

codingRMD_MNK

MNK

2024-08-06

Contents

1	Packages	3
2	Loading files (and predefined transformation)	3
2.1	EMC data	3
2.2	UMCG data	3
3	Data preprocessing	3
3.1	Pre defined transformation	3
3.2	Synchronizing names and compute totals	4
3.3	Creating ID values for UMCG sample	4
3.4	Data pooling	4
3.5	Renaming and var name sets	4
3.6	Missing values part 1; visualisation	4
4	Data checks	5
4.1	EMC sample data checks	5
4.1.1	Outliers - EMC	6
4.1.2	Normality - EMC	6
4.1.3	Imputing the SET_Gt variable some EMC missings due to different testing protocol .	7
4.2	UMCG sample data checks	8
4.2.1	Outliers - UMCG	8
4.2.2	Normality - UMCG	9
5	Stochastic imputation	10
5.1	Imputation predictor selection	12
5.1.1	Predictor selection ERTt	12
5.1.2	Predictor selection RSMSf1	12
5.1.3	Predictor selection MBIt	13

5.2	Imputation of ERTt, RSMSf, MBIIt	13
5.2.1	Visualisation ERTt imputations	14
5.2.2	Visualisation RSMSf1 imputations	14
5.2.3	Visualisation MBIIt imputations	15
5.3	Mid term missing data evaluation	16
5.4	Imputation predictor selection part 2	17
5.4.1	Predictor selection CART_Gt	17
5.4.2	Predictor selection SET_Gt	18
5.4.3	Predictor selection EKM60t	18
5.4.4	Predictor selection ERTt (second round)	18
5.5	Imputation of CART_Gt, SET_Gt, EKM60t, ERTt (2nd imputation round)	19
5.5.1	Check after imputation part 2	19
6	Final dataset	20
6.1	Characteristics	20
6.2	Z-scores	24
7	Analyses	24
7.1	CFA models	24
7.2	CFA EMC data (N=47)	25
7.3	CFA pooled data (N=128)	27
7.4	SEM models	28
7.5	SEM EMC data (N=47)	31
8	Results	33
8.1	CFA EMC data	33
8.1.1	Fit indices	33
8.1.2	Model comparisson	33
8.1.3	Favoured model	34
8.2	CFA pooled data	36
8.2.1	Fit indices	36
8.2.2	Model comparisson	37
8.2.3	Favoured model	38
8.3	SEM EMC data	40
8.3.1	Fit indices	40
8.3.2	Model comparisson	41
8.3.3	Favoured model	44

1 Packages

```
if(!requireNamespace("rempsyc")) install.packages("rempsyc"); library(rempsyc)
if(!requireNamespace("flextable")) install.packages("flextable"); library(flextable)
if(!requireNamespace("broom")) install.packages("broom"); library(broom)
if(!requireNamespace("report")) install.packages("report"); library(report)
if(!requireNamespace("effectsize")) install.packages("effectsize"); library(effectsize)
if(!requireNamespace("tinytex")) install.packages("tinytex"); library(tinytex)
if(!requireNamespace("haven")) install.packages("haven"); library(haven)
if(!requireNamespace("dplyr")) install.packages("dplyr"); library(dplyr)
if(!requireNamespace("visdat")) install.packages("visdat"); library(visdat)
if(!requireNamespace("naniar")) install.packages("naniar"); library(naniar)
if(!requireNamespace("psych")) install.packages("psych"); library(psych)
if(!requireNamespace("mice")) install.packages("mice"); library(mice)
if(!requireNamespace("Hmisc")) install.packages("Hmisc"); library(Hmisc)
if(!requireNamespace("knitr")) install.packages("knitr"); library(knitr)
if(!requireNamespace("kableExtra")) install.packages("kableExtra"); library(kableExtra)
if(!requireNamespace("lavaan")) install.packages("lavaan"); library(lavaan)
if(!requireNamespace("semPlot")) install.packages("semPlot"); library(semPlot)
```

2 Loading files (and predefined transformation)

2.1 EMC data

```
setwd("V:/Research/Dementie/Studenten/Studenten/Max/Databeheer")

dataEMC <- haven::read_sav("data_SCTQ_merged_16072024.sav") #load EMC collected data
dataEMC[dataEMC == 999] <- NA

dataEMC_copy <- dataEMC
```

2.2 UMCG data

```
setwd("V:/Research/Dementie/Studenten/Studenten/Max/Databeheer")

dataUMCG <- haven::read_sav("data_SC_UMCG_mnk.sav") #load UMCG collected data
dataUMCG[dataUMCG == 999] <- NA

dataUMCG_copy <- dataUMCG
```

3 Data preprocessing

3.1 Pre defined transformation

```
dataEMC$TAS20_fac1_tf <- 35 - dataEMC$TAS20_fac1_Identifieren_Gevoelens
dataUMCG$TAS20_fac1_tf <- 35 - dataUMCG$TAS20_fac1_Identifieren_Gevoelens
```

3.2 Synchronizing names and compute totals

```
dataEMC$FP_1_6_total <- dataEMC$FP_1t6_ToM + dataEMC$FP_1t6_empathy
colnames(dataUMCG)[which(colnames(dataUMCG) == "SET_UMCG_Totaal")] <- "SET_UMCG_total"
dataEMC$SET_UMCG_Cognitief_Totaal <- dataEMC$SET_UMCG_1 + dataEMC$SET_UMCG_2 + dataEMC$SET_UMCG_3
dataEMC$SET_UMCG_Affectief_Totaal <- dataEMC$SET_UMCG_4 + dataEMC$SET_UMCG_5 + dataEMC$SET_UMCG_6
```

3.3 Creating ID values for UMCG sample

```
dataUMCG$ID <- paste0("UMCG", seq(1, nrow(dataUMCG)))
```

3.4 Data pooling

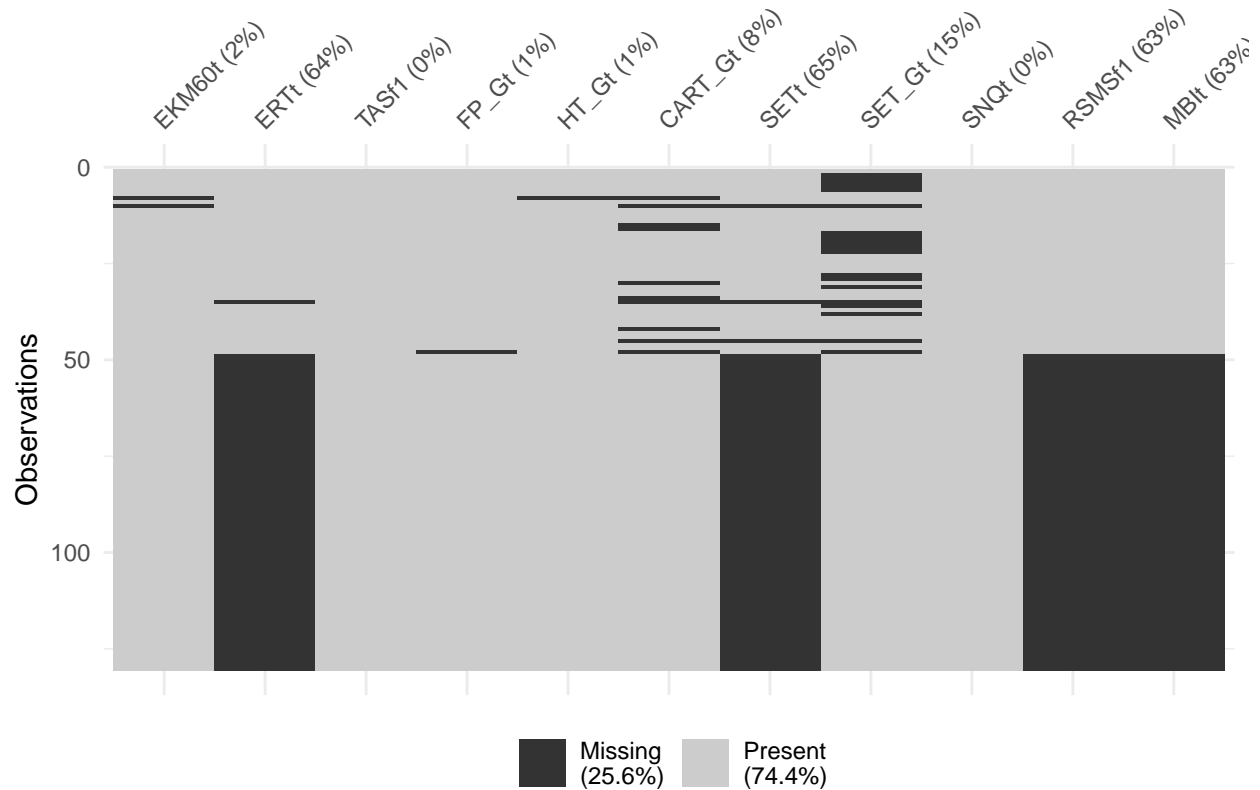
```
data_pooled <- dplyr::bind_rows(dataEMC, dataUMCG) # pooling the data
data_pooled_copy <- data_pooled
```

3.5 Renaming and var name sets

```
names_relevant_items <- c("Ekman60_total", "ERT_total", "TAS20_fac1_tf",
                          "FP_1_6_total", "hintingtask_total", "cartoons_total", "SET_total_mc", "SET_U",
                          "SNQ_NL_total", "RSMS_AMSP", "MBI_total")
names_rel_items_abbr <- c('EKM60t', 'ERTt', 'TASf1',
                          'FP_Gt', 'HT_Gt', 'CART_Gt', 'SETt', 'SET_Gt',
                          'SNQt', 'RSMSf1', 'MBIt')
names(data_pooled)[match(names_relevant_items, names(data_pooled))] <- names_rel_items_abbr #renaming
names_covariates <- c("age", "sex", "education_level")
```

3.6 Missing values part 1; visualisation

```
vis_miss(data_pooled[,names_rel_items_abbr])
```



In the above plot it is visible that we miss $\geq 63\%$ of data for four items (ERT, SET, RSMS, MBI). These items were not included in the data collection of UMCG, and therefore we assume that these missing values are missing completely at random (aka the characteristics of these participants did not influence whether this data is present/absent). Data for the UMCH sample will be imputed for those four items using stochastic imputation based on the two variables that have the highest and significant correlations with the items.

15% of data is missing for SET_Gt, all these missings are in the EMC data. For those people, SET was conducted with another testing protocol (solely multiple choice question, no open questions). We will impute SET_Gt scores for these people based on means in the EMC sample corrected for their scores on the MC questions.

The 8% missing values for the CART_Gt variable can probably be explained by lack of time (slower participants) because it was the last test in the EMC protocol. We will impute the scores using stochastic imputation based on the two variables that have the highest and significant correlation with this test.

Imputation will be continued after checks for outliers and non-normality.

4 Data checks

4.1 EMC sample data checks

Descriptives before outlier deletion and/or transformations

```
# Descriptives EMC start
psych::describe(data_pooled[data_pooled$centerID == "EMC", c(names_covariates, "MoCA_total", names_rel_
  select(n, min, max, mean, median, sd, skew, kurtosis)
```

	n	min	max	mean	median	sd	skew	kurtosis
## age	48	21.0	79	48.15	54.5	19.48	-0.13	-1.50
## sex	48	0.0	1	0.67	1.0	0.48	-0.69	-1.56
## education_level	48	2.0	7	5.69	6.0	1.11	-1.01	1.13
## MoCA_total	48	19.0	30	25.94	26.0	2.62	-0.69	0.04
## EKM60t	46	34.0	58	47.00	47.5	5.37	-0.49	0.09
## ERTt	47	34.0	76	56.74	57.0	9.74	-0.16	-0.68
## TASf1	48	5.0	28	20.04	21.0	5.42	-0.60	-0.05
## FP_Gt	47	10.5	24	20.40	21.0	2.48	-1.29	3.17
## HT_Gt	47	6.0	12	11.21	12.0	1.12	-2.31	7.58
## CART_Gt	38	2.5	12	8.00	8.0	2.63	-0.30	-0.89
## SETt	45	13.0	18	16.89	17.0	1.39	-1.36	1.02
## SET_Gt	28	9.0	12	11.46	12.0	0.79	-1.39	1.33
## SNQt	48	11.0	21	19.08	19.0	1.64	-2.53	10.04
## RSMSf1	48	11.0	32	22.60	23.0	4.73	-0.50	-0.21
## MBIt	48	42.0	70	54.44	55.0	7.16	0.16	-0.74

4.1.1 Outliers - EMC

```
dataEMC_subset_scaled <- cbind(data_pooled[data_pooled$centerID == "EMC"],[, c("ID", names_covariates,
                                                                    scale(data_pooled[data_pooled$centerID == "EMC"],[,names_rel_items_abbr])
moca_below26_EMC <- dataEMC_subset_scaled$MoCA_total < 26

outlierCheck_EMC <- data.frame(cbind(moca_below26_EMC, abs(dataEMC_subset_scaled[,names_rel_items_abbr])
#outlierCheck_EMC <- cbind(moca_below26_EMC, sapply(dataEMC_subset_scaled[,names_rel_items_abbr], FUN =
rowSums(outlierCheck_EMC, na.rm = TRUE)
```

```
## [1] 0 1 1 0 0 1 0 0 2 1 0 0 0 0 0 1 1 1 0 0 1 0 0 0 0 1 0 1 0 0 0 0 1 1 1 0 0 1
## [39] 1 0 0 1 0 0 1 0 1 2
```

```
meanitemSS_EMC <- rowMeans(dataEMC_subset_scaled[,c(
  'EKM60t', 'ERTt', 'TASf1',
  'FP_Gt', 'CART_Gt', 'SETt',
  'SNQt', 'RSMSf1', 'MBIt')], na.rm = TRUE)
SSmeanitemSS_EMC <- scale(meanitemSS_EMC)
outlier_EMC <- dataEMC_subset_scaled[which(abs(SSmeanitemSS_EMC)>3), "ID"]
length(outlier_EMC)
```

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

We have one participant that has an absolute scaled mean item scaled score of over 3, we will exclude this participant from analyses.

```
data_pooled <- data_pooled[-(which(data_pooled$ID==outlier_EMC)),]
```

4.1.2 Normality - EMC

```
psych::describe(data_pooled[data_pooled$centerID == "EMC", c(names_covariates, "MoCA_total", names_rel_
  select(n, min, max, mean, median, sd, skew, kurtosis)
```

```
##           n min max mean median sd skew kurtosis
## age      47 21.0 79 47.64  54.0 19.37 -0.09  -1.49
## sex      47  0.0  1  0.66   1.0  0.48 -0.65  -1.61
## education_level 47  2.0  7  5.72   6.0  1.10 -1.10   1.49
## MoCA_total 47 19.0 30 26.09  26.0  2.44 -0.56  -0.15
## EKM60t    45 34.0 58 47.29  48.0  5.06 -0.39   0.12
## ERTt      46 34.0 76 57.11  57.5  9.52 -0.17  -0.59
## TASf1     47  5.0 28 20.04  21.0  5.48 -0.59  -0.11
## FP_Gt     47 10.5 24 20.40  21.0  2.48 -1.29   3.17
## HT_Gt     46  6.0 12 11.26  12.0  1.08 -2.56   9.46
## CART_Gt   38  2.5 12  8.00   8.0  2.63 -0.30  -0.89
## SETt      44 13.0 18 16.98  17.0  1.27 -1.37   1.22
## SET_Gt    28  9.0 12 11.46  12.0  0.79 -1.39   1.33
## SNQt     47 16.0 21 19.26  19.0  1.13 -0.50  -0.18
## RSMSf1    47 11.0 32 22.74  23.0  4.67 -0.56  -0.05
## MBIt     47 42.0 70 54.30  55.0  7.17  0.20  -0.71
```

```
normalityCheck EMC <- psych::describe(data_pooled[data_pooled$centerID == "EMC", c(names_covariates, "M
  select(skew, kurtosis)
```

```
any(abs(normalityCheck EMC$skew)>3)
```

```
## [1] FALSE
```

```
any(abs(normalityCheck EMC$kurtosis)>10)
```

```
## [1] FALSE
```

No absolute skew>3 and no absolute kurtosis>10 is EMC sample after deleting 1 outlier.

