# 2. CLUSTERING _ Thi Tinh Lo (22236226)

2023-08-19

**Load library**

**CLUSTERING:**

**1. PCA**

**2. PCA & Kmeans**

**3. PCA & HIERARCHICAL**

**4. PCA & DBSCAN**

**Read data**

```
setwd('C:/Users/tinhl/OneDrive/Documents')
data <- read.csv(file ='data_processed.csv')
head(data)
```

```
##   State Response Coverage Education EmploymentStatus Gender Location_Code
## 1     5        0        3         3                2      1             2
## 2     1        0        2         3                5      1             2
## 3     3        0        1         3                2      1             2
## 4     2        0        3         3                5      2             2
## 5     5        0        3         3                2      2             3
## 6     4        1        3         3                2      1             3
##   Marital_Status Policy_Type Policy Renew_Offer_Type Sales_Channel
## 1              2           2      6                1             1
## 2              3           3      9                3             1
## 3              2           3      9                1             1
## 4              2           2      5                1             3
## 5              3           3      7                1             1
## 6              2           3      9                2             4
##   Vehicle_Class Vehicle_Size Customer_Lifetime_Value     Income
## 1             6            2             -0.76283596  0.6127939
## 2             1            2             -0.14923729 -1.2395490
## 3             6            2              0.71059732  0.3656898
## 4             5            2             -0.05226027 -1.2395490
## 5             1            2             -0.75553375  0.2033785
## 6             6            2              0.03658252  0.8309644
##   Monthly_Premium_Auto Months_Since_Last_Claim Months_Since_Policy_Inception
## 1          -0.70388612               1.6780075                   -1.5432025
```

```
## 2           0.02268979              -0.2081750                    -0.2173223
## 3           0.42957229               0.2881888                    -0.3606607
## 4           0.37144622               0.2881888                     0.6068735
## 5          -0.58763397              -0.3074478                    -0.1456531
## 6          -0.70388612              -0.1089022                     1.6460769
##   Number_of_Open_Complaints Number_of_Policies Total_Claim_Amount
## 1                -0.4222264         -0.8226028         -0.1696304
## 2                -0.4222264          2.1060447          2.4006056
## 3                -0.4222264         -0.4042246          0.4557088
## 4                -0.4222264          1.6876664          0.3297505
## 5                -0.4222264         -0.8226028         -1.0187877
## 6                -0.4222264         -0.4042246         -0.9456305
```

```
summary(data)
```

```
##      State          Response         Coverage       Education
##  Min.   :1.000   Min.   :0.0000   Min.   :1.000   Min.   :1.000
##  1st Qu.:2.000   1st Qu.:0.0000   1st Qu.:2.000   1st Qu.:3.000
##  Median :2.000   Median :0.0000   Median :3.000   Median :4.000
##  Mean   :2.742   Mean   :0.1432   Mean   :2.519   Mean   :3.712
##  3rd Qu.:4.000   3rd Qu.:0.0000   3rd Qu.:3.000   3rd Qu.:5.000
##  Max.   :5.000   Max.   :1.0000   Max.   :3.000   Max.   :5.000
##  EmploymentStatus     Gender     Location_Code   Marital_Status  Policy_Type
##  Min.   :1.000    Min.   :1.00   Min.   :1.000   Min.   :1.00   Min.   :1.000
##  1st Qu.:2.000    1st Qu.:1.00   1st Qu.:2.000   1st Qu.:2.00   1st Qu.:2.000
##  Median :2.000    Median :1.00   Median :2.000   Median :2.00   Median :3.000
##  Mean   :2.826    Mean   :1.49   Mean   :2.021   Mean   :2.12   Mean   :2.702
##  3rd Qu.:5.000    3rd Qu.:2.00   3rd Qu.:2.000   3rd Qu.:3.00   3rd Qu.:3.000
##  Max.   :5.000    Max.   :2.00   Max.   :3.000   Max.   :3.00   Max.   :3.000
##      Policy      Renew_Offer_Type Sales_Channel   Vehicle_Class
##  Min.   :1.000   Min.   :1.00     Min.   :1.000   Min.   :1.000
##  1st Qu.:6.000   1st Qu.:1.00     1st Qu.:1.000   1st Qu.:1.000
##  Median :8.000   Median :2.00     Median :2.000   Median :1.000
##  Mean   :7.425   Mean   :1.97     Mean   :2.103   Mean   :3.036
##  3rd Qu.:9.000   3rd Qu.:3.00     3rd Qu.:3.000   3rd Qu.:5.000
##  Max.   :9.000   Max.   :4.00     Max.   :4.000   Max.   :6.000
##   Vehicle_Size  Customer_Lifetime_Value     Income        Monthly_Premium_Auto
##  Min.   :1.00   Min.   :-0.8888         Min.   :-1.2395   Min.   :-0.9364
##  1st Qu.:2.00   1st Qu.:-0.5837         1st Qu.:-1.2395   1st Qu.:-0.7329
##  Median :2.00   Median :-0.3238         Median :-0.1240   Median :-0.2970
##  Mean   :1.91   Mean   : 0.0000         Mean   : 0.0000   Mean   : 0.0000
##  3rd Qu.:2.00   3rd Qu.: 0.1393         3rd Qu.: 0.8118   3rd Qu.: 0.4586
##  Max.   :3.00   Max.   :10.9621         Max.   : 2.0515   Max.   : 5.9515
##  Months_Since_Last_Claim Months_Since_Policy_Inception
##  Min.   :-1.4987         Min.   :-1.722376
##  1st Qu.:-0.9031         1st Qu.:-0.862345
##  Median :-0.1089         Median :-0.002315
##  Mean   : 0.0000         Mean   : 0.000000
##  3rd Qu.: 0.7846         3rd Qu.: 0.821881
##  Max.   : 1.9758         Max.   : 1.825250
##  Number_of_Open_Complaints Number_of_Policies Total_Claim_Amount
##  Min.   :-0.4222           Min.   :-0.8226    Min.   :-1.4939
##  1st Qu.:-0.4222           1st Qu.:-0.8226    1st Qu.:-0.5571
##  Median :-0.4222           Median :-0.4042    Median :-0.1726
```

```
##  Mean   : 0.0000           Mean   : 0.0000    Mean   : 0.0000
##  3rd Qu.:-0.4222           3rd Qu.: 0.4325    3rd Qu.: 0.3905
##  Max.   : 5.0700           Max.   : 2.5244    Max.   : 8.4652
```

