IST772– Problem Set 7

Sathish Kumar Rajendiran

Attribution statement: 1. I did this homework by myself, with help from the book and the professor.

Chapter 7, Exercise 3

Run cor.test() on the correlation between "speed" and "dist" in the cars data set (type "? cars" to see the documentation) and interpret the results. (1 pt) Note that you will have to use the "$" accessor to get at each of the two variables (like this: cars$speed, but without the backslash, needed since the dollar sign is a special character in R markdown). Make sure that you interpret both the confidence interval and the p-value that is generated by cor.test(). (1 pt)

1) cor.test:

- cor.test() - a R procedure for the null hypothesis test on significance based on correlation between the variables.

2) Correlation test result:

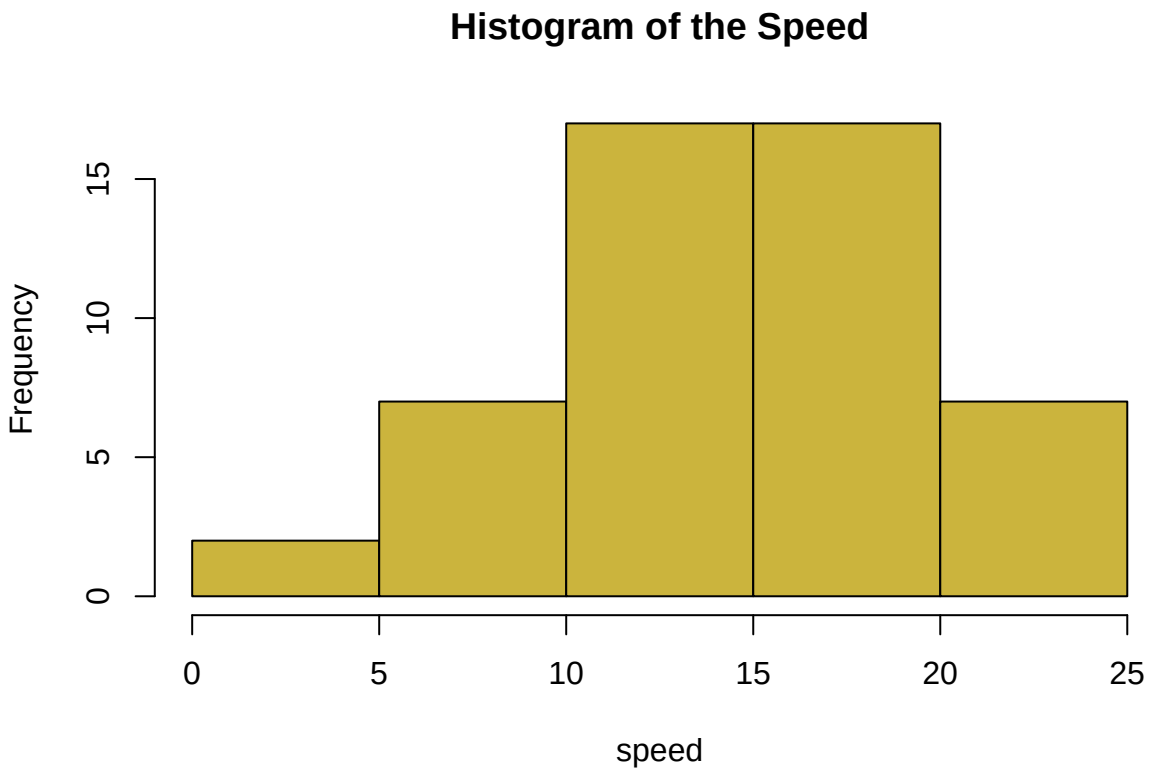
- cor.test(speed,dist) - from n=50, correlation test returns 3 sections
  - t-value : test statistic is a transformed version of the correlation coefficient.This test has yielded a stronger t-value of 9.464.Confirming the test as significant
  - df = 48 : degrees of freedom states that 48 out of 50 observations are free to vary in this statiscal test leaving 2 df for each variable.
  - p-value = 0.00000000000149; suggests that there is a 0.00000000000149 chance of observing an absolute value of "t" this high.
  - based on the conventional $p < 0.05$ threshold for alpha to evaluate the Null hypothesis test , we can reject the null hypothesis.
  - 95% confidence interval between 0.6816422 (2.5% lower bound) and 0.8862036 (97.5%) suggests that the population value of rho may lie between these values if the test is repeated 95 out of 100 times. In addition, the values are higher than 0. further supporting our decision to reject null hypothesis.
  - correlation coefficient (r) is 0.8068949 is closer to +1 suggesting stronger correlation between the variables.

