BASICS OF MACHINE LANGUAGE FINAL ASSIGNMENT

Harish Kumar uddandi

## R Markdown

library(caret)

## Loading required package: ggplot2

## Loading required package: lattice

library(ISLR)  
library(factoextra)

## Warning: package 'factoextra' was built under R version 4.2.2

## Welcome! Want to learn more? See two factoextra-related books at https://goo.gl/ve3WBa

library(tidyverse)

## Warning: package 'tidyverse' was built under R version 4.2.2

## ── Attaching packages  
## ───────────────────────────────────────  
## tidyverse 1.3.2 ──

## ✔ tibble 3.1.8 ✔ dplyr 1.0.10  
## ✔ tidyr 1.2.1 ✔ stringr 1.4.1   
## ✔ readr 2.1.3 ✔ forcats 0.5.2   
## ✔ purrr 0.3.4   
## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::lag() masks stats::lag()  
## ✖ purrr::lift() masks caret::lift()

library(flexclust)

## Warning: package 'flexclust' was built under R version 4.2.2

## Loading required package: grid  
## Loading required package: modeltools  
## Loading required package: stats4

library(cluster)  
library(gridExtra)

## Warning: package 'gridExtra' was built under R version 4.2.2

##   
## Attaching package: 'gridExtra'  
##   
## The following object is masked from 'package:dplyr':  
##   
## combine

#Loading dataset

data<-read.csv("C:/FALL/ML/fuel\_receipts\_costs\_eia923.csv")  
str(data)

## 'data.frame': 608565 obs. of 23 variables:  
## $ rowid : int 1 2 3 4 5 6 7 8 9 10 ...  
## $ plant\_id\_eia : int 3 3 3 7 7 7 7 8 8 8 ...  
## $ report\_date : chr "1/1/2008" "1/1/2008" "1/1/2008" "1/1/2008" ...  
## $ contract\_type\_code : chr "C" "C" "C" "C" ...  
## $ contract\_expiration\_date : chr "4/1/2008" "4/1/2008" "" "12/1/2015" ...  
## $ energy\_source\_code : chr "BIT" "BIT" "NG" "BIT" ...  
## $ fuel\_type\_code\_pudl : chr "coal" "coal" "gas" "coal" ...  
## $ fuel\_group\_code : chr "coal" "coal" "natural\_gas" "coal" ...  
## $ mine\_id\_pudl : int 0 0 NA 1 2 3 NA 4 4 1 ...  
## $ supplier\_name : chr "interocean coal" "interocean coal" "bay gas pipeline" "alabama coal" ...  
## $ fuel\_received\_units : int 259412 52241 2783619 25397 764 603 2341 8869 75442 206741 ...  
## $ fuel\_mmbtu\_per\_unit : num 23.1 22.8 1.04 24.61 24.45 ...  
## $ sulfur\_content\_pct : num 0.49 0.48 0 1.69 0.84 1.54 0 2.16 1.24 1.9 ...  
## $ ash\_content\_pct : num 5.4 5.7 0 14.7 15.5 14.6 0 15.4 11.9 15.4 ...  
## $ mercury\_content\_ppm : num NA NA NA NA NA NA NA NA NA NA ...  
## $ fuel\_cost\_per\_mmbtu : num 2.13 2.12 8.63 2.78 3.38 ...  
## $ primary\_transportation\_mode\_code : chr "RV" "RV" "PL" "TR" ...  
## $ secondary\_transportation\_mode\_code : chr "" "" "" "" ...  
## $ natural\_gas\_transport\_code : chr "firm" "firm" "firm" "firm" ...  
## $ natural\_gas\_delivery\_contract\_type\_code: chr "" "" "" "" ...  
## $ moisture\_content\_pct : num NA NA NA NA NA NA NA NA NA NA ...  
## $ chlorine\_content\_ppm : int NA NA NA NA NA NA NA NA NA NA ...  
## $ data\_maturity : chr "final" "final" "final" "final" ...

