Rebecca Leu

Northeastern University

Data Mining

PART 1 – INTRODUCTION AND THE DATA SET

In this assignment we will be data mining using decision trees, k-means clusters and Density-based clusters. I have chosen the Glass dataset in the “mlbench” package. The dataset looks at different elements to determine the type of glass. All the elements play a part in determining the type of glass, but some of the elements are more crucial than others as we will see from the classification and clustering we will be completing. This dataset has 214 observations with 10 Variables. The 10 variables are:

[,1] RI refractive index

[,2] Na Sodium

[,3] Mg Magnesium

[,4] Al Aluminum

[,5] Si Silicon

[,6] K Potassium

[,7] Ca Calcium

[,8] Ba Barium

[,9] Fe Iron

[,10] Type Type of glass (class attribute)

The type has 6 levels of glass labeled 1,2,3,5,6, and 7.

We begin by loading in the Glass dataset and reviewing the data using the str() code.

> #the glass dataset

> install.packages("mlbench")

> library(mlbench)

> data(Glass)

> ?Glass

> str(Glass)

'data.frame': 214 obs. of 10 variables:

$ RI : num 1.52 1.52 1.52 1.52 1.52 ...

$ Na : num 13.6 13.9 13.5 13.2 13.3 ...

$ Mg : num 4.49 3.6 3.55 3.69 3.62 3.61 3.6 3.61 3.58 3.6 ...

$ Al : num 1.1 1.36 1.54 1.29 1.24 1.62 1.14 1.05 1.37 1.36 ...

$ Si : num 71.8 72.7 73 72.6 73.1 ...

$ K : num 0.06 0.48 0.39 0.57 0.55 0.64 0.58 0.57 0.56 0.57 ...

$ Ca : num 8.75 7.83 7.78 8.22 8.07 8.07 8.17 8.24 8.3 8.4 ...

$ Ba : num 0 0 0 0 0 0 0 0 0 0 ...

$ Fe : num 0 0 0 0 0 0.26 0 0 0 0.11 ...

$ Type: Factor w/ 6 levels "1","2","3","5",..: 1 1 1 1 1 1 1 1 1 1 ...

PART 2 – SPLITING THE DATA

The first step is the split the data into training and test datasets so that the decision tree will formulate correctly.

> #split into training and test datasets

> set.seed(1234)

> ind<-sample(2, nrow(Glass), replace=T, prob=c(0.7, 0.3))

> Glass.train<-Glass[ind==1,]

> Glass.test <- Glass[ind==2, ]

PART 3 – BUILD A DECISION TREE

Next, we build the decision tree. This decision tree is a conditional inference tree (ctree) and we build it using “party”. Tree growth is based on statistical stopping rules so pruning is not required. We can see based off of this tree that Mg and Ca are two of the biggest qualifiers for the types of glass. This decision tree splits up the glass types into 7 nodes based off of the Mg, Ca, Ba, and Al variables only.

>

> #build a decision tree

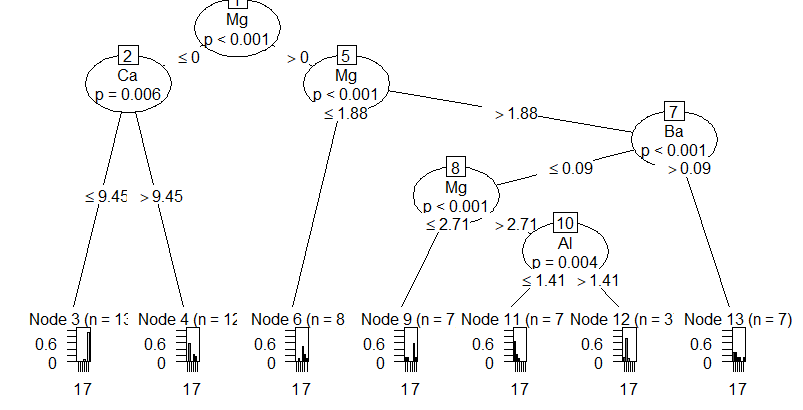
> library(party)

> Glass.formula <- Type ~ RI + Na +

+ Mg + Al + Si + K + Ca + Ba + Fe

> Glass.ctree <- ctree(Glass.formula, data=Glass.train)

> plot(Glass.ctree)



Part 4 – Predict using the test data

The test data has 59 observations of the 10 variables. Using the test data we check the prediction result.

> # predict on test data

> pred <- predict(Glass.ctree, newdata = Glass.test)

> # check prediction result

> table(pred, Glass.test$Type)

pred 1 2 3 5 6 7

1 14 8 5 0 0 1

2 1 12 0 2 0 0

3 0 0 0 0 0 0

5 0 1 0 1 0 1

6 0 0 0 0 0 0

7 0 0 0 1 1 11

Part 5 – K means clustering

K means clustering tries to cluster data based on their similarity. the algorithm finds patters in the data. for K means we do have to specity the number of clusters we want to create.

