Assignment 6 Walkthrough

Warren Mansur

Outputting the Datasets

- It can be helpful to output the datasets so you can manually view them.

```
data("swiss") # Load the data frame into memory
write.csv(swiss, # <rite it out as-is to your working directory
          file = "swiss.csv",
          row.names = TRUE)   # keeps the canton names as the first column
```

K-Means in Plain Terms

- Picture your data points as dots on a sheet of paper; k-means first drops k imaginary pins (centers) onto that sheet.

K-Means in Plain Terms

- Picture your data points as dots on a sheet of paper; k-means first drops k imaginary pins (centers) onto that sheet.
- Each dot looks around, figures out which pin is closest, and joins that pin's group—this makes k rough clusters.

K-Means in Plain Terms

- Picture your data points as dots on a sheet of paper; k-means first drops k imaginary pins (centers) onto that sheet.
- Each dot looks around, figures out which pin is closest, and joins that pin's group—this makes k rough clusters.
- For every cluster, the algorithm slides its pin to the exact middle (average position) of the dots now assigned to it.

K-Means in Plain Terms

- Picture your data points as dots on a sheet of paper; k-means first drops k imaginary pins (centers) onto that sheet.
- Each dot looks around, figures out which pin is closest, and joins that pin's group—this makes k rough clusters.
- For every cluster, the algorithm slides its pin to the exact middle (average position) of the dots now assigned to it.
- With the pins moved, some dots now find a nearer pin, so they switch groups; pins then move again.

K-Means in Plain Terms

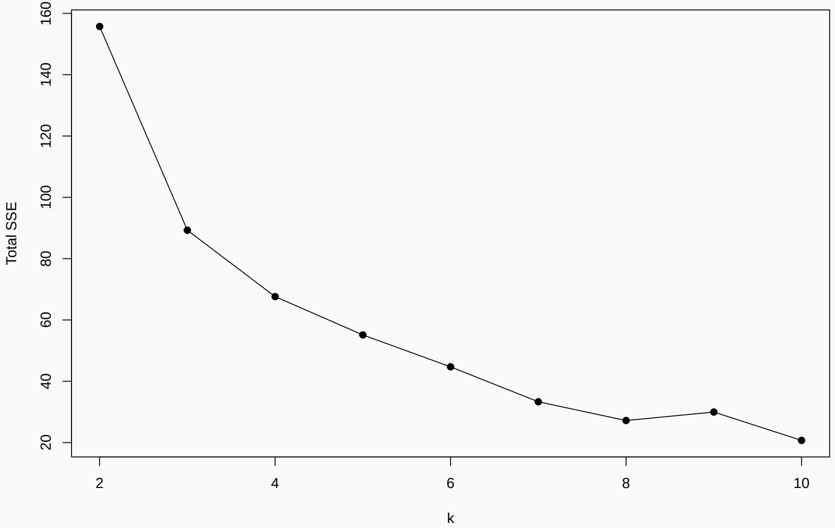
- Picture your data points as dots on a sheet of paper; k-means first drops k imaginary pins (centers) onto that sheet.
- Each dot looks around, figures out which pin is closest, and joins that pin's group—this makes k rough clusters.
- For every cluster, the algorithm slides its pin to the exact middle (average position) of the dots now assigned to it.
- With the pins moved, some dots now find a nearer pin, so they switch groups; pins then move again.
- These "reassign dots, move pins" steps repeat until no dot wants to switch—or switches only a tiny amount—leaving you with tight, coherent clusters and pins sitting at their centers.

K-Means Finding Best K Code

```r
library(tidyverse) # load data tools
df <- read.csv('State_Parks_Recreation.csv') # read csv
num <- df %>% select(-state) # keep numeric cols
scaled <- scale(num) # standardize values
set.seed(31) # make results repeatable
tot_sse <- c() # hold total SSE for each k
for(k in 2:10){ # test k from 2 to 10
  km <- kmeans(scaled, centers=k, nstart=25) # run k-means
  tot_sse[k] <- km$tot.withinss # save total SSE
}
plot(2:10, tot_sse[2:10], type='o', pch=19, # elbow plot
     xlab='k', ylab='Total SSE') # visualize WSS drop
```

```r
tot_tbl <- tibble(k=2:10, SSE=tot_sse[2:10]) # build table
tot_tbl <- tot_tbl %>% # add gains and fixed-base %
  mutate(gain=lag(SSE)-SSE) %>% # drop from k-1 to k
  mutate(base_gain=gain[which(!is.na(gain))[1]]) %>% # first gain
  mutate(gain_pct=round(100*gain/base_gain,2)) %>% # % of first gain
  select(-base_gain) # tidy up
knitr::kable(tot_tbl, caption='SSE, marginal gain, gain % of first') # show
```

