

# Cochlear Analysis

Channing Hambric

2024-09-30

#Libraries

```
library(tidyverse)
```

```
## Warning: package 'tidyverse' was built under R version 4.3.3
```

```
## Warning: package 'ggplot2' was built under R version 4.3.2
```

```
## Warning: package 'readr' was built under R version 4.3.3
```

```
## Warning: package 'forcats' was built under R version 4.3.3
```

```
## Warning: package 'lubridate' was built under R version 4.3.3
```

```
## -- Attaching core tidyverse packages ----- tidyverse 2.0.0 --
```

```
## v dplyr      1.1.3      v readr      2.1.5
```

```
## v forcats   1.0.0      v stringr   1.5.0
```

```
## v ggplot2   3.4.4      v tibble    3.2.1
```

```
## v lubridate 1.9.3      v tidyr     1.3.0
```

```
## v purrr     1.0.2
```

```
## -- Conflicts ----- tidyverse_conflicts() --
```

```
## x dplyr::filter() masks stats::filter()
```

```
## x dplyr::lag()     masks stats::lag()
```

```
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(ggthemes)
```

```
library(purrr)
```

```
library(ggpubr)
```

```
library(lubridate)
```

```
library(dplyr)
```

```
library(ggplot2)
```

```
library(hrbrthemes)
```

```
## Warning: package 'hrbrthemes' was built under R version 4.3.3
```

```
library(tidyboot)
```

```
## Warning: package 'tidyboot' was built under R version 4.3.3
```

```
library(gridExtra)
```

```
##  
## Attaching package: 'gridExtra'  
##  
## The following object is masked from 'package:dplyr':  
##  
##      combine
```

```
library(lme4)
```

```
## Loading required package: Matrix  
##  
## Attaching package: 'Matrix'  
##  
## The following objects are masked from 'package:tidyr':  
##  
##      expand, pack, unpack
```

```
library(emmeans)  
library(data.table)
```

```
## Warning: package 'data.table' was built under R version 4.3.3
```

```
##  
## Attaching package: 'data.table'  
##  
## The following objects are masked from 'package:lubridate':  
##  
##      hour, isoweek, mday, minute, month, quarter, second, wday, week,  
##      yday, year  
##  
## The following objects are masked from 'package:dplyr':  
##  
##      between, first, last  
##  
## The following object is masked from 'package:purrr':  
##  
##      transpose
```

```
library(paletteer)
```

```
## Warning: package 'paletteer' was built under R version 4.3.3
```

```
library(forcats)  
library(knitr)
```

```
#Importing Fluency/Clustering Data
```

```

#For assigning CI/NH status
participant_groupings <- read.csv("../participant_groupings.csv")

#Animals
#Animals - W2V 50 dim
animals_50_w2v_fluency <- read.csv("../output/animals/word2vec/50/individual_descriptive_stats.csv") %>%
#Animals - W2V 100 dim
animals_100_w2v_fluency <- read.csv("../output/animals/word2vec/100/individual_descriptive_stats.csv") %>%
#Animals - W2V 200 dim
animals_200_w2v_fluency <- read.csv("../output/animals/word2vec/200/individual_descriptive_stats.csv") %>%
#Animals - W2V 300 dim
animals_300_w2v_fluency <- read.csv("../output/animals/word2vec/300/individual_descriptive_stats.csv") %>%
#Animals - S2V 50 dim
animals_50_s2v_fluency <- read.csv("../output/animals/speech2vec/50/individual_descriptive_stats.csv") %>%
#Animals - S2V 100 dim
animals_100_s2v_fluency <- read.csv("../output/animals/speech2vec/100/individual_descriptive_stats.csv") %>%
#Animals - S2V 200 dim
animals_200_s2v_fluency <- read.csv("../output/animals/speech2vec/200/individual_descriptive_stats.csv") %>%
#Animals - S2V 300 dim
animals_300_s2v_fluency <- read.csv("../output/animals/speech2vec/300/individual_descriptive_stats.csv") %>%

full_animals_fluency = rbind(animals_50_w2v_fluency, animals_100_w2v_fluency, animals_200_w2v_fluency, animals_300_w2v_fluency,
                             animals_50_s2v_fluency, animals_100_s2v_fluency, animals_200_s2v_fluency, animals_300_s2v_fluency)

#Linking participant groupings
full_animals_fluency = full_animals_fluency %>%
  mutate("domain"="animals") %>%
  left_join(participant_groupings) %>% filter(!is.na(Group)) %>%
  mutate(Group = fct_recode(Group, "normal hearing" = "NH", "cochlear implant" = "CI"),
         Group = fct_relevel(Group, "normal hearing", "cochlear implant")) %>%
  rename("Numb_of_Items"="X._of_Items")

```

## Joining with 'by = join\_by(Subject)'

```

#Foods
#Foods - W2V 50 dim
foods_50_w2v_fluency <- read.csv("../output/foods/word2vec/50/individual_descriptive_stats.csv") %>% mutate("domain"="foods")
#Foods - W2V 100 dim
foods_100_w2v_fluency <- read.csv("../output/foods/word2vec/100/individual_descriptive_stats.csv") %>% mutate("domain"="foods")
#Foods - W2V 200 dim
foods_200_w2v_fluency <- read.csv("../output/foods/word2vec/200/individual_descriptive_stats.csv") %>% mutate("domain"="foods")
#Foods - W2V 300 dim
foods_300_w2v_fluency <- read.csv("../output/foods/word2vec/300/individual_descriptive_stats.csv") %>% mutate("domain"="foods")
#Foods - S2V 50 dim
foods_50_s2v_fluency <- read.csv("../output/foods/speech2vec/50/individual_descriptive_stats.csv") %>% mutate("domain"="foods")
#Foods - S2V 100 dim
foods_100_s2v_fluency <- read.csv("../output/foods/speech2vec/100/individual_descriptive_stats.csv") %>% mutate("domain"="foods")
#Foods - S2V 200 dim
foods_200_s2v_fluency <- read.csv("../output/foods/speech2vec/200/individual_descriptive_stats.csv") %>% mutate("domain"="foods")
#Foods - S2V 300 dim
foods_300_s2v_fluency <- read.csv("../output/foods/speech2vec/300/individual_descriptive_stats.csv") %>% mutate("domain"="foods")

full_foods_fluency = rbind(foods_50_w2v_fluency, foods_100_w2v_fluency, foods_200_w2v_fluency, foods_300_w2v_fluency,
                           foods_50_s2v_fluency, foods_100_s2v_fluency, foods_200_s2v_fluency, foods_300_s2v_fluency)

```

```

#Linking participant groupings
full_foods_fluency = full_foods_fluency%>%
  mutate("domain"="foods") %>%
  left_join(participant_groupings) %>% filter(!is.na(Group)) %>%
  mutate(Group = fct_recode(Group, "normal hearing" = "NH", "cochlear implant" = "CI"),
         Group = fct_relevel(Group, "normal hearing", "cochlear implant")) %>%
  rename("Numb_of_Items"="X._of_Items")

```

## Joining with 'by = join\_by(Subject)'

#Importing Lexical Data

```

#Animals
#Animals - W2V 50 dim
animals_50_w2v_lexical <- read.csv("../output/animals/word2vec/50/lexical_results.csv") %>%mutate("dimension"=50)
#Animals - W2V 100 dim
animals_100_w2v_lexical <- read.csv("../output/animals/word2vec/100/lexical_results.csv")%>%mutate("dimension"=100)
#Animals - W2V 200 dim
animals_200_w2v_lexical <- read.csv("../output/animals/word2vec/200/lexical_results.csv")%>%mutate("dimension"=200)
#Animals - W2V 300 dim
animals_300_w2v_lexical <- read.csv("../output/animals/word2vec/300/lexical_results.csv")%>%mutate("dimension"=300)
#Animals - S2V 50 dim
animals_50_s2v_lexical <- read.csv("../output/animals/speech2vec/50/lexical_results.csv")%>%mutate("dimension"=50)
#Animals - S2V 100 dim
animals_100_s2v_lexical <- read.csv("../output/animals/speech2vec/100/lexical_results.csv")%>%mutate("dimension"=100)
#Animals - S2V 200 dim
animals_200_s2v_lexical <- read.csv("../output/animals/speech2vec/200/lexical_results.csv")%>%mutate("dimension"=200)
#Animals - S2V 300 dim
animals_300_s2v_lexical <- read.csv("../output/animals/speech2vec/300/lexical_results.csv")%>%mutate("dimension"=300)

full_animals_lexical = rbind(animals_50_w2v_lexical,animals_100_w2v_lexical,animals_200_w2v_lexical,animals_300_w2v_lexical,
                             animals_50_s2v_lexical,animals_100_s2v_lexical,animals_200_s2v_lexical,animals_300_s2v_lexical)

#Linking participant groupings
full_animals_lexical = full_animals_lexical %>%
  mutate("domain"="animals") %>%
  left_join(participant_groupings) %>% filter(!is.na(Group)) %>%
  mutate(Group = fct_recode(Group, "normal hearing" = "NH", "cochlear implant" = "CI"),
         Group = fct_relevel(Group, "normal hearing", "cochlear implant"))

```

## Joining with 'by = join\_by(Subject)'

```

#Foods
#Foods - W2V 50 dim
foods_50_w2v_lexical <- read.csv("../output/foods/word2vec/50/lexical_results.csv") %>%mutate("dimension"=50)
#Foods - W2V 100 dim
foods_100_w2v_lexical <- read.csv("../output/foods/word2vec/100/lexical_results.csv")%>%mutate("dimension"=100)
#Foods - W2V 200 dim
foods_200_w2v_lexical <- read.csv("../output/foods/word2vec/200/lexical_results.csv")%>%mutate("dimension"=200)
#Foods - W2V 300 dim
foods_300_w2v_lexical <- read.csv("../output/foods/word2vec/300/lexical_results.csv")%>%mutate("dimension"=300)
#Foods - S2V 50 dim
foods_50_s2v_lexical <- read.csv("../output/foods/speech2vec/50/lexical_results.csv")%>%mutate("dimension"=50)

```

```

#Foods - S2V 100 dim
foods_100_s2v_lexical <- read.csv("../output/foods/speech2vec/100/lexical_results.csv") %>% mutate("dimension" = 100)
#Foods - S2V 200 dim
foods_200_s2v_lexical <- read.csv("../output/foods/speech2vec/200/lexical_results.csv") %>% mutate("dimension" = 200)
#Foods - S2V 300 dim
foods_300_s2v_lexical <- read.csv("../output/foods/speech2vec/300/lexical_results.csv") %>% mutate("dimension" = 300)

full_foods_lexical = rbind(foods_50_w2v_lexical, foods_100_w2v_lexical, foods_200_w2v_lexical, foods_300_w2v_lexical)

#Linking participant groupings
full_foods_lexical = full_foods_lexical %>%
  mutate("domain" = "foods") %>%
  left_join(participant_groupings) %>% filter(!is.na(Group)) %>%
  mutate(Group = fct_recode(Group, "normal hearing" = "NH", "cochlear implant" = "CI"),
         Group = fct_relevel(Group, "normal hearing", "cochlear implant"))

```

## Joining with 'by = join\_by(Subject)'

#Importing USE lexical data

```

animals_USE_lexical <- read.csv("../output/animals/USE/lexical_results.csv") %>% mutate("dimension" = 512)
#Foods - W2V 100 dim
foods_USE_lexical <- read.csv("../output/foods/USE/lexical_results.csv") %>% mutate("dimension" = 512) %>%
  mutate("domain" = "foods")

full_USE_lexical <- rbind(animals_USE_lexical, foods_USE_lexical)
full_USE_lexical = full_USE_lexical %>%
  left_join(participant_groupings) %>% filter(!is.na(Group)) %>%
  mutate(Group = fct_recode(Group, "normal hearing" = "NH", "cochlear implant" = "CI"),
         Group = fct_relevel(Group, "normal hearing", "cochlear implant"))

```

## Joining with 'by = join\_by(Subject)'

#Importing Model Results - S2V & W2V Only for now

```

#Animals
#Animals - W2V 50 dim
animals_50_w2v_model <- read.csv("../output/animals/word2vec/50/model_results.csv") %>% mutate("dimension" = 50)
#Animals - W2V 100 dim
animals_100_w2v_model <- read.csv("../output/animals/word2vec/100/model_results.csv") %>% mutate("dimension" = 100)
#Animals - W2V 200 dim
animals_200_w2v_model <- read.csv("../output/animals/word2vec/200/model_results.csv") %>% mutate("dimension" = 200)
#Animals - W2V 300 dim
animals_300_w2v_model <- read.csv("../output/animals/word2vec/300/model_results.csv") %>% mutate("dimension" = 300)
#Animals - S2V 50 dim
animals_50_s2v_model <- read.csv("../output/animals/speech2vec/50/model_results.csv") %>% mutate("dimension" = 50)
#Animals - S2V 100 dim
animals_100_s2v_model <- read.csv("../output/animals/speech2vec/100/model_results.csv") %>% mutate("dimension" = 100)
#Animals - S2V 200 dim
animals_200_s2v_model <- read.csv("../output/animals/speech2vec/200/model_results.csv") %>% mutate("dimension" = 200)
#Animals - S2V 300 dim
animals_300_s2v_model <- read.csv("../output/animals/speech2vec/300/model_results.csv") %>% mutate("dimension" = 300)

```

```

full_animals_model = rbind(animals_50_w2v_model, animals_100_w2v_model, animals_200_w2v_model, animals_300_w2v_model)

#Linking participant groupings
full_animals_model = full_animals_model %>%
  mutate("domain"="animals") %>%
  left_join(participant_groupings) %>% filter(!is.na(Group)) %>%
  mutate(Group = fct_recode(Group, "normal hearing" = "NH", "cochlear implant" = "CI"),
         Group = fct_relevel(Group, "normal hearing", "cochlear implant"))

```

## Joining with 'by = join\_by(Subject)'

```

#Foods
#Foods - W2V 50 dim
foods_50_w2v_model <- read.csv("../output/foods/word2vec/50/model_results.csv") %>% mutate("dimension"="W2V_50")
#Foods - W2V 100 dim
foods_100_w2v_model <- read.csv("../output/foods/word2vec/100/model_results.csv") %>% mutate("dimension"="W2V_100")
#Foods - W2V 200 dim
foods_200_w2v_model <- read.csv("../output/foods/word2vec/200/model_results.csv") %>% mutate("dimension"="W2V_200")
#Foods - W2V 300 dim
foods_300_w2v_model <- read.csv("../output/foods/word2vec/300/model_results.csv") %>% mutate("dimension"="W2V_300")
#Foods - S2V 50 dim
foods_50_s2v_model <- read.csv("../output/foods/speech2vec/50/model_results.csv") %>% mutate("dimension"="S2V_50")
#Foods - S2V 100 dim
foods_100_s2v_model <- read.csv("../output/foods/speech2vec/100/model_results.csv") %>% mutate("dimension"="S2V_100")
#Foods - S2V 200 dim
foods_200_s2v_model <- read.csv("../output/foods/speech2vec/200/model_results.csv") %>% mutate("dimension"="S2V_200")
#Foods - S2V 300 dim
foods_300_s2v_model <- read.csv("../output/foods/speech2vec/300/model_results.csv") %>% mutate("dimension"="S2V_300")

full_foods_model = rbind(foods_50_w2v_model, foods_100_w2v_model, foods_200_w2v_model, foods_300_w2v_model,
                          foods_50_s2v_model, foods_100_s2v_model, foods_200_s2v_model, foods_300_s2v_model)

#Linking participant groupings
full_foods_model = full_foods_model %>%
  mutate("domain"="foods") %>%
  left_join(participant_groupings) %>% filter(!is.na(Group)) %>%
  mutate(Group = fct_recode(Group, "normal hearing" = "NH", "cochlear implant" = "CI"),
         Group = fct_relevel(Group, "normal hearing", "cochlear implant"))

```

## Joining with 'by = join\_by(Subject)'

#Fluency Analyses

```

#ANIMALS
animals_fluency<- full_animals_fluency%>% drop_na()

#Getting average fluency per participant
average_animals_fluency = animals_fluency %>% group_by(Subject, Group) %>%
  summarise(average_animals_fluency_count= mean(Numb_of_Items))

```

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

```
#Getting average fluency per group
animals_group_fluency = average_animals_fluency %>%
  group_by(Group) %>%
  summarise(
    fluency_mean = mean(average_animals_fluency_count),
    fluency_sd = sd(average_animals_fluency_count))
kable(animals_group_fluency)
```

Group	fluency_mean	fluency_sd
normal hearing	21.28125	4.55953
cochlear implant	17.81250	5.38479

```
#Does fluency score differ by Group
animals_group_lm = lmer(data = animals_fluency, Numb_of_Items~ Group + (1|Subject))
```

```
## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, :
## Model failed to converge with max|grad| = 5.90022 (tol = 0.002, component 1)
```

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

```
summary(animals_group_lm)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: Numb_of_Items ~ Group + (1 | Subject)
## Data: animals_fluency
##
## REML criterion at convergence: -11007.6
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -3.064e-07 -5.437e-08  1.977e-08  1.285e-07  3.954e-07
##
## Random effects:
## Groups Name Variance Std.Dev.
## Subject (Intercept) 3.603e+00 1.898e+00
## Residual 5.166e-13 7.188e-07
## Number of obs: 512, groups: Subject, 64
##
## Fixed effects:
##              Estimate Std. Error t value
## (Intercept)      18.2618    0.3085  59.195
## Groupcochlear implant -0.1748    0.4562  -0.383
##
## Correlation of Fixed Effects:
##              (Intr)
## Grpcchlrmp -0.676
## optimizer (nloptwrap) convergence code: 0 (OK)
## Model failed to converge with max|grad| = 5.90022 (tol = 0.002, component 1)
## Model is nearly unidentifiable: very large eigenvalue
## - Rescale variables?
```

```
car::Anova(animals_group_lm)
```

```
## Analysis of Deviance Table (Type II Wald chisquare tests)
```

```
##
```

```
## Response: Numb_of_Items
```

```
##           Chisq Df Pr(>Chisq)
```

```
## Group 0.1469  1      0.7015
```

```
animals_numb<-animals_fluency %>%
```

```
  group_by(Group) %>% tidyboot_mean(Numb_of_Items, nboot = 1000, na.rm = T) %>%
```

```
  ggplot(aes(x = Group, y = empirical_stat, fill = Group)) +
```

```
    geom_bar(stat = 'identity', color="black") +
```

```
    geom_errorbar(aes(ymin = ci_lower, ymax = ci_upper), width = 0.05) +
```

```
    geom_point(data = animals_fluency, aes(x= Group, y = Numb_of_Items, group = Group),  
              position = position_jitterdodge(0.1), alpha = 0.3)+
```

```
    labs(y = 'number of items produced', x = "") +
```

```
    theme_few() +
```

```
    theme(aspect.ratio = 1, legend.position = 'none')+ annotate("text", x = 2.1, y = 50, label = "anima",  
                      hjust = 2, size = 4, fontface = "bold", color = "black") + scale_y_continuous(limits = c(0,
```

```
  scale_colour_paletteer_d("nationalparkcolors::Acadia")+
```

```
  scale_color_paletteer_d("nationalparkcolors::Acadia")+
```

```
  scale_fill_paletteer_d("nationalparkcolors::Acadia")
```

```
## Warning: There was 1 warning in 'dplyr::mutate()'.  
## i In argument: 'strap = purrr::map(strap, dplyr::as_data_frame)'.  
## Caused by warning:  
## ! 'as_data_frame()' was deprecated in tibble 2.0.0.  
## i Please use 'as_tibble()' (with slightly different semantics) to convert to a  
##   tibble, or 'as.data.frame()' to convert to a data frame.  
## i The deprecated feature was likely used in the purrr package.  
##   Please report the issue at <https://github.com/tidyverse/purrr/issues>.
```

```
## Warning: 'cols' is now required when using 'unnest()'.  
## i Please use 'cols = c(strap)'.
```

```
## Scale for colour is already present.
```

```
## Adding another scale for colour, which will replace the existing scale.
```

```
#Foods
```

```
foods_fluency<- full_foods_fluency%>% drop_na()
```

```
#Getting average fluencies per participant
```

```
average_foods_fluency = foods_fluency %>% group_by(Subject,Group) %>%
```

```
  summarise(average_foods_fluency_count= mean(Numb_of_Items))
```

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

```
## '.groups' argument.
```

```
#Average fluency per group
```

```
foods_group_fluency = average_foods_fluency %>%
```

```
  group_by(Group) %>%
```



```

summarise(
  fluency_mean = mean(average_foods_fluency_count),
  fluency_sd = sd(average_foods_fluency_count))
kable(foods_group_fluency)

```

Group	fluency_mean	fluency_sd
normal hearing	25.36000	7.543651
cochlear implant	19.63158	7.938727

```
#Does fluency score differ by Group
```

```
foods_group_lm = lmer(data = foods_fluency, Numb_of_Items ~ Group + (1|Subject))
```

```
## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, :
## Model failed to converge with max|grad| = 1.05926 (tol = 0.002, component 1)
```

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

```
summary(foods_group_lm)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: Numb_of_Items ~ Group + (1 | Subject)
## Data: foods_fluency
##
## REML criterion at convergence: -7469.7
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -1.683e-07 -4.430e-08  8.860e-09  9.081e-08  2.835e-07
##
## Random effects:
## Groups Name Variance Std.Dev.
## Subject (Intercept) 7.655e+00 2.767e+00
## Residual 6.431e-13 8.020e-07
## Number of obs: 352, groups: Subject, 44
##
## Fixed effects:
##              Estimate Std. Error t value
## (Intercept)    22.6977    0.5130  44.249
## Groupcochlear implant -3.3873    0.8083  -4.191
##
## Correlation of Fixed Effects:
##              (Intr)
## Grpcchlrmp -0.635
## optimizer (nloptwrap) convergence code: 0 (OK)
## Model failed to converge with max|grad| = 1.05926 (tol = 0.002, component 1)
## Model is nearly unidentifiable: very large eigenvalue
## - Rescale variables?
```

```
car::Anova(foods_group_lm)
```

```
## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: Numb_of_Items
##      Chisq Df Pr(>Chisq)
## Group 17.564  1  2.778e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
#Combining across animals and foods
```

```
comb_fluency <- rbind(animals_fluency,foods_fluency)
```

```
#Getting average fluencies per participant
```

```
average_comb_fluency = comb_fluency %>% group_by(Subject,Group,domain) %>%
  summarise(average_fluency_count= mean(Numb_of_Items))
```