4.1.3 Imputing the SET_Gt variable some EMC missings due to different testing protocol

```
data_pooled[data_pooled$centerID == "EMC" & is.na(data_pooled$SET_open_2), "SET_open_2"] <-
  as.numeric(colMeans(data_pooled[data_pooled$centerID == "EMC", "SET_open_2"], na.rm = TRUE))
data_pooled[data_pooled$centerID == "EMC" & is.na(data_pooled$SET_open_3), "SET_open_3"] <-
  as.numeric(colMeans(data_pooled[data_pooled$centerID == "EMC", "SET_open_3"], na.rm = TRUE))
data_pooled[data_pooled$centerID == "EMC" & is.na(data_pooled$SET_open_6), "SET_open_6"] <-
  as.numeric(colMeans(data_pooled[data_pooled$centerID == "EMC", "SET_open_6"], na.rm = TRUE))
data_pooled[data_pooled$centerID == "EMC" & is.na(data_pooled$SET_open_14), "SET_open_14"] <-
  as.numeric(colMeans(data_pooled[data_pooled$centerID == "EMC", "SET_open_14"], na.rm = TRUE))
data_pooled[data_pooled$centerID == "EMC" & is.na(data_pooled$SET_open_16), "SET_open_16"] <-
  as.numeric(colMeans(data_pooled[data_pooled$centerID == "EMC", "SET_open_16"], na.rm = TRUE))
data_pooled[data_pooled$centerID == "EMC" & is.na(data_pooled$SET_open_17), "SET_open_17"] <-
  as.numeric(colMeans(data_pooled[data_pooled$centerID == "EMC", "SET_open_17"], na.rm = TRUE))

data_pooled[data_pooled$centerID == "EMC", "SET_UMCG_1"] <-
```

```

data_pooled[data_pooled$centerID == "EMC",]$SET_open_2 + data_pooled[data_pooled$centerID == "EMC",]$
data_pooled[data_pooled$centerID == "EMC", "SET_UMCG_2"] <-
data_pooled[data_pooled$centerID == "EMC",]$SET_open_3 + data_pooled[data_pooled$centerID == "EMC",]$
data_pooled[data_pooled$centerID == "EMC", "SET_UMCG_3"] <-
data_pooled[data_pooled$centerID == "EMC",]$SET_open_6 + data_pooled[data_pooled$centerID == "EMC",]$
data_pooled[data_pooled$centerID == "EMC", "SET_UMCG_4"] <-
data_pooled[data_pooled$centerID == "EMC",]$SET_open_14 + data_pooled[data_pooled$centerID == "EMC",]$
data_pooled[data_pooled$centerID == "EMC", "SET_UMCG_5"] <-
data_pooled[data_pooled$centerID == "EMC",]$SET_open_16 + data_pooled[data_pooled$centerID == "EMC",]$
data_pooled[data_pooled$centerID == "EMC", "SET_UMCG_6"] <-
data_pooled[data_pooled$centerID == "EMC",]$SET_open_17 + data_pooled[data_pooled$centerID == "EMC",]$

data_pooled$SET_Gt <- rowSums(data_pooled[,c(which(names(data_pooled)=="SET_UMCG_1") : which(names(data_pooled)=="SET_UMCG_6"))])

```

4.2 UMCG sample data checks

Descriptives before outlier deletion and/or tranformations

```

# Descriptives UMCG start
psych::describe(data_pooled[data_pooled$centerID == "UMCG", c(names_covariates, "MoCA_total", names_rel_items_abbrev)],
  select(n, min, max, mean, median, sd, skew, kurtosis)

```

##		n	min	max	mean	median	sd	skew	kurtosis
##	age	82	18	66	28.71	22	14.07	1.36	0.16
##	sex	82	0	1	0.63	1	0.48	-0.55	-1.72
##	education_level	82	3	7	6.00	6	0.61	-1.30	6.07
##	MoCA_total	0	Inf	-Inf	NaN	NA	NA	NA	NA
##	EKM60t	82	38	56	46.99	48	4.52	-0.26	-0.88
##	ERTt	0	Inf	-Inf	NaN	NA	NA	NA	NA
##	TASf1	82	7	31	21.09	21	4.62	-0.68	0.53
##	FP_Gt	82	13	24	20.17	21	2.51	-0.81	0.21
##	HT_Gt	82	6	12	11.40	12	1.00	-2.66	9.62
##	CART_Gt	82	0	12	8.51	9	2.51	-0.49	-0.05
##	SETt	0	Inf	-Inf	NaN	NA	NA	NA	NA
##	SET_Gt	82	8	12	11.33	12	0.94	-1.38	1.28
##	SNQt	82	16	22	19.38	19	1.27	-0.40	-0.21
##	RSMSf1	0	Inf	-Inf	NaN	NA	NA	NA	NA
##	MBIt	0	Inf	-Inf	NaN	NA	NA	NA	NA

4.2.1 Outliers - UMCG

```

dataUMCG_subset_scaled <- cbind(data_pooled[data_pooled$centerID == "UMCG",][, c("ID", names_covariates, names_rel_items_abbrev)],
  moca_below26_UMCG <- dataUMCG_subset_scaled$MoCA_total < 26

outlierCheck_UMCG <- data.frame(cbind(moca_below26_UMCG, abs(dataUMCG_subset_scaled[,names_rel_items_abbrev])))
#outlierCheck_UMCG <- cbind(moca_below26_UMCG, apply(dataUMCG_subset_scaled[,names_rel_items_abbrev], MARGIN=2, FUN=abs))

rowSums(outlierCheck_UMCG, na.rm = TRUE)

```

```
## [1] 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 0 0 0 0 0 1 0 0 0 0 0
```



```
## [39] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
## [77] 0 0 0 0 1 0
```

```
meanitemSS_UMCG <- rowMeans(dataUMCG_subset_scaled[,c(
  'EKM60t', 'ERTt', 'TASf1',
  'FP_Gt', 'CART_Gt', 'SET_Gt',
  'SNQt', 'RSMSf1', 'MBIt')], na.rm = TRUE)
SSmeanitemSS_UMCG <- scale(meanitemSS_UMCG)
outlier_UMCG <- dataUMCG_subset_scaled[which(abs(SSmeanitemSS_UMCG)>3), "ID"]
length(outlier_UMCG)
```

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

We have one participant that has an absolute scaled mean item scaled score of over 3, we will exclude this participant from analyses.

```
data_pooled <- data_pooled[-(which(data_pooled$ID==outlier_UMCG)),]
```

4.2.2 Normality - UMCG

```
psych::describe(data_pooled[data_pooled$centerID == "UMCG", c(names_covariates, "MoCA_total", names_rel.
  select(n, min, max, mean, median, sd, skew, kurtosis)
```

##	n	min	max	mean	median	sd	skew	kurtosis
## age	81	18	66	28.70	22	14.16	1.35	0.12
## sex	81	0	1	0.63	1	0.49	-0.53	-1.74
## education_level	81	3	7	6.01	6	0.60	-1.36	6.61
## MoCA_total	0	Inf	-Inf	NaN	NA	NA	NA	NA
## EKM60t	81	38	56	47.06	48	4.50	-0.29	-0.83
## ERTt	0	Inf	-Inf	NaN	NA	NA	NA	NA
## TASf1	81	7	31	21.20	21	4.54	-0.70	0.72
## FP_Gt	81	14	24	20.26	21	2.40	-0.72	0.03
## HT_Gt	81	6	12	11.40	12	1.01	-2.64	9.49
## CART_Gt	81	0	12	8.55	9	2.49	-0.53	0.05
## SETt	0	Inf	-Inf	NaN	NA	NA	NA	NA
## SET_Gt	81	8	12	11.35	12	0.94	-1.44	1.50
## SNQt	81	16	22	19.41	19	1.25	-0.41	-0.12
## RSMSf1	0	Inf	-Inf	NaN	NA	NA	NA	NA
## MBIt	0	Inf	-Inf	NaN	NA	NA	NA	NA

```
normalityCheck_UMCG <- psych::describe(data_pooled[data_pooled$centerID == "UMCG", c(names_covariates,
  select(skew, kurtosis)
```

```
any(abs(normalityCheck_UMCG$skew)>3, na.rm = TRUE)
```

```
## [1] FALSE
```

```
any(abs(normalityCheck_UMCG$kurtosis)>10, na.rm = TRUE)
```

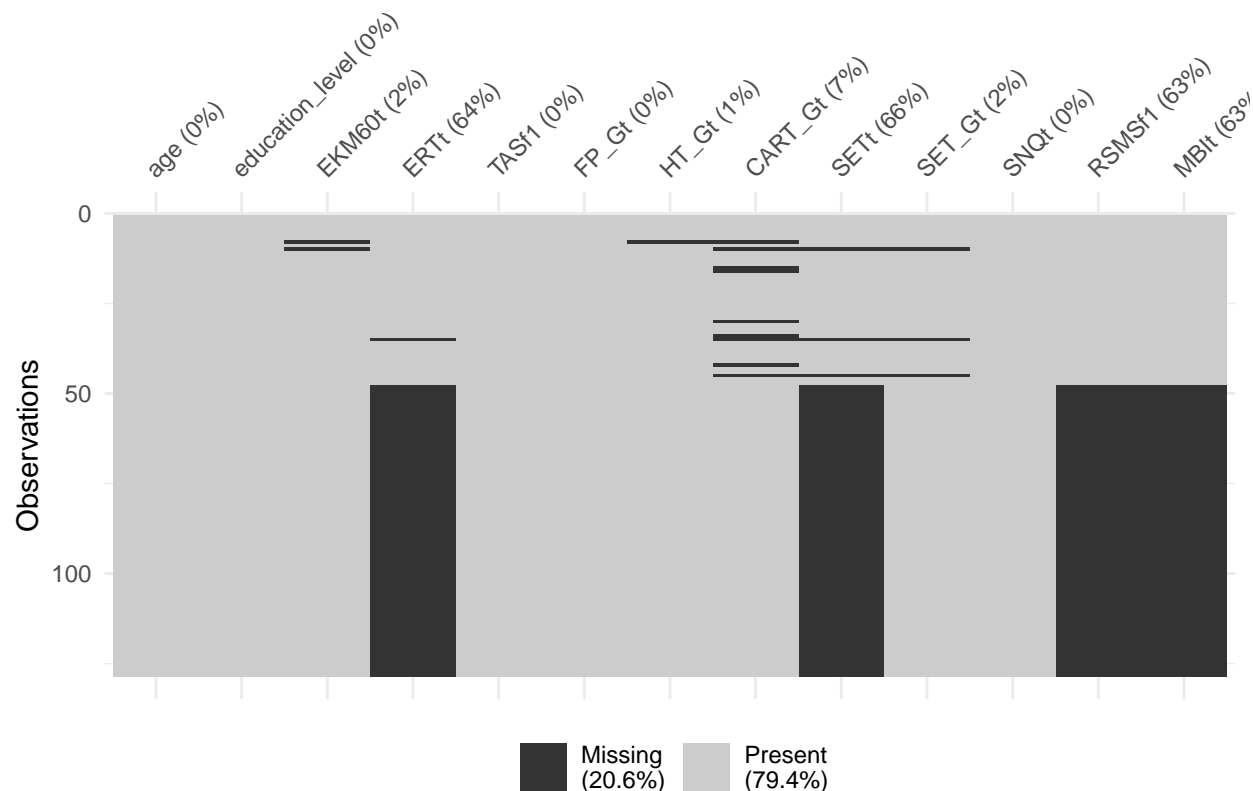
```
## [1] FALSE
```

No absolute skew>3 and no absolute kurtosis>10 is EMC sample after deleting 1 outlier.

5 Stochastic imputation

At this point, we have the following missing values.

```
vis_miss(data_pooled[,c("age", "education_level", names_rel_items_abbrev)])
```



And the following zero order paired correlation (respectively: corr, N, p-values).

```
rcorr(as.matrix(data_pooled[, c("age", "education_level",
                                'EKM60t', 'ERTt', 'TASf1',
                                'FP_Gt', 'CART_Gt', 'SET_Gt',
                                'SNQt', 'RSMSf1', 'MBIt')]))
```

```
##          age education_level EKM60t  ERTt TASf1 FP_Gt CART_Gt SET_Gt
## age          1.00         -0.28  -0.06 -0.42  0.04 -0.10   0.02 -0.20
## education_level -0.28          1.00   0.18  0.45  0.09  0.14   0.34  0.18
## EKM60t        -0.06          0.18   1.00  0.64  0.17  0.30   0.24  0.12
## ERTt          -0.42          0.45   0.64  1.00  0.17  0.35   0.25  0.35
## TASf1          0.04          0.09   0.17  0.17  1.00  0.11   0.02  0.04
## FP_Gt         -0.10          0.14   0.30  0.35  0.11  1.00   0.22  0.18
## CART_Gt        0.02          0.34   0.24  0.25  0.02  0.22   1.00  0.19
## SET_Gt        -0.20          0.18   0.12  0.35  0.04  0.18   0.19  1.00
## SNQt          -0.09          0.26   0.10  0.14  0.06  0.23   0.13  0.10
## RSMSf1        -0.07          0.00   0.29  0.09  0.35  0.24   0.29  0.13
## MBIt          0.33          -0.04   0.13  0.18  0.22  0.01   0.32  0.07
##          SNQt RSMSf1  MBIt
```

```

## age          -0.09  -0.07  0.33
## education_level 0.26   0.00 -0.04
## EKM60t        0.10   0.29  0.13
## ERTt          0.14   0.09  0.18
## TASf1         0.06   0.35  0.22
## FP_Gt         0.23   0.24  0.01
## CART_Gt       0.13   0.29  0.32
## SET_Gt        0.10   0.13  0.07
## SNQt          1.00  -0.14  0.05
## RSMSf1        -0.14   1.00 -0.09
## MBIt          0.05  -0.09  1.00
##
## n
##           age education_level EKM60t ERTt TASf1 FP_Gt CART_Gt SET_Gt SNQt
## age          128             128    126  46   128   128    119   125  128
## education_level 128             128    126  46   128   128    119   125  128
## EKM60t        126             126    126  44   126   126    119   124  126
## ERTt          46             46     44  46   46    46     38    44   46
## TASf1         128             128    126  46   128   128    119   125  128
## FP_Gt         128             128    126  46   128   128    119   125  128
## CART_Gt       119             119    119  38   119   119    119   119  119
## SET_Gt        125             125    124  44   125   125    119   125  125
## SNQt          128             128    126  46   128   128    119   125  128
## RSMSf1         47             47     45  46   47    47     38    44   47
## MBIt          47             47     45  46   47    47     38    44   47
##           RSMSf1 MBIt
## age          47    47
## education_level 47    47
## EKM60t        45    45
## ERTt          46    46
## TASf1         47    47
## FP_Gt         47    47
## CART_Gt       38    38
## SET_Gt        44    44
## SNQt          47    47
## RSMSf1        47    47
## MBIt          47    47
##
## P
##           age      education_level EKM60t ERTt   TASf1 FP_Gt  CART_Gt
## age          0.0015             0.5077 0.0036 0.6536 0.2670 0.8348
## education_level 0.0015             0.0425 0.0016 0.2938 0.1080 0.0002
## EKM60t        0.5077 0.0425             0.0000 0.0588 0.0007 0.0087
## ERTt          0.0036 0.0016             0.0000 0.2556 0.0182 0.1263
## TASf1         0.6536 0.2938             0.0588 0.2556 0.2054 0.7995
## FP_Gt         0.2670 0.1080             0.0007 0.0182 0.2054 0.0168
## CART_Gt       0.8348 0.0002             0.0087 0.1263 0.7995 0.0168
## SET_Gt        0.0269 0.0509             0.1999 0.0217 0.6607 0.0420 0.0409
## SNQt          0.2871 0.0028             0.2846 0.3457 0.4734 0.0107 0.1681
## RSMSf1        0.6426 0.9846             0.0516 0.5463 0.0161 0.1084 0.0797
## MBIt          0.0233 0.7803             0.3871 0.2414 0.1332 0.9620 0.0490
##           SET_Gt SNQt   RSMSf1 MBIt
## age          0.0269 0.2871 0.6426 0.0233
## education_level 0.0509 0.0028 0.9846 0.7803

