Modality exclusivity norms in Dutch: code with results

Pablo Bernabeu

August 2016

# Code for a psycholinguistic study about conceptual modality, part of P. Bernabeu's  
# MPhil thesis. 'Modality exclusivity norms for 747 properties and concepts in Dutch:  
# a replication of English'. More info at: https://goo.gl/Je4JGO.   
# Contact: pcbernabeu@gmail.com  
  
  
# READ-ME  
# The 'all.csv' file, created outside of R, in Excel, compiles all individual ratings.  
# Dutch and English data are described in separate columns. All analyses separate for   
# properties and concepts, except for a translation check.   
# Stat tests (specifying treatment of English and Dutch norms): reliability analysis  
# (only Dutch norms), Pearson’s correlation (norms independent and paired), one-sample   
# t-test (norms independent), Principal Components Analysis (norms independent), ANOVA   
# (norms paired), and multiple regression (norms independent).  
# The code is extensively annotated, but some clarifications are in order. Subsetting is  
# done throughout the code, and is essential due to the different norms (see 'normed'  
# column: English, Dutch, or both). Subsetting is often done on the basis of variables  
# that are unique to either norms, especially, 'Exclusivity' and 'exc\_eng'.  
# At first, the code must be run right from the top, as different objects bear the   
# same name. In its entirety, it takes ~20 mins. Note the annotations for theoretical   
# matters. Long variables are never presented entirely, but rather in sections or via  
# summaries. Yet, the reader is invited to edit and present them entirely.  
# Written on R version 3.2.2 (2015-08-14). The PDF and Word markdowns, with about 150   
# pages each, present the code chunks followed by the results.   
  
  
# INDEX  
# Libraries: please ensure every library loads, and otherwise install it via   
# install.packages("")  
# Data organization  
# Translation-dependent results  
# Translation-independent results  
# Modality  
# Sound-symbolism  
  
  
# \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ --- START --- \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
  
# Set your working directory here, to yield figures output:  
# setwd('C:/.../.../...')   
  
install.packages("gdata")

## Installing package into 'C:/Users/Pablo/Documents/R/win-library/3.3'  
## (as 'lib' is unspecified)

## package 'gdata' successfully unpacked and MD5 sums checked  
##   
## The downloaded binary packages are in  
## C:\Users\Pablo\AppData\Local\Temp\Rtmp4so1up\downloaded\_packages

install.packages("GPArotation")

## Installing package into 'C:/Users/Pablo/Documents/R/win-library/3.3'  
## (as 'lib' is unspecified)

## package 'GPArotation' successfully unpacked and MD5 sums checked  
##   
## The downloaded binary packages are in  
## C:\Users\Pablo\AppData\Local\Temp\Rtmp4so1up\downloaded\_packages

install.packages("psych")

## Installing package into 'C:/Users/Pablo/Documents/R/win-library/3.3'  
## (as 'lib' is unspecified)

## package 'psych' successfully unpacked and MD5 sums checked  
##   
## The downloaded binary packages are in  
## C:\Users\Pablo\AppData\Local\Temp\Rtmp4so1up\downloaded\_packages

install.packages("ggplot2")

## Installing package into 'C:/Users/Pablo/Documents/R/win-library/3.3'  
## (as 'lib' is unspecified)

## package 'ggplot2' successfully unpacked and MD5 sums checked  
##   
## The downloaded binary packages are in  
## C:\Users\Pablo\AppData\Local\Temp\Rtmp4so1up\downloaded\_packages

install.packages("car")

## Installing package into 'C:/Users/Pablo/Documents/R/win-library/3.3'  
## (as 'lib' is unspecified)

## package 'car' successfully unpacked and MD5 sums checked  
##   
## The downloaded binary packages are in  
## C:\Users\Pablo\AppData\Local\Temp\Rtmp4so1up\downloaded\_packages

install.packages("Rmisc")

## Installing package into 'C:/Users/Pablo/Documents/R/win-library/3.3'  
## (as 'lib' is unspecified)

## package 'Rmisc' successfully unpacked and MD5 sums checked  
##   
## The downloaded binary packages are in  
## C:\Users\Pablo\AppData\Local\Temp\Rtmp4so1up\downloaded\_packages

install.packages("corpcor")

## Installing package into 'C:/Users/Pablo/Documents/R/win-library/3.3'  
## (as 'lib' is unspecified)

## package 'corpcor' successfully unpacked and MD5 sums checked  
##   
## The downloaded binary packages are in  
## C:\Users\Pablo\AppData\Local\Temp\Rtmp4so1up\downloaded\_packages

install.packages('contrast')

## Installing package into 'C:/Users/Pablo/Documents/R/win-library/3.3'  
## (as 'lib' is unspecified)

## package 'contrast' successfully unpacked and MD5 sums checked  
##   
## The downloaded binary packages are in  
## C:\Users\Pablo\AppData\Local\Temp\Rtmp4so1up\downloaded\_packages

install.packages('doBy')

## Installing package into 'C:/Users/Pablo/Documents/R/win-library/3.3'  
## (as 'lib' is unspecified)

## package 'doBy' successfully unpacked and MD5 sums checked  
##   
## The downloaded binary packages are in  
## C:\Users\Pablo\AppData\Local\Temp\Rtmp4so1up\downloaded\_packages

install.packages('ltm')

## Installing package into 'C:/Users/Pablo/Documents/R/win-library/3.3'  
## (as 'lib' is unspecified)

## package 'ltm' successfully unpacked and MD5 sums checked  
##   
## The downloaded binary packages are in  
## C:\Users\Pablo\AppData\Local\Temp\Rtmp4so1up\downloaded\_packages

install.packages('MASS')

## Installing package into 'C:/Users/Pablo/Documents/R/win-library/3.3'  
## (as 'lib' is unspecified)

## package 'MASS' successfully unpacked and MD5 sums checked  
##   
## The downloaded binary packages are in  
## C:\Users\Pablo\AppData\Local\Temp\Rtmp4so1up\downloaded\_packages

install.packages('QuantPsyc')

## Installing package into 'C:/Users/Pablo/Documents/R/win-library/3.3'  
## (as 'lib' is unspecified)

## package 'QuantPsyc' successfully unpacked and MD5 sums checked  
##   
## The downloaded binary packages are in  
## C:\Users\Pablo\AppData\Local\Temp\Rtmp4so1up\downloaded\_packages

install.packages('qpcR')

## Installing package into 'C:/Users/Pablo/Documents/R/win-library/3.3'  
## (as 'lib' is unspecified)

## package 'qpcR' successfully unpacked and MD5 sums checked  
##   
## The downloaded binary packages are in  
## C:\Users\Pablo\AppData\Local\Temp\Rtmp4so1up\downloaded\_packages

install.packages('corpcor')

## Installing package into 'C:/Users/Pablo/Documents/R/win-library/3.3'  
## (as 'lib' is unspecified)

## package 'corpcor' successfully unpacked and MD5 sums checked  
##   
## The downloaded binary packages are in  
## C:\Users\Pablo\AppData\Local\Temp\Rtmp4so1up\downloaded\_packages

install.packages('lattice')

## Installing package into 'C:/Users/Pablo/Documents/R/win-library/3.3'  
## (as 'lib' is unspecified)

## package 'lattice' successfully unpacked and MD5 sums checked  
##   
## The downloaded binary packages are in  
## C:\Users\Pablo\AppData\Local\Temp\Rtmp4so1up\downloaded\_packages

install.packages('car')

## Installing package into 'C:/Users/Pablo/Documents/R/win-library/3.3'  
## (as 'lib' is unspecified)

## package 'car' successfully unpacked and MD5 sums checked  
##   
## The downloaded binary packages are in  
## C:\Users\Pablo\AppData\Local\Temp\Rtmp4so1up\downloaded\_packages

install.packages('pastecs')

## Installing package into 'C:/Users/Pablo/Documents/R/win-library/3.3'  
## (as 'lib' is unspecified)

## package 'pastecs' successfully unpacked and MD5 sums checked  
##   
## The downloaded binary packages are in  
## C:\Users\Pablo\AppData\Local\Temp\Rtmp4so1up\downloaded\_packages

install.packages('scales')

## Installing package into 'C:/Users/Pablo/Documents/R/win-library/3.3'  
## (as 'lib' is unspecified)

## package 'scales' successfully unpacked and MD5 sums checked  
##   
## The downloaded binary packages are in  
## C:\Users\Pablo\AppData\Local\Temp\Rtmp4so1up\downloaded\_packages

install.packages('reshape')

## Installing package into 'C:/Users/Pablo/Documents/R/win-library/3.3'  
## (as 'lib' is unspecified)

## package 'reshape' successfully unpacked and MD5 sums checked  
##   
## The downloaded binary packages are in  
## C:\Users\Pablo\AppData\Local\Temp\Rtmp4so1up\downloaded\_packages

install.packages('arules')

## Installing package into 'C:/Users/Pablo/Documents/R/win-library/3.3'  
## (as 'lib' is unspecified)

## package 'arules' successfully unpacked and MD5 sums checked  
##   
## The downloaded binary packages are in  
## C:\Users\Pablo\AppData\Local\Temp\Rtmp4so1up\downloaded\_packages

install.packages('plyr')

## Installing package into 'C:/Users/Pablo/Documents/R/win-library/3.3'  
## (as 'lib' is unspecified)

## package 'plyr' successfully unpacked and MD5 sums checked  
##   
## The downloaded binary packages are in  
## C:\Users\Pablo\AppData\Local\Temp\Rtmp4so1up\downloaded\_packages

install.packages('RColorBrewer')

## Installing package into 'C:/Users/Pablo/Documents/R/win-library/3.3'  
## (as 'lib' is unspecified)

## package 'RColorBrewer' successfully unpacked and MD5 sums checked  
##   
## The downloaded binary packages are in  
## C:\Users\Pablo\AppData\Local\Temp\Rtmp4so1up\downloaded\_packages

install.packages('dplyr')

## Installing package into 'C:/Users/Pablo/Documents/R/win-library/3.3'  
## (as 'lib' is unspecified)

## package 'dplyr' successfully unpacked and MD5 sums checked  
##   
## The downloaded binary packages are in  
## C:\Users\Pablo\AppData\Local\Temp\Rtmp4so1up\downloaded\_packages

install.packages('gdata')

## Installing package into 'C:/Users/Pablo/Documents/R/win-library/3.3'  
## (as 'lib' is unspecified)

## package 'gdata' successfully unpacked and MD5 sums checked  
##   
## The downloaded binary packages are in  
## C:\Users\Pablo\AppData\Local\Temp\Rtmp4so1up\downloaded\_packages

install.packages('gtools')

## Installing package into 'C:/Users/Pablo/Documents/R/win-library/3.3'  
## (as 'lib' is unspecified)

## package 'gtools' successfully unpacked and MD5 sums checked  
##   
## The downloaded binary packages are in  
## C:\Users\Pablo\AppData\Local\Temp\Rtmp4so1up\downloaded\_packages

install.packages('Hmisc')

## Installing package into 'C:/Users/Pablo/Documents/R/win-library/3.3'  
## (as 'lib' is unspecified)

## package 'Hmisc' successfully unpacked and MD5 sums checked  
##   
## The downloaded binary packages are in  
## C:\Users\Pablo\AppData\Local\Temp\Rtmp4so1up\downloaded\_packages

install.packages('png')

## Installing package into 'C:/Users/Pablo/Documents/R/win-library/3.3'  
## (as 'lib' is unspecified)

## package 'png' successfully unpacked and MD5 sums checked  
##   
## The downloaded binary packages are in  
## C:\Users\Pablo\AppData\Local\Temp\Rtmp4so1up\downloaded\_packages

library(ltm)

## Loading required package: MASS

## Loading required package: msm

## Loading required package: polycor

## Loading required package: mvtnorm

## Loading required package: sfsmisc

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(car)

##   
## Attaching package: 'car'

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

library(doBy)  
library(contrast)

## Loading required package: rms

## Loading required package: Hmisc

## Loading required package: survival

## Loading required package: Formula

## Loading required package: ggplot2

##   
## Attaching package: 'ggplot2'

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

##   
## Attaching package: 'Hmisc'

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

## The following object is masked from 'package:sfsmisc':  
##   
## errbar

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

## Loading required package: SparseM

##   
## Attaching package: 'SparseM'

## The following object is masked from 'package:base':  
##   
## backsolve

##   
## Attaching package: 'rms'

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

library(pastecs)

## Loading required package: boot

##   
## Attaching package: 'boot'

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

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

## 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

##   
## Attaching package: 'pastecs'

## The following object is masked from 'package:rms':  
##   
## specs

## The following object is masked from 'package:sfsmisc':  
##   
## last

library(scales)

##   
## Attaching package: 'scales'

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

library(ggplot2)  
library(psych)  
library(reshape)  
library(arules)

## Loading required package: Matrix

##   
## Attaching package: 'Matrix'

## The following object is masked from 'package:reshape':  
##   
## expand

##   
## Attaching package: 'arules'

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

## The following objects are masked from 'package:base':  
##   
## abbreviate, write

library(plyr)

##   
## Attaching package: 'plyr'

## The following objects are masked from 'package:reshape':  
##   
## rename, round\_any

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

library(RColorBrewer)  
library(Rmisc)  
library(corpcor)  
library(GPArotation)  
library(gdata)

## gdata: Unable to locate valid perl interpreter  
## gdata:   
## gdata: read.xls() will be unable to read Excel XLS and XLSX files  
## gdata: unless the 'perl=' argument is used to specify the location  
## gdata: of a valid perl intrpreter.  
## gdata:   
## gdata: (To avoid display of this message in the future, please  
## gdata: ensure perl is installed and available on the executable  
## gdata: search path.)

## gdata: Unable to load perl libaries needed by read.xls()  
## gdata: to support 'XLX' (Excel 97-2004) files.

##

## gdata: Unable to load perl libaries needed by read.xls()  
## gdata: to support 'XLSX' (Excel 2007+) files.

##

## gdata: Run the function 'installXLSXsupport()'  
## gdata: to automatically download and install the perl  
## gdata: libaries needed to support Excel XLS and XLSX formats.

##   
## Attaching package: 'gdata'

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

## The following object is masked from 'package:Hmisc':  
##   
## combine

## The following object is masked from 'package:sfsmisc':  
##   
## last

## The following object is masked from 'package:stats':  
##   
## nobs

## The following object is masked from 'package:utils':  
##   
## object.size

## The following object is masked from 'package:base':  
##   
## startsWith

library(QuantPsyc)

##   
## Attaching package: 'QuantPsyc'

## The following object is masked from 'package:Matrix':  
##   
## norm

## The following object is masked from 'package:SparseM':  
##   
## norm

## The following object is masked from 'package:base':  
##   
## norm

library(MASS)  
library(qpcR)

## Loading required package: minpack.lm

## Loading required package: rgl

## Loading required package: robustbase

##   
## Attaching package: 'robustbase'

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

## The following object is masked from 'package:survival':  
##   
## heart

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

library(dplyr)

##   
## Attaching package: 'dplyr'

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

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

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

## The following object is masked from 'package:reshape':  
##   
## rename

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

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

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

## The following object is masked from 'package:sfsmisc':  
##   
## 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

library(gtools)

##   
## Attaching package: 'gtools'

## The following objects are masked from 'package:boot':  
##   
## inv.logit, logit

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

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

library(Hmisc)  
library(png)  
  
  
# Calculate average percentange of unresponded items, i.e., unknown. Since there are   
# three ratings per word, and indeed the three were left blank whereever participants  
# ignored some word, the calculation includes a division by 3 (besides overall mean,  
# see specific percentage per file).  
  
file1 <- read.csv('file1\_gral.csv')  
file2 <- read.csv('file2\_gral.csv')  
file3 <- read.csv('file3\_gral.csv')  
file4 <- read.csv('file4\_gral.csv')  
file5 <- read.csv('file5\_gral.csv')  
file6 <- read.csv('file6\_gral.csv')  
  
(((100 \* (sum(is.na(file1)))) / (sum(!is.na(file1[,-1])) + sum(is.na(file1))) /3) +  
# 0.29  
  
((100 \* (sum(is.na(file2)))) / (sum(!is.na(file2[,-1])) + sum(is.na(file2))) /3) +  
# 1.42  
  
((100 \* (sum(is.na(file3)))) / (sum(!is.na(file3[,-1])) + sum(is.na(file3))) /3) +  
# 0.41  
  
# N.B. First participant is ignored because she completed only the first half of the   
# survey.  
((100 \* (sum(is.na(file4[,-c(1:4)])))) / (sum(!is.na(file4[,-c(1:4)])) +   
sum(is.na(file4[,-c(1:4)]))) /3) +  
# 2.85  
  
((100 \* (sum(is.na(file5)))) / (sum(!is.na(file5[,-1])) + sum(is.na(file5))) /3) +  
# 1.38  
  
((100 \* (sum(is.na(file6)))) / (sum(!is.na(file6[,-1])) + sum(is.na(file6))) /3)) /6

## [1] 1.308878

# 1.50  
# /6 = 1.31% = average unknown  
#\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
  
# Preprocessing:  
# There were 9 files with different items (mostly unrepeated) for concepts and   
# 10 files for properties. They were completed in different proportions, with an  
# average of eight participants per file.   
  
# RELIABILITY ANALYSIS: In putting together the ratings from each respondent, this   
# analysis allows to calculate the fit among those. In other words, is the mean   
# realistic or forced? Two measures are provided. First, interitem consistency   
# provides the fit among items independently of raters. Second, interrater   
# reliability measures the fit among raters, independently of items. A standard   
# minimum for both is alpha = .70.  
  
# Concepts  
all <- read.csv('all.csv')  
concs <- all[all$cat == 'conc',]  
  
# There were   
a\_concs<-concs[, c('a1', 'a2', 'a3', 'a4', 'a5', 'a6', 'a7', 'a8', 'a9')]   
psych::alpha(a\_concs)

##   
## Reliability analysis   
## Call: psych::alpha(x = a\_concs)  
##   
## raw\_alpha std.alpha G6(smc) average\_r S/N ase mean sd  
## 0.74 0.74 0.75 0.24 2.9 0.019 2 1  
##   
## lower alpha upper 95% confidence boundaries  
## 0.7 0.74 0.78   
##   
## Reliability if an item is dropped:  
## raw\_alpha std.alpha G6(smc) average\_r S/N alpha se  
## a1 0.72 0.72 0.73 0.24 2.5 0.021  
## a2 0.73 0.73 0.74 0.25 2.7 0.020  
## a3 0.73 0.73 0.74 0.26 2.7 0.020  
## a4 0.71 0.71 0.71 0.24 2.5 0.021  
## a5 0.71 0.71 0.72 0.24 2.5 0.021  
## a6 0.73 0.74 0.75 0.26 2.8 0.020  
## a7 0.70 0.70 0.71 0.23 2.4 0.022  
## a8 0.71 0.72 0.71 0.24 2.5 0.021  
## a9 0.70 0.70 0.71 0.23 2.4 0.022  
##   
## Item statistics   
## n raw.r std.r r.cor r.drop mean sd  
## a1 398 0.59 0.57 0.49 0.42 2.9 1.9  
## a2 408 0.51 0.52 0.41 0.35 2.0 1.6  
## a3 410 0.54 0.50 0.40 0.33 1.9 1.9  
## a4 409 0.61 0.60 0.54 0.45 2.4 1.6  
## a5 409 0.60 0.60 0.54 0.44 1.3 1.9  
## a6 407 0.47 0.48 0.36 0.31 2.2 1.9  
## a7 410 0.65 0.65 0.60 0.51 1.9 1.9  
## a8 269 0.56 0.59 0.54 0.43 1.4 1.6  
## a9 263 0.64 0.64 0.59 0.51 1.5 1.8

h\_concs<-concs[, c('h1', 'h2', 'h3', 'h4', 'h5', 'h6', 'h7', 'h8', 'h9')]  
psych::alpha(h\_concs)

##   
## Reliability analysis   
## Call: psych::alpha(x = h\_concs)  
##   
## raw\_alpha std.alpha G6(smc) average\_r S/N ase mean sd  
## 0.72 0.72 0.74 0.22 2.6 0.02 2 1  
##   
## lower alpha upper 95% confidence boundaries  
## 0.68 0.72 0.76   
##   
## Reliability if an item is dropped:  
## raw\_alpha std.alpha G6(smc) average\_r S/N alpha se  
## h1 0.70 0.70 0.72 0.23 2.3 0.022  
## h2 0.72 0.72 0.72 0.24 2.5 0.021  
## h3 0.71 0.70 0.71 0.23 2.4 0.021  
## h4 0.71 0.72 0.72 0.24 2.5 0.021  
## h5 0.68 0.68 0.70 0.21 2.1 0.024  
## h6 0.70 0.70 0.71 0.23 2.3 0.022  
## h7 0.67 0.67 0.68 0.20 2.0 0.025  
## h8 0.69 0.69 0.71 0.22 2.2 0.023  
## h9 0.69 0.69 0.69 0.22 2.2 0.022  
##   
## Item statistics   
## n raw.r std.r r.cor r.drop mean sd  
## h1 397 0.56 0.54 0.44 0.38 3.4 1.9  
## h2 408 0.43 0.46 0.36 0.27 1.8 1.7  
## h3 410 0.59 0.52 0.44 0.36 2.4 2.2  
## h4 409 0.46 0.46 0.36 0.30 2.5 1.6  
## h5 409 0.62 0.63 0.56 0.48 1.2 1.9  
## h6 407 0.53 0.54 0.45 0.38 1.8 1.7  
## h7 410 0.69 0.68 0.65 0.54 1.5 1.9  
## h8 269 0.61 0.60 0.52 0.44 1.7 1.7  
## h9 263 0.57 0.58 0.53 0.41 1.1 1.6

v\_concs<-concs[, c('v1', 'v2', 'v3', 'v4', 'v5', 'v6', 'v7', 'v8', 'v9')]   
psych::alpha(v\_concs)

##   
## Reliability analysis   
## Call: psych::alpha(x = v\_concs)  
##   
## raw\_alpha std.alpha G6(smc) average\_r S/N ase mean sd  
## 0.7 0.7 0.72 0.21 2.4 0.022 3.1 0.95  
##   
## lower alpha upper 95% confidence boundaries  
## 0.66 0.7 0.75   
##   
## Reliability if an item is dropped:  
## raw\_alpha std.alpha G6(smc) average\_r S/N alpha se  
## v1 0.69 0.70 0.71 0.22 2.3 0.023  
## v2 0.70 0.70 0.70 0.23 2.4 0.022  
## v3 0.66 0.66 0.67 0.19 1.9 0.025  
## v4 0.68 0.67 0.68 0.21 2.1 0.024  
## v5 0.67 0.67 0.69 0.20 2.0 0.024  
## v6 0.70 0.70 0.71 0.22 2.3 0.022  
## v7 0.66 0.65 0.65 0.19 1.9 0.025  
## v8 0.69 0.69 0.70 0.22 2.2 0.023  
## v9 0.65 0.65 0.66 0.19 1.8 0.026  
##   
## Item statistics   
## n raw.r std.r r.cor r.drop mean sd  
## v1 398 0.45 0.46 0.34 0.28 4.1 1.4  
## v2 408 0.40 0.43 0.32 0.25 2.5 1.6  
## v3 409 0.63 0.62 0.57 0.47 3.3 1.9  
## v4 409 0.55 0.56 0.49 0.39 3.2 1.5  
## v5 409 0.64 0.58 0.49 0.42 2.4 2.2  
## v6 407 0.47 0.45 0.32 0.27 3.6 1.7  
## v7 410 0.64 0.64 0.62 0.49 3.1 1.7  
## v8 269 0.47 0.48 0.38 0.31 2.5 1.7  
## v9 263 0.64 0.66 0.62 0.53 3.3 1.6

# RESULTS good. Lower than L&C, but they had more participants.  
# Interitem consistency  
# a: .74  
# h: .72  
# v: .70  
  
# Interrater reliability  
# a: .75  
# h: .74  
# v: .72  
  
  
# Properties  
props <- all[all$cat == 'prop',]  
  
a\_props<-props[, c('a1', 'a2', 'a3', 'a4', 'a5', 'a6', 'a7', 'a8', 'a9', 'a10')]  
psych::alpha(a\_props)

## Warning in psych::alpha(a\_props): Some items were negatively correlated  
## with the total scale and probably should be reversed. To do this, run the  
## function again with the 'check.keys=TRUE' option

## Some items ( a10 ) were negatively correlated with the total scale and probably should be reversed. To do this, run the function again with the 'check.keys=TRUE' option

##   
## Reliability analysis   
## Call: psych::alpha(x = a\_props)  
##   
## raw\_alpha std.alpha G6(smc) average\_r S/N ase mean sd  
## 0.78 0.78 0.89 0.27 3.6 0.016 1.7 1.3  
##   
## lower alpha upper 95% confidence boundaries  
## 0.75 0.78 0.81   
##   
## Reliability if an item is dropped:  
## raw\_alpha std.alpha G6(smc) average\_r S/N alpha se  
## a1 0.71 0.72 0.83 0.22 2.6 0.022  
## a2 0.74 0.75 0.87 0.25 2.9 0.019  
## a3 0.77 0.78 0.87 0.28 3.5 0.017  
## a4 0.72 0.73 0.85 0.23 2.6 0.021  
## a5 0.79 0.79 0.87 0.30 3.8 0.015  
## a6 0.72 0.72 0.85 0.22 2.6 0.021  
## a7 0.73 0.74 0.86 0.24 2.8 0.020  
## a8 0.73 0.73 0.86 0.23 2.7 0.020  
## a9 0.80 0.81 0.90 0.33 4.4 0.016  
## a10 0.85 0.84 0.90 0.37 5.4 0.011  
##   
## Item statistics   
## n raw.r std.r r.cor r.drop mean sd  
## a1 256 0.863 0.840 0.86 0.805 1.32 2.0  
## a2 305 0.755 0.713 0.67 0.638 1.61 1.7  
## a3 319 0.490 0.501 0.45 0.345 2.51 1.9  
## a4 320 0.844 0.829 0.83 0.760 1.35 1.9  
## a5 321 0.440 0.399 0.35 0.220 2.20 2.1  
## a6 214 0.882 0.855 0.86 0.800 1.36 1.7  
## a7 213 0.808 0.771 0.76 0.669 1.42 2.0  
## a8 217 0.824 0.803 0.79 0.727 2.00 1.7  
## a9 109 0.225 0.205 0.10 0.035 0.91 1.1  
## a10 109 -0.028 -0.087 -0.19 -0.258 2.23 2.1  
##   
## Non missing response frequency for each item  
## 0 0.5 1 2 3 4 5 miss  
## a1 0.64 0 0.05 0.04 0.05 0.05 0.16 0.25  
## a2 0.42 0 0.07 0.21 0.15 0.07 0.08 0.11  
## a3 0.25 0 0.08 0.15 0.14 0.18 0.20 0.07  
## a4 0.54 0 0.12 0.09 0.06 0.03 0.15 0.07  
## a5 0.43 0 0.02 0.08 0.12 0.11 0.25 0.06  
## a6 0.49 0 0.17 0.08 0.10 0.08 0.08 0.38  
## a7 0.64 0 0.03 0.02 0.07 0.06 0.18 0.38  
## a8 0.14 0 0.45 0.09 0.08 0.08 0.16 0.37  
## a9 0.53 0 0.14 0.25 0.06 0.01 0.01 0.68  
## a10 0.42 0 0.04 0.03 0.13 0.17 0.21 0.68

h\_props<-props[, c('h1', 'h2', 'h3', 'h4', 'h5', 'h6', 'h7', 'h8', 'h9', 'h10')]  
psych::alpha(h\_props)

## Warning in psych::alpha(h\_props): Some items were negatively correlated  
## with the total scale and probably should be reversed. To do this, run the  
## function again with the 'check.keys=TRUE' option

## Some items ( h10 ) were negatively correlated with the total scale and probably should be reversed. To do this, run the function again with the 'check.keys=TRUE' option

##   
## Reliability analysis   
## Call: psych::alpha(x = h\_props)  
##   
## raw\_alpha std.alpha G6(smc) average\_r S/N ase mean sd  
## 0.7 0.72 0.83 0.2 2.5 0.022 2 1.1  
##   
## lower alpha upper 95% confidence boundaries  
## 0.65 0.7 0.74   
##   
## Reliability if an item is dropped:  
## raw\_alpha std.alpha G6(smc) average\_r S/N alpha se  
## h1 0.60 0.65 0.76 0.17 1.8 0.029  
## h2 0.64 0.67 0.80 0.19 2.0 0.026  
## h3 0.70 0.72 0.82 0.22 2.6 0.022  
## h4 0.66 0.68 0.80 0.19 2.2 0.025  
## h5 0.69 0.71 0.81 0.21 2.5 0.023  
## h6 0.61 0.63 0.77 0.16 1.7 0.029  
## h7 0.60 0.64 0.79 0.16 1.8 0.031  
## h8 0.63 0.65 0.78 0.17 1.9 0.027  
## h9 0.70 0.73 0.83 0.23 2.7 0.022  
## h10 0.81 0.81 0.87 0.32 4.2 0.014  
##   
## Item statistics   
## n raw.r std.r r.cor r.drop mean sd  
## h1 256 0.82 0.77 0.79 0.69 2.28 2.0  
## h2 306 0.67 0.65 0.62 0.54 1.62 1.7  
## h3 319 0.36 0.38 0.30 0.21 2.63 1.6  
## h4 320 0.61 0.60 0.54 0.44 1.48 1.9  
## h5 321 0.50 0.45 0.40 0.27 2.10 2.1  
## h6 214 0.79 0.82 0.83 0.73 1.78 1.7  
## h7 213 0.82 0.80 0.80 0.71 2.21 2.1  
## h8 217 0.71 0.75 0.75 0.62 2.54 1.6  
## h9 109 0.37 0.32 0.22 0.13 0.96 1.0  
## h10 109 -0.15 -0.24 -0.39 -0.40 1.72 2.2  
##   
## Non missing response frequency for each item  
## 0 1 1.5 2 3 3.5 4 5 miss  
## h1 0.36 0.05 0 0.09 0.19 0 0.11 0.21 0.25  
## h2 0.42 0.09 0 0.16 0.16 0 0.11 0.06 0.11  
## h3 0.15 0.11 0 0.20 0.21 0 0.18 0.16 0.07  
## h4 0.52 0.09 0 0.09 0.10 0 0.05 0.14 0.07  
## h5 0.41 0.05 0 0.11 0.11 0 0.08 0.23 0.06  
## h6 0.38 0.08 0 0.12 0.27 0 0.08 0.07 0.38  
## h7 0.43 0.03 0 0.05 0.12 0 0.17 0.21 0.38  
## h8 0.08 0.29 0 0.11 0.17 0 0.20 0.15 0.37  
## h9 0.36 0.45 0 0.10 0.07 0 0.00 0.02 0.68  
## h10 0.61 0.00 0 0.00 0.06 0 0.09 0.24 0.68

v\_props<-props[, c('v1', 'v2', 'v3', 'v4', 'v5', 'v6', 'v7', 'v8', 'v9', 'v10')]  
psych::alpha(v\_props)

##   
## Reliability analysis   
## Call: psych::alpha(x = v\_props)  
##   
## raw\_alpha std.alpha G6(smc) average\_r S/N ase mean sd  
## 0.85 0.85 0.87 0.36 5.5 0.012 3.2 1.2  
##   
## lower alpha upper 95% confidence boundaries  
## 0.82 0.85 0.87   
##   
## Reliability if an item is dropped:  
## raw\_alpha std.alpha G6(smc) average\_r S/N alpha se  
## v1 0.83 0.83 0.86 0.34 4.7 0.013  
## v2 0.84 0.83 0.86 0.36 5.0 0.013  
## v3 0.84 0.84 0.87 0.37 5.2 0.013  
## v4 0.82 0.82 0.85 0.34 4.5 0.014  
## v5 0.85 0.85 0.87 0.38 5.5 0.012  
## v6 0.81 0.81 0.84 0.32 4.3 0.015  
## v7 0.82 0.82 0.85 0.33 4.4 0.015  
## v8 0.82 0.82 0.84 0.33 4.5 0.014  
## v9 0.86 0.87 0.89 0.42 6.4 0.011  
## v10 0.84 0.84 0.85 0.37 5.3 0.012  
##   
## Item statistics   
## n raw.r std.r r.cor r.drop mean sd  
## v1 256 0.82 0.71 0.67 0.61 3.6 1.7  
## v2 305 0.72 0.63 0.59 0.52 2.9 1.5  
## v3 319 0.59 0.59 0.52 0.47 3.5 1.4  
## v4 320 0.76 0.76 0.73 0.68 3.1 1.9  
## v5 321 0.53 0.52 0.43 0.39 3.1 1.7  
## v6 214 0.83 0.84 0.84 0.78 3.1 1.5  
## v7 213 0.80 0.79 0.77 0.72 3.6 1.8  
## v8 217 0.73 0.78 0.78 0.72 3.7 1.4  
## v9 109 0.30 0.31 0.18 0.16 0.9 1.3  
## v10 109 0.67 0.57 0.54 0.45 3.8 1.9  
##   
## Non missing response frequency for each item  
## 0 1 2 3 3.5 4 4.333333333 4.5 5 miss  
## v1 0.14 0.02 0.05 0.15 0 0.20 0 0 0.44 0.25  
## v2 0.09 0.08 0.22 0.22 0 0.27 0 0 0.13 0.11  
## v3 0.04 0.06 0.15 0.17 0 0.24 0 0 0.34 0.07  
## v4 0.18 0.05 0.11 0.18 0 0.11 0 0 0.36 0.07  
## v5 0.12 0.06 0.14 0.20 0 0.17 0 0 0.30 0.06  
## v6 0.05 0.14 0.10 0.29 0 0.21 0 0 0.22 0.38  
## v7 0.15 0.02 0.04 0.14 0 0.15 0 0 0.50 0.38  
## v8 0.00 0.12 0.07 0.12 0 0.35 0 0 0.33 0.37  
## v9 0.58 0.17 0.07 0.15 0 0.00 0 0 0.03 0.68  
## v10 0.17 0.01 0.02 0.06 0 0.14 0 0 0.61 0.68

# RESULTS: good. Lower than L&C, but they did have a few more participants.  
# Interitem consistency  
# a: .78  
# h: .70  
# v: .85  
  
# Interrater reliability  
# a: .89  
# h: .83  
# v: .87  
# \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
  
  
# TRANSLATION-DEPENDENT RESULTS  
  
# Read in the general norms file, 'all' (all seemed right, seeing as we have Dutch   
# and English properties and concepts)  
  
all <- read.csv('all.csv')  
  
# PROPERTIES  
props <- all[all$cat=='prop',]  
nrow(props) # 366 Dutch + a few from Lynott&Connell for comparisons

## [1] 343

# CONCEPTS   
concs <- all[all$cat=='conc',]  
nrow(concs) # 411 Dutch + a few from Lynott&Connell for comparison

## [1] 416

# Correlations  
  
# PROPERTIES  
# Modalities  
rcor.test(props[, c('Auditory', 'Aud\_eng')], use = 'complete.obs')

##   
## Auditory Aud\_eng  
## Auditory \*\*\*\*\* 0.795   
## Aud\_eng <0.001 \*\*\*\*\*   
##   
## upper diagonal part contains correlation coefficient estimates   
## lower diagonal part contains corresponding p-values

rcor.test(props[, c('Haptic', 'Hap\_eng')], use = 'complete.obs')

##   
## Haptic Hap\_eng  
## Haptic \*\*\*\*\* 0.690   
## Hap\_eng <0.001 \*\*\*\*\*   
##   
## upper diagonal part contains correlation coefficient estimates   
## lower diagonal part contains corresponding p-values

rcor.test(props[, c('Visual', 'Vis\_eng')], use = 'complete.obs')

##   
## Visual Vis\_eng  
## Visual \*\*\*\*\* 0.711   
## Vis\_eng <0.001 \*\*\*\*\*   
##   
## upper diagonal part contains correlation coefficient estimates   
## lower diagonal part contains corresponding p-values

# Significant, large correlations ranging from .69 to .80  
  
# Exclusivity   
rcor.test(props[, c('Exclusivity', 'exc\_eng')], use = 'complete.obs')

##   
## Exclusivity exc\_eng  
## Exclusivity \*\*\*\*\* 0.475   
## exc\_eng <0.001 \*\*\*\*\*   
##   
## upper diagonal part contains correlation coefficient estimates   
## lower diagonal part contains corresponding p-values

# Medium-sized corr Eng-Dutch  
  
  
# CONCEPTS  
# Modalities  
rcor.test(concs[, c('Auditory', 'Aud\_eng')], use = 'complete.obs')

##   
## Auditory Aud\_eng  
## Auditory \*\*\*\*\* 0.683   
## Aud\_eng <0.001 \*\*\*\*\*   
##   
## upper diagonal part contains correlation coefficient estimates   
## lower diagonal part contains corresponding p-values

rcor.test(concs[, c('Haptic', 'Hap\_eng')], use = 'complete.obs')

##   
## Haptic Hap\_eng  
## Haptic \*\*\*\*\* 0.624   
## Hap\_eng <0.001 \*\*\*\*\*   
##   
## upper diagonal part contains correlation coefficient estimates   
## lower diagonal part contains corresponding p-values

rcor.test(concs[, c('Visual', 'Vis\_eng')], use = 'complete.obs')

##   
## Visual Vis\_eng  
## Visual \*\*\*\*\* 0.659   
## Vis\_eng <0.001 \*\*\*\*\*   
##   
## upper diagonal part contains correlation coefficient estimates   
## lower diagonal part contains corresponding p-values

# Significant, large correlations ranging from .63 to .69  
  
# Exclusivity   
rcor.test(concs[, c('Exclusivity', 'exc\_eng')], use = 'complete.obs')

##   
## Exclusivity exc\_eng  
## Exclusivity \*\*\*\*\* 0.428   
## exc\_eng <0.001 \*\*\*\*\*   
##   
## upper diagonal part contains correlation coefficient estimates   
## lower diagonal part contains corresponding p-values

# Medium-sized corr Eng-Dutch  
  
  
# Descriptives: M, SD, SE...  
# English  
psych::describe(props$Aud\_eng)

## vars n mean sd median trimmed mad min max range skew kurtosis se  
## X1 1 343 1.73 1.67 1.05 1.54 1.2 0 5 5 0.86 -0.77 0.09

psych::describe(props$Hap\_eng)

## vars n mean sd median trimmed mad min max range skew kurtosis  
## X1 1 343 2.41 1.62 2.52 2.41 2.26 0 4.95 4.95 -0.07 -1.48  
## se  
## X1 0.09

psych::describe(props$Vis\_eng)

## vars n mean sd median trimmed mad min max range skew kurtosis  
## X1 1 343 3.8 1.06 4.17 3.94 0.81 0.52 5 4.48 -1.07 0.16  
## se  
## X1 0.06

psych::describe(concs$Aud\_eng)

## vars n mean sd median trimmed mad min max range skew kurtosis se  
## X1 1 392 2.16 1.09 2.06 2.12 1.13 0 5 5 0.34 -0.49 0.06

psych::describe(concs$Hap\_eng)

## vars n mean sd median trimmed mad min max range skew kurtosis  
## X1 1 392 1.86 1.13 1.65 1.78 1.16 0 4.76 4.76 0.57 -0.63  
## se  
## X1 0.06

psych::describe(concs$Vis\_eng)

## vars n mean sd median trimmed mad min max range skew kurtosis se  
## X1 1 392 3.55 0.8 3.65 3.61 0.87 0.76 5 4.24 -0.6 0.01 0.04

stat.desc(props$Aud\_eng)

## nbr.val nbr.null nbr.na min max   
## 343.0000000 7.0000000 0.0000000 0.0000000 5.0000000   
## range sum median mean SE.mean   
## 5.0000000 592.3040000 1.0480000 1.7268338 0.0900376   
## CI.mean.0.95 var std.dev coef.var   
## 0.1770972 2.7806219 1.6675197 0.9656515

stat.desc(props$Hap\_eng)

## nbr.val nbr.null nbr.na min max   
## 343.00000000 2.00000000 0.00000000 0.00000000 4.95200000   
## range sum median mean SE.mean   
## 4.95200000 828.33800000 2.52400000 2.41497959 0.08728861   
## CI.mean.0.95 var std.dev coef.var   
## 0.17169012 2.61342058 1.61660774 0.66940845

stat.desc(props$Vis\_eng)

## nbr.val nbr.null nbr.na min max   
## 3.430000e+02 0.000000e+00 0.000000e+00 5.240000e-01 5.000000e+00   
## range sum median mean SE.mean   
## 4.476000e+00 1.301717e+03 4.167000e+00 3.795093e+00 5.737112e-02   
## CI.mean.0.95 var std.dev coef.var   
## 1.128447e-01 1.128966e+00 1.062528e+00 2.799741e-01

stat.desc(concs$Aud\_eng)

## nbr.val nbr.null nbr.na min max   
## 392.00000000 1.00000000 24.00000000 0.00000000 5.00000000   
## range sum median mean SE.mean   
## 5.00000000 848.38600000 2.05900000 2.16425000 0.05501011   
## CI.mean.0.95 var std.dev coef.var   
## 0.10815262 1.18623618 1.08914470 0.50324348

stat.desc(concs$Hap\_eng)

## nbr.val nbr.null nbr.na min max   
## 392.0000000 1.0000000 24.0000000 0.0000000 4.7650000   
## range sum median mean SE.mean   
## 4.7650000 730.5720000 1.6470000 1.8637041 0.0569453   
## CI.mean.0.95 var std.dev coef.var   
## 0.1119573 1.2711649 1.1274595 0.6049563

stat.desc(concs$Vis\_eng)

## nbr.val nbr.null nbr.na min max   
## 3.920000e+02 0.000000e+00 2.400000e+01 7.650000e-01 5.000000e+00   
## range sum median mean SE.mean   
## 4.235000e+00 1.393415e+03 3.647000e+00 3.554630e+00 4.053360e-02   
## CI.mean.0.95 var std.dev coef.var   
## 7.969106e-02 6.440452e-01 8.025243e-01 2.257687e-01

# Dutch  
psych::describe(props$Auditory)

## vars n mean sd median trimmed mad min max range skew kurtosis se  
## X1 1 336 1.74 1.29 1.33 1.55 0.96 0 5 5 1.14 0.26 0.07

psych::describe(props$Haptic)

## vars n mean sd median trimmed mad min max range skew kurtosis  
## X1 1 336 1.96 1.12 1.75 1.88 1.11 0 4.75 4.75 0.54 -0.58  
## se  
## X1 0.06

psych::describe(props$Visual)

## vars n mean sd median trimmed mad min max range skew kurtosis  
## X1 1 336 3.22 1.15 3.4 3.31 1.26 0.5 5 4.5 -0.53 -0.68  
## se  
## X1 0.06

psych::describe(concs$Auditory)

## vars n mean sd median trimmed mad min max range skew kurtosis  
## X1 1 411 1.97 1.03 1.86 1.9 1.04 0 4.89 4.89 0.59 0.16  
## se  
## X1 0.05

psych::describe(concs$Haptic)

## vars n mean sd median trimmed mad min max range skew kurtosis  
## X1 1 411 1.96 1.04 1.78 1.87 0.99 0 4.78 4.78 0.7 -0.25  
## se  
## X1 0.05

psych::describe(concs$Visual)

## vars n mean sd median trimmed mad min max range skew kurtosis  
## X1 1 411 3.13 0.95 3.22 3.16 0.99 0.33 5 4.67 -0.3 -0.49  
## se  
## X1 0.05

stat.desc(props$Auditory)

## nbr.val nbr.null nbr.na min max   
## 336.00000000 3.00000000 7.00000000 0.00000000 5.00000000   
## range sum median mean SE.mean   
## 5.00000000 584.54500000 1.33300000 1.73971726 0.07061977   
## CI.mean.0.95 var std.dev coef.var   
## 0.13891408 1.67568307 1.29448178 0.74407595

stat.desc(props$Haptic)

## nbr.val nbr.null nbr.na min max   
## 336.00000000 1.00000000 7.00000000 0.00000000 4.75000000   
## range sum median mean SE.mean   
## 4.75000000 657.73300000 1.75000000 1.95753869 0.06113725   
## CI.mean.0.95 var std.dev coef.var   
## 0.12026128 1.25588834 1.12066424 0.57248638

stat.desc(props$Visual)

## nbr.val nbr.null nbr.na min max   
## 3.360000e+02 0.000000e+00 7.000000e+00 5.000000e-01 5.000000e+00   
## range sum median mean SE.mean   
## 4.500000e+00 1.083009e+03 3.400000e+00 3.223241e+00 6.296073e-02   
## CI.mean.0.95 var std.dev coef.var   
## 1.238482e-01 1.331922e+00 1.154089e+00 3.580524e-01

stat.desc(concs$Auditory)

## nbr.val nbr.null nbr.na min max   
## 411.00000000 1.00000000 5.00000000 0.00000000 4.88900000   
## range sum median mean SE.mean   
## 4.88900000 808.11100000 1.85700000 1.96620681 0.05082869   
## CI.mean.0.95 var std.dev coef.var   
## 0.09991736 1.06184159 1.03045698 0.52408372

stat.desc(concs$Haptic)

## nbr.val nbr.null nbr.na min max   
## 411.00000000 2.00000000 5.00000000 0.00000000 4.77800000   
## range sum median mean SE.mean   
## 4.77800000 807.03000000 1.77800000 1.96357664 0.05141212   
## CI.mean.0.95 var std.dev coef.var   
## 0.10106424 1.08635771 1.04228485 0.53080935

stat.desc(concs$Visual)

## nbr.val nbr.null nbr.na min max   
## 4.110000e+02 0.000000e+00 5.000000e+00 3.330000e-01 5.000000e+00   
## range sum median mean SE.mean   
## 4.667000e+00 1.285371e+03 3.222000e+00 3.127423e+00 4.695101e-02   
## CI.mean.0.95 var std.dev coef.var   
## 9.229475e-02 9.060075e-01 9.518442e-01 3.043541e-01

# Sample sizes for English and Dutch  
nrow(props[!is.na(props$exc\_eng),]) # total items w/ English norms = 343

## [1] 343

nrow(props[props$main\_eng=='a' & !is.na(props$exc\_eng),])

## [1] 68

nrow(props[props$main\_eng=='h' & !is.na(props$exc\_eng),])

## [1] 70

nrow(props[props$main\_eng=='v' & !is.na(props$exc\_eng),])

## [1] 205

nrow(props[!is.na(concs$exc\_eng),]) # total items w/ English norms = 392

## [1] 392

nrow(props[concs$main\_eng=='a' & !is.na(concs$exc\_eng),])

## [1] 42

nrow(props[concs$main\_eng=='h' & !is.na(concs$exc\_eng),])

## [1] 14

nrow(props[concs$main\_eng=='v' & !is.na(concs$exc\_eng),])

## [1] 336

nrow(props[!is.na(props$Exclusivity),]) # total props w/ Dutch norms = 336

## [1] 336

nrow(props[props$main=='a' & !is.na(props$Exclusivity),])

## [1] 64

nrow(props[props$main=='h' & !is.na(props$Exclusivity),])

## [1] 45

nrow(props[props$main=='v' & !is.na(props$Exclusivity),])

## [1] 227

nrow(props[!is.na(concs$Exclusivity),]) # total props w/ Dutch norms = 411

## [1] 411

nrow(props[concs$main=='a' & !is.na(concs$Exclusivity),])

## [1] 48

nrow(props[concs$main=='h' & !is.na(concs$Exclusivity),])

## [1] 45

nrow(props[concs$main=='v' & !is.na(concs$Exclusivity),])

## [1] 318

# \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
  
  
# CRITICAL RESULTS: not translation-influenced  
  
# Relation between modality strength, dominant modalities, and mod exclusivitY  
# ENGLISH  
# properties  
summaryBy(Aud\_eng ~ main\_eng, data=props, FUN=mean)

## main\_eng Aud\_eng.mean  
## 1 a 4.5887941  
## 2 h 1.1219429  
## 3 v 0.9840488

summaryBy(Hap\_eng ~ main\_eng, data=props, FUN=mean)

## main\_eng Hap\_eng.mean  
## 1 a 0.7042941  
## 2 h 4.3319143  
## 3 v 2.3278634

summaryBy(Vis\_eng ~ main\_eng, data=props, FUN=mean)

## main\_eng Vis\_eng.mean  
## 1 a 2.305074  
## 2 h 3.447314  
## 3 v 4.408098

summaryBy(exc\_eng ~ main\_eng, data=props, FUN=mean)

## main\_eng exc\_eng.mean  
## 1 a 0.5739265  
## 2 h 0.3701571  
## 3 v 0.4891659

# concepts  
summaryBy(Aud\_eng ~ main\_eng, data=concs, FUN=mean)

## main\_eng Aud\_eng.mean  
## 1 a 3.542810  
## 2 h 1.347643  
## 3 v 2.025955  
## 4 <NA> NA

summaryBy(Hap\_eng ~ main\_eng, data=concs, FUN=mean)

## main\_eng Hap\_eng.mean  
## 1 a 1.032119  
## 2 h 4.143143  
## 3 v 1.872676  
## 4 <NA> NA

summaryBy(Vis\_eng ~ main\_eng, data=concs, FUN=mean)

## main\_eng Vis\_eng.mean  
## 1 a 2.711643  
## 2 h 3.428714  
## 3 v 3.665250  
## 4 <NA> NA

summaryBy(exc\_eng ~ main\_eng, data=concs, FUN=mean)

## main\_eng exc\_eng.mean  
## 1 a 0.4413571  
## 2 h 0.3530714  
## 3 v 0.3917173  
## 4 <NA> NA

# DUTCH   
# properties  
summaryBy(Auditory ~ main, data=props, FUN=mean)

## main Auditory.mean  
## 1 a 3.816750  
## 2 h 1.374711  
## 3 v 1.226480  
## 4 <NA> NA

summaryBy(Haptic ~ main, data=props, FUN=mean)

## main Haptic.mean  
## 1 a 1.220469  
## 2 h 3.545978  
## 3 v 1.850458  
## 4 <NA> NA

summaryBy(Visual ~ main, data=props, FUN=mean)

## main Visual.mean  
## 1 a 1.704125  
## 2 h 2.722356  
## 3 v 3.750833  
## 4 <NA> NA

summaryBy(Exclusivity ~ main, data=props, FUN=mean)

## main Exclusivity.mean  
## 1 a 0.4284531  
## 2 h 0.2922667  
## 3 v 0.4124714  
## 4 <NA> NA

# concepts  
summaryBy(Auditory ~ main, data=concs, FUN=mean)

## main Auditory.mean  
## 1 a 3.447500  
## 2 h 1.520956  
## 3 v 1.805623  
## 4 <NA> NA

summaryBy(Haptic ~ main, data=concs, FUN=mean)

## main Haptic.mean  
## 1 a 1.498063  
## 2 h 3.341511  
## 3 v 1.838852  
## 4 <NA> NA

summaryBy(Visual ~ main, data=concs, FUN=mean)

## main Visual.mean  
## 1 a 2.382125  
## 2 h 2.721467  
## 3 v 3.297368  
## 4 <NA> NA

summaryBy(Exclusivity ~ main, data=concs, FUN=mean)

## main Exclusivity.mean  
## 1 a 0.2806875  
## 2 h 0.2571778  
## 3 v 0.2905597  
## 4 <NA> NA

# RESULTS: both languages strongly related on individual modalities and on   
# exclusivity. Correlations among modalities replicate previous norms, with visual   
# and haptic items related, and auditory ones independent.  
  
