Paper\_report

Elvio Blini GuFei

2020-03-25

# Experiment 1-in a virtual environment

We focus on accuracy and reaction times (RTs) for correct answers that were provided within 100-500 ms.

For any request or inquiry don’t hesitate to contact: [elvio.blini@gmail.com](mailto:elvio.blini@gmail.com)

#list packages  
packages= c("ggplot2", "plyr", "BayesFactor", "lme4", "reshape", "gridExtra", "plot3D", "afex", "effsize","dplyr")  
  
#load them  
packages <- lapply(packages, require, character.only= T)

## Loading required package: ggplot2

## Loading required package: plyr

## Loading required package: BayesFactor

## Loading required package: coda

## Loading required package: Matrix

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

## Loading required package: lme4

## Loading required package: reshape

##   
## Attaching package: 'reshape'

## The following object is masked from 'package:Matrix':  
##   
## expand

## The following objects are masked from 'package:plyr':  
##   
## rename, round\_any

## Loading required package: gridExtra

## Loading required package: plot3D

## Loading required package: afex

## 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

## Loading required package: effsize

## Loading required package: dplyr

##   
## Attaching package: 'dplyr'

## The following object is masked from 'package:gridExtra':  
##   
## combine

## The following object is masked from 'package:reshape':  
##   
## rename

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

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

# load data  
load("Exp 1 data.RData")  
head(data)

## SubjNb SubjID StimGp Phase Dist Height Color Correct.Answer Subject.Answer  
## 1 2 clem 0 practice 0.5 -0.2 green CUBE SPHERE  
## 2 2 clem 0 practice 3.0 -0.2 red CUBE SPHERE  
## 3 2 clem 0 practice 0.5 -0.2 green SPHERE NOANSWER  
## 4 2 clem 0 practice 3.0 -0.2 green SPHERE CUBE  
## 5 2 clem 0 practice 0.5 -0.2 red CUBE SPHERE  
## 6 2 clem 0 practice 0.5 -0.2 blue SPHERE CUBE  
## Accur. R\_time H\_Pos  
## 1 0 425.9443 Hand Close  
## 2 0 733.2939 Hand Close  
## 3 0 0.0000 Hand Close  
## 4 0 466.6616 Hand Close  
## 5 0 573.3799 Hand Close  
## 6 0 600.0080 Hand Close

## Preprocessing

The factors Distance (Dist) and Subject (SubjNb) are to be converted into factors. Accuracy (Accur.) is numeric instead.

data$Dist= as.factor(data$Dist)  
data$SubjNb= as.factor(data$SubjNb)  
data$Accur.= as.numeric(as.character(data$Accur.))  
str(data)

## 'data.frame': 10560 obs. of 12 variables:  
## $ SubjNb : Factor w/ 20 levels "1","2","3","4",..: 2 2 2 2 2 2 2 2 2 2 ...  
## $ SubjID : Factor w/ 20 levels "abko","alla",..: 6 6 6 6 6 6 6 6 6 6 ...  
## $ StimGp : int 0 0 0 0 0 0 0 0 0 0 ...  
## $ Phase : Factor w/ 2 levels "experiment","practice": 2 2 2 2 2 2 2 2 2 2 ...  
## $ Dist : Factor w/ 2 levels "0.5","3": 1 2 1 2 1 1 2 1 2 1 ...  
## $ Height : num -0.2 -0.2 -0.2 -0.2 -0.2 -0.2 -0.2 -0.2 -0.2 -0.2 ...  
## $ Color : Factor w/ 3 levels "blue","green",..: 2 3 2 2 3 1 1 3 2 2 ...  
## $ Correct.Answer: Factor w/ 2 levels "CUBE","SPHERE": 1 1 2 2 1 2 2 2 1 1 ...  
## $ Subject.Answer: Factor w/ 3 levels "CUBE","NOANSWER",..: 3 3 2 1 3 1 1 3 1 1 ...  
## $ Accur. : num 0 0 0 0 0 0 0 1 1 1 ...  
## $ R\_time : num 426 733 0 467 573 ...  
## $ H\_Pos : Factor w/ 2 levels "Hand Close","Hand Far": 1 1 1 1 1 1 1 1 1 1 ...

