Target Absent Stopping Rules descriptives

```
df$TATP <- 0

# Say when it's TA and when it's TP
df$TATP[df$key == "l"] <- "TP"
df$TATP[df$key == "r"] <- "TA"
```

```
total_observations
```

```
## [1] 34950
```

```
trials_per_participant = mean(number_of_trials$num_trials)
trials_per_participant
```

```
## [1] 50
```

Data collect from Essex for the SIBL group was removed as this data had some issues with block length and also the presence of a "difficulty" level that wasn't included in the study as it was deemed too difficult.

Each participant carried out 50 trials for each difficulty and in both conditions they took part in. Some responses were removed as participants pressed the wrong key.

The rt values in these were replaced with NA as the wrong key had been pressed. Also, as we were interested in looking at the influence of previous rt's we also removed the first trial from each block as this trial had no previous trial information.

Prior to removing these trials, there were 34950 trials across participants. After removing cases with NA values, we were left with 33321 trials across all participants (about 95% of the data). On average, this left participants with 48 trials per block, condition, and difficulty.

```
total_observations_nar
```

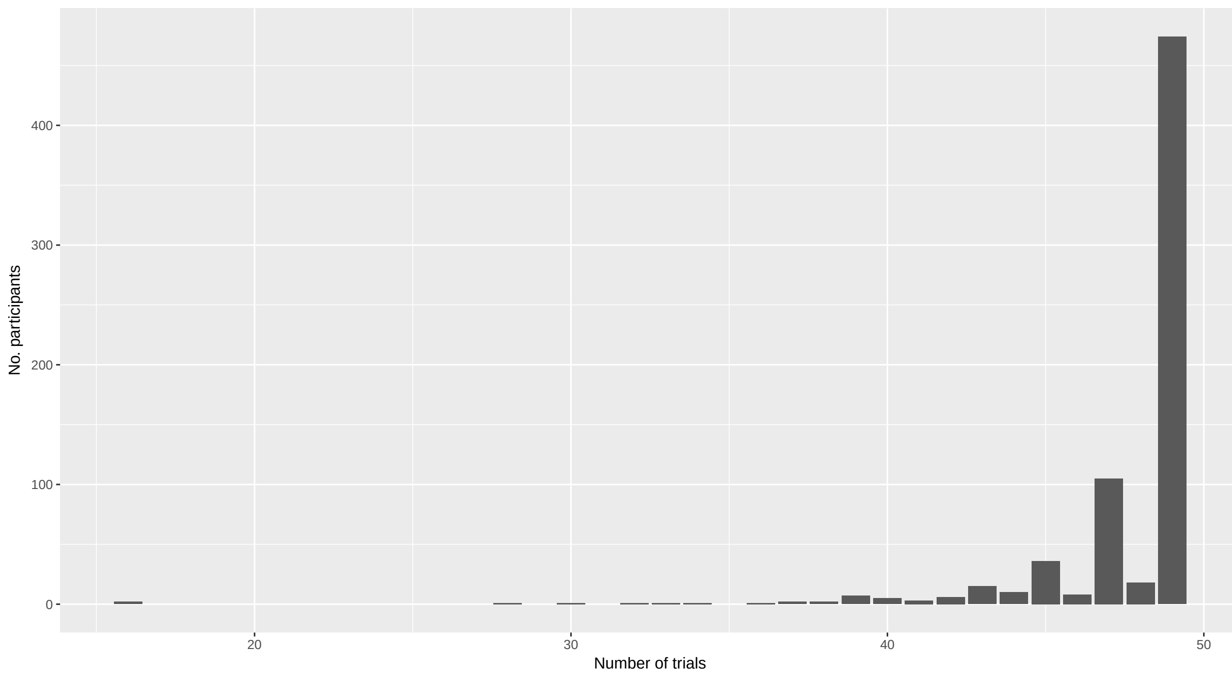
```
## [1] 33321
```

```
trials_per_participant_nar
```

