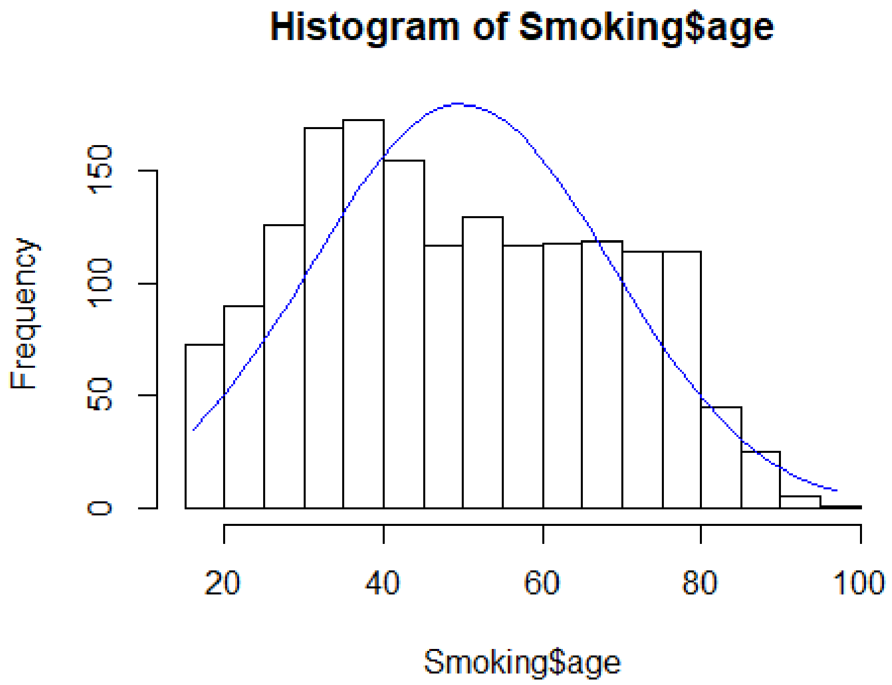
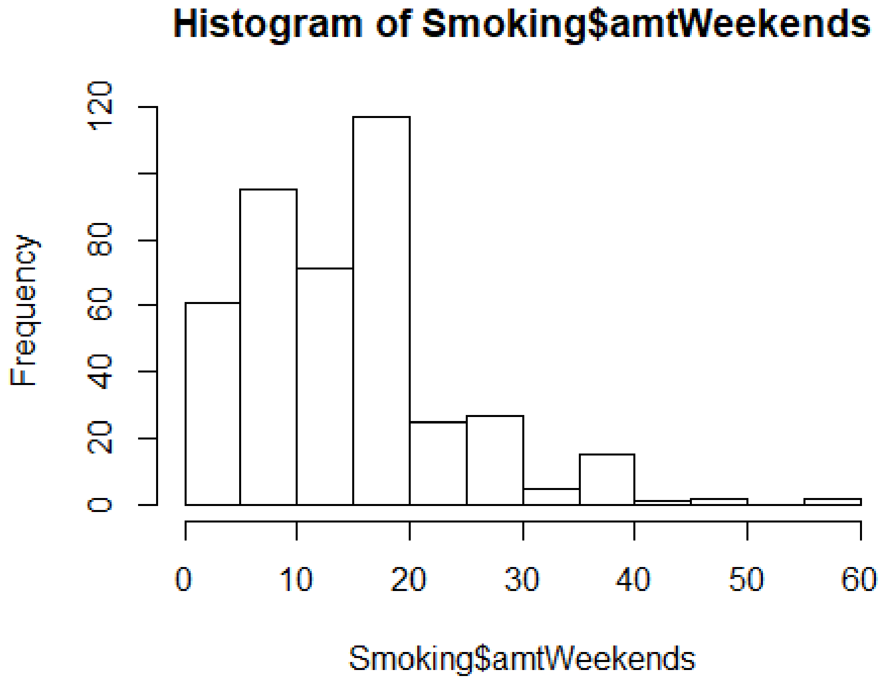
Description.R

```r
# install readxl package first
library(readxl)
Smoking<-read_excel("Smoking.xlsx", na="NA", col_names = TRUE)

# Some basic descriptive capabilities in the main R package
  # Numerical desciption
summary(Smoking)
##      record           sex                 age        maritalStatus
##  Min.   :   1.0   Length:1691        Min.   :16.00   Length:1691
##  1st Qu.: 423.5   Class :character   1st Qu.:34.00   Class :character
##  Median : 846.0   Mode  :character   Median :48.00   Mode  :character
##  Mean   : 846.0                      Mean   :49.84
##  3rd Qu.:1268.5                      3rd Qu.:65.50
##  Max.   :1691.0                      Max.   :97.00
##
##  grossIncome           region             smoke            amtWeekends
##  Length:1691        Length:1691        Length:1691        Min.   : 0.00
##  Class :character   Class :character   Class :character   1st Qu.:10.00
##  Mode  :character   Mode  :character   Mode  :character   Median :15.00
##                                                           Mean   :16.41
##                                                           3rd Qu.:20.00
##                                                           Max.   :60.00
##                                                           NA's   :1270
##   amtWeekdays
##  Min.   : 0.00
##  1st Qu.: 7.00
##  Median :12.00
##  Mean   :13.75
##  3rd Qu.:20.00
##  Max.   :55.00
##  NA's   :1270

mean(Smoking$amtWeekends, na.rm=T)
## [1] 16.41093
```

