.575
AS3	0.815	0.183	4.464	0.000	0.339	0.360
AS4	0.899	0.191	4.718	0.000	0.374	0.404
AS5	1.056	0.203	5.201	0.000	0.439	0.448
AS6	0.571	0.140	4.080	0.000	0.238	0.324
AS7	0.588	0.167	3.520	0.000	0.245	0.265
AS8	0.861	0.187	4.617	0.000	0.358	0.389
AS9	0.867	0.177	4.892	0.000	0.360	0.444
<b>AF =~</b>						
AF1	1.000				0.620	0.691
AF2	0.342	0.162	2.108	0.035	0.212	0.218
AF3	0.446	0.196	2.270	0.023	0.276	0.312
AF4	0.474	0.207	2.293	0.022	0.294	0.298
<b>MI =~</b>						
MI1	1.000				0.190	0.227
MI2	1.874	0.766	2.446	0.014	0.355	0.403
MI3	2.959	1.340	2.208	0.027	0.561	0.548

MI4	1.326	0.610	2.174	0.030	0.251	0.261
MI5	-0.043	0.401	-0.107	0.915	-0.008	-0.009
AUA =~						
AUA1	1.000				0.419	0.412
AUA2	0.794	0.370	2.147	0.032	0.333	0.373
AUA3	0.590	0.275	2.145	0.032	0.247	0.301
AUA4	0.444	0.245	1.813	0.070	0.186	0.196

## Covariances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
GRMS ~~						
AS	0.000				0.000	0.000
AF	0.000				0.000	0.000
MI	0.000				0.000	0.000
AUA	0.000				0.000	0.000
AS ~~						
AF	0.000				0.000	0.000
MI	0.000				0.000	0.000
AUA	0.000				0.000	0.000
AF ~~						
MI	0.000				0.000	0.000
AUA	0.000				0.000	0.000
MI ~~						
AUA	0.000				0.000	0.000

## Variances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
.AS1	0.536	0.050	10.644	0.000	0.536	0.638
.AS2	0.574	0.064	8.986	0.000	0.574	0.563
.AS3	0.612	0.054	11.277	0.000	0.612	0.693
.AS4	0.606	0.054	11.129	0.000	0.606	0.708
.AS5	0.566	0.054	10.518	0.000	0.566	0.591
.AS6	0.401	0.035	11.531	0.000	0.401	0.746
.AS7	0.637	0.055	11.605	0.000	0.637	0.750
.AS8	0.605	0.054	11.220	0.000	0.605	0.713
.AS9	0.472	0.044	10.829	0.000	0.472	0.718
.AF1	0.318	0.156	2.041	0.041	0.318	0.396
.AF2	0.628	0.059	10.673	0.000	0.628	0.666
.AF3	0.628	0.060	10.552	0.000	0.628	0.804
.AF4	0.688	0.067	10.333	0.000	0.688	0.710
.MI1	0.486	0.045	10.741	0.000	0.486	0.697
.MI2	0.548	0.067	8.199	0.000	0.548	0.705
.MI3	0.572	0.132	4.333	0.000	0.572	0.546
.MI4	0.742	0.067	11.024	0.000	0.742	0.802
.MI5	0.646	0.057	11.354	0.000	0.646	0.833
.AUA1	0.667	0.097	6.904	0.000	0.667	0.644
.AUA2	0.484	0.063	7.679	0.000	0.484	0.608
.AUA3	0.488	0.049	9.917	0.000	0.488	0.723

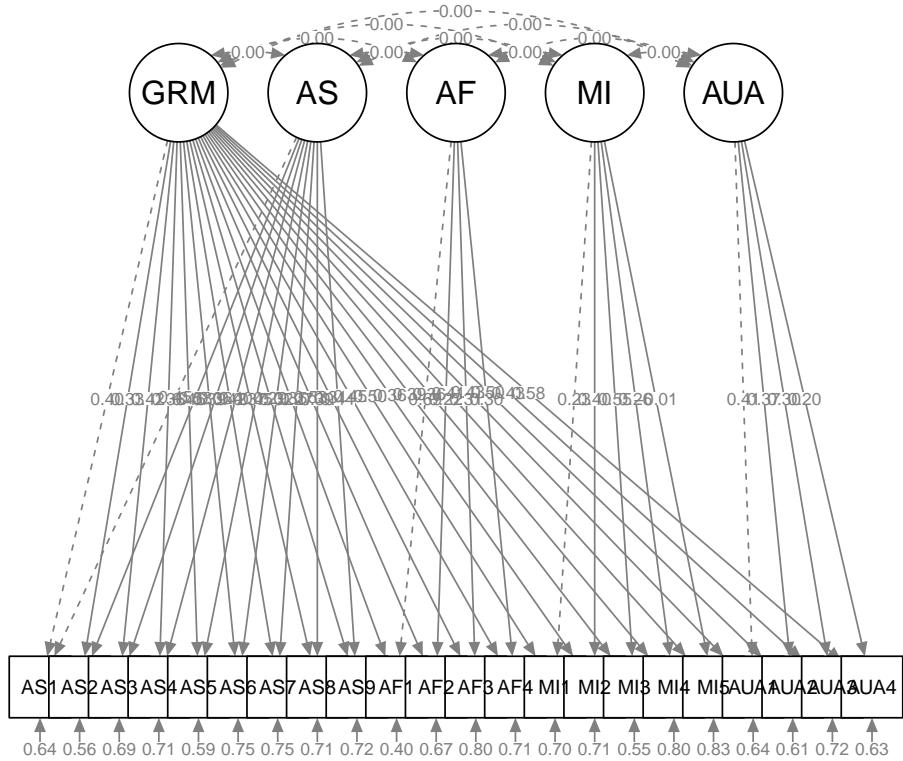
.AUA4	0.570	0.055	10.420	0.000	0.570	0.630
GRMS	0.132	0.043	3.053	0.002	1.000	1.000
AS	0.173	0.051	3.399	0.001	1.000	1.000
AF	0.384	0.165	2.327	0.020	1.000	1.000
MI	0.036	0.026	1.373	0.170	1.000	1.000
AUA	0.176	0.100	1.752	0.080	1.000	1.000

R-Square:

	Estimate
AS1	0.362
AS2	0.437
AS3	0.307
AS4	0.292
AS5	0.409
AS6	0.254
AS7	0.250
AS8	0.287
AS9	0.282
AF1	0.604
AF2	0.334
AF3	0.196
AF4	0.290
MI1	0.303
MI2	0.295
MI3	0.454
MI4	0.198
MI5	0.167
AUA1	0.356
AUA2	0.392
AUA3	0.277
AUA4	0.370

Providing a traditional diagram of the bifactor model requires some extra steps. The default from semPlot's *semPaths()* function produces this:

```
semPlot::semPaths(bifacF, layout = "tree", style = "lisrel", what = "col",
  whatLabels = "stand")
```



While it is an accurate depiction, I was seeking the traditional illustration. I found some instructive discussion on Sacha Epskamp's *semPlot* [repo](#) on GitHub.

We can think of the variables in our model as numbered. The items take the first numbers, followed by  $g$ , and then each of the factors. We need to represent them in a matrix of 0s and numbers. Let's start by mapping them out. The top row is the factors (4), the second row is items (22), the bottom row is  $g$  (1)

```
[1, ] 0 0 0 24 0 0 0 0 25 0 0 0 0 0 26 0 0 0 0 0 27 0 0 [2, ] 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17  
18 19 20 21 22 [3, ] 0 0 0 0 0 0 0 0 0 0 0 28 0 0 0 0 0 0 0 0 0
```

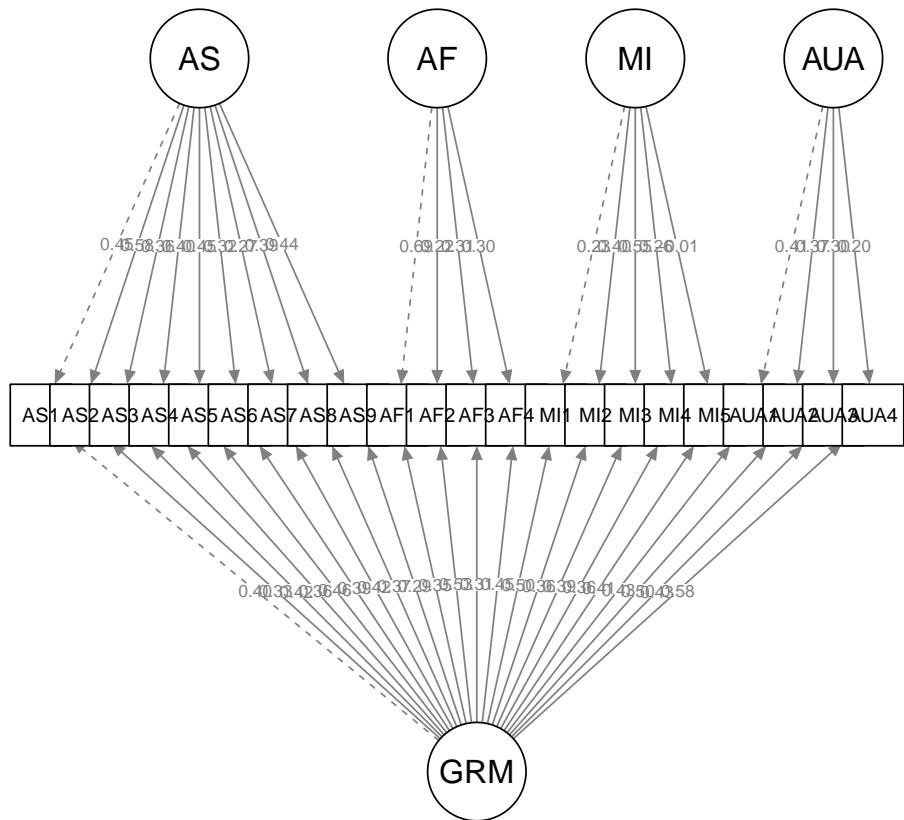
```

m = matrix(nrow = 3, ncol = 22)
m[1, ] = c(0, 0, 0, 0, 24, 0, 0, 0, 0, 0, 25, 0, 0, 0, 0, 0, 26, 0, 0, 0,
         0, 27, 0)
m[2, ] = 1:22
m[3, ] = c(0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 23, 0, 0, 0, 0, 0, 0, 0, 0, 0,
         0, 0)
m

```

```
[,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10] [,11] [,12] [,13] [,14]
[1,] 0 0 0 0 24 0 0 0 0 0 25 0 0 0
[2,] 1 2 3 4 5 6 7 8 9 10 11 12 13 14
[3,] 0 0 0 0 0 0 0 0 0 0 0 23 0 0
[,15] [,16] [,17] [,18] [,19] [,20] [,21] [,22]
[1,] 0 26 0 0 0 0 27 0
[2,] 15 16 17 18 19 20 21 22
[3,] 0 0 0 0 0 0 0 0
```

```
semPlot::semPaths(bifacF, "model", "std", layout = m, residuals = FALSE,
  exoCov = FALSE)
```



Among my first steps are also to write the code to export the results. The *tidySEM* package has useful functions to export the fit statistics, parameter estimates, and correlations among the latent variables (i.e., factors).

```
bifacFFitStats <- tidySEM::table_fit(bifacF)
bifacF_paramEsts <- tidySEM::table_results(bifacF, digits=3, columns = NULL)
bifacFCorrs <- tidySEM::table_corrs(bifacF, digits=3)
#to see each of the tables, remove the hashtab
#bifacFFitStats
#bifacF_paramEsts
#bifacFCorrs
```

Next, I export them.

```
write.csv(bifacFFitStats, file = "bifacFFitStats.csv")
write.csv(bifacF_paramEsts, file = "bifacF_paramEsts.csv")
write.csv(bifacFCorrs, file = "bifacFCorrs.csv")
```

**Troubleshooting tip:** If, while working with this function you get the error: “*Error in file(file, ifelse(append, “a”, “w”)) : cannot open the connection*”, it’s because the .csv file that received your table is still open. R is just trying to write over it. A similar error happens when knitting.

### 11.12.1 Interpreting the Output

Criteria	Our Results	Criteria met?
Factor loadings significant, strong, proper valence	GRMS: .29 to .58; AS: .27 to .58; AF: .22 to .69; MI: -.01 to .55; AUA: .20 to .41	
MI5 is near-zero and negative		
Non-significant chi-square	$\chi^2(187) = 191.208, p = 0.401$	Yes
$CFI \geq .95$	CFI = 0.997	Yes
$RMSEA \leq .05$ (but definitely < .10)	RMSEA = 0.009, 90%CI(0.000, 0.027)	Yes
$SRMR \leq .08$ (but definitely < .10)	SRMR = 0.039	Yes
Combination rule: $CFI \geq .95$ and $SRMR \leq .08$	CFI = 0.997, SRS = 0.039	Yes

As promised, even in spite of the wiggly factor loadings, the model fit improves. This is another example of the nesting model generally having the best fit.

### 11.12.2 Partial Write-up

**Bifactor model.** The bifactor model regressed each item on its respective factor while simultaneously regressing each indicator onto an overall GRMS scale. This model had the best fit of those compared thus far:  $\chi^2(187) = 164.080, p = .885$ ,  $CFI = 0.997$  ,  $RMSEA = 0.009$ , 90%CI(0.000, 0.027),  $SRMR = 0.039$ . Factor loadings for the four factors ranged from .27 to .58 for the AS scale, .22 to .69 for the AF scale, -.01 to .55 for the MI scale, and .20 to .41 for the AUA scale. Factor loadings for the overall GRMSAAW ( $g$ ) ranged from .29 to .58.

On the basis of this evaluation, we are finding all four models to be satisfactory (in terms of fit): the single-order uncorrelated factors (uncorrF), the single-order correlated factors model (corrF), the second order factor (secondF), and the bifactor model (bifacF). We can use *lavaan's* *lavTest()* function to compare them. No matter the order that we enter them, the function orders them according to their degrees of freedom.

```
lavaan:::lavTestLRT(uncorrF, corrF, secondF, bifacF)
```

#### Chi-Squared Difference Test

	Df	AIC	BIC	Chisq	Chisq diff	RMSEA	Df diff	Pr(>Chisq)
bifacF	187	16653	16898	191.21				
corrF	203	16662	16848	232.45	41.244	0.07204	16	0.000511
secondF	205	16660	16839	234.74	2.289	0.02178	2	0.318460
uncorrF	209	16879	17042	461.10	226.361	0.42762	4 < 0.00000000000000022	

bifacF
corrF ***
secondF
uncorrF ***
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

We see that the bifacF and corrF models are statistically significantly different from each other. The AIC favors the bifacF; the BIC favors the corrF. We may be interested in knowing how it compares to the secondF model. The two models are statistically significantly different. The lower value of the AIC favors the bifactor model; the lower value of the BIC favors the second-order model.

```
lavaan:::lavTestLRT(secondF, bifacF)
```

#### Chi-Squared Difference Test

	Df	AIC	BIC	Chisq	Chisq diff	RMSEA	Df diff	Pr(>Chisq)
bifacF	187	16653	16898	191.21				
secondF	205	16660	16839	234.74	43.533	0.068309	18	0.0006725 ***
---								
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1								

In the article, Keum et al. [2018] reported the best fit for the bifactor model. They reported strong, significant, and properly balanced loadings for the  $g$  factor as well as for each of the group factors. Our wiggly factor loadings on  $g$  and the MI scale are likely an artifact of simulating the data from the EFA factor loadings.

## 11.13 Another Look at Omega

Now that we've had an introduction to CFA/SEM – and the second-order and bifactor models in particular – let's look again the  $\omega$  grouping of reliability estimates.

In prior lessons we used the *psych* package's *omegaSem()* function with raw data. The package estimated a family of model-based estimates that examine the correlations or covariances of the items and decomposed the test variance into that which is

- common to all items (**g**, a general factor),
- specific to some items (**f**, orthogonal group factors), and
- unique to each item (confounding **s** specific, and **e** error variance).

When using raw data or a correlation matrix as the object for the omega analysis, it is possible to specify the number of factors, but the procedure is *exploratory* and there is no guarantee that the items will associate with the intended factor. When we are concerned with the omega reliability estimates for clearly specified factor structure we can feed our *lavaan::cfa* models to *semTools::reliability()* function and obtain the estimates.

### 11.13.1 Omega *h* for Bifactor Models

In bifactor models the general factor captures the variance common across all items and the specific factors account for what is left over. Specific factors represent what is common across members of that factor, separate from what is claimed by *g*.

In the context of a bifactor model, the reliability measure,  $\omega_h$ , represents the proportion of total-score variance due to a single, general construct that influences all items, despite the multidimensional nature of the item set [Flora, 2020a,b].

Stated in terms of the GRMSAAW,  $\omega_h$  represents the extent to which the GRMSAAW total score provides a reliable measure of a construct represented by a general factor that influences all items in a multidimensional scale over and above the AS, AF, MI, and AUA subscales.

If we use the *semTools::reliability()* function, we pass it the object we created from our *lavaan::cfa()*. Flora's article and supplementary materials [Flora, 2020a,b] provide an excellent description and review of how to specify and interpret  $\omega_h$  with *semTools::reliability()*.

```
semTools::reliability(bifacF, return.total = TRUE)
```

	GRMS	AS	AF	MI	AUA	total
alpha	0.8485768	0.8015849	0.6327835	0.6130925	0.6665705	0.8485768
omega	0.8469630	0.6910784	0.4645562	0.3778777	0.3888254	0.8725861
omega2	0.7051557	0.4323840	0.2953065	0.2210012	0.2061386	0.8725861
omega3	0.7059886	0.4324453	0.2949845	0.2193985	0.2060547	0.8736167
avevar	NA	NA	NA	NA	NA	0.3285899

In the case of the bifactor model, the estimates listed under the GRMS column pertain to the general GRMS factor. We typically focus on the omega2 and omega3 values as the indicators of

$\omega_h$ . Flora [2020a] indicates that *omega2* is calculated using the model-implied variance of the total score in its denominator and *omega3* is calculated using the observed sample variance of  $X$ . To the degree that these two values are different from each other, we may have concerns. In our data, *omega2* = 0.705 and *omega3* = 0.706. These values indicate the proportion of GRMS total-score variance that is due to a general factor, over-and-above, the influence of effects that are specific to the group factors (i.e., AS, AF, MI, AUA).

The *omega2* and *omega3* values in the AS through AUA columns are the *omega-hierarchical-subscale*. These analyses indicate how well a given subscale reliably measures a narrower construct that is *independent* from the broader higher-order construct that also influences the other subscales. Flora [2020b] notates these as  $\omega_{h-ss}$  (omega-higherarchical-subscale). Specifically,  $\omega_{h-ss}$  represents the proportion of variance in a subscale that is due to the corresponding specific factor, over and above the influence of the general factor. Comparing the relative values to each other can provide some indication of the source of reliable variance. We see that the AS, AF, and AUA factors are considerably lower than the GRMS, however they are holding their own. Taken together, the omega values from our bifactor model suggest a strong general factor with subscales that are meaningful and useful. Our results paralleled the pattern reported in Keum et al. [2018]. In their follow-up investigation of construct validity, they used structural equation modeling with the *bifactor* model to investigate the relationships between the GRMSAAW with relevant constructs.

In bifactor models, the multidimensionality of items (i.e., the existence of factors) is considered to be a “nuisance” [Flora, 2020b] for the measurement of a broad, general construct. This is different from hierarchical models such as the second-order factor structure. Since we can calculate  $\omega_h$  for it, let’s look at it, next.

### 11.13.2 $\omega_h$ for Second Order Models

In the second-order structure, the researcher hypothesizes that there is a broad, overarching construct indirectly influencing all items in a test through more conceptually narrow constructs that directly influence different groupings of items. This hypothesis implies that item-level data arise from a higher order model, in which the second-order factor, causes individual differences in the first-order factor, which directly influences the observed item responses.

In this case,  $\omega_{ho}$  ( $ho$  = higher order) [Flora, 2020b] represents the proportion of total-score variance that is due to the higher-order factor. As such, it represents the reliability of a total score for measuring a single construct that influences all items.

To use *semTools* we switch functions to *reliabilityL2*. The specification

```
semTools::reliabilityL2(secondF, "GRMS")
```

omegaL1	omegaL2	partialOmegaL1
0.7038702	0.8547458	0.8420019

The estimate listed under *omegaL1* ( $\omega_{ho}$ ) represents the proportion of GRMSAAW total score variance due to the higher-order factor. The value of 0.704 is consistent with what we saw in the bifactor model.

We can apply the *semTools::reliability()* function to the R object that holds the second-order results to obtain omega values for the subscales. Below the alpha coefficients, the omega values indicate

how reliably each subscale measures its lower order factor. For example, 80% of the total variance of a total score comprised of only the 9 items in the AS scale is explained by the AS lower order factor.

```
semTools::reliability(secondF)
```

	AS	AF	MI	AUA
alpha	0.8015849	0.6327835	0.6130925	0.66665705
omega	0.8045496	0.6370410	0.6184523	0.66666061
omega2	0.8045496	0.6370410	0.6184523	0.66666061
omega3	0.8052908	0.6352363	0.6193718	0.6637510
avevar	0.3194991	0.3111493	0.2500576	0.3361644

Using the omega2 values, the subscale omegas range from .62 to .80. It is helpful to see how the omega values align with the alpha coefficients.

### 11.13.3 Partial Write-up

Given that we landed on the bifactor model as our final solution, here's how I might represent the omega results.

As estimates of model-based internal consistency associated with the bifactor model, we calculated omega hierarchical ( $\omega_h$ ) and omega hierarchical subscale ( $\omega_{h-ss}$ ).  $\omega_h$ , represents the proportion of total-score variance due to a single, general construct that influences all items, despite the multidimensional nature of the item set [Flora, 2020a,b]. Our  $\omega_h$  value of 0.71 indicates that 71% of the variance of the GRMS total scores are attributable to individual differences on the general factor.  $\omega_{h-ss}$  for the subscales ranged from 0.21 to 0.43. Taken together, the omega values from our bifactor model suggest a strong general factor with subscales that are meaningful and useful.

## 11.14 Preparing an Overall APA Style Results Section

**Model testing.** To evaluate the models we, we used confirmatory factor analysis (CFA) in the R package, *lavaan* (v.0.6-17) with maximum likelihood estimation. Our sample size was 304. We selected fit criteria for their capacity to assess different aspects of the statistical analysis. As is common among SEM researchers, we reported the Chi-square goodness of fit ( $\chi^2$ ). This evaluates the discrepancy between the unrestricted sample matrix and the restricted covariance matrix. Although the associated  $p$  value indicates adequate fit when the value is non-significant, it is widely recognized that large sample size can result in a statistically significant  $p$  value [Byrne, 2016b]. The comparative fit index (CFI) is an incremental index, comparing the hypothesized model to the independent/baseline model. Adequate fit is determined when CFI values are at least .90 and perhaps higher than .95 [Kline, 2016]. The root mean square error of approximation (RMSEA) takes into account the error of approximation in the population and expresses it per degree of freedom. As such, the fit indicator considers the complexity

of the model. Ideal values are equal to or less than .05, values less than .08 represent reasonable fit, and values between .08 and .10 represent mediocre fit. The standardized root mean residual (SRMR) is a standardized measure of the mean absolute covariance residual – the overall difference between the observed and predicted correlations. Values greater than .10 may indicate poor fit and inspection of residuals is then advised. Because we were interested in comparing nested models we used the Chi-square difference test where a significant chi-square indicates statistically significant differences in models. Additionally, we used Akaike's Information Criterion (AIC) and the Bayesian Information Criterion (BIC) that take model complexity and sample size into consideration. Models with lower values on each are considered to be superior. Kline [2016] advised researchers to be cautious when using these criteria as strict cut-offs. Elements such as sample size and model complexity should be considered when evaluating fit. Table 1 provides a side-by-side comparison of the resulting parameter estimates and fit statistics; Figures 1 and 2 provide a graphic representation of the models tested.

To assess the factor structure of the GRMSAAW we examined five separate models: a unidimensional model, an uncorrelated factors model, a correlated factors model, a second-order model, and a bifactor models. Support for a unidimensional model would suggest that the model is best represented by a total scale score with no subfactors. Support for an uncorrelated factors model would suggest that the factors are largely independent. Support for a correlated factors model would suggest that the factors are related. Support for a second-order GRMS factor would suggest that the AS, AF, MI, and AUA subfactors represent facets of the higher order factor, GRMS. In the bifactor models, items for each scale are loaded onto both their respective subscale and the overall GRMS scale ( $g$ ). Support for this model would suggest that each subscale has both independent variance, and common variance that belongs to an underlying GRMS factor. When a bifactor model is the best representation of fit to the data, researchers can utilize bifactor indices to determine the proportion of variance accounted for by the subscales and the general factor, respectively.

Our first model was unidimensional where each of the 24 items loaded onto a single factor representing overall, gendered racial microaggressions towards Asian American women. The Chi-square index was statistically significant ( $\chi^2(209) = 444.451, p < .001$ ) indicating likely misfit. The CFI value of .81 indicated poor fit. In contrast, the RMSEA = 0.061, 90% CI(0.053, 0.069) and SRMR = 0.067 both fell within the ranges of acceptability. Our second model was an uncorrelated factors model. It demonstrated less than adequate fit to the data:  $\chi^2(209) = 461.102, p < 0.001, CFI = 0.791, RMSEA = 0.063, 90$ . Our third model was a single-order, correlated factors where each of the 22 items loaded onto one of four factors and the factors were free to correlate. The Chi-square index was not statistically significant ( $\chi^2(203) = 232.453, p = 0.076$ ) indicating reasonable fit. The CFI value of 0.972 exceeded the recommendation of .95. The RMSEA = 0.022 (90%CI[.000, 0.034]) was satisfactory. The SRMR value of 0.047 remained below the warning criteria of .10. Our fourth model represented a second order structure. Specifically, four first-order factors loaded onto a second factor model and demonstrated adequate fit to the data:  $\chi^2(205) = 234.741, p = 0.076, CFI = 0.975, RMSEA = 0.022, 90$ . The fifth model, a bifactor model regressed each item on its respective factor while simultaneously regressing each indicator onto an over-

all GRMS scale. This model had the best fit of those compared thus far:  $\chi^2(187) = 164.080, p = .885, CFI = 0.997, RMSEA = 0.009, 90$ .

As shown in our table of model comparisons, Chi-square difference tests between models showed statistically significant differences between the uncorrelated factors model and the correlated factors and second-order model, which do not differ from each other. Finally, there was a statistically significant difference between the second order factor model and the bifactor model ( $\chi^2(18) = 43.533, p < .001$ ). The CFI, RMSEA, SRMR, and AIC values favor the bifactor model; the BIC favored the second order model. Thus, all of the multidimensional models demonstrated adequate fit and are suitable for research and practice.

As estimates of model-based internal consistency associated with the bifactor model, we calculated omega hierarchical ( $\omega_h$ ) and omega hierarchical subscale ( $\omega_{h-ss}$ ).  $\omega_h$ , represents the proportion of total-score variance due to a single, general construct that influences all items, despite the multidimensional nature of the item set [Flora, 2020a,b]. Our  $\omega_h$  value of 0.71 indicated that 71% of the variance of the GRMS total scores are attributable to individual differences on the general factor.  $\omega_{h-ss}$  for the subscales ranged from 0.21 to 0.43. Taken together, the omega values from our bifactor model suggest a strong general factor with subscales that are meaningful and useful.

## 11.15 A Conversation with Dr. Keum

Doctoral student Jadvir Gill (Industrial-Organizational Psychology) and I were able to interview the first author (Brian TaeHyuk Keum, PhD) about the article used for the research vignette [Keum et al., 2018]. Here's a direct [link](#) to that interview.

Among other things, we asked:

- What challenges did you encounter in the research process and recruiting participants?
- Can you describe how the items (or subscales) captures intersectional microaggressions for the intended population?
- How would you like to see the scale used in future science, practice, and advocacy?
- How are you using this scale in your current and future research?

## 11.16 Practice Problems

In each of these lessons I provide suggestions for practice that allow you to select one or more problems that are graded in difficulty. In psychometrics, I strongly recommend that you have started with a dataset that has a minimum of three subscales and use it for all of the assignments in the OER. The suggestion for practice spans the prior chapter and this one. For this combination assignment, you should plan to:

- Prepare the data frame for CFA.

- Specify and run unidimensional, single order (with correlated factors), second-order, and bifactor models.
- Narrate the adequacy of fit with  $\chi^2$ , CFI, RMSEA, SRMR
  - Write a mini-results section for each
- Compare model fit with  $\chi^2\Delta$ , AIC, and BIC.
- Calculate the appropriate omega reliability estimates for the  $g$  and specific factors.
- Write an APA style results sections with table(s) and figures.

### 11.16.1 Problem #1: Play around with this simulation.

The least complex is to change the random seed in the research and rework the problem demonstrated in the lesson. The results *should* map onto the ones obtained in the lecture.

### 11.16.2 Problem #2: Use simulated data from other lessons.

The second option involves utilizing one of the simulated datasets available in this OER. The [last lesson](#) in the OER contains three simulations that could be used for all of the statistics-based practice suggestions. Especially if you started with one of these examples in an earlier lesson, I highly recommend you continue with that.

Alternatively, Lewis and Neville's [2015] Gendered Racial Microaggressions Scale for Black Women was used in the lessons for exploratory factor analysis and Conover et al.'s [2017] Ableist Microaggressions Scale is used in the lesson on invariance testing. Both of these would be suitable for the CFA homework assignments.

### 11.16.3 Problem #3: Try something entirely new.

As a third option, you are welcome to use data to which you have access and is suitable for CFA. This could include other simulated data, data found on an open access repository, data from the ReCentering Psych Stats survey described in the [Qualtrics lesson](#), or your own data (presuming you have permission to use it).

### 11.16.4 Grading Rubric

Using the lecture and workflow (chart) as a guide, please work through all the steps listed in the proposed assignment/grading rubric.

Assignment Component	Points Possible	Points Earned
1. Prepare data for CFA (items only df, reverse-scored)	5	_____
2. Specify, run, and plot a unidimensional model	5	_____
3. Narrate adequacy of fit with $\chi^2$ , CFI, RMSEA, SRMR (write a mini-results section)	5	_____

Assignment Component	Points Possible	Points Earned
4. Specify, run, and plot a single-order model with correlated factors	5	_____
5. Narrate adequacy of fit with $\chi^2$ , CFI, RMSEA, SRMR (write a mini-results section)	5	_____
6. Specify, run, and plot a second-order model	5	_____
7. Narrate adequacy of fit with $\chi^2$ , CFI, RMSEA, SRMR (write a mini-results section)	5	_____
8. Specify, run, and plot a bifactor model	5	_____
9. Narrate adequacy of fit with $\chi^2$ , CFI, RMSEA, SRMR (write a mini-results section)	5	_____
10. Compare model fit with $\chi^2\Delta$ , AIC, BIC	5	_____
11. Calculate omega hierarchical ( $\omega_h$ ) and omega-hierarchical-subscales ( $\omega_{h-ss}$ )	5	_____
12. APA style results with table(s) and figures	5	_____
13. Explanation to grader	5	_____
<b>Totals</b>	<b>60</b>	_____

## 11.17 Homeworked Example

### Screencast Link

For more information about the data used in this homeworked example, please refer to the description and codebook located at the end of the [introduction](#) in first volume of ReCentering Psych Stats.

As a brief review, this data is part of an IRB-approved study, with consent to use in teaching demonstrations and to be made available to the general public via the open science framework. Hence, it is appropriate to use in this context. You will notice there are student- and teacher- IDs. These numbers are not actual student and teacher IDs, rather they were further re-identified so that they could not be connected to actual people.

Because this is an actual dataset, if you wish to work the problem along with me, you will need to download the [ReC.rds](#) data file from the Worked\_Examples folder in the ReC\_Psychometrics project on the GitHub.

The course evaluation items can be divided into three subscales:

- **Valued by the student** includes the items: ValObjectives, IncrUnderstanding, IncrInterest
- **Traditional pedagogy** includes the items: ClearResponsibilities, EffectiveAnswers, Feedback, ClearOrganization, ClearPresentation
- **Socially responsive pedagogy** includes the items: InclusvClassrm, EquitableEval, Mult-Perspectives, DEIIntegration

In this Homeworked Example I will conduct all the analyses from the immediately prior (first order CFA models) and present CFA lessons. My hope is that the results will support my solution of three dimensions: valued-by-the-student, traditional pedagogy, socially responsive pedagogy. While the

repetition of the prior lesson's homeworked example is somewhat redundant, I am hopeful that this code will provide a fairly complete set of code for someone who is analyzing their own data from the beginning.

### 11.17.1 Prepare data for CFA (items only df, reverse-scored)

We can upload the data from the .rds file. The file should be in the same folder as the .rmd file. I've named the df object that holds the data "big."

```
big <- readRDS("ReC.rds")
```

For the demonstration of CFA models, I will create an items-only df.

```
library(tidyverse)
items <- big %>%
  dplyr::select(ValObjectives, IncrUnderstanding, IncrInterest, ClearResponsibilities,
    EffectiveAnswers, Feedback, ClearOrganization, ClearPresentation,
    MultPerspectives, InclusvClassrm, DEIintegration, EquitableEval)
```

Let's quickly check the structure. The variables should be numeric or integer.

```
str(items)
```

```
Classes 'data.table' and 'data.frame': 310 obs. of 12 variables:
 $ ValObjectives      : int  5 5 4 4 5 5 5 5 4 5 ...
 $ IncrUnderstanding   : int  2 3 4 3 4 4 5 2 4 5 ...
 $ IncrInterest        : int  5 3 4 2 4 3 5 3 2 5 ...
 $ ClearResponsibilities: int  5 5 4 4 5 4 5 4 4 5 ...
 $ EffectiveAnswers    : int  5 3 5 3 5 3 4 3 2 3 ...
 $ Feedback            : int  5 3 4 2 5 NA 5 4 4 5 ...
 $ ClearOrganization   : int  3 4 3 4 4 4 5 4 4 5 ...
 $ ClearPresentation   : int  4 4 4 2 5 3 4 4 4 5 ...
 $ MultPerspectives     : int  5 5 4 5 5 4 5 5 5 5 ...
 $ InclusvClassrm       : int  5 5 5 5 5 4 5 5 4 5 ...
 $ DEIintegration      : int  5 5 5 5 5 4 5 5 5 5 ...
 $ EquitableEval        : int  5 5 3 5 5 3 5 5 3 5 ...
 - attr(*, ".internal.selfref")=<externalptr>
```

### 11.17.2 Specify and run a unidimensional model

First we map the relations we want to analyze.

```
uniD <- "CourseEvals =~ ValObjectives + IncrUnderstanding + IncrInterest + ClearResponsibilities"
```

We analyze the relations by naming that object in our *lavaan* code.

```
set.seed(240311)
uniDfit <- lavaan:::cfa(uniD, data = items)
lavaan:::summary(uniDfit, fit.measures = TRUE, standardized = TRUE, rsquare = TRUE)
```

lavaan 0.6.17 ended normally after 32 iterations

Estimator	ML
Optimization method	NLMINB
Number of model parameters	24
	Used      Total
Number of observations	267      310

Model Test User Model:

Test statistic	344.973
Degrees of freedom	54
P-value (Chi-square)	0.000

Model Test Baseline Model:

Test statistic	1940.157
Degrees of freedom	66
P-value	0.000

User Model versus Baseline Model:

Comparative Fit Index (CFI)	0.845
Tucker-Lewis Index (TLI)	0.810

Loglikelihood and Information Criteria:

Loglikelihood user model (H0)	-3038.064
Loglikelihood unrestricted model (H1)	-2865.578
Akaike (AIC)	6124.129
Bayesian (BIC)	6210.223
Sample-size adjusted Bayesian (SABIC)	6134.129

Root Mean Square Error of Approximation:

RMSEA	0.142
90 Percent confidence interval - lower	0.128
90 Percent confidence interval - upper	0.157
P-value H_0: RMSEA <= 0.050	0.000
P-value H_0: RMSEA >= 0.080	1.000

Standardized Root Mean Square Residual:

SRMR	0.074
------	-------

Parameter Estimates:

Standard errors	Standard
Information	Expected
Information saturated (h1) model	Structured

Latent Variables:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
CourseEvals =~						
ValObjectives	1.000				0.309	0.515
IncrUnderstdng	1.715	0.223	7.684	0.000	0.530	0.642
IncrInterest	2.142	0.269	7.971	0.000	0.662	0.685
ClearRspnsblts	2.065	0.239	8.652	0.000	0.638	0.808
EffectivAnswrs	2.105	0.244	8.617	0.000	0.650	0.800
Feedback	2.143	0.259	8.285	0.000	0.662	0.738
ClearOrganiztn	2.678	0.314	8.516	0.000	0.828	0.780
ClearPresenttn	2.521	0.285	8.832	0.000	0.779	0.846
MultPerspectvs	2.067	0.246	8.392	0.000	0.639	0.757
InclusvClassrm	1.246	0.170	7.324	0.000	0.385	0.592
DEIIntegration	1.015	0.174	5.820	0.000	0.314	0.424
EquitableEval	1.435	0.179	8.027	0.000	0.443	0.694

Variances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
.ValObjectives	0.265	0.024	11.254	0.000	0.265	0.735
.IncrUnderstdng	0.401	0.037	10.970	0.000	0.401	0.588
.IncrInterest	0.494	0.046	10.815	0.000	0.494	0.530
.ClearRspnsblts	0.217	0.022	9.983	0.000	0.217	0.348
.EffectivAnswrs	0.237	0.024	10.060	0.000	0.237	0.359
.Feedback	0.367	0.035	10.555	0.000	0.367	0.455
.ClearOrganiztn	0.439	0.043	10.250	0.000	0.439	0.391
.ClearPresenttn	0.242	0.026	9.446	0.000	0.242	0.285
.MultPerspectvs	0.304	0.029	10.431	0.000	0.304	0.427
.InclusvClassrm	0.275	0.025	11.104	0.000	0.275	0.649
.DEIIntegration	0.449	0.040	11.372	0.000	0.449	0.820
.EquitableEval	0.211	0.020	10.777	0.000	0.211	0.518
CourseEvals	0.096	0.022	4.381	0.000	1.000	1.000

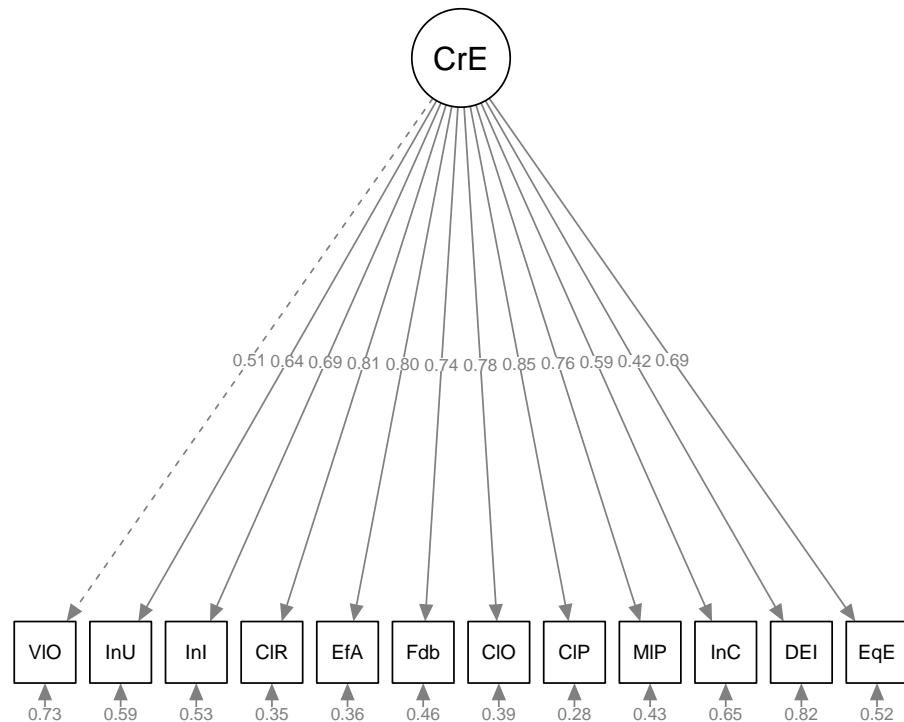
R-Square:

	Estimate
ValObjectives	0.265
IncrUnderstdng	0.412
IncrInterest	0.470
ClearRspnsblts	0.652

EffectivAnsrs	0.641
Feedback	0.545
ClearOrganiztn	0.609
ClearPresenttn	0.715
MultPerspectvs	0.573
InclusvClassrm	0.351
DEIintegration	0.180
EquitableEval	0.482

Let's plot the results to see if the figure resembles what we intended to specify.

```
semPlot::semPaths(uniDfit, layout = "tree", style = "lisrel", what = "col",
  whatLabels = "stand")
```



### 11.17.3 Narrate adequacy of fit with $\chi^2$ , CFI, RMSEA, SRMR (write a mini-results section)

**Model testing.** To evaluate the models we, we used confirmatory factor analysis (CFA) in the R package, lavaan (v.0-6.17) with maximum likelihood estimation. Our sample size was 267 We selected fit criteria for their capacity to assess different aspects of the statistical analysis. As is common among SEM researchers, we reported the Chi-square goodness of fit (2). This evaluates the discrepancy between the unrestricted sample matrix and the restricted covariance matrix. Although the associated  $p$ -value indicates adequate fit when the value is non-significant, it is widely recognized that a large sample size can result in a statistically significant  $p$  value (Byrne, 2016b). The comparative fit index (CFI) is an incremental index, comparing the hypothesized model to the independent/baseline model. Adequate fit is determined when CFI values are at least .90 and perhaps higher than .95 [Kline, 2016]. The root mean square error of approximation (RMSEA) takes into account the error of approximation in the population and expresses it per degree of freedom. As such, the fit indicator considers the complexity of the model. Ideal values are equal to or less than .05, values less than .08 represent reasonable fit, and values between .08 and .10 represent mediocre fit. The standardized root mean residual (SRMR) is a standardized measure of the mean absolute covariance residual – the overall difference between the observed and predicted correlations. Values greater than .10 may indicate poor fit and inspection of residuals is then advised. Kline (2016) advised researchers to be cautious when using these criteria as strict cut-offs. Elements such as sample size and model complexity should be considered when evaluating fit.

Our first model was unidimensional where each of the 12 items loaded onto a single factor representing overall course evaluations. The Chi-square index was statistically significant ( $\chi^2(54) = 344.97, p < .001$ ) indicating likely misfit. The CFI value of .85 indicated poor fit. The RMSEA = .14 (90% CI [.13, .16]) suggested serious problems. The SRMR value of .07 was below the warning criteria of .10. The AIC and BIC values were 6124.13 and 6134.13, respectively, and will become useful in comparing subsequent models.

### 11.17.4 Specify and run a single-order model with correlated factors

First we map the relations we want to analyze.

```
corrF <- "TradPed =~ ClearResponsibilities + EffectiveAnswers + Feedback + ClearOrganization +
          Valued =~ ValObjectives + IncrUnderstanding + IncrInterest
          SCRPed =~ MultPerspectives + InclusvClassrm + DEIntegration + EquitableEval

          TradPed~~Valued
          TradPed~~SCRPed
          Valued~~SCRPed
          "
```

Next we run the analysis.

```
set.seed(240311)
corrF_fit <- lavaan:::cfa(corrF, data = items)
lavaan:::summary(corrF_fit, fit.measures = TRUE, standardized = TRUE, rsquare = TRUE)
```

lavaan 0.6.17 ended normally after 42 iterations

Estimator	ML
Optimization method	NLMINB
Number of model parameters	27
	Used      Total
Number of observations	267      310

Model Test User Model:

Test statistic	224.795
Degrees of freedom	51
P-value (Chi-square)	0.000

Model Test Baseline Model:

Test statistic	1940.157
Degrees of freedom	66
P-value	0.000

User Model versus Baseline Model:

Comparative Fit Index (CFI)	0.907
Tucker-Lewis Index (TLI)	0.880

Loglikelihood and Information Criteria:

Loglikelihood user model (H0)	-2977.975
Loglikelihood unrestricted model (H1)	-2865.578
Akaike (AIC)	6009.951
Bayesian (BIC)	6106.807
Sample-size adjusted Bayesian (SABIC)	6021.201

Root Mean Square Error of Approximation:

RMSEA	0.113
90 Percent confidence interval - lower	0.098
90 Percent confidence interval - upper	0.128
P-value H_0: RMSEA <= 0.050	0.000
P-value H_0: RMSEA >= 0.080	1.000

Standardized Root Mean Square Residual:

SRMR	0.061
------	-------

Parameter Estimates:

Standard errors	Standard
Information	Expected
Information saturated (h1) model	Structured

Latent Variables:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
TradPed ==						
ClearRspnsblts	1.000				0.652	0.826
EffectivAnswrs	1.015	0.065	15.606	0.000	0.662	0.815
Feedback	1.010	0.075	13.481	0.000	0.659	0.735
ClearOrganiztn	1.295	0.086	15.106	0.000	0.845	0.797
ClearPresenttn	1.204	0.072	16.680	0.000	0.785	0.853
Valued ==						
ValObjectives	1.000				0.334	0.557
IncrUndrstndng	1.942	0.223	8.717	0.000	0.649	0.786
IncrInterest	2.438	0.273	8.932	0.000	0.815	0.844
SCRPed ==						
MultPerspectvs	1.000				0.713	0.846
InclusvClassrm	0.622	0.053	11.672	0.000	0.444	0.682
DEIIntegration	0.589	0.063	9.365	0.000	0.420	0.567
EquitableEval	0.642	0.052	12.410	0.000	0.458	0.717

Covariances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
TradPed ~~						
Valued	0.171	0.026	6.640	0.000	0.785	0.785
SCRPed	0.391	0.045	8.677	0.000	0.841	0.841
Valued ~~						
SCRPed	0.164	0.026	6.254	0.000	0.688	0.688

Variances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
.ClearRspnsblts	0.199	0.021	9.456	0.000	0.199	0.319
.EffectivAnswrs	0.222	0.023	9.618	0.000	0.222	0.336
.Feedback	0.371	0.036	10.415	0.000	0.371	0.460
.ClearOrganiztn	0.410	0.042	9.855	0.000	0.410	0.365
.ClearPresenttn	0.232	0.026	8.939	0.000	0.232	0.273
.ValObjectives	0.248	0.023	10.650	0.000	0.248	0.690
.IncrUndrstndng	0.260	0.032	8.041	0.000	0.260	0.382
.IncrInterest	0.268	0.043	6.308	0.000	0.268	0.288
.MultPerspectvs	0.203	0.029	7.052	0.000	0.203	0.285
.InclusvClassrm	0.226	0.023	10.028	0.000	0.226	0.534

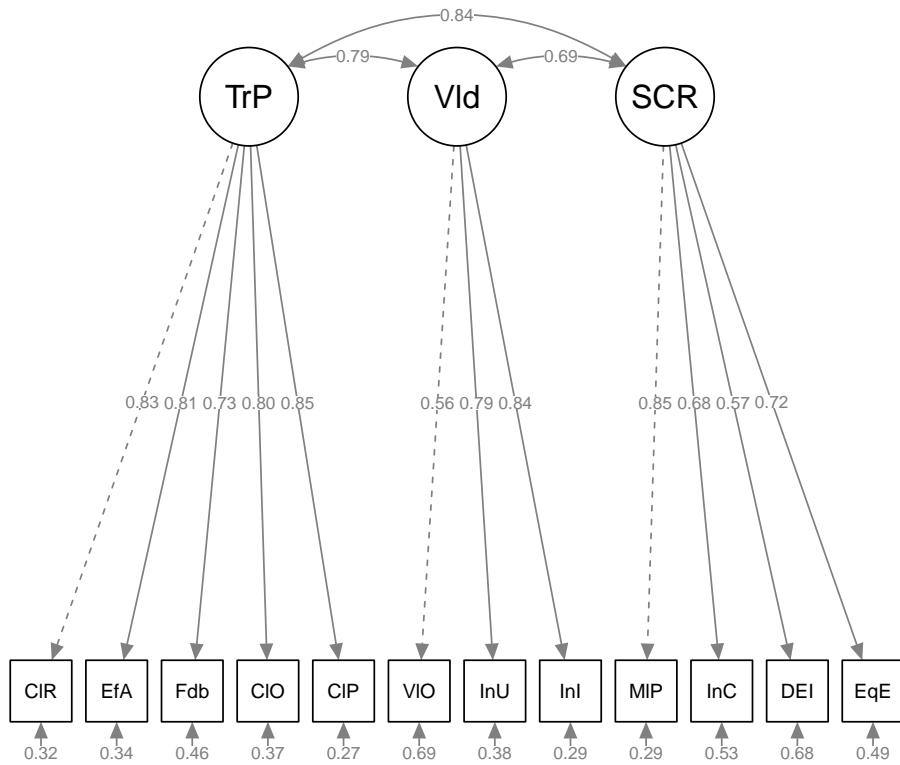
.DEIintegration	0.371	0.035	10.734	0.000	0.371	0.678
.EquitableEval	0.198	0.020	9.685	0.000	0.198	0.486
TradPed	0.426	0.053	8.085	0.000	1.000	1.000
Valued	0.112	0.024	4.595	0.000	1.000	1.000
SCRPed	0.509	0.063	8.039	0.000	1.000	1.000

R-Square:

	Estimate
ClearRspnsblts	0.681
EffectivAnswnrs	0.664
Feedback	0.540
ClearOrganiztn	0.635
ClearPresenttn	0.727
ValObjectives	0.310
IncrUndrstndng	0.618
IncrInterest	0.712
MultPerspectvs	0.715
InclusvClassrm	0.466
DEIintegration	0.322
EquitableEval	0.514

Plotting the results. Does it look like what we intended to specify?

```
semPlot::semPaths(corrF_fit, layout = "tree", style = "lisrel", what = "col",
whatLabels = "stand")
```



Code for saving the results as a .csv file.

```
corrFitStats <- tidySEM::table_fit(corrF_fit)
corrF_paramEsts <- tidySEM::table_results(corrF_fit, digits = 3, columns = NULL)
corrFCorrs <- tidySEM::table_corrs(corrF_fit, digits = 3)
# to see each of the tables, remove the hashtag corrFitStats
# corrF_paramEsts corrFCorrs
```

Next, I export them.

```
write.csv(corrFitStats, file = "corrFitStats.csv")
write.csv(corrF_paramEsts, file = "corrF_paramEsts.csv")
write.csv(corrFCorrs, file = "corrFCorrs.csv")
```

### 11.17.5 Narrate adequacy of fit with $\chi^2$ , CFI, RMSEA, SRMR (write a mini-results section)

Our second model was a single-order, multidimensional model where each of the 12 items loaded onto one of four factors. Standardized pattern coefficients ranged between .74 and .85 on the TradPed factor, between .56 and .84 on the Valued factor, and between .57 and .85 on the SCRPed factor. The Chi-square index was statistically significant ( $\chi^2(51) = 224.795, p < 0.001$ ) indicating some degree of misfit. The CFI value of .91 fell below the recommendation of .95. The RMSEA = .113 (90% CI [.098, .128]) was higher than recommended. The SRMR value of .061 remained below the warning criteria of .10. The AIC and BIC values were 6009.95 and 6021.20, respectively.

### 11.17.6 Specify and run a second-order model

```
secondM <- "TradPed =~ ClearResponsibilities + EffectiveAnswers + Feedback + ClearOrganization
             Valued =~ ValObjectives + IncrUnderstanding + IncrInterest
             SCRPed =~ MultPerspectives + InclusvClassrm + DEIIntegration + EquitableEval
             Evals =~ TradPed + Valued + SCRPed"
```

```
set.seed(240311)
secondF <- lavaan:::cfa(secondM, data = items)
lavaan:::summary(secondF, fit.measures = TRUE, standardized = TRUE, rsquare = TRUE)
```

lavaan 0.6.17 ended normally after 37 iterations

Estimator	ML
Optimization method	NLMINB
Number of model parameters	27
	Used      Total
Number of observations	267      310

Model Test User Model:

Test statistic	224.795
Degrees of freedom	51
P-value (Chi-square)	0.000

Model Test Baseline Model:

Test statistic	1940.157
Degrees of freedom	66
P-value	0.000

User Model versus Baseline Model:

Comparative Fit Index (CFI)	0.907
Tucker-Lewis Index (TLI)	0.880

Loglikelihood and Information Criteria:

Loglikelihood user model (H0)	-2977.975
Loglikelihood unrestricted model (H1)	-2865.578
Akaike (AIC)	6009.951
Bayesian (BIC)	6106.807
Sample-size adjusted Bayesian (SABIC)	6021.201

Root Mean Square Error of Approximation:

RMSEA	0.113
90 Percent confidence interval - lower	0.098
90 Percent confidence interval - upper	0.128
P-value H_0: RMSEA <= 0.050	0.000
P-value H_0: RMSEA >= 0.080	1.000

Standardized Root Mean Square Residual:

SRMR	0.061
------	-------

Parameter Estimates:

Standard errors	Standard
Information	Expected
Information saturated (h1) model	Structured

Latent Variables:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
TradPed =~						
ClearRspnsblts	1.000				0.652	0.826
EffectivAnswnrs	1.015	0.065	15.606	0.000	0.662	0.815
Feedback	1.010	0.075	13.481	0.000	0.659	0.735
ClearOrganiztn	1.295	0.086	15.106	0.000	0.845	0.797
ClearPresenttn	1.204	0.072	16.680	0.000	0.785	0.853
Valued =~						
ValObjectives	1.000				0.334	0.557
IncrUndrstndng	1.942	0.223	8.717	0.000	0.649	0.786
IncrInterest	2.438	0.273	8.932	0.000	0.815	0.844
SCRPed =~						
MultPerspectvs	1.000				0.713	0.846
InclusvClassrm	0.622	0.053	11.672	0.000	0.444	0.682
DEIIntegration	0.589	0.063	9.365	0.000	0.420	0.567
EquitableEval	0.642	0.052	12.410	0.000	0.458	0.717
Evals =~						

TradPed	1.000				0.980	0.980
Valued	0.419	0.055	7.630	0.000	0.801	0.801
SCRPed	0.958	0.086	11.128	0.000	0.859	0.859

Variances:

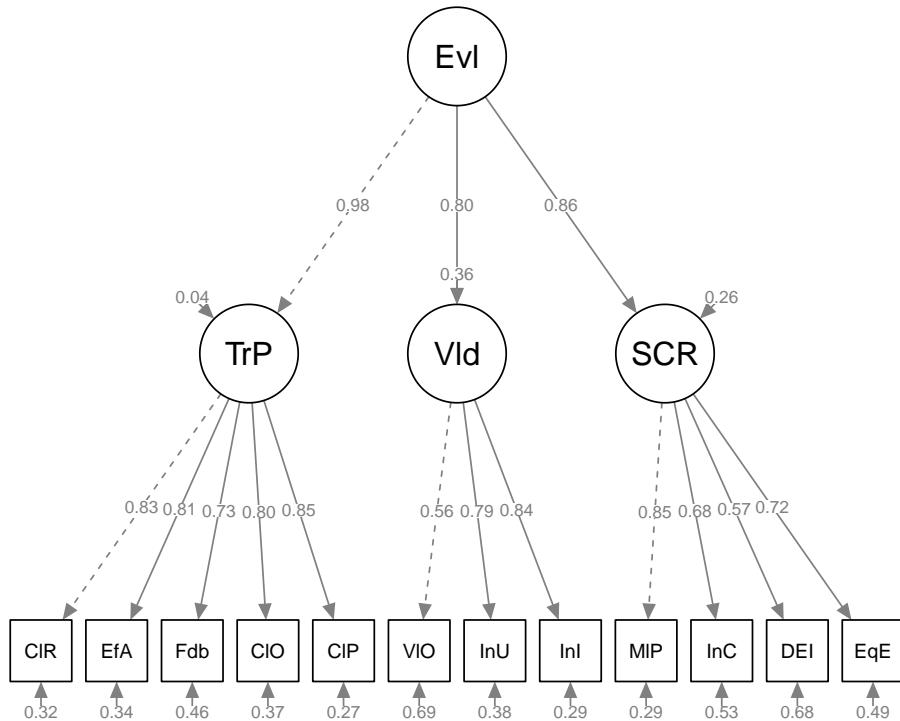
	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
.ClearRspnsblts	0.199	0.021	9.456	0.000	0.199	0.319
.EffectivAnswrs	0.222	0.023	9.618	0.000	0.222	0.336
.Feedback	0.371	0.036	10.415	0.000	0.371	0.460
.ClearOrganiztn	0.410	0.042	9.855	0.000	0.410	0.365
.ClearPresenttn	0.232	0.026	8.939	0.000	0.232	0.273
.ValObjectives	0.248	0.023	10.650	0.000	0.248	0.690
.IncrUndrstndng	0.260	0.032	8.041	0.000	0.260	0.382
.IncrInterest	0.268	0.043	6.308	0.000	0.268	0.288
.MultPerspectvs	0.203	0.029	7.052	0.000	0.203	0.285
.InclusvClassrm	0.226	0.023	10.028	0.000	0.226	0.534
.DEIintegration	0.371	0.035	10.734	0.000	0.371	0.678
.EquitableEval	0.198	0.020	9.685	0.000	0.198	0.486
.TradPed	0.017	0.023	0.728	0.467	0.039	0.039
.Valued	0.040	0.010	3.895	0.000	0.358	0.358
.SCRPed	0.134	0.032	4.219	0.000	0.263	0.263
Evals	0.409	0.056	7.281	0.000	1.000	1.000

R-Square:

	Estimate
ClearRspnsblts	0.681
EffectivAnswrs	0.664
Feedback	0.540
ClearOrganiztn	0.635
ClearPresenttn	0.727
ValObjectives	0.310
IncrUndrstndng	0.618
IncrInterest	0.712
MultPerspectvs	0.715
InclusvClassrm	0.466
DEIintegration	0.322
EquitableEval	0.514
TradPed	0.961
Valued	0.642
SCRPed	0.737

As we plot this model we expect to see a “second level” factor predicting each of the “first order” factors. The indicator was set on GRM → AS. Each of the four factors predicts only the items associated with their factor with one item for each factor (the first on the left) specified as the indicator variable.

```
semPlot::semPaths(secondF, layout = "tree", style = "lisrel", what = "col",
  whatLabels = "stand")
```



Code for saving the results as a .csv file.

```
secondFFitStats <- tidySEM::table_fit(secondF)
secondF_paramEsts <- tidySEM::table_results(secondF, digits = 3, columns = NULL)
# In a second order structure there are no correlations to request
# secondFCorrs <- tidySEM::table_cors(secondF, digits=3)

# to see each of the tables, remove the hashtag secondFFitStats
# secondF_paramEsts
```

Next, I export them.

```
write.csv(secondFFitStats, file = "secondFFitStats.csv")
write.csv(secondF_paramEsts, file = "secondF_paramEsts.csv")
```

### 11.17.7 Narrate adequacy of fit with $\chi^2$ , CFI, RMSEA, SRMR (write a mini-results section)

Our next model represented a second order structure where three first-order factors loaded onto a second factor model. Across a variety of indices, model fit improved:  $\chi^2(51) = 224.80, p < .001, CFI = .907, RMSEA = .113, 90$ . Factor loadings ranged from .75 to .85 for the TradPed scale, .56 to .84 for the Valued-by-Me scale, .57 to .85 for the SCRPed scale, and .80 to .98 for the total scale.

### 11.17.8 Specify and run a bifactor model

```
bifacM <- "Evals =~ ClearResponsibilities + EffectiveAnswers + Feedback + ClearOrganization +
TradPed =~ ClearResponsibilities + EffectiveAnswers + Feedback + ClearOrganization
Valued =~ ValObjectives + IncrUnderstanding + IncrInterest
SCRPed =~ MultPerspectives + InclusvClassrm + DEIintegration + EquitableEval

#fixes the relations between g and each of the factors to 0.0
Evals ~~ 0*TradPed
Evals ~~ 0*Valued
Evals ~~ 0*SCRPed

#fixes the relations (covariances) between each of the factors to 0.0
TradPed ~~ 0*Valued
TradPed ~~ 0*SCRPed
Valued ~~ 0*SCRPed

"
```

```
set.seed(240311)
bifacF <- lavaan:::cfa(bifacM, data = items)
```

Warning in lavaan:::lavaan(model = bifacM, data = items, int.ov.free = TRUE, : lavaan WARNING:  
the optimizer (NLMINB) claimed the model converged, but not all  
elements of the gradient are (near) zero; the optimizer may not  
have found a local solution use check.gradient = FALSE to skip  
this check.

```
lavaan:::summary(bifacF, fit.measures = TRUE, standardized = TRUE, rsquare = TRUE)
```

Warning in lav\_object\_summary(object = object, header = header, fit.measures = fit.measures, :

lavaan 0.6.17 did NOT end normally after 1256 iterations  
\*\* WARNING \*\* Estimates below are most likely unreliable

Estimator	ML
Optimization method	NLMINB
Number of model parameters	36
Number of observations	Used 267      Total 310

## Parameter Estimates:

Standard errors	Standard
Information	Expected
Information saturated (h1) model	Structured

## Latent Variables:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
Evals =~						
ClearRspnsblts	1.000				0.665	0.842
EffectivAnswrs	0.977	NA			0.650	0.800
Feedback	1.032	NA			0.686	0.765
ClearOrganiztn	1.251	NA			0.832	0.785
ClearPresenttn	1.227	NA			0.816	0.886
ValObjectives	0.436	NA			0.290	0.484
IncrUndrstndng	0.757	NA			0.504	0.610
IncrInterest	0.923	NA			0.614	0.636
MultPerspectvs	0.925	NA			0.615	0.730
InclusvClassrm	0.531	NA			0.353	0.543
DEIintegration	0.387	NA			0.258	0.348
EquitableEval	0.644	NA			0.429	0.671
TradPed =~						
ClearRspnsblts	1.000				0.018	0.023
EffectivAnswrs	-0.115	NA			-0.002	-0.003
Feedback	2.260	NA			0.040	0.045
ClearOrganiztn	-0.130	NA			-0.002	-0.002
ClearPresenttn	-146.233	NA			-2.614	-2.838
Valued =~						
ValObjectives	1.000				0.145	0.241
IncrUndrstndng	2.548	NA			0.368	0.446
IncrInterest	4.310	NA			0.623	0.645
SCRPed =~						
MultPerspectvs	1.000				0.267	0.316
InclusvClassrm	1.302	NA			0.347	0.534
DEIintegration	2.188	NA			0.584	0.788
EquitableEval	0.415	NA			0.111	0.173

## Covariances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
Evals ~~						
TradPed	0.000				0.000	0.000
Valued	0.000				0.000	0.000
SCRPed	0.000				0.000	0.000
TradPed ~~						
Valued	0.000				0.000	0.000
SCRPed	0.000				0.000	0.000
Valued ~~						
SCRPed	0.000				0.000	0.000

## Variances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
.ClearRspnsblts	0.182	NA			0.182	0.291
.EffectivAnswrs	0.238	NA			0.238	0.361
.Feedback	0.332	NA			0.332	0.413
.ClearOrganiztn	0.432	NA			0.432	0.384
.ClearPresenttn	-6.651	NA			-6.651	-7.837
.ValObjectives	0.255	NA			0.255	0.708
.IncrUndrstndng	0.292	NA			0.292	0.429
.IncrInterest	0.167	NA			0.167	0.179
.MultPerspectvs	0.262	NA			0.262	0.368
.InclusvClassrm	0.178	NA			0.178	0.421
.DEIIntegration	0.141	NA			0.141	0.257
.EquitableEval	0.212	NA			0.212	0.519
Evals	0.443	NA			1.000	1.000
TradPed	0.000	NA			1.000	1.000
Valued	0.021	NA			1.000	1.000
SCRPed	0.071	NA			1.000	1.000

## R-Square:

	Estimate
ClearRspnsblts	0.709
EffectivAnswrs	0.639
Feedback	0.587
ClearOrganiztn	0.616
ClearPresenttn	NA
ValObjectives	0.292
IncrUndrstndng	0.571
IncrInterest	0.821
MultPerspectvs	0.632
InclusvClassrm	0.579
DEIIntegration	0.743
EquitableEval	0.481

Unfortunately, it's all-too-common for complex models to fail to converge. When this happens it's a slow, tedious process to find a fix.

In the ReCentering Psych Stats example I was able to fix it by adding a `check.gradient=FALSE` statement. So, let's try that

```
set.seed(240311)
bifacF <- lavaan::cfa(bifacM, data = items, check.gradient = FALSE)
```

Warning in lav\_model\_vcov(lavmodel = lavmodel, lavsamplestats = lavsamplestats, : lavaan WARNING:  
 Could not compute standard errors! The information matrix could  
 not be inverted. This may be a symptom that the model is not  
 identified.

Warning in lav\_object\_post\_check(object): lavaan WARNING: some estimated ov  
 variances are negative

```
lavaan::summary(bifacF, fit.measures = TRUE, standardized = TRUE, rsquare = TRUE)
```

lavaan 0.6.17 ended normally after 1256 iterations

Estimator	ML
Optimization method	NLMINB
Number of model parameters	36
Number of observations	Used 267      Total 310

Model Test User Model:

Test statistic	121.965
Degrees of freedom	42
P-value (Chi-square)	0.000

Model Test Baseline Model:

Test statistic	1940.157
Degrees of freedom	66
P-value	0.000

User Model versus Baseline Model:

Comparative Fit Index (CFI)	0.957
Tucker-Lewis Index (TLI)	0.933

Loglikelihood and Information Criteria:

Loglikelihood user model (H0)	-2926.561
Loglikelihood unrestricted model (H1)	-2865.578

Akaike (AIC)	5925.121
Bayesian (BIC)	6054.262
Sample-size adjusted Bayesian (SABIC)	5940.121

Root Mean Square Error of Approximation:

RMSEA	0.084
90 Percent confidence interval - lower	0.067
90 Percent confidence interval - upper	0.102
P-value H_0: RMSEA <= 0.050	0.001
P-value H_0: RMSEA >= 0.080	0.679

Standardized Root Mean Square Residual:

SRMR	0.041
------	-------

Parameter Estimates:

Standard errors	Standard
Information	Expected
Information saturated (h1) model	Structured

Latent Variables:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
<b>Evals =~</b>						
ClearRspnsblts	1.000				0.665	0.842
EffectivAnswrs	0.977	NA			0.650	0.800
Feedback	1.032	NA			0.686	0.765
ClearOrganiztn	1.251	NA			0.832	0.785
ClearPresenttn	1.227	NA			0.816	0.886
ValObjectives	0.436	NA			0.290	0.484
IncrUndrstndng	0.757	NA			0.504	0.610
IncrInterest	0.923	NA			0.614	0.636
MultPerspectvs	0.925	NA			0.615	0.730
InclusvClassrm	0.531	NA			0.353	0.543
DEIIntegration	0.387	NA			0.258	0.348
EquitableEval	0.644	NA			0.429	0.671
<b>TradPed =~</b>						
ClearRspnsblts	1.000				0.018	0.023
EffectivAnswrs	-0.115	NA			-0.002	-0.003
Feedback	2.260	NA			0.040	0.045
ClearOrganiztn	-0.130	NA			-0.002	-0.002
ClearPresenttn	-146.233	NA			-2.614	-2.838
<b>Valued =~</b>						
ValObjectives	1.000				0.145	0.241
IncrUndrstndng	2.548	NA			0.368	0.446
IncrInterest	4.310	NA			0.623	0.645

SCRPed =~						
MultPerspectvs	1.000			0.267	0.316	
InclusvClassrm	1.302	NA		0.347	0.534	
DEIintegration	2.188	NA		0.584	0.788	
EquitableEval	0.415	NA		0.111	0.173	
<b>Covariances:</b>						
	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
Evals ~~						
TradPed	0.000				0.000	0.000
Valued	0.000				0.000	0.000
SCRPed	0.000				0.000	0.000
TradPed ~~						
Valued	0.000				0.000	0.000
SCRPed	0.000				0.000	0.000
Valued ~~						
SCRPed	0.000				0.000	0.000
<b>Variances:</b>						
	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
.ClearRspnsblts	0.182	NA			0.182	0.291
.EffectivAnswrs	0.238	NA			0.238	0.361
.Feedback	0.332	NA			0.332	0.413
.ClearOrganiztn	0.432	NA			0.432	0.384
.ClearPresenttn	-6.651	NA			-6.651	-7.837
.ValObjectives	0.255	NA			0.255	0.708
.IncrUndrstndng	0.292	NA			0.292	0.429
.IncrInterest	0.167	NA			0.167	0.179
.MultPerspectvs	0.262	NA			0.262	0.368
.InclusvClassrm	0.178	NA			0.178	0.421
.DEIintegration	0.141	NA			0.141	0.257
.EquitableEval	0.212	NA			0.212	0.519
Evals	0.443	NA			1.000	1.000
TradPed	0.000	NA			1.000	1.000
Valued	0.021	NA			1.000	1.000
SCRPed	0.071	NA			1.000	1.000
<b>R-Square:</b>						
	Estimate					
ClearRspnsblts	0.709					
EffectivAnswrs	0.639					
Feedback	0.587					
ClearOrganiztn	0.616					
ClearPresenttn	NA					
ValObjectives	0.292					
IncrUndrstndng	0.571					
IncrInterest	0.821					
MultPerspectvs	0.632					

InclusvClassrm	0.579
DEIintegration	0.743
EquitableEval	0.481

That did not work.

