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:

i. Attribute table = 10,000
ii. Business table = 10,000
iii. Category table = 10,000
iv. Checkin table = 10,000
v. elite_years table = 10,000
vi. friend table = 10,000
vii. hours table = 10,000
viii. photo table = 10,000
ix. review table = 10,000
x. tip table = 10,000
xi. user table = 10,000

2. Find the total distinct records by either the foreign key or primary key for each table. If two foreign keys are listed in the table, please specify which foreign key.

i. Business = id: 10,000
ii. Hours = business_id: 1,562
iii. Category = business_id: 2,643
iv. Attribute = business_id: 1,115
v. Review = id: 10,000, user_id: 9,581, business_id: 8,090

vi. Checkin = business_id: 493
vii. Photo = id: 10,000, business_id: 6,493
viii. Tip = user_id: 537, business_id: 3,979
ix. User = id: 10,000
x. Friend = user_id: 11
xi. Elite_years = user_id: 2,780

Note: Primary Keys are denoted in the ER-Diagram with a yellow key icon.

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 COUNT(*)
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. For each table and column listed below, display the smallest (minimum), largest (maximum), and average (mean) 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: 2,000 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) AS rev_count
FROM business
GROUP BY city
ORDER BY rev_count DESC
LIMIT 10;
```

Copy and Paste the Result Below:

city	rev_count
Las Vegas	82854
Phoenix	34503
Toronto	24113
Scottsdale	20614
Charlotte	12523
Henderson	10871
Tempe	10504
Pittsburgh	9798
Montréal	9448
Chandler	8112

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,
  sum(review_count)
FROM business
WHERE city = "Avon"
GROUP by stars;
```

Copy and Paste the Resulting Table Below (2 columns â€“ star rating and count):

stars	sum(review_count)
1.5	10
2.5	6
3.5	88
4.0	21
4.5	31
5.0	3

ii. Beachwood

SQL code used to arrive at answer:

```
SELECT
  stars,
  sum(review_count)
FROM business
WHERE city = "Beachwood"
GROUP by stars
```

Copy and Paste the Resulting Table Below (2 columns â€“ star rating and count):

stars	sum(review_count)
2.0	8
2.5	3
3.0	11
3.5	6
4.0	69
4.5	17
5.0	23

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

SQL code used to arrive at answer:

```
SELECT
  id,
  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?

Please explain your findings and interpretation of the results:

- Posting more review does not correlate with more fans. As I run below query on users with top 20 most review posted with fans column:

```
SELECT
  name,
  review_count,
  fans
FROM user
ORDER BY review_count DESC
LIMIT 20;
```

- Hypothetically if there is a correlation the fans number should also go in descending order without specifying in the query but instead the fans number does not go into this pattern. Therefore there should be no correlation between review frequency and fans number

- Result as below:

name	review_count	fans
Gerald	2000	253
Sara	1629	50
Yuri	1339	76
.Hon	1246	101
William	1215	126
Harald	1153	311
eric	1116	16
Roanna	1039	104
Mimi	968	497
Christine	930	173
Ed	904	38
Nicole	864	43
Fran	862	124
Mark	861	115

```
| Christina |          842 |   85 |
| Dominic   |          836 |   37 |
| Lissa     |          834 |  120 |
| Lisa      |          813 |  159 |
| Alison    |          775 |   61 |
| Sui       |          754 |   78 |
+-----------+--------------+------+
```

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

Answer: There are more review of “love” than that of “hate”

SQL code used to arrive at answer:

1.
```
SELECT count(*)
FROM review
WHERE text like "%love%"
```

Result:

```
+----------+
| count(*) |
+----------+
|     1780 |
+----------+
```

2.
```
SELECT count(*)
FROM review
WHERE text like "%hate%"
```

Result:

```
+----------+
| count(*) |
+----------+
|      232 |
+----------+
```

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

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.

i. Do the two groups you chose to analyze have a different distribution of hours?

Yes, businesses with 2-3stars rating have longer hours than that of businesses with 4-5stars rating.

ii. Do the two groups you chose to analyze have a different number of reviews?

Yes, 4-5 stars businesses have more reviews then 2-3 hours businesses in many cases.

iii. Are you able to infer anything from the location data provided between these two groups? Explain.

No since every business are in different postal code.

SQL code used for analysis:

```
SELECT
  b.name,
  h.hours,
  b.postal_code,
  b.stars,
  CASE
    WHEN hours LIKE "%monday%" THEN 1
    WHEN hours LIKE "%tuesday%" THEN 2
    WHEN hours LIKE "%wednesday%" THEN 3
    WHEN hours LIKE "%thursday%" THEN 4
    WHEN hours LIKE "%friday%" THEN 5
```

```
    WHEN hours LIKE "%saturday%" THEN 6
    WHEN hours LIKE "%sunday%" THEN 7
  END AS days,
  CASE
    WHEN B.stars BETWEEN 2.0 AND 3.0 THEN '2-3 stars'
    WHEN B.stars BETWEEN 4.0 AND 5.0 THEN '4-5 stars'
  END AS star_rating
FROM business AS b JOIN hours AS h
  ON b.id = h.business_id
JOIN category AS c
  ON c.business_id = b.id
WHERE (b.city = 'Toronto'AND c.category = 'Restaurants')
AND
(b.stars BETWEEN 2.0 AND 3.0 OR b.stars BETWEEN 4.0 AND 5.0)
GROUP BY days, stars
ORDER BY days, star_rating
```

2. Group business based on the ones that are open and the ones that are closed. What differences can you find between the ones that are still open and the ones that are closed? List at least two differences and the SQL code you used to arrive at your answer.

i. Difference 1:

Still open businesses have 7 times more review than that of closed ones

ii. Difference 2:

Avg rating of open businesses is slightly higher than closed ones.

SQL code used for analysis:

```
SELECT
  is_open,
  count(*) AS num,
  avg(stars) AS avg_rating,
  sum(review_count) AS review_count
FROM business
GROUP BY is_open
```

3. For this last part of your analysis, you are going to choose the type of analysis you want to conduct on the Yelp dataset and are going to prepare the data for analysis.

Ideas for analysis include: Parsing out keywords and business attributes for sentiment analysis, clustering businesses to find commonalities or anomalies between them, predicting the overall star rating for a business, predicting the number of fans a user will have, and so on. These are just a few examples to get you started, so feel free to be creative and come up with your own problem you want to solve. Provide answers, in-line, to all of the following:

i. Indicate the type of analysis you chose to do:

Analysis on business rating relations with reviews and photos

ii. Write 1-2 brief paragraphs on the type of data you will need for your analysis and why you chose that data:

- I've categorised the business into 3 category (low, medium and high raiting). Group by them and find out their relations with review and photos.
- Turns out in order to get high rating the review should be complimented as useful/funny/cool rather than getting higher number of reviews -> Medium rating business has over 800k review and high rating has 280k but they have similar useful/funny/cool on the reviews. So the content of the review gives a bigger impact onto the rating
- Photos are also affecting the rating. As higher the rating are, more photos are included. So users would rely on photos as evidence for rating.

iii. Output of your finished dataset:

rate_category	rev_count	useful_count	funny_count	cool_count	photo_count
1. Low Rating (below 2)	891	16	3	2	0
2. Medium Rating (2.1 - 3.9)	803280	575	191	260	255
3. High Rating (above 4)	285119	538	141	272	286

iv. Provide the SQL code you used to create your final dataset:

```
SELECT
  CASE
    WHEN b.stars BETWEEN 0 AND 2 THEN "1. Low Rating (below 2)"
    WHEN b.stars BETWEEN 2.1 AND 3.9 THEN "2. Medium Rating (2.1 - 3.9)"
    ELSE "3. High Rating (above 4)"
  END AS rate_category,
  sum(b.review_count) AS rev_count,
  sum(r.useful) AS useful_count,
  sum(r.funny) AS funny_count,
  sum(r.cool) AS cool_count,
  count(p.id) AS photo_count
FROM business AS b JOIN review AS r
  on b.id = r.business_id
LEFT JOIN photo AS p
  ON b.id = p.business_id
GROUP BY rate_category
ORDER BY rate_category
```