Here we are splitting the data into 6 clusters since we know there are 6 types of glass in the dataset. Not all datasets will have a clear number of clusters to create. For the plot I am using Ca and Mg as the x and y. The centers of the clusters are marked by the large \* with the color coding we can clearly see the clusters that have been created. For this dataset, k means clustering works well.

> set.seed(8953)

> Glass2 <- Glass

> # remove class IDs

> Glass2$Type <- NULL

> # k-means clustering

> Glass.kmeans <- kmeans(Glass2, 6)

> # check result

> table(Glass$Type, Glass.kmeans$cluster)

1 2 3 4 5 6

1 0 0 48 22 0 0

2 0 10 59 5 2 0

3 0 0 11 6 0 0

5 1 7 0 2 1 2

6 3 2 0 4 0 0

7 24 0 1 1 3 0

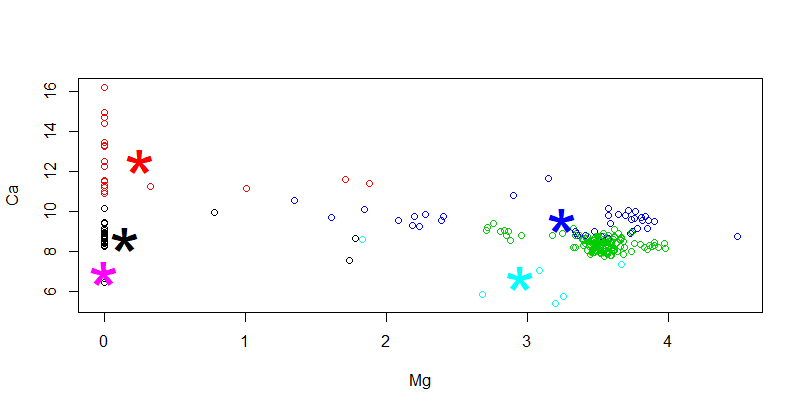
>

> #plot clusters and their centers

> plot(Glass2[c("Mg", "Ca")], col=Glass.kmeans$cluster)

> points(Glass.kmeans$centers[, c("Mg", "Ca")],

+ col=1:7, pch="\*", cex=5)



Part 6 – Density based clustering

Finally, we finish with a density-based cluster. Density based clusters differ from K means clusters as they do not need a predetermined number of clusters as a parameter. This helps if we are trying to find clusters in nonlinear shapes. Here we can find an area with higher density than the remaining area or the 0. With density clusters we can discover clusters in shape. The disadvantage though is there are no noise points detected and it is hard to determine the correct set of parameters.

> Glass2 <- Glass[-10] # remove class IDs

> # DBSCAN clustering

> ds <- dbscan(Glass2, eps = 0.42, MinPts = 5)

> # compare clusters with original class IDs

> table(ds$cluster, Glass$Type)

1 2 3 5 6 7

0 23 29 6 13 9 19

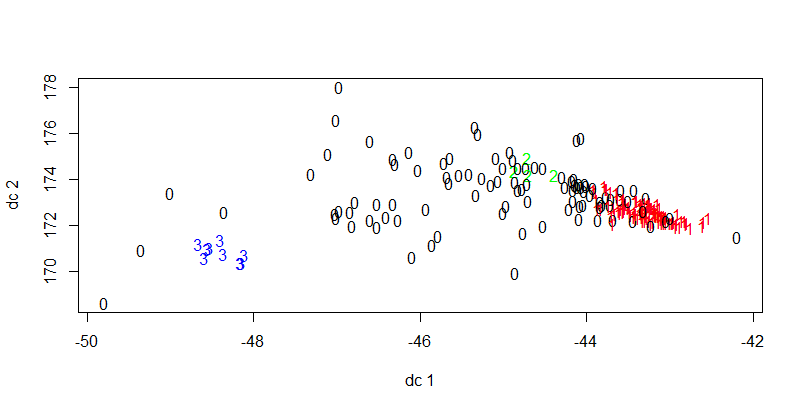
1 43 47 10 0 0 1

2 4 0 1 0 0 0

3 0 0 0 0 0 9

>

> plotcluster(Glass2, ds$cluster)

> 

References

Davo, & Gu, M. (2018, April 26). Building a classification tree in R. Retrieved May 3, 2020, from https://davetang.org/muse/2013/03/12/building-a-classification-tree-in-r/

Ivan, M. (n.d.). Density-based clustering in R. Retrieved May 3, 2020, from https://en.proft.me/2017/02/3/density-based-clustering-r/

K-means Cluster Analysis. (n.d.). Retrieved May 3, 2020, from https://uc-r.github.io/kmeans\_clustering

Kabacoff, R. (n.d.). Tree-Based Models. Retrieved May 3, 2020, from https://www.statmethods.net/advstats/cart.html

Kodali, T. (2015, December 28). K Means Clustering in R. Retrieved May 3, 2020, from https://www.r-bloggers.com/k-means-clustering-in-r/

Zhao, Y. (2015, May 8). Introduction to Data Mining with R. Retrieved May 3, 2020, from http://www.rdatamining.com/docs/introduction-to-data-mining-with-r