Fundamendals of Machine Learning-Final Project

Abi

2022-11-26

Loading dataset

```
File.Data.csv<- read.csv("C:/Users/abinaya/OneDrive/Desktop/File.Data.csv.csv")
str(File.Data.csv)
```

```
## 'data.frame':    608565 obs. of  30 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 ...
##  $ plant_id_eia_label                    : chr  "Barry" "Barry" "Barry" "Gadsden" ...
##  $ report_date                           : chr  "1/1/2008" "1/1/2008" "1/1/2008" "1/1/2008" ...
##  $ contract_type_code                    : chr  "C" "C" "C" "C" ...
##  $ contract_type_code_label              : 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" ...
##  $ energy_source_code_label              : 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 ...
##  $ mine_id_pudl_label                    : int  0 0 NA 1 2 3 NA 4 4 1 ...
##  $ supplier_name                         : chr  "interocean coal" "interocean coal" "bay gas pipeli
##  $ fuel_received_units                   : int  259412 52241 2783619 25397 764 603 2341 8869 75442
##  $ 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" ...
##  $ primary_transportation_mode_code_label  : chr  "RV" "RV" "PL" "TR" ...
##  $ secondary_transportation_mode_code    : chr  "" "" "" "" ...
##  $ secondary_transportation_mode_code_label: 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                  : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ data_maturity                         : chr  "final" "final" "final" "final" ...
##  $ data_maturity_label                   : chr  "final" "final" "final" "final" ...
```

```
# Loading Package
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
## v ggplot2 3.3.6      v purrr   0.3.4
## v tibble  3.1.8      v dplyr   1.0.10
## v tidyr   1.2.0      v stringr 1.4.1
## v readr   2.1.2      v forcats 0.5.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```r
# Selecting Variables For Analysis
df_fuel <- File.Data.csv[,c(10,15:18,20)]

# Checking missing values
colMeans(is.na(df_fuel))
```

```
## fuel_type_code_pudl fuel_received_units fuel_mmbtu_per_unit  sulfur_content_pct
##           0.0000000           0.0000000           0.0000000           0.0000000
##     ash_content_pct fuel_cost_per_mmbtu
##           0.0000000           0.3290363
```

```r
# Imputing NA values with mean value
df_fuel$fuel_cost_per_mmbtu [is.na(df_fuel$fuel_cost_per_mmbtu)] <- mean(df_fuel$fuel_cost_per_mmbtu ,na
head(df_fuel)
```

```
##   fuel_type_code_pudl fuel_received_units fuel_mmbtu_per_unit
## 1                coal              259412              23.100
## 2                coal               52241              22.800
## 3                 gas             2783619               1.039
## 4                coal               25397              24.610
## 5                coal                 764              24.446
## 6                coal                 603              24.577
##   sulfur_content_pct ash_content_pct fuel_cost_per_mmbtu
## 1               0.49             5.4               2.135
## 2               0.48             5.7               2.115
## 3               0.00             0.0               8.631
## 4               1.69            14.7               2.776
## 5               0.84            15.5               3.381
## 6               1.54            14.6               2.199
```

```r
library('caret')
```

```
## Loading required package: lattice

##
## Attaching package: 'caret'

## The following object is masked from 'package:purrr':
##
##     lift
```

```r
set.seed(8439)

# Sampling the data 2%
df <- df_fuel%>%sample_frac(0.02)

# Partitioning the data
Train_index <- createDataPartition(df$fuel_received_unit, p = 0.75, list = FALSE)
train.df = df[Train_index,]
test.df = df[-Train_index,]

# Normalization
subset_data<-train.df[,-c(1)]
Normal_Data <- preProcess(subset_data,method = "range")
df_Norm <- predict(Normal_Data,subset_data)
summary(df_Norm)
```

```
##  fuel_received_units fuel_mmbtu_per_unit sulfur_content_pct ash_content_pct
##  Min.   :0.0000000   Min.   :0.00000     Min.   :0.00000    Min.   :0.00000
##  1st Qu.:0.0002925   1st Qu.:0.03389     1st Qu.:0.00000    1st Qu.:0.00000
##  Median :0.0016523   Median :0.03507     Median :0.00000    Median :0.00000
##  Mean   :0.0186840   Mean   :0.29706     Mean   :0.06424    Mean   :0.04971
##  3rd Qu.:0.0079223   3rd Qu.:0.60114     3rd Qu.:0.05792    3rd Qu.:0.08225
##  Max.   :1.0000000   Max.   :1.00000     Max.   :1.00000    Max.   :1.00000
##  fuel_cost_per_mmbtu
##  Min.   :0.0000000
##  1st Qu.:0.0001133
##  Median :0.0002056
##  Mean   :0.0005223
##  3rd Qu.:0.0006403
##  Max.   :1.0000000
```

```r
colMeans(is.na(df_Norm))
```

```
## fuel_received_units fuel_mmbtu_per_unit  sulfur_content_pct     ash_content_pct
##                   0                   0                   0                   0
## fuel_cost_per_mmbtu
##                   0
```

