Expt2-HighvsLow

Jason Geller

12/29/2020

Contents

Introduction

This RMD file comes from a preregistered study where we examined the moderating role of test expectancy on the Sans Forgetica effect/disfluency effect. We preregistered a sample size of 232 participants (116 in each group) through MTurk. This file explains how I read in the data, analyzed it, and plotted the results. If you have any questions please reach out to me at:jason.geller@ruccs.rutgers.edu

Load Packages

```
#packages you will need
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --
```

```
## v ggplot2 3.3.2     v purrr   0.3.4
## v tibble  3.0.6     v dplyr   1.0.3
## v tidyr   1.1.2     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.5.0
```

```
## -- Conflicts ---------------------------------------- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

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

```
##
## Attaching package: 'data.table'
```

```
## The following objects are masked from 'package:dplyr':
##
##     between, first, last
```

```
## The following object is masked from 'package:purrr':
##
##     transpose
```

```
library(here)
```

```
## here() starts at /Users/jgeller1/Desktop/SF-Testing-New1
```

```
library(afex)
```

```
## Loading required package: lme4
```

```
## Loading required package: Matrix
```

```
##
## Attaching package: 'Matrix'
```

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

```
## Registered S3 methods overwritten by 'car':
##   method                          from
##   influence.merMod                lme4
##   cooks.distance.influence.merMod lme4
##   dfbeta.influence.merMod         lme4
##   dfbetas.influence.merMod        lme4
```

```
## ************
## 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
## - NEWS: library('emmeans') now needs to be called explicitly!
## - 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
```

```
library(Rmisc)
```

```
## Loading required package: lattice
```

```
## Loading required package: plyr
```

```
## ------------------------------------------------------------------------------
```

```
## You have loaded plyr after dplyr - this is likely to cause problems.
## If you need functions from both plyr and dplyr, please load plyr first, then dplyr:
## library(plyr); library(dplyr)
```

```
## ------------------------------------------------------------------------------
```

```
##
## Attaching package: 'plyr'
```

```
## The following object is masked from 'package:here':
##
##     here
```

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

```
## The following object is masked from 'package:purrr':
##
##     compact
```

```
library(cowplot)
library(see)
library(ggrepel)
library(report)
```

```
## report is in alpha - help us improve by reporting bugs on github.com/easystats/report/issues
```

```
library(emmeans)
library(BayesFactor)
```

```
## Loading required package: coda
```

```
## ************
## Welcome to BayesFactor 0.9.12-4.2. If you have questions, please contact Richard Morey (richarddmorey
##
## Type BFManual() to open the manual.
## ************
```

```
library(MOTE)
```

Read in Raw Data

First we must read in data for each counterbalanced list. There were four and there was no simple way to counterbalance online with PsychoPy and Pavlovia.

CB 1 (Low Test Expectancy)

Cued Recall Responses

```
dataset1 <- datasetlow1 %>%
    dplyr::group_by(participant, turkid)%>% # polygon_2 is a unqiue value for the test phase
    dplyr::filter(mouse_5.clicked_name=="polygon_2") %>% dplyr::select(textbox.text, cue1, targ1, font)
    mutate(textbox.text=tolower(textbox.text)) %>% # some folks capitalzied words
  dplyr::mutate(cond="Low Test Expectancy", cb="low1") %>%
 dplyr::mutate(new_id=ifelse(is.na(participant)| participant=="J", turkid,participant)) #Mturkers used
```

```
## Adding missing grouping variables: `participant`, `turkid`
```

```
as.data.frame(dplyr::count(dataset1, new_id))
```

```
##         participant         turkid          new_id  n
## 1    A182N7RLXGSCZG A182N7RLXGSCZG   A182N7RLXGSCZG 24
## 2    A1L1S0IAPZB4MO           <NA>   A1L1S0IAPZB4MO 24
## 3    A1O57XVTTB1PNZ A1O57XVTTB1PNZ   A1O57XVTTB1PNZ 24
## 4    A1S6C6EN8MPKC3 A1S6C6EN8MPKC3   A1S6C6EN8MPKC3 24
## 5    A231NY4QEQ58PJ A231NY4QEQ58PJ   A231NY4QEQ58PJ 24
## 6    A23EGLIF8IEH11 A23EGLIF8IEH11   A23EGLIF8IEH11 24
## 7    A2RLOB9VH2XSRT A2RLOB9VH2XSRT   A2RLOB9VH2XSRT 24
## 8    A345TDMHP3DQ3G A345TDMHP3DQ3G   A345TDMHP3DQ3G 24
## 9    A3GPYCHKB2KDLC A3GPYCHKB2KDLC   A3GPYCHKB2KDLC 24
## 10   A3IQRBKS1DUJVZ A3IQRBKS1DUJVZ   A3IQRBKS1DUJVZ 24
## 11   A3K1P4SPR8XS7R A3K1P4SPR8XS7R   A3K1P4SPR8XS7R 24
## 12   A3NG2O5QJ48IM4 A3NG2O5QJ48IM4   A3NG2O5QJ48IM4 24
## 13   A3VBNWON5XOUVS A3VBNWON5XOUVS   A3VBNWON5XOUVS 24
## 14              AA A1WSCND5HZZV5N              AA 24
## 15              AB  ANVLT6GSNXYIO              AB 24
## 16    ACJY49UF6GM13  ACJY49UF6GM13    ACJY49UF6GM13 24
## 17              Amy A1XDLU7D50OR8U             Amy 24
## 18    ANMD34DNHLOLC  ANMD34DNHLOLC    ANMD34DNHLOLC 24
## 19    AS4NIEQJWCG3M  AS4NIEQJWCG3M    AS4NIEQJWCG3M 24
## 20     AS5W4VV79F5V           <NA>     AS5W4VV79F5V 24
## 21    AYY7JSAD0SGXL  AYY7JSAD0SGXL    AYY7JSAD0SGXL 24
## 22                d A369K7I5SXBPNC                d 24
## 23              DAN A3NFGEUZAH9V5G              DAN 24
## 24           Donna A24NKUCAX1LURL           Donna 24
## 25          DOUGLAS      INSTITUTE          DOUGLAS 24
## 26              Ely A3TDA1NWHNQFE3              Ely 24
## 27             GAIL A1VV1SOHHGUXPM             GAIL 24
## 28              goo A3T5UAYFKJBVTA              goo 24
## 29                H  AHWF0QIK9VTDS                H 24
## 30            Henry A14T9T8AJK1XZI            Henry 24
## 31                J  AZUTZP7QKWWXN    AZUTZP7QKWWXN 24
## 32               J3 A39PRBPSKA7A49               J3 24
## 33     Jewel Julian  A2TKHXY755FAM     Jewel Julian 24
## 34               jl A2RCE6APFZ8EAD               jl 24
## 35             judy A3PQYTQ44C8UQ4             judy 24
## 36         Kristine A3LMCFV79DBWJR         Kristine 24
## 37               LS A3VZDRWPGJH46L               LS 24
## 38         Madeline A3G6PURCIIY2VQ         Madeline 24
## 39              Mel A3OHRFXPEKQMIE              Mel 24
```

```
## 40              MJB A3OOD9IMOHPPFQ              MJB 24
## 41             Mona A1ML6B8SBSWN7G             Mona 24
## 42 Nobody Important A2R6UQCQ8T3RZE Nobody Important 24
## 43        Nola Nola A1CC9FGFOGRBUY        Nola Nola 24
## 44          noname3        noname3          noname3 24
## 45            promi A2MS1GQLGAX9FZ            promi 24
## 46                r A1N1ULK71RHVMM                r 24
## 47             Raul A3L9K9DZI8LFM4             Raul 24
## 48              Ray A1ZNRERDDZZHGW              Ray 24
## 49             sean A3KW8XFFLI5TO5             sean 24
## 50 stacey schmeidel A3HEON28SY1PPN stacey schmeidel 24
## 51             TARA  AFH5Q22HIXRA2             TARA 24
## 52      Taylor Many A1W4RCPZK3UXCM      Taylor Many 24
## 53               TT A1SOFLJOEQB591               TT 24
## 54   Virginia Sorto              2   Virginia Sorto 24
## 55             <NA> A23HZ18KTCK2DA   A23HZ18KTCK2DA 24
## 56             <NA>  ATNNZOOIRXT1B    ATNNZOOIRXT1B 24
## 57             <NA>  AXB86SVXU5SEG    AXB86SVXU5SEG 24
## 58             <NA>        noname2          noname2 24
```

JOls

```
dataset1_jol <- datasetlow1 %>%
    dplyr::group_by(participant)%>%
     dplyr::select(participant,turkid, atypic_slider.response, normal_slider.response) %>%
  mutate(new_id=ifelse(is.na(participant), turkid, participant)) %>%
  mutate(new_id1=ifelse(is.na(new_id), participant, new_id))%>%
  mutate(cond="Low Test Expectancy")%>%
  ungroup() %>%
  select(new_id1,  cond, atypic_slider.response, normal_slider.response) %>%
  na.omit(.)%>%
  tidyr::pivot_longer(atypic_slider.response:normal_slider.response, names_to = "TypeFace", values_to =

as.data.frame(dplyr::count(dataset1_jol, new_id1))
```

```
##             new_id1 n
## 1    A182N7RLXGSCZG 2
## 2    A1L1SOIAPZB4MO 2
## 3    A1O57XVTTB1PNZ 2
## 4    A1S6C6EN8MPKC3 2
## 5    A231NY4QEQ58PJ 2
## 6    A23EGLIF8IEH11 2
## 7    A23HZ18KTCK2DA 2
## 8    A2RLOB9VH2XSRT 2
## 9    A345TDMHP3DQ3G 2
## 10   A3GPYCHKB2KDLC 2
## 11   A3IQRBKS1DUJVZ 2
## 12   A3K1P4SPR8XS7R 2
## 13   A3NG2O5QJ48IM4 2
## 14   A3VBNWON5XOUVS 2
## 15               AA 2
## 16               AB 2
## 17    ACJY49UF6GM13 2
## 18              Amy 2
```

```
## 19    ANMD34DNHLOLC 2
## 20    AS4NIEQJWCG3M 2
## 21     AS5W4VV79F5V 2
## 22    ATNNZOOIRXT1B 2
## 23    AXB86SVXU5SEG 2
## 24    AYY7JSADOSGXL 2
## 25                d 2
## 26              DAN 2
## 27            Donna 2
## 28          DOUGLAS 2
## 29              Ely 2
## 30             GAIL 2
## 31              goo 2
## 32                H 2
## 33            Henry 2
## 34                J 2
## 35               J3 2
## 36     Jewel Julian 2
## 37               jl 2
## 38             judy 2
## 39         Kristine 2
## 40               LS 2
## 41         Madeline 2
## 42              Mel 2
## 43              MJB 2
## 44             Mona 2
## 45 Nobody Important 2
## 46        Nola Nola 2
## 47          noname2 2
## 48          noname3 2
## 49            promi 2
## 50                r 2
## 51             Raul 2
## 52              Ray 2
## 53             sean 2
## 54 stacey schmeidel 2
## 55             TARA 2
## 56      Taylor Many 2
## 57               TT 2
## 58   Virginia Sorto 2
```

RTs

```
dataset1_rt <- datasetlow1 %>%
    dplyr::group_by(participant)%>%
     dplyr::select(participant,turkid, mouse_4.time, font) %>%
  dplyr::mutate(new_id=ifelse(is.na(participant), turkid, participant)) %>%
  dplyr::mutate(new_id1=ifelse(is.na(new_id), participant, new_id))%>%
  dplyr::mutate(cond="Low Test Expectancy")%>%
  dplyr::ungroup() %>%
  dplyr::select(new_id1, cond, font, mouse_4.time)

as.data.frame(dplyr::count(dataset1_rt, new_id1))
```

```
##             new_id1  n
## 1    A182N7RLXGSCZG 49
## 2    A1L1S0IAPZB4MO 49
## 3    A1O57XVTTB1PNZ 49
## 4    A1S6C6EN8MPKC3 49
## 5    A231NY4QEQ58PJ 49
## 6    A23EGLIF8IEH11 49
## 7    A23HZ18KTCK2DA 49
## 8    A2RLOB9VH2XSRT 49
## 9    A345TDMHP3DQ3G 49
## 10   A3GPYCHKB2KDLC 49
## 11   A3IQRBKS1DUJVZ 49
## 12   A3K1P4SPR8XS7R 49
## 13   A3NG2O5QJ48IM4 49
## 14   A3VBNWON5XOUVS 49
## 15               AA 49
## 16               AB 49
## 17    ACJY49UF6GM13 49
## 18              Amy 49
## 19    ANMD34DNHLOLC 49
## 20    AS4NIEQJWCG3M 49
## 21     AS5W4VV79F5V 49
## 22    ATNNZOOIRXT1B 49
## 23    AXB86SVXU5SEG 49
## 24    AYY7JSAD0SGXL 49
## 25               d 49
## 26             DAN 49
## 27           Donna 49
## 28          DOUGLAS 49
## 29             Ely 49
## 30            GAIL 49
## 31             goo 49
## 32               H 49
## 33           Henry 49
## 34               J 49
## 35              J3 49
## 36     Jewel Julian 49
## 37              jl 49
## 38            judy 49
## 39         Kristine 49
## 40              LS 49
## 41         Madeline 49
## 42              Mel 49
## 43              MJB 49
## 44             Mona 49
## 45 Nobody Important 49
## 46        Nola Nola 49
## 47          noname2 48
## 48          noname3 48
## 49            promi 49
## 50                r 49
## 51             Raul 49
## 52              Ray 49
## 53             sean 49
```

```
## 54 stacey schmeidel 49
## 55             TARA 49
## 56      Taylor Many 49
## 57               TT 49
## 58   Virginia Sorto 49
## 59             <NA>  2
```

CB 2 - Low Test

Cued Recall

```
dataset2 <- datasetlow2 %>%
    dplyr::group_by(participant, turkid)%>%
    dplyr::filter(mouse_5.clicked_name=="polygon_2") %>% dplyr::select(textbox.text, cue1, targ1, font)
  dplyr::mutate(textbox.text=tolower(textbox.text)) %>%
  dplyr::mutate(cond="Low Test Expectancy", cb="low2") %>%
 dplyr::mutate(new_id=ifelse(is.na(participant),turkid,participant))
```

```
## Adding missing grouping variables: `participant`, `turkid`
```

```
#Mturkers used "me" for participant name so we need to extract unique id
#check number of Ps in each CB and make sure names are unqiue
as.data.frame(dplyr::count(dataset2, new_id))
```