#selecting attributes

data\_df<-data[,c(8,12,13,14,16)]  
str(data\_df)

## 'data.frame': 608565 obs. of 5 variables:  
## $ fuel\_group\_code : chr "coal" "coal" "natural\_gas" "coal" ...  
## $ fuel\_mmbtu\_per\_unit: num 23.1 22.8 1.04 24.61 24.45 ...  
## $ sulfur\_content\_pct : num 0.49 0.48 0 1.69 0.84 1.54 0 2.16 1.24 1.9 ...  
## $ ash\_content\_pct : num 5.4 5.7 0 14.7 15.5 14.6 0 15.4 11.9 15.4 ...  
## $ fuel\_cost\_per\_mmbtu: num 2.13 2.12 8.63 2.78 3.38 ...

colMeans(is.na(data\_df))

## fuel\_group\_code fuel\_mmbtu\_per\_unit sulfur\_content\_pct ash\_content\_pct   
## 0.0000000 0.0000000 0.0000000 0.0000000   
## fuel\_cost\_per\_mmbtu   
## 0.3290363

#Data Imputing

data\_df$fuel\_cost\_per\_mmbtu[is.na(data\_df$fuel\_cost\_per\_mmbtu)] <- mean(data\_df$fuel\_cost\_per\_mmbtu, na.rm = TRUE)   
colMeans(is.na(data\_df)) #all the missing values has been imputed

## fuel\_group\_code fuel\_mmbtu\_per\_unit sulfur\_content\_pct ash\_content\_pct   
## 0 0 0 0   
## fuel\_cost\_per\_mmbtu   
## 0

#.sampling and partition of the data

set.seed(2424)  
sample\_data <- data\_df[sample(nrow(data\_df), size = 13500, replace = FALSE), ]  
train\_index <- createDataPartition(sample\_data$fuel\_cost\_per\_mmbtu, p=0.75, list = FALSE)  
train\_data<- sample\_data[train\_index,]  
test\_data<- sample\_data[-train\_index,]

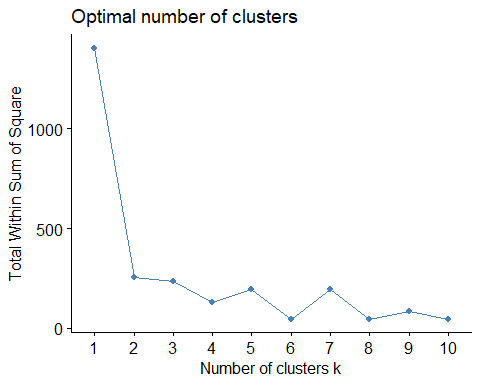
# normalization of the data.

cluster\_data <- train\_data %>% select( 'ash\_content\_pct', 'sulfur\_content\_pct','fuel\_mmbtu\_per\_unit','fuel\_cost\_per\_mmbtu')  
  
cluster\_train <- preProcess(cluster\_data, method = "range")  
cluster\_predict <- predict(cluster\_train, cluster\_data)  
  
summary(cluster\_predict)

## ash\_content\_pct sulfur\_content\_pct fuel\_mmbtu\_per\_unit fuel\_cost\_per\_mmbtu  
## Min. :0.00000 Min. :0.00000 Min. :0.00000 Min. :0.0000000   
## 1st Qu.:0.00000 1st Qu.:0.00000 1st Qu.:0.03155 1st Qu.:0.0002493   
## Median :0.00000 Median :0.00000 Median :0.03272 Median :0.0004398   
## Mean :0.05430 Mean :0.07506 Mean :0.29337 Mean :0.0009076   
## 3rd Qu.:0.08882 3rd Qu.:0.07091 3rd Qu.:0.59296 3rd Qu.:0.0013309   
## Max. :1.00000 Max. :1.00000 Max. :1.00000 Max. :1.0000000

#Elbow and Silhouette methods are used to find the optimal number of clusters. #Elbow Method

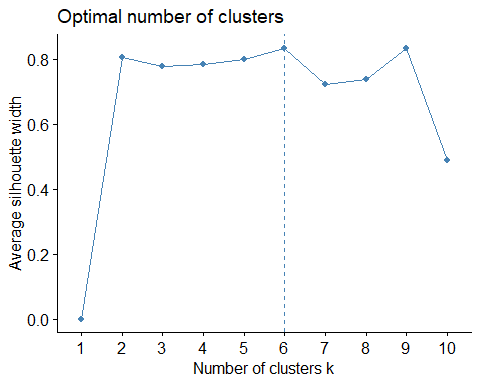
library(factoextra) # clustering algorithms & visualization  
library(flexclust)  
fviz\_nbclust(cluster\_predict,kmeans,method="wss")



#in the plot a clear elbow is at k = 2. Also as the above graph is not clear as it did not show any sharp point at 2. We can use 3 or 4 or 5 as the ‘K’ value too.

