COS-D419 Factor Analysis and Structural Equation Models 2023, Assignment 3

Rong Guang

2023-02-03

Contents

1 Preparation **1**
- 1.1 Read in the data set 1
- 1.2 Write functions 1
- 1.3 Inspect the data 1
- 1.4 Add variable labels 6

2 Factorial validity **8**
- 2.1 Define and estimate a CFA model 8

3 Vsiualize the CFA model **14**

4 Finetune model base on Modification Indices **16**
- 4.1 MI-based finetuning on initial model 16
- 4.2 MI-based finetuning on model 2 18
- 4.3 MI-based finetuning on model 3 20
- 4.4 MI-based finetuning on model 4 22
- 4.5 Summarize the models fitted 23
- 4.6 Draw the graph of the final model 24

5 More exploration **27**

1 Preparation

1.1 Read in the data set

```
library(tidyverse);library(readr);mbi <- read_csv("ELEMM1.CSV", show_col_types = FALSE)
```

1.2 Write functions

To control length of reports, codes already shown in the previous homework were not showing in the current report. Yet they are available in .rmd report.

1.2.1 to check unique values

1.2.2 to generate CFA results with improved readability

1.2.3 to generate a function for correlation matrix with numbers

1.2.4 to generate a function for histogram overlapping with density plot

1.2.5 to generate a function for violin overlapping with box plot

1.2.6 To generate a function describing continuous data set

1.3 Inspect the data

1.3.1 Data structure

Have a quick overview of the data structure

```
dim(mbi);mbi %>% apply(2, function(x)class(x));unique.levels(mbi)
```

```
## [1] 372  22
```

```
##     ITEM1     ITEM2     ITEM3     ITEM4     ITEM5     ITEM6     ITEM7     ITEM8
## "numeric" "numeric" "numeric" "numeric" "numeric" "numeric" "numeric" "numeric"
##     ITEM9    ITEM10    ITEM11    ITEM12    ITEM13    ITEM14    ITEM15    ITEM16
## "numeric" "numeric" "numeric" "numeric" "numeric" "numeric" "numeric" "numeric"
##    ITEM17    ITEM18    ITEM19    ITEM20    ITEM21    ITEM22
## "numeric" "numeric" "numeric" "numeric" "numeric" "numeric"
```

```
## [1] "Variable ITEM1 has values of 1,2,3,4,5,6,7"
## [1] "Variable ITEM2 has values of 1,2,3,4,5,6,7"
## [1] "Variable ITEM3 has values of 1,2,3,4,5,6,7"
## [1] "Variable ITEM4 has values of 2,3,4,5,6,7"
## [1] "Variable ITEM5 has values of 1,2,3,4,5,6,7"
## [1] "Variable ITEM6 has values of 1,2,3,4,5,6,7"
## [1] "Variable ITEM7 has values of 2,3,4,5,6,7"
## [1] "Variable ITEM8 has values of 1,2,3,4,5,6,7"
## [1] "Variable ITEM9 has values of 1,2,3,4,5,6,7"
## [1] "Variable ITEM10 has values of 1,2,3,4,5,6,7"
## [1] "Variable ITEM11 has values of 1,2,3,4,5,6,7"
## [1] "Variable ITEM12 has values of 1,2,3,4,5,6,7"
## [1] "Variable ITEM13 has values of 1,2,3,4,5,6,7"
## [1] "Variable ITEM14 has values of 1,2,3,4,5,6,7"
## [1] "Variable ITEM15 has values of 1,2,3,4,5,6,7"
## [1] "Variable ITEM16 has values of 1,2,3,4,5,6,7"
## [1] "Variable ITEM17 has values of 2,3,4,5,6,7"
## [1] "Variable ITEM18 has values of 1,2,3,4,5,6,7"
```

```
## [1] "Variable ITEM19 has values of 1,2,3,4,5,6,7"
## [1] "Variable ITEM20 has values of 1,2,3,4,5,6,7"
## [1] "Variable ITEM21 has values of 2,3,4,5,6,7"
## [1] "Variable ITEM22 has values of 1,2,3,4,5,6,7"
```

The data set contains 22 numeric variables of 372 obs. Their values appear to follow a consistent pattern covering the integer from 1 to 7, except for Items 4, 7, 17 and 21, which did not include a value of 1.

1.3.2 Descriptive statistics of measured variables

```
library(finalfit);library(kableExtra);descriptive(mbi)
```

Table 1: Descriptive statistics for measurements

			Central tendency		Dispersion tendency			
	n	n of NA	Mean	Median	SD	Min	Max	Q1~Q3
ITEM1	372	0	4.4	4.0	1.7	1.0	7.0	3.0 ~ 6.0
ITEM2	372	0	4.9	5.0	1.5	1.0	7.0	4.0 ~ 6.0
ITEM3	372	0	3.5	3.0	1.7	1.0	7.0	2.0 ~ 5.0
ITEM4	372	0	6.3	7.0	1.0	2.0	7.0	6.0 ~ 7.0
ITEM5	372	0	2.2	2.0	1.5	1.0	7.0	1.0 ~ 3.0
ITEM6	372	0	2.7	2.0	1.6	1.0	7.0	2.0 ~ 4.0
ITEM7	372	0	6.3	6.0	0.8	2.0	7.0	6.0 ~ 7.0
ITEM8	372	0	3.0	2.0	1.7	1.0	7.0	2.0 ~ 4.0
ITEM9	372	0	6.0	7.0	1.3	1.0	7.0	6.0 ~ 7.0
ITEM10	372	0	2.2	2.0	1.4	1.0	7.0	1.0 ~ 3.0
ITEM11	372	0	2.2	2.0	1.5	1.0	7.0	1.0 ~ 3.0
ITEM12	372	0	5.7	6.0	1.2	1.0	7.0	5.0 ~ 6.0
ITEM13	372	0	3.6	3.5	1.7	1.0	7.0	2.0 ~ 5.0
ITEM14	372	0	4.0	4.0	1.7	1.0	7.0	3.0 ~ 5.0
ITEM15	372	0	1.8	1.0	1.3	1.0	7.0	1.0 ~ 2.0
ITEM16	372	0	2.5	2.0	1.4	1.0	7.0	1.0 ~ 3.0
ITEM17	372	0	6.4	7.0	0.9	2.0	7.0	6.0 ~ 7.0
ITEM18	372	0	5.7	6.0	1.3	1.0	7.0	5.0 ~ 7.0
ITEM19	372	0	5.9	6.0	1.2	1.0	7.0	6.0 ~ 7.0
ITEM20	372	0	2.2	2.0	1.4	1.0	7.0	1.0 ~ 3.0
ITEM21	372	0	5.9	6.0	1.3	2.0	7.0	5.0 ~ 7.0
ITEM22	372	0	2.6	2.0	1.6	1.0	7.0	1.0 ~ 3.0

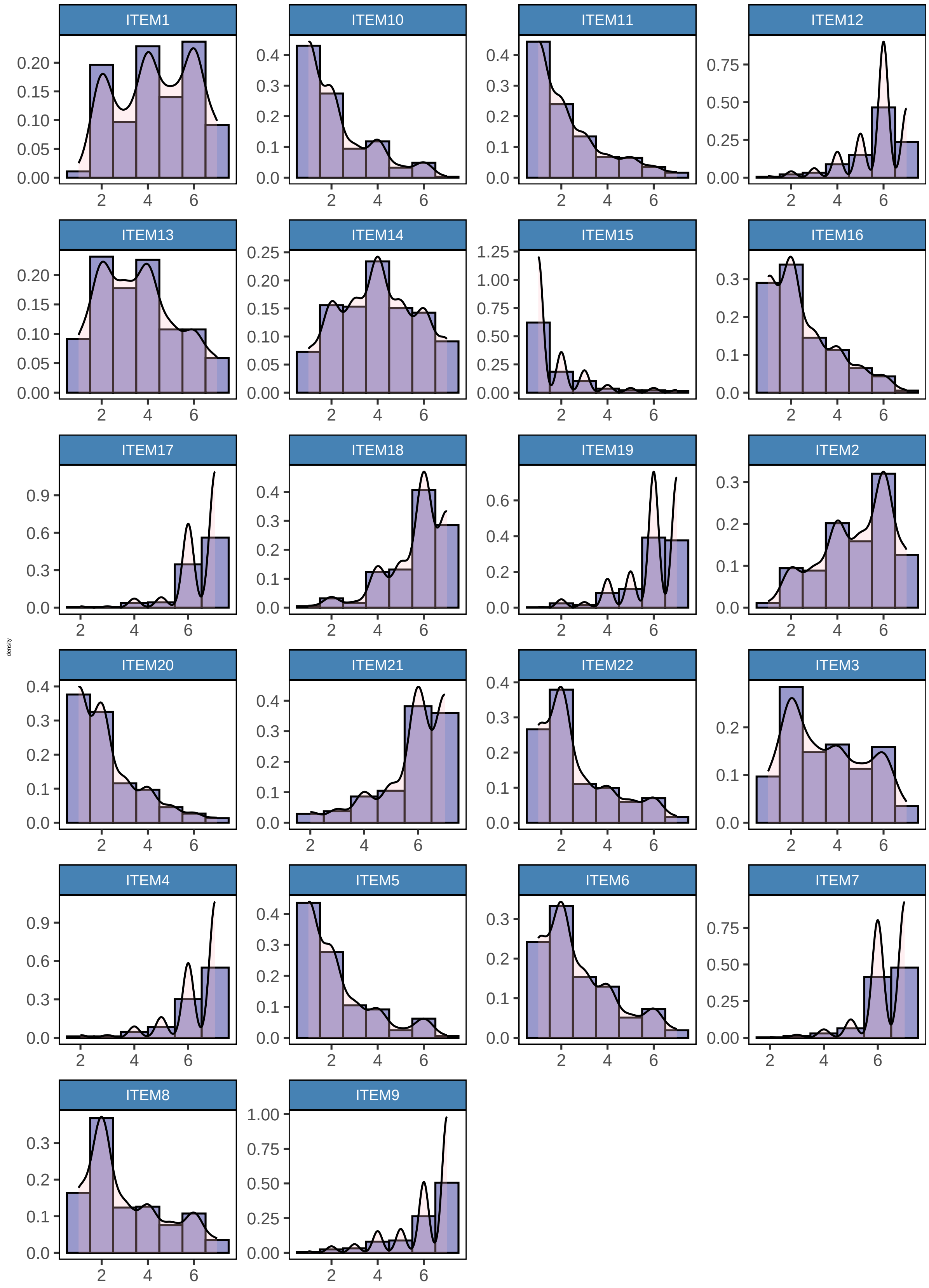
1.3.3 Visualization

(1) Histogram

Distribution of the data was examined via Histogram

```
corr.density(mbi, fig.num = 1)
```