```
##           participant         turkid           new_id  n
## 1      A1Q6VY3C3CEOA0 A1Q6VY3C3CEOA0     A1Q6VY3C3CEOA0 24
## 2      A1XRIEYLA7ZJ3B         female     A1XRIEYLA7ZJ3B 24
## 3      A20IA63WXHF683 A20IA63WXHF683     A20IA63WXHF683 24
## 4      A2H4LJ4CXZQ304 A2H4LJ4CXZQ304     A2H4LJ4CXZQ304 24
## 5      A33B85TN97HQ33 A33B85TN97HQ33     A33B85TN97HQ33 24
## 6      A3F51C49T9A34D           <NA>     A3F51C49T9A34D 24
## 7      A3PQVDFT21IISU A3PQVDFT21IISU     A3PQVDFT21IISU 24
## 8      A3QL867S9A4NY4 A3QL867S9A4NY4     A3QL867S9A4NY4 24
## 9       A55F6J0VK7TLV  A55F6J0VK7TLV      A55F6J0VK7TLV 24
## 10      A6O1VFV32LBS8  A6O1VFV32LBS8      A6O1VFV32LBS8 24
## 11       ANUG05IDFTWF   ANUG05IDFTWF       ANUG05IDFTWF 24
## 12      AOZIEKTQGHSJX  AOZIEKTQGHSJX      AOZIEKTQGHSJX 24
## 13      AUFE6HLRZCIYW  AUFE6HLRZCIYW      AUFE6HLRZCIYW 24
## 14      AVKMFV9CZOMLR           <NA>      AVKMFV9CZOMLR 24
## 15      AVOF14300525D  AVOF14300525D      AVOF14300525D 24
## 16                Ben A2P3V5VELJXAMP                Ben 24
## 17            Brandon A29KDPDITZSBRQ            Brandon 24
## 18                 by A3DK4DDLEWCMM1                 by 24
## 19            Candice  A4H31418X7MBU            Candice 24
## 20              Chloe  AOMFEAWQHU3D8              Chloe 24
## 21               Dave A1ADAWW4IHPCQ7               Dave 24
## 22             Dillon  A1KC342L9WAQ2             Dillon 24
## 23                 ed A2ULVVSHLNVHAY                 ed 24
## 24               Erin  AG8PCITDPLLKU               Erin 24
## 25              Ethan A2WPHVMLLEV5ZB              Ethan 24
## 26             Fatima A3C6V8AOIJXIV0             Fatima 24
## 27                 FL A3HSONMX6LNZR4                 FL 24
## 28                 gi A2EY0J9LOPU3TU                 gi 24
## 29              Gregg A3GNQDFPZALU92              Gregg 24
## 30                Jen A2O2Q0LVWOWJ1G                Jen 24
```

```
## 31           Jennifer A25F4VG7W3BTWV           Jennifer 24
## 32                jmm A2CIRSDBG6NQU1                jmm 24
## 33              Julie A1POAW2RQBTBXB              Julie 24
## 34               Karl A1BHW4QNL2D9EF               Karl 24
## 35              katie  ADUOSLOPPCDYX              katie 24
## 36                  m A2LRMK05AC21ZC                  m 24
## 37               Mary A3FNC8ELMK8YJA               Mary 24
## 38                 MD A2C7EYRPWL8B3L                 MD 24
## 39                 me A33QOAS590VTQ5                 me 24
## 40                me1 A1CM4AU052HWDO                me1 24
## 41            noname4           <NA>            noname4 24
## 42               Rema A1EPE4DKRJMWER               Rema 24
## 43                 sb  AWN8SISULPG1E                 sb 24
## 44                 SD A2EY0J9LOPU3TU                 SD 24
## 45 stephanie hatfield  AE8QS3RQ1G1WA stephanie hatfield 24
## 46              Steve  AKX5RHHO8BIUX              Steve 24
## 47         tacophopia A1YX741QQSMKPS         tacophopia 24
## 48                 TB  AXKNMOECIV3ZA                 TB 24
## 49   Timothy Buchanan  A8AHU8NCH1P9X   Timothy Buchanan 24
## 50               <NA> A187RIRDOFNE1B     A187RIRDOFNE1B 24
## 51               <NA> A1XCQA727LJVL2     A1XCQA727LJVL2 24
## 52               <NA> A27LC4TGHQ7GCG     A27LC4TGHQ7GCG 24
## 53               <NA> A2VB2MZFRO7VN5     A2VB2MZFRO7VN5 24
## 54               <NA> A3GTPW3MYSLMR7     A3GTPW3MYSLMR7 24
## 55               <NA>  A770OB1ZAGJ4G      A770OB1ZAGJ4G 24
## 56               <NA>  AK9CDGMQCBSIU      AK9CDGMQCBSIU 24
## 57               <NA>  AKFNJG4UMP6MX      AKFNJG4UMP6MX 24
## 58               <NA>           <NA>               <NA> 24
```

JOLs

```
dataset2_jol <- datasetlow2 %>%
    dplyr::group_by(participant)%>%
     dplyr::select(participant,turkid, atypic_slider.response, normal_slider.response) %>%
  mutate(new_id=ifelse(is.na(participant), turkid, participant)) %>%
  mutate(new_id1=ifelse(is.na(new_id), participant, new_id))%>%
  mutate(cond="Low Test Expectancy")%>%
  ungroup() %>%
  select(new_id1,  cond, atypic_slider.response, normal_slider.response) %>%
  na.omit(.)%>%
  tidyr::pivot_longer(atypic_slider.response:normal_slider.response, names_to = "TypeFace", values_to =

as.data.frame(dplyr::count(dataset2_jol, new_id1))
```

```
##              new_id1 n
## 1      A187RIRDOFNE1B 2
## 2      A1Q6VY3C3CEOA0 2
## 3      A1XCQA727LJVL2 2
## 4      A1XRIEYLA7ZJ3B 2
## 5      A20IA63WXHF683 2
## 6      A27LC4TGHQ7GCG 2
## 7      A2H4LJ4CXZQ304 2
## 8      A2VB2MZFRO7VN5 2
## 9      A33B85TN97HQ33 2
```

```
## 10     A3F51C49T9A34D 2
## 11     A3GTPW3MYSLMR7 2
## 12     A3PQVDFT21IISU 2
## 13     A3QL867S9A4NY4 2
## 14      A55F6J0VK7TLV 2
## 15      A6O1VFV32LBS8 2
## 16      A770OB1ZAGJ4G 2
## 17      AK9CDGMQCBSIU 2
## 18      AKFNJG4UMP6MX 2
## 19       ANUG05IDFTWF 2
## 20      AOZIEKTQGHSJX 2
## 21      AUFE6HLRZCIYW 2
## 22      AVKMFV9CZ0MLR 2
## 23      AVOF14300525D 2
## 24                Ben 2
## 25            Brandon 2
## 26                 by 2
## 27            Candice 2
## 28              Chloe 2
## 29               Dave 2
## 30             Dillon 2
## 31                 ed 2
## 32               Erin 2
## 33              Ethan 2
## 34             Fatima 2
## 35                 FL 2
## 36                 gi 2
## 37              Gregg 2
## 38                Jen 2
## 39           Jennifer 2
## 40                jmm 2
## 41              Julie 2
## 42               Karl 2
## 43              katie 2
## 44                  m 2
## 45               Mary 2
## 46                 MD 2
## 47                 me 2
## 48                me1 2
## 49            noname4 2
## 50               Rema 2
## 51                 sb 2
## 52                 SD 2
## 53 stephanie hatfield 2
## 54              Steve 2
## 55         tacophopia 2
## 56                 TB 2
## 57   Timothy Buchanan 2
```

RTs

```
dataset2_rt <- datasetlow2 %>%
    dplyr::group_by(participant)%>%
     dplyr::select(participant,turkid, mouse_4.time, font) %>%
```

```
  dplyr::mutate(new_id=ifelse(is.na(participant), turkid, participant)) %>%
  dplyr::mutate(new_id1=ifelse(is.na(new_id), participant, new_id))%>%
  dplyr::mutate(cond="Low Test Expectancy")%>%
  dplyr::ungroup() %>%
  dplyr::select(new_id1, cond, font, mouse_4.time)

as.data.frame(dplyr::count(dataset2_rt, new_id1))
```

```
##              new_id1  n
## 1      A187RIRDOFNE1B 49
## 2      A1Q6VY3C3CEOA0 49
## 3      A1XCQA727LJVL2 49
## 4      A1XRIEYLA7ZJ3B 49
## 5      A20IA63WXHF683 49
## 6      A27LC4TGHQ7GCG 49
## 7      A2H4LJ4CXZQ304 49
## 8      A2VB2MZFRO7VN5 49
## 9      A33B85TN97HQ33 49
## 10     A3F51C49T9A34D 49
## 11     A3GTPW3MYSLMR7 49
## 12     A3PQVDFT21IISU 49
## 13     A3QL867S9A4NY4 49
## 14      A55F6J0VK7TLV 49
## 15      A6O1VFV32LBS8 49
## 16      A770OB1ZAGJ4G 49
## 17      AK9CDGMQCBSIU 49
## 18      AKFNJG4UMP6MX 49
## 19       ANUG05IDFTWF 49
## 20      AOZIEKTQGHSJX 49
## 21      AUFE6HLRZCIYW 49
## 22      AVKMFV9CZ0MLR 49
## 23      AVOF14300525D 49
## 24               Ben 49
## 25            Brandon 49
## 26                 by 49
## 27            Candice 49
## 28              Chloe 49
## 29               Dave 49
## 30             Dillon 49
## 31                 ed 49
## 32               Erin 49
## 33              Ethan 49
## 34             Fatima 49
## 35                 FL 49
## 36                 gi 49
## 37              Gregg 49
## 38                Jen 49
## 39           Jennifer 49
## 40                jmm 49
## 41              Julie 49
## 42               Karl 49
## 43              katie 49
## 44                  m 49
## 45               Mary 49
```

```
## 46                 MD 49
## 47                 me 49
## 48                me1 49
## 49            noname4 48
## 50               Rema 49
## 51                 sb 49
## 52                 SD 49
## 53 stephanie hatfield 49
## 54              Steve 49
## 55         tacophopia 49
## 56                 TB 49
## 57   Timothy Buchanan 49
## 58               <NA> 50
```

CB 1 (High Test Expectancy)

Cued Responses

```
dataset3 <- datasethigh3 %>%
    dplyr::group_by(participant, turkid)%>%
    dplyr::filter(mouse_5.clicked_name=="polygon_2") %>% dplyr::select(textbox.text, cue1, targ1, font)
  dplyr::mutate(textbox.text=tolower(textbox.text)) %>%
  dplyr::mutate(cond="High Test Expectancy", cb="high1") %>%
 dplyr::mutate(new_id=ifelse(is.na(participant),turkid,participant))
```

```
## Adding missing grouping variables: `participant`, `turkid`
```

```
#Mturkers used "me" for participant name so we need to extract unique id
#check number of Ps in each CB and make sure names are unqiue
as.data.frame(dplyr::count(dataset2, new_id))
```

```
##           participant         turkid           new_id  n
## 1      A1Q6VY3C3CEOA0 A1Q6VY3C3CEOA0     A1Q6VY3C3CEOA0 24
## 2      A1XRIEYLA7ZJ3B         female     A1XRIEYLA7ZJ3B 24
## 3      A20IA63WXHF683 A20IA63WXHF683     A20IA63WXHF683 24
## 4      A2H4LJ4CXZQ304 A2H4LJ4CXZQ304     A2H4LJ4CXZQ304 24
## 5      A33B85TN97HQ33 A33B85TN97HQ33     A33B85TN97HQ33 24
## 6      A3F51C49T9A34D           <NA>     A3F51C49T9A34D 24
## 7      A3PQVDFT21IISU A3PQVDFT21IISU     A3PQVDFT21IISU 24
## 8      A3QL867S9A4NY4 A3QL867S9A4NY4     A3QL867S9A4NY4 24
## 9       A55F6J0VK7TLV  A55F6J0VK7TLV      A55F6J0VK7TLV 24
## 10      A6O1VFV32LBS8  A6O1VFV32LBS8      A6O1VFV32LBS8 24
## 11       ANUG05IDFTWF   ANUG05IDFTWF       ANUG05IDFTWF 24
## 12      AOZIEKTQGHSJX  AOZIEKTQGHSJX      AOZIEKTQGHSJX 24
## 13      AUFE6HLRZCIYW  AUFE6HLRZCIYW      AUFE6HLRZCIYW 24
## 14      AVKMFV9CZOMLR           <NA>      AVKMFV9CZOMLR 24
## 15      AVOF14300525D  AVOF14300525D      AVOF14300525D 24
## 16                Ben A2P3V5VELJXAMP                Ben 24
## 17            Brandon A29KDPDITZSBRQ            Brandon 24
## 18                 by A3DK4DDLEWCMM1                 by 24
## 19            Candice  A4H31418X7MBU            Candice 24
## 20              Chloe  AOMFEAWQHU3D8              Chloe 24
## 21               Dave A1ADAWW4IHPCQ7               Dave 24
## 22             Dillon  A1KC342L9WAQ2             Dillon 24
## 23                 ed A2ULVVSHLNVHAY                 ed 24
```

```
## 24               Erin  AG8PCITDPLLKU               Erin 24
## 25             Ethan A2WPHVMLLEV5ZB             Ethan 24
## 26            Fatima A3C6V8AOIJXIV0            Fatima 24
## 27                FL A3HSONMX6LNZR4                FL 24
## 28                gi A2EY0J9LOPU3TU                gi 24
## 29             Gregg A3GNQDFPZALU92             Gregg 24
## 30               Jen A2O2Q0LVWOWJ1G               Jen 24
## 31          Jennifer A25F4VG7W3BTWV          Jennifer 24
## 32               jmm A2CIRSDBG6NQU1               jmm 24
## 33             Julie A1POAW2RQBTBXB             Julie 24
## 34              Karl A1BHW4QNL2D9EF              Karl 24
## 35             katie  ADU0SL0PPCDYX             katie 24
## 36                 m A2LRMK05AC21ZC                 m 24
## 37              Mary A3FNC8ELMK8YJA              Mary 24
## 38                MD A2C7EYRPWL8B3L                MD 24
## 39                me A33QOAS590VTQ5                me 24
## 40               me1 A1CM4AU052HWDO               me1 24
## 41           noname4           <NA>           noname4 24
## 42              Rema A1EPE4DKRJMWER              Rema 24
## 43                sb  AWN8SISULPG1E                sb 24
## 44                SD A2EY0J9LOPU3TU                SD 24
## 45 stephanie hatfield  AE8QS3RQ1G1WA stephanie hatfield 24
## 46             Steve  AKX5RHHO8BIUX             Steve 24
## 47        tacophopia A1YX741QQSMKPS        tacophopia 24
## 48                TB  AXKNMOECIV3ZA                TB 24
## 49   Timothy Buchanan  A8AHU8NCH1P9X   Timothy Buchanan 24
## 50              <NA> A187RIRDOFNE1B    A187RIRDOFNE1B 24
## 51              <NA> A1XCQA727LJVL2    A1XCQA727LJVL2 24
## 52              <NA> A27LC4TGHQ7GCG    A27LC4TGHQ7GCG 24
## 53              <NA> A2VB2MZFRO7VN5    A2VB2MZFRO7VN5 24
## 54              <NA> A3GTPW3MYSLMR7    A3GTPW3MYSLMR7 24
## 55              <NA>  A770OB1ZAGJ4G     A770OB1ZAGJ4G 24
## 56              <NA>  AK9CDGMQCBSIU     AK9CDGMQCBSIU 24
## 57              <NA>  AKFNJG4UMP6MX     AKFNJG4UMP6MX 24
## 58              <NA>           <NA>              <NA> 24
```

JOLs

```
dataset3_jol <- datasethigh3 %>%
    dplyr::group_by(participant)%>%
     dplyr::select(participant,turkid, atypic_slider.response, normal_slider.response) %>%
  mutate(new_id=ifelse(is.na(participant), turkid, participant)) %>%
  mutate(new_id1=ifelse(is.na(new_id), participant, new_id))%>%
  mutate(cond="High Test Expectancy")%>%
  ungroup() %>%
  select(new_id1,  cond, atypic_slider.response, normal_slider.response) %>%
  na.omit(.)%>%
  tidyr::pivot_longer(atypic_slider.response:normal_slider.response, names_to = "TypeFace", values_to =

as.data.frame(dplyr::count(dataset3_jol, new_id1))
```

```
##                 new_id1 n
## 1                     ? 2
## 2        A10LHWALI4BZPC 2
```