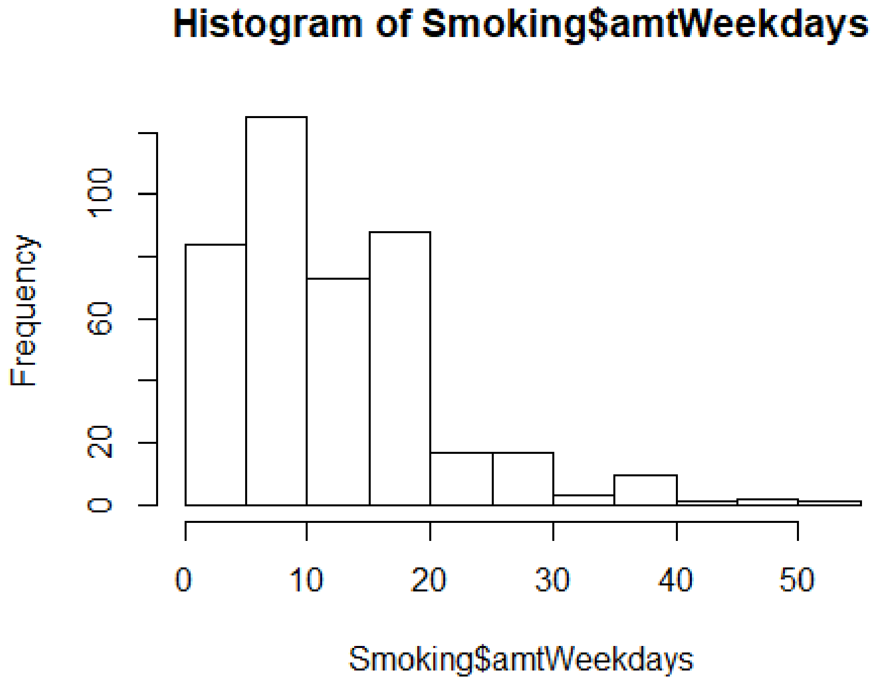
```
  # Graphical description
    # graphs for interval variables
hist(Smoking$age)

      # histogram: age, with normal curve
h <- hist(Smoking$age)
        # code to add normal curve (www.statmethods.net)
x <- Smoking$age
xfit <- seq(min(x), max(x), length = 40)
yfit <- dnorm(xfit, mean = mean(x), sd = sd(x))
yfit <- yfit*diff(h$mids[1:2])*length(x)
lines(xfit, yfit, col="blue")
```



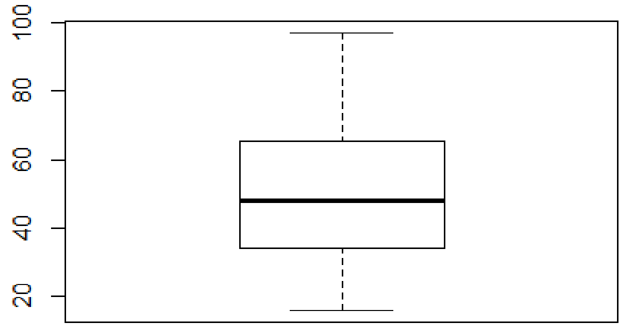
```
      # other histograms
hist(Smoking$amtWeekends)
```



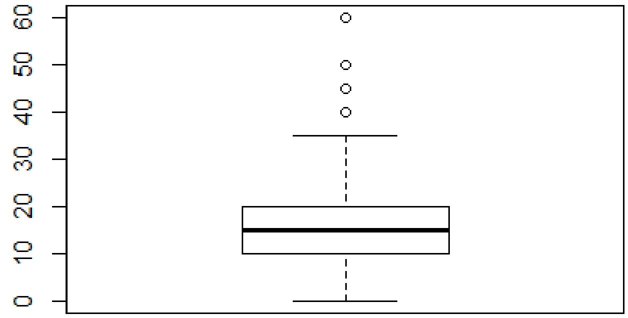
```
hist(Smoking$amtWeekdays)
```



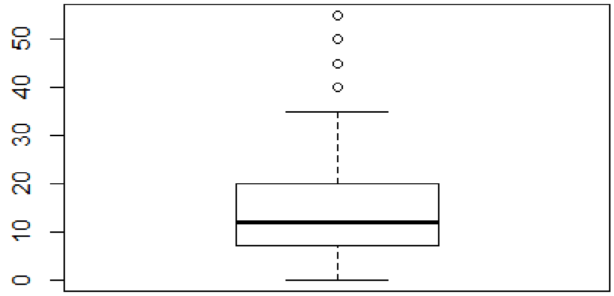
```
      # box plots
boxplot(Smoking$age)
```



