Hw 1

# Part 1 – Accuracy on the Test Data

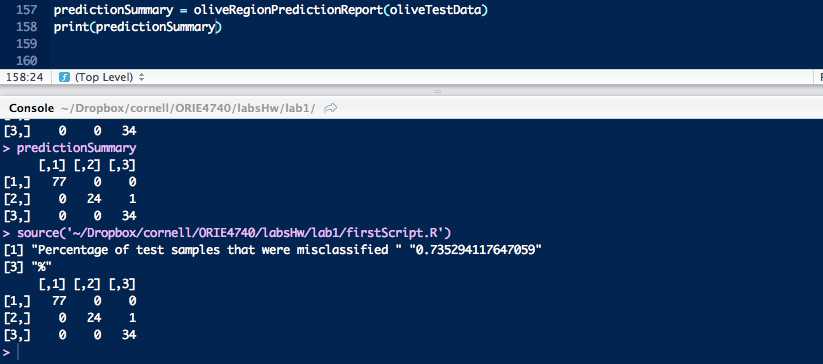
(code found in part A of Appendix)

My classification rule used 3 predictors to predict for each of the 3 regions. First, eiconsenoic acid was used to predict for region 1. If the eiconsenoic acid percentage level was above 5, then my classification would determine region 1. Eiconsenoic acid was used becase from the training data, we found that a significant portion of region 1 oils had eiconsenoic acid percentage levels above 5, while none of the samples from region 2 or 3 had eicosenoic acid percentage levels above 5.

If the eiconsenoic acid percentage level was below 5, then we do not conclude if the oil was from region 1. Rather, we then proceed to examine the linoleic acid percentage to distinguish between regions 2 and 3. It was found that linoleic acid levels were significantly different between samples from regions 2 and 3. And so in my classification, samples with a linoleic acid percentage less than 1040 was classified as belonging to region 3. The rest of the samples that failed to meet the linoleic acid and eicosenoic acid thresholds were classified as region 2

What was the error rate?

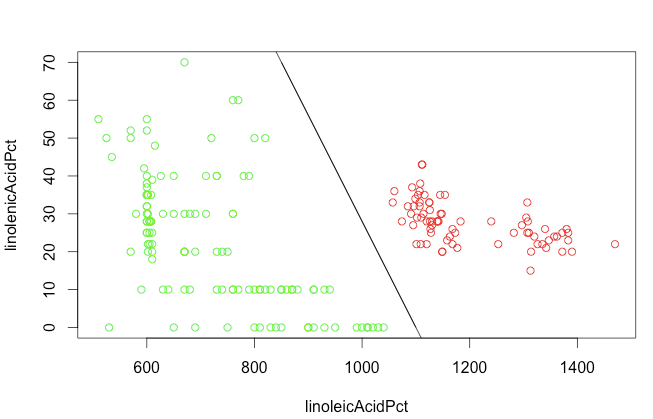
The error rate, as indicate by my code, was 0.73529%. This means that my classification tree made 1 error out of the 136 samples in the test data (as shown below by the summary of prediction errors).



The error occurred when the code predicted that a sample belonged to region 2 when in reality the sample had belonged to region 3

Part 2 (code found in part b)

After creating scatter plots that examined the relationship between the acids, we found that a scatter plot of linoleic and linolenic acid for regions 2 and presented us with a clustering opportunity as shown below:



We found that an approximate linear function could be used to distinguish between the 2 regions:

If sample.linolenicAcidPct > 308 -0.28\* sample.linoleicAcidPct then we classify the sample as belonging to region 2, otherwise we classify the sample as belonging to region 3.

For or classification tree in part 2, we also used eiconsenoic acid percentage as a predictor for region 1 (as we did in part 1). After determining whether or not a sample belonged to region 1 or not, we then applied the linear classifier to determine whether or not a sample belonged to region 2 (sample.linolenicAcidPct > 308 -0.28\* sample.linoleicAcidPct). The rest of the samples were classifed as region 3.

