Data Scientist Role Play: Profiling and Analyzing the Yelp Dataset Coursera Worksheet

This is a 2-part assignment. In the first part, you are asked a series of questions that will help you profile and understand the data just like a data scientist would. For this first part of the assignment, you will be assessed both on the correctness of your findings, as well as the code you used to arrive at your answer. You will be graded on how easy your code is to read, so remember to use proper formatting and comments where necessary.

In the second part of the assignment, you are asked to come up with your own inferences and analysis of the data for a particular research question you want to answer. You will be required to prepare the dataset for the analysis you choose to do. As with the first part, you will be graded, in part, on how easy your code is to read, so use proper formatting and comments to illustrate and communicate your intent as required.

For both parts of this assignment, use this "worksheet." It provides all the questions you are being asked, and your job will be to transfer your answers and SQL coding where indicated into this worksheet so that your peers can review your work. You should be able to use any Text Editor (Windows Notepad, Apple TextEdit, Notepad ++, Sublime Text, etc.) to copy and paste your answers. If you are going to use Word or some other page layout application, just be careful to make sure your answers and code are lined appropriately.
In this case, you may want to save as a PDF to ensure your formatting remains intact for you reviewer.

Part 1: Yelp Dataset Profiling and Understanding

1. Profile the data by finding the total number of records for each of the tables below:

Part 1: Yelp Dataset Profiling and Understanding

1. Profile the data by finding the total number of records for each of the tables below:

i. Attribute table = 10000
ii. Business table = 10000
iii. Category table = 10000
iv. Checkin table = 10000
v. elite_years table = 10000
vi. friend table = 10000
vii. hours table = 10000
viii. photo table = 10000
ix. review table = 10000
x. tip table = 10000
xi. user table = 10000

2. Find the total number of distinct records for each of the keys listed below:

i. Business = 10000 (id)
ii. Hours = 1562 (business_id)
iii. Category = 2643 (business_id)
iv. Attribute = 1115 (business_id)
v. Review = 10000 (id), 8090 (business_id), 9581 (user_id)
vi. Checkin = 493 (business_id)
vii. Photo = 10000 (id), 6493 (business_id)
viii. Tip = 537 (user_id), 3979 (business_id)
ix. User = 10000 (id)
x. Friend = 11 (user_id)
xi. Elite_years = 2780 (user_id)

3. Are there any columns with null values in the Users table? Indicate "yes," or "no."

Answer: No

SQL code used to arrive at answer:

```
         select id, name, review_count, yelping_since,
useful, funny, cool, fans, average_stars,
                 compliment_hot, compliment_more,
compliment_profile, compliment_cute, compliment_list,
                 compliment_note, compliment_plain,
compliment_cool, compliment_funny, compliment_writer,
compliment_photos
         from  user
         where    id is null
                  or name is null
                  or review_count is null
                  or yelping_since is null
                  or useful is null
                  or funny is null
                  or cool is null
                  or fans is null
                  or average_stars is null
                  or compliment_hot is null
                  or compliment_more is null
                  or compliment_profile is null
                  or compliment_cute is null
                  or compliment_list is null
                  or compliment_note is null
                  or compliment_plain is null
                  or compliment_cool is null
                  or compliment_funny is null
                  or compliment_writer is null
                  or compliment_photos is null
```

4. Find the minimum, maximum, and average value for the following fields:

i. Table: Review, Column: Stars

min: 1   max: 5   avg: 3.7082

ii. Table: Business, Column: Stars
    min: 1.0 max: 5.0 avg: 3.6549

iii. Table: Tip, Column: Likes
    min: 0   max: 2   avg: 0.0144

iv. Table: Checkin, Column: Count
    min: 1   max: 53   avg: 1.9414

v. Table: User, Column: Review_count
    min: 0   max: 2000 avg: 24.2995

5. List the cities with the most reviews in descending order:

SQL code used to arrive at answer:

```
select city, sum(review_count)
from business
group by city
order by sum(review_count) desc
```

Copy and Paste the Result Below:

```
+-----------------+-------------------+
| city            | sum(review_count) |
+-----------------+-------------------+
| Las Vegas       |             82854 |
| Phoenix         |             34503 |
| Toronto         |             24113 |
| Scottsdale      |             20614 |
| Charlotte       |             12523 |
| Henderson       |             10871 |
| Tempe           |             10504 |
| Pittsburgh      |              9798 |
| MontrÃ©al       |              9448 |
| Chandler        |              8112 |
| Mesa            |              6875 |
| Gilbert         |              6380 |
| Cleveland       |              5593 |
| Madison         |              5265 |
| Glendale        |              4406 |
| Mississauga     |              3814 |
| Edinburgh       |              2792 |
| Peoria          |              2624 |
| North Las Vegas |              2438 |
| Markham         |              2352 |
| Champaign       |              2029 |
| Stuttgart       |              1849 |
| Surprise        |              1520 |
| Lakewood        |              1465 |
| Goodyear        |              1155 |
+-----------------+-------------------+
```