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

```
#Does fluency differ by group & domain
```

```
comb_group_lm = lmer(data = comb_fluency, Numb_of_Items~ Group*domain + (1|Subject))
summary(comb_group_lm)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: Numb_of_Items ~ Group * domain + (1 | Subject)
##      Data: comb_fluency
##
## REML criterion at convergence: 4410.7
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.37635 -0.41219  0.00492  0.41106  2.44996
##
## Random effects:
##   Groups   Name                Variance Std.Dev.
##   Subject  (Intercept)  30.028     5.480
##   Residual                    7.229     2.689
## Number of obs: 864, groups: Subject, 64
##
## Fixed effects:
##                                     Estimate Std. Error t value
## (Intercept)                        21.2813     0.9832  21.646
## Groupcochlear implant               -3.4688     1.3904  -2.495
## domainfoods                         4.4683     0.2684  16.646
## Groupcochlear implant:domainfoods  -3.4444     0.4082  -8.438
##
## Correlation of Fixed Effects:
##              (Intr) Grpcci dmnfds
## Grpcchlrimp -0.707
## domainfoods -0.107  0.076
## Grpcimplnt:  0.070 -0.100 -0.658
```

```
car::Anova(comb_group_lm)
```

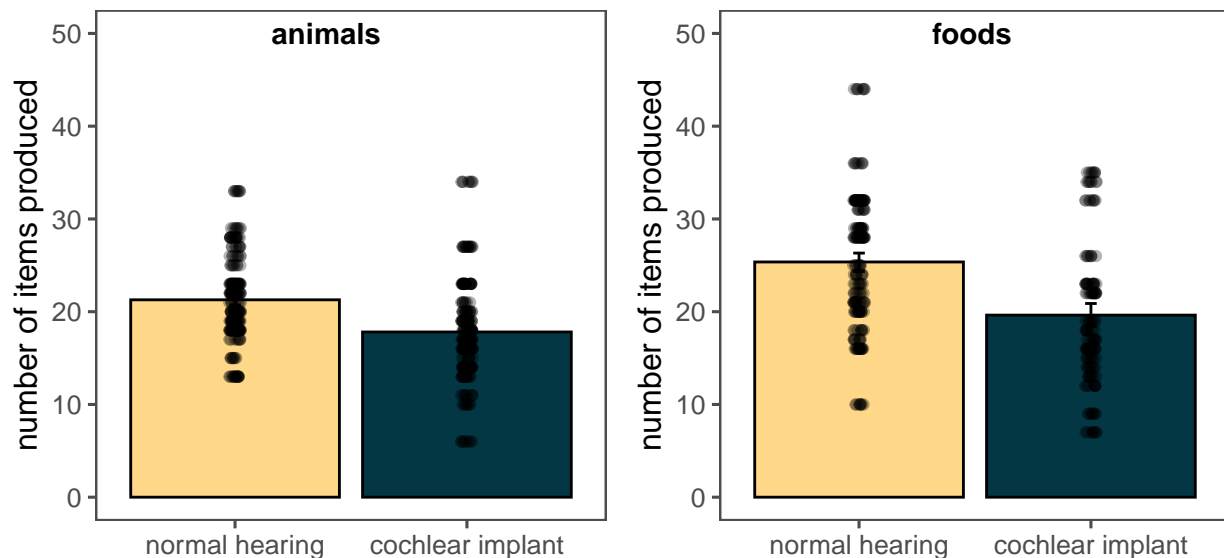
```
## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: Numb_of_Items
##           Chisq Df Pr(>Chisq)
## Group      11.230  1  0.0008048 ***
## domain     216.967  1 < 2.2e-16 ***
## Group:domain 71.208  1 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
foods_num<-foods_fluency %>%
  group_by(Group) %>% tidyboot_mean(Numb_of_Items, nboot = 1000, na.rm = T) %>%
  ggplot(aes(x = Group, y = empirical_stat, fill = Group)) +
    geom_bar(stat = 'identity',color="black") +
    geom_errorbar(aes(ymin = ci_lower, ymax = ci_upper), width = 0.05) +
    geom_point(data = foods_fluency, aes(x= Group, y = Numb_of_Items, group = Group),
              position = position_jitterdodge(0.1),alpha = 0.3)+
    labs(y = 'number of items produced', x = "") +
    theme_few() +
    theme(aspect.ratio = 1, legend.position = 'none')+ annotate("text", x = 2, y = 50, label = "foods",
      hjust = 2, size = 4, fontface = "bold", color = "black") +
    scale_colour_paletteer_d("nationalparkcolors::Acadia")+
    scale_color_paletteer_d("nationalparkcolors::Acadia")+
    scale_fill_paletteer_d("nationalparkcolors::Acadia")
```

```
## Warning: 'cols' is now required when using 'unnest()'.
## i Please use 'cols = c(strap)'.
```

```
## Scale for colour is already present.
## Adding another scale for colour, which will replace the existing scale.
```

```
gridExtra::grid.arrange(animals_num, foods_num, ncol=2)
```



#Lexical Analyses

*#ANIMALS*

*#getting avg semantic similarity - w2v & s2v*

```
animals_sem_sim_comparison = full_animals_lexical %>% group_by(Subject,Group,dimension,struct) %>%
  summarise(avg_sem_sim = mean(Semantic_Similarity),
            items = n())
```

## 'summarise()' has grouped output by 'Subject', 'Group', 'dimension'. You can  
## override using the '.groups' argument.

*#USE similarity*

```
all_struct_animals<-rbind(full_animals_lexical,full_USE_lexical %>% filter(domain=="animals"))
```

*#Avg semantic similarity for each method by participant*

```
animals_USE_sem_sim_comparison <-all_struct_animals %>%group_by(Subject,Group,dimension,struct) %>%
  summarise(avg_sem_sim = mean(Semantic_Similarity),
            items = n())
```

## 'summarise()' has grouped output by 'Subject', 'Group', 'dimension'. You can  
## override using the '.groups' argument.

*#JUST W2V/S2V: Does avg semantic similarity differ by group x struct x dim*

```
animals_sem_sim_lm = lmer(data = animals_sem_sim_comparison, avg_sem_sim ~ Group*dimension*struct+ (1|S
summary(animals_sem_sim_lm)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: avg_sem_sim ~ Group * dimension * struct + (1 | Subject)
## Data: animals_sem_sim_comparison
##
## REML criterion at convergence: -2425.9
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -3.02861 -0.55162 -0.03121  0.58808  2.76073
##
## Random effects:
## Groups Name Variance Std.Dev.
## Subject (Intercept) 0.0007945 0.02819
## Residual 0.0002629 0.01621
## Number of obs: 512, groups: Subject, 64
##
## Fixed effects:
##
## Estimate Std. Error t value
## (Intercept) 0.460032 0.005748 80.028
## Groupcochlear implant 0.012661 0.008129 1.557
## dimension200 -0.082877 0.004053 -20.446
## dimension300 -0.124188 0.004053 -30.638
## dimension50 0.085877 0.004053 21.187
## structw2v -0.009052 0.004053 -2.233
## Groupcochlear implant:dimension200 -0.005085 0.005732 -0.887
## Groupcochlear implant:dimension300 -0.006762 0.005732 -1.180
## Groupcochlear implant:dimension50 0.005719 0.005732 0.998
## Groupcochlear implant:structw2v 0.003569 0.005732 0.623
## dimension200:structw2v -0.007812 0.005732 -1.363
## dimension300:structw2v -0.012417 0.005732 -2.166
## dimension50:structw2v 0.023098 0.005732 4.029
## Groupcochlear implant:dimension200:structw2v -0.001353 0.008107 -0.167
## Groupcochlear implant:dimension300:structw2v 0.001209 0.008107 0.149
## Groupcochlear implant:dimension50:structw2v -0.006835 0.008107 -0.843
##
## Correlation matrix not shown by default, as p = 16 > 12.
## Use print(x, correlation=TRUE) or
## vcov(x) if you need it
```

```
car::Anova(animals_sem_sim_lm)
```

```
## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: avg_sem_sim
##
## Chisq Df Pr(>Chisq)
## Group 2.8037 1 0.09405 .
## dimension 15432.2855 3 < 2.2e-16 ***
## struct 26.8281 1 2.224e-07 ***
## Group:dimension 6.4855 3 0.09024 .
## Group:struct 0.4049 1 0.52456
## dimension:struct 72.8943 3 1.024e-15 ***
## Group:dimension:struct 1.1513 3 0.76470
```

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

emmeans(animals_sem_sim_lm, pairwise~dimension*struct, simple="struct")

## NOTE: Results may be misleading due to involvement in interactions

## $emmeans
## dimension struct emmean      SE df lower.CL upper.CL
## 100          s2v   0.466 0.00406 100    0.458    0.474
## 200          s2v   0.381 0.00406 100    0.373    0.389
## 300          s2v   0.339 0.00406 100    0.331    0.347
## 50           s2v   0.555 0.00406 100    0.547    0.563
## 100          w2v   0.459 0.00406 100    0.451    0.467
## 200          w2v   0.365 0.00406 100    0.357    0.373
## 300          w2v   0.320 0.00406 100    0.312    0.328
## 50           w2v   0.568 0.00406 100    0.559    0.576
##
## Results are averaged over the levels of: Group
## Degrees-of-freedom method: kenward-roger
## Confidence level used: 0.95
##
## $contrasts
## dimension = 100:
## contrast estimate      SE df t.ratio p.value
## s2v - w2v  0.00727 0.00287 434   2.536  0.0116
##
## dimension = 200:
## contrast estimate      SE df t.ratio p.value
## s2v - w2v  0.01576 0.00287 434   5.497 <.0001
##
## dimension = 300:
## contrast estimate      SE df t.ratio p.value
## s2v - w2v  0.01908 0.00287 434   6.657 <.0001
##
## dimension = 50:
## contrast estimate      SE df t.ratio p.value
## s2v - w2v -0.01241 0.00287 434  -4.331 <.0001
##
## Results are averaged over the levels of: Group
## Degrees-of-freedom method: kenward-roger

#Icl USE: Does semantic similarity differ by group x struct
animals_USE_sem_sim_lm = lmer(data = animals_USE_sem_sim_comparison, avg_sem_sim ~ Group*struct+ (1|Subj

## boundary (singular) fit: see help('isSingular')

summary(animals_USE_sem_sim_lm)

## Linear mixed model fit by REML ['lmerMod']
## Formula: avg_sem_sim ~ Group * struct + (1 | Subject)
```

```
## Data: animals_USE_sem_sim_comparison
##
## REML criterion at convergence: -1097.3
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.10786 -0.79377 -0.08577  0.68386  2.64387
##
## Random effects:
##   Groups   Name                Variance Std.Dev.
##   Subject  (Intercept)  0.000000  0.0000
##   Residual                    0.008154  0.0903
## Number of obs: 576, groups: Subject, 64
##
## Fixed effects:
##
##              Estimate Std. Error t value
## (Intercept)      0.429736   0.007982  53.841
## Groupcochlear implant      0.011129   0.011288   0.986
## structUSE        -0.192713   0.017847 -10.798
## structw2v        -0.008335   0.011288  -0.738
## Groupcochlear implant:structUSE -0.027482   0.025240  -1.089
## Groupcochlear implant:structw2v  0.001824   0.015963   0.114
##
## Correlation of Fixed Effects:
##              (Intr) Grpcci strUSE strct2 Gi:USE
## Grpcchlrmp -0.707
## structUSE   -0.447  0.316
## structw2v   -0.707  0.500  0.316
## Gimplnt:USE  0.316 -0.447 -0.707 -0.224
## Gimplnt:st2  0.500 -0.707 -0.224 -0.707  0.316
## optimizer (nloptwrap) convergence code: 0 (OK)
## boundary (singular) fit: see help('isSingular')
```

```
car::Anova(animals_USE_sem_sim_lm)
```

```
## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: avg_sem_sim
##              Chisq Df Pr(>Chisq)
## Group          1.3945  1    0.2377
## struct        287.6382  2    <2e-16 ***
## Group:struct    1.4192  2    0.4918
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
emmeans(animals_USE_sem_sim_lm, pairwise ~ Group * struct, simple = "struct")
```

```
## $emmeans
##   Group      struct emmean      SE df lower.CL upper.CL
## normal hearing s2v      0.430 0.00798 264    0.414    0.445
## cochlear implant s2v      0.441 0.00798 264    0.425    0.457
## normal hearing  USE      0.237 0.01596 570    0.206    0.268
## cochlear implant USE      0.221 0.01596 570    0.189    0.252
```

```
## normal hearing w2v      0.421 0.00798 264      0.406      0.437
## cochlear implant w2v    0.434 0.00798 264      0.419      0.450
##
## Degrees-of-freedom method: kenward-roger
## Confidence level used: 0.95
##
## $contrasts
## Group = normal hearing:
## contrast estimate      SE df t.ratio p.value
## s2v - USE  0.19271 0.0178 508  10.798 <.0001
## s2v - w2v  0.00833 0.0113 508   0.738  0.7408
## USE - w2v -0.18438 0.0178 508 -10.331 <.0001
##
## Group = cochlear implant:
## contrast estimate      SE df t.ratio p.value
## s2v - USE  0.22020 0.0178 508  12.338 <.0001
## s2v - w2v  0.00651 0.0113 508   0.577  0.8326
## USE - w2v -0.21368 0.0178 508 -11.973 <.0001
##
## Degrees-of-freedom method: kenward-roger
## P value adjustment: tukey method for comparing a family of 3 estimates
```

```
#getting avg phonological similarity
```

```
animals_phon_sim_comparison = full_animals_lexical %>% group_by(Subject,Group,dimension,struct) %>%
  summarise(avg_phon_sim = mean(Phonological_Similarity),
            items = n())
```

```
## 'summarise()' has grouped output by 'Subject', 'Group', 'dimension'. You can
## override using the '.groups' argument.
```

```
#Does avg semantic similarity differ by group
```

```
animals_phon_sim_lm = lmer(data = animals_phon_sim_comparison, avg_phon_sim ~ Group + (1|Subject))
```

```
## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, :
## Model failed to converge with max|grad| = 0.276428 (tol = 0.002, component 1)
```

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

```
summary(animals_phon_sim_lm)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: avg_phon_sim ~ Group + (1 | Subject)
## Data: animals_phon_sim_comparison
##
## REML criterion at convergence: -16100.4
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -3.830e-07 -9.850e-08 -5.490e-09  1.023e-07  5.766e-07
##
## Random effects:
```



```
## Groups Name Variance Std.Dev.
## Subject (Intercept) 9.924e-05 9.962e-03
## Residual 2.555e-17 5.054e-09
## Number of obs: 512, groups: Subject, 64
##
## Fixed effects:
## Estimate Std. Error t value
## (Intercept) 0.0693574 0.0016786 41.319
## Groupcochlear implant 0.0002793 0.0024347 0.115
##
## Correlation of Fixed Effects:
## (Intr)
## Grpcchlrmp -0.689
## optimizer (nloptwrap) convergence code: 0 (OK)
## Model failed to converge with max|grad| = 0.276428 (tol = 0.002, component 1)
## Model is nearly unidentifiable: very large eigenvalue
## - Rescale variables?
```

```
car::Anova(animals_phon_sim_lm)
```

```
## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: avg_phon_sim
## Chisq Df Pr(>Chisq)
## Group 0.0132 1 0.9087
```

```
#getting avg frequency by Group x dim x struct
animals_freq_comparison = full_animals_lexical %>% group_by(Subject,Group,dimension,struct) %>%
  summarise(avg_freq = mean(Frequency_Value),
            items = n())
```

```
## 'summarise()' has grouped output by 'Subject', 'Group', 'dimension'. You can
## override using the '.groups' argument.
```

```
#Does avg frequency differ by group
animals_freq_lm = lmer(data = animals_freq_comparison, avg_freq ~ Group + (1|Subject))
```

```
## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, : Model failed to converge
## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, : Model is nearly unidentifiable
## - Rescale variables?
```

```
summary(animals_freq_lm)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: avg_freq ~ Group + (1 | Subject)
## Data: animals_freq_comparison
##
## REML criterion at convergence: -13622.2
##
## Scaled residuals:
```

```
##           Min           1Q           Median           3Q           Max
## -1.220e-06 -2.757e-07 -1.109e-07  2.635e-07  9.570e-07
##
## Random effects:
##   Groups   Name      Variance Std.Dev.
##   Subject (Intercept) 2.626e-03 5.124e-02
##   Residual              4.101e-15 6.404e-08
## Number of obs: 512, groups: Subject, 64
##
## Fixed effects:
##               Estimate Std. Error t value
## (Intercept)      3.953099   0.008976 440.417
## Groupcochlear implant 0.076430   0.012748   5.996
##
## Correlation of Fixed Effects:
##              (Intr)
## Grpcchlrmp -0.704
## optimizer (nloptwrap) convergence code: 0 (OK)
## Model failed to converge with max|grad| = 6.82623 (tol = 0.002, component 1)
## Model is nearly unidentifiable: very large eigenvalue
## - Rescale variables?
```

```
car::Anova(animals_freq_lm)
```

```
## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: avg_freq
##           Chisq Df Pr(>Chisq)
## Group 35.947  1  2.027e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
#FOODS
```

```
#USE
```

```
all_struct_foods<-rbind(full_foods_lexical,full_USE_lexical %>% filter(domain=="foods"))
```

```
#JUST W2V & S2V: avg semantic similarity by Group x dim x struct
```

```
foods_sem_sim_comparison = full_foods_lexical %>% group_by(Subject,Group,dimension,struct) %>%
  summarise(avg_sem_sim = mean(Semantic_Similarity),
            items = n())
```

```
## 'summarise()' has grouped output by 'Subject', 'Group', 'dimension'. You can
## override using the '.groups' argument.
```

```
#Incl USE: avg semantic similarity by Group x dim x struct
```

```
foods_USE_sem_sim_comparison = all_struct_foods %>% group_by(Subject,Group,dimension,struct) %>%
  summarise(avg_sem_sim = mean(Semantic_Similarity),
            items = n())
```

```
## 'summarise()' has grouped output by 'Subject', 'Group', 'dimension'. You can
## override using the '.groups' argument.
```

```
#JUST W2V & S2V: Does avg semantic similarity differ by group x struct x dim
foods_sem_sim_lm = lmer(data = foods_sem_sim_comparison, avg_sem_sim ~ Group*struct*dimension+(1|Subject)
summary(foods_sem_sim_lm)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: avg_sem_sim ~ Group * struct * dimension + (1 | Subject)
## Data: foods_sem_sim_comparison
##
## REML criterion at convergence: -1532.2
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.82140 -0.55670 -0.02377  0.52926  2.69746
##
## Random effects:
## Groups Name Variance Std.Dev.
## Subject (Intercept) 0.0036041 0.06003
## Residual 0.0002983 0.01727
## Number of obs: 352, groups: Subject, 44
##
## Fixed effects:
##
## Estimate Std. Error t value
## (Intercept) 0.5215254 0.0124938 41.743
## Groupcochlear implant -0.0124880 0.0190128 -0.657
## structw2v -0.0032750 0.0048847 -0.670
## dimension200 -0.0680188 0.0048847 -13.925
## dimension300 -0.1003759 0.0048847 -20.549
## dimension50 0.0702441 0.0048847 14.380
## Groupcochlear implant:structw2v -0.0002009 0.0074335 -0.027
## Groupcochlear implant:dimension200 0.0045981 0.0074335 0.619
## Groupcochlear implant:dimension300 0.0025229 0.0074335 0.339
## Groupcochlear implant:dimension50 0.0016551 0.0074335 0.223
## structw2v:dimension200 -0.0189874 0.0069081 -2.749
## structw2v:dimension300 -0.0312459 0.0069081 -4.523
## structw2v:dimension50 0.0198303 0.0069081 2.871
## Groupcochlear implant:structw2v:dimension200 -0.0047907 0.0105125 -0.456
## Groupcochlear implant:structw2v:dimension300 -0.0032098 0.0105125 -0.305
## Groupcochlear implant:structw2v:dimension50 -0.0046439 0.0105125 -0.442
##
## Correlation matrix not shown by default, as p = 16 > 12.
## Use print(x, correlation=TRUE) or
## vcov(x) if you need it
```

```
car::Anova(foods_sem_sim_lm)
```

```
## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: avg_sem_sim
## Chisq Df Pr(>Chisq)
## Group 0.4251 1 0.5144
## struct 44.8377 1 2.141e-11 ***
```

```
## dimension          6624.5394  3  < 2.2e-16 ***
## Group:struct        0.8182  1    0.3657
## Group:dimension     0.3355  3    0.9532
## struct:dimension    110.9663  3  < 2.2e-16 ***
## Group:struct:dimension 0.2687  3    0.9658
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

