VR study script

January 24, 2018

1 R setup

The first thing we do is set a random seed for R. This ensures we can reproduce any analysis that has a random component.

```
our_seed <- 123
set.seed(our_seed)
```

We'll use '123' as our seed. Next we need to load the libraries.

```
require(knitr)
require(lme4)

## Loading required package:  lme4
## Loading required package:  Matrix
## Loading required package:  methods

require(afex)

## Loading required package:  afex
## Loading required package:  emmeans
## ************
## Welcome to afex.  For support visit:  http://afex.singmann.science/
## - Functions for ANOVAs:  aov_car(), aov_ez(), and aov_4()
## - Methods for calculating p-values with mixed():  'KR', 'S', 'LRT', and
'PB'
## - 'afex_aov' and 'mixed' objects can be passed to emmeans() for follow-up
tests
## - Get and set global package options with:  afex_options()
## - Set orthogonal sum-to-zero contrasts globally:  set_sum_contrasts()
## - For example analyses see:  browseVignettes("afex")
## ************
##
## Attaching package:  'afex'
## The following object is masked from 'package:lme4':
##
##    lmer

require(cowplot)

## Loading required package:  cowplot
## Loading required package:  ggplot2
##
## Attaching package:  'cowplot'
## The following object is masked from 'package:ggplot2':
##
##    ggsave
```

```r
require(DHARMa)

## Loading required package:  DHARMa

require(xtable)

## Loading required package:  xtable

require(rockchalk)

## Loading required package:  rockchalk

require(tikzDevice)

## Loading required package:  tikzDevice

require(parallel)

## Loading required package:  parallel

require(fifer)

## Loading required package:  fifer
## Loading required package:  MASS
##
## Attaching package:  'MASS'
## The following object is masked from 'package:rockchalk':
##
##    mvrnorm

require(multcomp)

## Loading required package:  multcomp
## Loading required package:  mvtnorm
## Loading required package:  survival
## Loading required package:  TH.data
##
## Attaching package:  'TH.data'
## The following object is masked from 'package:MASS':
##
##    geyser

require(vcdExtra)

## Loading required package:  vcdExtra
## Loading required package:  vcd
## Loading required package:  grid
## Loading required package:  gnm
```

2 Reading the files

We now read in the data files.

```
child.data <- read.csv("vr_data/childrenCSV.csv")
carsac.data <- read.csv("vr_data/selfSacCSV.csv")
sidewalk.data <- read.csv("vr_data/sidewalkCSV.csv")
```

Some of the variables need cleaning, so we rename them.

```
# rename gender variable
colnames(child.data)[2] <- "gender"

# rename driver variable
colnames(child.data)[3] <- "motorist"

# rename levels for driver
child.data$motorist <- factor(child.data$motorist,
                              levels = c("False", "True"),
                              labels = c("human", "self-driving"))

# set participant to factor
child.data$participant.ID <- factor(child.data$participant.ID)

# rename levels for gender
child.data$gender <- factor(child.data$gender,
                            levels = c("False", "True"),
                            labels = c("female", "male"))

# remove sanity check and perception check failures
child.sub <- subset(child.data, child.data$passedSanCheck == "True")
child.sub <- subset(child.sub, !(child.sub$perceivedCar == "Mensch" &
                                 child.sub$motorist == "self-driving"))



# carsac

# rename gender variable
colnames(carsac.data)[2] <- "gender"

# rename driver variable
colnames(carsac.data)[3] <- "motorist"
```

```r
# rename levels for driver
carsac.data$motorist <- factor(carsac.data$motorist,
                            levels = c("False", "True"),
                            labels = c("human", "self-driving"))

carsac.data$participant.ID <- factor(carsac.data$participant.ID)

# rename levels for gender
carsac.data$gender <- factor(carsac.data$gender,
                            levels = c("False", "True"),
                            labels = c("female", "male"))

carsac.sub <- subset(carsac.data, carsac.data$passedSanCheck == "True")
carsac.sub <- subset(carsac.sub, !(carsac.sub$perceivedCar == "Mensch" &
                                   carsac.sub$motorist == "self-driving"))



# sidewalk

# rename gender variable
colnames(sidewalk.data)[2] <- "gender"

# rename driver variable
colnames(sidewalk.data)[3] <- "motorist"

# set participant to factor
sidewalk.data$participant.ID <- factor(sidewalk.data$participant.ID)

# rename levels for driver
sidewalk.data$motorist <- factor(sidewalk.data$motorist,
                            levels = c("False", "True"),
                            labels = c("human", "self-driving"))


# rename levels for gender
sidewalk.data$gender <- factor(sidewalk.data$gender,
                            levels = c("False", "True"),
                            labels = c("female", "male"))

# sanity check
sidewalk.sub <- subset(sidewalk.data, sidewalk.data$passedSanCheck == "True")
sidewalk.sub <- subset(sidewalk.sub, !(sidewalk.sub$perceivedCar == "Mensch" &
```

```
                                         sidewalk.sub$motorist == "self-driving"))
```

3 Manipulation check

We use a χ^2 goodness of fit test to see if perspective affects identification. First we create a subset of only one trial, and collapse the two pedestrian perspectives.

```
manip_check.sub <- subset(child.sub, trial == "largeGroups")

manip_check.sub$perspective <- combineLevels(manip_check.sub$perspective,
                                             levs = c("PedLarge", "PedSmall"),
                                             newLabel = c("Pedestrian"))

## The original levels Observer Passenger PedLarge PedSmall
## have been replaced by Observer Passenger Pedestrian
```

Next we create a contingency table.

```
manip_check.xtab <- xtabs(~perspective + perceivedIden, data = manip_check.sub)
print(xtable(manip_check.xtab))
```

	Beobachter	Fußgänger	Mitfahrer
Observer	30	5	10
Passenger	7	3	34
Pedestrian	33	57	3

Then we compute the test statistic.

```
manip_check.chisq <- chisq.test(manip_check.xtab, simulate.p.value = TRUE)
manip_check.chisq

##
##  Pearson's Chi-squared test with simulated p-value (based on 2000
##  replicates)
##
## data:  manip_check.xtab
## X-squared = 114.01, df = NA, p-value = 0.0004998
```

If the test statistic is significant, we perform follow-up comparisons.

```
alpha <- 0.05
if (manip_check.chisq$p.value <= alpha){
    manip_check.post <- chisq.post.hoc(manip_check.xtab)
    }
```