Loading package

```r
library("factoextra")
```

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

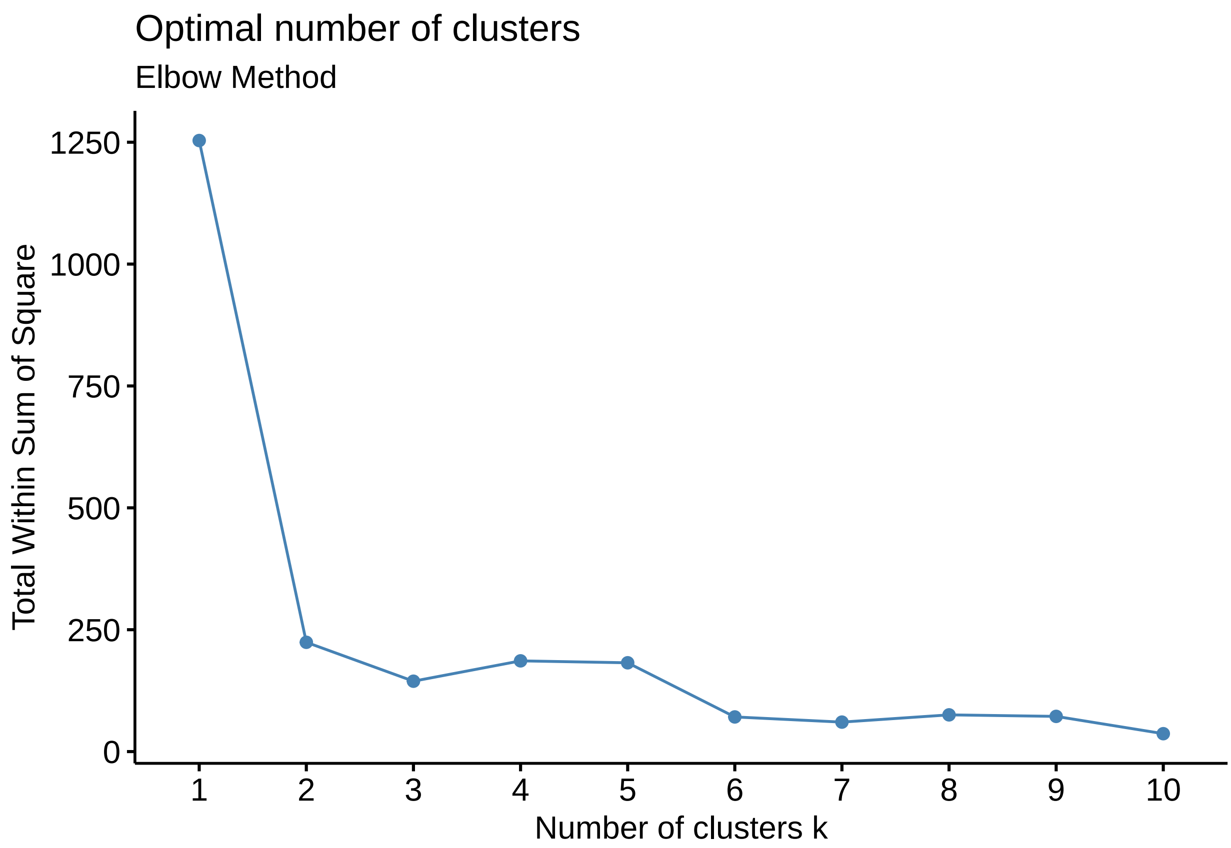
```r
library("cluster")
library("ggplot2")
library("gridExtra")
```

```
##
## Attaching package: 'gridExtra'
```

