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 = 10,000
ii. Hours = 1,562
iii. Category = 2,643
iv. Attribute = 1,115
v. Review = id: 10,000 / business_id: 8,090 / user_id: 9,581
vi. Checkin = 493
vii. Photo = id: 10,000 / business_id: 6,493
viii. Tip = user_id: 537 / business_id: 3,979
ix. User = 10,000
x. Friend = 11
xi. Elite_years = 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: 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_count
FROM business
GROUP BY city
ORDER BY review_count DESC;
```

Copy and Paste the Result Below:

```
+-----------------+--------------+
| city            | review_count |
+-----------------+--------------+
| 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 – 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 – 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?

Please explain your findings and interpretation of the results:

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
Tim	702	35
L	696	10
Angela	694	101
Crissy	676	25
Lyn	675	45

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

Posting more reviews does not correlate with more fans. Sara, Yuri, and eric have quite a lot of reviews, but the number of fans are quite small, whereas Mimi, Christine, and Lisa don't have as many as reviews, but they have a lot more fans compared to Sara, Yuri, and eric. Other factors also affect the number of fans one can have.

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

Answer:

Yes. There are 1780 reviews contain the word "love" and 232 reviews contain the word "hate".

SQL code used to arrive at answer:

```
SELECT COUNT(*) AS like
FROM review
WHERE text LIKE "%love%";

SELECT COUNT(*) AS hate
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.

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

I picked "Toronto" and "Food" category.
Yes. The businesses with higher star ratings open later compared to the one with lower star rating.

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

Yes. The businesses with higher star ratings have more reviews than the one with lower star ratings.

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

No. They are at different locations.

SQL code used for analysis:

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

```
+--------------+-------+-------+-------------+----------------------+
| name         | stars | count | postal_code | hours                |
+--------------+-------+-------+-------------+----------------------+
| Loblaws      |   2.5 |    10 | M6R 1X3     | Saturday|8:00-22:00  |
| Halo Brewery |   4.0 |    15 | M6H 1V5     | Saturday|11:00-21:00 |
| Cabin Fever  |   4.5 |    26 | M6P 1A6     | Saturday|16:00-2:00  |
+--------------+-------+-------+-------------+----------------------+
```

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.

```
+---------+---------------+--------+---------+--------------+
| is_open |    avg_rating | review | userful | contain_like |
+---------+---------------+--------+---------+--------------+
|       0 | 3.61290322581 |   3583 |      31 |           31 |
|       1 |  3.7869955157 |  77793 |     223 |          223 |
+---------+---------------+--------+---------+--------------+
```

i. Difference 1:

Businesses that are open have more reviews and more useful reviews.

ii. Difference 2:

Businesses that are open have more reviews contain the word "like" or "love".

SQL code used for analysis:

```
SELECT business.is_open
    , AVG(business.stars) AS avg_rating
    , SUM(business.review_count) AS review
    , COUNT(review.useful) AS userful
    , COUNT(review.text) AS contain_like
FROM business LEFT JOIN review ON business.id = review.business_id
WHERE review.text LIKE "%like%" OR review.text LIKE "%love%"
GROUP BY business.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:

To find the best of different types of Asian food.

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

1) Join "business" table and "category" table based on id
2) Choose category of interest. Here I chose a few Asian food - Chinese food, Japanese food, Korean, Vietnamese and Indian food
3) Analyze average star ratings and number of reviews each type of Asian food gets.
4) The results are grouped by city and food category, and sorted descendingly by average of star ratings.

iii. Output of your finished dataset:

city	state	category	avg_rating	review_count
Las Vegas	NV	Japanese	4.5	3

Toronto	ON	Korean	4.5	8
Brampton	ON	Indian	4.0	10
Cuyahoga Falls	OH	Korean	4.0	55
Las Vegas	NV	Chinese	4.0	768
Mississauga	ON	Japanese	4.0	61
Aurora	OH	Indian	3.5	32
Cleveland	OH	Vietnamese	3.5	62
Edinburgh	EDH	Chinese	3.5	3
Edinburgh	EDH	Indian	3.5	3
Fountain Hills	AZ	Chinese	3.5	21
Inverness	HLD	Indian	3.5	3
Montréal	QC	Indian	3.5	15
Toronto	ON	Japanese	3.5	88
Toronto	ON	Chinese	1.5	4

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

```
SELECT business.city
    , business.state
    , category.category
    , AVG(business.stars) AS avg_rating
    , SUM(business.review_count) AS review_count
FROM business INNER JOIN category ON business.id =
category.business_id
WHERE category.category IN ("Chinese", "Korean", "Japanese", "Indian",
"Vietnamese")
GROUP BY business.city, category.category
ORDER BY AVG(business.stars) DESC;
```