LDT Size Bias Analyses

Dani Larranaga

2023-06-20

Study 1: Direct Replication

Participant Level:

RT

```
t.PL.RT = t.test(LDT.participantLevel$RT_Small, LDT.participantLevel$RT_Large, paired = T)
t.PL.RT
```

```
##
##  Paired t-test
##
## data:  LDT.participantLevel$RT_Small and LDT.participantLevel$RT_Large
## t = 0.95292, df = 107, p-value = 0.3428
## alternative hypothesis: true mean difference is not equal to 0
## 95 percent confidence interval:
##  -1.791438  5.107937
## sample estimates:
## mean difference
##         1.65825
```

```
d = unname(t2d(t.PL.RT$statistic, n=108))
d
```

```
## [1] 0.1833901
```

Error Rate

```
t.PL.ER = t.test(LDT.participantLevel$Small.Accuracy, LDT.participantLevel$Large.Accuracy, paired = T)
t.PL.ER
```

```
##
##  Paired t-test
##
## data:  LDT.participantLevel$Small.Accuracy and LDT.participantLevel$Large.Accuracy
## t = -2.4269, df = 107, p-value = 0.0169
```

```
## alternative hypothesis: true mean difference is not equal to 0
## 95 percent confidence interval:
##  -0.013458024 -0.001356791
## sample estimates:
## mean difference
##    -0.007407407
```

```
d = unname(t2d(t.PL.ER$statistic, n=108))
d
```

```
## [1] -0.4670597
```

Inverse Efficiency Score (IES)

```
t.PL.IES = t.test(LDT.participantLevel$Small_IES, LDT.participantLevel$Large_IES, paired = T)
t.PL.IES
```

```
##
##  Paired t-test
##
## data:  LDT.participantLevel$Small_IES and LDT.participantLevel$Large_IES
## t = 2.3446, df = 107, p-value = 0.02089
## alternative hypothesis: true mean difference is not equal to 0
## 95 percent confidence interval:
##   0.8979614 10.7269230
## sample estimates:
## mean difference
##        5.812442
```

```
d = unname(t2d(t.PL.IES$statistic, n=108))
d
```

```
## [1] 0.4512184
```

Word-Pair Level:

RT

```
t.WP.RT = t.test(LDT.wordPairLevel$RT_Small, LDT.wordPairLevel$RT_Large, paired = T)
t.WP.RT
```

```
##
##  Paired t-test
##
## data:  LDT.wordPairLevel$RT_Small and LDT.wordPairLevel$RT_Large
## t = 0.45592, df = 44, p-value = 0.6507
## alternative hypothesis: true mean difference is not equal to 0
## 95 percent confidence interval:
```

```
##  -9.859799 15.625059
## sample estimates:
## mean difference
##         2.88263
```

```r
d = unname(t2d(t.WP.RT$statistic, n=108))
d
```

```
## [1] 0.08774231
```

Error Rate

```r
t.WP.ER = t.test(LDT.wordPairLevel$WP.Acc.Small, LDT.wordPairLevel$WP.Acc.Large, paired = T)
t.WP.ER
```

```
##
##  Paired t-test
##
## data:  LDT.wordPairLevel$WP.Acc.Small and LDT.wordPairLevel$WP.Acc.Large
## t = -1.0577, df = 44, p-value = 0.296
## alternative hypothesis: true mean difference is not equal to 0
## 95 percent confidence interval:
##  -0.020594353  0.006418108
## sample estimates:
## mean difference
##    -0.007088123
```

```r
d = unname(t2d(t.WP.ER$statistic, n=108))
d
```

```
## [1] -0.2035492
```

IES

```r
t.WP.IES = t.test(LDT.wordPairLevel$Small_IES, LDT.wordPairLevel$Large_IES, paired = T)
t.WP.IES
```

```
##
##  Paired t-test
##
## data:  LDT.wordPairLevel$Small_IES and LDT.wordPairLevel$Large_IES
## t = 0.76082, df = 44, p-value = 0.4508
## alternative hypothesis: true mean difference is not equal to 0
## 95 percent confidence interval:
##  -12.89162  28.52790
## sample estimates:
## mean difference
##        7.818141
```

```r
d = unname(t2d(t.WP.IES$statistic, n=108))
d
```

```
## [1] 0.1464201
```

Study 2: Norming Differences

```r
df.Norm.WP = pivot_wider(WR.scaledDimsOfInterest[,2:10],
                         id_cols = c("WordPair"),
                         names_from = c("size"),
                         values_from = c(3:9))
```