Please find more details from the R code below,

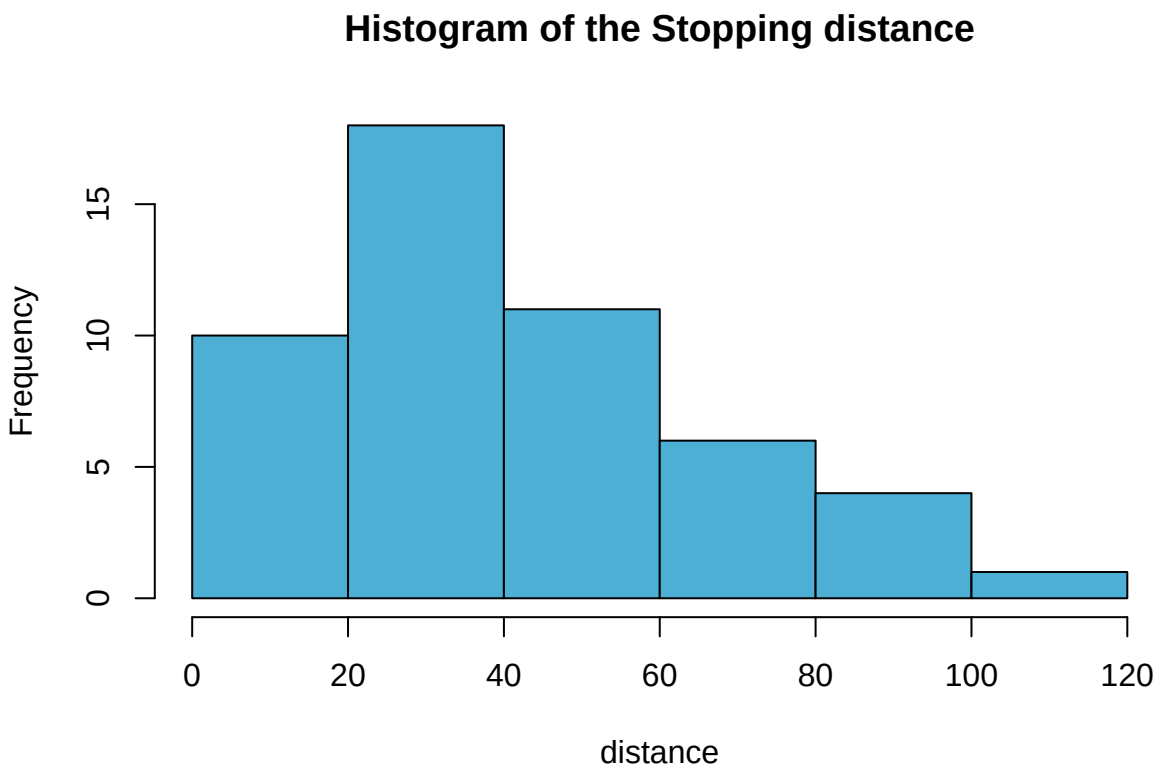
```r
summary(cars)
```

```
##      speed           dist       
##  Min.   : 4.0   Min.   :  2.00  
##  1st Qu.:12.0   1st Qu.: 26.00  
##  Median :15.0   Median : 36.00  
##  Mean   :15.4   Mean   : 42.98  
##  3rd Qu.:19.0   3rd Qu.: 56.00  
##  Max.   :25.0   Max.   :120.00
```