After updating our classification tree, we found that our error rate was 0% (no errors were made on the test data when applying our new classification tree)

Asdfasdfksldfk

A

K

Lsdfk

Part 3

Alsdf

Asdfkkaj

Appendix

# Part A

require(graphics)

rm(list = ls())

setwd("~/Dropbox/cornell/ORIE4740/labsHw/lab1")

oliveData = read.table("olive-train.dat")

oliveTestData = read.table("olive-test.dat")

colnames(oliveData)= c("region","area","palmiticAcidPct","palmitoleicAcidPct","stearicAcidPct","oleicAcidPct","linoleicAcidPct","linolenicAcidPct","arachidicAcidPct","eicosenoicAcidPct")

colnames(oliveTestData)= c("region","area","palmiticAcidPct","palmitoleicAcidPct","stearicAcidPct","oleicAcidPct","linoleicAcidPct","linolenicAcidPct","arachidicAcidPct","eicosenoicAcidPct")

dim(oliveData)

#print(oliveData[1:10,])

oliveData$region = as.factor(oliveData$region)

oliveData$area = as.factor(oliveData$area)

oliveTestData$region = as.factor(oliveTestData$region)

oliveTestData$area = as.factor(oliveTestData$area)

#reformatting all of the axes for the density plots

par(mfrow=c(3,1))

someFool1 = density(as.numeric(oliveData$palmiticAcidPct[oliveData$region == 1]))

someFool2 = density(as.numeric(oliveData$palmiticAcidPct[oliveData$region == 2]))

someFool3 = density(as.numeric(oliveData$palmiticAcidPct[oliveData$region == 3]))

maximumY = c(someFool1$y,someFool2$y,someFool3$y)

maximumY = max(maximumY);

plot(density(as.numeric(oliveData$palmiticAcidPct[oliveData$region == 1])),xlim = c(500, 1800),ylim = c(0,maximumY),main = "palmiticAcid & region 1")

plot(density(as.numeric(oliveData$palmiticAcidPct[oliveData$region == 2])),xlim = c(500, 1800),ylim = c(0,maximumY),main = "palmiticAcid & region 2")

plot(density(as.numeric(oliveData$palmiticAcidPct[oliveData$region == 3])),xlim = c(500, 1800),ylim = c(0,maximumY),main = "palmiticAcid & region 3")

par(mfrow=c(3,1)) #reformatting all of the axes for the density plots

someFool1 = density(as.numeric(oliveData$palmitoleicAcidPct[oliveData$region == 1]))

someFool2 = density(as.numeric(oliveData$palmitoleicAcidPct[oliveData$region == 2]))

someFool3 = density(as.numeric(oliveData$palmitoleicAcidPct[oliveData$region == 3]))

maximumY = c(someFool1$y,someFool2$y,someFool3$y)

maximumY = max(maximumY);

plot(density(as.numeric(oliveData$palmitoleicAcidPct[oliveData$region == 1])),xlim = c(0, 350),ylim = c(0,maximumY),main = "palmitoleicAcidPct & region 1")

plot(density(as.numeric(oliveData$palmitoleicAcidPct[oliveData$region == 2])),xlim = c(0, 350),ylim = c(0,maximumY),main = "palmitoleicAcidPct & region 2")

plot(density(as.numeric(oliveData$palmitoleicAcidPct[oliveData$region == 3])),xlim = c(0, 350),ylim = c(0,maximumY),main = "palmitoleicAcidPct & region 3")

par(mfrow=c(3,1)) #reformatting all of the axes for the density plots

someFool1 = density(as.numeric(oliveData$stearicAcidPct[oliveData$region == 1]))

someFool2 = density(as.numeric(oliveData$stearicAcidPct[oliveData$region == 2]))

someFool3 = density(as.numeric(oliveData$stearicAcidPct[oliveData$region == 3]))

maximumY = c(someFool1$y,someFool2$y,someFool3$y)

maximumY = max(maximumY);

plot(density(as.numeric(oliveData$stearicAcidPct[oliveData$region == 1])),xlim = c(120, 400),ylim = c(0,maximumY),main = "stearicAcidPct & region 1")

plot(density(as.numeric(oliveData$stearicAcidPct[oliveData$region == 2])),xlim = c(120, 400),ylim = c(0,maximumY),main = "stearicAcidPct & region 2")

plot(density(as.numeric(oliveData$stearicAcidPct[oliveData$region == 3])),xlim = c(120, 400),ylim = c(0,maximumY),main = "stearicAcidPct & region 3")

