Kmeans clustering of environmentally friendly plants

Franklin Ngochi

2023-05-07

## Loading required libraries and setting seed

set.seed(1243)  
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(factoextra)

## Loading required package: ggplot2

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

library(caret)

## Loading required package: lattice

#### Load fuel receipts data

Fuel<-read.csv(file="C:\\Users\\ngoch\\OneDrive\\Documents\\KSU\\Fundamentals of Machine Learning\\fuel\_receipts\_costs\_eia923.csv", header=TRUE, sep=",")  
  
#dropping variables with more than 50% NAs and rows with NAs.  
Fuel<-Fuel[,colSums(is.na(Fuel))<0.5\*nrow(Fuel)]  
  
#Sampling 2% of data.  
Fuel<-Fuel[sample(nrow(Fuel), floor(0.02\*nrow(Fuel))),]  
  
#Assigning proper classes to variables  
Fuel<- Fuel%>%mutate(plant\_id\_eia= as.factor(plant\_id\_eia),contract\_type\_code = as.factor(contract\_type\_code), energy\_source\_code=as.factor(energy\_source\_code), fuel\_type\_code\_pudl=as.factor(fuel\_type\_code\_pudl),fuel\_group\_code=as.factor(fuel\_group\_code), supplier\_name=as.factor(supplier\_name), primary\_transportation\_mode\_code=as.factor(primary\_transportation\_mode\_code), secondary\_transportation\_mode\_code=as.factor(secondary\_transportation\_mode\_code), natural\_gas\_transport\_code=as.factor(natural\_gas\_transport\_code))  
Fuel$contract\_expiration\_date<-as.Date(Fuel$contract\_expiration\_date)  
nrow(Fuel)

## [1] 12171

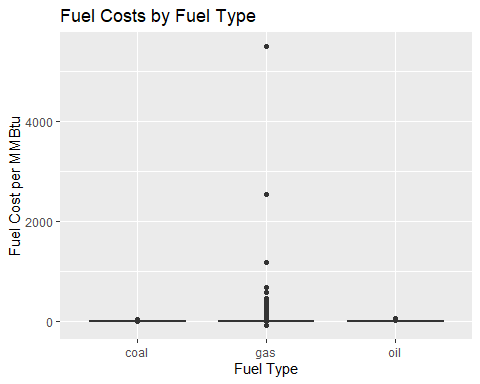
#Data exploration

summary(Fuel[,-1])

## plant\_id\_eia plant\_id\_eia\_label report\_date contract\_type\_code  
## 2964 : 116 Length:12171 Length:12171 : 7   
## 2965 : 104 Class :character Class :character C :5453   
## 2963 : 101 Mode :character Mode :character NC: 85   
## 55970 : 100 S :6458   
## 3478 : 89 T : 168   
## 7527 : 83   
## (Other):11578   
## contract\_type\_code\_label contract\_expiration\_date energy\_source\_code  
## Length:12171 Min. :0001-01-20 NG :6678   
## Class :character 1st Qu.:0010-01-20 BIT :2663   
## Mode :character Median :0012-01-20 SUB :1436   
## Mean :0010-03-13 DFO : 917   
## 3rd Qu.:0012-01-20 WC : 151   
## Max. :0012-01-20 RFO : 119   
## NA's :6967 (Other): 207   
## energy\_source\_code\_label fuel\_type\_code\_pudl fuel\_group\_code  
## Length:12171 coal:4376 coal :4315   
## Class :character gas :6727 natural\_gas :6678   
## Mode :character oil :1068 other\_gas : 49   
## petroleum :1068   
## petroleum\_coke: 61   
##   
##   
## supplier\_name fuel\_received\_units  
## various (natural gas spot purchases only): 512 Min. : 1   
## peabody coal sales : 259 1st Qu.: 3844   
## arch : 225 Median : 22532   
## arch coal sales : 211 Mean : 263499   
## sequent : 167 3rd Qu.: 114480   
## alpha coal : 164 Max. :48159765   
## (Other) :10633   
## fuel\_mmbtu\_per\_unit sulfur\_content\_pct ash\_content\_pct mercury\_content\_ppm  
## Min. : 0.000 Min. :0.0000 Min. : 0.000 Min. :0.000   
## 1st Qu.: 1.025 1st Qu.:0.0000 1st Qu.: 0.000 1st Qu.:0.000   
## Median : 1.060 Median :0.0000 Median : 0.000 Median :0.000   
## Mean : 8.771 Mean :0.5159 Mean : 3.552 Mean :0.008   
## 3rd Qu.:17.817 3rd Qu.:0.5000 3rd Qu.: 5.800 3rd Qu.:0.000   
## Max. :30.000 Max. :7.6600 Max. :67.300 Max. :0.900   
## NA's :5759   
## fuel\_cost\_per\_mmbtu primary\_transportation\_mode\_code  
## Min. : -71.930 PL :5676   
## 1st Qu.: 2.311 RR :2672   
## Median : 3.289 TR :1696   
## Mean : 6.786 :1182   
## 3rd Qu.: 4.812 RV : 784   
## Max. :5489.130 TC : 87   
## NA's :4006 (Other): 74   
## primary\_transportation\_mode\_code\_label secondary\_transportation\_mode\_code  
## Length:12171 :11547   
## Class :character TR : 234   
## Mode :character RV : 142   
## PL : 81   
## RR : 73   
## GL : 31   
## (Other): 63   
## secondary\_transportation\_mode\_code\_label natural\_gas\_transport\_code  
## Length:12171 :5309   
## Class :character firm :4731   
## Mode :character interruptible:2131   
##   
##   
##   
##   
## natural\_gas\_delivery\_contract\_type\_code data\_maturity data\_maturity\_label  
## Length:12171 Length:12171 Length:12171   
## Class :character Class :character Class :character   
## Mode :character Mode :character Mode :character   
##   
##   
##   
##