Adjusted p-values used the fdr method.

```
print(xtable(manip_check.post))
```

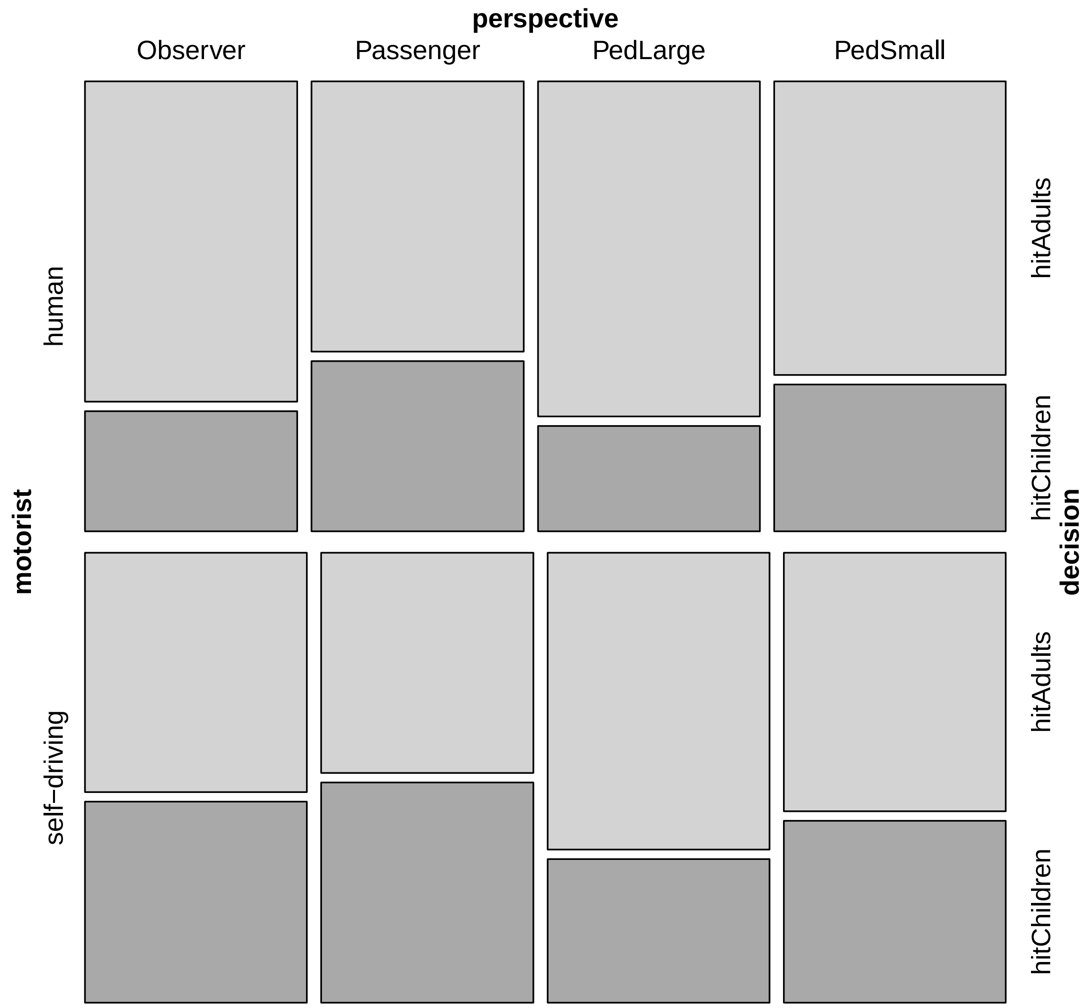
	comparison	raw.p	adj.p
1	Observer vs. Passenger	0.00	0.00
2	Observer vs. Pedestrian	0.00	0.00
3	Passenger vs. Pedestrian	0.00	0.00

4 Child vs adult dilemmas

4.1 Visualising the raw responses

First we look at the raw proportions of the responses.

```
child.xtabs <- xtabs(~ motorist + perspective + decision, data = child.sub)
child.mosaic <- mosaic(child.xtabs,
                       gp = gpar(fill = c("light gray", "dark gray")))
```



4.2 The model

We run a logit mixed model to predict the decision of the participants.

```
(nc <- detectCores())
```

[1] 12

```
cl <- makeCluster(rep("localhost", nc))

child_glmm <- mixed(decision ~ perspective + motorist +
                        perspective:motorist +
```

```
                        trial + gender + #age +
                        (1 | participant.ID),
                    method = "PB", # change to PB for final
                    family = "binomial", data = child.sub,
                    args_test = list(nsim = 1000, cl = cl), cl = cl,
                    control = glmerControl(optimizer = "bobyqa",
                                           optCtrl = list(maxfun = 2e5)))

## Contrasts set to contr.sum for the following variables:  decision, perspective,
motorist, trial, gender, participant.ID
```