```

```
## EKM60t      0.1999 0.2846 0.0516 0.3871
## ERTt        0.0217 0.3457 0.5463 0.2414
## TASf1       0.6607 0.4734 0.0161 0.1332
## FP_Gt       0.0420 0.0107 0.1084 0.9620
## CART_Gt     0.0409 0.1681 0.0797 0.0490
## SET_Gt      0.2611 0.4116 0.6504
## SNQt        0.2611      0.3499 0.7418
## RSMSf1      0.4116 0.3499      0.5658
## MBIt        0.6504 0.7418 0.5658
```

5.1 Imputation predictor selection

5.1.1 Predictor selection ERTt

Lets check the correlations of ERTt with other relevant variables.

```
matcor <- Hmisc:: rcorr(as.matrix(data_pooled[, c("age", "education_level",
        'EKM60t', 'ERTt', 'TASf1',
        'FP_Gt', 'HT_Gt', 'CART_Gt', 'SET_Gt',
        'SNQt', 'RSMSf1', 'MBIt')]))
```

#Check whether sex is also a significant predictor

```
sexERTt <- lm(ERTt ~ sex, data = data_pooled)
summary(sexERTt)$coefficients["sex", "Pr(>|t|)"] #Not significant
```

```
## [1] 0.1775048
```

```
sigpred_ERTt <- matcor$r["ERTt",matcor$P["ERTt",]<.05]; sigpred_ERTt #significant correlations of ERTt
```

```
##          age education_level      EKM60t      <NA>      FP_Gt
##    -0.4211357      0.4526528      0.6353631      NA      0.3468226
##          SET_Gt
##    0.3452348
```

```
# corrrplot(matcor$r, p.mat = matcor$P, sig.level = 0.05, method = "number", type = "lower", tl.col = "b
#          #insig = "blank",
#          addCoef.col = "black")
```

We will impute the missing ERTt values using predictive mean matching (stochastic) based on the significant predictors.

5.1.2 Predictor selection RSMSf1

Lets check the correlations of RSMSf1 with other relevant variables.

#Check whether sex is also a significant predictor

```
sexRSMSf1 <- lm(RSMSf1 ~ sex, data = data_pooled)
summary(sexRSMSf1)$coefficients["sex", "Pr(>|t|)"] #Not significant
```

```
## [1] 0.08581445
```

```
#Check numerical correlations
sigpred_RSMSf1 <- matcor$r["RSMSf1",matcor$P["RSMSf1"],]<.05]; sigpred_RSMSf1 #significant correlations
```

```
##      TASf1      <NA>
## 0.3494505      NA
```

5.1.3 Predictor selection MBI

Lets check the correlations of MBI with other relevant variables.

```
#Check whether sex is also a significant predictor
sexMBIt <- lm(MBI ~ sex, data = data_pooled)
summary(sexMBIt)$coefficients["sex", "Pr(>|t|)"] #Not significant
```

```
## [1] 0.33373
```

```
#Check numerical correlations
sigpred_MBI <- matcor$r["MBI",matcor$P["MBI"],]<.05]; sigpred_MBI #significant correlations of MBI
```

```
##      age  CART_Gt      <NA>
## 0.3305121 0.3216157      NA
```

5.2 Imputation of ERTt, RSMSf, MBI

```
data_pooled_subset <- data_pooled[,c("ID", "centerID", "sex", "age", "education_level",
                                     'EKM60t', 'ERTt', 'TASf1',
                                     'FP_Gt', 'CART_Gt', 'SET_Gt',
                                     'SNQt', 'RSMSf1', 'MBIt')]
```

```
predictor_matrix <- make.predictorMatrix(data_pooled_subset)
predictor_matrix[] <- 0
```

```
#assign the significant predictors to the prediction matrix
predictor_matrix["ERTt", ] <- colnames(predictor_matrix) %in% names(sigpred_ERTt)[!is.na(names(sigpred_ERTt))]
predictor_matrix["RSMSf1", ] <- colnames(predictor_matrix) %in% names(sigpred_RSMSf1)[!is.na(names(sigpred_RSMSf1))]
predictor_matrix["MBI", ] <- colnames(predictor_matrix) %in% names(sigpred_MBI)[!is.na(names(sigpred_MBI))]
```

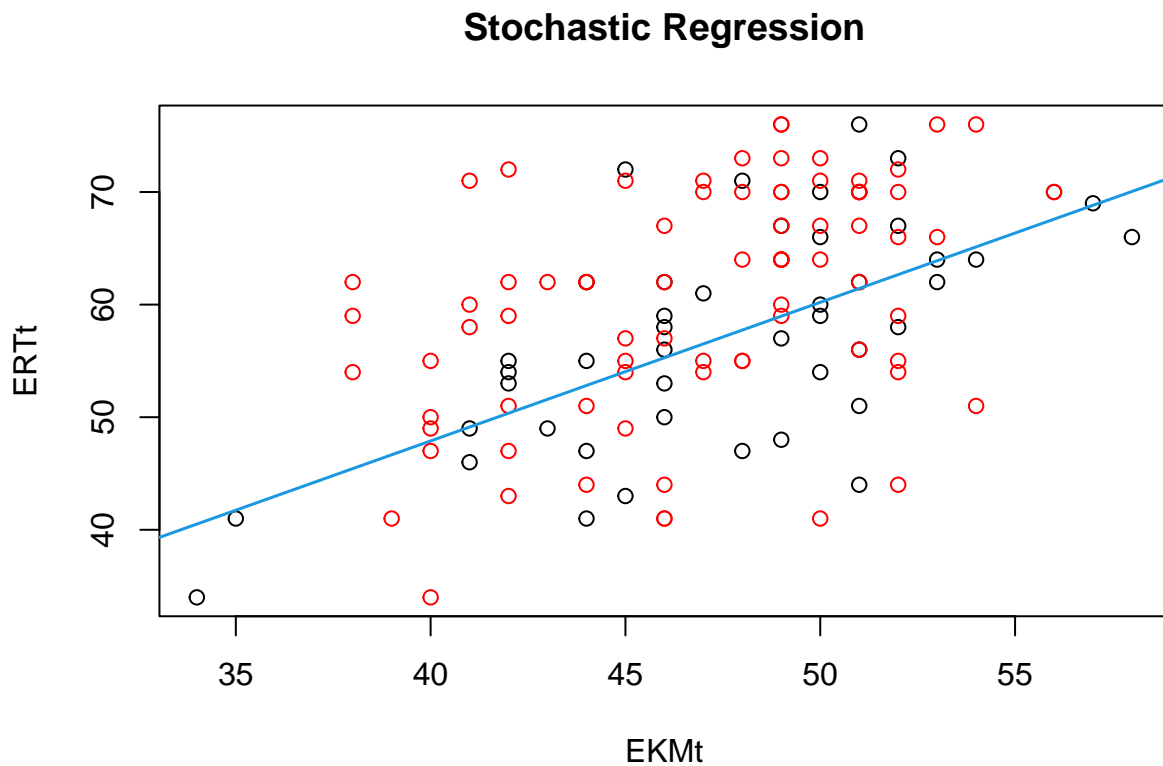
```
#assign imputation methods
imputation_methods <- make.method(data_pooled_subset)
imputation_methods[] <- "" # Set all methods to "" initially
imputation_methods[c("ERTt", "RSMSf1", "MBI")] <- "pmm" # Use "pmm" for ERTt
```

```
#imputation
data_pooled_subset_imp <- complete(mice(data_pooled_subset,
                                       method = imputation_methods,
                                       predictorMatrix = predictor_matrix,
                                       m = 1,
                                       seed = 42), 1)
```

```
##
## iter imp variable
## 1 1 ERTt RSMSf1 MBIt
## 2 1 ERTt RSMSf1 MBIt
## 3 1 ERTt RSMSf1 MBIt
## 4 1 ERTt RSMSf1 MBIt
## 5 1 ERTt RSMSf1 MBIt
```

5.2.1 Visualisation ERTt imputations

```
# Stochastic regression imputation plot for ERTt
plot(data_pooled_subset$EKM60t[!is.na(data_pooled_subset$ERTt)], data_pooled_subset_imp$ERTt[!is.na(data_pooled_subset_imp$ERTt)],
     main = "Stochastic Regression",
     xlab = "EKMt", ylab = "ERTt")
points(data_pooled_subset$EKM60t[is.na(data_pooled_subset$ERTt)], data_pooled_subset_imp$ERTt[is.na(data_pooled_subset$ERTt)],
       col = "red")
abline(lm(data_pooled_subset$ERTt ~ data_pooled_subset$EKM60t, data_pooled_subset_imp), col = "#1b98e0")
```



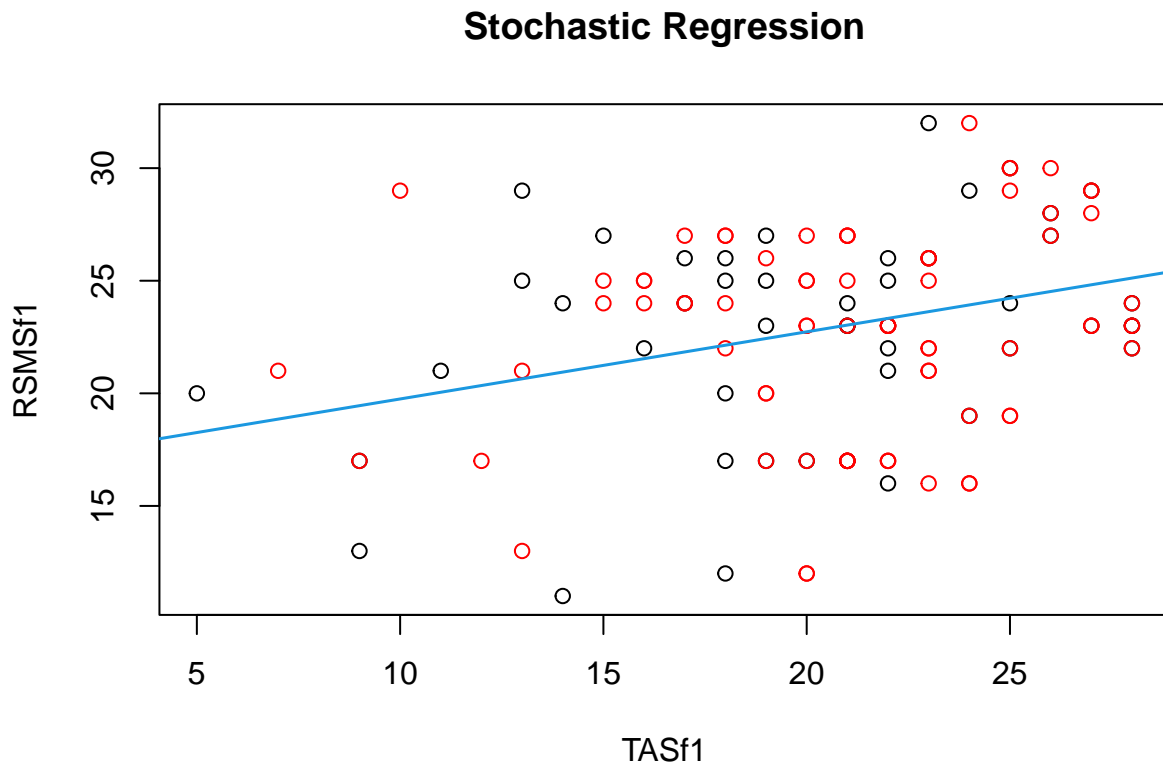
5.2.2 Visualisation RSMSf1 imputations

```
# Stochastic regression imputation plot for RSMSf1
plot(data_pooled_subset$TASf1[!is.na(data_pooled_subset$RSMSf1)], data_pooled_subset_imp$RSMSf1[!is.na(data_pooled_subset_imp$RSMSf1)],
     main = "Stochastic Regression",
     xlab = "TASf1", ylab = "RSMSf1")
points(data_pooled_subset$TASf1[is.na(data_pooled_subset$RSMSf1)], data_pooled_subset_imp$RSMSf1[is.na(data_pooled_subset$RSMSf1)],
       col = "red")
abline(lm(data_pooled_subset$RSMSf1 ~ data_pooled_subset$TASf1, data_pooled_subset_imp), col = "#1b98e0")
```

```

main = "Stochastic Regression",
xlab = "TASf1", ylab = "RSMSf1")
points(data_pooled_subset$TASf1[is.na(data_pooled_subset$RSMSf1)], data_pooled_subset_imp$RSMSf1[is.na(
  col = "red")
abline(lm(data_pooled_subset$RSMSf1 ~ data_pooled_subset$TASf1, data_pooled_subset_imp), col = "#1b98e0", l

```



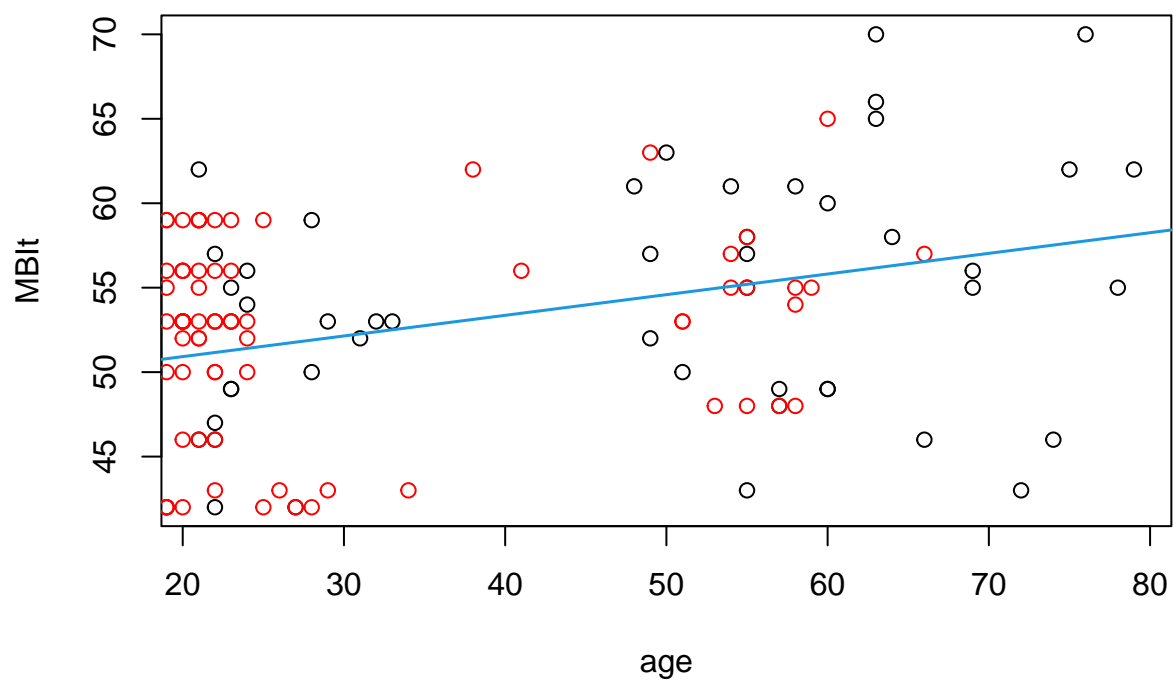
5.2.3 Visualisation MBI_t imputations

```

# Stochastic regression imputation plot for MBIt
plot(data_pooled_subset$age[!is.na(data_pooled_subset$MBIt)], data_pooled_subset_imp$MBIt[!is.na(data_p
  main = "Stochastic Regression",
  xlab = "age", ylab = "MBIt")
points(data_pooled_subset$age[is.na(data_pooled_subset$MBIt)], data_pooled_subset_imp$MBIt[is.na(data_p
  col = "red")
abline(lm(data_pooled_subset$MBIt ~ data_pooled_subset$age, data_pooled_subset_imp), col = "#1b98e0", l

