**Stepwise mixed effects modeling of the response times.** This is a printout of the code with the output rendered via rmarkdown::render().

**KEY:**

*# This is a comment*.

**this** and this **are commands**

## This is output.

Plot output is unmarked.

# MULTIVEL LME MODELING:  
  
install.packages("car")

## --- Please select a CRAN mirror for use in this session ---  
## package 'car' successfully unpacked and MD5 sums checked  
##   
## The downloaded binary packages are in  
## C:\Users\Pablo\AppData\Local\Temp\RtmpCw18BB\downloaded\_packages

install.packages("pastecs")

## package 'pastecs' successfully unpacked and MD5 sums checked  
##   
## The downloaded binary packages are in  
## C:\Users\Pablo\AppData\Local\Temp\RtmpCw18BB\downloaded\_packages

install.packages('gmodels')

## package 'gmodels' successfully unpacked and MD5 sums checked  
##   
## The downloaded binary packages are in  
## C:\Users\Pablo\AppData\Local\Temp\RtmpCw18BB\downloaded\_packages

install.packages("compute.es")

## package 'compute.es' successfully unpacked and MD5 sums checked  
##   
## The downloaded binary packages are in  
## C:\Users\Pablo\AppData\Local\Temp\RtmpCw18BB\downloaded\_packages

install.packages('trimr')

## package 'trimr' successfully unpacked and MD5 sums checked  
##   
## The downloaded binary packages are in  
## C:\Users\Pablo\AppData\Local\Temp\RtmpCw18BB\downloaded\_packages

install.packages('lme4')

## also installing the dependency 'lattice'

## package 'lattice' successfully unpacked and MD5 sums checked  
## package 'lme4' successfully unpacked and MD5 sums checked  
##   
## The downloaded binary packages are in  
## C:\Users\Pablo\AppData\Local\Temp\RtmpCw18BB\downloaded\_packages

install.packages('ltm')

## package 'ltm' successfully unpacked and MD5 sums checked  
##   
## The downloaded binary packages are in  
## C:\Users\Pablo\AppData\Local\Temp\RtmpCw18BB\downloaded\_packages

install.packages('psych')

## package 'psych' successfully unpacked and MD5 sums checked  
##   
## The downloaded binary packages are in  
## C:\Users\Pablo\AppData\Local\Temp\RtmpCw18BB\downloaded\_packages

install.packages('dae')

## package 'dae' successfully unpacked and MD5 sums checked  
##   
## The downloaded binary packages are in  
## C:\Users\Pablo\AppData\Local\Temp\RtmpCw18BB\downloaded\_packages

install.packages('SuppDists')

## package 'SuppDists' successfully unpacked and MD5 sums checked  
##   
## The downloaded binary packages are in  
## C:\Users\Pablo\AppData\Local\Temp\RtmpCw18BB\downloaded\_packages

install.packages('effects')

## package 'effects' successfully unpacked and MD5 sums checked  
##   
## The downloaded binary packages are in  
## C:\Users\Pablo\AppData\Local\Temp\RtmpCw18BB\downloaded\_packages

install.packages('plyr')

## package 'plyr' successfully unpacked and MD5 sums checked  
##   
## The downloaded binary packages are in  
## C:\Users\Pablo\AppData\Local\Temp\RtmpCw18BB\downloaded\_packages

install.packages('multcomp')

## package 'multcomp' successfully unpacked and MD5 sums checked  
##   
## The downloaded binary packages are in  
## C:\Users\Pablo\AppData\Local\Temp\RtmpCw18BB\downloaded\_packages

install.packages('lattice')

## package 'lattice' successfully unpacked and MD5 sums checked  
##   
## The downloaded binary packages are in  
## C:\Users\Pablo\AppData\Local\Temp\RtmpCw18BB\downloaded\_packages

install.packages('ggplot2')

## package 'ggplot2' successfully unpacked and MD5 sums checked  
##   
## The downloaded binary packages are in  
## C:\Users\Pablo\AppData\Local\Temp\RtmpCw18BB\downloaded\_packages

install.packages('doBy')

## package 'doBy' successfully unpacked and MD5 sums checked  
##   
## The downloaded binary packages are in  
## C:\Users\Pablo\AppData\Local\Temp\RtmpCw18BB\downloaded\_packages

install.packages('eeptools')

## package 'eeptools' successfully unpacked and MD5 sums checked  
##   
## The downloaded binary packages are in  
## C:\Users\Pablo\AppData\Local\Temp\RtmpCw18BB\downloaded\_packages

install.packages('sjPlot')

## package 'sjPlot' successfully unpacked and MD5 sums checked  
##   
## The downloaded binary packages are in  
## C:\Users\Pablo\AppData\Local\Temp\RtmpCw18BB\downloaded\_packages

install.packages("stargazer")

## package 'stargazer' successfully unpacked and MD5 sums checked  
##   
## The downloaded binary packages are in  
## C:\Users\Pablo\AppData\Local\Temp\RtmpCw18BB\downloaded\_packages

install.packages("Hmisc")

## package 'Hmisc' successfully unpacked and MD5 sums checked  
##   
## The downloaded binary packages are in  
## C:\Users\Pablo\AppData\Local\Temp\RtmpCw18BB\downloaded\_packages

install.packages("merTools")

## package 'merTools' successfully unpacked and MD5 sums checked  
##   
## The downloaded binary packages are in  
## C:\Users\Pablo\AppData\Local\Temp\RtmpCw18BB\downloaded\_packages

install.packages("MuMIn")

## package 'MuMIn' successfully unpacked and MD5 sums checked  
##   
## The downloaded binary packages are in  
## C:\Users\Pablo\AppData\Local\Temp\RtmpCw18BB\downloaded\_packages

library(ltm)

## Loading required package: MASS

## Loading required package: msm

## Loading required package: polycor

library(lattice)  
library(psych)

##   
## Attaching package: 'psych'

## The following object is masked from 'package:ltm':  
##   
## factor.scores

## The following object is masked from 'package:polycor':  
##   
## polyserial

library(gmodels)  
library(pastecs)

## Loading required package: boot

##   
## Attaching package: 'boot'

## The following object is masked from 'package:psych':  
##   
## logit

## The following object is masked from 'package:lattice':  
##   
## melanoma

## The following object is masked from 'package:msm':  
##   
## cav

library(plyr)  
library(SuppDists)  
library(car)

##   
## Attaching package: 'car'

## The following object is masked from 'package:boot':  
##   
## logit

## The following object is masked from 'package:psych':  
##   
## logit

library(compute.es)  
library(effects)

##   
## Attaching package: 'effects'

## The following object is masked from 'package:car':  
##   
## Prestige

library(multcomp)

## Loading required package: mvtnorm

## Loading required package: survival

##   
## Attaching package: 'survival'

## The following object is masked from 'package:boot':  
##   
## aml

## Loading required package: TH.data

##   
## Attaching package: 'TH.data'

## The following object is masked from 'package:MASS':  
##   
## geyser

library(Hmisc)

## Loading required package: Formula

## Loading required package: ggplot2

## Need help? Try the ggplot2 mailing list:  
## http://groups.google.com/group/ggplot2.

##   
## Attaching package: 'ggplot2'

## The following objects are masked from 'package:psych':  
##   
## %+%, alpha

##   
## Attaching package: 'Hmisc'

## The following objects are masked from 'package:plyr':  
##   
## is.discrete, summarize

## The following object is masked from 'package:psych':  
##   
## describe

## The following objects are masked from 'package:base':  
##   
## format.pval, round.POSIXt, trunc.POSIXt, units

library(ggplot2)  
library(dae)

##   
## Attaching package: 'dae'

## The following object is masked from 'package:mvtnorm':  
##   
## rmvnorm

## The following object is masked from 'package:psych':  
##   
## harmonic.mean

library(trimr)  
library(lme4)

## Loading required package: Matrix

library(doBy)  
library(eeptools)

## Welcome to eeptools for R version 0.9.1!

## Developed by Jared E. Knowles 2012-2015

## for the Wisconsin Department of Public Instruction

## Distributed without warranty.

library(sjPlot)  
library(merTools)

## Loading required package: arm

##   
## arm (Version 1.9-3, built: 2016-11-21)

## Working directory is C:/Users/Pablo/Dropbox/STUDIES/R/Experiment Data/Modality-switching experiment/RTs and accuracy

##   
## Attaching package: 'arm'

## The following object is masked from 'package:car':  
##   
## logit

## The following object is masked from 'package:boot':  
##   
## logit

## The following objects are masked from 'package:psych':  
##   
## logit, rescale, sim

## Loading required package: dplyr

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:Hmisc':  
##   
## combine, src, summarize

## The following object is masked from 'package:car':  
##   
## recode

## The following objects are masked from 'package:plyr':  
##   
## arrange, count, desc, failwith, id, mutate, rename, summarise,  
## summarize

## The following objects are masked from 'package:pastecs':  
##   
## first, last

## The following object is masked from 'package:MASS':  
##   
## select

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

##   
## Attaching package: 'merTools'

## The following object is masked from 'package:psych':  
##   
## ICC

library(MuMIn)  
library(stargazer)

##   
## Please cite as:

## Hlavac, Marek (2015). stargazer: Well-Formatted Regression and Summary Statistics Tables.

## R package version 5.2. http://CRAN.R-project.org/package=stargazer

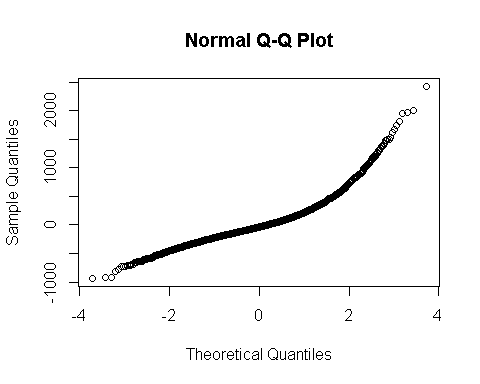
targetbeh\_responded\_OUT = readRDS('targetbeh\_responded\_OUT.rds')  
  
  
# Preparations:  
# Mean-center and scale, as built-in in R  
targetbeh\_responded\_OUT$s\_conc\_letters <- scale(targetbeh\_responded\_OUT$conc\_letters)  
targetbeh\_responded\_OUT$s\_prop\_letters <- scale(targetbeh\_responded\_OUT$prop\_letters)  
targetbeh\_responded\_OUT$s\_LSA\_distance <- scale(targetbeh\_responded\_OUT$LSA\_distance)  
targetbeh\_responded\_OUT$s\_prop\_lg10CD <- scale(targetbeh\_responded\_OUT$prop\_lg10CD)  
targetbeh\_responded\_OUT$s\_conc\_lg10CD <- scale(targetbeh\_responded\_OUT$conc\_lg10CD)  
targetbeh\_responded\_OUT$s\_conc\_exc <- scale(targetbeh\_responded\_OUT$conc\_exc)  
targetbeh\_responded\_OUT$s\_prop\_exc <- scale(targetbeh\_responded\_OUT$prop\_exc)  
targetbeh\_responded\_OUT$s\_trial <- scale(targetbeh\_responded\_OUT$trial)  
targetbeh\_responded\_OUT$s\_prop\_orthneigh <- scale(targetbeh\_responded\_OUT$prop\_orthneigh)  
targetbeh\_responded\_OUT$s\_conc\_orthneigh <- scale(targetbeh\_responded\_OUT$conc\_orthneigh)  
targetbeh\_responded\_OUT$s\_Age\_months <- scale(targetbeh\_responded\_OUT$Age\_months)  
str(targetbeh\_responded\_OUT)

