Stat4601_Bronx_Kmeans

Stephanie Cheng

2025-04-09

K-mean with PCA

Reduce dimensions and prepare data for clustering

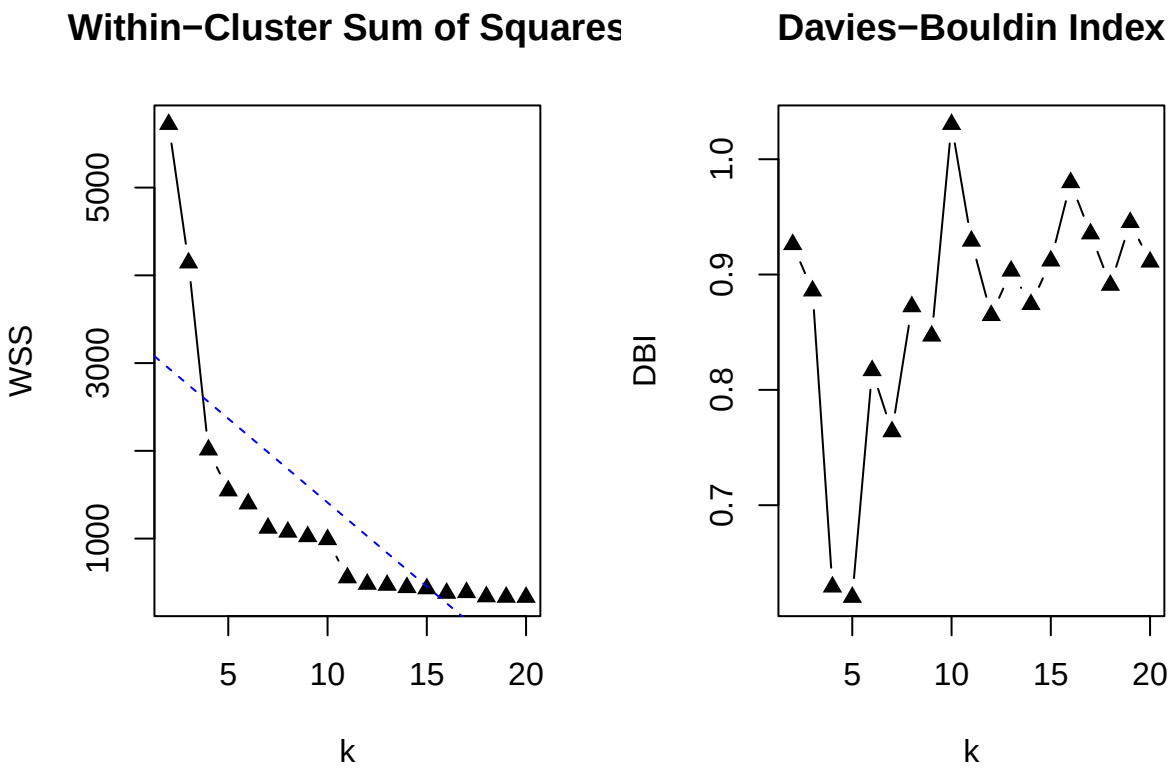
```
bronx_pca <- pca(bronx_data, "Bronx")
```

```
##
## ===== PCA Summary for Bronx =====
## Importance of components:
##                            PC1    PC2    PC3     PC4
## Standard deviation     1.4793 0.9729 0.9277 0.06785
## Proportion of Variance 0.5471 0.2366 0.2152 0.00115
## Cumulative Proportion  0.5471 0.7837 0.9989 1.00000
## Contributing variable for each PC:
##                 PC1                 PC2                 PC3                 PC4
##       "TOTAL.UNITS"        "YEAR.BUILT" "GROSS.SQUARE.FEET"       "TOTAL.UNITS"
```