par(mfrow=c(3,1)) #reformatting all of the axes for the density plots

someFool1 = density(as.numeric(oliveData$oleicAcidPct[oliveData$region == 1]))

someFool2 = density(as.numeric(oliveData$oleicAcidPct[oliveData$region == 2]))

someFool3 = density(as.numeric(oliveData$oleicAcidPct[oliveData$region == 3]))

maximumY = c(someFool1$y,someFool2$y,someFool3$y)

maximumY = max(maximumY);

plot(density(as.numeric(oliveData$oleicAcidPct[oliveData$region == 1])),xlim = c(6000, 8700),ylim = c(0,maximumY),main = "oleicAcidPct & region 1")

plot(density(as.numeric(oliveData$oleicAcidPct[oliveData$region == 2])),xlim = c(6000, 8700),ylim = c(0,maximumY),main = "oleicAcidPct & region 2")

plot(density(as.numeric(oliveData$oleicAcidPct[oliveData$region == 3])),xlim = c(6000, 8700),ylim = c(0,maximumY),main = "oleicAcidPct & region 3")

par(mfrow=c(3,1)) #reformatting all of the axes for the density plots

someFool1 = density(as.numeric(oliveData$linoleicAcidPct[oliveData$region == 1]))

someFool2 = density(as.numeric(oliveData$linoleicAcidPct[oliveData$region == 2]))

someFool3 = density(as.numeric(oliveData$linoleicAcidPct[oliveData$region == 3]))

maximumY = c(someFool1$y,someFool2$y,someFool3$y)

maximumY = max(maximumY);

plot(density(as.numeric(oliveData$linoleicAcidPct[oliveData$region == 1])),xlim = c(200, 1700),ylim = c(0,maximumY),main = "linoleicAcidPct & region 1")

plot(density(as.numeric(oliveData$linoleicAcidPct[oliveData$region == 2])),xlim = c(200, 1700),ylim = c(0,maximumY),main = "linoleicAcidPct & region 2")

plot(density(as.numeric(oliveData$linoleicAcidPct[oliveData$region == 3])),xlim = c(200, 1700),ylim = c(0,maximumY),main = "linoleicAcidPct & region 3")

par(mfrow=c(3,1)) #reformatting all of the axes for the density plots

someFool1 = density(as.numeric(oliveData$linolenicAcidPct[oliveData$region == 1]))

someFool2 = density(as.numeric(oliveData$linolenicAcidPct[oliveData$region == 2]))

someFool3 = density(as.numeric(oliveData$linolenicAcidPct[oliveData$region == 3]))

maximumY = c(someFool1$y,someFool2$y,someFool3$y)

maximumY = max(maximumY);

plot(density(as.numeric(oliveData$linolenicAcidPct[oliveData$region == 1])),xlim = c(0, 90),ylim = c(0,maximumY),main = "linolenicAcidPct & region 1")

plot(density(as.numeric(oliveData$linolenicAcidPct[oliveData$region == 2])),xlim = c(0, 90),ylim = c(0,maximumY),main = "linolenicAcidPct & region 2")

plot(density(as.numeric(oliveData$linolenicAcidPct[oliveData$region == 3])),xlim = c(0, 90),ylim = c(0,maximumY),main = "linolenicAcidPct & region 3")

par(mfrow=c(3,1)) #reformatting all of the axes for the density plots

someFool1 = density(as.numeric(oliveData$arachidicAcidPct[oliveData$region == 1]))

someFool2 = density(as.numeric(oliveData$arachidicAcidPct[oliveData$region == 2]))

someFool3 = density(as.numeric(oliveData$arachidicAcidPct[oliveData$region == 3]))

maximumY = c(someFool1$y,someFool2$y,someFool3$y)

maximumY = max(maximumY);

plot(density(as.numeric(oliveData$arachidicAcidPct[oliveData$region == 1])),xlim = c(0, 120),ylim = c(0,maximumY),main = "arachidicAcidPct & region 1")

plot(density(as.numeric(oliveData$arachidicAcidPct[oliveData$region == 2])),xlim = c(0, 120),ylim = c(0,maximumY),main = "arachidicAcidPct & region 2")

plot(density(as.numeric(oliveData$arachidicAcidPct[oliveData$region == 3])),xlim = c(0, 120),ylim = c(0,maximumY),main = "arachidicAcidPct & region 3")

