week7 exercise

Xiang Li

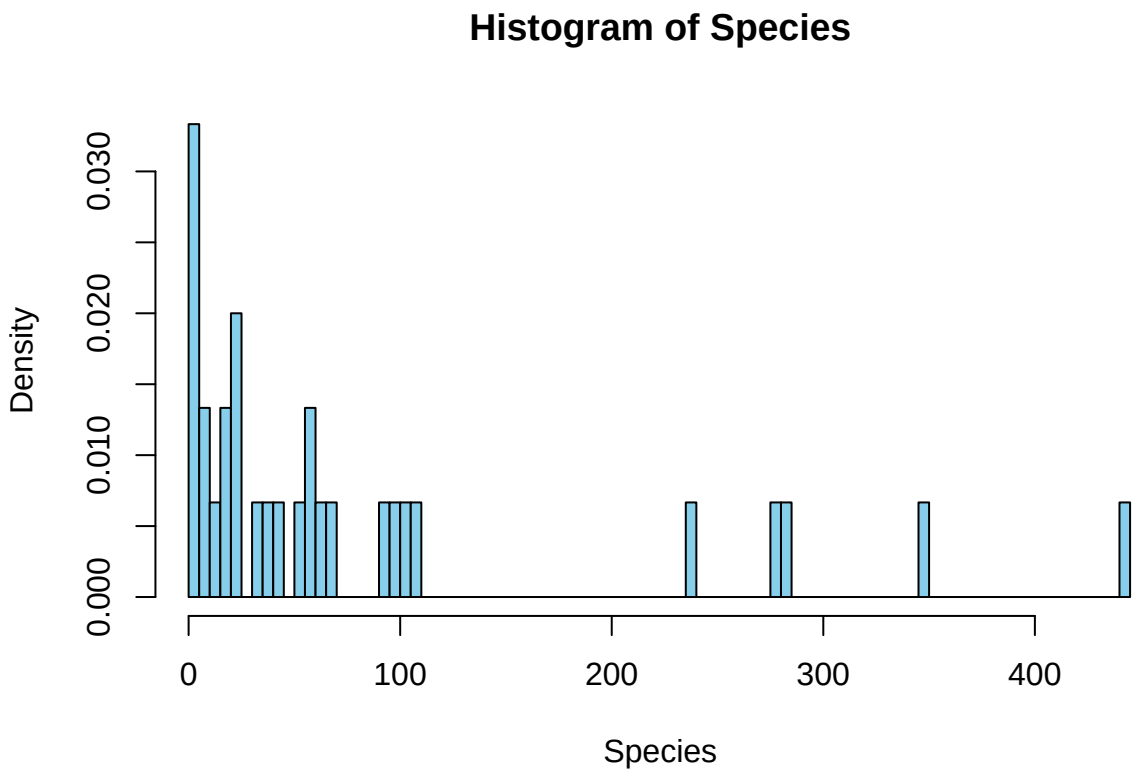
2023/11/16

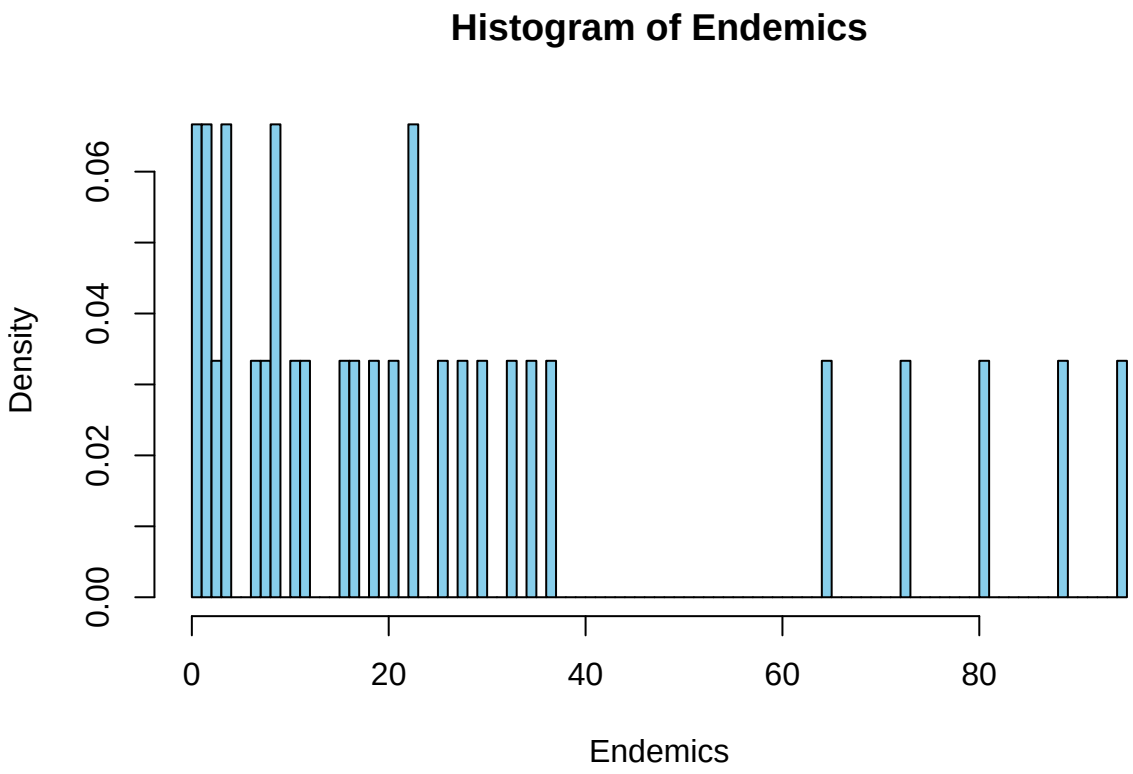
```
library(faraway)
library(MASS)
```

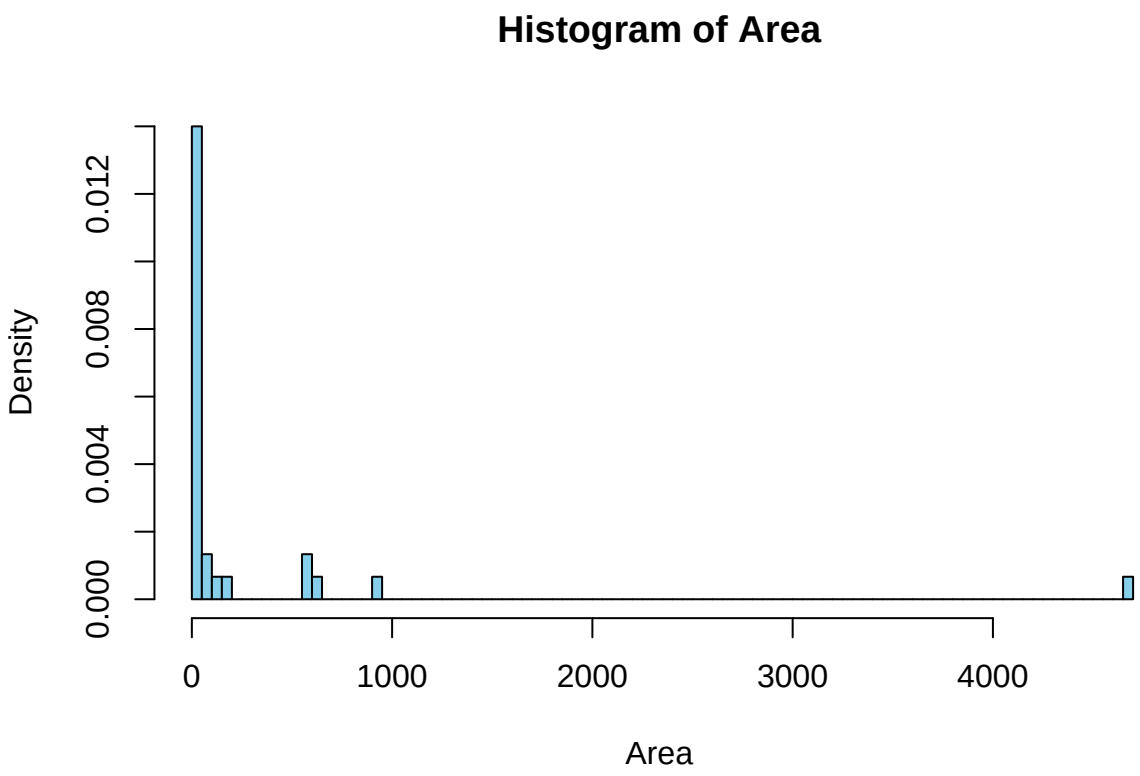
Exercise 2

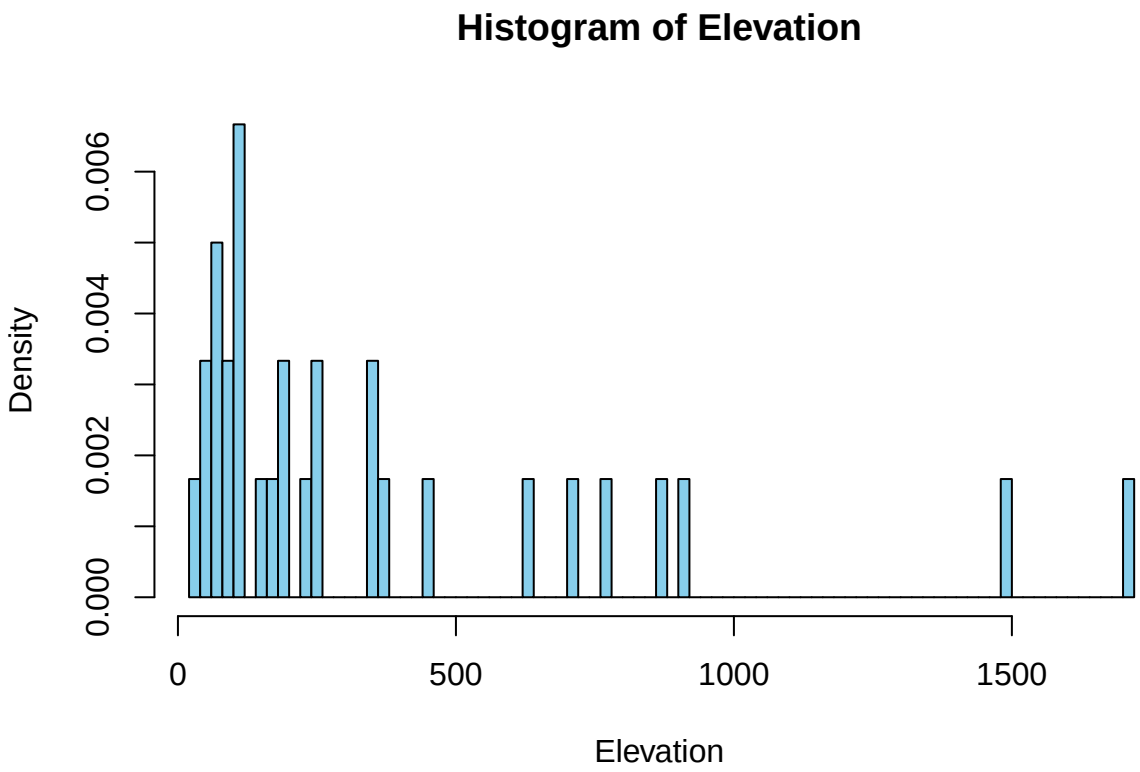
```
f = function(df) {
    num_col = sapply(df, is.numeric)
    char_col = sapply(df, is.character)
    fac_col = sapply(df, is.factor)
    type = num_col | char_col | fac_col
    if (length(type[type]) == 0) {
        return("This dataframe contains no numeric, character or factor values")
    } else {
        num_df = df[, num_col]
        if (ncol(num_df) > 0) {
            for (i in 1:ncol(num_df)) {
                i_name = colnames(num_df)[i]
                hist(num_df[, i], breaks = 100, xlab = i_name,
                  xlim = c(min(num_df[, i]), max(num_df[,
                    i])), col = "skyblue", main = paste0("Histogram of ",
                    i_name), probability = TRUE)
            }
        }
        char_df = df[, char_col | fac_col]
        if (ncol(char_df) > 0) {
            for (i in 1:ncol(char_df)) {
                i_name = colnames(char_df)[i]
                count = aggregate(char_df[, 1, drop = FALSE],
                  by = list(name = char_df[, i]), FUN = length)
                colnames(count)[2] = "count"
                barplot(height = count$count, names.arg = count$name,
                  xlab = i_name, col = "lightgreen", ylim = c(0,
                    max(count$count)), main = paste0("Barplot of ",
                    i_name))
            }
        }
    }
}
```