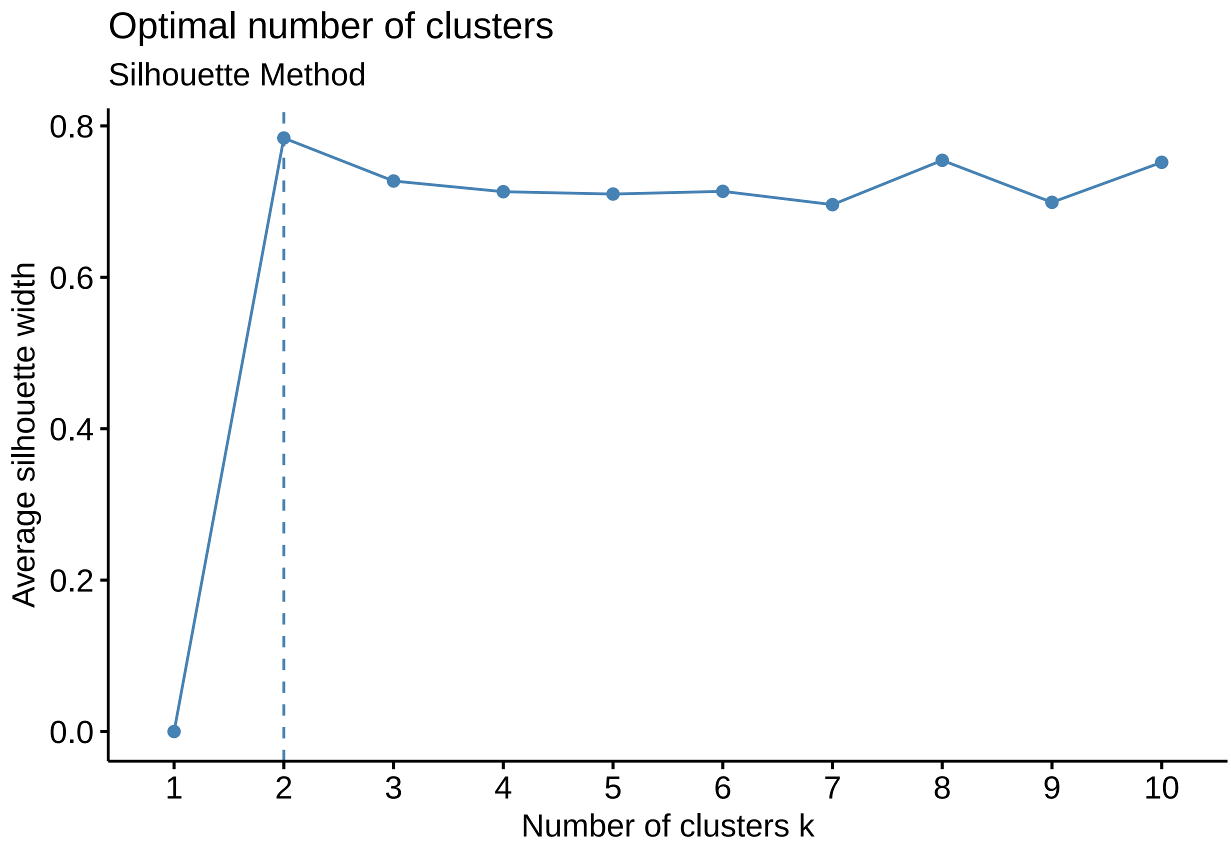
```
## The following object is masked from 'package:dplyr':
##
##     combine
```

K means clustering # Estimating the number of clusters

```
fviz_nbclust(df_Norm, kmeans, method = "wss")+ labs(subtitle = "Elbow Method")
```




```
fviz_nbclust(df_Norm, kmeans, method = "silhouette") + labs(subtitle = "Silhouette Method")
```



In Wss method choice of choosing K value is ambiguous. Therefore, I choose silhouette method with k=2.

