Solution for the assignment of the fourth class

Kovacs Marton

9/29/2021

Introduction

We run a study with my colleagues a few years ago that measured the mistakes that university student commit in their everyday life among some well know decision making skills that can affect the number of mistakes they experience. The study was never published but I will use its data for this assignment. The survey that we developed for measuring the mistakes called behavioral mistakes questionnaire (BAQ). We run an exploratory factor analysis before that identified two main factors behind the mistakes, planning error and inattention. We than calculated the mean frequency of committing a mistake by each factor for each participant.

Importing data

```r
# Descriptives
baq_desc <- read_csv("data/final_task_data/Study1_BAQ_Descriptives_Processed.csv")
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   id = col_double(),
##   age = col_double(),
##   sex = col_character(),
##   story_id = col_double(),
##   test_type = col_character(),
##   story = col_character(),
##   mist = col_double(),
##   sit = col_double(),
##   Factors = col_character()
## )
```

```r
# All the measures
baq_ind <- read_csv("data/final_task_data/Study1_Individual_Processed.csv")
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   id = col_double(),
##   age = col_double(),
```

```
##   sex = col_character(),
##   ses_status = col_double(),
##   BAQ_all = col_double(),
##   Inattention = col_double(),
##   PlanningError = col_double(),
##   crt_all = col_double(),
##   Raven_all = col_double(),
##   aot_all = col_double(),
##   sds_all = col_double()
## )
```

Data exploration

For the descriptives

```
skimr::skim(baq_desc) %>%
  kable()
```

skim_type	skim_variable	n_missing	complete_rate	character.min	character.max	character.empty	character.n_unique	character.whitespace	numeric.mean	numeric.sd	numeric.p0	numeric.p25	numeric.p50	numeric.p75	numeric.p100	numeric.hist
character	sex	0	1.0000000	2	5	0	2	0	NA	NA	NA	NA	NA	NA	NA	NA
character	test_type	0	1.0000000	3	3	0	1	0	NA	NA	NA	NA	NA	NA	NA	NA
character	story	0	1.0000000	13	58	0	27	0	NA	NA	NA	NA	NA	NA	NA	NA
character	Factors	0	1.0000000	7	13	0	3	0	NA	NA	NA	NA	NA	NA	NA	NA
numeric	id	0	1.0000000	NA	NA	NA	NA	NA	122.5000000	70.4401487	1	61.75	122.5	183.25	244	<U+2587><U+2587><U-
numeric	age	0	1.0000000	NA	NA	NA	NA	NA	21.3770492	2.3829326	18	20.00	21.0	22.00	42	<U+2587><U+2582><U-
numeric	story_id	0	1.0000000	NA	NA	NA	NA	NA	14.0000000	7.7894722	1	7.00	14.0	21.00	27	<U+2587><U+2587><U-
numeric	mist	561	0.9148452	NA	NA	NA	NA	NA	1.7303890	0.9451316	1	1.00	1.0	2.00	5	<U+2587><U+2585><U-
numeric	sit	0	1.0000000	NA	NA	NA	NA	NA	0.9148452	0.2791331	0	1.00	1.0	1.00	1	<U+2581><U+2581><U-

The mean age for the participants was 21.4 with an SD of 2.38.

Gender descriptives:

```
baq_desc %>% distinct(id, .keep_all = TRUE) %>% count(sex)
```

```
## # A tibble: 2 x 2
##   sex       n
##   <chr> <int>
## 1 Férfi    51
## 2 No      193
```

Number of mistakes that were assigned to factors:

```
baq_desc %>% filter(Factors != "Dropped") %>% distinct(story_id, .keep_all = T) %>% nrow()
```

```
## [1] 9
```