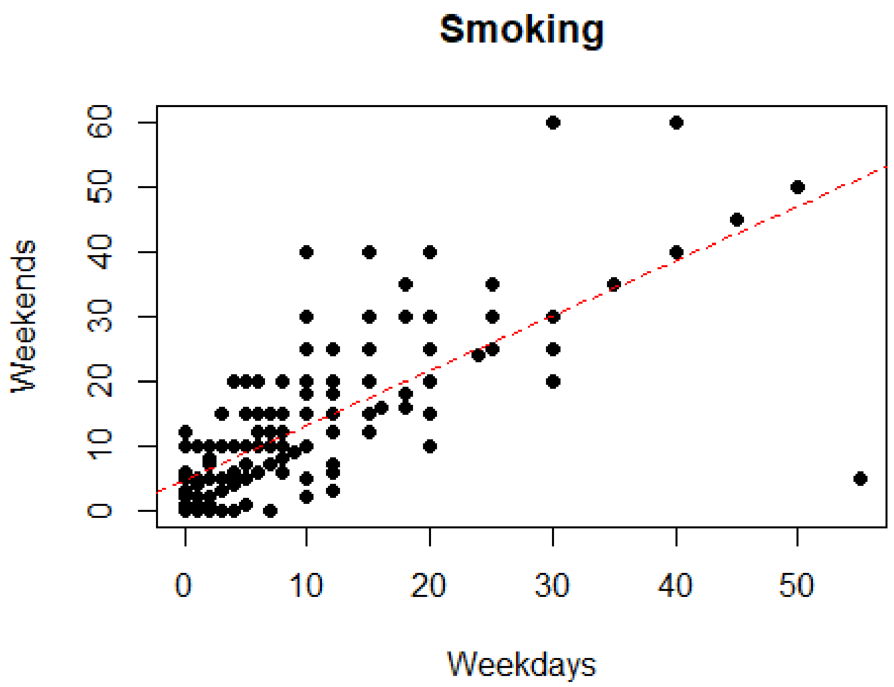
```
boxplot(Smoking$amtWeekends)
```



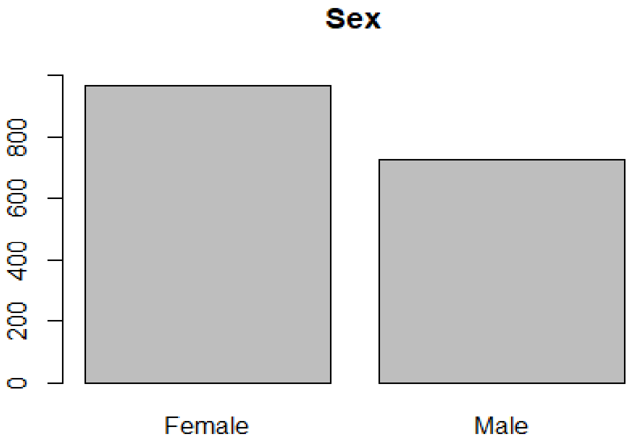
```
boxplot(Smoking$amtWeekdays)
```



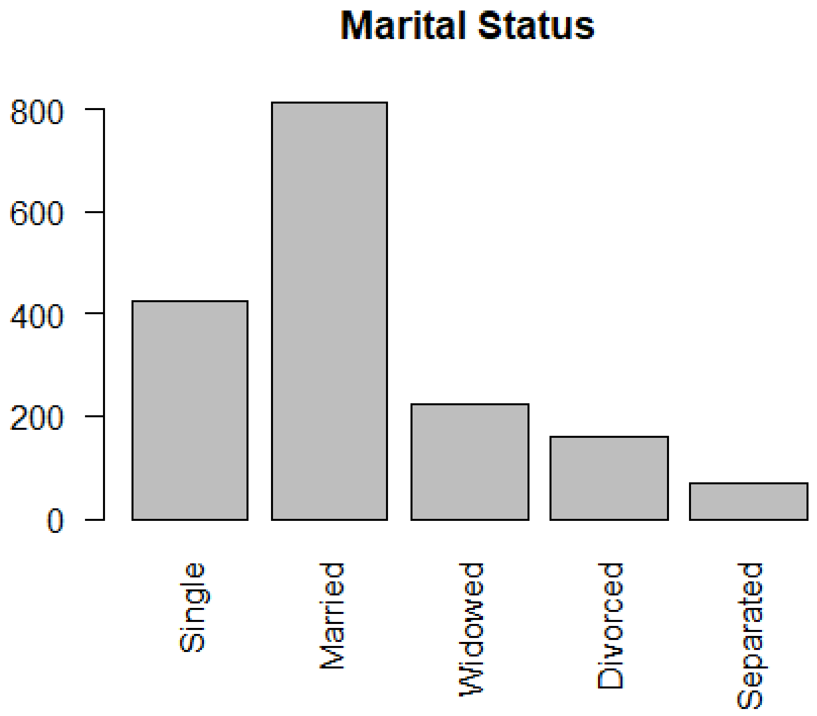
```
      # scatter plot
plot(Smoking$amtWeekdays, Smoking$amtWeekends, pch = 16, main = "Smoking",
xlab = "Weekdays", ylab = "Weekends")
abline(lm(Smoking$amtWeekends~Smoking$amtWeekdays), lty=2, col="red")
```