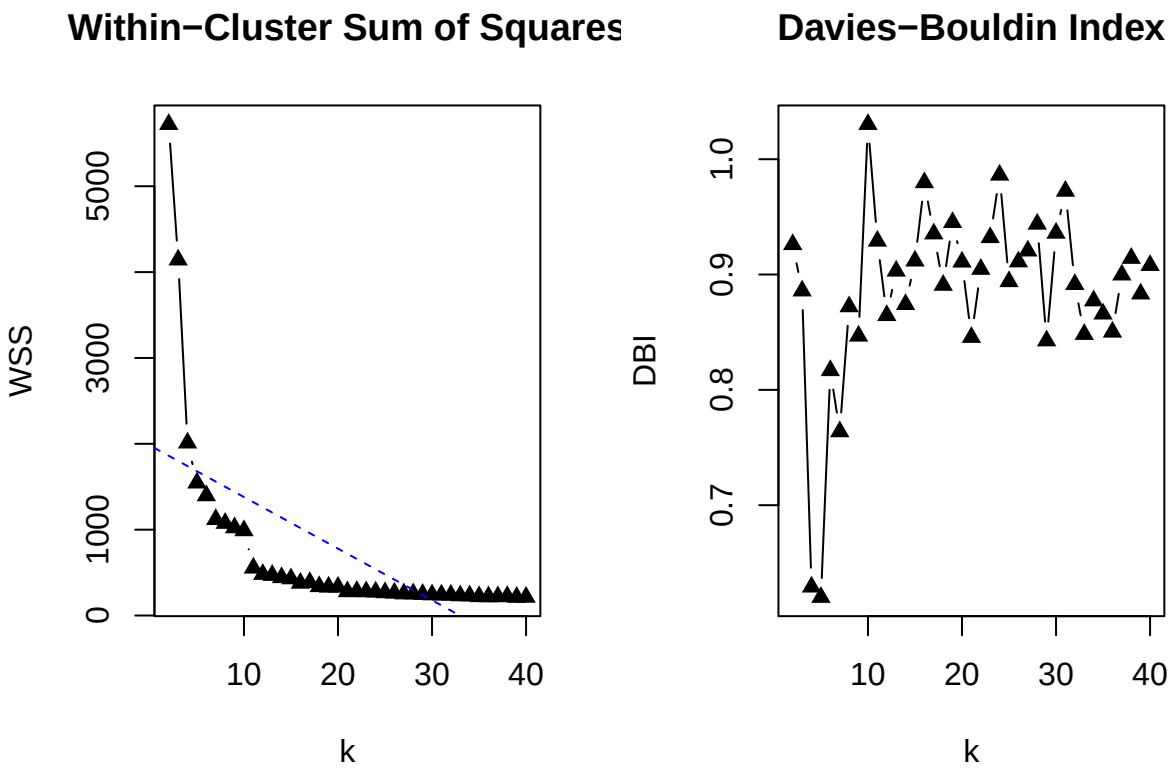
Calculate clustering evaluation with Davies Bouldin index & Within-cluster sum of squares. See the affect when K is increasing, then we can apply elbow method to avoid picking the best k within overfitting case.

```
bronx_k_stats_20 <- calculate_k_stats_PCA(bronx_pca, max_k = 20)
bronx_k_stats_40 <- calculate_k_stats_PCA(bronx_pca, max_k = 40)

# DBI & WSS plot
elbows_20 <- plot_kmeans(bronx_k_stats_20$errs, bronx_k_stats_20$DBI)
```



