automobiles

HImanshu Pareek

August 27, 2017

Loading data into R

#accessing file  
filename <- "AUTOMOBILES.csv"  
automobiles <- read.csv(filename, na.strings = c("NA", "na", "not available", "NOT AVAILABLE", "Not Available", "?", "-", "\_", "null", "NULL"))  
head (automobiles)

## symboling normalized.losses make fuel.type aspiration  
## 1 3 NA alfa-romero gas std  
## 2 3 NA alfa-romero gas std  
## 3 1 NA alfa-romero gas std  
## 4 2 164 audi gas std  
## 5 2 164 audi gas std  
## 6 2 NA audi gas std  
## num.of.doors body.style drive.wheels engine.location wheel.base length  
## 1 two convertible rwd front 88.6 168.8  
## 2 two convertible rwd front 88.6 168.8  
## 3 two hatchback rwd front 94.5 171.2  
## 4 four sedan fwd front 99.8 176.6  
## 5 four sedan 4wd front 99.4 176.6  
## 6 two sedan fwd front 99.8 177.3  
## width height curb.weight engine.type num.of.cylinders engine.size  
## 1 64.1 48.8 2548 dohc four 130  
## 2 64.1 48.8 2548 dohc four 130  
## 3 65.5 52.4 2823 ohcv six 152  
## 4 66.2 54.3 2337 ohc four 109  
## 5 66.4 54.3 2824 ohc five 136  
## 6 66.3 53.1 2507 ohc five 136  
## fuel.system bore stroke compression.ratio horsepower peak.rpm city.mpg  
## 1 mpfi 3.47 2.68 9.0 111 5000 21  
## 2 mpfi 3.47 2.68 9.0 111 5000 21  
## 3 mpfi 2.68 3.47 9.0 154 5000 19  
## 4 mpfi 3.19 3.40 10.0 102 5500 24  
## 5 mpfi 3.19 3.40 8.0 115 5500 18  
## 6 mpfi 3.19 3.40 8.5 110 5500 19  
## highway.mpg price  
## 1 27 13495  
## 2 27 16500  
## 3 26 16500  
## 4 30 13950  
## 5 22 17450  
## 6 25 15250

Data with na values

naData <- automobiles[rowSums(is.na(automobiles)) > 0, ]  
head (naData)

## symboling normalized.losses make fuel.type aspiration  
## 1 3 NA alfa-romero gas std  
## 2 3 NA alfa-romero gas std  
## 3 1 NA alfa-romero gas std  
## 6 2 NA audi gas std  
## 8 1 NA audi gas std  
## 10 0 NA audi gas turbo  
## num.of.doors body.style drive.wheels engine.location wheel.base length  
## 1 two convertible rwd front 88.6 168.8  
## 2 two convertible rwd front 88.6 168.8  
## 3 two hatchback rwd front 94.5 171.2  
## 6 two sedan fwd front 99.8 177.3  
## 8 four wagon fwd front 105.8 192.7  
## 10 two hatchback 4wd front 99.5 178.2  
## width height curb.weight engine.type num.of.cylinders engine.size  
## 1 64.1 48.8 2548 dohc four 130  
## 2 64.1 48.8 2548 dohc four 130  
## 3 65.5 52.4 2823 ohcv six 152  
## 6 66.3 53.1 2507 ohc five 136  
## 8 71.4 55.7 2954 ohc five 136  
## 10 67.9 52.0 3053 ohc five 131  
## fuel.system bore stroke compression.ratio horsepower peak.rpm city.mpg  
## 1 mpfi 3.47 2.68 9.0 111 5000 21  
## 2 mpfi 3.47 2.68 9.0 111 5000 21  
## 3 mpfi 2.68 3.47 9.0 154 5000 19  
## 6 mpfi 3.19 3.40 8.5 110 5500 19  
## 8 mpfi 3.19 3.40 8.5 110 5500 19  
## 10 mpfi 3.13 3.40 7.0 160 5500 16  
## highway.mpg price  
## 1 27 13495  
## 2 27 16500  
## 3 26 16500  
## 6 25 15250  
## 8 25 18920  
## 10 22 NA