I found an article on nonconvergence in bifactor models: <https://stackoverflow.com/questions/68837355/trouble-converging-bifactor-model-using-lavaan>

I took out the check.gradient command and swapped in the “std.lv=TRUE” command.

```
set.seed(240311)
bifacF <- lavaan:::cfa(bifacM, data = items, std.lv = TRUE)
```

Warning in lav\_object\_post\_check(object): lavaan WARNING: some estimated covariances are negative

```
lavaan:::summary(bifacF, fit.measures = TRUE, standardized = TRUE, rsquare = TRUE)
```

lavaan 0.6.17 ended normally after 521 iterations

Estimator	ML
Optimization method	NLMINB
Number of model parameters	36
Number of observations	Used 267      Total 310

Model Test User Model:

Test statistic	121.965
Degrees of freedom	42
P-value (Chi-square)	0.000

Model Test Baseline Model:

Test statistic	1940.157
Degrees of freedom	66
P-value	0.000

User Model versus Baseline Model:

Comparative Fit Index (CFI)	0.957
Tucker-Lewis Index (TLI)	0.933

Loglikelihood and Information Criteria:

Loglikelihood user model (H0)	-2926.561
Loglikelihood unrestricted model (H1)	-2865.578

Akaike (AIC)	5925.121
Bayesian (BIC)	6054.262
Sample-size adjusted Bayesian (SABIC)	5940.121

Root Mean Square Error of Approximation:

RMSEA	0.084
90 Percent confidence interval - lower	0.067
90 Percent confidence interval - upper	0.102
P-value H_0: RMSEA <= 0.050	0.001
P-value H_0: RMSEA >= 0.080	0.679

Standardized Root Mean Square Residual:

SRMR	0.041
------	-------

Parameter Estimates:

Standard errors	Standard
Information	Expected
Information saturated (h1) model	Structured

Latent Variables:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
Evals =~						
ClearRspnsblts	0.665	0.040	16.440	0.000	0.665	0.842
EffectivAnswrs	0.650	0.043	15.284	0.000	0.650	0.800
Feedback	0.686	0.048	14.204	0.000	0.686	0.765
ClearOrganiztn	0.832	0.056	14.873	0.000	0.832	0.785
ClearPresenttn	0.816	0.048	17.021	0.000	0.816	0.886
ValObjectives	0.290	0.035	8.203	0.000	0.290	0.484
IncrUndrstndng	0.504	0.047	10.784	0.000	0.504	0.610
IncrInterest	0.614	0.054	11.355	0.000	0.614	0.636
MultPerspectvs	0.615	0.045	13.572	0.000	0.615	0.730
InclusvClassrm	0.353	0.038	9.350	0.000	0.353	0.543
DEIIntegration	0.258	0.045	5.718	0.000	0.258	0.348
EquitableEval	0.429	0.035	12.164	0.000	0.429	0.671
TradPed =~						
ClearRspnsblts	-0.018	0.234	-0.076	0.940	-0.018	-0.022
EffectivAnswrs	0.002	0.028	0.073	0.942	0.002	0.002
Feedback	-0.040	0.529	-0.076	0.940	-0.040	-0.045
ClearOrganiztn	0.002	0.032	0.072	0.943	0.002	0.002
ClearPresenttn	2.637	34.860	0.076	0.940	2.637	2.863
Valued =~						
ValObjectives	0.145	0.042	3.472	0.001	0.145	0.241

IncrUndrstndng	0.368	0.076	4.878	0.000	0.368	0.446
IncrInterest	0.623	0.116	5.385	0.000	0.623	0.645
SCRPed =~						
MultPerspectvs	0.267	0.042	6.327	0.000	0.267	0.316
InclusvClassrm	0.347	0.042	8.251	0.000	0.347	0.534
DEIintegration	0.584	0.059	9.967	0.000	0.584	0.788
EquitableEval	0.111	0.034	3.224	0.001	0.111	0.173

## Covariances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
Evals ~~						
TradPed	0.000				0.000	0.000
Valued	0.000				0.000	0.000
SCRPed	0.000				0.000	0.000
TradPed ~~						
Valued	0.000				0.000	0.000
SCRPed	0.000				0.000	0.000
Valued ~~						
SCRPed	0.000				0.000	0.000

## Variances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
.ClearRspnsblts	0.182	0.021	8.669	0.000	0.182	0.291
.EffectivAnswrs	0.238	0.025	9.441	0.000	0.238	0.361
.Feedback	0.333	0.047	7.050	0.000	0.333	0.413
.ClearOrganiztn	0.432	0.045	9.663	0.000	0.432	0.384
.ClearPresenttn	-6.774	183.878	-0.037	0.971	-6.774	-7.982
.ValObjectives	0.255	0.023	10.940	0.000	0.255	0.708
.IncrUndrstndng	0.292	0.054	5.440	0.000	0.292	0.429
.IncrInterest	0.167	0.137	1.221	0.222	0.167	0.179
.MultPerspectvs	0.262	0.026	9.965	0.000	0.262	0.368
.InclusvClassrm	0.178	0.024	7.298	0.000	0.178	0.421
.DEIintegration	0.141	0.056	2.496	0.013	0.141	0.257
.EquitableEval	0.212	0.019	10.967	0.000	0.212	0.519
Evals	1.000				1.000	1.000
TradPed	1.000				1.000	1.000
Valued	1.000				1.000	1.000
SCRPed	1.000				1.000	1.000

## R-Square:

	Estimate
ClearRspnsblts	0.709
EffectivAnswrs	0.639
Feedback	0.587
ClearOrganiztn	0.616
ClearPresenttn	NA
ValObjectives	0.292
IncrUndrstndng	0.571

IncrInterest	0.821
MultPerspectvs	0.632
InclusvClassrm	0.579
DEIIntegration	0.743
EquitableEval	0.481

Great! It ran, but now I have negative variances (i.e., a “Heywood case”) so I need to fix that. In the “variances” you can see the negative values on: .ClearPresenttn

Here's an article about Heywood cases (i.e., where there are negative variances): [https://s3.amazonaws.com/assets.datacamp.com/production/course\\_6419/slides/chapter3.pdf](https://s3.amazonaws.com/assets.datacamp.com/production/course_6419/slides/chapter3.pdf)

```
items <- na.omit(items)
var(items$ClearPresentation)
```

```
[1] 0.8518206
```

I will add this statement:

```
ClearPresentation ~ ~ 0.852 * ClearPresentation
```

```
bifacM <- "Evals =~ ClearResponsibilities + EffectiveAnswers + Feedback + ClearOrganization +
TradPed =~ ClearResponsibilities + EffectiveAnswers + Feedback + ClearOrganization
Valued =~ ValObjectives + IncrUnderstanding + IncrInterest
SCRPed =~ MultPerspectives + InclusvClassrm + DEIIntegration + EquitableEval
ClearPresentation ~ ~ 0.852 * ClearPresentation

#fixes the relations between g and each of the factors to 0.0
Evals ~ ~ 0 * TradPed
Evals ~ ~ 0 * Valued
Evals ~ ~ 0 * SCRPed

#fixes the relations (covariances) between each of the factors to 0.0
TradPed ~ ~ 0 * Valued
TradPed ~ ~ 0 * SCRPed
Valued ~ ~ 0 * SCRPed

"
```

```
set.seed(240311)
bifacF <- lavaan:::cfa(bifacM, data = items, std.lv = TRUE)
lavaan:::summary(bifacF, fit.measures = TRUE, standardized = TRUE, rsquare = TRUE)
```

```
lavaan 0.6.17 ended normally after 48 iterations
```

Estimator	ML
Optimization method	NLMINB

Number of model parameters	35
Number of observations	267
<b>Model Test User Model:</b>	
Test statistic	262.623
Degrees of freedom	43
P-value (Chi-square)	0.000

**Model Test Baseline Model:**

Test statistic	1940.157
Degrees of freedom	66
P-value	0.000

**User Model versus Baseline Model:**

Comparative Fit Index (CFI)	0.883
Tucker-Lewis Index (TLI)	0.820

**Loglikelihood and Information Criteria:**

Loglikelihood user model (H0)	-2996.890
Loglikelihood unrestricted model (H1)	-2865.578
Akaike (AIC)	6063.779
Bayesian (BIC)	6189.333
Sample-size adjusted Bayesian (SABIC)	6078.362

**Root Mean Square Error of Approximation:**

RMSEA	0.138
90 Percent confidence interval - lower	0.122
90 Percent confidence interval - upper	0.155
P-value H_0: RMSEA <= 0.050	0.000
P-value H_0: RMSEA >= 0.080	1.000

**Standardized Root Mean Square Residual:**

SRMR	0.084
------	-------

**Parameter Estimates:**

Standard errors	Standard
Information	Expected
Information saturated (h1) model	Structured

## Latent Variables:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
<b>Evals =~</b>						
ClearRspnsblts	0.621	0.045	13.943	0.000	0.621	0.786
EffectivAnswrs	0.637	0.045	14.110	0.000	0.637	0.784
Feedback	0.679	0.050	13.532	0.000	0.679	0.756
ClearOrganiztn	0.798	0.061	13.135	0.000	0.798	0.753
ClearPresenttn	0.721	0.070	10.301	0.000	0.721	0.613
ValObjectives	0.293	0.037	7.996	0.000	0.293	0.488
IncrUndrstndng	0.496	0.048	10.227	0.000	0.496	0.600
IncrInterest	0.638	0.055	11.557	0.000	0.638	0.661
MultPerspectvs	0.643	0.046	13.886	0.000	0.643	0.763
InclusvClassrm	0.359	0.040	9.079	0.000	0.359	0.552
DEIintegration	0.286	0.047	6.062	0.000	0.286	0.386
EquitableEval	0.455	0.036	12.710	0.000	0.455	0.713
<b>TradPed =~</b>						
ClearRspnsblts	0.394	0.170	2.313	0.021	0.394	0.498
EffectivAnswrs	0.121	0.086	1.405	0.160	0.121	0.149
Feedback	0.085	0.081	1.050	0.294	0.085	0.095
ClearOrganiztn	0.215	0.136	1.582	0.114	0.215	0.203
ClearPresenttn	0.118	0.112	1.052	0.293	0.118	0.100
<b>Valued =~</b>						
ValObjectives	0.141	0.043	3.248	0.001	0.141	0.235
IncrUndrstndng	0.385	0.081	4.752	0.000	0.385	0.466
IncrInterest	0.578	0.111	5.188	0.000	0.578	0.598
<b>SCRPed =~</b>						
MultPerspectvs	0.226	0.047	4.824	0.000	0.226	0.268
InclusvClassrm	0.335	0.051	6.577	0.000	0.335	0.515
DEIintegration	0.574	0.073	7.859	0.000	0.574	0.775
EquitableEval	0.073	0.036	2.002	0.045	0.073	0.114

## Covariances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
<b>Evals ~~</b>						
TradPed	0.000				0.000	0.000
Valued	0.000				0.000	0.000
SCRPed	0.000				0.000	0.000
<b>TradPed ~~</b>						
Valued	0.000				0.000	0.000
SCRPed	0.000				0.000	0.000
<b>Valued ~~</b>						
SCRPed	0.000				0.000	0.000

## Variances:

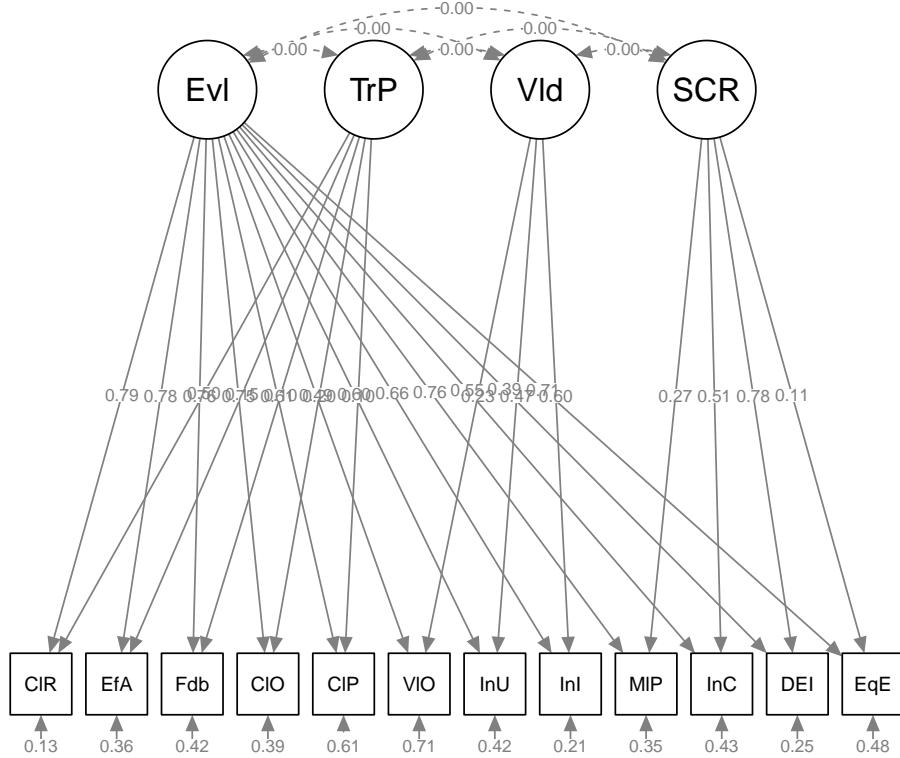
	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
.ClearPresenttn	0.852				0.852	0.615
.ClearRspnsblts	0.084	0.134	0.627	0.530	0.084	0.135
.EffectivAnswrs	0.240	0.025	9.637	0.000	0.240	0.363

.Feedback	0.337	0.034	9.812	0.000	0.337	0.419
.ClearOrganiztn	0.440	0.050	8.847	0.000	0.440	0.392
.ValObjectives	0.255	0.023	10.921	0.000	0.255	0.707
.IncrUnderstndng	0.288	0.058	4.942	0.000	0.288	0.422
.IncrInterest	0.191	0.120	1.598	0.110	0.191	0.205
.MultPerspectvs	0.247	0.027	9.281	0.000	0.247	0.347
.InclusvClassrm	0.182	0.028	6.557	0.000	0.182	0.430
.DEIIntegration	0.137	0.074	1.849	0.064	0.137	0.250
.EquitableEval	0.195	0.019	10.102	0.000	0.195	0.479
Evals	1.000				1.000	1.000
TradPed	1.000				1.000	1.000
Valued	1.000				1.000	1.000
SCRPed	1.000				1.000	1.000

R-Square:

	Estimate
ClearPresenttn	0.385
ClearRspnsblts	0.865
EffectivAnswrs	0.637
Feedback	0.581
ClearOrganiztn	0.608
ValObjectives	0.293
IncrUnderstndng	0.578
IncrInterest	0.795
MultPerspectvs	0.653
InclusvClassrm	0.570
DEIIntegration	0.750
EquitableEval	0.521

```
semPlot::semPaths(bifacF, layout = "tree", style = "lisrel", what = "col",
whatLabels = "stand")
```

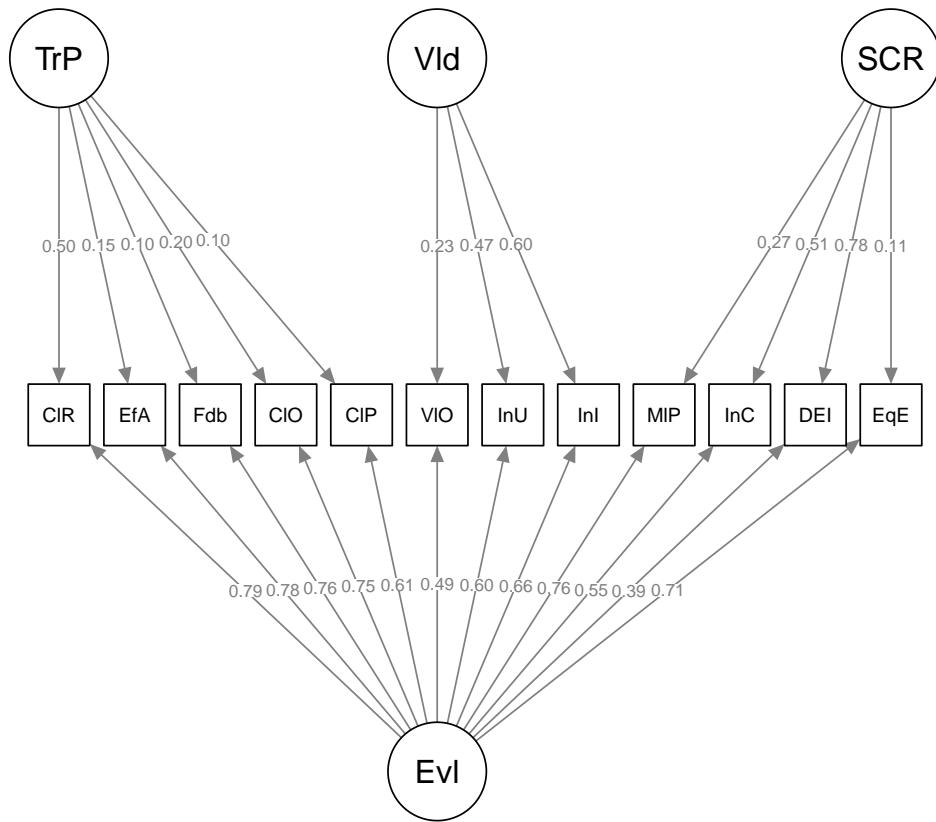


In making our map the first 12 values refer to the items, 13 refers to  $g$ , and 14-16 point to the factors.

```
m = matrix(nrow = 3, ncol = 12)
m[1, ] = c(14, 0, 0, 0, 0, 15, 0, 0, 0, 0, 0, 16)
m[2, ] = 1:12
m[3, ] = c(0, 0, 0, 0, 0, 13, 0, 0, 0, 0, 0, 0)
m
```

	[,1]	[,2]	[,3]	[,4]	[,5]	[,6]	[,7]	[,8]	[,9]	[,10]	[,11]	[,12]
[1,]	14	0	0	0	0	15	0	0	0	0	0	16
[2,]	1	2	3	4	5	6	7	8	9	10	11	12
[3,]	0	0	0	0	0	13	0	0	0	0	0	0

```
semPlot::semPaths(bifacF, "model", "std", layout = m, residuals = FALSE,
exoCov = FALSE)
```



Code for saving the results as a .csv file.

```

bifacFFitStats <- tidySEM::table_fit(bifacF)
bifacF_paramEsts <- tidySEM::table_results(bifacF, digits = 3, columns = NULL)
bifacFCorrs <- tidySEM::table_corrs(bifacF, digits = 3)
# to see each of the tables, remove the hashtag corrFitStats
# corrF_paramEsts corrFCorrs
  
```

Next, I export them.

```

write.csv(bifacFFitStats, file = "bifacFFitStats.csv")
write.csv(bifacF_paramEsts, file = "bifacF_paramEsts.csv")
write.csv(bifacFCorrs, file = "bifacFCorrs.csv")
  
```

### 11.17.9 Narrate adequacy of fit with $\chi^2$ , CFI, RMSEA, SRMR (write a mini-results section)

The bifactor model regressed each item on its respective factor while simultaneously regressing each indicator onto an overall evaluation scale. The initial specification failed to converge. Adding the “std.lv = TRUE” to the lavaan:::cfa function resulted in convergence. It also identified a negative covariance (i.e., a “Heywood” case). We resolved this issue by respecifying the model with a multiplication of the variance of the item (ClearPresentation) onto itself. This model had the second best fit of those compared thus far:  $\chi^2(43) = 262.62, p < .001, CFI = .883, RMSEA = .138, 90$  (the correlated factors and second order structures had the best fit). Factor loadings for the three factors ranged from .10 to .50 for the TradPed scale, .24 to .60 for the Valued-by-me scale, and .11 to .78 for the SCRPed scale. Factor loadings for the overall evaluation scale (g) ranged from .37 to .79.

### 11.17.10 Compare model fit with $\chi^2\Delta$ , AIC, BIC

```
lavaan:::lavTestLRT(uniDfit, corrF_fit, secondF, bifacF)
```

```
Warning in lavaan:::lavTestLRT(uniDfit, corrF_fit, secondF, bifacF): lavaan WARNING:
  Some restricted models fit better than less restricted models;
  either these models are not nested, or the less restricted model
  failed to reach a global optimum. Smallest difference =
  -37.8284756558401
```

```
Warning in lavaan:::lavTestLRT(uniDfit, corrF_fit, secondF, bifacF): lavaan
WARNING: some models have the same degrees of freedom
```

#### Chi-Squared Difference Test

	Df	AIC	BIC	Chisq	Chisq diff	RMSEA	Df diff
bifacF	43	6063.8	6189.3	262.62			
corrF_fit	51	6010.0	6106.8	224.79	-37.828	0.00000	8
secondF	51	6010.0	6106.8	224.79	0.000	0.00000	0
uniDfit	54	6124.1	6210.2	344.97	120.178	0.38248	3
				Pr(>Chisq)			
bifacF							
corrF_fit				1			
secondF							
uniDfit				<0.0000000000000002	***		
---							
Signif. codes:	0	'***'	0.001	'**'	0.01	'*'	0.05
	.	.	.	.	0.1	' '	1

```
lavaan:::lavTestLRT(secondF, bifacF)
```

Warning in lavaan:::lavTestLRT(secondF, bifacF): lavaan WARNING:  
 Some restricted models fit better than less restricted models;  
 either these models are not nested, or the less restricted model  
 failed to reach a global optimum. Smallest difference =  
 -37.8284756538201

#### Chi-Squared Difference Test

	Df	AIC	BIC	Chisq	Chisq diff	RMSEA	Df diff	Pr(>Chisq)
bifacF	43	6063.8	6189.3	262.62				
secondF	51	6010.0	6106.8	224.79	-37.828	0	8	1

### 11.17.11 Calculate omega hierarchical ( $\omega_h$ ) and omega-hierarchical-subsubscales ( $\omega_{h-ss}$ )

Given that our results favored the second-order structure, I will calculate model-based assessments of the omegas with that function (*semTools::reliabilityL2*) and object (*secondF*).

To use *semTools* we switch functions to *reliabilityL2*. The specification

```
semTools::reliabilityL2(secondF, "Evals")
```

	omegaL1	omegaL2	partialOmegaL1
	0.8832659	0.9237686	0.9336447

The estimate listed under *omegaL1* ( $\omega_{ho}$ ) represents the proportion of the course evaluation total score variance due to the higher-order factor. The value is .88 and, being above .80, indicates strong reliability.

We can apply the *semTools::reliability()* function to the second-order factor to obtain omega values for the subscales. Below the alpha coefficients, the omega values indicate how reliably each subscale measures its lower order factor. For example, 80% of the total variance of a total score comprised of only the 9 items in the AS scale is explained by the AS lower order factor.

```
semTools::reliability(secondF)
```

	TradPed	Valued	SCRPed
alpha	0.8973027	0.7573447	0.8037983
omega	0.9005696	0.8062411	0.8057420
omega2	0.9005696	0.8062411	0.8057420
omega3	0.9019489	0.8109489	0.7868658
avevar	0.6470672	0.6063643	0.5223301

Subscale omegas range from .81 to .90.

In this case,  $\omega_{ho}$  ( $ho$  = higher order) [Flora, 2020b] represents the proportion of total-score variance that is due to the higher-order factor. As such, it represents the reliability of a total score for measuring a single construct that influences all items.

We obtained estimates of model-based internal consistency associated with the second-order model. Specifically, we calculated  $\omega_{ho}$ , which represents the proportion of GRM-SAAW total score variance due to the higher-order factor. The value of 0.88 indicates strong reliability. Similarly, we obtained omega values for the subscales. These ranged were 0.90, 0.81, and 0.81 for Traditional Pedagogy, Valued-by-the-Student, and Socially Responsive Pedagogy, respectively. Taken together, the omega values from our second order model suggest suggest a strong general factor and strong subscales.

### 11.17.12 APA style results with table(s) and figure

Because we have written mini-results throughout, we can assemble them into a full results section. Keep in mind that most CFA models will continue testing multidimensional models. Thus, the entire analysis continues in the next lesson and associated practice problem.

**Model testing.** To evaluate the models we, we used confirmatory factor analysis (CFA) in the R package, lavaan (v.0-6.17) with maximum likelihood estimation. Our sample size was 267 We selected fit criteria for their capacity to assess different aspects of the statistical analysis. As is common among SEM researchers, we reported the Chi-square goodness of fit (2). This evaluates the discrepancy between the unrestricted sample matrix and the restricted covariance matrix. Although the associated *p*-value indicates adequate fit when the value is non-significant, it is widely recognized that a large sample size can result in a statistically significant *p* value (Byrne, 2016b). The comparative fit index (CFI) is an incremental index, comparing the hypothesized model to the independent/baseline model. Adequate fit is determined when CFI values are at least .90 and perhaps higher than .95 [Kline, 2016]. The root mean square error of approximation (RMSEA) takes into account the error of approximation in the population and expresses it per degree of freedom. As such, the fit indicator considers the complexity of the model. Ideal values are equal to or less than .05, values less than .08 represent reasonable fit, and values between .08 and .10 represent mediocre fit. The standardized root mean residual is a standardized measure of the mean absolute covariance residual – the overall difference between the observed and predicted correlations. Values greater than .10 may indicate poor fit and inspection of residuals is then advised. Kline (2016) advised researchers to be cautious when using these criteria as strict cut-offs. Elements such as sample size and model complexity should be considered when evaluating fit.

Our first model was unidimensional where each of the 12 items loaded onto a single factor representing overall course evaluations. The Chi-square index was statistically significant ( $\chi^2(54) = 344.97, p < .001$ ) indicating likely misfit. The CFI value of .85 indicated poor fit. The RMSEA = .14 (90% CI [.13, .16]) suggested serious problems. The SRMR value of .07 was below the warning criteria of .10. The AIC and BIC values were 6124.13 and 6134.13, respectively, and will become useful in comparing subsequent models.

Our second model was a single-order, multidimensional model where each of the 12 items loaded onto one of four factors. Standardized pattern coefficients ranged between .74 and .85 on the TradPed factor, between .56 and .84 on the Valued factor, and between .57 and .85 on the SCRPed factor. The Chi-square index was statistically significant ( $\chi^2(51) = 224.795, p < 0.001$ ) indicating some degree of misfit. The CFI value of .907 met the recommendation of .90 but fell below the recommendation of .95. The RMSEA = .113 (90% CI [.098, .128]) was higher than recommended. The SRMR value of .061 remained below the warning criteria of .10. The AIC and BIC values were 6009.95 and 6021.20, respectively.

Our third model represented a second order structure where three first-order factors loaded onto a second factor model. Across a variety of indices, model fit improved:  $\chi^2(51) = 224.80, p < .001$ ,  $CFI = .907$ ,  $RMSEA = .113, .90$ . Factor loadings ranged from .75 to .85 for the TradPed scale, .56 to .84 for the Valued-by-Me scale, .57 to .85 for the SCRPed scale, and .80 to .98 for the total scale.

The bifactor model regressed each item on its respective factor while simultaneously regressing each indicator onto an overall evaluation scale. The initial specification failed to converge. Adding the “std.lv = TRUE” to the lavaan::cfa function resulted in convergence. It also identified a negative covariance (i.e., a “Heywood” case). We resolved this issue by respecifying the model with a multiplication of the variance of the item (ClearPresentation) onto itself. This model had the second best fit of those compared thus far:  $\chi^2(43) = 262.62, p < .001$ ,  $CFI = .883$ ,  $RMSEA = .138, .90$  (the correlated factors and second order structures had the best fit). Factor loadings for the three factors ranged from .10 to .50 for the TradPed scale, .24 to .60 for the Valued-by-me scale, and .11 to .78 for the SCRPed scale. Factor loadings for the overall evaluation scale (g) ranged from .37 to .79.

As shown in our table of model comparisons, the Chi-square difference tests between models showed statistically significant differences between the unidimensional and correlated factors model ( $\chi^2(3) = 120.178, p < .001$ ). The second-order model and bifactor model did not differ from each other ( $\chi^2(8) = -37.828, p = 1.000$ ). The CFI, RMSEA, SRMR, AIC, and BIC values all favored the second-order and correlated factors models (which did not differ from each other).

We obtained estimates of model-based internal consistency associated with the second-order model. Specifically, we calculated  $\omega_{ho}$ , which represents the proportion of GRM-SAAW total score variance due to the higher-order factor. The value of 0.88 indicates strong reliability. Similarly, we obtained omega values for the subscales. These ranged were 0.90, 0.81, and 0.81 for Traditional Pedagogy, Valued-by-the-Student, and Socially Responsive Pedagogy, respectively. Taken together, the omega values from our second order model suggest a strong general factor and strong subscales.

### 11.17.13 Explanation to grader

# Chapter 12

## Invariance Testing

[Screencasted Lecture Link](#)

The focus of this lecture is invariance testing – that is, evaluating if a scale operates equivalently across two samples.

### 12.1 Navigating this Lesson

There is a little more than 1.5 hours of lecture. If you work through the materials with me it would be plan for an additional 1.5 hours.

While the majority of R objects and data you will need are created within the R script that sources the chapter, occasionally there are some that cannot be created from within the R framework. Additionally, sometimes links fail. All original materials are provided at the [GitHub site](#) that hosts the book. More detailed guidelines for ways to access all these materials are provided in the OER's [introduction](#).

#### 12.1.1 Learning Objectives

Focusing on this week's materials, make sure you can:

- Specify a series of models that will test for multigroup invariance.
- Interpret model adequacy and fit.
- Compare models on the basis of statistical criteria.
- Recall which parameters (e.g., structures, loadings, intercepts, residuals) are constrained in configural, weak, strong, and strict models
  - and know, without evaluation, which of these models will (necessarily) have the best fit
- interpret  $\chi^2_D$  and  $\Delta CFI$  tests to determine if there are statistically significant differences in model fit.

### 12.1.2 Planning for Practice

In each of these lessons I provide suggestions for practice that allow you to select one or more problems that are graded in difficulty. The least complex is to change the random seed and rework the problem demonstrated in the lesson. The results *should* map onto the ones obtained in the lecture.

The second option involves utilizing one of the simulated datasets available in this OER. The **last lesson** in the OER contains three simulations that could be used for all of the statistics-based practice suggestions. Especially if you started with one of these examples in an earlier lesson, I highly recommend you continue with that.

As a third option, you are welcome to use data to which you have access and is suitable for CFA. This could include other simulated data, data found on an open access repository, data from the ReCentering Psych Stats survey described in the **Qualtrics lesson**, or your own data (presuming you have permission to use it).

In any case, please plan to:

- Specify, interpret, and write up preliminary results for CFA models that examine
  - entire sample (making no distinction between groups)
  - configural invariance
  - weak invariance
  - strong invariance
  - strict invariance
- Create an APA style results section with appropriate table(s) and figure(s)
- Talk about it with someone

### 12.1.3 Readings & Resources

In preparing this chapter, I drew heavily from the following resource(s). Other resources are cited (when possible, linked) in the text with complete citations in the reference list.

Byrne, B. M. (2016). Adaptation of assessment scales in cross-national research: Issues, guidelines, and caveats. *International Perspectives in Psychology: Research, Practice, Consultation*, 5(1), 51–65. <https://doi.org/10.1037/ipp0000042>

Conover, K. J., Israel, T., & Nylund-Gibson, K. (n.d.). Development and Validation of the Ableist Microaggressions Scale. *The Counseling Psychologist*, 30.

- Our research vignette for this lesson

Hirschfeld, G., & von Brachel, R. (2014). Multiple-Group confirmatory factor analysis in R – A tutorial in measurement invariance with continuous and ordinal indicators. *19*(7), 12.

Kline, R. (2016). Principles and practice of structural equation modeling (Fourth ed., Methodology in the social sciences). New York: The Guilford Press.

- Chapter 16: Multiple-Samples Analysis and Measurement Invariance

## 12.2. INVARIANCE TESTING (AKA MULTIPLE-SAMPLES SEM OR MULTIPLE-GROUP CFA [MG-CFA])

Rosseel, Y. (2019). The *lavaan* tutorial. Belgium: Department of Data Analysis, Ghent University. <http://lavaan.ugent.be/tutorial/tutorial.pdf>

- Section 8/Multiple groups

### 12.1.4 Packages

The packages used in this lesson are embedded in this code. When the hashtags are removed, the script below will (a) check to see if the following packages are installed on your computer and, if not (b) install them.

```
# will install the package if not already installed
# if(!require(MASS)){install.packages('MASS')}
# if(!require(sjstats)){install.packages('sjstats')}
# if(!require(psych)){install.packages('psych')}
# if(!require(tidyverse)){install.packages('tidyverse')}
# if(!require(lavaan)){install.packages('lavaan')}
# if(!require(semPlot)){install.packages('semPlot')}
# if(!require(semTable)){install.packages('semTable')}
```

## 12.2 Invariance Testing (aka Multiple-Samples SEM or Multiple-Group CFA [MG-CFA])

### 12.2.1 Introducing the Topic and the Terminology

As we enter this topic of *invariance* it will be useful to take a few moments to look at the definitions. It is common for the use of the terms “invariance” and “noninvariance” to be confusing. As we look at each of these definitions I have often wondered why we don’t start with “variant” or “variation.”

- **Variance:** is not a term we are using today. However, recalling the notions of *variable* (which contrasts with *constant*), *variation*, *variability* may help you with *invariance* (which means, it doesn’t vary).
- **Invariance** is synonymous with equivalence. That is, there are not statistically significant differences between the two versions/models being compared.
- **Noninvariance** is synonymous with nonequivalence. That is, there are statistically significant differences between the two versions/models being compared
- **Equality constraints** are imposed by the researcher when we specify (require) two or more parameters to be equal. The particular constraints could be placed on between factor loadings, covariances between factors, intercepts, error variances, error covariances, and so forth. Such constraints simplify the analysis because only one coefficient is needed rather than two. In a multiple-samples/groups, like invariance testing, a cross-group equality constraint forces the computer to derive equal estimates of the same parameter across all groups. This specification corresponds to the null hypothesis that the parameter is equal in all populations from which

the samples are drawn. We then conduct formal difference tests to see if, in fact, the model fit is worse when the two groups are constrained to be equal on that parameter (or more likely, set of parameters).

**Measurement invariance** is a property when a set of indicators measures the same constructs with equal precision over different samples.

A scale is said to have measurement invariance (or, measurement equivalence) across groups if respondents with identical levels of the latent construct have the same expected raw scores on the measure [Hirschfeld and von Brachel, Ruth, 2014].

We can think of this in several ways:

- whether values of model parameters of substantive interest vary in meaningful ways across different samples,
- as an interaction – whether sample membership *moderates* the relations specified in a model; if there is evidence for a *group x model* interaction, then the program must be allowed to derive separate estimates of some parameters in each sample in order for the model to have acceptable fit over all samples involved,
- whether scores from the operationalization of a construct have the same meaning under differing conditions
  - these conditions could involve time of measurement, test administration methods, or populations (national samples, clinical/community samples, children/adults, and so forth)
  - absence of invariance says that findings of differences between persons cannot be unambiguously isolated from differences owing to time, methods, group membership (thus, there would be no clear basis for drawing inferences from the scores)

**Longitudinal measurement invariance** evaluates the stability in measurement parameters over time for the same population.

**Method invariance** is concerned with whether different methods of administration (online survey versus paper/pencil) are invariant.

My experience with invariance testing is the multiple language/cross-cultural/international context. Here we often ask, “What makes a test culturally transferable?” Byrne’s [2016a] article provides a current, excellent, thorough review. Highlights include:

- In the past we could claim that a test was culturally adaptive if it involved
  - Translation/backtranslation
  - Replication of factor structure within the culture
  - Replication of validity and reliability estimates
- Today, there is a movement toward testing adaptation
  - Including all the past steps, PLUS
  - Invariance testing to explore the factor structure across cultures
  - Investigation of item bias and construct relations

## 12.2. INVARIANCE TESTING (AKA MULTIPLE-SAMPLES SEM OR MULTIPLE-GROUP CFA [MG-CFA])

In this lesson we will focus rather narrowly on Byrne's [2016a] strategy for the statistical/psychometric evaluation of invariance. You might also be interested in Gerstein's, Systematic Test of Equivalence Procedure (STEP; [2021]), which walks the researcher, item-by-item, through six step analysis of the cultural appropriateness of each item. The researcher is prompted to consider why items are and are not appropriate/relevant and how they might be modified.

### 12.2.2 Evaluation Strategies

There are two primary options for establishing multigroup invariance.

#### 12.2.2.1 Free baseline approach

In the free baseline approach, testing for measurement invariance is a hierarchical, *model trimming*, strategy. Specifically, the *configural model* (the initial, unconstrained model) is gradually restricted by adding cross-group equality constraints in a sequence that corresponds to *weak*, *strong*, and *strict* invariance. At the point that the invariance hypothesis cannot be retained, testing stops (i.e., more restricted models are not considered).

In the free baseline approach:

- Respecification moves from nesting to nested models.
- Fit generally worsens in each subsequent model.
- The goal is non-significant differences in fit with each additional set of cross-group equality constraints.

#### 12.2.2.2 Constrained baseline approach

In the constrained baseline approach, testing for measurement invariance is a *model building* approach where the most restricted model (*strict*; with equal pattern coefficients, intercepts, and residuals) is the baseline. If necessary, these are sequentially released and compared backwards through the hierarchy (*strong*, *weak*, *configural*) but some researchers will switch around the order in which constraints are released.

In the constrained baseline approach:

- Respecification moves from nested to nesting models.
- Fit generally improves in each subsequent model
- The goal is to have satisfactory fit in the most restricted model (but this often is not the case).

Ideally-and-theoretically, model trimming and building approaches will end up in the same place, but this is not guaranteed.

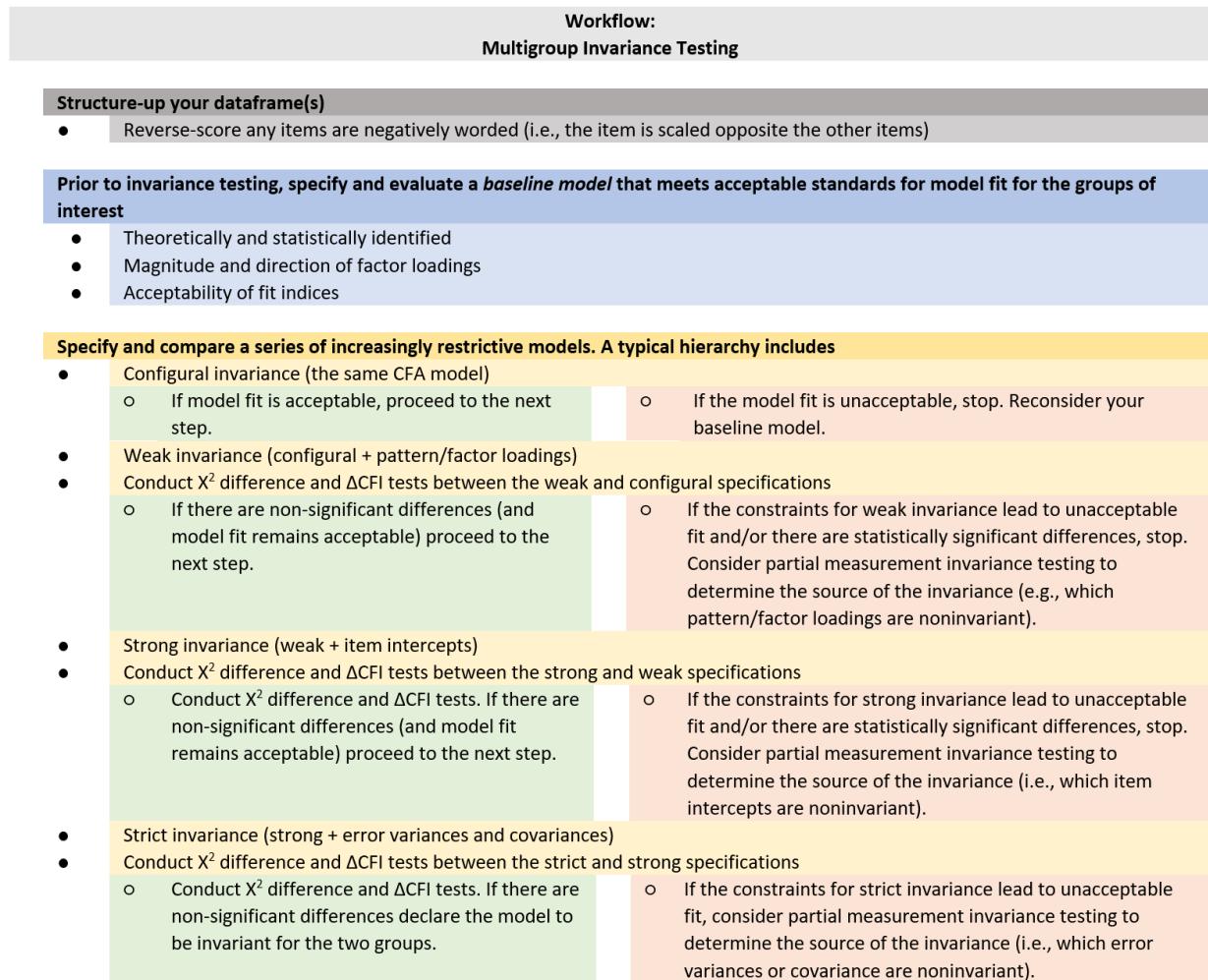


Figure 12.1: Image of a flowchart and decision-tree for multi-group invariance testing

### 12.2.3 Invariance Testing Workflow

Today we will use the *free baseline* approach in testing the measurement equivalence of a scale across two groups.

Multigroup invariance testing involves:

- Structuring up the item-level dataset (i.e., reverse-coding any variables)
- For the groups-of-interest, identifying a common *baseline model* that meets acceptable standards for model fit. These standards include:
  - theoretical and statistical identification
  - appropriate magnitude and direction of factor loadings.
- Specifying and comparing a series of increasingly restrictive models. These models include:
  - Configural invariance (the same CFA structure)
  - Weak invariance (configural + pattern/factor loadings)
  - Strong invariance (weak + item intercepts)
  - Strict (strong + error variances and covariances)
- At the point (i.e., the model) that fit is unacceptable, consider investigating and reporting the source of partial measurement invariance.

### 12.2.4 Successive Gradations of Measurement Invariance

There are four levels of invariance. Invariance testing is a sequential and incremental process. Thus, each successive level of variance is prerequisite on meeting the criteria of the prior. Different authors use different names for these same notions.

#### 12.2.4.1 Configural invariance

Configural invariance is the least restrictive level. It implies that the number of latent variables and the pattern of loadings on the latent variables on indicators are similar across the groups.

Configural invariance is tested by specifying the same CFA model in each group. Both the number of factors and the correspondence between factors and indicators are the same, but all parameters are freely estimated in each group.

- If this model is not consistent with the data, then measurement invariance does not hold at any subsequent level.
- If the model is retained, it says that the same factors are manifested (in potentially different ways) in each group. Differences could include:
  - unequal pattern coefficients
  - unequal intercepts
  - unequal error variances
- If there is only configural variance, a different weighting scheme would be needed for each group.

### 12.2.4.2 Weak invariance

Weak invariance is sometimes termed *pattern invariance* and *metric invariance*. Weak invariance requires *configural variance* plus equality of unstandardized pattern coefficients. That is, the magnitude of the loadings is similar across the groups.

The hypothesis of weak invariance is tested by:

- Imposing an equality constraint over groups on the unstandardized coefficient of each indicator. Then
- Comparing with the chi-square difference ( $\chi_D^2$ ) test the configural invariance model and the weak invariance model
  - In comparing models, if the fit of more restrictive invariance model tested is *not appreciably worse* than the immediately prior restrictive model, then the more restrictive weak invariance hypothesis is retained.
  - Thus, the weak invariance model would be compared to the configural invariance model. IF we use the  $\chi_D^2 > .05$  (and we'll learn later that there are better/more options), then we can claim weak invariance.
  - Think back to what we learned about comparing nested/hierarchical models. As we continue through this invariance *hierarchy* (configural, weak, strong, strict), each of the more restrictive measures will have worse fit (the prior, lesser restrictive model will be the nesting model with "more sticks" and fewer degrees of freedom). Therefore, we would really like there to be NO DIFFERENCE in model fit when we add between-group equality constraints.
- If weak invariance is supported, then we can claim that constructs are manifested the same way in each group. This means that:
  - slopes from regressing the indicators on their respective factors are equal across all groups, and
  - factor scores can be calculated using the same weighting scheme in all groups tested
- If weak invariance is rejected, then...
  - the factors (or at least a subset of items corresponding to those factors) have different meanings in different groups
  - *extreme response styles (ERS)* may affect response variability, for example, low ERS is the tendency to avoid endorsing the most extreme options (e.g., never, always); high ERS is the tendency to endorse the most extreme options
- If we can support weak invariance, we are justified in formally comparing estimated factor variances or covariances across different groups, but because indicators are affected by both factors and sources of unique (residual) variation, we need MORE in order to statistically compare observed variances or covariances over groups. This comes from the next level.

### 12.2.4.3 Strong invariance\*\*

Strong invariance is also termed *scalar invariance*; it is predicated on weak invariance. Strong invariance implies that the item loadings plus the item intercepts are similar across the groups. It

## 12.2. INVARIANCE TESTING (AKA MULTIPLE-SAMPLES SEM OR MULTIPLE-GROUP CFA [MG-CFA])53

also implies that there are no systematic response biases. It is required in order to meaningfully compare the means of latent variables across different groups.

- Item intercepts are considered to be the origin or starting value of the scale that your factor is based on. Thus, participants who have the same value on the latent construct should have equal values on the items on which the construct is based. These are related to the mean structure, hence you'll see some refer to this as means.
- The intercept estimates the score on an indicator given a true score of zero on the corresponding factor
- *Equality of intercepts* says that different groups use the response scale of that indicator in the same way; that is, a person from one group and a person from a different group with the same level on the factor should obtain the same score on the indicator.

The hypothesis of strong invariance is tested by:

- Imposing equality constraints on unstandardized pattern coefficients and intercepts. Then,
- Comparing this model with the model of the equality-constrained pattern coefficients (i.e., the weak invariance model) with the  $\chi^2_D$ 
  - If the fit of the more restrictive invariance model tested is *not appreciably worse* than the immediately prior restrictive model, then the more restrictive weak invariance hypothesis is retained.
  - Thus, the strong invariance model would be compared to the weak invariance model. If  $\chi^2_D > .05$ , then we can claim strong invariance.
- If strong invariance is rejected, then we may be concerned about a *differential additive (acquiescence) response style*: systematic influences unrelated to the factors that decrease or increase the overall level of responding on an indicator in a particular population
  - Example: if patients are weighed in street clothes in the clinic and in a gown at the hospital, an additive constant is added to true body weight, dependent upon where patients are tested; this contaminates the estimates of mean weight differences over the two clinics.
  - If a response style affects all indicators, then invariance testing will not detect this pattern; instead the estimates of the construct will be influenced by response styles that are uniform over all indicators.
- **Differential item functioning** is the pattern that an indicator has appreciably unequal pattern coefficients or intercepts over groups; DIF violates measurement invariance.
  - A goal in multiple-samples CFA is to locate the indicator(s) responsible for rejecting the hypothesis of weak or strong invariance
  - In test development, we flag items as candidates for revision or deletion
- If strong invariance is supported
  - group differences in estimated factor means will be unbiased

- group differences in indicators means or estimated factor scores will be directly related to the factor means and will not be distorted by differential additive response bias
- the factors have a common meaning over groups and any constant effects on the indicators are cancelled out when observed means are compared over groups
- strong invariance is the minimal level required for meaningful interpretation of group mean contrasts

#### 12.2.4.4 Strict invariance

Strict invariance requires strong invariance plus the equality in error variances and covariances across groups. This means that the indicators measure the same factors in each group with the same degree of precision. There are some rifts about what exactly constitutes strict invariance:

- residual invariance is required in order to claim that factors are measured identically across group (Deshon, 2004; Wu et al., 2007)
- Because unique (residual) error variance reflects random measurement error and systematic variance, the sum of these two components must be equal across groups (Little, 2013). Kline [2016] says that it may be too strict and somewhat unreasonable/unattainable. Little (2013) also cautioned against enforcing this requirement because if the sum of random and systematic parts of unique variance is not exactly equal, the amount of misfit due to equality-constrained residual variances must contaminate estimates elsewhere in the model.

#### 12.2.5 Tests for Model Comparison

It is not sufficient to declare any level of invariance (i.e., constrained (configural, weak, string, or strong) to be adequate on the basis of the traditional evaluation of fit (i.e., strength and significance of factor loadings, fit indices). Rather the whole models must be compared through formal statistical comparison. There are several options and they all have caveats.

A non-significant chi-square difference test ( $\chi^2_D > .05$ ) that compares less-and-more restrictive models indicates that the stricter invariance hypothesis should *not* be rejected. That is, it supports invariance for the more restricted model.

- In large samples, this could be statistically significant, even though the absolute differences in parameter estimates are trivial.
- Thus, the  $\chi^2_D$  could indicate lack of measurement invariance when the imposition of cross-group equality constraints makes relatively little difference in fit. Options for verifying:
  - compare unstandardized solutions across groups
  - inspect changes in approximate fit indices
  - BUT...there are few guidelines for how to do this

When the CFA change statistic is smaller than .01 ( $\Delta CFI < .01$ ) there is evidence that the stricter invariance hypothesis should *not* be rejected. That is, it supports invariance for the more restricted model.

- Simulation studies suggested that stability for different model characteristics such as number of indicators per factor. Here are some findings (guidelines?) for different testing scenarios:
  - In super large samples ( $\sim 6,000$ ) use  $\Delta CFI < .002$ .
  - When group sizes are small ( $n < 300$ ) and unequal, use  $\Delta CFI < .005$  and  $\Delta RMSEA < .010$ .
  - When group sizes are larger ( $n > 300$ ), equal, and the pattern of invariance was mixed (i.e., there are at least two invariant parameters, each of which is from a different category [pattern coefficient, intercept, residual variance]), use  $\Delta CFI < .010$  and  $\Delta RMSEA < .015$ .

$\Delta NCI$  was also stable, but Kline [2016] did not provide a threshold (and I don't see the NCI reported much in psychometrics papers).

The general practice seems to favor reporting both the  $\chi^2_D$  and  $\Delta CFI$ . Even if  $\chi^2_D > .05$ , a  $\Delta CFI < .01$  supports invariance between models.

### 12.2.6 Partial measurement invariance

The notion of partial measurement invariance was introduced by Byrne, Shavelson, and Muthén (1989) and is often used to describe an *intermediate* state of invariance. For example, weak invariance assumes cross-group equality of each unstandardized pattern coefficient. If some, but not all pattern coefficients are invariant, then only *partial weak invariance* can be claimed.

The researcher may investigate which pattern coefficients are noninvariant and relax (or free) them to differ across groups. Once freed, the research might choose to compare the models with the  $\chi^2_D > .05$ , a  $\Delta CFI < .01$ . Once enough pattern coefficients have been freed and the fit across models is equivalent, the researcher might continue the process of determining the degree of invariance in the more restricted evaluations (e.g., strong, strict).

Even if the researcher does not continue with testing for invariance in the increasingly restricting models, they have learned which pattern coefficients vary across groups.

## 12.3 Research Vignette

This lesson's research vignette emerges from Conover et al's Ableist Microaggressions Scale (AMS [2017]). The article reports on a series of three studies comprised the development, refinement, and psychometric evaluation of the AMS. I simulated data from the results of the exploratory factor analysis in the second study.

Conover et al. [2017] reported support for using a total scale score (22 items) or four, correlated, subscales. Below, I list the four subscales, their number of items, and a single example item. At the outset, let me provide a content warning. For those who hold this particular identity (or related identities) the content in the items may be upsetting. In other lessons, I often provide a variable name that gives an indication of the primary content of the item. In the case of the AMS, I will simply provide an abbreviation of the subscale name and its respective item number. This will allow us to easily inspect the alignment of the item with its intended factor, and hopefully minimize discomfort. If you are not a member of this particular identity, I encourage you to learn

about these microaggressions by reading the article in its entirety. Please do not ask members of this group to explain why these microaggressions are harmful or ask if they have encountered them.

There are 20 items on the AMS scale. The frequency scaling ranged from 0(*never*) to 5(*very frequently*). Higher scores indicate higher frequency of microaggressions.

The four factors, number of items, and sample item are as follows:

- Helplessness (5 items)
  - “People feel they need to do something to help me because I have a disability.”
  - Abbreviated in the simulated data as “Help#”
- Minimization (3 items)
  - “People minimize my disability or suggest that it could be worse.”
  - Abbreviated in the simulated data as “Min#”
- Denial of Personhood (5 items)
  - “People don’t see me as a whole person because I have a disability.”
  - Abbreviated in the simulated data as “Pers#”
- Otherization (7 items)
  - “People indicate that they would not date a person with a disability.”
  - Abbreviated in the simulated data as “Oth#”

In the simulation below, I use the same factor loadings from the EFA and correlations between factors for both the mild and severe groupings. I use the reported means for the mild group, and arbitrarily make them higher for the severe group.

Below I walk through the data simulation. This is not an essential portion of the lesson, but I will lecture it in case you are interested. None of the items are negatively worded (relative to the other items), so there is no need to reverse-score any items.

```
# Simulating the data for the respondents with mild disability
AMSmild_generating_model <- "
  #measurement model
  Help  =~ .74*Help1 + .75*Help2 + .65*Help3 + .58*Help4 + .62*Help5
  Minim =~ .71*Min1 + .52*Min2 + .47*Min3
  Person =~ .71*Per1 + .84*Per2 + .74*Per3 + .56*Per4 + .42*Per5
  Other =~ .89*Oth1 + .73*Oth2 + .70*Oth3 + .46*Oth4 + .41*Oth5 + .40*Oth6 + .32*Oth7

  #Means
  Help ~ 1.96*1
  Minim ~ 2.76*1
  Person ~ 1.51*1
  Other ~ 1.17*1

  #Correlations
  Help ~~ .27*Minim
```

```

    Help ~~ .66*Person
    Help ~~ .68* Other
    Minim ~~ .36*Person
    Minim ~~ .30*Other
    Person ~~ .76*Other

    ""

set.seed(240504)
AMSmild <- lavaan::simulateData(model = AMSmild_generating_model, model.type = "sem",
  meanstructure = T, sample.nobs = 548, standardized = FALSE)

# Adding a variable that denotes mild condition

AMSmild$Group <- "Mild"

AMSev_generating_model <- "
  #measurement model
  Help =~ .68*Help1 + .76*Help2 + .57*Help3 + .62*Help4 + .72*Help5
  Minim =~ .75*Min1 + .59*Min2 + .51*Min3
  Person =~ .69*Per1 + .79*Per2 + .79*Per3 + .58*Per4 + .51*Per5
  Other =~ .72*Oth1 + .71*Oth2 + .75*Oth3 + .51*Oth4 + .51*Oth5 + .36*Oth6 + .42*Oth7

  #means
  Help ~ 3.5*1
  Minim ~ 3.3*1
  Person ~ 3.01*1
  Other ~ 2.32*1

  #correlations
  Help ~~ .31*Minim
  Help ~~ .44*Person
  Help ~~ .55* Other
  Minim ~~ .40*Person
  Minim ~~ .35*Other
  Person ~~ .49*Other
  ""

set.seed(240504)
AMSev <- lavaan::simulateData(model = AMSev_generating_model, model.type = "sem",
  meanstructure = T, sample.nobs = 285, standardized = FALSE)
# used to retrieve column indices used in the rescaling script below

# Adding a variable that denotes severe condition
AMSev$Group <- "Severe"

# Binding the separate groups together in a single file

```

```

dfAMSi <- dplyr::bind_rows(AMSmild, AMSsev)

col_index <- as.data.frame(colnames(dfAMSi))

# The code below loops through each column of the dataframe and
# assigns the scaling accordingly All rows are the iBel scales,
# administrations A and B

for (i in 1:ncol(dfAMSi)) {
  if (i >= 1 & i <= 20) {
    dfAMSi[, i] <- scales::rescale(dfAMSi[, i], c(0, 5))
  }
}

# Now round to zero
library(tidyverse)
dfAMSi <- dfAMSi %>%
  mutate_if(is.numeric, round, digits = 0)

# quick check of my work psych::describe(dfAMSi)

```

The optional script below will let you save the simulated data to your computing environment as either a .csv file (think “Excel lite”) or .rds object (preserves any formatting you might do). If you save the .csv file and bring it back in, you will lose any formatting (e.g., ordered factors will be interpreted as character variables).

```

# write the simulated data as a .csv write.table(dfAMSi,
# file='dfAMSi.csv', sep=',', col.names=TRUE, row.names=FALSE) bring
# back the simulated dat from a .csv file dfAMSi <- read.csv
# ('dfAMSi.csv', header = TRUE) str(dfAMSi)

```

In this lesson I made the Severity variable a factor during the simulation. Importing the exported .csv file will lose that formating. Therefore, unless you need to use a .csv file outside of R, I recommend using the .rds file.

An .rds file preserves all formatting to variables prior to the export and re-import. If you already exported/imported the .csv file, you will need to re-run the simulation.

```

# to save the df as an .rds (think 'R object') file on your computer;
# it should save in the same file as the .rmd file you are working
# with saveRDS(dfAMSi, 'dfAMSi.rds') bring back the simulated dat
# from an .rds file
dfAMSi <- readRDS("dfAMSi.rds")

```

Let’s check the structure of the data:

```
str(dfAMSi)

'data.frame': 833 obs. of 21 variables:
 $ Help1: int 2 4 1 2 2 3 2 2 1 1 ...
 $ Help2: int 2 2 0 2 3 3 3 3 1 4 ...
 $ Help3: int 3 2 1 2 3 1 3 2 2 3 ...
 $ Help4: int 3 2 1 4 2 2 3 3 2 3 ...
 $ Help5: int 2 3 2 3 2 2 1 3 2 2 ...
 $ Min1 : int 2 4 1 2 3 3 3 3 3 3 ...
 $ Min2 : int 3 4 1 2 2 4 3 2 3 2 ...
 $ Min3 : int 3 3 3 1 2 3 2 3 4 3 ...
 $ Per1 : int 2 2 1 2 1 3 2 2 2 2 ...
 $ Per2 : int 3 3 1 2 2 3 3 3 1 2 ...
 $ Per3 : int 3 3 2 3 2 3 2 3 2 3 ...
 $ Per4 : int 3 4 1 2 3 2 3 2 3 2 ...
 $ Per5 : int 3 4 2 1 2 2 2 3 2 3 ...
 $ Oth1 : int 3 3 2 3 4 3 4 3 2 3 ...
 $ Oth2 : int 2 2 2 4 3 3 2 3 3 3 ...
 $ Oth3 : int 2 2 1 3 4 3 3 2 2 2 ...
 $ Oth4 : int 2 3 2 3 2 2 1 3 1 3 ...
 $ Oth5 : int 3 2 2 2 2 3 2 3 2 2 ...
 $ Oth6 : int 3 2 2 3 4 2 3 3 2 2 ...
 $ Oth7 : int 2 2 2 3 3 2 3 3 3 3 ...
 $ Group: chr "Mild" "Mild" "Mild" "Mild" ...
```

We need “Group” to be a factor.

```
dfAMSi[, "Group"] <- as.factor(dfAMSi[, "Group"])
```

## 12.4 Whole-Group and Baseline Analyses

Conover et al.[2017] conducted the invariance testing with the four-factor, correlated factors model. Let’s start by simply by creating an overall measurement model from the dataset without regard to group membership.

### 12.4.1 Whole Group CFA

With the number of items per scale ranging from 3 to 7 on this multidimensional, first-order, factor structure we are sufficiently *identified*. Remember, rule is at least 3 items/indicators per factor for unidimensional scales and 2 items/indicators per factor for a multidimensional scale.

```
set.seed(240504)
AMS4CorrMod <- "
    Helplessness =~ Help1 + Help2 + Help3 + Help4 + Help5
```

```

Minimization =~ Min1 + Min2 + Min3
DenialPersonhood =~ Per1 + Per2 + Per3 + Per4 + Per5
Otherization =~ Oth1 + Oth2 + Oth3 + Oth4 + Oth5 + Oth6 + Oth7
"
set.seed(240504)
AMS4CorrFit <- lavaan::cfa(AMS4CorrMod, data = dfAMSi)
lavaan::summary(AMS4CorrFit, fit.measures = TRUE, standardized = TRUE)

```

lavaan 0.6.17 ended normally after 48 iterations

Estimator	ML
Optimization method	NLMINB
Number of model parameters	46
Number of observations	833

#### Model Test User Model:

Test statistic	155.238
Degrees of freedom	164
P-value (Chi-square)	0.676

#### Model Test Baseline Model:

Test statistic	3297.768
Degrees of freedom	190
P-value	0.000

#### User Model versus Baseline Model:

Comparative Fit Index (CFI)	1.000
Tucker-Lewis Index (TLI)	1.003

#### Loglikelihood and Information Criteria:

Loglikelihood user model (H0)	-19314.357
Loglikelihood unrestricted model (H1)	-19236.738
Akaike (AIC)	38720.715
Bayesian (BIC)	38938.066
Sample-size adjusted Bayesian (SABIC)	38791.986

#### Root Mean Square Error of Approximation:

RMSEA	0.000
90 Percent confidence interval - lower	0.000
90 Percent confidence interval - upper	0.013

P-value H_0: RMSEA <= 0.050	1.000
P-value H_0: RMSEA >= 0.080	0.000

Standardized Root Mean Square Residual:

SRMR	0.023
------	-------

Parameter Estimates:

Standard errors	Standard
Information	Expected
Information saturated (h1) model	Structured

Latent Variables:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
<b>Helplessness =~</b>						
Help1	1.000				0.433	0.549
Help2	1.338	0.102	13.082	0.000	0.579	0.658
Help3	1.186	0.098	12.049	0.000	0.514	0.574
Help4	1.225	0.099	12.353	0.000	0.531	0.597
Help5	1.401	0.107	13.081	0.000	0.607	0.657
<b>Minimization =~</b>						
Min1	1.000				0.452	0.545
Min2	0.825	0.121	6.825	0.000	0.373	0.455
Min3	0.734	0.112	6.560	0.000	0.332	0.411
<b>DenialPersonhood =~</b>						
Per1	1.000				0.532	0.590
Per2	1.043	0.074	14.073	0.000	0.555	0.675
Per3	1.052	0.075	14.012	0.000	0.559	0.670
Per4	0.810	0.069	11.774	0.000	0.431	0.521
Per5	0.783	0.072	10.817	0.000	0.416	0.467
<b>Otherization =~</b>						
Oth1	1.000				0.553	0.629
Oth2	0.875	0.067	13.058	0.000	0.484	0.582
Oth3	0.946	0.070	13.527	0.000	0.523	0.611
Oth4	0.807	0.070	11.486	0.000	0.446	0.494
Oth5	0.622	0.059	10.581	0.000	0.344	0.448
Oth6	0.571	0.060	9.466	0.000	0.316	0.394
Oth7	0.621	0.063	9.897	0.000	0.344	0.414

Covariances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
<b>Helplessness ~~</b>						
Minimization	0.079	0.014	5.829	0.000	0.406	0.406
DenialPersonhd	0.176	0.018	9.754	0.000	0.764	0.764
Otherization	0.163	0.017	9.520	0.000	0.681	0.681
<b>Minimization ~~</b>						
DenialPersonhd	0.116	0.018	6.590	0.000	0.481	0.481

Otherization	0.126	0.018	6.838	0.000	0.503	0.503
DenialPersonhood ~~						
Otherization	0.205	0.021	9.909	0.000	0.698	0.698

Variances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
.Help1	0.434	0.024	18.210	0.000	0.434	0.698
.Help2	0.441	0.027	16.490	0.000	0.441	0.568
.Help3	0.537	0.030	17.903	0.000	0.537	0.671
.Help4	0.509	0.029	17.580	0.000	0.509	0.644
.Help5	0.483	0.029	16.493	0.000	0.483	0.568
.Min1	0.483	0.038	12.809	0.000	0.483	0.703
.Min2	0.532	0.034	15.739	0.000	0.532	0.793
.Min3	0.542	0.032	16.829	0.000	0.542	0.831
.Per1	0.531	0.030	17.657	0.000	0.531	0.652
.Per2	0.367	0.023	16.032	0.000	0.367	0.544
.Per3	0.383	0.024	16.150	0.000	0.383	0.551
.Per4	0.498	0.027	18.497	0.000	0.498	0.729
.Per5	0.620	0.033	18.981	0.000	0.620	0.782
.Oth1	0.468	0.028	16.664	0.000	0.468	0.605
.Oth2	0.456	0.026	17.482	0.000	0.456	0.661
.Oth3	0.459	0.027	16.998	0.000	0.459	0.626
.Oth4	0.615	0.033	18.574	0.000	0.615	0.756
.Oth5	0.471	0.025	18.987	0.000	0.471	0.799
.Oth6	0.544	0.028	19.371	0.000	0.544	0.845
.Oth7	0.570	0.030	19.236	0.000	0.570	0.828
Helplessness	0.187	0.024	7.710	0.000	1.000	1.000
Minimization	0.204	0.038	5.397	0.000	1.000	1.000
DenialPersonhd	0.283	0.034	8.375	0.000	1.000	1.000
Otherization	0.306	0.034	8.925	0.000	1.000	1.000

Among my first steps are also to write the code to export the results. The *tidySEM* package has useful functions to export the fit statistics, parameter estimates, and correlations among the latent variables (i.e., factors).

```
AllFitStats <- tidySEM:::table_fit(AMS4CorrFit)
```

Registered S3 method overwritten by 'tidySEM':

```
method      from
predict.MxModel OpenMx
```

```
All_paramEsts <- tidySEM:::table_results(AMS4CorrFit, digits = 3, columns = NULL)
AllCorrs <- tidySEM:::table_corrs(AMS4CorrFit, digits = 3)
# to see each of the tables, remove the hashtag AllFitStats
# All_paramEsts AllCorrs
```

Next, I export them.

```
write.csv(AllFitStats, file = "AllFitStats.csv")
write.csv(All_paramEsts, file = "All_paramEsts.csv")
write.csv(AllCorrs, file = "AllCorrs.csv")
```

### 12.4.2 Interpreting the Output

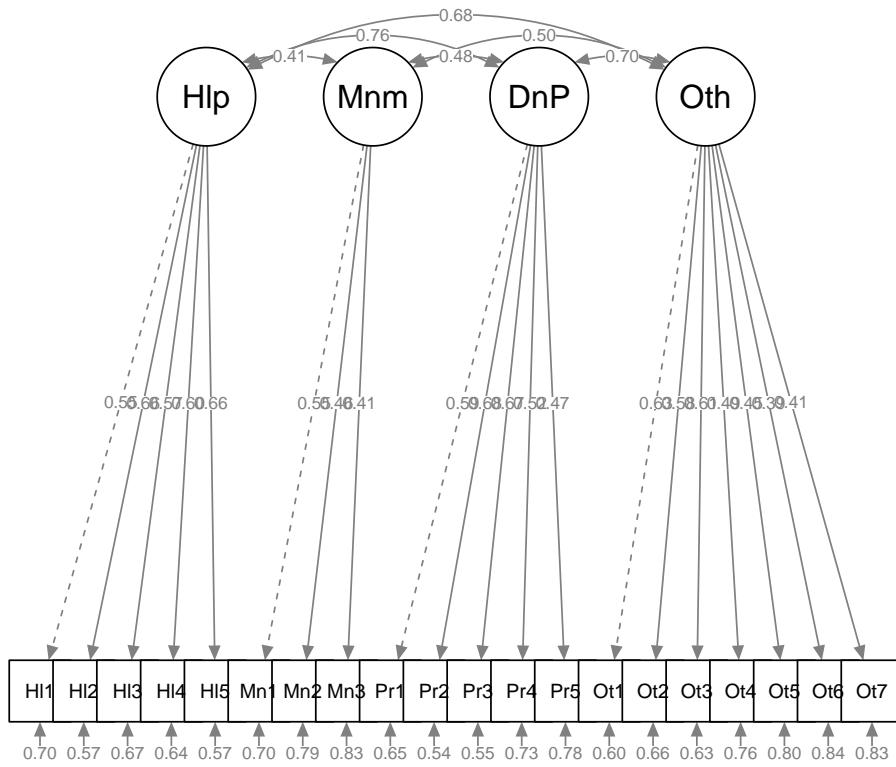
Criteria	Our Results	Criteria met?
Factor loadings significant, strong, proper valence	Help: 0.55 to 0.66; Min: 0.41 to 0.55; Pers: 0.47 to 0.68 Oth: 0.39 to 0.63	Yes
Non-significant chi-square	$\chi^2(164) = 155.24, p = 0.676$	Yes
$CFI \geq .95$	CFI = 1.000	Yes
$RMSEA \leq .05$ (but definitely < .10)	RMSEA = 0.000, 90%CI(0.000, 0.013)	Yes
$SRMR \leq .08$ (but definitely < .10)	SRMR = 0.023	Yes
Combination rule: $CFI \geq .95$ and $SRMR \leq .08$	CFI = 1.000, SRS = 0.023	Yes

### 12.4.3 Partial Write-up

**Correlated factors model for all in sample.** The model where factors were free to covary demonstrated the following fit to the  $\text{data}:\$/\text{chi}^2(164) = 155.24, p = 0.676$ , CFI = 1.000, RMSEA = 0.000, 90%CI(0.000, 0.013), SRMR = 0.023. Factor loadings ranged from 0.55 to 0.66 for the Helplessness scale, 0.41 to 0.55 for the Minimization scale, 0.47 to 0.68 for the Denial of Personhood scale, and 0.39 to 0.63 for the Otherization scale.

Producing a figure can be useful to represent what we did to others as well as checking our own work. That is, “Did we think we did what we intended?” When the \*what = “col”, whatLabels = “stand”) combination is shown, paths that are “fixed” are represented by dashed lines. Below, we expect to see each the four factors predicting only the items associated with their factor, one item for each factor (the first on the left) should be specified as the indicator variable (and represented with a dashed line), and the factors/latent variables should not be freed to covary (i.e., an uncorrelated traits or orthogonal model). Because they are “fixed” to be 0.00, they will be represented with dashed curves with double-headed arrows.

```
# displays standardized pattern coefficients
semPlot::semPaths(AMS4CorrFit, layout = "tree", style = "lisrel", what = "col",
  whatLabels = "stand")
```



```
# displays estimates/unstandardized regression weights
# semPlot::semPaths(AMS4CorrFit, layout = 'tree', style = 'lisrel',
# what = 'col', whatLabels = 'est')
```

Our fit is fairly similar to what Conover et al. reported in their article. Specifically, their four-factor, correlated factors model, had a statistically significant chi-square. Regarding fit: CFI = .89, SRMR = .07, and RMSEA = .07 CI90% (.06, .07). As researchers, they were satisfied with the result and they asked the question, “Is measure invariant across disability severity.” A first (but not complete) step is to evaluate the model, separately for the groups of interest. In their case it was mild (where they combined mild and moderate levels of severity) and severe (combining severe and very severe levels).