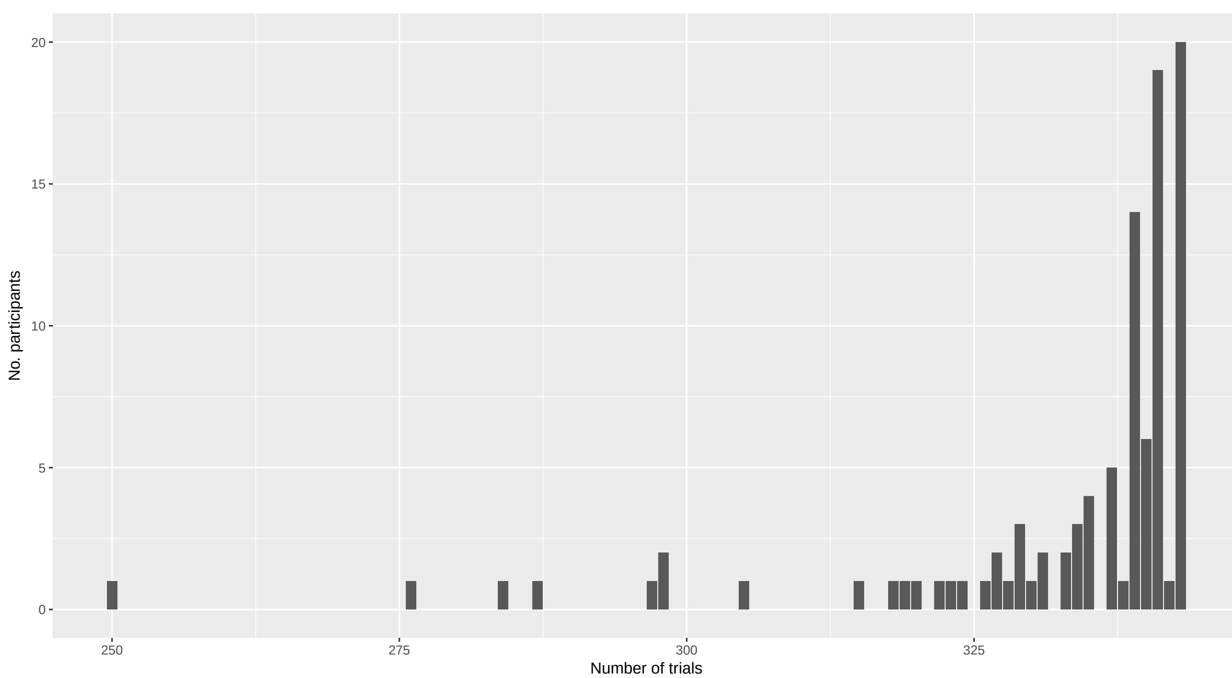
```
## [1] 47.66953
```

We can also plot how many participants carried out "x" amount of trials per block. This plot can be seen below as well as a plot containing information about how many trials in total were carried out and how many participants had this many total trials.

```
trials_per_block_plt
```