Remove na valued data from automobiles

automobiles <- na.omit(automobiles)  
head (automobiles)

## symboling normalized.losses make fuel.type aspiration num.of.doors  
## 4 2 164 audi gas std four  
## 5 2 164 audi gas std four  
## 7 1 158 audi gas std four  
## 9 1 158 audi gas turbo four  
## 11 2 192 bmw gas std two  
## 12 0 192 bmw gas std four  
## body.style drive.wheels engine.location wheel.base length width height  
## 4 sedan fwd front 99.8 176.6 66.2 54.3  
## 5 sedan 4wd front 99.4 176.6 66.4 54.3  
## 7 sedan fwd front 105.8 192.7 71.4 55.7  
## 9 sedan fwd front 105.8 192.7 71.4 55.9  
## 11 sedan rwd front 101.2 176.8 64.8 54.3  
## 12 sedan rwd front 101.2 176.8 64.8 54.3  
## curb.weight engine.type num.of.cylinders engine.size fuel.system bore  
## 4 2337 ohc four 109 mpfi 3.19  
## 5 2824 ohc five 136 mpfi 3.19  
## 7 2844 ohc five 136 mpfi 3.19  
## 9 3086 ohc five 131 mpfi 3.13  
## 11 2395 ohc four 108 mpfi 3.50  
## 12 2395 ohc four 108 mpfi 3.50  
## stroke compression.ratio horsepower peak.rpm city.mpg highway.mpg price  
## 4 3.4 10.0 102 5500 24 30 13950  
## 5 3.4 8.0 115 5500 18 22 17450  
## 7 3.4 8.5 110 5500 19 25 17710  
## 9 3.4 8.3 140 5500 17 20 23875  
## 11 2.8 8.8 101 5800 23 29 16430  
## 12 2.8 8.8 101 5800 23 29 16925

Change column names for ease of access

colNames <- c("symboling", "normalized\_lossed", "make", "fuel\_type", "aspiration",  
 "doors", "body\_style", "drive\_wheels", "engine\_loc", "wheel\_base",  
 "length", "width", "height", "curb\_weight", "engine\_type", "noc",  
 "engine\_size", "fuel\_system", "bore", "stroke", "comp\_ratio", "hp",  
 "peak\_rpm", "city\_mpg", "high\_mpg", "price")  
colnames(automobiles) <- colNames  
head (automobiles)

## symboling normalized\_lossed make fuel\_type aspiration doors body\_style  
## 4 2 164 audi gas std four sedan  
## 5 2 164 audi gas std four sedan  
## 7 1 158 audi gas std four sedan  
## 9 1 158 audi gas turbo four sedan  
## 11 2 192 bmw gas std two sedan  
## 12 0 192 bmw gas std four sedan  
## drive\_wheels engine\_loc wheel\_base length width height curb\_weight  
## 4 fwd front 99.8 176.6 66.2 54.3 2337  
## 5 4wd front 99.4 176.6 66.4 54.3 2824  
## 7 fwd front 105.8 192.7 71.4 55.7 2844  
## 9 fwd front 105.8 192.7 71.4 55.9 3086  
## 11 rwd front 101.2 176.8 64.8 54.3 2395  
## 12 rwd front 101.2 176.8 64.8 54.3 2395  
## engine\_type noc engine\_size fuel\_system bore stroke comp\_ratio hp  
## 4 ohc four 109 mpfi 3.19 3.4 10.0 102  
## 5 ohc five 136 mpfi 3.19 3.4 8.0 115  
## 7 ohc five 136 mpfi 3.19 3.4 8.5 110  
## 9 ohc five 131 mpfi 3.13 3.4 8.3 140  
## 11 ohc four 108 mpfi 3.50 2.8 8.8 101  
## 12 ohc four 108 mpfi 3.50 2.8 8.8 101  
## peak\_rpm city\_mpg high\_mpg price  
## 4 5500 24 30 13950  
## 5 5500 18 22 17450  
## 7 5500 19 25 17710  
## 9 5500 17 20 23875  
## 11 5800 23 29 16430  
## 12 5800 23 29 16925

