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.

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.

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

```
SQL CODE:
SELECT COUNT(*)
FROM table;
```

---

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

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

```
SQL CODE:
SELECT COUNT(DISTINCT *)
FROM table;
```

---

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: 2000 avg: 24.2995

SQL CODE:
```
SELECT MIN(column), MAX(column), AVG(column)
FROM table;
```

---

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

(Output limit exceeded, 25 of 362 total rows shown)

---

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 Stars, SUM (review_count) AS Count
FROM business
```

```
WHERE city = 'Avon'
GROUP BY stars;
```

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

Stars	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 AS Stars, SUM (review_count) AS Count
FROM business
WHERE city = 'Beachwood'
GROUP BY stars;
```

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

Stars	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 name AS Name, review_count AS Total_Reviews
FROM user
ORDER BY review_count DESC
LIMIT 3;
```

Copy and Paste the Result Below:

Name	Total_Reviews
Gerald	2000
Sara	1629
Yuri	1339

---

8. Does posing more reviews correlate with more fans?

Please explain your findings and interpretation of the results:
No, posting more does not correlate with more fans. When comparing the review count and the amount of fans, there does not exist a positive correlation between the two and the results are mixed. The person with the most amount of fans does not have the highest review count (503 fans and 609 review counts), as the highest review count is 2000.

SQL Code:
```
SELECT name, fans, review_count
FROM user
ORDER BY fans DESC;
```

name	fans	review_count
Amy	503	609
Mimi	497	968
Harald	311	1153
Gerald	253	2000
Christine	173	930
Lisa	159	813
Cat	133	377
William	126	1215
Fran	124	862
Lissa	120	834
Mark	115	861
Tiffany	111	408
bernice	105	255
Roanna	104	1039
Angela	101	694
.Hon	101	1246
Ben	96	307
Linda	89	584
Christina	85	842
Jessica	84	220
Greg	81	408
Nieves	80	178
Sui	78	754
Yuri	76	1339
Nicole	73	161

(Output limit exceeded, 25 of 10000 total rows shown)

---

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

Answer: "love" has 1780 reviews while "hate" only has 232 reviews

SQL code used to arrive at answer:
```
SELECT COUNT(text)
FROM review
WHERE text LIKE ('%love%');
```

```
SELECT COUNT(text)
FROM review
WHERE text LIKE ('%hate%');
```

---

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.

City = Las Vegas; Category = Shopping

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

Yes, they do. The place with 2.5 ratings opens on Saturday from 8:00-22:00 while the one with 4.0 rating opens on Saturday at a different time from 10:00-19:00.

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

Yes, the place with 2.5 stars only has 6 reviews while the place with 4.0 stars has 30 reviews.

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

No, because both of them have different locations in terms of address and postal code.

SQL code used for analysis:

```
SELECT business.name, business.city,
category.category, business.stars,
hours.hours, business.review_count,
business.address, business.postal_code
FROM (business INNER JOIN category ON business.id = category.business_id)
INNER JOIN hours ON hours.business_id = business.id
WHERE business.city = 'Las Vegas' AND category.category = "Food"
GROUP BY business.stars;
```

```
+----------------------------+-----------+----------+-------+--------------
-------+-------------+----------------------------+-------------+
| name                       | city      | category | stars | hours
| review_count | address                      | postal_code |
+----------------------------+-----------+----------+-------+--------------
-------+-------------+----------------------------+-------------+
| Walgreens                  | Las Vegas | Food     |   2.5 | Saturday|8:00-
22:00  |            6 | 3808 E Tropicana Ave        | 89121       |
| Sweet Ruby Jane Confections | Las Vegas | Food     |   4.0 |
Saturday|10:00-19:00 |           30 | 8975 S Eastern Ave, Ste 3-B | 89123
|
+----------------------------+-----------+----------+-------+--------------
-------+-------------+----------------------------+-------------+
```

---

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: The businesses that are still open have more review on average in comparison to the businesses that are closed.

ii. Difference 2: There are more businesses that are still open listed as "cool" or "funny".

SQL code used for analysis:

```
SELECT
AVG(b.stars),SUM(b.review_count),AVG(b.review_count),COUNT(r.funny)+COUNT(r.c
ool),is_open
FROM business b INNER JOIN review r ON b.id = r.id
GROUP BY b.is_open;
```

```
+--------------+--------------------+--------------------+---------------
--------------+---------+
|  AVG(b.stars) | SUM(b.review_count) | AVG(b.review_count) |
COUNT(r.funny)+COUNT(r.cool) | is_open |
+--------------+--------------------+--------------------+---------------
--------------+---------+
|           2.0 |                  4 |                4.0 |
2 |       0 |
| 2.96153846154 |                504 |       38.7692307692 |
26 |       1 |
+--------------+--------------------+--------------------+---------------
--------------+---------+
```

---

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.

i. Indicate the type of analysis you chose to do:
I will conduct a sentiment analysis by parsing out keywords and business attributes to evaluate the rating of restaurants.

ii. Write 1-2 brief paragraphs on the type of data you will need for your analysis and why you chose that data:
For my analysis, I would need to pick a specific category to evaluate – Restaurants. And then I have to evaluate the attribute, review text, and star ratings. After a quick analysis of the text, I have noticed usage of several key words to indicate positive connotation to the reviews (words like love, like, good, tasty, yum). However, I noticed some of these key words in the negative text, so, I added stars with the key words in the statement to account for some minor differences. In addition, I have added the sum of the attribute value and the review count to see if there is any correlation between these two variables.

iii. Output of your finished dataset:
Positive

name	sum(a.value)	stars	review_count	category
Sushi Osaka	3.0	4.5	8	Restaurants
The Cider Mill	14.0	4.0	91	Restaurants
Nabers Music, Bar & Eats	9.0	4.0	75	Restaurants
Hermanos Mexican Grill	6.0	4.0	69	Restaurants
Charlie D's Catfish & Chicken	10.0	4.5	7	Restaurants
Papa Murphy's	5.0	4.0	4	Restaurants
Masamune Japanese Restaurant	8.0	4.0	61	Restaurants
Bootleggers Modern American Smokehouse	195.0	4.0	431	Restaurants
Edulis	9.0	4.0	89	Restaurants

Negative:

name	sum(a.value)	stars	review_count	category
Fiesta Ranchera	1.0	2.0	4	Restaurants
Royal Dumpling	3.0	1.5	4	Restaurants
McDonald's	9.0	2.0	8	Restaurants
Burger King	2	1.0	4	Restaurants
Iron City Grille	7.0	2.0	3	Restaurants
99 Cent Sushi	6.0	2.0	5	Restaurants

Recommend the negatively perceived restaurants to address the concerns from the customers to increase the public perception of the restaurants on Yelp

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

```
SELECT b.name,
```

```
sum(a.value),
b.stars,
b.review_count,
c.category
FROM business b
INNER JOIN attribute a
ON b.id = a.business_id
LEFT JOIN review r
ON b.id = r.business_id
LEFT JOIN category c
on b.id = c.business_id
WHERE b.stars between 4 and 5 AND
c.category =='Restaurants' or (r.text like '%love%' or r.text like'%like%' or
r.text like'%good%' or r.text like'%clean%' or r.text like '%tasty%' or
r.text like '%yum%')
group by b.id
```

Negative Perception Restaurants:

```
SELECT b.name,
sum(a.value),
b.stars,
b.review_count,
c.category
FROM business b
INNER JOIN attribute a
ON b.id = a.business_id
LEFT JOIN review r
ON b.id = r.business_id
LEFT JOIN category c
on b.id = c.business_id
WHERE b.stars between 1 and 2
AND c.category =='Restaurants' or (r.text like '%hate%' or r.text
like'%bad%')
group by b.id
having b.stars
```