MID - Measurement Model Code

by Michael Demidenko

November 2022

Contents

The platform used to write and generate this .html document is `cat(platform)` on RStudio, `cat(r_info)`. The required packages to run all necessary code chunks are checked and installed using the pacman R package. `p_load` from the package is used to load and install all listed packages. The details for a number of packages are described before their use. If they are not, the `.rmd` file will provide the list of packages that are installed.

Simulating data

This section is unique only to the Stage 1 registered reported. We are not using any real data, so to ensure that each step is completed correctly, we use simulated data. We can operate under the assumption that all participants will have no missing values given how exclusion criteria are applied and fMRI data are extracted from the BOLD timeseries.

In this section, a population model is fit reflecting the approach and avoidance (measured constructs) that are [presumed] to be associated with stimuli within/across tasks (Manifest variables). As formalized here, in the MID task the contrasts reflect the manifest variables (i.e., reflective model) as opposed of the other way, whereby the items form the variables predict the construct (i.e., formative model).

Specify Population Model

Start off by specifying the **population** model. In this scenario, the individual runs load onto the specific Contrast and ROI combinations. Then, the ROIs are loaded onto the factors `approach` and `avoidance`. The approach and avoidance are specified as *negatively* correlated and the factor variances are fixed to `1`. This is because in the multidimensional circumplex model, when the span of the contrasts are considered, these measures will likely not be orthogonal but instead >90 degree in space.

```
population_model<-'
# By run loadings for bilateral regions
AWin_v_Neut_L_NAcc =~     .7*AWin_v_Neut_L_NAcc_run1    + .7*AWin_v_Neut_L_NAcc_run2
AWin_v_Neut_L_Insula =~   .7*AWin_v_Neut_L_Insula_run1  + .7*AWin_v_Neut_L_Insula_run2
BWin_v_Neut_L_NAcc =~     .7*BWin_v_Neut_L_NAcc_run1    + .7*BWin_v_Neut_L_NAcc_run2
BWin_v_Neut_L_Insula =~   .7*BWin_v_Neut_L_Insula_run1  + .7*BWin_v_Neut_L_Insula_run2
BWin_v_BLose_L_NAcc =~    .7*BWin_v_BLose_L_NAcc_run1   + .7*BWin_v_BLose_L_NAcc_run2
BWin_v_BLose_L_Insula =~  .7*BWin_v_BLose_L_Insula_run1 + .7*BWin_v_BLose_L_Insula_run2
ALose_v_Neut_L_NAcc =~    .7*ALose_v_Neut_L_NAcc_run1   + .7*ALose_v_Neut_L_NAcc_run2
ALose_v_Neut_L_Insula =~  .7*ALose_v_Neut_L_Insula_run1 + .7*ALose_v_Neut_L_Insula_run2
BLose_v_Neut_L_NAcc =~    .7*BLose_v_Neut_L_NAcc_run1   + .7*BLose_v_Neut_L_NAcc_run2
BLose_v_Neut_L_Insula =~  .7*BLose_v_Neut_L_Insula_run1 + .7*BLose_v_Neut_L_Insula_run2
BLose_v_BWin_L_NAcc =~    .7*BLose_v_BWin_L_NAcc_run1   + .7*BLose_v_BWin_L_NAcc_run2
BLose_v_BWin_L_Insula =~  .7*BLose_v_BWin_L_Insula_run1 + .7*BLose_v_BWin_L_Insula_run2

AWin_v_Neut_R_NAcc =~     .7*AWin_v_Neut_R_NAcc_run1    + .7*AWin_v_Neut_R_NAcc_run2
AWin_v_Neut_R_Insula =~   .7*AWin_v_Neut_R_Insula_run1  + .7*AWin_v_Neut_R_Insula_run2
BWin_v_Neut_R_NAcc =~     .7*BWin_v_Neut_R_NAcc_run1    + .7*BWin_v_Neut_R_NAcc_run2
BWin_v_Neut_R_Insula =~   .7*BWin_v_Neut_R_Insula_run1  + .7*BWin_v_Neut_R_Insula_run2
BWin_v_BLose_R_NAcc =~    .7*BWin_v_BLose_R_NAcc_run1   + .7*BWin_v_BLose_R_NAcc_run2
BWin_v_BLose_R_Insula =~  .7*BWin_v_BLose_R_Insula_run1 + .7*BWin_v_BLose_R_Insula_run2
ALose_v_Neut_R_NAcc =~    .7*ALose_v_Neut_R_NAcc_run1   + .7*ALose_v_Neut_R_NAcc_run2
ALose_v_Neut_R_Insula =~  .7*ALose_v_Neut_R_Insula_run1 + .7*ALose_v_Neut_R_Insula_run2
BLose_v_Neut_R_NAcc =~    .7*BLose_v_Neut_R_NAcc_run1   + .7*BLose_v_Neut_R_NAcc_run2
BLose_v_Neut_R_Insula =~  .7*BLose_v_Neut_R_Insula_run1 + .7*BLose_v_Neut_R_Insula_run2
BLose_v_BWin_R_NAcc =~    .7*BLose_v_BWin_R_NAcc_run1   + .7*BLose_v_BWin_R_NAcc_run2
BLose_v_BWin_R_Insula =~  .7*BLose_v_BWin_R_Insula_run1 + .7*BLose_v_BWin_R_Insula_run2

#Factor item loadings
Approach =~  .8*AWin_v_Neut_L_NAcc + .8*AWin_v_Neut_R_NAcc + .45*AWin_v_Neut_R_Insula +
            .7*BWin_v_Neut_L_NAcc +   .7*BWin_v_Neut_R_NAcc + .4*BWin_v_Neut_R_Insula +
```

```
            .8*BWin_v_BLose_L_NAcc +  .8*BWin_v_BLose_R_NAcc

Avoid =~  .8*ALose_v_Neut_L_Insula  + .8*ALose_v_Neut_R_Insula +
          .75*BLose_v_Neut_L_Insula + .75*BLose_v_Neut_R_Insula +
          .8*BLose_v_BWin_L_Insula  + .45*BLose_v_BWin_R_Insula

# Factor Covariances
Approach ~~ -.6*Avoid

# Fixing factor variances
Approach ~~ 1*Approach
Avoid ~~ 1*Avoid

'
```

General samples

Using the population model, simsem is used to create simulated data. This generates a `fake` dataset that is used to pilot the planned Mutligroup CFA, ESEM, EFA and Local SEM models

In this case, 50 repetitions are simulated per model for an *approximate* N sample for each study. Even though the factor variances are specified in the population model as '1', this model fixeds all latent variables using `std.lv = TRUE`.

1. AHRB N = 108
2. MLS N = 159
3. ABCD N = 1000

```
set.seed(25151215)
sim_AHRB <- simsem::sim(nRep = 50, model = "lavaan", n = 108,
           generate = population_model, std.lv = TRUE, lavaanfun = "sem",
           # std.lv ~ ix the variances of all the latent variables
           dataOnly=T, meanstructure = FALSE, seed=123)

sim_MLS <- simsem::sim(nRep = 50, model = "lavaan", n = 159,
                generate = population_model, std.lv = TRUE, lavaanfun = "sem",
                dataOnly=T, meanstructure = FALSE, seed=123)

sim_ABCD <- simsem::sim(nRep = 50, model = "lavaan", n = 1000,
               generate = population_model, std.lv = TRUE, lavaanfun = "sem",
               dataOnly=T, meanstructure = FALSE, seed=123)
```

Average each repetition for each simulated sample. For example, after 50 repetitions of 1000 participants for the population model of ABCD sample, an average estimate is derived using aaply. For each study, the `set` variable is created to differentiate which sample the data is associated with (i.e., grouping variable). In this case, `AHRB = 3`, `MLS = 2`, `ABCD = 1`.

```
sim_AHRB_data <- data.frame(aaply(laply(sim_AHRB, as.matrix), c(2,3), mean))
  sim_AHRB_data$set <-3

sim_MLS_data <- data.frame(aaply(laply(sim_MLS, as.matrix), c(2,3), mean))
  sim_MLS_data$set <-2

sim_ABCD_data <- data.frame(aaply(laply(sim_ABCD, as.matrix), c(2,3), mean))
  sim_ABCD_data$set <-1
```

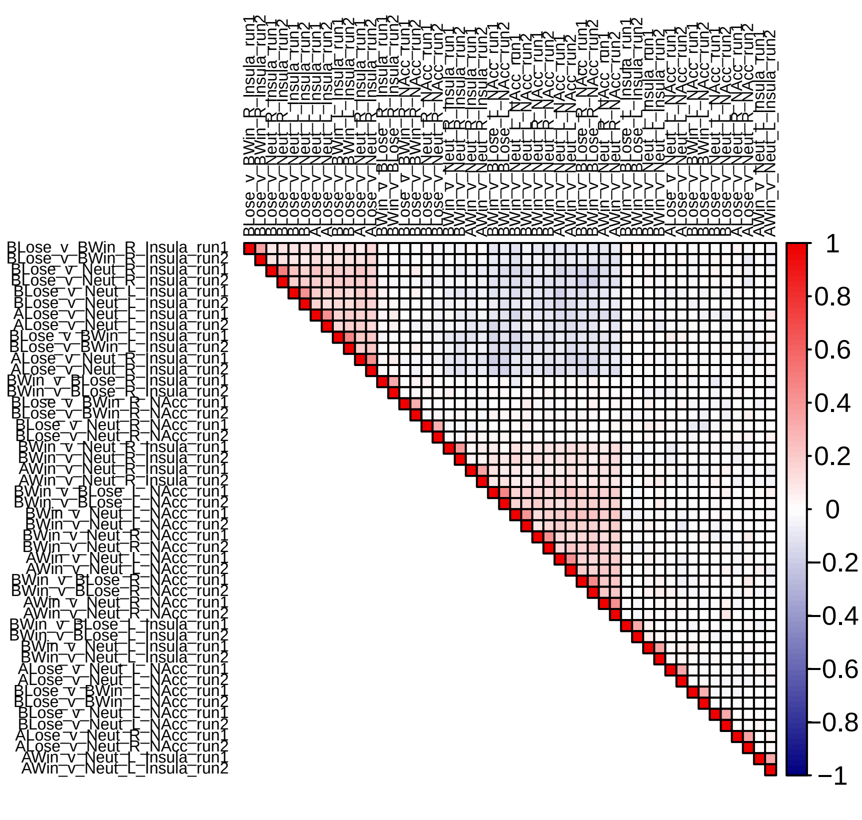
Next, row bind the data sets to form one complete data. This equivalent to stacking the datasets and only retaining a single header row (since all variable names are constant). This creates a 1267 x 49 data matrix (48 brain variables + sample label)

```
brain_set <- rbind(sim_AHRB_data,sim_MLS_data,sim_ABCD_data)
```

Correlation matrix of data

Here, a combination of rcorr and corrplot are used to visualize the associations among the brain variables in the dataset.

```
# Using Hmisc to create a 24x24 matrix for a list (3) that contains: the pearson's r corr, sample size
Brain_corr = rcorr(as.matrix(subset(brain_set,select=-c(set))), # excluding the set of data related to
                   type = "pearson")


# Using corrplot() to create heatmap of the data.
par(mfrow=c(1,1))
corrplot(Brain_corr$r, type = "upper",
         order = 'hclust',
         method =  "color",
         tl.cex = 0.5, tl.col = 'black',
         cl.pos = 'r', tl.pos = 'lt', outline = TRUE,
         col=colorRampPalette(c("navyblue","white","red2"))(100),# colours http://www.stat.columbia.edu
         mar = c(2,.15,.25,.15)#bottom, left, top and right,
         )
```



Running [restricted] Multigroup CFA

Run the CFA multi-group analysis for the three datasets. Multi-group CFA tests the measurement invariance across defined groups to determine whether soft and strict invariance criteria are met and the degree to which the derive estimates for an item in one study can be compared to the same item in another sample. In this case, the focus is on the configural (structure) and metric invariance (loadings). In short, this model evaluates whether factor structure and loadings for the approach and avoidance model are invariant (dont significant differ) across the samples.

Code here is based on measurement invariance models from Maasen et al. 2019, Measurement invariance presentation from Kate Xu and Multi-group CFA tutorial from Hirschfeld & Brachel (2014).

The issue of multi-group is invariance what is discussed in Borsboom (2006). In short, (1) Interpretation of group differences on observed scores DEPENDS on the invariance of measurement models & (2) many make conclusions without doing a single test of measurement invariance.

A tutorial on CFA broadly is available from Lizbeth Benzon and Nilam Ram here and tutorial of measurement invariance (in context of longitudinal data) from Nilam Ram is available here

CFA model

The below specified model will be used. The number of estimate parameters are fewer and may be more appropriate for the theoretical model. This model may result in few convergence issues if the number of participants ends up to be few and the coefficients/estimates are lower.(see Kline 2015 book on **Principles and Practice of Structural Equation Modeling**).

```
MID_model <-'

# Factor loadings
Approach =~ AWin_v_Neut_L_NAcc_run1  + AWin_v_Neut_R_NAcc_run1  + AWin_v_Neut_R_Insula_run1 +
            AWin_v_Neut_L_NAcc_run2  + AWin_v_Neut_R_NAcc_run2 + AWin_v_Neut_R_Insula_run2 +
            BWin_v_Neut_L_NAcc_run1  + BWin_v_Neut_R_NAcc_run1  + BWin_v_Neut_R_Insula_run1 +
            BWin_v_Neut_L_NAcc_run2  + BWin_v_Neut_R_NAcc_run2  + BWin_v_Neut_R_Insula_run2 +
            BWin_v_BLose_L_NAcc_run1 + BWin_v_BLose_R_NAcc_run1 +
            BWin_v_BLose_L_NAcc_run2 + BWin_v_BLose_R_NAcc_run2

Avoid =~    ALose_v_Neut_L_Insula_run1 + ALose_v_Neut_R_Insula_run1 +
            ALose_v_Neut_L_Insula_run2 + ALose_v_Neut_R_Insula_run2 +
            BLose_v_Neut_L_Insula_run1 + BLose_v_Neut_R_Insula_run1 +
            BLose_v_Neut_L_Insula_run2 + BLose_v_Neut_R_Insula_run2 +
            BLose_v_BWin_L_Insula_run1 + BLose_v_BWin_R_Insula_run1 +
            BLose_v_BWin_L_Insula_run2 + BLose_v_BWin_R_Insula_run2
'
```

Running CFA: Three Samples

Below is the CFA model that is used to test the proposed restricted model (see Figure 1 in the manuscript). The CFA fitting procedure is consistent with the description here. For each CFA model, the full sample is filtered for each type sample, e.g. AHRB, MLS, ABCD. The `std.lv= = TRUE` constrain the latent factor variances to *1*. The estimator being used is `MLR`, a maximum likelihood robust estimator. A CFA model is estimated for the complete data (i.e., all three datasets).

```
all_sample <- cfa(model = MID_model, data = brain_set,
                 estimator = "MLR", std.lv = TRUE, meanstructure = TRUE)
```

Fitting Configural CFA

Here, the `configular multigroup model` is fit. As described in D'Urso et al. (2022) measurement invariance pre-print, the configural model tests:

is the structure of the factors is invariannt across the samples ('set'). In other words, if we *a priori* propose a two-factor structure (FA 1 = approch and FA 2 = Avoidance), does this two factor structure represent the between-person variability in the items that reflect the factors across each sample?

If the variability in one sample suggests a one, three, or four factor structure, this will be degrade the fit statistics.

A pre-specified CFA model is used to evaluate whether the measures/items that reflect the factor are the same across groups. `group= 'set'` is used to define the grouping variable. All loadings and intercepts are free to vary across groups, and the factor variance is set to '1' via `std.lv = TRUE`

```
configural_cfa <- cfa(model = MID_model, data = brain_set, group = 'set',
                      estimator = "MLR", std.lv = TRUE, meanstructure = TRUE)
```

Fitting Metric CFA

After fitting the CFA configurial (factor structure) invariance, if the model fit is not poor, then the next step is to test the metric invariance. Metric invariance tests:

are the loadings are consistent across the groups. In other words,are the phenomena (i.e., approach and avoidance) reflected by the same pattern across the measures/items?

One cause for concern may be that the phenomenon are not invariant across age groups, in that the items/measures (ROIs for a given contrast) do not load in the same manner onto each factor. This 'soft' measure of invariance can determine whether the items functions differ across the items and so cannot be easily compared.

The model is fit using the same procedure as for configurial invariance with one exception: In metric invariance the loadings group equality constraint is added to the model via `group.equal=c("loadings")`. The model fit statistics are used to evaluate whether the fit is poor.

```
metric_cfa <-cfa(model = MID_model, data = brain_set,
                 group = 'set', group.equal=c("loadings"),
                 estimator = "MLR", std.lv = TRUE, meanstructure = TRUE)
```

Extracting Fit Statistics

Once the above models are fit, the following information is pulled out and saved into a `out` data frame:

1. Model name
2. Chi-square statistics
3. Model Degrees of Freedom (df)
4. Model p-value
5. RMSEA
6. CFI
7. SRMR
8. AIC
9. BIC

```
# Below selects specific fit data as described in Maassen et al. 2019 OSF. No comparisons are made to c
out <- matrix(NA, ncol = 10, nrow = 4)
colnames(out) <- c("model","chisq","df","pvalue", "rmsea", "cfi","tli", "srmr",
                   "AIC", "BIC")
```

```r
# save fit measures from models
out[1,2:8] <- round(data.matrix(fitmeasures(all_sample,
                                           fit.measures = c("chisq","df","pvalue",
                                                            "rmsea", "cfi","tli", "srmr"))),
                    digits=3)

out[2,2:8] <- round(data.matrix(fitmeasures(configural_cfa,
                                           fit.measures = c("chisq","df","pvalue",
                                                            "rmsea", "cfi","tli", "srmr"))),
                            digits=3)

out[3,2:8] <- round(data.matrix(fitmeasures(metric_cfa,
                                           fit.measures = c("chisq","df","pvalue",
                                                            "rmsea", "cfi","tli", "srmr"))),
                    digits=3)


# AIC models
out[1,9] <- round(AIC(all_sample),3)
out[2,9] <- round(AIC(configural_cfa),3)
out[3,9] <- round(AIC(metric_cfa),3)

# BIC models
out[1,10] <- round(BIC(all_sample),3)
out[2,10] <- round(BIC(configural_cfa),3)
out[3,10] <- round(BIC(metric_cfa),3)

out[1:3,1] <-  c("Overall CFA", "Configg MG-CFA", "Metric MG-CFA")
```

Model Parameter Summary

Reporting standardized coefficients.

All Sample CFA model

```r
##### Summarizing All Samples CFA model #####
parameters(all_sample, standardize = T)
```

```
## # Loading
##
## Link                                 | Coefficient |   SE |       95% CI |     z |      p
## ----------------------------------------------------------------------------------------
## Approach =~ AWin_v_Neut_L_NAcc_run1   |        0.44 | 0.03 | [0.38, 0.51] | 14.45 | < .001
## Approach =~ AWin_v_Neut_R_NAcc_run1   |        0.45 | 0.03 | [0.40, 0.51] | 15.78 | < .001
## Approach =~ AWin_v_Neut_R_Insula_run1 |        0.22 | 0.03 | [0.16, 0.29] |  7.08 | < .001
## Approach =~ AWin_v_Neut_L_NAcc_run2   |        0.48 | 0.03 | [0.43, 0.54] | 16.93 | < .001
## Approach =~ AWin_v_Neut_R_NAcc_run2   |        0.48 | 0.03 | [0.43, 0.53] | 17.82 | < .001
## Approach =~ AWin_v_Neut_R_Insula_run2 |        0.24 | 0.03 | [0.17, 0.30] |  7.47 | < .001
## Approach =~ BWin_v_Neut_L_NAcc_run1   |        0.48 | 0.03 | [0.42, 0.53] | 17.17 | < .001
## Approach =~ BWin_v_Neut_R_NAcc_run1   |        0.37 | 0.03 | [0.31, 0.43] | 12.53 | < .001
## Approach =~ BWin_v_Neut_R_Insula_run1 |        0.23 | 0.03 | [0.17, 0.29] |  7.32 | < .001
## Approach =~ BWin_v_Neut_L_NAcc_run2   |        0.40 | 0.03 | [0.35, 0.46] | 13.99 | < .001
## Approach =~ BWin_v_Neut_R_NAcc_run2   |        0.41 | 0.03 | [0.36, 0.47] | 14.73 | < .001
```

```
## Approach =~ BWin_v_Neut_R_Insula_run2 |        0.27 | 0.03 | [0.21, 0.33] |  8.30 | < .001
## Approach =~ BWin_v_BLose_L_NAcc_run1  |        0.45 | 0.03 | [0.39, 0.50] | 15.12 | < .001
## Approach =~ BWin_v_BLose_R_NAcc_run1  |        0.47 | 0.03 | [0.41, 0.53] | 16.47 | < .001
## Approach =~ BWin_v_BLose_L_NAcc_run2  |        0.42 | 0.03 | [0.37, 0.48] | 14.19 | < .001
## Approach =~ BWin_v_BLose_R_NAcc_run2  |        0.47 | 0.03 | [0.42, 0.53] | 16.34 | < .001
## Avoid =~ ALose_v_Neut_L_Insula_run1   |        0.46 | 0.03 | [0.40, 0.52] | 15.36 | < .001
## Avoid =~ ALose_v_Neut_R_Insula_run1   |        0.48 | 0.03 | [0.42, 0.53] | 16.58 | < .001
## Avoid =~ ALose_v_Neut_L_Insula_run2   |        0.42 | 0.03 | [0.36, 0.48] | 14.25 | < .001
## Avoid =~ ALose_v_Neut_R_Insula_run2   |        0.47 | 0.03 | [0.41, 0.52] | 16.74 | < .001
## Avoid =~ BLose_v_Neut_L_Insula_run1   |        0.44 | 0.03 | [0.38, 0.50] | 14.51 | < .001
## Avoid =~ BLose_v_Neut_R_Insula_run1   |        0.50 | 0.03 | [0.44, 0.56] | 16.42 | < .001
## Avoid =~ BLose_v_Neut_L_Insula_run2   |        0.41 | 0.03 | [0.35, 0.47] | 13.36 | < .001
## Avoid =~ BLose_v_Neut_R_Insula_run2   |        0.48 | 0.03 | [0.43, 0.54] | 16.06 | < .001
## Avoid =~ BLose_v_BWin_L_Insula_run1   |        0.45 | 0.03 | [0.40, 0.51] | 14.94 | < .001
## Avoid =~ BLose_v_BWin_R_Insula_run1   |        0.27 | 0.03 | [0.21, 0.34] |  8.55 | < .001
## Avoid =~ BLose_v_BWin_L_Insula_run2   |        0.45 | 0.03 | [0.39, 0.51] | 14.91 | < .001
## Avoid =~ BLose_v_BWin_R_Insula_run2   |        0.27 | 0.03 | [0.21, 0.34] |  8.53 | < .001
##
## # Correlation
##
## Link              | Coefficient |   SE |         95% CI |      z |      p
## ------------------------------------------------------------------------
## Approach ~~ Avoid |       -0.55 | 0.03 | [-0.61, -0.49] | -18.77 | < .001
```

Configural CFA model

```
##### Summarizing Configural MG-CFA model #####
parameters(configural_cfa, standardize = T)
```

```
## # Loading
##
## Link                                     | Coefficient |   SE |         95% CI |     z |       p | Group
## -------------------------------------------------------------------------------------------------------
## Approach =~ AWin_v_Neut_L_NAcc_run1      |        0.51 | 0.13 | [ 0.26, 0.76]  |  4.03 | < .001 |  1.00
## Approach =~ AWin_v_Neut_R_NAcc_run1      |        0.51 | 0.10 | [ 0.31, 0.71]  |  5.07 | < .001 |  1.00
## Approach =~ AWin_v_Neut_R_Insula_run1    |        0.19 | 0.13 | [-0.07, 0.45]  |  1.43 | 0.154  |  1.00
## Approach =~ AWin_v_Neut_L_NAcc_run2      |        0.47 | 0.13 | [ 0.21, 0.72]  |  3.63 | < .001 |  1.00
## Approach =~ AWin_v_Neut_R_NAcc_run2      |        0.38 | 0.12 | [ 0.15, 0.61]  |  3.23 | 0.001  |  1.00
## Approach =~ AWin_v_Neut_R_Insula_run2    |        0.18 | 0.14 | [-0.09, 0.45]  |  1.29 | 0.198  |  1.00
## Approach =~ BWin_v_Neut_L_NAcc_run1      |        0.33 | 0.11 | [ 0.12, 0.54]  |  3.12 | 0.002  |  1.00
## Approach =~ BWin_v_Neut_R_NAcc_run1      |        0.30 | 0.15 | [ 0.01, 0.59]  |  2.00 | 0.046  |  1.00
## Approach =~ BWin_v_Neut_R_Insula_run1    |        0.10 | 0.14 | [-0.16, 0.37]  |  0.76 | 0.445  |  1.00
## Approach =~ BWin_v_Neut_L_NAcc_run2      |        0.34 | 0.10 | [ 0.14, 0.55]  |  3.28 | 0.001  |  1.00
## Approach =~ BWin_v_Neut_R_NAcc_run2      |        0.45 | 0.11 | [ 0.25, 0.66]  |  4.30 | < .001 |  1.00
## Approach =~ BWin_v_Neut_R_Insula_run2    |        0.26 | 0.11 | [ 0.04, 0.48]  |  2.27 | 0.023  |  1.00
## Approach =~ BWin_v_BLose_L_NAcc_run1     |        0.34 | 0.14 | [ 0.07, 0.61]  |  2.45 | 0.014  |  1.00
## Approach =~ BWin_v_BLose_R_NAcc_run1     |        0.48 | 0.10 | [ 0.29, 0.67]  |  4.94 | < .001 |  1.00
## Approach =~ BWin_v_BLose_L_NAcc_run2     |        0.44 | 0.15 | [ 0.14, 0.73]  |  2.88 | 0.004  |  1.00
## Approach =~ BWin_v_BLose_R_NAcc_run2     |        0.56 | 0.09 | [ 0.38, 0.75]  |  6.06 | < .001 |  1.00
## Avoid =~ ALose_v_Neut_L_Insula_run1      |        0.52 | 0.10 | [ 0.32, 0.72]  |  5.16 | < .001 |  1.00
## Avoid =~ ALose_v_Neut_R_Insula_run1      |        0.53 | 0.09 | [ 0.34, 0.71]  |  5.62 | < .001 |  1.00
## Avoid =~ ALose_v_Neut_L_Insula_run2      |        0.55 | 0.11 | [ 0.33, 0.77]  |  4.86 | < .001 |  1.00
## Avoid =~ ALose_v_Neut_R_Insula_run2      |        0.54 | 0.10 | [ 0.34, 0.73]  |  5.33 | < .001 |  1.00
## Avoid =~ BLose_v_Neut_L_Insula_run1      |        0.50 | 0.10 | [ 0.30, 0.69]  |  4.95 | < .001 |  1.00
```

```
## Avoid =~ BLose_v_Neut_R_Insula_run1   |        0.41 | 0.10 | [ 0.21, 0.61] |  3.99 | < .001 |  1.00
## Avoid =~ BLose_v_Neut_L_Insula_run2   |        0.23 | 0.09 | [ 0.05, 0.41] |  2.54 | 0.011  |  1.00
## Avoid =~ BLose_v_Neut_R_Insula_run2   |        0.46 | 0.11 | [ 0.24, 0.68] |  4.11 | < .001 |  1.00
## Avoid =~ BLose_v_BWin_L_Insula_run1   |        0.34 | 0.10 | [ 0.15, 0.53] |  3.52 | < .001 |  1.00
## Avoid =~ BLose_v_BWin_R_Insula_run1   |        0.37 | 0.09 | [ 0.19, 0.56] |  3.98 | < .001 |  1.00
## Avoid =~ BLose_v_BWin_L_Insula_run2   |        0.41 | 0.12 | [ 0.17, 0.64] |  3.38 | < .001 |  1.00
## Avoid =~ BLose_v_BWin_R_Insula_run2   |        0.37 | 0.10 | [ 0.17, 0.57] |  3.61 | < .001 |  1.00
## Approach =~ AWin_v_Neut_L_NAcc_run1   |        0.42 | 0.10 | [ 0.22, 0.61] |  4.23 | < .001 |  2.00
## Approach =~ AWin_v_Neut_R_NAcc_run1   |        0.54 | 0.07 | [ 0.40, 0.69] |  7.43 | < .001 |  2.00
## Approach =~ AWin_v_Neut_R_Insula_run1 |        0.15 | 0.08 | [-0.01, 0.31] |  1.89 | 0.059  |  2.00
## Approach =~ AWin_v_Neut_L_NAcc_run2   |        0.51 | 0.08 | [ 0.36, 0.67] |  6.55 | < .001 |  2.00
## Approach =~ AWin_v_Neut_R_NAcc_run2   |        0.41 | 0.09 | [ 0.23, 0.59] |  4.52 | < .001 |  2.00
## Approach =~ AWin_v_Neut_R_Insula_run2 |    2.85e-03 | 0.09 | [-0.18, 0.18] |  0.03 | 0.975  |  2.00
## Approach =~ BWin_v_Neut_L_NAcc_run1   |        0.40 | 0.11 | [ 0.19, 0.61] |  3.73 | < .001 |  2.00
## Approach =~ BWin_v_Neut_R_NAcc_run1   |        0.32 | 0.10 | [ 0.11, 0.52] |  3.04 | 0.002  |  2.00
## Approach =~ BWin_v_Neut_R_Insula_run1 |        0.23 | 0.10 | [ 0.03, 0.44] |  2.23 | 0.026  |  2.00
## Approach =~ BWin_v_Neut_L_NAcc_run2   |        0.31 | 0.10 | [ 0.12, 0.49] |  3.23 | 0.001  |  2.00
## Approach =~ BWin_v_Neut_R_NAcc_run2   |        0.33 | 0.09 | [ 0.16, 0.50] |  3.80 | < .001 |  2.00
## Approach =~ BWin_v_Neut_R_Insula_run2 |        0.27 | 0.10 | [ 0.08, 0.47] |  2.74 | 0.006  |  2.00
## Approach =~ BWin_v_BLose_L_NAcc_run1  |        0.37 | 0.10 | [ 0.17, 0.56] |  3.71 | < .001 |  2.00
## Approach =~ BWin_v_BLose_R_NAcc_run1  |        0.51 | 0.08 | [ 0.35, 0.67] |  6.29 | < .001 |  2.00
## Approach =~ BWin_v_BLose_L_NAcc_run2  |        0.44 | 0.09 | [ 0.27, 0.61] |  5.10 | < .001 |  2.00
## Approach =~ BWin_v_BLose_R_NAcc_run2  |        0.36 | 0.09 | [ 0.19, 0.54] |  3.99 | < .001 |  2.00
## Avoid =~ ALose_v_Neut_L_Insula_run1   |        0.51 | 0.08 | [ 0.34, 0.67] |  6.14 | < .001 |  2.00
## Avoid =~ ALose_v_Neut_R_Insula_run1   |        0.37 | 0.09 | [ 0.20, 0.53] |  4.24 | < .001 |  2.00
## Avoid =~ ALose_v_Neut_L_Insula_run2   |        0.39 | 0.09 | [ 0.22, 0.56] |  4.38 | < .001 |  2.00
## Avoid =~ ALose_v_Neut_R_Insula_run2   |        0.51 | 0.07 | [ 0.38, 0.64] |  7.65 | < .001 |  2.00
## Avoid =~ BLose_v_Neut_L_Insula_run1   |        0.35 | 0.08 | [ 0.19, 0.51] |  4.24 | < .001 |  2.00
## Avoid =~ BLose_v_Neut_R_Insula_run1   |        0.50 | 0.07 | [ 0.35, 0.64] |  6.64 | < .001 |  2.00
## Avoid =~ BLose_v_Neut_L_Insula_run2   |        0.44 | 0.07 | [ 0.30, 0.58] |  6.24 | < .001 |  2.00
## Avoid =~ BLose_v_Neut_R_Insula_run2   |        0.47 | 0.09 | [ 0.31, 0.64] |  5.55 | < .001 |  2.00
## Avoid =~ BLose_v_BWin_L_Insula_run1   |        0.61 | 0.08 | [ 0.46, 0.76] |  7.97 | < .001 |  2.00
## Avoid =~ BLose_v_BWin_R_Insula_run1   |        0.48 | 0.07 | [ 0.33, 0.62] |  6.40 | < .001 |  2.00
## Avoid =~ BLose_v_BWin_L_Insula_run2   |        0.63 | 0.06 | [ 0.50, 0.75] |  9.85 | < .001 |  2.00
## Avoid =~ BLose_v_BWin_R_Insula_run2   |        0.44 | 0.08 | [ 0.29, 0.59] |  5.74 | < .001 |  2.00
## Approach =~ AWin_v_Neut_L_NAcc_run1   |        0.45 | 0.03 | [ 0.38, 0.51] | 13.11 | < .001 |  3.00
## Approach =~ AWin_v_Neut_R_NAcc_run1   |        0.43 | 0.03 | [ 0.37, 0.50] | 13.35 | < .001 |  3.00
## Approach =~ AWin_v_Neut_R_Insula_run1 |        0.24 | 0.04 | [ 0.17, 0.31] |  6.69 | < .001 |  3.00
## Approach =~ AWin_v_Neut_L_NAcc_run2   |        0.48 | 0.03 | [ 0.42, 0.55] | 14.87 | < .001 |  3.00
## Approach =~ AWin_v_Neut_R_NAcc_run2   |        0.49 | 0.03 | [ 0.44, 0.55] | 17.00 | < .001 |  3.00
## Approach =~ AWin_v_Neut_R_Insula_run2 |        0.27 | 0.03 | [ 0.20, 0.34] |  7.91 | < .001 |  3.00
## Approach =~ BWin_v_Neut_L_NAcc_run1   |        0.51 | 0.03 | [ 0.45, 0.56] | 16.98 | < .001 |  3.00
## Approach =~ BWin_v_Neut_R_NAcc_run1   |        0.39 | 0.03 | [ 0.32, 0.45] | 12.10 | < .001 |  3.00
## Approach =~ BWin_v_Neut_R_Insula_run1 |        0.24 | 0.03 | [ 0.17, 0.31] |  6.99 | < .001 |  3.00
## Approach =~ BWin_v_Neut_L_NAcc_run2   |        0.42 | 0.03 | [ 0.36, 0.48] | 13.20 | < .001 |  3.00
## Approach =~ BWin_v_Neut_R_NAcc_run2   |        0.42 | 0.03 | [ 0.36, 0.48] | 13.55 | < .001 |  3.00
## Approach =~ BWin_v_Neut_R_Insula_run2 |        0.27 | 0.04 | [ 0.20, 0.35] |  7.45 | < .001 |  3.00
## Approach =~ BWin_v_BLose_L_NAcc_run1  |        0.47 | 0.03 | [ 0.41, 0.53] | 15.00 | < .001 |  3.00
## Approach =~ BWin_v_BLose_R_NAcc_run1  |        0.46 | 0.03 | [ 0.40, 0.53] | 14.42 | < .001 |  3.00
## Approach =~ BWin_v_BLose_L_NAcc_run2  |        0.42 | 0.03 | [ 0.36, 0.48] | 12.84 | < .001 |  3.00
## Approach =~ BWin_v_BLose_R_NAcc_run2  |        0.48 | 0.03 | [ 0.42, 0.55] | 14.99 | < .001 |  3.00
## Avoid =~ ALose_v_Neut_L_Insula_run1   |        0.44 | 0.03 | [ 0.38, 0.51] | 12.82 | < .001 |  3.00
## Avoid =~ ALose_v_Neut_R_Insula_run1   |        0.50 | 0.03 | [ 0.43, 0.56] | 15.48 | < .001 |  3.00
## Avoid =~ ALose_v_Neut_L_Insula_run2   |        0.41 | 0.03 | [ 0.34, 0.47] | 12.44 | < .001 |  3.00
```

```
## Avoid =~ ALose_v_Neut_R_Insula_run2   |        0.46 | 0.03 | [ 0.39, 0.52] | 14.14 | < .001 |  3.00
## Avoid =~ BLose_v_Neut_L_Insula_run1   |        0.44 | 0.03 | [ 0.38, 0.51] | 12.90 | < .001 |  3.00
## Avoid =~ BLose_v_Neut_R_Insula_run1   |        0.51 | 0.03 | [ 0.44, 0.57] | 14.46 | < .001 |  3.00
## Avoid =~ BLose_v_Neut_L_Insula_run2   |        0.43 | 0.04 | [ 0.36, 0.50] | 11.76 | < .001 |  3.00
## Avoid =~ BLose_v_Neut_R_Insula_run2   |        0.48 | 0.03 | [ 0.42, 0.55] | 14.05 | < .001 |  3.00
## Avoid =~ BLose_v_BWin_L_Insula_run1   |        0.44 | 0.03 | [ 0.37, 0.51] | 12.62 | < .001 |  3.00
## Avoid =~ BLose_v_BWin_R_Insula_run1   |        0.23 | 0.04 | [ 0.16, 0.30] |  6.18 | < .001 |  3.00
## Avoid =~ BLose_v_BWin_L_Insula_run2   |        0.42 | 0.04 | [ 0.35, 0.49] | 11.89 | < .001 |  3.00
## Avoid =~ BLose_v_BWin_R_Insula_run2   |        0.23 | 0.04 | [ 0.16, 0.31] |  6.25 | < .001 |  3.00
##
## # Correlation
##
## Link               | Coefficient |   SE |         95% CI |      z |      p | Group
## ---------------------------------------------------------------------------------
## Approach ~~ Avoid |       -0.42 | 0.12 | [-0.65, -0.19] |  -3.63 | < .001 |  1.00
## Approach ~~ Avoid |       -0.54 | 0.09 | [-0.71, -0.36] |  -6.01 | < .001 |  2.00
## Approach ~~ Avoid |       -0.57 | 0.03 | [-0.63, -0.51] | -17.66 | < .001 |  3.00
```