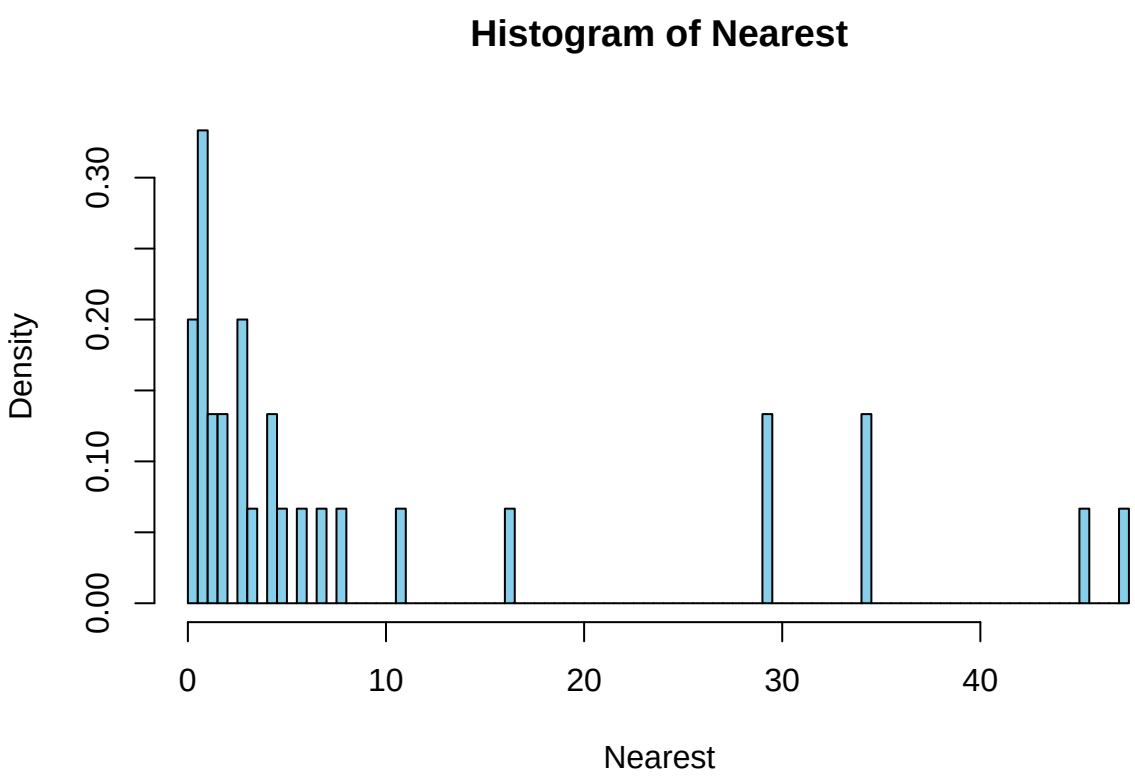
```
f(gala)
```

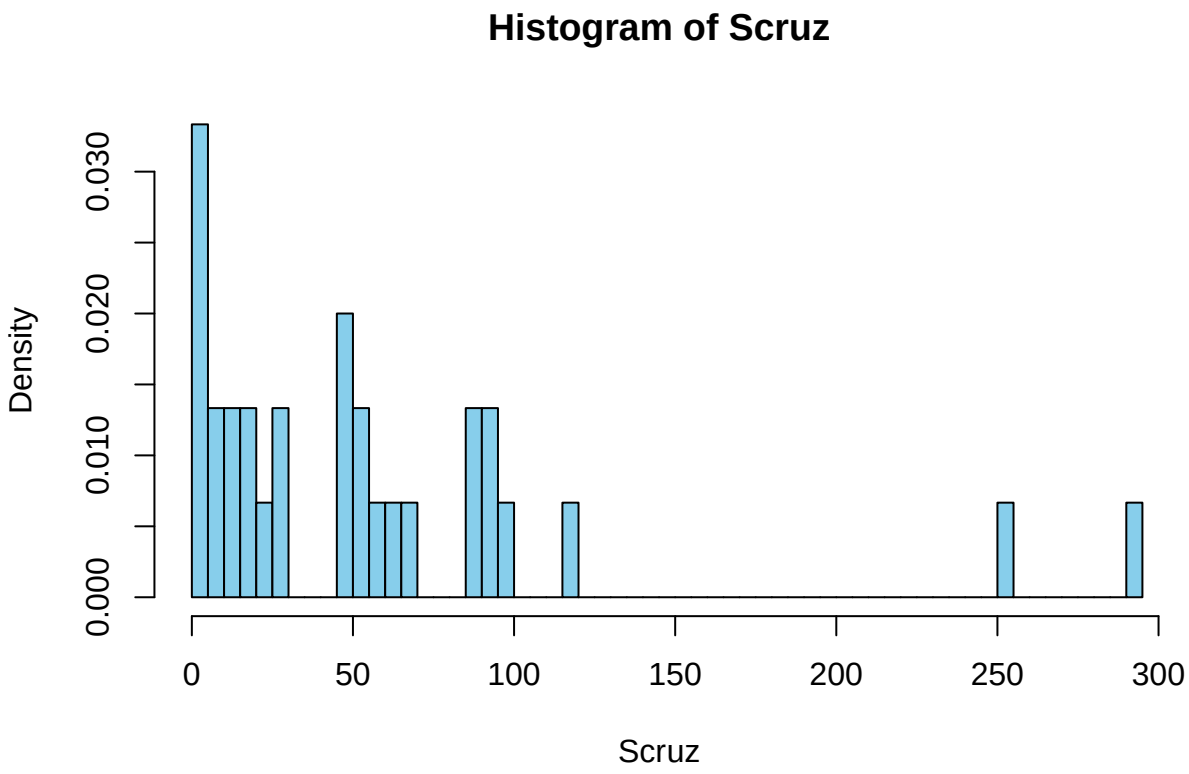

Histogram of Species
Density
0.000
0.010
0.020
0.030
0
100
200
300
400
Species

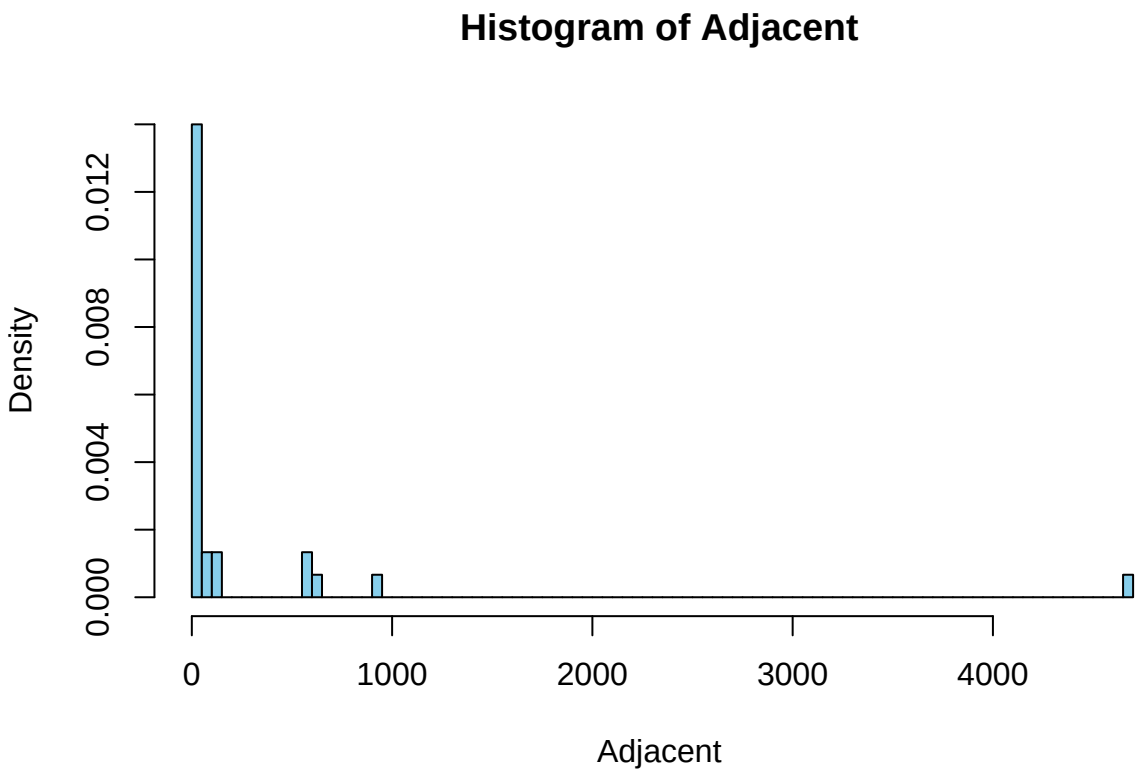

Histogram of Endemics
Density
0.00
0.02
0.04
0.06
0
20
40
60
80
Endemics


Histogram of Area
Density
0.000
0.004
0.008
0.012
0
1000
2000
3000
4000
Area

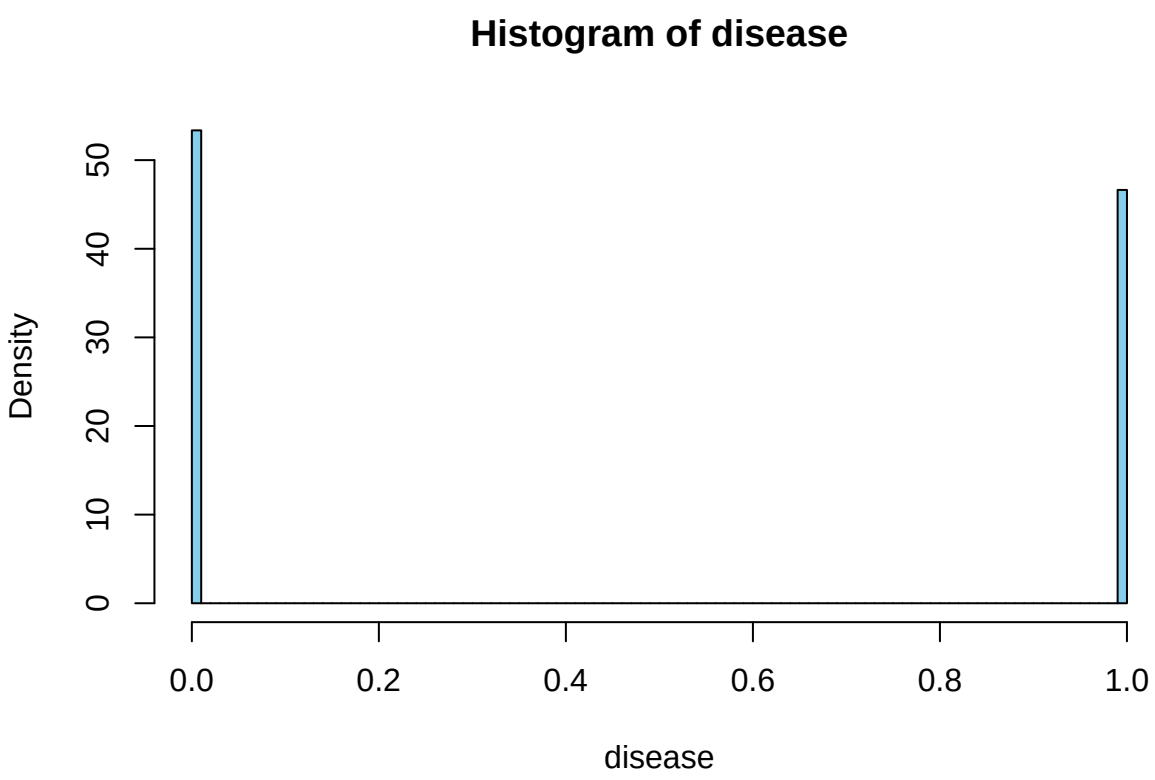

Histogram of Elevation