# Yet, there is clearly a greater exclusivity in the English norms.  
# Properties  
psych::describe(props$exc\_eng) # M = 0.48

## vars n mean sd median trimmed mad min max range skew kurtosis  
## X1 1 343 0.48 0.17 0.46 0.47 0.16 0.1 0.98 0.88 0.46 -0.11  
## se  
## X1 0.01

psych::describe(props$Exclusivity) # M = 0.40

## vars n mean sd median trimmed mad min max range skew kurtosis se  
## X1 1 336 0.4 0.18 0.38 0.39 0.18 0 1 1 0.42 -0.22 0.01

# Concepts  
psych::describe(concs$exc\_eng) # M = 0.40

## vars n mean sd median trimmed mad min max range skew kurtosis  
## X1 1 392 0.4 0.12 0.39 0.39 0.1 0.07 0.84 0.77 0.2 0.8  
## se  
## X1 0.01

psych::describe(concs$Exclusivity) # M = 0.29

## vars n mean sd median trimmed mad min max range skew kurtosis  
## X1 1 411 0.29 0.15 0.27 0.28 0.15 0.01 0.91 0.9 0.62 0.38  
## se  
## X1 0.01

# Indeed lower exclusivity and higher SD for Dutch items >> Check significance  
# Because the English and the Dutch norms are paired, the difference has to be   
# checked through a one-sample t-test, checking the mean of one language against   
# the other language (see further below).  
  
  
  
# Correlations among modalities within each category and language:  
  
# ENGLISH  
# PROPERTIES  
rcor.test(props[, c('Aud\_eng', 'Hap\_eng', 'Vis\_eng', 'exc\_eng')], use =   
'complete.obs')

##   
## Aud\_eng Hap\_eng Vis\_eng exc\_eng  
## Aud\_eng \*\*\*\*\* -0.417 -0.625 0.018   
## Hap\_eng <0.001 \*\*\*\*\* 0.234 -0.621   
## Vis\_eng <0.001 <0.001 \*\*\*\*\* -0.053   
## exc\_eng 0.740 <0.001 0.331 \*\*\*\*\*   
##   
## upper diagonal part contains correlation coefficient estimates   
## lower diagonal part contains corresponding p-values

corr3 = rcor.test(props[, c('Aud\_eng', 'Hap\_eng', 'Vis\_eng', 'exc\_eng')],   
use = 'complete.obs')  
write.csv(corr3$cor.mat, file = "corr3.csv",na="") # saved for manuscript  
  
# CONCEPTS  
rcor.test(concs[, c('Aud\_eng', 'Hap\_eng', 'Vis\_eng', 'exc\_eng')], use = 'complete.obs')

##   
## Aud\_eng Hap\_eng Vis\_eng exc\_eng  
## Aud\_eng \*\*\*\*\* -0.176 -0.008 -0.276   
## Hap\_eng <0.001 \*\*\*\*\* 0.554 -0.393   
## Vis\_eng 0.868 <0.001 \*\*\*\*\* -0.065   
## exc\_eng <0.001 <0.001 0.202 \*\*\*\*\*   
##   
## upper diagonal part contains correlation coefficient estimates   
## lower diagonal part contains corresponding p-values

corr4 = rcor.test(concs[, c('Aud\_eng', 'Hap\_eng', 'Vis\_eng', 'exc\_eng')], use =   
'complete.obs')  
write.csv(corr4$cor.mat, file = "corr4.csv",na="") # saved for manuscript  
  
  
# DUTCH  
# PROPERTIES  
rcor.test(props[, c('Auditory', 'Haptic', 'Visual', 'Exclusivity')], use =   
'complete.obs')

##   
## Auditory Haptic Visual Exclusivity  
## Auditory \*\*\*\*\* -0.228 -0.513 -0.173   
## Haptic <0.001 \*\*\*\*\* 0.193 -0.482   
## Visual <0.001 <0.001 \*\*\*\*\* 0.162   
## Exclusivity 0.001 <0.001 0.003 \*\*\*\*\*   
##   
## upper diagonal part contains correlation coefficient estimates   
## lower diagonal part contains corresponding p-values

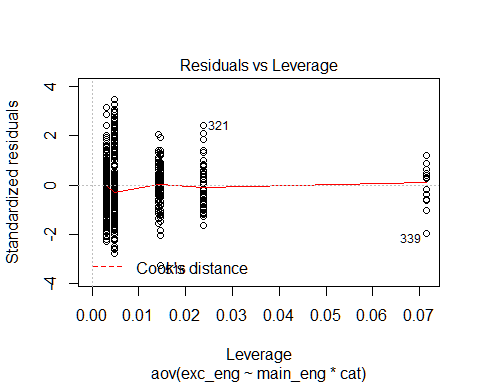
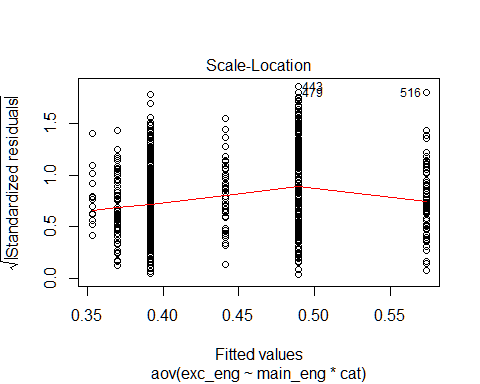
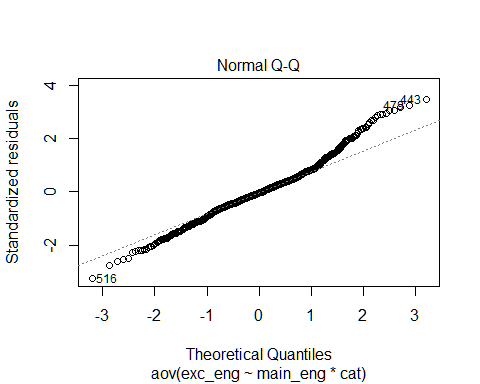
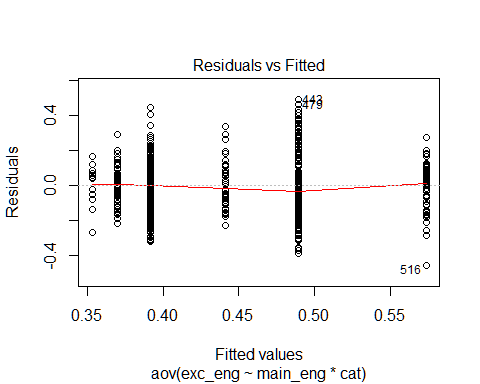
corr1 = rcor.test(props[, c('Auditory', 'Haptic', 'Visual', 'Exclusivity')],   
use = 'complete.obs')  
write.csv(corr1$cor.mat, file = "corr1.csv",na="") # saved for manuscript  
  
# CONCEPTS  
rcor.test(concs[, c('Auditory', 'Haptic', 'Visual', 'Exclusivity')], use =   
'complete.obs')

##   
## Auditory Haptic Visual Exclusivity  
## Auditory \*\*\*\*\* -0.009 0.085 -0.410   
## Haptic 0.863 \*\*\*\*\* 0.441 -0.317   
## Visual 0.086 <0.001 \*\*\*\*\* 0.122   
## Exclusivity <0.001 <0.001 0.013 \*\*\*\*\*   
##   
## upper diagonal part contains correlation coefficient estimates   
## lower diagonal part contains corresponding p-values

corr2 = rcor.test(concs[, c('Auditory', 'Haptic', 'Visual', 'Exclusivity')], use =   
'complete.obs')  
write.csv(corr2$cor.mat, file = "corr2.csv",na="") # saved for manuscript  
  
  
# Statistical tests for those differences  
# Yet the same again, but now with a statistical significance test  
  
# ENGLISH  
# Setting contrasts based on means  
contrasts(all$main\_eng) <- cbind(c(2,0,-2), c(-1,2,-1))  
# (1) Aud vs Vis; (2) Hap vs Aud-&-Vis  
contrasts(all$main\_eng)

## [,1] [,2]  
## a 2 -1  
## h 0 2  
## v -2 -1

fitt <- aov(exc\_eng ~ main\_eng \* cat, data=all)  
plot(fitt)



summary(fitt)

## Df Sum Sq Mean Sq F value Pr(>F)   
## main\_eng 2 1.264 0.6321 31.507 7.52e-14 \*\*\*  
## cat 1 1.561 1.5614 77.832 < 2e-16 \*\*\*  
## main\_eng:cat 2 0.107 0.0537 2.676 0.0695 .   
## Residuals 729 14.625 0.0201   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
## 24 observations deleted due to missingness

drop1(fitt,~.,test="F")

## Single term deletions  
##   
## Model:  
## exc\_eng ~ main\_eng \* cat  
## Df Sum of Sq RSS AIC F value Pr(>F)   
## <none> 14.624 -2867.1   
## main\_eng 2 0.11832 14.743 -2865.2 2.949 0.05302 .   
## cat 1 0.46228 15.087 -2846.2 23.044 1.922e-06 \*\*\*  
## main\_eng:cat 2 0.10737 14.732 -2865.8 2.676 0.06951 .   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Anova(fitt)

## Anova Table (Type II tests)  
##   
## Response: exc\_eng  
## Sum Sq Df F value Pr(>F)   
## main\_eng 1.4717 2 36.681 6.628e-16 \*\*\*  
## cat 1.5614 1 77.832 < 2.2e-16 \*\*\*  
## main\_eng:cat 0.1074 2 2.676 0.06951 .   
## Residuals 14.6245 729   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Anova(fitt, type = "II")

## Anova Table (Type II tests)  
##   
## Response: exc\_eng  
## Sum Sq Df F value Pr(>F)   
## main\_eng 1.4717 2 36.681 6.628e-16 \*\*\*  
## cat 1.5614 1 77.832 < 2.2e-16 \*\*\*  
## main\_eng:cat 0.1074 2 2.676 0.06951 .   
## Residuals 14.6245 729   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Anova(fitt, type = "III")

## Anova Table (Type III tests)  
##   
## Response: exc\_eng  
## Sum Sq Df F value Pr(>F)   
## (Intercept) 14.3252 1 714.082 < 2.2e-16 \*\*\*  
## main\_eng 0.1183 2 2.949 0.05302 .   
## cat 0.4623 1 23.044 1.922e-06 \*\*\*  
## main\_eng:cat 0.1074 2 2.676 0.06951 .   
## Residuals 14.6245 729   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

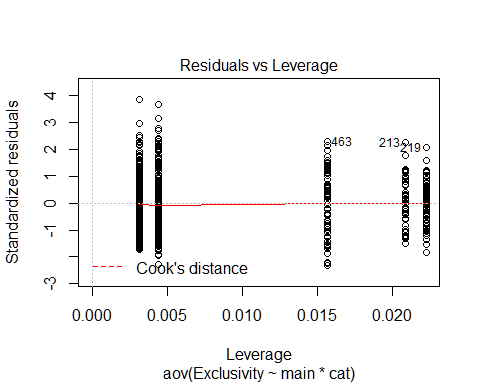
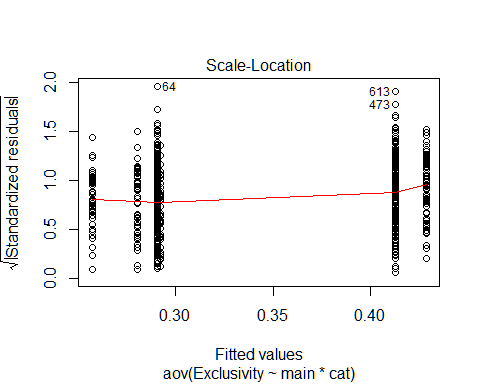
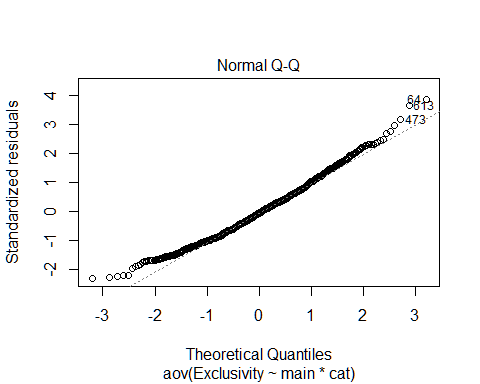
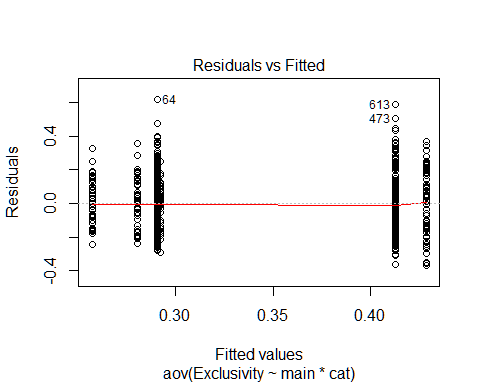
summary.lm(fitt)

##   
## Call:  
## aov(formula = exc\_eng ~ main\_eng \* cat, data = all)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.45793 -0.08089 -0.00472 0.06968 0.49083   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.395382 0.014796 26.722 < 2e-16 \*\*\*  
## main\_eng1 0.012410 0.005795 2.141 0.0326 \*   
## main\_eng2 -0.021155 0.013196 -1.603 0.1093   
## catprop 0.082368 0.017159 4.800 1.92e-06 \*\*\*  
## main\_eng1:catprop 0.008780 0.007625 1.152 0.2499   
## main\_eng2:catprop -0.032641 0.014727 -2.216 0.0270 \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.1416 on 729 degrees of freedom  
## (24 observations deleted due to missingness)  
## Multiple R-squared: 0.167, Adjusted R-squared: 0.1613   
## F-statistic: 29.24 on 5 and 729 DF, p-value: < 2.2e-16

# RESULTS: English properties with more exclusivity than concepts(\*\*\*)  
# Contrasts: (1) Aud vs Vis (\*)  
# (2) Haptic words show less exclusivity than auditory and visual ones within   
# properties but not within concepts (\*)  
  
  
# DUTCH  
# Setting contrasts based on means  
contrasts(all$main) <- cbind(c(2,0,-2), c(-1,2,-1))  
# (1) Aud vs Vis; (2) Hap vs Aud-&-Vis  
contrasts(all$main)

## [,1] [,2]  
## a 2 -1  
## h 0 2  
## v -2 -1

fitt <- aov(Exclusivity ~ main \* cat, data=all)  
plot(fitt) # must click over the plot several times in order to continue



summary(fitt)

## Df Sum Sq Mean Sq F value Pr(>F)   
## main 2 0.448 0.2240 8.679 0.000188 \*\*\*  
## cat 1 2.416 2.4157 93.616 < 2e-16 \*\*\*  
## main:cat 2 0.179 0.0897 3.477 0.031399 \*   
## Residuals 741 19.121 0.0258   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
## 12 observations deleted due to missingness

drop1(fitt,~.,test="F")

## Single term deletions  
##   
## Model:  
## Exclusivity ~ main \* cat  
## Df Sum of Sq RSS AIC F value Pr(>F)   
## <none> 19.121 -2726.0   
## main 2 0.04532 19.166 -2728.2 0.8782 0.4160   
## cat 1 1.05008 20.171 -2688.0 40.6939 3.134e-10 \*\*\*  
## main:cat 2 0.17945 19.300 -2723.0 3.4772 0.0314 \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Anova(fitt)

## Anova Table (Type II tests)  
##   
## Response: Exclusivity  
## Sum Sq Df F value Pr(>F)   
## main 0.4752 2 9.2071 0.0001123 \*\*\*  
## cat 2.4157 1 93.6158 < 2.2e-16 \*\*\*  
## main:cat 0.1795 2 3.4772 0.0313993 \*   
## Residuals 19.1210 741   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Anova(fitt, type = "II")

## Anova Table (Type II tests)  
##   
## Response: Exclusivity  
## Sum Sq Df F value Pr(>F)   
## main 0.4752 2 9.2071 0.0001123 \*\*\*  
## cat 2.4157 1 93.6158 < 2.2e-16 \*\*\*  
## main:cat 0.1795 2 3.4772 0.0313993 \*   
## Residuals 19.1210 741   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Anova(fitt, type = "III")

## Anova Table (Type III tests)  
##   
## Response: Exclusivity  
## Sum Sq Df F value Pr(>F)   
## (Intercept) 14.8547 1 575.6647 < 2.2e-16 \*\*\*  
## main 0.0453 2 0.8782 0.4160   
## cat 1.0501 1 40.6939 3.134e-10 \*\*\*  
## main:cat 0.1795 2 3.4772 0.0314 \*   
## Residuals 19.1210 741   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

summary.lm(fitt)

##   
## Call:  
## aov(formula = Exclusivity ~ main \* cat, data = all)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.36745 -0.11897 -0.00947 0.09959 0.61844   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.276142 0.011509 23.993 < 2e-16 \*\*\*  
## main1 -0.002468 0.006219 -0.397 0.69157   
## main2 -0.009482 0.008995 -1.054 0.29214   
## catprop 0.101589 0.015925 6.379 3.13e-10 \*\*\*  
## main1:catprop 0.006464 0.008425 0.767 0.44320   
## main2:catprop -0.033250 0.012608 -2.637 0.00854 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.1606 on 741 degrees of freedom  
## (12 observations deleted due to missingness)  
## Multiple R-squared: 0.1373, Adjusted R-squared: 0.1315   
## F-statistic: 23.59 on 5 and 741 DF, p-value: < 2.2e-16

# RESULTS: Dutch properties with more exclusivity than concepts(\*\*\*)  
# Contrasts: (1) Aud vs Vis (non-sig)  
# (2) Haptic words show less exclusivity than auditory and visual ones within   
# properties but not within concepts (\*\*)  
  
# Overall, these results stem from the nature of human perception. What exclusivity   
# seems to be indexing is the degree to which percepts will naturally co-occur. Thus,  
# visual and auditory words have relatively higher exclusivities because what we   
# see and hear often stands on its own. We can often see thing but not hear or touch   
# them. By the same token, we often hear things that we cannot see or touch. Now, in  
# contrast, if we can touch something, we likely can see and hear it too--hence the   
# low exclusivity of haptic items.  
  
  
  
# SAME PLOT-WISE:  
# Barplot of exclusivity percentiles within modalities for Dutch items (as in   
# van Dantzig et al., 2011, but separately for properties and concepts)  
  
all<-read.csv('all.csv')  
  
allNL = all[!all$main == '',]  
allNL$main = levels(droplevels(allNL$main))  
  
concs <- allNL[allNL$cat == 'conc' & !allNL$normed == 'English' & !allNL$main == '',]  
props <- allNL[allNL$cat == 'prop' & !allNL$normed == 'English' & !allNL$main == '',]  
  
concs$main = levels(droplevels(as.factor(concs$main)))  
props$main = levels(droplevels(as.factor(props$main)))  
concs$main = as.factor(concs$main)  
props$main = as.factor(props$main)  
nrow(concs$main)

## NULL

nrow(props$main)

## NULL

allNL$catmain <- with(allNL, interaction(cat, main))  
str(allNL$catmain)

## Factor w/ 6 levels "conc.a","prop.a",..: 1 3 5 1 3 5 1 3 5 1 ...

allNL$section = floor(allNL$Exclusivity \* 4)   
table(allNL$section)

##   
## 0 1 2 3 4   
## 252 368 114 12 1

str(allNL$section)

## num [1:759] 1 0 1 1 2 1 2 2 1 1 ...

table(allNL$section) # order = 01234

##   
## 0 1 2 3 4   
## 252 368 114 12 1

allNL$section = as.factor(allNL$section)  
revalue(allNL$section, c("0"="0-20%", "1"="20-40%", "2"="40-60%", "3"="60-80%",   
"4"="80-100%"))

## [1] 20-40% 0-20% 20-40% 20-40% 40-60% 20-40% 40-60% 40-60%   
## [9] 20-40% 20-40% 0-20% 0-20% 0-20% 0-20% 20-40% 0-20%   
## [17] 0-20% 20-40% 20-40% 0-20% 20-40% 0-20% 20-40% 20-40%   
## [25] 0-20% 40-60% 0-20% 0-20% 0-20% 0-20% 20-40% 40-60%   
## [33] 0-20% 20-40% 20-40% 20-40% 20-40% 0-20% 0-20% 20-40%   
## [41] 0-20% 0-20% 0-20% 0-20% 0-20% 0-20% 20-40% 0-20%   
## [49] 20-40% 40-60% 0-20% 0-20% 40-60% 0-20% 20-40% 40-60%   
## [57] 0-20% 20-40% 0-20% 0-20% 20-40% 20-40% 20-40% 60-80%   
## [65] 0-20% 0-20% 20-40% 0-20% 40-60% 0-20% 20-40% 0-20%   
## [73] 20-40% 0-20% 20-40% 0-20% 20-40% 0-20% 20-40% 0-20%   
## [81] 0-20% 0-20% 40-60% 0-20% 0-20% 0-20% 40-60% 20-40%   
## [89] 0-20% 0-20% 0-20% 20-40% 20-40% 0-20% 20-40% 20-40%   
## [97] 0-20% 20-40% 0-20% 20-40% 20-40% 0-20% 0-20% 20-40%   
## [105] 0-20% 20-40% 20-40% 0-20% 0-20% 0-20% 0-20% 0-20%   
## [113] 20-40% 0-20% 0-20% 0-20% 20-40% 0-20% 0-20% 0-20%   
## [121] 20-40% 0-20% 0-20% 0-20% 20-40% 40-60% 20-40% 0-20%   
## [129] 20-40% 0-20% 0-20% 0-20% 20-40% 20-40% 20-40% 20-40%   
## [137] 20-40% 0-20% 20-40% 0-20% 20-40% 20-40% 20-40% 0-20%   
## [145] 0-20% 0-20% 0-20% 0-20% 40-60% 40-60% 0-20% 0-20%   
## [153] 20-40% 20-40% 0-20% 20-40% 0-20% 20-40% 0-20% 0-20%   
## [161] 0-20% 20-40% 20-40% 20-40% 0-20% 20-40% 0-20% 0-20%   
## [169] 20-40% 20-40% 0-20% 0-20% 20-40% 20-40% 40-60% 20-40%   
## [177] 20-40% 20-40% 20-40% 40-60% 20-40% 0-20% 20-40% 20-40%   
## [185] 20-40% 40-60% 20-40% 0-20% 20-40% 20-40% 0-20% 0-20%   
## [193] 0-20% 0-20% 0-20% 0-20% 0-20% 0-20% 0-20% 0-20%   
## [201] 20-40% 20-40% 20-40% 20-40% 0-20% 20-40% 0-20% 20-40%   
## [209] 0-20% 20-40% 20-40% 20-40% 40-60% 0-20% 20-40% 20-40%   
## [217] 20-40% 0-20% 40-60% 0-20% 20-40% 20-40% 20-40% 20-40%   
## [225] 0-20% 20-40% 20-40% 20-40% 20-40% 0-20% 20-40% 20-40%   
## [233] 20-40% 20-40% 0-20% 20-40% 20-40% 20-40% 0-20% 0-20%   
## [241] 20-40% 20-40% 20-40% 20-40% 20-40% 20-40% 20-40% 0-20%   
## [249] 20-40% 20-40% 0-20% 20-40% 0-20% 20-40% 20-40% 0-20%   
## [257] 0-20% 0-20% 0-20% 20-40% 20-40% 20-40% 0-20% 0-20%   
## [265] 0-20% 0-20% 20-40% 0-20% 0-20% 0-20% 0-20% 0-20%   
## [273] 0-20% 20-40% 20-40% 0-20% 20-40% 0-20% 20-40% 20-40%   
## [281] 20-40% 20-40% 0-20% 20-40% 20-40% 20-40% 0-20% 0-20%   
## [289] 20-40% 0-20% 20-40% 20-40% 20-40% 20-40% 0-20% 0-20%   
## [297] 0-20% 20-40% 40-60% 20-40% 20-40% 20-40% 0-20% 20-40%   
## [305] 20-40% 40-60% 20-40% 0-20% 20-40% 20-40% 0-20% 20-40%   
## [313] 20-40% 0-20% 20-40% 20-40% 40-60% 20-40% 20-40% 0-20%   
## [321] 40-60% 0-20% 0-20% 0-20% 20-40% 0-20% 0-20% 0-20%   
## [329] 20-40% 0-20% 20-40% 20-40% 20-40% 0-20% 0-20% 20-40%   
## [337] 0-20% 40-60% 20-40% 20-40% 40-60% 0-20% 20-40% 20-40%   
## [345] 20-40% 0-20% 20-40% 0-20% 0-20% 0-20% 0-20% 0-20%   
## [353] 20-40% 0-20% 20-40% 20-40% 0-20% 20-40% 0-20% 40-60%   
## [361] 40-60% 20-40% 20-40% 20-40% 20-40% 20-40% 0-20% 20-40%   
## [369] 20-40% 20-40% 20-40% 60-80% 0-20% 20-40% 20-40% 20-40%   
## [377] 0-20% 20-40% 20-40% 40-60% 0-20% 0-20% 0-20% 0-20%   
## [385] 0-20% 20-40% 40-60% 20-40% 0-20% 20-40% 40-60% 0-20%   
## [393] 0-20% 20-40% 0-20% 20-40% 0-20% 20-40% 0-20% 20-40%   
## [401] 20-40% 0-20% 20-40% 0-20% 0-20% 0-20% 20-40% 20-40%   
## [409] 40-60% 0-20% 20-40% 0-20% 20-40% 20-40% 0-20% 20-40%   
## [417] 40-60% 40-60% 20-40% 20-40% 40-60% 20-40% 40-60% 40-60%   
## [425] 0-20% 40-60% 20-40% 20-40% 40-60% 20-40% 20-40% 20-40%   
## [433] 0-20% 0-20% 20-40% 0-20% 0-20% 20-40% 20-40% 60-80%   
## [441] 20-40% 40-60% 20-40% 0-20% 20-40% 20-40% 20-40% 20-40%   
## [449] 20-40% 20-40% 20-40% 40-60% 20-40% 40-60% 20-40% 40-60%   
## [457] 20-40% 0-20% 0-20% 40-60% 20-40% 0-20% 60-80% 20-40%   
## [465] 20-40% 40-60% 20-40% 40-60% 40-60% 20-40% 20-40% 20-40%   
## [473] 60-80% 40-60% 20-40% 20-40% 0-20% 40-60% 40-60% 40-60%   
## [481] 0-20% 20-40% 0-20% 20-40% 20-40% 20-40% 60-80% 40-60%   
## [489] 20-40% 40-60% 20-40% 0-20% 20-40% 20-40% 20-40% 20-40%   
## [497] 20-40% 20-40% 40-60% 40-60% 20-40% 20-40% 60-80% 20-40%   
## [505] 40-60% 40-60% 40-60% 40-60% 0-20% 20-40% 40-60% 20-40%   
## [513] 40-60% 20-40% 40-60% 20-40% 20-40% 20-40% 60-80% 0-20%   
## [521] 40-60% 20-40% 40-60% 0-20% 20-40% 20-40% 0-20% 40-60%   
## [529] 60-80% 20-40% 0-20% 20-40% 20-40% 0-20% 20-40% 0-20%   
## [537] 20-40% 20-40% 0-20% 0-20% 60-80% 20-40% 20-40% 40-60%   
## [545] 20-40% 0-20% 20-40% 40-60% 20-40% 20-40% 20-40% 40-60%   
## [553] 20-40% 0-20% 20-40% 20-40% 40-60% 0-20% 20-40% 20-40%   
## [561] 20-40% 40-60% 20-40% 20-40% 0-20% 40-60% 0-20% 0-20%   
## [569] 40-60% 40-60% 40-60% 20-40% 40-60% 20-40% 0-20% 20-40%   
## [577] 40-60% 0-20% 20-40% 40-60% 40-60% 40-60% 20-40% 40-60%   
## [585] 40-60% 20-40% 20-40% 40-60% 0-20% 40-60% 20-40% 20-40%   
## [593] 20-40% 20-40% 40-60% 40-60% 0-20% 40-60% 20-40% 0-20%   
## [601] <NA> <NA> <NA> <NA> <NA> <NA> <NA> <NA>   
## [609] <NA> <NA> <NA> <NA> 80-100% 20-40% 0-20% 20-40%   
## [617] 20-40% 20-40% 20-40% 20-40% 20-40% 20-40% 20-40% 40-60%   
## [625] 20-40% 40-60% 40-60% 60-80% 0-20% 20-40% 20-40% 40-60%   
## [633] 0-20% 0-20% 0-20% 20-40% 0-20% 20-40% 20-40% 20-40%   
## [641] 0-20% 20-40% 0-20% 0-20% 20-40% 20-40% 40-60% 40-60%   
## [649] 40-60% 0-20% 20-40% 20-40% 20-40% 0-20% 20-40% 40-60%   
## [657] 40-60% 20-40% 20-40% 20-40% 40-60% 0-20% 0-20% 40-60%   
## [665] 40-60% 20-40% 20-40% 0-20% 20-40% 20-40% 20-40% 0-20%   
## [673] 40-60% 0-20% 20-40% 20-40% 20-40% 40-60% 0-20% 40-60%   
## [681] 20-40% 20-40% 0-20% 20-40% 20-40% 20-40% 20-40% 0-20%   
## [689] 20-40% 20-40% 20-40% 20-40% 20-40% 20-40% 20-40% 20-40%   
## [697] 20-40% 0-20% 40-60% 20-40% 20-40% 0-20% 20-40% 20-40%   
## [705] 20-40% 20-40% 20-40% 0-20% 20-40% 40-60% 0-20% 20-40%   
## [713] 20-40% 20-40% 40-60% 0-20% 40-60% 40-60% 40-60% 20-40%   
## [721] 0-20% 20-40% 20-40% 0-20% 20-40% 20-40% 20-40% 40-60%   
## [729] 20-40% 40-60% 20-40% 0-20% 0-20% 0-20% 20-40% 20-40%   
## [737] 20-40% 40-60% 40-60% 20-40% 0-20% 0-20% 60-80% 20-40%   
## [745] 0-20% 0-20% 40-60% 20-40% 20-40% 20-40% 20-40% 40-60%   
## [753] 40-60% 20-40% 20-40% 0-20% 20-40% 40-60% 20-40%   
## Levels: 0-20% 20-40% 40-60% 60-80% 80-100%

allNL$section = mapvalues(allNL$section, from = c(0, 1, 2, 3, 4), to = c("0-20%",   
"20-40%", "40-60%", "60-80%", "80-100%"))  
table(allNL$section)

##   
## 0-20% 20-40% 40-60% 60-80% 80-100%   
## 252 368 114 12 1

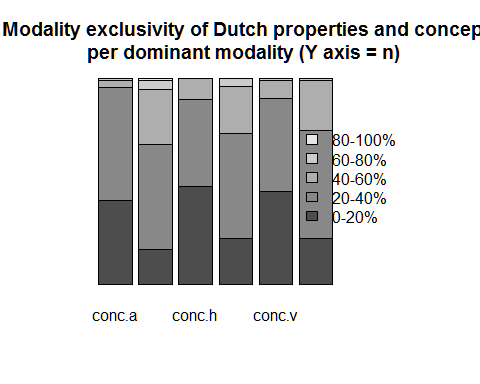
str(allNL$section)

## Factor w/ 5 levels "0-20%","20-40%",..: 2 1 2 2 3 2 3 3 2 2 ...

counts <- table(allNL$section, allNL$catmain)  
counts

##   
## conc.a prop.a conc.h prop.h conc.v prop.v  
## 0-20% 56 19 65 25 62 25  
## 20-40% 75 57 58 57 62 59  
## 40-60% 5 30 14 26 12 27  
## 60-80% 1 5 0 4 1 1  
## 80-100% 0 1 0 0 0 0

counts = prop.table(counts, 2)  
  
# see plot:  
barplot(counts, width=10, main = 'Modality exclusivity of Dutch properties and concepts   
per dominant modality (Y axis = n) ', legend = rownames(counts), xlim=c(0,100),   
axes=FALSE, args.legend = list(x = "topright", bty = "n", inset=c(.1, .2)))



# ! THE PLOT IS SHOWN BADLY ON HERE. PLEASE SEE THE SAVED PLOT.   
  
# Below, run first line, then return and keep running:  
png(file="stacked\_exc.png", units="in", width=6, height=6, res=1000)  
par(mar=c(2,-.3,3,-.3)+.4) # run twice, if necessary   
barplot(counts, width=10, main = 'Modality exclusivity of Dutch properties and concepts   
per dominant modality (Y axis = n) ', legend = rownames(counts), xlim=c(0,100),   
axes=FALSE, args.legend = list(x = "topright", bty = "n", inset=c(.1, .2)))  
dev.off()

## png   
## 2

# Same plot for the English items of Lynott and Connell (of course w/out gustatory   
# or olfactory)  
  
allENG = all[!all$main\_eng == '',]  
allENG$main\_eng = levels(droplevels(allENG$main\_eng))  
  
allENG$catmain <- with(allENG, interaction(cat, main\_eng))  
str(allENG$catmain)

## Factor w/ 6 levels "conc.a","prop.a",..: NA 3 5 1 3 5 1 3 5 1 ...

allENG$section = floor(allENG$exc\_eng \* 5)  
table(allENG$section)

##   
## 0 1 2 3 4   
## 33 288 313 81 20

str(allENG$section)

## num [1:759] NA 2 1 2 2 2 3 2 1 2 ...

table(allENG$section) # order = 01234

##   
## 0 1 2 3 4   
## 33 288 313 81 20

allENG$section = as.factor(allENG$section)  
revalue(allENG$section, c("0"="0-20%", "1"="20-40%", "2"="40-60%", "3"="60-80%",   
"4"="80-100%"))