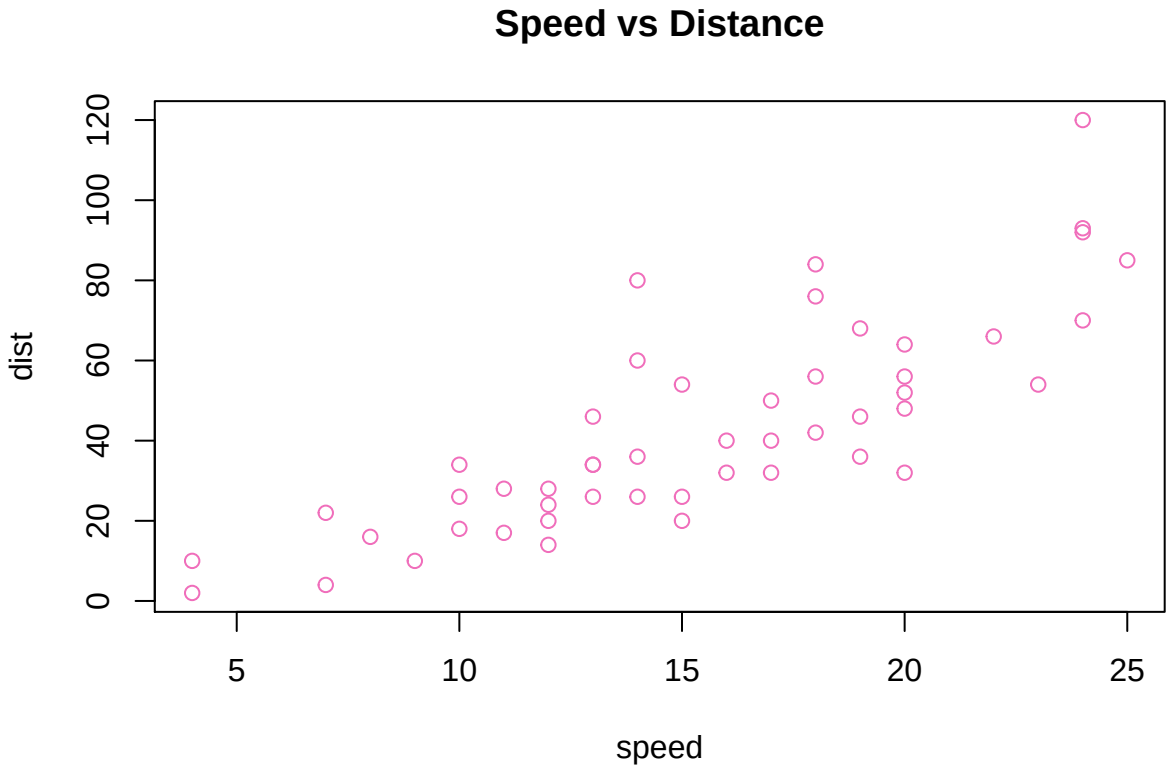
```r
#dim(cars) # [1] 50  2
# View(cars)

#assign variables
speed <- cars$speed  # assign all oservations of from speed variable from cars dataset to "speed"
dist <- cars$dist # assign all oservations of from dist variable from cars dataset to "dist"

# Plot Histogram of the Speed
hist(speed
     ,main="Histogram of the Speed"
     ,xlab="speed"
     ,col="#CBB43D")
```




```r
# Plot Histogram of the Stopping distance
hist(dist
     ,main="Histogram of the Stopping distance"
     ,xlab="distance"
     ,col="#4DAFD4")
```




```
# Scatter Plot of the Speed vs Distance
plot(speed,dist,col="#F06EBB",main="Speed vs Distance")
```



```r
options(scipen=999)  # turn-off scientific notation like 1e+48

# Correlation between Speed and distance from cars dataset
cor(speed,dist)
```

```
## [1] 0.8068949
```

```r
# preform correlation test on Speed and Distance variables
cor.test(speed,dist)
```

```
##
##  Pearson's product-moment correlation
##
## data:  speed and dist
## t = 9.464, df = 48, p-value = 0.00000000000149
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.6816422 0.8862036
## sample estimates:
##       cor
## 0.8068949
```

Chapter 7, Exercise 4

Below is a copy of the bfCorTest() custom function presented in this chapter; you can instead use the correlationBF function from the BayesFactor library. Conduct a Bayesian analysis of the correlation between "speed" and "dist" in the cars data set. (1 pt) Report the results. (1 pt)

1) Bayesian test:

- bfCorTest(speed,dist) - custom procedure returns the Bayesian analysis of correlation between "speed" and "dist" variablesfrom cars dataset with 10,000 posterior samples.
- This custom function returns 95% HDI for the correlation coefficients from the posterior population distribution.
- In addition,it also returns bayes factor; i.e the odds ratio that shows the odds in favor of the alternative hypothesis that the population correlation cofficient "rho" is not equal to 0.
- In addition, we have also run correlationBF(speed,dist) - R procedure to find the Bayes factor analysis.

2) Bayesian test Result:

- bfCorTest(speed,dist) returns 3 differnt secttions
- Empirical mean and standard deviation for each variable, plus standard error of the mean
- Quantiles for each variable representing 95% HDI ranges from 0.6148 to 0.9607. With coefficient almost equals to 1, suggesting stronger association between the variables with 10,000 posterior population distribution

- rhoNot0 : 3486525337 ±0.01%; Bayes factor strongly suggests a strong evidence in favor of alternative hypothesis.
- R procedere "correlationBF" als returns 95% HDI with 10,000 posterior population distribution reurns population coeffient (rho) between 0.6251 and 0.8595 ,suggesting stronger association between the variables
- plot below suggests the correlation coefficient variance across 10,000 distributions.Almost depicting a bell-shaped curve between 0.6251 and 0.8595 HDI intervals.
- Therefore, in this research situation, the Bayes factor and the null hypothesis concur with other.