Density
0.000
0.002
0.004
0.006
0
500
1000
1500
Elevation

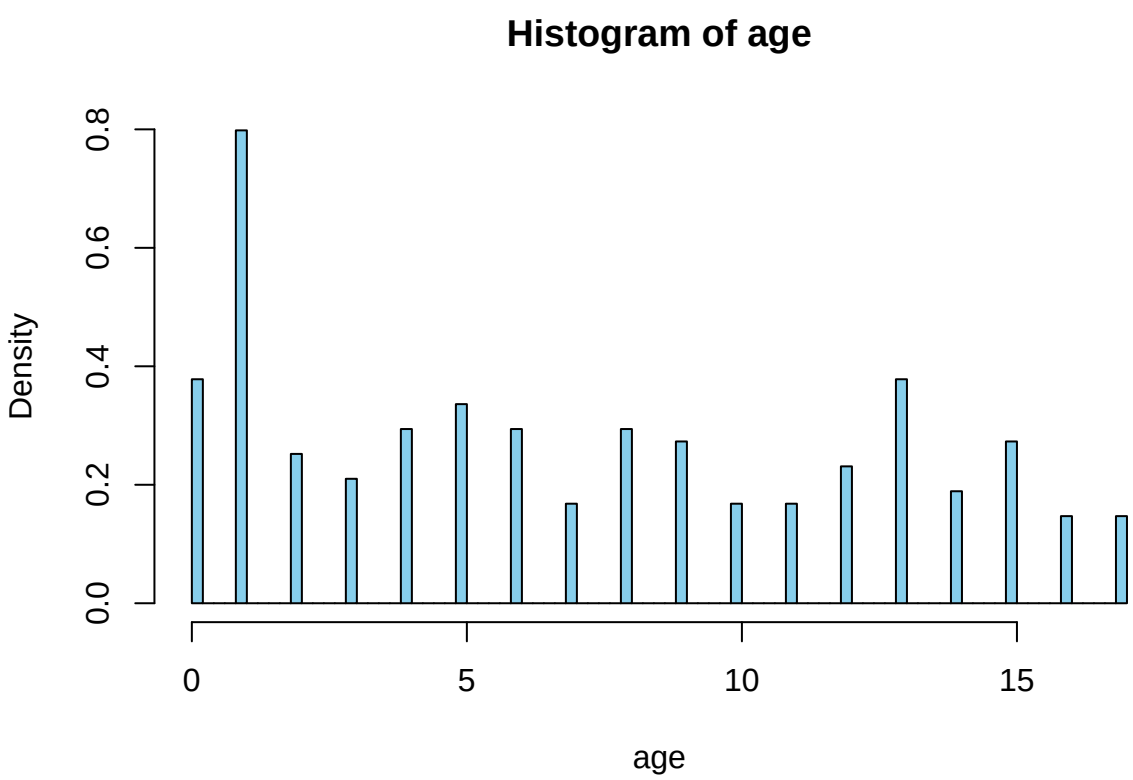

Histogram of Nearest
Density
0.00
0.10
0.20
0.30
0
10
20
30
40
Nearest

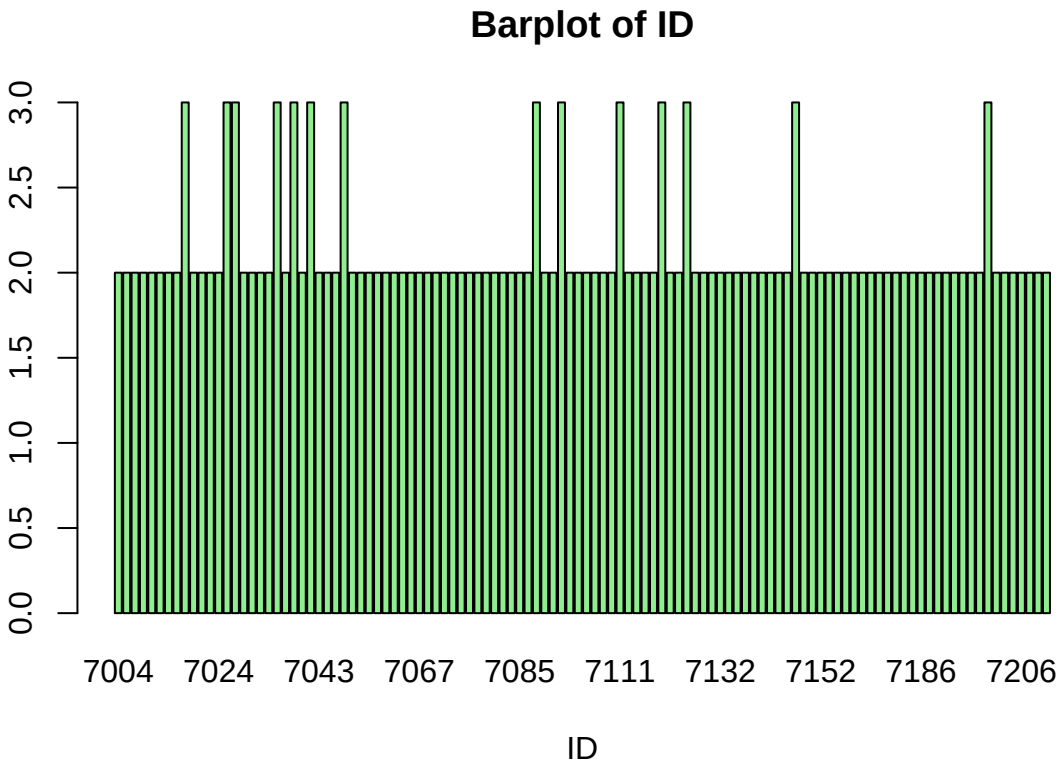

Histogram of Scruz
Density
0.000
0.010
0.020
0.030
0
50
100
150
200
250
300
Scruz

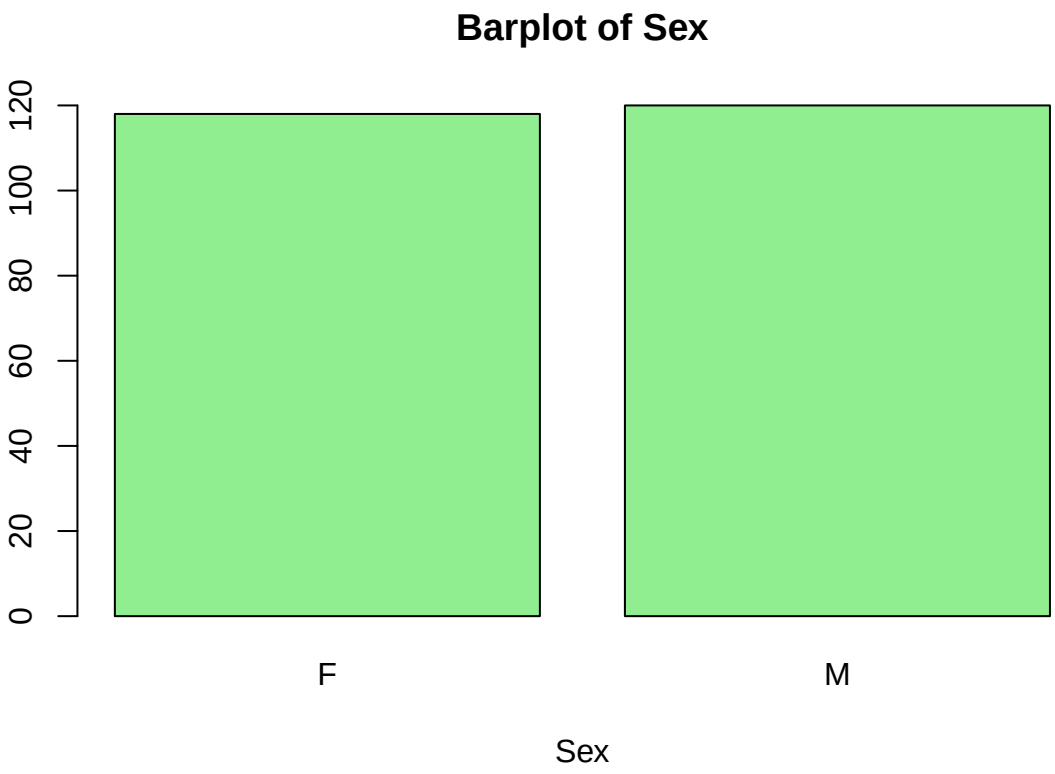

Histogram of Adjacent
Density
0.000
0.004
0.008
0.012
0
1000
2000
3000
4000
Adjacent