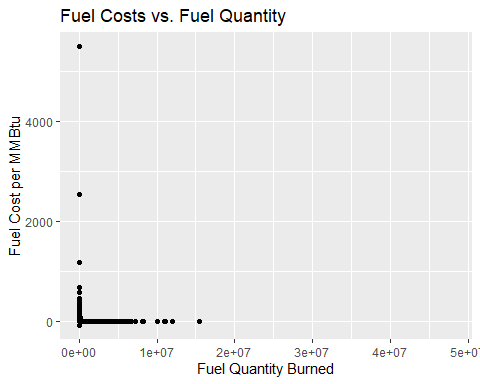
#Boxplot of fuel costs by fuel type  
ggplot(data = Fuel, aes(x = fuel\_type\_code\_pudl, y = fuel\_cost\_per\_mmbtu)) +  
 geom\_boxplot() +  
 xlab("Fuel Type") +  
 ylab("Fuel Cost per MMBtu") +  
 ggtitle("Fuel Costs by Fuel Type")

## Warning: Removed 4006 rows containing non-finite values (`stat\_boxplot()`).



#Scatterplot of fuel costs vs. fuel quantity  
ggplot(data = Fuel, aes(x = fuel\_received\_units, y = fuel\_cost\_per\_mmbtu)) +  
 geom\_point() +  
 xlab("Fuel Quantity Burned") +  
 ylab("Fuel Cost per MMBtu") +  
 ggtitle("Fuel Costs vs. Fuel Quantity")

## Warning: Removed 4006 rows containing missing values (`geom\_point()`).

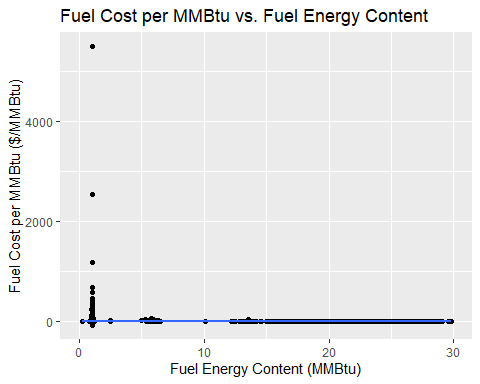


#Fuel Cost per MMBtu vs. Fuel Energy Content  
ggplot(data=Fuel, aes(x = fuel\_mmbtu\_per\_unit, y = fuel\_cost\_per\_mmbtu)) +   
 geom\_point() +  
 labs(title = "Fuel Cost per MMBtu vs. Fuel Energy Content",  
 x = "Fuel Energy Content (MMBtu)",  
 y = "Fuel Cost per MMBtu ($/MMBtu)") +  
 geom\_smooth(method = "lm", se = FALSE)

