Analysis

Stefan P. Thoma

3/10/2020

Setup

Install / load packages needed:

Load Data

```
data <- read_csv("data/cleanData.csv")
```

```
## Rows: 284 Columns: 131
```

```
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr    (9): id, sex, condition, Frage1, Frage2, Frage3, Leiter, Anmerkungen,...
## dbl  (121): time, iat, ccs1, ccs2, ccs3, ccs4, ccs5, ccs6, ccs7, ccs8, ccs9,...
## dttm   (1): StartDate
```

```
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
data <- data %>% dplyr::select(
  id, time, iat, ccs, nr, nep, ipq, sod, ses, age, edu, sex, pol, vr_exp, vr_eval1, vr_eval2, vr_eval3,
  vr_eval4, vr_eval5, span, seen, condition, starts_with("Frage"), hr_mean, Leiter, Anmerkungen, Zeit
)

head(data)
```

```
## # A tibble: 6 x 29
##   id     time    iat   ccs    nr   nep   ipq   sod   ses   age   edu sex     pol
##   <chr> <dbl>  <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <chr> <dbl>
## 1 2828~     1  0.179  1.08  3.48  4.07    NA    NA  1.76    54     5 W         3
## 2 2828~     2  0.409  1     3.10  4.2     NA    NA  1.71    54     5 W         3
## 3 7799~     1 -0.496  1.17  4.14  3.93    NA    NA  1.53    21     4 M         2
## 4 7799~     2 -0.362  1.08  4.38  4.27    NA    NA  1.35    21     4 M         2
## 5 4379~     1  0.517  1.33  3.43  3.73    NA    NA  1.59    25     5 W         2
## 6 4379~     2  0.634  1.33  3.38  3.93    NA    NA  1.53    25     5 W         2
## # ... with 16 more variables: vr_exp <dbl>, vr_eval1 <dbl>, vr_eval2 <dbl>,
## #   vr_eval3 <dbl>, vr_eval4 <dbl>, vr_eval5 <dbl>, span <dbl>, seen <dbl>,
## #   condition <chr>, Frage1 <chr>, Frage2 <chr>, Frage3 <chr>, hr_mean <dbl>,
## #   Leiter <chr>, Anmerkungen <chr>, Zeit <chr>
```

```
# factor for vr or not
data <- data %>% group_by(id) %>%
  mutate(
```

```
    vr = ifelse(condition %in% c("a", "b", "c"), TRUE, FALSE)
  )
```

Keep in mind the conditions coding:

a == abstract

b == realistic

c == realistic but badly so

Check for multivariate outlier

```
# mvoutlier::chisq.plot(data[,c(3:6)])
# [1] 124 162 161  29  54

# removing three most extreme cases
rmId <- data$id[c(124, 161, 162)]

data <- data %>% filter(!id %in% rmId)
```

```
data$id[c(142,144,275)]
```