```

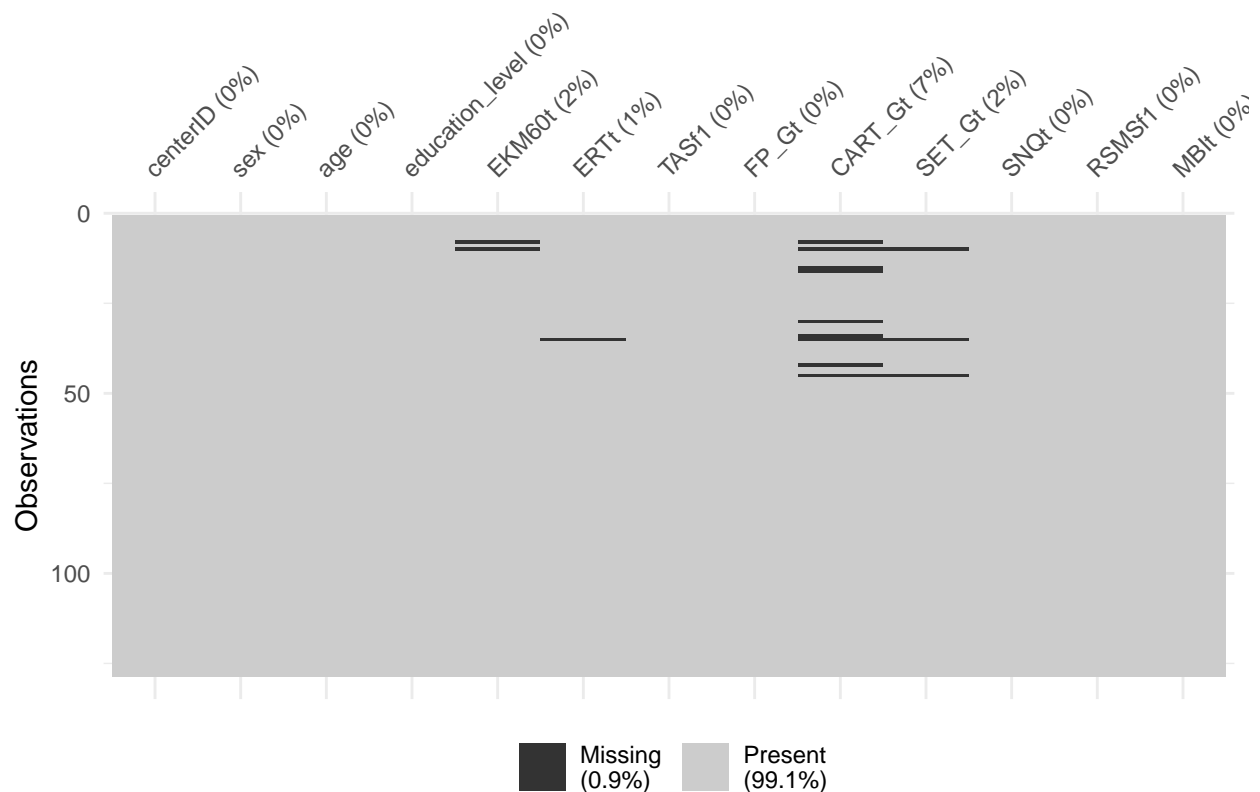
Stochastic Regression



5.3 Mid term missing data evaluation

At this point, we have the following missing data:

```
vis_miss(data_pooled_subset_imp[, -1])
```

We will impute the remaining missing values, again with a stochastic approach based on significant correlations or significant t.test for sex.

5.4 Imputation predictor selection part 2

5.4.1 Predictor selection CART_Gt

Lets check the correlations of CART_Gt with other relevant variables.

```
matcor2 <- Hmisc:: rcorr(as.matrix(data_pooled_subset_imp[, c("age", "education_level",
  'EKM60t', 'ERTt', 'TASf1',
  'FP_Gt', 'CART_Gt', 'SET_Gt',
  'SNQt', 'RSMSf1', 'MBIt')]))
```

#Check whether sex is also a significant predictor

```
sexCART_Gt <- lm(CART_Gt ~ sex, data = data_pooled_subset_imp)
summary(sexCART_Gt)$coefficients["sex", "Pr(>|t|)"] #Significant
```

```
## [1] 0.0005648991
```

```
sigpred_CART_Gt <- matcor2$r["CART_Gt",matcor2$p["CART_Gt",]< .05]; sigpred_CART_Gt #significant correla
```

```
## education_level      EKM60t      ERTt      FP_Gt      <NA>
##      0.3372836      0.2394804      0.2402233      0.2187845      NA
##      SET_Gt      RSMSf1
##      0.1877775      0.2021467
```

```
sigpred_CART_Gt$sex <- summary(sexCART_Gt)$coefficients["sex", "Pr(>|t|)"] #Significant
```

```
## Warning in sigpred_CART_Gt$sex <- summary(sexCART_Gt)$coefficients["sex", :  
## Coercing LHS to a list
```

5.4.2 Predictor selection SET_Gt

Lets check the correlations of SET_Gt with other relevant variables.

```
#Check whether sex is also a significant predictor  
sexSET_Gt <- lm(SET_Gt ~ sex, data = data_pooled_subset_imp)  
summary(sexSET_Gt)$coefficients["sex", "Pr(>|t|)"] #Not ignificant
```

```
## [1] 0.1354767
```

```
sigpred_SET_Gt <- matcor2$r["SET_Gt",matcor2$P["SET_Gt",]<.05]; sigpred_SET_Gt #significant correlation
```

```
##          age          ERTt          FP_Gt          CART_Gt          <NA>  
## -0.1980193  0.5010054  0.1821579  0.1877775          NA
```

5.4.3 Predictor selection EKM60t

Lets check the correlations of EKM60t with other relevant variables.

```
#Check whether sex is also a significant predictor  
sexEKM60t <- lm(EKM60t ~ sex, data = data_pooled_subset_imp)  
summary(sexEKM60t)$coefficients["sex", "Pr(>|t|)"] #Not significant
```

```
## [1] 0.2449073
```

```
sigpred_EKM60t <- matcor2$r["EKM60t",matcor2$P["EKM60t",]<.05]; sigpred_EKM60t #significant correlation
```

```
## education_level          <NA>          ERTt          FP_Gt          CART_Gt  
##      0.1810381          NA      0.5057871      0.2988278      0.2394804  
##      RSMSf1  
##      0.1924306
```

5.4.4 Predictor selection ERTt (second round)

Lets check the correlations of ERTt with other relevant variables.

```
#Check whether sex is also a significant predictor  
sexERTt_2nd <- lm(ERTt ~ sex, data = data_pooled_subset_imp)  
summary(sexERTt_2nd)$coefficients["sex", "Pr(>|t|)"] #Not significant
```

```
## [1] 0.7873681
```

```
sigpred_ERTt_2nd <- matcor2$r["ERTt",matcor2$P["ERTt",]<.<0.05]; sigpred_ERTt_2nd #significant correlation
```

```
##          age education_level      EKM60t      <NA>      FP_Gt
##      -0.3202251      0.2671958      0.5057871      NA      0.4173242
##          CART_Gt      SET_Gt
##      0.2402233      0.5010054
```

5.5 Imputation of CART_Gt, SET_Gt, EKM60t, ERTt (2nd imputation round)

```
data_pooled_subset_imp2 <- data_pooled_subset_imp

predictor_matrix2 <- make.predictorMatrix(data_pooled_subset_imp)
predictor_matrix2[] <- 0

#assign the significant predictors to the prediction matrix
predictor_matrix2["CART_Gt", ] <-
  colnames(predictor_matrix2) %in% names(sigpred_CART_Gt)[!is.na(names(sigpred_CART_Gt))]
predictor_matrix2["SET_Gt", ] <-
  colnames(predictor_matrix2) %in% names(sigpred_SET_Gt)[!is.na(names(sigpred_SET_Gt))]
predictor_matrix2["EKM60t", ] <-
  colnames(predictor_matrix2) %in% names(sigpred_EKM60t)[!is.na(names(sigpred_EKM60t))]
predictor_matrix2["ERTt", ] <-
  colnames(predictor_matrix2) %in% names(sigpred_ERTt_2nd)[!is.na(names(sigpred_ERTt_2nd))]

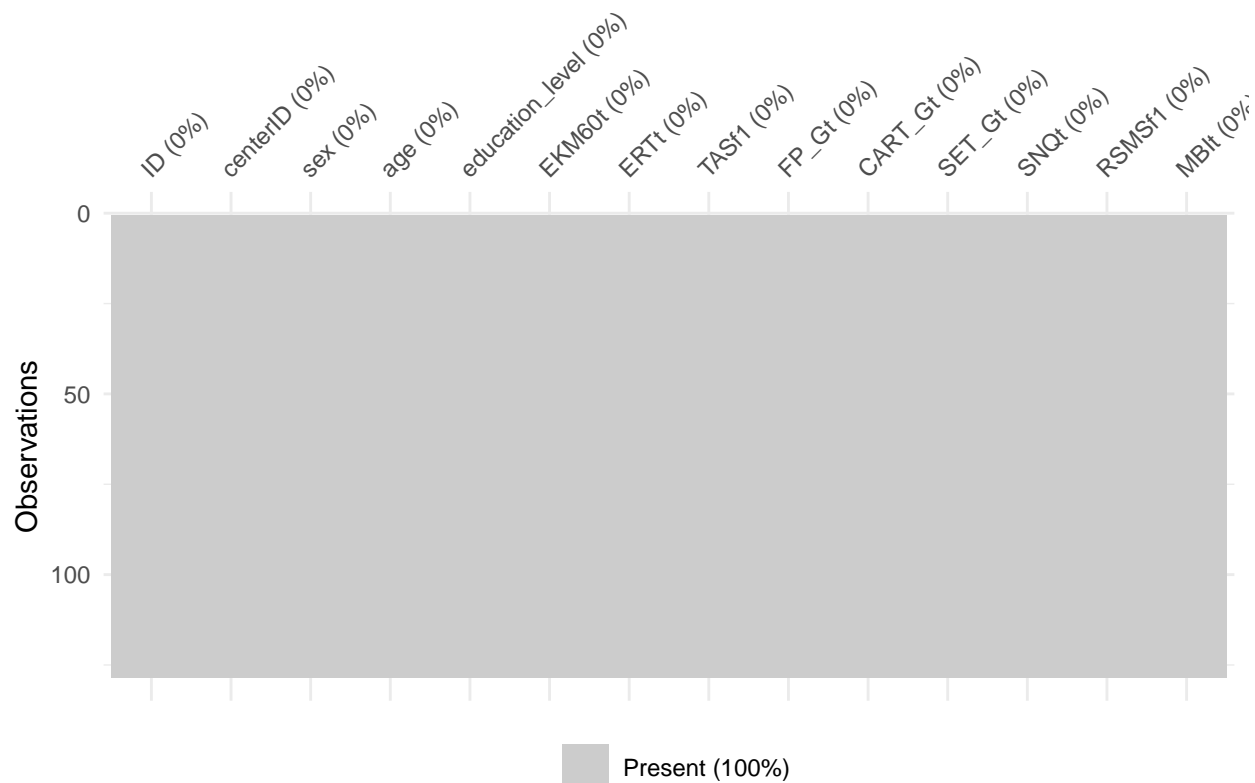
#assign imputation methods
imputation_methods2 <- make.method(data_pooled_subset_imp)
imputation_methods2[] <- "" # Set all methods to "" initially
imputation_methods2[c("CART_Gt", "SET_Gt", "EKM60t", "ERTt")] <- "pmm" # Use "pmm" for ERTt

#imputation
data_pooled_subset_imp2 <- complete(mice(data_pooled_subset_imp,
  method = imputation_methods2,
  predictorMatrix = predictor_matrix2,
  m = 1,
  seed = 42), 1)
```

```
##
## iter imp variable
## 1 1 EKM60t ERTt CART_Gt SET_Gt
## 2 1 EKM60t ERTt CART_Gt SET_Gt
## 3 1 EKM60t ERTt CART_Gt SET_Gt
## 4 1 EKM60t ERTt CART_Gt SET_Gt
## 5 1 EKM60t ERTt CART_Gt SET_Gt
```

5.5.1 Check after imputation part 2

```
vis_miss(data_pooled_subset_imp2)
```



No missing values anymore

6 Final dataset

6.1 Characteristics

```
data_final <- data_pooled_subset_imp2
data_final <- merge(data_final, data_pooled[,c('ID', 'TQ_total')], by = "ID", all.x = TRUE)
colnames(data_final)[colnames(data_final)=="TQ_total"] <- "TQt"

descriptives_data_final <- psych::describe(data_final[, -1]) %>%
  select(n, min, max, mean, median, sd, skew, kurtosis); descriptives_data_final
```