*#INCL USE: Does avg semantic similarity differ by group x struct*

```
foods_USE_sem_sim_lm = lmer(data = foods_USE_sem_sim_comparison, avg_sem_sim ~ Group*struct+(1|Subject))
summary(foods_USE_sem_sim_lm)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: avg_sem_sim ~ Group * struct + (1 | Subject)
## Data: foods_USE_sem_sim_comparison
##
## REML criterion at convergence: -778.4
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -1.7800 -0.7924 -0.0297  0.6651  2.1418
##
## Random effects:
## Groups Name Variance Std.Dev.
## Subject (Intercept) 0.002210 0.04701
## Residual 0.006427 0.08017
## Number of obs: 396, groups: Subject, 44
##
## Fixed effects:
##              Estimate Std. Error t value
## (Intercept) 0.496988 0.012356 40.221
## Groupcochlear implant -0.010294 0.018804 -0.547
## structUSE -0.211258 0.017927 -11.784
## structw2v -0.010876 0.011338 -0.959
## Groupcochlear implant:structUSE 0.004588 0.027281 0.168
## Groupcochlear implant:structw2v -0.003362 0.017254 -0.195
##
## Correlation of Fixed Effects:
##              (Intr) Grpcci strUSE strct2 Gi:USE
## Grpcchlrimp -0.657
## structUSE -0.290 0.191
## structw2v -0.459 0.301 0.316
## Gimplnt:USE 0.191 -0.290 -0.657 -0.208
## Gimplnt:st2 0.301 -0.459 -0.208 -0.657 0.316
```

```
car::Anova(foods_USE_sem_sim_lm)
```

```
## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: avg_sem_sim
##              Chisq Df Pr(>Chisq)
## Group      0.4696  1    0.4932
## struct    253.1164  2    <2e-16 ***
```

```
## Group:struct    0.0966  2    0.9528
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
emmeans(foods_USE_sem_sim_lm, pairwise ~ Group * struct, simple = "struct")
```

```
## $emmeans
##      Group      struct emmean      SE    df lower.CL upper.CL
## normal hearing  s2v      0.497 0.0124  70.8    0.472    0.522
## cochlear implant s2v      0.487 0.0142  70.8    0.458    0.515
## normal hearing  USE      0.286 0.0186 250.9    0.249    0.322
## cochlear implant USE      0.280 0.0213 250.9    0.238    0.322
## normal hearing  w2v      0.486 0.0124  70.8    0.461    0.511
## cochlear implant w2v      0.472 0.0142  70.8    0.444    0.501
##
## Degrees-of-freedom method: kenward-roger
## Confidence level used: 0.95
##
## $contrasts
## Group = normal hearing:
## contrast estimate      SE  df t.ratio p.value
## s2v - USE    0.2113 0.0179 348  11.784 <.0001
## s2v - w2v    0.0109 0.0113 348   0.959 0.6032
## USE - w2v   -0.2004 0.0179 348 -11.178 <.0001
##
## Group = cochlear implant:
## contrast estimate      SE  df t.ratio p.value
## s2v - USE    0.2067 0.0206 348  10.050 <.0001
## s2v - w2v    0.0142 0.0130 348   1.095 0.5180
## USE - w2v   -0.1924 0.0206 348  -9.358 <.0001
##
## Degrees-of-freedom method: kenward-roger
## P value adjustment: tukey method for comparing a family of 3 estimates
```

```
#avg phonological similarity by Group x dim x struct
foods_phon_sim_comparison = full_foods_lexical %>% group_by(Subject, Group, dimension, struct) %>%
  summarise(avg_phon_sim = mean(Phonological_Similarity),
            items = n())
```

```
## 'summarise()' has grouped output by 'Subject', 'Group', 'dimension'. You can
## override using the '.groups' argument.
```

```
#Does avg phon similarity differ by group
foods_phon_sim_lm = lmer(data = foods_phon_sim_comparison, avg_phon_sim ~ Group + (1|Subject))
```

```
## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, : Model failed to converge
## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, : Model is nearly unidentifiable:
## - Rescale variables?
```

```
summary(foods_phon_sim_lm)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: avg_phon_sim ~ Group + (1 | Subject)
## Data: foods_phon_sim_comparison
##
## REML criterion at convergence: -11184.2
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -3.428e-07 -4.797e-08  3.084e-08  9.338e-08  2.741e-07
##
## Random effects:
## Groups Name Variance Std.Dev.
## Subject (Intercept) 1.45e-04 1.204e-02
## Residual 1.64e-17 4.050e-09
## Number of obs: 352, groups: Subject, 44
##
## Fixed effects:
##              Estimate Std. Error t value
## (Intercept) 0.089728 0.002390 37.538
## Groupcochlear implant 0.006863 0.003634 1.888
##
## Correlation of Fixed Effects:
##              (Intr)
## Grpcchlrmp -0.658
## optimizer (nloptwrap) convergence code: 0 (OK)
## Model failed to converge with max|grad| = 1.04014 (tol = 0.002, component 1)
## Model is nearly unidentifiable: very large eigenvalue
## - Rescale variables?
```

```
car::Anova(foods_phon_sim_lm)
```

```
## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: avg_phon_sim
##      Chisq Df Pr(>Chisq)
## Group 3.5662 1 0.05897 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
#avg frequency by Group x dim x struct
foods_freq_comparison = full_foods_lexical %>% group_by(Subject,Group,dimension,struct) %>%
  summarise(avg_freq = mean(Frequency_Value),
            items = n())
```

```
## 'summarise()' has grouped output by 'Subject', 'Group', 'dimension'. You can
## override using the '.groups' argument.
```

```
#Does avg frequency differ by group
foods_freq_lm = lmer(data = foods_freq_comparison, avg_freq ~ Group + (1|Subject))
```

```
## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, : Model failed to converge
## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, : Model is nearly unidentifiable:
## - Rescale variables?
```

```
summary(foods_freq_lm)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: avg_freq ~ Group + (1 | Subject)
## Data: foods_freq_comparison
##
## REML criterion at convergence: -9632
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -3.583e-07 -1.520e-07 -2.171e-08  1.737e-07  1.107e-06
##
## Random effects:
## Groups Name Variance Std.Dev.
## Subject (Intercept) 3.033e-03 5.508e-02
## Residual 1.673e-15 4.091e-08
## Number of obs: 352, groups: Subject, 44
##
## Fixed effects:
##              Estimate Std. Error t value
## (Intercept)      4.073922   0.010995 370.528
## Groupcochlear implant 0.001959   0.016746   0.117
##
## Correlation of Fixed Effects:
##              (Intr)
## Grpcchlrmp -0.657
## optimizer (nloptwrap) convergence code: 0 (OK)
## Model failed to converge with max|grad| = 0.817777 (tol = 0.002, component 1)
## Model is nearly unidentifiable: very large eigenvalue
## - Rescale variables?
```

```
car::Anova(foods_freq_lm)
```

```
## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: avg_freq
##      Chisq Df Pr(>Chisq)
## Group 0.0137 1 0.9069
```

```
#COMBINED DOMAINS
```

```
full_comb_lexical<-rbind(full_animals_lexical,full_foods_lexical,full_USE_lexical)
```

```
#Just S2V & W2V
```

```
comb_sem_sim_comparison <- full_comb_lexical %>% filter(!struct=="USE")%>%group_by(Subject,Group,domain)
  summarise(avg_sem_sim = mean(Semantic_Similarity),
            items = n())
```

```
## 'summarise()' has grouped output by 'Subject', 'Group', 'domain', 'dimension'.
## You can override using the '.groups' argument.
```

```
#Incl USE
```

```
comb_USE_sem_sim_comparison <- full_comb_lexical%>%group_by(Subject,Group,domain,dimension,struct) %>%
  summarise(avg_sem_sim = mean(Semantic_Similarity),
            items = n())
```

```
## 'summarise()' has grouped output by 'Subject', 'Group', 'domain', 'dimension'.
## You can override using the '.groups' argument.
```

```
#JUST W2V/S2V:Does avg semantic similarity differ by group x domain x struct x dim
```

```
comb_sem_sim_lm = lmer(data = comb_sem_sim_comparison, avg_sem_sim ~ Group*dimension*struct*domain+(1|S
summary(comb_sem_sim_lm)
```

```
## Linear mixed model fit by REML ['lmerMod']
```

```
## Formula: avg_sem_sim ~ Group * dimension * struct * domain + (1 | Subject)
```

```
## Data: comb_sem_sim_comparison
```

```
##
```

```
## REML criterion at convergence: -3139
```

```
##
```

```
## Scaled residuals:
```

```
##      Min       1Q   Median       3Q      Max
## -4.2625 -0.4677 -0.0234  0.4656  3.8757
```

```
##
```

```
## Random effects:
```

```
## Groups   Name                Variance Std.Dev.
```

```
## Subject (Intercept) 0.0011404 0.03377
```

```
## Residual              0.0009643 0.03105
```

```
## Number of obs: 864, groups: Subject, 64
```

```
##
```

```
## Fixed effects:
```

```
##
```

	Estimate	Std. Error
(Intercept)	0.460032	0.008110
Groupcochlear implant	0.012661	0.011469
dimension200	-0.082877	0.007763
dimension300	-0.124188	0.007763
dimension50	0.085877	0.007763
structw2v	-0.009052	0.007763
domainfoods	0.062549	0.008346
Groupcochlear implant:dimension200	-0.005085	0.010979
Groupcochlear implant:dimension300	-0.006762	0.010979
Groupcochlear implant:dimension50	0.005719	0.010979
Groupcochlear implant:structw2v	0.003569	0.010979
dimension200:structw2v	-0.007812	0.010979
dimension300:structw2v	-0.012417	0.010979
dimension50:structw2v	0.023098	0.010979
Groupcochlear implant:domainfoods	-0.029622	0.012364
dimension200:domainfoods	0.014858	0.011722
dimension300:domainfoods	0.023812	0.011722
dimension50:domainfoods	-0.015633	0.011722
structw2v:domainfoods	0.005777	0.011722
Groupcochlear implant:dimension200:structw2v	-0.001353	0.015526



```

## Groupcochlear implant:dimension300:structw2v      0.001209    0.015526
## Groupcochlear implant:dimension50:structw2v      -0.006835    0.015526
## Groupcochlear implant:dimension200:domainfoods    0.009683    0.017297
## Groupcochlear implant:dimension300:domainfoods    0.009285    0.017297
## Groupcochlear implant:dimension50:domainfoods    -0.004064    0.017297
## Groupcochlear implant:structw2v:domainfoods      -0.003770    0.017297
## dimension200:structw2v:domainfoods                -0.011176    0.016578
## dimension300:structw2v:domainfoods                -0.018829    0.016578
## dimension50:structw2v:domainfoods                -0.003268    0.016578
## Groupcochlear implant:dimension200:structw2v:domainfoods -0.003437    0.024461
## Groupcochlear implant:dimension300:structw2v:domainfoods -0.004419    0.024461
## Groupcochlear implant:dimension50:structw2v:domainfoods  0.002191    0.024461
##                                                    t value
## (Intercept)                                     56.725
## Groupcochlear implant                           1.104
## dimension200                                    -10.676
## dimension300                                    -15.997
## dimension50                                      11.062
## structw2v                                       -1.166
## domainfoods                                     7.494
## Groupcochlear implant:dimension200              -0.463
## Groupcochlear implant:dimension300              -0.616
## Groupcochlear implant:dimension50                0.521
## Groupcochlear implant:structw2v                 0.325
## dimension200:structw2v                          -0.712
## dimension300:structw2v                          -1.131
## dimension50:structw2v                           2.104
## Groupcochlear implant:domainfoods               -2.396
## dimension200:domainfoods                        1.268
## dimension300:domainfoods                        2.031
## dimension50:domainfoods                        -1.334
## structw2v:domainfoods                          0.493
## Groupcochlear implant:dimension200:structw2v    -0.087
## Groupcochlear implant:dimension300:structw2v     0.078
## Groupcochlear implant:dimension50:structw2v     -0.440
## Groupcochlear implant:dimension200:domainfoods   0.560
## Groupcochlear implant:dimension300:domainfoods   0.537
## Groupcochlear implant:dimension50:domainfoods   -0.235
## Groupcochlear implant:structw2v:domainfoods     -0.218
## dimension200:structw2v:domainfoods               -0.674
## dimension300:structw2v:domainfoods               -1.136
## dimension50:structw2v:domainfoods                -0.197
## Groupcochlear implant:dimension200:structw2v:domainfoods -0.141
## Groupcochlear implant:dimension300:structw2v:domainfoods -0.181
## Groupcochlear implant:dimension50:structw2v:domainfoods  0.090

##
## Correlation matrix not shown by default, as p = 32 > 12.
## Use print(x, correlation=TRUE) or
##      vcov(x)      if you need it

```

```
car::Anova(comb_sem_sim_lm)
```

```
## Analysis of Deviance Table (Type II Wald chisquare tests)
```

```
##
## Response: avg_sem_sim
##
##           Chisq Df Pr(>Chisq)
## Group           0.0681 1      0.7941
## dimension       6212.7418 3 < 2.2e-16 ***
## struct          19.8817 1  8.239e-06 ***
## domain          553.5079 1 < 2.2e-16 ***
## Group:dimension  0.7238 3      0.8676
## Group:struct     0.0039 1      0.9501
## dimension:struct  50.3515 3  6.724e-11 ***
## Group:domain     36.9207 1  1.230e-09 ***
## dimension:domain  42.5294 3  3.098e-09 ***
## struct:domain     1.3047 1      0.2533
## Group:dimension:struct  0.3030 3      0.9595
## Group:dimension:domain  1.1480 3      0.7655
## Group:struct:domain  0.3596 1      0.5488
## dimension:struct:domain  3.7837 3      0.2858
## Group:dimension:struct:domain  0.0940 3      0.9925
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
emmeans(comb_sem_sim_lm,pairwise~dimension*struct,simple="struct")
```

```
## NOTE: Results may be misleading due to involvement in interactions
```

```
## $emmeans
##   dimension struct emmean      SE df lower.CL upper.CL
##   100          s2v   0.490 0.00523 126    0.480    0.501
##   200          s2v   0.415 0.00523 126    0.404    0.425
##   300          s2v   0.377 0.00523 126    0.367    0.387
##   50           s2v   0.570 0.00523 126    0.560    0.580
##   100          w2v   0.485 0.00523 126    0.475    0.495
##   200          w2v   0.394 0.00523 126    0.384    0.405
##   300          w2v   0.349 0.00523 126    0.339    0.360
##   50           w2v   0.583 0.00523 126    0.573    0.594
##
## Results are averaged over the levels of: Group, domain
## Degrees-of-freedom method: kenward-roger
## Confidence level used: 0.95
##
## $contrasts
## dimension = 100:
##   contrast estimate      SE df t.ratio p.value
##   s2v - w2v  0.00532 0.00432 770    1.231  0.2188
##
## dimension = 200:
##   contrast estimate      SE df t.ratio p.value
##   s2v - w2v  0.02026 0.00432 770    4.685 <.0001
##
## dimension = 300:
##   contrast estimate      SE df t.ratio p.value
##   s2v - w2v  0.02765 0.00432 770    6.395 <.0001
##
```

```

## dimension = 50:
## contrast estimate SE df t.ratio p.value
## s2v - w2v -0.01327 0.00432 770 -3.069 0.0022
##
## Results are averaged over the levels of: Group, domain
## Degrees-of-freedom method: kenward-roger

#INCL USE: Does avg semantic similarity differ by group x domain x struct
comb_USE_sem_sim_lm = lmer(data = comb_USE_sem_sim_comparison, avg_sem_sim ~ Group*struct*domain+(1|Subj
summary(comb_USE_sem_sim_lm)

## Linear mixed model fit by REML ['lmerMod']
## Formula: avg_sem_sim ~ Group * struct * domain + (1 | Subject)
## Data: comb_USE_sem_sim_comparison
##
## REML criterion at convergence: -1845.9
##
## Scaled residuals:
## Min 1Q Median 3Q Max
## -2.57707 -0.78722 -0.06973 0.65840 2.33096
##
## Random effects:
## Groups Name Variance Std.Dev.
## Subject (Intercept) 0.0005595 0.02365
## Residual 0.0077507 0.08804
## Number of obs: 972, groups: Subject, 64
##
## Fixed effects:
## Estimate Std. Error t value
## (Intercept) 0.429736 0.008834 48.647
## Groupcochlear implant 0.011129 0.012493 0.891
## structUSE -0.192713 0.017400 -11.075
## structw2v -0.008335 0.011005 -0.757
## domainfoods 0.067541 0.011875 5.687
## Groupcochlear implant:structUSE -0.027482 0.024607 -1.117
## Groupcochlear implant:structw2v 0.001824 0.015563 0.117
## Groupcochlear implant:domainfoods -0.023141 0.017630 -1.313
## structUSE:domainfoods -0.018544 0.026273 -0.706
## structw2v:domainfoods -0.002541 0.016617 -0.153
## Groupcochlear implant:structUSE:domainfoods 0.032070 0.038768 0.827
## Groupcochlear implant:structw2v:domainfoods -0.005186 0.024519 -0.212
##
## Correlation of Fixed Effects:
## (Intr) Grpcci strUSE strct2 dmnfds Gim:USE Gimp:2 Gimpl: stUSE:
## Grpcchlrimp -0.707
## structUSE -0.394 0.279
## structw2v -0.623 0.440 0.316
## domainfoods -0.577 0.408 0.293 0.463
## Gimplnt:USE 0.279 -0.394 -0.707 -0.224 -0.207
## Gimplnt:st2 0.440 -0.623 -0.224 -0.707 -0.328 0.316
## Grpcimplnt: 0.389 -0.550 -0.197 -0.312 -0.674 0.279 0.441
## strctUSE:dm 0.261 -0.184 -0.662 -0.209 -0.442 0.468 0.148 0.298
## strctw2v:dm 0.413 -0.292 -0.209 -0.662 -0.700 0.148 0.468 0.471 0.316
## Gimpln:USE: -0.177 0.250 0.449 0.142 0.300 -0.635 -0.201 -0.440 -0.678

```

```
## Gimplnt:s2: -0.280  0.395  0.142  0.449  0.474 -0.201  -0.635 -0.695 -0.214
##          strc2: Gi:USE:
## Grpcchlrmp
## structUSE
## structw2v
## domainfoods
## Gimplnt:USE
## Gimplnt:st2
## Grpcimplnt:
## strctUSE:dm
## strctw2v:dm
## Gimpln:USE: -0.214
## Gimplnt:s2: -0.678  0.316
```

```
car::Anova(comb_USE_sem_sim_lm)
```

```
## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: avg_sem_sim
##              Chisq Df Pr(>Chisq)
## Group          0.0071  1    0.93300
## struct        512.3543  2    < 2e-16 ***
## domain        83.7367  1    < 2e-16 ***
## Group:struct    0.6403  2    0.72606
## Group:domain    3.3295  1    0.06805 .
## struct:domain   0.1678  2    0.91953
## Group:struct:domain 0.9330  2    0.62720
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
emmeans(comb_USE_sem_sim_lm, pairwise ~ Group * struct, simple = "struct")
```

```
## NOTE: Results may be misleading due to involvement in interactions
```

```
## $emmeans
##      Group      struct emmean      SE df lower.CL upper.CL
## normal hearing  s2v      0.464 0.00727 140    0.449    0.478
## cochlear implant s2v      0.463 0.00775 167    0.448    0.478
## normal hearing   USE      0.262 0.01250 678    0.237    0.286
## cochlear implant USE      0.250 0.01349 740    0.223    0.276
## normal hearing  w2v      0.454 0.00727 140    0.440    0.468
## cochlear implant w2v      0.453 0.00775 167    0.437    0.468
##
## Results are averaged over the levels of: domain
## Degrees-of-freedom method: kenward-roger
## Confidence level used: 0.95
##
## $contrasts
## Group = normal hearing:
## contrast estimate      SE df t.ratio p.value
## s2v - USE  0.20199 0.01314 899  15.376 <.0001
## s2v - w2v  0.00961 0.00831 899   1.156 0.4799
```

```
## USE - w2v -0.19238 0.01314 899 -14.644 <.0001
##
## Group = cochlear implant:
## contrast estimate SE df t.ratio p.value
## s2v - USE 0.21343 0.01425 899 14.974 <.0001
## s2v - w2v 0.01037 0.00901 899 1.151 0.4831
## USE - w2v -0.20306 0.01425 899 -14.246 <.0001
##
## Results are averaged over the levels of: domain
## Degrees-of-freedom method: kenward-roger
## P value adjustment: tukey method for comparing a family of 3 estimates
```

```
#avg phonological similarity
```

```
comb_phon_sim_comparison = full_comb_lexical %>% group_by(Subject,Group,domain,dimension,struct) %>%
  summarise(avg_phon_sim = mean(Phonological_Similarity),
            items = n())
```

```
## 'summarise()' has grouped output by 'Subject', 'Group', 'domain', 'dimension'.
## You can override using the '.groups' argument.
```

```
#Does phonological similarity differ by group
```

```
comb_phon_sim_lm = lmer(data = comb_phon_sim_comparison, avg_phon_sim ~ Group*domain + (1|Subject))
summary(comb_phon_sim_lm)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: avg_phon_sim ~ Group * domain + (1 | Subject)
## Data: comb_phon_sim_comparison
##
## REML criterion at convergence: -4449
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -4.0955 -0.6098 -0.0405  0.6437  2.5459
##
## Random effects:
## Groups Name Variance Std.Dev.
## Subject (Intercept) 0.0005469 0.02339
## Residual 0.0004810 0.02193
## Number of obs: 972, groups: Subject, 64
##
## Fixed effects:
##              Estimate Std. Error t value
## (Intercept) 0.069888 0.004331 16.136
## Groupcochlear implant 0.002918 0.006125 0.476
## domainfoods 0.024113 0.002058 11.718
## Groupcochlear implant:domainfoods -0.003558 0.003124 -1.139
##
## Correlation of Fixed Effects:
##              (Intr) Grpccci dmnfds
## Grpcchlrmp -0.707
## domainfoods -0.187 0.133
## Grpcimplnt 0.123 -0.175 -0.659
```

```
car::Anova(comb_phon_sim_lm)
```

```
## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: avg_phon_sim
##           Chisq Df Pr(>Chisq)
## Group       0.0795  1    0.7780
## domain     212.5034  1    <2e-16 ***
## Group:domain  1.2972  1    0.2547
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
#avg frequency by Group x dim x struct
```

```
comb_freq_comparison = full_comb_lexical %>% group_by(Subject,Group,domain,dimension,struct) %>%
  summarise(avg_freq = mean(Frequency_Value),
            items = n())
```

```
## 'summarise()' has grouped output by 'Subject', 'Group', 'domain', 'dimension'.
## You can override using the '.groups' argument.
```

```
#Does avg frequency differ by group
```

```
comb_freq_lm = lmer(data = comb_freq_comparison, avg_freq ~ Group*domain+(1|Subject))
summary(comb_freq_lm)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: avg_freq ~ Group * domain + (1 | Subject)
## Data: comb_freq_comparison
##
## REML criterion at convergence: -1604
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -3.8114 -0.5312 -0.0180  0.5304  3.9805
##
## Random effects:
##  Groups   Name                Variance Std.Dev.
## Subject (Intercept) 0.016705 0.12925
## Residual          0.008811 0.09387
## Number of obs: 972, groups: Subject, 64
##
## Fixed effects:
##
##              Estimate Std. Error t value
## (Intercept)      3.950462   0.023508 168.045
## Groupcochlear implant  0.058012   0.033246   1.745
## domainfoods       0.151181   0.008823  17.135
## Groupcochlear implant:domainfoods -0.071402   0.013407  -5.326
##
## Correlation of Fixed Effects:
##              (Intr) Grpcci dmnfds
## Grpcchlrmp -0.707
## domainfoods -0.148  0.104
## Grpcimplnt: 0.097 -0.137 -0.658
```

```
car::Anova(comb_freq_lm)
```