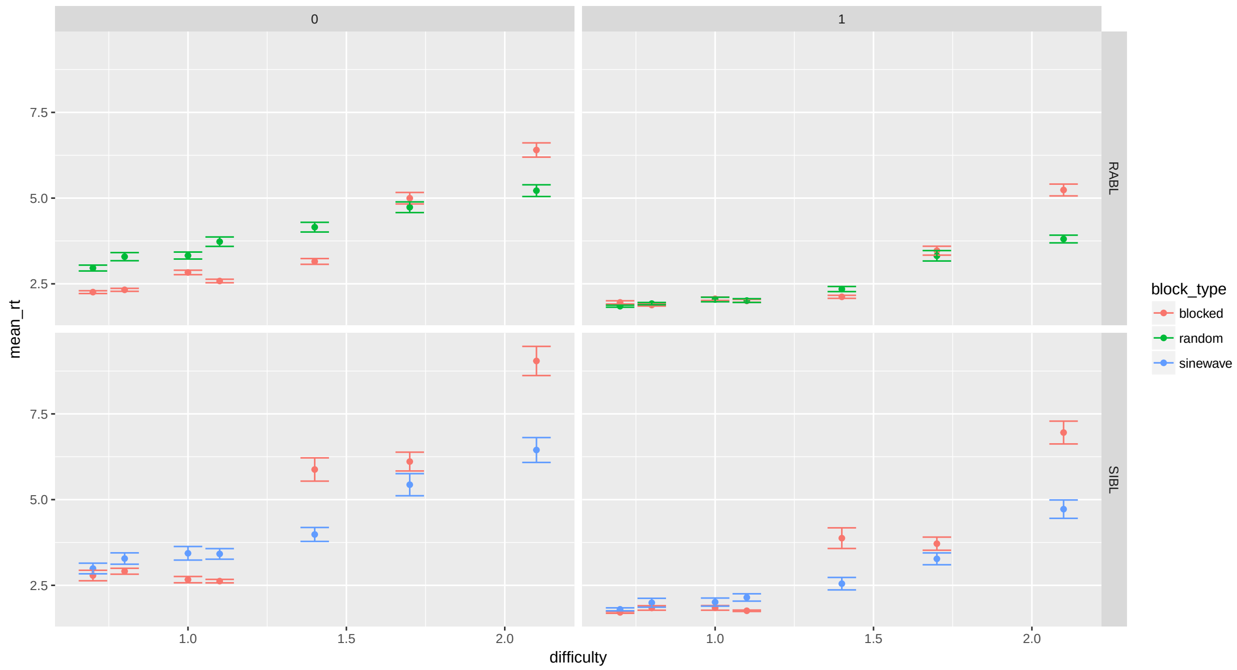



```
trials_total_plt
```

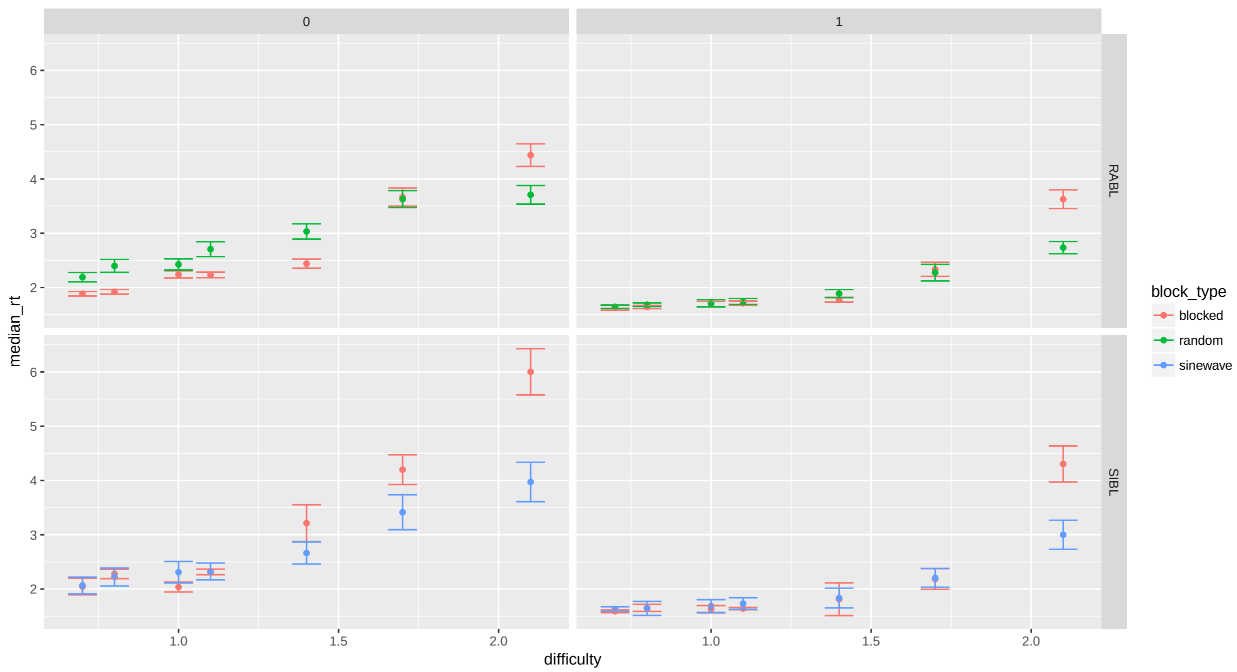


Below are plots of the mean and median reaction times separated by group (either RABL or SIBL) and also whether the target was present or absent (1 or 0 respectively). The first graph is the mean reaction times with standard error bars, the second is of the median reaction times. Along the x-axis is the “difficulty” of the trial. Higher numbers mean more variance in the distractors and therefore a more difficult trial. The number relates to how much the distractor line segments could vary about an orientation that was orthogonal to the target item.

```
mean_rt_plot
```