## [1] <NA> 40-60% 20-40% 40-60% 40-60% 40-60% 60-80% 40-60%   
## [9] 20-40% 40-60% 40-60% 20-40% 40-60% 20-40% 40-60% 20-40%   
## [17] 20-40% 40-60% 40-60% 40-60% 20-40% 0-20% 20-40% 20-40%   
## [25] 60-80% 40-60% <NA> 40-60% 40-60% 20-40% 60-80% 40-60%   
## [33] 20-40% 40-60% 40-60% 20-40% 20-40% 20-40% 20-40% 20-40%   
## [41] 20-40% 20-40% 20-40% 40-60% 20-40% 40-60% 20-40% 40-60%   
## [49] 40-60% 20-40% 40-60% 20-40% 20-40% 20-40% 20-40% 20-40%   
## [57] 20-40% 20-40% 20-40% 40-60% 40-60% 40-60% 20-40% 40-60%   
## [65] 40-60% 40-60% 40-60% 20-40% 40-60% 0-20% 20-40% 0-20%   
## [73] 20-40% 40-60% <NA> 60-80% 40-60% 40-60% 40-60% 20-40%   
## [81] 20-40% 20-40% 20-40% 20-40% 20-40% 20-40% 60-80% 40-60%   
## [89] 20-40% 20-40% 20-40% 20-40% 0-20% 20-40% 40-60% 20-40%   
## [97] 20-40% 40-60% 20-40% 20-40% 20-40% 40-60% 20-40% 20-40%   
## [105] 20-40% 20-40% 40-60% 20-40% 20-40% 20-40% 20-40% <NA>   
## [113] <NA> 20-40% 20-40% 40-60% 40-60% <NA> 0-20% 0-20%   
## [121] <NA> 20-40% 40-60% 20-40% 40-60% 40-60% 20-40% 20-40%   
## [129] 40-60% <NA> 40-60% 40-60% <NA> 40-60% 40-60% 40-60%   
## [137] 40-60% 20-40% 60-80% 0-20% 40-60% 60-80% 60-80% 40-60%   
## [145] 20-40% 40-60% 20-40% 20-40% 40-60% 40-60% 40-60% 20-40%   
## [153] <NA> 20-40% 20-40% 20-40% 40-60% 20-40% 20-40% 20-40%   
## [161] 20-40% <NA> 40-60% 40-60% 40-60% 40-60% 40-60% 40-60%   
## [169] 40-60% 20-40% 40-60% 20-40% 40-60% 20-40% 40-60% 40-60%   
## [177] 40-60% 40-60% 40-60% 80-100% 40-60% 20-40% 60-80% 40-60%   
## [185] 40-60% 40-60% 40-60% 40-60% 40-60% 40-60% 40-60% 20-40%   
## [193] 20-40% 20-40% 20-40% 20-40% 20-40% 20-40% 40-60% 0-20%   
## [201] 40-60% 40-60% <NA> 20-40% 40-60% 20-40% 40-60% 20-40%   
## [209] 20-40% 20-40% <NA> 20-40% 40-60% 20-40% <NA> 40-60%   
## [217] 40-60% 20-40% <NA> 20-40% 20-40% 20-40% 40-60% 60-80%   
## [225] 20-40% 40-60% 0-20% 20-40% 40-60% 40-60% 20-40% 40-60%   
## [233] 40-60% 20-40% <NA> 40-60% 40-60% 20-40% 20-40% 20-40%   
## [241] 40-60% 20-40% 20-40% 40-60% 20-40% 40-60% 40-60% 40-60%   
## [249] 20-40% 40-60% 0-20% 40-60% <NA> 0-20% 20-40% 20-40%   
## [257] 20-40% 20-40% 20-40% 40-60% 60-80% 20-40% 20-40% 40-60%   
## [265] 40-60% 20-40% 40-60% 20-40% <NA> 40-60% 40-60% 40-60%   
## [273] 0-20% 20-40% <NA> 20-40% <NA> 20-40% 60-80% 20-40%   
## [281] 20-40% 40-60% 20-40% 40-60% 20-40% 0-20% 20-40% 20-40%   
## [289] 40-60% 20-40% 40-60% 20-40% 40-60% 20-40% 40-60% 20-40%   
## [297] 20-40% 40-60% 40-60% 60-80% <NA> 20-40% 20-40% 40-60%   
## [305] 60-80% 20-40% 40-60% 20-40% 40-60% 0-20% 20-40% 20-40%   
## [313] 20-40% 20-40% 40-60% 40-60% 40-60% 40-60% 40-60% 20-40%   
## [321] 60-80% 40-60% 0-20% 20-40% <NA> 40-60% 40-60% 20-40%   
## [329] 40-60% 20-40% 20-40% <NA> 40-60% 20-40% 0-20% 20-40%   
## [337] 40-60% 40-60% 0-20% 20-40% 40-60% 20-40% 20-40% 40-60%   
## [345] <NA> 0-20% 0-20% 40-60% 20-40% 20-40% 40-60% 20-40%   
## [353] 40-60% 20-40% 0-20% 40-60% 0-20% 20-40% 40-60% 60-80%   
## [361] 60-80% 20-40% 40-60% 40-60% 20-40% 40-60% 20-40% 40-60%   
## [369] 40-60% 20-40% 20-40% 20-40% 20-40% 40-60% 40-60% 20-40%   
## [377] 40-60% 20-40% 20-40% 40-60% 20-40% 20-40% 40-60% 20-40%   
## [385] 40-60% 0-20% 60-80% 40-60% 20-40% 20-40% 40-60% 20-40%   
## [393] 20-40% 40-60% 20-40% 40-60% 20-40% 40-60% 40-60% 40-60%   
## [401] 40-60% 20-40% 20-40% 20-40% 20-40% 20-40% 60-80% 40-60%   
## [409] 20-40% 20-40% 60-80% 20-40% 40-60% 60-80% 20-40% 40-60%   
## [417] 80-100% 20-40% 40-60% 20-40% 80-100% 40-60% 80-100% 80-100%  
## [425] 20-40% 60-80% 40-60% 40-60% 80-100% 40-60% 40-60% 20-40%   
## [433] 40-60% 0-20% 40-60% 20-40% 40-60% 0-20% 40-60% 60-80%   
## [441] 40-60% 80-100% 80-100% 40-60% 40-60% 40-60% 40-60% 20-40%   
## [449] 20-40% 20-40% 40-60% 60-80% 40-60% 20-40% 40-60% 60-80%   
## [457] 40-60% 20-40% 20-40% 40-60% 20-40% 40-60% 80-100% 20-40%   
## [465] 20-40% 40-60% 40-60% 60-80% 40-60% 60-80% 60-80% 60-80%   
## [473] 20-40% 60-80% 80-100% 40-60% 20-40% 40-60% 80-100% 60-80%   
## [481] 20-40% 60-80% 20-40% 40-60% 40-60% 20-40% 60-80% 60-80%   
## [489] 40-60% 40-60% 40-60% 40-60% 40-60% 40-60% 40-60% 40-60%   
## [497] 20-40% 40-60% 40-60% 40-60% 60-80% 20-40% 40-60% 20-40%   
## [505] 60-80% 60-80% 60-80% 60-80% 20-40% 40-60% 60-80% 20-40%   
## [513] 80-100% 40-60% 60-80% 0-20% 40-60% 20-40% 0-20% 40-60%   
## [521] 40-60% 20-40% 80-100% 20-40% 40-60% 40-60% 20-40% 40-60%   
## [529] 60-80% 40-60% 40-60% 40-60% 20-40% 20-40% 40-60% 60-80%   
## [537] 40-60% 40-60% 20-40% 20-40% 80-100% 40-60% 20-40% 40-60%   
## [545] 40-60% 40-60% 40-60% 60-80% 40-60% 20-40% 20-40% 40-60%   
## [553] 20-40% 0-20% 40-60% 40-60% 60-80% 20-40% 40-60% 40-60%   
## [561] 20-40% 40-60% 20-40% 20-40% 20-40% 60-80% 60-80% 40-60%   
## [569] 40-60% 40-60% 40-60% 60-80% 40-60% 40-60% 20-40% 40-60%   
## [577] 40-60% 0-20% 20-40% 20-40% 60-80% 40-60% 20-40% 60-80%   
## [585] 60-80% 40-60% 20-40% 60-80% 20-40% 40-60% 40-60% 60-80%   
## [593] 20-40% 40-60% 60-80% 20-40% 0-20% 60-80% 40-60% 20-40%   
## [601] 20-40% 40-60% 40-60% 20-40% 20-40% 60-80% 40-60% 60-80%   
## [609] 20-40% 40-60% 20-40% 40-60% 20-40% 20-40% 20-40% 40-60%   
## [617] 40-60% 20-40% 40-60% 40-60% 20-40% 20-40% 20-40% 60-80%   
## [625] 40-60% 40-60% 40-60% 80-100% 20-40% 20-40% 60-80% 40-60%   
## [633] 20-40% 40-60% 20-40% 20-40% 20-40% 40-60% 20-40% 60-80%   
## [641] 20-40% 40-60% 40-60% 20-40% 40-60% 40-60% 40-60% 40-60%   
## [649] 40-60% 40-60% 40-60% 40-60% 20-40% 40-60% 40-60% 80-100%  
## [657] 80-100% 0-20% 20-40% 40-60% 60-80% 20-40% 40-60% 20-40%   
## [665] 60-80% 60-80% 20-40% 20-40% 40-60% 40-60% 60-80% 20-40%   
## [673] 60-80% 20-40% 80-100% 60-80% 20-40% 40-60% 40-60% 20-40%   
## [681] 40-60% 40-60% 20-40% 60-80% 40-60% 20-40% 20-40% 20-40%   
## [689] 40-60% 40-60% 20-40% 20-40% 60-80% 60-80% 40-60% 40-60%   
## [697] 60-80% 40-60% 40-60% 40-60% 0-20% 20-40% 60-80% 20-40%   
## [705] 40-60% 60-80% 40-60% 40-60% 40-60% 40-60% 40-60% 20-40%   
## [713] 40-60% 60-80% 60-80% 40-60% 40-60% 60-80% 40-60% 40-60%   
## [721] 40-60% 20-40% 40-60% 60-80% 40-60% 20-40% 40-60% 40-60%   
## [729] 40-60% 60-80% 20-40% 40-60% 40-60% 0-20% 20-40% 20-40%   
## [737] 20-40% 60-80% 20-40% 40-60% 40-60% 20-40% 80-100% 40-60%   
## [745] 20-40% 20-40% 20-40% 20-40% 40-60% 60-80% 40-60% 60-80%   
## [753] 40-60% 60-80% 40-60% 20-40% 0-20% 80-100% 20-40%   
## Levels: 0-20% 20-40% 40-60% 60-80% 80-100%

allENG$section = mapvalues(allENG$section, from = c(0, 1, 2, 3, 4), to = c("0-20%",   
"20-40%", "40-60%", "60-80%", "80-100%"))  
table(allENG$section)

##   
## 0-20% 20-40% 40-60% 60-80% 80-100%   
## 33 288 313 81 20

str(allENG$section)

## Factor w/ 5 levels "0-20%","20-40%",..: NA 3 2 3 3 3 4 3 2 3 ...

counts <- table(allENG$section, allENG$catmain)  
counts

##   
## conc.a prop.a conc.h prop.h conc.v prop.v  
## 0-20% 6 2 10 5 6 4  
## 20-40% 60 27 61 42 64 34  
## 40-60% 54 54 58 44 52 51  
## 60-80% 7 23 4 18 9 20  
## 80-100% 0 8 0 5 1 6

counts = prop.table(counts, 2)  
  
# below, run first line, then return and keep running:  
png(file="stacked\_exc\_eng.png", units="in", width=6, height=6, res=1000)  
par(mar=c(2,-.3,3,-.3)+.4) # run twice, if necessary   
barplot(counts, width=10, main = 'Modality exclusivity of English properties and concepts   
per dominant modality (Y axis = n) ', legend = rownames(counts), xlim=c(0,100),   
axes=FALSE, args.legend = list(x = "topright", bty = "n", inset=c(.1, .2)))  
dev.off()

## png   
## 2

# See in folder and compare.  
  
  
# Comparison English Dutch on exclusivity  
# Properties  
t.test(props$exc\_eng, mu = 0.40)

##   
## One Sample t-test  
##   
## data: props$exc\_eng  
## t = 8.5326, df = 335, p-value = 5.031e-16  
## alternative hypothesis: true mean is not equal to 0.4  
## 95 percent confidence interval:  
## 0.4626335 0.5001642  
## sample estimates:  
## mean of x   
## 0.4813988

# The difference is considerable, t(734) = 18.8, p < .001  
# dz = t/vn = 0.47  
  
# Concepts  
t.test(concs$exc\_eng, mu = 0.29)

##   
## One Sample t-test  
##   
## data: concs$exc\_eng  
## t = 16.857, df = 386, p-value < 2.2e-16  
## alternative hypothesis: true mean is not equal to 0.29  
## 95 percent confidence interval:  
## 0.3831297 0.4077230  
## sample estimates:  
## mean of x   
## 0.3954264

# The difference is considerable, t(734) = 18.8, p < .001  
# dz = t/vn = 0.83  
# \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
  
# RELATION AMONG MODALITIES  
  
# Below, very informative plots based on Principal Components Analysis (PCA),   
# as in Lynott and Connell (2009, 2013)  
# Firstly it is performed on the Dutch norms, then on the English ones, leaving out   
# gustatory and olfactory scores and words. At the end, Dutch and English plots are   
# compared.  
  
all <- read.csv('all.csv')  
nrow(all) # 747 used in Dutch norms + English not used

## [1] 759

# ON ENGLISH NORMS  
# PCA plotting on the English norms, as based on Lynott and Connell's   
# supplementary materials (http://www.lancaster.ac.uk/people/connelll/lab/norms.html).   
  
# ENG PROPERTIES  
# check conditions for a PCA  
# matrix  
  
eng\_prop <- all[all$cat == 'prop', c('Aud\_eng', 'Hap\_eng', 'Vis\_eng')]  
nrow(eng\_prop)

## [1] 343

eng\_prop\_matrix <- cor(eng\_prop, use = 'complete.obs')  
eng\_prop\_matrix

## Aud\_eng Hap\_eng Vis\_eng  
## Aud\_eng 1.0000000 -0.4165084 -0.6247598  
## Hap\_eng -0.4165084 1.0000000 0.2344421  
## Vis\_eng -0.6247598 0.2344421 1.0000000

round(eng\_prop\_matrix, 2)

## Aud\_eng Hap\_eng Vis\_eng  
## Aud\_eng 1.00 -0.42 -0.62  
## Hap\_eng -0.42 1.00 0.23  
## Vis\_eng -0.62 0.23 1.00

# OK: correlations good for a PCA, with enough < .3  
  
# now on the raw vars:  
nrow(eng\_prop)

## [1] 343

cortest.bartlett(eng\_prop)

## R was not square, finding R from data

## $chisq  
## [1] 233.5851  
##   
## $p.value  
## [1] 2.320842e-50  
##   
## $df  
## [1] 3

# GOOD: Bartlett's test significant   
  
# KMO: Kaiser-Meyer-Olkin Measure of Sampling Adequacy  
KMO(eng\_prop\_matrix)

## Kaiser-Meyer-Olkin factor adequacy  
## Call: KMO(r = eng\_prop\_matrix)  
## Overall MSA = 0.56  
## MSA for each item =   
## Aud\_eng Hap\_eng Vis\_eng   
## 0.54 0.64 0.55

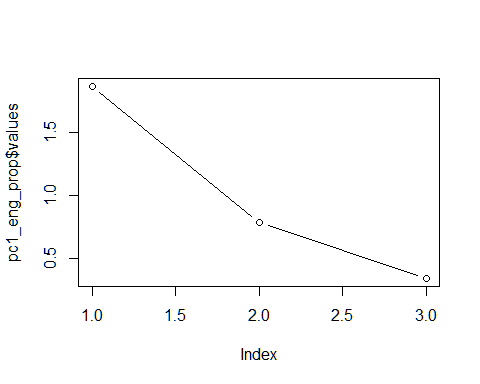
# Result: .56 = mediocre. PCA not strongly recommended. But we still do it  
# because the purpose is graphical only.  
  
# check determinant  
det(eng\_prop\_matrix)

## [1] 0.5032448

# GOOD: >0.00001  
  
# start off with unrotated PCA  
pc1\_eng\_prop <- psych::principal(eng\_prop, nfactors = 3, rotate = "none")  
pc1\_eng\_prop

## Principal Components Analysis  
## Call: psych::principal(r = eng\_prop, nfactors = 3, rotate = "none")  
## Standardized loadings (pattern matrix) based upon correlation matrix  
## PC1 PC2 PC3 h2 u2 com  
## Aud\_eng -0.89 0.13 0.44 1 -2.2e-16 1.5  
## Hap\_eng 0.64 0.75 0.15 1 1.1e-16 2.0  
## Vis\_eng 0.81 -0.46 0.36 1 -4.4e-16 2.0  
##   
## PC1 PC2 PC3  
## SS loadings 1.87 0.79 0.34  
## Proportion Var 0.62 0.26 0.11  
## Cumulative Var 0.62 0.89 1.00  
## Proportion Explained 0.62 0.26 0.11  
## Cumulative Proportion 0.62 0.89 1.00  
##   
## Mean item complexity = 1.9  
## Test of the hypothesis that 3 components are sufficient.  
##   
## The root mean square of the residuals (RMSR) is 0   
## with the empirical chi square 0 with prob < NA   
##   
## Fit based upon off diagonal values = 1

# RESULT: Extract either one PC, acc to Kaiser's criterion, or two RCs, acc to   
# Joliffe's (Field, Miles, & Field, 2012)  
  
# Unrotated: scree plot  
plot(pc1\_eng\_prop$values, type = "b")



# Result: again one or two RCs should be extracted  
  
# Now with varimax rotation, Kaiser-normalized (by default)  
pc2\_eng\_prop <- psych::principal(eng\_prop, nfactors = 2, rotate = "varimax",   
scores = TRUE)  
pc2\_eng\_prop

## Principal Components Analysis  
## Call: psych::principal(r = eng\_prop, nfactors = 2, rotate = "varimax",   
## scores = TRUE)  
## Standardized loadings (pattern matrix) based upon correlation matrix  
## RC1 RC2 h2 u2 com  
## Aud\_eng -0.82 -0.36 0.81 0.190 1.4  
## Hap\_eng 0.16 0.98 0.98 0.022 1.1  
## Vis\_eng 0.93 0.04 0.87 0.130 1.0  
##   
## RC1 RC2  
## SS loadings 1.57 1.09  
## Proportion Var 0.52 0.36  
## Cumulative Var 0.52 0.89  
## Proportion Explained 0.59 0.41  
## Cumulative Proportion 0.59 1.00  
##   
## Mean item complexity = 1.1  
## Test of the hypothesis that 2 components are sufficient.  
##   
## The root mean square of the residuals (RMSR) is 0.1   
## with the empirical chi square 21.7 with prob < NA   
##   
## Fit based upon off diagonal values = 0.95

pc2\_eng\_prop$loadings

##   
## Loadings:  
## RC1 RC2   
## Aud\_eng -0.825 -0.360  
## Hap\_eng 0.156 0.977  
## Vis\_eng 0.932   
##   
## RC1 RC2  
## SS loadings 1.573 1.085  
## Proportion Var 0.524 0.362  
## Cumulative Var 0.524 0.886

# two components are good, as they both have eigenvalues over 1  
  
pc2\_eng\_prop$residual

## Aud\_eng Hap\_eng Vis\_eng  
## Aud\_eng 0.18971667 0.06403298 0.15723160  
## Hap\_eng 0.06403298 0.02161235 0.05306865  
## Vis\_eng 0.15723160 0.05306865 0.13030894

pc2\_eng\_prop$fit

## [1] 0.9724565

pc2\_eng\_prop$communality

## Aud\_eng Hap\_eng Vis\_eng   
## 0.8102833 0.9783877 0.8696911

# Results based on a Kaiser-normalizalized orthogonal (varimax) rotation  
# (by default in psych::stats). Residuals bad: more than 50% have absolute   
# values > 0.05. Model fit good, > .90. Communalities good,   
# all > .7.   
  
# subset and add PCs  
eng\_props <- all[all$cat == 'prop', ]  
nrow(eng\_props)

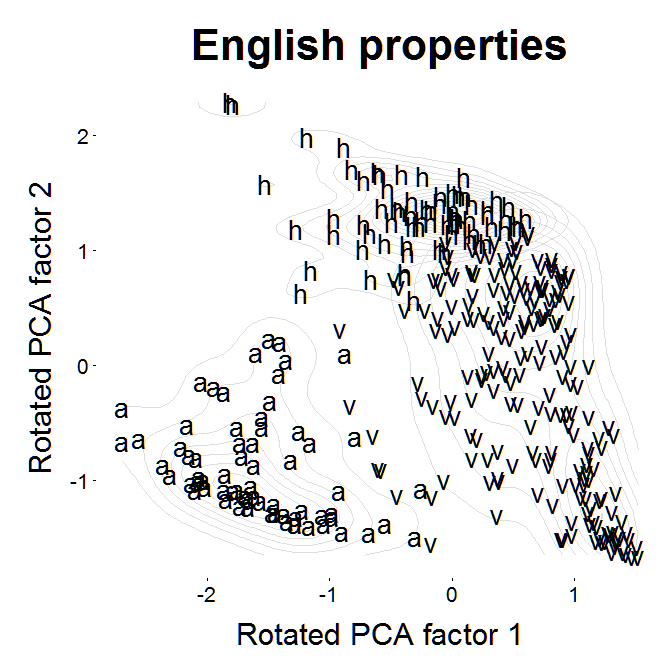
## [1] 343

eng\_props <- cbind(eng\_props, pc2\_eng\_prop$scores)  
nrow(eng\_props)

## [1] 343

# Finally, plot  
Engprops <- ggplot(eng\_props,  
 aes(RC1, RC2, label = as.character(main\_eng))) +  
 aes (x = RC1, y = RC2, by = main\_eng) + stat\_density2d (color = "gray87") +  
 geom\_text(size = 7) +  
 ggtitle ('English properties') +  
 theme\_bw() + # theme with white background  
 theme( # clear background, gridlines, chart border  
 plot.background = element\_blank()  
 ,panel.grid.major = element\_blank()  
 ,panel.grid.minor = element\_blank()  
 ,panel.border = element\_blank()  
 ) +  
 theme(axis.line = element\_line(color = 'black')) + # draw x and y lines  
 theme(axis.title.x = element\_text(colour = 'black', size = 23,   
 margin=margin(15,15,15,15)),  
 axis.title.y = element\_text(colour = 'black', size = 23,   
 margin=margin(15,15,15,15)),  
 axis.text.x = element\_text(size=16),  
 axis.text.y = element\_text(size=16)) +  
 labs(x = "Rotated PCA factor 1", y = "Rotated PCA factor 2") +  
 theme(plot.title = element\_text(size = 32, face = "bold",   
 margin=margin(15,15,15,15)))

plot(Engprops) # ! THE PLOT IS SHOWN BADLY ON HERE. PLEASE SEE THE SAVED PLOT + THEN



# THE COMBINED PLOTS  
  
# Now to save, run first line below and return to keep running. See your folder.  
png(file="Engprops\_highres.png", units="in", width=13, height=13, res=900)  
plot(Engprops)  
dev.off()

## png   
## 2

# Adjust for combined plots:  
  
Engprops4 <- ggplot(eng\_props,  
 aes(RC1, RC2, label = as.character(main\_eng))) +  
 aes (x = RC1, y = RC2, by = main\_eng) + stat\_density2d (color = "gray87") +  
 geom\_text(size = 7) +  
 ggtitle ('English properties') +  
 theme\_bw() + # theme with white background  
 theme( # clear background, gridlines, chart border  
 plot.background = element\_blank()  
 ,panel.grid.major = element\_blank()  
 ,panel.grid.minor = element\_blank()  
 ,panel.border = element\_blank()  
 ) +  
 theme(axis.line = element\_line(color = 'black')) + # draw x and y lines  
 theme(axis.title.x = element\_text(colour = 'black', size = 23,   
 margin=margin(15,15,15,15)),  
 axis.title.y = element\_text(colour = 'black', size = 23,   
 margin=margin(15,15,15,15)),  
 axis.text.x = element\_text(size=16),  
 axis.text.y = element\_text(size=16)) +  
 labs(x = "", y = "Rotated PCA factor 2") +  
 theme(plot.title = element\_text(size = 32, face = "bold",   
 margin=margin(15,15,15,15)))  
  
  
# ENG CONCEPTS  
# check conditions for a PCA  
# matrix  
eng\_conc <- all[all$cat == 'conc', c('Aud\_eng', 'Hap\_eng', 'Vis\_eng')]  
nrow(eng\_conc)

## [1] 416

eng\_conc\_matrix <- cor(eng\_conc, use = 'complete.obs')  
eng\_conc\_matrix

## Aud\_eng Hap\_eng Vis\_eng  
## Aud\_eng 1.000000000 -0.1760092 -0.008395838  
## Hap\_eng -0.176009214 1.0000000 0.554494445  
## Vis\_eng -0.008395838 0.5544944 1.000000000

round(eng\_conc\_matrix, 2)

## Aud\_eng Hap\_eng Vis\_eng  
## Aud\_eng 1.00 -0.18 -0.01  
## Hap\_eng -0.18 1.00 0.55  
## Vis\_eng -0.01 0.55 1.00

# POOR: correlations not apt for a PCA, with too many below .3  
  
# now on the raw data:  
nrow(eng\_conc)

## [1] 416

cortest.bartlett(eng\_conc)

## R was not square, finding R from data

## $chisq  
## [1] 169.7255  
##   
## $p.value  
## [1] 1.458581e-36  
##   
## $df  
## [1] 3

# GOOD: Bartlett's test significant   
  
# KMO: Kaiser-Meyer-Olkin Measure of Sampling Adequacy  
KMO(eng\_conc\_matrix)

## Kaiser-Meyer-Olkin factor adequacy  
## Call: KMO(r = eng\_conc\_matrix)  
## Overall MSA = 0.48  
## MSA for each item =   
## Aud\_eng Hap\_eng Vis\_eng   
## 0.36 0.49 0.48

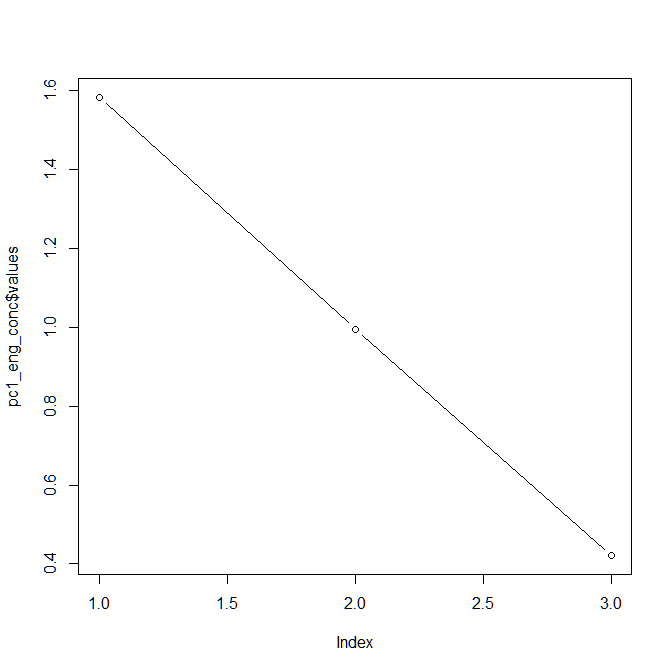
# Result: .48 = poor. PCA not strongly recommended. But we still do it  
# because the purpose is graphical really.  
  
# check determinant  
det(eng\_conc\_matrix)

## [1] 0.663125

# GOOD: >0.00001  
  
# start off with unrotated PCA  
pc1\_eng\_conc <- psych::principal(eng\_conc, nfactors = 3, rotate = "none")  
pc1\_eng\_conc

## Principal Components Analysis  
## Call: psych::principal(r = eng\_conc, nfactors = 3, rotate = "none")  
## Standardized loadings (pattern matrix) based upon correlation matrix  
## PC1 PC2 PC3 h2 u2 com  
## Aud\_eng -0.28 0.95 0.13 1 0.0e+00 1.2  
## Hap\_eng 0.89 0.01 0.46 1 -2.2e-16 1.5  
## Vis\_eng 0.85 0.30 -0.44 1 0.0e+00 1.8  
##   
## PC1 PC2 PC3  
## SS loadings 1.58 1.00 0.42  
## Proportion Var 0.53 0.33 0.14  
## Cumulative Var 0.53 0.86 1.00  
## Proportion Explained 0.53 0.33 0.14  
## Cumulative Proportion 0.53 0.86 1.00  
##   
## Mean item complexity = 1.5  
## Test of the hypothesis that 3 components are sufficient.  
##   
## The root mean square of the residuals (RMSR) is 0   
## with the empirical chi square 0 with prob < NA   
##   
## Fit based upon off diagonal values = 1

# RESULT: Extract either one PC, acc to Kaiser's criterion, or two RCs, acc to   
# Joliffe's (Field, Miles, & Field, 2012)  
  
# Unrotated: scree plot  
plot(pc1\_eng\_conc$values, type = "b")



# Result: two PCs obtain.  
  
# Now with varimax rotation, Kaiser-normalized (by default):  
# always preferable because it captures explained variance best.   
pc2\_eng\_conc <- psych::principal(eng\_conc, nfactors = 2, rotate = "varimax",   
scores = TRUE)  
pc2\_eng\_conc

## Principal Components Analysis  
## Call: psych::principal(r = eng\_conc, nfactors = 2, rotate = "varimax",   
## scores = TRUE)  
## Standardized loadings (pattern matrix) based upon correlation matrix  
## RC1 RC2 h2 u2 com  
## Aud\_eng -0.04 0.99 0.98 0.018 1.0  
## Hap\_eng 0.87 -0.20 0.79 0.211 1.1  
## Vis\_eng 0.89 0.09 0.81 0.192 1.0  
##   
## RC1 RC2  
## SS loadings 1.55 1.03  
## Proportion Var 0.52 0.34  
## Cumulative Var 0.52 0.86  
## Proportion Explained 0.60 0.40  
## Cumulative Proportion 0.60 1.00  
##   
## Mean item complexity = 1  
## Test of the hypothesis that 2 components are sufficient.  
##   
## The root mean square of the residuals (RMSR) is 0.13   
## with the empirical chi square 39.63 with prob < NA   
##   
## Fit based upon off diagonal values = 0.86

pc2\_eng\_conc$loadings

##   
## Loadings:  
## RC1 RC2   
## Aud\_eng 0.990  
## Hap\_eng 0.865 -0.201  
## Vis\_eng 0.894   
##   
## RC1 RC2  
## SS loadings 1.551 1.029  
## Proportion Var 0.517 0.343  
## Cumulative Var 0.517 0.860

pc2\_eng\_conc$residual

## Aud\_eng Hap\_eng Vis\_eng  
## Aud\_eng 0.01775551 0.06123023 -0.05834261  
## Hap\_eng 0.06123023 0.21115367 -0.20119566  
## Vis\_eng -0.05834261 -0.20119566 0.19170728

pc2\_eng\_conc$fit

## [1] 0.9518855

pc2\_eng\_conc$communality

## Aud\_eng Hap\_eng Vis\_eng   
## 0.9822445 0.7888463 0.8082927

# Results based on a Kaiser-normalizalized orthogonal (varimax) rotation   
# (by default in psych::stats). Residuals bad: over 50% have absolute   
# values > 0.05. Model fit good, > .90. Communalities good, all > .7.  
  
# subset and add PCs  
eng\_concs <- all[all$cat == 'conc', ]  
nrow(eng\_concs)

## [1] 416

eng\_concs <- cbind(eng\_concs, pc2\_eng\_conc$scores)  
summary(eng\_concs$RC1, eng\_concs$RC2)

## Min. 1st Qu. Median Mean 3rd Qu. Max. NA's   
## -2.52800 -0.72020 0.03826 0.00000 0.72380 2.18400 24

eng\_concs <- eng\_concs[eng\_concs$normed == 'Dut\_Eng' | eng\_concs$normed ==   
'English',]  
nrow(eng\_concs)

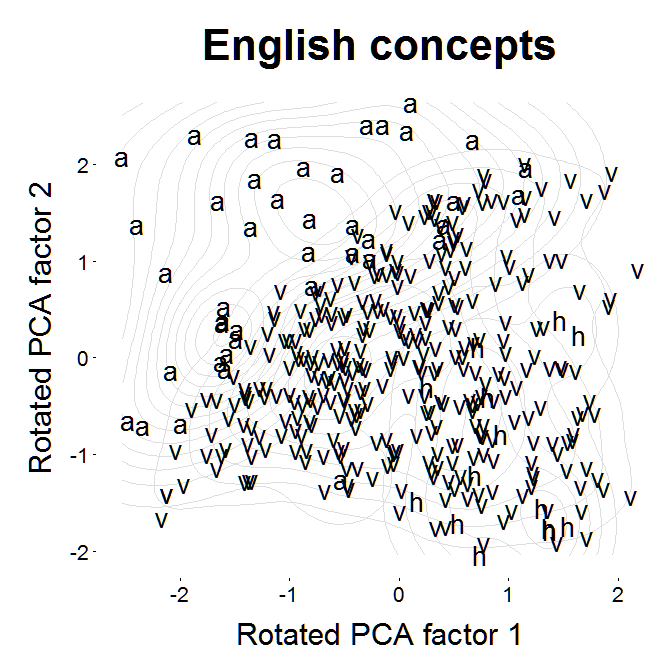
## [1] 392

summary(eng\_concs$RC1, eng\_concs$RC2)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## -2.52800 -0.72020 0.03826 0.00000 0.72380 2.18400

# Finally, plot  
Engconcs <- ggplot(eng\_concs,  
 aes(RC1, RC2, label = as.character(main\_eng))) +  
 aes (x = RC1, y = RC2, by = main\_eng) + stat\_density2d (color = "gray87") +  
 geom\_text(size = 7) +  
 ggtitle ('English concepts') +  
 theme\_bw() + # theme with white background  
 theme( # clear background, gridlines, chart border  
 plot.background = element\_blank()  
 ,panel.grid.major = element\_blank()  
 ,panel.grid.minor = element\_blank()  
 ,panel.border = element\_blank()  
 ) +  
 theme(axis.line = element\_line(color = 'black')) + # draw x and y lines  
 theme(axis.title.x = element\_text(colour = 'black', size = 23,   
 margin=margin(15,15,15,15)),  
 axis.title.y = element\_text(colour = 'black', size = 23,   
 margin=margin(15,15,15,15)),  
 axis.text.x = element\_text(size=16),  
 axis.text.y = element\_text(size=16)) +  
 labs(x = "Rotated PCA factor 1", y = "Rotated PCA factor 2") +  
 theme(plot.title = element\_text(size = 32, face = "bold",   
 margin=margin(15,15,15,15)))

Engconcs # ! THE PLOT IS SHOWN BADLY ON HERE. PLEASE SEE THE SAVED PLOT



# Now to save, run first line below and return to keep running. See your folder.  
png(file="Engconcs\_highres.png", units="in", width=13, height=13, res=900)  
plot(Engconcs)  
dev.off()

## png   
## 2

# ON DUTCH NORMS  
  
# properties  
# check conditions for a PCA  
  
# matrix  
prop <- all[all$cat == 'prop', c('Auditory', 'Haptic', 'Visual')]  
nrow(prop)

## [1] 343

prop\_matrix <- cor(prop, use = 'complete.obs')  
prop\_matrix

## Auditory Haptic Visual  
## Auditory 1.0000000 -0.2280165 -0.5134304  
## Haptic -0.2280165 1.0000000 0.1930402  
## Visual -0.5134304 0.1930402 1.0000000

round(prop\_matrix, 2)

## Auditory Haptic Visual  
## Auditory 1.00 -0.23 -0.51  
## Haptic -0.23 1.00 0.19  
## Visual -0.51 0.19 1.00

# POOR: correlations not apt for a PCA, with too many below .3  
  
# now on the raw vars:  
nrow(prop)

## [1] 343

cortest.bartlett(prop)

## R was not square, finding R from data

## $chisq  
## [1] 125.0759  
##   
## $p.value  
## [1] 6.224181e-27  
##   
## $df  
## [1] 3

# GOOD: Bartlett's test significant   
  
# KMO: Kaiser-Meyer-Olkin Measure of Sampling Adequacy  
KMO(prop\_matrix)

## Kaiser-Meyer-Olkin factor adequacy  
## Call: KMO(r = prop\_matrix)  
## Overall MSA = 0.56  
## MSA for each item =   
## Auditory Haptic Visual   
## 0.54 0.74 0.55

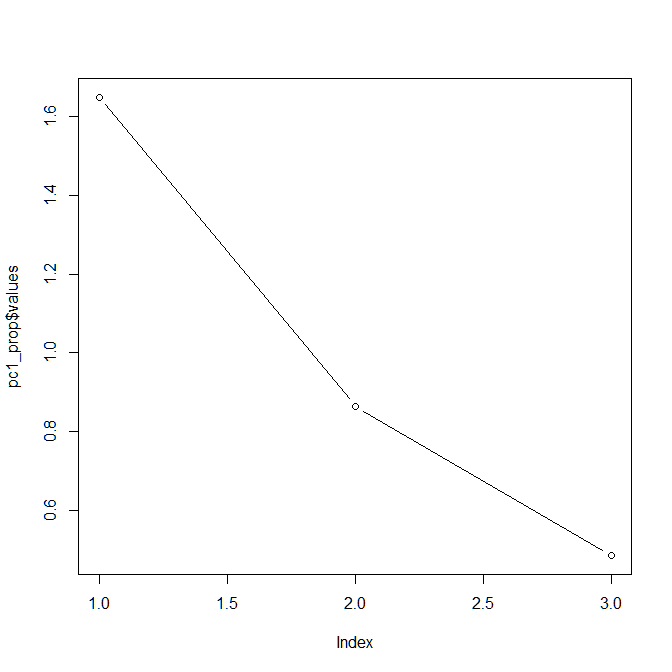
# Result: .56 = mediocre. PCA not strongly recommended. But we still do it  
# because the purpose is graphical only.  
  
# check determinant  
det(prop\_matrix)

## [1] 0.6923318

# GOOD: >0.00001  
  
# start off with unrotated PCA  
pc1\_prop <- psych::principal(prop, nfactors = 3, rotate = "none")  
pc1\_prop

## Principal Components Analysis  
## Call: psych::principal(r = prop, nfactors = 3, rotate = "none")  
## Standardized loadings (pattern matrix) based upon correlation matrix  
## PC1 PC2 PC3 h2 u2 com  
## Auditory -0.83 0.23 0.50 1 0.0e+00 1.8  
## Haptic 0.54 0.84 0.04 1 -1.6e-15 1.7  
## Visual 0.82 -0.31 0.48 1 -8.9e-16 2.0  
##   
## PC1 PC2 PC3  
## SS loadings 1.65 0.86 0.49  
## Proportion Var 0.55 0.29 0.16  
## Cumulative Var 0.55 0.84 1.00  
## Proportion Explained 0.55 0.29 0.16  
## Cumulative Proportion 0.55 0.84 1.00  
##   
## Mean item complexity = 1.8  
## Test of the hypothesis that 3 components are sufficient.  
##   
## The root mean square of the residuals (RMSR) is 0   
## with the empirical chi square 0 with prob < NA   
##   
## Fit based upon off diagonal values = 1

# RESULT: Only PC1, with eigenvalue > 1, should be extracted,   
# acc to Kaiser's criterion (Jolliffe's threshold of 0.7 way too lax;   
# Field, Miles, & Field, 2012)  
  
# Unrotated: scree plot  
plot(pc1\_prop$values, type = "b")



# Result: one or two RCs should be extracted, converging with eigenvalues  
  
# Now with varimax rotation, Kaiser-normalized (by default).   
# Always preferable because it captures explained variance best.   
# Compare eigenvalues w/ 1 & 2 factors  
  
pc2\_prop <- psych::principal(prop, nfactors = 2, rotate = "varimax", scores = TRUE)  
pc2\_prop

## Principal Components Analysis  
## Call: psych::principal(r = prop, nfactors = 2, rotate = "varimax",   
## scores = TRUE)  
## Standardized loadings (pattern matrix) based upon correlation matrix  
## RC1 RC2 h2 u2 com  
## Auditory -0.85 -0.16 0.75 0.2498 1.1  
## Haptic 0.11 0.99 1.00 0.0016 1.0  
## Visual 0.87 0.08 0.77 0.2337 1.0  
##   
## RC1 RC2  
## SS loadings 1.5 1.02  
## Proportion Var 0.5 0.34  
## Cumulative Var 0.5 0.84  
## Proportion Explained 0.6 0.40  
## Cumulative Proportion 0.6 1.00  
##   
## Mean item complexity = 1  
## Test of the hypothesis that 2 components are sufficient.  
##   
## The root mean square of the residuals (RMSR) is 0.14   
## with the empirical chi square 40.59 with prob < NA   
##   
## Fit based upon off diagonal values = 0.83

pc2\_prop$loadings

##   
## Loadings:  
## RC1 RC2   
## Auditory -0.852 -0.158  
## Haptic 0.107 0.993  
## Visual 0.872   
##   
## RC1 RC2  
## SS loadings 1.497 1.018  
## Proportion Var 0.499 0.339  
## Cumulative Var 0.499 0.838

# good to extract 2 factors, as they both explain quite the same variance,  
# and both surpass 1 eigenvalue  
  
  
pc2\_prop$residual

## Auditory Haptic Visual  
## Auditory 0.24984223 0.020051207 0.24163179  
## Haptic 0.02005121 0.001609219 0.01939227  
## Visual 0.24163179 0.019392274 0.23369117

pc2\_prop$fit

## [1] 0.9364867

pc2\_prop$communality

## Auditory Haptic Visual   
## 0.7501578 0.9983908 0.7663088

# Results based on a Kaiser-normalizalized orthogonal (varimax) rotation  
# (by default in psych::stats). Residuals OK: fewer than 50% have absolute   
# values > 0.05 (exactly 50% do).Model fit good, > .90.   
# Communalities good, all > .7 (av = .83).   
  
# subset and add PCs  
props <- all[all$cat == 'prop', ]  
nrow(props)

## [1] 343

props <- cbind(props, pc2\_prop$scores)  
nrow(props)

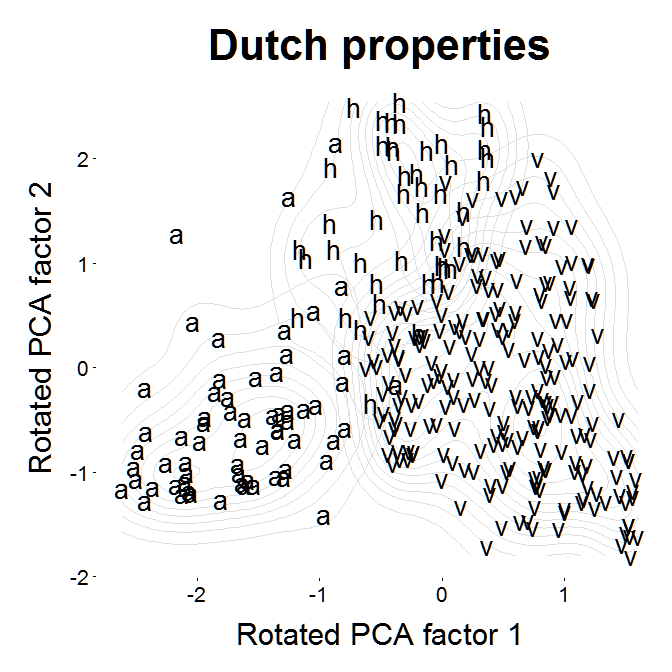
## [1] 343

# Finally, plot: letters+density (cf. Lynott & Connell, 2009, 2013)  
  
NLprops <- ggplot(props,  
 aes(RC1, RC2, label = as.character(main))) +  
 aes (x = RC1, y = RC2, by = main) + stat\_density2d (color = "gray87") +  
 geom\_text(size = 7) +  
 ggtitle ('Dutch properties') +  
 theme\_bw() + # theme with white background  
 theme( # clear background, gridlines, chart border  
 plot.background = element\_blank()  
 ,panel.grid.major = element\_blank()  
 ,panel.grid.minor = element\_blank()  
 ,panel.border = element\_blank()  
 ) +  
 theme(axis.line = element\_line(color = 'black')) + # draw x and y lines  
 theme(axis.title.x = element\_text(colour = 'black', size = 23,   
 margin=margin(15,15,15,15)),  
 axis.title.y = element\_text(colour = 'black', size = 23,   
 margin=margin(15,15,15,15)),  
 axis.text.x = element\_text(size=16),  
 axis.text.y = element\_text(size=16)) +  
 labs(x = "Rotated PCA factor 1", y = "Rotated PCA factor 2") +  
 theme(plot.title = element\_text(size = 32, face = "bold",   
 margin=margin(15,15,15,15)))

NLprops # ! THE PLOT IS SHOWN BADLY ON HERE. PLEASE SEE THE SAVED PLOT

## Warning: Removed 7 rows containing non-finite values (stat\_density2d).

## Warning: Removed 7 rows containing missing values (geom\_text).



# Now to save, run first line below and return to keep running. See your folder.  
png(file="NLprops\_highres.png", units="in", width=13, height=13, res=900)  
plot(NLprops)

## Warning: Removed 7 rows containing non-finite values (stat\_density2d).  
  
## Warning: Removed 7 rows containing missing values (geom\_text).

# warning normal: just removing English properties not used in Dutch  
dev.off()

## png   
## 2

# Adjust for combined plots:  
  
NLprops2 <- ggplot(props,  
 aes(RC1, RC2, label = as.character(main))) +  
 aes (x = RC1, y = RC2, by = main) + stat\_density2d (color = "gray87") +  
 geom\_text(size = 7) +  
 ggtitle ('Dutch properties') +  
 theme\_bw() + # theme with white background  
 theme( # clear background, gridlines, chart border  
 plot.background = element\_blank()  
 ,panel.grid.major = element\_blank()  
 ,panel.grid.minor = element\_blank()  
 ,panel.border = element\_blank()  
 ) +  
 theme(axis.line = element\_line(color = 'black')) + # draw x and y lines  
 theme(axis.title.x = element\_text(colour = 'black', size = 23,   
 margin=margin(15,15,15,15)),  
 axis.title.y = element\_text(colour = 'black', size = 23,   
 margin=margin(15,15,15,15)),  
 axis.text.x = element\_text(size=16),  
 axis.text.y = element\_text(size=16)) +  
 labs(x = "Rotated PCA factor 1", y = "") +  
 theme(plot.title = element\_text(size = 32, face = "bold",   
 margin=margin(15,15,15,15)))  
  
# Next:  
  
NLprops4 <- ggplot(props,  
 aes(RC1, RC2, label = as.character(main))) +  
 aes (x = RC1, y = RC2, by = main) + stat\_density2d (color = "gray87") +  
 geom\_text(size = 7) +  
 ggtitle ('Dutch properties') +  
 theme\_bw() + # theme with white background  
 theme( # clear background, gridlines, chart border  
 plot.background = element\_blank()  
 ,panel.grid.major = element\_blank()  
 ,panel.grid.minor = element\_blank()  
 ,panel.border = element\_blank()  
 ) +  
 theme(axis.line = element\_line(color = 'black')) + # draw x and y lines  
 theme(axis.title.x = element\_text(colour = 'black', size = 23,   
 margin=margin(15,15,15,15)),  
 axis.title.y = element\_text(colour = 'black', size = 23,   
 margin=margin(15,15,15,15)),  
 axis.text.x = element\_text(size=16),  
 axis.text.y = element\_text(size=16)) +  
 labs(x = "", y = "") +  
 theme(plot.title = element\_text(size = 32, face = "bold",   
 margin=margin(15,15,15,15)))  
  
  
  
# CONCEPTS  
# check conditions for a PCA  
# matrix  
conc <- all[all$cat == 'conc', c('Auditory', 'Haptic', 'Visual')]  
nrow(conc)

## [1] 416

conc\_matrix <- cor(conc, use = 'complete.obs')  
conc\_matrix

## Auditory Haptic Visual  
## Auditory 1.000000000 -0.008508063 0.08486561  
## Haptic -0.008508063 1.000000000 0.44144835  
## Visual 0.084865608 0.441448353 1.00000000

round(conc\_matrix, 2)

## Auditory Haptic Visual  
## Auditory 1.00 -0.01 0.08  
## Haptic -0.01 1.00 0.44  
## Visual 0.08 0.44 1.00

# POOR: correlations not apt for a PCA, with too many below .3  
  
# now on the raw data:  
nrow(conc)

## [1] 416

cortest.bartlett(conc)

## R was not square, finding R from data

## $chisq  
## [1] 93.63824  
##   
## $p.value  
## [1] 3.621992e-20  
##   
## $df  
## [1] 3

# GOOD: Bartlett's test significant   
  
# KMO: Kaiser-Meyer-Olkin Measure of Sampling Adequacy  
KMO(conc\_matrix)

## Kaiser-Meyer-Olkin factor adequacy  
## Call: KMO(r = conc\_matrix)  
## Overall MSA = 0.49  
## MSA for each item =   
## Auditory Haptic Visual   
## 0.37 0.49 0.49

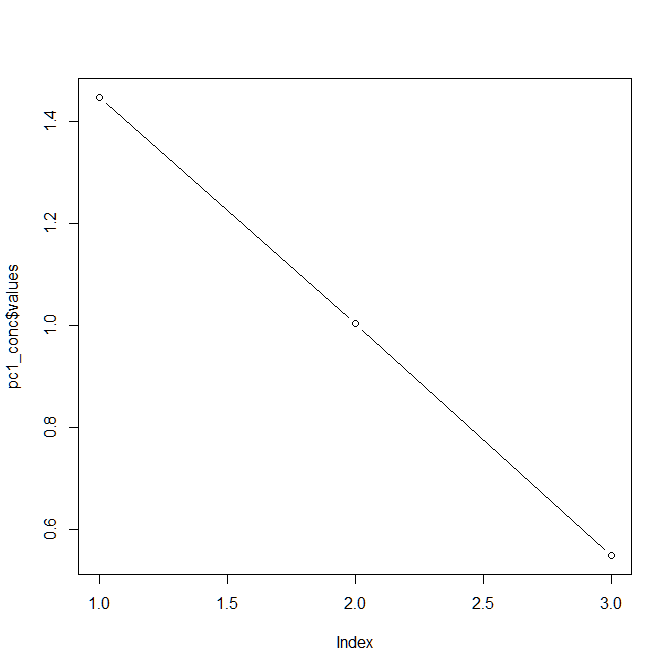
# Result: .49 = poor. PCA not strongly recommended. But we still do it  
# because the purpose is graphical really.  
  
