Thesis Script: Motivated Responses to a Masculinity Threat in a German Cultural Context

2025-11-30

```r
df <- read.csv(here("data", "fragile_masculinity_motivation_anonymized.csv"))
df <- preprocessData(df, all_items)
```

```
## Number of Observations: 196
## Condition threat: 101 participants
## Condition noThreat: 95 participants
```

1. Study Inclusion Criteria

1.1 Debrief Consent

Participants that withdrew their consent after the debrief are excluded

```r
df <- exclude_participants(
  df,
  DEBRIEFCONSENT == "Y",
  vars = "DEBRIEFCONSENT",
  description = "Debrief consent"
)
```

```
## Debrief consent: Excluded 8 participants (8 of 196).
## Remaining: 188
## Excluded responses summary for `DEBRIEFCONSENT`:  DEBRIEFCONSENT n
## 1              N 8
```

1.2 Demographic criteria

Participants have to be older than 18 years, self-identify as male and native level German skills

```r
df <- exclude_participants(df, AGE >= 18, vars = "AGE", description = "Age")
```

```
## Age: Excluded 3 participants (3 of 188).
## Remaining: 185
## Excluded responses summary for `AGE`:  AGE n
## 1   0 1
## 2   3 1
## 3  12 1
```

```r
df <- exclude_participants(df, SEX == "A001", vars = "SEX", description = "Gender")
```

```
## Gender: Excluded 9 participants (9 of 185).
## Remaining: 176
## Excluded responses summary for `SEX`:   SEX n
## 1 A002 7
## 2 A004 2
```

```r
df <- exclude_participants(df, GER %in% c("A004", "A005"), vars = "GER", description =
 ↪  "German skills")
```

```
## German skills: Excluded 1 participants (1 of 176).
## Remaining: 175
## Excluded responses summary for `GER`:   GER n
## 1 A003 1
```

1.3 Suspicion

Participants may not indicate a strong suspicion about the study (coded in column SUSPICIONEXCLUSION, free text are excluded in anonymous data set)

```r
df <- exclude_participants(
  df,
  SUSPICIONEXCLUSION == FALSE,
  vars = "SUSPICIONEXCLUSION",
  description = "Suspicion"
)
```

```
## Suspicion: Excluded 9 participants (9 of 175).
## Remaining: 166
## Excluded responses summary for `SUSPICIONEXCLUSION`:  SUSPICIONEXCLUSION n
## 1                TRUE 9
```

1.4 WFCT comprehension

Participants should complete more than 50% of word fragments with existing words

```r
df <- exclude_participants(
  df,
  validWordCompletionScore >= 0.5,
  vars = "validWordCompletionScore",
  description = "WFCT comprehension"
)
```

```
## WFCT comprehension: Excluded 11 participants (11 of 166).
## Remaining: 155
## Excluded responses summary for `validWordCompletionScore`:  validWordCompletionScore n
## 1                     0.00 7
## 2                     0.40 2
## 3                     0.05 1
## 4                     0.15 1
```

1.5 Responses to Motivation for Masculine Behaviour

Participants should respond to at least one item of each MMB scale

```
df <- exclude_participants(
  df,
  rowSums(!is.na(df[mmb_pressured_items])) >= 1,
  vars = mmb_pressured_items,
  description = "Missing all Pressured items"
)
```

```
## Missing all Pressured items: Excluded 1 participants (1 of 155).
## Remaining: 154
## Excluded responses summary for `MMBi1`:  MMBi1 n
## 1    NA 1
##
## Excluded responses summary for `MMBi2`:  MMBi2 n
## 1    NA 1
##
## Excluded responses summary for `MMBi3`:  MMBi3 n
## 1    NA 1
##
## Excluded responses summary for `MMBi4`:  MMBi4 n
## 1    NA 1
##
## Excluded responses summary for `MMBi5`:  MMBi5 n
## 1    NA 1
```

```
df <- exclude_participants(
  df,
  rowSums(!is.na(df[mmb_autonomous_items])) >= 1,
  vars = mmb_autonomous_items,
  description = "Missing all Autonomous items"
)
```

```
## Missing all Autonomous items: Excluded 0 participants (0 of 154).
## Remaining: 154
```

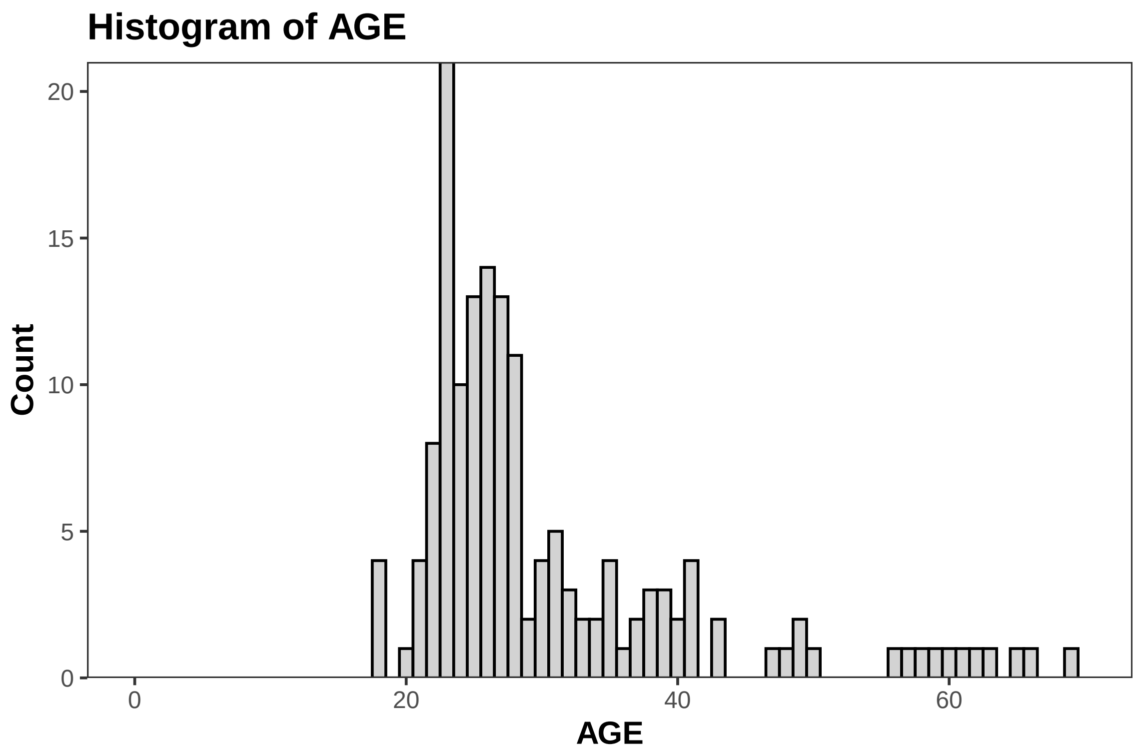
2. Demographics

2.2 Age

```
summary_mean_sd(df, "AGE")
```

```
## # A tibble: 1 x 5
##   variable  mean    sd     n missing
##   <chr>    <dbl> <dbl> <dbl>   <dbl>
## 1 AGE       30.6  10.9   154       0
```

```r
plotHist("AGE", c(0, 70), condition = NULL, binwidth = 1)
```

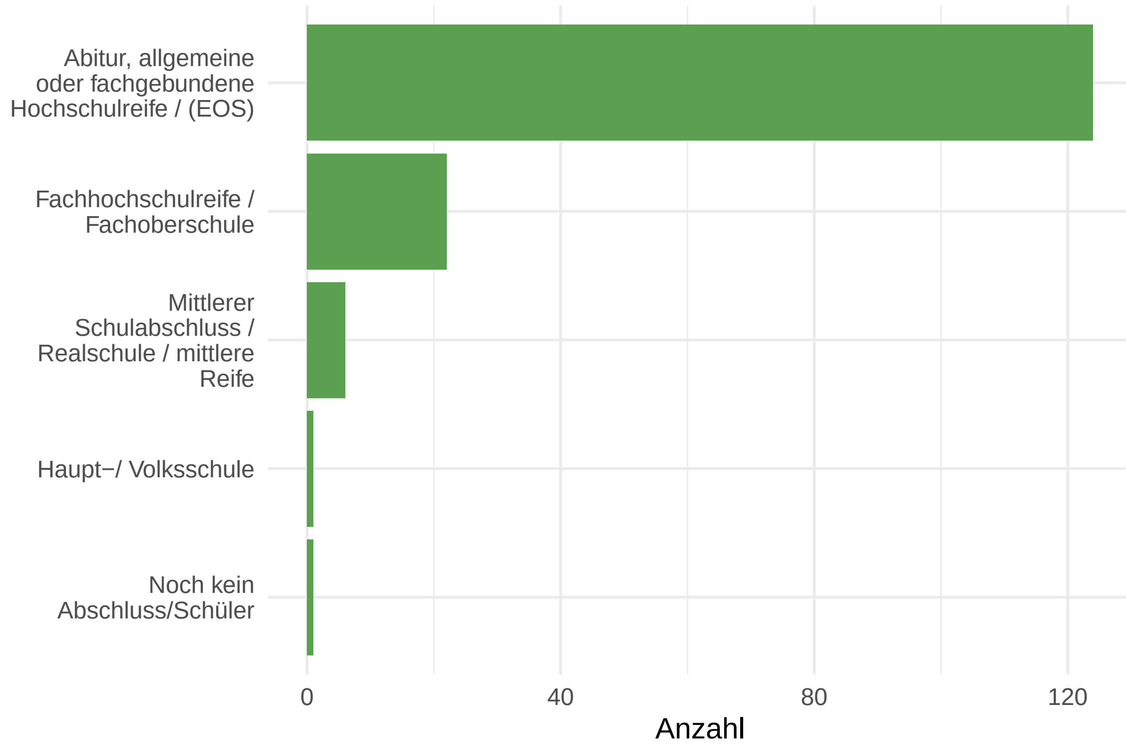


2.3 Educational Degree

```r
freq_table(df, EDU_label)
```

```
## # A tibble: 5 x 3
##   EDU_label                                                      Count Percent
##   <chr>                                                          <int>   <dbl>
## 1 "Abitur, allgemeine\noder fachgebundene\nHochschulreife / (EOS)"   124    80.5
## 2 "Fachhochschulreife /\nFachoberschule"                              22    14.3
## 3 "Mittlerer\nSchulabschluss /\nRealschule / mittlere\nReife"          6     3.9
## 4 "Haupt-/ Volksschule"                                                1     0.6
## 5 "Noch kein\nAbschluss/Schüler"                                       1     0.6
```

```r
plotBarHorizontal("EDU_label") +
  labs(
    x = NULL,
    y = "Anzahl"
  )
```




```r
freq_table(df, EDUPUPIL_label)
```

```
## # A tibble: 2 x 3
##   EDUPUPIL_label      Count Percent
##   <chr>               <int>   <dbl>
## 1 <NA>                  153    99.4
## 2 Haupt-/ Volksschule     1     0.6
```

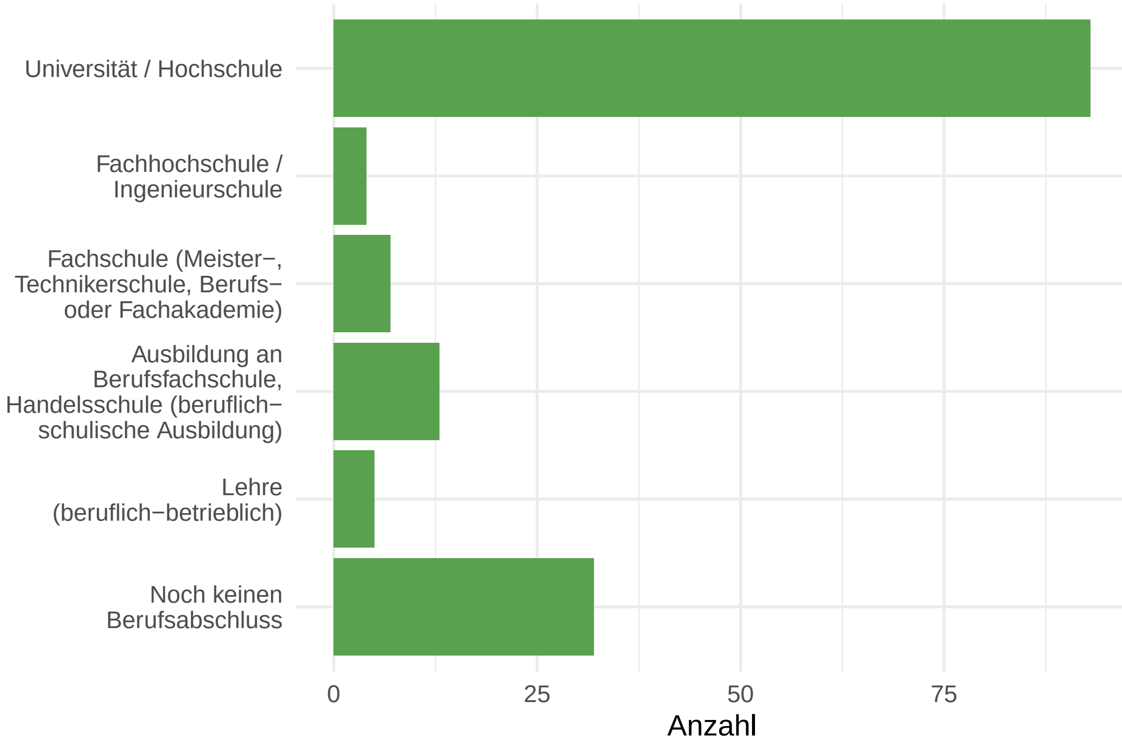
2.4 Occupational Degree

```r
freq_table(df, OCC_label)
```

```
## # A tibble: 6 x 3
##   OCC_label                                                          Count Percent
##   <chr>                                                              <int>   <dbl>
## 1 "Universität / Hochschule"                                            93    60.4
## 2 "Noch keinen\nBerufsabschluss"                                        32    20.8
## 3 "Ausbildung an\nBerufsfachschule,\nHandelsschule (beruflich-\ns~    13     8.4
## 4 "Fachschule (Meister-,\nTechnikerschule, Berufs-\noder Fachakad~     7     4.5
## 5 "Lehre\n(beruflich-betrieblich)"                                       5     3.2
## 6 "Fachhochschule /\nIngenieurschule"                                    4     2.6
```

```r
plotBarHorizontal("OCC_label") +
  labs(
```

```
    x = NULL,
    y = "Anzahl"
  )
```



3. Experimental Manipulation Check

3.1 Conditions

```
printNumberOfParticipants(df, "threatCondition")
```

```
## Number of Observations: 154
## Condition threat: 75 participants
## Condition noThreat: 79 participants
```

