FML Project - Nikhil Kumar Sampath - 811222899

Setting default values to get a clean output

```
knitr::opts_chunk$set(message = FALSE)
knitr::opts_chunk$set(warning = FALSE)
```

Loading the required packages

```
library("dplyr")
library("ISLR")
library("caret")
library("class")
library("ggplot2")
library("tidyverse")
library("esquisse")
library("gmodels")
library("factoextra")
library("fpc")
library("cluster")
```

Loading the data

```
Fuel_Receipts <- read.csv("Fuel_Receipts.csv")
```

Data Cleaning & Transformation

```
#Removing Unnecessary Variables
Fuel_Data <- Fuel_Receipts[,-c(1:7,9,12:14,21:30)]

#Looking for NA Values
colMeans(is.na(Fuel_Data))
```

```
##  energy_source_code fuel_type_code_pudl     fuel_group_code fuel_received_units
##           0.0000000           0.0000000           0.0000000           0.0000000
## fuel_mmbtu_per_unit  sulfur_content_pct     ash_content_pct mercury_content_ppm
##           0.0000000           0.0000000           0.0000000           0.4756797
## fuel_cost_per_mmbtu
##           0.3290363
```

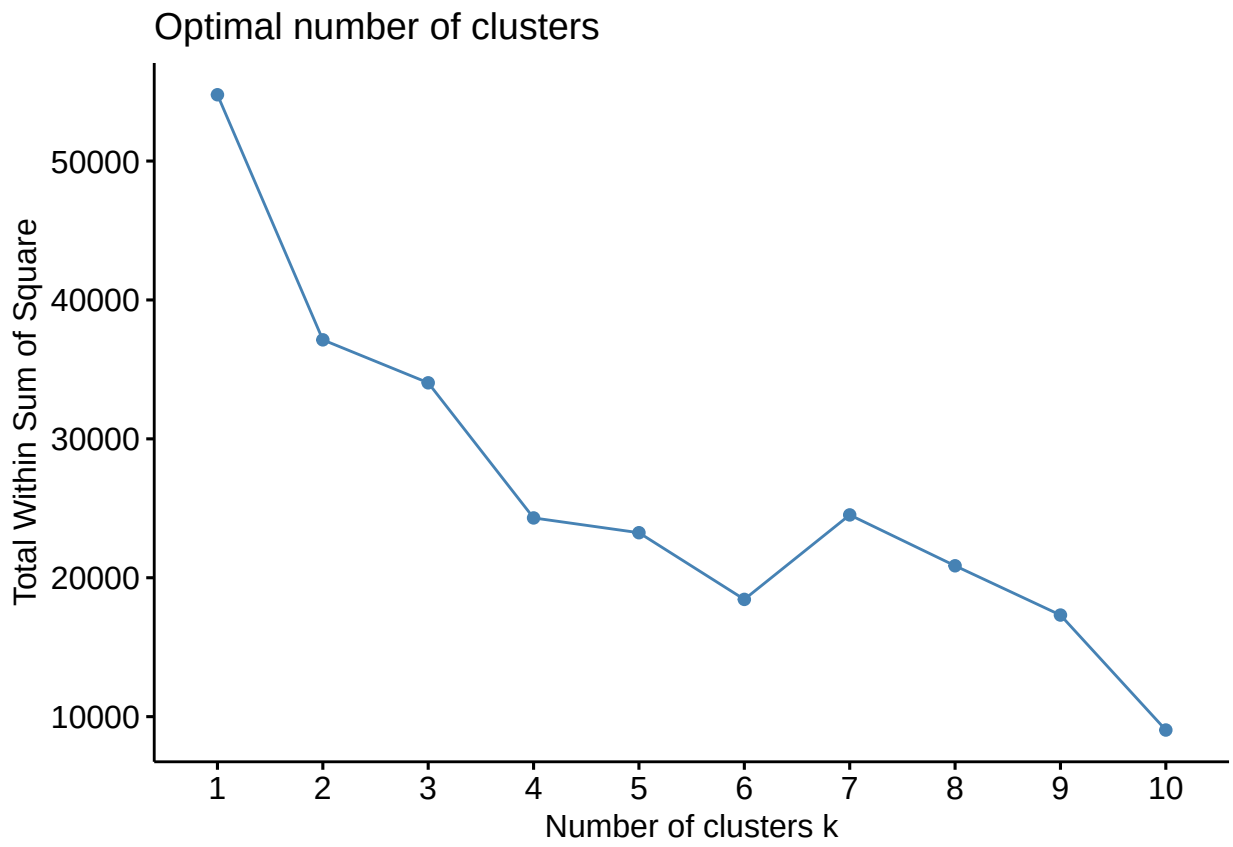
```
#Imputing the median values to fill the missing values
Fuel_Data$fuel_cost_per_mmbtu[is.na(Fuel_Data$fuel_cost_per_mmbtu)] <- median(Fuel_Data$fuel_cost_per_m
Fuel_Data$mercury_content_ppm[is.na(Fuel_Data$mercury_content_ppm)] <- median(Fuel_Data$mercury_content
```