#Silhouttes method

#Silhouttes method  
fviz\_nbclust(cluster\_predict,kmeans,method="silhouette")



#As observed in elbow method, the optimal clusters identified as 2, but when we have used Silhouttes method, we got the value as 6. As the elbow method was not clear in determining the optimal cluster, we shall use Silhouttes method here #We have identified the number of clusters. Now we shall apply K-means algorithm

#Applying K-means Algorithm  
KMean\_chk <- kmeans(cluster\_predict, centers = 6, nstart = 25) #Number of restarts = 25

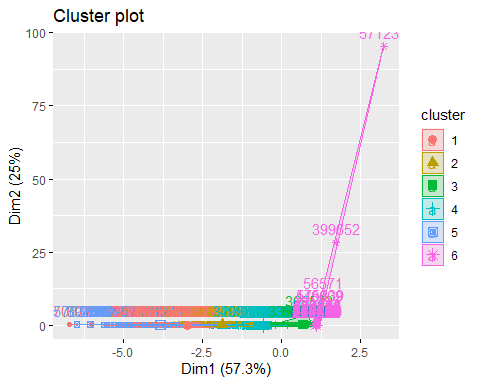
#Centers

KMean\_chk$centers

## ash\_content\_pct sulfur\_content\_pct fuel\_mmbtu\_per\_unit fuel\_cost\_per\_mmbtu  
## 1 1.465087e-01 0.46727151 0.79613549 0.0005826013  
## 2 1.648348e-01 0.15590428 0.80950591 0.0005792193  
## 3 6.710601e-05 0.02692564 0.19251079 0.0015399202  
## 4 8.646250e-02 0.04652277 0.57698279 0.0004115665  
## 5 6.037410e-01 0.19292950 0.44180138 0.0012258850  
## 6 0.000000e+00 0.00000000 0.03165675 0.0010494599

#Plotting the cluster using k K-means Algorithm

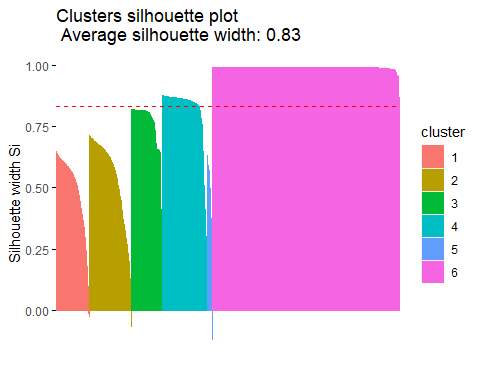
fviz\_cluster(KMean\_chk, data = cluster\_data)



#Plotting the Sillohuette average

si <- silhouette(KMean\_chk$cluster, dist(cluster\_predict))  
fviz\_silhouette(si)

## cluster size ave.sil.width  
## 1 1 962 0.49  
## 2 2 1266 0.53  
## 3 3 890 0.75  
## 4 4 1338 0.81  
## 5 5 140 0.45  
## 6 6 5531 0.99

 #Hence Si(silHouetee coeffient ) value > 0 , i.e 0.83, hence it is a good clustered.

#The final cluster

fcluster<- KMean\_chk$cluster  
f\_cluster<- cbind(train\_data, fcluster)  
f\_cluster$fcluster<-as.factor(f\_cluster$fcluster)  
head(f\_cluster)

## fuel\_group\_code fuel\_mmbtu\_per\_unit sulfur\_content\_pct ash\_content\_pct  
## 32310 coal 23.980 1.14 12.8  
## 591187 natural\_gas 1.036 0.00 0.0  
## 454496 natural\_gas 1.084 0.00 0.0  
## 181440 coal 17.630 0.22 4.6  
## 145791 natural\_gas 1.006 0.00 0.0  
## 224681 natural\_gas 1.030 0.00 0.0  
## fuel\_cost\_per\_mmbtu fcluster  
## 32310 3.73700 2  
## 591187 3.41000 6  
## 454496 14.18427 6  
## 181440 14.18427 4  
## 145791 4.63100 6  
## 224681 14.18427 6

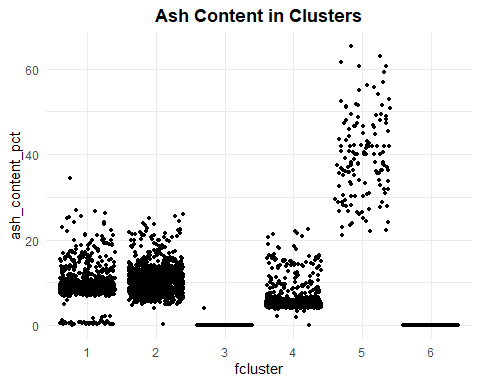
# We find the mean of all the quantitative variables

f\_cluster%>%group\_by(fcluster)%>%  
 summarize(  
 fuel\_mmbtu\_per\_unit=mean(fuel\_mmbtu\_per\_unit),  
 fuel\_cost\_per\_mmbtu=mean(fuel\_cost\_per\_mmbtu),  
 sulfur\_content=mean(sulfur\_content\_pct),  
 ash\_content=mean(ash\_content\_pct))