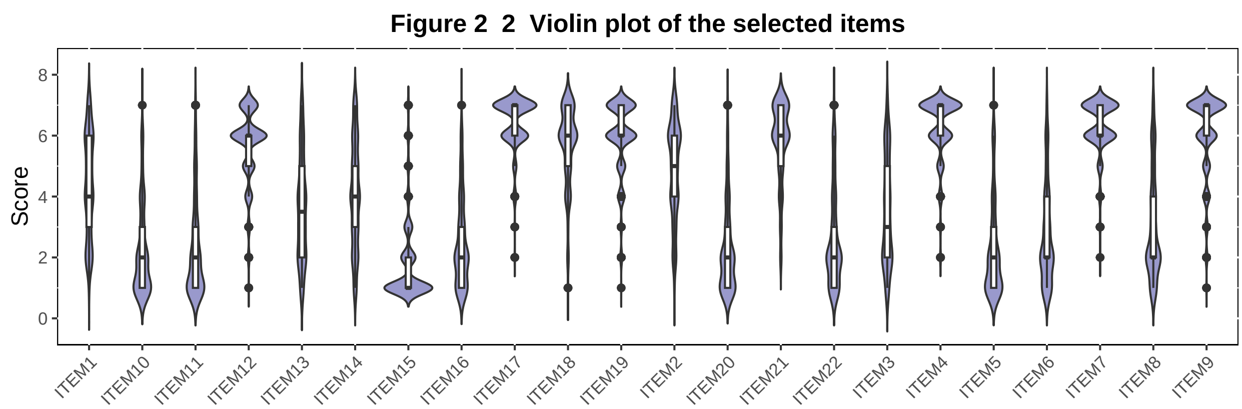
Figure 1 Distribution of selected items



(2) Violin plot

Violin plot also provides information on distribution, plus ideas on out-liers.

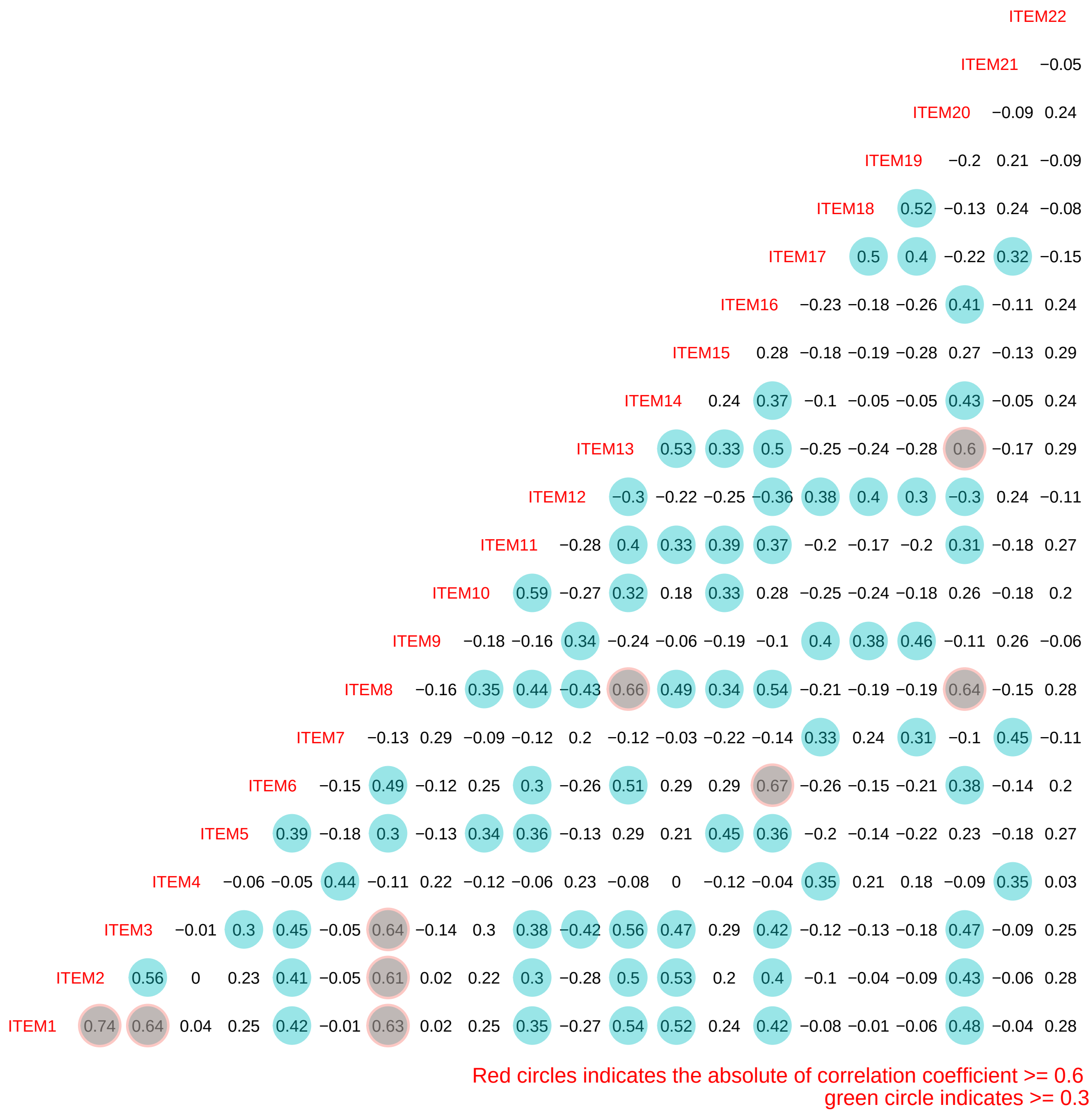
```
violin.box(mbi, fig.num = 2)
```



(3) Correlation among items

```
mymatrix(mbi, fig.num = 3)
```

Figure 3 Pearson correlation matrix of the selected items



All variables had a pearson correlation coefficient >0.3 with at least one other variable, except for ITEM22.

1.4 Add variable labels

```
library(expss)
mbi <-  apply_labels(mbi,
                     ITEM1 = "I feel emotionally drained from my work",
                     ITEM2 = "I feel used up at the end of the workday.",
                     ITEM3 = "I feel fatigued when I get up in the morning and
                     have to faceanother day on the job.",
                     ITEM4 = "I can easily understand how my students feel
                     about things.",
```

```
                       ITEM5 = "I feel I treat some students as if they were
                       impersonal objects.",
                       ITEM6 = "Working with people all day is really a
                       strain for me.",
                       ITEM7 = "I deal very effectively with the problems of
                       my students.",
                       ITEM8 = "I feel burned out from my work.",
                       ITEM9 = "I feel I'm positively influencing other people's
                       lives through my work.",
                       ITEM10 = "I've become more callous toward people since I
                       took this job.",
                       ITEM11 = "I worry that this job is hardening me
                       emotionally.",
                       ITEM12 = "I feel very energetic.",
                       ITEM13 = " I feel frustrated by my job.",
                       ITEM14 = "I feel I'm working too hard on my job.",
                       ITEM15 = "I don't care what happens to some students.",
                       ITEM16 = "Working with people directly puts too much
                       stress on me.",
                       ITEM17 = " I can easily create a relaxed atmosphere
                       with my students.",
                       ITEM18 = " I feel exhilarated after working closely
                       with my students.",
                       ITEM19 = "I have accomplished many worthwhile things in
                       this job.",
                       ITEM20 = "I feel like I'm at the end of my rope.",
                       ITEM21 = "In my work, I deal with emotional problems
                       very calmly.",
                       ITEM22 = "I feel students blame me for some of their
                       problems."
                       )
```

```
library(sjlabelled)
get_label(mbi) %>% kable()
```

	x
ITEM1	I feel emotionally drained from my work
ITEM2	I feel used up at the end of the workday.
ITEM3	I feel fatigued when I get up in the morning and have to faceanother day on the job.
ITEM4	I can easily understand how my students feel about things.
ITEM5	I feel I treat some students as if they were impersonal objects.
ITEM6	Working with people all day is really a strain for me.
ITEM7	I deal very effectively with the problems of my students.
ITEM8	I feel burned out from my work.
ITEM9	I feel I'm positively influencing other people's lives through my work.
ITEM10	I've become more callous toward people since I took this job.
ITEM11	I worry that this job is hardening me emotionally.
ITEM12	I feel very energetic.
ITEM13	I feel frustrated by my job.
ITEM14	I feel I'm working too hard on my job.
ITEM15	I don't care what happens to some students.
ITEM16	Working with people directly puts too much stress on me.
ITEM17	I can easily create a relaxed atmosphere with my students.
ITEM18	I feel exhilarated after working closely with my students.
ITEM19	I have accomplished many worthwhile things in this job.
ITEM20	I feel like I'm at the end of my rope.
ITEM21	In my work, I deal with emotional problems very calmly.
ITEM22	I feel students blame me for some of their problems.