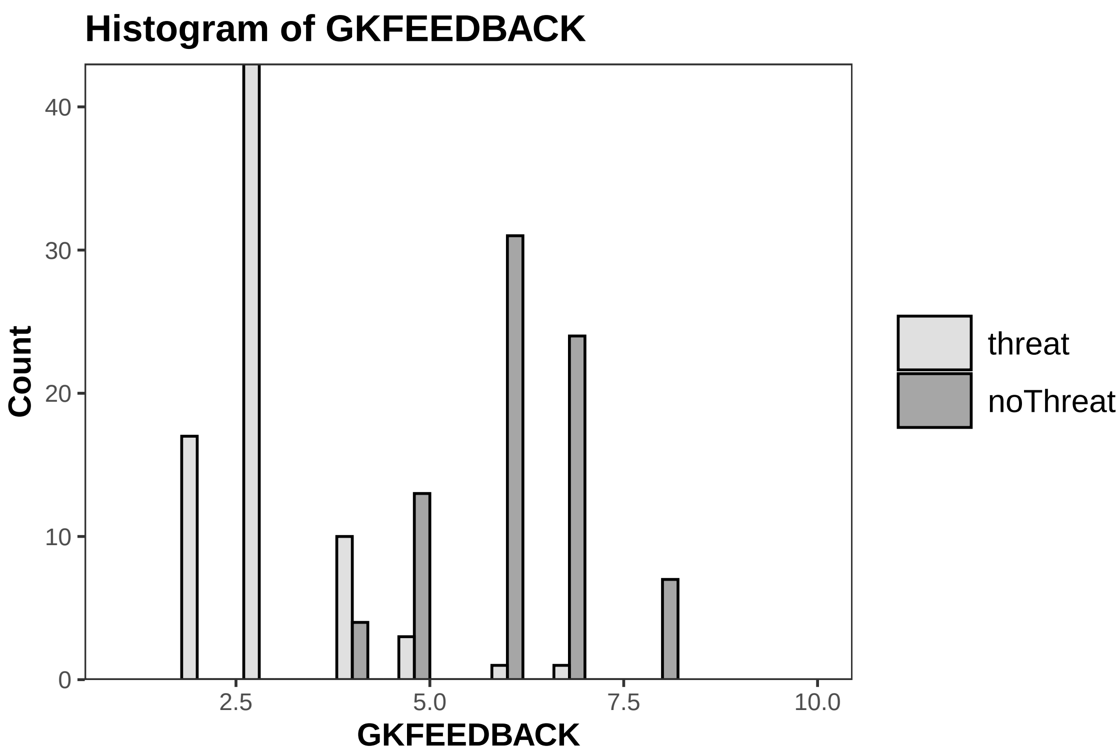
3.2 Gender Knowledge Feedback

[GKFEEDBACK] Welche Rückmeldung haben Sie im Verlauf der Studie zu Ihrem Wissen in geschlechtsspezifischen Themen erhalten? Scale 1 to 10

```
summary_gkfeedback <- summary_mean_sd(df, "GKFEEDBACK", "threatCondition")
summary_gkfeedback
```

```
## # A tibble: 2 x 6
##   threatCondition variable    mean    sd     n missing
##   <fct>           <chr>      <dbl> <dbl> <dbl>   <dbl>
## 1 threat          GKFEEDBACK  3.08  0.93    75       0
## 2 noThreat        GKFEEDBACK  6.22  1       79       0
```

```
plotHist("GKFEEDBACK", c(1, 10), "threatCondition")
```



Interpretation

- There is a clar split between the two conditions, meaning that participants paid attention to the feedback on the GenderKnoledge task and that the feedback was comprehensible

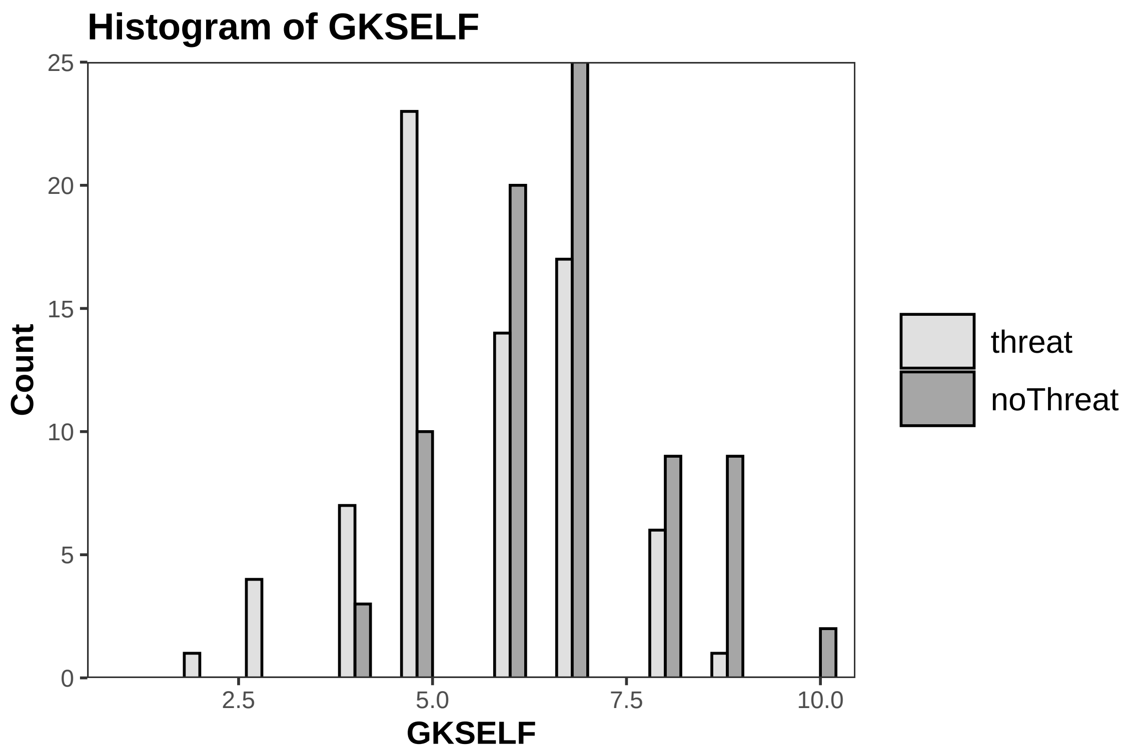
3.3 Self evaluation of Gender Knowledge

[GKSELF] Wie würden Sie Ihr Wissen in geschlechtsspezifischen Themen einschätzen?

```
summary_gkself <- summary_mean_sd(df, "GKSELF", "threatCondition")
summary_gkself
```

```
## # A tibble: 2 x 6
##   threatCondition variable  mean    sd     n missing
##   <fct>           <chr>    <dbl> <dbl> <dbl>   <dbl>
## 1 threat          GKSELF    5.71  1.43    73       2
## 2 noThreat        GKSELF    6.79  1.38    78       1
```

```r
plotHist("GKSELF", c(1, 10), "threatCondition")
```



Interpretation

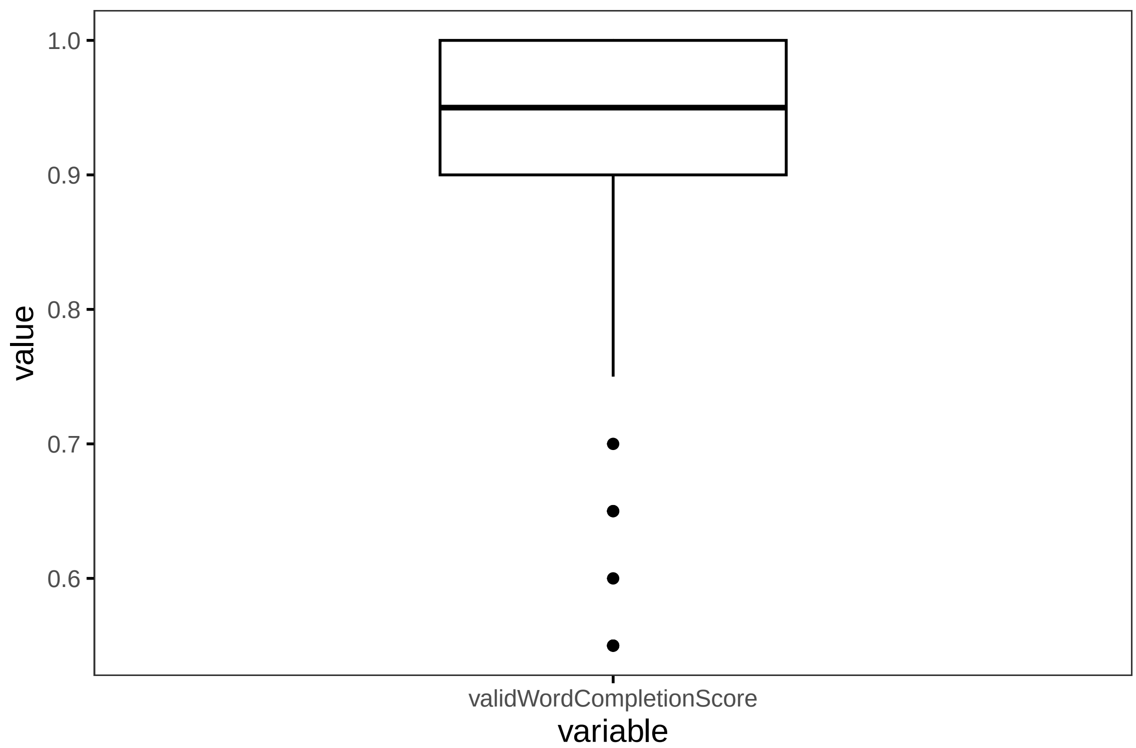
- The mean of self evalutated gender knowledge is lower in the threat condition, than noThreat condition
- This shows that the feedback did have an effect on the self perception of gender knowledge
- The split is not very clear though…

3.4 Valid Word Completions

```r
summary_mean_sd(df, "validWordCompletionScore")
```

```
## # A tibble: 1 x 5
##   variable                  mean    sd     n missing
##   <chr>                    <dbl> <dbl> <dbl>   <dbl>
## 1 validWordCompletionScore  0.92   0.1   154       0
```

```r
plotBox("validWordCompletionScore")
```

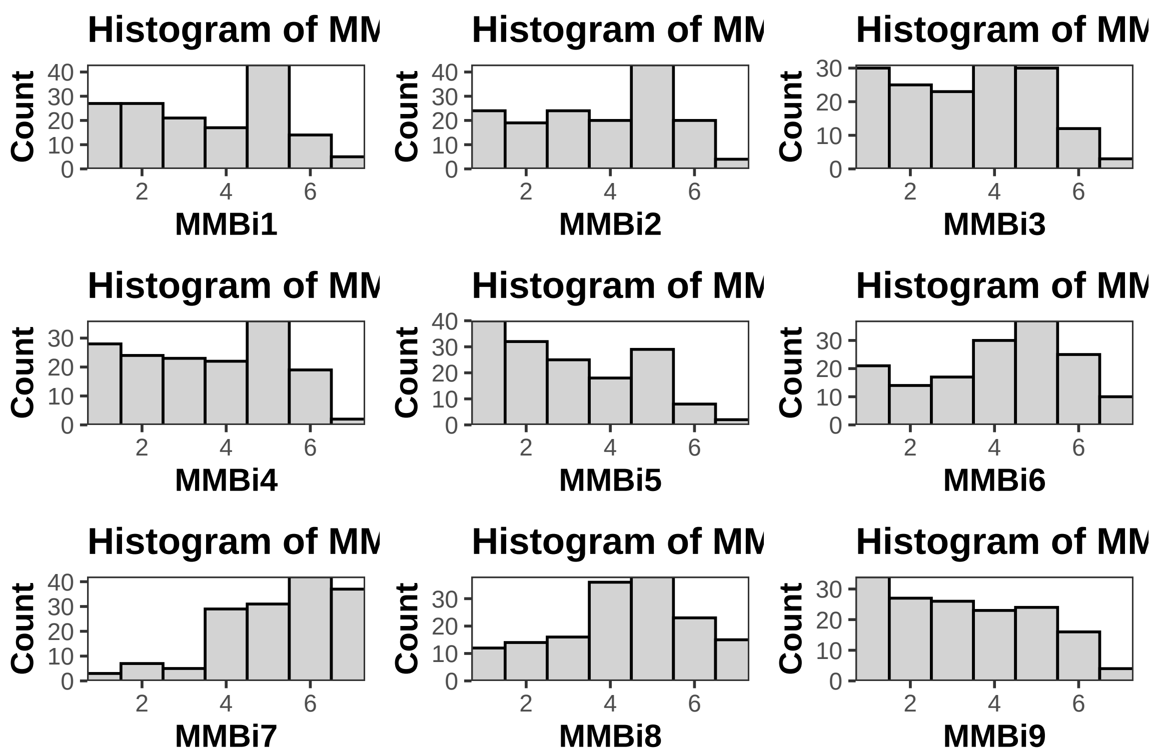


Interpretation

- Participants with a score lower than 50% were excluded
- Generally participants were able to complete the WCFT

3.5 Motivation for Masculine Behaviour

```
plots <- lapply(MMB_item_codes, function(item) {
  plotHist(item, c(1, 7), condition = NULL, binwidth = 1)
})

do.call(grid.arrange, c(plots, ncol = 3))
```

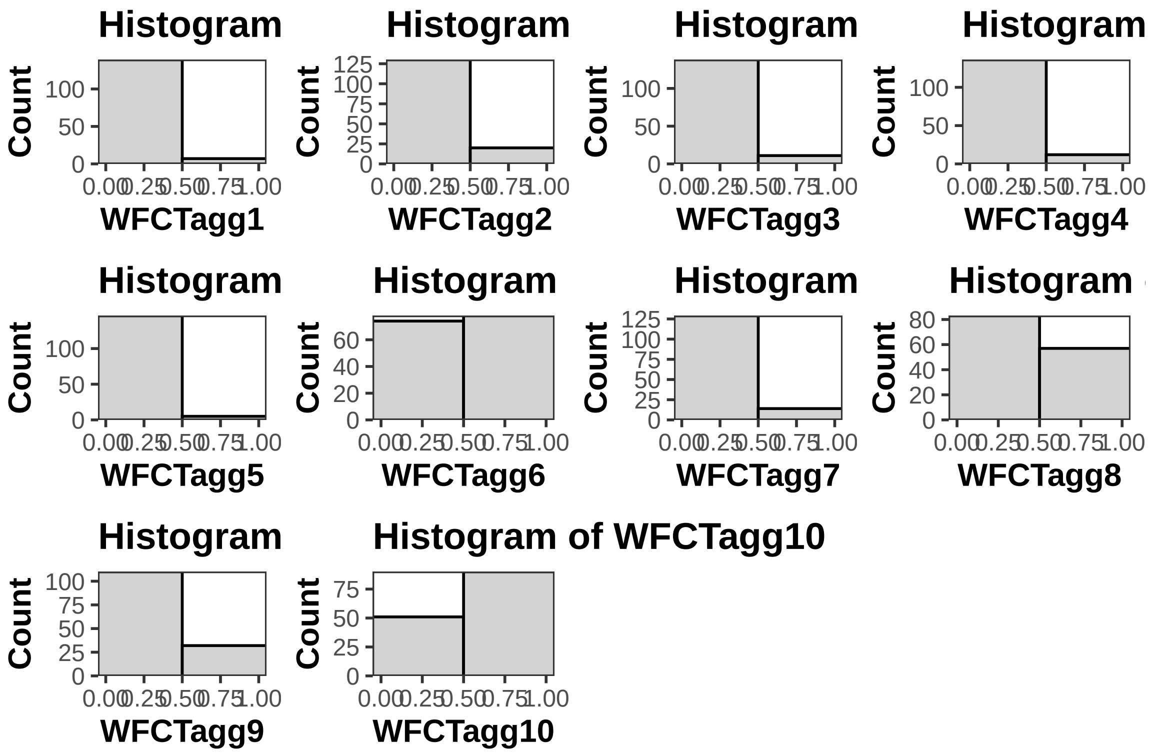



```r
summary_mean_sd(df, MMB_item_codes)
```

```
## # A tibble: 9 x 5
##   variable  mean    sd     n missing
##   <chr>    <dbl> <dbl> <dbl>   <dbl>
## 1 MMBi1     3.55  1.78   154       0
## 2 MMBi2     3.75  1.74   154       0
## 3 MMBi3     3.35  1.68   154       0
## 4 MMBi4     3.51  1.74   154       0
## 5 MMBi5     2.97  1.68   154       0
## 6 MMBi6     4.06  1.79   154       0
## 7 MMBi7     5.29  1.49   154       0
## 8 MMBi8     4.32  1.68   154       0
## 9 MMBi9     3.26  1.77   154       0
```

