Homework 1

Mary Silva, Wyara Moura Silva, and Peter Trubey

10/16/2017

Chapter 1

Question 6

```r
library(htmltab)
```

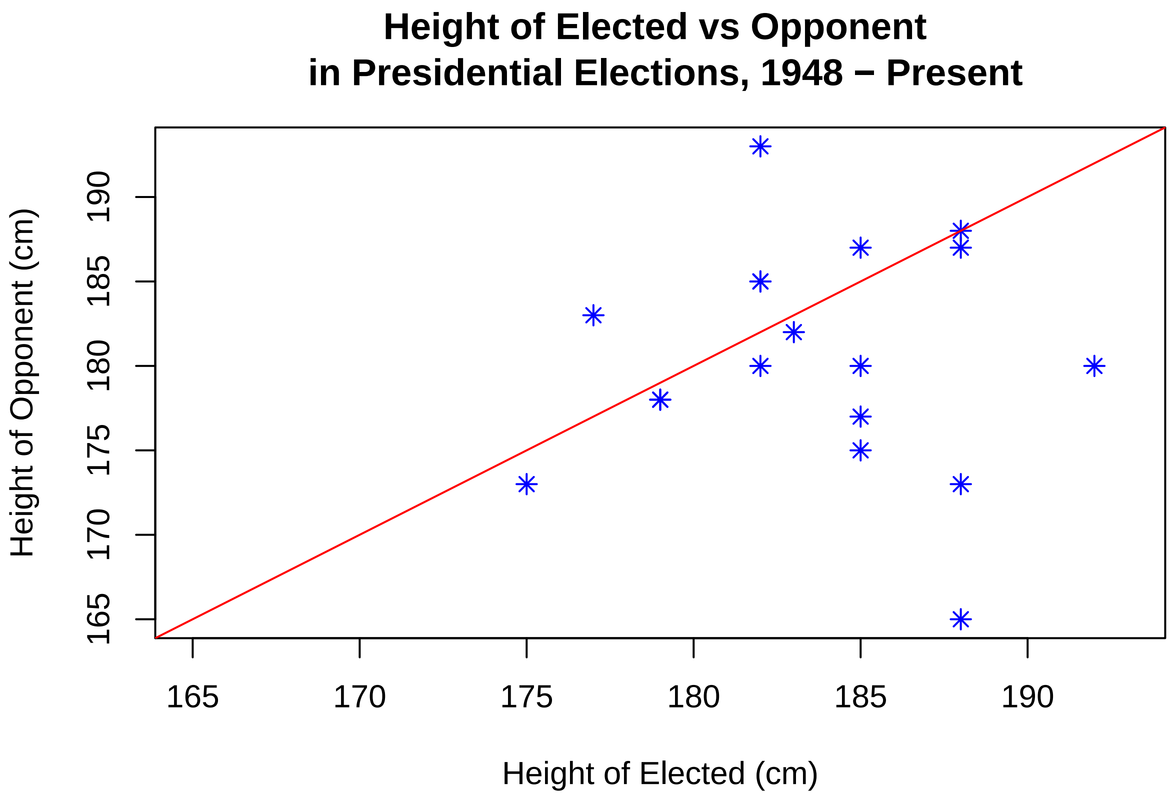
```
## Warning: package 'htmltab' was built under R version 3.2.5
```

```r
table_url = paste(
  'http://en.wikipedia.org',
  'wiki',
  'Heights_of_presidents_and_presidential_candidates_of_the_United_States',
  sep = '/'
  )
table = htmltab(table_url, which = 4)
table_names = c(
  'election', 'winner', 'trash1', 'height_win',
  'opponent', 'trash2', 'height_lose',
  'trash3','height_diff'
)
```

```r
names(table) = table_names
table = table[
  1:which(table$election == '1948'),
  -which(names(table) %in% c('trash1','trash2','trash3'))
  ]
table$height_win = as.numeric(gsub('[^0-9]','',table$height_win))
table$height_lose = as.numeric(gsub('[^0-9]','',table$height_lose))
table$height_diff = as.numeric(gsub('[^0-9]','',table$height_diff))

min_height = min(table[c('height_win','height_lose')])
max_height = max(table[c('height_win','height_lose')])
plot_lims = c(min_height, max_height)
```

```r
plot(
  height_lose ~ height_win,
  data = table,
  xlim = plot_lims, ylim = plot_lims,
  main = 'Height of Elected vs Opponent \n in Presidential Elections, 1948 - Present',
  xlab = 'Height of Elected (cm)', ylab = 'Height of Opponent (cm)',
  pch = 8, col = 4
  )
abline(0,1,col = 2)
```



The plot in the wikipedia article includes additional text, additional boundary lines (indicating where one party was shorter or taller than the other, in ranges of 10, clearly labelled), as well as a grid and background shading. It's not clear that these things add anything to the plot in terms of ease of consumption.

The wikipedia article plot also includes a longer range of time. I used the range 1948 to present, because that was what was used in example 1.2, and it made data cleaning easier when pulling from the wikipedia table.

Question 7

```
q1.7_answerer = function(n, lambda){
  data = rpois(n, lambda = lambda)
  freqs = table(data)
  actual = freqs / n
  theoretical = dpois(0:max(data), lambda = lambda)
  # Output
  print(freqs)
  print(c(Mean = mean(data), Variance = var(data)))
  print(cbind(actual, theoretical))
}
```

For $n = 1000$

```
q1.7_answerer(1e3, lambda = 0.61)
```

```
## data
##   0   1   2   3   4   5
## 519 335 123  16   6   1
##      Mean  Variance
## 0.6580000 0.6596957
```

```
##   actual  theoretical
## 0  0.519 0.5433508691
## 1  0.335 0.3314440301
## 2  0.123 0.1010904292
## 3  0.016 0.0205550539
## 4  0.006 0.0031346457
## 5  0.001 0.0003824268
```

For $n = 10000$:

```
q1.7_answerer(1e4, lambda = 0.61)
```

```
## data
##    0    1    2    3    4    5    6
## 5521 3220  989  229   35    5    1
##      Mean  Variance
## 0.6056000 0.6291116
##   actual  theoretical
## 0 0.5521 5.433509e-01
## 1 0.3220 3.314440e-01
## 2 0.0989 1.010904e-01
## 3 0.0229 2.055505e-02
## 4 0.0035 3.134646e-03
## 5 0.0005 3.824268e-04
## 6 0.0001 3.888006e-05
```

Question 8

```
q1.8_answerer = function(n, lambda){
  data = rpois(n, lambda = lambda)
  freqs = table(data)
  actual_d = freqs / n
  actual_c = cumsum(actual_d)
  theoretical_d = dpois(0:max(data), lambda = lambda)
  theoretical_c = ppois(0:max(data), lambda = lambda)

  df = data.frame(freqs, actual_d, theoretical_d, actual_c, theoretical_c)
  df = df[,-3]
  names(df) = c('Value','Frequency','EmpDist','TheoryDist','EmpCumDist','TheoryCumDist')
  ## Output
  print(df)
}
```

For $n = 1000$

```
q1.8_answerer(1e3, lambda = 0.61)
```

```
##   Value Frequency EmpDist   TheoryDist EmpCumDist TheoryCumDist
## 0     0       525   0.525 0.5433508691      0.525     0.5433509
## 1     1       334   0.334 0.3314440301      0.859     0.8747949
## 2     2       110   0.110 0.1010904292      0.969     0.9758853
## 3     3        26   0.026 0.0205550539      0.995     0.9964404
## 4     4         3   0.003 0.0031346457      0.998     0.9995750
## 5     5         2   0.002 0.0003824268      1.000     0.9999575
```

