Final Project

2022-11-27

#Packages used for the current environment:

```
library(caret)
library(class)
library(tidyverse)
library(dlookr)
library(missRanger)
library(factoextra)
library(esquisse)
```

#1.Importing the dataset:

```
data<-read.csv("fuel.csv")
```

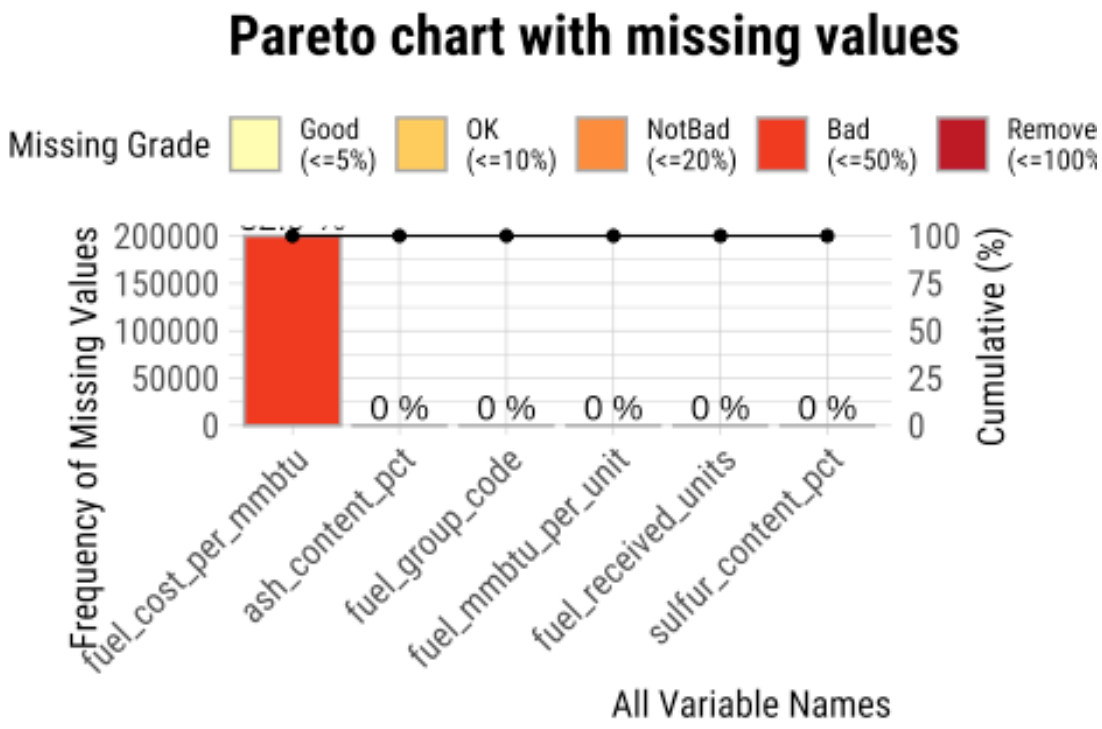
#2. Removing insignnificant variables and selecting main attributes for clustering to understand Power generation:

```
data_new<-data[,c(8,11:14,16)]
str(data_new)
```

```
## 'data.frame':    608565 obs. of  6 variables:
##  $ fuel_group_code    : chr  "coal" "coal" "natural_gas" "coal" ...
##  $ fuel_received_units: num  259412 52241 2783619 25397 764 ...
##  $ 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 ...
```

#3. Plotting missing values from the above dataset to check for missing values(Using the dlookr package): #dlookr package helps visually plot how many values are missing from each variable in percentages. This helps to understand the dataset and decide whether the missing values must be imputed or removed.

```
plot_na_pareto(data_new)
```



#The visual plot shows that fuel_cost_per_mmbtu has missing values of 32.9%. fuel_cost_per_mmbtu is an important predicting factor in understanding the heat generation and type of fuel sources. Therefore it is important to impute the missing values rather than completely removing them.