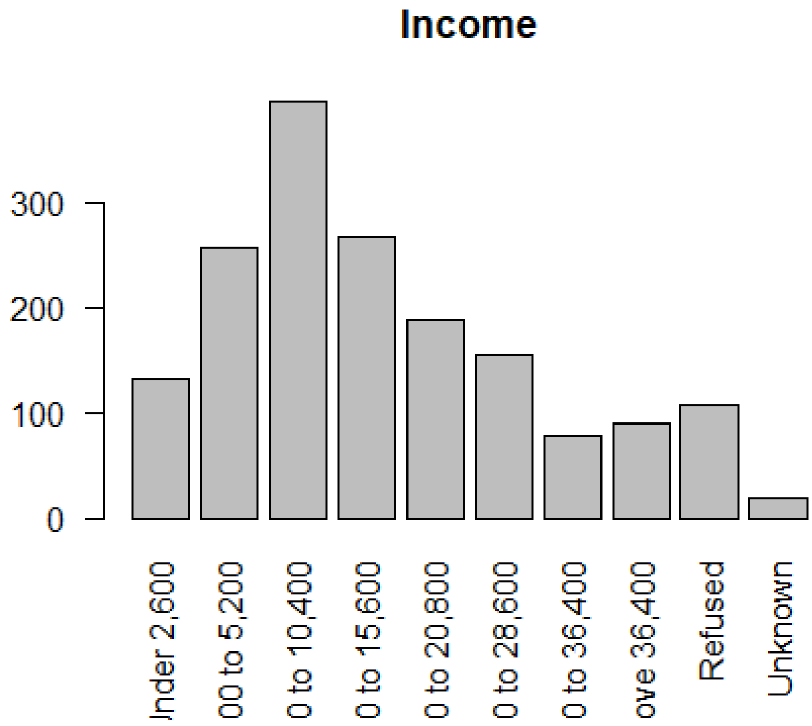
```
    # bar charts for nominal and ordinal variables
      # sex
sexCount <- table(Smoking$sex)
sexCount
##
## Female   Male
##    965    726
barplot(sexCount, ylim = c(0,1000), main="Sex")
```



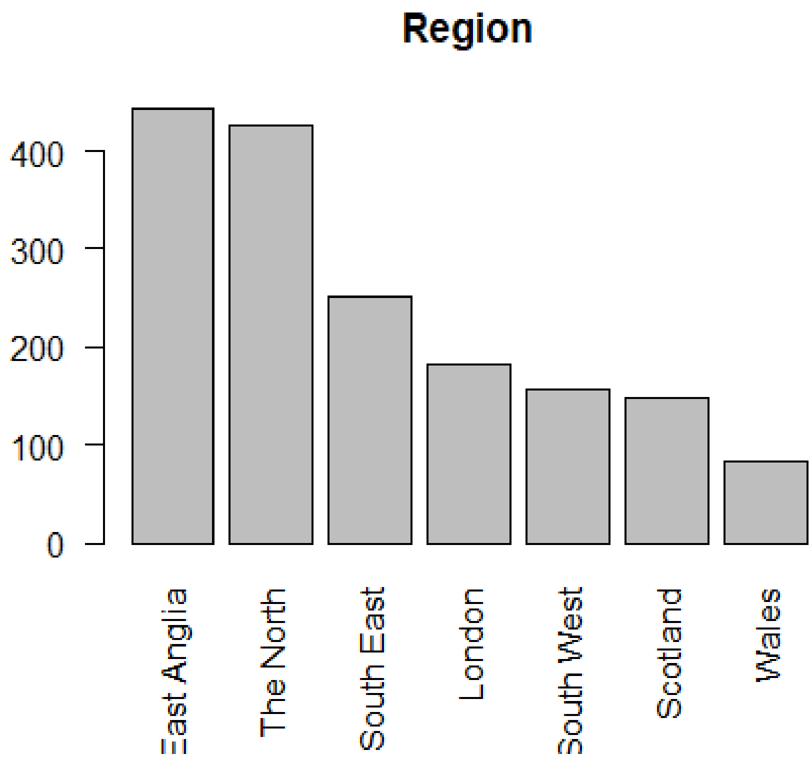
```
      # marital status
table(Smoking$maritalStatus)
##
##  Divorced   Married Separated    Single   Widowed
##       161       812        68       427       223
        # factor function to reorder the categories before graphing
maritalSort<-factor(Smoking$maritalStatus, levels = c("Single", "Married",
"Widowed","Divorced","Separated"))
maritalCount <- table(maritalSort)
        #  see the re-ordered categories:
maritalCount
## maritalSort
##    Single   Married   Widowed  Divorced Separated
##       427       812       223       161        68
        # bar chart, las=2 to make x-axis labels vertical
barplot(maritalCount, main = "Marital Status", las = 2)
```