2 Factorial validity

This is to test for the factorial validity of the MBI for elementary teachers using a confirmatory factor analytic approach.

2.1 Define and estimate a CFA model

This report started by estimating the factorial validity of initially postulated model. As was stated by Byrne in the 1991 study:

"The CFA model in the present study hypothesized a priori that: (a) responses to the MBI could be explained by three factors, (b) each item would have a non-zero loading on the burnout factor it was designed to measure, and zero loadings on all other factors, (c) the three factors would be correlated and, (d) the error-uniqueness terms for the item variables would be uncorrelated."

2.1.1 Hypothesis testing via different estimation

(1) ML estimation

```
library(lavaan);library(kableExtra)
#define model
model1 <- '
# CFA model for the burnout:
# EE: EmotionalExhaustion
# DP: Depersonalization
# PA: PersonalAccomplishment
  EE =~ ITEM1 + ITEM2 + ITEM3 + ITEM6 + ITEM8 +
```

```
        ITEM13 + ITEM14 + ITEM16 + ITEM20
  DP =~ ITEM5 + ITEM10 + ITEM11 + ITEM15 +ITEM22
  PA =~ ITEM4 + ITEM7 + ITEM9 + ITEM12 +
        ITEM17 + ITEM18 + ITEM19 + ITEM21
'
# Estimate the model with the default (ML) estimator:
cfa1 <- cfa(model1, data = mbi)

# Numerical summary of the model:
cfa.summary.ml.a(cfa1, 3, 22, "ML");cfa.summary.b(cfa1, 3, 22, "ML");
```

Table 2: Goodness-of-fit and subjective indices of fit for 3 factor model ML

Measure	Value
chi square	695.719
df	206.000
p value	0.000
CFI	0.848
TLI	0.830
RMSEA	0.080
RMSEA p value	0.000
SRMR	0.073

Table 3: Factor Loadings for 3 factor CFA model estimated by ML

Latent Factor	Indicator	B	SE	Z	p-value	Beta
EE	ITEM1	1.000	0.000	NA	NA	0.768
EE	ITEM2	0.887	0.061	14.621	0	0.732
EE	ITEM3	1.021	0.068	15.085	0	0.752
EE	ITEM6	0.764	0.064	12.013	0	0.616
EE	ITEM8	1.143	0.066	17.299	0	0.845
EE	ITEM13	1.017	0.065	15.544	0	0.772
EE	ITEM14	0.848	0.069	12.251	0	0.627
EE	ITEM16	0.715	0.058	12.410	0	0.634
EE	ITEM20	0.753	0.056	13.410	0	0.679
DP	ITEM5	1.000	0.000	NA	NA	0.565
DP	ITEM10	1.142	0.127	8.986	0	0.663
DP	ITEM11	1.353	0.142	9.511	0	0.743
DP	ITEM15	0.905	0.109	8.318	0	0.586
DP	ITEM22	0.768	0.121	6.361	0	0.408
PA	ITEM4	1.000	0.000	NA	NA	0.440
PA	ITEM7	0.970	0.150	6.482	0	0.507
PA	ITEM9	1.780	0.254	7.007	0	0.594
PA	ITEM12	1.499	0.221	6.769	0	0.552
PA	ITEM17	1.348	0.181	7.463	0	0.695
PA	ITEM18	1.918	0.262	7.329	0	0.662
PA	ITEM19	1.716	0.238	7.205	0	0.634
PA	ITEM21	1.356	0.218	6.219	0	0.471

```r
cfa.summary.c(cfa1, 3, 22, "ML");cfa.summary.d(cfa1, 3, 22, "ML")
```

Table 4: Variances for 3 factor model estimated by ML

Type	Indicator	B	SE	Z	p-value	Beta
Residual variance	ITEM1	1.128	0.095	11.861	0	0.410
Residual variance	ITEM2	1.105	0.090	12.214	0	0.464
Residual variance	ITEM3	1.301	0.108	12.031	0	0.434
Residual variance	ITEM6	1.553	0.121	12.888	0	0.621
Residual variance	ITEM8	0.852	0.081	10.553	0	0.286
Residual variance	ITEM13	1.142	0.097	11.821	0	0.404
Residual variance	ITEM14	1.804	0.140	12.844	0	0.607
Residual variance	ITEM16	1.235	0.096	12.812	0	0.598
Residual variance	ITEM20	1.075	0.085	12.585	0	0.539
Residual variance	ITEM5	1.503	0.125	12.026	0	0.681
Residual variance	ITEM10	1.169	0.107	10.901	0	0.560
Residual variance	ITEM11	1.044	0.112	9.330	0	0.447
Residual variance	ITEM15	1.106	0.093	11.838	0	0.657
Residual variance	ITEM22	2.076	0.160	12.958	0	0.833
Residual variance	ITEM4	0.802	0.062	12.901	0	0.806
Residual variance	ITEM7	0.523	0.042	12.572	0	0.743
Residual variance	ITEM9	1.117	0.093	11.952	0	0.647
Residual variance	ITEM12	0.987	0.080	12.287	0	0.695
Residual variance	ITEM17	0.375	0.035	10.739	0	0.517
Residual variance	ITEM18	0.909	0.081	11.224	0	0.562
Residual variance	ITEM19	0.844	0.073	11.557	0	0.598
Residual variance	ITEM21	1.245	0.098	12.764	0	0.778
Total variance	EE	1.625	0.190	8.551	0	1.000
Total variance	DP	0.705	0.132	5.321	0	1.000
Total variance	PA	0.193	0.048	4.047	0	1.000

Table 5: Covariances for 3 factor model estimated by ML

Type	B	SE	Z	p-value	Beta
EE with DP	0.701	0.099	7.061	0	0.655
EE with PA	-0.192	0.042	-4.537	0	-0.343

(2) MLM estimation

```r
# Use a robust (MLM) estimator:
cfa2 <- cfa(model1, data = mbi, estimator = "MLM")

# Numerical summary of the model:
cfa.summary.mlm.a(cfa2, 3, 22, "MLM");cfa.summary.b(cfa2, 3, 22, "MLM");
```

Table 6: Goodness-of-fit and subjective indices of fit for 3 factor model MLM

Measure	Value
chi square	567.753
df	206.000
p value	0.000
CFI	0.865
TLI	0.849
RMSEA	0.069
RMSEA p value	0.000
SRMR	0.073
CSF	1.225

Table 7: Factor Loadings for 3 factor CFA model estimated by MLM

Latent Factor	Indicator	B	SE	Z	p-value	Beta
EE	ITEM1	1.000	0.000	NA	NA	0.768
EE	ITEM2	0.887	0.040	22.391	0	0.732
EE	ITEM3	1.021	0.053	19.310	0	0.752
EE	ITEM6	0.764	0.070	10.974	0	0.616
EE	ITEM8	1.143	0.059	19.366	0	0.845
EE	ITEM13	1.017	0.062	16.340	0	0.772
EE	ITEM14	0.848	0.058	14.584	0	0.627
EE	ITEM16	0.715	0.066	10.826	0	0.634
EE	ITEM20	0.753	0.061	12.303	0	0.679
DP	ITEM5	1.000	0.000	NA	NA	0.565
DP	ITEM10	1.142	0.152	7.509	0	0.663
DP	ITEM11	1.353	0.162	8.368	0	0.743
DP	ITEM15	0.905	0.123	7.366	0	0.586
DP	ITEM22	0.768	0.122	6.284	0	0.408
PA	ITEM4	1.000	0.000	NA	NA	0.440
PA	ITEM7	0.970	0.128	7.563	0	0.507
PA	ITEM9	1.780	0.322	5.529	0	0.594
PA	ITEM12	1.499	0.241	6.232	0	0.552
PA	ITEM17	1.348	0.200	6.757	0	0.695
PA	ITEM18	1.918	0.298	6.435	0	0.662
PA	ITEM19	1.716	0.287	5.978	0	0.634
PA	ITEM21	1.356	0.227	5.984	0	0.471

```
cfa.summary.c(cfa2, 3, 22, "MLM");cfa.summary.d(cfa2, 3, 22, "MLM")
```

Table 8: Variances for 3 factor model estimated by MLM

Type	Indicator	B	SE	Z	p-value	Beta
Residual variance	ITEM1	1.128	0.093	12.177	0	0.410
Residual variance	ITEM2	1.105	0.088	12.506	0	0.464
Residual variance	ITEM3	1.301	0.106	12.317	0	0.434
Residual variance	ITEM6	1.553	0.134	11.550	0	0.621