6. Find the distribution of star ratings to the business in the following cities:

     i. Avon

         SQL code used to arrive at answer:

```
              select stars as [Star Rating], count(stars) as
[Count]
              from business b
              where city = 'Avon'
              group by stars
```

         Copy and Paste the Resulting Table Below (2 columns - star rating and count):

```
              +-------------+-------+
              | Star Rating | Count |
              +-------------+-------+
              |         1.5 |     1 |
              |         2.5 |     2 |
              |         3.5 |     3 |
              |         4.0 |     2 |
              |         4.5 |     1 |
              |         5.0 |     1 |
              +-------------+-------+
```

ii. Beachwood

SQL code used to arrive at answer:

```
     select stars as [Star Rating], count(stars) as
[Count]
     from business b
     where city = 'Beachwood'
     group by stars
```

Copy and Paste the Resulting Table Below (2 columns - star rating and count):

```
+-------------+-------+
| Star Rating | Count |
+-------------+-------+
|         2.0 |     1 |
|         2.5 |     1 |
|         3.0 |     2 |
|         3.5 |     2 |
|         4.0 |     1 |
|         4.5 |     2 |
|         5.0 |     5 |
+-------------+-------+
```

7. Find the top 3 users based on their total number of reviews:

SQL code used to arrive at answer:

```
select name, review_count
from user
order by review_count desc
limit 3
```

Copy and Paste the Result Below:

```
+--------+--------------+
| name   | review_count |
+--------+--------------+
| Gerald |         2000 |
| Sara   |         1629 |
| Yuri   |         1339 |
+--------+--------------+
```

8. Does posing more reviews correlate with more fans?

- No

Please explain your findings and interpretation of the results:

- N/A

SQL code:

```
select name, review_count, fans
from user
order by fans desc
limit 10
```

Results:

```
+-----------+--------------+------+
| name      | review_count | fans |
+-----------+--------------+------+
| Amy       |          609 |  503 |
| Mimi      |          968 |  497 |
| Harald    |         1153 |  311 |
| Gerald    |         2000 |  253 |
| Christine |          930 |  173 |
| Lisa      |          813 |  159 |
| Cat       |          377 |  133 |
| William   |         1215 |  126 |
| Fran      |          862 |  124 |
| Lissa     |          834 |  120 |
+-----------+--------------+------+
```

9. Are there more reviews with the word "love" or with the word "hate" in them?

Answer: more reviews with the word "love"

SQL code used to arrive at answer:

```
select (select count(text)
          from review
          where text like "%love%") as  love_text,

        (select count(text)
          from review
          where text like "%hate%") as hate_text
```

Results:

```
+-----------+-----------+
| love_text | hate_text |
+-----------+-----------+
|      1780 |       232 |
+-----------+-----------+
```

OR:

```
SELECT 'love' Word, COUNT(text) [Total Count]
FROM review
WHERE text LIKE '%love%'
UNION
SELECT 'hate' Word, COUNT(text) [Total Count]
FROM review
WHERE text LIKE '%hate%'
```

```
+------+-------------+
| Word | Total Count |
+------+-------------+
| hate |         232 |
| love |        1780 |
+------+-------------+
```

10. Find the top 10 users with the most fans:

SQL code used to arrive at answer:

```
          select name, fans
          from user
          order by fans desc
          limit 10
```

Copy and Paste the Result Below:

```
          +-----------+------+
          | name      | fans |
          +-----------+------+
          | Amy       |  503 |
          | Mimi      |  497 |
          | Harald    |  311 |
          | Gerald    |  253 |
          | Christine |  173 |
          | Lisa      |  159 |
          | Cat       |  133 |
          | William   |  126 |
          | Fran      |  124 |
          | Lissa     |  120 |
          +-----------+------+
```

11. Is there a strong correlation between having a high number of fans and being listed
     as "useful" or "funny?"

SQL code used to arrive at answer:

```
          select name, fans, useful, funny
          from user
          order by fans desc, useful desc, funny desc
          limit 20
```