## 'data.frame': 5000 obs. of 69 variables:  
## $ Ptp : Factor w/ 47 levels "1","2","3","4",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ Group : Factor w/ 3 levels "","Quick","Selfpaced": 2 2 2 2 2 2 2 2 2 2 ...  
## $ Gender : Factor w/ 2 levels "F","M": 1 1 1 1 1 1 1 1 1 1 ...  
## $ Lefthanded : Factor w/ 2 levels "N","Y": 1 1 1 1 1 1 1 1 1 1 ...  
## $ A\_V\_OK : Factor w/ 2 levels "N","Y": 2 2 2 2 2 2 2 2 2 2 ...  
## $ Event : int 48 18 5 105 100 20 43 16 51 29 ...  
## $ Trial : Factor w/ 108 levels "2","4","6","8",..: 48 18 5 105 100 20 43 16 51 29 ...  
## $ Item : int 1 2 3 4 5 6 7 8 9 10 ...  
## $ Condition : Factor w/ 3 levels "visual2visual",..: 2 2 2 2 2 2 2 2 2 2 ...  
## $ Accuracy : Factor w/ 3 levels "","rightanswer",..: 2 3 3 3 2 2 3 3 3 2 ...  
## $ RT : int 544 356 516 317 1219 943 870 620 563 576 ...  
## $ Position : Factor w/ 2 levels "prime","target": 2 2 2 2 2 2 2 2 2 2 ...  
## $ cat : Factor w/ 3 levels "auditory","haptic",..: 3 3 3 3 3 3 3 3 3 3 ...  
## $ catspec : Factor w/ 6 levels "a\_de","a\_het",..: 6 5 5 5 5 5 6 5 5 5 ...  
## $ solution : Factor w/ 2 levels "Ja","Nee": 2 1 2 1 1 2 1 1 1 2 ...  
## $ property : Factor w/ 216 levels "Aardse","Absorberende",..: 100 216 4 7 183 75 170 147 163 97 ...  
## $ concept : Factor w/ 107 levels "Aankondiging",..: 51 48 86 48 86 77 49 2 65 107 ...  
## $ LSA\_distance : num 0.902 0.974 NA 0.879 0.928 ...  
## $ mean\_perceptualstrength : num 2.31 2.73 2.89 3.33 3.19 ...  
## $ mean\_concreteness : num 1.93 2.87 1.93 3.33 2.5 ...  
## $ mean\_exc : num 0.131 0.148 0.213 0.273 0.274 ...  
## $ prop\_concreteness : num NA 2.53 NA 3.47 3.07 2.07 2.33 2.8 2.13 2.33 ...  
## $ prop\_exc : num 0.145 0.25 0.108 0.5 0.231 ...  
## $ prop\_lg10CD : num 1.77 1.48 1.57 1.2 2.38 2.16 1.04 1.26 2.57 3.01 ...  
## $ prop\_orthneigh : int 4 2 2 0 1 6 0 0 7 11 ...  
## $ prop\_letters : int 6 6 9 12 7 5 7 12 5 6 ...  
## $ conc\_concreteness : num 1.93 3.2 1.93 3.2 1.93 NA 3.07 3.53 2.93 3.33 ...  
## $ conc\_exc : num 0.1176 0.0455 0.3182 0.0455 0.3182 ...  
## $ conc\_lg10CD : num 1.49 2.19 3.13 2.19 3.13 0.6 2.8 1.76 3.07 2.98 ...  
## $ conc\_letters : int 7 8 10 8 10 12 4 11 9 6 ...  
## $ mean\_lg10CD : num 1.63 1.83 2.35 1.7 2.75 ...  
## $ conc\_orthneigh : int 0 1 3 1 3 0 13 0 0 3 ...  
## $ trial : num 96 36 10 210 200 40 86 32 102 58 ...  
## $ accuracy : num 1 0 0 0 1 1 0 0 0 1 ...  
## $ MRT : num 753 753 753 753 753 ...  
## $ RT.based\_Groups : Factor w/ 2 levels "Quick","Slow": 2 2 2 2 2 2 2 2 2 2 ...  
## $ Age\_months : num 275 275 275 275 275 ...  
## $ ConditionRe : chr " Haptic / Visual" " Haptic / Visual" " Haptic / Visual" " Haptic / Visual" ...  
## $ s\_conc\_letters : num [1:5000, 1] 0.131 0.541 1.36 0.541 1.36 ...  
## ..- attr(\*, "scaled:center")= num 6.68  
## ..- attr(\*, "scaled:scale")= num 2.44  
## $ s\_prop\_letters : num [1:5000, 1] -0.524 -0.524 0.9313 2.3867 -0.0389 ...  
## ..- attr(\*, "scaled:center")= num 7.08  
## ..- attr(\*, "scaled:scale")= num 2.06  
## $ s\_LSA\_distance : num [1:5000, 1] -0.176 0.613 NA -0.42 0.115 ...  
## ..- attr(\*, "scaled:center")= num 0.918  
## ..- attr(\*, "scaled:scale")= num 0.0914  
## $ s\_prop\_lg10CD : num [1:5000, 1] -0.227 -0.648 -0.518 -1.054 0.658 ...  
## ..- attr(\*, "scaled:center")= num 1.93  
## ..- attr(\*, "scaled:scale")= num 0.689  
## $ s\_conc\_lg10CD : num [1:5000, 1] -1.317 -0.403 0.824 -0.403 0.824 ...  
## ..- attr(\*, "scaled:center")= num 2.5  
## ..- attr(\*, "scaled:scale")= num 0.766  
## $ s\_conc\_exc : num [1:5000, 1] -1.2 -1.7 0.17 -1.7 0.17 ...  
## ..- attr(\*, "scaled:center")= num 0.293  
## ..- attr(\*, "scaled:scale")= num 0.146  
## $ s\_prop\_exc : num [1:5000, 1] -1.475 -0.894 -1.68 0.485 -1 ...  
## ..- attr(\*, "scaled:center")= num 0.412  
## ..- attr(\*, "scaled:scale")= num 0.181  
## $ s\_trial : num [1:5000, 1] -0.215 -1.179 -1.596 1.615 1.455 ...  
## ..- attr(\*, "scaled:center")= num 109  
## ..- attr(\*, "scaled:scale")= num 62.3  
## $ s\_prop\_orthneigh : num [1:5000, 1] 0.118 -0.343 -0.343 -0.803 -0.573 ...  
## ..- attr(\*, "scaled:center")= num 3.49  
## ..- attr(\*, "scaled:scale")= num 4.34  
## $ s\_conc\_orthneigh : num [1:5000, 1] -0.775 -0.551 -0.103 -0.551 -0.103 ...  
## ..- attr(\*, "scaled:center")= num 3.46  
## ..- attr(\*, "scaled:scale")= num 4.47  
## $ s\_Age\_months : num [1:5000, 1] 0.227 0.227 0.227 0.227 0.227 ...  
## ..- attr(\*, "scaled:center")= num 267  
## ..- attr(\*, "scaled:scale")= num 34  
## $ targetbeh\_responded\_OUT$RT^2 : num 295936 126736 266256 100489 1485961 ...  
## $ sqrt(targetbeh\_responded\_OUT$RT) : num 23.3 18.9 22.7 17.8 34.9 ...  
## $ log(targetbeh\_responded\_OUT$RT + 1): num 6.3 5.88 6.25 5.76 7.11 ...  
## $ 1/targetbeh\_responded\_OUT$RT : num 0.00184 0.00281 0.00194 0.00315 0.00082 ...  
## $ 1/sqrt(targetbeh\_responded\_OUT$RT) : num 0.0429 0.053 0.044 0.0562 0.0286 ...  
## $ sqRT : num 295936 126736 266256 100489 1485961 ...  
## $ sqrtRT : num 23.3 18.9 22.7 17.8 34.9 ...  
## $ logRT : num 6.3 5.88 6.25 5.76 7.11 ...  
## $ recRT : num 0.00184 0.00281 0.00194 0.00315 0.00082 ...  
## $ recsqrtRT : num 0.0429 0.053 0.044 0.0562 0.0286 ...  
## $ targetbeh\_responded\_OUT$RT^2 : num 295936 126736 266256 100489 1485961 ...  
## $ sqrt(targetbeh\_responded\_OUT$RT) : num 23.3 18.9 22.7 17.8 34.9 ...  
## $ log(targetbeh\_responded\_OUT$RT + 1): num 6.3 5.88 6.25 5.76 7.11 ...  
## $ 1/targetbeh\_responded\_OUT$RT : num 0.00184 0.00281 0.00194 0.00315 0.00082 ...  
## $ 1/sqrt(targetbeh\_responded\_OUT$RT) : num 0.0429 0.053 0.044 0.0562 0.0286 ...  
## $ targetbeh\_responded\_OUT$RT^2 : num 295936 126736 266256 100489 1485961 ...  
## $ sqrt(targetbeh\_responded\_OUT$RT) : num 23.3 18.9 22.7 17.8 34.9 ...  
## $ log(targetbeh\_responded\_OUT$RT + 1): num 6.3 5.88 6.25 5.76 7.11 ...  
## $ 1/targetbeh\_responded\_OUT$RT : num 0.00184 0.00281 0.00194 0.00315 0.00082 ...  
## $ 1/sqrt(targetbeh\_responded\_OUT$RT) : num 0.0429 0.053 0.044 0.0562 0.0286 ...

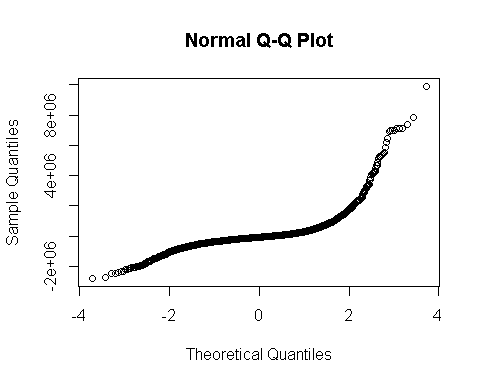
# The standardization above does not consider experimental cells. Below is code   
# that does consider it, with specific standardization for each RT.based\_Groups and each   
# Condition. It is finally not applied, however, because one cannot assume those   
# effects--and indeed, when thus transformed, the variables lose the effects they   
# do present when in the more standard form (regardless of experimental cells).   
# Still, the code is maintained below, should anyone wish to use it.  
  
