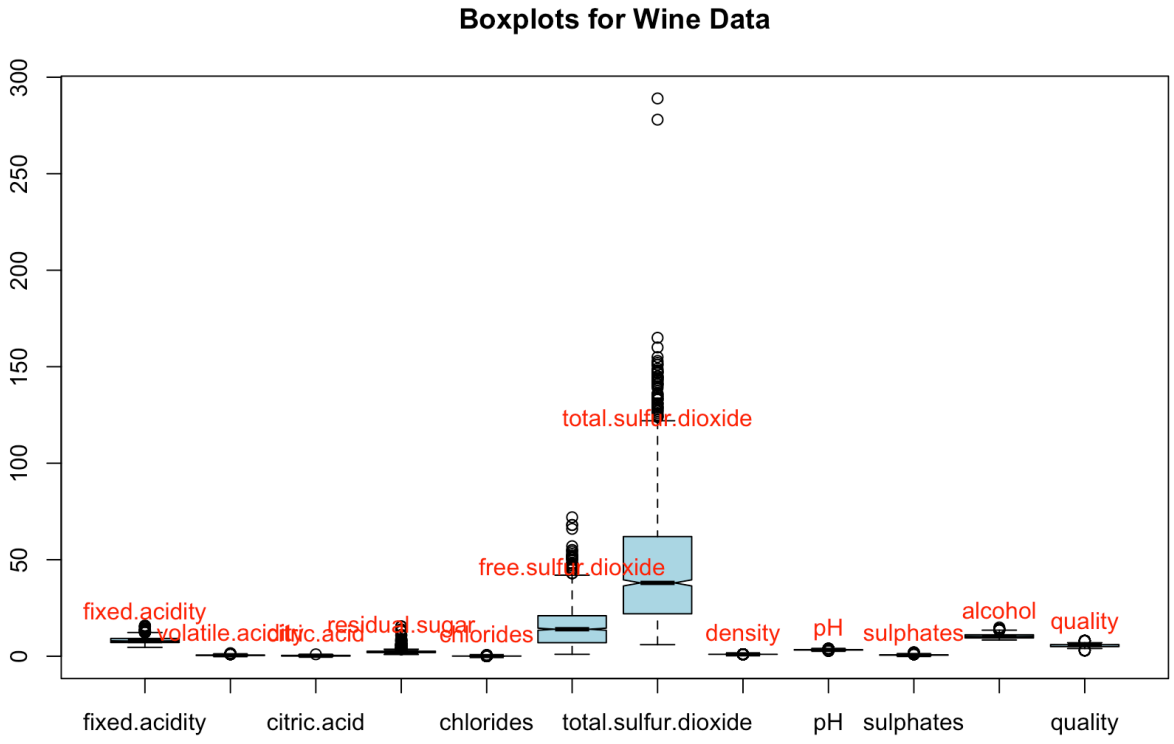
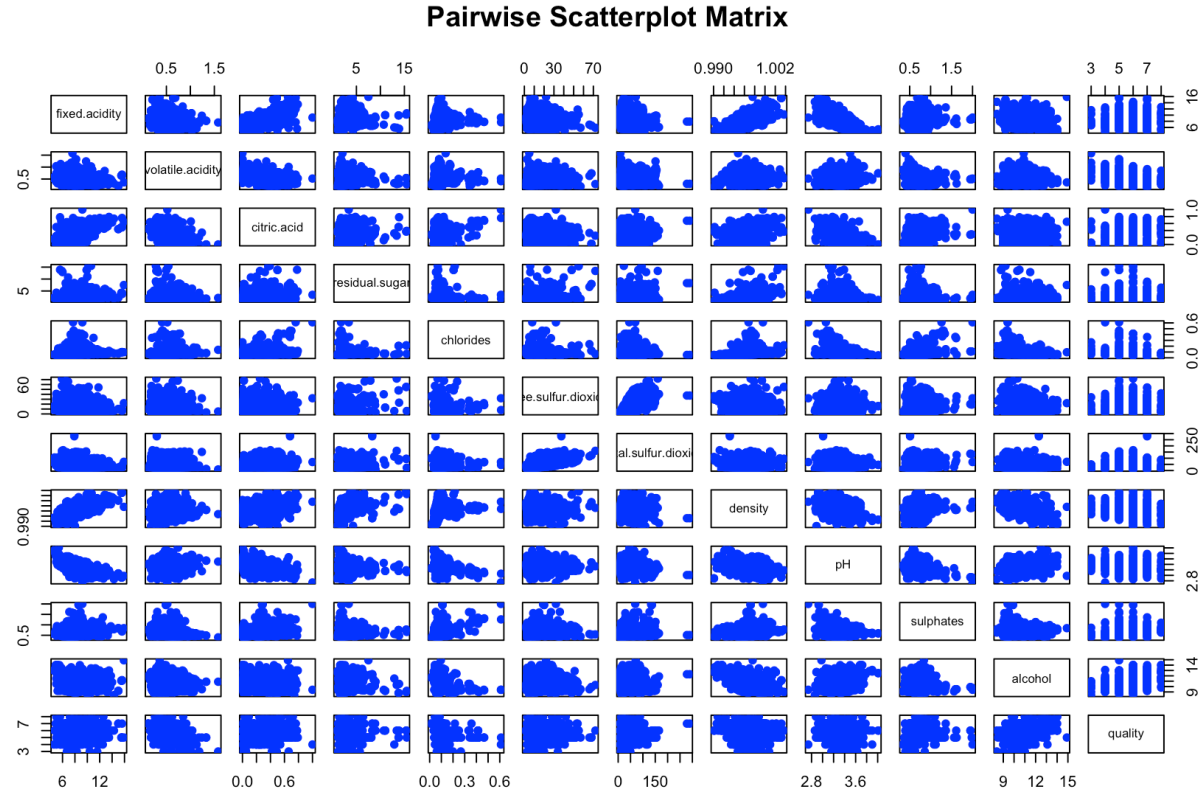
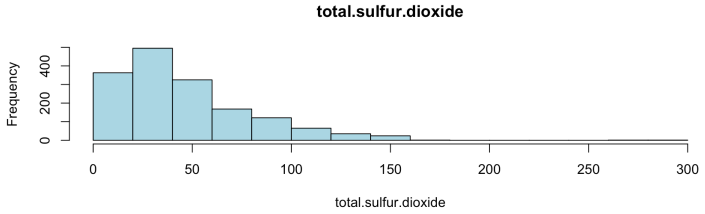
# CMPE 343