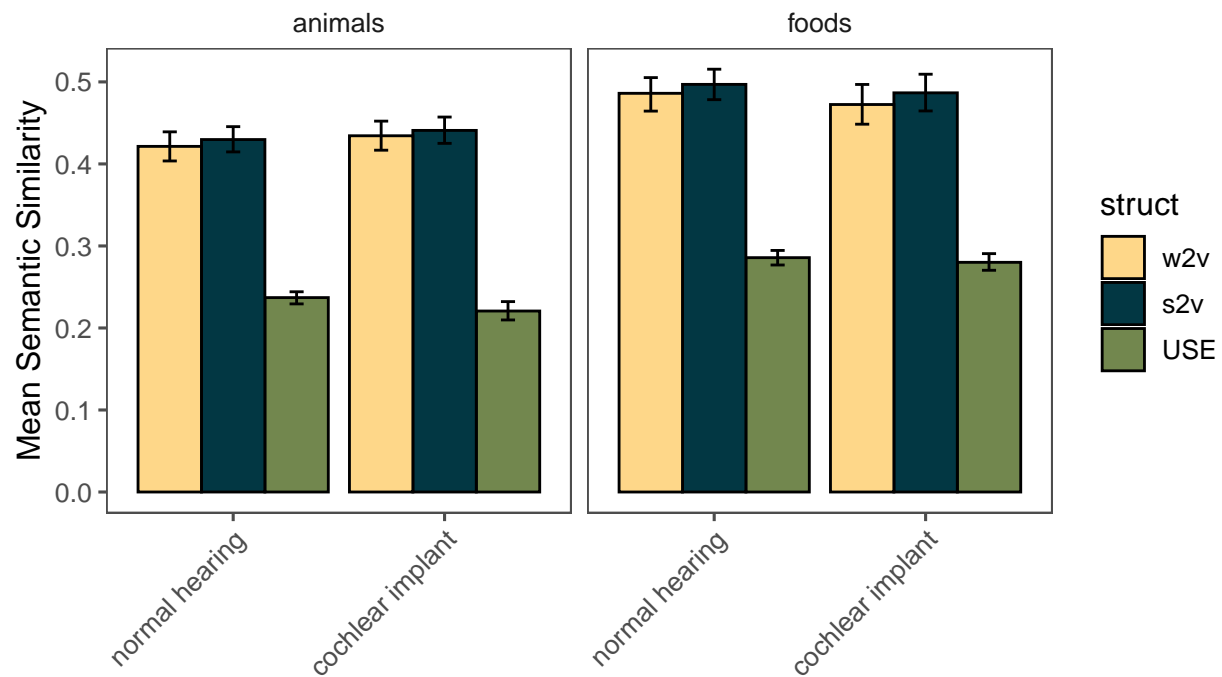
```
## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: avg_freq
##           Chisq Df Pr(>Chisq)
## Group      1.0476  1    0.3061
## domain    327.6927  1 < 2.2e-16 ***
## Group:domain 28.3642  1  1.005e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
emmeans(comb_freq_lm, pairwise~Group*domain, simple="Group")
```

```
## $emmeans
##   Group      domain emmean      SE   df lower.CL upper.CL
## normal hearing animals  3.95 0.0235 64.4      3.90      4.00
## cochlear implant animals 4.01 0.0235 64.4      3.96      4.06
## normal hearing  foods   4.10 0.0239 68.2      4.05      4.15
## cochlear implant foods   4.09 0.0244 73.9      4.04      4.14
##
## Degrees-of-freedom method: kenward-roger
## Confidence level used: 0.95
##
## $contrasts
## domain = animals:
## contrast              estimate      SE   df t.ratio p.value
## normal hearing - cochlear implant -0.0580 0.0332 64.4 -1.745  0.0858
##
## domain = foods:
## contrast              estimate      SE   df t.ratio p.value
## normal hearing - cochlear implant  0.0134 0.0341 71.0  0.393  0.6957
##
## Degrees-of-freedom method: kenward-roger
```

```
comb_USE_sem_sim_comparison %>%
  group_by(Group, domain, struct) %>%
  tidyboot_mean(avg_sem_sim, nboot = 1000, na.rm = TRUE) %>%
  mutate(struct = factor(struct, levels = c("w2v", "s2v", "USE"))) %>% # Set the order of 'struct'
  ggplot(aes(x = Group, y = empirical_stat, fill = struct)) +
    geom_bar(stat = 'identity', color = "black", position = position_dodge(width = 0.9)) +
    geom_errorbar(aes(ymin = ci_lower, ymax = ci_upper), width = 0.2, position = position_dodge(width = 0.9)) +
    labs(y = 'Mean Semantic Similarity', x = "") +
    theme_few() +
    theme(aspect.ratio = 1,
          legend.position = 'right',
          axis.text.x = element_text(angle = 45, hjust = 1)) +
    scale_fill_paletteer_d("nationalparkcolors::Acadia") +
    facet_wrap(~ domain)
```

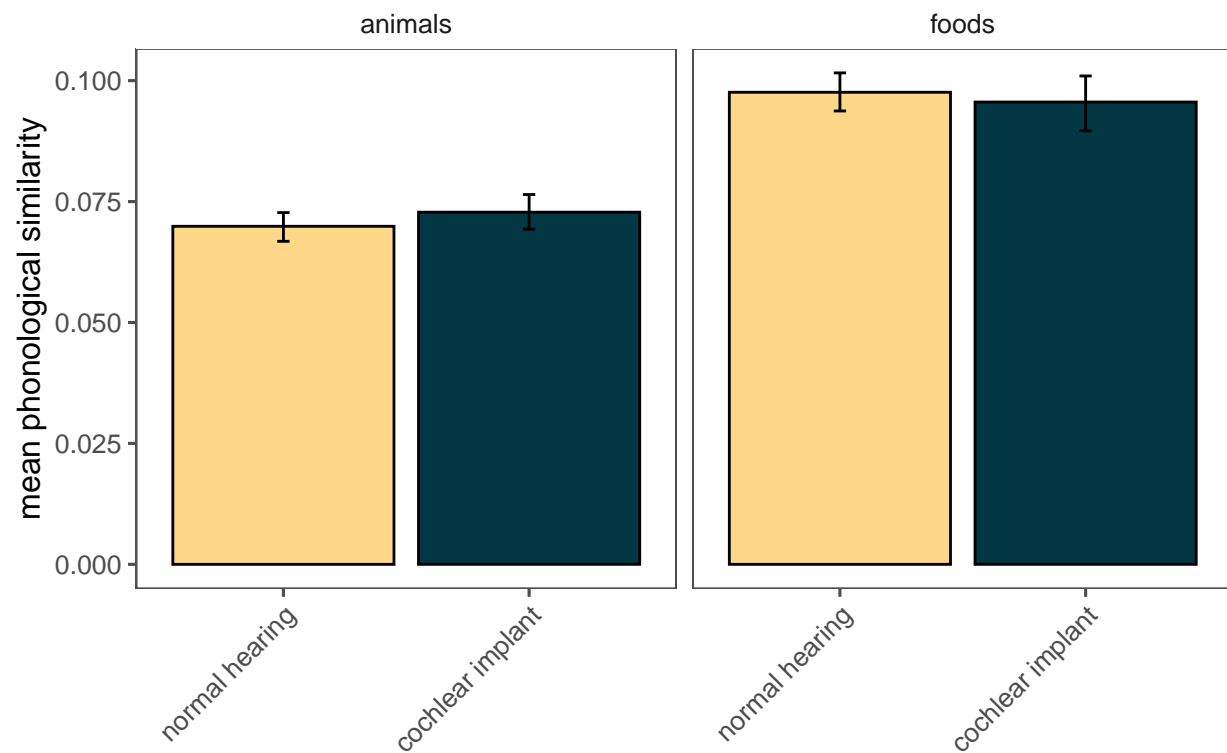
```
## Warning: 'cols' is now required when using 'unnest()'.
## i Please use 'cols = c(strap)'.
```



```
comb_phon_sim_comparison %>%
  group_by(Group, domain) %>% tidyboot_mean(avg_phon_sim, nboot = 1000, na.rm = T) %>%
  ggplot(aes(x = Group, y = empirical_stat, fill = Group)) +
    geom_bar(stat = 'identity', color = "black", position = position_dodge(width = 1)) +
    geom_errorbar(aes(ymin = ci_lower, ymax = ci_upper), width = 0.05) +
    labs(y = 'mean phonological similarity', x = "") +
    theme_few() +
    theme(aspect.ratio = 1, legend.position = 'none', axis.text.x = element_text(angle = 45, hjust = 1)) +
    facet_wrap(~domain) +
    scale_fill_paletteer_d("nationalparkcolors::Acadia")
```

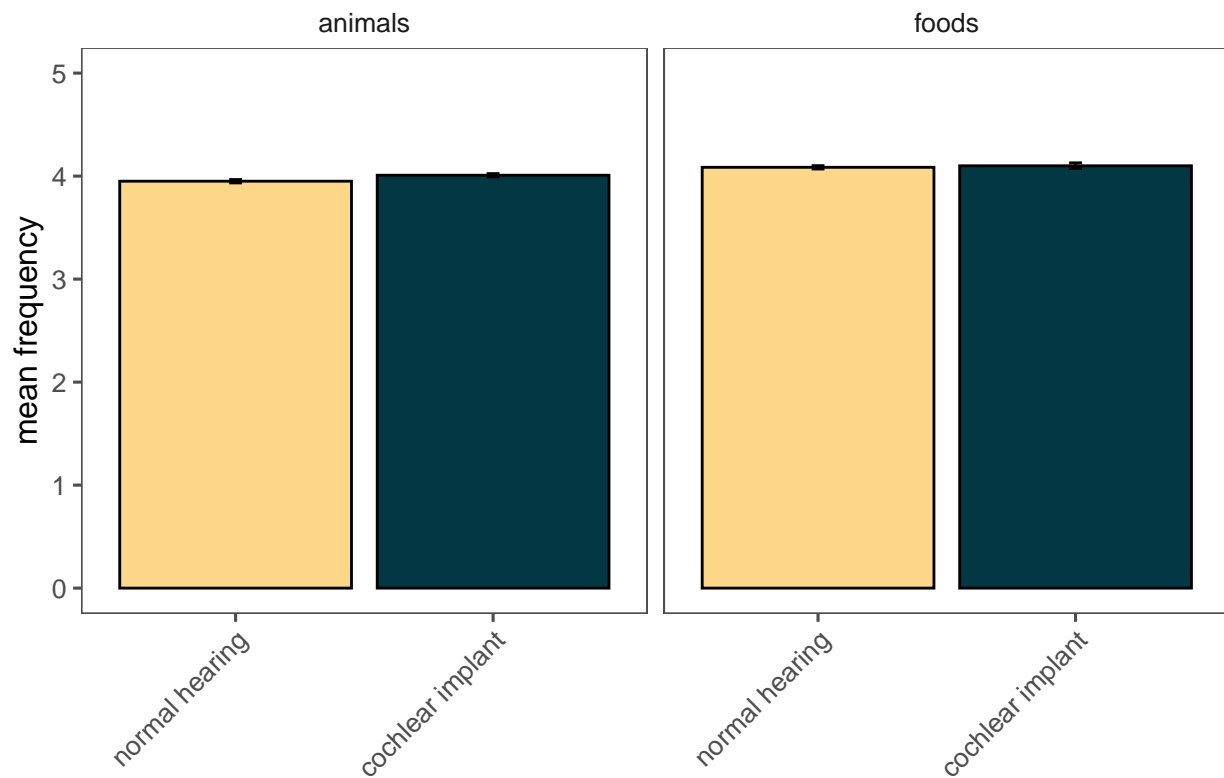
```
## Warning: 'cols' is now required when using 'unnest()'.
## i Please use 'cols = c(strap)'.
```



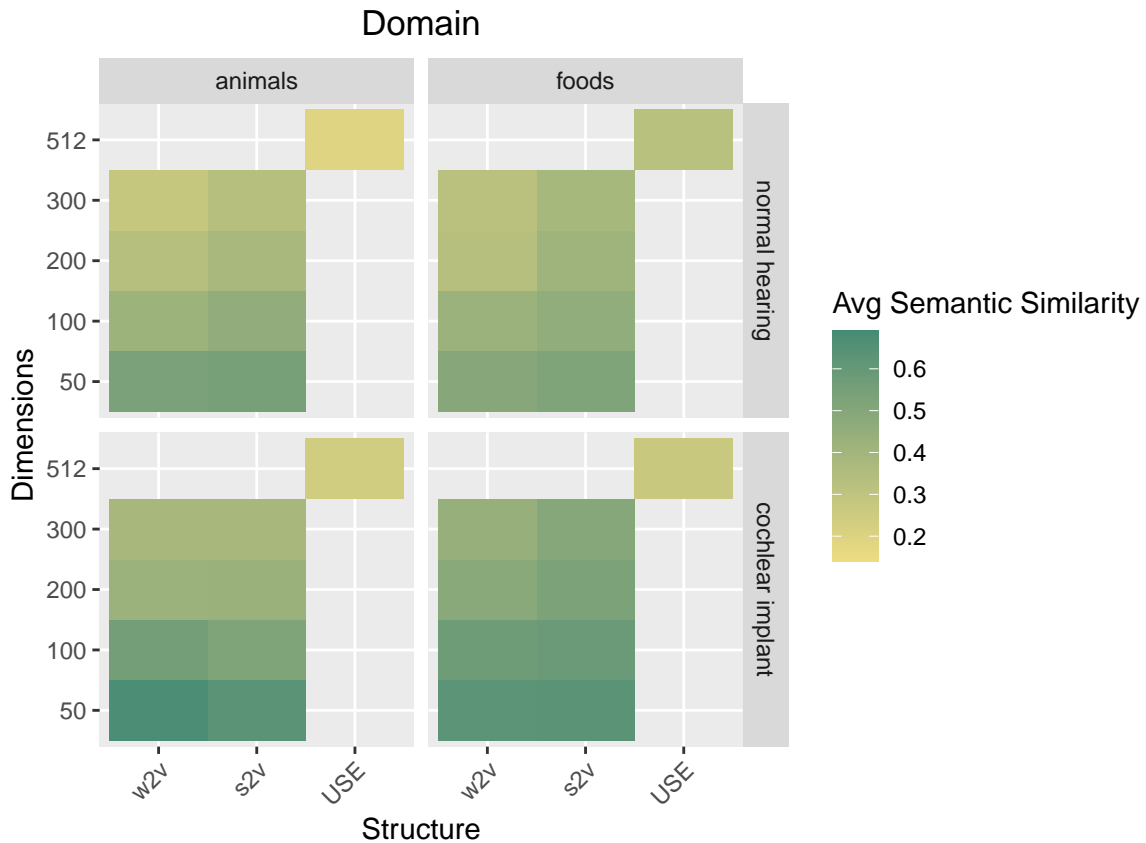


```
comb_freq_comparison %>%
  group_by(Group, domain) %>% tidyboot_mean(avg_freq, nboot = 1000, na.rm = T) %>%
  ggplot(aes(x = Group, y = empirical_stat, fill = Group)) +
    geom_bar(stat = 'identity', color = "black", position = position_dodge(width = 1)) + scale_y_continuous
    geom_errorbar(aes(ymin = ci_lower, ymax = ci_upper), width = 0.05) +
    labs(y = 'mean frequency', x = "") +
    theme_few() +
    theme(aspect.ratio = 1, legend.position = 'none', axis.text.x = element_text(angle = 45, hjust = 1))
    facet_wrap(~domain) +
    scale_fill_paletteer_d("nationalparkcolors::Acadia")
```

```
## Warning: 'cols' is now required when using 'unnest()'.
## i Please use 'cols = c(strap)'.
```



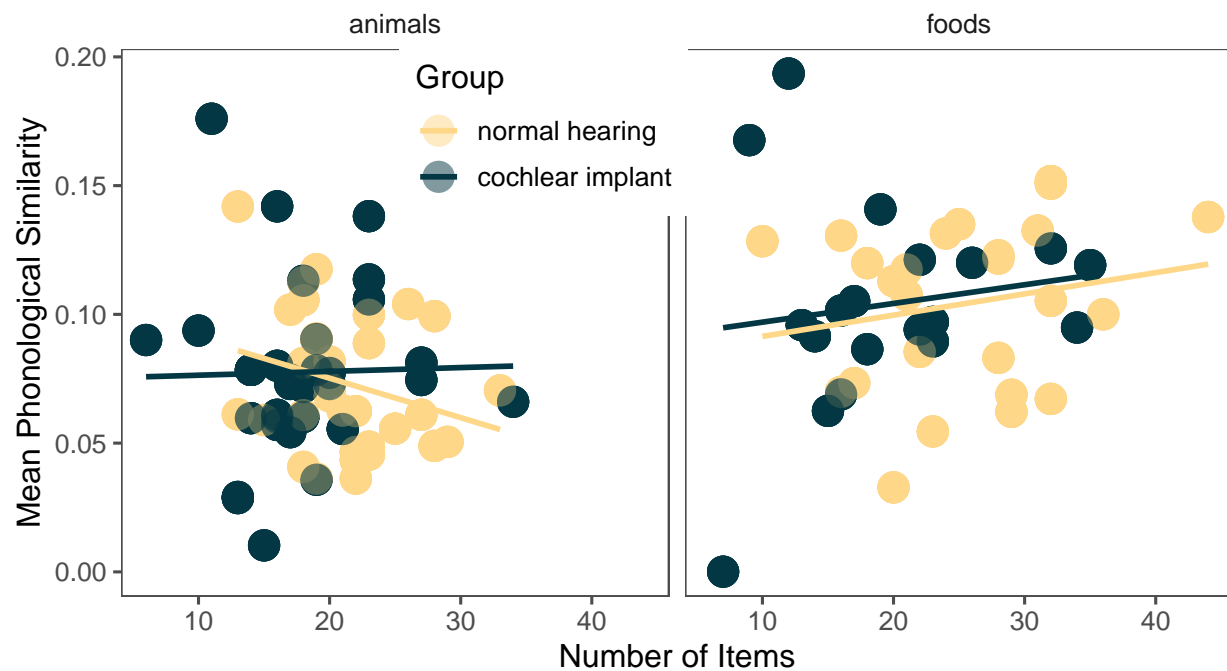
```
#Looking at similarity across structures/dims
desired_order_dim <- c("50", "100", "200", "300", "512") # Replace with your actual group names
desired_order_struct <- c("w2v", "s2v", "USE")
comp_USE_sem_sim_across <- comb_USE_sem_sim_comparison %>%
  mutate(
    dimension = factor(dimension, levels = desired_order_dim), # Ensure this is correctly set
    struct = factor(struct, levels = desired_order_struct)      # Set the structure order
  ) %>%
  ggplot(aes(x = struct, y = dimension, fill = avg_sem_sim)) +
  geom_tile() +
  scale_fill_gradient(low = "lightgoldenrod2", high = "aquamarine4", name = "Avg Semantic Similarity") +
  labs(x = "Structure", y = "Dimensions", title = "Domain") +
  theme(
    aspect.ratio = 1,
    plot.title = element_text(hjust = 0.5, size = rel(1.2)),
    axis.text.x = element_text(angle = 45, hjust = 1)
  ) +
  facet_grid(Group ~ domain)
comp_USE_sem_sim_across
```



#Lexical x Fluency Analyses

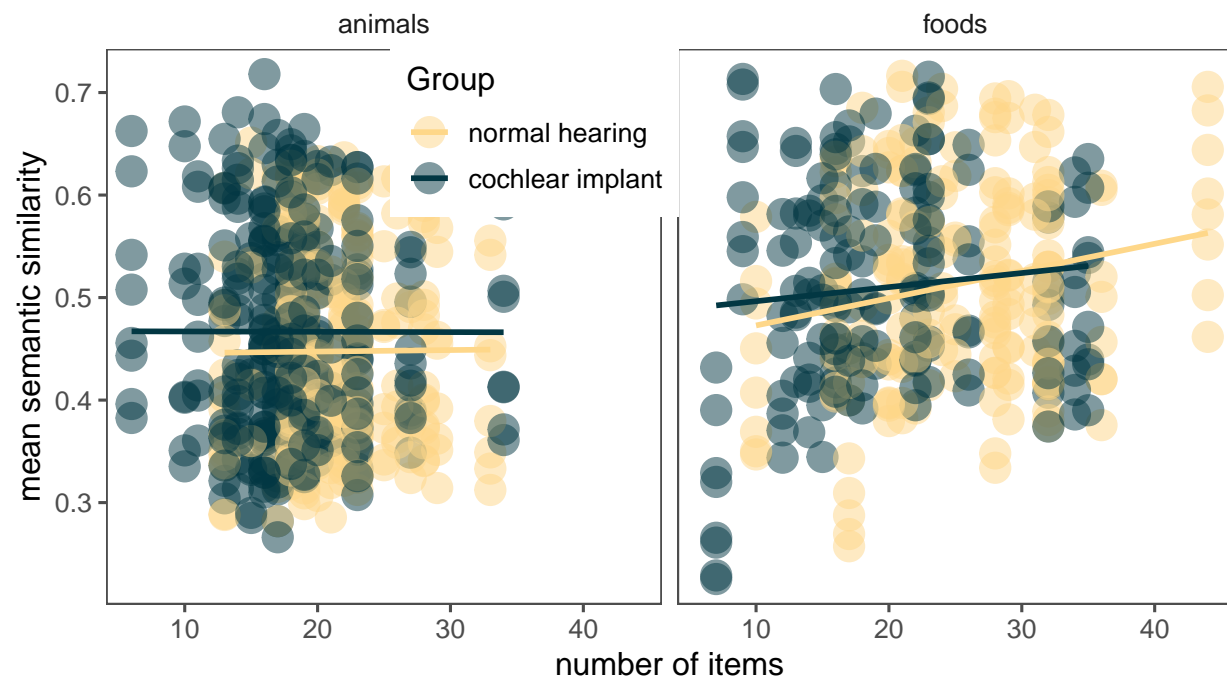
```
#Similarity by items produced, collapsed across structures and dimensions
comb_fluency %>%
  group_by(Group, domain) %>%
  ggplot(aes(y = Phonological_Similarity_mean, x = Numb_of_Items, group = Group, color = Group)) +
    geom_point(alpha = 0.5, size=5) +
    labs(x = "Number of Items", y = "Mean Phonological Similarity") +
    theme_few() +
    geom_smooth(method = "lm", se = FALSE) +
    theme(aspect.ratio = 1, legend.position = c(0.38, 0.85)) +
    facet_wrap(~domain) +
    scale_color_paletteer_d("nationalparkcolors::Acadia")
```

```
## 'geom_smooth()' using formula = 'y ~ x'
```