For the individual measures

```
skimr::skim(baq_ind) %>%
  kable()
```

skim_type	skim_variable	n_missing	complete_rate	character.min	character.max	character.empty	character.n_unique	character.whitespace	numeric.mean	numeric.sd	numeric.p0	numeric.p25	numeric.p50	numeric.p75	numeric.p100	numeric.hist
character	sex	0	1	2	5	0	2	0	NA	NA	NA	NA	NA	NA	NA	NA
numeric	id	0	1	NA	NA	NA	NA	NA	122.500	70.58091228	[illegible]	61.7500	122.500	183.25	244.00	<U+2587><U+2587><U-
numeric	age	0	1	NA	NA	NA	NA	NA	21.3772	2.438764895	[illegible]	20.0000	21.000	22.00	42.00	<U+2587><U+2582><U-
numeric	ses_status	0	1	NA	NA	NA	NA	NA	7.05731	1.6395281	[illegible]	6.0000	7.000	8.00	10.00	<U+2581><U+2581><U-
numeric	BAQ_all	0	1	NA	NA	NA	NA	NA	2.05500	0.5641662	[illegible]	1.6575	1.980	2.35	4.00	<U+2585><U+2587><U-
numeric	Inattention	0	1	NA	NA	NA	NA	NA	1.66893	0.4581359	[illegible]	1.2500	1.600	2.00	4.00	<U+2587><U+2583><U-
numeric	PlanningError	0	1	NA	NA	NA	NA	NA	2.44131	0.7909675	[illegible]	1.7500	2.375	3.00	4.67	<U+2583><U+2587><U-
numeric	crt_all	0	1	NA	NA	NA	NA	NA	1.41803	1.3956035	[illegible]	0.0000	1.000	3.00	3.00	<U+2587><U+2585><U-
numeric	Raven_all	0	1	NA	NA	NA	NA	NA	6.44672	2.1271461	[illegible]	4.0000	6.000	9.00	12.00	<U+2583><U+2583><U-
numeric	aot_all	0	1	NA	NA	NA	NA	NA	55.7540	6.9880662	[illegible]	51.0000	56.000	61.00	74.00	<U+2581><U+2585><U-
numeric	sds_all	0	1	NA	NA	NA	NA	NA	89.9959	13.8734723	[illegible]	81.0000	90.000	100.00	128.00	<U+2581><U+2583><U-

Other tests that were used: * socialeconomic status * BAQ * Inattention subscale * Planning error subscale * CRT * Raven test * AOT * SDS

Lets check out the correlations between the individual measures

```
cor_data <-
  baq_ind %>%
  select(-id, -sex)

Hmisc::rcorr(as.matrix(cor_data), type = "spearman")
```

```
##                 age ses_status BAQ_all Inattention PlanningError crt_all
## age            1.00      -0.01   -0.08       -0.07         -0.04    0.24
## ses_status    -0.01       1.00   -0.06       -0.03         -0.07   -0.01
## BAQ_all       -0.08      -0.06    1.00        0.69          0.89    0.01
## Inattention   -0.07      -0.03    0.69        1.00          0.31   -0.13
## PlanningError -0.04      -0.07    0.89        0.31          1.00    0.09
## crt_all        0.24      -0.01    0.01       -0.13          0.09    1.00
## Raven_all      0.15      -0.02   -0.03       -0.18          0.09    0.42
## aot_all        0.06       0.02   -0.05       -0.11          0.01    0.32
## sds_all        0.02       0.05   -0.26       -0.20         -0.20    0.03
##               Raven_all aot_all sds_all
## age                0.15    0.06    0.02
## ses_status        -0.02    0.02    0.05
## BAQ_all           -0.03   -0.05   -0.26
## Inattention       -0.18   -0.11   -0.20
## PlanningError      0.09    0.01   -0.20
## crt_all            0.42    0.32    0.03
## Raven_all          1.00    0.34    0.09
## aot_all            0.34    1.00    0.09
```

```
## sds_all            0.09    0.09    1.00
##
## n= 244
##
##
## P
##               age    ses_status BAQ_all Inattention PlanningError crt_all
## age                  0.9279     0.2324  0.3087      0.5104        0.0002
## ses_status    0.9279            0.3227  0.6381      0.2758        0.8481
## BAQ_all       0.2324 0.3227             0.0000      0.0000        0.9246
## Inattention   0.3087 0.6381     0.0000              0.0000        0.0461
## PlanningError 0.5104 0.2758     0.0000  0.0000                    0.1441
## crt_all       0.0002 0.8481     0.9246  0.0461      0.1441
## Raven_all     0.0180 0.7171     0.6240  0.0049      0.1482        0.0000
## aot_all       0.3297 0.7361     0.4375  0.0987      0.8820        0.0000
## sds_all       0.8155 0.4102     0.0000  0.0020      0.0014        0.6198
##               Raven_all aot_all sds_all
## age           0.0180    0.3297  0.8155
## ses_status    0.7171    0.7361  0.4102
## BAQ_all       0.6240    0.4375  0.0000
## Inattention   0.0049    0.0987  0.0020
## PlanningError 0.1482    0.8820  0.0014
## crt_all       0.0000    0.0000  0.6198
## Raven_all               0.0000  0.1588
## aot_all       0.0000            0.1446
## sds_all       0.1588    0.1446
```