Copy and Paste the Result Below:

```
          +-----------+------+--------+--------+
          | name      | fans | useful |  funny |
          +-----------+------+--------+--------+
          | Amy       |  503 |   3226 |   2554 |
          | Mimi      |  497 |    257 |    138 |
          | Harald    |  311 | 122921 | 122419 |
          | Gerald    |  253 |  17524 |   2324 |
          | Christine |  173 |   4834 |   6646 |
          | Lisa      |  159 |     48 |     13 |
          | Cat       |  133 |   1062 |    672 |
          | William   |  126 |   9363 |   9361 |
          | Fran      |  124 |   9851 |   7606 |
          | Lissa     |  120 |    455 |    150 |
          | Mark      |  115 |   4008 |    570 |
          | Tiffany   |  111 |   1366 |    984 |
          | bernice   |  105 |    120 |    112 |
          | Roanna    |  104 |   2995 |   1188 |
          | .Hon      |  101 |   7850 |   5851 |
          | Angela    |  101 |    158 |    164 |
          | Ben       |   96 |   1180 |   1155 |
          | Linda     |   89 |   3177 |   2736 |
          | Christina |   85 |    158 |     34 |
          | Jessica   |   84 |   2161 |   2091 |
          +-----------+------+--------+--------+
```

Part 2: Inferences and Analysis

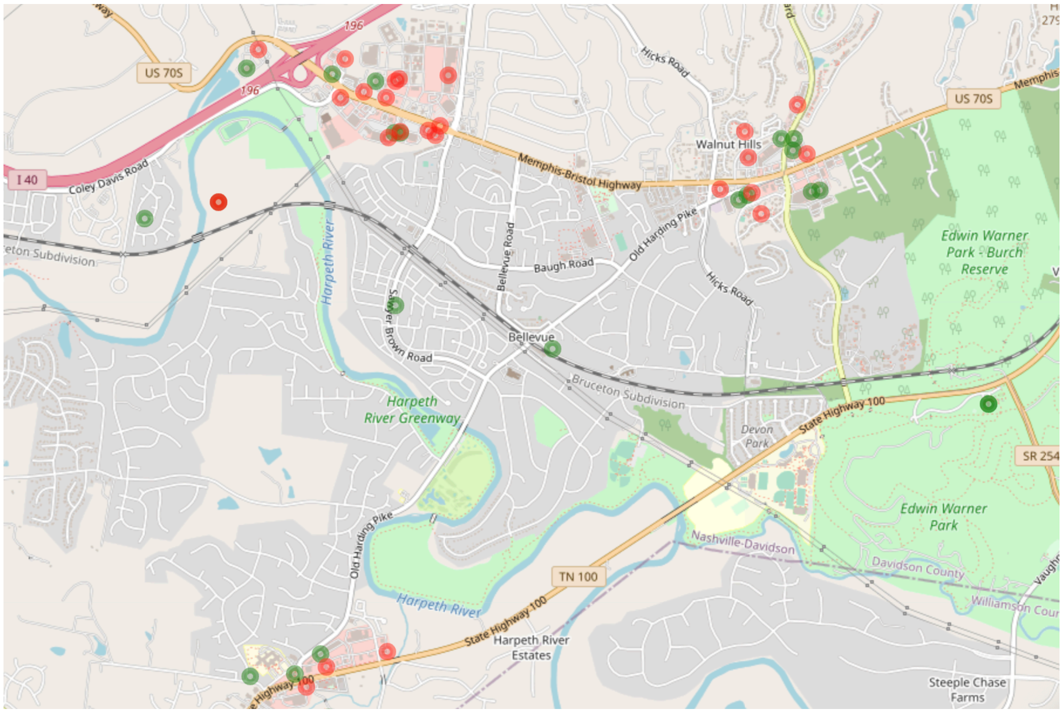
1. Pick one city and category of your choice and group the businesses in that city or category by their overall star rating. Compare the businesses with 2-3 stars to the businesses with 4-5 stars and answer the following questions. Include your code.

Nashville TN Zip Code 37221

i. Do the two groups you chose to analyze have a different distribution of hours? NO

ii. Do the two groups you chose to analyze have a different number of reviews? YES - More for 4-5 star reviews

iii. Are you able to infer anything from the location data provided between these two groups? Explain. No. The locations are very tightly clustered in shopping regions. Red are 2-3 star rated. Green are 4-5 star rated.



Caption

Executive Summary:

The TNBus dataset contains information on 12,056 businesses in Tennessee, including variables such as name, address, city, state, zip code, latitude, longitude, rating, and review count. The data was extracted from the Yelp Academic Dataset To better understand the differences between top-rated (4-5 stars) and bottom-rated (2-3 stars) businesses, we conducted an analysis focusing on the zip code 37221.

I first extracted businesses located in the 37221 zip code and then categorized them into two groups: the top 25 businesses and the bottom 25 businesses based on their ratings. The top 25 businesses had an average rating of 4-5 stars, while the bottom 25 businesses had an average rating of 2-3 stars.