par(mfrow=c(3,1)) #reformatting all of the axes for the density plots

someFool1 = density(as.numeric(oliveData$eicosenoicAcidPct[oliveData$region == 1]))

someFool2 = density(as.numeric(oliveData$eicosenoicAcidPct[oliveData$region == 2]))

someFool3 = density(as.numeric(oliveData$eicosenoicAcidPct[oliveData$region == 3]))

maximumY = c(someFool1$y,someFool2$y,someFool3$y)

maximumY = max(maximumY);

plot(density(as.numeric(oliveData$eicosenoicAcidPct[oliveData$region == 1])),xlim = c(0, 70),ylim = c(0,maximumY),main = "eicosenoicAcidPct & region 1")

plot(density(as.numeric(oliveData$eicosenoicAcidPct[oliveData$region == 2])),xlim = c(0, 70),ylim = c(0,maximumY),main = "eicosenoicAcidPct & region 2")

plot(density(as.numeric(oliveData$eicosenoicAcidPct[oliveData$region == 3])),xlim = c(0, 70),ylim = c(0,maximumY),main = "eicosenoicAcidPct & region 3")

#eiconsenoic for region 1

oliveRegionPredict <- function(testDataFrame){

inputOliveData = testDataFrame

predictionArray = array(0,c(dim(inputOliveData)[1],1))

for( i in 1:dim(inputOliveData)[1] ) # how many rows there are

{

loopSample = inputOliveData[i,];

if ( loopSample$eicosenoicAcidPct >5 )

{

predictionArray[i] = 1;

}

else

{

if ( loopSample$linoleicAcidPct < 1040 )

{

predictionArray[i] = 3;

}

else

{

predictionArray[i] = 2

}

}

}

return (predictionArray)

} #Returns an array of predictions

oliveRegionPredictionReport <- function (testDataFrame) {

reportMatrix = array(0,dim = c(3,3))

predictionArray = oliveRegionPredict(testDataFrame)

errors = 0

for (i in 1:length(predictionArray))

{

reportMatrix[predictionArray[i],testDataFrame$region[i]] = reportMatrix[predictionArray[i],testDataFrame$region[i]] + 1

if (predictionArray[i] != testDataFrame$region[i])

{

errors = errors + 1

}

}

print(c("Percentage of test samples that were misclassified ",100\*errors/length(predictionArray), "%"))

return (reportMatrix)

}

pika = as.numeric(oliveTestData$region)

pikachu = as.numeric(t(oliveRegionPredict(oliveTestData)))

predictionSummary = oliveRegionPredictionReport(oliveTestData)

print(predictionSummary)

## Code for Part B

require(graphics)

rm(list = ls())

setwd("~/Dropbox/cornell/ORIE4740/labsHw/lab1")

oliveData = read.table("olive-train.dat")

oliveTestData = read.table("olive-test.dat")

colnames(oliveData)= c("region","area","palmiticAcidPct","palmitoleicAcidPct","stearicAcidPct","oleicAcidPct","linoleicAcidPct","linolenicAcidPct","arachidicAcidPct","eicosenoicAcidPct")

colnames(oliveTestData)= c("region","area","palmiticAcidPct","palmitoleicAcidPct","stearicAcidPct","oleicAcidPct","linoleicAcidPct","linolenicAcidPct","arachidicAcidPct","eicosenoicAcidPct")

oliveData = oliveData[ oliveData$region != 1,]

dim(oliveData)

#print(oliveData[1:10,])

oliveData$region = as.factor(oliveData$region)

oliveData$area = as.factor(oliveData$area)

oliveTestData$region = as.factor(oliveTestData$region)

oliveTestData$area = as.factor(oliveTestData$area)

fit <- lm(oliveData$oleicAcidPct~oliveData$linoleicAcidPct, data=faithful)

par(mfrow=c(1,1))

#plot(oliveData$oleicAcidPct,oliveData$linoleicAcidPct, main="linoleic vs oleic",col = c("red","green")[ oliveData$region ],xlim = c(7250,7750),ylim = c(800,1200))

plot(oliveData$oleicAcidPct,oliveData$linoleicAcidPct, main="linoleic vs oleic",col = c("red","green")[ oliveData$region ])

#lines( oliveData$oleicAcidPct, oliveData$linoleicAcidPct, fitted(fit), col="blue")