Type	Indicator	B	SE	Z	p-value	Beta
Residual variance	ITEM8	0.852	0.082	10.450	0	0.286
Residual variance	ITEM13	1.142	0.124	9.173	0	0.404
Residual variance	ITEM14	1.804	0.142	12.730	0	0.607
Residual variance	ITEM16	1.235	0.110	11.278	0	0.598
Residual variance	ITEM20	1.075	0.137	7.860	0	0.539
Residual variance	ITEM5	1.503	0.179	8.381	0	0.681
Residual variance	ITEM10	1.169	0.147	7.959	0	0.560
Residual variance	ITEM11	1.044	0.141	7.398	0	0.447
Residual variance	ITEM15	1.106	0.153	7.220	0	0.657
Residual variance	ITEM22	2.076	0.184	11.266	0	0.833
Residual variance	ITEM4	0.802	0.113	7.124	0	0.806
Residual variance	ITEM7	0.523	0.075	7.010	0	0.743
Residual variance	ITEM9	1.117	0.149	7.487	0	0.647
Residual variance	ITEM12	0.987	0.126	7.852	0	0.695
Residual variance	ITEM17	0.375	0.056	6.635	0	0.517
Residual variance	ITEM18	0.909	0.143	6.376	0	0.562
Residual variance	ITEM19	0.844	0.111	7.622	0	0.598
Residual variance	ITEM21	1.245	0.133	9.338	0	0.778
Total variance	EE	1.625	0.148	11.004	0	1.000
Total variance	DP	0.705	0.158	4.452	0	1.000
Total variance	PA	0.193	0.050	3.839	0	1.000

Table 9: Covariances for 3 factor model estimated by MLM

Type	B	SE	Z	p-value	Beta
EE with DP	0.701	0.106	6.608	0	0.655
EE with PA	-0.192	0.040	-4.796	0	-0.343

2.1.2 Results comparison (ML vs MLM)

```
options(scipen = 999)
ml.names <- c("chisq", "df", "pvalue","cfi", "tli","rmsea",
              "rmsea.ci.lower", "rmsea.ci.upper", "srmr")
mlm.names <- c("chisq.scaled", "df.scaled", "pvalue.scaled",
               "cfi.scaled", "tli.scaled","rmsea.scaled",
               "rmsea.ci.lower.robust", "rmsea.ci.upper.robust",
               "srmr", "chisq.scaling.factor")
merge.names <- c(ml.names, "csf") #csf is for shisq.scaling.factor
#obtain measures from ML estimation
ml.indicator <- fitMeasures(cfa1,    #obtain specified measured.
                            ml.names,
                            output = "matrix") %>%
                round(3)
colnames(ml.indicator) <- "ML"
ml.indicator <- rbind(ml.indicator, 999) %>% na_if(999)

#obtain measures from MLM estimation
mlm.indicator <- fitMeasures(cfa2,
```

```r
                             mlm.names,
                             output = "matrix") %>%
                 round(3)
colnames(mlm.indicator) <- "MLM"
rownames(ml.indicator) <- merge.names
rownames(mlm.indicator) <- merge.names

compare.tab <- data.frame(ML = ml.indicator,
                          MLM = mlm.indicator)

compare.tab <- compare.tab %>%
  mutate(contrast = MLM-ML) %>%
  round(3)

compare.tab <- as.data.frame(t(compare.tab))

compare.tab <- compare.tab %>%
  mutate(chisq.df.p = paste(chisq, "(", df, " ,", pvalue, ")"),
         rmsea.ci =
           paste(rmsea, "(", rmsea.ci.lower, ", ", rmsea.ci.upper, ")")) %>%
  select(chisq.df.p, cfi, tli, rmsea.ci, srmr, csf) %>%
  mutate(csf = csf %>% as.character(),
        csf = replace_na(csf, "--"))

colnames(compare.tab) <- c("chi-square (df,p)*", "CFI†", "TLI†",
                           "RMSEA(95%CI)*", "SRMR*", "CSF‡")

rownames(compare.tab) <- c("ML estimation", "MLM estimation",
                           "Estimator Contrast§")
compare.tab %>%
  kable(booktabs =T,
        linesep = "",
        align = "r",
        caption = "Comparison of fit indices between ML and MLM estimators") %>%
  kable_styling(full_width = TRUE) %>%
  column_spec(1, "3.2cm") %>%
  column_spec(2,"3cm") %>%
  column_spec(3, "0.7cm") %>%
  column_spec(4, "0.7cm") %>%
  column_spec(5, "3.8cm") %>%
  footnote(symbol = c("Smaller value indicates better fit",
                      "Larger value indicates better fit",
                      ">1 indicates the data violates normality assumption and MLM is a better estimato
                      "Constrast of results from estimations rather than direct model values: MLM-ML"))
```

According to the table, comparing to the statistics estimated by ML, MLM estimation provided smaller chi square (by 127), higher CFI (by 0.017), higher TLI by 0.19 and smaller RMSEA (by 0.011). The scaling correct factor is 1.225 (>1), indicating some violence of the normality assumption and MLM is more optimal a estimator for the data.

Table 10: Comparison of fit indices between ML and MLM estimators

	chi-square (df,p)*	CFI†	TLI†	RMSEA(95%CI)*	SRMR*	CSF‡
ML estimation	695.719 (206 , 0)	0.848	0.830	0.08 (0.073 , 0.087)	0.073	–
MLM estimation	567.753 (206 , 0)	0.865	0.849	0.069 (0.069 , 0.084)	0.073	1.225
Estimator Contrast§	-127.966 (0 , 0)	0.017	0.019	-0.011 (-0.004 , -0.003)	0.000	–

* Smaller value indicates better fit

† Larger value indicates better fit

‡ >1 indicates the data violates normality assumption and MLM is a better estimator

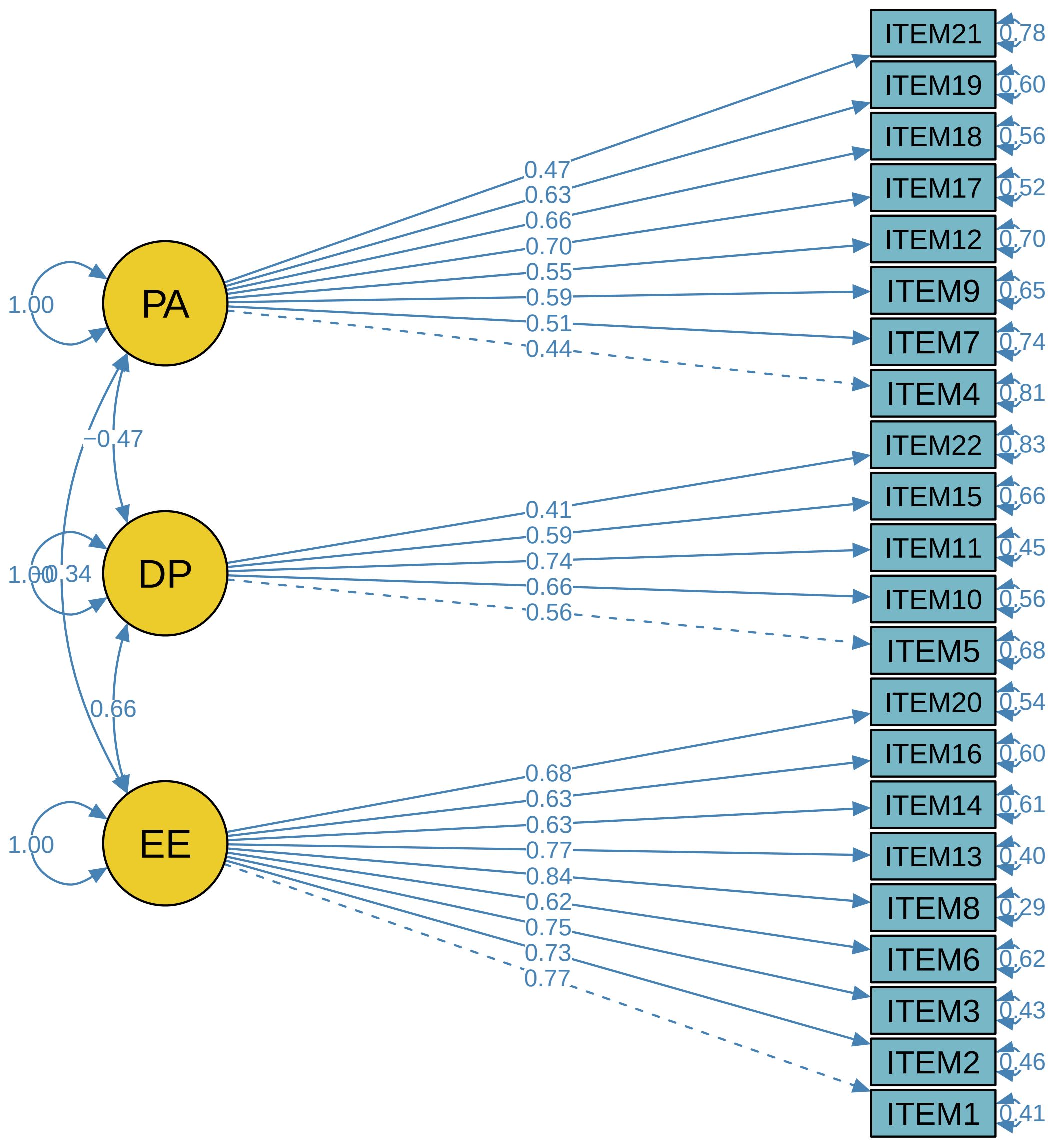
§ Constrast of results from estimations rather than direct model values: MLM-ML