# GM\_conc\_letters <- aggregate(targetbeh\_responded\_OUT$conc\_letters,   
# list(targetbeh\_responded\_OUT$Condition, targetbeh\_responded\_OUT$RT.based\_Groups),   
# FUN = mean, data=targetbeh\_responded\_OUT)  
# names(GM\_conc\_letters)<- c('Condition', 'RT.based\_Groups','GM\_conc\_letters')  
#   
# GM\_prop\_letters <- aggregate(targetbeh\_responded\_OUT$prop\_letters,   
# list(targetbeh\_responded\_OUT$Condition, targetbeh\_responded\_OUT$RT.based\_Groups),   
# FUN = mean, data=targetbeh\_responded\_OUT)  
# names(GM\_prop\_letters)<- c('Condition', 'RT.based\_Groups','GM\_prop\_letters')  
#   
# GM\_LSA\_distance <- aggregate(targetbeh\_responded\_OUT$LSA\_distance,   
# list(targetbeh\_responded\_OUT$Condition, targetbeh\_responded\_OUT$RT.based\_Groups),   
# FUN = mean, data=targetbeh\_responded\_OUT)  
# names(GM\_LSA\_distance)<- c('Condition', 'RT.based\_Groups','GM\_LSA\_distance')  
#   
# GM\_prop\_lg10CD <- aggregate(targetbeh\_responded\_OUT$prop\_lg10CD,   
# list(targetbeh\_responded\_OUT$Condition, targetbeh\_responded\_OUT$RT.based\_Groups),   
# FUN = mean, data=targetbeh\_responded\_OUT)  
# names(GM\_prop\_lg10CD)<- c('Condition', 'RT.based\_Groups','GM\_prop\_lg10CD')  
#   
# GM\_conc\_lg10CD <- aggregate(targetbeh\_responded\_OUT$conc\_lg10CD,   
# list(targetbeh\_responded\_OUT$Condition, targetbeh\_responded\_OUT$RT.based\_Groups),   
# FUN = mean, data=targetbeh\_responded\_OUT)  
# names(GM\_conc\_lg10CD)<- c('Condition', 'RT.based\_Groups','GM\_conc\_lg10CD')  
#   
# GM\_conc\_exc <- aggregate(targetbeh\_responded\_OUT$conc\_exc,   
# list(targetbeh\_responded\_OUT$Condition, targetbeh\_responded\_OUT$RT.based\_Groups),   
# FUN = mean, data=targetbeh\_responded\_OUT)  
# names(GM\_conc\_exc)<- c('Condition', 'RT.based\_Groups','GM\_conc\_exc')  
#   
# GM\_prop\_exc <- aggregate(targetbeh\_responded\_OUT$prop\_exc,   
# list(targetbeh\_responded\_OUT$Condition, targetbeh\_responded\_OUT$RT.based\_Groups),   
# FUN = mean, data=targetbeh\_responded\_OUT)  
# names(GM\_prop\_exc)<- c('Condition', 'RT.based\_Groups','GM\_prop\_exc')  
#   
# GM\_trial <- aggregate(targetbeh\_responded\_OUT$trial,   
# list(targetbeh\_responded\_OUT$Condition, targetbeh\_responded\_OUT$RT.based\_Groups),   
# FUN = mean, data=targetbeh\_responded\_OUT)  
# names(GM\_trial)<- c('Condition', 'RT.based\_Groups','GM\_trial')  
#   
# GM\_prop\_orthneigh <- aggregate(targetbeh\_responded\_OUT$prop\_orthneigh,   
# list(targetbeh\_responded\_OUT$Condition, targetbeh\_responded\_OUT$RT.based\_Groups),   
# FUN = mean, data=targetbeh\_responded\_OUT)  
# names(GM\_prop\_orthneigh)<- c('Condition', 'RT.based\_Groups','GM\_prop\_orthneigh')  
#   
# GM\_conc\_orthneigh <- aggregate(targetbeh\_responded\_OUT$conc\_orthneigh,   
# list(targetbeh\_responded\_OUT$Condition, targetbeh\_responded\_OUT$RT.based\_Groups),   
# FUN = mean, data=targetbeh\_responded\_OUT)  
# names(GM\_conc\_orthneigh)<- c('Condition', 'RT.based\_Groups','GM\_conc\_orthneigh')  
#   
# GM\_Age\_months <- aggregate(targetbeh\_responded\_OUT$Age\_months,   
# list(targetbeh\_responded\_OUT$Condition, targetbeh\_responded\_OUT$RT.based\_Groups),   
# FUN = mean, data=targetbeh\_responded\_OUT)  
# names(GM\_Age\_months)<- c('Condition', 'RT.based\_Groups','GM\_Age\_months')  
#   
# Add to data set:  
# targetbeh\_responded\_OUT2 <- merge(targetbeh\_responded\_OUT, GM\_conc\_letters,  
# by = c('Condition', 'RT.based\_Groups'))  
# unique(targetbeh\_responded\_OUT2$GM\_conc\_letters) # One mean per cell (inc 0-group)  
#   
# targetbeh\_responded\_OUT3 <- merge(targetbeh\_responded\_OUT2, GM\_prop\_letters,  
# by = c('Condition', 'RT.based\_Groups'))  
#   
# targetbeh\_responded\_OUT4 <- merge(targetbeh\_responded\_OUT3, GM\_LSA\_distance,  
# by = c('Condition', 'RT.based\_Groups'))  
#   
# targetbeh\_responded\_OUT5 <- merge(targetbeh\_responded\_OUT4, GM\_prop\_lg10CD,  
# by = c('Condition', 'RT.based\_Groups'))  
#   
# targetbeh\_responded\_OUT6 <- merge(targetbeh\_responded\_OUT5, GM\_conc\_lg10CD,  
# by = c('Condition', 'RT.based\_Groups'))  
#   
# targetbeh\_responded\_OUT7 <- merge(targetbeh\_responded\_OUT6, GM\_conc\_exc,  
# by = c('Condition', 'RT.based\_Groups'))  
#   
# targetbeh\_responded\_OUT8 <- merge(targetbeh\_responded\_OUT7, GM\_prop\_exc,  
# by = c('Condition', 'RT.based\_Groups'))  
#   
# targetbeh\_responded\_OUT9 <- merge(targetbeh\_responded\_OUT8, GM\_trial,  
# by = c('Condition', 'RT.based\_Groups'))   
#   
# targetbeh\_responded\_OUT10 <- merge(targetbeh\_responded\_OUT9, GM\_prop\_orthneigh,  
# by = c('Condition', 'RT.based\_Groups'))   
#   
# targetbeh\_responded\_OUT11 <- merge(targetbeh\_responded\_OUT10, GM\_conc\_orthneigh,  
# by = c('Condition', 'RT.based\_Groups'))   
#   
# targetbeh\_responded\_OUT12 <- merge(targetbeh\_responded\_OUT11, GM\_Age\_months,  
# by = c('Condition', 'RT.based\_Groups'))   
  
  
# Scaling = dividing by mean by SD  
# save = summaryBy(conc\_letters ~ c(RT.based\_Groups, Condition), targetbeh\_responded\_OUT,   
# FUN=sd)  
# targetbeh\_responded\_OUT <- merge(targetbeh\_responded\_OUT, save, by = c('Condition',   
# 'RT.based\_Groups'))  
# targetbeh\_responded\_OUT$GMCS\_conc\_letters =   
# targetbeh\_responded\_OUT$GM\_conc\_letters / targetbeh\_responded\_OUT$conc\_letters.sd  
# unique(targetbeh\_responded\_OUT$GMCS\_conc\_letters)  
#   
# save = summaryBy(prop\_letters ~ c(RT.based\_Groups, Condition), targetbeh\_responded\_OUT,   
# FUN=sd)  
# targetbeh\_responded\_OUT <- merge(targetbeh\_responded\_OUT, save, by = c('Condition',   
# 'RT.based\_Groups'))  
# targetbeh\_responded\_OUT$GMCS\_prop\_letters =   
# targetbeh\_responded\_OUT$GM\_prop\_letters / targetbeh\_responded\_OUT$prop\_letters.sd  
#   
# save = summaryBy(LSA\_distance ~ c(RT.based\_Groups, Condition), targetbeh\_responded\_OUT,   
# FUN=sd)  
# targetbeh\_responded\_OUT <- merge(targetbeh\_responded\_OUT, save, by = c('Condition',   
# 'RT.based\_Groups'))  
# targetbeh\_responded\_OUT$GMCS\_LSA\_distance =   
# targetbeh\_responded\_OUT$GM\_LSA\_distance / targetbeh\_responded\_OUT$LSA\_distance.sd  
#   
# save = summaryBy(prop\_lg10CD ~ c(RT.based\_Groups, Condition), targetbeh\_responded\_OUT,   
# FUN=sd)  
# targetbeh\_responded\_OUT <- merge(targetbeh\_responded\_OUT, save, by = c('Condition',   
# 'RT.based\_Groups'))  
# targetbeh\_responded\_OUT$GMCS\_prop\_lg10CD =   
# targetbeh\_responded\_OUT$GM\_prop\_lg10CD / targetbeh\_responded\_OUT$prop\_lg10CD.sd  
#   
# save = summaryBy(conc\_lg10CD ~ c(RT.based\_Groups, Condition), targetbeh\_responded\_OUT,   
# FUN=sd)  
# targetbeh\_responded\_OUT <- merge(targetbeh\_responded\_OUT, save, by = c('Condition',   
# 'RT.based\_Groups'))  
# targetbeh\_responded\_OUT$GMCS\_conc\_lg10CD =   
# targetbeh\_responded\_OUT$GM\_conc\_lg10CD / targetbeh\_responded\_OUT$conc\_lg10CD.sd  
#   
# save = summaryBy(conc\_exc ~ c(RT.based\_Groups, Condition), targetbeh\_responded\_OUT, FUN=sd)  
# targetbeh\_responded\_OUT <- merge(targetbeh\_responded\_OUT, save, by = c('Condition',   
# 'RT.based\_Groups'))  
# targetbeh\_responded\_OUT$GMCS\_conc\_exc =   
# targetbeh\_responded\_OUT$GM\_conc\_exc / targetbeh\_responded\_OUT$conc\_exc.sd  
#   
# save = summaryBy(prop\_exc ~ c(RT.based\_Groups, Condition), targetbeh\_responded\_OUT, FUN=sd)  
# targetbeh\_responded\_OUT <- merge(targetbeh\_responded\_OUT, save, by = c('Condition',   
# 'RT.based\_Groups'))  
# targetbeh\_responded\_OUT$GMCS\_prop\_exc =   
# targetbeh\_responded\_OUT$GM\_prop\_exc / targetbeh\_responded\_OUT$prop\_exc.sd  
#   
# save = summaryBy(trial ~ c(RT.based\_Groups, Condition), targetbeh\_responded\_OUT, FUN=sd)  
# targetbeh\_responded\_OUT <- merge(targetbeh\_responded\_OUT, save, by = c('Condition',   
# 'RT.based\_Groups'))  
# targetbeh\_responded\_OUT$GMCS\_trial =   
# targetbeh\_responded\_OUT$GM\_trial / targetbeh\_responded\_OUT$trial.sd  
#   
# save = summaryBy(prop\_orthneigh ~ c(RT.based\_Groups, Condition), targetbeh\_responded\_OUT,   
# FUN=sd)  
# targetbeh\_responded\_OUT <- merge(targetbeh\_responded\_OUT, save, by = c('Condition',   
# 'RT.based\_Groups'))  
# targetbeh\_responded\_OUT$GMCS\_prop\_orthneigh =   
# targetbeh\_responded\_OUT$GM\_prop\_orthneigh / targetbeh\_responded\_OUT$prop\_orthneigh.sd  
#   
# save = summaryBy(conc\_orthneigh ~ c(RT.based\_Groups, Condition), targetbeh\_responded\_OUT,   
# FUN=sd)  
# targetbeh\_responded\_OUT <- merge(targetbeh\_responded\_OUT, save, by = c('Condition',   
# 'RT.based\_Groups'))  
# targetbeh\_responded\_OUT$GMCS\_conc\_orthneigh =   
# targetbeh\_responded\_OUT$GM\_conc\_orthneigh / targetbeh\_responded\_OUT$conc\_orthneigh.sd  
#   
# save = summaryBy(Age\_months ~ c(RT.based\_Groups, Condition), targetbeh\_responded\_OUT,   
# FUN=sd)  
# targetbeh\_responded\_OUT <- merge(targetbeh\_responded\_OUT, save, by = c('Condition',   
# 'RT.based\_Groups'))  
# targetbeh\_responded\_OUT$GMCS\_Age\_months =   
# targetbeh\_responded\_OUT$GM\_Age\_months / targetbeh\_responded\_OUT$Age\_months.sd  
#   
  
# Residuals of the multilevel models are slightly non-normal. Perhaps transform.  
# Five different transformations will be compared with the original measure:  
targetbeh\_responded\_OUT <- cbind(targetbeh\_responded\_OUT,   
targetbeh\_responded\_OUT$RT^2)  
targetbeh\_responded\_OUT <- cbind(targetbeh\_responded\_OUT,   
sqrt(targetbeh\_responded\_OUT$RT))  
targetbeh\_responded\_OUT <- cbind(targetbeh\_responded\_OUT,   
log(targetbeh\_responded\_OUT$RT +1))  
targetbeh\_responded\_OUT <- cbind(targetbeh\_responded\_OUT,   
1/targetbeh\_responded\_OUT$RT)  
targetbeh\_responded\_OUT <- cbind(targetbeh\_responded\_OUT,   
1/sqrt(targetbeh\_responded\_OUT$RT))  
  
targetbeh\_responded\_OUT$sqRT <- targetbeh\_responded\_OUT$RT^2  
targetbeh\_responded\_OUT$sqrtRT <- sqrt(targetbeh\_responded\_OUT$RT)  
targetbeh\_responded\_OUT$logRT <- log(targetbeh\_responded\_OUT$RT +1)  
targetbeh\_responded\_OUT$recRT <- 1/targetbeh\_responded\_OUT$RT  
targetbeh\_responded\_OUT$recsqrtRT <- 1/sqrt(targetbeh\_responded\_OUT$RT)  
  