#4. Imputing missing values in fuel_cost_per_mmbtu using missRanger package:
#Imputation refers to replacing the missing values with different values that help complete the dataset. Imputation can be done in various methods. missRanger package imputes values of missing variables by using other variables as predictors. The process is repeated until the error rate stops improving.

```
data_clean<- missRanger(data_new, formula = .~., num.trees = 100, seed = 3)

##
## Missing value imputation by random forests
##
##   Variables to impute:       fuel_cost_per_mmbtu
##   Variables used to impute:  fuel_group_code, fuel_received_units, fuel_mm
btu_per_unit, sulfur_content_pct, ash_content_pct, fuel_cost_per_mmbtu
## iter 1:  .
```

#5.Sampling data and splitting data: #The population dataset with observations of 608565 has sampled to sample of 2% by setting the seed value as (9596).

```
set.seed(9596)
sample_data <- data_clean[sample(nrow(data_clean), size = 12000, replace = FA
LSE), ]
```

#6. Dataset has been partitioned into training and test sets with respect to the fuel_cost_per_mmbtu. Since fuel_cost_per_mmbtu helps understand how the heat output of the received fuel units behaves, the fuel cost has been set as an important factor in classifying the data.

```
train_index <- createDataPartition(sample_data$fuel_cost_per_mmbtu, p=0.75, l
ist = FALSE)
train_data<- sample_data[train_index,]
test_data<- sample_data[-train_index,]
```

#7.Subsetting numerical variables for the purpose of scaling and clustering:

```
cluster_data <- train_data %>% select('fuel_received_units', 'fuel_mmbtu_per_
unit', 'sulfur_content_pct', 'ash_content_pct', 'fuel_cost_per_mmbtu') #For t
he basis of clustering, the data set has been filtered to only represent only
numerical variables

cluster_train <- preProcess(cluster_data, method = c("center", "scale")) #Nor
malization of numerical values using center, scale. Center and scale was used
as the mean values to 0 and standard deviation to 1. This reduces the impact
of outliers in the data set as mean considers the lowest and  highest values
to calculate the average.
cluster_predict <- predict(cluster_train, cluster_data)
summary(cluster_predict)
```

```
##  fuel_received_units fuel_mmbtu_per_unit sulfur_content_pct ash_content_pc
t
##  Min.   :-0.3280     Min.   :-0.8987     Min.   :-0.51544   Min.   :-0.545
4
##  1st Qu.:-0.3233     1st Qu.:-0.7957     1st Qu.:-0.51544   1st Qu.:-0.545
4
##  Median :-0.3005     Median :-0.7917     Median :-0.51544   Median :-0.545
4
##  Mean   : 0.0000     Mean   : 0.0000     Mean   : 0.00000   Mean   : 0.000
0
##  3rd Qu.:-0.1894     3rd Qu.: 0.9173     3rd Qu.:-0.01812   3rd Qu.: 0.343
5
##  Max.   :18.9153     Max.   : 2.1649     Max.   : 7.10359   Max.   : 9.278
7
##  fuel_cost_per_mmbtu
##  Min.   :-0.15509
##  1st Qu.:-0.10356
##  Median :-0.07648
##  Mean   : 0.00000
```