# 1. PCA

Load required library

Perform PCA with 2 components

```
pca_2 <- prcomp(data, retx = TRUE, rank = 2)
```

Extract the reduced data with 2 principal components

```
reduced_data <- as.matrix(pca_2$x)
head(reduced_data)
```

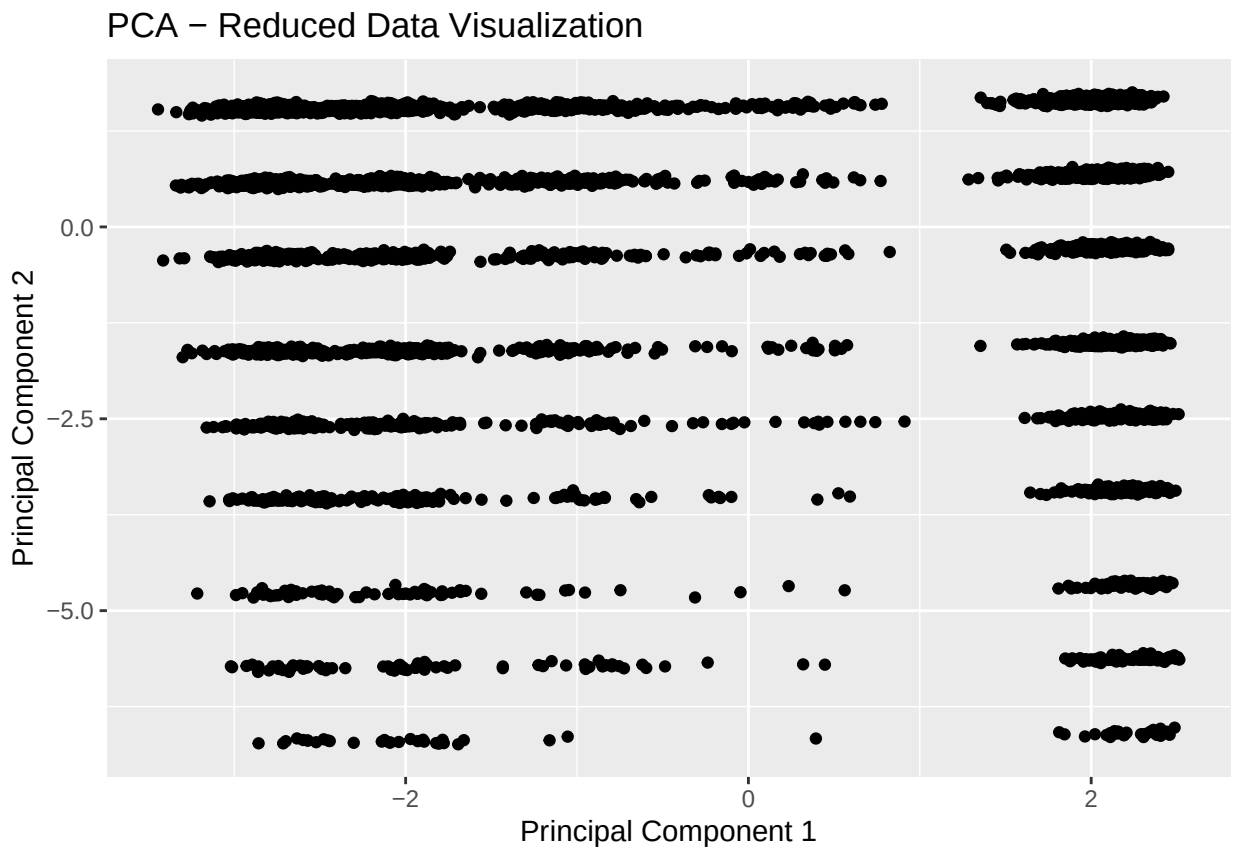
```
##            PC1        PC2
## [1,] -2.708670 -1.6141241
## [2,]  1.689366  1.6235686
## [3,] -3.124832  1.5136796
## [4,] -2.008754 -2.5456237
## [5,]  2.212617 -0.2484373
## [6,] -2.717361  1.5577718
```

PLOT THE COMPONENTS Convert the reduced_data matrix to a data frame

```
reduced_data_df <- as.data.frame(reduced_data)
```

Create a scatter plot of the reduced data

```
ggplot(data = reduced_data_df, aes(x = PC1, y = PC2)) +
  geom_point() +
  labs(x = "Principal Component 1", y = "Principal Component 2", title = "PCA - Reduced Data Visualizati
```




# 2. PCA & Kmeans

### Determine number of clusters:

Select k on reduced data: wss cho ca bo du lieu (1 cum, 2 thuoc tinh)