# check determinant  
det(conc\_matrix)

## [1] 0.7972113

# GOOD: >0.00001  
  
# start off with unrotated PCA  
pc1\_conc <- psych::principal(conc, nfactors = 3, rotate = "none")  
pc1\_conc

## Principal Components Analysis  
## Call: psych::principal(r = conc, nfactors = 3, rotate = "none")  
## Standardized loadings (pattern matrix) based upon correlation matrix  
## PC1 PC2 PC3 h2 u2 com  
## Auditory 0.15 0.98 0.11 1 1.1e-16 1.1  
## Haptic 0.84 -0.19 0.51 1 4.4e-16 1.8  
## Visual 0.85 0.02 -0.52 1 0.0e+00 1.7  
##   
## PC1 PC2 PC3  
## SS loadings 1.45 1.00 0.55  
## Proportion Var 0.48 0.33 0.18  
## Cumulative Var 0.48 0.82 1.00  
## Proportion Explained 0.48 0.33 0.18  
## Cumulative Proportion 0.48 0.82 1.00  
##   
## Mean item complexity = 1.5  
## Test of the hypothesis that 3 components are sufficient.  
##   
## The root mean square of the residuals (RMSR) is 0   
## with the empirical chi square 0 with prob < NA   
##   
## Fit based upon off diagonal values = 1

# RESULT good: PC1 and PC2, with eigenvalue > 1, should be extracted,   
# acc to Kaiser's criterion (Jolliffe's threshold of 0.7 way too lax;   
# Field, Miles, & Field, 2012)  
  
# Unrotated: scree plot  
plot(pc1\_conc$values, type = "b")



# Result: with no point of inflexion along the y axis, two PCs would obtain.  
  
# Now with varimax rotation, Kaiser-normalized (by default):  
# Always preferable because it captures explained variance best.   
# Compare eigenvalues w/ 1 & 2 factors  
  
pc2\_conc <- psych::principal(conc, nfactors = 2, rotate = "varimax", scores = TRUE)  
pc2\_conc

## Principal Components Analysis  
## Call: psych::principal(r = conc, nfactors = 2, rotate = "varimax",   
## scores = TRUE)  
## Standardized loadings (pattern matrix) based upon correlation matrix  
## RC1 RC2 h2 u2 com  
## Auditory 0.03 0.99 0.99 0.012 1  
## Haptic 0.85 -0.09 0.74 0.264 1  
## Visual 0.84 0.12 0.73 0.273 1  
##   
## RC1 RC2  
## SS loadings 1.44 1.01  
## Proportion Var 0.48 0.34  
## Cumulative Var 0.48 0.82  
## Proportion Explained 0.59 0.41  
## Cumulative Proportion 0.59 1.00  
##   
## Mean item complexity = 1  
## Test of the hypothesis that 2 components are sufficient.  
##   
## The root mean square of the residuals (RMSR) is 0.16   
## with the empirical chi square 65.21 with prob < NA   
##   
## Fit based upon off diagonal values = 0.61

pc2\_conc$loadings

##   
## Loadings:  
## RC1 RC2   
## Auditory 0.994  
## Haptic 0.854   
## Visual 0.844 0.120  
##   
## RC1 RC2  
## SS loadings 1.442 1.010  
## Proportion Var 0.481 0.337  
## Cumulative Var 0.481 0.817

# good to extract 2 factors, as they both explain quite the same variance,   
# and both surpass 1 eigenvalue  
  
pc2\_conc$residual

## Auditory Haptic Visual  
## Auditory 0.01167046 0.0554842 -0.05648251  
## Haptic 0.05548420 0.2637854 -0.26853166  
## Visual -0.05648251 -0.2685317 0.27336330

pc2\_conc$fit

## [1] 0.911523

pc2\_conc$communality

## Auditory Haptic Visual   
## 0.9883295 0.7362146 0.7266367

# Results based on a Kaiser-normalizalized orthogonal (varimax) rotation  
# (by default in psych::stats). Residuals bad: over 50% have absolute   
# values > 0.05. Model fit good, > .90. Communalities good, all > .7 (av = .82).   
  
# subset and add PCs  
concs <- all[all$cat == 'conc', ]  
nrow(concs)

## [1] 416

concs <- cbind(concs, pc2\_conc$scores)  
nrow(concs)

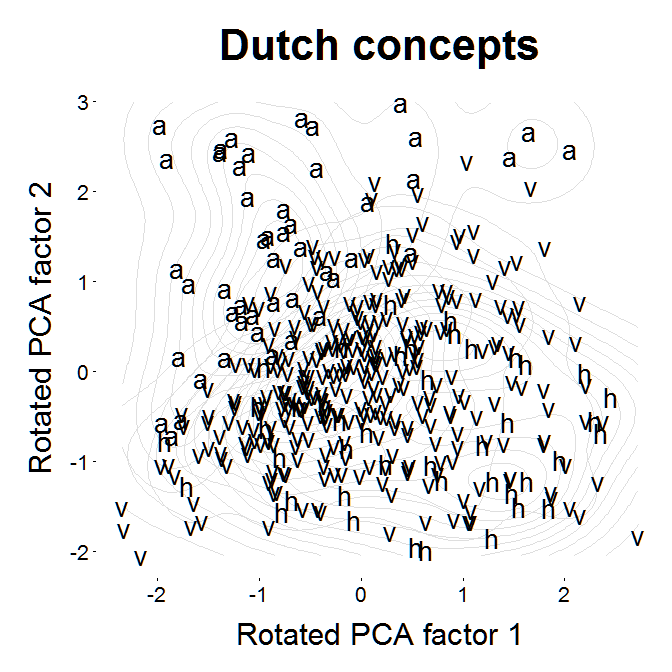
## [1] 416

# Finally, plot  
NLconcs <- ggplot(concs,  
 aes(RC1, RC2, label = as.character(main))) +  
 aes (x = RC1, y = RC2, by = main) + stat\_density2d (color = "gray87") +  
 geom\_text(size = 7) +  
 ggtitle ('Dutch concepts') +  
 theme\_bw() + # theme with white background  
 theme( # clear background, gridlines, chart border  
 plot.background = element\_blank()  
 ,panel.grid.major = element\_blank()  
 ,panel.grid.minor = element\_blank()  
 ,panel.border = element\_blank()  
 ) +  
 theme(axis.line = element\_line(color = 'black')) + # draw x and y lines  
 theme(axis.title.x = element\_text(colour = 'black', size = 23,   
 margin=margin(15,15,15,15)),  
 axis.title.y = element\_text(colour = 'black', size = 23,   
 margin=margin(15,15,15,15)),  
 axis.text.x = element\_text(size=16),  
 axis.text.y = element\_text(size=16)) +  
 labs(x = "Rotated PCA factor 1", y = "Rotated PCA factor 2") +  
 theme(plot.title = element\_text(size = 32, face = "bold",   
 margin=margin(15,15,15,15)))

NLconcs # ! THE PLOT IS SHOWN BADLY ON HERE. PLEASE SEE THE SAVED PLOT

## Warning: Removed 5 rows containing non-finite values (stat\_density2d).

## Warning: Removed 5 rows containing missing values (geom\_text).



# Now to save, run first line below and return to keep running. See your folder.  
png(file="NLconcs\_highres.png", units="in", width=13, height=13, res=900)  
plot(NLconcs)

## Warning: Removed 5 rows containing non-finite values (stat\_density2d).  
  
## Warning: Removed 5 rows containing missing values (geom\_text).

# warning normal: just removing English concepts not used in Dutch  
dev.off()

## png   
## 2

# Adjust for combined plots:  
  
NLconcs2 <- ggplot(concs,  
 aes(RC1, RC2, label = as.character(main))) +  
 aes (x = RC1, y = RC2, by = main) + stat\_density2d (color = "gray87") +  
 geom\_text(size = 7) +  
 ggtitle ('Dutch concepts') +  
 theme\_bw() + # theme with white background  
 theme( # clear background, gridlines, chart border  
 plot.background = element\_blank()  
 ,panel.grid.major = element\_blank()  
 ,panel.grid.minor = element\_blank()  
 ,panel.border = element\_blank()  
 ) +  
 theme(axis.line = element\_line(color = 'black')) + # draw x and y lines  
 theme(axis.title.x = element\_text(colour = 'black', size = 23,   
 margin=margin(15,15,15,15)),  
 axis.title.y = element\_text(colour = 'black', size = 23,   
 margin=margin(15,15,15,15)),  
 axis.text.x = element\_text(size=16),  
 axis.text.y = element\_text(size=16)) +  
 labs(x = "Rotated PCA factor 1", y = "") +  
 theme(plot.title = element\_text(size = 32, face = "bold",   
 margin=margin(15,15,15,15)))  
  
  
  
# Combined plots:  
  
# Below, run first line, get back and run next.   
# High resolution (may be changed at 'res='). Beware of high memory usage.  
  
png(file="allfour\_highres.png", units="in", width=19, height=19, res=1200)  
multiplot(Engprops4, Engconcs, NLprops4, NLconcs2, cols = 2)

## Warning: Removed 7 rows containing non-finite values (stat\_density2d).

## Warning: Removed 7 rows containing missing values (geom\_text).

## Warning: Removed 5 rows containing non-finite values (stat\_density2d).

## Warning: Removed 5 rows containing missing values (geom\_text).

# warning normal: just those English items that were not used in Dutch  
dev.off()

## png   
## 2

png(file="proppair\_highres.png", units="in", width=18, height=9, res=1000)  
multiplot(Engprops, NLprops2, cols = 2)

## Warning: Removed 7 rows containing non-finite values (stat\_density2d).

## Warning: Removed 7 rows containing missing values (geom\_text).

# warning normal: just those English items that were not used in Dutch  
dev.off()

## png   
## 2

png(file="concpair\_highres.png", units="in", width=18, height=9, res=1000)  
multiplot(Engconcs, NLconcs2, cols = 2)

## Warning: Removed 5 rows containing non-finite values (stat\_density2d).

## Warning: Removed 5 rows containing missing values (geom\_text).

# warning normal: just those English items that were not used in Dutch  
dev.off()

## png   
## 2

# Find all plots in your working directory  
  
# With a naked eye, one can see the different relationships. The significance of   
# these comparisons is notable. First, it demonstrates visually the difference   
# between modality exclusivity and each of the modality strengths (which of course   
# is only natural considering how modality exclusivity was calculated). The two   
# variables then must be different indeed because in the exclusivity analysis, the   
# visual and the auditory modalities were the most similar ones, with their higher   
# exclusivities. In contrast, in the independent strengths analysis, the visual and   
# the haptic modalities show a clear interlock, which leaves the auditory experience   
# rather on its own.  
# \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
  
  
  
# ICONICITY  
  
# Last tests: iconicity/sound symbolism on concepts and properties separately.  
# Regressions include same lexical vars (DVs) as Lynott and Connell, plus   
# concreteness and age of acquisition.  
  
# Note that the selection is based on p-value thresholds, as in L&C, but also on  
# AIC, which is a bayesian, relative method more appropriate with such a large  
# sample. Importantly, AIC and F/p-value criteria resulted in the same inclusions   
# and exclusions for every regression.  
  
# For both props and concs, we start with PCA with all lexical variables in order  
# to isolate them, because they are intercorrelated (see Table 5 in Lynott & Connell,  
# 2013)  
  
all <- read.csv('all.csv')  
nrow(all)

## [1] 759

# Length is 759 but only 747 are from these norms. Rest are from Lynott and Connell   
# (2009, 2013) for comparative analyses. These extra items do not have an id number   
# in the file.

# Iconicity within properties alone, as in Lynott and Connell (2013). As a novelty,   
# the iconicity analysis is hereby performed also on the Dutch properties, in   
# addition to the concepts.  
  
props <- subset(all, subset = cat == 'prop')  
nrow(props)

## [1] 343

# There aren't lexical data for every single word.  
# Nr of properties per lexical variable (from the Dutch items only of course)  
describe(complete.cases(props[complete.cases(props$Exclusivity),]  
$phonemes\_DUTCHPOND))

## complete.cases(props[complete.cases(props$Exclusivity), ]$phonemes\_DUTCHPOND)   
## n missing unique   
## 336 0 2   
##   
## FALSE (151, 45%), TRUE (185, 55%)

describe(complete.cases(props[complete.cases(props$Exclusivity),]  
$phon\_neighbours\_DUTCHPOND))

## complete.cases(props[complete.cases(props$Exclusivity), ]$phon\_neighbours\_DUTCHPOND)   
## n missing unique value   
## 336 0 1 TRUE

describe(complete.cases(props[complete.cases(props$Exclusivity),]  
$orth\_neighbours\_DUTCHPOND))

## complete.cases(props[complete.cases(props$Exclusivity), ]$orth\_neighbours\_DUTCHPOND)   
## n missing unique value   
## 336 0 1 TRUE

describe(complete.cases(props[complete.cases(props$Exclusivity),]  
$freq\_lg10CD\_SUBTLEXNL))

## complete.cases(props[complete.cases(props$Exclusivity), ]$freq\_lg10CD\_SUBTLEXNL)   
## n missing unique   
## 336 0 2   
##   
## FALSE (46, 14%), TRUE (290, 86%)

describe(complete.cases(props[complete.cases(props$Exclusivity),]  
$freq\_lg10WF\_SUBTLEXNL))

## complete.cases(props[complete.cases(props$Exclusivity), ]$freq\_lg10WF\_SUBTLEXNL)   
## n missing unique   
## 336 0 2   
##   
## FALSE (46, 14%), TRUE (290, 86%)

describe(complete.cases(props[complete.cases(props$Exclusivity),]  
$freq\_CELEX\_lem))

## complete.cases(props[complete.cases(props$Exclusivity), ]$freq\_CELEX\_lem)   
## n missing unique   
## 336 0 2   
##   
## FALSE (89, 26%), TRUE (247, 74%)

describe(complete.cases(props[complete.cases(props$Exclusivity),]  
$AoA\_Brysbaertetal2014))

## complete.cases(props[complete.cases(props$Exclusivity), ]$AoA\_Brysbaertetal2014)   
## n missing unique   
## 336 0 2   
##   
## FALSE (103, 31%), TRUE (233, 69%)

describe(complete.cases(props[complete.cases(props$Exclusivity),]  
$concrete\_Brysbaertetal2014))

## complete.cases(props[complete.cases(props$Exclusivity), ]$concrete\_Brysbaertetal2014)   
## n missing unique   
## 336 0 2   
##   
## FALSE (103, 31%), TRUE (233, 69%)

# M, SD  
stat.desc(props$letters)

## nbr.val nbr.null nbr.na min max   
## 336.0000000 0.0000000 7.0000000 3.0000000 14.0000000   
## range sum median mean SE.mean   
## 11.0000000 2391.0000000 7.0000000 7.1160714 0.1234165   
## CI.mean.0.95 var std.dev coef.var   
## 0.2427690 5.1178305 2.2622623 0.3179089

stat.desc(props$phonemes\_DUTCHPOND)

## nbr.val nbr.null nbr.na min max   
## 185.0000000 0.0000000 158.0000000 2.0000000 11.0000000   
## range sum median mean SE.mean   
## 9.0000000 996.0000000 5.0000000 5.3837838 0.1433762   
## CI.mean.0.95 var std.dev coef.var   
## 0.2828727 3.8029965 1.9501273 0.3622224

stat.desc(props$phon\_neighbours\_DUTCHPOND)

## nbr.val nbr.null nbr.na min max   
## 336.0000000 106.0000000 7.0000000 0.0000000 42.0000000   
## range sum median mean SE.mean   
## 42.0000000 1536.0000000 1.0000000 4.5714286 0.4126023   
## CI.mean.0.95 var std.dev coef.var   
## 0.8116178 57.2008529 7.5631245 1.6544335

stat.desc(props$orth\_neighbours\_DUTCHPOND)

## nbr.val nbr.null nbr.na min max   
## 336.0000000 102.0000000 7.0000000 0.0000000 24.0000000   
## range sum median mean SE.mean   
## 24.0000000 1115.0000000 1.0000000 3.3184524 0.2655425   
## CI.mean.0.95 var std.dev coef.var   
## 0.5223409 23.6923152 4.8674752 1.4667907

stat.desc(props$freq\_lg10CD\_SUBTLEXNL)

## nbr.val nbr.null nbr.na min max   
## 290.00000000 0.00000000 53.00000000 0.30000000 3.86000000   
## range sum median mean SE.mean   
## 3.56000000 521.89000000 1.61000000 1.79962069 0.05958115   
## CI.mean.0.95 var std.dev coef.var   
## 0.11726801 1.02947494 1.01463045 0.56380239

stat.desc(props$freq\_lg10WF\_SUBTLEXNL)

## nbr.val nbr.null nbr.na min max   
## 290.00000000 0.00000000 53.00000000 0.30000000 4.64000000   
## range sum median mean SE.mean   
## 4.34000000 545.51000000 1.69000000 1.88106897 0.06419049   
## CI.mean.0.95 var std.dev coef.var   
## 0.12634014 1.19492169 1.09312474 0.58111890

stat.desc(props$freq\_CELEX\_lem)

## nbr.val nbr.null nbr.na min max   
## 247.00000000 40.00000000 96.00000000 0.00000000 3.08600000   
## range sum median mean SE.mean   
## 3.08600000 266.89000000 0.95400000 1.08052632 0.05081189   
## CI.mean.0.95 var std.dev coef.var   
## 0.10008186 0.63771661 0.79857160 0.73905799

stat.desc(props$AoA\_Brysbaertetal2014)

## nbr.val nbr.null nbr.na min max   
## 233.0000000 0.0000000 110.0000000 3.9100000 14.0800000   
## range sum median mean SE.mean   
## 10.1700000 1857.6700000 8.0900000 7.9728326 0.1394971   
## CI.mean.0.95 var std.dev coef.var   
## 0.2748431 4.5340523 2.1293314 0.2670734

stat.desc(props$concrete\_Brysbaertetal2014)

## nbr.val nbr.null nbr.na min max   
## 233.00000000 0.00000000 110.00000000 1.33000000 4.67000000   
## range sum median mean SE.mean   
## 3.34000000 761.91000000 3.40000000 3.27000000 0.04601172   
## CI.mean.0.95 var std.dev coef.var   
## 0.09065422 0.49327931 0.70233846 0.21478240

# See and print correlation of all lexical variables:  
  
mat\_lexicals\_props <- as.matrix(props[c('letters', 'phonemes\_DUTCHPOND',   
'orth\_neighbours\_DUTCHPOND', 'phon\_neighbours\_DUTCHPOND', 'freq\_lg10CD\_SUBTLEXNL',   
'freq\_lg10WF\_SUBTLEXNL', 'freq\_CELEX\_lem', 'AoA\_Brysbaertetal2014',   
'concrete\_Brysbaertetal2014')])  
  
rcor.test(mat\_lexicals\_props, use='complete.obs')

##   
## letters phonemes\_DUTCHPOND  
## letters \*\*\*\*\* 0.940   
## phonemes\_DUTCHPOND <0.001 \*\*\*\*\*   
## orth\_neighbours\_DUTCHPOND <0.001 <0.001   
## phon\_neighbours\_DUTCHPOND <0.001 <0.001   
## freq\_lg10CD\_SUBTLEXNL <0.001 <0.001   
## freq\_lg10WF\_SUBTLEXNL <0.001 <0.001   
## freq\_CELEX\_lem <0.001 <0.001   
## AoA\_Brysbaertetal2014 <0.001 <0.001   
## concrete\_Brysbaertetal2014 0.300 0.187   
## orth\_neighbours\_DUTCHPOND  
## letters -0.727   
## phonemes\_DUTCHPOND -0.716   
## orth\_neighbours\_DUTCHPOND \*\*\*\*\*   
## phon\_neighbours\_DUTCHPOND <0.001   
## freq\_lg10CD\_SUBTLEXNL <0.001   
## freq\_lg10WF\_SUBTLEXNL <0.001   
## freq\_CELEX\_lem <0.001   
## AoA\_Brysbaertetal2014 <0.001   
## concrete\_Brysbaertetal2014 0.168   
## phon\_neighbours\_DUTCHPOND freq\_lg10CD\_SUBTLEXNL  
## letters -0.703 -0.508   
## phonemes\_DUTCHPOND -0.732 -0.486   
## orth\_neighbours\_DUTCHPOND 0.895 0.467   
## phon\_neighbours\_DUTCHPOND \*\*\*\*\* 0.477   
## freq\_lg10CD\_SUBTLEXNL <0.001 \*\*\*\*\*   
## freq\_lg10WF\_SUBTLEXNL <0.001 <0.001   
## freq\_CELEX\_lem <0.001 <0.001   
## AoA\_Brysbaertetal2014 <0.001 <0.001   
## concrete\_Brysbaertetal2014 0.060 0.088   
## freq\_lg10WF\_SUBTLEXNL freq\_CELEX\_lem  
## letters -0.509 -0.550   
## phonemes\_DUTCHPOND -0.486 -0.555   
## orth\_neighbours\_DUTCHPOND 0.470 0.518   
## phon\_neighbours\_DUTCHPOND 0.478 0.517   
## freq\_lg10CD\_SUBTLEXNL 0.995 0.838   
## freq\_lg10WF\_SUBTLEXNL \*\*\*\*\* 0.832   
## freq\_CELEX\_lem <0.001 \*\*\*\*\*   
## AoA\_Brysbaertetal2014 <0.001 <0.001   
## concrete\_Brysbaertetal2014 0.097 0.270   
## AoA\_Brysbaertetal2014  
## letters 0.405   
## phonemes\_DUTCHPOND 0.457   
## orth\_neighbours\_DUTCHPOND -0.417   
## phon\_neighbours\_DUTCHPOND -0.438   
## freq\_lg10CD\_SUBTLEXNL -0.654   
## freq\_lg10WF\_SUBTLEXNL -0.646   
## freq\_CELEX\_lem -0.700   
## AoA\_Brysbaertetal2014 \*\*\*\*\*   
## concrete\_Brysbaertetal2014 <0.001   
## concrete\_Brysbaertetal2014  
## letters -0.090   
## phonemes\_DUTCHPOND -0.090   
## orth\_neighbours\_DUTCHPOND 0.118   
## phon\_neighbours\_DUTCHPOND 0.156   
## freq\_lg10CD\_SUBTLEXNL -0.166   
## freq\_lg10WF\_SUBTLEXNL -0.161   
## freq\_CELEX\_lem -0.100   
## AoA\_Brysbaertetal2014 -0.254   
## concrete\_Brysbaertetal2014 \*\*\*\*\*   
##   
## upper diagonal part contains correlation coefficient estimates   
## lower diagonal part contains corresponding p-values

corrs\_props = rcor.test(mat\_lexicals\_props, use='complete.obs')  
write.csv(corrs\_props$cor.mat, file = "corrs\_props.csv",na="") # find table in folder  
# (saved just for the manuscript)  
  
  
# go on to PCA. This does not include age of acquisition or concreteness for a   
# better comparison with the English data, and because no correlations > .7 (i.e. half  
# of variance explained)  
  
lexicals\_props <- props[c('letters', 'phonemes\_DUTCHPOND', 'orth\_neighbours\_DUTCHPOND',   
'phon\_neighbours\_DUTCHPOND', 'freq\_lg10CD\_SUBTLEXNL', 'freq\_lg10WF\_SUBTLEXNL',   
'freq\_CELEX\_lem')]  
  
str(lexicals\_props)

## 'data.frame': 343 obs. of 7 variables:  
## $ letters : int 5 11 5 8 7 5 11 10 10 7 ...  
## $ phonemes\_DUTCHPOND : int 4 NA 5 NA 6 4 NA 10 NA 7 ...  
## $ orth\_neighbours\_DUTCHPOND: int 6 1 1 2 6 1 0 2 1 1 ...  
## $ phon\_neighbours\_DUTCHPOND: int 4 1 1 2 3 1 0 1 1 1 ...  
## $ freq\_lg10CD\_SUBTLEXNL : num 1.53 0.3 1.97 0.3 1.45 1.76 1.26 2.06 0.3 1.79 ...  
## $ freq\_lg10WF\_SUBTLEXNL : num 1.71 0.3 2.55 0.3 1.52 1.88 1.26 2.14 0.3 1.84 ...  
## $ freq\_CELEX\_lem : num 1.431 NA 0 NA 0.699 ...

# start with PCA for lexical variables, done as in Lynott and Connell (2013)  
# Check conditions for a PCA  
# Correlations  
  
cor(lexicals\_props, use = 'complete.obs')

## letters phonemes\_DUTCHPOND  
## letters 1.0000000 0.9410143  
## phonemes\_DUTCHPOND 0.9410143 1.0000000  
## orth\_neighbours\_DUTCHPOND -0.7263499 -0.7167381  
## phon\_neighbours\_DUTCHPOND -0.7002995 -0.7304257  
## freq\_lg10CD\_SUBTLEXNL -0.5166005 -0.4955464  
## freq\_lg10WF\_SUBTLEXNL -0.5177211 -0.4961595  
## freq\_CELEX\_lem -0.5569857 -0.5623170  
## orth\_neighbours\_DUTCHPOND  
## letters -0.7263499  
## phonemes\_DUTCHPOND -0.7167381  
## orth\_neighbours\_DUTCHPOND 1.0000000  
## phon\_neighbours\_DUTCHPOND 0.8960621  
## freq\_lg10CD\_SUBTLEXNL 0.4715896  
## freq\_lg10WF\_SUBTLEXNL 0.4739747  
## freq\_CELEX\_lem 0.5187955  
## phon\_neighbours\_DUTCHPOND freq\_lg10CD\_SUBTLEXNL  
## letters -0.7002995 -0.5166005  
## phonemes\_DUTCHPOND -0.7304257 -0.4955464  
## orth\_neighbours\_DUTCHPOND 0.8960621 0.4715896  
## phon\_neighbours\_DUTCHPOND 1.0000000 0.4814090  
## freq\_lg10CD\_SUBTLEXNL 0.4814090 1.0000000  
## freq\_lg10WF\_SUBTLEXNL 0.4826222 0.9947196  
## freq\_CELEX\_lem 0.5185262 0.8423256  
## freq\_lg10WF\_SUBTLEXNL freq\_CELEX\_lem  
## letters -0.5177211 -0.5569857  
## phonemes\_DUTCHPOND -0.4961595 -0.5623170  
## orth\_neighbours\_DUTCHPOND 0.4739747 0.5187955  
## phon\_neighbours\_DUTCHPOND 0.4826222 0.5185262  
## freq\_lg10CD\_SUBTLEXNL 0.9947196 0.8423256  
## freq\_lg10WF\_SUBTLEXNL 1.0000000 0.8357985  
## freq\_CELEX\_lem 0.8357985 1.0000000

# Result: all variables fit for PCA, as they have few scores below .3   
# The correlations broadly replicate Lynott and Connell.   
  
# now on the raw vars:  
cortest.bartlett(lexicals\_props)

## R was not square, finding R from data

## $chisq  
## [1] 4265.57  
##   
## $p.value  
## [1] 0  
##   
## $df  
## [1] 21

# GOOD: Bartlett's test significant   
  
# KMO: Kaiser-Meyer-Olkin Measure of Sampling Adequacy  
lexicals\_props\_matrix <- cor(lexicals\_props, use = 'complete.obs')  
KMO(lexicals\_props\_matrix)

## Kaiser-Meyer-Olkin factor adequacy  
## Call: KMO(r = lexicals\_props\_matrix)  
## Overall MSA = 0.78  
## MSA for each item =   
## letters phonemes\_DUTCHPOND   
## 0.76 0.75   
## orth\_neighbours\_DUTCHPOND phon\_neighbours\_DUTCHPOND   
## 0.78 0.77   
## freq\_lg10CD\_SUBTLEXNL freq\_lg10WF\_SUBTLEXNL   
## 0.73 0.73   
## freq\_CELEX\_lem   
## 0.97

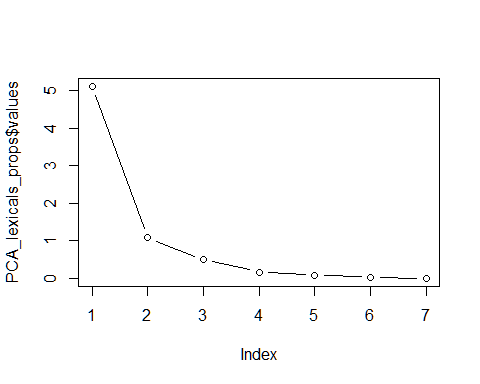
# Result: .78 = good.  
  
# determinant  
det(lexicals\_props\_matrix)

## [1] 1.766926e-05

# GOOD: above 0.00001  
  
# start off with unrotated PCA  
  
PCA\_lexicals\_props <- psych::principal(lexicals\_props, nfactors = 7, scores = TRUE)  
PCA\_lexicals\_props

## Principal Components Analysis  
## Call: psych::principal(r = lexicals\_props, nfactors = 7, scores = TRUE)  
## Standardized loadings (pattern matrix) based upon correlation matrix  
## RC4 RC3 RC1 RC2 RC5 RC6 RC7 h2  
## letters -0.35 0.86 -0.33 -0.03 -0.04 0.15 0.00 1  
## phonemes\_DUTCHPOND -0.23 0.88 -0.38 -0.11 0.04 -0.14 0.00 1  
## orth\_neighbours\_DUTCHPOND 0.28 -0.37 0.86 0.09 0.20 -0.02 0.00 1  
## phon\_neighbours\_DUTCHPOND 0.30 -0.36 0.86 0.06 -0.20 0.02 0.00 1  
## freq\_lg10CD\_SUBTLEXNL 0.94 -0.25 0.24 0.02 0.00 0.00 -0.04 1  
## freq\_lg10WF\_SUBTLEXNL 0.94 -0.25 0.24 0.01 0.00 -0.01 0.04 1  
## freq\_CELEX\_lem 0.78 -0.25 0.29 0.49 0.00 0.00 0.00 1  
## u2 com  
## letters 1.1e-16 1.7  
## phonemes\_DUTCHPOND 1.6e-15 1.6  
## orth\_neighbours\_DUTCHPOND 1.2e-15 1.8  
## phon\_neighbours\_DUTCHPOND 1.4e-15 1.8  
## freq\_lg10CD\_SUBTLEXNL 1.4e-15 1.3  
## freq\_lg10WF\_SUBTLEXNL 1.6e-15 1.3  
## freq\_CELEX\_lem 1.6e-15 2.2  
##   
## RC4 RC3 RC1 RC2 RC5 RC6 RC7  
## SS loadings 2.71 1.98 1.92 0.27 0.08 0.04 0  
## Proportion Var 0.39 0.28 0.27 0.04 0.01 0.01 0  
## Cumulative Var 0.39 0.67 0.94 0.98 0.99 1.00 1  
## Proportion Explained 0.39 0.28 0.27 0.04 0.01 0.01 0  
## Cumulative Proportion 0.39 0.67 0.94 0.98 0.99 1.00 1  
##   
## Mean item complexity = 1.7  
## Test of the hypothesis that 7 components are sufficient.  
##   
## The root mean square of the residuals (RMSR) is 0   
## with the empirical chi square 0 with prob < NA   
##   
## Fit based upon off diagonal values = 1

# By all standards, extract 3 components  
  
  
# scree analysis  
plot(PCA\_lexicals\_props$values, type = "b")



# result: again, extract 3 components  
  
  
PCA\_lexicals\_props <- psych::principal(lexicals\_props, nfactors = 3, rotate =   
"varimax", scores = TRUE)  
  
PCA\_lexicals\_props # eigenvalues and exp variances good

## Principal Components Analysis  
## Call: psych::principal(r = lexicals\_props, nfactors = 3, rotate = "varimax",   
## scores = TRUE)  
## Standardized loadings (pattern matrix) based upon correlation matrix  
## RC2 RC1 RC3 h2 u2 com  
## letters -0.35 0.86 -0.33 0.98 0.024 1.6  
## phonemes\_DUTCHPOND -0.25 0.87 -0.39 0.98 0.025 1.6  
## orth\_neighbours\_DUTCHPOND 0.29 -0.37 0.86 0.96 0.042 1.6  
## phon\_neighbours\_DUTCHPOND 0.31 -0.36 0.86 0.96 0.040 1.6  
## freq\_lg10CD\_SUBTLEXNL 0.93 -0.25 0.23 0.98 0.023 1.3  
## freq\_lg10WF\_SUBTLEXNL 0.93 -0.25 0.23 0.98 0.024 1.3  
## freq\_CELEX\_lem 0.85 -0.24 0.31 0.88 0.120 1.4  
##   
## RC2 RC1 RC3  
## SS loadings 2.81 1.95 1.94  
## Proportion Var 0.40 0.28 0.28  
## Cumulative Var 0.40 0.68 0.96  
## Proportion Explained 0.42 0.29 0.29  
## Cumulative Proportion 0.42 0.71 1.00  
##   
## Mean item complexity = 1.5  
## Test of the hypothesis that 3 components are sufficient.  
##   
## The root mean square of the residuals (RMSR) is 0.02   
## with the empirical chi square 6.46 with prob < 0.091   
##   
## Fit based upon off diagonal values = 1

PCA\_lexicals\_props$loadings

##   
## Loadings:  
## RC2 RC1 RC3   
## letters -0.350 0.862 -0.332  
## phonemes\_DUTCHPOND -0.246 0.873 -0.392  
## orth\_neighbours\_DUTCHPOND 0.294 -0.368 0.858  
## phon\_neighbours\_DUTCHPOND 0.309 -0.355 0.859  
## freq\_lg10CD\_SUBTLEXNL 0.928 -0.252 0.227  
## freq\_lg10WF\_SUBTLEXNL 0.927 -0.250 0.233  
## freq\_CELEX\_lem 0.852 -0.244 0.309  
##   
## RC2 RC1 RC3  
## SS loadings 2.812 1.952 1.940  
## Proportion Var 0.402 0.279 0.277  
## Cumulative Var 0.402 0.681 0.958

# The PCA replicates Lynott and Connell. Standdized correlation coeffs  
# between each PC and its corresponding set of variables are all above .89,  
# while the rest of coefficients are all below .33.   
  
PCA\_lexicals\_props

## Principal Components Analysis  
## Call: psych::principal(r = lexicals\_props, nfactors = 3, rotate = "varimax",   
## scores = TRUE)  
## Standardized loadings (pattern matrix) based upon correlation matrix  
## RC2 RC1 RC3 h2 u2 com  
## letters -0.35 0.86 -0.33 0.98 0.024 1.6  
## phonemes\_DUTCHPOND -0.25 0.87 -0.39 0.98 0.025 1.6  
## orth\_neighbours\_DUTCHPOND 0.29 -0.37 0.86 0.96 0.042 1.6  
## phon\_neighbours\_DUTCHPOND 0.31 -0.36 0.86 0.96 0.040 1.6  
## freq\_lg10CD\_SUBTLEXNL 0.93 -0.25 0.23 0.98 0.023 1.3  
## freq\_lg10WF\_SUBTLEXNL 0.93 -0.25 0.23 0.98 0.024 1.3  
## freq\_CELEX\_lem 0.85 -0.24 0.31 0.88 0.120 1.4  
##   
## RC2 RC1 RC3  
## SS loadings 2.81 1.95 1.94  
## Proportion Var 0.40 0.28 0.28  
## Cumulative Var 0.40 0.68 0.96  
## Proportion Explained 0.42 0.29 0.29  
## Cumulative Proportion 0.42 0.71 1.00  
##   
## Mean item complexity = 1.5  
## Test of the hypothesis that 3 components are sufficient.  
##   
## The root mean square of the residuals (RMSR) is 0.02   
## with the empirical chi square 6.46 with prob < 0.091   
##   
## Fit based upon off diagonal values = 1

# RC1 = length // RC2 = frequency // RC3 = distinctiveness  
  
PCA\_lexicals\_props$residual

## letters phonemes\_DUTCHPOND  
## letters 0.023517045 -0.023962895  
## phonemes\_DUTCHPOND -0.023962895 0.024527905  
## orth\_neighbours\_DUTCHPOND -0.012514682 0.013421737  
## phon\_neighbours\_DUTCHPOND 0.007817494 -0.008192472  
## freq\_lg10CD\_SUBTLEXNL -0.004433168 0.005936642  
## freq\_lg10WF\_SUBTLEXNL -0.006253943 0.007759574  
## freq\_CELEX\_lem 0.015885789 -0.019361066  
## orth\_neighbours\_DUTCHPOND  
## letters -0.012514682  
## phonemes\_DUTCHPOND 0.013421737  
## orth\_neighbours\_DUTCHPOND 0.041585846  
## phon\_neighbours\_DUTCHPOND -0.039472453  
## freq\_lg10CD\_SUBTLEXNL 0.001286716  
## freq\_lg10WF\_SUBTLEXNL 0.001353789  
## freq\_CELEX\_lem -0.004178748  
## phon\_neighbours\_DUTCHPOND freq\_lg10CD\_SUBTLEXNL  
## letters 0.007817494 -0.004433168  
## phonemes\_DUTCHPOND -0.008192472 0.005936642  
## orth\_neighbours\_DUTCHPOND -0.039472453 0.001286716  
## phon\_neighbours\_DUTCHPOND 0.040374981 0.006222638  
## freq\_lg10CD\_SUBTLEXNL 0.006222638 0.023113551  
## freq\_lg10WF\_SUBTLEXNL 0.006494202 0.020975916  
## freq\_CELEX\_lem -0.014031271 -0.050843434  
## freq\_lg10WF\_SUBTLEXNL freq\_CELEX\_lem  
## letters -0.006253943 0.015885789  
## phonemes\_DUTCHPOND 0.007759574 -0.019361066  
## orth\_neighbours\_DUTCHPOND 0.001353789 -0.004178748  
## phon\_neighbours\_DUTCHPOND 0.006494202 -0.014031271  
## freq\_lg10CD\_SUBTLEXNL 0.020975916 -0.050843434  
## freq\_lg10WF\_SUBTLEXNL 0.023956553 -0.052102821  
## freq\_CELEX\_lem -0.052102821 0.119626897

PCA\_lexicals\_props$fit

## [1] 0.9985986

# Results based on a Kaiser-normalizalized orthogonal (varimax) rotation  
# (by default in psych::stats pack). Residuals good: less than half w/ absolute   
# values > 0.05. Model fit good, > .90. Communalities (h2) good, all well > .7  
  
props <- cbind(props, PCA\_lexicals\_props$scores)  
  
  
  
# REGRESSION  
  
# standardize (mean-center and scale)  
props$s\_Auditory <- scale(props$Auditory)  
props$s\_Haptic <- scale(props$Haptic)  
props$s\_Visual <- scale(props$Visual)  
props$s\_freq\_lg10CD\_SUBTLEXNL <- scale(props$freq\_lg10CD\_SUBTLEXNL)  
props$s\_freq\_lg10WF\_SUBTLEXNL <- scale(props$freq\_lg10WF\_SUBTLEXNL)  
props$s\_freq\_CELEX\_lem <- scale(props$freq\_CELEX\_lem)  
props$s\_AoA\_Brysbaertetal2014 <- scale(props$AoA\_Brysbaertetal2014)  
props$s\_concrete\_Brysbaertetal2014 <- scale(props$concrete\_Brysbaertetal2014)  
props$s\_letters <- scale(props$letters)  
props$s\_phonemes\_DUTCHPOND <- scale(props$phonemes\_DUTCHPOND)  
props$s\_orth\_neighbours\_DUTCHPOND <- scale(props$orth\_neighbours\_DUTCHPOND)  
props$s\_phon\_neighbours\_DUTCHPOND <- scale(props$phon\_neighbours\_DUTCHPOND)  
props$s\_RC1\_lexicals <- scale(props$RC1)  
props$s\_RC2\_lexicals <- scale(props$RC2)   
props$s\_RC3\_lexicals <- scale(props$RC3)  
  
# length: letters  
fit\_letters\_props <- lm(props$s\_letters ~ props$s\_Auditory + props$s\_Haptic +   
props$s\_Visual, data = props)  
stat.desc(fit\_letters\_props$residuals, norm = TRUE)

## x  
## nbr.val 3.360000e+02  
## nbr.null 0.000000e+00  
## nbr.na 0.000000e+00  
## min -1.951464e+00  
## max 3.056793e+00  
## range 5.008258e+00  
## sum 5.710710e-15  
## median -5.660484e-03  
## mean 1.694389e-17  
## SE.mean 5.381262e-02  
## CI.mean.0.95 1.058532e-01  
## var 9.729883e-01  
## std.dev 9.864017e-01  
## coef.var 5.821579e+16  
## skewness 2.021250e-01  
## skew.2SE 7.596443e-01  
## kurtosis -5.827220e-01  
## kurt.2SE -1.098206e+00  
## normtest.W 9.848674e-01  
## normtest.p 1.348370e-03

# residuals distribution: kurtose. Raw scores/2.SE > 1  
# have to log-transform DV and re-run regression  
  
psych::describe(props$s\_letters)

## vars n mean sd median trimmed mad min max range skew kurtosis  
## X1 1 336 0 1 -0.05 -0.03 1.31 -1.82 3.04 4.86 0.15 -0.63  
## se  
## X1 0.05

props$log\_s\_letters <- log(3 + props$s\_letters)  
  
fit\_letters\_props <- lm(props$log\_s\_letters ~ props$s\_Auditory + props$s\_Haptic +   
props$s\_Visual, data = props)  
  
# check residuals again  
stat.desc(fit\_letters\_props$residuals, norm = TRUE)

## x  
## nbr.val 3.360000e+02  
## nbr.null 0.000000e+00  
## nbr.na 0.000000e+00  
## min -9.191181e-01  
## max 7.831737e-01  
## range 1.702292e+00  
## sum 2.737394e-15  
## median 5.820411e-02  
## mean 8.121029e-18  
## SE.mean 1.947699e-02  
## CI.mean.0.95 3.831262e-02  
## var 1.274627e-01  
## std.dev 3.570191e-01  
## coef.var 4.396230e+16  
## skewness -4.372009e-01  
## skew.2SE -1.643128e+00  
## kurtosis -5.110083e-01  
## kurt.2SE -9.630531e-01  
## normtest.W 9.708083e-01  
## normtest.p 2.669646e-06

# same; go back  
fit\_letters\_props <- lm(props$s\_letters ~ props$s\_Auditory + props$s\_Haptic +   
props$s\_Visual, data = props)  
  
# Check multicollinearity: largest VIF (pref. < 10), mean VIF (pref. around 1), and   
# tolerance (pref. > 0.2)  
vif(fit\_letters\_props)

## props$s\_Auditory props$s\_Haptic props$s\_Visual   
## 1.390569 1.063636 1.369298

mean(vif(fit\_letters\_props))

## [1] 1.274501

1/vif(fit\_letters\_props)

## props$s\_Auditory props$s\_Haptic props$s\_Visual   
## 0.7191299 0.9401711 0.7303013

# RESULTS: all good  
  
step\_letters\_props\_AIC <- stepAIC(fit\_letters\_props, direction="both")

## Start: AIC=-2.2  
## props$s\_letters ~ props$s\_Auditory + props$s\_Haptic + props$s\_Visual  
##   
## Df Sum of Sq RSS AIC  
## - props$s\_Visual 1 0.05459 326.01 -4.1460  
## <none> 325.95 -2.2023  
## - props$s\_Haptic 1 2.28519 328.24 -1.8548  
## - props$s\_Auditory 1 3.10531 329.06 -1.0164  
##   
## Step: AIC=-4.15  
## props$s\_letters ~ props$s\_Auditory + props$s\_Haptic  
##   
## Df Sum of Sq RSS AIC  
## <none> 326.01 -4.1460  
## - props$s\_Haptic 1 2.3695 328.38 -3.7127  
## + props$s\_Visual 1 0.0546 325.95 -2.2023  
## - props$s\_Auditory 1 4.6444 330.65 -1.3930

step\_letters\_props\_F <- stepAIC(fit\_letters\_props, direction="both", test="F")