#### 12.4.4 Baseline Model when Severity = Mild

Let's start by subsetting the data.

```
mild_df <- subset(dfAMSi, Group == "Mild")
severe_df <- subset(dfAMSi, Group == "Severe")
```

Let's run the CFA model for those participants whose data were classified as "mild."

```
set.seed(240504)
MildFit <- lavaan::cfa(AMS4CorrMod, data = mild_df)
lavaan::summary(MildFit, fit.measures = TRUE, standardized = TRUE)
```

`lavaan 0.6.17 ended normally after 46 iterations`

Estimator	ML
Optimization method	NLMINB
Number of model parameters	46
Number of observations	548

Model Test User Model:

Test statistic	165.227
Degrees of freedom	164
P-value (Chi-square)	0.458

Model Test Baseline Model:

Test statistic	1561.967
Degrees of freedom	190
P-value	0.000

User Model versus Baseline Model:

Comparative Fit Index (CFI)	0.999
Tucker-Lewis Index (TLI)	0.999

Loglikelihood and Information Criteria:

Loglikelihood user model (H0)	-12518.845
Loglikelihood unrestricted model (H1)	-12436.231
Akaike (AIC)	25129.689
Bayesian (BIC)	25327.778
Sample-size adjusted Bayesian (SABIC)	25181.755

Root Mean Square Error of Approximation:

RMSEA	0.004
90 Percent confidence interval - lower	0.000

90 Percent confidence interval - upper	0.020
P-value H_0: RMSEA <= 0.050	1.000
P-value H_0: RMSEA >= 0.080	0.000

Standardized Root Mean Square Residual:

SRMR	0.031
------	-------

Parameter Estimates:

Standard errors	Standard
Information	Expected
Information saturated (h1) model	Structured

Latent Variables:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
<b>Helplessness =~</b>						
Help1	1.000				0.435	0.574
Help2	1.066	0.116	9.227	0.000	0.464	0.581
Help3	0.966	0.115	8.382	0.000	0.420	0.499
Help4	0.857	0.110	7.803	0.000	0.373	0.451
Help5	0.959	0.113	8.473	0.000	0.417	0.507
<b>Minimization =~</b>						
Min1	1.000				0.529	0.639
Min2	0.550	0.137	4.000	0.000	0.291	0.358
Min3	0.509	0.129	3.934	0.000	0.269	0.339
<b>DenialPersonhood =~</b>						
Per1	1.000				0.422	0.501
Per2	1.254	0.132	9.487	0.000	0.530	0.691
Per3	1.082	0.121	8.912	0.000	0.457	0.597
Per4	0.736	0.105	7.007	0.000	0.311	0.405
Per5	0.736	0.110	6.717	0.000	0.311	0.382
<b>Otherization =~</b>						
Oth1	1.000				0.554	0.645
Oth2	0.811	0.084	9.691	0.000	0.449	0.542
Oth3	0.819	0.083	9.892	0.000	0.454	0.558
Oth4	0.662	0.084	7.835	0.000	0.367	0.417
Oth5	0.511	0.073	7.040	0.000	0.283	0.369
Oth6	0.545	0.076	7.176	0.000	0.302	0.377
Oth7	0.448	0.073	6.104	0.000	0.248	0.315

Covariances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
<b>Helplessness ~~</b>						
Minimization	0.057	0.018	3.143	0.002	0.249	0.249
DenialPersonhd	0.128	0.019	6.889	0.000	0.699	0.699
Otherization	0.153	0.021	7.266	0.000	0.634	0.634
<b>Minimization ~~</b>						

DenialPersonhd	0.088	0.019	4.612	0.000	0.396	0.396
Otherization	0.098	0.023	4.201	0.000	0.335	0.335
<b>DenialPersonhood ~~</b>						
Otherization	0.165	0.023	7.268	0.000	0.705	0.705

Variances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
.Help1	0.384	0.029	13.388	0.000	0.384	0.670
.Help2	0.423	0.032	13.286	0.000	0.423	0.663
.Help3	0.533	0.037	14.441	0.000	0.533	0.751
.Help4	0.543	0.036	14.930	0.000	0.543	0.796
.Help5	0.503	0.035	14.349	0.000	0.503	0.743
.Min1	0.406	0.071	5.741	0.000	0.406	0.592
.Min2	0.575	0.041	13.887	0.000	0.575	0.872
.Min3	0.558	0.039	14.255	0.000	0.558	0.885
.Per1	0.531	0.036	14.736	0.000	0.531	0.749
.Per2	0.306	0.027	11.455	0.000	0.306	0.522
.Per3	0.378	0.028	13.521	0.000	0.378	0.644
.Per4	0.493	0.032	15.501	0.000	0.493	0.836
.Per5	0.566	0.036	15.637	0.000	0.566	0.854
.Oth1	0.431	0.035	12.460	0.000	0.431	0.584
.Oth2	0.484	0.034	14.178	0.000	0.484	0.706
.Oth3	0.456	0.033	13.976	0.000	0.456	0.689
.Oth4	0.640	0.042	15.366	0.000	0.640	0.826
.Oth5	0.509	0.033	15.667	0.000	0.509	0.864
.Oth6	0.553	0.035	15.621	0.000	0.553	0.858
.Oth7	0.561	0.035	15.935	0.000	0.561	0.901
Helplessness	0.189	0.031	6.153	0.000	1.000	1.000
Minimization	0.280	0.074	3.774	0.000	1.000	1.000
DenialPersonhd	0.178	0.033	5.434	0.000	1.000	1.000
Otherization	0.307	0.043	7.182	0.000	1.000	1.000

Not surprisingly, our results are similar to the total group. I notice that the pattern coefficients wiggle around a little more (one as low as .13) but that the fit indices seem a little stronger.

Criteria	Mild	Severe
Factor loadings: Help	0.451 to 0.581	
Factor loadings: Min	0.339 to 0.639	
Factor loadings: Pers	0.382 to 0.691	
Factor loadings: Oth	0.315 to 0.645	
Non-significant chi-square	$p = 0.458$	
$CFI \geq .95$	$CFI = 0.999$	
$SRMR \leq .08$ (but definitely $< .10$ )	$SRMR = 0.031$	
$RMSEA \leq .05$ (but definitely $< .10$ )	$RMSEA = 0.004, 90\% CI(0.000, 0.020)$	

### 12.4.5 Baseline Model when Severity = Severe

Let's run the CFA model again for those participants whose data were classified as "severe."

```
set.seed(240504)
SevereFit <- lavaan::cfa(AMS4CorrMod, data = severe_df)
lavaan::summary(SevereFit, fit.measures = TRUE, standardized = TRUE)
```

lavaan 0.6.17 ended normally after 57 iterations

Estimator	ML
Optimization method	NLMINB
Number of model parameters	46
Number of observations	285

Model Test User Model:

Test statistic	162.345
Degrees of freedom	164
P-value (Chi-square)	0.522

Model Test Baseline Model:

Test statistic	892.916
Degrees of freedom	190
P-value	0.000

User Model versus Baseline Model:

Comparative Fit Index (CFI)	1.000
Tucker-Lewis Index (TLI)	1.003

Loglikelihood and Information Criteria:

Loglikelihood user model (H0)	-6567.990
Loglikelihood unrestricted model (H1)	-6486.817
Akaike (AIC)	13227.980
Bayesian (BIC)	13395.994
Sample-size adjusted Bayesian (SABIC)	13250.125

Root Mean Square Error of Approximation:

RMSEA	0.000
90 Percent confidence interval - lower	0.000
90 Percent confidence interval - upper	0.026

P-value H_0: RMSEA <= 0.050	1.000
P-value H_0: RMSEA >= 0.080	0.000

Standardized Root Mean Square Residual:

SRMR	0.044
------	-------

Parameter Estimates:

Standard errors	Standard
Information	Expected
Information saturated (h1) model	Structured

Latent Variables:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
<b>Helplessness =~</b>						
Help1	1.000				0.314	0.408
Help2	1.454	0.285	5.096	0.000	0.457	0.553
Help3	1.448	0.290	5.001	0.000	0.455	0.527
Help4	1.313	0.262	5.007	0.000	0.412	0.529
Help5	1.741	0.325	5.352	0.000	0.546	0.642
<b>Minimization =~</b>						
Min1	1.000				0.322	0.407
Min2	1.151	0.317	3.633	0.000	0.371	0.472
Min3	1.331	0.362	3.676	0.000	0.429	0.527
<b>DenialPersonhood =~</b>						
Per1	1.000				0.488	0.558
Per2	0.895	0.145	6.180	0.000	0.437	0.544
Per3	1.163	0.171	6.795	0.000	0.568	0.687
Per4	0.653	0.130	5.036	0.000	0.319	0.403
Per5	0.647	0.144	4.491	0.000	0.316	0.349
<b>Otherization =~</b>						
Oth1	1.000				0.542	0.621
Oth2	0.823	0.114	7.202	0.000	0.446	0.578
Oth3	0.807	0.119	6.811	0.000	0.438	0.534
Oth4	0.838	0.127	6.619	0.000	0.454	0.514
Oth5	0.548	0.097	5.663	0.000	0.297	0.423
Oth6	0.566	0.106	5.325	0.000	0.307	0.393
Oth7	0.672	0.117	5.738	0.000	0.364	0.429

Covariances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
<b>Helplessness ~~</b>						
Minimization	0.026	0.012	2.100	0.036	0.259	0.259
DenialPersonhd	0.083	0.021	3.972	0.000	0.543	0.543
Otherization	0.091	0.022	4.123	0.000	0.535	0.535
<b>Minimization ~~</b>						
DenialPersonhd	0.047	0.020	2.403	0.016	0.299	0.299

Otherization	0.091	0.026	3.464	0.001	0.522	0.522
DenialPersonhood ~~						
Otherization	0.126	0.029	4.372	0.000	0.476	0.476

Variances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
.Help1	0.494	0.045	10.995	0.000	0.494	0.834
.Help2	0.474	0.048	9.831	0.000	0.474	0.694
.Help3	0.536	0.053	10.097	0.000	0.536	0.722
.Help4	0.437	0.043	10.083	0.000	0.437	0.720
.Help5	0.426	0.050	8.568	0.000	0.426	0.588
.Min1	0.522	0.054	9.754	0.000	0.522	0.834
.Min2	0.481	0.055	8.742	0.000	0.481	0.778
.Min3	0.477	0.063	7.638	0.000	0.477	0.722
.Per1	0.527	0.055	9.660	0.000	0.527	0.688
.Per2	0.455	0.046	9.829	0.000	0.455	0.704
.Per3	0.361	0.048	7.469	0.000	0.361	0.528
.Per4	0.523	0.048	10.983	0.000	0.523	0.837
.Per5	0.722	0.064	11.260	0.000	0.722	0.878
.Oth1	0.467	0.050	9.390	0.000	0.467	0.614
.Oth2	0.397	0.040	9.910	0.000	0.397	0.666
.Oth3	0.480	0.046	10.329	0.000	0.480	0.715
.Oth4	0.575	0.055	10.493	0.000	0.575	0.736
.Oth5	0.406	0.037	11.066	0.000	0.406	0.821
.Oth6	0.516	0.046	11.207	0.000	0.516	0.846
.Oth7	0.587	0.053	11.031	0.000	0.587	0.816
Helplessness	0.099	0.033	3.029	0.002	1.000	1.000
Minimization	0.104	0.042	2.455	0.014	1.000	1.000
DenialPersonhd	0.238	0.056	4.223	0.000	1.000	1.000
Otherization	0.294	0.059	4.985	0.000	1.000	1.000

Our visual inspection of the similarity of psychometric characteristics suggests that the measure is functioning similarly across the two levels of severity.

Criteria	Mild	Severe
Factor loadings: Help	0.451 to 0.581	0.408 to 0.642
Factor loadings: Min	0.339 to 0.639	0.407 to 0.527
Factor loadings: Pers	0.382 to 0.691	0.349 to 0.687
Factor loadings: Oth	0.315 to 0.645	0.393 to 0.621
Non-significant chi-square	$p = 0.458$	$p = 0.522$
$CFI \geq .95$	$CFI = 0.999$	$CFI = 1.000$
$SRMR \leq .08$ (but definitely $< .10$ )	$SRMR = 0.044$	
$RMSEA \leq .05$ (but definitely $< .10$ )	$RMSEA = 0.004,$ $90\%CI(0.000, 0.026)$	

This, though, does not constitute a formal evaluation. Thus, we continue with testing for multigroup invariance.

## 12.5 Configural Invariance

Configural invariance is our least restrictive level. We are essentially specifying ONE STRUCTURE – four correlated factors, each with 3 to 7 items/indicators. Each model is allowed to have its own loadings, error variances, and so forth. It's only the structure (the *configuration*) that is consistent.

The same model we had before works. We create the configural model simply by specifying *group* = “Severity” in the *cfa()* function.

```
set.seed(240504)
configural <- lavaan::cfa(AMS4CorrMod, data = dfAMSi, group = "Group")
lavaan::summary(configural, fit.measures = TRUE, standardized = TRUE)
```

lavaan 0.6.17 ended normally after 87 iterations

Estimator	ML
Optimization method	NLMINB
Number of model parameters	132

Number of observations per group:	
Mild	548
Severe	285

Model Test User Model:

Test statistic	327.572
Degrees of freedom	328
P-value (Chi-square)	0.496
Test statistic for each group:	
Mild	165.227
Severe	162.345

Model Test Baseline Model:

Test statistic	2454.883
Degrees of freedom	380
P-value	0.000

User Model versus Baseline Model:

Comparative Fit Index (CFI)	1.000
Tucker-Lewis Index (TLI)	1.000

Loglikelihood and Information Criteria:

Loglikelihood user model (H0)	-19086.834
Loglikelihood unrestricted model (H1)	-18923.048

Akaike (AIC)	38437.669
Bayesian (BIC)	39061.373
Sample-size adjusted Bayesian (SABIC)	38642.187

Root Mean Square Error of Approximation:

RMSEA	0.000
90 Percent confidence interval - lower	0.000
90 Percent confidence interval - upper	0.018
P-value H_0: RMSEA <= 0.050	1.000
P-value H_0: RMSEA >= 0.080	0.000

Standardized Root Mean Square Residual:

SRMR	0.034
------	-------

Parameter Estimates:

Standard errors	Standard
Information	Expected
Information saturated (h1) model	Structured

Group 1 [Mild]:

Latent Variables:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
Helplessness =~						
Help1	1.000				0.435	0.574
Help2	1.066	0.116	9.227	0.000	0.464	0.581
Help3	0.966	0.115	8.382	0.000	0.420	0.499
Help4	0.857	0.110	7.803	0.000	0.373	0.451
Help5	0.959	0.113	8.473	0.000	0.417	0.507
Minimization =~						
Min1	1.000				0.529	0.639
Min2	0.550	0.137	4.000	0.000	0.291	0.358
Min3	0.509	0.129	3.934	0.000	0.269	0.339
DenialPersonhood =~						
Per1	1.000				0.422	0.501
Per2	1.254	0.132	9.487	0.000	0.530	0.691
Per3	1.082	0.121	8.913	0.000	0.457	0.597
Per4	0.736	0.105	7.007	0.000	0.311	0.405
Per5	0.736	0.110	6.717	0.000	0.311	0.382
Otherization =~						
Oth1	1.000				0.554	0.645
Oth2	0.811	0.084	9.691	0.000	0.449	0.542
Oth3	0.819	0.083	9.892	0.000	0.454	0.558
Oth4	0.662	0.084	7.835	0.000	0.367	0.417

Oth5	0.511	0.073	7.040	0.000	0.283	0.369
Oth6	0.545	0.076	7.176	0.000	0.302	0.377
Oth7	0.448	0.073	6.104	0.000	0.248	0.315

## Covariances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
Helplessness ~~						
Minimization	0.057	0.018	3.143	0.002	0.249	0.249
DenialPersonhd	0.128	0.019	6.889	0.000	0.699	0.699
Otherization	0.153	0.021	7.266	0.000	0.634	0.634
Minimization ~~						
DenialPersonhd	0.088	0.019	4.612	0.000	0.396	0.396
Otherization	0.098	0.023	4.201	0.000	0.335	0.335
DenialPersonhood ~~						
Otherization	0.165	0.023	7.269	0.000	0.705	0.705

## Intercepts:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
.Help1	1.978	0.032	61.199	0.000	1.978	2.614
.Help2	2.400	0.034	70.342	0.000	2.400	3.005
.Help3	2.137	0.036	59.392	0.000	2.137	2.537
.Help4	2.226	0.035	63.090	0.000	2.226	2.695
.Help5	2.020	0.035	57.490	0.000	2.020	2.456
.Min1	2.401	0.035	67.896	0.000	2.401	2.900
.Min2	2.343	0.035	67.532	0.000	2.343	2.885
.Min3	2.631	0.034	77.574	0.000	2.631	3.314
.Per1	2.086	0.036	57.955	0.000	2.086	2.476
.Per2	2.109	0.033	64.478	0.000	2.109	2.754
.Per3	2.553	0.033	78.028	0.000	2.553	3.333
.Per4	2.381	0.033	72.580	0.000	2.381	3.100
.Per5	2.148	0.035	61.774	0.000	2.148	2.639
.Oth1	2.673	0.037	72.839	0.000	2.673	3.112
.Oth2	2.628	0.035	74.257	0.000	2.628	3.172
.Oth3	2.117	0.035	60.924	0.000	2.117	2.603
.Oth4	2.407	0.038	64.037	0.000	2.407	2.736
.Oth5	2.489	0.033	75.903	0.000	2.489	3.242
.Oth6	2.620	0.034	76.425	0.000	2.620	3.265
.Oth7	2.104	0.034	62.432	0.000	2.104	2.667

## Variances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
.Help1	0.384	0.029	13.388	0.000	0.384	0.670
.Help2	0.423	0.032	13.286	0.000	0.423	0.663
.Help3	0.533	0.037	14.441	0.000	0.533	0.751
.Help4	0.543	0.036	14.930	0.000	0.543	0.796
.Help5	0.503	0.035	14.349	0.000	0.503	0.743
.Min1	0.406	0.071	5.741	0.000	0.406	0.592
.Min2	0.575	0.041	13.887	0.000	0.575	0.872

.Min3	0.558	0.039	14.255	0.000	0.558	0.885
.Per1	0.531	0.036	14.736	0.000	0.531	0.749
.Per2	0.306	0.027	11.455	0.000	0.306	0.522
.Per3	0.378	0.028	13.521	0.000	0.378	0.644
.Per4	0.493	0.032	15.501	0.000	0.493	0.836
.Per5	0.566	0.036	15.637	0.000	0.566	0.854
.Oth1	0.431	0.035	12.460	0.000	0.431	0.584
.Oth2	0.484	0.034	14.178	0.000	0.484	0.706
.Oth3	0.456	0.033	13.976	0.000	0.456	0.689
.Oth4	0.640	0.042	15.366	0.000	0.640	0.826
.Oth5	0.509	0.033	15.667	0.000	0.509	0.864
.Oth6	0.553	0.035	15.621	0.000	0.553	0.858
.Oth7	0.561	0.035	15.935	0.000	0.561	0.901
Helplessness	0.189	0.031	6.153	0.000	1.000	1.000
Minimization	0.280	0.074	3.774	0.000	1.000	1.000
DenialPersonhd	0.178	0.033	5.434	0.000	1.000	1.000
Otherization	0.307	0.043	7.182	0.000	1.000	1.000

Group 2 [Severe]:

Latent Variables:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
Helplessness =~						
Help1	1.000				0.314	0.408
Help2	1.454	0.285	5.096	0.000	0.457	0.553
Help3	1.448	0.290	5.001	0.000	0.455	0.527
Help4	1.313	0.262	5.007	0.000	0.412	0.529
Help5	1.741	0.325	5.352	0.000	0.546	0.642
Minimization =~						
Min1	1.000				0.322	0.407
Min2	1.151	0.317	3.633	0.000	0.371	0.472
Min3	1.331	0.362	3.676	0.000	0.429	0.527
DenialPersonhood =~						
Per1	1.000				0.488	0.558
Per2	0.895	0.145	6.180	0.000	0.437	0.544
Per3	1.163	0.171	6.796	0.000	0.568	0.687
Per4	0.653	0.130	5.036	0.000	0.319	0.403
Per5	0.647	0.144	4.491	0.000	0.316	0.349
Otherization =~						
Oth1	1.000				0.542	0.621
Oth2	0.823	0.114	7.202	0.000	0.446	0.578
Oth3	0.807	0.119	6.811	0.000	0.438	0.534
Oth4	0.838	0.127	6.619	0.000	0.454	0.514
Oth5	0.548	0.097	5.663	0.000	0.297	0.423
Oth6	0.566	0.106	5.325	0.000	0.307	0.393
Oth7	0.672	0.117	5.738	0.000	0.364	0.429

## Covariances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
<b>Helplessness ~~</b>						
Minimization	0.026	0.012	2.100	0.036	0.259	0.259
DenialPersonhd	0.083	0.021	3.972	0.000	0.543	0.543
Otherization	0.091	0.022	4.123	0.000	0.535	0.535
<b>Minimization ~~</b>						
DenialPersonhd	0.047	0.020	2.403	0.016	0.299	0.299
Otherization	0.091	0.026	3.464	0.001	0.522	0.522
<b>DenialPersonhood ~~</b>						
Otherization	0.126	0.029	4.372	0.000	0.476	0.476

## Intercepts:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
.Help1	2.411	0.046	52.851	0.000	2.411	3.131
.Help2	3.140	0.049	64.193	0.000	3.140	3.802
.Help3	2.733	0.051	53.536	0.000	2.733	3.171
.Help4	2.996	0.046	64.929	0.000	2.996	3.846
.Help5	2.860	0.050	56.731	0.000	2.860	3.360
.Min1	2.712	0.047	57.873	0.000	2.712	3.428
.Min2	2.677	0.047	57.464	0.000	2.677	3.404
.Min3	2.849	0.048	59.141	0.000	2.849	3.503
.Per1	2.698	0.052	52.077	0.000	2.698	3.085
.Per2	2.660	0.048	55.883	0.000	2.660	3.310
.Per3	3.133	0.049	63.960	0.000	3.133	3.789
.Per4	2.986	0.047	63.795	0.000	2.986	3.779
.Per5	2.730	0.054	50.838	0.000	2.730	3.011
.Oth1	3.025	0.052	58.540	0.000	3.025	3.468
.Oth2	3.021	0.046	66.060	0.000	3.021	3.913
.Oth3	2.667	0.049	54.944	0.000	2.667	3.255
.Oth4	2.818	0.052	53.832	0.000	2.818	3.189
.Oth5	2.867	0.042	68.820	0.000	2.867	4.077
.Oth6	2.842	0.046	61.425	0.000	2.842	3.638
.Oth7	2.481	0.050	49.362	0.000	2.481	2.924

## Variances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
.Help1	0.494	0.045	10.995	0.000	0.494	0.834
.Help2	0.474	0.048	9.831	0.000	0.474	0.694
.Help3	0.536	0.053	10.098	0.000	0.536	0.722
.Help4	0.437	0.043	10.083	0.000	0.437	0.720
.Help5	0.426	0.050	8.568	0.000	0.426	0.588
.Min1	0.522	0.054	9.754	0.000	0.522	0.834
.Min2	0.481	0.055	8.742	0.000	0.481	0.778
.Min3	0.477	0.063	7.638	0.000	0.477	0.722
.Per1	0.527	0.055	9.660	0.000	0.527	0.688
.Per2	0.455	0.046	9.829	0.000	0.455	0.704
.Per3	0.361	0.048	7.469	0.000	0.361	0.528

.Per4	0.523	0.048	10.983	0.000	0.523	0.837
.Per5	0.722	0.064	11.260	0.000	0.722	0.878
.Oth1	0.467	0.050	9.390	0.000	0.467	0.614
.Oth2	0.397	0.040	9.910	0.000	0.397	0.666
.Oth3	0.480	0.046	10.329	0.000	0.480	0.715
.Oth4	0.575	0.055	10.493	0.000	0.575	0.736
.Oth5	0.406	0.037	11.066	0.000	0.406	0.821
.Oth6	0.516	0.046	11.207	0.000	0.516	0.846
.Oth7	0.587	0.053	11.031	0.000	0.587	0.816
Helplessness	0.099	0.033	3.029	0.002	1.000	1.000
Minimization	0.104	0.042	2.455	0.014	1.000	1.000
DenialPersonhd	0.238	0.056	4.223	0.000	1.000	1.000
Otherization	0.294	0.059	4.985	0.000	1.000	1.000

Let's format these results into tables.

```
ConfigFitStats <- tidySEM::table_fit(configural)
Config_paramEsts <- tidySEM::table_results(configural, digits = 3, columns = NULL)
ConfigCorrs <- tidySEM::table_corrs(configural, digits = 3)
# to see each of the tables, remove the hashtag ConfigFitStats
# Config_paramEsts ConfigCorrs
```

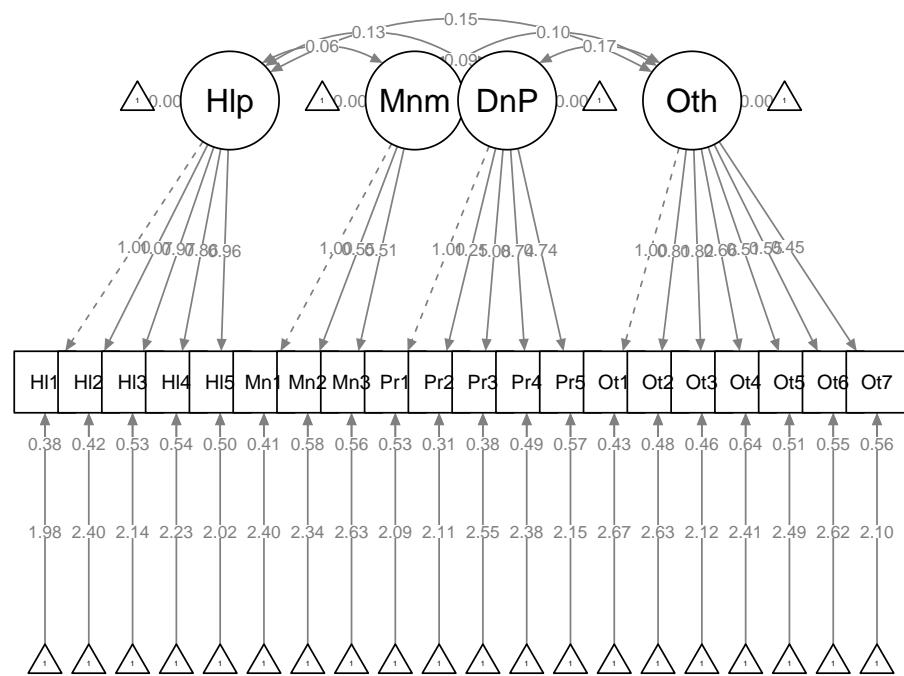
Then, export them.

```
write.csv(ConfigFitStats, file = "ConfigFitStats.csv")
write.csv(Config_paramEsts, file = "Config_paramEsts.csv")
write.csv(ConfigCorrs, file = "ConfigCorrs.csv")
```

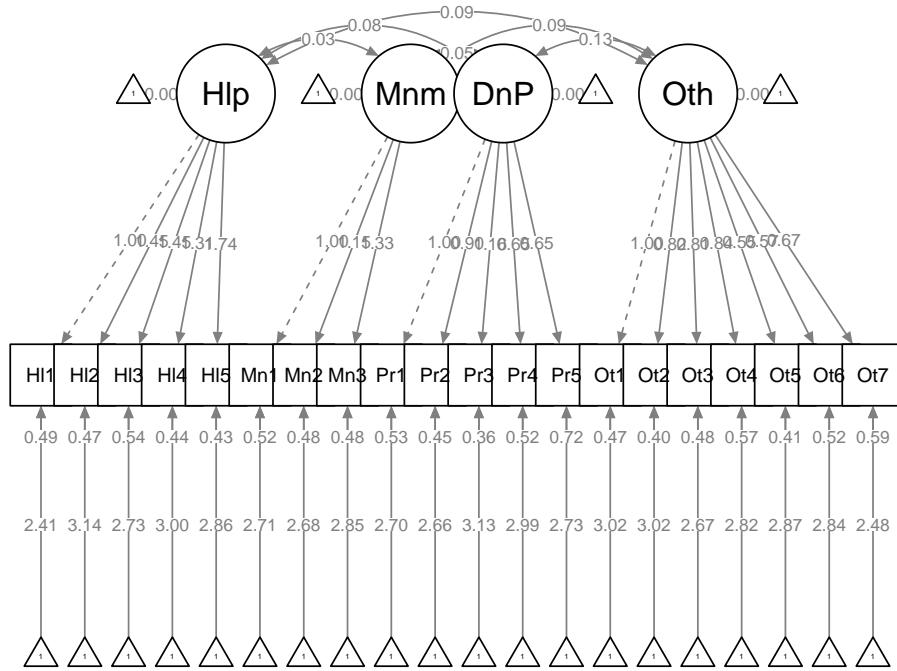
Examining the plots can help us understand what we've just done. This will result in two tables, one for each of the models. Recall, we are requiring the structure to be the same, but allowing the values to vary.

```
# semPlot::semPaths(configural, layout = 'tree', style = 'lisrel',
# what = 'col', whatLabels = 'stand')
semPlot::semPaths(configural, layout = "tree", style = "lisrel", what = "col",
whatLabels = "est")
```

1



2



```
# If R stalls, open the console. I received the instruction, 'Hit
# <Return> to see next plot:' Then it ran!
```

`semPath()` automatically produced TWO figures. Toggling between them, we see the configuration is the same, but some of the values change on the paths. In the next models we'll tighten those down.

### 12.5.1 Interpreting the Output

Criteria	Our Results	Criteria met?
Mild: factor loadings significant, strong, proper valence Yes	Help: 0.45 to 0.58; Min: 0.34 to 0.64; Pers: 0.38 to 0.69; Oth: 0.32 to 0.65	
Severe: factor loadings significant, strong, proper valence	Help: 0.41 to 0.64; Min: 0.41 to 0.47; Pers: 0.35 to 0.56; Oth: 0.39 to 0.62	

Criteria	Our Results	Criteria met?
Yes		
Non-significant chi-square	$\chi^2(328) = 327.57, p = 0.496$	Yes
$CFI \geq .95$ or $CFI \geq .90$	CFI = 1.000	Yes
$RMSEA \leq .05$ (but definitely < .10)	RMSEA = 0.000, 90%CI(0.000, 0.018)	Yes
$SRMR \leq .08$ (but definitely < .10)	SRMR = 0.032	Yes
Combination rule: $CFI \geq .95$ and $SRMR \leq .08$	CFI = 1.000 SRMR = 0.034	Yes

### 12.5.2 Partial Write-up

**Configural Model.** The configural model, which constrained only the relative configuration of variables in the model to be the same in both groups had adequate fit to the data:  $\chi^2(328) = 327.57, p = 0.496$ , CFI = 1.000, SRMR = 0.034, RMSEA = 0.000, 90%CI(0.000, 0.018).

## 12.6 Weak Invariance

Weak invariance is predicated on configural invariance and it adds cross-group equality constraints on the pattern (factor) loadings.

A priori, we know this will not (can not) be better than configural invariance. We are simply hoping that it is the same or not statistically, significantly different.

```
set.seed(240504)
weak <- lavaan:::cfa(AMS4CorrMod, data = dfAMSi, group = "Group", group.equal = "loadings")
lavaan:::summary(weak, fit.measures = TRUE, standardized = TRUE)
```

lavaan 0.6.17 ended normally after 65 iterations

Estimator	ML
Optimization method	NLMINB
Number of model parameters	132
Number of equality constraints	16

Number of observations per group:	
Mild	548
Severe	285

Model Test User Model:

Test statistic	352.993
Degrees of freedom	344

P-value (Chi-square)	0.357
Test statistic for each group:	
Mild	173.702
Severe	179.291

Model Test Baseline Model:

Test statistic	2454.883
Degrees of freedom	380
P-value	0.000

User Model versus Baseline Model:

Comparative Fit Index (CFI)	0.996
Tucker-Lewis Index (TLI)	0.995

Loglikelihood and Information Criteria:

Loglikelihood user model (H0)	-19099.545
Loglikelihood unrestricted model (H1)	-18923.048
Akaike (AIC)	38431.089
Bayesian (BIC)	38979.193
Sample-size adjusted Bayesian (SABIC)	38610.817

Root Mean Square Error of Approximation:

RMSEA	0.008
90 Percent confidence interval - lower	0.000
90 Percent confidence interval - upper	0.020
P-value H_0: RMSEA <= 0.050	1.000
P-value H_0: RMSEA >= 0.080	0.000

Standardized Root Mean Square Residual:

SRMR	0.039
------	-------

Parameter Estimates:

Standard errors	Standard
Information	Expected
Information saturated (h1) model	Structured

Group 1 [Mild]:

Latent Variables:

Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
----------	---------	---------	---------	--------	---------

Helplessness =~						
Help1		1.000			0.393	0.528
Help2 (.p2.)		1.161	0.110	10.543	0.000	0.456
Help3 (.p3.)		1.084	0.110	9.866	0.000	0.426
Help4 (.p4.)		0.982	0.103	9.554	0.000	0.385
Help5 (.p5.)		1.150	0.111	10.317	0.000	0.452
Minimization =~						
Min1		1.000			0.434	0.530
Min2 (.p7.)		0.750	0.133	5.625	0.000	0.326
Min3 (.p8.)		0.751	0.134	5.622	0.000	0.326
DenialPersonhood =~						
Per1		1.000			0.439	0.518
Per2 (.10.)		1.141	0.100	11.455	0.000	0.501
Per3 (.11.)		1.097	0.098	11.195	0.000	0.481
Per4 (.12.)		0.720	0.083	8.730	0.000	0.316
Per5 (.13.)		0.701	0.087	8.047	0.000	0.307
Otherization =~						
Oth1		1.000			0.542	0.635
Oth2 (.15.)		0.819	0.068	12.115	0.000	0.444
Oth3 (.16.)		0.816	0.068	11.999	0.000	0.443
Oth4 (.17.)		0.720	0.070	10.253	0.000	0.391
Oth5 (.18.)		0.520	0.058	8.993	0.000	0.282
Oth6 (.19.)		0.556	0.062	8.977	0.000	0.301
Oth7 (.20.)		0.518	0.062	8.301	0.000	0.281
Covariances:						
	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
Helplessness ~~						
Minimization	0.045	0.014	3.104	0.002	0.262	0.262
DenialPersonhd	0.120	0.016	7.415	0.000	0.699	0.699
Otherization	0.135	0.018	7.530	0.000	0.636	0.636
Minimization ~~						
DenialPersonhd	0.077	0.017	4.517	0.000	0.405	0.405
Otherization	0.084	0.020	4.179	0.000	0.356	0.356
DenialPersonhood ~~						
Otherization	0.167	0.021	7.987	0.000	0.703	0.703
Intercepts:						
	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
.Help1	1.978	0.032	62.311	0.000	1.978	2.662
.Help2	2.400	0.034	70.483	0.000	2.400	3.011
.Help3	2.137	0.036	59.244	0.000	2.137	2.531
.Help4	2.226	0.035	62.823	0.000	2.226	2.684
.Help5	2.020	0.036	56.683	0.000	2.020	2.421
.Min1	2.401	0.035	68.644	0.000	2.401	2.932
.Min2	2.343	0.035	67.268	0.000	2.343	2.874
.Min3	2.631	0.034	76.874	0.000	2.631	3.284
.Per1	2.086	0.036	57.629	0.000	2.086	2.462

.Per2	2.109	0.032	65.455	0.000	2.109	2.796
.Per3	2.553	0.033	77.033	0.000	2.553	3.291
.Per4	2.381	0.033	72.525	0.000	2.381	3.098
.Per5	2.148	0.035	61.873	0.000	2.148	2.643
.Oth1	2.673	0.036	73.298	0.000	2.673	3.131
.Oth2	2.628	0.035	74.445	0.000	2.628	3.180
.Oth3	2.117	0.035	61.266	0.000	2.117	2.617
.Oth4	2.407	0.038	63.475	0.000	2.407	2.712
.Oth5	2.489	0.033	75.949	0.000	2.489	3.244
.Oth6	2.620	0.034	76.426	0.000	2.620	3.265
.Oth7	2.104	0.034	61.765	0.000	2.104	2.638

Variances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
.Help1	0.398	0.028	14.249	0.000	0.398	0.721
.Help2	0.427	0.031	13.649	0.000	0.427	0.673
.Help3	0.532	0.037	14.545	0.000	0.532	0.746
.Help4	0.540	0.036	14.958	0.000	0.540	0.784
.Help5	0.492	0.035	14.104	0.000	0.492	0.707
.Min1	0.482	0.047	10.160	0.000	0.482	0.719
.Min2	0.559	0.041	13.582	0.000	0.559	0.841
.Min3	0.536	0.040	13.416	0.000	0.536	0.834
.Per1	0.525	0.036	14.683	0.000	0.525	0.732
.Per2	0.319	0.026	12.373	0.000	0.319	0.560
.Per3	0.370	0.028	13.281	0.000	0.370	0.616
.Per4	0.491	0.032	15.530	0.000	0.491	0.831
.Per5	0.566	0.036	15.707	0.000	0.566	0.857
.Oth1	0.435	0.034	12.947	0.000	0.435	0.597
.Oth2	0.486	0.034	14.444	0.000	0.486	0.711
.Oth3	0.458	0.032	14.305	0.000	0.458	0.700
.Oth4	0.635	0.041	15.312	0.000	0.635	0.806
.Oth5	0.509	0.032	15.757	0.000	0.509	0.865
.Oth6	0.553	0.035	15.710	0.000	0.553	0.859
.Oth7	0.557	0.035	15.820	0.000	0.557	0.876
Helplessness	0.154	0.024	6.399	0.000	1.000	1.000
Minimization	0.189	0.044	4.307	0.000	1.000	1.000
DenialPersonhd	0.192	0.030	6.427	0.000	1.000	1.000
Otherization	0.294	0.037	7.962	0.000	1.000	1.000

Group 2 [Severe]:

Latent Variables:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
Helplessness =~						
Help1	1.000				0.407	0.510
Help2 (.p2.)	1.161	0.110	10.543	0.000	0.473	0.570
Help3 (.p3.)	1.084	0.110	9.866	0.000	0.442	0.515

Help4	(.p4.)	0.982	0.103	9.554	0.000	0.400	0.517
Help5	(.p5.)	1.150	0.111	10.317	0.000	0.468	0.565
<b>Minimization</b> =~							
Min1		1.000				0.432	0.534
Min2	(.p7.)	0.750	0.133	5.625	0.000	0.324	0.415
Min3	(.p8.)	0.751	0.134	5.622	0.000	0.324	0.405
<b>DenialPersonhood</b> =~							
Per1		1.000				0.455	0.525
Per2	(.10.)	1.141	0.100	11.455	0.000	0.519	0.625
Per3	(.11.)	1.097	0.098	11.195	0.000	0.499	0.618
Per4	(.12.)	0.720	0.083	8.730	0.000	0.328	0.415
Per5	(.13.)	0.701	0.087	8.047	0.000	0.319	0.351
<b>Otherization</b> =~							
Oth1		1.000				0.565	0.640
Oth2	(.15.)	0.819	0.068	12.115	0.000	0.462	0.596
Oth3	(.16.)	0.816	0.068	11.999	0.000	0.461	0.557
Oth4	(.17.)	0.720	0.070	10.253	0.000	0.407	0.468
Oth5	(.18.)	0.520	0.058	8.993	0.000	0.294	0.417
Oth6	(.19.)	0.556	0.062	8.977	0.000	0.314	0.402
Oth7	(.20.)	0.518	0.062	8.301	0.000	0.292	0.352

**Covariances:**

		Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
<b>Helplessness</b> ~~							
Minimization		0.042	0.020	2.142	0.032	0.240	0.240
DenialPersonhd		0.105	0.020	5.251	0.000	0.565	0.565
Otherization		0.119	0.023	5.175	0.000	0.516	0.516
<b>Minimization</b> ~~							
DenialPersonhd		0.060	0.022	2.669	0.008	0.305	0.305
Otherization		0.130	0.029	4.450	0.000	0.531	0.531
<b>DenialPersonhood</b> ~~							
Otherization		0.120	0.025	4.801	0.000	0.466	0.466

**Intercepts:**

		Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
.Help1		2.411	0.047	50.902	0.000	2.411	3.015
.Help2		3.140	0.049	63.944	0.000	3.140	3.788
.Help3		2.733	0.051	53.787	0.000	2.733	3.186
.Help4		2.996	0.046	65.419	0.000	2.996	3.875
.Help5		2.860	0.049	58.177	0.000	2.860	3.446
.Min1		2.712	0.048	56.634	0.000	2.712	3.355
.Min2		2.677	0.046	57.887	0.000	2.677	3.429
.Min3		2.849	0.047	60.155	0.000	2.849	3.563
.Per1		2.698	0.051	52.630	0.000	2.698	3.118
.Per2		2.660	0.049	54.068	0.000	2.660	3.203
.Per3		3.133	0.048	65.507	0.000	3.133	3.880
.Per4		2.986	0.047	63.887	0.000	2.986	3.784
.Per5		2.730	0.054	50.677	0.000	2.730	3.002

.Oth1	3.025	0.052	57.831	0.000	3.025	3.426
.Oth2	3.021	0.046	65.767	0.000	3.021	3.896
.Oth3	2.667	0.049	54.354	0.000	2.667	3.220
.Oth4	2.818	0.052	54.700	0.000	2.818	3.240
.Oth5	2.867	0.042	68.742	0.000	2.867	4.072
.Oth6	2.842	0.046	61.423	0.000	2.842	3.638
.Oth7	2.481	0.049	50.360	0.000	2.481	2.983

Variances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
.Help1	0.473	0.045	10.530	0.000	0.473	0.740
.Help2	0.464	0.047	9.972	0.000	0.464	0.675
.Help3	0.541	0.052	10.477	0.000	0.541	0.735
.Help4	0.438	0.042	10.440	0.000	0.438	0.733
.Help5	0.469	0.047	10.023	0.000	0.469	0.681
.Min1	0.467	0.058	8.110	0.000	0.467	0.715
.Min2	0.505	0.050	10.114	0.000	0.505	0.828
.Min3	0.534	0.052	10.226	0.000	0.534	0.836
.Per1	0.542	0.052	10.354	0.000	0.542	0.724
.Per2	0.420	0.045	9.244	0.000	0.420	0.609
.Per3	0.403	0.043	9.337	0.000	0.403	0.618
.Per4	0.515	0.046	11.086	0.000	0.515	0.828
.Per5	0.725	0.064	11.375	0.000	0.725	0.877
.Oth1	0.460	0.048	9.515	0.000	0.460	0.591
.Oth2	0.388	0.039	10.000	0.000	0.388	0.644
.Oth3	0.473	0.046	10.386	0.000	0.473	0.690
.Oth4	0.591	0.054	10.973	0.000	0.591	0.781
.Oth5	0.409	0.037	11.208	0.000	0.409	0.826
.Oth6	0.512	0.045	11.278	0.000	0.512	0.839
.Oth7	0.606	0.053	11.460	0.000	0.606	0.876
Helplessness	0.166	0.030	5.514	0.000	1.000	1.000
Minimization	0.186	0.050	3.705	0.000	1.000	1.000
DenialPersonhd	0.207	0.037	5.613	0.000	1.000	1.000
Otherization	0.319	0.048	6.674	0.000	1.000	1.000

```
lavaan::anova(configural, weak)
```

#### Chi-Squared Difference Test

	Df	AIC	BIC	Chisq	Chisq diff	RMSEA	Df	diff	Pr(>Chisq)
configural	328	38438	39061	327.57					
weak	344	38431	38979	352.99	25.42	0.037598	16	0.06275	.
<hr/>									
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1									

Let's format these results into tables.

```
WeakFitStats <- tidySEM::table_fit(weak)
Weak_paramEsts <- tidySEM::table_results(weak, digits = 3, columns = NULL)
WeakCorrs <- tidySEM::table_cors(weak, digits = 3)
# to see each of the tables, remove the hashtag WeakFitStats
# Weak_paramEsts WeakCorrs
```

Then, export them.

```
write.csv(WeakFitStats, file = "WeakFitStats.csv")
write.csv(Weak_paramEsts, file = "Weak_paramEsts.csv")
write.csv(WeakCorrs, file = "WeakCorrs.csv")
```

### 12.6.1 Interpreting the Output

Note that although the “Std.all” values differ from each other, the “Estimates” (factor loadings) are identical across Mild and Severe groups. Each also has a “label” (e.g., .p2., .p3.) which indicates that they have been constrained to be equal. The “Std.all” differ between degree of disability severity due to the difference in standard deviations of the indicators.

Criteria	Our Results	Criteria met?
Mild: factor loadings significant, strong, proper valence	Help: 0.47 to 0.57; Min: 0.40 to 0.53; Pers: 0.38 to 0.66; Oth: 0.35 to 0.64	
Yes		
Severe: factor loadings significant, strong, proper valence	Help: 0.41 to 0.57; Min: 0.41 to 0.53; Pers: 0.35 to 0.63; Oth: 0.35 to 0.64	
Yes		
Non-significant chi-square	$\chi^2(344) = 353.00, p = 0.357$	Yes
$CFI \geq .95$ or $CFI \geq .90$	CFI = 0.996	Yes
$RMSEA \leq .05$ (but definitely < .10)	RMSEA = 0.008, 90%CI(0.000, 0.020)	Yes
$SRMR \leq .08$ (but definitely < .10)	SRMR = 0.035	Yes
Combination rule: $CFI \geq .95$ and $SRMR \leq .08$	CFI = 1.000, SRMR = 0.035	Yes

### 12.6.2 Partial Write-up

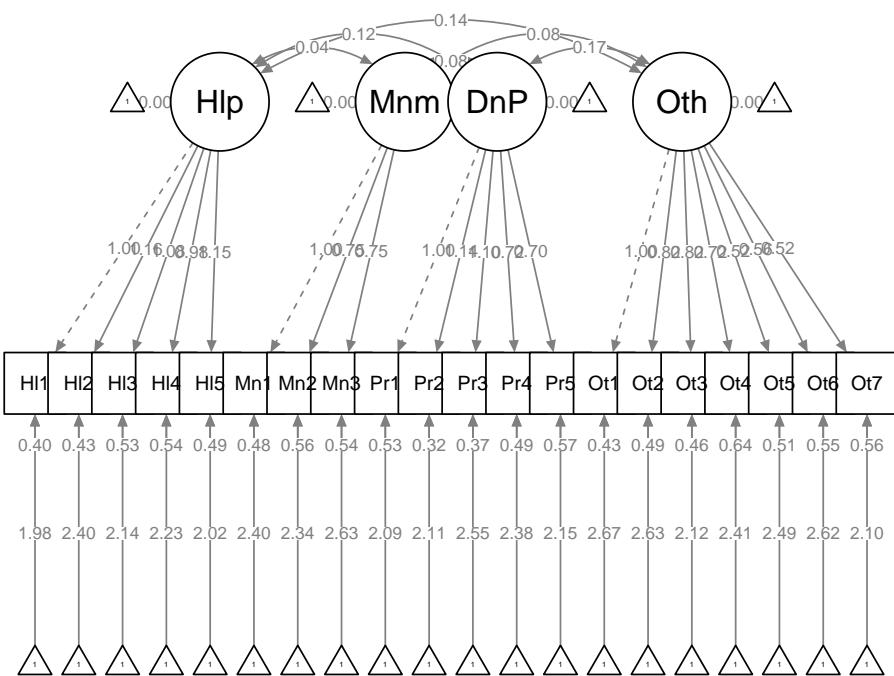
**Weak invariance model.** The weak invariance model constrained the configuration of variables and all factor loadings to be constant across groups. Fit indices were comparable to the configural model:  $\chi^2(344) = 353.00, p = 0.357$ , CFI = 0.996, SRMR = 0.039, RMSEA = 0.008, 90%CI(0.000, 0.020). Invariance of the factor loadings was supported by the non-significant difference tests that assessed model similarity:  $\chi_D^2(16) = 25.42, p = 0.063$ ;  $\Delta CFI = 0.004$

```
# The CFI difference test is calculated by simple subtraction
1 - 0.996
```

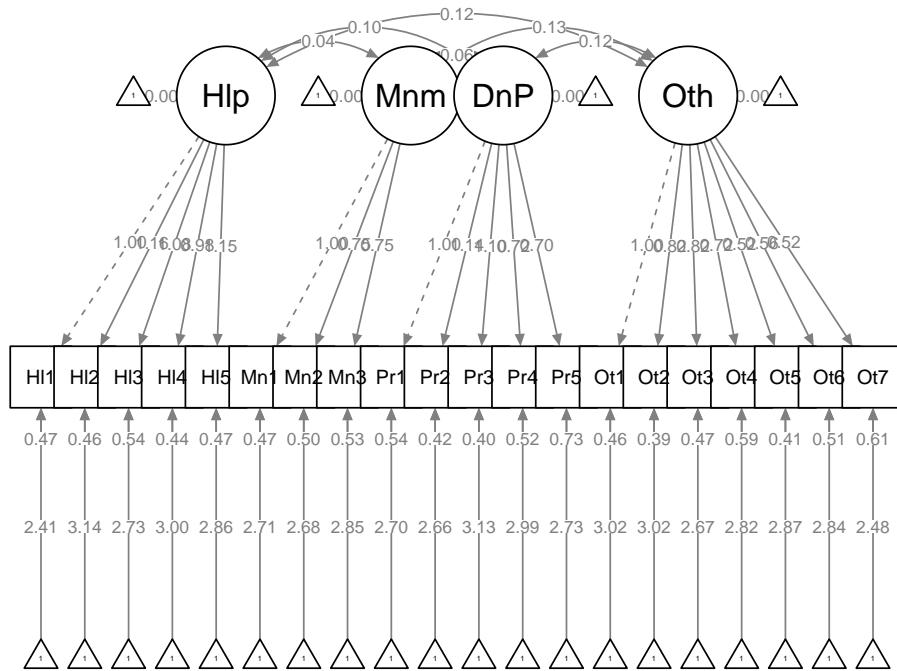
```
[1] 0.004
```

```
# semPlot::semPaths(weak, layout = 'tree', style = 'lisrel', what =
# 'col', whatLabels = 'stand')
semPlot::semPaths(weak, layout = "tree", style = "lisrel", what = "col",
  whatLabels = "est")
```

**1**



2



```
# If R stalls, open the console. I received the instruction, 'Hit
# <Return> to see next plot:' Then it ran!
```

## 12.7 Strong Invariance

Strong invariance is predicated on configural and weak invariance, but also constrains the indicator means/intercepts.

```
set.seed(240504)
strong <- lavaan::cfa(AM4CorrMod, data = dfAMSi, group = "Group", group.equal = c("loadings",
  "intercepts"))
lavaan::summary(strong, fit.measures = TRUE, standardized = TRUE)
```

```
lavaan 0.6.17 ended normally after 87 iterations
```

Estimator	ML
Optimization method	NLMINB
Number of model parameters	136
Number of equality constraints	36
 Number of observations per group:	
Mild	548
Severe	285
 Model Test User Model:	
Test statistic	412.746
Degrees of freedom	360
P-value (Chi-square)	0.029
Test statistic for each group:	
Mild	210.341
Severe	202.405
 Model Test Baseline Model:	
Test statistic	2454.883
Degrees of freedom	380
P-value	0.000
 User Model versus Baseline Model:	
Comparative Fit Index (CFI)	0.975
Tucker-Lewis Index (TLI)	0.973
 Loglikelihood and Information Criteria:	
Loglikelihood user model (H0)	-19129.421
Loglikelihood unrestricted model (H1)	-18923.048
Akaike (AIC)	38458.843
Bayesian (BIC)	38931.346
Sample-size adjusted Bayesian (SABIC)	38613.781
 Root Mean Square Error of Approximation:	
RMSEA	0.019
90 Percent confidence interval - lower	0.007
90 Percent confidence interval - upper	0.027
P-value H_0: RMSEA <= 0.050	1.000
P-value H_0: RMSEA >= 0.080	0.000
 Standardized Root Mean Square Residual:	

SRMR	0.045
------	-------

Parameter Estimates:

Standard errors Information Information saturated (h1) model	Standard Expected Structured
--	------------------------------------

Group 1 [Mild]:

Latent Variables:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
Helplessness =~						
Help1	1.000				0.329	0.452
Help2 (.p2.)	1.368	0.106	12.897	0.000	0.451	0.566
Help3 (.p3.)	1.198	0.101	11.831	0.000	0.395	0.473
Help4 (.p4.)	1.291	0.103	12.520	0.000	0.425	0.504
Help5 (.p5.)	1.464	0.112	13.092	0.000	0.482	0.571
Minimization =~						
Min1	1.000				0.416	0.509
Min2 (.p7.)	0.851	0.120	7.087	0.000	0.354	0.432
Min3 (.p8.)	0.748	0.111	6.728	0.000	0.311	0.389
DenialPersonhood =~						
Per1	1.000				0.444	0.524
Per2 (.10.)	1.044	0.074	14.168	0.000	0.464	0.625
Per3 (.11.)	1.036	0.074	14.036	0.000	0.460	0.600
Per4 (.12.)	0.817	0.068	11.957	0.000	0.363	0.463
Per5 (.13.)	0.786	0.072	10.914	0.000	0.349	0.423
Otherization =~						
Oth1	1.000				0.496	0.591
Oth2 (.15.)	0.873	0.066	13.176	0.000	0.433	0.526
Oth3 (.16.)	0.951	0.070	13.628	0.000	0.472	0.576
Oth4 (.17.)	0.803	0.070	11.516	0.000	0.398	0.447
Oth5 (.18.)	0.620	0.058	10.693	0.000	0.308	0.397
Oth6 (.19.)	0.571	0.060	9.517	0.000	0.283	0.355
Oth7 (.20.)	0.616	0.063	9.838	0.000	0.305	0.379

Covariances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
Helplessness ~~						
Minimization	0.036	0.012	3.106	0.002	0.263	0.263
DenialPersonhd	0.103	0.013	7.913	0.000	0.706	0.706
Otherization	0.103	0.014	7.588	0.000	0.632	0.632
Minimization ~~						
DenialPersonhd	0.075	0.016	4.612	0.000	0.404	0.404
Otherization	0.074	0.018	4.232	0.000	0.360	0.360
DenialPersonhood ~~						

Otherization	0.156	0.018	8.416	0.000	0.706	0.706	
<b>Intercepts:</b>							
.Help1	(.51.)	1.944	0.029	66.425	0.000	1.944	2.667
.Help2	(.52.)	2.400	0.033	73.804	0.000	2.400	3.013
.Help3	(.53.)	2.119	0.034	63.226	0.000	2.119	2.537
.Help4	(.54.)	2.254	0.034	66.598	0.000	2.254	2.673
.Help5	(.55.)	2.037	0.034	59.104	0.000	2.037	2.410
.Min1	(.56.)	2.394	0.033	71.944	0.000	2.394	2.930
.Min2	(.57.)	2.362	0.033	72.669	0.000	2.362	2.882
.Min3	(.58.)	2.621	0.031	83.690	0.000	2.621	3.279
.Per1	(.59.)	2.089	0.034	61.496	0.000	2.089	2.462
.Per2	(.60.)	2.087	0.030	68.542	0.000	2.087	2.812
.Per3	(.61.)	2.538	0.031	81.214	0.000	2.538	3.307
.Per4	(.62.)	2.418	0.031	77.819	0.000	2.418	3.081
.Per5	(.63.)	2.179	0.033	66.537	0.000	2.179	2.638
.Oth1	(.64.)	2.630	0.034	78.432	0.000	2.630	3.135
.Oth2	(.65.)	2.616	0.032	81.837	0.000	2.616	3.182
.Oth3	(.66.)	2.146	0.033	65.814	0.000	2.146	2.618
.Oth4	(.67.)	2.414	0.034	70.725	0.000	2.414	2.711
.Oth5	(.68.)	2.519	0.029	86.645	0.000	2.519	3.252
.Oth6	(.69.)	2.601	0.030	86.656	0.000	2.601	3.258
.Oth7	(.70.)	2.129	0.031	69.263	0.000	2.129	2.646
<b>Variances:</b>							
.Help1		0.423	0.028	15.147	0.000	0.423	0.796
.Help2		0.431	0.031	13.925	0.000	0.431	0.680
.Help3		0.542	0.036	14.982	0.000	0.542	0.777
.Help4		0.530	0.036	14.696	0.000	0.530	0.746
.Help5		0.482	0.035	13.866	0.000	0.482	0.674
.Min1		0.495	0.044	11.338	0.000	0.495	0.741
.Min2		0.546	0.041	13.253	0.000	0.546	0.813
.Min3		0.542	0.039	14.020	0.000	0.542	0.848
.Per1		0.522	0.035	14.706	0.000	0.522	0.726
.Per2		0.335	0.025	13.319	0.000	0.335	0.609
.Per3		0.377	0.027	13.755	0.000	0.377	0.640
.Per4		0.484	0.032	15.237	0.000	0.484	0.786
.Per5		0.560	0.036	15.494	0.000	0.560	0.821
.Oth1		0.457	0.033	13.768	0.000	0.457	0.650
.Oth2		0.489	0.033	14.609	0.000	0.489	0.723
.Oth3		0.449	0.032	13.994	0.000	0.449	0.669
.Oth4		0.634	0.041	15.289	0.000	0.634	0.800
.Oth5		0.505	0.032	15.617	0.000	0.505	0.842
.Oth6		0.557	0.035	15.831	0.000	0.557	0.874
.Oth7		0.554	0.035	15.701	0.000	0.554	0.856
Helplessness		0.109	0.016	6.725	0.000	1.000	1.000

Minimization	0.173	0.037	4.734	0.000	1.000	1.000
DenialPersonhd	0.197	0.026	7.454	0.000	1.000	1.000
Otherization	0.246	0.031	7.963	0.000	1.000	1.000

Group 2 [Severe]:

Latent Variables:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
Helplessness =~						
Help1	1.000				0.345	0.442
Help2 (.p2.)	1.368	0.106	12.897	0.000	0.473	0.569
Help3 (.p3.)	1.198	0.101	11.831	0.000	0.414	0.487
Help4 (.p4.)	1.291	0.103	12.520	0.000	0.446	0.564
Help5 (.p5.)	1.464	0.112	13.092	0.000	0.506	0.602
Minimization =~						
Min1	1.000				0.417	0.517
Min2 (.p7.)	0.851	0.120	7.087	0.000	0.355	0.451
Min3 (.p8.)	0.748	0.111	6.728	0.000	0.312	0.391
DenialPersonhood =~						
Per1	1.000				0.461	0.531
Per2 (.10.)	1.044	0.074	14.168	0.000	0.481	0.588
Per3 (.11.)	1.036	0.074	14.036	0.000	0.477	0.595
Per4 (.12.)	0.817	0.068	11.957	0.000	0.376	0.465
Per5 (.13.)	0.786	0.072	10.914	0.000	0.362	0.392
Otherization =~						
Oth1	1.000				0.517	0.594
Oth2 (.15.)	0.873	0.066	13.176	0.000	0.451	0.584
Oth3 (.16.)	0.951	0.070	13.628	0.000	0.492	0.584
Oth4 (.17.)	0.803	0.070	11.516	0.000	0.415	0.477
Oth5 (.18.)	0.620	0.058	10.693	0.000	0.321	0.449
Oth6 (.19.)	0.571	0.060	9.517	0.000	0.295	0.379
Oth7 (.20.)	0.616	0.063	9.838	0.000	0.318	0.380

Covariances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
Helplessness ~~						
Minimization	0.034	0.016	2.153	0.031	0.239	0.239
DenialPersonhd	0.089	0.017	5.379	0.000	0.560	0.560
Otherization	0.094	0.018	5.303	0.000	0.525	0.525
Minimization ~~						
DenialPersonhd	0.060	0.022	2.737	0.006	0.312	0.312
Otherization	0.115	0.025	4.506	0.000	0.532	0.532
DenialPersonhood ~~						
Otherization	0.113	0.023	4.926	0.000	0.474	0.474

Intercepts:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
--	----------	---------	---------	---------	--------	---------

.Help1	(.51.)	1.944	0.029	66.425	0.000	1.944	2.486
.Help2	(.52.)	2.400	0.033	73.804	0.000	2.400	2.890
.Help3	(.53.)	2.119	0.034	63.226	0.000	2.119	2.492
.Help4	(.54.)	2.254	0.034	66.598	0.000	2.254	2.850
.Help5	(.55.)	2.037	0.034	59.104	0.000	2.037	2.425
.Min1	(.56.)	2.394	0.033	71.944	0.000	2.394	2.970
.Min2	(.57.)	2.362	0.033	72.669	0.000	2.362	3.004
.Min3	(.58.)	2.621	0.031	83.690	0.000	2.621	3.285
.Per1	(.59.)	2.089	0.034	61.496	0.000	2.089	2.407
.Per2	(.60.)	2.087	0.030	68.542	0.000	2.087	2.553
.Per3	(.61.)	2.538	0.031	81.214	0.000	2.538	3.168
.Per4	(.62.)	2.418	0.031	77.819	0.000	2.418	2.991
.Per5	(.63.)	2.179	0.033	66.537	0.000	2.179	2.358
.Oth1	(.64.)	2.630	0.034	78.432	0.000	2.630	3.022
.Oth2	(.65.)	2.616	0.032	81.837	0.000	2.616	3.386
.Oth3	(.66.)	2.146	0.033	65.814	0.000	2.146	2.546
.Oth4	(.67.)	2.414	0.034	70.725	0.000	2.414	2.773
.Oth5	(.68.)	2.519	0.029	86.645	0.000	2.519	3.529
.Oth6	(.69.)	2.601	0.030	86.656	0.000	2.601	3.343
.Oth7	(.70.)	2.129	0.031	69.263	0.000	2.129	2.540
Hlplssn		0.541	0.044	12.314	0.000	1.567	1.567
Minmztn		0.332	0.051	6.518	0.000	0.797	0.797
DnlPrsn		0.604	0.050	12.008	0.000	1.311	1.311
Othrztn		0.485	0.049	9.804	0.000	0.938	0.938

## Variances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
.Help1	0.492	0.045	11.034	0.000	0.492	0.805
.Help2	0.466	0.046	10.115	0.000	0.466	0.676
.Help3	0.552	0.051	10.771	0.000	0.552	0.763
.Help4	0.427	0.042	10.155	0.000	0.427	0.682
.Help5	0.450	0.046	9.753	0.000	0.450	0.637
.Min1	0.476	0.055	8.720	0.000	0.476	0.733
.Min2	0.493	0.050	9.798	0.000	0.493	0.797
.Min3	0.539	0.051	10.493	0.000	0.539	0.847
.Per1	0.541	0.052	10.349	0.000	0.541	0.718
.Per2	0.437	0.045	9.782	0.000	0.437	0.654
.Per3	0.414	0.043	9.688	0.000	0.414	0.645
.Per4	0.512	0.047	10.829	0.000	0.512	0.784
.Per5	0.722	0.064	11.220	0.000	0.722	0.846
.Oth1	0.490	0.049	10.072	0.000	0.490	0.647
.Oth2	0.393	0.039	10.155	0.000	0.393	0.659
.Oth3	0.468	0.046	10.173	0.000	0.468	0.659
.Oth4	0.585	0.054	10.938	0.000	0.585	0.772
.Oth5	0.407	0.037	11.076	0.000	0.407	0.798
.Oth6	0.518	0.046	11.371	0.000	0.518	0.856
.Oth7	0.601	0.053	11.374	0.000	0.601	0.856
Helplessness	0.119	0.021	5.803	0.000	1.000	1.000

Minimization	0.174	0.044	3.983	0.000	1.000	1.000
DenialPersonhd	0.212	0.034	6.202	0.000	1.000	1.000
Otherization	0.267	0.040	6.679	0.000	1.000	1.000

```
lavaan::anova(configural, weak, strong)
```

### Chi-Squared Difference Test

	Df	AIC	BIC	Chisq	Chisq diff	RMSEA	Df diff	Pr(>Chisq)			
configural	328	38438	39061	327.57							
weak	344	38431	38979	352.99	25.420	0.037598	16	0.06275 .			
strong	360	38459	38931	412.75	59.754	0.081029	16	0.0000005758 ***			
---											
Signif. codes:	0	'***'	0.001	'**'	0.01	'*'	0.05	'. '	0.1	' '	1

Let's format these results into tables.

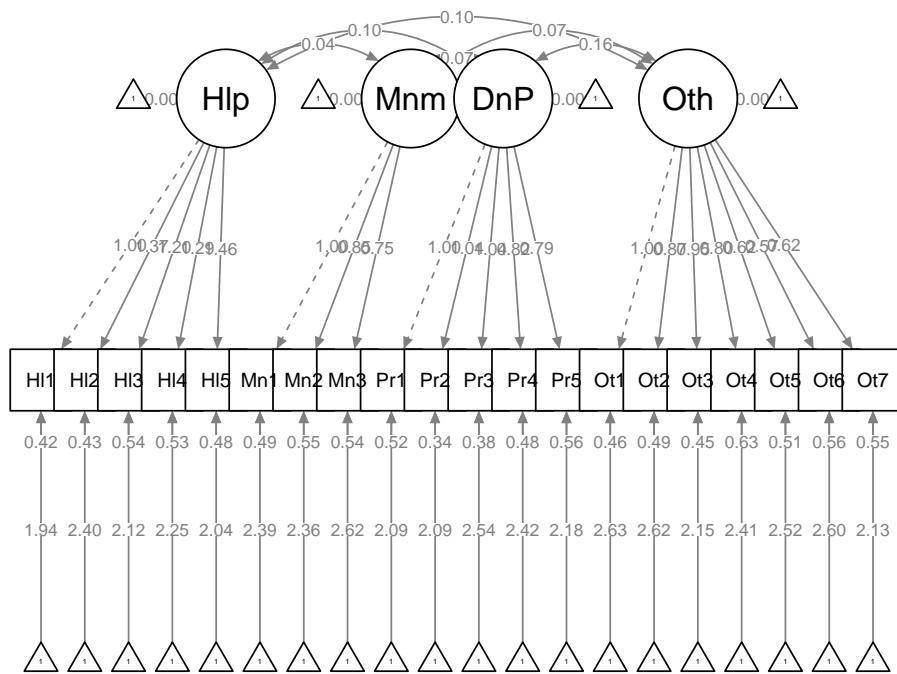
```
strongFitStats <- tidySEM::table_fit(strong)
strong_paramEsts <- tidySEM::table_results(strong, digits = 3, columns = NULL)
strongCorrs <- tidySEM::table_corrs(strong, digits = 3)
# to see each of the tables, remove the hashtag
# strongFitStats
# strong_paramEsts
# strongCorrs
```

Then, export them.

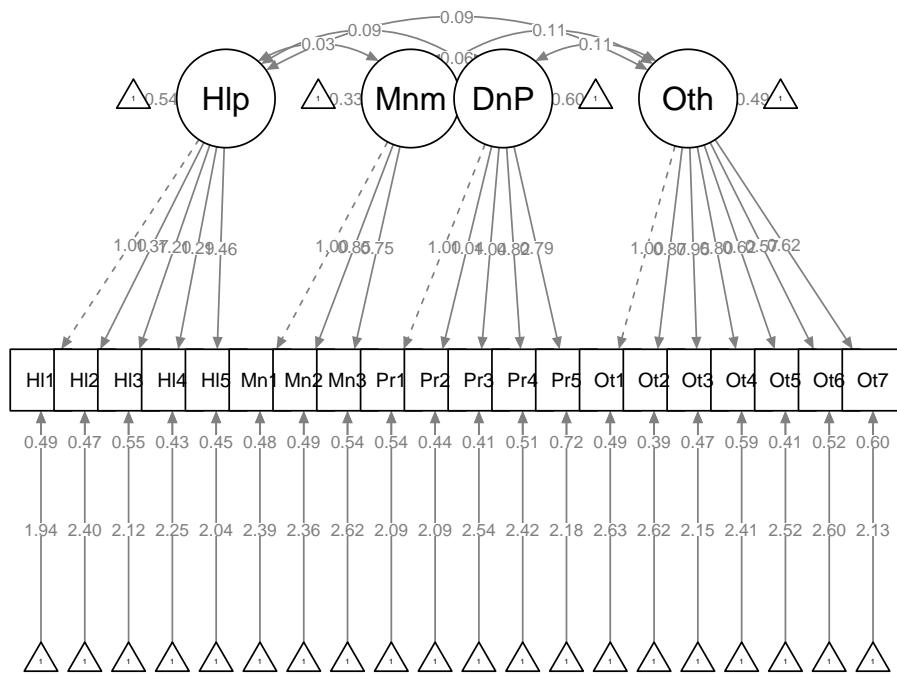
```
write.csv(strongFitStats, file = "strongFitStats.csv")
write.csv(strong_paramEsts, file = "strong_paramEsts.csv")
write.csv(strongCorrs, file = "strongCorrs.csv")
```

```
# semPlot::semPaths(strong, layout = 'tree', style = 'lisrel', what =
# 'col', whatLabels = 'stand')
semPlot::semPaths(strong, layout = "tree", style = "lisrel", what = "col",
whatLabels = "est")
```

1



2



```
# If R stalls, open the console. I received the instruction, 'Hit
# <Return> to see next plot:' Then it ran!
```

### 12.7.1 Interpreting the Output

Note that although the “Std.all” values differ from each other, the “Estimates” (factor loadings) are identical across Mild and Severe groups. Each also has a “label” (e.g., .p2., .p3.) which indicates that they have been constrained to be equal. The “Std.all” differ between degree of disability severity due to the difference in standard deviations of the indicators.

