ML Final Project

# Loading the necessary Libraries for project

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

## Loading required package: lattice

library(missForest)

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

library(corrplot)

## 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(cluster)

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

# Reading the CSV File

# reading file  
Fuel\_Receipts\_Costs\_Data=read.csv("C:/Users/Pavan Chaitanya/Downloads/fuel\_receipts\_costs\_eia923 (1).csv")  
  
# head part of file  
head(Fuel\_Receipts\_Costs\_Data,5)

## rowid plant\_id\_eia report\_date contract\_type\_code contract\_expiration\_date  
## 1 1 3 2008-01-01 C 2008-04-01  
## 2 2 3 2008-01-01 C 2008-04-01  
## 3 3 3 2008-01-01 C   
## 4 4 7 2008-01-01 C 2015-12-01  
## 5 5 7 2008-01-01 S 2008-11-01  
## energy\_source\_code fuel\_type\_code\_pudl fuel\_group\_code mine\_id\_pudl  
## 1 BIT coal coal 0  
## 2 BIT coal coal 0  
## 3 NG gas natural\_gas NA  
## 4 BIT coal coal 1  
## 5 BIT coal coal 2  
## supplier\_name fuel\_received\_units fuel\_mmbtu\_per\_unit sulfur\_content\_pct  
## 1 interocean coal 259412 23.100 0.49  
## 2 interocean coal 52241 22.800 0.48  
## 3 bay gas pipeline 2783619 1.039 0.00  
## 4 alabama coal 25397 24.610 1.69  
## 5 d & e mining 764 24.446 0.84  
## ash\_content\_pct mercury\_content\_ppm fuel\_cost\_per\_mmbtu  
## 1 5.4 NA 2.135  
## 2 5.7 NA 2.115  
## 3 0.0 NA 8.631  
## 4 14.7 NA 2.776  
## 5 15.5 NA 3.381  
## primary\_transportation\_mode\_code secondary\_transportation\_mode\_code  
## 1 RV   
## 2 RV   
## 3 PL   
## 4 TR   
## 5 TR   
## natural\_gas\_transport\_code natural\_gas\_delivery\_contract\_type\_code  
## 1 firm   
## 2 firm   
## 3 firm   
## 4 firm   
## 5 firm   
## moisture\_content\_pct chlorine\_content\_ppm data\_maturity  
## 1 NA NA final  
## 2 NA NA final  
## 3 NA NA final  
## 4 NA NA final  
## 5 NA NA final

#Checiking NA's  
colMeans(is.na(Fuel\_Receipts\_Costs\_Data))

## rowid plant\_id\_eia   
## 0.0000000 0.0000000   
## report\_date contract\_type\_code   
## 0.0000000 0.0000000   
## contract\_expiration\_date energy\_source\_code   
## 0.0000000 0.0000000   
## fuel\_type\_code\_pudl fuel\_group\_code   
## 0.0000000 0.0000000   
## mine\_id\_pudl supplier\_name   
## 0.6440512 0.0000000   
## fuel\_received\_units fuel\_mmbtu\_per\_unit   
## 0.0000000 0.0000000   
## sulfur\_content\_pct ash\_content\_pct   
## 0.0000000 0.0000000   
## mercury\_content\_ppm fuel\_cost\_per\_mmbtu   
## 0.4756797 0.3290363   
## primary\_transportation\_mode\_code secondary\_transportation\_mode\_code   
## 0.0000000 0.0000000   
## natural\_gas\_transport\_code natural\_gas\_delivery\_contract\_type\_code   
## 0.0000000 0.0000000   
## moisture\_content\_pct chlorine\_content\_ppm   
## 0.8488641 0.8488641   
## data\_maturity   
## 0.0000000

# Data Cleaning and Removing the Unnecessary Colums that are present in dataset

# Randonmly Assigning the seed value  
set.seed(2875)  
  
#checking the NA Values  
Fuel\_Receipts\_Costs\_Data[Fuel\_Receipts\_Costs\_Data==""] = NA  
  
#Converting the mean values to the percentage  
Filtering\_NA = Fuel\_Receipts\_Costs\_Data[,(colMeans(is.na(Fuel\_Receipts\_Costs\_Data))\*100)<50]  
  
