

# Analysis of learning via the IAT using race stimuli

*Ian Hussey*

## Contents

Hypotheses . . . . .	1
<b>Descriptive statistics</b>	<b>4</b>
All participants . . . . .	4
Sample descriptive statistics . . . . .	5
<b>H1: Differences in IAT effects between contrast pair conditions</b>	<b>5</b>
Outlier removal . . . . .	5
Descriptive statistics . . . . .	6
Plots . . . . .	6
Preregistered hypothesis test . . . . .	10
Alternative method . . . . .	11
<b>H2: Differences in self-reported ratings between contrast pair conditions</b>	<b>15</b>
Descriptive statistics . . . . .	15
Plots . . . . .	15
Preregistered hypothesis test . . . . .	18

## Hypotheses

H1: Completing an IAT serves to train attitudes as well as test them. Participants who complete a race IAT will demonstrate more negative implicit evaluations of the outgroup (black people) on the SCIAT than participants who completed a control (flowers-insects) IAT.

H2: Completing an IAT serves to train attitudes as well as test them. Participants who complete a race IAT will demonstrate more negative self-reported evaluations of the outgroup (black people) on the ratings than participants who completed a control (flowers-insects) IAT.

TO DO: alternative IAT (D1) analysis should include modern racism scale and then plot marginal means.

```
# dependencies
```

```
library(tidyverse)
```

```
## Loading tidyverse: ggplot2
```

```
## Loading tidyverse: tibble
```

```
## Loading tidyverse: tidyr
```

```
## Loading tidyverse: readr
```

```
## Loading tidyverse: purrr
```

```
## Loading tidyverse: dplyr
```

```
## Conflicts with tidy packages -----
```

```
## filter(): dplyr, stats
```

```
## lag():    dplyr, stats
```

```
library(psych)
```

```
##
```

```
## Attaching package: 'psych'
```

```

## The following objects are masked from 'package:ggplot2':
##
##      %+%, alpha
library(afex)
## Loading required package: lme4
## Loading required package: Matrix
##
## Attaching package: 'Matrix'
## The following object is masked from 'package:tidyr':
##
##      expand
## Loading required package: reshape2
##
## Attaching package: 'reshape2'
## The following object is masked from 'package:tidyr':
##
##      smiths
## Loading required package: lsmeans
## Loading required package: estimability
## *****
## Welcome to afex. Important changes in the current version:
## - Functions for ANOVAs have been renamed to: aov_car(), aov_ez(), and aov_4().
## - ANOVA functions return an object of class 'afex_aov' as default, see: ?aov_car
## - 'afex_aov' objects can be passed to lsmeans for contrasts and follow-up tests.
## - Reset previous (faster) behavior via: afex_options(return_aov='nice')
## - Many more arguments can now be set globally via options, see: afex_options()
## *****
library(effsize)
library(weights) # for rd(), a round() alternative
## Loading required package: Hmisc
## Loading required package: lattice
## Loading required package: survival
## Loading required package: Formula
##
## Attaching package: 'Hmisc'
## The following object is masked from 'package:psych':
##
##      describe
## The following objects are masked from 'package:dplyr':
##
##      combine, src, summarize
## The following objects are masked from 'package:base':
##

```

```

##      format.pval, round.POSIXt, trunc.POSIXt, units
## Loading required package: gdata
## gdata: read.xls support for 'XLS' (Excel 97-2004) files ENABLED.
##
## gdata: read.xls support for 'XLSX' (Excel 2007+) files ENABLED.
##
## Attaching package: 'gdata'
## The following object is masked from 'package:Hmisc':
##
##      combine
## The following objects are masked from 'package:dplyr':
##
##      combine, first, last
## The following object is masked from 'package:purrr':
##
##      keep
## The following object is masked from 'package:stats':
##
##      nobs
## The following object is masked from 'package:utils':
##
##      object.size
## The following object is masked from 'package:base':
##
##      startsWith
## Loading required package: mice
## Loading required package: Rcpp
## mice 2.25 2015-11-09
##
## Attaching package: 'mice'
## The following object is masked from 'package:tidyr':
##
##      complete
library(plotrix) # for std.error
##
## Attaching package: 'plotrix'
## The following object is masked from 'package:psych':
##
##      rescale
library(lme4)
library(effects)

```

# Descriptive statistics

## All participants

Descriptive data for sample.

```
setwd(params$location_of_data)

data_df <-
  read.csv("processed data/wide all tasks data.csv") %>%
  mutate(gender = as.factor(gender))

colnames(data_df)

## [1] "participant"
## [2] "condition"
## [3] "IAT_condition"
## [4] "block_order"
## [5] "task_order"
## [6] "gender"
## [7] "age"
## [8] "mean_rating"
## [9] "IAT_D1"
## [10] "IAT_mean_RT"
## [11] "IAT_perc_acc"
## [12] "IAT_exclude_based_on_fast_trials"
## [13] "SCIAT_D1"
## [14] "SCIAT_mean_RT"
## [15] "SCIAT_perc_acc"
## [16] "SCIAT_exclude_based_on_fast_trials"
## [17] "modern_racism_scale_total"

data_df %>% dplyr::count(gender)

## # A tibble: 6 × 2
##   gender      n
##   <fctr> <int>
## 1      24      1
## 2      33      1
## 3 female    48
## 4 feminine  1
## 5      male  101
## 6 nonbinary  1

data_df %>%
  dplyr::select(age) %>%
  psych::describe(fast = TRUE, # subset of descriptive stats
                  ranges = FALSE,
                  trim = 0) %>%
  dplyr::select(-vars, -se)

##      n mean  sd
## age 153 31.93 11.03
```

## Sample descriptive statistics

```
passers_df <-  
  data_df %>%  
    filter(IAT_exclude_based_on_fast_trials == FALSE & SCIAT_exclude_based_on_fast_trials == FALSE)  
  
passers_df %>% dplyr::count(IAT_condition)  
  
## # A tibble: 2 × 2  
##       IAT_condition     n  
##       <fctr> <int>  
## 1 Flowers-Insects IAT     75  
## 2           Race IAT     71  
  
passers_df %>%  
  dplyr::select(IAT_mean_RT,  
                IAT_perc_acc,  
                SCIAT_mean_RT,  
                SCIAT_perc_acc) %>%  
  psych::describe(fast = TRUE, # subset of descriptive stats  
                 ranges = FALSE,  
                 trim = 0) %>%  
  dplyr::select(-vars, -se)  
  
##           n    mean    sd  
## IAT_mean_RT  146 828.19 187.70  
## IAT_perc_acc  146   0.93   0.07  
## SCIAT_mean_RT 146 692.54 136.30  
## SCIAT_perc_acc 146   0.93   0.05
```