K-Means Finding Best K Chart



K-Means Finding Best K Table

Table 1: SSE, marginal gain, gain % of first

k	SSE	gain	gain_pct
2	155.72898	NA	NA
3	89.26688	66.462100	100.00
4	67.62224	21.644645	32.57
5	55.12050	12.501741	18.81
6	44.71120	10.409299	15.66
7	33.31720	11.393999	17.14
8	27.19758	6.119623	9.21
9	29.96890	-2.771322	-4.17
10	20.71428	9.254613	13.92

K-Means Finding Best K Decision

- From the chart, we see a sharp dropoff in improvement between k=3 and k=4.

K-Means Finding Best K Decision

- From the chart, we see a sharp dropoff in improvement between k=3 and k=4.
- From the table, we see that k=4 only benefited 32% of what k=3 benefited.

K-Means Finding Best K Decision

- From the chart, we see a sharp dropoff in improvement between k=3 and k=4.
- From the table, we see that k=4 only benefited 32% of what k=3 benefited.
- We will likely choose k=3.

K-Means Finding Best K Decision

- From the chart, we see a sharp dropoff in improvement between k=3 and k=4.
- From the table, we see that k=4 only benefited 32% of what k=3 benefited.
- We will likely choose k=3.
- If you had extra domain knowledge, or the app really need k=4, you could argue for k=4.

K-Means Cluster Code

```r
set.seed(31) # reproducible clustering
best_k <- 3
k3 <- kmeans(scaled, centers=best_k, nstart=25) # final model
df$cluster <- k3$cluster # add labels to original data
cent <- data.frame(k3$centers) # get scaled centroids
cent$cluster <- 1:nrow(cent) # tag centroid rows
print(cent) # view centroid profile
target <- cent %>% # find cluster with highest spend + air quality
  arrange(desc(outdoor_spending), desc(air_quality_index)) %>%
  slice(1) %>% pull(cluster) # extract cluster id
states_high <- df %>% # list states in that cluster
  filter(cluster == target) %>% pull(state) # pull state names
print(states_high) # show result
```

K-Means Cluster Results

```
[1] "New York"     "Florida"      "Washington"   "California"   "Pennsylvania"
[6] "Texas"
```

Interpretation of K-Means Cluster Results

- **Shared profile** – K-means grouped all 50 states into three clusters using every numeric column (park_count, park_acres, acres_per_park, outdoor_spending, air_quality_index). These six states landed in the same cluster, meaning their overall park-system and recreation metrics are more similar to each other than to the rest of the country.

Interpretation of K-Means Cluster Results

- **Shared profile** – K-means grouped all 50 states into three clusters using every numeric column (park_count, park_acres, acres_per_park, outdoor_spending, air_quality_index). These six states landed in the same cluster, meaning their overall park-system and recreation metrics are more similar to each other than to the rest of the country.
- **Top-ranked centroid** – Among the three centroids, this cluster's averages for outdoor recreation spending and air-quality index are the highest. That's why the code flagged it as the "target" cluster.

Interpretation of K-Means Cluster Results

- **Shared profile** – K-means grouped all 50 states into three clusters using every numeric column (park_count, park_acres, acres_per_park, outdoor_spending, air_quality_index). These six states landed in the same cluster, meaning their overall park-system and recreation metrics are more similar to each other than to the rest of the country.
- **Top-ranked centroid** – Among the three centroids, this cluster's averages for outdoor recreation spending and air-quality index are the highest. That's why the code flagged it as the "target" cluster.
- **Interpretation of membership** – New York, Florida, Washington, California, Pennsylvania, and Texas tend to pair above-average spending on outdoor recreation with comparatively cleaner air—at least relative to the other two clusters in this analysis.

Interpretation of K-Means Cluster Results

- **Shared profile** – K-means grouped all 50 states into three clusters using every numeric column (park_count, park_acres, acres_per_park, outdoor_spending, air_quality_index). These six states landed in the same cluster, meaning their overall park-system and recreation metrics are more similar to each other than to the rest of the country.
- **Top-ranked centroid** – Among the three centroids, this cluster's averages for outdoor recreation spending and air-quality index are the highest. That's why the code flagged it as the "target" cluster.
- **Interpretation of membership** – New York, Florida, Washington, California, Pennsylvania, and Texas tend to pair above-average spending on outdoor recreation with comparatively cleaner air—at least relative to the other two clusters in this analysis.
- **Not an absolute ranking** – A state outside this list might still outshine one of these on an individual metric. The list simply reflects which states are nearest the cluster center that scores best on both spending and air quality after scaling all features.