##This Part scatterplots out all of the different combinations of variables to show you the potential clusters to be found when looking

par(mfrow=c(2,5))

for(i in 3:10)

{

#par(mfrow = c(6,7))

for (j in 3:10)

{

if (i != j)

{

xName = names(oliveData)[i]

yName = names(oliveData)[j]

plot(oliveData[[i]],oliveData[[j]],xlab = xName,ylab = yName,col = c("red","green")[ oliveData$region ])

}

}

}

par(mfrow = c(1,1))

plot(oliveData$linoleicAcidPct,oliveData$linolenicAcidPct,xlab = "linoleicAcidPct",ylab = "linolenicAcidPct",col = c("red","green")[ oliveData$region ])

abline(308,-0.28)

lines(c(850,1100),c(70,0))

oliveRegionPredict <- function(testDataFrame){

inputOliveData = testDataFrame

predictionArray = array(0,c(dim(inputOliveData)[1],1))

for( i in 1:dim(inputOliveData)[1] ) # how many rows there are

{

loopSample = inputOliveData[i,];

if ( loopSample$eicosenoicAcidPct >5 )

{

predictionArray[i] = 1;

}

else

{

if ( loopSample$linolenicAcidPct > -0.28\*loopSample$linoleicAcidPct + 308 )

{

predictionArray[i] = 2;

}

else

{

predictionArray[i] = 3

}

}

}

return (predictionArray)

} #Returns an array of predictions

oliveRegionPredictionReport <- function (testDataFrame) {

reportMatrix = array(0,dim = c(3,3))

predictionArray = oliveRegionPredict(testDataFrame)

errors = 0

for (i in 1:length(predictionArray))

{

reportMatrix[predictionArray[i],testDataFrame$region[i]] = reportMatrix[predictionArray[i],testDataFrame$region[i]] + 1

if (predictionArray[i] != testDataFrame$region[i])

{

errors = errors + 1

}

}

print(c("Percentage of test samples that were misclassified ",100\*errors/length(predictionArray), "%"))

print(reportMatrix)

return (reportMatrix)

}

predictionReport = oliveRegionPredictionReport(oliveTestData)

#

#reformatting all of the axes for the density plots

# par(mfrow=c(3,1))

# someFool1 = density(as.numeric(oliveData$palmiticAcidPct[oliveData$region == 1]))

# someFool2 = density(as.numeric(oliveData$palmiticAcidPct[oliveData$region == 2]))

# someFool3 = density(as.numeric(oliveData$palmiticAcidPct[oliveData$region == 3]))

# maximumY = c(someFool1$y,someFool2$y,someFool3$y)

# maximumY = max(maximumY);

#

# plot(density(as.numeric(oliveData$palmiticAcidPct[oliveData$region == 1])),xlim = c(500, 1800),ylim = c(0,maximumY),main = "palmiticAcid & region 1")

# plot(density(as.numeric(oliveData$palmiticAcidPct[oliveData$region == 2])),xlim = c(500, 1800),ylim = c(0,maximumY),main = "palmiticAcid & region 2")

# plot(density(as.numeric(oliveData$palmiticAcidPct[oliveData$region == 3])),xlim = c(500, 1800),ylim = c(0,maximumY),main = "palmiticAcid & region 3")

#

# par(mfrow=c(3,1)) #reformatting all of the axes for the density plots

# someFool1 = density(as.numeric(oliveData$palmitoleicAcidPct[oliveData$region == 1]))

# someFool2 = density(as.numeric(oliveData$palmitoleicAcidPct[oliveData$region == 2]))

# someFool3 = density(as.numeric(oliveData$palmitoleicAcidPct[oliveData$region == 3]))

# maximumY = c(someFool1$y,someFool2$y,someFool3$y)

# maximumY = max(maximumY);

#

# plot(density(as.numeric(oliveData$palmitoleicAcidPct[oliveData$region == 1])),xlim = c(0, 350),ylim = c(0,maximumY),main = "palmitoleicAcidPct & region 1")

# plot(density(as.numeric(oliveData$palmitoleicAcidPct[oliveData$region == 2])),xlim = c(0, 350),ylim = c(0,maximumY),main = "palmitoleicAcidPct & region 2")