Data Partition and Normalization

```
#Data Partition
set.seed(1234)
Data_Part_Train <- createDataPartition(Fuel_Data$fuel_cost_per_mmbtu,p=0.015,list=F)

Train_Data <- Fuel_Data[Data_Part_Train,]
Excess_Data <- Fuel_Data[-Data_Part_Train,]

Data_Part_Train_1 <- createDataPartition(Excess_Data$fuel_cost_per_mmbtu,p=0.005,list=F)
Test_Data <- Excess_Data[Data_Part_Train_1,]
Excess_Data_1 <- Excess_Data[-Data_Part_Train_1,]

#Normalization
Model_Z_Normalized <- preProcess(Train_Data[,-c(1:3)],method=c("center","scale"))

Normalized_Train <- predict(Model_Z_Normalized,Train_Data)

Normalized_Test <- predict(Model_Z_Normalized,Test_Data)
```

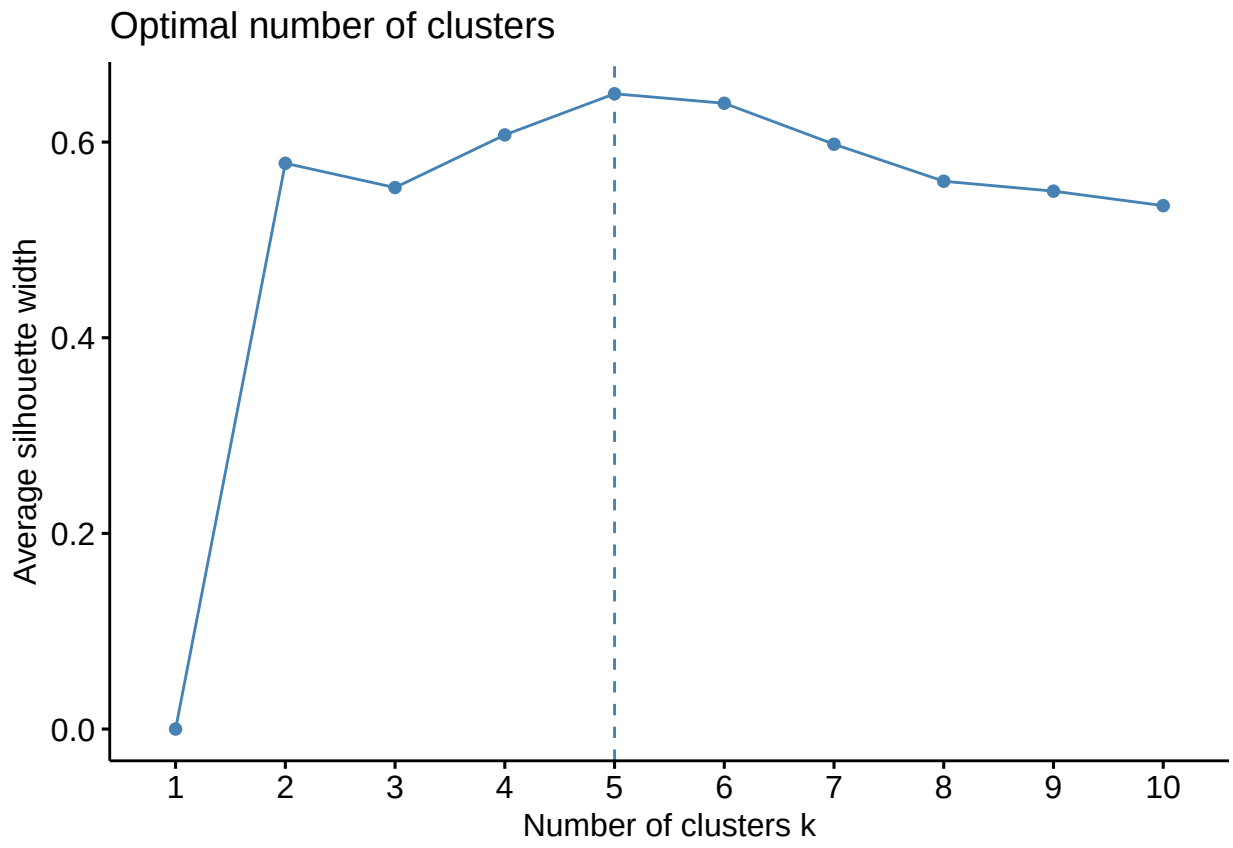
Finding the Optimal K

```
#WSS
set.seed(890)
WSS <- fviz_nbclust(Normalized_Train[,-c(1:3)],kmeans,method="wss")
WSS
```