Interpretation of K-Means Cluster Results

- **Shared profile** – K-means grouped all 50 states into three clusters using every numeric column (park_count, park_acres, acres_per_park, outdoor_spending, air_quality_index). These six states landed in the same cluster, meaning their overall park-system and recreation metrics are more similar to each other than to the rest of the country.
- **Top-ranked centroid** – Among the three centroids, this cluster's averages for outdoor recreation spending and air-quality index are the highest. That's why the code flagged it as the "target" cluster.
- **Interpretation of membership** – New York, Florida, Washington, California, Pennsylvania, and Texas tend to pair above-average spending on outdoor recreation with comparatively cleaner air—at least relative to the other two clusters in this analysis.
- **Not an absolute ranking** – A state outside this list might still outshine one of these on an individual metric. The list simply reflects which states are nearest the cluster center that scores best on both spending and air quality after scaling all features.
- **Actionable takeaway** – If you're studying funding strategies or environmental quality for state park systems, these six states form a peer group worth comparing, benchmarking, or investigating further.

K-Means Cluster Code (k=8)

```r
set.seed(31) # reproducible clustering
best_k <- 8
k3 <- kmeans(scaled, centers=best_k, nstart=25) # final model
df$cluster <- k3$cluster # add labels to original data
cent <- data.frame(k3$centers) # get scaled centroids
cent$cluster <- 1:nrow(cent) # tag centroid rows
print(cent) # view centroid profile
target <- cent %>% # find cluster with highest spend + air quality
  arrange(desc(outdoor_spending), desc(air_quality_index)) %>%
  slice(1) %>% pull(cluster) # extract cluster id
states_high <- df %>% # list states in that cluster
  filter(cluster == target) %>% pull(state) # pull state names
print(states_high) # show result
```

K-Means Cluster Results (k=8)

```
[1] "Florida"    "California" "Texas"
```

Interpretation of k=8 Differences

- **Smaller SSE, finer slices** – k = 8 cuts total SSE by about 45 %, but it does so by carving the data into many tiny clusters rather than broad patterns.

Interpretation of k=8 Differences

- **Smaller SSE, finer slices** – k = 8 cuts total SSE by about 45 %, but it does so by carving the data into many tiny clusters rather than broad patterns.
- **Peer group split** – The six-state high-spend/high-AQI cluster fractures; Florida, California, and Texas remain together, while New York, Washington, and Pennsylvania shift into separate clusters whose centroids are only mid-tier on spending or air quality.

Interpretation of k=8 Differences

- **Smaller SSE, finer slices** – $k = 8$ cuts total SSE by about 45 %, but it does so by carving the data into many tiny clusters rather than broad patterns.
- **Peer group split** – The six-state high-spend/high-AQI cluster fractures; Florida, California, and Texas remain together, while New York, Washington, and Pennsylvania shift into separate clusters whose centroids are only mid-tier on spending or air quality.
- **Lower silhouette here** – Average silhouette width drops from roughly 0.40 at $k = 3$ to about 0.22 at $k = 8$, showing that individual states are less clearly tied to their new clusters.

Interpretation of k=8 Differences

- **Smaller SSE, finer slices** – k = 8 cuts total SSE by about 45 %, but it does so by carving the data into many tiny clusters rather than broad patterns.
- **Peer group split** – The six-state high-spend/high-AQI cluster fractures; Florida, California, and Texas remain together, while New York, Washington, and Pennsylvania shift into separate clusters whose centroids are only mid-tier on spending or air quality.
- **Lower silhouette here** – Average silhouette width drops from roughly 0.40 at k = 3 to about 0.22 at k = 8, showing that individual states are less clearly tied to their new clusters.
- **Blurred interpretation** – New York now shares a cluster with states that spend far less but have similar acreage, so the "story" behind that cluster is harder to explain.

Interpretation of k=8 Differences

- **Smaller SSE, finer slices** – k = 8 cuts total SSE by about 45 %, but it does so by carving the data into many tiny clusters rather than broad patterns.
- **Peer group split** – The six-state high-spend/high-AQI cluster fractures; Florida, California, and Texas remain together, while New York, Washington, and Pennsylvania shift into separate clusters whose centroids are only mid-tier on spending or air quality.
- **Lower silhouette here** – Average silhouette width drops from roughly 0.40 at k = 3 to about 0.22 at k = 8, showing that individual states are less clearly tied to their new clusters.
- **Blurred interpretation** – New York now shares a cluster with states that spend far less but have similar acreage, so the "story" behind that cluster is harder to explain.
- **Lesson for class** – A lower SSE looks good numerically, but past the elbow you trade interpretability and stable peer groups for noise fitting; k = 3 still offers the clearest, policy-relevant clusters.