#Sampling the 2 % of the data   
Creating\_Two\_data\_Partition = createDataPartition(Filtering\_NA$plant\_id\_eia,p=0.02,list = FALSE)  
Creating\_Two\_data\_Partition1 = Filtering\_NA[Creating\_Two\_data\_Partition,]  
# Printing the 2% data  
head(Creating\_Two\_data\_Partition1,10)

## rowid plant\_id\_eia report\_date contract\_type\_code energy\_source\_code  
## 120 120 130 2008-01-01 C BIT  
## 125 125 136 2008-01-01 C BIT  
## 142 142 160 2008-01-01 C SUB  
## 219 219 525 2008-01-01 C BIT  
## 275 275 535 2008-01-01 S NG  
## 309 309 564 2008-01-01 C BIT  
## 351 351 619 2008-01-01 C NG  
## 389 389 666 2008-01-01 S NG  
## 486 486 876 2008-01-01 NC SUB  
## 619 619 1077 2008-01-01 C PC  
## fuel\_type\_code\_pudl fuel\_group\_code supplier\_name  
## 120 coal coal arch  
## 125 coal coal alliance coal  
## 142 coal coal rio tinto  
## 219 coal coal peabody coal  
## 275 gas natural\_gas suncor energy  
## 309 coal coal icg  
## 351 gas natural\_gas florida gas transmission  
## 389 gas natural\_gas florida gas transmission  
## 486 coal coal rio tinto  
## 619 coal petroleum\_coke petcoke  
## fuel\_received\_units fuel\_mmbtu\_per\_unit sulfur\_content\_pct ash\_content\_pct  
## 120 21769 24.700 0.79 10.50  
## 125 56274 23.376 2.88 7.10  
## 142 13105 20.764 0.40 5.00  
## 219 115560 22.512 0.50 10.20  
## 275 7 1.000 0.00 0.00  
## 309 11096 22.190 1.18 11.10  
## 351 732643 1.026 0.00 0.00  
## 389 48274 1.054 0.00 0.00  
## 486 31664 17.530 0.29 6.20  
## 619 3380 28.000 5.80 0.54  
## mercury\_content\_ppm fuel\_cost\_per\_mmbtu primary\_transportation\_mode\_code  
## 120 NA 2.300 RR  
## 125 NA 2.201 RR  
## 142 NA 1.661 RR  
## 219 NA 1.431 RR  
## 275 NA 9.703 <NA>  
## 309 NA 2.761 RR  
## 351 NA 9.386 <NA>  
## 389 NA 10.715 <NA>  
## 486 NA NA RR  
## 619 NA 1.944 TR  
## natural\_gas\_transport\_code data\_maturity  
## 120 <NA> final  
## 125 <NA> final  
## 142 <NA> final  
## 219 <NA> final  
## 275 firm final  
## 309 <NA> final  
## 351 firm final  
## 389 interruptible final  
## 486 <NA> final  
## 619 <NA> final

colMeans(is.na(Creating\_Two\_data\_Partition1))\*100

## rowid plant\_id\_eia   
## 0.00000000 0.00000000   
## report\_date contract\_type\_code   
## 0.00000000 0.04107451   
## energy\_source\_code fuel\_type\_code\_pudl   
## 0.00000000 0.00000000   
## fuel\_group\_code supplier\_name   
## 0.00000000 0.00000000   
## fuel\_received\_units fuel\_mmbtu\_per\_unit   
## 0.00000000 0.00000000   
## sulfur\_content\_pct ash\_content\_pct   
## 0.00000000 0.00000000   
## mercury\_content\_ppm fuel\_cost\_per\_mmbtu   
## 47.96681180 32.81853282   
## primary\_transportation\_mode\_code natural\_gas\_transport\_code   
## 9.79216298 43.76899696   
## data\_maturity   
## 0.00000000

#converting the date to date format  
Creating\_Two\_data\_Partition1$report\_date <- as.Date(Creating\_Two\_data\_Partition1$report\_date)  
  
Creating\_Two\_data\_Partition1$report\_date <- as.numeric(format(Creating\_Two\_data\_Partition1$report\_date, "%Y"))  
  