Computing K-means clustering for centers k= 2,Silhouette:

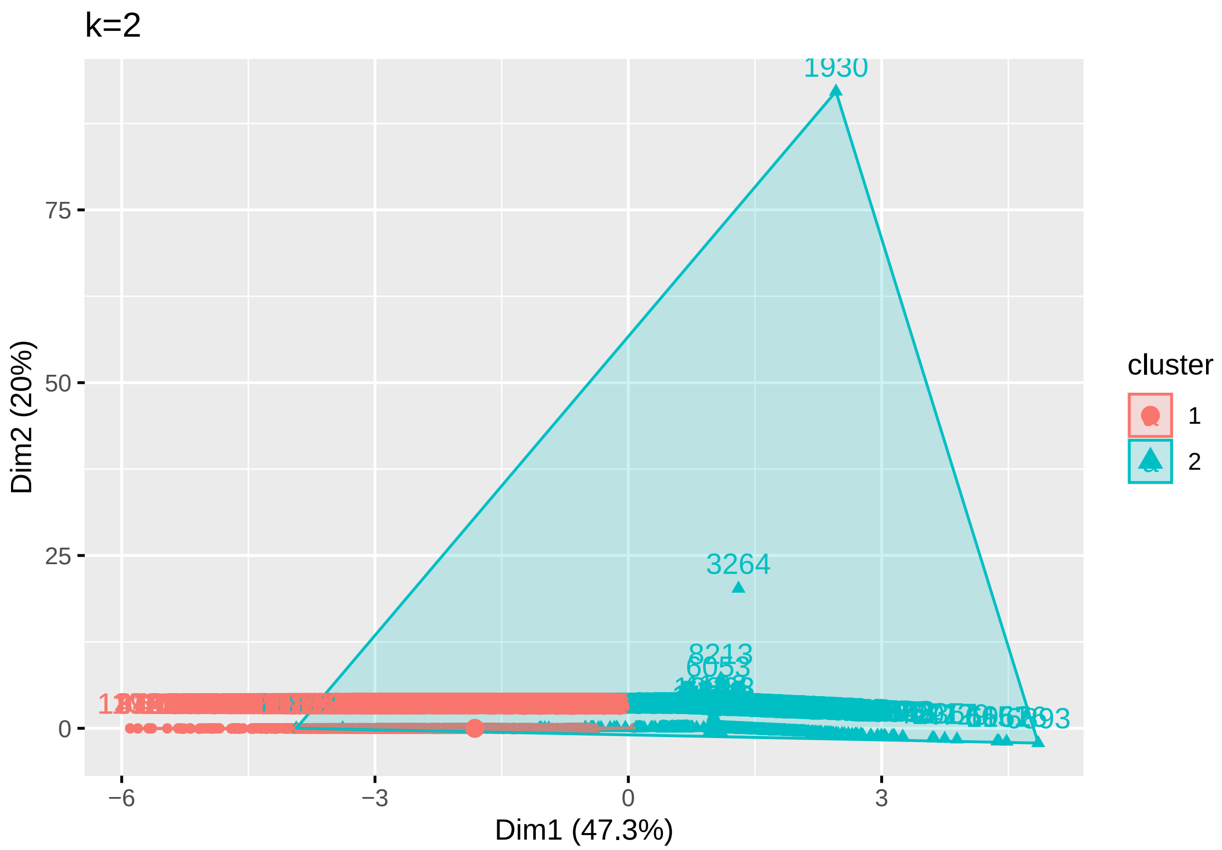
```
# k= 2
set.seed(345)
k2 <- kmeans(df_Norm, centers = 2, nstart = 25)
# The cluster centres
k2$centers
```

```
##   fuel_received_units fuel_mmbtu_per_unit sulfur_content_pct ash_content_pct
## 1         0.003608352          0.72095076        0.172326352    0.1368630898
## 2         0.027215850          0.05716859        0.003063332    0.0003811048
##   fuel_cost_per_mmbtu
## 1        0.0002534869
## 2        0.0006745075
```