```
## 3        A17Y73URHQYGAA 2
## 4        A1BQI9JKLVCIL0 2
## 5        A1DWQO9X2ZQ35G 2
## 6        A1U69QDM8UIXZ5 2
## 7        A2488KXHWJTJR7 2
## 8        A2JM1IVEZCXX3U 2
## 9        a2mqmmcvzjtfl2 2
## 10       A34CPKFZXBX1PO 2
## 11       A3D5VZQM079JQJ 2
## 12       A3TS0KMBDEEUVX 2
## 13        A49A0FF6ZXYDU 2
## 14         ADVCIFLB5A9B 2
## 15                Amy T 2
## 16              ANADEAU 2
## 17        AOB6ZJTDB416G 2
## 18               ashley 2
## 19                Beclu 2
## 20               Brandi 2
## 21                Brett 2
## 22            cb1_high1 2
## 23            cb1_high2 2
## 24            cb1_high3 2
## 25            cb1_high4 2
## 26               Cherry 2
## 27               chriss 2
## 28            Christina 2
## 29  Christine Parkerson 2
## 30                 Cole 2
## 31 COPIEDA1C5J40P3I65EI 2
## 32                   DC 2
## 33                   DS 2
## 34                    E 2
## 35                    j 2
## 36     Jennifer Tunnell 2
## 37         Jesse Harris 2
## 38                   JM 2
## 39                  joe 2
## 40                 john 2
## 41            Johnathan 2
## 42                   JP 2
## 43         juliastelter 2
## 44               Justin 2
## 45                   ky 2
## 46                   ll 2
## 47                 Male 2
## 48               marcia 2
## 49                 mike 2
## 50               randal 2
## 51               rusong 2
## 52                Saber 2
## 53       sherri sanders 2
## 54               Swetha 2
## 55                    T 2
## 56                  thy 2
```

```
## 57                  viv 2
## 58                  yhh 2
```

RTs

```
dataset3_rt <- datasethigh3 %>%
    dplyr::group_by(participant)%>%
     dplyr::select(participant,turkid, mouse_4.time, font) %>%
  dplyr::mutate(new_id=ifelse(is.na(participant), turkid, participant)) %>%
  dplyr::mutate(new_id1=ifelse(is.na(new_id), participant, new_id))%>%
  dplyr::mutate(cond="High Test Expectancy")%>%
  dplyr::ungroup() %>%
  dplyr::select(new_id1, cond, font, mouse_4.time)

as.data.frame(dplyr::count(dataset3_rt, new_id1))
```

```
##                new_id1  n
## 1                    ? 49
## 2        A10LHWALI4BZPC 49
## 3        A17Y73URHQYGAA 49
## 4        A1BQI9JKLVCIL0 49
## 5        A1DWQO9X2ZQ35G 49
## 6        A1U69QDM8UIXZ5 49
## 7        A2488KXHWJTJR7 49
## 8        A2JM1IVEZCXX3U 49
## 9        a2mqmmcvzjtfl2 49
## 10       A34CPKFZXBX1PO 49
## 11       A3D5VZQM079JQJ 49
## 12       A3TS0KMBDEEUVX 49
## 13        A49A0FF6ZXYDU 49
## 14         ADVCIFLB5A9B 49
## 15                Amy T 49
## 16              ANADEAU 49
## 17        AOB6ZJTDB416G 49
## 18               ashley 49
## 19                Beclu 49
## 20               Brandi 49
## 21                Brett 49
## 22            cb1_high1 48
## 23            cb1_high2 48
## 24            cb1_high3 48
## 25            cb1_high4 48
## 26               Cherry 49
## 27               chriss 49
## 28            Christina 49
## 29  Christine Parkerson 49
## 30                 Cole 49
## 31 COPIEDA1C5J40P3I65EI 49
## 32                   DC 49
## 33                   DS 49
## 34                    E 49
## 35                    j 49
## 36     Jennifer Tunnell 49
## 37         Jesse Harris 49
```

```
## 38                   JM 49
## 39                  joe 49
## 40                 john 49
## 41            Johnathan 49
## 42                   JP 49
## 43          juliastelter 49
## 44                Justin 49
## 45                   ky 49
## 46                   ll 49
## 47                 Male 49
## 48                marcia 49
## 49                  mike 49
## 50                randal 49
## 51                rusong 49
## 52                 Saber 49
## 53       sherri sanders 49
## 54                Swetha 49
## 55                     T 49
## 56                   thy 49
## 57                   viv 49
## 58                   yhh 49
## 59                  <NA>  4
```

CB 2 (High Expectancy)

Cued Responses

```
dataset4 <- datasethigh4 %>%
    dplyr::group_by(participant, turkid)%>%
    dplyr::filter(mouse_5.clicked_name=="polygon_2") %>% dplyr::select(textbox.text, cue1, targ1, font)
  dplyr::mutate(textbox.text=tolower(textbox.text)) %>%
  dplyr::mutate(cond="High Test Expectancy", cb="high2") %>%
 dplyr::mutate(new_id=ifelse(is.na(participant),turkid,participant))
```

```
## Adding missing grouping variables: `participant`, `turkid`
```

```
#Mturkers used "me" for participant name so we need to extract unique id
#check number of Ps in each CB and make sure names are unqiue
as.data.frame(dplyr::count(dataset2, new_id))
```

```
##           participant         turkid            new_id  n
## 1      A1Q6VY3C3CEOA0 A1Q6VY3C3CEOA0     A1Q6VY3C3CEOA0 24
## 2      A1XRIEYLA7ZJ3B         female     A1XRIEYLA7ZJ3B 24
## 3      A20IA63WXHF683 A20IA63WXHF683     A20IA63WXHF683 24
## 4      A2H4LJ4CXZQ304 A2H4LJ4CXZQ304     A2H4LJ4CXZQ304 24
## 5      A33B85TN97HQ33 A33B85TN97HQ33     A33B85TN97HQ33 24
## 6      A3F51C49T9A34D           <NA>     A3F51C49T9A34D 24
## 7      A3PQVDFT21IISU A3PQVDFT21IISU     A3PQVDFT21IISU 24
## 8      A3QL867S9A4NY4 A3QL867S9A4NY4     A3QL867S9A4NY4 24
## 9       A55F6J0VK7TLV  A55F6J0VK7TLV      A55F6J0VK7TLV 24
## 10      A6O1VFV32LBS8  A6O1VFV32LBS8      A6O1VFV32LBS8 24
## 11       ANUG05IDFTWF   ANUG05IDFTWF       ANUG05IDFTWF 24
## 12      AOZIEKTQGHSJX  AOZIEKTQGHSJX      AOZIEKTQGHSJX 24
## 13      AUFE6HLRZCIYW  AUFE6HLRZCIYW      AUFE6HLRZCIYW 24
## 14      AVKMFV9CZ0MLR           <NA>      AVKMFV9CZ0MLR 24
```

```
## 15      AVOF14300525D  AVOF14300525D      AVOF14300525D 24
## 16                Ben A2P3V5VELJXAMP                Ben 24
## 17            Brandon A29KDPDITZSBRQ            Brandon 24
## 18                 by A3DK4DDLEWCMM1                 by 24
## 19            Candice  A4H31418X7MBU            Candice 24
## 20              Chloe  AOMFEAWQHU3D8              Chloe 24
## 21               Dave A1ADAWW4IHPCQ7               Dave 24
## 22             Dillon  A1KC342L9WAQ2             Dillon 24
## 23                 ed A2ULVVSHLNVHAY                 ed 24
## 24               Erin  AG8PCITDPLLKU               Erin 24
## 25              Ethan A2WPHVMLLEV5ZB              Ethan 24
## 26             Fatima A3C6V8AOIJXIV0             Fatima 24
## 27                 FL A3HSONMX6LNZR4                 FL 24
## 28                 gi A2EY0J9LOPU3TU                 gi 24
## 29              Gregg A3GNQDFPZALU92              Gregg 24
## 30                Jen A2O2Q0LVWOWJ1G                Jen 24
## 31           Jennifer A25F4VG7W3BTWV           Jennifer 24
## 32                jmm A2CIRSDBG6NQU1                jmm 24
## 33              Julie A1POAW2RQBTBXB              Julie 24
## 34               Karl A1BHW4QNL2D9EF               Karl 24
## 35              katie  ADU0SL0PPCDYX              katie 24
## 36                  m A2LRMK05AC21ZC                  m 24
## 37               Mary A3FNC8ELMK8YJA               Mary 24
## 38                 MD A2C7EYRPWL8B3L                 MD 24
## 39                 me A33QOAS590VTQ5                 me 24
## 40                me1 A1CM4AU052HWDO                me1 24
## 41            noname4           <NA>            noname4 24
## 42               Rema A1EPE4DKRJMWER               Rema 24
## 43                 sb  AWN8SISULPG1E                 sb 24
## 44                 SD A2EY0J9LOPU3TU                 SD 24
## 45 stephanie hatfield  AE8QS3RQ1G1WA stephanie hatfield 24
## 46              Steve  AKX5RHHO8BIUX              Steve 24
## 47         tacophopia A1YX741QQSMKPS         tacophopia 24
## 48                 TB  AXKNMOECIV3ZA                 TB 24
## 49   Timothy Buchanan  A8AHU8NCH1P9X   Timothy Buchanan 24
## 50               <NA> A187RIRDOFNE1B    A187RIRDOFNE1B 24
## 51               <NA> A1XCQA727LJVL2    A1XCQA727LJVL2 24
## 52               <NA> A27LC4TGHQ7GCG    A27LC4TGHQ7GCG 24
## 53               <NA> A2VB2MZFRO7VN5    A2VB2MZFRO7VN5 24
## 54               <NA> A3GTPW3MYSLMR7    A3GTPW3MYSLMR7 24
## 55               <NA>  A770OB1ZAGJ4G     A770OB1ZAGJ4G 24
## 56               <NA>  AK9CDGMQCBSIU     AK9CDGMQCBSIU 24
## 57               <NA>  AKFNJG4UMP6MX     AKFNJG4UMP6MX 24
## 58               <NA>           <NA>              <NA> 24
```

JOLs

```
dataset4_jol <- datasethigh4 %>%
    dplyr::group_by(participant)%>%
     dplyr::select(participant,turkid, atypic_slider.response, normal_slider.response) %>%
  mutate(new_id=ifelse(is.na(participant), turkid, participant)) %>%
  mutate(new_id1=ifelse(is.na(new_id), participant, new_id))%>%
  mutate(cond="High Test Expectancy")%>%
  ungroup() %>%
```

```
  select(new_id1,  cond, atypic_slider.response, normal_slider.response) %>%
  na.omit(.)%>%
  tidyr::pivot_longer(atypic_slider.response:normal_slider.response, names_to = "TypeFace", values_to =

as.data.frame(dplyr::count(dataset3_jol, new_id1))
```

```
##                 new_id1 n
## 1                     ? 2
## 2        A10LHWALI4BZPC 2
## 3        A17Y73URHQYGAA 2
## 4        A1BQI9JKLVCIL0 2
## 5        A1DWQO9X2ZQ35G 2
## 6        A1U69QDM8UIXZ5 2
## 7        A2488KXHWJTJR7 2
## 8        A2JM1IVEZCXX3U 2
## 9        a2mqmmcvzjtfl2 2
## 10       A34CPKFZXBX1PO 2
## 11       A3D5VZQM079JQJ 2
## 12       A3TS0KMBDEEUVX 2
## 13        A49A0FF6ZXYDU 2
## 14         ADVCIFLB5A9B 2
## 15                Amy T 2
## 16              ANADEAU 2
## 17        AOB6ZJTDB416G 2
## 18               ashley 2
## 19                Beclu 2
## 20               Brandi 2
## 21                Brett 2
## 22            cb1_high1 2
## 23            cb1_high2 2
## 24            cb1_high3 2
## 25            cb1_high4 2
## 26               Cherry 2
## 27               chriss 2
## 28            Christina 2
## 29  Christine Parkerson 2
## 30                 Cole 2
## 31 COPIEDA1C5J40P3I65EI 2
## 32                   DC 2
## 33                   DS 2
## 34                    E 2
## 35                    j 2
## 36     Jennifer Tunnell 2
## 37         Jesse Harris 2
## 38                   JM 2
## 39                  joe 2
## 40                 john 2
## 41            Johnathan 2
## 42                   JP 2
## 43         juliastelter 2
## 44               Justin 2
## 45                   ky 2
## 46                   ll 2
## 47                 Male 2
```

```
## 48               marcia 2
## 49                 mike 2
## 50               randal 2
## 51               rusong 2
## 52                Saber 2
## 53       sherri sanders 2
## 54               Swetha 2
## 55                    T 2
## 56                  thy 2
## 57                  viv 2
## 58                  yhh 2
```

RTs

```
dataset4_rt <- datasethigh4 %>%
    dplyr::group_by(participant)%>%
     dplyr::select(participant,turkid, mouse_4.time, font) %>%
  dplyr::mutate(new_id=ifelse(is.na(participant), turkid, participant)) %>%
  dplyr::mutate(new_id1=ifelse(is.na(new_id), participant, new_id))%>%
  dplyr::mutate(cond="High Test Expectancy")%>%
  dplyr::ungroup() %>%
  dplyr::select(new_id1, cond, font, mouse_4.time)

as.data.frame(dplyr::count(dataset4_rt, new_id1))
```

```
##               new_id1  n
## 1      A14LOABUGAITBM 49
## 2        A1FG2G9TZSDN 49
## 3      A1NHFCWPAQNUT6 49
## 4       A2QC2W3GCGKHB 49
## 5      A2W5BVSBD1GO8A 49
## 6      A314XJY8V1YL12 49
## 7       AHB886J4P46JW 49
## 8       AK1GE19RT3PK6 49
## 9                 Ali 49
## 10              Amber 49
## 11                Ant 49
## 12      ARPBDM5QZ4XQC 49
## 13                 AS 49
## 14      AU01P6YB9J5JX 49
## 15      AVWU9JDY5G81E 49
## 16              barri 49
## 17          bma5rocks 49
## 18  Cameron Goldsmith 49
## 19              Carol 49
## 20          cb2_high1 48
## 21          cb2_high2 48
## 22          cb4_high4 48
## 23                 CS 49
## 24              Emily 49
## 25             female 49
## 26                 gv 49
## 27                 j1 49
## 28           jennifer 49
```

```
## 29                jim 49
## 30                Jim 49
## 31            jocelyn 49
## 32               Joel 49
## 33             Kaylee 49
## 34              Keith 49
## 35               Lexy 49
## 36                 lm 49
## 37               Luke 49
## 38                  M 49
## 39               Mia 49
## 40 Michele Richardson 49
## 41                Moe 49
## 42                  N 49
## 43        Participant 49
## 44                Pat 49
## 45  penguincatcher123 49
## 46             Reagan 49
## 47                 RF 49
## 48              Robin 49
## 49                 RR 49
## 50                 RT 49
## 51                Sam 49
## 52              shawn 49
## 53             Steffi 49
## 54                 VB 49
## 55    Walter Tressler 49
## 56              weerd 49
## 57             WORKER 49
## 58                  y 49
## 59               <NA>  3
```

Combine High and Low CB lists

Cued

```
all_c<- rbind(dataset1, dataset2, dataset3, dataset4)

all_c<-mutate(all_c, new_id=ifelse(is.na(new_id), "NA1", new_id))


as.data.frame(dplyr::count(all_c, new_id))
```

```
##             participant              turkid              new_id  n
## 1                     ?      A2LX9M4S8GKZXO                   ? 24
## 2        A17Y73URHQYGAA      A17Y73URHQYGAA      A17Y73URHQYGAA 24
## 3        A182N7RLXGSCZG      A182N7RLXGSCZG      A182N7RLXGSCZG 24
## 4        A1DWQO9X2ZQ35G      A1DWQO9X2ZQ35G      A1DWQO9X2ZQ35G 24
## 5          A1FG2G9TZSDN        A1FG2G9TZSDN        A1FG2G9TZSDN 24
## 6        A1L1S0IAPZB4MO               <NA>      A1L1S0IAPZB4MO 24
## 7        A1NHFCWPAQNUT6      A1NHFCWPAQNUT6      A1NHFCWPAQNUT6 24
## 8        A1O57XVTTB1PNZ      A1O57XVTTB1PNZ      A1O57XVTTB1PNZ 24
## 9        A1Q6VY3C3CEOA0      A1Q6VY3C3CEOA0      A1Q6VY3C3CEOA0 24
```