Upon examining the mean values of the numeric columns for both groups, I found that top-rated businesses tended to have a higher number of reviews on average. This could be an indication of better customer engagement and overall satisfaction. The comparison of mean longitude and latitude values did not reveal any significant geographic clustering of top or bottom-rated businesses within the 37221 zip code.

To further explore differences between the top and bottom-rated businesses, we examined the textual data from the `categories` column. We identified the top 10 words used to describe each group, which provided insights into the type of businesses and services offered by each group.

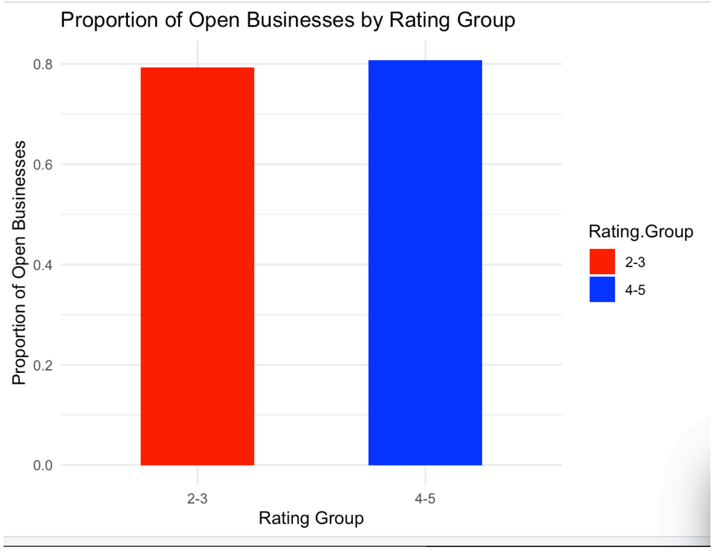
In conclusion, our analysis of the TNBus dataset has shown that there are notable differences between top and bottom-rated businesses in the 37221 zip code, particularly in terms of review count and types of businesses. However, it is important to note that other factors not included in the dataset, such as specific attributes or services, could also impact business ratings. Further analysis of these additional factors could provide a more comprehensive understanding of the differences between top and bottom-rated businesses.

Find the GitHub Code and More Information Here https://github.com/BethMara/YelpUsers

```
# Import necessary libraries
library(tidyverse) library(leaflet) # Import Business and TNBus datasets
Business <- read.csv("yelp_business.csv") TNBus <- read.csv("tnbus.csv") # Rename the column names in TNBus
to match those in Business
colnames(TNBus) [1] <- "ID" colnames(TNBus) [2] <- "name" colnames(TNBus) [6] <- "zip" colnames(TNBus) [7] <-
"lat" colnames(TNBus) [8] <- "long" # Join the Business and TNBus datasets based on ID
merged_data <- merge(Business, TNBus, by = "ID") # Filter merged_data to only include businesses in zip code
37221
zip_37221 <- merged_data %>% filter(zip == "37221") # Create a new dataframe with only the top 25 rated
businesses in zip code 37221
top_25 <- zip_37221 %>% arrange(
  desc(rating)
) %>% # sort by descending rating
select
  (
    name, rating, review_count, attributes,
    open
  ) %>% # select specific columns
  head(25) # only take top 25 businesses

# Display the structure of the dataframe
str(TNBus)

# Provide a summary of the dataframe
summary(TNBus)

# Return the column names of the dataframe
names(TNBus)

# Open the dataframe in a spreadsheet-like view within RStudio
View(TNBus)
```

```
# Open the dataframe in a spreadsheet-like view within RStudio
> View(TNBus)
>
> # Create separate data frames for businesses with ratings 2-3 and 4-5
> TNBus_2to3 <- subset(TNBus, rating >= 2 & rating <= 3)
> TNBus_4to5 <- subset(TNBus, rating >= 4 & rating <= 5)
>
> # Calculate the proportion of open businesses in each group
> open_2to3 <- sum(TNBus_2to3$open == "1") / nrow(TNBus_2to3)
> open_4to5 <- sum(TNBus_4to5$open == "1") / nrow(TNBus_4to5)
>
> # Create a summary table
> summary_table <- data.frame(Rating.Group = c("2-3", "4-5"),
+                             Proportion.Open = c(open_2to3, open_4to5))
>
> # Print the summary table
> print(summary_table)
```