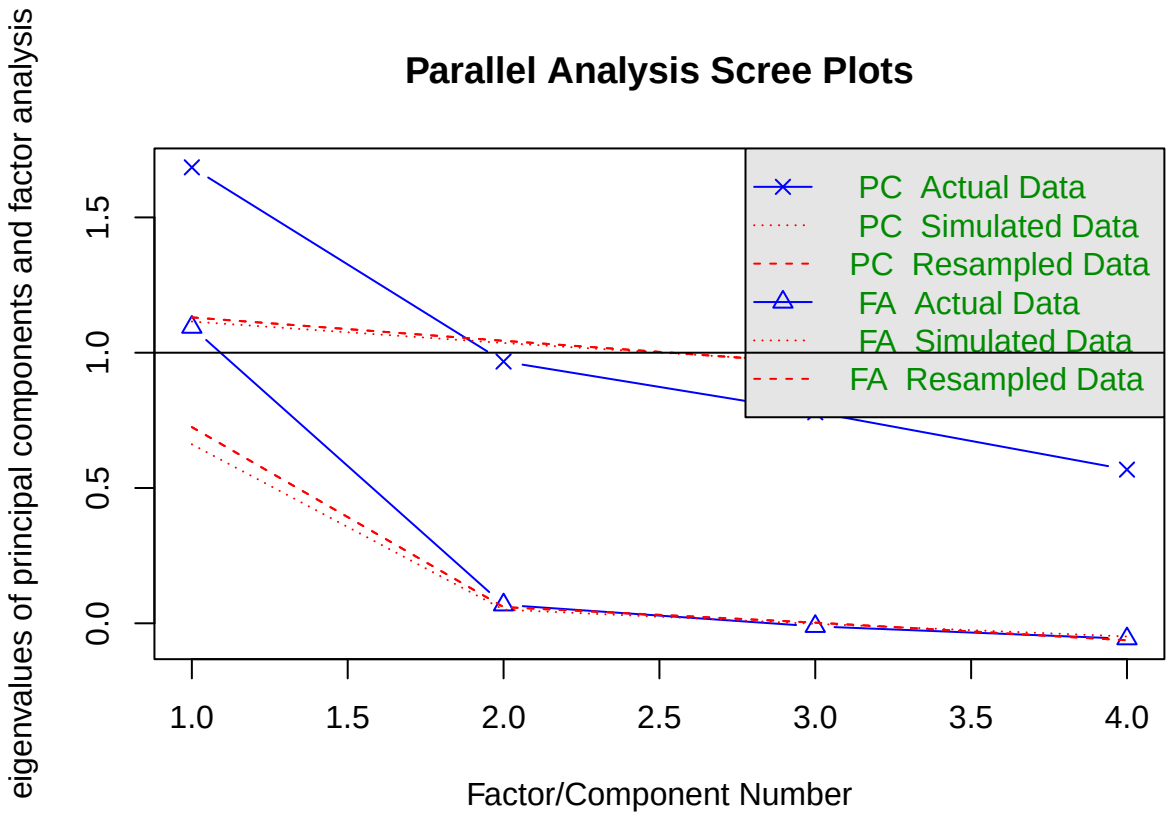
```
## [1] "90811768" "78489627" "04901439"
```

Analysis

We try to find an acceptable model for each DV.

First, I would like to calculate a principal component of all dependent variables (dvs)

```
df_env <- data[c("iat", "ccs", "nr", "nep")]
psych::fa.parallel(df_env)
```




```
## Parallel analysis suggests that the number of factors =  1  and the number of components =  1
```

```
prc_env <- princomp(df_env, cor = TRUE)
# Seems like one factor might just be enough. However, this may be more revealing when done
# on raw data
summary(prc_env)
```

```
## Importance of components:
##                           Comp.1    Comp.2    Comp.3    Comp.4
## Standard deviation     1.2980576 0.9835519 0.8829992 0.7536475
## Proportion of Variance 0.4212384 0.2418436 0.1949219 0.1419962
## Cumulative Proportion  0.4212384 0.6630819 0.8580038 1.0000000
```

```
data$env_pc <- prc_env$scores[,1]
```

Unidimensionality could be assumed. The scores of the first principal component were stored in `data$env_pc`. This vector can now be used as a dependent variable in further exploratory analyses.

This plot is not very useful I guess. Too crowded.

Univariate HLM

So we will create two models for each dependent variables:

First, the model will have the formula:

```
dv ~ condition * time + (time | id)
```

This will be simplified to the following model if model fit is singular:

```
dv ~ condition * time + (1 | id)
```

This will estimate a random intercept for each participant. This model will only take as input the vr conditions (`a`, `b` & `c`), or: `vr == TRUE`.

The second model will have the formula:

```
dv ~ vr * time + (time | condition) + (1 | id)
```

Where a random slope for time is estimated per condition. Further, there is a random intercept per condition, and per id.

Should model fit be singular, we would simplify the model to:

```
dv ~ vr * time + (1 | condition) + (1 | id)
```

If still singular, we would simplify to:

```
dv ~ vr * time + (1 | id)
```

Helping function

```
fit.lme <- function(form, dat){
  lme4::lmer(formula = form, data = dat)
}
```

```
fit_models <- function(dv, dat){
# this function returns a function which fits a model based on a formula minus the predictors.
# This function can be used in the next function which implements the conditions for reducing model com

  function(predictors){
    form <- formula(paste(dv, predictors, sep = " ~ "))
    print(form)
    fit <- fit.lme(form = form, dat = dat)
  }
}
```

```
predictors.vr <-  c("condition * time + (time | id)", "condition * time + (1 | id)")
predictors.all <- c("vr * time + (time | condition) + (1 | id)", "vr * time + (1 | condition) + (1 | id

# function to fit various models based on different inputs of predictors
fit_many <- function(pred.vector, dat, dv){
  fit_model <- fit_models(dv, dat)

  sing <- TRUE
  i <- 1
  while((sing) & i<=length(pred.vector)){
    model <- try(fit_model(pred.vector[i]))

    if(class(model)!="try-error"){
      sing <- isSingular(model)
    }

    i <- i + 1
  }
  print(paste("is model singular: ", sing))
  model
}
```