# removing the unnecessary Colums  
Creating\_Two\_data\_Partition1=Creating\_Two\_data\_Partition1[,-c(6,8,17)]  
  
# Printing the data data frame after removing unnecessary columns  
head(Creating\_Two\_data\_Partition1,10)

## rowid plant\_id\_eia report\_date contract\_type\_code energy\_source\_code  
## 120 120 130 2008 C BIT  
## 125 125 136 2008 C BIT  
## 142 142 160 2008 C SUB  
## 219 219 525 2008 C BIT  
## 275 275 535 2008 S NG  
## 309 309 564 2008 C BIT  
## 351 351 619 2008 C NG  
## 389 389 666 2008 S NG  
## 486 486 876 2008 NC SUB  
## 619 619 1077 2008 C PC  
## fuel\_group\_code fuel\_received\_units fuel\_mmbtu\_per\_unit sulfur\_content\_pct  
## 120 coal 21769 24.700 0.79  
## 125 coal 56274 23.376 2.88  
## 142 coal 13105 20.764 0.40  
## 219 coal 115560 22.512 0.50  
## 275 natural\_gas 7 1.000 0.00  
## 309 coal 11096 22.190 1.18  
## 351 natural\_gas 732643 1.026 0.00  
## 389 natural\_gas 48274 1.054 0.00  
## 486 coal 31664 17.530 0.29  
## 619 petroleum\_coke 3380 28.000 5.80  
## ash\_content\_pct mercury\_content\_ppm fuel\_cost\_per\_mmbtu  
## 120 10.50 NA 2.300  
## 125 7.10 NA 2.201  
## 142 5.00 NA 1.661  
## 219 10.20 NA 1.431  
## 275 0.00 NA 9.703  
## 309 11.10 NA 2.761  
## 351 0.00 NA 9.386  
## 389 0.00 NA 10.715  
## 486 6.20 NA NA  
## 619 0.54 NA 1.944  
## primary\_transportation\_mode\_code natural\_gas\_transport\_code  
## 120 RR <NA>  
## 125 RR <NA>  
## 142 RR <NA>  
## 219 RR <NA>  
## 275 <NA> firm  
## 309 RR <NA>  
## 351 <NA> firm  
## 389 <NA> interruptible  
## 486 RR <NA>  
## 619 TR <NA>

# Data Imputation

# Converting the variables of char to factor type for data impuataion   
Creating\_Two\_data\_Partition1$report\_date = as.factor(Creating\_Two\_data\_Partition1$report\_date)  
  
Creating\_Two\_data\_Partition1$contract\_type\_code = as.factor(Creating\_Two\_data\_Partition1$contract\_type\_code)  
  
Creating\_Two\_data\_Partition1$energy\_source\_code = as.factor(Creating\_Two\_data\_Partition1$energy\_source\_code)  
  
Creating\_Two\_data\_Partition1$fuel\_group\_code = as.factor(Creating\_Two\_data\_Partition1$fuel\_group\_code)  
  
Creating\_Two\_data\_Partition1$primary\_transportation\_mode\_code = as.factor(Creating\_Two\_data\_Partition1$primary\_transportation\_mode\_code)  
  
Creating\_Two\_data\_Partition1$natural\_gas\_transport\_code = as.factor(Creating\_Two\_data\_Partition1$natural\_gas\_transport\_code)  
  
# Computing the Data Imputation  
Genertated\_Data = missForest(Creating\_Two\_data\_Partition1)  
  
#Taking only the ximp data frame   
Imputed = Genertated\_Data$ximp  
  
#Printing the data frame after computation of the missing values  
head(Imputed,10)