There are 244 participants.

There is a medium correlation between the subscales of the BAQ test with $r = 0.31$.

Interestingly BAQ summarized score does not have a correlation with any other measures. It seems like these measures do not have a relationship with the frequency of commiting mistakes in the everyday life.

However, Inattention has a small negative correlation with IQ with $r =$ -0.18. Lets check out the significance.

```r
cor.test(cor_data$Inattention, cor_data$Raven_all, method = "spearman")
```

```
## Warning in cor.test.default(cor_data$Inattention, cor_data$Raven_all, method =
## "spearman"): Cannot compute exact p-value with ties
```

```
##
##  Spearman's rank correlation rho
##
## data:  cor_data$Inattention and cor_data$Raven_all
## S = 2855973, p-value = 0.004888
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##       rho
## -0.179623
```

Participants with higher IQ commit less mistakes in their everyday life.

And both subscales correlate moderately negatively with the SDS scale that measures whether the participant is likely to lie on these questionnaire with $r =$ -.20. So participants who tend to modify their scores to make a better impression about themselves give lower frequency scores for committing mistakes.

Lets calculate a Cronbachs alpha for the two subscales of BAQ.

First we have to get the individual frequency scores for the different mistakes for each subscale from the descriptive table.

Some participants did not answer to all the stories so I will drop them here.

For planning error mistakes.

```
planning_data <-
  baq_desc %>%
  filter(Factors == "PlanningError") %>%
  select(id, story_id, mist) %>%
  mutate(story_id = paste0("mistake", story_id)) %>%
  spread(key = story_id, value = mist) %>%
  filter_all(all_vars(!is.na(.))) %>%
  select(-id)
```

Number of participants left:

```
nrow(planning_data)
```

```
## [1] 233
```

Number of mistakes belonging to this factor.

```
ncol(planning_data)
```

```
## [1] 4
```

Calculating the Cronbachs alpha.

```
ltm::cronbach.alpha(planning_data, CI = TRUE)
```

```
##
## Cronbach's alpha for the 'planning_data' data-set
##
## Items: 4
## Sample units: 233
## alpha: 0.659
##
## Bootstrap 95% CI based on 1000 samples
##  2.5% 97.5%
## 0.583 0.718
```

For inattention mistakes.

```
inattention_data <-
  baq_desc %>%
  filter(Factors == "Inattention") %>%
  select(id, story_id, mist) %>%
  mutate(story_id = paste0("mistake", story_id)) %>%
  spread(key = story_id, value = mist) %>%
  filter_all(all_vars(!is.na(.))) %>%
  select(-id)
```

Number of participants left:

```
nrow(inattention_data)
```

```
## [1] 189
```

Number of mistakes belonging to this factor.

```
ncol(inattention_data)
```

```
## [1] 5
```

Calculating the Cronbachs alpha.

```
ltm::cronbach.alpha(inattention_data, CI = TRUE)
```

```
##
## Cronbach's alpha for the 'inattention_data' data-set
##
## Items: 5
## Sample units: 189
## alpha: 0.655
##
## Bootstrap 95% CI based on 1000 samples
##  2.5% 97.5%
## 0.557 0.725
```

Lets see whether there is a difference between the gender in committing different types of mistakes.

To test this I will run an ANOVA and I will look at the interaction between gender and the types of mistakes. As every participant answered multiple items from both types I am running a mixed ANOVA.