Please find more details from the R code below,

```
library("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
##
## Type BFManual() to open the manual.
## ************
```

```
bfCorTest <- function (x,y) # Get r from BayesFactor
{
  zx <- scale(x)  # standardize X
  zy <- scale(y)  # standardize Y
  # rhoNot0 is meant to refer alterntive hypothesis that population correlation cofficient "rho" is not
  zData <- data.frame(x=zx,rhoNot0=zy) # put in a data frame
  bfOut <- generalTestBF(x ~ rhoNot0, data=zData) # linear coefficient ;

  mcmcOut <- posterior(bfOut,iterations=10000) # posterior samples
  print(summary(mcmcOut[,"rhoNot0"])) # Show the HDI for r
  return(bfOut) # Return Bayes factor object
}

bfCorTest(speed,dist)
```

```
##
## Iterations = 1:10000
## Thinning interval = 1
## Number of chains = 1
## Sample size per chain = 10000
##
## 1. Empirical mean and standard deviation for each variable,
##    plus standard error of the mean:
##
##           Mean             SD       Naive SE Time-series SE
```

```
##      0.7865146      0.0893622      0.0008936      0.0009138
##
## 2. Quantiles for each variable:
##
##   2.5%    25%    50%    75%  97.5%
## 0.6120 0.7274 0.7875 0.8458 0.9587
```

```
## Bayes factor analysis
## --------------
## [1] rhoNot0 : 3486525337 ±0.01%
##
## Against denominator:
##   Intercept only
## ---
## Bayes factor type: BFlinearModel, JZS
```

```
# Bayesian approach using BayesFactor package
bf <- correlationBF(speed, dist)
bf
```

```
## Bayes factor analysis
## --------------
## [1] Alt., r=0.333 : 1744357848 ±0%
##
## Against denominator:
##   Null, rho = 0
## ---
## Bayes factor type: BFcorrelation, Jeffreys-beta*
```

```
# Posterior distribution using BayesFactor package
bfPost <- posterior(bf, iterations = 10000)
```

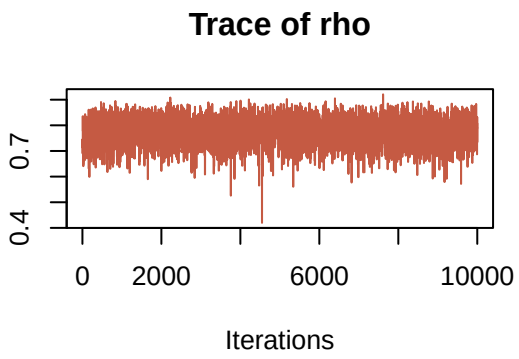
```
## Independent-candidate M-H acceptance rate: 60%
```

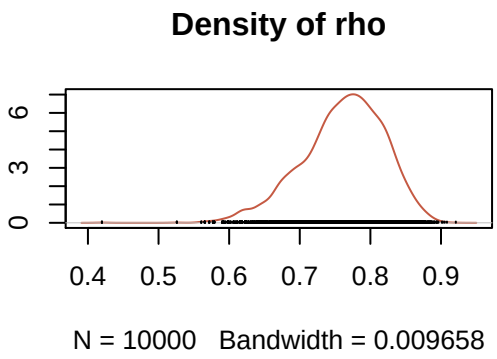
```
# summary(bfPost)
print(summary(bfPost)) # Show the HDI for r
```

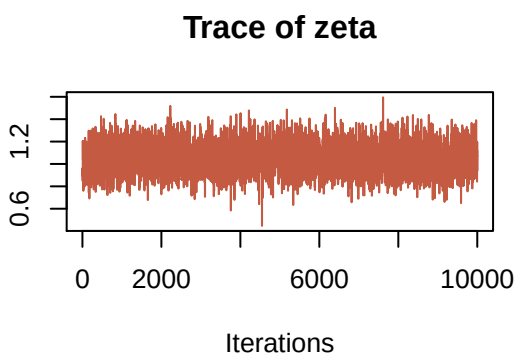
```
##
## Iterations = 1:10000
## Thinning interval = 1
## Number of chains = 1
## Sample size per chain = 10000
##
## 1. Empirical mean and standard deviation for each variable,
##    plus standard error of the mean:
##
##        Mean      SD  Naive SE Time-series SE
## rho  0.7612 0.05813 0.0005813       0.001348
## zeta 1.0135 0.13878 0.0013878       0.002972
##
## 2. Quantiles for each variable:
##
```