For $n = 10000$:

```r
q1.8_answerer(1e4, lambda = 0.61)
```

```
##   Value Frequency EmpDist   TheoryDist EmpCumDist TheoryCumDist
## 0     0      5428  0.5428 5.433509e-01     0.5428     0.5433509
## 1     1      3318  0.3318 3.314440e-01     0.8746     0.8747949
## 2     2      1003  0.1003 1.010904e-01     0.9749     0.9758853
## 3     3       218  0.0218 2.055505e-02     0.9967     0.9964404
## 4     4        29  0.0029 3.134646e-03     0.9996     0.9995750
## 5     5         3  0.0003 3.824268e-04     0.9999     0.9999575
## 6     6         1  0.0001 3.888006e-05     1.0000     0.9999963
```

Chapter 2

Question 2

```r
data(iris)
iris %>%
  group_by(Species) %>%
  summarize(
    SepalW.avg = mean(Sepal.Width),
    SepalL.avg = mean(Sepal.Length),
    PetalW.Avg = mean(Petal.Width),
    PetalL.avg = mean(Petal.Length)
    )
```

```
## # A tibble: 3 x 5
##      Species SepalW.avg SepalL.avg PetalW.Avg PetalL.avg
##       <fctr>      <dbl>      <dbl>      <dbl>      <dbl>
## 1     setosa      3.428      5.006      0.246      1.462
## 2 versicolor      2.770      5.936      1.326      4.260
## 3  virginica      2.974      6.588      2.026      5.552
```

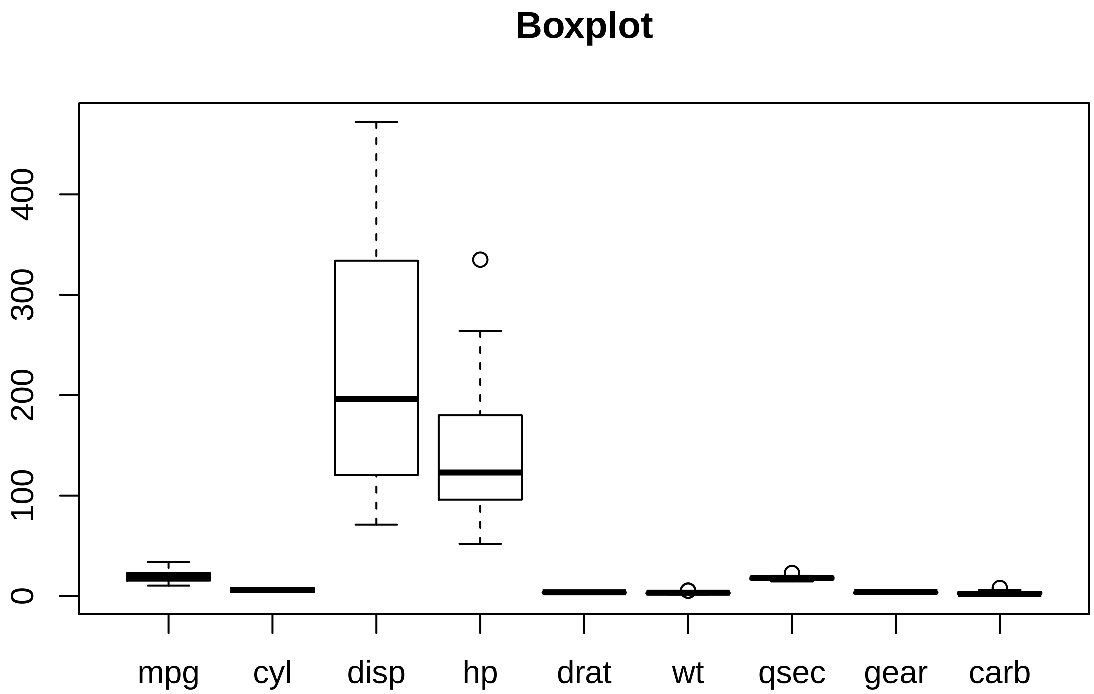
Question 3

```r
data(mtcars)
head(mtcars)
```

```
##                    mpg cyl disp  hp drat    wt  qsec vs am gear carb
## Mazda RX4         21.0   6  160 110 3.90 2.620 16.46  0  1    4    4
## Mazda RX4 Wag     21.0   6  160 110 3.90 2.875 17.02  0  1    4    4
## Datsun 710        22.8   4  108  93 3.85 2.320 18.61  1  1    4    1
## Hornet 4 Drive    21.4   6  258 110 3.08 3.215 19.44  1  0    3    1
## Hornet Sportabout 18.7   8  360 175 3.15 3.440 17.02  0  0    3    2
## Valiant           18.1   6  225 105 2.76 3.460 20.22  1  0    3    1
```

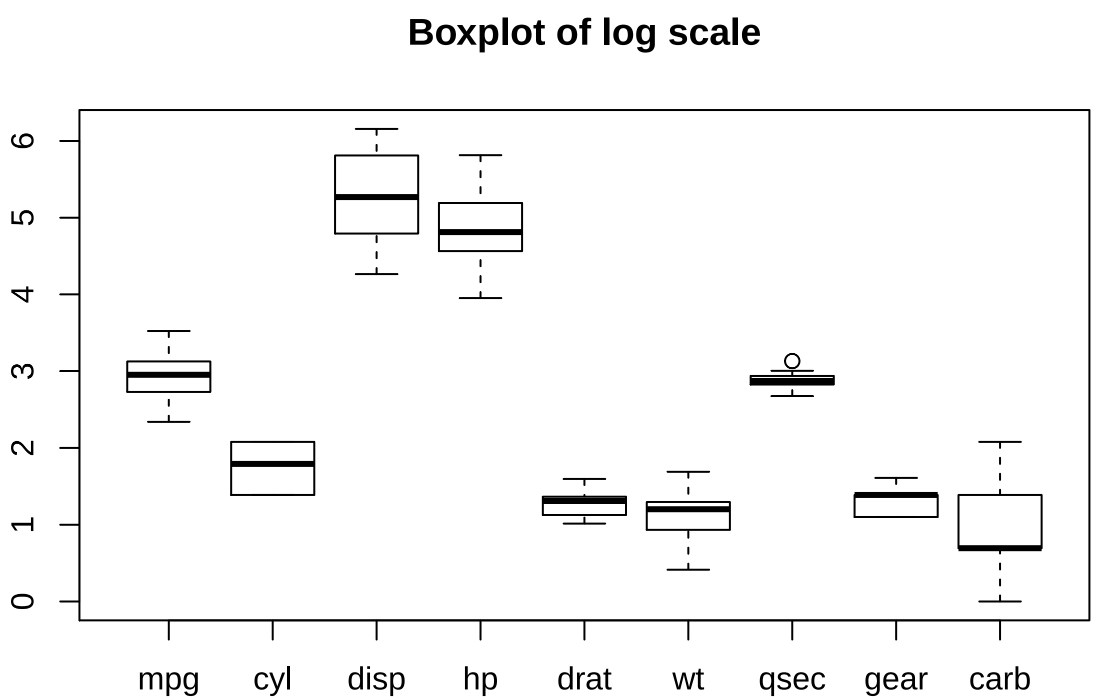
```r
#?mtcars
attach(mtcars)
variables1 = mtcars[,-(8:9)]
```

```r
boxplot(variables1, main = "Boxplot")
```