Intraclass Correlations

USING THE SCALED VARIABLES!!! (because GN has them on different scales)

```r
m = "two"
t = "agreement"
s = "single"


size_icc = icc(cbind(WR.scaledDimsOfInterest$Size,
                     WR.scaledDimsOfInterest$SIZE.Glasgow_Norms),
               model = m, type = t, unit = s)
val_icc  = icc(cbind(WR.scaledDimsOfInterest$Valence,
                    WR.scaledDimsOfInterest$VAL.Glasgow_Norms),
               model = m, type = t, unit = s)
gend_icc = icc(cbind(WR.scaledDimsOfInterest$Gender,
                     WR.scaledDimsOfInterest$GEND.Glasgow_Norms),
               model = m, type = t, unit = s)
img_icc  = icc(cbind(WR.scaledDimsOfInterest$Imagability,
                     WR.scaledDimsOfInterest$IMAG.Glasgow_Norms),
               model = m, type = t, unit = s)
conc_icc = icc(cbind(WR.scaledDimsOfInterest$Concreteness,
                     WR.scaledDimsOfInterest$CNC.Glasgow_Norms),
               model = m, type = t, unit = s)
arou_icc = icc(cbind(WR.scaledDimsOfInterest$Arousal,
                     WR.scaledDimsOfInterest$AROU.Glasgow_Norms),
               model = m, type = t, unit = s)
fam_icc  = icc(cbind(WR.scaledDimsOfInterest$Familiarity,
                     WR.scaledDimsOfInterest$FAM.Glasgow_Norms),
               model = m, type = t, unit = s)

ests = c(size_icc$value, val_icc$value, gend_icc$value, img_icc$value,
         conc_icc$value, arou_icc$value, fam_icc$value)
lbs = c(size_icc$lbound, val_icc$lbound, gend_icc$lbound, img_icc$lbound,
        conc_icc$lbound, arou_icc$lbound, fam_icc$lbound)
ubs = c(size_icc$ubound, val_icc$ubound, gend_icc$ubound, img_icc$ubound,
        conc_icc$ubound, arou_icc$ubound, fam_icc$ubound)
```

```r
sigs = c(size_icc$p.value, val_icc$p.value, gend_icc$p.value, img_icc$p.value,
         conc_icc$p.value, arou_icc$p.value, fam_icc$p.value)

forkable = data.frame(ests, lbs, ubs, sigs)
rownames(forkable) = c("size", "valence", "gender", "imageability",
                       "concreteness", "aorusal", "familiarity")


print(knitr::kable(forkable[order(forkable$ests, decreasing = T),], bookend = T, digits = 3))
```

	ests	lbs	ubs	sigs
size	0.967	0.950	0.978	0
gender	0.946	0.919	0.965	0
valence	0.934	0.901	0.956	0
concreteness	0.822	0.740	0.880	0
imageability	0.720	0.600	0.808	0
familiarity	0.628	0.481	0.740	0
aorusal	0.609	0.458	0.726	0

*t*s between US and UK

```r
forkable = data.frame(Dimension = c(""),
                      t.value = c(0),
                      p.value = c(0))
forkable = forkable[-1,]

for(d in 1:7){
  d.PU = d*2+2
  d.GN = d*2+3

  v = names(WR.scaledDimsOfInterest)[d.PU]

  tmp.t = t.test(WR.scaledDimsOfInterest[,d.PU], WR.scaledDimsOfInterest[,d.GN], paired = T)

  forkable[d,] = cbind(v, round(tmp.t$statistic,3), round(tmp.t$p.value,3))
}

print(knitr::kable(forkable[order(forkable$t.value),], bookend = T, row.names = F, digits = 3))
```

Dimension	t.value	p.value
Arousal	-0.006	0.995
Valence	-0.038	0.97
Familiarity	-0.097	0.923
Size	-0.223	0.824
Imagability	0.128	0.899
Concreteness	0.252	0.802
Gender	0.43	0.668