Practice trials were removed.

data= data[data$Phase=="experiment",]

The Oculus Rift has a fixed latency of 20 ms that we subtract from subjects’ reaction times. Note that this is a constant value and does not affect statistics afterwards.

data$R\_time= data$R\_time - 20

A response was considered Valid if it was correct and provided within 100 and 500 ms. We create a variable equals to 1 if this condition is met.

data$Valid= ifelse(data$Accur. == 1 &   
 data$R\_time>100 &   
 data$R\_time<500, 1, 0)

## Summarise accuracy

正确率

We are interested, for this part, in the role of distance (depth of the shape). We also manipulated the position of the non-dominant hand, that could be close to the body or at approximately 50 cm away.

# tapply按照第二个参数进行分组同统计  
acc.m.s= tapply(data$Accur.,  
 list(data$SubjNb, data$Dist, data$H\_Pos),  
 mean)

Then we obtain the grand average, sd, and sem.

#this averages across subjects (displayed)  
(acc.m= apply(acc.m.s, c(2:3), mean))

## Hand Close Hand Far  
## 0.5 0.89125 0.890  
## 3 0.88375 0.895

# print("SD")  
# sprintf("\n")  
cat("\nSD\n")

##   
## SD

#this calculates the standard deviation between subjects  
(acc.sd= apply(acc.m.s, c(2:3), sd))

## Hand Close Hand Far  
## 0.5 0.06906196 0.08862992  
## 3 0.07342274 0.09602479

# 标准误  
#this divides the sd by the square root of subjects' N  
acc.sem= acc.sd/(sqrt(length(levels(data$SubjNb))))

去掉错误的和时间不在100-500的剩余比例 What is the average percentage of trials omitted for both incorrect responses and slow RTs?

val.m.s= tapply(data$Valid,  
 list(data$SubjNb, data$Dist, data$H\_Pos),  
 mean)  
  
#this averages across subjects (displayed)  
(val.m= apply(val.m.s, c(2:3), mean))

## Hand Close Hand Far  
## 0.5 0.8304167 0.8350000  
## 3 0.8133333 0.8216667

# We now want to save the data frame as a separate object to run analyses over accuracy afterwards.  
acc.an= data  
# Indeed, we now exclude wrong (plus too slow or too fast) responses to further assess RTs.  
data= data[data$Valid==1,]

# Summarise RTs

rts.m.s= tapply(data$R\_time,  
 list(data$SubjNb, data$Dist, data$H\_Pos),  
 mean)  
  
#this averages across subjects (displayed)  
(rts.m= apply(rts.m.s, c(2:3), mean))

## Hand Close Hand Far  
## 0.5 367.1763 364.5071  
## 3 376.5204 374.4020

#this calculates the standard deviation between subjects  
rts.sd= apply(rts.m.s, c(2:3), sd)  
  
#this divides the sd by the square root of subjects' N  
rts.sem= rts.sd/(sqrt(length(levels(data$SubjNb))))

# Analyses - Mixed Models

We now run statistics using (general) linear mixed-effects models.

The general strategy is to evaluate beforehand the random effects that increase model fitting, as to reach a parsimonious solution (i.e. supported by data). We create several different (nested) models and evaluate them against a simpler reference one through likelihood ratio tests (LRT). This holds for both random and fixed effects testing.