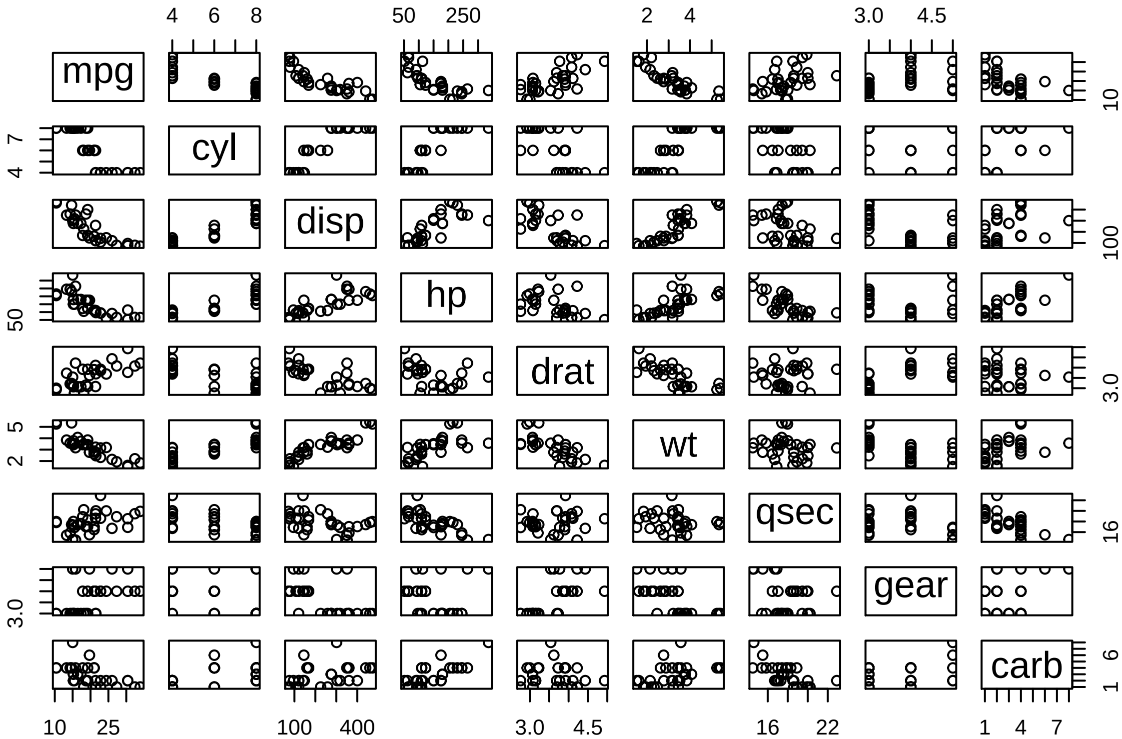
The boxplot is not very easy to interpret, so we try taking the log of the quantitative variables and produce a second boxplot.

```
log_variables = log(variables1)
boxplot(log_variables, main = "Boxplot of log scale")
```



The above boxplot is simpler to interpret. But each of the variables are represented/measured on completely different scales, so this plot isnt very useful. Next we look at the pairs plot:

```
pairs(variables1)
```



Some of the variables in the pairs plot appear to have a linear relationship. For instance mpg and displacement, mph and weight, weight and displacement, weight and rear axle ratio, weight and gross horsepower all have a possible linear relationship.

Question 4

```
mammals$r = mammals$brain/mammals$body
mammals = mammals[order(mammals$r, decreasing = TRUE),]
```

Mammals with largest ratio of brain to body size:

```
head(mammals)
```

```
##                             body  brain        r
## Ground squirrel            0.101   4.00 39.60396
## Owl monkey                 0.480  15.50 32.29167
## Lesser short-tailed shrew 0.005   0.14 28.00000
## Rhesus monkey              6.800 179.00 26.32353
## Galago                     0.200   5.00 25.00000
## Little brown bat           0.010   0.25 25.00000
```

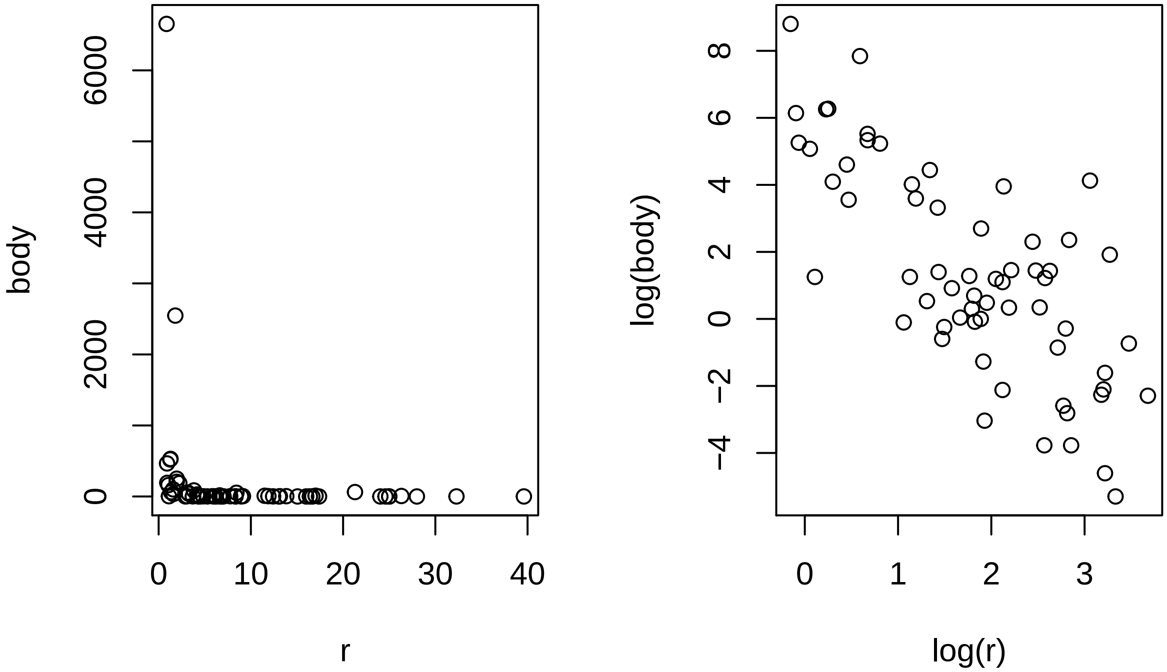
Mammals with smallest ratio of brain to body size:

```
tail(mammals)
```

```
##                     body  brain         r
## Horse              521.0  655.0 1.2571977
## Water opossum        3.5    3.9 1.1142857
## Brazilian tapir    160.0  169.0 1.0562500
## Pig                192.0  180.0 0.9375000
## Cow                465.0  423.0 0.9096774
## African elephant  6654.0 5712.0 0.8584310
```