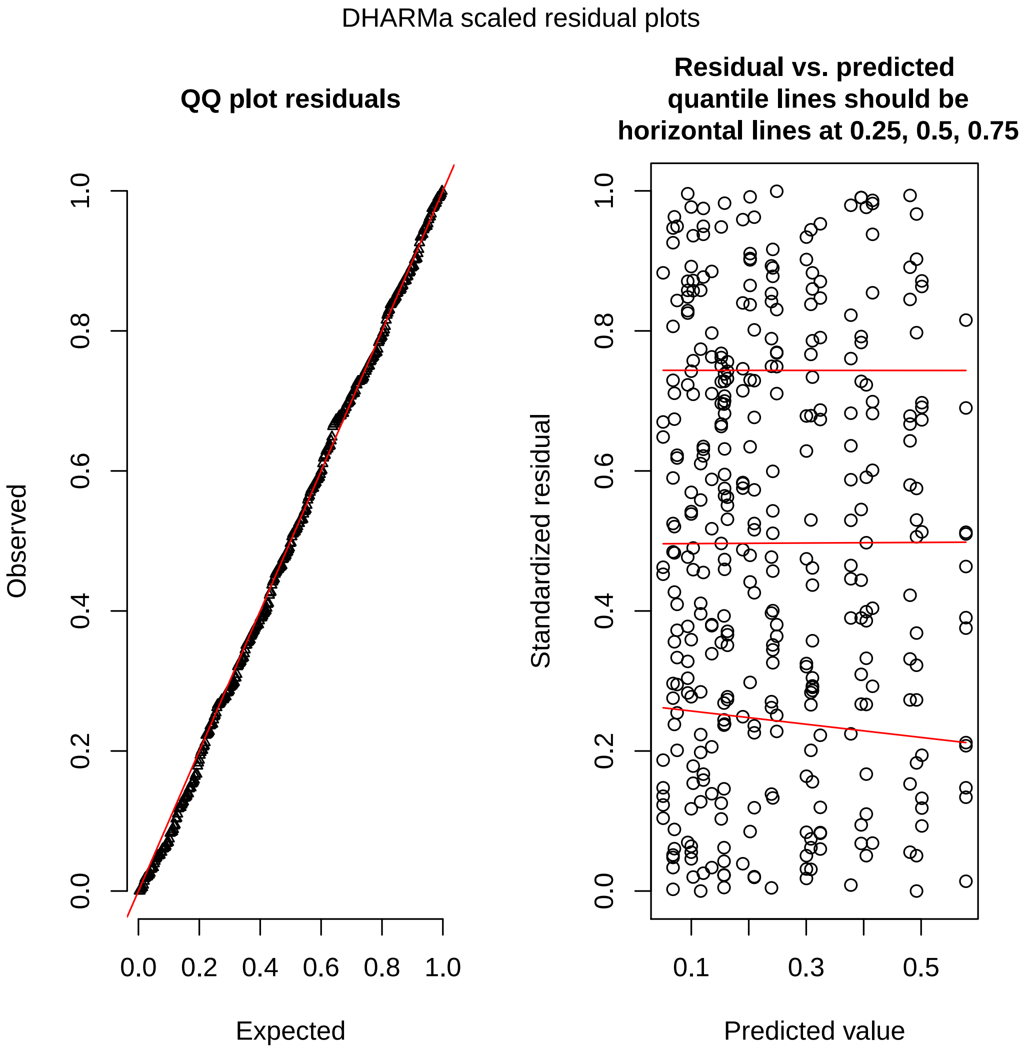
Fitting 6 (g)lmer() models. Obtaining 5 p-values: [.....]

```
stopCluster(cl)
child_glmm_anova <- child_glmm$anova_table
print(xtable(child_glmm_anova))
```

	Chisq	Chi Df	Pr(>Chisq)	Pr(>PB)
perspective	3.90	3	0.2724	0.3986
motorist	4.23	1	0.0398	0.0675
trial	1.21	1	0.2713	0.2714
gender	0.30	1	0.5820	0.6128
perspective:motorist	0.45	3	0.9291	0.9328

We then check the residuals of the model.

```
child_glmm.resid <- simulateResiduals(
    fittedModel = child_glmm$full_model,
    n = 2000)
child_glmm_resid.plot <- plotSimulatedResiduals(
    simulationOutput = child_glmm.resid)
```



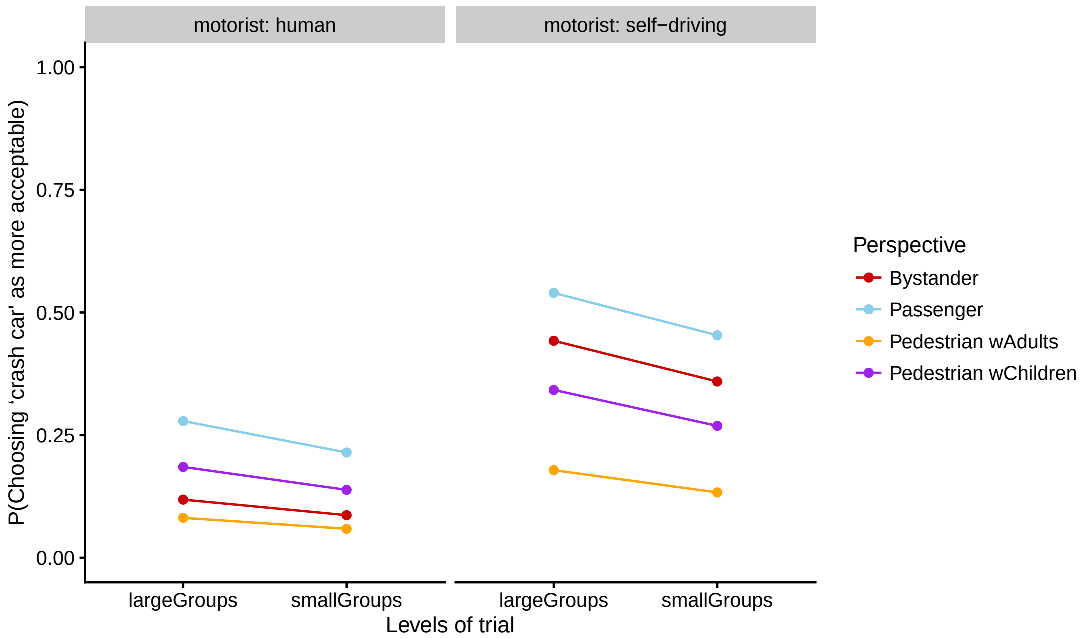
If they look okay, we follow up any significant differences.

4.3 Plotting

Depending on which effects or interactions we find, we probably want to change this plot. It currently just plots the Estimated Marginal Means for the 2-way interaction.

```
child_emm_plot <- emmip(child_glmm, perspective ~ trial | motorist, type = "response") +
    theme_cowplot(font_size = 10) +
    scale_y_continuous(name = "P(Choosing `crash car' as more acceptable)",
                       limits = c(0, 1)) + coord_equal(ratio = 3) +
    scale_color_manual(name = "Perspective",
                       labels = c("Bystander",
```

```
                                        "Passenger",
                                        "Pedestrian wAdults",
                                        "Pedestrian wChildren"),
                          values = c("red3", "skyblue", "orange1", "purple"))
child_emm_plot
```