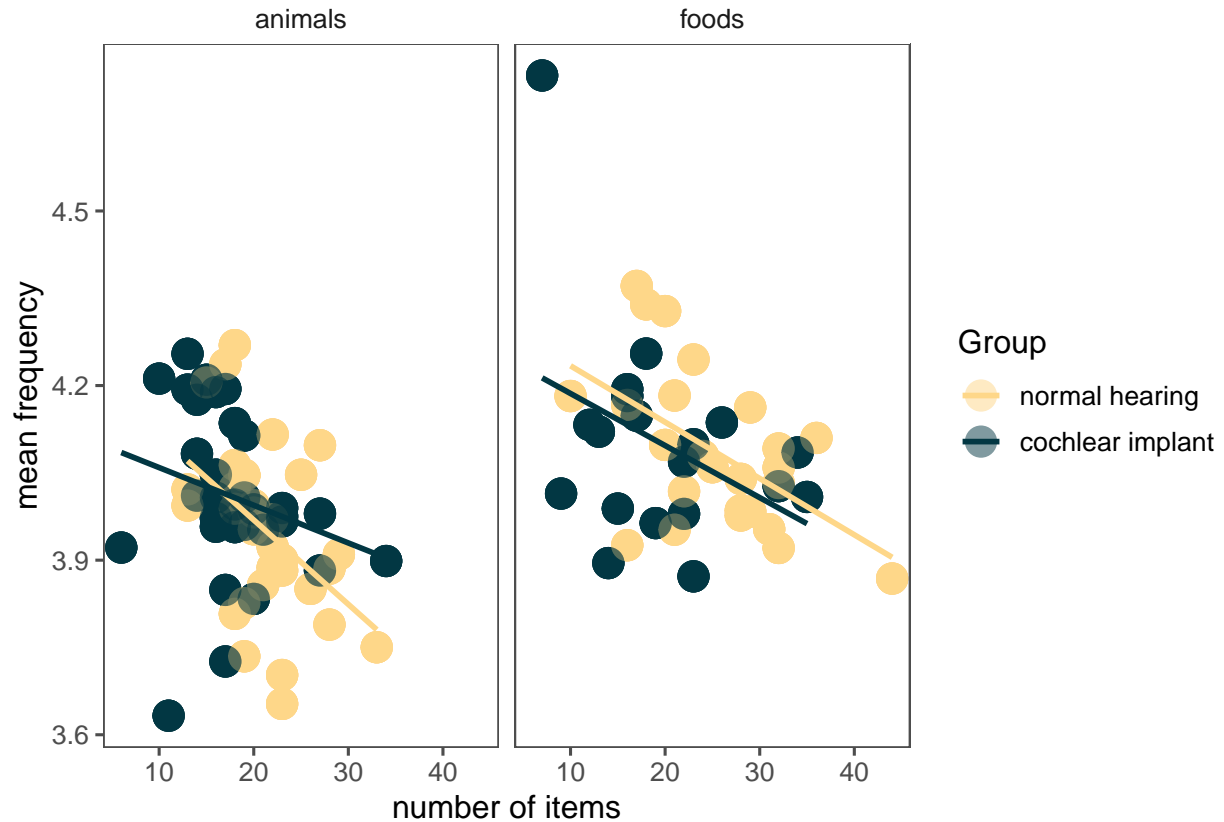
```
comb_fluency %>% group_by(Group, domain) %>%
  ggplot(aes(y = Semantic_Similarity_mean, x = Numb_of_Items, group = Group, color = Group)) +
  geom_point(alpha = 0.5, size = 5) +
  labs(x = "number of items", y = "mean semantic similarity") +
  theme_few() +
  geom_smooth(method = "lm", se = FALSE) +
  theme(aspect.ratio = 1, legend.position = c(.38, 0.85)) +
  facet_wrap(~domain) +
  scale_color_paletteer_d("nationalparkcolors::Acadia")
```

```
## 'geom_smooth()' using formula = 'y ~ x'
```



```
comb_fluency %>% group_by(Group, domain) %>%
  ggplot(aes(y = Frequency_Value_mean, x = Numb_of_Items, group = Group, color = Group)) +
  geom_point(alpha = 0.5, size = 5) +
  labs(x = "number of items", y = "mean frequency") +
  theme_few() +
  geom_smooth(method = "lm", se = FALSE) +
  facet_wrap(~domain) +
  scale_color_paletteer_d("nationalparkcolors::Acadia")
```

```
## 'geom_smooth()' using formula = 'y ~ x'
```



```
#Does mean phonological similarity x domain type predict numb items produced
comb_psim_model = lmer(data = comb_fluency,
  Numb_of_Items ~ Phonological_Similarity_mean*Group*domain + (1|Subject))
summary(comb_psim_model)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: Numb_of_Items ~ Phonological_Similarity_mean * Group * domain +
##      (1 | Subject)
##      Data: comb_fluency
##
## REML criterion at convergence: 4312.5
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.57941 -0.51313  0.01096  0.46342  2.64310
##
## Random effects:
##  Groups   Name                Variance Std.Dev.
##  Subject  (Intercept)  34.087    5.838
##  Residual                    6.564    2.562
## Number of obs: 864, groups: Subject, 64
##
## Fixed effects:
##                                     Estimate
## (Intercept)                        15.359
## Phonological_Similarity_mean        80.837
```

```
## Groupcochlear implant                2.532
## domainfoods                          10.067
## Phonological_Similarity_mean:Groupcochlear implant -81.859
## Phonological_Similarity_mean:domainfoods -74.485
## Groupcochlear implant:domainfoods -7.307
## Phonological_Similarity_mean:Groupcochlear implant:domainfoods 57.977
##                                     Std. Error
## (Intercept)                        1.258
## Phonological_Similarity_mean        9.579
## Groupcochlear implant               1.849
## domainfoods                        1.141
## Phonological_Similarity_mean:Groupcochlear implant 14.675
## Phonological_Similarity_mean:domainfoods 12.451
## Groupcochlear implant:domainfoods 1.861
## Phonological_Similarity_mean:Groupcochlear implant:domainfoods 20.330
##                                     t value
## (Intercept)                       12.207
## Phonological_Similarity_mean        8.439
## Groupcochlear implant               1.370
## domainfoods                        8.820
## Phonological_Similarity_mean:Groupcochlear implant -5.578
## Phonological_Similarity_mean:domainfoods -5.982
## Groupcochlear implant:domainfoods -3.927
## Phonological_Similarity_mean:Groupcochlear implant:domainfoods 2.852
##
## Correlation of Fixed Effects:
##          (Intr) Phn_S_ Grpcci dmnfds Ph_S_:Gi Ph_S_: Gimpl:
## Phnlgcl_Sm_ -0.558
## Grpcchlrmp -0.681  0.380
## domainfoods -0.379  0.647  0.258
## Phnlg_S_:Gi  0.364 -0.653 -0.601 -0.422
## Phnlgcl_S_  0.428 -0.767 -0.291 -0.964  0.501
## Grpcimplnt:  0.232 -0.397 -0.470 -0.613  0.754  0.591
## Phnl_S_:Gi: -0.262  0.470  0.505  0.590 -0.838 -0.612 -0.972
```

```
car::Anova(comb_psim_model)
```

```
## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: Numb_of_Items
##                                     Chisq Df Pr(>Chisq)
## Phonological_Similarity_mean        4.7689  1  0.028979 *
## Group                             10.5933  1  0.001135 **
## domain                          122.8260  1 < 2.2e-16 ***
## Phonological_Similarity_mean:Group   34.0950  1  5.249e-09 ***
## Phonological_Similarity_mean:domain  28.7112  1  8.402e-08 ***
## Group:domain                       23.9078  1  1.011e-06 ***
## Phonological_Similarity_mean:Group:domain  8.1331  1  0.004347 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
#Does mean semantic similarity x domain type predict numb items produced
comb_semsim_model = lmer(data = comb_fluency,
```

```

Numb_of_Items ~ Semantic_Similarity_mean*Group*domain + (1|Subject))
summary(comb_semsim_model)

```

```

## Linear mixed model fit by REML ['lmerMod']
## Formula: Numb_of_Items ~ Semantic_Similarity_mean * Group * domain + (1 |
##      Subject)
##      Data: comb_fluency
##
## REML criterion at convergence: 4384.6
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.42201 -0.46598  0.00134  0.46773  2.60712
##
## Random effects:
##      Groups      Name      Variance Std.Dev.
##      Subject (Intercept) 29.975   5.475
##      Residual              7.145   2.673
## Number of obs: 864, groups: Subject, 64
##
## Fixed effects:
##
##                                     Estimate Std. Error
## (Intercept)                        20.7684      1.2600
## Semantic_Similarity_mean             1.1459      1.7637
## Groupcochlear implant              -3.3601      1.7760
## domainfoods                         1.6314      1.3506
## Semantic_Similarity_mean:Groupcochlear implant -0.2798      2.4235
## Semantic_Similarity_mean:domainfoods         5.3745      2.7186
## Groupcochlear implant:domainfoods           1.8841      1.9738
## Semantic_Similarity_mean:Groupcochlear implant:domainfoods -10.3308      3.9250
##
##                                     t value
## (Intercept)                        16.483
## Semantic_Similarity_mean             0.650
## Groupcochlear implant              -1.892
## domainfoods                         1.208
## Semantic_Similarity_mean:Groupcochlear implant -0.115
## Semantic_Similarity_mean:domainfoods         1.977
## Groupcochlear implant:domainfoods           0.955
## Semantic_Similarity_mean:Groupcochlear implant:domainfoods -2.632
##
## Correlation of Fixed Effects:
##      (Intr) Smn_S_ Grpcci dmnfds Sm_S_:Gi Sm_S_: Gimpl:
## Smntc_Smlr_ -0.626
## Grpcchlrmp -0.709  0.444
## domainfoods -0.375  0.572  0.266
## Smntc_S_:Gi  0.456 -0.728 -0.623 -0.416
## Smntc_Sml_:  0.400 -0.638 -0.284 -0.978  0.464
## Grpcimplnt:  0.256 -0.392 -0.357 -0.684  0.547   0.669
## Smnt_S_:Gi: -0.277  0.442  0.376  0.677 -0.603  -0.693 -0.977

```

```

car::Anova(comb_semsim_model)

```

```

## Analysis of Deviance Table (Type II Wald chisquare tests)

```



```
##
## Response: Numb_of_Items
##
##           Chisq Df Pr(>Chisq)
## Semantic_Similarity_mean      1.8320  1  0.1758901
## Group                        11.3328  1  0.0007615 ***
## domain                      180.2000  1  < 2.2e-16 ***
## Semantic_Similarity_mean:Group      4.5597  1  0.0327329 *
## Semantic_Similarity_mean:domain      0.0455  1  0.8310287
## Group:domain                    57.7842  1  2.925e-14 ***
## Semantic_Similarity_mean:Group:domain  6.9277  1  0.0084870 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
#Does mean phonological similarity x domain type predict numb items produced
comb_freq_model = lmer(data = comb_fluency,
  Numb_of_Items ~ Frequency_Value_mean*Group*domain + (1|Subject))
summary(comb_freq_model)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: Numb_of_Items ~ Frequency_Value_mean * Group * domain + (1 |
##   Subject)
##   Data: comb_fluency
##
## REML criterion at convergence: 4133.7
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.53959 -0.49999 -0.00426  0.48417  2.60806
##
## Random effects:
##   Groups   Name                Variance Std.Dev.
##   Subject  (Intercept)  30.707     5.541
##   Residual                    5.207     2.282
## Number of obs: 864, groups:  Subject, 64
##
## Fixed effects:
##
##              Estimate Std. Error
## (Intercept)      -9.194      6.635
## Frequency_Value_mean       7.714      1.661
## Groupcochlear implant     70.822     12.710
## domainfoods        153.857     10.055
## Frequency_Value_mean:Groupcochlear implant    -18.645      3.164
## Frequency_Value_mean:domainfoods             -36.853      2.505
## Groupcochlear implant:domainfoods           -197.723     16.021
## Frequency_Value_mean:Groupcochlear implant:domainfoods  48.012      3.976
##
##              t value
## (Intercept)      -1.386
## Frequency_Value_mean       4.645
## Groupcochlear implant     5.572
## domainfoods        15.302
## Frequency_Value_mean:Groupcochlear implant     -5.893
## Frequency_Value_mean:domainfoods            -14.710
## Groupcochlear implant:domainfoods            -12.342
## Frequency_Value_mean:Groupcochlear implant:domainfoods  12.076
```

```
##
## Correlation of Fixed Effects:
##          (Intr) Frq_V_ Grpcci dmnfds Fr_V_:Gi Fr_V_: Gimpl:
## Frqncy_Vl_m -0.989
## Grpcchlrmp -0.522  0.516
## domainfoods -0.711  0.719  0.371
## Frqnc_V_:Gi  0.519 -0.525 -0.994 -0.377
## Frqncy_Vl_:  0.723 -0.731 -0.377 -1.000  0.384
## Grpcimplnt:  0.446 -0.451 -0.813 -0.628  0.818    0.627
## Frqn_V_:Gi: -0.455  0.461  0.819  0.630 -0.824   -0.630 -1.000
```

```
car::Anova(comb_freq_model)
```

```
## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: Numb_of_Items
##
##              Chisq Df Pr(>Chisq)
## Frequency_Value_mean      62.486  1  2.683e-15 ***
## Group                    11.506  1  0.0006937 ***
## domain                  316.973  1  < 2.2e-16 ***
## Frequency_Value_mean:Group    51.518  1  7.093e-13 ***
## Frequency_Value_mean:domain    83.629  1  < 2.2e-16 ***
## Group:domain              114.581  1  < 2.2e-16 ***
## Frequency_Value_mean:Group:domain 145.827  1  < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

#Clustering Analyses

```
comb_clustering<-rbind(full_animals_fluency,full_foods_fluency)

comb_clusters<- comb_clustering %>% mutate(fullswitchmethod=Switch_Method)%>%
  separate(Switch_Method,
           into = c("method", "param1", "param2","param3"), sep = "_",fill = "right")

#Average cluster sizes across switch methods
comb_cluster_size = comb_clusters %>%
  group_by(Group,method,param1,param2,struct,dimension,domain) %>%
  summarise(
    cluster_mean = mean(Cluster_Size_mean),
    cluster_sd = sd(Cluster_Size_std),
    num_switches = mean(Number_of_Switches),
    sd_switches = mean(Number_of_Switches))
```

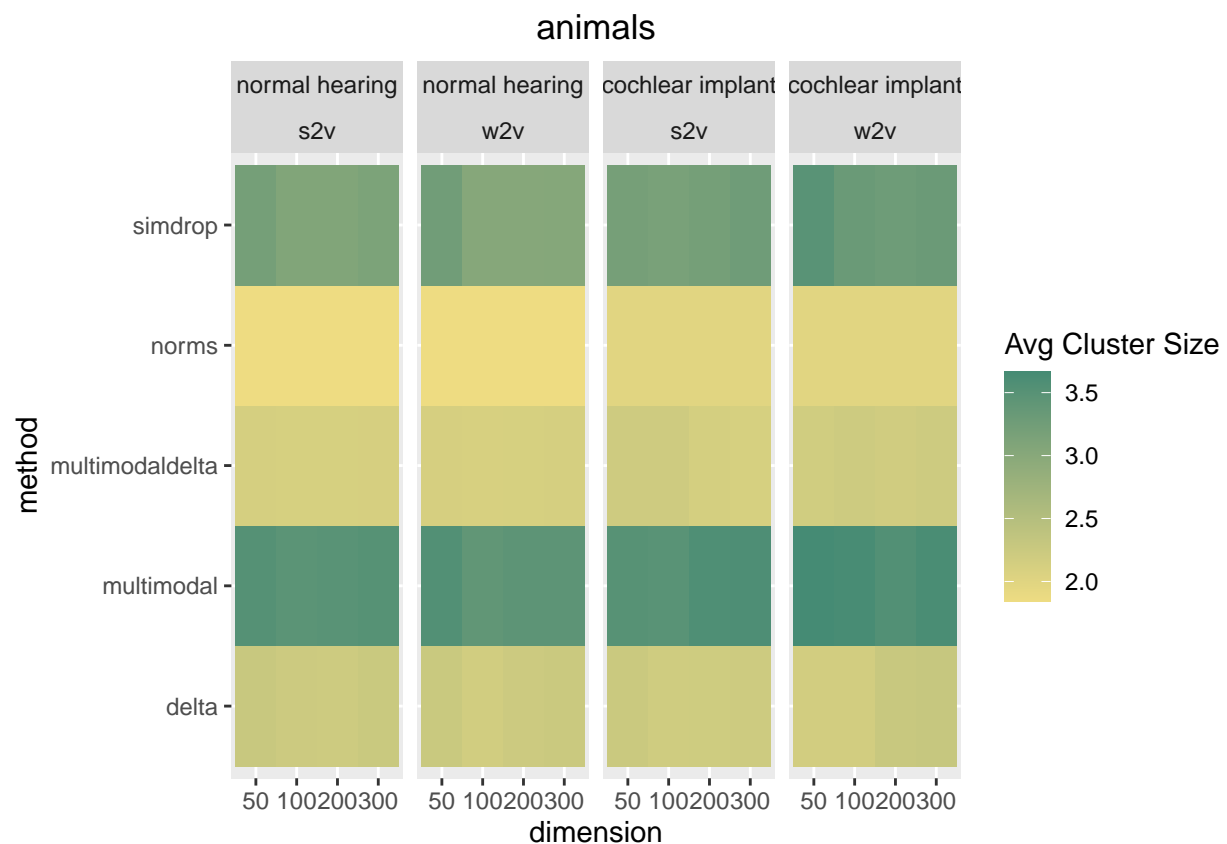
```
## 'summarise()' has grouped output by 'Group', 'method', 'param1', 'param2',
## 'struct', 'dimension'. You can override using the '.groups' argument.
```

```
#Across Switch Method params
coll_cluster_size = comb_clusters %>%
  group_by(Group,method,struct,dimension,domain) %>%
  summarise(
    cluster_mean = mean(Cluster_Size_mean),
```

```
cluster_sd = sd(Cluster_Size_std),
num_switches = mean(Number_of_Switches),
sd_switches = mean(Number_of_Switches))
```

## 'summarise()' has grouped output by 'Group', 'method', 'struct', 'dimension'.  
## You can override using the '.groups' argument.

```
#cluster size
#animals
desired_order <- c("50", "100", "200", "300")
coll_cluster_size <- coll_cluster_size %>%
  mutate(dimension = factor(dimension, levels = desired_order))
coll_cluster_size %>% filter(domain=="animals")%>%
  ggplot(aes(dimension, method, fill = cluster_mean)) +
  geom_tile() +
  facet_grid(~ Group * struct) +
  scale_fill_gradient(low = "lightgoldenrod2", high = "aquamarine4") +
  labs(title="animals", fill = "Avg Cluster Size")+
  theme(
    plot.title = element_text(hjust = 0.5)
  )
```

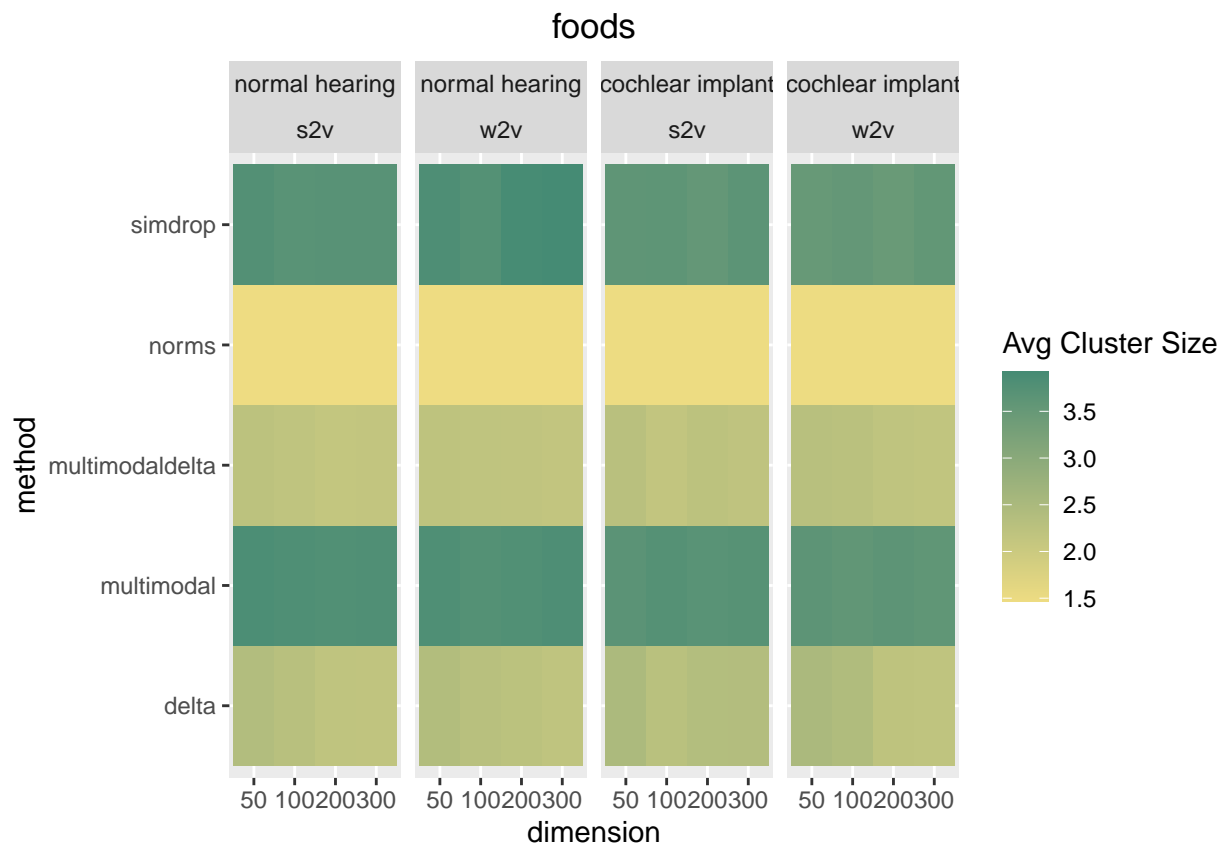


```
#foods
coll_cluster_size <- coll_cluster_size %>%
```

```

mutate(dimension = factor(dimension, levels = desired_order))
coll_cluster_size %>% filter(domain=="foods")%>%
  ggplot(aes(dimension, method, fill = cluster_mean)) +
  geom_tile() +
  facet_grid(~ Group * struct) +
  scale_fill_gradient(low = "lightgoldenrod2", high = "aquamarine4") +
  labs(title="foods", fill = "Avg Cluster Size")+
  theme(
    plot.title = element_text(hjust = 0.5)
  )

