Stat4601_Queens_Kmeans

Stephanie Cheng

2025-04-09

K-mean with PCA

Reduce dimensions and prepare data for clustering

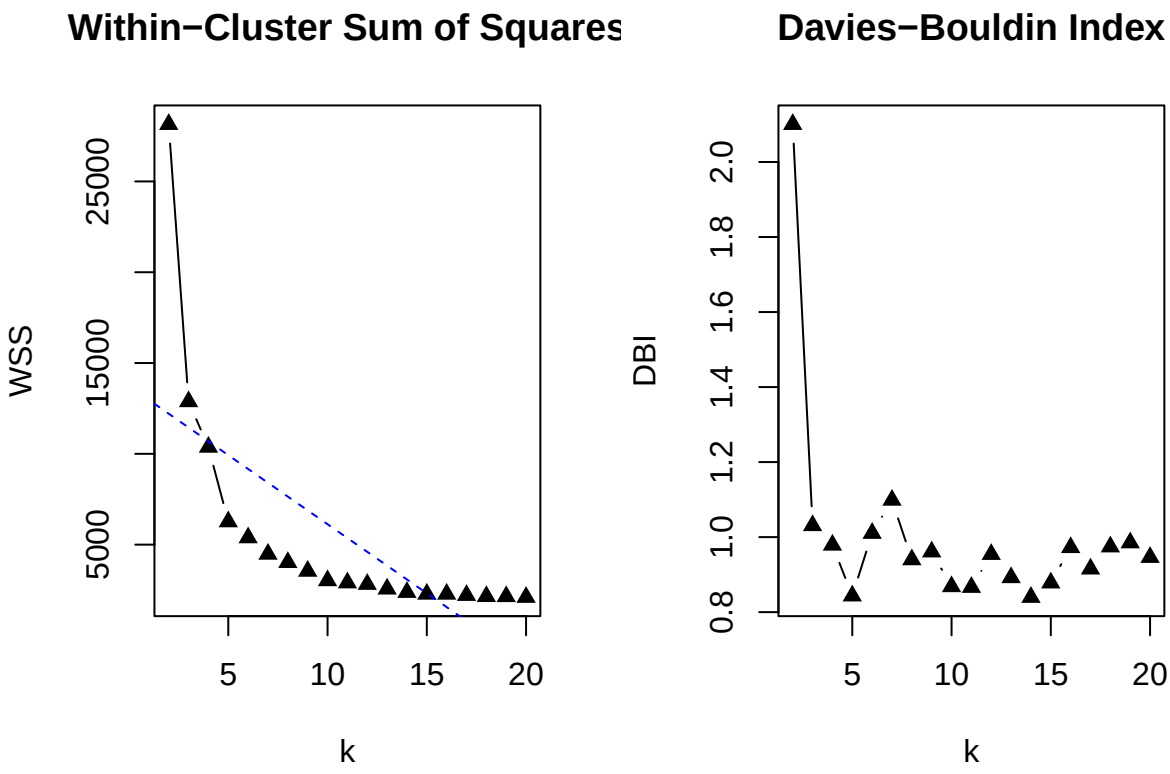
```
queens_pca <- pca(queens_data, "Queens")
```

```
##
## ===== PCA Summary for Queens =====
## Importance of components:
##                            PC1    PC2    PC3     PC4
## Standard deviation     1.6107 0.9935 0.6393 0.09931
## Proportion of Variance 0.6486 0.2468 0.1022 0.00247
## Cumulative Proportion  0.6486 0.8954 0.9975 1.00000
## Contributing variable for each PC:
##                 PC1                 PC2                 PC3                 PC4
##       "TOTAL.UNITS"        "YEAR.BUILT" "GROSS.SQUARE.FEET"       "TOTAL.UNITS"
```

