```r
setwd("C:/Users/Min Seong Kim/Dropbox/R_programming/lecture")

data.set <- read.csv("county_data.csv", stringsAsFactors = FALSE)

head(data.set)
tail(data.set)
str(data.set)

data.1 <- data.set[1:111,]
tail(data.1)

data.state <- factor(data.1$State, levels=c("Alabama", "Alaska", "Arizona"))
summary(data.state) # or table(data.state)

plot(data.state)                    # If plot() function is used for a single factor, it will count frequencies (bar chart).
title(main="Number of Counties")

plot(data.state, data.1$IncomePerCap, range=1, las=1)  # If plot() is used for x=factor, y=numeric, it will produce boxplots.
boxplot(data.1$IncomePerCap~data.state, range=1, las = 2)  # How the distribution of y changes conditional on x
                                                           # whiskers extend to the most extreme data point which is no more
                                                           # than range times the interquartile range

big <- NA
big[data.1$TotalPop>70000] <- "big"
big[data.1$TotalPop<=70000] <- "small"

par(mar=c(6,4,2,1))                                        # bottom, left, up, right
boxplot(data.1$IncomePerCap~big*data.state, col=c("blue", "red"), main="Income per Capital", range=1, las = 2)

plot(data.1[,3:ncol(data.1)])
pairs(data.1[,3:ncol(data.1)])   # scatterplot matrix

# highlight outliers
data.2 <- data.set[data.set$State=="California",]
plot(data.2$TotalPop)
abline(h=10000000, lty=2)
index.1 <- which(data.2$TotalPop > 10000000)
points(index.1,data.2$TotalPop[index.1], pch=16, col="red" )

abline(h=2000000, lty=2)
index.2 <- which(data.2$TotalPop > 2000000 & data.2$TotalPop < 10000000  )
points(index.2,data.2$TotalPop[index.2], pch=16, col="blue" )

text(c(index.1,index.2),data.2$TotalPop[c(index.1,index.2)], labels = data.2$County[c(index.1,index.2)], cex=0.8)
# labels: character vector specifying the text to be written

# Adding a straight line on a plot
proportion <- data.1$Women/data.1$TotalPop
plot(proportion, data.1$Unemployment, pch=20, xlab="Women/Total Population", ylab="Unemployment rate")
linear.fit <- lm(data.1$Unemployment~proportion)
abline(linear.fit, lwd=3, col="red")

linear.fit
# You can also specify coefficient directly to use abline()
abline(a=-3, b=30, lwd=3, lty=4, col="blue")

abline(v=0.5, col="red", lwd=3, lty=2)
abline(h=15, col="blue", lwd=3, lty=1)

# hist(), image(): histogram and heatmap
```

```r
# heat.colors(), topo.colors(), etc: create a color vector
# density(): estimate density, which can be plotted

# Plotting a histogram
# To plot a histogram of a numeric vector, use hist()

ave.income <- data.set$IncomePerCap
hist(ave.income)

# Histogram options
# Several options are available as arguments to hist(), such as col, freq,
# breaks, xlab, ylab, main

hist(ave.income, col="pink", freq=TRUE) # Frequency scale, default

hist(ave.income, col="pink", freq=FALSE, # Probability scale, and more options
     breaks=seq(0,70000,by=10000), xlab="Imcome per capita", main="County Level Average
     Income per capital")

hist(ave.income, col="pink", freq=FALSE, # Probability scale, and more options
     breaks=seq(0,70000,by=50), xlab="Imcome per capita", main="County Level Average
     Income per capital")

hist(ave.income, col="pink", freq=FALSE, # Probability scale, and more options
     breaks=seq(0,70000,by=2000), xlab="Imcome per capita", main="County Level Average
     Income per capital")


# Adding a histogram to an existing plot

# To add a histogram to an existing plot (say, another histogram), use hist()
# with add=TRUE

hist(ave.income + 20000, col=rgb(0.1,0.1,0.5,0.2), # Note: Using a transparent color:
red, green, blue, alpha(degree of transparency)
     freq=FALSE, breaks=seq(0,90000,by=2000), add=TRUE)

# Adding a density curve to a histogram

# To estimate a density from a numeric vector, use density().
# This returns a list; it has components x and y, so we can actually
# call lines() directly on the returned object

hist(ave.income, col="pink", freq=FALSE, # Probability scale, and more options
     breaks=seq(0,70000,by=2000), xlab="Imcome per capita", main="County Level Average
     Income per capital")

density.est = density(ave.income, adjust=1.5) # 1.5 times the default bandwidth, try
different values

lines(density.est, lwd=3)

# Exercise
data.set <- read.csv("histogram.csv", stringsAsFactors = FALSE )
head(data.set)
data.set <- data.set[data.set$YRDATA==2013,]

# Using plot() and points() functions, highlight the 4 best schools in CT in terms of
expenditure per student (TOTALEXP/ENROLL).

data1 <- data.set[data.set$STATE == "Connecticut",]
data1 <- data1[data1$ENROLL != 0,]
exp_per_stu <- data1$TOTALEXP/data1$ENROLL

plot(exp_per_stu)

index1 <- which(exp_per_stu > 30)
points(index1,exp_per_stu[index1], pch=16, col="red" )

```

```
text(index1,exp_per_stu[index1]-1, labels = data1[index1, 3], cex=0.8)

# Make two histograms together for expenditure per student (TOTALEXP/ENROLL) of Connecticut and Alabama

head(data.set)
data.set <- data.set[data.set$ENROLL != 0,]
data.hist = data.set$TOTALEXP/data.set$ENROLL
# data.hist <- as.numeric(exp_per_stu)
hist(data.hist[data.set$STATE == "Connecticut"], col="pink", freq=FALSE,
     breaks=seq(0,40,by=2), xlab="Total Expenditure", main="Education expenditure in CT and MI")
hist(data.hist[data.set$STATE == "Alabama"], col=rgb(0.3,0.5,0.5,0.5), # Note: Using a transparent color
     freq=FALSE, breaks=seq(0,40,by=2), add=TRUE)
```