# save for later  
saveRDS(targetbeh\_responded\_OUT, 'targetbeh\_responded\_OUT.rds')  
  
# read in (once saved, you may take it from here afterwards)  
targetbeh\_responded\_OUT = readRDS('targetbeh\_responded\_OUT.rds')  
head(targetbeh\_responded\_OUT)

## Ptp Group Gender Lefthanded A\_V\_OK Event Trial Item Condition  
## 1 1 Quick F N Y 48 96 1 haptic2visual  
## 2 1 Quick F N Y 18 36 2 haptic2visual  
## 3 1 Quick F N Y 5 10 3 haptic2visual  
## 4 1 Quick F N Y 105 210 4 haptic2visual  
## 5 1 Quick F N Y 100 200 5 haptic2visual  
## 6 1 Quick F N Y 20 40 6 haptic2visual  
## Accuracy RT Position cat catspec solution property  
## 1 rightanswer 544 target visual v\_het Nee Kokend  
## 2 wronganswer 356 target visual v\_de Ja Zwoele  
## 3 wronganswer 516 target visual v\_de Nee Barstende  
## 4 wronganswer 317 target visual v\_de Ja Beschimmelde  
## 5 rightanswer 1219 target visual v\_de Ja Stevige  
## 6 rightanswer 943 target visual v\_de Nee Grove  
## concept LSA\_distance mean\_perceptualstrength mean\_concreteness  
## 1 Mandaat 0.901574 2.306 1.930  
## 2 Levering 0.973728 2.733 2.865  
## 3 Veiligheid NA 2.889 1.930  
## 4 Levering 0.879330 3.333 3.335  
## 5 Veiligheid 0.928228 3.188 2.500  
## 6 Sierlijkheid 0.931037 3.446 2.070  
## mean\_exc prop\_concreteness prop\_exc prop\_lg10CD prop\_orthneigh  
## 1 0.1311920 NA 0.1447368 1.77 4  
## 2 0.1477273 2.53 0.2500000 1.48 2  
## 3 0.2129371 NA 0.1076923 1.57 2  
## 4 0.2727273 3.47 0.5000000 1.20 0  
## 5 0.2744755 3.07 0.2307692 2.38 1  
## 6 0.2744878 2.07 0.2535211 2.16 6  
## prop\_letters conc\_concreteness conc\_exc conc\_lg10CD conc\_letters  
## 1 6 1.93 0.11764706 1.49 7  
## 2 6 3.20 0.04545455 2.19 8  
## 3 9 1.93 0.31818182 3.13 10  
## 4 12 3.20 0.04545455 2.19 8  
## 5 7 1.93 0.31818182 3.13 10  
## 6 5 NA 0.29545455 0.60 12  
## mean\_lg10CD conc\_orthneigh trial accuracy MRT RT.based\_Groups  
## 1 1.630 0 96 1 753.0556 Slow  
## 2 1.835 1 36 0 753.0556 Slow  
## 3 2.350 3 10 0 753.0556 Slow  
## 4 1.695 1 210 0 753.0556 Slow  
## 5 2.755 3 200 1 753.0556 Slow  
## 6 1.380 0 40 1 753.0556 Slow  
## Age\_months ConditionRe s\_conc\_letters s\_prop\_letters s\_LSA\_distance  
## 1 274.5172 Haptic / Visual 0.1314846 -0.52403452 -0.1763344  
## 2 274.5172 Haptic / Visual 0.5408388 -0.52403452 0.6131250  
## 3 274.5172 Haptic / Visual 1.3595472 0.93134742 NA  
## 4 274.5172 Haptic / Visual 0.5408388 2.38672936 -0.4197129  
## 5 274.5172 Haptic / Visual 1.3595472 -0.03890721 0.1152953  
## 6 274.5172 Haptic / Visual 2.1782557 -1.00916184 0.1460295  
## s\_prop\_lg10CD s\_conc\_lg10CD s\_conc\_exc s\_prop\_exc s\_trial  
## 1 -0.2274064 -1.3167464 -1.20132824 -1.4752361 -0.2153609  
## 2 -0.6481800 -0.4030299 -1.69515818 -0.8943639 -1.1788579  
## 3 -0.5175951 0.8239608 0.17042161 -1.6796585 -1.5963733  
## 4 -1.0544441 -0.4030299 -1.69515818 0.4852077 1.6152835  
## 5 0.6576691 0.8239608 0.17042161 -1.0004848 1.4547007  
## 6 0.3384616 -2.4784716 0.01495663 -0.8749333 -1.1146248  
## s\_prop\_orthneigh s\_conc\_orthneigh s\_Age\_months  
## 1 0.1177815 -0.7745407 0.2270487  
## 2 -0.3428422 -0.5507755 0.2270487  
## 3 -0.3428422 -0.1032452 0.2270487  
## 4 -0.8034658 -0.5507755 0.2270487  
## 5 -0.5731540 -0.1032452 0.2270487  
## 6 0.5784051 -0.7745407 0.2270487  
## targetbeh\_responded\_OUT$RT^2 sqrt(targetbeh\_responded\_OUT$RT)  
## 1 295936 23.32381  
## 2 126736 18.86796  
## 3 266256 22.71563  
## 4 100489 17.80449  
## 5 1485961 34.91418  
## 6 889249 30.70831  
## log(targetbeh\_responded\_OUT$RT + 1) 1/targetbeh\_responded\_OUT$RT  
## 1 6.300786 0.0018382353  
## 2 5.877736 0.0028089888  
## 3 6.248043 0.0019379845  
## 4 5.762051 0.0031545741  
## 5 7.106606 0.0008203445  
## 6 6.850126 0.0010604454  
## 1/sqrt(targetbeh\_responded\_OUT$RT) sqRT sqrtRT logRT  
## 1 0.04287465 295936 23.32381 6.300786  
## 2 0.05299989 126736 18.86796 5.877736  
## 3 0.04402255 266256 22.71563 6.248043  
## 4 0.05616560 100489 17.80449 5.762051  
## 5 0.02864166 1485961 34.91418 7.106606  
## 6 0.03256448 889249 30.70831 6.850126  
## recRT recsqrtRT targetbeh\_responded\_OUT$RT^2  
## 1 0.0018382353 0.04287465 295936  
## 2 0.0028089888 0.05299989 126736  
## 3 0.0019379845 0.04402255 266256  
## 4 0.0031545741 0.05616560 100489  
## 5 0.0008203445 0.02864166 1485961  
## 6 0.0010604454 0.03256448 889249  
## sqrt(targetbeh\_responded\_OUT$RT) log(targetbeh\_responded\_OUT$RT + 1)  
## 1 23.32381 6.300786  
## 2 18.86796 5.877736  
## 3 22.71563 6.248043  
## 4 17.80449 5.762051  
## 5 34.91418 7.106606  
## 6 30.70831 6.850126  
## 1/targetbeh\_responded\_OUT$RT 1/sqrt(targetbeh\_responded\_OUT$RT)  
## 1 0.0018382353 0.04287465  
## 2 0.0028089888 0.05299989  
## 3 0.0019379845 0.04402255  
## 4 0.0031545741 0.05616560  
## 5 0.0008203445 0.02864166  
## 6 0.0010604454 0.03256448  
## targetbeh\_responded\_OUT$RT^2 sqrt(targetbeh\_responded\_OUT$RT)  
## 1 295936 23.32381  
## 2 126736 18.86796  
## 3 266256 22.71563  
## 4 100489 17.80449  
## 5 1485961 34.91418  
## 6 889249 30.70831  
## log(targetbeh\_responded\_OUT$RT + 1) 1/targetbeh\_responded\_OUT$RT  
## 1 6.300786 0.0018382353  
## 2 5.877736 0.0028089888  
## 3 6.248043 0.0019379845  
## 4 5.762051 0.0031545741  
## 5 7.106606 0.0008203445  
## 6 6.850126 0.0010604454  
## 1/sqrt(targetbeh\_responded\_OUT$RT) targetbeh\_responded\_OUT$RT^2  
## 1 0.04287465 295936  
## 2 0.05299989 126736  
## 3 0.04402255 266256  
## 4 0.05616560 100489  
## 5 0.02864166 1485961  
## 6 0.03256448 889249  
## sqrt(targetbeh\_responded\_OUT$RT) log(targetbeh\_responded\_OUT$RT + 1)  
## 1 23.32381 6.300786  
## 2 18.86796 5.877736  
## 3 22.71563 6.248043  
## 4 17.80449 5.762051  
## 5 34.91418 7.106606  
## 6 30.70831 6.850126  
## 1/targetbeh\_responded\_OUT$RT 1/sqrt(targetbeh\_responded\_OUT$RT)  
## 1 0.0018382353 0.04287465  
## 2 0.0028089888 0.05299989  
## 3 0.0019379845 0.04402255  
## 4 0.0031545741 0.05616560  
## 5 0.0008203445 0.02864166  
## 6 0.0010604454 0.03256448

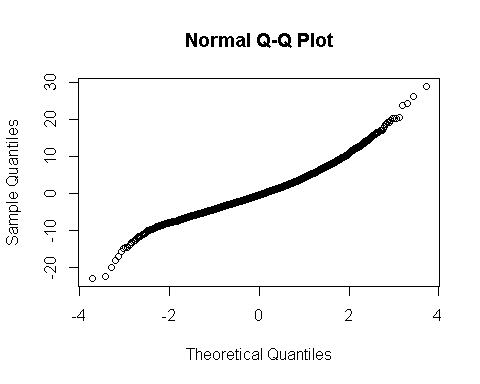
# Check any improvements on the basis of the final model. Q-Q plot is used   
# because z-scores of skewness and kurtosis cannot be checked due to having  
# over 5,000 observations. But Q-Q plot is nearly as straightforward. The  
# ideal distribution presents a uniform, linear increase through the X and   
# Y axes--that is, a diagonal line on the plot.  
  
m9 = lmer(RT ~ 1 + (1 | Item) + (1 | s\_conc\_letters)  
 + (1 | s\_conc\_letters : Ptp)+ (1+ s\_trial | Ptp) + (1| s\_conc\_lg10CD)  
 + (1| s\_conc\_lg10CD : Ptp) + s\_trial + I(s\_trial^2) + s\_conc\_letters,  
 data = targetbeh\_responded\_OUT)  
qqnorm(resid(m9))



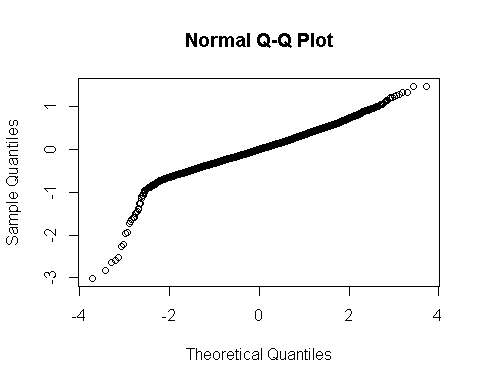
m9 = lmer(sqRT ~ 1 + (1 | Item) + (1 | s\_conc\_letters)  
 + (1 | s\_conc\_letters : Ptp)+ (1+ s\_trial | Ptp) + (1| s\_conc\_lg10CD)  
 + (1| s\_conc\_lg10CD : Ptp) + s\_trial + I(s\_trial^2) + s\_conc\_letters,  
 data = targetbeh\_responded\_OUT)  
qqnorm(resid(m9))



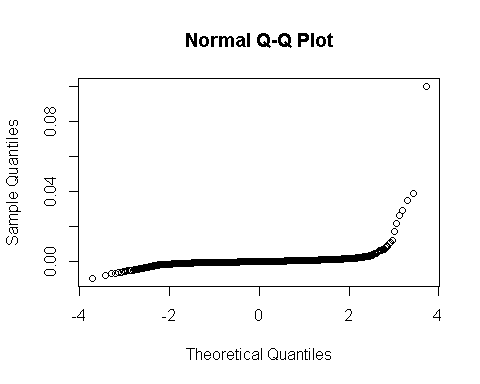
m9 = lmer(sqrtRT ~ 1 + (1 | Item) + (1 | s\_conc\_letters)  
 + (1 | s\_conc\_letters : Ptp)+ (1+ s\_trial | Ptp) + (1| s\_conc\_lg10CD)  
 + (1| s\_conc\_lg10CD : Ptp) + s\_trial + I(s\_trial^2) + s\_conc\_letters,  
 data = targetbeh\_responded\_OUT)  
qqnorm(resid(m9))