## rowid plant\_id\_eia report\_date contract\_type\_code energy\_source\_code  
## 120 120 130 2008 C BIT  
## 125 125 136 2008 C BIT  
## 142 142 160 2008 C SUB  
## 219 219 525 2008 C BIT  
## 275 275 535 2008 S NG  
## 309 309 564 2008 C BIT  
## 351 351 619 2008 C NG  
## 389 389 666 2008 S NG  
## 486 486 876 2008 NC SUB  
## 619 619 1077 2008 C PC  
## fuel\_group\_code fuel\_received\_units fuel\_mmbtu\_per\_unit sulfur\_content\_pct  
## 120 coal 21769 24.700 0.79  
## 125 coal 56274 23.376 2.88  
## 142 coal 13105 20.764 0.40  
## 219 coal 115560 22.512 0.50  
## 275 natural\_gas 7 1.000 0.00  
## 309 coal 11096 22.190 1.18  
## 351 natural\_gas 732643 1.026 0.00  
## 389 natural\_gas 48274 1.054 0.00  
## 486 coal 31664 17.530 0.29  
## 619 petroleum\_coke 3380 28.000 5.80  
## ash\_content\_pct mercury\_content\_ppm fuel\_cost\_per\_mmbtu  
## 120 10.50 1.655000e-02 2.300000  
## 125 7.10 1.318733e-02 2.201000  
## 142 5.00 2.240737e-02 1.661000  
## 219 10.20 1.781000e-02 1.431000  
## 275 0.00 -2.234844e-16 9.703000  
## 309 11.10 1.932000e-02 2.761000  
## 351 0.00 -2.314121e-16 9.386000  
## 389 0.00 -2.581269e-16 10.715000  
## 486 6.20 1.446737e-02 1.634491  
## 619 0.54 1.980333e-02 1.944000  
## primary\_transportation\_mode\_code natural\_gas\_transport\_code  
## 120 RR firm  
## 125 RR firm  
## 142 RR firm  
## 219 RR firm  
## 275 PL firm  
## 309 RR firm  
## 351 PL firm  
## 389 PL interruptible  
## 486 RR firm  
## 619 TR firm

# Partitioning the 2 % data into 75 % training data.

Data\_Partition = createDataPartition(Imputed$plant\_id\_eia,p=0.75,list = FALSE)  
  
Data\_Partition\_Trained = Imputed[Data\_Partition,]  
  
Data\_Partition\_Tested = Imputed[-Data\_Partition,]

# As data has Outliers we are making sure that the outlier are removed.

# For the fuel received units performing the quartile ranges and IQR   
Quartiled\_data = quantile(Data\_Partition\_Trained$fuel\_received\_units, probs=c(.25, .75), na.rm = FALSE)  
Data\_Partition\_Quartiled = IQR(Data\_Partition\_Trained$fuel\_received\_units)  
  
  
Fuelunits\_Lower = Quartiled\_data[1] - 1.5\*Data\_Partition\_Quartiled  
Fuelunits\_Upper = Quartiled\_data[2] + 1.5\*Data\_Partition\_Quartiled   
   
Data\_With\_No\_Outliers = subset(Data\_Partition\_Trained, Data\_Partition\_Trained$fuel\_received\_units > Fuelunits\_Lower & Data\_Partition\_Trained$fuel\_received\_units < Fuelunits\_Upper)  
  
# For the fuel cost per mmbtu performing the quartile ranges and IQR  
Range\_of\_Fuel = quantile(Data\_With\_No\_Outliers$fuel\_cost\_per\_mmbtu, probs=c(.25, .75), na.rm = FALSE)  
Fuelcost\_IQR <- IQR(Data\_With\_No\_Outliers$fuel\_cost\_per\_mmbtu)  
   
Fuelcost\_Lower = Range\_of\_Fuel[1] - 1.5\*Fuelcost\_IQR  
Fuelcost\_Upper = Range\_of\_Fuel[2] + 1.5\*Fuelcost\_IQR   
   
No\_Outlier\_Data = subset(Data\_With\_No\_Outliers, Data\_With\_No\_Outliers$fuel\_cost\_per\_mmbtu > Fuelcost\_Lower & Data\_With\_No\_Outliers$fuel\_cost\_per\_mmbtu < Fuelcost\_Upper)

# Choosing and Normalising the selected variables

All\_Numeric\_Variables=No\_Outlier\_Data[,c(7,8,9,10,11,12)]  
head(All\_Numeric\_Variables,12)