Vector containing name of all dv's

```r
dvs <-  c("iat", "ccs", "nr", "nep", "env_pc")
```

vr data

```r
# split data frame:
data.vr <- data %>% filter(!vr)
```

```r
vr.models <- lapply(dvs, FUN = function(dv) fit_many(pred.vector = predictors.vr, dat = data.vr, dv = dv
```

```
## iat ~ condition * time + (time | id)
## <environment: 0x7fbc691aac50>
## Error : number of observations (=146) <= number of random effects (=146) for term (time | id); the ra
## iat ~ condition * time + (1 | id)
## <environment: 0x7fbc6958ac50>
## [1] "is model singular:  FALSE"
## ccs ~ condition * time + (time | id)
## <environment: 0x7fbc6a0ae000>
## Error : number of observations (=146) <= number of random effects (=146) for term (time | id); the ra
## ccs ~ condition * time + (1 | id)
## <environment: 0x7fbc6a30b0e8>
## [1] "is model singular:  FALSE"
## nr ~ condition * time + (time | id)
## <environment: 0x7fbc9e6c1a70>
## Error : number of observations (=146) <= number of random effects (=146) for term (time | id); the ra
## nr ~ condition * time + (1 | id)
## <environment: 0x7fbc6a92b040>
## [1] "is model singular:  FALSE"
## nep ~ condition * time + (time | id)
## <environment: 0x7fbc6b162a20>
## Error : number of observations (=146) <= number of random effects (=146) for term (time | id); the ra
## nep ~ condition * time + (1 | id)
## <environment: 0x7fbc6b3c9e78>
## [1] "is model singular:  FALSE"
## env_pc ~ condition * time + (time | id)
## <environment: 0x7fbc6ae5bcf0>
## Error : number of observations (=146) <= number of random effects (=146) for term (time | id); the ra
## env_pc ~ condition * time + (1 | id)
## <environment: 0x7fbc6b8c2748>
## [1] "is model singular:  FALSE"
```

```r
all.models  <-  lapply(dvs, FUN = function(dv) fit_many(pred.vector = predictors.all, dat = data, dv = d
```

```
## iat ~ vr * time + (time | condition) + (1 | id)
## <environment: 0x7fbc6bedf158>
```

```
## boundary (singular) fit: see ?isSingular
```

```
## iat ~ vr * time + (1 | condition) + (1 | id)
## <environment: 0x7fbc9d92b7b0>
## [1] "is model singular:  FALSE"
## ccs ~ vr * time + (time | condition) + (1 | id)
## <environment: 0x7fbc9c0df008>
```

```
## boundary (singular) fit: see ?isSingular
```

```
## ccs ~ vr * time + (1 | condition) + (1 | id)
```

```
## <environment: 0x7fbcae548678>

## boundary (singular) fit: see ?isSingular

## ccs ~ vr * time + (1 | id)
## <environment: 0x7fbcab77a638>
## [1] "is model singular:  FALSE"
## nr ~ vr * time + (time | condition) + (1 | id)
## <environment: 0x7fbc9f1b0a80>

## boundary (singular) fit: see ?isSingular

## nr ~ vr * time + (1 | condition) + (1 | id)
## <environment: 0x7fbc68fb7580>
## [1] "is model singular:  FALSE"
## nep ~ vr * time + (time | condition) + (1 | id)
## <environment: 0x7fbc699c6070>

## boundary (singular) fit: see ?isSingular

## nep ~ vr * time + (1 | condition) + (1 | id)
## <environment: 0x7fbc9eca25b8>

## boundary (singular) fit: see ?isSingular

## nep ~ vr * time + (1 | id)
## <environment: 0x7fbcba52ad00>
## [1] "is model singular:  FALSE"
## env_pc ~ vr * time + (time | condition) + (1 | id)
## <environment: 0x7fbc6aa22da0>

## boundary (singular) fit: see ?isSingular

## env_pc ~ vr * time + (1 | condition) + (1 | id)
## <environment: 0x7fbc6b4a3708>
## [1] "is model singular:  FALSE"
```