3.6 Aggressive Cognition

```r
plots <- lapply(WFCT_aggression_items, function(item) {
  plotHist(item, c(0, 1), condition = NULL, binwidth = 1)
})

do.call(grid.arrange, c(plots, ncol = 4))
```

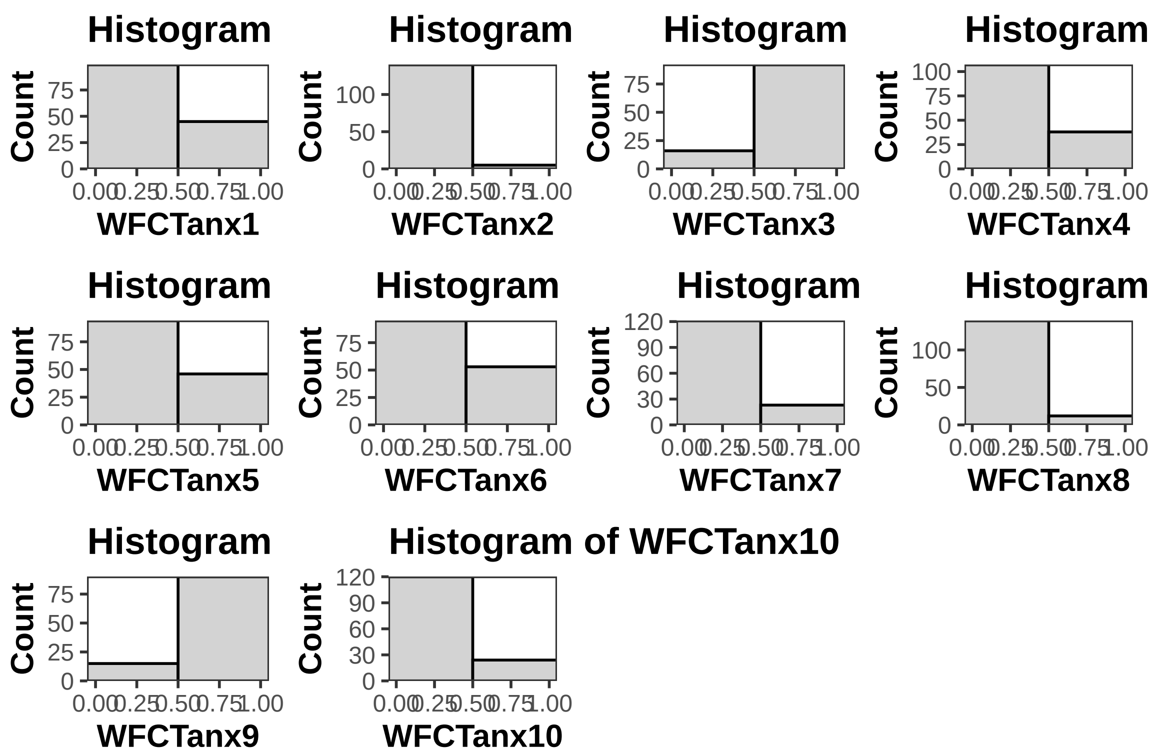



```
summary_mean_sd(df, WFCT_aggression_items)
```

```
## # A tibble: 10 x 5
##    variable   mean    sd     n missing
##    <chr>     <dbl> <dbl> <dbl>   <dbl>
##  1 WFCTagg1   0.05  0.21   146       8
##  2 WFCTagg2   0.13  0.34   150       4
##  3 WFCTagg3   0.07  0.26   149       5
##  4 WFCTagg4   0.08  0.27   148       6
##  5 WFCTagg5   0.03  0.18   151       3
##  6 WFCTagg6   0.51  0.5    152       2
##  7 WFCTagg7   0.1   0.3    143      11
##  8 WFCTagg8   0.41  0.49   140      14
##  9 WFCTagg9   0.23  0.42   142      12
## 10 WFCTagg10  0.64  0.48   141      13
```

3.7 Anxious Cognition

```
plots <- lapply(WFCT_anxiety_items, function(item) {
  plotHist(item, c(0, 1), condition = NULL, binwidth = 1)
})

do.call(grid.arrange, c(plots, ncol = 4))
```




```r
summary_mean_sd(df, WFCT_anxiety_items)
```

```
## # A tibble: 10 x 5
##    variable   mean    sd     n missing
##    <chr>     <dbl> <dbl> <dbl>   <dbl>
##  1 WFCTanx1   0.31  0.47   144      10
##  2 WFCTanx2   0.03  0.18   145       9
##  3 WFCTanx3   0.85  0.36   108      46
##  4 WFCTanx4   0.26  0.44   145       9
##  5 WFCTanx5   0.33  0.47   140      14
##  6 WFCTanx6   0.36  0.48   148       6
##  7 WFCTanx7   0.16  0.37   144      10
##  8 WFCTanx8   0.08  0.27   151       3
##  9 WFCTanx9   0.86  0.35   105      49
## 10 WFCTanx10  0.17  0.37   144      10
```

4. Test Quality Criteria: Motivation for Masculine Behaviour

Objectives:

- Assess the *validity* and *reliability* of the scale Motivation for Masculine Behaviour (MMB)
- Compute participant latent trait scores

Methodology:

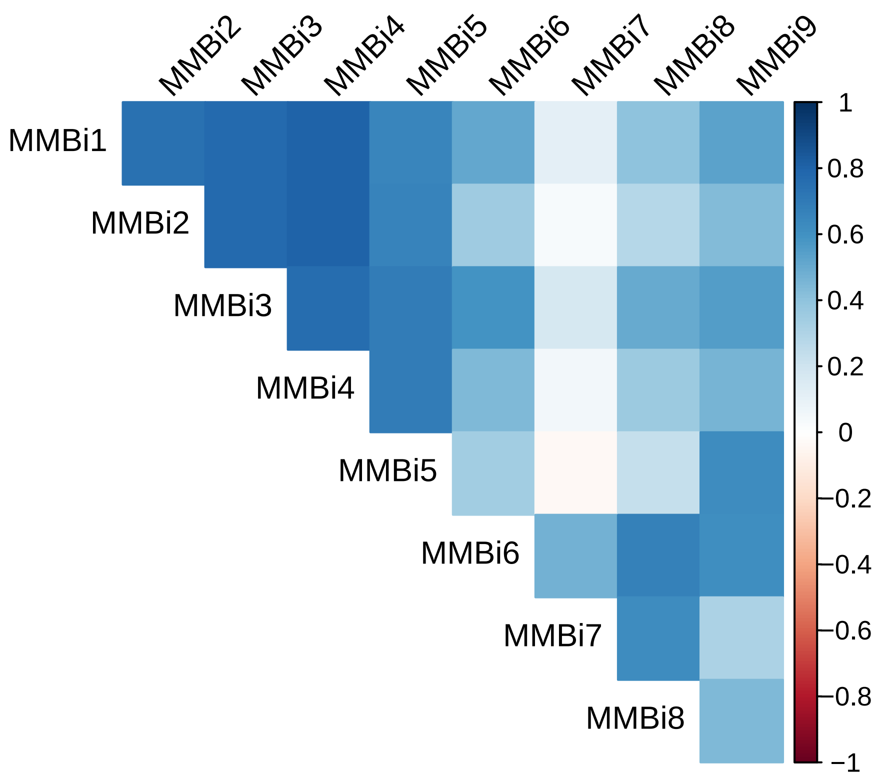
- Visualization & Descriptives

- CFA and EFA to => construct validity
- McDonald's Omega => Reliabiltiy
- Regression scoring => latent trait scores

4.1 Construct Validity

4.1.1 Visualization & Descriptives

```
cor_mmb <- psych::polychoric(df[MMB_item_codes])$rho

corrplot(cor_mmb,
  method = "color",
  type = "upper",
  order = "original",
  tl.col = "black",
  tl.srt = 45,
  diag = FALSE
)
```



Interpretation

- The correlation plit shows a very clear block for MMB 1-5 items (pressured scale)
- The block for for autonomous items (MMB 6 - 9) is way less clear

```r
summary_mean_sd(df, MMB_item_codes)
```

```
## # A tibble: 9 x 5
##   variable  mean    sd     n missing
##   <chr>    <dbl> <dbl> <dbl>   <dbl>
## 1 MMBi1     3.55  1.78   154       0
## 2 MMBi2     3.75  1.74   154       0
## 3 MMBi3     3.35  1.68   154       0
## 4 MMBi4     3.51  1.74   154       0
## 5 MMBi5     2.97  1.68   154       0
## 6 MMBi6     4.06  1.79   154       0
## 7 MMBi7     5.29  1.49   154       0
## 8 MMBi8     4.32  1.68   154       0
## 9 MMBi9     3.26  1.77   154       0
```

Interpretation

- The item scores ranged between 1 and 7
- Most items show a mean in the center of the scale
- Item MMBi7 shows a very high mean (5.27). "Ich bin gerne männlich"

- Generally, the items do not show bottom or ceiling effects, and there is some variance to the responses

4.1.2 Check sampling adequacy for Factory Analysis

```r
KMO(cor_mmb)
```

```
## Kaiser-Meyer-Olkin factor adequacy
## Call: KMO(r = cor_mmb)
## Overall MSA =  0.85
## MSA for each item =
## MMBi1 MMBi2 MMBi3 MMBi4 MMBi5 MMBi6 MMBi7 MMBi8 MMBi9
##  0.92  0.88  0.89  0.88  0.83  0.83  0.69  0.81  0.80
```

```r
cortest.bartlett(cor_mmb, n = nrow(df))
```

```
## $chisq
## [1] 1034.431
##
## $p.value
## [1] 9.376131e-194
##
## $df
## [1] 36
```

KMO (Kaiser-Meyer-Olkin) should be > 0.6 (ideally > 0.8) Bartlett's test should be significant ($p < .05$)

Interpretation

- The overall KMO is 0.86 showing a good smapling adequacy for factor analysis
- Individual scores of items are good as well, except MMBi7, which has a sufficient, yet lower KMO value (0.72) due to it's ceiling effect
- The Bartlett test is significant as well, showing a sampling adequacy for factor analysis

4.1.3 Two-Factor Uncorrelated CFA

Testing the proposed 2 factor structure by Stanaland & Gaither (2021)

- 2 factors
- no covariance between factors

```
model_twof <- "
  # latent variables
  Pressured =~ MMBi1 + MMBi2 + MMBi3 + MMBi4 + MMBi5
  Autonomous =~ MMBi6 + MMBi7 + MMBi8 + MMBi9

  # constrain covariance to 0
  Pressured ~~ 0*Autonomous
"

results_twof <- run_cfa_model(model_twof, df, model_name = "Two-Factor Uncorrelated",
↪  show_mi = TRUE)
```

```
##
## -------------------------
## MODEL SUMMARY
## -------------------------
## lavaan 0.6.15 ended normally after 22 iterations
##
##   Estimator                                         ML
##   Optimization method                           NLMINB
##   Number of model parameters                        18
##
##   Number of observations                           154
##
## Model Test User Model:
##
##   Test statistic                               159.497
##   Degrees of freedom                                27
##   P-value (Chi-square)                           0.000
##
## Model Test Baseline Model:
##
##   Test statistic                               923.528
##   Degrees of freedom                                36
##   P-value                                        0.000
##
## User Model versus Baseline Model:
##
##   Comparative Fit Index (CFI)                    0.851
##   Tucker-Lewis Index (TLI)                       0.801
```

```
##
## Loglikelihood and Information Criteria:
##
##   Loglikelihood user model (H0)              -2317.331
##   Loglikelihood unrestricted model (H1)      -2237.582
##
##   Akaike (AIC)                                4670.661
##   Bayesian (BIC)                              4725.326
##   Sample-size adjusted Bayesian (SABIC)       4668.354
##
## Root Mean Square Error of Approximation:
##
##   RMSEA                                          0.179
##   90 Percent confidence interval - lower         0.152
##   90 Percent confidence interval - upper         0.206
##   P-value H_0: RMSEA <= 0.050                    0.000
##   P-value H_0: RMSEA >= 0.080                    1.000
##
## Standardized Root Mean Square Residual:
##
##   SRMR                                           0.248
##
## 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
##   Pressured =~
##     MMBi1             1.529    0.116   13.203    0.000    1.529    0.862
##     MMBi2             1.491    0.114   13.101    0.000    1.491    0.858
##     MMBi3             1.439    0.110   13.104    0.000    1.439    0.858
##     MMBi4             1.553    0.111   14.012    0.000    1.553    0.894
##     MMBi5             1.246    0.117   10.617    0.000    1.246    0.746
##   Autonomous =~
##     MMBi6             1.458    0.134   10.848    0.000    1.458    0.818
##     MMBi7             0.893    0.119    7.523    0.000    0.893    0.603
##     MMBi8             1.304    0.127   10.254    0.000    1.304    0.781
##     MMBi9             1.064    0.142    7.520    0.000    1.064    0.603
##
## Covariances:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##   Pressured ~~
##     Autonomous        0.000                               0.000    0.000
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##    .MMBi1             0.807    0.118    6.857    0.000    0.807    0.257
##    .MMBi2             0.797    0.115    6.929    0.000    0.797    0.264
##    .MMBi3             0.741    0.107    6.927    0.000    0.741    0.264
##    .MMBi4             0.606    0.099    6.131    0.000    0.606    0.201
##    .MMBi5             1.239    0.156    7.961    0.000    1.239    0.444
```

```
##    .MMBi6             1.048    0.226    4.629    0.000    1.048    0.330
##    .MMBi7             1.394    0.182    7.670    0.000    1.394    0.636
##    .MMBi8             1.088    0.199    5.457    0.000    1.088    0.390
##    .MMBi9             1.982    0.258    7.671    0.000    1.982    0.636
##     Pressured         1.000                               1.000    1.000
##     Autonomous        1.000                               1.000    1.000
##
##
## -------------------------
## MODEL FIT INDICES
## -------------------------
##       Index Value Criteria
## cfi     cfi 0.851     0.95
## tli     tli 0.801     0.95
## rmsea rmsea 0.179     0.06
## srmr   srmr 0.248     0.08
##
## -------------------------
## STANDARDIZED FACTOR LOADINGS
## -------------------------
##       Factor  Item Std_Loading
## 1  Pressured MMBi1       0.862
## 2  Pressured MMBi2       0.858
## 3  Pressured MMBi3       0.858
## 4  Pressured MMBi4       0.894
## 5  Pressured MMBi5       0.746
## 6 Autonomous MMBi6       0.818
## 7 Autonomous MMBi7       0.603
## 8 Autonomous MMBi8       0.781
## 9 Autonomous MMBi9       0.603
##
## -------------------------
## TOP MODIFICATION INDICES
## -------------------------
##           lhs op        rhs     mi   epc sepc.lv sepc.all sepc.nox
## 10  Pressured ~~ Autonomous 37.076 0.559   0.559    0.559    0.559
## 60      MMBi5 ~~      MMBi9 30.371 0.767   0.767    0.489    0.489
## 64      MMBi7 ~~      MMBi8 19.500 0.745   0.745    0.605    0.605
## 63      MMBi6 ~~      MMBi9 19.500 0.992   0.992    0.689    0.689
## 28 Autonomous =~      MMBi3 17.903 0.359   0.359    0.214    0.214
```