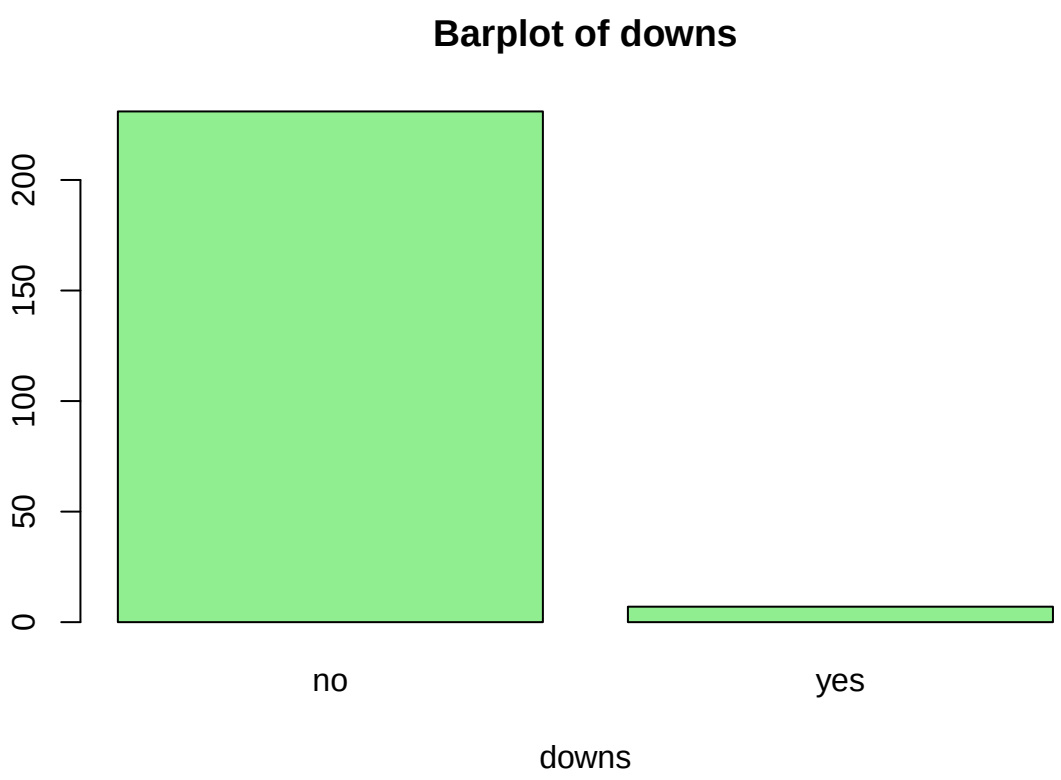
```
f(amlxray)
```

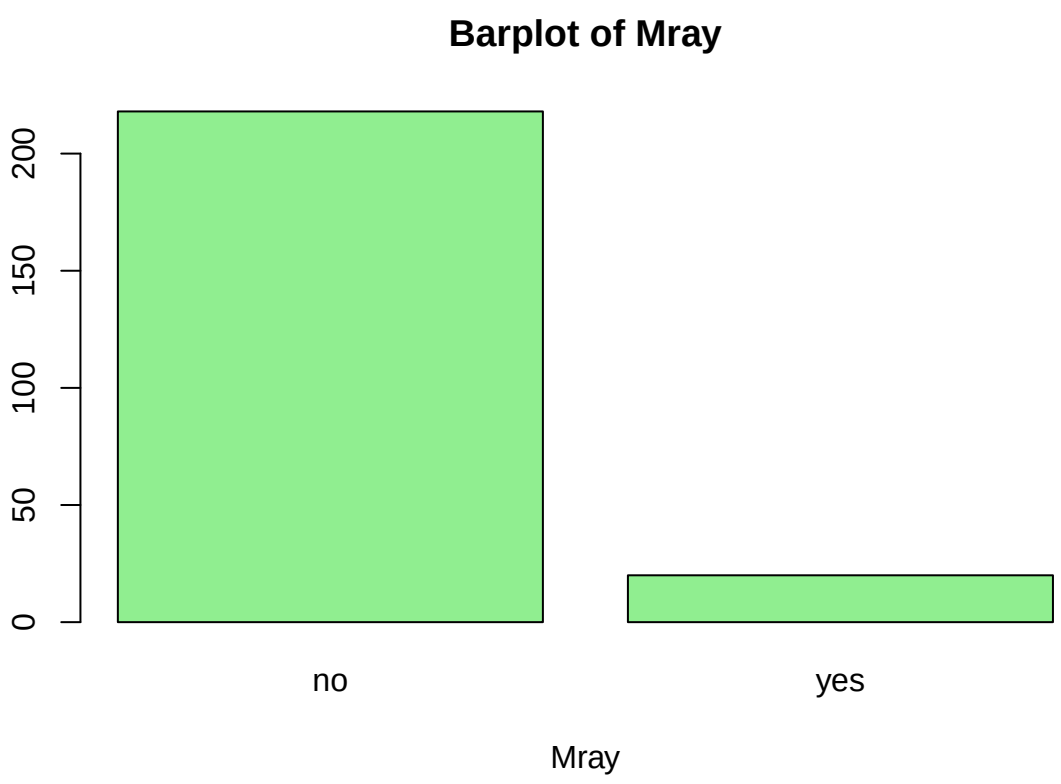

Histogram of disease
Density
0
10
20
30
40
50
0.0
0.2
0.4
0.6
0.8
1.0
disease

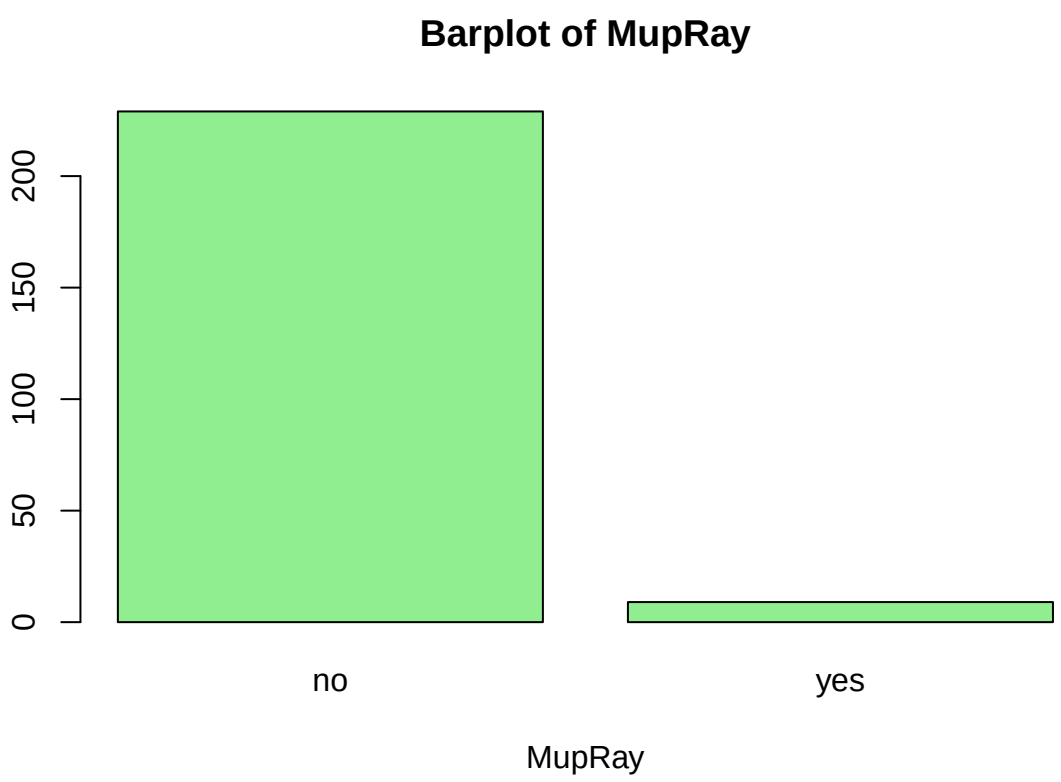

Histogram of age
Density
0.0
0.2
0.4
0.6
0.8
0
5
10
15
age

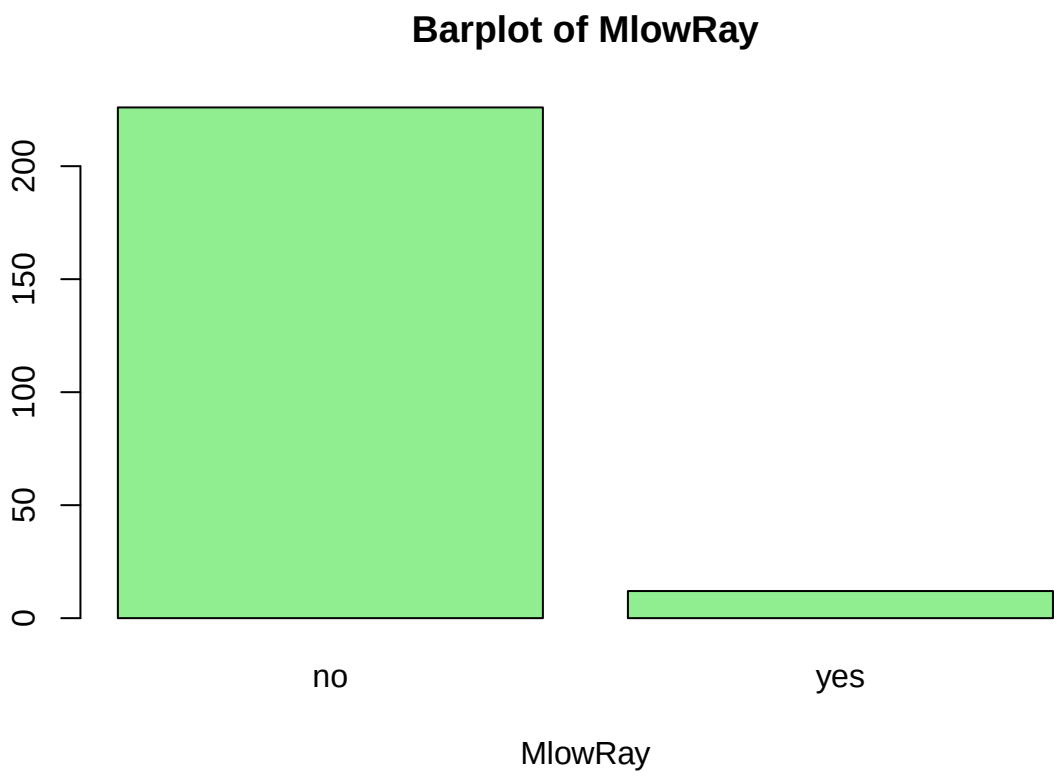

Barplot of ID
0.0
0.5
1.0
1.5
2.0
2.5
3.0
7004
7024
7043
7067
7085
7111
7132
7152
7186
7206
ID