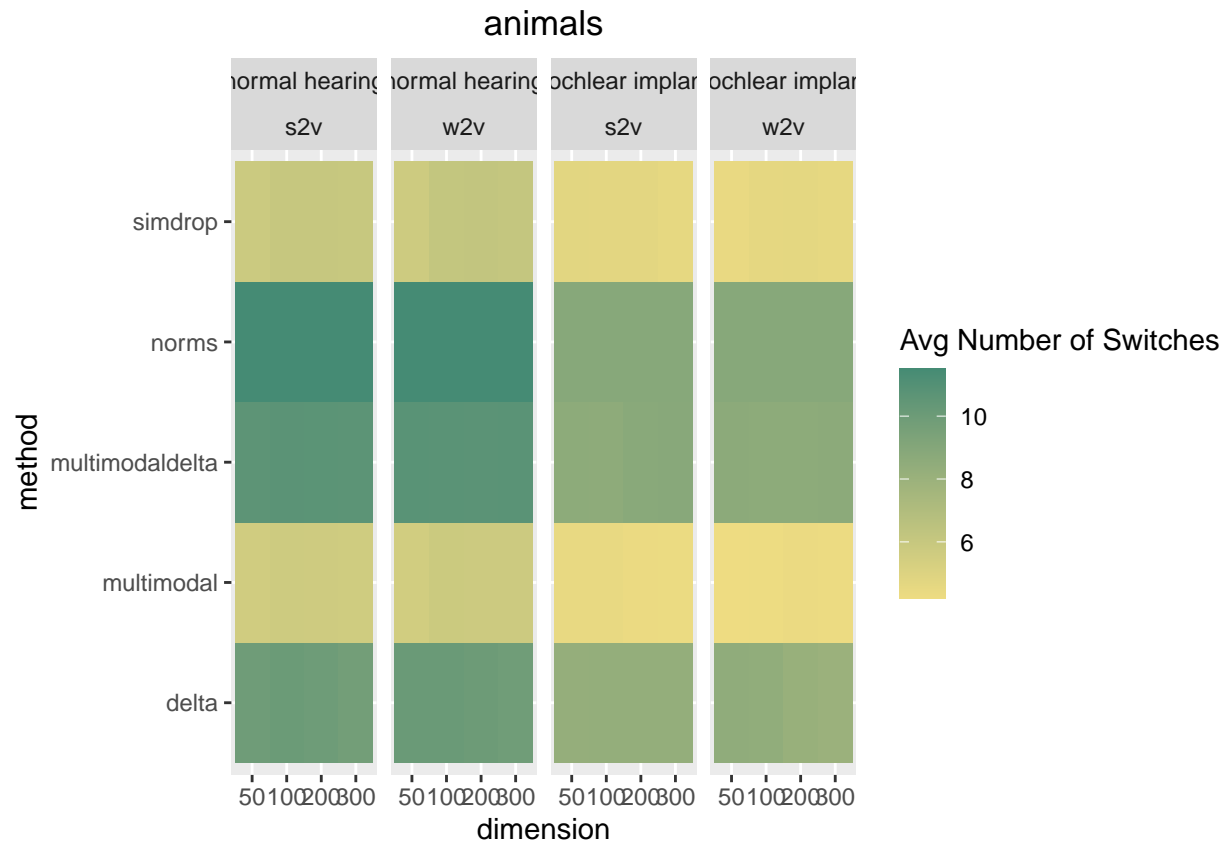
```



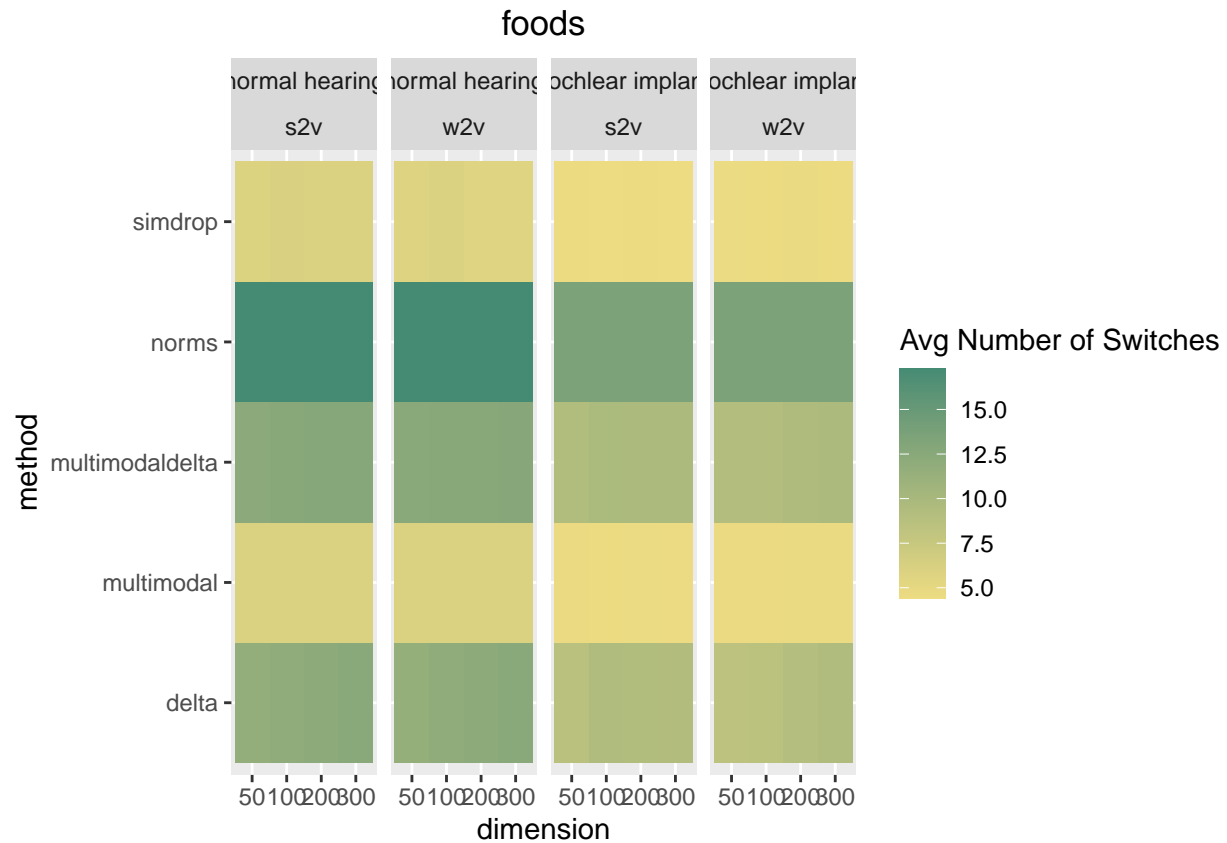
```

#Number of switches across methods, structures, and dimensions
#animals
coll_cluster_size <- coll_cluster_size %>%
  mutate(dimension = factor(dimension, levels = desired_order))
coll_cluster_size %>% filter(domain=="animals")%>%
  ggplot(aes(dimension, method, fill = num_switches)) +
  geom_tile() +
  facet_grid(~ Group * struct) +
  scale_fill_gradient(low = "lightgoldenrod2", high = "aquamarine4") +
  labs(title="animals", fill = "Avg Number of Switches")+
  theme(
    plot.title = element_text(hjust = 0.5)
  )

```



```
#foods
coll_cluster_size <- coll_cluster_size %>%
  mutate(dimension = factor(dimension, levels = desired_order))
coll_cluster_size %>% filter(domain=="foods")%>%
  ggplot(aes(dimension, method, fill = num_switches)) +
  geom_tile() +
  facet_grid(~ Group * struct) +
  scale_fill_gradient(low = "lightgoldenrod2", high = "aquamarine4") +
  labs(title="foods", fill = "Avg Number of Switches")+
  theme(
    plot.title = element_text(hjust = 0.5)
  )
```



```
##Does method determining numb of switches sig differ for each Group x domain
comp_switch_lm = lmer(data = comb_clusters, Number_of_Switches ~ Group*method*domain+ (1|Subject))
summary(comp_switch_lm)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: Number_of_Switches ~ Group * method * domain + (1 | Subject)
## Data: comb_clusters
##
## REML criterion at convergence: 1471759
##
## Scaled residuals:
## Min      1Q  Median      3Q      Max
## -4.0257 -0.6489 -0.0140  0.6107  5.3098
##
## Random effects:
## Groups Name Variance Std.Dev.
## Subject (Intercept) 8.259 2.874
## Residual 13.264 3.642
## Number of obs: 271296, groups: Subject, 64
##
## Fixed effects:
## Estimate Std. Error
## (Intercept) 9.98406 0.51008
## Groupcochlear implant -1.68141 0.72136
## methodmultimodal -4.28094 0.08236
## methodmultimodaldelta 0.75085 0.04755
```

```

## methodnorms                1.53156    0.16727
## methodsimdrop              -3.95672    0.23213
## domainfoods                2.33524    0.06908
## Groupcochlear implant:methodmultimodal    0.33339    0.11647
## Groupcochlear implant:methodmultimodaldelta -0.36604    0.06725
## Groupcochlear implant:methodnorms        -0.94359    0.23656
## Groupcochlear implant:methodsimdrop       0.29078    0.32829
## Groupcochlear implant:domainfoods        -2.02135    0.10221
## methodmultimodal:domainfoods            -1.80306    0.12436
## methodmultimodaldelta:domainfoods        -0.21253    0.07180
## methodnorms:domainfoods                3.68444    0.25257
## methodsimdrop:domainfoods             -2.21728    0.35052
## Groupcochlear implant:methodmultimodal:domainfoods    1.36879    0.18350
## Groupcochlear implant:methodmultimodaldelta:domainfoods 0.35855    0.10594
## Groupcochlear implant:methodnorms:domainfoods    0.21180    0.37269
## Groupcochlear implant:methodsimdrop:domainfoods    1.38059    0.51721
##                                t value
## (Intercept)                  19.574
## Groupcochlear implant        -2.331
## methodmultimodal            -51.979
## methodmultimodaldelta       15.791
## methodnorms                  9.156
## methodsimdrop              -17.045
## domainfoods                 33.806
## Groupcochlear implant:methodmultimodal    2.862
## Groupcochlear implant:methodmultimodaldelta -5.443
## Groupcochlear implant:methodnorms        -3.989
## Groupcochlear implant:methodsimdrop       0.886
## Groupcochlear implant:domainfoods        -19.776
## methodmultimodal:domainfoods            -14.499
## methodmultimodaldelta:domainfoods        -2.960
## methodnorms:domainfoods                14.588
## methodsimdrop:domainfoods             -6.326
## Groupcochlear implant:methodmultimodal:domainfoods    7.459
## Groupcochlear implant:methodmultimodaldelta:domainfoods 3.384
## Groupcochlear implant:methodnorms:domainfoods    0.568
## Groupcochlear implant:methodsimdrop:domainfoods    2.669

##
## Correlation matrix not shown by default, as p = 20 > 12.
## Use print(x, correlation=TRUE) or
##     vcov(x)         if you need it

```

```
car::Anova(comp_switch_lm)
```

```

## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: Number_of_Switches
##              Chisq Df Pr(>Chisq)
## Group          12.624  1  0.0003809 ***
## method        20809.344  4 < 2.2e-16 ***
## domain         8030.346  1 < 2.2e-16 ***
## Group:method    253.248  4 < 2.2e-16 ***

```

```
## Group:domain          2795.573  1  < 2.2e-16 ***
## method:domain         714.240  4  < 2.2e-16 ***
## Group:method:domain   60.267  4  2.549e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
emmeans(comp_switch_lm,pairwise~Group*method,simple="method")
```

```
## Note: D.f. calculations have been disabled because the number of observations exceeds 3000.
## To enable adjustments, add the argument 'pbkrtest.limit = 271296' (or larger)
## [or, globally, 'set emm_options(pbkrtest.limit = 271296)' or larger];
## but be warned that this may result in large computation time and memory use.
## Note: D.f. calculations have been disabled because the number of observations exceeds 3000.
## To enable adjustments, add the argument 'lmerTest.limit = 271296' (or larger)
## [or, globally, 'set emm_options(lmerTest.limit = 271296)' or larger];
## but be warned that this may result in large computation time and memory use.
## NOTE: Results may be misleading due to involvement in interactions
```

```
## $emmeans
## Group          method      emmean    SE df asymp.LCL asymp.UCL
## normal hearing  delta        11.15 0.509 Inf    10.15    12.15
## cochlear implant delta        8.46 0.509 Inf     7.46     9.46
## normal hearing  multimodal     5.97 0.511 Inf     4.97     6.97
## cochlear implant multimodal     4.29 0.511 Inf     3.29     5.30
## normal hearing  multimodaldelta 11.80 0.508 Inf    10.80    12.79
## cochlear implant multimodaldelta 8.92 0.508 Inf     7.92     9.91
## normal hearing  norms         14.53 0.522 Inf    13.50    15.55
## cochlear implant norms         11.00 0.525 Inf     9.97    12.02
## normal hearing  simdrop        6.09 0.536 Inf     5.04     7.14
## cochlear implant simdrop        4.38 0.541 Inf     3.31     5.44
##
```

```
## Results are averaged over the levels of: domain
## Degrees-of-freedom method: asymptotic
## Confidence level used: 0.95
##
```

```
## $contrasts
## Group = normal hearing:
## contrast          estimate    SE df  z.ratio p.value
## delta - multimodal    5.1825 0.0622 Inf   83.347 <.0001
## delta - multimodaldelta -0.6446 0.0359 Inf  -17.956 <.0001
## delta - norms        -3.3738 0.1263 Inf  -26.715 <.0001
## delta - simdrop       5.0654 0.1753 Inf   28.902 <.0001
## multimodal - multimodaldelta -5.8271 0.0528 Inf -110.273 <.0001
## multimodal - norms      -8.5563 0.1321 Inf  -64.768 <.0001
## multimodal - simdrop    -0.1171 0.1795 Inf   -0.652 0.9662
## multimodaldelta - norms  -2.7292 0.1220 Inf  -22.378 <.0001
## multimodaldelta - simdrop  5.7099 0.1722 Inf   33.165 <.0001
## norms - simdrop        8.4391 0.2105 Inf   40.095 <.0001
##
```

```
## Group = cochlear implant:
## contrast          estimate    SE df  z.ratio p.value
## delta - multimodal    4.1647 0.0675 Inf   61.730 <.0001
## delta - multimodaldelta -0.4578 0.0390 Inf  -11.754 <.0001
```



```
## delta - norms -2.5361 0.1370 Inf -18.508 <.0001
## delta - simdrop 4.0843 0.1902 Inf 21.478 <.0001
## multimodal - multimodaldelta -4.6225 0.0573 Inf -80.622 <.0001
## multimodal - norms -6.7008 0.1433 Inf -46.748 <.0001
## multimodal - simdrop -0.0804 0.1948 Inf -0.413 0.9939
## multimodaldelta - norms -2.0783 0.1323 Inf -15.705 <.0001
## multimodaldelta - simdrop 4.5421 0.1868 Inf 24.315 <.0001
## norms - simdrop 6.6204 0.2284 Inf 28.989 <.0001
##
## Results are averaged over the levels of: domain
## Degrees-of-freedom method: asymptotic
## P value adjustment: tukey method for comparing a family of 5 estimates
```

```
#Across switch methods, does avg number of switches differ by Group, domain, struct, & domain
switch_lm = lmer(data = comb_clusters, Number_of_Switches ~ Group*domain*struct*dimension + (1|Subject))
summary(switch_lm)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: Number_of_Switches ~ Group * domain * struct * dimension + (1 |
## Subject)
## Data: comb_clusters
##
## REML criterion at convergence: 1492601
##
## Scaled residuals:
## Min 1Q Median 3Q Max
## -4.1199 -0.6594 -0.0110 0.6151 5.2390
##
## Random effects:
## Groups Name Variance Std.Dev.
## Subject (Intercept) 8.259 2.874
## Residual 14.320 3.784
## Number of obs: 271296, groups: Subject, 64
##
## Fixed effects:
## Estimate Std. Error
## (Intercept) 10.502389 0.509425
## Groupcochlear implant -2.068670 0.720436
## domainfoods 2.056338 0.057439
## structw2v 0.025279 0.053389
## dimension200 -0.032544 0.053389
## dimension300 -0.060012 0.053389
## dimension50 -0.081210 0.053389
## Groupcochlear implant:domainfoods -1.298770 0.085127
## Groupcochlear implant:structw2v -0.001294 0.075503
## domainfoods:structw2v -0.045024 0.080616
## Groupcochlear implant:dimension200 0.190585 0.075503
## Groupcochlear implant:dimension300 0.266222 0.075503
## Groupcochlear implant:dimension50 0.093949 0.075503
## domainfoods:dimension200 0.195092 0.080616
## domainfoods:dimension300 0.289693 0.080616
## domainfoods:dimension50 -0.106815 0.080616
## structw2v:dimension200 0.004279 0.075503
## structw2v:dimension300 0.032245 0.075503
```

```

## structw2v:dimension50                0.101513    0.075503
## Groupcochlear implant:domainfoods:structw2v -0.517345    0.118953
## Groupcochlear implant:domainfoods:dimension200 -0.431913    0.118953
## Groupcochlear implant:domainfoods:dimension300 -0.549038    0.118953
## Groupcochlear implant:domainfoods:dimension50 -0.382122    0.118953
## Groupcochlear implant:structw2v:dimension200 -0.183021    0.106778
## Groupcochlear implant:structw2v:dimension300 -0.243432    0.106778
## Groupcochlear implant:structw2v:dimension50 -0.083599    0.106778
## domainfoods:structw2v:dimension200 -0.108738    0.114008
## domainfoods:structw2v:dimension300 -0.084857    0.114008
## domainfoods:structw2v:dimension50 -0.014252    0.114008
## Groupcochlear implant:domainfoods:structw2v:dimension200 0.673668    0.168225
## Groupcochlear implant:domainfoods:structw2v:dimension300 0.825711    0.168225
## Groupcochlear implant:domainfoods:structw2v:dimension50 0.457451    0.168225
##                                     t value
## (Intercept)                        20.616
## Groupcochlear implant              -2.871
## domainfoods                        35.800
## structw2v                          0.473
## dimension200                       -0.610
## dimension300                       -1.124
## dimension50                        -1.521
## Groupcochlear implant:domainfoods -15.257
## Groupcochlear implant:structw2v   -0.017
## domainfoods:structw2v             -0.559
## Groupcochlear implant:dimension200  2.524
## Groupcochlear implant:dimension300  3.526
## Groupcochlear implant:dimension50  1.244
## domainfoods:dimension200           2.420
## domainfoods:dimension300           3.594
## domainfoods:dimension50            -1.325
## structw2v:dimension200              0.057
## structw2v:dimension300              0.427
## structw2v:dimension50               1.344
## Groupcochlear implant:domainfoods:structw2v -4.349
## Groupcochlear implant:domainfoods:dimension200 -3.631
## Groupcochlear implant:domainfoods:dimension300 -4.616
## Groupcochlear implant:domainfoods:dimension50 -3.212
## Groupcochlear implant:structw2v:dimension200 -1.714
## Groupcochlear implant:structw2v:dimension300 -2.280
## Groupcochlear implant:structw2v:dimension50 -0.783
## domainfoods:structw2v:dimension200 -0.954
## domainfoods:structw2v:dimension300 -0.744
## domainfoods:structw2v:dimension50 -0.125
## Groupcochlear implant:domainfoods:structw2v:dimension200 4.005
## Groupcochlear implant:domainfoods:structw2v:dimension300 4.908
## Groupcochlear implant:domainfoods:structw2v:dimension50 2.719

##
## Correlation matrix not shown by default, as p = 32 > 12.
## Use print(x, correlation=TRUE) or
##     vcov(x)         if you need it

```

```
car::Anova(switch_lm)
```

```
## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: Number_of_Switches
##
##           Chisq Df Pr(>Chisq)
## Group      12.6243  1  0.0003808 ***
## domain    7438.2697  1 < 2.2e-16 ***
## struct      9.8052  1  0.0017402 **
## dimension  86.1851  3 < 2.2e-16 ***
## Group:domain 2589.3502  1 < 2.2e-16 ***
## Group:struct  23.0786  1  1.555e-06 ***
## domain:struct  13.7505  1  0.0002088 ***
## Group:dimension  8.0707  3  0.0445723 *
## domain:dimension 85.5668  3 < 2.2e-16 ***
## struct:dimension 15.8562  3  0.0012136 **
## Group:domain:struct  0.2238  1  0.6361514
## Group:domain:dimension 3.9932  3  0.2622021
## Group:struct:dimension 1.9242  3  0.5882933
## domain:struct:dimension 13.0762  3  0.0044747 **
## Group:domain:struct:dimension 27.3919  3  4.872e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
emmeans(switch_lm,pairwise~Group*struct,simple="struct")
```

```
## Note: D.f. calculations have been disabled because the number of observations exceeds 3000.
## To enable adjustments, add the argument 'pbkrtest.limit = 271296' (or larger)
## [or, globally, 'set emm_options(pbkrtest.limit = 271296)' or larger];
## but be warned that this may result in large computation time and memory use.
## Note: D.f. calculations have been disabled because the number of observations exceeds 3000.
## To enable adjustments, add the argument 'lmerTest.limit = 271296' (or larger)
## [or, globally, 'set emm_options(lmerTest.limit = 271296)' or larger];
## but be warned that this may result in large computation time and memory use.
## NOTE: Results may be misleading due to involvement in interactions
```

```
## $emmeans
## Group      struct emmean    SE df asymp.LCL asymp.UCL
## normal hearing s2v      11.53 0.508 Inf      10.54      12.53
## cochlear implant s2v      8.78 0.508 Inf       7.79       9.78
## normal hearing w2v      11.55 0.508 Inf      10.55      12.54
## cochlear implant w2v      8.65 0.508 Inf       7.66       9.65
##
## Results are averaged over the levels of: domain, dimension
## Degrees-of-freedom method: asymptotic
## Confidence level used: 0.95
##
## $contrasts
## Group = normal hearing:
## contrast estimate    SE df z.ratio p.value
## s2v - w2v  -0.0113 0.0202 Inf  -0.560  0.5752
##
```

```

## Group = cochlear implant:
## contrast estimate SE df z.ratio p.value
## s2v - w2v 0.1316 0.0219 Inf 6.017 <.0001
##
## Results are averaged over the levels of: domain, dimension
## Degrees-of-freedom method: asymptotic

#Does cluster size mean sig differ for each method for each Group x domain
comp_clustersize_lm = lmer(data = comb_clusters, Number_of_Switches ~ Group*method*domain+ (1|Subject))
summary(comp_clustersize_lm)

## Linear mixed model fit by REML ['lmerMod']
## Formula: Number_of_Switches ~ Group * method * domain + (1 | Subject)
## Data: comb_clusters
##
## REML criterion at convergence: 1471759
##
## Scaled residuals:
## Min 1Q Median 3Q Max
## -4.0257 -0.6489 -0.0140 0.6107 5.3098
##
## Random effects:
## Groups Name Variance Std.Dev.
## Subject (Intercept) 8.259 2.874
## Residual 13.264 3.642
## Number of obs: 271296, groups: Subject, 64
##
## Fixed effects:
##
## Estimate Std. Error
## (Intercept) 9.98406 0.51008
## Groupcochlear implant -1.68141 0.72136
## methodmultimodal -4.28094 0.08236
## methodmultimodaldelta 0.75085 0.04755
## methodnorms 1.53156 0.16727
## methodsimdrop -3.95672 0.23213
## domainfoods 2.33524 0.06908
## Groupcochlear implant:methodmultimodal 0.33339 0.11647
## Groupcochlear implant:methodmultimodaldelta -0.36604 0.06725
## Groupcochlear implant:methodnorms -0.94359 0.23656
## Groupcochlear implant:methodsimdrop 0.29078 0.32829
## Groupcochlear implant:domainfoods -2.02135 0.10221
## methodmultimodal:domainfoods -1.80306 0.12436
## methodmultimodaldelta:domainfoods -0.21253 0.07180
## methodnorms:domainfoods 3.68444 0.25257
## methodsimdrop:domainfoods -2.21728 0.35052
## Groupcochlear implant:methodmultimodal:domainfoods 1.36879 0.18350
## Groupcochlear implant:methodmultimodaldelta:domainfoods 0.35855 0.10594
## Groupcochlear implant:methodnorms:domainfoods 0.21180 0.37269
## Groupcochlear implant:methodsimdrop:domainfoods 1.38059 0.51721
## t value
## (Intercept) 19.574
## Groupcochlear implant -2.331
## methodmultimodal -51.979
## methodmultimodaldelta 15.791

```

```
## methodnorms 9.156
## methodsimdrop -17.045
## domainfoods 33.806
## Groupcochlear implant:methodmultimodal 2.862
## Groupcochlear implant:methodmultimodaldelta -5.443
## Groupcochlear implant:methodnorms -3.989
## Groupcochlear implant:methodsimdrop 0.886
## Groupcochlear implant:domainfoods -19.776
## methodmultimodal:domainfoods -14.499
## methodmultimodaldelta:domainfoods -2.960
## methodnorms:domainfoods 14.588
## methodsimdrop:domainfoods -6.326
## Groupcochlear implant:methodmultimodal:domainfoods 7.459
## Groupcochlear implant:methodmultimodaldelta:domainfoods 3.384
## Groupcochlear implant:methodnorms:domainfoods 0.568
## Groupcochlear implant:methodsimdrop:domainfoods 2.669
```