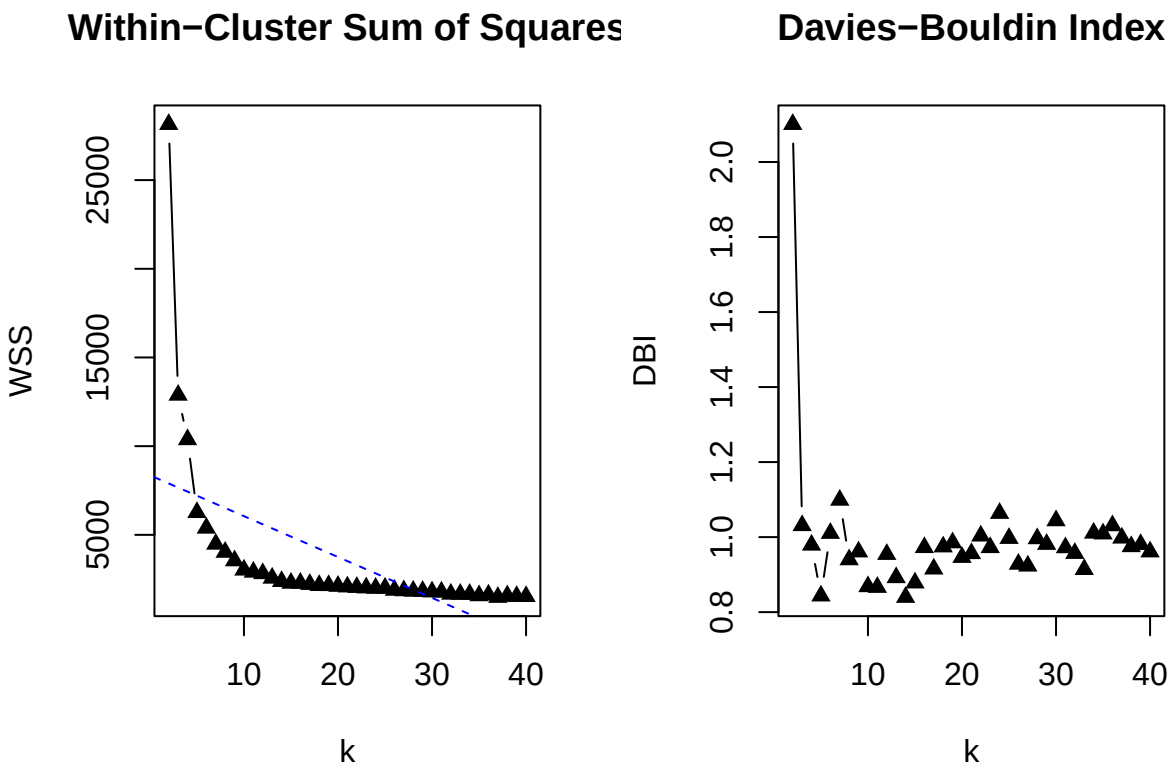
Calculate clustering evaluation with Davies Bouldin index & Within-cluster sum of squares. See the affect when K is increasing, then we can apply elbow method to avoid picking the best k within overfitting case.

```
queens_k_stats_20 <- calculate_k_stats_PCA(queens_pca, max_k = 20)
queens_k_stats_40 <- calculate_k_stats_PCA(queens_pca, max_k = 40)

# DBI & WSS plot
elbows_20 <- plot_kmeans(queens_k_stats_20$errs, queens_k_stats_20$DBI)
```