First I have to transform the data to long format.

```
anova_data <-
  baq_ind %>%
  select(id, sex, Inattention, PlanningError) %>%
  gather(key = "subscale", value = "score", -id, -sex) %>%
  arrange(id)
```

Lets run the ANOvA.

```
ezANOVA(data = anova_data, dv = score, wid = id, between = sex, within = subscale, detailed = TRUE)
```

```
## Warning: Converting "id" to factor for ANOVA.
```

```
## Warning: Converting "subscale" to factor for ANOVA.
```

```
## Warning: Converting "sex" to factor for ANOVA.
```

```
## Warning: Data is unbalanced (unequal N per group). Make sure you specified a
## well-considered value for the type argument to ezANOVA().
```

```
## $ANOVA
##          Effect DFn DFd          SSn       SSd           F              p p<.05
## 1   (Intercept)   1 242 2061.0828074 153.83508 3242.316641 3.719717e-142     *
## 2           sex   1 242    0.9724128 153.83508    1.529715  2.173541e-01
## 3      subscale   1 242   72.7810893  72.28893  243.647607  1.796390e-38     *
## 4 sex:subscale   1 242    0.6298855  72.28893    2.108653  1.477634e-01
##           ges
## 1 0.901135305
## 2 0.004281938
## 3 0.243492301
## 4 0.002777838
```

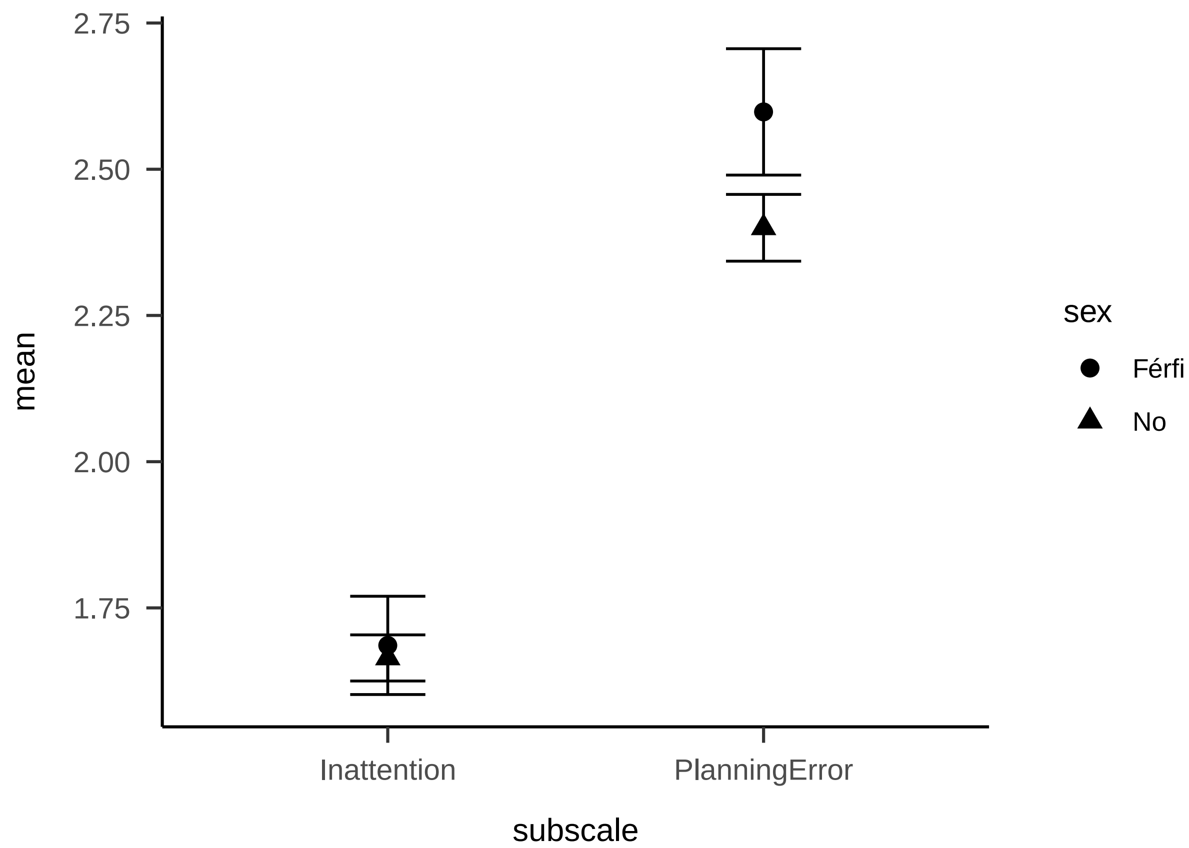
It seems like there is no interaction between gender and the different reasons for committing a mistake.