# plot(density(as.numeric(oliveData$palmitoleicAcidPct[oliveData$region == 3])),xlim = c(0, 350),ylim = c(0,maximumY),main = "palmitoleicAcidPct & region 3")

#

# par(mfrow=c(3,1)) #reformatting all of the axes for the density plots

# someFool1 = density(as.numeric(oliveData$stearicAcidPct[oliveData$region == 1]))

# someFool2 = density(as.numeric(oliveData$stearicAcidPct[oliveData$region == 2]))

# someFool3 = density(as.numeric(oliveData$stearicAcidPct[oliveData$region == 3]))

# maximumY = c(someFool1$y,someFool2$y,someFool3$y)

# maximumY = max(maximumY);

#

# plot(density(as.numeric(oliveData$stearicAcidPct[oliveData$region == 1])),xlim = c(120, 400),ylim = c(0,maximumY),main = "stearicAcidPct & region 1")

# plot(density(as.numeric(oliveData$stearicAcidPct[oliveData$region == 2])),xlim = c(120, 400),ylim = c(0,maximumY),main = "stearicAcidPct & region 2")

# plot(density(as.numeric(oliveData$stearicAcidPct[oliveData$region == 3])),xlim = c(120, 400),ylim = c(0,maximumY),main = "stearicAcidPct & region 3")

#

# par(mfrow=c(3,1)) #reformatting all of the axes for the density plots

# someFool1 = density(as.numeric(oliveData$oleicAcidPct[oliveData$region == 1]))

# someFool2 = density(as.numeric(oliveData$oleicAcidPct[oliveData$region == 2]))

# someFool3 = density(as.numeric(oliveData$oleicAcidPct[oliveData$region == 3]))

# maximumY = c(someFool1$y,someFool2$y,someFool3$y)

# maximumY = max(maximumY);

#

# plot(density(as.numeric(oliveData$oleicAcidPct[oliveData$region == 1])),xlim = c(6000, 8700),ylim = c(0,maximumY),main = "oleicAcidPct & region 1")

# plot(density(as.numeric(oliveData$oleicAcidPct[oliveData$region == 2])),xlim = c(6000, 8700),ylim = c(0,maximumY),main = "oleicAcidPct & region 2")

# plot(density(as.numeric(oliveData$oleicAcidPct[oliveData$region == 3])),xlim = c(6000, 8700),ylim = c(0,maximumY),main = "oleicAcidPct & region 3")

#

# par(mfrow=c(3,1)) #reformatting all of the axes for the density plots

# someFool1 = density(as.numeric(oliveData$linoleicAcidPct[oliveData$region == 1]))

# someFool2 = density(as.numeric(oliveData$linoleicAcidPct[oliveData$region == 2]))

# someFool3 = density(as.numeric(oliveData$linoleicAcidPct[oliveData$region == 3]))

# maximumY = c(someFool1$y,someFool2$y,someFool3$y)

# maximumY = max(maximumY);

#

#

# plot(density(as.numeric(oliveData$linoleicAcidPct[oliveData$region == 1])),xlim = c(200, 1700),ylim = c(0,maximumY),main = "linoleicAcidPct & region 1")

# plot(density(as.numeric(oliveData$linoleicAcidPct[oliveData$region == 2])),xlim = c(200, 1700),ylim = c(0,maximumY),main = "linoleicAcidPct & region 2")

# plot(density(as.numeric(oliveData$linoleicAcidPct[oliveData$region == 3])),xlim = c(200, 1700),ylim = c(0,maximumY),main = "linoleicAcidPct & region 3")

#

# par(mfrow=c(3,1)) #reformatting all of the axes for the density plots

# someFool1 = density(as.numeric(oliveData$linolenicAcidPct[oliveData$region == 1]))

# someFool2 = density(as.numeric(oliveData$linolenicAcidPct[oliveData$region == 2]))

# someFool3 = density(as.numeric(oliveData$linolenicAcidPct[oliveData$region == 3]))

# maximumY = c(someFool1$y,someFool2$y,someFool3$y)

# maximumY = max(maximumY);

#

# plot(density(as.numeric(oliveData$linolenicAcidPct[oliveData$region == 1])),xlim = c(0, 90),ylim = c(0,maximumY),main = "linolenicAcidPct & region 1")

