

Data analysis Pilot 2

2024-11-22

Contents

Fit algorithm definition	1
Dataset loading	3
Dataset preparation	3
Variables plot (violin & boxplot + density)	4
Statistical models	5

Fit algorithm definition

```
fit_algo <- function(new_call,
                     newData = NULL,
                     type = "lmer")
{
  if (type != "lmer" && type != "glmer")
  {
    stop("type must be lmer or glmer")
  }
  new_call["start"] <- NULL
  new_call["control"] <-
    parse(text = paste0(
      type,
      'Control(optimizer = "bobyqa", optCtrl = list(maxfun=1000000))'
    ))
  error_msg <- tryCatch({
    rval <- eval(new_call)
    if (length(rval@optinfo$conv$lme4) != 0)
    {
      "convergence"
    }
    else
    {
      "ok"
    }
  }, error = function(e) {
    "error"
  })
  count <- 0
  while (error_msg != "ok" && count < 4)
  {
    if (error_msg == "convergence")
    {
      new_call["start"] <- parse(text = "list(fixef = fixef(rval))")
    }
  }
```

```

count <- count + 1
error_msg <- tryCatch({
  rval <- eval(new_call)
  if (length(rval@optinfo$conv$lme4) != 0)
  {
    "convergence"
  }
  else
  {
    "ok"
  }
}, error = function(e) {
  "error"
})
}
new_call["start"] <- NULL
new_call["control"] <-
  parse(text = paste0(
    type,
    'Control(optimizer = "Nelder_Mead", optCtrl = list(maxfun=1000000))'
  ))
count <- 0
while (error_msg != "ok" && count < 5)
{
  count <- count + 1
  error_msg <- tryCatch({
    rval <- eval(new_call)
    if (length(rval@optinfo$conv$lme4) != 0)
    {
      "convergence"
    }
    else
    {
      "ok"
    }
  }, error = function(e) {
    "error"
  })
  if (error_msg == "convergence")
  {
    new_call["start"] <- parse(text = "list(fixef = fixef(rval))")
  }
}

}
new_call["start"] <- NULL
new_call["control"] <- NULL
count <- 0
while (error_msg != "ok" && count < 5)
{
  count <- count + 1
  error_msg <- tryCatch({
    rval <- eval(new_call)
    if (length(rval@optinfo$conv$lme4) != 0)

```

```

    {
      "convergence"
    }
    else
    {
      "ok"
    }
  }, error = function(e) {
    "error"
  })
  if (error_msg == "convergence")
  {
    new_call["start"] <- parse(text = "list(fixef = fixef(rval))")
  }
}
if (!exists("rval"))
{
  eval(new_call)
}
return(rval)
}

```

Dataset loading

```

datasets <-
  list.files(file.path("final_data"),
            pattern = "*.csv",
            full.names = TRUE)
datasets <- lapply(datasets, read_csv, show_col_types = FALSE)

d <- datasets[[1]]
for (i in seq(from = 2, to = length(datasets)))
{
  d <- rbind(d, datasets[[i]])
}

```

Dataset preparation

```

d <- d %>% mutate(Subject = rep(seq_len(length(datasets)), each = 100))

```

Then we remove total dwell times ≤ 1500 ms, scale the Agreement feature and take its square, and log-transform Dwell.Time. We do the same with Agreement.Contacts

```

d_final <- d %>% filter(Dwell.Time > 1500) %>%
  mutate(
    Dwell.Time.Log = log(Dwell.Time),
    Agreement.Scaled = (Agreement - mean(Agreement)) / sd(Agreement),
    Squared.Agreement.Scaled = Agreement.Scaled ^ 2,
    Agreement.Contacts.Scaled = (Agreement.Contacts - mean(Agreement.Contacts)) /
      sd(Agreement.Contacts),
    Squared.Agreement.Contacts.Scaled = Agreement.Contacts.Scaled ^
      2,

```

```

    Index = as.factor(Index),
    Subject = as.factor(Subject)
  )
d_final$Condition <- as.factor(d_final$Condition_Share)
levels(d_final$Condition) <- c("Reading", "Sharing")