## H1: Differences in IAT effects between contrast pair conditions

### Outlier removal

```
# get data  
setwd(params$location_of_data)  
  
# exclude participants who had more than 10% of trials in less than 300ms, and make variables factors  
SCIAT_data <-  
  read.csv("processed data/long SCIAT data.csv") %>%  
  mutate(participant = as.factor(participant),  
         block = as.factor(ifelse(block == "compatible", "black-positive",  
                                   ifelse(block == "incompatible", "black-negative", NA))),  
         IAT_condition = as.factor(IAT_condition)) %>%  
  filter(IAT_exclude_based_on_fast_trials == FALSE & SCIAT_exclude_based_on_fast_trials == FALSE)  
  
# exclude outliers  
SCIAT_data_outliers_removed <-  
  SCIAT_data %>%  
  schoRsch::outlier(dv = "rt", # exclude individual RTs that are greater than 2.5 SD from the mean  
                   todo="elim",  
                   upper.z = 2.5,  
                   lower.z = -2.5)
```

```
## [1] "Function outlier() has screened 20440 trial(s) in total. Of these were 113 trial(s) identified as outliers."
```

## Descriptive statistics

```
SCIAT_summary_data <-  
  SCIAT_data_outliers_removed %>%  
  group_by(IAT_condition, block) %>%  
  dplyr::summarize(mean_rt = round(mean(rt), 0),  
                   sd_rt = round(sd(rt), 0),  
                   se_rt = round(std.error(rt), 2))
```

```
SCIAT_summary_data
```

```
## Source: local data frame [4 x 5]  
## Groups: IAT_condition [?]  
##  
##           IAT_condition      block mean_rt sd_rt se_rt  
##           <fctr>          <fctr>   <dbl> <dbl> <dbl>  
## 1 Flowers-Insects IAT black-negative    671   260  3.59  
## 2 Flowers-Insects IAT black-positive    664   265  3.67  
## 3           Race IAT black-negative    677   260  3.70  
## 4           Race IAT black-positive    688   283  4.02
```

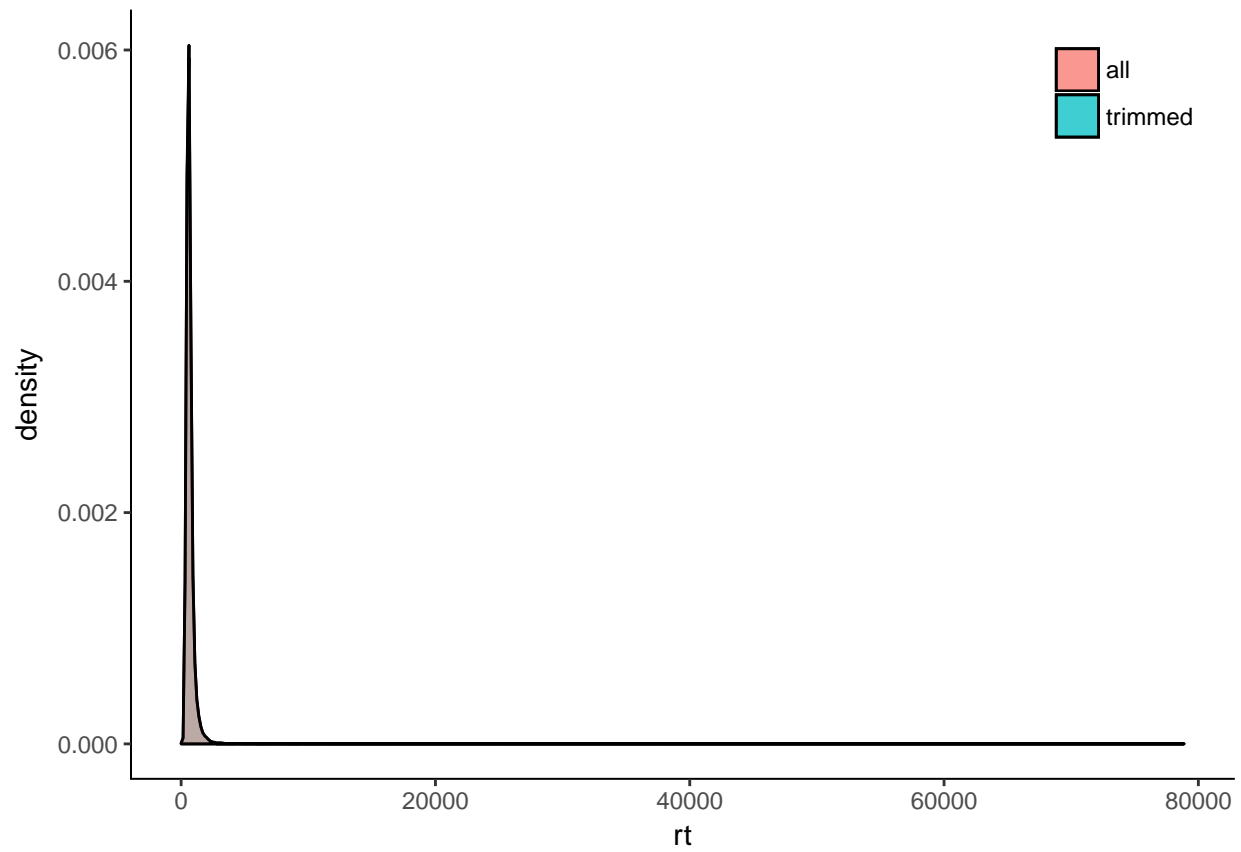
## Plots

```
# apa theme for all plots  
apatheme <-  
  theme_bw() +  
  theme(panel.grid.major = element_blank(),  
        panel.grid.minor = element_blank(),  
        panel.background = element_blank(),  
        panel.border = element_blank(),  
        #text = element_text(family='Arial'), # doesn't play nice with knitr  
        legend.title = element_blank(),  
        legend.position = c(.9,.9),  
        axis.line.x = element_line(color='black'),  
        axis.line.y = element_line(color='black'))  
  
# add a combined condition*IAT block variable for plotting  
SCIAT_data_outliers_removed <-  
  SCIAT_data_outliers_removed %>%  
  mutate(exp_factor = paste(IAT_condition, block, sep = "_"))
```