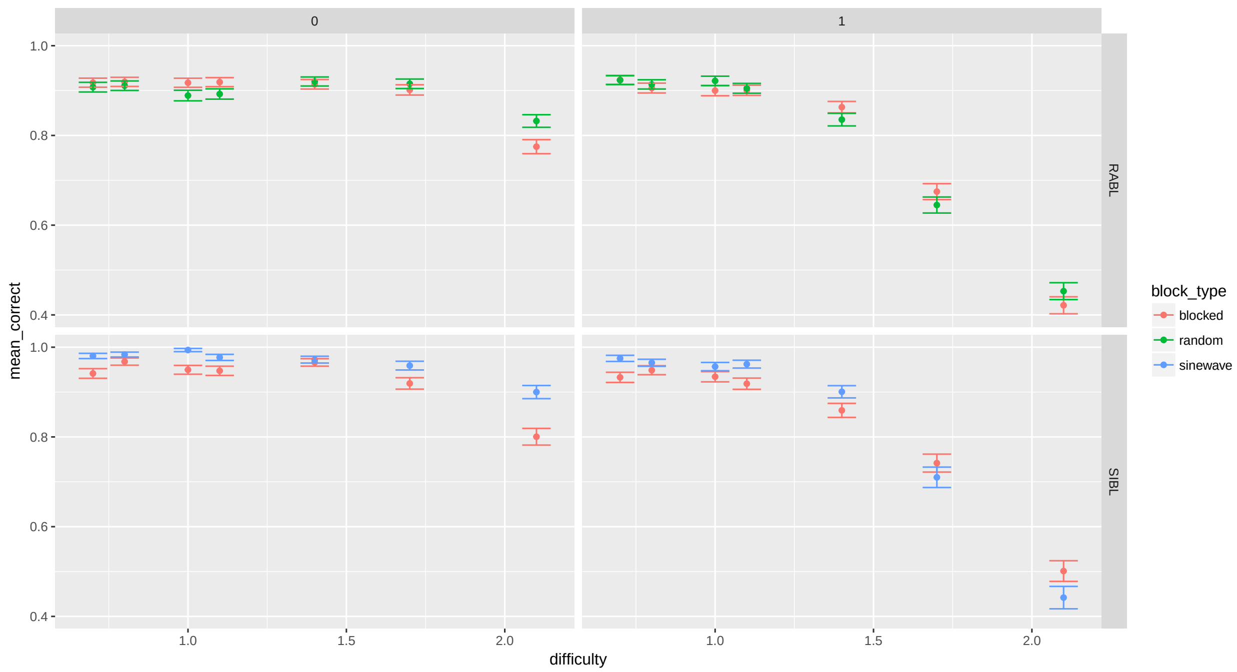



```
median_rt_plot
```



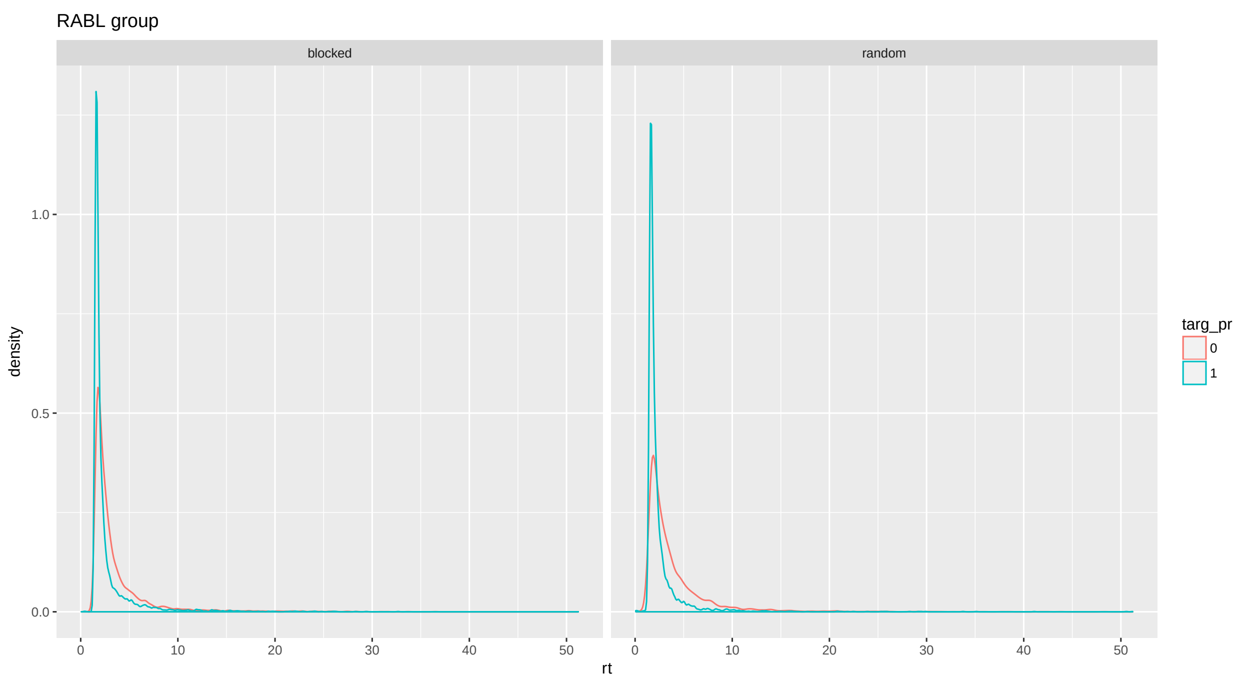
Below are the plots of mean accuracy over different difficulties, block types, and target presence.

```
mean_acc_plot
```