```

Variables plot (violin & boxplot + density)

```

violin_fixation_log <-
  ggplot(d_final, aes(x = "Dwell time", y = Dwell.Time.Log)) +
  ylab("Dwell Time (log)") + theme_minimal() +
  theme(legend.position = "none") +
  geom_violin(width = 1.2) + geom_boxplot(width = 0.1, alpha = 0.2)

violin_fixation <-
  ggplot(d_final, aes(x = "Dwell time", y = Dwell.Time)) +
  ylab("Dwell Time (log)") + theme_minimal() +
  theme(legend.position = "none") +
  geom_violin(width = 1.2) + geom_boxplot(width = 0.1, alpha = 0.2)

violin_agreement <-
  ggplot(d_final, aes(x = "Agreement", y = Agreement.Scaled)) +
  ylab("Agreement (scaled)") + theme_minimal() +
  theme(legend.position = "none") +
  geom_violin(width = 1.2) + geom_boxplot(width = 0.1, alpha = 0.2)

violin_agreement_contacts <-
  ggplot(d_final,
    aes(x = "Agreement (contacts)", y = Agreement.Contacts.Scaled)) +
  ylab("Agreement (contacts; scaled)") + theme_minimal() +
  theme(legend.position = "none") +
  geom_violin(width = 1.2) + geom_boxplot(width = 0.1, alpha = 0.2)

density_fixation_log <-
  ggplot(d_final, aes(x = Dwell.Time.Log)) + geom_density()

density_fixation <-
  ggplot(d_final, aes(x = Dwell.Time)) + geom_density()

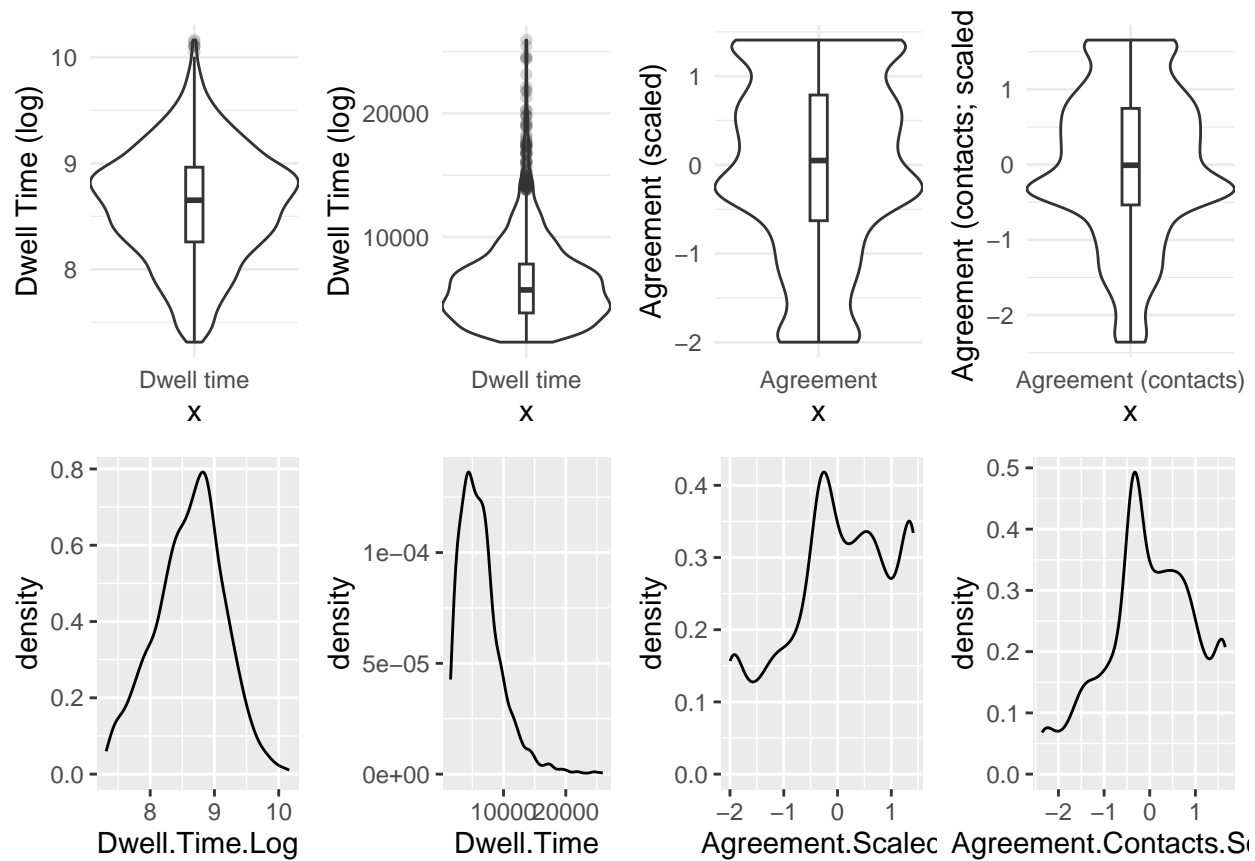
density_agreement <-
  ggplot(d_final, aes(x = Agreement.Scaled)) + geom_density()

density_agreement_contacts <-
  ggplot(d_final, aes(x = Agreement.Contacts.Scaled)) + geom_density()

grid.arrange(
  violin_fixation_log,
  violin_fixation,
  violin_agreement,
  violin_agreement_contacts,
  density_fixation_log,
  density_fixation,
  density_agreement,