The optimal value of k can be considered as k = 2 by using "WSS Method".

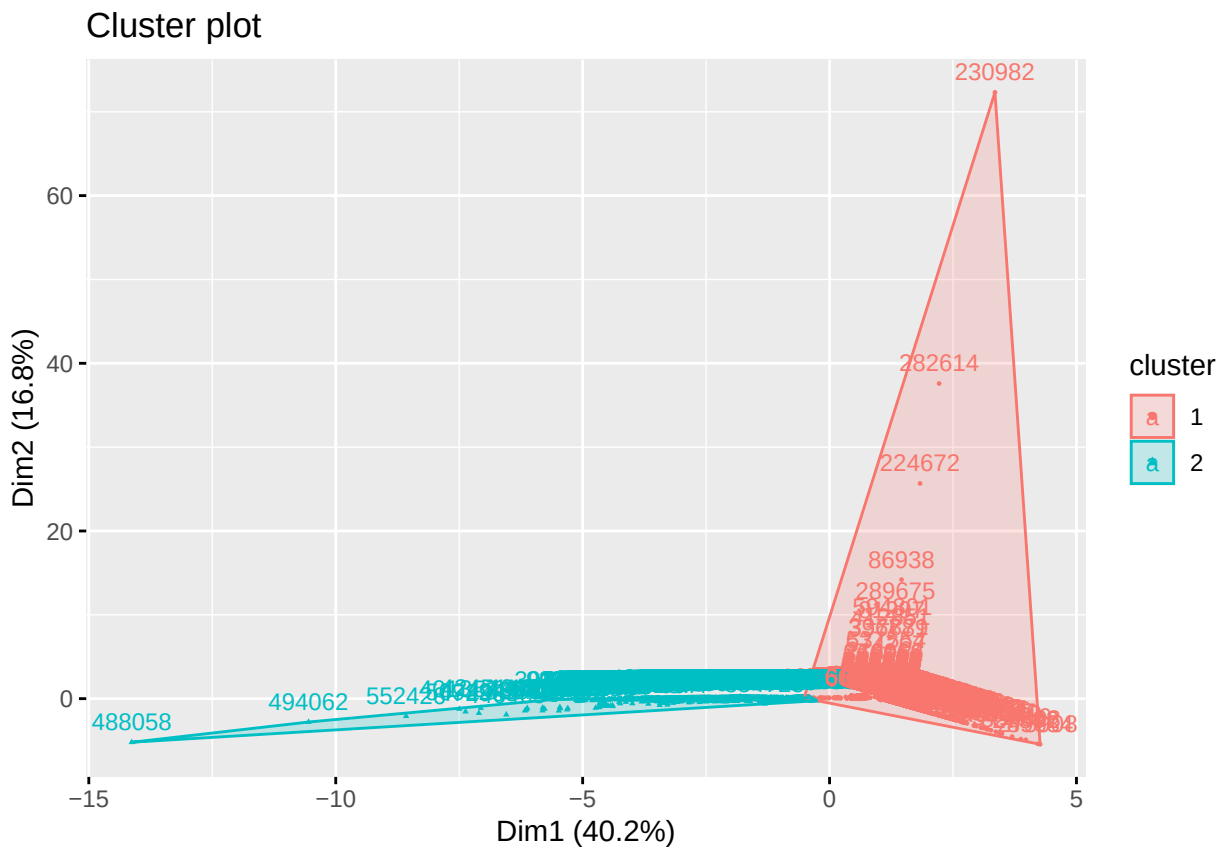
```
#Silhouette
set.seed(890)
silhouette <- fviz_nbclust(Normalized_Train[,-c(1:3)],kmeans,method="silhouette")
silhouette
```