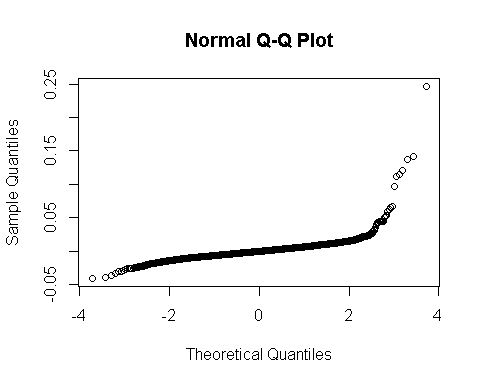
m9 = lmer(logRT ~ 1 + (1 | Item) + (1 | s\_conc\_letters)  
 + (1 | s\_conc\_letters : Ptp)+ (1+ s\_trial | Ptp) + (1| s\_conc\_lg10CD)  
 + (1| s\_conc\_lg10CD : Ptp) + s\_trial + I(s\_trial^2) + s\_conc\_letters,  
 data = targetbeh\_responded\_OUT)  
qqnorm(resid(m9))



m9 = lmer(recRT ~ 1 + (1 | Item) + (1 | s\_conc\_letters)  
 + (1 | s\_conc\_letters : Ptp)+ (1+ s\_trial | Ptp) + (1| s\_conc\_lg10CD)  
 + (1| s\_conc\_lg10CD : Ptp) + s\_trial + I(s\_trial^2) + s\_conc\_letters,  
 data = targetbeh\_responded\_OUT)  
qqnorm(resid(m9))



m9 = lmer(recsqrtRT ~ 1 + (1 | Item) + (1 | s\_conc\_letters)  
 + (1 | s\_conc\_letters : Ptp)+ (1+ s\_trial | Ptp) + (1| s\_conc\_lg10CD)  
 + (1| s\_conc\_lg10CD : Ptp) + s\_trial + I(s\_trial^2) + s\_conc\_letters,  
 data = targetbeh\_responded\_OUT)  
qqnorm(resid(m9))



# Conclusion on transformations check: Best with the raw RT. It'll be maintained.  
  
# IN for the modeling...  
  
# METHOD:  
# These are random slope models with random intercepts and slopes for sub-  
# jects. Both the RT and the EEG data are modeled with the most optimal   
# method as of today. The main characteristic is the maximal, data-driven   
# incorporation of effects, whereby the critical variables of interest are   
# analyzed on the basis of a null model. This null model contains all pos-  
# sible random intercepts and slopes, including interactions, as well as all  
# possible fixed effects and interactions, insofar as they add significantly   
# to that null model (Barr, Levy, Scheepers, & Tily, 2013: J Mem Lang). Both   
# random and fixed effects are analyzed one by one, stepwise, always keeping  
# the degrees of freedom as similar as possible in the two models compared.  
# The null hypothesis significance test is then the Chi-Square based on the   
# Likelihood Ratio Test. This method is optimal with the number of subjects   
# and items of this study (Luke, 2016: Behav Res).  
# IMPORTANT: Models are named ad-hoc. Their numbers are NOT ordinal.  
  
  
# Each model comparison will be commented.   
# First, specify random effects structure.  
  
m0 = lmer(RT ~ 1 + (1 | Ptp),  
 data = targetbeh\_responded\_OUT)  
# random intercepts for participant  
  
m0.1 = lmer(RT ~ 1 + (1 | Ptp) + (1 | Item),  
 data = targetbeh\_responded\_OUT)  
 # + (1 | Ptp : Item) = not accepted, too many groupings  
anova(m0, m0.1) # random intercepts Item: IN

## refitting model(s) with ML (instead of REML)

## Data: targetbeh\_responded\_OUT  
## Models:  
## m0: RT ~ 1 + (1 | Ptp)  
## m0.1: RT ~ 1 + (1 | Ptp) + (1 | Item)  
## Df AIC BIC logLik deviance Chisq Chi Df Pr(>Chisq)   
## m0 3 71818 71837 -35906 71812   
## m0.1 4 71741 71767 -35867 71733 78.547 1 < 2.2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

m1 = lmer(RT ~ 1 + (1 | Ptp) + (1 | Item) + (1 | s\_conc\_letters),   
 data = targetbeh\_responded\_OUT)  
anova(m0.1, m1) # random ints for number letters concept word: IN

## refitting model(s) with ML (instead of REML)

## Data: targetbeh\_responded\_OUT  
## Models:  
## m0.1: RT ~ 1 + (1 | Ptp) + (1 | Item)  
## m1: RT ~ 1 + (1 | Ptp) + (1 | Item) + (1 | s\_conc\_letters)  
## Df AIC BIC logLik deviance Chisq Chi Df Pr(>Chisq)   
## m0.1 4 71741 71767 -35867 71733   
## m1 5 71731 71763 -35860 71721 12.2 1 0.0004779 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

m2.0 = lmer(RT ~ 1+ (1 | Ptp) + (1 | Item) + (1 | s\_conc\_letters)  
 + (1 | s\_conc\_letters : Ptp),   
 data = targetbeh\_responded\_OUT)  
anova(m1, m2.0) # random ints interact concept letters w/ Ptp: IN

## refitting model(s) with ML (instead of REML)

## Data: targetbeh\_responded\_OUT  
## Models:  
## m1: RT ~ 1 + (1 | Ptp) + (1 | Item) + (1 | s\_conc\_letters)  
## m2.0: RT ~ 1 + (1 | Ptp) + (1 | Item) + (1 | s\_conc\_letters) + (1 |   
## m2.0: s\_conc\_letters:Ptp)  
## Df AIC BIC logLik deviance Chisq Chi Df Pr(>Chisq)   
## m1 5 71731 71763 -35860 71721   
## m2.0 6 71725 71764 -35857 71713 7.7914 1 0.00525 \*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

m3 = lmer(RT ~ 1 + (1 | Ptp) + (1 | Item) + (1 | s\_conc\_letters)  
 + (1 | s\_conc\_letters : Ptp) + (1| s\_trial),   
 # + (1| Ptp : s\_trial) = not accepted, too many groupings  
 data = targetbeh\_responded\_OUT)  
anova(m2.0, m3) # random ints trial number: IN

## refitting model(s) with ML (instead of REML)

## Data: targetbeh\_responded\_OUT  
## Models:  
## m2.0: RT ~ 1 + (1 | Ptp) + (1 | Item) + (1 | s\_conc\_letters) + (1 |   
## m2.0: s\_conc\_letters:Ptp)  
## m3: RT ~ 1 + (1 | Ptp) + (1 | Item) + (1 | s\_conc\_letters) + (1 |   
## m3: s\_conc\_letters:Ptp) + (1 | s\_trial)  
## Df AIC BIC logLik deviance Chisq Chi Df Pr(>Chisq)   
## m2.0 6 71725 71764 -35857 71713   
## m3 7 71625 71671 -35806 71611 101.88 1 < 2.2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

m3.2 = lmer(RT ~ 1 + (1 | Item) + (1 | s\_conc\_letters)   
 + (1 | s\_conc\_letters : Ptp) + (1+ s\_trial | Ptp),   
 data = targetbeh\_responded\_OUT)  
anova(m3, m3.2) # random slopes trial number per Ptp: IN

## refitting model(s) with ML (instead of REML)

## Data: targetbeh\_responded\_OUT  
## Models:  
## m3: RT ~ 1 + (1 | Ptp) + (1 | Item) + (1 | s\_conc\_letters) + (1 |   
## m3: s\_conc\_letters:Ptp) + (1 | s\_trial)  
## m3.2: RT ~ 1 + (1 | Item) + (1 | s\_conc\_letters) + (1 | s\_conc\_letters:Ptp) +   
## m3.2: (1 + s\_trial | Ptp)  
## Df AIC BIC logLik deviance Chisq Chi Df Pr(>Chisq)   
## m3 7 71625 71671 -35806 71611   
## m3.2 8 71401 71453 -35692 71385 226.5 1 < 2.2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

m4 = lmer(RT ~ 1 + (1 | Item) + (1 | s\_conc\_letters)   
 + (1 | s\_conc\_letters : Ptp)+ (1+ s\_trial | Ptp) + (1| Accuracy),   
 data = targetbeh\_responded\_OUT)  
anova(m3.2, m4) # random ints Accuracy: out

## refitting model(s) with ML (instead of REML)

## Data: targetbeh\_responded\_OUT  
## Models:  
## m3.2: RT ~ 1 + (1 | Item) + (1 | s\_conc\_letters) + (1 | s\_conc\_letters:Ptp) +   
## m3.2: (1 + s\_trial | Ptp)  
## m4: RT ~ 1 + (1 | Item) + (1 | s\_conc\_letters) + (1 | s\_conc\_letters:Ptp) +   
## m4: (1 + s\_trial | Ptp) + (1 | Accuracy)  
## Df AIC BIC logLik deviance Chisq Chi Df Pr(>Chisq)  
## m3.2 8 71401 71453 -35692 71385   
## m4 9 71403 71461 -35692 71385 0 1 1

m5 = lmer(RT ~ 1 + (1 | Item) + (1 | s\_conc\_letters)   
 + (1 | s\_conc\_letters : Ptp)+ (1+ s\_trial | Ptp) + (1| s\_conc\_lg10CD),   
 data = targetbeh\_responded\_OUT)  
anova(m3.2, m5) # random ints word freq of concept: IN

## refitting model(s) with ML (instead of REML)

## Data: targetbeh\_responded\_OUT  
## Models:  
## m3.2: RT ~ 1 + (1 | Item) + (1 | s\_conc\_letters) + (1 | s\_conc\_letters:Ptp) +   
## m3.2: (1 + s\_trial | Ptp)  
## m5: RT ~ 1 + (1 | Item) + (1 | s\_conc\_letters) + (1 | s\_conc\_letters:Ptp) +   
## m5: (1 + s\_trial | Ptp) + (1 | s\_conc\_lg10CD)  
## Df AIC BIC logLik deviance Chisq Chi Df Pr(>Chisq)   
## m3.2 8 71401 71453 -35692 71385   
## m5 9 71399 71457 -35690 71381 4.021 1 0.04494 \*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

m5.01 = lmer(RT ~ 1 + (1 | Item) + (1 | s\_conc\_letters)   
 + (1 | s\_conc\_letters : Ptp)+ (1+ s\_trial | Ptp) + (1| s\_conc\_lg10CD)  
 + (1| s\_conc\_lg10CD : Ptp),   
 data = targetbeh\_responded\_OUT)  
anova(m5, m5.01) # random ints interact wordfreq concept x participant: out

## refitting model(s) with ML (instead of REML)

## Data: targetbeh\_responded\_OUT  
## Models:  
## m5: RT ~ 1 + (1 | Item) + (1 | s\_conc\_letters) + (1 | s\_conc\_letters:Ptp) +   
## m5: (1 + s\_trial | Ptp) + (1 | s\_conc\_lg10CD)  
## m5.01: RT ~ 1 + (1 | Item) + (1 | s\_conc\_letters) + (1 | s\_conc\_letters:Ptp) +   
## m5.01: (1 + s\_trial | Ptp) + (1 | s\_conc\_lg10CD) + (1 | s\_conc\_lg10CD:Ptp)  
## Df AIC BIC logLik deviance Chisq Chi Df Pr(>Chisq)  
## m5 9 71399 71457 -35690 71381   
## m5.01 10 71399 71464 -35690 71379 1.4225 1 0.233

m5.1 = lmer(RT ~ 1 + (1 | Item) + (1 | s\_conc\_letters)   
 + (1 | s\_conc\_letters : Ptp)+ (1+ s\_trial | Ptp) + (1| s\_conc\_lg10CD)  
 + (1| s\_conc\_orthneigh),   
 data = targetbeh\_responded\_OUT)  
anova(m5, m5.1) # random ints orthograph neighbourhood size concept: out