Question 5

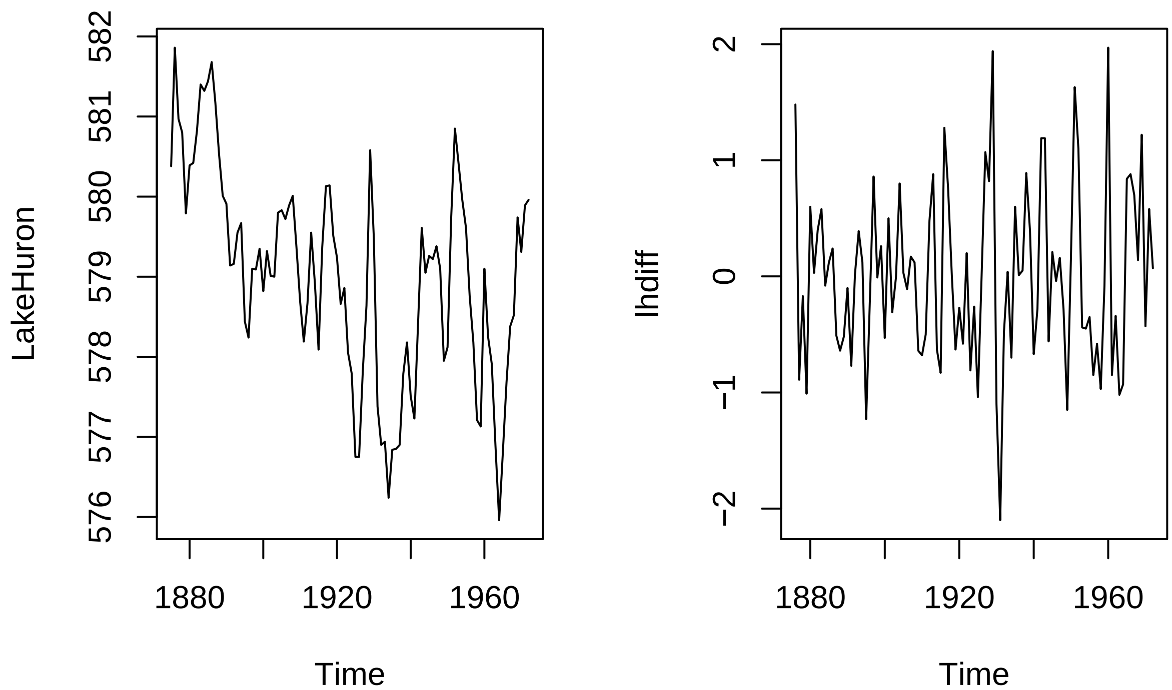
```
par(mfrow = c(1,2))
plot(body ~ r, data = mammals)
plot(log(body) ~ log(r), data = mammals)
```




```
par(mfrow = c(1,1))
```

Question 6

```
lhdiff = diff(LakeHuron)
par(mfrow = c(1,2))
plot(LakeHuron)
plot(lhdiff)
```



The mean does appear to change with respect to time during the years from 1880 to 1900, but we don't have any indication of what came before 1880, so that might be a premature conclusiojn. It appears that the mean may have settled down post 1920, but the variance seems to be getting increasingly more erratic.

After taking the first difference, the mean has stabilized.

Question 7

```
q2.7_answerer = function(n,m){
  random_numbers = matrix(runif(n*m, min = 0, max = 1), ncol = m)
  print('Column Means')
  print(apply(random_numbers,2,mean))
  print('Covariance Matrix')
  print(var(random_numbers))
  print('Diagonal of Covariance Matrix')
  print(diag(var(random_numbers)))
  print('Pairwise Correlation Matrix')
  print(cor(random_numbers))
  #cloud(z ~ x + y, data = random_numbers)
  random_means = as.matrix(random_numbers) %*% rep(1,m) / m
  truehist(random_means)
  curve(dnorm(x, 1/2, sqrt(1/(12*m))), col = 'red', lwd = 2, add = TRUE)
}
```

```
q2.7_answerer(400,3)
```

```
## [1] "Column Means"
## [1] 0.4941445 0.5389658 0.5084955
## [1] "Covariance Matrix"
##               [,1]         [,2]          [,3]
## [1,]  0.0810472226 -0.007856142  0.0008253281
## [2,] -0.0078561419  0.083520526 -0.0031168145
```

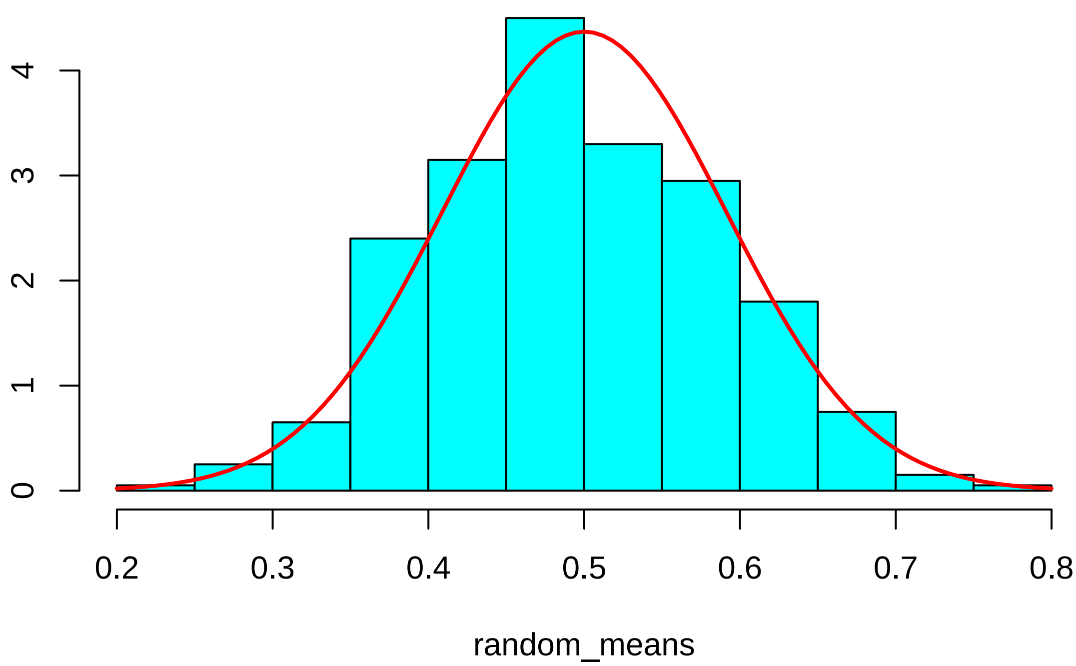
```
## [3,]  0.0008253281 -0.003116815  0.0781246441
## [1] "Diagonal of Covariance Matrix"
## [1] 0.08104722 0.08352053 0.07812464
## [1] "Pairwise Correlation Matrix"
##             [,1]        [,2]        [,3]
## [1,]  1.00000000 -0.09548687  0.01037203
## [2,] -0.09548687  1.00000000 -0.03858516
## [3,]  0.01037203 -0.03858516  1.00000000
```