3 Vsiualize the CFA model

```
library(semPlot)
library(wesanderson)#a handful of color palettes from Wes anderson movies
mycols <- wes_palette(name = "Zissou1", n = 3, type = "discrete")
colorlist <- list(man = mycols[2], lat = mycols[3])

semPaths(cfa2,
         "par",
         "std",
         weighted = FALSE, #no weight
         curvature = 1,  #curvature strength
         shapeMan = "rectangle", #manifest variable's shape
         sizeMan = 8, # manifest variable's font size
         sizeMan2 = 3, # manifest variable's tile size
         rotation = 2, # turn vertical
         color = colorlist, #specify color by calling colorlist defined above
         edge.color = "steelblue", #specify line color
         edge.label.cex = 0.7,
         title = T) #specify line label font size
title("Figure 4. Three factor MBI CFA model diagram")
```

Figure 4. Three factor MBI CFA model diagram



4 Finetune model base on Modification Indices

4.1 MI-based finetuning on initial model

4.1.1 Inspect MIs

```
#Generate MI table
MI1 <- modindices(cfa2,
                  standardized = TRUE,
                  sort. = TRUE,
                  maximum.number = 10)

#improve readibility and select columns
MI1 <- MI1 %>% mutate(op = if_else(op == "~~",
                                  "↔",
                                  "→"))%>%
  mutate (Parameter = paste(lhs, " ", op, " ", rhs)) %>%
  select(Parameter, MI = mi, EPC = epc, SEPC = sepc.all)

#remove row names
rownames(MI1) <- NULL

#display table
MI1 %>%
  kable(booktabs = T,
        caption = "Largest modification indices for fixed parameters",
        linesep = "",
        digits = 3,
        align = "lrrr") %>%
  kable_styling(latex_options = "striped") %>%
  footnote(general =c("→ indicates a factor loading. ↔ indicates a covariance.",
                      "MI, modification index; (S)EPC, (standardized) expected
                      parameter change")) %>%
  column_spec(1, width = "6cm") %>%
  column_spec(2, width = "2cm") %>%
  column_spec(3, width = "2cm") %>%
  column_spec(4, width = "2cm")
```

According to Byrne, the contents of the items Item16 and Item6 overlap (they essentially ask the same question). They were checked below.

```
get_label(mbi, ITEM6, ITEM16) %>% kable
```

	x
ITEM6	Working with people all day is really a strain for me.
ITEM16	Working with people directly puts too much stress on me.

Indeed, both focused on the pressure felt during working with people. Not much a difference. I will modify the model by allowing them freely to co-vary with each other, albeit deleting one of the item might be another solution.

Table 11: Largest modification indices for fixed parameters

Parameter	MI	EPC	SEPC
ITEM6 ←→ ITEM16	91.282	0.733	0.529
ITEM1 ←→ ITEM2	82.448	0.613	0.549
EE → ITEM12	41.517	-0.313	-0.335
ITEM10 ←→ ITEM11	38.081	0.580	0.525
ITEM7 ←→ ITEM21	33.529	0.263	0.326
ITEM4 ←→ ITEM7	33.432	0.209	0.324
PA → ITEM1	28.732	0.872	0.231
ITEM18 ←→ ITEM19	18.607	0.250	0.285
ITEM6 ←→ ITEM5	17.193	0.354	0.232
ITEM5 ←→ ITEM15	15.584	0.313	0.243

Note:
→ indicates a factor loading. ←→ indicates a covariance.
makecell[c]MI, modification index; (S)EPC, (standardized) expected parameter change

4.1.2 Modify the model (1st time)

```
model2 <- '
  EE =~ ITEM1 + ITEM2 + ITEM3 + ITEM6 + ITEM8 +
        ITEM13 + ITEM14 + ITEM16 + ITEM20
  DP =~ ITEM5 + ITEM10 + ITEM11 + ITEM15 +ITEM22
  PA =~ ITEM4 + ITEM7 + ITEM9 + ITEM12 +
        ITEM17 + ITEM18 + ITEM19 + ITEM21
  ITEM6 ~~ITEM16
'
cfa.modified.1 <- cfa(model2, data = mbi, estimator = "MLM")
#summary(cfa.modified.1, fit.measures = TRUE, standardized = TRUE)
cfa.summary.mlm.a(cfa.modified.1, 3, 22, "MLM")
```

Table 12: Goodness-of-fit and subjective indices of fit for 3 factor model MLM

Measure	Value
chi square	493.398
df	205.000
p value	0.000
CFI	0.892
TLI	0.879
RMSEA	0.061
RMSEA p value	0.002
SRMR	0.071
CSF	1.211

4.2 MI-based finetuning on model 2

4.2.1 Inspect MIs of model 2

```
#Generate MI table
MI.modified.1 <- modindices(cfa.modified.1,
                  standardized = TRUE,
                  sort. = TRUE,
                  maximum.number = 10)

#improve readability and select columns
MI.modified.1 <- MI.modified.1 %>% mutate(op = if_else(op == "~~",
                                  "↔",
                                  "→")) %>%
  mutate (Parameter = paste(lhs, " ", op, " ", rhs)) %>%
  select(Parameter, MI = mi, EPC = epc, SEPC = sepc.all)

#remove row names
rownames(MI.modified.1) <- NULL

#display table
MI.modified.1 %>%
  kable(booktabs = T,
        caption = "Largest modification indices for fixed parameters",
        linesep = "",
        digits = 3,
        align = "lrrr") %>%
  kable_styling(latex_options = "striped") %>%
  footnote(general =c("→ indicates a factor loading. ↔ indicates a covariance.",
                      "MI, modification index; (S)EPC, (standardized) expected
                      parameter change.")) %>%
  column_spec(1, width = "6cm") %>%
  column_spec(2, width = "2cm") %>%
  column_spec(3, width = "2cm") %>%
  column_spec(4, width = "2cm")
```

Residual covariance related to items 1 and 2 remains a strongly misspecific parameter (MI = 78.275, EPC = 0.591). Also (according to Byrne) we have clear overlap of content with these items. I had checked it as below.

```
get_label(mbi, ITEM1, ITEM2) %>% kable
```

	x
ITEM1	I feel emotionally drained from my work
ITEM2	I feel used up at the end of the workday.

It appears item 1 focuses on about exhaustion on emotional side, while item 2 looks at exhaustion all-round. Indeed, they have overlap but are not measuring exactly the same thing. I will set the parameter free in the model.

Table 13: Largest modification indices for fixed parameters

Parameter	MI	EPC	SEPC
ITEM1 $\longleftrightarrow$ ITEM2	78.275	0.591	0.545
EE $\rightarrow$ ITEM12	41.936	-0.310	-0.336
ITEM10 $\longleftrightarrow$ ITEM11	37.348	0.578	0.526
ITEM7 $\longleftrightarrow$ ITEM21	33.497	0.263	0.326
ITEM4 $\longleftrightarrow$ ITEM7	33.386	0.209	0.323
PA $\rightarrow$ ITEM1	28.188	0.851	0.225
ITEM18 $\longleftrightarrow$ ITEM19	18.617	0.250	0.285
ITEM5 $\longleftrightarrow$ ITEM15	16.067	0.318	0.246
ITEM3 $\longleftrightarrow$ ITEM12	15.294	-0.253	-0.225
PA $\rightarrow$ ITEM13	14.632	-0.628	-0.164

Note:
$\rightarrow$ indicates a factor loading. $\longleftrightarrow$ indicates a covariance.
makecell[c]MI, modification index; (S)EPC, (standardized) expected parameter change.

4.2.2 Modify the model (2nd time)

```
model3 <- '
  EE =~ ITEM1 + ITEM3 + ITEM2 + ITEM6 + ITEM8 +
        ITEM13 + ITEM16 + ITEM14 + ITEM20
  DP =~ ITEM5 + ITEM10 + ITEM15 + ITEM11 +ITEM22
  PA =~ ITEM4 + ITEM7 + ITEM9 + ITEM12 +
        ITEM17 + ITEM18 + ITEM19 + ITEM21
  ITEM6 ~~ITEM16
  ITEM1 ~~ITEM2
'
cfa.modified.2 <- cfa(model3, data = mbi, estimator = "MLM")
#summary(cfa.modified.2, fit.measures = TRUE, standardized = TRUE)
cfa.summary.mlm.a(cfa.modified.2, 3, 22, "MLM")
```

Table 14: Goodness-of-fit and subjective indices of fit for 3 factor model MLM

Measure	Value
chi square	431.496
df	204.000
p value	0.000
CFI	0.915
TLI	0.904
RMSEA	0.055
RMSEA p value	0.114
SRMR	0.069
CSF	1.206

4.3 MI-based finetuning on model 3

4.3.1 Inspect MIs of model 3

```
#Generate MI table
MI.modified.2 <- modindices(cfa.modified.2,
                  standardized = TRUE,
                  sort. = TRUE,
                  maximum.number = 10)

#improve readability and select columns
MI.modified.2 <- MI.modified.2 %>% mutate(op = if_else(op == "~~",
                                 "↔",
                                 "→")) %>%
  mutate (Parameter = paste(lhs, " ", op, " ", rhs)) %>%
  select(Parameter, MI = mi, EPC = epc, SEPC = sepc.all)

#remove row names
rownames(MI.modified.2) <- NULL

#display table
MI.modified.2 %>%
  kable(booktabs = T,
        #format = "markdown",
        caption = "Largest modification indices for fixed parameters",
        linesep = "",
        digits = 3,
        align = "lrrr") %>%
  kable_styling(latex_options = "striped") %>%
  footnote(general =c("→ indicates a factor loading. ↔ indicates a covariance.",
                     "MI, modification index; (S)EPC, (standardized) expected
                     parameter change.")) %>%
  column_spec(1, width = "6cm") %>%
  column_spec(2, width = "2cm") %>%
  column_spec(3, width = "2cm") %>%
  column_spec(4, width = "2cm")
```

The largest MI is the mis-specified factor loading of EE onto item 12. However, according to Byrne it seems evident and logical to have this cross-loading. I tried to find out why as below.

```
get_label(mbi, ITEM12) %>% kable
```

	x
ITEM12	I feel very energetic.