## refitting model(s) with ML (instead of REML)

## Data: targetbeh\_responded\_OUT  
## Models:  
## m5: RT ~ 1 + (1 | Item) + (1 | s\_conc\_letters) + (1 | s\_conc\_letters:Ptp) +   
## m5: (1 + s\_trial | Ptp) + (1 | s\_conc\_lg10CD)  
## m5.1: RT ~ 1 + (1 | Item) + (1 | s\_conc\_letters) + (1 | s\_conc\_letters:Ptp) +   
## m5.1: (1 + s\_trial | Ptp) + (1 | s\_conc\_lg10CD) + (1 | s\_conc\_orthneigh)  
## Df AIC BIC logLik deviance Chisq Chi Df Pr(>Chisq)  
## m5 9 71399 71457 -35690 71381   
## m5.1 10 71400 71465 -35690 71380 1.0599 1 0.3032

m5.2 = lmer(RT ~ 1 + (1 | Item) + (1 | s\_conc\_letters)   
 + (1 | s\_conc\_letters : Ptp)+ (1+ s\_trial | Ptp) + (1| s\_conc\_lg10CD)  
 + (1| s\_prop\_orthneigh),   
 data = targetbeh\_responded\_OUT)  
anova(m5, m5.2) # random ints orthograph neighbourhood size property: out

## refitting model(s) with ML (instead of REML)

## Data: targetbeh\_responded\_OUT  
## Models:  
## m5: RT ~ 1 + (1 | Item) + (1 | s\_conc\_letters) + (1 | s\_conc\_letters:Ptp) +   
## m5: (1 + s\_trial | Ptp) + (1 | s\_conc\_lg10CD)  
## m5.2: RT ~ 1 + (1 | Item) + (1 | s\_conc\_letters) + (1 | s\_conc\_letters:Ptp) +   
## m5.2: (1 + s\_trial | Ptp) + (1 | s\_conc\_lg10CD) + (1 | s\_prop\_orthneigh)  
## Df AIC BIC logLik deviance Chisq Chi Df Pr(>Chisq)  
## m5 9 71399 71457 -35690 71381   
## m5.2 10 71399 71464 -35690 71379 1.5022 1 0.2203

# To check LSA/semantic distance property-concept, subset to trials w/ scores,  
# i.e., 79% of the trials.  
mWith = lmer(RT ~ 1 + (1 | Item) + (1 | s\_conc\_letters)   
 + (1 | s\_conc\_letters : Ptp)+ (1+ s\_trial | Ptp) + (1| s\_conc\_lg10CD),   
 data = targetbeh\_responded\_OUT[!is.na(targetbeh\_responded\_OUT$s\_LSA\_distance),])  
  
m5.22 = lmer(RT ~ 1 + (1 | Item) + (1 | s\_conc\_letters)   
 + (1 | s\_conc\_letters : Ptp)+ (1+ s\_trial | Ptp) + (1| s\_conc\_lg10CD)  
 + (1| s\_LSA\_distance),   
 data = targetbeh\_responded\_OUT[!is.na(targetbeh\_responded\_OUT$s\_LSA\_distance),])

## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, : Model is nearly unidentifiable: large eigenvalue ratio  
## - Rescale variables?

# random intercepts semantic distance property-concept: unidentifiable  
  
#...Back to the full data  
m5.3 = lmer(RT ~ 1 + (1 | Item) + (1 | s\_conc\_letters)  
 + (1 | s\_conc\_letters : Ptp)+ (1+ s\_trial | Ptp) + (1| s\_conc\_lg10CD)  
 + (1| solution),  
 data = targetbeh\_responded\_OUT)  
anova(m5, m5.3) # random ints solution: out

## refitting model(s) with ML (instead of REML)

## Data: targetbeh\_responded\_OUT  
## Models:  
## m5: RT ~ 1 + (1 | Item) + (1 | s\_conc\_letters) + (1 | s\_conc\_letters:Ptp) +   
## m5: (1 + s\_trial | Ptp) + (1 | s\_conc\_lg10CD)  
## m5.3: RT ~ 1 + (1 | Item) + (1 | s\_conc\_letters) + (1 | s\_conc\_letters:Ptp) +   
## m5.3: (1 + s\_trial | Ptp) + (1 | s\_conc\_lg10CD) + (1 | solution)  
## Df AIC BIC logLik deviance Chisq Chi Df Pr(>Chisq)  
## m5 9 71399 71457 -35690 71381   
## m5.3 10 71401 71466 -35690 71381 0 1 1

m6 = lmer(RT ~ 1 + (1 | Item) + (1 | s\_conc\_letters)   
 + (1 | s\_conc\_letters : Ptp)+ (1+ s\_trial | Ptp) + (1| s\_conc\_lg10CD)  
 + (1| RT.based\_Groups),  
 data = targetbeh\_responded\_OUT)  
anova(m5, m6) # random ints Group: IN

## refitting model(s) with ML (instead of REML)

## Data: targetbeh\_responded\_OUT  
## Models:  
## m5: RT ~ 1 + (1 | Item) + (1 | s\_conc\_letters) + (1 | s\_conc\_letters:Ptp) +   
## m5: (1 + s\_trial | Ptp) + (1 | s\_conc\_lg10CD)  
## m6: RT ~ 1 + (1 | Item) + (1 | s\_conc\_letters) + (1 | s\_conc\_letters:Ptp) +   
## m6: (1 + s\_trial | Ptp) + (1 | s\_conc\_lg10CD) + (1 | RT.based\_Groups)  
## Df AIC BIC logLik deviance Chisq Chi Df Pr(>Chisq)   
## m5 9 71399 71457 -35690 71381   
## m6 10 71381 71447 -35681 71361 19.334 1 1.097e-05 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

m7 = lmer(RT ~ 1 + (1 | Item) + (1 | s\_conc\_letters)   
 + (1 | s\_conc\_letters : Ptp)+ (1+ s\_trial | Ptp) + (1| s\_conc\_lg10CD)  
 + (1| RT.based\_Groups) + (1| Condition),   
 data = targetbeh\_responded\_OUT)  
anova(m6, m7) # Random ints Condition: OUT

## refitting model(s) with ML (instead of REML)

## Data: targetbeh\_responded\_OUT  
## Models:  
## m6: RT ~ 1 + (1 | Item) + (1 | s\_conc\_letters) + (1 | s\_conc\_letters:Ptp) +   
## m6: (1 + s\_trial | Ptp) + (1 | s\_conc\_lg10CD) + (1 | RT.based\_Groups)  
## m7: RT ~ 1 + (1 | Item) + (1 | s\_conc\_letters) + (1 | s\_conc\_letters:Ptp) +   
## m7: (1 + s\_trial | Ptp) + (1 | s\_conc\_lg10CD) + (1 | RT.based\_Groups) +   
## m7: (1 | Condition)  
## Df AIC BIC logLik deviance Chisq Chi Df Pr(>Chisq)  
## m6 10 71381 71447 -35681 71361   
## m7 11 71383 71455 -35681 71361 0 1 1

m7.01 = lmer(RT ~ 1 + (1 | Item) + (1 | s\_conc\_letters)  
 + (1 | s\_conc\_letters : Ptp)+ (1+ s\_trial | Ptp) + (1| s\_conc\_lg10CD)  
 + (1| RT.based\_Groups) + (1| s\_Age\_months),  
 data = targetbeh\_responded\_OUT)  
anova(m6, m7.01) # random ints age: unidentifiable

## refitting model(s) with ML (instead of REML)

## Data: targetbeh\_responded\_OUT  
## Models:  
## m6: RT ~ 1 + (1 | Item) + (1 | s\_conc\_letters) + (1 | s\_conc\_letters:Ptp) +   
## m6: (1 + s\_trial | Ptp) + (1 | s\_conc\_lg10CD) + (1 | RT.based\_Groups)  
## m7.01: RT ~ 1 + (1 | Item) + (1 | s\_conc\_letters) + (1 | s\_conc\_letters:Ptp) +   
## m7.01: (1 + s\_trial | Ptp) + (1 | s\_conc\_lg10CD) + (1 | RT.based\_Groups) +   
## m7.01: (1 | s\_Age\_months)  
## Df AIC BIC logLik deviance Chisq Chi Df Pr(>Chisq)  
## m6 10 71381 71447 -35681 71361   
## m7.01 11 71383 71455 -35681 71361 0 1 1

m7.1 = lmer(RT ~ 1 + (1 | Item) + (1 | s\_conc\_letters)  
 + (1 | s\_conc\_letters : Ptp)+ (1+ s\_trial | Ptp) + (1| s\_conc\_lg10CD)  
 + (1| RT.based\_Groups) + (1| Gender),  
 data = targetbeh\_responded\_OUT)  
anova(m6, m7.1) # random ints gender: OUT

## refitting model(s) with ML (instead of REML)

## Data: targetbeh\_responded\_OUT  
## Models:  
## m6: RT ~ 1 + (1 | Item) + (1 | s\_conc\_letters) + (1 | s\_conc\_letters:Ptp) +   
## m6: (1 + s\_trial | Ptp) + (1 | s\_conc\_lg10CD) + (1 | RT.based\_Groups)  
## m7.1: RT ~ 1 + (1 | Item) + (1 | s\_conc\_letters) + (1 | s\_conc\_letters:Ptp) +   
## m7.1: (1 + s\_trial | Ptp) + (1 | s\_conc\_lg10CD) + (1 | RT.based\_Groups) +   
## m7.1: (1 | Gender)  
## Df AIC BIC logLik deviance Chisq Chi Df Pr(>Chisq)  
## m6 10 71381 71447 -35681 71361   
## m7.1 11 71383 71455 -35681 71361 0 1 1

m7.11 = lmer(RT ~ 1 + (1 | Item) + (1 | s\_conc\_letters)  
 + (1 | s\_conc\_letters : Ptp)+ (1+ s\_trial | Ptp) + (1| s\_conc\_lg10CD)  
 + (1| RT.based\_Groups) + (1| Lefthanded),  
 data = targetbeh\_responded\_OUT)  
anova(m6, m7.11) # random ints Handedness: OUT

## refitting model(s) with ML (instead of REML)

## Data: targetbeh\_responded\_OUT  
## Models:  
## m6: RT ~ 1 + (1 | Item) + (1 | s\_conc\_letters) + (1 | s\_conc\_letters:Ptp) +   
## m6: (1 + s\_trial | Ptp) + (1 | s\_conc\_lg10CD) + (1 | RT.based\_Groups)  
## m7.11: RT ~ 1 + (1 | Item) + (1 | s\_conc\_letters) + (1 | s\_conc\_letters:Ptp) +   
## m7.11: (1 + s\_trial | Ptp) + (1 | s\_conc\_lg10CD) + (1 | RT.based\_Groups) +   
## m7.11: (1 | Lefthanded)  
## Df AIC BIC logLik deviance Chisq Chi Df Pr(>Chisq)  
## m6 10 71381 71447 -35681 71361   
## m7.11 11 71383 71455 -35681 71361 0 1 1

########### Random effs structure set at m6. On to fixed effects:  
  
m7.2 = lmer(RT ~ (1 | Item) + (1 | s\_conc\_letters)  
 + (1 | s\_conc\_letters : Ptp)+ (1+ s\_trial | Ptp) + (1| s\_conc\_lg10CD)  
 + (1| RT.based\_Groups) + s\_trial,  
 data = targetbeh\_responded\_OUT)  
anova(m6, m7.2) # trial number: IN