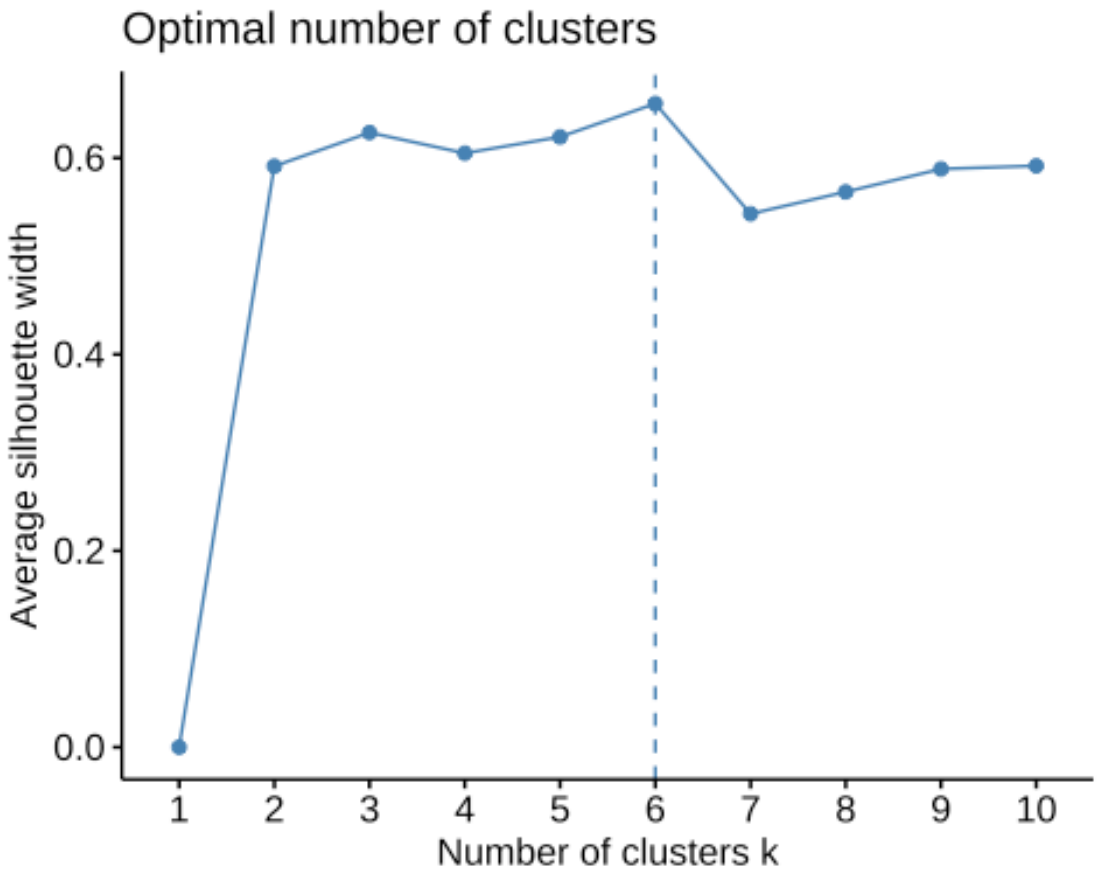
```
##  3rd Qu.:-0.02504
##  Max.   :74.77485
```

#8.Using the Silhouette method to find the optimal centers for clustering: #Clustering refers to a grouping of similar objects under one group. K-means clustering algorithm clusters the groups with the help of the K value, where each k value represents what group represents based on the centers of the data set and how various data points behave around these centers. Therefore, it is important to ascertain the value of k.

#Silhoutte method is one such method that helps ascertain the value of k. silhouette method defines the values of the cluster based on how data points behave within its own cluster and how each cluster is different from other clusters.

#Understanding the Business objective: The dataset is classified based on fuel_cost_per_mmbtu; silhouette helps understand how the data points in the cluster behave to cost within each cluster and how they differ from other clusters. This helps to analyze each cluster based on heat output which is sulfur and ash content which helps in determining the optimal cluster.

```
fviz_nbclust(cluster_predict, kmeans, method = "silhouette")
```