Question 8

```
q2.7_answerer(400,10)
```

```
## [1] "Column Means"
##  [1] 0.4841873 0.5142989 0.4844120 0.4882563 0.4830727 0.5037321 0.5090053
##  [8] 0.5054082 0.5022667 0.4587518
## [1] "Covariance Matrix"
##               [,1]          [,2]          [,3]         [,4]          [,5]
##  [1,]  7.957090e-02 -4.689904e-03 -9.335973e-06  0.003392878  0.0055613652
##  [2,] -4.689904e-03  8.545416e-02 -7.515600e-05 -0.001537049  0.0004618758
##  [3,] -9.335973e-06 -7.515600e-05  7.760066e-02  0.004731897 -0.0039018265
##  [4,]  3.392878e-03 -1.537049e-03  4.731897e-03  0.079491425  0.0020219737
##  [5,]  5.561365e-03  4.618758e-04 -3.901826e-03  0.002021974  0.0902961102
##  [6,] -3.161368e-03  2.254087e-03  1.436268e-03 -0.007832144 -0.0043681094
##  [7,]  6.534302e-04 -9.385186e-03  3.183597e-03 -0.004903373 -0.0030347661
##  [8,]  8.453328e-03  1.936223e-03  3.206574e-03 -0.001406436  0.0058411299
##  [9,]  6.428154e-03 -3.292692e-03  2.990486e-03  0.002437849 -0.0072261492
## [10,]  1.223142e-03  4.524243e-05 -3.731526e-03 -0.005988715 -0.0079247175
##               [,6]          [,7]         [,8]          [,9]         [,10]
##  [1,] -0.0031613684  0.0006534302  0.008453328  0.0064281536  1.223142e-03
##  [2,]  0.0022540874 -0.0093851855  0.001936223 -0.0032926915  4.524243e-05
##  [3,]  0.0014362679  0.0031835969  0.003206574  0.0029904857 -3.731526e-03
```

```
##  [4,] -0.0078321439 -0.0049033731 -0.001406436  0.0024378487 -5.988715e-03
##  [5,] -0.0043681094 -0.0030347661  0.005841130 -0.0072261492 -7.924718e-03
##  [6,]  0.0833934362  0.0053745462 -0.004601729  0.0003061487  4.943386e-03
##  [7,]  0.0053745462  0.0876269505  0.002527913  0.0044253992  5.672270e-03
##  [8,] -0.0046017293  0.0025279132  0.080996395  0.0010123693  4.451276e-03
##  [9,]  0.0003061487  0.0044253992  0.001012369  0.0852180556  6.123537e-04
## [10,]  0.0049433863  0.0056722700  0.004451276  0.0006123537  8.389959e-02
## [1] "Diagonal of Covariance Matrix"
##  [1] 0.07957090 0.08545416 0.07760066 0.07949142 0.09029611 0.08339344
##  [7] 0.08762695 0.08099639 0.08521806 0.08389959
## [1] "Pairwise Correlation Matrix"
##               [,1]          [,2]          [,3]        [,4]         [,5]
##  [1,]  1.0000000000 -0.0568748484 -0.0001188091  0.04266100  0.065609958
##  [2,] -0.0568748484  1.0000000000 -0.0009229207 -0.01864923  0.005258042
##  [3,] -0.0001188091 -0.0009229207  1.0000000000  0.06024797 -0.046612321
##  [4,]  0.0426609971 -0.0186492326  0.0602479749  1.00000000  0.023866067
##  [5,]  0.0656099577  0.0052580421 -0.0466123210  0.02386607  1.000000000
##  [6,] -0.0388089630  0.0267016566  0.0178540527 -0.09619546 -0.050337663
##  [7,]  0.0078253401 -0.1084569558  0.0386070416 -0.05875109 -0.034117080
##  [8,]  0.1052974189  0.0232731925  0.0404460088 -0.01752778  0.068301384
##  [9,]  0.0780626519 -0.0385850114  0.0367741986  0.02961971 -0.082377118
## [10,]  0.0149699341  0.0005343175 -0.0462459645 -0.07333201 -0.091047775
##              [,6]        [,7]        [,8]         [,9]         [,10]
##  [1,] -0.038808963  0.00782534  0.10529742  0.078062652  0.0149699341
##  [2,]  0.026701657 -0.10845696  0.02327319 -0.038585011  0.0005343175
##  [3,]  0.017854053  0.03860704  0.04044601  0.036774199 -0.0462459645
##  [4,] -0.096195462 -0.05875109 -0.01752778  0.029619709 -0.0733320113
##  [5,] -0.050337663 -0.03411708  0.06830138 -0.082377118 -0.0910477754
##  [6,]  1.000000000  0.06287196 -0.05599152  0.003631622  0.0590988030
##  [7,]  0.062871963  1.00000000  0.03000616  0.051211535  0.0661543057
##  [8,] -0.055991524  0.03000616  1.00000000  0.012185415  0.0539972650
##  [9,]  0.003631622  0.05121154  0.01218541  1.000000000  0.0072419686
## [10,]  0.059098803  0.06615431  0.05399726  0.007241969  1.0000000000
```



The Central Limit Theorem tells us that as the sample size increases, the distribution of the sample means will tend more and more towards a normal distribution.

Question 12

```
mammals = mammals[order(mammals$brain, decreasing = TRUE),]
```

Mammals with largest brain size:

```
head(mammals)
```

```
##                  body brain         r
## African elephant 6654  5712  0.858431
## Asian elephant   2547  4603  1.807224
## Human              62  1320 21.290323
## Giraffe           529   680  1.285444
## Horse             521   655  1.257198
## Okapi             250   490  1.960000
```

Mammals with smallest brain size:

```
tail(mammals)
```

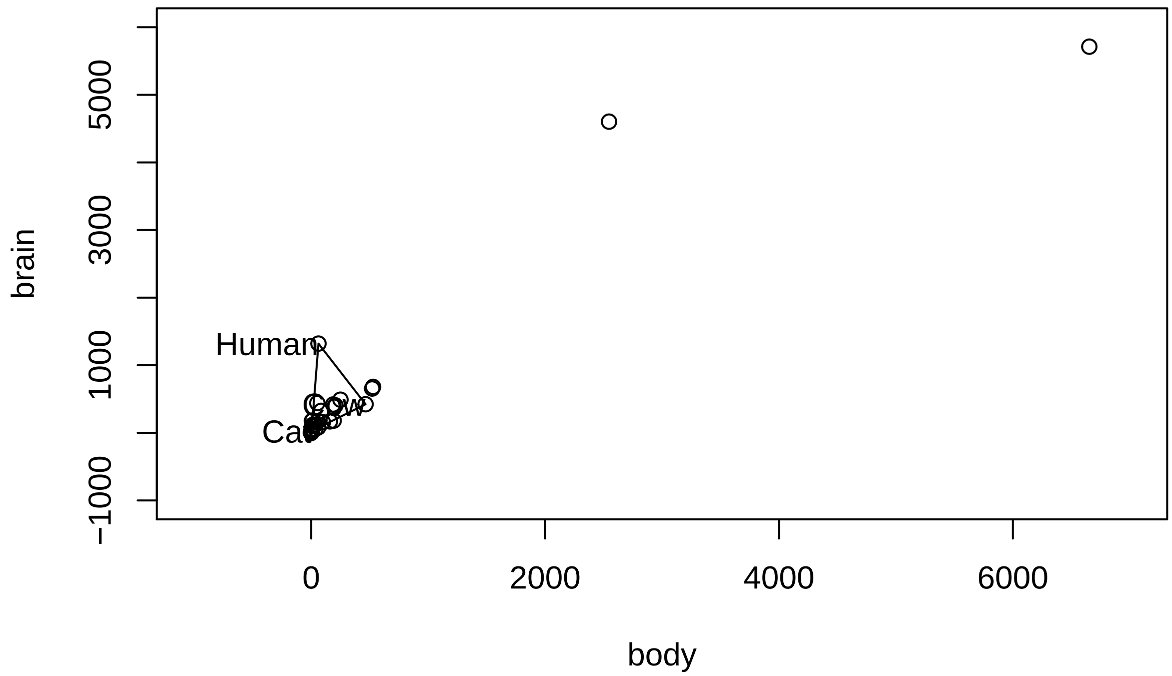
```
##                            body brain         r
## Golden hamster            0.120  1.00  8.333333
## Mouse                     0.023  0.40 17.391304
## Musk shrew                0.048  0.33  6.875000
## Big brown bat             0.023  0.30 13.043478
## Little brown bat          0.010  0.25 25.000000
## Lesser short-tailed shrew 0.005  0.14 28.000000
```

Question 13

```
data(mammals)

plot(mammals$body,mammals$brain,xlab="body", ylab="brain", ylim = c(-1000,6000),
     xlim = c(-1000,7000))

y = mammals[c("Cat", "Cow", "Human"), ]
polygon(y)
text(y, rownames(y), adj=c(1, .5))
```



The scatterplot in figure 2.19 is easier to see and interpret since it is on the log-log scale. The observations on this plot with the original scaling are too close.

