Portfolio 3: The Reading Experiment

Sebastian Scott Engen

11/09/2017

## Portfolio 3: The Reading Experiment

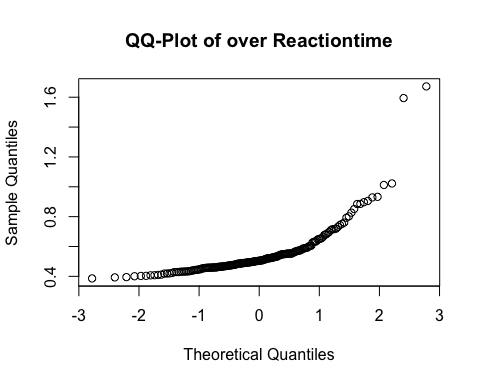
In this portfolio assignment, you will be conducting analyses of data from your reading experiment. The portfolio has two overall sections asking different questions: 1) A correlational section investigating what aspects of words predict reading time, and 2) an experimental section asking about the way that contextual expectancies affect our reading time by contrasting two conditions.

For each analyses, remember to make qualified choices of statistical test based on assumption testing. Futhermore, for each analysis, provide a written report following APA conventions and a plot that illustrates the result

# 1) What properties of words predict word-by-word reading time?  
  
Conduct minimum three correlational analyses where you investigate the relation between reading time and relevant predictors such as e.g. word length, frequencies, etc.

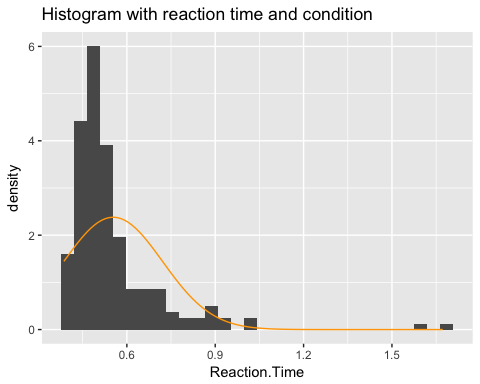
# Checking assumptions

# First we start out by choosing a set of data from our reading experiment - we chose Thomas's data  
data <- read.csv("logfile\_Thomas Hagen Hansen2.csv", sep = ",")  
  
# We start out making a qqplot over reactiontime  
qqnorm(data$Reaction.Time, main = "QQ-Plot of over Reactiontime")



# it doesn't look normally distributed  
  
#We'll try to get a better look by making a histogram over reactiontime   
ggplot(data, aes(x = Reaction.Time))+  
 geom\_histogram(aes(y = ..density..))+  
 stat\_function(fun = dnorm,  
 args = list(mean = mean(data$Reaction.Time),   
 sd = sd(data$Reaction.Time)), color = "orange")+  
 labs(title = "Histogram with reaction time and condition")

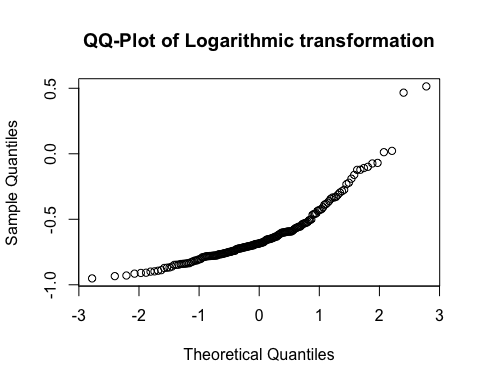
## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



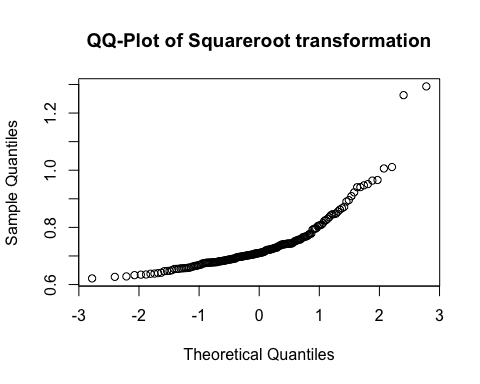
#This also shows us that the data aren't "really" normally distributed  
  