Metric CFA model

```r
##### Summarizing Metric Multi-group CFA model #####
parameters(metric_cfa, standardize = T)
```

```
## # Loading
##
## Link                                          | Coefficient |   SE |       95% CI |     z |      p |
## -----------------------------------------------------------------------------------------------------
## Approach =~ AWin_v_Neut_L_NAcc_run1 (.p1.)    |        0.43 | 0.05 | [0.33, 0.53] |  8.39 | < .001 |
## Approach =~ AWin_v_Neut_R_NAcc_run1 (.p2.)    |        0.43 | 0.06 | [0.32, 0.54] |  7.56 | < .001 |
## Approach =~ AWin_v_Neut_R_Insula_run1 (.p3.)  |        0.20 | 0.04 | [0.13, 0.27] |  5.63 | < .001 |
## Approach =~ AWin_v_Neut_L_NAcc_run2 (.p4.)    |        0.46 | 0.05 | [0.36, 0.56] |  9.09 | < .001 |
## Approach =~ AWin_v_Neut_R_NAcc_run2 (.p5.)    |        0.46 | 0.05 | [0.36, 0.55] |  9.65 | < .001 |
## Approach =~ AWin_v_Neut_R_Insula_run2 (.p6.)  |        0.23 | 0.04 | [0.14, 0.31] |  5.41 | < .001 |
## Approach =~ BWin_v_Neut_L_NAcc_run1 (.p7.)    |        0.43 | 0.05 | [0.34, 0.53] |  8.94 | < .001 |
## Approach =~ BWin_v_Neut_R_NAcc_run1 (.p8.)    |        0.36 | 0.05 | [0.26, 0.45] |  7.54 | < .001 |
## Approach =~ BWin_v_Neut_R_Insula_run1 (.p9.)  |        0.22 | 0.04 | [0.15, 0.30] |  5.99 | < .001 |
## Approach =~ BWin_v_Neut_L_NAcc_run2 (.p10.)   |        0.39 | 0.05 | [0.30, 0.48] |  8.24 | < .001 |
## Approach =~ BWin_v_Neut_R_NAcc_run2 (.p11.)   |        0.38 | 0.05 | [0.28, 0.47] |  7.77 | < .001 |
## Approach =~ BWin_v_Neut_R_Insula_run2 (.p12.) |        0.29 | 0.04 | [0.20, 0.38] |  6.53 | < .001 |
## Approach =~ BWin_v_BLose_L_NAcc_run1 (.p13.)  |        0.40 | 0.05 | [0.31, 0.49] |  8.48 | < .001 |
## Approach =~ BWin_v_BLose_R_NAcc_run1 (.p14.)  |        0.45 | 0.05 | [0.36, 0.55] |  9.18 | < .001 |
## Approach =~ BWin_v_BLose_L_NAcc_run2 (.p15.)  |        0.36 | 0.05 | [0.27, 0.45] |  7.80 | < .001 |
## Approach =~ BWin_v_BLose_R_NAcc_run2 (.p16.)  |        0.50 | 0.06 | [0.38, 0.61] |  8.40 | < .001 |
## Avoid =~ ALose_v_Neut_L_Insula_run1 (.p17.)   |        0.47 | 0.05 | [0.37, 0.57] |  9.59 | < .001 |
## Avoid =~ ALose_v_Neut_R_Insula_run1 (.p18.)   |        0.48 | 0.05 | [0.38, 0.58] |  9.33 | < .001 |
## Avoid =~ ALose_v_Neut_L_Insula_run2 (.p19.)   |        0.41 | 0.05 | [0.32, 0.51] |  8.48 | < .001 |
## Avoid =~ ALose_v_Neut_R_Insula_run2 (.p20.)   |        0.44 | 0.05 | [0.34, 0.54] |  8.57 | < .001 |
## Avoid =~ BLose_v_Neut_L_Insula_run1 (.p21.)   |        0.46 | 0.05 | [0.36, 0.55] |  9.63 | < .001 |
## Avoid =~ BLose_v_Neut_R_Insula_run1 (.p22.)   |        0.48 | 0.04 | [0.39, 0.57] | 10.63 | < .001 |
## Avoid =~ BLose_v_Neut_L_Insula_run2 (.p23.)   |        0.40 | 0.04 | [0.32, 0.49] |  9.31 | < .001 |
## Avoid =~ BLose_v_Neut_R_Insula_run2 (.p24.)   |        0.51 | 0.05 | [0.41, 0.61] | 10.19 | < .001 |
## Avoid =~ BLose_v_BWin_L_Insula_run1 (.p25.)   |        0.48 | 0.04 | [0.39, 0.56] | 11.05 | < .001 |
## Avoid =~ BLose_v_BWin_R_Insula_run1 (.p26.)   |        0.30 | 0.04 | [0.21, 0.39] |  6.78 | < .001 |
## Avoid =~ BLose_v_BWin_L_Insula_run2 (.p27.)   |        0.47 | 0.05 | [0.37, 0.56] |  9.97 | < .001 |
```

```
## Avoid =~ BLose_v_BWin_R_Insula_run2 (.p28.)   |         0.28 | 0.04 | [0.19, 0.36] |  6.44 | < .001 |
## Approach =~ AWin_v_Neut_L_NAcc_run1 (.p1.)    |         0.42 | 0.04 | [0.33, 0.50] |  9.65 | < .001 |
## Approach =~ AWin_v_Neut_R_NAcc_run1 (.p2.)    |         0.44 | 0.05 | [0.34, 0.53] |  9.03 | < .001 |
## Approach =~ AWin_v_Neut_R_Insula_run1 (.p3.)  |         0.22 | 0.03 | [0.15, 0.29] |  6.46 | < .001 |
## Approach =~ AWin_v_Neut_L_NAcc_run2 (.p4.)    |         0.44 | 0.04 | [0.35, 0.52] | 10.49 | < .001 |
## Approach =~ AWin_v_Neut_R_NAcc_run2 (.p5.)    |         0.44 | 0.04 | [0.37, 0.52] | 10.97 | < .001 |
## Approach =~ AWin_v_Neut_R_Insula_run2 (.p6.)  |         0.22 | 0.03 | [0.16, 0.28] |  7.38 | < .001 |
## Approach =~ BWin_v_Neut_L_NAcc_run1 (.p7.)    |         0.42 | 0.04 | [0.34, 0.51] | 10.00 | < .001 |
## Approach =~ BWin_v_Neut_R_NAcc_run1 (.p8.)    |         0.32 | 0.04 | [0.25, 0.40] |  8.44 | < .001 |
## Approach =~ BWin_v_Neut_R_Insula_run1 (.p9.)  |         0.20 | 0.03 | [0.14, 0.27] |  6.20 | < .001 |
## Approach =~ BWin_v_Neut_L_NAcc_run2 (.p10.)   |         0.38 | 0.04 | [0.30, 0.45] |  9.46 | < .001 |
## Approach =~ BWin_v_Neut_R_NAcc_run2 (.p11.)   |         0.36 | 0.04 | [0.28, 0.44] |  9.06 | < .001 |
## Approach =~ BWin_v_Neut_R_Insula_run2 (.p12.) |         0.24 | 0.04 | [0.17, 0.32] |  6.51 | < .001 |
## Approach =~ BWin_v_BLose_L_NAcc_run1 (.p13.)  |         0.41 | 0.04 | [0.33, 0.48] | 10.32 | < .001 |
## Approach =~ BWin_v_BLose_R_NAcc_run1 (.p14.)  |         0.45 | 0.04 | [0.36, 0.54] | 10.14 | < .001 |
## Approach =~ BWin_v_BLose_L_NAcc_run2 (.p15.)  |         0.37 | 0.04 | [0.29, 0.45] |  9.01 | < .001 |
## Approach =~ BWin_v_BLose_R_NAcc_run2 (.p16.)  |         0.41 | 0.04 | [0.33, 0.49] | 10.34 | < .001 |
## Avoid =~ ALose_v_Neut_L_Insula_run1 (.p17.)   |         0.48 | 0.05 | [0.39, 0.57] | 10.43 | < .001 |
## Avoid =~ ALose_v_Neut_R_Insula_run1 (.p18.)   |         0.50 | 0.04 | [0.43, 0.58] | 13.51 | < .001 |
## Avoid =~ ALose_v_Neut_L_Insula_run2 (.p19.)   |         0.44 | 0.04 | [0.36, 0.53] | 10.74 | < .001 |
## Avoid =~ ALose_v_Neut_R_Insula_run2 (.p20.)   |         0.53 | 0.04 | [0.45, 0.61] | 12.91 | < .001 |
## Avoid =~ BLose_v_Neut_L_Insula_run1 (.p21.)   |         0.45 | 0.04 | [0.37, 0.53] | 11.26 | < .001 |
## Avoid =~ BLose_v_Neut_R_Insula_run1 (.p22.)   |         0.56 | 0.04 | [0.48, 0.64] | 13.66 | < .001 |
## Avoid =~ BLose_v_Neut_L_Insula_run2 (.p23.)   |         0.46 | 0.04 | [0.38, 0.55] | 10.64 | < .001 |
## Avoid =~ BLose_v_Neut_R_Insula_run2 (.p24.)   |         0.52 | 0.04 | [0.43, 0.60] | 12.04 | < .001 |
## Avoid =~ BLose_v_BWin_L_Insula_run1 (.p25.)   |         0.51 | 0.05 | [0.42, 0.60] | 10.92 | < .001 |
## Avoid =~ BLose_v_BWin_R_Insula_run1 (.p26.)   |         0.33 | 0.05 | [0.23, 0.43] |  6.59 | < .001 |
## Avoid =~ BLose_v_BWin_L_Insula_run2 (.p27.)   |         0.50 | 0.05 | [0.40, 0.59] | 10.01 | < .001 |
## Avoid =~ BLose_v_BWin_R_Insula_run2 (.p28.)   |         0.28 | 0.04 | [0.20, 0.36] |  6.93 | < .001 |
## Approach =~ AWin_v_Neut_L_NAcc_run1 (.p1.)    |         0.45 | 0.03 | [0.39, 0.51] | 14.42 | < .001 |
## Approach =~ AWin_v_Neut_R_NAcc_run1 (.p2.)    |         0.46 | 0.03 | [0.40, 0.51] | 15.97 | < .001 |
## Approach =~ AWin_v_Neut_R_Insula_run1 (.p3.)  |         0.23 | 0.03 | [0.16, 0.29] |  7.03 | < .001 |
## Approach =~ AWin_v_Neut_L_NAcc_run2 (.p4.)    |         0.49 | 0.03 | [0.43, 0.55] | 16.52 | < .001 |
## Approach =~ AWin_v_Neut_R_NAcc_run2 (.p5.)    |         0.48 | 0.03 | [0.43, 0.54] | 17.30 | < .001 |
## Approach =~ AWin_v_Neut_R_Insula_run2 (.p6.)  |         0.24 | 0.03 | [0.17, 0.30] |  7.16 | < .001 |
## Approach =~ BWin_v_Neut_L_NAcc_run1 (.p7.)    |         0.50 | 0.03 | [0.44, 0.55] | 17.20 | < .001 |
## Approach =~ BWin_v_Neut_R_NAcc_run1 (.p8.)    |         0.38 | 0.03 | [0.32, 0.44] | 12.64 | < .001 |
## Approach =~ BWin_v_Neut_R_Insula_run1 (.p9.)  |         0.24 | 0.03 | [0.17, 0.30] |  7.28 | < .001 |
## Approach =~ BWin_v_Neut_L_NAcc_run2 (.p10.)   |         0.41 | 0.03 | [0.35, 0.47] | 13.61 | < .001 |
## Approach =~ BWin_v_Neut_R_NAcc_run2 (.p11.)   |         0.42 | 0.03 | [0.37, 0.48] | 14.80 | < .001 |
## Approach =~ BWin_v_Neut_R_Insula_run2 (.p12.) |         0.27 | 0.03 | [0.21, 0.34] |  8.27 | < .001 |
## Approach =~ BWin_v_BLose_L_NAcc_run1 (.p13.)  |         0.46 | 0.03 | [0.40, 0.52] | 15.10 | < .001 |
## Approach =~ BWin_v_BLose_R_NAcc_run1 (.p14.)  |         0.47 | 0.03 | [0.42, 0.53] | 16.39 | < .001 |
## Approach =~ BWin_v_BLose_L_NAcc_run2 (.p15.)  |         0.43 | 0.03 | [0.37, 0.49] | 14.26 | < .001 |
## Approach =~ BWin_v_BLose_R_NAcc_run2 (.p16.)  |         0.49 | 0.03 | [0.43, 0.54] | 16.43 | < .001 |
## Avoid =~ ALose_v_Neut_L_Insula_run1 (.p17.)   |         0.46 | 0.03 | [0.40, 0.52] | 15.12 | < .001 |
## Avoid =~ ALose_v_Neut_R_Insula_run1 (.p18.)   |         0.47 | 0.03 | [0.41, 0.53] | 15.57 | < .001 |
## Avoid =~ ALose_v_Neut_L_Insula_run2 (.p19.)   |         0.42 | 0.03 | [0.36, 0.48] | 14.03 | < .001 |
## Avoid =~ ALose_v_Neut_R_Insula_run2 (.p20.)   |         0.46 | 0.03 | [0.40, 0.51] | 16.08 | < .001 |
## Avoid =~ BLose_v_Neut_L_Insula_run1 (.p21.)   |         0.43 | 0.03 | [0.37, 0.49] | 13.74 | < .001 |
## Avoid =~ BLose_v_Neut_R_Insula_run1 (.p22.)   |         0.49 | 0.03 | [0.42, 0.55] | 15.06 | < .001 |
## Avoid =~ BLose_v_Neut_L_Insula_run2 (.p23.)   |         0.40 | 0.03 | [0.34, 0.47] | 12.48 | < .001 |
## Avoid =~ BLose_v_Neut_R_Insula_run2 (.p24.)   |         0.47 | 0.03 | [0.41, 0.53] | 15.11 | < .001 |
## Avoid =~ BLose_v_BWin_L_Insula_run1 (.p25.)   |         0.44 | 0.03 | [0.38, 0.51] | 14.04 | < .001 |
```

```
## Avoid =~ BLose_v_BWin_R_Insula_run1 (.p26.)   |        0.27 | 0.03 | [0.21, 0.33] |  8.91 | < .001 |
## Avoid =~ BLose_v_BWin_L_Insula_run2 (.p27.)   |        0.44 | 0.03 | [0.38, 0.50] | 14.73 | < .001 |
## Avoid =~ BLose_v_BWin_R_Insula_run2 (.p28.)   |        0.27 | 0.03 | [0.21, 0.33] |  8.55 | < .001 |
##
## # Correlation
##
## Link              | Coefficient |   SE |         95% CI |      z |      p | Group
## ---------------------------------------------------------------------------------
## Approach ~~ Avoid |       -0.40 | 0.11 | [-0.62, -0.18] |  -3.61 | < .001 |  1.00
## Approach ~~ Avoid |       -0.53 | 0.09 | [-0.71, -0.36] |  -6.03 | < .001 |  2.00
## Approach ~~ Avoid |       -0.57 | 0.03 | [-0.64, -0.51] | -17.75 | < .001 |  3.00
```

Comparing models w/ BIC/AIC (anova)

The below compares whether the complete data (across all three samples) in the `all_cfa` model is significantly improved by the configural invariance model. A significant value indicates that the configural model is significantly better than the full sample cfa.

```
anova(all_sample, configural_cfa)
```

```
## Scaled Chi-Squared Difference Test (method = "satorra.bentler.2001")
##
## lavaan NOTE:
##     The "Chisq" column contains standard test statistics, not the
##     robust test that should be reported per model. A robust difference
##     test is a function of two standard (not robust) statistics.
##
##                  Df    AIC    BIC  Chisq Chisq diff Df diff Pr(>Chisq)
## all_sample      349 -21672 -21235 2200.6
## configural_cfa 1047 -21482 -20170 2961.4     777.17     698    0.01964 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Next, anova is used to compare the model improve in AIC/BIC by between the configural and metric invariance. A significantly result in the anova would indicate a significant improvement of the metric model over the configural model.

```
anova(configural_cfa, metric_cfa)
```

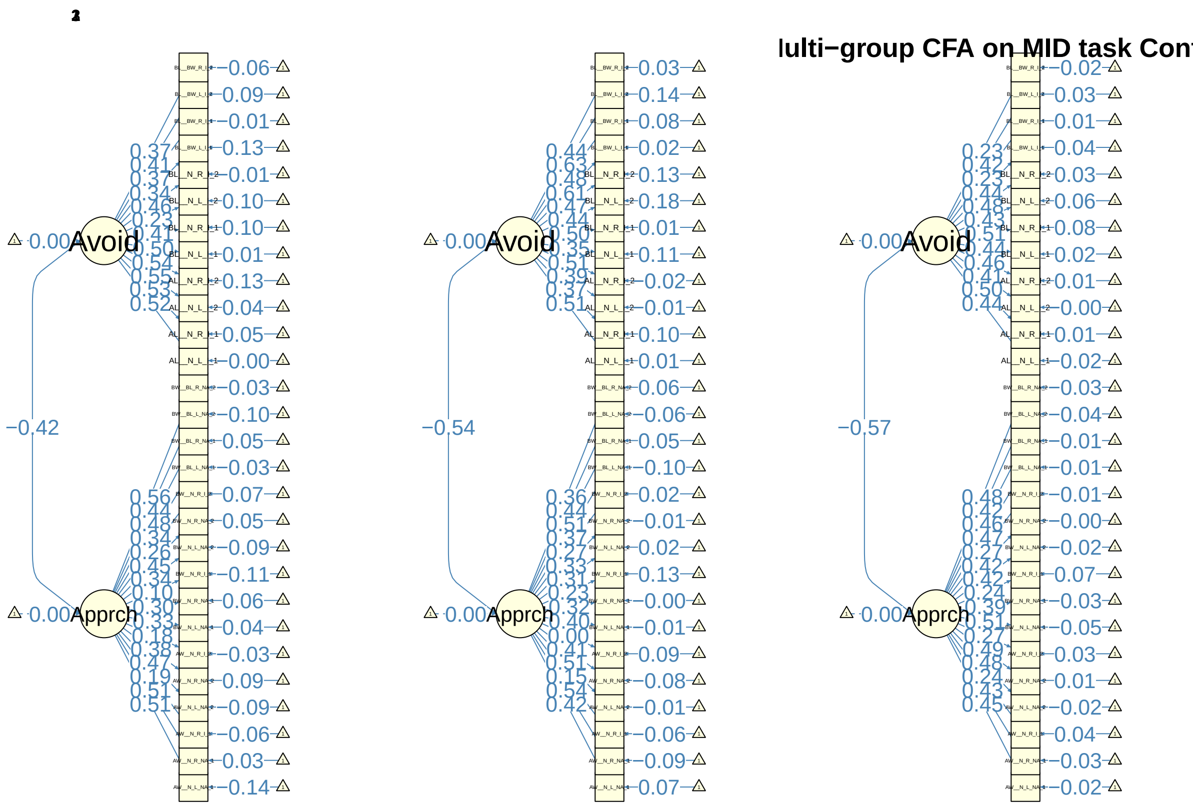
```
## Scaled Chi-Squared Difference Test (method = "satorra.bentler.2001")
##
## lavaan NOTE:
##     The "Chisq" column contains standard test statistics, not the
##     robust test that should be reported per model. A robust difference
##     test is a function of two standard (not robust) statistics.
##
##                  Df    AIC    BIC  Chisq Chisq diff Df diff Pr(>Chisq)
## configural_cfa 1047 -21482 -20170 2961.4
## metric_cfa     1099 -21523 -20479 3024.3     53.862      52      0.403
```

Plotting multi-group config. CFA

Use semPaths to plot the configural invariance CFA multigroup model

```
# this plottinig is not function with runs loading onto ROIs

layout(t(1:3))
```

```r
semPaths(configural_cfa,
         color = "lightyellow",
         theme="colorblind",
         whatLabels = "std",
         style = "lisrel",
         sizeLat = 10,
         sizeLat2 = 10,
         sizeMan = 6,
         edge.color = "steelblue",
         edge.label.cex = 2,
         label.cex = 2,
         rotation = 2,
         layout = "tree2",
         intercepts = TRUE,
         residuals = FALSE,
         #residScale = 10,
         curve = 2,
         title = T,
         title.color = "black",
         cardinal = "lat cov",
         curvePivot = T,
         nCharNodes = 6,
         #nodeLabels = label,
         mar = c(2,5,2,6))
# Title
title("Multi-group CFA on MID task Contrasts")
```



Running [semi-restricted] ESEM Model

As described in the manuscript, the restricted CFA may incorrectly account for some measurement error in the items. This may degrade the fit statistics. See Marsh et al. (2014) for an in-depth discussion.

In this case, Exploratory Structural Equation Modeling (ESEM) is used to fit a CFA pre-specified model that allows for non-zero loadings. The technique and application of ESEM is available through the psych esem and esemcomp package. Here, the `esemcomp` package is used to fit a model using the steps described by Mateus Silvestrin here and by Guàrdia-Olmos et al.. The github code for esemcomp is available here. Below can be used to download the esemComp package – which worked with R version 4.2.1 on x86_64-apply-dawin17.0 during September 2022.

```
devtools::install_github("MateusPsi/esemComp", build_vignettes = TRUE)
```

Selected items for ESEM

First, select the items that are consistent with those in the CFA model

```
# ordering so can specify numerically
esem_data = brain_set[,c("AWin_v_Neut_L_NAcc_run1"  ,"AWin_v_Neut_L_NAcc_run2" ,
                         "BWin_v_Neut_L_NAcc_run1"  ,"BWin_v_Neut_L_NAcc_run2" ,
                         "BWin_v_BLose_L_NAcc_run1" ,"BWin_v_BLose_L_NAcc_run2",
                          "AWin_v_Neut_R_NAcc_run1" , "AWin_v_Neut_R_NAcc_run2",
                          "BWin_v_Neut_R_NAcc_run1" , "BWin_v_Neut_R_NAcc_run2",
                          "BWin_v_BLose_R_NAcc_run1", "BWin_v_BLose_R_NAcc_run2",
                         # insula values apprach
                         "AWin_v_Neut_R_Insula_run1","AWin_v_Neut_R_Insula_run2",
```

```r
                             "BWin_v_Neut_R_Insula_run1","BWin_v_Neut_R_Insula_run2",
                             # avoidance
                             "ALose_v_Neut_L_Insula_run1","ALose_v_Neut_L_Insula_run2",
                             "BLose_v_Neut_L_Insula_run1","BLose_v_Neut_L_Insula_run2",
                             "BLose_v_BWin_L_Insula_run1","BLose_v_BWin_L_Insula_run2",
                             "ALose_v_Neut_R_Insula_run1","ALose_v_Neut_R_Insula_run2",
                             "BLose_v_Neut_R_Insula_run1","BLose_v_Neut_R_Insula_run2",
                             "BLose_v_BWin_R_Insula_run1","BLose_v_BWin_R_Insula_run2",
                             "set")]
```

Specify EFA Model

As described in March et al. (2014), create a target rotation for items onto factors. In this case two factors are specified by the CFA model, so factor 1 and factor 2 are specified in `make_target`.

```r
target_rot <- make_target(28,mainloadings = list(f1 = 1:16, f2 = 17:28))
esem.efa <- esem_efa(data = esem_data[,1:28], nfactors = 2,
                     target = target_rot, fm = "ml")
```

```
## Loading required namespace: GPArotation
```

```r
esem.efa$loadings
```

```
##
## Loadings:
##                            ML1    ML2
## AWin_v_Neut_L_NAcc_run1     0.421
## AWin_v_Neut_L_NAcc_run2     0.459
## BWin_v_Neut_L_NAcc_run1     0.455
## BWin_v_Neut_L_NAcc_run2     0.414
## BWin_v_BLose_L_NAcc_run1    0.398
## BWin_v_BLose_L_NAcc_run2    0.364
## AWin_v_Neut_R_NAcc_run1     0.516
## AWin_v_Neut_R_NAcc_run2     0.564  0.112
## BWin_v_Neut_R_NAcc_run1     0.391
## BWin_v_Neut_R_NAcc_run2     0.419
## BWin_v_BLose_R_NAcc_run1    0.450
## BWin_v_BLose_R_NAcc_run2    0.461
## AWin_v_Neut_R_Insula_run1   0.204
## AWin_v_Neut_R_Insula_run2   0.226
## BWin_v_Neut_R_Insula_run1   0.245
## BWin_v_Neut_R_Insula_run2   0.255
## ALose_v_Neut_L_Insula_run1         0.470
## ALose_v_Neut_L_Insula_run2         0.435
## BLose_v_Neut_L_Insula_run1         0.450
## BLose_v_Neut_L_Insula_run2         0.431
## BLose_v_BWin_L_Insula_run1         0.462
## BLose_v_BWin_L_Insula_run2         0.451
## ALose_v_Neut_R_Insula_run1         0.466
## ALose_v_Neut_R_Insula_run2         0.427
## BLose_v_Neut_R_Insula_run1         0.502
## BLose_v_Neut_R_Insula_run2         0.489
## BLose_v_BWin_R_Insula_run1         0.228
## BLose_v_BWin_R_Insula_run2         0.215
##
```

```
##                   ML1   ML2
## SS loadings    2.616 2.243
## Proportion Var 0.093 0.080
## Cumulative Var 0.093 0.174
```

Using item that loads highest on factor 1 and lowest on factor 2 and vice versa, and define as anchor using `find_referents`

```r
# per the example from Mateus Silverstrin, need to define anchor for each factor (value to loads higher
anchor <- find_referents(efa_object = esem.efa,factor_names = c("f1","f2"))
```

Once the esem efa and anchors are defined, use `syntax_composer` to specied the esem model. This will produce a lavaan specified model that references starting values that will be used in the cfa model

```r
# Pull starting parameters
esem_mid_model <- syntax_composer(efa_object = esem.efa, referents = anchor)
```

Run ESEM model

Specified Model

The starting values are printed below to provide reference for how starting values differ from a strict CFA model. Notice, how some values that were original not fit onto the Approach factor (f1), such as big lose contrasts, they are now specified with loading values that are between .05 to -.05.

```r
cat(esem_mid_model)
```

```
## f1 =~ start(0.421)*AWin_v_Neut_L_NAcc_run1+
## start(0.459)*AWin_v_Neut_L_NAcc_run2+
## start(0.455)*BWin_v_Neut_L_NAcc_run1+
## start(0.414)*BWin_v_Neut_L_NAcc_run2+
## start(0.398)*BWin_v_BLose_L_NAcc_run1+
## start(0.364)*BWin_v_BLose_L_NAcc_run2+
## start(0.516)*AWin_v_Neut_R_NAcc_run1+
## start(0.564)*AWin_v_Neut_R_NAcc_run2+
## start(0.391)*BWin_v_Neut_R_NAcc_run1+
## start(0.419)*BWin_v_Neut_R_NAcc_run2+
## start(0.45)*BWin_v_BLose_R_NAcc_run1+
## start(0.461)*BWin_v_BLose_R_NAcc_run2+
## start(0.204)*AWin_v_Neut_R_Insula_run1+
## start(0.226)*AWin_v_Neut_R_Insula_run2+
## start(0.245)*BWin_v_Neut_R_Insula_run1+
## start(0.255)*BWin_v_Neut_R_Insula_run2+
## start(0.012)*ALose_v_Neut_L_Insula_run1+
## start(0.016)*ALose_v_Neut_L_Insula_run2+
## start(0.012)*BLose_v_Neut_L_Insula_run1+
## start(0.025)*BLose_v_Neut_L_Insula_run2+
## start(0.01)*BLose_v_BWin_L_Insula_run1+
## start(0)*BLose_v_BWin_L_Insula_run2+
## start(-0.019)*ALose_v_Neut_R_Insula_run1+
## start(-0.054)*ALose_v_Neut_R_Insula_run2+
## 0.004*BLose_v_Neut_R_Insula_run1+
## start(0.009)*BLose_v_Neut_R_Insula_run2+
## start(-0.058)*BLose_v_BWin_R_Insula_run1+
## start(-0.073)*BLose_v_BWin_R_Insula_run2
##
## f2 =~ start(-0.034)*AWin_v_Neut_L_NAcc_run1+
```

```
## start(-0.034)*AWin_v_Neut_L_NAcc_run2+
## start(-0.037)*BWin_v_Neut_L_NAcc_run1+
## start(0.019)*BWin_v_Neut_L_NAcc_run2+
## start(-0.063)*BWin_v_BLose_L_NAcc_run1+
## start(-0.084)*BWin_v_BLose_L_NAcc_run2+
## start(0.079)*AWin_v_Neut_R_NAcc_run1+
## 0.112*AWin_v_Neut_R_NAcc_run2+
## start(0.03)*BWin_v_Neut_R_NAcc_run1+
## start(0.011)*BWin_v_Neut_R_NAcc_run2+
## start(-0.029)*BWin_v_BLose_R_NAcc_run1+
## start(-0.02)*BWin_v_BLose_R_NAcc_run2+
## start(-0.03)*AWin_v_Neut_R_Insula_run1+
## start(-0.012)*AWin_v_Neut_R_Insula_run2+
## start(0.021)*BWin_v_Neut_R_Insula_run1+
## start(-0.026)*BWin_v_Neut_R_Insula_run2+
## start(0.47)*ALose_v_Neut_L_Insula_run1+
## start(0.435)*ALose_v_Neut_L_Insula_run2+
## start(0.45)*BLose_v_Neut_L_Insula_run1+
## start(0.431)*BLose_v_Neut_L_Insula_run2+
## start(0.462)*BLose_v_BWin_L_Insula_run1+
## start(0.451)*BLose_v_BWin_L_Insula_run2+
## start(0.466)*ALose_v_Neut_R_Insula_run1+
## start(0.427)*ALose_v_Neut_R_Insula_run2+
## start(0.502)*BLose_v_Neut_R_Insula_run1+
## start(0.489)*BLose_v_Neut_R_Insula_run2+
## start(0.228)*BLose_v_BWin_R_Insula_run1+
## start(0.215)*BLose_v_BWin_R_Insula_run2
```

Running full ESEM model

After the EFA loadings are extracted using a target rotation, starting values are now available. These are now used to specify a less restrictive CFA model

```
esem_mid_fit<- cfa(esem_mid_model, esem_data[,1:28], std.lv=TRUE, meanstructure = TRUE,
                   estimator = "MLR")
```

Pull and add fit statistics to the `out` dataframe and print results to see decreases in AIC/BIC

```
# adding values to the CFA model fit indices
out[4,2:8] <- round(data.matrix(fitmeasures(esem_mid_fit,
                                            fit.measures = c("chisq","df","pvalue",
                                                             "rmsea", "cfi","tli", "srmr"))),
                    digits=3)
out[4,9] <- round(AIC(esem_mid_fit),3)
out[4,10] <- round(BIC(esem_mid_fit),3)
out[4,1] <-  c("Overall ESEM")

out <- as.data.frame(out)

out %>%
  knitr::kable(
    col.names = c("Model", "Chi-sq", "DF", "p value", "RMSEA", "CFI", "TLI","SRMR", "AIC", "BIC"),
    caption = "Fit statistics from MG-CFA and ESEM models",
    booktabs = TRUE
    )
```

Table 1: Fit statistics from MG-CFA and ESEM models

Model	Chi-sq	DF	p value	RMSEA	CFI	TLI	SRMR	AIC	BIC
Overall CFA	2200.622	349	0	0.065	0.657	0.628	0.053	-21672.085	-21234.81
Configg MG-CFA	2961.409	1047	0	0.066	0.65	0.621	0.06	-21482.229	-20170.405
Metric MG-CFA	3024.264	1099	0	0.064	0.648	0.637	0.062	-21523.374	-20479.059
Overall ESEM	2174.511	323	0	0.067	0.657	0.598	0.052	-21646.197	-21075.167

Aim: 2. Running EFA [Unrestricted] model

Here, a data-driven exploratory factor analysis is performed as implemented using the (https://www.rdocumentation.org/packages/stats/versions/3.6.2/topics/factanal)[https://www.rdocumentation.org/packages/stats/versions/3.6.2/topics/factanal] in the stats package. The same variables as in the CFA and ESEM dataset are used. A tutorial from Nilam Ram on EFA is also available here

By Sample EFA

Used the (factanal)[https://www.rdocumentation.org/packages/stats/versions/3.6.2/topics/factanal] to run EFA model. Specifying the number of factors and using the `promax` (non-orthogonal) rotation.

```
abcd_efa_df <- esem_data %>% filter(set == 1)
mls_efa_df <- esem_data %>% filter(set == 2)
ahrb_efa_df <- esem_data %>% filter(set == 3)
```

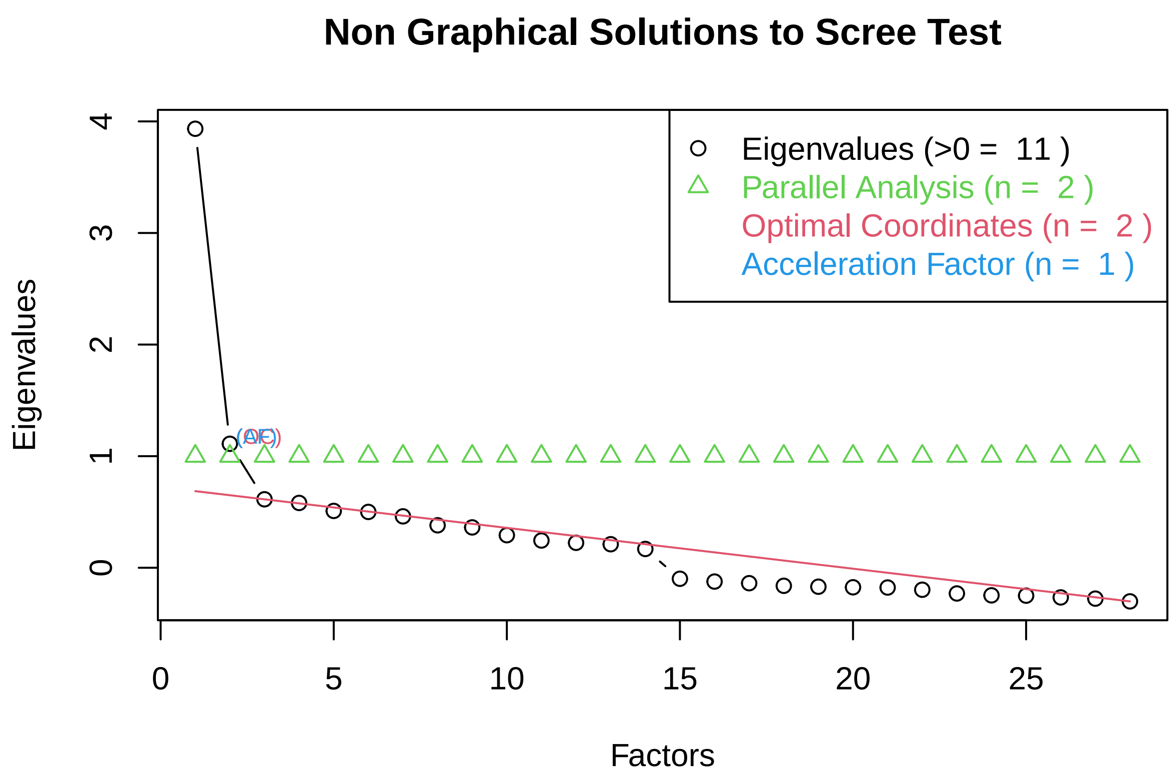
Sample: *ABCD*

A number of methods can be used to estimate the recommended data driven factors in the data. There are nuanced differences in calculations between packages and methods. Therefore, a number of packages are used to acquire consistent evidence to acquire the most parsimonious model.

Using nFactors package see the recommended factors for the EFA model using a number of models.

```
fa_abcd <- subset(abcd_efa_df[,1:28])
```

```
plot(nScree(x=fa_abcd,model="factors"))
```

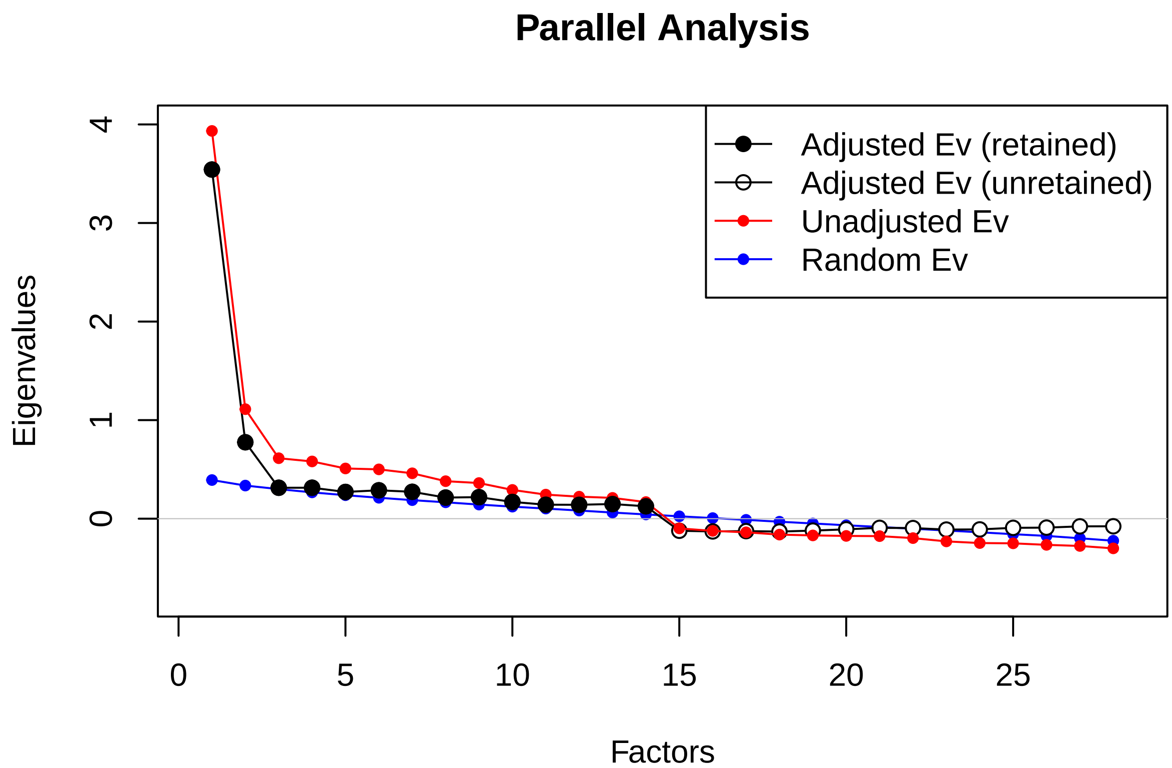


To avoid biasing of packages in different calculations of recommendation factors that depend on strong correlations between bilateral regions by using parallel analysis. Parallel analysis is also implemented using the paran package.

```
paran(x = fa_abcd,
      iterations = 1000, quietly = FALSE, centile = 95,
      status = FALSE, all = TRUE, cfa = TRUE, graph = TRUE, color = TRUE,
      col = c("black", "red", "blue"), lty = c(1, 2, 3), lwd = 1, legend = TRUE,
      seed = 100)
```

```
##
## Using eigendecomposition of correlation matrix.
##
## Results of Horn's Parallel Analysis for factor retention
## 1000 iterations, using the 95 centile estimate
##
## --------------------------------------------------
## Factor      Adjusted    Unadjusted    Estimated
##             Eigenvalue  Eigenvalue    Bias
## --------------------------------------------------
## No components passed.
## --------------------------------------------------
## 1           3.542178    3.934015      0.391836
## 2           0.775202    1.111071      0.335869
## 3           0.313670    0.613506      0.299835
## 4           0.313864    0.580679      0.266815
## 5           0.271159    0.509467      0.238307
```

```
## 6           0.287997    0.500248      0.212250
## 7           0.272560    0.460549      0.187988
## 8           0.214122    0.380351      0.166228
## 9           0.218417    0.361595      0.143178
## 10          0.169955    0.291872      0.121916
## 11          0.141768    0.244225      0.102457
## 12          0.140901    0.223414      0.082512
## 13          0.148606    0.210788      0.062181
## 14          0.125450    0.168313      0.042863
## 15         -0.122634   -0.09868      0.023946
## 16         -0.129530   -0.12376      0.005761
## 17         -0.126362   -0.13831     -0.01194
## 18         -0.130790   -0.16166     -0.03087
## 19         -0.121125   -0.16974     -0.04861
## 20         -0.108212   -0.17507     -0.06685
## 21         -0.093136   -0.17720     -0.08406
## 22         -0.095347   -0.19714     -0.10179
## 23         -0.110512   -0.23053     -0.12002
## 24         -0.109017   -0.24716     -0.13814
## 25         -0.092890   -0.25014     -0.15725
## 26         -0.090723   -0.26576     -0.17504
## 27         -0.077950   -0.27668     -0.19873
## 28         -0.077263   -0.30109     -0.22383
## --------------------------------------------------
##
## Adjusted eigenvalues > 0 indicate dimensions to retain.
## (14 factors    retained)
```



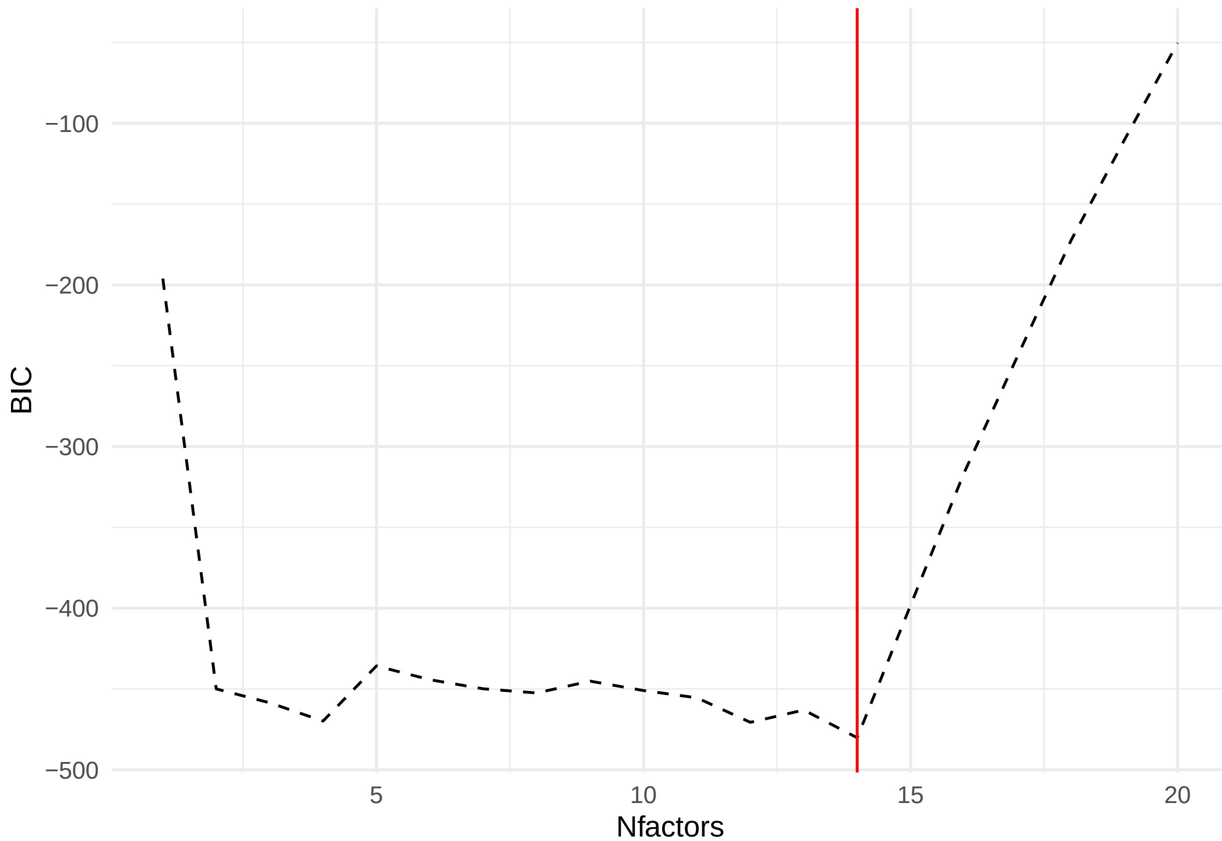
Comparing the above with the BIC comparison of an EFA model to determine the best fitting model based on fit statistics. Factor Analysis is submitted across a range of factors, e.g., 1-5, and the BIC is extracted from the model to determine the optimal number of factors

```
abcd_rec_factors <- matrix(NA, ncol = 2, nrow = 20)
colnames(abcd_rec_factors) <- c("Nfactors","BIC")

for (f in 1:20) {
  test_fac <- fa(r = fa_abcd,  #raw data
            nfactors = f, fm = 'minres',
            rotate = "oblimin")
  abcd_rec_factors[f,1] <- f
  abcd_rec_factors[f,2] <-test_fac$BIC
}
```

```
## Warning in GPFoblq(L, Tmat = Tmat, normalize = normalize, eps = eps, maxit =
## maxit, : convergence not obtained in GPFoblq. 1000 iterations used.

## Warning in GPFoblq(L, Tmat = Tmat, normalize = normalize, eps = eps, maxit =
## maxit, : convergence not obtained in GPFoblq. 1000 iterations used.

## Warning in GPFoblq(L, Tmat = Tmat, normalize = normalize, eps = eps, maxit =
## maxit, : convergence not obtained in GPFoblq. 1000 iterations used.