```
## 10       A1S6C6EN8MPKC3       A1S6C6EN8MPKC3       A1S6C6EN8MPKC3 24
## 11       A1U69QDM8UIXZ5       A1U69QDM8UIXZ5       A1U69QDM8UIXZ5 24
## 12       A1XRIEYLA7ZJ3B               female       A1XRIEYLA7ZJ3B 24
## 13       A20IA63WXHF683       A20IA63WXHF683       A20IA63WXHF683 24
## 14       A231NY4QEQ58PJ       A231NY4QEQ58PJ       A231NY4QEQ58PJ 24
## 15       A23EGLIF8IEH11       A23EGLIF8IEH11       A23EGLIF8IEH11 24
## 16       A2488KXHWJTJR7       A2488KXHWJTJR7       A2488KXHWJTJR7 24
## 17       A2H4LJ4CXZQ304       A2H4LJ4CXZQ304       A2H4LJ4CXZQ304 24
## 18       A2RLOB9VH2XSRT       A2RLOB9VH2XSRT       A2RLOB9VH2XSRT 24
## 19       A2W5BVSBD1G08A       A2W5BVSBD1G08A       A2W5BVSBD1G08A 24
## 20       A33B85TN97HQ33       A33B85TN97HQ33       A33B85TN97HQ33 24
## 21       A345TDMHP3DQ3G       A345TDMHP3DQ3G       A345TDMHP3DQ3G 24
## 22       A34CPKFZXBX1PO       A34CPKFZXBX1PO       A34CPKFZXBX1PO 24
## 23       A3F51C49T9A34D                 <NA>       A3F51C49T9A34D 24
## 24       A3GPYCHKB2KDLC       A3GPYCHKB2KDLC       A3GPYCHKB2KDLC 24
## 25       A3IQRBKS1DUJVZ       A3IQRBKS1DUJVZ       A3IQRBKS1DUJVZ 24
## 26       A3K1P4SPR8XS7R       A3K1P4SPR8XS7R       A3K1P4SPR8XS7R 24
## 27       A3NG205QJ48IM4       A3NG205QJ48IM4       A3NG205QJ48IM4 24
## 28       A3PQVDFT21IISU       A3PQVDFT21IISU       A3PQVDFT21IISU 24
## 29       A3QL867S9A4NY4       A3QL867S9A4NY4       A3QL867S9A4NY4 24
## 30       A3VBNWON5XOUVS       A3VBNWON5XOUVS       A3VBNWON5XOUVS 24
## 31        A49AOFF6ZXYDU        A49AOFF6ZXYDU        A49AOFF6ZXYDU 24
## 32        A55F6JOVK7TLV        A55F6JOVK7TLV        A55F6JOVK7TLV 24
## 33        A6O1VFV32LBS8        A6O1VFV32LBS8        A6O1VFV32LBS8 24
## 34                   AA       A1WSCND5HZZV5N                   AA 24
## 35                   AB        ANVLT6GSNXYIO                   AB 24
## 36        ACJY49UF6GM13        ACJY49UF6GM13        ACJY49UF6GM13 24
## 37         ADVCIFLB5A9B         ADVCIFLB5A9B         ADVCIFLB5A9B 24
## 38        AK1GE19RT3PK6        AK1GE19RT3PK6        AK1GE19RT3PK6 24
## 39                  Ali       A1LY42RR96EA3P                  Ali 24
## 40               Amber        ALB15OP3N4OU9               Amber 24
## 41                 Amy       A1XDLU7D50OR8U                 Amy 24
## 42               Amy T       A38HKLJZ5U3GZF               Amy T 24
## 43             ANADEAU       A3S8SMCLCGTX9U             ANADEAU 24
## 44       ANMD34DNHLOLC        ANMD34DNHLOLC       ANMD34DNHLOLC 24
## 45                 Ant                 <NA>                 Ant 24
## 46        ANUGO5IDFTWF         ANUGO5IDFTWF        ANUGO5IDFTWF 24
## 47        AOB6ZJTDB416G         AOB6ZJTDB416G        AOB6ZJTDB416G 24
## 48        AOZIEKTQGHSJX         AOZIEKTQGHSJX        AOZIEKTQGHSJX 24
## 49        ARPBDM5QZ4XQC         ARPBDM5QZ4XQC        ARPBDM5QZ4XQC 24
## 50                   AS       A3B2YRXL26RPVH                   AS 24
## 51       AS4NIEQJWCG3M        AS4NIEQJWCG3M       AS4NIEQJWCG3M 24
## 52        AS5W4VV79F5V                 <NA>        AS5W4VV79F5V 24
## 53               ashley       A10Q4U3BRHXXPP               ashley 24
## 54       AUFE6HLRZCIYW        AUFE6HLRZCIYW       AUFE6HLRZCIYW 24
## 55       AVKMFV9CZOMLR                 <NA>       AVKMFV9CZOMLR 24
## 56       AVOF14300525D        AVOF14300525D       AVOF14300525D 24
## 57       AYY7JSADOSGXL        AYY7JSADOSGXL       AYY7JSADOSGXL 24
## 58               barri       A3TW179ZOGG8JG               barri 24
## 59              Beclu                 <NA>              Beclu 24
## 60                Ben       A2P3V5VELJXAMP                Ben 24
## 61          bma5rocks       A2M8H9ZEAX8UUV          bma5rocks 24
## 62             Brandi       A2YW8EMCUIPUKO             Brandi 24
## 63            Brandon       A29KDPDITZSBRQ            Brandon 24
```

```
## 64                Brett       A1S8KMGKJACGVO                Brett 24
## 65                   by       A3DK4DDLEWCMM1                   by 24
## 66    Cameron Goldsmith        AQGJ6V57FZ7L7    Cameron Goldsmith 24
## 67              Candice        A4H31418X7MBU              Candice 24
## 68                Carol       A3T0ZNQ2KRCC66                Carol 24
## 69            cb1_high1                 <NA>            cb1_high1 24
## 70            cb1_high2                 <NA>            cb1_high2 24
## 71            cb1_high3                 <NA>            cb1_high3 24
## 72            cb1_high4                 <NA>            cb1_high4 24
## 73            cb2_high1                 <NA>            cb2_high1 24
## 74            cb2_high2                 <NA>            cb2_high2 24
## 75            cb4_high4                 <NA>            cb4_high4 24
## 76               Cherry       A37CLHPN6SWGBF               Cherry 24
## 77                Chloe        AOMFEAWQHU3D8                Chloe 24
## 78               chriss       A3IJ7SO3Z5PGT1               chriss 24
## 79            Christina       A1QEQ64E22TNUZ            Christina 24
## 80  Christine Parkerson       A2VX8NJYYQS9VC  Christine Parkerson 24
## 81                 Cole       A2XS49LRNX8PD4                 Cole 24
## 82                   CS       A3U2S6OPJNS4AQ                   CS 24
## 83                    d       A369K7I5SXBPNC                    d 24
## 84                  DAN       A3NFGEUZAH9V5G                  DAN 24
## 85                Dave        A1ADAWW4IHPCQ7                Dave 24
## 86                   DC       A1XUXY84RLKDTW                   DC 24
## 87               Dillon        A1KC342L9WAQ2               Dillon 24
## 88                Donna       A24NKUCAX1LURL                Donna 24
## 89              DOUGLAS            INSTITUTE              DOUGLAS 24
## 90                   DS        AAA96Z6PLD746                   DS 24
## 91                    E        ABN102C3INHDF                    E 24
## 92                   ed       A2ULVVSHLNVHAY                   ed 24
## 93                  Ely       A3TDA1NWHNQFE3                  Ely 24
## 94                Emily       A1O6C7F5OKGBVU                Emily 24
## 95                 Erin        AG8PCITDPLLKU                 Erin 24
## 96                Ethan       A2WPHVMLLEV5ZB                Ethan 24
## 97               Fatima       A3C6V8AOIJXIV0               Fatima 24
## 98               female       A3V9FP3JACXUCH               female 24
## 99                   FL       A3HSONMX6LNZR4                   FL 24
## 100                GAIL       A1VV1SOHHGUXPM                GAIL 24
## 101                  gi       A2EY0J9LOPU3TU                  gi 24
## 102                 goo       A3T5UAYFKJBVTA                 goo 24
## 103               Gregg       A3GNQDFPZALU92               Gregg 24
## 104                  gv        AJBB72MM58BBY                  gv 24
## 105                   H        AHWF0QIK9VTDS                   H 24
## 106               Henry       A14T9T8AJK1XZI               Henry 24
## 107                   j       A1FLEFIVFT809G                   j 24
## 108                   J        AZUTZP7QKWWXN       AZUTZP7QKWWXN 24
## 109                  j1       A2NC71VJJ9KTVC                  j1 24
## 110                  J3       A39PRBPSKA7A49                  J3 24
## 111                 Jen       A2O2Q0LVWOWJ1G                 Jen 24
## 112            jennifer       A36QGCT3MMXC4Q            jennifer 24
## 113            Jennifer       A25F4VG7W3BTWV            Jennifer 24
## 114    Jennifer Tunnell       A39HRVAJ5ISRL7    Jennifer Tunnell 24
## 115        Jesse Harris       A1NWYNEBQYP836        Jesse Harris 24
## 116        Jewel Julian        A2TKHXY755FAM        Jewel Julian 24
## 117                 jim       A277UPR6JWGMWY                 jim 24
```

```
## 118                 Jim               <NA>                 Jim 24
## 119                  jl       A2RCE6APFZ8EAD                  jl 24
## 120                  JM        AF0M3066S5UF1                  JM 24
## 121                 jmm       A2CIRSDBG6NQU1                 jmm 24
## 122             jocelyn       A2JZTR2ZSALGH4             jocelyn 24
## 123                 joe       A3B6ANBQU81J1J                 joe 24
## 124               Joel        A16E38HXNQBLCU               Joel 24
## 125                john       A1W3PGRM5NNVW8                john 24
## 126           Johnathan       A1MXV3UQA0G711           Johnathan 24
## 127                  JP       A2IFRL6OD4R8XZ                  JP 24
## 128                judy       A3PQYTQ44C8UQ4                judy 24
## 129        juliastelter       A1WP99J2Y8YMXM        juliastelter 24
## 130               Julie       A1POAW2RQBTBXB               Julie 24
## 131              Justin       A3219XJ5IMYROB              Justin 24
## 132                Karl       A1BHW4QNL2D9EF                Karl 24
## 133               katie        ADUOSLOPPCDYX               katie 24
## 134              Kaylee       A3IV3ZORENPEZL              Kaylee 24
## 135               Keith       A1DUH3RLI00YQM               Keith 24
## 136            Kristine       A3LMCFV79DBWJR            Kristine 24
## 137                  ky        AGWIWHORY2BWO                  ky 24
## 138                Lexy       A2NYHRY8ELO4FW                Lexy 24
## 139                  ll        A3O4WYGKH31OX                  ll 24
## 140                  lm        AHQXCSWPOU6KO                  lm 24
## 141                  LS       A3VZDRWPGJH46L                  LS 24
## 142                Luke       A3Q43S8UL38MSK                Luke 24
## 143                   m       A2LRMKO5AC21ZC                   m 24
## 144                   M        A726EC4AQNNNC                   M 24
## 145            Madeline       A3G6PURCIIY2VQ            Madeline 24
## 146                Male        ATCW7GWYA922U                Male 24
## 147              marcia       A3TD4VJPN33IWZ              marcia 24
## 148                Mary       A3FNC8ELMK8YJA                Mary 24
## 149                  MD       A2C7EYRPWL8B3L                  MD 24
## 150                  me       A33QOAS590VTQ5                  me 24
## 151                 me1       A1CM4AUO52HWDO                 me1 24
## 152                 Mel       A3OHRFXPEKQMIE                 Mel 24
## 153                 Mia       A3DGF3CNJVTOQF                 Mia 24
## 154  Michele Richardson       A2BZJHI275NG4P  Michele Richardson 24
## 155                mike       A1MUZEHEFOQBRQ                mike 24
## 156                 MJB       A3OOD9IMOHPPFQ                 MJB 24
## 157                 Moe        A1SWGSFIMZJQE                 Moe 24
## 158                Mona       A1ML6B8SBSWN7G                Mona 24
## 159                   N       A1UHJ56MF9CSQM                   N 24
## 160    Nobody Important       A2R6UQCQ8T3RZE   Nobody Important 24
## 161           Nola Nola       A1CC9FGFOGRBUY           Nola Nola 24
## 162             noname3              noname3             noname3 24
## 163             noname4               <NA>             noname4 24
## 164         Participant       A11JC9IZ7WY72T         Participant 24
## 165                 Pat       A2G3G60TB7HCJE                 Pat 24
## 166   penguincatcher123  COPIEDALIDCAKF04FG9   penguincatcher123 24
## 167               promi       A2MS1GQLGAX9FZ               promi 24
## 168                   r       A1N1ULK71RHVMM                   r 24
## 169              randal       A3Q4IMJOLCLNPY              randal 24
## 170                Raul       A3L9K9DZI8LFM4                Raul 24
## 171                 Ray       A1ZNRERDDZZHGW                 Ray 24
```

```
## 172              Reagan       A2FCHLMKIAOX2D              Reagan 24
## 173               Rema        A1EPE4DKRJMWER               Rema 24
## 174                 RF        A2DBNCUF4O3EMG                 RF 24
## 175              Robin        A2UNFOZWOY7GK3              Robin 24
## 176                 RR        A1R8I1H63TWK79                 RR 24
## 177                 RT        A11N2XYG9AGJ3O                 RT 24
## 178             rusong        A3VR4DVGAQVUVQ             rusong 24
## 179              Saber        A3KHKFLJZNOOQI              Saber 24
## 180                Sam         AV1IVFVS4VWZP                Sam 24
## 181                 sb         AWN8SISULPG1E                 sb 24
## 182                 SD        A2EYOJ9LOPU3TU                 SD 24
## 183               sean        A3KW8XFFLI5TO5               sean 24
## 184              shawn        A1MGNXJMXRWTLY              shawn 24
## 185      sherri sanders        A30O49UKY8LOZ7      sherri sanders 24
## 186   stacey schmeidel        A3HEON28SY1PPN   stacey schmeidel 24
## 187             Steffi        A2YQXWI76OKK7N             Steffi 24
## 188 stephanie hatfield         AE8QS3RQ1G1WA stephanie hatfield 24
## 189              Steve         AKX5RHHO8BIUX              Steve 24
## 190             Swetha        A2H9V3NM6OR295             Swetha 24
## 191                  T        A34P1UTHMRAAL1                  T 24
## 192         tacophopia        A1YX741QQSMKPS         tacophopia 24
## 193               TARA         AFH5Q22HIXRA2               TARA 24
## 194        Taylor Many        A1W4RCPZK3UXCM        Taylor Many 24
## 195                 TB         AXKNMOECIV3ZA                 TB 24
## 196                thy        A3BFST7DO3LRQ6                thy 24
## 197   Timothy Buchanan         A8AHU8NCH1P9X   Timothy Buchanan 24
## 198                 TT        A1SOFLJOEQB591                 TT 24
## 199                 VB         AQOOFSO4DWJOL                 VB 24
## 200     Virginia Sorto                    2      Virginia Sorto 24
## 201                viv        A1DIOT6SGONJU6                viv 24
## 202    Walter Tressler        A26O14Y7FVMDYL    Walter Tressler 24
## 203              weerd        A14OQ52EFQAN2W              weerd 24
## 204             WORKER         A1PEUF8MH49BX             WORKER 24
## 205                  y        A1F17GSMFCANQD                  y 24
## 206                yhh        A2EYOJ9LOPU3TU                yhh 24
## 207               <NA>        A10LHWALI4BZPC      A10LHWALI4BZPC 24
## 208               <NA>        A14LOABUGAITBM      A14LOABUGAITBM 24
## 209               <NA>        A187RIRDOFNE1B      A187RIRDOFNE1B 24
## 210               <NA>        A1BQI9JKLVCILO      A1BQI9JKLVCILO 24
## 211               <NA>        A1XCQA727LJVL2      A1XCQA727LJVL2 24
## 212               <NA>        A23HZ18KTCK2DA      A23HZ18KTCK2DA 24
## 213               <NA>        A27LC4TGHQ7GCG      A27LC4TGHQ7GCG 24
## 214               <NA>        A2JM1IVEZCXX3U      A2JM1IVEZCXX3U 24
## 215               <NA>        a2mqmmcvzjtfl2      a2mqmmcvzjtfl2 24
## 216               <NA>         A2QC2W3GCGKHB       A2QC2W3GCGKHB 24
## 217               <NA>        A2VB2MZFRO7VN5      A2VB2MZFRO7VN5 24
## 218               <NA>        A314XJY8V1YL12      A314XJY8V1YL12 24
## 219               <NA>        A3D5VZQMO79JQJ      A3D5VZQMO79JQJ 24
## 220               <NA>        A3GTPW3MYSLMR7      A3GTPW3MYSLMR7 24
## 221               <NA>        A3TSOKMBDEEUVX      A3TSOKMBDEEUVX 24
## 222               <NA>         A77OOB1ZAGJ4G       A77OOB1ZAGJ4G 24
## 223               <NA>         AHB886J4P46JW       AHB886J4P46JW 24
## 224               <NA>         AK9CDGMQCBSIU       AK9CDGMQCBSIU 24
## 225               <NA>         AKFNJG4UMP6MX       AKFNJG4UMP6MX 24
```