## Outlier removal

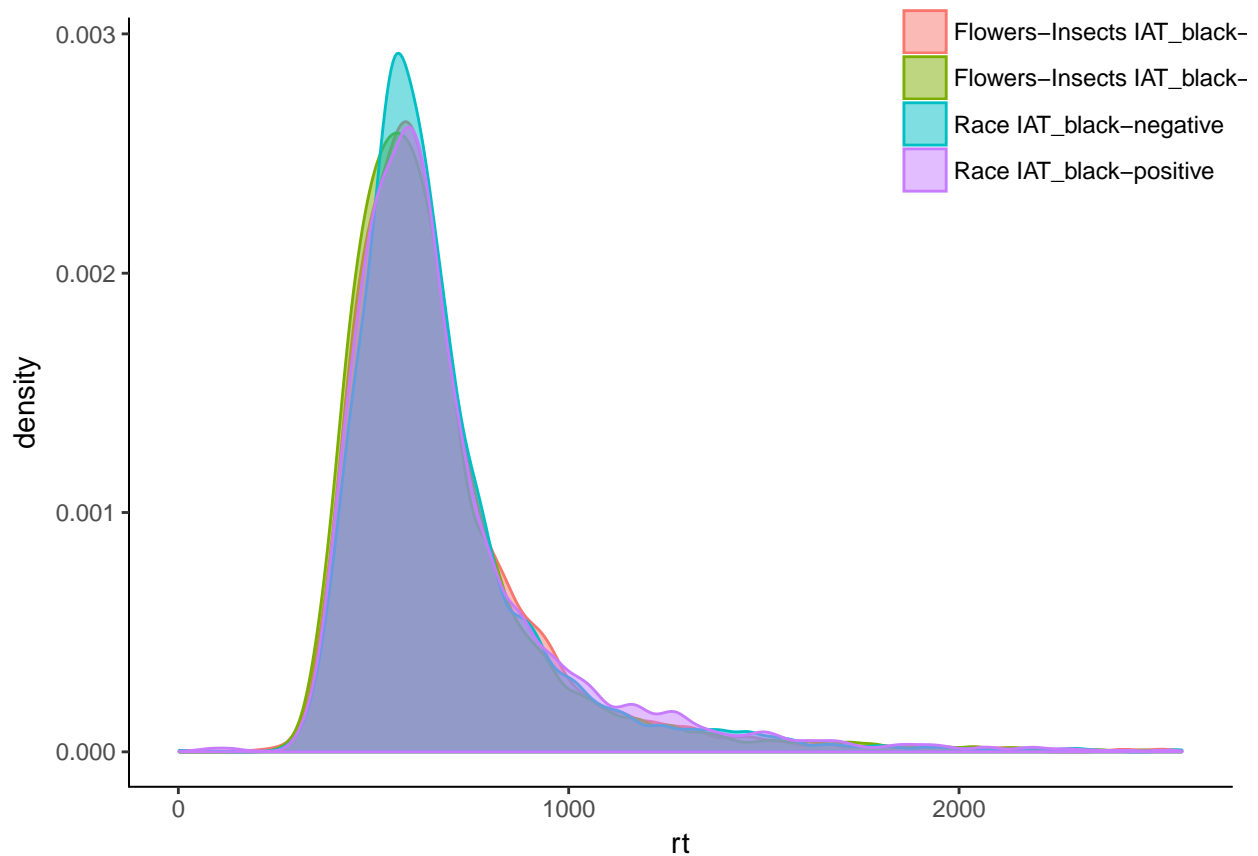
Distribution of RTs before and after trimming outliers.

```
ggplot(data = SCIAT_data_outliers_removed, aes(rt, fill = "trimmed")) +  
  geom_density(alpha=0.50) +  
  geom_density(data = SCIAT_data,  
              aes(rt, fill = "all"),  
              alpha=0.50) +  
  apatheme
```



### Density plot split by factor

```
ggplot(SCIAT_data_outliers_removed,  
  aes(rt, colour = exp_factor, fill = exp_factor)) +  
  geom_density(alpha=0.50) +  
  apatheme
```