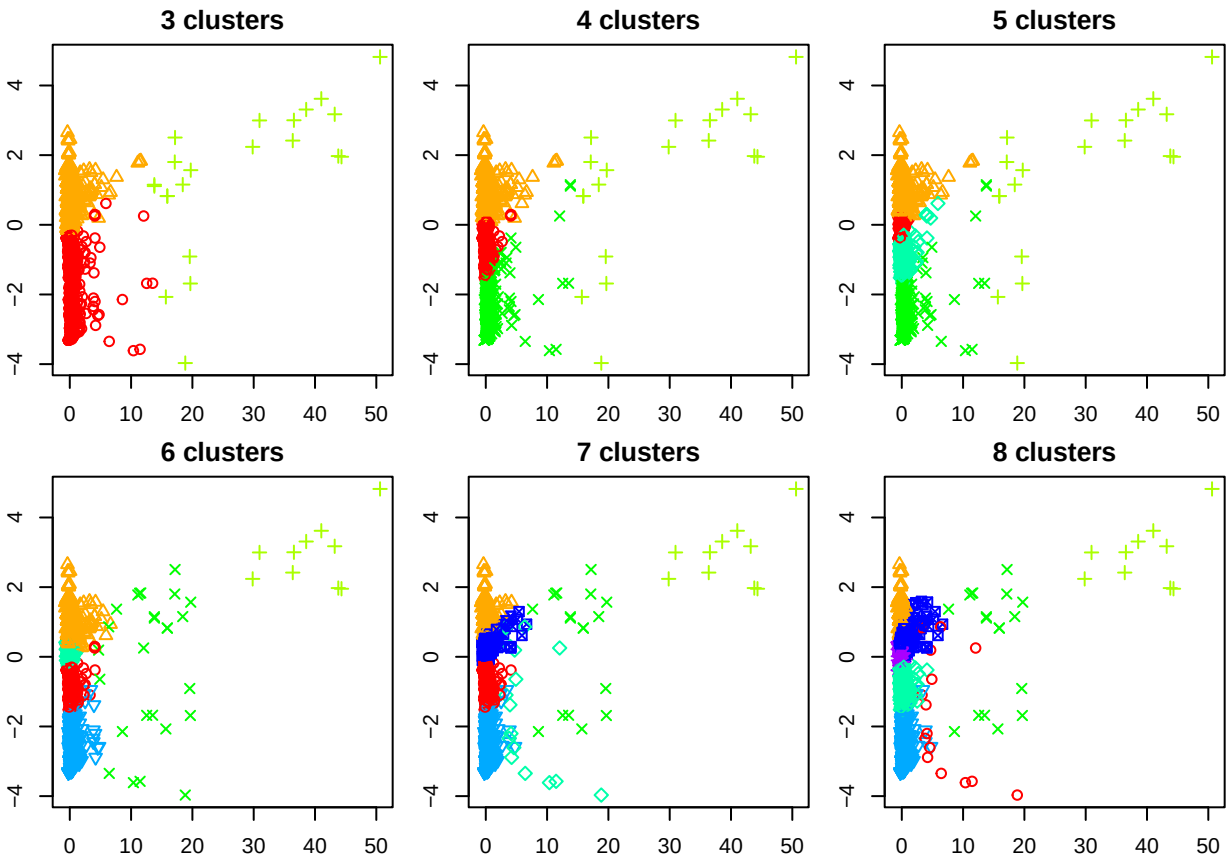

```
elbows_40 <- plot_kmeans(queens_k_stats_40$errs, queens_k_stats_40$DBI)
```




```
best_k <- 5
```

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

```
plot_clusters(queens_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(queens_pca$x, centers = best_k, nstart = 25)
summarize_kmeans(km, "Queens")
```

```
##
## ===== K-means Model Performance Summary for Queens =====
## Total within-cluster sum of squares (WSS): 6263.34
##
## Cluster sizes:
## [1]  745 3390 4457   10   26
##
## Cluster centers (in PCA space):
##   TOTAL.UNITS YEAR.BUILT GROSS.SQUARE.FEET  TOTAL.UNITS
## 1   0.3988670 -2.5002215       -0.11168724 -0.005548417
## 2  -0.1064499 -0.3734978       -0.03361711  0.003365383
## 3  -0.1502622  0.6967903        0.02864466 -0.001753757
## 4  39.5236170  2.9501908       -3.66287267  0.142498799
## 5  13.0073737 -0.2412922        4.08186471 -0.033983510
```

Interpret what the clusters mean with the original data

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

```
##   cluster BOROUGH RESIDENTIAL.UNITS TOTAL.UNITS GROSS.SQUARE.FEET YEAR.BUILT
## 1       1       4          2.111409    2.327517          3580.217   2004.944
## 2       2       4          1.377876    1.459882          1934.005   1950.786
## 3       3       4          1.672874    1.793134          1998.477   1924.025
## 4       4       4        155.300000  156.000000        123032.500   1978.600
## 5       5       4         38.846154   41.538462         73565.385   1972.577
```

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(queens_data,
          file = "after_cluster_dataset/cleaned_queens_with_clusters.csv",
          row.names = FALSE)
```