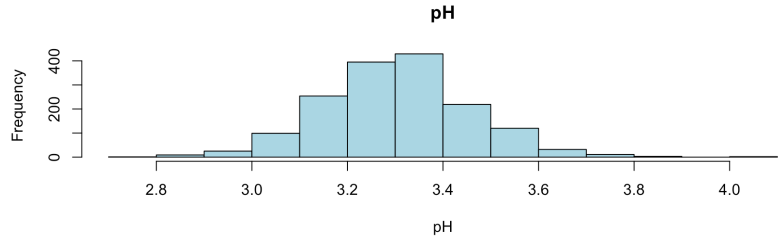
Homework 1

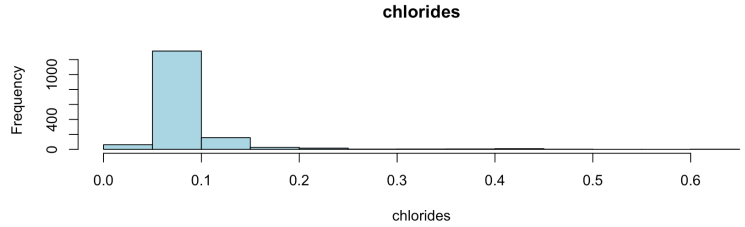
Güney Berkay Ateş:120200047
Batıkan Yılmaz:120200036

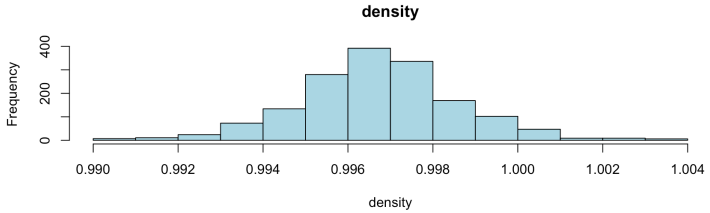

Boxplots for Wine Data
0
50
100
150
200
250
300
fixed.acidity
volatile.acidity
citric.acid
residual.sugar
chlorides
free.sulfur.dioxide
total.sulfur.dioxide
density
pH
sulphates
alcohol
quality