```
##
## Correlation matrix not shown by default, as p = 20 > 12.
## Use print(x, correlation=TRUE) or
## vcov(x) if you need it
```

```
car::Anova(comp_clustersize_lm)
```

```
## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: Number_of_Switches
##           Chisq Df Pr(>Chisq)
## Group      12.624  1  0.0003809 ***
## method    20809.344  4 < 2.2e-16 ***
## domain     8030.346  1 < 2.2e-16 ***
## Group:method  253.248  4 < 2.2e-16 ***
## Group:domain  2795.573  1 < 2.2e-16 ***
## method:domain  714.240  4 < 2.2e-16 ***
## Group:method:domain  60.267  4 2.549e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
emmeans(comp_clustersize_lm, pairwise ~ Group * method, simple = "method")
```

```
## Note: D.f. calculations have been disabled because the number of observations exceeds 3000.
## To enable adjustments, add the argument 'pbkrtest.limit = 271296' (or larger)
## [or, globally, 'set emm_options(pbkrtest.limit = 271296)' or larger];
## but be warned that this may result in large computation time and memory use.
## Note: D.f. calculations have been disabled because the number of observations exceeds 3000.
## To enable adjustments, add the argument 'lmerTest.limit = 271296' (or larger)
## [or, globally, 'set emm_options(lmerTest.limit = 271296)' or larger];
## but be warned that this may result in large computation time and memory use.
## NOTE: Results may be misleading due to involvement in interactions
```

```
## $emmeans
## Group      method      emmean      SE df asymp.LCL asymp.UCL
```

```
## normal hearing delta 11.15 0.509 Inf 10.15 12.15
## cochlear implant delta 8.46 0.509 Inf 7.46 9.46
## normal hearing multimodal 5.97 0.511 Inf 4.97 6.97
## cochlear implant multimodal 4.29 0.511 Inf 3.29 5.30
## normal hearing multimodaldelta 11.80 0.508 Inf 10.80 12.79
## cochlear implant multimodaldelta 8.92 0.508 Inf 7.92 9.91
## normal hearing norms 14.53 0.522 Inf 13.50 15.55
## cochlear implant norms 11.00 0.525 Inf 9.97 12.02
## normal hearing simdrop 6.09 0.536 Inf 5.04 7.14
## cochlear implant simdrop 4.38 0.541 Inf 3.31 5.44
```

```
##
## Results are averaged over the levels of: domain
## Degrees-of-freedom method: asymptotic
## Confidence level used: 0.95
```

```
##
## $contrasts
## Group = normal hearing:
## contrast estimate SE df z.ratio p.value
## delta - multimodal 5.1825 0.0622 Inf 83.347 <.0001
## delta - multimodaldelta -0.6446 0.0359 Inf -17.956 <.0001
## delta - norms -3.3738 0.1263 Inf -26.715 <.0001
## delta - simdrop 5.0654 0.1753 Inf 28.902 <.0001
## multimodal - multimodaldelta -5.8271 0.0528 Inf -110.273 <.0001
## multimodal - norms -8.5563 0.1321 Inf -64.768 <.0001
## multimodal - simdrop -0.1171 0.1795 Inf -0.652 0.9662
## multimodaldelta - norms -2.7292 0.1220 Inf -22.378 <.0001
## multimodaldelta - simdrop 5.7099 0.1722 Inf 33.165 <.0001
## norms - simdrop 8.4391 0.2105 Inf 40.095 <.0001
```

```
##
## Group = cochlear implant:
## contrast estimate SE df z.ratio p.value
## delta - multimodal 4.1647 0.0675 Inf 61.730 <.0001
## delta - multimodaldelta -0.4578 0.0390 Inf -11.754 <.0001
## delta - norms -2.5361 0.1370 Inf -18.508 <.0001
## delta - simdrop 4.0843 0.1902 Inf 21.478 <.0001
## multimodal - multimodaldelta -4.6225 0.0573 Inf -80.622 <.0001
## multimodal - norms -6.7008 0.1433 Inf -46.748 <.0001
## multimodal - simdrop -0.0804 0.1948 Inf -0.413 0.9939
## multimodaldelta - norms -2.0783 0.1323 Inf -15.705 <.0001
## multimodaldelta - simdrop 4.5421 0.1868 Inf 24.315 <.0001
## norms - simdrop 6.6204 0.2284 Inf 28.989 <.0001
```

```
##
## Results are averaged over the levels of: domain
## Degrees-of-freedom method: asymptotic
## P value adjustment: tukey method for comparing a family of 5 estimates
```

```
#Does cluster size mean differ by Group x domain x struct x dimension
```

```
cluster_size_lm = lmer(data = comb_clusters, Cluster_Size_mean ~ Group*domain*struct*dimension + (1|Subject), data = comb_clusters)
summary(cluster_size_lm)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: Cluster_Size_mean ~ Group * domain * struct * dimension + (1 | Subject)
## Data: comb_clusters
```

```

##
## REML criterion at convergence: 753989.4
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.5971 -0.6040 -0.2498  0.3481 17.5572
##
## Random effects:
##   Groups   Name                Variance Std.Dev.
##   Subject  (Intercept)  0.06041  0.2458
##   Residual                    0.94113  0.9701
## Number of obs: 271296, groups: Subject, 64
##
## Fixed effects:
##
##                                     Estimate Std. Error
## (Intercept)                        2.170941    0.044515
## Groupcochlear implant              0.087417    0.062954
## domainfoods                       0.089377    0.014725
## structw2v                         -0.016903    0.013687
## dimension200                      0.002304    0.013687
## dimension300                      0.012988    0.013687
## dimension50                       0.019479    0.013687
## Groupcochlear implant:domainfoods -0.089884    0.021822
## Groupcochlear implant:structw2v   0.028010    0.019356
## domainfoods:structw2v             0.008214    0.020667
## Groupcochlear implant:dimension200 -0.066815    0.019356
## Groupcochlear implant:dimension300 -0.090516    0.019356
## Groupcochlear implant:dimension50 -0.015159    0.019356
## domainfoods:dimension200          -0.063015    0.020667
## domainfoods:dimension300          -0.068180    0.020667
## domainfoods:dimension50           0.029172    0.020667
## structw2v:dimension200            0.014517    0.019356
## structw2v:dimension300            0.014979    0.019356
## structw2v:dimension50             0.012382    0.019356
## Groupcochlear implant:domainfoods:structw2v 0.094190    0.030495
## Groupcochlear implant:domainfoods:dimension200 0.213574    0.030495
## Groupcochlear implant:domainfoods:dimension300 0.197570    0.030495
## Groupcochlear implant:domainfoods:dimension50 0.105262    0.030495
## Groupcochlear implant:structw2v:dimension200 0.024998    0.027374
## Groupcochlear implant:structw2v:dimension300 0.071877    0.027374
## Groupcochlear implant:structw2v:dimension50 -0.048302    0.027374
## domainfoods:structw2v:dimension200 0.040362    0.029227
## domainfoods:structw2v:dimension300 -0.001101    0.029227
## domainfoods:structw2v:dimension50 -0.025770    0.029227
## Groupcochlear implant:domainfoods:structw2v:dimension200 -0.247225    0.043126
## Groupcochlear implant:domainfoods:structw2v:dimension300 -0.247216    0.043126
## Groupcochlear implant:domainfoods:structw2v:dimension50 -0.046440    0.043126
##                                     t value
## (Intercept)                        48.769
## Groupcochlear implant              1.389
## domainfoods                       6.070
## structw2v                         -1.235
## dimension200                      0.168
## dimension300                      0.949

```

```
## dimension50 1.423
## Groupcochlear implant:domainfoods -4.119
## Groupcochlear implant:structw2v 1.447
## domainfoods:structw2v 0.397
## Groupcochlear implant:dimension200 -3.452
## Groupcochlear implant:dimension300 -4.676
## Groupcochlear implant:dimension50 -0.783
## domainfoods:dimension200 -3.049
## domainfoods:dimension300 -3.299
## domainfoods:dimension50 1.412
## structw2v:dimension200 0.750
## structw2v:dimension300 0.774
## structw2v:dimension50 0.640
## Groupcochlear implant:domainfoods:structw2v 3.089
## Groupcochlear implant:domainfoods:dimension200 7.004
## Groupcochlear implant:domainfoods:dimension300 6.479
## Groupcochlear implant:domainfoods:dimension50 3.452
## Groupcochlear implant:structw2v:dimension200 0.913
## Groupcochlear implant:structw2v:dimension300 2.626
## Groupcochlear implant:structw2v:dimension50 -1.765
## domainfoods:structw2v:dimension200 1.381
## domainfoods:structw2v:dimension300 -0.038
## domainfoods:structw2v:dimension50 -0.882
## Groupcochlear implant:domainfoods:structw2v:dimension200 -5.733
## Groupcochlear implant:domainfoods:structw2v:dimension300 -5.732
## Groupcochlear implant:domainfoods:structw2v:dimension50 -1.077
```

```
##
## Correlation matrix not shown by default, as p = 32 > 12.
## Use print(x, correlation=TRUE) or
##     vcov(x)         if you need it
```

```
car::Anova(cluster_size_lm)
```

```
## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: Cluster_Size_mean
##
```

	Chisq	Df	Pr(>Chisq)
## Group	1.3179	1	0.2509721
## domain	355.2329	1	< 2.2e-16 ***
## struct	7.2504	1	0.0070885 **
## dimension	107.7329	3	< 2.2e-16 ***
## Group:domain	5.0446	1	0.0247033 *
## Group:struct	9.9778	1	0.0015844 **
## domain:struct	0.9125	1	0.3394546
## Group:dimension	6.7383	3	0.0807241 .
## domain:dimension	70.0587	3	4.147e-15 ***
## struct:dimension	12.9283	3	0.0047942 **
## Group:domain:struct	7.2414	1	0.0071243 **
## Group:domain:dimension	22.2364	3	5.824e-05 ***
## Group:struct:dimension	16.3428	3	0.0009645 ***
## domain:struct:dimension	29.9638	3	1.404e-06 ***
## Group:domain:struct:dimension	55.1165	3	6.484e-12 ***



```

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

emmeans(cluster_size_lm, pairwise ~ Group * struct, simple = "struct")

## Note: D.f. calculations have been disabled because the number of observations exceeds 3000.
## To enable adjustments, add the argument 'pbkrtest.limit = 271296' (or larger)
## [or, globally, 'set emm_options(pbkrtest.limit = 271296)' or larger];
## but be warned that this may result in large computation time and memory use.
## Note: D.f. calculations have been disabled because the number of observations exceeds 3000.
## To enable adjustments, add the argument 'lmerTest.limit = 271296' (or larger)
## [or, globally, 'set emm_options(lmerTest.limit = 271296)' or larger];
## but be warned that this may result in large computation time and memory use.
## NOTE: Results may be misleading due to involvement in interactions

## $emmeans
## Group          struct emmean      SE df asymp.LCL asymp.UCL
## normal hearing  s2v      2.21 0.0436 Inf      2.13      2.30
## cochlear implant s2v      2.28 0.0437 Inf      2.19      2.36
## normal hearing  w2v      2.21 0.0436 Inf      2.13      2.30
## cochlear implant w2v      2.29 0.0437 Inf      2.21      2.38
##
## Results are averaged over the levels of: domain, dimension
## Degrees-of-freedom method: asymptotic
## Confidence level used: 0.95
##
## $contrasts
## Group = normal hearing:
## contrast estimate      SE df z.ratio p.value
## s2v - w2v  0.00064 0.00517 Inf   0.124  0.9014
##
## Group = cochlear implant:
## contrast estimate      SE df z.ratio p.value
## s2v - w2v -0.01900 0.00561 Inf  -3.389  0.0007
##
## Results are averaged over the levels of: domain, dimension
## Degrees-of-freedom method: asymptotic

#Does cluster size x Group predict # of switches
switch_clustsize_lm = lmer(data = comb_clusters, Number_of_Switches ~ Group * Cluster_Size_mean + (1 | Subject),
summary(switch_clustsize_lm)

## Linear mixed model fit by REML ['lmerMod']
## Formula: Number_of_Switches ~ Group * Cluster_Size_mean + (1 | Subject)
## Data: comb_clusters
##
## REML criterion at convergence: 1284787
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.4387 -0.5854 -0.1048  0.4304 19.4611
##

```

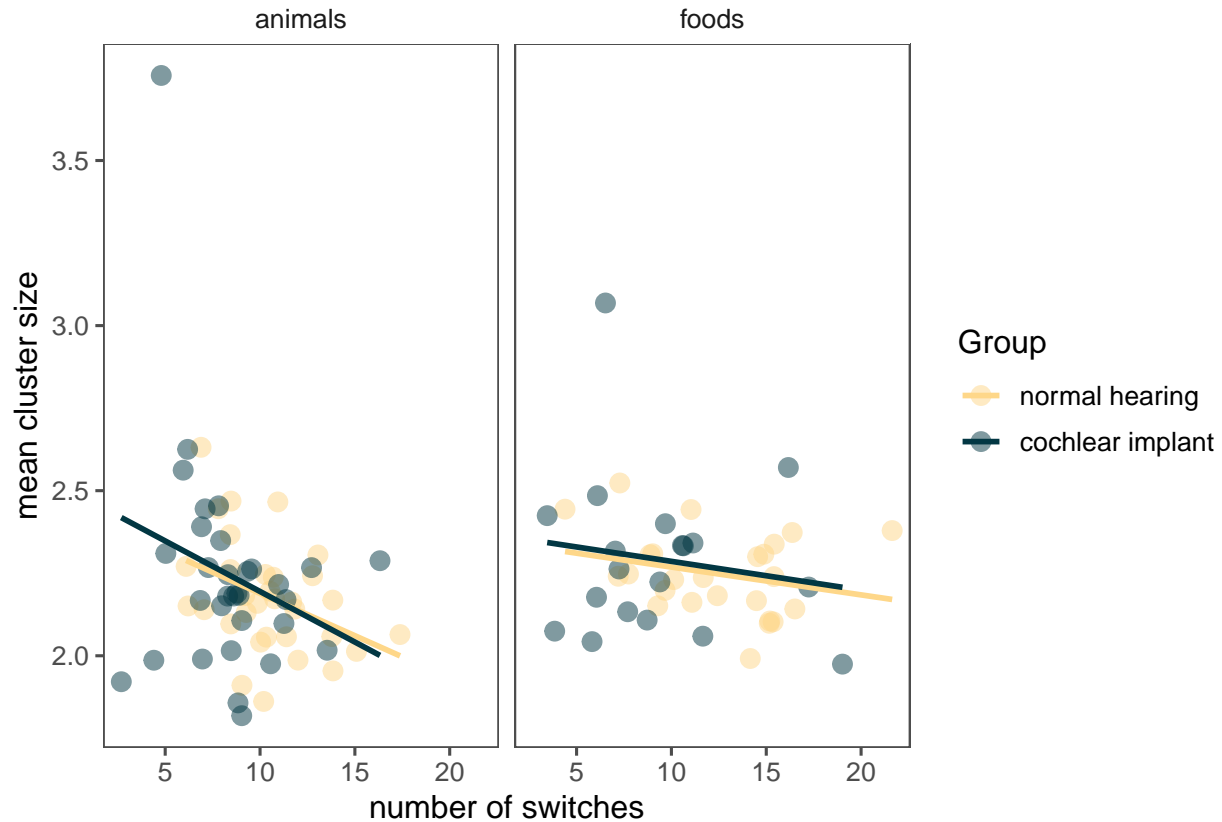
```
## Random effects:
## Groups Name Variance Std.Dev.
## Subject (Intercept) 7.919 2.814
## Residual 6.658 2.580
## Number of obs: 271296, groups: Subject, 64
##
## Fixed effects:
## Estimate Std. Error t value
## (Intercept) 19.418265 0.497796 39.008
## Groupcochlear implant -5.912257 0.703973 -8.398
## Cluster_Size_mean -3.679454 0.007551 -487.313
## Groupcochlear implant:Cluster_Size_mean 1.527192 0.010241 149.132
##
## Correlation of Fixed Effects:
## (Intr) Grpcci Cls_S_
## Grpcchlrmp -0.707
## Clstr_Sz_mn -0.033 0.024
## Gimpln:C_S_ 0.025 -0.032 -0.737
```

```
car::Anova(switch_clustsize_lm)
```

```
## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: Number_of_Switches
## Chisq Df Pr(>Chisq)
## Group 12.652 1 0.0003752 ***
## Cluster_Size_mean 312022.810 1 < 2.2e-16 ***
## Group:Cluster_Size_mean 22240.300 1 < 2.2e-16 ***
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
comb_clusters %>%
  group_by(Subject, Group, domain) %>%
  summarise(
    cluster_mean = mean(Cluster_Size_mean),
    cluster_sd = sd(Cluster_Size_std),
    num_switches = mean(Number_of_Switches),
    sd_switches = mean(Number_of_Switches)) %>%
  ggplot(aes(x = num_switches, y = cluster_mean, group = Group, color = Group)) +
  geom_point(alpha = 0.5, size = 3) +
  labs(y = "mean cluster size", x = "number of switches") +
  theme_few() +
  geom_smooth(method = "lm", se = FALSE) +
  facet_wrap(~domain) +
  scale_color_palette_d("nationalparkcolors::Acadia")
```

```
## 'summarise()' has grouped output by 'Subject', 'Group'. You can override using
## the '.groups' argument.
## 'geom_smooth()' using formula = 'y ~ x'
```



#Model Analyses

```
#combining model results
agg_model_results = rbind(full_animals_model,full_foods_model) %>%
  mutate(fullname=Model)%>%
  separate(Model,
    into = c("forage", "foraging_type", "method", "param1", "param2", "param3"), sep = "_",fill = "na")

#Dropping inf values
agg_model_results_clean <- agg_model_results %>%
  filter(!is.na(Negative_Log_Likelihood_Optimized) &
    is.finite(Negative_Log_Likelihood_Optimized))

#Summing nLL
agg_model_sum_nLL <- agg_model_results_clean %>% group_by(fullname,foraging_type,method,param1,param2,param3) %>%
  summarise(sum_nLL = sum(Negative_Log_Likelihood_Optimized)) %>%
  arrange(sum_nLL)

## 'summarise()' has grouped output by 'fullname', 'foraging_type', 'method',
## 'param1', 'param2', 'param3', 'Group', 'domain', 'struct'. You can override
## using the '.groups' argument.

#Does avg nLL differ across Groups x foraging_type
avg_nLL_model = lm(data = agg_model_sum_nLL,sum_nLL ~ Group*foraging_type)
summary(avg_nLL_model)
```

##

```
## Call:
## lm(formula = sum_nLL ~ Group * foraging_type, data = agg_model_sum_nLL)
##
## Residuals:
```

	Min	1Q	Median	3Q	Max
	-1365.43	-280.79	-22.83	157.13	2293.87

```
##
## Coefficients:
```

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	3607.466	7.035	512.762	< 2e-16
Groupcochlear implant	-1050.579	9.950	-105.591	< 2e-16
foraging_typephonologicaldynamicglobal	-3399.596	9.950	-341.685	< 2e-16
foraging_typephonologicaldynamiclocal	-2066.622	9.950	-207.711	< 2e-16
foraging_typephonologicaldynamicswitch	-3018.700	9.950	-303.402	< 2e-16
foraging_typephonologicalstatic	-3398.938	124.865	-27.221	< 2e-16
foraging_typerandom	358.034	124.865	2.867	0.00414
foraging_typestatic	32.462	124.865	0.260	0.79488
Groupcochlear implant:foraging_typephonologicaldynamicglobal	1011.559	14.071	71.891	< 2e-16
Groupcochlear implant:foraging_typephonologicaldynamiclocal	748.835	14.071	53.219	< 2e-16
Groupcochlear implant:foraging_typephonologicaldynamicswitch	957.891	14.071	68.077	< 2e-16
Groupcochlear implant:foraging_typephonologicalstatic	1010.728	176.586	5.724	1.05e-08
Groupcochlear implant:foraging_typerandom	-52.205	176.586	-0.296	0.76751
Groupcochlear implant:foraging_typestatic	-23.893	176.586	-0.135	0.89237

```
##
```

```
## (Intercept) ***
## Groupcochlear implant ***
## foraging_typephonologicaldynamicglobal ***
## foraging_typephonologicaldynamiclocal ***
## foraging_typephonologicaldynamicswitch ***
## foraging_typephonologicalstatic ***
## foraging typerandom **
## foraging_typestatic
## Groupcochlear implant:foraging_typephonologicaldynamicglobal ***
## Groupcochlear implant:foraging_typephonologicaldynamiclocal ***
## Groupcochlear implant:foraging_typephonologicaldynamicswitch ***
## Groupcochlear implant:foraging_typephonologicalstatic ***
## Groupcochlear implant:foraging typerandom
## Groupcochlear implant:foraging_typestatic
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 498.7 on 40274 degrees of freedom
## Multiple R-squared:  0.8423, Adjusted R-squared:  0.8423
## F-statistic: 1.655e+04 on 13 and 40274 DF,  p-value: < 2.2e-16
```

```
car::Anova(avg_nLL_model)
```

```
## Anova Table (Type II tests)
##
## Response: sum_nLL
##
```

	Sum Sq	Df	F value	Pr(>F)
## Group	1.3929e+09	1	5601.6	< 2.2e-16 ***
## foraging_type	5.0465e+10	6	33823.4	< 2.2e-16 ***
## Group:foraging_type	1.6527e+09	6	1107.7	< 2.2e-16 ***
## Residuals	1.0015e+10	40274		

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

```
emmeans(avg_nLL_model,pairwise~Group*foraging_type, simple="foraging_type")
```

```
## $emmeans
## Group      foraging_type      emmean      SE      df lower.CL
## normal hearing dynamic      3607.5    7.035 40274 3593.68
## cochlear implant dynamic      2556.9    7.035 40274 2543.10
## normal hearing phonologicaldynamicglobal 207.9    7.035 40274 194.08
## cochlear implant phonologicaldynamicglobal 168.8    7.035 40274 155.06
## normal hearing phonologicaldynamiclocal 1540.8    7.035 40274 1527.05
## cochlear implant phonologicaldynamiclocal 1239.1    7.035 40274 1225.31
## normal hearing phonologicaldynamicswitch 588.8    7.035 40274 574.98
## cochlear implant phonologicaldynamicswitch 496.1    7.035 40274 482.29
## normal hearing phonologicalstatic 208.5 124.667 40274 -35.82
## cochlear implant phonologicalstatic 168.7 124.667 40274 -75.67
## normal hearing random 3965.5 124.667 40274 3721.15
## cochlear implant random 2862.7 124.667 40274 2618.37
## normal hearing static 3639.9 124.667 40274 3395.58
## cochlear implant static 2565.5 124.667 40274 2321.11
## upper.CL
```

