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

SELECT COUNT(\*) AS total\_AttributeRecords

FROM attribute;

= 10000

ii. Business table

SELECT COUNT(\*) AS total\_BusinessRecords

FROM business;

= 10000

iii. Category table

SELECT COUNT(\*) AS total\_CategoryRecords

FROM category;

= 10000

iv. Checkin table

SELECT COUNT(\*) AS total\_CheckinRecords

FROM checkin;

= 10000

v. elite\_years table

SELECT COUNT(\*) AS total\_EYRecords

FROM elite\_years;

= 10000

vi. friend table

SELECT COUNT(\*) AS total\_FriendRecords

FROM friend;

= 10000

vii. hours table

SELECT COUNT(\*) AS total\_HoursRecords

FROM hours;

= 10000

viii. photo table

SELECT COUNT(\*) AS total\_PhotoRecords

FROM photo;

= 10000

ix. review table

SELECTCOUNT(\*) AS total\_ReviewRecords

FROM review;

= 10000

x. tip table

SELECT COUNT(\*) AS total\_TipRecords

FROM tip;

= 10000

xi. user table

SELECT COUNT(\*) AS total\_UserRecords

FROM user;

= 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

SELECT COUNT(DISTINCT id) AS total\_distinct\_business\_records

FROM Business;

total\_distinct\_business\_records=10000

ii. Hours

SELECT COUNT(DISTINCT business\_id) AS total\_distinct\_hours\_records FROM Hours;

total\_distinct\_hours\_records = 1562

iii. Category

SELECT COUNT(DISTINCT business\_id) AS total\_distinct\_category\_records

FROM Category;

total\_distinct\_category\_records= 2643

iv. Attribute

SELECT COUNT(DISTINCT business\_id) AS total\_distinct\_attribute\_records FROM attribute;

Total\_distinct\_attribute\_records = 1115

v. Review

SELECT COUNT(DISTINCT id) AS total\_distinct\_review\_record

FROM review;

total\_distinct\_review\_record= 10000

vi. Checkin

SELECT COUNT(DISTINCT business\_id) AS total\_distinct\_checkin\_records

FROM checkin;

Total\_distinct\_checkin\_records = 493

vii. Photo

SELECT COUNT(DISTINCT id) AS total\_distinct\_photo\_records

FROM photo;

Total\_distinct\_photo\_records = 10000

viii. Tip

SELECT COUNT(business\_id) AS total\_distinct\_tip\_records

FROM tip;

Total\_distinct\_tip\_records = 10000

ix. User

SELECT COUNT(id) AS total\_distinct\_user\_record

FROM user;

total\_distinct\_user\_records=10000

x. Friend

SELECT COUNT(user\_id) AS total\_distinct\_friend\_records

FROM friend;

Total\_distinct\_friend\_records = 10000

xi. Elite\_years

SELECT COUNT(user\_id) AS total\_distinct\_EY\_records

FROM elite\_years;

total\_distinct\_EY\_records FROM elite\_years = 10000

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(CASE WHEN id IS NULL THEN 1 END) AS idNullCount,

