K Means Clustering - Auto Data

V2 Maestros

Contents

Problem Statement

The input data contains samples of cars and technical / price information about them. The goal of this problem is to group these cars into 4 clusters based on their attributes

Techniques Used

1. K-Means Clustering
2. Centering and Scaling

Data Engineering & Analysis

```
setwd("C:/Personal/V2Maestros/Modules/Machine Learning Algorithms/Clustering")

auto_data <- read.csv("auto-data.csv")

str(auto_data)
```

Loading and understanding the dataset

```
## 'data.frame':    197 obs. of  12 variables:
##  $ MAKE     : Factor w/ 21 levels "alfa-romero",..: 18 4 9 19 12 6 13 5 15 9 ...
##  $ FUELTYPE : Factor w/ 2 levels "diesel","gas": 2 2 2 2 2 2 2 2 2 2 ...
##  $ ASPIRE   : Factor w/ 2 levels "std","turbo": 1 1 1 1 1 1 1 1 1 1 ...
##  $ DOORS    : Factor w/ 2 levels "four","two": 2 2 2 2 2 2 2 2 2 2 ...
##  $ BODY     : Factor w/ 5 levels "convertible",..: 3 3 3 3 3 3 4 3 3 3 ...
```

```
##  $ DRIVE    : Factor w/ 3 levels "4wd","fwd","rwd": 2 2 2 2 2 2 2 2 2 2 ...
##  $ CYLINDERS: Factor w/ 7 levels "eight","five",..: 3 5 3 3 3 3 3 3 3 3 ...
##  $ HP       : int  69 48 68 62 68 60 69 68 68 68 ...
##  $ RPM      : int  4900 5100 5000 4800 5500 5500 5200 5500 5500 5000 ...
##  $ MPG.CITY : int  31 47 30 35 37 38 31 37 37 31 ...
##  $ MPG.HWY  : int  36 53 31 39 41 42 37 41 41 38 ...
##  $ PRICE    : int  5118 5151 5195 5348 5389 5399 5499 5572 5572 6095 ...
```

```
summary(auto_data)
```

```
##          MAKE      FUELTYPE     ASPIRE     DOORS              BODY
##  toyota    :32   diesel: 19   std  :162   four:112   convertible: 6
##  nissan    :18   gas   :178   turbo: 35   two : 85   hardtop    : 8
##  mazda     :16                                       hatchback  :67
##  honda     :13                                       sedan      :92
##  mitsubishi:13                                       wagon      :24
##  subaru    :12
##  (Other)   :93
##  DRIVE      CYLINDERS         HP            RPM          MPG.CITY
##  4wd:  8   eight :  4   Min.   : 48   Min.   :4150   Min.   :13.0
##  fwd:114   five  : 10   1st Qu.: 70   1st Qu.:4800   1st Qu.:19.0
##  rwd: 75   four  :153   Median : 95   Median :5200   Median :24.0
##            six   : 24   Mean   :104   Mean   :5118   Mean   :25.1
##            three :  1   3rd Qu.:116   3rd Qu.:5500   3rd Qu.:30.0
##            twelve:  1   Max.   :262   Max.   :6600   Max.   :49.0
##            two   :  4
##     MPG.HWY         PRICE
##  Min.   :16.0   Min.   : 5118
##  1st Qu.:25.0   1st Qu.: 7775
##  Median :30.0   Median :10345
##  Mean   :30.6   Mean   :13280
##  3rd Qu.:34.0   3rd Qu.:16503
##  Max.   :54.0   Max.   :45400
##
```

```
head(auto_data)
```

```
##         MAKE FUELTYPE ASPIRE DOORS      BODY DRIVE CYLINDERS HP  RPM
## 1     subaru      gas    std   two hatchback   fwd      four 69 4900
## 2  chevrolet      gas    std   two hatchback   fwd     three 48 5100
## 3      mazda      gas    std   two hatchback   fwd      four 68 5000
## 4     toyota      gas    std   two hatchback   fwd      four 62 4800
## 5 mitsubishi      gas    std   two hatchback   fwd      four 68 5500
## 6      honda      gas    std   two hatchback   fwd      four 60 5500
##   MPG.CITY MPG.HWY PRICE
## 1       31      36  5118
## 2       47      53  5151
## 3       30      31  5195
## 4       35      39  5348
## 5       37      41  5389
## 6       38      42  5399
```