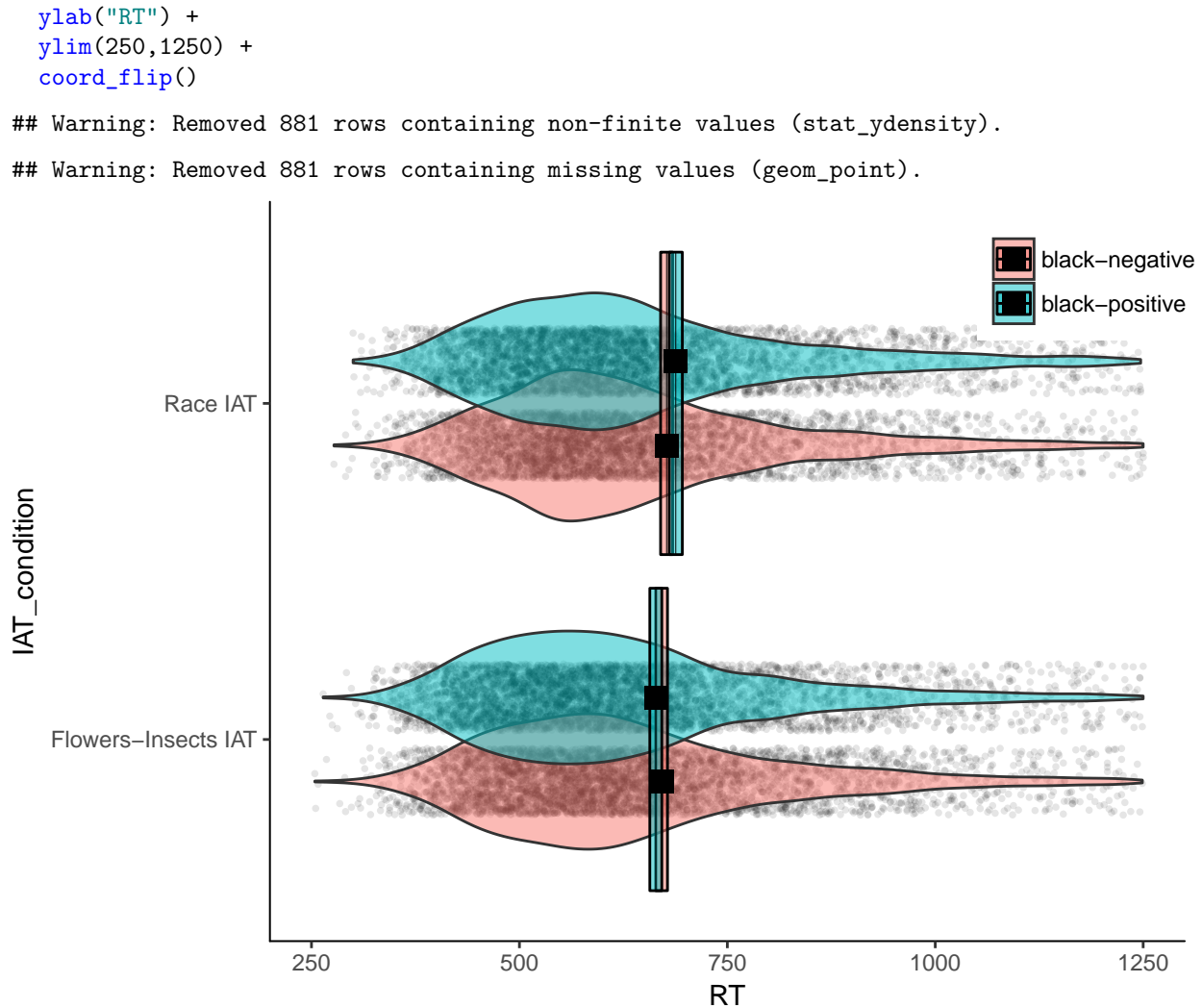


### Distribution and inference plot

Black squares are means, horizontal lines are 95% CIs, coloured shapes are distributions. I've chosen to omit presenting jittered raw data as it looks overplotted. NB scale was limited to 250 to 1250ms to make it more informative, although (non-outlier) values extend beyond the visible plot (and are included in the analysis).

```
ggplot(data = SCIAT_summary_data,
  aes(x = IAT_condition, y = mean_rt, fill = block)) +
  geom_point(data = SCIAT_data_outliers_removed,
    aes(x = IAT_condition, y = rt, fill = block),
    size = 1,
    shape = 16,
    alpha = 0.1,
    position = position_jitterdodge(dodge.width = .5)) +
  geom_violin(data = SCIAT_data_outliers_removed,
    aes(x = IAT_condition, y = rt, fill = block),
    alpha = 0.5,
    position = position_dodge(width = .5)) +
  geom_crossbar(aes(ymin = mean_rt + (-1.96*se_rt),
    ymax = mean_rt + (1.96*se_rt)),
    alpha = 0.5,
    fatten = 0) +
  geom_point(size = 4,
    shape = 15,
    position = position_dodge(width = .5)) +
  apatheme +
```





### Greyscale inference plot - marginal means

```

# calculate marginal means
model_1_forplot <- lmer(rt ~ block * IAT_condition + modern_racism_scale_total + (1 | participant),
  contrasts = list(block = "contr.sum", IAT_condition = "contr.sum"),
  data = SCIAT_data_outliers_removed)

m1_marginal_means <- as.data.frame(effect("block:IAT_condition", model_1_forplot))
m1_marginal_means

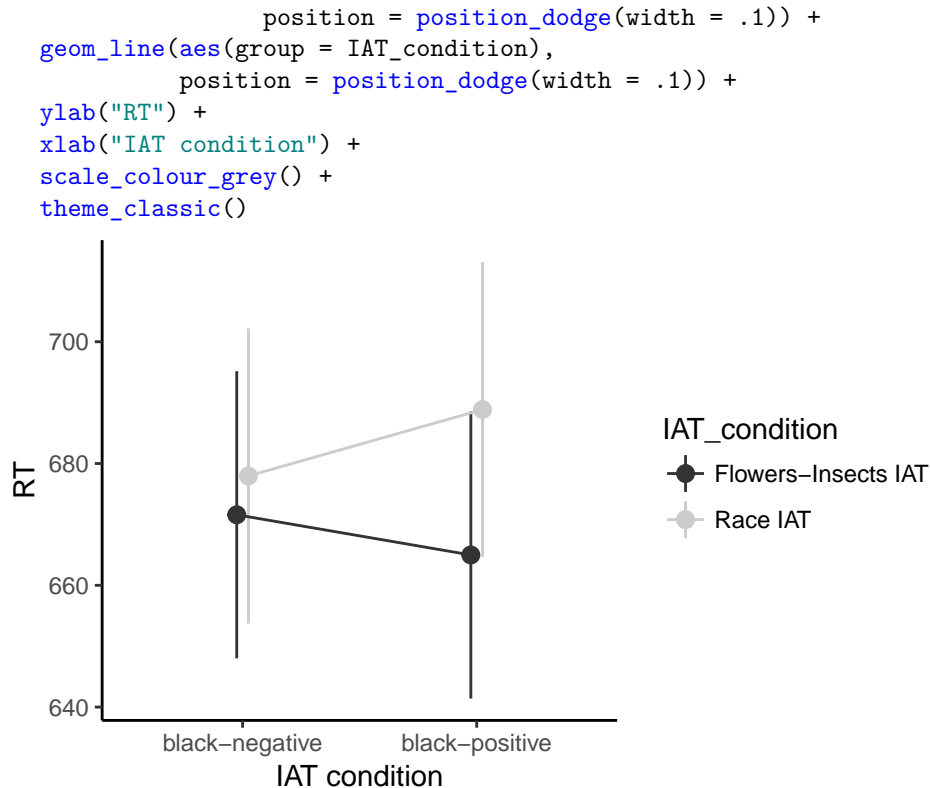
```

##	block	IAT_condition	fit	se	lower	upper
## 1	black-negative	Flowers-Insects IAT	671.5811	12.02989	648.0015	695.1607
## 2	black-positive	Flowers-Insects IAT	664.9815	12.03144	641.3990	688.5641
## 3	black-negative	Race IAT	677.9618	12.36424	653.7269	702.1967
## 4	black-positive	Race IAT	688.8631	12.36381	664.6290	713.0971

```

ggplot(data = m1_marginal_means,
  aes(x = block, y = fit, colour = IAT_condition)) +
  geom_pointrange(aes(ymin = lower,
    ymax = upper),

```



## Preregistered hypothesis test

Frequentist mixed linear effects model with participant as a random effect. Entering participant as a random effect acknowledges the non-independence of the multiple data points for each participants. Specifically, allowing for a random intercept allows for differences in general responding speed between participants, therefore controlling for age, IQ, etc., as the *D1* score does. Using a LMM here therefore serves a dual purpose of controlling for within-subject variation (traditionally served by *D1* scoring) and between subject analysis, while making use of all data points rather than collapsing the IAT's 80 RTs into just one score.