COUNT(CASE WHEN name IS NULL THEN 1 END) AS NameNullCount,

COUNT(CASE WHEN review\_count IS NULL THEN 1 END) AS NullReviewCount,

COUNT(CASE WHEN yelping\_since IS NULL THEN 1 END) AS YSNullCount,

COUNT(CASE WHEN useful IS NULL THEN 1 END) AS UsNullCount,

COUNT(CASE WHEN funny IS NULL THEN 1 END) AS FunnyNullCount,

COUNT(CASE WHEN cool IS NULL THEN 1 END) AS CoolNullCount,

COUNT(CASE WHEN fans IS NULL THEN 1 END) AS FansNullCount,

COUNT(CASE WHEN average\_stars IS NULL THEN 1 END) AS AvStarsNullCount,

COUNT(CASE WHEN compliment\_hot IS NULL THEN 1 END) AS CHotNullCount,

COUNT(CASE WHEN compliment\_more IS NULL THEN 1 END) AS CmoreNullCount,

COUNT(CASE WHEN compliment\_profile IS NULL THEN 1 END) AS CProfileNullCount,

COUNT(CASE WHEN compliment\_cute IS NULL THEN 1 END) AS CcuteNullCount,

COUNT(CASE WHEN compliment\_list IS NULL THEN 1 END) AS CListNullCount,

COUNT(CASE WHEN compliment\_note IS NULL THEN 1 END) AS CnoteNullCount,

COUNT(CASE WHEN compliment\_plain IS NULL THEN 1 END) AS CPlainNullCount,

COUNT(CASE WHEN compliment\_cool IS NULL THEN 1 END) AS CcoolNullCount,

COUNT(CASE WHEN compliment\_funny IS NULL THEN 1 END) AS CfunnyNullCount,

COUNT(CASE WHEN compliment\_writer IS NULL THEN 1 END) AS CWriterNullCount,

COUNT(CASE WHEN compliment\_photos IS NULL THEN 1 END) AS CPhotosNullCount

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: 3.7082

SELECT MIN(stars) AS min\_stars, MAX(stars) AS max\_stars, AVG(stars) AS avg\_stars FROM Review;

ii. Table: Business, Column: Stars

min: 1 max: 5 avg: 3.6549

SELECT MIN(stars) AS min\_stars, MAX(stars) AS max\_stars, AVG(stars) AS avg\_stars FROM Business;

iii. Table: Tip, Column: Likes

min: 0 max: 2 avg: 0.0144

SELECT MIN(likes) AS min\_stars, MAX(likes) AS max\_stars, AVG(likes) AS avg\_stars FROM tip;

iv. Table: Checkin, Column: Count

min: 1 max: 53 avg: 1.9414

SELECT MIN(count) AS min\_stars, MAX(count) AS max\_stars, AVG(count) AS avg\_stars FROM checkin;

v. Table: User, Column: Review\_count

min: 0 max: 2000 avg: 24.995

SELECT MIN(review\_count) AS min\_stars, MAX(review\_count) AS max\_stars, AVG(review\_count) AS avg\_stars FROM user;

5. List the cities with the most reviews in descending order:

SQL code used to arrive at answer:

SELECT sum(review\_count) AS review\_count, city

FROM business

GROUP BY city

ORDER BY review\_count DESC;

Copy and Paste the Result Below:

82854 | Las Vegas

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, count(stars) AS counts

FROM business

WHERE city = 'Avon'

GROUP BY stars

ORDER BY stars DESC;

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

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

| stars | count |

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

| 5.0 | 1 |

| 4.5 | 1 |

| 4.0 | 2 |

| 3.5 | 3 |

| 2.5 | 2 |

| 1.5 | 1 |

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

ii. Beachwood

SQL code used to arrive at answer:

SELECT stars, count(stars) AS counts

FROM business

WHERE city = 'Beachwood'

GROUP BY stars

ORDER BY stars DESC;

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

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

| stars | counts |

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

| 5.0 | 5 |

| 4.5 | 2 |

| 4.0 | 1 |

| 3.5 | 2 |

| 3.0 | 2 |

| 2.5 | 1 |

| 2.0 | 1 |

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

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

SQL code used to arrive at answer:

SELECT name, sum(review\_count) AS ReviewCounts

FROM user

GROUP BY name

ORDER BY ReviewCounts DESC;

Copy and Paste the Result Below:

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

| name | ReviewCounts |

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

| Nicole | 2397 |

| Sara | 2253 |

| Gerald | 2034 |

| Lisa | 2021 |

8. Does posing more reviews correlate with more fans?

Posing more reviews does not correlate with more fans

The users with the most reviews do not have the most followers, some users with less reviews have more fans