```

```
density_agreement_contacts,
nrow = 2
)
```



Statistical models

Agreement

```
personal_log_call <-
  parse(
    text = 'lmer(Dwell.Time.Log ~ Agreement.Scaled + Squared.Agreement.Scaled + Condition_Share + Agreement.Scaled + Squared.Agreement.Scaled | Subject),
            (1 | Index) +
            (Agreement.Scaled + Squared.Agreement.Scaled | Subject),
            data=d_final)'
  )[[1]]
```

```
fit_personal_log <- fit_algo(personal_log_call)
```

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

```

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

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: Dwell.Time.Log ~ Agreement.Scaled + Squared.Agreement.Scaled +
##      Condition_Share + Agreement.Scaled * Condition_Share + Squared.Agreement.Scaled *
##      Condition_Share + (1 | Index) + (Agreement.Scaled + Squared.Agreement.Scaled |
##      Subject)
##      Data: d_final
##
## REML criterion at convergence: 2632
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -3.6092 -0.6398 -0.0055  0.5816  4.0266
##
## Random effects:
##      Groups      Name                Variance Std.Dev. Corr
##      Index      (Intercept)          2.270e-02 0.150654
##      Subject     (Intercept)          1.118e-01 0.334403
##               Agreement.Scaled        8.792e-05 0.009377  0.58
##               Squared.Agreement.Scaled 5.818e-04 0.024120 -0.50 -1.00
##      Residual                1.615e-01 0.401920
## Number of obs: 2315, groups:  Index, 100; Subject, 26
##
## Fixed effects:
##
##              Estimate Std. Error      df
## (Intercept)      8.701902   0.095539  25.021471
## Agreement.Scaled -0.012183   0.013592 149.340784
## Squared.Agreement.Scaled -0.005727   0.014419  24.948833
## Condition_Share -0.280702   0.133497  23.856726
## Agreement.Scaled:Condition_Share  0.045423   0.019247 162.805445
## Squared.Agreement.Scaled:Condition_Share  0.020520   0.020802  27.114539
##
##              t value Pr(>|t|)
## (Intercept)      91.082   <2e-16 ***
## Agreement.Scaled  -0.896   0.3715
## Squared.Agreement.Scaled -0.397   0.6946
## Condition_Share  -2.103   0.0462 *
## Agreement.Scaled:Condition_Share  2.360   0.0195 *
## Squared.Agreement.Scaled:Condition_Share  0.986   0.3326
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##              (Intr) Agrm.S Sq.A.S Cndt_S A.S:C_
## Agrmnt.Scld  0.058
## Sqrd.Agrm.S -0.343  0.168
## Conditn_Shr -0.697 -0.042  0.241
## Agrmn.S:C_S -0.044 -0.618 -0.105  0.060
## Sqr.A.S:C_S  0.234 -0.114 -0.668 -0.339  0.234
## optimizer (nloptwrap) convergence code: 0 (OK)
## boundary (singular) fit: see help('isSingular')