Criteria	Our Results	Criteria met?
Mild: factor loadings significant, strong, proper valence Yes	Help: 0.45 to 0.57; Min: 0.39 to 0.51; Pers: 0.42 to 0.63; Oth: 0.36 to 0.59	

Criteria	Our Results	Criteria met?
Severe: factor loadings significant, strong, proper valence Yes	Help: 0.44 to 0.60; Min: 0.39 to 0.52; Pers: 0.39 to 0.60; Oth: 0.38 to 0.59	
Non-significant chi-square	$\chi^2(360) = 412.75, p = 0.029$	No
$CFI \geq .95$ or $CFI \geq .90$	CFI = 0.975	Yes
$RMSEA \leq .05$ (but definitely < .10)	RMSEA = 0.019, CI90%(0.007 to 0.027)	Yes
$SRMR \leq .08$ (but definitely < .10)	SRMR = 0.045	Yes
Combination rule: $CFI \geq .95$ and $SRMR \leq .08$	CFI = .805, SRMR = 0.036	Yes

### 12.7.2 Partial Write-up

**Strong invariance model.** In the strong invariance model, configuration, factor loadings, and indicator means/intercepts were constrained to be the same for each group. Fit indices were less than ideal:  $\chi^2(360) = 412.75, p = 0.029$ , CFI = 0.975, SRMR = 0.045, RMSEA = 0.019, CI90%(0.007 to 0.027). The difference tests that evaluated model similarity suggested there was factorial noninvariance:  $(\chi^2_D(16) = 59.754, p = 0.01;$   $\Delta CFI = 0.021$ . Given that the  $\chi^2_D$  test is statistically significant and the  $\Delta CFI > 0.01$  we cannot claim strong invariance and we therefore do not test stricter models.

0.996 – 0.975

[1] 0.021

#### Should I be worried if measurement invariance stops here?

Byrne [2016b] wrote, “Historically, the Joreskog tradition of invariance testing held that the equality of error variances and their covariances should also be tested. However, it is now widely accepted that to do so represents an overly restrictive test of the data” (p. 230).

Further, in an awesome article examining the factorial invariance of the Calling & Vocation Questionnaire [Autin et al., 2017] in a binational sample, strict invariance (the next level of restraint) was not even mentioned. Further, after strong invariance was not achieved the authors wrote, “Therefore, poor fit in only this model does not necessarily indicate the factor structure operates differently for different groups” (p. 695).

SO...as a researcher, I would be happy if I had configural (just the shape) and weak (factor loadings) invariance.

Plus..a little later in the lecture we head into *partial measurement invariance*.

If we fail at this stage (or at any earlier stage), we would normally not continue. Because this lesson is for training, we will continue onto the last model.

## 12.8 Strict Invariance

Strict invariance is predicated on configural, weak, and strong invariance. To that, it adds cross-group equality constraints on the residuals.

```
set.seed(240504)
strict <- lavaan::cfa(AMSA4CorrMod, data = dfAMSA, group = "Group", group.equal = c("loadings",
  "intercepts", "residuals"))
lavaan::summary(strict, fit.measures = TRUE, standardized = TRUE, )
```

lavaan 0.6.17 ended normally after 81 iterations

Estimator	ML
Optimization method	NLMINB
Number of model parameters	136
Number of equality constraints	56

Number of observations per group:	
Mild	548
Severe	285

Model Test User Model:

Test statistic	439.692
Degrees of freedom	380
P-value (Chi-square)	0.019
Test statistic for each group:	
Mild	221.270
Severe	218.422

Model Test Baseline Model:

Test statistic	2454.883
Degrees of freedom	380
P-value	0.000

User Model versus Baseline Model:

Comparative Fit Index (CFI)	0.971
Tucker-Lewis Index (TLI)	0.971

Loglikelihood and Information Criteria:

Loglikelihood user model (H0)	-19142.894
Loglikelihood unrestricted model (H1)	-18923.048
Akaike (AIC)	38445.789

Bayesian (BIC)	38823.792
Sample-size adjusted Bayesian (SABIC)	38569.739

Root Mean Square Error of Approximation:

RMSEA	0.019
90 Percent confidence interval - lower	0.009
90 Percent confidence interval - upper	0.027
P-value H_0: RMSEA <= 0.050	1.000
P-value H_0: RMSEA >= 0.080	0.000

Standardized Root Mean Square Residual:

SRMR	0.046
------	-------

Parameter Estimates:

Standard errors	Standard
Information	Expected
Information saturated (h1) model	Structured

Group 1 [Mild]:

Latent Variables:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
Helplessness =~						
Help1	1.000				0.326	0.438
Help2 (.p2.)	1.383	0.107	12.902	0.000	0.450	0.560
Help3 (.p3.)	1.212	0.102	11.826	0.000	0.395	0.472
Help4 (.p4.)	1.299	0.105	12.389	0.000	0.423	0.515
Help5 (.p5.)	1.477	0.113	13.041	0.000	0.481	0.574
Minimization =~						
Min1	1.000				0.424	0.520
Min2 (.p7.)	0.841	0.119	7.044	0.000	0.357	0.440
Min3 (.p8.)	0.738	0.110	6.711	0.000	0.313	0.391
DenialPersonhood =~						
Per1	1.000				0.439	0.518
Per2 (.10.)	1.030	0.072	14.230	0.000	0.453	0.596
Per3 (.11.)	1.033	0.073	14.122	0.000	0.454	0.588
Per4 (.12.)	0.819	0.068	12.052	0.000	0.360	0.456
Per5 (.13.)	0.790	0.071	11.061	0.000	0.347	0.405
Otherization =~						
Oth1	1.000				0.497	0.587
Oth2 (.15.)	0.876	0.067	13.107	0.000	0.435	0.542
Oth3 (.16.)	0.953	0.070	13.637	0.000	0.474	0.574
Oth4 (.17.)	0.801	0.070	11.449	0.000	0.398	0.452
Oth5 (.18.)	0.620	0.059	10.577	0.000	0.308	0.409

Oth6	(.19.)	0.573	0.060	9.500	0.000	0.284	0.360
Oth7	(.20.)	0.619	0.063	9.887	0.000	0.308	0.377

## Covariances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
<b>Helplessness ~~</b>						
Minimization	0.036	0.012	3.096	0.002	0.259	0.259
DenialPersonhd	0.102	0.013	7.884	0.000	0.712	0.712
Otherization	0.102	0.013	7.560	0.000	0.631	0.631
<b>Minimization ~~</b>						
DenialPersonhd	0.075	0.016	4.614	0.000	0.405	0.405
Otherization	0.075	0.018	4.238	0.000	0.355	0.355
<b>DenialPersonhood ~~</b>						
Otherization	0.155	0.018	8.404	0.000	0.711	0.711

## Intercepts:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all	
.Help1	(.51.)	1.943	0.030	65.457	0.000	1.943	2.612
.Help2	(.52.)	2.400	0.033	73.272	0.000	2.400	2.986
.Help3	(.53.)	2.119	0.034	63.095	0.000	2.119	2.531
.Help4	(.54.)	2.252	0.033	67.814	0.000	2.252	2.741
.Help5	(.55.)	2.037	0.034	59.474	0.000	2.037	2.430
.Min1	(.56.)	2.394	0.033	71.865	0.000	2.394	2.934
.Min2	(.57.)	2.361	0.032	73.225	0.000	2.361	2.916
.Min3	(.58.)	2.622	0.031	83.617	0.000	2.622	3.277
.Per1	(.59.)	2.088	0.034	61.558	0.000	2.088	2.460
.Per2	(.60.)	2.084	0.031	67.533	0.000	2.084	2.744
.Per3	(.61.)	2.537	0.031	80.958	0.000	2.537	3.284
.Per4	(.62.)	2.419	0.031	77.688	0.000	2.419	3.067
.Per5	(.63.)	2.183	0.034	65.090	0.000	2.183	2.546
.Oth1	(.64.)	2.628	0.034	77.964	0.000	2.628	3.108
.Oth2	(.65.)	2.617	0.032	82.717	0.000	2.617	3.258
.Oth3	(.66.)	2.147	0.033	65.562	0.000	2.147	2.604
.Oth4	(.67.)	2.415	0.034	71.063	0.000	2.415	2.740
.Oth5	(.68.)	2.516	0.029	87.481	0.000	2.516	3.342
.Oth6	(.69.)	2.601	0.030	86.991	0.000	2.601	3.292
.Oth7	(.70.)	2.130	0.031	68.797	0.000	2.130	2.612

## Variances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all	
.Help1	(.21.)	0.447	0.024	18.643	0.000	0.447	0.808
.Help2	(.22.)	0.443	0.026	16.948	0.000	0.443	0.686
.Help3	(.23.)	0.545	0.030	18.272	0.000	0.545	0.778
.Help4	(.24.)	0.496	0.028	17.699	0.000	0.496	0.735
.Help5	(.25.)	0.471	0.028	16.684	0.000	0.471	0.670
.Min1	(.26.)	0.486	0.037	13.237	0.000	0.486	0.730
.Min2	(.27.)	0.529	0.033	15.800	0.000	0.529	0.806
.Min3	(.28.)	0.542	0.032	16.988	0.000	0.542	0.847

.Per1	(.29.)	0.528	0.030	17.766	0.000	0.528	0.732
.Per2	(.30.)	0.372	0.023	16.379	0.000	0.372	0.645
.Per3	(.31.)	0.391	0.024	16.557	0.000	0.391	0.655
.Per4	(.32.)	0.493	0.027	18.526	0.000	0.493	0.792
.Per5	(.33.)	0.615	0.032	19.010	0.000	0.615	0.836
.Oth1	(.34.)	0.468	0.028	16.733	0.000	0.468	0.655
.Oth2	(.35.)	0.456	0.026	17.522	0.000	0.456	0.706
.Oth3	(.36.)	0.455	0.027	16.979	0.000	0.455	0.670
.Oth4	(.37.)	0.618	0.033	18.638	0.000	0.618	0.796
.Oth5	(.38.)	0.472	0.025	19.023	0.000	0.472	0.833
.Oth6	(.39.)	0.544	0.028	19.385	0.000	0.544	0.870
.Oth7	(.40.)	0.570	0.030	19.266	0.000	0.570	0.858
Hplssn		0.106	0.016	6.684	0.000	1.000	1.000
Minmztn		0.180	0.037	4.881	0.000	1.000	1.000
DnlPrsn		0.193	0.026	7.442	0.000	1.000	1.000
Othrztn		0.247	0.031	7.955	0.000	1.000	1.000

Group 2 [Severe]:

Latent Variables:

		Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
Helplessness =~							
Help1		1.000				0.342	0.455
Help2	(.p2.)	1.383	0.107	12.902	0.000	0.473	0.579
Help3	(.p3.)	1.212	0.102	11.826	0.000	0.414	0.489
Help4	(.p4.)	1.299	0.105	12.389	0.000	0.444	0.533
Help5	(.p5.)	1.477	0.113	13.041	0.000	0.505	0.593
Minimization =~							
Min1		1.000				0.411	0.508
Min2	(.p7.)	0.841	0.119	7.044	0.000	0.345	0.429
Min3	(.p8.)	0.738	0.110	6.711	0.000	0.303	0.381
DenialPersonhood =~							
Per1		1.000				0.471	0.544
Per2	(.10.)	1.030	0.072	14.230	0.000	0.485	0.623
Per3	(.11.)	1.033	0.073	14.122	0.000	0.487	0.614
Per4	(.12.)	0.819	0.068	12.052	0.000	0.386	0.481
Per5	(.13.)	0.790	0.071	11.061	0.000	0.372	0.429
Otherization =~							
Oth1		1.000				0.514	0.600
Oth2	(.15.)	0.876	0.067	13.107	0.000	0.450	0.555
Oth3	(.16.)	0.953	0.070	13.637	0.000	0.490	0.587
Oth4	(.17.)	0.801	0.070	11.449	0.000	0.412	0.464
Oth5	(.18.)	0.620	0.059	10.577	0.000	0.319	0.421
Oth6	(.19.)	0.573	0.060	9.500	0.000	0.294	0.371
Oth7	(.20.)	0.619	0.063	9.887	0.000	0.318	0.388

Covariances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all	
<b>Helplessness ~~</b>							
Minimization	0.036	0.016	2.234	0.026	0.255	0.255	
DenialPersonhd	0.090	0.017	5.439	0.000	0.558	0.558	
Otherization	0.093	0.018	5.289	0.000	0.527	0.527	
<b>Minimization ~~</b>							
DenialPersonhd	0.060	0.022	2.704	0.007	0.308	0.308	
Otherization	0.115	0.026	4.494	0.000	0.543	0.543	
<b>DenialPersonhood ~~</b>							
Otherization	0.113	0.023	4.914	0.000	0.466	0.466	
<b>Intercepts:</b>							
	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all	
.Help1	(.51.)	1.943	0.030	65.457	0.000	1.943	2.587
.Help2	(.52.)	2.400	0.033	73.272	0.000	2.400	2.940
.Help3	(.53.)	2.119	0.034	63.095	0.000	2.119	2.503
.Help4	(.54.)	2.252	0.033	67.814	0.000	2.252	2.705
.Help5	(.55.)	2.037	0.034	59.474	0.000	2.037	2.390
.Min1	(.56.)	2.394	0.033	71.865	0.000	2.394	2.959
.Min2	(.57.)	2.361	0.032	73.225	0.000	2.361	2.934
.Min3	(.58.)	2.622	0.031	83.617	0.000	2.622	3.292
.Per1	(.59.)	2.088	0.034	61.558	0.000	2.088	2.412
.Per2	(.60.)	2.084	0.031	67.533	0.000	2.084	2.674
.Per3	(.61.)	2.537	0.031	80.958	0.000	2.537	3.202
.Per4	(.62.)	2.419	0.031	77.688	0.000	2.419	3.020
.Per5	(.63.)	2.183	0.034	65.090	0.000	2.183	2.515
.Oth1	(.64.)	2.628	0.034	77.964	0.000	2.628	3.071
.Oth2	(.65.)	2.617	0.032	82.717	0.000	2.617	3.225
.Oth3	(.66.)	2.147	0.033	65.562	0.000	2.147	2.575
.Oth4	(.67.)	2.415	0.034	71.063	0.000	2.415	2.721
.Oth5	(.68.)	2.516	0.029	87.481	0.000	2.516	3.323
.Oth6	(.69.)	2.601	0.030	86.991	0.000	2.601	3.277
.Oth7	(.70.)	2.130	0.031	68.797	0.000	2.130	2.599
Hplssn		0.536	0.043	12.323	0.000	1.567	1.567
Minmztn		0.333	0.051	6.525	0.000	0.812	0.812
DnlPrsn		0.606	0.050	12.024	0.000	1.286	1.286
Othrztn		0.484	0.049	9.810	0.000	0.942	0.942
<b>Variances:</b>							
	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all	
.Help1	(.21.)	0.447	0.024	18.643	0.000	0.447	0.793
.Help2	(.22.)	0.443	0.026	16.948	0.000	0.443	0.665
.Help3	(.23.)	0.545	0.030	18.272	0.000	0.545	0.761
.Help4	(.24.)	0.496	0.028	17.699	0.000	0.496	0.716
.Help5	(.25.)	0.471	0.028	16.684	0.000	0.471	0.649
.Min1	(.26.)	0.486	0.037	13.237	0.000	0.486	0.742
.Min2	(.27.)	0.529	0.033	15.800	0.000	0.529	0.816
.Min3	(.28.)	0.542	0.032	16.988	0.000	0.542	0.855

.Per1	(.29.)	0.528	0.030	17.766	0.000	0.528	0.704
.Per2	(.30.)	0.372	0.023	16.379	0.000	0.372	0.612
.Per3	(.31.)	0.391	0.024	16.557	0.000	0.391	0.623
.Per4	(.32.)	0.493	0.027	18.526	0.000	0.493	0.768
.Per5	(.33.)	0.615	0.032	19.010	0.000	0.615	0.816
.Oth1	(.34.)	0.468	0.028	16.733	0.000	0.468	0.639
.Oth2	(.35.)	0.456	0.026	17.522	0.000	0.456	0.692
.Oth3	(.36.)	0.455	0.027	16.979	0.000	0.455	0.655
.Oth4	(.37.)	0.618	0.033	18.638	0.000	0.618	0.785
.Oth5	(.38.)	0.472	0.025	19.023	0.000	0.472	0.823
.Oth6	(.39.)	0.544	0.028	19.385	0.000	0.544	0.863
.Oth7	(.40.)	0.570	0.030	19.266	0.000	0.570	0.849
Hlplssn		0.117	0.020	5.838	0.000	1.000	1.000
Minmztn		0.169	0.042	4.001	0.000	1.000	1.000
DnlPrsn		0.222	0.035	6.398	0.000	1.000	1.000
Othrztn		0.264	0.039	6.697	0.000	1.000	1.000

```
lavaan:::anova(configural, weak, strong, strict)
```

### Chi-Squared Difference Test

	Df	AIC	BIC	Chisq	Chisq diff	RMSEA	Df diff	Pr(>Chisq)
configural	328	38438	39061	327.57				
weak	344	38431	38979	352.99	25.420	0.037598	16	0.06275 .
strong	360	38459	38931	412.75	59.754	0.081029	16	0.0000005758 ***
strict	380	38446	38824	439.69	26.946	0.028876	20	0.13679
<hr/>								
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1								

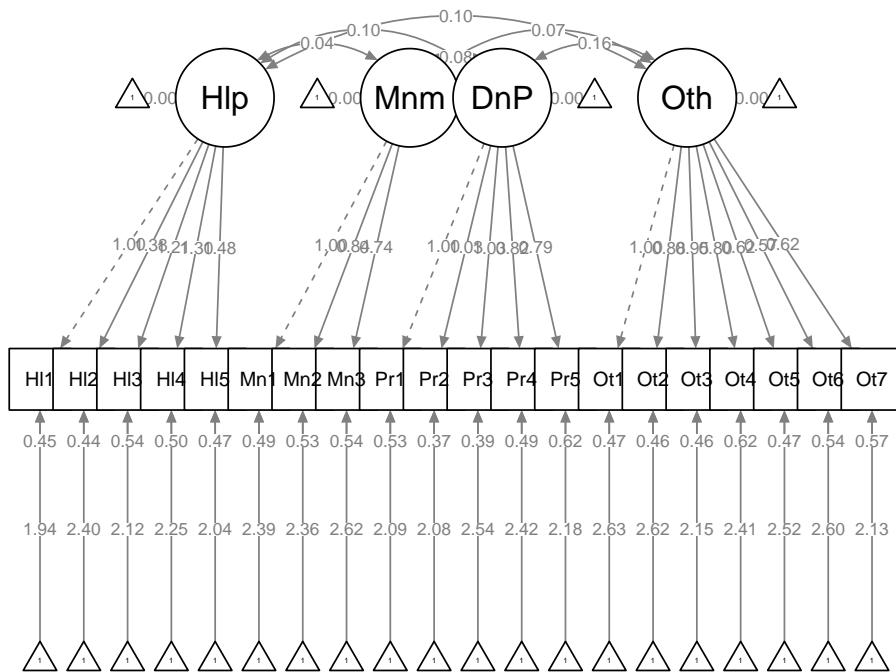
Let's format these results into tables.

```
strictFitStats <- tidySEM:::table_fit(strict)
strict_paramEsts <- tidySEM:::table_results(strict, digits = 3, columns = NULL)
strictCorrs <- tidySEM:::table_corrs(strict, digits = 3)
# to see each of the tables, remove the hashtag strictFitStats
# strict_paramEsts strictCorrs
```

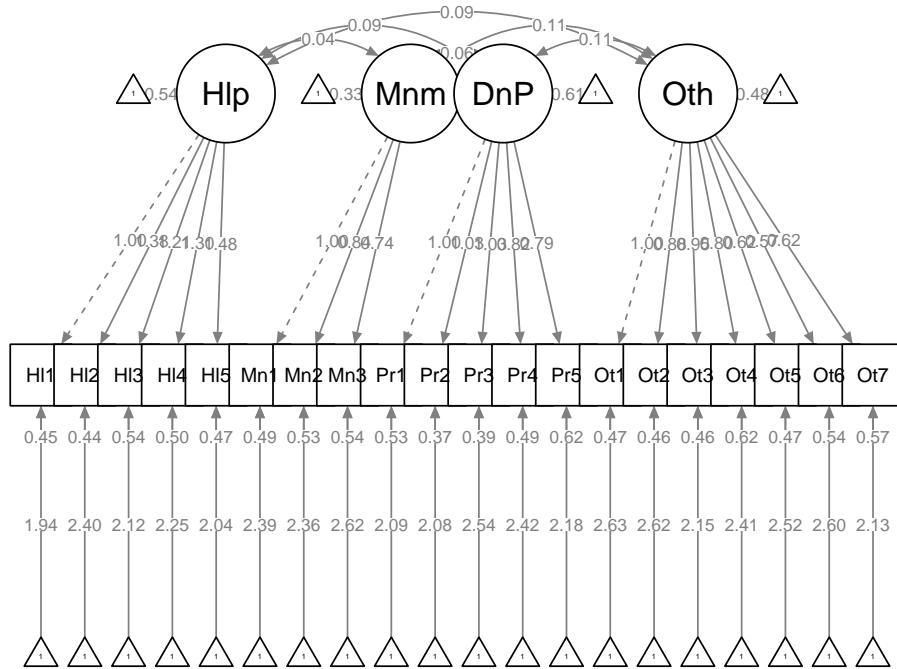
Then, export them.

```
write.csv(strictFitStats, file = "strictFitStats.csv")
write.csv(strict_paramEsts, file = "strict_paramEsts.csv")
write.csv(strictCorrs, file = "strictCorrs.csv")
```

```
# semPlot::semPaths(strict, layout = 'tree', style = 'lisrel', what =
# 'col', whatLabels = 'stand')
semPlot::semPaths(strict, layout = "tree", style = "lisrel", what = "col",
    whatLabels = "est")
```

**1**

2



```
# If R stalls, open the console. I received the instruction, 'Hit
# <Return> to see next plot:' Then it ran!
```

### 12.8.1 Interpreting the Output

Note that although the “Std.all” values differ from each other, the “Estimates” (factor loadings) are identical across Mild and Severe groups. Each also has a “label” (e.g., .p2., .p3.) which indicates that they have been constrained to be equal. The “Std.all” differ between degree of disability severity due to the difference in standard deviations of the indicators.

Criteria		Our Results	Criteria met?
Mild: factor loadings significant, strong, proper valence		Help: .44 to .57; Min: .44 to .52; Pers: .41 to .60; Oth: .36 to .58	
Severe: factor loadings significant, strong, proper valence		Help: .46 to .59; Min: .38 to .51 Pers: .43 to .62; Oth: .37 to .60	

Criteria	Our Results	Criteria met?
Yes		
Non-significant chi-square	$\chi^2(380) = 439.69, p < 0.019$	No
$CFI \geq .95$ or $CFI \geq .90$	$CFI = 0.971$	No
$RMSEA \leq .05$ (but definitely < .10)	$RMSEA = 0.019, CI90\%(0.009 \text{ to } 0.027)$	Yes(ish)
$SRMR \leq .08$ (but definitely < .10)	$SRMR = 0.046$	Yes(ish)
Combination rule: $CFI \geq .95$ and $SRMR \leq .08$	$CFI = 0.971, SRMR = 0.046$	No

### 12.8.2 Partial Write-up

**Strict invariance model.** In the strict invariance model, configuration, factor loadings, indicator means/intercepts, and residuals were constrained to be the same for each group. Fit indices were less than ideal:  $\chi^2(380) = 439.69, p < 0.019$ ,  $CFI = 0.971$ ,  $SRMR = 0.046$ ,  $RMSEA = 0.019$ , ( $90\%CI = 0.009 \text{ to } 0.027$ ). Although the non-significant chi-square difference test and the change CFI test indicate that the strong and strict invariance models are not statistically significant from each other ( $\chi^2_D(20) = 26.946, p = 0.137$ ;  $\Delta CFI = 0.004$ ) our earlier data indicated that we cannot claim invariance beyond the weak model.

```
# CFI difference test, calculated by hand
0.975 - 0.971
```

[1] 0.004

## 12.9 Partial Measurement Invariance

*Partial measurement invariance* can be seen as an intermediate state of invariance and whatever stage (beyond configural; weak, strong, strict) that the criteria for invariance is not met. For example:

- if the model failed at weak invariance, tests of partial measurement invariance could determine which factor loadings are (and are not) invariant across groups;
- if the model failed at strong invariance, tests of partial measurement invariance could determine which intercepts/means are (and are not) invariant across groups;
- (this one is aspirational) if the model failed at strict invariance, tests of partial measurement invariance could determine which residuals are (and are not) invariant across groups.

Using some of the investigative tools in *lavaan* and the associated packages, researchers can identify which elements are noninvariant. They can free the constraints until the fit statistics are acceptable and the Chi-square difference and  $\Delta CFI$  tests are no longer significant.

Conover et al. [2017] reported that the AMS was invariant at configural and weak invariance (i.e., constraining factor loadings to be equal). At the level of strong invariance (i.e., adding constraints to the intercepts), results, the majority of fit indices remained acceptable ( $CFI = .90$ ,  $RMSEA = .06$ ,  $SRMR = .08$ ). However the chi-square difference and CFI change tests were statistically significant:  $\chi^2_D(20) = 78.83, p < .01$ ;  $\Delta CFI = -.010$ . The Conover et al. article did not report further investigation regarding partial measurement invariance.

## 12.10 APA Style Write-up of the Results

As in the Conover et al. [2017] article, the write-up of invariance testing would likely be part of a multi-stage evaluation. Therefore, this section would be preceded by a variety of steps in a psychometric evaluation. Here is an example of how I might write this up.

### 12.10.1 Measurement Invariance Across Disability Severity

To test if the factor structures of the AMS were stable across disability severity, we used measurement invariance analyses. First, we constructed CFA models with the *lavaan* (v. 0.6-17) package in R and created two groups representing mild and severe. Within *lavaan* we successively constrained parameters representing the configural, weak (loadings), strong (intercepts), and strict (residuals) structures [Hirschfeld and von Brachel, Ruth, 2014, Kline, 2016]. A poor fit in any of these models suggests that the aspect being constrained does not operate consistently for the different groups. The degree of invariance was determined jointly when  $\chi^2_D, p > .05$ ; and a  $\Delta CFI < .01$ .

The configural model constrains only the relative configuration of variables in the model to be the same in both groups. In other words, no factor loadings or indicator means are constrained to be the same across groups, but the organization of indicators is the same for both groups. Next, in weak factorial invariance, the configuration of variables and all factor loadings are constrained to be the same for each group. Poor fit here suggests that the factor loadings vary in size between the two groups. In strong factorial invariance, both the configuration, factor loadings, and the indicator means are constrained to be the same for each group. A reduction in fit here, but not in the previous steps, suggests that indicators have different means in both groups, which might be expected when comparing two groups of people, such as in a t-test. Therefore, poor fit in only this model does not necessarily indicate the factor structure operates differently for different groups. Finally, strict factorial invariance requires strong invariance and equality in error variances and covariances across groups. This means that the indicators measure the same factors in each group with the same degree of precision.

We selected fit criteria for their capacity to assess different aspects of the statistical analysis. As is common among SEM researchers, we reported the chi-square goodness of fit ( $\chi^2$ ). This evaluates the discrepancy between the unrestricted sample matrix and the restricted covariance matrix. Although the associated  $p$  value indicates adequate fit when the value is non-significant, it is widely recognized that large sample size can result in a statistically significant  $p$  value [Byrne, 2016c]. The comparative fit index (CFI) is an incremental index, comparing the hypothesized model with the baseline

model, and should be at least .90 and perhaps higher than .95 [Hu and Bentler, 1999]. The root mean square error of approximation (RMSEA) takes into account the error of approximation in the population and expresses it per degree of freedom. As such, the fit indicator considers the complexity of the model. Ideal values are equal to or less than .05, values less than .08 represent reasonable fit, and values between .08 and .10 represent mediocre fit. The standardized root mean residual (SRMR) is a standardized measure of the mean absolute covariance residual – the overall difference between the observed and predicted correlations. Values greater than .10 may indicate poor fit and inspection of residuals is then advised. Researchers have advised caution when using these criteria as strict cutoffs, and other factors such as sample size and model complexity should be considered [Kline, 2016].

AMS items each were loaded on their respective correlated factors. The configural model, which constrained only the relative configuration of variables in the model to be the same in both groups had adequate fit to the data:  $\chi^2(328) = 327.57, p = 0.496$ , CFI = 1.000, SRMR = 0.034, RMSEA = 0.000, 90%CI(0.000, 0.018). The weak invariance model constrained the configuration of variables and all factor loadings to be constant across groups. Fit indices were comparable to the configural model:  $\chi^2(344) = 353.00, p = 0.357$ , CFI = 0.996, SRMR = 0.039, RMSEA = 0.008, 90%CI(0.000, 0.020). Invariance of the factor loadings was supported by the non-significant difference tests that assessed model similarity:  $\chi_D^2(16) = 25.42, p = 0.063$ ;  $\Delta CFI = 0.004$ . In the strong invariance model, configuration, factor loadings, and indicator means/intercepts were constrained to be the same for each group. Fit indices were less than ideal:  $\chi^2(360) = 412.75, p = 0.029$ , CFI = 0.975, SRMR = 0.045, RMSEA = 0.019, CI90%(0.007 to 0.027). The difference tests that evaluated model similarity suggested there was factorial noninvariance:  $(\chi_D^2(16) = 59.754, p = 0.01; \Delta CFI = 0.021)$ . Given that the  $\chi_D^2$  test is statistically significant and the  $\Delta CFI < 0.01$  we cannot claim strong invariance and we therefore do not test stricter models. Because we found noninvariance at the strong level, we did not attempt to model strict invariance.

Overall, this analysis suggests that the factor structure of the AMS was stable for mild/moderate and severe/very severe levels of disability. Figure 1 provides an illustration of the factor structure. Tables 1 and 2 provide fit indices for each of the factor structures and a summary of the measurement invariance tests.

## 12.11 Practice Problems

In each of these lessons I provide suggestions for practice that allow you to select one or more problems that are graded in difficulty. The least complex is to change the random seed and rework the problem demonstrated in the lesson. The results *should* map onto the ones obtained in the lecture.

The second option would be to adapt one of the codes in the [simluations chapter](#) to create two groups for which invariance testing would be appropriate for that measure.

As a third option, you are welcome to use data to which you have access and is suitable for invariance testing. In either case, you will be expected to:

- Specify, interpret, and write up preliminary results for CFA models that examine
  - entire sample (making no distinction between groups)
  - configural invariance
  - weak invariance
  - strong invariance
  - strict invariance
- Create an APA style results section with appropriate table(s) and figure(s)
- Talk about it with someone

### **12.11.1 Problem #1: Play around with this simulation.**

Copy the script for the simulation and then change the number in “set.seed(211023)” from 211023 to something else. Your results should parallel those obtained in the lecture, making it easier for you to check your work as you go.

### **12.11.2 Problem #2: Adapt one of the simulated data sets.**

The **Simulations** includes simulated data from many of the research vignettes used in this volume. Using guidance provided in this lesson, adapt one of those simulations to include at least two groups for which invariance testing would be appropriate.

### **12.11.3 Problem #3: Try something entirely new.**

Using data for which you have permission and access (e.g., IRB approved data you have collected or from your lab; data you simulate from a published article; data from an open science repository) complete the multi-group invariance testing process.

### **12.11.4 Grading Rubric**

Assignment Component	Points Possible	Points Earned
1. Check and, if needed, format data	5	_____
2. Specify, evaluate, and interpret the CFA for the entire sample (making no distinction between groups). Write up the preliminary results.	5	_____
3. Specify, evaluate, and interpret the CFA for configural invariance. Write up the preliminary results.	5	_____
4. Specify, evaluate, and interpret the CFA for weak invariance. Conduct the analysis to compare fit the weak and configural models. Write up the preliminary results.	5	_____

Assignment Component	Points Possible	Points Earned
5. Specify, evaluate, and interpret the CFA for strong invariance. Conduct the analysis to compare fit the strong and weak models. Write up the preliminary results.	5	_____
6. Specify, evaluate, and interpret the CFA for strict invariance. Conduct the analysis to compare fit the strict and strong models. Write up the preliminary results.	5	_____
7. Create an APA style results section. Do not report any invariance tests past the one that failed. Include a table(s) and figure(s)	10	_____
8. Explanation to grader	5	_____
<b>Totals</b>	<b>45</b>	_____

## 12.12 Homeworked Example

### Screencast Link

For more information about the data used in this homeworked example, please refer to the description and codebook located at the end of the [introduction](#) in first volume of ReCentering Psych Stats.

As a brief review, this data is part of an IRB-approved study, with consent to use in teaching demonstrations and to be made available to the general public via the open science framework. Hence, it is appropriate to use in this context. You will notice there are student- and teacher- IDs. These numbers are not actual student and teacher IDs, rather they were further re-identified so that they could not be connected to actual people.

Because this is an actual dataset, if you wish to work the problem along with me, you will need to download the [ReC.rds](#) data file from the Worked\_Examples folder in the ReC\_Psychometrics project on the GitHub.

The course evaluation items can be divided into three subscales:

- **Valued by the student** includes the items: ValObjectives, IncrUnderstanding, IncrInterest
- **Traditional pedagogy** includes the items: ClearResponsibilities, EffectiveAnswers, Feedback, ClearOrganization, ClearPresentation
- **Socially responsive pedagogy** includes the items: InclusvClassrm, EquitableEval, Mult-Perspectives, DEIIntegration

In this Homeworked Example I will assess for invariance of the a first-level, correlated factors model for the course evaluation data across students in the Clinical Psychology (CPY) and Industrial-Organizational Psychology (ORG) doctoral programs, asking “Is the structure of the course evaluation item data invariant across CPY and IOP?”

Check and, if needed, format data

```
big <- readRDS("ReC.rds")
```

The invariance testing will be easiest if I trim the dataset to the items and the grouping variable (i.e., Dept).

```
library(tidyverse)
items <- big %>%
  dplyr::select(Dept, ValObjectives, IncrUnderstanding, IncrInterest,
    ClearResponsibilities, EffectiveAnswers, Feedback, ClearOrganization,
    ClearPresentation, MultPerspectives, InclusvClassrm, DEIIntegration,
    EquitableEval)
```

Let's check the structure of the data. The items should be numerical or integer; the Dept variable must be a factor.

```
str(items)
```

```
Classes 'data.table' and 'data.frame': 310 obs. of 13 variables:
 $ Dept           : chr "CPY" "CPY" "CPY" "CPY" ...
 $ ValObjectives  : int 5 5 4 4 5 5 5 5 4 5 ...
 $ IncrUnderstanding : int 2 3 4 3 4 4 5 2 4 5 ...
 $ IncrInterest   : int 5 3 4 2 4 3 5 3 2 5 ...
 $ ClearResponsibilities: int 5 5 4 4 5 4 5 4 4 5 ...
 $ EffectiveAnswers : int 5 3 5 3 5 3 4 3 2 3 ...
 $ Feedback        : int 5 3 4 2 5 NA 5 4 4 5 ...
 $ ClearOrganization : int 3 4 3 4 4 4 5 4 4 5 ...
 $ ClearPresentation : int 4 4 4 2 5 3 4 4 4 5 ...
 $ MultPerspectives  : int 5 5 4 5 5 4 5 5 5 5 ...
 $ InclusvClassrm   : int 5 5 5 5 5 4 5 5 4 5 ...
 $ DEIIntegration   : int 5 5 5 5 5 4 5 5 5 5 ...
 $ EquitableEval     : int 5 5 3 5 5 3 5 5 3 5 ...
 - attr(*, ".internal.selfref")=<externalptr>
```

Let's change the Dept variable to be a factor.

```
items$Dept <- as.factor(items$Dept)
```

Specify, evaluate, and interpret the CFA for the entire sample (making no distinction between groups). Write up the preliminary results.

```

corrF <- "TradPed =~ ClearResponsibilities + EffectiveAnswers + Feedback + ClearOrganization +
          Valued =~ ValObjectives + IncrUnderstanding + IncrInterest
          SCRPed =~ MultPerspectives + InclusvClassrm + DEIintegration + EquitableEval

          TradPed~~Valued
          TradPed~~SCRPed
          Valued~~SCRPed
"

```

```

set.seed(240505)
corrF_fit <- lavaan:::cfa(corrF, data = items)
lavaan:::summary(corrF_fit, fit.measures = TRUE, standardized = TRUE, rsquare = TRUE)

```

`lavaan 0.6.17 ended normally after 42 iterations`

Estimator	ML
Optimization method	NLMINB
Number of model parameters	27
	Used      Total
Number of observations	267      310

Model Test User Model:

Test statistic	224.795
Degrees of freedom	51
P-value (Chi-square)	0.000

Model Test Baseline Model:

Test statistic	1940.157
Degrees of freedom	66
P-value	0.000

User Model versus Baseline Model:

Comparative Fit Index (CFI)	0.907
Tucker-Lewis Index (TLI)	0.880

Loglikelihood and Information Criteria:

Loglikelihood user model (H0)	-2977.975
Loglikelihood unrestricted model (H1)	-2865.578
Akaike (AIC)	6009.951
Bayesian (BIC)	6106.807

Sample-size adjusted Bayesian (SABIC) 6021.201

Root Mean Square Error of Approximation:

RMSEA	0.113
90 Percent confidence interval - lower	0.098
90 Percent confidence interval - upper	0.128
P-value H_0: RMSEA <= 0.050	0.000
P-value H_0: RMSEA >= 0.080	1.000

Standardized Root Mean Square Residual:

SRMR	0.061
------	-------

Parameter Estimates:

Standard errors	Standard
Information	Expected
Information saturated (h1) model	Structured

Latent Variables:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
<b>TradPed</b> =~						
ClearRspnsblts	1.000				0.652	0.826
EffectivAnswrs	1.015	0.065	15.606	0.000	0.662	0.815
Feedback	1.010	0.075	13.481	0.000	0.659	0.735
ClearOrganiztn	1.295	0.086	15.106	0.000	0.845	0.797
ClearPresenttn	1.204	0.072	16.680	0.000	0.785	0.853
<b>Valued</b> =~						
Val0bjectives	1.000				0.334	0.557
IncrUndrstndng	1.942	0.223	8.717	0.000	0.649	0.786
IncrInterest	2.438	0.273	8.932	0.000	0.815	0.844
<b>SCRPed</b> =~						
MultPerspectvs	1.000				0.713	0.846
InclusvClassrm	0.622	0.053	11.672	0.000	0.444	0.682
DEIIntegration	0.589	0.063	9.365	0.000	0.420	0.567
EquitableEval	0.642	0.052	12.410	0.000	0.458	0.717

Covariances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
<b>TradPed</b> ~~						
Valued	0.171	0.026	6.640	0.000	0.785	0.785
SCRPed	0.391	0.045	8.677	0.000	0.841	0.841
<b>Valued</b> ~~						
SCRPed	0.164	0.026	6.254	0.000	0.688	0.688

Variances:

Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
----------	---------	---------	---------	--------	---------

.ClearRspnsblts	0.199	0.021	9.456	0.000	0.199	0.319
.EffectivAnswrs	0.222	0.023	9.618	0.000	0.222	0.336
.Feedback	0.371	0.036	10.415	0.000	0.371	0.460
.ClearOrganiztn	0.410	0.042	9.855	0.000	0.410	0.365
.ClearPresenttn	0.232	0.026	8.939	0.000	0.232	0.273
.ValObjectives	0.248	0.023	10.650	0.000	0.248	0.690
.IncrUndrstndng	0.260	0.032	8.041	0.000	0.260	0.382
.IncrInterest	0.268	0.043	6.308	0.000	0.268	0.288
.MultPerspectvs	0.203	0.029	7.052	0.000	0.203	0.285
.InclusvClassrm	0.226	0.023	10.028	0.000	0.226	0.534
.DEIIntegration	0.371	0.035	10.734	0.000	0.371	0.678
.EquitableEval	0.198	0.020	9.685	0.000	0.198	0.486
TradPed	0.426	0.053	8.085	0.000	1.000	1.000
Valued	0.112	0.024	4.595	0.000	1.000	1.000
SCRPed	0.509	0.063	8.039	0.000	1.000	1.000

R-Square:

	Estimate
ClearRspnsblts	0.681
EffectivAnswrs	0.664
Feedback	0.540
ClearOrganiztn	0.635
ClearPresenttn	0.727
ValObjectives	0.310
IncrUndrstndng	0.618
IncrInterest	0.712
MultPerspectvs	0.715
InclusvClassrm	0.466
DEIIntegration	0.322
EquitableEval	0.514

Among my first steps are also to write the code to export the results. The *tidySEM* package has useful functions to export the fit statistics, parameter estimates, and correlations among the latent variables (i.e., factors).

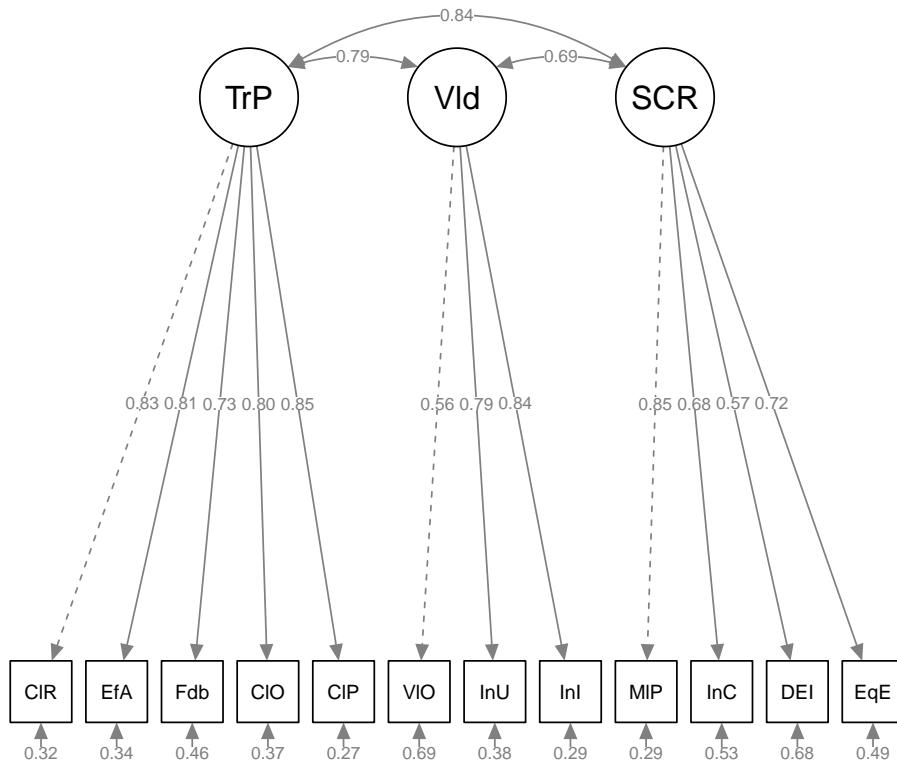
```
corrF_FitStats <- tidySEM:::table_fit(corrF_fit)
corrF_paramEsts <- tidySEM:::table_results(corrF_fit, digits = 3, columns = NULL)
corrF_Corrs <- tidySEM:::table_cors(corrF_fit, digits = 3)
# to see each of the tables, remove the hashtag corrF_FitStats
# bifacF_paramEsts bifacFCorrs
```

Next, I export them.

```
write.csv(corrF_FitStats, file = "corrF_FitStats.csv")
write.csv(corrF_paramEsts, file = "corrF_paramEsts.csv")
write.csv(corrF_Corrs, file = "corrF_Corrs")
```

**Troubleshooting tip:** If, while working with this function you get the error: “*Error in file(file, ifelse(append, “a”, “w”)) : cannot open the connection*”, it’s because the .csv file that received your table is still open. R is just trying to write over it. A similar error happens when knitting.

```
semPlot::semPaths(corrF_fit, layout = "tree", style = "lisrel", what = "col",
  whatLabels = "stand")
```



We evaluated a single-order, multidimensional model where each of the 12 items loaded onto one of three correlated factors. Standardized pattern coefficients ranged between 0.74 and 0.85 on the TradPed factor, between 0.56 and 0.84 on the Valued factor, and between 0.57 and 0.85 on the SCRPed factor. The Chi-square index was statistically significant ( $\chi^2(51) = 224.795, p < 0.001$ ) indicating some degree of misfit. The CFI value of 0.91 exceeded the bare minimum recommendation of 0.90. The RMSEA = 0.113 (90% CI [0.098, 0.128]) was higher than recommended value of 0.10. The SRMR value of 0.061 remained below the warning criteria of 0.10. The AIC and BIC values were 6009.95 and 6106.81, respectively.

At the outset, let me acknowledge that starting the invariance testing with fit indices that are on the margins of acceptability suggests that we may not get very far.

**Specify, evaluate, and interpret the CFA for configural invariance. Write up the preliminary results.**

The only addition to the prior code is to add the grouping variable.

```
configural <- lavaan:::cfa(corrF, data = items, group = "Dept")
lavaan:::summary(configural, fit.measures = TRUE, standardized = TRUE)
```

lavaan 0.6.17 ended normally after 71 iterations

Estimator	ML
Optimization method	NLMINB
Number of model parameters	78
Number of observations per group:	
CPY	Used
ORG	Total
	202
	65
	232
	78

Model Test User Model:

Test statistic	339.428
Degrees of freedom	102
P-value (Chi-square)	0.000
Test statistic for each group:	
CPY	202.821
ORG	136.608

Model Test Baseline Model:

Test statistic	2156.987
Degrees of freedom	132
P-value	0.000

User Model versus Baseline Model:

Comparative Fit Index (CFI)	0.883
Tucker-Lewis Index (TLI)	0.848

Loglikelihood and Information Criteria:

Loglikelihood user model (H0)	-2897.967
Loglikelihood unrestricted model (H1)	-2728.252
Akaike (AIC)	5951.933

Bayesian (BIC)	6231.739
Sample-size adjusted Bayesian (SABIC)	5984.433

Root Mean Square Error of Approximation:

RMSEA	0.132
90 Percent confidence interval - lower	0.117
90 Percent confidence interval - upper	0.148
P-value H_0: RMSEA <= 0.050	0.000
P-value H_0: RMSEA >= 0.080	1.000

Standardized Root Mean Square Residual:

SRMR	0.073
------	-------

Parameter Estimates:

Standard errors	Standard
Information	Expected
Information saturated (h1) model	Structured

Group 1 [CPY]:

Latent Variables:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
TradPed =~						
ClearRspnsblts	1.000				0.580	0.764
EffectivAnshrs	1.017	0.087	11.659	0.000	0.590	0.782
Feedback	1.048	0.098	10.661	0.000	0.608	0.724
ClearOrganiztn	1.393	0.121	11.540	0.000	0.808	0.775
ClearPresenttn	1.380	0.105	13.134	0.000	0.801	0.865
Valued =~						
Val0bjectives	1.000				0.360	0.603
IncrUndrstndng	1.960	0.226	8.668	0.000	0.705	0.820
IncrInterest	2.251	0.259	8.688	0.000	0.810	0.823
SCRPed =~						
MultPerspectvs	1.000				0.681	0.875
InclusvClassrm	0.830	0.061	13.701	0.000	0.566	0.812
DEIIntegration	0.736	0.072	10.187	0.000	0.501	0.656
EquitableEval	0.590	0.056	10.501	0.000	0.402	0.671

Covariances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
TradPed ~~						
Valued	0.185	0.030	6.227	0.000	0.885	0.885
SCRPed	0.329	0.045	7.373	0.000	0.833	0.833
Valued ~~						

SCRPed	0.160	0.028	5.663	0.000	0.653	0.653
--------	-------	-------	-------	-------	-------	-------

## Intercepts:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
.ClearRspnsblts	4.510	0.053	84.410	0.000	4.510	5.939
.EffectivAnswrs	4.416	0.053	83.154	0.000	4.416	5.851
.Feedback	4.327	0.059	73.232	0.000	4.327	5.153
.ClearOrganiztn	4.109	0.073	56.009	0.000	4.109	3.941
.ClearPresenttn	4.287	0.065	65.775	0.000	4.287	4.628
.ValObjectives	4.554	0.042	108.498	0.000	4.554	7.634
.IncrUnderstndng	4.287	0.060	70.863	0.000	4.287	4.986
.IncrInterest	3.941	0.069	56.961	0.000	3.941	4.008
.MultPerspectvs	4.490	0.055	81.955	0.000	4.490	5.766
.InclusvClassrm	4.649	0.049	94.832	0.000	4.649	6.672
.DEIIntegration	4.550	0.054	84.599	0.000	4.550	5.952
.EquitableEval	4.604	0.042	109.361	0.000	4.604	7.695

## Variances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
.ClearRspnsblts	0.240	0.027	8.916	0.000	0.240	0.416
.EffectivAnswrs	0.221	0.025	8.775	0.000	0.221	0.388
.Feedback	0.335	0.037	9.163	0.000	0.335	0.475
.ClearOrganiztn	0.434	0.049	8.831	0.000	0.434	0.399
.ClearPresenttn	0.217	0.028	7.607	0.000	0.217	0.252
.ValObjectives	0.227	0.025	9.223	0.000	0.227	0.636
.IncrUnderstndng	0.242	0.035	6.920	0.000	0.242	0.328
.IncrInterest	0.311	0.046	6.832	0.000	0.311	0.322
.MultPerspectvs	0.143	0.024	5.963	0.000	0.143	0.235
.InclusvClassrm	0.166	0.022	7.588	0.000	0.166	0.341
.DEIIntegration	0.333	0.036	9.125	0.000	0.333	0.570
.EquitableEval	0.197	0.022	9.045	0.000	0.197	0.550
TradPed	0.337	0.054	6.279	0.000	1.000	1.000
Valued	0.129	0.029	4.462	0.000	1.000	1.000
SCRPed	0.464	0.062	7.515	0.000	1.000	1.000

## Group 2 [ORG]:

## Latent Variables:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
TradPed ==						
ClearRspnsblts	1.000				0.810	0.924
EffectivAnswrs	1.101	0.086	12.866	0.000	0.892	0.920
Feedback	0.875	0.129	6.808	0.000	0.709	0.682
ClearOrganiztn	1.096	0.118	9.325	0.000	0.888	0.810
ClearPresenttn	1.015	0.081	12.502	0.000	0.822	0.911
Valued ==						
ValObjectives	1.000				0.224	0.366

IncrUndrstndng	2.193	0.824	2.661	0.008	0.490	0.693
IncrInterest	3.450	1.262	2.735	0.006	0.771	0.865
SCRPed =~						
MultPerspectvs	1.000				0.901	0.895
InclusvClassrm	0.146	0.068	2.155	0.031	0.131	0.278
DEIIntegration	0.242	0.093	2.596	0.009	0.218	0.331
EquitableEval	0.569	0.095	6.002	0.000	0.513	0.687

## Covariances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
TradPed ~~						
Valued	0.106	0.047	2.263	0.024	0.586	0.586
SCRPed	0.577	0.128	4.497	0.000	0.791	0.791
Valued ~~						
SCRPed	0.161	0.067	2.415	0.016	0.798	0.798

## Intercepts:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
.ClearRspnsblts	4.431	0.109	40.755	0.000	4.431	5.055
.EffectivAnswrs	4.369	0.120	36.321	0.000	4.369	4.505
.Feedback	4.108	0.129	31.857	0.000	4.108	3.951
.ClearOrganiztn	3.892	0.136	28.602	0.000	3.892	3.548
.ClearPresenttn	4.215	0.112	37.642	0.000	4.215	4.669
.ValObjectives	4.523	0.076	59.745	0.000	4.523	7.410
.IncrUndrstndng	4.338	0.088	49.425	0.000	4.338	6.130
.IncrInterest	4.138	0.111	37.392	0.000	4.138	4.638
.MultPerspectvs	4.308	0.125	34.506	0.000	4.308	4.280
.InclusvClassrm	4.738	0.059	80.735	0.000	4.738	10.014
.DEIIntegration	4.523	0.082	55.348	0.000	4.523	6.865
.EquitableEval	4.523	0.093	48.854	0.000	4.523	6.060

## Variances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
.ClearRspnsblts	0.112	0.028	3.960	0.000	0.112	0.146
.EffectivAnswrs	0.145	0.036	4.070	0.000	0.145	0.154
.Feedback	0.578	0.106	5.453	0.000	0.578	0.535
.ClearOrganiztn	0.415	0.080	5.153	0.000	0.415	0.345
.ClearPresenttn	0.139	0.033	4.260	0.000	0.139	0.170
.ValObjectives	0.323	0.058	5.527	0.000	0.323	0.866
.IncrUndrstndng	0.260	0.058	4.492	0.000	0.260	0.520
.IncrInterest	0.201	0.092	2.182	0.029	0.201	0.252
.MultPerspectvs	0.202	0.088	2.294	0.022	0.202	0.199
.InclusvClassrm	0.207	0.037	5.640	0.000	0.207	0.923
.DEIIntegration	0.386	0.069	5.611	0.000	0.386	0.890
.EquitableEval	0.294	0.059	5.009	0.000	0.294	0.528
TradPed	0.656	0.135	4.867	0.000	1.000	1.000
Valued	0.050	0.036	1.400	0.161	1.000	1.000
SCRPed	0.811	0.192	4.229	0.000	1.000	1.000

Among my first steps are also to write the code to export the results. The *tidySEM* package has useful functions to export the fit statistics, parameter estimates, and correlations among the latent variables (i.e., factors).

```
configural_FitStats <- tidySEM::table_fit(configural)
configural_paramEsts <- tidySEM::table_results(configural, digits = 3,
    columns = NULL)
configural_Corrs <- tidySEM::table_corrs(configural, digits = 3)
# to see each of the tables, remove the hashtag configural_FitStats
# configural_paramEsts configural_Corrs
```

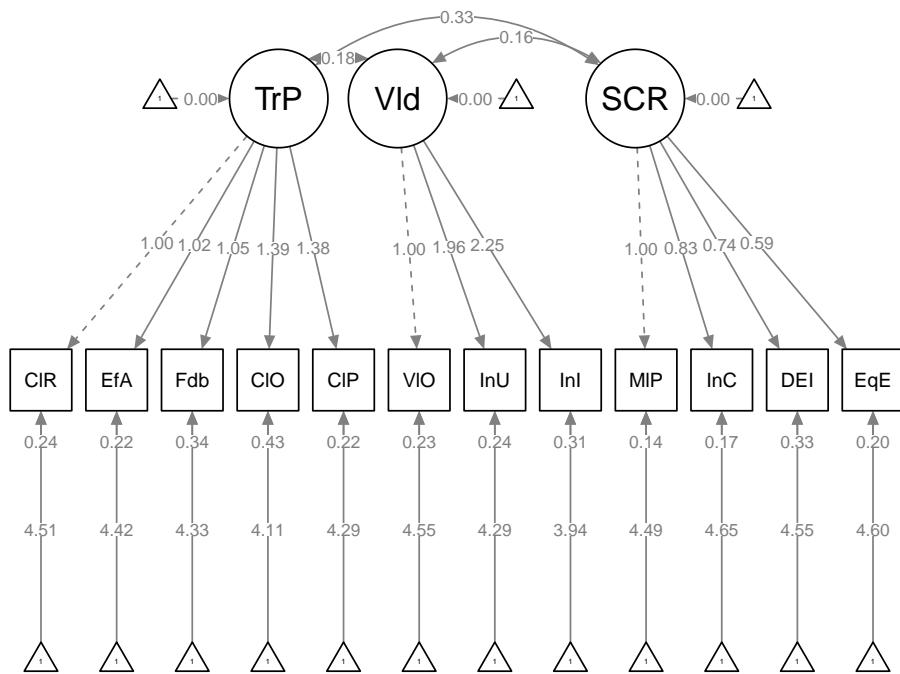
Next, I export them.

```
write.csv(configural_FitStats, file = "configural_FitStats.csv")
write.csv(configural_paramEsts, file = "configural_paramEsts.csv")
write.csv(configural_Corrs, file = "configural_Corrs.csv")
```

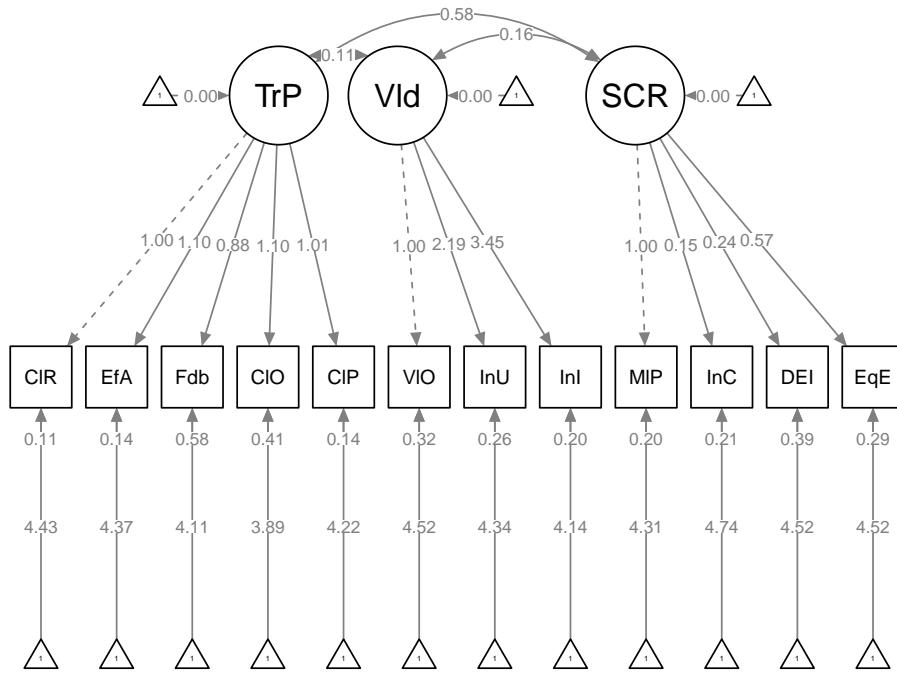
We can also plot our work

```
semPlot::semPaths(configural, layout = "tree", style = "lisrel", what = "col",
    whatLabels = "est")
```

1



2



```
# If the procedure stalls open the Consult and follow the
# instructions to Hit Return to see the plot
```

**Configural Model.** The configural model, which constrained only the relative configuration of variables in the model to be the same in both groups had less than adequate fit to the data:  $\chi^2(102) = 339.43, p < 0.001, CFI = 0.883, SRMR = 0.073, RMSEA = 0.132, 90$ . For CPY, standardized pattern coefficients ranged between 0.72 and 0.87 on the TradPed factor, between 0.60 and 0.82 on the Valued factor, and between 0.66 and 0.88 on the SCRPed factor. For ORG, standardized pattern coefficients ranged between 0.58 and 0.92 for the TradPed factor, between 0.37 and 0.87 on the Valued factor, and between 0.28 and 0.90 on the SCRPed factor.

At this point we notice that the factor loadings for CPY are consistently above 0.60. In contrast, some factor loadings dip as low as 0.28. Excepting the SRMR (which is below the 0.08) the remaining fit indices would suggest that we cannot claim configural invariance between these two groups. Because this is a demonstration, I will continue with all of the steps.

Specify, evaluate, and interpret the CFA for the entire sample (making no distinction between groups). Write up the preliminary results. {-} Specify, evaluate, and interpret the CFA for weak invariance. Conduct the analysis to compare fit the weak and configural models. Write up the preliminary results.

To the code we wrote for testing configural invariance, we add the *group.equal = "loadings"* command.

```
weak <- lavaan::cfa(corrF, data = items, group = "Dept", group.equal = "loadings")
lavaan::summary(weak, fit.measures = TRUE, standardized = TRUE)
```

lavaan 0.6.17 ended normally after 55 iterations

Estimator	ML
Optimization method	NLMINB
Number of model parameters	78
Number of equality constraints	9
Number of observations per group:	Used Total
CPY	202 232
ORG	65 78

Model Test User Model:

Test statistic	400.290
Degrees of freedom	111
P-value (Chi-square)	0.000
Test statistic for each group:	
CPY	213.288
ORG	187.001

Model Test Baseline Model:

Test statistic	2156.987
Degrees of freedom	132
P-value	0.000

User Model versus Baseline Model:

Comparative Fit Index (CFI)	0.857
Tucker-Lewis Index (TLI)	0.830

Loglikelihood and Information Criteria:

Loglikelihood user model (H0)	-2928.397
Loglikelihood unrestricted model (H1)	-2728.252
Akaike (AIC)	5994.795

Bayesian (BIC)	6242.315
Sample-size adjusted Bayesian (SABIC)	6023.544

Root Mean Square Error of Approximation:

RMSEA	0.140
90 Percent confidence interval - lower	0.125
90 Percent confidence interval - upper	0.155
P-value H_0: RMSEA <= 0.050	0.000
P-value H_0: RMSEA >= 0.080	1.000

Standardized Root Mean Square Residual:

SRMR	0.101
------	-------

Parameter Estimates:

Standard errors	Standard
Information	Expected
Information saturated (h1) model	Structured

Group 1 [CPY]:

Latent Variables:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
TradPed =~						
ClrRspn	1.000				0.619	0.788
EffctvA (.p2.)	1.041	0.062	16.874	0.000	0.644	0.811
Feedbck (.p3.)	0.977	0.074	13.277	0.000	0.605	0.726
ClrOrgn (.p4.)	1.260	0.083	15.218	0.000	0.780	0.762
ClrPrsn (.p5.)	1.190	0.066	17.996	0.000	0.737	0.834
Valued =~						
V10bjct	1.000				0.355	0.598
IncrUnd (.p7.)	1.952	0.214	9.121	0.000	0.693	0.812
IncrInt (.p8.)	2.330	0.253	9.226	0.000	0.827	0.834
SCRPed =~						
MltPrsp	1.000				0.696	0.881
InclsC (.10.)	0.743	0.055	13.417	0.000	0.517	0.773
DEIntgr (.11.)	0.693	0.066	10.435	0.000	0.483	0.638
EqtblEv (.12.)	0.626	0.053	11.750	0.000	0.436	0.707

Covariances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
TradPed ~~						
Valued	0.192	0.029	6.542	0.000	0.875	0.875
SCRPed	0.362	0.046	7.837	0.000	0.840	0.840
Valued ~~						

SCRPed	0.163	0.028	5.783	0.000	0.658	0.658
--------	-------	-------	-------	-------	-------	-------

## Intercepts:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
.ClearRspnsblts	4.510	0.055	81.633	0.000	4.510	5.744
.EffectivAnswrs	4.416	0.056	78.997	0.000	4.416	5.558
.Feedback	4.327	0.059	73.822	0.000	4.327	5.194
.ClearOrganiztn	4.109	0.072	57.069	0.000	4.109	4.015
.ClearPresenttn	4.287	0.062	69.002	0.000	4.287	4.855
.ValObjectives	4.554	0.042	108.931	0.000	4.554	7.664
.IncrUnderstndng	4.287	0.060	71.430	0.000	4.287	5.026
.IncrInterest	3.941	0.070	56.440	0.000	3.941	3.971
.MultPerspectvs	4.490	0.056	80.795	0.000	4.490	5.685
.InclusvClassrm	4.649	0.047	98.789	0.000	4.649	6.951
.DEIIntegration	4.550	0.053	85.411	0.000	4.550	6.009
.EquitableEval	4.604	0.043	106.079	0.000	4.604	7.464

## Variances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
.ClearRspnsblts	0.233	0.027	8.729	0.000	0.233	0.379
.EffectivAnswrs	0.216	0.025	8.490	0.000	0.216	0.342
.Feedback	0.328	0.036	9.146	0.000	0.328	0.473
.ClearOrganiztn	0.439	0.049	8.932	0.000	0.439	0.419
.ClearPresenttn	0.237	0.029	8.188	0.000	0.237	0.304
.ValObjectives	0.227	0.025	9.243	0.000	0.227	0.643
.IncrUnderstndng	0.247	0.035	7.081	0.000	0.247	0.340
.IncrInterest	0.300	0.046	6.546	0.000	0.300	0.305
.MultPerspectvs	0.139	0.024	5.692	0.000	0.139	0.223
.InclusvClassrm	0.180	0.022	8.183	0.000	0.180	0.402
.DEIIntegration	0.340	0.037	9.218	0.000	0.340	0.594
.EquitableEval	0.190	0.022	8.816	0.000	0.190	0.500
TradPed	0.383	0.052	7.385	0.000	1.000	1.000
Valued	0.126	0.027	4.619	0.000	1.000	1.000
SCRPed	0.485	0.063	7.656	0.000	1.000	1.000

## Group 2 [ORG]:

## Latent Variables:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
TradPed ==~						
ClrRspn	1.000				0.760	0.907
EffctvA (.p2.)	1.041	0.062	16.874	0.000	0.792	0.889
Feedbck (.p3.)	0.977	0.074	13.277	0.000	0.743	0.695
ClrOrgn (.p4.)	1.260	0.083	15.218	0.000	0.958	0.835
ClrPrsn (.p5.)	1.190	0.066	17.996	0.000	0.905	0.926
Valued ==~						
VlObjct	1.000				0.285	0.460

IncrUnd (.p7.)	1.952	0.214	9.121	0.000	0.556	0.761
IncrInt (.p8.)	2.330	0.253	9.226	0.000	0.664	0.772
SCRPed =~						
MltPrsp	1.000				0.534	0.594
InclsC (.10.)	0.743	0.055	13.417	0.000	0.397	0.669
DEIntgr (.11.)	0.693	0.066	10.435	0.000	0.370	0.542
EqtblEv (.12.)	0.626	0.053	11.750	0.000	0.335	0.506

## Covariances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
TradPed ~~						
Valued	0.126	0.038	3.361	0.001	0.583	0.583
SCRPed	0.272	0.073	3.722	0.000	0.669	0.669
Valued ~~						
SCRPed	0.113	0.032	3.518	0.000	0.740	0.740

## Intercepts:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
.ClearRspnsblts	4.431	0.104	42.624	0.000	4.431	5.287
.EffectivAnswrs	4.369	0.110	39.553	0.000	4.369	4.906
.Feedback	4.108	0.133	30.969	0.000	4.108	3.841
.ClearOrganiztn	3.892	0.142	27.348	0.000	3.892	3.392
.ClearPresenttn	4.215	0.121	34.783	0.000	4.215	4.314
.ValObjectives	4.523	0.077	58.826	0.000	4.523	7.296
.IncrUnderstndng	4.338	0.091	47.876	0.000	4.338	5.938
.IncrInterest	4.138	0.107	38.771	0.000	4.138	4.809
.MultPerspectvs	4.308	0.112	38.629	0.000	4.308	4.791
.InclusvClassrm	4.738	0.074	64.420	0.000	4.738	7.990
.DEIIntegration	4.523	0.085	53.338	0.000	4.523	6.616
.EquitableEval	4.523	0.082	55.126	0.000	4.523	6.837

## Variances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
.ClearRspnsblts	0.124	0.029	4.221	0.000	0.124	0.177
.EffectivAnswrs	0.166	0.037	4.519	0.000	0.166	0.210
.Feedback	0.591	0.109	5.419	0.000	0.591	0.517
.ClearOrganiztn	0.399	0.080	4.990	0.000	0.399	0.303
.ClearPresenttn	0.136	0.036	3.796	0.000	0.136	0.142
.ValObjectives	0.303	0.056	5.377	0.000	0.303	0.789
.IncrUnderstndng	0.224	0.058	3.901	0.000	0.224	0.420
.IncrInterest	0.300	0.079	3.776	0.000	0.300	0.405
.MultPerspectvs	0.523	0.106	4.920	0.000	0.523	0.647
.InclusvClassrm	0.194	0.043	4.480	0.000	0.194	0.552
.DEIIntegration	0.330	0.065	5.109	0.000	0.330	0.706
.EquitableEval	0.326	0.062	5.221	0.000	0.326	0.744
TradPed	0.578	0.115	5.041	0.000	1.000	1.000
Valued	0.081	0.024	3.328	0.001	1.000	1.000
SCRPed	0.286	0.080	3.590	0.000	1.000	1.000

Let's write and export the data into .csv files that we can view through Excel.

```
weak_FitStats <- tidySEM::table_fit(weak)
weak_paramEsts <- tidySEM::table_results(weak, digits = 3, columns = NULL)
weak_Corrs <- tidySEM::table_corrs(weak, digits = 3)
# to see each of the tables, remove the hashtag weak_FitStats
# weak_paramEsts weak_Corrs

write.csv(weak_FitStats, file = "weak_FitStats.csv")
write.csv(weak_paramEsts, file = "weak_paramEsts.csv")
write.csv(weak_Corrs, file = "weak_Corrs.csv")
```

with the `lavaan::anova` function we can formally compare the configural and weak tests.

```
lavaan::anova(configural, weak)
```

#### Chi-Squared Difference Test

	Df	AIC	BIC	Chisq	Chisq diff	RMSEA	Df diff	Pr(>Chisq)			
configural	102	5951.9	6231.7	339.43							
weak	111	5994.8	6242.3	400.29	60.862	0.20776	9	0.000000009145			
configural											
weak				***							
---											
Signif. codes:	0	'***'	0.001	'**'	0.01	'*'	0.05	'. '	0.1	' '	1

The chi-square difference tests indicates that they are statistically significantly different from each other:  $\chi^2_D(9) = 60.86, p < .001$ .