## refitting model(s) with ML (instead of REML)

## Data: targetbeh\_responded\_OUT  
## Models:  
## m6: RT ~ 1 + (1 | Item) + (1 | s\_conc\_letters) + (1 | s\_conc\_letters:Ptp) +   
## m6: (1 + s\_trial | Ptp) + (1 | s\_conc\_lg10CD) + (1 | RT.based\_Groups)  
## m7.2: RT ~ (1 | Item) + (1 | s\_conc\_letters) + (1 | s\_conc\_letters:Ptp) +   
## m7.2: (1 + s\_trial | Ptp) + (1 | s\_conc\_lg10CD) + (1 | RT.based\_Groups) +   
## m7.2: s\_trial  
## Df AIC BIC logLik deviance Chisq Chi Df Pr(>Chisq)   
## m6 10 71381 71447 -35681 71361   
## m7.2 11 71358 71430 -35668 71336 25.099 1 5.446e-07 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

m8 = lmer(RT ~ (1 | Item) + (1 | s\_conc\_letters)  
 + (1 | s\_conc\_letters : Ptp)+ (1+ s\_trial | Ptp) + (1| s\_conc\_lg10CD)  
 + (1| RT.based\_Groups) + s\_trial + I(s\_trial^2),  
 data = targetbeh\_responded\_OUT)  
anova(m7.2, m8) # quadratic trend for trial number: IN

## refitting model(s) with ML (instead of REML)

## Data: targetbeh\_responded\_OUT  
## Models:  
## m7.2: RT ~ (1 | Item) + (1 | s\_conc\_letters) + (1 | s\_conc\_letters:Ptp) +   
## m7.2: (1 + s\_trial | Ptp) + (1 | s\_conc\_lg10CD) + (1 | RT.based\_Groups) +   
## m7.2: s\_trial  
## m8: RT ~ (1 | Item) + (1 | s\_conc\_letters) + (1 | s\_conc\_letters:Ptp) +   
## m8: (1 + s\_trial | Ptp) + (1 | s\_conc\_lg10CD) + (1 | RT.based\_Groups) +   
## m8: s\_trial + I(s\_trial^2)  
## Df AIC BIC logLik deviance Chisq Chi Df Pr(>Chisq)   
## m7.2 11 71358 71430 -35668 71336   
## m8 12 71334 71413 -35655 71310 25.856 1 3.679e-07 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

m8.0 = lmer(RT ~ (1 | Item) + (1 | s\_conc\_letters)  
 + (1 | s\_conc\_letters : Ptp)+ (1+ s\_trial | Ptp) + (1| s\_conc\_lg10CD)  
 + (1| RT.based\_Groups) + s\_trial + I(s\_trial^2)+ I(s\_trial^3),  
 data = targetbeh\_responded\_OUT)  
anova(m8, m8.0) # cubic trend for trial number: IN

## refitting model(s) with ML (instead of REML)

## Data: targetbeh\_responded\_OUT  
## Models:  
## m8: RT ~ (1 | Item) + (1 | s\_conc\_letters) + (1 | s\_conc\_letters:Ptp) +   
## m8: (1 + s\_trial | Ptp) + (1 | s\_conc\_lg10CD) + (1 | RT.based\_Groups) +   
## m8: s\_trial + I(s\_trial^2)  
## m8.0: RT ~ (1 | Item) + (1 | s\_conc\_letters) + (1 | s\_conc\_letters:Ptp) +   
## m8.0: (1 + s\_trial | Ptp) + (1 | s\_conc\_lg10CD) + (1 | RT.based\_Groups) +   
## m8.0: s\_trial + I(s\_trial^2) + I(s\_trial^3)  
## Df AIC BIC logLik deviance Chisq Chi Df Pr(>Chisq)   
## m8 12 71334 71413 -35655 71310   
## m8.0 13 71332 71417 -35653 71306 4.1239 1 0.04228 \*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

m8.01 = lmer(RT ~ (1 | Item) + (1 | s\_conc\_letters)  
 + (1 | s\_conc\_letters : Ptp)+ (1+ s\_trial | Ptp) + (1| s\_conc\_lg10CD)  
 + (1| RT.based\_Groups) + s\_trial + I(s\_trial^2)+ I(s\_trial^3) + s\_conc\_lg10CD,  
 data = targetbeh\_responded\_OUT)  
anova(m8.0, m8.01) # word frequency of the concept: out

## refitting model(s) with ML (instead of REML)

## Data: targetbeh\_responded\_OUT  
## Models:  
## m8.0: RT ~ (1 | Item) + (1 | s\_conc\_letters) + (1 | s\_conc\_letters:Ptp) +   
## m8.0: (1 + s\_trial | Ptp) + (1 | s\_conc\_lg10CD) + (1 | RT.based\_Groups) +   
## m8.0: s\_trial + I(s\_trial^2) + I(s\_trial^3)  
## m8.01: RT ~ (1 | Item) + (1 | s\_conc\_letters) + (1 | s\_conc\_letters:Ptp) +   
## m8.01: (1 + s\_trial | Ptp) + (1 | s\_conc\_lg10CD) + (1 | RT.based\_Groups) +   
## m8.01: s\_trial + I(s\_trial^2) + I(s\_trial^3) + s\_conc\_lg10CD  
## Df AIC BIC logLik deviance Chisq Chi Df Pr(>Chisq)  
## m8.0 13 71332 71417 -35653 71306   
## m8.01 14 71333 71425 -35653 71305 0.8181 1 0.3657

m8.1 = lmer(RT ~ (1 | Item) + (1 | s\_conc\_letters)  
 + (1 | s\_conc\_letters : Ptp)+ (1+ s\_trial | Ptp) + (1| s\_conc\_lg10CD)  
 + (1| RT.based\_Groups) + s\_trial + I(s\_trial^2)+ I(s\_trial^3) + s\_Age\_months,  
 data = targetbeh\_responded\_OUT)  
anova(m8.0, m8.1) # age: out

## refitting model(s) with ML (instead of REML)

## Data: targetbeh\_responded\_OUT  
## Models:  
## m8.0: RT ~ (1 | Item) + (1 | s\_conc\_letters) + (1 | s\_conc\_letters:Ptp) +   
## m8.0: (1 + s\_trial | Ptp) + (1 | s\_conc\_lg10CD) + (1 | RT.based\_Groups) +   
## m8.0: s\_trial + I(s\_trial^2) + I(s\_trial^3)  
## m8.1: RT ~ (1 | Item) + (1 | s\_conc\_letters) + (1 | s\_conc\_letters:Ptp) +   
## m8.1: (1 + s\_trial | Ptp) + (1 | s\_conc\_lg10CD) + (1 | RT.based\_Groups) +   
## m8.1: s\_trial + I(s\_trial^2) + I(s\_trial^3) + s\_Age\_months  
## Df AIC BIC logLik deviance Chisq Chi Df Pr(>Chisq)   
## m8.0 13 71332 71417 -35653 71306   
## m8.1 14 71331 71423 -35652 71303 2.8072 1 0.09384 .  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

m8.2 = lmer(RT ~ (1 | Item) + (1 | s\_conc\_letters)  
 + (1 | s\_conc\_letters : Ptp)+ (1+ s\_trial | Ptp) + (1| s\_conc\_lg10CD)  
 + (1| RT.based\_Groups) + s\_trial + I(s\_trial^2)+ I(s\_trial^3) + Gender,  
 data = targetbeh\_responded\_OUT)  
anova(m8.0, m8.2) # participant's gender: out

## refitting model(s) with ML (instead of REML)

## Data: targetbeh\_responded\_OUT  
## Models:  
## m8.0: RT ~ (1 | Item) + (1 | s\_conc\_letters) + (1 | s\_conc\_letters:Ptp) +   
## m8.0: (1 + s\_trial | Ptp) + (1 | s\_conc\_lg10CD) + (1 | RT.based\_Groups) +   
## m8.0: s\_trial + I(s\_trial^2) + I(s\_trial^3)  
## m8.2: RT ~ (1 | Item) + (1 | s\_conc\_letters) + (1 | s\_conc\_letters:Ptp) +   
## m8.2: (1 + s\_trial | Ptp) + (1 | s\_conc\_lg10CD) + (1 | RT.based\_Groups) +   
## m8.2: s\_trial + I(s\_trial^2) + I(s\_trial^3) + Gender  
## Df AIC BIC logLik deviance Chisq Chi Df Pr(>Chisq)  
## m8.0 13 71332 71417 -35653 71306   
## m8.2 14 71333 71425 -35653 71305 0.7804 1 0.377

m8.3 = lmer(RT ~ (1 | Item) + (1 | s\_conc\_letters)  
 + (1 | s\_conc\_letters : Ptp)+ (1+ s\_trial | Ptp) + (1| s\_conc\_lg10CD)  
 + (1| RT.based\_Groups) + s\_trial + I(s\_trial^2)+ I(s\_trial^3) + Lefthanded,  
 data = targetbeh\_responded\_OUT)  
anova(m8.0, m8.3) # handedness: out

## refitting model(s) with ML (instead of REML)

## Data: targetbeh\_responded\_OUT  
## Models:  
## m8.0: RT ~ (1 | Item) + (1 | s\_conc\_letters) + (1 | s\_conc\_letters:Ptp) +   
## m8.0: (1 + s\_trial | Ptp) + (1 | s\_conc\_lg10CD) + (1 | RT.based\_Groups) +   
## m8.0: s\_trial + I(s\_trial^2) + I(s\_trial^3)  
## m8.3: RT ~ (1 | Item) + (1 | s\_conc\_letters) + (1 | s\_conc\_letters:Ptp) +   
## m8.3: (1 + s\_trial | Ptp) + (1 | s\_conc\_lg10CD) + (1 | RT.based\_Groups) +   
## m8.3: s\_trial + I(s\_trial^2) + I(s\_trial^3) + Lefthanded  
## Df AIC BIC logLik deviance Chisq Chi Df Pr(>Chisq)  
## m8.0 13 71332 71417 -35653 71306   
## m8.3 14 71334 71425 -35653 71306 0.2827 1 0.5949

m9 = lmer(RT ~ (1 | Item) + (1 | s\_conc\_letters)  
 + (1 | s\_conc\_letters : Ptp)+ (1+ s\_trial | Ptp) + (1| s\_conc\_lg10CD)  
 + (1| RT.based\_Groups) + s\_trial + I(s\_trial^2)+ I(s\_trial^3) + s\_conc\_letters,  
 data = targetbeh\_responded\_OUT)  
anova(m8.0, m9) # number of letters of the concept: IN

## refitting model(s) with ML (instead of REML)

## Data: targetbeh\_responded\_OUT  
## Models:  
## m8.0: RT ~ (1 | Item) + (1 | s\_conc\_letters) + (1 | s\_conc\_letters:Ptp) +   
## m8.0: (1 + s\_trial | Ptp) + (1 | s\_conc\_lg10CD) + (1 | RT.based\_Groups) +   
## m8.0: s\_trial + I(s\_trial^2) + I(s\_trial^3)  
## m9: RT ~ (1 | Item) + (1 | s\_conc\_letters) + (1 | s\_conc\_letters:Ptp) +   
## m9: (1 + s\_trial | Ptp) + (1 | s\_conc\_lg10CD) + (1 | RT.based\_Groups) +   
## m9: s\_trial + I(s\_trial^2) + I(s\_trial^3) + s\_conc\_letters  
## Df AIC BIC logLik deviance Chisq Chi Df Pr(>Chisq)   
## m8.0 13 71332 71417 -35653 71306   
## m9 14 71327 71418 -35649 71299 7.6205 1 0.005771 \*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

########## Null model set: m8  
# Data-driven null model: m8  
  
# 1. ME Condition  
mCondition = lmer(RT ~ (1 | Item) + (1 | s\_conc\_letters)  
 + (1 | s\_conc\_letters : Ptp)+ (1+ s\_trial | Ptp) + (1| s\_conc\_lg10CD)  
 + (1| RT.based\_Groups) + s\_trial + I(s\_trial^2)+ I(s\_trial^3) + s\_conc\_letters   
 + Condition,  
 data = targetbeh\_responded\_OUT)  
anova(m9, mCondition) # Condition: out