# At last we'll use the stat.desc function:  
round(stat.desc(data$Reaction.Time, basic = FALSE, norm = TRUE),3)

## median mean SE.mean CI.mean.0.95 var   
## 0.504 0.554 0.012 0.024 0.028   
## std.dev coef.var skewness skew.2SE kurtosis   
## 0.168 0.303 3.494 9.752 17.455   
## kurt.2SE normtest.W normtest.p   
## 24.488 0.672 0.000

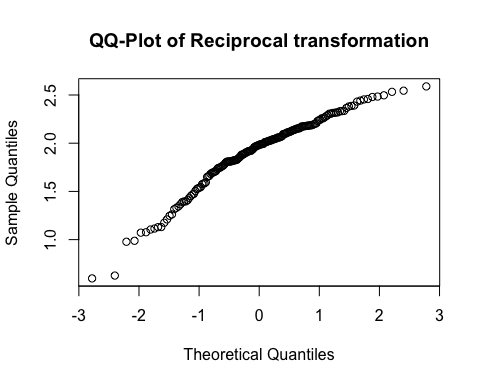
# Which shows us, that yes. We suffer from both kurtosis and positive skew  
  
# We try to do 3 different transformations on our reading data to get it normally distributed  
data$log <- log(data$Reaction.Time)  
qqnorm(data$log, main = "QQ-Plot of Logarithmic transformation")



data$sqrt <- sqrt(data$Reaction.Time)  
qqnorm(data$sqrt, main = "QQ-Plot of Squareroot transformation")



data$RT <- 1/(data$Reaction.Time)  
qqnorm(data$RT, main = "QQ-Plot of Reciprocal transformation")



# # The reciprocal qqplot seems fairly normal - We'll go ahead with these transformed data and check to be sure that it's really normally distribute.   
#We'll return normal distribution statistics on our reciprocal transformation:  
round(stat.desc(data$RT, basic = FALSE, norm = TRUE),3)

## median mean SE.mean CI.mean.0.95 var   
## 1.983 1.908 0.027 0.054 0.138   
## std.dev coef.var skewness skew.2SE kurtosis   
## 0.371 0.194 -0.909 -2.537 0.863   
## kurt.2SE normtest.W normtest.p   
## 1.211 0.947 0.000

#Our skew.2SE and kurt.2SE is -2.537 and 1.211 seperately, which is lower than -1 and higher than 1 seperately, and this means that the skewness and kurtosis is still significantly different from zero  
# So.. Good thing we checked! Now we found out that the reciprocal data still wasn't normally distributed, so we'll use the more robust non-parametric spearmans-test in the coming correlational studies

# Correlational study 1 - Reaction Time and Word Length  
  
# First we'll make a new column in our "Thomas" data with the mutate function, and use the stri\_length function to count the letters in all the words  
#But before we do that we need to remove all the special caracters and make the words Uppercase, so we later on can get some statistics from the 'MRC Psycholinguistic Database'  
data$clean\_words <- str\_replace\_all(data$Word, pattern = "[[:punct:]]", replacement = "")  
data$clean\_words <- toupper(data$clean\_words) #make words upper case   
data <- mutate(data, Word\_length = stri\_length(data$clean\_words))  
# Now we'll make a correlation test to see if the reactiontime and the word length are correlated, and we'll use spearmans method since we didn't get any normal distributions with our earlier transformations of the Thomas data.  
cor.test(data$Word\_length , data$Reaction.Time, method = "spearman")

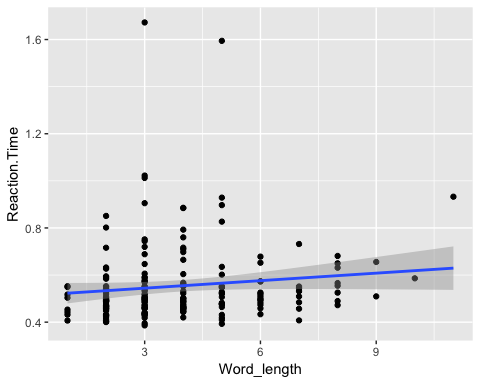
## Warning in cor.test.default(data$Word\_length, data$Reaction.Time, method =  
## "spearman"): Cannot compute exact p-value with ties