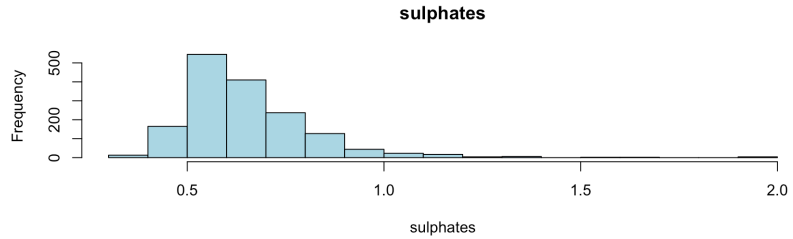

Pairwise Scatterplot Matrix
fixed.acidity
volatile.acidity
citric.acid
residual.sugar
chlorides
density
pH
sulphates
alcohol
quality

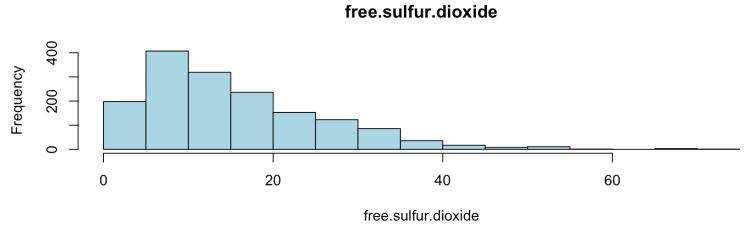

total.sulfur.dioxide
Frequency
0
200
400
0
50
100
150
200
250
300
total.sulfur.dioxide

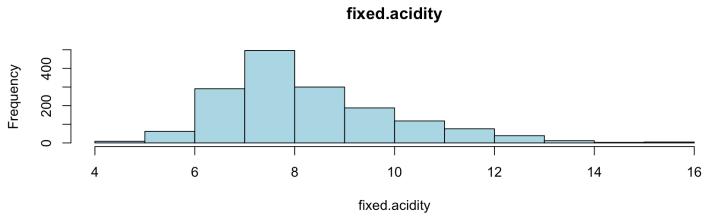

pH
Frequency
0
200
400
2.8
3.0
3.2
3.4
3.6
3.8
4.0
pH

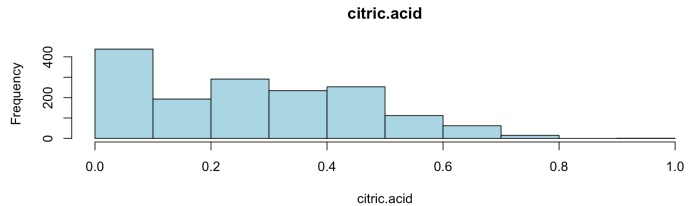

chlorides
Frequency
0
400
1000
0.0
0.1
0.2
0.3
0.4
0.5
0.6
chlorides

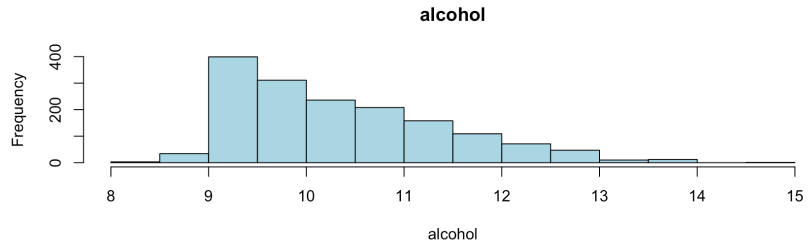

density
Frequency
0
200
400
0.990
0.992
0.994
0.996
0.998
1.000
1.002
1.004
density

