

# Homework 3

11/21/2020

1. Run the following command on the R command line to download the data-set (careful when you copy and paste the link!):

```
download.file("https://raw.githubusercontent.com/vasishth/Freq_CogSci/master/data/grodnergibsonE1crit.txt",destfile = "dat.txt")
```

The data will be in your working directory with the name “dat.txt”. Then use the appropriate command to load the data into R. Remember that you need to be in the correct working directory for R to find the file.

```
data <- read.table("dat.txt", header=T)
```

2. Find out the number of columns, the name of the columns and the number of rows using the relevant R functions.

```
ncol(data)
```

```
## [1] 4
```

```
colnames(data)
```

```
## [1] "subject" "item" "condition" "rawRT"
```

```
nrow(data)
```

```
## [1] 672
```

3. What’s the class of the column “subject”?

```
class(data$subject) # integer
```

```
## [1] "integer"
```

4. Convert the columns “subject” and “item” to factor.

```
data$subject <- as.factor(data$subject)
data$item <- as.factor(data$item)
```

5. How many subjects are there? and items?

```
length(unique(data$subject))
```

```
## [1] 42
```

```
length(unique(data$item))
```

```
## [1] 16
```

6. How many instances of each condition does each subject see? Answer: There are only two conditions, and there are 16 items, assuming that the experiment is balanced, there would be 8 items per condition. We can check this with xtabs:

```
data$condition <- as.factor(data$condition)
xtabs(~ condition + subject, data=data)
```

```
##           subject
## condition 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26
##  objgap   8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
##  subjgap  8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
##           subject
## condition 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42
##  objgap    8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
##  subjgap    8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
```

7. Compute the mean reading time and sd in objgap and in subjgap.

```
tapply(data$rawRT, data$condition, mean)
```

```
##  objgap  subjgap
## 471.3601 369.0744
```

```
tapply(data$rawRT, data$condition, sd)
```

```
##  objgap  subjgap
## 464.4060 177.2674
```

8. Remove all reading times higher than 5000 ms and smaller than 200 ms. Make sure that you did this correctly by computing the minimum and maximum values of the reading times.

```
data <- subset(data, rawRT>200 & rawRT<5000)
min(data$rawRT)
```

```
## [1] 201
```

```
max(data$rawRT)
```

```
## [1] 3365
```

9. Log-transform the rawRT observations (in a new column called logRT).

```
data$logRT <- log(data$rawRT)
```

10. Created a new column that contains -1 for subjgap observations, and 1 for objgap observations.

```
data$contrasts <- ifelse(data$condition=="subjgap", -1,1)
```

10. Create two new dataframes, one for the condition objgap and another one for the condition subjgap.

```
subjgap <- droplevels(subset(data, condition=="subjgap"))  
objgap <- droplevels(subset(data, condition=="objgap"))
```

11. Create a user-defined function that takes as an argument a vector of numbers. Call this argument "numbers". The function has to calculate and return the mean of the numbers. DO NOT use the function mean(). Instead, you should add up all of the numbers of the vector and divide this by the length of the vector. You can use the functions sum() and length().

```
compute.mean <- function(numbers){  
  length <- length(numbers)  
  mean <- sum(numbers)/length  
  return(mean)  
}
```

11. Use your function to calculate the mean of the rawRT column in the dataframes objgap and subjgap that you created in question 9.

```
compute.mean(numbers=objgap$rawRT)
```

```
## [1] 484.4771
```

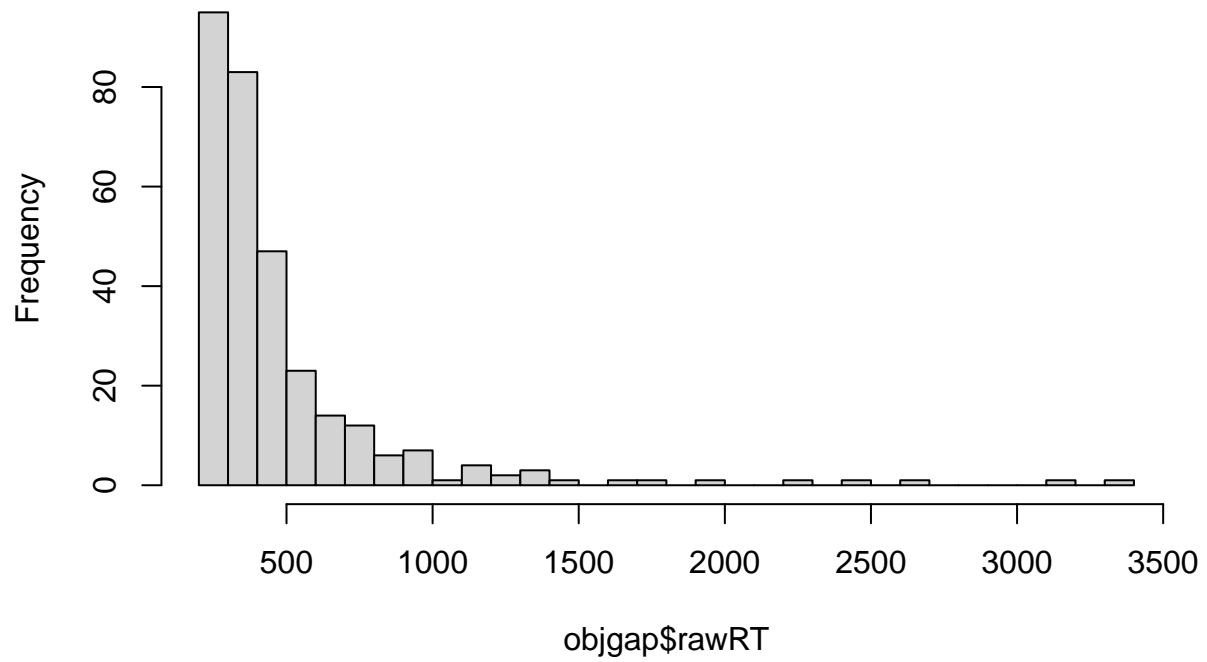
```
compute.mean(numbers=subjgap$rawRT)
```

```
## [1] 385.6861
```

12. Plot a histogram of the rawRT in objgap and in subjgap.

```
hist(objgap$rawRT,breaks=30)
```

**Histogram of objgap\$rawRT**



```
hist(subjgap$rawRT,breaks=30)
```

**Histogram of subjgap\$rawRT**