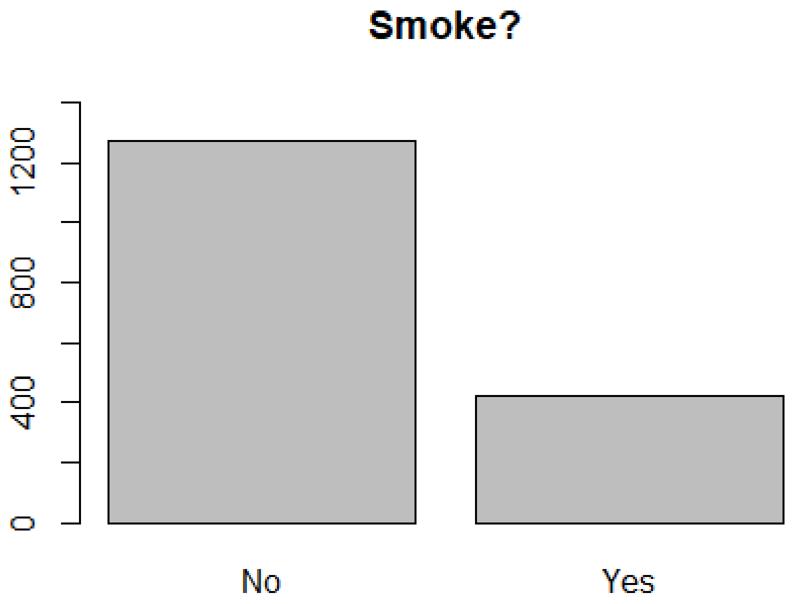
```
      # income
incomeSort<-factor(Smoking$grossIncome, levels = c("Under 2,600","2,600 to
5,200","5,200 to 10,400","10,400 to 15,600","15,600 to 20,800","20,800 to
28,600","28,600 to 36,400","Above 36,400","Refused","Unknown"))
incomeCount <- table(incomeSort)
incomeCount
## incomeSort
##      Under 2,600   2,600 to 5,200  5,200 to 10,400 10,400 to 15,600
##              133              257              396              268
## 15,600 to 20,800 20,800 to 28,600 28,600 to 36,400     Above 36,400
##              188              155               79               89
##          Refused          Unknown
##              108               18
barplot(incomeCount, main = "Income", las = 2)
```



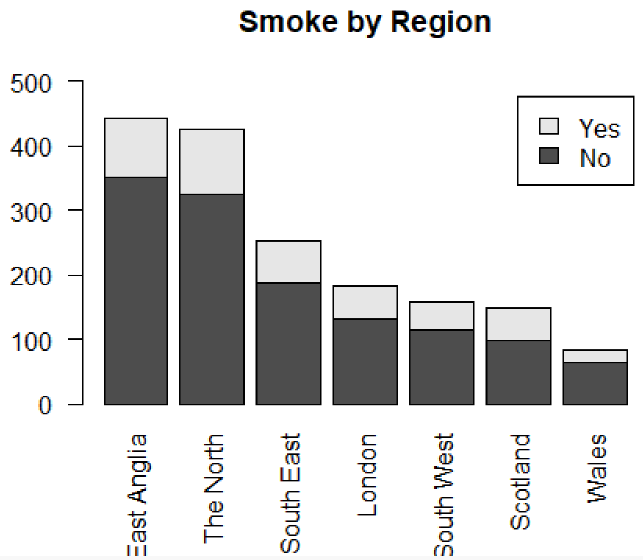
```
      # region
regionSort<-factor(Smoking$region, levels = c("Midlands & East Anglia","The
North","South East","London","South West","Scotland","Wales"))
regionCount <- table(regionSort)
regionCount
## regionSort
## Midlands & East Anglia              The North             South East
##                    443                    426                    252
##                 London             South West               Scotland
##                    182                    157                    148
##                  Wales
##                     83
barplot(regionCount, main = "Region", las = 2)
```