```

```

personal_call <-
  parse(text = 'lmer(Dwell.Time ~ Agreement.Scaled + Squared.Agreement.Scaled + Condition_Share + Agreement.Scaled * Condition_Share + (1 | Index) + (1 | Subject),
    data=d_final)')[[1]]

fit_personal <- fit_algo(personal_call)

summary(fit_personal)

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula:
## Dwell.Time ~ Agreement.Scaled + Squared.Agreement.Scaled + Condition_Share +
##   Agreement.Scaled * Condition_Share + Squared.Agreement.Scaled *
##   Condition_Share + (1 | Index) + (1 | Subject)
## Data: d_final
## Control: lmerControl(optimizer = "bobyqa", optCtrl = list(maxfun = 1e+06))
##
## REML criterion at convergence: 43324.4
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.6058 -0.6081 -0.1498  0.4009  6.0044
##
## Random effects:
##   Groups   Name                Variance Std.Dev.
##   Index    (Intercept) 1148897   1072
##   Subject  (Intercept) 3379417   1838
##   Residual                    7298004  2701
## Number of obs: 2315, groups:  Index, 100; Subject, 26
##
## Fixed effects:
##              Estimate Std. Error      df t value
## (Intercept)    6717.48    533.63    27.18  12.588
## Agreement.Scaled    -20.07     89.10  2289.05  -0.225
## Squared.Agreement.Scaled    -41.39     83.83  2274.14  -0.494
## Condition_Share   -1315.29    739.96    25.15  -1.778
## Agreement.Scaled:Condition_Share     294.71    126.14  2225.07   2.336