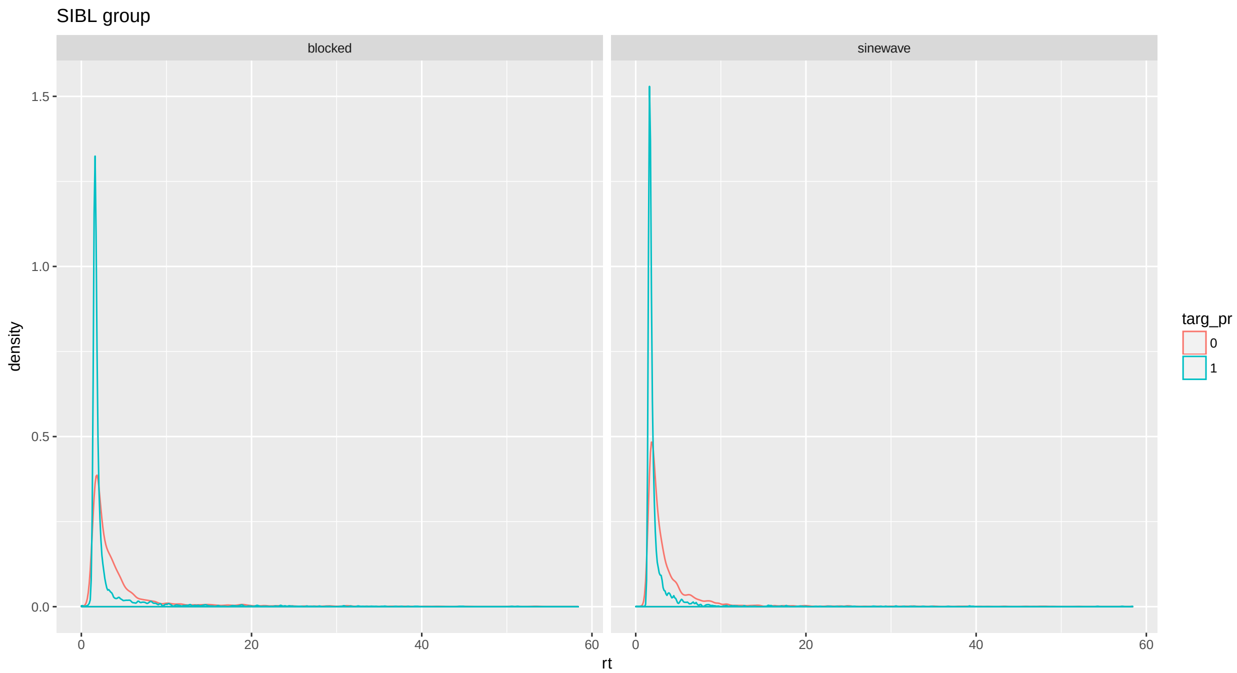


These are the density plots for Target absent and target present responses across groups and the block type.

```
density_plot_RABL
```

