Final Exam Machine Learning

AKANKSHA NADUKULA

2022-12-07

```
library("readr")

## Warning: package 'readr' was built under R version 4.1.3

library(readr)
Fuel_Receipts<-read.csv('C:/Users/VIJAY
KUMAR/Downloads/fuel_receipts_costs_eia923 (1).csv')
str(Fuel_Receipts)

## 'data.frame':    608565 obs. of  23 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 ...
##  $ report_date                          : chr  "2008-01-01" "2008-01-01"
"2008-01-01" "2008-01-01" ...
##  $ contract_type_code                   : chr  "C" "C" "C" "C" ...
##  $ contract_expiration_date             : chr  "2008-04-01" "2008-04-01"
"" "2015-12-01" ...
##  $ energy_source_code                   : 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 ...
##  $ supplier_name                        : chr  "interocean coal"
"interocean coal" "bay gas pipeline" "alabama 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 ...
##  $ 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" ...
##  $ secondary_transportation_mode_code   : 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" ...

#installing required libraries
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)

## Warning: package 'caret' was built under R version 4.1.3

## Loading required package: ggplot2

## Warning: package 'ggplot2' was built under R version 4.1.3

## Loading required package: lattice

library(missForest)

## Warning: package 'missForest' was built under R version 4.1.3

library(corrplot)

## Warning: package 'corrplot' was built under R version 4.1.3

## corrplot 0.92 loaded

library(factoextra)

## Warning: package 'factoextra' was built under R version 4.1.3

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

library(fpc)

## Warning: package 'fpc' was built under R version 4.1.3

library(StatMatch)
```

```
## Warning: package 'StatMatch' was built under R version 4.1.3

## Loading required package: proxy

## Warning: package 'proxy' was built under R version 4.1.3

##
## Attaching package: 'proxy'

## The following objects are masked from 'package:stats':
##
##     as.dist, dist

## The following object is masked from 'package:base':
##
##     as.matrix

## Loading required package: survey

## Warning: package 'survey' was built under R version 4.1.3

## Loading required package: grid

## Loading required package: Matrix

## Warning: package 'Matrix' was built under R version 4.1.3

## Loading required package: survival

##
## Attaching package: 'survival'

## The following object is masked from 'package:caret':
##
##     cluster

##
## Attaching package: 'survey'

## The following object is masked from 'package:graphics':
##
##     dotchart

## Loading required package: lpSolve

## Warning: package 'lpSolve' was built under R version 4.1.3

library(cluster)

set.seed(4567)
fuel_meter = Fuel_Receipts[,c(1,11,12,13,14,15,16)]
str(fuel_meter)
```

