Pie Chart, Histogram and Boxplot in R Markdown

Dr. Suborna Ahmed

# Preparing directory and data setup

In this lab, we will be creating a pie chart, a histogram and boxplot using the VRI\_data.csv file. Import this into R by using the read.csv() function.

# current directory  
getwd()

## [1] "C:/Syd/UBC/Teaching/GAA FRST 232/Archivos finales/FinalJun10"

# VRI data  
  
vri\_data<-read.csv("VRI\_data.csv",header = T)  
  
attach(vri\_data)  
  
# separating data by species  
d1<- subset(vri\_data, (SPECIES\_CD\_1 =='CW'))  
d2<- subset(vri\_data, (SPECIES\_CD\_1 =='PLC'))

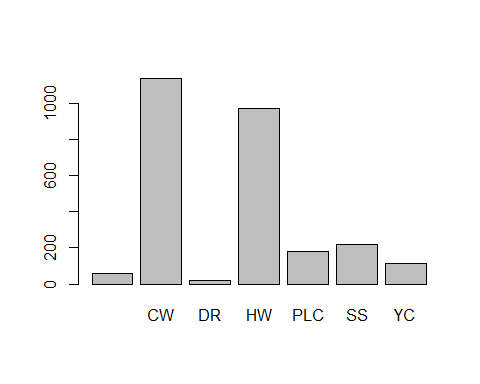
# Pie charts

Pie charts are used to display frequencies of categorical variables. This method uses colored blocks around a circumference of a circle to represent frequency or proportions. To create a pie chart, we must first create a table with the variable that we want to plot using the table() function. Then we have to use the pie() function to make the graph; to change the size of the circumference we can use radius= into this function. If we want to add a square around the pie plot we can use box(). For this last option, make sure that you run the pie() and box() functions together.

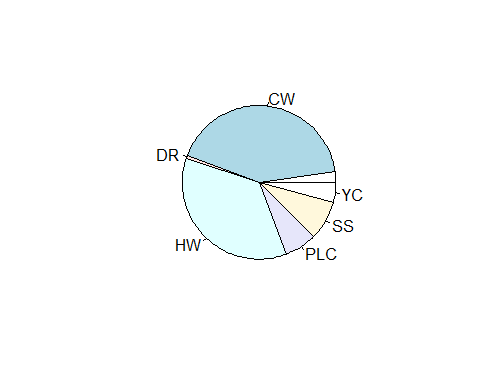
class(SPECIES\_CD\_1)

## [1] "character"

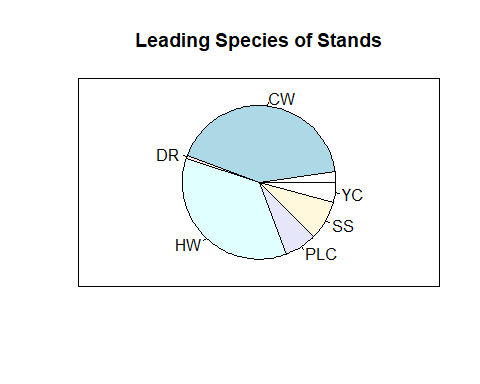
#Pie chart 1  
#COnvert the character variable as a factor and then plot  
plot(factor(SPECIES\_CD\_1))



#Pie chart 2  
leading\_spec <- table(SPECIES\_CD\_1)  
percent\_spec <- (table(SPECIES\_CD\_1)\*100)/2700  
pie(leading\_spec)



#Pie chart 3  
pie(percent\_spec, main = "Leading Species of Stands")  
  
#Pie chart 4  
pie(percent\_spec, main = "Leading Species of Stands", radius = .8)  
box()



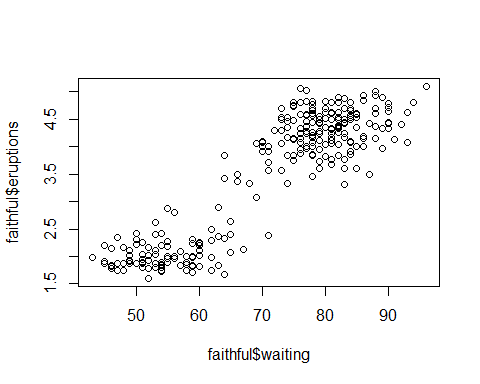
# Plots in R: using dataset in R

In R there are some available datasets that we can use to create graphs. These datasets are in R by default so we don’t need to import them into the program. For this part, we will work with a dataset that is called “faithful”. We can check the name of the variables that are in this dataset by using the names() function. We can also get more information regarding the dataset by using the “?” sign followed by the name of the dataset (i.e. ?faithful); this will show us specific information about this dataset in the “Help” tab/window. Using this option we can get information about the variables, background of the study made, and even the link of the paper that was published using this dataset. Once we have identified the variables contained in the dataset we can use it to create a scatterplot, for example, and modify different parameters on the graph.