look at model summaries

vr

```
lapply(vr.models, FUN = summary)
```

```
## [[1]]
## Linear mixed model fit by REML ['lmerMod']
## Formula: iat ~ condition * time + (1 | id)
##    Data: dat
##
## REML criterion at convergence: 146.3
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -1.98432 -0.60501  0.02046  0.55242  1.80005
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  id       (Intercept) 0.08092  0.2845
##  Residual             0.08530  0.2921
## Number of obs: 146, groups:  id, 73
```

```
##
## Fixed effects:
##                           Estimate Std. Error t value
## (Intercept)               0.469249   0.142465   3.294
## conditiontext.bild       -0.220540   0.203564  -1.083
## conditionvideo           -0.043282   0.203564  -0.213
## time                     -0.114739   0.082606  -1.389
## conditiontext.bild:time   0.098658   0.118034   0.836
## conditionvideo:time      -0.004853   0.118034  -0.041
##
## Correlation of Fixed Effects:
##             (Intr) cndtn. cndtnv time   cndt.:
## cndtntxt.bl -0.700
## conditionvd -0.700  0.490
## time        -0.870  0.609  0.609
## cndtntxt.b:  0.609 -0.870 -0.426 -0.700
## condtnvd:tm  0.609 -0.426 -0.870 -0.700  0.490
##
## [[2]]
## Linear mixed model fit by REML ['lmerMod']
## Formula: ccs ~ condition * time + (1 | id)
##    Data: dat
##
## REML criterion at convergence: 156.3
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.4729 -0.3998 -0.1459  0.2664  4.4433
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  id       (Intercept) 0.15794  0.3974
##  Residual             0.06402  0.2530
## Number of obs: 146, groups:  id, 73
##
## Fixed effects:
##                           Estimate Std. Error t value
## (Intercept)               1.52667    0.13828  11.040
## conditiontext.bild       -0.08222    0.19759  -0.416
## conditionvideo           -0.04750    0.19759  -0.240
## time                     -0.05333    0.07157  -0.745
## conditiontext.bild:time   0.02903    0.10226   0.284
## conditionvideo:time       0.05333    0.10226   0.522
##
## Correlation of Fixed Effects:
##             (Intr) cndtn. cndtnv time   cndt.:
## cndtntxt.bl -0.700
## conditionvd -0.700  0.490
## time        -0.776  0.543  0.543
## cndtntxt.b:  0.543 -0.776 -0.380 -0.700
## condtnvd:tm  0.543 -0.380 -0.776 -0.700  0.490
##
## [[3]]
## Linear mixed model fit by REML ['lmerMod']
```

```
## Formula: nr ~ condition * time + (1 | id)
##    Data: dat
##
## REML criterion at convergence: 161.6
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.61940 -0.36602  0.06386  0.45047  1.98254
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  id       (Intercept) 0.31246  0.5590
##  Residual             0.03951  0.1988
## Number of obs: 146, groups:  id, 73
##
## Fixed effects:
##                            Estimate Std. Error t value
## (Intercept)                3.792381   0.142827  26.552
## conditiontext.bild         0.066746   0.204081   0.327
## conditionvideo             0.042937   0.204081   0.210
## time                      -0.026667   0.056218  -0.474
## conditiontext.bild:time -0.007063   0.080328  -0.088
## conditionvideo:time        0.084206   0.080328   1.048
##
## Correlation of Fixed Effects:
##             (Intr) cndtn. cndtnv time   cndt.:
## cndtntxt.bl -0.700
## conditionvd -0.700  0.490
## time        -0.590  0.413  0.413
## cndtntxt.b:  0.413 -0.590 -0.289 -0.700
## condtnvd:tm  0.413 -0.289 -0.590 -0.700  0.490
##
## [[4]]
## Linear mixed model fit by REML ['lmerMod']
## Formula: nep ~ condition * time + (1 | id)
##    Data: dat
##
## REML criterion at convergence: 108.8
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -1.87427 -0.48795  0.04744  0.46649  1.99779
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  id       (Intercept) 0.15379  0.3922
##  Residual             0.03594  0.1896
## Number of obs: 146, groups:  id, 73
##
## Fixed effects:
##                           Estimate Std. Error t value
## (Intercept)                3.84800    0.11550  33.317
## conditiontext.bild        -0.08689    0.16503  -0.527
## conditionvideo            -0.08689    0.16503  -0.527
```

```
## time                    -0.02133    0.05362  -0.398
## conditiontext.bild:time  0.13244    0.07662   1.729
## conditionvideo:time      0.09911    0.07662   1.294
##
## Correlation of Fixed Effects:
##             (Intr) cndtn. cndtnv time   cndt.:
## cndtntxt.bl -0.700
## conditionvd -0.700  0.490
## time        -0.696  0.487  0.487
## cndtntxt.b:  0.487 -0.696 -0.341 -0.700
## condtnvd:tm  0.487 -0.341 -0.696 -0.700  0.490
##
## [[5]]
## Linear mixed model fit by REML ['lmerMod']
## Formula: env_pc ~ condition * time + (1 | id)
##    Data: dat
##
## REML criterion at convergence: 398.2
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.38447 -0.38996  0.01163  0.46369  1.88977
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  id       (Intercept) 1.6726   1.2933
##  Residual             0.2164   0.4652
## Number of obs: 146, groups:  id, 73
##
## Fixed effects:
##                          Estimate Std. Error t value
## (Intercept)              -0.04105    0.33193  -0.124
## conditiontext.bild       -0.10604    0.47428  -0.224
## conditionvideo           -0.05936    0.47428  -0.125
## time                     -0.07178    0.13157  -0.546
## conditiontext.bild:time   0.21611    0.18799   1.150
## conditionvideo:time       0.17183    0.18799   0.914
##
## Correlation of Fixed Effects:
##             (Intr) cndtn. cndtnv time   cndt.:
## cndtntxt.bl -0.700
## conditionvd -0.700  0.490
## time        -0.595  0.416  0.416
## cndtntxt.b:  0.416 -0.595 -0.291 -0.700
## condtnvd:tm  0.416 -0.291 -0.595 -0.700  0.490
```