*t*s between word groups

```
forkable = data.frame(Dimension = c(""),
                      `S_Mean_PU` = c(0),
                      `L_Mean_PU` = c(0),
                      `S_95CI_PU` = c(0),
                      `L_95CI_PU` = c(0),
                      t.value.PU = c(0),
                      p.value.PU = c(0),
                      `S Mean GN` = c(0),
                      `L Mean GN` = c(0),
                      t.value.GN = c(0),
                      p.value.GN = c(0))
forkable = forkable[-1,]

for(d in 1:8){
  d.s.PU = d*4-2
  d.l.PU = d*4-1
  d.s.GN = d*4
  d.l.GN = d*4+1

  v = gsub("_Small", "", names(WR.wordPairs)[d.s.PU])

  sm.PU      = mean(WR.wordPairs[,d.s.PU], na.rm = T)
  lm.PU      = mean(WR.wordPairs[,d.l.PU], na.rm = T)
  sm.PU.95CI = 1.96*(sd(WR.wordPairs[,d.s.PU], na.rm = T) / sqrt(length(WR.wordPairs[,d.s.PU])))
  lm.PU.95CI = 1.96*(sd(WR.wordPairs[,d.l.PU], na.rm = T) / sqrt(length(WR.wordPairs[,d.l.PU])))

  sm.GN = mean(WR.wordPairs[,d.s.GN], na.rm = T)
  lm.GN = mean(WR.wordPairs[,d.l.GN], na.rm = T)

  tmp.t.PU = t.test(WR.wordPairs[, d.l.PU], WR.wordPairs[, d.s.PU], paired = T)
  tmp.t.GN = t.test(WR.wordPairs[, d.l.GN], WR.wordPairs[, d.s.GN], paired = T)

  forkable[d,] = cbind(v, round(sm.PU,3), round(lm.PU,3), round(sm.PU.95CI,4), round(lm.PU.95CI,4), rou
                          round(sm.GN,3), round(lm.GN,3), round(tmp.t.GN$statistic,3), round(tmp.t.GN$p
}

print(knitr::kable(forkable, bookend = T, row.names = F, digits = 3))
```

Dimension	S_Mean_PU	L_Mean_PU	S_95CI_PU	L_95CI_PU	t.value.PU	p.value.PU	S.Mean.GN	L.Mean.GN	t.value.GN	p.value.GN
Size	2.431	5.947	0.1536	0.1395	29.214	0	2.409	5.412	23.371	0
Arousal	3.113	3.315	0.2148	0.1621	1.431	0.16	4.414	4.858	2.557	0.014
Valence	4.445	4.771	0.3392	0.2301	1.487	0.144	5.305	5.626	1.233	0.225
Familiarity	5.863	5.505	0.1395	0.17	-3.601	0.001	5.751	5.473	-2.425	0.02
Imagability	5.961	5.941	0.1915	0.1674	-0.164	0.87	6.361	6.535	1.541	0.131
Concreteness	5.508	5.392	0.18	0.2312	-0.758	0.453	6.403	6.463	0.337	0.738
Gender	2.745	3.495	0.2274	0.2081	4.989	0	3.641	4.601	5.244	0
Dominance	3.724	4.894	0.1888	0.2144	8.553	0	5.112	5.114	0.007	0.994

```
forGraph = pivot_longer(forkable[,1:5],2:5, names_to = c("size","scale", "locus"), names_sep = "_")
forGraph = data.frame(pivot_wider(forGraph,id_cols = c("size"), names_from = c(1,3)))

forGraph$size = factor(forGraph$size, levels = c("S", "L"))
for(co in 2:15){
  forGraph[,co] = as.numeric(forGraph[,co])
}
```