4.4 Follow-up comparisons

For a good example, see https://osf.io/q2a3b/

If we find an interaction between perspective and motorist-type we should follow that up with comparisons of perspective within each level of motorist.

```r
emm_child_i <- emmeans(child_glmm, ~ perspective | motorist, type = "response",
                       contr = "pairwise")
xtable(emm_child_i$contrasts)
```

contrast	odds.ratio	SE	df	z.ratio	p.value
motorist = human					
Observer / Passenger	0.3479	0.4008	Inf	-0.917	0.7960
Observer / PedLarge	1.5163	1.7108	Inf	0.369	0.9829
Observer / PedSmall	0.5917	0.6627	Inf	-0.469	0.9659
Passenger / PedLarge	4.3590	5.0289	Inf	1.276	0.5782
Passenger / PedSmall	1.7009	1.9212	Inf	0.470	0.9656
PedLarge / PedSmall	0.3902	0.4354	Inf	-0.843	0.8337
motorist = self-driving					
Observer / Passenger	0.6760	0.7553	Inf	-0.350	0.9852
Observer / PedLarge	3.6493	4.0984	Inf	1.153	0.6568
Observer / PedSmall	1.5251	1.6707	Inf	0.385	0.9806
Passenger / PedLarge	5.3985	6.3213	Inf	1.440	0.4743
Passenger / PedSmall	2.2562	2.5623	Inf	0.716	0.8906
PedLarge / PedSmall	0.4179	0.4686	Inf	-0.778	0.8644

Results are averaged over the levels of: trial, gender

P value adjustment: tukey method for comparing a family of 4 estimates

Tests are performed on the log odds ratio scale

If we find no interaction but a significant main effect of perspective, we need to do follow-up multiple comparisons to find out where the difference is.

```r
emm_child_persp <- emmeans(child_glmm, pairwise ~ perspective,
                           type = 'response')

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

xtable(emm_child_persp$contrasts)
```

If we find a main effect of motorist-type, we need to calculate the marginal means to see the effect.

```r
emm_child_motorist <- emmeans(child_glmm, pairwise ~ motorist,
                              type = 'response')

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

xtable(emm_child_motorist$contrasts)
```

contrast	odds.ratio	SE	df	z.ratio	p.value
Observer / Passenger	0.4849	0.3907	Inf	-0.898	0.8057
Observer / PedLarge	2.3523	1.8836	Inf	1.068	0.7089
Observer / PedSmall	0.9499	0.7427	Inf	-0.066	0.9999
Passenger / PedLarge	4.8510	4.0801	Inf	1.878	0.2377
Passenger / PedSmall	1.9590	1.5768	Inf	0.835	0.8376
PedLarge / PedSmall	0.4038	0.3218	Inf	-1.138	0.6661

Results are averaged over the levels of: motorist, trial, gender

P value adjustment: tukey method for comparing a family of 4 estimates

Tests are performed on the log odds ratio scale

contrast	odds.ratio	SE	df	z.ratio	p.value
human / self-driving	0.3158	0.1869	Inf	-1.948	0.0514

Results are averaged over the levels of: perspective, trial, gender

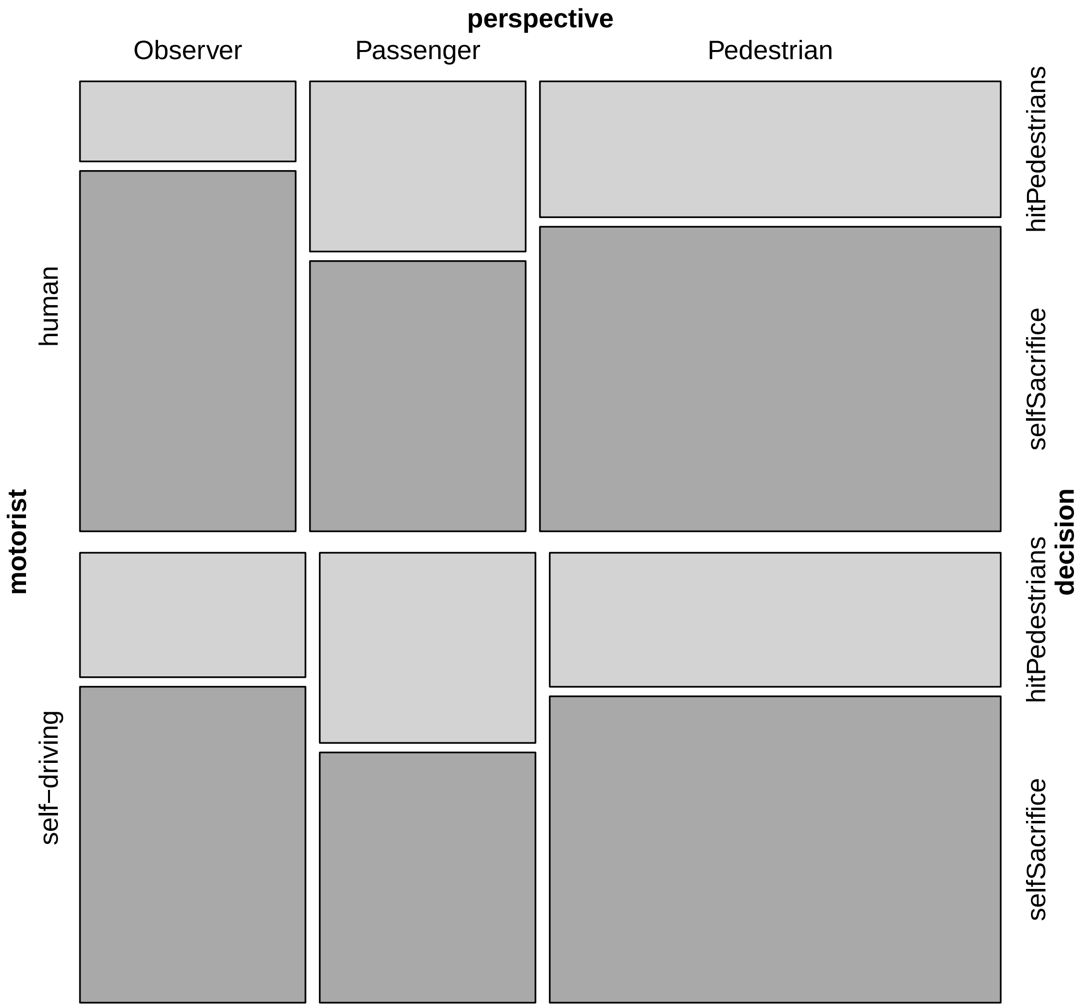
Tests are performed on the log odds ratio scale

5 Car occupants vs pedestrians dilemma

We collapse both pedestrian perspectives for this dilemma.

```
carsac.sub$perspective <- combineLevels(carsac.sub$perspective,
                                        levs=c("PedLarge", "PedSmall"),
                                        newLabel = "Pedestrian")

## The original levels Observer Passenger PedLarge PedSmall
## have been replaced by Observer Passenger Pedestrian
```

```
carsac.xtabs <- xtabs(~ motorist + perspective + decision, data = carsac.sub)
carsac.mosaic <- mosaic(carsac.xtabs,
                       gp = gpar(fill = c("light gray", "dark gray")))
```