```
## 226                <NA>        ATNNZOOIRXT1B        ATNNZOOIRXT1B 24
## 227                <NA>        AU01P6YB9J5JX        AU01P6YB9J5JX 24
## 228                <NA>        AVWU9JDY5G81E        AVWU9JDY5G81E 24
## 229                <NA>        AXB86SVXU5SEG        AXB86SVXU5SEG 24
## 230                <NA> COPIEDA1C5J40P3I65EI COPIEDA1C5J40P3I65EI 24
## 231                <NA>              noname2              noname2 24
## 232                <NA>                 <NA>                  NA1 24
```

```
#write.csv(all_c, file="test_expect.csv")
```

JOLs

Bind and tidy wide for plotting

```
all_jol <- rbind(dataset1_jol, dataset2_jol, dataset3_jol, dataset4_jol)

all_jol_wide= all_jol %>%
  tidyr::pivot_wider(names_from = "TypeFace", values_from = "jols") %>%
  dplyr::mutate(Difference=atypic_slider.response-normal_slider.response)

all_jol_mean_wide <- all_jol_wide %>%
  dplyr::group_by(cond) %>%
  dplyr::summarise(mean=mean(Difference))
```

RTs

```
rt_all <- rbind(dataset1_rt, dataset2_rt, dataset3_rt, dataset4_rt)

rt_all1<- rt_all %>%
  dplyr::group_by(new_id1, font, cond) %>%
  dplyr::mutate(rt=mouse_4.time*1000) %>%
dplyr::mutate(sdabove = mean(rt, na.rm=TRUE) +  2.5*sd(rt, na.rm=TRUE)) %>%
    dplyr::filter(rt > 150 || rt > sdabove) %>%
  dplyr::summarise(mean_rt= mean(log(rt), na.rm=TRUE)) %>%
   mutate(font=ifelse(font=="flu", "Arial", "Sans Forgetica")) %>%
  mutate(new_id1=ifelse(is.na(new_id1), "na1", new_id1)) %>%
  ungroup()
```

```
## `summarise()` has grouped output by 'new_id1', 'font'. You can override using the `.groups` argument
```

```
rt_all_wide <- rt_all1 %>%
  tidyr::pivot_wider(names_from = "font", values_from = "mean_rt")%>%
  dplyr::mutate(Difference= `Sans Forgetica` - Arial)


rt_all_wide_mean <- rt_all_wide %>%
  dplyr::group_by(cond) %>%
  dplyr::summarise(mean=mean(Difference))
```

Analysis

We employed 2 x 2 MIXED ANOVA.

Cued Recall

LRD

LRD can be used as a shiny application or as a package. Below shows how to score cued recall automatically using the lrd package. We take the trial level data and then aggregate across fonts to get proportion correct for Sans Forgetica and Arial, seperately.

```
library(lrd)
```

```
##
## Attaching package: 'lrd'
```

```
## The following object is masked from 'package:base':
##
##     kappa
```

```
all_c$textbox.text[is.na(all_c$textbox.text)] <- "" # does not work if NAs exists

all_c<-as.data.frame(all_c) # needs to be data frame

all_c$trial_id<-rep(1:5568) # needs to have unique rows for some reason
# run lrd
scored_cued = prop_correct_cued(all_c,
                                responses = "textbox.text",
                                id = "new_id",
                                id.trial = "trial_id",
                                key = "targ1",
                                key.trial =  "trial_id",
                                cutoff = 3,
                                group.by = c("font"))

#recall_highlow<-scored_cued$DF_Participant #if you want total proportion

recall_highlow<-scored_cued$DF_Scored


### get total counts for Ps
#recall_highlow_score<- recall_highlow %>%
 # group_by(new_id)%>%
  #summarise(totalCorrect=sum(Scored)) %>%
  #ungroup()


#recall_highlow_score_font<- recall_highlow %>%
  #dplyr::group_by(Sub.ID, font, cond)%>%
 # dplyr::summarise(correct=sum(Scored)) %>%
  #dplyr::ungroup() %>%
  #pivot_wider(names_from = "font", values_from = "correct") %>%
  #right_join(., recall_highlow_score) %>%
  #dplyr::mutate(ArialProp=flu/12, SFProp=SF/12, totalProp=totalCorrect/24, totalArialProp=flu/24, tota

#write.csv(recall_highlow_score_font, file="expt2_wife_prop_total.csv")
#write.csv(recall_highlow, file="expt2_long_lrd.csv")

#write.csv(recall_highlow_score_font, file="expt2_wide_counts_prop_summary.csv")
```

Aggregate

```
recall_highlow_agg_wide <- recall_highlow %>%
  dplyr::group_by(Sub.ID, font, cond) %>%
  dplyr::summarise(Proportion.Correct=mean(Scored))%>%
  tidyr::pivot_wider(names_from = "font", values_from = "Proportion.Correct") %>%
  dplyr::mutate(Difference=SF - flu)
```

```
## `summarise()` has grouped output by 'Sub.ID', 'font'. You can override using the `.groups` argument.
```

```
recall_means_wide <- recall_highlow_agg_wide %>%
  dplyr::group_by(cond) %>%
  dplyr::summarise(mean=mean(Difference))


write.csv(recall_highlow_agg_wide, file="recall_expt2_summary.csv")


recall_highlow_agg <- recall_highlow %>%
 dplyr::group_by(Sub.ID, font, cond) %>%
  dplyr::summarise(Proportion.Correct=mean(Scored))
```

```
## `summarise()` has grouped output by 'Sub.ID', 'font'. You can override using the `.groups` argument.
```

```
write.csv(recall_highlow_agg, file="expt2_long_summary.csv")
```

ANOVA

```
#ANOVA

a1 <- aov_ez("Sub.ID", "Proportion.Correct", recall_highlow_agg,
            within=c("font"), between=c("cond")) # mixed
```

```
## Converting to factor: cond
```

```
## Contrasts set to contr.sum for the following variables: cond
```

```
summary(a1)
```

```
##
## Univariate Type III Repeated-Measures ANOVA Assuming Sphericity
##
##             Sum Sq num Df Error SS den Df F value    Pr(>F)
## (Intercept) 65.876      1  25.9410    230 584.080 < 2.2e-16 ***
## cond         4.016      1  25.9410    230  35.606 9.057e-09 ***
## font         0.238      1   3.7609    230  14.531 0.0001771 ***
## cond:font    0.168      1   3.7609    230  10.284 0.0015325 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
a1
```

```
## Anova Table (Type 3 tests)
##
## Response: Proportion.Correct
##      Effect     df  MSE         F  ges p.value
## 1      cond 1, 230 0.11 35.61 *** .119   <.001
## 2      font 1, 230 0.02 14.53 *** .008   <.001
```

```
## 3 cond:font 1, 230 0.02  10.28 ** .006    .002
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '+' 0.1 ' ' 1
```

Main Effects

```
font <- emmeans(a1, ~ font)
```

fonts

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

```
##  font emmean     SE  df lower.CL upper.CL
##  flu   0.354 0.0167 295    0.321    0.387
##  SF    0.399 0.0167 295    0.367    0.432
##
## Results are averaged over the levels of: cond
## Warning: EMMs are biased unless design is perfectly balanced
## Confidence level used: 0.95
```

```
cond <- emmeans(a1, ~ cond)
```

Testing Effect

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

```
##  cond                 emmean    SE  df lower.CL upper.CL
##  High Test Expectancy  0.470 0.022 230    0.426    0.513
##  Low Test Expectancy   0.284 0.022 230    0.240    0.327
##
## Results are averaged over the levels of: font
## Warning: EMMs are biased unless design is perfectly balanced
## Confidence level used: 0.95
```

```
Within_Fitted_Interaction <- emmeans(a1, ~ font|cond)

Within_Fitted_Interaction
```

Testing Interaction

```
## cond = High Test Expectancy:
##  font emmean     SE  df lower.CL upper.CL
##  flu   0.466 0.0236 295    0.420    0.513
##  SF    0.473 0.0236 295    0.427    0.520
##
## cond = Low Test Expectancy:
##  font emmean     SE  df lower.CL upper.CL
##  flu   0.242 0.0236 295    0.196    0.289
##  SF    0.325 0.0236 295    0.279    0.372
##
## Warning: EMMs are biased unless design is perfectly balanced
## Confidence level used: 0.95
```

Planned Comparisons

```
pairs(Within_Fitted_Interaction) ## pairwise comparison with no correction
```

```
## cond = High Test Expectancy:
##  contrast estimate     SE  df t.ratio p.value
##  flu - SF -0.00718 0.0168 230 -0.428  0.6692
##
## cond = Low Test Expectancy:
##  contrast estimate     SE  df t.ratio p.value
##  flu - SF -0.08333 0.0168 230 -4.963  <.0001
```

Effect sizes

```
### get d_avg for high

recall_high <- recall_highlow_agg%>%
  #dplyr::group_by(Sub.ID, font, cond) %>%
  #dplyr::summarise(Proportion.Correct=mean(Scored))%>%
  tidyr::pivot_wider(names_from = "font", values_from = "Proportion.Correct")%>%
dplyr::filter(cond=="High Test Expectancy")%>%
  dplyr::ungroup() %>%
  summarise(mean1=mean(flu), sd1=sd(flu), mean2=mean(SF), sd2=sd(SF))




h=d.dep.t.avg(m1 = recall_high$mean1, m2 = recall_high$mean2, sd1 = recall_high$sd1,
                sd2 = recall_high$sd2, n = 116, a = .05)


#### get d_avg for low


recall_low <- recall_highlow_agg %>%
  #dplyr::group_by(Sub.ID, font, cond) %>%
 # dplyr::summarise(mean_acc=mean(Scored))%>%
  tidyr::pivot_wider(names_from = "font", values_from = "Proportion.Correct")%>%
dplyr::filter(cond=="Low Test Expectancy")%>%
  ungroup() %>%
  summarise(mean1=mean(flu), sd1=sd(flu), mean2=mean(SF), sd2=sd(SF))


l=d.dep.t.avg(m1 = recall_low$mean1, m2 = recall_low$mean2, sd1 = recall_low$sd1,
                sd2 = recall_low$sd2, n = 116, a = .05)


h
```

```
## $d
## [1] -0.02597187
##
## $dlow
## [1] -0.2079245
```

```
##
## $dhigh
## [1] 0.1560935
##
## $M1
## [1] 0.4662356
##
## $sd1
## [1] 0.2890887
##
## $se1
## [1] 0.02684121
##
## $M1low
## [1] 0.4130684
##
## $M1high
## [1] 0.5194029
##
## $M2
## [1] 0.4734195
##
## $sd2
## [1] 0.2641181
##
## $se2
## [1] 0.02452275
##
## $M2low
## [1] 0.4248447
##
## $M2high
## [1] 0.5219944
##
## $n
## [1] 116
##
## $df
## [1] 115
##
## $estimate
## [1] "$d_{av}$ = -0.03, 95\\% CI [-0.21, 0.16]"
```

```
l
```

```
## $d
## [1] -0.3637922
##
## $dlow
## [1] -0.5509896
##
## $dhigh
## [1] -0.1751087
##
## $M1
```

```
## [1] 0.2420977
##
## $sd1
## [1] 0.2299912
##
## $se1
## [1] 0.02135415
##
## $M1low
## [1] 0.1997992
##
## $M1high
## [1] 0.2843962
##
## $M2
## [1] 0.325431
##
## $sd2
## [1] 0.2281459
##
## $se2
## [1] 0.02118281
##
## $M2low
## [1] 0.283472
##
## $M2high
## [1] 0.3673901
##
## $n
## [1] 116
##
## $df
## [1] 115
##
## $estimate
## [1] "$d_{av}$ = -0.36, 95\\% CI [-0.55, -0.18]"
```

Bayesian Analysis

```
recall_highlow$Sub.ID<-rep(1:232, each=2)
recall_highlow$Sub.ID<-as.factor(recall_highlow$Sub.ID)
recall_highlow$cond<-as.factor(recall_highlow$cond)
recall_highlow$font<-as.factor(recall_highlow$font)
bfcue = anovaBF(Proportion.Correct ~ cond*font + Sub.ID, recall_highlow,
          whichRandom=c("new_id1"))
```

JOLs

ANOVA

```
#ANOVA

jol_a <- aov_ez("new_id1", "jols", all_jol,
```

```
            within=c("TypeFace"), between=c("cond")) # mixed
```

```
## Converting to factor: cond
```

```
## Contrasts set to contr.sum for the following variables: cond
```

```
summary(jol_a)
```

```
##
## Univariate Type III Repeated-Measures ANOVA Assuming Sphericity
##
##                 Sum Sq num Df Error SS den Df  F value    Pr(>F)
## (Intercept)   1540704      1    68159    229 5176.437 < 2.2e-16 ***
## cond             4038      1    68159    229   13.566 0.0002875 ***
## TypeFace        26494      1    69700    229   87.046 < 2.2e-16 ***
## cond:TypeFace    4156      1    69700    229   13.654 0.0002749 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Testing Interaction

```
Within_Fitted_Interaction <- emmeans(jol_a, ~ TypeFace|cond)

Within_Fitted_Interaction
```

```
## cond = High Test Expectancy:
##  TypeFace               emmean   SE  df lower.CL upper.CL
##  atypic_slider.response   50.2 1.61 458     47.0     53.4
##  normal_slider.response   59.4 1.61 458     56.2     62.5
##
## cond = Low Test Expectancy:
##  TypeFace               emmean   SE  df lower.CL upper.CL
##  atypic_slider.response   50.1 1.62 458     46.9     53.3
##  normal_slider.response   71.3 1.62 458     68.1     74.4
##
## Warning: EMMs are biased unless design is perfectly balanced
## Confidence level used: 0.95
```