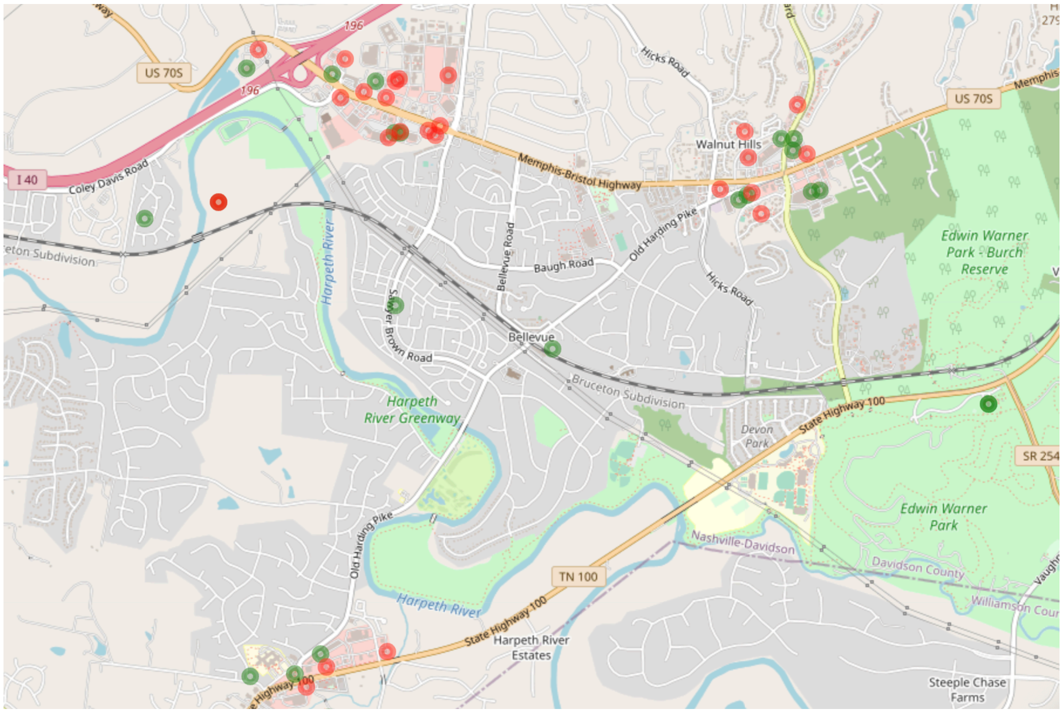
```
  Rating.Group Proportion.Open
1          2-3       0.7936508
2          4-5       0.8072045
>
> # Install ggplot2 if not already installed
> if (!requireNamespace("ggplot2", quietly = TRUE)) {
+     install.packages("ggplot2")
+ }
>
> # Load ggplot2
> library(ggplot2)
>
> # Create a histogram
> ggplot(summary_table, aes(x = Rating.Group, y = Proportion.Open, fill = Rating.Group)) +
+     geom_bar(stat = "identity", width = 0.5) +
+     scale_fill_manual(values = c("2-3" = "red", "4-5" = "blue")) +
+     labs(title = "Proportion of Open Businesses by Rating Group",
+          x = "Rating Group",
```



```
+      There is negligible difference between Low and High rated businesses remaining open.
+      theme_minimal()

> # Create a new dataframe with businesses in zip code 37221
> zip_37221 <- subset(TNBus, zip == "37221")
>
> # Order the dataframe by rating, ascending
> zip_37221_ordered <- zip_37221[order(zip_37221$rating),]
>
> # Select the bottom 25 rated businesses
> bottom_25 <- head(zip_37221_ordered, 25)
> print(bottom_25)

> # Print the names, rating, and the number of ratings for the bottom 25 businesses
> print(bottom_25[, c("name", "rating", "review_count")])
                                     name rating review_count
6787                          Captain D's    1.0            7
8155                           Eco Movers    1.0           13
10694                           AT&T Store    1.0            5
2681                    Vue at Warner Park    1.5           13
```

```
4606                              Sonic Drive-In 1.5          11
5256                               AT&T Internet 1.5          30
6745                                   Pizza Hut 1.5          13
9580                                         KFC 1.5          19
9621                                   Bar Louie 1.5           5
10799                                  Pizza Hut 1.5          24
11679                        Southeast Financial 1.5          55
11973                                    Shoneys 1.5          14
2277  Microtel Inn & Suites by Wyndham Nashville 2.0          22
2342                                     Chili's 2.0          91
2825                                   Walgreens 2.0          17
3027                           Sears Auto Center 2.0           6
3069                           Papa John's Pizza 2.0          44
3559                              Baskin Robbins 2.0          15
5107                                      Arby's 2.0          17
5432                      AMC Classic Bellevue 8 2.0          29
5871                            Harpeth Cleaners 2.0           7
6504                                  Applebee's 2.0          18
6779          Nail Time & Spa By Hollywood Nails 2.0          34
7084                                    Michaels 2.0           6
7415                                     Wendy's 2.0          21

> # Print the names, rating, and the number of ratings for the top 25 businesses
> print(top_25[, c("name", "rating", "review_count")])
                                                  name rating
848                             Nashville Pet Products    5.0
1745                            Franklin Juice Company    5.0
3079                                 The Pilates Place    5.0
5910                                    Bedzzz Express    5.0
6100  Jeanette Wirz Permanent Cosmetics & Microblading    5.0
6264                            Neko Press Art Studios    5.0
7323                                           MyEyeDr    5.0
9867                             Bellevue 1st Plumbers    5.0
9952                  Beautiful eyebrows threading spa    5.0
10723                       Dental Partners - Bellevue    5.0
10953                            Bellevue Coin Laundry    5.0
11120                                     Warner Parks    5.0
11434                              JB Custom Tailoring    5.0
11786                                 Nashville Smiles    5.0
11827                                  The Vapor Route    5.0
168                                 Red Spirits & Wine    4.5
212                                 Nashville Hypnosis    4.5
572                                        Royal Range    4.5
675                            Iroquois Wine & Spirits    4.5
751                                  Percy Warner Park    4.5
1463                          Workout Anytime Bellevue    4.5
1604                                 Edwin Warner Park    4.5
1667                           Sakura Japanese Cuisine    4.5
1851                                   Bellevue Smiles    4.5
2027                 Harpeth Valley Animal Hospital PC    4.5
      review_count
848             19
1745             5
3079             6
5910             7
6100            10
6264             7
7323             6
```