There is a significant difference between the frequency of committing a mistake by the different subscales, but not because of gender.

Lets visualize the results.

```
anova_data %>%
  group_by(sex, subscale) %>%
  summarize(
    mean = mean(score),
    sd = sd(score),
    n = n(),
    se = sd / sqrt(n)
  ) %>%
  ggplot() +
  aes(
    x = subscale,
    y = mean,
    shape = sex
  ) +
  geom_point(size = 3) +
  geom_errorbar(aes(ymin = mean - se, ymax = mean + se), width = 0.2) +
  papaja::theme_apa()
```

```
## `summarise()` has grouped output by 'sex'. You can override using the `.groups` argument.
```



The errorbar on the figure is the SE.

The difference between the genders in inattention is really small, however there is a difference for mistakes committed because of a planning error. The female participants committed less mistakes due to a planning error than the male participants.

We should run a post hoc test for the gender for the Planning error subscale.

```
post_hoc_data <-
  anova_data %>%
  filter(subscale == "PlanningError")

pairwise.t.test(post_hoc_data$score, post_hoc_data$sex, p.adj = "bonf")
```

```
##
##  Pairwise comparisons using t tests with pooled SD
##
## data:  post_hoc_data$score and post_hoc_data$sex
##
##    Férfi
## No 0.11
##
## P value adjustment method: bonferroni
```

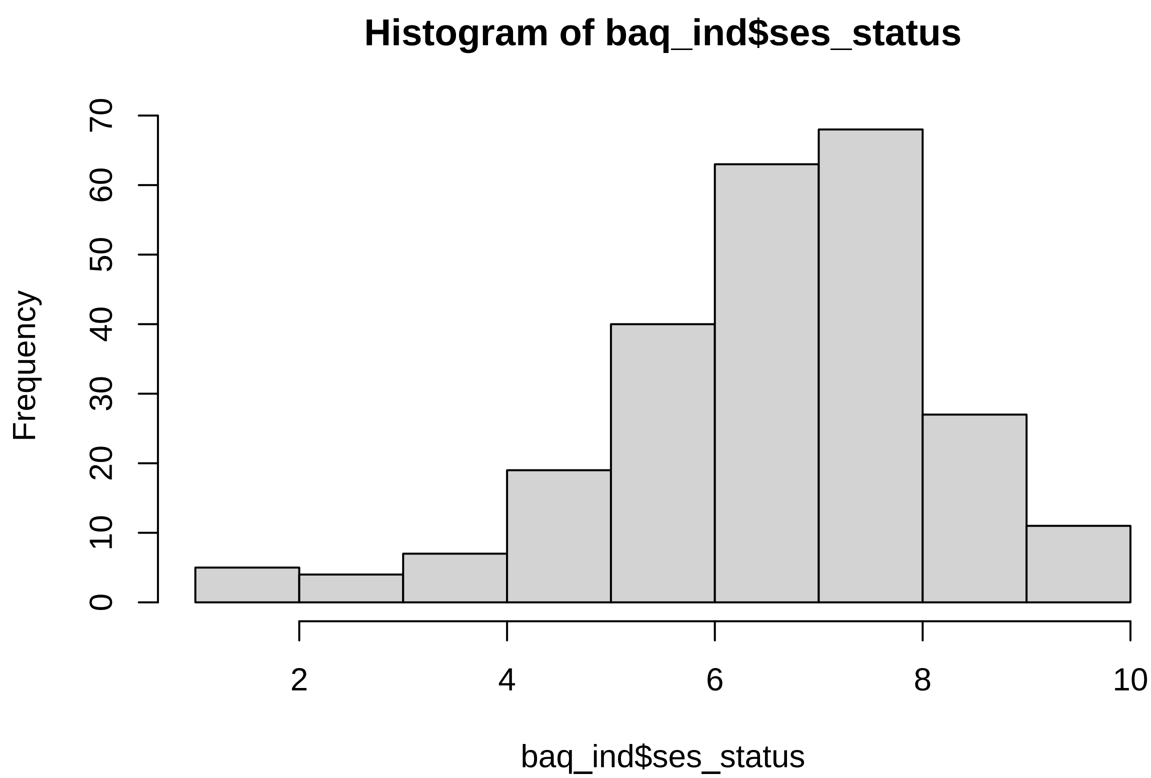
The pairwise test suggests that there is no significant difference in the frequency of commiting a mistake because of a planning error between genders.