Using the CFI values from the configural and weak models we hand-calculate (see below) the change CFI statistic:  $\Delta CFI = .026$ . These statistically significant differences suggest that the weak invariance model is worse.

```
#CFI diff
.883 - .857
```

[1] 0.026

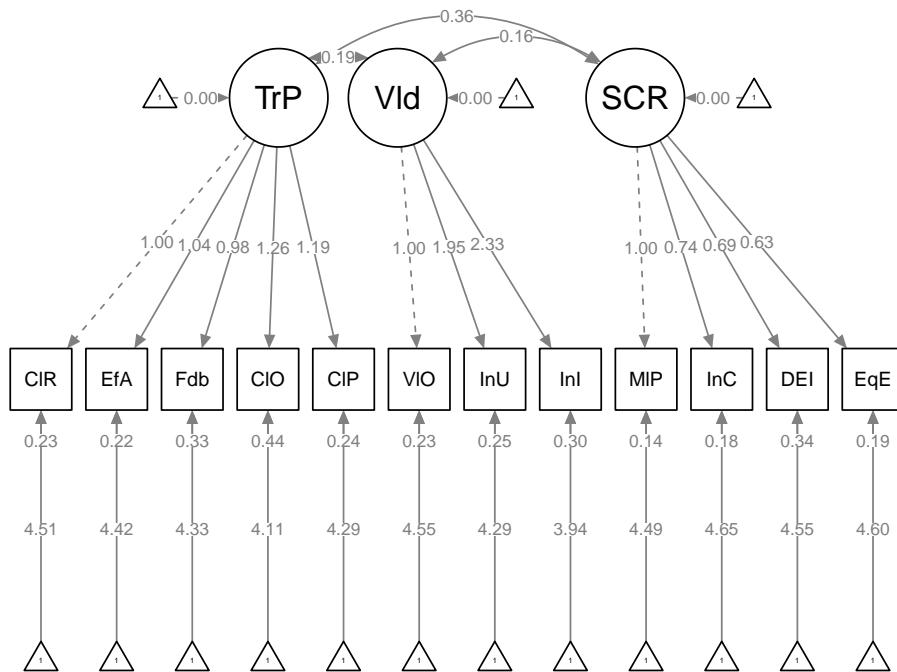
**Weak invariance model.** The weak invariance model constrained the configuration of variables and all factor loadings to be constant across groups. Fit suggested poor fit:  $\chi^2(111) = 400.29, p < 0.001, CFI = 0.857, SRMR = 0.101, RMSEA = 0.140(90)$ . Noninvariance of the factor loadings was supported by the significant difference tests that assessed model similarity:  $\chi^2_D(9) = 60.86, p < .001; \Delta CFI = .026$ . For CPY, standardized pattern coefficients ranged between 0.73 and 0.83 on the TradPed factor,

between 0.60 and 0.83 on the Valued factor, and between 0.64 and 0.88 on the SCRPed factor. For ORG, standardized pattern coefficients ranged between 0.70 and 0.93 for the TradPed factor, between 0.46 and 0.77 on the Valued factor, and between 0.54 and 0.67 on the SCRPed factor.

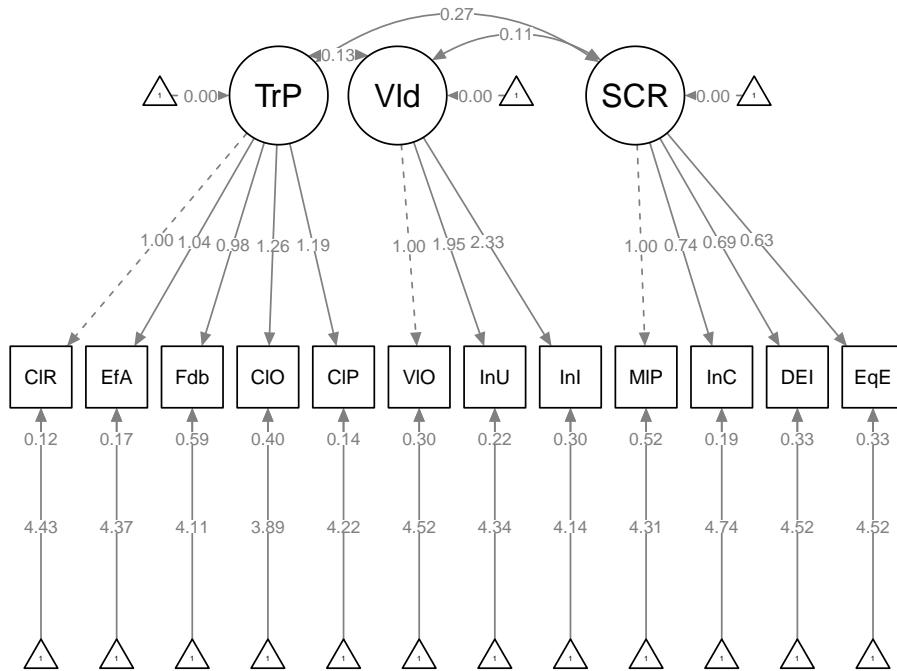
As noted before, we could not claim configural invariance, so we must conclude that this model is also noninvariant.

```
semPlot::semPaths(weak, layout = "tree", style = "lisrel", what = "col",
  whatLabels = "est")
```

**1**



2



```
# Remember to open the consult and hit ENTER twice to view the plots
```

Specify, evaluate, and interpret the CFA for strong invariance. Conduct the analysis to compare fit the strong and weak models. Write up the preliminary results.

To the weak invariance code we add “intercepts” to a concatenated list in the *group.equal* command.

```
strong <- lavaan:::cfa(corrF, data = items, group = "Dept", group.equal = c("loadings",
  "intercepts"))
lavaan:::summary(strong, fit.measures = TRUE, standardized = TRUE)
```

lavaan 0.6.17 ended normally after 107 iterations

Estimator

Optimization method

ML

NLMINB

Number of model parameters	81	
Number of equality constraints	21	
Number of observations per group:	Used	Total
CPY	202	232
ORG	65	78

Model Test User Model:

Test statistic	413.226
Degrees of freedom	120
P-value (Chi-square)	0.000
Test statistic for each group:	
CPY	215.909
ORG	197.317

Model Test Baseline Model:

Test statistic	2156.987
Degrees of freedom	132
P-value	0.000

User Model versus Baseline Model:

Comparative Fit Index (CFI)	0.855
Tucker-Lewis Index (TLI)	0.841

Loglikelihood and Information Criteria:

Loglikelihood user model (H0)	-2934.866
Loglikelihood unrestricted model (H1)	-2728.252
Akaike (AIC)	5989.731
Bayesian (BIC)	6204.966
Sample-size adjusted Bayesian (SABIC)	6014.731

Root Mean Square Error of Approximation:

RMSEA	0.135
90 Percent confidence interval - lower	0.121
90 Percent confidence interval - upper	0.150
P-value H_0: RMSEA <= 0.050	0.000
P-value H_0: RMSEA >= 0.080	1.000

Standardized Root Mean Square Residual:

SRMR	0.103
------	-------

## Parameter Estimates:

Standard errors	Standard
Information	Expected
Information saturated (h1) model	Structured

## Group 1 [CPY]:

## Latent Variables:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
<b>TradPed =~</b>						
ClrRspn	1.000				0.619	0.788
EffctvA (.p2.)	1.039	0.062	16.868	0.000	0.643	0.810
Feedbck (.p3.)	0.982	0.074	13.309	0.000	0.607	0.728
ClrOrgn (.p4.)	1.265	0.083	15.243	0.000	0.783	0.763
ClrPrsn (.p5.)	1.188	0.066	18.011	0.000	0.735	0.833
<b>Valued =~</b>						
VlObjct	1.000				0.353	0.595
IncrUnd (.p7.)	1.960	0.216	9.074	0.000	0.692	0.812
IncrInt (.p8.)	2.351	0.256	9.185	0.000	0.830	0.835
<b>SCRPed =~</b>						
MltPrsp	1.000				0.697	0.882
InclsvC (.10.)	0.740	0.055	13.350	0.000	0.515	0.771
DEIntgr (.11.)	0.690	0.066	10.401	0.000	0.481	0.636
EqtblEv (.12.)	0.626	0.053	11.757	0.000	0.436	0.707

## Covariances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
<b>TradPed ~~</b>						
Valued	0.191	0.029	6.525	0.000	0.875	0.875
SCRPed	0.363	0.046	7.840	0.000	0.841	0.841
<b>Valued ~~</b>						
SCRPed	0.162	0.028	5.773	0.000	0.658	0.658

## Intercepts:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
.ClrRspn (.31.)	4.513	0.052	86.495	0.000	4.513	5.750
.EffctvA (.32.)	4.429	0.054	82.644	0.000	4.429	5.584
.Feedbck (.33.)	4.307	0.057	75.837	0.000	4.307	5.159
.ClrOrgn (.34.)	4.082	0.069	59.369	0.000	4.082	3.980
.ClrPrsn (.35.)	4.299	0.060	71.957	0.000	4.299	4.872
.VlObjct (.36.)	4.539	0.039	115.407	0.000	4.539	7.650
.IncrUnd (.37.)	4.276	0.058	73.163	0.000	4.276	5.017
.IncrInt (.38.)	3.961	0.069	57.724	0.000	3.961	3.983
.MltPrsp (.39.)	4.479	0.055	81.030	0.000	4.479	5.666
.InclsvC (.40.)	4.675	0.046	102.360	0.000	4.675	6.991
.DEIntgr (.41.)	4.550	0.050	91.154	0.000	4.550	6.016

.EqtblEv (.42.)	4.595	0.042	109.486	0.000	4.595	7.449
-----------------	-------	-------	---------	-------	-------	-------

## Variances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
.ClearRspnsblts	0.233	0.027	8.728	0.000	0.233	0.379
.EffectivAnswrs	0.216	0.025	8.498	0.000	0.216	0.344
.Feedback	0.328	0.036	9.137	0.000	0.328	0.471
.ClearOrganiztn	0.439	0.049	8.924	0.000	0.439	0.418
.ClearPresenttn	0.238	0.029	8.199	0.000	0.238	0.306
.ValObjectives	0.227	0.025	9.254	0.000	0.227	0.646
.IncrUndrstndng	0.248	0.035	7.095	0.000	0.248	0.341
.IncrInterest	0.300	0.046	6.515	0.000	0.300	0.304
.MultPerspectvs	0.139	0.025	5.678	0.000	0.139	0.223
.InclusvClassrm	0.181	0.022	8.205	0.000	0.181	0.406
.DEIIntegration	0.341	0.037	9.224	0.000	0.341	0.596
.EquitableEval	0.190	0.022	8.811	0.000	0.190	0.500
TradPed	0.383	0.052	7.388	0.000	1.000	1.000
Valued	0.125	0.027	4.595	0.000	1.000	1.000
SCRPed	0.486	0.063	7.655	0.000	1.000	1.000

## Group 2 [ORG]:

## Latent Variables:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
TradPed =~						
ClrRspn	1.000				0.761	0.908
EffctvA (.p2.)	1.039	0.062	16.868	0.000	0.790	0.887
Feedbck (.p3.)	0.982	0.074	13.309	0.000	0.747	0.693
ClrOrgn (.p4.)	1.265	0.083	15.243	0.000	0.962	0.834
ClrPrsn (.p5.)	1.188	0.066	18.011	0.000	0.904	0.926
Valued =~						
VlObjct	1.000				0.283	0.454
IncrUnd (.p7.)	1.960	0.216	9.074	0.000	0.554	0.758
IncrInt (.p8.)	2.351	0.256	9.185	0.000	0.664	0.769
SCRPed =~						
MltPrsp	1.000				0.537	0.592
InclsC (.10.)	0.740	0.055	13.350	0.000	0.397	0.656
DEIntgr (.11.)	0.690	0.066	10.401	0.000	0.370	0.540
EqtblEv (.12.)	0.626	0.053	11.757	0.000	0.336	0.506

## Covariances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
TradPed ~~						
Valued	0.126	0.037	3.357	0.001	0.584	0.584
SCRPed	0.277	0.074	3.745	0.000	0.678	0.678
Valued ~~						
SCRPed	0.114	0.032	3.530	0.000	0.750	0.750

## Intercepts:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
.ClrRspn (.31.)	4.513	0.052	86.495	0.000	4.513	5.386
.EffctvA (.32.)	4.429	0.054	82.644	0.000	4.429	4.976
.Feedbck (.33.)	4.307	0.057	75.837	0.000	4.307	3.995
.ClrOrgn (.34.)	4.082	0.069	59.369	0.000	4.082	3.537
.ClrPrsn (.35.)	4.299	0.060	71.957	0.000	4.299	4.404
.VlObjct (.36.)	4.539	0.039	115.407	0.000	4.539	7.289
.IncrUnd (.37.)	4.276	0.058	73.163	0.000	4.276	5.855
.IncrInt (.38.)	3.961	0.069	57.724	0.000	3.961	4.583
.MltPrsp (.39.)	4.479	0.055	81.030	0.000	4.479	4.941
.InclsC (.40.)	4.675	0.046	102.360	0.000	4.675	7.728
.DEIntgr (.41.)	4.550	0.050	91.154	0.000	4.550	6.629
.EqtblEv (.42.)	4.595	0.042	109.486	0.000	4.595	6.919
TradPed	-0.088	0.107	-0.820	0.412	-0.116	-0.116
Valued	0.049	0.049	0.991	0.322	0.172	0.172
SCRPed	-0.040	0.097	-0.416	0.678	-0.075	-0.075

## Variances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
.ClearRspnsblts	0.124	0.029	4.209	0.000	0.124	0.176
.EffectivAnswnrs	0.168	0.037	4.530	0.000	0.168	0.212
.Feedback	0.605	0.112	5.421	0.000	0.605	0.520
.ClearOrganiztn	0.406	0.081	4.994	0.000	0.406	0.305
.ClearPresenttn	0.136	0.036	3.795	0.000	0.136	0.143
.Val0bjectives	0.308	0.057	5.385	0.000	0.308	0.794
.IncrUnderstdndng	0.227	0.058	3.917	0.000	0.227	0.425
.IncrInterest	0.306	0.081	3.791	0.000	0.306	0.409
.MultPerspectvs	0.534	0.108	4.924	0.000	0.534	0.649
.InclusvClassrm	0.208	0.046	4.558	0.000	0.208	0.569
.DEIintegrtion	0.334	0.065	5.113	0.000	0.334	0.709
.EquitableEval	0.328	0.063	5.219	0.000	0.328	0.744
TradPed	0.578	0.115	5.042	0.000	1.000	1.000
Valued	0.080	0.024	3.308	0.001	1.000	1.000
SCRPed	0.288	0.081	3.558	0.000	1.000	1.000

Let's export the data into .csv files that we can manipulate outside of the R environment.

```
strong_FitStats <- tidySEM:::table_fit(strong)
strong_paramEsts <- tidySEM:::table_results(strong, digits = 3, columns = NULL)
strong_Corrs <- tidySEM:::table_corrs(strong, digits = 3)
# to see each of the tables, remove the hashtag strong_FitStats
# strong_paramEsts strong_Corrs
```

```
write.csv(strong_FitStats, file = "strong_FitStats.csv")
write.csv(strong_paramEsts, file = "strong_paramEsts.csv")
write.csv(strong_Corrs, file = "strong_Corrs.csv")
```

with the `lavaan::anova` function we can formally compare the configural, weak, and strong tests.

```
lavaan::anova(configural, weak, strong)
```

#### Chi-Squared Difference Test

	Df	AIC	BIC	Chisq	Chisq diff	RMSEA	Df diff	Pr(>Chisq)
configural	102	5951.9	6231.7	339.43				
weak	111	5994.8	6242.3	400.29	60.862	0.207759	9	0.0000000009145
strong	120	5989.7	6205.0	413.23	12.936	0.057238	9	0.1655

configural  
 weak \*\*\*  
 strong  
 ---  
 Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Within this overall picture of poorly fitting models, the two are not statistically significantly different from each other:  $\chi^2_D(9) = 12.936, p = 0.166$ .

Below we can calculate the change CFI test from the CFI values from the weak and strong tests:  $\Delta CFI = .002$

```
# CFI change test
0.857 - 0.855
```

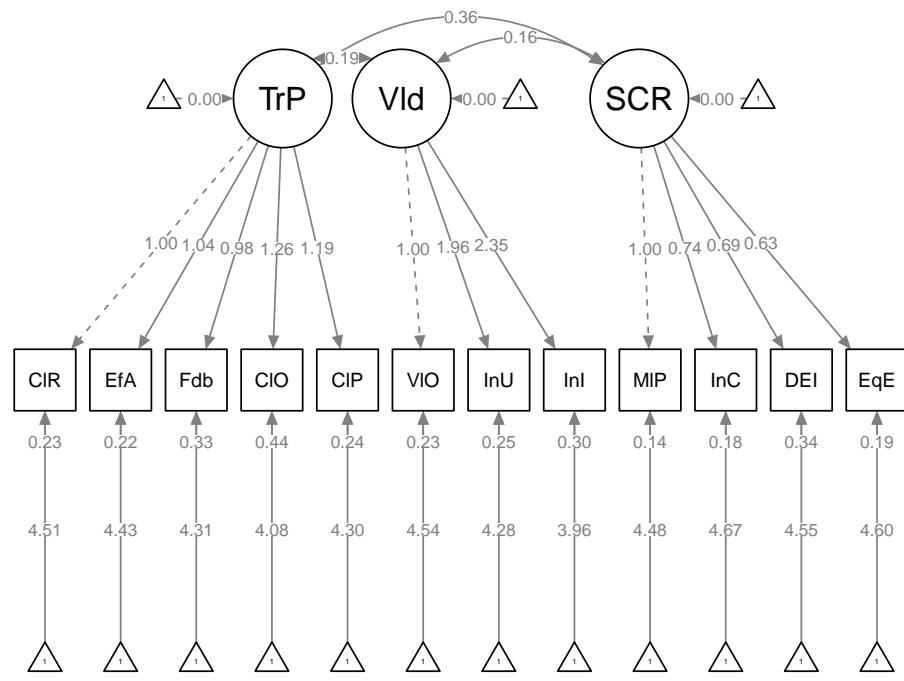
```
[1] 0.002
```

Don't be fooled by this test that falls within the  $\Delta CFI < 0.01$  criteria. It doesn't help us that our "poor fit" doesn't differ significantly across models.

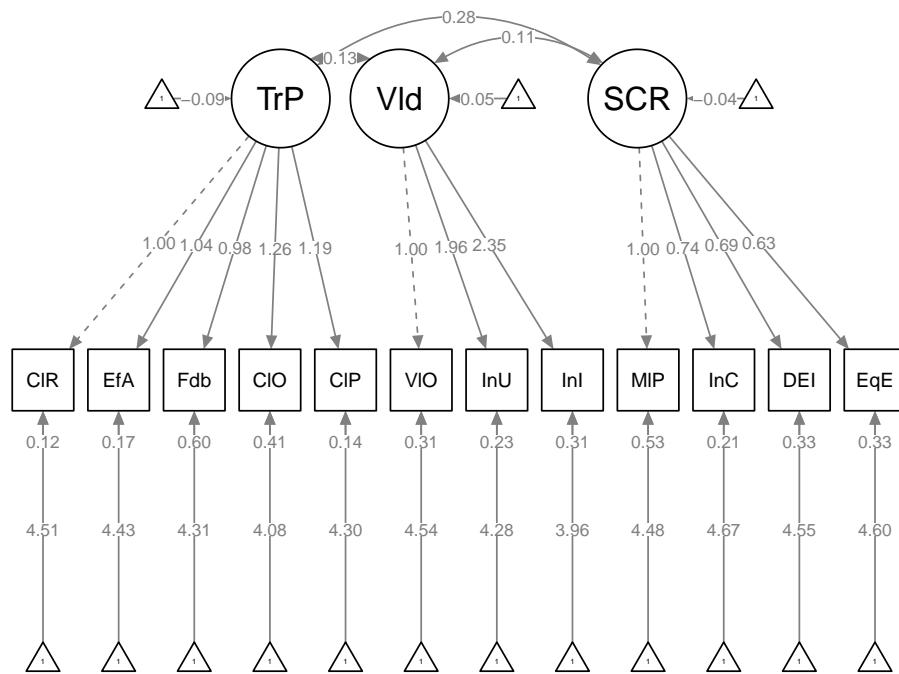
**Strong invariance model.** In the strong invariance model, configuration, factor loadings, and indicator means/intercepts were constrained to be the same for each group. Fit indices were less than ideal:  $\chi^2(120) = 413.226, p < 0.001, CFI = 0.855, SRMR = 0.103, RMSEA = 0.135(90)$ . The difference tests that evaluated model similarity between the weak and strong constraints were not statistically significant:  $\chi^2_D(9) = 12.936, p = 0.166; \Delta CFI = .002$ .

```
semPlot::semPaths(strong, layout = "tree", style = "lisrel", what = "col",
  whatLabels = "est")
```

1



2



```

# Depress ENTER a couple of times if R appears to freeze this will
# display both plots
  
```

Specify, evaluate, and interpret the CFA for strict invariance. Conduct the analysis to compare fit the strict and strong models. Write up the preliminary results.

To the weak invariance code we add “residuals” to a concatenated list in the *group.equal* command.

```

strict <- lavaan:::cfa(corrF, data = items, group = "Dept", group.equal = c("loadings",
  "intercepts", "residuals"))
lavaan:::summary(strict, fit.measures = TRUE, standardized = TRUE, )
  
```

*lavaan* 0.6.17 ended normally after 82 iterations

Estimator

ML

Optimization method	NLMINB	
Number of model parameters	81	
Number of equality constraints	33	
Number of observations per group:	Used	Total
CPY	202	232
ORG	65	78

## Model Test User Model:

Test statistic	459.401
Degrees of freedom	132
P-value (Chi-square)	0.000
Test statistic for each group:	
CPY	240.235
ORG	219.166

## Model Test Baseline Model:

Test statistic	2156.987
Degrees of freedom	132
P-value	0.000

## User Model versus Baseline Model:

Comparative Fit Index (CFI)	0.838
Tucker-Lewis Index (TLI)	0.838

## Loglikelihood and Information Criteria:

Loglikelihood user model (H0)	-2957.953
Loglikelihood unrestricted model (H1)	-2728.252
Akaike (AIC)	6011.906
Bayesian (BIC)	6184.094
Sample-size adjusted Bayesian (SABIC)	6031.906

## Root Mean Square Error of Approximation:

RMSEA	0.136
90 Percent confidence interval - lower	0.123
90 Percent confidence interval - upper	0.150
P-value H_0: RMSEA <= 0.050	0.000
P-value H_0: RMSEA >= 0.080	1.000

## Standardized Root Mean Square Residual:

SRMR	0.116
------	-------

Parameter Estimates:

Standard errors	Standard
Information	Expected
Information saturated (h1) model	Structured

Group 1 [CPY]:

Latent Variables:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
TradPed ==						
ClrRspn	1.000				0.610	0.800
EffctvA (.p2.)	1.040	0.066	15.724	0.000	0.634	0.810
Feedbck (.p3.)	1.010	0.077	13.173	0.000	0.616	0.706
ClrOrgn (.p4.)	1.300	0.088	14.795	0.000	0.792	0.773
ClrPrsn (.p5.)	1.231	0.073	16.748	0.000	0.750	0.848
Valued ==						
VlObjct	1.000				0.347	0.571
IncrUnd (.p7.)	2.004	0.226	8.856	0.000	0.696	0.819
IncrInt (.p8.)	2.390	0.268	8.926	0.000	0.830	0.834
SCRPed ==						
MltPrsp	1.000				0.707	0.843
InclsuC (.10.)	0.622	0.053	11.698	0.000	0.440	0.678
DEIntgr (.11.)	0.591	0.063	9.426	0.000	0.418	0.566
EqtblEv (.12.)	0.644	0.052	12.496	0.000	0.455	0.716

Covariances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
TradPed ~~						
Valued	0.185	0.029	6.388	0.000	0.873	0.873
SCRPed	0.374	0.048	7.805	0.000	0.869	0.869
Valued ~~						
SCRPed	0.166	0.029	5.688	0.000	0.675	0.675

Intercepts:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
.ClrRspn (.31.)	4.514	0.052	87.278	0.000	4.514	5.924
.EffctvA (.32.)	4.429	0.053	83.153	0.000	4.429	5.656
.Feedbck (.33.)	4.297	0.058	74.126	0.000	4.297	4.927
.ClrOrgn (.34.)	4.087	0.069	59.115	0.000	4.087	3.988
.ClrPrsn (.35.)	4.298	0.061	70.788	0.000	4.298	4.860
.VlObjct (.36.)	4.535	0.040	114.691	0.000	4.535	7.453
.IncrUnd (.37.)	4.277	0.058	73.215	0.000	4.277	5.032
.IncrInt (.38.)	3.961	0.069	57.702	0.000	3.961	3.981
.MltPrsp (.39.)	4.470	0.058	77.085	0.000	4.470	5.329
.InclsuC (.40.)	4.685	0.043	108.763	0.000	4.685	7.229

.DEIntgr (.41.)	4.557	0.048	95.088	0.000	4.557	6.173
.EqtblEv (.42.)	4.600	0.043	107.915	0.000	4.600	7.233

## Variances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
.ClrRspn (.16.)	0.209	0.021	9.784	0.000	0.209	0.360
.EffctvA (.17.)	0.211	0.022	9.657	0.000	0.211	0.344
.Feedbck (.18.)	0.381	0.036	10.573	0.000	0.381	0.501
.ClrOrgn (.19.)	0.422	0.042	10.079	0.000	0.422	0.402
.ClrPrsn (.20.)	0.219	0.024	8.978	0.000	0.219	0.280
.VlObjct (.21.)	0.250	0.023	10.729	0.000	0.250	0.674
.IncrUnd (.22.)	0.238	0.030	7.866	0.000	0.238	0.329
.IncrInt (.23.)	0.301	0.041	7.406	0.000	0.301	0.304
.MltPrsp (.24.)	0.204	0.028	7.179	0.000	0.204	0.290
.InclsVC (.25.)	0.227	0.023	10.077	0.000	0.227	0.540
.DEIntgr (.26.)	0.370	0.034	10.748	0.000	0.370	0.679
.EqtblEv (.27.)	0.197	0.020	9.721	0.000	0.197	0.488
TradPed	0.372	0.051	7.259	0.000	1.000	1.000
Valued	0.121	0.027	4.447	0.000	1.000	1.000
SCRPed	0.499	0.069	7.249	0.000	1.000	1.000

## Group 2 [ORG]:

## Latent Variables:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
TradPed ==						
ClrRspn	1.000				0.738	0.850
EffctvA (.p2.)	1.040	0.066	15.724	0.000	0.767	0.858
Feedbck (.p3.)	1.010	0.077	13.173	0.000	0.745	0.770
ClrOrgn (.p4.)	1.300	0.088	14.795	0.000	0.959	0.828
ClrPrsn (.p5.)	1.231	0.073	16.748	0.000	0.908	0.889
Valued ==						
VlObjct	1.000				0.278	0.487
IncrUnd (.p7.)	2.004	0.226	8.856	0.000	0.558	0.753
IncrInt (.p8.)	2.390	0.268	8.926	0.000	0.665	0.772
SCRPed ==						
MltPrsp	1.000				0.724	0.849
InclsVC (.10.)	0.622	0.053	11.698	0.000	0.451	0.687
DEIntgr (.11.)	0.591	0.063	9.426	0.000	0.428	0.576
EqtblEv (.12.)	0.644	0.052	12.496	0.000	0.466	0.724

## Covariances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
TradPed ~~						
Valued	0.127	0.037	3.450	0.001	0.616	0.616
SCRPed	0.414	0.094	4.425	0.000	0.775	0.775
Valued ~~						

SCRPed	0.153	0.040	3.843	0.000	0.760	0.760
<b>Intercepts:</b>						
.ClrRspn (.31.)	4.514	0.052	87.278	0.000	4.514	5.200
.EffctvA (.32.)	4.429	0.053	83.153	0.000	4.429	4.952
.Feedbck (.33.)	4.297	0.058	74.126	0.000	4.297	4.439
.ClrOrgn (.34.)	4.087	0.069	59.115	0.000	4.087	3.527
.ClrPrsn (.35.)	4.298	0.061	70.788	0.000	4.298	4.207
.V1Objct (.36.)	4.535	0.040	114.691	0.000	4.535	7.931
.IncrUnd (.37.)	4.277	0.058	73.215	0.000	4.277	5.773
.IncrInt (.38.)	3.961	0.069	57.702	0.000	3.961	4.595
.MltPrsp (.39.)	4.470	0.058	77.085	0.000	4.470	5.236
.InclsC (.40.)	4.685	0.043	108.763	0.000	4.685	7.146
.DEIntgr (.41.)	4.557	0.048	95.088	0.000	4.557	6.124
.EqtblEv (.42.)	4.600	0.043	107.915	0.000	4.600	7.141
TradPed	-0.096	0.105	-0.912	0.362	-0.130	-0.130
Valued	0.047	0.048	0.978	0.328	0.169	0.169
SCRPed	-0.099	0.113	-0.878	0.380	-0.136	-0.136
<b>Variances:</b>						
.ClrRspn (.16.)	0.209	0.021	9.784	0.000	0.209	0.277
.EffctvA (.17.)	0.211	0.022	9.657	0.000	0.211	0.264
.Feedbck (.18.)	0.381	0.036	10.573	0.000	0.381	0.407
.ClrOrgn (.19.)	0.422	0.042	10.079	0.000	0.422	0.315
.ClrPrsn (.20.)	0.219	0.024	8.978	0.000	0.219	0.210
.V1Objct (.21.)	0.250	0.023	10.729	0.000	0.250	0.763
.IncrUnd (.22.)	0.238	0.030	7.866	0.000	0.238	0.433
.IncrInt (.23.)	0.301	0.041	7.406	0.000	0.301	0.405
.MltPrsp (.24.)	0.204	0.028	7.179	0.000	0.204	0.280
.InclsC (.25.)	0.227	0.023	10.077	0.000	0.227	0.528
.DEIntgr (.26.)	0.370	0.034	10.748	0.000	0.370	0.669
.EqtblEv (.27.)	0.197	0.020	9.721	0.000	0.197	0.475
TradPed	0.544	0.112	4.855	0.000	1.000	1.000
Valued	0.077	0.023	3.333	0.001	1.000	1.000
SCRPed	0.525	0.116	4.533	0.000	1.000	1.000

Let's export the data into .csv files that we can manipulate outside of the R environment.

```
strict_FitStats <- tidySEM::table_fit(strict)
strict_paramEsts <- tidySEM::table_results(strict, digits = 3, columns = NULL)
strict_Corrs <- tidySEM::table_cors(strict, digits = 3)
# to see each of the tables, remove the hashtag strict_FitStats
# strict_paramEsts strict_Corrs
```

```
write.csv(strict_FitStats, file = "strict_FitStats.csv")
write.csv(strict_paramEsts, file = "strict_paramEsts.csv")
write.csv(strict_Corrs, file = "strict_Corrs.csv")
```

with the *lavaan::anova* function we can formally compare the configural, weak, strong, and strict tests.

```
lavaan::anova(configural, weak, strong, strict)
```

#### Chi-Squared Difference Test

	Df	AIC	BIC	Chisq	Chisq diff	RMSEA	Df diff	Pr(>Chisq)
configural	102	5951.9	6231.7	339.43				
weak	111	5994.8	6242.3	400.29	60.862	0.207759	9	0.0000000009145
strong	120	5989.7	6205.0	413.23	12.936	0.057238	9	0.1655
strict	132	6011.9	6184.1	459.40	46.175	0.146057	12	0.0000064729187

configural  
 weak \*\*\*  
 strong  
 strict \*\*\*  
 ---  
 Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

The Chi-square difference test indicates statistically significant differences between the strict and strong models:  $\chi^2_D(12) = 46.175, p < 0.001$ .

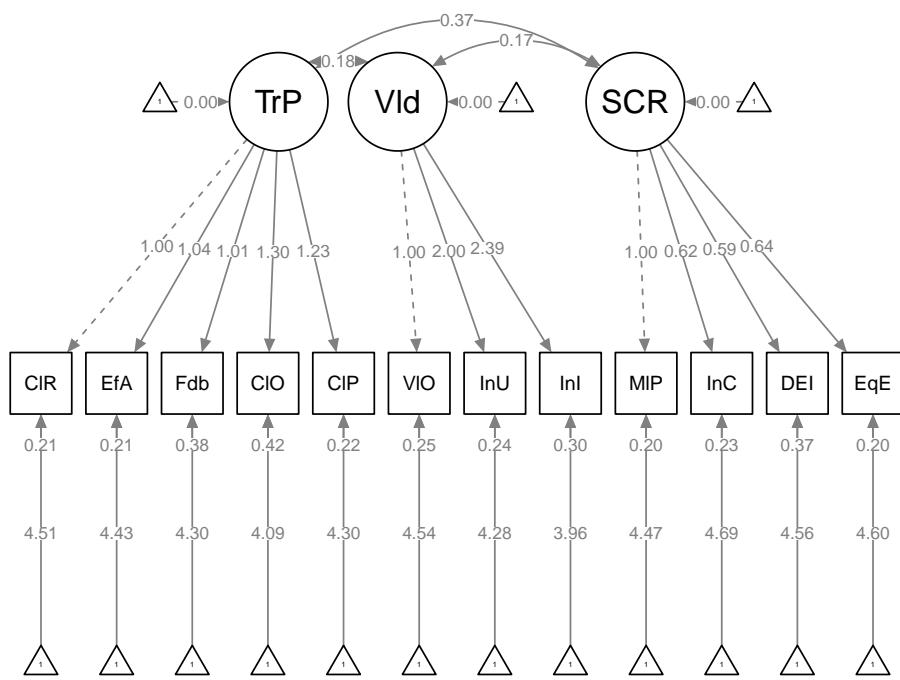
Below we hand calculate the change CFI test. This difference between the CFI tests from the strong and strict models is 0.017 and slightly exceeds the  $\Delta CFI < 0.01$ . criteria.

```
0.855 - 0.838 #change CFI test
```

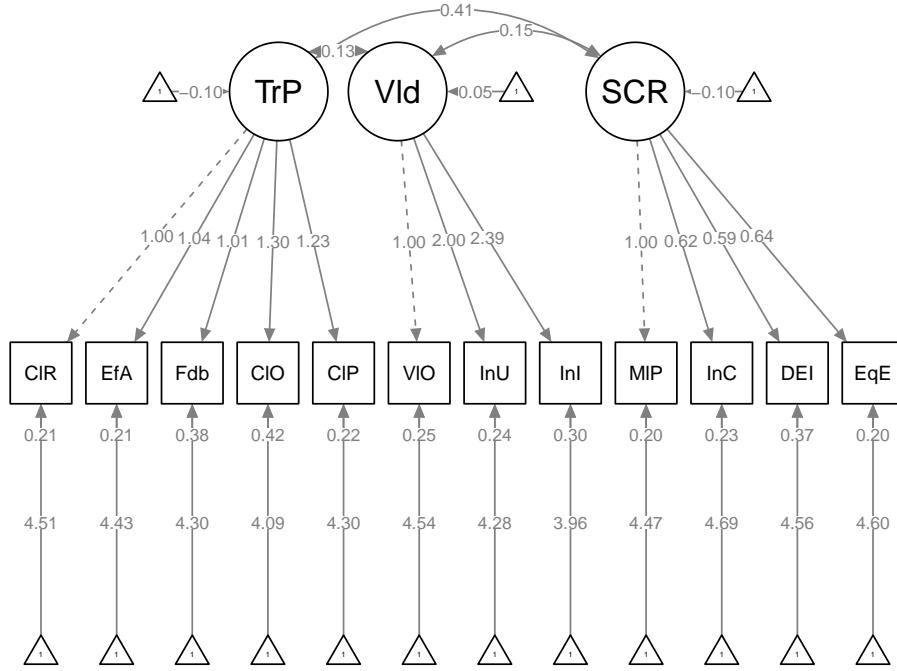
```
[1] 0.017
```

**Strict invariance model.** In the strict invariance model, configuration, factor loadings, indicator means/intercepts, and residuals were constrained to be the same for each group. Fit indices were less than ideal:  $\chi^2(132) = 459.401, p < .001, CFI = 0.838, SRMS = 0.116, RMSEA = 0.1360(90)$ . Factorial noninvariance was already suggested in the restriction from weak to strong, this continues to be true:  $\chi^2_D(12) = 46.175, p < 0.001; \Delta CFI = 0.017$ . For CPY, standardized pattern coefficients ranged between 0.71 and 0.85 on the TradPed factor, between 0.57 and 0.83 on the Valued factor, and between 0.68 and 0.84 on the SCRPed factor. For ORG, standardized pattern coefficients ranged between 0.77 and 0.89 for the TradPed factor, between 0.49 and 0.77 on the Valued factor, and between 0.58 and 0.85 on the SCRPed factor.

```
semPlot::semPaths(strict, layout = "tree", style = "lisrel", what = "col",
  whatLabels = "est")
```

**1**

2



```
# If R appears to stall depress the ENTER key twice to see both plots
```

Create an APA style results section. Do not report any invariance tests past the one that failed. Include a table(s) and figure(s).

To test if the factor structures of the course evaluations were stable across department (CPY and ORG), we used measurement invariance analyses. First, we constructed CFA models with the lavaan (v. 0.6-17) package in R and created two groups representing CPY and ORG. Within lavaan we successively constrained parameters representing the configural, weak (loadings), strong (intercepts), and strict (residuals) structures (Hirschfeld & von Brachel, 2014; Kline, 2016). A poor fit in any of these models suggests that the aspect being constrained does not operate consistently for the different groups. The degree of invariance was determined jointly when  $\chi^2 p > .05$  and a  $\Delta CFI < .01$ .

The configural model constrains only the relative configuration of variables in the model to be the same in both groups. In other words, no factor loadings or indicator means are

constrained to be the same across groups, but the organization of indicators is the same for both groups. Next, in weak factorial invariance, the configuration of variables and all factor loadings are constrained to be the same for each group. Poor fit here suggests that the factor loadings vary in size between the two groups. In strong factorial invariance, both the configuration, factor loadings, and the indicator means are constrained to be the same for each group. A reduction in fit here, but not in the previous steps, suggests that indicators have different means in both groups, which might be expected when comparing two groups of people, such as in a t-test. Therefore, poor fit in only this model does not necessarily indicate the factor structure operates differently for different groups. Finally, strict factorial invariance requires strong invariance and equality in error variances and covariances across groups. This means that the indicators measure the same factors in each group with the same degree of precision.

We selected fit criteria for their capacity to assess different aspects of the statistical analysis. As is common among SEM researchers, we reported the chi-square goodness of fit ( $\chi^2$ ). This evaluates the discrepancy between the unrestricted sample matrix and the restricted covariance matrix. Although the associated p value indicates adequate fit when the value is non-significant, it is widely recognized that large sample size can result in a statistically significant *p* value (Byrne, 2016). The comparative fit index (CFI) is an incremental index, comparing the hypothesized model with the baseline model, and should be at least .90 and perhaps higher than .95 (Hu & Bentler, 1999). The root mean square error of approximation (RMSEA) takes into account the error of approximation in the population and expresses it per degree of freedom. As such, the fit indicator considers the complexity of the model. Ideal values are equal to or less than .05, values less than .08 represent reasonable fit, and values between .08 and .10 represent mediocre fit. The standardized root mean residual (SRMR) is a standardized measure of the mean absolute covariance residual – the overall difference between the observed and predicted correlations. Values greater than .10 may indicate poor fit and inspection of residuals is then advised. Researchers have advised caution when using these criteria as strict cutoffs, and other factors such as sample size and model complexity should be considered (Kline, 2016).

Course evaluation items each were loaded on their respective correlated factors. The configural model, which constrained only the relative configuration of variables in the model to be the same in both groups had less than adequate fit to the data:  $\chi^2(102) = 339.43, p < 0.001, CFI = 0.883, SRMR = 0.073, RMSEA = 0.132, 90$ . Even though some of the fit criteria were less than adequate, we cautiously proceeded to the next stage of invariance testing. The weak invariance model constrained the configuration of variables and all factor loadings to be constant across groups. Unfortunately, all indicators suggested poor fit:  $\chi^2(111) = 400.29, p < 0.001, CFI = 0.857, SRMR = 0.101, RMSEA = 0.140(90)$ . Further, noninvariance of the factor loadings was supported by the significant difference tests that assessed model similarity:  $\chi^2_D(9) = 60.86, p < .001; \Delta CFI = .026$ . Because we found noninvariance at the weak level, we did not attempt to model strong nor strict invariance.



# Chapter 13

## Hybrid Models

### [Screencasted Lecture Link](#)

Psychometrics courses usually focus on evaluating the psychometric properties of an instrument and outlining the steps/procedures in instrument development. I believe it is also important to understand how to incorporate those psychometrically credible measures in research designs and particularly, SEM. Thus, the purpose of this lecture is to walk through a real dataset from missing data analysis, to analyzing and managing missing data, to assessing the distributional characteristics of the data, to creating a measurement model, and finally recrafting it as a structural mode.

Our goal is:

- Starting with a raw dataset:
  - examine missing patterns mechanism and managing missing data,
  - evaluate the distributional characteristics of the data, and
  - conducting preliminary analyses.
- Specify our measurement model (all the measures, allowing them all to correlate).
- Briefly examine *identification* in the measurement and structural portions of the model.
- Specify and evaluate our a priorily defined structural model.
- Write it up!

At the outset, please know that this lesson (a) skips a few steps and (b) introduces (entirely too quickly) some steps that will be new to those who are not exposed to structural equation modeling (SEM). While somewhat regrettable, this is intentional. In the program where I have taught, we teach psychometrics before the multivariate class and this is the last lesson in psychometrics. One intent is to provide an advanced cognitive organizer for what is yet to come in the [SEM lessons](#) in multivariate modeling. In-so-doing, I'm hoping to show how the entire psychometric process is critical to our final, tested, models.

## 13.1 Navigating this Lesson

There is about 1 hour and 45 minutes of lecture. If you work through the materials with me it would be plan for an additional two hours.

While the majority of R objects and data you will need are created within the R script that sources the chapter, occasionally there are some that cannot be created from within the R framework. Additionally, sometimes links fail. All original materials are provided at the [Github site](#) that hosts the book. More detailed guidelines for ways to access all these materials are provided in the OER's [introduction](#).

### 13.1.1 Learning Objectives

Focusing on this week's materials, make sure you can:

- Identify steps in preparing data for structural equation modeling.
- Differentiate a measurement model from a structural model and know which one *will* have better fit.
- List the general steps in evaluating a hybrid model.
- Specify and interpret the results of a measurement model.
- Specify and interpret the results of a structural model.

### 13.1.2 Planning for Practice

In each of these lessons I provide suggestions for practice that allow you to select one or more problems that are graded in difficulty. The least complex is to import the latest [Rate-a-Recent-Course: A ReCentering Psych Stats Exercise](#) data from Qualtrics and rework the problem as written in the lesson. For an increased challenge, swap out one or more variables/scales. And for a maximal challenge, try something entirely new with data (simulated or real) that you have permission to use.

Regardless of your choic(es) please work through the following:

- Structure up your dataframe.
- Analyze and manage missing data.
- Evaluate the assumptions for multivariate analysis.
- Conduct appropriate preliminary analyses.
- Specify and evaluate a measurement model.
- Prepare an APA style results section with table(s) and figure(s).
- Explain it to somebody.

### 13.1.3 Readings & Resources

In preparing this chapter, I drew heavily from the following resource(s). Other resources are cited (when possible, linked) in the text with complete citations in the reference list.

Kline, R. (2016). Principles and practice of structural equation modeling (Fourth ed., Methodology in the social sciences). New York: The Guilford Press.

- Chapter 4, Data Preparation and Psychometrics Review
- Chapter 10, Specification and Identification of Structural Regression Models
- Chapter 11, Estimation and Local Fit Testing
- Chapter 13, Analysis of CFA Models

Little, T. D., Cunningham, W. A., Shahar, G., & Widaman, K. F. (2002). To parcel or not to parcel: Exploring the question, weighing the merits. *Structural Equation Modeling*, 9(2), 151–173. [https://doi.org/10.1207/S15328007SEM0902\\_1](https://doi.org/10.1207/S15328007SEM0902_1)

Little, T. D., Rhemtulla, M., Gibson, K., & Schoemann, A. M. (2013). Why the items versus parcels controversy needn't be one. *Psychological Methods*, 18(3), 285–300. <https://doi.org/10.1037/a0033266>

Rosseel, Y. (2019). The *lavaan* tutorial. Belgium: Department of Data Analysis, Ghent University. <http://lavaan.ugent.be/tutorial/tutorial.pdf>

### 13.1.4 Packages

The packages used in this lesson are embedded in this code. When the hashtags are removed, the script below will (a) check to see if the following packages are installed on your computer and, if not (b) install them.

```
# will install the package if not already installed
# if(!require(psych)){install.packages('psych')}
# if(!require(lavaan)){install.packages('lavaan')}
# if(!require(semPlot)){install.packages('semPlot')}
# if(!require(psych)){install.packages('psych')}
# if(!require(semTable)){install.packages('semTable')}
# if(!require(semTools)){install.packages('semTools')}
# if(!require(sempretools)){install.packages('sempretools')}
```

## 13.2 Introducing the Statistic

The model we are testing in this lesson is *hybrid* that is, it contains both CFA and the structural paths. Although there are several detour along the way, the analytic approach has two large stages:

- Testing the *measurement model* which includes of each of the factors and its indicators with covariances between each of the latent variables.
  - The measurement model will have the best fit because all of the structural paths are saturated (i.e., there is a covariance between them).
  - If the fit of the measurement model is below the threshold, a technique like parceling could be helpful, but we will not get there today. More information on that is found in [ReCentering Psych Stats: Multivariate Modeling](#).
- Test the *structural model*. This means we delete the covariances and respecify the model to include the directional paths and covariances we hypothesized.

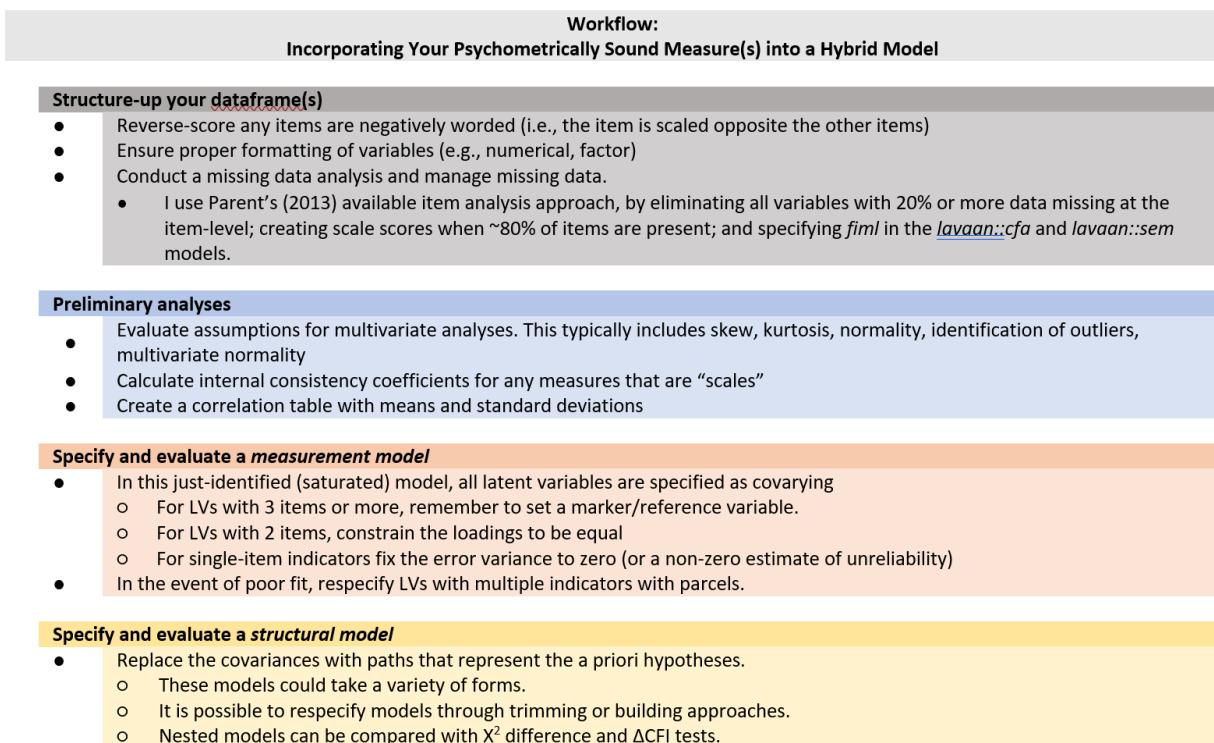


Figure 13.1: Image of a flowchart and decision-tree for evaluating hybrid SEM models

The steps in working the STATISTIC generally include,

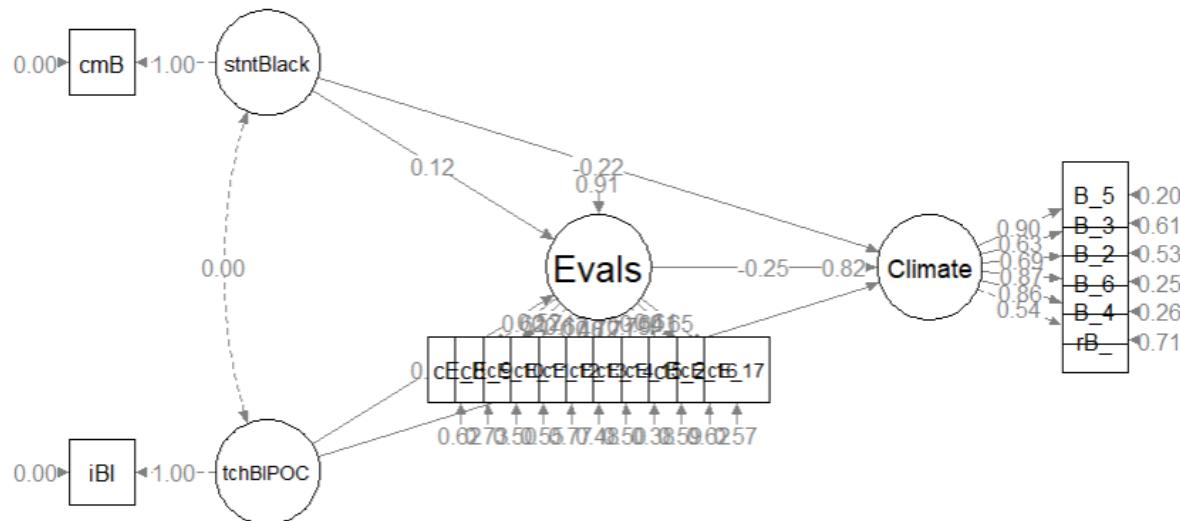
- Structuring -up your dataframe(s)
  - Reverse-score any items are negatively worded (i.e., the item is scaled opposite the other items)
  - Ensure proper formatting of variables (e.g., numerical, factor)
  - Conduct a missing data analysis and manage missing data.
- Preliminary analyses
  - Evaluate assumptions for multivariate analyses.
  - Calculate internal consistency coefficients for any measures that are “scales”
  - Create a correlation table with means and standard deviations
- Specify and evaluate a measurement model. In this just-identified (saturated) model, all latent variables are specified as covarying
  - In the event of poor fit, respecify LVs with multiple indicators with [parcels](#).
- Specify and evaluate a structural model by replacing the covariances with paths that represent the a priori hypotheses.
  - These models could take a variety of forms.
  - It is possible to respecify models through trimming or building approaches.
  - Nested models can be compared with  $\chi^2_D$  and  $\Delta CFI$  tests.

### 13.3 Research Vignette

The research vignette comes from the open survey titled, [Rate-a-Recent-Course: A ReCentering Psych Stats Exercise](#). A series of lessons devoted to preparing data for analysis provide information about the specific variables and link to the codebook. They are available in the [Multivariate Modeling](#) volume.

If you are 18 years or older and have recently taken any type of course (e.g., college, graduate, continuing education), please consider taking the survey. Each time someone responds to the survey, it will allow users to follow along with a slightly different database.

The Rate-a-Recent-Course has a number of scales and variables. We will use four variables/scales to specify a parallel mediation predicting the perceptions of campus climate for students who are Black from the percent of classmates who are Black, the proportion of instructional staff who are BIPOC (Black, Indigenous, and Persons of Color), and course evaluation ratings that assess the degree to which the pedagogy is socially responsive.



Variables in the model:

- Perceived Campus Climate for Black Students includes 6 items, one of which was reverse scored. This scale was adapted from Szymanski et al.'s [2020] Campus Climate for LGBTQ students. It has not been evaluated for use with other groups. The Szymanski et al. analysis suggested that it could be used as a total scale score, or divided into three items each that assess the college's response and experienced stigma. Items were assessed on a 7-point scale ranging from *strongly disagree* to *strongly agree* with higher scores indicating a more hostile campus climate. Example items from our revised scale include:
  - College response: “My college is unresponsive to the needs of Black students.”
  - Stigma: “Anti-Black racism is visible in my college.”
- Course evaluation items assessed the the degree to which the pedagogy of course reviewed by the respondent was socially and culturally responsive. In developing this survey, we chose items after reviewing evaluation items from several institutions of higher education

and reviewing evaluative tools for open education resources. Eleven items were assessed on a 5-point scale ranging from *strongly disagree* to *strongly agree* with higher scores indicating a more positive evaluation of the course. Example items include:

- “Course content included materials authored by members of communities that are often marginalized (e.g., BIPOC, LGBTQ+, emerging economies).”
- “A land acknowledgement was made (i.e., formal statement naming the indigenous people who originally inhabited the land).”
- “Course materials (e.g., textbooks, articles, videos/podcasts) were free/no-cost to the students.”
- Percent of Black classmates was a single item that asked respondents to estimate the proportion of students in various racial categories.
- Percent of BIPOC instructional staff, similarly, asked respondents to identify the racial category of each member of their instructional staff.

Our design has notable limitations. Briefly, (a) owing to the open source aspect of the data we do not ask about the demographic characteristics of the respondent; (b) the items that ask respondents to *guess* the identities of the instructional staff and to place them in broad categories, (c) we do not provide options for “write-in” responses. We made these decisions after extensive conversation with stakeholders. The primary reason for these decisions was to prevent potential harm (a) to respondents who could be identified if/when the revealed private information in this open-source survey, and (b) trolls who would write inappropriate or harmful comments.

*I would like to assess the model by having the instructional staff variable to be the %Black instructional staff. At the time that this lecture is being prepared, there is not sufficient Black representation in this variable to model this.*

## 13.4 Importing and Preparing the Data

This section of the lesson describes how to import data directly from Qualtrics and do some formatting. Because the survey remains open, if you import the data from Qualtrics you will get different results than are in the lesson. If you want to get the same results as me, [download the dataset from the GitHub](#), save it in the same place as your working .rmd file, and skip to “START HERE”. For practice, you might consider downloading the data directly from Qualtrics for updated results.

Three chapters in the [Multivariate Modeling](#) volume of ReCentering Psych Stats provide greater detail about the process of importing and preparing data from the [Rate-a-Recent-Course: A Re-Centering Psych Stats Exercise](#) survey.

We start with an intRavenous import directly from Qualtrics. This is a two-step process.

- Establishing a connection to the Qualtrics account by supplying the base URL and API credentials.
  - Be very careful with these, they provide access to everything in your Qualtrics account. This Qualtrics account has only this survey and nothing else.
  - If the API token becomes operational, please let me know. Qualtrics security may have a protocol to replace/disable them.

- Naming the survey (via its identification number) and importing the results.

```
# The hashtags line of code makes the connection to the
# institution's Qualtrics account and the individual Qualtrics
# account within that institutional brand. Once that connection is
# made, hashtag it out to avoid glitches. If you are changing from
# one account to another you will likely need to restart R.
# qualtrics::qualtrics_api_credentials(api_key =
# 'mUgPMYkiWpMFkwHale1QE5HNmh5LRUaA8d9PDg', base_url =
# 'spupsych.az1.qualtrics.com', overwrite = TRUE, install = TRUE)
# surveys <- qualtrics::all_surveys() QTRX_df
# <- qualtrics::fetch_survey(surveyID = 'SV_b2cClqAllGQ6nLU',
# time_zone = NULL, verbose = FALSE, label=TRUE, force_request =
# TRUE, import_id = FALSE) convert=FALSE,
```

In the next set of code, I quickly prepare the data that I will use for the hybrid SEM. In the next set of script we:

- Delete “previews” (those “tester” surveys taken prior to the official launch).
- Rename a few variables to make them easier to manipulate.
  - Most variable naming was completed inside the Qualtrics survey, prior to importing, but some variables were impossible to rename and we did not anticipate all of our needs.
- Create an ID number for each case and moving it to the front of the dataframe.
- Create a df that includes only the variables needed to specify the hybrid model.

```
# eliminating previews QTRX_df <- dplyr::filter (QTRX_df,
# DistributionChannel != 'preview')

# renaming variables that start with numbers QTRX_df <-
# dplyr::rename(QTRX_df, iRace1 = '1_iRace', iRace2 = '2_iRace',
# iRace3 = '3_iRace', iRace4 = '4_iRace', iRace5 = '5_iRace', iRace6
# = '6_iRace', iRace7 = '7_iRace', iRace8 = '8_iRace', iRace9 =
# '9_iRace', iRace10 = '10_iRace')

# renaming variables about classmates race/ethnicity QTRX_df <-
# dplyr::rename(QTRX_df, cmBiMulti = Race_10, cmBlack = Race_1,
# cmNBPoC = Race_7, cmWhite = Race_8, cmUnsure = Race_2)

library(tidyverse) #opening this package to be able to use pipes
# creating ID variable and moving it to the front QTRX_df <- QTRX_df
# %>% dplyr::mutate(ID = row_number()) QTRX_df <-
# QTRX_df%>%dplyr::select(ID, everything())

# downsizing df to have just variables of interest Model_df <-(select
# (QTRX_df, ID, iRace1, iRace2, iRace3, iRace4, iRace5, iRace6,
```

```
# iRace7, iRace8, iRace9, iRace10, cmBiMulti, cmBlack, cmNBPoC,
# cmWhite, cmUnsure, Blst_1:Blst_6, cEval_8, cEval_9, cEval_10,
# cEval_11, cEval_12, cEval_13, cEval_14, cEval_15, cEval_20,
# cEval_16, cEval_17))
```

The optional script below will let you save the imported data to your computing environment as either a .csv file (think “Excel lite”) or .rds object (preserves any formatting you might do). If you save the .csv file and bring it back in, you will lose any formatting (e.g., ordered factors will be interpreted as character variables).

```
# write the simulated data as a .csv write.table(Model_df,
# file='Model_df.csv', sep=',', col.names=TRUE, row.names=FALSE)
# bring back the simulated dat from a .csv file Model_df <- read.csv
# ('Model_df.csv', header = TRUE)
```

An .rds file preserves all formatting to variables prior to the export and re-import. For the purpose of this chapter, you don’t need to do either. That is, you can re-simulate the data each time you work the problem.

**START HERE** to upload the [data you downloaded from the GitHub]([https://github.com/lhbikos/ReC\\_Psychometrics/blob/main/Model\\_df211010.rds](https://github.com/lhbikos/ReC_Psychometrics/blob/main/Model_df211010.rds)). This will produce the same results in this lesson

```
# to save the df as an .rds (think 'R object') file on your computer;
# it should save in the same file as the .rmd file you are working
# with saveRDS(Model_df, 'Model_df.rds') bring back the simulated dat
# from an .rds file
Model_df <- readRDS("Model_df211010.rds") #For this lesson, I saved and imported this set of
# Model_df <- readRDS('Model_df.rds')
```

As a multicategorical variable, race/ethnicity frequently takes some thought and manipulation. I would have liked to have evaluated instructor race as the proportion of the instructional staff who is Black. At this time, there is so little variability in the instructional staff variable that we are using proportion of instructional staff who is BIPOC.

Given that classes may be teamtaught (and/or include teaching assistants) in the survey, respondents indicated how many instructional staff taught their class. For each, the respondent indicated the race/ethnicity of the instructor. It was possible to list up to 10 instructors per class. We need to get these 10 responses summarized as one variable representing the proportion of instructional faculty (per respondent/class) who were BIPOC. The code below:

- Transforms each race identification variable into a factor.
- Calculates the proportion of BIPOC instructional faculty for each respondent’s class.

```
# str(Model_df$iRace1)

Model_df$tRace1 = factor(Model_df$iRace1, levels = c(0, 1, 2, 3, 4), labels = c("Black",
```

```

  "nBpoc", "BiMulti", "White", "NotNotice"))
Model_df$tRace2 = factor(Model_df$iRace2, levels = c(0, 1, 2, 3, 4), labels = c("Black",
  "nBpoc", "BiMulti", "White", "NotNotice"))
Model_df$tRace3 = factor(Model_df$iRace3, levels = c(0, 1, 2, 3, 4), labels = c("Black",
  "nBpoc", "BiMulti", "White", "NotNotice"))
Model_df$tRace4 = factor(Model_df$iRace4, levels = c(0, 1, 2, 3, 4), labels = c("Black",
  "nBpoc", "BiMulti", "White", "NotNotice"))
Model_df$tRace5 = factor(Model_df$iRace5, levels = c(0, 1, 2, 3, 4), labels = c("Black",
  "nBpoc", "BiMulti", "White", "NotNotice"))
Model_df$tRace6 = factor(Model_df$iRace6, levels = c(0, 1, 2, 3, 4), labels = c("Black",
  "nBpoc", "BiMulti", "White", "NotNotice"))
Model_df$tRace7 = factor(Model_df$iRace7, levels = c(0, 1, 2, 3, 4), labels = c("Black",
  "nBpoc", "BiMulti", "White", "NotNotice"))
Model_df$tRace8 = factor(Model_df$iRace8, levels = c(0, 1, 2, 3, 4), labels = c("Black",
  "nBpoc", "BiMulti", "White", "NotNotice"))
Model_df$tRace9 = factor(Model_df$iRace9, levels = c(0, 1, 2, 3, 4), labels = c("Black",
  "nBpoc", "BiMulti", "White", "NotNotice"))
Model_df$tRace10 = factor(Model_df$iRace10, levels = c(0, 1, 2, 3, 4),
  labels = c("Black", "nBpoc", "BiMulti", "White", "NotNotice"))
# checking to see that they are factors glimpse(Model_df)

# counting non-White instructional staff by creating the variable
# 'count.BIPOC' by summing across tRace1 thru tRace10 and assigning a
# count of '1' each time the factor value was Black, nBpoc, or
# BiMulti
Model_df$count.BIPOC <- apply(Model_df[c("tRace1", "tRace2", "tRace3",
  "tRace4", "tRace5", "tRace6", "tRace7", "tRace8", "tRace9", "tRace10")],
  1, function(x) sum(x %in% c("Black", "nBpoc", "BiMulti")))

# created a variable that counted the number of non-missing values
# across the tRace1 thru tRace10 vars
Model_df$count.nMiss <- apply(Model_df[c("tRace1", "tRace2", "tRace3",
  "tRace4", "tRace5", "tRace6", "tRace7", "tRace8", "tRace9", "tRace10")],
  1, function(x) sum(!is.na(x)))

# calculate proportion of BIPOC instructional faculty for each case
Model_df$iBIPOC_pr = Model_df$count.BIPOC/Model_df$count.nMiss

```

The scale assessing perceptions of campus climate for Black students had six items. One was worded in the opposite direction of the rest, therefore we must reverse-score it. Following the reverse-coding, I once again trimmed the dataframe so that it includes only the variables we need for the next step.

```

library(tidyverse)
Model_df <- Model_df %>%
  dplyr::mutate(rBlst_1 = 8 - Blst_1) #if you had multiple items, you could add a pipe (%>%

```

```
# selecting the variables we want
Model_df <- dplyr::select(Model_df, ID, iBIPOC_pr, cmBlack, rBlst_1, Blst_2:Blst_6,
  cEval_8:cEval_17)
```

### 13.4.1 Analyzing and Managing Missingness

The series of lessons on data preparation in the [Multivariate Modeling](#) volume provide a more detailed review of analyzing and managing missing data. Much of the script below is copied from those lessons and my review and explanation in this lesson is significantly shorter.

Structural equation models lend themselves to managing missing data with Parent's [2013] *available information analysis* (AIA) approach. My approach is to:

- Create a dataframe that includes only the variables that will be used in the analysis.
- Delete all cases with greater than 20% missingness.
- If scale scores (or parcels) are used, calculate them if ~80% of the data for the calculation is present.
- Use the *full information maximum likelihood* (FIML) estimation procedure in *lavaan*; this allows item-level missingness.

```
cases1 <- nrow(Model_df) #I produced this object for the sole purpose of feeding the number o
cases1
```

```
[1] 70
```

```
library(tidyverse)
# Create a variable (n_miss) that counts the number missing
Model_df$n_miss <- Model_df %>%
  dplyr::select(iBIPOC_pr:cEval_17) %>%
  is.na %>%
  rowSums

# Create a proportion missing by dividing n_miss by the total number
# of variables (21) Sort in order of descending frequency to get a
# sense of the missingness
Model_df <- Model_df %>%
  dplyr::mutate(prop_miss = (n_miss/21) * 100) %>%
  arrange(desc(n_miss))

PrMiss1 <- psych::describe(Model_df$prop_miss)
PrMiss1
```

	vars	n	mean	sd	median	trimmed	mad	min	max	range	skew	kurtosis	se
X1	1	70	27.48	40.41	0	23.04	0	0	90.48	90.48	0.87	-1.2	4.83

```
MissMin1 <- formattable::digits(PrMiss1$min, 0) #this object is displayed below and I use inp  
MissMax1 <- formattable::digits(PrMiss1$max, 0)  
MissMin1
```

```
[1] 0
```

```
MissMax1
```

```
[1] 90
```

```
CellsMissing1 <- formattable::percent(mean(is.na(Model_df))) #percent missing across df  
RowsMissing1 <- formattable::percent(mean(complete.cases(Model_df))) #percent of rows with no  
CellsMissing1
```

```
[1] 26.23%
```

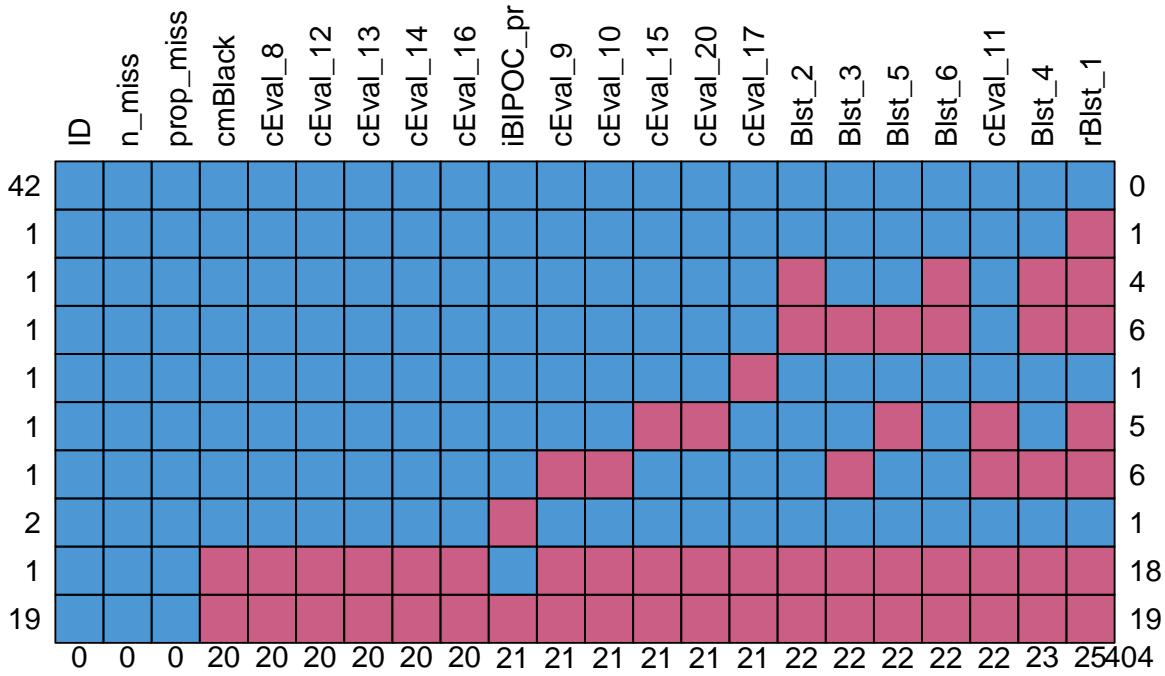
```
RowsMissing1
```

```
[1] 60.00%
```

Our initial inspection of the data indicated that 70 attempted the survey. The proportion of missingness in the responses ranged from 0 to 90. Across the dataframe there was 26% of missingness across the cells. Approximately 60% of the cases had nonmissing data.