5.1 The model

```
(nc <- detectCores())
```

```
[1] 12
```

```
cl <- makeCluster(rep("localhost", nc))

carsac_glmm <- mixed(decision ~ perspective + motorist +
                        perspective:motorist +
                        trial + gender + #age +
```

```
                    (1 | participant.ID),
                method = "PB", # change to PB for final
                family = "binomial", data = carsac.sub,
                args_test = list(nsim = 1000, cl = cl), cl = cl,
                control = glmerControl(optimizer = "bobyqa",
                                       optCtrl = list(maxfun = 2e5)))

## Contrasts set to contr.sum for the following variables:  decision, perspective,
motorist, trial, gender, participant.ID
```

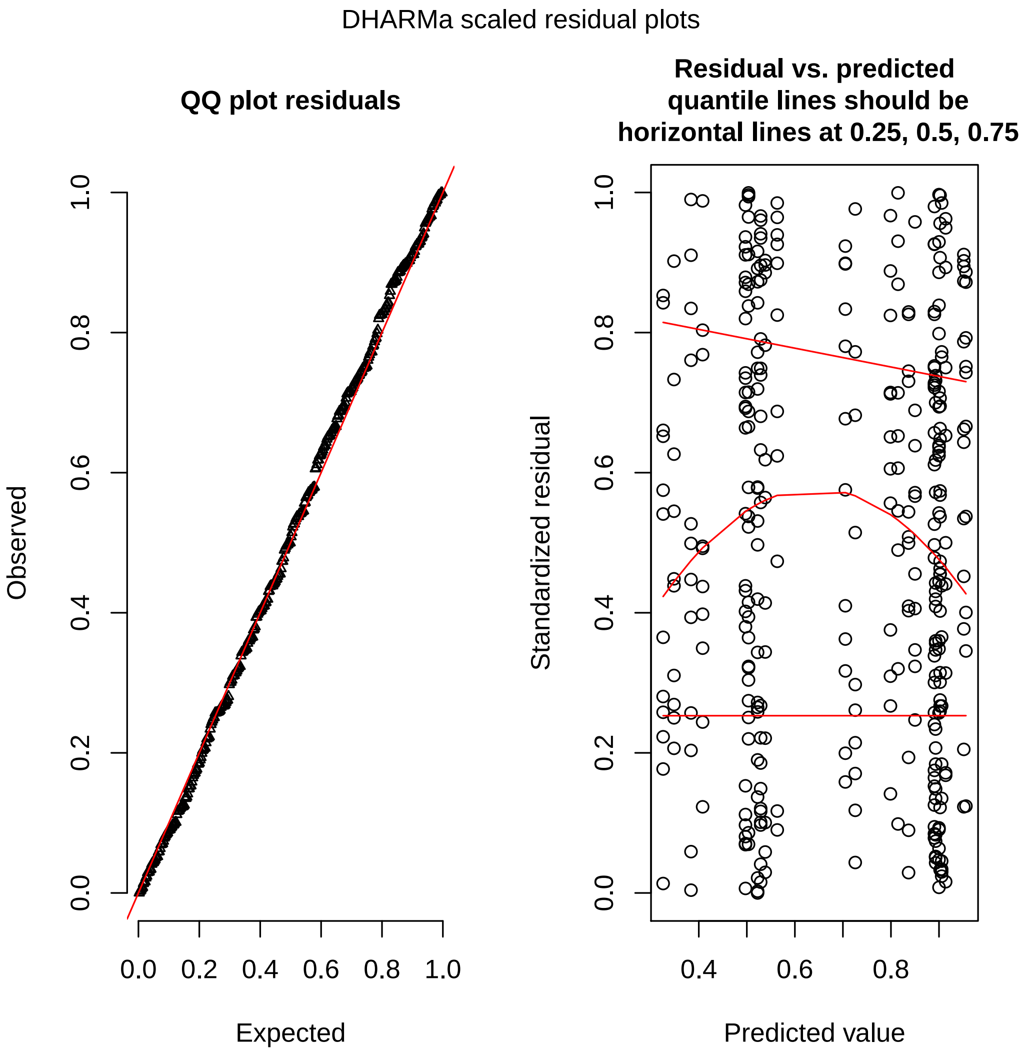
Fitting 6 (g)lmer() models. Obtaining 5 p-values: [.....]

```
stopCluster(cl)
xtable(carsac_glmm$anova_table)
```

	Chisq	Chi Df	Pr(>Chisq)	Pr(>PB)
perspective	7.50	2	0.0235	0.0311
motorist	1.08	1	0.2994	0.3102
trial	61.24	1	0.0000	0.0010
gender	0.13	1	0.7228	0.7472
perspective:motorist	1.05	2	0.5903	0.6148

Then we check the residuals.

```
carsac_glmm.resid <- simulateResiduals(
    fittedModel = carsac_glmm$full_model, n = 2000)
carsac_glmm_resid.plot <- plotSimulatedResiduals(
    simulationOutput = carsac_glmm.resid)
```