##	n	min	max	mean	median	sd	skew	kurtosis
## centerID*	128	1.0	2	1.63	2	0.48	-0.54	-1.72
## sex	128	0.0	1	0.64	1	0.48	-0.58	-1.68
## age	128	18.0	79	35.66	24	18.61	0.75	-1.00
## education_level	128	2.0	7	5.91	6	0.83	-1.57	4.69
## EKM60t	128	34.0	58	47.16	48	4.66	-0.35	-0.27
## ERTt	128	34.0	76	59.30	60	10.03	-0.34	-0.68

```
## TASf1      128  5.0  31 20.77      21  4.92 -0.71      0.49
## FP_Gt      128 10.5  24 20.31      21  2.42 -0.95      1.32
## CART_Gt    128  0.0  12  8.39       9  2.57 -0.41     -0.41
## SET_Gt     128  8.0  12 11.36      12  0.85 -1.52      2.03
## SNQt       128 16.0  22 19.35      19  1.21 -0.43     -0.07
## RSMSf1     128 11.0  32 22.63      23  4.57 -0.39     -0.37
## MBI_t      128 42.0  70 52.94      53  6.39  0.03     -0.31
## TQt        47 10.0  18 14.66      15  2.25 -0.21     -1.03
```

```
corr_data_final <- Hmisc::rcorr(as.matrix(data_final[, -c(1:2)])); corr_data_final
```

```
##          sex  age education_level EKM60t  ERTt TASf1 FP_Gt CART_Gt
## sex      1.00 -0.02                -0.14  0.10 -0.03 -0.03 -0.11  -0.35
## age     -0.02  1.00                -0.28  -0.06 -0.32  0.04 -0.10  -0.04
## education_level -0.14 -0.28          1.00  0.18  0.27  0.09  0.14   0.38
## EKM60t   0.10 -0.06                0.18  1.00  0.50  0.16  0.30   0.28
## ERTt     -0.03 -0.32                0.27  0.50  1.00  0.10  0.42   0.25
## TASf1    -0.03  0.04                0.09  0.16  0.10  1.00  0.11   0.01
## FP_Gt    -0.11 -0.10                0.14  0.30  0.42  0.11  1.00   0.26
## CART_Gt  -0.35 -0.04                0.38  0.28  0.25  0.01  0.26   1.00
## SET_Gt   -0.13 -0.18                0.15  0.11  0.50  0.05  0.17   0.19
## SNQt      0.02 -0.09                0.26  0.09  0.09  0.06  0.23   0.12
## RSMSf1   -0.22  0.05                -0.01  0.19  0.01  0.25  0.01   0.21
## MBI_t     0.15  0.31                0.05  0.06 -0.07  0.12 -0.07   0.08
## TQt       0.11 -0.05                0.25  0.39  0.26  0.15  0.14   0.25
##
##          SET_Gt SNQt RSMSf1  MBI_t  TQt
## sex      -0.13  0.02  -0.22  0.15  0.11
## age     -0.18 -0.09   0.05  0.31 -0.05
## education_level 0.15  0.26  -0.01  0.05  0.25
## EKM60t   0.11  0.09   0.19  0.06  0.39
## ERTt     0.50  0.09   0.01 -0.07  0.26
## TASf1    0.05  0.06   0.25  0.12  0.15
## FP_Gt    0.17  0.23   0.01 -0.07  0.14
## CART_Gt  0.19  0.12   0.21  0.08  0.25
## SET_Gt   1.00  0.10   0.06 -0.05  0.26
## SNQt     0.10  1.00   0.00  0.04 -0.10
## RSMSf1   0.06  0.00   1.00  0.00  0.18
## MBI_t    -0.05  0.04   0.00  1.00 -0.10
## TQt      0.26 -0.10   0.18 -0.10  1.00
##
## n
##          sex age education_level EKM60t ERTt TASf1 FP_Gt CART_Gt SET_Gt
## sex      128 128                128  128  128  128  128  128  128
## age      128 128                128  128  128  128  128  128  128
## education_level 128 128          128  128  128  128  128  128  128
## EKM60t    128 128                128  128  128  128  128  128  128
## ERTt      128 128                128  128  128  128  128  128  128
## TASf1     128 128                128  128  128  128  128  128  128
## FP_Gt     128 128                128  128  128  128  128  128  128
## CART_Gt   128 128                128  128  128  128  128  128  128
## SET_Gt    128 128                128  128  128  128  128  128  128
## SNQt      128 128                128  128  128  128  128  128  128
## RSMSf1    128 128                128  128  128  128  128  128  128
## MBI_t     128 128                128  128  128  128  128  128  128
```

```

## TQt          47  47          47      47  47  47  47  47  47
##          SNQt RSMSf1 MBIt TQt
## sex          128    128  128  47
## age          128    128  128  47
## education_level 128    128  128  47
## EKM60t        128    128  128  47
## ERTt          128    128  128  47
## TASf1         128    128  128  47
## FP_Gt         128    128  128  47
## CART_Gt       128    128  128  47
## SET_Gt        128    128  128  47
## SNQt          128    128  128  47
## RSMSf1        128    128  128  47
## MBIt          128    128  128  47
## TQt          47     47   47  47
##
## P
##          sex      age      education_level EKM60t ERTt  TASf1  FP_Gt
## sex          0.7994  0.1036          0.2440 0.7703 0.6977 0.2066
## age          0.7994          0.0015          0.5290 0.0002 0.6536 0.2670
## education_level 0.1036 0.0015          0.0399 0.0023 0.2938 0.1080
## EKM60t        0.2440 0.5290 0.0399          0.0000 0.0765 0.0006
## ERTt          0.7703 0.0002 0.0023          0.0000          0.2653 0.0000
## TASf1         0.6977 0.6536 0.2938          0.0765 0.2653          0.2054
## FP_Gt         0.2066 0.2670 0.1080          0.0006 0.0000 0.2054
## CART_Gt       0.0000 0.6772 0.0000          0.0015 0.0052 0.8905 0.0031
## SET_Gt        0.1567 0.0399 0.0819          0.2271 0.0000 0.6087 0.0497
## SNQt          0.8589 0.2871 0.0028          0.3173 0.3092 0.4734 0.0107
## RSMSf1        0.0121 0.5607 0.9181          0.0304 0.8924 0.0046 0.8941
## MBIt          0.0884 0.0004 0.5449          0.4963 0.4276 0.1680 0.4558
## TQt          0.4532 0.7220 0.0878          0.0069 0.0811 0.3159 0.3399
##          CART_Gt SET_Gt SNQt  RSMSf1 MBIt  TQt
## sex          0.0000 0.1567 0.8589 0.0121 0.0884 0.4532
## age          0.6772 0.0399 0.2871 0.5607 0.0004 0.7220
## education_level 0.0000 0.0819 0.0028 0.9181 0.5449 0.0878
## EKM60t        0.0015 0.2271 0.3173 0.0304 0.4963 0.0069
## ERTt          0.0052 0.0000 0.3092 0.8924 0.4276 0.0811
## TASf1         0.8905 0.6087 0.4734 0.0046 0.1680 0.3159
## FP_Gt         0.0031 0.0497 0.0107 0.8941 0.4558 0.3399
## CART_Gt       0.0000 0.0279 0.1654 0.0158 0.3729 0.0951
## SET_Gt        0.0279          0.2702 0.5180 0.5542 0.0836
## SNQt          0.1654 0.2702          0.9812 0.6244 0.4961
## RSMSf1        0.0158 0.5180 0.9812          0.9830 0.2266
## MBIt          0.3729 0.5542 0.6244 0.9830          0.5206
## TQt          0.0951 0.0836 0.4961 0.2266 0.5206

```

```
corr_data_final_r <- round(as.data.frame(corr_data_final$r), 3); corr_data_final_r
```

```

##          sex      age education_level EKM60t  ERTt  TASf1  FP_Gt
## sex          1.000 -0.023          -0.145  0.104 -0.026 -0.035 -0.112
## age          -0.023  1.000          -0.277 -0.056 -0.320  0.040 -0.099
## education_level -0.145 -0.277          1.000  0.182  0.267  0.094  0.143
## EKM60t         0.104 -0.056          0.182  1.000  0.504  0.157  0.299
## ERTt          -0.026 -0.320          0.267  0.504  1.000  0.099  0.418

```

```
## TASf1      -0.035  0.040      0.094  0.157  0.099  1.000  0.113
## FP_Gt      -0.112 -0.099      0.143  0.299  0.418  0.113  1.000
## CART_Gt    -0.354 -0.037      0.379  0.278  0.245  0.012  0.260
## SET_Gt     -0.126 -0.182      0.154  0.108  0.500  0.046  0.174
## SNQt       0.016 -0.095      0.262  0.089  0.091  0.064  0.225
## RSMSf1     -0.221  0.052     -0.009  0.191  0.012  0.249  0.012
## MBIt       0.151  0.306      0.054  0.061 -0.071  0.123 -0.066
## TQt        0.112 -0.053      0.252  0.389  0.257  0.149  0.142
##           CART_Gt SET_Gt  SNQt RSMSf1  MBIt  TQt
## sex        -0.354 -0.126  0.016 -0.221  0.151  0.112
## age        -0.037 -0.182 -0.095  0.052  0.306 -0.053
## education_level  0.379  0.154  0.262 -0.009  0.054  0.252
## EKM60t     0.278  0.108  0.089  0.191  0.061  0.389
## ERTt       0.245  0.500  0.091  0.012 -0.071  0.257
## TASf1      0.012  0.046  0.064  0.249  0.123  0.149
## FP_Gt      0.260  0.174  0.225  0.012 -0.066  0.142
## CART_Gt    1.000  0.194  0.123  0.213  0.079  0.246
## SET_Gt     0.194  1.000  0.098  0.058 -0.053  0.255
## SNQt       0.123  0.098  1.000 -0.002  0.044 -0.102
## RSMSf1     0.213  0.058 -0.002  1.000  0.002  0.180
## MBIt       0.079 -0.053  0.044  0.002  1.000 -0.096
## TQt        0.246  0.255 -0.102  0.180 -0.096  1.000
```

```
corr_data_final_P <- round(as.data.frame(corr_data_final$P), 3); corr_data_final_P
```

```
##           sex   age education_level EKM60t  ERTt TASf1 FP_Gt CART_Gt
## sex           NA 0.799           0.104  0.244 0.770 0.698 0.207  0.000
## age          0.799   NA           0.002  0.529 0.000 0.654 0.267  0.677
## education_level 0.104 0.002           NA  0.040 0.002 0.294 0.108  0.000
## EKM60t         0.244 0.529           0.040   NA 0.000 0.076 0.001  0.001
## ERTt           0.770 0.000           0.002  0.000   NA 0.265 0.000  0.005
## TASf1          0.698 0.654           0.294  0.076 0.265   NA 0.205  0.890
## FP_Gt          0.207 0.267           0.108  0.001 0.000 0.205   NA  0.003
## CART_Gt        0.000 0.677           0.000  0.001 0.005 0.890 0.003   NA
## SET_Gt         0.157 0.040           0.082  0.227 0.000 0.609 0.050  0.028
## SNQt           0.859 0.287           0.003  0.317 0.309 0.473 0.011  0.165
## RSMSf1         0.012 0.561           0.918  0.030 0.892 0.005 0.894  0.016
## MBIt           0.088 0.000           0.545  0.496 0.428 0.168 0.456  0.373
## TQt            0.453 0.722           0.088  0.007 0.081 0.316 0.340  0.095
##           SET_Gt SNQt RSMSf1  MBIt  TQt
## sex           0.157 0.859  0.012 0.088 0.453
## age           0.040 0.287  0.561 0.000 0.722
## education_level 0.082 0.003  0.918 0.545 0.088
## EKM60t        0.227 0.317  0.030 0.496 0.007
## ERTt           0.000 0.309  0.892 0.428 0.081
## TASf1          0.609 0.473  0.005 0.168 0.316
## FP_Gt          0.050 0.011  0.894 0.456 0.340
## CART_Gt        0.028 0.165  0.016 0.373 0.095
## SET_Gt         NA 0.270  0.518 0.554 0.084
## SNQt           0.270   NA  0.981 0.624 0.496
## RSMSf1         0.518 0.981   NA 0.983 0.227
## MBIt           0.554 0.624  0.983   NA 0.521
## TQt            0.084 0.496  0.227 0.521   NA
```

```
#knitr::kable(corr_data_final_r, format = "latex", booktabs = TRUE, caption = "Correlation matrix final")
#knitr::kable(corr_data_final_P, format = "latex", booktabs = TRUE, caption = "P-values corresponding to")
```

6.2 Z-scores

```
names_items_analysis <- c("EKM60t", "ERTt", "TASf1", "CART_Gt", "FP_Gt", "SET_Gt", "SNQt", "MBIt", "RSM")
data_final_Z <- cbind(data_final[, c("ID", "centerID", names_covariates)], round(scale(data_final[, c(n
```

7 Analyses

7.1 CFA models

```
#full 3 factor model
m1_f3fm <- '
# Defining the factors (latent variables)
f1_P =~ EKM60t + ERTt + TASf1
f2_U =~ SET_Gt + FP_Gt + CART_Gt
f3_BR =~ RSMSf1 + SNQt + MBIt

# Allow factors to be correlated
f1_P ~~ f2_U
f1_P ~~ f3_BR
f2_U ~~ f3_BR
'

#two factor model: f1 = f2
m2_2fm_1eq2 <- '
# Defining the factors (latent variables)
f1_P =~ EKM60t + ERTt + TASf1
f2_U =~ SET_Gt + FP_Gt + CART_Gt
f3_BR =~ RSMSf1 + SNQt + MBIt

# Allow factors to be correlated
f1_P ~~ 1*f2_U # Fix correlation between f1_P and f2_U at 1
f1_P ~~ f3_BR
f2_U ~~ f3_BR
'

#model 3: two factor model: f1 = f3 (CART)
m3_2fm_1eq3 <- '
# Defining the factors (latent variables)
f1_P =~ EKM60t + ERTt + TASf1
f2_U =~ SET_Gt + FP_Gt + CART_Gt
f3_BR =~ RSMSf1 + SNQt + MBIt

# Allow factors to be correlated
f1_P ~~ f2_U
f1_P ~~ 1*f3_BR # Fix correlation between f1_P and f3_BR at 1
```



```

f2_U ~~ f3_BR
'

#model 4: two factor model: f2 = f3 (CART)
m4_2fm_2eq3 <- '
# Defining the factors (latent variables)
f1_P =~ EKM60t + ERTt + TASf1
f2_U =~ SET_Gt + FP_Gt + CART_Gt
f3_BR =~ RSMSf1 + SNQt + MBIt

# Allow factors to be correlated
f1_P ~~ f2_U
f1_P ~~ f3_BR
f2_U ~~ 1*f3_BR # Fix correlation between f2_U and f3_BR at 1
'

#model 5: one factor model: f1 = f2 = f3 (CART)
m5_1fm <- '
# Defining the factors (latent variables)
f1_P =~ EKM60t + ERTt + TASf1
f2_U =~ SET_Gt + FP_Gt + CART_Gt
f3_BR =~ RSMSf1 + SNQt + MBIt

# Fixing all interfactor correlations at 1
f1_P ~~ 1*f2_U
f1_P ~~ 1*f3_BR
f2_U ~~ 1*f3_BR
'

#model 6: 3 one factor model: independent factors (CART)
m6_3x1fm <- '
# Defining the factors (latent variables)
f1_P =~ EKM60t + ERTt + TASf1
f2_U =~ SET_Gt + FP_Gt + CART_Gt
f3_BR =~ RSMSf1 + SNQt + MBIt

# Fixing all interfactor correlations at 0
f1_P ~~ 0*f2_U
f1_P ~~ 0*f3_BR
f2_U ~~ 0*f3_BR
'

```

7.2 CFA EMC data (N=47)

```

#model 1; EMC data
fit_m1 EMC <- cfa(model = m1_f3fm, data = data_final_Z[data_final_Z$centerID == "EMC",])

## Warning: lavaan->lav_object_post_check():
##      some estimated lv variances are negative