Interpretation

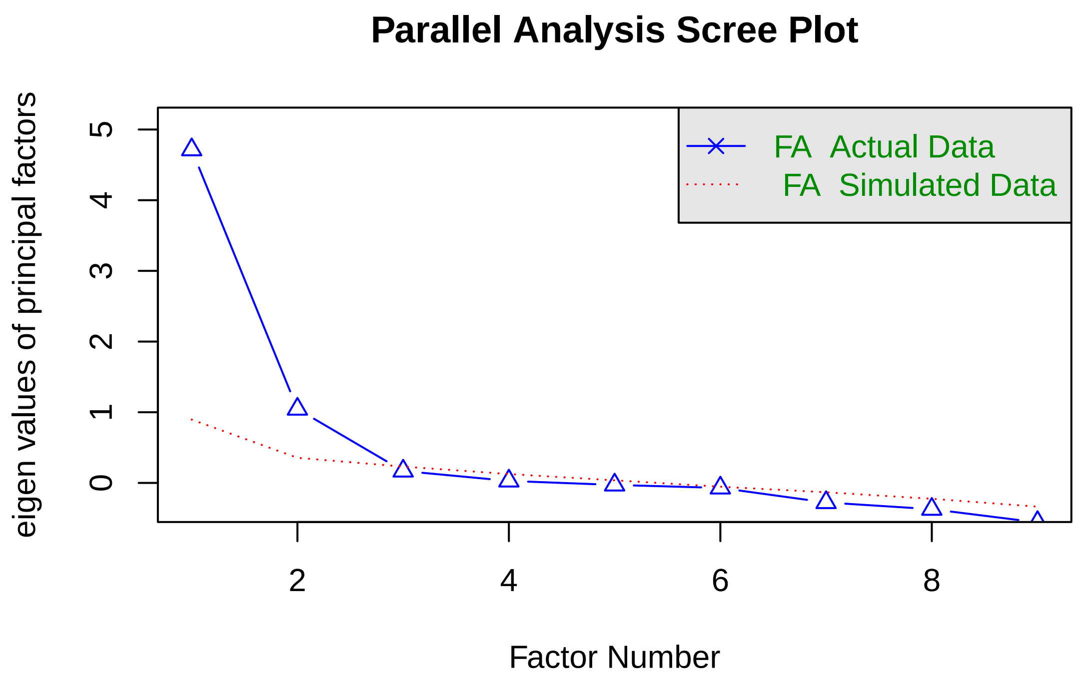
- None of the model fit indices is satisfying
- Mod Indices suggest a correlation between latent pressured and autonomous factors

4.1.4 Exploratory Factor Analysis

```r
fa.parallel(cor_mmb,
  fm = "ml",
  fa = "fa", n.iter = 100, show.legend = TRUE,
```

```r
  main = "Parallel Analysis Scree Plot"
)
```

Determine the number of latent factors




```
## Parallel analysis suggests that the number of factors =  2  and the number of components =  NA
```

Interpretation:

- The "elbow" of the scree plot suggest a two factor model

```r
efa_result_twof <- fa(df[MMB_item_codes], nfactors = 2, rotate = "oblimin", fm = "ml")
print(efa_result_twof, cut = 0.3) # only loadings > .30
```

Compute EFA

```
## Factor Analysis using method =  ml
## Call: fa(r = df[MMB_item_codes], nfactors = 2, rotate = "oblimin",
##     fm = "ml")
## Standardized loadings (pattern matrix) based upon correlation matrix
##          ML1   ML2   h2   u2 com
## MMBi1  0.83        0.75 0.25 1.0
## MMBi2  0.90        0.75 0.25 1.0
```

```
## MMBi3  0.79       0.78 0.22 1.1
## MMBi4  0.89       0.79 0.21 1.0
## MMBi5  0.78       0.58 0.42 1.0
## MMBi6        0.69 0.64 0.36 1.2
## MMBi7        0.74 0.47 0.53 1.2
## MMBi8        0.80 0.67 0.33 1.0
## MMBi9  0.39  0.39 0.42 0.58 2.0
##
##                        ML1  ML2
## SS loadings           3.87 1.96
## Proportion Var        0.43 0.22
## Cumulative Var        0.43 0.65
## Proportion Explained  0.66 0.34
## Cumulative Proportion 0.66 1.00
##
##  With factor correlations of
##     ML1 ML2
## ML1 1.0 0.4
## ML2 0.4 1.0
##
## Mean item complexity =  1.2
## Test of the hypothesis that 2 factors are sufficient.
##
## df null model =  36  with the objective function =  6 with Chi Square =  894.54
## df of  the model are 19  and the objective function was  0.38
##
## The root mean square of the residuals (RMSR) is  0.04
## The df corrected root mean square of the residuals is  0.06
##
## The harmonic n.obs is  154 with the empirical chi square  20.92  with prob <  0.34
## The total n.obs was  154  with Likelihood Chi Square =  56.32  with prob <  1.5e-05
##
## Tucker Lewis Index of factoring reliability =  0.917
## RMSEA index =  0.113  and the 90 % confidence intervals are  0.08 0.148
## BIC =  -39.38
## Fit based upon off diagonal values = 0.99
## Measures of factor score adequacy
##                                                    ML1  ML2
## Correlation of (regression) scores with factors   0.97 0.91
## Multiple R square of scores with factors          0.94 0.83
## Minimum correlation of possible factor scores     0.87 0.66
```

```r
# Create loadings matrix
loadings_matrix <- as.data.frame(efa_result_twof$loadings[1:length(mmb_labels), ])

# Format table
loading_table <- data.frame(
  Item = MMB_item_codes,
  Label = mmb_labels,
  Factor1 = round(loadings_matrix$ML1, 2),
  Factor2 = round(loadings_matrix$ML2, 2)
)

# Hide loadings < .30
```

```r
loading_table <- loading_table %>%
  mutate(
    Factor1 = ifelse(abs(Factor1) < 0.30, "", as.character(Factor1)),
    Factor2 = ifelse(abs(Factor2) < 0.30, "", as.character(Factor2))
  )

# Render nice APA-style table
loading_table %>%
  kbl(
    caption = "EFA Loadings for MMB Items (2-Factor Solution)",
    align = "lccc",
    booktabs = TRUE
  ) %>%
  kable_styling(
    full_width = FALSE,
    position = "center",
    bootstrap_options = c("striped", "hover", "condensed"),
    latex_options = "hold_position"
  ) %>%
  column_spec(2, width = "10cm") %>%
  row_spec(0, bold = TRUE, background = "#f2f2f2") %>%
  row_spec(1:nrow(loading_table), background = "white", color = "black")
```

Table 1: EFA Loadings for MMB Items (2-Factor Solution)

Item	Label	Factor1	Factor2
MMBi1	Allgemein verhalte ich mich männlich, weil ich die Akzeptanz und Anerkennung anderer möchte	0.83	
MMBi2	Allgemein bin ich männlich, weil das von mir erwartet wird	0.9	
MMBi3	Ich verhalte mich männlich, weil ich möchte, dass man mich mag	0.79	
MMBi4	Ich verhalte mich in Gegenwart anderer männlich, um ihre Erwartungen zu erfüllen	0.89	
MMBi5	Ich verhalte mich nicht weiblich, weil ich glaube, dass mich die Leute sonst nicht mögen würden	0.78	
MMBi6	Es ist mir wichtig, männlich zu sein		0.69
MMBi7	Ich bin gerne männlich		0.74
MMBi8	Es macht mich glücklich, mich männlich zu verhalten		0.8
MMBi9	Es ist mir wichtig, mich nicht weiblich zu verhalten	0.39	0.39

Interpretation

- EFA sows a very distinct factor for pressured motivation
- The factor for autonomous motivation is less clear and MMBi9 shows loadings on both factors

4.1.5 Two-Factor Correlated CFA

- 2 latent factors,
- correlation allowed between factors

```r
model_twof_cor <- "
  # latent variables
  Pressured =~ MMBi1 + MMBi2 + MMBi3 + MMBi4 + MMBi5
  Autonomous =~ MMBi6 + MMBi7 + MMBi8 + MMBi9

  # allow correlation between factors
  Pressured ~~ Autonomous
"

results_twof_cor <- run_cfa_model(model_twof_cor, df, model_name = "Two-Factor
↪  Correlated", show_mi = TRUE)
```

```
##
## -------------------------
## MODEL SUMMARY
## -------------------------
## lavaan 0.6.15 ended normally after 21 iterations
##
##   Estimator                                         ML
##   Optimization method                           NLMINB
##   Number of model parameters                        19
##
##   Number of observations                           154
##
## Model Test User Model:
##
##   Test statistic                               113.810
##   Degrees of freedom                                26
##   P-value (Chi-square)                           0.000
##
## Model Test Baseline Model:
##
##   Test statistic                               923.528
##   Degrees of freedom                                36
##   P-value                                        0.000
##
## User Model versus Baseline Model:
##
##   Comparative Fit Index (CFI)                    0.901
##   Tucker-Lewis Index (TLI)                       0.863
##
## Loglikelihood and Information Criteria:
##
##   Loglikelihood user model (H0)              -2294.487
##   Loglikelihood unrestricted model (H1)      -2237.582
##
##   Akaike (AIC)                                4626.974
##   Bayesian (BIC)                              4684.676
##   Sample-size adjusted Bayesian (SABIC)       4624.539
##
## Root Mean Square Error of Approximation:
##
##   RMSEA                                          0.148
```

```
##   90 Percent confidence interval - lower         0.121
##   90 Percent confidence interval - upper         0.176
##   P-value H_0: RMSEA <= 0.050                    0.000
##   P-value H_0: RMSEA >= 0.080                    1.000
##
## Standardized Root Mean Square Residual:
##
##   SRMR                                           0.099
##
## 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
##   Pressured =~
##     MMBi1             1.537    0.115   13.329    0.000    1.537    0.867
##     MMBi2             1.474    0.115   12.868    0.000    1.474    0.848
##     MMBi3             1.460    0.109   13.421    0.000    1.460    0.870
##     MMBi4             1.541    0.111   13.857    0.000    1.541    0.887
##     MMBi5             1.244    0.117   10.599    0.000    1.244    0.745
##   Autonomous =~
##     MMBi6             1.524    0.128   11.898    0.000    1.524    0.856
##     MMBi7             0.790    0.120    6.593    0.000    0.790    0.533
##     MMBi8             1.229    0.126    9.787    0.000    1.229    0.736
##     MMBi9             1.161    0.137    8.494    0.000    1.161    0.658
##
## Covariances:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##   Pressured ~~
##     Autonomous        0.587    0.064    9.126    0.000    0.587    0.587
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##    .MMBi1             0.782    0.114    6.830    0.000    0.782    0.249
##    .MMBi2             0.849    0.119    7.134    0.000    0.849    0.281
##    .MMBi3             0.682    0.101    6.762    0.000    0.682    0.242
##    .MMBi4             0.641    0.100    6.382    0.000    0.641    0.212
##    .MMBi5             1.244    0.156    7.989    0.000    1.244    0.446
##    .MMBi6             0.849    0.201    4.222    0.000    0.849    0.268
##    .MMBi7             1.568    0.193    8.123    0.000    1.568    0.715
##    .MMBi8             1.277    0.192    6.662    0.000    1.277    0.458
##    .MMBi9             1.766    0.236    7.477    0.000    1.766    0.567
##     Pressured         1.000                               1.000    1.000
##     Autonomous        1.000                               1.000    1.000
##
##
## -------------------------
## MODEL FIT INDICES
## -------------------------
##       Index Value Criteria
## cfi     cfi 0.901     0.95
```

```
## tli     tli 0.863     0.95
## rmsea rmsea 0.148     0.06
## srmr   srmr 0.099     0.08
##
## -------------------------
## STANDARDIZED FACTOR LOADINGS
## -------------------------
##       Factor  Item Std_Loading
## 1  Pressured MMBi1       0.867
## 2  Pressured MMBi2       0.848
## 3  Pressured MMBi3       0.870
## 4  Pressured MMBi4       0.887
## 5  Pressured MMBi5       0.745
## 6 Autonomous MMBi6       0.856
## 7 Autonomous MMBi7       0.533
## 8 Autonomous MMBi8       0.736
## 9 Autonomous MMBi9       0.658
##
## -------------------------
## TOP MODIFICATION INDICES
## -------------------------
##           lhs op   rhs     mi    epc sepc.lv sepc.all sepc.nox
## 60      MMBi5 ~~ MMBi9 31.894  0.750   0.750    0.506    0.506
## 64      MMBi7 ~~ MMBi8 24.886  0.717   0.717    0.506    0.506
## 28 Autonomous =~ MMBi3 15.682  0.434   0.434    0.259    0.259
## 23  Pressured =~ MMBi7 14.810 -0.560  -0.560   -0.378   -0.378
## 25  Pressured =~ MMBi9 13.735  0.610   0.610    0.345    0.345
```

Interpretation

- Allowing for a correlation between latent factors creates a better fit but still not sufficient
- Mod indices suggest A correlation between Item MMBi5 and MMBi9, This makes conceptually sense, because they both deal with anti femininity.