Chapter 3

Question 2

```
die1 = sample(1:6, 1000, replace = TRUE)
die2 = sample(1:6, 1000, replace = TRUE)
die.sum = die1 + die2
print(
  data.frame(
    Sum = 2:12,
    Frequency = as.vector(table(die.sum)),
    EmpProb = as.vector(table(die.sum)/1000),
    AbsProb = getSumProbs(ndicePerRoll = 2,nsidesPerDie = 6)$probabilities[,2]
    )
  )
```

```
##    Sum Frequency EmpProb    AbsProb
## 1    2        25   0.025 0.02777778
## 2    3        42   0.042 0.05555556
## 3    4        82   0.082 0.08333333
## 4    5       119   0.119 0.11111111
## 5    6       152   0.152 0.13888889
## 6    7       166   0.166 0.16666667
## 7    8       131   0.131 0.13888889
## 8    9       126   0.126 0.11111111
## 9   10        66   0.066 0.08333333
## 10  11        67   0.067 0.05555556
## 11  12        24   0.024 0.02777778
```

Question 3

```r
#(a)
pujols = data.frame(
  nhits = c('0', '1', '2', '3+'),
  freq = c(17,31,17,5),
  expected = dbinom(c(0,1,2,3), size = 4, p = 0.312)
  )
pujols[4,3] = 1-sum(pujols[1:3,3]) # fix for "3 or more"
chisq.test(pujols$freq, p = pujols$expected)
```

```
##
##  Chi-squared test for given probabilities
##
## data:  pujols$freq
## X-squared = 0.97692, df = 3, p-value = 0.8068
```

$$H_0 : \text{the counts follow binomial(4,3.12) distribution}$$
$$H_a : \text{the counts do not follow binomial(4,3.12) distribution}$$

From the R output, there is not enough evidence to reject the null hypothesis that Pujol's batting follows a binomial distribution, with $p = 0.312$.

```r
#(b)
pujols = data.frame(
  nhits = c('0', '1', '2', '3+'),
  freq = c(5,6,4,11),
  expected = dbinom(c(0,1,2,3), size = 5, p = 0.312)
  )
pujols[4,3] = 1-sum(pujols[1:3,3]) # fix for "3 or more"
chisq.test(pujols$freq, p = pujols$expected)
```

```
## Warning in chisq.test(pujols$freq, p = pujols$expected): Chi-squared
## approximation may be incorrect

##
##  Chi-squared test for given probabilities
##
## data:  pujols$freq
## X-squared = 12.094, df = 3, p-value = 0.007068
```

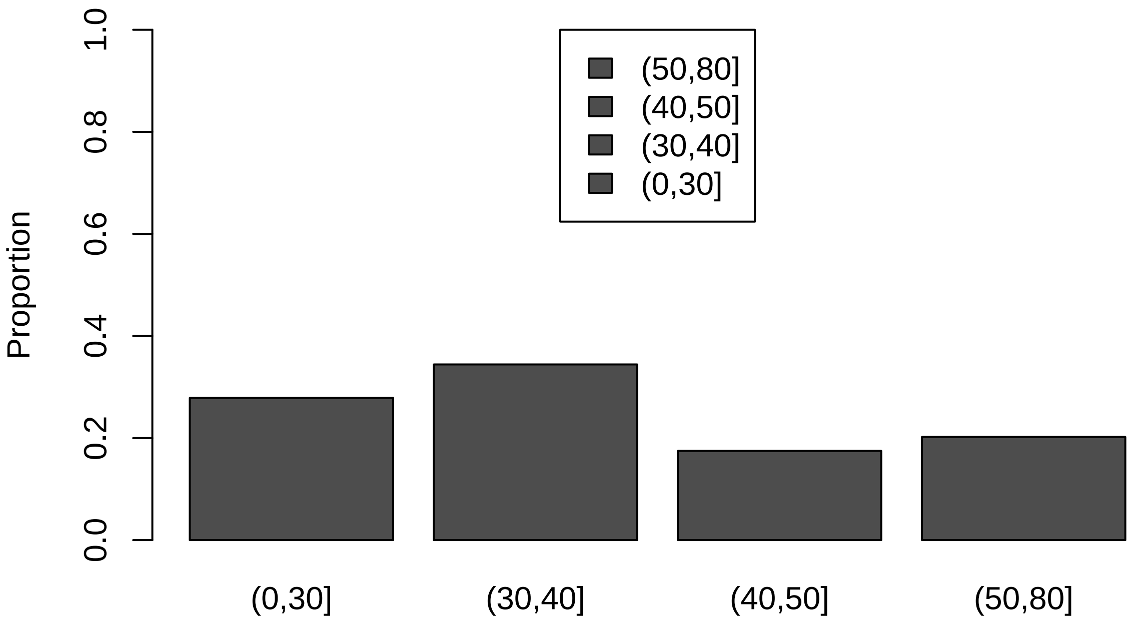
$$H_0 : \text{the counts follow a binomial(5, 0.312) distribution}$$
$$H_a : \text{the counts do not follow a binomial(5, 0.312) distribution}$$

There is strong evidence to reject the hypothesis that Pujol's batting follows a binomial distribution, with $p = 0.007068$.

Question 4

```r
twins=fread('http://personal.bgsu.edu/~mrizzo/Rx/Rx-data/twins.txt',
            header=TRUE, sep=",", na.strings=".")
#twins$AGE
#twins$HRWAGEL

c.age = cut(twins$AGE, breaks=c(0,30, 40, 50,80))
P1=table(c.age)
P2=prop.table(P1)
barplot(t(P2), ylim=c(0, 1), ylab="Proportion", legend.text=dimnames(P2)$c.age,
        args.legend=list(x = "top"))
```



Question 5

```r
#(a)
c.age = cut(twins$AGE, breaks=c(0,30, 40, 50,80))
c.wagel = cut(twins$HRWAGEL, c(0, 7, 13, 20, 150))
table(c.age)
```

```
## c.age
##  (0,30] (30,40] (40,50] (50,80]
##      51      63      32      37
```

```r
table(c.wagel)
```

```
## c.wagel
##    (0,7]   (7,13]  (13,20] (20,150]
##       47       58       38       19
```

```r
#(b)
taw=table(c.age,c.wagel)
taw
```

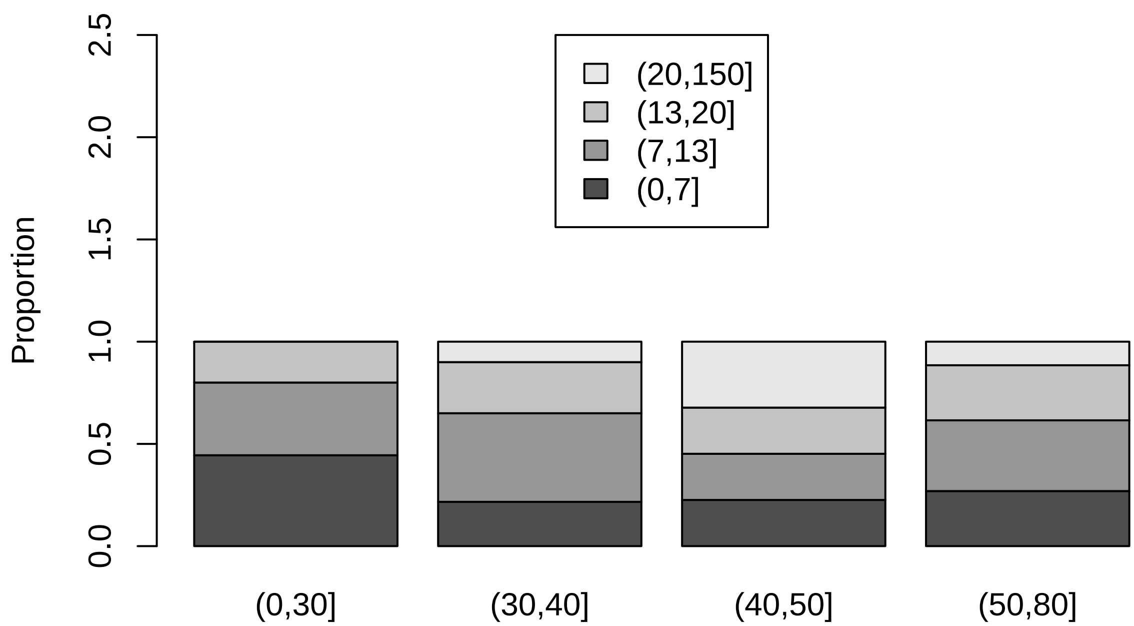
```
##          c.wagel
## c.age     (0,7] (7,13] (13,20] (20,150]
```

```
##    (0,30]     20     16       9        0
##    (30,40]    13     26      15        6
##    (40,50]     7      7       7       10
##    (50,80]     7      9       7        3
```