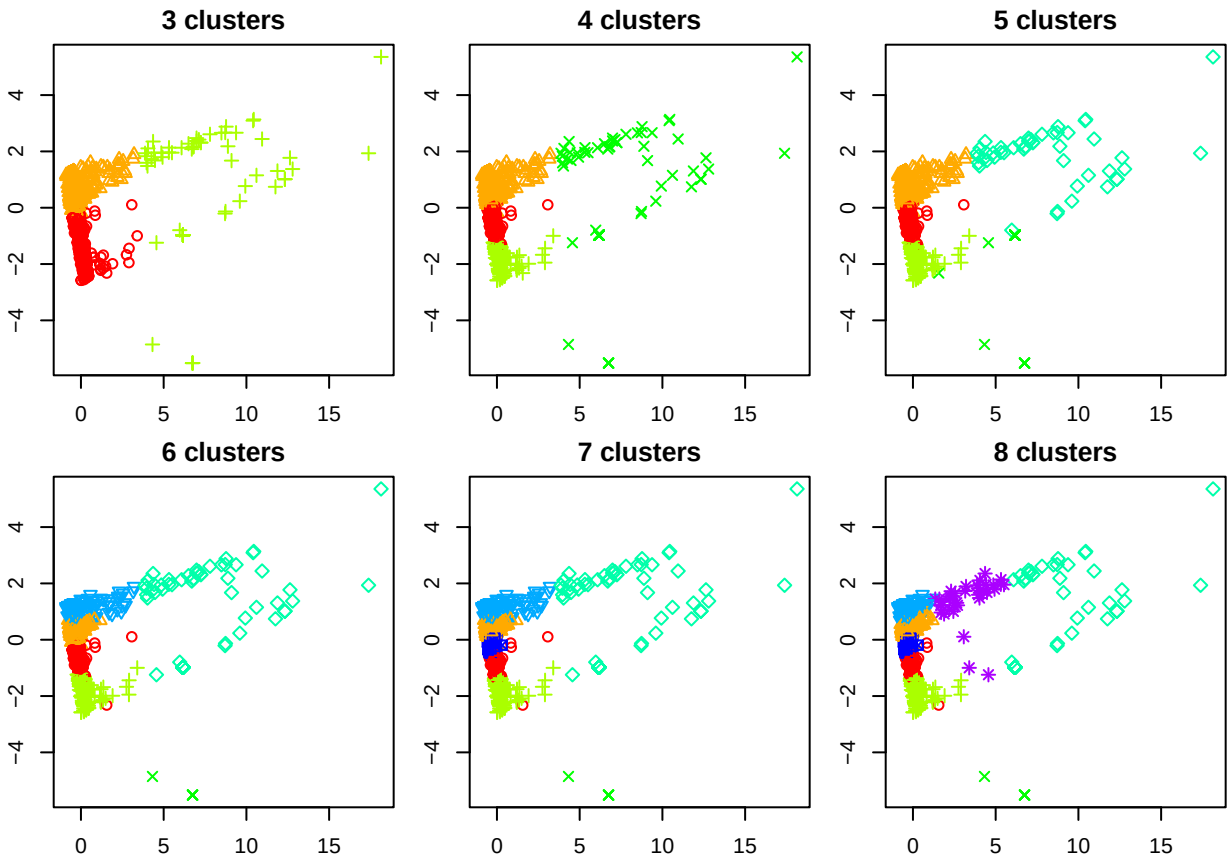

```
elbows_40 <- plot_kmeans(bronx_k_stats_40$errs, bronx_k_stats_40$DBI)
```




```
best_k <- 5
```

Plot all clusters from 3 to 8 as the best k clusters is within that range.

```
plot_clusters(bronx_k_stats_20$X.syn, min_k = 3, max_k = 8)
```



K-means on PCA as PCA gives a lower-dimensional variable that improves clustering quality

```
km <- kmeans(bronx_pca$x, centers = best_k, nstart = 25)
summarize_kmeans(km, "Bronx")
```

```
##
## ===== K-means Model Performance Summary for Bronx =====
## Total within-cluster sum of squares (WSS): 1544.846
##
## Cluster sizes:
## [1]  327  816 1376    6   78
##
## Cluster centers (in PCA space):
##   TOTAL.UNITS YEAR.BUILT GROSS.SQUARE.FEET  TOTAL.UNITS
## 1   0.2471721 -1.7929564        -0.5781557 -0.002782923
## 2  -0.2439522 -0.4161153        -0.1358207  0.004810778
## 3  -0.3610645  0.6505734         0.1873868 -0.002810606
## 4   6.3414766 -5.4058509        17.6950103  0.059998964
## 5   7.3976358  0.8088840        -0.8221507  0.006305407
```

Interpret what the clusters mean with the original data

```
bronx_data$cluster <- km$cluster
aggregate(. ~ cluster, data = bronx_data, mean)
```

```
##   cluster BOROUGH RESIDENTIAL.UNITS TOTAL.UNITS GROSS.SQUARE.FEET YEAR.BUILT
## 1       1       2          2.455657    2.654434          4101.743   2000.853
## 2       2       2          1.531863    1.648284          2391.548   1953.853
## 3       3       2          2.327762    2.567587          3197.442   1919.430
## 4       4       2          0.000000    5.166667        636931.667   1977.333
## 5       5       2         50.551282   50.897436         52171.051   1977.500
```

Export the clusters for Supervised learning

```
dir.create("after_cluster_dataset")
```

```
## Warning in dir.create("after_cluster_dataset"): 'after_cluster_dataset' already
## exists
```

```
write.csv(bronx_data,
          file = "after_cluster_dataset/cleaned_bronx_with_clusters.csv",
          row.names = FALSE)
```