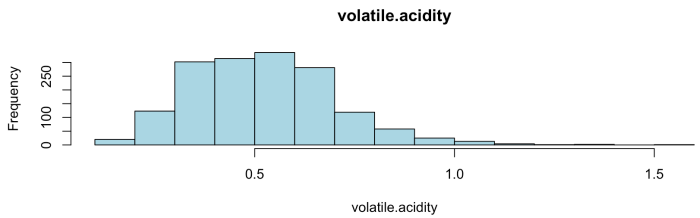

sulphates
Frequency
0
200
500
0.5
1.0
1.5
2.0
sulphates

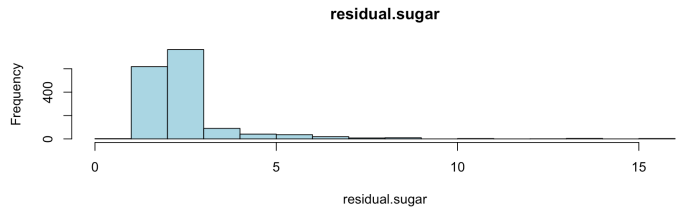

free.sulfur.dioxide
Frequency
0
200
400
0
20
40
60
free.sulfur.dioxide

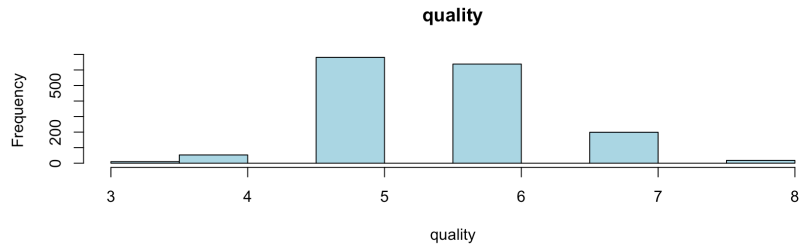

fixed.acidity
Frequency
0
200
400
4
6
8
10
12
14
16
fixed.acidity


citric.acid
Frequency
0
200
400
0.0
0.2
0.4
0.6
0.8
1.0
citric.acid


alcohol
Frequency
0
200
400
8
9
10
11
12
13
14
15
alcohol


volatile.acidity
Frequency
0
100
250
0.5
1.0
1.5
volatile.acidity

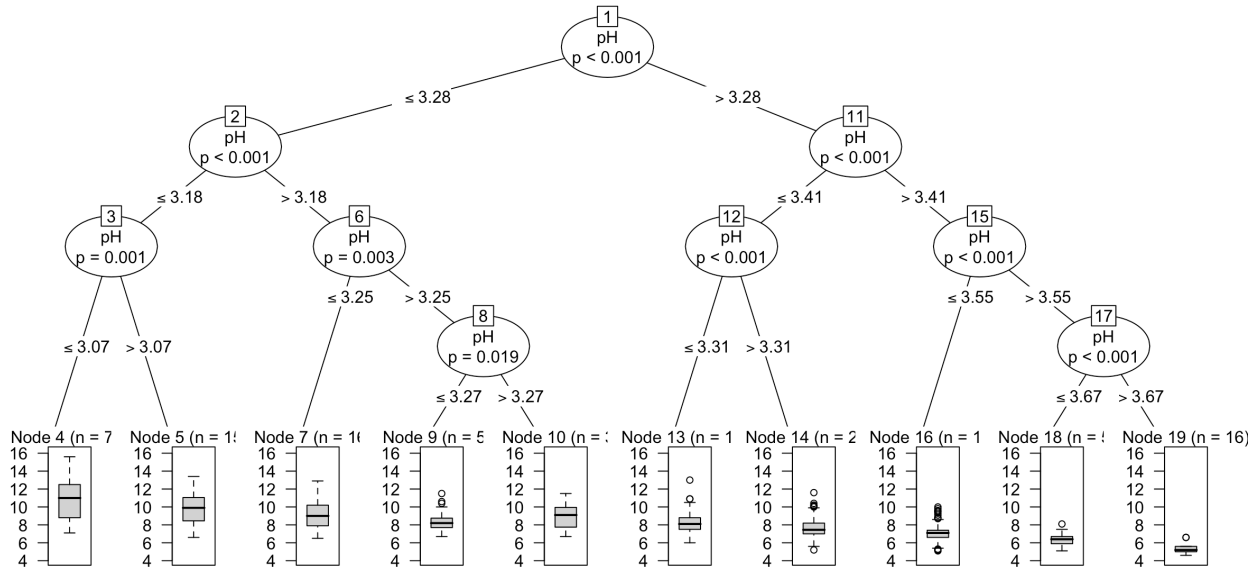

residual.sugar
Frequency
0
400
0
5
10
15
residual.sugar


quality
Frequency
0
200
500
3
4
5
6
7
8
quality

```
        Conditional inference tree with 10 terminal nodes

Response:  fixed.acidity
Input:  pH
Number of observations:  1130

1) pH <= 3.28; criterion = 1, statistic = 524.623
  2) pH <= 3.18; criterion = 1, statistic = 69.857
    3) pH <= 3.07; criterion = 0.999, statistic = 10.263
      4)*  weights = 70
    3) pH > 3.07
      5)*  weights = 159
  2) pH > 3.18
    6) pH <= 3.25; criterion = 0.997, statistic = 8.716
      7)*  weights = 169
    6) pH > 3.25
      8) pH <= 3.27; criterion = 0.981, statistic = 5.484
        9)*  weights = 56
      8) pH > 3.27
        10)*  weights = 32
1) pH > 3.28
  11) pH <= 3.41; criterion = 1, statistic = 185.438
    12) pH <= 3.31; criterion = 1, statistic = 29.735
      13)*  weights = 102
    12) pH > 3.31
      14)*  weights = 294
  11) pH > 3.41
    15) pH <= 3.55; criterion = 1, statistic = 60.296
      16)*  weights = 182
    15) pH > 3.55
      17) pH <= 3.67; criterion = 1, statistic = 15.59
        18)*  weights = 50
      17) pH > 3.67
        19)*  weights = 16
```



```r
# Load necessary library
library(datasets)

# Load Red Wine Quality dataset
data(WineQuality.RedWine)
wine_data <- WineQuality.RedWine

# Display the first few rows of the dataset
head(wine_data)

# Function to find univariate outliers using the IQR method
find_outliers <- function(x) {
