STAT 479 HW4

Caitlin Bolz

```r
library(knitr)
library(readr)
library(ggplot2)
library(janeaustenr)
library(tidytext)
library(dplyr)
library(MASS)
library(tibble)
library(purrr)
```

(1) Polio incidence

In this problem, we will use a heatmap to visualize a large collection of time series. The data, prepared by the Tycho Project, contain weekly counts of new polio cases in the United States, starting as early as 1912.

Part A

Pivot the raw dataset so that states appear along rows and weeks appear along columns. Fill weeks that don't have any cases with 0's.

```r
polio <- read_csv("https://uwmadison.box.com/shared/static/nm7yku4y9q7ylvz5kbxya3ouj2njd0x6.csv")
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   state = col_character(),
##   period_start_date = col_date(format = ""),
##   cases = col_double()
## )
```

Part B

Use the `superheat` package to make a heatmap of the data from (a). Have the color of each tile represent `log(1 + cases)`, rather than the raw counts. Reorder the states using a hierarchical clustering by setting `pretty.order.rows= TRUE`.

Part C

Supplement the view from part (b) with a barchart showing the US total incidence during every given week. Interpret the resulting visualization. *Hint: use the `yt` argument of superheat.*

Part D

Describe types of annotation would improve the informativeness of the plot made in part (c). Also, describe one advantage and one disadvantage of visualizing a collection of time series as a heatmap, instead of as a collection of line plots.

(2) Silhouette statistics simulation

This problem uses simulation to build intuition about silhouette statistics. The function below simulates a mixture of three Gaussians, with means evenly spaced out between `(start, 0)` and `(end, 0)`. See Figure 2 for an example simulated dataset. We will investigate what happens to the silhouette statistics for each point as the three clusters are made to gradually overlap.

```r
mixture_data <- function(start = 0, end = 10, K = 3, n = 100) {
  mu <- cbind(seq(start, end, length.out = K), 0)
  map_dfr(1:K, ~ mvrnorm(n, mu[., ], diag(2)) %>% as_tibble())
}

ggplot(mixture_data()) +
  geom_point(aes(x = V1, y = V2)) +
  coord_fixed() +
  labs(x = "x", y = "y") +
  theme(
    panel.background = element_rect(fill = "#f7f7f7"),
    panel.border = element_rect(fill = NA, color = "#0c0c0c", size = 0.6),
    panel.grid.minor = element_blank()
  )
```

```
## Warning: The ‘x‘ argument of ‘as_tibble.matrix()‘ must have unique column names if ‘.name_repair‘ is
## Using compatibility ‘.name_repair‘.
## This warning is displayed once every 8 hours.
## Call ‘lifecycle::last_warnings()‘ to see where this warning was generated.
```

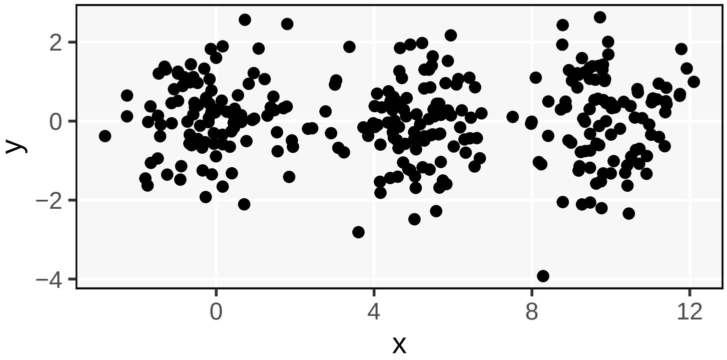

Figure 1: An example simulated dataset from problem 2.

Part A

Simulate versions of these data where the `end` argument is decreased from 10 to 0.

Part B

Write a function that performs K-means and computes the silhouette statistic given any one of the datasets generated in part (a). Use this function to compute the silhouette statistics for each point in the simulated datasets.

Part C

Visualize the silhouette statistics from part (b) overlaid onto the simulated data. Discuss the results.

