FINAL EXAM

ALLEN RICHARDS

2023-12-05

Summary : The code applies k-means clustering to a fuel receipts dataset, experiments with various configurations to find the ideal number of clusters, and outputs summaries and visualizations of the clustering outcomes. It also looks at how different fuel types are distributed within the designated clusters. The code loads the necessary libraries, imports the dataset, and fixes any missing values. After being extracted, numerical columns are scaled in preparation for clustering. To help determine the ideal number of clusters, a preliminary k-means clustering with k=3 is carried out. Cluster visualizations, such as within-cluster sum of squares and silhouette width plots, are produced. For this, the elbow method and the silhouette method are used. Based on the elbow and silhouette approaches, two more k-means clustering studies are carried out, each with a different k value. Together with a comprehensive analysis of the dataset, the code verifies variable names and data types and displays a table showing the distribution of fuel types across the clusters that have been detected. Lastly, bar plots offer a thorough examination of the dataset's clustering patterns by visualizing the distribution of fuel types and energy source codes inside clusters for k values of 2 and 3.

> What is the best K? The Elbow approach and silhouette analysis are two popular techniques for figuring out the optimal value for k (number of clusters) in k-means clustering. I consider these are the best "K".

> Describe your clusters. Provide relevant tables and graphs to support your conclusion. Information about the features and composition of the clusters. I can recognize patterns in the parallel coordinate plot, decipher the cluster centers plot's average behavior, and evaluate the scatterplot's spacing between clusters. It appears that the primary focus is on exploring and visualizing clusters derived from k-means clustering

> What can you say about the relative composition of the different fuel types in relation to your clusters? Most likely, the table consists of cells with numbers, rows that indicate clusters, and columns that represent different types of fuel. Distinct patterns in the dataset are revealed by analyzing the relative content of various fuel types within the identified clusters. We have clustered the data using k-means clustering, which is based on similarities in the numerical features. Important information about the distribution of fuel types among these clusters may be found in the cross-tabulation table that compares fuel types with cluster assignments. The proportionate representation of different fuel types within each cluster is shown in this table, which helps us identify preferences or traits unique to a particular cluster. The accompanying bar charts, which show the distribution of fuel types within each cluster graphically, add even more insight to the output.

```
library('tidyverse')
```

```
## Warning: package 'ggplot2' was built under R version 4.3.2
```

```
## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──
## ✓ dplyr     1.1.3     ✓ readr     2.1.4
## ✓ forcats   1.0.0     ✓ stringr   1.5.0
## ✓ ggplot2   3.4.4     ✓ tibble    3.2.1
## ✓ lubridate 1.9.2     ✓ tidyr     1.3.0
## ✓ purrr     1.0.2
## ── Conflicts ────────────────────────────────────────── tidyverse_conflicts() ──
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library('ISLR')
library('cluster')
library('factoextra')
```

```
## Warning: package 'factoextra' was built under R version 4.3.2
```

```
## 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 4.3.2
```

```
library('ggplot2')
library('gridExtra')
```

```
## Warning: package 'gridExtra' was built under R version 4.3.2
```

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

```
# Import the DATASET
Fuel.data.actual <- read.csv("D:/KSU SEM-1/FUNDAMENTAL OF MACHINE LEARNING/Dataset/fuel_receipts1.csv")
```

```
# Calculate the column means of logical values
colMeans(is.na(Fuel.data.actual))
```

```
##                   X               rowid        plant_id_eia  energy_source_code
##                   0                   0                   0                   0
## fuel_type_code_pudl     fuel_group_code       supplier_name fuel_received_units
##                   0                   0                   0                   0
## fuel_mmbtu_per_unit  sulfur_content_pct     ash_content_pct fuel_cost_per_mmbtu
##                   0                   0                   0                   0
```

Interpretation : The percentage of missing values in each dataset column is evaluated with the aid of this operation. A vector of mean values, each of which reflects the percentage of missing values in the relevant column, will be provided as the output.

```
Fuel.data.numeric <- Fuel.data.actual[ ,-c(1,2,3,4,5,6,7)]
Fuel.data.numeric
```