This item belongs to personal accomplishment (PA) in the initial model. Yet, by MI we were informed that it had cross-loading with emotional exhaustion (EE). I could see how feeling energetic being one important component of personal accomplishment. A sense of accomplishment is often rewarding mentally. On the other hand, it is not difficult to see the fact that “energetic” happens to be the reverse of “exhaustion”, at least linguistically. Such a semantic connection would very possible lead the respondent to thinking congeneric aspects (though reversely) of their lives, and hence we saw the negative correlation (SEPC=-0.339). This reminds us the caution should be taken in using synonyms and antonyms in the wording of items expected to load on different factors of one scale.

Table 15: Largest modification indices for fixed parameters

Parameter	MI	EPC	SEPC
EE $\rightarrow$ ITEM12	41.026	-0.332	-0.339
ITEM10 $\longleftrightarrow$ ITEM11	37.190	0.575	0.523
ITEM7 $\longleftrightarrow$ ITEM21	33.636	0.264	0.327
ITEM4 $\longleftrightarrow$ ITEM7	33.523	0.210	0.324
ITEM18 $\longleftrightarrow$ ITEM19	18.591	0.250	0.285
ITEM3 $\longleftrightarrow$ ITEM12	16.431	-0.265	-0.233
ITEM5 $\longleftrightarrow$ ITEM15	15.931	0.316	0.245
PA $\rightarrow$ ITEM1	14.440	0.560	0.148
DP $\rightarrow$ ITEM12	14.001	-0.329	-0.230
ITEM1 $\longleftrightarrow$ ITEM3	13.922	0.248	0.192

Note:

$\rightarrow$ indicates a factor loading. $\longleftrightarrow$ indicates a covariance.

makecell[c]MI, modification index; (S)EPC, (standardized) expected parameter change.

Additionally, residual covariance related to items 11 and 10 remains a strongly misspecific parameter (MI = 37.190; SEPC = 0.523). I checked the items as follows.

```
get_label(mbi, ITEM10, ITEM11) %>% kable
```

	x
ITEM10	I've become more callous toward people since I took this job.
ITEM11	I worry that this job is hardening me emotionally.

The words "callous" (from item 10) and "hardening" (item 11) seem to be of similar connotations. And both items concern how the respondents feel about the job. They are to me more like one questions worded in different ways. Though deleting one of them looks more sensible to me, I would still follow the steps of the slides and set this parameter free to estimate.

4.3.2 Modify the model (3rd time)

```
model4 <- '
  EE =~ ITEM1 + ITEM2 + ITEM3 + ITEM6 + ITEM8 +
        ITEM16 + ITEM14 + ITEM20 + ITEM13
  DP =~ ITEM5 + ITEM10 + ITEM11 + ITEM15 +ITEM22
  PA =~ ITEM4 + ITEM7 + ITEM9 + ITEM12 +
        ITEM17 + ITEM18 + ITEM19 + ITEM21
  ITEM6 ~~ITEM16
  ITEM1 ~~ITEM2
  ITEM10 ~~ ITEM11
'
cfa.modified.3 <- cfa(model4, data = mbi, estimator = "MLM")
#summary(cfa.modified.3, fit.measures = TRUE, standardized = TRUE)
cfa.summary.mlm.a(cfa.modified.3, 3, 22, "MLM")
```

Table 16: Goodness-of-fit and subjective indices of fit for 3 factor model MLM

Measure	Value
chi square	403.049
df	203.000
p value	0.000
CFI	0.925
TLI	0.915
RMSEA	0.051
RMSEA p value	0.351
SRMR	0.068
CSF	1.211

4.4 MI-based finetuning on model 4

I started by checking the MI of model 4.

```
#Generate MI table
MI.modified.3 <- modindices(cfa.modified.3,
                  standardized = TRUE,
                  sort. = TRUE,
                  maximum.number = 10)

#improve readability and select columns
MI.modified.3 <- MI.modified.3 %>% mutate(op = if_else(op == "~~",
                                  "↔",
                                  "→")) %>%
  mutate (Parameter = paste(lhs, " ", op, " ", rhs)) %>%
  select(Parameter, MI = mi, EPC = epc, SEPC = sepc.all)

#remove row names
rownames(MI.modified.3) <- NULL

#display table
MI.modified.3 %>%
  kable(booktabs = T,
        #format = "markdown",
        caption = "Largest modification indices for fixed parameters",
        linesep = "",
        digits = 3,
        align = "lrrr") %>%
  kable_styling(latex_options = "striped") %>%
  footnote(general =c("→ indicates a factor loading. ↔ indicates a covariance.",
                      "MI, modification index; (S)EPC, (standardized) expected
                      parameter change.")) %>%
  column_spec(1, width = "6cm") %>%
  column_spec(2, width = "2cm") %>%
  column_spec(3, width = "2cm") %>%
  column_spec(4, width = "2cm")
```

As expected, the cross-loading of Item12 onto Factor 1 is still very strong, with the highest MI (40.621). As discussed above, there are reasons to include this parameter in the model.

Table 17: Largest modification indices for fixed parameters

Parameter	MI	EPC	SEPC
EE → ITEM12	40.621	-0.331	-0.337
ITEM7 ←→ ITEM21	33.404	0.262	0.326
ITEM4 ←→ ITEM7	33.318	0.209	0.323
ITEM18 ←→ ITEM19	18.400	0.248	0.284
ITEM3 ←→ ITEM12	16.749	-0.268	-0.236
PA → ITEM1	14.481	0.561	0.148
DP → ITEM12	14.270	-0.324	-0.243
ITEM1 ←→ ITEM3	13.974	0.249	0.192
ITEM4 ←→ ITEM21	13.190	0.201	0.201
ITEM7 ←→ ITEM18	12.056	-0.147	-0.213

Note:
→ indicates a factor loading. ←→ indicates a covariance.
makecell[c]MI, modification index; (S)EPC, (standardized) expected parameter change.

4.5 Summarize the models fitted

All the four models fitted were tabulated in one table for easy and clear comparison.

```
#define the indicator and parameters I need to obtain.
options(scipen = 999)

indicator.names <- c("chisq.scaled", "df.scaled",
               "cfi.scaled", "tli.scaled","rmsea.scaled",
               "srmr")
parameter.names <- c("lhs", "op", "rhs", "pvalue", "std.all")
```

```
#obtain fit.measures of initial model to model 3
indicator.initial <- fitMeasures(cfa2, indicator.names, output = "matrix")
indicator.modified1 <- fitMeasures(cfa.modified.1, indicator.names, output = "matrix")
indicator.modified2 <- fitMeasures(cfa.modified.2, indicator.names, output = "matrix")
indicator.modified3 <- fitMeasures(cfa.modified.3, indicator.names, output = "matrix")

#assign their column names as model0 to model 3
colnames(indicator.initial) <- "model0"
colnames(indicator.modified1) <- "model1"
colnames(indicator.modified2) <- "model2"
colnames(indicator.modified3) <- "model3"

#bind the columns
indicator.tab <- cbind(indicator.initial, indicator.modified1,
                       indicator.modified2, indicator.modified3)
```

```
#merge the two table
summary.table <- rbind(parameter.tab,indicator.tab)
summary.table <- summary.table %>%
  t() %>%
  as.matrix
```

```
#correct the variable type and round the data to the 3rd position after dot.
summary.table <- summary.table %>%
  as.data.frame() %>%
  mutate(across(4:11, as.numeric)) %>%
  mutate(across(4:11, round, 3)) %>%
  na_if(999)

#calcuate, select and rename needed variabbles.
summary.table <- summary.table %>%
  mutate(parameter = paste(lhs, "↔", rhs)) %>%
  select("Chi square*" = chisq.scaled,
         "df" = df.scaled,
         "CFI†" = cfi.scaled,
         "TLI†" = tli.scaled,
         "RMSEA*" = rmsea.scaled,
         "Parameter" = parameter,
         "p§" = pvalue,
         "Estimate§" = std.all)

#Initial model is the reference
summary.table[1,6] <- NA
```

```
#finetune aesthetics of the table and display it
summary.table %>%
  kable(booktab = T,
        #format = "markdown",
        linesep = "",
        caption = "Model modification summary") %>%
  add_header_above(c(" ",
                     "Fit Indexes" = 5,
                     "Parameter set free‡" = 3)) %>%
  kable_styling() %>%
  footnote(general = " ↔ indicates a covariance.",
           symbol = c("Smaller value indicates better fit",
                      "Larger value indicates better fit",
                      "Parameters set free to estimate compared to model 0",
                      "p and Standardized regression coeffient for the parameter"))
```

From model 0 to 3, Chi square value and RMSEA kept decreasing, while CFI and TLI continued increasing. All the parameters newly set free were significant and the standardized regression coefficient were 0.369~0.497, which were fairly considerable correlation. This demonstrated that, with the progression of our modification, we were achieving better and better model.