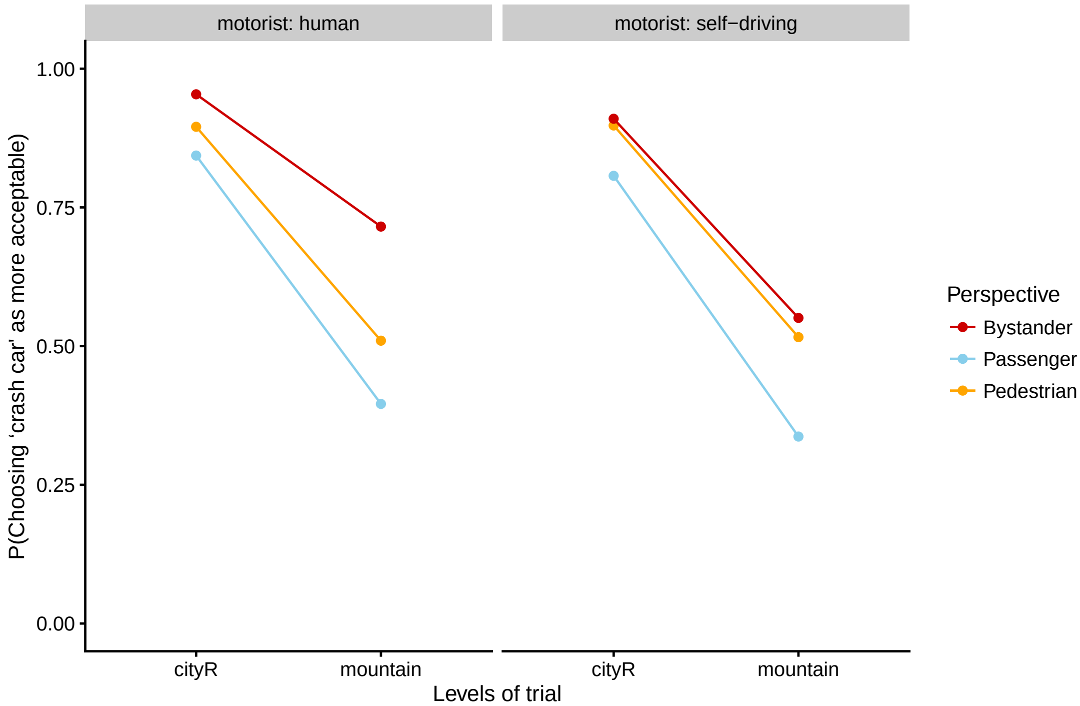


5.2 Plotting

Depending on which effects or interactions we find, we probably want to change this plot. It currently just plots the Estimated Marginal Means for the 2-way interaction.

```
carsac.plot <- emmip(carsac_glmm, perspective ~ trial | motorist, type = "response") +
    theme_cowplot(font_size = 10) +
    scale_y_continuous(name = "P(Choosing `crash car' as more acceptable)",
                       limits = c(0, 1)) + coord_equal(ratio = 3) +
    scale_color_manual(name = "Perspective",
                       labels = c("Bystander",
                                  "Passenger",
```

```
                                          "Pedestrian"),
                                   values = c("red3", "skyblue", "orange1" , "purple"))

carsac.plot
```



5.3 Follow-up analyses

If we find an interaction between perspective and motorist-type we should follow that up with comparisons of perspective within each level of motorist.

```r
emm_carsac_i <- emmeans(carsac_glmm, pairwise ~ perspective | motorist,
                        type = 'response')
xtable(emm_carsac_i$contrasts)
```

contrast	odds.ratio	SE	df	z.ratio	p.value
motorist = human					
Observer / Passenger	3.8387	2.3773	Inf	2.172	0.0760
Observer / Pedestrian	2.4179	1.3164	Inf	1.622	0.2364
Passenger / Pedestrian	0.6299	0.3037	Inf	-0.959	0.6032
motorist = self-driving					
Observer / Passenger	2.4128	1.3701	Inf	1.551	0.2672
Observer / Pedestrian	1.1494	0.5673	Inf	0.282	0.9571
Passenger / Pedestrian	0.4764	0.2310	Inf	-1.529	0.2772

Results are averaged over the levels of: trial, gender

P value adjustment: tukey method for comparing a family of 3 estimates

Tests are performed on the log odds ratio scale

If we find no interaction but a significant main effect of perspective, we need to do follow-up multiple comparisons to find out where the difference is.

```r
emm_carsac_persp <- emmeans(carsac_glmm, pairwise ~ perspective,
                            type = 'response')

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

xtable(emm_carsac_persp$contrasts)
```

contrast	odds.ratio	SE	df	z.ratio	p.value
Observer / Passenger	3.0433	1.3025	Inf	2.600	0.0252
Observer / Pedestrian	1.6671	0.6142	Inf	1.387	0.3475
Passenger / Pedestrian	0.5478	0.1889	Inf	-1.745	0.1885

Results are averaged over the levels of: motorist, trial, gender

P value adjustment: tukey method for comparing a family of 3 estimates

Tests are performed on the log odds ratio scale

If we find a main effect of motorist-type, we need to calculate the marginal means to see the effect.

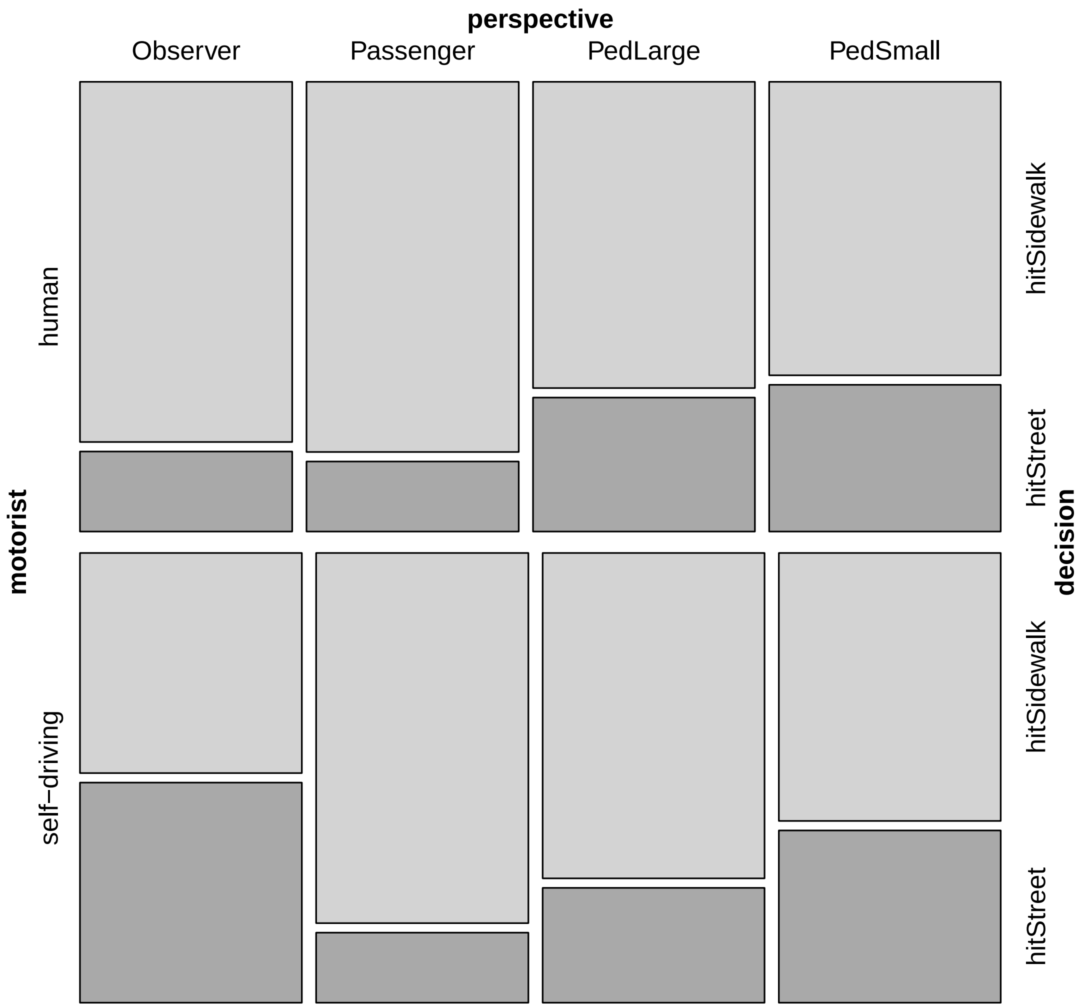
```r
emm_carsac_motorist <- emmeans(carsac_glmm, pairwise ~ motorist,
                               type = 'response')

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

xtable(emm_carsac_motorist$contrasts)
```