Let's conduct an analysis of missingness with the *mice::md.pattern()* function.

```
missingness <- mice::md.pattern(Model_df, plot = TRUE, rotate.names = TRUE)
```



## missingness

```

1      1      1      1      1      1      1      0      1      1
1      1      1      1      1      1      0      0      0      0
1      1      1      1      1      0      1      1      1      1
1      1      1      1      0      0      1      1      1      0
1      1      0      0      1      1      1      1      0      1
2      0      1      1      1      1      1      1      1      1
1      1      0      0      0      0      0      0      0      0
19     0      0      0      0      0      0      0      0      0
                21      21      21      21      21      21      22      22      22
Blst_6 cEval_11 Blst_4 rBlst_1
42     1      1      1      1      0
1      1      1      1      0      1
1      0      1      0      0      4
1      0      1      0      0      6
1      1      1      1      1      1
1      1      0      1      0      5
1      1      0      0      0      6
2      1      1      1      1      1
1      0      0      0      0      18
19     0      0      0      0      19
                22      22      23      25 404

```

We need to decide what is our retention threshold. Twenty percent seems to be a general rule of thumb. Let's delete all cases with missingness at 20% or greater.

```

Model_df <- filter(Model_df, prop_miss <= 20) #update df to have only those with at least 20% missing

Model_df <- dplyr::select(Model_df, iBIPOC_pr:cEval_17) #the variable selection just lops off the last few columns

CasesIncluded <- nrow(Model_df)
CasesIncluded #this object is displayed below and I use input from it for the inline text usage

```

```
[1] 47
```

We should check the missingness characteristics again.

```

CellsMissing2 <- formattable::percent(mean(is.na(Model_df))) #percent missing across df
RowsMissing2 <- formattable::percent(mean(complete.cases(Model_df))) #percent of rows with no missing values
CellsMissing2

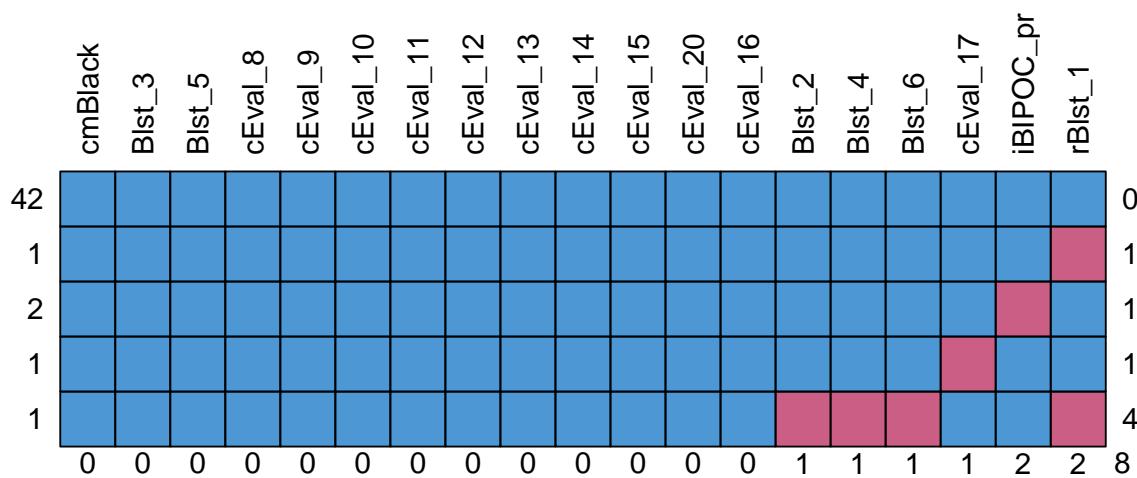
```

```
[1] 0.90%
```

```
RowsMissing2
```

```
[1] 89.36%
```

```
missingness2 <- mice::md.pattern(Model_df, plot = TRUE, rotate.names = TRUE)
```



```
missingness2
```

	cmBlack	Blst_3	Blst_5	cEval_8	cEval_9	cEval_10	cEval_11	cEval_12	cEval_13	cEval_14	cEval_15	cEval_20	cEval_16	Blst_2	Blst_4	Blst_6	cEval_17	iBIPOC_pr	rBlst_1
42	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	cEval_14	cEval_15	cEval_20	cEval_16	Blst_2	Blst_4	Blst_6	cEval_17	iBIPOC_pr	rBlst_1									
42	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	

```

1      1      1      1      1      0      0      0      1      1
      0      0      0      0      1      1      1      1      1
rBlst_1
42     1 0
1      0 1
2      1 1
1      1 1
1      0 4
2 8

```

Write up of results so far:

Our initial inspection of the data indicated that 70 attempted the survey. The proportion of missingness in the responses ranged from 0 to 90. Across the dataframe there was 26% of missingness across the cells. Approximately 60% of the cases had nonmissing data. The predominant pattern of missingness included individuals opening the survey without completing any of the items. Beyond that, our inspection of a missingness map indicated a haphazard pattern of missingness [Enders, 2010].

We decided to delete all cases with greater than 20% missinnes. We reinspected the missingness of the dataset with 47 cases. Across the dataframe there was less than 1% of missingness across the cells. Approximately 89% of the cases had nonmissing data.

### 13.4.2 Assessing the Distributional Characteristics of the Data

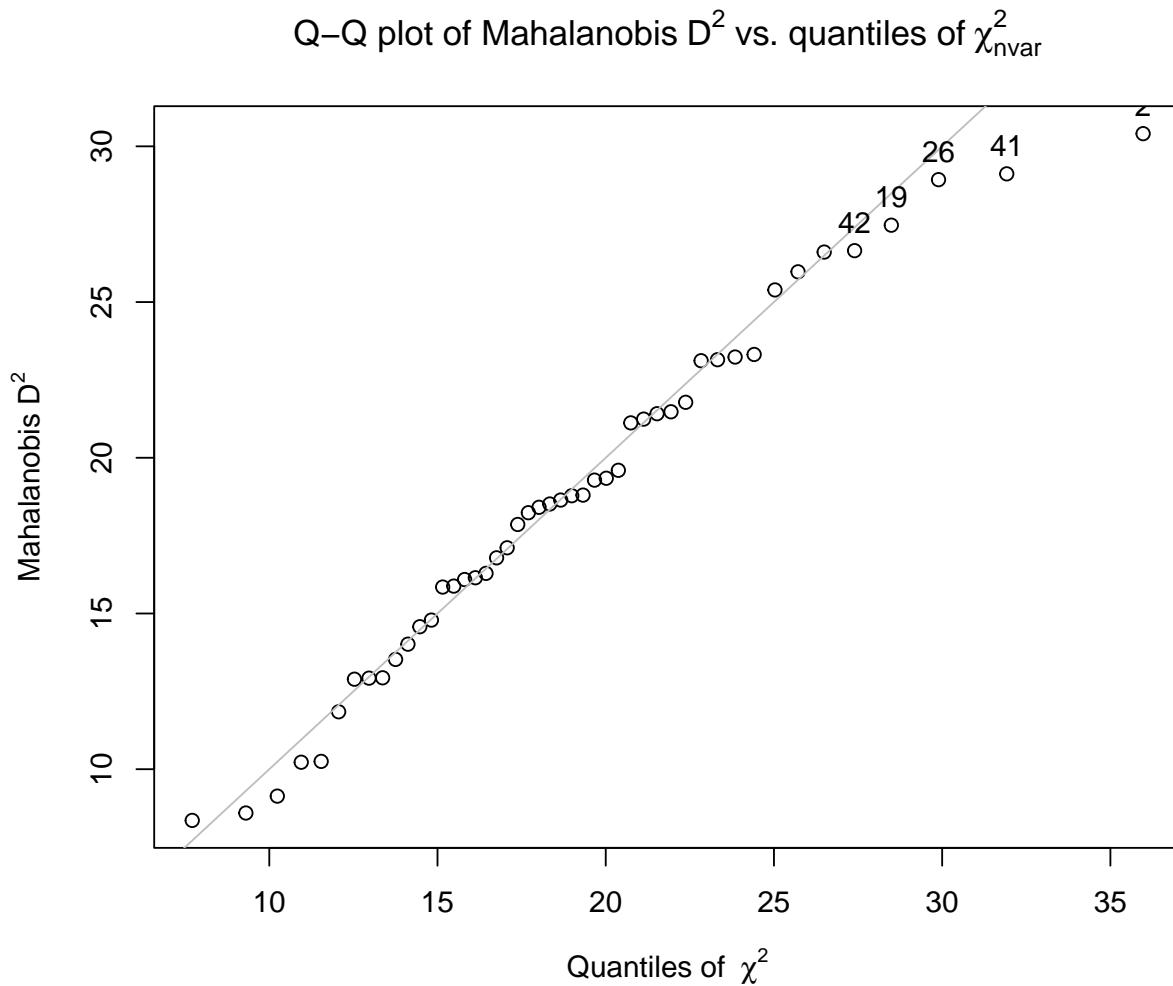
```
psych::describe(Model_df)
```

	vars	n	mean	sd	median	trimmed	mad	min	max	range	skew	kurtosis
iBIPOC_pr	1	45	0.33	0.40	0.25	0.30	0.37	0	1	1	0.76	-1.06
cmBlack	2	47	6.66	7.64	5.00	5.46	7.41	0	29	29	1.29	0.85
rBlst_1	3	45	3.42	1.60	3.00	3.35	1.48	1	7	6	0.48	-0.60
Blst_2	4	46	2.67	1.59	2.00	2.58	1.48	1	6	5	0.30	-1.42
Blst_3	5	47	2.11	1.29	2.00	1.95	1.48	1	6	5	1.06	0.25
Blst_4	6	46	2.35	1.25	2.00	2.24	1.48	1	5	4	0.61	-0.80
Blst_5	7	47	2.00	1.20	2.00	1.87	1.48	1	5	4	0.89	-0.58
Blst_6	8	46	2.26	1.34	2.00	2.08	1.48	1	6	5	0.99	0.04
cEval_8	9	47	4.30	0.83	4.00	4.44	1.48	1	5	4	-1.69	3.98
cEval_9	10	47	3.81	1.06	4.00	3.87	1.48	2	5	3	-0.49	-1.00
cEval_10	11	47	4.00	0.98	4.00	4.13	1.48	1	5	4	-0.96	0.53
cEval_11	12	47	3.40	1.15	3.00	3.41	1.48	1	5	4	0.02	-1.21
cEval_12	13	47	3.85	1.27	4.00	3.97	1.48	1	5	4	-0.73	-0.83
cEval_13	14	47	3.83	1.05	4.00	3.97	0.00	1	5	4	-1.21	1.18
cEval_14	15	47	3.72	1.36	4.00	3.87	1.48	1	5	4	-0.97	-0.30
cEval_15	16	47	3.26	1.19	3.00	3.31	1.48	1	5	4	-0.41	-0.78
cEval_20	17	47	3.55	1.14	4.00	3.67	1.48	1	5	4	-0.73	-0.10

```
cEval_16    18 47 3.13 1.30    3.00    3.15 1.48    1    5     4 -0.23    -1.11
cEval_17    19 46 2.67 1.43    2.00    2.61 1.48    1    5     4   0.35    -1.31
      se
iBIPOC_pr  0.06
cmBlack    1.11
rBlst_1    0.24
Blst_2     0.23
Blst_3     0.19
Blst_4     0.18
Blst_5     0.17
Blst_6     0.20
cEval_8     0.12
cEval_9     0.15
cEval_10    0.14
cEval_11    0.17
cEval_12    0.18
cEval_13    0.15
cEval_14    0.20
cEval_15    0.17
cEval_20    0.17
cEval_16    0.19
cEval_17    0.21
```

Regarding the distributional characteristics of the data, skew and kurtosis values of the variables fell below the values of 3 (skew) and 8 to 20 (kurtosis) that Kline suggests are concerning [2016].

```
Model_df$Mahal <- psych::outlier(Model_df)
```



```
library(dplyr)
# str(item_scores_df$Mahal)
Model_df$MOutlier <- if_else(Model_df$Mahal > (median(Model_df$Mahal) +
  (3 * sd(Model_df$Mahal))), TRUE, FALSE)
```

```
OutlierCount <- Model_df %>%
  count(MOutlier)
OutlierCount
```

```
# A tibble: 1 x 2
  MOutlier     n
  <lgl>     <int>
1 FALSE       47
```

```
NumOutliers <- nrow(Model_df) - OutlierCount #calculating how many outliers
NumOutliers #this object is used for the inline text for the reesults
```

```

MOutlier n
1      47 0

NumOutliers

MOutlier n
1      47 0

head(Model_df)  #shows us the first 6 rows of the data so we can see the new variables (Mahal,
# A tibble: 6 x 21
  iBIPOC_pr cmBlack rBlst_1 Blst_2 Blst_3 Blst_4 Blst_5 Blst_6 cEval_8 cEval_9
    <dbl>    <dbl>    <dbl>    <dbl>    <dbl>    <dbl>    <dbl>    <dbl>    <dbl>    <dbl>
1     1        0     NA     NA      2     NA      2     NA      5      4
2     0        5      2      6      2      2      4      1      5      5
3   NaN       5      4      5      4      5      3      4      4      4
4   NaN      14      3      4      1      3      1      2      4      3
5    0.25     0     NA      1      1      1      1      1      4      5
6    0.333    0      3      3      5      2      2      2      5      5
# i 11 more variables: cEval_10 <dbl>, cEval_11 <dbl>, cEval_12 <dbl>,
#   cEval_13 <dbl>, cEval_14 <dbl>, cEval_15 <dbl>, cEval_20 <dbl>,
#   cEval_16 <dbl>, cEval_17 <dbl>, Mahal <dbl>, MOutlier <lgl>

```

We evaluated multivariate normality with the Mahalanobis distance test. Specifically, we used the `outlier()` function in the `psych` package and included all continuous variables in the calculation. Our visual inspection of the Q-Q plot suggested that the plotted line strayed from the straight line as the quantiles increased. Additionally, we appended the Mahalanobis distance scores as a variable to the data. Analyzing this variable, we found that no exceed three standard deviations beyond the median. *Thus, with no outliers, we assumed multivariate normality and proceeded with the data analysis*

### 13.4.3 Preliminary Analyses

#### 13.4.3.1 Internal Consistency Coefficients

Most research projects start with some preliminary statistics. Even though we will be using item-level data in our hybrid model, for any instruments that are scales, we typically compute internal consistency coefficients and include these values in the last sentence of in the description of the respective measure. In this example we used two scales: Perceptions of the Campus Climate for Black Students and the Course Evaluation items that evaluated the degree to which the pedagogy was socially and culturally responsive. A more thorough description of internal consistency coefficients are found in the [reliability](#) chapter of this volume.

My process for calculating alpha coefficients is to first create a subset of item-level data that is consistently scaled in the same direction. That is, reverse score any items before creating the subset of data.

```
ClimateItems <- dplyr::select(Model_df, rBlst_1, Blst_2, Blst_3, Blst_4,
                                Blst_5, Blst_6)
CEvalItems <- dplyr::select(Model_df, cEval_8, cEval_9, cEval_10, cEval_11,
                            cEval_12, cEval_13, cEval_14, cEval_15, cEval_16, cEval_17, cEval_20)
```

Next, in separate analyses, we apply the *psych::alpha()* function to the scale items.

```
ClimateAlpha <- psych::alpha(ClimateItems)
ClimateAlpha
```

```
Reliability analysis
Call: psych::alpha(x = ClimateItems)

  raw_alpha std.alpha G6(smc) average_r S/N   ase mean   sd median_r
  0.87      0.88     0.88      0.55 7.4 0.029  2.5 1.1      0.53

  95% confidence boundaries
            lower alpha upper
Feldt      0.81  0.87  0.92
Duhachek  0.82  0.87  0.93

Reliability if an item is dropped:
  raw_alpha std.alpha G6(smc) average_r S/N alpha se var.r med.r
rBlst_1    0.88      0.89     0.88      0.62 8.0 0.027 0.016 0.60
Blst_2     0.86      0.87     0.86      0.56 6.5 0.034 0.026 0.53
Blst_3     0.87      0.88     0.87      0.59 7.3 0.031 0.020 0.57
Blst_4     0.83      0.84     0.83      0.52 5.4 0.039 0.019 0.50
Blst_5     0.83      0.83     0.82      0.50 5.1 0.040 0.018 0.47
Blst_6     0.83      0.84     0.83      0.52 5.3 0.039 0.019 0.48

Item statistics
  n raw.r std.r r.cor r.drop mean   sd
rBlst_1 45 0.68 0.65 0.53 0.51 3.4 1.6
Blst_2  46 0.78 0.76 0.70 0.65 2.7 1.6
Blst_3  47 0.69 0.70 0.61 0.57 2.1 1.3
Blst_4  46 0.86 0.86 0.85 0.79 2.3 1.3
Blst_5  47 0.88 0.89 0.89 0.84 2.0 1.2
Blst_6  46 0.86 0.87 0.86 0.79 2.3 1.3

Non missing response frequency for each item
  1   2   3   4   5   6   7 miss
rBlst_1 0.09 0.24 0.22 0.22 0.09 0.09 0.04 0.04
Blst_2  0.37 0.15 0.09 0.24 0.13 0.02 0.00 0.02
Blst_3  0.43 0.30 0.09 0.15 0.02 0.02 0.00 0.00
Blst_4  0.30 0.33 0.15 0.15 0.07 0.00 0.00 0.02
Blst_5  0.47 0.28 0.06 0.17 0.02 0.00 0.00 0.00
```

```
Blst_6 0.35 0.35 0.11 0.11 0.07 0.02 0.00 0.02
```

We learn that the Cronbach's alpha coefficient for the scale assessing perceptions of campus climate for students who are Black is 0.881. This exceeds the recommended threshold of .80. I would simply add a sentence similar to the following to the end of my description of the scale in the Method/Measures section: In our study the estimated internal consistency reliability of the total scale score assessing campus climate was 0.881.

Let's repeat the process for the items assessing the degree to which the pedagogy was socially and culturally responsive.

```
CEvalAlpha <- psych::alpha(CEvalItems)
CEvalAlpha
```

```
Reliability analysis
Call: psych::alpha(x = CEvalItems)

  raw_alpha std.alpha G6(smc) average_r S/N    ase mean   sd median_r
  0.88      0.89     0.92      0.42 7.8 0.025  3.6 0.8     0.41

  95% confidence boundaries
            lower alpha upper
Feldt      0.83  0.88  0.93
Duhachek  0.83  0.88  0.93

Reliability if an item is dropped:
  raw_alpha std.alpha G6(smc) average_r S/N alpha se var.r med.r
cEval_8      0.87      0.88     0.90      0.42 7.2 0.027 0.017  0.42
cEval_9      0.88      0.88     0.91      0.43 7.6 0.026 0.018  0.43
cEval_10     0.87      0.87     0.90      0.41 6.8 0.028 0.019  0.40
cEval_11     0.87      0.87     0.91      0.41 6.9 0.028 0.022  0.39
cEval_12     0.88      0.89     0.92      0.44 7.8 0.025 0.018  0.43
cEval_13     0.87      0.87     0.90      0.41 6.8 0.028 0.019  0.40
cEval_14     0.87      0.87     0.90      0.41 6.9 0.029 0.019  0.41
cEval_15     0.86      0.87     0.90      0.40 6.6 0.030 0.017  0.39
cEval_16     0.87      0.88     0.91      0.42 7.2 0.028 0.021  0.40
cEval_17     0.87      0.88     0.91      0.42 7.2 0.027 0.017  0.41
cEval_20     0.87      0.88     0.90      0.42 7.2 0.028 0.017  0.41

Item statistics
  n raw.r std.r r.cor r.drop mean   sd
cEval_8 47 0.63 0.67 0.65 0.57 4.3 0.83
cEval_9 47 0.57 0.60 0.56 0.48 3.8 1.06
cEval_10 47 0.72 0.75 0.73 0.66 4.0 0.98
cEval_11 47 0.72 0.72 0.68 0.64 3.4 1.15
cEval_12 47 0.55 0.56 0.49 0.44 3.9 1.27
cEval_13 47 0.72 0.74 0.73 0.66 3.8 1.05
```

```
cEval_14 47 0.74 0.72 0.70 0.66 3.7 1.36
cEval_15 47 0.80 0.79 0.79 0.74 3.3 1.19
cEval_16 47 0.68 0.66 0.61 0.59 3.1 1.30
cEval_17 46 0.70 0.66 0.63 0.59 2.7 1.43
cEval_20 47 0.68 0.66 0.64 0.60 3.6 1.14
```

Non missing response frequency for each item

	1	2	3	4	5	miss
--	---	---	---	---	---	------

```
cEval_8 0.02 0.02 0.04 0.47 0.45 0.00
cEval_9 0.00 0.17 0.15 0.38 0.30 0.00
cEval_10 0.02 0.06 0.15 0.43 0.34 0.00
cEval_11 0.02 0.23 0.30 0.21 0.23 0.00
cEval_12 0.04 0.17 0.11 0.26 0.43 0.00
cEval_13 0.06 0.04 0.13 0.53 0.23 0.00
cEval_14 0.15 0.02 0.13 0.36 0.34 0.00
cEval_15 0.11 0.15 0.26 0.36 0.13 0.00
cEval_16 0.15 0.17 0.23 0.30 0.15 0.00
cEval_17 0.26 0.28 0.13 0.17 0.15 0.02
cEval_20 0.09 0.06 0.26 0.40 0.19 0.00
```

The alpha coefficient for the course evaluation items assessing a socially and culturally responsive pedagogy was 0.887. I would add this sentence to the description of this measure.

### 13.4.3.2 Means, SDs, r-matrix

Means, standard deviations, and a correlation matrix are also commonly reported. Because two of our constructs are scales, we will need to calculate their means for cases that have met the minimum threshold for nonmissingness.

```
# create lists of the items
ClimateVars <- c("rBlst_1", "Blst_2", "Blst_3", "Blst_4", "Blst_5", "Blst_6")
CEvalVars <- c("cEval_8", "cEval_9", "cEval_10", "cEval_11", "cEval_12",
  "cEval_13", "cEval_14", "cEval_15", "cEval_16", "cEval_17", "cEval_20")

# calculate means for when a specified proportion of items are
# non-missing
Model_df$ClimateM <- sjstats::mean_n(Model_df[, ClimateVars], 0.8) #will create the mean for each row
Model_df$CEvalM <- sjstats::mean_n(Model_df[, CEvalVars], 0.8) #will create the mean for each row
```

The *apaTables::cor.table* function creates the standard table that will include the means, standard deviations, and correlation matrix.

```
apaTables::apa.cor.table(Model_df[c("ClimateM", "CEvalM", "iBIPOC_pr",
  "cmBlack")], landscape = TRUE, table.number = 1, filename = "Table1_Prelim.doc")
```

Table 1

Means, standard deviations, and correlations with confidence intervals

Variable	M	SD	1	2	3
1. ClimateM	2.46	1.09			
2. CEvalM	3.60	0.80	-.30*		
			[-.55, -.02]		
3. iBIPOC_pr	0.33	0.40	-.26	.24	
			[-.52, .04]	[-.06, .50]	
4. cmBlack	6.66	7.64	-.19	.08	-.07
			[-.45, .11]	[-.21, .36]	[-.36, .23]

Note. M and SD are used to represent mean and standard deviation, respectively.  
Values in square brackets indicate the 95% confidence interval.

The confidence interval is a plausible range of population correlations that could have caused the sample correlation (Cumming, 2014).

\* indicates  $p < .05$ . \*\* indicates  $p < .01$ .

### 13.4.4 Summary of Data Preparation

We began by creating a dataset that included only the variables of interest. Our initial inspection of the data indicated that 70 attempted the survey. The proportion of missingness in the responses ranged from 0 to 90. Across the dataframe there was 26% of missingness across the cells. Approximately 60% of the cases had nonmissing data. The predominant pattern of missingness included individuals opening the survey without completing any of the items. Beyond that, our inspection of a missingness map indicated a haphazard pattern of missingness [Enders, 2010].

Using Parent's *available item analysis* [AIA; -Parent [2013]] as a guide, we deleted all cases where there was greater than 20% of data missing. We reinspected the missingness of the dataset with 47 cases. Across the dataframe there was less than 1% of missingness across the cells. Approximately 89% of the cases had nonmissing data.

Regarding the distributional characteristics of the data, skew and kurtosis values of the variables fell below the values of 3 (skew) and 8 to 20 (kurtosis) that Kline suggests are concerning [2016]. We evaluated multivariate normality with the Mahalanobis distance test. Specifically, we used the *outlier()* function in the *psych* package and included all continuous variables in the calculation. Our visual inspection of the Q-Q plot suggested

that the plotted line strayed from the straight line as the quantiles increased. Additionally, we appended the Mahalanobis distance scores as a variable to the data. Analyzing this variable, we found that no exceed three standard deviations beyond the median. Thus, with no outliers, we assumed multivariate normality and proceeded with the data analysis. Means, standard deviations, and a correlation matrix are found in Table 1.

## 13.5 The Measurement Model: Specification and Evaluation

Structural regression (e.g., structural equation, hybrid) models include both *measurement* and *structural* portions. The **measurement model** examines the relationship between latent variables and their measures.

- Testing the measurement model means *saturating* it, such that  $df = 0$  and it is *just-identified*.
- Essentially, the measurement model is a correlated factors model. However, rather than having subscales of a larger scale, these are all the LVs involved in your model.
- Testing the measurement model points out any misfit in the measurement model (that you need to fix). Heywood cases(usually a negative error variance, which is an impossible solution) are an example of a problem that would need to be addressed before fixing.

The **structural model** evaluates the hypothesized relations between the latent variables.

- The structural model is typically more parsimonious (i.e., not saturated) than the measurement model and is characterized by directional paths (not covariances) between some (not all) of the variables.

The specification of our measurement model resembles the first-order, correlated traits specifications in prior lessons. What differs is that we include all latent variables and their specifications. Below, there are no surprises about the Climate and CourseEval latent variables, because these are traditional scales and they have at least three items/indicators. In contrast, latent variables with one and two indicators requires special treatment.

For two-indicator latent variables, Little et al. [2002] recommended placing an equality constraint on the two loadings associated with the construct because this would locate the construct at the true intersection of the two selected indicators. Procedurally this is fairly straightforward. If we wanted to create a latent variable from the proportions of (a) instructional staff and (b) classmates who are Black we would simply assign labels to the two indicators:

```
#PrBlack =~ v1*iBIPOC_pr + v1*cmBlack
```

For single indicator latent variables, Little et al. [2002] wrote, “a single-indicator latent variable is essentially equivalent to a manifest variable. In this case, the error of measurement is either fixed at zero or fixed at a non-zero estimate of unreliability; additionally a second corresponding parameter would also need to be fixed because of issue of identification.”

Our proportion of instructional staff who are BIPOC and estimated proportion of classmates who are Black were estimated with one item each. In order to include single items as latent variables, we set the observed variable to be 0.00. In essence, this says that the latent variable will account for all of the variance in the observed variable. Note that for each of the single-item variables, there are two lines of code. The first, defines the LV from the item; the second specifies the error variance of the single observed variable to be 0.00.

```
msmt <- "
#latent variable definitions for the factors with 3 or more indicators
Climate =~ rBlst_1 + Blst_4 + Blst_6 + Blst_2 + Blst_3 + Blst_5
CourseEval =~ cEval_8 + cEval_9 + cEval_10 + cEval_11 + cEval_12 + cEval_13 + cEval_14 + cE

#latent variable definitions for the factors with 1 indicator; we set variance of the observed
tBIPOC =~ iBIPOC_pr #for the factor \"t\" is teacher; for variable \"i\" is instructor
sBlack =~ cmBlack #for factor \"s\" is student; for variable \"cm\" is classmates

iBIPOC_pr ~~ 0*iBIPOC_pr #this specifies the error variance of the single observed variable
cmBlack ~~ 0*cmBlack
"
```

### 13.5.0.1 Managing missing data with FIML

If the data contain missing values, the default behavior in *lavaan* is listwise deletion. If we can presume that the missing mechanism is MCAR or MAR (e.g., there is no systematic missingness), we can specify a *full information maximum likelihood* (FIML) estimation procedure with the argument *missing = "ml"* (or its alias *missing = "fiml"*). Recall that we retained cases if they had 20% or less missing. Usin the "fiml" option is part of the AIA approach [Parent, 2013].

```
msmt_fit <- lavaan::cfa(msmt, data = Model_df, missing = "fiml", check.gradient = FALSE)
# msmt_fit <- lavaan::cfa(msmt, data = Model_df, missing = 'fiml',
# estimator = 'ML', bounds = 'wide')
mifitsum <- lavaan::summary(msmt_fit, fit.measures = TRUE, standardized = TRUE,
    rsquare = TRUE)
# missing = 'fiml',
mifitsum
```

*lavaan* 0.6.17 ended normally after 80 iterations

Estimator	ML
Optimization method	NLMINB
Number of model parameters	61
Number of observations	47
Number of missing patterns	5

Model Test User Model:

Test statistic	239.689
Degrees of freedom	148
P-value (Chi-square)	0.000

Model Test Baseline Model:

Test statistic	590.701
Degrees of freedom	171
P-value	0.000

User Model versus Baseline Model:

Comparative Fit Index (CFI)	0.782
Tucker-Lewis Index (TLI)	0.748
Robust Comparative Fit Index (CFI)	0.774
Robust Tucker-Lewis Index (TLI)	0.739

Loglikelihood and Information Criteria:

Loglikelihood user model (H0)	-1282.408
Loglikelihood unrestricted model (H1)	-1162.563
Akaike (AIC)	2686.815
Bayesian (BIC)	2799.674
Sample-size adjusted Bayesian (SABIC)	2608.355

Root Mean Square Error of Approximation:

RMSEA	0.115
90 Percent confidence interval - lower	0.087
90 Percent confidence interval - upper	0.141
P-value H_0: RMSEA <= 0.050	0.000
P-value H_0: RMSEA >= 0.080	0.980
Robust RMSEA	0.118
90 Percent confidence interval - lower	0.090
90 Percent confidence interval - upper	0.144
P-value H_0: Robust RMSEA <= 0.050	0.000
P-value H_0: Robust RMSEA >= 0.080	0.985

Standardized Root Mean Square Residual:

SRMR	0.100
------	-------

Parameter Estimates:

Standard errors	Standard
-----------------	----------

	Information	Observed				
	Observed information based on	Hessian				
<b>Latent Variables:</b>						
	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
Climate ==						
rBlst_1	1.000				0.851	0.538
Blst_4	1.242	0.322	3.856	0.000	1.057	0.860
Blst_6	1.343	0.347	3.866	0.000	1.142	0.867
Blst_2	1.267	0.368	3.441	0.001	1.078	0.687
Blst_3	0.935	0.289	3.232	0.001	0.796	0.624
Blst_5	1.247	0.319	3.909	0.000	1.061	0.895
CourseEval ==						
cEval_8	1.000				0.508	0.617
cEval_9	1.067	0.338	3.153	0.002	0.542	0.519
cEval_10	1.352	0.331	4.087	0.000	0.687	0.710
cEval_11	1.503	0.404	3.719	0.000	0.763	0.668
cEval_12	1.184	0.409	2.895	0.004	0.601	0.479
cEval_13	1.475	0.361	4.089	0.000	0.749	0.722
cEval_14	1.870	0.505	3.701	0.000	0.949	0.704
cEval_15	1.824	0.460	3.961	0.000	0.926	0.788
cEval_20	1.413	0.428	3.301	0.001	0.717	0.637
cEval_16	1.549	0.457	3.391	0.001	0.786	0.613
cEval_17	1.835	0.539	3.403	0.001	0.932	0.654
tBIPOC ==						
iBIPOC_pr	1.000				0.396	1.000
sBlack ==						
cmBlack	1.000				7.560	1.000
<b>Covariances:</b>						
	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
Climate ~~						
CourseEval	-0.138	0.086	-1.604	0.109	-0.319	-0.319
tBIPOC	-0.075	0.057	-1.320	0.187	-0.224	-0.224
sBlack	-1.519	1.067	-1.423	0.155	-0.236	-0.236
CourseEval ~~						
tBIPOC	0.054	0.034	1.588	0.112	0.267	0.267
sBlack	0.382	0.605	0.631	0.528	0.099	0.099
tBIPOC ~~						
sBlack	-0.201	0.442	-0.455	0.649	-0.067	-0.067
<b>Intercepts:</b>						
	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
.rBlst_1	3.402	0.234	14.507	0.000	3.402	2.151
.Blst_4	2.346	0.180	13.030	0.000	2.346	1.909
.Blst_6	2.259	0.193	11.712	0.000	2.259	1.715
.Blst_2	2.672	0.230	11.605	0.000	2.672	1.704
.Blst_3	2.106	0.186	11.321	0.000	2.106	1.651

.Blst_5	2.000	0.173	11.571	0.000	2.000	1.688
.cEval_8	4.298	0.120	35.804	0.000	4.298	5.223
.cEval_9	3.809	0.152	24.997	0.000	3.809	3.646
.cEval_10	4.000	0.141	28.342	0.000	4.000	4.134
.cEval_11	3.404	0.167	20.433	0.000	3.404	2.980
.cEval_12	3.851	0.183	21.047	0.000	3.851	3.070
.cEval_13	3.830	0.151	25.295	0.000	3.830	3.690
.cEval_14	3.723	0.197	18.936	0.000	3.723	2.762
.cEval_15	3.255	0.171	18.987	0.000	3.255	2.769
.cEval_20	3.553	0.164	21.629	0.000	3.553	3.155
.cEval_16	3.128	0.187	16.727	0.000	3.128	2.440
.cEval_17	2.704	0.209	12.923	0.000	2.704	1.898
.iBIPOC_pr	0.330	0.059	5.585	0.000	0.330	0.831
.cmBlack	6.660	1.103	6.039	0.000	6.660	0.881

Variances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
.iBIPOC_pr	0.000				0.000	0.000
.cmBlack	0.000				0.000	0.000
.rBlst_1	1.776	0.388	4.579	0.000	1.776	0.710
.Blst_4	0.394	0.111	3.544	0.000	0.394	0.261
.Blst_6	0.430	0.126	3.425	0.001	0.430	0.248
.Blst_2	1.298	0.295	4.393	0.000	1.298	0.528
.Blst_3	0.994	0.217	4.577	0.000	0.994	0.611
.Blst_5	0.279	0.094	2.986	0.003	0.279	0.199
.cEval_8	0.419	0.098	4.295	0.000	0.419	0.619
.cEval_9	0.797	0.176	4.542	0.000	0.797	0.731
.cEval_10	0.465	0.119	3.920	0.000	0.465	0.496
.cEval_11	0.723	0.165	4.374	0.000	0.723	0.554
.cEval_12	1.212	0.260	4.666	0.000	1.212	0.770
.cEval_13	0.516	0.127	4.062	0.000	0.516	0.479
.cEval_14	0.916	0.221	4.139	0.000	0.916	0.504
.cEval_15	0.524	0.145	3.608	0.000	0.524	0.379
.cEval_20	0.754	0.178	4.229	0.000	0.754	0.594
.cEval_16	1.025	0.229	4.483	0.000	1.025	0.624
.cEval_17	1.161	0.275	4.230	0.000	1.161	0.572
Climate	0.724	0.381	1.901	0.057	1.000	1.000
CourseEval	0.258	0.119	2.172	0.030	1.000	1.000
tBIPOC	0.157	0.033	4.757	0.000	1.000	1.000
sBlack	57.161	11.791	4.848	0.000	1.000	1.000

R-Square:

	Estimate
iBIPOC_pr	1.000
cmBlack	1.000
rBlst_1	0.290
Blst_4	0.739
Blst_6	0.752

Blst_2	0.472
Blst_3	0.389
Blst_5	0.801
cEval_8	0.381
cEval_9	0.269
cEval_10	0.504
cEval_11	0.446
cEval_12	0.230
cEval_13	0.521
cEval_14	0.496
cEval_15	0.621
cEval_20	0.406
cEval_16	0.376
cEval_17	0.428

### 13.5.0.2 Interpreting the Output

Criteria	Our Results	Criteria met?
Factor loadings significant, strong, proper valence		Yes
Non-significant chi-square	$\chi^2(148) = 239.689, *p* < 0.001$	No
$CFI \geq .95$	$CFI = 0.782$	No
$RMSEA \leq .05$ (but definitely < .10)	$RMSEA = 0.115, 90\%CI(0.087, 0.141)$	No
$SRMR \leq .08$ (but definitely < .10)	$SRMR = 0.100$	No
Combination rule: $CFI \geq .95$ and $SRMR \leq .08$	$CFI = '0.782 , SRMR = 0.100$	No

**Measurement model.** A model that allowed the latent variables (representing the measurement models of all the latent variables) to correlate had clearly unacceptable fit to the data:  $\chi^2(148) = 239.689, *p* < 0.001, CFI = 0.782, RMSEA = 0.115, 90$ .

Before discussing our options, let's look at what we have just specified and evaluated.

The following code can be used to write a table to a .csv file for use in creating tables for APA style results.

```
vb1s <- c(rBlst_1 = "My university provides a supportive environment for Black students",
  Blst_4 = "My university is unresponsive to the needs of Black students",
  Blst_6 = "My university is cold and uncaring toward Black students and race-related issues",
  Blst_2 = "Anti-Black racism is visible in my campus", Blst_3 = "Negative attitudes toward p
  Blst_5 = "Students who are Black are harassed in my university", cEval_8 = "Students felt r
  cEval_9 = "A sense of community developed among the course participants",
  cEval_10 = "The learning environment was inclusive for students with diverse backgrounds an
  cEval_11 = "Elements of universal design were used to increase accessibility",
```

```

cEval_12 = "Course materials were free or no cost to students", cEval_13 = "Where applicable, course materials were free or no cost to students",
cEval_14 = "There was a discussion about race ethnicity culture and course content",
cEval_15 = "Course content included topics related to social justice",
cEval_16 = "Students and instructors shared personal pronouns", cEval_17 = "A land acknowledgement was present in the course",
cEval_20 = "Course content included topics related to social justice",
iBIPOC_pr = "Proportion of Instructors who are BIPOC", cmBlack = "Proportion of Classmates who are Black"

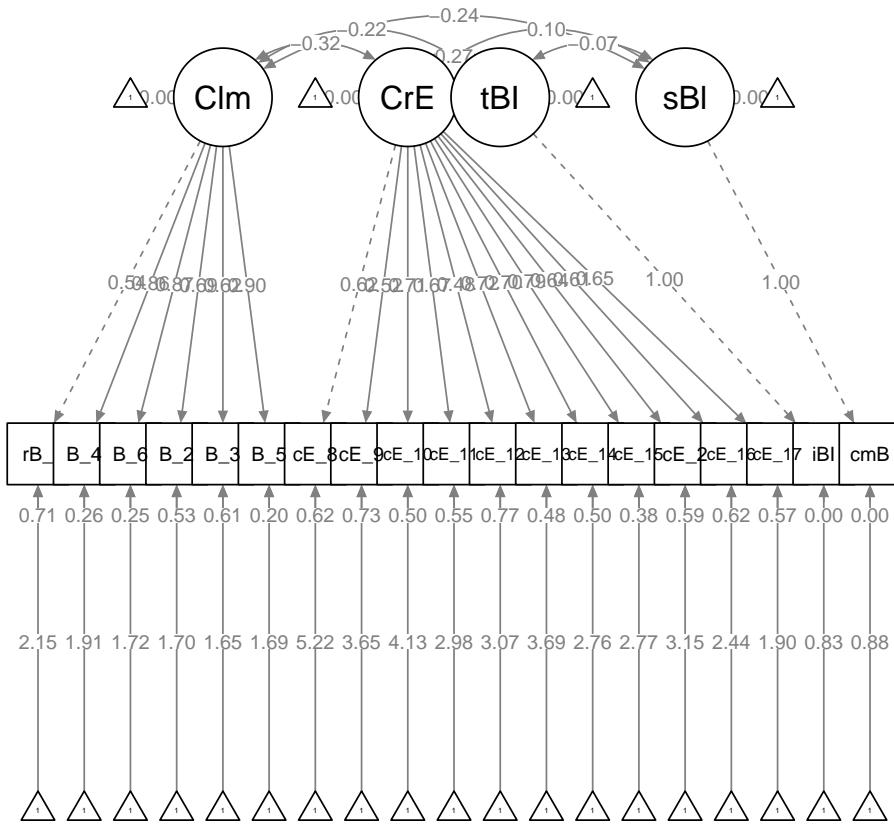
```

Table <- semTable::semTable(msmt\_fit, columnLabels = c(eststars = "Estimate",
 se = "SE", p = "p-value"), fits = c("chisq", "df", "pvalue", "cfi",
 "rmsea", "rmsea.ci.lower", "rmsea.ci.upper", "srmr", "aic", "bic"),
 varLabels = vbls, file = "msmt\_fit", type = "csv", print.results = TRUE)

```

plot_m1 <- semPlot::semPaths(msmt_fit, what = "col", whatLabels = "stand",
  sizeMan = 5, node.width = 1, edge.label.cex = 0.75, style = "lisrel",
  mar = c(5, 5, 5, 5))

```



We can further edit the `semPlot::semPath` object to illustrate how all the latent variables are free to covary.

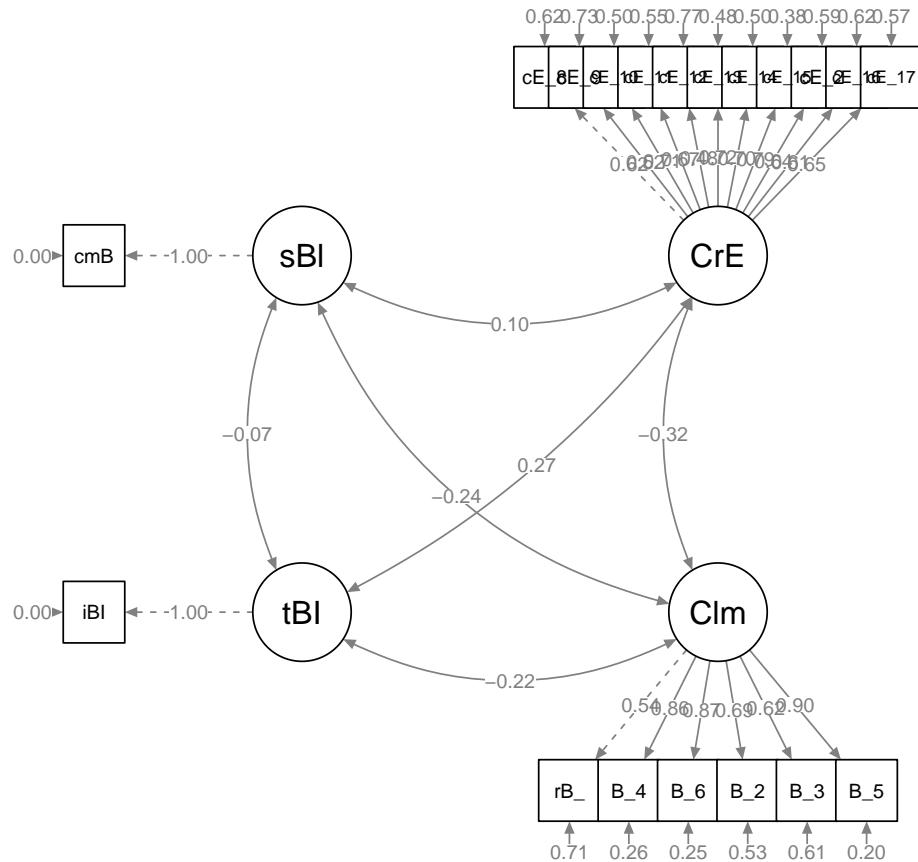
```

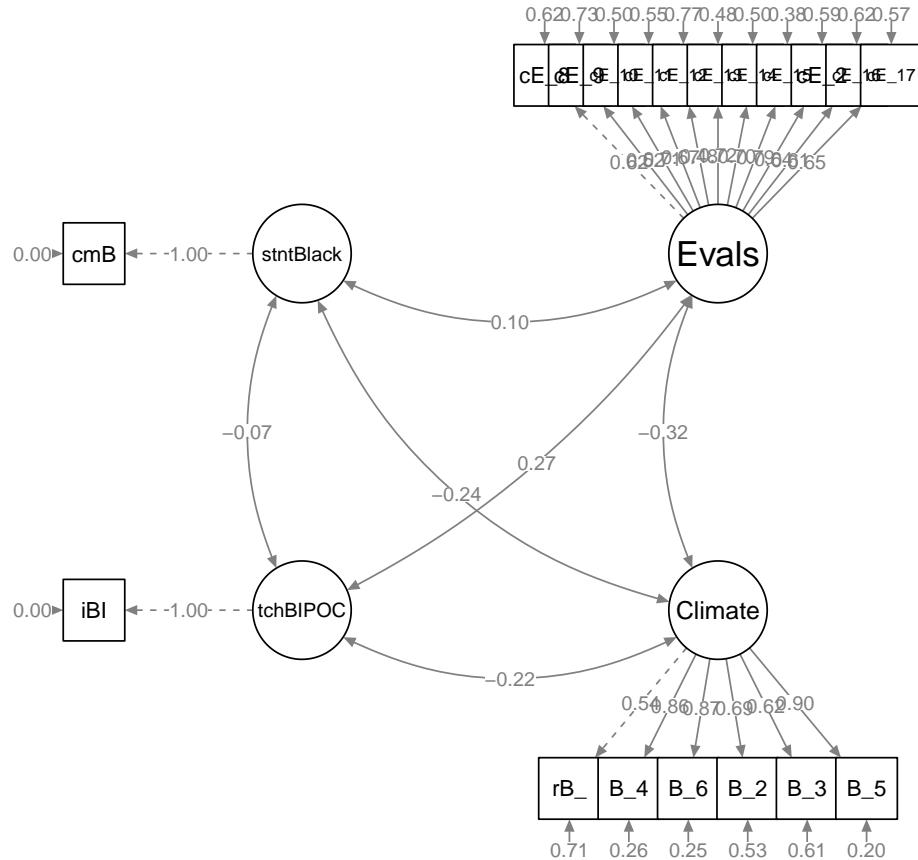
#IMPORTANT: Must use the node names (take directly from the SemPlot) assigned by SemPlot
#You can change them as the last thing
m1_msmt <- semptools::layout_matrix(sBl = c(1,1),
                                    tBI = c(2,1),
                                    CrE = c(1,2),
                                    Clm = c(2,2))
#m_msmt #can check to see if it is what you thought you did

#tell where you want the indicators to face
m1_point_to <- semptools::layout_matrix (left = c(1,1),
                                         left = c(2,1),
                                         up = c(1,2),
                                         down = c(2,2))
#the next two codes -- indicator_order and indicator_factor are paired together, they specify
m1_indicator_order <- c("cmB",
                        "iBI",
                        "cE_8","cE_9","cE_10","cE_11","cE_12","cE_13","cE_14","cE_15","cE_2","cE_16",
                        "rB_","B_4", "B_6", "B_2", "B_3", "B_5")
m1_indicator_factor <- c("sBl",
                          "tBI",
                          "CrE","CrE","CrE","CrE","CrE","CrE","CrE","CrE","CrE",
                          "Clm", "Clm", "Clm", "Clm", "Clm", "Clm")
#next set of code pushes the indicator variables away from the factor
m1_indicator_push <- c(sBl = 2.5, #pushing the 1-item indicators only a little way away
                       tBI = 2.5,
                       CrE = 2.5, #pushing the multi-item indicators further away)
                       Clm = 2.5)
m1_indicator_spread <- c(CrE = 2, #spreading the boxes away from each other
                           Clm = 2)

msmtpplot1 <- semptools::set_sem_layout(plot_m1,
                                         indicator_order = m1_indicator_order,
                                         indicator_factor = m1_indicator_factor,
                                         factor_layout = m1_msmt,
                                         factor_point_to = m1_point_to,
                                         indicator_push = m1_indicator_push,
                                         indicator_spread = m1_indicator_spread)
plot(msmtpplot1)

```





As we can see in the figure, our measurement model has allowed all the latent variables to correlate. Unfortunately, the fit is extremely sub par. In other words: this fit stinks. For this lesson, I will move onto testing a structural model. However, there is hope. Researchers might consider parceling. This is explained more fully in [ReCentering Psych Stats: Multivariate Modeling](#).

## 13.6 The Structural Model: Specification and Evaluation

Here's a quick reminder of the hypothesized model. The model is *hybrid* because it include measurement models (the CFAs for the two Course Evaluation and Perceptions of Campus Climate for Black Students scales), plus the hypothesized paths.

Having just confirmed that our measurement model is adequate, we now replace the covariances between latent variables with the paths (directional) and covariances we hypothesize. These paths and covariances are *soft* hypotheses. That is, we are “freeing” them to relate. The *hard* hypotheses are where no path/covariance exists and the relationship between these variables is “fixed” to zero. This is directly related to degrees of freedom and the identification status (just-identified, over-identified, underidentified) of the model.

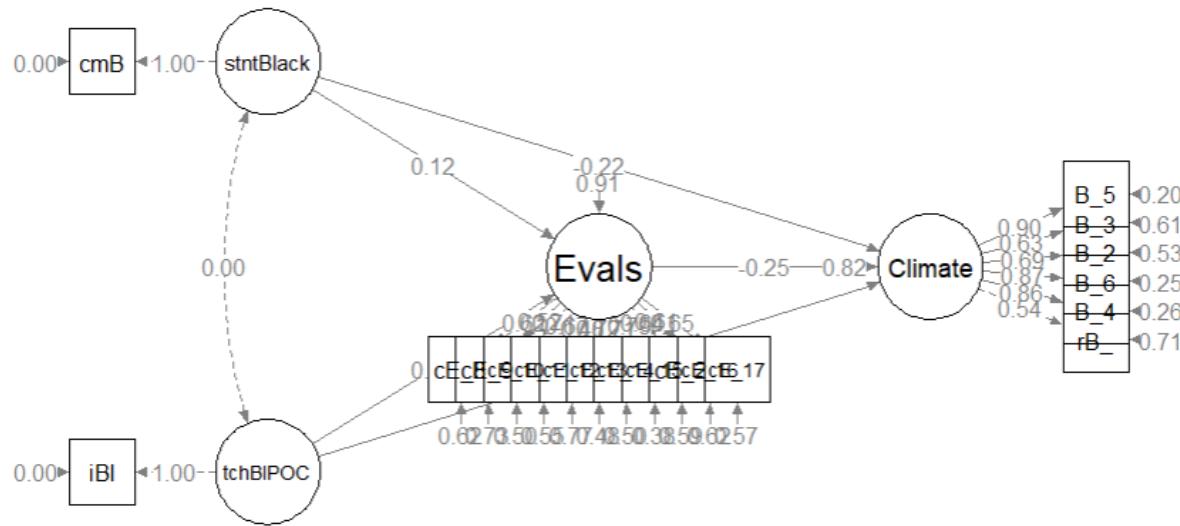


Figure 13.2: Image of the proposed statistical model

### 13.6.1 Model Identification

There are two necessary elements for identifying any type of SEM [Kline, 2016], these include

- having degrees of freedom greater-than-or-equal to zero ( $df_M \geq 0$ ), and
- assigning a scale to every latent variable (including disturbances or error terms).
  - We covered this criterion in the lessons on CFA.

In the case of the specification of standard CFA models (i.e., the models we use in the psychometric evaluation of measures and surveys), the extent of our “your model must be identified” conversation stopped at:

- unidimensional models need to have a minimum of 3 items/indicators (manifest variables) per factor/scale (latent variable)
- multidimensional models need to have a minimum of 2 items/indicators (manifest variables) per factor/scale (latent variable)
- second order factors need three first-order factors in order to be identified
- nonstandard models include error variances that are free to correlate – they need closer scrutiny with regard to identification status

Model identification, though, is more complicated than that. Let’s take a closer look at model identification in hybrid models as it relates to the  $df_M \geq 0$  criteria.

**Underidentified or undetermined** models have fewer observations (knowns) than free model parameters (unknowns). This results in negative degrees of freedom ( $df_M \leq 0$ ). This means that it is impossible to find a unique set of estimates. The classic example for this is:  $a + b = 6$  where there are an infinite number of solutions.

**Just-identified or just-determined** models have an equal number of observations (knowns) as free parameters (unknowns). This results in zero degrees of freedom ( $df_M = 0$ ). Just-identified scenarios will result in a unique solution. The classic example for this is

$$a + b = 6$$

$$2a + b = 10$$

The unique solution is  $a = 4$ ,  $b = 2$ .

**Over-identified or overdetermined** models have more observations (knowns) than free parameters (unknowns). This results in positive degrees of freedom ( $df_M > 0$ ). In this circumstance, there is no single solution, but one can be calculated when a statistical criterion is applied. For example, there is no single solution that satisfies all three of these formulas:

$$a + b = 6$$

$$2a + b = 10$$

$$3a + b = 12$$

When we add this instruction “Find value of  $a$  and  $b$  that yield total scores such that the sum of squared differences between the observations (6, 10, 12) and these total scores is as small as possible.” Curious about the answer? An excellent description is found in Kline [2016].

Model identification is an incredibly complex topic. It is possible to have theoretically identified models and yet they are statistically unidentified and then the researcher must hunt for the source of the problem. For this lesson, I will simply walk through the steps that are commonly used in determining the identification status of a structural model.

### 13.6.1.1 Model identification for the overall SEM

In order to be evaluated, structural models need to be *just identified* ( $df_M = 0$ ) or *overidentified* ( $df_M > 0$ ). Computer programs are not (yet) good at estimating identification status because it is based on symbolism and not numbers. Therefore, we researchers must do the mental math to ensure that our *knowns* (measured/observed variables) are equal (just-identified) or greater than (overidentified) our *unknowns* (parameters that will be estimated).

We calculate the *knowns* by identifying the number of measured variables ( $n$ ) and popping that number into this equation:  $\frac{n(n+1)}{2}$ . *Unknowns* are counted and include: measurement regression paths, structural regression paths, error covariances, residual error variances, and covariances.

Lets calculate this for our model.

- **Knowns:** There are 19 observed variables, so we have  $190(19(19+1)/2)$  pieces of information from which to drive the parameters of the model.

- **Unknowns:** We must estimate the following parameters
  - 17 measurement regression paths (we don't count the marker variables or the single-indicator items)
  - 5 structural regression paths
  - 17 error covariances (1 for each indicator variable)
  - 2 residual error variances (any endogenous latent variable has one of these)
  - 0 covariances
  - We have a total of: 41 unknowns

```
19*(19+1)/2
```

```
[1] 190
```

```
17 + 5 + 17 + 2
```

```
[1] 41
```

Our overall model is overidentified with  $df_M = 41$ . We know this because subtracted the unknowns (41) from the knowns (190). If we calculated this correctly, 41 will be the degrees of freedom associated with the chi-square test.

```
190-41
```

```
[1] 149
```

### 13.6.1.2 Model identification for the structural portion of the model

It is possible to have an overidentified model but still be underidentified in the structural portion. In order to be evaluated, structural models need to be *just identified* ( $df_M = 0$ ) or *overidentified* ( $df_M > 0$ ). Before continuing, it is essential to understand that the structural part is (generally) the relations between the latent variables (although in some models there could be observed variables). In our case, our structural model consists only of four latent variables.

Especially for the structural portion of the model, statistical packages are not (yet) good at estimating identification status because it is based on symbolism and not numbers. Therefore, we researchers must make the calculations to ensure that our *knowns* are equal (just-identified) or greater than (overidentified) our *unknowns*.

- **Knowns:**  $\frac{k(k+1)}{2}$  where  $k$  is the number of *constructs* (humoR: *konstructs?*) in the model. In our case, we have four constructs:  $4(4+1)/2 = 10$
- **Unknowns:** are calculated with the following
  - Exogenous (predictor) variables (1 variance estimated for each): we have 2 (stntBlack, tchBIPOC)

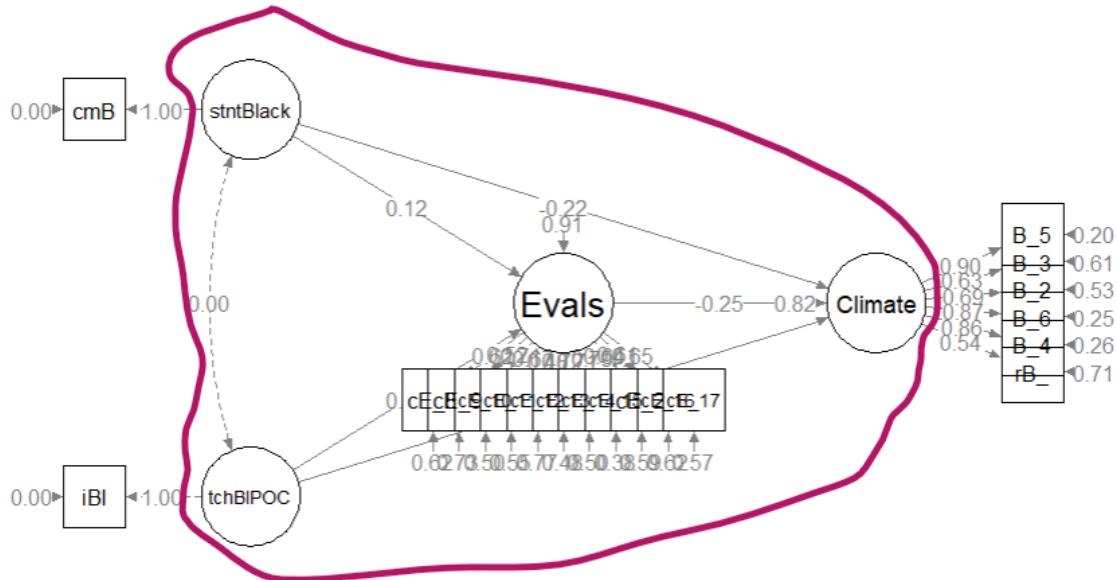


Figure 13.3: A red circle identifies the structural portion of our hybrid model

- Endogenous (predicted) variables (1 disturbance variance for each): we have 2 (Evals, Climate)
- Correlations between variables (1 covariance for each pairing): we have 0 (the potential covariance between stntBlack and tchBIPOC is not specified)
- Regression paths (arrows linking exogenous variables to endogenous variables): we have 5

With 10 knowns and 9 unknowns, we have 1 degree of freedom in the structural portion of the model. This is an overidentified model and we are all set to evaluate it.

As a reminder, the *measurement* model will always have better fit because it is a fully saturated (i.e., covariances between all latent variables), just-identified,  $df_M = 0$ , structure will best replicate the sample covariance matrix. Our hope is that replacing covariances (double-headed arrows) with unidirectional paths and constraining some relations to be 0.0 will not result in a substantial deterioration of fit. That is to say, we hope that our more parsimonious model explains (or captures) the pattern of relations happening in the variance/covariance matrix.

### 13.6.2 Specifying and Evaluating the Structural Model

In the script below we retain the measurement definitions for the latent variables. Our structural paths, though, reflect our hypotheses. The topic of [parallel mediation](#) is addressed in the context

of path analysis in the [Multivariate Modeling](#) volume. Describing it is beyond the scope of this chapter.

```
struct1 <- '
#latent variable definitions for the factors with 3 or more indicators
Climate =~ rBlst_1 + Blst_4 + Blst_6 + Blst_2 + Blst_3 + Blst_5
CourseEval =~ cEval_8 + cEval_9 + cEval_10 + cEval_11 + cEval_12 + cEval_13 + cEval_14 + cEval_15

#latent variable definitions for the factors with 1 indicator; we set variance of the observed variables to 1
tBIPOC =~ iBIPOC_pr #for the factor "t" is teacher; for variable "i" is instructor
sBlack =~ cmBlack #for factor "s" is student; for variable "cm" is classmates

iBIPOC_pr ~~ 0*iBIPOC_pr #this specifies the error variance of the single observed variable
cmBlack ~~ 0*cmBlack

#structural paths
Climate ~ b*CourseEval + c_p1*tBIPOC + c_p2*sBlack
CourseEval ~ a1*tBIPOC + a2*sBlack

#script that produces information about indirect, direct, and total effects
indirect1 := a1 * b
indirect2 := a2 * b
contrast := indirect1 - indirect2
total_indirects := indirect1 + indirect2
total_c := c_p1 + c_p2 + (indirect1) + (indirect2)
direct1 := c_p1
direct2 := c_p2
'
```

Next we use the `lavaan::sem()` function to run the script.

```
# note change in script from cfa to sem
struct1_fit <- lavaan::sem(struct1, data = Model_df, missing = "fiml",
    orthogonal = TRUE)
s1fitsum <- lavaan::summary(struct1_fit, fit.measures = TRUE, standardized = TRUE,
    rsquare = TRUE)
s1fitsum
```

`lavaan 0.6.17 ended normally after 67 iterations`

Estimator	ML
Optimization method	NLMINB
Number of model parameters	60
Number of observations	47
Number of missing patterns	5

Model Test User Model:

Test statistic	239.897
Degrees of freedom	149
P-value (Chi-square)	0.000

Model Test Baseline Model:

Test statistic	590.701
Degrees of freedom	171
P-value	0.000

User Model versus Baseline Model:

Comparative Fit Index (CFI)	0.783
Tucker-Lewis Index (TLI)	0.751
Robust Comparative Fit Index (CFI)	0.776
Robust Tucker-Lewis Index (TLI)	0.742

Loglikelihood and Information Criteria:

Loglikelihood user model (H0)	-1282.512
Loglikelihood unrestricted model (H1)	-1162.563
Akaike (AIC)	2685.024
Bayesian (BIC)	2796.032
Sample-size adjusted Bayesian (SABIC)	2607.850

Root Mean Square Error of Approximation:

RMSEA	0.114
90 Percent confidence interval - lower	0.087
90 Percent confidence interval - upper	0.140
P-value H_0: RMSEA <= 0.050	0.000
P-value H_0: RMSEA >= 0.080	0.977
Robust RMSEA	0.117
90 Percent confidence interval - lower	0.089
90 Percent confidence interval - upper	0.143
P-value H_0: Robust RMSEA <= 0.050	0.000
P-value H_0: Robust RMSEA >= 0.080	0.983

Standardized Root Mean Square Residual:

SRMR	0.100
------	-------

Parameter Estimates:

	Standard errors	Standard				
	Information	Observed				
	Observed information based on	Hessian				
<b>Latent Variables:</b>						
	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
Climate ==						
rBlst_1	1.000				0.854	0.540
Blst_4	1.242	0.322	3.856	0.000	1.061	0.861
Blst_6	1.343	0.347	3.865	0.000	1.147	0.868
Blst_2	1.267	0.368	3.441	0.001	1.082	0.689
Blst_3	0.935	0.289	3.232	0.001	0.799	0.625
Blst_5	1.247	0.319	3.909	0.000	1.065	0.896
CourseEval ==						
cEval_8	1.000				0.509	0.618
cEval_9	1.067	0.339	3.153	0.002	0.543	0.519
cEval_10	1.352	0.331	4.086	0.000	0.688	0.710
cEval_11	1.503	0.404	3.719	0.000	0.764	0.669
cEval_12	1.184	0.409	2.895	0.004	0.602	0.480
cEval_13	1.476	0.361	4.089	0.000	0.751	0.722
cEval_14	1.870	0.505	3.701	0.000	0.951	0.705
cEval_15	1.824	0.461	3.961	0.000	0.928	0.789
cEval_20	1.413	0.428	3.301	0.001	0.719	0.638
cEval_16	1.549	0.457	3.391	0.001	0.788	0.614
cEval_17	1.835	0.539	3.403	0.001	0.934	0.655
tBIPOC ==						
iBIPOC_pr	1.000				0.396	1.000
sBlack ==						
cmBlack	1.000				7.560	1.000
<b>Regressions:</b>						
	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
Climate ~						
CorsEvl (b)	-0.421	0.295	-1.428	0.153	-0.251	-0.251
tBIPOC (c_p1)	-0.369	0.339	-1.087	0.277	-0.171	-0.171
sBlack (c_p2)	-0.025	0.017	-1.466	0.143	-0.222	-0.222
CourseEval ~						
tBIPOC (a1)	0.353	0.204	1.732	0.083	0.275	0.275
sBlack (a2)	0.008	0.010	0.773	0.439	0.117	0.117
<b>Covariances:</b>						
	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
tBIPOC ~~						
sBlack	0.000				0.000	0.000
<b>Intercepts:</b>						
	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all

.rBlst_1	3.402	0.235	14.491	0.000	3.402	2.149
.Blst_4	2.346	0.181	12.992	0.000	2.346	1.903
.Blst_6	2.259	0.193	11.677	0.000	2.259	1.710
.Blst_2	2.672	0.231	11.583	0.000	2.672	1.701
.Blst_3	2.106	0.186	11.304	0.000	2.106	1.649
.Blst_5	2.000	0.173	11.534	0.000	2.000	1.682
.cEval_8	4.298	0.120	35.775	0.000	4.298	5.218
.cEval_9	3.809	0.152	24.983	0.000	3.809	3.644
.cEval_10	4.000	0.141	28.312	0.000	4.000	4.130
.cEval_11	3.404	0.167	20.413	0.000	3.404	2.978
.cEval_12	3.851	0.183	21.037	0.000	3.851	3.068
.cEval_13	3.830	0.152	25.267	0.000	3.830	3.686
.cEval_14	3.723	0.197	18.916	0.000	3.723	2.759
.cEval_15	3.255	0.172	18.962	0.000	3.255	2.766
.cEval_20	3.553	0.164	21.610	0.000	3.553	3.152
.cEval_16	3.128	0.187	16.713	0.000	3.128	2.438
.cEval_17	2.704	0.209	12.911	0.000	2.704	1.897
.iBIPOC_pr	0.330	0.059	5.590	0.000	0.330	0.832
.cmBlack	6.660	1.103	6.039	0.000	6.660	0.881

Variances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
.iBIPOC_pr	0.000				0.000	0.000
.cmBlack	0.000				0.000	0.000
.rBlst_1	1.776	0.388	4.579	0.000	1.776	0.709
.Blst_4	0.394	0.111	3.544	0.000	0.394	0.259
.Blst_6	0.430	0.126	3.425	0.001	0.430	0.246
.Blst_2	1.298	0.295	4.393	0.000	1.298	0.526
.Blst_3	0.994	0.217	4.577	0.000	0.994	0.609
.Blst_5	0.279	0.094	2.985	0.003	0.279	0.198
.cEval_8	0.420	0.098	4.295	0.000	0.420	0.618
.cEval_9	0.797	0.176	4.542	0.000	0.797	0.730
.cEval_10	0.465	0.119	3.920	0.000	0.465	0.495
.cEval_11	0.723	0.165	4.375	0.000	0.723	0.553
.cEval_12	1.212	0.260	4.666	0.000	1.212	0.770
.cEval_13	0.516	0.127	4.062	0.000	0.516	0.478
.cEval_14	0.916	0.221	4.139	0.000	0.916	0.503
.cEval_15	0.524	0.145	3.608	0.000	0.524	0.378
.cEval_20	0.754	0.178	4.229	0.000	0.754	0.593
.cEval_16	1.025	0.229	4.483	0.000	1.025	0.623
.cEval_17	1.161	0.274	4.230	0.000	1.161	0.571
.Climate	0.600	0.316	1.899	0.058	0.822	0.822
.CourseEval	0.236	0.109	2.171	0.030	0.910	0.910
tBIPOC	0.157	0.033	4.755	0.000	1.000	1.000
sBlack	57.161	11.791	4.848	0.000	1.000	1.000

R-Square:

Estimate

iBIPOC_pr	1.000
cmBlack	1.000
rBlst_1	0.291
Blst_4	0.741
Blst_6	0.754
Blst_2	0.474
Blst_3	0.391
Blst_5	0.802
cEval_8	0.382
cEval_9	0.270
cEval_10	0.505
cEval_11	0.447
cEval_12	0.230
cEval_13	0.522
cEval_14	0.497
cEval_15	0.622
cEval_20	0.407
cEval_16	0.377
cEval_17	0.429
Climate	0.178
CourseEval	0.090

Defined Parameters:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
indirect1	-0.149	0.129	-1.153	0.249	-0.069	-0.069
indirect2	-0.003	0.005	-0.690	0.490	-0.029	-0.029
contrast	-0.145	0.127	-1.144	0.253	-0.040	-0.040
total_indircts	-0.152	0.131	-1.160	0.246	-0.098	-0.098
total_c	-0.546	0.348	-1.570	0.116	-0.491	-0.491
direct1	-0.369	0.339	-1.087	0.277	-0.171	-0.171
direct2	-0.025	0.017	-1.466	0.143	-0.222	-0.222

Here's how I might write up the results of the overall fit.

**Hybrid model.** A test of the hypothesized structural model had less than acceptable fit to the data:  $\chi^2(171) = 590.70, p < 0.001, CFI = 0.783, RMSEA = 0.0114(90)$ .

Plotting what we did is helpful for a conceptual understanding and to check our work. Let's plan ahead to update the default result of *semPaths*::*semPlot* to reflect what we did. If we've made an error, this will show up in the product. Producing the map starts with understanding the coded (row/column) location of our variables.