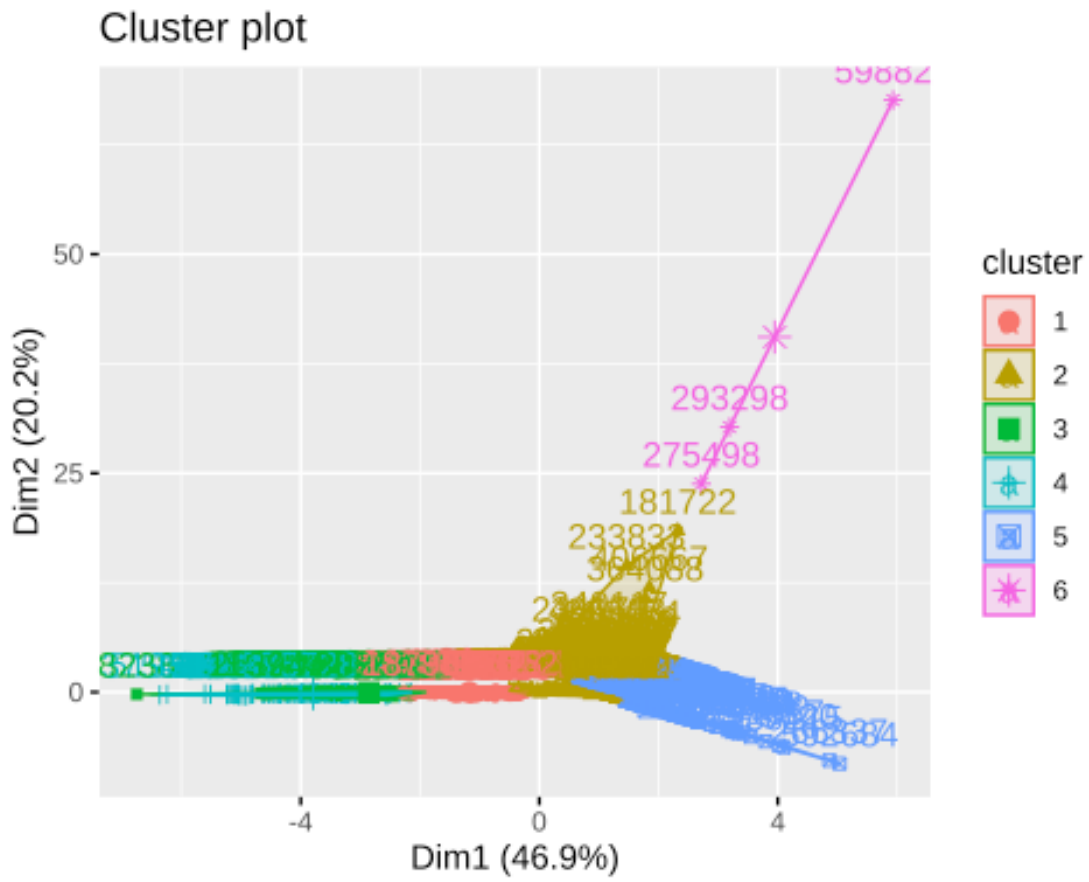


#9. Predicting clusters on k-means based on centers shown from silhouette method: #With the help of silhouette, we have already determined the centers = 6.

```
set.seed(9596)
kmeans_data <- kmeans(cluster_predict, centers = 6, nstart = 25)
```

#10.plotting of clusters based on clusters formed with the numerical dataset:

```
fviz_cluster(kmeans_data, data= cluster_data)
```



#11. Binding the clusters formed to the original numeric variables dataset: #Binding of the values of the cluster to the original data set helps us understand where all data points fall in different clusters.

```
cluster_group<- kmeans_data$cluster
group_cluster <- cbind(cluster_data, cluster_group)
```

#12. Checking the middlemost value of each cluster i.e., the median of each cluster: #With the help of aggregate function-Median, it helps us determine the middlemost value of each cluster.

```
aggregate(group_cluster,by=list(group_cluster$cluster_group), FUN="median")
```

```
##   Group.1 fuel_received_units fuel_mmbtu_per_unit sulfur_content_pct
## 1       1               25185              17.980                0.45
## 2       2               14212               1.030                0.00
## 3       3               18221              23.691                2.90
```

```
## 4       4                4217              13.240                0.73
## 5       5             2520150               1.029                0.00
## 6       6                  75               1.032                0.00
##   ash_content_pct fuel_cost_per_mmbtu cluster_group
## 1             6.2            2.398000             1
## 2             0.0            5.827617             2
## 3             8.8            2.353000             3
## 4            40.9            2.527628             4
## 5             0.0            5.801320             5
## 6             0.0         1600.589000             6
```

#Cluster 1 and Cluster 3 and 4: show a high fuel_mmbtu_per_unit median value with a lower median value of fuel_cost_per_mmbtu, which signifies that this cluster produces high heat for less cost. It also shows a significant amount of sulfur and ash content.

#Cluster 2 and #Cluster 5: Both clusters' median values show minimal heat output and cost incurred. The value of sulfur and ash output is shown as zero.

#Cluster 6: This cluster can be called an outlier as the median values of heat output is minimal, and the cost incurred is very high.

#13. Binding the final cluster to each fuel_group_code to interpret the clusters: #This helps us understand where all the data points of clustered data with respect to fuel sources used are classified.