This stage helps to separate the quantitative components of the data from perhaps category or non-numeric factors and is frequently carried out when concentrating on numerical features for additional analysis or modeling. This part

assigns the resulting subset to a new data frame named Fuel.data.numeric

```
Fuel.data.scaled <- scale(Fuel.data.numeric)
```

```
# K-means clustering on a scaled numeric data set
set.seed(2)
k <- 3
mod.kmeans <- kmeans(Fuel.data.scaled, k)
mod.kmeans
```

```
## K-means clustering with 3 clusters of sizes 334, 91, 331
##
## Cluster means:
##   fuel_received_units fuel_mmbtu_per_unit sulfur_content_pct ash_content_pct
## 1           0.1736253          -1.0841387        -0.66090643      -0.9756003
## 2          -0.1378751           0.9802038         2.30196591       0.8048236
## 3          -0.1372937           0.8244827         0.03402975       0.7631769
##   fuel_cost_per_mmbtu
## 1           0.8649276
## 2          -0.7312936
## 3          -0.6717163
##
## Clustering vector:
##   [1] 3 3 1 3 3 3 1 2 3 3 3 3 3 1 1 3 3 3 3 3 3 1 3 3 3 3 3 3 3 3 3 3 3 3 3 3
##  [38] 3 3 1 1 3 3 3 3 3 3 2 3 3 2 3 3 3 2 1 1 1 1 1 3 3 1 1 3 3 3 3 3 3 3 1 1 1
##  [75] 3 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 3 3 3 3 1 3 3 3 1 1 1 1 1 1 1 3
## [112] 3 2 3 3 3 2 3 3 3 3 2 2 2 2 2 1 1 1 1 1 1 1 1 1 1 1 1 3 1 3 3 3 3 1 1 3 3 3
## [149] 1 1 1 3 3 1 1 1 1 1 1 1 1 1 1 1 3 3 3 1 3 3 3 3 3 1 1 3 3 3 3 3 1 3 1 3 3 1
## [186] 1 3 1 3 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
## [223] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
## [260] 1 1 1 1 3 3 3 3 3 3 1 1 1 1 1 1 1 1 1 1 1 3 3 3 3 3 3 3 3 3 3 3 3 1 1 1
## [297] 1 1 3 3 1 3 3 3 3 1 3 3 1 1 1 3 1 1 1 1 1 3 1 2 2 2 2 1 1 1 1 1 1 1 1 3 3
## [334] 3 3 3 3 1 1 1 1 1 1 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 1 3 3 3 3 3 3 1 3 3
## [371] 3 3 3 1 3 3 3 3 3 3 3 3 3 3 3 1 1 1 1 2 2 1 2 1 1 1 2 2 2 2 2 2 2 2 2 1 1 3
## [408] 3 3 3 3 1 1 3 3 2 2 1 1 3 1 2 2 2 2 2 2 1 2 1 2 1 3 3 3 2 2 1 3 2 2 2 3 3
## [445] 1 1 2 1 1 1 1 1 1 3 1 3 2 2 2 2 3 3 3 3 2 3 2 1 2 3 1 3 1 1 1 3 1 1 1 3 3 1
## [482] 2 3 3 3 1 1 3 3 3 3 3 3 3 3 3 1 1 1 3 3 3 3 3 3 3 3 3 1 1 3 3 3 3 3 3 1 1
## [519] 3 3 3 3 1 3 1 2 3 1 1 1 3 3 1 3 3 2 1 1 1 1 1 1 1 2 3 1 1 2 2 1 1 3 3 3 3
## [556] 3 3 3 3 3 3 3 3 1 1 1 3 1 3 1 1 1 1 3 3 1 1 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
## [593] 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 1 3 3 3 3 3 1 1 1 3 3 3 3 2 2 3 3 3 2 2 2
## [630] 3 3 2 2 2 1 1 2 2 2 2 2 1 1 1 1 3 3 2 2 1 2 2 2 2 2 2 2 2 1 1 2 2 2 2 2 2
## [667] 2 2 2 1 2 3 3 3 3 3 3 2 1 1 1 3 3 3 3 3 3 3 3 3 1 1 3 3 3 1 1 1 3 1 1 1 1 2
## [704] 3 3 3 3 3 3 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
## [741] 1 1 3 3 3 3 3 3 3 3 3 3 3 3 3 3
##
## Within cluster sum of squares by cluster:
## [1] 1048.1921  106.3856  251.3804
##  (between_SS / total_SS =  62.8 %)
##
## Available components:
##
## [1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss"
## [6] "betweenss"    "size"         "iter"         "ifault"
```