## Squared.Agreement.Scaled:Condition_Share  123.03    121.71  2234.12   1.011
##              Pr(>|t|)
## (Intercept)    7.46e-13 ***
## Agreement.Scaled    0.8218
## Squared.Agreement.Scaled    0.6216
## Condition_Share    0.0876 .
## Agreement.Scaled:Condition_Share    0.0196 *
## Squared.Agreement.Scaled:Condition_Share  0.3122
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##              (Intr) Agrm.S Sq.A.S Cndt_S A.S:C_
## Agrmnt.Scld -0.060
## Sqrd.Agrm.S -0.161  0.320

```

```

## Conditn_Shr -0.691  0.043  0.111
## Agrmn.S:C_S  0.039 -0.612 -0.207 -0.061
## Sqr.A.S:C_S  0.106 -0.217 -0.656 -0.158  0.392

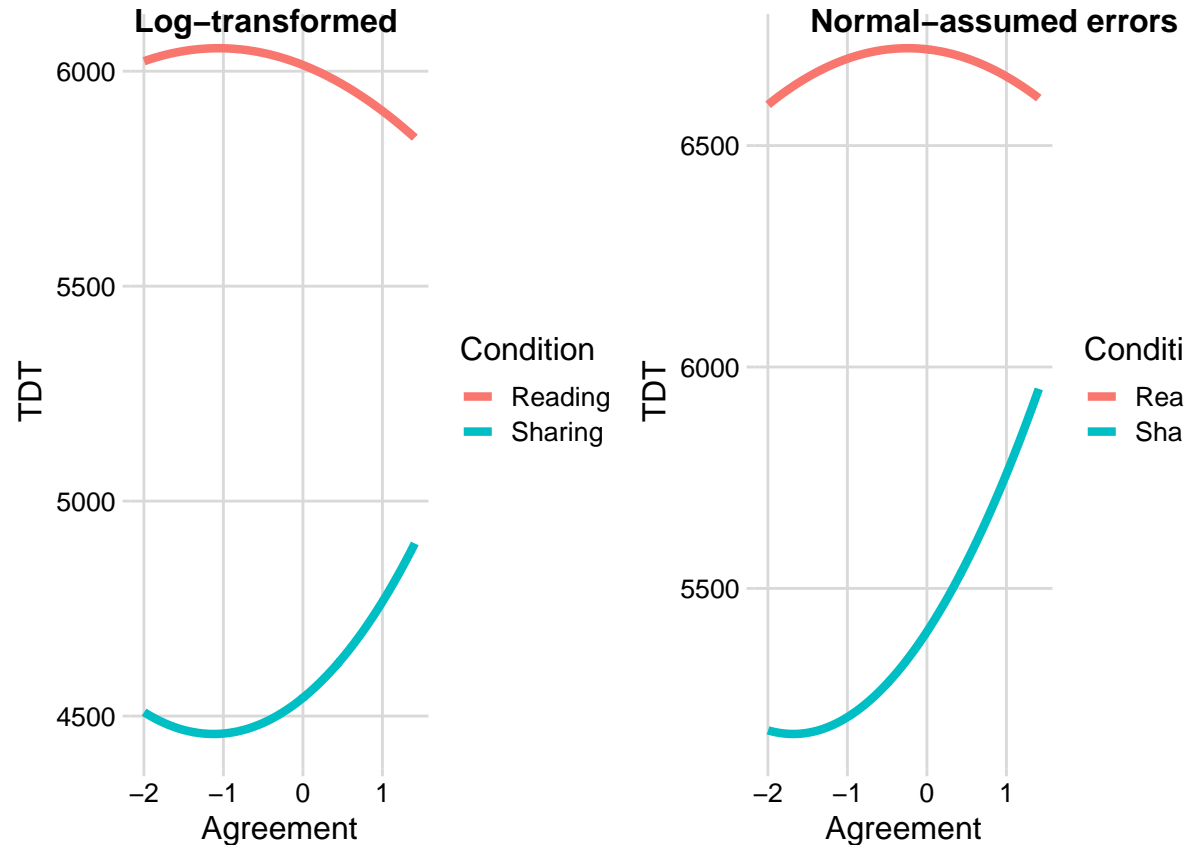
d_final$personal_log_prediction <-
  exp(predict(fit_personal_log, re.form = NA))
d_final$personal_prediction <- predict(fit_personal, re.form = NA)

p1 <- ggplot(d_final) +
  geom_line(
    aes(
      x = Agreement.Scaled,
      personal_log_prediction,
      color = Condition,
      group = Condition
    ),
    linewidth = 1.5
  ) +
  theme_minimal_grid(12) + labs(y = "TDT", x = "Agreement")

p2 <- ggplot(d_final) +
  geom_line(
    aes(
      x = Agreement.Scaled,
      personal_prediction,
      color = Condition,
      group = Condition
    ),
    linewidth = 1.5
  ) +
  theme_minimal_grid(12) + labs(y = "TDT", x = "Agreement")

plot_grid(
  p1,
  p2,
  labels = c("Log-transformed", "Normal-assumed errors"),
  label_size = 12
)