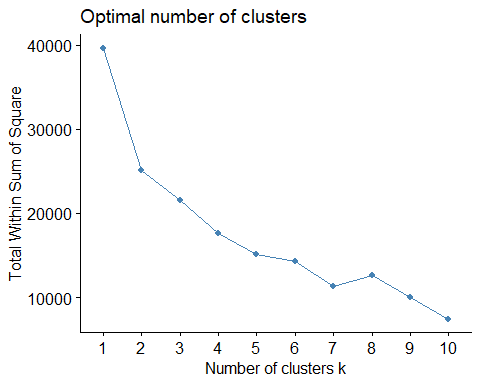
## fuel\_received\_units fuel\_mmbtu\_per\_unit sulfur\_content\_pct ash\_content\_pct  
## 120 21769 24.700 0.79 10.50  
## 125 56274 23.376 2.88 7.10  
## 219 115560 22.512 0.50 10.20  
## 309 11096 22.190 1.18 11.10  
## 389 48274 1.054 0.00 0.00  
## 486 31664 17.530 0.29 6.20  
## 619 3380 28.000 5.80 0.54  
## 685 10905 22.082 3.96 16.20  
## 709 40051 1.011 0.00 0.00  
## 737 20400 24.790 0.98 10.30  
## 747 17889 24.006 1.54 12.70  
## 796 33756 1.025 0.00 0.00  
## mercury\_content\_ppm fuel\_cost\_per\_mmbtu  
## 120 1.655000e-02 2.300000  
## 125 1.318733e-02 2.201000  
## 219 1.781000e-02 1.431000  
## 309 1.932000e-02 2.761000  
## 389 -2.581269e-16 10.715000  
## 486 1.446737e-02 1.634491  
## 619 1.980333e-02 1.944000  
## 685 1.850000e-02 1.765000  
## 709 -2.546574e-16 8.329000  
## 737 1.080000e-02 2.182000  
## 747 1.134091e-02 2.425000  
## 796 -2.361653e-16 8.633000

Scaled\_Data = scale(All\_Numeric\_Variables)  
head(Scaled\_Data,12)

## fuel\_received\_units fuel\_mmbtu\_per\_unit sulfur\_content\_pct ash\_content\_pct  
## 120 -0.343145722 1.3206695 0.1248810 0.8434456  
## 125 0.294744214 1.1942800 2.0336736 0.3548104  
## 219 1.390757626 1.1118022 -0.1399754 0.8003308  
## 309 -0.540456237 1.0810639 0.4810672 0.9296754  
## 389 0.146849141 -0.9365870 -0.5966243 -0.6655750  
## 486 -0.160218004 0.6362185 -0.3317679 0.2254658  
## 619 -0.683101034 1.6356888 4.7005035 -0.5879682  
## 685 -0.543987231 1.0707542 3.0200354 1.6626283  
## 709 -0.005168507 -0.9406918 -0.5966243 -0.6655750  
## 737 -0.368454267 1.3292610 0.2984076 0.8147024  
## 747 -0.414874833 1.2544200 0.8098545 1.1596214  
## 796 -0.121543443 -0.9393554 -0.5966243 -0.6655750  
## mercury\_content\_ppm fuel\_cost\_per\_mmbtu  
## 120 0.076866008 -0.6919802  
## 125 -0.007240504 -0.7397989  
## 219 0.108380938 -1.1117215  
## 309 0.146148830 -0.4693096  
## 389 -0.337080099 3.3726032  
## 486 0.024775636 -1.0134318  
## 619 0.158237891 -0.8639341  
## 685 0.125639114 -0.9503940  
## 709 -0.337080099 2.2201259  
## 737 -0.066952126 -0.7489762  
## 747 -0.053422989 -0.6316032  
## 796 -0.337080099 2.3669629

# K-Means Clustering

#wss  
fviz\_nbclust(Scaled\_Data, kmeans, method = "wss")



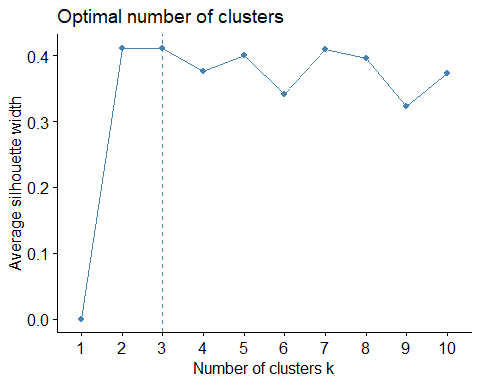
# We feel that k=2 is best.  
wss\_k2 = kmeans(Scaled\_Data, centers=2,nstart=50)  
wss\_group=wss\_k2$cluster  
wss\_k2$withinss