The simplest model to start with only includes the random intercept for Subjects (baseline level). We then start testing random slopes one by one, following this order:

1. Distance
2. Hand Position
3. Color of the shape
4. Shape (that is, correct answer)

Each random slope - that informs about variability in performance across levels of a factor, e.g. differences in experimental manipulations across subjects - will be retained in the model if the LRT is proven significant. Following evaluations will be made with reference models that include this slope. For example, if Distance improves model fit as random slope, Hand Position will be evaluated against the model including it. As a second step we introduce interactions for all combinations of slopes proven significant (this is to respect marginality, and thus include high-order terms only together with their lower-order ones).

Fixed effects testing will use a similar (type 2) approach.

(To avoid verbosity, only the p value is shown).

## Accuracy

We start with the simplest model. Note that accuracy is binomial, thus we call for the general linear mixed effect regression (glmer) function and specify the family accordingly. I’m also asking for the “bobyqa” optimizer, which handles convergence problems very well.

使用的是glmer，即广义的线性模型，需要指定family

mod0=glmer(Accur. ~ (1|SubjNb), data=acc.an, family=binomial,   
 control=glmerControl(optimizer="bobyqa"))  
cat("\nDistance\n")

##   
## Distance

#random slope for distance  
mod0a=glmer(Accur. ~ (1+Dist|SubjNb), data=acc.an, family=binomial,   
 control=glmerControl(optimizer="bobyqa"))  
  
#LRT  
anova(mod0, mod0a)$`Pr(>Chisq)`[2] #nope

## [1] 0.06280606

cat("\nHand position\n")

##   
## Hand position

#random slope for hand position  
mod0b=glmer(Accur. ~ (1+H\_Pos|SubjNb), data=acc.an, family=binomial,  
 control=glmerControl(optimizer="bobyqa"))  
  
#LRT   
anova(mod0, mod0b)$`Pr(>Chisq)`[2] #yes

## [1] 0.007067707

cat("\nColor\n")

##   
## Color

#slope for color  
mod0c=glmer(Accur. ~ (1+H\_Pos+Color|SubjNb), data=acc.an, family=binomial,  
 control=glmerControl(optimizer="bobyqa"))

## boundary (singular) fit: see ?isSingular

anova(mod0b, mod0c)$`Pr(>Chisq)`[2]

## [1] 0.9202306

cat("\nShape of the stimulus\n")

##   
## Shape of the stimulus

#slope for correct response  
mod0d=glmer(Accur. ~ (1+H\_Pos+Correct.Answer|SubjNb), data=acc.an, family=binomial,  
 control=glmerControl(optimizer="bobyqa"))  
  
anova(mod0b, mod0d)$`Pr(>Chisq)`[2]

## [1] 7.131084e-05

# 有些问题，是因为颜色的虚拟变量相关太高  
isSingular(mod0c,tol = 1e-5)

## [1] TRUE

summary(mod0c)

## Generalized linear mixed model fit by maximum likelihood (Laplace  
## Approximation) [glmerMod]  
## Family: binomial ( logit )  
## Formula: Accur. ~ (1 + H\_Pos + Color | SubjNb)  
## Data: acc.an  
## Control: glmerControl(optimizer = "bobyqa")  
##   
## AIC BIC logLik deviance df.resid   
## 6238.3 6317.2 -3108.2 6216.3 9589   
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -7.4046 0.2185 0.2849 0.3739 0.6637   
##   
## Random effects:  
## Groups Name Variance Std.Dev. Corr   
## SubjNb (Intercept) 0.48303 0.6950   
## H\_PosHand Far 0.10954 0.3310 0.49   
## Colorgreen 0.02438 0.1561 -0.21 -0.95   
## Colorred 0.01862 0.1364 -0.29 -0.98 1.00  
## Number of obs: 9600, groups: SubjNb, 20  
##   
## Fixed effects:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) 2.2016 0.1871 11.77 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
## convergence code: 0  
## boundary (singular) fit: see ?isSingular

cat("\nInteraction of Hand and Shape\n")