(3) Taxi trips

In this problem, we will use hierarchical clustering to find typical taxi trip trajectories in Porto, Portugal. The data are a subset from the the ECML / PKDD 2015 Challenge – the link provides a complete data dictionary. We have pre-processed it into two formats. The first wide includes each taxi trip on its own row, with latitude and longitude coordinates along the journey given as separate columns (`x_0, y_0` is the origin of the trip and `x_15`, `y_15` is the destination). The second long format spreads each point of the journey into a separate row.

```
trips_wide <- read_csv("https://uwmadison.box.com/shared/static/cv0lij4d9gn3s8m2k98t2ue34oz6sbj5.csv")
trips_long <- read_csv("https://uwmadison.box.com/shared/static/098cjaetm8vy0mufq21mc8i9nue2rr2b.csv")
```

Part A

Filter the `long` form of the data down to trip `1389986517620000047` and plot its trajectory as a sequence of points.

Part B

We could hierarchically cluster rows in either the `wide` or the `long` format datasets. How would the interpretation of the results differ between the two approaches?

Part C

Compute a hierarchical clustering of the `wide` format data, using only the columns starting with `x` or `y` as features.

Part D

Cut the hierarchical clustering tree so that 8 clusters are produced. Visualize the trajectories of the taxi trips either colored or faceted by their cluster.

(4) Food nutrients

This problem will use PCA to provide a low-dimensional view of a 14-dimensional nutritional facts dataset. The data were originally curated by the USDA and are regularly used in visualization studies.

```
nutrients <- read_csv("https://uwmadison.box.com/shared/static/nmgouzobq5367aex45pnbzgkhm7sur63.csv")
```

Part A

Define a tidymodels `recipe` that normalizes all nutrient features and specifies that PCA should be performed.

Part B

Visualize the top 6 principal components. What types of food do you expect to have low or high values for PC1 or PC2?

Part C

Compute the average value of PC2 within each category of the `group` column. Give the names of the groups sorted by this average.

Part D

Visualize the scores of each food item with respect to the first two principal components. Facet the visualization according to the `group` column, and sort the facets according to the results of part (c). How does the result compare with your guess from part (b)?

(5) Modeling topics in *Pride and Prejudice*

This problem uses LDA to analyze the full text of *Pride and Prejudice*. The code below creates two R objects, `paragraph` and `dtm`. `paragraph` is a data.frame whose rows are paragraphs from the book. We've filtered very short paragraphs; e.g., from dialogue. `dtm` is a `DocumentTermMatrix` containing word counts across the same paragraphs – the i^{th} row of `dtm` corresponds to the i^{th} row of `paragraph`.

Part A

Fit an LDA model to `dtm` using 6 topics. Set the seed by using the argument `control = list(seed = 479)` to remove any randomness in the result.

Part B

Visualize the top 30 words within each of the fitted topics. Specifically, create a faceted bar chart where the heights of the bars correspond to word probabilities and the facets correspond to topics. Reorder the bars so that each topic's top words are displayed in order of decreasing probability.

Part C

Find the paragraph that is the purest representative of Topic 2. That is, if γ_{ik} denotes the weight of topic k in paragraph i, then print out paragraph i^* where $i^* = \arg\max_i \gamma_{i2}$. Verify that the at least a few of the words with high probability for this topic appear. Only copy the first sentence into your solution.

(6) Single-Cell Genomics

In this problem, we will apply UMAP to a dataset of Peripheral Blood Mononuclear Cells (PBMC) released by 10X Genomics. The first column, `cell_tag`, gives an identifier for each cell in the dataset. All other columns are molecules that were detected in that cell. For example, CD74 is a molecule often found on the surface of T-cells.

```
pbmc <- read_csv("https://uwmadison.box.com/shared/static/ai539s30rjsw5ke4vxbjrxjaiihq7edk.csv")
```

Part A

Define a tidymodels `recipe` that specifies that UMAP should be performed with the parameters `learn_rate = 0.1` and `neighbors = 5`. There is no need to normalize these data, as they have been normalized in advance using methods tailored to single-cell genomics data.

Part B

Compute the UMAP embeddings across cells. Color points in by their value of the GNLY molecule.

Feedback

a. How much time did you spend on this homework?
b. Which problem did you find most valuable?