Effect Sizes

```
recall_highjol <- all_jol %>%
  dplyr::group_by(new_id1, TypeFace, cond) %>%
  dplyr::summarise(mean_jol=mean(jols))%>%
  tidyr::pivot_wider(names_from = "TypeFace", values_from = "mean_jol")%>%
dplyr::filter(cond=="High Test Expectancy")%>%
  ungroup() %>%
  summarise(mean1=mean(normal_slider.response ), sd1=sd(normal_slider.response), mean2=mean(atypic_slid
```

```
## `summarise()` has grouped output by 'new_id1', 'TypeFace'. You can override using the `.groups` argu
```

```
h_jol=d.dep.t.avg(m1 = recall_highjol$mean1, m2 = recall_highjol$mean2, sd1 = recall_highjol$sd1,
                sd2 = recall_highjol$sd2, n = 116, a = .05)

recall_lowjol <- all_jol %>%
  dplyr::group_by(new_id1, TypeFace, cond) %>%
```

```
  dplyr::summarise(mean_jol=mean(jols))%>%
  tidyr::pivot_wider(names_from = "TypeFace", values_from = "mean_jol")%>%
dplyr::filter(cond=="Low Test Expectancy")%>%
  ungroup() %>%
  summarise(mean1=mean(normal_slider.response ), sd1=sd(normal_slider.response), mean2=mean(atypic_slid
```

```
## `summarise()` has grouped output by 'new_id1', 'TypeFace'. You can override using the `.groups` argu
```

```
l_jol=d.dep.t.avg(m1 = recall_lowjol$mean1, m2 = recall_lowjol$mean2, sd1 = recall_lowjol$sd1,
                  sd2 = recall_lowjol$sd2, n = 115, a = .05)


h_jol
```

```
## $d
## [1] 0.7186438
##
## $dlow
## [1] 0.5131285
##
## $dhigh
## [1] 0.9216512
##
## $M1
## [1] 59.36598
##
## $sd1
## [1] 24.3853
##
## $se1
## [1] 2.264118
##
## $M1low
## [1] 54.8812
##
## $M1high
## [1] 63.85077
##
## $M2
## [1] 50.21896
##
## $sd2
## [1] 1.071046
##
## $se2
## [1] 0.09944414
##
## $M2low
## [1] 50.02198
##
## $M2high
## [1] 50.41594
##
## $n
```

```
## [1] 116
##
## $df
## [1] 115
##
## $estimate
## [1] "$d_{av}$ = 0.72, 95\\% CI [0.51, 0.92]"
```

```
l_jol
```

```
## $d
## [1] 1.649358
##
## $dlow
## [1] 1.366439
##
## $dhigh
## [1] 1.929031
##
## $M1
## [1] 71.27712
##
## $sd1
## [1] 24.64311
##
## $se1
## [1] 2.297982
##
## $M1low
## [1] 66.72483
##
## $M1high
## [1] 75.8294
##
## $M2
## [1] 50.13296
##
## $sd2
## [1] 0.9961423
##
## $se2
## [1] 0.09289075
##
## $M2low
## [1] 49.94894
##
## $M2high
## [1] 50.31697
##
## $n
## [1] 115
##
## $df
## [1] 114
##
```

```
## $estimate
## [1] "$d_{av}$ = 1.65, 95\\% CI [1.37, 1.93]"
```

```
pairs(Within_Fitted_Interaction) ## pairwise comparison with no correction
```

```
## cond = High Test Expectancy:
##  contrast                                          estimate   SE  df t.ratio
##  atypic_slider.response - normal_slider.response    -9.15 2.29 229 -3.993
##  p.value
##  0.0001
##
## cond = Low Test Expectancy:
##  contrast                                          estimate   SE  df t.ratio
##  atypic_slider.response - normal_slider.response   -21.14 2.30 229 -9.190
##  p.value
##  <.0001
```

RTs

ANOVA

```
#ANOVA

rt_a <- aov_ez("new_id1", "mean_rt",rt_all1,
            within=c("font"), between=c("cond")) # mixed
```

```
## Converting to factor: cond
```

```
## Contrasts set to contr.sum for the following variables: cond
```

```
summary(rt_a)
```

```
##
## Univariate Type III Repeated-Measures ANOVA Assuming Sphericity
##
##              Sum Sq num Df Error SS den Df    F value    Pr(>F)
## (Intercept) 25078.8      1  183.020    230 31516.3740 < 2.2e-16 ***
## cond           13.5      1  183.020    230    17.0245 5.160e-05 ***
## font            0.5      1    3.847    230    27.7350 3.192e-07 ***
## cond:font       0.0      1    3.847    230     0.3899     0.533
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
rt_a
```

```
## Anova Table (Type 3 tests)
##
## Response: mean_rt
##      Effect     df  MSE         F   ges p.value
## 1      cond 1, 230 0.80 17.02 ***  .068   <.001
## 2      font 1, 230 0.02 27.73 ***  .002   <.001
## 3 cond:font 1, 230 0.02      0.39 <.001    .533
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '+' 0.1 ' ' 1
```

Main Effects No interaction, but main effect

```
Within_font <- emmeans(rt_a, ~ font)
```

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

```
Within_font
```

```
##  font           emmean     SE  df lower.CL upper.CL
##  Arial            7.32 0.0418 240     7.24     7.40
##  Sans.Forgetica   7.38 0.0418 240     7.30     7.47
##
## Results are averaged over the levels of: cond
## Warning: EMMs are biased unless design is perfectly balanced
## Confidence level used: 0.95
```

```
Within_cond <- emmeans(rt_a, ~ cond)
```

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

```
Within_cond
```

```
##  cond                 emmean     SE  df lower.CL upper.CL
##  High Test Expectancy   7.52 0.0586 230     7.41     7.64
##  Low Test Expectancy    7.18 0.0586 230     7.07     7.30
##
## Results are averaged over the levels of: font
## Warning: EMMs are biased unless design is perfectly balanced
## Confidence level used: 0.95
```

Bayesian Analyis

Main-effect vs. interaction model

```
rt_all1$new_id1<-rep(1:232, each=2)

rt_all1$new_id1<-as.factor(rt_all1$new_id1)

rt_all1$cond<-as.factor(rt_all1$cond)

rt_all1$font<-as.factor(rt_all1$font)

bfrt = anovaBF(mean_rt ~ cond*font + new_id1, rt_all1,
          whichRandom="new_id1")

#Bayes factor analysis
#--------------
#[1] font + cond + new_id1 : 4.322303 ±4.6%

##Against denominator:
#  mean_rt ~ font + cond + font:cond + new_id1
#---
#Bayes factor type: BFlinearModel, JZS
```

Plot

Cued Recall

```
library(see)
library(ggrepel)

#load in violin plot code
source("https://gist.githubusercontent.com/benmarwick/2a1bb0133ff568cbe28d/raw/fb53bd97121f7f9ce947837e

bold <- element_text(face = "bold", color = "black", size = 14)

recall_highlow_agg <- recall_highlow_agg%>%
  dplyr::mutate(Typeface=ifelse(font=="SF", "Sans Forgetica", "Arial"))

#means by test and typeface
means = recall_highlow_agg %>%
  dplyr::group_by(cond, Typeface)%>%
  dplyr::summarise(mean=mean(Proportion.Correct))
```

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

```
# get withinsub CIs
sfarial_wsci=summarySEwithin(data = recall_highlow_agg, measurevar = "Proportion.Correct",
                        withinvars = "Typeface", betweenvars = "cond", idvar = "Sub.ID")
```

```
## Automatically converting the following non-factors to factors: cond, Typeface
```

```
recall_highlow_agg
```

```
## # A tibble: 464 x 5
## # Groups:   Sub.ID, font [464]
##    Sub.ID         font  cond                 Proportion.Correct Typeface
##    <chr>          <chr> <chr>                             <dbl> <chr>
##  1 ?              flu   High Test Expectancy             0.25   Arial
##  2 ?              SF    High Test Expectancy             0.5    Sans Forgetica
##  3 A10LHWALI4BZPC flu   High Test Expectancy             1      Arial
##  4 A10LHWALI4BZPC SF    High Test Expectancy             0.917  Sans Forgetica
##  5 A14LOABUGAITBM flu   High Test Expectancy             0      Arial
##  6 A14LOABUGAITBM SF    High Test Expectancy             0.0833 Sans Forgetica
##  7 A17Y73URHQYGAA flu   High Test Expectancy             1      Arial
##  8 A17Y73URHQYGAA SF    High Test Expectancy             0.917  Sans Forgetica
##  9 A182N7RLXGSCZG flu   Low Test Expectancy              0      Arial
## 10 A182N7RLXGSCZG SF    Low Test Expectancy              0      Sans Forgetica
## # ... with 454 more rows
```

```
fig2a <- ggplot(recall_highlow_agg,aes(x=Typeface,y=Proportion.Correct,fill=Typeface))+
  facet_grid(~cond) +
  #geom_flat_violin(position = position_nudge(x = .2, y = 0), alpha = .4,adjust=4)+
  geom_point(position=position_jitter(width = .15),size = 1, alpha = 0.2) +
  geom_boxplot(aes(x = Typeface, y = Proportion.Correct ),outlier.shape = NA,
               alpha = 0.3, width = .1, colour = "BLACK") +
    #stat_summary(fun="mean", geom="point", colour="darkred", size=3)+
  geom_line(data=sfarial_wsci,aes(y=Proportion.Correct, group=1), size=1)+
  geom_pointrange(data=sfarial_wsci, aes(y=Proportion.Correct, ymin=Proportion.Correct, ymax=Proportion
  scale_colour_brewer(palette = "Dark2")+
  scale_fill_brewer(palette = "Dark2") +
  labs(y = "Porportion Correct on Final Test", x = "Typeface") + theme(legend.position = "none") +
```

```
   geom_label_repel(data=sfarial_wsci, aes(y=Proportion.Correct, label=round(Proportion.Correct, 2)),se
   theme_cowplot(font_size=14)+
  theme(legend.position = "none") +
  theme(axis.title = bold)

# plot difference plots
fig2adiff <- ggplot(recall_highlow_agg_wide,aes(x=cond,y=Difference, fill=cond)) +
  geom_flat_violin(position = position_nudge(x = .2, y = 0), alpha = .4,adjust=4)+
  geom_point(position=position_jitter(width = .18),size = 1, alpha = 0.2) +
  geom_boxplot(aes(x = cond, y = Difference),outlier.shape = NA,
               alpha = 0.3, width = .1, colour = "BLACK") +
  stat_summary(fun.data="mean_cl_boot", colour="darkred", size=.8)+
  #geom_line(data=sfarial_wsci,aes(y=mean_acc, group=1), size=1)+
  #geom_pointrange(data=sfarial_wsci, aes(y=mean_acc, ymin=mean_acc-ci, ymax=mean_acc+ci), size=.5, col
  scale_colour_brewer(palette = "Accent")+
  scale_fill_brewer(palette = "Accent") +
  labs(y = "Test Difference (Sans Forgetica - Arial", x = "Test Expectancy")+
     theme_cowplot(font_size=14)+
  theme(legend.position = "none") +
  theme(axis.title =bold) +
  geom_hline(yintercept = 0, linetype="dotted") +
   geom_label_repel(data=recall_means_wide, aes(y=mean, label=round(mean, 2)), seed=42, box.padding=0.8


fig2a
```

```
fig2adiff
```

JOL

```
jol_rename <- all_jol %>%
  mutate(Typeface=ifelse(TypeFace=="atypic_slider.response", "Sans Forgetica", "Arial"))

means = jol_rename %>%
  dplyr::group_by(cond, Typeface)%>%
  dplyr::summarise(mean=mean(jols))
```

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

```
# get withinsubject CIs
sfgenjol_wsci= Rmisc::summarySEwithin(data = jol_rename, measurevar = "jols",
                      withinvars = "Typeface", betweenvars = "cond", idvar = "new_id1")
```

```
## Automatically converting the following non-factors to factors: cond, Typeface
```

```
fig2b <- ggplot(jol_rename,aes(x=Typeface,y=jols,fill=Typeface))+
  facet_grid(~cond) +
  #geom_flat_violin(position = position_nudge(x = .2, y = 0), alpha = .4,adjust=4)+
  geom_point(position=position_jitter(width = .15),size = 1, alpha = 0.2) +
  geom_boxplot(aes(x = Typeface, y = jols),outlier.shape = NA,
               alpha = 0.3, width = .1, colour = "BLACK") +
   #stat_summary(fun="mean", geom="point", colour="darkred", size=3)+
  geom_line(data=sfgenjol_wsci,aes(y=jols, group=1), size=1)+
  geom_pointrange(data=sfgenjol_wsci, aes(y=jols, ymin=jols, ymax=jols), size=.8, color="darkred")+
  scale_colour_brewer(palette = "Dark2")+
```

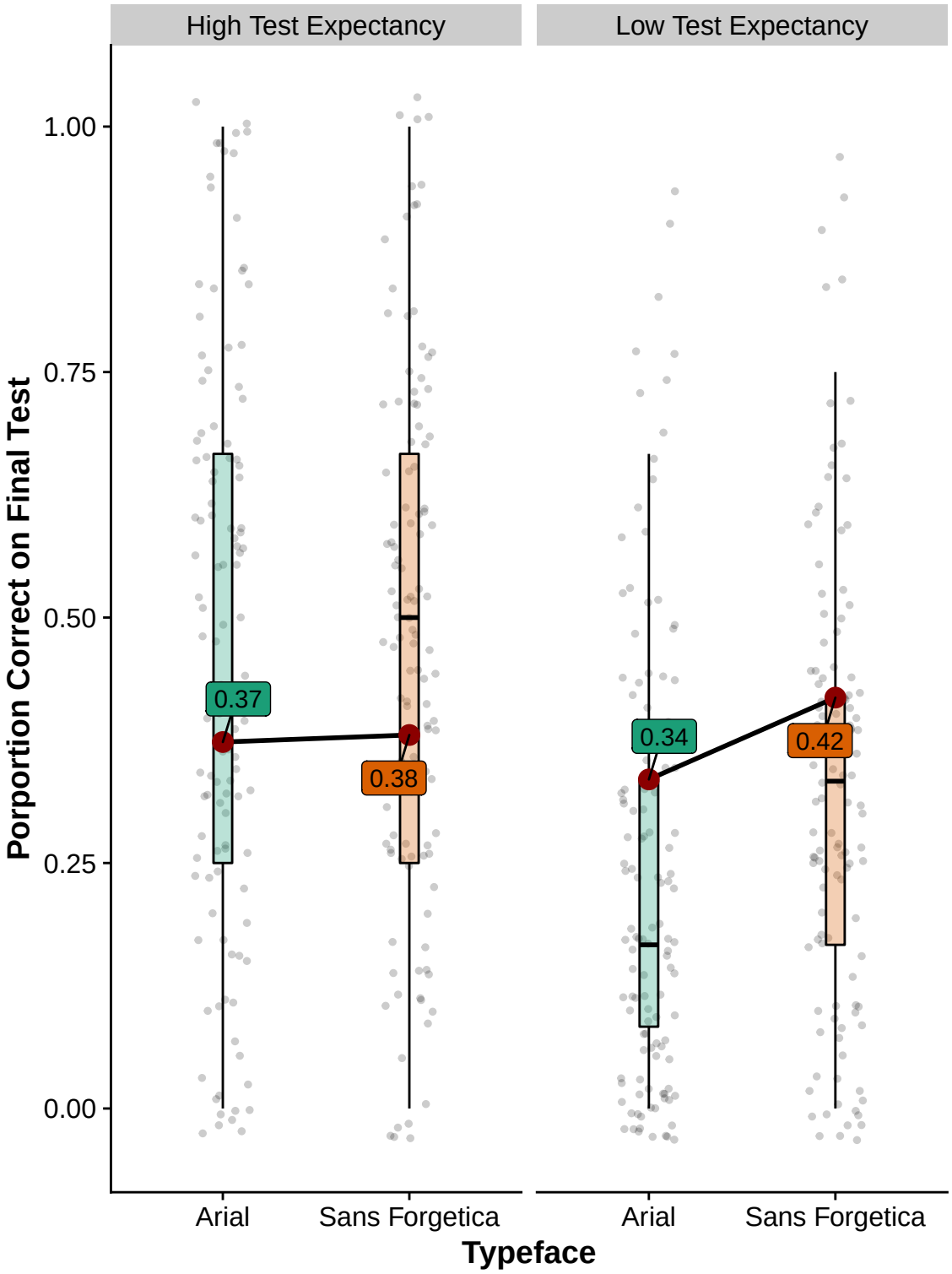


Figure 1: Raincloud plots (Allen et al., 2019) depicting raw data (dots), box plots, and half violin kernel desntiy plots, with mean (red dot) and within-participant 95 CIs. Cued recall accuracy as a function of test expectancy for Experiment 2.

