Data Scientist Role Play: Profiling and Analyzing the Yelp Data set 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 Text Edit, 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 Data set 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. Check-in 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 = 10000
vi. Check-in = 493
vii. Photo = 10000
viii. Tip = 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."

There are no null values in the table

SQL code used to arrive at answer:

```
SELECT count(*) - count(id),
count(*) - count(name),
count(*) - count(review_count),
count(*) - count(yelping_since),
count(*) - count(useful),
count(*) - count(cool),
count(*) - count(fans),
count(*) - count(average_stars),
count(*) - count(compliment_hot),
count(*) - count(compliment_more),
count(*) - count(compliment_profile),
count(*) - count(compliment_cute),
count(*) - count(compliment_list),
count(*) - count(compliment_note),
count(*) - count(compliment_plain),
count(*) - count(compliment_cool),
count(*) - count(compliment_funny),
count(*) - count(compliment_writer),
count(*) - count(compliment_photos)
from user
```

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: 3.7

ii. Table: Business, Column: Stars
   min: 1 max: 5 avg: 3.65

iii. Table: Tip, Column: Likes
   min: 0 max: 2 avg: 0.014

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 total from business
group by city
```

```
order by total desc
```

Copy and Paste the Result Below:

```
+-----------------+-------+
| city            | total |
+-----------------+-------+
| 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 as Star_rating, count(stars) as Count from business
where city = "Avon"
group by stars
```

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

```
+-------------+-------+
| Star_rating | Count |
+-------------+-------+
|         1.5 |     1 |
|         2.5 |     2 |
|         3.5 |     3 |
|         4.0 |     2 |
|         4.5 |     1 |
|         5.0 |     1 |
+-------------+-------+
```

ii. Beachwood

SQL code used to arrive at answer:

```
select stars as Star_rating, count(stars) as Count from business
where city = "Beachwood"
group by stars
```

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

```
+-------------+-------+
| Star_rating | Count |
+-------------+-------+
|         2.0 |     1 |
|         2.5 |     1 |
|         3.0 |     2 |
|         3.5 |     2 |
|         4.0 |     1 |
|         4.5 |     2 |
|         5.0 |     5 |
+-------------+-------+
```

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 03
```

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:

No, there are no solid evidence that posing more review correlate with more fans, because there were people with more fans by posing less reviews and vice versa, So, I think it depends on the content of the person review than the number of review.

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

Answer: There are more reviews with word love than word hate, there are 1280 reviews with word love and 232 with hate

SQL code used to arrive at answer:

```
select text from review
where text like '%love%'

select 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.

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

Explain:

Yes the two groups have different distribution of hours.

SQL Code:

```
select
case when stars >=4 Then "4-5 Stars"
when stars >=2 Then "2-3 Stars"
else "below 2"
End rating,
name,
count(distinct(h.business_id)) as company_count,
count(h.hours) as wokring_days
from category c join business b
on b.id = c.business_id join hours h on h.business_id = c.business_id
where city = 'Las Vegas' and category = ‘Shopping’
group by rating
```

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

Explain: Yes the two groups have different number of reviews as 17 and 36.

SQL Code:

```
select
```

```
case when stars >=4 Then "4-5 Stars"
when stars >=2 Then "2-3 Stars"
else "below 2"
End rating,
name,
city,
sum(review_count) as total_reviews
from category c join business b
on b.id = c.business_id
where city = 'Las Vegas' and category = 'Shopping'
group by rating
```

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

Explain: Stores that have 2-3 stars are within the same area, whereas 4-5 stars stores are apart from each other according to the result of postal code.

SQL code used for analysis:

```
SELECT
city,
neighborhood,
name,
review_count,
stars,
postal_code
from (business b INNER JOIN category c ON b.id = c.business_id)
WHERE CITY = 'Las Vegas' AND c.category ='Shopping'
order BY stars
```

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: Total Number of reviews for the business that are still open are high compared to the closed one.

ii. Difference 2: The average stars for both the case is quite closer, whereas the number of company that are still open are significantly higher compared to the closed ones.

SQL code used for analysis:

```
SELECT CASE WHEN is_open = 1 THEN "STILL OPEN"
WHEN is_open = 0 THEN "CLOSED"
END status,
count(distinct id) AS num_company,
```

```
sum(review_count) AS total_review,
round(avg(review_count),2) AS avg_review,
round(avg(stars),2) AS avg_stars
FROM business
GROUP BY is_open
ORDER BY status DESC
```

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:

Which category of business is performing better compared to other?

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

Among the categories, Average star rating is calculated and proportion of the business opened and number of the business open are calculated and the analysis is restricted to the number of business > 10 and ordered by average star rating and average proportion of store open

iii. Output of your finished data set:

category	num_business	avg_stars	avg_isopen
Local Services	12	4.21	0.83
Health & Medical	17	4.09	0.94
Home Services	16	4.0	0.94
Shopping	30	3.98	0.83
Beauty & Spas	13	3.88	0.92
American (Traditional)	11	3.82	0.73
Food	23	3.78	0.87
Bars	17	3.5	0.65
Nightlife	20	3.48	0.6
Restaurants	71	3.46	0.75

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

```
SELECT category.category,
count(business.id) num_business,
round(avg(business.stars),2) avg_stars,
round(avg(business.is_open),2) avg_isopen
FROM (business INNER JOIN category ON business.id = category.business_id)
GROUP BY category.category
HAVING num_business > 10
ORDER BY avg_stars DESC, avg_isopen DESC
```