##   
## Spearman's rank correlation rho  
##   
## data: data$Word\_length and data$Reaction.Time  
## S = 834370, p-value = 0.007555  
## alternative hypothesis: true rho is not equal to 0  
## sample estimates:  
## rho   
## 0.1963504

# we calculate rho^2  
((0.1963504^2)\*100)

## [1] 3.855348

#as can be seen in the correlation test, p-value is .007 and therefore below .05, so the relation between variables is significant. Rho is .196 which is above 0.1 which means we have a small effect -> Rho^2 is 0.04, which means that word\_length captures 4% of the variance in Thomas's reaction times  
  
# We'll now show the relation with a scatterplot and and a linear model.  
ggplot(data, aes(x = Word\_length, y = Reaction.Time )) +   
 geom\_point()+  
 geom\_smooth(method = lm)



#as seen on the graph, we can see that by increasing word length we increase the reaction time

Report goes here: Word length was found to be positively correlated with   
the reaction time of the participant Thomas: r\_s = .20, p < .05

# Correlational study 2 - Reaction Time and Word Frequency  
  
#Here we make a word\_string of our variable clea\_words taht we can plug into the UWC psycholinguistic database  
word\_list <- data$clean\_words #turn column into a list  
word\_list <- unique(word\_list) #removes duplicates from text   
word\_string <- paste( unlist(word\_list), collapse=" ") #put a space in between all elements   
word\_string

## [1] "ONCE IN THE WINTERTIME WHEN SNOW WAS VERY DEEP A POOR BOY HAD TO GO OUT AND FETCH WOOD ON SLED AFTER HE GATHERED IT TOGETHER LOADED DID NOT WANT STRAIGHT HOME BECAUSE SO FROZEN BUT INSTEAD MAKE FIRE WARM HIMSELF LITTLE FIRST SCRAPED AWAY WHILE THUS CLEARING GROUND FOUND SMALL GOLDEN KEY NOW BELIEVED THAT WHERE THERE MUST ALSO BE LOCK DUG IRON CHEST IF ONLY FITS THOUGHT CERTAINLY ARE VALUABLE THINGS LOOKED NO KEYHOLE FINALLY ONE COULD SCARCELY SEEN TRIED FORTUNATELY FITTED THEN TURNED WE WAIT UNTIL HAS FINISHED LICKING OPENED LID SHALL FIND WHAT KIND OF WONDERFUL WERE"

#We copy the string into MRC database with brown-frequency ratings, manually copy it into excel, Use the text to columns function to seperate the data, and finish of saving the output as CSV.  
# then we read the CSV file  
freq <- read.csv("brown\_frequency.csv", header = F, sep = ";",na.strings = "-")  
  
  
colnames(freq) = c("clean\_words", "frequency") # we rename the columns in our new Dataframe  
merged\_df <-merge(data, freq, by = "clean\_words") #then merge the new predictor with the old dataset  
  
#And finish off runing a cor.test on the browns word frequency and Reactiontime  
cor.test(x= merged\_df$freq, y = merged\_df$Reaction.Time, method = "spearman")

## Warning in cor.test.default(x = merged\_df$freq, y = merged\_df  
## $Reaction.Time, : Cannot compute exact p-value with ties

##   
## Spearman's rank correlation rho  
##   
## data: merged\_df$freq and merged\_df$Reaction.Time  
## S = 1058000, p-value = 0.02923  
## alternative hypothesis: true rho is not equal to 0  
## sample estimates:  
## rho   
## -0.1644064

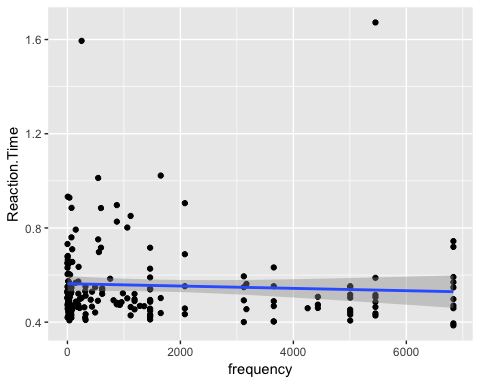
# we calculate rho^2  
(((-0.1644064)^2)\*100)