The optimal value of k can be considered as k = 5 by using "Silhouette Method".

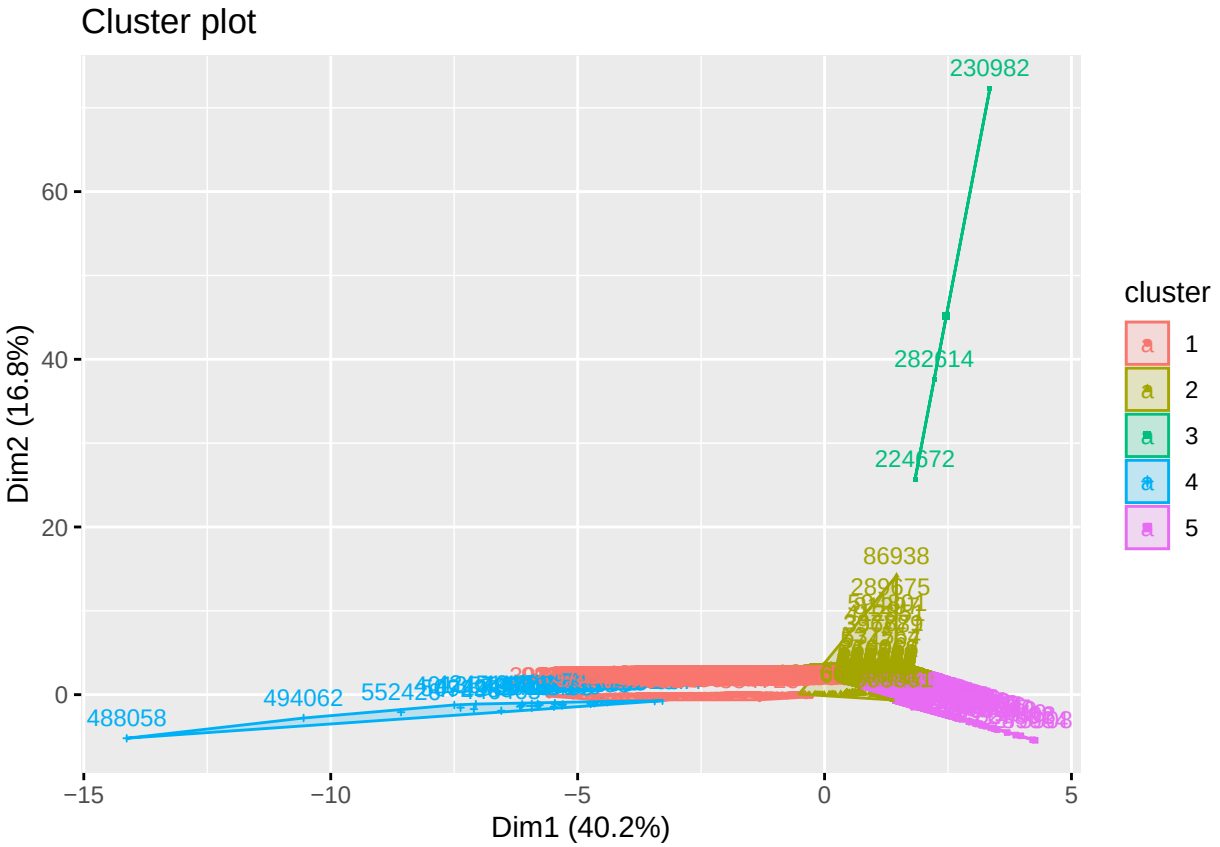
Formulation of clusters

```
set.seed(342)
#Using K Means - WSS
clus_wss_kmeans <- kmeans(Normalized_Train[,-c(1:3)],centers = 2,nstart=25)

clus_kmeans_wss <- clus_wss_kmeans$cluster

clus_wss <- cbind(Train_Data,clus_kmeans_wss)

plot.wss <- fviz_cluster(clus_wss_kmeans, data=Train_Data[,-c(1:3)],pointsize = 0.5,labelsize=10)
plot.wss
```