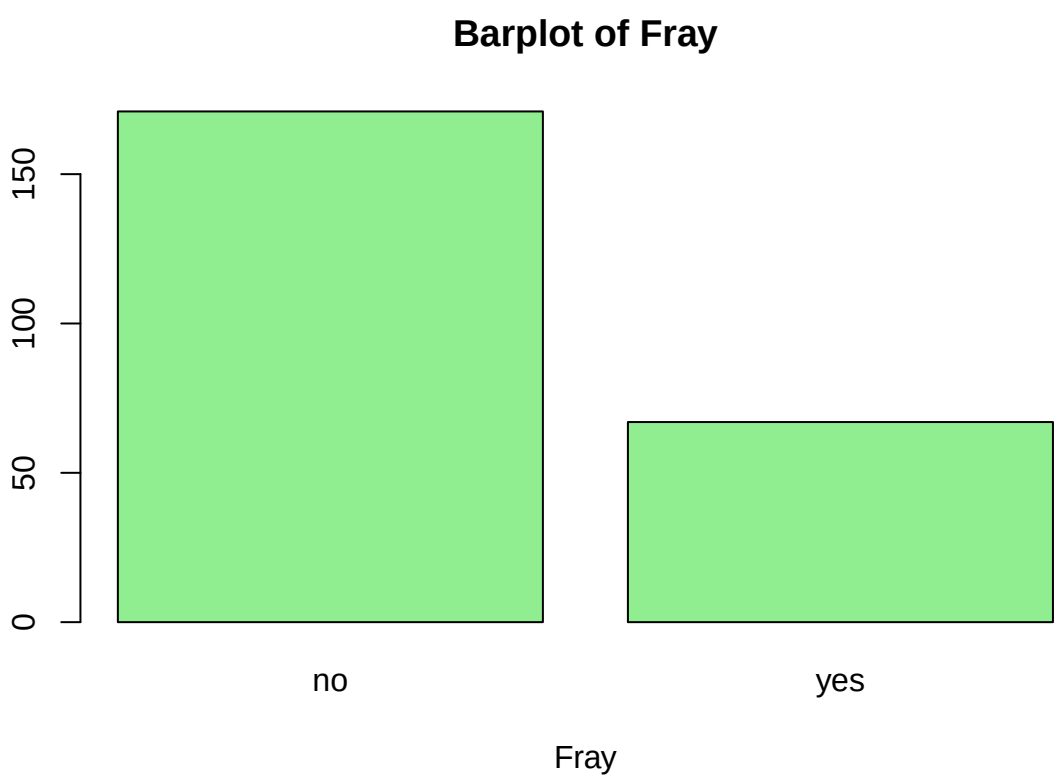

Barplot of Sex
0
20
40
60
80
100
120
F
M
Sex

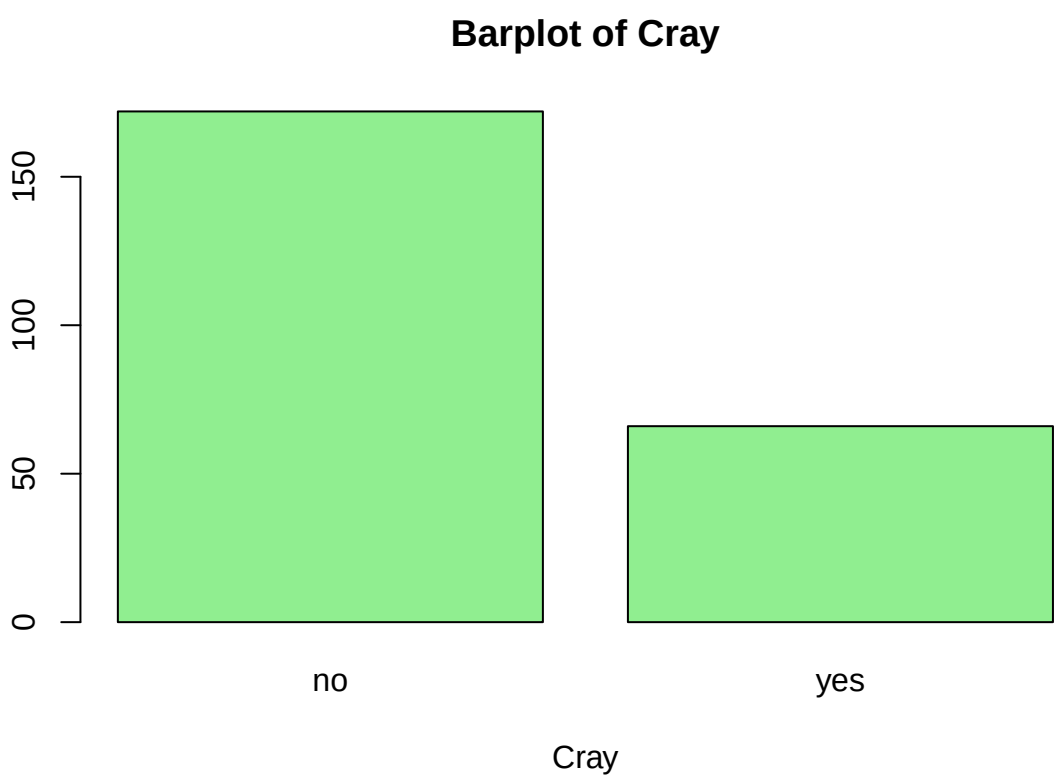

Barplot of downs
0
50
100
150
200
no
yes
downs

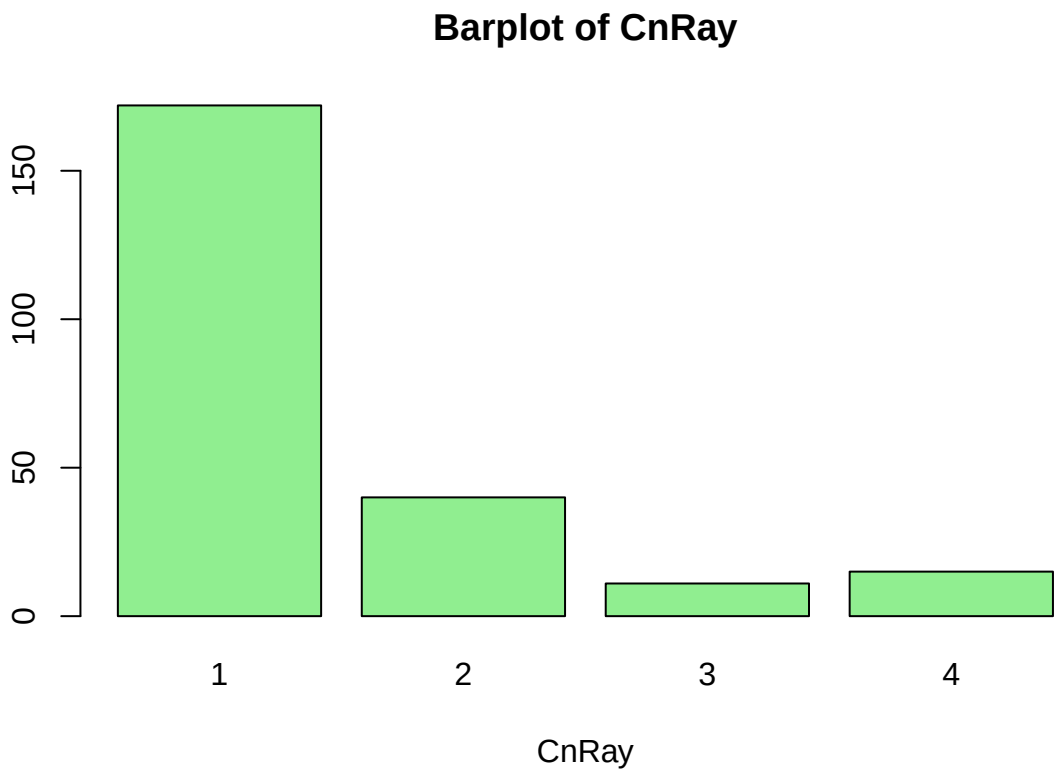

Barplot of Mray
0
50
100
150
200
no
yes
Mray


Barplot of MupRay
0
50
100
150
200
no
yes
MupRay


Barplot of MlowRay
0
50
100
150
200
no
yes
MlowRay


Barplot of Fray
150
100
50
0
no
yes
Fray


Barplot of Cray
150
100
50
0
no
yes
Cray



Exercise 3

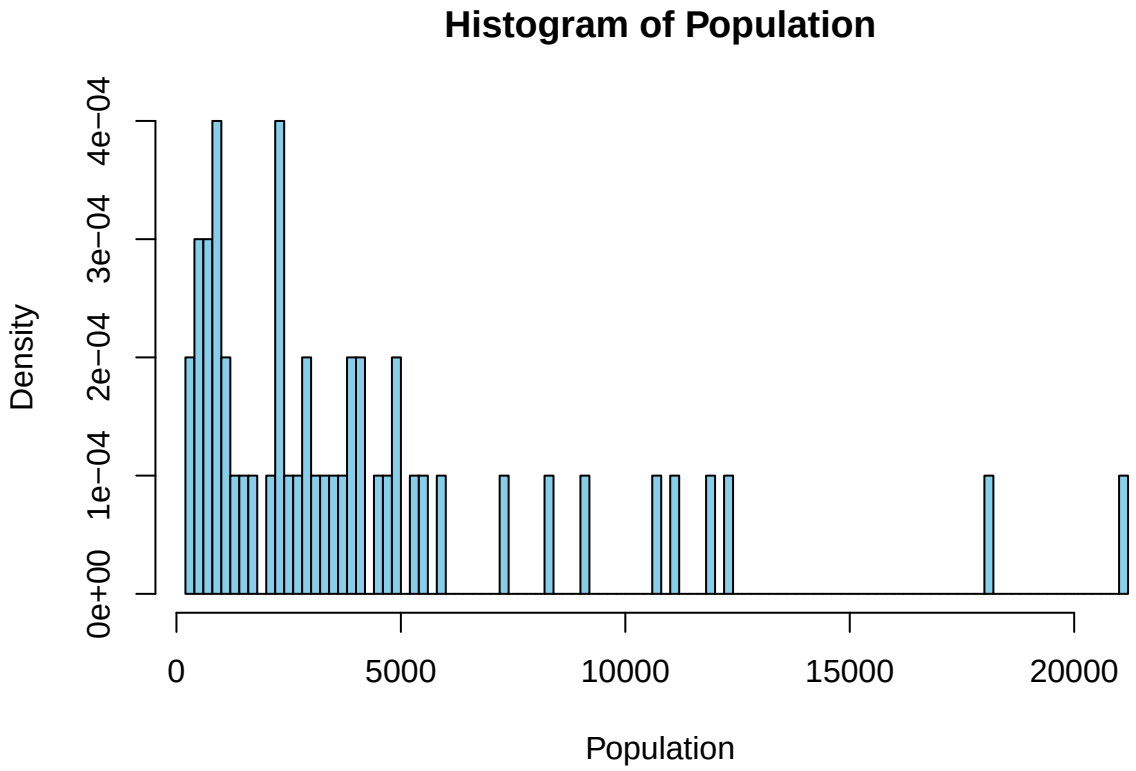
1

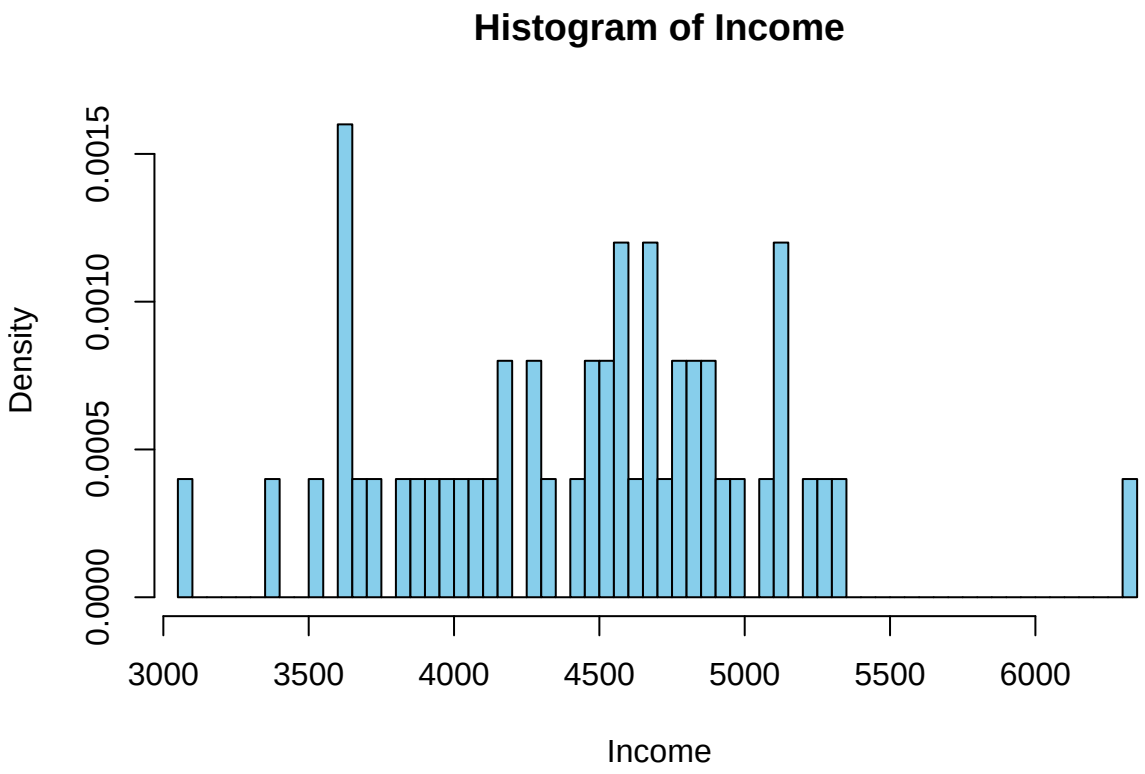
```
mean_v = apply(state.x77, 2, mean)
median_v = apply(state.x77, 2, median)
sd_v = apply(state.x77, 2, sd)
skew_v = apply(state.x77, 2, psych::skew)
sum_sta_df = data.frame(mean = mean_v, median = median_v,
    standard_deviation = sd_v, skewness = skew_v)
sum_sta_df
```