ANOVAs

```
# Compare GN to PU on *scaled* dims o interest
forkable = data.frame(Dimension = c(""),
                      `S_Mean_PU` = c(0),
                      `L_Mean_PU` = c(0),
                      `S Mean GN` = c(0),
                      `L Mean GN` = c(0),
                      effect = c(""),
                      F.val = c(0),
                      df_a = c(0),
                      df_e = c(0),
                      eta_sq = c(0),
                      p.val = c(0))
forkable = forkable[-1,]

for(d in names(WR.scaledInteract)[5:11]){

  sm.PU = mean(dplyr::pull(WR.scaledInteract[WR.scaledInteract$size == "Small" & WR.scaledInteract$Loca
                     na.rm = T)
  lm.PU = mean(dplyr::pull(WR.scaledInteract[WR.scaledInteract$size == "Large" &
                                             WR.scaledInteract$Location == "PU", d]),
                     na.rm = T)
  sm.GN = mean(dplyr::pull(WR.scaledInteract[WR.scaledInteract$size == "Small" &
                                             WR.scaledInteract$Location == "GN", d]),
                     na.rm = T)
  lm.GN = mean(dplyr::pull(WR.scaledInteract[WR.scaledInteract$size == "Large" &
                                             WR.scaledInteract$Location == "GN", d]),
                     na.rm = T)
  forkable = rbind(forkable,
                   data.frame(Dimension = d,
                              `S_Mean_PU` = sm.PU,
                              `L_Mean_PU` = lm.PU,
                              `S Mean GN` = sm.GN,
                              `L Mean GN` = lm.GN,
                              effect = "",
                              F.val = NaN,
                              df_a = NaN,
                              df_e = NaN,
                              eta_sq = NaN,
                              p.val = NaN)
                   )

  tmp.ANOVA = WR.scaledInteract %>%
```

```r
                anova_test(dv = d,
                           wid = WordPair,
                           within = size,
                           between = Location)
  cat(paste("\n\n", d, "\n\n"))
  print(tmp.ANOVA)
  for (r in 1:nrow(tmp.ANOVA)) {
    forkable = rbind(forkable,
                     data.frame(Dimension = NA,
                              `S_Mean_PU` = NaN,
                              `L_Mean_PU` = NaN,
                              `S Mean GN` = NaN,
                              `L Mean GN` = NaN,
                              effect = tmp.ANOVA$Effect[r],
                              F.val = round(tmp.ANOVA$F[r], 3),
                              df_a = round(tmp.ANOVA$DFn[r],3),
                              df_e = round(tmp.ANOVA$DFd[r], 3),
                              eta_sq = round(tmp.ANOVA$ges[r],3),
                              p.val = round(tmp.ANOVA$p[r], 3))
    )
  }
}
```

```r
options(knitr.kable.NA = "")
print(knitr::kable(forkable, bookend = T, row.names = F, digits = 3))
```

Dimension	S_Mean_PU	L_Mean_PU	S.Mean.GN	L.Mean.GN	effect	F.val	df_a	df_e	eta_sq	p.val
Siz	-0.957	0.957	-0.915	0.937						
					Location	0.005	1	85	0.000	0.946
					size	1357.029	1	85	0.900	0.000
					Location:size	0.490	1	85	0.003	0.486
Aro	-0.154	0.154	-0.228	0.234						
					Location	0.004	1	85	0.000	0.952
					size	7.639	1	85	0.041	0.007
					Location:size	0.427	1	85	0.002	0.515
Val	-0.163	0.163	-0.155	0.158						
					Location	0.058	1	85	0.000	0.810
					size	3.688	1	85	0.023	0.058
					Location:size	0.027	1	85	0.000	0.871
Fam	0.320	-0.320	0.207	-0.212						
					Location	0.002	1	85	0.000	0.968
					size	17.990	1	85	0.075	0.000
					Location:size	0.527	1	85	0.002	0.470
Ima	0.016	-0.016	-0.185	0.189						
					Location	0.003	1	85	0.000	0.953
					size	1.130	1	85	0.007	0.291
					Location:size	1.634	1	85	0.009	0.205
Con	0.082	-0.082	-0.062	0.063						
					Location	0.022	1	85	0.000	0.884
					size	0.064	1	85	0.000	0.801
					Location:size	0.574	1	85	0.004	0.451
Gen	-0.451	0.451	-0.469	0.480						

Dimension	S_Mean_HU	L_Mean_HU	S.Mean.GN	L.Mean.GN	effect	F.val	df_a	df_e	eta_sq	p.val
					Location	0.007	1	85	0.000	0.932
					size	52.327	1	85	0.217	0.000
					Location:size	0.062	1	85	0.000	0.804

Study 3: Multi-level Mediation

RT

```
LDT.finalData$Familiarity_c = LDT.finalData$Familiarity - mean(LDT.finalData$Familiarity)
LDT.finalData$Gender   = 7 - LDT.finalData$Femininity
LDT.finalData$Gender_c = LDT.finalData$Gender - mean(LDT.finalData$Gender)
LDT.finalData$Size_c   = LDT.finalData$Size - mean(LDT.finalData$Size)

WR.data$Familiarity_c = WR.data$Familiarity - mean(WR.data$Familiarity)
WR.data$Size_c = WR.data$Size - mean(WR.data$Size)
WR.data$Gender_c = WR.data$Gender - mean(WR.data$Gender)

mlm_1 = lmer(RT ~ Size_c *  + (1 | SubID), LDT.finalData)
summary(mlm_1)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: RT ~ Size_c * +(1 | SubID)
##    Data: LDT.finalData
##
## REML criterion at convergence: 102819.9
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.6287 -0.6657 -0.0896  0.5470  6.4364
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  SubID    (Intercept) 3853     62.07
##  Residual             6456     80.35
## Number of obs: 8820, groups:  SubID, 108
##
## Fixed effects:
##              Estimate Std. Error        df t value Pr(>|t|)
## (Intercept)  522.5662     6.0342  106.9182  86.601  < 2e-16 ***
## Size_c        -1.5196     0.4688 8711.3687  -3.241  0.00119 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##        (Intr)
## Size_c -0.001
```