Class of different columns of automobiles data

colClasses <- sapply(automobiles, class)  
colClasses

## symboling normalized\_lossed make fuel\_type   
## "integer" "integer" "factor" "factor"   
## aspiration doors body\_style drive\_wheels   
## "factor" "factor" "factor" "factor"   
## engine\_loc wheel\_base length width   
## "factor" "numeric" "numeric" "numeric"   
## height curb\_weight engine\_type noc   
## "numeric" "integer" "factor" "factor"   
## engine\_size fuel\_system bore stroke   
## "integer" "factor" "numeric" "numeric"   
## comp\_ratio hp peak\_rpm city\_mpg   
## "numeric" "integer" "integer" "integer"   
## high\_mpg price   
## "integer" "integer"

According to NOIR topology, category of data are symboling - Nominal normalized\_losses - Interval make - Nominal fuel\_type - Nominal aspiration - Nominal doors - Ordinal body\_style - Nominal drive\_wheels - Nominal engine\_loc - Nominal wheele\_base - Interval length - Interval width - Interval height - Interval curb\_weight - Interval engine\_type - Nominal noc - Ordinal engine\_size - Interval fuel\_system - Nominal bore - Interval stroke - Interval comp\_ratio - Rational hp - Interval peak\_rpm - Interval city\_mpg - Interval high\_mpg - Interval prive - Interval

We will take the following attributes 1. Nominal - make 2. Ordinal - noc 3. Interval - length 4. Rational - comp\_ratio

## 1. Nominal (make)

subsetting data

make\_data <- subset(automobiles, select = make)  
head (make\_data)

## make  
## 4 audi  
## 5 audi  
## 7 audi  
## 9 audi  
## 11 bmw  
## 12 bmw

grouping data

library (dplyr)

## Warning: package 'dplyr' was built under R version 3.3.3

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

make\_data <- group\_by (make\_data, make)  
head (make\_data)

## # A tibble: 6 x 1  
## # Groups: make [2]  
## make  
## <fctr>  
## 1 audi  
## 2 audi  
## 3 audi  
## 4 audi  
## 5 bmw  
## 6 bmw

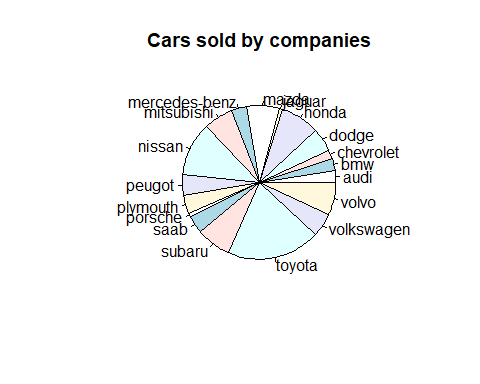
making count of each factor

make\_data <- summarize(make\_data, count = n())  
head (make\_data)

## # A tibble: 6 x 2  
## make count  
## <fctr> <int>  
## 1 audi 4  
## 2 bmw 4  
## 3 chevrolet 3  
## 4 dodge 8  
## 5 honda 13  
## 6 jaguar 1

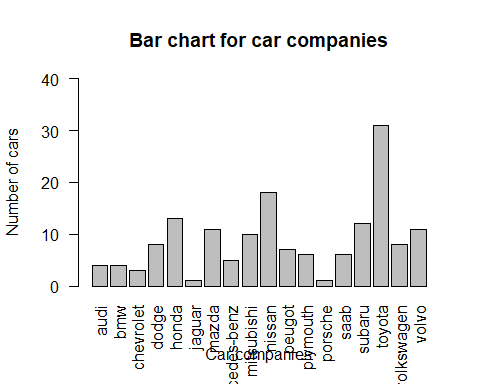
Pie chart

pie(make\_data$count, labels = make\_data$make, main = "Cars sold by companies")