Two important considerations:

1. Production of *p* values (over LRs, etc.) is contentious, but cite the following as recent evidence for the use of Kenward Roger method estimation: <http://link.springer.com/article/10.3758%2Fs13428-016-0809-y>
2. No effect sizes are produced due to contention over how to use the random factor error. See <http://stats.stackexchange.com/questions/95054/how-to-get-an-overall-p-value-and-effect-size-for-a-categorical-factor-in-a-mi>

The model is `rt ~ block * IAT_condition + modern_racism_scale_total + (1 | participant)`. That is, RT is predicted by the interaction between SCIAT block and training IAT condition, after controlling for differences in racism, and while allowing for participants to have a random intercept (i.e., variable mean RT).

Our preregistered *a priori* hypothesis, that SCIAT effects would differ between conditions, relates to the interaction effect and not the main effects. We therefore employ type 3 sum of squares and examine only the results of the interaction.

```

# Check that variables that should be factors are indeed factors
apply(SCIAT_data_outliers_removed, class)

```

```

##                                participant                                block

```

```

##          "factor"          "factor"
##          trial_n          rt
##          "integer"        "integer"
##          accuracy          IAT_condition
##          "integer"        "factor"
##          block_order      task_order
##          "factor"        "factor"
##          gender          age
##          "factor"        "integer"
##          modern_racism_scale_total  IAT_exclude_based_on_fast_trials
##          "integer"        "logical"
##          SCIAT_exclude_based_on_fast_trials  outlier
##          "logical"        "numeric"
##          zscores          exp_factor
##          "numeric"        "character"

# LME analysis
model_1 <- afex::mixed(rt ~ block * IAT_condition + modern_racism_scale_total + (1 | participant),
  contrasts = TRUE,
  data = SCIAT_data_outliers_removed,
  type = 3,
  method = "LR",
  progress = TRUE,
  return = "mixed")

## Fitting 5 (g)lmer() models:
## [.....]

print(model_1)

##          Effect df  Chisq p.value
## 1          block  1   0.38   .54
## 2      IAT_condition  1   0.82   .37
## 3 modern_racism_scale_total  1   0.26   .61
## 4      block:IAT_condition  1  6.31 *   .01

# sigma/z scores
H1_z_score <- qnorm(1-model_1$anova_table$`Pr(>Chisq)`[4]) # 4th member is the interaction effect
H1_z_score

## [1] 2.256748

```

## Alternative method

### Plots

### Distribution and inference plot

```

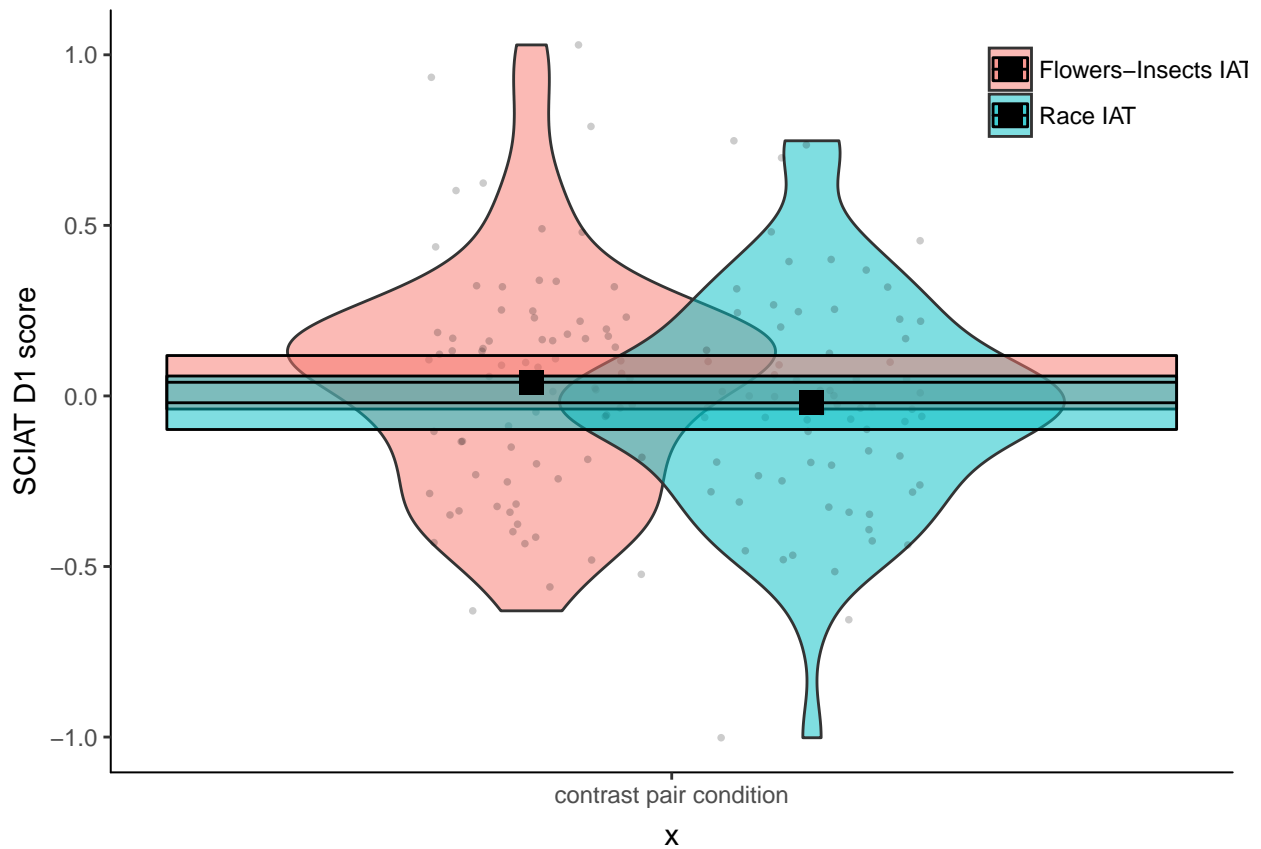
# descriptives stats
D1_summary_data <-
  passers_df %>%
  group_by(IAT_condition) %>%
  dplyr::summarize(mean_SCIAT_D1 = round(mean(SCIAT_D1), 2),
    sd_SCIAT_D1 = round(sd(SCIAT_D1), 2),
    se_SCIAT_D1 = round(std.error(SCIAT_D1), 2))

D1_summary_data