## # A tibble: 6 × 5  
## fcluster fuel\_mmbtu\_per\_unit fuel\_cost\_per\_mmbtu sulfur\_content ash\_content  
## <fct> <dbl> <dbl> <dbl> <dbl>  
## 1 1 24.0 6.27 3.23 9.57   
## 2 2 24.4 6.23 1.08 10.8   
## 3 3 5.85 16.4 0.186 0.00438  
## 4 4 17.4 4.46 0.321 5.65   
## 5 5 13.3 13.1 1.33 39.4   
## 6 6 1.03 11.2 0 0

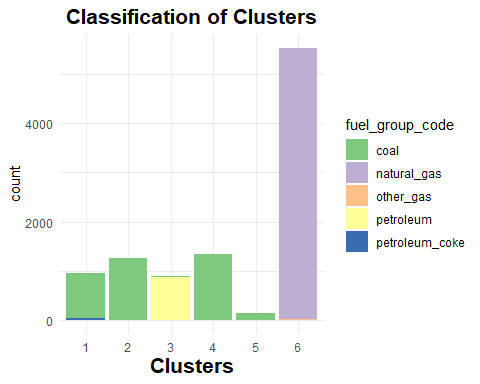
# Plotting number of ash contents

ggplot(f\_cluster) +  
 aes(x = fcluster, y = ash\_content\_pct) +  
 geom\_jitter(size = 1.2) +  
 labs(title = "Ash Content in Clusters") +  
 theme\_minimal() +  
 theme(plot.title = element\_text(face = "bold", hjust = 0.5))



# Plotting number of clusters

ggplot(f\_cluster) +  
  
 aes(x = fcluster, fill = fuel\_group\_code) +  
  
 geom\_bar() +  
  
 scale\_fill\_brewer(palette = "Accent", direction = 1) +  
  
 labs(x = "Clusters", title = "Classification of Clusters") +  
  
 theme\_minimal() +  
  
 theme(  
  
 plot.title = element\_text(size = 16L,  
  
 face = "bold",  
  
 hjust = 0.5),  
  
 axis.title.x = element\_text(size = 16L,  
  
 face = "bold")  
  
 )



#Use multiple-linear regression to determine the best set of variables to predict fuel\_cost\_per\_mmbtu.

#training data

ML\_df<- f\_cluster  
fuel<-ML\_df[,-c(1)]  
fuel\_ML<- preProcess(fuel, method = "range")  
fuel\_predict <- predict(fuel\_ML, fuel)  
head(fuel\_predict)

## fuel\_mmbtu\_per\_unit sulfur\_content\_pct ash\_content\_pct  
## 32310 0.79706520 0.16497829 0.1960184  
## 591187 0.03188261 0.00000000 0.0000000  
## 454496 0.03348341 0.00000000 0.0000000  
## 181440 0.58529265 0.03183792 0.0704441  
## 145791 0.03088211 0.00000000 0.0000000  
## 224681 0.03168251 0.00000000 0.0000000  
## fuel\_cost\_per\_mmbtu fcluster  
## 32310 0.0003430433 2  
## 591187 0.0003121240 6  
## 454496 0.0013308785 6  
## 181440 0.0013308785 4  
## 145791 0.0004275749 6  
## 224681 0.0013308785 6

#performing multiple linear regression model on training data

k<-fuel\_predict$fuel\_cost\_per\_mmbtu  
  
   
Z5<- fuel\_predict$fuel\_mmbtu\_per\_unit   
Z6<- fuel\_predict$sulfur\_content\_pct  
Z7<- fuel\_predict$ash\_content\_pct

model\_check <- lm(fuel\_cost\_per\_mmbtu~.,data=fuel\_predict)  
summary(model\_check)

