Machine\_learning-ExtraCredits

Snehitha Anpur

2022-12-12

Energy\_Data=read.csv("D:\\MSBA\\rTutorial\\Rtutorial\\fuel\_receipts\_costs\_eia923.csv") # Reading csv file  
  
set.seed(1234) # Setting Seed value  
library(dplyr)

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

library(caret)

## Loading required package: ggplot2

## Warning: package 'ggplot2' was built under R version 4.2.2

## Loading required package: lattice

library(missForest)

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

library(corrplot)

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

## corrplot 0.92 loaded

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(fpc)

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

library(StatMatch)

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

## Loading required package: proxy

##   
## Attaching package: 'proxy'

## The following objects are masked from 'package:stats':  
##   
## as.dist, dist

## The following object is masked from 'package:base':  
##   
## as.matrix

## Loading required package: survey

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

## Loading required package: grid

## Loading required package: Matrix

## Loading required package: survival

##   
## Attaching package: 'survival'

## The following object is masked from 'package:caret':  
##   
## cluster

##   
## Attaching package: 'survey'

## The following object is masked from 'package:graphics':  
##   
## dotchart

## Loading required package: lpSolve

library(cluster)

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

library(ggplot2)  
library(cowplot)

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

Energy\_Data[Energy\_Data==""] = NA # Replacing spaces with NA values  
  
Filtered\_Energy\_Data = Energy\_Data[,(colMeans(is.na(Energy\_Data))\*100)<50] #Removing the columns with more than 50% NA values  
  
Partitioned\_EnergyData\_Index = createDataPartition(Filtered\_Energy\_Data$rowid,p=0.02,list = FALSE) #Partitioning 2 percent of the data  
Partitioned\_EnergyData = Filtered\_Energy\_Data[Partitioned\_EnergyData\_Index,]  
  
colMeans(is.na(Partitioned\_EnergyData))\*100 # Checking the NA values percentage

## rowid plant\_id\_eia   
## 0.000000000 0.000000000   
## report\_date contract\_type\_code   
## 0.000000000 0.041077884   
## energy\_source\_code fuel\_type\_code\_pudl   
## 0.000000000 0.000000000   
## fuel\_group\_code supplier\_name   
## 0.000000000 0.008215577   
## fuel\_received\_units fuel\_mmbtu\_per\_unit   
## 0.000000000 0.000000000   
## sulfur\_content\_pct ash\_content\_pct   
## 0.000000000 0.000000000   
## mercury\_content\_ppm fuel\_cost\_per\_mmbtu   
## 47.436740059 33.043049622   
## primary\_transportation\_mode\_code natural\_gas\_transport\_code   
## 9.324679593 44.199802826   
## data\_maturity   
## 0.000000000

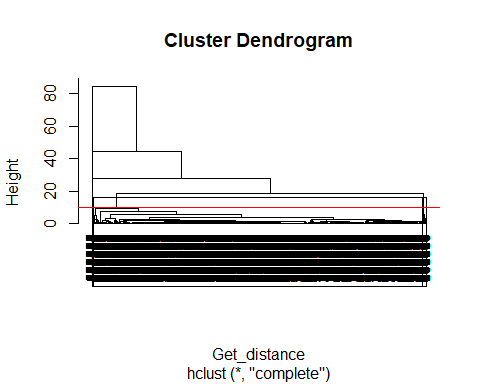
Partitioned\_EnergyData$report\_date <- as.Date(Partitioned\_EnergyData$report\_date) #Converting report\_date to date type  
  
Partitioned\_EnergyData$report\_date <- as.numeric(format(Partitioned\_EnergyData$report\_date, "%Y")) #Formatting the date to Years  
  
Partitioned\_Final\_EnergyData=Partitioned\_EnergyData[,-c(1,6,8,17)] #Removing specific column  
#row Id-Identifier, fuel\_group\_code\_pudl- Similar values as fuel\_group\_code,Data Maturity- Single value

Data Imputation

#Converting the character to factor type  
  
  
Partitioned\_Final\_EnergyData$report\_date = as.factor(Partitioned\_Final\_EnergyData$report\_date)  
  
Partitioned\_Final\_EnergyData$contract\_type\_code = as.factor(Partitioned\_Final\_EnergyData$contract\_type\_code)  
  
Partitioned\_Final\_EnergyData$energy\_source\_code = as.factor(Partitioned\_Final\_EnergyData$energy\_source\_code)  
  
Partitioned\_Final\_EnergyData$fuel\_group\_code = as.factor(Partitioned\_Final\_EnergyData$fuel\_group\_code)  
  