Data Cleansing

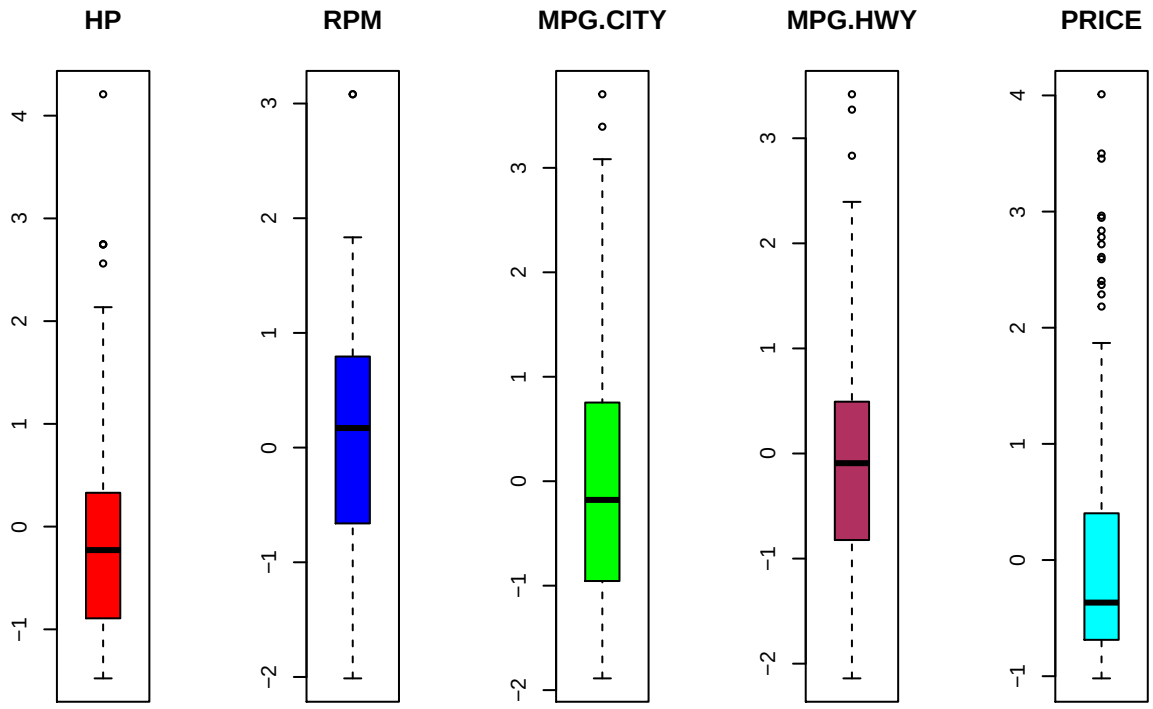
1. The ranges of values in each of the variables (columns) look ok without any kind of outliers
2. Clustering needs all numeric values to be in the same range. Hence we need to center and scale data set

```
scaled_num <- scale( auto_data[8:12])
#put the attributes back into the main data frame
auto_data[,8:12] <- scaled_num
summary(auto_data)
```

```
##          MAKE      FUELTYPE     ASPIRE     DOORS              BODY
##  toyota    :32   diesel: 19   std  :162   four:112   convertible: 6
##  nissan    :18   gas   :178   turbo: 35   two : 85   hardtop    : 8
##  mazda     :16                                       hatchback  :67
##  honda     :13                                       sedan      :92
##  mitsubishi:13                                       wagon      :24
##  subaru    :12
##  (Other)   :93
##  DRIVE      CYLINDERS         HP               RPM             MPG.CITY
##  4wd:  8   eight :  4   Min.   :-1.477   Min.   :-2.012   Min.   :-1.888
##  fwd:114   five  : 10   1st Qu.:-0.893   1st Qu.:-0.661   1st Qu.:-0.956
##  rwd: 75   four  :153   Median :-0.229   Median : 0.170   Median :-0.179
##            six   : 24   Mean   : 0.000   Mean   : 0.000   Mean   : 0.000
##            three :  1   3rd Qu.: 0.329   3rd Qu.: 0.794   3rd Qu.: 0.753
##            twelve:  1   Max.   : 4.208   Max.   : 3.081   Max.   : 3.704
##            two   :  4
##     MPG.HWY           PRICE
##  Min.   :-2.140   Min.   :-1.019
##  1st Qu.:-0.823   1st Qu.:-0.687
##  Median :-0.092   Median :-0.366
##  Mean   : 0.000   Mean   : 0.000
##  3rd Qu.: 0.493   3rd Qu.: 0.402
##  Max.   : 3.419   Max.   : 4.010
##
```

