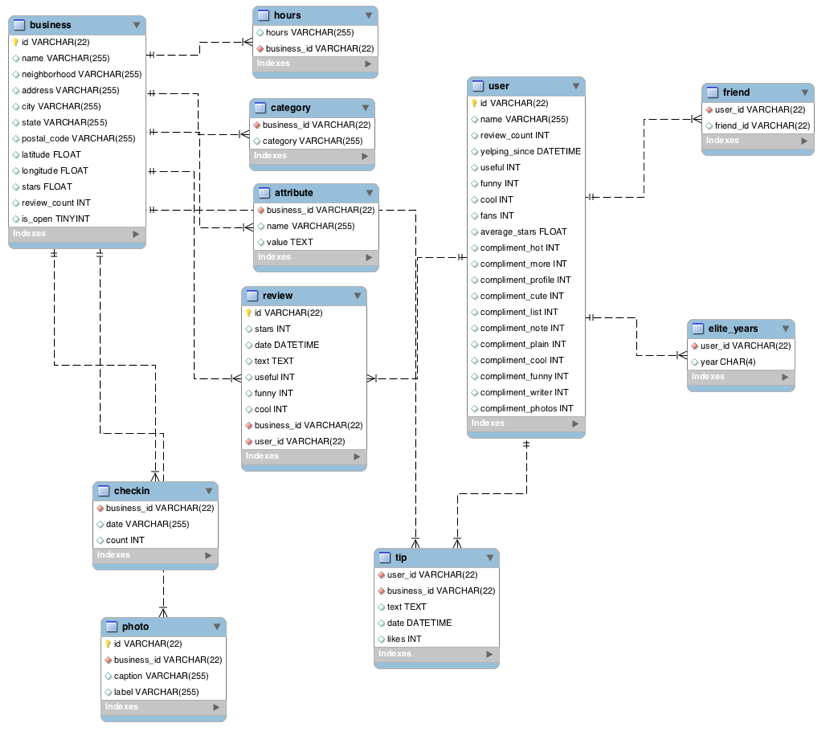
Data Scientist Role Play: Profiling and Analyzing the Yelp Dataset Coursera Worksheet

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.

**Note: FK means Foreign Key**

i. Business = 10,000

ii. Hours = 1562 (business\_id = FK)

iii. Category = 2643 (business\_id = FK)

iv. Attribute = 1115 (business\_id = FK)

v. Review = 10,000(id); 8090(business\_id = FK); 9581(user\_id = FK)

vi. Checkin = 493

vii. Photo = 10,000(id) ;6493(business\_id = FK)

viii. Tip = 537(user\_id=FK), 3979 (business\_id=FK)

ix. User = 10,000

x. Friend = 11 (user\_id=FK)

xi. Elite\_years = 2780 (user\_id=FK)

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 \* FROM user

WHERE id 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)

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 |

| 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 |

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

6. Find the distribution of star ratings to the business in the following cities:

i. Avon

SQL code used to arrive at answer:

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

| STAR\_RATING | COUNT |

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

| 1.5 | 1 |

| 2.5 | 2 |

| 3.5 | 3 |

| 4.0 | 2 |

| 4.5 | 1 |

| 5.0 | 1 |

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

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

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 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 3;

Copy and Paste the Result Below:

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

| name | review\_count |

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

| Gerald | 2000 |

| Sara | 1629 |

| Yuri | 1339 |

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

8. Does posting more reviews correlate with more fans? No

Please explain your findings and interpretation of the results:

Fans usually write numerous reviews which explains why the review count is higher than the fans.

The code used to confirm this idea is:

SELECT name, review\_count, fans

FROM user

ORDER BY fans DESC;

The results of the code are as follows:

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

| 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 |

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

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

Answer: **Love.** Love has 1780 results whereas hate has 232 results.

SQL code used to arrive at answer:

SELECT (SELECT COUNT(text)

FROM review

WHERE text LIKE "%love%") AS lovecount,

(SELECT COUNT(text) AS hatecount

FROM review

WHERE text LIKE "%hate%") AS hatecount;

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?

It appears that the group with 4-5 stars are open on Saturday in the evening whereas the locations with 2-3 stars are open earlier in the day.

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

Yes, generally the 4-5 star group has more reviews than the 2-3 star group in Toronto

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

A lot of people like the restaurant Edulis in the neighborhood Niagra. It is the only restaurant in the neighborhood. Most of the restaurants are the only resturants in the neighborhood except for the area Downtown Core, and Willowdale.