The cluster formation is depicted to show that "wss" method just tries to minimize the distance between the points in a cluster.

```
set.seed(342)
#Using K Means -Silhouette
clus_silhouette_kmeans <- kmeans(Normalized_Train[,-c(1:3)],centers = 5,nstart=25)

clus_kmeans_silhouette <- clus_silhouette_kmeans$cluster

clus_silhouette <- cbind(Train_Data,clus_kmeans_silhouette)

plot.silhouette <- fviz_cluster(clus_silhouette_kmeans, data=Train_Data[,-c(1:3)],pointsize = 0.5,labels
plot.silhouette
```



Whereas, "silhouette" as a method of finding optimal k gives the analyst/user a wider scope to understand the problem which will lay him/her on the set track to reach to their objective functions, here silhouette focuses on not only minimizing the distance between the points in a cluster but also it focuses on maximizing the distance between the clusters.

Thus, we can also say that by proceeding with k=5 we can ideally have a wider vision to look and also understand about the power generation in the US, this can be the main reason for proceeding with k = 5 thereby emphasizing more towards the objective statement.

Analyzing the clusters - Silhouette

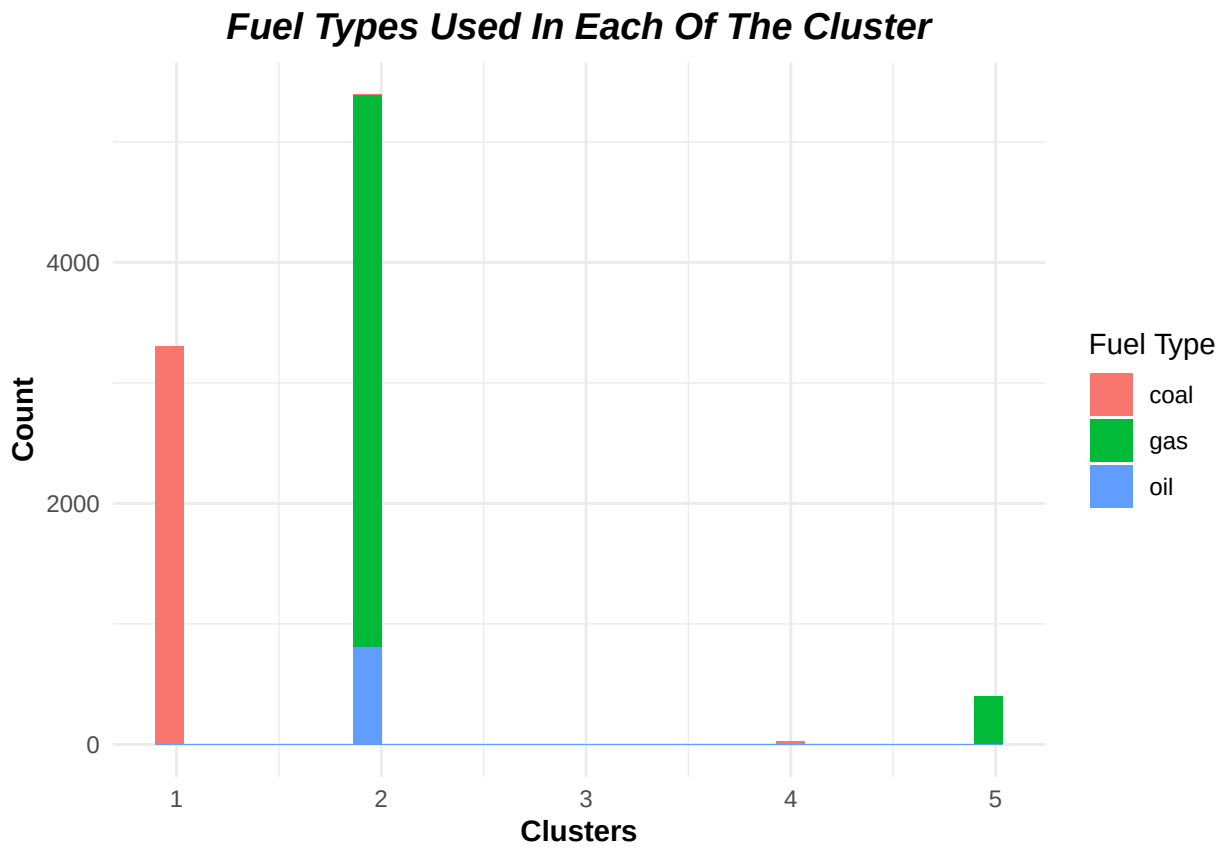
```
#i.
Fuel_Silhouette_FC <- clus_silhouette %>% select(fuel_type_code_pudl,clus_kmeans_silhouette) %>% group_
Fuel_Silhouette_FC
```

```
## # A tibble: 7 x 3
## # Groups:   fuel_type_code_pudl, clus_kmeans_silhouette [7]
##   fuel_type_code_pudl clus_kmeans_silhouette     n
##   <chr>                                <int> <int>
## 1 coal                                     1  3305
## 2 coal                                     2     4
## 3 gas                                      2  4578
## 4 oil                                      2   810
## 5 gas                                      3     3
## 6 coal                                     4    31
## 7 gas                                      5   399
```