Exploratory Data Analysis Typically, for Clustering problems, EDA is only required for finding out outliers and errors. If outliers are found, we would want to eliminate them since they might skew the clusters formed by moving the centeroids signficantly.

```
par(mfrow=c(1,5))

boxplot( auto_data$HP,col="red")
title("HP")

boxplot( auto_data$RPM,col="blue")
title("RPM")

boxplot( auto_data$MPG.CITY,col="green")
title("MPG.CITY")

boxplot( auto_data$MPG.HWY, col="maroon")
title("MPG.HWY")

boxplot( auto_data$PRICE, col="cyan")
title("PRICE")
```



We choose not to remove the outliers (dots in the charts) since they are many (hence may not be outliers !).

Modeling & Prediction

Build Clusters for 2 variables In order to demonstrate the clusters being formed on a 2-dimensional plot, we will only use 100 samples and 2 attributes - HP and PRICE to create 4 clusters.

```
library(class)
```

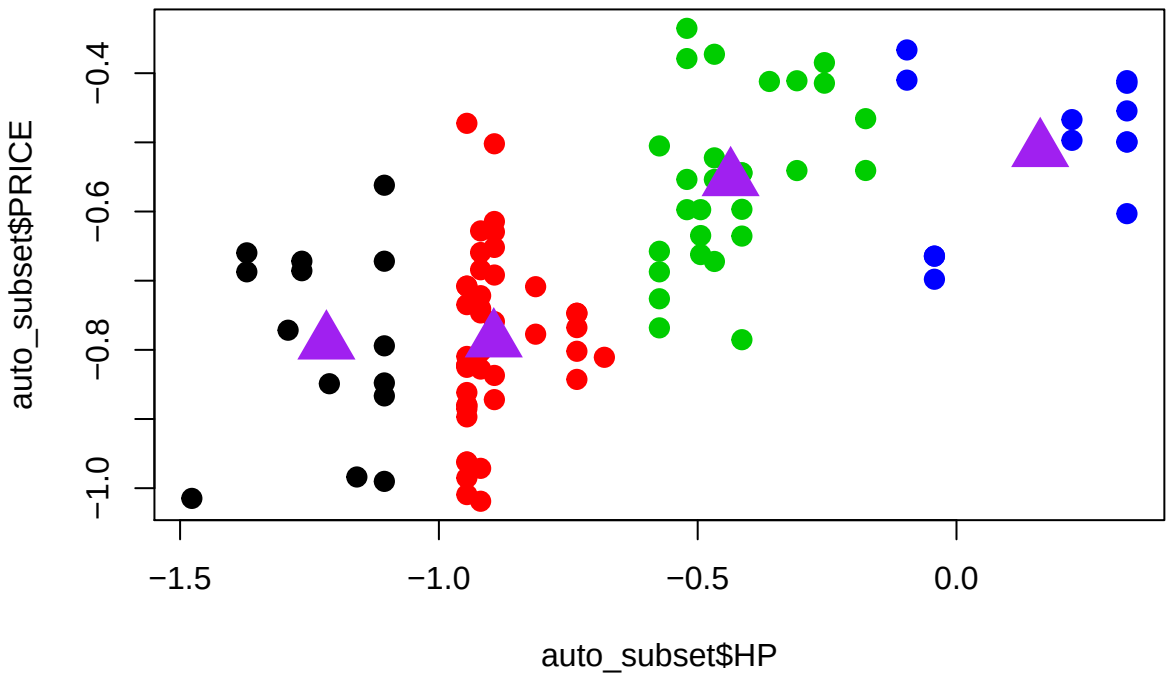
```
## Warning: package 'class' was built under R version 3.1.1
```

```
#keep the same seed for each execution. Seed impacts the initial centroid position and hence may impact
#actual clusters formed.

set.seed(11111)
auto_subset <- auto_data[1:100, c(8,12) ]
clusters<- kmeans(auto_subset,4)
clusters
```

```
## K-means clustering with 4 clusters of sizes 14, 45, 28, 13
##
## Cluster means:
##        HP   PRICE
## 1 -1.2173 -0.7897
```

```
## 2 -0.8940 -0.7867
## 3 -0.4364 -0.5535
## 4  0.1618 -0.5115
##
## Clustering vector:
##   1   2   3   4   5   6   7   8   9  10  11  12  13  14  15  16  17  18
##   2   1   2   1   2   1   2   2   2   2   2   2   2   2   1   2   1   1
##  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36
##   2   2   2   2   2   2   2   2   2   2   2   1   2   3   2   1   3   2
##  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54
##   2   2   2   2   2   2   3   2   2   2   2   4   2   3   1   1   2   3
##  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72
##   1   1   4   4   3   1   2   3   2   3   3   2   2   2   4   3   3   3
##  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89  90
##   1   3   3   3   3   3   3   3   3   2   4   4   4   2   4   3   4   4
##  91  92  93  94  95  96  97  98  99 100
##   3   3   3   4   4   3   3   3   4   3
##
## Within cluster sum of squares by cluster:
## [1] 0.4631 0.8946 0.7937 0.5805
##  (between_SS / total_SS =  87.1 %)
##
## Available components:
##
## [1] "cluster"      "centers"      "totss"        "withinss"
## [5] "tot.withinss" "betweenss"    "size"         "iter"
## [9] "ifault"
```

```
par(mfrow=c(1,1))
plot(auto_subset$HP, auto_subset$PRICE, col=clusters$cluster, pch=20, cex=2)
points(clusters$centers, col="purple", pch=17, cex=3)
```




```
#First convert all factors to numeric
for ( i in 1:8) {
  auto_data[, i ] = as.numeric(auto_data[,i])
}
summary(auto_data)
```