4.1.6 Two-Factor Correlated + Residuals CFA

- 2 latent factors
- correlation allowed between factors
- allow correlated residuals

```
model_twof_cor_mod <- "
  # latent variables
  Pressured =~ MMBi1 + MMBi2 + MMBi3 + MMBi4 + MMBi5
  Autonomous =~ MMBi6 + MMBi7 + MMBi8 + MMBi9

  Pressured ~~ Autonomous

  # theory-based correlated residuals
  MMBi3 ~~ MMBi4
  MMBi5 ~~ MMBi9
  MMBi7 ~~ MMBi8
"
```

```r
results_twof_cor_mod <- run_cfa_model(model_twof_cor_mod, df, model_name = "Two-Factor
 ↪  Correlated Modified", show_mi = TRUE)
```

```
##
## -------------------------
## MODEL SUMMARY
## -------------------------
## lavaan 0.6.15 ended normally after 26 iterations
##
##   Estimator                                         ML
##   Optimization method                           NLMINB
##   Number of model parameters                        22
##
##   Number of observations                           154
##
## Model Test User Model:
##
##   Test statistic                                49.124
##   Degrees of freedom                                23
##   P-value (Chi-square)                           0.001
##
## Model Test Baseline Model:
##
##   Test statistic                               923.528
##   Degrees of freedom                                36
##   P-value                                        0.000
##
## User Model versus Baseline Model:
##
##   Comparative Fit Index (CFI)                    0.971
##   Tucker-Lewis Index (TLI)                       0.954
##
## Loglikelihood and Information Criteria:
##
##   Loglikelihood user model (H0)              -2262.144
##   Loglikelihood unrestricted model (H1)      -2237.582
##
##   Akaike (AIC)                                4568.288
##   Bayesian (BIC)                              4635.101
##   Sample-size adjusted Bayesian (SABIC)       4565.468
##
## Root Mean Square Error of Approximation:
##
##   RMSEA                                          0.086
##   90 Percent confidence interval - lower         0.052
##   90 Percent confidence interval - upper         0.119
##   P-value H_0: RMSEA <= 0.050                    0.041
##   P-value H_0: RMSEA >= 0.080                    0.642
##
## 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
##   Pressured =~
##     MMBi1             1.502    0.116   12.902    0.000    1.502    0.847
##     MMBi2             1.450    0.115   12.608    0.000    1.450    0.835
##     MMBi3             1.513    0.108   14.031    0.000    1.513    0.902
##     MMBi4             1.590    0.110   14.413    0.000    1.590    0.915
##     MMBi5             1.154    0.111   10.378    0.000    1.154    0.711
##   Autonomous =~
##     MMBi6             1.547    0.129   12.023    0.000    1.547    0.869
##     MMBi7             0.690    0.123    5.585    0.000    0.690    0.466
##     MMBi8             1.165    0.128    9.136    0.000    1.165    0.698
##     MMBi9             1.211    0.132    9.200    0.000    1.211    0.679
##
## Covariances:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##   Pressured ~~
##     Autonomous        0.615    0.060   10.175    0.000    0.615    0.615
##  .MMBi3 ~~
##    .MMBi4            -0.228    0.078   -2.928    0.003   -0.228   -0.451
##  .MMBi5 ~~
##    .MMBi9             0.759    0.146    5.202    0.000    0.759    0.508
##  .MMBi7 ~~
##    .MMBi8             0.618    0.158    3.905    0.000    0.618    0.395
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##    .MMBi1             0.887    0.120    7.379    0.000    0.887    0.282
##    .MMBi2             0.917    0.121    7.555    0.000    0.917    0.303
##    .MMBi3             0.523    0.104    5.027    0.000    0.523    0.186
##    .MMBi4             0.489    0.105    4.644    0.000    0.489    0.162
##    .MMBi5             1.305    0.157    8.323    0.000    1.305    0.495
##    .MMBi6             0.778    0.209    3.720    0.000    0.778    0.245
##    .MMBi7             1.715    0.208    8.263    0.000    1.715    0.783
##    .MMBi8             1.430    0.201    7.102    0.000    1.430    0.513
##    .MMBi9             1.710    0.237    7.212    0.000    1.710    0.539
##     Pressured         1.000                               1.000    1.000
##     Autonomous        1.000                               1.000    1.000
##
##
## -------------------------
## MODEL FIT INDICES
## -------------------------
##       Index Value Criteria
## cfi     cfi 0.971     0.95
## tli     tli 0.954     0.95
## rmsea rmsea 0.086     0.06
## srmr   srmr 0.074     0.08
```

```
##
## -------------------------
## STANDARDIZED FACTOR LOADINGS
## -------------------------
##       Factor  Item Std_Loading
## 1  Pressured MMBi1       0.847
## 2  Pressured MMBi2       0.835
## 3  Pressured MMBi3       0.902
## 4  Pressured MMBi4       0.915
## 5  Pressured MMBi5       0.711
## 6 Autonomous MMBi6       0.869
## 7 Autonomous MMBi7       0.466
## 8 Autonomous MMBi8       0.698
## 9 Autonomous MMBi9       0.679
##
## -------------------------
## TOP MODIFICATION INDICES
## -------------------------
##           lhs op   rhs     mi    epc sepc.lv sepc.all sepc.nox
## 31 Autonomous =~ MMBi3 12.225  0.398   0.398    0.237    0.237
## 26  Pressured =~ MMBi7 10.161 -0.431  -0.431   -0.291   -0.291
## 30 Autonomous =~ MMBi2  9.785 -0.378  -0.378   -0.217   -0.217
## 28  Pressured =~ MMBi9  5.333  0.392   0.392    0.220    0.220
## 52      MMBi3 ~~ MMBi8  4.164  0.176   0.176    0.203    0.203
```

Interpretation

- …

4.2 Reliability

```r
fit_twof_cor_mod <- cfa(model_twof_cor_mod, data = df, std.lv = TRUE)
rel <- reliability(fit_twof_cor_mod)
print(rel)
```

```
##        Pressured Autonomous
## alpha  0.9249521  0.7899473
## omega  0.9341165  0.7559467
## omega2 0.9341165  0.7559467
## omega3 0.9141315  0.7697912
## avevar 0.7182846  0.5026560
```

Interpretation

- Pressured scale shows an excellent reliability (> .9)
- Autonomous scale shows an acceptable reliability (> .7) => reliability is sufficient, so we use the proposed factor structure to compute factor scored for autonomous and pressured motivation

Omega values .70 (acceptable), .80 (good). => Alpha assumes tau-equivalence (all items same loading), which isn't true here, so we trust omega more

4.3 Latent Trait Scores

4.3.1 Compute Scores

Use regression scoring from the chosen model, to compute latent trait scores

```
fs_reg <- lavPredict(fit_twof_cor_mod, method = "regression")

df$regPressured <- scale(fs_reg[, "Pressured"])
df$regAutonomous <- scale(fs_reg[, "Autonomous"])
```

Also compute Row means for visualizations

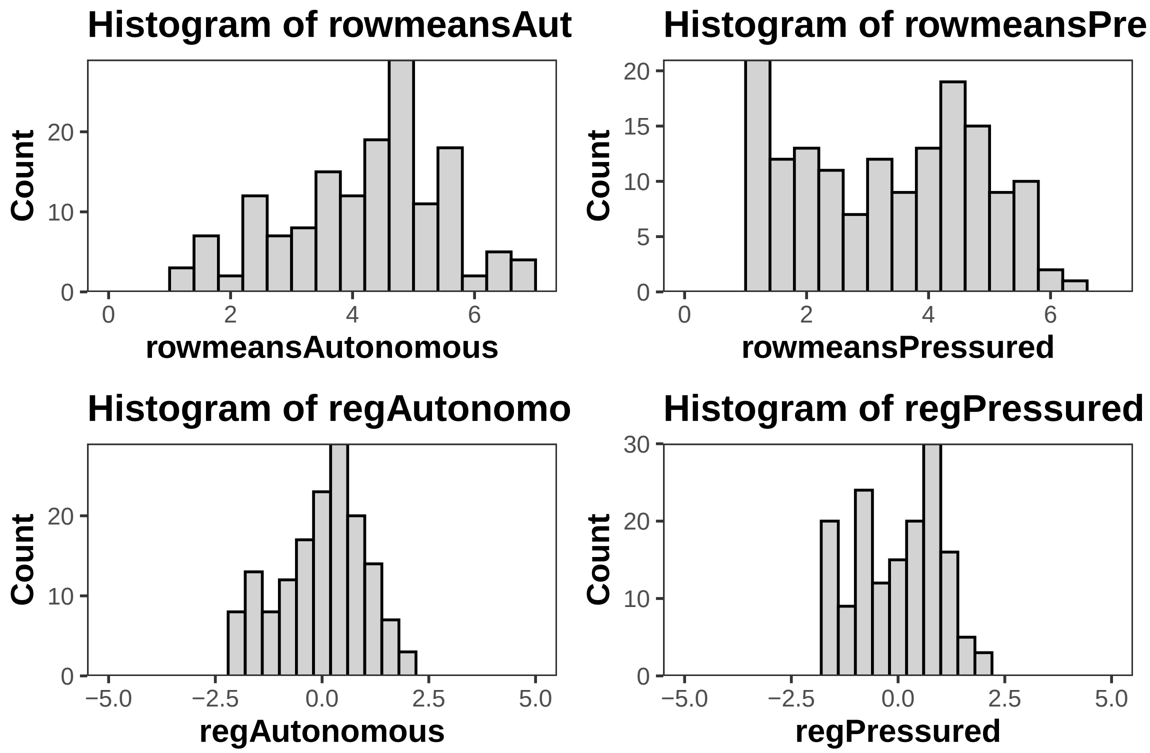
```
df$rowmeansAutonomous <- rowMeans(
  df[mmb_autonomous_items],
  na.rm = TRUE
)

df$rowmeansPressured <- rowMeans(
  df[mmb_pressured_items],
  na.rm = TRUE
)
```

4.3.2 Visualization & Descriptives

```
summary_mean_sd(df, c("rowmeansAutonomous", "rowmeansPressured", "regAutonomous",
↪  "regPressured"))
```

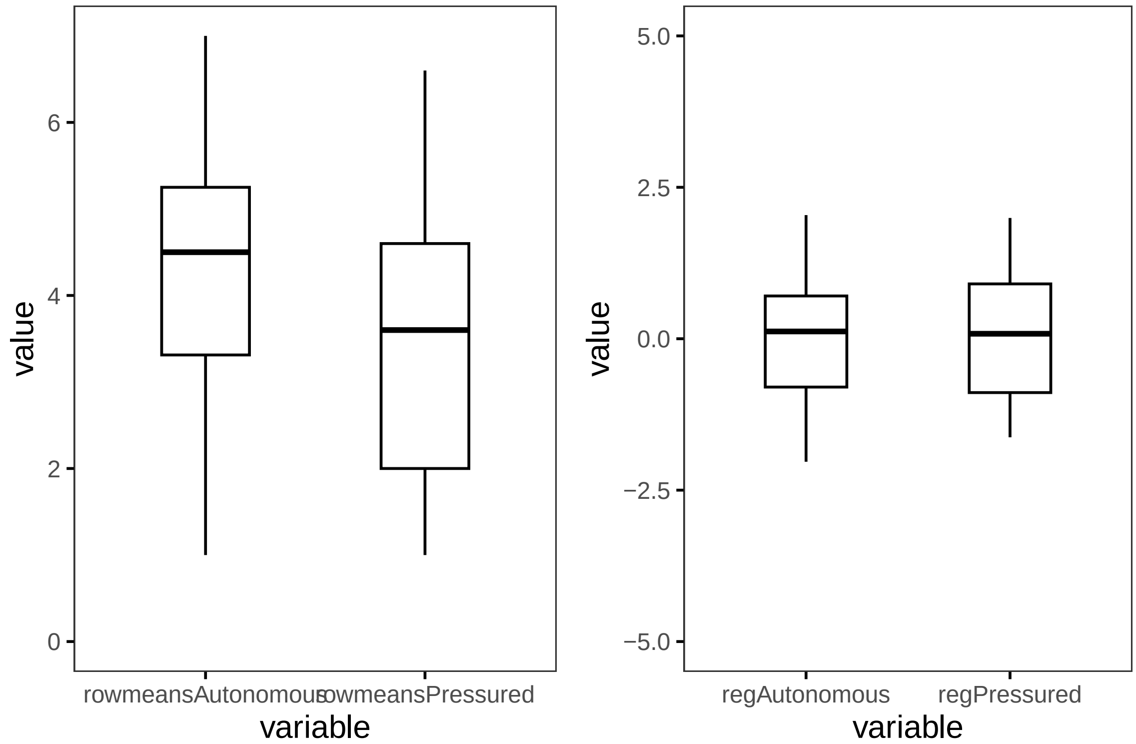
```
## # A tibble: 4 x 5
##   variable            mean    sd     n missing
##   <chr>              <dbl> <dbl> <dbl>   <dbl>
## 1 rowmeansAutonomous  4.23  1.32   154       0
## 2 rowmeansPressured   3.43  1.51   154       0
## 3 regAutonomous       0     1      154       0
## 4 regPressured        0     1      154       0
```

```
rm_aut_hist <- plotHist("rowmeansAutonomous", c(0, 7))
rm_pres_hist <- plotHist("rowmeansPressured", c(0, 7))
reg_aut_hist <- plotHist("regAutonomous", c(-5, 5))
reg_pres_hist <- plotHist("regPressured", c(-5, 5))

grid.arrange(rm_aut_hist, rm_pres_hist, reg_aut_hist, reg_pres_hist, ncol = 2)
```



-> regression scoring reduces measurement error at the price of losing variance

```
rm_box <- plotBox(c("rowmeansAutonomous", "rowmeansPressured"))
# ggpar(rm_box, ylim = c(0, 7))

reg_box <- plotBox(c("regAutonomous", "regPressured"))
# ggpar(rm_box, ylim = c(-5, 5))


grid.arrange(ggpar(rm_box, ylim = c(0, 7)), ggpar(reg_box, ylim = c(-5, 5)), ncol = 2)
```




```
cor(df$regAutonomous, df$regPressured)
```

```
##           [,1]
## [1,] 0.6702902
```

```
cor(df$rowmeansAutonomous, df$rowmeansPressured)
```

```
## [1] 0.4914834
```

Interpretation

- moderate to high correlation between pressured and autonomous factor
- the correlation increases with the regression scores (because factors are modeled as correlative)

5. Test Quality Criteria: Aggressive Cognition

Objectives:

- Assess the *validity* and *reliability* of the scale for aggressive cognition
- Compute participant latent trait scores

Methodology:

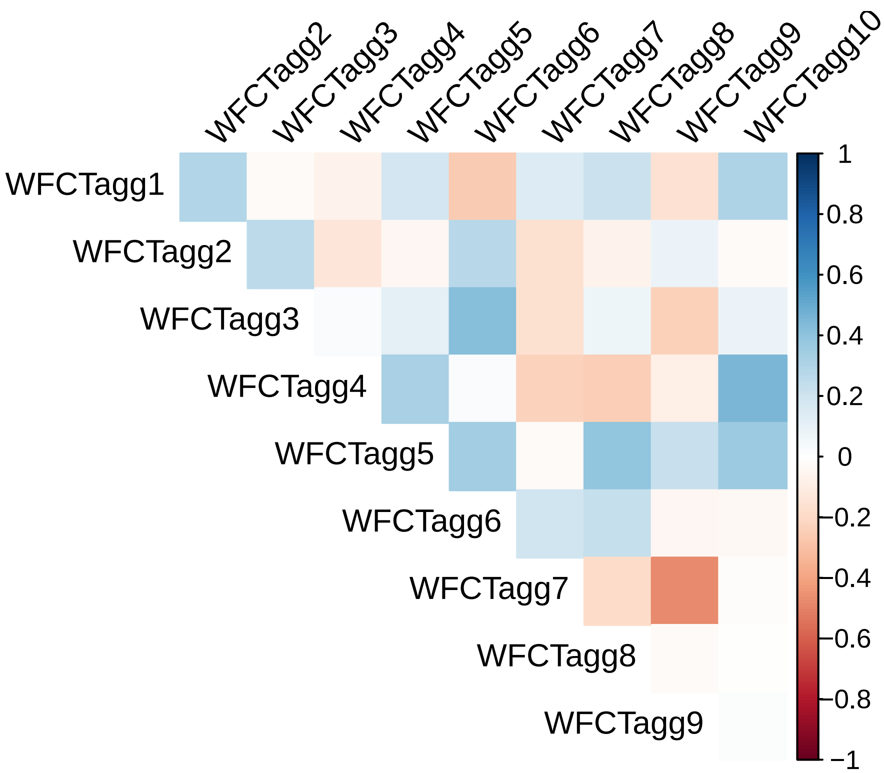
- Visualization & Descriptives

5.1 Construct Validity

5.1.1 Visualization & Descriptives

```
cor_aggr <- psych::tetrachoric( df[, WFCT_aggression_items])$rho
```

```
corrplot::corrplot(cor_aggr,
  method = "color",
  type = "upper",
  order = "original",
  tl.col = "black",
  tl.srt = 45,
  diag = FALSE
)
```



-> a tetrachoric correlation is used, because responses are always binary (aggressive, vs. non-aggressive response)

Interpretation

- The correlation plot shows no clear pattern between the items
- If all items were to measure the same latent construct (aggressive cognition) we would expect (somewhat) positive correlations between all the items

```
summary_mean_sd(df, WFCT_aggression_items)
```

```
## # A tibble: 10 x 5
##    variable   mean    sd     n missing
##    <chr>     <dbl> <dbl> <dbl>   <dbl>
##  1 WFCTagg1   0.05  0.21   146       8
##  2 WFCTagg2   0.13  0.34   150       4
##  3 WFCTagg3   0.07  0.26   149       5
##  4 WFCTagg4   0.08  0.27   148       6
##  5 WFCTagg5   0.03  0.18   151       3
##  6 WFCTagg6   0.51  0.5    152       2
##  7 WFCTagg7   0.1   0.3    143      11
##  8 WFCTagg8   0.41  0.49   140      14
##  9 WFCTagg9   0.23  0.42   142      12
## 10 WFCTagg10  0.64  0.48   141      13
```