## Warning in GPFoblq(L, Tmat = Tmat, normalize = normalize, eps = eps, maxit =
## maxit, : convergence not obtained in GPFoblq. 1000 iterations used.
```

```
abcd_bic_fact = as.data.frame(abcd_rec_factors)
```

```
abcd_lowest_bic <- which.min(abcd_bic_fact$BIC)

abcd_bic_fact %>%
  ggplot(aes(x = Nfactors, y = BIC)) +
  geom_line(colour = 'black', linetype = 'dashed') +
  geom_vline(xintercept = abcd_bic_fact$Nfactors[abcd_lowest_bic], colour = 'red')+
  theme_minimal()
```



Running EFA

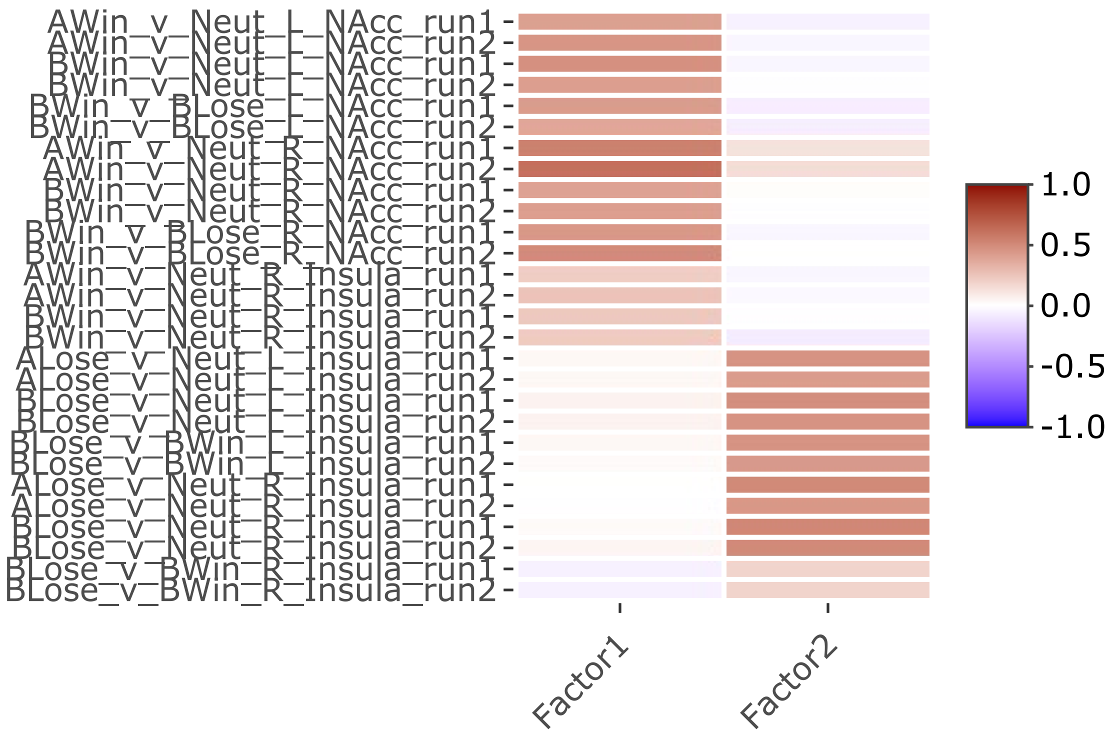
```
abcd_efa <- factanal(x = fa_abcd,  #raw data
              factors = 2, fm ='minres', rotation = "promax" # oblique rotation allow for non-orthogona
              )
```

```
heatmaply(round(abcd_efa$loadings[,1:2],2) %>% print(sort = T),
          scale_fill_gradient_fun = ggplot2::scale_fill_gradient2(
                 low = "blue",
                 high = "darkred",
                 space = "Lab",
                 midpoint = 0,
                 limits = c(-1, 1)
               ),
               dendrogram = "none",
               xlab = "", ylab = "",
               main = "",
```

```r
               margins = c(60,100,40,20),
               grid_color = "white",
               grid_width = 0.00001,
               titleX = FALSE,
               hide_colorbar = FALSE,
               branches_lwd = 0.1,
               label_names = c("Brain:", "Feature:", "Value"),
               fontsize_row = 9, fontsize_col = 9,
               labCol = colnames(abcd_efa$loadings[,1:2]),
               labRow = rownames(abcd_efa$loadings[,1:2]),
               heatmap_layers = theme(axis.line=element_blank()),

)
```

```
##                            Factor1 Factor2
## AWin_v_Neut_L_NAcc_run1       0.40   -0.06
## AWin_v_Neut_L_NAcc_run2       0.45   -0.04
## BWin_v_Neut_L_NAcc_run1       0.47   -0.04
## BWin_v_Neut_L_NAcc_run2       0.41   -0.01
## BWin_v_BLose_L_NAcc_run1      0.42   -0.07
## BWin_v_BLose_L_NAcc_run2      0.37   -0.07
## AWin_v_Neut_R_NAcc_run1       0.54    0.12
## AWin_v_Neut_R_NAcc_run2       0.61    0.14
## BWin_v_Neut_R_NAcc_run1       0.39    0.01
## BWin_v_Neut_R_NAcc_run2       0.41   -0.01
## BWin_v_BLose_R_NAcc_run1      0.44   -0.04
## BWin_v_BLose_R_NAcc_run2      0.49    0.00
## AWin_v_Neut_R_Insula_run1     0.21   -0.04
## AWin_v_Neut_R_Insula_run2     0.25   -0.03
## BWin_v_Neut_R_Insula_run1     0.23   -0.01
## BWin_v_Neut_R_Insula_run2     0.22   -0.08
## ALose_v_Neut_L_Insula_run1    0.03    0.46
## ALose_v_Neut_L_Insula_run2    0.03    0.42
## BLose_v_Neut_L_Insula_run1    0.05    0.48
## BLose_v_Neut_L_Insula_run2    0.05    0.46
## BLose_v_BWin_L_Insula_run1    0.03    0.46
## BLose_v_BWin_L_Insula_run2    0.02    0.43
## ALose_v_Neut_R_Insula_run1    0.00    0.50
## ALose_v_Neut_R_Insula_run2   -0.01    0.44
## BLose_v_Neut_R_Insula_run1    0.02    0.51
## BLose_v_Neut_R_Insula_run2    0.04    0.50
## BLose_v_BWin_R_Insula_run1   -0.06    0.18
## BLose_v_BWin_R_Insula_run2   -0.06    0.18
```

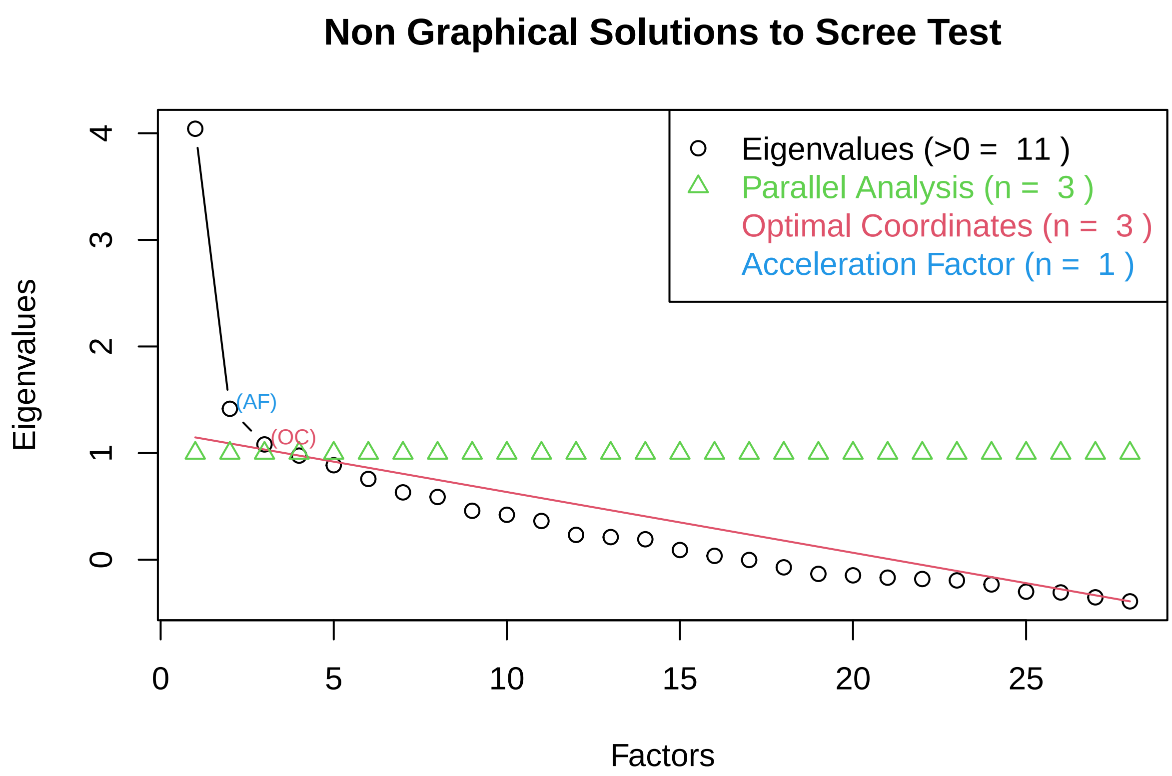
```
## Warning: `gather_()` was deprecated in tidyr 1.2.0.
## i Please use `gather()` instead.
## i The deprecated feature was likely used in the plotly package.
##   Please report the issue at <https://github.com/plotly/plotly.R/issues>.
```



Sample: *MLS*

```
fa_mls <- subset(mls_efa_df[,1:28])
```

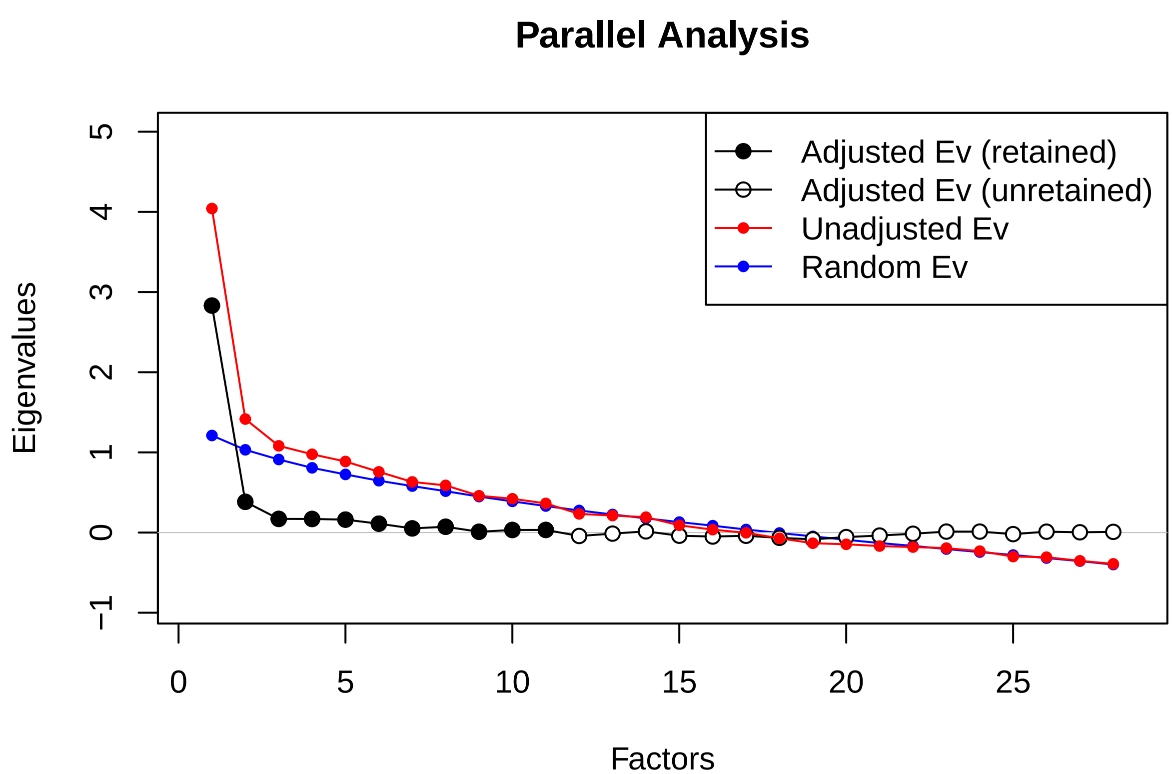
```
plot(nScree(x=fa_mls,model="factors"))
```




```r
paran(x = fa_mls,
      iterations = 1000, quietly = FALSE, centile = 95,
      status = FALSE, all = TRUE, cfa = TRUE, graph = TRUE, color = TRUE,
      col = c("black", "red", "blue"), lty = c(1, 2, 3), lwd = 1, legend = TRUE,
      seed = 100)
```

```
##
## Using eigendecomposition of correlation matrix.
##
## Results of Horn's Parallel Analysis for factor retention
## 1000 iterations, using the 95 centile estimate
##
## --------------------------------------------------
## Factor      Adjusted    Unadjusted    Estimated
##             Eigenvalue  Eigenvalue    Bias
## --------------------------------------------------
## 1           2.831197    4.041832      1.210634
## 2           0.383345    1.415416      1.032070
## 3           0.170152    1.081438      0.911286
## 4           0.169402    0.976468      0.807066
## 5           0.161286    0.886233      0.724946
## 6           0.110471    0.757309      0.646837
## 7           0.051888    0.631413      0.579524
## 8           0.072342    0.588307      0.515965
## 9           0.009946    0.458769      0.448823
## 10          0.032104    0.421235      0.389130
```

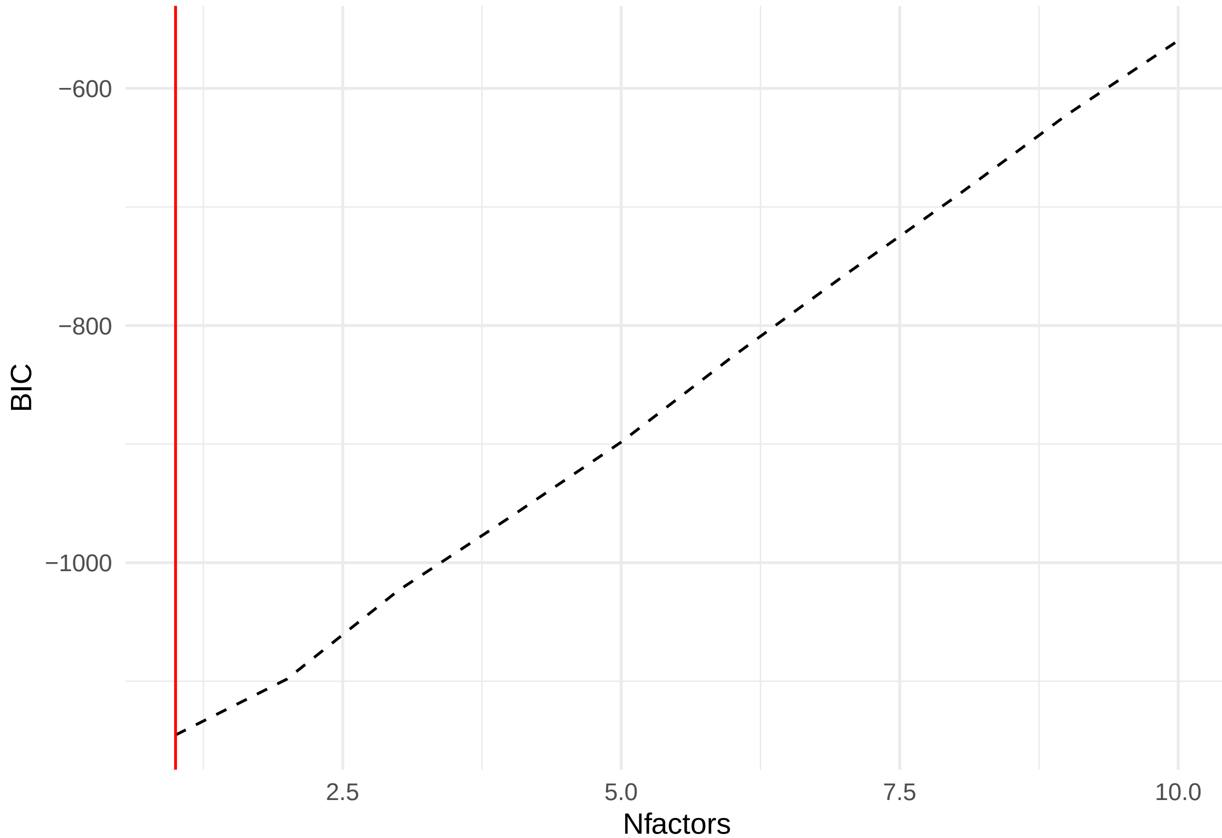
```
## 11          0.032538    0.363316      0.330777
## 12         -0.042810    0.232814      0.275625
## 13         -0.013344    0.211961      0.225305
## 14          0.014754    0.191760      0.177006
## 15         -0.039773    0.090890      0.130663
## 16         -0.049639    0.036074      0.085713
## 17         -0.040487   -0.00265      0.037831
## 18         -0.066096   -0.07167     -0.00558
## 19         -0.084986   -0.13281     -0.04782
## 20         -0.056403   -0.14665     -0.09025
## 21         -0.037420   -0.16846     -0.13104
## 22         -0.014872   -0.18121     -0.16634
## 23          0.011672   -0.19418     -0.20586
## 24          0.011884   -0.23185     -0.24374
## 25         -0.019505   -0.29958     -0.28007
## 26          0.010809   -0.30709     -0.31790
## 27          0.003662   -0.35286     -0.35652
## 28          0.008372   -0.39046     -0.39884
## --------------------------------------------------
##
## Adjusted eigenvalues > 0 indicate dimensions to retain.
## (11 factors    retained)
```




```r
mls_rec_factors <- matrix(NA, ncol = 2, nrow = 10)
colnames(mls_rec_factors) <- c("Nfactors","BIC")

for (f in 1:10) {
```

```
  test_fac <- fa(r = fa_mls,  #raw data
            nfactors = f, fm = 'minres',
            rotate = "promax")
  mls_rec_factors[f,1] <- f
  mls_rec_factors[f,2] <-test_fac$BIC
}

mls_bic_fact = as.data.frame(mls_rec_factors)
```

```
mls_lowest_bic <- which.min(mls_bic_fact$BIC)

mls_bic_fact %>%
  ggplot(aes(x = Nfactors, y = BIC)) +
  geom_line(colour = 'black', linetype = 'dashed') +
  geom_vline(xintercept = mls_bic_fact$Nfactors[mls_lowest_bic], colour = 'red')+
  theme_minimal()
```



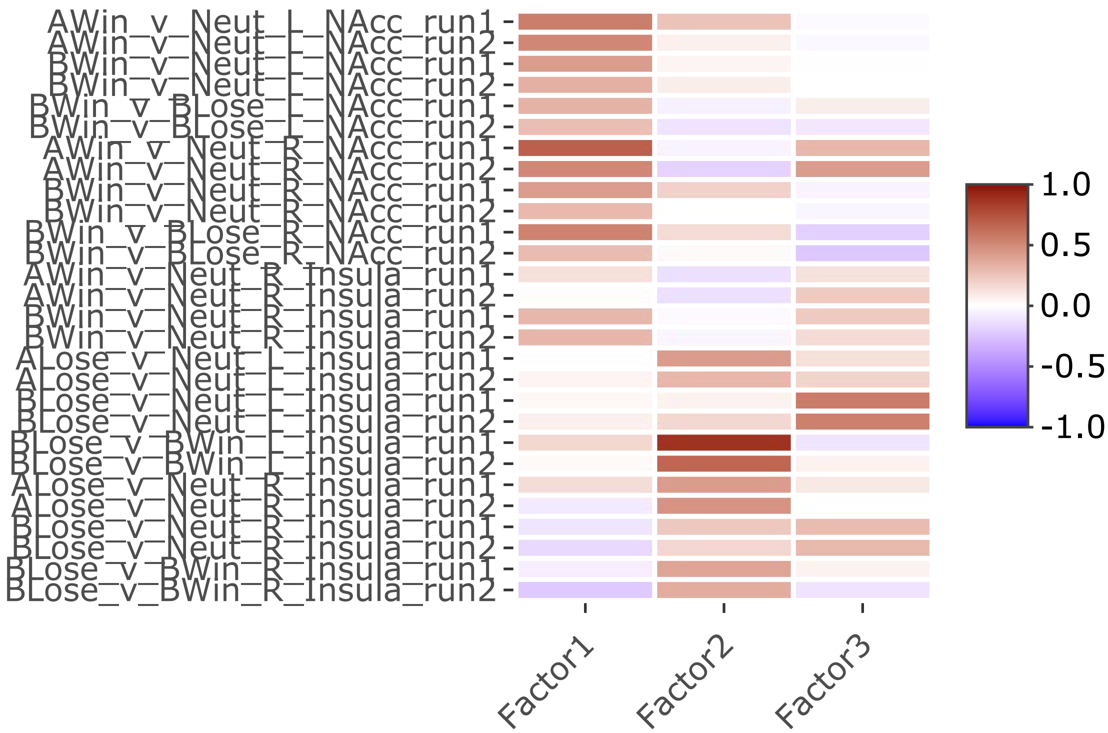
MLS factor analysis

```
mls_efa <- factanal(x = fa_mls,  #raw data
              factors = 3, rotation = "promax" # oblique rotation allow for non-orthogonal structure
              )
```

```
heatmaply(round(mls_efa$loadings[,1:3],2) %>% print(sort = T),
          scale_fill_gradient_fun = ggplot2::scale_fill_gradient2(
                low = "blue",
                high = "darkred",
```

```
                  space = "Lab",
                  midpoint = 0,
                  limits = c(-1, 1)
                ),
                dendrogram = "none",
                xlab = "", ylab = "",
                main = "",
                margins = c(60,100,40,20),
                grid_color = "white",
                grid_width = 0.00001,
                titleX = FALSE,
                hide_colorbar = FALSE,
                branches_lwd = 0.1,
                label_names = c("Brain:", "Feature:", "Value"),
                fontsize_row = 9, fontsize_col = 9,
                labCol = colnames(mls_efa$loadings[,1:3]),
                labRow = rownames(mls_efa$loadings[,1:3]),
                heatmap_layers = theme(axis.line=element_blank()),

)
```

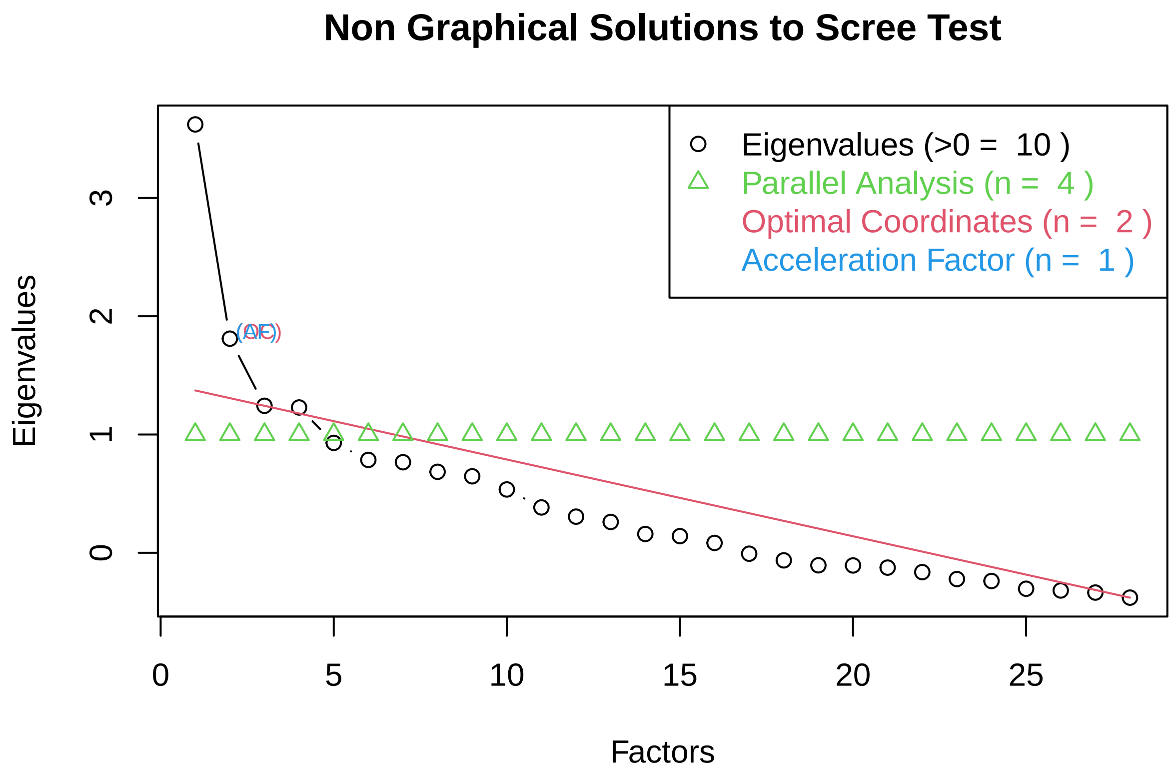
```
##                              Factor1 Factor2 Factor3
## AWin_v_Neut_L_NAcc_run1        0.55    0.25   -0.02
## AWin_v_Neut_L_NAcc_run2        0.51    0.06   -0.03
## BWin_v_Neut_L_NAcc_run1        0.42    0.04   -0.01
## BWin_v_Neut_L_NAcc_run2        0.33    0.07    0.00
## BWin_v_BLose_L_NAcc_run1       0.32   -0.06    0.07
## BWin_v_BLose_L_NAcc_run2       0.27   -0.12   -0.10
## AWin_v_Neut_R_NAcc_run1        0.68   -0.05    0.30
## AWin_v_Neut_R_NAcc_run2        0.51   -0.19    0.42
## BWin_v_Neut_R_NAcc_run1        0.42    0.19   -0.05
## BWin_v_Neut_R_NAcc_run2        0.29    0.00   -0.04
## BWin_v_BLose_R_NAcc_run1       0.53    0.15   -0.20
## BWin_v_BLose_R_NAcc_run2       0.28    0.02   -0.24
## AWin_v_Neut_R_Insula_run1      0.13   -0.13    0.12
## AWin_v_Neut_R_Insula_run2      0.01   -0.13    0.22
## BWin_v_Neut_R_Insula_run1      0.30   -0.02    0.22
## BWin_v_Neut_R_Insula_run2      0.30   -0.04    0.15
## ALose_v_Neut_L_Insula_run1    -0.01    0.42    0.13
## ALose_v_Neut_L_Insula_run2     0.04    0.30    0.18
## BLose_v_Neut_L_Insula_run1     0.03    0.05    0.56
## BLose_v_Neut_L_Insula_run2     0.06    0.16    0.54
## BLose_v_BWin_L_Insula_run1     0.16    0.87   -0.11
## BLose_v_BWin_L_Insula_run2     0.02    0.65    0.05
## ALose_v_Neut_R_Insula_run1     0.14    0.42    0.09
## ALose_v_Neut_R_Insula_run2    -0.09    0.46    0.00
## BLose_v_Neut_R_Insula_run1    -0.11    0.23    0.28
## BLose_v_Neut_R_Insula_run2    -0.16    0.17    0.29
## BLose_v_BWin_R_Insula_run1    -0.07    0.38    0.05
## BLose_v_BWin_R_Insula_run2    -0.23    0.35   -0.12
```



Sample: ***AHRB***

```r
fa_ahrb <- subset(ahrb_efa_df[,1:28])
```

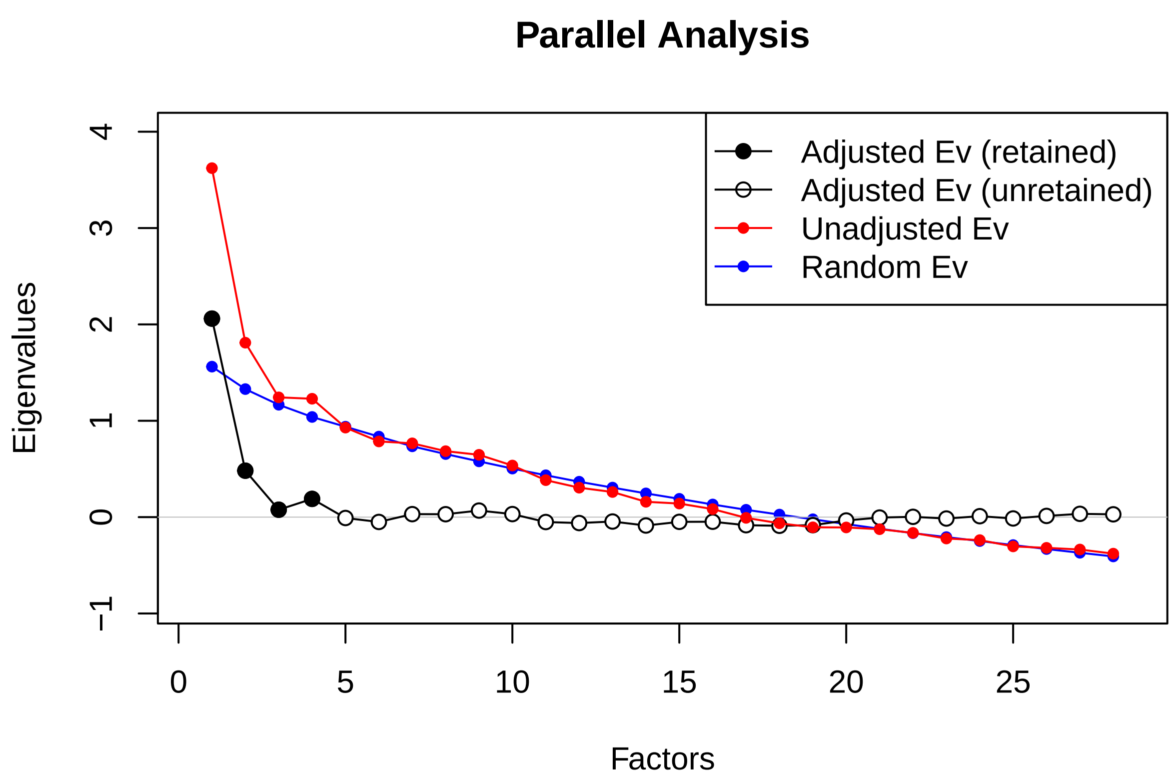
```r
plot(nScree(x=fa_ahrb,model="factors"))
```




```
paran(x = fa_ahrb,
      iterations = 1000, quietly = FALSE, centile = 95,
      status = FALSE, all = TRUE, cfa = TRUE, graph = TRUE, color = TRUE,
      col = c("black", "red", "blue"), lty = c(1, 2, 3), lwd = 1, legend = TRUE,
      seed = 100)
```

```
##
## Using eigendecomposition of correlation matrix.
##
## Results of Horn's Parallel Analysis for factor retention
## 1000 iterations, using the 95 centile estimate
##
## --------------------------------------------------
## Factor      Adjusted    Unadjusted    Estimated
##             Eigenvalue  Eigenvalue    Bias
## --------------------------------------------------
## 1            2.060093   3.621941      1.561847
## 2            0.481724   1.810384      1.328660
## 3            0.076629   1.242826      1.166197
## 4            0.189243   1.228597      1.039354
## 5           -0.009360   0.929113      0.938474
## 6           -0.050147   0.785260      0.835407
## 7            0.030826   0.765686      0.734860
## 8            0.030113   0.684464      0.654351
## 9            0.068039   0.646615      0.578576
## 10           0.031492   0.535404      0.503911
```

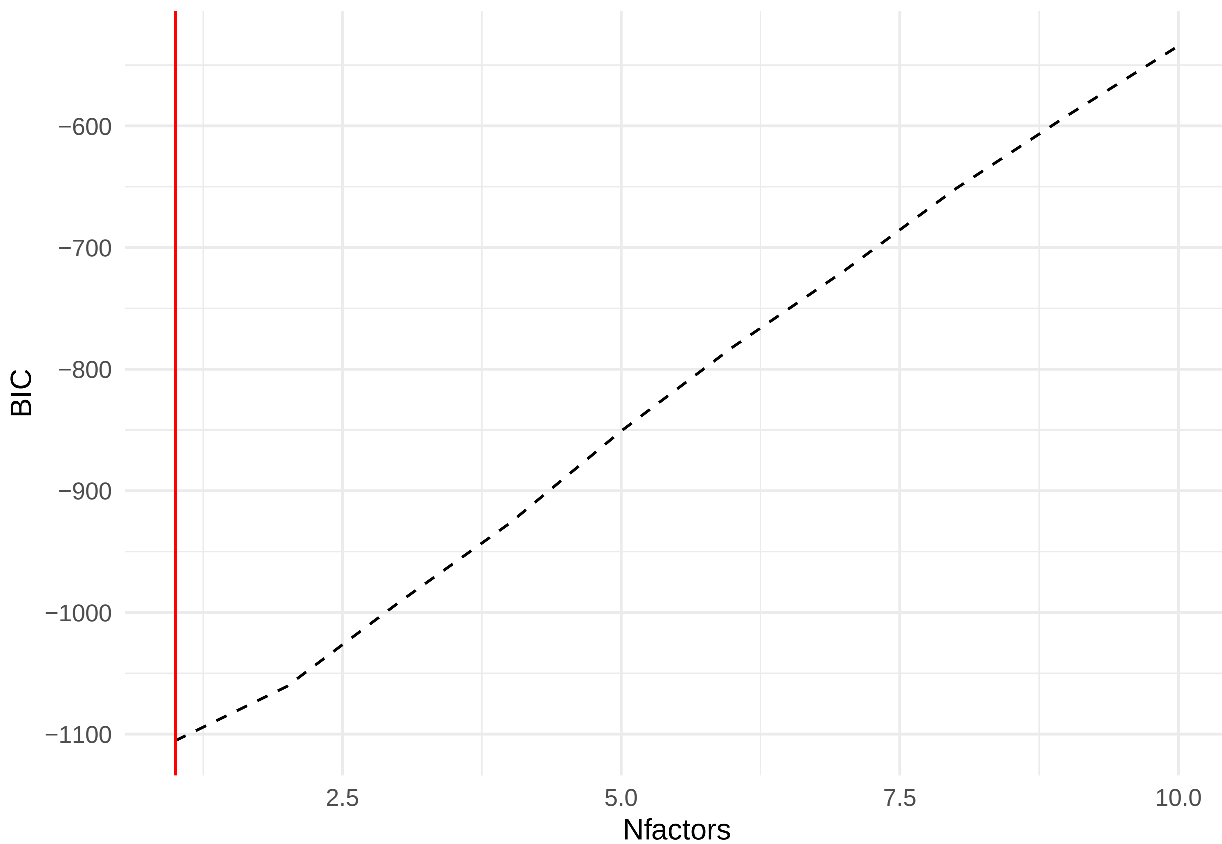
```
## 11          -0.050998    0.383378      0.434377
## 12          -0.061551    0.305570      0.367122
## 13          -0.045811    0.260598      0.306410
## 14          -0.087183    0.158940      0.246124
## 15          -0.048996    0.140764      0.189761
## 16          -0.047886    0.083183      0.131069
## 17          -0.083831   -0.00786      0.075962
## 18          -0.090172   -0.06397      0.026192
## 19          -0.083539   -0.10575     -0.02221
## 20          -0.034889   -0.10723     -0.07234
## 21          -0.004630   -0.12541     -0.12078
## 22           0.002853   -0.16359     -0.16644
## 23          -0.014767   -0.22211     -0.20734
## 24           0.009603   -0.23872     -0.24833
## 25          -0.014051   -0.30408     -0.29003
## 26           0.012237   -0.31880     -0.33104
## 27           0.033983   -0.33629     -0.37027
## 28           0.028916   -0.37918     -0.40809
## --------------------------------------------------
##
## Adjusted eigenvalues > 0 indicate dimensions to retain.
## (4 factors    retained)
```




```r
ahrb_rec_factors <- matrix(NA, ncol = 2, nrow = 10)
colnames(ahrb_rec_factors) <- c("Nfactors","BIC")

for (f in 1:10) {
```

```
  test_fac <- fa(r = fa_ahrb,  #raw data
             nfactors = f, fm = 'minres',
             rotate = "promax")
  ahrb_rec_factors[f,1] <- f
  ahrb_rec_factors[f,2] <-test_fac$BIC
}

ahrb_bic_fact = as.data.frame(ahrb_rec_factors)
```

```
ahrb_lowest_bic <- which.min(ahrb_bic_fact$BIC)

ahrb_bic_fact %>%
  ggplot(aes(x = Nfactors, y = BIC)) +
  geom_line(colour = 'black', linetype = 'dashed') +
  geom_vline(xintercept = ahrb_bic_fact$Nfactors[ahrb_lowest_bic], colour = 'red')+
  theme_minimal()
```



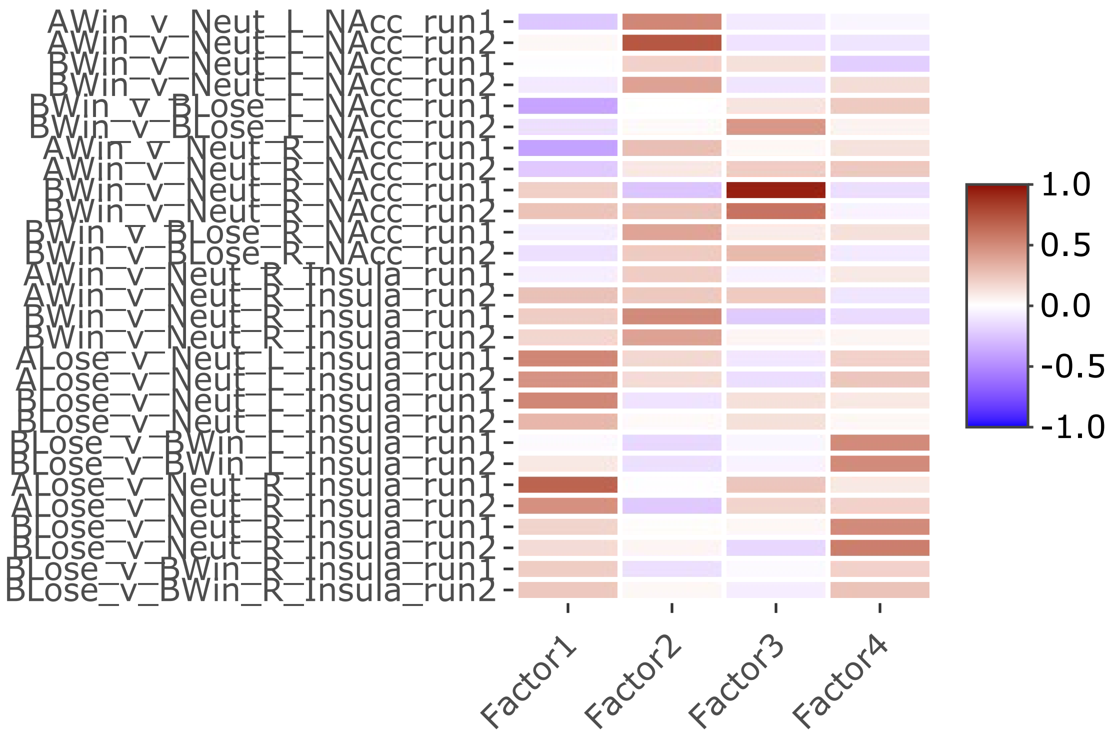
Factor Analysis Model

```
ahrb_efa <- factanal(x = fa_ahrb,  #raw data
              factors = 4, rotation = "promax" # oblique rotation allow for non-orthogonal structure
              )
```

```
heatmaply(round(ahrb_efa$loadings[,1:4],2) %>% print(sort = T),
          scale_fill_gradient_fun = ggplot2::scale_fill_gradient2(
                 low = "blue",
                 high = "darkred",
```

```r
                space = "Lab",
                midpoint = 0,
                limits = c(-1, 1)
              ),
              dendrogram = "none",
              xlab = "", ylab = "",
              main = "",
              margins = c(60,100,40,20),
              grid_color = "white",
              grid_width = 0.00001,
              titleX = FALSE,
              hide_colorbar = FALSE,
              branches_lwd = 0.1,
              label_names = c("Brain:", "Feature:", "Value"),
              fontsize_row = 9, fontsize_col = 9,
              labCol = colnames(ahrb_efa$loadings[,1:4]),
              labRow = rownames(ahrb_efa$loadings[,1:4]),
              heatmap_layers = theme(axis.line=element_blank()),

)
```

```
##                              Factor1 Factor2 Factor3 Factor4
## AWin_v_Neut_L_NAcc_run1        -0.24    0.51   -0.09   -0.04
## AWin_v_Neut_L_NAcc_run2         0.03    0.72   -0.12   -0.11
## BWin_v_Neut_L_NAcc_run1        -0.01    0.19    0.13   -0.21
## BWin_v_Neut_L_NAcc_run2        -0.09    0.39   -0.11    0.14
## BWin_v_BLose_L_NAcc_run1       -0.39    0.00    0.11    0.22
## BWin_v_BLose_L_NAcc_run2       -0.13    0.02    0.44    0.05
## AWin_v_Neut_R_NAcc_run1        -0.40    0.27    0.03    0.12
## AWin_v_Neut_R_NAcc_run2        -0.23    0.10    0.21    0.23
## BWin_v_Neut_R_NAcc_run1         0.20   -0.25    0.93   -0.14
## BWin_v_Neut_R_NAcc_run2         0.25    0.26    0.59   -0.05
## BWin_v_BLose_R_NAcc_run1       -0.08    0.38    0.08    0.13
## BWin_v_BLose_R_NAcc_run2       -0.13    0.22    0.29   -0.09
## AWin_v_Neut_R_Insula_run1      -0.07    0.21   -0.06    0.09
## AWin_v_Neut_R_Insula_run2       0.26    0.23    0.22   -0.11
## BWin_v_Neut_R_Insula_run1       0.21    0.49   -0.22   -0.15
## BWin_v_Neut_R_Insula_run2       0.17    0.39    0.04    0.04
## ALose_v_Neut_L_Insula_run1      0.51    0.16   -0.10    0.19
## ALose_v_Neut_L_Insula_run2      0.46    0.15   -0.14    0.24
## BLose_v_Neut_L_Insula_run1      0.51   -0.12    0.13    0.09
## BLose_v_Neut_L_Insula_run2      0.30    0.02    0.13    0.03
## BLose_v_BWin_L_Insula_run1     -0.02   -0.17   -0.04    0.49
## BLose_v_BWin_L_Insula_run2      0.09   -0.13   -0.05    0.49
## ALose_v_Neut_R_Insula_run1      0.66   -0.01    0.24    0.09
## ALose_v_Neut_R_Insula_run2      0.47   -0.23    0.18    0.19
## BLose_v_Neut_R_Insula_run1      0.18    0.01    0.03    0.49
## BLose_v_Neut_R_Insula_run2      0.15    0.04   -0.17    0.55
## BLose_v_BWin_R_Insula_run1      0.21   -0.13   -0.02    0.19
## BLose_v_BWin_R_Insula_run2      0.23    0.03   -0.07    0.25
```



Quant. Convergence

Calculating a coefficient of factor congruence across the three sample's EFA models. Using function fa.congruence

