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 = 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 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 = business_id : 8090 , user_id : 9581
vi. Checkin = 493
vii. Photo = id :10000, business_id : 6493
viii. Tip = busines id :3979 , user_ id: 537
ix. User = 10000
x. Friend = 11
xi. Elite_years =2780

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 max: 5 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) as review
FROM business
GROUP BY city
ORDER BY review desc
```

Copy and Paste the Result Below:

```
+-----------------+--------+
| city            | review |
+-----------------+--------+
| Las Vegas       |  82854 |
| Phoenix         |  34503 |
| Toronto         |  24113 |
| Scottsdale      |  20614 |
| Charlotte       |  12523 |
| Henderson       |  10871 |
| Tempe           |  10504 |
| Pittsburgh       |   9798 |
| Montréal         |   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, 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, 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, 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, because there are so many restaurants with a lot of reviews but less fans than a less reviewed restaurant.

Please explain your findings and interpretation of the results:

```
SELECT fans, review_count
FROM user
ORDER BY review_count desc

+------+--------------+
| fans | review_count |
+------+--------------+
|  253 |         2000 |
|   50 |         1629 |
|   76 |         1339 |
|  101 |         1246 |
|  126 |         1215 |
|  311 |         1153 |
|   16 |         1116 |
|  104 |         1039 |
|  497 |          968 |
|  173 |          930 |
|   38 |          904 |
|   43 |          864 |
|  124 |          862 |
|  115 |          861 |
|   85 |          842 |
|   37 |          836 |
|  120 |          834 |
|  159 |          813 |
|   61 |          775 |
|   78 |          754 |
|   35 |          702 |
|   10 |          696 |
|  101 |          694 |
|   25 |          676 |
|   45 |          675 |
+------+--------------+
(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

SQL code used to arrive at answer:

```
SELECT count(text)
FROM review
where text like "%hate%"  -->232

SELECT count(text)
FROM review
where text like "%love%" -->1780
```

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

SQL code used to arrive at answer:

```
SELECT fans, name
FROM user
ORDER BY fans desc
LIMIT 10
```

Copy and Paste the Result Below:

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

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?
No, the groups have exactly the same hours everyday.

ii. Do the two groups you chose to analyze have a different number of reviews?
yes, 2-3 stars have 6 reviews, on the other hand 4-5 stars have 30 reviews.

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

Yes, the restaurants address which has 2-3 stars is 3808 E Tropicana Ave and the restaurants address which has 4-5 stars is 8975 S Eastern Ave, Ste 3-B

SQL code used for analysis:

```
select b.city, c. category, b.id, b.stars, h.hours, b.review_count,
b.address,
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 ord,
               CASE
                  WHEN B.stars BETWEEN 2 AND 3 THEN '2-3 stars'
                  WHEN B.stars BETWEEN 4 AND 5 THEN '4-5 stars'
               END AS star_rating
from business b, hours h
inner join category c on b.id = c.business_id
where city = "Las Vegas" and category ="Food"
GROUP BY stars,ord
ORDER BY ord,star_rating ASC
```

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 ones that are open have more review count, the ones that are closed have less review count.

ii. Difference 2:

Open restaurants have slightly more average stars than the closed ones.

SQL code used for analysis:

```
SELECT COUNT(DISTINCT(id)),
AVG(review_count),
SUM(review_count),
AVG(stars),
is_open
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:
I want to learn if there is a relationship between the stars that the restaurant gets with the reviews that thinks the restaurant is cool or funny or useful.

ii. Write 1-2 brief paragraphs on the type of data you will need for your analysis and why you chose that data:
Firstly I will need to know which restaurant gets the “cool” or “funny” or “useful” reviews and compare the number of cool reviews with their stars rating. Then all I need to do is compare the data with the stars and cool and funny and useful to get some results.

iii. Output of your finished dataset:
I could not find any relationship between cool and funny but there is a relationship that is useful. When people find the restaurant useful they give a higher rate of stars.

```
+------------------------------------+-------+------+-------+--------+
| name                               | stars | cool | funny | useful |
+------------------------------------+-------+------+-------+--------+
| BCT Flooring and Showers           |   1.0 |    0 |     0 |      0 |
| Showtime Tours                     |   1.5 |    2 |     0 |      3 |
| Shafa Medical Clinic               |   1.5 |    0 |     0 |      0 |
| Jimmy Johns                        |   2.0 |    0 |     0 |      0 |
| Fiesta Ranchera                    |   2.0 |    0 |     0 |      0 |
| Neubert Painting                   |   2.0 |    0 |     0 |      0 |
| Belmont Cleaners and Laundry       |   2.0 |    0 |     0 |      0 |
| China Restaurant                   |   3.0 |    1 |     0 |      0 |
| Hardee's                           |   3.0 |    1 |     0 |      1 |
| Fit4Less                           |   3.5 |    0 |     0 |      0 |
| Vegas Uncork'd: The Grand Tasting  |   4.0 |    1 |     0 |      1 |
| Impressions Tile & Marble          |   5.0 |    0 |     0 |      2 |
| My Biz Niche                       |   5.0 |    0 |     0 |      1 |
| Arizona Goldendoodles              |   5.0 |    0 |     0 |      4 |
+------------------------------------+-------+------+-------+--------+
```

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

```sql
select b.name, b.stars, r.cool, r.funny, r.useful
from business b
inner join review r on b.id = r.id
order by b.stars
```