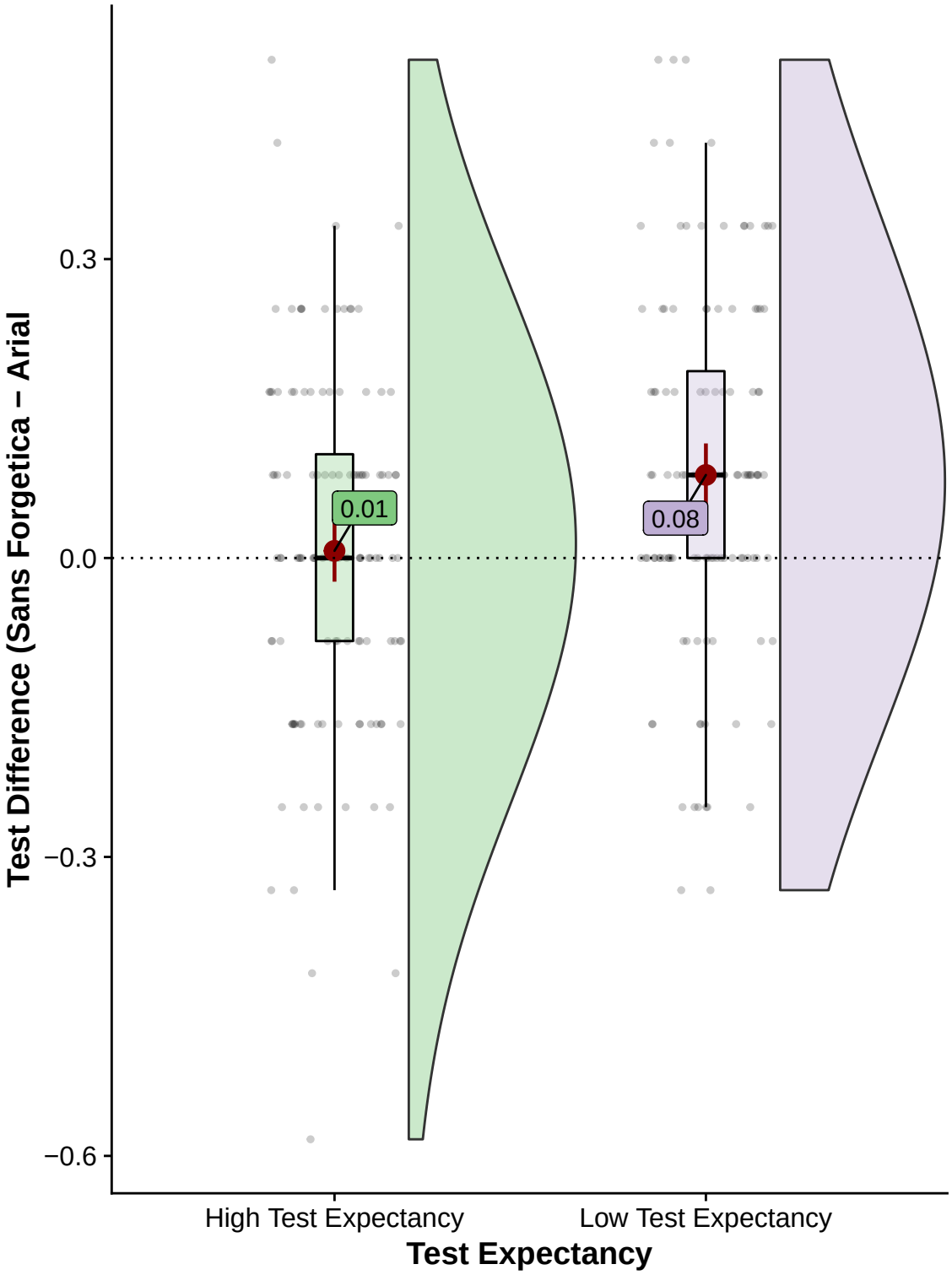


Figure 2: Raincloud plots (Allen et al., 2019) depicting raw data (dots), box plots, and half violin kernel desntiy plots, with mean (red dot) and within-participant 95 CIs. Cued recall accuracy as a function of test expectancy for Experiment 2.

```
  scale_fill_brewer(palette = "Dark2") +
  labs(y = "Judgements of Learning", x = "Typeface") +  theme(legend.position = "none")+
  geom_label_repel(data=sfgenjol_wsci, aes(y=jols, label=round(jols, 2)), seed = 42, box.padding = 0.8)
   theme_cowplot(font_size=14)+
  theme(legend.position = "none") +
  theme(axis.title = bold)


fig2b_diff <- ggplot(all_jol_wide,aes(x=cond,y=Difference,fill=cond)) +
  geom_flat_violin(position = position_nudge(x = .2, y = 0), alpha = .4,adjust=4)+
  geom_point(position=position_jitter(width = .15),size = 1, alpha = 0.2) +
  geom_boxplot(aes(x = cond, y = Difference),outlier.shape = NA,
               alpha = 0.3, width = .1, colour = "BLACK") +
  stat_summary(fun.data="mean_cl_boot", colour="darkred", size=.8)+
   #stat_summary(fun="mean", geom="point", colour="darkred", size=3)+
 # geom_line(data=sfgenjol_wsci,aes(y=jols, group=1), size=1)+
 # geom_pointrange(data=sfgenjol_wsci, aes(y=jols, ymin=jols-ci, ymax=jols+ci), size=.3, color="red")+
  scale_colour_brewer(palette = "Accent")+
  scale_fill_brewer(palette = "Accent") +
  labs(y = "JOL Difference (Sans Forgetica - Arial)", x = "Test Expectancy") +  theme(legend.position =
  geom_label_repel(data=all_jol_mean_wide, aes(y=mean , label=round(mean, 2)), min.segment.length = 0, s
   theme_cowplot(font_size=14)+
  theme(legend.position = "none") +
  geom_hline(yintercept = 0, linetype="dotted") +
  theme(axis.title = bold)


fig2b
```

```
fig2b_diff
```

RTs

```
means = rt_all1 %>%
  dplyr::group_by(cond, font)%>%
  dplyr::summarise(mean=mean(font))
```

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

```
# get withinsubject CIs
sfgenrt_wsci= Rmisc::summarySEwithin(data = rt_all1, measurevar = "mean_rt",
                      withinvars = "font", betweenvars = "cond", idvar = "new_id1")
```

```
## Automatically converting the following non-factors to factors: cond, font
```

```
fig2c <- ggplot(rt_all1,aes(x=font,y=mean_rt,fill=font))+ facet_grid(~cond) +
  #geom_flat_violin(position = position_nudge(x = .2, y = 0), alpha = .4,adjust=4)+
  geom_point(position=position_jitter(width = .15),size = 1, alpha = 0.2) +
  geom_boxplot(aes(x = font , y = mean_rt),outlier.shape = NA,
               alpha = 0.3, width = .1, colour = "BLACK") +
  #stat_summary(fun="mean", geom="point", colour="darkred", size=3)+
  geom_line(data=sfgenrt_wsci,aes(y=mean_rt, group=1), size=1)+
  geom_pointrange(data=sfgenrt_wsci, aes(y=mean_rt, ymin=mean_rt, ymax=mean_rt),size=.8, color="darkred
  scale_colour_brewer(palette = "Dark2")+
  scale_fill_brewer(palette = "Dark2") +
```

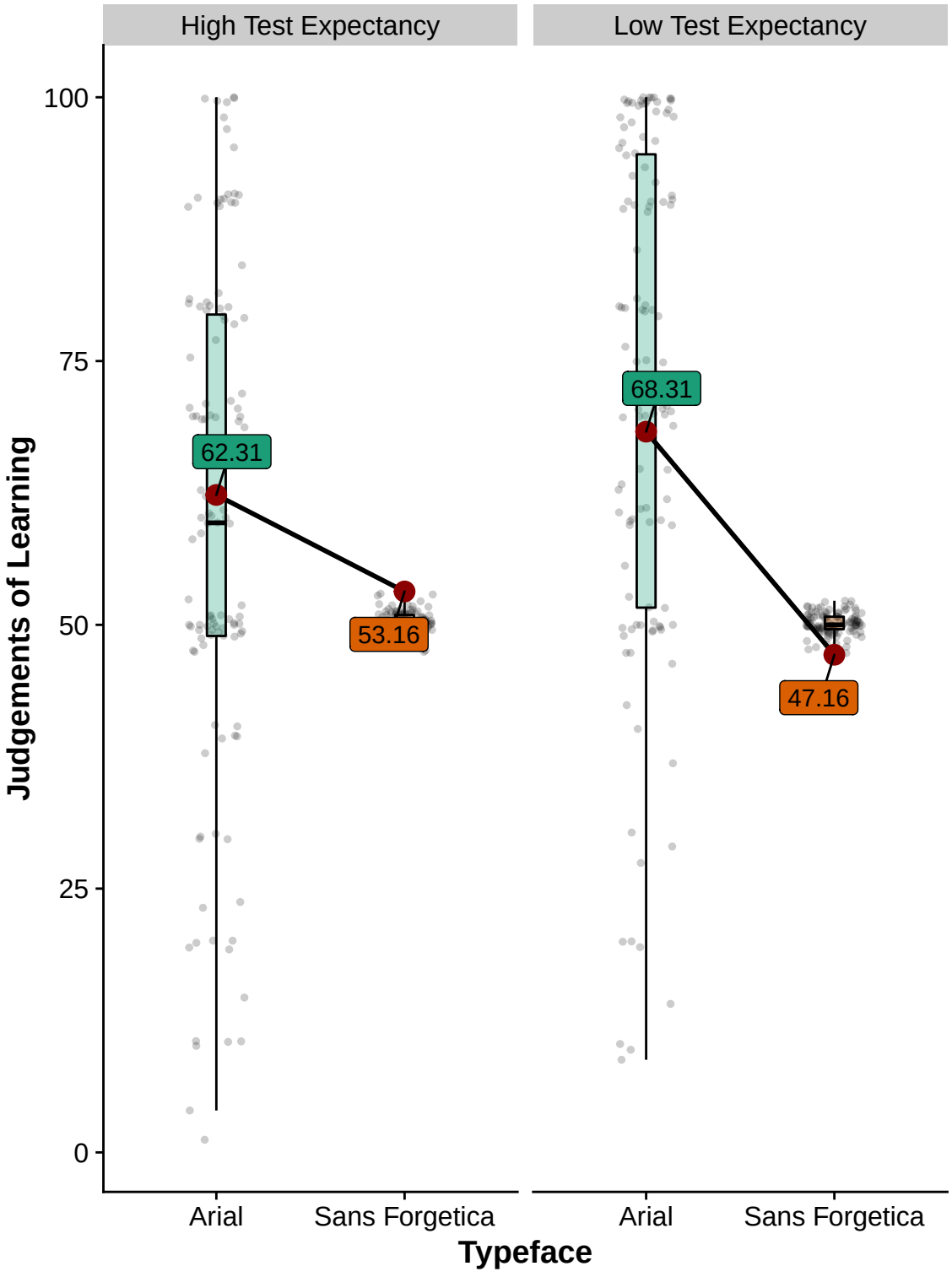


Figure 3: Raincloud plots (Allen et al., 2019) depicting raw data (dots), box plots, and half violin kernel desntiy plots, with mean (red dot) and within-participant 95 CIs. JOLs as a function of testing expectancy in Experiment 2.