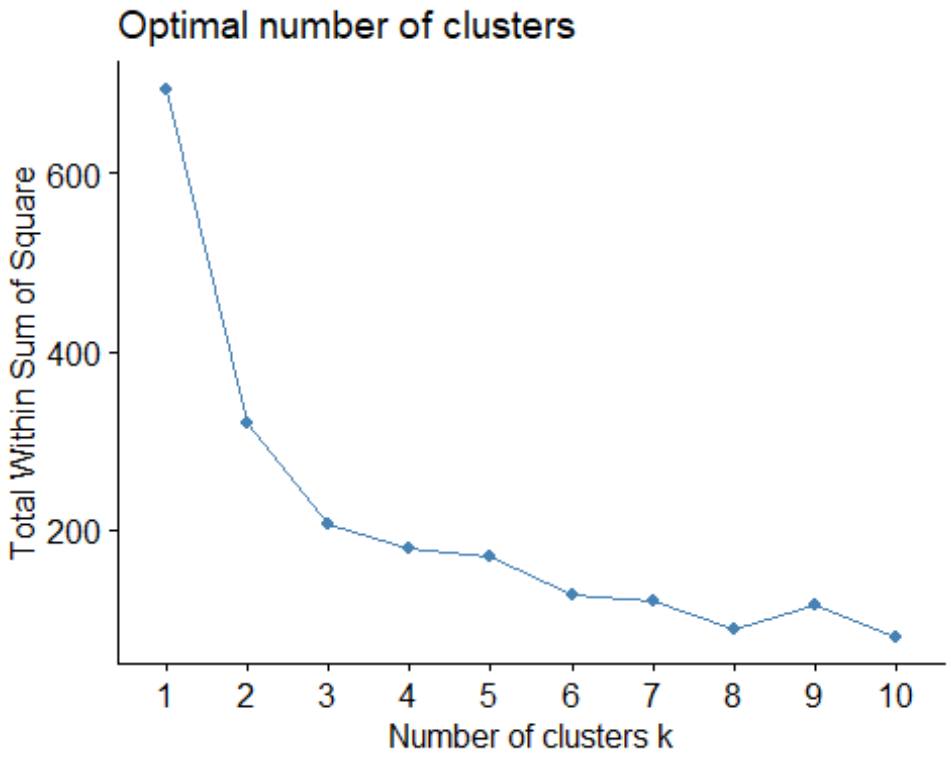
```
## 'data.frame':    608565 obs. of  7 variables:
##  $ rowid              : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ fuel_received_units: num  259412 52241 2783619 25397 764 ...
##  $ fuel_mmbtu_per_unit: num  23.1 22.8 1.04 24.61 24.45 ...
##  $ sulfur_content_pct : num  0.49 0.48 0 1.69 0.84 1.54 0 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 ...
##  $ 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 ...

colMeans(is.na(fuel_meter))

##               rowid fuel_received_units fuel_mmbtu_per_unit
sulfur_content_pct
##           0.0000000           0.0000000           0.0000000
0.0000000
##     ash_content_pct mercury_content_ppm fuel_cost_per_mmbtu
##           0.0000000           0.4756797           0.3290363

fuel_meter=na.omit(fuel_meter)
colSums(is.na(fuel_meter))

##               rowid fuel_received_units fuel_mmbtu_per_unit
sulfur_content_pct
##                   0                   0                   0
0
##     ash_content_pct mercury_content_ppm fuel_cost_per_mmbtu
##                   0                   0                   0

fuel_meter=fuel_meter %>% sample_frac(0.02)
```

#Cleanning Data

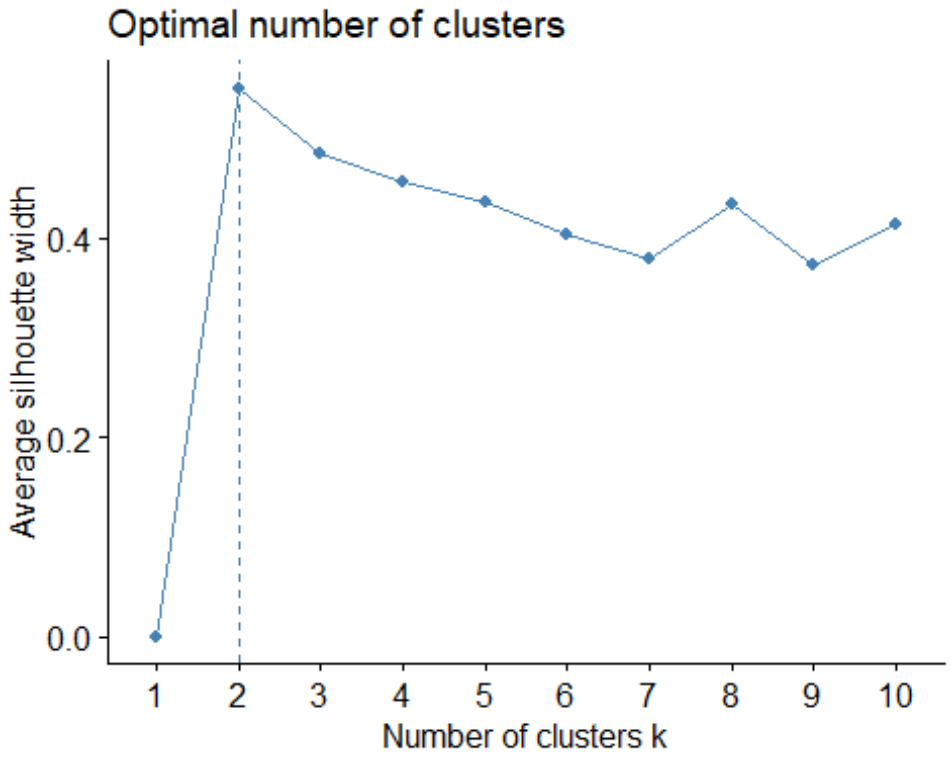
```
set.seed(1632)
data_partition=createDataPartition(fuel_meter$sulfur_content_pct,p=0.75,list
= FALSE)
data.train = fuel_meter[data_partition,]
data.test = fuel_meter[-data_partition,]
```