Please explain your findings and interpretation of the results:

| name | ReviewCounts | fans |

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

| Nicole | 2397 | 0 |

| Sara | 2253 | 0 |

| Gerald | 2034 | 253 |

| Lisa | 2021 | 0 |

| Mark | 1945 | 0 |

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

Answer: Yes , there are more values with the word love compared to hate

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

| love\_count | hate\_count |

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

| 1780 | 232 |

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

SQL code used to arrive at answer:

SELECT

COUNT(CASE WHEN text LIKE '%love%' THEN 1 END) AS love\_count,

COUNT(CASE WHEN text LIKE '%hate%' THEN 1 END) AS hate\_count

FROM review;

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

SQL code used to arrive at answer:

SELECT name, fans

FROM user

GROUP BY name

ORDER BY fans DESC

LIMIT 10;

Copy and Paste the Result Below:

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

| name | fans |

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

| Gerald | 253 |

| Lissa | 120 |

| bernice | 105 |

| Roanna | 104 |

| .Hon | 101 |

| Nieves | 80 |

| Sui | 78 |

| Koizumi | 73 |

| rebecca | 69 |

| Princeton | 64 |

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

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.

City = ‘Las Vegas’

Category = ‘Shopping’

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

Yes

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

yes

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

To my understanding the location data is neighborhood, address, i would need more data on the population of the locations so as to analyze and get insights on the relationship between the shopping category and the location.

SQL code used for analysis:

SELECT distinct b.name,b.id, b.neighborhood,b.city,b.address,b.stars,b.review\_count,h.hours,c.category

FROM business b

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

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

WHERE city = 'Las Vegas' AND category = 'Shopping'

GROUP BY b.name

ORDER BY stars DESC ;

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:

Difference in hours , bar with more hours has a higher review count

ii. Difference 2:

Difference in rating, bar with more stars are still open

SQL code used for analysis:

SELECT id, name, city,review\_count, stars, is\_open,h.hours, c.category

FROM business b

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

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

WHERE category = 'Bars'

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:

Predict overall star rating

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

For the analysis, i took data from various attributes, category, location, reviewcount. These attributes can be used for predicting the overall star rating as they can impact customer “mentality” when it comes to the business and affect its rating.

An example, a b/st in a popular location gets more reviews and has higher ratings due to the exposure. When you add these features, we can build a model that can accurately predict the star rating of a business based on these features and attributes.

iii. Output of your finished dataset:

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

| business\_id | name | city | review\_count | category | stars |

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

| -9lOQ0Lfm8wiu8eSdUXS8A | Moondogs Pub | Pittsburgh | 7 | Nightlife | 2.0 |

| -Eu04UHRqmGGyvYRDY8-tg | West Side Market | Cleveland | 723 | Meat Shops | 4.5 |

| -Za5mjo-CYYUMsd1r8GC7Q | Ashbridges Bay Park | Toronto | 29 | Parks | 4.0 |

| 0NDbUCHi9YsRwgG3iZO8Kg | Cafe Tandoor | Aurora | 32 | Restaurants | 4.0 |

| 1ZnVfS-qP19upP\_fwOhZsA | Big Wong Restaurant | Las Vegas | 768 | Asian Fusion | 4.5 |

| 1veVZUawy7IhIc5oDpRRQA | Slyman's Restaurant | Cleveland | 361 | Restaurants | 5.0 |

| 20ib4z2Yo2wlfARFMcFwSQ | Vanilla Pastry Studio | Pittsburgh | 72 | Food | 5.0 |

| 24Td\_CQH1bonWKff1rt2vg | Matt's Big Breakfast | Phoenix | 188 | Restaurants | 4.5 |

| 2skQeu3C36VCiB653MIfrw | Bootleggers Modern American Smokehouse | Phoenix | 431 | Barbeque | 4.66666666667 |

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

SELECT

b.id AS business\_id,

b.name,

b.city,

b.review\_count,

c.category,

AVG(r.stars) AS stars

FROM business b

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

JOIN review r ON b.id = r.business\_id

GROUP BY b.id