## [1] 2.702946

#it can be seen that p-value (.03) is below .05 which means that the relation is significant. Rho is negative (-0.16) which indicates that with higher frequencyscores follow smaller reactiontimes, furthermore it indicates that we see a little effect. Rho^2 is 2.7, which means that we have incaptured 2.7% of the variance  
  
# We'll now show the relation with a scatterplot and and a linear model.  
ggplot(merged\_df, aes(x = frequency, y = Reaction.Time )) +   
 geom\_point()+  
 geom\_smooth(method = lm)

## Warning: Removed 8 rows containing non-finite values (stat\_smooth).

## Warning: Removed 8 rows containing missing values (geom\_point).



#as seen on the graph the reaction time is negatively correlated with frequencyscore

Report goes here: Brown frequency was found to be negatively correlated with   
the reaction time of the participant Thomas: r\_s = -.16, p < .05

# Correlational study 3 - Reaction Time and Imaginability  
  
#We copy the string into MRC database with imaginability ratings, manually copy it into excel, Use the text to columns function to seperate the data, and finish of saving the output as CSV.  
# then we read the CSV file  
imaginability <- read.csv("imaginability.csv", header = F, sep = ";",na.strings = "-")  
colnames(imaginability) = c("clean\_words", "imaginability") #rename the columns  
merged\_df2 <-merge(data, imaginability, by = "clean\_words") #merge new predictor with the old dataset  
  
cor.test(x = merged\_df2 $imaginability, y = merged\_df2$Reaction.Time, method = "spearman")

## Warning in cor.test.default(x = merged\_df2$imaginability, y =  
## merged\_df2$Reaction.Time, : Cannot compute exact p-value with ties

##   
## Spearman's rank correlation rho  
##   
## data: merged\_df2$imaginability and merged\_df2$Reaction.Time  
## S = 610200, p-value = 0.8357  
## alternative hypothesis: true rho is not equal to 0  
## sample estimates:  
## rho   
## 0.01679731

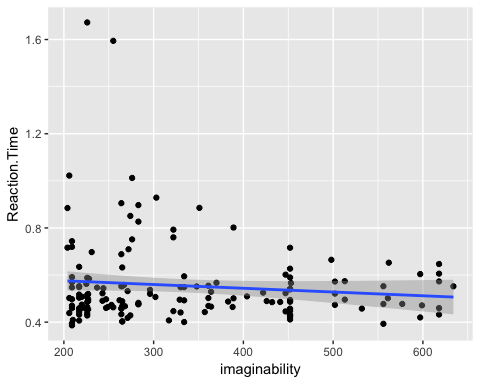
# we calculate rho^2  
(((0.01679731)^2)\*100)

## [1] 0.02821496

#0.03%  
  
#we can see that p-value is not significant and therefore we must acknowledge that we cannot observe any significant relation between imaginability and reaction time. Furthermore, rho is to small to indicate any effect. rho^2 squared tells us that we only capture 0.03% of the variance  
  
# We'll now show the insignificant relation with a scatterplot and and a linear model.  
ggplot(merged\_df2, aes(x = imaginability, y = Reaction.Time )) +   
 geom\_point()+  
 geom\_smooth(method = lm)

## Warning: Removed 29 rows containing non-finite values (stat\_smooth).

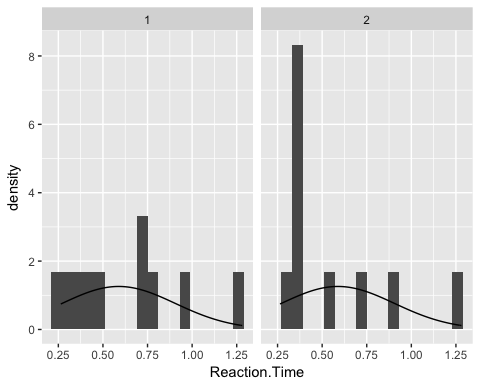
## Warning: Removed 29 rows containing missing values (geom\_point).