Interpretation

- *Low Means:* On items 1, 2, 3, 4, 5, 7
- *Missing Values:* 10% missings on items 7, 8, 9, 10

5.1.2 Check sampling adequacy for Factor Analysis

```
KMO(cor_aggr)
```

```
## Kaiser-Meyer-Olkin factor adequacy
## Call: KMO(r = cor_aggr)
## Overall MSA =  0.16
## MSA for each item =
##  WFCTagg1  WFCTagg2  WFCTagg3  WFCTagg4  WFCTagg5  WFCTagg6  WFCTagg7  WFCTagg8
##      0.20      0.14      0.14      0.13      0.32      0.17      0.11      0.10
##  WFCTagg9 WFCTagg10
##      0.12      0.50
```

```
cortest.bartlett(cor_aggr, n = nrow(df))
```

```
## $chisq
## [1] 623.5155
##
## $p.value
## [1] 7.520443e-103
##
## $df
## [1] 45
```

KMO (Kaiser-Meyer-Olkin) should be > 0.6 (ideally > 0.8) Bartlett's test should be significant (p < .05)

Interpretation

- The KMO is extremely low, indicating that the items are not suited for a factor analysis
- This is because the items share very little variance (are not correlating)

5.2 Reliability

```r
psych::alpha(cor_aggr, check.keys = FALSE)
```

```
## Some items ( WFCTagg7 ) were negatively correlated with the first principal component and
## probably should be reversed.
## To do this, run the function again with the 'check.keys=TRUE' option
```

```
##
## Reliability analysis
## Call: psych::alpha(x = cor_aggr, check.keys = FALSE)
##
##   raw_alpha std.alpha G6(smc) average_r  S/N median_r
##       0.36      0.36    0.78     0.053 0.55     0.01
##
##     95% confidence boundaries
##       lower alpha upper
## Feldt -0.46  0.36  0.81
##
##  Reliability if an item is dropped:
##           raw_alpha std.alpha G6(smc) average_r  S/N var.r   med.r
## WFCTagg1       0.31      0.31    0.72     0.048 0.45 0.047  0.0025
## WFCTagg2       0.33      0.33    0.72     0.052 0.50 0.049  0.0180
## WFCTagg3       0.32      0.32    0.71     0.050 0.47 0.047 -0.0106
## WFCTagg4       0.38      0.38    0.68     0.063 0.61 0.043  0.0409
## WFCTagg5       0.11      0.11    0.67     0.014 0.12 0.043 -0.0197
## WFCTagg6       0.24      0.24    0.62     0.033 0.31 0.044 -0.0106
## WFCTagg7       0.47      0.47    0.67     0.091 0.90 0.040  0.0497
## WFCTagg8       0.34      0.34    0.65     0.055 0.52 0.047  0.0180
## WFCTagg9       0.46      0.46    0.72     0.086 0.84 0.042  0.0497
## WFCTagg10      0.24      0.24    0.76     0.034 0.32 0.047  0.0011
##
##  Item statistics
##               r   r.cor r.drop
## WFCTagg1  0.426  0.3756  0.180
## WFCTagg2  0.385  0.3209  0.134
## WFCTagg3  0.410  0.3519  0.162
## WFCTagg4  0.281  0.2471  0.022
## WFCTagg5  0.750  0.7454  0.595
## WFCTagg6  0.566  0.5742  0.347
## WFCTagg7  0.024 -0.0200 -0.230
## WFCTagg8  0.365  0.3440  0.111
## WFCTagg9  0.075  0.0081 -0.183
## WFCTagg10 0.557  0.4518  0.336
```

Interpretation

5.3 Latent Trait Scores

5.3.1 Compute Scores

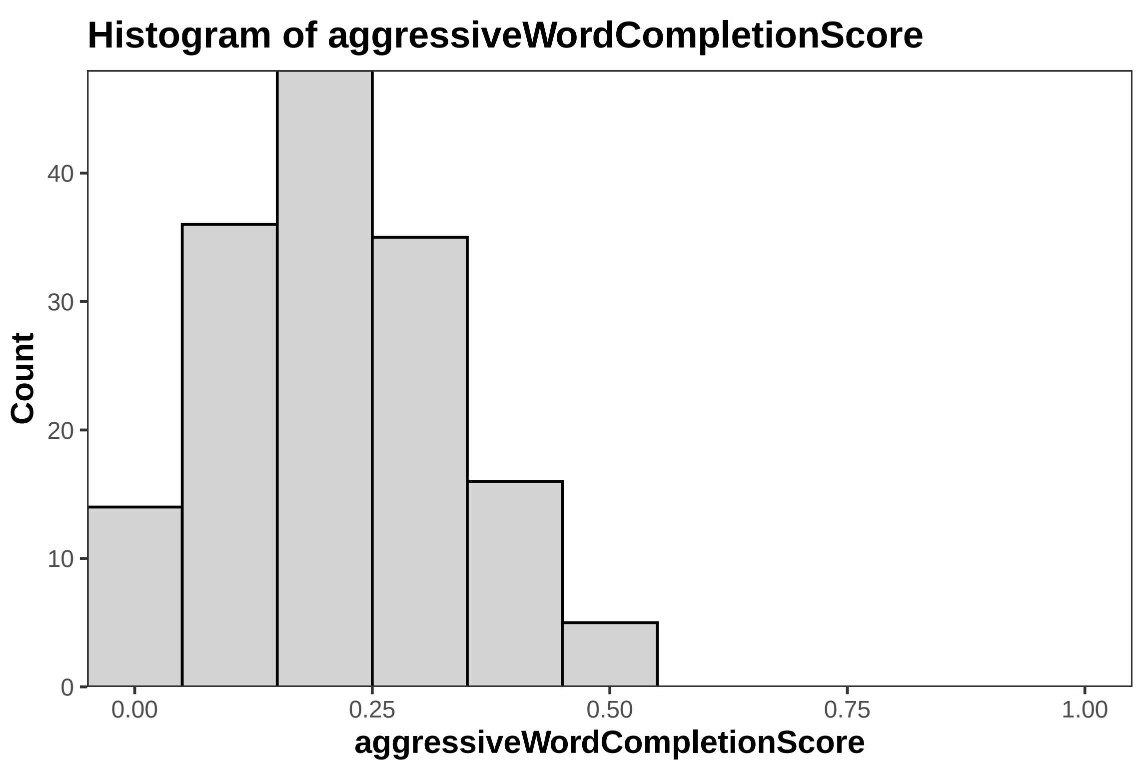
```r
df$aggressiveWordCompletionScore <- rowMeans(
  replace(df[WFCT_aggression_items], is.na(df[WFCT_aggression_items]), 0)
)

df$aggressiveWordCompletionScoreTransformed <-
↪  asin(sqrt(df$aggressiveWordCompletionScore))
```

5.3.2 Visualization & Descriptives

```r
summary_mean_sd(df, c("aggressiveWordCompletionScore"))
```

```
## # A tibble: 1 x 5
##   variable                       mean    sd     n missing
##   <chr>                         <dbl> <dbl> <dbl>   <dbl>
## 1 aggressiveWordCompletionScore  0.21  0.12   154       0
```

```r
plotHist("aggressiveWordCompletionScore", c(0, 1), condition = NULL, binwidth = 0.1)
```



Summary

…

6. Test Quality Criteria: Anxious Cognition

Objectives:

- Assess the *validity* and *reliability* of the scale for anxious cognition
- Compute participant latent trait scores

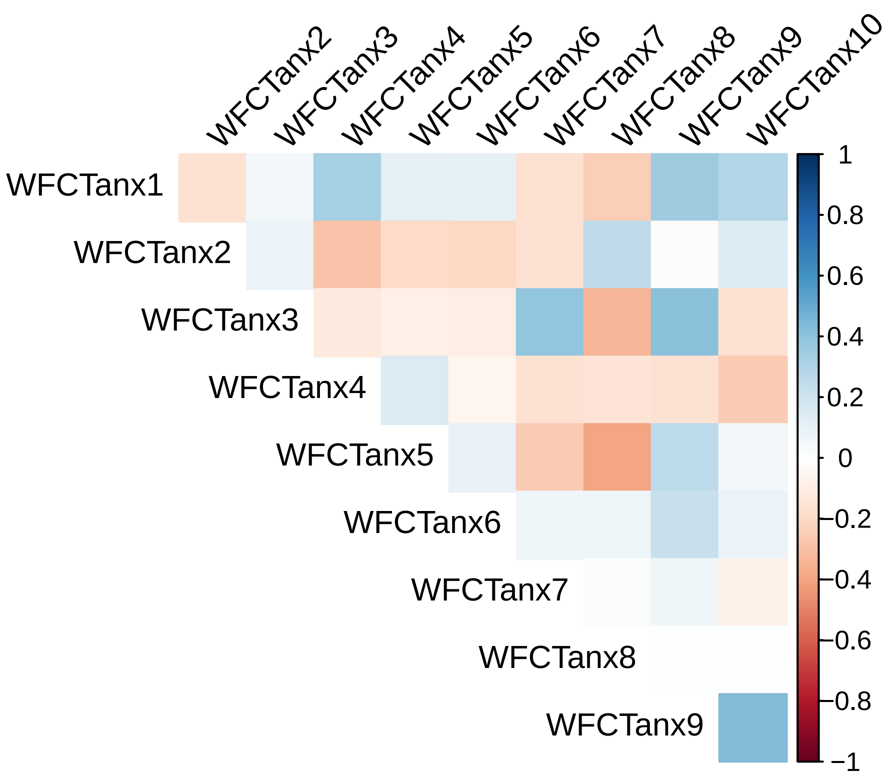
Methodology:

- Descriptives * Visualization

6.1 Construct Validity

```
cor_anx <- psych::tetrachoric(df[, WFCT_anxiety_items])$rho
```

```
corrplot::corrplot(cor_anx,
  method = "color",
  type = "upper",
  order = "original",
  tl.col = "black",
  tl.srt = 45,
  diag = FALSE
)
```



-> a tetrachoric correlation is used, because responses are always binary (aggressive, vs. non-aggressive)

6.1.1 Visualization & Descriptives

```r
summary_mean_sd(df, WFCT_anxiety_items)
```

```
## # A tibble: 10 x 5
##    variable   mean    sd     n missing
##    <chr>     <dbl> <dbl> <dbl>   <dbl>
##  1 WFCTanx1   0.31  0.47   144      10
##  2 WFCTanx2   0.03  0.18   145       9
##  3 WFCTanx3   0.85  0.36   108      46
##  4 WFCTanx4   0.26  0.44   145       9
##  5 WFCTanx5   0.33  0.47   140      14
##  6 WFCTanx6   0.36  0.48   148       6
##  7 WFCTanx7   0.16  0.37   144      10
##  8 WFCTanx8   0.08  0.27   151       3
##  9 WFCTanx9   0.86  0.35   105      49
## 10 WFCTanx10  0.17  0.37   144      10
```

Interpretation

- *Low Means:*
- *Missing Values:* …

Interpretation

- …

6.1.2 Check sampling adequacy for Factor Analysis

```r
KMO(cor_anx)
```

```
## Kaiser-Meyer-Olkin factor adequacy
## Call: KMO(r = cor_anx)
## Overall MSA =  0.3
## MSA for each item =
##  WFCTanx1  WFCTanx2  WFCTanx3  WFCTanx4  WFCTanx5  WFCTanx6  WFCTanx7  WFCTanx8
##      0.48      0.35      0.22      0.53      0.27      0.47      0.43      0.22
##  WFCTanx9 WFCTanx10
##      0.26      0.27
```

```r
cortest.bartlett(cor_anx, n = nrow(df))
```

```
## $chisq
## [1] 432.2395
##
## $p.value
## [1] 1.010417e-64
##
## $df
## [1] 45
```

KMO (Kaiser-Meyer-Olkin) should be > 0.6 (ideally > 0.8) Bartlett's test should be significant ($p < .05$)

Interpretation

- The KMO is extremely low, indicating that the items are not suited for a factor analysis
- This is because the items share very little variance …

6.2 Reliability

```r
psych::alpha(cor_anx, check.keys = FALSE)
```

```
## Some items ( WFCTanx2 WFCTanx7 WFCTanx8 ) were negatively correlated with the first principal compone
## probably should be reversed.
## To do this, run the function again with the 'check.keys=TRUE' option
```

```
##
## Reliability analysis
## Call: psych::alpha(x = cor_anx, check.keys = FALSE)
##
##   raw_alpha std.alpha G6(smc) average_r   S/N median_r
##      0.086     0.086    0.51    0.0093 0.094   0.0075
##
##     95% confidence boundaries
##       lower alpha upper
## Feldt -1.08  0.09  0.73
##
##  Reliability if an item is dropped:
##           raw_alpha std.alpha G6(smc) average_r    S/N var.r   med.r
## WFCTanx1     -0.075    -0.075   0.405   -0.0078 -0.069 0.043 -0.0051
## WFCTanx2      0.195     0.195   0.548    0.0263  0.243 0.047  0.0345
## WFCTanx3      0.058     0.058   0.338    0.0068  0.062 0.042  0.0090
## WFCTanx4      0.222     0.222   0.566    0.0307  0.285 0.045  0.0589
## WFCTanx5      0.145     0.145   0.487    0.0185  0.170 0.045  0.0074
## WFCTanx6      0.029     0.029   0.509    0.0033  0.030 0.052 -0.0051
## WFCTanx7      0.146     0.146   0.531    0.0186  0.171 0.047  0.0329
## WFCTanx8      0.235     0.235   0.494    0.0330  0.307 0.043  0.0589
## WFCTanx9     -0.403    -0.403   0.063   -0.0329 -0.287 0.037 -0.0644
## WFCTanx10    -0.034    -0.034   0.386   -0.0036 -0.032 0.045 -0.0050
##
##  Item statistics
##               r  r.cor r.drop
## WFCTanx1  0.516  0.480  0.240
## WFCTanx2  0.144 -0.081 -0.160
## WFCTanx3  0.356  0.381  0.056
## WFCTanx4  0.095 -0.145 -0.206
## WFCTanx5  0.228  0.105 -0.077
## WFCTanx6  0.394  0.195  0.098
## WFCTanx7  0.227  0.039 -0.079
## WFCTanx8  0.070 -0.073 -0.228
## WFCTanx9  0.791  0.991  0.623
## WFCTanx10 0.470  0.461  0.185
```