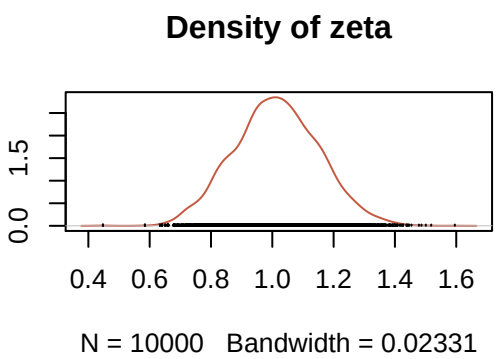
```
##          2.5%    25%    50%    75%  97.5%
## rho  0.6309 0.7261 0.7664 0.8031 0.8588
## zeta 0.7430 0.9204 1.0114 1.1073 1.2888
```

```
plot(bfPost,col="#C55A43")
```









Chapter 7, Exercise 8

The data set called UCBAdmissions (see "? UCBAdmissions" for documentation) contains data on applicants to graduate school at Berkeley for the six largest departments in 1973 classified by admission and sex. You can access the data for the first department like this: UCBAdmissions[, , 1]. Make sure you put two commas before the 1: this is a three dimensional contingency table that we are subsetting down to two dimensions. Run chisq.test() on the subset of the data set for department 1 (1 pt) and make sense of the results. (1 pt)

1) chisq.test:

- chi-square test is official significance test of null hypothesis; on the subset of UCBAdmissions[, , 1] dataset suggesting 2 * 2 table of number of students admitted/rejected for Department A
- chisq.test(deptA) - procedure returns, X-squared = 16.372, df = 1, p-value = 0.00005205
- null hypothesis: there is independence between gender and admission status

2) chisq.test() Result:

- chisq.test(deptA) has returned a contingency table with

  - 512 male admitted to dept A
  - 313 male rejected to dept A
  - 89 female admitted to dept A
  - 19 female rejected to dept A
- In addition, it returned values as X-squared = 16.372, df = 1, p-value = 0.00005205
- based on the conventional $p < 0.05$ threshold for alpha to evaluate the Null hypothesis test; with p-value = 0.00005205, we can reject the null hypothesis of independence between gender and admission status. These two variables are not independent and by inspecting 2*2 contingency table , we can see that the percentage female getting admitted (82%) vs male (62%).
- Bigger the chi-square value (greater than 1) , stronger the evidence is.

Please find more details from the R code below,

```
options(scipen=999)  # turn-off scientific notation like 1e+48
# ?UCBAdmissions
# summary(UCBAdmissions)
# View(UCBAdmissions)
deptA <- UCBAdmissions[ , , 1]
cat("\n.........................................\n")
```

```
##
## .........................................
```

```
deptA
```

```
##           Gender
## Admit      Male Female
##   Admitted  512     89
##   Rejected  313     19
```

```
cat(".........................................\n")
```

```
## .........................................
```

```
# chisq.test()
chisq.test(deptA,correct=FALSE) # correct=FALSE suppresses Yates correction
```

```
##
##  Pearson's Chi-squared test
##
## data:  deptA
## X-squared = 17.248, df = 1, p-value = 0.0000328
```

```
512/(512+313)
```

```
## [1] 0.6206061
```

```r
89/(89+19)
```

```
## [1] 0.8240741
```

Chapter 7, Exercise 9

Use contingencyTableBF() to conduct a Bayes factor analysis on the UCB admissions data for department 1. (1 pt) Report and interpret the Bayes factor. (1 pt)

1) contingencyTableBF:

- contingencyTableBF() - provides Bayesian factor analysis on the chi-square test. It can optionally generate posterior distribution for the frequencies (or proportions) in the cells of the contingency table. with contingency tables, the strategy used to collect the data affects the choice of prior probabilities.Help evaluate alternate hypothesis

```
#            Admit Admitted Rejected
#      Gender
#      Male              512      313
#      Female             89       19
```

2) contingencyTableBF Result:

- contingencyTableBF() with posterior=FALSE - leaves posterion population distribution as optional
- sample type "poisson" suggests there is no specific target for the number of observations
- 1111.64 ±0% against denominator suggests that the Bayes factor is in favor of alternate hypothesis.Becuase its greater than 3:1, we can treat it as positive evidence in favor of non-independence.Therefore, in this research situation, the Bayes factor and the null hypothesis concur with other.