```

```

summ_m1 EMC <- summary(fit_m1 EMC, standardized = TRUE, fit.measures = TRUE)
fm_m1 EMC <- fitMeasures(fit_m1 EMC, c("npar", "chisq", "df", "pvalue", "aic", "srmr", "cfi", "ifi", "n

#model 2; EMC data
fit_m2 EMC <- cfa(model = m2_2fm_1eq2, data = data_final_Z[data_final_Z$centerID == "EMC",])

## Warning: lavaan->lav_start_check_cov():
##   starting values imply a correlation larger than 1; variables involved are:
##   f1_P f2_U

## Warning: lavaan->lav_object_post_check():
##   some estimated lv variances are negative

summ_m2 EMC <- summary(fit_m2 EMC, standardized = TRUE, fit.measures = TRUE)
fm_m2 EMC <- fitMeasures(fit_m2 EMC, c("npar", "chisq", "df", "pvalue", "aic", "srmr", "cfi", "ifi", "n

#model 3; EMC data
fit_m3 EMC <- cfa(model = m3_2fm_1eq3, data = data_final_Z[data_final_Z$centerID == "EMC",])

## Warning: lavaan->lav_start_check_cov():
##   starting values imply a correlation larger than 1; variables involved are:
##   f1_P f3_BR

## Warning: lavaan->lav_object_post_check():
##   some estimated lv variances are negative

summ_m3 EMC <- summary(fit_m3 EMC, standardized = TRUE, fit.measures = TRUE)
fm_m3 EMC <- fitMeasures(fit_m3 EMC, c("npar", "chisq", "df", "pvalue", "aic", "srmr", "cfi", "ifi", "n

#model 4; EMC data
fit_m4 EMC <- cfa(model = m4_2fm_2eq3, data = data_final_Z[data_final_Z$centerID == "EMC",])

## Warning: lavaan->lav_start_check_cov():
##   starting values imply a correlation larger than 1; variables involved are:
##   f2_U f3_BR

## Warning: lavaan->lav_object_post_check():
##   some estimated lv variances are negative

summ_m4 EMC <- summary(fit_m4 EMC, standardized = TRUE, fit.measures = TRUE)
fm_m4 EMC <- fitMeasures(fit_m4 EMC, c("npar", "chisq", "df", "pvalue", "aic", "srmr", "cfi", "ifi", "n

#model 5; EMC data
fit_m5 EMC <- cfa(model = m5_1fm, data = data_final_Z[data_final_Z$centerID == "EMC",])

## Warning: lavaan->lav_start_check_cov():
##   starting values imply a correlation larger than 1; variables involved are:
##   f1_P f2_U

```

```
## Warning: lavaan->lav_start_check_cov():
##   starting values imply a correlation larger than 1; variables involved are:
##   f1_P f3_BR

## Warning: lavaan->lav_object_post_check():
##   some estimated lv variances are negative

summ_m5 EMC <- summary(fit_m5 EMC, standardized = TRUE, fit.measures = TRUE)
fm_m5 EMC <- fitMeasures(fit_m5 EMC, c("npar", "chisq", "df", "pvalue", "aic", "srmr", "cfi", "ifi", "n

# #model 6; EMC data
# fit_m6 EMC <- cfa(model = m6_3x1fm, data = data_final_Z[data_final_Z$centerID == "EMC",])
# summ_m6 EMC <- summary(fit_m6 EMC, standardized = TRUE, fit.measures = TRUE)
# fm_m6 EMC <- fitMeasures(fit_m6 EMC, c("npar", "chisq", "df", "pvalue", "aic", "srmr", "cfi", "ifi", "n
```

7.3 CFA pooled data (N=128)

```
#model 1; pooled data
fit_m1_pld <- cfa(model = m1_f3fm, data = data_final_Z)

## Warning: lavaan->lav_object_post_check():
##   some estimated lv variances are negative

summ_m1_pld <- summary(fit_m1_pld, standardized = TRUE, fit.measures = TRUE)
fm_m1_pld <- fitMeasures(fit_m1_pld, c("npar", "chisq", "df", "pvalue", "aic", "srmr", "cfi", "ifi", "n

#model 2; pooled data
fit_m2_pld <- cfa(model = m2_2fm_1eq2, data = data_final_Z)

## Warning: lavaan->lav_start_check_cov():
##   starting values imply a correlation larger than 1; variables involved are:
##   f1_P f2_U

## Warning: lavaan->lav_object_post_check():
##   some estimated lv variances are negative

summ_m2_pld <- summary(fit_m2_pld, standardized = TRUE, fit.measures = TRUE)
fm_m2_pld <- fitMeasures(fit_m2_pld, c("npar", "chisq", "df", "pvalue", "aic", "srmr", "cfi", "ifi", "n

#model 3; pooled data
fit_m3_pld <- cfa(model = m3_2fm_1eq3, data = data_final_Z)

## Warning: lavaan->lav_start_check_cov():
##   starting values imply a correlation larger than 1; variables involved are:
##   f1_P f3_BR

## Warning: lavaan->lav_object_post_check():
##   covariance matrix of latent variables is not positive definite ; use
##   lavInspect(fit, "cov.lv") to investigate.
```

```
summ_m3_pld <- summary(fit_m3_pld, standardized = TRUE, fit.measures = TRUE)
fm_m3_pld <- fitMeasures(fit_m3_pld, c("npar", "chisq", "df", "pvalue", "aic", "srmr", "cfi", "ifi", "n
```

```
#model 4; pooled data
fit_m4_pld <- cfa(model = m4_2fm_2eq3, data = data_final_Z)
```

```
## Warning: lavaan->lav_start_check_cov():
##   starting values imply a correlation larger than 1; variables involved are:
##   f2_U f3_BR
```

```
## Warning: lavaan->lav_object_post_check():
##   some estimated ov variances are negative
```

```
## Warning: lavaan->lav_object_post_check():
##   covariance matrix of latent variables is not positive definite ; use
##   lavInspect(fit, "cov.lv") to investigate.
```

```
summ_m4_pld <- summary(fit_m4_pld, standardized = TRUE, fit.measures = TRUE)
fm_m4_pld <- fitMeasures(fit_m4_pld, c("npar", "chisq", "df", "pvalue", "aic", "srmr", "cfi", "ifi", "n
```

```
#model 5; pooled data
fit_m5_pld <- cfa(model = m5_1fm, data = data_final_Z)
```

```
## Warning: lavaan->lav_start_check_cov():
##   starting values imply a correlation larger than 1; variables involved are:
##   f1_P f2_U
```

```
## Warning: lavaan->lav_start_check_cov():
##   starting values imply a correlation larger than 1; variables involved are:
##   f1_P f3_BR
```

```
## Warning: lavaan->lav_object_post_check():
##   covariance matrix of latent variables is not positive definite ; use
##   lavInspect(fit, "cov.lv") to investigate.
```

```
summ_m5_pld <- summary(fit_m5_pld, standardized = TRUE, fit.measures = TRUE)
fm_m5_pld <- fitMeasures(fit_m5_pld, c("npar", "chisq", "df", "pvalue", "aic", "srmr", "cfi", "ifi", "n
```

```
#model 6; pooled data
fit_m6_pld <- cfa(model = m6_3x1fm, data = data_final_Z)
```

```
## Warning: lavaan->lav_object_post_check():
##   some estimated lv variances are negative
```

```
summ_m6_pld <- summary(fit_m6_pld, standardized = TRUE, fit.measures = TRUE)
fm_m6_pld <- fitMeasures(fit_m6_pld, c("npar", "chisq", "df", "pvalue", "aic", "srmr", "cfi", "ifi", "n
```

7.4 SEM models

```

#3 paths model
#full 3 factor model
sem1 <- '
# Defining the factors (latent variables)
f1_P =~ EKM60t + ERTt + TASf1
f2_U =~ SET_Gt + FP_Gt + CART_Gt
f3_BR =~ RSMSf1 + SNQt + MBIt

#regression
TQt ~ f1_P + f2_U + f3_BR

# Allow factors to be correlated
f1_P ~~ f2_U
f1_P ~~ f3_BR
f2_U ~~ f3_BR
'

#2 path models
#path P->TQ=0
sem2 <- '
# Defining the factors (latent variables)
f1_P =~ EKM60t + ERTt + TASf1
f2_U =~ SET_Gt + FP_Gt + CART_Gt
f3_BR =~ RSMSf1 + SNQt + MBIt

#regression
TQt ~ 0*f1_P + f2_U + f3_BR

# Allow factors to be correlated
f1_P ~~ f2_U
f1_P ~~ f3_BR
f2_U ~~ f3_BR
'

#path U->TQ=0
sem3 <- '
# Defining the factors (latent variables)
f1_P =~ EKM60t + ERTt + TASf1
f2_U =~ SET_Gt + FP_Gt + CART_Gt
f3_BR =~ RSMSf1 + SNQt + MBIt

#regression
TQt ~ f1_P + 0*f2_U + f3_BR

# Allow factors to be correlated
f1_P ~~ f2_U
f1_P ~~ f3_BR
f2_U ~~ f3_BR
'

#path BR->TQ=0
sem4 <- '
# Defining the factors (latent variables)

```

```

f1_P =~ EKM60t + ERTt + TASf1
f2_U =~ SET_Gt + FP_Gt + CART_Gt
f3_BR =~ RSMSf1 + SNQt + MBIt

#regression
TQt ~ f1_P + f2_U + 0*f3_BR

# Allow factors to be correlated
f1_P ~~ f2_U
f1_P ~~ f3_BR
f2_U ~~ f3_BR
'

#1 path models
#only path P
sem5 <- '
# Defining the factors (latent variables)
f1_P =~ EKM60t + ERTt + TASf1
f2_U =~ SET_Gt + FP_Gt + CART_Gt
f3_BR =~ RSMSf1 + SNQt + MBIt

#regression
TQt ~ f1_P + 0*f2_U + 0*f3_BR

# Allow factors to be correlated
f1_P ~~ f2_U
f1_P ~~ f3_BR
f2_U ~~ f3_BR
'

#only path U
sem6 <- '
# Defining the factors (latent variables)
f1_P =~ EKM60t + ERTt + TASf1
f2_U =~ SET_Gt + FP_Gt + CART_Gt
f3_BR =~ RSMSf1 + SNQt + MBIt

#regression
TQt ~ 0*f1_P + f2_U + 0*f3_BR

# Allow factors to be correlated
f1_P ~~ f2_U
f1_P ~~ f3_BR
f2_U ~~ f3_BR
'

#only path BR
sem7 <- '
# Defining the factors (latent variables)
f1_P =~ EKM60t + ERTt + TASf1
f2_U =~ SET_Gt + FP_Gt + CART_Gt
f3_BR =~ RSMSf1 + SNQt + MBIt

```

```

#regression
TQt ~ 0*f1_P + 0*f2_U + f3_BR

# Allow factors to be correlated
f1_P ~~ f2_U
f1_P ~~ f3_BR
f2_U ~~ f3_BR
'

#no path model (baseline)
sem8 <- '
# Defining the factors (latent variables)
f1_P =~ EKM6Ot + ERTt + TASf1
f2_U =~ SET_Gt + FP_Gt + CART_Gt
f3_BR =~ RMSsf1 + SNQt + MBIt

#regression
TQt ~ 0*f1_P + 0*f2_U + 0*f3_BR

# Allow factors to be correlated
f1_P ~~ f2_U
f1_P ~~ f3_BR
f2_U ~~ f3_BR
'

```

7.5 SEM EMC data (N=47)

```

#3 paths model
fit_sem1 <- sem(model = sem1, data = data_final_Z[data_final_Z$centerID == 'EMC',])

## Warning: lavaan->lav_object_post_check():
## some estimated lv variances are negative

summ_sem1 <- summary(fit_sem1, standardized = TRUE, fit.measures = TRUE)
fm_sem1 <- fitMeasures(fit_sem1, c("npar", "chisq", "df", "pvalue", "aic", "srmsr", "cfi", "ifi", "nfi",
stcoef_sem1 <- standardizedSolution(fit_sem1)[10:12,1:7]

#2 path models
fit_sem2 <- sem(model = sem2, data = data_final_Z[data_final_Z$centerID == 'EMC',])

## Warning: lavaan->lav_object_post_check():
## some estimated lv variances are negative

summ_sem2 <- summary(fit_sem2, standardized = TRUE, fit.measures = TRUE)
fm_sem2 <- fitMeasures(fit_sem2, c("npar", "chisq", "df", "pvalue", "aic", "srmsr", "cfi", "ifi", "nfi",
stcoef_sem2 <- standardizedSolution(fit_sem2)[10:12,1:7]

fit_sem3 <- sem(model = sem3, data = data_final_Z[data_final_Z$centerID == 'EMC',])

```

```
## Warning: lavaan->lav_object_post_check():  
##      some estimated lv variances are negative
```

```
summ_sem3 <- summary(fit_sem3, standardized = TRUE, fit.measures = TRUE)  
fm_sem3 <- fitMeasures(fit_sem3, c("npar", "chisq", "df", "pvalue", "aic", "srmr", "cfi", "ifi", "nfi",  
stcoef_sem3 <- standardizedSolution(fit_sem3)[10:12,1:7]
```

```
fit_sem4 <- sem(model = sem4, data = data_final_Z[data_final_Z$centerID == 'EMC',])
```

```
## Warning: lavaan->lav_object_post_check():  
##      some estimated lv variances are negative
```

```
summ_sem4 <- summary(fit_sem4, standardized = TRUE, fit.measures = TRUE)  
fm_sem4 <- fitMeasures(fit_sem4, c("npar", "chisq", "df", "pvalue", "aic", "srmr", "cfi", "ifi", "nfi",  
stcoef_sem4 <- standardizedSolution(fit_sem4)[10:12,1:7]
```

#1 path models

```
fit_sem5 <- sem(model = sem5, data = data_final_Z[data_final_Z$centerID == 'EMC',])
```

```
## Warning: lavaan->lav_object_post_check():  
##      some estimated lv variances are negative
```

```
summ_sem5 <- summary(fit_sem5, standardized = TRUE, fit.measures = TRUE)  
fm_sem5 <- fitMeasures(fit_sem5, c("npar", "chisq", "df", "pvalue", "aic", "srmr", "cfi", "ifi", "nfi",  
stcoef_sem5 <- standardizedSolution(fit_sem5)[10:12,1:7]
```

```
fit_sem6 <- sem(model = sem6, data = data_final_Z[data_final_Z$centerID == 'EMC',])
```

```
## Warning: lavaan->lav_object_post_check():  
##      some estimated lv variances are negative
```

```
summ_sem6 <- summary(fit_sem6, standardized = TRUE, fit.measures = TRUE)  
fm_sem6 <- fitMeasures(fit_sem6, c("npar", "chisq", "df", "pvalue", "aic", "srmr", "cfi", "ifi", "nfi",  
stcoef_sem6 <- standardizedSolution(fit_sem6)[10:12,1:7]
```

```
fit_sem7 <- sem(model = sem7, data = data_final_Z[data_final_Z$centerID == 'EMC',])
```

```
## Warning: lavaan->lav_object_post_check():  
##      some estimated lv variances are negative
```

```
summ_sem7 <- summary(fit_sem7, standardized = TRUE, fit.measures = TRUE)  
fm_sem7 <- fitMeasures(fit_sem7, c("npar", "chisq", "df", "pvalue", "aic", "srmr", "cfi", "ifi", "nfi",  
stcoef_sem7 <- standardizedSolution(fit_sem7)[10:12,1:7]
```

#no path model

```
fit_sem8 <- sem(model = sem8, data = data_final_Z[data_final_Z$centerID == 'EMC',])
```



```
## Warning: lavaan->lav_object_post_check():
##      some estimated lv variances are negative
```

```
summ_sem8 <- summary(fit_sem8, standardized = TRUE, fit.measures = TRUE)
fm_sem8 <- fitMeasures(fit_sem8, c("npar", "chisq", "df", "pvalue", "aic", "srmr", "cfi", "ifi", "nfi",
stcoef_sem8 <- standardizedSolution(fit_sem8)[10:12,1:7]
```