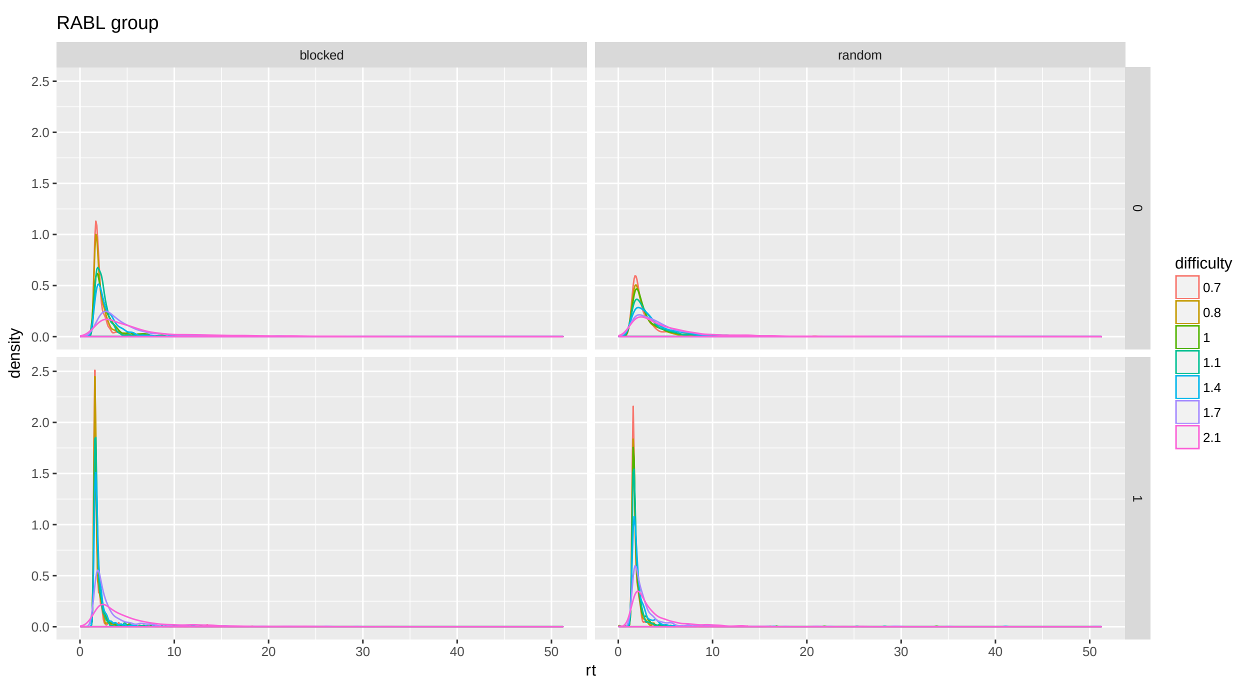



```
density_plot_SIBL
```

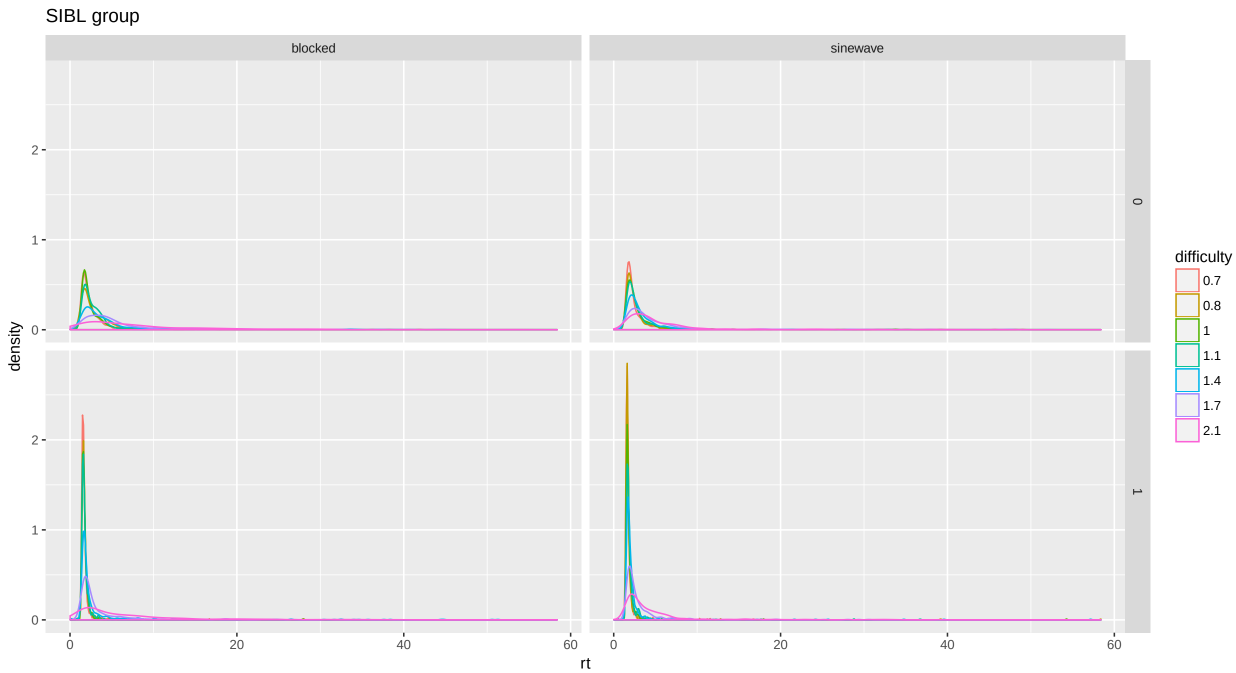


These are the plots when we distinguish across the different difficulty levels. For the purpose of these graphs, difficulty was treated as a "discrete" factor, but for the modelling this should be a continuous variable instead. I created a different data frame for these graphs so the main data frame (df) should remain unchanged.

```
density_plot_RABL_diff
```