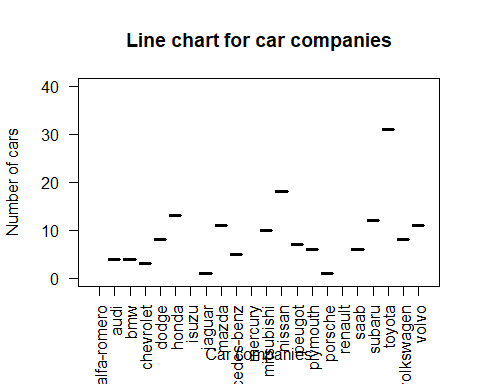
Bar plot

barplot(make\_data$count, names.arg = make\_data$make, main = "Bar chart for car companies",  
 xlab = "Car companies", ylab = "Number of cars", las = 2, ylim = c(0, 40))



Line chart

plot(make\_data$make, make\_data$count,  
 main = "Line chart for car companies",  
 xlab = "Car companies",  
 ylab = "Number of cars", las = 2, ylim = c(0, 40))

 We can see that maximum number of cars has been sold by 'toyota', i.e. mode of this distribution is toyota

## 2. Ordinal (number\_of\_cylinders)

sbsetting data

noc\_data <- subset(automobiles, select = noc)  
head (noc\_data)

## noc  
## 4 four  
## 5 five  
## 7 five  
## 9 five  
## 11 four  
## 12 four

grouping data

noc\_data <- group\_by(noc\_data, noc)  
noc\_data <- summarize(noc\_data, count = n())  
head (noc\_data)

## # A tibble: 5 x 2  
## noc count  
## <fctr> <int>  
## 1 eight 1  
## 2 five 7  
## 3 four 136  
## 4 six 14  
## 5 three 1

vector to order the factors

numOfCyls <- c("one" = 1,  
 "two" = 2,  
 "three" = 3,  
 "four" = 4,  
 "five" = 5,  
 "six" = 6,  
 "seven" = 7,  
 "eight" = 8,  
 "nine" = 9,  
 "ten" = 10  
 )

extra column to set order of noc

noc\_data$noc <- as.character(noc\_data$noc)  
noc\_data <- mutate(noc\_data, lab = numOfCyls[noc])

## Warning: package 'bindrcpp' was built under R version 3.3.3

head (noc\_data)

## # A tibble: 5 x 3  
## noc count lab  
## <chr> <int> <dbl>  
## 1 eight 1 8  
## 2 five 7 5  
## 3 four 136 4  
## 4 six 14 6  
## 5 three 1 3

sorting w.r.t. lab (noc)

noc\_data <- arrange(noc\_data, lab)  
noc\_data

## # A tibble: 5 x 3  
## noc count lab  
## <chr> <int> <dbl>  
## 1 three 1 3  
## 2 four 136 4  
## 3 five 7 5  
## 4 six 14 6  
## 5 eight 1 8

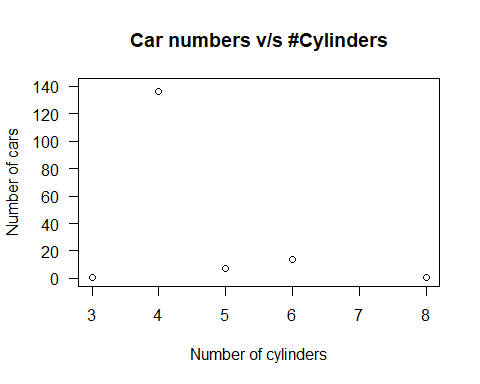
removing lab column from noc\_data

noc\_data$noc <- noc\_data$lab  
noc\_data <- subset (noc\_data, select = -lab)  
noc\_data

## # A tibble: 5 x 2  
## noc count  
## <dbl> <int>  
## 1 3 1  
## 2 4 136  
## 3 5 7  
## 4 6 14  
## 5 8 1

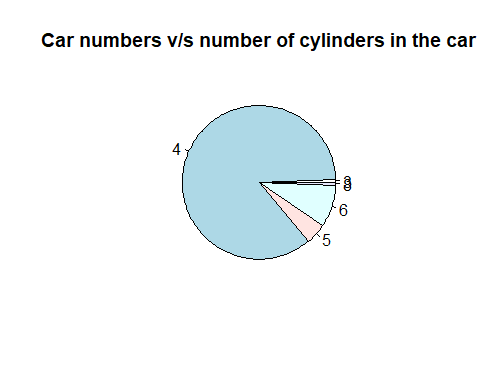
Line plot

plot(noc\_data$noc, noc\_data$count, main = "Car numbers v/s #Cylinders",  
 xlab = "Number of cylinders", ylab = "Number of cars", las = 1,  
 ylim = c(0, 140))