8 Results

8.1 CFA EMC data

8.1.1 Fit indices

```
results_CFA EMC <- data.frame(round(rbind(fm_m1 EMC, fm_m2 EMC, fm_m3 EMC, fm_m4 EMC, fm_m5 EMC
                                     #, fm_m6 EMC
                                     ), 3))
modelNames_CFA <- c("Full three-factor", "Two-factor: P = U", "Two-factor: P = BR", "Two-factor: U = BR",
                    #, "Independent three factor"
                    )
rownames(results_CFA EMC) <- NULL
results_CFA EMC <- cbind(Model = modelNames_CFA, results_CFA EMC)
results_CFA EMC
```

```
##           Model npar  chisq df pvalue      aic  srmr   cfi   ifi   nfi
## 1      Full three-factor   21 25.952 24  0.356 1176.076 0.084 0.962 0.969 0.705
## 2      Two-factor: P = U   20 43.386 25  0.013 1191.510 0.374 0.646 0.708 0.506
## 3      Two-factor: P = BR   20 35.310 25  0.083 1183.434 0.248 0.801 0.836 0.598
## 4      Two-factor: U = BR   20 49.263 25  0.003 1197.387 0.432 0.532 0.614 0.439
## 5 One-factor: P = U = BR   18 50.079 27  0.004 1194.204 0.464 0.555 0.621 0.430
##      rmsea      bic
## 1 0.042 1214.929
## 2 0.125 1228.513
## 3 0.094 1220.437
## 4 0.144 1234.390
## 5 0.135 1227.506
```

8.1.2 Model comparisson

```
anova(fit_m1 EMC, fit_m2 EMC)
```

```
##
## Chi-Squared Difference Test
##
##           Df      AIC      BIC  Chisq Chisq diff  RMSEA Df diff Pr(>Chisq)
## fit_m1 EMC 24 1176.1 1214.9 25.952
## fit_m2 EMC 25 1191.5 1228.5 43.386      17.434 0.59133      1 2.974e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
anova(fit_m1 EMC, fit_m3 EMC)
```

```
##
## Chi-Squared Difference Test
##
##           Df      AIC      BIC  Chisq Chisq diff  RMSEA Df diff Pr(>Chisq)
## fit_m1 EMC 24 1176.1 1214.9 25.952
## fit_m3 EMC 25 1183.4 1220.4 35.310      9.3578 0.42169      1 0.00222 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
anova(fit_m1 EMC, fit_m4 EMC)
```

```
##
## Chi-Squared Difference Test
##
##           Df      AIC      BIC  Chisq Chisq diff  RMSEA Df diff Pr(>Chisq)
## fit_m1 EMC 24 1176.1 1214.9 25.952
## fit_m4 EMC 25 1197.4 1234.4 49.263      23.311 0.68898      1 1.378e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
anova(fit_m1 EMC, fit_m5 EMC)
```

```
##
## Chi-Squared Difference Test
##
##           Df      AIC      BIC  Chisq Chisq diff  RMSEA Df diff Pr(>Chisq)
## fit_m1 EMC 24 1176.1 1214.9 25.952
## fit_m5 EMC 27 1194.2 1227.5 50.079      24.128 0.38709      3 2.349e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
#anova(fit_m1 EMC, fit_m6 EMC)
```

8.1.3 Favoured model

```
stcoefs_fav_CFA EMC <- standardizedSolution(fit_m1 EMC)
nice_table(stcoefs_fav_CFA EMC)
```

```
## Warning: fonts used in 'flextable' are ignored because the 'pdflatex' engine is
## used and not 'xelatex' or 'lualatex'. You can avoid this warning by using the
## 'set_flextable_defaults(fonts_ignore=TRUE)' command or use a compatible engine
## by defining 'latex_engine: xelatex' in the YAML header of the R Markdown
## document.
```

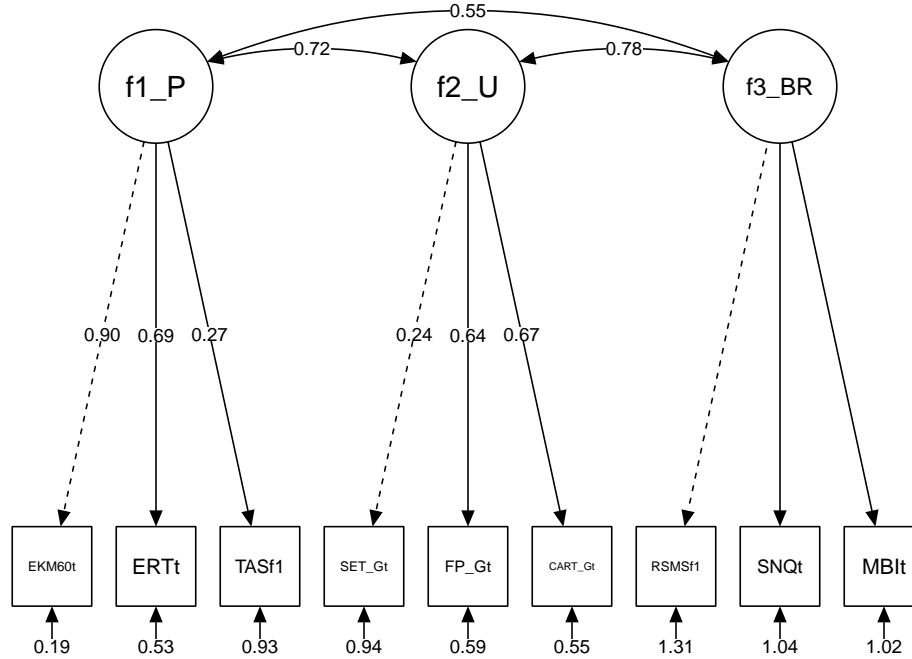
lhs	op	rhs	est.std	se	z	p	ci.lower	ci.upper
f1_P	=~	EKM60t	0.90	0.11	8.49	< .001***	0.69	1.11
f1_P	=~	ERTt	0.69	0.11	6.22	< .001***	0.47	0.90
f1_P	=~	TASf1	0.27	0.15	1.79	.074	-0.03	0.56
f2_U	=~	SET_Gt	0.24	0.16	1.48	.139	-0.08	0.56
f2_U	=~	FP_Gt	0.64	0.13	5.12	< .001***	0.40	0.89
f2_U	=~	CART_Gt	0.67	0.12	5.36	< .001***	0.42	0.91
f3_BR	=~	RSMSf1						
f3_BR	=~	SNQt						
f3_BR	=~	MBIt						
f1_P	~~	f2_U	0.72	0.15	4.74	< .001***	0.42	1.02
f1_P	~~	f3_BR	0.55	0.46	1.19	.234	-0.36	1.45
f2_U	~~	f3_BR	0.78	0.62	1.26	.208	-0.43	2.00
EKM60t	~~	EKM60t	0.19	0.19	0.97	.333	-0.19	0.56
ERTt	~~	ERTt	0.53	0.15	3.49	< .001***	0.23	0.83
TASf1	~~	TASf1	0.93	0.08	11.75	< .001***	0.77	1.08
SET_Gt	~~	SET_Gt	0.94	0.08	12.01	< .001***	0.79	1.10
FP_Gt	~~	FP_Gt	0.59	0.16	3.63	< .001***	0.27	0.90
CART_Gt	~~	CART_Gt	0.55	0.17	3.32	.001***	0.23	0.88
RSMSf1	~~	RSMSf1	1.31	0.48	2.71	.007**	0.36	2.25
SNQt	~~	SNQt	1.04	0.06	18.30	< .001***	0.93	1.15
MBIt	~~	MBIt	1.02	0.04	25.44	< .001***	0.94	1.10
f1_P	~~	f1_P	1.00	0.00			1.00	1.00
f2_U	~~	f2_U	1.00	0.00			1.00	1.00
f3_BR	~~	f3_BR						

```

# print(nice_table(stcoefs_fav_CFA EMC), preview = "docx") # Word format output
semPaths(fit_m1 EMC,
  whatLabels = "std", # Display standardized estimates
  layout = "tree2", # Choose a layout, "tree" is one option
  style = "lisrel", # Style for the diagram
  nCharNodes = 0, # Full variable names
  sizeMan = 7, # Size of manifest variables
  sizeLat = 10, # Size of latent variables
  edge.label.cex = 0.75, # Adjust the size of the edge labels
  label.cex = 0.8, # Adjust the size of the node labels
  residuals = TRUE, # Show residuals
  intercepts = FALSE, # Hide intercepts
  thresholds = FALSE,
  fade = FALSE,
  edge.color = "black"

```

)



8.2 CFA pooled data

8.2.1 Fit indices

```
results_CFA_pld <- data.frame(round(rbind(fm_m1_pld, fm_m2_pld, fm_m3_pld, fm_m4_pld, fm_m5_pld, fm_m6_pld), 3))
modelNames_CFA <- c("Full three-factor", "Two-factor: P = U", "Two-factor: P = BR", "Two-factor: U = BR", "One-factor: P = U = BR", "Independent three factor")
rownames(results_CFA_pld) <- NULL
results_CFA_pld <- cbind(Model = modelNames_CFA, results_CFA_pld)
results_CFA_pld
```

##	Model	npar	chisq	df	pvalue	aic	srmr	cfi	ifi
## 1	Full three-factor	21	45.363	24	0.005	3185.383	0.075	0.831	0.845
## 2	Two-factor: P = U	20	80.179	25	0.000	3218.199	0.298	0.563	0.598
## 3	Two-factor: P = BR	20	106.314	25	0.000	3244.334	0.317	0.356	0.407
## 4	Two-factor: U = BR	20	95.357	25	0.000	3233.377	0.294	0.442	0.487
## 5	One-factor: P = U = BR	18	116.076	27	0.000	3250.096	0.457	0.294	0.341
## 6	Independent three factor	18	105.401	27	0.000	3239.420	0.146	0.379	0.420

##	nfi	rmsea	bic
## 1	0.720	0.083	3245.275
## 2	0.506	0.131	3275.239
## 3	0.344	0.159	3301.374

```
## 4 0.412 0.148 3290.417
## 5 0.284 0.161 3301.432
## 6 0.350 0.151 3290.757
```

8.2.2 Model comparisson

```
anova(fit_m1_pld, fit_m2_pld)
```

```
##
## Chi-Squared Difference Test
##
##           Df      AIC      BIC  Chisq Chisq diff  RMSEA Df diff Pr(>Chisq)
## fit_m1_pld 24 3185.4 3245.3 45.363
## fit_m2_pld 25 3218.2 3275.2 80.179      34.816 0.51399      1 3.624e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
anova(fit_m1_pld, fit_m3_pld)
```

```
##
## Chi-Squared Difference Test
##
##           Df      AIC      BIC  Chisq Chisq diff  RMSEA Df diff Pr(>Chisq)
## fit_m1_pld 24 3185.4 3245.3 45.363
## fit_m3_pld 25 3244.3 3301.4 106.314      60.951 0.68438      1 5.85e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
anova(fit_m1_pld, fit_m4_pld)
```

```
##
## Chi-Squared Difference Test
##
##           Df      AIC      BIC  Chisq Chisq diff  RMSEA Df diff Pr(>Chisq)
## fit_m1_pld 24 3185.4 3245.3 45.363
## fit_m4_pld 25 3233.4 3290.4 95.357      49.994 0.61868      1 1.542e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
anova(fit_m1_pld, fit_m5_pld)
```

```
##
## Chi-Squared Difference Test
##
##           Df      AIC      BIC  Chisq Chisq diff  RMSEA Df diff Pr(>Chisq)
## fit_m1_pld 24 3185.4 3245.3 45.363
## fit_m5_pld 27 3250.1 3301.4 116.076      70.713 0.41992      3 3.003e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
anova(fit_m1_pld, fit_m6_pld)
```

```
##
## Chi-Squared Difference Test
##
##           Df      AIC      BIC   Chisq Chisq diff  RMSEA Df diff Pr(>Chisq)
## fit_m1_pld 24 3185.4 3245.3  45.363
## fit_m6_pld 27 3239.4 3290.8 105.401    60.038 0.3854      3 5.77e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# anova(fit_m2_pld, fit_m5_pld)
# anova(fit_m3_pld, fit_m5_pld)
# anova(fit_m4_pld, fit_m5_pld)
```

8.2.3 Favoured model

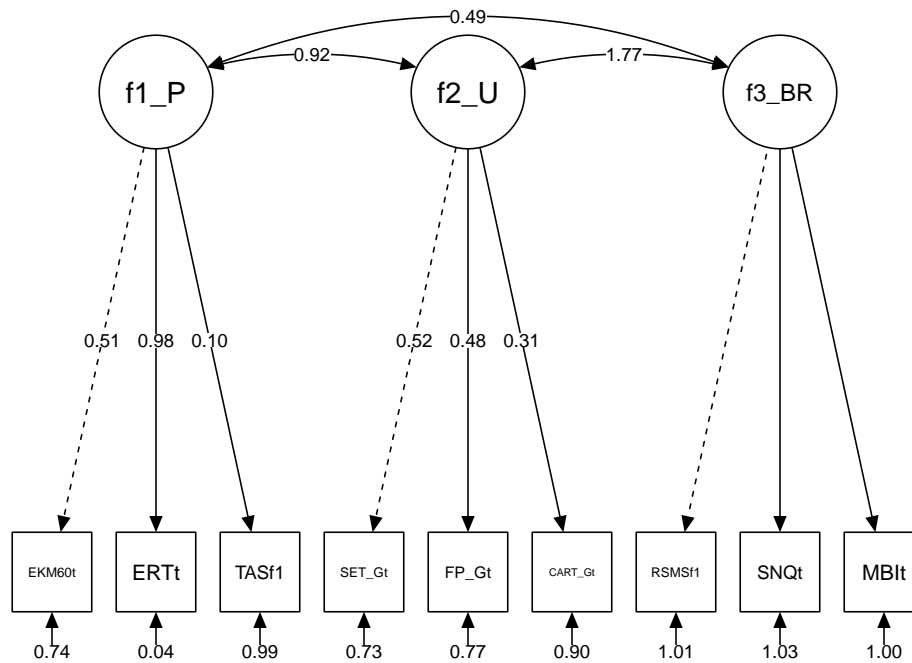
```
stcoefs_fav_CFA_pld <- standardizedSolution(fit_m1_pld)
nice_table(stcoefs_fav_CFA_pld)
```

```
## Warning: fonts used in 'flextable' are ignored because the 'pdflatex' engine is
## used and not 'xelatex' or 'lualatex'. You can avoid this warning by using the
## 'set_flextable_defaults(fonts_ignore=TRUE)' command or use a compatible engine
## by defining 'latex_engine: xelatex' in the YAML header of the R Markdown
## document.
```

lhs	op	rhs	est.std	se	z	p	ci.lower	ci.upper
f1_P	==	EKM60t	0.51	0.08	6.29	< .001***	0.35	0.67
f1_P	==	ERTt	0.98	0.09	10.46	< .001***	0.80	1.17
f1_P	==	TASf1	0.10	0.09	1.16	.244	-0.07	0.28
f2_U	==	SET_Gt	0.52	0.09	5.97	< .001***	0.35	0.69
f2_U	==	FP_Gt	0.48	0.09	5.48	< .001***	0.31	0.65
f2_U	==	CART_Gt	0.31	0.09	3.50	< .001***	0.14	0.49
f3_BR	==	RSMSf1						
f3_BR	==	SNQt						
f3_BR	==	MBIt						
f1_P	~~	f2_U	0.92	0.14	6.73	< .001***	0.65	1.19
f1_P	~~	f3_BR	0.49	1.31	0.38	.707	-2.07	3.05
f2_U	~~	f3_BR	1.77	4.48	0.39	.693	-7.02	10.56
EKM60t	~~	EKM60t	0.74	0.08	8.82	< .001***	0.57	0.90
ERTt	~~	ERTt	0.04	0.18	0.19	.849	-0.33	0.40

lhs	op	rhs	est.std	se	z	p	ci.lower	ci.upper
TASf1	~~	TASf1	0.99	0.02	53.07	< .001***	0.95	1.03
SET_Gt	~~	SET_Gt	0.73	0.09	8.14	< .001***	0.56	0.91
FP_Gt	~~	FP_Gt	0.77	0.08	9.35	< .001***	0.61	0.94
CART_Gt	~~	CART_Gt	0.90	0.06	16.18	< .001***	0.79	1.01
RSMSf1	~~	RSMSf1	1.01	0.04	25.35	< .001***	0.93	1.09
SNQt	~~	SNQt	1.03	0.17	6.11	< .001***	0.70	1.37
MBIt	~~	MBIt	1.00	0.01	128.58	< .001***	0.99	1.02
f1_P	~~	f1_P	1.00	0.00			1.00	1.00
f2_U	~~	f2_U	1.00	0.00			1.00	1.00
f3_BR	~~	f3_BR						

```
#print(nice_table(stcoefs_fav_CFA_pld), preview = "docx") #Word format output
semPaths(fit_m1_pld,
  whatLabels = "std", # Display standardized estimates
  layout = "tree2", # Choose a layout, "tree" is one option
  style = "lisrel", # Style for the diagram
  nCharNodes = 0, # Full variable names
  sizeMan = 7, # Size of manifest variables
  sizeLat = 10, # Size of latent variables
  edge.label.cex = 0.75, # Adjust the size of the edge labels
  label.cex = 0.8, # Adjust the size of the node labels
  residuals = TRUE, # Show residuals
  intercepts = FALSE, # Hide intercepts
  thresholds = FALSE,
  fade = FALSE,
  edge.color = "black"
)
```