```
fa.congruence(x = list(abcd_efa, mls_efa, ahrb_efa), digits = 2) %>%
  knitr::kable(
    col.names = c("1. ABCD F1", "2. ABCD F2",
                  "3. MLS F1", "4. MLS F2","5. MLS F3",
                  "6. AHRB F1", "7. AHRB F2","8. AHRB F3", "8. AHRB F4"),
    caption = "ABCD, MLS and AHRB EFA Factor Congruence",
    booktabs = TRUE
    )
```

Table 2: ABCD, MLS and AHRB EFA Factor Congruence

	1. ABCD F1	2. ABCD F2	3. MLS F1	4. MLS F2	5. MLS F3	6. AHRB F1	7. AHRB F2	8. AHRB F3	8. AHRB F4
Factor1	1.00	0.03	0.94	0.03	0.20	-0.17	0.68	0.47	0.07
Factor2	0.03	1.00	-0.02	0.76	0.58	0.68	-0.11	0.04	0.77
Factor1	0.94	-0.02	1.00	0.07	0.15	-0.21	0.68	0.38	-0.01
Factor2	0.03	0.76	0.07	1.00	0.05	0.51	-0.07	0.02	0.67
Factor3	0.20	0.58	0.15	0.05	1.00	0.37	0.13	0.00	0.34
Factor1	-0.17	0.68	-0.21	0.51	0.37	1.00	-0.06	0.17	0.22
Factor2	0.68	-0.11	0.68	-0.07	0.13	-0.06	1.00	-0.12	-0.09

	1. ABCD F1	2. ABCD F2	3. MLS F1	4. MLS F2	5. MLS F3	6. AHRB F1	7. AHRB F2	8. AHRB F3	8. AHRB F4
Factor3	0.47	0.04	0.38	0.02	0.00	0.17	-0.12	1.00	-0.13
Factor4	0.07	0.77	-0.01	0.67	0.34	0.22	-0.09	-0.13	1.00

Running Local SEM

Running CFA for the pubertal variables in the ABCD sample using the local SEM framework described in Olaru et al (2020) implemented using the sirt package

Run LSEM

Specifying the model for the ABCD data below. For now, using the CFA model. In future [real data] implementation, will apply the EFA CFA from n = 1000 ABCD sample in the held out n = 1000 ABCD sample. Piloting on the simulated PDS scale

```
#first adding random PDS variable
set.seed(110)
sim_ABCD_data$PDS <- as.integer(rtnorm(n=1000, mean = 3.5, sd = 1.5,
                                       lower = 1, upper = 7))

lsem.MID <- sirt::lsem.estimate(data = sim_ABCD_data, moderator = 'PDS', # moderator variable
                                moderator.grid = seq(1,5,1), # moderator levels, PDS 1 - 5
                                lavmodel = MID_model, # model
                                h = 2, # bandwidth parameter
                                residualize = FALSE, # allow mean level differences
                                meanstructure = TRUE,
                                std.lv=TRUE
                                )
```

```
## ** Fit lavaan model
## |*****|
## |-----|
## ** Parameter summary
```

Summary LSEM

Summarizing output of the `lsem.estimate`

```
summary(lsem.MID)
```

```
## -----------------------------------------------------------------
## Local Structural Equation Model
##
## sirt 3.12-66 (2022-05-16 12:27:54)
## lavaan 0.6-12 (2022-07-04 16:40:02 UTC)
##
## R version 4.2.1 (2022-06-23) x86_64, darwin17.0 | nodename=DN0a241705.SUNet | login=root
##
## Function 'sirt::lsem.estimate', type='LSEM'
##
##
## Call:
```

```
## sirt::lsem.estimate(data = sim_ABCD_data, moderator = "PDS",
##     moderator.grid = seq(1, 5, 1), lavmodel = MID_model, h = 2,
##     residualize = FALSE, meanstructure = TRUE, std.lv = TRUE)
##
## Date of Analysis: 2022-11-17 16:10:50
## Time difference of 4.933373 secs
## Computation Time: 4.933373
##
## Number of observations in datasets = 1000
## Used observations in analysis = 1000
## Used sampling weights: FALSE
## Bandwidth factor = 2
## Bandwidth = 0.648
## Number of focal points for moderator = 5
##
## Used joint estimation: FALSE
## Used sufficient statistics: FALSE
## Used local linear smoothing: FALSE
## Used pseudo weights: FALSE
## Used lavaan package: TRUE
## Used lavaan.survey package: FALSE
##
## Mean structure modelled: TRUE
##
## lavaan Model
##
##
## # Factor loadings
## Approach =~ AWin_v_Neut_L_NAcc_run1  + AWin_v_Neut_R_NAcc_run1  + AWin_v_Neut_R_Insula_run1 +
##             AWin_v_Neut_L_NAcc_run2  + AWin_v_Neut_R_NAcc_run2 + AWin_v_Neut_R_Insula_run2 +
##             BWin_v_Neut_L_NAcc_run1  + BWin_v_Neut_R_NAcc_run1  + BWin_v_Neut_R_Insula_run1 +
##             BWin_v_Neut_L_NAcc_run2  + BWin_v_Neut_R_NAcc_run2  + BWin_v_Neut_R_Insula_run2 +
##             BWin_v_BLose_L_NAcc_run1 + BWin_v_BLose_R_NAcc_run1 +
##             BWin_v_BLose_L_NAcc_run2 + BWin_v_BLose_R_NAcc_run2
##
## Avoid =~    ALose_v_Neut_L_Insula_run1 + ALose_v_Neut_R_Insula_run1 +
##             ALose_v_Neut_L_Insula_run2 + ALose_v_Neut_R_Insula_run2 +
##             BLose_v_Neut_L_Insula_run1 + BLose_v_Neut_R_Insula_run1 +
##             BLose_v_Neut_L_Insula_run2 + BLose_v_Neut_R_Insula_run2 +
##             BLose_v_BWin_L_Insula_run1 + BLose_v_BWin_R_Insula_run1 +
##             BLose_v_BWin_L_Insula_run2 + BLose_v_BWin_R_Insula_run2
##
##
## Parameter Estimate Summary
##
##                                         par parindex     M    SD
## 1                Approach=~AWin_v_Neut_L_NAcc_run1        1  0.082 0.006
## 2                Approach=~AWin_v_Neut_R_NAcc_run1        2  0.086 0.004
## 3              Approach=~AWin_v_Neut_R_Insula_run1        3  0.041 0.004
## 4                Approach=~AWin_v_Neut_L_NAcc_run2        4  0.090 0.002
## 5                Approach=~AWin_v_Neut_R_NAcc_run2        5  0.096 0.006
## 6              Approach=~AWin_v_Neut_R_Insula_run2        6  0.049 0.007
## 7                Approach=~BWin_v_Neut_L_NAcc_run1        7  0.095 0.009
## 8                Approach=~BWin_v_Neut_R_NAcc_run1        8  0.070 0.007
```

```
## 9                    Approach=~BWin_v_Neut_R_Insula_run1        9  0.044 0.005
## 10                     Approach=~BWin_v_Neut_L_NAcc_run2       10  0.083 0.007
## 11                     Approach=~BWin_v_Neut_R_NAcc_run2       11  0.079 0.015
## 12                   Approach=~BWin_v_Neut_R_Insula_run2       12  0.050 0.011
## 13                    Approach=~BWin_v_BLose_L_NAcc_run1       13  0.092 0.006
## 14                    Approach=~BWin_v_BLose_R_NAcc_run1       14  0.089 0.005
## 15                    Approach=~BWin_v_BLose_L_NAcc_run2       15  0.079 0.008
## 16                    Approach=~BWin_v_BLose_R_NAcc_run2       16  0.095 0.004
## 17                     Avoid=~ALose_v_Neut_L_Insula_run1       17  0.087 0.003
## 18                     Avoid=~ALose_v_Neut_R_Insula_run1       18  0.094 0.009
## 19                     Avoid=~ALose_v_Neut_L_Insula_run2       19  0.078 0.008
## 20                     Avoid=~ALose_v_Neut_R_Insula_run2       20  0.090 0.012
## 21                     Avoid=~BLose_v_Neut_L_Insula_run1       21  0.086 0.013
## 22                     Avoid=~BLose_v_Neut_R_Insula_run1       22  0.092 0.006
## 23                     Avoid=~BLose_v_Neut_L_Insula_run2       23  0.082 0.006
## 24                     Avoid=~BLose_v_Neut_R_Insula_run2       24  0.091 0.005
## 25                     Avoid=~BLose_v_BWin_L_Insula_run1       25  0.085 0.016
## 26                     Avoid=~BLose_v_BWin_R_Insula_run1       26  0.040 0.010
## 27                     Avoid=~BLose_v_BWin_L_Insula_run2       27  0.077 0.014
## 28                     Avoid=~BLose_v_BWin_R_Insula_run2       28  0.041 0.004
## 29       AWin_v_Neut_L_NAcc_run1~~AWin_v_Neut_L_NAcc_run1       29  0.028 0.001
## 30       AWin_v_Neut_R_NAcc_run1~~AWin_v_Neut_R_NAcc_run1       30  0.031 0.002
## 31   AWin_v_Neut_R_Insula_run1~~AWin_v_Neut_R_Insula_run1       31  0.029 0.002
## 32       AWin_v_Neut_L_NAcc_run2~~AWin_v_Neut_L_NAcc_run2       32  0.027 0.001
## 33       AWin_v_Neut_R_NAcc_run2~~AWin_v_Neut_R_NAcc_run2       33  0.028 0.002
## 34   AWin_v_Neut_R_Insula_run2~~AWin_v_Neut_R_Insula_run2       34  0.028 0.002
## 35       BWin_v_Neut_L_NAcc_run1~~BWin_v_Neut_L_NAcc_run1       35  0.025 0.001
## 36       BWin_v_Neut_R_NAcc_run1~~BWin_v_Neut_R_NAcc_run1       36  0.029 0.001
## 37   BWin_v_Neut_R_Insula_run1~~BWin_v_Neut_R_Insula_run1       37  0.031 0.001
## 38       BWin_v_Neut_L_NAcc_run2~~BWin_v_Neut_L_NAcc_run2       38  0.030 0.001
## 39       BWin_v_Neut_R_NAcc_run2~~BWin_v_Neut_R_NAcc_run2       39  0.030 0.002
## 40   BWin_v_Neut_R_Insula_run2~~BWin_v_Neut_R_Insula_run2       40  0.031 0.001
## 41     BWin_v_BLose_L_NAcc_run1~~BWin_v_BLose_L_NAcc_run1       41  0.028 0.002
## 42     BWin_v_BLose_R_NAcc_run1~~BWin_v_BLose_R_NAcc_run1       42  0.030 0.003
## 43     BWin_v_BLose_L_NAcc_run2~~BWin_v_BLose_L_NAcc_run2       43  0.028 0.001
## 44     BWin_v_BLose_R_NAcc_run2~~BWin_v_BLose_R_NAcc_run2       44  0.028 0.002
## 45 ALose_v_Neut_L_Insula_run1~~ALose_v_Neut_L_Insula_run1       45  0.030 0.002
## 46 ALose_v_Neut_R_Insula_run1~~ALose_v_Neut_R_Insula_run1       46  0.027 0.002
## 47 ALose_v_Neut_L_Insula_run2~~ALose_v_Neut_L_Insula_run2       47  0.029 0.001
## 48 ALose_v_Neut_R_Insula_run2~~ALose_v_Neut_R_Insula_run2       48  0.029 0.003
## 49 BLose_v_Neut_L_Insula_run1~~BLose_v_Neut_L_Insula_run1       49  0.029 0.001
## 50 BLose_v_Neut_R_Insula_run1~~BLose_v_Neut_R_Insula_run1       50  0.025 0.001
## 51 BLose_v_Neut_L_Insula_run2~~BLose_v_Neut_L_Insula_run2       51  0.030 0.002
## 52 BLose_v_Neut_R_Insula_run2~~BLose_v_Neut_R_Insula_run2       52  0.029 0.002
## 53 BLose_v_BWin_L_Insula_run1~~BLose_v_BWin_L_Insula_run1       53  0.030 0.002
## 54 BLose_v_BWin_R_Insula_run1~~BLose_v_BWin_R_Insula_run1       54  0.031 0.002
## 55 BLose_v_BWin_L_Insula_run2~~BLose_v_BWin_L_Insula_run2       55  0.029 0.002
## 56 BLose_v_BWin_R_Insula_run2~~BLose_v_BWin_R_Insula_run2       56  0.029 0.001
## 57                                     Approach~~Approach       57  1.000 0.000
## 58                                           Avoid~~Avoid       58  1.000 0.000
## 59                                        Approach~~Avoid       59 -0.577 0.037
## 60                              AWin_v_Neut_L_NAcc_run1~1       60 -0.003 0.016
## 61                              AWin_v_Neut_R_NAcc_run1~1       61 -0.004 0.009
## 62                            AWin_v_Neut_R_Insula_run1~1       62  0.007 0.004
```