Interpretation: K-means clustering with 2 clusters of sizes 3300, 5831 compactness: 82.1 %

Cluster Plot

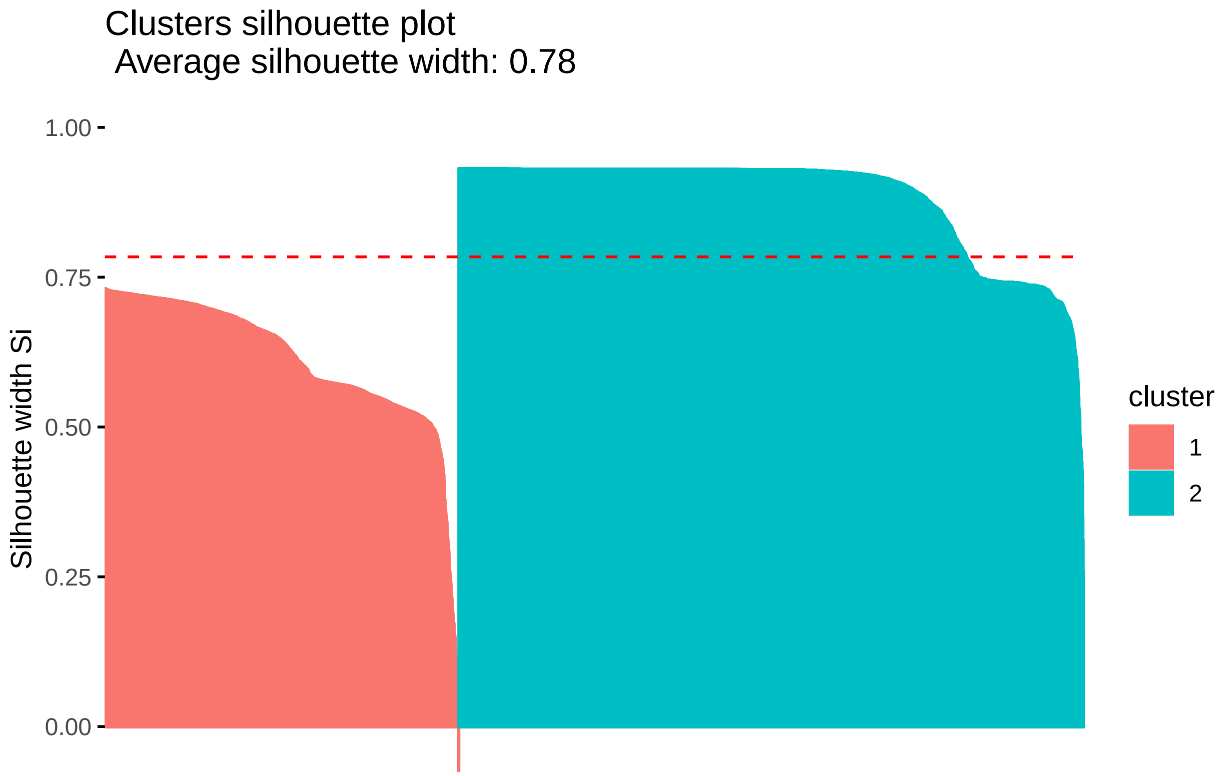
```
fviz_cluster(k2, data = df_Norm)+ggtitle("k=2")
```



Sillohuette Average

```
sil <- silhouette(k2$cluster, dist(df_Norm))
fviz_silhouette(sil)
```

```
##   cluster size ave.sil.width
## 1       1 3300          0.61
## 2       2 5831          0.88
```



Si: 0.78, since si>0, the observation is well clustered. The range of the Silhouette value is between +1 and -1. A high value is desirable and indicates that the point is placed in the correct cluster.