##   
## Call:  
## lm(formula = fuel\_cost\_per\_mmbtu ~ ., data = fuel\_predict)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.00116 -0.00061 -0.00025 0.00028 0.99895   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)  
## (Intercept) -9.710e-04 3.467e-03 -0.280 0.779  
## fuel\_mmbtu\_per\_unit 1.489e-03 3.963e-03 0.376 0.707  
## sulfur\_content\_pct 3.760e-04 2.181e-03 0.172 0.863  
## ash\_content\_pct 1.316e-03 3.749e-03 0.351 0.726  
## fcluster2 6.968e-05 8.349e-04 0.083 0.933  
## fcluster3 2.214e-03 2.769e-03 0.799 0.424  
## fcluster4 3.924e-04 1.313e-03 0.299 0.765  
## fcluster5 6.722e-04 1.867e-03 0.360 0.719  
## fcluster6 1.973e-03 3.351e-03 0.589 0.556  
##   
## Residual standard error: 0.01047 on 10118 degrees of freedom  
## Multiple R-squared: 0.0009669, Adjusted R-squared: 0.000177   
## F-statistic: 1.224 on 8 and 10118 DF, p-value: 0.28

#Use the anova analysis

anova(model\_check)

## Analysis of Variance Table  
##   
## Response: fuel\_cost\_per\_mmbtu  
## Df Sum Sq Mean Sq F value Pr(>F)   
## fuel\_mmbtu\_per\_unit 1 0.00060 0.00059808 5.4523 0.01956 \*  
## sulfur\_content\_pct 1 0.00003 0.00002737 0.2496 0.61740   
## ash\_content\_pct 1 0.00001 0.00000652 0.0594 0.80745   
## fcluster 5 0.00044 0.00008845 0.8063 0.54488   
## Residuals 10118 1.10988 0.00010969   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

#Test data

Check\_df<- test\_data  
fuel<-Check\_df[,-c(1)]  
fuel\_chk<- preProcess(fuel, method = "range")  
fuel\_check <- predict(fuel\_chk, fuel)  
head(fuel\_check)

## fuel\_mmbtu\_per\_unit sulfur\_content\_pct ash\_content\_pct  
## 312063 0.03158950 0.00000000 0.0000000  
## 209115 0.03172322 0.00000000 0.0000000  
## 265611 0.44877152 0.04962406 0.5349183  
## 498345 0.03794083 0.00000000 0.0000000  
## 557162 0.86441584 0.35639098 0.1114413  
## 146003 0.03108808 0.00000000 0.0000000  
## fuel\_cost\_per\_mmbtu  
## 312063 2.297100e-04  
## 209115 2.297100e-04  
## 265611 2.297100e-04  
## 498345 4.187949e-05  
## 557162 2.845408e-05  
## 146003 1.295414e-04

#performing multiple linear regression model on test data

M<-fuel\_check$fuel\_cost\_per\_mmbtu  
  
C6<- fuel\_predict$fuel\_mmbtu\_per\_unit   
C7<- fuel\_predict$sulfur\_content\_pct  
C8<- fuel\_predict$ash\_content\_pct

model\_check1 <- lm(fuel\_cost\_per\_mmbtu~.,data=fuel\_check)  
summary(model\_check1)

##   
## Call:  
## lm(formula = fuel\_cost\_per\_mmbtu ~ ., data = fuel\_check)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.00066 -0.00059 -0.00044 -0.00009 0.99933   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 6.989e-04 4.077e-04 1.714 0.0865 .  
## fuel\_mmbtu\_per\_unit -9.279e-04 1.460e-03 -0.635 0.5252   
## sulfur\_content\_pct 3.370e-04 2.760e-03 0.122 0.9028   
## ash\_content\_pct -8.051e-05 3.871e-03 -0.021 0.9834   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.01723 on 3369 degrees of freedom  
## Multiple R-squared: 0.0002577, Adjusted R-squared: -0.0006326   
## F-statistic: 0.2895 on 3 and 3369 DF, p-value: 0.833

#Use the anova analysis to predict the model

anova(model\_check1)

## Analysis of Variance Table  
##   
## Response: fuel\_cost\_per\_mmbtu  
## Df Sum Sq Mean Sq F value Pr(>F)  
## fuel\_mmbtu\_per\_unit 1 0.00025 2.5330e-04 0.8534 0.3557  
## sulfur\_content\_pct 1 0.00000 4.3280e-06 0.0146 0.9039  
## ash\_content\_pct 1 0.00000 1.2800e-07 0.0004 0.9834  
## Residuals 3369 1.00001 2.9683e-04

The cluster information does not plays an important role to predict fuel\_cost\_per\_mmbtu, since the primary objective of my model is find the ash content, so cost is not playing crucial role.