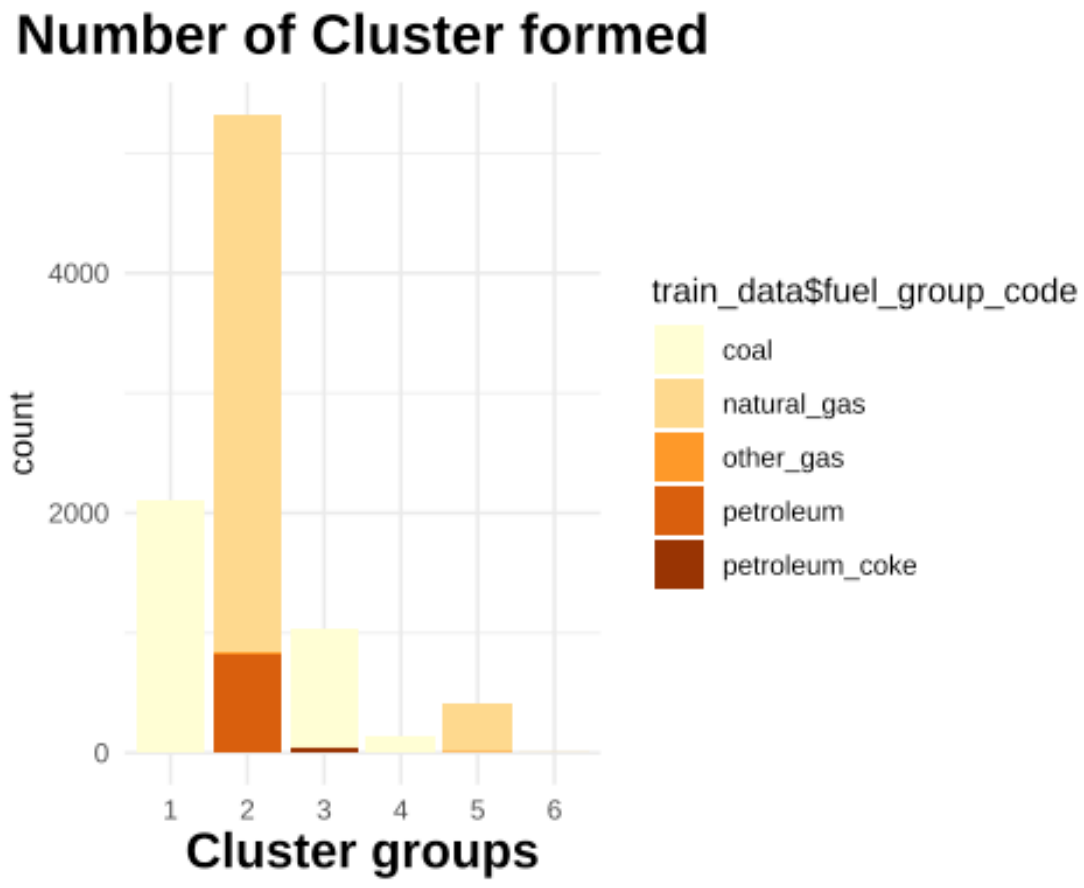
```
group_cluster$cluster_group <- as.factor(group_cluster$cluster_group)
final_cluster<- cbind(group_cluster, train_data$fuel_group_code)
head(final_cluster)
```

```
##        fuel_received_units fuel_mmbtu_per_unit sulfur_content_pct
## 197464                8569              26.048                1.81
## 466360               43236               1.020                0.00
## 174622              105552               1.012                0.00
## 36335                16334               1.017                0.00
## 606187                 893               5.712                0.00
## 163134                2591               5.750                0.25
##        ash_content_pct fuel_cost_per_mmbtu cluster_group
## 197464             7.5               2.873             3
## 466360             0.0               2.780             2
## 174622             0.0               6.469             2
## 36335              0.0               8.147             2
## 606187             0.0              17.632             2
## 163134             0.0              16.829             2
##        train_data$fuel_group_code
## 197464                       coal
## 466360                natural_gas
## 174622                natural_gas
## 36335                 natural_gas
## 606187                  petroleum
## 163134                  petroleum
```

#14.Visual presentation of number of clusters formed showed in form of ggplot2:

```
#esquisser()

ggplot(final_cluster) +
  aes(x = cluster_group, fill = `train_data$fuel_group_code`) +
  geom_bar() +
  scale_fill_brewer(palette = "YlOrBr", direction = 1) +
  labs(
    x = "Cluster groups",
    title = "Number of Cluster formed"
  ) +
  theme_minimal() +
  theme(
    plot.title = element_text(size = 18L,
    face = "bold",
    hjust = 0.5),
    axis.title.x = element_text(size = 16L,
    face = "bold")
  )
```



#15. The final dataset has been filtered to understand what each cluster represents: #With the silhouette, we have already determined that each cluster has been classified based on

the similarities of their data points. Therefore, filtering and understanding a few data points can help us conclude the overall behavior of the cluster. This can be used to find the optimal cluster for our business goal.