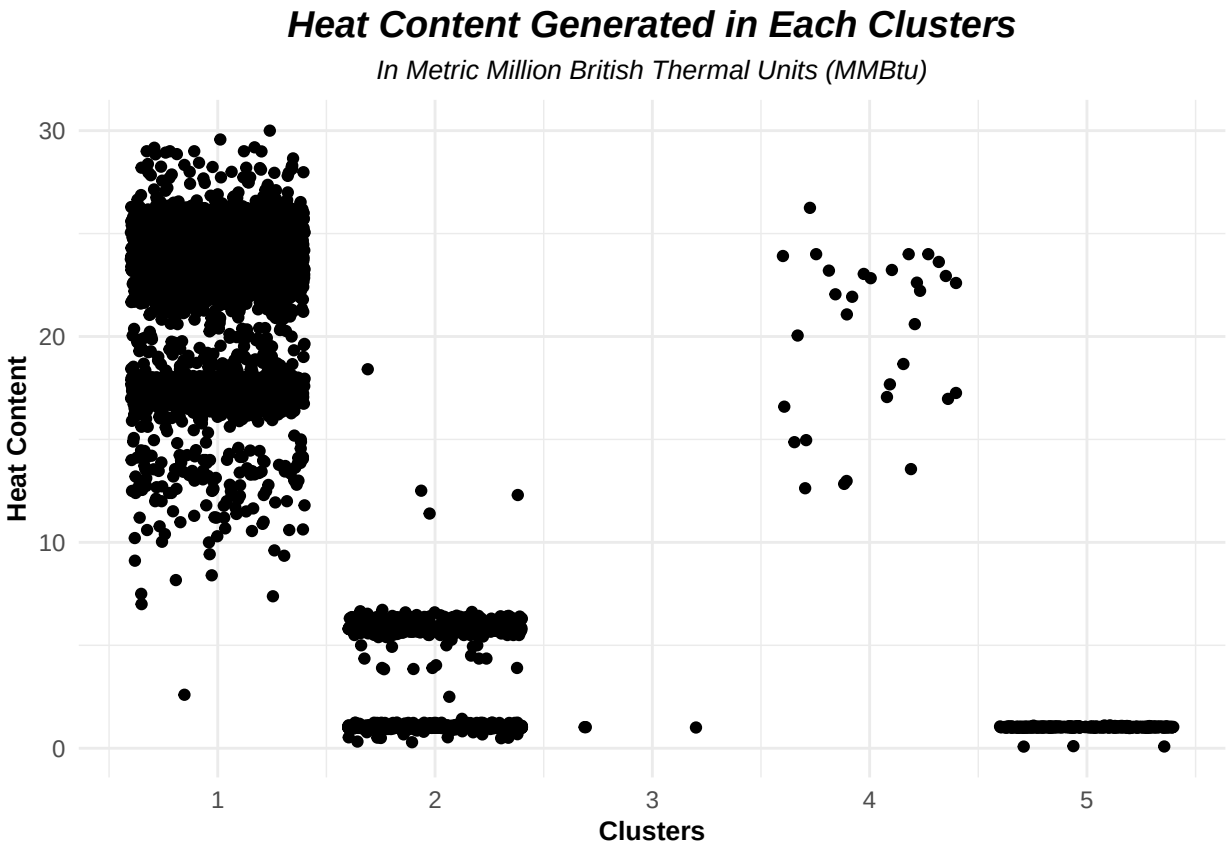
```r
#ii.
Fuel_Silhouette_Median <- clus_silhouette %>% group_by(clus_kmeans_silhouette) %>% summarise(median_cos
Fuel_Silhouette_Median
```

```
## # A tibble: 5 x 7
##   clus_kmeans_silhouette median_cost median_mm~1 media~2 media~3 media~4 media~5
##                    <int>       <dbl>       <dbl>   <dbl>   <dbl>   <dbl>   <dbl>
## 1                      1        2.72       22.6   2.12e4  0.0084    0.83 0
## 2                      2        3.28        1.03  1.50e4  0         0    0
## 3                      3     3571.          1.02  2.9 e1  0         0    0
## 4                      4        3.28       21.9   5.33e3  0.0213    2.05 0.00039
## 5                      5        3.28        1.03  2.43e6  0         0    0
## # ... with abbreviated variable names 1: median_mmbtu,
## #   2: median_received_units, 3: median_sulfur, 4: median_ash,
## #   5: median_mercury
```

```r
#iii.
Fuel_Plot <- ggplot(clus_silhouette) +
  aes(x = clus_kmeans_silhouette, fill = fuel_type_code_pudl) +
  geom_histogram(bins = 30L) +