##   
## Interaction of Hand and Shape

#interaction  
mod0e=glmer(Accur. ~ (1+H\_Pos\*Correct.Answer|SubjNb), data=acc.an, family=binomial,  
 control=glmerControl(optimizer="bobyqa"))

## boundary (singular) fit: see ?isSingular

anova(mod0d, mod0e)$`Pr(>Chisq)`[2]

## [1] 0.6816193

# 确定最后的模型  
mod.null= mod0d

正式开始检验固定效应，被试是随机因子，模型中+表示两个的主效应，\*表示主效应和交互作用，:表示只是交互作用

cat("\n距离对正确率不显著\n")

##   
## 距离对正确率不显著

# 距离对正确率不显著  
#distance as fixed effect  
mod1=glmer(Accur. ~ Dist+ (1+H\_Pos+Correct.Answer|SubjNb), data=acc.an, family=binomial,  
 control=glmerControl(optimizer="bobyqa"))  
  
#hand position as fixed effect  
mod2=glmer(Accur. ~ H\_Pos+ (1+H\_Pos+Correct.Answer|SubjNb), data=acc.an, family=binomial,  
 control=glmerControl(optimizer="bobyqa"))  
  
#LRTs  
anova(mod.null, mod1)

## Data: acc.an  
## Models:  
## mod.null: Accur. ~ (1 + H\_Pos + Correct.Answer | SubjNb)  
## mod1: Accur. ~ Dist + (1 + H\_Pos + Correct.Answer | SubjNb)  
## Df AIC BIC logLik deviance Chisq Chi Df Pr(>Chisq)  
## mod.null 7 6211.1 6261.3 -3098.6 6197.1   
## mod1 8 6213.1 6270.4 -3098.5 6197.1 0.0409 1 0.8397

cat("\n手的位置对正确率不显著\n")

##   
## 手的位置对正确率不显著

# 手的位置对正确率不显著  
mod3=glmer(Accur. ~ Dist+H\_Pos+ (1+H\_Pos+Correct.Answer|SubjNb), data=acc.an, family=binomial,  
 control=glmerControl(optimizer="bobyqa"))  
mod4=glmer(Accur. ~ Dist\*H\_Pos+ (1+H\_Pos+Correct.Answer|SubjNb), data=acc.an, family=binomial,  
 control=glmerControl(optimizer="bobyqa"))  
  
#LRT  
anova(mod3, mod4)

## Data: acc.an  
## Models:  
## mod3: Accur. ~ Dist + H\_Pos + (1 + H\_Pos + Correct.Answer | SubjNb)  
## mod4: Accur. ~ Dist \* H\_Pos + (1 + H\_Pos + Correct.Answer | SubjNb)  
## Df AIC BIC logLik deviance Chisq Chi Df Pr(>Chisq)  
## mod3 9 6212.4 6276.9 -3097.2 6194.4   
## mod4 10 6213.4 6285.1 -3096.7 6193.4 1.0086 1 0.3152

## Reaction times

We use the same selection procedure as above. For random effects we use restricted maximum likelihood (REML, that works well when fixed effects in the to-be-compared models are exactly the same). We’ll need to prevent the LRT to refit models. Here’s the simplest model.

rtmod0=lmer(R\_time ~ (1|SubjNb), data=data, REML=T,  
 control=lmerControl(optimizer="bobyqa"))  
cat("\n距离\n")

##   
## 距离

rtmod0a=lmer(R\_time ~ (1+Dist|SubjNb), data=data, REML=T,   
 control=lmerControl(optimizer="bobyqa"))  
   
anova(rtmod0, rtmod0a, refit=F)$`Pr(>Chisq)`[2] #significant, keep it in

## [1] 1.925675e-12

cat("\n增加位置\n")

##   
## 增加位置

rtmod0b=lmer(R\_time ~ (1+Dist+H\_Pos|SubjNb), data=data, REML=T,  
 control=lmerControl(optimizer="bobyqa"))  
   
anova(rtmod0a, rtmod0b, refit=F)$`Pr(>Chisq)`[2] #significant, keep it in

## [1] 1.313702e-15

cat("\n增加颜色\n")