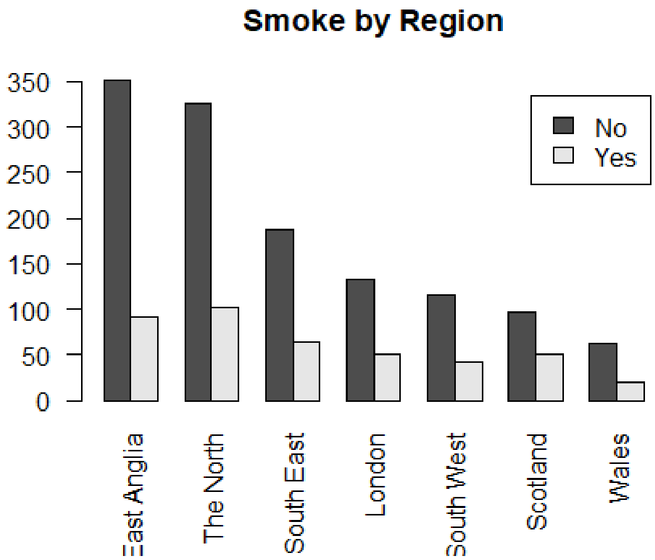
```
      # smoke: yes or no
smokeCount <- table(Smoking$smoke)
smokeCount
##
##   No  Yes
## 1270  421
barplot(smokeCount, ylim = c(0,1400),main = "Smoke?")
```



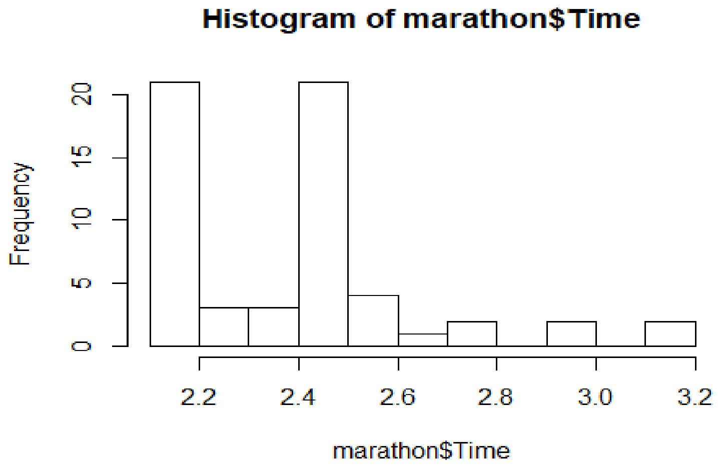
```
      # stacked bar chart: smoke by region
smoke_regionCount<-table(Smoking$smoke, regionSort)
smoke_regionCount
##        regionSort
##         Midlands & East Anglia The North South East London South West
##     No                     351       325        187    132        115
##     Yes                     92       101         65     50         42
##        regionSort
##         Scotland Wales
##     No        97    63
##     Yes       51    20
barplot(smoke_regionCount, main="Smoke by Region", las = 2, ylim = c(0, 500),
legend = rownames(smoke_regionCount))
```



```
      # grouped bar chart: smoke by region
barplot(smoke_regionCount, main="Smoke by Region", las = 2, legend =
rownames(smoke_regionCount), beside = T)
```



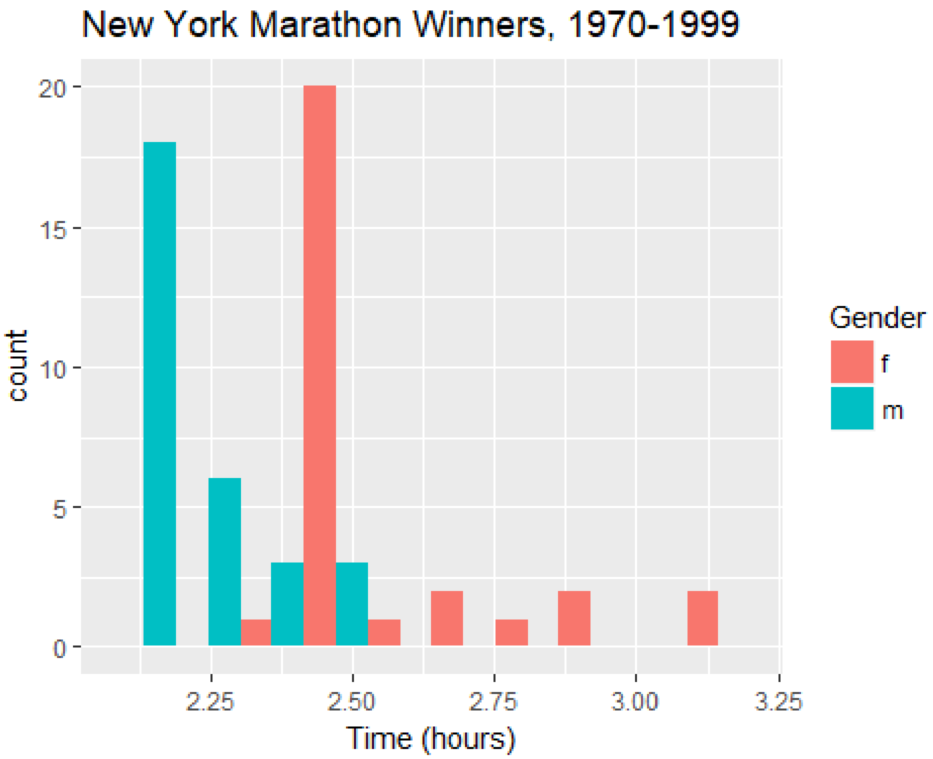
```r
marathon <-  read.table("marathon.csv", header = TRUE, sep = ",", strip.white
= TRUE)
hist(marathon$Time, breaks = 10)
```