```

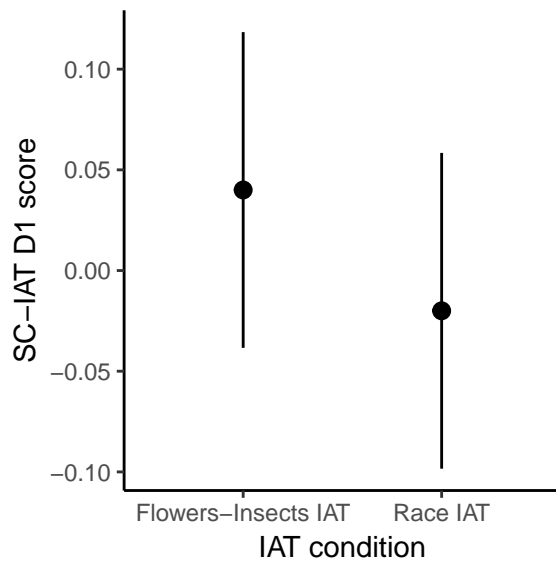
```
## # A tibble: 2 × 4
##       IAT_condition mean_SCIAT_D1 sd_SCIAT_D1 se_SCIAT_D1
##       <fctr>         <dbl>         <dbl>         <dbl>
## 1 Flowers-Insects IAT          0.04          0.33          0.04
## 2           Race IAT         -0.02          0.32          0.04

# plot
ggplot(data = D1_summary_data,
       aes(x = "contrast pair condition",
           y = mean_SCIAT_D1,
           fill = IAT_condition)) +
  geom_violin(data = passers_df,
             aes(x = "contrast pair condition",
                 y = SCIAT_D1,
                 fill = IAT_condition),
             alpha = 0.5,
             position = position_dodge(width = .5)) +
  geom_point(data = passers_df,
            aes(x = "contrast pair condition",
                y = SCIAT_D1,
                fill = IAT_condition),
            size = 1,
            shape = 16,
            alpha = 0.2,
            position = position_jitterdodge(dodge.width = .5)) +
  geom_crossbar(aes(ymax = mean_SCIAT_D1 + (1.96*se_SCIAT_D1),
                    ymin = mean_SCIAT_D1 + (-1.96*se_SCIAT_D1)),
               alpha = 0.5,
               fatten = 1) +
  geom_point(size = 4,
            shape = 15,
            position = position_dodge(width = .5)) +
  apatheme +
  ylab("SCIAT D1 score")
```



### Greyscale inference plot

```
ggplot(data = D1_summary_data,
       aes(x = IAT_condition, y = mean_SCIAT_D1)) +
  geom_pointrange(aes(ymax = mean_SCIAT_D1 + (1.96*se_SCIAT_D1),
                     ymin = mean_SCIAT_D1 + (-1.96*se_SCIAT_D1))) +
  ylab("SC-IAT D1 score") +
  xlab("IAT condition") +
  scale_colour_grey() +
  theme_classic()
```



Cohen's d with 95% CIs

```
attach(passers_df)

# Welch's t test
E1_t_test <- t.test(formula = SCIAT_D1 ~ IAT_condition,
                    alternative = "less",
                    paired = FALSE)

E1_t_test

##
## Welch Two Sample t-test
##
## data: SCIAT_D1 by IAT_condition
## t = 1.1605, df = 143.99, p-value = 0.8761
## alternative hypothesis: true difference in means is less than 0
## 95 percent confidence interval:
##      -Inf 0.1512895
## sample estimates:
## mean in group Flowers-Insects IAT      mean in group Race IAT
##                0.03905333                -0.02329577

# sigma/z scores
E1_z_score <- qnorm(1 - E1_t_test$p.value)
E1_z_score

## [1] -1.155837

# Cohen's d
E1_cohens_d <- cohen.d(SCIAT_D1 ~ IAT_condition)
E1_cohens_d

##
## Cohen's d
##
## d estimate: 0.1919157 (negligible)
## 95 percent confidence interval:
```

```
##          inf          sup
## -0.1384049  0.5222362
```