# scatterplot  
names(faithful)

## [1] "eruptions" "waiting"

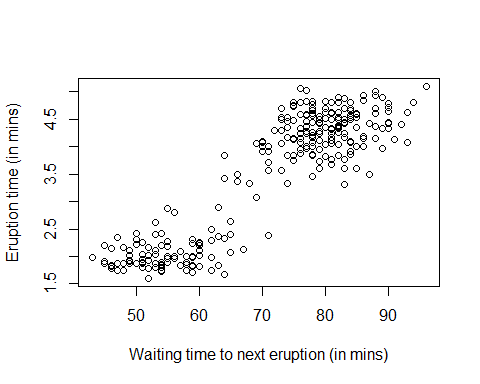
plot(faithful$waiting,faithful$eruptions)



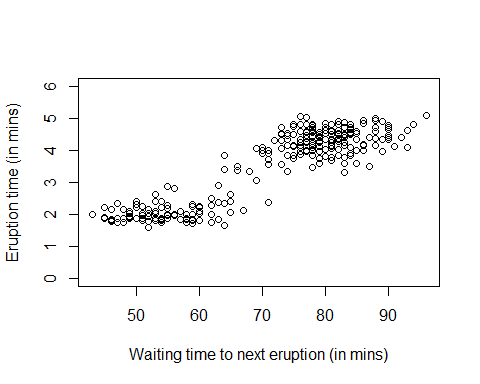
?faithful

## starting httpd help server ... done

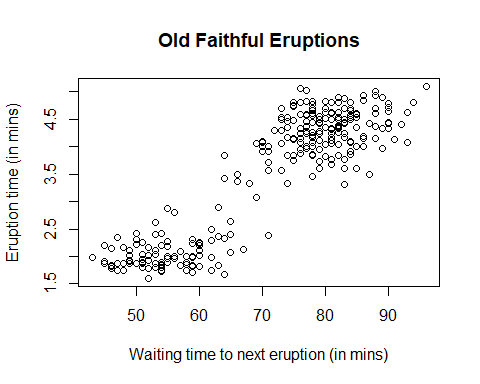
# add Axis labels  
plot(faithful$waiting,faithful$eruptions,  
 xlab = "Waiting time to next eruption (in mins)",  
 ylab = "Eruption time (in mins)")



# change Y-axis range  
plot(faithful$waiting,faithful$eruptions,  
 xlab = "Waiting time to next eruption (in mins)", ylab = "Eruption time (in mins)",  
 ylim = c(0, 6))



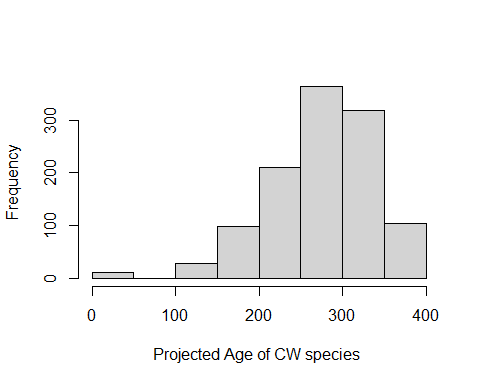
# include title  
plot(faithful$waiting,faithful$eruptions,  
 xlab = "Waiting time to next eruption (in mins)", ylab = "Eruption time (in mins)",  
 main = "Old Faithful Eruptions")



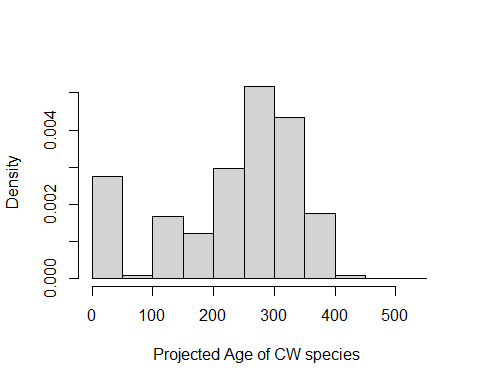
# Histogram

We use histograms to display a frequency distribution for a numerical variable. The data values in a histogram are split into consecutive intervals or bins, where the frequency of observation falling into each bin is displayed. To create this type of graph we use the hist() function follow by the name of the dataset that we want to use and the specific variable that we want to plot (i.e hist(name\_dataset[name\_of\_variable= “variable”])). To specify the number of bins in the histogram use breaks=.