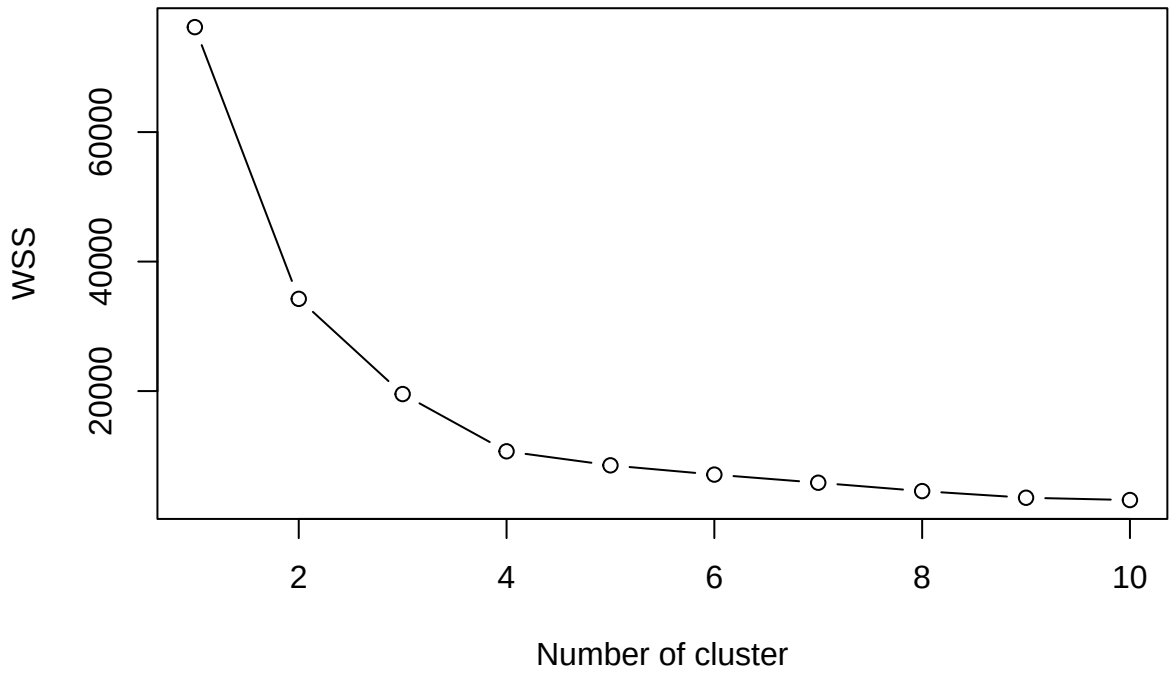
```
wss2 = (nrow(reduced_data) -1)*sum(apply(reduced_data, 2, var))

for (i in 1:10) {
  wss2[i] = sum(kmeans(reduced_data,
                       centers = i,
                       nstart = 20)$withinss)
  cat("Number of clusters (k):", i, "\tWSS:", wss2[i], "\n")
}
```

```
## Number of clusters (k): 1    WSS: 76211.11
## Number of clusters (k): 2    WSS: 34241.06
## Number of clusters (k): 3    WSS: 19538.75
## Number of clusters (k): 4    WSS: 10697.5
## Number of clusters (k): 5    WSS: 8530.665
## Number of clusters (k): 6    WSS: 7098.129
## Number of clusters (k): 7    WSS: 5844.288
## Number of clusters (k): 8    WSS: 4552.105
## Number of clusters (k): 9    WSS: 3525.282
## Number of clusters (k): 10   WSS: 3171.885
```

Plot wss

```
plot(1:10, wss2, type='b', xlab="Number of cluster",
     ylab="WSS")
```




-> select k=4

### Buil model su dung kmeans

```
set.seed(20)
k.means.fit <- kmeans(reduced_data, centers = 4) # k = 4
```

Cluster

```
cluster_assignments <- k.means.fit$cluster
head(cluster_assignments)
```

```
## [1] 2 1 3 2 1 3
```

Cluster size

```r
cluster_size <- k.means.fit$size
cluster_size
```

```
## [1] 3529 1125 3259 1221
```

### Compute cluster statistics

```r
km_stats <- cluster.stats(dist(reduced_data), cluster_assignments)
km_stats
```

```
## $n
## [1] 9134
##
## $cluster.number
## [1] 4
##
## $cluster.size
## [1] 3529 1125 3259 1221
##
## $min.cluster.size
## [1] 1125
##
## $noisen
## [1] 0
##
## $diameter
## [1] 3.232247 5.836666 3.834554 5.580128
##
## $average.distance
## [1] 0.9244987 1.7798028 1.2042583 1.5844477
##
## $median.distance
## [1] 0.9670176 1.4051993 1.0940504 1.0262449
##
## $separation
## [1] 0.06202023 0.18218947 0.06202023 0.18218947
##
## $average.toother
## [1] 4.540266 4.784090 4.582812 4.734827
##
## $separation.matrix
##            [,1]      [,2]       [,3]      [,4]
## [1,] 0.00000000 1.1825070 0.06202023 1.0919865
## [2,] 1.18250702 0.0000000 1.10546696 0.1821895
## [3,] 0.06202023 1.1054670 0.00000000 1.2309758
## [4,] 1.09198651 0.1821895 1.23097577 0.0000000
##
## $ave.between.matrix
##          [,1]     [,2]     [,3]     [,4]
## [1,] 0.000000 5.757412 4.417741 3.745849
## [2,] 5.757412 0.000000 3.760963 4.701800
```