```
## 63                                      AWin_v_Neut_L_NAcc_run2~1       63 -0.002 0.009
## 64                                      AWin_v_Neut_R_NAcc_run2~1       64  0.003 0.007
## 65                                    AWin_v_Neut_R_Insula_run2~1       65  0.005 0.003
## 66                                      BWin_v_Neut_L_NAcc_run1~1       66 -0.011 0.003
## 67                                      BWin_v_Neut_R_NAcc_run1~1       67 -0.006 0.008
## 68                                    BWin_v_Neut_R_Insula_run1~1       68  0.013 0.004
## 69                                      BWin_v_Neut_L_NAcc_run2~1       69 -0.005 0.003
## 70                                      BWin_v_Neut_R_NAcc_run2~1       70 -0.001 0.007
## 71                                    BWin_v_Neut_R_Insula_run2~1       71 -0.001 0.006
## 72                                     BWin_v_BLose_L_NAcc_run1~1       72 -0.003 0.009
## 73                                     BWin_v_BLose_R_NAcc_run1~1       73 -0.005 0.011
## 74                                     BWin_v_BLose_L_NAcc_run2~1       74 -0.008 0.010
## 75                                     BWin_v_BLose_R_NAcc_run2~1       75 -0.008 0.008
## 76                                   ALose_v_Neut_L_Insula_run1~1       76 -0.005 0.004
## 77                                   ALose_v_Neut_R_Insula_run1~1       77  0.003 0.005
## 78                                   ALose_v_Neut_L_Insula_run2~1       78 -0.001 0.005
## 79                                   ALose_v_Neut_R_Insula_run2~1       79  0.002 0.005
## 80                                   BLose_v_Neut_L_Insula_run1~1       80  0.002 0.009
## 81                                   BLose_v_Neut_R_Insula_run1~1       81  0.014 0.008
## 82                                   BLose_v_Neut_L_Insula_run2~1       82  0.013 0.003
## 83                                   BLose_v_Neut_R_Insula_run2~1       83  0.005 0.006
## 84                                   BLose_v_BWin_L_Insula_run1~1       84  0.008 0.006
## 85                                   BLose_v_BWin_R_Insula_run1~1       85  0.002 0.005
## 86                                   BLose_v_BWin_L_Insula_run2~1       86  0.005 0.004
## 87                                   BLose_v_BWin_R_Insula_run2~1       87 -0.003 0.012
## 88                                                     Approach~1       88  0.000 0.000
## 89                                                        Avoid~1       89  0.000 0.000
## 90                                                          rmsea       90  0.075 0.005
## 91                                                            cfi       91  0.598 0.039
## 92                                                            tli       92  0.564 0.042
## 93                                                            gfi       93  0.860 0.014
## 94                                                           srmr       94  0.061 0.004
##      MAD   Min   Max lin_int lin_slo SD_nonlin
## 1  0.006 0.074 0.090   0.092  -0.003     0.005
## 2  0.003 0.081 0.092   0.087   0.000     0.003
## 3  0.003 0.036 0.047   0.043  -0.001     0.003
## 4  0.002 0.085 0.093   0.095  -0.002     0.001
## 5  0.003 0.086 0.109   0.107  -0.004     0.004
## 6  0.006 0.039 0.061   0.033   0.005     0.003
## 7  0.005 0.071 0.105   0.115  -0.007     0.004
## 8  0.006 0.063 0.080   0.060   0.003     0.005
## 9  0.003 0.034 0.055   0.056  -0.004     0.003
## 10 0.005 0.069 0.090   0.080   0.001     0.006
## 11 0.012 0.056 0.100   0.042   0.013     0.002
## 12 0.009 0.037 0.072   0.074  -0.008     0.006
## 13 0.005 0.076 0.097   0.101  -0.003     0.005
## 14 0.004 0.083 0.099   0.082   0.002     0.004
## 15 0.005 0.061 0.086   0.070   0.003     0.007
## 16 0.003 0.086 0.099   0.100  -0.002     0.003
## 17 0.002 0.082 0.092   0.088   0.000     0.003
## 18 0.008 0.080 0.105   0.072   0.007     0.003
## 19 0.007 0.068 0.088   0.096  -0.006     0.003
## 20 0.009 0.068 0.104   0.079   0.004     0.011
## 21 0.012 0.058 0.097   0.067   0.006     0.011
```

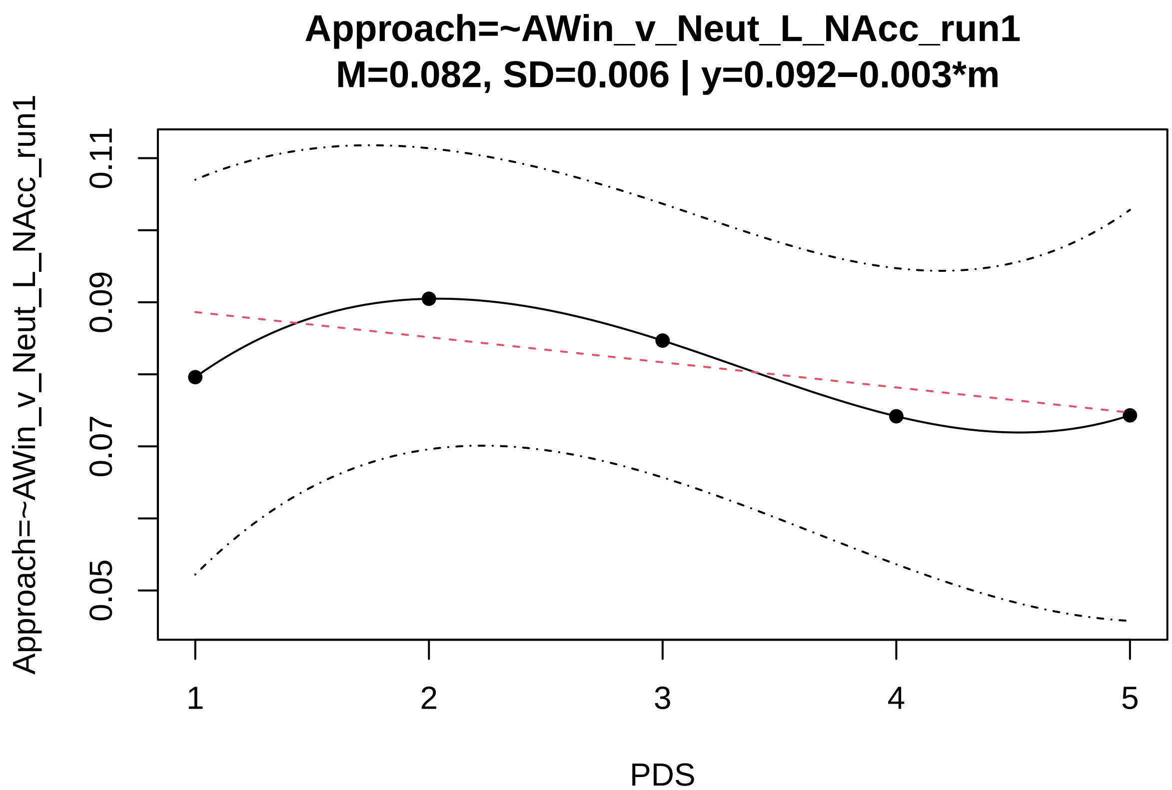
```
## 22 0.005  0.083  0.099   0.103  -0.004      0.004
## 23 0.005  0.070  0.088   0.073   0.003      0.005
## 24 0.004  0.081  0.095   0.089   0.001      0.005
## 25 0.015  0.062  0.104   0.048   0.013      0.007
## 26 0.009  0.028  0.056   0.057  -0.006      0.008
## 27 0.012  0.057  0.097   0.044   0.011      0.006
## 28 0.004  0.037  0.050   0.049  -0.003      0.003
## 29 0.001  0.026  0.029   0.027   0.000      0.001
## 30 0.002  0.028  0.033   0.026   0.002      0.001
## 31 0.002  0.026  0.032   0.029   0.000      0.002
## 32 0.001  0.025  0.027   0.026   0.000      0.001
## 33 0.001  0.027  0.033   0.025   0.001      0.001
## 34 0.002  0.025  0.032   0.034  -0.002      0.001
## 35 0.001  0.023  0.028   0.022   0.001      0.001
## 36 0.001  0.027  0.030   0.029   0.000      0.001
## 37 0.001  0.029  0.032   0.033  -0.001      0.001
## 38 0.001  0.028  0.032   0.033  -0.001      0.001
## 39 0.001  0.024  0.032   0.032  -0.001      0.002
## 40 0.001  0.029  0.033   0.031   0.000      0.001
## 41 0.001  0.025  0.033   0.032  -0.001      0.001
## 42 0.002  0.021  0.032   0.035  -0.002      0.002
## 43 0.001  0.027  0.029   0.030  -0.001      0.001
## 44 0.002  0.024  0.029   0.023   0.002      0.001
## 45 0.002  0.026  0.032   0.032  -0.001      0.002
## 46 0.002  0.024  0.030   0.032  -0.002      0.001
## 47 0.001  0.028  0.032   0.028   0.000      0.001
## 48 0.003  0.026  0.036   0.036  -0.002      0.002
## 49 0.001  0.025  0.029   0.031  -0.001      0.001
## 50 0.001  0.024  0.026   0.026   0.000      0.001
## 51 0.002  0.027  0.033   0.031   0.000      0.002
## 52 0.001  0.026  0.031   0.024   0.002      0.000
## 53 0.002  0.027  0.031   0.032  -0.001      0.002
## 54 0.002  0.028  0.035   0.036  -0.002      0.001
## 55 0.001  0.025  0.031   0.032  -0.001      0.001
## 56 0.001  0.028  0.030   0.030   0.000      0.001
## 57 0.000  1.000  1.000   1.000   0.000      0.000
## 58 0.000  1.000  1.000   1.000   0.000      0.000
## 59 0.033 -0.621 -0.517  -0.517  -0.020      0.028
## 60 0.014 -0.017  0.028   0.034  -0.013      0.006
## 61 0.008 -0.020  0.007   0.003  -0.002      0.008
## 62 0.003 -0.001  0.010   0.005   0.001      0.004
## 63 0.007 -0.012  0.013   0.018  -0.007      0.002
## 64 0.005 -0.011  0.011   0.016  -0.004      0.004
## 65 0.003  0.001  0.009   0.010  -0.002      0.003
## 66 0.003 -0.014 -0.005  -0.013   0.001      0.003
## 67 0.006 -0.011  0.014   0.007  -0.004      0.006
## 68 0.003  0.008  0.017   0.008   0.002      0.003
## 69 0.003 -0.009  0.000  -0.007   0.001      0.003
## 70 0.005 -0.006  0.018   0.010  -0.004      0.006
## 71 0.004 -0.010  0.015   0.011  -0.004      0.004
## 72 0.007 -0.010  0.016   0.005  -0.003      0.008
## 73 0.010 -0.019  0.010   0.018  -0.008      0.007
## 74 0.010 -0.017  0.011   0.007  -0.005      0.008
## 75 0.008 -0.017  0.005   0.003  -0.004      0.007
```

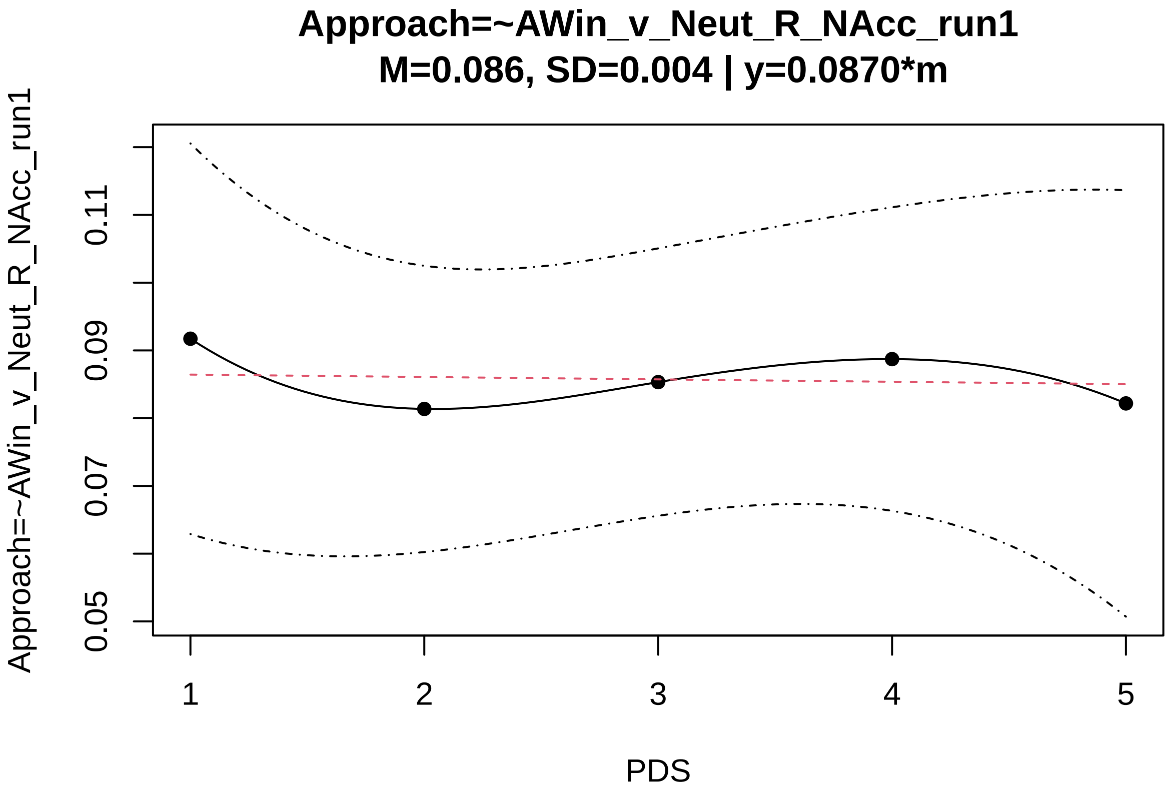
```
## 76 0.004 -0.009  0.003  -0.008   0.001     0.004
## 77 0.004 -0.008  0.009   0.015  -0.004     0.001
## 78 0.004 -0.008  0.007  -0.005   0.001     0.005
## 79 0.003 -0.010  0.007   0.009  -0.002     0.004
## 80 0.007 -0.006  0.024   0.012  -0.003     0.008
## 81 0.007  0.003  0.025  -0.006   0.007     0.003
## 82 0.003  0.009  0.016   0.011   0.001     0.003
## 83 0.006 -0.003  0.011  -0.002   0.002     0.005
## 84 0.005  0.001  0.016   0.018  -0.003     0.004
## 85 0.003 -0.011  0.006   0.008  -0.002     0.004
## 86 0.003 -0.002  0.009   0.007  -0.001     0.004
## 87 0.010 -0.030  0.005  -0.028   0.008     0.007
## 88 0.000  0.000  0.000   0.000   0.000     0.000
## 89 0.000  0.000  0.000   0.000   0.000     0.000
## 90 0.005  0.071  0.086   0.071   0.001     0.005
## 91 0.030  0.527  0.635   0.594   0.001     0.038
## 92 0.033  0.488  0.605   0.560   0.001     0.042
## 93 0.013  0.830  0.872   0.872  -0.004     0.013
## 94 0.003  0.058  0.069   0.060   0.000     0.004
##
## Distribution of Moderator: Density and Effective Sample Size
##
## M=3.083 | SD=1.289
##
##   moderator   wgt    Neff
## 1         1 0.125 188.687
## 2         2 0.227 345.959
## 3         3 0.307 433.724
## 4         4 0.243 353.279
## 5         5 0.098 179.382
##
##    variable       M      SD     min     max
## 1 moderator   3.083   1.289   1.000   6.000
## 2       wgt   0.200   0.087   0.098   0.307
## 3      Neff 300.206 111.548 179.382 433.724
```

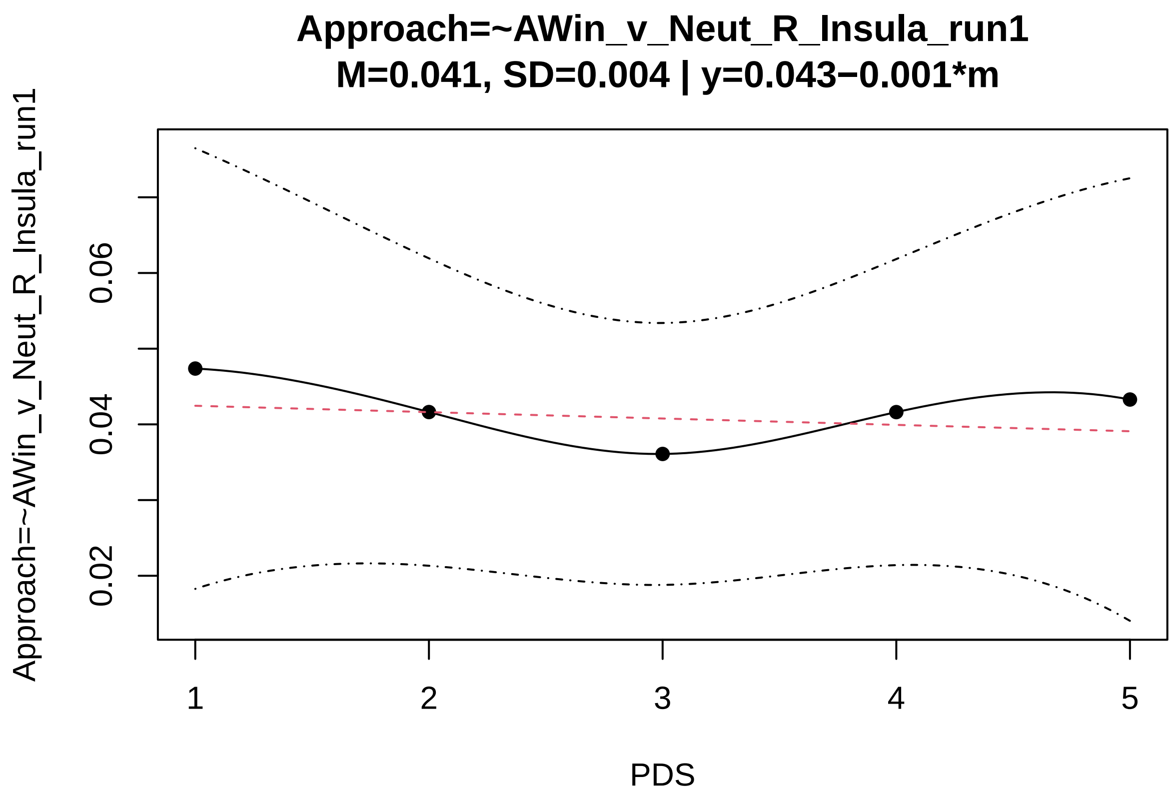
Plot LSEM

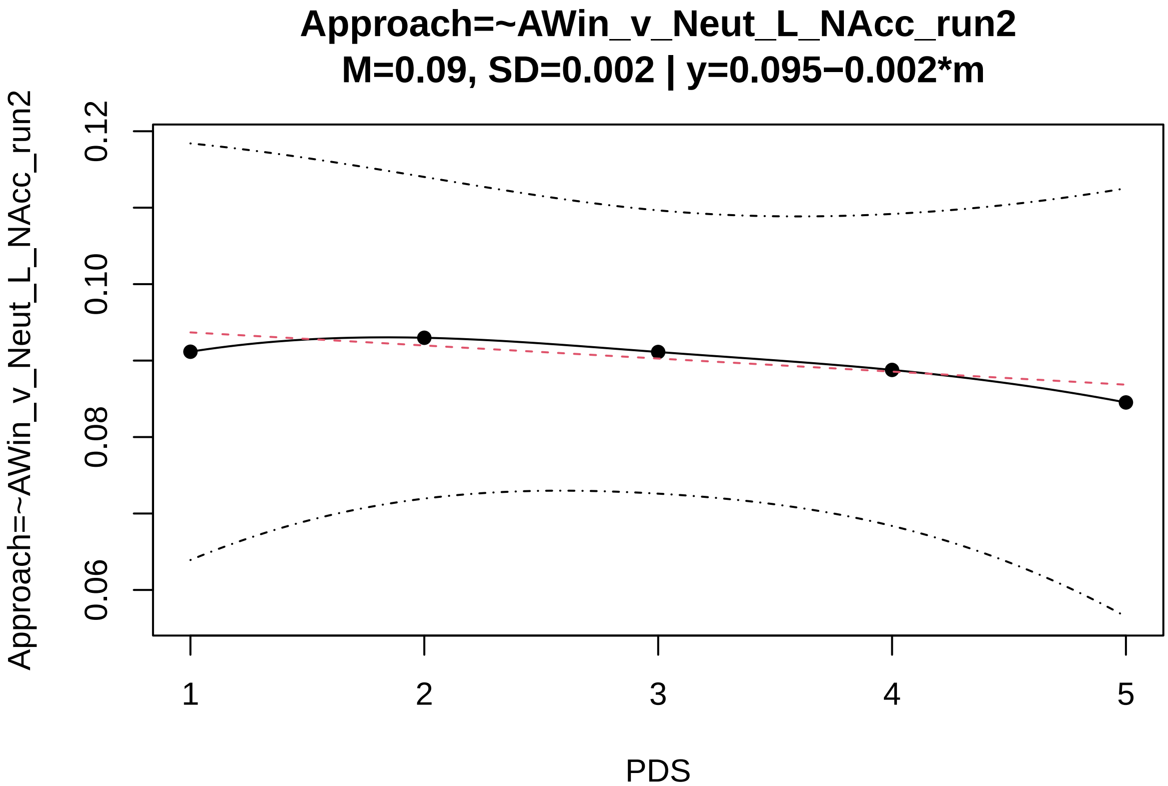
Plotting the `lsem.estimate` for the first 20 indexes.

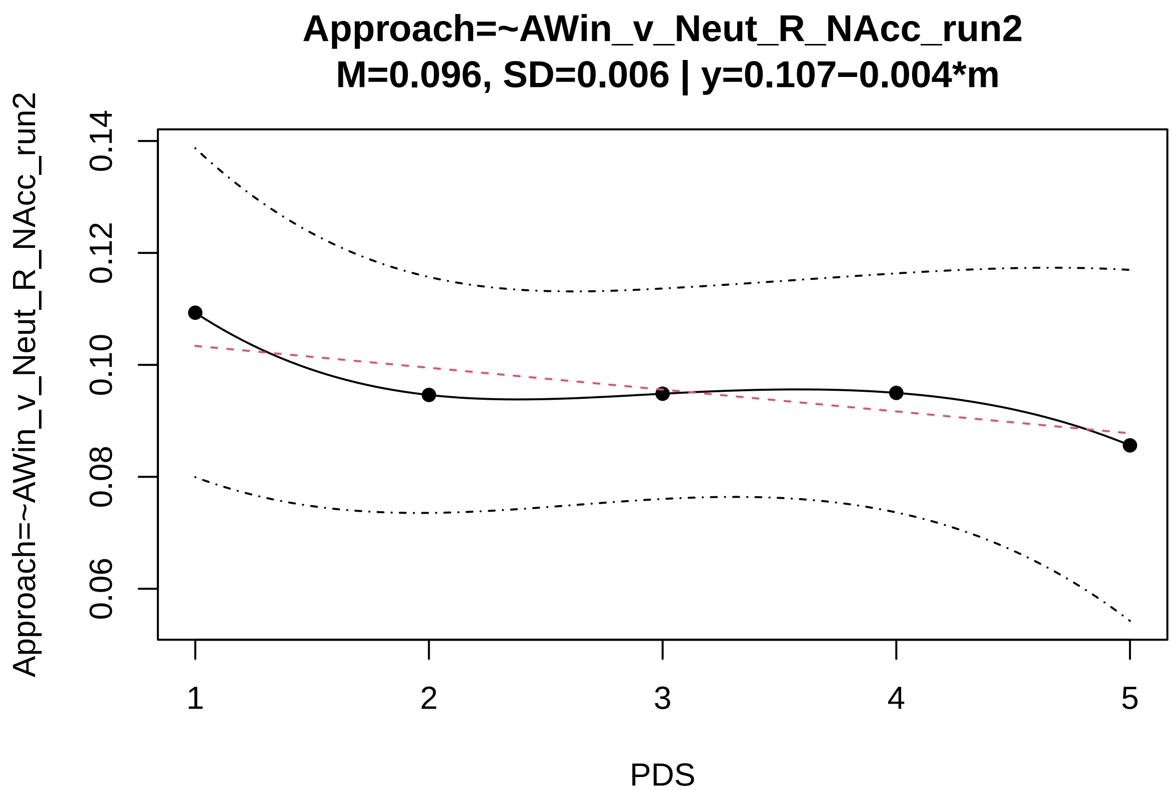
```
plot(lsem.MID, parindex=1:20)
```

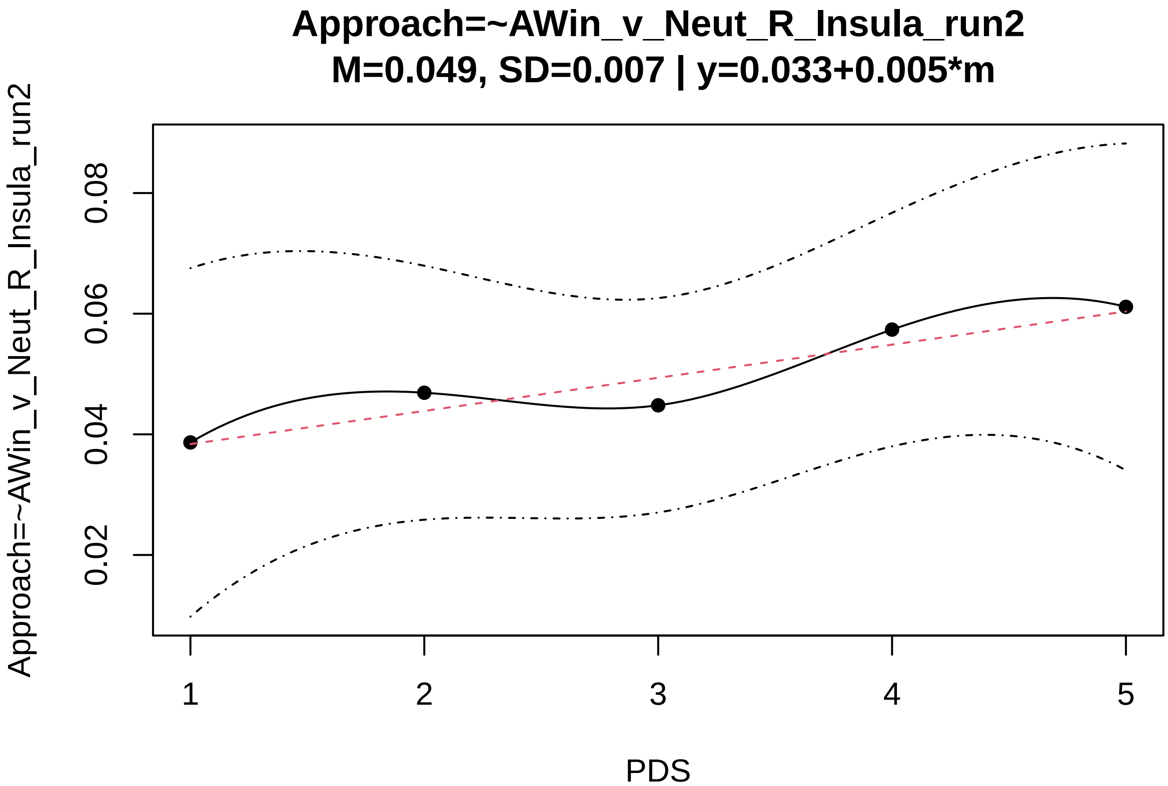

Approach=~AWin_v_Neut_L_NAcc_run1
M=0.082, SD=0.006 | y=0.092−0.003*m
Approach=~AWin_v_Neut_L_NAcc_run1
0.05
0.07
0.09
0.11
1
2
3
4
5
PDS


Approach=~AWin_v_Neut_R_NAcc_run1
M=0.086, SD=0.004 | y=0.0870*m
Approach=~AWin_v_Neut_R_NAcc_run1
0.05
0.07
0.09
0.11
1
2
3
4
5
PDS

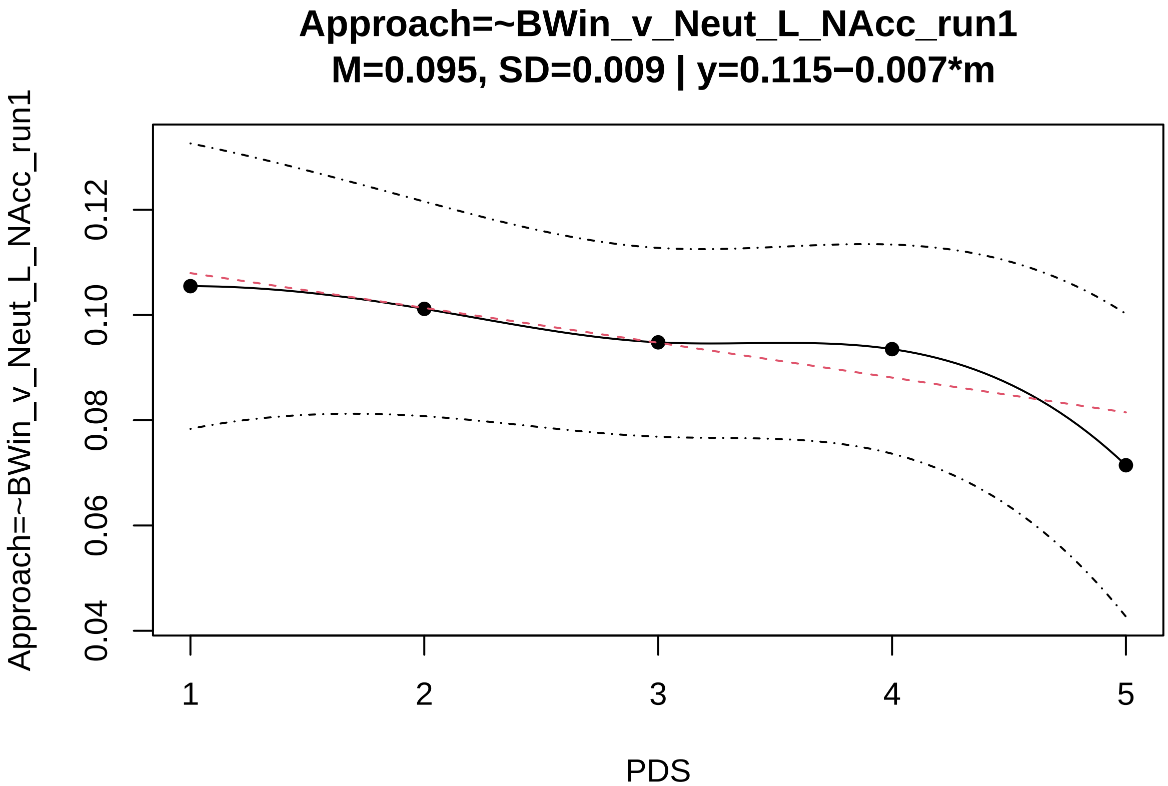

Approach=~AWin_v_Neut_R_Insula_run1
M=0.041, SD=0.004 | y=0.043−0.001*m
Approach=~AWin_v_Neut_R_Insula_run1
0.02
0.04
0.06
1
2
3
4
5
PDS

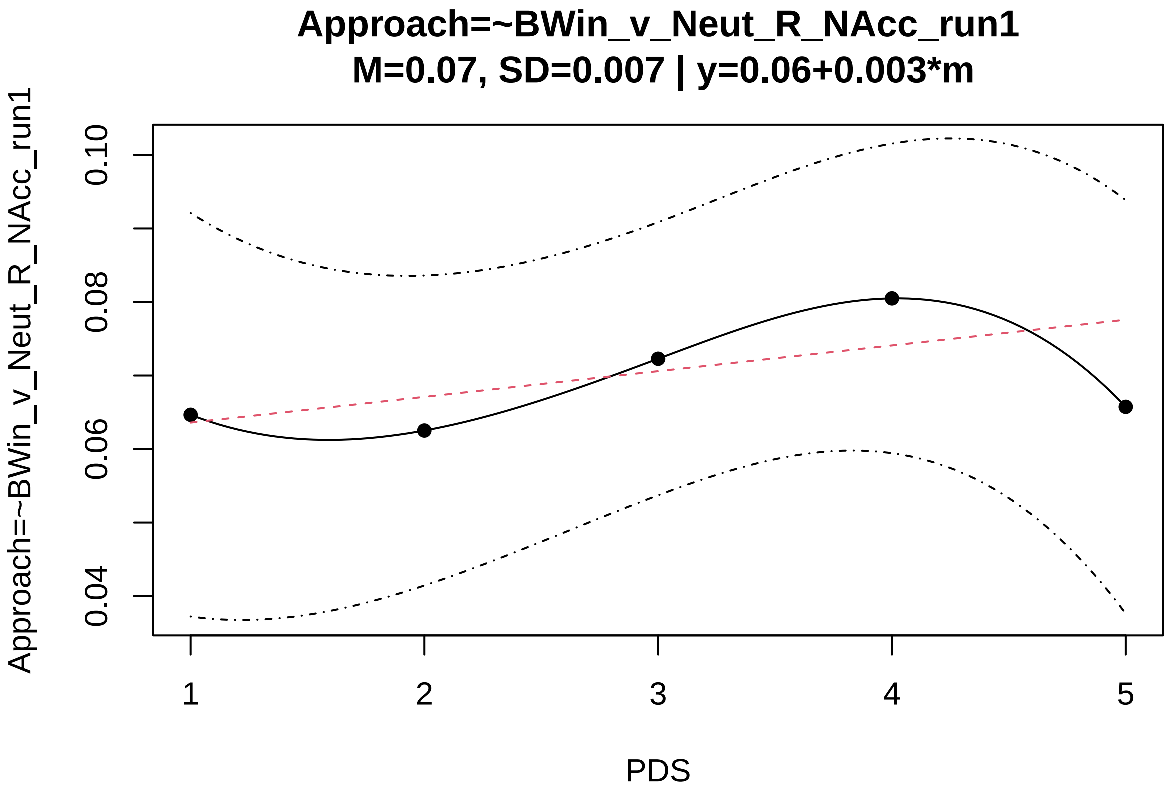

Approach=~AWin_v_Neut_L_NAcc_run2
M=0.09, SD=0.002 | y=0.095−0.002*m
Approach=~AWin_v_Neut_L_NAcc_run2
0.06
0.08
0.10
0.12
1
2
3
4
5
PDS

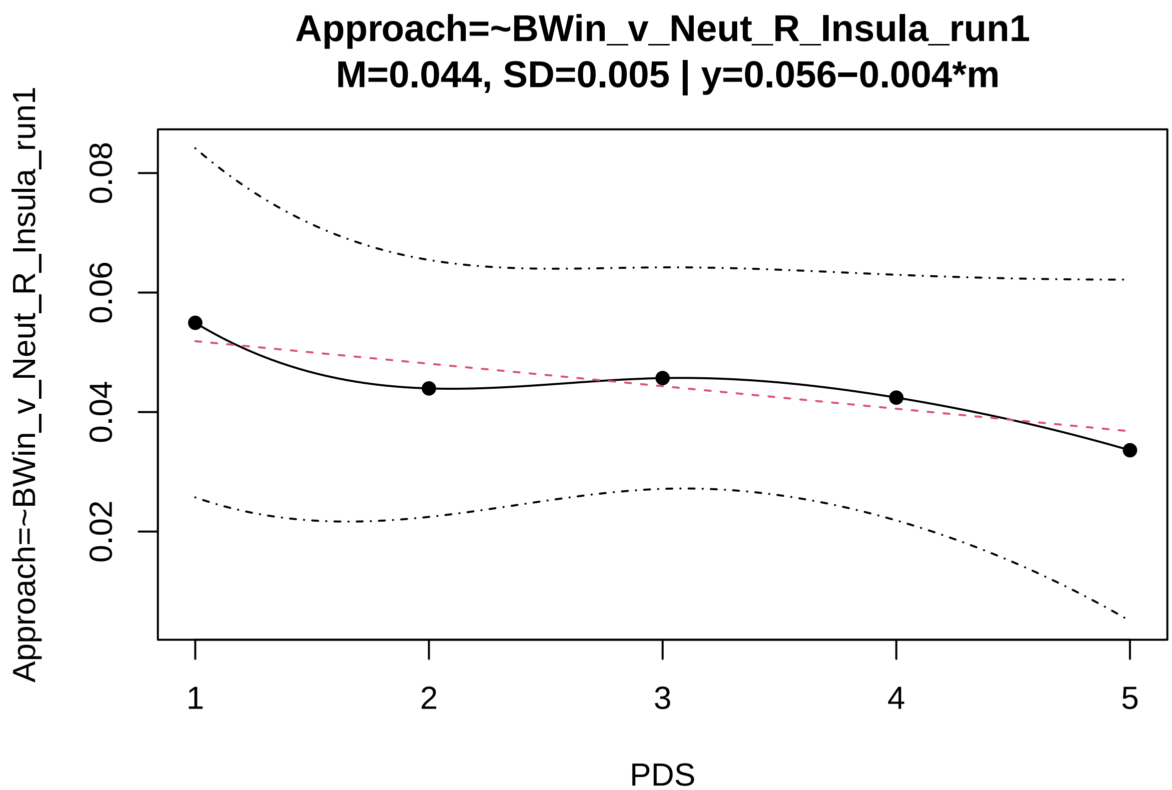

Approach=~AWin_v_Neut_R_NAcc_run2
M=0.096, SD=0.006 | y=0.107−0.004*m
Approach=~AWin_v_Neut_R_NAcc_run2
0.06
0.08
0.10
0.12
0.14
1
2
3
4
5
PDS

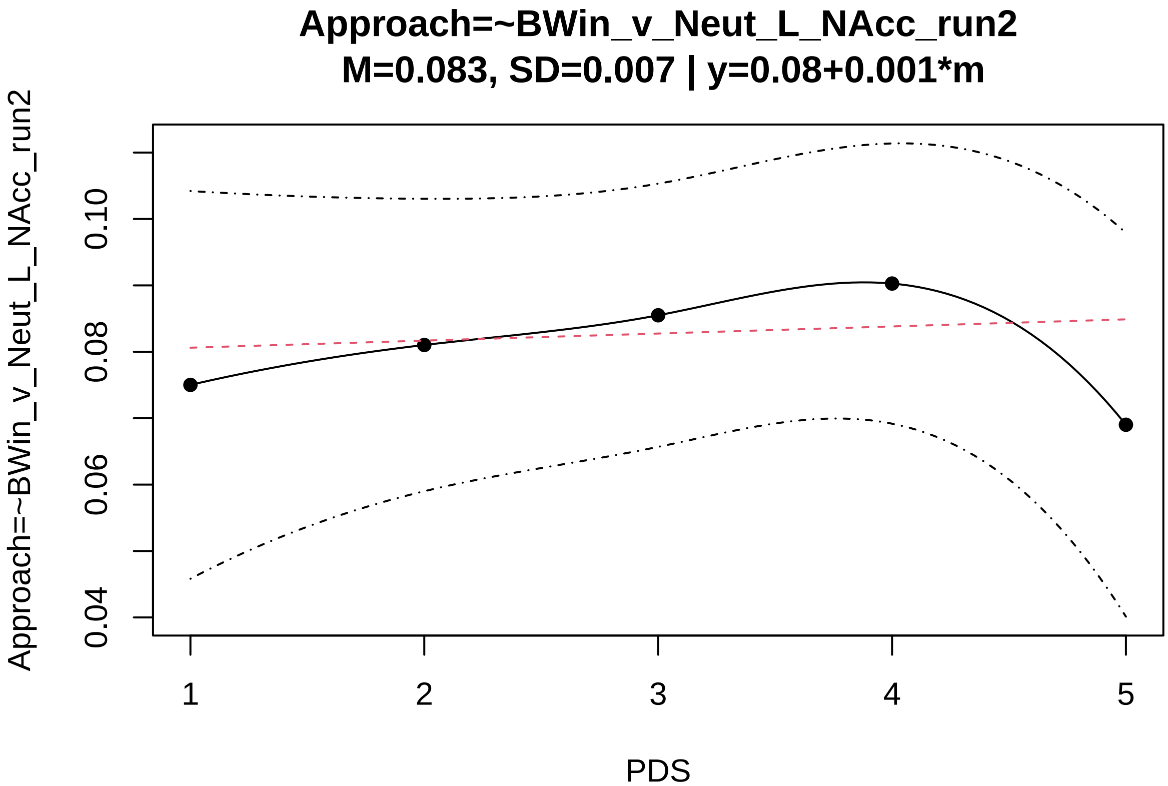


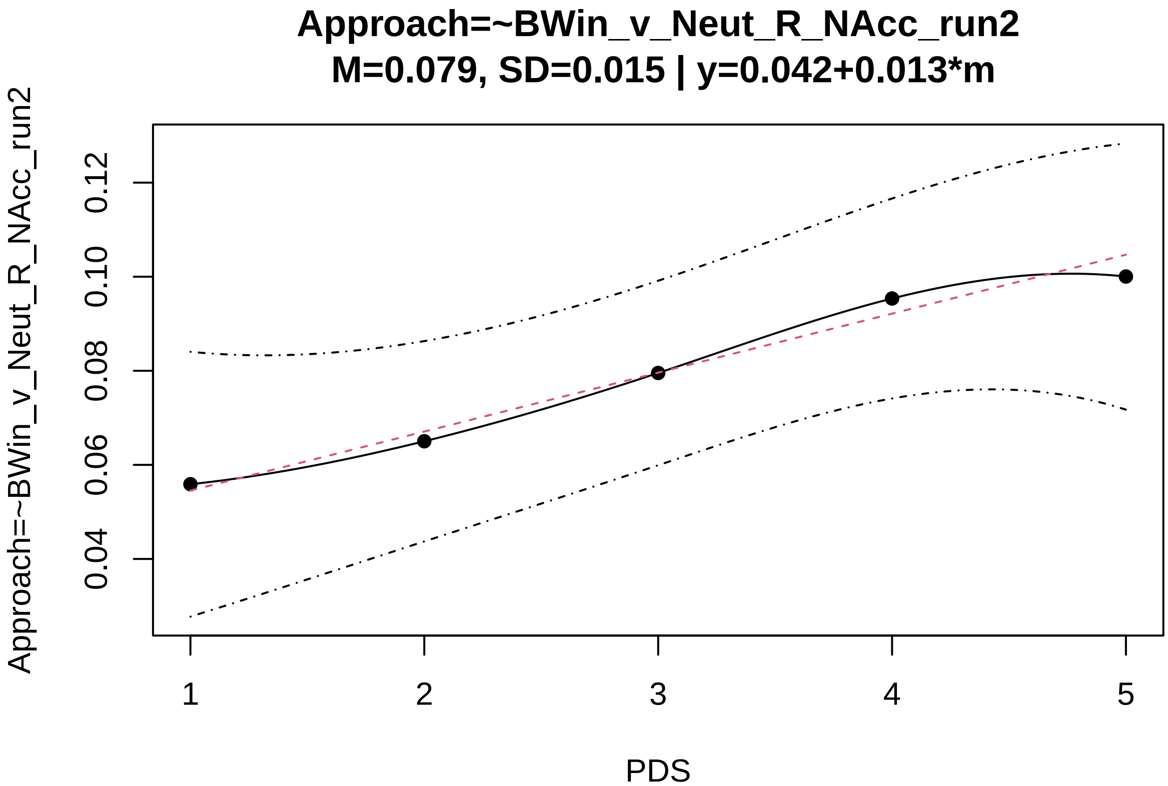
Approach=~AWin_v_Neut_R_Insula_run2
M=0.049, SD=0.007 | y=0.033+0.005*m
Approach=~AWin_v_Neut_R_Insula_run2
0.02
0.04
0.06
0.08
1
2
3
4
5
PDS


Approach=~BWin_v_Neut_L_NAcc_run1
M=0.095, SD=0.009 | y=0.115−0.007*m
Approach=~BWin_v_Neut_L_NAcc_run1
0.04
0.06
0.08
0.10
0.12
1
2
3
4
5
PDS

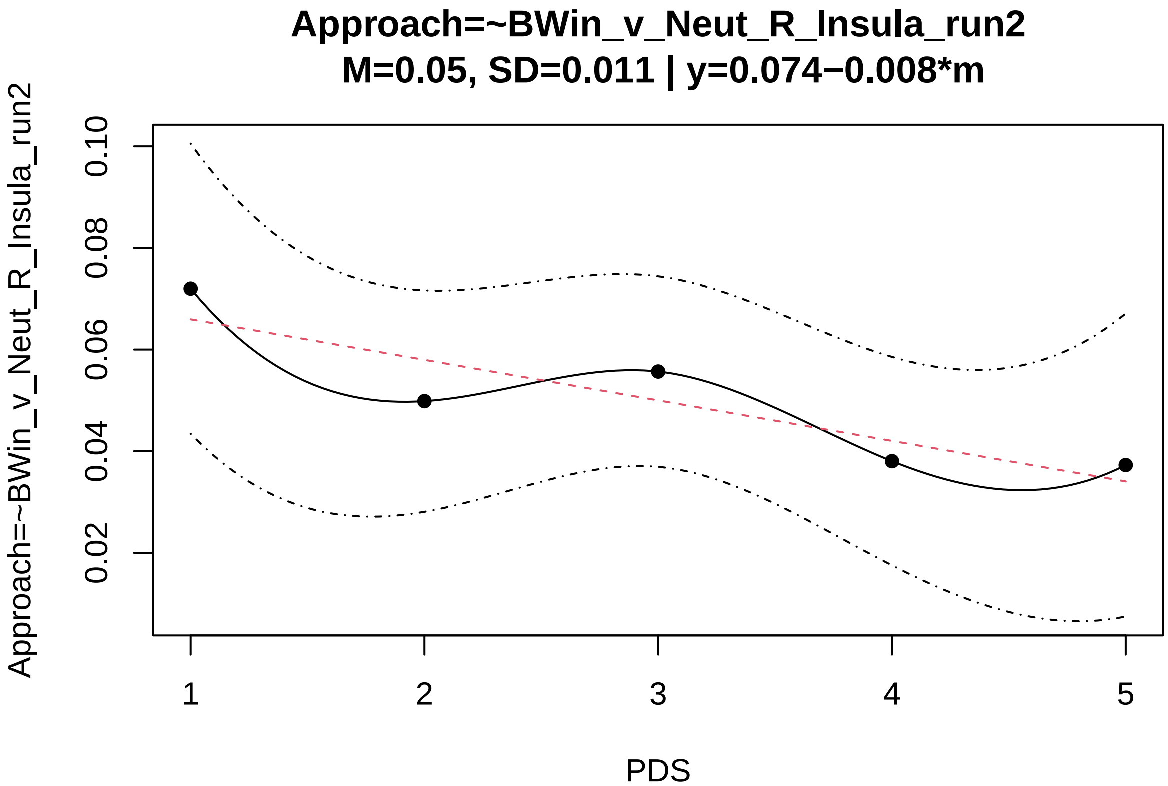

Approach=~BWin_v_Neut_R_NAcc_run1
M=0.07, SD=0.007 | y=0.06+0.003*m
Approach=~BWin_v_Neut_R_NAcc_run1
0.04
0.06
0.08
0.10
1
2
3
4
5
PDS

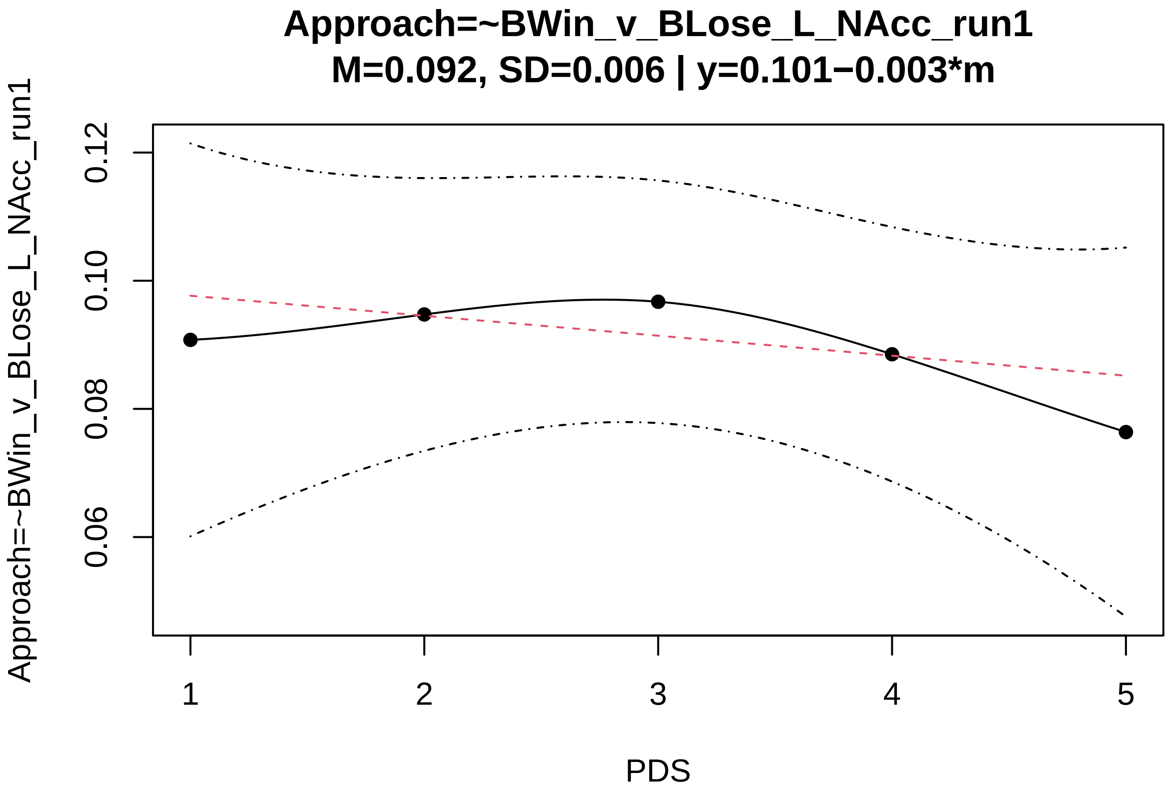

Approach=~BWin_v_Neut_R_Insula_run1
M=0.044, SD=0.005 | y=0.056−0.004*m
Approach=~BWin_v_Neut_R_Insula_run1
0.02
0.04
0.06
0.08
1
2
3
4
5
PDS


Approach=~BWin_v_Neut_L_NAcc_run2
M=0.083, SD=0.007 | y=0.08+0.001*m
Approach=~BWin_v_Neut_L_NAcc_run2
0.04
0.06
0.08
0.10
1
2
3
4
5
PDS

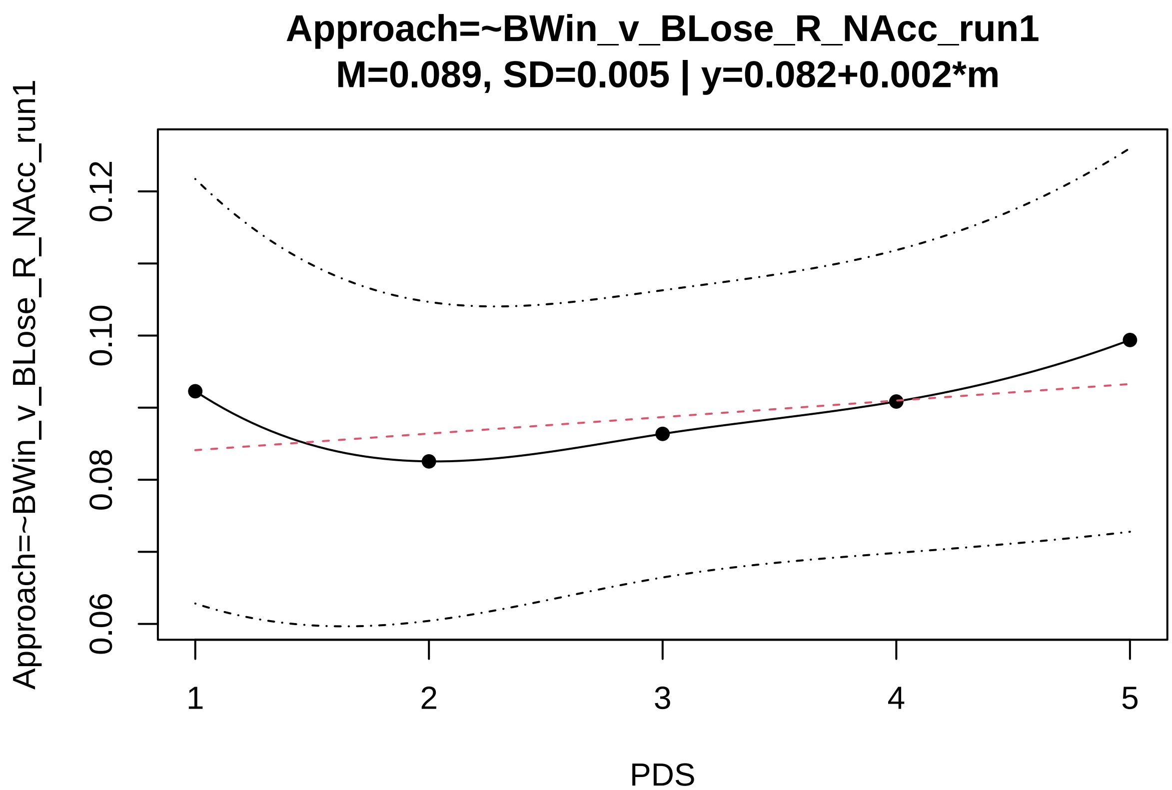


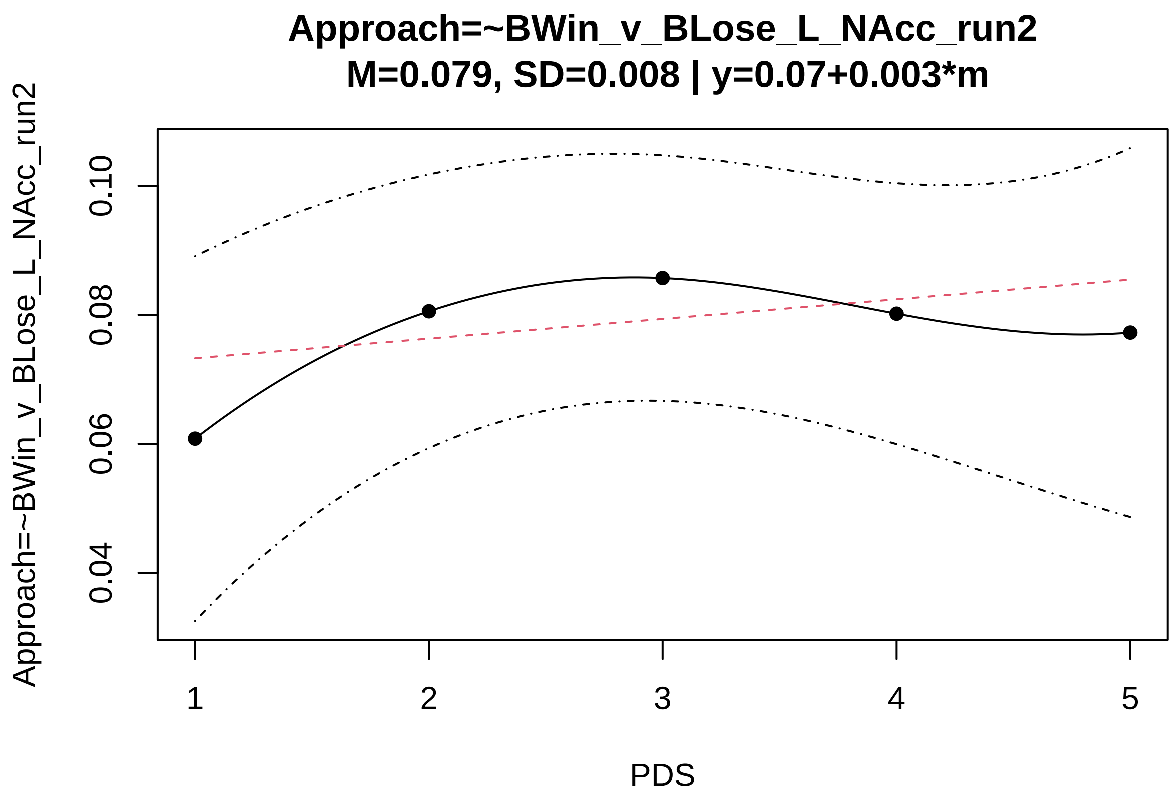
Approach=~BWin_v_Neut_R_NAcc_run2
M=0.079, SD=0.015 | y=0.042+0.013*m
Approach=~BWin_v_Neut_R_NAcc_run2
0.04
0.06
0.08
0.10
0.12
1
2
3
4
5
PDS

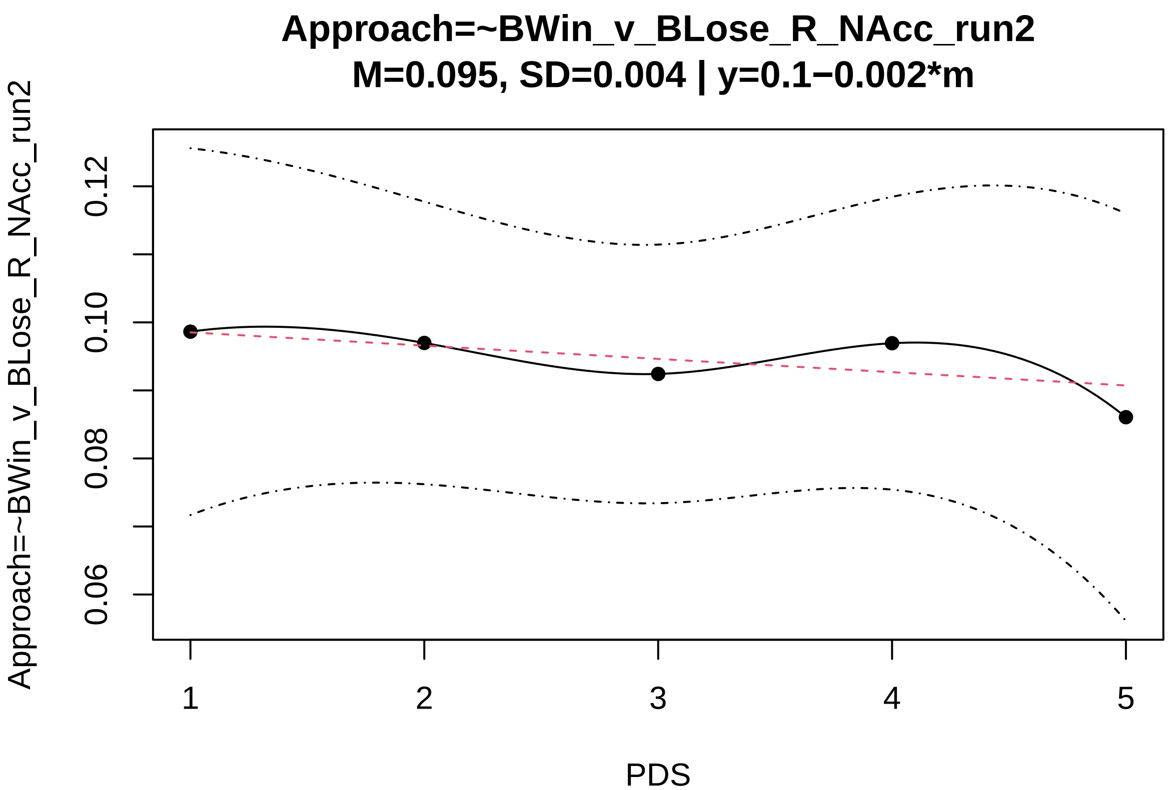

Approach=~BWin_v_Neut_R_Insula_run2
M=0.05, SD=0.011 | y=0.074−0.008*m
Approach=~BWin_v_Neut_R_Insula_run2
0.02
0.04
0.06
0.08
0.10
1
2
3
4
5
PDS



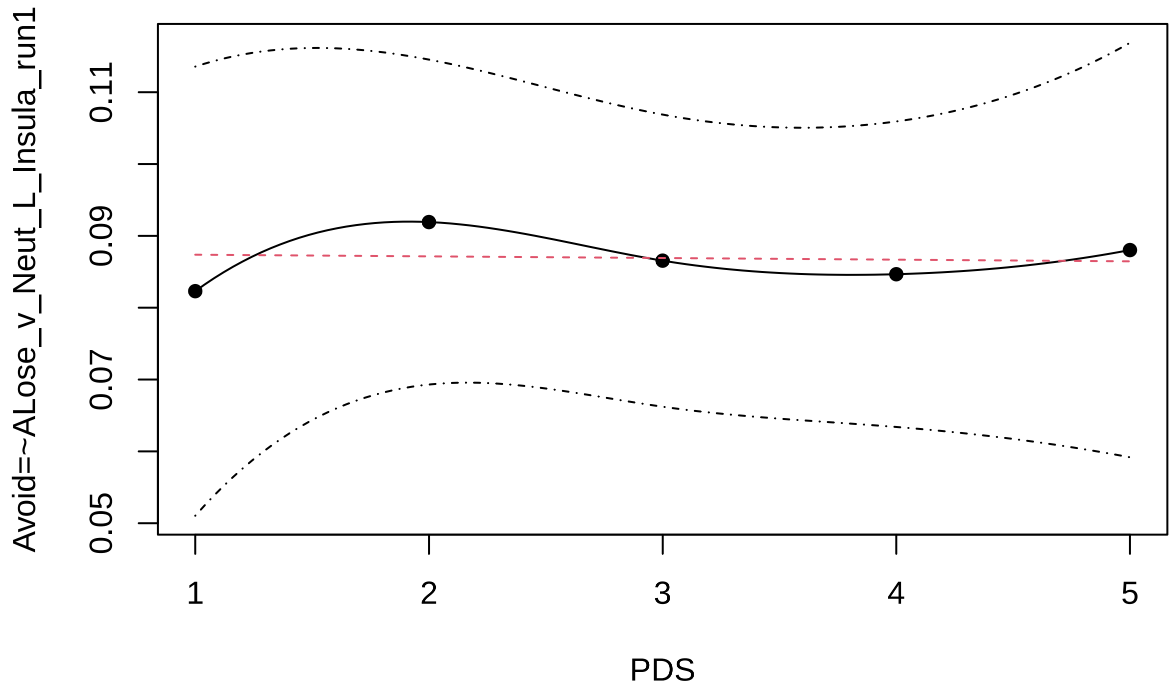
Approach=~BWin_v_BLose_L_NAcc_run1
M=0.092, SD=0.006 | y=0.101−0.003*m
Approach=~BWin_v_BLose_L_NAcc_run1
0.06
0.08
0.10
0.12
1
2
3
4
5
PDS


Approach=~BWin_v_BLose_R_NAcc_run1
M=0.089, SD=0.005 | y=0.082+0.002*m
Approach=~BWin_v_BLose_R_NAcc_run1
0.06
0.08
0.10
0.12
1
2
3
4
5
PDS

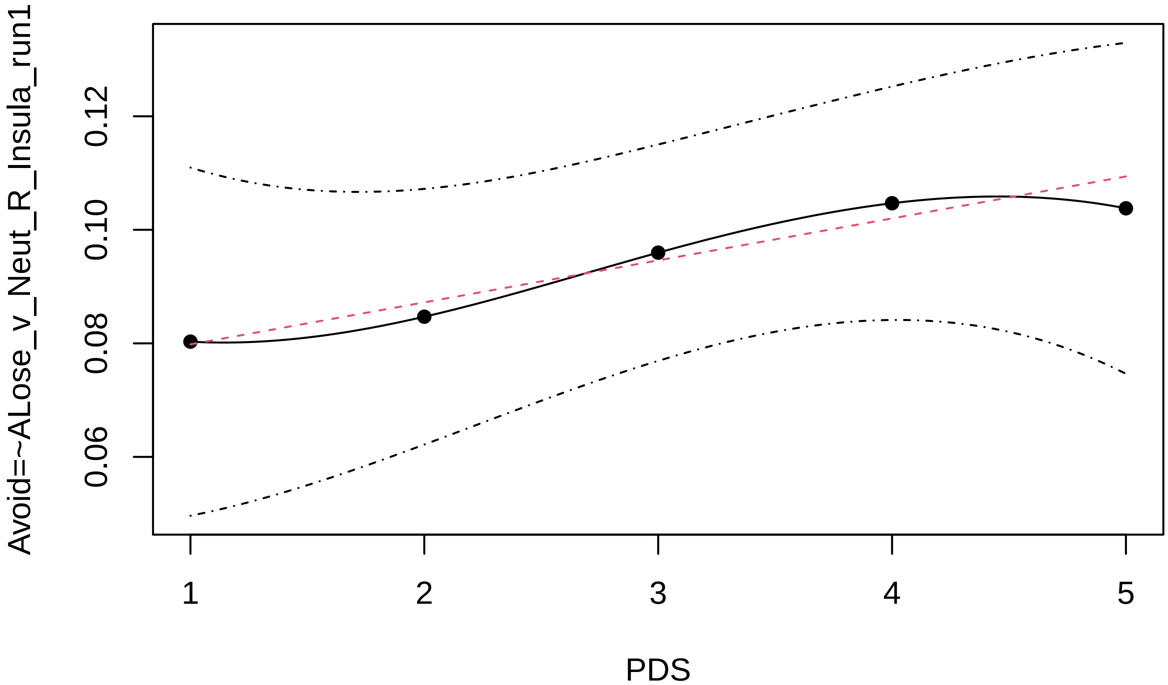

Approach=~BWin_v_BLose_L_NAcc_run2
M=0.079, SD=0.008 | y=0.07+0.003*m
Approach=~BWin_v_BLose_L_NAcc_run2
0.04
0.06
0.08
0.10
1
2
3
4
5
PDS

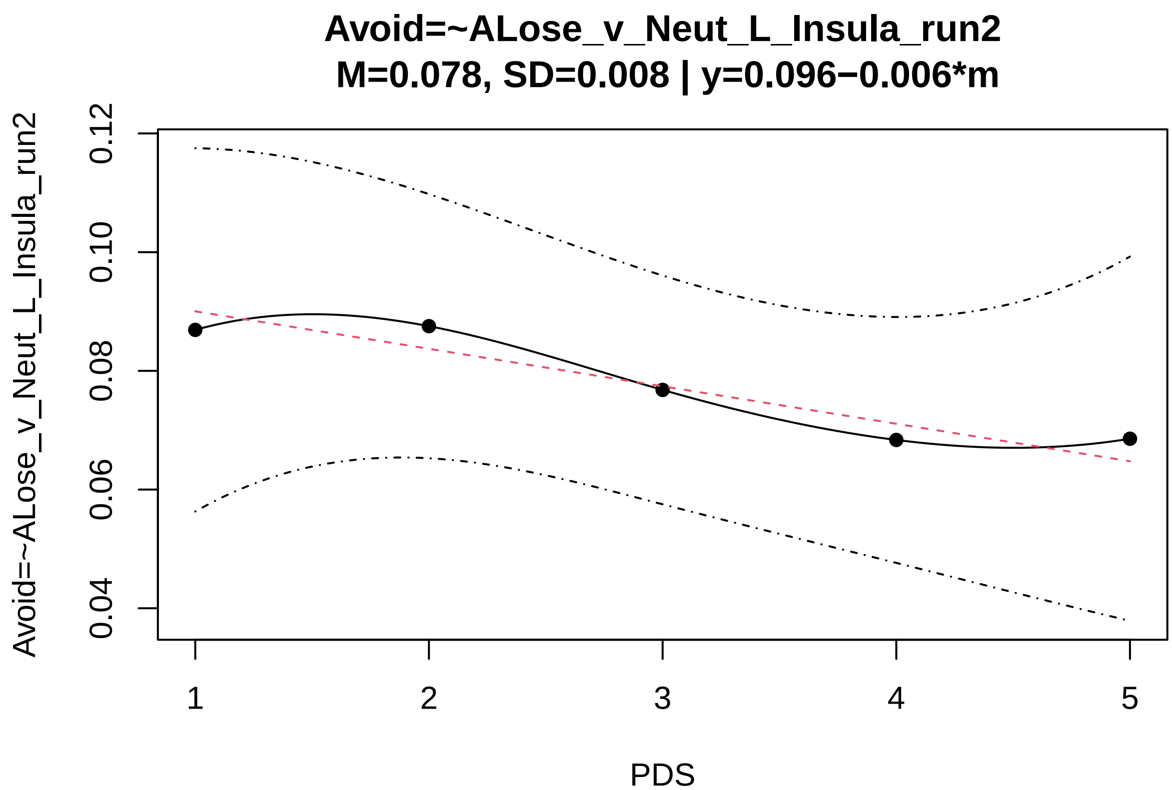

Approach=~BWin_v_BLose_R_NAcc_run2
M=0.095, SD=0.004 | y=0.1−0.002*m
Approach=~BWin_v_BLose_R_NAcc_run2
0.06
0.08
0.10
0.12
1
2
3
4
5
PDS

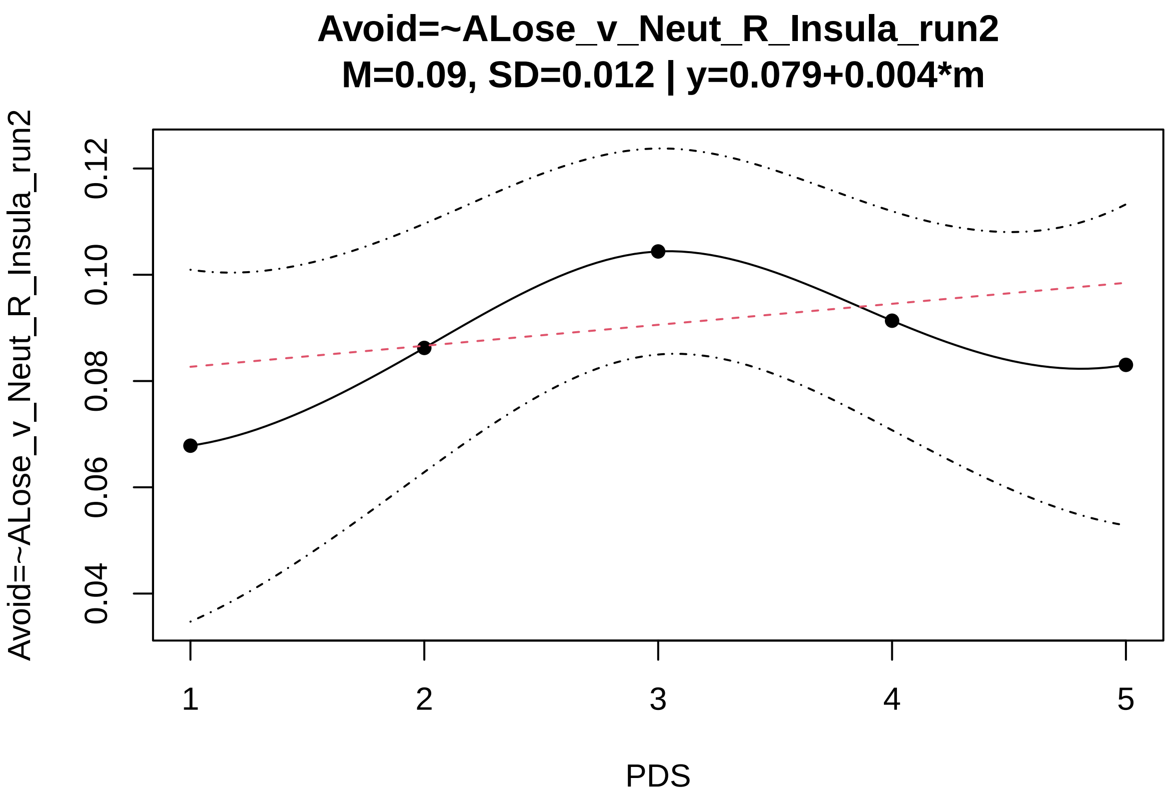
Avoid=~ALose_v_Neut_L_Insula_run1
M=0.087, SD=0.003 | y=0.0880*m



Avoid=~ALose_v_Neut_R_Insula_run1
M=0.094, SD=0.009 | y=0.072+0.007*m




Avoid=~ALose_v_Neut_L_Insula_run2
M=0.078, SD=0.008 | y=0.096−0.006*m
Avoid=~ALose_v_Neut_L_Insula_run2
0.04
0.06
0.08
0.10
0.12
1
2
3
4
5
PDS



Permutation Test LSEM

Running permutation test of LSEM model. In this case, using 10 permutation to save on time. In future iterations, permutations will be 1000.