```

Fixed effects plots

```
ggsave(p2, filename = "study2_plot1.png", type = "cairo")
```

```
## Saving 6.5 x 4.5 in image
```

```
## Warning: Using ragg device as default. Ignoring `type` and `antialias`
```

```
## arguments
```

Agreement (contacts)

```
contacts_log_call <-  
  parse(  
    text = 'lmer(Dwell.Time.Log ~ Agreement.Contacts.Scaled + Squared.Agreement.Contacts.Scaled + Condi  
              (1 | Index) +  
              (Agreement.Contacts.Scaled + Squared.Agreement.Contacts.Scaled | Subject),  
              data=d_final)'  
  )[[1]]
```

```
fit_contacts_log <- fit_algo(contacts_log_call)
```

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

```

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

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula:
## Dwell.Time.Log ~ Agreement.Contacts.Scaled + Squared.Agreement.Contacts.Scaled +
##   Condition_Share + Agreement.Contacts.Scaled * Condition_Share +
##   Squared.Agreement.Contacts.Scaled * Condition_Share + (1 |
##   Index) + (Agreement.Contacts.Scaled + Squared.Agreement.Contacts.Scaled |
##   Subject)
## Data: d_final
##
## REML criterion at convergence: 2636.9
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -3.6476 -0.6340  0.0012  0.5744  3.9408
##
## Random effects:
##   Groups   Name                                Variance Std.Dev. Corr
##   Index    (Intercept)                        0.0228457 0.151148
##   Subject   (Intercept)                        0.1079789 0.328601
##            Agreement.Contacts.Scaled           0.0001021 0.010104 -1.00
##            Squared.Agreement.Contacts.Scaled 0.0000495 0.007035 -1.00  1.00
## Residual                                     0.1623486 0.402925
## Number of obs: 2315, groups:  Index, 100; Subject, 26
##
## Fixed effects:
##                                     Estimate Std. Error
## (Intercept)                        8.706192   0.093677
## Agreement.Contacts.Scaled           0.006698   0.013365
## Squared.Agreement.Contacts.Scaled  -0.012983   0.010674
## Condition_Share                    -0.287188   0.130958
## Agreement.Contacts.Scaled:Condition_Share  0.016318   0.018881
## Squared.Agreement.Contacts.Scaled:Condition_Share 0.028508   0.015926
##                                     df t value Pr(>|t|)
## (Intercept)                        24.853105  92.939 <2e-16
## Agreement.Contacts.Scaled          267.499796   0.501  0.6167
## Squared.Agreement.Contacts.Scaled  334.703612 -1.216  0.2247
## Condition_Share                    23.743559 -2.193  0.0383
## Agreement.Contacts.Scaled:Condition_Share 270.638917  0.864  0.3882
## Squared.Agreement.Contacts.Scaled:Condition_Share 413.064541  1.790  0.0742
##
## (Intercept)                        ***
## Agreement.Contacts.Scaled
## Squared.Agreement.Contacts.Scaled
## Condition_Share                      *
```

```

## Agreement.Contacts.Scaled:Condition_Share
## Squared.Agreement.Contacts.Scaled:Condition_Share .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##      (Intr) Ag.C.S Sq.A.C.S Cndt_S A.C.S:
## Agrmnt.Cn.S -0.230
## Sqrd.Ag.C.S -0.287  0.257
## Conditn_Shr -0.696  0.164  0.203
## Agr.C.S:C_S  0.162 -0.647 -0.173  -0.242
## S.A.C.S:C_S  0.191 -0.173 -0.652  -0.285  0.341
## optimizer (nloptwrap) convergence code: 0 (OK)
## boundary (singular) fit: see help('isSingular')

contacts_call <-
  parse(text = 'lmer(Dwell.Time ~ Agreement.Contacts.Scaled + Squared.Agreement.Contacts.Scaled + Condi
              (1 | Index) +
              (1 | Subject),
              data=d_final)')[[1]]

fit_contacts <- fit_algo(contacts_call)

summary(fit_contacts)

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula:
## Dwell.Time ~ Agreement.Contacts.Scaled + Squared.Agreement.Contacts.Scaled +
##   Condition_Share + Agreement.Contacts.Scaled * Condition_Share +
##   Squared.Agreement.Contacts.Scaled * Condition_Share + (1 |
##   Index) + (1 | Subject)
## Data: d_final
## Control: lmerControl(optimizer = "bobyqa", optCtrl = list(maxfun = 1e+06))
##
## REML criterion at convergence: 43329.4
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.6140 -0.6151 -0.1511  0.3950  6.1987
##
## Random effects:
##   Groups   Name                Variance Std.Dev.
##   Index    (Intercept) 1148847   1072
##   Subject  (Intercept) 3344721   1829
##   Residual                    7313227  2704
## Number of obs: 2315, groups:  Index, 100; Subject, 26
##
## Fixed effects:
##
##              Estimate Std. Error
## (Intercept)    6752.037    528.759
## Agreement.Contacts.Scaled      72.086     88.260
## Squared.Agreement.Contacts.Scaled    -79.218     70.935
## Condition_Share   -1364.082    733.992
## Agreement.Contacts.Scaled:Condition_Share     -3.012    124.555