## `geom\_smooth()` using formula = 'y ~ x'

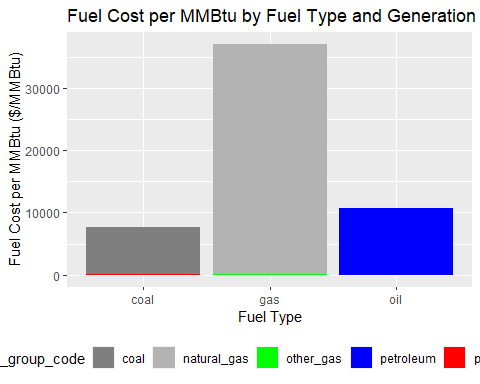
## Warning: Removed 4006 rows containing non-finite values (`stat\_smooth()`).

## Warning: Removed 4006 rows containing missing values (`geom\_point()`).

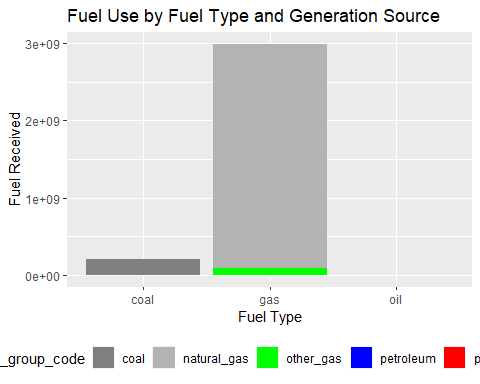


#Fuel Cost per MMBtu by Fuel Type and Generation Source  
ggplot(data=Fuel, aes(x = fuel\_type\_code\_pudl, y = fuel\_cost\_per\_mmbtu, fill = fuel\_group\_code)) +  
 geom\_bar(stat = "identity") +  
 labs(title = "Fuel Cost per MMBtu by Fuel Type and Generation Source",  
 x = "Fuel Type",  
 y = "Fuel Cost per MMBtu ($/MMBtu)") +  
 scale\_fill\_manual(values = c("gray50", "gray70", "green", "blue", "red")) +  
 theme(legend.position = "bottom")

## Warning: Removed 4006 rows containing missing values (`position\_stack()`).



#Fuel Cost per MMBtu by Fuel Type and Generation Source  
ggplot(data=Fuel, aes(x = fuel\_type\_code\_pudl, y = fuel\_received\_units, fill = fuel\_group\_code)) +  
 geom\_bar(stat = "identity") +  
 labs(title = "Fuel Use by Fuel Type and Generation Source",  
 x = "Fuel Type",  
 y = "Fuel Received") +  
 scale\_fill\_manual(values = c("gray50", "gray70", "green", "blue", "red")) +  
 theme(legend.position = "bottom")



#Select variables to perform Kmeans clustering of power plants by fuel usage pattern

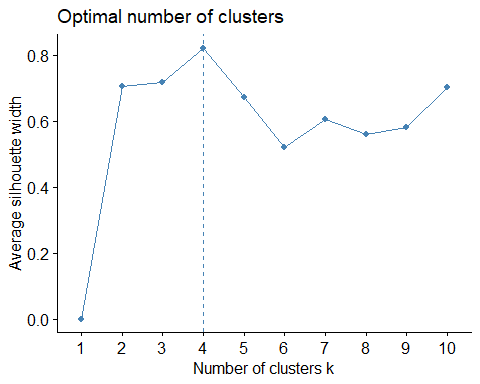
# Select a subset of variables relevant to fuel usage  
fuel\_vars <- Fuel %>%select(plant\_id\_eia\_label,fuel\_type\_code\_pudl, fuel\_mmbtu\_per\_unit, sulfur\_content\_pct, ash\_content\_pct, mercury\_content\_ppm) %>%na.omit()  
  
  
fuel\_vars$fuel\_type\_code\_pudl<-as.factor(fuel\_vars$fuel\_type\_code\_pudl)  
fuel\_vars$plant\_id\_eia\_label<-as.factor(fuel\_vars$plant\_id\_eia\_label)

#Data transformation and scaling

#Transform factor variable to dummy  
dummy\_fuel<-fastDummies::dummy\_cols(fuel\_vars, select\_columns = c("fuel\_type\_code\_pudl"), remove\_first\_dummy = FALSE, remove\_selected\_columns = TRUE)  
  