```
lsem.permuted <- sirt::lsem.permutationTest(lsem.object = lsem.MID,
                                            B = 10, # permutations
                                            residualize = FALSE)
```

```
## Permutation test LSEM
## 1  2  3  4  5  6  7  8  9  10
```

```
summary(lsem.permuted) # examine results
```

```
## -----------------------------------------------------------------
## Permutation Test for Local Structural Equation Model
##
## sirt 3.12-66 (2022-05-16 12:27:54)
## lavaan 0.6-12 (2022-07-04 16:40:02 UTC)
##
## Function 'sirt::lsem.permutationTest'
##
##
## Call:
## sirt::lsem.permutationTest(lsem.object = lsem.MID, B = 10, residualize = FALSE)
##
## Date of Analysis: 2022-11-17 16:11:41
```

```
## Time difference of 50.20561 secs
## Computation Time: 50.20561
##
## Number of permutations = 10
## Percentage of non-converged datasets = 0
## Number of observations=1000
## Bandwidth factor=2
## Bandwidth=0.648
## Number of focal points for moderator=5
##
## lavaan Model
##
##
## # Factor loadings
## Approach =~ AWin_v_Neut_L_NAcc_run1  + AWin_v_Neut_R_NAcc_run1  + AWin_v_Neut_R_Insula_run1 +
##             AWin_v_Neut_L_NAcc_run2  + AWin_v_Neut_R_NAcc_run2 + AWin_v_Neut_R_Insula_run2 +
##             BWin_v_Neut_L_NAcc_run1  + BWin_v_Neut_R_NAcc_run1  + BWin_v_Neut_R_Insula_run1 +
##             BWin_v_Neut_L_NAcc_run2  + BWin_v_Neut_R_NAcc_run2  + BWin_v_Neut_R_Insula_run2 +
##             BWin_v_BLose_L_NAcc_run1 + BWin_v_BLose_R_NAcc_run1 +
##             BWin_v_BLose_L_NAcc_run2 + BWin_v_BLose_R_NAcc_run2
##
## Avoid =~    ALose_v_Neut_L_Insula_run1 + ALose_v_Neut_R_Insula_run1 +
##             ALose_v_Neut_L_Insula_run2 + ALose_v_Neut_R_Insula_run2 +
##             BLose_v_Neut_L_Insula_run1 + BLose_v_Neut_R_Insula_run1 +
##             BLose_v_Neut_L_Insula_run2 + BLose_v_Neut_R_Insula_run2 +
##             BLose_v_BWin_L_Insula_run1 + BLose_v_BWin_R_Insula_run1 +
##             BLose_v_BWin_L_Insula_run2 + BLose_v_BWin_R_Insula_run2
##
##
## Global Test Statistics
##
##                                              par     M    SD SD_p
## 1                     Approach=~AWin_v_Neut_L_NAcc_run1  0.082 0.006  0.6
## 2                     Approach=~AWin_v_Neut_R_NAcc_run1  0.086 0.004  0.9
## 3                   Approach=~AWin_v_Neut_R_Insula_run1  0.041 0.004  0.8
## 4                     Approach=~AWin_v_Neut_L_NAcc_run2  0.090 0.002  1.0
## 5                     Approach=~AWin_v_Neut_R_NAcc_run2  0.096 0.006  0.8
## 6                   Approach=~AWin_v_Neut_R_Insula_run2  0.049 0.007  0.3
## 7                     Approach=~BWin_v_Neut_L_NAcc_run1  0.095 0.009  0.2
## 8                     Approach=~BWin_v_Neut_R_NAcc_run1  0.070 0.007  0.2
## 9                   Approach=~BWin_v_Neut_R_Insula_run1  0.044 0.005  0.7
## 10                    Approach=~BWin_v_Neut_L_NAcc_run2  0.083 0.007  0.7
## 11                    Approach=~BWin_v_Neut_R_NAcc_run2  0.079 0.015  0.0
## 12                  Approach=~BWin_v_Neut_R_Insula_run2  0.050 0.011  0.1
## 13                   Approach=~BWin_v_BLose_L_NAcc_run1  0.092 0.006  0.6
## 14                   Approach=~BWin_v_BLose_R_NAcc_run1  0.089 0.005  0.8
## 15                   Approach=~BWin_v_BLose_L_NAcc_run2  0.079 0.008  0.5
## 16                   Approach=~BWin_v_BLose_R_NAcc_run2  0.095 0.004  1.0
## 17                    Avoid=~ALose_v_Neut_L_Insula_run1  0.087 0.003  1.0
## 18                    Avoid=~ALose_v_Neut_R_Insula_run1  0.094 0.009  0.4
## 19                    Avoid=~ALose_v_Neut_L_Insula_run2  0.078 0.008  0.4
## 20                    Avoid=~ALose_v_Neut_R_Insula_run2  0.090 0.012  0.2
## 21                    Avoid=~BLose_v_Neut_L_Insula_run1  0.086 0.013  0.0
## 22                    Avoid=~BLose_v_Neut_R_Insula_run1  0.092 0.006  0.7
```

```
## 23                        Avoid=~BLose_v_Neut_L_Insula_run2  0.082 0.006   0.7
## 24                        Avoid=~BLose_v_Neut_R_Insula_run2  0.091 0.005   0.8
## 25                        Avoid=~BLose_v_BWin_L_Insula_run1  0.085 0.016   0.1
## 26                        Avoid=~BLose_v_BWin_R_Insula_run1  0.040 0.010   0.1
## 27                        Avoid=~BLose_v_BWin_L_Insula_run2  0.077 0.014   0.0
## 28                        Avoid=~BLose_v_BWin_R_Insula_run2  0.041 0.004   0.7
## 29       AWin_v_Neut_L_NAcc_run1~~AWin_v_Neut_L_NAcc_run1  0.028 0.001   0.6
## 30       AWin_v_Neut_R_NAcc_run1~~AWin_v_Neut_R_NAcc_run1  0.031 0.002   0.2
## 31   AWin_v_Neut_R_Insula_run1~~AWin_v_Neut_R_Insula_run1  0.029 0.002   0.0
## 32       AWin_v_Neut_L_NAcc_run2~~AWin_v_Neut_L_NAcc_run2  0.027 0.001   0.8
## 33       AWin_v_Neut_R_NAcc_run2~~AWin_v_Neut_R_NAcc_run2  0.028 0.002   0.5
## 34   AWin_v_Neut_R_Insula_run2~~AWin_v_Neut_R_Insula_run2  0.028 0.002   0.3
## 35       BWin_v_Neut_L_NAcc_run1~~BWin_v_Neut_L_NAcc_run1  0.025 0.001   0.6
## 36       BWin_v_Neut_R_NAcc_run1~~BWin_v_Neut_R_NAcc_run1  0.029 0.001   0.8
## 37   BWin_v_Neut_R_Insula_run1~~BWin_v_Neut_R_Insula_run1  0.031 0.001   0.8
## 38       BWin_v_Neut_L_NAcc_run2~~BWin_v_Neut_L_NAcc_run2  0.030 0.001   0.7
## 39       BWin_v_Neut_R_NAcc_run2~~BWin_v_Neut_R_NAcc_run2  0.030 0.002   0.1
## 40   BWin_v_Neut_R_Insula_run2~~BWin_v_Neut_R_Insula_run2  0.031 0.001   0.8
## 41     BWin_v_BLose_L_NAcc_run1~~BWin_v_BLose_L_NAcc_run1  0.028 0.002   0.0
## 42     BWin_v_BLose_R_NAcc_run1~~BWin_v_BLose_R_NAcc_run1  0.030 0.003   0.1
## 43     BWin_v_BLose_L_NAcc_run2~~BWin_v_BLose_L_NAcc_run2  0.028 0.001   0.7
## 44     BWin_v_BLose_R_NAcc_run2~~BWin_v_BLose_R_NAcc_run2  0.028 0.002   0.0
## 45 ALose_v_Neut_L_Insula_run1~~ALose_v_Neut_L_Insula_run1  0.030 0.002   0.2
## 46 ALose_v_Neut_R_Insula_run1~~ALose_v_Neut_R_Insula_run1  0.027 0.002   0.1
## 47 ALose_v_Neut_L_Insula_run2~~ALose_v_Neut_L_Insula_run2  0.029 0.001   0.7
## 48 ALose_v_Neut_R_Insula_run2~~ALose_v_Neut_R_Insula_run2  0.029 0.003   0.0
## 49 BLose_v_Neut_L_Insula_run1~~BLose_v_Neut_L_Insula_run1  0.029 0.001   0.7
## 50 BLose_v_Neut_R_Insula_run1~~BLose_v_Neut_R_Insula_run1  0.025 0.001   0.9
## 51 BLose_v_Neut_L_Insula_run2~~BLose_v_Neut_L_Insula_run2  0.030 0.002   0.1
## 52 BLose_v_Neut_R_Insula_run2~~BLose_v_Neut_R_Insula_run2  0.029 0.002   0.5
## 53 BLose_v_BWin_L_Insula_run1~~BLose_v_BWin_L_Insula_run1  0.030 0.002   0.7
## 54 BLose_v_BWin_R_Insula_run1~~BLose_v_BWin_R_Insula_run1  0.031 0.002   0.0
## 55 BLose_v_BWin_L_Insula_run2~~BLose_v_BWin_L_Insula_run2  0.029 0.002   0.3
## 56 BLose_v_BWin_R_Insula_run2~~BLose_v_BWin_R_Insula_run2  0.029 0.001   1.0
## 57                                     Approach~~Approach  1.000 0.000   1.0
## 58                                           Avoid~~Avoid  1.000 0.000   1.0
## 59                                        Approach~~Avoid -0.577 0.037   0.6
## 60                              AWin_v_Neut_L_NAcc_run1~1 -0.003 0.016   0.0
## 61                              AWin_v_Neut_R_NAcc_run1~1 -0.004 0.009   0.2
## 62                            AWin_v_Neut_R_Insula_run1~1  0.007 0.004   0.6
## 63                              AWin_v_Neut_L_NAcc_run2~1 -0.002 0.009   0.1
## 64                              AWin_v_Neut_R_NAcc_run2~1  0.003 0.007   0.5
## 65                            AWin_v_Neut_R_Insula_run2~1  0.005 0.003   0.8
## 66                              BWin_v_Neut_L_NAcc_run1~1 -0.011 0.003   0.9
## 67                              BWin_v_Neut_R_NAcc_run1~1 -0.006 0.008   0.3
## 68                            BWin_v_Neut_R_Insula_run1~1  0.013 0.004   0.8
## 69                              BWin_v_Neut_L_NAcc_run2~1 -0.005 0.003   0.8
## 70                              BWin_v_Neut_R_NAcc_run2~1 -0.001 0.007   0.4
## 71                            BWin_v_Neut_R_Insula_run2~1 -0.001 0.006   0.5
## 72                             BWin_v_BLose_L_NAcc_run1~1 -0.003 0.009   0.3
## 73                             BWin_v_BLose_R_NAcc_run1~1 -0.005 0.011   0.1
## 74                             BWin_v_BLose_L_NAcc_run2~1 -0.008 0.010   0.0
## 75                             BWin_v_BLose_R_NAcc_run2~1 -0.008 0.008   0.1
## 76                           ALose_v_Neut_L_Insula_run1~1 -0.005 0.004   0.8
```

```
## 77                             ALose_v_Neut_R_Insula_run1~1  0.003 0.005  0.7
## 78                             ALose_v_Neut_L_Insula_run2~1 -0.001 0.005  0.8
## 79                             ALose_v_Neut_R_Insula_run2~1  0.002 0.005  0.8
## 80                             BLose_v_Neut_L_Insula_run1~1  0.002 0.009  0.0
## 81                             BLose_v_Neut_R_Insula_run1~1  0.014 0.008  0.5
## 82                             BLose_v_Neut_L_Insula_run2~1  0.013 0.003  0.9
## 83                             BLose_v_Neut_R_Insula_run2~1  0.005 0.006  0.4
## 84                             BLose_v_BWin_L_Insula_run1~1  0.008 0.006  0.3
## 85                             BLose_v_BWin_R_Insula_run1~1  0.002 0.005  0.7
## 86                             BLose_v_BWin_L_Insula_run2~1  0.005 0.004  0.7
## 87                             BLose_v_BWin_R_Insula_run2~1 -0.003 0.012  0.0
## 88                                               Approach~1  0.000 0.000  1.0
## 89                                                  Avoid~1  0.000 0.000  1.0
## 90                                                    rmsea  0.075 0.005  0.7
## 91                                                      cfi  0.598 0.039  0.4
## 92                                                      tli  0.564 0.042  0.4
## 93                                                      gfi  0.860 0.014  0.6
## 94                                                     srmr  0.061 0.004  0.5
##      MAD MAD_p lin_slo lin_slo_p
## 1  0.006   0.5  -0.003       0.2
## 2  0.003   0.9   0.000       0.4
## 3  0.003   0.8  -0.001       0.8
## 4  0.002   1.0  -0.002       0.6
## 5  0.003   0.8  -0.004       0.6
## 6  0.006   0.2   0.005       0.2
## 7  0.005   0.4  -0.007       0.0
## 8  0.006   0.2   0.003       0.2
## 9  0.003   0.9  -0.004       0.8
## 10 0.005   0.8   0.001       1.0
## 11 0.012   0.0   0.013       0.0
## 12 0.009   0.1  -0.008       0.2
## 13 0.005   0.5  -0.003       0.4
## 14 0.004   0.8   0.002       0.8
## 15 0.005   0.5   0.003       0.6
## 16 0.003   1.0  -0.002       1.0
## 17 0.002   1.0   0.000       0.8
## 18 0.008   0.5   0.007       0.4
## 19 0.007   0.2  -0.006       0.2
## 20 0.009   0.4   0.004       0.8
## 21 0.012   0.0   0.006       0.2
## 22 0.005   0.6  -0.004       0.4
## 23 0.005   0.6   0.003       0.6
## 24 0.004   0.8   0.001       0.8
## 25 0.015   0.0   0.013       0.0
## 26 0.009   0.1  -0.006       0.0
## 27 0.012   0.1   0.011       0.0
## 28 0.004   0.7  -0.003       0.4
## 29 0.001   0.6   0.000       0.8
## 30 0.002   0.1   0.002       0.0
## 31 0.002   0.0   0.000       0.8
## 32 0.001   0.8   0.000       0.8
## 33 0.001   0.6   0.001       0.0
## 34 0.002   0.1  -0.002       0.0
## 35 0.001   0.9   0.001       0.4
```

```
## 36 0.001   0.6   0.000        0.8
## 37 0.001   0.6  -0.001        0.2
## 38 0.001   0.7  -0.001        0.8
## 39 0.001   0.3  -0.001        0.2
## 40 0.001   0.6   0.000        1.0
## 41 0.001   0.3  -0.001        0.2
## 42 0.002   0.1  -0.002        0.0
## 43 0.001   0.6  -0.001        0.4
## 44 0.002   0.0   0.002        0.0
## 45 0.002   0.1  -0.001        0.8
## 46 0.002   0.1  -0.002        0.2
## 47 0.001   0.7   0.000        1.0
## 48 0.003   0.0  -0.002        0.0
## 49 0.001   0.7  -0.001        0.4
## 50 0.001   0.8   0.000        1.0
## 51 0.002   0.1   0.000        0.8
## 52 0.001   0.7   0.002        0.2
## 53 0.002   0.6  -0.001        0.4
## 54 0.002   0.0  -0.002        0.0
## 55 0.001   0.3  -0.001        0.0
## 56 0.001   0.9   0.000        1.0
## 57 0.000   1.0   0.000        1.0
## 58 0.000   1.0   0.000        1.0
## 59 0.033   0.6  -0.020        0.2
## 60 0.014   0.0  -0.013        0.0
## 61 0.008   0.2  -0.002        0.2
## 62 0.003   0.6   0.001        1.0
## 63 0.007   0.0  -0.007        0.0
## 64 0.005   0.4  -0.004        0.2
## 65 0.003   0.5  -0.002        0.0
## 66 0.003   0.9   0.001        0.8
## 67 0.006   0.3  -0.004        0.6
## 68 0.003   0.7   0.002        0.2
## 69 0.003   0.8   0.001        0.8
## 70 0.005   0.6  -0.004        0.0
## 71 0.004   0.6  -0.004        0.2
## 72 0.007   0.3  -0.003        0.6
## 73 0.010   0.0  -0.008        0.0
## 74 0.010   0.0  -0.005        0.2
## 75 0.008   0.0  -0.004        0.0
## 76 0.004   0.8   0.001        0.6
## 77 0.004   0.6  -0.004        0.0
## 78 0.004   0.7   0.001        0.8
## 79 0.003   0.9  -0.002        0.8
## 80 0.007   0.2  -0.003        0.4
## 81 0.007   0.3   0.007        0.0
## 82 0.003   0.8   0.001        0.6
## 83 0.006   0.1   0.002        0.2
## 84 0.005   0.3  -0.003        0.2
## 85 0.003   0.8  -0.002        0.6
## 86 0.003   0.7  -0.001        1.0
## 87 0.010   0.0   0.008        0.0
## 88 0.000   1.0   0.000        1.0
## 89 0.000   1.0   0.000        1.0
```

```
## 90 0.005   0.2   0.001       0.0
## 91 0.030   0.5   0.001       0.8
## 92 0.033   0.5   0.001       0.8
## 93 0.013   0.3  -0.004       0.0
## 94 0.003   0.4   0.000       0.6
##
## Pointwise Test Statistics
##
##                                                 par parindex moderator
## 1                         Approach=~AWin_v_Neut_L_NAcc_run1        1         1
## 2                         Approach=~AWin_v_Neut_L_NAcc_run1        1         2
## 3                         Approach=~AWin_v_Neut_L_NAcc_run1        1         3
## 4                         Approach=~AWin_v_Neut_L_NAcc_run1        1         4
## 5                         Approach=~AWin_v_Neut_L_NAcc_run1        1         5
## 6                         Approach=~AWin_v_Neut_R_NAcc_run1        2         1
## 7                         Approach=~AWin_v_Neut_R_NAcc_run1        2         2
## 8                         Approach=~AWin_v_Neut_R_NAcc_run1        2         3
## 9                         Approach=~AWin_v_Neut_R_NAcc_run1        2         4
## 10                        Approach=~AWin_v_Neut_R_NAcc_run1        2         5
## 11                      Approach=~AWin_v_Neut_R_Insula_run1        3         1
## 12                      Approach=~AWin_v_Neut_R_Insula_run1        3         2
## 13                      Approach=~AWin_v_Neut_R_Insula_run1        3         3
## 14                      Approach=~AWin_v_Neut_R_Insula_run1        3         4
## 15                      Approach=~AWin_v_Neut_R_Insula_run1        3         5
## 16                        Approach=~AWin_v_Neut_L_NAcc_run2        4         1
## 17                        Approach=~AWin_v_Neut_L_NAcc_run2        4         2
## 18                        Approach=~AWin_v_Neut_L_NAcc_run2        4         3
## 19                        Approach=~AWin_v_Neut_L_NAcc_run2        4         4
## 20                        Approach=~AWin_v_Neut_L_NAcc_run2        4         5
## 21                        Approach=~AWin_v_Neut_R_NAcc_run2        5         1
## 22                        Approach=~AWin_v_Neut_R_NAcc_run2        5         2
## 23                        Approach=~AWin_v_Neut_R_NAcc_run2        5         3
## 24                        Approach=~AWin_v_Neut_R_NAcc_run2        5         4
## 25                        Approach=~AWin_v_Neut_R_NAcc_run2        5         5
## 26                      Approach=~AWin_v_Neut_R_Insula_run2        6         1
## 27                      Approach=~AWin_v_Neut_R_Insula_run2        6         2
## 28                      Approach=~AWin_v_Neut_R_Insula_run2        6         3
## 29                      Approach=~AWin_v_Neut_R_Insula_run2        6         4
## 30                      Approach=~AWin_v_Neut_R_Insula_run2        6         5
## 31                        Approach=~BWin_v_Neut_L_NAcc_run1        7         1
## 32                        Approach=~BWin_v_Neut_L_NAcc_run1        7         2
## 33                        Approach=~BWin_v_Neut_L_NAcc_run1        7         3
## 34                        Approach=~BWin_v_Neut_L_NAcc_run1        7         4
## 35                        Approach=~BWin_v_Neut_L_NAcc_run1        7         5
## 36                        Approach=~BWin_v_Neut_R_NAcc_run1        8         1
## 37                        Approach=~BWin_v_Neut_R_NAcc_run1        8         2
## 38                        Approach=~BWin_v_Neut_R_NAcc_run1        8         3
## 39                        Approach=~BWin_v_Neut_R_NAcc_run1        8         4
## 40                        Approach=~BWin_v_Neut_R_NAcc_run1        8         5
## 41                      Approach=~BWin_v_Neut_R_Insula_run1        9         1
## 42                      Approach=~BWin_v_Neut_R_Insula_run1        9         2
## 43                      Approach=~BWin_v_Neut_R_Insula_run1        9         3
## 44                      Approach=~BWin_v_Neut_R_Insula_run1        9         4
## 45                      Approach=~BWin_v_Neut_R_Insula_run1        9         5
```

```
## 46                        Approach=~BWin_v_Neut_L_NAcc_run2      10          1
## 47                        Approach=~BWin_v_Neut_L_NAcc_run2      10          2
## 48                        Approach=~BWin_v_Neut_L_NAcc_run2      10          3
## 49                        Approach=~BWin_v_Neut_L_NAcc_run2      10          4
## 50                        Approach=~BWin_v_Neut_L_NAcc_run2      10          5
## 51                        Approach=~BWin_v_Neut_R_NAcc_run2      11          1
## 52                        Approach=~BWin_v_Neut_R_NAcc_run2      11          2
## 53                        Approach=~BWin_v_Neut_R_NAcc_run2      11          3
## 54                        Approach=~BWin_v_Neut_R_NAcc_run2      11          4
## 55                        Approach=~BWin_v_Neut_R_NAcc_run2      11          5
## 56                      Approach=~BWin_v_Neut_R_Insula_run2      12          1
## 57                      Approach=~BWin_v_Neut_R_Insula_run2      12          2
## 58                      Approach=~BWin_v_Neut_R_Insula_run2      12          3
## 59                      Approach=~BWin_v_Neut_R_Insula_run2      12          4
## 60                      Approach=~BWin_v_Neut_R_Insula_run2      12          5
## 61                       Approach=~BWin_v_BLose_L_NAcc_run1      13          1
## 62                       Approach=~BWin_v_BLose_L_NAcc_run1      13          2
## 63                       Approach=~BWin_v_BLose_L_NAcc_run1      13          3
## 64                       Approach=~BWin_v_BLose_L_NAcc_run1      13          4
## 65                       Approach=~BWin_v_BLose_L_NAcc_run1      13          5
## 66                       Approach=~BWin_v_BLose_R_NAcc_run1      14          1
## 67                       Approach=~BWin_v_BLose_R_NAcc_run1      14          2
## 68                       Approach=~BWin_v_BLose_R_NAcc_run1      14          3
## 69                       Approach=~BWin_v_BLose_R_NAcc_run1      14          4
## 70                       Approach=~BWin_v_BLose_R_NAcc_run1      14          5
## 71                       Approach=~BWin_v_BLose_L_NAcc_run2      15          1
## 72                       Approach=~BWin_v_BLose_L_NAcc_run2      15          2
## 73                       Approach=~BWin_v_BLose_L_NAcc_run2      15          3
## 74                       Approach=~BWin_v_BLose_L_NAcc_run2      15          4
## 75                       Approach=~BWin_v_BLose_L_NAcc_run2      15          5
## 76                       Approach=~BWin_v_BLose_R_NAcc_run2      16          1
## 77                       Approach=~BWin_v_BLose_R_NAcc_run2      16          2
## 78                       Approach=~BWin_v_BLose_R_NAcc_run2      16          3
## 79                       Approach=~BWin_v_BLose_R_NAcc_run2      16          4
## 80                       Approach=~BWin_v_BLose_R_NAcc_run2      16          5
## 81                        Avoid=~ALose_v_Neut_L_Insula_run1      17          1
## 82                        Avoid=~ALose_v_Neut_L_Insula_run1      17          2
## 83                        Avoid=~ALose_v_Neut_L_Insula_run1      17          3
## 84                        Avoid=~ALose_v_Neut_L_Insula_run1      17          4
## 85                        Avoid=~ALose_v_Neut_L_Insula_run1      17          5
## 86                        Avoid=~ALose_v_Neut_R_Insula_run1      18          1
## 87                        Avoid=~ALose_v_Neut_R_Insula_run1      18          2
## 88                        Avoid=~ALose_v_Neut_R_Insula_run1      18          3
## 89                        Avoid=~ALose_v_Neut_R_Insula_run1      18          4
## 90                        Avoid=~ALose_v_Neut_R_Insula_run1      18          5
## 91                        Avoid=~ALose_v_Neut_L_Insula_run2      19          1
## 92                        Avoid=~ALose_v_Neut_L_Insula_run2      19          2
## 93                        Avoid=~ALose_v_Neut_L_Insula_run2      19          3
## 94                        Avoid=~ALose_v_Neut_L_Insula_run2      19          4
## 95                        Avoid=~ALose_v_Neut_L_Insula_run2      19          5
## 96                        Avoid=~ALose_v_Neut_R_Insula_run2      20          1
## 97                        Avoid=~ALose_v_Neut_R_Insula_run2      20          2
## 98                        Avoid=~ALose_v_Neut_R_Insula_run2      20          3
## 99                        Avoid=~ALose_v_Neut_R_Insula_run2      20          4
```

```
## 100                        Avoid=~ALose_v_Neut_R_Insula_run2        20           5
## 101                        Avoid=~BLose_v_Neut_L_Insula_run1        21           1
## 102                        Avoid=~BLose_v_Neut_L_Insula_run1        21           2
## 103                        Avoid=~BLose_v_Neut_L_Insula_run1        21           3
## 104                        Avoid=~BLose_v_Neut_L_Insula_run1        21           4
## 105                        Avoid=~BLose_v_Neut_L_Insula_run1        21           5
## 106                        Avoid=~BLose_v_Neut_R_Insula_run1        22           1
## 107                        Avoid=~BLose_v_Neut_R_Insula_run1        22           2
## 108                        Avoid=~BLose_v_Neut_R_Insula_run1        22           3
## 109                        Avoid=~BLose_v_Neut_R_Insula_run1        22           4
## 110                        Avoid=~BLose_v_Neut_R_Insula_run1        22           5
## 111                        Avoid=~BLose_v_Neut_L_Insula_run2        23           1
## 112                        Avoid=~BLose_v_Neut_L_Insula_run2        23           2
## 113                        Avoid=~BLose_v_Neut_L_Insula_run2        23           3
## 114                        Avoid=~BLose_v_Neut_L_Insula_run2        23           4
## 115                        Avoid=~BLose_v_Neut_L_Insula_run2        23           5
## 116                        Avoid=~BLose_v_Neut_R_Insula_run2        24           1
## 117                        Avoid=~BLose_v_Neut_R_Insula_run2        24           2
## 118                        Avoid=~BLose_v_Neut_R_Insula_run2        24           3
## 119                        Avoid=~BLose_v_Neut_R_Insula_run2        24           4
## 120                        Avoid=~BLose_v_Neut_R_Insula_run2        24           5
## 121                        Avoid=~BLose_v_BWin_L_Insula_run1        25           1
## 122                        Avoid=~BLose_v_BWin_L_Insula_run1        25           2
## 123                        Avoid=~BLose_v_BWin_L_Insula_run1        25           3
## 124                        Avoid=~BLose_v_BWin_L_Insula_run1        25           4
## 125                        Avoid=~BLose_v_BWin_L_Insula_run1        25           5
## 126                        Avoid=~BLose_v_BWin_R_Insula_run1        26           1
## 127                        Avoid=~BLose_v_BWin_R_Insula_run1        26           2
## 128                        Avoid=~BLose_v_BWin_R_Insula_run1        26           3
## 129                        Avoid=~BLose_v_BWin_R_Insula_run1        26           4
## 130                        Avoid=~BLose_v_BWin_R_Insula_run1        26           5
## 131                        Avoid=~BLose_v_BWin_L_Insula_run2        27           1
## 132                        Avoid=~BLose_v_BWin_L_Insula_run2        27           2
## 133                        Avoid=~BLose_v_BWin_L_Insula_run2        27           3
## 134                        Avoid=~BLose_v_BWin_L_Insula_run2        27           4
## 135                        Avoid=~BLose_v_BWin_L_Insula_run2        27           5
## 136                        Avoid=~BLose_v_BWin_R_Insula_run2        28           1
## 137                        Avoid=~BLose_v_BWin_R_Insula_run2        28           2
## 138                        Avoid=~BLose_v_BWin_R_Insula_run2        28           3
## 139                        Avoid=~BLose_v_BWin_R_Insula_run2        28           4
## 140                        Avoid=~BLose_v_BWin_R_Insula_run2        28           5
## 141        AWin_v_Neut_L_NAcc_run1~~AWin_v_Neut_L_NAcc_run1        29           1
## 142        AWin_v_Neut_L_NAcc_run1~~AWin_v_Neut_L_NAcc_run1        29           2
## 143        AWin_v_Neut_L_NAcc_run1~~AWin_v_Neut_L_NAcc_run1        29           3
## 144        AWin_v_Neut_L_NAcc_run1~~AWin_v_Neut_L_NAcc_run1        29           4
## 145        AWin_v_Neut_L_NAcc_run1~~AWin_v_Neut_L_NAcc_run1        29           5
## 146        AWin_v_Neut_R_NAcc_run1~~AWin_v_Neut_R_NAcc_run1        30           1
## 147        AWin_v_Neut_R_NAcc_run1~~AWin_v_Neut_R_NAcc_run1        30           2
## 148        AWin_v_Neut_R_NAcc_run1~~AWin_v_Neut_R_NAcc_run1        30           3
## 149        AWin_v_Neut_R_NAcc_run1~~AWin_v_Neut_R_NAcc_run1        30           4
## 150        AWin_v_Neut_R_NAcc_run1~~AWin_v_Neut_R_NAcc_run1        30           5
## 151    AWin_v_Neut_R_Insula_run1~~AWin_v_Neut_R_Insula_run1        31           1
## 152    AWin_v_Neut_R_Insula_run1~~AWin_v_Neut_R_Insula_run1        31           2
## 153    AWin_v_Neut_R_Insula_run1~~AWin_v_Neut_R_Insula_run1        31           3
```

```
## 154   AWin_v_Neut_R_Insula_run1~~AWin_v_Neut_R_Insula_run1       31           4
## 155   AWin_v_Neut_R_Insula_run1~~AWin_v_Neut_R_Insula_run1       31           5
## 156       AWin_v_Neut_L_NAcc_run2~~AWin_v_Neut_L_NAcc_run2       32           1
## 157       AWin_v_Neut_L_NAcc_run2~~AWin_v_Neut_L_NAcc_run2       32           2
## 158       AWin_v_Neut_L_NAcc_run2~~AWin_v_Neut_L_NAcc_run2       32           3
## 159       AWin_v_Neut_L_NAcc_run2~~AWin_v_Neut_L_NAcc_run2       32           4
## 160       AWin_v_Neut_L_NAcc_run2~~AWin_v_Neut_L_NAcc_run2       32           5
## 161       AWin_v_Neut_R_NAcc_run2~~AWin_v_Neut_R_NAcc_run2       33           1
## 162       AWin_v_Neut_R_NAcc_run2~~AWin_v_Neut_R_NAcc_run2       33           2
## 163       AWin_v_Neut_R_NAcc_run2~~AWin_v_Neut_R_NAcc_run2       33           3
## 164       AWin_v_Neut_R_NAcc_run2~~AWin_v_Neut_R_NAcc_run2       33           4
## 165       AWin_v_Neut_R_NAcc_run2~~AWin_v_Neut_R_NAcc_run2       33           5
## 166   AWin_v_Neut_R_Insula_run2~~AWin_v_Neut_R_Insula_run2       34           1
## 167   AWin_v_Neut_R_Insula_run2~~AWin_v_Neut_R_Insula_run2       34           2
## 168   AWin_v_Neut_R_Insula_run2~~AWin_v_Neut_R_Insula_run2       34           3
## 169   AWin_v_Neut_R_Insula_run2~~AWin_v_Neut_R_Insula_run2       34           4
## 170   AWin_v_Neut_R_Insula_run2~~AWin_v_Neut_R_Insula_run2       34           5
## 171       BWin_v_Neut_L_NAcc_run1~~BWin_v_Neut_L_NAcc_run1       35           1
## 172       BWin_v_Neut_L_NAcc_run1~~BWin_v_Neut_L_NAcc_run1       35           2
## 173       BWin_v_Neut_L_NAcc_run1~~BWin_v_Neut_L_NAcc_run1       35           3
## 174       BWin_v_Neut_L_NAcc_run1~~BWin_v_Neut_L_NAcc_run1       35           4
## 175       BWin_v_Neut_L_NAcc_run1~~BWin_v_Neut_L_NAcc_run1       35           5
## 176       BWin_v_Neut_R_NAcc_run1~~BWin_v_Neut_R_NAcc_run1       36           1
## 177       BWin_v_Neut_R_NAcc_run1~~BWin_v_Neut_R_NAcc_run1       36           2
## 178       BWin_v_Neut_R_NAcc_run1~~BWin_v_Neut_R_NAcc_run1       36           3
## 179       BWin_v_Neut_R_NAcc_run1~~BWin_v_Neut_R_NAcc_run1       36           4
## 180       BWin_v_Neut_R_NAcc_run1~~BWin_v_Neut_R_NAcc_run1       36           5
## 181   BWin_v_Neut_R_Insula_run1~~BWin_v_Neut_R_Insula_run1       37           1
## 182   BWin_v_Neut_R_Insula_run1~~BWin_v_Neut_R_Insula_run1       37           2
## 183   BWin_v_Neut_R_Insula_run1~~BWin_v_Neut_R_Insula_run1       37           3
## 184   BWin_v_Neut_R_Insula_run1~~BWin_v_Neut_R_Insula_run1       37           4
## 185   BWin_v_Neut_R_Insula_run1~~BWin_v_Neut_R_Insula_run1       37           5
## 186       BWin_v_Neut_L_NAcc_run2~~BWin_v_Neut_L_NAcc_run2       38           1
## 187       BWin_v_Neut_L_NAcc_run2~~BWin_v_Neut_L_NAcc_run2       38           2
## 188       BWin_v_Neut_L_NAcc_run2~~BWin_v_Neut_L_NAcc_run2       38           3
## 189       BWin_v_Neut_L_NAcc_run2~~BWin_v_Neut_L_NAcc_run2       38           4
## 190       BWin_v_Neut_L_NAcc_run2~~BWin_v_Neut_L_NAcc_run2       38           5
## 191       BWin_v_Neut_R_NAcc_run2~~BWin_v_Neut_R_NAcc_run2       39           1
## 192       BWin_v_Neut_R_NAcc_run2~~BWin_v_Neut_R_NAcc_run2       39           2
## 193       BWin_v_Neut_R_NAcc_run2~~BWin_v_Neut_R_NAcc_run2       39           3
## 194       BWin_v_Neut_R_NAcc_run2~~BWin_v_Neut_R_NAcc_run2       39           4
## 195       BWin_v_Neut_R_NAcc_run2~~BWin_v_Neut_R_NAcc_run2       39           5
## 196   BWin_v_Neut_R_Insula_run2~~BWin_v_Neut_R_Insula_run2       40           1
## 197   BWin_v_Neut_R_Insula_run2~~BWin_v_Neut_R_Insula_run2       40           2
## 198   BWin_v_Neut_R_Insula_run2~~BWin_v_Neut_R_Insula_run2       40           3
## 199   BWin_v_Neut_R_Insula_run2~~BWin_v_Neut_R_Insula_run2       40           4
## 200   BWin_v_Neut_R_Insula_run2~~BWin_v_Neut_R_Insula_run2       40           5
## 201     BWin_v_BLose_L_NAcc_run1~~BWin_v_BLose_L_NAcc_run1       41           1
## 202     BWin_v_BLose_L_NAcc_run1~~BWin_v_BLose_L_NAcc_run1       41           2
## 203     BWin_v_BLose_L_NAcc_run1~~BWin_v_BLose_L_NAcc_run1       41           3
## 204     BWin_v_BLose_L_NAcc_run1~~BWin_v_BLose_L_NAcc_run1       41           4
## 205     BWin_v_BLose_L_NAcc_run1~~BWin_v_BLose_L_NAcc_run1       41           5
## 206     BWin_v_BLose_R_NAcc_run1~~BWin_v_BLose_R_NAcc_run1       42           1
## 207     BWin_v_BLose_R_NAcc_run1~~BWin_v_BLose_R_NAcc_run1       42           2
```

```
## 208     BWin_v_BLose_R_NAcc_run1~~BWin_v_BLose_R_NAcc_run1       42          3
## 209     BWin_v_BLose_R_NAcc_run1~~BWin_v_BLose_R_NAcc_run1       42          4
## 210     BWin_v_BLose_R_NAcc_run1~~BWin_v_BLose_R_NAcc_run1       42          5
## 211     BWin_v_BLose_L_NAcc_run2~~BWin_v_BLose_L_NAcc_run2       43          1
## 212     BWin_v_BLose_L_NAcc_run2~~BWin_v_BLose_L_NAcc_run2       43          2
## 213     BWin_v_BLose_L_NAcc_run2~~BWin_v_BLose_L_NAcc_run2       43          3
## 214     BWin_v_BLose_L_NAcc_run2~~BWin_v_BLose_L_NAcc_run2       43          4
## 215     BWin_v_BLose_L_NAcc_run2~~BWin_v_BLose_L_NAcc_run2       43          5
## 216     BWin_v_BLose_R_NAcc_run2~~BWin_v_BLose_R_NAcc_run2       44          1
## 217     BWin_v_BLose_R_NAcc_run2~~BWin_v_BLose_R_NAcc_run2       44          2
## 218     BWin_v_BLose_R_NAcc_run2~~BWin_v_BLose_R_NAcc_run2       44          3
## 219     BWin_v_BLose_R_NAcc_run2~~BWin_v_BLose_R_NAcc_run2       44          4
## 220     BWin_v_BLose_R_NAcc_run2~~BWin_v_BLose_R_NAcc_run2       44          5
## 221 ALose_v_Neut_L_Insula_run1~~ALose_v_Neut_L_Insula_run1       45          1
## 222 ALose_v_Neut_L_Insula_run1~~ALose_v_Neut_L_Insula_run1       45          2
## 223 ALose_v_Neut_L_Insula_run1~~ALose_v_Neut_L_Insula_run1       45          3
## 224 ALose_v_Neut_L_Insula_run1~~ALose_v_Neut_L_Insula_run1       45          4
## 225 ALose_v_Neut_L_Insula_run1~~ALose_v_Neut_L_Insula_run1       45          5
## 226 ALose_v_Neut_R_Insula_run1~~ALose_v_Neut_R_Insula_run1       46          1
## 227 ALose_v_Neut_R_Insula_run1~~ALose_v_Neut_R_Insula_run1       46          2
## 228 ALose_v_Neut_R_Insula_run1~~ALose_v_Neut_R_Insula_run1       46          3
## 229 ALose_v_Neut_R_Insula_run1~~ALose_v_Neut_R_Insula_run1       46          4
## 230 ALose_v_Neut_R_Insula_run1~~ALose_v_Neut_R_Insula_run1       46          5
## 231 ALose_v_Neut_L_Insula_run2~~ALose_v_Neut_L_Insula_run2       47          1
## 232 ALose_v_Neut_L_Insula_run2~~ALose_v_Neut_L_Insula_run2       47          2
## 233 ALose_v_Neut_L_Insula_run2~~ALose_v_Neut_L_Insula_run2       47          3
## 234 ALose_v_Neut_L_Insula_run2~~ALose_v_Neut_L_Insula_run2       47          4
## 235 ALose_v_Neut_L_Insula_run2~~ALose_v_Neut_L_Insula_run2       47          5
## 236 ALose_v_Neut_R_Insula_run2~~ALose_v_Neut_R_Insula_run2       48          1
## 237 ALose_v_Neut_R_Insula_run2~~ALose_v_Neut_R_Insula_run2       48          2
## 238 ALose_v_Neut_R_Insula_run2~~ALose_v_Neut_R_Insula_run2       48          3
## 239 ALose_v_Neut_R_Insula_run2~~ALose_v_Neut_R_Insula_run2       48          4
## 240 ALose_v_Neut_R_Insula_run2~~ALose_v_Neut_R_Insula_run2       48          5
## 241 BLose_v_Neut_L_Insula_run1~~BLose_v_Neut_L_Insula_run1       49          1
## 242 BLose_v_Neut_L_Insula_run1~~BLose_v_Neut_L_Insula_run1       49          2
## 243 BLose_v_Neut_L_Insula_run1~~BLose_v_Neut_L_Insula_run1       49          3
## 244 BLose_v_Neut_L_Insula_run1~~BLose_v_Neut_L_Insula_run1       49          4
## 245 BLose_v_Neut_L_Insula_run1~~BLose_v_Neut_L_Insula_run1       49          5
## 246 BLose_v_Neut_R_Insula_run1~~BLose_v_Neut_R_Insula_run1       50          1
## 247 BLose_v_Neut_R_Insula_run1~~BLose_v_Neut_R_Insula_run1       50          2
## 248 BLose_v_Neut_R_Insula_run1~~BLose_v_Neut_R_Insula_run1       50          3
## 249 BLose_v_Neut_R_Insula_run1~~BLose_v_Neut_R_Insula_run1       50          4
## 250 BLose_v_Neut_R_Insula_run1~~BLose_v_Neut_R_Insula_run1       50          5
## 251 BLose_v_Neut_L_Insula_run2~~BLose_v_Neut_L_Insula_run2       51          1
## 252 BLose_v_Neut_L_Insula_run2~~BLose_v_Neut_L_Insula_run2       51          2
## 253 BLose_v_Neut_L_Insula_run2~~BLose_v_Neut_L_Insula_run2       51          3
## 254 BLose_v_Neut_L_Insula_run2~~BLose_v_Neut_L_Insula_run2       51          4
## 255 BLose_v_Neut_L_Insula_run2~~BLose_v_Neut_L_Insula_run2       51          5
## 256 BLose_v_Neut_R_Insula_run2~~BLose_v_Neut_R_Insula_run2       52          1
## 257 BLose_v_Neut_R_Insula_run2~~BLose_v_Neut_R_Insula_run2       52          2
## 258 BLose_v_Neut_R_Insula_run2~~BLose_v_Neut_R_Insula_run2       52          3
## 259 BLose_v_Neut_R_Insula_run2~~BLose_v_Neut_R_Insula_run2       52          4
## 260 BLose_v_Neut_R_Insula_run2~~BLose_v_Neut_R_Insula_run2       52          5
## 261 BLose_v_BWin_L_Insula_run1~~BLose_v_BWin_L_Insula_run1       53          1
```