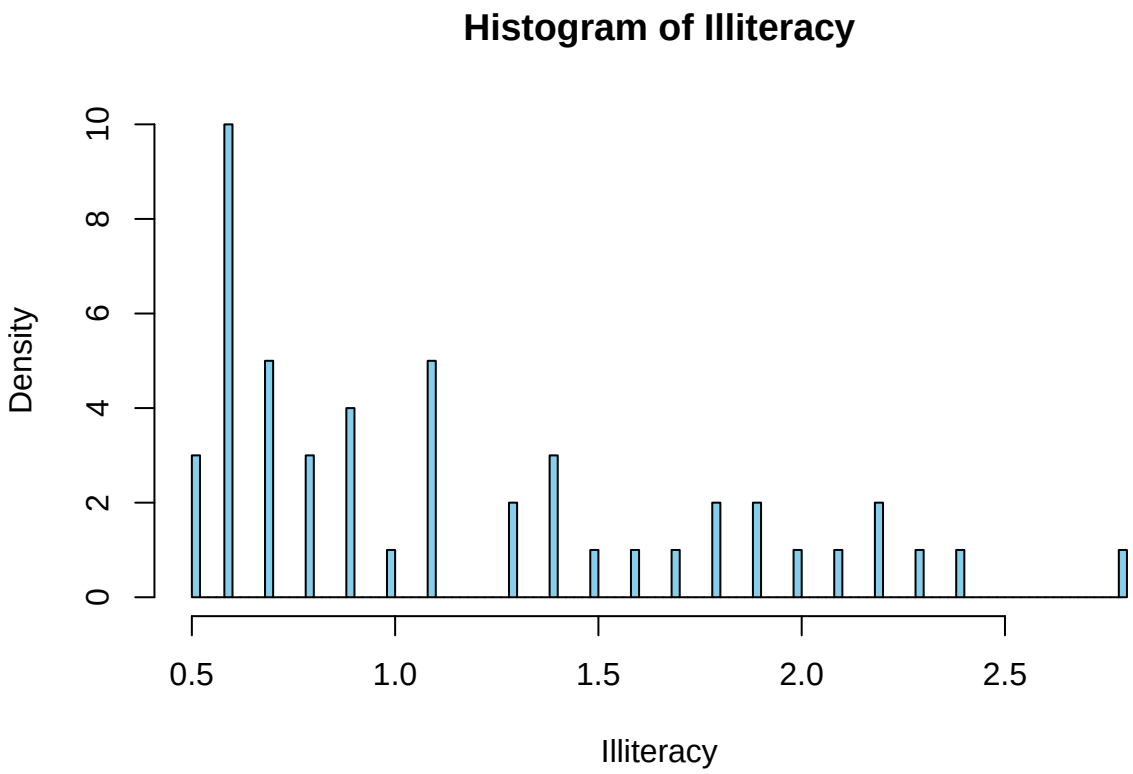
```
##                 mean    median standard_deviation   skewness
## Population  4246.4200  2838.500       4.464491e+03  1.9222511
## Income      4435.8000  4519.000       6.144699e+02  0.2046903
## Illiteracy     1.1700     0.950       6.095331e-01  0.8185809
## Life Exp      70.8786    70.675       1.342394e+00 -0.1534995
## Murder         7.3780     6.850       3.691540e+00  0.1293391
## HS Grad       53.1080    53.250       8.076998e+00 -0.3192442
## Frost        104.4600   114.500       5.198085e+01 -0.3663767
## Area       70735.8800 54277.000       8.532730e+04  4.0983574
```

2

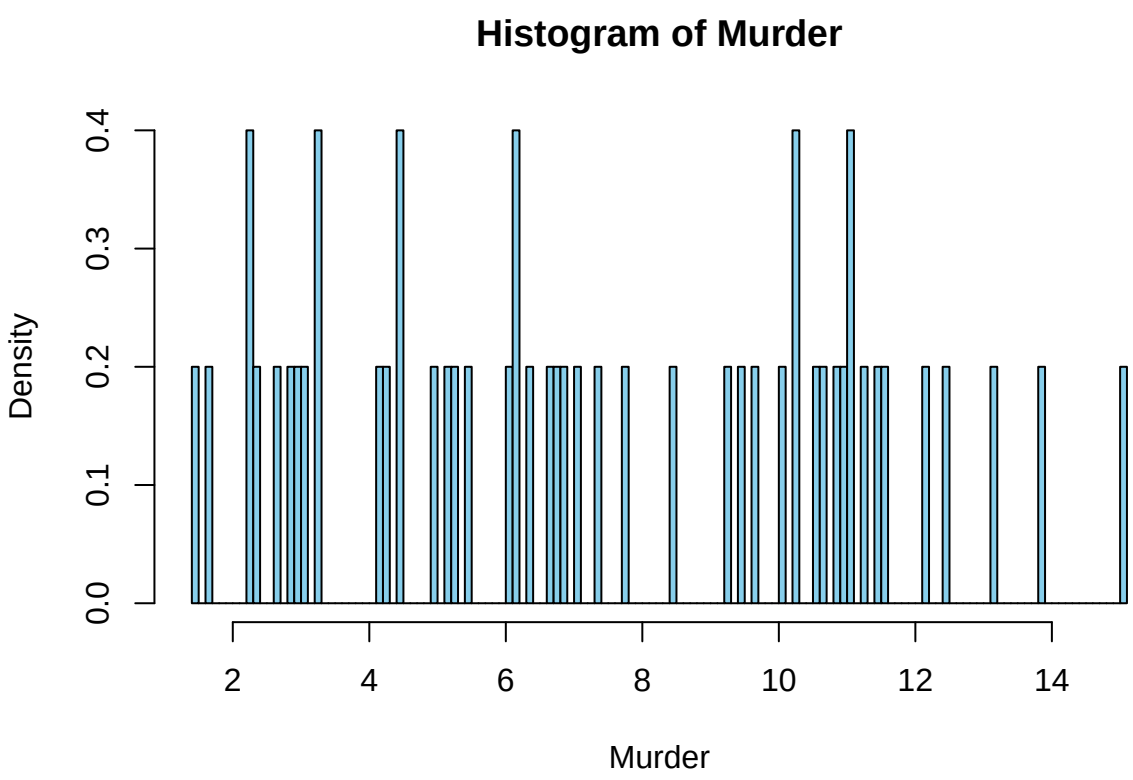
```
cns = colnames(state.x77)
for (i in 1:ncol(state.x77)) {
    i_name = cns[i]
    hist(state.x77[, i], breaks = 100, xlab = i_name, xlim = c(min(state.x77[,
        i]), max(state.x77[, i])), col = "skyblue", main = paste0("Histogram of ",
        i_name), probability = TRUE, plot = TRUE)
}
```

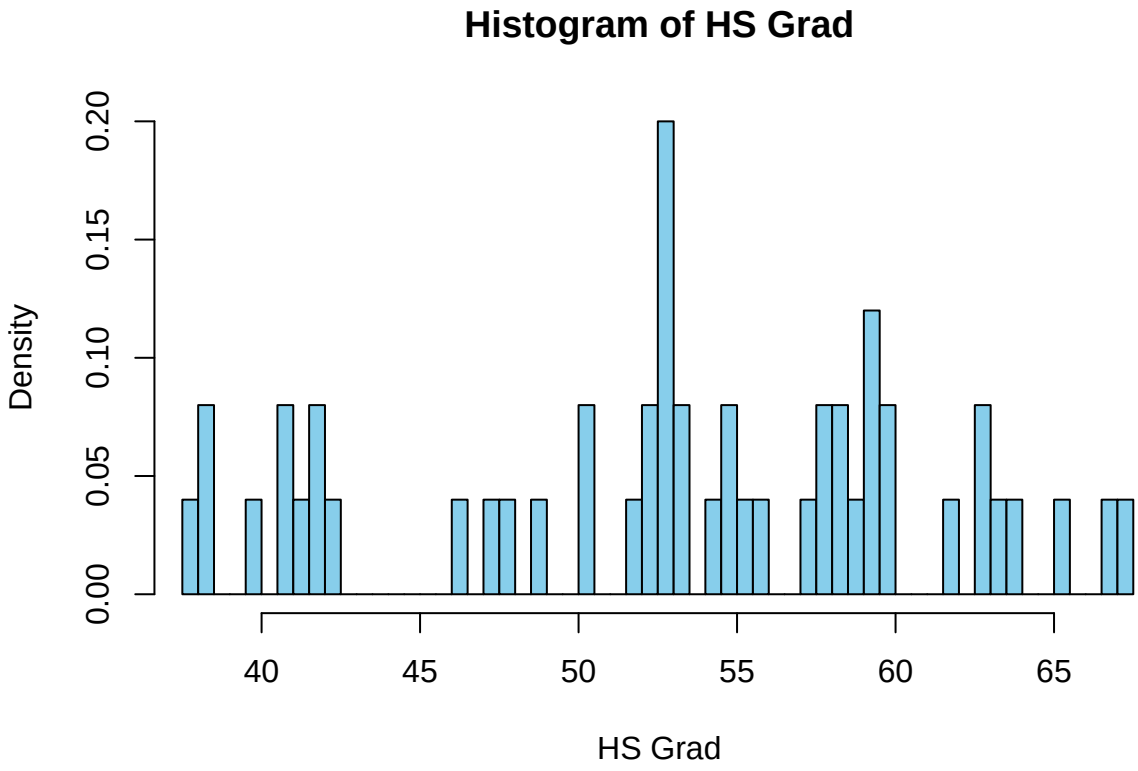


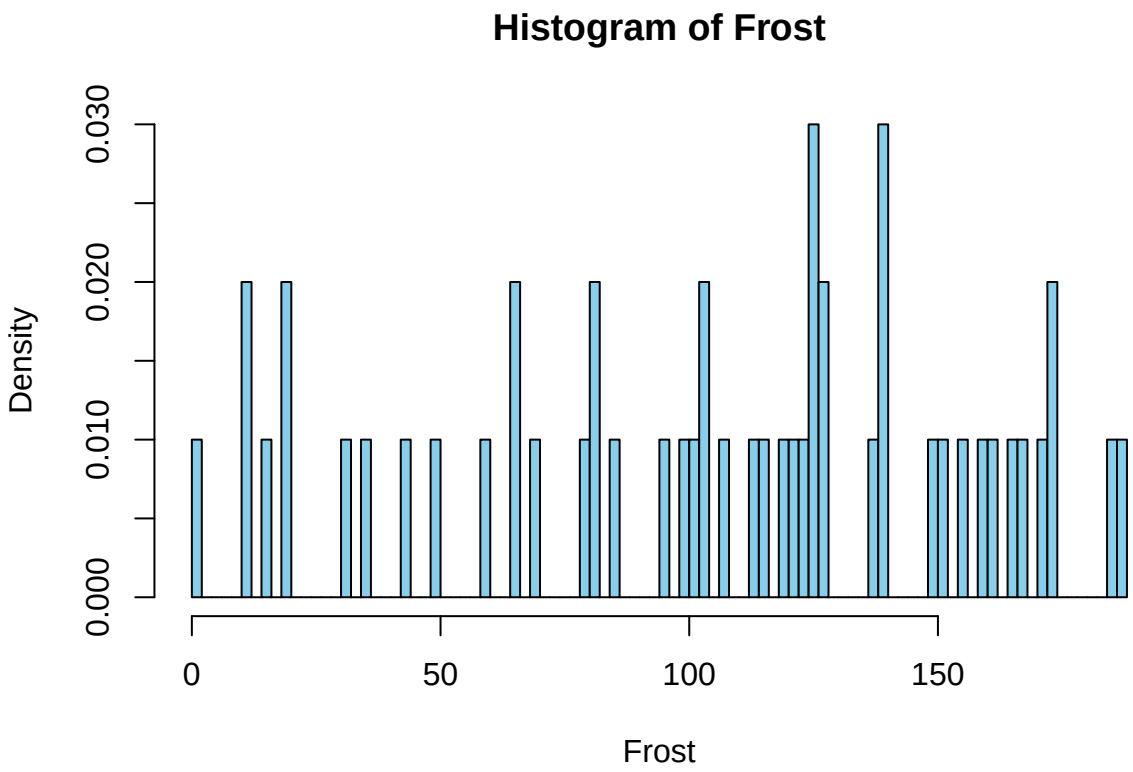

Histogram of Income
Density
0.0000
0.0005
0.0010
0.0015
3000
3500
4000
4500
5000
5500
6000
Income