Please find more details from the R code below,

```r
# contingencyTableBF - Bayes factor analysis:
library(BayesFactor)
deptAM <- ftable(deptA,row.vars = 2,col.vars = "Admit")
ctBFout <- contingencyTableBF(deptAM, sampleType="poisson",posterior=FALSE) # sample type "poisson" sug
ctBFout
```

```
## Bayes factor analysis
## --------------
## [1] Non-indep. (a=1) : 1111.64 ±0%
##
## Against denominator:
##   Null, independence, a = 1
## ---
## Bayes factor type: BFcontingencyTable, poisson
```

```
deptAM
```

```
##        Admit Admitted Rejected
## Gender
## Male              512      313
## Female             89       19
```

Chapter 7, Exercise 10

Using the UCBAdmissions data for department 1, run contingencyTableBF() with posterior sampling. (1 pt) Use the results to calculate a 95% HDI of the difference in proportions between the columns. (1 pt for extracting proportions, 1 pt for HDI, 1 pt for interpretation)

1) contingencyTableBF with posterior sampling:

- contingencyTableBF() - provides Bayesian factor analysis on the chi-square test. with posterior=TRUE and iterations=10000, it suggests posterior distribution population of chi-square analysis to Help evaluate alternate hypothesis.
- contingencyTableBF(deptA, sampleType="poisson", posterior=TRUE, iterations=10000) is assigned to variable ctMCMCOut

2) Extract Proportions:

- ctMCMCOut is further broken into two rows as below,
  - Row1 <- ctMCMCOut[,"lambda[1,1]"] / ctMCMCOut[,"lambda[1,2]"] # Number of male admitted vs nNumber of male rejected
  - Row2 <- ctMCMCOut[,"lambda[2,1]"] / ctMCMCOut[,"lambda[2,2]"] # Number of female admitted vs nNumber of female rejected

3) HDI of Difference:

- 95% HDI differences in population correlation coefficient is taken by calculating the differences of 10,000 values from Row1 and Row2 and assigned to variable called Diff * HDI differnce low is at 2.5% is -20.26309 # low end HDI differences in proportion * HDI differnce low is at 97.5% is -4.54917 # high end HDI differences in proportion
- HDI differnces for all 10,000 samples as below # 2.5% 25% 50% 75% 97.5% # lambda[1,1] 466.88 495.64 510.78 526.1 556.80 # lambda[2,1] 72.11 83.06 89.16 95.7 108.56 # lambda[1,2] 278.05 300.80 312.02 324.1 346.88 # lambda[2,2] 12.22 16.77 19.64 22.8 29.58

4) Result summary:

- In this contingencyTableBF() test, we added couple of paramters; posterior=TRUE and iterations=10000 to enable the procedure to sample from the posterior distribution and the later asks for 10000 samples.
- 10,000 samples contingencyTableBF() returns HDI values for each variable association

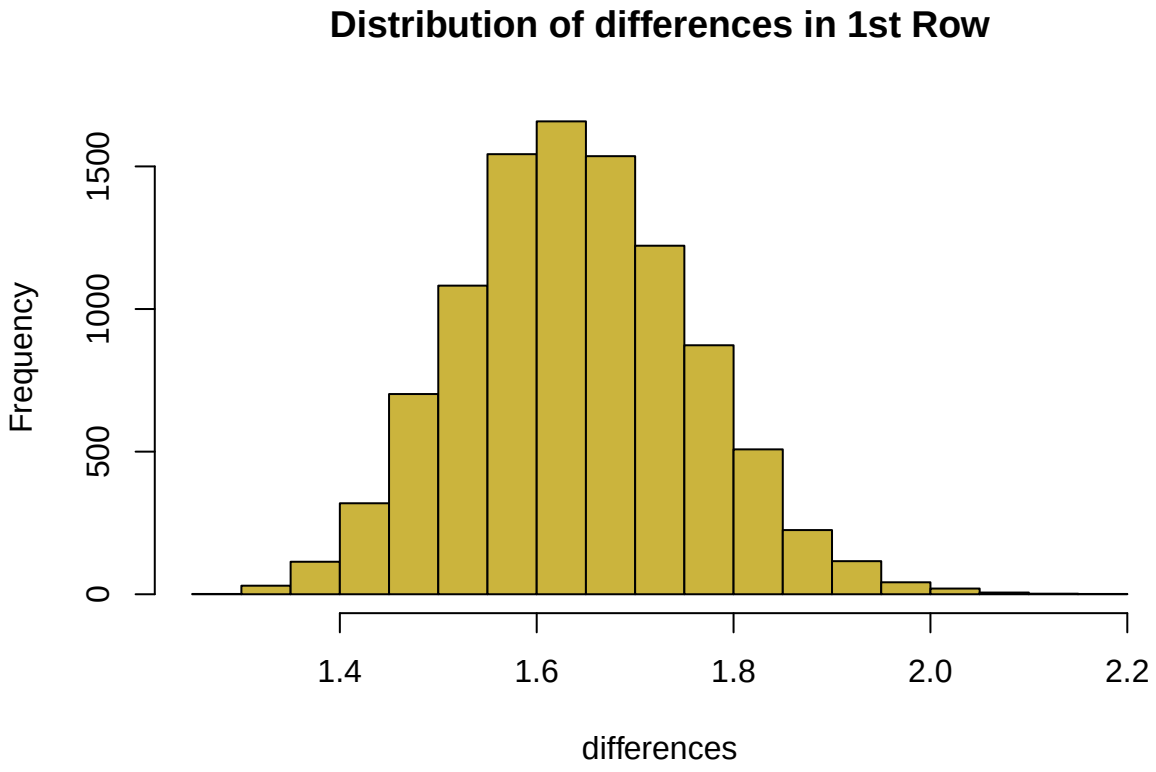
- Next step is to extract proportions of admission status by gender. respoective plots shows the distribution
  - Row1 <- ctMCMCOut[,“lambda[1,1]”] / ctMCMCOut[,“lambda[1,2]”] # Number of male admitted vs Number of female admitted
  - Row2 <- ctMCMCOut[,“lambda[2,1]”] / ctMCMCOut[,“lambda[2,2]”] # Number of male rejected vs nNumber of female rejected
- finally, HDI differnce plots the range between 2.5% and 97.5% across two rows proportions * HDI differnce low is at 2.5% is -20.26309 # low end HDI differences in proportion * HDI differnce low is at 97.5% is -4.54917 # high end HDI differences in proportion
- mean value is -3.085007
- we have analzed both by finding means of correlation cofficients and tests of independence using contingency tables.