Clustering for all Data

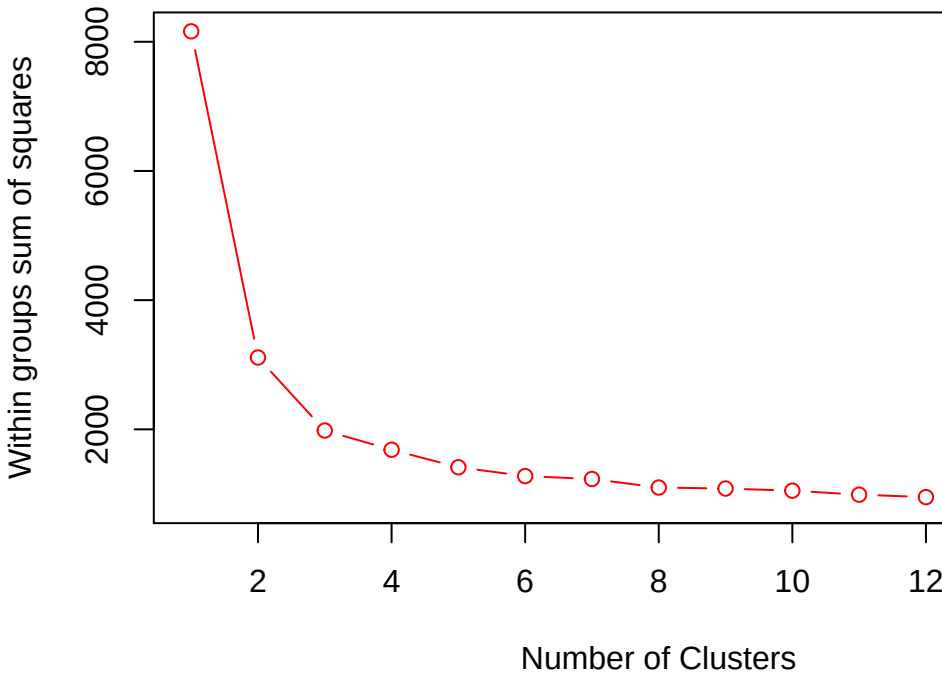
```
##       MAKE       FUELTYPE       ASPIRE         DOORS           BODY
##  Min.   : 1   Min.   :1.0   Min.   :1.00   Min.   :1.00   Min.   :1.00
##  1st Qu.: 9   1st Qu.:2.0   1st Qu.:1.00   1st Qu.:1.00   1st Qu.:3.00
##  Median :13   Median :2.0   Median :1.00   Median :1.00   Median :4.00
##  Mean   :13   Mean   :1.9   Mean   :1.18   Mean   :1.43   Mean   :3.61
##  3rd Qu.:19   3rd Qu.:2.0   3rd Qu.:1.00   3rd Qu.:2.00   3rd Qu.:4.00
##  Max.   :21   Max.   :2.0   Max.   :2.00   Max.   :2.00   Max.   :5.00
##      DRIVE        CYLINDERS          HP               RPM
##  Min.   :1.00   Min.   :1.00   Min.   :-1.477   Min.   :-2.012
##  1st Qu.:2.00   1st Qu.:3.00   1st Qu.:-0.893   1st Qu.:-0.661
##  Median :2.00   Median :3.00   Median :-0.229   Median : 0.170
##  Mean   :2.34   Mean   :3.14   Mean   : 0.000   Mean   : 0.000
##  3rd Qu.:3.00   3rd Qu.:3.00   3rd Qu.: 0.329   3rd Qu.: 0.794
##  Max.   :3.00   Max.   :7.00   Max.   : 4.208   Max.   : 3.081
##     MPG.CITY         MPG.HWY           PRICE
```

```
##  Min.   :-1.888   Min.   :-2.140   Min.   :-1.019
##  1st Qu.:-0.956   1st Qu.:-0.823   1st Qu.:-0.687
##  Median :-0.179   Median :-0.092   Median :-0.366
##  Mean   : 0.000   Mean   : 0.000   Mean   : 0.000
##  3rd Qu.: 0.753   3rd Qu.: 0.493   3rd Qu.: 0.402
##  Max.   : 3.704   Max.   : 3.419   Max.   : 4.010
```

```
set.seed(11111)
clusters<- kmeans(auto_data[,7:12],4)
clusters
```

```