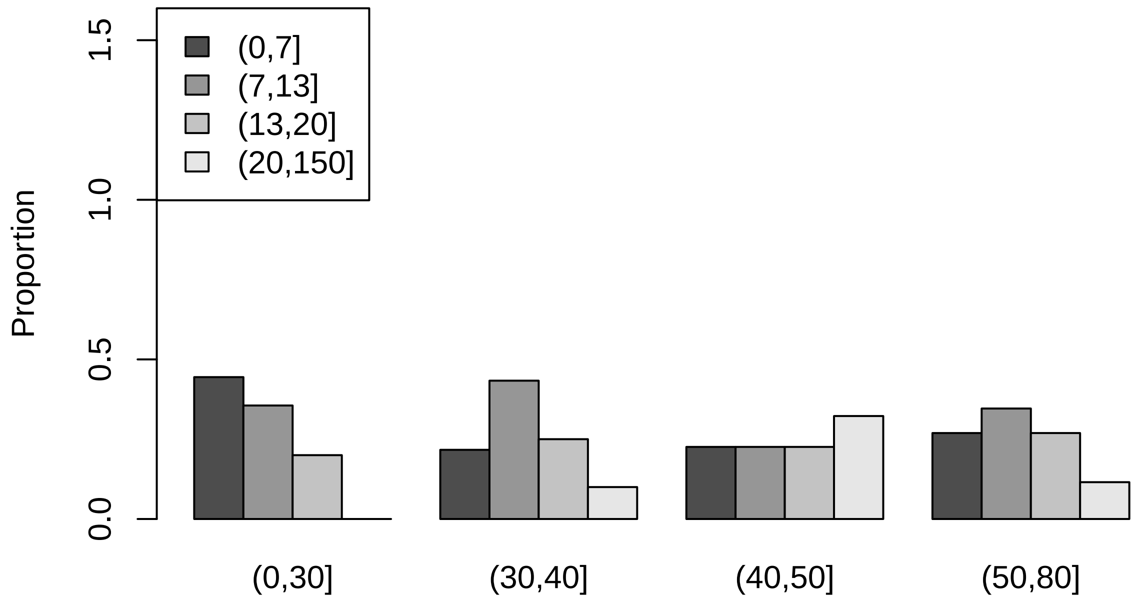
```r
#(c)
prop.table(taw, margin=1)
```

```
##          c.wagel
## c.age          (0,7]    (7,13]   (13,20]  (20,150]
##    (0,30]  0.4444444 0.3555556 0.2000000 0.0000000
##    (30,40] 0.2166667 0.4333333 0.2500000 0.1000000
##    (40,50] 0.2258065 0.2258065 0.2258065 0.3225806
##    (50,80] 0.2692308 0.3461538 0.2692308 0.1153846
```

```r
#(d)
P=prop.table(taw, margin=1)
barplot(t(P), ylim=c(0, 2.5), ylab="Proportion", legend.text=dimnames(P)$c.wagel,
        args.legend=list(x = "top"))
```




```r
barplot(t(P), ylim=c(0, 1.6), beside=T, legend.text=dimnames(P)$c.wagel,
        args.legend=list(x="topleft"), ylab="Proportion")
```




```r
#(e)
```

The twins aged between (0,30] have the lowest hourly wages, and the twins aged between (40,50] tend to have higher hourly wages. There is no clear relation about the older the twins are the hourly wages are higher or vice versa

Question 6

```r
#(a)
tawt=table(c.age,c.wagel)
S = chisq.test(tawt)
```

```
## Warning in chisq.test(tawt): Chi-squared approximation may be incorrect
```

```r
print(S)
```

```
##
##  Pearson's Chi-squared test
##
## data:  tawt
## X-squared = 24.771, df = 9, p-value = 0.003235
```

```r
testqs=sum((tawt - S$expected)^2 / S$expected)
1 - pchisq(testqs, df=9)
```

```
## [1] 0.003235285
```

We perform a test of independence where

$$
\begin{aligned}
H_0 &: \text{ age and wage are independent} \\
H_a &: \text{ age and wage are not independent}
\end{aligned}
$$

From the R output, we have a very small p-value which is strong evidence to reject the null hypothesis in favor of the alternative hypothesis. We conclude that age and wage are dependent.

```r
S$expected
```

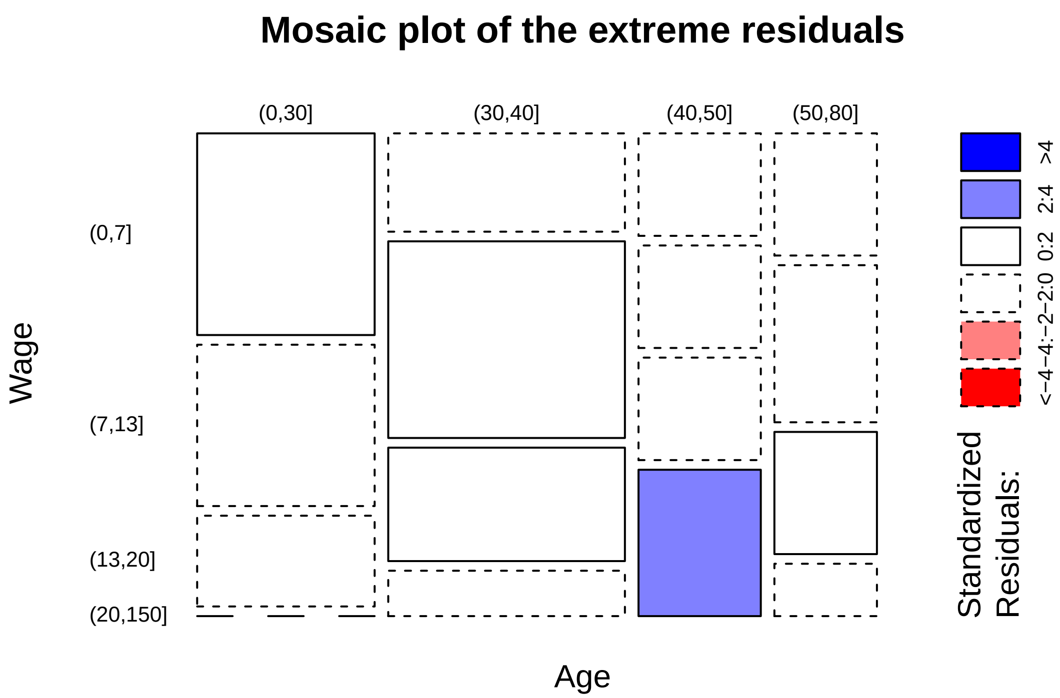
```
##          c.wagel
## c.age         (0,7]    (7,13]   (13,20] (20,150]
##   (0,30]  13.055556 16.111111 10.555556 5.277778
##   (30,40] 17.407407 21.481481 14.074074 7.037037
##   (40,50]  8.993827 11.098765  7.271605 3.635802
##   (50,80]  7.543210  9.308642  6.098765 3.049383
```

We should note that some of the expected values are less than 5, which could indicate that the the Chi-square test of independence is not appropriate. R gives us this warning output. Fisher's exact test may be an alternative.

```r
#b)
S$residuals
```

```
##          c.wagel
## c.age           (0,7]      (7,13]     (13,20]    (20,150]
##   (0,30]   1.92194002 -0.02768183 -0.47878990 -2.29734146
##   (30,40] -1.05637022  0.97490871  0.24681203 -0.39093031
##   (40,50] -0.66483709 -1.23031333 -0.10072158  3.33767089
##   (50,80] -0.19778327 -0.10116070  0.36493614 -0.02827932
```

```r
#(c)
mosaicplot(tawt, shade=TRUE, main = "Mosaic plot of the extreme residuals", ylab = "Wage",
           xlab = "Age", las = 1)
```




```r
#(d)
```

The residuals of the cell containing age less than 30 and wage greater than 20 exceeds 2 in absolute value. This means that fewer people under 30 are earning wages over $20 than expected under the independence model. Also, the residuals of the cell containing ages 40-50 and wage greater than 20 exceeds 2 in absolute value. This means that more 40-50 year olds are earning wages greater than $20 than expected under the independence model.