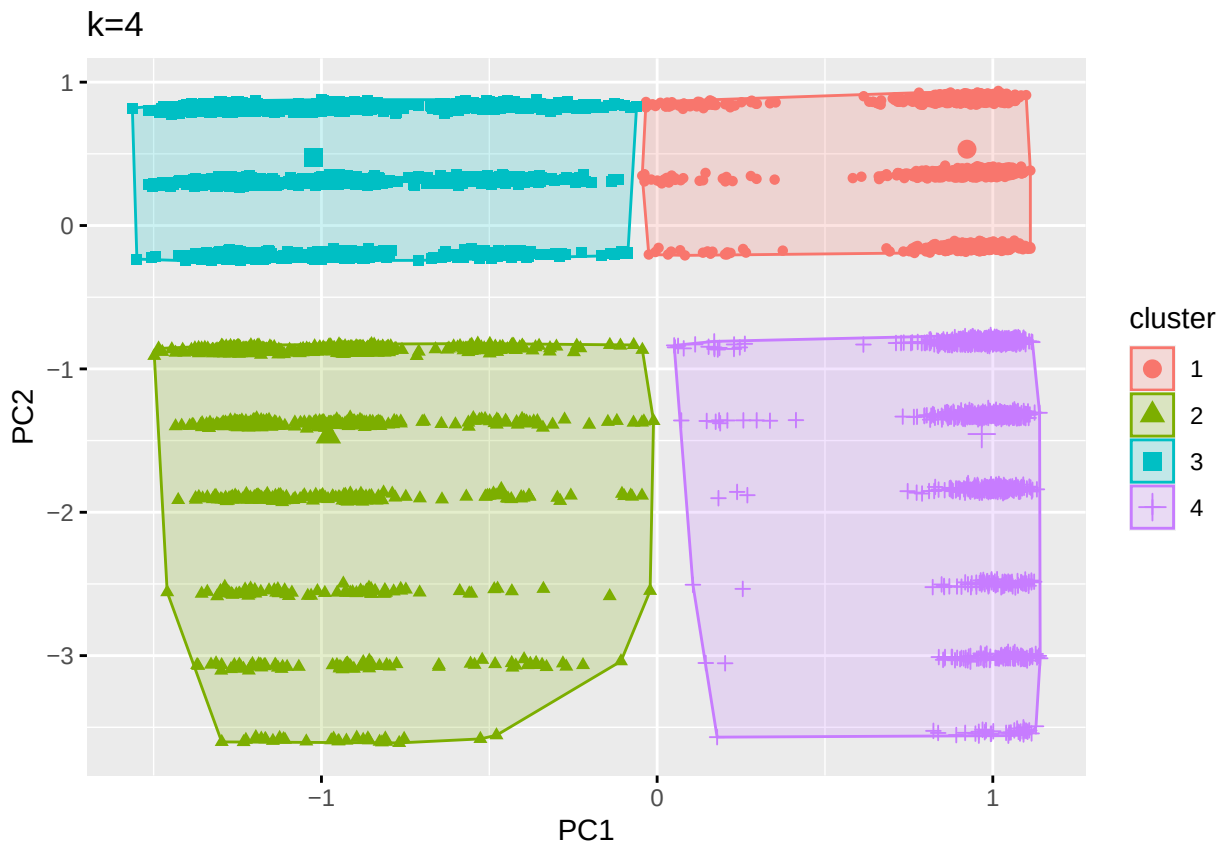
```
## [3,] 4.417741 3.760963 0.000000 5.817139
## [4,] 3.745849 4.701800 5.817139 0.000000
##
## $average.between
## [1] 4.625186
##
## $average.within
## [1] 1.217881
##
## $n.between
## [1] 28799284
##
## $n.within
## [1] 12911127
##
## $max.diameter
## [1] 5.836666
##
## $min.separation
## [1] 0.06202023
##
## $within.cluster.ss
## [1] 10697.5
##
## $clus.avg.silwidths
##         1         2         3         4
## 0.7221208 0.4848006 0.6412090 0.5470960
##
## $avg.silwidth
## [1] 0.6406251
##
## $g2
## NULL
##
## $g3
## NULL
##
## $pearsongamma
## [1] 0.7948632
##
## $dunn
## [1] 0.01062597
##
## $dunn2
## [1] 2.104643
##
## $entropy
## [1] 1.262072
##
## $wb.ratio
## [1] 0.263315
##
## $ch
## [1] 18637.97
```

```
##
## $cwidegap
## [1] 0.8239565 1.0679435 0.8076171 1.1398885
##
## $widestgap
## [1] 1.139889
##
## $sindex
## [1] 0.8893256
##
## $corrected.rand
## NULL
##
## $vi
## NULL
```

- Within-Cluster Sum of Squares (WCSS): 10697.5
- Silhouette Score: 0.6406251

Plot

```
fviz_cluster(k.means.fit, geo= "point",
             data = reduced_data) + ggtitle("k=4")
```