all

```
lapply(all.models, FUN = summary)
```

```
## [[1]]
## Linear mixed model fit by REML ['lmerMod']
## Formula: iat ~ vr * time + (1 | condition) + (1 | id)
##    Data: dat
```

```
##
## REML criterion at convergence: 311.1
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.91016 -0.59989  0.01681  0.57888  2.05187
##
## Random effects:
##  Groups    Name        Variance Std.Dev.
##  id        (Intercept) 0.097343 0.31200
##  condition (Intercept) 0.009798 0.09899
##  Residual              0.094308 0.30710
## Number of obs: 284, groups:  id, 142; condition, 6
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)  0.38216    0.10516   3.634
## vrTRUE      -0.30385    0.15023  -2.023
## time        -0.08390    0.05083  -1.651
## vrTRUE:time  0.14922    0.07292   2.046
##
## Correlation of Fixed Effects:
##             (Intr) vrTRUE time
## vrTRUE      -0.700
## time        -0.725  0.508
## vrTRUE:time  0.505 -0.728 -0.697
##
## [[2]]
## Linear mixed model fit by REML ['lmerMod']
## Formula: ccs ~ vr * time + (1 | id)
##    Data: dat
##
## REML criterion at convergence: 279.3
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.0893 -0.3890 -0.1527  0.2999  4.6990
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  id       (Intercept) 0.18845  0.4341
##  Residual             0.05201  0.2281
## Number of obs: 284, groups:  id, 142
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)  1.48402    0.07838  18.933
## vrTRUE      -0.02387    0.11245  -0.212
## time        -0.02626    0.03775  -0.696
## vrTRUE:time  0.01901    0.05415   0.351
##
## Correlation of Fixed Effects:
##             (Intr) vrTRUE time
## vrTRUE      -0.697
```

```
## time        -0.722  0.504
## vrTRUE:time  0.504 -0.722 -0.697
##
## [[3]]
## Linear mixed model fit by REML ['lmerMod']
## Formula: nr ~ vr * time + (1 | condition) + (1 | id)
##    Data: dat
##
## REML criterion at convergence: 260.8
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.61061 -0.44890  0.02847  0.47704  1.79110
##
## Random effects:
##  Groups    Name        Variance Std.Dev.
##  id        (Intercept) 0.21855  0.4675
##  condition (Intercept) 0.01365  0.1168
##  Residual              0.03999  0.2000
## Number of obs: 284, groups:  id, 142; condition, 6
##
## Fixed effects:
##              Estimate Std. Error t value
## (Intercept)  3.829043   0.101404  37.760
## vrTRUE      -0.022280   0.144559  -0.154
## time        -0.001305   0.033101  -0.039
## vrTRUE:time  0.077909   0.047486   1.641
##
## Correlation of Fixed Effects:
##             (Intr) vrTRUE time
## vrTRUE      -0.701
## time        -0.490  0.343
## vrTRUE:time  0.341 -0.493 -0.697
##
## [[4]]
## Linear mixed model fit by REML ['lmerMod']
## Formula: nep ~ vr * time + (1 | id)
##    Data: dat
##
## REML criterion at convergence: 184.7
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.15318 -0.52795 -0.02328  0.48668  2.18887
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  id       (Intercept) 0.14277  0.3779
##  Residual             0.03538  0.1881
## Number of obs: 284, groups:  id, 142
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)  3.79087    0.06618  57.285
```

```
## vrTRUE      -0.07009    0.09493  -0.738
## time         0.05479    0.03114   1.760
## vrTRUE:time  0.03796    0.04467   0.850
##
## Correlation of Fixed Effects:
##             (Intr) vrTRUE time
## vrTRUE      -0.697
## time        -0.706  0.492
## vrTRUE:time  0.492 -0.706 -0.697
##
## [[5]]
## Linear mixed model fit by REML ['lmerMod']
## Formula: env_pc ~ vr * time + (1 | condition) + (1 | id)
##    Data: dat
##
## REML criterion at convergence: 766.2
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.34047 -0.41346  0.01685  0.45159  1.90714
##
## Random effects:
##  Groups    Name        Variance Std.Dev.
##  id        (Intercept) 1.468131 1.21166
##  condition (Intercept) 0.006021 0.07759
##  Residual              0.227836 0.47732
## Number of obs: 284, groups:  id, 142; condition, 6
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept) -0.09526    0.19423  -0.490
## vrTRUE      -0.28484    0.27842  -1.023
## time         0.05577    0.07901   0.706
## vrTRUE:time  0.20595    0.11334   1.817
##
## Correlation of Fixed Effects:
##             (Intr) vrTRUE time
## vrTRUE      -0.698
## time        -0.610  0.426
## vrTRUE:time  0.425 -0.611 -0.697
```