6.3 Latent Trait Scores

6.3.1 Compute Scores

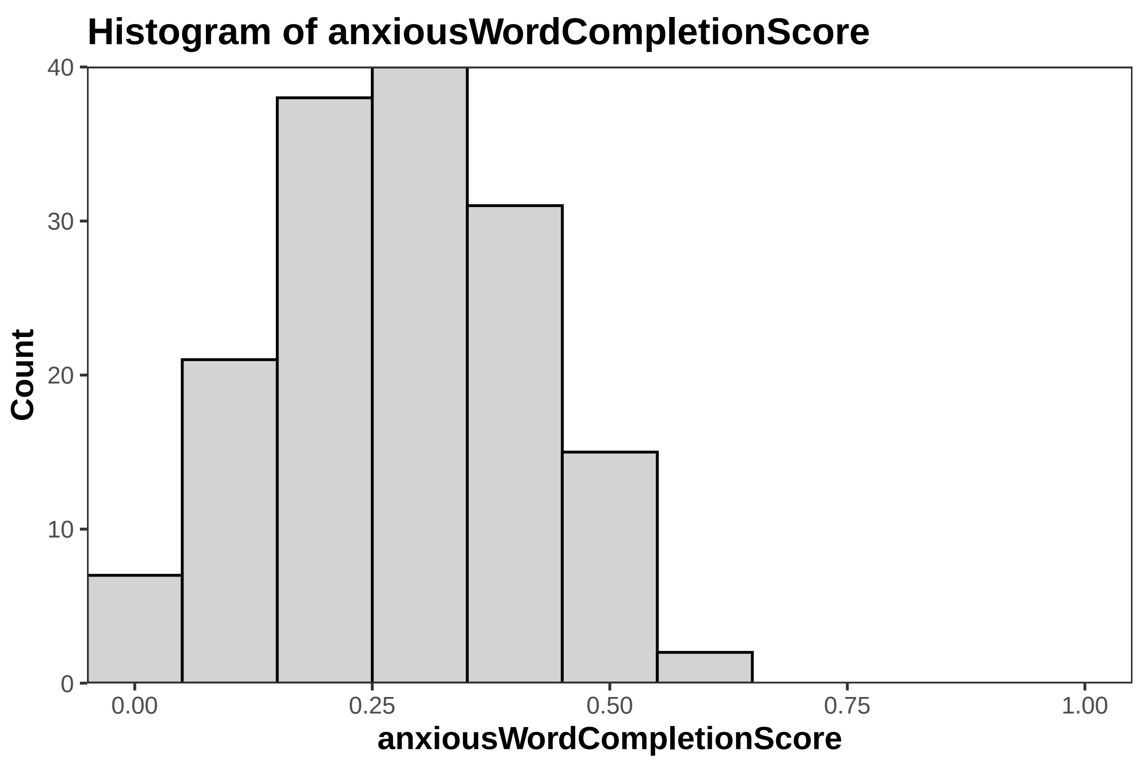
```
df$anxiousWordCompletionScore <- rowMeans(
  replace(df[WFCT_anxiety_items], is.na(df[WFCT_anxiety_items]), 0)
)

df$anxiousWordCompletionScoreTransformed <- asin(sqrt(df$anxiousWordCompletionScore))
```

6.3.2 Visualization & Descriptives

```
summary_mean_sd(df, c("anxiousWordCompletionScore"))
```

```
## # A tibble: 1 x 5
##   variable                    mean    sd     n missing
##   <chr>                      <dbl> <dbl> <dbl>   <dbl>
## 1 anxiousWordCompletionScore  0.28  0.14   154       0
```

```
plotHist("anxiousWordCompletionScore", c(0, 1), condition = NULL, binwidth = 0.1)
```



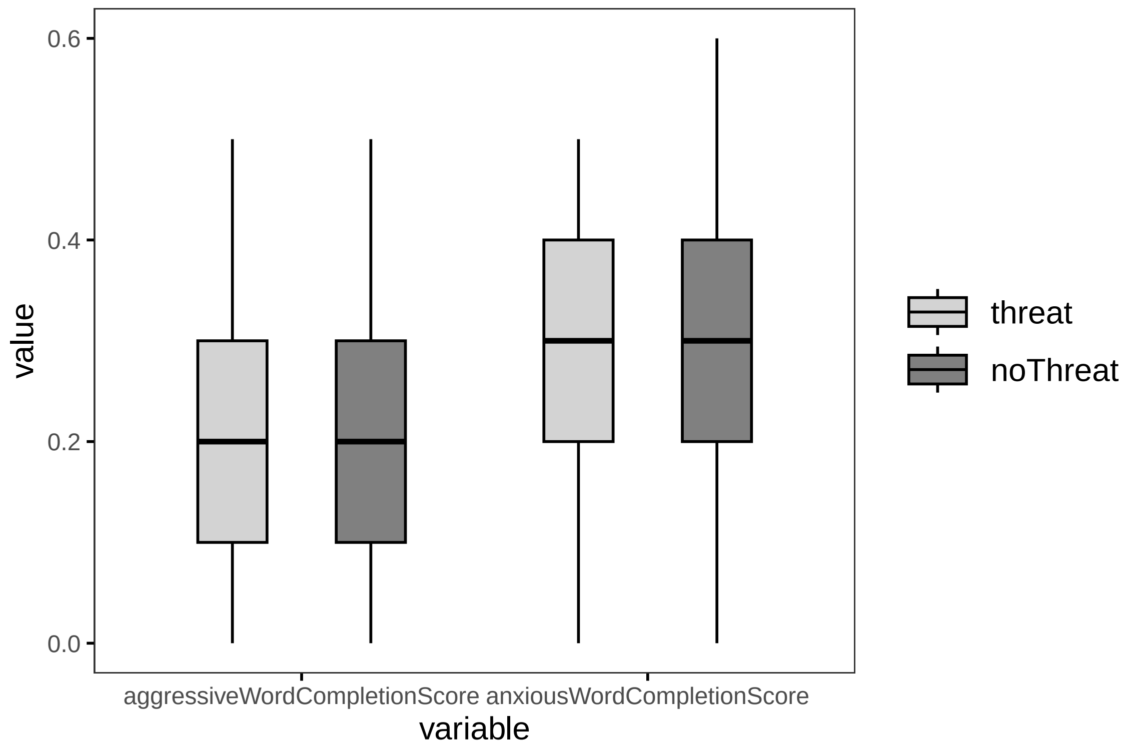
Summary

Hypothesis 1a & 1b

Visualization

Aggressive and anxious word completions by threat condition

```
plotBox(
  c("aggressiveWordCompletionScore", "anxiousWordCompletionScore"),
  "threatCondition"
)
```




```
summary_mean_sd(df, c("aggressiveWordCompletionScore", "anxiousWordCompletionScore"))
```

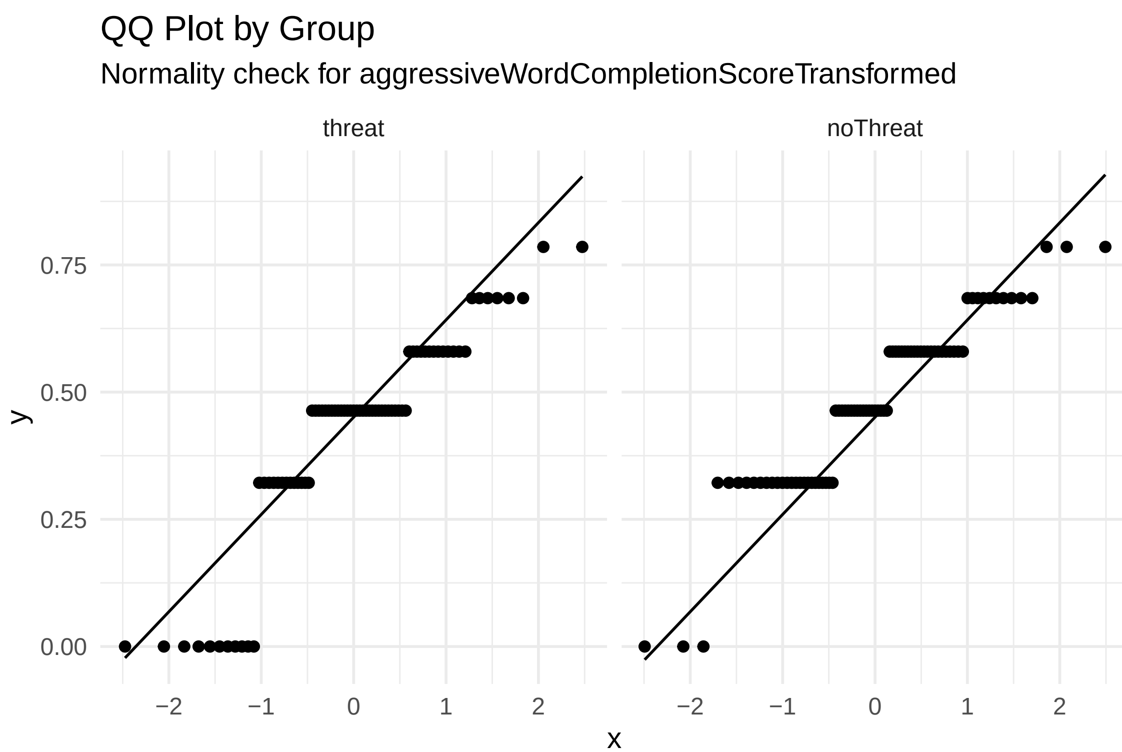
```
## # A tibble: 2 x 5
##   variable                       mean    sd     n missing
##   <chr>                         <dbl> <dbl> <dbl>   <dbl>
## 1 aggressiveWordCompletionScore  0.21  0.12   154       0
## 2 anxiousWordCompletionScore     0.28  0.14   154       0
```

Hypothesis 1a t-test

Assumption 1: Normality withing each group

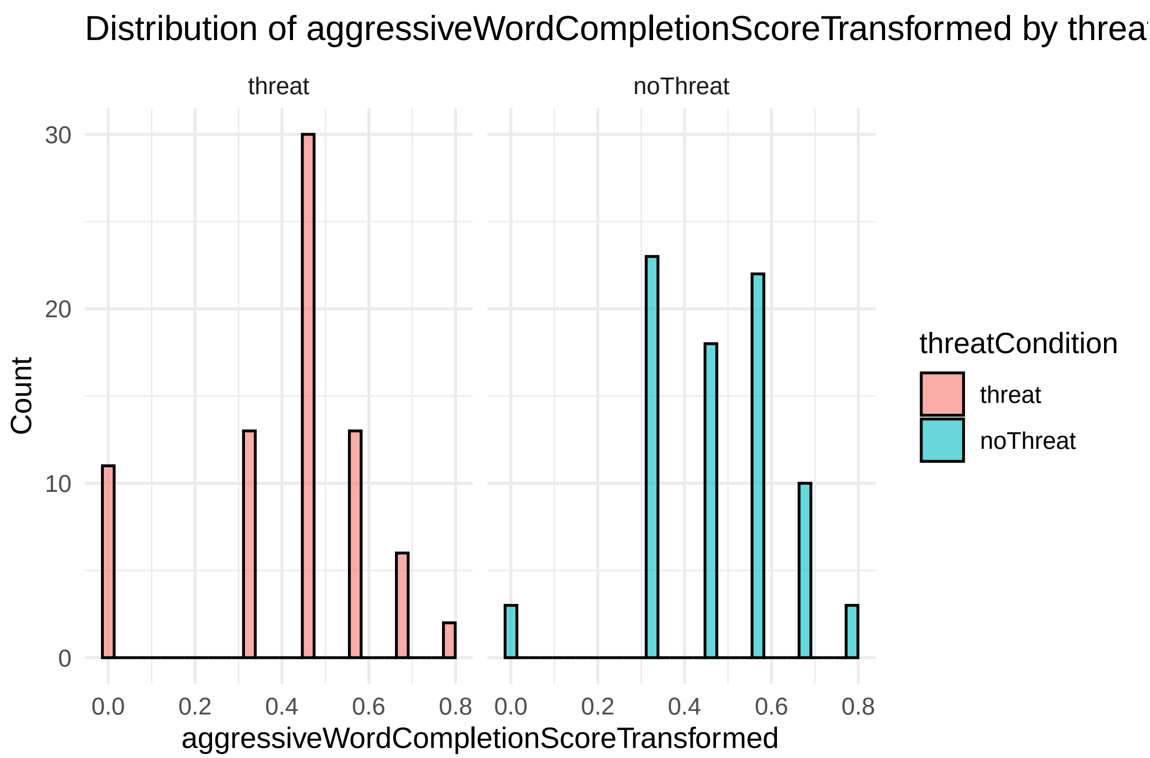
Check Visually

```r
check_normality_qq(
  df, aggressiveWordCompletionScoreTransformed,
  threatCondition
)
```



-> data points should be roughly on the line

```r
check_normality_hist(
  df, aggressiveWordCompletionScoreTransformed,
  threatCondition
)
```



-> should roughly resemble a normal distribution

Assumption 2: Homogeneity of variance

Levene Test

```
check_homogeneity(
  df, aggressiveWordCompletionScoreTransformed,
  threatCondition
)
```

```
## Levene's Test for Homogeneity of Variance (center = median)
##        Df F value Pr(>F)
## group   1  0.1237 0.7256
##       152
```

p > .05 → equal variances (use Student's t-test: var.equal = TRUE)

```
summary_mean_sd(df, "aggressiveWordCompletionScore", group = "threatCondition")
```

```
## # A tibble: 2 x 6
##   threatCondition variable                      mean    sd     n missing
##   <fct>           <chr>                        <dbl> <dbl> <dbl>   <dbl>
## 1 threat          aggressiveWordCompletionScore  0.19  0.12    75       0
## 2 noThreat        aggressiveWordCompletionScore  0.23  0.12    79       0
```

```r
run_ttest(df, aggressiveWordCompletionScoreTransformed, threatCondition,
  alternative = "greater", var_equal = TRUE
)
```

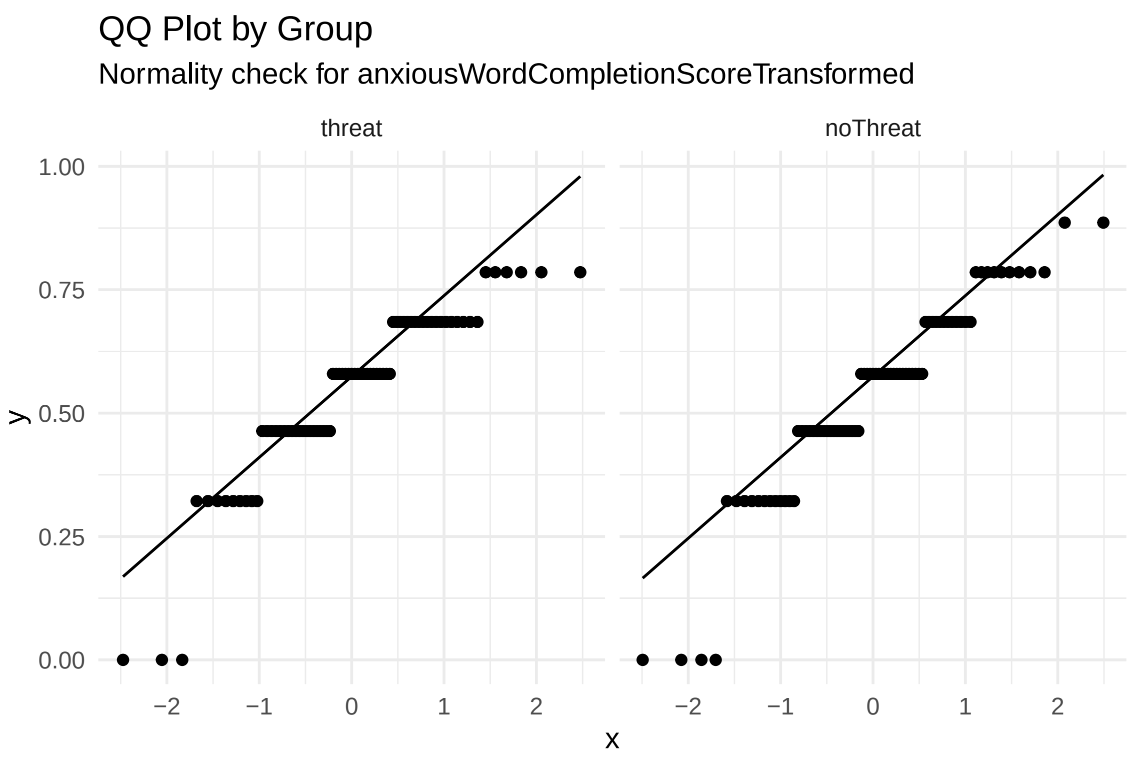
```
##
##  Two Sample t-test
##
## data:  aggressiveWordCompletionScoreTransformed by threatCondition
## t = -1.9812, df = 152, p-value = 0.9753
## alternative hypothesis: true difference in means between group threat and group noThreat is greater t
## 95 percent confidence interval:
##  -0.1097734        Inf
## sample estimates:
##   mean in group threat mean in group noThreat
##              0.4174215              0.4772327
```