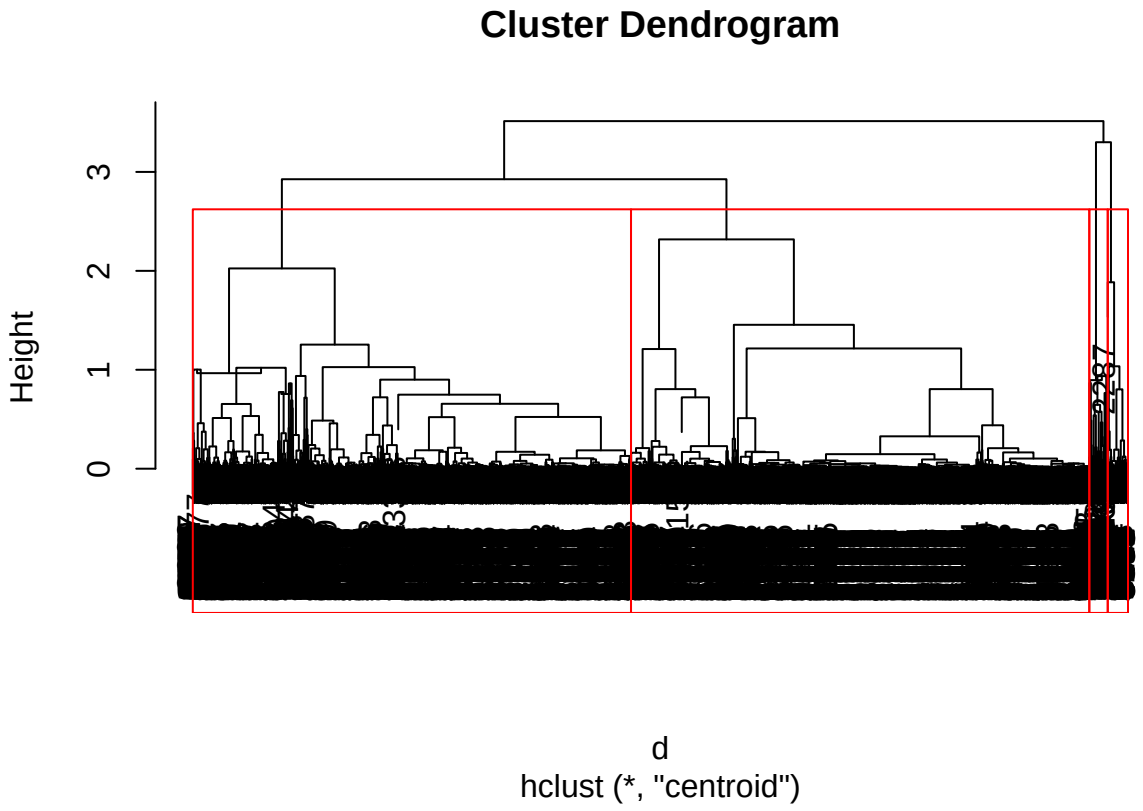


# 3. PCA & HIERARCHICAL

## Build model

```
d <- dist(reduced_data, method = "euclidean")
H.fit <- hclust(d, method="centroid")
plot(H.fit) # display dendogram
groups <- cutree(H.fit, k=4)

# draw dendogram with red borders around the 4 clusters
rect.hclust(H.fit, k=4, border="red")
```




## Compute cluster statistics

```
km_stats2 <- cluster.stats(dist(reduced_data), groups)
km_stats2
```

```
## $n
## [1] 9134
##
## $cluster.number
## [1] 4
```

```
##
## $cluster.size
## [1] 4280 4476  198  180
##
## $min.cluster.size
## [1] 180
##
## $noisen
## [1] 0
##
## $diameter
## [1] 6.460961 5.671284 2.903425 3.958864
##
## $average.distance
## [1] 1.9676881 1.6864982 0.8390447 1.2661665
##
## $median.distance
## [1] 1.8305758 1.0745543 0.9618243 1.1471376
##
## $separation
## [1] 0.06202023 0.06202023 0.96362898 0.96362898
##
## $average.toother
## [1] 4.892643 4.890309 6.457954 6.446843
##
## $separation.matrix
##            [,1]       [,2]     [,3]     [,4]
## [1,] 0.00000000 0.06202023 1.707449 1.067944
## [2,] 0.06202023 0.00000000 1.121918 1.670830
## [3,] 1.70744875 1.12191831 0.000000 0.963629
## [4,] 1.06794350 1.67082989 0.963629 0.000000
##
## $ave.between.matrix
##          [,1]     [,2]     [,3]     [,4]
## [1,] 0.000000 4.752696 7.266779 5.761122
## [2,] 4.752696 0.000000 5.769424 7.195410
## [3,] 7.266779 5.769424 0.000000 4.347357
## [4,] 5.761122 7.195410 4.347357 0.000000
##
## $average.between
## [1] 5.008759
##
## $average.within
## [1] 1.791604
##
## $n.between
## [1] 22502688
##
## $n.within
## [1] 19207723
##
## $max.diameter
## [1] 6.460961
##
```

```
## $min.separation
## [1] 0.06202023
##
## $within.cluster.ss
## [1] 22663.27
##
## $clus.avg.silwidths
##         1         2         3         4
## 0.5054180 0.5741699 0.8060398 0.6861620
##
## $avg.silwidth
## [1] 0.5491875
##
## $g2
## NULL
##
## $g3
## NULL
##
## $pearsongamma
## [1] 0.7796714
##
## $dunn
## [1] 0.009599228
##
## $dunn2
## [1] 2.209373
##
## $entropy
## [1] 0.8651775
##
## $wb.ratio
## [1] 0.3576942
##
## $ch
## [1] 7190.663
##
## $cwidegap
## [1] 1.1054670 1.0919865 1.4217827 0.8684766
##
## $widestgap
## [1] 1.421783
##
## $sindex
## [1] 0.9817452
##
## $corrected.rand
## NULL
##
## $vi
## NULL
```

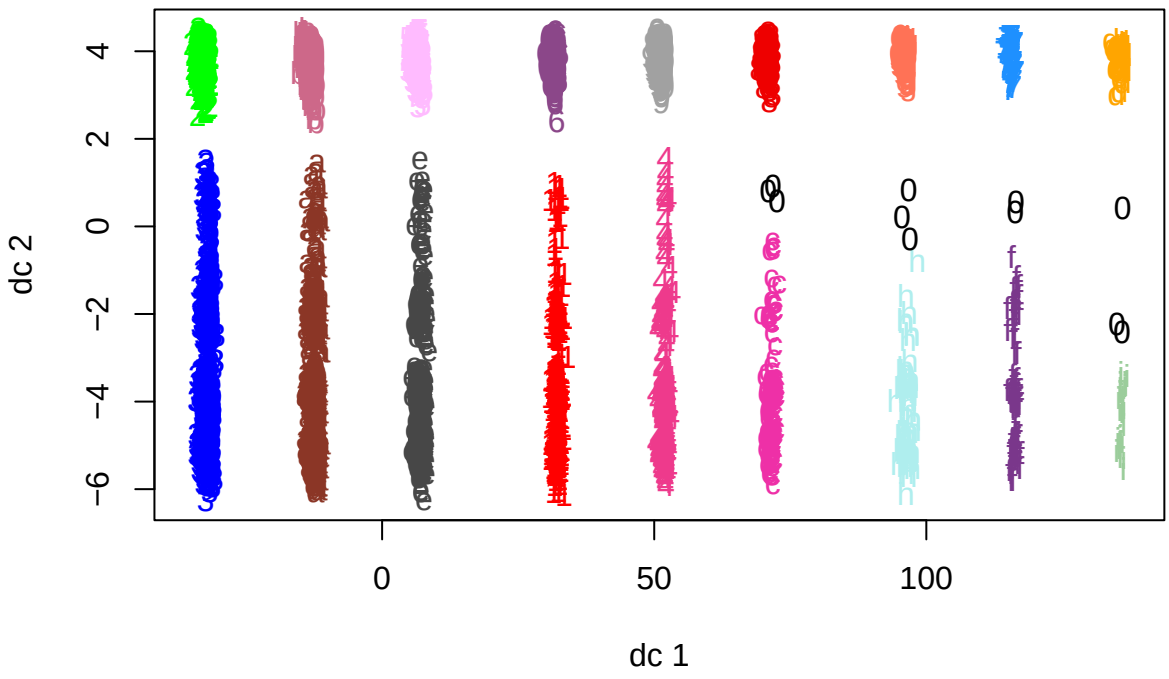
- Within-Cluster Sum of Squares (WCSS): 22663.27
- Silhouette Score: 0.5491875