Interpretation : Given that the k variable is set to 3, the algorithm should produce three clusters inside the dataset. Next, the scaled numerical data (Fuel.data.scaled) is subjected to k-means clustering using the kmeans() method. The outcome is kept in the mod.kmeans object. The outcomes of the clustering process are disclosed in the mod.kmeans output. The coordinates of the cluster centroids in the scaled feature space (mod.kmeans $centers) and the cluster assignments for each observation i cluster) are the essential elements. The total within-cluster sum of squares (mod.kmeans$tot.withinss), a gauge of the clusters' compactness, is also included in the output.

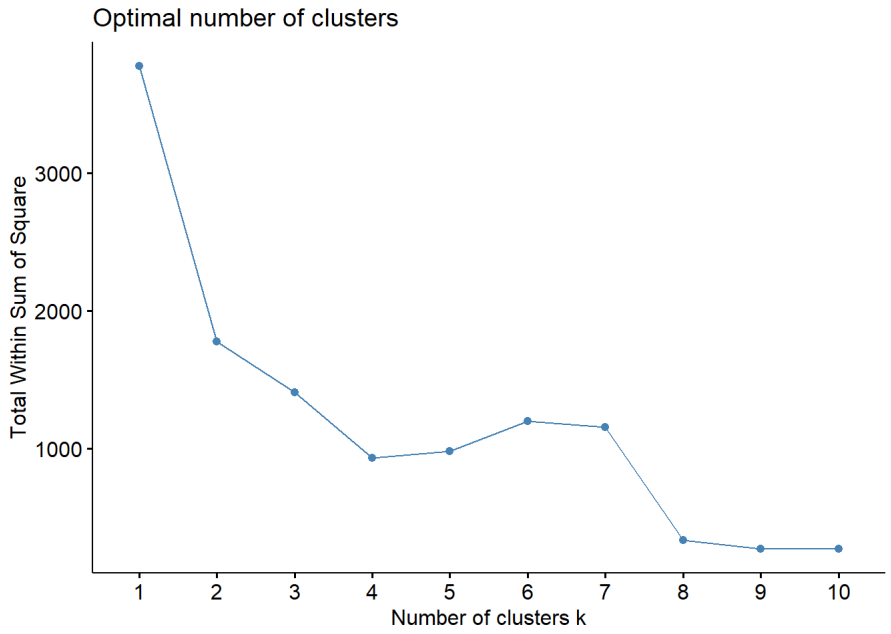
```
cluster_assignments <- mod.kmeans$cluster
cluster_means <- aggregate(Fuel.data.scaled, by = list(cluster_assignments), FUN = mean)
head(cluster_means)
```

```
##   Group.1 fuel_received_units fuel_mmbtu_per_unit sulfur_content_pct
## 1       1           0.1736253          -1.0841387        -0.66090643
## 2       2          -0.1378751           0.9802038         2.30196591
## 3       3          -0.1372937           0.8244827         0.03402975
##   ash_content_pct fuel_cost_per_mmbtu
## 1      -0.9756003           0.8649276
## 2       0.8048236          -0.7312936
## 3       0.7631769          -0.6717163
```

Lastly, the first few rows of the generated data frame are shown using the head(cluster_means) function, giving an overview of the scaled feature mean values for each cluster. Analyzing the average values of each variable inside each cluster is necessary to interpret the results. This data provides insights into the scaled numeric feature average profile of observations within each cluster. For example, if the mean values of some variables are greater in a given cluster, it implies that the observations in that cluster typically have higher values for those specific features. This type of study aids in identifying and comprehending the unique patterns that each cluster exhibits, offering important details for additional research or cluster-based decision-making.