Familiarity and Gender Plots

```
# Familiarity
mlm_3 = lmer(RT ~ Familiarity_c + Size_c + (1 | SubID), LDT.finalData)
summary(mlm_3)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: RT ~ Familiarity_c + Size_c + (1 | SubID)
##    Data: LDT.finalData
##
## REML criterion at convergence: 102585.5
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.8036 -0.6544 -0.0925  0.5414  6.6994
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  SubID    (Intercept) 3840     61.97
##  Residual             6288     79.29
## Number of obs: 8820, groups:  SubID, 108
##
## Fixed effects:
##                Estimate Std. Error        df t value Pr(>|t|)
## (Intercept)    522.9317     6.0222  106.9235  86.834  < 2e-16 ***
## Familiarity_c  -24.4703     1.5973 8710.4267 -15.320  < 2e-16 ***
## Size_c          -3.7630     0.4853 8710.3861  -7.754 9.88e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr) Fmlrt_
## Familirty_c -0.004
## Size_c      -0.003  0.302
```

```
leg_1 = lmer(Familiarity_c ~ Size_c + (1|SubID), LDT.finalData[LDT.finalData$RT,])
summary(leg_1)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: Familiarity_c ~ Size_c + (1 | SubID)
##    Data: LDT.finalData[LDT.finalData$RT, ]
##
## REML criterion at convergence: 10054.9
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.2149 -0.6355  0.4821  0.7999  2.1913
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
```

```
##  SubID    (Intercept) 0.007394 0.08599
##  Residual             0.179512 0.42369
## Number of obs: 8820, groups:  SubID, 108
##
## Fixed effects:
##               Estimate Std. Error         df t value Pr(>|t|)
## (Intercept)  2.308e-02  9.590e-03  1.130e+02   2.407   0.0177 *
## Size_c      -1.064e-01  2.951e-03  8.814e+03 -36.071   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##        (Intr)
## Size_c 0.169
```

```r
# rt_fam_bmlm = mlm(LDT.finalData[!is.na(LDT.finalData$RT),],
#                    id = "SubID",
#                    x = "Size_c",
#                    m = "Familiarity_c",
#                    y = "RT")
mlm_summary(rt_fam_bmlm)
```

```
##   Parameter   Mean   SE Median   2.5%  97.5% n_eff Rhat
## 1         a  -0.09 0.00  -0.09  -0.10  -0.09  6832    1
## 2         b -24.49 1.67 -24.50 -27.72 -21.27  7505    1
## 3        cp  -3.76 0.49  -3.76  -4.73  -2.83  6672    1
## 4        me   2.24 0.17   2.24   1.91   2.59  7232    1
## 5         c  -1.51 0.47  -1.52  -2.46  -0.59  7025    1
## 6       pme  -1.74 1.68  -1.48  -3.90  -0.87  4232    1
```

```r
# Gender
mlm_2 = lmer(RT ~ Gender_c + Size_c + (1 | SubID), LDT.finalData)
summary(mlm_2)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: RT ~ Gender_c + Size_c + (1 | SubID)
##    Data: LDT.finalData
##
## REML criterion at convergence: 102810.9
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.5795 -0.6643 -0.0917  0.5441  6.4890
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  SubID    (Intercept) 3853     62.07
##  Residual             6452     80.32
## Number of obs: 8820, groups:  SubID, 108
##
## Fixed effects:
##              Estimate Std. Error        df t value Pr(>|t|)
```

```