##   
## 增加颜色

rtmod0c=lmer(R\_time ~ (1+Dist+H\_Pos+Color|SubjNb), data=data, REML=T,   
 control=lmerControl(optimizer="bobyqa"))

## boundary (singular) fit: see ?isSingular

anova(rtmod0b, rtmod0c, refit=F)$`Pr(>Chisq)`[2] #nope

## [1] 0.2688566

cat("\n增加形状（应该按的键）\n")

##   
## 增加形状（应该按的键）

rtmod0d=lmer(R\_time ~ (1+Dist+H\_Pos+Correct.Answer|SubjNb), data=data, REML=T,   
 control=lmerControl(optimizer="bobyqa"))  
   
anova(rtmod0b, rtmod0d, refit=F)$`Pr(>Chisq)`[2] #significant, keep it in

## [1] 9.502911e-09

cat("\n增加两两之间的交互\n")

##   
## 增加两两之间的交互

rtmod0e=lmer(R\_time ~ (1+Dist\*H\_Pos+Correct.Answer|SubjNb), data=data, REML=T,   
 control=lmerControl(optimizer="bobyqa"))  
  
anova(rtmod0d, rtmod0e, refit=F)$`Pr(>Chisq)`[2]

## [1] 0.4118331

rtmod0f=lmer(R\_time ~ (1+Dist+H\_Pos\*Correct.Answer|SubjNb), data=data, REML=T,   
 control=lmerControl(optimizer="bobyqa"))  
  
anova(rtmod0d, rtmod0f, refit=F)$`Pr(>Chisq)`[2]

## [1] 0.6211647

rtmod0g=lmer(R\_time ~ (1+H\_Pos+Dist\*Correct.Answer|SubjNb), data=data, REML=T,   
 control=lmerControl(optimizer="bobyqa"))  
   
anova(rtmod0d, rtmod0g, refit=F)$`Pr(>Chisq)`[2] #yes

## [1] 6.337025e-12

rtmod.null= rtmod0g  
rtmod.null= update(rtmod.null, REML=F)

开始检验固定效应I’m still writing models manually, avoiding update for clarity reasons. Another possibility is using afex::mixed, which is a wrapper around lmer useful when fixed factors are many.

cat("\n距离的主效应，和只有随机效应的比较\n")

##   
## 距离的主效应，和只有随机效应的比较

rtmod1=lmer(R\_time ~ Dist+ (1+H\_Pos+Dist\*Correct.Answer|SubjNb), data=data, REML=F,  
 control=lmerControl(optimizer="bobyqa"))  
anova(rtmod.null, rtmod1)

## Data: data  
## Models:  
## rtmod.null: R\_time ~ (1 + H\_Pos + Dist \* Correct.Answer | SubjNb)  
## rtmod1: R\_time ~ Dist + (1 + H\_Pos + Dist \* Correct.Answer | SubjNb)  
## Df AIC BIC logLik deviance Chisq Chi Df Pr(>Chisq)   
## rtmod.null 17 85764 85883 -42865 85730   
## rtmod1 18 85749 85875 -42856 85713 17.523 1 2.838e-05 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

cat("\n手位置的主效应，和只有随机效应的比较\n")

##   
## 手位置的主效应，和只有随机效应的比较

rtmod2=lmer(R\_time ~ H\_Pos+ (1+H\_Pos+Dist\*Correct.Answer|SubjNb), data=data, REML=F,  
 control=lmerControl(optimizer="bobyqa"))  
  
anova(rtmod.null, rtmod2)