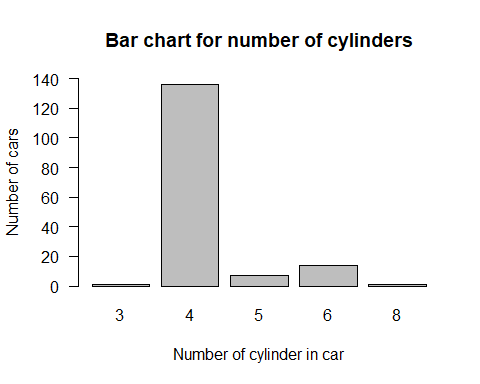
Pie chart

pie(noc\_data$count, labels = noc\_data$noc,  
 main = "Car numbers v/s number of cylinders in the car")



Bar plot

barplot(noc\_data$count, names.arg = noc\_data$noc, main = "Bar chart for number of cylinders",  
 xlab = "Number of cylinder in car", ylab = "Number of cars", las = 1, ylim = c(0, 140))

 We can see taht maximum number of cars correspond to 4 cylinders in the car, so mode is 4.

## 3. Interval (length)

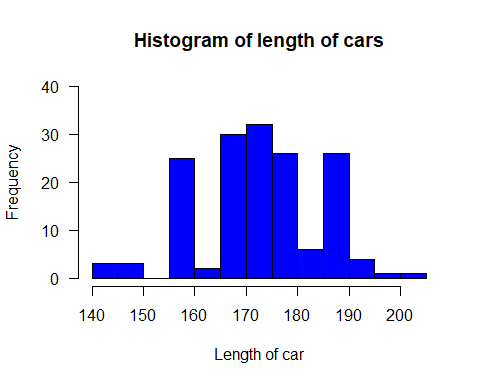
subsetting data

length\_data <- subset(automobiles, select = length)  
head (length\_data)

## length  
## 4 176.6  
## 5 176.6  
## 7 192.7  
## 9 192.7  
## 11 176.8  
## 12 176.8

Histogram of length of cars

hist (length\_data$length,  
 col = "blue",  
 main = "Histogram of length of cars",  
 ylim = c(0, 40),  
 xlab = "Length of car",  
 las = 1)



Calculate mean, variance and standard deviation

mean\_len <- mean(length\_data$length)  
var\_len <- var(length\_data$length)  
sd\_len <- sd(length\_data$length)  
sprintf("%s ===> %f", c("Mean", "Variance", "Standard deiation"), c(mean\_len, var\_len, sd\_len))

## [1] "Mean ===> 172.413836" "Variance ===> 132.783605"   
## [3] "Standard deiation ===> 11.523177"

Calculate q-values (5 points)

q <- quantile(length\_data$length, probs = c(0, 0.25, 0.50, 0.75, 1.0))  
q

## 0% 25% 50% 75% 100%   
## 141.10 165.65 172.40 177.80 202.60

We observe that mean is 172.413836, median is 172.40 and mod is 172.50 (approximately)

## 4: Rational - comp\_ratio

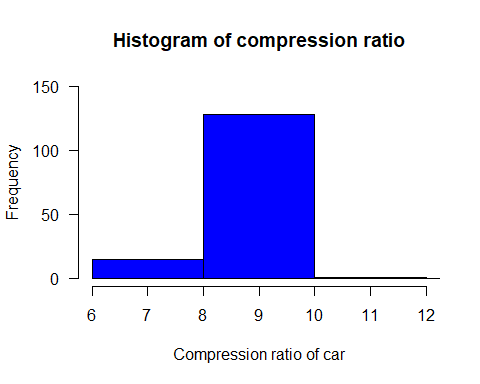
subsetting data

comp\_data <- subset(automobiles, select = comp\_ratio)  
head (comp\_data)