4.6 Draw the graph of the final model

Since I had decided upon the forth model to be the final model, its model diagram was plotted.

```
grps <- list(EE = c("ITEM1", "ITEM2", "ITEM3", "ITEM6", "ITEM8",
        "ITEM13", "ITEM14",  "ITEM16", "ITEM20"),
        DP = c("ITEM5", "ITEM10", "ITEM11", "ITEM15","ITEM22"),
        PA = c("ITEM4", "ITEM7", "ITEM9", "ITEM12",
               "ITEM17", "ITEM18", "ITEM19", "ITEM21"))
```

Table 18: Model modification summary

	Fit Indexes					Parameter set free‡		
	Chi square*	df	CFI†	TLI†	RMSEA*	Parameter	p§	Estimate§
model0	567.753	206	0.865	0.849	0.069	NA	NA	NA
model1	493.398	205	0.892	0.879	0.061	ITEM6 $\longleftrightarrow$ ITEM16	0	0.497
model2	431.496	204	0.915	0.904	0.055	ITEM1 $\longleftrightarrow$ ITEM2	0	0.473
model3	403.049	203	0.925	0.915	0.051	ITEM10 $\longleftrightarrow$ ITEM11	0	0.369

Note:

$\longleftrightarrow$ indicates a covariance.

[*] Smaller value indicates better fit

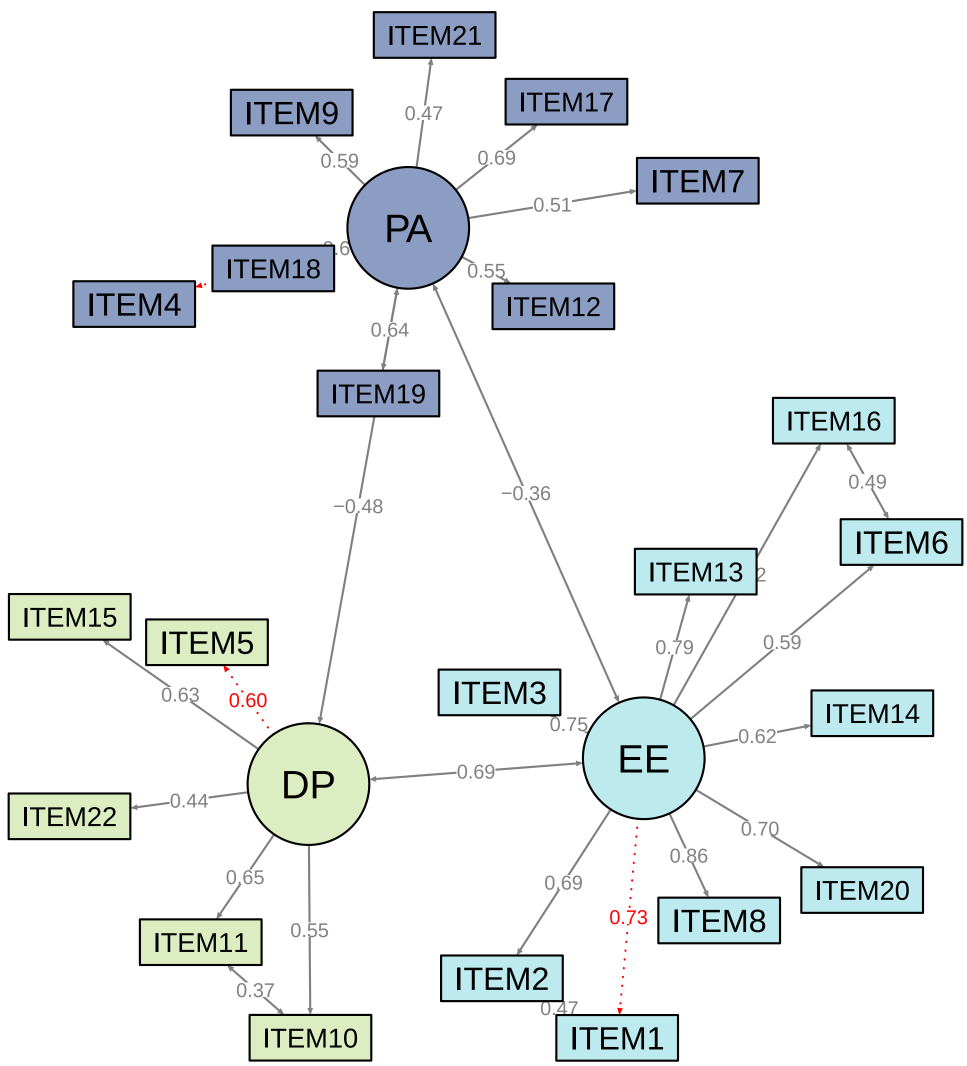
[†] Larger value indicates better fit

[‡] Parameters set free to estimate compared to model 0

[§] p and Standardized regression coeffient for the parameter

```r
semPaths(cfa.modified.3,
         "par",
         "std",
         residuals = F,
         weighted = FALSE, #no weight
         groups = grps,
         posColor = c("red", "purple", "yellow", "black"),
         color = c("#bdeaee", "#dcedc1", "#8b9dc3"),
         curvature = 1,  #curvature strength
         shapeMan = "rectangle", #manifest variable's shape
         sizeMan = 8, # manifest variable's font size
         sizeMan2 = 3, # manifest variable's tile size
         rotation = 2, # turn vertical
         edge.color = "steelblue", #specify line color
         edge.label.cex = 0.6,
         title = T,
         curve =1.5,
         legend = F,
         asize = 1,
         style = "ram",
         #layoutSplit= T,
         fixedStyle = c("red",3),
         #edge.color = "#bdeaee",
         layout = "spring") #specify line label font size
title("Figure 5.Modifed MBI CFA model diagram")
```

Figure 5.Modifed MBI CFA model diagram



5 More exploration

The modifications cannot be statistical decisions (alone), they require good knowledge about the data and the theory. This is definitely true. However, I am curious about what would happen if we keep modifying the model by always following what MI has informed us, regardless of any theory. This way, had this shown equally same results from theory-aided practice, or had we seen a trend of consistently improving fit indices, we would then see the importance of using MI with tremendous domain knowledge, since statistics would anyway provide better and better results (And my assumption is this is true).This would be very dangerous in practice.

Considering we still have 231 degrees of freedom left at the final model, I tested my assumption by setting up a loop that automatically performs MI-based model modification for 100 times. Some model fit indices and model parameters for each model fitted were archived during this process. In this automation, model modification was always done indiscriminately according to the largest MI at each iteration. For each largest MI from a BY relationship, set free was indiscriminately the residual covariance of the items; for each largest MI from a WITH relationship, set free was indiscriminately the cross-loading of item on the identified factor. I understand, there were considerably huge problem in this process, like, in practice covariance might also indicate item deletion and cross-lading might be addressed by switching the item to the identified factor. But it did not hurt to do as an exercise.

```
 #initial model
  formula <- "EE =~ ITEM1 + ITEM2 + ITEM3 + ITEM6 + ITEM8 +
        ITEM16 + ITEM14 + ITEM20 + ITEM13
  DP =~ ITEM5 + ITEM10 + ITEM11 + ITEM15 +ITEM22
  PA =~ ITEM4 + ITEM7 + ITEM9 + ITEM12 +
        ITEM17 + ITEM18 + ITEM19 + ITEM21
  ITEM6 ~~ITEM16
  ITEM1 ~~ITEM2
  ITEM10 ~~ ITEM11"

#fit indices used
mlm.names <- c("chisq.scaled", "df.scaled", "pvalue.scaled",
               "cfi.scaled", "tli.scaled","rmsea.scaled",
               "rmsea.ci.lower.robust", "rmsea.ci.upper.robust",
               "srmr", "chisq.scaling.factor")

#MI-related indicators
mi.names <- c("lhs", "op", "rhs", "mi", "epc", "sepc.all")

#combine them
matrix.names <- c(mlm.names, mi.names)

#generate the matrix
mymatrix <- matrix(NA, 100, 16)

#set column names
colnames(mymatrix) <- matrix.names

#go through the loop
for (i in 1:100){
  #fit
  model <- cfa(formula, data = mbi, estimator = "MLM")
  mi.tab <- modindices(model, standardized = TRUE, sort. = TRUE,
                       maximum.number = 1)
```

```
  mlm.indicator <- fitMeasures(model, mlm.names, output = "matrix") %>% t()
  mi.tab <- mi.tab %>% select(mi.names)
  rownames(mi.tab) <- NULL
  one.row <- cbind(mlm.indicator, mi.tab)
  one.row <- one.row %>% mutate(across(where(is.numeric), round, 3))
  #save the values into a row of the matrix
  mymatrix[i,] <- unname(unlist(one.row[1,]))
  #refit the modified model
  free.parameter <- paste(unname(unlist(select(one.row, lhs, op, rhs)[1,])),
                          collapse = "")
  formula <- paste(formula, "\n",free.parameter)
}
```

```
#further selected some representative indicators
myframe <- mymatrix  %>% as.data.frame()
myframe <- myframe %>%
  mutate(iteration.num = 1:100) %>%
  dplyr::select(iteration.num, everything()) %>%
  select(-lhs, -op, -rhs, -epc )

#turn character variable to numeric
myframe <- myframe %>%
  lapply(as.numeric) %>%
  data.frame()

#remae the indicators to improve clarity
names(myframe) <- c("Iteration", "Chi square", "df", "p-value", "CFI", "TLI",
                    "RMSEA", "RMSEA upper 95%CI", "RMSEA lower 95%CI", "SRMR",
                    "CSF", "MI", "SEPC")

#display fit indices for the first 20 iterations of automatic model
myframe %>%
  head(25) %>%
  kable(booktab = T,
        linesep = "",
        caption =
          "Fit indices for the first 20 iterations of automatic model
        modification (other 80 not shown)") %>%
  kable_styling(latex_options = "striped") %>%
  footnote(general = "CSF: Chi-squre sclaing factor; SEPC: standardized expected
                      parameter change") %>%
  landscape()
```