## Start: AIC=-2.2  
## props$s\_letters ~ props$s\_Auditory + props$s\_Haptic + props$s\_Visual  
##   
## Df Sum of Sq RSS AIC F Value Pr(F)   
## - props$s\_Visual 1 0.05459 326.01 -4.1460 0.0556 0.81372   
## <none> 325.95 -2.2023   
## - props$s\_Haptic 1 2.28519 328.24 -1.8548 2.3276 0.12805   
## - props$s\_Auditory 1 3.10531 329.06 -1.0164 3.1629 0.07624 .  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Step: AIC=-4.15  
## props$s\_letters ~ props$s\_Auditory + props$s\_Haptic  
##   
## Df Sum of Sq RSS AIC F Value Pr(F)   
## <none> 326.01 -4.1460   
## - props$s\_Haptic 1 2.3695 328.38 -3.7127 2.4203 0.1207   
## + props$s\_Visual 1 0.0546 325.95 -2.2023 0.0556 0.8137   
## - props$s\_Auditory 1 4.6444 330.65 -1.3930 4.7441 0.0301 \*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

summary(fit\_letters\_props)

##   
## Call:  
## lm(formula = props$s\_letters ~ props$s\_Auditory + props$s\_Haptic +   
## props$s\_Visual, data = props)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.95146 -0.79178 -0.00566 0.71461 3.05679   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -6.413e-17 5.406e-02 0.000 1.0000   
## props$s\_Auditory 1.135e-01 6.384e-02 1.778 0.0762 .  
## props$s\_Haptic -8.518e-02 5.583e-02 -1.526 0.1280   
## props$s\_Visual -1.494e-02 6.335e-02 -0.236 0.8137   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.9908 on 332 degrees of freedom  
## (7 observations deleted due to missingness)  
## Multiple R-squared: 0.02701, Adjusted R-squared: 0.01822   
## F-statistic: 3.072 on 3 and 332 DF, p-value: 0.02793

# length: phonemes\_DUTCHPOND  
fit\_phonemes\_DUTCHPOND\_props <- lm(props$s\_phonemes\_DUTCHPOND ~ props$s\_Auditory +   
props$s\_Haptic + props$s\_Visual, data = props)  
stat.desc(fit\_phonemes\_DUTCHPOND\_props$residuals, norm = TRUE)

## x  
## nbr.val 1.850000e+02  
## nbr.null 0.000000e+00  
## nbr.na 0.000000e+00  
## min -1.781220e+00  
## max 2.893531e+00  
## range 4.674751e+00  
## sum -7.140122e-15  
## median -1.586697e-01  
## mean -3.864448e-17  
## SE.mean 7.203029e-02  
## CI.mean.0.95 1.421115e-01  
## var 9.598471e-01  
## std.dev 9.797179e-01  
## coef.var -2.535208e+16  
## skewness 3.927364e-01  
## skew.2SE 1.099162e+00  
## kurtosis -6.193431e-01  
## kurt.2SE -8.711822e-01  
## normtest.W 9.699743e-01  
## normtest.p 5.167728e-04

# residuals distribution: skew. Raw scores/2.SE > 1  
# have to log-transform DV and re-run regression  
  
psych::describe(props$s\_phonemes\_DUTCHPOND)

## vars n mean sd median trimmed mad min max range skew kurtosis  
## X1 1 185 0 1 -0.2 -0.05 1.52 -1.74 2.88 4.62 0.43 -0.54  
## se  
## X1 0.07

props$log\_s\_phonemes\_DUTCHPOND <- log(3 + props$s\_phonemes\_DUTCHPOND)  
  
fit\_phonemes\_DUTCHPOND\_props <- lm(props$log\_s\_phonemes\_DUTCHPOND ~ props$s\_Auditory  
 + props$s\_Haptic + props$s\_Visual, data = props)  
  
# check residuals again  
stat.desc(fit\_phonemes\_DUTCHPOND\_props$residuals, norm = TRUE)

## x  
## nbr.val 1.850000e+02  
## nbr.null 0.000000e+00  
## nbr.na 0.000000e+00  
## min -8.201762e-01  
## max 7.325980e-01  
## range 1.552774e+00  
## sum 2.125036e-15  
## median 5.242801e-04  
## mean 1.148437e-17  
## SE.mean 2.459029e-02  
## CI.mean.0.95 4.851518e-02  
## var 1.118662e-01  
## std.dev 3.344641e-01  
## coef.var 2.912340e+16  
## skewness -1.073812e-01  
## skew.2SE -3.005307e-01  
## kurtosis -8.521788e-01  
## kurt.2SE -1.198694e+00  
## normtest.W 9.803325e-01  
## normtest.p 1.039941e-02

# worse; back  
fit\_phonemes\_DUTCHPOND\_props <- lm(props$s\_phonemes\_DUTCHPOND ~ props$s\_Auditory +   
props$s\_Haptic + props$s\_Visual, data = props)  
  
# Check multicollinearity: largest VIF (pref. < 10), mean VIF (pref. around 1), and   
# tolerance (pref. > 0.2)  
vif(fit\_phonemes\_DUTCHPOND\_props)

## props$s\_Auditory props$s\_Haptic props$s\_Visual   
## 1.106936 1.001006 1.107723

mean(vif(fit\_phonemes\_DUTCHPOND\_props))

## [1] 1.071888

1/vif(fit\_phonemes\_DUTCHPOND\_props)

## props$s\_Auditory props$s\_Haptic props$s\_Visual   
## 0.9033949 0.9989951 0.9027527

# RESULTS: all good  
  
step\_phonemes\_DUTCHPOND\_props\_AIC <- stepAIC(fit\_phonemes\_DUTCHPOND\_props,   
direction="both")

## Start: AIC=-0.58  
## props$s\_phonemes\_DUTCHPOND ~ props$s\_Auditory + props$s\_Haptic +   
## props$s\_Visual  
##   
## Df Sum of Sq RSS AIC  
## - props$s\_Haptic 1 1.3289 177.94 -1.1974  
## - props$s\_Visual 1 1.4171 178.03 -1.1057  
## <none> 176.61 -0.5842  
## - props$s\_Auditory 1 5.7163 182.33 3.3087  
##   
## Step: AIC=-1.2  
## props$s\_phonemes\_DUTCHPOND ~ props$s\_Auditory + props$s\_Visual  
##   
## Df Sum of Sq RSS AIC  
## - props$s\_Visual 1 1.5050 179.45 -1.63928  
## <none> 177.94 -1.19741  
## + props$s\_Haptic 1 1.3289 176.61 -0.58424  
## - props$s\_Auditory 1 5.8055 183.75 2.74206  
##   
## Step: AIC=-1.64  
## props$s\_phonemes\_DUTCHPOND ~ props$s\_Auditory  
##   
## Df Sum of Sq RSS AIC  
## <none> 179.45 -1.63928  
## + props$s\_Visual 1 1.5050 177.94 -1.19741  
## + props$s\_Haptic 1 1.4168 178.03 -1.10571  
## - props$s\_Auditory 1 4.5542 184.00 0.99729

step\_phonemes\_DUTCHPOND\_props\_F <- stepAIC(fit\_phonemes\_DUTCHPOND\_props,   
direction="both", test="F")

## Start: AIC=-0.58  
## props$s\_phonemes\_DUTCHPOND ~ props$s\_Auditory + props$s\_Haptic +   
## props$s\_Visual  
##   
## Df Sum of Sq RSS AIC F Value Pr(F)   
## - props$s\_Haptic 1 1.3289 177.94 -1.1974 1.3619 0.24474   
## - props$s\_Visual 1 1.4171 178.03 -1.1057 1.4524 0.22972   
## <none> 176.61 -0.5842   
## - props$s\_Auditory 1 5.7163 182.33 3.3087 5.8584 0.01649 \*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Step: AIC=-1.2  
## props$s\_phonemes\_DUTCHPOND ~ props$s\_Auditory + props$s\_Visual  
##   
## Df Sum of Sq RSS AIC F Value Pr(F)   
## - props$s\_Visual 1 1.5050 179.45 -1.63928 1.5393 0.21631   
## <none> 177.94 -1.19741   
## + props$s\_Haptic 1 1.3289 176.61 -0.58424 1.3619 0.24474   
## - props$s\_Auditory 1 5.8055 183.75 2.74206 5.9380 0.01578 \*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Step: AIC=-1.64  
## props$s\_phonemes\_DUTCHPOND ~ props$s\_Auditory  
##   
## Df Sum of Sq RSS AIC F Value Pr(F)   
## <none> 179.45 -1.63928   
## + props$s\_Visual 1 1.5050 177.94 -1.19741 1.5393 0.21631   
## + props$s\_Haptic 1 1.4168 178.03 -1.10571 1.4484 0.23035   
## - props$s\_Auditory 1 4.5542 184.00 0.99729 4.6444 0.03246 \*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

summary(fit\_phonemes\_DUTCHPOND\_props)

##   
## Call:  
## lm(formula = props$s\_phonemes\_DUTCHPOND ~ props$s\_Auditory +   
## props$s\_Haptic + props$s\_Visual, data = props)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.7812 -0.7401 -0.1587 0.8007 2.8935   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.05145 0.08057 0.639 0.5239   
## props$s\_Auditory 0.25688 0.10613 2.420 0.0165 \*  
## props$s\_Haptic -0.08477 0.07264 -1.167 0.2447   
## props$s\_Visual 0.10714 0.08891 1.205 0.2297   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.9878 on 181 degrees of freedom  
## (158 observations deleted due to missingness)  
## Multiple R-squared: 0.04015, Adjusted R-squared: 0.02424   
## F-statistic: 2.524 on 3 and 181 DF, p-value: 0.05917

# distinctiveness: orth neigh size  
fit\_orth\_neighbours\_DUTCHPOND\_props <- lm(props$s\_orth\_neighbours\_DUTCHPOND ~   
props$s\_Auditory + props$s\_Haptic + props$s\_Visual, data = props)  
stat.desc(fit\_orth\_neighbours\_DUTCHPOND\_props$residuals, norm = TRUE)

## x  
## nbr.val 3.360000e+02  
## nbr.null 0.000000e+00  
## nbr.na 0.000000e+00  
## min -9.012905e-01  
## max 4.201627e+00  
## range 5.102917e+00  
## sum 1.110223e-15  
## median -3.445061e-01  
## mean 3.250509e-18  
## SE.mean 5.397738e-02  
## CI.mean.0.95 1.061773e-01  
## var 9.789552e-01  
## std.dev 9.894217e-01  
## coef.var 3.043898e+17  
## skewness 2.005223e+00  
## skew.2SE 7.536211e+00  
## kurtosis 4.014171e+00  
## kurt.2SE 7.565161e+00  
## normtest.W 7.527604e-01  
## normtest.p 4.267770e-22

# residuals distribution: skewed and kurtosed. Raw scores/2.SE > 1  
# have to log-transform DV and re-run regression  
  
psych::describe(props$s\_orth\_neighbours\_DUTCHPOND)

## vars n mean sd median trimmed mad min max range skew kurtosis se  
## X1 1 336 0 1 -0.48 -0.22 0.3 -0.68 4.25 4.93 2.06 4.06 0.05

props$log\_s\_orth\_neighbours\_DUTCHPOND <- log(2 + props$s\_orth\_neighbours\_DUTCHPOND)  
  
fit\_orth\_neighbours\_DUTCHPOND\_props <- lm(props$log\_s\_orth\_neighbours\_DUTCHPOND ~   
props$s\_Auditory + props$s\_Haptic + props$s\_Visual, data = props)  
  
# check residuals again  
stat.desc(fit\_orth\_neighbours\_DUTCHPOND\_props$residuals, norm = TRUE)

## x  
## nbr.val 3.360000e+02  
## nbr.null 0.000000e+00  
## nbr.na 0.000000e+00  
## min -3.965545e-01  
## max 1.203044e+00  
## range 1.599599e+00  
## sum 2.985459e-15  
## median -1.371519e-01  
## mean 8.902735e-18  
## SE.mean 2.089711e-02  
## CI.mean.0.95 4.110610e-02  
## var 1.467276e-01  
## std.dev 3.830504e-01  
## coef.var 4.302615e+16  
## skewness 1.257764e+00  
## skew.2SE 4.727041e+00  
## kurtosis 7.776239e-01  
## kurt.2SE 1.465521e+00  
## normtest.W 8.480804e-01  
## normtest.p 1.534991e-17

# quite better  
  
# Check multicollinearity: largest VIF (pref. < 10), mean VIF (pref. around 1), and   
# tolerance (pref. > 0.2)  
vif(fit\_orth\_neighbours\_DUTCHPOND\_props)

## props$s\_Auditory props$s\_Haptic props$s\_Visual   
## 1.390569 1.063636 1.369298

mean(vif(fit\_orth\_neighbours\_DUTCHPOND\_props))

## [1] 1.274501

1/vif(fit\_orth\_neighbours\_DUTCHPOND\_props)

## props$s\_Auditory props$s\_Haptic props$s\_Visual   
## 0.7191299 0.9401711 0.7303013

# RESULTS: all good  
  
step\_orth\_neighbours\_DUTCHPOND\_props\_AIC <-   
stepAIC(fit\_orth\_neighbours\_DUTCHPOND\_props, direction="both")

## Start: AIC=-637.85  
## props$log\_s\_orth\_neighbours\_DUTCHPOND ~ props$s\_Auditory + props$s\_Haptic +   
## props$s\_Visual  
##   
## Df Sum of Sq RSS AIC  
## - props$s\_Visual 1 0.01250 49.166 -639.76  
## - props$s\_Haptic 1 0.05865 49.212 -639.44  
## <none> 49.154 -637.85  
## - props$s\_Auditory 1 0.69898 49.853 -635.10  
##   
## Step: AIC=-639.76  
## props$log\_s\_orth\_neighbours\_DUTCHPOND ~ props$s\_Auditory + props$s\_Haptic  
##   
## Df Sum of Sq RSS AIC  
## - props$s\_Haptic 1 0.05428 49.221 -641.39  
## <none> 49.166 -639.76  
## + props$s\_Visual 1 0.01250 49.154 -637.85  
## - props$s\_Auditory 1 0.80432 49.971 -636.31  
##   
## Step: AIC=-641.39  
## props$log\_s\_orth\_neighbours\_DUTCHPOND ~ props$s\_Auditory  
##   
## Df Sum of Sq RSS AIC  
## <none> 49.221 -641.39  
## + props$s\_Haptic 1 0.05428 49.166 -639.76  
## + props$s\_Visual 1 0.00813 49.212 -639.44  
## - props$s\_Auditory 1 0.95192 50.172 -636.95

step\_orth\_neighbours\_DUTCHPOND\_props\_F <-   
stepAIC(fit\_orth\_neighbours\_DUTCHPOND\_props, direction="both", test="F")

## Start: AIC=-637.85  
## props$log\_s\_orth\_neighbours\_DUTCHPOND ~ props$s\_Auditory + props$s\_Haptic +   
## props$s\_Visual  
##   
## Df Sum of Sq RSS AIC F Value Pr(F)   
## - props$s\_Visual 1 0.01250 49.166 -639.76 0.0845 0.7715   
## - props$s\_Haptic 1 0.05865 49.212 -639.44 0.3962 0.5295   
## <none> 49.154 -637.85   
## - props$s\_Auditory 1 0.69898 49.853 -635.10 4.7212 0.0305 \*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Step: AIC=-639.76  
## props$log\_s\_orth\_neighbours\_DUTCHPOND ~ props$s\_Auditory + props$s\_Haptic  
##   
## Df Sum of Sq RSS AIC F Value Pr(F)   
## - props$s\_Haptic 1 0.05428 49.221 -641.39 0.3676 0.54471   
## <none> 49.166 -639.76   
## + props$s\_Visual 1 0.01250 49.154 -637.85 0.0845 0.77153   
## - props$s\_Auditory 1 0.80432 49.971 -636.31 5.4476 0.02019 \*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Step: AIC=-641.39  
## props$log\_s\_orth\_neighbours\_DUTCHPOND ~ props$s\_Auditory  
##   
## Df Sum of Sq RSS AIC F Value Pr(F)   
## <none> 49.221 -641.39   
## + props$s\_Haptic 1 0.05428 49.166 -639.76 0.3676 0.54471   
## + props$s\_Visual 1 0.00813 49.212 -639.44 0.0550 0.81469   
## - props$s\_Auditory 1 0.95192 50.172 -636.95 6.4595 0.01149 \*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

summary(fit\_orth\_neighbours\_DUTCHPOND\_props)

##   
## Call:  
## lm(formula = props$log\_s\_orth\_neighbours\_DUTCHPOND ~ props$s\_Auditory +   
## props$s\_Haptic + props$s\_Visual, data = props)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.3966 -0.2658 -0.1371 0.1314 1.2030   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.605335 0.020991 28.837 <2e-16 \*\*\*  
## props$s\_Auditory -0.053865 0.024790 -2.173 0.0305 \*   
## props$s\_Haptic 0.013646 0.021681 0.629 0.5295   
## props$s\_Visual -0.007149 0.024600 -0.291 0.7715   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.3848 on 332 degrees of freedom  
## (7 observations deleted due to missingness)  
## Multiple R-squared: 0.0203, Adjusted R-squared: 0.01145   
## F-statistic: 2.294 on 3 and 332 DF, p-value: 0.07784

# distinctiveness: phon neigh size  
fit\_phon\_neighbours\_DUTCHPOND\_props <- lm(props$s\_phon\_neighbours\_DUTCHPOND ~   
props$s\_Auditory + props$s\_Haptic + props$s\_Visual, data = props)  
stat.desc(fit\_phon\_neighbours\_DUTCHPOND\_props$residuals, norm = TRUE)

## x  
## nbr.val 3.360000e+02  
## nbr.null 0.000000e+00  
## nbr.na 0.000000e+00  
## min -8.455782e-01  
## max 4.812038e+00  
## range 5.657616e+00  
## sum 2.758210e-16  
## median -3.634595e-01  
## mean 8.576410e-19  
## SE.mean 5.382661e-02  
## CI.mean.0.95 1.058807e-01  
## var 9.734942e-01  
## std.dev 9.866581e-01  
## coef.var 1.150432e+18  
## skewness 2.192935e+00  
## skew.2SE 8.241687e+00  
## kurtosis 4.900577e+00  
## kurt.2SE 9.235695e+00  
## normtest.W 7.165177e-01  
## normtest.p 1.771880e-23

# residuals distribution: skewed and kurtosed. Raw scores/2.SE > 1  
# have to log-transform DV and re-run regression  
  
psych::describe(props$s\_phon\_neighbours\_DUTCHPOND)

## vars n mean sd median trimmed mad min max range skew kurtosis se  
## X1 1 336 0 1 -0.47 -0.24 0.2 -0.6 4.95 5.55 2.27 5.04 0.05

props$log\_s\_phon\_neighbours\_DUTCHPOND <- log(2 + props$s\_phon\_neighbours\_DUTCHPOND)  
  
fit\_phon\_neighbours\_DUTCHPOND\_props <- lm(props$log\_s\_phon\_neighbours\_DUTCHPOND ~   
props$s\_Auditory + props$s\_Haptic + props$s\_Visual, data = props)  
  
# check residuals again  
stat.desc(fit\_phon\_neighbours\_DUTCHPOND\_props$residuals, norm = TRUE)

## x  
## nbr.val 3.360000e+02  
## nbr.null 0.000000e+00  
## nbr.na 0.000000e+00  
## min -3.517218e-01  
## max 1.277575e+00  
## range 1.629297e+00  
## sum -3.259545e-15  
## median -1.356554e-01  
## mean -9.718796e-18  
## SE.mean 2.007484e-02  
## CI.mean.0.95 3.948862e-02  
## var 1.354077e-01  
## std.dev 3.679779e-01  
## coef.var -3.786250e+16  
## skewness 1.486941e+00  
## skew.2SE 5.588356e+00  
## kurtosis 1.393719e+00  
## kurt.2SE 2.626621e+00  
## normtest.W 8.046259e-01  
## normtest.p 8.008540e-20

# quite better  
  
# Check multicollinearity: largest VIF (pref. < 10), mean VIF (pref. around 1), and   
# tolerance (pref. > 0.2)  
vif(fit\_phon\_neighbours\_DUTCHPOND\_props)

## props$s\_Auditory props$s\_Haptic props$s\_Visual   
## 1.390569 1.063636 1.369298

mean(vif(fit\_phon\_neighbours\_DUTCHPOND\_props))

## [1] 1.274501

1/vif(fit\_phon\_neighbours\_DUTCHPOND\_props)

## props$s\_Auditory props$s\_Haptic props$s\_Visual   
## 0.7191299 0.9401711 0.7303013

# RESULTS: all good  
  
step\_phon\_neighbours\_DUTCHPOND\_props\_AIC <-   
stepAIC(fit\_phon\_neighbours\_DUTCHPOND\_props, direction="both")

## Start: AIC=-664.82  
## props$log\_s\_phon\_neighbours\_DUTCHPOND ~ props$s\_Auditory + props$s\_Haptic +   
## props$s\_Visual  
##   
## Df Sum of Sq RSS AIC  
## - props$s\_Visual 1 0.01518 45.377 -666.71  
## - props$s\_Haptic 1 0.08053 45.442 -666.23  
## <none> 45.362 -664.82  
## - props$s\_Auditory 1 0.87817 46.240 -660.38  
##   
## Step: AIC=-666.71  
## props$log\_s\_phon\_neighbours\_DUTCHPOND ~ props$s\_Auditory + props$s\_Haptic  
##   
## Df Sum of Sq RSS AIC  
## - props$s\_Haptic 1 0.07492 45.452 -668.16  
## <none> 45.377 -666.71  
## + props$s\_Visual 1 0.01518 45.362 -664.82  
## - props$s\_Auditory 1 1.01291 46.390 -661.29  
##   
## Step: AIC=-668.16  
## props$log\_s\_phon\_neighbours\_DUTCHPOND ~ props$s\_Auditory  
##   
## Df Sum of Sq RSS AIC  
## <none> 45.452 -668.16  
## + props$s\_Haptic 1 0.07492 45.377 -666.71  
## + props$s\_Visual 1 0.00957 45.442 -666.23  
## - props$s\_Auditory 1 1.20508 46.657 -661.36

step\_phon\_neighbours\_DUTCHPOND\_props\_F <-  
 stepAIC(fit\_phon\_neighbours\_DUTCHPOND\_props, direction="both", test="F")

## Start: AIC=-664.82  
## props$log\_s\_phon\_neighbours\_DUTCHPOND ~ props$s\_Auditory + props$s\_Haptic +   
## props$s\_Visual  
##   
## Df Sum of Sq RSS AIC F Value Pr(F)   
## - props$s\_Visual 1 0.01518 45.377 -666.71 0.1111 0.7391   
## - props$s\_Haptic 1 0.08053 45.442 -666.23 0.5894 0.4432   
## <none> 45.362 -664.82   
## - props$s\_Auditory 1 0.87817 46.240 -660.38 6.4273 0.0117 \*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Step: AIC=-666.71  
## props$log\_s\_phon\_neighbours\_DUTCHPOND ~ props$s\_Auditory + props$s\_Haptic  
##   
## Df Sum of Sq RSS AIC F Value Pr(F)   
## - props$s\_Haptic 1 0.07492 45.452 -668.16 0.5498 0.458928   
## <none> 45.377 -666.71   
## + props$s\_Visual 1 0.01518 45.362 -664.82 0.1111 0.739066   
## - props$s\_Auditory 1 1.01291 46.390 -661.29 7.4333 0.006742 \*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Step: AIC=-668.16  
## props$log\_s\_phon\_neighbours\_DUTCHPOND ~ props$s\_Auditory  
##   
## Df Sum of Sq RSS AIC F Value Pr(F)   
## <none> 45.452 -668.16   
## + props$s\_Haptic 1 0.07492 45.377 -666.71 0.5498 0.458928   
## + props$s\_Visual 1 0.00957 45.442 -666.23 0.0701 0.791309   
## - props$s\_Auditory 1 1.20508 46.657 -661.36 8.8555 0.003135 \*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

summary(fit\_phon\_neighbours\_DUTCHPOND\_props)

##   
## Call:  
## lm(formula = props$log\_s\_phon\_neighbours\_DUTCHPOND ~ props$s\_Auditory +   
## props$s\_Haptic + props$s\_Visual, data = props)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.3517 -0.2599 -0.1357 0.1060 1.2776   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.609359 0.020165 30.218 <2e-16 \*\*\*  
## props$s\_Auditory -0.060376 0.023815 -2.535 0.0117 \*   
## props$s\_Haptic 0.015990 0.020828 0.768 0.4432   
## props$s\_Visual -0.007878 0.023632 -0.333 0.7391   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.3696 on 332 degrees of freedom  
## (7 observations deleted due to missingness)  
## Multiple R-squared: 0.02776, Adjusted R-squared: 0.01897   
## F-statistic: 3.16 on 3 and 332 DF, p-value: 0.02486

# freq: SUBTLEX-NL log-10 CD  
  
fit\_freq\_lg10CD\_SUBTLEXNL\_props <- lm(props$s\_freq\_lg10CD\_SUBTLEXNL ~   
props$s\_Auditory + props$s\_Haptic + props$s\_Visual, data = props)  
stat.desc(fit\_freq\_lg10CD\_SUBTLEXNL\_props$residuals, norm = TRUE)

## x  
## nbr.val 2.900000e+02  
## nbr.null 0.000000e+00  
## nbr.na 0.000000e+00  
## min -1.692625e+00  
## max 2.415049e+00  
## range 4.107675e+00  
## sum 1.418483e-14  
## median -2.189748e-01  
## mean 4.907464e-17  
## SE.mean 5.563757e-02  
## CI.mean.0.95 1.095062e-01  
## var 8.977063e-01  
## std.dev 9.474736e-01  
## coef.var 1.930679e+16  
## skewness 4.242613e-01  
## skew.2SE 1.482369e+00  
## kurtosis -7.702721e-01  
## kurt.2SE -1.350186e+00  
## normtest.W 9.623283e-01  
## normtest.p 7.531921e-07

# residuals distribution: skew and kurtosed. Raw scores/2.SE > 1  
# have to log-transform DV and re-run regression  
  
psych::describe(props$s\_freq\_lg10CD\_SUBTLEXNL)

## vars n mean sd median trimmed mad min max range skew kurtosis  
## X1 1 290 0 1 -0.19 -0.04 1.12 -1.48 2.03 3.51 0.35 -1.06  
## se  
## X1 0.06

props$log\_s\_freq\_lg10CD\_SUBTLEXNL <- log(3 + props$s\_freq\_lg10CD\_SUBTLEXNL)  
  
fit\_freq\_lg10CD\_SUBTLEXNL\_props <- lm(props$log\_s\_freq\_lg10CD\_SUBTLEXNL ~   
props$s\_Auditory + props$s\_Haptic + props$s\_Visual, data = props)  
  
# check residuals again  
stat.desc(fit\_freq\_lg10CD\_SUBTLEXNL\_props$residuals, norm = TRUE)

## x  
## nbr.val 2.900000e+02  
## nbr.null 0.000000e+00  
## nbr.na 0.000000e+00  
## min -6.999745e-01  
## max 7.664373e-01  
## range 1.466412e+00  
## sum 9.454243e-16  
## median -3.012791e-02  
## mean 3.198747e-18  
## SE.mean 1.873435e-02  
## CI.mean.0.95 3.687306e-02  
## var 1.017829e-01  
## std.dev 3.190344e-01  
## coef.var 9.973731e+16  
## skewness 1.133701e-02  
## skew.2SE 3.961154e-02  
## kurtosis -8.184529e-01  
## kurt.2SE -1.434641e+00  
## normtest.W 9.840065e-01  
## normtest.p 2.534289e-03

# quite better  
  
# Check multicollinearity: largest VIF (pref. < 10), mean VIF (pref. around 1), and   
# tolerance (pref. > 0.2)  
vif(fit\_freq\_lg10CD\_SUBTLEXNL\_props)

## props$s\_Auditory props$s\_Haptic props$s\_Visual   
## 1.382103 1.039118 1.351219

mean(vif(fit\_freq\_lg10CD\_SUBTLEXNL\_props))

## [1] 1.25748

1/vif(fit\_freq\_lg10CD\_SUBTLEXNL\_props)

## props$s\_Auditory props$s\_Haptic props$s\_Visual   
## 0.7235353 0.9623547 0.7400726

# RESULTS: all good  
  
step\_freq\_lg10CD\_SUBTLEXNL\_props\_AIC <- stepAIC(fit\_freq\_lg10CD\_SUBTLEXNL\_props,   
direction="both")

## Start: AIC=-655.63  
## props$log\_s\_freq\_lg10CD\_SUBTLEXNL ~ props$s\_Auditory + props$s\_Haptic +   
## props$s\_Visual  
##   
## Df Sum of Sq RSS AIC  
## - props$s\_Haptic 1 0.00023 29.416 -657.62  
## <none> 29.415 -655.63  
## - props$s\_Visual 1 1.01036 30.426 -647.83  
## - props$s\_Auditory 1 1.03613 30.451 -647.59  
##   
## Step: AIC=-657.62  
## props$log\_s\_freq\_lg10CD\_SUBTLEXNL ~ props$s\_Auditory + props$s\_Visual  
##   
## Df Sum of Sq RSS AIC  
## <none> 29.416 -657.62  
## + props$s\_Haptic 1 0.00023 29.415 -655.63  
## - props$s\_Visual 1 1.01031 30.426 -649.83  
## - props$s\_Auditory 1 1.05585 30.471 -649.40

step\_freq\_lg10CD\_SUBTLEXNL\_\_propsF <- stepAIC(fit\_freq\_lg10CD\_SUBTLEXNL\_props,   
direction="both", test="F")

## Start: AIC=-655.63  
## props$log\_s\_freq\_lg10CD\_SUBTLEXNL ~ props$s\_Auditory + props$s\_Haptic +   
## props$s\_Visual  
##   
## Df Sum of Sq RSS AIC F Value Pr(F)   
## - props$s\_Haptic 1 0.00023 29.416 -657.62 0.0023 0.962128   
## <none> 29.415 -655.63   
## - props$s\_Visual 1 1.01036 30.426 -647.83 9.8236 0.001902 \*\*  
## - props$s\_Auditory 1 1.03613 30.451 -647.59 10.0742 0.001668 \*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Step: AIC=-657.62  
## props$log\_s\_freq\_lg10CD\_SUBTLEXNL ~ props$s\_Auditory + props$s\_Visual  
##   
## Df Sum of Sq RSS AIC F Value Pr(F)   
## <none> 29.416 -657.62   
## + props$s\_Haptic 1 0.00023 29.415 -655.63 0.0023 0.962128   
## - props$s\_Visual 1 1.01031 30.426 -649.83 9.8574 0.001868 \*\*  
## - props$s\_Auditory 1 1.05585 30.471 -649.40 10.3017 0.001480 \*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

summary(fit\_freq\_lg10CD\_SUBTLEXNL\_props)

##   
## Call:  
## lm(formula = props$log\_s\_freq\_lg10CD\_SUBTLEXNL ~ props$s\_Auditory +   
## props$s\_Haptic + props$s\_Visual, data = props)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.69997 -0.25070 -0.03013 0.26364 0.76644   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 1.0348471 0.0188686 54.845 < 2e-16 \*\*\*  
## props$s\_Auditory -0.0725470 0.0228568 -3.174 0.00167 \*\*   
## props$s\_Haptic -0.0009166 0.0192867 -0.048 0.96213   
## props$s\_Visual 0.0691961 0.0220773 3.134 0.00190 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.3207 on 286 degrees of freedom  
## (53 observations deleted due to missingness)  
## Multiple R-squared: 0.1252, Adjusted R-squared: 0.116   
## F-statistic: 13.64 on 3 and 286 DF, p-value: 2.415e-08

# freq: SUBTLEX-NL log-10 WF  
fit\_freq\_lg10WF\_SUBTLEXNL\_props <- lm(props$s\_freq\_lg10WF\_SUBTLEXNL ~   
props$s\_Auditory + props$s\_Haptic + props$s\_Visual, data = props)  
stat.desc(fit\_freq\_lg10WF\_SUBTLEXNL\_props$residuals, norm = TRUE)

## x  
## nbr.val 2.900000e+02  
## nbr.null 0.000000e+00  
## nbr.na 0.000000e+00  
## min -1.661721e+00  
## max 2.533130e+00  
## range 4.194851e+00  
## sum 8.913009e-15  
## median -2.272985e-01  
## mean 3.085995e-17  
## SE.mean 5.576327e-02  
## CI.mean.0.95 1.097536e-01  
## var 9.017672e-01  
## std.dev 9.496143e-01  
## coef.var 3.077174e+16  
## skewness 5.452976e-01  
## skew.2SE 1.905270e+00  
## kurtosis -5.163686e-01  
## kurt.2SE -9.051267e-01  
## normtest.W 9.599049e-01  
## normtest.p 3.588950e-07

# residuals distribution: skew. Raw scores/2.SE > 1  
# have to log-transform DV and re-run regression  
  
psych::describe(props$s\_freq\_lg10WF\_SUBTLEXNL)

## vars n mean sd median trimmed mad min max range skew kurtosis  
## X1 1 290 0 1 -0.17 -0.06 1.14 -1.45 2.52 3.97 0.46 -0.84  
## se  
## X1 0.06

props$log\_s\_freq\_lg10WF\_SUBTLEXNL <- log(3 + props$s\_freq\_lg10WF\_SUBTLEXNL)  
  
fit\_freq\_lg10WF\_SUBTLEXNL\_props <- lm(props$log\_s\_freq\_lg10WF\_SUBTLEXNL ~   
props$s\_Auditory + props$s\_Haptic + props$s\_Visual, data = props)  
  
# check residuals again  
stat.desc(fit\_freq\_lg10WF\_SUBTLEXNL\_props$residuals, norm = TRUE)

## x  
## nbr.val 2.900000e+02  
## nbr.null 0.000000e+00  
## nbr.na 0.000000e+00  
## min -6.780013e-01  
## max 7.683617e-01  
## range 1.446363e+00  
## sum -9.228729e-16  
## median -3.591764e-02  
## mean -3.201621e-18  
## SE.mean 1.847247e-02  
## CI.mean.0.95 3.635763e-02  
## var 9.895732e-02  
## std.dev 3.145748e-01  
## coef.var -9.825486e+16  
## skewness 9.385398e-02  
## skew.2SE 3.279259e-01  
## kurtosis -7.561993e-01  
## kurt.2SE -1.325519e+00  
## normtest.W 9.858284e-01  
## normtest.p 5.880988e-03

# quite better  
  
# Check multicollinearity: largest VIF (pref. < 10), mean VIF (pref. around 1), and   
# tolerance (pref. > 0.2)  
vif(fit\_freq\_lg10WF\_SUBTLEXNL\_props)

## props$s\_Auditory props$s\_Haptic props$s\_Visual   
## 1.382103 1.039118 1.351219

mean(vif(fit\_freq\_lg10WF\_SUBTLEXNL\_props))

## [1] 1.25748

1/vif(fit\_freq\_lg10WF\_SUBTLEXNL\_props)

## props$s\_Auditory props$s\_Haptic props$s\_Visual   
## 0.7235353 0.9623547 0.7400726

# RESULTS: all good  
  
step\_freq\_lg10WF\_SUBTLEXNL\_props\_AIC <- stepAIC(fit\_freq\_lg10WF\_SUBTLEXNL\_props,   
direction="both")

## Start: AIC=-663.79  
## props$log\_s\_freq\_lg10WF\_SUBTLEXNL ~ props$s\_Auditory + props$s\_Haptic +   
## props$s\_Visual  
##   
## Df Sum of Sq RSS AIC  
## - props$s\_Haptic 1 0.00042 28.599 -665.79  
## <none> 28.599 -663.79  
## - props$s\_Visual 1 0.97077 29.569 -656.11  
## - props$s\_Auditory 1 0.97099 29.570 -656.11  
##   
## Step: AIC=-665.79  
## props$log\_s\_freq\_lg10WF\_SUBTLEXNL ~ props$s\_Auditory + props$s\_Visual  
##   
## Df Sum of Sq RSS AIC  
## <none> 28.599 -665.79  
## + props$s\_Haptic 1 0.00042 28.599 -663.79  
## - props$s\_Visual 1 0.97040 29.570 -658.11  
## - props$s\_Auditory 1 0.98769 29.587 -657.94

step\_freq\_lg10WF\_SUBTLEXNL\_props\_F <- stepAIC(fit\_freq\_lg10WF\_SUBTLEXNL\_props,   
direction="both", test="F")

## Start: AIC=-663.79  
## props$log\_s\_freq\_lg10WF\_SUBTLEXNL ~ props$s\_Auditory + props$s\_Haptic +   
## props$s\_Visual  
##   
## Df Sum of Sq RSS AIC F Value Pr(F)   
## - props$s\_Haptic 1 0.00042 28.599 -665.79 0.0042 0.948236   
## <none> 28.599 -663.79   
## - props$s\_Visual 1 0.97077 29.569 -656.11 9.7082 0.002021 \*\*  
## - props$s\_Auditory 1 0.97099 29.570 -656.11 9.7103 0.002019 \*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Step: AIC=-665.79  
## props$log\_s\_freq\_lg10WF\_SUBTLEXNL ~ props$s\_Auditory + props$s\_Visual  
##   
## Df Sum of Sq RSS AIC F Value Pr(F)   
## <none> 28.599 -665.79   
## + props$s\_Haptic 1 0.00042 28.599 -663.79 0.0042 0.948236   
## - props$s\_Visual 1 0.97040 29.570 -658.11 9.7383 0.001989 \*\*  
## - props$s\_Auditory 1 0.98769 29.587 -657.94 9.9118 0.001815 \*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

summary(fit\_freq\_lg10WF\_SUBTLEXNL\_props)

##   
## Call:  
## lm(formula = props$log\_s\_freq\_lg10WF\_SUBTLEXNL ~ props$s\_Auditory +   
## props$s\_Haptic + props$s\_Visual, data = props)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.67800 -0.23858 -0.03592 0.25485 0.76836   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 1.036229 0.018605 55.697 < 2e-16 \*\*\*  
## props$s\_Auditory -0.070229 0.022537 -3.116 0.00202 \*\*   
## props$s\_Haptic -0.001236 0.019017 -0.065 0.94824   
## props$s\_Visual 0.067827 0.021769 3.116 0.00202 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.3162 on 286 degrees of freedom  
## (53 observations deleted due to missingness)  
## Multiple R-squared: 0.1224, Adjusted R-squared: 0.1132   
## F-statistic: 13.3 on 3 and 286 DF, p-value: 3.75e-08

# freq: CELEX log-10 lemma WF  
fit\_freq\_CELEX\_lem\_props <- lm(props$s\_freq\_CELEX\_lem ~ props$s\_Auditory +   
props$s\_Haptic + props$s\_Visual, data = props)  
stat.desc(fit\_freq\_CELEX\_lem\_props$residuals, norm = TRUE)

## x  
## nbr.val 2.470000e+02  
## nbr.null 0.000000e+00  
## nbr.na 0.000000e+00  
## min -1.514183e+00  
## max 2.434544e+00  
## range 3.948728e+00  
## sum -9.034440e-15  
## median -1.388975e-01  
## mean -3.669410e-17  
## SE.mean 6.316656e-02  
## CI.mean.0.95 1.244163e-01  
## var 9.855335e-01  
## std.dev 9.927404e-01  
## coef.var -2.705450e+16  
## skewness 3.722828e-01  
## skew.2SE 1.201518e+00  
## kurtosis -8.446284e-01  
## kurt.2SE -1.368342e+00  
## normtest.W 9.590002e-01  
## normtest.p 1.738309e-06

# residuals distribution: skew and kurtosed. Raw scores/2.SE > 1  
# have to log-transform DV and re-run regression  
  
psych::describe(props$s\_freq\_CELEX\_lem)

## vars n mean sd median trimmed mad min max range skew kurtosis  
## X1 1 247 0 1 -0.16 -0.06 1.21 -1.35 2.51 3.86 0.37 -0.84  
## se  
## X1 0.06

props$log\_s\_freq\_CELEX\_lem <- log(3 + props$s\_freq\_CELEX\_lem)  
  
fit\_freq\_CELEX\_lem\_props <- lm(props$log\_s\_freq\_CELEX\_lem ~ props$s\_Auditory +   
props$s\_Haptic + props$s\_Visual, data = props)  
  
# check residuals again  
stat.desc(fit\_freq\_CELEX\_lem\_props$residuals, norm = TRUE)

## x  
## nbr.val 2.470000e+02  
## nbr.null 0.000000e+00  
## nbr.na 0.000000e+00  
## min -6.009444e-01  
## max 6.821853e-01  
## range 1.283130e+00  
## sum -6.834810e-16  
## median 1.096647e-02  
## mean -2.754839e-18  
## SE.mean 2.162772e-02  
## CI.mean.0.95 4.259913e-02  
## var 1.155363e-01  
## std.dev 3.399063e-01  
## coef.var -1.233852e+17  
## skewness -1.081089e-01  
## skew.2SE -3.489141e-01  
## kurtosis -1.047055e+00  
## kurt.2SE -1.696284e+00  
## normtest.W 9.622853e-01  
## normtest.p 4.369197e-06

# same; go back  
fit\_freq\_CELEX\_lem\_props <- lm(props$s\_freq\_CELEX\_lem ~ props$s\_Auditory +   
props$s\_Haptic + props$s\_Visual, data = props)  
  
# Check multicollinearity: largest VIF (pref. < 10), mean VIF (pref. around 1), and   
# tolerance (pref. > 0.2)  
vif(fit\_freq\_CELEX\_lem\_props)

## props$s\_Auditory props$s\_Haptic props$s\_Visual   
## 1.263215 1.013850 1.259341

mean(vif(fit\_freq\_CELEX\_lem\_props))

## [1] 1.178802

1/vif(fit\_freq\_CELEX\_lem\_props)

## props$s\_Auditory props$s\_Haptic props$s\_Visual   
## 0.7916311 0.9863389 0.7940662

# RESULTS: all good  
  
step\_freq\_CELEX\_lem\_props\_AIC <- stepAIC(fit\_freq\_CELEX\_lem\_props, direction="both")

## Start: AIC=3.4  
## props$s\_freq\_CELEX\_lem ~ props$s\_Auditory + props$s\_Haptic +   
## props$s\_Visual  
##   
## Df Sum of Sq RSS AIC  
## - props$s\_Haptic 1 0.2674 242.71 1.6709  
## - props$s\_Auditory 1 0.3161 242.76 1.7205  
## <none> 242.44 3.3986  
## - props$s\_Visual 1 3.3187 245.76 4.7568  
##   
## Step: AIC=1.67  
## props$s\_freq\_CELEX\_lem ~ props$s\_Auditory + props$s\_Visual  
##   
## Df Sum of Sq RSS AIC  
## - props$s\_Auditory 1 0.3623 243.07 0.0393  
## <none> 242.71 1.6709  
## - props$s\_Visual 1 3.2356 245.94 2.9420  
## + props$s\_Haptic 1 0.2674 242.44 3.3986  
##   
## Step: AIC=0.04  
## props$s\_freq\_CELEX\_lem ~ props$s\_Visual  
##   
## Df Sum of Sq RSS AIC  
## <none> 243.07 0.03930  
## - props$s\_Visual 1 2.92911 246.00 0.99797  
## + props$s\_Auditory 1 0.36229 242.71 1.67087  
## + props$s\_Haptic 1 0.31352 242.76 1.72050

step\_freq\_CELEX\_lem\_props\_F <- stepAIC(fit\_freq\_CELEX\_lem\_props, direction="both",   
test="F")

## Start: AIC=3.4  
## props$s\_freq\_CELEX\_lem ~ props$s\_Auditory + props$s\_Haptic +   
## props$s\_Visual  
##   
## Df Sum of Sq RSS AIC F Value Pr(F)   
## - props$s\_Haptic 1 0.2674 242.71 1.6709 0.2680 0.60516   
## - props$s\_Auditory 1 0.3161 242.76 1.7205 0.3169 0.57402   
## <none> 242.44 3.3986   
## - props$s\_Visual 1 3.3187 245.76 4.7568 3.3264 0.06941 .  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Step: AIC=1.67  
## props$s\_freq\_CELEX\_lem ~ props$s\_Auditory + props$s\_Visual  
##   
## Df Sum of Sq RSS AIC F Value Pr(F)   
## - props$s\_Auditory 1 0.3623 243.07 0.0393 0.3642 0.54673   
## <none> 242.71 1.6709   
## - props$s\_Visual 1 3.2356 245.94 2.9420 3.2529 0.07253 .  
## + props$s\_Haptic 1 0.2674 242.44 3.3986 0.2680 0.60516   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Step: AIC=0.04  
## props$s\_freq\_CELEX\_lem ~ props$s\_Visual  
##   
## Df Sum of Sq RSS AIC F Value Pr(F)   
## <none> 243.07 0.03930   
## - props$s\_Visual 1 2.92911 246.00 0.99797 2.95236 0.08702 .  
## + props$s\_Auditory 1 0.36229 242.71 1.67087 0.36422 0.54673   
## + props$s\_Haptic 1 0.31352 242.76 1.72050 0.31513 0.57507   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

summary(fit\_freq\_CELEX\_lem\_props)