## comp\_ratio  
## 4 10.0  
## 5 8.0  
## 7 8.5  
## 9 8.3  
## 11 8.8  
## 12 8.8

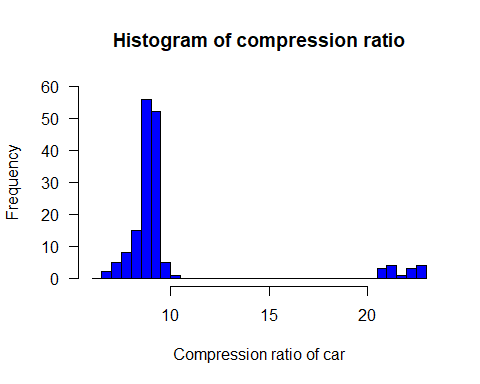
Histogram of compression ratio

hist (comp\_data$comp\_ratio,  
 col = "blue",  
 main = "Histogram of compression ratio",  
 ylim = c(0, 150),  
 xlim = c(6, 12),  
 xlab = "Compression ratio of car",  
 las = 1)

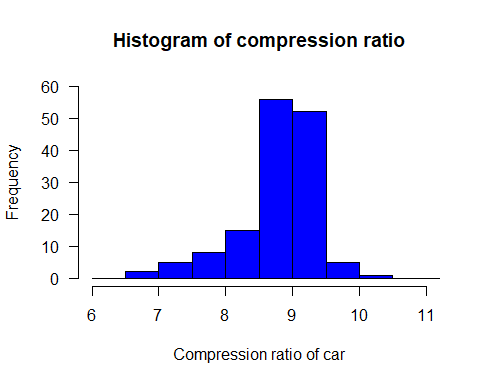


Another histogram

hist (comp\_data$comp\_ratio,  
 col = "blue",  
 main = "Histogram of compression ratio",  
 ylim = c(0, 60),  
 xlab = "Compression ratio of car",  
 breaks = seq(6, 23, 0.5),  
 las = 1)

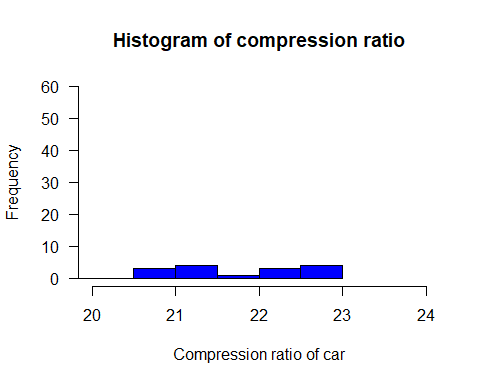
 Another hsitogram

hist (comp\_data$comp\_ratio,  
 col = "blue",  
 main = "Histogram of compression ratio",  
 ylim = c(0, 60),  
 xlim = c(6, 11),  
 xlab = "Compression ratio of car",  
 breaks = seq(6, 23, 0.5),  
 las = 1)



Another histogram

hist (comp\_data$comp\_ratio,  
 col = "blue",  
 main = "Histogram of compression ratio",  
 ylim = c(0, 60),  
 xlim = c(20, 24),  
 xlab = "Compression ratio of car",  
 breaks = seq(6, 23, 0.5),  
 las = 1)



Calculate mean, variance and standard deviation

mean\_comp <- mean(comp\_data$comp\_ratio)  
var\_comp <- var(comp\_data$comp\_ratio)  
sd\_comp <- sd(comp\_data$comp\_ratio)  
sprintf("%s ===> %f", c("Mean", "Variance", "Standard deiation"), c(mean\_comp, var\_comp, sd\_comp))

## [1] "Mean ===> 10.161132" "Variance ===> 15.128013"   
## [3] "Standard deiation ===> 3.889475"

Calculate q-values (5 points)

q <- quantile(comp\_data$comp\_ratio, probs = c(0, 0.25, 0.50, 0.75, 1.0))  
q

## 0% 25% 50% 75% 100%   
## 7.0 8.7 9.0 9.4 23.0

We observe that mean is 10.161132, median is 9.0 and mode is 8.75 (approximately)