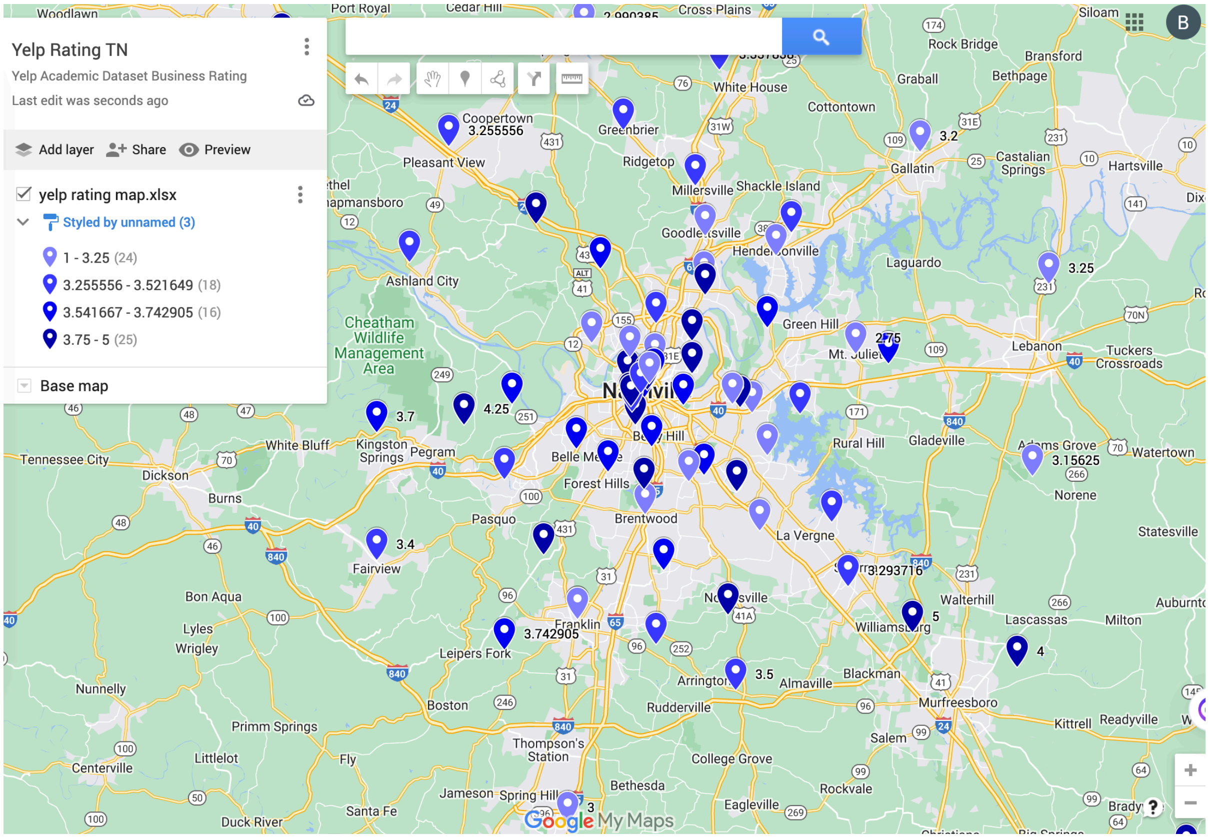
```
9867             6
9952             7
10723           10
10953           26
11120            7
11434           21
11786           13
11827            9
168             62
212              6
572             61
675             16
751            112
1463            11
1604            45
1667           114
1851             5
2027            20
>


>  # Install and load the leaflet package
> if (!requireNamespace("leaflet", quietly = TRUE)) {
+     install.packages("leaflet")
+ }
> library(leaflet)
>
> # Create the map
> map <- leaflet() %>%
+     addTiles() %>%
+     addCircleMarkers(
+         data = top_25,
+         lng = ~as.numeric(long),
+         lat = ~as.numeric(lat),
+         color = "green",
+         popup = ~name,
+         label = ~name,
+         radius = 5
+     ) %>%
+     addCircleMarkers(
+         data = bottom_25,
+         lng = ~as.numeric(long),
+         lat = ~as.numeric(lat),
+         color = "red",
+         popup = ~name,
+         label = ~name,
+         radius = 5
+     )
>
> # Print the map
> map
> > # Calculate means for numeric columns in top_25 and bottom_25 dataframes
> top_means <- top_25 %>%
+     summarize(
+         mean_rating = mean(rating),
+         mean_review_count = mean(as.numeric(review_count)),
+         mean_longitude = mean(as.numeric(longitude)),
+         mean_latitude = mean(as.numeric(latitude))
```