Let's start with the *semPlot*::*semPaths* default result.

```
p <- semPlot::semPaths(struct1_fit, what = "col", whatLabels = "stand",
  sizeMan = 5, node.width = 1, edge.label.cex = 0.75, style = "lisrel",
  mar = c(5, 5, 5, 5))
```

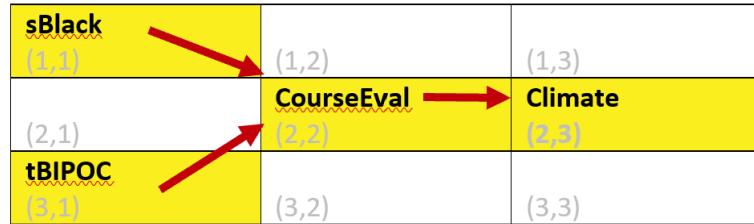
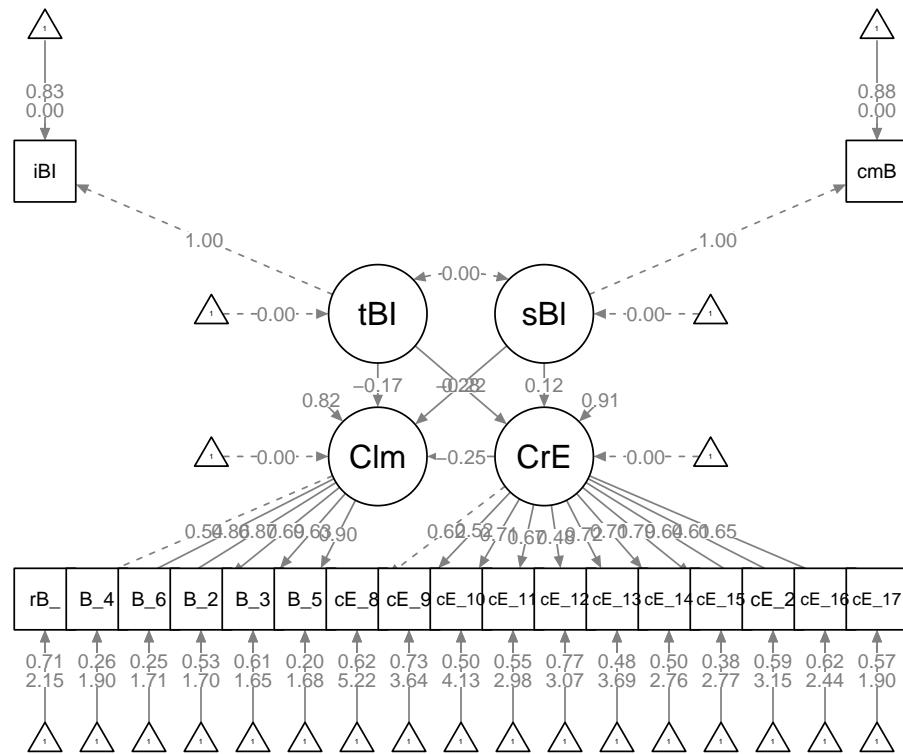
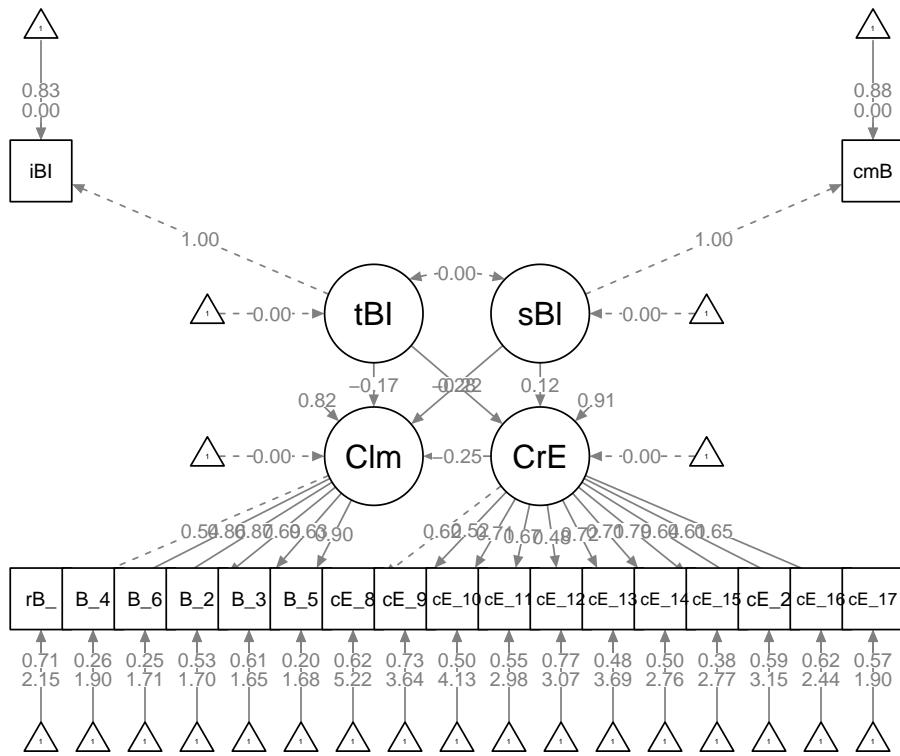


Figure 13.4: Image of the grid used for the semPlot



The code below included steps in creating a custom plot.

```
p <- semPlot::semPaths (struct1_fit, what = "col", whatLabels = "stand", sizeMan = 5, node.wid
```

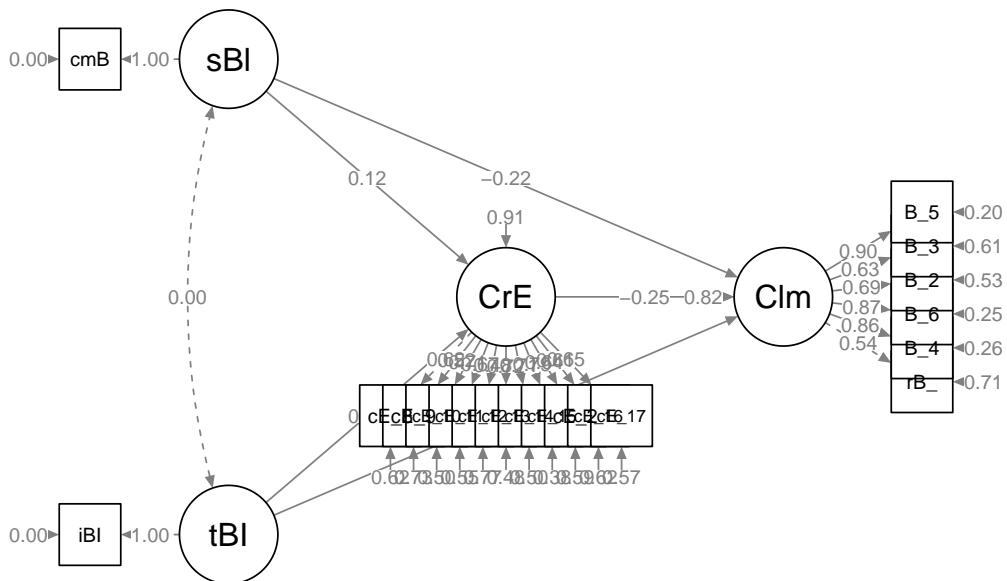


```

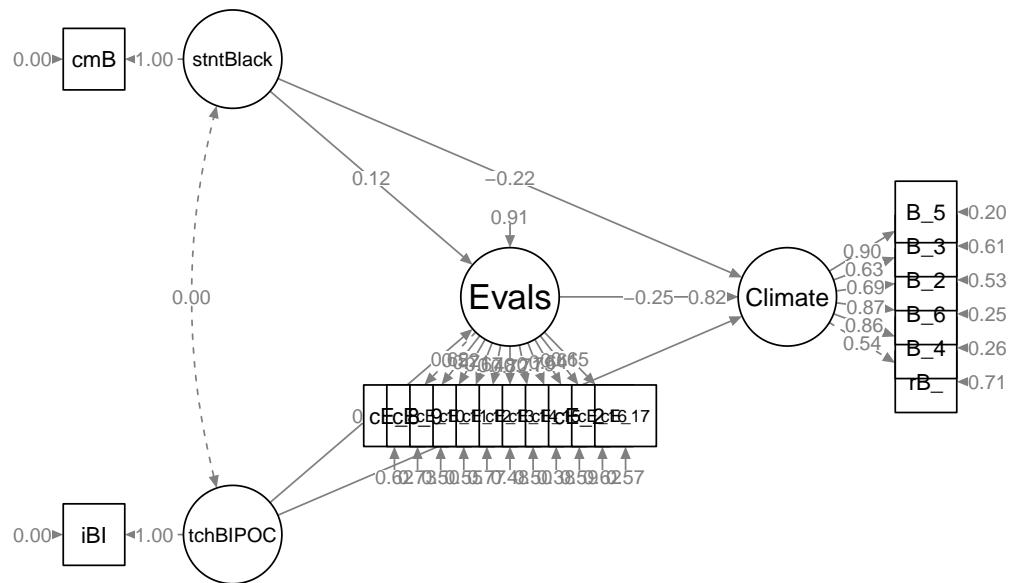
#the next two codes -- indicator_order and indicator_factor are paired together, they specify
indicator_order <- c("cmb",
                      "IBI",
                      "cE_8", "cE_9", "cE_10", "cE_11", "cE_12", "cE_13", "cE_14", "cE_15", "cE_2", "cE_16",
                      "rB_1", "B_4", "B_6", "B_2", "B_3", "B_5")
indicator_factor <- c("sBl",
                      "tBI",
                      "CrE", "CrE",
                      "Clm", "Clm", "Clm", "Clm", "Clm", "Clm")
#next set of code pushes the indicator variables away from the factor
indicator_push <- c(sBl = 1.5, #pushing the 1-item indicators only a little way away
                     tBI = 1.5,
                     CrE = 2.5, #pushing the multi-item indicators further away)
                     Clm = 2.5)
indicator_spread <- c(CrE = 2, #spreading the boxes away from each other
                     Clm = 2)

p2 <- semtools:::set_sem_layout(p,
                                  indicator_order = indicator_order,
                                  indicator_factor = indicator_factor,
                                  factor_layout = m_sem,
                                  factor_point_to = point_to,
                                  indicator_push = indicator_push,
                                  indicator_spread = indicator_spread)
plot(p2)

```



```
#changing node labels
p3 <- semtools::change_node_label(p2,
  c(sBl = "stntBlack",
    tBI = "tchBIPOC",
    Cre = "Evals",
    Clm = "Climate"),
  label.cex = 1.1)
plot(p3)
```



Model Coefficients Assessing M1 and M2 as Parallel Mediators Between X and Y

IV	M	DV	B for a and b paths	=	B	SE	p
tBIPOC -	Evals -	Climate (0.353)	X (-0.421)	=	-0.149	0.129	0.249
>	>						
cmBlack -	Evals -	Climate (0.008)	X (-0.421)	=	-0.003	0.005	0.490
>	>						

	B	SE	p
Total indirect effect	-0.152	0.131	0.246
Direct effect of tBIPOC on Climate (c'1 path)	-0.369	0.339	0.277
Direct effect of cmBlack on Climate (c'2 path)	-0.025	0.017	0.143

*Note.* X =definition; M1 = definition; M2 = definition; Y = definition. The significance of the indirect effects was calculated with bias-corrected confidence intervals (.95) bootstrap analysis.

### 13.6.3 APA Style Write-up of the Results

#### 13.6.3.1 Preliminary Analyses

We began by creating a dataset that included only the variables of interest. Our initial inspection of the data indicated that 70 attempted the survey. The proportion of missingness in the responses ranged from 0 to 90. Across the dataframe there was 26% of missingness across the cells. Approximately 60% of the cases had nonmissing data. The predominant pattern of missingness included individuals opening the survey without completing any of the items. Beyond that, our inspection of a missingness map indicated a haphazard pattern of missingness [Enders, 2010].

Using Parent's *available item analysis* [AIA; -Parent [2013]] as a guide, we deleted all cases where there was greater than 20% of data missing. We reinspected the missingness of the dataset with 47 cases. Across the dataframe there was less than 1% of missingness across the cells. Approximately 89% of the cases had nonmissing data.

Regarding the distributional characteristics of the data, skew and kurtosis values of the variables fell below the values of 3 (skew) and 8 to 20 (kurtosis) that Kline suggests are concerning [2016]. We evaluated multivariate normality with the Mahalanobis distance test. Specifically, we used the *outlier()* function in the *psych* package and included all continuous variables in the calculation. Our visual inspection of the Q-Q plot suggested that the plotted line strayed from the straight line as the quantiles increased. Additionally, we appended the Mahalanobis distance scores as a variable to the data. Analyzing this variable, we found that no exceed three standard deviations beyond the median. Thus, with no outliers, we assumed multivariate normality and proceeded with the data analysis. Means, standard deviations, and a correlation matrix are found in Table 1.

#### 13.6.3.2 Primary Analyses

A measurement model that allowed all latent variables to correlate had clearly unacceptable fit to the data:  $\chi^2(148) = 239.689, *p* < 0.001, CFI = 0.782, RMSEA = 0.115, 90$ . A test of the hypothesized structural model also had less than acceptable fit to the data:  $\chi^2(171) = 590.70, p < 0.001, CFI = 0.783, RMSEA = 0.0.114(90)$ . Similarly, there were no significant direct or indirect effects. Results are found in Table 2 and represented in Figure 1.

## 13.7 Practice Problems

In each of these lessons I provide suggestions for practice that allow you to select one or more problems that are graded in difficulty. With each of these options I encourage you to:

### 13.7.1 Problem #1: Download a fresh sample

As an open survey the [Rate-a-Recent-Course: A ReCentering Psych Stats Exercise](#) has the possibility of always being updated. There has been more data added to the survey since this lesson was rendered and/or I lectured it. If not, consider taking the survey and rating another course. Rerun the analyses with the updated data. Has it changed since the lesson was last lectured/updated?

### 13.7.2 Problem #2: Swap one or more of the variables

The Rate-a-Recent-Course survey is composed of a number of variables. Select a different constellation of variables for the hybrid analysis.

### 13.7.3 Problem #3: Try something entirely new.

Conduct a hybrid analysis using data for which you have permission and access (e.g., IRB approved data you have collected or from your lab; data you simulate from a published article; data from an open science repository; data from the simulations found at the end of this OER).

Regardless of your choic(es) complete all the elements listed in the grading rubric.

### 13.7.4 Grading Rubric

Using the lecture and workflow (chart) as a guide, please work through all the steps listed in the proposed assignment/grading rubric.

Assignment Component	Points Possible	Points Earned
1. Describe and draw the research model you will evaluate.	3	_____
2. Import the data and format the variables in the model.	5	_____
3. Analyze and manage missing data.	5	_____
4. Assess the distributional characteristics of the data.	5	_____
5. Conduct appropriate preliminary analyses ( <i>Ms</i> , <i>SDs</i> , <i>r</i> -matrix).	5	_____
6. Specify and evaluate a measurement model.	5	_____
7. Specify and evaluate a structural model; tweak as necessary.	5	_____
8. APA style results with table(s) and figure.	5	_____
9. Explanation to grader.	5	_____
<b>Totals</b>	<b>43</b>	_____

## 13.8 Homeworked Example

[Screencast Link](#)

For more information about the data used in this homeworked example, please refer to the description and codebook located at the end of the [introductory lesson in ReCentering Psych Stats](#). An .rds file which holds the data is located in the [Worked Examples](#) folder at the GitHub site the hosts the OER. The file name is *ReC.rds*.

The suggested practice problem for this chapter is to evaluate the measurement model that would precede the evaluation of a structural model. Next we respecify that model as the structural model. This model will (in all likelihood) be more parsimonious (i.e., have fewer paths) and have worse fit. The goal, though, is that our more parsimonious, structural model, is just as explanatory as the measurement model, where all factors were allowed to covary.

As we end the set of lessons on psychometric development and evaluation, and important part of this lesson is to start with very raw data and work through the scrubbing, scoring, and data diagnostics as we prepare it for the structural equation model that is the primary analysis

### Describe and draw the research model you will evaluate

It should have a minimum of three variables and could be one of the prior path-level models you already examined.

With a simple linear regression, I want to predict students' valuation (Valued, Y) of the statistics course from

- Traditional pedagogy (TradPed, X1)
- Socially responsive pedagogy (SRPed (X2))
- Centering status: 0 = precentered; 1 = recentered (Centering, X3)

X1 = Centering: explicit recentering (0 = precentered; 1 = recentered) X2 = TradPed: traditional pedagogy (continuously scaled with higher scores being more favorable) X3 = SRPed: socially responsive pedagogy (continuously scaled with higher scores being more favorable) Y = Valued: valued by me (continuously scaled with higher scores being more favorable)

I am hypothesizing that all three predictors will have a statistically significant, positive, effect on the outcome.

It helps me to make a quick sketch:

### Import the data and format the variables in the model.

```
raw <- readRDS("ReC.rds")
```

The multiple regression approach we are using does not allow dependency in the data. Therefore, we will include only those who took the psychometrics class (i.e., excluding responses for the ANOVA and multivariate courses).

```
raw <- (dplyr::filter(raw, Course == "Psychometrics"))
```

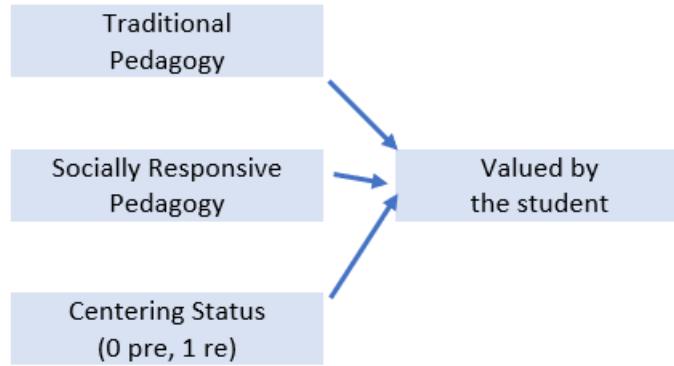


Figure 13.5: An image of the linear regression model for the homeworked example

Although this dataset is overall small, I will go ahead and make a babydf with the item-level variables.

```
babydf <- dplyr::select(raw, Centering, ClearResponsibilities, EffectiveAnswers,
  Feedback, ClearOrganization, ClearPresentation, ValObjectives, IncrUnderstanding,
  IncrInterest, InclusvClassrm, EquitableEval, MultPerspectives, DEIIntegration)
```

Let's check the structure of the variables:

```
str(babydf)
```

```
Classes 'data.table' and 'data.frame': 112 obs. of 13 variables:
 $ Centering          : Factor w/ 2 levels "Pre","Re": 2 2 2 2 2 2 2 2 2 2 ...
 $ ClearResponsibilities: int  3 5 5 4 4 3 4 4 5 5 ...
 $ EffectiveAnswers     : int  3 5 5 3 3 2 4 4 4 4 ...
 $ Feedback             : int  2 5 5 4 4 4 4 4 4 4 ...
 $ ClearOrganization    : int  2 5 5 3 3 2 4 4 4 4 ...
 $ ClearPresentation    : int  2 4 4 2 4 1 4 3 4 4 ...
 $ ValObjectives        : int  3 4 4 4 4 5 4 4 5 4 ...
 $ IncrUnderstanding    : int  4 4 4 2 4 2 5 4 3 4 ...
 $ IncrInterest          : int  1 4 4 2 4 1 5 3 4 2 ...
 $ InclusvClassrm        : int  3 5 5 4 5 3 5 5 5 4 ...
 $ EquitableEval         : int  3 5 5 4 4 2 4 5 4 5 ...
 $ MultPerspectives      : int  1 5 5 5 4 3 5 3 5 NA ...
 $ DEIIntegration         : int  2 5 5 5 5 5 5 5 5 4 ...
 - attr(*, ".internal.selfref")=<externalptr>
```

All of the item-level variables are integers (i.e., numerical). This is fine.

The centering variable will need to be dummy coded as 0/1:

```
babydf$CEN <- as.numeric(babydf$Centering)
babydf$CEN <- (babydf$CEN - 1)
str(babydf)
```

```
Classes 'data.table' and 'data.frame': 112 obs. of 14 variables:
 $ Centering          : Factor w/ 2 levels "Pre","Re": 2 2 2 2 2 2 2 2 2 ...
 $ ClearResponsibilities: int  3 5 5 4 4 3 4 4 5 5 ...
 $ EffectiveAnswers    : int  3 5 5 3 3 2 4 4 4 4 ...
 $ Feedback             : int  2 5 5 4 4 4 4 4 4 4 ...
 $ ClearOrganization    : int  2 5 5 3 3 2 4 4 4 4 ...
 $ ClearPresentation    : int  2 4 4 2 4 1 4 3 4 4 ...
 $ ValObjectives        : int  3 4 4 4 4 5 4 4 5 4 ...
 $ IncrUnderstanding    : int  4 4 4 2 4 2 5 4 3 4 ...
 $ IncrInterest          : int  1 4 4 2 4 1 5 3 4 2 ...
 $ InclusvClassrm       : int  3 5 5 4 5 3 5 5 5 4 ...
 $ EquitableEval         : int  3 5 5 4 4 2 4 5 4 5 ...
 $ MultPerspectives      : int  1 5 5 5 4 3 5 3 5 NA ...
 $ DEIIntegration        : int  2 5 5 5 5 5 5 5 5 4 ...
 $ CEN                  : num  1 1 1 1 1 1 1 1 1 1 ...
 - attr(*, ".internal.selfref")=<externalptr>
```

### Analyze and manage missing data.

Structural equation models lend themselves to managing missing data with Parent's [2013] *available information analysis* (AIA) approach. My approach is to:

- Create a dataframe that includes only the variables that will be used in the analysis.
- Delete all cases with greater than 20% missingness.
- If scale scores (or parcels) are used, calculate them if ~80% of the data for the calculation is present.
- Use the *full information maximum likelihood* (FIML) estimation procedure in *lavaan*; this allows item-level missingness.

```
cases1 <- nrow(babydf) #I produced this object for the sole purpose of feeding the number of
cases1
```

```
[1] 112
```

112 students completed at least some of the course evaluation.

The next code creates a variable that counts the number of cells with missing data.

```
library(tidyverse)
# Create a variable (n_miss) that counts the number missing
babydf$n_miss <- babydf %>%
```

```
dplyr::select(Centering:DEIintegration) %>%
  is.na %>%
  rowSums
```

The next code creates a variable that calculates the proportion of the data that is missing. Additionally, it sorts the data from highest to lowest proportion of missingness.

```
# Create a proportion missing by dividing n_miss by the total number
# of variables (21) Sort in order of descending frequency to get a
# sense of the missingness
babydf <- babydf %>%
  dplyr::mutate(prop_miss = (n_miss/13) * 100) %>%
  arrange(desc(n_miss))
```

From the code below we can see the average missingness as well as the range.

```
PrMiss1 <- psych::describe(babydf$prop_miss)
PrMiss1
```

	vars	n	mean	sd	median	trimmed	mad	min	max	range	skew	kurtosis	se
X1	1	112	1.24	3.66	0	0.17	0	0	15.38	15.38	2.94	7.69	0.35

We see that row (or case) level missingness ranged from 0 to 15%. This is great! Most students are completing the entire evaluation.

Finally, we can write code to provide the proportion of missingness across the entire data frame, and then the percent of cases/rows with nonmissing data.

```
CellsMissing1 <- formattable::percent(mean(is.na(babydf))) #percent missing across df
RowsMissing1 <- formattable::percent(mean(complete.cases(babydf))) #percent of rows with non-
CellsMissing1
```

```
[1] 1.00%
```

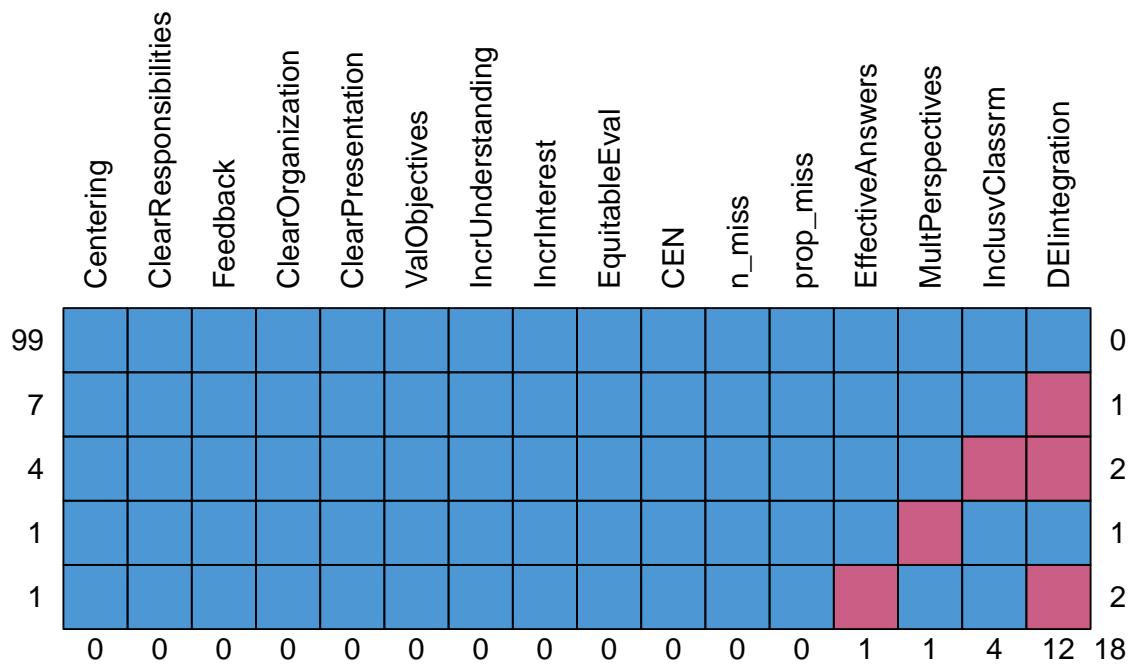
```
RowsMissing1
```

```
[1] 88.39%
```

Across the data frame, 1% of the data is missingness. Further, 88% of students have non-missing data.

Let's conduct an analysis of missingness with the *mice::md.pattern()* function.

```
missingness <- mice::md.pattern(babydf, plot = TRUE, rotate.names = TRUE)
```



missingness

	Centering	ClearResponsibilities	Feedback	ClearOrganization	ClearPresentation	
99	1		1		1	1
7	1		1	1		1
4	1		1		1	1
1	1		1		1	1
1	1		1		1	1
	0		0		0	0
Val	Objectives	IncrUnderstanding	IncrInterest	EquitableEval	CEN	n_miss
99	1		1		1	1
7	1		1	1	1	1
4	1		1	1	1	1
1	1		1		1	1

	1	1	1	1	1	1	1
	0	0	0	0	0	0	0
	prop_miss	EffectiveAnswers	MultPerspectives	InclusvClassrm	DEIintegration		
99	1	1	1	1	1	1	0
7	1	1	1	1	1	0	1
4	1	1	1	1	0	0	2
1	1	1	0	0	1	1	1
1	1	0	1	1	1	0	2
	0	1	1	4		12	18

Tentative write-up:

Our initial inspection of the data indicated that 112 attempted the course evaluation. Across cases, the proportion of missingness in the responses ranged from 0% to 15%. Across the dataframe there was 1% of missingness across the cells. Approximately 88% of the cases had nonmissing data. Our inspection of a missingness map indicated missingness on several of the items assessing socially responsive pedagogy (e.g., multiple perspectives, inclusive classroom, DEI integration). This could be because these items were added to the course evaluations after the study began. Because missingess was so low, we retained all cases and did not inspect further.

### Assess the distributional characteristics of the data.

```
psych::describe(babydf)
```

	vars	n	mean	sd	median	trimmed	mad	min	max	range
Centering*	1	112	1.42	0.50	1.0	1.40	0.00	1	2.00	1.00
ClearResponsibilities	2	112	4.62	0.60	5.0	4.72	0.00	3	5.00	2.00
EffectiveAnswers	3	111	4.52	0.69	5.0	4.64	0.00	2	5.00	3.00
Feedback	4	112	4.38	0.75	5.0	4.51	0.00	2	5.00	3.00
ClearOrganization	5	112	4.22	0.92	4.0	4.38	1.48	1	5.00	4.00
ClearPresentation	6	112	4.35	0.85	5.0	4.51	0.00	1	5.00	4.00
ValObjectives	7	112	4.53	0.57	5.0	4.58	0.00	3	5.00	2.00
IncrUnderstanding	8	112	4.38	0.76	4.5	4.50	0.74	1	5.00	4.00
IncrInterest	9	112	3.93	1.03	4.0	4.08	1.48	1	5.00	4.00
InclusvClassrm	10	108	4.63	0.65	5.0	4.76	0.00	2	5.00	3.00
EquitableEval	11	112	4.66	0.56	5.0	4.74	0.00	2	5.00	3.00
MultPerspectives	12	111	4.47	0.80	5.0	4.62	0.00	1	5.00	4.00
DEIIntegration	13	100	4.56	0.73	5.0	4.72	0.00	2	5.00	3.00
CEN	14	112	0.42	0.50	0.0	0.40	0.00	0	1.00	1.00
n_miss	15	112	0.16	0.48	0.0	0.02	0.00	0	2.00	2.00
prop_miss	16	112	1.24	3.66	0.0	0.17	0.00	0	15.38	15.38
			skew	kurtosis	se					
Centering*			0.32	-1.91	0.05					
ClearResponsibilities			-1.30	0.59	0.06					

EffectiveAnswers	-1.42	1.87	0.07
Feedback	-1.25	1.52	0.07
ClearOrganization	-1.35	1.78	0.09
ClearPresentation	-1.69	3.51	0.08
ValObjectives	-0.69	-0.58	0.05
IncrUnderstanding	-1.61	3.66	0.07
IncrInterest	-1.04	0.69	0.10
InclusvClassrm	-1.70	2.31	0.06
EquitableEval	-1.70	3.45	0.05
MultPerspectives	-1.56	2.50	0.08
DEIIntegration	-1.60	1.88	0.07
CEN	0.32	-1.91	0.05
n_miss	2.94	7.69	0.04
prop_miss	2.94	7.69	0.35

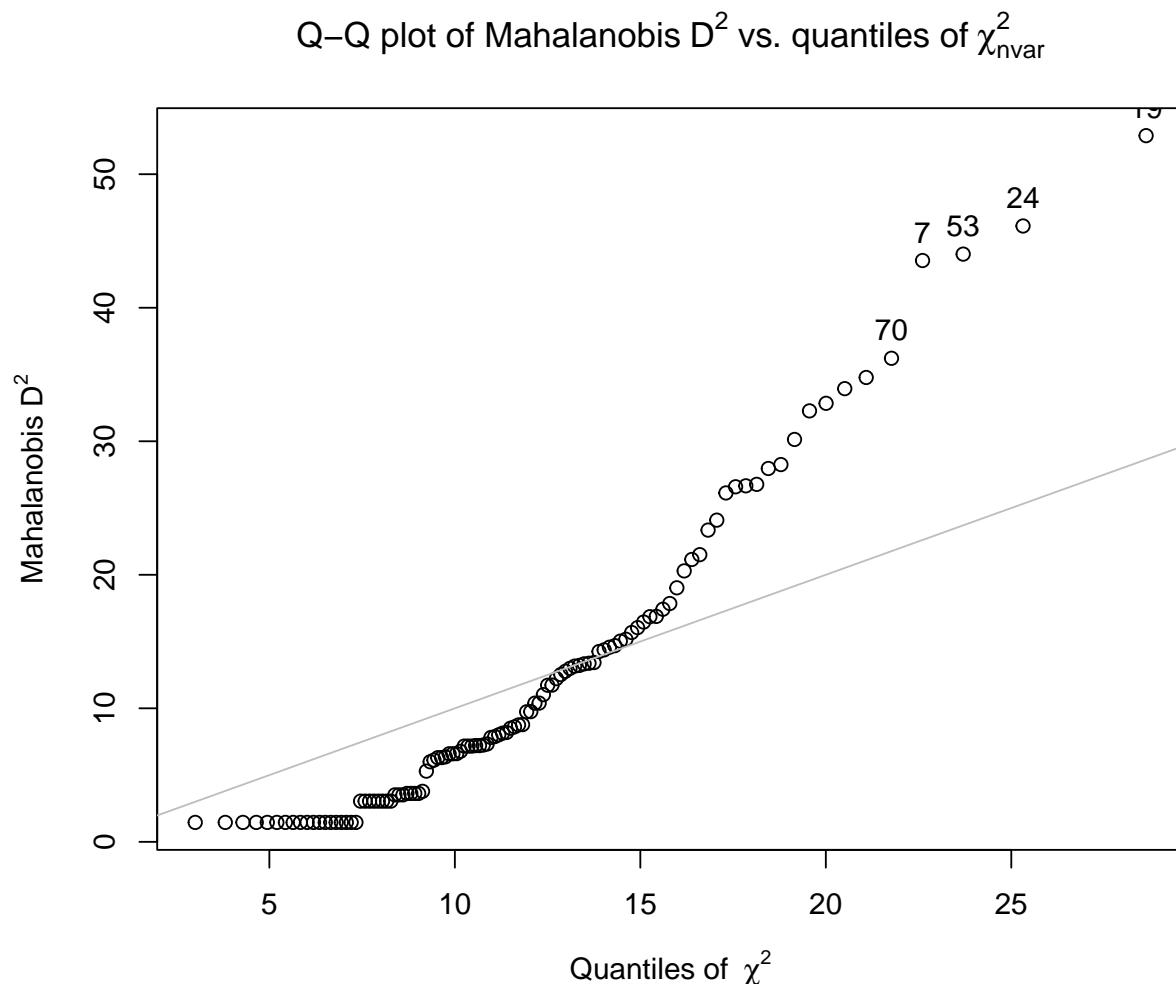
Regarding the distributional characteristics of the data, skew and kurtosis values of the variables fell below the values of 3 (skew) and 8 to 20 (kurtosis) that Kline suggests are concerning [2016].

Next I will conduct an outlier analysis with the Mahalanobis test. The dataset for this can only have continuously scaled variables. First I will create a Mahal\_df that excludes the Centering variable (and those other variables we created to assess missingness).

```
Mahal_df <- dplyr::select(babydf, ClearResponsibilities, EffectiveAnswers,
  Feedback, ClearOrganization, ClearPresentation, ValObjectives, IncrUnderstanding,
  IncrInterest, InclusvClassrm, EquitableEval, MultPerspectives, DEIIntegration)
```

We'll append a variable to the Mahal\_df that calculates the distance from the *centroid* of the data. To the degree that the data strays from the diagonal line (and particularly if there are numbered variables), we have outliers.

```
Mahal_df$Mahal <- psych::outlier(Mahal_df)
```



The code below appends a TRUE/FALSE variable to the data. Cases are TRUE if the Mahalanobis distance test is more than three standard deviations from the centroid.

```
library(dplyr)
# str(item_scores_df$Mahal)
Mahal_df$MOutlier <- if_else(Mahal_df$Mahal > (median(Mahal_df$Mahal) +
  (3 * sd(Mahal_df$Mahal))), TRUE, FALSE)

head(Mahal_df) #shows us the first 6 rows of the data so we can see the new variables (Mahal,
```

	ClearResponsibilities	EffectiveAnswers	Feedback	ClearOrganization
1:	4	4	4	3
2:	5	5	4	5
3:	5	NA	4	5
4:	5	5	4	4
5:	5	5	5	5
6:	5	4	4	4

```

ClearPresentation ValObjectives IncrUnderstanding IncrInterest
1:          4          4          4          3
2:          5          5          4          3
3:          5          5          5          4
4:          5          4          5          4
5:          5          4          5          5
6:          4          4          4          2
InclusvClassrm EquitableEval MultPerspectives DEIintegration      Mahal
1:        NA          4          3        NA 12.529051
2:        NA          5          4        NA 10.386495
3:          5          5          5        NA 6.607345
4:        NA          5          5        NA 7.815809
5:        NA          5          5        NA 7.250698
6:          4          5        NA        4 16.461642
MOutlier
1: FALSE
2: FALSE
3: FALSE
4: FALSE
5: FALSE
6: FALSE

```

Below, the code will count the number of outliers.

```

OutlierCount <- Mahal_df %>%
  count(MOutlier)
OutlierCount

```

```

MOutlier   n
1: FALSE 108
2: TRUE   4

```

We have only 4 outliers. Should we delete any? One common practice is to sort the data by the “Mahal” variable and look for “jumps.” If there is a consistent increase, then many researchers leave the data in.

Next we evaluated the distributional characteristics of the data. From a univariate perspective, skew and kurtosis values of the variables fell below the values of 3 (skew) and 8 to 20 (kurtosis) that Kline suggests are concerning [2016]. We evaluated multivariate normality with the Mahalanobis distance test. Specifically, we used the *outlier()* function in the *psych* package and included all continuous variables in the calculation. Our visual inspection of the Q-Q plot suggested that the plotted line strayed from the straight line as the quantiles increased. Additionally, we appended the Mahalanobis distance scores as a variable to the data. Analyzing this variable, we found that four cases exceeded three standard deviations beyond the median. We sorted the Mahalanobis values. Because there was a continuous increase with no “jumps” we retained all cases.

Conduct appropriate preliminary analyses (*Ms*, *SDs*, *r-matrix*).

### Internal consistency (alpha) coefficients

Although these are typically reported in the Method section in the description of the measures, we should calculate internal consistency coefficients for our three scales.

```
ValuedVars <- c("ValObjectives", "IncrUnderstanding", "IncrInterest")
TradPedVars <- c("ClearResponsibilities", "EffectiveAnswers", "Feedback",
  "ClearOrganization", "ClearPresentation")
SRPedVars <- c("InclusvClassrm", "EquitableEval", "MultPerspectives", "DEIntegration")

psych::alpha(babydf[, ..ValuedVars])
```

```
Reliability analysis
Call: psych::alpha(x = babydf[, ..ValuedVars])

  raw_alpha std.alpha G6(smc) average_r S/N   ase   mean     sd median_r
    0.78        0.8       0.74      0.57 3.9 0.03   4.3 0.67      0.54

  95% confidence boundaries
    lower alpha upper
Feldt      0.70  0.78  0.84
Duhachek  0.72  0.78  0.84

  Reliability if an item is dropped:
    raw_alpha std.alpha G6(smc) average_r S/N alpha se var.r
ValObjectives        0.80      0.82      0.69      0.69 4.5  0.036   NA
IncrUnderstanding     0.63      0.70      0.54      0.54 2.4  0.056   NA
IncrInterest         0.62      0.63      0.46      0.46 1.7  0.069   NA
    med.r
ValObjectives        0.69
IncrUnderstanding     0.54
IncrInterest          0.46

  Item statistics
    n raw.r std.r r.cor r.drop mean     sd
ValObjectives     112  0.73  0.79  0.61   0.55  4.5 0.57
IncrUnderstanding 112  0.86  0.85  0.76   0.69  4.4 0.76
IncrInterest      112  0.92  0.88  0.81   0.73  3.9 1.03

  Non missing response frequency for each item
    1   2   3   4   5 miss
ValObjectives     0.00 0.00 0.04 0.40 0.56    0
IncrUnderstanding 0.01 0.03 0.04 0.43 0.50    0
IncrInterest      0.04 0.07 0.13 0.45 0.31    0
```

```
# if this code throws an error for you rewrite deleting the two dots
# in front of the ValuedVars object
```

Alpha for the Valued-by-Me dimension is 0.80

```
psych::alpha(babydf[, ..TradPedVars])
```

Reliability analysis

```
Call: psych::alpha(x = babydf[, ..TradPedVars])
```

	raw_alpha	std.alpha	G6(smc)	average_r	S/N	ase	mean	sd	median_r
	0.88	0.89	0.88	0.62	8.1	0.017	4.4	0.63	0.64

95% confidence boundaries

lower alpha upper

Feldt	0.84	0.88	0.91
Duhachek	0.85	0.88	0.92

Reliability if an item is dropped:

	raw_alpha	std.alpha	G6(smc)	average_r	S/N	alpha	se	var.r	med.r
ClearResponsibilities	0.85	0.86	0.83	0.60	5.9	0.023	0.0095		0.61
EffectiveAnswers	0.83	0.84	0.81	0.57	5.3	0.025	0.0065		0.58
Feedback	0.88	0.90	0.88	0.68	8.7	0.017	0.0041		0.65
ClearOrganization	0.86	0.87	0.85	0.63	6.7	0.021	0.0137		0.60
ClearPresentation	0.85	0.86	0.84	0.61	6.3	0.023	0.0130		0.61

Item statistics

	n	raw.r	std.r	r.cor	r.drop	mean	sd
ClearResponsibilities	112	0.85	0.86	0.84	0.77	4.6	0.60
EffectiveAnswers	111	0.89	0.90	0.89	0.83	4.5	0.69
Feedback	112	0.73	0.74	0.62	0.59	4.4	0.75
ClearOrganization	112	0.84	0.82	0.76	0.72	4.2	0.92
ClearPresentation	112	0.85	0.84	0.79	0.75	4.3	0.85

Non missing response frequency for each item

	1	2	3	4	5	miss
ClearResponsibilities	0.00	0.00	0.06	0.26	0.68	0.00
EffectiveAnswers	0.00	0.02	0.05	0.32	0.61	0.01
Feedback	0.00	0.04	0.05	0.40	0.51	0.00
ClearOrganization	0.02	0.04	0.09	0.39	0.46	0.00

```
ClearPresentation    0.02 0.03 0.05 0.39 0.51 0.00
```

Alpha for Traditional Pedagogy dimension is 0.89

```
psych::alpha(babydf[, ..SRPedVars])
```

Reliability analysis

```
Call: psych::alpha(x = babydf[, ..SRPedVars])
```

	raw_alpha	std.alpha	G6(smc)	average_r	S/N	ase	mean	sd	median_r
	0.86	0.87	0.84	0.62	6.4	0.021	4.6	0.58	0.64

95% confidence boundaries

lower alpha upper

Feldt	0.81	0.86	0.9
Duhachek	0.82	0.86	0.9

Reliability if an item is dropped:

	raw_alpha	std.alpha	G6(smc)	average_r	S/N	alpha	se	var.r	med.r
InclusvClassrm	0.80	0.80	0.74	0.58	4.1	0.031	0.00522		
EquitableEval	0.84	0.85	0.79	0.65	5.6	0.026	0.00039		
MultPerspectives	0.82	0.82	0.77	0.60	4.6	0.030	0.00892		
DEIIntegration	0.83	0.84	0.78	0.63	5.2	0.027	0.00064		
InclusvClassrm	0.60								
EquitableEval	0.65								
MultPerspectives	0.64								
DEIIntegration	0.64								

Item statistics

	n	raw.r	std.r	r.cor	r.drop	mean	sd
InclusvClassrm	108	0.87	0.88	0.83	0.77	4.6	0.65
EquitableEval	112	0.79	0.81	0.72	0.67	4.7	0.56
MultPerspectives	111	0.88	0.86	0.79	0.73	4.5	0.80
DEIIntegration	100	0.84	0.83	0.75	0.69	4.6	0.73

Non missing response frequency for each item

	1	2	3	4	5	miss
InclusvClassrm	0.00	0.01	0.06	0.21	0.71	0.04
EquitableEval	0.00	0.01	0.02	0.28	0.70	0.00
MultPerspectives	0.01	0.01	0.11	0.25	0.62	0.01
DEIIntegration	0.00	0.02	0.08	0.22	0.68	0.11

Alpha for the SCR Pedagogy dimension is 0.87

### Means, standard deviations, and a correlation matrix

Means, standard deviations, and a correlation matrix are also commonly reported. Because three of our constructs are scales, we will need to calculate their means for cases that have met the minimum threshold for nonmissingness.

```
# calculate means for when a specified proportion of items are
# non-missing
babydf$Valued <- sjstats::mean_n(babydf[, ..ValuedVars], 0.66) #will create the mean for each
babydf$TradPed <- sjstats::mean_n(babydf[, ..TradPedVars], 0.66) #will create the mean for ea
babydf$SRPed <- sjstats::mean_n(babydf[, ..SRPedVars], 0.66) #will create the mean for each i
```

The *apaTables::cor.table* function creates the standard table that will include the means, standard deviations, and correlation matrix. For some reason my concatenated list function isn't working so I'm just creating a corr\_df and will conduct the analyses with those.

```
corr_df <- dplyr::select(babydf, Valued, TradPed, SRPed, CEN)
apaTables::apa.cor.table(corr_df)
```

### Means, standard deviations, and correlations with confidence intervals

Variable	M	SD	1	2	3
1. Valued	4.28	0.67			
2. TradPed	4.42	0.63	.76**		
			[.66, .83]		
3. SRPed	4.58	0.58	.56**	.76**	
			[.42, .68]	[.67, .83]	
4. CEN	0.42	0.50	-.19*	-.18	-.09
			[-.37, -.01]	[-.35, .00]	[-.27, .10]

Note. M and SD are used to represent mean and standard deviation, respectively. Values in square brackets indicate the 95% confidence interval.

The confidence interval is a plausible range of population correlations that could have caused the sample correlation (Cumming, 2014).

\* indicates  $p < .05$ . \*\* indicates  $p < .01$ .

#### 13.8.0.1 Write up of preliminary analyses

Our initial inspection of the data indicated that 112 attempted the course evaluation. Across cases, the proportion of missingness in the responses ranged from 0% to 15%.

Across the dataframe there was 1% of missingness across the cells. Approximately 88% of the cases had nonmissing data. The predominant pattern of missingness included individuals opening the survey without completing any of the items. Beyond that, our inspection of a missingness map indicated a missingness on several of the items assessing socially responsive pedagogy (e.g., multiple perspectives, inclusive classroom, DEI integration). This could be because these items were added to the course evaluations after the study began. Because missingess was so low, we retained all cases and did not inspect further.

Next we evaluated the distributional characteristics of the data. From a univariate perspective, skew and kurtosis values of the variables fell below the values of 3 (skew) and 8 to 20 (kurtosis) that Kline suggests are concerning [2016]. We evaluated multivariate normality with the Mahalanobis distance test. Specifically, we used the *outlier()* function in the *psych* package and included all continuous variables in the calculation. Our visual inspection of the Q-Q plot suggested that the plotted line strayed from the straight line as the quantiles increased. Additionally, we appended the Mahalanobis distance scores as a variable to the data. Analyzing this variable, we found that four cases exceeded three standard deviations beyond the median. We sorted the Mahalanobis values. Because there was a continuous increase with no “jumps” we retained all cases. Means, standard deviations, and a correlation matrix are found in Table 1.

### Specify and evaluate a *measurement* model.

A measurement model includes all of the variables to be included in the model in a structure that allows all the latent variables to freely correlate with each other. This model WILL HAVE the best fit of all models where there are fewer paths.

```
msmt_mod <- "
  ##measurement model
  CTR =~ CEN
  TrP =~ ClearResponsibilities + EffectiveAnswers + Feedback + ClearOrganization + Clear
  SRP =~ InclusvClassrm + EquitableEval + MultPerspectives + DEIintegration
  Val =~ ValObjectives + IncrUnderstanding + IncrInterest

  # Variance of the single item indicator
  CTR ~~ 0*CEN

  # Covariances
  CTR ~~ TrP
  CTR ~~ SRP
  CTR ~~ Val
  TrP ~~ SRP
  TrP ~~ Val
  SRP ~~ Val

  " "
```

```
set.seed(230916)
msmt_fit <- lavaan::cfa(msmt_mod, data = babydf, missing = "fiml")
lavaan::summary(msmt_fit, fit.measures = TRUE, standardized = TRUE)
```

lavaan 0.6.17 ended normally after 91 iterations

Estimator	ML
Optimization method	NLMINB
Number of model parameters	44
Number of observations	112
Number of missing patterns	5

Model Test User Model:

Test statistic	212.694
Degrees of freedom	60
P-value (Chi-square)	0.000

Model Test Baseline Model:

Test statistic	1040.629
Degrees of freedom	78
P-value	0.000

User Model versus Baseline Model:

Comparative Fit Index (CFI)	0.841
Tucker-Lewis Index (TLI)	0.794
Robust Comparative Fit Index (CFI)	0.831
Robust Tucker-Lewis Index (TLI)	0.780

Loglikelihood and Information Criteria:

Loglikelihood user model (H0)	-1123.976
Loglikelihood unrestricted model (H1)	-1017.629
Akaike (AIC)	2335.952
Bayesian (BIC)	2455.565
Sample-size adjusted Bayesian (SABIC)	2316.510

Root Mean Square Error of Approximation:

RMSEA	0.151
90 Percent confidence interval - lower	0.129
90 Percent confidence interval - upper	0.173

P-value H_0: RMSEA <= 0.050	0.000
P-value H_0: RMSEA >= 0.080	1.000
Robust RMSEA	0.158
90 Percent confidence interval - lower	0.136
90 Percent confidence interval - upper	0.181
P-value H_0: Robust RMSEA <= 0.050	0.000
P-value H_0: Robust RMSEA >= 0.080	1.000

Standardized Root Mean Square Residual:

SRMR	0.071
------	-------

Parameter Estimates:

Standard errors	Standard
Information	Observed
Observed information based on	Hessian

Latent Variables:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
CTR =~						
CEN	1.000				0.494	1.000
TrP =~						
ClearRspnsblts	1.000				0.499	0.830
EffectivAnswnrs	1.222	0.101	12.109	0.000	0.610	0.895
Feedback	0.981	0.130	7.558	0.000	0.490	0.655
ClearOrganiztn	1.375	0.151	9.126	0.000	0.686	0.751
ClearPresenttn	1.389	0.136	10.205	0.000	0.693	0.823
SRP =~						
InclusvClassrm	1.000				0.532	0.824
EquitableEval	0.839	0.094	8.929	0.000	0.446	0.797
MultPerspectvs	1.186	0.128	9.283	0.000	0.631	0.798
DEIintegrtion	0.975	0.125	7.811	0.000	0.519	0.711
Val =~						
ValObjectives	1.000				0.379	0.670
IncrUndrstndng	1.571	0.232	6.781	0.000	0.596	0.785
IncrInterest	2.246	0.317	7.078	0.000	0.852	0.832

Covariances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
CTR ~~						
.CEN	0.000				0.000	NaN
TrP	-0.052	0.025	-2.080	0.038	-0.212	-0.212
SRP	-0.030	0.027	-1.105	0.269	-0.114	-0.114
Val	-0.040	0.020	-1.968	0.049	-0.214	-0.214
TrP ~~						
SRP	0.231	0.041	5.684	0.000	0.871	0.871

Val	0.164	0.033	4.948	0.000	0.868	0.868
SRP ~~						
Val	0.139	0.031	4.411	0.000	0.687	0.687

## Intercepts:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
.CEN	0.420	0.047	8.999	0.000	0.420	0.850
.ClearRspnsblts	4.616	0.057	81.248	0.000	4.616	7.677
.EffectivAnswrs	4.526	0.064	70.214	0.000	4.526	6.643
.Feedback	4.384	0.071	62.112	0.000	4.384	5.869
.ClearOrganiztn	4.223	0.086	48.938	0.000	4.223	4.624
.ClearPresenttn	4.348	0.080	54.647	0.000	4.348	5.164
.InclusvClassrm	4.631	0.062	75.290	0.000	4.631	7.170
.EquitableEval	4.661	0.053	88.099	0.000	4.661	8.325
.MultPerspectvs	4.466	0.075	59.682	0.000	4.466	5.651
.DEIintegration	4.544	0.071	63.735	0.000	4.544	6.228
.ValObjectives	4.527	0.054	84.595	0.000	4.527	7.993
.IncrUndrstndng	4.384	0.072	61.141	0.000	4.384	5.777
.IncrInterest	3.929	0.097	40.604	0.000	3.929	3.837

## Variances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
.CEN	0.000				0.000	0.000
.ClearRspnsblts	0.113	0.019	6.057	0.000	0.113	0.311
.EffectivAnswrs	0.093	0.018	5.090	0.000	0.093	0.200
.Feedback	0.318	0.045	7.057	0.000	0.318	0.571
.ClearOrganiztn	0.363	0.054	6.770	0.000	0.363	0.435
.ClearPresenttn	0.228	0.037	6.233	0.000	0.228	0.322
.InclusvClassrm	0.134	0.025	5.265	0.000	0.134	0.322
.EquitableEval	0.115	0.020	5.612	0.000	0.115	0.365
.MultPerspectvs	0.227	0.039	5.783	0.000	0.227	0.363
.DEIintegration	0.263	0.044	5.956	0.000	0.263	0.494
.ValObjectives	0.177	0.027	6.461	0.000	0.177	0.551
.IncrUndrstndng	0.221	0.040	5.487	0.000	0.221	0.383
.IncrInterest	0.322	0.071	4.549	0.000	0.322	0.307
CTR	0.244	0.033	7.483	0.000	1.000	1.000
TrP	0.249	0.047	5.276	0.000	1.000	1.000
SRP	0.283	0.056	5.033	0.000	1.000	1.000
Val	0.144	0.038	3.755	0.000	1.000	1.000

Below is script that will export the global fit indices (via *tidySEM::table\_fit*) and the parameter estimates (e.g., factor loadings, structural regression weights, and parameters we requested such as the indirect effect) to .csv files that you can manipulate outside of R.

```
# global fit indices
msmt_globalfit <- tidySEM::table_fit(msmt_fit)
```

Registered S3 method overwritten by 'tidySEM':

```

method      from
predict.MxModel OpenMx

msmt_fit_pEsts <- tidySEM:::table_results(msmt_fit, digits = 3, columns = NULL)
write.csv(msmt_globalfit, file = "msmt_globalfit.csv")
# the code below writes the parameter estimates into a .csv file
write.csv(msmt_fit_pEsts, file = "msmt_fit_pEsts.csv")

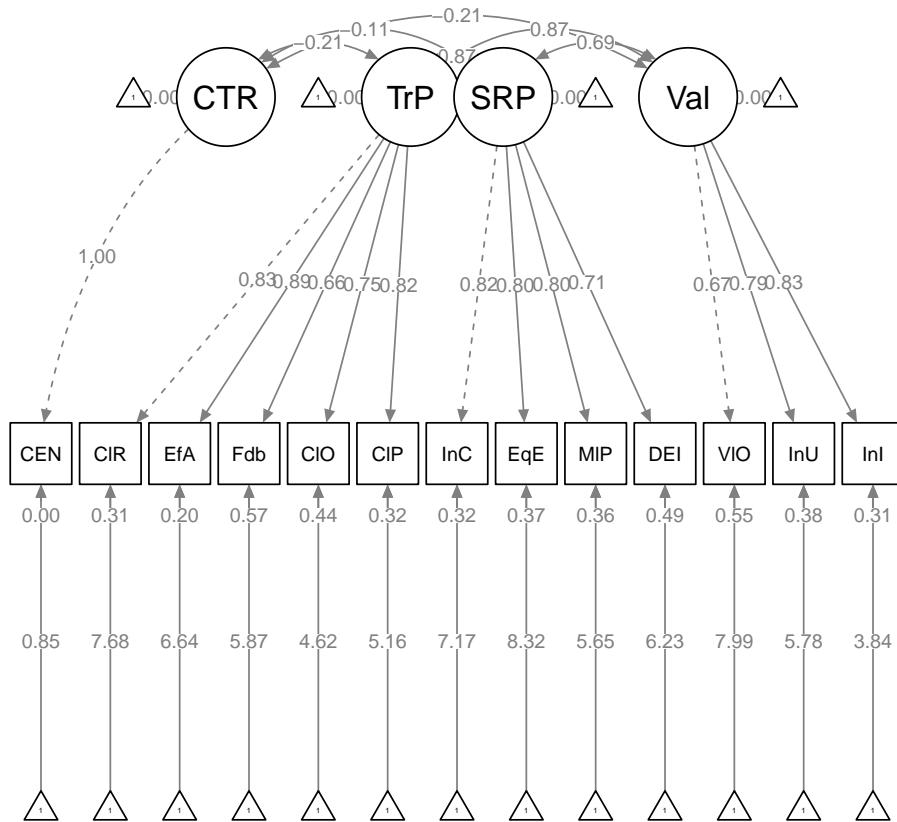
```

Let's plot what we did>

```

msmt_m1 <- semPlot::semPaths(msmt_fit, what = "col", whatLabels = "stand",
  sizeMan = 5, node.width = 1, edge.label.cex = 0.75, style = "lisrel",
  mar = c(5, 5, 5, 5))

```



With the exception of the SRMR, the fit of the measurement model was poor:  $\chi^2(60) = 212.69, *p* < 0.001, CFI = .841, RMSEA = 0.151, CI90(0.129, 0.173), SRMR = 0.071$ , SRMR = 0.071.

If this project were for actual publication, I would take this a step further and respecify the model with parcels. Why? Because a parcelled model (which approaches a *just-identified model* will usually have better global fit). I address this topic more completely in [ReCentering Psych Stats: Multivariate Modeling](#).

### Specify and evaluate a *structural* model

As a reminder, I am hypothesizing a multiple regression predicting value to the student from traditional pedagogy, socially responsive pedagogy, and centering.

X1 = Centering: explicit recentering (0 = precentered; 1 = recentered) X2 = TradPed: traditional pedagogy (continuously scaled with higher scores being more favorable) X3 = SRPed: socially responsive pedagogy (continuously scaled with higher scores being more favorable) Y = Valued: valued by me (continuously scaled with higher scores being more favorable)

This structural model will have fewer paths than its predecessor, the measurement model. The `lavaan::sem` specification looks quite similar to the measurement model. Differences are:

- The map of equations predicts Valued by centering, traditional pedagogy, and socially responsive pedagogy.
- The `lavaan::sem` code includes the statement “`auto.cov.lv.x=FALSE`”; this prevents lavaan from allowing the latent variables to correlate with each other.

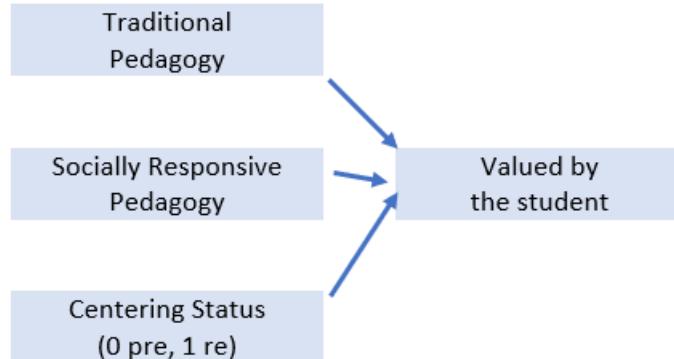


Figure 13.6: An image of the parallel mediation model for the homeworked example

```

ReC_struct_mod1 <- "
  #measurement model
  CTR =~ CEN #this is a single item indicator, I had to add code below to set the variance
  TrP =~ ClearResponsibilities + EffectiveAnswers + Feedback + ClearOrganization + ClearExpectations
  SRP =~ InclusivClassrm + EquitableEval + MultPerspectives + DEIIntegration
  Val =~ ValObjectives + IncrUnderstanding + IncrInterest

  # Variance of the single item indicator
  CTR ~~ 0*CEN
  
```

```

#structural model
Val ~ CTR + TrP + SRP

"
set.seed(230916) #needed for reproducibility
ReC_struct_fit1 <- lavaan::sem(ReC_struct_mod1, data = babydf, missing = "fiml",
  fixed.x = FALSE, auto.cov.lv.x = FALSE)
lavaan::summary(ReC_struct_fit1, fit.measures = TRUE, standardized = TRUE,
  rsq = TRUE)

```

lavaan 0.6.17 ended normally after 87 iterations

Estimator	ML
Optimization method	NLMINB
Number of model parameters	41
Number of observations	112
Number of missing patterns	5

Model Test User Model:

Test statistic	314.627
Degrees of freedom	63
P-value (Chi-square)	0.000

Model Test Baseline Model:

Test statistic	1040.629
Degrees of freedom	78
P-value	0.000

User Model versus Baseline Model:

Comparative Fit Index (CFI)	0.739
Tucker-Lewis Index (TLI)	0.676
Robust Comparative Fit Index (CFI)	0.734
Robust Tucker-Lewis Index (TLI)	0.671

Loglikelihood and Information Criteria:

Loglikelihood user model (H0)	-1174.942
Loglikelihood unrestricted model (H1)	-1017.629
Akaike (AIC)	2431.885
Bayesian (BIC)	2543.343
Sample-size adjusted Bayesian (SABIC)	2413.769

Root Mean Square Error of Approximation:

RMSEA	0.189
90 Percent confidence interval - lower	0.168
90 Percent confidence interval - upper	0.210
P-value H_0: RMSEA <= 0.050	0.000
P-value H_0: RMSEA >= 0.080	1.000
Robust RMSEA	0.194
90 Percent confidence interval - lower	0.173
90 Percent confidence interval - upper	0.215
P-value H_0: Robust RMSEA <= 0.050	0.000
P-value H_0: Robust RMSEA >= 0.080	1.000

Standardized Root Mean Square Residual:

SRMR	0.289
------	-------

Parameter Estimates:

Standard errors	Standard
Information	Observed
Observed information based on	Hessian

Latent Variables:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
CTR =~						
CEN	1.000				0.494	1.000
TrP =~						
ClearRspnsblts	1.000				0.496	0.824
EffectivAnswrs	1.259	0.102	12.299	0.000	0.624	0.916
Feedback	0.939	0.133	7.048	0.000	0.465	0.623
ClearOrganiztn	1.402	0.155	9.041	0.000	0.695	0.761
ClearPresenttn	1.377	0.141	9.760	0.000	0.682	0.810
SRP =~						
InclusvClassrm	1.000				0.554	0.857
EquitableEval	0.732	0.090	8.166	0.000	0.406	0.725
MultPerspectvs	1.132	0.131	8.623	0.000	0.628	0.794
DEIIntegration	1.012	0.119	8.504	0.000	0.561	0.771
Val =~						
ValObjectives	1.000				0.369	0.659
IncrUndrstndng	1.582	0.234	6.772	0.000	0.584	0.781
IncrInterest	2.261	0.319	7.078	0.000	0.835	0.828

Regressions:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
Val ~						

CTR	-0.039	0.054	-0.716	0.474	-0.052	-0.052
TrP	0.638	0.125	5.102	0.000	0.856	0.856
SRP	0.016	0.080	0.196	0.844	0.024	0.024

## Covariances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
CTR ~~						
.CEN	0.000				0.000	NaN

## Intercepts:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
.CEN	0.420	0.047	8.999	0.000	0.420	0.850
.ClearRspnsblts	4.616	0.057	81.248	0.000	4.616	7.677
.EffectivAnswrs	4.526	0.064	70.232	0.000	4.526	6.644
.Feedback	4.384	0.071	62.112	0.000	4.384	5.869
.ClearOrganiztn	4.223	0.086	48.938	0.000	4.223	4.624
.ClearPresenttn	4.348	0.080	54.647	0.000	4.348	5.164
.InclusvClassrm	4.628	0.062	75.037	0.000	4.628	7.151
.EquitableEval	4.661	0.053	88.099	0.000	4.661	8.325
.MultPerspectvs	4.465	0.075	59.618	0.000	4.465	5.645
.DEIintegration	4.540	0.071	63.975	0.000	4.540	6.235
.ValObjectives	4.527	0.053	85.399	0.000	4.527	8.069
.IncrUndrstndng	4.384	0.071	61.956	0.000	4.384	5.854
.IncrInterest	3.929	0.095	41.213	0.000	3.929	3.894

## Variances:

	Estimate	Std.Err	z-value	P(> z )	Std.lv	Std.all
.CEN	0.000				0.000	0.000
.ClearRspnsblts	0.116	0.019	6.047	0.000	0.116	0.321
.EffectivAnswrs	0.075	0.018	4.054	0.000	0.075	0.161
.Feedback	0.341	0.048	7.094	0.000	0.341	0.612
.ClearOrganiztn	0.352	0.054	6.481	0.000	0.352	0.422
.ClearPresenttn	0.244	0.040	6.071	0.000	0.244	0.343
.InclusvClassrm	0.111	0.027	4.060	0.000	0.111	0.266
.EquitableEval	0.149	0.024	6.157	0.000	0.149	0.475
.MultPerspectvs	0.232	0.045	5.176	0.000	0.232	0.370
.DEIintegration	0.215	0.039	5.498	0.000	0.215	0.406
.ValObjectives	0.178	0.027	6.494	0.000	0.178	0.566
.IncrUndrstndng	0.219	0.040	5.473	0.000	0.219	0.391
.IncrInterest	0.320	0.070	4.552	0.000	0.320	0.315
CTR	0.244	0.033	7.483	0.000	1.000	1.000
TrP	0.246	0.047	5.210	0.000	1.000	1.000
SRP	0.307	0.059	5.187	0.000	1.000	1.000
.Val	0.036	0.013	2.802	0.005	0.264	0.264

## R-Square:

	Estimate
CEN	1.000

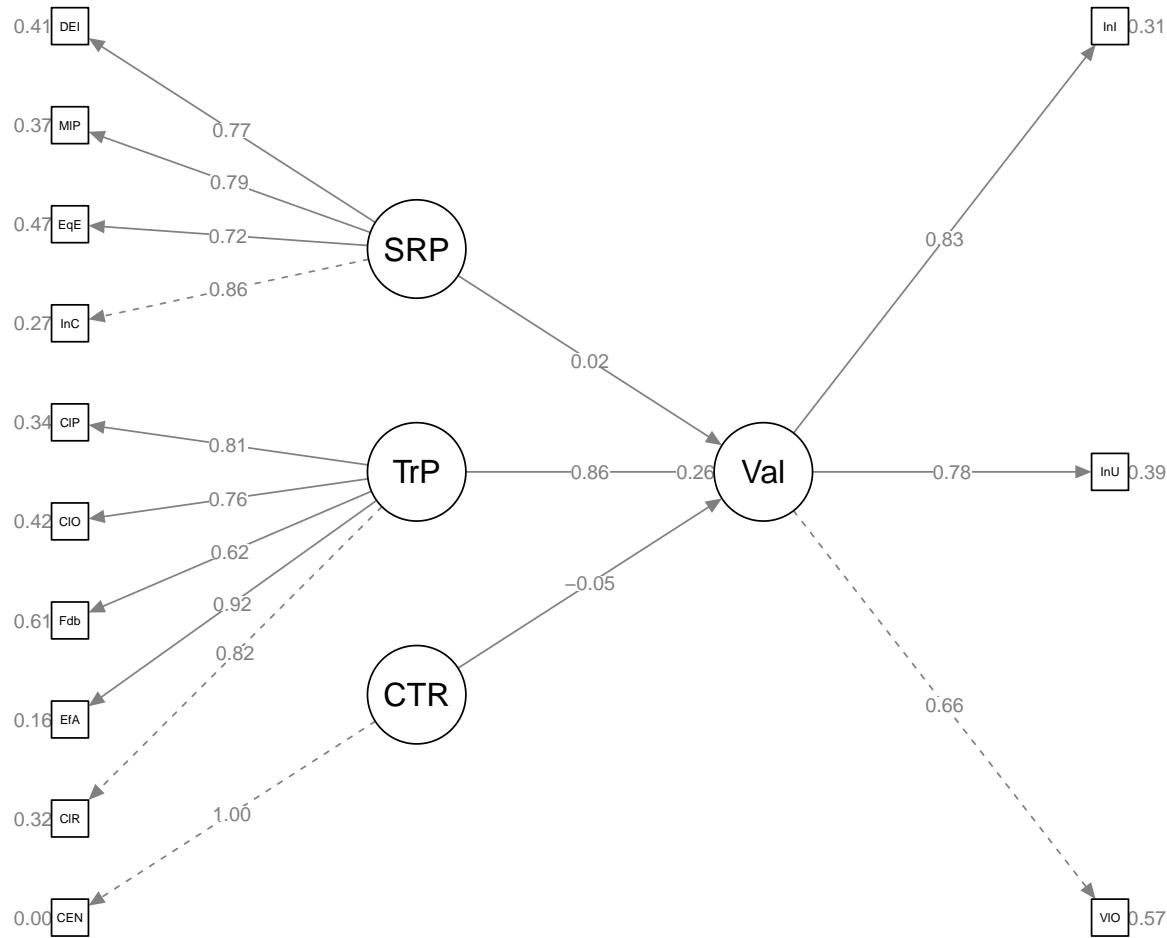
ClearRspnsblts	0.679
EffectivAnswrs	0.839
Feedback	0.388
ClearOrganiztn	0.578
ClearPresenttn	0.657
InclusvClassrm	0.734
EquitableEval	0.525
MultPerspectvs	0.630
DEIintegration	0.594
ValObjectives	0.434
IncrUndrstndng	0.609
IncrInterest	0.685
Val	0.736

Below is script that will export the global fit indices (via *tidySEM::table\_fit*) and the parameter estimates (e.g., factor loadings, structural regression weights, and parameters we requested such as the indirect effect) to .csv files that you can manipulate outside of R.

```
# global fit indices
ReC_globalfit1 <- tidySEM::table_fit(ReC_struct_fit1)
ReC_struct_pEsts <- tidySEM::table_results(ReC_struct_fit1, digits = 3,
  columns = NULL)
write.csv(ReC_globalfit1, file = "ReC_globalfit1.csv")
# the code below writes the parameter estimates into a .csv file
write.csv(ReC_struct_pEsts, file = "ReC_struct_pEsts1.csv")
```

Let's work up a figure. Luckily, the *layout* and *rotate* commands within within *semPlot:semPlot* make it easy to produce a reasonable figure.

```
plot_ReC_struct1 <- semPlot::semPaths(ReC_struct_fit1, what = "col", whatLabels = "stand",
  sizeMan = 3, node.width = 1, edge.label.cex = 0.75, style = "lisrel",
  mar = c(2, 2, 2, 2), structural = FALSE, curve = FALSE, intercepts = FALSE,
  layout = "tree", rotation = 2)
```



If we table the results, here's what we have:

**Table 2**

Model Coefficients Assessing the Effect of Perceived Value from Recentering, Traditional Pedagogy, and Socially Responsive Pedagogy

Predictor	B	SE <sub>B</sub>	p	β
Centering	-0.039	0.054	0.474	-0.052
Traditional Pedagogy	0.638	0.125	< 0.001	1.218
Socially Responsive Pedagogy	-0.016	0.080	0.844	0.024

Our structural model was a multiple regression, predicting perceived value to the student directly from centering, traditional pedagogy, and socially responsive pedagogy. Results of the global fit indices all fell below the thresholds of acceptability ( $\chi^2(63) = 314.63, p < 0.001, CFI = 0.739, RMSEA = 0.189, 90CI[0.168, 0.210], SRMR = 0.289$ ). As shown in Table 2, only traditional pedagogy was a significant predictor of value to the student.

### 13.8.1 APA style results with table(s) and figure.

**Preliminary Analyses** >Our initial inspection of the data indicated that 112 attempted the course evaluation. Across cases, the proportion of missingness in the responses ranged from 0% to 15%. Across the dataframe there was 1% of missingness across the cells. Approximately 88% of the cases had nonmissing data. The predominant pattern of missingness included individuals opening the survey without completing any of the items. Beyond that, our inspection of a missingness map indicated a missingness on several of the items assessing socially responsive pedagogy (e.g., multiple perspectives, inclusive classroom, DEI integration). This could be because these items were added to the course evaluations after the study began. Because missingess was so low, we retained all cases and did not inspect further.

Next we evaluated the distributional characteristics of the data. From a univariate perspective, skew and kurtosis values of the variables fell below the values of 3 (skew) and 8 to 20 (kurtosis) that Kline suggests are concerning [2016]. We evaluated multivariate normality with the Mahalanobis distance test. Specifically, we used the *outlier()* function in the *psych* package and included all continuous variables in the calculation. Our visual inspection of the Q-Q plot suggested that the plotted line strayed from the straight line as the quantiles increased. Additionally, we appended the Mahalanobis distance scores as a variable to the data. Analyzing this variable, we found that four cases exceeded three standard deviations beyond the median. We sorted the Mahalanobis values. Because there was a continuous increase with no “jumps” we retained all cases.