Please find more details from the R code below,

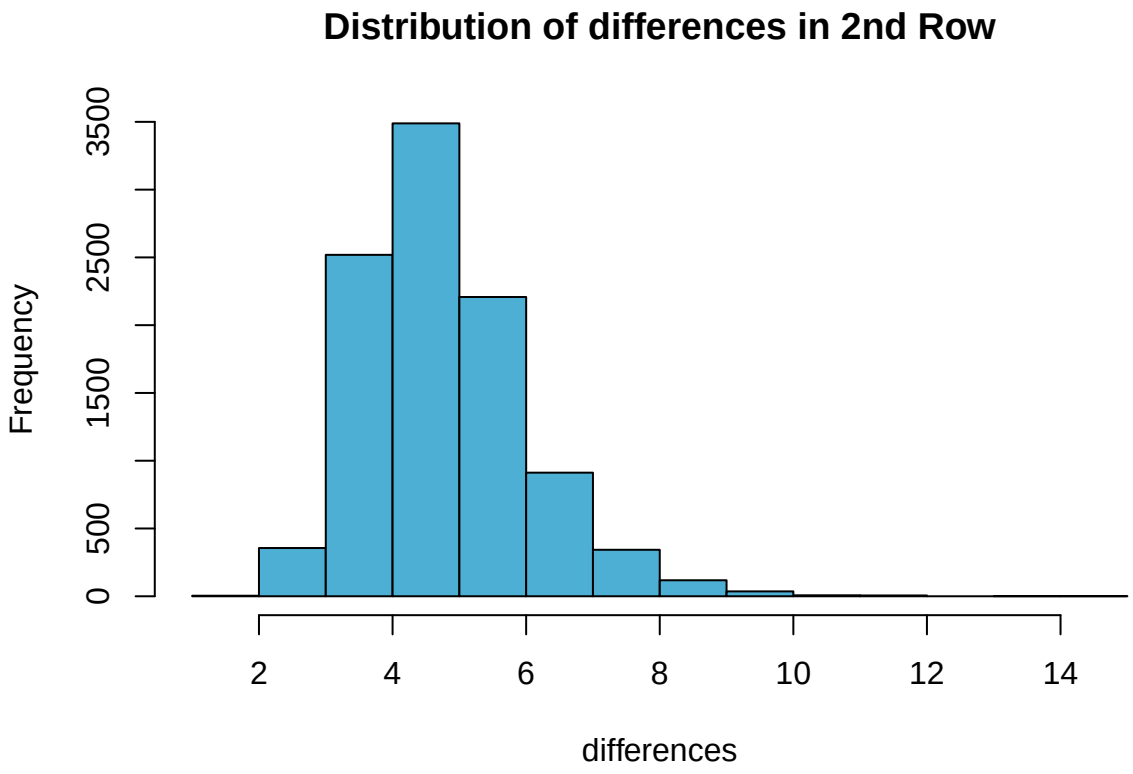
```
# contingencyTableBF
library(BayesFactor)
ctMCMCOut <- contingencyTableBF(deptAM, sampleType="poisson", posterior=TRUE, iterations=10000)
summary(ctMCMCOut)
```

```
##
## Iterations = 1:10000
## Thinning interval = 1
## Number of chains = 1
## Sample size per chain = 10000
##
## 1. Empirical mean and standard deviation for each variable,
##    plus standard error of the mean:
##
##               Mean     SD Naive SE Time-series SE
## lambda[1,1] 511.00 22.496  0.22496        0.22920
## lambda[2,1]  89.76  9.565  0.09565        0.09565
## lambda[1,2] 312.51 17.715  0.17715        0.18106
## lambda[2,2]  19.85  4.419  0.04419        0.04686
##
## 2. Quantiles for each variable:
##
##               2.5%    25%    50%    75%  97.5%
## lambda[1,1] 467.49 495.81 510.69 526.26 555.67
## lambda[2,1]  72.10  83.17  89.41  96.02 109.69
## lambda[1,2] 278.77 300.46 312.16 324.20 348.19
## lambda[2,2]  12.24  16.71  19.53  22.60  29.47
```