## Data: data  
## Models:  
## rtmod.null: R\_time ~ (1 + H\_Pos + Dist \* Correct.Answer | SubjNb)  
## rtmod2: R\_time ~ H\_Pos + (1 + H\_Pos + Dist \* Correct.Answer | SubjNb)  
## Df AIC BIC logLik deviance Chisq Chi Df Pr(>Chisq)  
## rtmod.null 17 85764 85883 -42865 85730   
## rtmod2 18 85765 85891 -42865 85729 1.0567 1 0.304

cat("\n交互作用，和只有主效应的比较\n")

##   
## 交互作用，和只有主效应的比较

rtmod3=lmer(R\_time ~ Dist+H\_Pos + (1+H\_Pos+Dist\*Correct.Answer|SubjNb), data=data, REML=F,  
 control=lmerControl(optimizer="bobyqa"))  
rtmod4=lmer(R\_time ~ Dist\*H\_Pos + (1+H\_Pos+Dist\*Correct.Answer|SubjNb), data=data, REML=F,  
 control=lmerControl(optimizer="bobyqa"))  
   
anova(rtmod3, rtmod4)

## Data: data  
## Models:  
## rtmod3: R\_time ~ Dist + H\_Pos + (1 + H\_Pos + Dist \* Correct.Answer |   
## rtmod3: SubjNb)  
## rtmod4: R\_time ~ Dist \* H\_Pos + (1 + H\_Pos + Dist \* Correct.Answer |   
## rtmod4: SubjNb)  
## Df AIC BIC logLik deviance Chisq Chi Df Pr(>Chisq)  
## rtmod3 19 85750 85883 -42856 85712   
## rtmod4 20 85752 85892 -42856 85712 0.0989 1 0.7532

cat("最终的模型，固定效应的估计是β，对应的显著性不是主效应")

## 最终的模型，固定效应的估计是β，对应的显著性不是主效应

summary(rtmod1)

## Linear mixed model fit by maximum likelihood . t-tests use Satterthwaite's  
## method [lmerModLmerTest]  
## Formula: R\_time ~ Dist + (1 + H\_Pos + Dist \* Correct.Answer | SubjNb)  
## Data: data  
## Control: lmerControl(optimizer = "bobyqa")  
##   
## AIC BIC logLik deviance df.resid   
## 85749.0 85874.6 -42856.5 85713.0 7903   
##   
## Scaled residuals:   
## Min 1Q Median 3Q Max   
## -4.4513 -0.6882 -0.0274 0.6633 3.0743   
##   
## Random effects:  
## Groups Name Variance Std.Dev. Corr   
## SubjNb (Intercept) 316.69 17.796   
## H\_PosHand Far 165.19 12.853 -0.67   
## Dist3 87.88 9.374 0.03 0.09   
## Correct.AnswerSPHERE 138.71 11.777 -0.07 -0.06 0.24   
## Dist3:Correct.AnswerSPHERE 568.17 23.836 -0.06 0.06 -0.82 -0.61  
## Residual 2865.65 53.532   
## Number of obs: 7921, groups: SubjNb, 20  
##   
## Fixed effects:  
## Estimate Std. Error df t value Pr(>|t|)   
## (Intercept) 362.39 3.09 20.09 117.286 < 2e-16 \*\*\*  
## Dist3 13.24 1.65 19.29 8.024 1.44e-07 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Correlation of Fixed Effects:  
## (Intr)  
## Dist3 -0.174

# cat("通过bootstrap得到置信区间")  
# set.seed(1)  
# (ci= confint(rtmod1, method="boot", nsim=500, parm= 18))

cat("距离效应的Cohen's d")

## 距离效应的Cohen's d

X= apply(rts.m.s, 1:2, mean)  
  
cohen.d(X[,2], X[,1], conf.level = 0.95,  
 hedges.correction = FALSE, paired= T)

##   
## Cohen's d  
##   
## d estimate: 0.6198865 (medium)  
## 95 percent confidence interval:  
## lower upper   
## 0.3578346 0.8819384

cat("位置效应的Cohen's d")