Lets see if there is a difference between participants from different socialeconomic background in the AOT scores if we control for gender and age.

For this we group the participants with different socialeconomic background to four groups.

Lets look at the distribution of the SES scores. It ranges from 1 to 10, and with 10 meaning higher socialeconomic status.

```
hist(baq_ind$ses_status)
```



The histogram is skewed to the right. We will try to group the participants to five groups. The sample sizes will be unequal but we want to make sure that one participant only belongs to one group.

```
ses_data <-
  baq_ind %>%
  mutate(ses_group = case_when(ses_status %in% c(1, 2) ~ "g1",
                               ses_status %in% c(3, 4) ~ "g2",
                               ses_status %in% c(5, 6) ~ "g3",
                               ses_status %in% c(7, 8) ~ "g4",
                               ses_status %in% c(9, 10) ~ "g5")) %>%
  select(id, sex, age, ses_group, aot_all)
```

Lets look at the number of participants in each group.

```r
ses_data %>%
  count(ses_group)
```

```
## # A tibble: 5 x 2
##   ses_group     n
##   <chr>     <int>
## 1 g1            5
## 2 g2           11
## 3 g3           59
## 4 g4          131
## 5 g5           38
```

I will drop the first two groups because of the low response rate.

```r
`%ni%` <- Negate(`%in%`)

ses_data <-
  ses_data %>%
  filter(ses_group %ni% c("g1", "g2")) %>%
  mutate(id = as.factor(id),
         ses_group = as.factor(ses_group),
         sex = as.factor(sex))
```

Run the ANCOVA with covariates of age and gender.

```r
ezANOVA(data = ses_data, dv = aot_all, wid = id, between = ses_group, between_covariates = .(age, sex)
```

```
## Warning: Data is unbalanced (unequal N per group). Make sure you specified a
## well-considered value for the type argument to ezANOVA().

## Warning: Implementation of ANCOVA in this version of ez is experimental and
## not yet fully validated. Also, note that ANCOVA is intended purely as a tool
## to increase statistical power; ANCOVA can not eliminate confounds in the data.
## Specifically, covariates should: (1) be uncorrelated with other predictors and
## (2) should have effects on the DV that are independent of other predictors.
## Failure to meet these conditions may dramatically increase the rate of false-
## positives.

## Warning: Covariate"age" is numeric and will therefore be fit to a linear effect.

## Coefficient covariances computed by hccm()

## $ANOVA
##      Effect DFn DFd    SSn      SSd        F         p p<.05        ges
## 1 ses_group   2 225 111.98 10455.75 1.204864 0.3016592       0.01059641
##
## $`Levene's Test for Homogeneity of Variance`
##   DFn DFd      SSn      SSd          F         p p<.05
## 1   2 225 1.029283 3993.492 0.02899576 0.9714242
```

There is a not significant difference between participants from different social economic background on the Actively Open Minded scale with $F(2, 225) = 1.2$, $p = 0.3$. The eta square effect size estimate is 0.01.

CFA on a different data

Finally, we collected data from a different group of participants. I will run a CFA in these results and check out the two factor design.