  iqr <- IQR(x)
  q1 <- quantile(x)[2]
  q3 <- quantile(x)[4]
  lower_bound <- q1 - 1.5 * iqr
  upper_bound <- q3 + 1.5 * iqr
  outliers <- x[x < lower_bound | x > upper_bound]
  return(outliers)
}

# Apply the find_outliers function to each column of the dataset
outliers_univariate <- sapply(wine_data, find_outliers)

# Create boxplots for each variable in the dataset
boxplot(wine_data, main = "Boxplots for Wine Data", col = "lightblue", notch = TRUE)

# Annotate extreme values in each boxplot
for (i in 1:ncol(wine_data)) {
  text(i, quantile(wine_data[, i], 0.95), labels = names(wine_data)[i], pos = 3, col = "red")
}

# Create histograms for each variable in the dataset in a 2x1 layout
par(mfrow = c(2, 1))
for (i in 1:ncol(wine_data)) {
  hist(wine_data[[i]], main = colnames(wine_data)[i], col = "lightblue", xlab = colnames(wine_data)[i], ylab =
"Frequency")
}
par(mfrow = c(1, 1))

# Create a pairwise scatterplot matrix for the variables in the dataset
pairs(wine_data, pch = 19, col = "blue", main = "Pairwise Scatterplot Matrix")
```

```r
# Load the necessary library
library(datasets)

# Load the dataset
data(WineQuality.RedWine)
wine_data <- WineQuality.RedWine

# Display the structure of the dataset
str(wine_data)

# Randomly split the dataset into training and testing sets
ind <- sample(2, nrow(wine_data), replace=TRUE, prob=c(0.7, 0.3))
trainData <- wine_data[ind == 1,]
testData <- wine_data[ind == 2,]

# Load the 'party' library for conditional inference trees
library(party)

# Specify the formula for the conditional inference tree
myFormula <- fixed.acidity ~ pH

# Create a conditional inference tree using the training data
wine_ctree <- ctree(myFormula, data=trainData)

# Display a table of predicted vs. actual values for the 'fixed.acidity' variable on the training
data
table(predict(wine_ctree), trainData$fixed.acidity)

# Print the details of the trained conditional inference tree
print(wine_ctree)

# Plot the conditional inference tree
plot(wine_ctree)
```