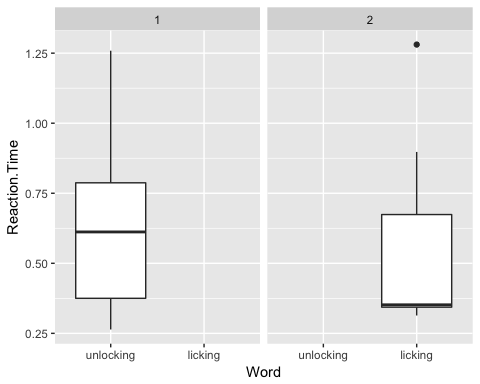
#although it's not significant a linear model seems to point in the direction that there is a negative correlation between imaginability and reactiontime

Report goes here: imaginability wasn't found to be significantly correlated with the reaction time of the participant Thomas: r\_s = .02, p > 0.05  
  
# 2) How does contextual expectations affect reading time?  
  
Conduct a contrastive analysis of the two conditions in your reading experiment. Single out reading time of the words that differ between the two versions of your text/story and compare the means using a t-test.

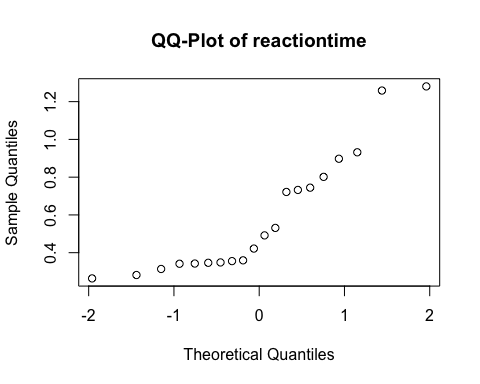
# code for the t-test study   
  
#Now we'll import all of our colelcted datafiles from the reading experiment (20 in total)  
#First we look trough our Working Directory and make list of filenames  
filenames <- list.files(pattern = "logfile\*")  
t\_test\_data = data.frame() #create a new empty df for this second part of the portfolio  
for (i in filenames) { #the we loop over the list of files  
 file = read.csv(i) #we import the current file  
 t\_test\_data = rbind(t\_test\_data, file[162,])   
 #and append the 162th line of the current file to our new df 't\_test\_data'using the rbind function  
}  
  
#we want to test if our data are normally distributed   
ggplot(t\_test\_data, aes(x = Reaction.Time))+  
 geom\_histogram(aes(y = ..density..), binwidth = 0.06)+  
 stat\_function(fun = dnorm, args = list(mean = mean(t\_test\_data$Reaction.Time), sd = sd(t\_test\_data$Reaction.Time)))+ facet\_wrap(~ Condition)



#according to the histogram it seems that we are approching normal distribution in condition 1 but condition 2 seems to be more positively skewed - Let's see it visualized  
ggplot(t\_test\_data, aes(x=Word,y = Reaction.Time)) + geom\_boxplot() + facet\_wrap(~Condition)



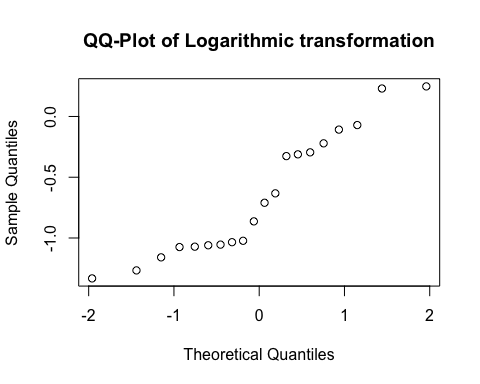
#we'll try to draw qqplots to see it in another way  
qqnorm(t\_test\_data$Reaction.Time, main = "QQ-Plot of reactiontime")