##   
## Call:  
## lm(formula = props$s\_freq\_CELEX\_lem ~ props$s\_Auditory + props$s\_Haptic +   
## props$s\_Visual, data = props)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.5142 -0.8016 -0.1389 0.7363 2.4345   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -0.01352 0.06784 -0.199 0.8422   
## props$s\_Auditory 0.05143 0.09137 0.563 0.5740   
## props$s\_Haptic -0.03235 0.06250 -0.518 0.6052   
## props$s\_Visual 0.14572 0.07990 1.824 0.0694 .  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.9988 on 243 degrees of freedom  
## (96 observations deleted due to missingness)  
## Multiple R-squared: 0.01447, Adjusted R-squared: 0.002299   
## F-statistic: 1.189 on 3 and 243 DF, p-value: 0.3146

# length: RC1 lexicals  
fit\_RC1\_lexicals\_props <- lm(props$s\_RC1\_lexicals ~ props$s\_Auditory + props$s\_Haptic   
+ props$s\_Visual, data = props)  
stat.desc(fit\_RC1\_lexicals\_props$residuals, norm = TRUE)

## x  
## nbr.val 1.700000e+02  
## nbr.null 0.000000e+00  
## nbr.na 0.000000e+00  
## min -1.822622e+00  
## max 3.222212e+00  
## range 5.044834e+00  
## sum -3.788636e-15  
## median -1.386471e-01  
## mean -2.226473e-17  
## SE.mean 7.506144e-02  
## CI.mean.0.95 1.481788e-01  
## var 9.578174e-01  
## std.dev 9.786814e-01  
## coef.var -4.395658e+16  
## skewness 7.538996e-01  
## skew.2SE 2.024025e+00  
## kurtosis 2.557611e-01  
## kurt.2SE 3.452572e-01  
## normtest.W 9.558991e-01  
## normtest.p 3.484999e-05

# residuals distribution: skewed. Raw scores/2.SE > 1  
# have to log-transform DV and re-run regression  
  
psych::describe(props$s\_RC1\_lexicals)

## vars n mean sd median trimmed mad min max range skew kurtosis  
## X1 1 170 0 1 -0.11 -0.09 0.93 -2.08 3.63 5.71 0.93 0.94  
## se  
## X1 0.08

props$log\_s\_RC1\_lexicals\_props <- log(4 + props$s\_RC1\_lexicals)  
  
fit\_RC1\_lexicals\_props <- lm(props$log\_s\_RC1\_lexicals ~ props$s\_Auditory +   
props$s\_Haptic + props$s\_Visual, data = props)  
  
# check residuals again  
stat.desc(fit\_RC1\_lexicals\_props$residuals, norm = TRUE)

## x  
## nbr.val 1.700000e+02  
## nbr.null 0.000000e+00  
## nbr.na 0.000000e+00  
## min -6.408548e-01  
## max 6.186645e-01  
## range 1.259519e+00  
## sum 6.973588e-16  
## median -9.930210e-03  
## mean 4.093142e-18  
## SE.mean 1.804027e-02  
## CI.mean.0.95 3.561330e-02  
## var 5.532672e-02  
## std.dev 2.352163e-01  
## coef.var 5.746595e+16  
## skewness 1.943265e-01  
## skew.2SE 5.217161e-01  
## kurtosis -4.515675e-01  
## kurt.2SE -6.095802e-01  
## normtest.W 9.897230e-01  
## normtest.p 2.566697e-01

# good!  
  
# Check multicollinearity: largest VIF (pref. < 10), mean VIF (pref. around 1), and   
# tolerance (pref. > 0.2)  
vif(fit\_RC1\_lexicals\_props)

## props$s\_Auditory props$s\_Haptic props$s\_Visual   
## 1.139419 1.002843 1.141874

mean(vif(fit\_RC1\_lexicals\_props))

## [1] 1.094712

1/vif(fit\_RC1\_lexicals\_props)

## props$s\_Auditory props$s\_Haptic props$s\_Visual   
## 0.8776401 0.9971650 0.8757534

# RESULTS: all good  
  
step\_RC1\_lexicals\_props\_AIC <- stepAIC(fit\_RC1\_lexicals\_props, direction="both")

## Start: AIC=-485.07  
## props$log\_s\_RC1\_lexicals ~ props$s\_Auditory + props$s\_Haptic +   
## props$s\_Visual  
##   
## Df Sum of Sq RSS AIC  
## - props$s\_Haptic 1 0.075722 9.4259 -485.70  
## - props$s\_Visual 1 0.080229 9.4304 -485.62  
## <none> 9.3502 -485.07  
## - props$s\_Auditory 1 0.246461 9.5967 -482.64  
##   
## Step: AIC=-485.7  
## props$log\_s\_RC1\_lexicals ~ props$s\_Auditory + props$s\_Visual  
##   
## Df Sum of Sq RSS AIC  
## - props$s\_Visual 1 0.088911 9.5148 -486.10  
## <none> 9.4259 -485.70  
## + props$s\_Haptic 1 0.075722 9.3502 -485.07  
## - props$s\_Auditory 1 0.253554 9.6795 -483.18  
##   
## Step: AIC=-486.1  
## props$log\_s\_RC1\_lexicals ~ props$s\_Auditory  
##   
## Df Sum of Sq RSS AIC  
## <none> 9.5148 -486.10  
## + props$s\_Visual 1 0.088911 9.4259 -485.70  
## + props$s\_Haptic 1 0.084404 9.4304 -485.62  
## - props$s\_Auditory 1 0.181715 9.6966 -484.88

step\_RC1\_lexicals\_props\_F <- stepAIC(fit\_RC1\_lexicals\_props, direction="both",   
test="F")

## Start: AIC=-485.07  
## props$log\_s\_RC1\_lexicals ~ props$s\_Auditory + props$s\_Haptic +   
## props$s\_Visual  
##   
## Df Sum of Sq RSS AIC F Value Pr(F)   
## - props$s\_Haptic 1 0.075722 9.4259 -485.70 1.3443 0.24794   
## - props$s\_Visual 1 0.080229 9.4304 -485.62 1.4243 0.23439   
## <none> 9.3502 -485.07   
## - props$s\_Auditory 1 0.246461 9.5967 -482.64 4.3756 0.03798 \*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Step: AIC=-485.7  
## props$log\_s\_RC1\_lexicals ~ props$s\_Auditory + props$s\_Visual  
##   
## Df Sum of Sq RSS AIC F Value Pr(F)   
## - props$s\_Visual 1 0.088911 9.5148 -486.10 1.5752 0.21120   
## <none> 9.4259 -485.70   
## + props$s\_Haptic 1 0.075722 9.3502 -485.07 1.3443 0.24794   
## - props$s\_Auditory 1 0.253554 9.6795 -483.18 4.4922 0.03553 \*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Step: AIC=-486.1  
## props$log\_s\_RC1\_lexicals ~ props$s\_Auditory  
##   
## Df Sum of Sq RSS AIC F Value Pr(F)   
## <none> 9.5148 -486.10   
## + props$s\_Visual 1 0.088911 9.4259 -485.70 1.5752 0.21120   
## + props$s\_Haptic 1 0.084404 9.4304 -485.62 1.4947 0.22322   
## - props$s\_Auditory 1 0.181715 9.6966 -484.88 3.2085 0.07506 .  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

summary(fit\_RC1\_lexicals\_props)

##   
## Call:  
## lm(formula = props$log\_s\_RC1\_lexicals ~ props$s\_Auditory + props$s\_Haptic +   
## props$s\_Visual, data = props)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.64085 -0.18979 -0.00993 0.15912 0.61866   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 1.37128 0.02076 66.043 <2e-16 \*\*\*  
## props$s\_Auditory 0.05991 0.02864 2.092 0.038 \*   
## props$s\_Haptic -0.02067 0.01783 -1.159 0.248   
## props$s\_Visual 0.02760 0.02313 1.193 0.234   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.2373 on 166 degrees of freedom  
## (173 observations deleted due to missingness)  
## Multiple R-squared: 0.03572, Adjusted R-squared: 0.01829   
## F-statistic: 2.05 on 3 and 166 DF, p-value: 0.1089

# distinctiveness: RC3 lexicals  
fit\_RC3\_lexicals\_props <- lm(props$s\_RC3\_lexicals ~ props$s\_Auditory +   
props$s\_Haptic + props$s\_Visual, data = props)  
stat.desc(fit\_RC3\_lexicals\_props$residuals, norm = TRUE)

## x  
## nbr.val 1.700000e+02  
## nbr.null 0.000000e+00  
## nbr.na 0.000000e+00  
## min -1.355748e+00  
## max 4.005520e+00  
## range 5.361267e+00  
## sum 2.810252e-15  
## median -2.582047e-01  
## mean 1.657107e-17  
## SE.mean 7.489141e-02  
## CI.mean.0.95 1.478432e-01  
## var 9.534830e-01  
## std.dev 9.764646e-01  
## coef.var 5.892585e+16  
## skewness 1.582698e+00  
## skew.2SE 4.249133e+00  
## kurtosis 2.764799e+00  
## kurt.2SE 3.732258e+00  
## normtest.W 8.564774e-01  
## normtest.p 1.279852e-11

# residuals distribution: skewed and kurtosed. Raw scores/2.SE > 1  
# have to log-transform DV and re-run regression  
  
psych::describe(props$s\_RC3\_lexicals)

## vars n mean sd median trimmed mad min max range skew kurtosis  
## X1 1 170 0 1 -0.37 -0.17 0.57 -1.16 4.03 5.19 1.62 2.42  
## se  
## X1 0.08

props$log\_s\_RC3\_lexicals <- log(3 + props$s\_RC3\_lexicals)  
  
fit\_RC3\_lexicals\_props <- lm(props$log\_s\_RC3\_lexicals ~ props$s\_Auditory +   
props$s\_Haptic + props$s\_Visual, data = props)  
  
# check residuals again  
stat.desc(fit\_RC3\_lexicals\_props$residuals, norm = TRUE)

## x  
## nbr.val 1.700000e+02  
## nbr.null 0.000000e+00  
## nbr.na 0.000000e+00  
## min -4.797756e-01  
## max 9.240221e-01  
## range 1.403798e+00  
## sum -5.325601e-16  
## median -7.295993e-02  
## mean -3.147455e-18  
## SE.mean 2.151459e-02  
## CI.mean.0.95 4.247196e-02  
## var 7.868919e-02  
## std.dev 2.805159e-01  
## coef.var -8.912469e+16  
## skewness 9.224203e-01  
## skew.2SE 2.476459e+00  
## kurtosis 4.920814e-01  
## kurt.2SE 6.642707e-01  
## normtest.W 9.373994e-01  
## normtest.p 8.913216e-07

# quite better  
  
# Check multicollinearity: largest VIF (pref. < 10), mean VIF (pref. around 1), and   
# tolerance (pref. > 0.2)  
vif(fit\_RC3\_lexicals\_props)

## props$s\_Auditory props$s\_Haptic props$s\_Visual   
## 1.139419 1.002843 1.141874

mean(vif(fit\_RC3\_lexicals\_props))

## [1] 1.094712

1/vif(fit\_RC3\_lexicals\_props)

## props$s\_Auditory props$s\_Haptic props$s\_Visual   
## 0.8776401 0.9971650 0.8757534

# RESULTS: all good  
  
step\_RC3\_lexicals\_props\_AIC <- stepAIC(fit\_RC3\_lexicals\_props, direction="both")

## Start: AIC=-425.19  
## props$log\_s\_RC3\_lexicals ~ props$s\_Auditory + props$s\_Haptic +   
## props$s\_Visual  
##   
## Df Sum of Sq RSS AIC  
## - props$s\_Haptic 1 0.01036 13.309 -427.05  
## <none> 13.299 -425.19  
## - props$s\_Auditory 1 0.17132 13.470 -425.01  
## - props$s\_Visual 1 0.59945 13.898 -419.69  
##   
## Step: AIC=-427.05  
## props$log\_s\_RC3\_lexicals ~ props$s\_Auditory + props$s\_Visual  
##   
## Df Sum of Sq RSS AIC  
## <none> 13.309 -427.05  
## - props$s\_Auditory 1 0.17356 13.482 -426.85  
## + props$s\_Haptic 1 0.01036 13.299 -425.19  
## - props$s\_Visual 1 0.60949 13.918 -421.44

step\_RC3\_lexicals\_props\_F <- stepAIC(fit\_RC3\_lexicals\_props, direction="both",   
test="F")

## Start: AIC=-425.19  
## props$log\_s\_RC3\_lexicals ~ props$s\_Auditory + props$s\_Haptic +   
## props$s\_Visual  
##   
## Df Sum of Sq RSS AIC F Value Pr(F)   
## - props$s\_Haptic 1 0.01036 13.309 -427.05 0.1293 0.719589   
## <none> 13.299 -425.19   
## - props$s\_Auditory 1 0.17132 13.470 -425.01 2.1386 0.145527   
## - props$s\_Visual 1 0.59945 13.898 -419.69 7.4828 0.006907 \*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Step: AIC=-427.05  
## props$log\_s\_RC3\_lexicals ~ props$s\_Auditory + props$s\_Visual  
##   
## Df Sum of Sq RSS AIC F Value Pr(F)   
## <none> 13.309 -427.05   
## - props$s\_Auditory 1 0.17356 13.482 -426.85 2.1779 0.141891   
## + props$s\_Haptic 1 0.01036 13.299 -425.19 0.1293 0.719589   
## - props$s\_Visual 1 0.60949 13.918 -421.44 7.6480 0.006323 \*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

summary(fit\_RC3\_lexicals\_props)

##   
## Call:  
## lm(formula = props$log\_s\_RC3\_lexicals ~ props$s\_Auditory + props$s\_Haptic +   
## props$s\_Visual, data = props)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.47978 -0.20764 -0.07296 0.13226 0.92402   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 1.058987 0.024762 42.766 < 2e-16 \*\*\*  
## props$s\_Auditory -0.049950 0.034157 -1.462 0.14553   
## props$s\_Haptic 0.007645 0.021259 0.360 0.71959   
## props$s\_Visual -0.075441 0.027579 -2.735 0.00691 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.283 on 166 degrees of freedom  
## (173 observations deleted due to missingness)  
## Multiple R-squared: 0.04616, Adjusted R-squared: 0.02892   
## F-statistic: 2.678 on 3 and 166 DF, p-value: 0.04882

# freq: RC2 lexicals  
fit\_RC2\_lexicals\_props <- lm(props$s\_RC2\_lexicals ~ props$s\_Auditory + props$s\_Haptic  
 + props$s\_Visual, data = props)  
stat.desc(fit\_RC2\_lexicals\_props$residuals, norm = TRUE)

## x  
## nbr.val 1.700000e+02  
## nbr.null 0.000000e+00  
## nbr.na 0.000000e+00  
## min -2.521274e+00  
## max 2.428216e+00  
## range 4.949490e+00  
## sum -3.816392e-15  
## median -1.168067e-01  
## mean -2.235880e-17  
## SE.mean 7.481168e-02  
## CI.mean.0.95 1.476858e-01  
## var 9.514538e-01  
## std.dev 9.754249e-01  
## coef.var -4.362600e+16  
## skewness 2.099120e-01  
## skew.2SE 5.635592e-01  
## kurtosis -7.566392e-01  
## kurt.2SE -1.021403e+00  
## normtest.W 9.788257e-01  
## normtest.p 1.063544e-02

# residuals distribution: kurtosed. Raw scores/2.SE < 1  
# have to log-transform DV and re-run regression  
  
psych::describe(props$s\_RC2\_lexicals)

## vars n mean sd median trimmed mad min max range skew kurtosis  
## X1 1 170 0 1 -0.02 -0.03 1.15 -2.37 2.29 4.66 0.2 -0.67  
## se  
## X1 0.08

props$log\_s\_RC2\_lexicals <- log(3 + props$s\_RC2\_lexicals)  
  
fit\_RC2\_lexicals\_props <- lm(props$log\_s\_RC2\_lexicals ~ props$s\_Auditory +   
props$s\_Haptic + props$s\_Visual, data = props)  
  
# check residuals again  
stat.desc(fit\_RC2\_lexicals\_props$residuals, norm = TRUE)

## x  
## nbr.val 1.700000e+02  
## nbr.null 0.000000e+00  
## nbr.na 0.000000e+00  
## min -1.548065e+00  
## max 7.294142e-01  
## range 2.277479e+00  
## sum 8.968520e-16  
## median 1.992494e-02  
## mean 5.315142e-18  
## SE.mean 2.746269e-02  
## CI.mean.0.95 5.421412e-02  
## var 1.282139e-01  
## std.dev 3.580697e-01  
## coef.var 6.736786e+16  
## skewness -6.661503e-01  
## skew.2SE -1.788441e+00  
## kurtosis 9.513419e-01  
## kurt.2SE 1.284236e+00  
## normtest.W 9.666991e-01  
## normtest.p 4.244338e-04

# worse; back  
fit\_RC2\_lexicals\_props <- lm(props$s\_RC2\_lexicals ~ props$s\_Auditory + props$s\_Haptic  
 + props$s\_Visual, data = props)  
  
# Check multicollinearity: largest VIF (pref. < 10), mean VIF (pref. around 1), and   
# tolerance (pref. > 0.2)  
vif(fit\_RC2\_lexicals\_props)

## props$s\_Auditory props$s\_Haptic props$s\_Visual   
## 1.139419 1.002843 1.141874

mean(vif(fit\_RC2\_lexicals\_props))

## [1] 1.094712

1/vif(fit\_RC2\_lexicals\_props)

## props$s\_Auditory props$s\_Haptic props$s\_Visual   
## 0.8776401 0.9971650 0.8757534

# RESULTS: all good  
  
step\_RC2\_lexicals\_props\_AIC <- stepAIC(fit\_RC2\_lexicals\_props, direction="both")

## Start: AIC=-1.46  
## props$s\_RC2\_lexicals ~ props$s\_Auditory + props$s\_Haptic + props$s\_Visual  
##   
## Df Sum of Sq RSS AIC  
## - props$s\_Haptic 1 0.8921 161.69 -2.5223  
## <none> 160.80 -1.4629  
## - props$s\_Visual 1 3.7779 164.57 0.4851  
## - props$s\_Auditory 1 5.6714 166.47 2.4298  
##   
## Step: AIC=-2.52  
## props$s\_RC2\_lexicals ~ props$s\_Auditory + props$s\_Visual  
##   
## Df Sum of Sq RSS AIC  
## <none> 161.69 -2.52228  
## + props$s\_Haptic 1 0.8921 160.80 -1.46286  
## - props$s\_Visual 1 3.9852 165.67 -0.38301  
## - props$s\_Auditory 1 5.7890 167.48 1.45786

step\_RC2\_lexicals\_props\_F <- stepAIC(fit\_RC2\_lexicals\_props, direction="both",   
test="F")

## Start: AIC=-1.46  
## props$s\_RC2\_lexicals ~ props$s\_Auditory + props$s\_Haptic + props$s\_Visual  
##   
## Df Sum of Sq RSS AIC F Value Pr(F)   
## - props$s\_Haptic 1 0.8921 161.69 -2.5223 0.9210 0.33861   
## <none> 160.80 -1.4629   
## - props$s\_Visual 1 3.7779 164.57 0.4851 3.9002 0.04994 \*  
## - props$s\_Auditory 1 5.6714 166.47 2.4298 5.8549 0.01661 \*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Step: AIC=-2.52  
## props$s\_RC2\_lexicals ~ props$s\_Auditory + props$s\_Visual  
##   
## Df Sum of Sq RSS AIC F Value Pr(F)   
## <none> 161.69 -2.52228   
## + props$s\_Haptic 1 0.8921 160.80 -1.46286 0.9210 0.33861   
## - props$s\_Visual 1 3.9852 165.67 -0.38301 4.1161 0.04406 \*  
## - props$s\_Auditory 1 5.7890 167.48 1.45786 5.9792 0.01551 \*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

summary(fit\_RC2\_lexicals\_props)

##   
## Call:  
## lm(formula = props$s\_RC2\_lexicals ~ props$s\_Auditory + props$s\_Haptic +   
## props$s\_Visual, data = props)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -2.5213 -0.8128 -0.1168 0.7561 2.4282   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.04703 0.08610 0.546 0.5856   
## props$s\_Auditory 0.28739 0.11877 2.420 0.0166 \*  
## props$s\_Haptic -0.07094 0.07392 -0.960 0.3386   
## props$s\_Visual 0.18939 0.09590 1.975 0.0499 \*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.9842 on 166 degrees of freedom  
## (173 observations deleted due to missingness)  
## Multiple R-squared: 0.04855, Adjusted R-squared: 0.03135   
## F-statistic: 2.823 on 3 and 166 DF, p-value: 0.04046

# additional var: age of acquisition  
fit\_AoA\_Brysbaertetal2014\_props <- lm(props$s\_AoA\_Brysbaertetal2014 ~   
props$s\_Auditory + props$s\_Haptic + props$s\_Visual, data = props)  
stat.desc(fit\_AoA\_Brysbaertetal2014\_props$residuals, norm = TRUE)

## x  
## nbr.val 2.330000e+02  
## nbr.null 0.000000e+00  
## nbr.na 0.000000e+00  
## min -2.112829e+00  
## max 2.593592e+00  
## range 4.706422e+00  
## sum 2.532696e-16  
## median 3.452361e-02  
## mean 1.144694e-18  
## SE.mean 6.406750e-02  
## CI.mean.0.95 1.262285e-01  
## var 9.563821e-01  
## std.dev 9.779479e-01  
## coef.var 8.543312e+17  
## skewness 7.860513e-02  
## skew.2SE 2.464866e-01  
## kurtosis -4.681725e-01  
## kurt.2SE -7.370869e-01  
## normtest.W 9.881971e-01  
## normtest.p 5.248873e-02

# residuals distribution: good  
  
# Check multicollinearity: largest VIF (pref. < 10), mean VIF (pref. around 1), and   
# tolerance (pref. > 0.2)  
vif(fit\_AoA\_Brysbaertetal2014\_props)

## props$s\_Auditory props$s\_Haptic props$s\_Visual   
## 1.292539 1.013180 1.284802

mean(vif(fit\_AoA\_Brysbaertetal2014\_props))

## [1] 1.19684

1/vif(fit\_AoA\_Brysbaertetal2014\_props)

## props$s\_Auditory props$s\_Haptic props$s\_Visual   
## 0.7736713 0.9869918 0.7783300

# RESULTS: all good  
  
step\_AoA\_Brysbaertetal2014\_props\_AIC <- stepAIC(fit\_AoA\_Brysbaertetal2014\_props,  
direction="both")

## Start: AIC=-3.39  
## props$s\_AoA\_Brysbaertetal2014 ~ props$s\_Auditory + props$s\_Haptic +   
## props$s\_Visual  
##   
## Df Sum of Sq RSS AIC  
## - props$s\_Haptic 1 0.0101 221.89 -5.3829  
## - props$s\_Auditory 1 0.1463 222.03 -5.2398  
## <none> 221.88 -3.3934  
## - props$s\_Visual 1 6.7408 228.62 1.5798  
##   
## Step: AIC=-5.38  
## props$s\_AoA\_Brysbaertetal2014 ~ props$s\_Auditory + props$s\_Visual  
##   
## Df Sum of Sq RSS AIC  
## - props$s\_Auditory 1 0.1538 222.04 -7.2214  
## <none> 221.89 -5.3829  
## + props$s\_Haptic 1 0.0101 221.88 -3.3934  
## - props$s\_Visual 1 6.7630 228.65 -0.3874  
##   
## Step: AIC=-7.22  
## props$s\_AoA\_Brysbaertetal2014 ~ props$s\_Visual  
##   
## Df Sum of Sq RSS AIC  
## <none> 222.04 -7.2214  
## + props$s\_Auditory 1 0.1538 221.89 -5.3829  
## + props$s\_Haptic 1 0.0176 222.03 -5.2398  
## - props$s\_Visual 1 9.9555 232.00 0.9978

step\_AoA\_Brysbaertetal2014\_props\_F <- stepAIC(fit\_AoA\_Brysbaertetal2014\_props,   
direction="both", test="F")

## Start: AIC=-3.39  
## props$s\_AoA\_Brysbaertetal2014 ~ props$s\_Auditory + props$s\_Haptic +   
## props$s\_Visual  
##   
## Df Sum of Sq RSS AIC F Value Pr(F)   
## - props$s\_Haptic 1 0.0101 221.89 -5.3829 0.0104 0.918933   
## - props$s\_Auditory 1 0.1463 222.03 -5.2398 0.1510 0.697929   
## <none> 221.88 -3.3934   
## - props$s\_Visual 1 6.7408 228.62 1.5798 6.9571 0.008921 \*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Step: AIC=-5.38  
## props$s\_AoA\_Brysbaertetal2014 ~ props$s\_Auditory + props$s\_Visual  
##   
## Df Sum of Sq RSS AIC F Value Pr(F)   
## - props$s\_Auditory 1 0.1538 222.04 -7.2214 0.1594 0.690035   
## <none> 221.89 -5.3829   
## + props$s\_Haptic 1 0.0101 221.88 -3.3934 0.0104 0.918933   
## - props$s\_Visual 1 6.7630 228.65 -0.3874 7.0101 0.008666 \*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Step: AIC=-7.22  
## props$s\_AoA\_Brysbaertetal2014 ~ props$s\_Visual  
##   
## Df Sum of Sq RSS AIC F Value Pr(F)   
## <none> 222.04 -7.2214   
## + props$s\_Auditory 1 0.1538 221.89 -5.3829 0.1594 0.690035   
## + props$s\_Haptic 1 0.0176 222.03 -5.2398 0.0182 0.892810   
## - props$s\_Visual 1 9.9555 232.00 0.9978 10.3570 0.001475 \*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

summary(fit\_AoA\_Brysbaertetal2014\_props)

##   
## Call:  
## lm(formula = props$s\_AoA\_Brysbaertetal2014 ~ props$s\_Auditory +   
## props$s\_Haptic + props$s\_Visual, data = props)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -2.11283 -0.73187 0.03452 0.64255 2.59359   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.054540 0.067911 0.803 0.42274   
## props$s\_Auditory 0.035560 0.091508 0.389 0.69793   
## props$s\_Haptic -0.006453 0.063334 -0.102 0.91893   
## props$s\_Visual -0.209357 0.079373 -2.638 0.00892 \*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.9843 on 229 degrees of freedom  
## (110 observations deleted due to missingness)  
## Multiple R-squared: 0.04362, Adjusted R-squared: 0.03109   
## F-statistic: 3.481 on 3 and 229 DF, p-value: 0.01667

# additional var: concreteness  
fit\_concrete\_Brysbaertetal2014\_props <- lm(props$s\_concrete\_Brysbaertetal2014 ~   
props$s\_Auditory + props$s\_Haptic + props$s\_Visual, data = props)  
stat.desc(fit\_concrete\_Brysbaertetal2014\_props$residuals, norm = TRUE)

## x  
## nbr.val 2.330000e+02  
## nbr.null 0.000000e+00  
## nbr.na 0.000000e+00  
## min -2.594543e+00  
## max 1.852867e+00  
## range 4.447410e+00  
## sum -2.477185e-14  
## median 1.768909e-01  
## mean -1.062963e-16  
## SE.mean 6.295441e-02  
## CI.mean.0.95 1.240354e-01  
## var 9.234389e-01  
## std.dev 9.609573e-01  
## coef.var -9.040369e+15  
## skewness -4.207250e-01  
## skew.2SE -1.319291e+00  
## kurtosis -3.351918e-01  
## kurt.2SE -5.277232e-01  
## normtest.W 9.790502e-01  
## normtest.p 1.581048e-03

# residuals distribution: skew. Raw scores/2.SE > 1  
# have to log-transform DV and re-run regression  
  
psych::describe(props$s\_concrete\_Brysbaertetal2014)

## vars n mean sd median trimmed mad min max range skew kurtosis  
## X1 1 233 0 1 0.19 0.04 0.99 -2.76 1.99 4.76 -0.37 -0.41  
## se  
## X1 0.07

props$log\_s\_concrete\_Brysbaertetal2014 <- log(4 + props$s\_concrete\_Brysbaertetal2014)  
  
fit\_concrete\_Brysbaertetal2014\_props <- lm(props$log\_s\_concrete\_Brysbaertetal2014 ~   
props$s\_Auditory + props$s\_Haptic + props$s\_Visual, data = props)  
  
# check residuals again  
stat.desc(fit\_concrete\_Brysbaertetal2014\_props$residuals, norm = TRUE)

## x  
## nbr.val 2.330000e+02  
## nbr.null 0.000000e+00  
## nbr.na 0.000000e+00  
## min -1.084648e+00  
## max 4.612322e-01  
## range 1.545880e+00  
## sum -4.888451e-15  
## median 8.076837e-02  
## mean -2.101351e-17  
## SE.mean 1.814043e-02  
## CI.mean.0.95 3.574103e-02  
## var 7.667449e-02  
## std.dev 2.769016e-01  
## coef.var -1.317731e+16  
## skewness -1.152705e+00  
## skew.2SE -3.614602e+00  
## kurtosis 1.563516e+00  
## kurt.2SE 2.461586e+00  
## normtest.W 9.220715e-01  
## normtest.p 1.003247e-09

# worse; back  
fit\_concrete\_Brysbaertetal2014\_props <- lm(props$s\_concrete\_Brysbaertetal2014 ~   
props$s\_Auditory + props$s\_Haptic + props$s\_Visual, data = props)  
  
# Check multicollinearity: largest VIF (pref. < 10), mean VIF (pref. around 1), and   
# tolerance (pref. > 0.2)  
vif(fit\_concrete\_Brysbaertetal2014\_props)

## props$s\_Auditory props$s\_Haptic props$s\_Visual   
## 1.292539 1.013180 1.284802

mean(vif(fit\_concrete\_Brysbaertetal2014\_props))

## [1] 1.19684

1/vif(fit\_concrete\_Brysbaertetal2014\_props)

## props$s\_Auditory props$s\_Haptic props$s\_Visual   
## 0.7736713 0.9869918 0.7783300

# RESULTS: all good  
  
step\_concrete\_Brysbaertetal2014\_props\_AIC <-   
stepAIC(fit\_concrete\_Brysbaertetal2014\_props, direction="both")

## Start: AIC=-11.56  
## props$s\_concrete\_Brysbaertetal2014 ~ props$s\_Auditory + props$s\_Haptic +   
## props$s\_Visual  
##   
## Df Sum of Sq RSS AIC  
## - props$s\_Haptic 1 1.2182 215.46 -12.2396  
## <none> 214.24 -11.5607  
## - props$s\_Auditory 1 3.2391 217.48 -10.0643  
## - props$s\_Visual 1 5.0553 219.29 -8.1266  
##   
## Step: AIC=-12.24  
## props$s\_concrete\_Brysbaertetal2014 ~ props$s\_Auditory + props$s\_Visual  
##   
## Df Sum of Sq RSS AIC  
## <none> 215.46 -12.240  
## + props$s\_Haptic 1 1.2182 214.24 -11.561  
## - props$s\_Auditory 1 3.6026 219.06 -10.376  
## - props$s\_Visual 1 5.2123 220.67 -8.670

step\_concrete\_Brysbaertetal2014\_props\_F <-   
stepAIC(fit\_concrete\_Brysbaertetal2014\_props, direction="both", test="F")

## Start: AIC=-11.56  
## props$s\_concrete\_Brysbaertetal2014 ~ props$s\_Auditory + props$s\_Haptic +   
## props$s\_Visual  
##   
## Df Sum of Sq RSS AIC F Value Pr(F)   
## - props$s\_Haptic 1 1.2182 215.46 -12.2396 1.3022 0.25501   
## <none> 214.24 -11.5607   
## - props$s\_Auditory 1 3.2391 217.48 -10.0643 3.4623 0.06406 .  
## - props$s\_Visual 1 5.0553 219.29 -8.1266 5.4036 0.02097 \*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Step: AIC=-12.24  
## props$s\_concrete\_Brysbaertetal2014 ~ props$s\_Auditory + props$s\_Visual  
##   
## Df Sum of Sq RSS AIC F Value Pr(F)   
## <none> 215.46 -12.240   
## + props$s\_Haptic 1 1.2182 214.24 -11.561 1.3022 0.25501   
## - props$s\_Auditory 1 3.6026 219.06 -10.376 3.8457 0.05108 .  
## - props$s\_Visual 1 5.2123 220.67 -8.670 5.5641 0.01917 \*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

summary(fit\_concrete\_Brysbaertetal2014\_props)

##   
## Call:  
## lm(formula = props$s\_concrete\_Brysbaertetal2014 ~ props$s\_Auditory +   
## props$s\_Haptic + props$s\_Visual, data = props)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -2.5945 -0.6710 0.1769 0.6787 1.8529   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -0.08550 0.06673 -1.281 0.2014   
## props$s\_Auditory -0.16731 0.08992 -1.861 0.0641 .  
## props$s\_Haptic 0.07102 0.06223 1.141 0.2550   
## props$s\_Visual 0.18130 0.07799 2.325 0.0210 \*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.9672 on 229 degrees of freedom  
## (110 observations deleted due to missingness)  
## Multiple R-squared: 0.07656, Adjusted R-squared: 0.06446   
## F-statistic: 6.329 on 3 and 229 DF, p-value: 0.0003852

# RESULTS: iconicity properties:   
# Auditory strength either was the strongest predictor or presented an opposite   
# polarity from the main predictor. This held for all lexical DVs except age of   
# acquisition.  
  
  
  
# \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
  
# Iconicity within concepts alone, as in Lynott and Connell (2013)  
  
concs <- all[all$cat == 'conc' & c(all$normed == 'Dutch' | all$normed == 'Dut\_Eng'),]  
nrow(concs)

## [1] 411

# There aren't lexical data for every single word.  
# Percentage of concepts per lexical variable (from items w/ Dutch norms)  
describe(complete.cases(concs[complete.cases(concs$Exclusivity),]  
$phonemes\_DUTCHPOND))

## complete.cases(concs[complete.cases(concs$Exclusivity), ]$phonemes\_DUTCHPOND)   
## n missing unique   
## 411 0 2   
##   
## FALSE (22, 5%), TRUE (389, 95%)

describe(complete.cases(concs[complete.cases(concs$Exclusivity),]  
$phon\_neighbours\_DUTCHPOND))

## complete.cases(concs[complete.cases(concs$Exclusivity), ]$phon\_neighbours\_DUTCHPOND)   
## n missing unique value   
## 411 0 1 TRUE

describe(complete.cases(concs[complete.cases(concs$Exclusivity),]  
$orth\_neighbours\_DUTCHPOND))

## complete.cases(concs[complete.cases(concs$Exclusivity), ]$orth\_neighbours\_DUTCHPOND)   
## n missing unique value   
## 411 0 1 TRUE

describe(complete.cases(concs[complete.cases(concs$Exclusivity),]  
$freq\_lg10CD\_SUBTLEXNL))

## complete.cases(concs[complete.cases(concs$Exclusivity), ]$freq\_lg10CD\_SUBTLEXNL)   
## n missing unique   
## 411 0 2   
##   
## FALSE (4, 1%), TRUE (407, 99%)

describe(complete.cases(concs[complete.cases(concs$Exclusivity),]  
$freq\_lg10WF\_SUBTLEXNL))

## complete.cases(concs[complete.cases(concs$Exclusivity), ]$freq\_lg10WF\_SUBTLEXNL)   
## n missing unique   
## 411 0 2   
##   
## FALSE (4, 1%), TRUE (407, 99%)

describe(complete.cases(concs[complete.cases(concs$Exclusivity),]  
$freq\_CELEX\_lem))

## complete.cases(concs[complete.cases(concs$Exclusivity), ]$freq\_CELEX\_lem)   
## n missing unique   
## 411 0 2   
##   
## FALSE (12, 3%), TRUE (399, 97%)

describe(complete.cases(concs[complete.cases(concs$Exclusivity),]  
$AoA\_Brysbaertetal2014))

## complete.cases(concs[complete.cases(concs$Exclusivity), ]$AoA\_Brysbaertetal2014)   
## n missing unique   
## 411 0 2   
##   
## FALSE (13, 3%), TRUE (398, 97%)

describe(complete.cases(concs[complete.cases(concs$Exclusivity),]  
$concrete\_Brysbaertetal2014))

## complete.cases(concs[complete.cases(concs$Exclusivity), ]$concrete\_Brysbaertetal2014)   
## n missing unique   
## 411 0 2   
##   
## FALSE (13, 3%), TRUE (398, 97%)

# M, SD  
stat.desc(concs$letters)

## nbr.val nbr.null nbr.na min max   
## 411.0000000 0.0000000 0.0000000 3.0000000 17.0000000   
## range sum median mean SE.mean   
## 14.0000000 2759.0000000 6.0000000 6.7128954 0.1254065   
## CI.mean.0.95 var std.dev coef.var   
## 0.2465199 6.4637114 2.5423830 0.3787312

stat.desc(concs$phonemes\_DUTCHPOND)

## nbr.val nbr.null nbr.na min max   
## 389.0000000 0.0000000 22.0000000 2.0000000 15.0000000   
## range sum median mean SE.mean   
## 13.0000000 2265.0000000 6.0000000 5.8226221 0.1104353   
## CI.mean.0.95 var std.dev coef.var   
## 0.2171266 4.7442292 2.1781252 0.3740798

stat.desc(concs$phon\_neighbours\_DUTCHPOND)

## nbr.val nbr.null nbr.na min max   
## 411.0000000 139.0000000 0.0000000 0.0000000 49.0000000   
## range sum median mean SE.mean   
## 49.0000000 2273.0000000 1.0000000 5.5304136 0.4080256   
## CI.mean.0.95 var std.dev coef.var   
## 0.8020832 68.4252923 8.2719582 1.4957214

stat.desc(concs$orth\_neighbours\_DUTCHPOND)

## nbr.val nbr.null nbr.na min max   
## 411.0000000 143.0000000 0.0000000 0.0000000 32.0000000   
## range sum median mean SE.mean   
## 32.0000000 1658.0000000 1.0000000 4.0340633 0.2825354   
## CI.mean.0.95 var std.dev coef.var   
## 0.5553987 32.8085930 5.7278786 1.4198782

stat.desc(concs$freq\_lg10CD\_SUBTLEXNL)

## nbr.val nbr.null nbr.na min max   
## 4.070000e+02 0.000000e+00 4.000000e+00 6.000000e-01 3.900000e+00   
## range sum median mean SE.mean   
## 3.300000e+00 1.080720e+03 2.700000e+00 2.655332e+00 3.275722e-02   
## CI.mean.0.95 var std.dev coef.var   
## 6.439494e-02 4.367254e-01 6.608521e-01 2.488774e-01

stat.desc(concs$freq\_lg10WF\_SUBTLEXNL)

## nbr.val nbr.null nbr.na min max   
## 4.070000e+02 0.000000e+00 4.000000e+00 6.000000e-01 4.770000e+00   
## range sum median mean SE.mean   
## 4.170000e+00 1.157980e+03 2.850000e+00 2.845160e+00 3.775280e-02   
## CI.mean.0.95 var std.dev coef.var   
## 7.421538e-02 5.800866e-01 7.616342e-01 2.676947e-01

stat.desc(concs$freq\_CELEX\_lem)

## nbr.val nbr.null nbr.na min max   
## 399.00000000 11.00000000 12.00000000 0.00000000 3.18800000   
## range sum median mean SE.mean   
## 3.18800000 612.79100000 1.55600000 1.53581704 0.03185900   
## CI.mean.0.95 var std.dev coef.var   
## 0.06263295 0.40498325 0.63638294 0.41436117

stat.desc(concs$AoA\_Brysbaertetal2014)

## nbr.val nbr.null nbr.na min max   
## 398.0000000 0.0000000 13.0000000 3.4100000 13.1800000   
## range sum median mean SE.mean   
## 9.7700000 3210.3200000 8.1500000 8.0661307 0.1086208   
## CI.mean.0.95 var std.dev coef.var   
## 0.2135439 4.6957941 2.1669781 0.2686515

stat.desc(concs$concrete\_Brysbaertetal2014)

## nbr.val nbr.null nbr.na min max   
## 3.980000e+02 0.000000e+00 1.300000e+01 1.200000e+00 5.000000e+00   
## range sum median mean SE.mean   
## 3.800000e+00 1.203330e+03 2.800000e+00 3.023442e+00 5.356266e-02   
## CI.mean.0.95 var std.dev coef.var   
## 1.053019e-01 1.141845e+00 1.068572e+00 3.534288e-01

# See and print correlation of all lexical variables:  
  
mat\_lexicals\_concs <- as.matrix(concs[c('letters', 'phonemes\_DUTCHPOND',   
'orth\_neighbours\_DUTCHPOND', 'phon\_neighbours\_DUTCHPOND', 'freq\_lg10CD\_SUBTLEXNL',   
'freq\_lg10WF\_SUBTLEXNL', 'freq\_CELEX\_lem', 'AoA\_Brysbaertetal2014',   
'concrete\_Brysbaertetal2014')])  
  
rcor.test(mat\_lexicals\_concs, use='complete.obs')

##   
## letters phonemes\_DUTCHPOND  
## letters \*\*\*\*\* 0.942   
## phonemes\_DUTCHPOND <0.001 \*\*\*\*\*   
## orth\_neighbours\_DUTCHPOND <0.001 <0.001   
## phon\_neighbours\_DUTCHPOND <0.001 <0.001   
## freq\_lg10CD\_SUBTLEXNL <0.001 <0.001   
## freq\_lg10WF\_SUBTLEXNL <0.001 <0.001   
## freq\_CELEX\_lem <0.001 <0.001   
## AoA\_Brysbaertetal2014 <0.001 <0.001   
## concrete\_Brysbaertetal2014 <0.001 <0.001   
## orth\_neighbours\_DUTCHPOND  
## letters -0.647   
## phonemes\_DUTCHPOND -0.617   
## orth\_neighbours\_DUTCHPOND \*\*\*\*\*   
## phon\_neighbours\_DUTCHPOND <0.001   
## freq\_lg10CD\_SUBTLEXNL <0.001   
## freq\_lg10WF\_SUBTLEXNL <0.001   
## freq\_CELEX\_lem <0.001   
## AoA\_Brysbaertetal2014 <0.001   
## concrete\_Brysbaertetal2014 <0.001   
## phon\_neighbours\_DUTCHPOND freq\_lg10CD\_SUBTLEXNL  
## letters -0.630 -0.364   
## phonemes\_DUTCHPOND -0.633 -0.362   
## orth\_neighbours\_DUTCHPOND 0.879 0.329   
## phon\_neighbours\_DUTCHPOND \*\*\*\*\* 0.349   
## freq\_lg10CD\_SUBTLEXNL <0.001 \*\*\*\*\*   
## freq\_lg10WF\_SUBTLEXNL <0.001 <0.001   
## freq\_CELEX\_lem <0.001 <0.001   
## AoA\_Brysbaertetal2014 <0.001 <0.001   
## concrete\_Brysbaertetal2014 <0.001 0.378   
## freq\_lg10WF\_SUBTLEXNL freq\_CELEX\_lem  
## letters -0.381 -0.212   
## phonemes\_DUTCHPOND -0.369 -0.237   
## orth\_neighbours\_DUTCHPOND 0.338 0.201   
## phon\_neighbours\_DUTCHPOND 0.352 0.220   
## freq\_lg10CD\_SUBTLEXNL 0.987 0.776   
## freq\_lg10WF\_SUBTLEXNL \*\*\*\*\* 0.757   
## freq\_CELEX\_lem <0.001 \*\*\*\*\*   
## AoA\_Brysbaertetal2014 <0.001 <0.001   
## concrete\_Brysbaertetal2014 0.157 0.028   
## AoA\_Brysbaertetal2014  
## letters 0.491   
## phonemes\_DUTCHPOND 0.513   
## orth\_neighbours\_DUTCHPOND -0.467   
## phon\_neighbours\_DUTCHPOND -0.437   
## freq\_lg10CD\_SUBTLEXNL -0.585   
## freq\_lg10WF\_SUBTLEXNL -0.601   
## freq\_CELEX\_lem -0.430   
## AoA\_Brysbaertetal2014 \*\*\*\*\*   
## concrete\_Brysbaertetal2014 <0.001   
## concrete\_Brysbaertetal2014  
## letters -0.415   
## phonemes\_DUTCHPOND -0.397   
## orth\_neighbours\_DUTCHPOND 0.391   
## phon\_neighbours\_DUTCHPOND 0.348   
## freq\_lg10CD\_SUBTLEXNL 0.007   
## freq\_lg10WF\_SUBTLEXNL 0.039   
## freq\_CELEX\_lem -0.124   
## AoA\_Brysbaertetal2014 -0.569   
## concrete\_Brysbaertetal2014 \*\*\*\*\*   
##   
## upper diagonal part contains correlation coefficient estimates   
## lower diagonal part contains corresponding p-values

corrs\_concs = rcor.test(mat\_lexicals\_concs, use='complete.obs')  
write.csv(corrs\_concs$cor.mat, file = "corrs\_concs.csv",na="") # find table in folder  
  
  
# go on to PCA. This does not include age of acquisition or concreteness for a   
# better comparison with the English data, and because no correlations > .7 (i.e. half  
# of variance explained)  
  
lexicals\_concs <- concs[c('letters', 'phonemes\_DUTCHPOND', 'orth\_neighbours\_DUTCHPOND',   
'phon\_neighbours\_DUTCHPOND', 'freq\_lg10CD\_SUBTLEXNL', 'freq\_lg10WF\_SUBTLEXNL',   
'freq\_CELEX\_lem')]  
  
nrow(lexicals\_concs)

## [1] 411

# start with PCA for lexical variables, done as in Lynott and Connell (2013)  
# Check conditions for a PCA  
# Correlations  
  
cor(lexicals\_concs, use = 'complete.obs')

## letters phonemes\_DUTCHPOND  
## letters 1.0000000 0.9458316  
## phonemes\_DUTCHPOND 0.9458316 1.0000000  
## orth\_neighbours\_DUTCHPOND -0.6300830 -0.6067762  
## phon\_neighbours\_DUTCHPOND -0.6130258 -0.6214952  
## freq\_lg10CD\_SUBTLEXNL -0.3777808 -0.3760751  
## freq\_lg10WF\_SUBTLEXNL -0.3915313 -0.3811413  
## freq\_CELEX\_lem -0.2145080 -0.2359925  
## orth\_neighbours\_DUTCHPOND  
## letters -0.6300830  
## phonemes\_DUTCHPOND -0.6067762  
## orth\_neighbours\_DUTCHPOND 1.0000000  
## phon\_neighbours\_DUTCHPOND 0.8793924  
## freq\_lg10CD\_SUBTLEXNL 0.3333547  
## freq\_lg10WF\_SUBTLEXNL 0.3431698  
## freq\_CELEX\_lem 0.2009497  
## phon\_neighbours\_DUTCHPOND freq\_lg10CD\_SUBTLEXNL  
## letters -0.6130258 -0.3777808  
## phonemes\_DUTCHPOND -0.6214952 -0.3760751  
## orth\_neighbours\_DUTCHPOND 0.8793924 0.3333547  
## phon\_neighbours\_DUTCHPOND 1.0000000 0.3535024  
## freq\_lg10CD\_SUBTLEXNL 0.3535024 1.0000000  
## freq\_lg10WF\_SUBTLEXNL 0.3561152 0.9874908  
## freq\_CELEX\_lem 0.2178926 0.7693247  
## freq\_lg10WF\_SUBTLEXNL freq\_CELEX\_lem  
## letters -0.3915313 -0.2145080  
## phonemes\_DUTCHPOND -0.3811413 -0.2359925  
## orth\_neighbours\_DUTCHPOND 0.3431698 0.2009497  
## phon\_neighbours\_DUTCHPOND 0.3561152 0.2178926  
## freq\_lg10CD\_SUBTLEXNL 0.9874908 0.7693247  
## freq\_lg10WF\_SUBTLEXNL 1.0000000 0.7508591  
## freq\_CELEX\_lem 0.7508591 1.0000000

# Result: all variables fit for PCA, as they have few scores below .3   
# The correlations broadly replicate Lynott and Connell.   
  