Final cluster Analysis

```
clr_sil <- k2$cluster
# Binding cluster with train data
f_clr <- cbind(train.df,clr_sil)
f_clr$cluster <- as.factor(f_clr$clr_sil)
head(f_clr)
```

```
##    fuel_type_code_pudl fuel_received_units fuel_mmbtu_per_unit
## 1                 coal                5000              17.790
## 4                  gas                6963               1.005
## 6                  oil                2643               5.825
## 8                  gas              373845               1.030
## 10                 gas               20265               1.029
## 11                coal               26308              23.776
##    sulfur_content_pct ash_content_pct fuel_cost_per_mmbtu clr_sil cluster
## 1                0.40             6.2             2.09200       1       1
## 4                0.00             0.0            14.18427       2       2
## 6                0.00             0.0            14.18427       2       2
## 8                0.00             0.0            14.18427       2       2
```

```
## 10                 0.00              0.0              4.84800       2       2
## 11                 1.97             15.6              4.59000       1       1
```

Aggregating

```
d<-f_clr%>%group_by(clr_sil)%>%
  summarize(
    fuel_received_units=median(fuel_received_units),
    fuel_mmbtu_per_unit=median(fuel_mmbtu_per_unit),
            fuel_cost_per_mmbtu=median(fuel_cost_per_mmbtu),
            sulfur_content=median(sulfur_content_pct),
    ash_content=median(ash_content_pct))
d
```

```
## # A tibble: 2 x 6
##   clr_sil fuel_received_units fuel_mmbtu_per_unit fuel_cost_pe~1 sulfu~2 ash_c~3
##     <int>               <dbl>               <dbl>          <dbl>   <dbl>   <dbl>
## 1       1              22100.               22.7            2.73    0.85     8.3
## 2       2              21348                 1.03           7.39    0        0
## # ... with abbreviated variable names 1: fuel_cost_per_mmbtu,
## #   2: sulfur_content, 3: ash_content
```

Plotting number of cluster

```
ggplot(f_clr) +aes(x = clr_sil, fill = fuel_type_code_pudl) +
geom_bar() + scale_fill_brewer(palette = "Accent", direction = 1) +
labs(x = "Number of Clusters", title = "CLUSTERS") + theme_minimal() +theme(plot.title = element_text(s
```

Multiple-linear regression to determine the best set of variables to predict fuel_cost_per_mmbtu

```
df_reg <- test.df
dim(df_reg)  # dimension/shape of test dataset
```

```
## [1] 3040    6
```

```
df<-df_reg[,-c(1)]
df<-scale(df)
head(df)
```

```
##    fuel_received_units fuel_mmbtu_per_unit sulfur_content_pct ash_content_pct
## 2           -0.3475831          -0.2932255         -0.5090764      -0.5325361
## 3            5.1568141          -0.7814709         -0.5090764      -0.5325361
## 5           -0.3498992          -0.7818824         -0.5090764      -0.5325361
## 7           -0.3298699          -0.7783847         -0.5090764      -0.5325361
## 9           -0.1993458           1.4719060          2.6314905       0.6946738
## 14           3.4167108          -0.7768415         -0.5090764      -0.5325361
##    fuel_cost_per_mmbtu
## 2            1.0805156
## 3           -0.2390159
```

```
## 5           -0.2914214
## 7           -0.3999078
## 9           -0.4472089
## 14          -0.3775844
```

```
Y <-test.df$fuel_cost_per_mmbtu

X1<-test.df$fuel_received_units
X2<- test.df$fuel_mmbtu_per_unit
X3<- test.df$sulfur_content_pct
X4<- test.df$ash_content_pct
```

```
model <- lm(Y ~ X1+X2+X3+X4)
summary(model)
```

```
##
## Call:
## lm(formula = Y ~ X1 + X2 + X3 + X4)
##
## Residuals:
##    Min     1Q Median     3Q    Max
##  -9.08  -5.36  -3.12   4.42 443.97
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.004e+01  3.881e-01  25.873  < 2e-16 ***
## X1          -1.017e-07  3.991e-07  -0.255    0.799
## X2          -2.369e-01  4.463e-02  -5.309 1.18e-07 ***
## X3           5.860e-01  4.029e-01   1.454    0.146
## X4           7.559e-02  5.295e-02   1.427    0.154
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 14.61 on 3035 degrees of freedom
## Multiple R-squared:  0.01271,    Adjusted R-squared:  0.01141
## F-statistic: 9.771 on 4 and 3035 DF,  p-value: 7.487e-08
```

```
anova(model)
```

```
## Analysis of Variance Table
##
## Response: Y
##             Df Sum Sq Mean Sq F value    Pr(>F)
## X1           1    252   252.1  1.1812   0.27720
## X2           1   7058  7058.2 33.0713 9.768e-09 ***
## X3           1    597   596.5  2.7950   0.09466 .
## X4           1    435   434.9  2.0377   0.15354
## Residuals 3035 647739   213.4
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Interpretation using test set: fuel_mmbtu_per_unit- Heat content of the fuel in millions of Btus per physical unit. fuel_mmbtu_per_unit is the best set of variables to predict fuel_cost_per_mmbtu, According to the mean square(relative values of sum squares). Fuel's heat content(fuel_mmbtu_per_unit) of the house explains 7058.2 units of variability of the heat produced cost(fuel_cost_per_mmbtu).

Multiple-linear regression for cluster