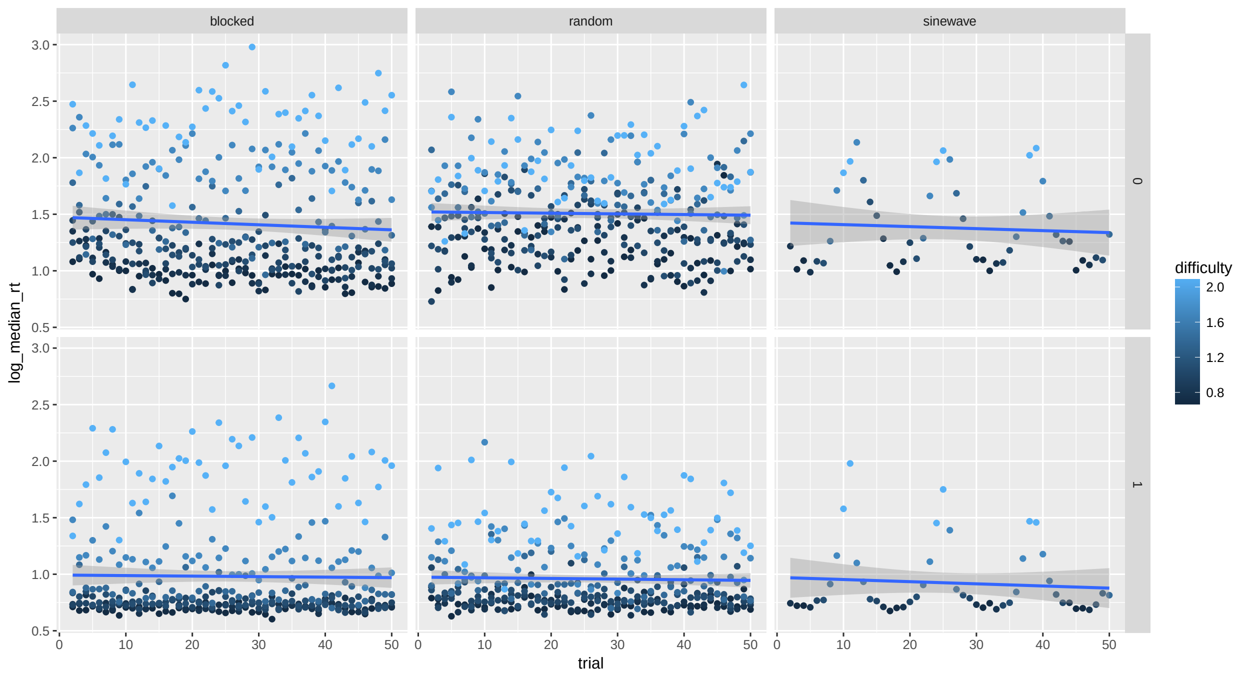



```
density_plot_SIBL_diff
```



We took a look at trial number (across blocks) to look at whether there were any practice effects that may need to be taken account of when creating a model of the data.

```
temp = group_by(df, difficulty, block_type, targ_pr, trial)
trial_desc_stats = summarise(temp, mean_rt = mean(rt),
                             median_rt = median(rt),
                             sdev = sd(rt),
                             N = length(rt),
                             se = sdev/sqrt(N),
                             upper_mean = mean_rt + se,
                             lower_mean = mean_rt - se,
                             upper_median = median_rt + se,
                             lower_median = median_rt - se,
                             # log mean
                             log_mean_rt = mean(log2(rt)),
                             log_sdev = sd(log2(rt)),
                             log_se = log_sdev/sqrt(N),
                             log_mean_upper = log_mean_rt + log_se,
                             log_mean_lower = log_mean_rt - log_se,
                             # log median
                             log_median_rt = median(log2(rt)),
                             log_median_upper = log_median_rt + log_se,
                             log_median_lower = log_median_rt - log_se)
# tidy
rm(temp)

trial_plt = ggplot(trial_desc_stats, aes(trial, log_median_rt, colour = difficulty))
trial_plt = trial_plt + geom_point()
trial_plt = trial_plt + geom_smooth(method = lm)
trial_plt = trial_plt + facet_grid(targ_pr~block_type)

trial_plt
```



Modelling data creation

Below is a record of what happened for the modelling of the data. Every step taken to produce each model and the results of each can be seen here.

Initially, we centred all contniuos predictors according to the grand mean

```
# centre p_rt
df$c_p_rt <- df$p_rt - mean(df$p_rt)
```

Difficulty was scaled by dividing everything by pi. This means that 0 is the easiest difficulty (no variance in the distractors) and 1 is virtually impossible (the distractors vary through 360$deg;)

Difficulty was also changed to be called theta

```
# change difficulty name
#colnames(df)[7] <- "theta"

#rescale difficulty for the models
df$theta <- round(df$difficulty/pi, digits = 3)

# create a centred version
df$theta_c <- df$theta - mean(df$theta)
```

We created dummy variables for the contrasts between block_type. Blocked being 0

```
df$isra <- ifelse(df$block_type == "random", 1,0)

df$issi <- ifelse(df$block_type == "sinewave",1,0)
```

Need to fix participant so it is a continuous string of numbers, otherwise R gets confused

```
df$participant <- as.factor(df$participant)
df$participant <- as.numeric(df$participant)
```