Partitioned\_Final\_EnergyData$primary\_transportation\_mode\_code = as.factor(Partitioned\_Final\_EnergyData$primary\_transportation\_mode\_code)  
  
Partitioned\_Final\_EnergyData$natural\_gas\_transport\_code = as.factor(Partitioned\_Final\_EnergyData$natural\_gas\_transport\_code)  
  
ImputedData = missForest(Partitioned\_Final\_EnergyData) # Imputed NA values with missForest package  
  
Imputed\_EnergyData = ImputedData$ximp # Copying the Imputed Data  
  
Imputed\_EnergyData$supplier\_name = Partitioned\_EnergyData$supplier\_name # Adding the supplier name for the analysis

Train\_label = createDataPartition(Imputed\_EnergyData$plant\_id\_eia,p=0.75,list = FALSE) # Partitioning 75% for Train and 25% for Test  
  
Train\_EnergyData = Imputed\_EnergyData[Train\_label,]  
  
Test\_EnergyData = Imputed\_EnergyData[-Train\_label,]

Cluster\_variables = Train\_EnergyData[,c(6,7,10,11)] # Considering only the numerical variables  
  
Norm\_EnergyData = scale(Cluster\_variables) # Normalizing the data  
  
Get\_distance= dist(Norm\_EnergyData,method="euclidean") # Finding distance measure from euclidean  
  
hclustering=hclust(Get\_distance,method = "complete") #Running the hierarchical clustering method  
  
plot(hclustering,cex=0.9,hang=-8); rect.hclust(hclustering,k=6,border=1.4);abline(h = 10, col = 'red')



h\_cluster = cutree(hclustering, h=10)

library(e1071)  
library(caTools)

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

library(class)  
  
Mul\_reg\_Fuel\_received = lm(fuel\_cost\_per\_mmbtu~fuel\_received\_units,data=Train\_EnergyData)  
  
Mul\_reg\_Fuel\_mmbut = lm(fuel\_cost\_per\_mmbtu~fuel\_mmbtu\_per\_unit,data=Train\_EnergyData)  
  
Mul\_reg\_sul = lm(fuel\_cost\_per\_mmbtu~sulfur\_content\_pct,data=Train\_EnergyData)  
  
Mul\_reg\_ash= lm(fuel\_cost\_per\_mmbtu~ash\_content\_pct,data=Train\_EnergyData)  
  
Mul\_reg\_mer = lm(fuel\_cost\_per\_mmbtu~mercury\_content\_ppm,data=Train\_EnergyData)  
  
Multiple\_regression=lm(fuel\_cost\_per\_mmbtu~fuel\_mmbtu\_per\_unit,data = Train\_EnergyData)  
  
summary(Multiple\_regression)

##   
## Call:  
## lm(formula = fuel\_cost\_per\_mmbtu ~ fuel\_mmbtu\_per\_unit, data = Train\_EnergyData)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -18.0 -14.0 -7.9 -0.3 15050.3   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 19.2505 2.6073 7.383 1.68e-13 \*\*\*  
## fuel\_mmbtu\_per\_unit -0.7169 0.1971 -3.637 0.000278 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 184.8 on 9128 degrees of freedom  
## Multiple R-squared: 0.001447, Adjusted R-squared: 0.001337   
## F-statistic: 13.23 on 1 and 9128 DF, p-value: 0.0002776

prediction= predict(Mul\_reg\_Fuel\_mmbut, newdata = Test\_EnergyData, type="response")  
  
cor(prediction,Test\_EnergyData$fuel\_cost\_per\_mmbtu)

## [1] 0.1270709

Here, From the lm models we see that only fuel\_mmbtu\_per\_unit is statistically significant with the fuel\_cost\_mmbtu\_per\_unit.

Train\_data\_cluster=cbind(Train\_EnergyData,Cluster=as.factor(h\_cluster))  
  
train\_scale=scale(Train\_EnergyData[,6:11])  
  
test\_scale=scale(Test\_EnergyData[,6:11])  
  
classifier\_knn <- knn(train = train\_scale,  
 test = test\_scale,  
 cl = Train\_data\_cluster$Cluster,  
 k = 1)  
  
test\_data\_cluster=cbind(Test\_EnergyData,Cluster=as.factor(classifier\_knn))  
Multiple\_regression2=lm(fuel\_cost\_per\_mmbtu~fuel\_mmbtu\_per\_unit+Cluster,data = Train\_data\_cluster)  
  
prediction2= predict(Multiple\_regression2, newdata = test\_data\_cluster, type="response")  
  
cor(prediction2, test\_data\_cluster$fuel\_cost\_per\_mmbtu)

## [1] 0.7935603

Hence, By Adding the cluster information, prediction has improved which is identified from correlation value