  scale_fill_hue(direction = 1) +
  labs(
    x = "Clusters",
    y = "Count",
    title = "Fuel Types Used In Each Of The Cluster",
    fill = "Fuel Type"
  ) +
  theme_minimal() +
  theme(
    plot.title = element_text(size = 14L,
    face = "bold.italic",
    hjust = 0.5),
    axis.title.y = element_text(face = "bold"),
    axis.title.x = element_text(face = "bold")
  )
Fuel_Plot
```




```
#iv.
Heat_Content_Plot <- ggplot(clus_silhouette) +
  aes(x = clus_kmeans_silhouette, y = fuel_mmbtu_per_unit) +
  geom_jitter(size = 1.5) +
  labs(
    x = "Clusters",
    y = "Heat Content",
    title = "Heat Content Generated in Each Clusters",
    subtitle = "In Metric Million British Thermal Units (MMBtu)"
  ) +
  theme_minimal() +
  theme(
    plot.title = element_text(size = 14L,
    face = "bold.italic",
    hjust = 0.5),
    plot.subtitle = element_text(size = 10L,
    face = "italic",
    hjust = 0.5),
    axis.title.y = element_text(size = 10L,
    face = "bold"),
    axis.title.x = element_text(size = 10L,
    face = "bold")
  )
Heat_Content_Plot
```