# plot(density(as.numeric(oliveData$linolenicAcidPct[oliveData$region == 2])),xlim = c(0, 90),ylim = c(0,maximumY),main = "linolenicAcidPct & region 2")

# plot(density(as.numeric(oliveData$linolenicAcidPct[oliveData$region == 3])),xlim = c(0, 90),ylim = c(0,maximumY),main = "linolenicAcidPct & region 3")

#

# par(mfrow=c(3,1)) #reformatting all of the axes for the density plots

# someFool1 = density(as.numeric(oliveData$arachidicAcidPct[oliveData$region == 1]))

# someFool2 = density(as.numeric(oliveData$arachidicAcidPct[oliveData$region == 2]))

# someFool3 = density(as.numeric(oliveData$arachidicAcidPct[oliveData$region == 3]))

# maximumY = c(someFool1$y,someFool2$y,someFool3$y)

# maximumY = max(maximumY);

#

# plot(density(as.numeric(oliveData$arachidicAcidPct[oliveData$region == 1])),xlim = c(0, 120),ylim = c(0,maximumY),main = "arachidicAcidPct & region 1")

# plot(density(as.numeric(oliveData$arachidicAcidPct[oliveData$region == 2])),xlim = c(0, 120),ylim = c(0,maximumY),main = "arachidicAcidPct & region 2")

# plot(density(as.numeric(oliveData$arachidicAcidPct[oliveData$region == 3])),xlim = c(0, 120),ylim = c(0,maximumY),main = "arachidicAcidPct & region 3")

#

# par(mfrow=c(3,1)) #reformatting all of the axes for the density plots

# someFool1 = density(as.numeric(oliveData$eicosenoicAcidPct[oliveData$region == 1]))

# someFool2 = density(as.numeric(oliveData$eicosenoicAcidPct[oliveData$region == 2]))

# someFool3 = density(as.numeric(oliveData$eicosenoicAcidPct[oliveData$region == 3]))

# maximumY = c(someFool1$y,someFool2$y,someFool3$y)

# maximumY = max(maximumY);

#

# plot(density(as.numeric(oliveData$eicosenoicAcidPct[oliveData$region == 1])),xlim = c(0, 70),ylim = c(0,maximumY),main = "eicosenoicAcidPct & region 1")

# plot(density(as.numeric(oliveData$eicosenoicAcidPct[oliveData$region == 2])),xlim = c(0, 70),ylim = c(0,maximumY),main = "eicosenoicAcidPct & region 2")

# plot(density(as.numeric(oliveData$eicosenoicAcidPct[oliveData$region == 3])),xlim = c(0, 70),ylim = c(0,maximumY),main = "eicosenoicAcidPct & region 3")

#

# #eiconsenoic for region 1

#

# oliveRegionPredict <- function(testDataFrame){

# inputOliveData = testDataFrame

# predictionArray = array(0,c(dim(inputOliveData)[1],1))

#

# for( i in 1:dim(inputOliveData)[1] ) # how many rows there are

# {

# loopSample = inputOliveData[i,];

# if ( loopSample$eicosenoicAcidPct >5 )

# {

# predictionArray[i] = 1;

# }

# else

# {

# if ( loopSample$linoleicAcidPct < 1040 )

# {

# predictionArray[i] = 3;

# }

# else

# {

# predictionArray[i] = 2

# }

# }

# }

#

# return (predictionArray)

# } #Returns an array of predictions

# oliveRegionPredictionReport <- function (testDataFrame) {

#

# reportMatrix = array(0,dim = c(3,3))

# predictionArray = oliveRegionPredict(testDataFrame)

# errors = 0

#

# for (i in 1:length(predictionArray))

# {

# reportMatrix[predictionArray[i],testDataFrame$region[i]] = reportMatrix[predictionArray[i],testDataFrame$region[i]] + 1

# if (predictionArray[i] != testDataFrame$region[i])

# {

# errors = errors + 1

# }

# }

# print(c("Percentage of test samples that were misclassified ",100\*errors/length(predictionArray), "%"))

# return (reportMatrix)

# }

#

#

# pika = as.numeric(oliveTestData$region)

# pikachu = as.numeric(t(oliveRegionPredict(oliveTestData)))

#

# predictionSummary = oliveRegionPredictionReport(oliveTestData)

#linoleic and oleic