Elbow

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



This graphic is meant to aid in figuring out how many clusters would be best for the dataset. The figure aids in locating a “elbow” point by showing the WSS values for various values of k. The elbow point is the location on the plot where the WSS is not substantially reduced by adding more clusters. The theory is that adding clusters significantly improves fit before to the elbow, while the marginal improvement decreases beyond the elbow. We can determine k to be 4 and Finding the location where the WSS starts to level out and take on the shape of an elbow is necessary to interpret the plot. This point's recommended ideal number of clusters is frequently linked to it. It's crucial to remember that elbow identification is somewhat arbitrary, and in situations when an elbow is difficult to identify, further techniques or domain expertise could be needed in order to determine the right number of clusters. In the context of clustering analysis, the visualization is a useful diagnostic tool that helps choose an appropriate value for k.

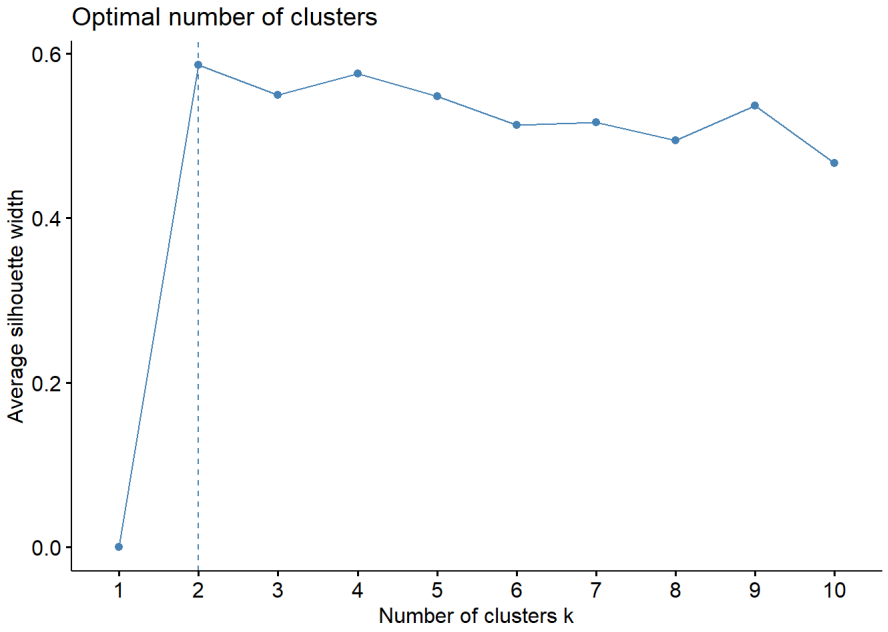
```
set.seed(121)
k <- 4
mod.kmeanew <- kmeans(Fuel.data.scaled, k)
mod.kmeanew
```

```
## K-means clustering with 4 clusters of sizes 67, 422, 261, 6
##
## Cluster means:
##   fuel_received_units fuel_mmbtu_per_unit sulfur_content_pct ash_content_pct
## 1         -0.22527524          -0.7663744         -0.5734317      -0.9756003
## 2         -0.13741908           0.8580624          0.5230871       0.7721576
## 3          0.05274269          -1.1635620         -0.6824909      -0.9756003
## 4          9.88640796          -1.1775916         -0.6987812      -0.9756003
##   fuel_cost_per_mmbtu
## 1           2.6109757
## 2          -0.6845636
## 3           0.4230302
## 4           0.5899257
##
## Clustering vector:
##   [1] 2 2 4 2 2 2 3 2 2 2 2 2 2 1 3 2 2 2 2 2 2 3 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
##  [38] 2 2 3 3 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 1 1 1 1 2 2 3 3 2 2 2 2 2 2 2 1 3 3
##  [75] 2 3 3 3 3 3 1 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 2 2 2 2 3 2 2 2 3 3 3 3 3 3 3 2
## [112] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 1 3 1 3 3 3 3 3 3 3 3 2 3 2 2 2 3 3 2 2 2
## [149] 1 3 