Question 7

```r
#a)
die1=sample(6,1000, replace=TRUE)
die2=sample(6,1000, replace=TRUE)
```

```r
#b)
max.rolls=pmax(die1,die2)
sum.rolls=die1+die2
```

```r
#(c)
```

The contingency table of the maximum roll and sum of rolls is:

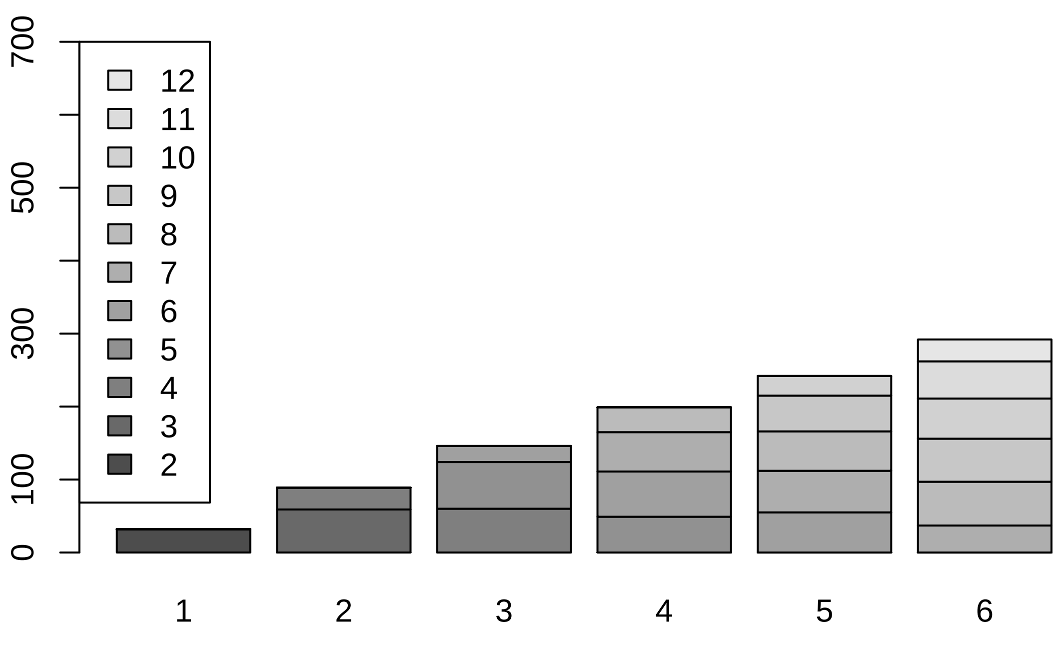
```r
(tp=table(max.rolls,sum.rolls))
```

```
##           sum.rolls
## max.rolls  2  3  4  5  6  7  8  9 10 11 12
##         1 32  0  0  0  0  0  0  0  0  0  0
##         2  0 59 30  0  0  0  0  0  0  0  0
##         3  0  0 60 64 22  0  0  0  0  0  0
##         4  0  0  0 49 62 54 34  0  0  0  0
##         5  0  0  0  0 55 57 54 49 27  0  0
##         6  0  0  0  0  0 37 60 59 55 51 30
```

```r
#(d)
```

We use a barplot to explore the relationship between the maximum roll and the sum of rolls:

```r
barplot(t(tp),legend.text=dimnames(tp)$sum.rolls, args.legend=list(x = "topleft"),ylim=c(0, 700))
```



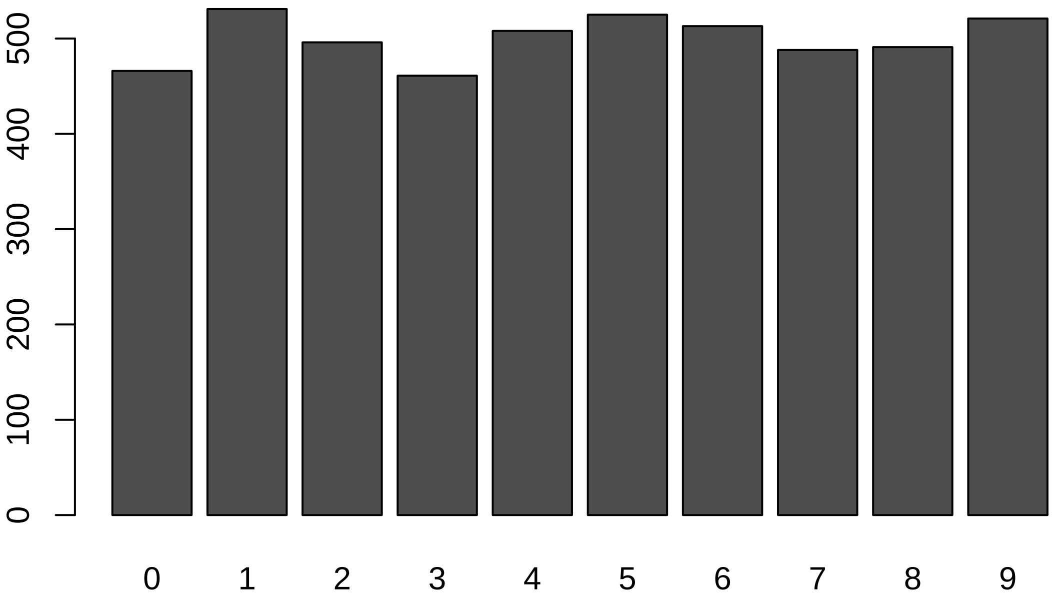
Question 8

```r
#(a)
pidigits = read.table("http://www.itl.nist.gov/div898/strd/univ/data/PiDigits.dat",skip=60)
```

```r
(tpi=table(pidigits))
```

```
## pidigits
##   0   1   2   3   4   5   6   7   8   9 
## 466 531 496 461 508 525 513 488 491 521
```

```r
#(b)
barplot(t(tpi))
```




```r
#(c)
```

We construct a hypothesis test

$$
\begin{aligned}
H_0 &: \text{ the digits 1 through 9 are equally probably in the digits of} \pi \\
H_1 &: \text{ the digits 1 through 9 are not equally probable in the digits of} \pi
\end{aligned}
$$

```r
(spi=chisq.test(tpi))
```

```
## 
##  Chi-squared test for given probabilities
## 
## data:  tpi
## X-squared = 10.356, df = 9, p-value = 0.3224
```

Based on the p-value, there is not enough evidence to reject the null hypothesis. We conclude that the digits 1 through 9 are equally probably in the digits of π.

```r
spi$expected
```

```
##   0   1   2   3   4   5   6   7   8   9 
## 500 500 500 500 500 500 500 500 500 500
```