## K-means clustering with 4 clusters of sizes 58, 17, 53, 69
##
## Cluster means:
##   CYLINDERS      HP     RPM MPG.CITY  MPG.HWY   PRICE
## 1     3.034 -0.9157  0.2618   1.1547  1.16138 -0.7813
## 2     3.059  1.9797 -0.2270  -1.3120 -1.38277  2.5989
## 3     3.057 -0.3193 -0.9494   0.1024  0.04592 -0.2110
## 4     3.304  0.5272  0.5651  -0.7260 -0.67083  0.1785
##
## Clustering vector:
##   [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 3 1 1 1 1 1 3 3 1 1
##  [36] 1 1 1 1 1 1 1 3 1 3 1 1 4 1 3 1 1 1 1 3 1 4 4 1 1 1 3 3 3 1 3 1 3 3 3
##  [71] 1 3 3 1 3 3 3 3 3 1 3 3 4 4 4 1 4 3 3 4 3 4 3 3 4 3 3 1 4 3 3 3 4 3 3
## [106] 3 3 3 3 4 3 4 4 4 4 4 4 4 4 4 4 4 4 3 4 4 4 4 4 4 3 3 4 4 4 4 4 4 4 4 4
## [141] 4 4 4 4 4 4 4 4 4 4 4 4 4 3 4 3 4 4 3 4 3 4 4 4 3 4 4 4 4 4 4 2 3 3 4
## [176] 4 3 4 4 3 3 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
##
## Within cluster sum of squares by cluster:
## [1] 106.38  88.46  75.83 174.93
##  (between_SS / total_SS =  59.6 %)
##
## Available components:
##
## [1] "cluster"      "centers"      "totss"        "withinss"
## [5] "tot.withinss" "betweenss"    "size"         "iter"
## [9] "ifault"
```

```
#finding the optimum no. of clusters
wssplot <- function(data, nc=15, seed=1234){
  wss <- (nrow(data)-1)*sum(apply(data,2,var))
  for (i in 2:nc){
    set.seed(seed)
    wss[i] <- sum(kmeans(data, centers=i)$withinss)}
  plot(1:nc, wss, type="b", xlab="Number of Clusters",
       ylab="Within groups sum of squares", col="red")}

wssplot(auto_data)
```



Finding optimal number of Clusters

3 seems to be the optimal number of clusters for this dataset

Conclusions

K-means clustering is a fast and easy way to group data based on similarities in data

---