#a. Cluster 1 shows coal is major source of heat produced and with minimal cost.

```
cluster1<-final_cluster %>% select(fuel_mmbtu_per_unit,fuel_cost_per_mmbtu, c
luster_group) %>% group_by(train_data$fuel_group_code) %>% arrange(desc(fuel_
mmbtu_per_unit)) %>% filter(cluster_group == 1) %>% head()
cluster1
```

```
## # A tibble: 6 × 4
## # Groups:   train_data$fuel_group_code [1]
##   fuel_mmbtu_per_unit fuel_cost_per_mmbtu cluster_group train_data$fuel_gr
oup_…¹
##                 <dbl>               <dbl> <fct>         <chr>
## 1                29                  3.42 1             coal
## 2                29                  3.41 1             coal
## 3                29                  3.66 1             coal
## 4                29                  3.24 1             coal
## 5                27.8                4.24 1             coal
## 6                27.7                4.03 1             coal
## # … with abbreviated variable name ¹`train_data$fuel_group_code`
```

#b.From the below representation, it is evident that even though cluster 3 has high heat output for minimal cost, both coal and petroleum coke have sulfur and ash output.

```
cluster_imp<-final_cluster %>% select(fuel_mmbtu_per_unit,fuel_cost_per_mmbtu
, sulfur_content_pct, ash_content_pct , cluster_group, `train_data$fuel_group
_code`) %>% group_by(train_data$fuel_group_code) %>% arrange(desc(sulfur_cont
ent_pct))  %>% head()
cluster_imp
```

```
## # A tibble: 6 × 6
## # Groups:   train_data$fuel_group_code [2]
##   fuel_mmbtu_per_unit fuel_cost_per_mmbtu sulfur_conte…¹ ash_c…² clust…³ t
rain…⁴
##                 <dbl>               <dbl>          <dbl>   <dbl> <fct>   <
chr>
## 1                20.1                1.81           7.66    28.4 3       c
oal
## 2                28.6                2.13           6.93     0   3       p
etrol…
## 3                29.3                0.888          6.8      2.2 3       p
etrol…
## 4                28.3                2.52           6.6      0.4 3       p
etrol…
## 5                29.8                0.954          6.5      1.7 3       p
etrol…
## 6                29.0                2.53           6.39     0.2 3       p
etrol…
```

```
## # … with abbreviated variable names ¹sulfur_content_pct, ²ash_content_pct,
## #   ³cluster_group, ⁴`train_data$fuel_group_code`
```

#c.From above since we already know that the median values of cluster 2 have zero sulfur and ash output. we can observe that their heat and cost are minimal, although one data point shows a high cost. This could be because this cluster has outliers.

```
cluster2<-final_cluster %>% select(fuel_mmbtu_per_unit,fuel_cost_per_mmbtu, c
luster_group, `train_data$fuel_group_code`) %>% filter(train_data$fuel_group_
code =='natural_gas') %>%arrange(desc(fuel_mmbtu_per_unit)) %>% filter(cluste
r_group == 2) %>% head()
cluster2
```

```
##   fuel_mmbtu_per_unit fuel_cost_per_mmbtu cluster_group
## 1               1.248               3.707             2
## 2               1.248               4.177             2
## 3               1.247               8.554             2
## 4               1.245              14.766             2
## 5               1.244               5.173             2
## 6               1.242               4.159             2
##   train_data$fuel_group_code
## 1                 natural_gas
## 2                 natural_gas
## 3                 natural_gas
## 4                 natural_gas
## 5                 natural_gas
## 6                 natural_gas
```