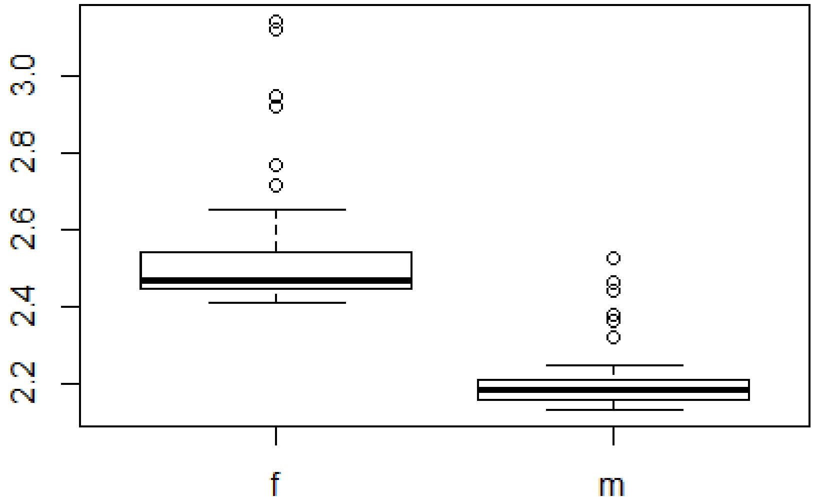
```r
# Some other descriptive capabilities in other packages
  # Numerical description
    #install pastecs package first
library(pastecs)
## Warning: package 'pastecs' was built under R version 3.4.4
    #useful function within pastecs package
stat.desc(Smoking[,c('age','amtWeekends','amtWeekdays')])
##                       age  amtWeekends  amtWeekdays
## nbr.val      1.691000e+03  421.0000000  421.0000000
## nbr.null     0.000000e+00    6.0000000   16.0000000
## nbr.na       0.000000e+00 1270.0000000 1270.0000000
## min          1.600000e+01    0.0000000    0.0000000
## max          9.700000e+01   60.0000000   55.0000000
## range        8.100000e+01   60.0000000   55.0000000
## sum          8.427300e+04 6909.0000000 5789.0000000
## median       4.800000e+01   15.0000000   12.0000000
## mean         4.983619e+01   16.4109264   13.7505938
## SE.mean      4.556431e-01    0.4821547    0.4575574
## CI.mean.0.95 8.936841e-01    0.9477370    0.8993877
## var          3.510696e+02   97.8712137   88.1400294
## std.dev      1.873685e+01    9.8929881    9.3882921
## coef.var     3.759688e-01    0.6028294    0.6827554
```

```
  # Graphical description
    # install ggplot2 package first
    # ggplot2 is a popular package with a lot of capabilities for creating
better looking graphics
library(ggplot2)
## Warning: package 'ggplot2' was built under R version 3.4.4
```