We also created a dataset just to look at correct trials.

```
df_correct_only <- df[df$correct == 1,]
```

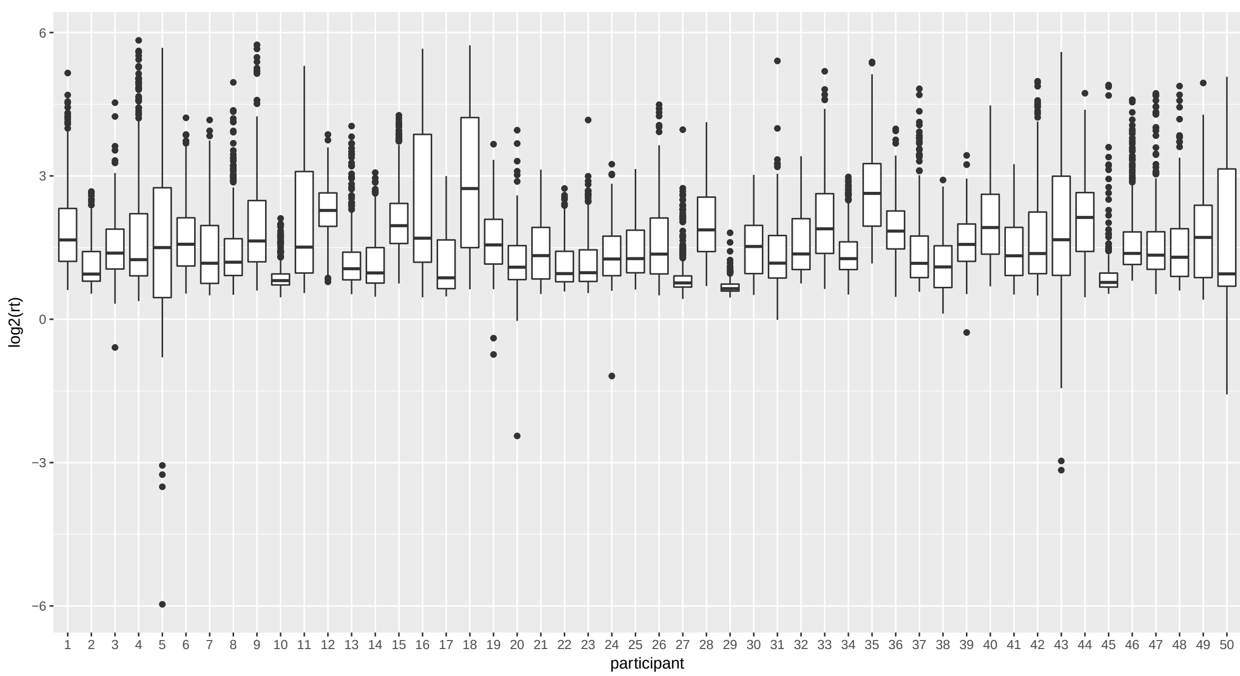
Working on TA trials only

As target absent judegemnts were our primary interest, we decided to create a dataset of only "Target Absent" judgements. The thought process is that making a Target Present judgement is due to detecting the target, whereas we are interested in how people make the decision that the target is not there.

```
# subset the data
df_TA <- df[df$TATP == "TA",]

#tidy up
rm(df, df_correct_only)

# save this
save(df_TA, file = "scratch/processed_data_nar_TA.rda")
```

Create some box plots of participants RTs for TA trials only

```
df_TA_bxplt <- df_TA
df_TA_bxplt$participant <- as.factor(df_TA_bxplt$participant)

boxplt <- ggplot(df_TA_bxplt, aes(x = participant, y = log2(rt)))
boxplt <- boxplt + geom_boxplot()

boxplt
```



Notes

Internal numbering

Rethinking's internal numbering is like 1,10,11,etc. so this needs to be considered when extracting information from the models. For example, when looking at the intercepts, a_p[2] actually corresponds to participant 10, not 2. This fix has been added to the modelling scripts where this is important.

Dummy Variable coding

This was used when looking at block type. For all models, "isra" refers to trial in the random condition, "issi" refers to trials in the sine wave condition.

Informed priors

We can use the data from Anna's study to get a feel for what the intercept and slope should be in our model.

Remembering that we are operating on a log scale, we need to change the values so they are appropriate for the model.

Anna's data suggests that at the easiest difficulty, the mean was 1.75 (sd = 0.13) for the easiest trials. For the hard trials it was 3.94 (SD = 2.19).

For our model, we should set the intercept value (or a in the model) to be the log(1.75) which is 0.56. As for the σ component of $a \sim \text{dnorm}(0.56, \sigma)$, this can be set to 1 as this gives a resonable range for RTs.