Lets load this dataset.

```
baq_desc_two <- read_csv("data/final_task_data/Study2_BAQ_Descriptives.csv")
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   id = col_double(),
##   age = col_double(),
##   sex = col_character(),
##   story_id = col_double(),
##   test_type = col_character(),
##   story = col_character(),
##   decision = col_double(),
##   mist = col_double(),
##   sit = col_double(),
##   Factors = col_character()
## )
```

Exploratory data analysis

The number of participants.

```
baq_desc_two %>%
  distinct(id) %>%
  nrow()
```

```
## [1] 362
```

The number of participants is quite low for CFA.

Data transformation

I modify the data to be ready for the CFA.

```
cfa_data <-
  baq_desc_two %>%
  filter(Factors != "Dropped") %>%
  select(id, story_id, mist) %>%
  mutate(story_id = paste0("mistake", story_id)) %>%
  spread(key = story_id, value = mist) %>%
  filter_all(all_vars(!is.na(.))) %>%
  select(-id)
```

The number of participants remaining after transformation.

```
nrow(cfa_data)
```

```
## [1] 222
```

The sample size is really small now, we should keep that in mind as a limitation.

Creating the model based on the results of the FA.

Lets look at which story belongs to which factor based on the FA.

```
baq_desc_two %>%
  filter(Factors != "Dropped") %>%
  select(id, story_id, Factors) %>%
  mutate(story_id = paste0("mistake", story_id)) %>%
  rename(mistake = story_id) %>%
  distinct(Factors, mistake) %>%
  arrange(Factors)
```

```
## # A tibble: 9 x 2
##   mistake   Factors
##   <chr>     <chr>
## 1 mistake2  Inattention
## 2 mistake4  Inattention
## 3 mistake5  Inattention
## 4 mistake25 Inattention
## 5 mistake27 Inattention
## 6 mistake1  PlanningError
## 7 mistake3  PlanningError
## 8 mistake10 PlanningError
## 9 mistake17 PlanningError
```

There are 5 items in the Inattention factor and four items in the planning error factor.

```
model <- '
inattention =~ mistake2 + mistake4 + mistake5 + mistake25 + mistake27
planningerror =~ mistake1 + mistake3  + mistake10 + mistake17'
```

Running the CFA.

```
cfa_res <- cfa(model, data = cfa_data)

summary(cfa_res, standardized = TRUE, fit.measures = TRUE)
```

```
## lavaan 0.6-9 ended normally after 64 iterations
##
##   Estimator                                         ML
##   Optimization method                           NLMINB
##   Number of model parameters                        19
##
##   Number of observations                           222
##
## Model Test User Model:
```