#Split 75/25 between training and validation set  
train\_idx <- createDataPartition(y = dummy\_fuel$fuel\_mmbtu\_per\_unit, p = 0.75, list = FALSE)  
train\_data <- dummy\_fuel[train\_idx, ]  
test\_data <- dummy\_fuel[-train\_idx, ]  
  
# Scale the variables  
scaled\_train <- scale(train\_data[, -c(1)])  
scaled\_test <- scale(test\_data[, -c(1)])

## Determine the optimal number of clusters

nb\_clusters <- fviz\_nbclust(scaled\_train, kmeans, method = 'silhouette')  
nb\_clusters



# Perform kmeans clustering on training data for k=4

kmeans\_model <- kmeans(scaled\_train, centers = 4)  
paste(kmeans\_model$centers)

## [1] "1.31906489528042" "1.65074480783634" "-0.176509360304651"  
## [4] "-0.695717055576699" "0.0652727316246226" "2.40352934994684"   
## [7] "-0.377199029983965" "-0.448677688864477" "0.654242390011784"   
## [10] "1.64736166424643" "-0.459570411344165" "-0.459570411344185"  
## [13] "0.262415403559194" "0.6662210456707" "-0.185295167851625"  
## [16] "-0.185295167851625" "1.51558123255881" "1.51558123255881"   
## [19] "-0.659675721450865" "-0.659675721450867" "-1.25560085740939"   
## [22] "-1.25560085740939" "-1.25560085740939" "0.79626589700521"   
## [25] "-0.304376135372508" "-0.304376135372508" "3.2847257942283"   
## [28] "-0.304376135372528"

kmeans\_model$size

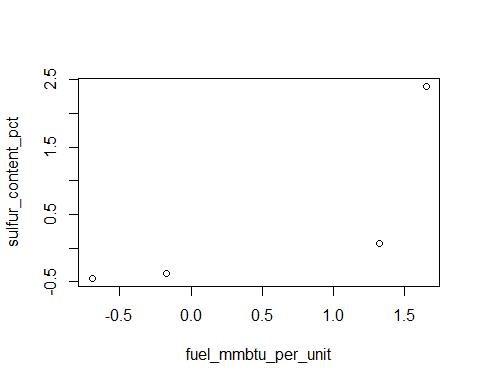
## [1] 869 590 408 2944

kmeans\_model$centers

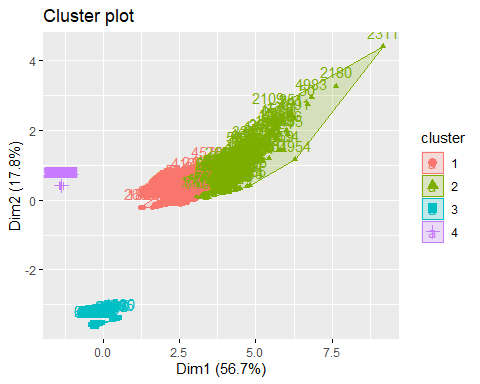
## fuel\_mmbtu\_per\_unit sulfur\_content\_pct ash\_content\_pct mercury\_content\_ppm  
## 1 1.3190649 0.06527273 0.6542424 0.2624154  
## 2 1.6507448 2.40352935 1.6473617 0.6662210  
## 3 -0.1765094 -0.37719903 -0.4595704 -0.1852952  
## 4 -0.6957171 -0.44867769 -0.4595704 -0.1852952  
## fuel\_type\_code\_pudl\_coal fuel\_type\_code\_pudl\_gas fuel\_type\_code\_pudl\_oil  
## 1 1.5155812 -1.2556009 -0.3043761  
## 2 1.5155812 -1.2556009 -0.3043761  
## 3 -0.6596757 -1.2556009 3.2847258  
## 4 -0.6596757 0.7962659 -0.3043761

plot(kmeans\_model$centers)

#Cluster 3 and 4 have less environmental impact than clusters 1 and 2 because their emissions are lower. However, the clusters with low environmental impact also generate less heat.