## 位置效应的Cohen's d

X= apply(rts.m.s, c(1, 3), mean)  
  
cohen.d(X[,2], X[,1], conf.level = 0.95,  
 hedges.correction = FALSE, paired= T)

##   
## Cohen's d  
##   
## d estimate: -0.1403023 (negligible)  
## 95 percent confidence interval:  
## lower upper   
## -0.5282632 0.2476586

# Robustness checks

Mixed models有很多设置的自由，怎么设置参数有一些tricky，Barr建议把随机效应都放进去，但会有不收敛和singular matrix的问题，所以他们自己建立了一套方法，同时这里用方差分析的方法进行了验证

## ANOVA

DF= ddply(data, c("SubjNb", "Dist", "H\_Pos"), summarise, dv= mean(R\_time))  
  
AOV= summary(aov\_ez(id= "SubjNb", dv= "dv", within= c("Dist", "H\_Pos"), data= DF))  
AOV

##   
## Univariate Type III Repeated-Measures ANOVA Assuming Sphericity  
##   
## Sum Sq num Df Error SS den Df F value Pr(>F)   
## (Intercept) 10990600 1 16936.4 19 12329.7216 < 2.2e-16 \*\*\*  
## Dist 1851 1 1286.2 19 27.3378 4.784e-05 \*\*\*  
## H\_Pos 115 1 4023.3 19 0.5412 0.4709   
## Dist:H\_Pos 2 1 879.5 19 0.0328 0.8583   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

#partial eta square  
#can also be seen with:  
#DescTools::EtaSq(aov\_ez(id= "SubjNb", dv= "dv", within= c("Dist", "H\_Pos"), data= DF, return= "aov"), type=1)  
AOV$univariate.tests[,1]/(AOV$univariate.tests[,1] + AOV$univariate.tests[,3])

## (Intercept) Dist H\_Pos Dist:H\_Pos   
## 0.998461379 0.589967535 0.027697793 0.001721779

## Bayesian ANOVA

Second, we perform a Bayesian ANOVA, useful to provide evidence for the lack of Hand Position effect. We use objective priors to avoid a few degrees of freedom (results partly depend on prior choice). This is possible thanks to the BayesFactor package. A Bayes Factor > 1 supports the alternative hypothesis, the null if < 1.

BF= anovaBF(dv ~ Dist\*H\_Pos + SubjNb, data= DF, whichRandom= "SubjNb")  
sort(BF, decreasing= T)

## Bayes factor analysis  
## --------------  
## [1] Dist + SubjNb : 195.9323 ±1%  
## [2] Dist + H\_Pos + SubjNb : 72.19458 ±1.98%  
## [3] Dist + H\_Pos + Dist:H\_Pos + SubjNb : 23.55848 ±5.44%  
## [4] H\_Pos + SubjNb : 0.3272457 ±0.88%  
##   
## Against denominator:  
## dv ~ SubjNb   
## ---  
## Bayes factor type: BFlinearModel, JZS

BF[4]/BF[3]

## Bayes factor analysis  
## --------------  
## [1] Dist + H\_Pos + Dist:H\_Pos + SubjNb : 0.3263192 ±5.79%  
##   
## Against denominator:  
## dv ~ Dist + H\_Pos + SubjNb   
## ---  
## Bayes factor type: BFlinearModel, JZS

## t-test

DF= ddply(DF, c("SubjNb", "Dist"), summarise, dv= mean(dv))  
  
t.test(x= DF$dv[DF$Dist== "3"],   
 y= DF$dv[DF$Dist== "0.5"], paired= TRUE)

##   
## Paired t-test  
##   
## data: DF$dv[DF$Dist == "3"] and DF$dv[DF$Dist == "0.5"]  
## t = 5.2286, df = 19, p-value = 4.784e-05  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## 5.768754 13.470249  
## sample estimates:  
## mean of the differences   
## 9.619501