# 4. PCA & DBSCAN

Compute DBSCAN clustering

```
dbscanOutput <- dbscan(reduced_data, eps = 0.5, MinPts = 4)

plotcluster(reduced_data, dbscanOutput$cluster)
```




### Compute cluster statistics

```
# Convert DBSCAN clustering result to a factor (or character) vector
dbscan_groups <- as.numeric(dbscanOutput$cluster)

# Compute cluster statistics using silhouette analysis
km_stats3 <- cluster.stats(dist(reduced_data), dbscan_groups)
```

```
## Warning in cluster.stats(dist(reduced_data), dbscan_groups): clustering
## renumbered because maximum != number of clusters
```

```
km_stats3
```

```
## $n
```

```
## [1] 9134
##
## $cluster.number
## [1] 19
##
## $cluster.size
##  [1]   11  518 1742 1684  293  640  496   89  193  302 1071 1051  163   36  600
## [16]   73   72   73   27
##
## $min.cluster.size
## [1] 11
##
## $noisen
## [1] 0
##
## $diameter
##  [1] 3.6327285 3.8797206 1.0663997 4.2235206 4.0729273 0.9448388 1.1097313
##  [8] 0.6655829 0.8485908 0.8996427 4.1122224 1.1696807 3.0448889 0.6760789
## [15] 4.2409950 2.7824910 0.6695508 2.9049055 1.1980592
##
## $average.distance
##  [1] 1.7358521 0.7905011 0.1825330 0.8155939 0.8428800 0.1715980 0.1793347
##  [8] 0.1944357 0.1733012 0.1889430 0.7707866 0.1936637 0.7499257 0.1981019
## [15] 0.8464209 0.8428943 0.1794547 0.6641884 0.4262730
##
## $median.distance
##  [1] 1.6654233 0.6400105 0.1476807 0.6446970 0.6277721 0.1408765 0.1417026
##  [8] 0.1601176 0.1409518 0.1561879 0.6213090 0.1549261 0.5951093 0.1679974
## [15] 0.6588583 0.7187752 0.1488905 0.5991694 0.3836002
##
## $separation
##  [1] 0.2740646 0.7782195 0.5833978 0.5833978 0.7024019 0.6810997 0.7782195
##  [8] 0.8395609 0.8331072 0.7024019 0.5134423 0.5134423 0.5031280 0.8587491
## [15] 0.6810997 0.5572189 0.8395609 0.2740646 0.5004112
##
## $average.toother
##  [1] 5.737141 3.868740 3.963803 4.061594 4.314327 3.334757 3.736673 6.531148
##  [9] 4.798965 4.190580 3.570079 3.387299 4.903436 7.357506 3.472702 6.574767
## [17] 5.707968 5.823454 7.448882
##
## $separation.matrix
##            [,1]      [,2]      [,3]      [,4]      [,5]      [,6]      [,7]
##  [1,] 0.0000000 1.8600058 5.1337158 5.0037591 0.9016094 3.2893941 2.0917384
##  [2,] 1.8600058 0.0000000 3.2414504 3.0093438 0.8280823 1.5334331 0.7782195
##  [3,] 5.1337158 3.2414504 0.0000000 0.5833978 4.1485727 1.7742974 3.0031184
##  [4,] 5.0037591 3.0093438 0.5833978 0.0000000 3.9764038 2.0350709 3.2024041
##  [5,] 0.9016094 0.8280823 4.1485727 3.9764038 0.0000000 2.2823605 1.0800171
##  [6,] 3.2893941 1.5334331 1.7742974 2.0350709 2.2823605 0.0000000 1.0919865
##  [7,] 2.0917384 0.7782195 3.0031184 3.2024041 1.0800171 1.0919865 0.0000000
##  [8,] 1.4047847 4.2343028 7.1321612 7.2932176 3.2266930 5.2187461 3.9853569
##  [9,] 1.0533482 2.1734079 4.9312076 5.1149602 1.1810626 3.0132650 1.7908427
## [10,] 1.4453044 1.3984311 3.9529111 4.1737136 0.7024019 2.0397224 0.8065580
## [11,] 4.0466663 2.0532454 1.1899256 0.8026288 3.0057374 1.1575562 2.2265090
## [12,] 4.1589751 2.2722980 0.8021872 1.1082034 3.1757089 0.8239565 2.0533029
```

```
## [13,] 0.5031280 1.7677656 5.3294028 4.9097871 0.8390351 3.5721202 2.4463949
## [14,] 1.4217827 5.1456808 8.1090576 8.2335068 4.1479695 6.1979387 4.9633371
## [15,] 3.0947733 1.1054670 2.0081093 1.7714055 2.0625290 0.6810997 1.3331747
## [16,] 0.5572189 3.9813181 7.4498940 7.1260034 3.0195084 5.6227993 4.4224369
## [17,] 1.2471097 3.3572899 6.1852410 6.3660285 2.3442880 4.2709370 3.0385308
## [18,] 0.2740646 2.9933505 6.6463129 6.1402762 2.0331375 4.8521569 3.6766415
## [19,] 0.5004112 4.9816256 8.8352909 8.1345033 4.0324535 7.1047665 5.9562471
##           [,8]      [,9]     [,10]     [,11]     [,12]     [,13]     [,14]
##  [1,] 1.4047847 1.0533482 1.4453044 4.0466663 4.1589751 0.5031280 1.4217827
##  [2,] 4.2343028 2.1734079 1.3984311 2.0532454 2.2722980 1.7677656 5.1456808
##  [3,] 7.1321612 4.9312076 3.9529111 1.1899256 0.8021872 5.3294028 8.1090576
##  [4,] 7.2932176 5.1149602 4.1737136 0.8026288 1.1082034 4.9097871 8.2335068