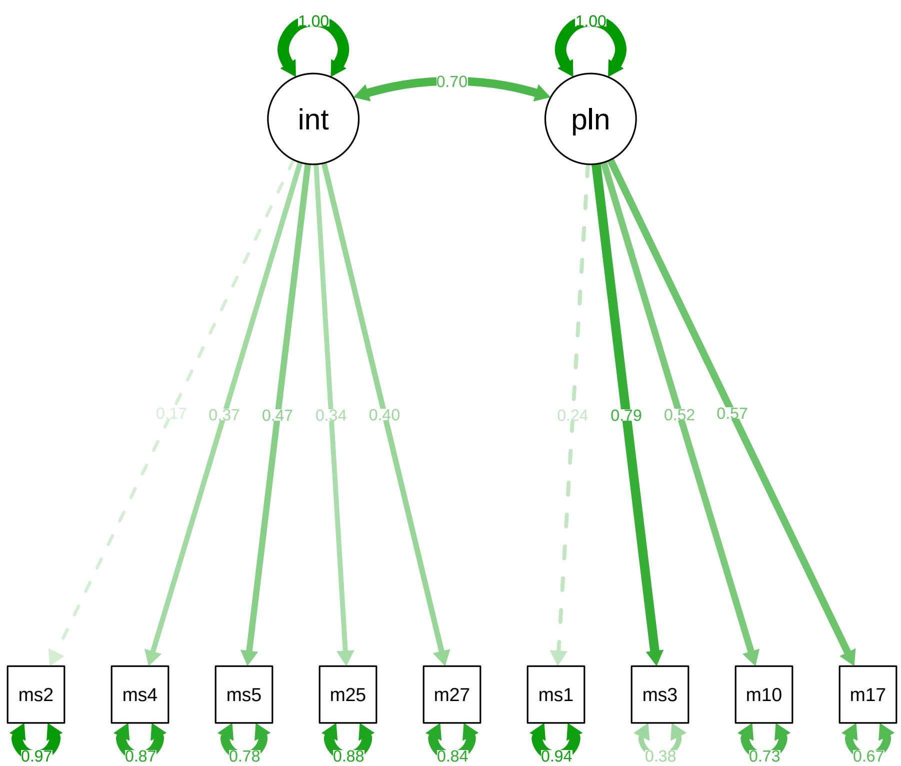
```
##
##   Test statistic                                49.091
##   Degrees of freedom                                26
##   P-value (Chi-square)                           0.004
##
## Model Test Baseline Model:
##
##   Test statistic                               223.897
##   Degrees of freedom                                36
##   P-value                                        0.000
##
## User Model versus Baseline Model:
##
##   Comparative Fit Index (CFI)                    0.877
##   Tucker-Lewis Index (TLI)                       0.830
##
## Loglikelihood and Information Criteria:
##
##   Loglikelihood user model (H0)              -2580.606
##   Loglikelihood unrestricted model (H1)      -2556.060
##
##   Akaike (AIC)                                5199.212
##   Bayesian (BIC)                              5263.862
##   Sample-size adjusted Bayesian (BIC)         5203.650
##
## Root Mean Square Error of Approximation:
##
##   RMSEA                                          0.063
##   90 Percent confidence interval - lower         0.035
##   90 Percent confidence interval - upper         0.090
##   P-value RMSEA <= 0.05                          0.196
##
## Standardized Root Mean Square Residual:
##
##   SRMR                                           0.060
##
## Parameter Estimates:
##
##   Standard errors                             Standard
##   Information                                 Expected
##   Information saturated (h1) model          Structured
##
## Latent Variables:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##   inattention =~
##     mistake2          1.000                               0.150    0.170
##     mistake4          1.715    0.980    1.749    0.080    0.258    0.367
##     mistake5          3.039    1.681    1.808    0.071    0.457    0.472
##     mistake25         1.813    1.051    1.724    0.085    0.273    0.340
##     mistake27         1.851    1.043    1.775    0.076    0.278    0.404
##   planningerror =~
##     mistake1          1.000                               0.325    0.244
##     mistake3          2.409    0.794    3.035    0.002    0.784    0.786
##     mistake10         1.802    0.612    2.945    0.003    0.586    0.522
```

```
##     mistake17         1.680    0.561    2.993    0.003    0.547    0.575
##
## Covariances:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##   inattention ~~
##     planningerror     0.034    0.021    1.596    0.111    0.700    0.700
##
## Variances:
##                    Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
##    .mistake2          0.762    0.074   10.301    0.000    0.762    0.971
##    .mistake4          0.426    0.046    9.272    0.000    0.426    0.865
##    .mistake5          0.727    0.089    8.134    0.000    0.727    0.777
##    .mistake25         0.568    0.060    9.482    0.000    0.568    0.884
##    .mistake27         0.397    0.044    8.941    0.000    0.397    0.837
##    .mistake1          1.672    0.162   10.295    0.000    1.672    0.940
##    .mistake3          0.380    0.091    4.187    0.000    0.380    0.382
##    .mistake10         0.917    0.103    8.942    0.000    0.917    0.727
##    .mistake17         0.605    0.073    8.338    0.000    0.605    0.669
##     inattention       0.023    0.024    0.959    0.337    1.000    1.000
##     planningerror     0.106    0.067    1.581    0.114    1.000    1.000
```

Both the CFI and the TLI are really small, and the RMSEA is nonsignificant, which indicates a bad fit. Howeverm the model test statistics are significant.

Lets plot the results.

```r
semPlot::semPaths(cfa_res, "std")
```



The standardized model parameter estimates are quite low!