#d.This cluster is the same as cluster 1 as it is dominated by coal but petroleum coke has high heat output and low cost.

```
cluster3<-final_cluster %>% select(fuel_mmbtu_per_unit,fuel_cost_per_mmbtu, c
luster_group) %>% group_by(train_data$fuel_group_code) %>% arrange(desc(fuel_
mmbtu_per_unit)) %>% filter(cluster_group == 3) %>% head()
cluster3
```

```
## # A tibble: 6 × 4
## # Groups:   train_data$fuel_group_code [1]
##   fuel_mmbtu_per_unit fuel_cost_per_mmbtu cluster_group train_data$fuel_gr
oup_…¹
##                 <dbl>               <dbl> <fct>         <chr>
## 1                30                 2.44  3             petroleum_coke
## 2                29.8               0.954 3             petroleum_coke
## 3                29.6               3.68  3             petroleum_coke
## 4                29.3               4.00  3             petroleum_coke
## 5                29.3               0.888 3             petroleum_coke
## 6                29.3               2.09  3             petroleum_coke
## # … with abbreviated variable name ¹`train_data$fuel_group_code`
```

#e.This cluster is the same as cluster 1 as it is dominated by coal with high heat output and minimal cost.

```
cluster4<-final_cluster %>% select(fuel_mmbtu_per_unit,fuel_cost_per_mmbtu, c
luster_group) %>% group_by(train_data$fuel_group_code) %>% arrange(desc(fuel_
mmbtu_per_unit)) %>% filter(cluster_group == 4) %>% head()
cluster4
```

```
## # A tibble: 6 × 4
## # Groups:   train_data$fuel_group_code [1]
##   fuel_mmbtu_per_unit fuel_cost_per_mmbtu cluster_group train_data$fuel_gr
oup_…¹
##                 <dbl>               <dbl> <fct>         <chr>
## 1                19.9                2.21 4             coal
## 2                19.7                1.52 4             coal
## 3                19.7                2.7  4             coal
## 4                19.6                2.05 4             coal
## 5                19.0                2.15 4             coal
## 6                18.8                2.22 4             coal
## # … with abbreviated variable name ¹`train_data$fuel_group_code`
```

#f.This cluster shows uniform characteristics with minimal heat and cost, and all data points in this cluster are represented by Natural gas. This could be called an optimal cluster for recommending current business problems.

```
cluster5<-final_cluster %>% select(fuel_mmbtu_per_unit,fuel_cost_per_mmbtu, c
luster_group) %>% group_by(train_data$fuel_group_code) %>% arrange(desc(fuel_
mmbtu_per_unit)) %>% filter(cluster_group == 5) %>% head()
cluster5
```

```
## # A tibble: 6 × 4
## # Groups:   train_data$fuel_group_code [1]
##   fuel_mmbtu_per_unit fuel_cost_per_mmbtu cluster_group train_data$fuel_gr
oup_…¹
##                 <dbl>               <dbl> <fct>         <chr>
## 1                1.12                3.36 5             natural_gas
## 2                1.10                4.33 5             natural_gas
## 3                1.1                 6.27 5             natural_gas
## 4                1.1                 4.70 5             natural_gas
## 5                1.09                3.69 5             natural_gas
## 6                1.09                2.50 5             natural_gas
## # … with abbreviated variable name ¹`train_data$fuel_group_code`
```

#g.This cluster has only 3 data points which signifies that it has outliers as the heat output is minimal, and the cost output is very high.

```
cluster6<-final_cluster %>% select(fuel_mmbtu_per_unit,fuel_cost_per_mmbtu, c
luster_group) %>% group_by(train_data$fuel_group_code) %>% arrange(desc(fuel_
mmbtu_per_unit)) %>% filter(cluster_group == 6)
cluster6
```

```
## # A tibble: 3 × 4
## # Groups:   train_data$fuel_group_code [1]
##   fuel_mmbtu_per_unit fuel_cost_per_mmbtu cluster_group train_data$fuel_gr
```

```
oup_…¹
##                   <dbl>               <dbl> <fct>         <chr>
## 1                  1.04               1601. 6             natural_gas
## 2                  1.03               1258. 6             natural_gas
## 3                  1.01               3577. 6             natural_gas
## # … with abbreviated variable name ¹`train_data$fuel_group_code`
```