Caption

```
+     )
>
> bottom_means <- bottom_25 %>%
+     summarize(
+         mean_rating = mean(rating),
+         mean_review_count = mean(as.numeric(review_count)),
+         mean_longitude = mean(as.numeric(longitude)),
+         mean_latitude = mean(as.numeric(latitude))
+     )
>
> # Combine the means into a single table
> mean_comparison <- bind_rows(
+     mutate(top_means, group = "Top 25"),
+     mutate(bottom_means, group = "Bottom 25")
+ )
>
> # Print the table
> mean_comparison
  mean_rating mean_review_count mean_longitude mean_latitude
1         4.8             24.44      -86.93059      36.07157
2         1.7             21.44      -86.94376      36.07325
      group
1    Top 25
2 Bottom 25
```

For a bit of extra fun, I also made a Google Map of the businesses in TN that were ranked 2-3 stars vs 4-5 stars.



Yelp Rating TN

Yelp Academic Dataset Business Rating

Last edit was seconds ago

Add layer · Share · Preview

☑ yelp rating map.xlsx

Styled by unnamed (3)

- 1 - 3.25 (24)
- 3.255556 - 3.521649 (18)
- 3.541667 - 3.742905 (16)
- 3.75 - 5 (25)

Base map

Interactive Map https://www.google.com/maps/d/u/0/edit?mid=1jHg1D3pN-3ar7RcQOrAFMUITsDGHL94&usp=sharing

```
This is the same code run thru SQLFormatter which some people
may find easier to read

# Import necessary libraries
library(tidyverse) library(leaflet) # Import Business and TNBus
datasets
Business <- read.csv("yelp_business.csv") TNBus <-
read.csv("tnbus.csv") # Rename the column names in TNBus to
match those in Business
colnames(TNBus) [1] <- "ID" colnames(TNBus) [2] <- "name"
colnames(TNBus) [6] <- "zip" colnames(TNBus) [7] <- "lat"
colnames(TNBus) [8] <- "long" # Join the Business and TNBus
datasets based on ID
merged_data <- merge(Business, TNBus, by = "ID") # Filter
merged_data to only include businesses in zip code 37221
```

```
zip_37221 <- merged_data %>% filter(zip == "37221") # Create a
new dataframe with only the top 25 rated businesses in zip code
37221
top_25 <- zip_37221 %>% arrange(
  desc(rating)
) %>% # sort by descending rating
select
  (
    name, rating, review_count, attributes,
    open
  ) %>% # select specific columns
  head(25) # only take top 25 businesses
  # Display the structure of the dataframe
  str(TNBus) # Provide a summary of the dataframe
  summary(TNBus) # Return the column names of the dataframe
  names(TNBus) # Open the dataframe in a spreadsheet-like view
within RStudio
  View(TNBus) # Open the dataframe in a spreadsheet-like view
within RStudio
  > View(TNBus) > > # Create separate data frames for businesses
with ratings 2-3 and 4-5
  > TNBus_2to3 <- subset(TNBus, rating >= 2 & rating <= 3) >
TNBus_4to5 <- subset(TNBus, rating >= 4 & rating <= 5) > > #
Calculate the proportion of open businesses in each group
  > open_2to3 <- sum(TNBus_2to3$open == "1") / nrow(TNBus_2to3)
> open_4to5 <- sum(TNBus_4to5$open == "1") / nrow(TNBus_4to5) >
> # Create a summary table
  > summary_table <- data.frame(
    Rating.Group = c("2-3", "4-5"),
    + Proportion.Open = c(open_2to3, open_4to5)
  ) > > # Print the summary table
  > print(summary_table) Rating.Group Proportion.Open 1 2 - 3
0.7936508 2 4 - 5 0.8072045 > > # Install ggplot2 if not already
installed
  > if (
    ! requireNamespace("ggplot2", quietly = TRUE)
  ) { + install.packages("ggplot2") + } > > # Load ggplot2