Describing the cluster

Cluster - 1

Ember Cluster – The name states out the fuel used to generate power is coal (coal and petroleum coke), where the medial cost to produce power units of 22.551 MMBtu of heat content is 2.717$. The number of units thus received to produce power in this cluster is 21,179 units with the impurities levels of sulphur being 0.0081 ppm and the ash content in it being 0.83 ppm.

Cluster - 2

Commingled Cluster – This cluster has a mix of all the fuel types coal, gas and oil which is further divided into coal, natural gas, other gas and petroleum. The median cost of generating power units of 1.030 MMBtu of heat content in this cluster is 3.276$. The medial amount of units received and thereby used for power generation is 14,971.5 units with no impurities of sulphur, mercury and ash content in it.

Cluster - 3

Bohemian Cluster – This cluster has three data points, which are extreme outliers in the fuel cost variable where the median price for 29 natural gas units is pitched out to be 3571.218 $. There aren't any impurities seen in this cluster.

Cluster - 4

Smut Cluster – The fuel source used in this cluster is again coal. The price induced to generate power units of 21.930 MMBtu of heat content is 3.276$. The fuel units thus utilized to generate power is 5,328 units.

Lastly, the impurities of ash are greater than the permissible levels of 0.2 ppm i.e. 2.05 ppm and sulphur is greater as well i.e. 0.0213 ppm when compared to that of 0.005 ppm.

Cluster - 5

Ritzy Cluster – Among all the clusters thereby formed this cluster has the highest amount of gas units being received for producing power i.e. 24,32,817 units with zero (0) impurities. The median cost induced in generating power is 3.276$ for 1.030 MMBtu of heat content. The fuel type used for power generation in this cluster is primarily "Gas" with its subsets being Natural Gas and Other Gases.

Predictions

Prediction of fuel cost over the test set

```
set.seed(814)
model_lm <- lm(fuel_cost_per_mmbtu~.,data=Train_Data[,-c(1:3)])

#Running the multiple linear regression model using just two variables which have greater statistical s
set.seed(908)
model_lm_imp <- lm(fuel_cost_per_mmbtu~fuel_mmbtu_per_unit+fuel_received_units,data=Train_Data[,-c(1:3)]
summary(model_lm_imp)
```

```
##
## Call:
## lm(formula = fuel_cost_per_mmbtu ~ fuel_mmbtu_per_unit + fuel_received_units,
##     data = Train_Data[, -c(1:3)])
##
## Residuals:
##    Min     1Q Median     3Q    Max
##   -9.8   -6.0   -3.1    0.1 6867.3
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)             1.028e+01  1.346e+00   7.637 2.45e-14 ***
## fuel_mmbtu_per_unit -3.295e-01  9.714e-02  -3.392 0.000696 ***
## fuel_received_units -2.395e-06  1.326e-06  -1.806 0.070891 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 88.76 on 9127 degrees of freedom
## Multiple R-squared:  0.001394,   Adjusted R-squared:  0.001175
## F-statistic: 6.371 on 2 and 9127 DF,  p-value: 0.001719
```

```
model_lm_predict <-predict(model_lm_imp,Test_Data,type="response")
Test_Predict <- cbind(Test_Data,model_lm_predict)
#The predicted value seems far away from the actual values, this can be referred by looking at the "Tes
```