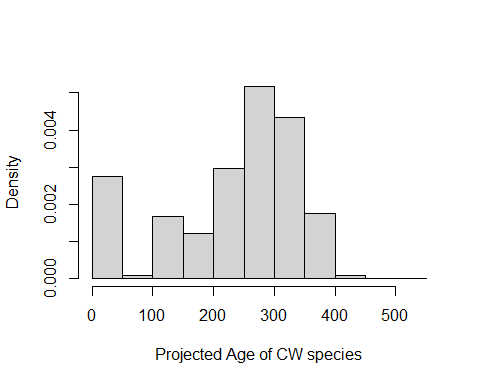
# VRI data  
hist(PROJ\_AGE\_1[SPECIES\_CD\_1 == "CW"], xlab="Projected Age of CW species ", main="")



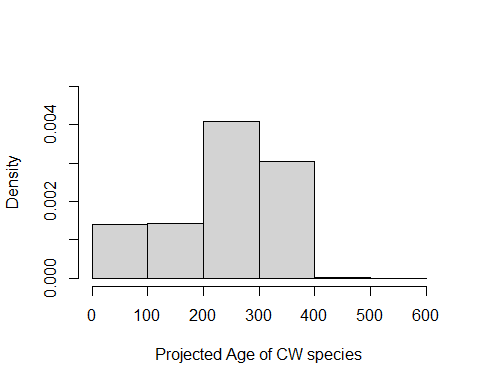
# setting freq argument to false or setting prob argument to true  
hist(PROJ\_AGE\_1, freq = F, xlab="Projected Age of CW species ", main="")



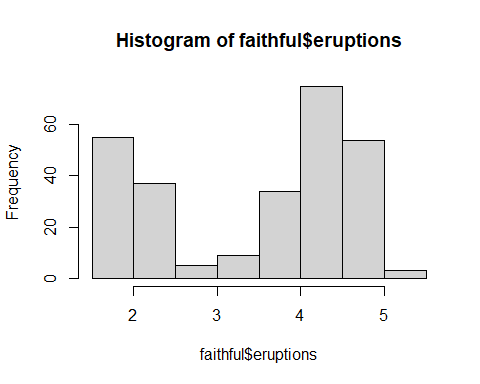
hist(PROJ\_AGE\_1, prob = T, xlab="Projected Age of CW species", main="")



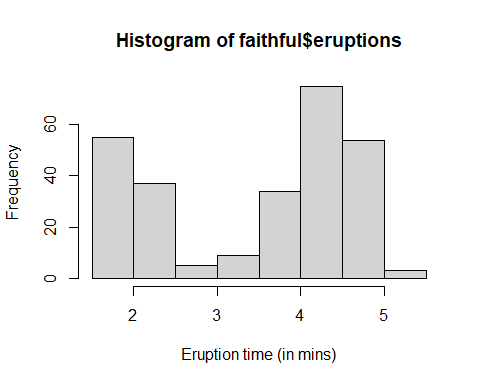
# specifying breaks  
hist(PROJ\_AGE\_1, freq = F, ylim = c(0,0.005), breaks = 4, xlab="Projected Age of CW species", main="")



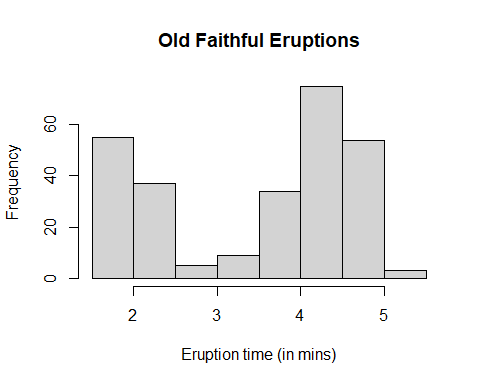
# histogram using R data file  
hist(faithful$eruptions)



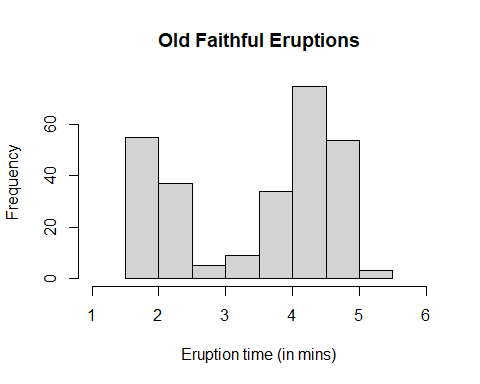
# change X-axis label  
hist(faithful$eruptions, xlab = "Eruption time (in mins)")