### Primary Analyses

We used structural equation modeling to predict course value to the student (Value) from traditional pedagogy (TradPed), socially responsive pedagogy (SRPed), and centering status (Centering, 0 = precentered, 1 = recentered).

We began by specifying a measurement model that permitted all latent variables to covary. Fit of this model was poor:  $\chi^2(60) = 212.69, *p* < 0.001, CFI = .841, RMSEA = 0.151, CI90[0.129, 0.173], SRMR = 0.071$ . Our structural model was a multiple regression, predicting perceived value to the student directly from centering, traditional pedagogy, and socially responsive pedagogy. Results of the global fit indices for the structural model were worse than the measurement model:  $(\chi^2(63) = 314.63, p < 0.001, CFI = 0.739, RMSEA = 0.189, 90CI[0.168, 0.210], SRMR = 0.289)$ . As shown in Table 2, only traditional pedagogy was a significant predictor of the course’s value to the student.



# Chapter 14

## Additional Simulations

### [Screencast Link](#)

In each of the suggestions for practice (i.e., “homework”), I suggest that an option for a middle-level degree of difficulty is to use data from another set of simulated data. One option is to use simulations from other lessons. Especially in the early chapters, it is difficult to know what data from later chapters might be appropriate for such analyses. Because of this challenge, I am providing several simulations that are not used in any of the lessons that could be used as practice for *all* of the lessons that involve psychometric evaluation. In this way, you would follow the construction of a scale through the standard process.

There are a few caveats to this process:

- In each of the simulations, I have added the code, “`eval = FALSE`” in the top of the chunk. This prevents the chunk from executing when the OER is *building*. If you copy the script you should have no problems (but if you try to knit an .rmd file and wonder why no data is being produced, this is why).
- Each of the psychometric evaluations included a number of additional measures used to assess convergent, discriminant, concurrent, predictive, and incremental validity. While I would have liked to have included them all,
  - I could not always locate source articles, and
  - Adding too many scales caused my simulation code to fail.
  - Thus, each simulation includes at least three additional measures that can be used for evaluating validity as it relates to measures within the nomological net (e.g., convergent, discriminant, concurrent, predictive, incremental).
- There are two simulations with each scale.
  - The first simulation provides the focal scale and the additional instruments.
  - The second simulation creates “A” and “B” data for the focal scale. When you are practicing test-retest reliability, you can *pretend* that A and B are from the same respondents. Later, when you are practicing invariance testing you can *pretend* that the data are from two different groups.
- At least one of these scales has five subscales. In certain analyses (e.g., item analysis) this may become unwieldy. In such cases you may wish to select three of those subscales and run the

analyses with just those. After all, “it’s just practice.” I do recommend that you complete the majority of practice assignments with a minimum of three subscales.

- Each of the simulations includes a random seed. If you and homework partners would like to work on the same simulation, you could gently increase your challenge by setting different random seeds and comparing the subtle differences in the solutions.
- I recommend that you retrieve the original psychometric article so that you can compare your analytic choices and results with those completed in the article. At least two of the articles are available as preprints or in open-source journals.

## 14.1 iBelong Scale

The iBelong Scale [Lee and Neville, 2024] is a measure of racial-ethnic-cultural (REC) belonging for BIPOC people. The scale includes 25 items with responses rated on a 6-point Likert scale ranging from 1 (*strongly disagree*) to 6 (*strongly agree*). At the start of the survey, participants are asked to self-identify their REC identity (i.e., “In terms of my racial-ethnic-cultural [REC] identity I consider myself to be:\_\_\_\_\_”). Higher scores indicate a stronger sense of REC belonging. Lee and Neville [2024] reported that a five-factor solution resulted in the best fit.

Beneath each of the five factors are a list of the items; variable names follow in parentheses.

- Authenticity:
  - In general, I do not have to change my behaviors when I am with REC group members (Auth1)
  - I am free to be myself with members of my REC group (Auth2)
  - In general, I do not have to hide parts of who I am with members of my REC group (Auth3)
  - Members of my REC group see me for who I am (Auth4)
  - I feel accepted by my REC group for who I am (Auth5)
- Connection:
  - I have similar experiences to others in my REC group (Conn1)
  - Others in my REC group see the world in a similar way as me (Conn2)
  - My success is connected to the success of members of my REC group (Conn3)
  - I have many things in common with people from my REC group (Conn4)
  - I feel connected to my REC group because of our shared experiences (Conn5)
- Home
  - I feel close to other members of my REC group (Home1)
  - I have shared interests with others in my REC group (Home2)
  - I have shared values with others in my REC group (Home3)
  - In general, I am at ease when I am with people from my REC group (Home4)
  - In general, I feel “at home” when I am with members of my REC group (Home5)
- REC Thriving
  - I am happy that I am a member of my REC group (Thriv1)
  - I am proud to be a member of my REC group (Thriv2)

- I like learning about my REC group’s history (Thriv3)
- I value my REC group (Thriv4)
- I consider it an honor to be a member of my REC group (Thriv5) \*Self-Definition
- I define what belonging to my REC group means to me (Self1)
- I am a member of my REC group on my own terms (Self2)
- I decide what my REC group membership is (Self3)
- I do not have to prove that I belong to my REC group to others in my REC group (Self4)
- Others do not define my belonging to my REC group (Self5)

A preprint of the article is available at ResearchGate. Below is the script for simulating item-level data from the factor loadings, means, and sample size presented in the published article.

Six additional scales were reported in the Lee and Neville [2024] article. Optimizing the ability for the simulation to converge and also to provide an array of scales within the nomological net, I chose the three scales below. Unfortunately, I could not locate factor loadings for all of them; in these cases, I uniformly assigned the factor loading of 0.8. The scales, their original citation, and information about how I simulated data for each are listed below.

- **General Belongingness Scale** [GBS; Malone et al. [2012]] is a 12-item item scale with Likert scaling ranging from 1 (*strongly disagree*) to 7 (*strongly agree*). There are two six-item subscales: Acceptance/Inclusion (“When I am with other people, I feel included”) and Rejection/Exclusion (“I feel like an outsider”). The items on the Rejection/Exclusion subscale must be rescored such that higher scores represent a greater sense of belonging in general. In the iBELONG study [Lee and Neville, 2024], only a total GBS scale was used. My simulation used factor loadings from Malone et al. [2012] but treated them as a single scale. This means that there is need to reverse-score the Rejection/Exclusion items in this simulated data.
- **The Collective Self-Esteem Scale—Race Specific Version** [CSESR; Crocker et al. [1994]] is a 16-item scale with Likert scaling ranging from 1 (*strongly disagree*) to 7 (*strongly agree*). Higher scores represent a stronger sense of membership with one’s own racial/ethnic group. There are four, four-item subscales: Membership CSE, Public CSE, Private CSE, and Importance to Identity, however the IBelong study [Lee and Neville, 2024] used a total scale score. An example item is, “The racial-ethnic group I belong to is an important reflection of who I am.” Because I was not able to locate factor loadings from a psychometric evaluation, I simulated the data by specifying a 0.8 as a standardized factor loading for each of the items on the general factor.
- **The Multigroup Ethnic Identity Measure – Revised** [MEIMR; Brown et al. [2014]] is a 7-item scale. The first item is an open-ended question for identification of respondent ethnic group. The remaining six items are assessed with Likert scaling ranging from 1 (*strongly disagree*) to 5 (*strongly agree*). There are two, three-item subscales which assess exploration (e.g., “I have spent time trying to find out more about my ethnic group, such as its history, tradition, and customs”) and commitment (“I have a strong sense of belonging to my own ethnic group.”) Higher scores indicate more positive ethnic identity. In the iBELONG study [Lee and Neville, 2024], only a total MEIMR scale was used. My simulation used factor loadings from Brown et al. [2014] but treated them as a single scale.

Because data is collected at the item level (and I want this resource to be as practical as possible, I have simulated the data for each of the scales at the item level. Simulating the data involved using

factor loadings, means, and correlations between the scales. Because the simulation will produce “out-of-bounds” values, the code below re-scales the scores into the range of the Likert-type scaling and rounds them to whole values.

```
#Entering the intercorrelations, means, and standard deviations from the journal article
```

```
iBelong_generating_model <- '
  #measurement model
  Authenticity =~ .66*Auth1 + .64*Auth2 + .63*Auth3 + .58*Auth4 + .55*Auth5
  Connection =~ .72*Conn1 + .60*Conn2 + .59*Conn3 + .58*Conn4 + .58*Conn5
  Home =~ .64*Home1 + .63*Home2 + .57*Home3 + .57*Home4 + .53*Home5
  RECthriving =~ .68*Thriv1 + .66*Thriv2 + .59*Thriv3 + .54*Thriv4 + .49*Thriv5
  SelfDefinition =~ .82*Self1 + .65*Self2 + .65*Self3 + .54*Self4 + .49*Self5
  GBS =~ .70*GBS1 + .67*GBS2 + .65*GBS5 + .67*GBS8 + .70*GBS10 + .78*GBS11 + .78*GBS3 +
  CSESR =~ .8*CSESR1 + .8*CSESR2 + .8*CSESR3 + .8*CSESR4 + .8*CSESR5 + .8*CSESR6 + .8*CSESR7
  MEIMR =~ .60*MEIMR1 + .93*MEIMR4 + .81*MEIMR5 + .86*MEIMR2 + .84*MEIMR3 + .89*MEIMR6

  #Means
  Authenticity ~ 4.30*1
  Connection ~ 4.16*1
  Home ~ 4.51*1
  RECthriving ~ 4.81*1
  SelfDefinition ~ 4.53*1
  GBS ~ 4.84*1
  CSESR ~ 5.15*1
  MEIMR ~ 3.81*1

  #Correlations
  Authenticity ~~ .73*Connection
  Authenticity ~~ .78*Home
  Authenticity ~~ .64*RECthriving
  Authenticity ~~ .52*SelfDefinition
  Authenticity ~~ .58*GBS
  Authenticity ~~ .58*CSESR
  Authenticity ~~ .48*MEIMR

  Connection ~~ .81*Home
  Connection ~~ .70*RECthriving
  Connection ~~ .43*SelfDefinition
  Connection ~~ .36*GBS
  Connection ~~ .63*CSESR
  Connection ~~ .62*MEIMR

  Home ~~ .74*RECthriving
  Home ~~ .48*SelfDefinition
  Home ~~ .51*GBS
  Home ~~ .65*CSESR
  Home ~~ .63*MEIMR
```

```

RECthriving ~~ .46*SelfDefinition
RECthriving ~~ .42*GBS
RECthriving ~~ .70*CSESR
RECthriving ~~ .70*MEIMR

SelfDefinition ~~ .29*GBS
SelfDefinition ~~ .38*CSESR
SelfDefinition ~~ .35*MEIMR

!

set.seed(240326)
iBel <- lavaan::simulateData(model = iBelong_generating_model,
                             model.type = "sem",
                             meanstructure = T,
                             sample.nobs=500,
                             standardized=FALSE)

#used to retrieve column indices used in the rescaling script below
col_index <- as.data.frame(colnames(iBel))

#The code below loops through each column of the dataframe and assigns the scaling accordingly
#Rows 1 thru 25 are the iBelong scale
#Rows 26 thru 37 are the General Belonging Scale
#Rows 38 thru 53 are the Collective Self Esteem Scale Race Specific Version
#Rows 54 thru 59 are the Multigroup Etnic Identity measure Revised

for(i in 1:ncol(iBel)){
  if(i >= 1 & i <= 25){
    iBel[,i] <- scales::rescale(iBel[,i], c(1, 6))
  }
  if(i >= 26 & i <= 37){
    iBel[,i] <- scales::rescale(iBel[,i], c(1, 7))
  }
  if(i >= 38 & i <= 53){
    iBel[,i] <- scales::rescale(iBel[,i], c(1, 7))
  }
  if(i >= 54 & i <= 59){
    iBel[,i] <- scales::rescale(iBel[,i], c(1, 5))
  }
}

#rounding to integers so that the data resembles that which was collected
library(tidyverse)
iBel <- iBel %>% round(0)

```

```
#quick check of my work
#psych::describe(iBel)
```

The optional script below will let you save the simulated data to your computing environment as either an .rds object (preserves any formatting you might do) or a.csv file (think “Excel lite”).

```
# to save the df as an .rds (think 'R object') file on your computer;
# it should save in the same file as the .rmd file you are working
# with saveRDS(iBel, 'iBel.rds') bring back the simulated dat from an
# .rds file iBel <- readRDS('iBel.rds')

# write the simulated data as a .csv write.table(iBel,
# file='iBel.csv', sep=',', col.names=TRUE, row.names=FALSE) bring
# back the simulated dat from a .csv file iBel <-
# read.csv('iBel.csv', header = TRUE)
```

Lessons in this volume teach test-retest reliability and invariance testing. In both of these circumstances, data from two different administrations of the focal test are required. For test-retest, these would be completed by the same person; for invariance-testing, these would be completed by different people. Below I have simulated two sets of data for the iBelong Scale [Lee and Neville, 2024]. I have named them “A” and “B”. For your homework purposes, you can determine if they will represent testing with the same individual (appropriate for test-retest reliability) or different groups (appropriate for invariance testing).

```
iBelong_generating_modelAB <- '
  #measurement model
  AuthenticityA =~ .66*Auth1a + .64*Auth2a + .63*Auth3a + .58*Auth4a + .55*Auth5a
  ConnectionA =~ .72*Conn1a + .60*Conn2a + .59*Conn3a + .58*Conn4a + .58*Conn5a
  HomeA =~ .64*Home1a + .63*Home2a + .57*Home3a + .57*Home4a + .53*Home5a
  RECthrivingA =~ .68*Thriv1a + .66*Thriv2a + .59*Thriv3a + .54*Thriv4a + .49*Thriv5a
  SelfDefinitionA =~ .82*Self1a + .65*Self2a + .65*Self3a + .54*Self4a + .49*Self5a
  AuthenticityB =~ .66*Auth1b + .64*Auth2b + .63*Auth3b + .58*Auth4b + .55*Auth5b
  ConnectionB =~ .72*Conn1b + .60*Conn2b + .59*Conn3b + .58*Conn4b + .58*Conn5b
  HomeB =~ .64*Home1b + .63*Home2b + .57*Home3b + .57*Home4b + .53*Home5b
  RECthrivingB =~ .68*Thriv1b + .66*Thriv2b + .59*Thriv3b + .54*Thriv4b + .49*Thriv5b
  SelfDefinitionB =~ .82*Self1b + .65*Self2b + .65*Self3b + .54*Self4b + .49*Self5b

  #Means
  AuthenticityA ~ 4.30*1
  ConnectionA ~ 4.16*1
  HomeA ~ 4.51*1
  RECthrivingA ~ 4.81*1
  SelfDefinitionA ~ 4.53*1

  AuthenticityB ~ 4.28*1
```

```

ConnectionB ~ 4.20*1
HomeB ~ 4.44*1
RECthrivingB ~ 4.9*1
SelfDefinitionB ~ 4.49*1

#Correlations
AuthenticityA ~~ .81*AuthenticityB
ConnectionA ~~ .85*ConnectionB
HomeA ~~ .77*HomeB
RECthrivingA ~~ .85*RECthrivingB
SelfDefinitionA ~~ .84*SelfDefinitionB

'

set.seed(240326)
iBelAB <- lavaan::simulateData(model = iBelong_generating_modelAB,
                                model.type = "sem",
                                meanstructure = T,
                                sample.nobs=500,
                                standardized=FALSE)

#used to retrieve column indices used in the rescaling script below
col_index <- as.data.frame(colnames(iBelAB))

#The code below loops through each column of the dataframe and assigns the scaling accordingly
#All rows are the iBel scales, administrations A and B

for(i in 1:ncol(iBelAB)){
  if(i >= 1 & i <= 50){
    iBelAB[,i] <- scales::rescale(iBelAB[,i], c(1, 6))
  }
}

#rounding to integers so that the data resembles that which was collected
library(tidyverse)
iBelAB <- iBelAB %>% round(0)

#quick check of my work
psych::describe(iBelAB)

```

## 14.2 Identity Threat

The Identity Threat Scale [George et al., 2023] is a measure of threat to identity value, meanings, and enactment. The measure is designed to adapt to different identities. The scale includes 19 items with responses rated on a 6-point Likert scale ranging from 1 (*strongly disagree*) to 5 (*strongly agree*).

*agree).* Items are written such that researchers can provide the focal identity (e.g., “a teacher,” “LGBTQ”). Across all three scales, higher scores reflect higher threat. It appears that the authors landed on a single-order, correlated factors, structure.

Beneath each of the three factors are a list of the items. Variable names follow in parentheses.

- Identity Value:
  - I feel that there is a negative value attached to my identity as [LGBTQ]. (Val1)
  - Being [LGBTQ] is worth less in the eyes of others than before. (Val2)
  - I feel that others attach a negative value to my identity as [LGBTQ]. (Val3)
  - I feel that my identity as [LGBTQ] is devalued by others. (Val4)
  - I feel that others see little value in my identity as [LGBTQ].(Val5)
- Identity Meaning:
  - I am no longer sure what it means to be [LGBTQ].(Mng1)
  - I am questioning what it means to be [LGBTQ]. (Mng2)
  - I find myself questioning what it means to be [LGBTQ]. (Mng3)
  - The core of what it means to be [LGBTQ] is changing in a way I do not like. (Mng4)
  - What it means to be [LGBTQ] is changing in a way I do not like. (Mng5)
  - What it means to be [LGBTQ] is being called into question. (Mng6)
  - Being [LGBTQ] used to mean something different. (Mng7) -I feel that being [LGBTQ] does not mean the same thing anymore. (Mng8)
- Identity Enactment
  - I am limited in my ability to express my identity as [LGBTQ]. (Enact1)
  - I may no longer be able to engage in activities that express my identity as [LGBTQ]. (Enact2)
  - I may no longer be able to show that I am [LGBTQ]. (Enact3)
  - I worry about no longer being able to express my identity as [LGBTQ]. (Enact4)
  - I worry that I cannot behave in the way [an LGBTQ person] behaves. (Enact5)
  - I worry that I cannot show people that I am [LGBTQ]. (Enact6)

Although I have simulated data, the authors posted their data and codebooks from the different phases at the [OSF repository](#). Additional materials are available as a [supplement](#) to the article.

There were a number of scales utilized in this study. Because including them all caused problems of convergence, beyond the subscales in the focal measure, I have simulated data for only a few more. This will provide practice in evaluating convergent, discriminant, and incremental validity.

- **Self-esteem** was assessed with [Rosenberg’s 10-item scale](#). Responses are measured on a 4-point Likert scale ranging from 1 (*strongly agree*) to 4 (*strongly disagree*). Some items are reverse-coded. However, using the data from this simulation presumes that reverse-scoring has already been completed. Higher scores indicate higher self-esteem. An example item is “On the whole, I am satisfied with myself.” Factor loadings were obtained from Mullen et al. [\[2013\]](#).
- **Identity suppression** was assessed with 4 items from Madera et al.’s [\[2012\]](#) 10 item scale. It is a bit unclear, but it appears that items were rated on a 1 (*strongly disagree*) to 7 (*strongly agree*) scale with higher scores reflecting higher identity suppression. A sample item is, “I refrain from talking about my identity as LGBTQ at work.”

- **Workplace microaggressions** was captured with 11 items from Resnick and Galupo's [Resnick and Galupo, 2019] scale. Responses are measured on a Likert type scale ranging from 1 (*never*) to 5 (*a great deal*). Higher scores indicate higher frequency of microaggressions. An example item is "Having my behaviors mimicked in a joking way due to my LGBTQ identity." It is unclear if they used one or more scales. I simulated the data from Resnick and Galupo, selecting 3 and 4 items from three subscales.
- **Green behaviors** were captured with three items from Norton et al.'s [Norton et al., 2017] scale. Responses are measured on a Likert scale ranging from 1 (*strongly disagree*) to 5 (*strongly agree*). The response stem was, "Tomorrow, I intend to..." and items included, "...act in environmentally friendly ways," "...carry out environmentally friendly behaviors at work," and "...perform pro-environmental behaviors while at work. In the absence of factor loadings, I made them up.

Because data is collected at the item level (and I want this resource to be as practical as possible, I have simulated the data for each of the scales at the item level. I guided the simulation using factor loadings from the exploratory factor analysis results and means and correlations from the Stage 5 analysis where 516 individuals who identified as LGBTQ completed the survey. Because the simulation will produce "out-of-bounds" values, the code below rescales the scores into the range of the Likert-type scaling and rounds them to whole values.

*#Entering the intercorrelations, means, and standard deviations from the journal article*

```
IdentityThreat_generating_model <- '
  #measurement model
  Value   =~ .83*Val1 + .78*Val2 + .88*Val3 + .80*Val4 + .73*Val5
  Meaning =~ .66*Mng1 + .81*Mng2 + .85*Mng3 + .78*Mng4 + .74*Mng5 + .64*Mng6 + .65*Mng7 +
  Enactment =~ .66*Enact1 + .64*Enact2 + .82*Enact3 + .70*Enact4 + .66*Enact5 + .94*Enact6
  SelfEsteem =~ .95*SE1 + .98*SE2 + .66*SE3 + .81*SE4 + .55*SE5 + .80*SE6 + .75*SE7 + .68*SE8
  IDsuppression =~ .78*IDs1 + .67*IDs2 + .82*IDs3 + .90*IDs4
  WkplcMicroAgg =~ .84*Micro1 + .87*Micro2 + .72*Micro3 + .78*Micro4 + .67*Micro5 + .81*Micro6
  GreenBx =~ .77*G1 + .81*G2 + .83*G3

  #Means
  Value ~ 2.15*1
  Meaning ~ 2.10*1
  Enactment ~ 2.17*1
  SelfEsteem ~ 3.44*1
  IDsuppression ~ 3.13*1
  WkplcMicroAgg ~ 1.24*1
  GreenBx ~ 3.49*1

  #Correlations
  Value ~~ .58*Meaning
  Value ~~ .73*Enactment
  Value ~~ -.21*SelfEsteem
  Value ~~ .35*IDsuppression
  Value ~~ .41*WkplcMicroAgg
  Value ~~ -.09*GreenBx
```

```

Meaning ~~ .60*Enactment
Meaning ~~ .63*SelfEsteem
Meaning ~~ .26*IDsuppression
Meaning ~~ .24*WkplcMicroAgg
Meaning ~~ -.08*GreenBx

Enactment ~~ -.31*SelfEsteem
Enactment ~~ .54*IDsuppression
Enactment ~~ .24*WkplcMicroAgg
Enactment ~~ -.06*GreenBx

SelfEsteem ~~ -.24*IDsuppression
SelfEsteem ~~ -.06*WkplcMicroAgg
SelfEsteem ~~ .15*GreenBx

IDsuppression ~~ .00*WkplcMicroAgg
IDsuppression ~~ -.02*GreenBx

'

set.seed(240330)
IdThreat <- lavaan::simulateData(model = IdentityThreat_generating_model,
                                    model.type = "sem",
                                    meanstructure = T,
                                    sample.nobs=516,
                                    standardized=FALSE)

#used to retrieve column indices used in the rescaling script below
col_index <- as.data.frame(colnames(IdThreat))

#The code below loops through each column of the dataframe and assigns the scaling accordingly
#Rows 1 thru 19 are the Identity Threat
#Rows 20 thru 29 are Self Esteem
#Rows 30 thru 33 are Identity Suppression
#Rows 34 thru 44 are Workplace Microaggressions
#Rows 45 thru 47 are Green Behaviors

for(i in 1:ncol(IdThreat)){
  if(i >= 1 & i <= 19){
    IdThreat[,i] <- scales::rescale(IdThreat[,i], c(1, 7))
  }
  if(i >= 20 & i <= 29){
    IdThreat[,i] <- scales::rescale(IdThreat[,i], c(1, 4))
  }
  if(i >= 30 & i <= 33){
    IdThreat[,i] <- scales::rescale(IdThreat[,i], c(1, 7))
  }
}

```

```

if(i >= 34 & i <= 44){
  IdThreat[,i] <- scales::rescale(IdThreat[,i], c(1, 5))
}
if(i >= 45 & i <= 47){
  IdThreat[,i] <- scales::rescale(IdThreat[,i], c(1, 5))
}
}

#rounding to integers so that the data resembles that which was collected
library(tidyverse)
IdThreat <- IdThreat %>% round(0)

#quick check of my work
#psych::describe(IdThreat)

```

The optional script below will let you save the simulated data to your computing environment as either an .rds object (preserves any formatting you might do) or a.csv file (think “Excel lite”).

```

# to save the df as an .rds (think 'R object') file on your computer;
# it should save in the same file as the .rmd file you are working
# with saveRDS(IdThreat, 'IdThreat.rds') bring back the simulated dat
# from an .rds file IdThreat <- readRDS('IdThreat.rds')

# write the simulated data as a .csv write.table(IdThreat,
# file='IdThreat.csv', sep=',', col.names=TRUE, row.names=FALSE)
# bring back the simulated dat from a .csv file IdThreat <-
# read.csv('IdThreat.csv', header = TRUE)

```

Lessons in this volume teach test-retest reliability and invariance testing. In both of these circumstances, data from two different administrations of the focal test are required. For test-retest, these would be completed by the same person; for invariance-testing, these would be completed by different people. Below I have simulated two sets of data for The Identity Threat Scale [George et al., 2023]. I have named them “A” and “B”. For your homework purposes, you can determine if they will represent testing with the same individual (appropriate for test-retest reliability) or different groups (appropriate for invariance testing).

```

idThreat_generating_modelAB <- '
  #measurement model
  ValueA =~ .83*Val1a + .78*Val2a + .88*Val3a + .80*Val4a + .73*Val5a
  MeaningA =~ .66*Mng1a + .81*Mng2a + .85*Mng3a + .78*Mng4a + .74*Mng5a + .64*Mng6a + .64*Mng7a
  EnactmentA =~ .66*Enact1a + .64*Enact2a + .82*Enact3a + .70*Enact4a + .66*Enact5a + .94*Enact6a
  ValueB =~ .83*Val1b + .78*Val2b + .88*Val3b + .80*Val4b + .73*Val5b
  MeaningB =~ .66*Mng1b + .81*Mng2b + .85*Mng3b + .78*Mng4b + .74*Mng5b + .64*Mng6b + .64*Mng7b
  EnactmentB =~ .66*Enact1b + .64*Enact2b + .82*Enact3b + .70*Enact4b + .66*Enact5b + .94*Enact6b

  #Means

```

```

ValueA ~ 1.97*1
MeaningA ~ 2.16*1
EnactmentA ~ 2.14*1
ValueB ~ 2.10*1
MeaningB ~ 2.28*1
EnactmentB ~ 2.24*1

#Correlations
ValueA ~~ .80*MeaningA
ValueA ~~ .80*EnactmentA
ValueA ~~ .63*ValueB
ValueA ~~ .57*MeaningB
ValueA ~~ .56*EnactmentB

MeaningA ~~ .77*EnactmentA
MeaningA ~~ .59*ValueB
MeaningA ~~ .70*MeaningB
MeaningA ~~ .58*EnactmentB

EnactmentA ~~ .54*ValueB
EnactmentA ~~ .79*MeaningB
EnactmentA ~~ .66*EnactmentB

ValueB ~~ .81*MeaningB
ValueB ~~ .82*EnactmentB

MeaningB ~~ .81*EnactmentB

'

set.seed(240326)
idThreatAB <- lavaan::simulateData(model = idThreat_generating_modelAB,
                                      model.type = "sem",
                                      meanstructure = T,
                                      sample.nobs=500,
                                      standardized=FALSE)

#used to retrieve column indices used in the rescaling script below
col_index <- as.data.frame(colnames(idThreatAB))

#The code below loops through each column of the dataframe and assigns the scaling accordingly
#All rows are the iBel scales, administrations A and B

for(i in 1:ncol(idThreatAB)){
  if(i >= 1 & i <= 38){

```

```

    idThreatAB[,i] <- scales::rescale(idThreatAB[,i], c(1, 5))
  }
}

#rounding to integers so that the data resembles that which was collected
library(tidyverse)
idThreatAB <- idThreatAB %>% round(0)

#quick check of my work
#psych::describe(iBelAB)

```

### 14.3 Anti-Racism Behavioral Inventory

The Anti-Racism Behavioral Inventory [Pieterse et al., 2022] is a measure that assesses anti-racism awareness and behavior among white Americans. The scale includes 21 items with responses rated on a 5-point Likert scale ranging from 1 (*strongly disagree*) to 5 (*strongly agree*). Higher scores indicate a greater anti-racist awareness and behavior.

Beneath each of the five factors are a list of the items. Variable names follow in parentheses.

- Individual Advocacy:
  - When I hear people telling racist jokes and using negative racial stereotypes, I usually confront them (Advoc1)
  - I actively seek to understand how I participate in both intentional and unintentional racism (Advoc2)
  - I actively seek to educate myself about the experience of racism (Advoc3)
  - I interrupt racist conversations and jokes when I heard my friends talking that way (Advoc4)
  - I have challenged acts of racism that I have witnessed in my workplace or at school (Advoc5)
  - I make it a point to educate myself about the experience of historically oppressed groups in the US (e.g. slavery, internment of Japanese, American Indians, and the trail of tears etc.) (Advoc6)
  - I often speak to my friends about the problem of racism in the US, and what we can do about it (Advoc7)
  - I do not like to talk about racism in public (Advoc8) NEEDS REVERSING
  - I interrupt racist conversations and jokes when I hear them in my family (Advoc9)
- Awareness of Racism:
  - I feel guilty and ashamed when I think of the history of racism and slavery in the US (Aware1)
  - It bothers me that my country has yet to acknowledge the impact of slavery (Aware2)
  - The US should offer some type of payment to the descendants of slaves (Aware3)
  - The US has not acknowledged the impact of slavery (Aware4)
  - Because of racism in the US, Blacks do not have the same educational opportunities as compared to Whites (Aware5)

- Within the US, racism is largely perpetuated by the White racial majority (Aware6)
- The police unfairly target Black men and Latinos (Aware7)
- Institutional Advocacy
  - I give money to organizations working against racism and discrimination (Inst1)
  - When I read articles in newspapers or magazines that are perpetuating racist ideas, I generally write a letter to the editor (Inst2)
  - I am actively involved in exposing companies that uphold exclusionary and racist practices (Inst3)
  - I write letters to local and state politicians to voice my concerns about racism (Inst4)
  - I volunteer with anti-racist or racial justice organizations (Inst5)

This article was published in the open access Journal for Social Action in Counseling and Psychology and is available at its [website](#). Below is the script for simulating item-level data from the factor loadings, means, and sample size presented in the published article.

Four additional scales were reported in the Pieterse et al. [2022] article. Unfortunately, I could not locate factor loadings for all of them; in these cases, I uniformly assigned the factor loading of 0.8. The scales, their original citation, and information about how I simulated data for each are listed below.

- **White Privilege** was assessed using a 13-item subscale from The Privilege and Oppression Inventory\*\* [Hays et al., 2007]. Items are rated on a Likert scale ranging from 1 (*strongly disagree*) to 6 (*strongly agree*). An example item is, “Whites have the power to exclude other groups.” Higher scores indicate greater levels of White privilege.
- **Color Blind Racial Attitudes** [Neville et al., 2000] is a 21-item scale that assesses implicit racism-related attitudes. The 21 items are assessed on a Likert scale ranging from 1 to 5 with higher scores reflecting greater levels of color-blind racial attitudes. I was able to simulate two of the three subscales of the instrument: Racial Privilege (RP) and Blatant Racial Issues (BR).

Because data is collected at the item level (and I want this resource to be as practical as possible, I have simulated the data for each of the scales at the item level. Simulating the data involved using factor loadings, means, and correlations between the scales. Where possible, factor loadings are retrieved from original (or replication) articles that report on their psychometric evaluation. The means and correlations are from Pietierse et al. [Pieterse et al., 2022] Because the simulation will produce “out-of-bounds” values, the code below rescales the scores into the range of the Likert-type scaling and rounds them to whole values.

```
#Entering the intercorrelations, means, and standard deviations from the journal article
```

```
Antiracism_generating_model <- '
  #measurement model
  ARBI =~ .274*Advoc1 + .699*Advoc2 + .752*Advoc3 + .465*Advoc4 + .479*Advoc5 + .617*Advoc6
  Advocacy =~ .618*Advoc1 + .055*Advoc2 + .035*Advoc3 + .712*Advoc4 + .275*Advoc5 + .124*Advoc6
  Awareness =~ .211*Aware1 + .669*Aware2 + .428*Aware3 + .692*Aware4 + .445*Aware5 + .508*Aware6
  Institutional =~ .381*Inst1 + .547*Inst2 + .648*Inst3 + .618*Inst4 + .392*Inst5
```

```

WhitePriv =~ .80*WP1 + .59*WP2 + .62*WP3 + .65*WP4 + .80*WP5 + .50*WP6 + .78*WP7 + .57
CoBRAS_RP =~ .68*RP1 + .66*RP2 + .64*RP3 + .62*RP4 + .62*RP5 + .58*RP6 + .54*RP7
CoBRAS_BR =~ .77*BR1 + .67*BR2 + .65*BR3 + .65*BR4 + .62*BR5 + .55*BR6 + .48*BR7

#Means
ARBI ~ 3.91*1
Advocacy ~5.33*1
Awareness ~ 4.26*1
Institutional ~ 1.66*1
WhitePriv ~ 5.20*1
CoBRAS_RP ~ 2.54*1
CoBRAS_BR ~ 1.65*1

#Correlations
ARBI ~~ .78*Awareness
ARBI ~~ .87*Advocacy
ARBI ~~ .75*Institutional
ARBI ~~ .64*WhitePriv
ARBI ~~ -.65*CoBRAS_RP
ARBI ~~ -.60*CoBRAS_BR

Awareness ~~ .50*Advocacy
Awareness ~~ .34*Institutional
Awareness ~~ .81*WhitePriv
Awareness ~~ -.78*CoBRAS_RP
Awareness ~~ -.62*CoBRAS_BR

Advocacy ~~ .55*Institutional
Advocacy ~~ .43*WhitePriv
Advocacy ~~ -.45*CoBRAS_RP
Advocacy ~~ -.46*CoBRAS_BR

Institutional ~~ .27*WhitePriv
Institutional ~~ -.31*CoBRAS_RP
Institutional ~~ -.34*CoBRAS_BR

WhitePriv ~~ -.85*CoBRAS_RP
WhitePriv ~~ -.6*CoBRAS_BR

CoBRAS_RP ~~ .60*CoBRAS_BR
'

set.seed(240218)
AntiR <- lavaan:::simulateData(model = Antiracism_generating_model,
                                model.type = "sem",
                                meanstructure = T,
                                sample.nobs=153,

```

```

    standardized=FALSE)

#used to retrieve column indices used in the rescaling script below
col_index <- as.data.frame(colnames(AntiR))

#The code below loops through each column of the dataframe and assigns the scaling accordingly
#Rows 1 thru 21 are the ARBI
#Rows 22 thru 34 are the White Privilege Scale
#Rows 35 thru 48 are the CoBRAS

for(i in 1:ncol(AntiR)){
  if(i >= 1 & i <= 21){
    AntiR[,i] <- scales::rescale(AntiR[,i], c(1, 5))
  }
  if(i >= 22 & i <= 34){
    AntiR[,i] <- scales::rescale(AntiR[,i], c(1, 6))
  }
  if(i >= 35 & i <= 48){
    AntiR[,i] <- scales::rescale(AntiR[,i], c(1, 5))
  }
}

#rounding to integers so that the data resembles that which was collected
library(tidyverse)
AntiR <- AntiR %>% round(0)

#quick check of my work
#psych::describe(AntiR)

#Reversing the supportive item on the Perceptions of LGBTQ Campus Climate Scale so that the ex

```

The optional script below will let you save the simulated data to your computing environment as either an .rds object (preserves any formatting you might do) or a.csv file (think “Excel lite”).

```

# to save the df as an .rds (think 'R object') file on your computer;
# it should save in the same file as the .rmd file you are working
# with saveRDS(AntiR, 'AntiR.rds') bring back the simulated dat from
# an .rds file AntiR <- readRDS('AntiR.rds')

```

```

# write the simulated data as a .csv write.table(AntiR,
# file='AntiR.csv', sep=',', col.names=TRUE, row.names=FALSE) bring
# back the simulated dat from a .csv file AntiR <-
# read.csv('AntiR.csv', header = TRUE)

```

Lessons in this volume teach test-retest reliability and invariance testing. In both of these circumstances, data from two different administrations of the focal test are required. For test-retest,

these would be completed by the same person; for invariance-testing, these would be completed by different people. Below I have simulated two sets of data for the Anti-Racism Behavioral Inventory [Pieterse et al., 2022]. I have named them “A” and “B”. For your homework purposes, you can determine if they will represent testing with the same individual (appropriate for test-retest reliability) or different groups (appropriate for invariance testing).

The Pieterse et al.[2022] article did not report invariance testing nor test-retest reliabilities. Therefore, I simulated this data following the general patterns of such testing.

```
ARBI_generating_modelAB <- '
  #measurement model
  AdvocacyA =~ .450*Advoc1a + .671*Advoc2a + .715*Advoc3a + .650*Advoc4a + .548*Advoc5a
  AwarenessA =~ .239*Aware1a + .836*Aware2a + .754*Aware3a + .877*Aware4a + .791*Aware5a
  InstitutionalA =~ .698*Inst1a + .397*Inst2a + .680*Inst3a + .589*Inst4a + .798*Inst5a
  AdvocacyB =~ .450*Advoc1b + .671*Advoc2b + .715*Advoc3b + .650*Advoc4b + .548*Advoc5b
  AwarenessB =~ .239*Aware1b + .836*Aware2b + .754*Aware3b + .877*Aware4b + .791*Aware5b
  InstitutionalB =~ .698*Inst1b + .397*Inst2b + .680*Inst3b + .589*Inst4b + .798*Inst5b

  #Means
  AdvocacyA ~5.33*1
  AwarenessA ~ 4.26*1
  InstitutionalA ~ 1.66*1
  AdvocacyB ~5.25*1
  AwarenessB ~ 4.30*1
  InstitutionalB ~ 1.71*1

  #Correlations
  AdvocacyA ~~ .80*AwarenessA
  AdvocacyA ~~ .80*InstitutionalA
  AdvocacyA ~~ .63*AdvocacyB
  AdvocacyA ~~ .57*AwarenessB
  AdvocacyA ~~ .56*InstitutionalB

  AwarenessA ~~ .77*InstitutionalA
  AwarenessA ~~ .59*AdvocacyB
  AwarenessA ~~ .70*AwarenessB
  AwarenessA ~~ .58*InstitutionalB

  InstitutionalA ~~ .54*AdvocacyB
  InstitutionalA ~~ .79*AwarenessB
  InstitutionalA ~~ .66*InstitutionalB

  AdvocacyB ~~ .81*AwarenessB
  AdvocacyB ~~ .82*InstitutionalB

  AwarenessB ~~ .81*InstitutionalB
```

```
 1

set.seed(240326)
ARBI_AB <- lavaan::simulateData(model = ARBI_generating_modelAB,
                                 model.type = "sem",
                                 meanstructure = T,
                                 sample.nobs=300,
                                 standardized=FALSE)

#used to retrieve column indices used in the rescaling script below
col_index <- as.data.frame(colnames(ARBI_AB))

#The code below loops through each column of the dataframe and assigns the scaling accordingly
#All rows are the iBel scales, administrations A and B

for(i in 1:ncol(ARBI_AB)){
  if(i >= 1 & i <= 42){
    ARBI_AB[,i] <- scales::rescale(ARBI_AB[,i], c(1, 5))
  }
}

#rounding to integers so that the data resembles that which was collected
library(tidyverse)
ARBI_AB <- ARBI_AB %>% round(0)

#quick check of my work
#psych::describe(iBelAB)
```

# REFERENCES



# Bibliography

FACT SHEET: Anti-Asian Prejudice March 2020, a. URL <https://www.csusb.edu/sites/default/files/FACT%20SHEET-%20Anti-Asian%20Hate%202020%203.2.21.pdf>.

STOP AAPI HATE, b. URL <https://stopaapihate.org/>.

Hector Y. Adames, Nayeli Y. Chavez-Dueñas, and Maryam M. Jernigan. The fallacy of a raceless Latinidad: Action guidelines for centering Blackness in Latinx psychology. *Journal of Latinx Psychology*, 9(1):26–44, February 2021. ISSN 2578-8086. doi: 10.1037/lat0000179. URL <https://ezproxy.spu.edu/login?url=http://search.ebscohost.com/login.aspx?direct=true&AuthType=ip&db=psyh&AN=2020-69785-001&site=ehost-live>. Publisher: Educational Publishing Foundation.

Anne Anastasi and Susana Urbina. *Psychological testing*, 7th ed. Prentice Hall/Pearson Education, Upper Saddle River, NJ, 1997. ISBN 978-0-02-303085-7. URL <https://ezproxy.spu.edu/login?url=https://search.ebscohost.com/login.aspx?direct=true&AuthType=ip&db=psyh&AN=1998-07223-000&site=ehost-live>.

Kelsey L. Autin, Blake A. Allan, Meenakshi Palaniappan, and Ryan D. Duffy. Career Calling in India and the United States: A Cross-Cultural Measurement Study. *Journal of Career Assessment*, 25(4):688–702, November 2017. ISSN 1069-0727, 1552-4590. doi: 10.1177/1069072716665860. URL <http://journals.sagepub.com/doi/10.1177/1069072716665860>.

A Bandura and Albert Bandura. Guide for creating self-efficacy scales. In F. Pajares and T. Urdan, editors, *Self-efficacy beliefs of adolescents*, pages 307–337. Information Age Publishing, Greenwich, CT, 2006.

Martin Brattmyr, Martin Schevik Lindberg, Stian Solem, Odin Hjemdal, and Audun Havnen. Factor structure, measurement invariance, and concurrent validity of the Patient Health Questionnaire-9 and the Generalized Anxiety Disorder Scale-7 in a Norwegian psychiatric outpatient sample. *BMC Psychiatry*, 22(1):461, July 2022. ISSN 1471-244X. doi: 10.1186/s12888-022-04101-z. URL <https://doi.org/10.1186/s12888-022-04101-z>.

Susan D. Brown, Kirsten A. Unger Hu, Ashley A. Mevi, Monique M. Hedderson, Jun Shan, Charles P. Quesenberry, and Assiamira Ferrara. The Multigroup Ethnic Identity Measure-Revised: Measurement invariance across racial and ethnic groups. *Journal of counseling psychology*, 61(1):154–161, January 2014. ISSN 0022-0167. doi: 10.1037/a0034749. URL <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4110058/>.

Elizabeth A. Buchanan and Erin E. Hvizdak. Online Survey Tools: Ethical and Methodological Concerns of Human Research Ethics Committees. *Journal of Empirical Research on Human Research Ethics; London*, 4(2):37–48, June 2009. ISSN 15562646. doi: <http://dx.doi.org/>

- ezproxy.spu.edu/10.1525/jer.2009.4.2.37. URL <http://search.proquest.com/docview/219055147/abstract/BE05412DDB224725PQ/9>.
- Barbara M. Byrne. Adaptation of assessment scales in cross-national research: Issues, guidelines, and caveats. *International Perspectives in Psychology: Research, Practice, Consultation*, 5(1):51–65, January 2016a. ISSN 2157-3883. doi: 10.1037/ipp0000042. URL <http://ezproxy.spu.edu/login?url=http://search.ebscohost.com/login.aspx?direct=true&AuthType=ip&db=psyh&AN=2015-47625-001&site=ehost-live>.
- Barbara M. Byrne. *Structural Equation Modeling with AMOS: Basic Concepts, Applications, and Programming, Third Edition*. Taylor & Francis Group, London, UNITED KINGDOM, 2016b. ISBN 978-1-317-63313-6. URL <http://ebookcentral.proquest.com/lib/spu/detail.action?docID=4556523>.
- Barbara M. Byrne. Structural Equation Modeling: The basics (Chapter 1). In *Structural Equation Modeling with AMOS: Basic Concepts, Applications, and Programming, Third Edition*. Taylor & Francis Group, London, UNITED KINGDOM, 2016c. ISBN 978-1-317-63313-6. URL <http://ebookcentral.proquest.com/lib/spu/detail.action?docID=4556523>.
- Cristalis Capielo Rosario, Hector Y. Adames, Nayeli Y. Chavez-Dueñas, and Roberto Renteria. Acculturation Profiles of Central Florida Puerto Ricans: Examining the Influence of Skin Color, Perceived Ethnic-Racial Discrimination, and Neighborhood Ethnic-Racial Composition. *Journal of Cross-Cultural Psychology*, 50(4):556–576, May 2019. ISSN 0022-0221, 1552-5422. doi: 10.1177/0022022119835979. URL <http://journals.sagepub.com/doi/10.1177/0022022119835979>.
- Andrew Young Choi, Tania Israel, and Hotaka Maeda. Development and evaluation of the Internalized Racism in Asian Americans Scale (IRAAS): Journal of Counseling Psychology. *Journal of Counseling Psychology*, 64(1):52–64, January 2017. ISSN 0022-0167. doi: 10.1037/cou0000183. URL <https://ezproxy.spu.edu/login?url=https://search.ebscohost.com/login.aspx?direct=true&AuthType=ip&db=psyh&AN=2017-00131-004&site=ehost-live>. Publisher: American Psychological Association.
- Seung Youn Chyung, Katherine Roberts, Ieva Swanson, and Andrea Hankinson. Evidence-Based Survey Design: The Use of a Midpoint on the Likert Scale. *Performance Improvement*, 56(10):15–23, November 2017. ISSN 10908811. doi: 10.1002/pfi.21727. URL <http://ezproxy.spu.edu/login?url=http://search.ebscohost.com/login.aspx?direct=true&AuthType=ip&db=eue&AN=126315473&site=ehost-live>.
- Seung Youn Chyung, Julie R.2 Barkin, and Jennifer A.3 Shamsy. Evidence-Based Survey Design: The Use of Negatively Worded Items in Surveys. *Performance Improvement*, 57(3):16–25, March 2018a. ISSN 10908811. doi: 10.1002/pfi.21749. URL <http://ezproxy.spu.edu/login?url=http://search.ebscohost.com/login.aspx?direct=true&AuthType=ip&db=eue&AN=128710204&site=ehost-live>.
- Seung Youn Chyung, Megan Kennedy, and Ingrid Campbell. Evidence-Based Survey Design: The Use of Ascending or Descending Order of Likert-Type Response Options. *Performance Improvement*, 57(9):9–16, 2018b. ISSN 1930-8272. doi: 10.1002/pfi.21800. URL <http://onlinelibrary.wiley.com/doi/abs/10.1002/pfi.21800>.
- Seung Youn Chyung, Ieva Swanson, Katherine Roberts, and Andrea Hankinson. Evidence-Based Survey Design: The Use of Continuous Rating Scales in Surveys. *Performance Improvement*, 57(5):38–48, May 2018c. ISSN 10908811. doi: 10.1002/pfi.

21763. URL <http://ezproxy.spu.edu/login?url=http://search.ebscohost.com/login.aspx?direct=true&AuthType=ip&db=bth&AN=129411016&site=ehost-live>.
- David Colton and Robert W. Covert. *Designing and Constructing Instruments for Social Research and Evaluation*. John Wiley & Sons, June 2015. ISBN 978-1-119-17797-5. Google-Books-ID: MLD5CQAAQBAJ.
- Kristin J Conover, Tania Israel, and Karen Nylund-Gibson. Development and Validation of the Ableist Microaggressions Scale. *The Counseling Psychologist*, 45(4):30, 2017. doi: 10.1177/0011000017715317.
- Jose M. Cortina. What is coefficient alpha? An examination of theory and applications. *Journal of Applied Psychology*, 78(1):98–104, 1993. ISSN 0021-9010. doi: 10.1037/0021-9010.78.1.98. URL <http://doi.apa.org/getdoi.cfm?doi=10.1037/0021-9010.78.1.98>.
- Jennifer Crocker, Riia Luhtanen, Bruce Blaine, and Stephanie Broadnax. Collective self-esteem and psychological well-being among White, Black, and Asian college students: Personality and Social Psychology Bulletin. *Personality and Social Psychology Bulletin*, 20(5):503–513, October 1994. ISSN 0146-1672. doi: 10.1177/0146167294205007. Publisher: Sage Publications.
- Craig K. Enders. *Applied missing data analysis*. Guilford Press, New York, NY, 2010. ISBN 978-1-60623-639-0.
- Andy P. Field. *Discovering statistics using R*. Sage, Thousand Oaks, California, 2012. ISBN 978-1-4462-0046-9.
- David B. Flora. Supplemental Material to: Your Coefficient Alpha Is Probably Wrong, but Which Coefficient Omega Is Right? A Tutorial on Using R to Obtain Better Reliability Estimates. *Advances in Methods and Practices in Psychological Science*, 3(4):484–501, December 2020a. ISSN 2515-2459. doi: 10.1177/2515245920951747. URL <https://doi.org/10.1177/2515245920951747>. Publisher: SAGE Publications Inc.
- David B. Flora. Your Coefficient Alpha Is Probably Wrong, but Which Coefficient Omega Is Right? A Tutorial on Using R to Obtain Better Reliability Estimates. *Advances in Methods and Practices in Psychological Science*, 3(4):484–501, December 2020b. ISSN 2515-2459. doi: 10.1177/2515245920951747. URL <https://doi.org/10.1177/2515245920951747>. Publisher: SAGE Publications Inc.
- Mailys M. George, Karoline Strauss, Julija N. Mell, and Heather C. Vough. When 'who I am' is under threat: Measures of threat to identity value, meanings, and enactment: Journal of Applied Psychology. *Journal of Applied Psychology*, 108(12):1952–1978, December 2023. ISSN 0021-9010. doi: 10.1037/apl0001114. supp. URL <https://ezproxy.spu.edu/login?url=https://search.ebscohost.com/login.aspx?direct=true&AuthType=ip&db=psyh&AN=2023-93532-001&site=ehost-live>. Publisher: American Psychological Association.
- Lawrence H. Gerstein. Theory and intervention equivalence and bias: New constructs and adaptation of the systematic test of equivalence procedure. *Current Psychology*, February 2021. ISSN 1046-1310, 1936-4733. doi: 10.1007/s12144-020-01256-4. URL <http://link.springer.com/10.1007/s12144-020-01256-4>.

Malcolm Gladwell. *Outliers: the story of success*. New York Times best sellers. Little, Brown and Company, New York, first edition. edition, 2008. ISBN 978-0-316-01792-3.

Samuel B. Green and Neil J. Salkind. *Using SPSS for Windows and Macintosh: analyzing and understanding data*. Pearson, Boston, eighth edition. edition, 2017. ISBN 978-0-13-431988-9.

Danica G. Hays, Catherine Y. Chang, and Scott L. Decker. Initial development and psychometric data for the Privilege and Oppression Inventory: Measurement and Evaluation in Counseling and Development. *Measurement and Evaluation in Counseling and Development*, 40(2):66–79, July 2007. ISSN 0748-1756. doi: 10.1080/07481756.2007.11909806. URL <https://ezproxy.spu.edu/login?url=https://search.ebscohost.com/login.aspx?direct=true&AuthType=ip&db=psyh&AN=2007-10909-001&site=ehost-live>. Publisher: American Counseling Assn.

Edward G. Helm, William E. Sedlacek, and Dario O. Prieto. The Relationship Between Attitudes Toward Diversity and Overall Satisfaction of University Students by Race. *Journal of College Counseling*, 1(2):111–120, September 1998. ISSN 1099-0399, 2161-1882. doi: 10.1002/j.2161-1882.1998.tb00130.x. URL <https://onlinelibrary.wiley.com/doi/10.1002/j.2161-1882.1998.tb00130.x>.

G. M. Herek. Documenting prejudice against lesbians and gay men on campus: the Yale Sexual Orientation Survey. *Journal of Homosexuality*, 25(4):15–30, 1993. ISSN 0091-8369. doi: 10.1300/J082v25n04\_02.

Clara E. Hill. *Helping skills: facilitating exploration, insight, and action*. American Psychological Association, Washington, DC, fifth edition. edition, 2020. ISBN 978-1-4338-3137-9.

Gerrit Hirschfeld and von Brachel, Ruth. Multiple-Group confirmatory factor analysis in R – A tutorial in measurement invariance with continuous and ordinal indicators. 19(7):12, 2014.

Li-tze Hu and Peter M. Bentler. Cutoff criteria for fit indexes in covariance structure analysis: Conventional criteria versus new alternatives. *Structural Equation Modeling*, 6(1):1–55, 1999. ISSN 1070-5511. doi: 10.1080/10705519909540118. URL <https://ezproxy.spu.edu/login?url=https://search.ebscohost.com/login.aspx?direct=true&AuthType=ip&db=psyh&AN=1998-03102-001&site=ehost-live>. Publisher: Lawrence Erlbaum.

Sylvia Hurtado. Linking Diversity with the Educational and Civic Missions of Higher Education. *Review of Higher Education: Journal of the Association for the Study of Higher Education*, 30(2):185–196, 2007. ISSN 0162-5748. doi: 10.1353/rhe.2006.0070. URL <https://ezproxy.spu.edu/login?url=http://search.ebscohost.com/login.aspx?direct=true&AuthType=ip&db=psyh&AN=2006-23268-004&site=ehost-live>. Publisher: Johns Hopkins University Press.

Juliette M. Iacovino and Sherman A. James. Retaining Students of Color in Higher Education: Expanding Our Focus to Psychosocial Adjustment and Mental Health. In *The Crisis of Race in Higher Education: A Day of Discovery and Dialogue*, volume 19 of *Diversity in Higher Education*, pages 61–84. Emerald Group Publishing Limited, January 2016. ISBN 978-1-78635-710-6 978-1-78635-709-0. doi: 10.1108/S1479-364420160000019004. URL <https://doi.org/10.1108/S1479-364420160000019004>.

- Rajiv S. Jhangiani, I.-Chant A. Chiang, Carrie Cuttler, and Dana C. Leighton. *Research Methods in Psychology*. August 2019. ISBN 978-1-9991981-0-7. doi: 10.17605/OSF.IO/HF7DQ. URL <https://kpu.pressbooks.pub/psychmethods4e/>.
- D. A. Kenny. SEM: Identification, 2012. URL [http://davidakenny.net/cm/identify\\_formal.htm](http://davidakenny.net/cm/identify_formal.htm).
- Brian TaeHyuk Keum, Jennifer L. Brady, Rajni Sharma, Yun Lu, Young Hwa Kim, and Christina J. Thai. Gendered Racial Microaggressions Scale for Asian American Women: Development and initial validation. *Journal of Counseling Psychology*, 65(5):571–585, October 2018. ISSN 0022-0167. doi: 10.1037/cou0000305. URL <https://ezproxy.spu.edu/login?url=https://search.ebscohost.com/login.aspx?direct=true&AuthType=ip&db=psyh&AN=2018-36321-001&site=ehost-live>.
- Paul Kim. Yes, Asians and Asian Americans experience racism, March 2021a. URL <https://www.seattletimes.com/opinion/yes-asians-and-asian-americans-experience-racism/>. Section: Opinion.
- Paul Y. Kim. Guest Post: Anti-Asian Racism during the Pandemic: How Faculty on Christian Campuses Can Support Asian and Asian American Students, March 2021b. URL <https://christianscholars.com/guest-post-anti-asian-racism-during-the-pandemic-how-faculty-on-christian-campuses-can-support-asian-and-asian-american-students/>.
- Rex B. Kline. *Principles and practice of structural equation modeling*. Guilford Publications, New York, UNITED STATES, 4th edition, 2016. ISBN 978-1-4625-2336-8. URL <http://ebookcentral.proquest.com/lib/spu/detail.action?docID=4000663>.
- Elizabeth A. Klonoff and Hope Landrine. The Schedule of Sexist Events: A measure of lifetime and recent sexist discrimination in women's lives. *Psychology of Women Quarterly*, 19(4):439–472, December 1995. ISSN 0361-6843. doi: 10.1111/j.1471-6402.1995.tb00086.x. Publisher: Blackwell Publishing.
- David R. Krathwohl. *Methods of Educational and Social Science Research: The Logic of Methods, Third Edition*. Waveland Press, February 2009. ISBN 978-1-4786-3425-6. Google-Books-ID: 0lMbDQAQBAJ.
- K. Kroenke, R. L. Spitzer, and J. B. Williams. The PHQ-9: validity of a brief depression severity measure. *Journal of General Internal Medicine*, 16(9):606–613, September 2001. ISSN 0884-8734. doi: 10.1046/j.1525-1497.2001.016009606.x.
- B. Andi Lee and Helen A. Neville. The iBelong Scale: Construction and validation of a measure of racial–ethnic–cultural belonging. *Journal of Counseling Psychology*, 71(3):139–154, April 2024. ISSN 0022-0167. doi: 10.1037/cou0000716. URL <https://ezproxy.spu.edu/login?url=https://search.ebscohost.com/login.aspx?direct=true&AuthType=ip&db=psyh&AN=2024-48470-001&site=ehost-live>. Publisher: American Psychological Association.
- Jioni A. Lewis and Helen A. Neville. Construction and initial validation of the Gendered Racial Microaggressions Scale for Black women. *Journal of Counseling Psychology*, 62(2):289–302, April 2015. ISSN 0022-0167. doi: 10.1037/cou0000062. URL <https://ezproxy.spu.edu/login?url=https://search.ebscohost.com/login.aspx?direct=true&AuthType=ip&db=psyh&AN=2015-15500-004&site=ehost-live>. Publisher: American Psychological Association.

- Kaleea R. Lewis and Payal P. Shah. Black students' narratives of diversity and inclusion initiatives and the campus racial climate: An interest-convergence analysis. *Journal of Diversity in Higher Education*, October 2019. ISSN 1938-8926. doi: 10.1037/dhe0000147. URL <https://ezproxy.spu.edu/login?url=https://search.ebscohost.com/login.aspx?direct=true&AuthType=ip&db=psyh&AN=2019-59378-001&site=ehost-live>. Publisher: Educational Publishing Foundation.
- Roderick J. A. Little and Donald B. Rubin. *Statistical analysis with missing data*. Wiley, Hoboken, second edition. edition, 2002. ISBN 978-1-118-62586-6. URL <http://site.ebrary.com/id/10921256>.
- Juan M. Madera, Eden B. King, and Michelle R. Hebl. Bringing social identity to work: The influence of manifestation and suppression on perceived discrimination, job satisfaction, and turnover intentions: Cultural Diversity and Ethnic Minority Psychology. *Cultural Diversity and Ethnic Minority Psychology*, 18(2):165–170, April 2012. ISSN 1099-9809. doi: 10.1037/a0027724. URL <https://ezproxy.spu.edu/login?url=https://search.ebscohost.com/login.aspx?direct=true&AuthType=ip&db=psyh&AN=2012-09819-006&site=ehost-live>. Publisher: Educational Publishing Foundation.
- Brent Mallinckrodt, Joseph R. Miles, and Jacob J. Levy. The scientist-practitioner-advocate model: Addressing contemporary training needs for social justice advocacy. *Training and Education in Professional Psychology*, 8(4):303–311, November 2014. ISSN 1931-3918. doi: 10.1037/tep0000045. URL <https://ezproxy.spu.edu/login?url=https://search.ebscohost.com/login.aspx?direct=true&AuthType=ip&db=psyh&AN=2014-25072-001&site=ehost-live>. Publisher: Educational Publishing Foundation.
- Glenn P. Malone, David R. Pillow, and Augustine Osman. The General Belongingness Scale (GBS): Assessing achieved belongingness. *Personality and Individual Differences*, 52(3):311–316, February 2012. ISSN 0191-8869. doi: 10.1016/j.paid.2011.10.027. URL <https://www.sciencedirect.com/science/article/pii/S019188691100482X>.
- Daniel McNeish. Thanks coefficient alpha, we'll take it from here. *Psychological Methods*, 23(3):412–433, September 2018. ISSN 1082-989X. doi: 10.1037/met0000144. URL <https://ezproxy.spu.edu/login?url=https://search.ebscohost.com/login.aspx?direct=true&AuthType=ip&db=psyh&AN=2017-23572-001&site=ehost-live>. Publisher: American Psychological Association.
- Samuel Messick. Consequences of test interpretation and use: The fusion of validity and values in psychological assessment. In Richard D. Goffin and Edward Helmes, editors, *Problems and solutions in human assessment: Honoring Douglas N. Jackson at seventy.*, pages 3–20. Kluwer Academic/Plenum Publishers, New York, NY, 2000. ISBN 978-0-7923-7768-9. doi: 10.1007/978-1-4615-4397-8\_1. URL <https://ezproxy.spu.edu/login?url=https://search.ebscohost.com/login.aspx?direct=true&AuthType=ip&db=psyh&AN=2001-16052-001&site=ehost-live>.
- Della V. Mosley, Helen A. Neville, Nayeli Y. Chavez-Dueñas, Hector Y. Adames, Jioni A. Lewis, and Bryana H. French. Radical hope in revolting times: Proposing a culturally relevant psychological framework. *Social and Personality Psychology Compass*, 14(1), January 2020. ISSN 1751-9004. doi: 10.1111/spc.12512. Publisher: Wiley-Blackwell Publishing Ltd.
- Della V. Mosley, Candice N. Hargons, Carolyn Meiller, Blanka Angyal, Paris Wheeler, Candice Davis, and Danelle Stevens-Watkins. Critical consciousness of anti-Black

- racism: A practical model to prevent and resist racial trauma. *Journal of Counseling Psychology*, 68(1):1–16, January 2021. ISSN 0022-0167. doi: 10.1037/cou0000430. URL <https://ezproxy.spu.edu/login?url=https://search.ebscohost.com/login.aspx?direct=true&AuthType=ip&db=psyh&AN=2020-20397-001&site=ehost-live>. Publisher: American Psychological Association.
- Sean P. Mullen, Neha P. Gothe, and Edward McAuley. Evaluation of the Factor Structure of the Rosenberg Self-Esteem Scale in Older Adults. *Personality and individual differences*, 54(2):153–157, January 2013. ISSN 0191-8869. doi: 10.1016/j.paid.2012.08.009. URL <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3505083/>.
- Hector Najera Catalan. Reliability, Population Classification and Weighting in Multidimensional Poverty Measurement: A Monte Carlo Study. *Social Indicators Research*, 142, April 2019. doi: 10.1007/s11205-018-1950-z.
- Helen A. Neville, Roderick L. Lilly, Georgia Duran, Richard M. Lee, and LaVonne Browne. Construction and initial validation of the Color-Blind Racial Attitudes Scale (CoBRAS): Journal of Counseling Psychology. *Journal of Counseling Psychology*, 47(1):59–70, January 2000. ISSN 0022-0167. doi: 10.1037/0022-0167.47.1.59. URL <https://ezproxy.spu.edu/login?url=https://search.ebscohost.com/login.aspx?direct=true&AuthType=ip&db=psyh&AN=2000-13659-006&site=ehost-live>. Publisher: American Psychological Association.
- Thomas A. Norton, Hannes Zacher, Stacey L. Parker, and Neal M. Ashkanasy. Bridging the gap between green behavioral intentions and employee green behavior: The role of green psychological climate: Journal of Organizational Behavior. *Journal of Organizational Behavior*, 38(7):996–1015, September 2017. ISSN 0894-3796. doi: 10.1002/job.2178. URL <https://ezproxy.spu.edu/login?url=https://search.ebscohost.com/login.aspx?direct=true&AuthType=ip&db=psyh&AN=2017-01910-001&site=ehost-live>. Publisher: John Wiley & Sons.
- Mike C. Parent. Handling item-level missing data: Simpler is just as good. *The Counseling Psychologist*, 41(4):568–600, May 2013. ISSN 0011-0000, 1552-3861. doi: 10.1177/0011000012445176. URL <http://journals.sagepub.com/doi/10.1177/0011000012445176>.
- Ernest T. Pascarella and Patrick T. Terenzini. Predicting Freshman Persistence and Voluntary Dropout Decisions from a Theoretical Model. *The Journal of Higher Education*, 51(1):60–75, January 1980. ISSN 0022-1546, 1538-4640. doi: 10.1080/00221546.1980.11780030. URL <https://www.tandfonline.com/doi/full/10.1080/00221546.1980.11780030>.
- Elazar J. Pedhazur and Liora Pedhazur Schmelkin. *Measurement, design, and analysis: an integrated approach*. Lawrence Erlbaum Associates, Hillsdale, N.J., 1991. ISBN 978-0-89859-555-0.
- James A. Pershing and Jana L. Pershing. Ineffective reaction evaluation. *Human Resource Development Quarterly*, 12(1):73–90, 2001. ISSN 1532-1096. doi: 10.1002/1532-1096(200101/02)12:1<73::AID-HRDQ6>3.0.CO;2-7. URL <https://onlinelibrary.wiley.com/doi/abs/10.1002/1532-1096%28200101/02%2912%3A1%3C73%3A%3AAID-HRDQ6%3E3.0.CO%3B2-7>.
- Alex L. Pieterse, Katherine Kirkinis, Lynsay Paiko, and Matthew Miller. The Anti-Racism Behavioral Inventory: A Validation Study. *Journal for Social Action in Counseling & Psychology*, 14(2):48–63, 2022. ISSN 2159-8142. doi: 10.33043/JSACP.14.2.48-63. URL <https://openjournals.bsu.edu/jsacp/article/view/3679>. Number: 2.

- Courtney A. Resnick and M. Paz Galupo. Assessing experiences with LGBT microaggressions in the workplace: Development and validation of the Microaggression Experiences at Work Scale: Journal of Homosexuality. *Journal of Homosexuality*, 66(10):1380–1403, 2019. ISSN 0091-8369. doi: 10.1080/00918369.2018.1542207. Publisher: Taylor & Francis.
- William Revelle. An introduction to the psych package: Part II Scale construction and psychometrics. page 97.
- William Revelle and David M. Condon. Reliability from to : A tutorial. Online supplement. *Psychological Assessment*, August 2019a. ISSN 1939-134X, 1040-3590. doi: 10.1037/pas0000754. URL <http://doi.apa.org/getdoi.cfm?doi=10.1037/pas0000754>.
- William Revelle and David M. Condon. Reliability from to : A tutorial. *Psychological Assessment*, August 2019b. ISSN 1939-134X, 1040-3590. doi: 10.1037/pas0000754. URL <http://doi.apa.org/getdoi.cfm?doi=10.1037/pas0000754>.
- Revelle, William. The Personality Project: An introduction to psychometric theory. URL <http://www.personality-project.org/r/book/#chapter7>.
- Michael T. Roberson and Eric Sundstrom. Questionnaire design, return rates, and response favorableness in an employee attitude questionnaire. *Journal of Applied Psychology*, 75(3):354–357, 1990. ISSN 1939-1854(Electronic),0021-9010(Print). doi: 10.1037/0021-9010.75.3.354.
- Robert G. Green, Katrina D. Murphy, and Shelita M. Snyder. Should demographics be placed at the end or at the beginning of a mailed questionnaires? An empirical answer to a persistent methodological question. *Social Work Research*, 24(4):237–240, December 2000.
- Jenny Rowley. Designing and using research questionnaires. *Management Research Review*, March 2014. ISSN 2040-8269. doi: 10.1108/MRR-02-2013-0027. URL <https://www.emerald.com/insight/content/doi/10.1108/MRR-02-2013-0027/full/html>.
- John Schmid and John M. Leiman. The development of hierarchical factor solutions. *Psychometrika*, 22:53–61, 1957. ISSN 1860-0980. doi: 10.1007/BF02289209. Place: Germany Publisher: Springer.
- Michael F. Schober and Frederick G. Conrad. Survey Interviews and New Communication Technologies. In Frederick G. Conrad and Michael F. Schober, editors, *Envisioning the Survey Interview of the Future*, pages 1–30. John Wiley & Sons, Inc., Hoboken, NJ, USA, November 2007. ISBN 978-0-470-18337-3 978-0-471-78627-6. doi: 10.1002/9780470183373.ch1. URL <http://doi.wiley.com/10.1002/9780470183373.ch1>.
- Norbert Schwarz and Daphna Oyserman. Asking Questions About Behavior: Cognition, Communication, and Questionnaire Construction. *AMERICAN JOURNAL OF EVALUATION*, page 34, 2001.
- Anneliese Singh. Building a Counseling Psychology of Liberation: The Path Behind Us, Under Us, and Before Us. *The Counseling Psychologist*, 48(8):1109–1130, November 2020. ISSN 0011-0000, 1552-3861. doi: 10.1177/0011000020959007. URL <http://journals.sagepub.com/doi/10.1177/0011000020959007>.
- Robert L. Spitzer, Kurt Kroenke, Janet B. W. Williams, and Bernd Löwe. A Brief Measure for Assessing Generalized Anxiety Disorder: The GAD-7. *Archives of Internal Medicine*, 166(10):1092, May 2006. ISSN 0003-9926. doi: 10.1001/archinte.166.10.1092. URL <http://archinte.jamanetwork.com/article.aspx?doi=10.1001/archinte.166.10.1092>.

Dawn M. Szymanski and Danielle Bissonette. Perceptions of the LGBTQ College Campus Climate Scale: Development and psychometric evaluation. *Journal of Homosexuality*, 67(10):1412–1428, August 2020. ISSN 0091-8369, 1540-3602. doi: 10.1080/00918369.2019.1591788. URL <https://www.tandfonline.com/doi/full/10.1080/00918369.2019.1591788>.

Susan R. Torres-Harding, Alejandro L. Jr. Andrade, and Crist E. Romero Diaz. The Racial Microaggressions Scale (RMAS): A new scale to measure experiences of racial microaggressions in people of color: Cultural Diversity and Ethnic Minority Psychology. *Cultural Diversity and Ethnic Minority Psychology*, 18(2):153–164, April 2012. ISSN 1099-9809. doi: 10.1037/a0027658. URL <https://ezproxy.spu.edu/login?url=https://search.ebscohost.com/login.aspx?direct=true&AuthType=ip&db=psyh&AN=2012-09819-005&sit=ehost-live>. Publisher: Educational Publishing Foundation.