Model diagnostics

I save model diagnostics as pdfs separately, for visibility reasons.

```
plot_diagn <- function(model){

  filename <- paste( model@call$formula[2], sub("\\ .*", "", model@call$formula[3]), sep = "_")
  pdf(file = paste("analysisOutputs/diagnostics/", filename, ".pdf", sep = ""),   # The directory you w
    #paper = "a3",
    height = 5.9*4,
    width = 4.2*4
    )
```

```
  print(performance::check_model(model)  )

  dev.off()
}
```

```
lapply(vr.models, FUN = plot_diagn)
lapply(all.models, FUN = plot_diagn)
```

I focus model diagnostic on the vr models. They include all data.

IAT

\begin{figure}


Linearity
Reference line should be flat and horizontal
Residuals
Fitted values
Homogeneity of Variance
Reference line should be flat and horizontal
√|Std. residuals|
Fitted values
Collinearity
Higher bars (>5) indicate potential collinearity issues
Variance Inflation Factor (VIF)
time
vr
vr:time
low (< 5)
moderate (< 10)
high (>= 10)
Influential Observations
Points should be inside the contour lines
Std. Residuals
Leverage (hii)
Normality of Residuals
Dots should fall along the line
Sample Quantiles
Standard Normal Distribution Quantiles
Normality of Residuals
Distribution should be close to the normal curve
Density
Residuals
Normality of Random Effects (id)
Dots should be plotted along the line
RE Quantiles
Theoretical Quantiles
Normality of Random Effects (condition)
Dots should be plotted along the line
RE Quantiles
Theoretical Quantiles

{

}
\caption{iat_vr_diagnostics} \end{figure}