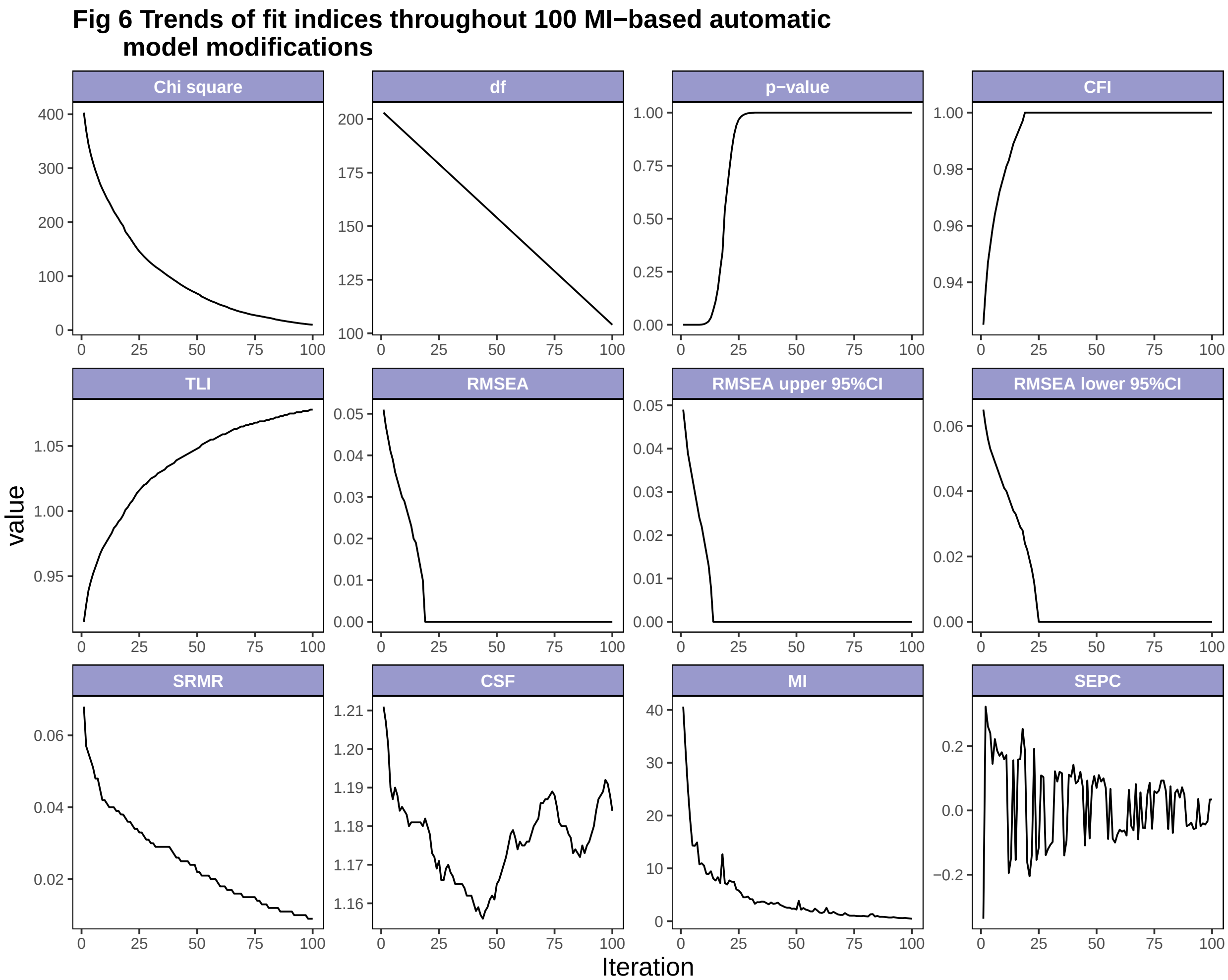
Table 19: Fit indices for the first 20 iterations of automatic model modification (other 80 not shown)

Iteration	Chi square	df	p-value	CFI	TLI	RMSEA	RMSEA upper 95%CI	RMSEA lower 95%CI	SRMR	CSF	MI	SEPC
1	403.049	203	0.000	0.925	0.915	0.051	0.049	0.065	0.068	1.211	40.621	-0.337
2	369.998	202	0.000	0.937	0.928	0.047	0.044	0.060	0.057	1.207	32.503	0.323
3	344.361	201	0.000	0.947	0.939	0.044	0.039	0.056	0.055	1.201	25.273	0.261
4	325.363	200	0.000	0.953	0.946	0.041	0.036	0.053	0.053	1.190	19.270	0.241
5	309.886	199	0.000	0.959	0.952	0.039	0.033	0.051	0.051	1.187	14.331	0.145
6	295.786	198	0.000	0.964	0.957	0.036	0.030	0.049	0.048	1.190	14.262	0.222
7	283.645	197	0.000	0.968	0.962	0.034	0.027	0.047	0.048	1.188	14.912	0.186
8	271.290	196	0.000	0.972	0.967	0.032	0.024	0.045	0.045	1.184	10.779	0.170
9	261.718	195	0.001	0.975	0.971	0.030	0.022	0.043	0.042	1.185	10.957	0.181
10	252.806	194	0.003	0.978	0.974	0.029	0.019	0.041	0.042	1.184	10.517	0.159
11	243.762	193	0.008	0.981	0.977	0.027	0.016	0.040	0.041	1.183	8.978	0.172
12	236.635	192	0.016	0.983	0.980	0.025	0.013	0.038	0.040	1.180	8.939	-0.195
13	228.220	191	0.034	0.986	0.983	0.023	0.008	0.036	0.040	1.181	9.444	-0.148
14	219.688	190	0.069	0.989	0.987	0.020	0.000	0.034	0.040	1.181	8.066	0.156
15	213.117	189	0.110	0.991	0.989	0.019	0.000	0.033	0.039	1.181	7.698	-0.154
16	206.460	188	0.169	0.993	0.992	0.016	0.000	0.031	0.039	1.181	8.324	0.158
17	199.085	187	0.259	0.995	0.994	0.013	0.000	0.029	0.038	1.181	7.229	0.160
18	193.301	186	0.342	0.997	0.997	0.010	0.000	0.028	0.038	1.180	12.675	0.254
19	182.428	185	0.540	1.000	1.001	0.000	0.000	0.024	0.037	1.182	7.238	0.186
20	176.636	184	0.638	1.000	1.003	0.000	0.000	0.022	0.036	1.180	6.927	-0.162
21	170.649	183	0.734	1.000	1.006	0.000	0.000	0.019	0.036	1.178	7.719	-0.205
22	164.106	182	0.825	1.000	1.008	0.000	0.000	0.016	0.035	1.173	7.468	-0.137
23	157.596	181	0.895	1.000	1.011	0.000	0.000	0.012	0.034	1.172	7.467	0.192
24	151.437	180	0.940	1.000	1.014	0.000	0.000	0.006	0.034	1.169	6.018	-0.154
25	145.794	179	0.967	1.000	1.016	0.000	0.000	0.000	0.033	1.171	5.805	-0.115

Note:

makecell[l]CSF: Chi-squre sclaing factor; SEPC: standardized expected parameter change

```r
#fine-tune the table aesthetics and dispaly it
myframe %>% pivot_longer(cols = 'Chi square':'SEPC', names_to = "indicator", values_to = "value") %>%
  ggplot(aes(x = Iteration, y = value))+
  geom_line()+
  facet_wrap(~factor(indicator, levels = names(myframe)),
             scales = "free") +
  theme(axis.title = element_text(size = 15),
        panel.background = element_rect(fill = "white",
                                        color = "black"),
        panel.grid = element_blank(),
        strip.background = element_rect(color = "black",
                                        fill = "#9999CC"),
        strip.text = element_text(color = "white",
                                  face = "bold",
                                  size =  10),
        plot.title = element_text(size = 15,
                                  face = "bold"))+
  labs(title = "Fig 6 Trends of fit indices throughout 100 MI-based automatic
       model modifications")
```



Some comments on the graph:

(1)Over the progression of 100 model modifications, the chi square value kept decreasing from 400 to almost

0.

(2)I used 100 degrees of freedom during this trial. Note that p-value for chi square statistics might be influence by it.

(3)At around the 20th~25th modification, p-value of chi square increased over 0.50, achieving statistical significance (if multiple comparison was not a concern). The increasing trend was clear and consistent and at around the 25th modification, the p value had hit the ceiling of 1 and stayed there throughout the trial.

(4)for CFI, the increasing tread was much in line with the change of p value, with its ceiling hit at roughly 24th modification.

(5)TLI could be over 1 and hence the trend of increasing is more linear. Still, the growing trend is clear and consistent throughout the progression of modifications.

(6)RMSEA and its 95%CI dropped dramatically in the beginning 23~25 modifications. Then it remained at 0 throughout the trial.

(7)SRMA showed a consistent and clear decreasing trend throughout the trial.

(8)CSF (Chi square scaling factor) fluctuated continuously with its value always being over 1, showing MLM had been a proper estimator all the way.

(9)SPEC (standardized expected parameter change) was equivalent to the correlation coefficient. Its value kept going up and down around value of 0 throughout the modification However, the strength of correlation was all the way decreasing.

The conclusion is if we followed what the data had told us, it would always result in a better fit in all the above indicators. However, this does not mean we should do that. Instead, it highlights the importance of making all the decisions on MI-based model modifications with as much domain knowledge as possible, since it is the only referee for our optimal practice.