  > library(ggplot2) > > # Create a histogram
  > ggplot(
    summary_table,
    aes(
      x = Rating.Group, y = Proportion.Open,
      fill = Rating.Group
    )
  ) + + geom_bar(stat = "identity", width = 0.5) + +
scale_fill_manual(
    values
      = c("2-3" = "red", "4-5" = "blue")
  ) + + labs(
    title = "Proportion of Open Businesses by Rating Group",
    + x = "Rating Group",
    + There is negligible difference between Low
    and High rated businesses remaining open.theme_minimal() > #
Create a new dataframe with businesses in zip code 37221
    > zip_37221 <- subset(TNBus, zip == "37221") > > # Order the
dataframe by rating, ascending
    > zip_37221_ordered <- zip_37221[order(zip_37221$rating),
    ] > > # Select the bottom 25 rated businesses
    > bottom_25 <- head(zip_37221_ordered, 25) >
print(bottom_25) > # Print the names, rating, and the number of
ratings for the bottom 25 businesses
    > print(
      bottom_25[,
      c("name", "rating", "review_count") ]
    ) name rating review_count 6787 Captain D 's    1.0
7
8155                                      Eco Movers    1.0
13
10694                                     AT&T Store    1.0
5
2681                              Vue at Warner Park    1.5
13
4606                                 Sonic Drive-In    1.5
11
5256                                   AT&T Internet    1.5
30
6745                                       Pizza Hut    1.5
13
9580                                             KFC    1.5
19
9621                                       Bar Louie    1.5
5
10799                                      Pizza Hut    1.5
24
11679                            Southeast Financial    1.5
55
11973                                        Shoneys    1.5
14
2277  Microtel Inn & Suites by Wyndham Nashville    2.0
22
```

```
2342                                     Chili' s 2.0 91 2825
Walgreens 2.0 17 3027 Sears Auto Center 2.0 6 3069 Papa John 's
Pizza    2.0           44
3559                              Baskin Robbins    2.0
15
5107                                      Arby' s 2.0 17 5432
AMC Classic Bellevue 8 2.0 29 5871 Harpeth Cleaners 2.0 7 6504
Applebee 's    2.0          18
6779          Nail Time & Spa By Hollywood Nails    2.0
34
7084                                    Michaels    2.0
6
7415                                     Wendy' s 2.0 21 > #
Print the names, rating, and the number of ratings for the top
25 businesses
    > print(
      top_25[,
      c("name", "rating", "review_count") ]
    ) name rating 848 Nashville Pet Products 5.0 1745 Franklin
Juice Company 5.0 3079 The Pilates Place 5.0 5910 Bedzzz Express
5.0 6100 Jeanette Wirz Permanent Cosmetics & Microblading 5.0
6264 Neko Press Art Studios 5.0 7323 MyEyeDr 5.0 9867 Bellevue
1st Plumbers 5.0 9952 Beautiful eyebrows threading spa 5.0 10723
Dental Partners - Bellevue 5.0 10953 Bellevue Coin Laundry 5.0
11120 Warner Parks 5.0 11434 JB Custom Tailoring 5.0 11786
Nashville Smiles 5.0 11827 The Vapor Route 5.0 168 Red Spirits &
Wine 4.5 212 Nashville Hypnosis 4.5 572 Royal Range 4.5 675
Iroquois Wine & Spirits 4.5 751 Percy Warner Park 4.5 1463
Workout Anytime Bellevue 4.5 1604 Edwin Warner Park 4.5 1667
Sakura Japanese Cuisine 4.5 1851 Bellevue Smiles 4.5 2027
Harpeth Valley Animal Hospital PC 4.5 review_count 848 19 1745 5
3079 6 5910 7 6100 10 6264 7 7323 6 9867 6 9952 7 10723 10 10953
```