```
## 262 BLose_v_BWin_L_Insula_run1~~BLose_v_BWin_L_Insula_run1        53           2
## 263 BLose_v_BWin_L_Insula_run1~~BLose_v_BWin_L_Insula_run1        53           3
## 264 BLose_v_BWin_L_Insula_run1~~BLose_v_BWin_L_Insula_run1        53           4
## 265 BLose_v_BWin_L_Insula_run1~~BLose_v_BWin_L_Insula_run1        53           5
## 266 BLose_v_BWin_R_Insula_run1~~BLose_v_BWin_R_Insula_run1        54           1
## 267 BLose_v_BWin_R_Insula_run1~~BLose_v_BWin_R_Insula_run1        54           2
## 268 BLose_v_BWin_R_Insula_run1~~BLose_v_BWin_R_Insula_run1        54           3
## 269 BLose_v_BWin_R_Insula_run1~~BLose_v_BWin_R_Insula_run1        54           4
## 270 BLose_v_BWin_R_Insula_run1~~BLose_v_BWin_R_Insula_run1        54           5
## 271 BLose_v_BWin_L_Insula_run2~~BLose_v_BWin_L_Insula_run2        55           1
## 272 BLose_v_BWin_L_Insula_run2~~BLose_v_BWin_L_Insula_run2        55           2
## 273 BLose_v_BWin_L_Insula_run2~~BLose_v_BWin_L_Insula_run2        55           3
## 274 BLose_v_BWin_L_Insula_run2~~BLose_v_BWin_L_Insula_run2        55           4
## 275 BLose_v_BWin_L_Insula_run2~~BLose_v_BWin_L_Insula_run2        55           5
## 276 BLose_v_BWin_R_Insula_run2~~BLose_v_BWin_R_Insula_run2        56           1
## 277 BLose_v_BWin_R_Insula_run2~~BLose_v_BWin_R_Insula_run2        56           2
## 278 BLose_v_BWin_R_Insula_run2~~BLose_v_BWin_R_Insula_run2        56           3
## 279 BLose_v_BWin_R_Insula_run2~~BLose_v_BWin_R_Insula_run2        56           4
## 280 BLose_v_BWin_R_Insula_run2~~BLose_v_BWin_R_Insula_run2        56           5
## 281                                     Approach~~Approach        57           1
## 282                                     Approach~~Approach        57           2
## 283                                     Approach~~Approach        57           3
## 284                                     Approach~~Approach        57           4
## 285                                     Approach~~Approach        57           5
## 286                                           Avoid~~Avoid        58           1
## 287                                           Avoid~~Avoid        58           2
## 288                                           Avoid~~Avoid        58           3
## 289                                           Avoid~~Avoid        58           4
## 290                                           Avoid~~Avoid        58           5
## 291                                        Approach~~Avoid        59           1
## 292                                        Approach~~Avoid        59           2
## 293                                        Approach~~Avoid        59           3
## 294                                        Approach~~Avoid        59           4
## 295                                        Approach~~Avoid        59           5
## 296                              AWin_v_Neut_L_NAcc_run1~1        60           1
## 297                              AWin_v_Neut_L_NAcc_run1~1        60           2
## 298                              AWin_v_Neut_L_NAcc_run1~1        60           3
## 299                              AWin_v_Neut_L_NAcc_run1~1        60           4
## 300                              AWin_v_Neut_L_NAcc_run1~1        60           5
## 301                              AWin_v_Neut_R_NAcc_run1~1        61           1
## 302                              AWin_v_Neut_R_NAcc_run1~1        61           2
## 303                              AWin_v_Neut_R_NAcc_run1~1        61           3
## 304                              AWin_v_Neut_R_NAcc_run1~1        61           4
## 305                              AWin_v_Neut_R_NAcc_run1~1        61           5
## 306                            AWin_v_Neut_R_Insula_run1~1        62           1
## 307                            AWin_v_Neut_R_Insula_run1~1        62           2
## 308                            AWin_v_Neut_R_Insula_run1~1        62           3
## 309                            AWin_v_Neut_R_Insula_run1~1        62           4
## 310                            AWin_v_Neut_R_Insula_run1~1        62           5
## 311                              AWin_v_Neut_L_NAcc_run2~1        63           1
## 312                              AWin_v_Neut_L_NAcc_run2~1        63           2
## 313                              AWin_v_Neut_L_NAcc_run2~1        63           3
## 314                              AWin_v_Neut_L_NAcc_run2~1        63           4
## 315                              AWin_v_Neut_L_NAcc_run2~1        63           5
```

```
## 316                      AWin_v_Neut_R_NAcc_run2~1     64          1
## 317                      AWin_v_Neut_R_NAcc_run2~1     64          2
## 318                      AWin_v_Neut_R_NAcc_run2~1     64          3
## 319                      AWin_v_Neut_R_NAcc_run2~1     64          4
## 320                      AWin_v_Neut_R_NAcc_run2~1     64          5
## 321                    AWin_v_Neut_R_Insula_run2~1     65          1
## 322                    AWin_v_Neut_R_Insula_run2~1     65          2
## 323                    AWin_v_Neut_R_Insula_run2~1     65          3
## 324                    AWin_v_Neut_R_Insula_run2~1     65          4
## 325                    AWin_v_Neut_R_Insula_run2~1     65          5
## 326                      BWin_v_Neut_L_NAcc_run1~1     66          1
## 327                      BWin_v_Neut_L_NAcc_run1~1     66          2
## 328                      BWin_v_Neut_L_NAcc_run1~1     66          3
## 329                      BWin_v_Neut_L_NAcc_run1~1     66          4
## 330                      BWin_v_Neut_L_NAcc_run1~1     66          5
## 331                      BWin_v_Neut_R_NAcc_run1~1     67          1
## 332                      BWin_v_Neut_R_NAcc_run1~1     67          2
## 333                      BWin_v_Neut_R_NAcc_run1~1     67          3
## 334                      BWin_v_Neut_R_NAcc_run1~1     67          4
## 335                      BWin_v_Neut_R_NAcc_run1~1     67          5
## 336                    BWin_v_Neut_R_Insula_run1~1     68          1
## 337                    BWin_v_Neut_R_Insula_run1~1     68          2
## 338                    BWin_v_Neut_R_Insula_run1~1     68          3
## 339                    BWin_v_Neut_R_Insula_run1~1     68          4
## 340                    BWin_v_Neut_R_Insula_run1~1     68          5
## 341                      BWin_v_Neut_L_NAcc_run2~1     69          1
## 342                      BWin_v_Neut_L_NAcc_run2~1     69          2
## 343                      BWin_v_Neut_L_NAcc_run2~1     69          3
## 344                      BWin_v_Neut_L_NAcc_run2~1     69          4
## 345                      BWin_v_Neut_L_NAcc_run2~1     69          5
## 346                      BWin_v_Neut_R_NAcc_run2~1     70          1
## 347                      BWin_v_Neut_R_NAcc_run2~1     70          2
## 348                      BWin_v_Neut_R_NAcc_run2~1     70          3
## 349                      BWin_v_Neut_R_NAcc_run2~1     70          4
## 350                      BWin_v_Neut_R_NAcc_run2~1     70          5
## 351                    BWin_v_Neut_R_Insula_run2~1     71          1
## 352                    BWin_v_Neut_R_Insula_run2~1     71          2
## 353                    BWin_v_Neut_R_Insula_run2~1     71          3
## 354                    BWin_v_Neut_R_Insula_run2~1     71          4
## 355                    BWin_v_Neut_R_Insula_run2~1     71          5
## 356                     BWin_v_BLose_L_NAcc_run1~1     72          1
## 357                     BWin_v_BLose_L_NAcc_run1~1     72          2
## 358                     BWin_v_BLose_L_NAcc_run1~1     72          3
## 359                     BWin_v_BLose_L_NAcc_run1~1     72          4
## 360                     BWin_v_BLose_L_NAcc_run1~1     72          5
## 361                     BWin_v_BLose_R_NAcc_run1~1     73          1
## 362                     BWin_v_BLose_R_NAcc_run1~1     73          2
## 363                     BWin_v_BLose_R_NAcc_run1~1     73          3
## 364                     BWin_v_BLose_R_NAcc_run1~1     73          4
## 365                     BWin_v_BLose_R_NAcc_run1~1     73          5
## 366                     BWin_v_BLose_L_NAcc_run2~1     74          1
## 367                     BWin_v_BLose_L_NAcc_run2~1     74          2
## 368                     BWin_v_BLose_L_NAcc_run2~1     74          3
## 369                     BWin_v_BLose_L_NAcc_run2~1     74          4
```

```
## 370                     BWin_v_BLose_L_NAcc_run2~1   74    5
## 371                     BWin_v_BLose_R_NAcc_run2~1   75    1
## 372                     BWin_v_BLose_R_NAcc_run2~1   75    2
## 373                     BWin_v_BLose_R_NAcc_run2~1   75    3
## 374                     BWin_v_BLose_R_NAcc_run2~1   75    4
## 375                     BWin_v_BLose_R_NAcc_run2~1   75    5
## 376                   ALose_v_Neut_L_Insula_run1~1   76    1
## 377                   ALose_v_Neut_L_Insula_run1~1   76    2
## 378                   ALose_v_Neut_L_Insula_run1~1   76    3
## 379                   ALose_v_Neut_L_Insula_run1~1   76    4
## 380                   ALose_v_Neut_L_Insula_run1~1   76    5
## 381                   ALose_v_Neut_R_Insula_run1~1   77    1
## 382                   ALose_v_Neut_R_Insula_run1~1   77    2
## 383                   ALose_v_Neut_R_Insula_run1~1   77    3
## 384                   ALose_v_Neut_R_Insula_run1~1   77    4
## 385                   ALose_v_Neut_R_Insula_run1~1   77    5
## 386                   ALose_v_Neut_L_Insula_run2~1   78    1
## 387                   ALose_v_Neut_L_Insula_run2~1   78    2
## 388                   ALose_v_Neut_L_Insula_run2~1   78    3
## 389                   ALose_v_Neut_L_Insula_run2~1   78    4
## 390                   ALose_v_Neut_L_Insula_run2~1   78    5
## 391                   ALose_v_Neut_R_Insula_run2~1   79    1
## 392                   ALose_v_Neut_R_Insula_run2~1   79    2
## 393                   ALose_v_Neut_R_Insula_run2~1   79    3
## 394                   ALose_v_Neut_R_Insula_run2~1   79    4
## 395                   ALose_v_Neut_R_Insula_run2~1   79    5
## 396                   BLose_v_Neut_L_Insula_run1~1   80    1
## 397                   BLose_v_Neut_L_Insula_run1~1   80    2
## 398                   BLose_v_Neut_L_Insula_run1~1   80    3
## 399                   BLose_v_Neut_L_Insula_run1~1   80    4
## 400                   BLose_v_Neut_L_Insula_run1~1   80    5
## 401                   BLose_v_Neut_R_Insula_run1~1   81    1
## 402                   BLose_v_Neut_R_Insula_run1~1   81    2
## 403                   BLose_v_Neut_R_Insula_run1~1   81    3
## 404                   BLose_v_Neut_R_Insula_run1~1   81    4
## 405                   BLose_v_Neut_R_Insula_run1~1   81    5
## 406                   BLose_v_Neut_L_Insula_run2~1   82    1
## 407                   BLose_v_Neut_L_Insula_run2~1   82    2
## 408                   BLose_v_Neut_L_Insula_run2~1   82    3
## 409                   BLose_v_Neut_L_Insula_run2~1   82    4
## 410                   BLose_v_Neut_L_Insula_run2~1   82    5
## 411                   BLose_v_Neut_R_Insula_run2~1   83    1
## 412                   BLose_v_Neut_R_Insula_run2~1   83    2
## 413                   BLose_v_Neut_R_Insula_run2~1   83    3
## 414                   BLose_v_Neut_R_Insula_run2~1   83    4
## 415                   BLose_v_Neut_R_Insula_run2~1   83    5
## 416                   BLose_v_BWin_L_Insula_run1~1   84    1
## 417                   BLose_v_BWin_L_Insula_run1~1   84    2
## 418                   BLose_v_BWin_L_Insula_run1~1   84    3
## 419                   BLose_v_BWin_L_Insula_run1~1   84    4
## 420                   BLose_v_BWin_L_Insula_run1~1   84    5
## 421                   BLose_v_BWin_R_Insula_run1~1   85    1
## 422                   BLose_v_BWin_R_Insula_run1~1   85    2
## 423                   BLose_v_BWin_R_Insula_run1~1   85    3
```

```
## 424                   BLose_v_BWin_R_Insula_run1~1      85          4
## 425                   BLose_v_BWin_R_Insula_run1~1      85          5
## 426                   BLose_v_BWin_L_Insula_run2~1      86          1
## 427                   BLose_v_BWin_L_Insula_run2~1      86          2
## 428                   BLose_v_BWin_L_Insula_run2~1      86          3
## 429                   BLose_v_BWin_L_Insula_run2~1      86          4
## 430                   BLose_v_BWin_L_Insula_run2~1      86          5
## 431                   BLose_v_BWin_R_Insula_run2~1      87          1
## 432                   BLose_v_BWin_R_Insula_run2~1      87          2
## 433                   BLose_v_BWin_R_Insula_run2~1      87          3
## 434                   BLose_v_BWin_R_Insula_run2~1      87          4
## 435                   BLose_v_BWin_R_Insula_run2~1      87          5
## 436                                     Approach~1      88          1
## 437                                     Approach~1      88          2
## 438                                     Approach~1      88          3
## 439                                     Approach~1      88          4
## 440                                     Approach~1      88          5
## 441                                        Avoid~1      89          1
## 442                                        Avoid~1      89          2
## 443                                        Avoid~1      89          3
## 444                                        Avoid~1      89          4
## 445                                        Avoid~1      89          5
## 446                                          rmsea      90          1
## 447                                          rmsea      90          2
## 448                                          rmsea      90          3
## 449                                          rmsea      90          4
## 450                                          rmsea      90          5
## 451                                            cfi      91          1
## 452                                            cfi      91          2
## 453                                            cfi      91          3
## 454                                            cfi      91          4
## 455                                            cfi      91          5
## 456                                            tli      92          1
## 457                                            tli      92          2
## 458                                            tli      92          3
## 459                                            tli      92          4
## 460                                            tli      92          5
## 461                                            gfi      93          1
## 462                                            gfi      93          2
## 463                                            gfi      93          3
## 464                                            gfi      93          4
## 465                                            gfi      93          5
## 466                                           srmr      94          1
## 467                                           srmr      94          2
## 468                                           srmr      94          3
## 469                                           srmr      94          4
## 470                                           srmr      94          5
##        est   p
## 1   -0.002 0.8
## 2    0.009 0.4
## 3    0.003 0.6
## 4   -0.008 0.4
## 5   -0.007 0.0
## 6    0.006 0.2
```

```
## 7   -0.004 0.6
## 8    0.000 0.8
## 9    0.003 1.0
## 10  -0.004 0.8
## 11   0.007 0.4
## 12   0.001 0.8
## 13  -0.005 0.6
## 14   0.001 0.8
## 15   0.002 0.8
## 16   0.001 0.6
## 17   0.003 0.6
## 18   0.001 1.0
## 19  -0.002 0.4
## 20  -0.006 0.4
## 21   0.014 0.2
## 22  -0.001 0.8
## 23  -0.001 0.2
## 24  -0.001 0.8
## 25  -0.010 0.6
## 26  -0.011 0.0
## 27  -0.002 1.0
## 28  -0.004 0.4
## 29   0.008 0.0
## 30   0.012 0.2
## 31   0.011 0.4
## 32   0.006 0.4
## 33   0.000 1.0
## 34  -0.001 0.6
## 35  -0.024 0.0
## 36  -0.006 0.8
## 37  -0.008 0.0
## 38   0.002 0.6
## 39   0.010 0.0
## 40  -0.005 1.0
## 41   0.010 0.6
## 42  -0.001 1.0
## 43   0.001 0.6
## 44  -0.002 1.0
## 45  -0.011 0.8
## 46  -0.008 0.8
## 47  -0.002 0.8
## 48   0.003 0.6
## 49   0.008 0.4
## 50  -0.014 0.2
## 51  -0.023 0.0
## 52  -0.014 0.0
## 53   0.000 1.0
## 54   0.016 0.0
## 55   0.021 0.2
## 56   0.022 0.0
## 57   0.000 0.8
## 58   0.005 0.0
## 59  -0.012 0.0
## 60  -0.013 0.4
```

```
## 61  -0.001 0.8
## 62   0.003 0.6
## 63   0.005 0.6
## 64  -0.003 0.8
## 65  -0.015 0.0
## 66   0.004 0.8
## 67  -0.006 0.2
## 68  -0.002 1.0
## 69   0.002 0.6
## 70   0.011 0.4
## 71  -0.018 0.2
## 72   0.001 0.4
## 73   0.006 0.6
## 74   0.001 0.8
## 75  -0.002 0.8
## 76   0.004 0.8
## 77   0.002 0.8
## 78  -0.002 1.0
## 79   0.002 0.8
## 80  -0.009 0.8
## 81  -0.005 0.8
## 82   0.005 1.0
## 83   0.000 1.0
## 84  -0.002 0.8
## 85   0.001 0.8
## 86  -0.014 0.4
## 87  -0.010 0.4
## 88   0.002 0.8
## 89   0.010 0.2
## 90   0.009 0.4
## 91   0.009 0.8
## 92   0.010 0.2
## 93  -0.001 0.8
## 94  -0.009 0.2
## 95  -0.009 0.4
## 96  -0.023 0.2
## 97  -0.004 1.0
## 98   0.014 0.0
## 99   0.001 0.8
## 100 -0.007 0.6
## 101 -0.028 0.0
## 102 -0.006 0.6
## 103  0.011 0.0
## 104  0.010 0.4
## 105 -0.010 0.4
## 106  0.006 0.6
## 107  0.003 0.6
## 108  0.003 0.6
## 109 -0.009 0.2
## 110 -0.002 1.0
## 111 -0.011 0.6
## 112 -0.004 0.6
## 113  0.006 0.2
## 114  0.003 0.8
```

```
## 115 -0.003 0.6
## 116 -0.010 0.4
## 117  0.002 0.8
## 118  0.003 0.6
## 119  0.002 0.8
## 120 -0.006 1.0
## 121 -0.023 0.0
## 122 -0.021 0.0
## 123  0.006 0.4
## 124  0.019 0.0
## 125  0.011 0.0
## 126  0.016 0.0
## 127  0.009 0.4
## 128 -0.006 0.0
## 129 -0.012 0.0
## 130  0.007 0.4
## 131 -0.013 0.2
## 132 -0.020 0.0
## 133  0.001 1.0
## 134  0.015 0.0
## 135  0.020 0.4
## 136  0.009 0.2
## 137  0.003 0.4
## 138 -0.003 0.6
## 139 -0.004 0.4
## 140  0.000 0.8
## 141 -0.002 0.6
## 142 -0.001 1.0
## 143  0.001 0.0
## 144  0.001 0.8
## 145 -0.001 0.6
## 146 -0.003 0.2
## 147 -0.002 0.0
## 148  0.001 0.8
## 149  0.002 0.0
## 150  0.002 0.4
## 151  0.003 0.0
## 152  0.000 0.8
## 153 -0.002 0.0
## 154  0.001 0.4
## 155  0.002 0.2
## 156 -0.002 0.6
## 157  0.001 1.0
## 158  0.000 0.6
## 159  0.000 0.8
## 160 -0.001 0.8
## 161  0.000 0.4
## 162 -0.001 0.4
## 163 -0.001 0.8
## 164  0.001 0.4
## 165  0.004 0.0
## 166  0.004 0.2
## 167  0.002 0.2
## 168 -0.001 0.4
```

```
## 169 -0.002 0.2
## 170 -0.003 0.0
## 171 -0.001 0.6
## 172  0.000 0.8
## 173  0.000 0.6
## 174  0.000 0.8
## 175  0.003 0.0
## 176 -0.001 0.6
## 177  0.001 0.8
## 178  0.000 0.6
## 179  0.001 0.4
## 180 -0.002 0.6
## 181  0.001 0.6
## 182  0.001 0.0
## 183 -0.001 0.6
## 184  0.000 0.8
## 185 -0.001 0.4
## 186  0.000 0.8
## 187  0.001 0.8
## 188  0.001 0.6
## 189 -0.002 0.4
## 190 -0.002 0.8
## 191  0.000 0.4
## 192  0.001 0.4
## 193  0.002 0.0
## 194 -0.001 0.2
## 195 -0.005 0.2
## 196 -0.001 0.4
## 197  0.002 0.2
## 198 -0.001 1.0
## 199 -0.001 0.6
## 200  0.001 1.0
## 201  0.005 0.0
## 202  0.000 1.0
## 203  0.000 0.2
## 204 -0.003 0.0
## 205  0.000 0.8
## 206  0.000 0.8
## 207  0.002 0.0
## 208  0.002 0.2
## 209 -0.001 0.4
## 210 -0.009 0.0
## 211  0.001 1.0
## 212  0.000 0.4
## 213  0.001 0.4
## 214 -0.001 0.4
## 215 -0.001 0.8
## 216 -0.004 0.2
## 217 -0.002 0.0
## 218  0.001 0.8
## 219  0.002 0.0
## 220  0.002 0.4
## 221  0.001 0.6
## 222 -0.001 0.0
```

```
## 223  0.002 0.0
## 224 -0.001 1.0
## 225 -0.004 0.4
## 226  0.002 0.2
## 227  0.003 0.2
## 228  0.000 0.6
## 229 -0.003 0.0
## 230 -0.002 0.8
## 231 -0.001 0.6
## 232  0.000 1.0
## 233  0.000 0.6
## 234 -0.001 0.4
## 235  0.002 0.2
## 236  0.007 0.2
## 237  0.003 0.0
## 238 -0.002 0.0
## 239 -0.003 0.0
## 240  0.000 0.8
## 241  0.000 1.0
## 242  0.001 0.8
## 243  0.001 0.6
## 244  0.000 0.8
## 245 -0.003 0.4
## 246 -0.001 0.4
## 247  0.001 0.6
## 248  0.001 0.6
## 249  0.000 1.0
## 250 -0.001 1.0
## 251 -0.003 0.0
## 252  0.000 1.0
## 253  0.003 0.0
## 254 -0.001 0.8
## 255 -0.004 0.0
## 256 -0.003 0.4
## 257 -0.002 0.8
## 258  0.000 0.8
## 259  0.002 0.4
## 260  0.003 0.6
## 261  0.002 0.4
## 262  0.002 0.2
## 263 -0.001 0.4
## 264 -0.003 0.0
## 265  0.002 0.2
## 266  0.005 0.0
## 267  0.002 0.0
## 268  0.000 0.8
## 269 -0.003 0.0
## 270 -0.002 0.6
## 271  0.002 0.6
## 272  0.002 0.0
## 273 -0.001 0.2
## 274  0.000 1.0
## 275 -0.003 0.0
## 276  0.000 1.0
```

```
## 277  0.001 0.8
## 278  0.001 0.6
## 279 -0.001 0.4
## 280  0.000 0.8
## 281  0.000 1.0
## 282  0.000 1.0
## 283  0.000 1.0
## 284  0.000 1.0
## 285  0.000 1.0
## 286  0.000 1.0
## 287  0.000 1.0
## 288  0.000 1.0
## 289  0.000 1.0
## 290  0.000 1.0
## 291  0.061 0.2
## 292  0.028 0.2
## 293 -0.040 0.0
## 294  0.010 1.0
## 295 -0.043 0.4
## 296  0.031 0.0
## 297  0.014 0.2
## 298 -0.008 0.0
## 299 -0.013 0.0
## 300 -0.015 0.2
## 301 -0.007 1.0
## 302  0.002 0.8
## 303  0.011 0.0
## 304 -0.006 0.2
## 305 -0.016 0.2
## 306 -0.008 0.4
## 307  0.003 0.6
## 308  0.003 0.6
## 309 -0.002 0.8
## 310 -0.001 0.6
## 311  0.015 0.0
## 312  0.008 0.2
## 313 -0.002 0.4
## 314 -0.010 0.0
## 315 -0.008 0.4
## 316  0.000 0.6
## 317  0.008 0.4
## 318  0.003 1.0
## 319 -0.005 0.2
## 320 -0.014 0.2
## 321  0.003 0.0
## 322 -0.001 0.8
## 323  0.004 0.6
## 324 -0.004 0.2
## 325 -0.003 0.0
## 326  0.002 0.4
## 327 -0.003 0.8
## 328 -0.002 0.4
## 329  0.005 0.2
## 330 -0.003 0.8
```

```
## 331  0.020 0.0
## 332  0.000 0.4
## 333 -0.005 0.2
## 334 -0.006 0.6
## 335  0.003 0.4
## 336  0.003 0.6
## 337 -0.005 0.4
## 338 -0.002 0.6
## 339  0.004 0.2
## 340  0.004 0.6
## 341  0.002 0.6
## 342  0.000 1.0
## 343 -0.004 0.2
## 344  0.005 0.8
## 345  0.000 0.6
## 346  0.019 0.0
## 347 -0.001 0.8
## 348 -0.005 0.2
## 349 -0.003 0.2
## 350  0.001 1.0
## 351  0.016 0.0
## 352 -0.003 0.8
## 353 -0.001 0.6
## 354 -0.001 1.0
## 355 -0.009 0.4
## 356  0.019 0.2
## 357 -0.002 0.4
## 358 -0.005 0.4
## 359 -0.007 0.4
## 360  0.012 0.2
## 361  0.013 0.0
## 362  0.015 0.0
## 363 -0.005 0.0
## 364 -0.014 0.0
## 365 -0.001 0.6
## 366  0.018 0.0
## 367  0.008 0.4
## 368 -0.009 0.0
## 369 -0.009 0.2
## 370  0.008 0.2
## 371  0.013 0.0
## 372  0.008 0.0
## 373 -0.010 0.0
## 374 -0.004 0.0
## 375  0.006 0.4
## 376  0.005 1.0
## 377 -0.003 0.6
## 378 -0.004 1.0
## 379  0.002 1.0
## 380  0.008 0.8
## 381  0.006 0.8
## 382  0.005 0.2
## 383  0.001 0.8
## 384 -0.005 0.2
```

```
## 385 -0.011 0.4
## 386  0.001 0.8
## 387  0.001 0.8
## 388 -0.007 0.6
## 389  0.004 0.6
## 390  0.008 0.6
## 391 -0.002 0.4
## 392  0.005 0.4
## 393  0.002 0.4
## 394  0.000 0.6
## 395 -0.013 0.0
## 396  0.022 0.0
## 397 -0.002 0.6
## 398 -0.008 0.0
## 399 -0.001 1.0
## 400  0.005 0.8
## 401 -0.009 0.4
## 402 -0.011 0.0
## 403  0.002 0.6
## 404  0.008 0.2
## 405  0.012 0.0
## 406 -0.003 0.4
## 407 -0.002 1.0
## 408  0.002 0.2
## 409  0.003 0.8
## 410 -0.004 0.8
## 411 -0.008 0.4
## 412 -0.006 0.2
## 413  0.006 0.0
## 414  0.005 0.4
## 415 -0.006 0.4
## 416 -0.002 0.8
## 417  0.008 0.0
## 418  0.002 0.2
## 419 -0.007 0.2
## 420 -0.007 0.6
## 421  0.002 0.4
## 422  0.000 0.8
## 423  0.004 0.4
## 424  0.000 0.8
## 425 -0.013 0.6
## 426  0.000 0.8
## 427  0.000 0.8
## 428  0.004 0.2
## 429 -0.007 0.0
## 430  0.004 0.8
## 431 -0.027 0.0
## 432 -0.007 0.2
## 433  0.008 0.0
## 434  0.008 0.0
## 435  0.006 0.4
## 436  0.000 1.0
## 437  0.000 1.0
## 438  0.000 1.0
```

```
## 439  0.000 1.0
## 440  0.000 1.0
## 441  0.000 1.0
## 442  0.000 1.0
## 443  0.000 1.0
## 444  0.000 1.0
## 445  0.000 1.0
## 446  0.006 0.0
## 447 -0.004 0.2
## 448 -0.004 0.8
## 449  0.002 0.0
## 450  0.011 0.2
## 451 -0.065 0.8
## 452  0.010 1.0
## 453  0.037 0.2
## 454  0.006 0.8
## 455 -0.071 0.4
## 456 -0.071 0.8
## 457  0.010 1.0
## 458  0.040 0.2
## 459  0.007 0.8
## 460 -0.077 0.4
## 461 -0.016 0.0
## 462  0.012 0.4
## 463  0.012 0.8
## 464 -0.005 0.0
## 465 -0.029 0.2
## 466  0.006 0.4
## 467 -0.002 0.8
## 468 -0.004 0.8
## 469  0.000 0.6
## 470  0.008 0.4
```

Sensitivity Analyses

In the manuscript, several sensitivity analyses are proposed:

1. Differences (CFA/EFA) in effects across data collected only at UM site.
2. Resampling of CFA/EFA in effects across larger ABCD data to evaluate the *stability* of model effects

UM Specific CFA/ESEM/EFA

In the chunk below, the PDS simulated data will be used as the filter, assuming that PDS 3 are the values reflecting the site of interest.

```
ABCD_site_spec <- sim_ABCD_data %>% filter(PDS == 3)

sim_AHRB_data$PDS <- NA
sim_MLS_data$PDS <- NA

UM_site_spec <- rbind(sim_AHRB_data,
                      sim_MLS_data,
                      ABCD_site_spec # UM specified ABCD data
                      )
```

CFA

```
UM_all_sample <- cfa(model = MID_model, data = UM_site_spec,
               estimator = "MLR", std.lv = TRUE, meanstructure = TRUE)

UM_config_cfa <- cfa(model = MID_model, data = UM_site_spec, group = 'set',
                      estimator = "MLR", std.lv = TRUE, meanstructure = TRUE)

UM_metric_cfa <-cfa(model = MID_model, data = UM_site_spec,
                 group = 'set', group.equal=c("loadings"),
                 estimator = "MLR", std.lv = TRUE, meanstructure = TRUE)

UMsite_out <- matrix(NA, ncol = 9, nrow = 3)
colnames(UMsite_out) <- c("model","chisq","df","pvalue", "rmsea", "cfi", "srmr",
                   "AIC", "BIC")

# save fit measures from models
UMsite_out[1,2:7] <- round(data.matrix(fitmeasures(UM_all_sample,
                                          fit.measures = c("chisq","df","pvalue",
                                                           "rmsea", "cfi", "srmr"))),
                    digits=3)

UMsite_out[2,2:7] <- round(data.matrix(fitmeasures(UM_config_cfa,
                                          fit.measures = c("chisq","df","pvalue",
                                                           "rmsea", "cfi", "srmr"))),
                            digits=3)

UMsite_out[3,2:7] <- round(data.matrix(fitmeasures(UM_metric_cfa,
                                          fit.measures = c("chisq","df","pvalue",
                                                           "rmsea", "cfi", "srmr"))),
                    digits=3)

UMsite_out[1,8] <- round(AIC(UM_all_sample),3)
UMsite_out[2,8] <- round(AIC(UM_config_cfa),3)
UMsite_out[3,8] <- round(AIC(UM_metric_cfa),3)

# BIC models
UMsite_out[1,9] <- round(BIC(UM_all_sample),3)
UMsite_out[2,9] <- round(BIC(UM_config_cfa),3)
UMsite_out[3,9] <- round(BIC(UM_metric_cfa),3)

UMsite_out[1:3,1] <-  c("Overall CFA", "Config MG-CFA", "Metric MG-CFA")


UMsite_out %>%
  knitr::kable(
    caption = "Fit statistics from MG-CFA and ESEM models",
    booktabs = TRUE
    )
```

Table 3: Fit statistics from MG-CFA and ESEM models

model	chisq	df	pvalue	rmsea	cfi	srmr	AIC	BIC
Overall CFA	1167.367	349	0	0.065	0.665	0.057	-9462.261	-9094.083
Config MG-CFA	1915.884	1047	0	0.067	0.654	0.073	-9275.648	-8171.115
Metric MG-CFA	1974.859	1099	0	0.065	0.651	0.078	-9320.674	-8441.379

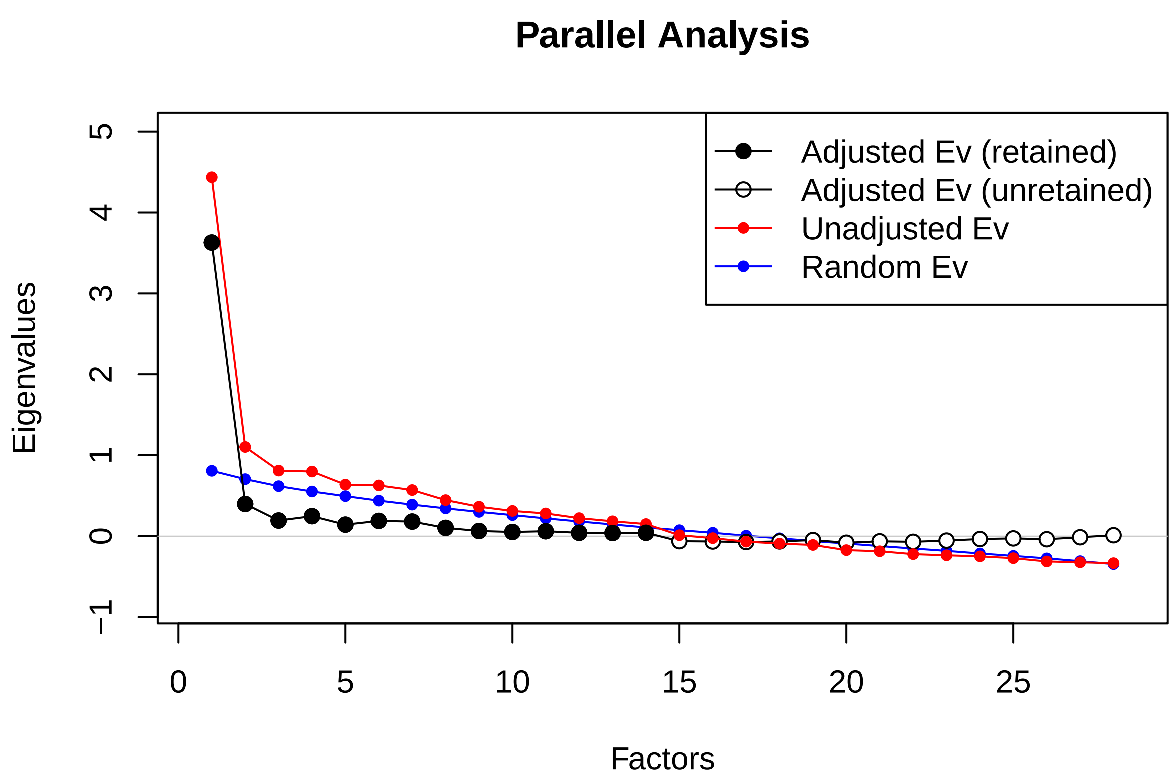
EFA

In this case, only the data for the ABCD sample changes (reduced from 21 –> 1 site [UM]). Hence, rerunning only the ABCD EFA. Factor congruence estimation will use EFA from above for AHRB and MLS.

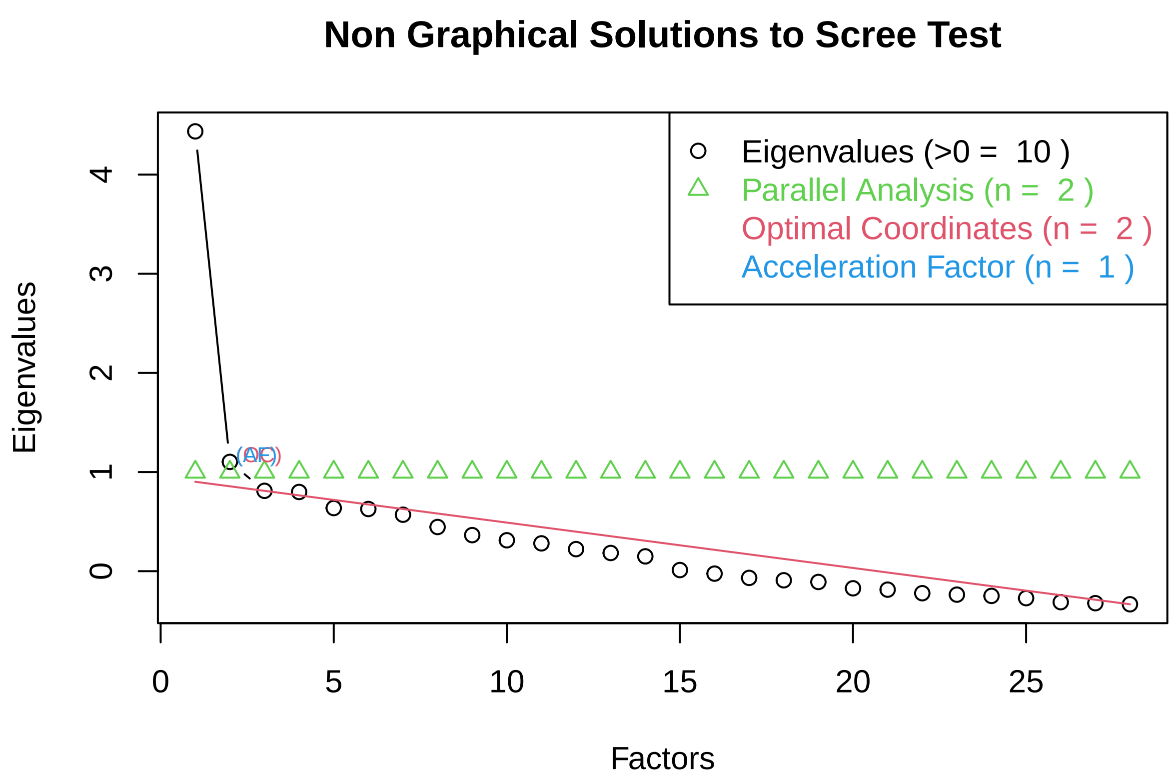
```
UM_abcd_efadata = subset(UM_site_spec[,c("AWin_v_Neut_L_NAcc_run1"  ,"AWin_v_Neut_L_NAcc_run2" ,
                         "BWin_v_Neut_L_NAcc_run1"  ,"BWin_v_Neut_L_NAcc_run2" ,
                         "BWin_v_BLose_L_NAcc_run1" ,"BWin_v_BLose_L_NAcc_run2",
                          "AWin_v_Neut_R_NAcc_run1" , "AWin_v_Neut_R_NAcc_run2",
                          "BWin_v_Neut_R_NAcc_run1" , "BWin_v_Neut_R_NAcc_run2",
                          "BWin_v_BLose_R_NAcc_run1", "BWin_v_BLose_R_NAcc_run2",
                         # insula values apprach
                         "AWin_v_Neut_R_Insula_run1","AWin_v_Neut_R_Insula_run2",
                         "BWin_v_Neut_R_Insula_run1","BWin_v_Neut_R_Insula_run2",
                         # avoidance
                         "ALose_v_Neut_L_Insula_run1","ALose_v_Neut_L_Insula_run2",
                         "BLose_v_Neut_L_Insula_run1","BLose_v_Neut_L_Insula_run2",
                         "BLose_v_BWin_L_Insula_run1","BLose_v_BWin_L_Insula_run2",
                         "ALose_v_Neut_R_Insula_run1","ALose_v_Neut_R_Insula_run2",
                         "BLose_v_Neut_R_Insula_run1","BLose_v_Neut_R_Insula_run2",
                         "BLose_v_BWin_R_Insula_run1","BLose_v_BWin_R_Insula_run2",
                         "set")] %>% filter(set==1))
paran(x = UM_abcd_efadata[,1:28],
      iterations = 1000, quietly = FALSE, centile = 95,
      status = FALSE, all = TRUE, cfa = TRUE, graph = TRUE, color = TRUE,
      col = c("black", "red", "blue"), lty = c(1, 2, 3), lwd = 1, legend = TRUE,
      seed = 100)
```

```
##
## Using eigendecomposition of correlation matrix.
##
## Results of Horn's Parallel Analysis for factor retention
## 1000 iterations, using the 95 centile estimate
##
## --------------------------------------------------
## Factor      Adjusted    Unadjusted    Estimated
##             Eigenvalue  Eigenvalue    Bias
## --------------------------------------------------
## No components passed.
## --------------------------------------------------
## 1           3.628570    4.436251      0.807681
## 2           0.397564    1.102327      0.704762
## 3           0.192784    0.810407      0.617622
## 4           0.246962    0.798866      0.551903
## 5           0.142275    0.636878      0.494602
## 6           0.188467    0.627145      0.438678
## 7           0.179993    0.569582      0.389589
```

```
## 8           0.102263    0.445540      0.343277
## 9           0.063590    0.363779      0.300188
## 10          0.051445    0.311718      0.260272
## 11          0.060812    0.280569      0.219757
## 12          0.041416    0.222636      0.181219
## 13          0.038991    0.183582      0.144591
## 14          0.042094    0.149378      0.107284
## 15         -0.062117    0.011462      0.073580
## 16         -0.066274   -0.02475      0.041519
## 17         -0.073236   -0.06771      0.005522
## 18         -0.065103   -0.09174     -0.02664
## 19         -0.048208   -0.10881     -0.06060
## 20         -0.081346   -0.17265     -0.09130
## 21         -0.063693   -0.18635     -0.12266
## 22         -0.069725   -0.22210     -0.15238
## 23         -0.054965   -0.23672     -0.18176
## 24         -0.035486   -0.24929     -0.21380
## 25         -0.027979   -0.27179     -0.24381
## 26         -0.038075   -0.31251     -0.27444
## 27         -0.013812   -0.32266     -0.30885
## 28          0.011275   -0.33361     -0.34489
## --------------------------------------------------
##
## Adjusted eigenvalues > 0 indicate dimensions to retain.
## (14 factors    retained)
```



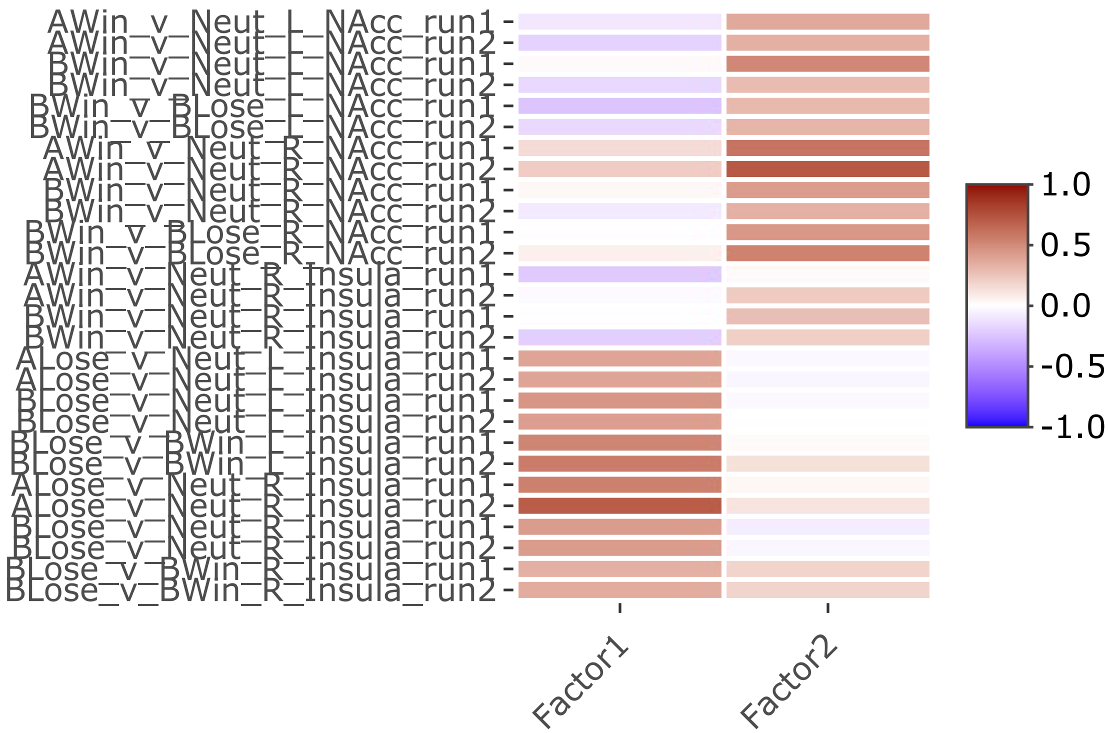
```r
plot(nScree(x=UM_abcd_efadata[,1:28],model="factors"))
```