```

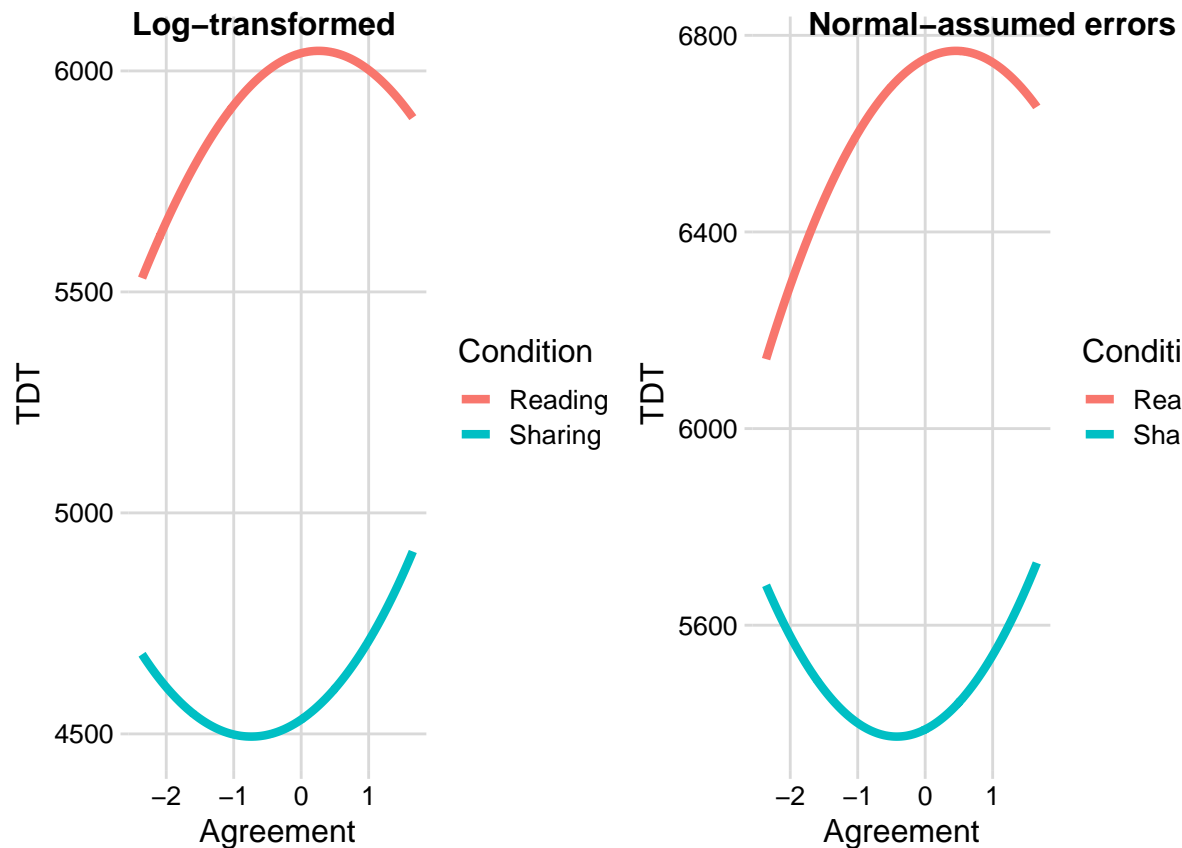
```
## Squared.Agreement.Contacts.Scaled:Condition_Share    161.221    105.953
##                                                    df t value Pr(>|t|)
## (Intercept)                                           26.704   12.770 6.98e-13
## Agreement.Contacts.Scaled                           2296.505    0.817    0.414
## Squared.Agreement.Contacts.Scaled                   2261.690   -1.117    0.264
## Condition_Share                                       24.810   -1.858    0.075
## Agreement.Contacts.Scaled:Condition_Share           2235.007   -0.024    0.981
## Squared.Agreement.Contacts.Scaled:Condition_Share    2235.681    1.522    0.128
##
## (Intercept) ***
## Agreement.Contacts.Scaled
## Squared.Agreement.Contacts.Scaled
## Condition_Share .
## Agreement.Contacts.Scaled:Condition_Share
## Squared.Agreement.Contacts.Scaled:Condition_Share
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##          (Intr) Ag.C.S Sq.A.C.S Cndt_S A.C.S:
## Agrmnt.Cn.S -0.032
## Sqrd.Ag.C.S -0.132  0.235
## Conditn_Shr -0.690  0.022  0.092
## Agr.C.S:C_S  0.021 -0.643 -0.157  -0.043
## S.A.C.S:C_S  0.086 -0.158 -0.650  -0.138  0.324
```

```
d_final$contacts_log_prediction <-
  exp(predict(fit_contacts_log, re.form = NA))
d_final$contacts_prediction <- predict(fit_contacts, re.form = NA)

p1 <- ggplot(d_final) +
  geom_line(
    aes(
      x = Agreement.Contacts.Scaled,
      contacts_log_prediction,
      color = Condition,
      group = Condition
    ),
    linewidth = 1.5
  ) +
  theme_minimal_grid(12) + labs(y = "TDT", x = "Agreement")

p2 <- ggplot(d_final) +
  geom_line(
    aes(
      x = Agreement.Contacts.Scaled,
      contacts_prediction,
      color = Condition,
      group = Condition
    ),
    linewidth = 1.5
  ) +
  theme_minimal_grid(12) + labs(y = "TDT", x = "Agreement")
```

```
plot_grid(
  p1,
  p2,
  labels = c("Log-transformed", "Normal-assumed errors"),
  label_size = 12
)
```