## Peak RPM

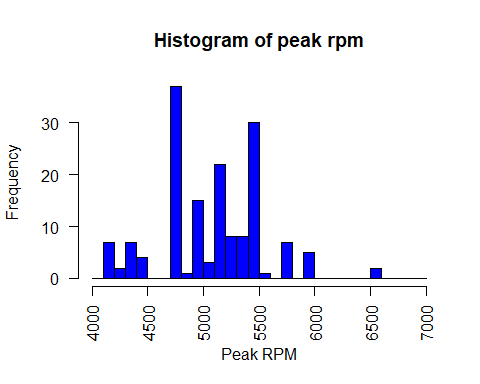
subsetting data

peak\_data <- subset(automobiles, select = peak\_rpm)  
head (peak\_data)

## peak\_rpm  
## 4 5500  
## 5 5500  
## 7 5500  
## 9 5500  
## 11 5800  
## 12 5800

Histogram of peak rpm

hist (peak\_data$peak\_rpm,  
 col = "blue",  
 main = "Histogram of peak rpm",  
 xlab = "Peak RPM",  
 las = 2,  
 breaks = seq(4000, 7000, 100))



Calculate mean, variance and standard deviation

mean\_pr <- mean(peak\_data$peak\_rpm)  
var\_pr <- var(peak\_data$peak\_rpm)  
sd\_pr <- sd(peak\_data$peak\_rpm)  
sprintf("%s ===> %f", c("Mean", "Variance", "Standard deviation"), c(mean\_pr, var\_pr, sd\_pr))

## [1] "Mean ===> 5113.836478"   
## [2] "Variance ===> 216927.593344"   
## [3] "Standard deviation ===> 465.754864"

Calculate q-values (5 points)

q <- quantile(peak\_data$peak\_rpm, probs = c(0, 0.25, 0.50, 0.75, 1.0))  
q

## 0% 25% 50% 75% 100%   
## 4150 4800 5200 5500 6600

We observe that mean is 5113.836478, median is 5200 and mode is 4750 (approximately)

## City MPG

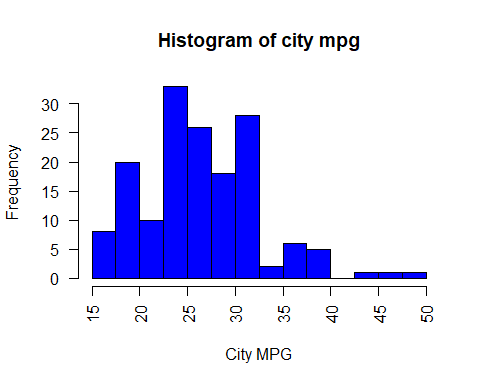
subsetting data

city\_mpg <- subset (automobiles, select = city\_mpg)  
head (city\_mpg)

## city\_mpg  
## 4 24  
## 5 18  
## 7 19  
## 9 17  
## 11 23  
## 12 23

Histogram of city mpg

hist (city\_mpg$city\_mpg,  
 col = "blue",  
 main = "Histogram of city mpg",  
 xlab = "City MPG",  
 breaks = seq(15.0, 50.0, 2.5),  
 las = 2)



Calculate mean, variance and standard deviation

mean\_cm <- mean (city\_mpg$city\_mpg)  
var\_cm <- var (city\_mpg$city\_mpg)  
sd\_cm <- sd (city\_mpg$city\_mpg)  
sprintf ("%s ===> %f", c("Mean", "Variance", "Standard deviation"),  
 c(mean\_cm, var\_cm, sd\_cm))

## [1] "Mean ===> 26.522013" "Variance ===> 37.175145"   
## [3] "Standard deviation ===> 6.097142"

Calculating q-values (5 points)

q <- quantile (city\_mpg$city\_mpg, probs = c(0, 0.25, 0.50, 0.75, 1.0))  
q

## 0% 25% 50% 75% 100%   
## 15 23 26 31 49

We observe that this is left skewed normal distribution for which mean is 26.522013, median is 26 and mod is 23.75 (approximately)