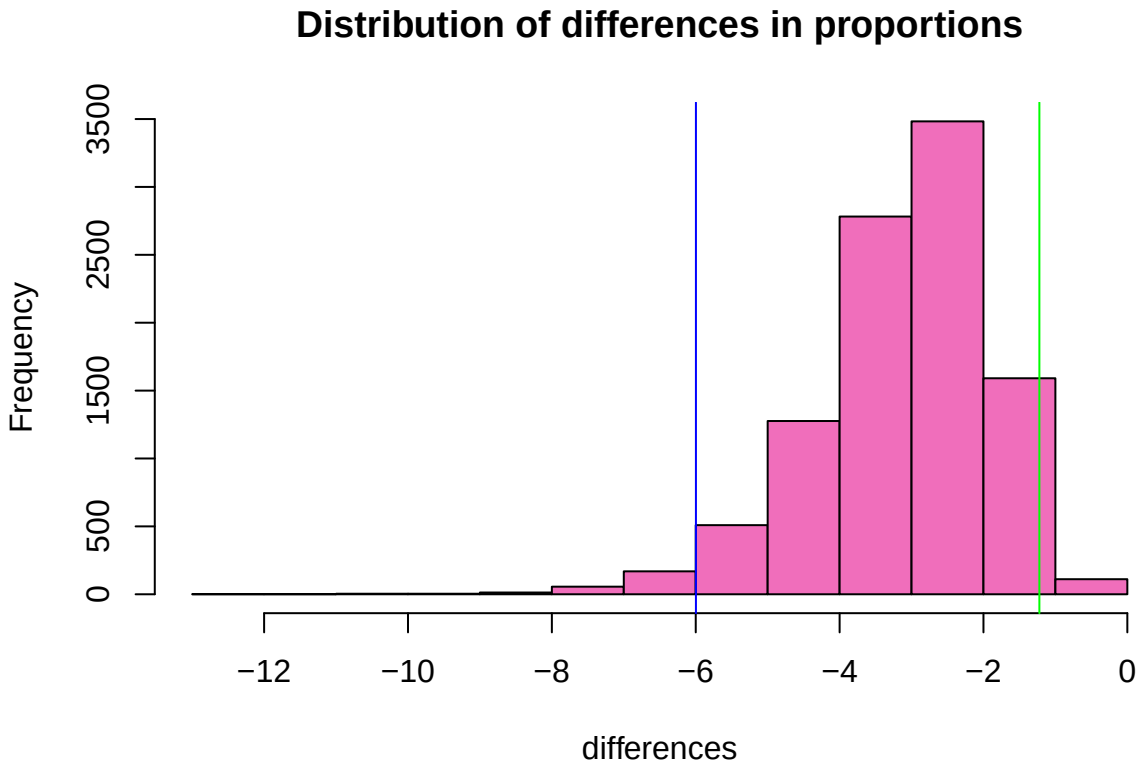
```
# Extract Proportions
Row1 <- ctMCMCOut[,"lambda[1,1]"] / ctMCMCOut[,"lambda[1,2]"]
#Row1
hist(Row1
     ,main="Distribution of differences in 1st Row"
     ,xlab="differences"
     ,col="#CBB43D")
```




```
# Extract Proportions
Row2 <- ctMCMCOut[,"lambda[2,1]"] / ctMCMCOut[,"lambda[2,2]"]
hist(Row2
     ,main="Distribution of differences in 2nd Row"
     ,xlab="differences"
     ,col="#4DAFD4")
```




```
# HDI of Difference

Diff<-Row1-Row2

hist(Diff
     ,main="Distribution of differences in proportions"
     ,xlab="differences"
     ,col="#F06EBB")
abline(v=quantile(Diff,0.025),col="blue")  # low end HDI differences in proportion
abline(v=quantile(Diff,0.975),col="green") # high end HDI differences in proportion
```




```
quantile(Diff,0.975)
```

```
##     97.5%
## -1.222916
```

```
quantile(Diff,0.025)
```

```
##      2.5%
## -5.997246
```

```
mean(Diff)
```

```
## [1] -3.115668
```