Hypothesis 1b t-test

Assumption 1: Normality withing each group

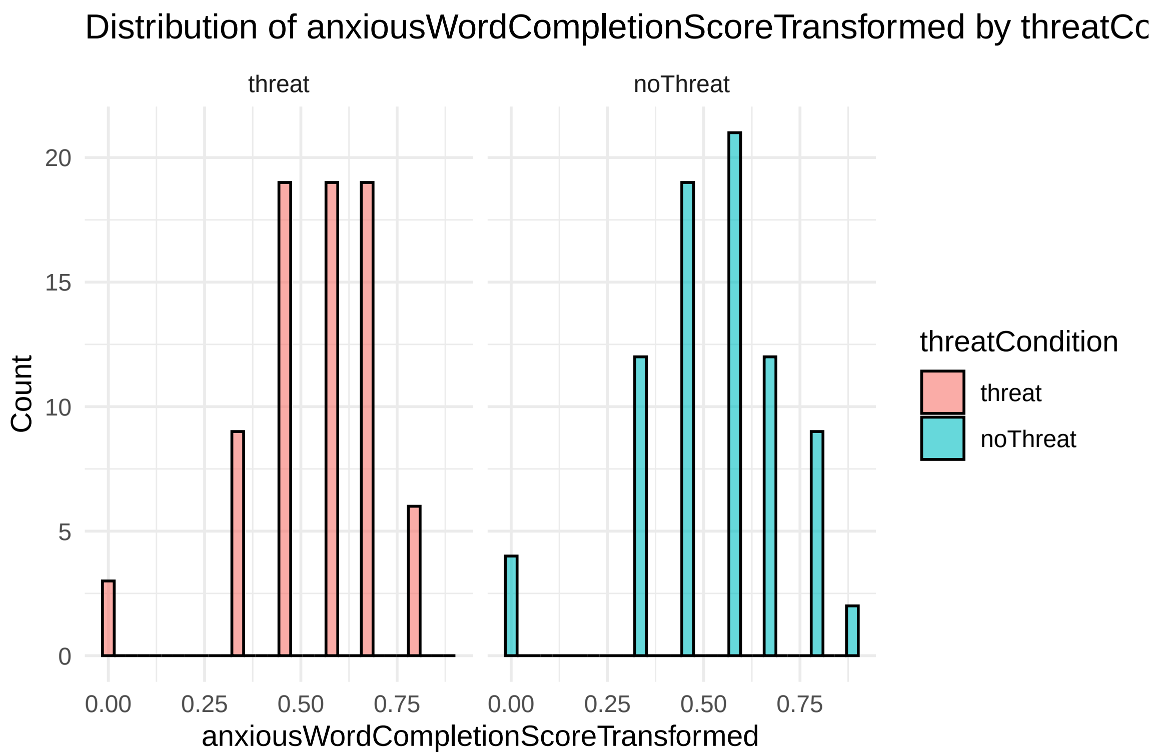
Check Visually

```r
check_normality_qq(df, anxiousWordCompletionScoreTransformed, threatCondition)
```



-> data points should be roughly on the line

```r
check_normality_hist(
  df, anxiousWordCompletionScoreTransformed,
  threatCondition
)
```



-> should roughly resemble a normal distribution

Assumption 2: Homogeneity of variance

Levene Test

```r
check_homogeneity(df, anxiousWordCompletionScoreTransformed, threatCondition)
```

```
## Levene's Test for Homogeneity of Variance (center = median)
##        Df F value Pr(>F)
## group   1    0.65 0.4214
##       152
```

p > .05 → equal variances (use Student's t-test: var.equal = TRUE)

```r
summary_mean_sd(df, "anxiousWordCompletionScore", group = "threatCondition")
```

```
## # A tibble: 2 x 6
##   threatCondition variable                    mean    sd     n missing
##   <fct>           <chr>                      <dbl> <dbl> <dbl>   <dbl>
## 1 threat          anxiousWordCompletionScore  0.28  0.13    75       0
## 2 noThreat        anxiousWordCompletionScore  0.28  0.15    79       0
```

Run inference statistic

```
run_ttest(df, anxiousWordCompletionScoreTransformed, threatCondition,
  alternative = "greater", var_equal = TRUE
)
```

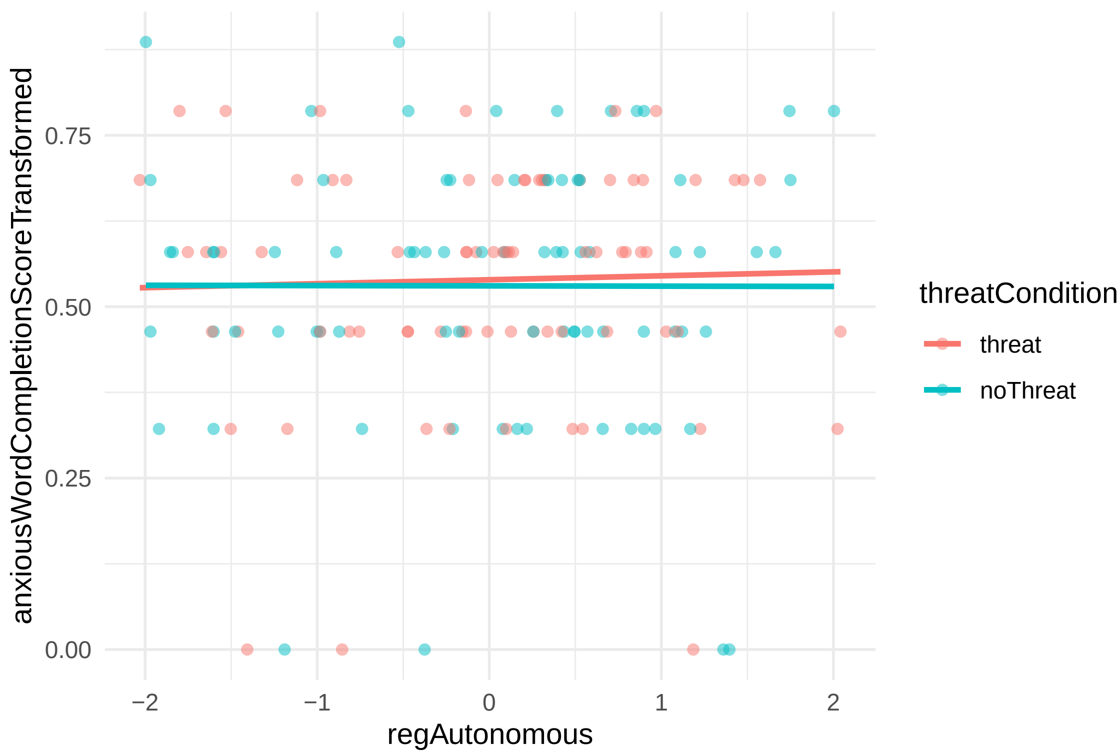
```
##
##  Two Sample t-test
##
## data:  anxiousWordCompletionScoreTransformed by threatCondition
## t = 0.29844, df = 152, p-value = 0.3829
## alternative hypothesis: true difference in means between group threat and group noThreat is greater t
## 95 percent confidence interval:
##  -0.0401006        Inf
## sample estimates:
##   mean in group threat mean in group noThreat
##              0.5392036              0.5303812
```

Hypothesis 2a & 2b: Moderation of Threat Response

Visualization

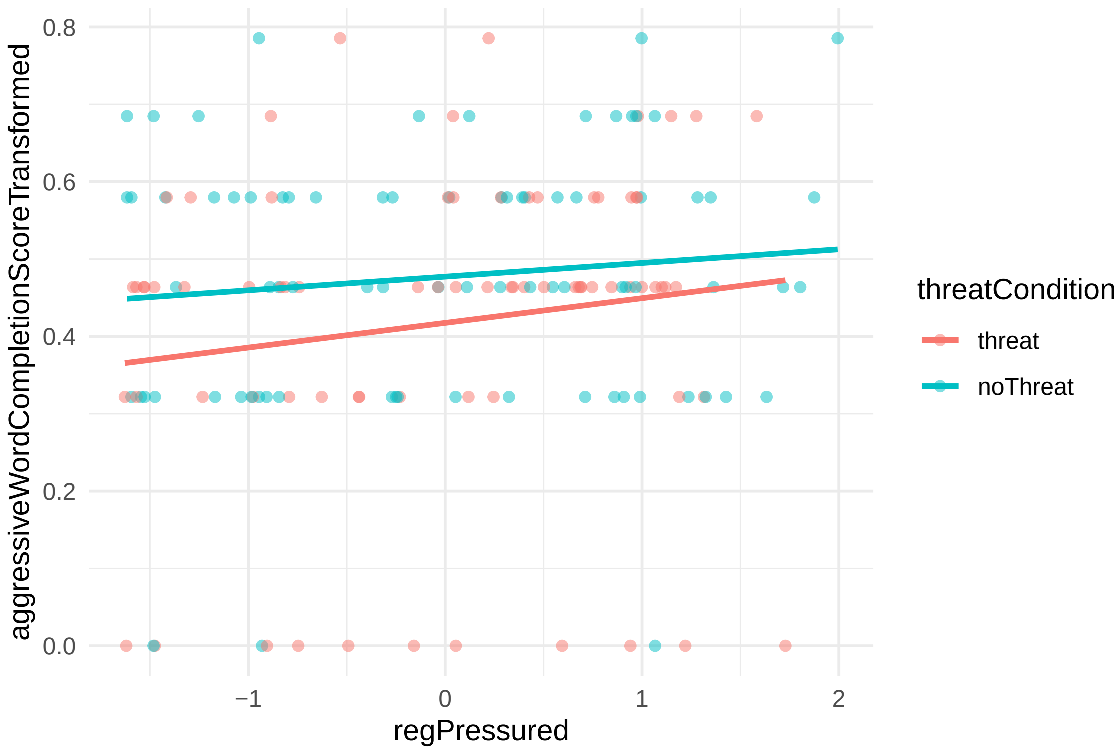
2a: Moderation autonomous motivation for masculine behavior on aggressive word completions

```
plotLine("regAutonomous", "anxiousWordCompletionScoreTransformed",
  condition = "threatCondition", df = df
)
```



2b: Moderation pressured motivation for masculine behavior on aggressive word completions

```
plotLine("regPressured", "aggressiveWordCompletionScoreTransformed",
  condition = "threatCondition", df = df
)
```



Hypothesis 2a

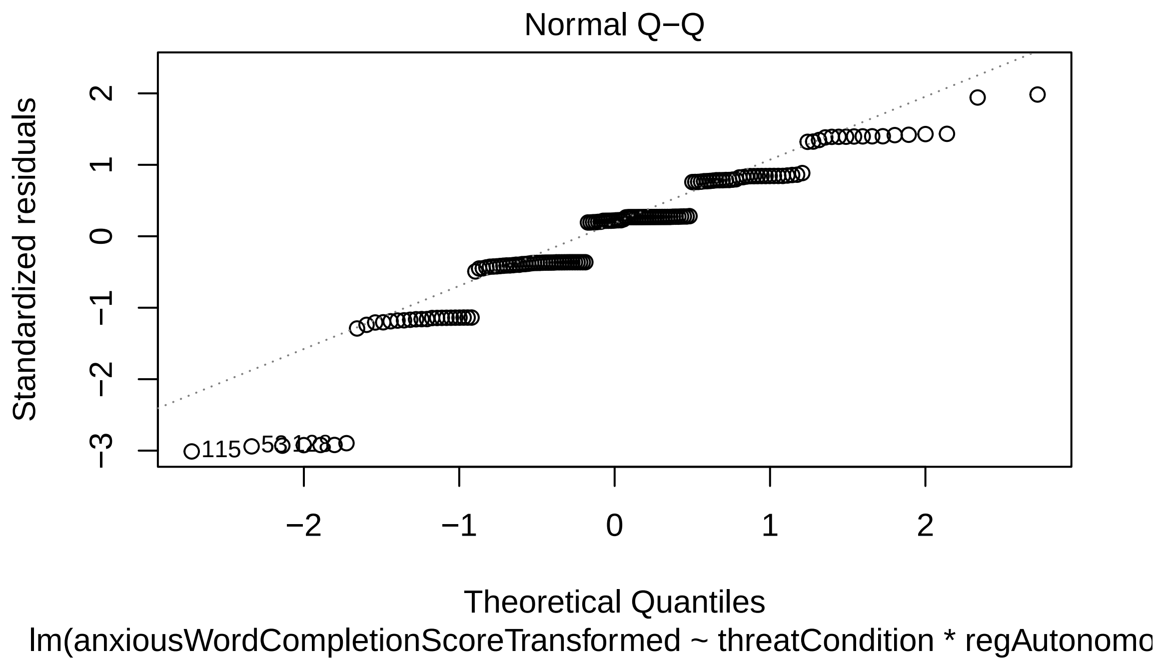
Multiple linear regression

Establish model

```
model_anxious <- lm(anxiousWordCompletionScoreTransformed ~
  threatCondition * regAutonomous, data = df)
```

```
plot(model_anxious, which = 2)
```

Assumption 1: Normality



-> Q-Q plot points roughly on the line → residuals ~ normal. -> identify possible outliers (highly influential points)

Assumption 2: Outliers Examine outliers via residuals

```
res <- residuals(model_anxious)

# Standardized residuals
res_std <- rstandard(model_anxious)

# Flag extreme residuals
outliers <- which(abs(res_std) > 2)
outliers
```

```
##  35  53  73  96 115 128 141
##  35  53  73  96 115 128 141
```

Examine outliers via cook's distance

```
influential_obs <- df
influential_obs$cooksd <- cooks.distance(model_anxious)

n <- nrow(df)
k <- length(coef(model_anxious)) - 1
threshold <- 4 / (n - k - 1)
```

```r
# influential_obs %>%
# filter(cooksd > threshold)
```

Cook's distance > 4/(n-k-1) → potentially influential

```r
cor(df$regAutonomous, as.numeric(factor(df$threatCondition,
  levels = c("noThreat", "threat")
)) - 1)
```

Assumption 3: No Multikollinearity

```
##             [,1]
## [1,] -0.01320326
```

-> would also make no sense, since I randomly assign threatCondition

Inference statistics

```r
summary(model_anxious)
```

```
##
## Call:
## lm(formula = anxiousWordCompletionScoreTransformed ~ threatCondition *
##     regAutonomous, data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.54610 -0.07455  0.04050  0.14365  0.35547
##
## Coefficients:
##                                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)                          0.539281   0.021311  25.305   <2e-16 ***
## threatConditionnoThreat             -0.008895   0.029754  -0.299    0.765
## regAutonomous                        0.005748   0.022312   0.258    0.797
## threatConditionnoThreat:regAutonomous -0.006172   0.030009  -0.206    0.837
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1845 on 150 degrees of freedom
## Multiple R-squared:  0.001031,   Adjusted R-squared:  -0.01895
## F-statistic: 0.05158 on 3 and 150 DF,  p-value: 0.9845
```

Post Hoc Analysis

…