contrast	odds.ratio	SE	df	z.ratio	p.value
human / self-driving	1.3710	0.4191	Inf	1.032	0.3020

Results are averaged over the levels of: perspective, trial, gender
Tests are performed on the log odds ratio scale

6 Sidewalk vs road dilemma

```
sidewalk.xtabs <- xtabs(~ motorist + perspective + decision, data = sidewalk.sub)
sidewalk.mosaic <- mosaic(sidewalk.xtabs,
                          gp = gpar(fill = c("light gray", "dark gray")))
```



6.1 The model

```
(nc <- detectCores())
```

[1] 12

```
cl <- makeCluster(rep("localhost", nc))

sidewalk_glmm <- mixed(decision == 'hitSidewalk' ~ perspective +
                        motorist +
                        perspective:motorist +
                        trial + gender + #age +
                        (1 | participant.ID),
                    method = "PB", # change to PB for final
                    family = "binomial", data = sidewalk.sub,
                    args_test = list(nsim = 1000, cl = cl), cl = cl,
                    control = glmerControl(optimizer = "bobyqa",
                                           optCtrl = list(maxfun = 2e5)))

## Contrasts set to contr.sum for the following variables:  decision, perspective,
motorist, trial, gender, participant.ID
```

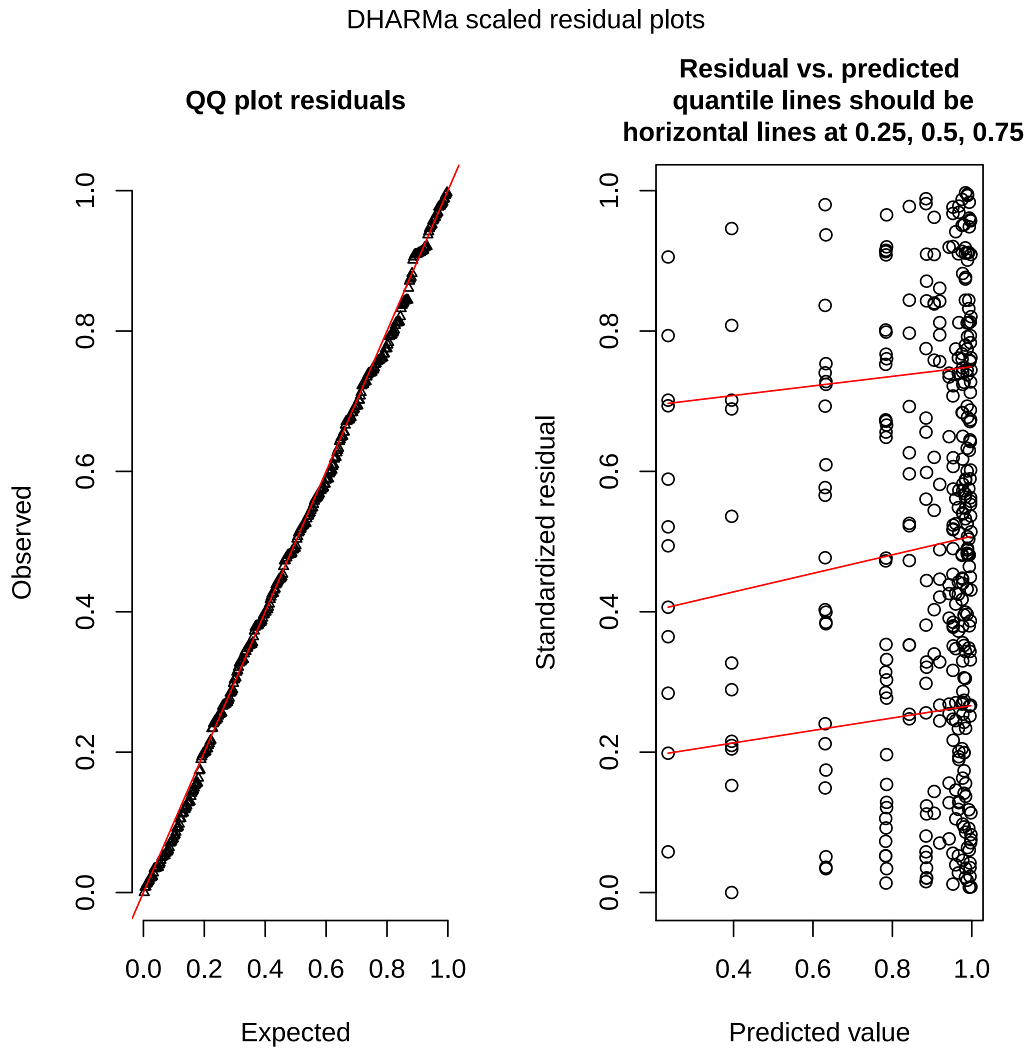
Fitting 6 (g)lmer() models. Obtaining 5 p-values: [.....]

```
stopCluster(cl)
xtable(sidewalk_glmm$anova_table)
```

	Chisq	Chi Df	Pr(>Chisq)	Pr(>PB)
perspective	7.93	3	0.0475	0.0526
motorist	2.12	1	0.1455	0.2086
trial	4.00	1	0.0455	0.0909
gender	4.90	1	0.0268	0.0058
perspective:motorist	6.64	3	0.0842	0.0536