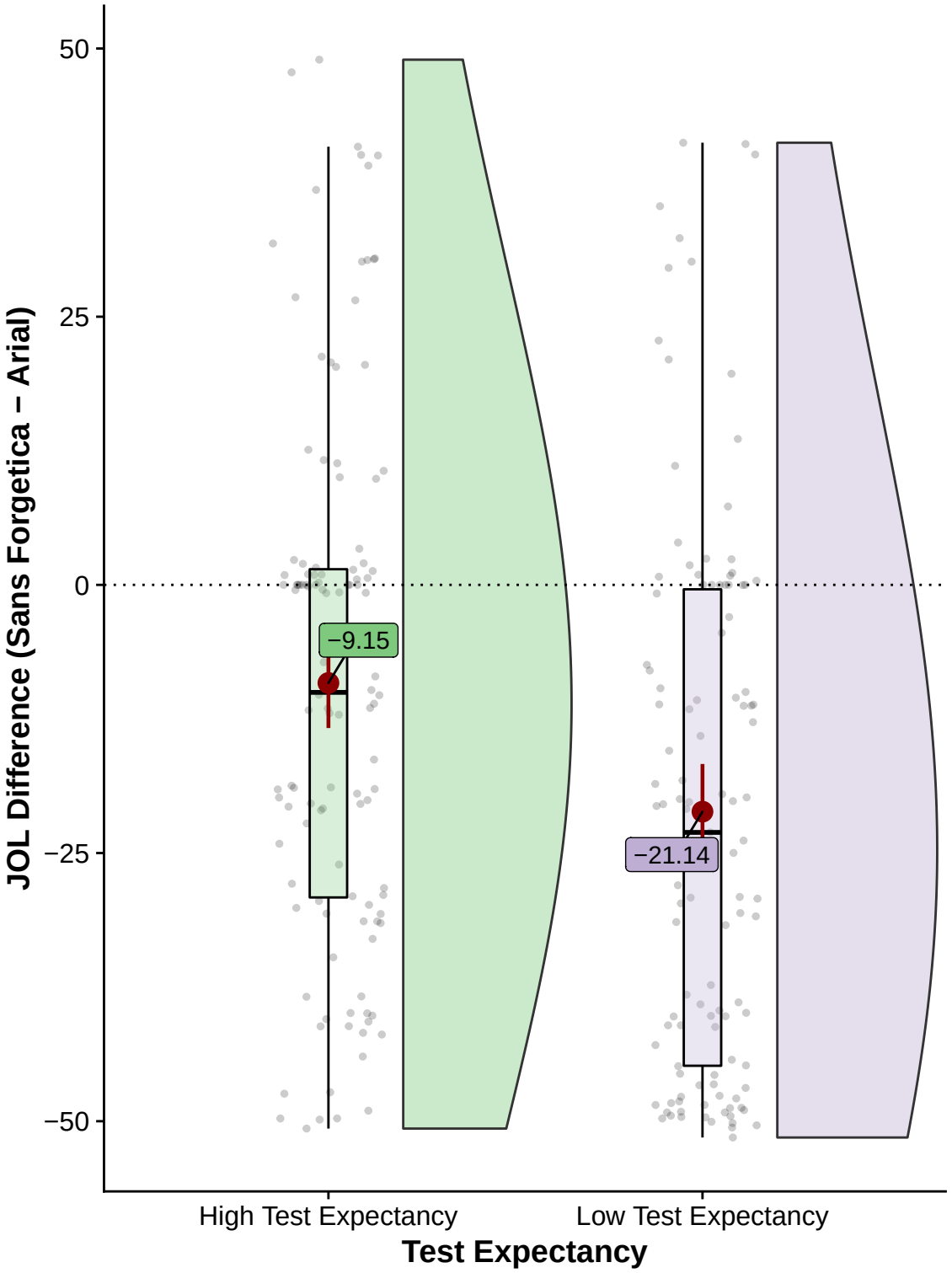
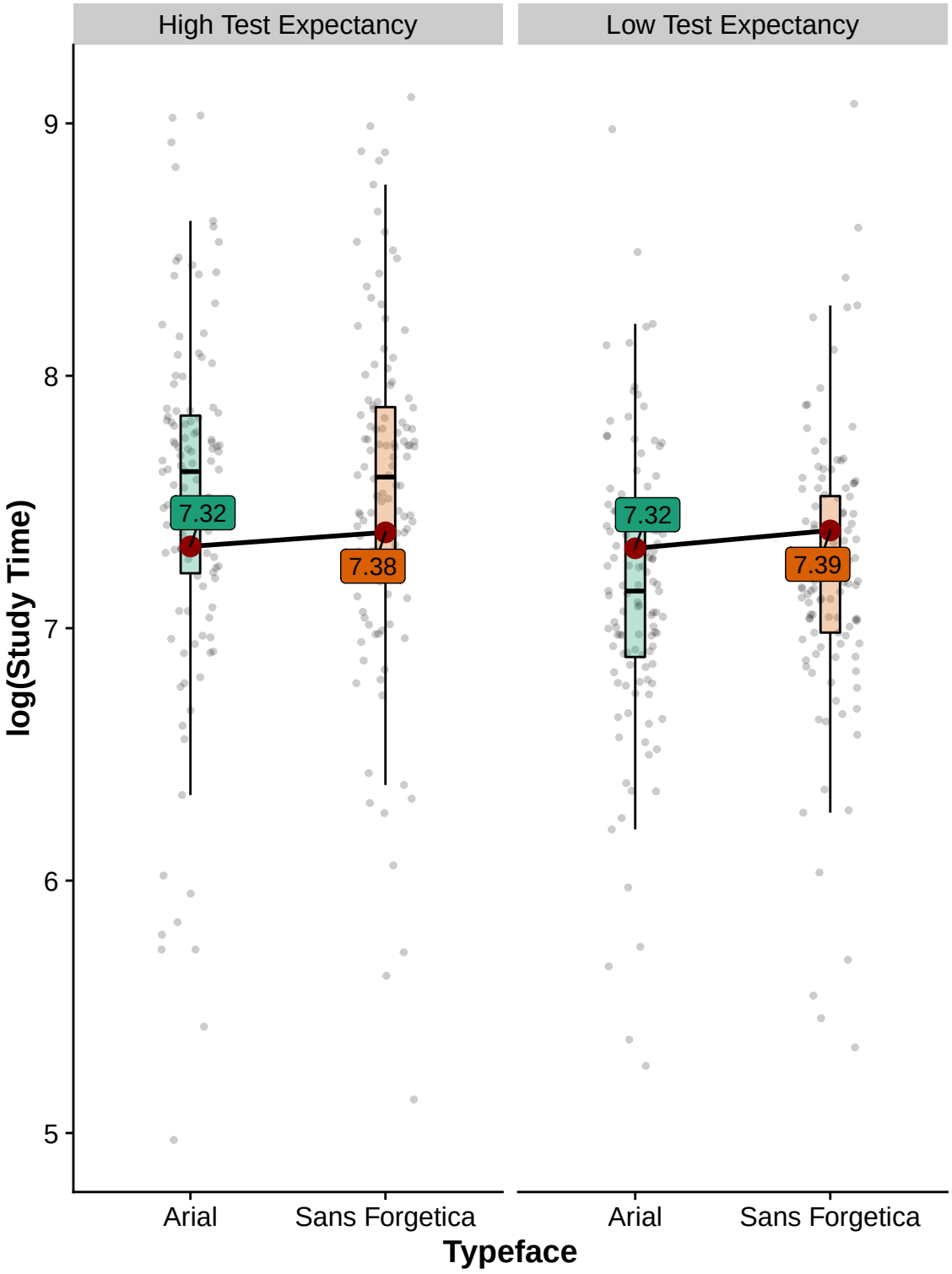
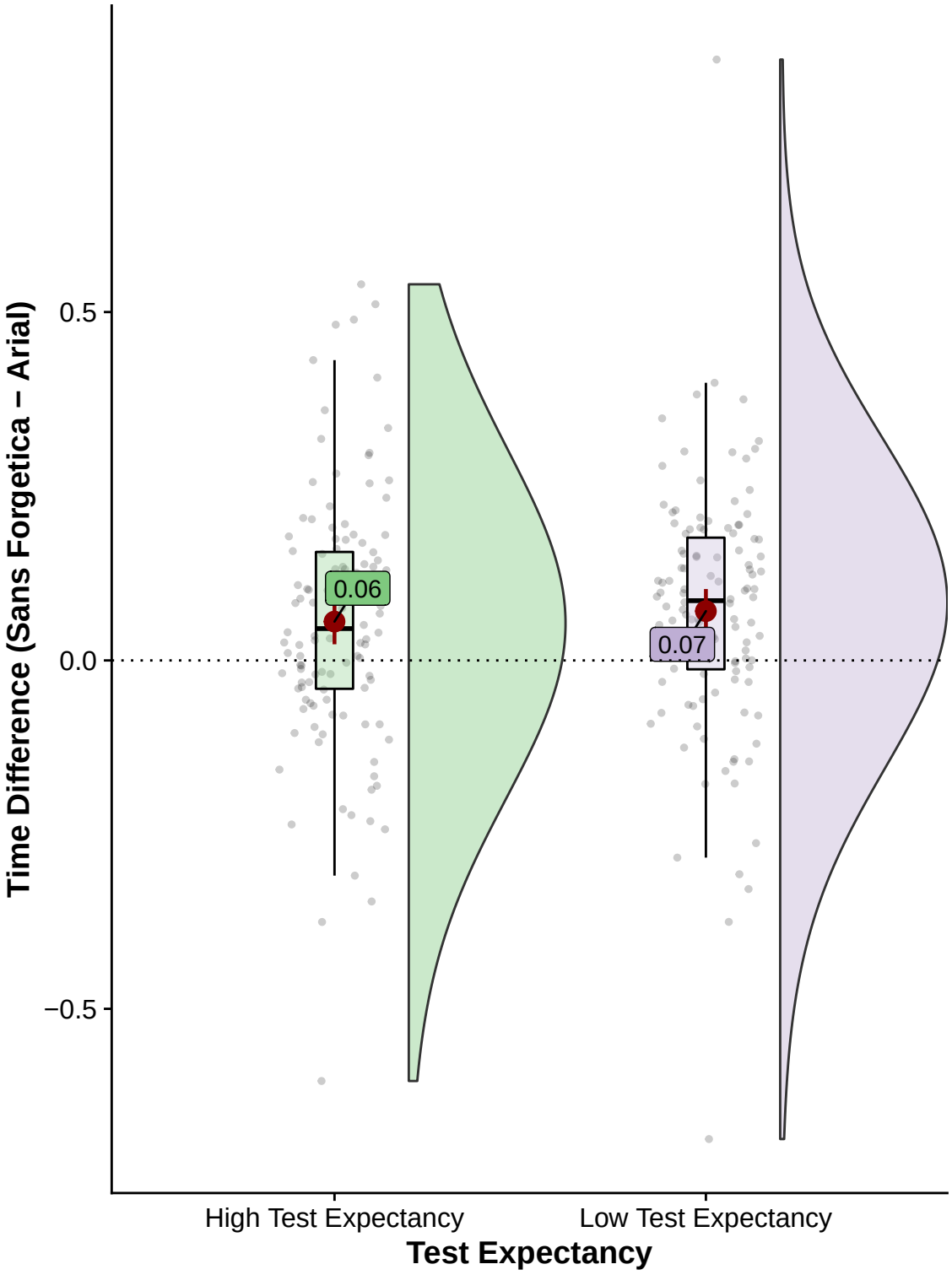


Figure 4: Raincloud plots (Allen et al., 2019) depicting raw data (dots), box plots, and half violin kernel desntiy plots, with mean (red dot) and within-participant 95 CIs. JOLs as a function of testing expectancy in Experiment 2.

```
  labs(y = "log(Study Time)", x = "Typeface") + theme(legend.position = "none") +
   geom_label_repel(data=sfgenrt_wsci, aes(y=mean_rt, label=round(mean_rt, 2)), min.segment.length = 0,
theme_cowplot(font_size=14) +
  theme(legend.position = "none", axis.title = bold)


fig2c_diff <- ggplot(rt_all_wide,aes(x=cond,y=Difference,fill=cond)) +
  geom_flat_violin(position = position_nudge(x = .2, y = 0), alpha = .4,adjust=4)+
  geom_point(position=position_jitter(width = .15),size = 1, alpha = 0.2) +
  geom_boxplot(aes(x = cond , y = Difference),outlier.shape = NA,
               alpha = 0.3, width = .1, colour = "BLACK") +
   stat_summary(fun.data="mean_cl_boot", colour="darkred", size=.8)+
   #stat_summary(fun="mean", geom="point", colour="darkred", size=3)+
  #geom_line(data=sfgenrt_wsci,aes(y=mean_rt, group=1), size=1)+
  #geom_pointrange(data=sfgenrt_wsci, aes(y=mean_rt, ymin=mean_rt-ci, ymax=mean_rt+ci),size=.3, color="
  scale_colour_brewer(palette = "Accent")+
  scale_fill_brewer(palette = "Accent") +
  labs(y = "Time Difference (Sans Forgetica - Arial)", x = "Test Expectancy") + theme(legend.position =
   geom_label_repel(data=rt_all_wide_mean, aes(y=mean, label=round(mean, 2)), seed = 42, box.padding = 0
theme_cowplot(font_size=14) +
    geom_hline(yintercept = 0, linetype="dotted") +
  theme(legend.position = "none", axis.title = bold)


fig2c
```

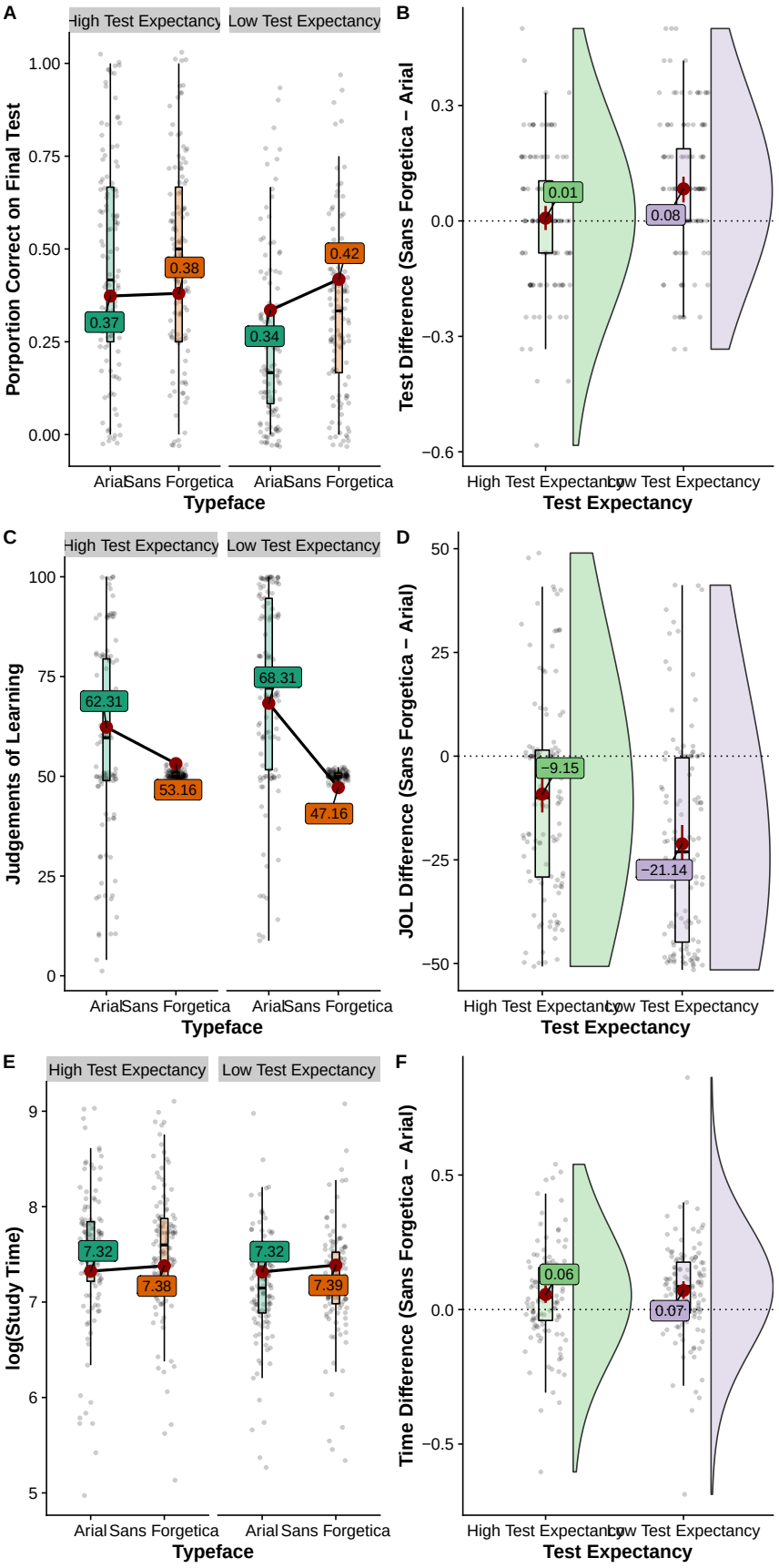

High Test Expectancy
Low Test Expectancy
log(Study Time)
9
8
7
6
5
7.32
7.38
7.32
7.39
Arial
Sans Forgetica
Arial
Sans Forgetica
Typeface

```
fig2c_diff
```



Combine Plots

```
fig2 <- plot_grid(
  fig2a,fig2b,fig2c,
  labels = "AUTO", ncol= 1, nrow = 3
)

ggsave("figexpt2.png", width=10, height=14, dpi=500)


fig2_diff <- plot_grid(
  fig2adiff,fig2b_diff,fig2c_diff,
  labels = "AUTO", ncol= 1, nrow = 3
)

ggsave("figexpt2b.png", width=10, height=14, dpi=500)


fig2 <- plot_grid(
  fig2a,fig2adiff, fig2b, fig2b_diff, fig2c, fig2c_diff,
  labels = "AUTO", ncol= 2, nrow = 3
)

ggsave("figexpt2b_all.png", width=12, height=14, dpi=500)

fig2
```


A
High Test Expectancy
Low Test Expectancy
Porportion Correct on Final Test
1.00
0.75
0.50
0.25
0.00
0.37
0.38
0.34
0.42
Arial
Sans Forgetica
Typeface
B
Test Difference (Sans Forgetica − Arial)
0.3
0.0
−0.3
−0.6
0.01
0.08
High Test Expectancy
Low Test Expectancy
Test Expectancy
C
Judgements of Learning
100
75
50
25
0
62.31
53.16
68.31
47.16
D
JOL Difference (Sans Forgetica − Arial)
50
25
0
−25
−50
−9.15
−21.14
E
log(Study Time)
9
8
7
6
5
7.32
7.38
7.32
7.39
F
Time Difference (Sans Forgetica − Arial)
0.5
0.0
−0.5
0.06
0.07