#Visualize clusters  
fviz\_cluster(kmeans\_model, data=scaled\_train)



#Using multiple-linear regression to determine the best set of variables to predict fuel\_cost\_per\_mmbtu

# Select a subset of variables relevant to fuel cost  
fuel\_vars1 <- Fuel %>%select(fuel\_cost\_per\_mmbtu, fuel\_received\_units, fuel\_type\_code\_pudl, fuel\_mmbtu\_per\_unit) %>%na.omit()  
fuel\_vars1$fuel\_type\_code\_pudl<-as.factor(fuel\_vars1$fuel\_type\_code\_pudl)  
  
#Transform factor variable to dummy  
dummy\_fuel1<-fastDummies::dummy\_cols(fuel\_vars1, select\_columns = c("fuel\_type\_code\_pudl","primary\_transportation\_mode\_code", "secondary\_transportation\_mode\_code", "natural\_gas\_transport\_code"), remove\_first\_dummy = FALSE, remove\_selected\_columns = TRUE)

## Warning in fastDummies::dummy\_cols(fuel\_vars1, select\_columns = c("fuel\_type\_code\_pudl", : NOTE: The following select\_columns input(s) is not a column in data.  
##

#Split 75/25 between training and validation set  
train\_idx1 <- createDataPartition(y = dummy\_fuel1$fuel\_cost\_per\_mmbtu, p = 0.75, list = FALSE)  
train\_data1 <- dummy\_fuel1[train\_idx, ]  
test\_data1 <- dummy\_fuel1[-train\_idx, ]  
model <- lm(fuel\_cost\_per\_mmbtu~ ., data = train\_data1)  
  
summary(model)

##   
## Call:  
## lm(formula = fuel\_cost\_per\_mmbtu ~ ., data = train\_data1)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -81.1 -5.1 -1.5 0.0 5479.9   
##   
## Coefficients: (1 not defined because of singularities)  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 1.597e+01 5.300e+00 3.012 0.00261 \*\*  
## fuel\_received\_units -1.911e-06 1.899e-06 -1.006 0.31426   
## fuel\_mmbtu\_per\_unit 1.217e-01 5.383e-01 0.226 0.82112   
## fuel\_type\_code\_pudl\_coal -1.599e+01 9.590e+00 -1.668 0.09541 .   
## fuel\_type\_code\_pudl\_gas -6.896e+00 5.262e+00 -1.311 0.19007   
## fuel\_type\_code\_pudl\_oil NA NA NA NA   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 81.32 on 4806 degrees of freedom  
## Multiple R-squared: 0.002671, Adjusted R-squared: 0.001841   
## F-statistic: 3.218 on 4 and 4806 DF, p-value: 0.012

# From model summary, we cannot reject the null hypothesis that the coefficient of "fuel\_mmbtu\_per\_unit" is equal to zero . The "variable fuel\_type\_code\_pudl\_oil" also presents NA. This suggests that these two variables are not good predictors of fuel cost

#Dropping non-significant variables and checking prediction on test set

new\_model <- update(model, fuel\_cost\_per\_mmbtu~fuel\_received\_units+fuel\_type\_code\_pudl\_coal+fuel\_type\_code\_pudl\_gas, data = train\_data1)  
summary(new\_model)

##   
## Call:  
## lm(formula = fuel\_cost\_per\_mmbtu ~ fuel\_received\_units + fuel\_type\_code\_pudl\_coal +   
## fuel\_type\_code\_pudl\_gas, data = train\_data1)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -81.1 -5.1 -1.6 0.0 5479.9   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 1.668e+01 4.268e+00 3.908 9.45e-05 \*\*\*  
## fuel\_received\_units -1.921e-06 1.898e-06 -1.012 0.31166   
## fuel\_type\_code\_pudl\_coal -1.410e+01 4.673e+00 -3.018 0.00256 \*\*   
## fuel\_type\_code\_pudl\_gas -7.479e+00 4.588e+00 -1.630 0.10317   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 81.31 on 4807 degrees of freedom  
## Multiple R-squared: 0.00266, Adjusted R-squared: 0.002038   
## F-statistic: 4.274 on 3 and 4807 DF, p-value: 0.005073

predicted <- predict(new\_model, newdata = test\_data1)  
R2 <- cor(test\_data1$fuel\_cost\_per\_mmbtu, predicted)^2  
RMSE <- sqrt(mean((test\_data1$fuel\_cost\_per\_mmbtu - predicted)^2))  
R2

## [1] 0.006945111

RMSE

## [1] 49.89317