##  [5,] 3.2266930 1.1810626 0.7024019 3.0057374 3.1757089 0.8390351 4.1479695
##  [6,] 5.2187461 3.0132650 2.0397224 1.1575562 0.8239565 3.5721202 6.1979387
##  [7,] 3.9853569 1.7908427 0.8065580 2.2265090 2.0533029 2.4463949 4.9633371
##  [8,] 0.0000000 2.0668560 3.0330420 6.3140816 6.1821238 2.8679969 0.8587491
##  [9,] 2.0668560 0.0000000 0.8331072 4.1408046 3.9787184 1.7447886 3.0447221
## [10,] 3.0330420 0.8331072 0.0000000 3.1999498 3.0026581 1.9971512 4.0096045
## [11,] 6.3140816 4.1408046 3.1999498 0.0000000 0.5134423 3.9674624 7.2578680
## [12,] 6.1821238 3.9787184 3.0026581 0.5134423 0.0000000 4.3617876 7.1592462
## [13,] 2.8679969 1.7447886 1.9971512 3.9674624 4.3617876 0.0000000 3.6136556
## [14,] 0.8587491 3.0447221 4.0096045 7.2578680 7.1592462 3.6136556 0.0000000
## [15,] 5.3942505 3.2375903 2.3011100 0.8076171 1.0509281 3.0030394 6.3349726
## [16,] 2.0869436 2.9066283 3.6852863 6.1853555 6.4761296 2.0720739 2.2426389
## [17,] 0.8395609 1.1219183 2.0870324 5.3871267 5.2350118 2.2505963 1.8137206
## [18,] 2.3017837 2.3861256 3.0284982 5.1925649 5.6754002 1.0679435 2.7564375
## [19,] 3.6673350 4.6185224 5.3232849 7.1582793 7.8768710 3.0667859 3.4755144
##           [,15]     [,16]     [,17]     [,18]     [,19]
##  [1,] 3.0947733 0.5572189 1.2471097 0.2740646 0.5004112
##  [2,] 1.1054670 3.9813181 3.3572899 2.9933505 4.9816256
##  [3,] 2.0081093 7.4498940 6.1852410 6.6463129 8.8352909
##  [4,] 1.7714055 7.1260034 6.3660285 6.1402762 8.1345033
##  [5,] 2.0625290 3.0195084 2.3442880 2.0331375 4.0324535
##  [6,] 0.6810997 5.6227993 4.2709370 4.8521569 7.1047665
##  [7,] 1.3331747 4.4224369 3.0385308 3.6766415 5.9562471
##  [8,] 5.3942505 2.0869436 0.8395609 2.3017837 3.6673350
##  [9,] 3.2375903 2.9066283 1.1219183 2.3861256 4.6185224
## [10,] 2.3011100 3.6852863 2.0870324 3.0284982 5.3232849
## [11,] 0.8076171 6.1853555 5.3871267 5.1925649 7.1582793
## [12,] 1.0509281 6.4761296 5.2350118 5.6754002 7.8768710
## [13,] 3.0030394 2.0720739 2.2505963 1.0679435 3.0667859
## [14,] 6.3349726 2.2426389 1.8137206 2.7564375 3.4755144
## [15,] 0.0000000 5.2207022 4.4723651 4.2234519 6.2159555
## [16,] 5.2207022 0.0000000 2.2625735 0.8515705 0.8684766
## [17,] 4.4723651 2.2625735 0.0000000 2.1235486 3.9928280
## [18,] 4.2234519 0.8515705 2.1235486 0.0000000 1.8429257
## [19,] 6.2159555 0.8684766 3.9928280 1.8429257 0.0000000
##
## $ave.between.matrix
##           [,1]     [,2]      [,3]     [,4]     [,5]      [,6]      [,7]
##  [1,] 0.000000 4.324866 7.0429497 7.123242 3.605267 5.2421881 4.1671732
##  [2,] 4.324866 0.000000 5.3376771 3.325686 1.362781 4.4560045 4.2866938
##  [3,] 7.042950 5.337677 0.0000000 4.259247 5.966318 1.9413991 3.1677132
##  [4,] 7.123242 3.325686 4.2592470 0.000000 4.266414 4.6921818 5.3386284
```

```
##  [5,] 3.605267 1.362781 5.9663177 4.266414 0.000000 4.8200704 4.3930227
##  [6,] 5.242188 4.456004 1.9413991 4.692182 4.820070 0.0000000 1.2493174
##  [7,] 4.167173 4.286694 3.1677132 5.338628 4.393023 1.2493174 0.0000000
##  [8,] 2.516063 5.959138 7.2821788 8.459060 5.330141 5.3539717 4.1253220
##  [9,] 2.792995 4.685100 5.0874868 6.650705 4.361812 3.1606382 1.9356206
## [10,] 3.398712 4.380971 4.1280058 5.952804 4.261769 2.2035042 0.9880646
## [11,] 6.237989 2.407100 4.4079953 1.342664 3.330331 4.4093908 4.8514704
## [12,] 6.130512 4.816888 0.9922248 4.368168 5.334123 0.9895985 2.2069250
## [13,] 3.014176 2.157884 6.6831512 5.191648 1.347678 5.3525352 4.7313100
## [14,] 2.888140 6.671797 8.2525688 9.308835 5.963998 6.3241181 5.0949921
## [15,] 5.361362 1.575433 4.6831017 2.191097 2.443632 4.2513833 4.4447833
## [16,] 2.489814 4.250999 8.4290792 7.357643 3.334993 6.8282500 5.9158647
## [17,] 2.441430 5.335326 6.3212609 7.647335 4.815081 4.3933596 3.1654430
## [18,] 2.661647 3.293503 7.7112177 6.390490 2.395260 6.2264340 5.4341470
## [19,] 2.950062 5.155370 9.3900403 8.289832 4.217562 7.7515890 6.7934745
##           [,8]      [,9]     [,10]    [,11]     [,12]    [,13]     [,14]
##  [1,] 2.5160634 2.7929950 3.3987121 