3 2 2 1 3 1 3 3 3 4 3 3 3 3 3 2 2 2 3 2 2 2 2 3 3 2 2 2 2 3 2 1 2 2 3
## [186] 3 2 3 2 1 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 1 3
## [223] 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
## [260] 3 3 3 3 2 2 2 2 2 2 1 3 3 3 3 3 4 4 3 3 4 4 3 2 2 2 2 2 2 2 2 2 2 2 3 3 3
## [297] 3 3 2 2 3 2 2 2 2 1 2 2 1 3 3 2 3 3 3 3 3 2 3 2 2 2 2 3 3 3 3 3 1 3 3 2 2
## [334] 2 2 2 2 3 3 3 3 3 3 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 1 2 2
## [371] 2 2 2 1 2 2 2 2 2 2 2 2 2 2 1 1 3 3 2 2 1 2 1 3 3 2 2 2 2 2 2 2 2 2 1 1 2
## [408] 2 2 2 2 1 1 2 2 2 2 1 3 2 1 2 2 2 2 2 2 1 2 3 2 3 2 2 2 2 2 3 2 2 2 2 2 2
## [445] 1 3 2 3 3 3 3 3 3 2 3 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 1 2 3 3 2 3 3 3 2 2 3
## [482] 2 2 2 2 1 3 2 2 2 2 2 2 2 2 2 1 1 3 2 2 2 2 2 2 2 2 2 3 3 2 2 2 2 2 2 1 3
## [519] 2 2 2 2 1 2 3 2 2 3 3 3 2 2 3 2 2 2 3 3 3 3 3 3 3 2 2 3 3 2 2 1 1 2 2 2 2
## [556] 2 2 2 2 2 2 2 3 3 3 2 3 2 3 3 3 3 2 2 3 3 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
## [593] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 1 1 3 2 2 2 2 2 2 2 2 2 2 2 2
## [630] 2 2 2 2 2 2 1 1 2 2 2 2 2 1 1 1 3 2 2 2 2 3 2 2 2 2 2 2 2 3 3 2 2 2 2 2 2
## [667] 2 2 2 1 2 2 2 2 2 2 2 2 1 1 1 2 2 2 2 2 2 2 2 2 2 1 1 2 2 2 1 1 1 2 3 3 3 3 2
## [704] 2 2 2 2 2 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 1 3 1 3 1 3 3 3 1
## [741] 3 1 2 2 2 2 2 2 2 2 2 2 2 2 2
##
## Within cluster sum of squares by cluster:
## [1]  15.75483 727.00285 165.08684  22.05764
##  (between_SS / total_SS =  75.4 %)
##
## Available components:
##
## [1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss"
## [6] "betweenss"    "size"         "iter"         "ifault"
```

In order to interpret this result, one must comprehend the composition of the new clusters and the way in which the scaled numeric attributes are used to group observations together. The effects of the number of clusters on clustering patterns and within-cluster variability can be understood by comparing the findings with different values of k (in this case, compared with the prior model where k=3). In k-means clustering analysis, this iterative procedure is typical when many values of k are tried in order to identify an appropriate configuration that most accurately captures the underlying structure of the data.