Then we check the residuals.

```
sidewalk_glmm.resid <- simulateResiduals(
    fittedModel = sidewalk_glmm$full_model, n = 2000)
sidewalk_glmm_resid.plot <- plotSimulatedResiduals(
    simulationOutput = sidewalk_glmm.resid)
```

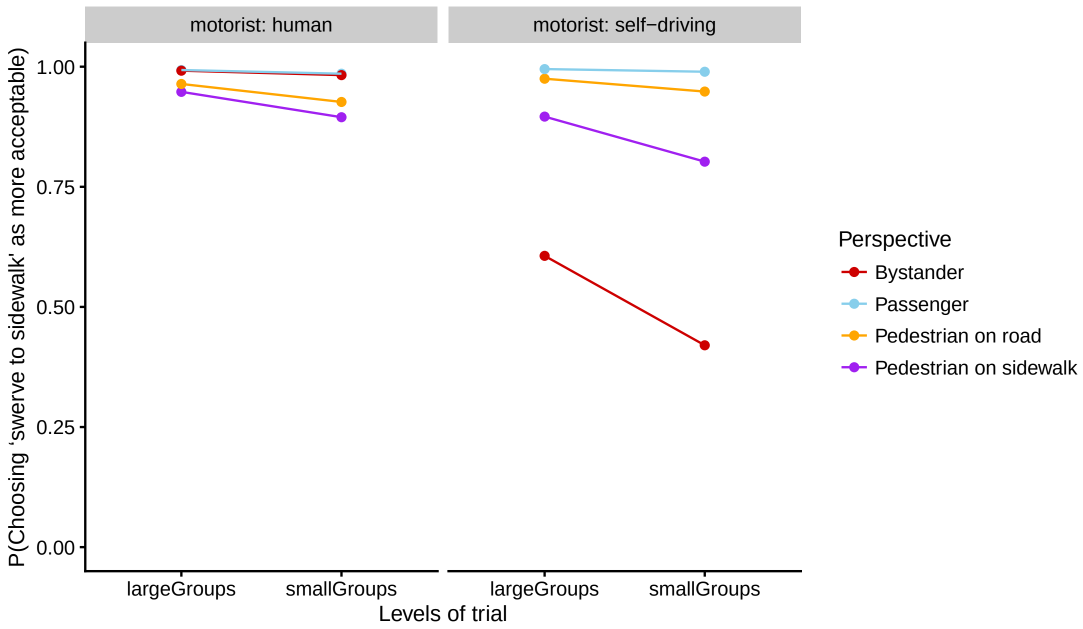


6.2 Plotting

Depending on which effects or interactions we find, we probably want to change this plot. It currently just plots the Estimated Marginal Means for the 2-way interaction.

```
sidewalk.plot <- emmip(sidewalk_glmm, perspective ~ trial | motorist,
                       type = "response") +
    theme_cowplot(font_size = 10) +
    scale_y_continuous(
        name = "P(Choosing `swerve to sidewalk' as more acceptable)",
                       limits = c(0, 1)) + coord_equal(ratio = 3) +
    scale_color_manual(name = "Perspective",
```

```
                             labels = c("Bystander",
                                        "Passenger",
                                        "Pedestrian on road",
                                        "Pedestrian on sidewalk"),
                             values = c("red3", "skyblue",
                                        "orange1", "purple"))

sidewalk.plot
```



6.3 Follow-up analyses

If we find an interaction between perspective and motorist-type we should follow that up with comparisons of perspective within each level of motorist.

```
emm_sidewalk_i <- emmeans(sidewalk_glmm, pairwise ~ perspective | motorist,
                          type = 'response')
xtable(emm_sidewalk_i$contrasts)
```

contrast	odds.ratio	SE	df	z.ratio	p.value
motorist = human					
Observer / Passenger	0.8300	1.1971	Inf	0.693	0.8997
Observer / PedLarge	4.4380	6.5893	Inf	0.674	0.9071
Observer / PedSmall	6.5827	9.7895	Inf	0.672	0.9075
Passenger / PedLarge	5.3471	7.9321	Inf	0.674	0.9069
Passenger / PedSmall	7.9310	11.7982	Inf	0.672	0.9076
PedLarge / PedSmall	1.4832	2.2475	Inf	0.660	0.9121
motorist = self-driving					
Observer / Passenger	0.0077	0.0151	Inf	0.510	0.9568
Observer / PedLarge	0.0395	0.0721	Inf	0.549	0.9469
Observer / PedSmall	0.1785	0.3171	Inf	0.563	0.9430
Passenger / PedLarge	5.1422	7.5438	Inf	0.682	0.9041
Passenger / PedSmall	23.2135	37.5966	Inf	0.617	0.9266
PedLarge / PedSmall	4.5143	7.0392	Inf	0.641	0.9186

Results are averaged over the levels of: trial, gender

Unknown transformation "==": no transformation done

P value adjustment: tukey method for comparing a family of 4 estimates

If we find no interaction but a significant main effect of perspective, we need to do follow-up multiple comparisons to find out where the difference is.

```
emm_sidewalk_persp <- emmeans(sidewalk_glmm, pairwise ~ perspective,
                              type = 'response')

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

xtable(emm_sidewalk_persp$contrasts)
```

If we find a main effect of motorist-type, we need to calculate the marginal means to see the effect.

```
emm_sidewalk_motorist <- emmeans(sidewalk_glmm, pairwise ~ motorist,
                                 type = 'response')
```

contrast	odds.ratio	SE	df	z.ratio	p.value
Observer / Passenger	0.0799	0.0977	Inf	0.818	0.8462
Observer / PedLarge	0.4189	0.4855	Inf	0.863	0.8240
Observer / PedSmall	1.0841	1.2186	Inf	0.890	0.8103
Passenger / PedLarge	5.2436	5.5014	Inf	0.953	0.7760
Passenger / PedSmall	13.5686	15.2436	Inf	0.890	0.8100
PedLarge / PedSmall	2.5876	2.8321	Inf	0.914	0.7976

Results are averaged over the levels of: motorist, trial, gender

Unknown transformation "==": no transformation done

P value adjustment: tukey method for comparing a family of 4 estimates

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

xtable(emm_sidewalk_motorist$contrasts)
```

contrast	odds.ratio	SE	df	z.ratio	p.value
human / self-driving	2.9910	2.4391	Inf	1.226	0.2201

Results are averaged over the levels of: perspective, trial, gender

Unknown transformation "==": no transformation done