## refitting model(s) with ML (instead of REML)

## Data: targetbeh\_responded\_OUT  
## Models:  
## m9: RT ~ (1 | Item) + (1 | s\_conc\_letters) + (1 | s\_conc\_letters:Ptp) +   
## m9: (1 + s\_trial | Ptp) + (1 | s\_conc\_lg10CD) + (1 | RT.based\_Groups) +   
## m9: s\_trial + I(s\_trial^2) + I(s\_trial^3) + s\_conc\_letters  
## mCondition: RT ~ (1 | Item) + (1 | s\_conc\_letters) + (1 | s\_conc\_letters:Ptp) +   
## mCondition: (1 + s\_trial | Ptp) + (1 | s\_conc\_lg10CD) + (1 | RT.based\_Groups) +   
## mCondition: s\_trial + I(s\_trial^2) + I(s\_trial^3) + s\_conc\_letters +   
## mCondition: Condition  
## Df AIC BIC logLik deviance Chisq Chi Df Pr(>Chisq)  
## m9 14 71327 71418 -35649 71299   
## mCondition 16 71328 71432 -35648 71296 2.5808 2 0.2752

# Group effects  
mGroup = lmer(RT ~ (1 | Item) + (1 | s\_conc\_letters)  
 + (1 | s\_conc\_letters : Ptp)+ (1+ s\_trial | Ptp) + (1| s\_conc\_lg10CD)  
 + (1| RT.based\_Groups) + s\_trial + I(s\_trial^2)+ I(s\_trial^3) + s\_conc\_letters  
 + RT.based\_Groups,  
 data = targetbeh\_responded\_OUT)  
anova(m9, mGroup) # Group: IN

## refitting model(s) with ML (instead of REML)

## Data: targetbeh\_responded\_OUT  
## Models:  
## m9: RT ~ (1 | Item) + (1 | s\_conc\_letters) + (1 | s\_conc\_letters:Ptp) +   
## m9: (1 + s\_trial | Ptp) + (1 | s\_conc\_lg10CD) + (1 | RT.based\_Groups) +   
## m9: s\_trial + I(s\_trial^2) + I(s\_trial^3) + s\_conc\_letters  
## mGroup: RT ~ (1 | Item) + (1 | s\_conc\_letters) + (1 | s\_conc\_letters:Ptp) +   
## mGroup: (1 + s\_trial | Ptp) + (1 | s\_conc\_lg10CD) + (1 | RT.based\_Groups) +   
## mGroup: s\_trial + I(s\_trial^2) + I(s\_trial^3) + s\_conc\_letters +   
## mGroup: RT.based\_Groups  
## Df AIC BIC logLik deviance Chisq Chi Df Pr(>Chisq)   
## m9 14 71327 71418 -35649 71299   
## mGroup 15 71321 71419 -35645 71291 7.6677 1 0.005622 \*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

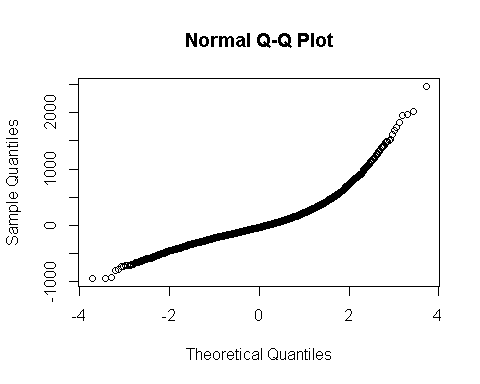
# Interaction Condition\*Group  
mIntGroupCondition = lmer(RT ~ (1 | Item) + (1 | s\_conc\_letters)  
 + (1 | s\_conc\_letters : Ptp)+ (1+ s\_trial | Ptp) + (1| s\_conc\_lg10CD)  
 + (1| RT.based\_Groups) + s\_trial + I(s\_trial^2)+ I(s\_trial^3) + s\_conc\_letters  
 + RT.based\_Groups : Condition,  
 data = targetbeh\_responded\_OUT)

## fixed-effect model matrix is rank deficient so dropping 1 column / coefficient

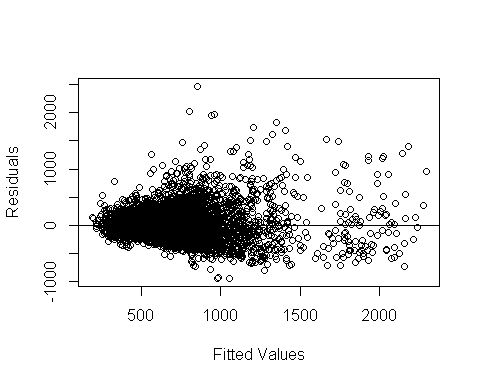
# Rank deficient: model would want to have main eff of either factor, but  
# those were tested out above already.  
  
  
# MODEL SELECTED:  
summary(mGroup)

## Linear mixed model fit by REML ['lmerMod']  
## Formula: RT ~ (1 | Item) + (1 | s\_conc\_letters) + (1 | s\_conc\_letters:Ptp) +   
## (1 + s\_trial | Ptp) + (1 | s\_conc\_lg10CD) + (1 | RT.based\_Groups) +   
## s\_trial + I(s\_trial^2) + I(s\_trial^3) + s\_conc\_letters + RT.based\_Groups  
## Data: targetbeh\_responded\_OUT  
##   
## REML criterion at convergence: 71249.2  
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -3.2631 -0.5793 -0.1362 0.4021 8.5385   
##   
## Random effects:  
## Groups Name Variance Std.Dev. Corr   
## s\_conc\_letters:Ptp (Intercept) 1392.2 37.31   
## Item (Intercept) 1796.8 42.39   
## Ptp (Intercept) 40911.5 202.27   
## s\_trial 4419.8 66.48 -0.23  
## s\_conc\_lg10CD (Intercept) 898.5 29.97   
## s\_conc\_letters (Intercept) 721.3 26.86   
## RT.based\_Groups (Intercept) 16541.8 128.62   
## Residual 83637.5 289.20   
## Number of obs: 5000, groups:   
## s\_conc\_letters:Ptp, 470; Item, 108; Ptp, 47; s\_conc\_lg10CD, 45; s\_conc\_letters, 10; RT.based\_Groups, 2  
##   
## Fixed effects:  
## Estimate Std. Error t value  
## (Intercept) 539.653 135.657 3.978  
## s\_trial -40.607 14.162 -2.867  
## I(s\_trial^2) 23.355 4.617 5.058  
## I(s\_trial^3) -10.733 5.248 -2.045  
## s\_conc\_letters 32.497 11.321 2.871  
## RT.based\_GroupsSlow 346.276 191.041 1.813  
##   
## Correlation of Fixed Effects:  
## (Intr) s\_tril I(\_^2) I(\_^3) s\_cnc\_  
## s\_trial -0.034   
## I(s\_tril^2) -0.034 -0.009   
## I(s\_tril^3) 0.000 -0.668 0.016   
## s\_cnc\_lttrs -0.019 0.004 -0.004 -0.004   
## RT.bsd\_GrpS -0.703 0.000 0.000 0.000 0.000

# Diagnostics:  
# Normal Q-Q plot of residuals:  
qqnorm(resid(mGroup))

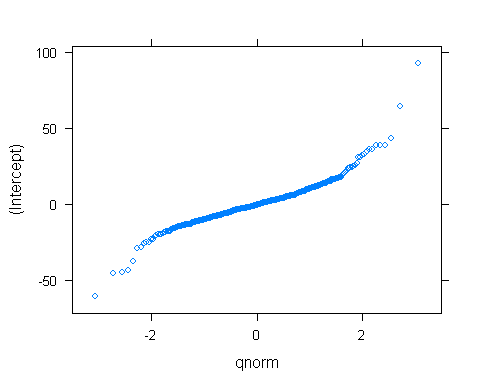


# Plot of Residuals vs. Fitted values:  
plot(fitted(mGroup),resid(mGroup),xlab='Fitted Values', ylab='Residuals');   
abline(h=0)

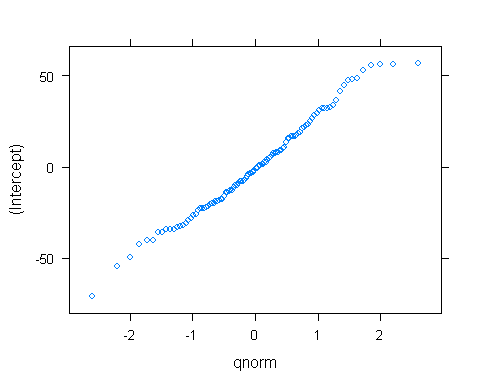


# random effs  
plot(ranef(mGroup))

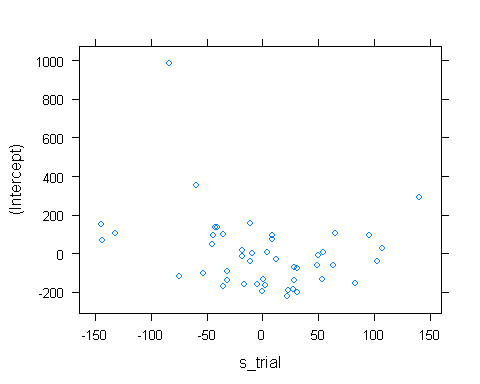
## $`s\_conc\_letters:Ptp`



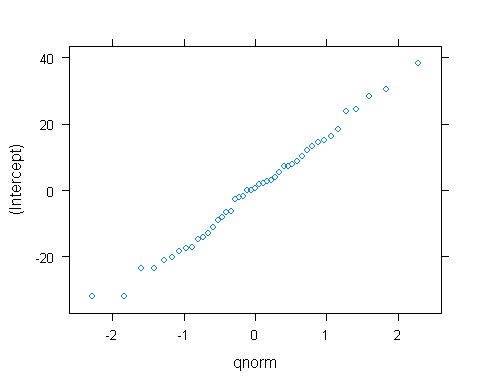
##   
## $Item



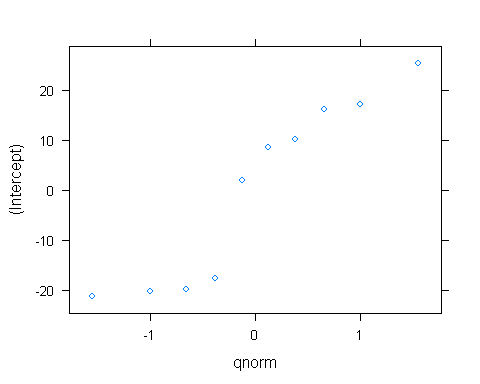
##   
## $Ptp



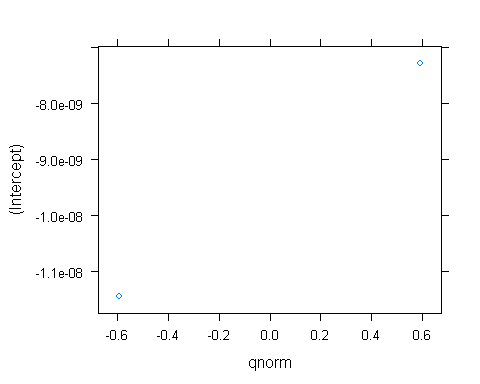
##   
## $s\_conc\_lg10CD



##   
## $s\_conc\_letters



##   
## $RT.based\_Groups



# RESULTS: non-normal residuals.  
#  
  
# Fit  
1-var(residuals(mGroup))/(var(model.response(model.frame(mGroup)))) # Omega^2

## [1] 0.5207764

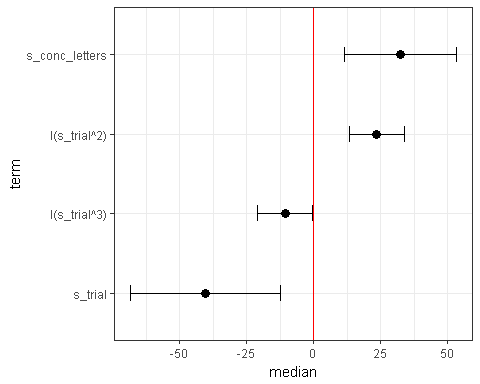
r.squaredGLMM(mGroup) # R2m: fixed effs. R2c: fixed + random effs

## R2m R2c   
## 0.1898139 0.5492111

RMSE.merMod(mGroup, scale = FALSE) # Root MSE

## [1] 283.1468

plotFEsim(FEsim(m9))



#