## (Intercept)  522.5949     6.0340  106.9190  86.608  < 2e-16 ***
## Gender_c      -3.0248     1.1563 8710.3182  -2.616  0.00891 **
## Size_c        -0.9001     0.5251 8710.3042  -1.714  0.08653 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##          (Intr) Gndr_c
## Gender_c -0.002
## Size_c    0.000 -0.451
```

```
leg_2 = lm(Gender ~ Size, WR.data)
summary(leg_2)
```

```
##
## Call:
## lm(formula = Gender ~ Size, data = WR.data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.72003 -0.49080 -0.00952  0.47316  1.80037
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.26212    0.19688  11.490  < 2e-16 ***
## Size         0.20476    0.04308   4.753 7.77e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7466 on 88 degrees of freedom
## Multiple R-squared:  0.2043, Adjusted R-squared:  0.1952
## F-statistic: 22.59 on 1 and 88 DF,  p-value: 7.768e-06
```

```
# rt_gend_bmlm = mlm(LDT.finalData[!is.na(LDT.finalData$RT),],
#                    id = "SubID",
#                    x = "Size_c",
#                    m = "Gender_c",
#                    y = "RT")
mlm_summary(rt_gend_bmlm)
```

```
##   Parameter  Mean   SE Median  2.5% 97.5% n_eff Rhat
## 1         a  0.20 0.00   0.20  0.20  0.21  9405    1
## 2         b -3.01 1.43  -3.01 -5.88 -0.11  2938    1
## 3        cp -0.89 0.54  -0.88 -1.96  0.15  4575    1
## 4        me -0.62 0.29  -0.62 -1.20 -0.03  2932    1
## 5         c -1.50 0.50  -1.50 -2.49 -0.54  5707    1
## 6       pme  0.46 0.63   0.41  0.01  1.21  3729    1
```

```
# Double
mlm_4 = lmer(RT ~ Familiarity_c + Gender_c + Size_c + (1 | SubID), LDT.finalData)
summary(mlm_4)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: RT ~ Familiarity_c + Gender_c + Size_c + (1 | SubID)
##    Data: LDT.finalData
##
## REML criterion at convergence: 102541.8
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.7249 -0.6550 -0.0858  0.5464  6.8612
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  SubID    (Intercept) 3838     61.95
##  Residual             6258     79.11
## Number of obs: 8820, groups:  SubID, 108
##
## Fixed effects:
##                Estimate Std. Error        df t value Pr(>|t|)
## (Intercept)    523.0408     6.0206  106.9257  86.875  < 2e-16 ***
## Familiarity_c  -26.9641     1.6397 8709.4804 -16.445  < 2e-16 ***
## Gender_c        -7.5648     1.1718 8709.3658  -6.456 1.14e-10 ***
## Size_c          -2.4423     0.5256 8709.3024  -4.647 3.42e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr) Fmlrt_ Gndr_c
## Familirty_c -0.005
## Gender_c    -0.003  0.236
## Size_c      -0.001  0.178 -0.389
```

```
# Fit
anova(mlm_1, mlm_2, mlm_3, mlm_4)
```

```
## refitting model(s) with ML (instead of REML)
```

```
## Data: LDT.finalData
## Models:
## mlm_1: RT ~ Size_c * +(1 | SubID)
## mlm_2: RT ~ Gender_c + Size_c + (1 | SubID)
## mlm_3: RT ~ Familiarity_c + Size_c + (1 | SubID)
## mlm_4: RT ~ Familiarity_c + Gender_c + Size_c + (1 | SubID)
##       npar    AIC    BIC logLik deviance    Chisq Df Pr(>Chisq)
## mlm_1    4 102834 102862 -51413   102826
## mlm_2    5 102829 102864 -51409   102819   6.8423  1   0.008902 **
## mlm_3    5 102604 102639 -51297   102594 224.8290  0
## mlm_4    6 102564 102607 -51276   102552  41.5907  1  1.125e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

ACC