## H2: Differences in self-reported ratings between contrast pair conditions

### Descriptive statistics

```
# get data
setwd(params$location_of_data)

# exclude participants who had more than 10% of trials in less than 300ms, and make variables factors
ratings_data <-
  read.csv("processed data/long ratings data.csv") %>%
  mutate(participant = as.factor(participant),
         IAT_condition = as.factor(IAT_condition)) %>%
  filter(IAT_exclude_based_on_fast_trials == FALSE & SCIAT_exclude_based_on_fast_trials == FALSE)

ratings_summary_data <-
  ratings_data %>%
  group_by(IAT_condition) %>%
  dplyr::summarize(mean_rating = round(mean(rating), 2),
                  sd_rating = round(sd(rating), 2),
                  se_rating = round(std.error(rating), 2))

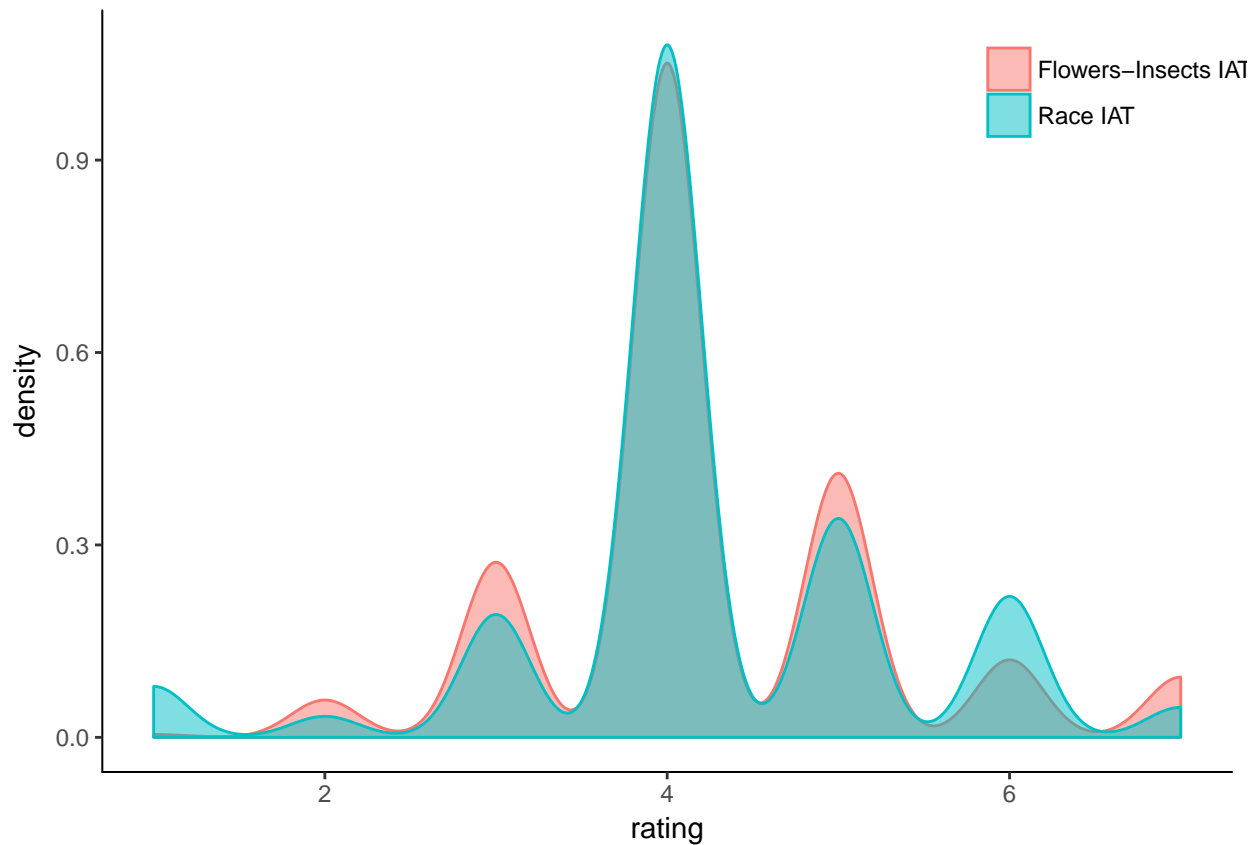
ratings_summary_data

## # A tibble: 2 × 4
##       IAT_condition mean_rating sd_rating se_rating
##       <fctr>         <dbl>      <dbl>    <dbl>
## 1 Flowers-Insects IAT      4.26      1.03     0.05
## 2 Race IAT          4.21      1.14     0.06
```

### Plots

#### Density plot split by factor

```
ggplot(ratings_data,
       aes(rating, colour = IAT_condition, fill = IAT_condition)) +
  geom_density(alpha=0.50) +
  apatheme
```

