

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

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

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

```
*****SQL CODE*****  
SELECT COUNT(DISTINCT Keys)  
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) AS NUM
FROM business
GROUP BY city
ORDER BY NUM DESC
```

Copy and Paste the Result Below:

city	NUM
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 SUM(review_count) AS Numbers, stars
FROM business
WHERE city == "Avon"
GROUP BY stars
```

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

+-----+-----+

| Numbers | stars |

+-----+-----+

| 10 | 1.5 |

| 6 | 2.5 |

| 88 | 3.5 |

| 21 | 4.0 |

| 31 | 4.5 |

| 3 | 5.0 |

+-----+-----+

ii. Beachwood

SQL code used to arrive at answer:

```
SELECT SUM(review_count) AS Numbers, stars
FROM business
WHERE city == "Beachwood"
GROUP BY stars
```

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

+-----+-----+

| Numbers | stars |

	8	2.0
	3	2.5
	11	3.0
	6	3.5
	69	4.0
	17	4.5
	23	5.0

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

SQL code used to arrive at answer:

```
SELECT review_count, name
FROM user
ORDER BY review_count DESC
LIMIT 3
```

Copy and Paste the Result Below:

	review_count	name
	2000	Gerald
	1629	Sara
	1339	Yuri

8. Does posing more reviews correlate with more fans?

Please explain your findings and interpretation of the results:

Not necessarily correlated. Amy, who has the most fans, only has 609 reviews. Yuri has only 76 fans, but has the third most reviews. Some other factors should also be considered.

```
*****SQL CODE*****
SELECT name,review_count,fans
FROM user
ORDER BY fans DESC
```

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
Mark	861	115
Tiffany	408	111
bernice	255	105
Roanna	1039	104
Angela	694	101
.Hon	1246	101
Ben	307	96
Linda	584	89
Christina	842	85
Jessica	220	84
Greg	408	81
Nieves	178	80
Sui	754	78
Yuri	1339	76
Nicole	161	73

(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: Yes. There are 1780 reviews with "love" and 232 reviews with "hate"

SQL code used to arrive at answer:

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

```
SELECT COUNT(*)
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

11. Is there a strong relationship (or correlation) between having a high number of fans and being listed as "useful" or "funny?" Out of the top 10 users with the highest number of fans, what percent are also listed as "useful" or "funny"?

Key:

0% - 25% - Low relationship
26% - 75% - Medium relationship
76% - 100% - Strong relationship

SQL code used to arrive at answer:

```
SELECT name,fans,useful,funny
FROM user
ORDER BY fans DESC
LIMIT 10
```

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

Please explain your findings and interpretation of the results:

Out of the top 10 users with the highest number of fans, 100% are also listed as either "useful" or "funny". So there is a strong correlation between having a high number of fans and being listed as "useful" or "funny".

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 for this question. Yes. The restaurants with only 2.5 stars open from 8:00-22:00 on Saturday. The places with higher rating stars open late on Saturday.

The results from SQL is shown below:

```
+-----+-----+-----+-----+
+-----+
| name          | city    | category | stars | hours
|
+-----+-----+-----+-----+
+-----+
| Loblaws       | Toronto | Food     | 2.5   | Saturday|
8:00-22:00 |
| Halo Brewery  | Toronto | Food     | 4.0   | Saturday|
11:00-21:00 |
| Cabin Fever   | Toronto | Food     | 4.5   | Saturday|
16:00-2:00  |
+-----+-----+-----+-----+
+-----+
```

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

Yes. The group with 2-3 stars has less review (10) compared with the group with higher rating stars.

```
+-----+-----+-----+-----+
+-----+
| name          | city    | category | stars | hours
| review_count |
+-----+-----+-----+-----+
+-----+
| Loblaws       | Toronto | Food     | 2.5   | Saturday|
8:00-22:00 | 10 |
| Halo Brewery  | Toronto | Food     | 4.0   | Saturday|
11:00-21:00 | 15 |
```

Cabin Fever	Toronto	Food	4.5	Saturday
16:00-2:00		26		

```

+-----+-----+-----+-----+
+-----+-----+-----+-----+

```

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

No really. They have different locations.

```

+-----+-----+-----+-----+
+-----+-----+-----+-----+
+-----+
| name      | city    | category | stars | hours
| review_count | address | postal_code |
+-----+-----+-----+-----+
+-----+-----+-----+-----+
+-----+
| Loblaws   | Toronto | Food     | 2.5   | Saturday|
8:00-22:00 |         | 10 | 2280 Dundas Street W | M6R 1X3
|
| Halo Brewery | Toronto | Food     | 4.0   | Saturday|
11:00-21:00 |         | 15 | 247 Wallace Avenue   | M6H 1V5
|
| Cabin Fever | Toronto | Food     | 4.5   | Saturday|
16:00-2:00  |         | 26 | 1669 Bloor Street W  | M6P 1A6
|
+-----+-----+-----+-----+
+-----+-----+-----+-----+
+-----+

```

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 =

```


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:

Here I chose to study the preference among different types of food on yelp.

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

I will pick several types of food including "Chinese", "Mexican", "Korean", "French", "Italian", "Japanese" and "Indian". Then I will analyze their star ratings and number of reviews so that I can get some insights on which type of food is popular on yelp.

iii. Output of your finished dataset:

```
+-----+-----+-----+
+-----+-----+
| category | Number_Of_Resturants |   AVG(stars) |
AVG(review_count) | city      |
+-----+-----+-----+
+-----+-----+
| Korean   |           7 |         4.5 |
8.0 | Toronto |
| French   |          12 |         4.0 |
135.083333333 | Las Vegas |
| Chinese  |          13 | 3.76923076923 |
423.230769231 | Las Vegas |
| Mexican  |          28 |         3.625 |
73.0 | Edinburgh |
| Italian  |          13 | 3.53846153846 |
78.2307692308 | Montréal  |
| Indian   |           6 |         3.5 |
32.0 | Aurora    |
| Japanese |          20 |         3.475 |
```

22.85 | Toronto |

+-----+-----+-----+-----+
+-----+-----+-----+-----+

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

```
SELECT c.category,COUNT(b.name) AS  
Number_Of_Resturants,AVG(stars),AVG(review_count),b.city  
FROM (business b INNER JOIN hours h ON b.id = h.business_id)  
INNER JOIN category c ON c.business_id = b.id  
WHERE c.category IN  
("Chinese","Mexican","French","Italian","Korean","Japanese","Ind  
ian")  
GROUP BY c.category  
ORDER BY AVG(stars) DESC
```