Fixed effects plots

```
ggsave(p2, filename = "study2_plot2.png", type = "cairo")
```

```
## Saving 6.5 x 4.5 in image
```

```
## Warning: Using ragg device as default. Ignoring `type` and `antialias`
```

```
## arguments
```

```
vertex_reading_personal <-
  (-summary(fit_personal)[["coefficients"]][2]) / (2 * summary(fit_personal)[["coefficients"]][3])
vertex_sharing_personal <-
  -(summary(fit_personal)[["coefficients"]][2] + summary(fit_personal)[["coefficients"]][5]) /
  (2 * (summary(fit_personal)[["coefficients"]][3] + summary(fit_personal)[["coefficients"]][6]))

pred_reading_personal <-
  predict(
    fit_personal,
    re.form = NA,
    newdata = tibble(
      Condition_Share = c(0, 0, 0),
      Agreement.Scaled = c(
```

```

        min(d_final$Agreement.Scaled),
        vertex_reading_personal,
        max(d_final$Agreement.Scaled)
    ),
    Squared.Agreement.Scaled = Agreement.Scaled ** 2
)
)

pred_sharing_personal <-
predict(
  fit_personal,
  re.form = NA,
  newdata = tibble(
    Condition_Share = c(1, 1, 1),
    Agreement.Scaled = c(
      min(d_final$Agreement.Scaled),
      vertex_sharing_personal,
      max(d_final$Agreement.Scaled)
    ),
    Squared.Agreement.Scaled = Agreement.Scaled ** 2
  )
)

vertex_reading_contacts <-
(-summary(fit_contacts)[["coefficients"]][2]) / (2 * summary(fit_contacts)[["coefficients"]][3])
vertex_sharing_contacts <-
-(summary(fit_contacts)[["coefficients"]][2] + summary(fit_contacts)[["coefficients"]][5]) /
(2 * (summary(fit_contacts)[["coefficients"]][3] + summary(fit_contacts)[["coefficients"]][6]))

pred_reading_contacts <-
predict(
  fit_contacts,
  re.form = NA,
  newdata = tibble(
    Condition_Share = c(0, 0, 0),
    Agreement.Contacts.Scaled = c(
      min(d_final$Agreement.Contacts.Scaled),
      vertex_reading_contacts,
      max(d_final$Agreement.Contacts.Scaled)
    ),
    Squared.Agreement.Contacts.Scaled = Agreement.Contacts.Scaled ** 2
  )
)

pred_sharing_contacts <-
predict(
  fit_contacts,
  re.form = NA,
  newdata = tibble(
    Condition_Share = c(1, 1, 1),
    Agreement.Contacts.Scaled = c(
      min(d_final$Agreement.Contacts.Scaled),
      vertex_sharing_contacts,

```

```

        max(d_final$Agreement.Contacts.Scaled)
    ),
    Squared.Agreement.Contacts.Scaled = Agreement.Contacts.Scaled ** 2
)
)

table <-
  round(matrix(
    c(
      vertex_reading_personal,
      vertex_sharing_personal,
      vertex_reading_contacts,
      vertex_sharing_contacts,
      min(d_final$Agreement.Scaled),
      min(d_final$Agreement.Scaled),
      min(d_final$Agreement.Contacts.Scaled),
      min(d_final$Agreement.Contacts.Scaled),
      max(d_final$Agreement.Scaled),
      max(d_final$Agreement.Scaled),
      max(d_final$Agreement.Contacts.Scaled),
      max(d_final$Agreement.Contacts.Scaled),
      pred_reading_personal[2] - pred_reading_personal[1],
      pred_sharing_personal[2] - pred_sharing_personal[1],
      pred_reading_contacts[2] - pred_reading_contacts[1],
      pred_sharing_contacts[2] - pred_sharing_contacts[1],
      pred_reading_personal[2] - pred_reading_personal[3],
      pred_sharing_personal[2] - pred_sharing_personal[3],
      pred_reading_contacts[2] - pred_reading_contacts[3],
      pred_sharing_contacts[2] - pred_sharing_contacts[3]
    ),
    ncol = 4,
    byrow = TRUE
  ), 3)

rownames(table) <-
  c(
    'Vertex position',
    'min agreement',
    'max agreement',
    'diff min vs. vertex',
    'diff max vs. vertex'
  )
colnames(table) <-
  c(
    "Personal agreement, reading",
    "Personal agreement, sharing",
    "Contacts agreement, reading",
    "Contacts agreement, sharing"
  )
table <- as.table(table)
table

```

```

##               Personal agreement, reading Personal agreement, sharing
## Vertex position                -0.242                -1.682

```

## min agreement	-1.998	-1.998
## max agreement	1.408	1.408
## diff min vs. vertex	127.567	-8.158
## diff max vs. vertex	112.704	-779.358
##	Contacts agreement, reading	Contacts agreement, sharing
## Vertex position	0.455	-0.421
## min agreement	-2.359	-2.359
## max agreement	1.655	1.655
## diff min vs. vertex	627.130	-307.830
## diff max vs. vertex	114.001	-353.337