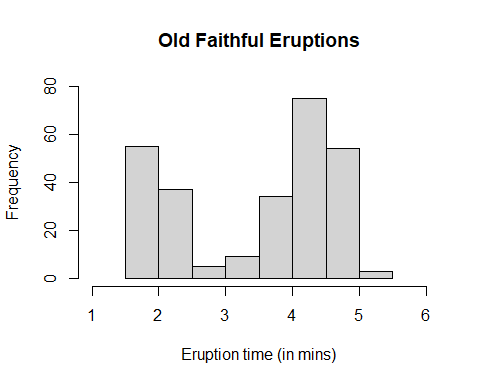
# change title  
hist(faithful$eruptions, xlab = "Eruption time (in mins)", main = "Old Faithful Eruptions")



# change range of axes  
hist(faithful$eruptions, xlab = "Eruption time (in mins)", main = "Old Faithful Eruptions",  
 xlim = c(1, 6))



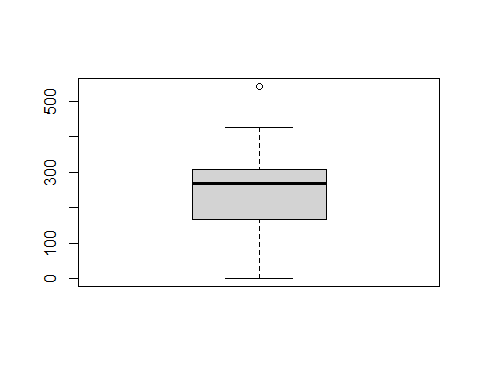
hist(faithful$eruptions, xlab = "Eruption time (in mins)", main = "Old Faithful Eruptions",  
 xlim = c(1, 6), ylim = c(0,80))



# Box Plot

We use box plots to show associations between one numerical variable and one categorical variable. The lines and the rectangular box in this graph display the median, quartiles, range, and extreme measurements of the data. To plot this graph use the boxplot() function and specify the name of the numerical variable that you want to graph. To display multiple boxplots side-by-side we use the name of the numerical variable followed by a tilde and the name of the categorical variable (i.e. boxplot(name\_numerical\_variable ~ name\_categorical\_variable)). To plot selected categories or variables in the box plot use boxplot(name\_numerical\_variable[name\_categorical\_variable==“name\_of\_specfic\_variable”]).

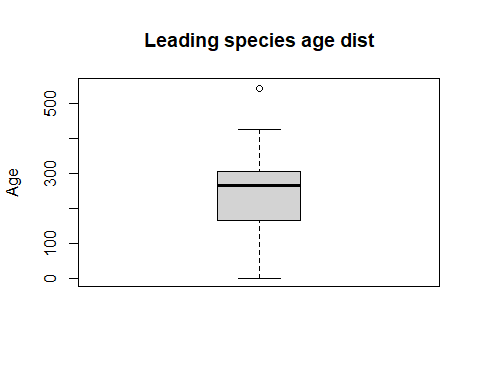
boxplot(PROJ\_AGE\_1)



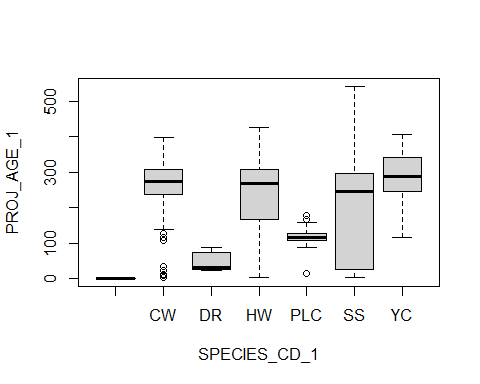
quantile(PROJ\_AGE\_1, probs= c(0, 0.25, 0.5, 0.75, 1))

## 0% 25% 50% 75% 100%   
## 0 167 267 307 543

# controlling box, min & max width  
# boxwex: increase or decrease the box width   
# staplewex: increase or decrease the width of max and min lines  
  
boxplot(PROJ\_AGE\_1, main="Leading species age dist", ylab="Age", ylim=c(0,550),  
 pars = list(boxwex = 0.5, staplewex = 0.5))



# multiple categories side-by-side  
boxplot(PROJ\_AGE\_1 ~ SPECIES\_CD\_1)



# reduced catagories  
boxplot(PROJ\_AGE\_1[SPECIES\_CD\_1=="CW"],PROJ\_AGE\_1[SPECIES\_CD\_1=="SS"], PROJ\_AGE\_1[SPECIES\_CD\_1=="YC"], names=c("CW", "SS", "YC"))