```

##      3621.3
##      2570.7
##      221.7
##      182.6
##      1554.6
##      1252.9
##      602.6
##      509.9
##      452.9
##      413.0
##      4209.9
##      3107.1
##      3884.3
##      2809.8
##
## Confidence level used: 0.95
##
## $contrasts
## Group = normal hearing:
## contrast                estimate      SE    df
## dynamic - phonologicaldynamicglobal 3399.596   9.95 40274
## dynamic - phonologicaldynamiclocal  2066.622   9.95 40274
## dynamic - phonologicaldynamicswitch 3018.700   9.95 40274
## dynamic - phonologicalstatic        3398.938 124.87 40274
## dynamic - random                   -358.034 124.87 40274
## dynamic - static                   -32.462 124.87 40274
## phonologicaldynamicglobal - phonologicaldynamiclocal -1332.973   9.95 40274
## phonologicaldynamicglobal - phonologicaldynamicswitch -380.895   9.95 40274
## phonologicaldynamicglobal - phonologicalstatic        -0.657 124.87 40274
## phonologicaldynamicglobal - random                   -3757.630 124.87 40274
## phonologicaldynamicglobal - static                   -3432.058 124.87 40274
## phonologicaldynamiclocal - phonologicaldynamicswitch   952.078   9.95 40274
## phonologicaldynamiclocal - phonologicalstatic        1332.316 124.87 40274
## phonologicaldynamiclocal - random                   -2424.657 124.87 40274
## phonologicaldynamiclocal - static                   -2099.084 124.87 40274
## phonologicaldynamicswitch - phonologicalstatic        380.238 124.87 40274
## phonologicaldynamicswitch - random                   -3376.735 124.87 40274
## phonologicaldynamicswitch - static                   -3051.162 124.87 40274
## phonologicalstatic - random                       -3756.973 176.31 40274
## phonologicalstatic - static                       -3431.400 176.31 40274
## random - static                                    325.572 176.31 40274
## t.ratio p.value
## 341.685 <.0001
## 207.711 <.0001
## 303.402 <.0001
## 27.221 <.0001
## -2.867 0.0629
## -0.260 1.0000
## -133.974 <.0001
## -38.283 <.0001
## -0.005 1.0000
## -30.093 <.0001
## -27.486 <.0001
## 95.691 <.0001

```

```

##      10.670 <.0001
##     -19.418 <.0001
##     -16.811 <.0001
##       3.045 0.0376
##     -27.043 <.0001
##     -24.436 <.0001
##     -21.309 <.0001
##     -19.463 <.0001
##       1.847 0.5162
##
## Group = cochlear implant:
## contrast                estimate      SE    df
## dynamic - phonologicaldynamicglobal 2388.037   9.95 40274
## dynamic - phonologicaldynamiclocal  1317.788   9.95 40274
## dynamic - phonologicaldynamicswitch 2060.810   9.95 40274
## dynamic - phonologicalstatic        2388.210 124.87 40274
## dynamic - random                   -305.830 124.87 40274
## dynamic - static                    -8.570 124.87 40274
## phonologicaldynamicglobal - phonologicaldynamiclocal -1070.249   9.95 40274
## phonologicaldynamicglobal - phonologicaldynamicswitch -327.227   9.95 40274
## phonologicaldynamicglobal - phonologicalstatic         0.173 124.87 40274
## phonologicaldynamicglobal - random                   -2693.867 124.87 40274
## phonologicaldynamicglobal - static                   -2396.607 124.87 40274
## phonologicaldynamiclocal - phonologicaldynamicswitch   743.022   9.95 40274
## phonologicaldynamiclocal - phonologicalstatic         1070.422 124.87 40274
## phonologicaldynamiclocal - random                   -1623.617 124.87 40274
## phonologicaldynamiclocal - static                   -1326.357 124.87 40274
## phonologicaldynamicswitch - phonologicalstatic        327.400 124.87 40274
## phonologicaldynamicswitch - random                   -2366.639 124.87 40274
## phonologicaldynamicswitch - static                   -2069.379 124.87 40274
## phonologicalstatic - random                       -2694.039 176.31 40274
## phonologicalstatic - static                       -2396.779 176.31 40274
## random - static                                   297.260 176.31 40274
## t.ratio p.value
##    240.016 <.0001
##    132.447 <.0001
##    207.127 <.0001
##     19.126 <.0001
##     -2.449 0.1783
##     -0.069 1.0000
##   -107.568 <.0001
##    -32.889 <.0001
##      0.001 1.0000
##   -21.574 <.0001
##   -19.194 <.0001
##    74.679 <.0001
##     8.573 <.0001
##   -13.003 <.0001
##   -10.622 <.0001
##     2.622 0.1193
##   -18.953 <.0001
##   -16.573 <.0001
##   -15.280 <.0001
##   -13.594 <.0001

```

```
##      1.686  0.6255
##
## P value adjustment: tukey method for comparing a family of 7 estimates
```

```
#best models for each Group x domain
best_models <- agg_model_sum_nLL %>%
  group_by(Group, domain) %>%
  slice_min(order_by = sum_nLL, n = 1) %>%
  ungroup() %>%
  select(-fullname) %>%
  mutate(foraging_type = as.factor(foraging_type),
         model_type = fct_recode(foraging_type,
                                `pstatic` = "phonologicalstatic",
                                `plocal` = "phonologicaldynamiclocal",
                                `pglobal` = "phonologicaldynamicglobal",
                                `pswitch` = "phonologicaldynamicswitch",
                                `static` = "static",
                                `dynamic` = "dynamic",
                                `random` = "random")) %>%
  select(Group, domain, struct, dimension, model_type, method, param1, param2, param3, sum_nLL)
```

```
## Warning: There was 1 warning in 'mutate()'.
## i In argument: 'model_type = fct_recode(...)'.
## Caused by warning:
## ! Unknown levels in 'f': phonologicalstatic, phonologicaldynamiclocal, static, dynamic, random
```

```
kable(best_models)
```

Group	domain	struct	dimension	model_type	method	param1	param2	param3	sum_nLL
normal hearing	animals	w2v	50	pswitch	multimodal	delta=0.4	rise=0.8	fall=0.5	227.3255
normal hearing	foods	w2v	50	pglobal	multimodal	delta=1.0	rise=0.8	fall=1.0	167.6816
cochlear implant	animals	s2v	50	pglobal	multimodal	delta=0.3	rise=0.5	fall=1.0	178.5507
cochlear implant	foods	w2v	100	pswitch	multimodal	delta=0.7	rise=0.8	fall=0.5	140.6336

```
#Getting betas
subject_best_models = agg_model_results_clean %>%
  group_by(Group, domain, Subject) %>%
  slice_min(Negative_Log_Likelihood_Optimized)

betas = subject_best_models %>%
  pivot_longer(names_to = "beta", cols = c(Beta_Frequency, Beta_Semantic, Beta_Phonological))

#Does use of semantic similarity differ across Groups x domains
beta_semantic_model = lmer(data = betas %>% filter(beta == "Beta_Semantic"), value ~ Group*domain + (1|S
summary(beta_semantic_model)
```

```
## Linear mixed model fit by REML ['lmerMod']
```



```
## Formula: value ~ Group * domain + (1 | Subject)
## Data: betas %>% filter(beta == "Beta_Semantic")
##
## REML criterion at convergence: 546.2
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.12743 -0.56355 -0.09113  0.41258  2.89072
##
## Random effects:
##   Groups   Name            Variance Std.Dev.
##   Subject  (Intercept)  0.8955    0.9463
##   Residual                8.0891    2.8441
## Number of obs: 110, groups: Subject, 64
##
## Fixed effects:
##                                     Estimate Std. Error t value
## (Intercept)                        7.2357     0.5299  13.655
## Groupcochlear implant                0.5556     0.7402   0.751
## domainfoods                       -3.7237     0.7638  -4.875
## Groupcochlear implant:domainfoods -0.8530     1.1261  -0.757
##
## Correlation of Fixed Effects:
##              (Intr) Grpcci dmnfds
## Grpcchlrmp -0.716
## domainfoods -0.625  0.447
## Grpcimplnt:  0.424 -0.592 -0.678
```

```
car::Anova(beta_semantic_model)
```

```
## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: value
##              Chisq Df Pr(>Chisq)
## Group          0.1407  1    0.7076
## domain        53.7849  1  2.237e-13 ***
## Group:domain   0.5738  1    0.4488
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
emmeans(beta_semantic_model, pairwise~Group*domain, simple="Group")
```

```
## $emmeans
##   Group      domain emmean    SE df lower.CL upper.CL
## normal hearing animals  7.24 0.530 105     6.19     8.29
## cochlear implant animals  7.79 0.518 102     6.76     8.82
## normal hearing  foods   3.51 0.602 106     2.32     4.71
## cochlear implant foods   3.21 0.693 106     1.84     4.59
##
## Degrees-of-freedom method: kenward-roger
## Confidence level used: 0.95
##
## $contrasts
```

```
## domain = animals:
## contrast estimate SE df t.ratio p.value
## normal hearing - cochlear implant -0.556 0.741 104 -0.750 0.4550
##
## domain = foods:
## contrast estimate SE df t.ratio p.value
## normal hearing - cochlear implant 0.297 0.918 106 0.324 0.7465
##
## Degrees-of-freedom method: kenward-roger
```

*#Does use of frequency differ across Groups x domains*

```
beta_freq_model = lmer(data = betas %>% filter(beta == "Beta_Frequency"), value ~ Group*domain +(1|Subject)
summary(beta_freq_model)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: value ~ Group * domain + (1 | Subject)
## Data: betas %>% filter(beta == "Beta_Frequency")
##
## REML criterion at convergence: 430.2
##
## Scaled residuals:
## Min 1Q Median 3Q Max
## -2.3910 -0.4956 0.0020 0.4582 4.0181
##
## Random effects:
## Groups Name Variance Std.Dev.
## Subject (Intercept) 0.6678 0.8172
## Residual 2.3885 1.5455
## Number of obs: 110, groups: Subject, 64
##
## Fixed effects:
## Estimate Std. Error t value
## (Intercept) 4.8133 0.3090 15.575
## Groupcochlear implant 0.2231 0.4328 0.516
## domainfoods 0.1215 0.4180 0.291
## Groupcochlear implant:domainfoods -0.1292 0.6197 -0.208
##
## Correlation of Fixed Effects:
## (Intr) Grpcci dmnfds
## Grpcchlrmp -0.714
## domainfoods -0.578 0.413
## Grpcimplnt: 0.390 -0.546 -0.675
```

```
car::Anova(beta_freq_model)
```

```
## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: value
## Chisq Df Pr(>Chisq)
## Group 0.2299 1 0.6316
## domain 0.0413 1 0.8389
## Group:domain 0.0435 1 0.8349
```

```
emmeans(beta_freq_model, pairwise ~ Group * domain, simple = "Group")
```

```
## $emmeans
##   Group          domain  emmean    SE    df lower.CL upper.CL
##   normal hearing  animals   4.81 0.309 102.0    4.20    5.43
##   cochlear implant animals   5.04 0.303  98.1    4.43    5.64
##   normal hearing   foods    4.93 0.349 105.0    4.24    5.63
##   cochlear implant foods    5.03 0.400 106.0    4.24    5.82
##
## Degrees-of-freedom method: kenward-roger
## Confidence level used: 0.95
##
## $contrasts
## domain = animals:
##   contrast                      estimate    SE  df t.ratio p.value
##   normal hearing - cochlear implant  -0.223 0.433 100  -0.515  0.6075
##
## domain = foods:
##   contrast                      estimate    SE  df t.ratio p.value
##   normal hearing - cochlear implant  -0.094 0.531 106  -0.177  0.8600
##
## Degrees-of-freedom method: kenward-roger
```

```
#Does use of phonological similarity differ across Groups x domains
```

```
beta_phon_model = lmer(data = betas %>% filter(beta == "Beta_Phonological"), value ~ Group * domain + (1 | Subject))
```

```
## boundary (singular) fit: see help('isSingular')
```

```
summary(beta_phon_model)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: value ~ Group * domain + (1 | Subject)
##   Data: betas %>% filter(beta == "Beta_Phonological")
##
## REML criterion at convergence: 309.1
##
## Scaled residuals:
##   Min       1Q   Median       3Q      Max
## -2.9391 -0.3615  0.0242  0.5728  3.3491
##
## Random effects:
##   Groups   Name                Variance Std.Dev.
##   Subject  (Intercept)  0.000      0.000
##   Residual                    2.288      1.513
## Number of obs: 85, groups: Subject, 57
##
## Fixed effects:
##                                Estimate Std. Error t value
## (Intercept)                   -0.7433     0.3382  -2.198
## Groupcochlear implant          -0.5942     0.4624  -1.285
## domainfoods                     1.2901     0.4624   2.790
## Groupcochlear implant:domainfoods  0.4276     0.6586   0.649
```

```
##
## Correlation of Fixed Effects:
##          (Intr) Grpcci dmnfds
## Grpcchlrmp -0.731
## domainfoods -0.731  0.535
## Grpcimplnt:  0.514 -0.702 -0.702
## optimizer (nloptwrap) convergence code: 0 (OK)
## boundary (singular) fit: see help('isSingular')
```

```
car::Anova(beta_phon_model)
```

```
## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: value
##           Chisq Df Pr(>Chisq)
## Group       1.3558  1    0.2443
## domain      20.7788  1  5.155e-06 ***
## Group:domain  0.4216  1    0.5162
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

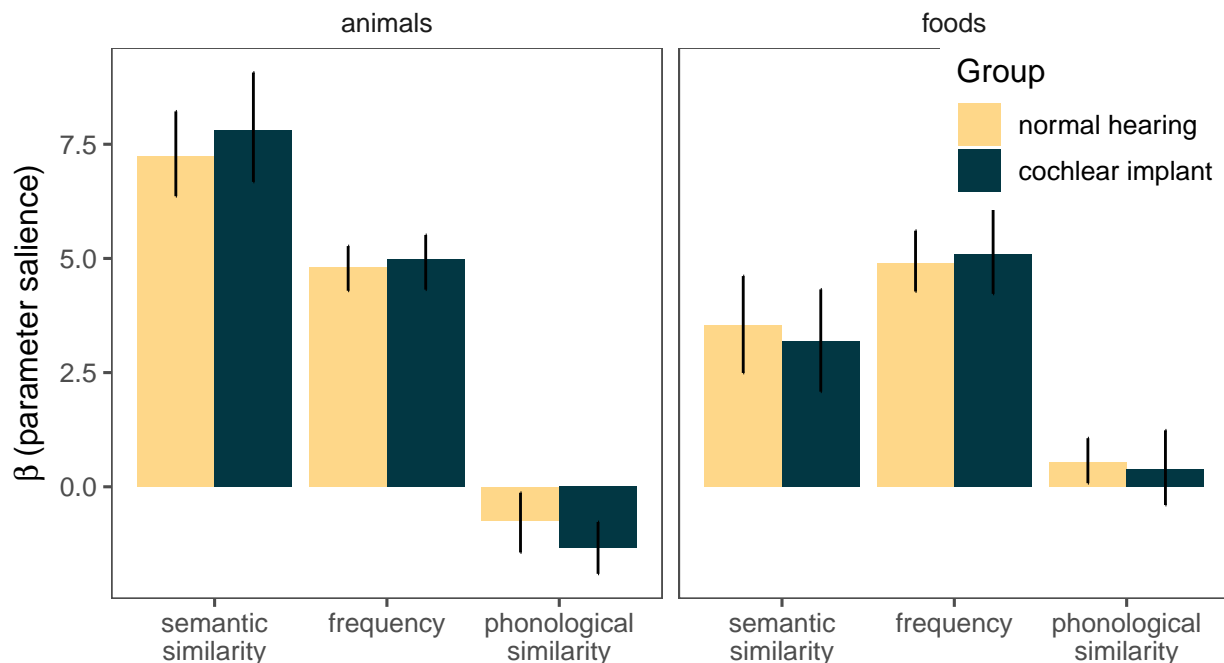
```
emmeans(beta_phon_model, pairwise~Group*domain, simple="Group")
```

```
## $emmeans
##   Group          domain emmean    SE df lower.CL upper.CL
## normal hearing  animals -0.743 0.342 81  -1.4234  -0.0631
## cochlear implant animals -1.337 0.318 80  -1.9708  -0.7041
## normal hearing  foods    0.547 0.318 81  -0.0855   1.1790
## cochlear implant foods    0.380 0.351 81  -0.3178   1.0782
##
## Degrees-of-freedom method: kenward-roger
## Confidence level used: 0.95
##
## $contrasts
## domain = animals:
## contrast                estimate    SE  df t.ratio p.value
## normal hearing - cochlear implant  0.594 0.467 80.8  1.272  0.2069
##
## domain = foods:
## contrast                estimate    SE  df t.ratio p.value
## normal hearing - cochlear implant  0.167 0.473 81.0  0.352  0.7258
##
## Degrees-of-freedom method: kenward-roger
```

```
betas %>%
  group_by(Group, beta, domain) %>%
  tidyboot_mean(value, nboot = 1000, na.rm = T) %>%
  separate(beta, into = c("b", "beta")) %>%
  mutate(beta = tolower(beta),
         beta = fct_recode(beta,
                           `semantic\nsimilarity` = "semantic", `phonological\nsimilarity` = "phonological",
                           `frequency` = "frequency")) %>%
```

```
mutate(beta = fct_relevel(beta, "semantic\nsimilarity", "frequency", "phonological\nsimilarity")) %>%
ggplot(aes(x = beta, y = empirical_stat, group = Group, fill = Group)) +
  geom_bar(stat = 'identity', position = "dodge") +
  geom_errorbar(aes(ymin = ci_lower, ymax = ci_upper), width = 0, position = position_dodge(0.9)) +
  labs(y = bquote(beta ~ "(parameter salience)"), x = "") +
  theme_few() +
  theme(aspect.ratio = 1, legend.position = c(0.87,0.86))+ facet_wrap(~domain)+
  scale_fill_paletteer_d("nationalparkcolors::Acadia")
```

```
## Warning: 'cols' is now required when using 'unnest()'.
## i Please use 'cols = c(strap)'.
```



```
#Looking at nLL's across methods
agg_model_sum_nLL_lab <- agg_model_sum_nLL %>%
  mutate(foraging_type = as.factor(foraging_type),
         model_type = fct_recode(foraging_type,
                                `pstatic` = "phonologicalstatic",
                                `plocal` = "phonologicaldynamiclocal",
                                `pglobal` = "phonologicaldynamicglobal",
                                `pswitch` = "phonologicaldynamicswitch",
                                `static` = "static",
                                `dynamic` = "dynamic",
                                `random` = "random"))
```

```

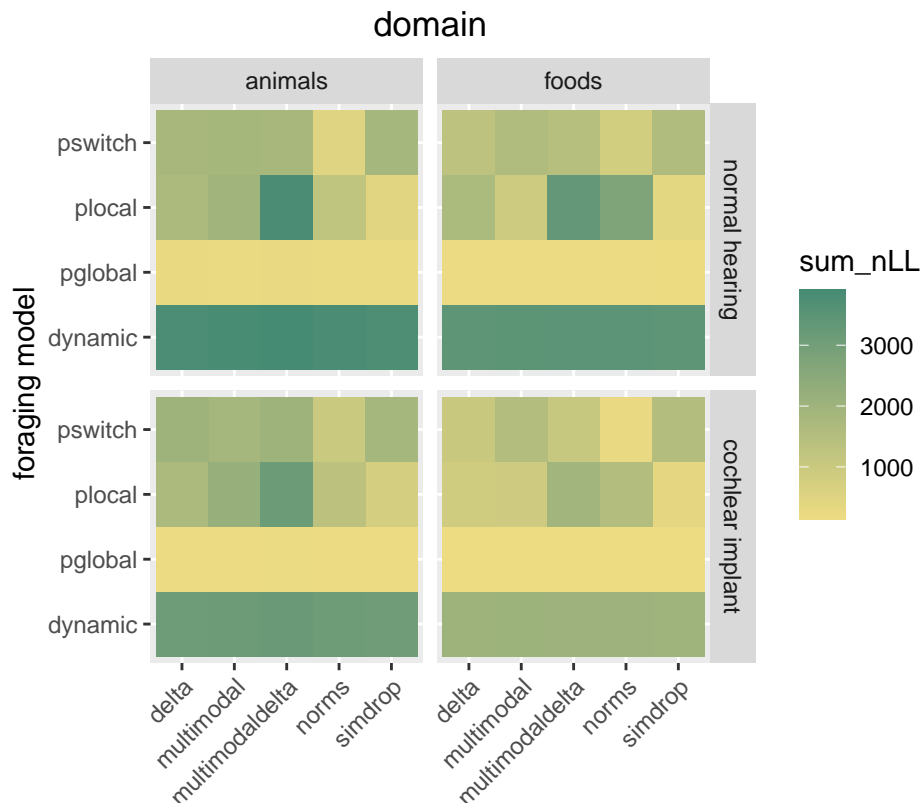
models <-c("dynamic","pswitch","pglobal","plocal")
methods <- c("norms", "simdrop", "delta", "multimodal", "multimodaldelta")
coll_agg_model_sum_nLL <- agg_model_sum_nLL_lab %>%
  filter(model_type %in% models) %>%
  filter(method %in% methods)

coll_agg_model_sum_nLL %>% ggplot(aes(x = method, y = model_type, fill = `sum_nLL`)) +
  geom_tile()+
  scale_fill_gradient2(midpoint =3450) +
  labs(x = "", y = "foraging model", title = "domain") +
  theme(aspect.ratio = 1,
        plot.title = element_text(hjust = 0.5, size = rel(1.2)),
        axis.text.x = element_text(angle = 45, hjust = 1))+ facet_grid(Group~domain)+scale_fill_gradient2

```

## Scale for fill is already present.

## Adding another scale for fill, which will replace the existing scale.



```

#Best models across participants
subject_best_models = subject_best_models %>%
  mutate(model_type = fct_recode(foraging_type,
    `pstatic` = "phonologicalstatic",
    `plocal` = "phonologicaldynamiclocal",
    `pglobal` = "phonologicaldynamicglobal",
    `pswitch` = "phonologicaldynamicswitch",
    `static` = "static", `dynamic` = "dynamic", `random` = "random"))

```

```
## Warning: There were 108 warnings in 'mutate()'.
## The first warning was:
## i In argument: 'model_type = fct_recode(...)'.
## i In group 1: 'Group = normal hearing', 'domain = "animals"', 'Subject =
##   "CAF-657"'.
## Caused by warning:
## ! Unknown levels in 'f': phonologicalstatic, phonologicaldynamiclocal, phonologicaldynamicglobal, st
## i Run 'dplyr::last_dplyr_warnings()' to see the 107 remaining warnings.
```

```
subject_best_models %>%
  group_by(Group, domain, model_type) %>%
  count() %>%
  ggplot(aes(x = Group, y = n, group = model_type, fill = model_type)) +
  geom_col() +
  labs(y = "Number of Participants", x = "", fill = "Best Model Type") +
  theme_few() +
  scale_fill_calc() +
  theme(aspect.ratio = 1, legend.position = "right") + facet_wrap(~domain) +
  scale_fill_paletteer_d("nationalparkcolors::Acadia")
```

```
## Scale for fill is already present.
## Adding another scale for fill, which will replace the existing scale.
```