Silhouette

```
fviz_nbclust(Fuel.data.scaled, kmeans, method = "silhouette")
```



A statistic known as the silhouette

approach evaluates how well each data point fits into the designated cluster. A higher number on the silhouette width scale, which goes from -1 to 1, denotes more clearly delineated clusters. When a data point's silhouette width is near to 1, it indicates that it matches its own cluster well and its nearby clusters poorly. Finding the location where the average silhouette width is maximized is a crucial step in interpreting the plot. For every value of k, the average silhouette width is computed, and the plot assists in determining the number of clusters that produce distinct and well-organized clusters.

```
set.seed(2)
k <- 2
mod.kmeans_sil <- kmeans(Fuel.data.scaled, k)
mod.kmeans_sil
```

```
## K-means clustering with 2 clusters of sizes 334, 422
##
## Cluster means:
##   fuel_received_units fuel_mmbtu_per_unit sulfur_content_pct ash_content_pct
## 1           0.1736253          -1.0841387         -0.6609064      -0.9756003
## 2          -0.1374191           0.8580624          0.5230871       0.7721576
##   fuel_cost_per_mmbtu
## 1           0.8649276
## 2          -0.6845636
##
## Clustering vector:
##   [1] 2 2 1 2 2 2 1 2 2 2 2 2 2 1 1 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2
##  [38] 2 2 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 1 1 1 1 2 2 1 1 2 2 2 2 2 2 1 1 1
##  [75] 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 1 2 2 2 1 1 1 1 1 1 1 2
## [112] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 1 1 1 1 1 1 1 1 1 1 2 1 2 2 2 1 1 2 2 2
## [149] 1 1 1 2 2 1 1 1 1 1 1 1 1 1 1 1 2 2 2 1 2 2 2 2 1 1 2 2 2 2 1 2 1 2 2 1
## [186] 1 2 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
## [223] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
## [260] 1 1 1 1 2 2 2 2 2 2 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 1 1 1
## [297] 1 1 2 2 1 2 2 2 2 1 2 2 1 1 1 2 1 1 1 1 2 1 2 2 2 2 1 1 1 1 1 1 1 2 2 2
## [334] 2 2 2 2 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 1 2 2
## [371] 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 1 1 1 1 2 2 1 2 1 1 1 2 2 2 2 2 2 2 2 1 1 2
## [408] 2 2 2 2 1 1 2 2 2 2 1 1 2 1 2 2 2 2 2 2 1 2 1 2 1 2 2 2 2 2 1 2 2 2 2 2 2
## [445] 1 1 2 1 1 1 1 1 1 2 1 2 2 2 2 2 2 2 2 2 2 2 1 2 2 1 2 1 1 2 1 1 1 2 2 1
## [482] 2 2 2 2 1 1 2 2 2 2 2 2 2 2 2 1 1 1 2 2 2 2 2 2 2 2 2 1 1 2 2 2 2 2 2 1 1
## [519] 2 2 2 2 1 2 1 2 2 1 1 1 2 2 1 2 2 2 1 1 1 1 1 1 1 2 2 1 1 2 2 1 1 2 2 2 2
## [556] 2 2 2 2 2 2 2 1 1 1 2 1 2 1 1 1 1 2 2 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
## [593] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2
## [630] 2 2 2 2 2 2 1 1 2 2 2 2 2 1 1 1 1 2 2 2 2 1 2 2 2 2 2 2 2 2 1 1 2 2 2 2 2
## [667] 2 2 2 1 2 2 2 2 2 2 2 2 1 1 1 2 2 2 2 2 2 2 2 2 1 1 2 2 2 1 1 1 2 1 1 1 1
## [704] 2 2 2 2 2 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
## [741] 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2
##
## Within cluster sum of squares by cluster:
## [1] 1048.1921  727.0029
##  (between_SS / total_SS =  53.0 %)
##
## Available components:
##
## [1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss"
## [6] "betweenss"    "size"         "iter"         "ifault"
```

This output includes an understanding of the new clusters' structure and the way observations are classified according to how similar they are to each other in terms of the scaled numerical attributes. It is possible to analyze the data to find patterns and traits within the two clusters that are suggested by the selection of k=2. This procedure is essential to evaluating the effects of various cluster number selections on the data's clustering structure.

Comparing all the numeric columns according to mean value

```
cluster.silhouette.assignment <- mod.kmeans_sil$cluster
cluster.means.silhouette <- aggregate(Fuel.data.scaled, by = list(cluster.silhouette.assignment), FUN = mean)
```