```r
mlm_1 = glmer(ACCURACY ~ Size + (1 | SubID), LDT.finalData, family = "binomial", control=glmerControl(op
summary(mlm_1)
```

```
## Generalized linear mixed model fit by maximum likelihood (Laplace
##   Approximation) [glmerMod]
##  Family: binomial  ( logit )
## Formula: ACCURACY ~ Size + (1 | SubID)
##    Data: LDT.finalData
## Control: glmerControl(optimizer = "bobyqa", optCtrl = list(maxfun = 2e+06))
##
##      AIC      BIC   logLik deviance df.resid
##   2432.5   2454.0  -1213.2   2426.5     9717
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -8.3466  0.1260  0.1491  0.1821  0.3256
##
## Random effects:
##  Groups Name        Variance Std.Dev.
##  SubID  (Intercept) 0.4269   0.6534
## Number of obs: 9720, groups:  SubID, 108
##
## Fixed effects:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  3.37654    0.16433  20.547  < 2e-16 ***
## Size         0.09263    0.03361   2.756  0.00584 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##      (Intr)
## Size -0.797
```

```r
exp(fixef(mlm_1))
```

```
## (Intercept)        Size
##   29.269405    1.097061
```

Familiarity and Gender Plots

```r
# Familiarity
mlm_3 = glmer(ACCURACY ~ Familiarity_c + Size_c + (1 | SubID), LDT.finalData, family = "binomial", cont
summary(mlm_3)
```

```
## Generalized linear mixed model fit by maximum likelihood (Laplace
##   Approximation) [glmerMod]
##  Family: binomial  ( logit )
```

```
## Formula: ACCURACY ~ Familiarity_c + Size_c + (1 | SubID)
##    Data: LDT.finalData
## Control: glmerControl(optimizer = "bobyqa", optCtrl = list(maxfun = 2e+05))
##
##      AIC      BIC   logLik deviance df.resid
##   2408.5   2437.3  -1200.3   2400.5     9716
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -9.2390  0.1210  0.1463  0.1813  0.4723
##
## Random effects:
##  Groups Name        Variance Std.Dev.
##  SubID  (Intercept) 0.4317   0.657
## Number of obs: 9720, groups:  SubID, 108
##
## Fixed effects:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)     3.8098     0.1017  37.480  < 2e-16 ***
## Familiarity_c   0.5609     0.1050   5.343 9.14e-08 ***
## Size_c          0.1415     0.0344   4.113 3.90e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr) Fmlrt_
## Familirty_c 0.172
## Size_c      0.140  0.280
```

```
exp(fixef(mlm_3))
```

```
##   (Intercept) Familiarity_c        Size_c
##     45.142728      1.752178      1.152026
```

```
leg_1 = lm(Familiarity ~ Size, WR.data)
summary(leg_1)
```

```
##
## Call:
## lm(formula = Familiarity ~ Size, data = WR.data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.5561 -0.2800  0.1031  0.4070  0.8985
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  6.05586    0.14178   42.71  < 2e-16 ***
## Size        -0.08874    0.03102   -2.86  0.00528 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5376 on 88 degrees of freedom
```

```
## Multiple R-squared:  0.08507,    Adjusted R-squared:  0.07467
## F-statistic: 8.182 on 1 and 88 DF,  p-value: 0.005284
```

```r
# acc_fam_bmlm = mlm(LDT.finalData[!is.na(LDT.finalData$RT),],
#                    id = "SubID",
#                    x = "Size_c",
#                    m = "Familiarity_c",
#                    y = "ACCURACY",
#                    binary_y = T)
mlm_summary(acc_fam_bmlm)
```

```
##   Parameter    Mean     SE  Median     2.5%  97.5% n_eff Rhat
## 1         a   -0.09   0.00   -0.09    -0.10  -0.09  3117 1.00
## 2         b -229.34 526.04 -184.22 -1364.97 716.49  1073 1.00
## 3        cp  -16.69 264.45  -14.77  -544.56 521.01  1066 1.01
## 4        me   21.03  48.24   16.79   -66.13 125.07  1076 1.00
## 5         c    4.34 264.59   11.04  -522.75 548.28  1070 1.00
## 6       pme    0.05   7.19    0.02    -2.44   3.10  3963 1.00
```

```r
# Gender
mlm_2 = glmer(ACCURACY ~ Gender_c + Size_c + (1 | SubID), LDT.finalData, family = "binomial", control=g
summary(mlm_2)
```

```
## Generalized linear mixed model fit by maximum likelihood (Laplace
##   Approximation) [glmerMod]
##  Family: binomial  ( logit )
## Formula: ACCURACY ~ Gender_c + Size_c + (1 | SubID)
##    Data: LDT.finalData
## Control: glmerControl(optimizer = "bobyqa", optCtrl = list(maxfun = 2e+05))
##
##      AIC      BIC   logLik deviance df.resid
##   2431.0   2459.7  -1211.5   2423.0     9716
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -8.6517  0.1262  0.1489  0.1798  0.3575
##
## Random effects:
##  Groups Name        Variance Std.Dev.
##  SubID  (Intercept) 0.4275   0.6539
## Number of obs: 9720, groups:  SubID, 108
##
## Fixed effects:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  3.77098    0.10011  37.669   <2e-16 ***
## Gender_c     0.15574    0.08173   1.905   0.0567 .
## Size_c       0.06082    0.03770   1.613   0.1067
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##          (Intr) Gndr_c
## Gender_c  0.067
## Size_c    0.059 -0.448
```