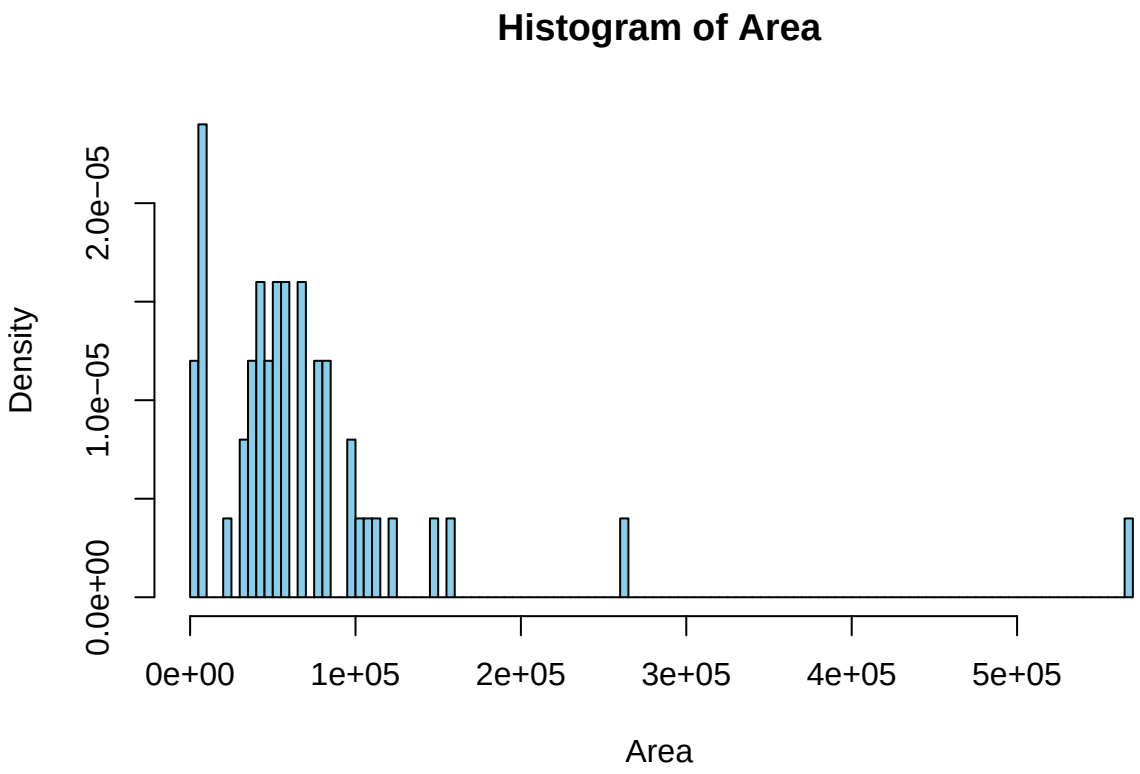

Histogram of Illiteracy
Density
0
2
4
6
8
10
0.5
1.0
1.5
2.0
2.5
Illiteracy


Histogram of Life Exp
Density
0.0
0.2
0.4
0.6
0.8
1.0
1.2
68
69
70
71
72
73
Life Exp


Histogram of Murder
Density
0.0
0.1
0.2
0.3
0.4
2
4
6
8
10
12
14
Murder


Histogram of HS Grad
Density
0.00
0.05
0.10
0.15
0.20
40
45
50
55
60
65
HS Grad


Histogram of Frost
Density
0.000
0.010
0.020
0.030
0
50
100
150
Frost

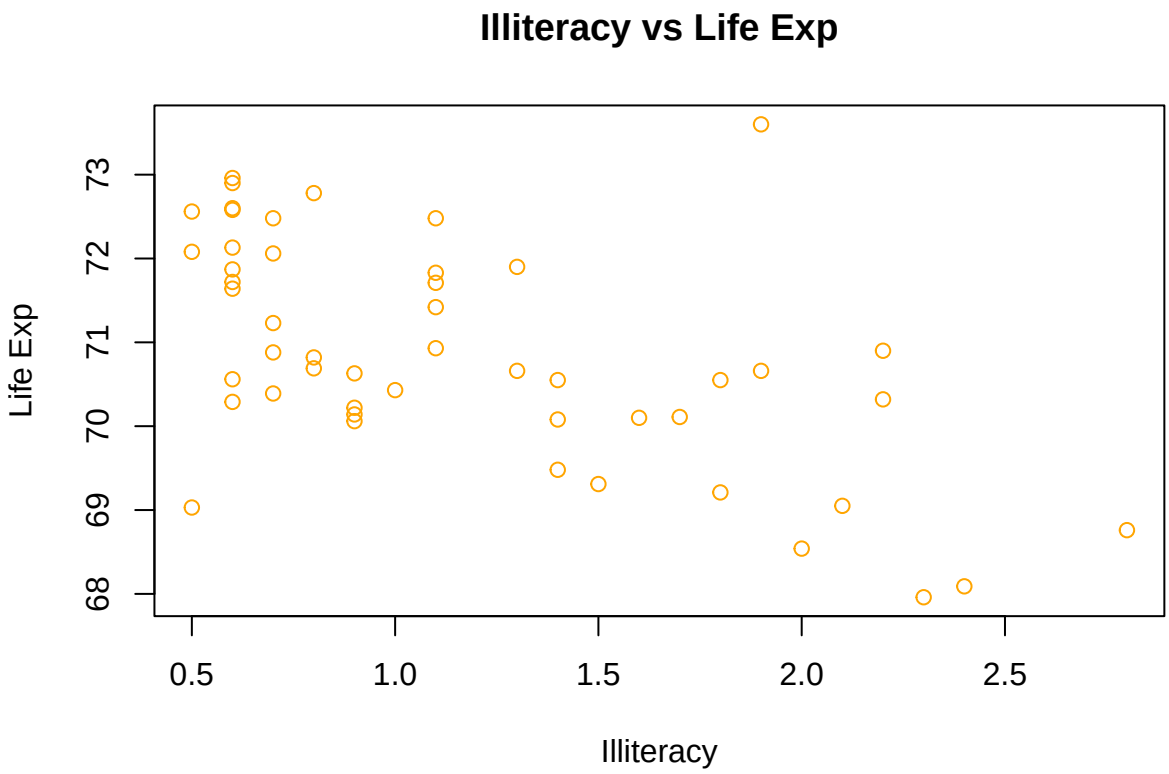


3

The Income, Life Exp and Murder are almost symmetric.Population, Illiteracy and Area are positive skewed. HS Grad and Frost are negative skewed.

4

```
plot(x = state.x77[, "Illiteracy"], y = state.x77[, "Life Exp"],
    type = "p", col = "orange", xlab = "Illiteracy", ylab = "Life Exp",
    main = "Illiteracy vs Life Exp")
```



With the increase of Illiteracy rate, the life expectancy decreases.

5

```
cor.test(x = state.x77[, "Illiteracy"], y = state.x77[,
    "Life Exp"], alternative = "two.sided", method = "pearson")
```

```
##
##  Pearson's product-moment correlation
##
## data:  state.x77[, "Illiteracy"] and state.x77[, "Life Exp"]
## t = -5.0427, df = 48, p-value = 6.969e-06
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.7448226 -0.3708811
## sample estimates:
##       cor
## -0.5884779
```

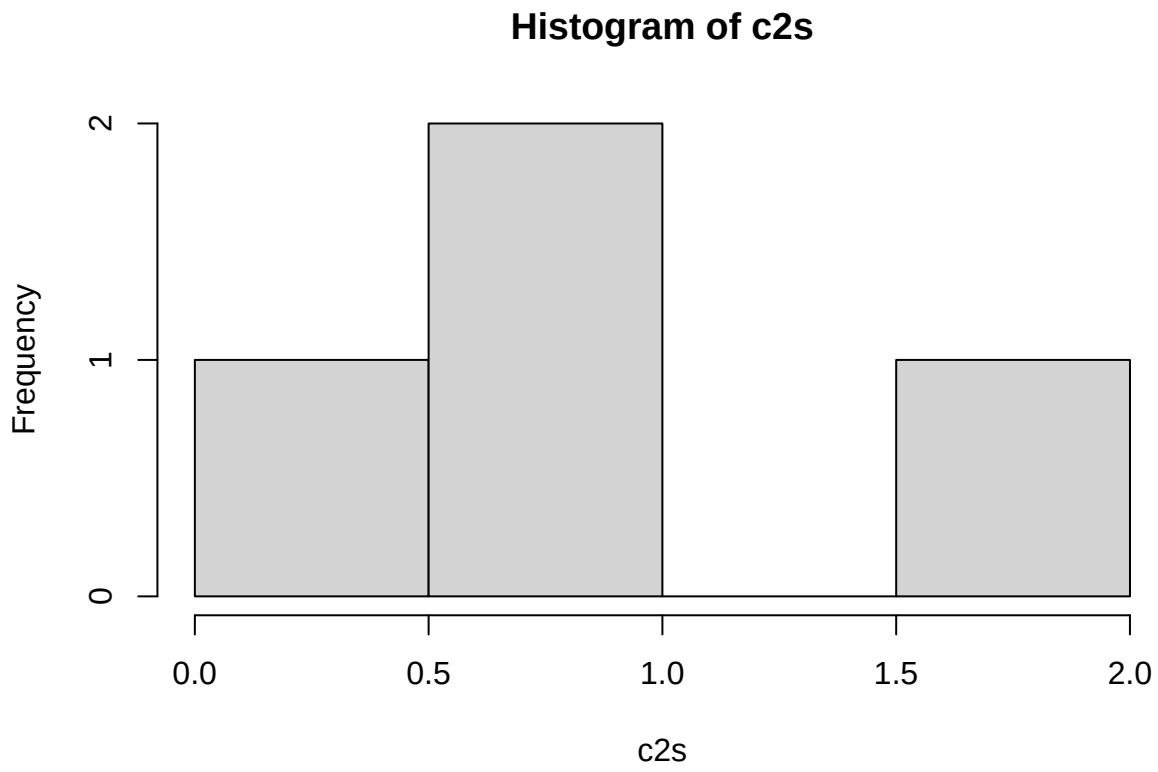
Lower Illiteracy rate means higher education level, which will improve people's life quality and increase life expectancy.

Exercise 4

```
c1 = c(0, 1)
c2 = expand.grid(c1, c1)
c2s = rowSums(c2)
```

1

```
hist(c2s, breaks = 3, main = "Histogram of c2s")
```



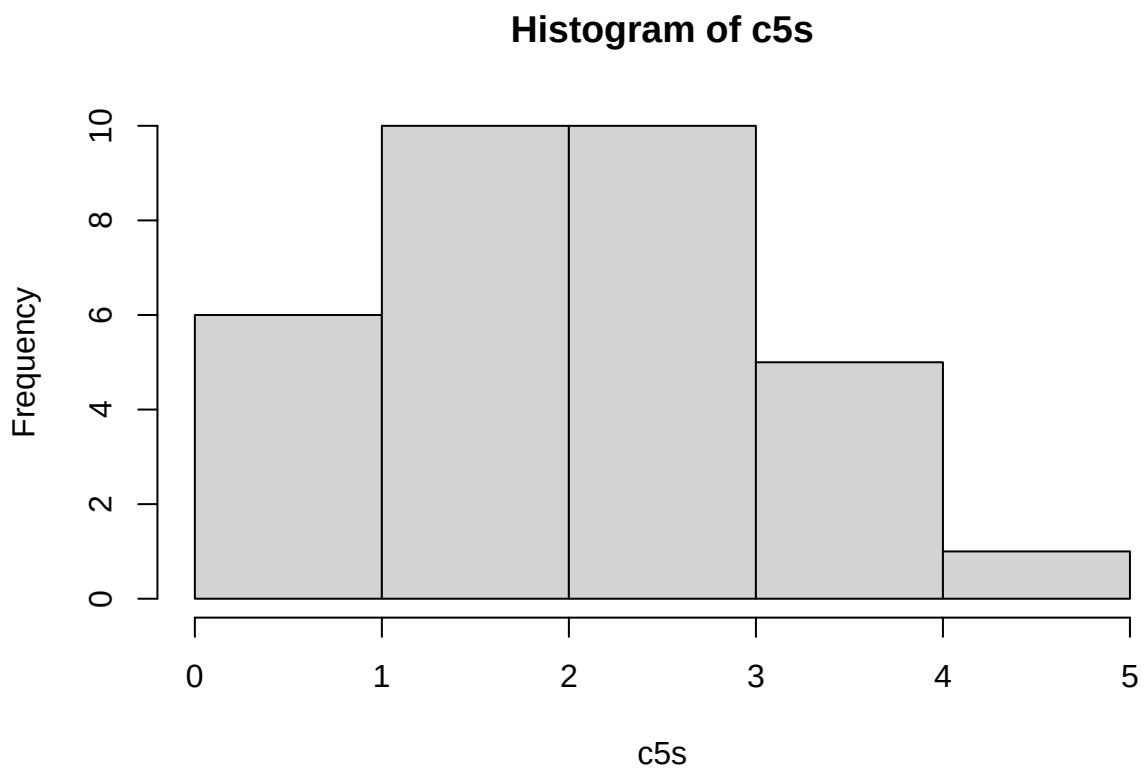

```
dim(c2)
```

```
## [1] 4 2
```

`expand.grid()` lists all combinations from given vectors. `rowSums()` lists the sum of each combination from given vectors.