6.237989 6.1305124 3.014176 2.8881395
##  [2,] 5.9591382 4.6850999 4.3809709 2.407100 4.8168883 2.157884 6.6717969
##  [3,] 7.2821788 5.0874868 4.1280058 4.407995 0.9922248 6.683151 8.2525688
##  [4,] 8.4590599 6.6507049 5.9528043 1.342664 4.3681682 5.191648 9.3088351
##  [5,] 5.3301405 4.3618121 4.2617687 3.330331 5.3341229 1.347678 5.9639985
##  [6,] 5.3539717 3.1606382 2.2035042 4.409391 0.9895985 5.352535 6.3241181
##  [7,] 4.1253220 1.9356206 0.9880646 4.851470 2.2069250 4.731310 5.0949921
##  [8,] 0.0000000 2.2071527 3.1653367 7.660632 6.3178999 4.826631 0.9991272
##  [9,] 2.2071527 0.0000000 0.9856117 5.964499 4.1239004 4.277170 3.1747244
## [10,] 3.1653367 0.9856117 0.0000000 5.349952 3.1653665 4.401604 4.1342899
## [11,] 7.6606318 5.9644988 5.3499524 0.000000 4.2925449 4.241637 8.4804452
## [12,] 6.3178999 4.1239004 3.1653665 4.292545 0.0000000 5.974096 7.2881381
## [13,] 4.8266311 4.2771705 4.4016037 4.241637 5.9740965 0.000000 5.3447393
## [14,] 0.9991272 3.1747244 4.1342899 8.480445 7.2881381 5.344739 0.0000000
## [15,] 6.8611905 5.3050373 4.7967597 1.352951 4.3592363 3.323698 7.6454453
## [16,] 4.1899716 4.7365166 5.2702662 6.400309 7.6070652 2.398861 4.3043645
## [17,] 0.9867580 1.2528702 2.2069062 6.888961 5.3571728 4.468300 1.9454313
## [18,] 4.4944343 4.5514411 4.9167232 5.432395 6.9385543 1.488589 4.8019892
## [19,] 4.5808654 5.4762997 6.0928355 7.326291 8.5529210 3.249353 4.4772396
##          [,15]    [,16]    [,17]    [,18]    [,19]
##  [1,] 5.361362 2.489814 2.441430 2.661647 2.950062
##  [2,] 1.575433 4.250999 5.335326 3.293503 5.155370
##  [3,] 4.683102 8.429079 6.321261 7.711218 9.390040
##  [4,] 2.191097 7.357643 7.647335 6.390490 8.289832
##  [5,] 2.443632 3.334993 4.815081 2.395260 4.217562
##  [6,] 4.251383 6.828250 4.393360 6.226434 7.751589
##  [7,] 4.444783 5.915865 3.165443 5.434147 6.793474
##  [8,] 6.861190 4.189972 0.986758 4.494434 4.580865
##  [9,] 5.305037 4.736517 1.252870 4.551441 5.476300
## [10,] 4.796760 5.270266 2.206906 4.916723 6.092836
## [11,] 1.352951 6.400309 6.888961 5.432395 7.326291
## [12,] 4.359236 7.607065 5.357173 6.938554 8.552921
## [13,] 3.323698 2.398861 4.468300 1.488589 3.249353
## [14,] 7.645445 4.304365 1.945431 4.801989 4.477240
## [15,] 0.000000 5.459429 6.138430 4.497260 6.378801
## [16,] 5.459429 0.000000 4.306340 1.306977 1.251941
## [17,] 6.138430 4.306340 0.000000 4.391362 4.879387
## [18,] 4.497260 1.306977 4.391362 0.000000 2.042898
```

```
## [19,] 6.378801 1.251941 4.879387 2.042898 0.000000
##
## $average.between
## [1] 3.93986
##
## $average.within
## [1] 0.4898487
##
## $n.between
## [1] 36878587
##
## $n.within
## [1] 4831824
##
## $max.diameter
## [1] 4.240995
##
## $min.separation
## [1] 0.2740646
##
## $within.cluster.ss
## [1] 2734.483
##
## $clus.avg.silwidths
##           1           2           3           4           5           6
## -0.006356621  0.431551912  0.817073907  0.404445638  0.381935353  0.827586672
##           7           8           9          10          11          12
##  0.819603681  0.801385164  0.825346034  0.805479502  0.434681132  0.802148964
##          13          14          15          16          17          18
##  0.455573103  0.802887620  0.385587724  0.314317227  0.818614524  0.495835791
##          19
##  0.661043396
##
## $avg.silwidth
## [1] 0.6162586
##
## $g2
## NULL
##
## $g3
## NULL
##
## $pearsongamma
## [1] 0.5417533
##
## $dunn
## [1] 0.06462272
##
## $dunn2
## [1] 0.567797
##
## $entropy
## [1] 2.397957
##
```

```
## $wb.ratio
## [1] 0.1243315
##
## $ch
## [1] 13606.87
##
## $cwidegap
##  [1] 1.44858419 0.21660155 0.08045847 0.11243927 0.18218947 0.08969957
##  [7] 0.21478880 0.05496253 0.06190555 0.07342215 0.16328964 0.11406510
## [13] 0.33641363 0.12213437 0.24294558 0.27838591 0.06828678 0.44529089
## [19] 0.16444935
##
## $widestgap
## [1] 1.448584
##
## $sindex
## [1] 0.8273376
##
## $corrected.rand
## NULL
##
## $vi
## NULL
```

- Within-Cluster Sum of Squares (WCSS): 2734.483
- Silhouette Score: 0.6162586