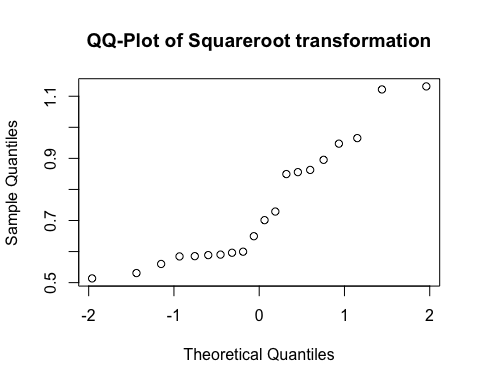
#Now let's get some numbers using the stat.desc function  
round(stat.desc(t\_test\_data$Reaction.Time,basic = FALSE, norm = TRUE),3)

## median mean SE.mean CI.mean.0.95 var   
## 0.457 0.588 0.071 0.148 0.101   
## std.dev coef.var skewness skew.2SE kurtosis   
## 0.317 0.539 0.863 0.843 -0.503   
## kurt.2SE normtest.W normtest.p   
## -0.253 0.849 0.005

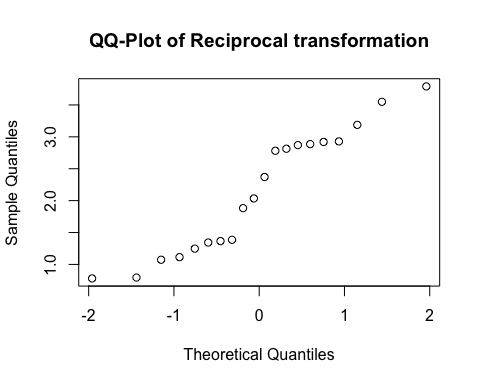
#normtest.p is significant (<.05), which means that the scores are significantly different from a normal distribution, but skew2SE and kurt2SE are between 1 and -1, which is relatively small.  
# we'll try to use some transformations to bring about a normaldistribution  
  
t\_test\_data <- mutate(t\_test\_data, logRT = log(Reaction.Time), sqrtRT = sqrt(Reaction.Time), RT = 1/(Reaction.Time))  
  
# we'll run a few QQplots to check if the transformations worked in our favour  
qqnorm(t\_test\_data$logRT, main = "QQ-Plot of Logarithmic transformation")



qqnorm(t\_test\_data$sqrtRT, main = "QQ-Plot of Squareroot transformation")



qqnorm(t\_test\_data$RT, main = "QQ-Plot of Reciprocal transformation")



# We'll include some normal distribution statistics  
round(stat.desc(t\_test\_data$logRT,basic = FALSE, norm = TRUE),3)

## median mean SE.mean CI.mean.0.95 var   
## -0.787 -0.657 0.113 0.237 0.256   
## std.dev coef.var skewness skew.2SE kurtosis   
## 0.506 -0.771 0.372 0.363 -1.372   
## kurt.2SE normtest.W normtest.p   
## -0.691 0.905 0.051

round(stat.desc(t\_test\_data$sqrtRT,basic = FALSE, norm = TRUE),3)

## median mean SE.mean CI.mean.0.95 var   
## 0.675 0.743 0.044 0.092 0.038   
## std.dev coef.var skewness skew.2SE kurtosis   
## 0.196 0.264 0.605 0.591 -1.030   
## kurt.2SE normtest.W normtest.p   
## -0.519 0.883 0.020

round(stat.desc(t\_test\_data$RT,basic = FALSE, norm = TRUE),3)

## median mean SE.mean CI.mean.0.95 var   
## 2.202 2.156 0.215 0.450 0.925   
## std.dev coef.var skewness skew.2SE kurtosis   
## 0.962 0.446 0.033 0.032 -1.535   
## kurt.2SE normtest.W normtest.p   
## -0.773 0.918 0.089