8.3 SEM EMC data

8.3.1 Fit indices

```
results_SEM <- data.frame(round(rbind(fm_sem1, fm_sem2, fm_sem3, fm_sem4, fm_sem5, fm_sem6, fm_sem7, fm_sem8, fm_sem9, fm_sem10, fm_sem11, fm_sem12, fm_sem13, fm_sem14, fm_sem15, fm_sem16, fm_sem17, fm_sem18, fm_sem19, fm_sem20, fm_sem21, fm_sem22, fm_sem23, fm_sem24, fm_sem25, fm_sem26, fm_sem27, fm_sem28, fm_sem29, fm_sem30, fm_sem31, fm_sem32, fm_sem33, fm_sem34, fm_sem35, fm_sem36, fm_sem37, fm_sem38, fm_sem39, fm_sem40, fm_sem41, fm_sem42, fm_sem43, fm_sem44, fm_sem45, fm_sem46, fm_sem47, fm_sem48, fm_sem49, fm_sem50, fm_sem51, fm_sem52, fm_sem53, fm_sem54, fm_sem55, fm_sem56, fm_sem57, fm_sem58, fm_sem59, fm_sem60, fm_sem61, fm_sem62, fm_sem63, fm_sem64, fm_sem65, fm_sem66, fm_sem67, fm_sem68, fm_sem69, fm_sem70, fm_sem71, fm_sem72, fm_sem73, fm_sem74, fm_sem75, fm_sem76, fm_sem77, fm_sem78, fm_sem79, fm_sem80, fm_sem81, fm_sem82, fm_sem83, fm_sem84, fm_sem85, fm_sem86, fm_sem87, fm_sem88, fm_sem89, fm_sem90, fm_sem91, fm_sem92, fm_sem93, fm_sem94, fm_sem95, fm_sem96, fm_sem97, fm_sem98, fm_sem99, fm_sem100), 2))
modelNames_SEM <- c("3p", "2p: U & BR", "2p: P & BR", "2p: P & U", "1p: P", "1p: U", "1p: BR", "0p")
rownames(results_SEM) <- NULL
results_SEM <- cbind(Model = modelNames_SEM, results_SEM)
results_SEM
```

```
##      Model npar  chisq df pvalue      aic  srmr   cfi   ifi   nfi rmsea
## 1      3p    25 30.673 30  0.432 1308.220 0.084 0.988 0.990 0.696 0.022
## 2 2p: U & BR  24 32.021 31  0.416 1307.567 0.083 0.982 0.985 0.682 0.026
## 3 2p: P & BR  24 30.674 31  0.483 1306.221 0.084 1.000 1.005 0.696 0.000
## 4 2p: P & U   24 31.052 31  0.464 1306.598 0.085 0.999 0.999 0.692 0.006
## 5    1p: P    23 31.053 32  0.514 1304.600 0.085 1.000 1.014 0.692 0.000
## 6    1p: U    23 33.028 32  0.417 1306.574 0.085 0.982 0.985 0.672 0.026
## 7    1p: BR   23 32.655 32  0.435 1306.201 0.087 0.988 0.990 0.676 0.021
## 8      0p    22 38.901 33  0.221 1310.447 0.117 0.894 0.913 0.614 0.062
##      bic
## 1 1354.473
## 2 1351.971
## 3 1350.624
```



```
## 4 1351.002
## 5 1347.153
## 6 1349.127
## 7 1348.754
## 8 1351.150
```

```
results_SEM[,c("Model", "npar", "df", "chisq", "pvalue", "aic", "srmr", "cfi", "ifi")]
```

```
##      Model npar df  chisq pvalue      aic  srmr  cfi  ifi
## 1      3p   25 30 30.673  0.432 1308.220 0.084 0.988 0.990
## 2 2p: U & BR 24 31 32.021  0.416 1307.567 0.083 0.982 0.985
## 3 2p: P & BR 24 31 30.674  0.483 1306.221 0.084 1.000 1.005
## 4 2p: P & U  24 31 31.052  0.464 1306.598 0.085 0.999 0.999
## 5      1p: P  23 32 31.053  0.514 1304.600 0.085 1.000 1.014
## 6      1p: U  23 32 33.028  0.417 1306.574 0.085 0.982 0.985
## 7      1p: BR  23 32 32.655  0.435 1306.201 0.087 0.988 0.990
## 8      0p   22 33 38.901  0.221 1310.447 0.117 0.894 0.913
```

```
#print(nice_table(results_SEM[,c("Model", "npar", "df", "chisq", "pvalue", "aic", "srmr", "cfi", "ifi")])
```

8.3.2 Model comparisson

```
anova(fit_sem1, fit_sem2) #not significant
```

```
##
## Chi-Squared Difference Test
##
##      Df      AIC      BIC  Chisq Chisq diff  RMSEA Df diff Pr(>Chisq)
## fit_sem1 30 1308.2 1354.5 30.673
## fit_sem2 31 1307.6 1352.0 32.021      1.3475 0.085991      1      0.2457
```

```
anova(fit_sem1, fit_sem3) #not significant
```

```
##
## Chi-Squared Difference Test
##
##      Df      AIC      BIC  Chisq Chisq diff RMSEA Df diff Pr(>Chisq)
## fit_sem1 30 1308.2 1354.5 30.673
## fit_sem3 31 1306.2 1350.6 30.674 0.00075944      0      1      0.978
```

```
anova(fit_sem1, fit_sem4) #not significant
```

```
##
## Chi-Squared Difference Test
##
##      Df      AIC      BIC  Chisq Chisq diff RMSEA Df diff Pr(>Chisq)
## fit_sem1 30 1308.2 1354.5 30.673
## fit_sem4 31 1306.6 1351.0 31.052      0.37839      0      1      0.5385
```

#So: models 2-4 are not fitting sign than model 1

```
anova(fit_sem1, fit_sem5) #not significant
```

```
##  
## Chi-Squared Difference Test  
##  
##           Df      AIC      BIC  Chisq Chisq diff RMSEA Df diff Pr(>Chisq)  
## fit_sem1 30 1308.2 1354.5 30.673  
## fit_sem5 32 1304.6 1347.2 31.053    0.37977    0      2    0.8271
```

```
anova(fit_sem1, fit_sem6) #not significant
```

```
##  
## Chi-Squared Difference Test  
##  
##           Df      AIC      BIC  Chisq Chisq diff  RMSEA Df diff Pr(>Chisq)  
## fit_sem1 30 1308.2 1354.5 30.673  
## fit_sem6 32 1306.6 1349.1 33.028    2.3543 0.061397      2    0.3081
```

```
anova(fit_sem1, fit_sem7) #not significant
```

```
##  
## Chi-Squared Difference Test  
##  
##           Df      AIC      BIC  Chisq Chisq diff RMSEA Df diff Pr(>Chisq)  
## fit_sem1 30 1308.2 1354.5 30.673  
## fit_sem7 32 1306.2 1348.8 32.655    1.9813    0      2    0.3713
```

#So: models 5-7 are not fitting sign than model 1

```
anova(fit_sem2, fit_sem6) #not significant
```

```
##  
## Chi-Squared Difference Test  
##  
##           Df      AIC      BIC  Chisq Chisq diff  RMSEA Df diff Pr(>Chisq)  
## fit_sem2 31 1307.6 1352.0 32.021  
## fit_sem6 32 1306.6 1349.1 33.028    1.0068 0.012034      1    0.3157
```

```
anova(fit_sem2, fit_sem7) #not significant
```

```
##  
## Chi-Squared Difference Test  
##  
##           Df      AIC      BIC  Chisq Chisq diff RMSEA Df diff Pr(>Chisq)  
## fit_sem2 31 1307.6 1352.0 32.021  
## fit_sem7 32 1306.2 1348.8 32.655    0.63377    0      1    0.426
```

#So: models 6-7 are not fitting sign than model 2

```
anova(fit_sem3, fit_sem5) #not significant
```

```
##
```

```
## Chi-Squared Difference Test
```

```
##
```

	Df	AIC	BIC	Chisq	Chisq diff	RMSEA	Df diff	Pr(>Chisq)
## fit_sem3	31	1306.2	1350.6	30.674				
## fit_sem5	32	1304.6	1347.2	31.053	0.37901	0	1	0.5381

```
anova(fit_sem3, fit_sem7) #not significant
```

```
##
```

```
## Chi-Squared Difference Test
```

```
##
```

	Df	AIC	BIC	Chisq	Chisq diff	RMSEA	Df diff	Pr(>Chisq)
## fit_sem3	31	1306.2	1350.6	30.674				
## fit_sem7	32	1306.2	1348.8	32.655	1.9805	0.14444	1	0.1593

#So: models 5 & 7 are not fitting sign than model 3

```
anova(fit_sem4, fit_sem5) #not significant
```

```
##
```

```
## Chi-Squared Difference Test
```

```
##
```

	Df	AIC	BIC	Chisq	Chisq diff	RMSEA	Df diff	Pr(>Chisq)
## fit_sem4	31	1306.6	1351.0	31.052				
## fit_sem5	32	1304.6	1347.2	31.053	0.0013797	0	1	0.9704

```
anova(fit_sem4, fit_sem6) #not significant
```

```
##
```

```
## Chi-Squared Difference Test
```

```
##
```

	Df	AIC	BIC	Chisq	Chisq diff	RMSEA	Df diff	Pr(>Chisq)
## fit_sem4	31	1306.6	1351.0	31.052				
## fit_sem6	32	1306.6	1349.1	33.028	1.976	0.1441	1	0.1598

#So: models 5-6 are not fitting sign than model 4

```
anova(fit_sem5, fit_sem8) #significant
```

```
##
```

```
## Chi-Squared Difference Test
```

```
##
```

	Df	AIC	BIC	Chisq	Chisq diff	RMSEA	Df diff	Pr(>Chisq)
## fit_sem5	32	1304.6	1347.2	31.053				
## fit_sem8	33	1310.5	1351.2	38.901	7.8476	0.3817	1	0.005089 **
## ---								

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

```
anova(fit_sem6, fit_sem8) #significant
```

```
##
## Chi-Squared Difference Test
##
##           Df      AIC      BIC  Chisq Chisq diff RMSEA Df diff Pr(>Chisq)
## fit_sem6 32 1306.6 1349.1 33.028
## fit_sem8 33 1310.5 1351.2 38.901      5.873 0.322      1 0.01537 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
anova(fit_sem7, fit_sem8) #significant
```

```
##
## Chi-Squared Difference Test
##
##           Df      AIC      BIC  Chisq Chisq diff  RMSEA Df diff Pr(>Chisq)
## fit_sem7 32 1306.2 1348.8 32.655
## fit_sem8 33 1310.5 1351.2 38.901      6.2461 0.33409      1 0.01245 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

#So, model 8 fits significantly worse than models 5-7

#Out of models 5-7, model 5 is in favour based on the fit indices

8.3.3 Favoured model

```
stcoefs_fav_SEM <- standardizedSolution(fit_sem5)
nice_table(stcoefs_fav_SEM)
```

```
## Warning: fonts used in 'flextable' are ignored because the 'pdflatex' engine is
## used and not 'xelatex' or 'lualatex'. You can avoid this warning by using the
## 'set_flextable_defaults(fonts_ignore=TRUE)' command or use a compatible engine
## by defining 'latex_engine: xelatex' in the YAML header of the R Markdown
## document.
```

lhs	op	rhs	est.std	se	z	p	ci.lower	ci.upper
f1_P	=~	EKM60t	0.91	0.09	10.11	< .001***	0.74	1.09
f1_P	=~	ERTt	0.68	0.10	6.52	< .001***	0.47	0.88
f1_P	=~	TASf1	0.27	0.15	1.80	.071	-0.02	0.55
f2_U	=~	SET_Gt	0.25	0.16	1.51	.131	-0.07	0.56
f2_U	=~	FP_Gt	0.64	0.13	5.07	< .001***	0.39	0.88
f2_U	=~	CART_Gt	0.67	0.12	5.42	< .001***	0.43	0.92

lhs	op	rhs	est.std	se	z	p	ci.lower	ci.upper
f3_BR	=~	RSMSf1						
f3_BR	=~	SNQt						
f3_BR	=~	MBIt						
TQt	~	f1_P	0.42	0.13	3.17	.002**	0.16	0.68
TQt	~	f2_U	0.00	0.00			0.00	0.00
TQt	~	f3_BR						
f1_P	~~	f2_U	0.72	0.15	4.85	< .001***	0.43	1.01
f1_P	~~	f3_BR	0.53	0.45	1.17	.242	-0.35	1.41
f2_U	~~	f3_BR	0.77	0.62	1.24	.214	-0.44	1.98
EKM60t	~~	EKM60t	0.16	0.17	0.99	.321	-0.16	0.49
ERTt	~~	ERTt	0.54	0.14	3.86	< .001***	0.27	0.82
TASf1	~~	TASf1	0.93	0.08	11.93	< .001***	0.78	1.08
SET_Gt	~~	SET_Gt	0.94	0.08	11.81	< .001***	0.78	1.10
FP_Gt	~~	FP_Gt	0.59	0.16	3.71	< .001***	0.28	0.91
CART_Gt	~~	CART_Gt	0.55	0.17	3.27	.001**	0.22	0.87
RSMSf1	~~	RSMSf1	1.33	0.52	2.55	.011*	0.31	2.35
SNQt	~~	SNQt	1.04	0.06	18.71	< .001***	0.93	1.15
MBIt	~~	MBIt	1.02	0.04	26.18	< .001***	0.95	1.10
TQt	~~	TQt	0.82	0.11	7.32	< .001***	0.60	1.04
f1_P	~~	f1_P	1.00	0.00			1.00	1.00
f2_U	~~	f2_U	1.00	0.00			1.00	1.00
f3_BR	~~	f3_BR						

```
#print(nice_table(stcoefs_fav_SEM), preview = "docx") #Word format output
semPaths(fit_sem5,
  whatLabels = "std", # Display standardized estimates
  layout = "tree2", # Choose a layout, "tree" is one option
  style = "lisrel", # Style for the diagram
  nCharNodes = 0, # Full variable names
  sizeMan = 7, # Size of manifest variables
  sizeLat = 10, # Size of latent variables
  edge.label.cex = 0.75, # Adjust the size of the edge labels
  label.cex = 0.8, # Adjust the size of the node labels
  residuals = TRUE, # Show residuals
  intercepts = FALSE, # Hide intercepts
  thresholds = FALSE,
  fade = FALSE,
  edge.color = "black"
)
```