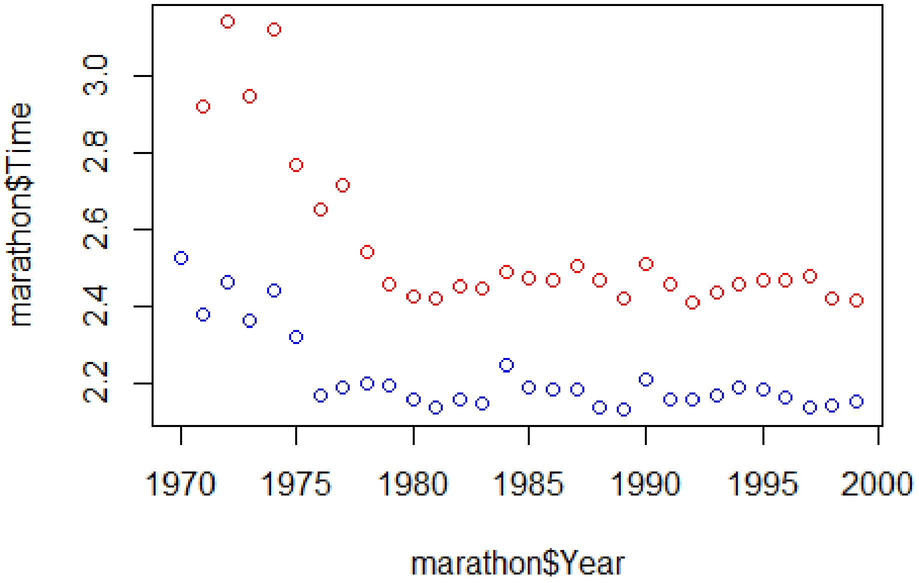
```
    # ggplot is a function within the popular ggplot2 package
    # ggplot() is used to construct a plot incrementally, using the +
operator to add layers to the existing ggplot object
      # Histogram of Times by Gender
Gender <- marathon$Gender
ggplot(marathon, aes(x=marathon$Time, fill = Gender)) +
geom_histogram(position = "dodge", bins = 10) + xlab("Time (hours)") +
ggtitle("New York Marathon Winners, 1970-1999")
```



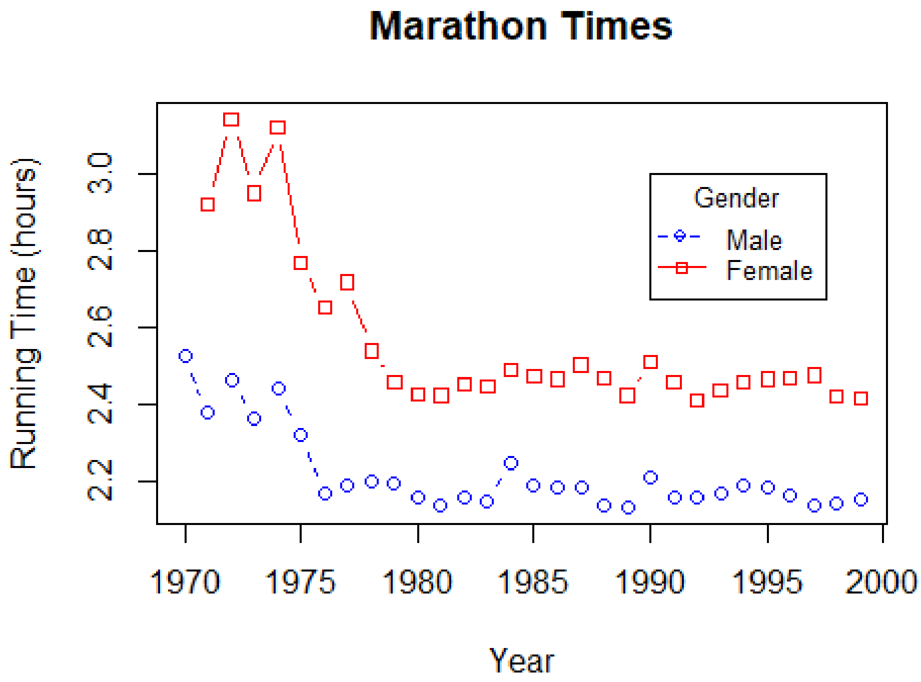
```
# Returning to the techniques in the main R package
  # Scatter Plot of Times by Gender, Year
boxplot(marathon$Time~marathon$Gender)
```



```
plot(marathon$Year, marathon$Time, col=c("red","blue")[marathon$Gender])
```



```
  # line chart
    # reorder Marathon data frame by year
marathon<-marathon[order(marathon$Year),]
marathon
##    Year Gender    Time
## 41 1970      m 2.52722
## 42 1971      m 2.38167
## 51 1971      f 2.92278
## 43 1972      m 2.46444
## 52 1972      f 3.14472
…


    # plot set up
plot(marathon$Year, marathon$Time, type = "n",
col=c("red","blue")[marathon$Gender], xlab = "Year", ylab = "Running Time
(hours)")
    # add lines and points
LineF <- subset(marathon, marathon$Gender=="f")
LineM <- subset(marathon, marathon$Gender=="m")
lines(LineF$Year, LineF$Time, type = "b", col = "red", pch = 22)
lines(LineM$Year, LineM$Time, type = "b", col = "blue", pch = 21, lty = 2)
    # add legend and title
title("Marathon Times")
legend(1990, 3, c("Male", "Female"), cex = .8, col=c("blue","red"), pch =
21:22, lty = 2:1, title = "Gender")
```


Marathon Times
Running Time (hours)
Year
Gender
Male
Female