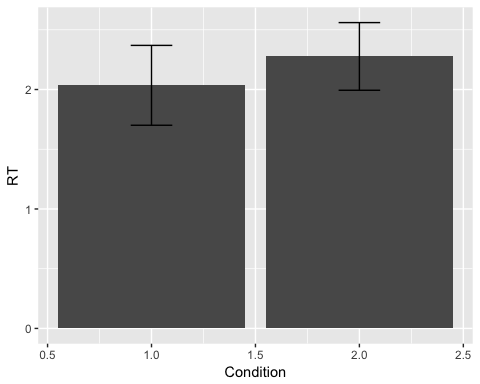
# Both the Logarithmic and Reciprocal transformation worked. the p-value (>.05) is not significantly different from a normaldistribution - we'll go ahead with the reciprocal transformed data, sice the show the higher P-value (0.09)

T-testing

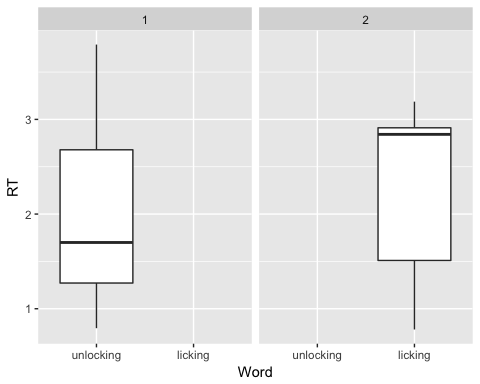
#We got the t\_test\_data normally distributed with the reciprocal transformation  
#Now we'll run the test  
t.test(RT ~ Condition , data = t\_test\_data)

##   
## Welch Two Sample t-test  
##   
## data: RT by Condition  
## t = -0.55207, df = 17.527, p-value = 0.5879  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## -1.1640788 0.6803468  
## sample estimates:  
## mean in group 1 mean in group 2   
## 2.035059 2.276925

#here we see that the p-value is over .05, which means that we can't tell if there's any significant difference in means between the two conditions.  
#This can be visualized in a barplot  
ggplot(t\_test\_data, aes( x = Condition ,y = RT ))+  
 geom\_bar(stat = "summary", fun.y= mean)+  
 geom\_errorbar(stat = "summary", fun.data = mean\_se, width = 0.2)



#in the bar chart we can see that error bars overlap, which means we can't significantly tell their means apart.  
ggplot(t\_test\_data, aes(x=Word,y = RT)) + geom\_boxplot() + facet\_wrap(~Condition)



#Futhermore the t-test tells us that the confidence interval crosses 0, with -1.16 and 0.68, which means we can't tell if there is a negative or positive relationship between the two conditions.  
  
# Calculating the Standard error  
by(t\_test\_data$RT, t\_test\_data$Condition, stat.desc, basic = FALSE, norm = TRUE)

## t\_test\_data$Condition: 1  
## median mean SE.mean CI.mean.0.95 var   
## 1.6998600 2.0350586 0.3342605 0.7561497 1.1173005   
## std.dev coef.var skewness skew.2SE kurtosis   
## 1.0570244 0.5194073 0.4586754 0.3338040 -1.4686625   
## kurt.2SE normtest.W normtest.p   
## -0.5503706 0.9041352 0.2430838   
## --------------------------------------------------------   
## t\_test\_data$Condition: 2  
## median mean SE.mean CI.mean.0.95 var   
## 2.84201576 2.27692459 0.28321073 0.64066717 0.80208315   
## std.dev coef.var skewness skew.2SE kurtosis   
## 0.89559095 0.39333360 -0.52934533 -0.38523454 -1.62020035   
## kurt.2SE normtest.W normtest.p   
## -0.60715827 0.81937988 0.02490819

#Calculating the Rho coefficient  
t <- -0.55207  
df <- 17.527  
r <- sqrt((t^2)/((t^2)+df))  
r

## [1] 0.1307365

#and effectsize  
(r^2)\*100

## [1] 1.709203

# r= 0.1307365, which means we see a small effect and r^2 = 1.709203, which means that we can explain 1.7% of the variance better than a mean model

Report goes here:   
“On average, participants did not experience significantly greater readingtime from an unexpected word (M = 2.28, SE = 0.28) than from an expected words (M = 2.04, SE = 0.33),t(17.53) = -0.55 , p > .05, r=.13   
-> The degrees of freedom is not 18, because it was adjusted using the Welsch-test