```r
exp(fixef(mlm_2))
```

```
## (Intercept)    Gender_c      Size_c
##   43.422670    1.168518    1.062703
```

```r
leg_2 = lm(Gender ~ Size, WR.data)
summary(leg_2)
```

```
##
## Call:
## lm(formula = Gender ~ Size, data = WR.data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.72003 -0.49080 -0.00952  0.47316  1.80037
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.26212    0.19688  11.490  < 2e-16 ***
## Size         0.20476    0.04308   4.753 7.77e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7466 on 88 degrees of freedom
## Multiple R-squared:  0.2043, Adjusted R-squared:  0.1952
## F-statistic: 22.59 on 1 and 88 DF,  p-value: 7.768e-06
```

```r
# acc_gend_bmlm = mlm(LDT.finalData[!is.na(LDT.finalData$RT),],
#                    id = "SubID",
#                    x = "Size_c",
#                    m = "Gender_c",
#                    y = "ACCURACY",
#                    binary_y = T)
mlm_summary(acc_gend_bmlm)
```

```
##   Parameter   Mean     SE Median    2.5%  97.5% n_eff Rhat
## 1         a   0.20   0.00   0.20    0.20   0.21  3703    1
## 2         b  -9.98 411.11  -9.05 -834.57 812.01  1427    1
## 3        cp -36.86 284.61 -40.15 -609.56 555.05  1213    1
## 4        me  -2.03  84.30  -1.80 -170.90 165.55  1428    1
## 5         c -38.89 276.17 -48.61 -580.20 544.55  1332    1
## 6       pme   0.45  28.00   0.02   -3.10   4.13  4012    1
```

```r
# save(list = c("rt_gend_bmlm", "rt_fam_bmlm", "acc_fam_bmlm", "acc_gend_bmlm"),
#      file = "All_Bayesian_Mediations.RData")


## Double
mlm_4 = glmer(ACCURACY ~ Familiarity_c + Gender_c + Size_c + (1 | SubID), LDT.finalData, family = "binor
summary(mlm_4)
```

```
## Generalized linear mixed model fit by maximum likelihood (Laplace
##   Approximation) [glmerMod]
##  Family: binomial  ( logit )
## Formula: ACCURACY ~ Familiarity_c + Gender_c + Size_c + (1 | SubID)
##    Data: LDT.finalData
## Control: glmerControl(optimizer = "bobyqa", optCtrl = list(maxfun = 2e+05))
##
##      AIC      BIC   logLik deviance df.resid
##   2400.3   2436.2  -1195.1   2390.3     9715
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -9.6496  0.1183  0.1460  0.1811  0.4583
##
## Random effects:
##  Groups Name        Variance Std.Dev.
##  SubID  (Intercept) 0.4334   0.6583
## Number of obs: 9720, groups:  SubID, 108
##
## Fixed effects:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)    3.82998    0.10250  37.365  < 2e-16 ***
## Familiarity_c  0.65661    0.10956   5.993 2.05e-09 ***
## Gender_c       0.27648    0.08456   3.270  0.00108 **
## Size_c         0.09475    0.03756   2.523  0.01164 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr) Fmlrt_ Gndr_c
## Familirty_c  0.198
## Gender_c     0.117  0.275
## Size_c       0.085  0.154 -0.385
```

```r
exp(fixef(mlm_4))
```

```
##   (Intercept) Familiarity_c      Gender_c        Size_c
##     46.061434      1.928252      1.318475      1.099389
```

```r
# Best Fit
anova(mlm_1, mlm_2, mlm_3, mlm_4)
```

```
## Data: LDT.finalData
## Models:
## mlm_1: ACCURACY ~ Size + (1 | SubID)
## mlm_2: ACCURACY ~ Gender_c + Size_c + (1 | SubID)
## mlm_3: ACCURACY ~ Familiarity_c + Size_c + (1 | SubID)
## mlm_4: ACCURACY ~ Familiarity_c + Gender_c + Size_c + (1 | SubID)
##       npar    AIC    BIC  logLik deviance   Chisq Df Pr(>Chisq)
## mlm_1    3 2432.5 2454.0 -1213.2   2426.5
## mlm_2    4 2431.0 2459.7 -1211.5   2423.0  3.4852  1   0.061921 .
## mlm_3    4 2408.5 2437.3 -1200.3   2400.5 22.4498  0
## mlm_4    5 2400.2 2436.2 -1195.1   2390.2 10.2807  1   0.001344 **
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```