SQL code used for analysis:

SELECT b.city, b.stars, b.name, b.review\_count, h.hours, c.category, b.is\_open, b.neighborhood, b.postal\_code

FROM business b

LEFT JOIN hours h ON h.business\_id=b.id

LEFT JOIN category c ON c.business\_id=b.id

WHERE b.city='Toronto' AND c.category='Restaurants'

GROUP BY b.name;

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: Generally, the businesses which are open has more reviews

ii. Difference 2: Generally, the businesses which closed has more stars than the ones which are open.

SQL code used for analysis:

SELECT b.city, b.stars, b.name, b.review\_count, h.hours, c.category, b.is\_open, b.neighborhood, b.postal\_code

FROM business b

LEFT JOIN hours h ON h.business\_id=b.id

LEFT JOIN category c ON c.business\_id=b.id

GROUP BY b.name, b.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:

ABC Company is looking for 5 individuals who currently has an average rating of 4 or above to become paid influencers to market the platform.

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

The type of data needed for this analysis will be the user table in the ERD diagram provided. The fields which will be utilized mainly are name, fans, review\_count, yelping since, and average stars. The remaining fields can be utilized when marketing to depict the positive aspects of the influencers who are picked. To pick an influencer we will also have to analyze their Youtube channel, Youtube statistics, Instagram (IG) page, Instagram statistics, and any other platform which can be utilized to find out information about the prospective influencers. Many times people look at the amount of followers one has on IG or the number of followers on Youtube. However it is best to pick the persons who have a good conversion rate when it comes to selling a product or service. Based on this dataset we do not have access to outside sources thus we will utilize the first 5 individuals who has the highest fans and average of 4 or above. Based on the code run below Mimi, Harald, Lisa, William, and Fran will be chosen as Yelp influencers.

iii. Output of your finished dataset:

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

| name | fans | review\_count | yelping\_since | average\_stars |

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

| Mimi | 497 | 968 | 2011-03-30 00:00:00 | 4.05 |

| Harald | 311 | 1153 | 2012-11-27 00:00:00 | 4.4 |

| Lisa | 159 | 813 | 2009-10-05 00:00:00 | 4.09 |

| William | 126 | 1215 | 2015-02-19 00:00:00 | 4.41 |

| Fran | 124 | 862 | 2012-04-05 00:00:00 | 4.1 |

| Tiffany | 111 | 408 | 2008-10-28 00:00:00 | 4.09 |

| Linda | 89 | 584 | 2005-08-07 00:00:00 | 4.06 |

| Christina | 85 | 842 | 2012-10-08 00:00:00 | 4.1 |

| Jessica | 84 | 220 | 2009-01-12 00:00:00 | 4.1 |

| Yuri | 76 | 1339 | 2008-01-03 00:00:00 | 4.11 |

| Koizumi | 73 | 160 | 2006-09-22 00:00:00 | 4.19 |

| Nelson | 70 | 7 | 2012-01-27 00:00:00 | 4.43 |

| rebecca | 69 | 6 | 2008-08-26 00:00:00 | 4.33 |

| Debbie | 68 | 249 | 2008-02-01 00:00:00 | 4.08 |

| Mike | 65 | 346 | 2014-03-19 00:00:00 | 4.12 |

| Princeton | 64 | 376 | 2009-03-04 00:00:00 | 4.15 |

| Susan | 63 | 6 | 2015-04-08 00:00:00 | 4.5 |

| Jenn | 62 | 657 | 2006-05-20 00:00:00 | 4.36 |

| Jayd | 60 | 39 | 2009-02-07 00:00:00 | 4.23 |

| Annie | 52 | 651 | 2005-12-14 00:00:00 | 4.0 |

| Echo | 46 | 42 | 2013-04-16 00:00:00 | 4.04 |

| Cynthia | 45 | 54 | 2011-09-10 00:00:00 | 4.54 |

| Sean | 45 | 213 | 2010-01-29 00:00:00 | 4.0 |

| Lyn | 45 | 675 | 2009-11-07 00:00:00 | 4.06 |

| Nicole | 43 | 864 | 2006-08-02 00:00:00 | 4.0 |

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

(Output limit exceeded, 25 of 4705 total rows shown)

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

SELECT name, fans, review\_count, yelping\_since, average\_stars

FROM user

WHERE average\_stars>=4

ORDER BY fans DESC, yelping\_since DESC, average\_stars DESC;