2

```
c5 = expand.grid(c1, c1, c1, c1, c1)
c5s = rowSums(c5)
hist(c5s, breaks = 6, main = "Histogram of c5s")
```



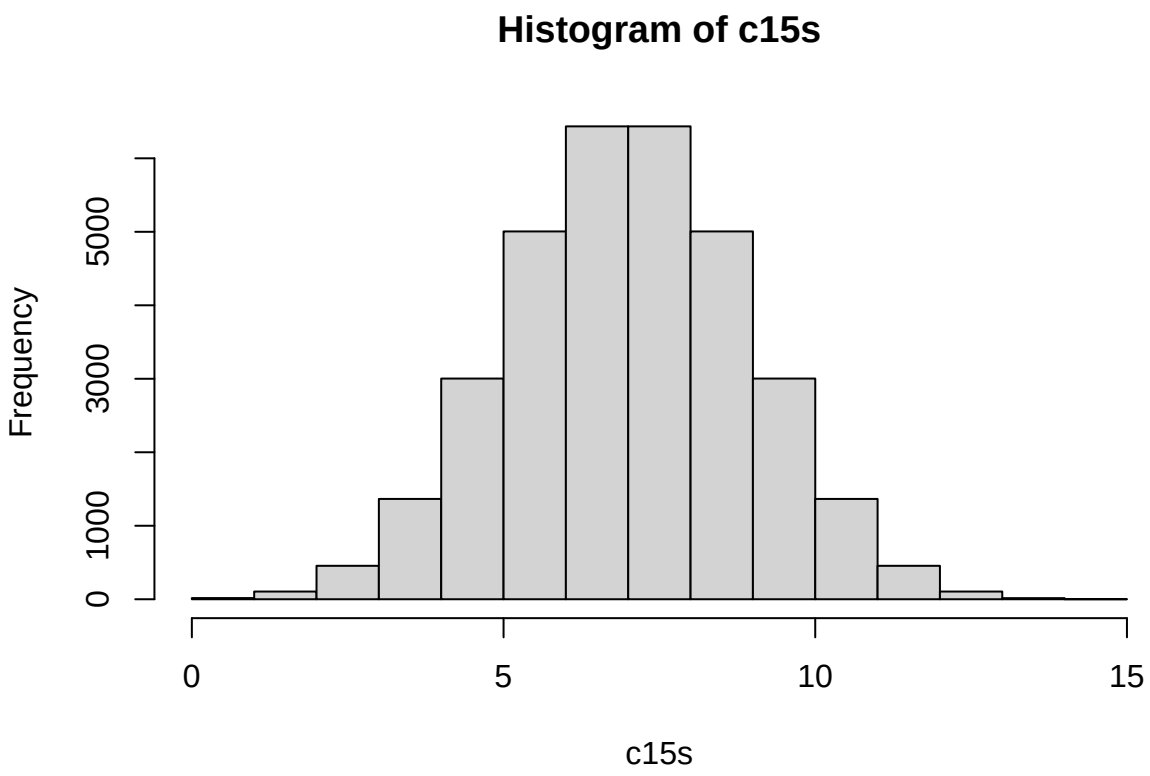

```
dim(c5)
```

```
## [1] 32  5
```

The distribution of c5s gets a peak in center.

3

```
c15 = expand.grid(replicate(15, c1, simplify = F))
c15s = rowSums(c15)
hist(c15s, breaks = 16, main = "Histogram of c15s")
```




```
dim(c15) == c(2^15, 15)
```

```
## [1] TRUE TRUE
```

4

```
sta_ls = list(range = range(c15s), mean = mean(c15s), std = sd(c15s))
sta_ls
```

```
## $range
## [1]  0 15
##
## $mean
## [1] 7.5
##
## $std
## [1] 1.936521
```

```
c(sta_ls$mean, 15 * 0.5)
```

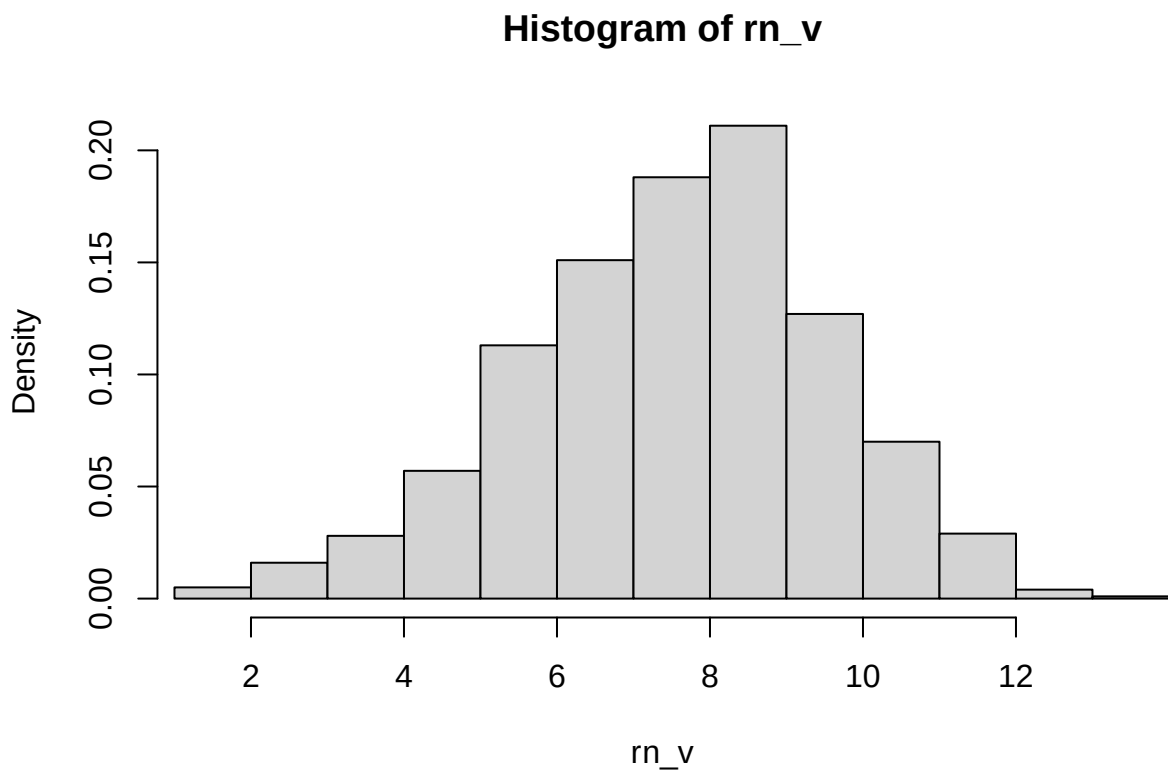
```
## [1] 7.5 7.5
```

```r
c(sta_ls$std, (15 * 0.5 * 0.5)^0.5)
```

```
## [1] 1.936521 1.936492
```

5

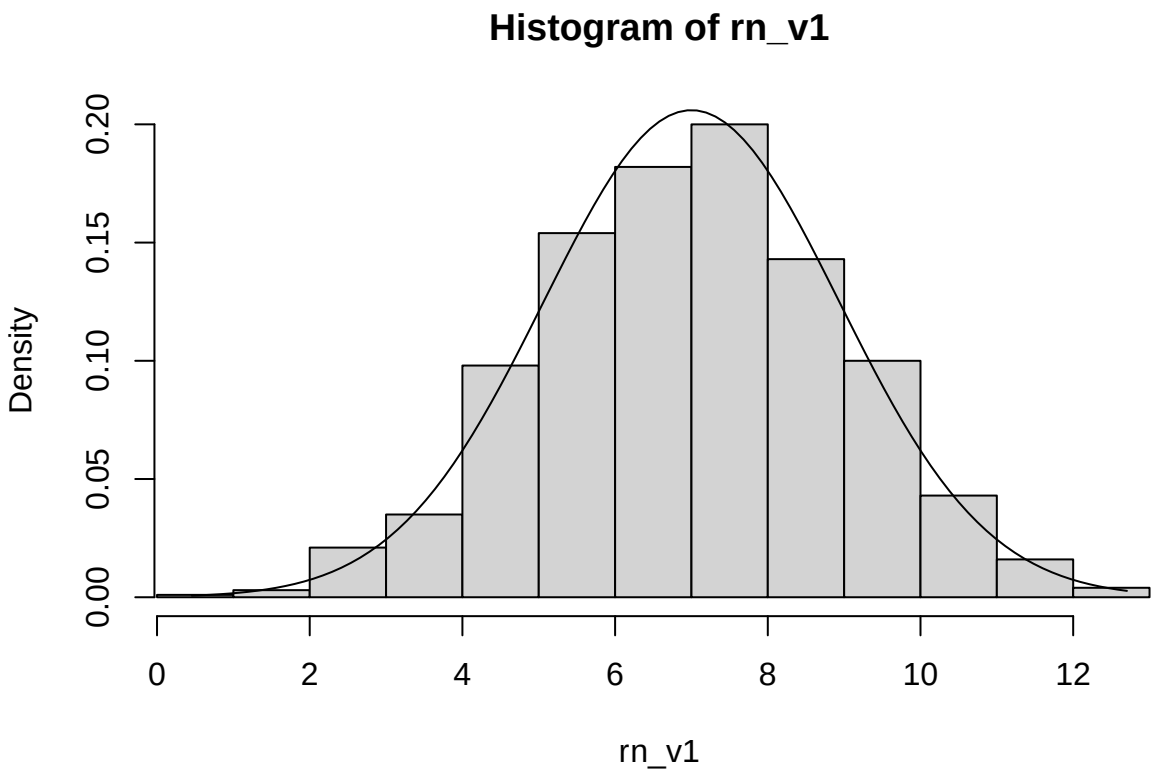
```r
rn_v = rnorm(1000, mean = sta_ls$mean, sd = sta_ls$std)
hist(rn_v, breaks = 16, xlim = c(min(rn_v), max(rn_v)),
    main = "Histogram of rn_v", probability = TRUE)
```



The distribution of 10^3 random numbers is similar to the distribution of c15s.

6

```r
rn_v1 = rnorm(1000, mean = sta_ls$mean - 0.5, sd = sta_ls$std)
hist(rn_v1, breaks = 16, xlim = c(min(rn_v1), max(rn_v1)),
    main = "Histogram of rn_v1", probability = TRUE)
curve(expr = dnorm(x, sta_ls$mean - 0.5, sta_ls$std), add = TRUE)
```



Exercise 5

```
fib = c(1, 1)
i = 2
fib_sum = sum(fib)
while (fib_sum < 22000) {
    i = i + 1
    fib[i] = fib[i - 2] + fib[i - 1]
    fib_sum = sum(fib)
}
length(fib)
```

```
## [1] 21
```