```r
UM_abcd_efa <- factanal(x = UM_abcd_efadata[,1:28],  #raw data
              factors = 2, rotation = "promax" # oblique rotation allow for non-orthogonal structure
              )

heatmaply(round(UM_abcd_efa$loadings[,1:2],2) %>% print(sort = T),
          scale_fill_gradient_fun = ggplot2::scale_fill_gradient2(
                 low = "blue",
                 high = "darkred",
                 space = "Lab",
                 midpoint = 0,
                 limits = c(-1, 1)
              ),
              dendrogram = "none",
              xlab = "", ylab = "",
              main = "",
              margins = c(60,100,40,20),
              grid_color = "white",
              grid_width = 0.00001,
              titleX = FALSE,
              hide_colorbar = FALSE,
              branches_lwd = 0.1,
              label_names = c("Brain:", "Feature:", "Value"),
              fontsize_row = 9, fontsize_col = 9,
              labCol = colnames(UM_abcd_efa$loadings[,1:2]),
```

```r
             labRow = rownames(UM_abcd_efa$loadings[,1:2]),
             heatmap_layers = theme(axis.line=element_blank()),

)
```

```
##                              Factor1 Factor2
## AWin_v_Neut_L_NAcc_run1        -0.10    0.36
## AWin_v_Neut_L_NAcc_run2        -0.19    0.33
## BWin_v_Neut_L_NAcc_run1         0.02    0.51
## BWin_v_Neut_L_NAcc_run2        -0.17    0.28
## BWin_v_BLose_L_NAcc_run1       -0.25    0.29
## BWin_v_BLose_L_NAcc_run2       -0.16    0.31
## AWin_v_Neut_R_NAcc_run1         0.15    0.59
## AWin_v_Neut_R_NAcc_run2         0.21    0.71
## BWin_v_Neut_R_NAcc_run1         0.03    0.42
## BWin_v_Neut_R_NAcc_run2        -0.09    0.33
## BWin_v_BLose_R_NAcc_run1       -0.01    0.44
## BWin_v_BLose_R_NAcc_run2        0.06    0.53
## AWin_v_Neut_R_Insula_run1      -0.23    0.02
## AWin_v_Neut_R_Insula_run2      -0.02    0.22
## BWin_v_Neut_R_Insula_run1      -0.01    0.27
## BWin_v_Neut_R_Insula_run2      -0.21    0.20
## ALose_v_Neut_L_Insula_run1      0.38   -0.03
## ALose_v_Neut_L_Insula_run2      0.38   -0.04
## BLose_v_Neut_L_Insula_run1      0.45   -0.03
## BLose_v_Neut_L_Insula_run2      0.41    0.00
## BLose_v_BWin_L_Insula_run1      0.52    0.02
## BLose_v_BWin_L_Insula_run2      0.56    0.13
## ALose_v_Neut_R_Insula_run1      0.54    0.03
## ALose_v_Neut_R_Insula_run2      0.69    0.11
## BLose_v_Neut_R_Insula_run1      0.42   -0.08
## BLose_v_Neut_R_Insula_run2      0.42   -0.04
## BLose_v_BWin_R_Insula_run1      0.33    0.18
## BLose_v_BWin_R_Insula_run2      0.35    0.18
```




```r
fa.congruence(x = list(UM_abcd_efa, mls_efa, ahrb_efa), digits = 2) %>%
  knitr::kable(
    col.names = c("1. ABCD F1", "2. ABCD F2",
                  "3. MLS F1", "4. MLS F2","5. MLS F3",
                  "6. AHRB F1", "7. AHRB F2","8. AHRB F3", "8. AHRB F4"),
    caption = "ABCD, MLS and AHRB EFA Factor Congruence",
    booktabs = TRUE
    )
```

Table 4: ABCD, MLS and AHRB EFA Factor Congruence

	1. ABCD F1	2. ABCD F2	3. MLS F1	4. MLS F2	5. MLS F3	6. AHRB F1	7. AHRB F2	8. AHRB F3	8. AHRB F4
Factor1	1.00	0.05	-0.10	0.79	0.44	0.66	-0.25	0.06	0.69
Factor2	0.05	1.00	0.89	0.12	0.14	-0.15	0.57	0.48	0.08
Factor1	-0.10	0.89	1.00	0.07	0.15	-0.21	0.68	0.38	-0.01
Factor2	0.79	0.12	0.07	1.00	0.05	0.51	-0.07	0.02	0.67
Factor3	0.44	0.14	0.15	0.05	1.00	0.37	0.13	0.00	0.34
Factor1	0.66	-0.15	-0.21	0.51	0.37	1.00	-0.06	0.17	0.22
Factor2	-0.25	0.57	0.68	-0.07	0.13	-0.06	1.00	-0.12	-0.09
Factor3	0.06	0.48	0.38	0.02	0.00	0.17	-0.12	1.00	-0.13
Factor4	0.69	0.08	-0.01	0.67	0.34	0.22	-0.09	-0.13	1.00

Resampling ABCD CFA/EFA

```
abcd_resamp_df <- subset(brain_set[,c("AWin_v_Neut_L_NAcc_run1"  ,"AWin_v_Neut_L_NAcc_run2" ,
                        "BWin_v_Neut_L_NAcc_run1"  ,"BWin_v_Neut_L_NAcc_run2" ,
                        "BWin_v_BLose_L_NAcc_run1" ,"BWin_v_BLose_L_NAcc_run2",
                         "AWin_v_Neut_R_NAcc_run1" , "AWin_v_Neut_R_NAcc_run2",
                         "BWin_v_Neut_R_NAcc_run1" , "BWin_v_Neut_R_NAcc_run2",
                         "BWin_v_BLose_R_NAcc_run1", "BWin_v_BLose_R_NAcc_run2",
                        # insula values apprach
                        "AWin_v_Neut_R_Insula_run1","AWin_v_Neut_R_Insula_run2",
                        "BWin_v_Neut_R_Insula_run1","BWin_v_Neut_R_Insula_run2",
                        # avoidance
                        "ALose_v_Neut_L_Insula_run1","ALose_v_Neut_L_Insula_run2",
                        "BLose_v_Neut_L_Insula_run1","BLose_v_Neut_L_Insula_run2",
                        "BLose_v_BWin_L_Insula_run1","BLose_v_BWin_L_Insula_run2",
                        "ALose_v_Neut_R_Insula_run1","ALose_v_Neut_R_Insula_run2",
                        "BLose_v_Neut_R_Insula_run1","BLose_v_Neut_R_Insula_run2",
                        "BLose_v_BWin_R_Insula_run1","BLose_v_BWin_R_Insula_run2",
                        "set")] %>% filter(set==1))
```

For resampling of the CFA model, using bootstrapLavaan for lavaan. First fitting the cfa model, then creating 1000 bootstrapped sampled and extracting fit statistics

```
set.seed(1111)
boot_cfa <- cfa(model = MID_model, data = abcd_resamp_df,
                estimator = "MLR", std.lv = TRUE, meanstructure = TRUE)

out_cfaboot <- bootstrapLavaan(object = boot_cfa, R = 1000,
                               FUN = fitMeasures,
                               fit.measures=c("chisq","rmsea","cfi",
                                              "tli","srmr","AIC","BIC"),
                               parallel="multicore", ncpus=4)

cfaboot_df <- data.frame(out_cfaboot)
```

For the resampling of the EFA model, using sample_n to get [1000] samples of N = 1000 (w/ replacement) of ABCD data. Then, estimating recommended N of factors per paralell analysis. Saving this to dataframe

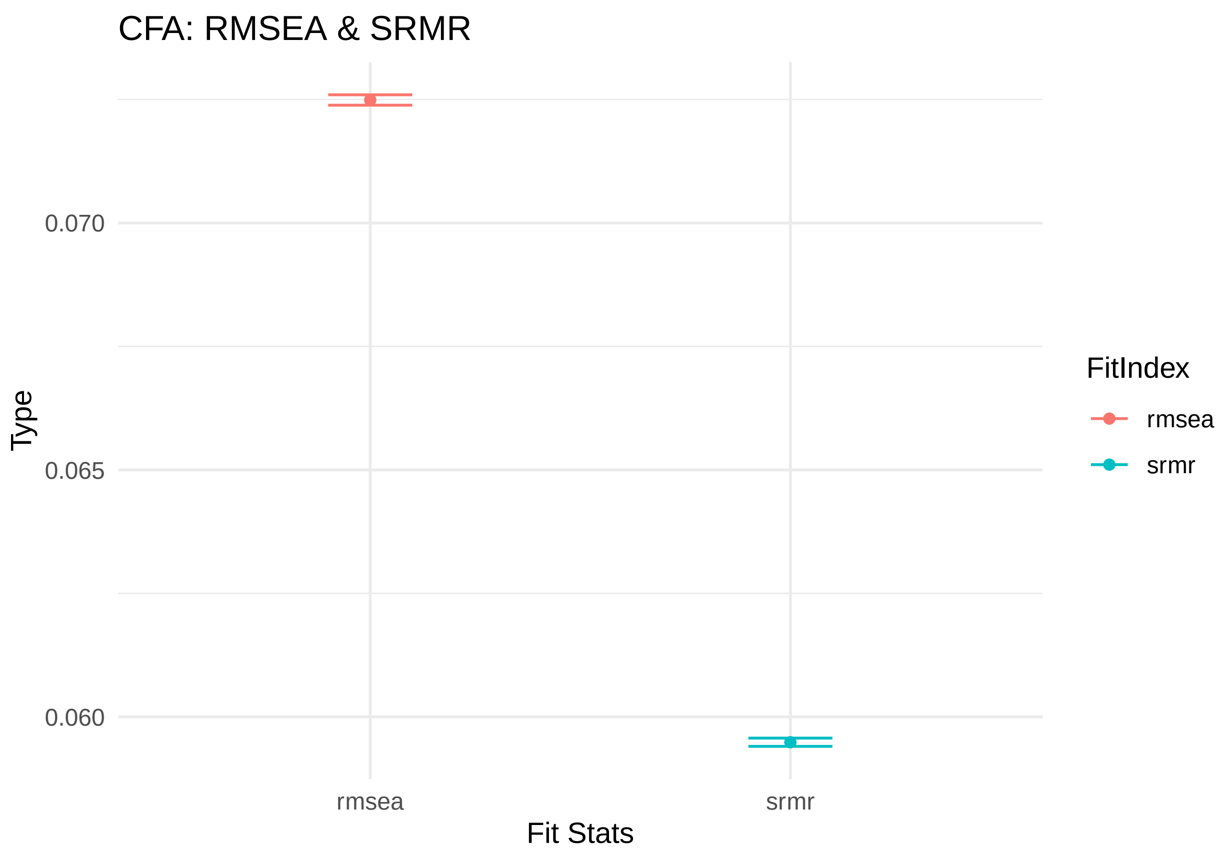
```
# create empty df where data will be saved and compiled for models
#resampled_cfa <- matrix(NA, ncol = 11, nrow = 50) # resampling 50x for Config + #Metric models
#colnames(resampled_cfa) <- c("Sample",
#                              "chisq","df","pvalue",
#                              "rmsea","cfi","tli","srmr","AIC","BIC","Factors")
#
#n_loop = 50

resampled_cfa <- matrix(NA, ncol = 2, nrow = 1000) # resampling 50x for Config + Metric models
colnames(resampled_cfa) <- c("Sample","Factors")

samples = 1000

for (s in 1:samples) {

  sub_df <- sample_n(tbl = abcd_resamp_df, size = 1000, replace = TRUE)
```

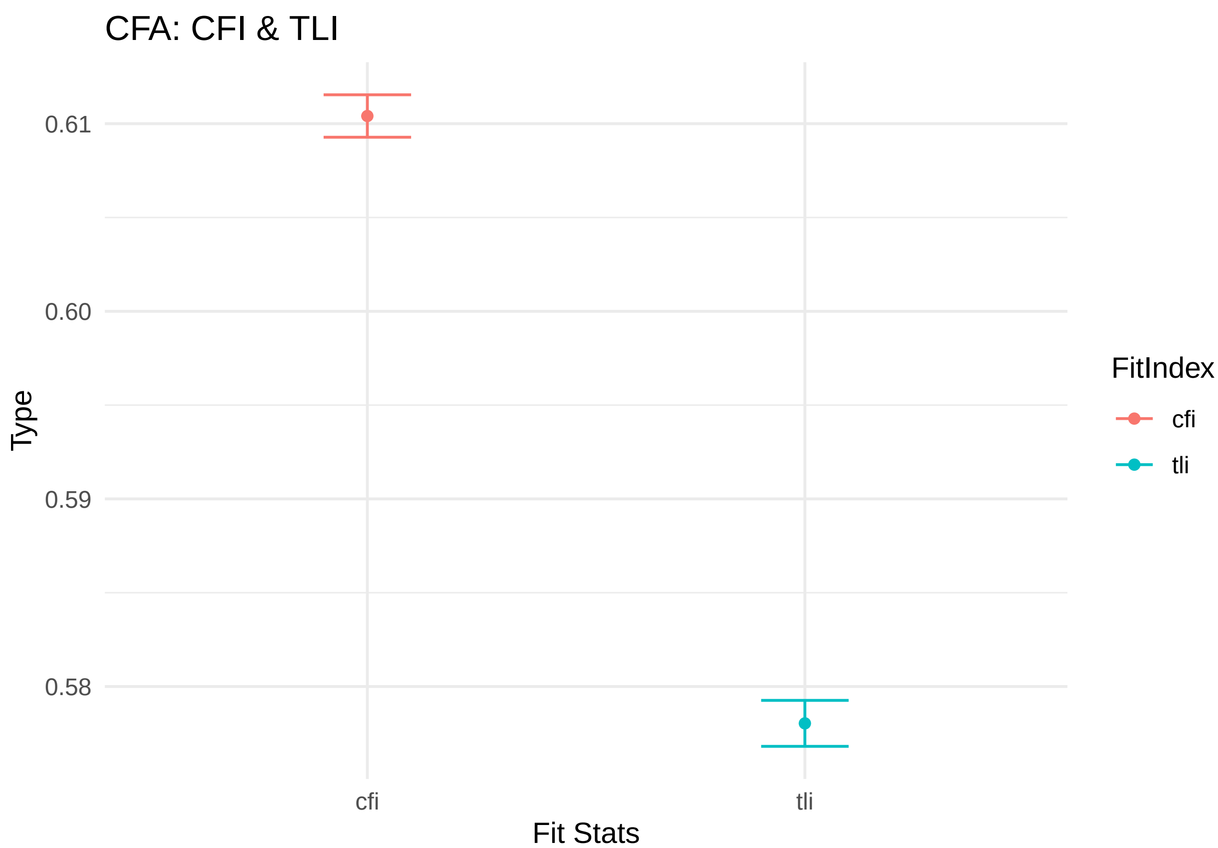
```
  resampled_cfa[s,1] <- s

  val <- nScree(x=sub_df[,1:28],model="factors")
  resampled_cfa[s,2] <- as.integer(val$Components[3])



}

resampled_res <- data.frame(resampled_cfa)
```

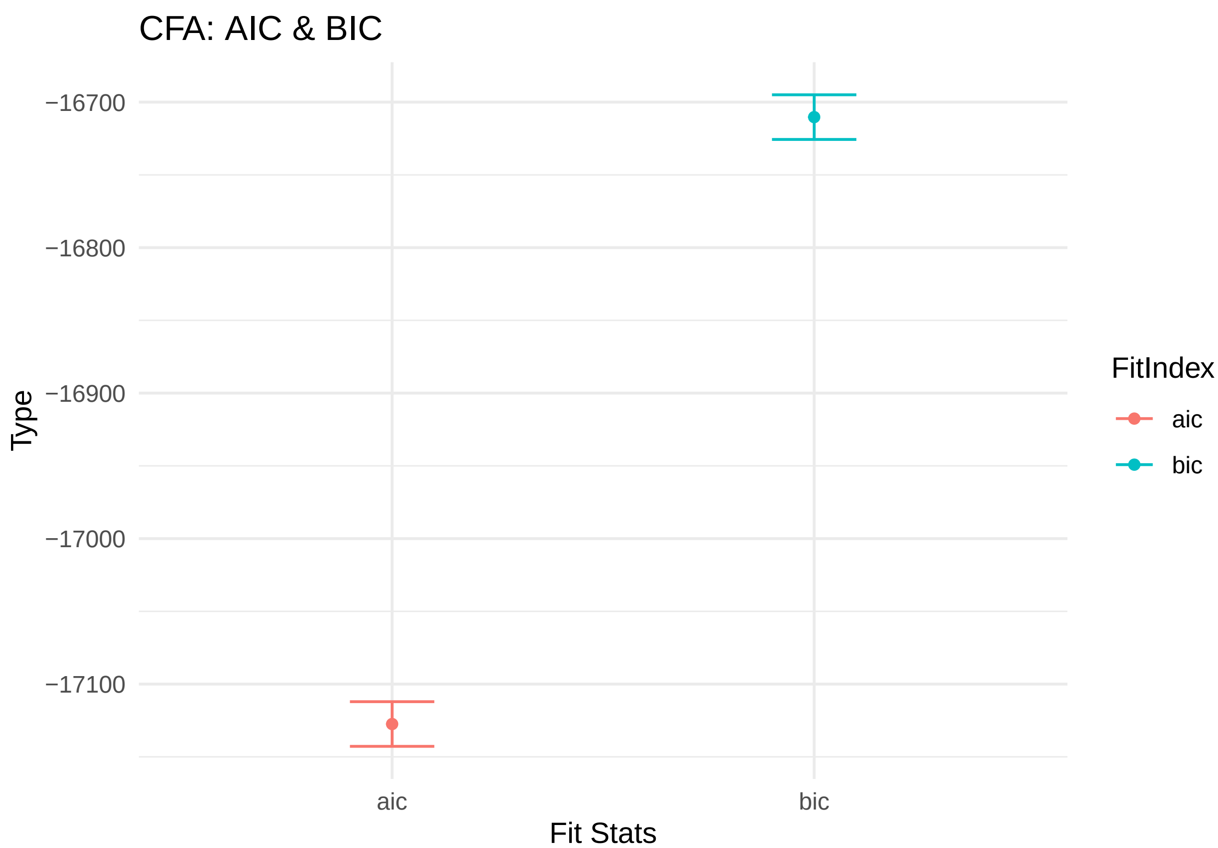
Plotting the mean + 95% Confidence interval of values

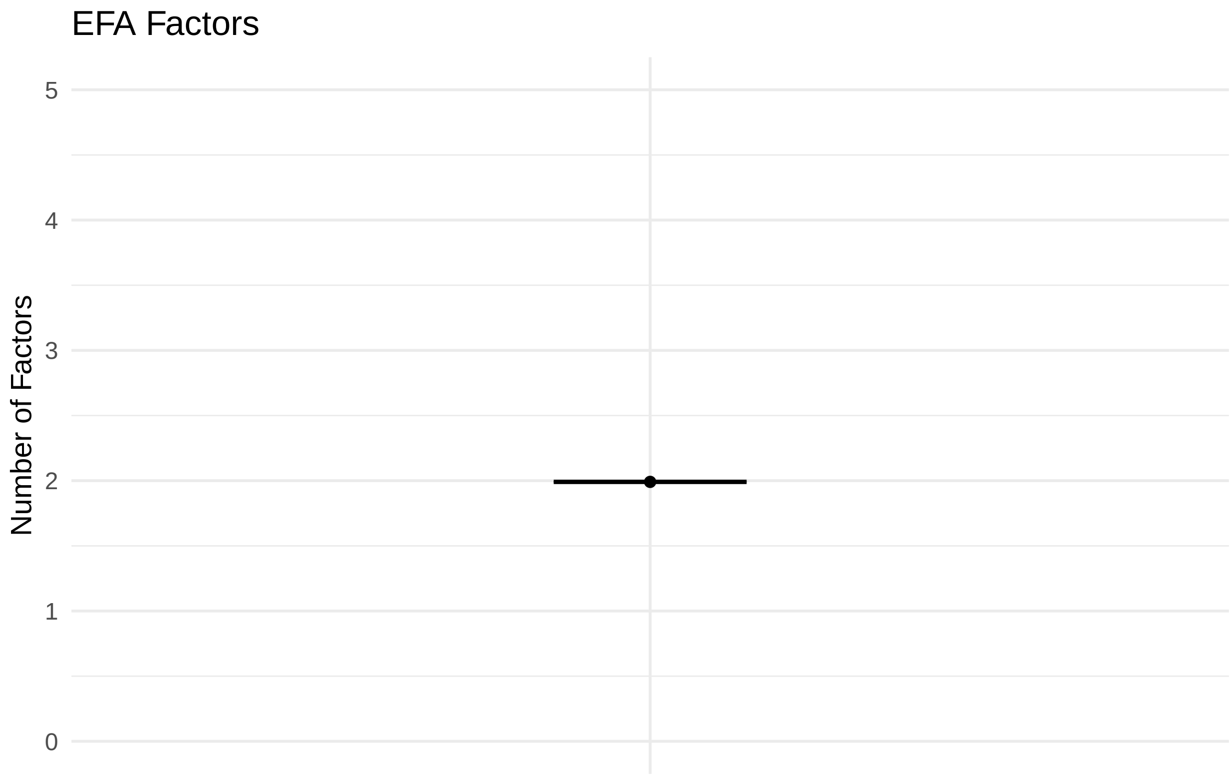
```
n = 1000

ci_plt1 <- cfaboot_df %>%
  gather(key = "FitIndex", value = "Statistic",
         srmr,rmsea) %>%
  dplyr::group_by(FitIndex) %>%
  dplyr::summarize(m = mean(Statistic), stdev = sd(Statistic)) %>%
  ggplot(aes(x =FitIndex, y = m, fill = FitIndex, color=FitIndex)) +
  geom_point()+
  geom_errorbar(aes(ymin=m-(1.96*stdev/sqrt(n)),
                    ymax=m+(1.96*stdev/sqrt(n))),
                width=.2,
                position=position_dodge(0.05))+
  labs(
    title = 'CFA: RMSEA & SRMR',
    x = 'Fit Stats',
    y = 'Type',
  )+
  theme_minimal()


ci_plt2 <- cfaboot_df %>%
  gather(key = "FitIndex", value = "Statistic",
         cfi,tli) %>%
  dplyr::group_by(FitIndex) %>%
  dplyr::summarize(m = mean(Statistic), stdev = sd(Statistic)) %>%
  ggplot(aes(x =FitIndex, y = m, fill = FitIndex, color=FitIndex)) +
  geom_point()+
  geom_errorbar(aes(ymin=m-(1.96*stdev/sqrt(n)),
                    ymax=m+(1.96*stdev/sqrt(n))),
                width=.2,
                position=position_dodge(0.05))+
  labs(
    title = 'CFA: CFI & TLI',
    x = 'Fit Stats',
    y = 'Type',
  )+
  theme_minimal()


ci_plt3 <- cfaboot_df %>%
```

```
  gather(key = "FitIndex", value = "Statistic",
         aic,bic) %>%
  dplyr::group_by(FitIndex) %>%
  dplyr::summarize(m = mean(Statistic), stdev = sd(Statistic)) %>%
  ggplot(aes(x =FitIndex, y = m, fill = FitIndex, color=FitIndex)) +
  geom_point()+
  geom_errorbar(aes(ymin=m-(1.96*stdev/sqrt(n)),
                    ymax=m+(1.96*stdev/sqrt(n))),
                width=.2,
                position=position_dodge(0.05))+
  labs(
    title = 'CFA: AIC & BIC',
    x = 'Fit Stats',
    y = 'Type',
  )+
  theme_minimal()


ci_plt4 = resampled_res %>%
  dplyr::summarize(m = mean(Factors), stdev = sd(Factors)) %>%
  ggplot(aes(x ="", y = m)) +
  geom_point()+
  geom_errorbar(aes(ymin=m-(1.96*stdev/sqrt(n)),
                    ymax=m+(1.96*stdev/sqrt(n))),
                width=.2,
                position=position_dodge(0.05))+
  ylim(0,5)+
  labs(
    title = 'EFA Factors',
    x = '',
    y = 'Number of Factors',
  )+
  theme_minimal()

ci_plt1;ci_plt2;ci_plt3;ci_plt4
```


CFA: RMSEA & SRMR
0.070
0.065
0.060
Type
rmsea
srmr
Fit Stats
FitIndex
rmsea
srmr


CFA: CFI & TLI
0.61
0.60
0.59
0.58
Type
cfi
tli
Fit Stats
FitIndex
cfi
tli

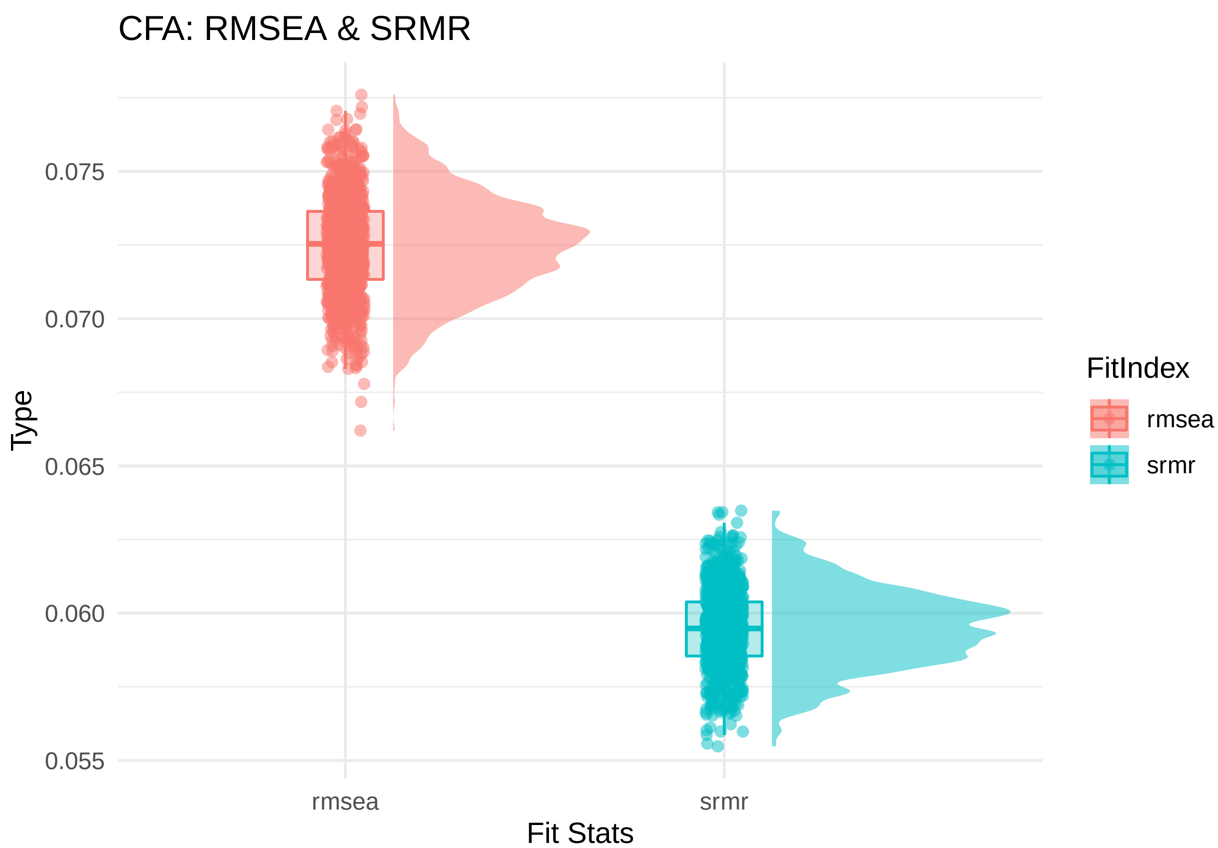

CFA: AIC & BIC
−16700
−16800
−16900
−17000
−17100
Type
aic
bic
Fit Stats
FitIndex
aic
bic

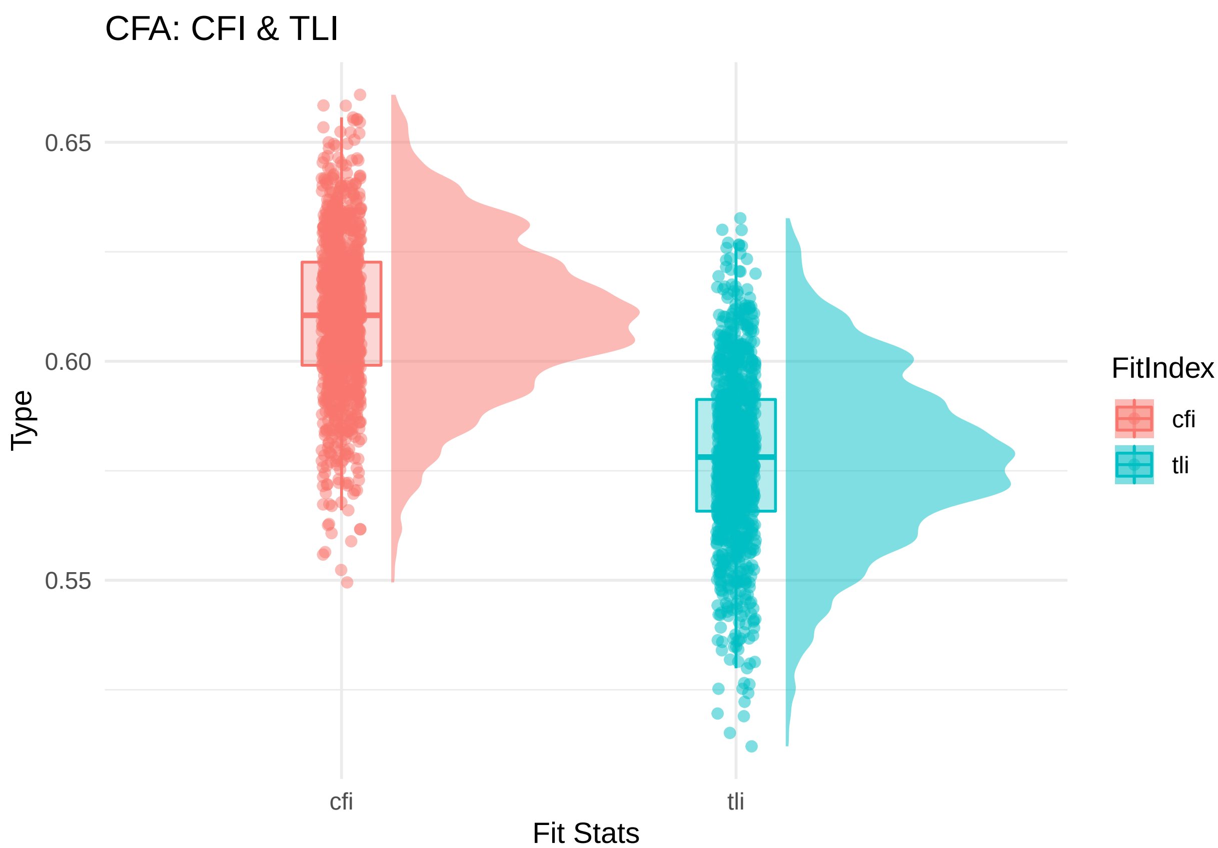


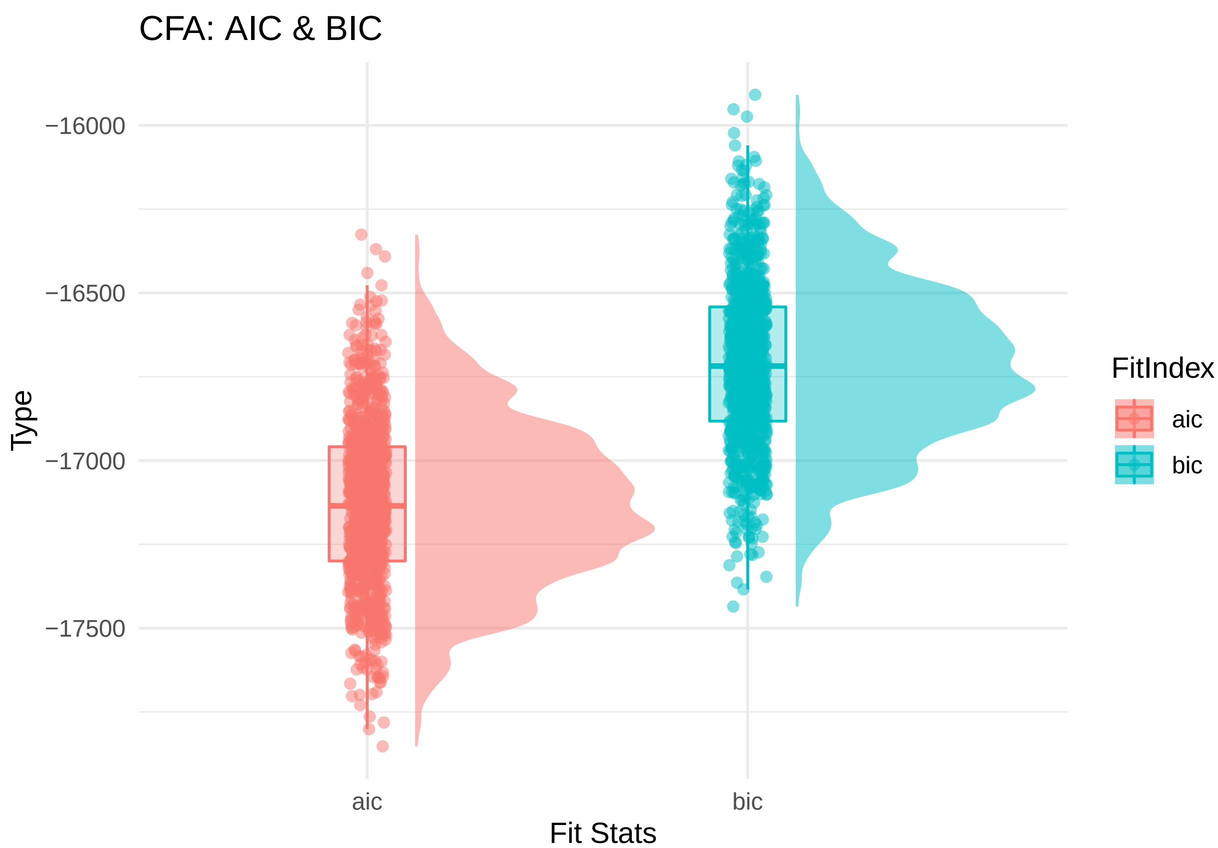
Plotting the distribution of values

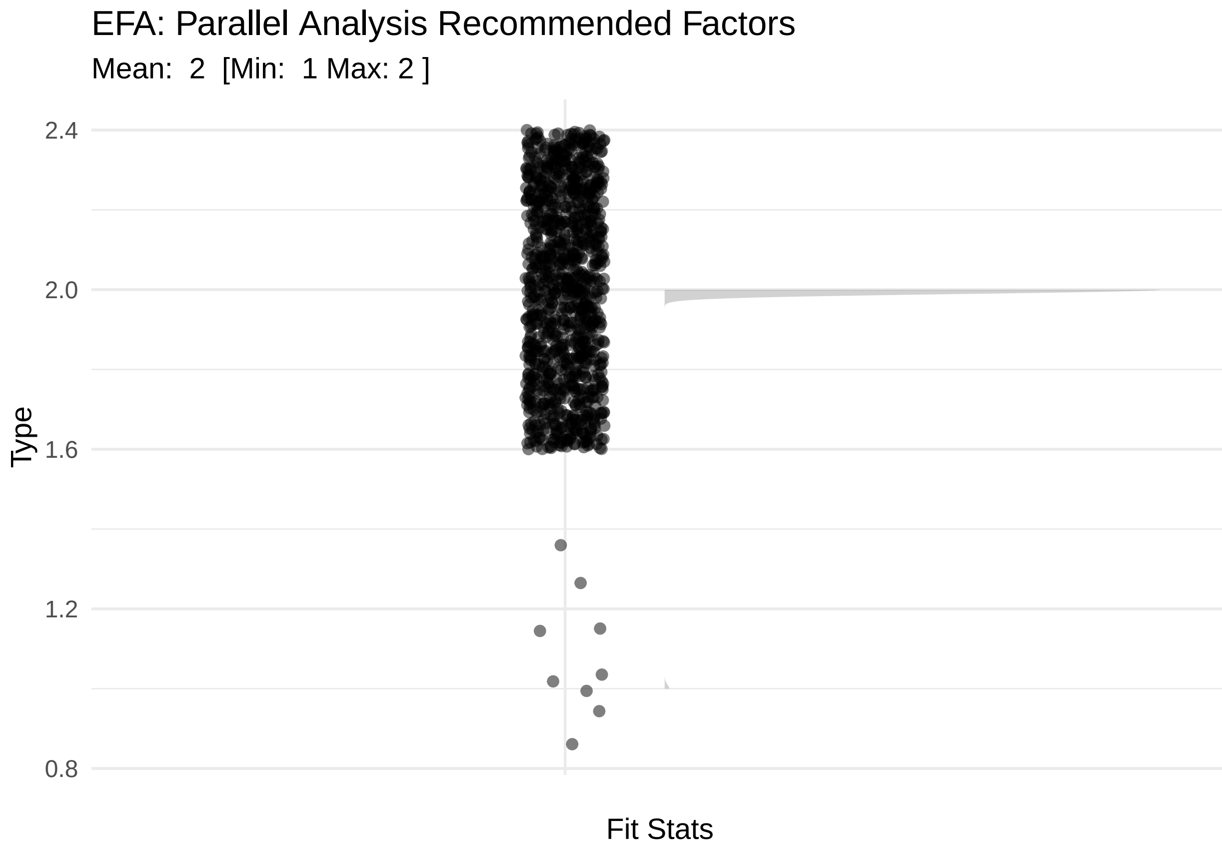
```
plt1 <- cfaboot_df %>%
  gather(key = "FitIndex", value = "Statistic",
         srmr,rmsea) %>%
  ggplot(aes(x =FitIndex, y = Statistic, fill = FitIndex, color=FitIndex)) +
  ggdist::stat_halfeye(adjust = .5, width = .7, .width = 0, justification = -.2,
                       point_colour = NA, alpha = .5) +
  geom_boxplot(width = .2, outlier.shape = NA, alpha = .3) +
  geom_jitter(width = .05, alpha = .5) +
  theme_minimal()+
  labs(
    title = 'CFA: RMSEA & SRMR',
    x = 'Fit Stats',
    y = 'Type',
  )

plt2 = cfaboot_df %>%
  gather(key = "FitIndex", value = "Statistic",
         cfi,tli) %>%
  ggplot(aes(x =FitIndex, y = Statistic, fill = FitIndex, color=FitIndex)) +
  ggdist::stat_halfeye(adjust = .5, width = .7, .width = 0, justification = -.2,
                       point_colour = NA, alpha = .5) +
  geom_boxplot(width = .2, outlier.shape = NA, alpha = .3) +
  geom_jitter(width = .05, alpha = .5) +
  theme_minimal()+
  labs(
```

```
    title = 'CFA: CFI & TLI',
    x = 'Fit Stats',
    y = 'Type',
  )


plt3 = cfaboot_df %>%
  gather(key = "FitIndex", value = "Statistic",
         aic,bic) %>%
  ggplot(aes(x =FitIndex, y = Statistic, fill = FitIndex, color=FitIndex)) +
  ggdist::stat_halfeye(adjust = .5, width = .7, .width = 0, justification = -.2,
                       point_colour = NA, alpha = .5) +
  geom_boxplot(width = .2, outlier.shape = NA, alpha = .3) +
  geom_jitter(width = .05, alpha = .5) +
  theme_minimal()+
  labs(
    title = 'CFA: AIC & BIC',
    x = 'Fit Stats',
    y = 'Type',
  )


avg = round(mean(resampled_res$Factors),1)
minimum = min(resampled_res$Factors)
maximum = max(resampled_res$Factors)

plt4 = resampled_res %>%
  ggplot(aes(x ="", y = Factors)) +
  ggdist::stat_halfeye(adjust = .5, width = .7, .width = 0, justification = -.2,
                       point_colour = NA, alpha = .5) +
  geom_jitter(width = .05, alpha = .5) +
  theme_minimal()+
  labs(
    title = 'EFA: Parallel Analysis Recommended Factors',
    subtitle = paste("Mean: ",avg," [Min: ", minimum, "Max:", maximum,"]"),
    x = 'Fit Stats',
    y = 'Type',
  )

plt1;plt2;plt3;plt4
```


CFA: RMSEA & SRMR
Type
0.075
0.070
0.065
0.060
0.055
rmsea
srmr
Fit Stats
FitIndex
rmsea
srmr


CFA: CFI & TLI
0.65
0.60
0.55
Type
cfi
tli
Fit Stats
FitIndex
cfi
tli


CFA: AIC & BIC
−16000
−16500
−17000
−17500
Type
aic
bic
Fit Stats
FitIndex
aic
bic


EFA: Parallel Analysis Recommended Factors
Mean: 2 [Min: 1 Max: 2]
Type
2.4
2.0
1.6
1.2
0.8
Fit Stats