# now on the raw vars:  
cortest.bartlett(lexicals\_concs)

## R was not square, finding R from data

## $chisq  
## [1] 3798.943  
##   
## $p.value  
## [1] 0  
##   
## $df  
## [1] 21

# GOOD: Bartlett's test significant   
  
# KMO: Kaiser-Meyer-Olkin Measure of Sampling Adequacy  
lexicals\_concs\_matrix <- cor(lexicals\_concs, use = 'complete.obs')  
KMO(lexicals\_concs\_matrix)

## Kaiser-Meyer-Olkin factor adequacy  
## Call: KMO(r = lexicals\_concs\_matrix)  
## Overall MSA = 0.71  
## MSA for each item =   
## letters phonemes\_DUTCHPOND   
## 0.68 0.68   
## orth\_neighbours\_DUTCHPOND phon\_neighbours\_DUTCHPOND   
## 0.71 0.72   
## freq\_lg10CD\_SUBTLEXNL freq\_lg10WF\_SUBTLEXNL   
## 0.67 0.68   
## freq\_CELEX\_lem   
## 0.94

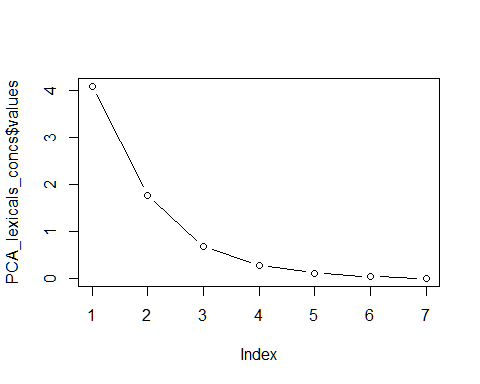
# Result: .71 = good.  
  
# determinant  
det(lexicals\_concs\_matrix)

## [1] 0.000105064

# GOOD: above 0.00001  
  
# start off with unrotated PCA  
  
PCA\_lexicals\_concs <- psych::principal(lexicals\_concs, nfactors = 7, scores = TRUE)  
PCA\_lexicals\_concs

## Principal Components Analysis  
## Call: psych::principal(r = lexicals\_concs, nfactors = 7, scores = TRUE)  
## Standardized loadings (pattern matrix) based upon correlation matrix  
## RC2 RC1 RC3 RC4 RC5 RC6 RC7 h2  
## letters -0.19 0.91 -0.32 -0.04 0.03 0.16 0.00 1  
## phonemes\_DUTCHPOND -0.16 0.92 -0.31 -0.05 -0.03 -0.16 0.00 1  
## orth\_neighbours\_DUTCHPOND 0.15 -0.33 0.90 0.04 -0.24 -0.01 0.00 1  
## phon\_neighbours\_DUTCHPOND 0.17 -0.33 0.90 0.05 0.24 0.01 0.00 1  
## freq\_lg10CD\_SUBTLEXNL 0.96 -0.17 0.15 0.17 0.01 0.01 0.07 1  
## freq\_lg10WF\_SUBTLEXNL 0.96 -0.18 0.16 0.15 0.00 -0.01 -0.07 1  
## freq\_CELEX\_lem 0.65 -0.07 0.07 0.75 0.00 0.00 0.00 1  
## u2 com  
## letters 4.4e-16 1.4  
## phonemes\_DUTCHPOND 8.9e-16 1.4  
## orth\_neighbours\_DUTCHPOND 1.2e-15 1.5  
## phon\_neighbours\_DUTCHPOND 2.0e-15 1.5  
## freq\_lg10CD\_SUBTLEXNL 1.8e-15 1.2  
## freq\_lg10WF\_SUBTLEXNL 1.6e-15 1.2  
## freq\_CELEX\_lem 2.0e-15 2.0  
##   
## RC2 RC1 RC3 RC4 RC5 RC6 RC7  
## SS loadings 2.37 1.96 1.86 0.62 0.12 0.05 0.01  
## Proportion Var 0.34 0.28 0.27 0.09 0.02 0.01 0.00  
## Cumulative Var 0.34 0.62 0.89 0.97 0.99 1.00 1.00  
## Proportion Explained 0.34 0.28 0.27 0.09 0.02 0.01 0.00  
## Cumulative Proportion 0.34 0.62 0.89 0.97 0.99 1.00 1.00  
##   
## Mean item complexity = 1.5  
## Test of the hypothesis that 7 components are sufficient.  
##   
## The root mean square of the residuals (RMSR) is 0   
## with the empirical chi square 0 with prob < NA   
##   
## Fit based upon off diagonal values = 1

# by Kaiser's and Joliffe's standard, extract 3 RCs  
  
# scree analysis  
plot(PCA\_lexicals\_concs$values, type = "b")



# result: again, extract 3 components  
  
PCA\_lexicals\_concs <- psych::principal(lexicals\_concs, nfactors = 3, rotate =   
"varimax", scores = TRUE)  
  
PCA\_lexicals\_concs #-> check explained variance along components

## Principal Components Analysis  
## Call: psych::principal(r = lexicals\_concs, nfactors = 3, rotate = "varimax",   
## scores = TRUE)  
## Standardized loadings (pattern matrix) based upon correlation matrix  
## RC2 RC1 RC3 h2 u2 com  
## letters -0.19 0.91 -0.33 0.97 0.027 1.4  
## phonemes\_DUTCHPOND -0.16 0.92 -0.33 0.97 0.027 1.3  
## orth\_neighbours\_DUTCHPOND 0.15 -0.32 0.90 0.94 0.058 1.3  
## phon\_neighbours\_DUTCHPOND 0.17 -0.32 0.90 0.94 0.059 1.3  
## freq\_lg10CD\_SUBTLEXNL 0.95 -0.18 0.16 0.95 0.048 1.1  
## freq\_lg10WF\_SUBTLEXNL 0.94 -0.19 0.17 0.94 0.056 1.2  
## freq\_CELEX\_lem 0.89 -0.05 0.07 0.81 0.193 1.0  
##   
## RC2 RC1 RC3  
## SS loadings 2.68 1.94 1.90  
## Proportion Var 0.38 0.28 0.27  
## Cumulative Var 0.38 0.66 0.93  
## Proportion Explained 0.41 0.30 0.29  
## Cumulative Proportion 0.41 0.71 1.00  
##   
## Mean item complexity = 1.2  
## Test of the hypothesis that 3 components are sufficient.  
##   
## The root mean square of the residuals (RMSR) is 0.03   
## with the empirical chi square 20.18 with prob < 0.00016   
##   
## Fit based upon off diagonal values = 1

PCA\_lexicals\_concs$loadings

##   
## Loadings:  
## RC2 RC1 RC3   
## letters -0.185 0.910 -0.334  
## phonemes\_DUTCHPOND -0.163 0.917 -0.326  
## orth\_neighbours\_DUTCHPOND 0.148 -0.322 0.903  
## phon\_neighbours\_DUTCHPOND 0.169 -0.317 0.901  
## freq\_lg10CD\_SUBTLEXNL 0.945 -0.181 0.161  
## freq\_lg10WF\_SUBTLEXNL 0.937 -0.192 0.168  
## freq\_CELEX\_lem 0.894   
##   
## RC2 RC1 RC3  
## SS loadings 2.683 1.945 1.905  
## Proportion Var 0.383 0.278 0.272  
## Cumulative Var 0.383 0.661 0.933

# The PCA replicates Lynott and Connell. Standdized correlation coefficients  
# between each PC and its corresponding set of variables are all above .89,  
# while the rest of coefficients are all below .33.   
  
PCA\_lexicals\_concs

## Principal Components Analysis  
## Call: psych::principal(r = lexicals\_concs, nfactors = 3, rotate = "varimax",   
## scores = TRUE)  
## Standardized loadings (pattern matrix) based upon correlation matrix  
## RC2 RC1 RC3 h2 u2 com  
## letters -0.19 0.91 -0.33 0.97 0.027 1.4  
## phonemes\_DUTCHPOND -0.16 0.92 -0.33 0.97 0.027 1.3  
## orth\_neighbours\_DUTCHPOND 0.15 -0.32 0.90 0.94 0.058 1.3  
## phon\_neighbours\_DUTCHPOND 0.17 -0.32 0.90 0.94 0.059 1.3  
## freq\_lg10CD\_SUBTLEXNL 0.95 -0.18 0.16 0.95 0.048 1.1  
## freq\_lg10WF\_SUBTLEXNL 0.94 -0.19 0.17 0.94 0.056 1.2  
## freq\_CELEX\_lem 0.89 -0.05 0.07 0.81 0.193 1.0  
##   
## RC2 RC1 RC3  
## SS loadings 2.68 1.94 1.90  
## Proportion Var 0.38 0.28 0.27  
## Cumulative Var 0.38 0.66 0.93  
## Proportion Explained 0.41 0.30 0.29  
## Cumulative Proportion 0.41 0.71 1.00  
##   
## Mean item complexity = 1.2  
## Test of the hypothesis that 3 components are sufficient.  
##   
## The root mean square of the residuals (RMSR) is 0.03   
## with the empirical chi square 20.18 with prob < 0.00016   
##   
## Fit based upon off diagonal values = 1

# RC1 = length // RC2 = frequency // RC3 = distinctiveness  
  
PCA\_lexicals\_concs$residual

## letters phonemes\_DUTCHPOND  
## letters 0.026594010 -0.025634342  
## phonemes\_DUTCHPOND -0.025634342 0.027463542  
## orth\_neighbours\_DUTCHPOND -0.007939846 0.007595230  
## phon\_neighbours\_DUTCHPOND 0.007967311 -0.008984482  
## freq\_lg10CD\_SUBTLEXNL 0.004139691 0.006696974  
## freq\_lg10WF\_SUBTLEXNL 0.001306927 0.010591926  
## freq\_CELEX\_lem -0.005090237 -0.018061566  
## orth\_neighbours\_DUTCHPOND  
## letters -0.007939846  
## phonemes\_DUTCHPOND 0.007595230  
## orth\_neighbours\_DUTCHPOND 0.058173584  
## phon\_neighbours\_DUTCHPOND -0.058160242  
## freq\_lg10CD\_SUBTLEXNL -0.003609941  
## freq\_lg10WF\_SUBTLEXNL -0.000955949  
## freq\_CELEX\_lem 0.005890390  
## phon\_neighbours\_DUTCHPOND freq\_lg10CD\_SUBTLEXNL  
## letters 0.007967311 0.004139691  
## phonemes\_DUTCHPOND -0.008984482 0.006696974  
## orth\_neighbours\_DUTCHPOND -0.058160242 -0.003609941  
## phon\_neighbours\_DUTCHPOND 0.058821233 -0.001681398  
## freq\_lg10CD\_SUBTLEXNL -0.001681398 0.048039445  
## freq\_lg10WF\_SUBTLEXNL -0.004899638 0.040904630  
## freq\_CELEX\_lem 0.005469160 -0.090683850  
## freq\_lg10WF\_SUBTLEXNL freq\_CELEX\_lem  
## letters 0.001306927 -0.005090237  
## phonemes\_DUTCHPOND 0.010591926 -0.018061566  
## orth\_neighbours\_DUTCHPOND -0.000955949 0.005890390  
## phon\_neighbours\_DUTCHPOND -0.004899638 0.005469160  
## freq\_lg10CD\_SUBTLEXNL 0.040904630 -0.090683850  
## freq\_lg10WF\_SUBTLEXNL 0.055871031 -0.098544553  
## freq\_CELEX\_lem -0.098544553 0.192845909

PCA\_lexicals\_concs$fit

## [1] 0.9950821

PCA\_lexicals\_concs$communality

## letters phonemes\_DUTCHPOND   
## 0.9734060 0.9725365   
## orth\_neighbours\_DUTCHPOND phon\_neighbours\_DUTCHPOND   
## 0.9418264 0.9411788   
## freq\_lg10CD\_SUBTLEXNL freq\_lg10WF\_SUBTLEXNL   
## 0.9519606 0.9441290   
## freq\_CELEX\_lem   
## 0.8071541

# Results based on a Kaiser-normalizalized orthogonal (varimax) rotation  
# (by default in psych::stats pack). Residuals good: less than half w/ absolute   
# values > 0.05. Model fit good, > .90. Communalities (h2) good, all well > .7  
  
concs <- cbind(concs, PCA\_lexicals\_concs$scores)  
  
  
# REGRESSION  
  
# standardize (mean-center and scale)  
concs$s\_Auditory <- scale(concs$Auditory)  
concs$s\_Haptic <- scale(concs$Haptic)  
concs$s\_Visual <- scale(concs$Visual)  
concs$s\_freq\_lg10CD\_SUBTLEXNL <- scale(concs$freq\_lg10CD\_SUBTLEXNL)  
concs$s\_freq\_lg10WF\_SUBTLEXNL <- scale(concs$freq\_lg10WF\_SUBTLEXNL)  
concs$s\_freq\_CELEX\_lem <- scale(concs$freq\_CELEX\_lem)  
concs$s\_AoA\_Brysbaertetal2014 <- scale(concs$AoA\_Brysbaertetal2014)  
concs$s\_concrete\_Brysbaertetal2014 <- scale(concs$concrete\_Brysbaertetal2014)  
concs$s\_letters <- scale(concs$letters)  
concs$s\_phonemes\_DUTCHPOND <- scale(concs$phonemes\_DUTCHPOND)  
concs$s\_orth\_neighbours\_DUTCHPOND <- scale(concs$orth\_neighbours\_DUTCHPOND)  
concs$s\_phon\_neighbours\_DUTCHPOND <- scale(concs$phon\_neighbours\_DUTCHPOND)  
concs$s\_RC1\_lexicals <- scale(concs$RC1)  
concs$s\_RC2\_lexicals <- scale(concs$RC2)   
concs$s\_RC3\_lexicals <- scale(concs$RC3)  
  
# length: letters  
fit\_letters\_concs <- lm(concs$s\_letters ~ concs$s\_Auditory + concs$s\_Haptic +   
concs$s\_Visual, data = concs)  
stat.desc(fit\_letters\_concs$residuals, norm = TRUE)

## x  
## nbr.val 4.110000e+02  
## nbr.null 0.000000e+00  
## nbr.na 0.000000e+00  
## min -1.869411e+00  
## max 4.100072e+00  
## range 5.969482e+00  
## sum -2.588207e-15  
## median -1.709015e-01  
## mean -6.333218e-18  
## SE.mean 4.751817e-02  
## CI.mean.0.95 9.340965e-02  
## var 9.280284e-01  
## std.dev 9.633423e-01  
## coef.var -1.521095e+17  
## skewness 7.313500e-01  
## skew.2SE 3.037507e+00  
## kurtosis 3.972703e-01  
## kurt.2SE 8.269597e-01  
## normtest.W 9.582384e-01  
## normtest.p 2.101496e-09

# residuals distribution: skew. Raw scores/2.SE > 1  
# have to log-transform DV and re-run regression  
  
psych::describe(concs$s\_letters)

## vars n mean sd median trimmed mad min max range skew kurtosis  
## X1 1 411 0 1 -0.28 -0.09 1.17 -1.46 4.05 5.51 0.74 0.28  
## se  
## X1 0.05

concs$log\_s\_letters <- log(3 + concs$s\_letters)  
  
fit\_letters\_concs <- lm(concs$log\_s\_letters ~ concs$s\_Auditory + concs$s\_Haptic +   
concs$s\_Visual, data = concs)  
  
# check residuals again  
stat.desc(fit\_letters\_concs$residuals, norm = TRUE)

## x  
## nbr.val 4.110000e+02  
## nbr.null 0.000000e+00  
## nbr.na 0.000000e+00  
## min -7.546621e-01  
## max 9.308451e-01  
## range 1.685507e+00  
## sum 9.228729e-16  
## median -1.147895e-02  
## mean 2.265877e-18  
## SE.mean 1.549036e-02  
## CI.mean.0.95 3.045044e-02  
## var 9.862002e-02  
## std.dev 3.140382e-01  
## coef.var 1.385946e+17  
## skewness 9.384492e-02  
## skew.2SE 3.897649e-01  
## kurtosis -6.833286e-01  
## kurt.2SE -1.422420e+00  
## normtest.W 9.883934e-01  
## normtest.p 2.360257e-03

# better though still skew/kurtose  
  
# Check multicollinearity: largest VIF (pref. < 10), mean VIF (pref. around 1),   
# and tolerance (pref. > 0.2)  
vif(fit\_letters\_concs)

## concs$s\_Auditory concs$s\_Haptic concs$s\_Visual   
## 1.009925 1.245338 1.254282

mean(vif(fit\_letters\_concs))

## [1] 1.169848

1/vif(fit\_letters\_concs)

## concs$s\_Auditory concs$s\_Haptic concs$s\_Visual   
## 0.9901729 0.8029946 0.7972690

# RESULTS: all good  
  
step\_letters\_concs\_AIC <- stepAIC(fit\_letters\_concs, direction="both")

## Start: AIC=-945.07  
## concs$log\_s\_letters ~ concs$s\_Auditory + concs$s\_Haptic + concs$s\_Visual  
##   
## Df Sum of Sq RSS AIC  
## - concs$s\_Visual 1 0.19201 40.626 -945.13  
## <none> 40.434 -945.07  
## - concs$s\_Auditory 1 0.99968 41.434 -937.04  
## - concs$s\_Haptic 1 1.51007 41.944 -932.01  
##   
## Step: AIC=-945.13  
## concs$log\_s\_letters ~ concs$s\_Auditory + concs$s\_Haptic  
##   
## Df Sum of Sq RSS AIC  
## <none> 40.626 -945.13  
## + concs$s\_Visual 1 0.19201 40.434 -945.07  
## - concs$s\_Auditory 1 0.92402 41.550 -937.88  
## - concs$s\_Haptic 1 2.52263 43.149 -922.37

step\_letters\_concs\_F <- stepAIC(fit\_letters\_concs, direction="both", test="F")

## Start: AIC=-945.07  
## concs$log\_s\_letters ~ concs$s\_Auditory + concs$s\_Haptic + concs$s\_Visual  
##   
## Df Sum of Sq RSS AIC F Value Pr(F)   
## - concs$s\_Visual 1 0.19201 40.626 -945.13 1.9327 0.1652242   
## <none> 40.434 -945.07   
## - concs$s\_Auditory 1 0.99968 41.434 -937.04 10.0625 0.0016279 \*\*   
## - concs$s\_Haptic 1 1.51007 41.944 -932.01 15.1999 0.0001131 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Step: AIC=-945.13  
## concs$log\_s\_letters ~ concs$s\_Auditory + concs$s\_Haptic  
##   
## Df Sum of Sq RSS AIC F Value Pr(F)   
## <none> 40.626 -945.13   
## + concs$s\_Visual 1 0.19201 40.434 -945.07 1.9327 0.165224   
## - concs$s\_Auditory 1 0.92402 41.550 -937.88 9.2797 0.002468 \*\*   
## - concs$s\_Haptic 1 2.52263 43.149 -922.37 25.3342 7.245e-07 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

summary(fit\_letters\_concs)

##   
## Call:  
## lm(formula = concs$log\_s\_letters ~ concs$s\_Auditory + concs$s\_Haptic +   
## concs$s\_Visual, data = concs)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.75466 -0.24161 -0.01148 0.23692 0.93085   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 1.04489 0.01555 67.207 < 2e-16 \*\*\*  
## concs$s\_Auditory 0.04962 0.01564 3.172 0.001628 \*\*   
## concs$s\_Haptic -0.06773 0.01737 -3.899 0.000113 \*\*\*  
## concs$s\_Visual -0.02424 0.01743 -1.390 0.165224   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.3152 on 407 degrees of freedom  
## Multiple R-squared: 0.08311, Adjusted R-squared: 0.07635   
## F-statistic: 12.3 on 3 and 407 DF, p-value: 1.025e-07

# length: phonemes\_DUTCHPOND  
fit\_phonemes\_DUTCHPOND\_concs <- lm(concs$s\_phonemes\_DUTCHPOND ~ concs$s\_Auditory +   
concs$s\_Haptic + concs$s\_Visual, data = concs)  
stat.desc(fit\_phonemes\_DUTCHPOND\_concs$residuals, norm = TRUE)

## x  
## nbr.val 3.890000e+02  
## nbr.null 0.000000e+00  
## nbr.na 0.000000e+00  
## min -2.130401e+00  
## max 4.299017e+00  
## range 6.429418e+00  
## sum 5.016820e-15  
## median -1.585982e-01  
## mean 1.286522e-17  
## SE.mean 4.897103e-02  
## CI.mean.0.95 9.628179e-02  
## var 9.328850e-01  
## std.dev 9.658597e-01  
## coef.var 7.507528e+16  
## skewness 6.756933e-01  
## skew.2SE 2.730764e+00  
## kurtosis 6.167434e-01  
## kurt.2SE 1.249403e+00  
## normtest.W 9.707429e-01  
## normtest.p 4.832522e-07

# residuals distribution: skew and kurtose. Raw scores/2.SE > 1  
# have to log-transform DV and re-run regression  
  
psych::describe(concs$s\_phonemes\_DUTCHPOND)

## vars n mean sd median trimmed mad min max range skew kurtosis  
## X1 1 389 0 1 0.08 -0.07 1.36 -1.76 4.21 5.97 0.7 0.41  
## se  
## X1 0.05

concs$log\_s\_phonemes\_DUTCHPOND <- log(3 + concs$s\_phonemes\_DUTCHPOND)  
  
fit\_phonemes\_DUTCHPOND\_concs <- lm(concs$log\_s\_phonemes\_DUTCHPOND ~ concs$s\_Auditory  
 + concs$s\_Haptic + concs$s\_Visual, data = concs)  
  
# check residuals again  
stat.desc(fit\_phonemes\_DUTCHPOND\_concs$residuals, norm = TRUE)

## x  
## nbr.val 3.890000e+02  
## nbr.null 0.000000e+00  
## nbr.na 0.000000e+00  
## min -9.530293e-01  
## max 9.618234e-01  
## range 1.914853e+00  
## sum -4.623038e-15  
## median -2.033824e-03  
## mean -1.187383e-17  
## SE.mean 1.628015e-02  
## CI.mean.0.95 3.200835e-02  
## var 1.031018e-01  
## std.dev 3.210947e-01  
## coef.var -2.704223e+16  
## skewness -1.089115e-01  
## skew.2SE -4.401577e-01  
## kurtosis -2.710183e-01  
## kurt.2SE -5.490309e-01  
## normtest.W 9.942364e-01  
## normtest.p 1.496887e-01

# good  
  
# Check multicollinearity: largest VIF (pref. < 10), mean VIF (pref. around 1), and   
# tolerance (pref. > 0.2)  
vif(fit\_phonemes\_DUTCHPOND\_concs)

## concs$s\_Auditory concs$s\_Haptic concs$s\_Visual   
## 1.005252 1.250064 1.252494

mean(vif(fit\_phonemes\_DUTCHPOND\_concs))

## [1] 1.16927

1/vif(fit\_phonemes\_DUTCHPOND\_concs)

## concs$s\_Auditory concs$s\_Haptic concs$s\_Visual   
## 0.9947753 0.7999588 0.7984071

# RESULTS: all good  
  
step\_phonemes\_DUTCHPOND\_concs\_AIC <- stepAIC(fit\_phonemes\_DUTCHPOND\_concs,   
direction="both")

## Start: AIC=-876.82  
## concs$log\_s\_phonemes\_DUTCHPOND ~ concs$s\_Auditory + concs$s\_Haptic +   
## concs$s\_Visual  
##   
## Df Sum of Sq RSS AIC  
## - concs$s\_Visual 1 0.07544 40.079 -878.09  
## <none> 40.004 -876.82  
## - concs$s\_Haptic 1 1.11568 41.119 -868.12  
## - concs$s\_Auditory 1 1.20234 41.206 -867.30  
##   
## Step: AIC=-878.09  
## concs$log\_s\_phonemes\_DUTCHPOND ~ concs$s\_Auditory + concs$s\_Haptic  
##   
## Df Sum of Sq RSS AIC  
## <none> 40.079 -878.09  
## + concs$s\_Visual 1 0.07544 40.004 -876.82  
## - concs$s\_Auditory 1 1.16707 41.246 -868.93  
## - concs$s\_Haptic 1 1.73650 41.815 -863.59

step\_phonemes\_DUTCHPOND\_concs\_F <- stepAIC(fit\_phonemes\_DUTCHPOND\_concs,   
direction="both", test="F")

## Start: AIC=-876.82  
## concs$log\_s\_phonemes\_DUTCHPOND ~ concs$s\_Auditory + concs$s\_Haptic +   
## concs$s\_Visual  
##   
## Df Sum of Sq RSS AIC F Value Pr(F)   
## - concs$s\_Visual 1 0.07544 40.079 -878.09 0.7261 0.3946957   
## <none> 40.004 -876.82   
## - concs$s\_Haptic 1 1.11568 41.119 -868.12 10.7374 0.0011452 \*\*   
## - concs$s\_Auditory 1 1.20234 41.206 -867.30 11.5715 0.0007399 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Step: AIC=-878.09  
## concs$log\_s\_phonemes\_DUTCHPOND ~ concs$s\_Auditory + concs$s\_Haptic  
##   
## Df Sum of Sq RSS AIC F Value Pr(F)   
## <none> 40.079 -878.09   
## + concs$s\_Visual 1 0.07544 40.004 -876.82 0.7261 0.3946957   
## - concs$s\_Auditory 1 1.16707 41.246 -868.93 11.2401 0.0008796 \*\*\*  
## - concs$s\_Haptic 1 1.73650 41.815 -863.59 16.7242 5.263e-05 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

summary(fit\_phonemes\_DUTCHPOND\_concs)

##   
## Call:  
## lm(formula = concs$log\_s\_phonemes\_DUTCHPOND ~ concs$s\_Auditory +   
## concs$s\_Haptic + concs$s\_Visual, data = concs)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.95303 -0.22895 -0.00203 0.23763 0.96182   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 1.04396 0.01636 63.821 < 2e-16 \*\*\*  
## concs$s\_Auditory 0.05577 0.01640 3.402 0.00074 \*\*\*  
## concs$s\_Haptic -0.05941 0.01813 -3.277 0.00115 \*\*   
## concs$s\_Visual -0.01571 0.01844 -0.852 0.39470   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.3223 on 385 degrees of freedom  
## (22 observations deleted due to missingness)  
## Multiple R-squared: 0.07083, Adjusted R-squared: 0.06359   
## F-statistic: 9.783 on 3 and 385 DF, p-value: 3.109e-06

# distinctiveness: orth neigh size  
fit\_orth\_neighbours\_DUTCHPOND\_concs <- lm(concs$s\_orth\_neighbours\_DUTCHPOND ~   
concs$s\_Auditory + concs$s\_Haptic + concs$s\_Visual, data = concs)  
stat.desc(fit\_orth\_neighbours\_DUTCHPOND\_concs$residuals, norm = TRUE)

## x  
## nbr.val 4.110000e+02  
## nbr.null 0.000000e+00  
## nbr.na 0.000000e+00  
## min -1.418943e+00  
## max 4.983802e+00  
## range 6.402745e+00  
## sum 1.929013e-15  
## median -3.388470e-01  
## mean 4.732767e-18  
## SE.mean 4.699016e-02  
## CI.mean.0.95 9.237170e-02  
## var 9.075189e-01  
## std.dev 9.526379e-01  
## coef.var 2.012856e+17  
## skewness 1.758073e+00  
## skew.2SE 7.301783e+00  
## kurtosis 3.397510e+00  
## kurt.2SE 7.072272e+00  
## normtest.W 8.209900e-01  
## normtest.p 4.657007e-21

# residuals distribution: skewed and kurtosed. Raw scores/2.SE > 1  
# have to log-transform DV and re-run regression  
  
psych::describe(concs$s\_orth\_neighbours\_DUTCHPOND)

## vars n mean sd median trimmed mad min max range skew kurtosis se  
## X1 1 411 0 1 -0.53 -0.22 0.26 -0.7 4.88 5.59 1.88 3.42 0.05

concs$log\_s\_orth\_neighbours\_DUTCHPOND <- log(2 + concs$s\_orth\_neighbours\_DUTCHPOND)  
  
fit\_orth\_neighbours\_DUTCHPOND\_concs <- lm(concs$log\_s\_orth\_neighbours\_DUTCHPOND ~   
concs$s\_Auditory + concs$s\_Haptic + concs$s\_Visual, data = concs)  
  
# check residuals again  
stat.desc(fit\_orth\_neighbours\_DUTCHPOND\_concs$residuals, norm = TRUE)

## x  
## nbr.val 4.110000e+02  
## nbr.null 0.000000e+00  
## nbr.na 0.000000e+00  
## min -6.413729e-01  
## max 1.377855e+00  
## range 2.019228e+00  
## sum 2.515349e-15  
## median -1.365831e-01  
## mean 6.152518e-18  
## SE.mean 1.874673e-02  
## CI.mean.0.95 3.685169e-02  
## var 1.444417e-01  
## std.dev 3.800549e-01  
## coef.var 6.177226e+16  
## skewness 1.033319e+00  
## skew.2SE 4.291669e+00  
## kurtosis 3.381673e-01  
## kurt.2SE 7.039305e-01  
## normtest.W 9.016737e-01  
## normtest.p 1.225133e-15

# better though still skew/kurtose  
  
# Check multicollinearity: largest VIF (pref. < 10), mean VIF (pref. around 1), and   
# tolerance (pref. > 0.2)  
vif(fit\_orth\_neighbours\_DUTCHPOND\_concs)

## concs$s\_Auditory concs$s\_Haptic concs$s\_Visual   
## 1.009925 1.245338 1.254282

mean(vif(fit\_orth\_neighbours\_DUTCHPOND\_concs))

## [1] 1.169848

1/vif(fit\_orth\_neighbours\_DUTCHPOND\_concs)

## concs$s\_Auditory concs$s\_Haptic concs$s\_Visual   
## 0.9901729 0.8029946 0.7972690

# RESULTS: all good  
  
step\_orth\_neighbours\_DUTCHPOND\_concs\_AIC <-   
stepAIC(fit\_orth\_neighbours\_DUTCHPOND\_concs, direction="both")

## Start: AIC=-788.24  
## concs$log\_s\_orth\_neighbours\_DUTCHPOND ~ concs$s\_Auditory + concs$s\_Haptic +   
## concs$s\_Visual  
##   
## Df Sum of Sq RSS AIC  
## - concs$s\_Visual 1 0.0959 59.317 -789.57  
## <none> 59.221 -788.24  
## - concs$s\_Auditory 1 0.9578 60.179 -783.64  
## - concs$s\_Haptic 1 3.8929 63.114 -764.07  
##   
## Step: AIC=-789.57  
## concs$log\_s\_orth\_neighbours\_DUTCHPOND ~ concs$s\_Auditory + concs$s\_Haptic  
##   
## Df Sum of Sq RSS AIC  
## <none> 59.317 -789.57  
## + concs$s\_Visual 1 0.0959 59.221 -788.24  
## - concs$s\_Auditory 1 0.9077 60.225 -785.33  
## - concs$s\_Haptic 1 5.5464 64.863 -754.83

step\_orth\_neighbours\_DUTCHPOND\_concs\_F <-   
stepAIC(fit\_orth\_neighbours\_DUTCHPOND\_concs, direction="both", test="F")

## Start: AIC=-788.24  
## concs$log\_s\_orth\_neighbours\_DUTCHPOND ~ concs$s\_Auditory + concs$s\_Haptic +   
## concs$s\_Visual  
##   
## Df Sum of Sq RSS AIC F Value Pr(F)   
## - concs$s\_Visual 1 0.0959 59.317 -789.57 0.6590 0.41737   
## <none> 59.221 -788.24   
## - concs$s\_Auditory 1 0.9578 60.179 -783.64 6.5826 0.01065 \*   
## - concs$s\_Haptic 1 3.8929 63.114 -764.07 26.7539 3.634e-07 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Step: AIC=-789.57  
## concs$log\_s\_orth\_neighbours\_DUTCHPOND ~ concs$s\_Auditory + concs$s\_Haptic  
##   
## Df Sum of Sq RSS AIC F Value Pr(F)   
## <none> 59.317 -789.57   
## + concs$s\_Visual 1 0.0959 59.221 -788.24 0.659 0.41737   
## - concs$s\_Auditory 1 0.9077 60.225 -785.33 6.244 0.01286 \*   
## - concs$s\_Haptic 1 5.5464 64.863 -754.83 38.150 1.586e-09 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

summary(fit\_orth\_neighbours\_DUTCHPOND\_concs)

##   
## Call:  
## lm(formula = concs$log\_s\_orth\_neighbours\_DUTCHPOND ~ concs$s\_Auditory +   
## concs$s\_Haptic + concs$s\_Visual, data = concs)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.6414 -0.2734 -0.1366 0.1982 1.3779   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.60117 0.01882 31.951 < 2e-16 \*\*\*  
## concs$s\_Auditory -0.04857 0.01893 -2.566 0.0107 \*   
## concs$s\_Haptic 0.10874 0.02102 5.172 3.63e-07 \*\*\*  
## concs$s\_Visual 0.01713 0.02110 0.812 0.4174   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.3815 on 407 degrees of freedom  
## Multiple R-squared: 0.1001, Adjusted R-squared: 0.09348   
## F-statistic: 15.09 on 3 and 407 DF, p-value: 2.477e-09

# distinctiveness: phon neigh size  
fit\_phon\_neighbours\_DUTCHPOND\_concs <- lm(concs$s\_phon\_neighbours\_DUTCHPOND ~   
concs$s\_Auditory + concs$s\_Haptic + concs$s\_Visual, data = concs)  
stat.desc(fit\_phon\_neighbours\_DUTCHPOND\_concs$residuals, norm = TRUE)

## x  
## nbr.val 4.110000e+02  
## nbr.null 0.000000e+00  
## nbr.na 0.000000e+00  
## min -1.352384e+00  
## max 4.807540e+00  
## range 6.159924e+00  
## sum -1.264613e-15  
## median -3.437850e-01  
## mean -3.056870e-18  
## SE.mean 4.739867e-02  
## CI.mean.0.95 9.317474e-02  
## var 9.233666e-01  
## std.dev 9.609197e-01  
## coef.var -3.143476e+17  
## skewness 2.025920e+00  
## skew.2SE 8.414228e+00  
## kurtosis 4.788769e+00  
## kurt.2SE 9.968323e+00  
## normtest.W 7.877461e-01  
## normtest.p 9.278171e-23

# residuals distribution: skewed and kurtosed. Raw scores/2.SE > 1  
# have to log-transform DV and re-run regression  
  
psych::describe(concs$s\_phon\_neighbours\_DUTCHPOND)

## vars n mean sd median trimmed mad min max range skew kurtosis  
## X1 1 411 0 1 -0.55 -0.22 0.18 -0.67 5.26 5.92 2.06 4.49  
## se  
## X1 0.05

concs$log\_s\_phon\_neighbours\_DUTCHPOND <- log(2 + concs$s\_phon\_neighbours\_DUTCHPOND)  
  
fit\_phon\_neighbours\_DUTCHPOND\_concs <- lm(concs$log\_s\_phon\_neighbours\_DUTCHPOND ~   
concs$s\_Auditory + concs$s\_Haptic + concs$s\_Visual, data = concs)  
  
# check residuals again  
stat.desc(fit\_phon\_neighbours\_DUTCHPOND\_concs$residuals, norm = TRUE)

## x  
## nbr.val 4.110000e+02  
## nbr.null 0.000000e+00  
## nbr.na 0.000000e+00  
## min -6.105674e-01  
## max 1.390839e+00  
## range 2.001407e+00  
## sum 2.714842e-15  
## median -1.406069e-01  
## mean 6.635001e-18  
## SE.mean 1.839574e-02  
## CI.mean.0.95 3.616174e-02  
## var 1.390838e-01  
## std.dev 3.729394e-01  
## coef.var 5.620788e+16  
## skewness 1.199464e+00  
## skew.2SE 4.981718e+00  
## kurtosis 9.001905e-01  
## kurt.2SE 1.873840e+00  
## normtest.W 8.836707e-01  
## normtest.p 4.522529e-17

# better but not perfect  
  
# Check multicollinearity: largest VIF (pref. < 10), mean VIF (pref. around 1), and   
# tolerance (pref. > 0.2)  
vif(fit\_phon\_neighbours\_DUTCHPOND\_concs)

## concs$s\_Auditory concs$s\_Haptic concs$s\_Visual   
## 1.009925 1.245338 1.254282

mean(vif(fit\_phon\_neighbours\_DUTCHPOND\_concs))

## [1] 1.169848

1/vif(fit\_phon\_neighbours\_DUTCHPOND\_concs)

## concs$s\_Auditory concs$s\_Haptic concs$s\_Visual   
## 0.9901729 0.8029946 0.7972690

# RESULTS: all good  
  
step\_phon\_neighbours\_DUTCHPOND\_concs\_AIC <-   
stepAIC(fit\_phon\_neighbours\_DUTCHPOND\_concs, direction="both")

## Start: AIC=-803.77  
## concs$log\_s\_phon\_neighbours\_DUTCHPOND ~ concs$s\_Auditory + concs$s\_Haptic +   
## concs$s\_Visual  
##   
## Df Sum of Sq RSS AIC  
## - concs$s\_Visual 1 0.0168 57.041 -805.65  
## <none> 57.024 -803.77  
## - concs$s\_Auditory 1 0.6885 57.713 -800.84  
## - concs$s\_Haptic 1 3.8278 60.852 -779.07  
##   
## Step: AIC=-805.65  
## concs$log\_s\_phon\_neighbours\_DUTCHPOND ~ concs$s\_Auditory + concs$s\_Haptic  
##   
## Df Sum of Sq RSS AIC  
## <none> 57.041 -805.65  
## + concs$s\_Visual 1 0.0168 57.024 -803.77  
## - concs$s\_Auditory 1 0.6740 57.715 -802.82  
## - concs$s\_Haptic 1 5.0509 62.092 -772.78

step\_phon\_neighbours\_DUTCHPOND\_concs\_F <- stepAIC(fit\_phon\_neighbours\_DUTCHPOND\_concs,   
direction="both", test="F")

## Start: AIC=-803.77  
## concs$log\_s\_phon\_neighbours\_DUTCHPOND ~ concs$s\_Auditory + concs$s\_Haptic +   
## concs$s\_Visual  
##   
## Df Sum of Sq RSS AIC F Value Pr(F)   
## - concs$s\_Visual 1 0.0168 57.041 -805.65 0.1199 0.72937   
## <none> 57.024 -803.77   
## - concs$s\_Auditory 1 0.6885 57.713 -800.84 4.9142 0.02719 \*   
## - concs$s\_Haptic 1 3.8278 60.852 -779.07 27.3204 2.761e-07 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Step: AIC=-805.65  
## concs$log\_s\_phon\_neighbours\_DUTCHPOND ~ concs$s\_Auditory + concs$s\_Haptic  
##   
## Df Sum of Sq RSS AIC F Value Pr(F)   
## <none> 57.041 -805.65   
## + concs$s\_Visual 1 0.0168 57.024 -803.77 0.120 0.72937   
## - concs$s\_Auditory 1 0.6740 57.715 -802.82 4.821 0.02868 \*   
## - concs$s\_Haptic 1 5.0509 62.092 -772.78 36.128 4.101e-09 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

summary(fit\_phon\_neighbours\_DUTCHPOND\_concs)

##   
## Call:  
## lm(formula = concs$log\_s\_phon\_neighbours\_DUTCHPOND ~ concs$s\_Auditory +   
## concs$s\_Haptic + concs$s\_Visual, data = concs)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.6106 -0.2524 -0.1406 0.1897 1.3908   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.604084 0.018463 32.718 < 2e-16 \*\*\*  
## concs$s\_Auditory -0.041182 0.018577 -2.217 0.0272 \*   
## concs$s\_Haptic 0.107827 0.020629 5.227 2.76e-07 \*\*\*  
## concs$s\_Visual 0.007167 0.020703 0.346 0.7294   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.3743 on 407 degrees of freedom  
## Multiple R-squared: 0.09194, Adjusted R-squared: 0.08525   
## F-statistic: 13.74 on 3 and 407 DF, p-value: 1.499e-08

# freq: SUBTLEX-NL log-10 CD  
  
fit\_freq\_lg10CD\_SUBTLEXNL\_concs <- lm(concs$s\_freq\_lg10CD\_SUBTLEXNL ~   
concs$s\_Auditory + concs$s\_Haptic + concs$s\_Visual, data = concs)  
stat.desc(fit\_freq\_lg10CD\_SUBTLEXNL\_concs$residuals, norm = TRUE)