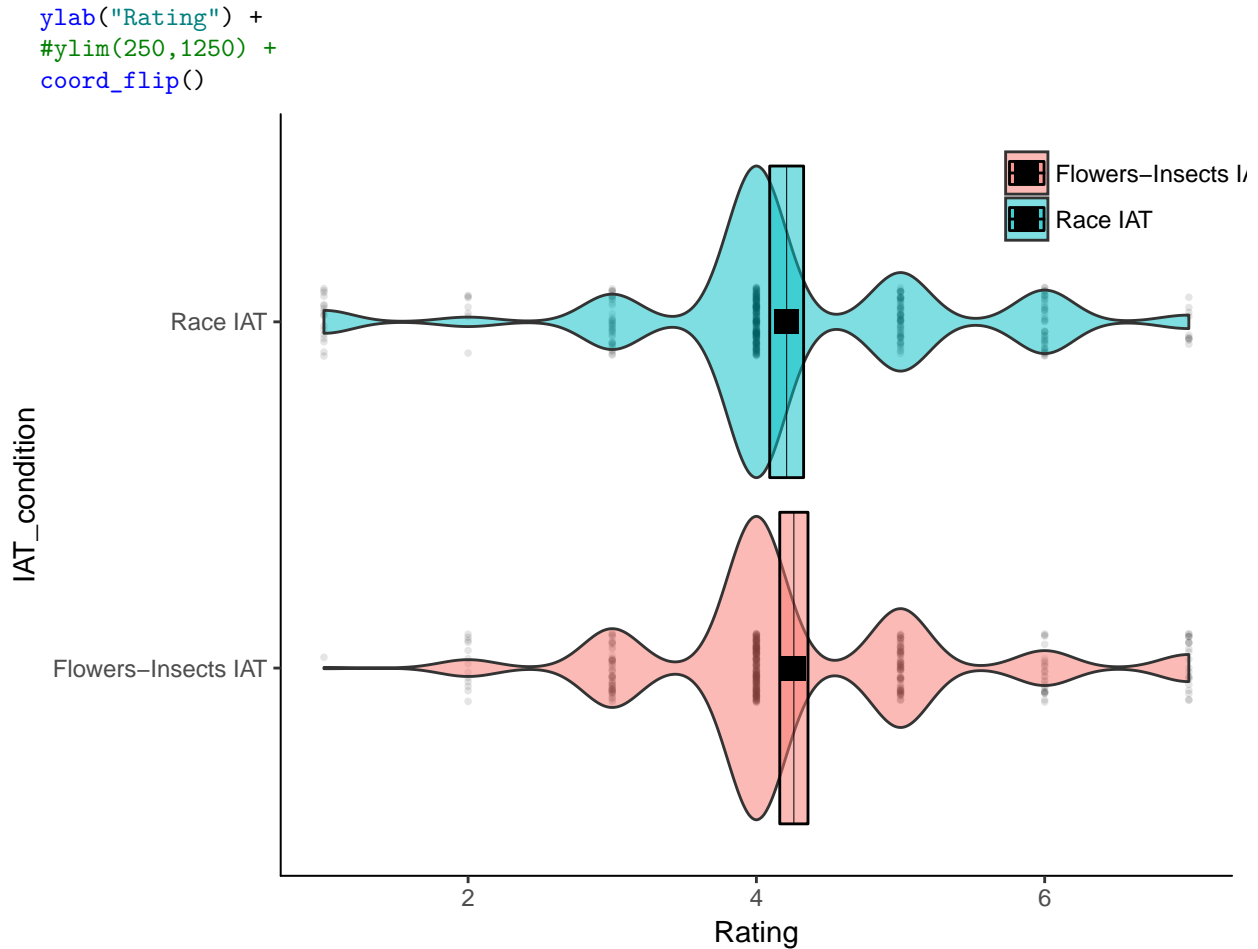


### Distribution and inference plot

Black squares are means, horizontal lines are 95% CIs, coloured shapes are distributions. I've chosen to omit presenting jittered raw data as it looks overplotted. NB scale was limited to 250 to 1250ms to make it more informative, although (non-outlier) values extend beyond the visible plot (and are included in the analysis).

```
ggplot(data = ratings_summary_data,
  aes(x = IAT_condition, y = mean_rating, fill = IAT_condition)) +
  geom_point(data = ratings_data,
    aes(x = IAT_condition, y = rating, fill = IAT_condition),
    size = 1,
    shape = 16,
    alpha = 0.1,
    position = position_jitterdodge(dodge.width = .5)) +
  geom_violin(data = ratings_data,
    aes(x = IAT_condition, y = rating, fill = IAT_condition),
    alpha = 0.5,
    position = position_dodge(width = .5)) +
  geom_crossbar(aes(ymin = mean_rating + (-1.96*se_rating),
    ymax = mean_rating + (1.96*se_rating)),
    alpha = 0.5,
    fatten = 0) +
  geom_point(size = 4,
    shape = 15,
    position = position_dodge(width = .5)) +
  apatheme +
```





### Greyscale inference plot - marginal means

```

# calculate marginal means
model_2_forplot <- lmer(rating ~ IAT_condition + modern_racism_scale_total + (1 | participant),
                        contrasts = list(block = "contr.sum", IAT_condition = "contr.sum"),
                        data = ratings_data)

## Warning in model.matrix.default(fixedform, fr, contrasts): variable 'block'
## is absent, its contrast will be ignored

m2_marginal_means <- as.data.frame(effect("IAT_condition", model_2_forplot))

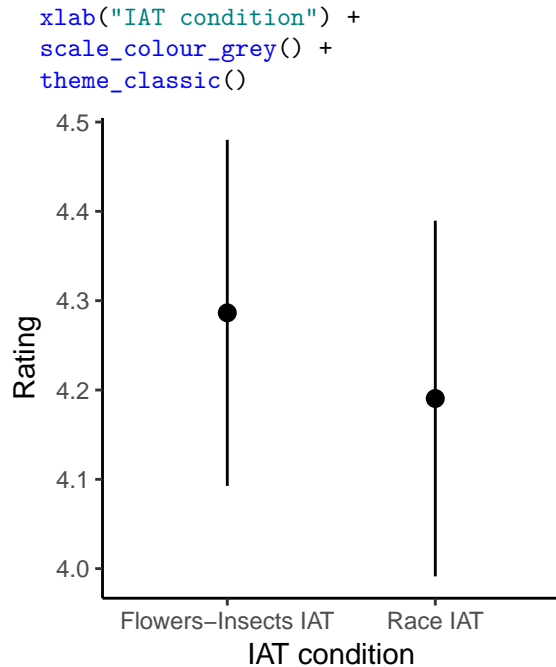
## Warning in model.matrix.default(mt, mf, contrasts): variable 'block' is
## absent, its contrast will be ignored

m2_marginal_means

##           IAT_condition      fit      se    lower    upper
## 1 Flowers-Insects IAT 4.286381 0.09871651 4.092631 4.480130
## 2              Race IAT 4.190443 0.10146316 3.991303 4.389583

ggplot(data = m2_marginal_means,
       aes(x = IAT_condition, y = fit)) +
  geom_pointrange(aes(ymax = upper,
                     ymin = lower)) +
  ylab("Rating") +

```



## Preregistered hypothesis test

Frequentist mixed linear effects model with participant as a random effect. Entering participant as a random effect acknowledges the non-independence of the multiple data points for each participants. Specifically, allowing for a random intercept allows for differences in general responding speed between participants, therefore controlling for age, IQ, etc., as the *D1* score does. Using a LMM here therefore serves a dual purpose of controlling for within-subject variation (traditionally served by *D1* scoring) and between subject analysis, while making use of all data points rather than collapsing the IAT's 80 RTs into just one score.

Two important considerations:

1. Production of *p* values (over LRs, etc.) is contentious, but cite the following as recent evidence for the use of Kenward Roger method estimation: <http://link.springer.com/article/10.3758%2Fs13428-016-0809-y>
2. No effect sizes are produced due to contention over how to use the random factor error. See <http://stats.stackexchange.com/questions/95054/how-to-get-an-overall-p-value-and-effect-size-for-a-categorical-factor-in-a-mi>

The model is `rating ~ IAT_condition + (1 | participant)`. That is, rating is predicted by training IAT condition, after controlling for racism and while allowing participants to have a random intercept (i.e., acknowledging the non-independence of participants ratings of the images).

Our preregistered *a priori* hypothesis, that ratings would differ between conditions, relates to the main effect for IAT\_condition. We therefore examine only the results of this main effect.

```
# Check that variables that should be factors are indeed factors
suppress(ratings_data, class)
```

```
##           participant           trial_n
##           "factor"           "integer"
##           rating             IAT_condition
##           "integer"          "factor"
##           block_order        task_order
##           "factor"           "factor"
##           gender             age
```

```

##                                "factor"                                "integer"
##      modern_racism_scale_total  IAT_exclude_based_on_fast_trials
##                                "integer"                                "logical"
##  SCIAT_exclude_based_on_fast_trials
##                                "logical"

# LME analysis
model_2 <- afex::mixed(rating ~ IAT_condition + modern_racism_scale_total + (1 | participant),
                      contrasts = TRUE,
                      data = ratings_data,
                      type = 3,
                      method = "LR",
                      progress = TRUE,
                      return = "mixed")

## Fitting 3 (g)lmer() models:
## [...]

print(model_2)

##              Effect df      Chisq p.value
## 1          IAT_condition  1      0.47    .49
## 2 modern_racism_scale_total  1 32.20 *** <.0001

# precise p values
model_2$anova_table$`Pr(>Chisq)`[1]

## [1] 0.4941388

# sigma/z scores
H2_z_score <- qnorm(1-model_2$anova_table$`Pr(>Chisq)`[1]) # 1st member is IAT condition
H2_z_score

## [1] 0.0146923

```