#Data Normalization

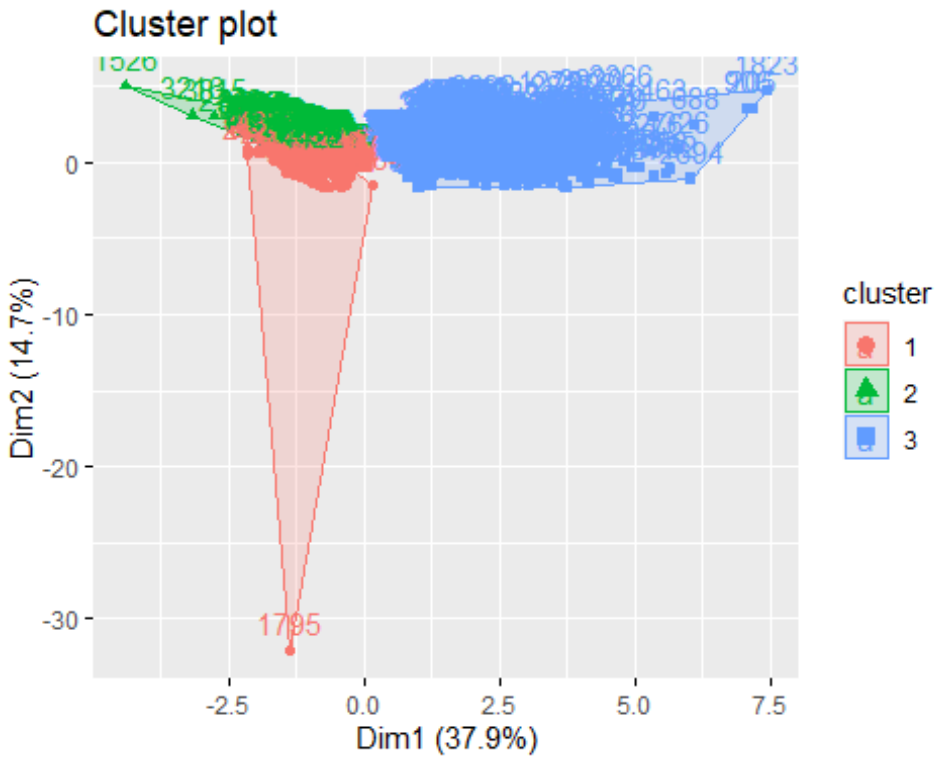
```
process=preProcess(data.train,method = "range")
normalization.data=predict(process,as.data.frame(data.train))

wss=fviz_nbclust(normalization.data,kmeans,method = "wss")
wss
```




```
silho.=fviz_nbclust(normalization.data,kmeans,method = "silhouette")
silho.
```



```
elbow_kmeans=kmeans(normalization.data,centers = 3,nstart = 50)
silhoue_kmeans=kmeans(normalization.data,centers = 2,nstart = 50)
fviz_cluster(elbow_kmeans,data=normalization.data)
```




```
fviz_cluster(silhoue_kmeans,data = normalization.data)
```


```
data.train$cluster = silhoue_kmeans$cluster
data.train%>%group_by(cluster)%>%
summarise(Avg_receivedunits=mean(fuel_received_units),content_of_sulphur =
mean(sulfur_content_pct), avg_ash =
mean(ash_content_pct),avg_fuel_cost=mean(fuel_mmbtu_per_unit))

## # A tibble: 2 x 5
##   cluster Avg_receivedunits content_of_sulphur avg_ash avg_fuel_cost
##     <int>             <dbl>              <dbl>   <dbl>         <dbl>
## 1       1           296269.            0.00298    0             1.61
## 2       2            56123.            1.32       7.97         20.9
```