## [1] 8364.451 16771.092

wss\_k2$tot.withinss

## [1] 25135.54

fviz\_nbclust(Scaled\_Data, kmeans, method = "silhouette")



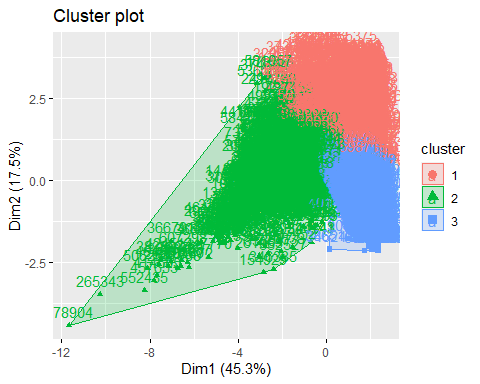
# Silhouette shows that k=3 is best.  
Sil\_k3 = kmeans(Scaled\_Data, centers=3,nstart=50)  
Silhouette\_group=Sil\_k3$cluster  
Sil\_k3$withinss

## [1] 1916.686 15046.584 4047.306

Sil\_k3$tot.withinss

## [1] 21010.58

# By comparing the both methods and by finding the withiness we have come to an idea that k=3 is the best k for our project.   
# ie Sil\_k3$tot.withinss is less that of Wss\_k2$tot.withinss  
# 2101.58 is less than 25135.54  
  
fviz\_cluster(Sil\_k3,data=Scaled\_Data)

 Interpretation

Silhouette\_group = as.data.frame(Silhouette\_group)  
Sil\_bind=cbind(All\_Numeric\_Variables,Silhouette\_group)  
Cluster\_mean= Sil\_bind %>% group\_by(Silhouette\_group) %>%  
summarise\_all("mean")  
Cluster\_mean

## # A tibble: 3 × 7  
## Silhouette\_group fuel\_received\_units fuel\_mm…¹ sulfu…² ash\_c…³ mercu…⁴ fuel\_…⁵  
## <int> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 1 161116. 5.23 0.111 1.55 3.45e-3 3.70  
## 2 2 29500. 21.6 1.42 9.86 2.90e-2 2.64  
## 3 3 18050. 1.18 0.00413 0.00863 2.13e-5 4.89  
## # … with abbreviated variable names ¹​fuel\_mmbtu\_per\_unit, ²​sulfur\_content\_pct,  
## # ³​ash\_content\_pct, ⁴​mercury\_content\_ppm, ⁵​fuel\_cost\_per\_mmbtu

#   
# As Sulfer content,ash content,mercury content are less than 0.001 m they can be neglected for intrepretation.

# Cluster 1  
#   
# The power Plants present in this cluster receives fuel of 161115.82 which is high than all the 3 clsuters.  
# Their heat content in the fuel is 5.231477 which is very good wrt to the fuel recieves compared to other 2 clsuters.  
# The fuel cost per mmbtu is also very good(3.704139) wrt to fuel recieved and the heat content.  
# This Cluster is the preferred one to recommend for the Us Government beacuse by looking all the factors like (fuel recieved,heat content,fuel cost per mmbtu).

# Cluster 2  
#   
# The power Plants present in this cluster receives fuel of 29500.21 which is slightly above the Cluster 3 but not cluster 1.  
# Their heat content in the fuel is very very high of 21.607668 comapared to all the 3 clsuters.  
# The fuel cost per mmbtu is lower(2.635552) than all the 3 clusters formed.  
# This cluster is also not a preferred one to recommend for us Government because of fuel mmbtu per unit.

# Cluster 3  
#   
# The power plants present in this cluster recieves fuel of 18049.93 which is low compared to other plants.  
# As they are receiving low fuel their heat content in fuel(fuel\_mmbtu) is also low (1.183889).  
# The fuel cost per mmbtu is higher (4.889421) than all the 3 clusters formed.  
# This Cluster is not a preferred one to recommend for Us Government because of fuel cost per mmbtu.

# Hierarchial Clustering for visualizing the data

# Getting distance  
distance= dist(Scaled\_Data,method="euclidean")  
# Computing method  
hclust\_ward=hclust(distance,method = "ward.D2")  
#plotting   
plot(hclust\_ward,cex=0.6,hang=-1);   
rect.hclust(hclust\_ward,k=3,border=1:4)