```
df_re <- f_clr
subset<-df_re[,-c(1)]
Normal_Data <- preProcess(subset,method = "range")
df_Norm7 <- predict(Normal_Data,subset)
summary(df_Norm7)
```

```
##  fuel_received_units fuel_mmbtu_per_unit sulfur_content_pct ash_content_pct
##  Min.   :0.0000000   Min.   :0.00000     Min.   :0.00000    Min.   :0.00000
##  1st Qu.:0.0002925   1st Qu.:0.03389     1st Qu.:0.00000    1st Qu.:0.00000
##  Median :0.0016523   Median :0.03507     Median :0.00000    Median :0.00000
##  Mean   :0.0186840   Mean   :0.29706     Mean   :0.06424    Mean   :0.04971
##  3rd Qu.:0.0079223   3rd Qu.:0.60114     3rd Qu.:0.05792    3rd Qu.:0.08225
##  Max.   :1.0000000   Max.   :1.00000     Max.   :1.00000    Max.   :1.00000
##  fuel_cost_per_mmbtu    clr_sil        cluster
##  Min.   :0.0000000   Min.   :0.0000   1:3300
##  1st Qu.:0.0001133   1st Qu.:0.0000   2:5831
##  Median :0.0002056   Median :1.0000
##  Mean   :0.0005223   Mean   :0.6386
##  3rd Qu.:0.0006403   3rd Qu.:1.0000
##  Max.   :1.0000000   Max.   :1.0000
```

```
Z <-df_Norm7$fuel_cost_per_mmbtu

X5<-df_Norm7$fuel_received_units
X6<- df_Norm7$fuel_mmbtu_per_unit
X7<- df_Norm7$sulfur_content_pct
X8<- df_Norm7$ash_content_pct
```

```
model2 <- lm(Z ~ X5+X6+X7+X8)
summary(model2)
```

```
##
## Call:
## lm(formula = Z ~ X5 + X6 + X7 + X8)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.00070 -0.00047 -0.00017  0.00000  0.99928
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.0007413  0.0001653   4.484 7.41e-06 ***
## X5          -0.0014383  0.0020759  -0.693    0.488
## X6          -0.0007416  0.0005709  -1.299    0.194
## X7           0.0003428  0.0012791   0.268    0.789
```

```
## X8           0.0001248  0.0017384   0.072    0.943
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.01078 on 9126 degrees of freedom
## Multiple R-squared:  0.0003691,  Adjusted R-squared:  -6.901e-05
## F-statistic: 0.8425 on 4 and 9126 DF,  p-value: 0.498
```

```
anova(model)
```

```
## Analysis of Variance Table
##
## Response: Y
##             Df Sum Sq Mean Sq F value    Pr(>F)
## X1           1    252   252.1  1.1812   0.27720
## X2           1   7058  7058.2 33.0713 9.768e-09 ***
## X3           1    597   596.5  2.7950   0.09466 .
## X4           1    435   434.9  2.0377   0.15354
## Residuals 3035 647739   213.4
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Interpretation using Cluster information:

fuel_mmbtu_per_unit is the best set of variables to predict fuel_cost_per_mmbtu, According to the mean square(relative values of sum squares). Fuel's heat content(fuel_mmbtu_per_unit) of the house explains 7058.2 units of variability of the heat produced cost(fuel_cost_per_mmbtu).