## x  
## nbr.val 4.070000e+02  
## nbr.null 0.000000e+00  
## nbr.na 0.000000e+00  
## min -3.048064e+00  
## max 2.307864e+00  
## range 5.355928e+00  
## sum -7.049916e-15  
## median 1.144372e-02  
## mean -1.729615e-17  
## SE.mean 4.797448e-02  
## CI.mean.0.95 9.430939e-02  
## var 9.367312e-01  
## std.dev 9.678487e-01  
## coef.var -5.595745e+16  
## skewness -2.807029e-01  
## skew.2SE -1.160194e+00  
## kurtosis 7.175888e-02  
## kurt.2SE 1.486536e-01  
## normtest.W 9.927886e-01  
## normtest.p 4.745518e-02

# residuals distribution: skew. Raw scores/2.SE > 1  
# have to log-transform DV and re-run regression  
  
psych::describe(concs$s\_freq\_lg10CD\_SUBTLEXNL)

## vars n mean sd median trimmed mad min max range skew kurtosis  
## X1 1 407 0 1 0.07 0.04 1.01 -3.11 1.88 4.99 -0.39 -0.14  
## se  
## X1 0.05

concs$log\_s\_freq\_lg10CD\_SUBTLEXNL <- log(5 + concs$s\_freq\_lg10CD\_SUBTLEXNL)  
  
fit\_freq\_lg10CD\_SUBTLEXNL\_concs <- lm(concs$log\_s\_freq\_lg10CD\_SUBTLEXNL ~   
concs$s\_Auditory + concs$s\_Haptic + concs$s\_Visual, data = concs)  
  
# check residuals again  
stat.desc(fit\_freq\_lg10CD\_SUBTLEXNL\_concs$residuals, norm = TRUE)

## x  
## nbr.val 4.070000e+02  
## nbr.null 0.000000e+00  
## nbr.na 0.000000e+00  
## min -9.360006e-01  
## max 4.459662e-01  
## range 1.381967e+00  
## sum -2.066923e-15  
## median 2.387622e-02  
## mean -5.070726e-18  
## SE.mean 1.060014e-02  
## CI.mean.0.95 2.083802e-02  
## var 4.573176e-02  
## std.dev 2.138498e-01  
## coef.var -4.217342e+16  
## skewness -9.892655e-01  
## skew.2SE -4.088806e+00  
## kurtosis 1.832425e+00  
## kurt.2SE 3.795999e+00  
## normtest.W 9.507730e-01  
## normtest.p 2.130839e-10

# worse! back   
fit\_freq\_lg10CD\_SUBTLEXNL\_concs <- lm(concs$s\_freq\_lg10CD\_SUBTLEXNL ~   
concs$s\_Auditory + concs$s\_Haptic + concs$s\_Visual, data = concs)  
  
# Check multicollinearity: largest VIF (pref. < 10), mean VIF (pref. around 1), and   
# tolerance (pref. > 0.2)  
vif(fit\_freq\_lg10CD\_SUBTLEXNL\_concs)

## concs$s\_Auditory concs$s\_Haptic concs$s\_Visual   
## 1.007087 1.251386 1.256979

mean(vif(fit\_freq\_lg10CD\_SUBTLEXNL\_concs))

## [1] 1.171817

1/vif(fit\_freq\_lg10CD\_SUBTLEXNL\_concs)

## concs$s\_Auditory concs$s\_Haptic concs$s\_Visual   
## 0.9929629 0.7991139 0.7955584

# RESULTS: all good  
  
step\_freq\_lg10CD\_SUBTLEXNL\_concs\_AIC <- stepAIC(fit\_freq\_lg10CD\_SUBTLEXNL\_concs,   
direction="both")

## Start: AIC=-19.6  
## concs$s\_freq\_lg10CD\_SUBTLEXNL ~ concs$s\_Auditory + concs$s\_Haptic +   
## concs$s\_Visual  
##   
## Df Sum of Sq RSS AIC  
## - concs$s\_Haptic 1 1.122 381.43 -20.4034  
## <none> 380.31 -19.6023  
## - concs$s\_Visual 1 3.709 384.02 -17.6523  
## - concs$s\_Auditory 1 16.168 396.48 -4.6579  
##   
## Step: AIC=-20.4  
## concs$s\_freq\_lg10CD\_SUBTLEXNL ~ concs$s\_Auditory + concs$s\_Visual  
##   
## Df Sum of Sq RSS AIC  
## <none> 381.43 -20.4034  
## + concs$s\_Haptic 1 1.1220 380.31 -19.6023  
## - concs$s\_Visual 1 7.2094 388.64 -14.7826  
## - concs$s\_Auditory 1 15.7897 397.22 -5.8947

step\_freq\_lg10CD\_SUBTLEXNL\_\_concsF <- stepAIC(fit\_freq\_lg10CD\_SUBTLEXNL\_concs,   
direction="both", test="F")

## Start: AIC=-19.6  
## concs$s\_freq\_lg10CD\_SUBTLEXNL ~ concs$s\_Auditory + concs$s\_Haptic +   
## concs$s\_Visual  
##   
## Df Sum of Sq RSS AIC F Value Pr(F)   
## - concs$s\_Haptic 1 1.122 381.43 -20.4034 1.1889 0.2762   
## <none> 380.31 -19.6023   
## - concs$s\_Visual 1 3.709 384.02 -17.6523 3.9302 0.0481 \*   
## - concs$s\_Auditory 1 16.168 396.48 -4.6579 17.1321 4.248e-05 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Step: AIC=-20.4  
## concs$s\_freq\_lg10CD\_SUBTLEXNL ~ concs$s\_Auditory + concs$s\_Visual  
##   
## Df Sum of Sq RSS AIC F Value Pr(F)   
## <none> 381.43 -20.4034   
## + concs$s\_Haptic 1 1.1220 380.31 -19.6023 1.1889 0.276198   
## - concs$s\_Visual 1 7.2094 388.64 -14.7826 7.6359 0.005983 \*\*   
## - concs$s\_Auditory 1 15.7897 397.22 -5.8947 16.7238 5.219e-05 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

summary(fit\_freq\_lg10CD\_SUBTLEXNL\_concs)

##   
## Call:  
## lm(formula = concs$s\_freq\_lg10CD\_SUBTLEXNL ~ concs$s\_Auditory +   
## concs$s\_Haptic + concs$s\_Visual, data = concs)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -3.04806 -0.54760 0.01144 0.66048 2.30786   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -0.00433 0.04816 -0.090 0.9284   
## concs$s\_Auditory 0.20136 0.04865 4.139 4.25e-05 \*\*\*  
## concs$s\_Haptic 0.05869 0.05382 1.090 0.2762   
## concs$s\_Visual 0.10775 0.05435 1.982 0.0481 \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.9714 on 403 degrees of freedom  
## (4 observations deleted due to missingness)  
## Multiple R-squared: 0.06327, Adjusted R-squared: 0.0563   
## F-statistic: 9.073 on 3 and 403 DF, p-value: 7.971e-06

# freq: SUBTLEX-NL log-10 WF  
fit\_freq\_lg10WF\_SUBTLEXNL\_concs <-   
lm(concs$s\_freq\_lg10WF\_SUBTLEXNL ~ concs$s\_Auditory + concs$s\_Haptic + concs$s\_Visual,  
 data = concs)  
stat.desc(fit\_freq\_lg10WF\_SUBTLEXNL\_concs$residuals, norm = TRUE)

## x  
## nbr.val 4.070000e+02  
## nbr.null 0.000000e+00  
## nbr.na 0.000000e+00  
## min -2.895265e+00  
## max 2.942859e+00  
## range 5.838124e+00  
## sum 6.078471e-15  
## median 2.548493e-02  
## mean 1.480728e-17  
## SE.mean 4.802025e-02  
## CI.mean.0.95 9.439937e-02  
## var 9.385194e-01  
## std.dev 9.687721e-01  
## coef.var 6.542537e+16  
## skewness 2.645810e-02  
## skew.2SE 1.093559e-01  
## kurtosis 1.608904e-01  
## kurt.2SE 3.332959e-01  
## normtest.W 9.969441e-01  
## normtest.p 6.460055e-01

# residuals distribution: good. Raw scores/2.SE < 1  
  
# Check multicollinearity: largest VIF (pref. < 10), mean VIF (pref. around 1), and   
# tolerance (pref. > 0.2)  
vif(fit\_freq\_lg10WF\_SUBTLEXNL\_concs)

## concs$s\_Auditory concs$s\_Haptic concs$s\_Visual   
## 1.007087 1.251386 1.256979

mean(vif(fit\_freq\_lg10WF\_SUBTLEXNL\_concs))

## [1] 1.171817

1/vif(fit\_freq\_lg10WF\_SUBTLEXNL\_concs)

## concs$s\_Auditory concs$s\_Haptic concs$s\_Visual   
## 0.9929629 0.7991139 0.7955584

# RESULTS: all good  
  
step\_freq\_lg10WF\_SUBTLEXNL\_concs\_AIC <- stepAIC(fit\_freq\_lg10WF\_SUBTLEXNL\_concs,   
direction="both")

## Start: AIC=-18.83  
## concs$s\_freq\_lg10WF\_SUBTLEXNL ~ concs$s\_Auditory + concs$s\_Haptic +   
## concs$s\_Visual  
##   
## Df Sum of Sq RSS AIC  
## - concs$s\_Haptic 1 0.6561 381.70 -20.1258  
## <none> 381.04 -18.8261  
## - concs$s\_Visual 1 4.3316 385.37 -16.2255  
## - concs$s\_Auditory 1 15.5396 396.58 -4.5572  
##   
## Step: AIC=-20.13  
## concs$s\_freq\_lg10WF\_SUBTLEXNL ~ concs$s\_Auditory + concs$s\_Visual  
##   
## Df Sum of Sq RSS AIC  
## <none> 381.70 -20.1258  
## + concs$s\_Haptic 1 0.6561 381.04 -18.8261  
## - concs$s\_Visual 1 7.4743 389.17 -14.2330  
## - concs$s\_Auditory 1 15.2642 396.96 -6.1667

step\_freq\_lg10WF\_SUBTLEXNL\_concs\_F <- stepAIC(fit\_freq\_lg10WF\_SUBTLEXNL\_concs,   
direction="both", test="F")

## Start: AIC=-18.83  
## concs$s\_freq\_lg10WF\_SUBTLEXNL ~ concs$s\_Auditory + concs$s\_Haptic +   
## concs$s\_Visual  
##   
## Df Sum of Sq RSS AIC F Value Pr(F)   
## - concs$s\_Haptic 1 0.6561 381.70 -20.1258 0.6940 0.40531   
## <none> 381.04 -18.8261   
## - concs$s\_Visual 1 4.3316 385.37 -16.2255 4.5812 0.03292 \*   
## - concs$s\_Auditory 1 15.5396 396.58 -4.5572 16.4352 6.043e-05 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Step: AIC=-20.13  
## concs$s\_freq\_lg10WF\_SUBTLEXNL ~ concs$s\_Auditory + concs$s\_Visual  
##   
## Df Sum of Sq RSS AIC F Value Pr(F)   
## <none> 381.70 -20.1258   
## + concs$s\_Haptic 1 0.6561 381.04 -18.8261 0.6940 0.405312   
## - concs$s\_Visual 1 7.4743 389.17 -14.2330 7.9111 0.005153 \*\*   
## - concs$s\_Auditory 1 15.2642 396.96 -6.1667 16.1561 6.959e-05 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

summary(fit\_freq\_lg10WF\_SUBTLEXNL\_concs)

##   
## Call:  
## lm(formula = concs$s\_freq\_lg10WF\_SUBTLEXNL ~ concs$s\_Auditory +   
## concs$s\_Haptic + concs$s\_Visual, data = concs)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -2.89527 -0.59131 0.02548 0.59277 2.94286   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -0.004361 0.048207 -0.090 0.9280   
## concs$s\_Auditory 0.197415 0.048696 4.054 6.04e-05 \*\*\*  
## concs$s\_Haptic 0.044880 0.053875 0.833 0.4053   
## concs$s\_Visual 0.116441 0.054402 2.140 0.0329 \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.9724 on 403 degrees of freedom  
## (4 observations deleted due to missingness)  
## Multiple R-squared: 0.06148, Adjusted R-squared: 0.05449   
## F-statistic: 8.8 on 3 and 403 DF, p-value: 1.155e-05

# freq: CELEX log-10 lemma WF  
fit\_freq\_CELEX\_lem\_concs <- lm(concs$s\_freq\_CELEX\_lem ~ concs$s\_Auditory +   
concs$s\_Haptic + concs$s\_Visual, data = concs)  
stat.desc(fit\_freq\_CELEX\_lem\_concs$residuals, norm = TRUE)

## x  
## nbr.val 3.990000e+02  
## nbr.null 0.000000e+00  
## nbr.na 0.000000e+00  
## min -2.596448e+00  
## max 2.747406e+00  
## range 5.343854e+00  
## sum -3.178013e-15  
## median 1.094298e-02  
## mean -7.790903e-18  
## SE.mean 4.913264e-02  
## CI.mean.0.95 9.659194e-02  
## var 9.631926e-01  
## std.dev 9.814237e-01  
## coef.var -1.259705e+17  
## skewness -3.551406e-02  
## skew.2SE -1.453467e-01  
## kurtosis -8.965432e-02  
## kurt.2SE -1.839131e-01  
## normtest.W 9.973725e-01  
## normtest.p 7.811050e-01

# residuals distribution: good. Raw scores/2.SE < 1  
  
# Check multicollinearity: largest VIF (pref. < 10), mean VIF (pref. around 1), and   
# tolerance (pref. > 0.2)  
vif(fit\_freq\_CELEX\_lem\_concs)

## concs$s\_Auditory concs$s\_Haptic concs$s\_Visual   
## 1.009928 1.241384 1.249539

mean(vif(fit\_freq\_CELEX\_lem\_concs))

## [1] 1.16695

1/vif(fit\_freq\_CELEX\_lem\_concs)

## concs$s\_Auditory concs$s\_Haptic concs$s\_Visual   
## 0.9901697 0.8055526 0.8002952

# RESULTS: all good  
  
step\_freq\_CELEX\_lem\_concs\_AIC <- stepAIC(fit\_freq\_CELEX\_lem\_concs, direction="both")

## Start: AIC=-7.96  
## concs$s\_freq\_CELEX\_lem ~ concs$s\_Auditory + concs$s\_Haptic +   
## concs$s\_Visual  
##   
## Df Sum of Sq RSS AIC  
## - concs$s\_Haptic 1 0.7423 384.09 -9.1927  
## <none> 383.35 -7.9645  
## - concs$s\_Visual 1 4.5620 387.91 -5.2444  
## - concs$s\_Auditory 1 8.7073 392.06 -1.0032  
##   
## Step: AIC=-9.19  
## concs$s\_freq\_CELEX\_lem ~ concs$s\_Auditory + concs$s\_Visual  
##   
## Df Sum of Sq RSS AIC  
## <none> 384.09 -9.1927  
## + concs$s\_Haptic 1 0.7423 383.35 -7.9645  
## - concs$s\_Visual 1 3.8276 387.92 -7.2362  
## - concs$s\_Auditory 1 9.0221 393.12 -1.9288

step\_freq\_CELEX\_lem\_concs\_F <- stepAIC(fit\_freq\_CELEX\_lem\_concs, direction="both",   
test="F")

## Start: AIC=-7.96  
## concs$s\_freq\_CELEX\_lem ~ concs$s\_Auditory + concs$s\_Haptic +   
## concs$s\_Visual  
##   
## Df Sum of Sq RSS AIC F Value Pr(F)   
## - concs$s\_Haptic 1 0.7423 384.09 -9.1927 0.7649 0.382346   
## <none> 383.35 -7.9645   
## - concs$s\_Visual 1 4.5620 387.91 -5.2444 4.7006 0.030749 \*   
## - concs$s\_Auditory 1 8.7073 392.06 -1.0032 8.9719 0.002914 \*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Step: AIC=-9.19  
## concs$s\_freq\_CELEX\_lem ~ concs$s\_Auditory + concs$s\_Visual  
##   
## Df Sum of Sq RSS AIC F Value Pr(F)   
## <none> 384.09 -9.1927   
## + concs$s\_Haptic 1 0.7423 383.35 -7.9645 0.7649 0.382346   
## - concs$s\_Visual 1 3.8276 387.92 -7.2362 3.9462 0.047665 \*   
## - concs$s\_Auditory 1 9.0221 393.12 -1.9288 9.3018 0.002443 \*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

summary(fit\_freq\_CELEX\_lem\_concs)

##   
## Call:  
## lm(formula = concs$s\_freq\_CELEX\_lem ~ concs$s\_Auditory + concs$s\_Haptic +   
## concs$s\_Visual, data = concs)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -2.59645 -0.62930 0.01094 0.68222 2.74741   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.0005927 0.0493208 0.012 0.99042   
## concs$s\_Auditory 0.1484357 0.0495560 2.995 0.00291 \*\*  
## concs$s\_Haptic -0.0479850 0.0548676 -0.875 0.38235   
## concs$s\_Visual 0.1190287 0.0549004 2.168 0.03075 \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.9851 on 395 degrees of freedom  
## (12 observations deleted due to missingness)  
## Multiple R-squared: 0.03681, Adjusted R-squared: 0.02949   
## F-statistic: 5.032 on 3 and 395 DF, p-value: 0.001966

# length: RC1 lexicals  
fit\_RC1\_lexicals\_concs <- lm(concs$s\_RC1\_lexicals ~ concs$s\_Auditory + concs$s\_Haptic   
+ concs$s\_Visual, data = concs)  
stat.desc(fit\_RC1\_lexicals\_concs$residuals, norm = TRUE)

## x  
## nbr.val 3.830000e+02  
## nbr.null 0.000000e+00  
## nbr.na 0.000000e+00  
## min -1.790833e+00  
## max 4.935675e+00  
## range 6.726507e+00  
## sum 2.078199e-14  
## median -1.912252e-01  
## mean 5.428045e-17  
## SE.mean 4.997352e-02  
## CI.mean.0.95 9.825761e-02  
## var 9.564860e-01  
## std.dev 9.780010e-01  
## coef.var 1.801755e+16  
## skewness 1.096509e+00  
## skew.2SE 4.397413e+00  
## kurtosis 1.924309e+00  
## kurt.2SE 3.868479e+00  
## normtest.W 9.373125e-01  
## normtest.p 1.267186e-11

# residuals distribution: skewed and kurtosed. Raw scores/2.SE > 1  
# have to log-transform DV and re-run regression  
  
psych::describe(concs$s\_RC1\_lexicals)

## vars n mean sd median trimmed mad min max range skew kurtosis se  
## X1 1 383 0 1 -0.23 -0.1 0.83 -1.7 4.9 6.6 1.14 1.84 0.05

concs$log\_s\_RC1\_lexicals\_concs <- log(3 + concs$s\_RC1\_lexicals)  
  
fit\_RC1\_lexicals\_concs <- lm(concs$log\_s\_RC1\_lexicals ~ concs$s\_Auditory +   
concs$s\_Haptic + concs$s\_Visual, data = concs)  
  
# check residuals again  
stat.desc(fit\_RC1\_lexicals\_concs$residuals, norm = TRUE)

## x  
## nbr.val 3.830000e+02  
## nbr.null 0.000000e+00  
## nbr.na 0.000000e+00  
## min -7.573554e-01  
## max 1.034698e+00  
## range 1.792054e+00  
## sum -6.081941e-15  
## median -1.684797e-02  
## mean -1.589181e-17  
## SE.mean 1.574504e-02  
## CI.mean.0.95 3.095779e-02  
## var 9.494806e-02  
## std.dev 3.081364e-01  
## coef.var -1.938964e+16  
## skewness 1.960662e-01  
## skew.2SE 7.862989e-01  
## kurtosis -1.815582e-01  
## kurt.2SE -3.649903e-01  
## normtest.W 9.946692e-01  
## normtest.p 2.064935e-01

# good  
  
# Check multicollinearity: largest VIF (pref. < 10), mean VIF (pref. around 1), and   
# tolerance (pref. > 0.2)  
vif(fit\_RC1\_lexicals\_concs)

## concs$s\_Auditory concs$s\_Haptic concs$s\_Visual   
## 1.005934 1.244426 1.246755

mean(vif(fit\_RC1\_lexicals\_concs))

## [1] 1.165705

1/vif(fit\_RC1\_lexicals\_concs)

## concs$s\_Auditory concs$s\_Haptic concs$s\_Visual   
## 0.9941011 0.8035830 0.8020825

# RESULTS: all good  
  
step\_RC1\_lexicals\_concs\_AIC <- stepAIC(fit\_RC1\_lexicals\_concs, direction="both")

## Start: AIC=-894.75  
## concs$log\_s\_RC1\_lexicals ~ concs$s\_Auditory + concs$s\_Haptic +   
## concs$s\_Visual  
##   
## Df Sum of Sq RSS AIC  
## - concs$s\_Visual 1 0.05502 36.325 -896.17  
## <none> 36.270 -894.75  
## - concs$s\_Haptic 1 0.37136 36.642 -892.84  
## - concs$s\_Auditory 1 1.08098 37.351 -885.50  
##   
## Step: AIC=-896.17  
## concs$log\_s\_RC1\_lexicals ~ concs$s\_Auditory + concs$s\_Haptic  
##   
## Df Sum of Sq RSS AIC  
## <none> 36.325 -896.17  
## + concs$s\_Visual 1 0.05502 36.270 -894.75  
## - concs$s\_Haptic 1 0.63244 36.958 -891.55  
## - concs$s\_Auditory 1 1.05179 37.377 -887.23

step\_RC1\_lexicals\_concs\_F <- stepAIC(fit\_RC1\_lexicals\_concs, direction="both",   
test="F")

## Start: AIC=-894.75  
## concs$log\_s\_RC1\_lexicals ~ concs$s\_Auditory + concs$s\_Haptic +   
## concs$s\_Visual  
##   
## Df Sum of Sq RSS AIC F Value Pr(F)   
## - concs$s\_Visual 1 0.05502 36.325 -896.17 0.5749 0.4487959   
## <none> 36.270 -894.75   
## - concs$s\_Haptic 1 0.37136 36.642 -892.84 3.8805 0.0495776 \*   
## - concs$s\_Auditory 1 1.08098 37.351 -885.50 11.2955 0.0008559 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Step: AIC=-896.17  
## concs$log\_s\_RC1\_lexicals ~ concs$s\_Auditory + concs$s\_Haptic  
##   
## Df Sum of Sq RSS AIC F Value Pr(F)   
## <none> 36.325 -896.17   
## + concs$s\_Visual 1 0.05502 36.270 -894.75 0.5749 0.4487959   
## - concs$s\_Haptic 1 0.63244 36.958 -891.55 6.6159 0.0104860 \*   
## - concs$s\_Auditory 1 1.05179 37.377 -887.23 11.0028 0.0009974 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

summary(fit\_RC1\_lexicals\_concs)

##   
## Call:  
## lm(formula = concs$log\_s\_RC1\_lexicals ~ concs$s\_Auditory + concs$s\_Haptic +   
## concs$s\_Visual, data = concs)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.75736 -0.21704 -0.01685 0.21379 1.03470   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 1.04781 0.01582 66.251 < 2e-16 \*\*\*  
## concs$s\_Auditory 0.05347 0.01591 3.361 0.000856 \*\*\*  
## concs$s\_Haptic -0.03450 0.01751 -1.970 0.049578 \*   
## concs$s\_Visual -0.01345 0.01774 -0.758 0.448796   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.3094 on 379 degrees of freedom  
## (28 observations deleted due to missingness)  
## Multiple R-squared: 0.04695, Adjusted R-squared: 0.03941   
## F-statistic: 6.224 on 3 and 379 DF, p-value: 0.0003907

# distinctiveness: RC3 lexicals  
fit\_RC3\_lexicals\_concs <- lm(concs$s\_RC3\_lexicals ~ concs$s\_Auditory + concs$s\_Haptic   
+ concs$s\_Visual, data = concs)  
stat.desc(fit\_RC3\_lexicals\_concs$residuals, norm = TRUE)

## x  
## nbr.val 3.830000e+02  
## nbr.null 0.000000e+00  
## nbr.na 0.000000e+00  
## min -1.701688e+00  
## max 4.550983e+00  
## range 6.252670e+00  
## sum -1.091835e-14  
## median -2.680258e-01  
## mean -2.845592e-17  
## SE.mean 4.911158e-02  
## CI.mean.0.95 9.656288e-02  
## var 9.237760e-01  
## std.dev 9.611326e-01  
## coef.var -3.377619e+16  
## skewness 1.673295e+00  
## skew.2SE 6.710544e+00  
## kurtosis 3.357824e+00  
## kurt.2SE 6.750305e+00  
## normtest.W 8.551890e-01  
## normtest.p 2.367318e-18

# residuals distribution: skewed and kurtosed. Raw scores/2.SE > 1  
# have to log-transform DV and re-run regression  
  
psych::describe(concs$s\_RC3\_lexicals)

## vars n mean sd median trimmed mad min max range skew kurtosis  
## X1 1 383 0 1 -0.27 -0.17 0.59 -1.44 4.99 6.43 1.83 3.67  
## se  
## X1 0.05

concs$log\_s\_RC3\_lexicals <- log(3 + concs$s\_RC3\_lexicals)  
  
fit\_RC3\_lexicals\_concs <- lm(concs$log\_s\_RC3\_lexicals ~ concs$s\_Auditory +   
concs$s\_Haptic + concs$s\_Visual, data = concs)  
  
# check residuals again  
stat.desc(fit\_RC3\_lexicals\_concs$residuals, norm = TRUE)

## x  
## nbr.val 3.830000e+02  
## nbr.null 0.000000e+00  
## nbr.na 0.000000e+00  
## min -6.271740e-01  
## max 9.026380e-01  
## range 1.529812e+00  
## sum -4.562323e-15  
## median -5.616343e-02  
## mean -1.190429e-17  
## SE.mean 1.395190e-02  
## CI.mean.0.95 2.743214e-02  
## var 7.455311e-02  
## std.dev 2.730442e-01  
## coef.var -2.293663e+16  
## skewness 8.766790e-01  
## skew.2SE 3.515812e+00  
## kurtosis 7.122003e-01  
## kurt.2SE 1.431751e+00  
## normtest.W 9.479679e-01  
## normtest.p 2.347627e-10

# better though still non-normal  
  
# Check multicollinearity: largest VIF (pref. < 10), mean VIF (pref. around 1), and   
# tolerance (pref. > 0.2)  
vif(fit\_RC3\_lexicals\_concs)

## concs$s\_Auditory concs$s\_Haptic concs$s\_Visual   
## 1.005934 1.244426 1.246755

mean(vif(fit\_RC3\_lexicals\_concs))

## [1] 1.165705

1/vif(fit\_RC3\_lexicals\_concs)

## concs$s\_Auditory concs$s\_Haptic concs$s\_Visual   
## 0.9941011 0.8035830 0.8020825

# RESULTS: all good  
  
step\_RC3\_lexicals\_concs\_AIC <- stepAIC(fit\_RC3\_lexicals\_concs, direction="both")

## Start: AIC=-987.36  
## concs$log\_s\_RC3\_lexicals ~ concs$s\_Auditory + concs$s\_Haptic +   
## concs$s\_Visual  
##   
## Df Sum of Sq RSS AIC  
## - concs$s\_Visual 1 0.02418 28.503 -989.04  
## <none> 28.479 -987.36  
## - concs$s\_Auditory 1 0.43934 28.919 -983.50  
## - concs$s\_Haptic 1 1.64314 30.122 -967.88  
##   
## Step: AIC=-989.04  
## concs$log\_s\_RC3\_lexicals ~ concs$s\_Auditory + concs$s\_Haptic  
##   
## Df Sum of Sq RSS AIC  
## <none> 28.503 -989.04  
## + concs$s\_Visual 1 0.02418 28.479 -987.36  
## - concs$s\_Auditory 1 0.45652 28.960 -984.95  
## - concs$s\_Haptic 1 1.82968 30.333 -967.21

step\_RC3\_lexicals\_concs\_F <- stepAIC(fit\_RC3\_lexicals\_concs, direction="both",   
test="F")

## Start: AIC=-987.36  
## concs$log\_s\_RC3\_lexicals ~ concs$s\_Auditory + concs$s\_Haptic +   
## concs$s\_Visual  
##   
## Df Sum of Sq RSS AIC F Value Pr(F)   
## - concs$s\_Visual 1 0.02418 28.503 -989.04 0.3218 0.57085   
## <none> 28.479 -987.36   
## - concs$s\_Auditory 1 0.43934 28.919 -983.50 5.8467 0.01608 \*   
## - concs$s\_Haptic 1 1.64314 30.122 -967.88 21.8668 4.071e-06 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Step: AIC=-989.04  
## concs$log\_s\_RC3\_lexicals ~ concs$s\_Auditory + concs$s\_Haptic  
##   
## Df Sum of Sq RSS AIC F Value Pr(F)   
## <none> 28.503 -989.04   
## + concs$s\_Visual 1 0.02418 28.479 -987.36 0.3218 0.57085   
## - concs$s\_Auditory 1 0.45652 28.960 -984.95 6.0862 0.01406 \*   
## - concs$s\_Haptic 1 1.82968 30.333 -967.21 24.3927 1.179e-06 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

summary(fit\_RC3\_lexicals\_concs)

##   
## Call:  
## lm(formula = concs$log\_s\_RC3\_lexicals ~ concs$s\_Auditory + concs$s\_Haptic +   
## concs$s\_Visual, data = concs)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.62717 -0.18400 -0.05616 0.13150 0.90264   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 1.054305 0.014015 75.229 < 2e-16 \*\*\*  
## concs$s\_Auditory -0.034090 0.014098 -2.418 0.0161 \*   
## concs$s\_Haptic 0.072572 0.015520 4.676 4.07e-06 \*\*\*  
## concs$s\_Visual -0.008916 0.015716 -0.567 0.5708   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.2741 on 379 degrees of freedom  
## (28 observations deleted due to missingness)  
## Multiple R-squared: 0.07665, Adjusted R-squared: 0.06934   
## F-statistic: 10.49 on 3 and 379 DF, p-value: 1.216e-06

# freq: RC2 lexicals  
fit\_RC2\_lexicals\_concs <- lm(concs$s\_RC2\_lexicals ~ concs$s\_Auditory + concs$s\_Haptic  
 + concs$s\_Visual, data = concs)  
stat.desc(fit\_RC2\_lexicals\_concs$residuals, norm = TRUE)

## x  
## nbr.val 3.830000e+02  
## nbr.null 0.000000e+00  
## nbr.na 0.000000e+00  
## min -2.701781e+00  
## max 2.799024e+00  
## range 5.500804e+00  
## sum -9.728329e-15  
## median -5.204421e-03  
## mean -2.532129e-17  
## SE.mean 4.939207e-02  
## CI.mean.0.95 9.711437e-02  
## var 9.343578e-01  
## std.dev 9.666218e-01  
## coef.var -3.817428e+16  
## skewness 1.177816e-01  
## skew.2SE 4.723486e-01  
## kurtosis -3.253570e-02  
## kurt.2SE -6.540720e-02  
## normtest.W 9.975488e-01  
## normtest.p 8.476297e-01

# residuals distribution: good. Raw scores/2.SE < 1  
  
# Check multicollinearity: largest VIF (pref. < 10), mean VIF (pref. around 1), and   
# tolerance (pref. > 0.2)  
vif(fit\_RC2\_lexicals\_concs)

## concs$s\_Auditory concs$s\_Haptic concs$s\_Visual   
## 1.005934 1.244426 1.246755

mean(vif(fit\_RC2\_lexicals\_concs))

## [1] 1.165705

1/vif(fit\_RC2\_lexicals\_concs)

## concs$s\_Auditory concs$s\_Haptic concs$s\_Visual   
## 0.9941011 0.8035830 0.8020825

# RESULTS: all good  
  
step\_RC2\_lexicals\_concs\_AIC <- stepAIC(fit\_RC2\_lexicals\_concs, direction="both")

## Start: AIC=-19.01  
## concs$s\_RC2\_lexicals ~ concs$s\_Auditory + concs$s\_Haptic + concs$s\_Visual  
##   
## Df Sum of Sq RSS AIC  
## - concs$s\_Haptic 1 0.5196 357.44 -20.4482  
## <none> 356.92 -19.0054  
## - concs$s\_Visual 1 3.1284 360.05 -17.6631  
## - concs$s\_Auditory 1 20.6663 377.59 0.5524  
##   
## Step: AIC=-20.45  
## concs$s\_RC2\_lexicals ~ concs$s\_Auditory + concs$s\_Visual  
##   
## Df Sum of Sq RSS AIC  
## <none> 357.44 -20.4482  
## - concs$s\_Visual 1 2.6135 360.06 -19.6580  
## + concs$s\_Haptic 1 0.5196 356.92 -19.0054  
## - concs$s\_Auditory 1 21.1095 378.55 -0.4722

step\_RC2\_lexicals\_concs\_F <- stepAIC(fit\_RC2\_lexicals\_concs, direction="both",   
test="F")

## Start: AIC=-19.01  
## concs$s\_RC2\_lexicals ~ concs$s\_Auditory + concs$s\_Haptic + concs$s\_Visual  
##   
## Df Sum of Sq RSS AIC F Value Pr(F)   
## - concs$s\_Haptic 1 0.5196 357.44 -20.4482 0.5518 0.45806   
## <none> 356.92 -19.0054   
## - concs$s\_Visual 1 3.1284 360.05 -17.6631 3.3219 0.06915 .   
## - concs$s\_Auditory 1 20.6663 377.59 0.5524 21.9445 3.918e-06 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Step: AIC=-20.45  
## concs$s\_RC2\_lexicals ~ concs$s\_Auditory + concs$s\_Visual  
##   
## Df Sum of Sq RSS AIC F Value Pr(F)   
## <none> 357.44 -20.4482   
## - concs$s\_Visual 1 2.6135 360.06 -19.6580 2.7785 0.09636 .   
## + concs$s\_Haptic 1 0.5196 356.92 -19.0054 0.5518 0.45806   
## - concs$s\_Auditory 1 21.1095 378.55 -0.4722 22.4415 3.065e-06 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

summary(fit\_RC2\_lexicals\_concs)

##   
## Call:  
## lm(formula = concs$s\_RC2\_lexicals ~ concs$s\_Auditory + concs$s\_Haptic +   
## concs$s\_Visual, data = concs)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -2.7018 -0.6471 -0.0052 0.6136 2.7990   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -0.006071 0.049614 -0.122 0.9027   
## concs$s\_Auditory 0.233808 0.049911 4.684 3.92e-06 \*\*\*  
## concs$s\_Haptic -0.040811 0.054942 -0.743 0.4581   
## concs$s\_Visual 0.101407 0.055638 1.823 0.0691 .   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.9704 on 379 degrees of freedom  
## (28 observations deleted due to missingness)  
## Multiple R-squared: 0.06564, Adjusted R-squared: 0.05825   
## F-statistic: 8.875 on 3 and 379 DF, p-value: 1.067e-05

# additional var: age of acquisition  
fit\_AoA\_Brysbaertetal2014\_concs <- lm(concs$s\_AoA\_Brysbaertetal2014 ~   
concs$s\_Auditory + concs$s\_Haptic + concs$s\_Visual, data = concs)  
stat.desc(fit\_AoA\_Brysbaertetal2014\_concs$residuals, norm = TRUE)

## x  
## nbr.val 3.980000e+02  
## nbr.null 0.000000e+00  
## nbr.na 0.000000e+00  
## min -3.025869e+00  
## max 2.147610e+00  
## range 5.173479e+00  
## sum 1.313359e-14  
## median -1.358284e-02  
## mean 3.302938e-17  
## SE.mean 4.390837e-02  
## CI.mean.0.95 8.632199e-02  
## var 7.673221e-01  
## std.dev 8.759693e-01  
## coef.var 2.652091e+16  
## skewness -8.877953e-02  
## skew.2SE -3.628914e-01  
## kurtosis -2.872845e-01  
## kurt.2SE -5.885934e-01  
## normtest.W 9.957891e-01  
## normtest.p 3.666147e-01

# residuals distribution: good. Raw scores/2.SE < 1  
  
# Check multicollinearity: largest VIF (pref. < 10), mean VIF (pref. around 1), and   
# tolerance (pref. > 0.2)  
vif(fit\_AoA\_Brysbaertetal2014\_concs)

## concs$s\_Auditory concs$s\_Haptic concs$s\_Visual   
## 1.006555 1.256294 1.261234

mean(vif(fit\_AoA\_Brysbaertetal2014\_concs))

## [1] 1.174694

1/vif(fit\_AoA\_Brysbaertetal2014\_concs)

## concs$s\_Auditory concs$s\_Haptic concs$s\_Visual   
## 0.9934875 0.7959923 0.7928746

# RESULTS: all good  
  
step\_AoA\_Brysbaertetal2014\_concs\_AIC <- stepAIC(fit\_AoA\_Brysbaertetal2014\_concs,  
direction="both")

## Start: AIC=-98.41  
## concs$s\_AoA\_Brysbaertetal2014 ~ concs$s\_Auditory + concs$s\_Haptic +   
## concs$s\_Visual  
##   
## Df Sum of Sq RSS AIC  
## <none> 304.63 -98.411  
## - concs$s\_Auditory 1 4.3063 308.93 -94.824  
## - concs$s\_Haptic 1 20.6142 325.24 -74.350  
## - concs$s\_Visual 1 27.0765 331.70 -66.520

step\_AoA\_Brysbaertetal2014\_concs\_F <- stepAIC(fit\_AoA\_Brysbaertetal2014\_concs,   
direction="both", test="F")

## Start: AIC=-98.41  
## concs$s\_AoA\_Brysbaertetal2014 ~ concs$s\_Auditory + concs$s\_Haptic +   
## concs$s\_Visual  
##   
## Df Sum of Sq RSS AIC F Value Pr(F)   
## <none> 304.63 -98.411   
## - concs$s\_Auditory 1 4.3063 308.93 -94.824 5.570 0.01876 \*   
## - concs$s\_Haptic 1 20.6142 325.24 -74.350 26.662 3.854e-07 \*\*\*  
## - concs$s\_Visual 1 27.0765 331.70 -66.520 35.020 7.097e-09 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

summary(fit\_AoA\_Brysbaertetal2014\_concs)

##   
## Call:  
## lm(formula = concs$s\_AoA\_Brysbaertetal2014 ~ concs$s\_Auditory +   
## concs$s\_Haptic + concs$s\_Visual, data = concs)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -3.02587 -0.58392 -0.01358 0.66456 2.14761   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.007778 0.044092 0.176 0.8601   
## concs$s\_Auditory -0.104344 0.044213 -2.360 0.0188 \*   
## concs$s\_Haptic -0.254806 0.049347 -5.164 3.85e-07 \*\*\*  
## concs$s\_Visual -0.294074 0.049693 -5.918 7.10e-09 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.8793 on 394 degrees of freedom  
## (13 observations deleted due to missingness)  
## Multiple R-squared: 0.2327, Adjusted R-squared: 0.2268   
## F-statistic: 39.82 on 3 and 394 DF, p-value: < 2.2e-16

# additional var: concreteness  
fit\_concrete\_Brysbaertetal2014\_concs <- lm(concs$s\_concrete\_Brysbaertetal2014 ~   
concs$s\_Auditory + concs$s\_Haptic + concs$s\_Visual, data = concs)  
stat.desc(fit\_concrete\_Brysbaertetal2014\_concs$residuals, norm = TRUE)

## x  
## nbr.val 3.980000e+02  
## nbr.null 0.000000e+00  
## nbr.na 0.000000e+00  
## min -2.047456e+00  
## max 2.338430e+00  
## range 4.385886e+00  
## sum 1.426810e-14  
## median -3.294205e-02  
## mean 3.585668e-17  
## SE.mean 3.940998e-02  
## CI.mean.0.95 7.747834e-02  
## var 6.181523e-01  
## std.dev 7.862266e-01  
## coef.var 2.192692e+16  
## skewness 2.636678e-01  
## skew.2SE 1.077757e+00  
## kurtosis -8.873867e-02  
## kurt.2SE -1.818093e-01  
## normtest.W 9.907920e-01  
## normtest.p 1.382913e-02

# residuals distribution: skew. Raw scores/2.SE > 1  
# have to log-transform DV and re-run regression  
  
psych::describe(concs$s\_concrete\_Brysbaertetal2014)

## vars n mean sd median trimmed mad min max range skew kurtosis  
## X1 1 398 0 1 -0.21 -0.04 1.11 -1.71 1.85 3.56 0.38 -1.13  
## se  
## X1 0.05

concs$log\_s\_concrete\_Brysbaertetal2014 <- log(3 + concs$s\_concrete\_Brysbaertetal2014)  
  
fit\_concrete\_Brysbaertetal2014\_concs <- lm(concs$log\_s\_concrete\_Brysbaertetal2014 ~   
concs$s\_Auditory + concs$s\_Haptic + concs$s\_Visual, data = concs)  
  
# check residuals again  
stat.desc(fit\_concrete\_Brysbaertetal2014\_concs$residuals, norm = TRUE)

## x  
## nbr.val 3.980000e+02  
## nbr.null 0.000000e+00  
## nbr.na 0.000000e+00  
## min -7.098116e-01  
## max 8.091234e-01  
## range 1.518935e+00  
## sum -2.052178e-15  
## median 2.483684e-02  
## mean -5.137822e-18  
## SE.mean 1.358202e-02  
## CI.mean.0.95 2.670168e-02  
## var 7.341959e-02  
## std.dev 2.709605e-01  
## coef.var -5.273840e+16  
## skewness -8.997464e-03  
## skew.2SE -3.677765e-02  
## kurtosis -2.456084e-01  
## kurt.2SE -5.032068e-01  
## normtest.W 9.933368e-01  
## normtest.p 7.598647e-02

# good  
  
# Check multicollinearity: largest VIF (pref. < 10), mean VIF (pref. around 1), and   
# tolerance (pref. > 0.2)  
vif(fit\_concrete\_Brysbaertetal2014\_concs)

## concs$s\_Auditory concs$s\_Haptic concs$s\_Visual   
## 1.006555 1.256294 1.261234

mean(vif(fit\_concrete\_Brysbaertetal2014\_concs))

## [1] 1.174694

1/vif(fit\_concrete\_Brysbaertetal2014\_concs)

## concs$s\_Auditory concs$s\_Haptic concs$s\_Visual   
## 0.9934875 0.7959923 0.7928746

# RESULTS: all good  
  
step\_concrete\_Brysbaertetal2014\_concs\_AIC <-   
stepAIC(fit\_concrete\_Brysbaertetal2014\_concs, direction="both")

## Start: AIC=-1032.4  
## concs$log\_s\_concrete\_Brysbaertetal2014 ~ concs$s\_Auditory + concs$s\_Haptic +   
## concs$s\_Visual  
##   
## Df Sum of Sq RSS AIC  
## <none> 29.148 -1032.40  
## - concs$s\_Auditory 1 1.1547 30.302 -1018.94  
## - concs$s\_Haptic 1 3.5383 32.686 -988.80  
## - concs$s\_Visual 1 4.8804 34.028 -972.79

step\_concrete\_Brysbaertetal2014\_concs\_F <-   
stepAIC(fit\_concrete\_Brysbaertetal2014\_concs, direction="both", test="F")

## Start: AIC=-1032.4  
## concs$log\_s\_concrete\_Brysbaertetal2014 ~ concs$s\_Auditory + concs$s\_Haptic +   
## concs$s\_Visual  
##   
## Df Sum of Sq RSS AIC F Value Pr(F)   
## <none> 29.148 -1032.40   
## - concs$s\_Auditory 1 1.1547 30.302 -1018.94 15.609 9.230e-05 \*\*\*  
## - concs$s\_Haptic 1 3.5383 32.686 -988.80 47.828 1.888e-11 \*\*\*  
## - concs$s\_Visual 1 4.8804 34.028 -972.79 65.971 5.946e-15 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

summary(fit\_concrete\_Brysbaertetal2014\_concs)

##   
## Call:  
## lm(formula = concs$log\_s\_concrete\_Brysbaertetal2014 ~ concs$s\_Auditory +   
## concs$s\_Haptic + concs$s\_Visual, data = concs)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.70981 -0.21810 0.02484 0.19189 0.80912   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 1.04056 0.01364 76.294 < 2e-16 \*\*\*  
## concs$s\_Auditory -0.05403 0.01368 -3.951 9.23e-05 \*\*\*  
## concs$s\_Haptic 0.10557 0.01526 6.916 1.89e-11 \*\*\*  
## concs$s\_Visual 0.12485 0.01537 8.122 5.95e-15 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.272 on 394 degrees of freedom  
## (13 observations deleted due to missingness)  
## Multiple R-squared: 0.3571, Adjusted R-squared: 0.3522   
## F-statistic: 72.94 on 3 and 394 DF, p-value: < 2.2e-16

# Results: Iconicity of concepts and comparison with properties:   
# The properties sample was characterized by smaller advantages for Auditory   
# predictor, compared to the concepts sample. The tendency of either larger or   
# opposite scores for the Auditory strength was less evident, even though it was   
# still marginally present. This raw-figure difference was not statistically tested.  
  
  
# END