John Kendall

11692198

CptS 451 | Due 04/04/2024

Question 6

In differentiating what should be described as “popular” and “successful”, our proposed metrics follow the following criteria:

* Popular businesses are those which show signs of high congregation or interaction at times, regardless of possible perceivable long-term financial stability (success). Businesses with lots of reviews and foot traffic can be deemed popular without taking a concrete stance on a businesses success. Additionally, a business that receives more comparative check-ins on Friday and Saturday nights may very well be popular, while not comparatively successful (More on this example will be gone into below).
* Successful businesses are those which are relatively definitively successful. Meaning, this entirely depends on the success of a business, which is directly related to financial income. For this, since we have access to reviews, those will be parsed and taken seriously as a representation of a how successful a business is. For instance, The total count of num\_reviews would matter less for determining success in our metrics than the review\_rating would (conversely, num\_ reviews should be taken into account when determining popularity).

A metric for determining popularity we will be using is checking if a business has a higher num\_reviews AND num\_checkins than the average num\_reviews AND num\_checkins within its own zipcode for the same category, and then considering it to be popular if so. This is used because we feel that there can be no doubt that a business which receives a lot of relative checkins and reviews from a certain zipcode and category must be popular, regardless of whether those reviews are high or low, and regardless of whether the check-ins actually incur a financial gain for the company over time. In summary, Businesses with higher-than-average num\_reviews and num\_checkins can be classified as popular.

Using Queries, this can be done through first calculating the average num\_reviews and the average num\_checkins for each pair of zipcodes and business categories. Then, a join with Business. Lastly, compare the num\_reviews and num\_checkins of each business with the averages of the area, getting a list of businesses with their num\_reviews, num\_checkins, average num\_reviews, and average num\_checkins for each category, with a popularity status being assigned to assign the business as either popular or not popular. Note that in this, both the number of reviews (count) AND the number of check-ins must be above both the area/category average number of reviews AND average number of check-ins (not an OR operation here, in this metric for determining popularity).

**[See Figure 1-1 below for Queries/Descriptions/Steps].**

A metric for determining success we will be using is comparing whether the review\_rating of a certain business is greater than or less than the average review\_rating of its zipcode area of search, for businesses within a certain category. This is because the businesses with higher ratings and reviews are likely to be more successful than their counterparts with low ratings, with the ratings being both a possible cause and effect of the business’s success. Summary: Businesses with higher-than-average review ratings can be classified as successful.

Using Queries, this can be done through first calculating average review\_rating for each pair of zipcode and business, joining with Business, and lastly comparing calculated average with the review\_rating of each business, to get a list of businesses, their review\_ratings, the average review\_rating for the specific category and zipcode pair, and a determination of whether they are deemed successful or not successful from this.

**[See Figure 1-2 below for Queries/Descriptions/Steps].**

A metric for determining popularity we are considering using is determining whether a business in a certain category has greater or fewer check-ins on weekends (Friday/Saturdays) than the average in its zipcode. This can be a, albeit vast generalization, signifier that a business could be termed “more popular” than others its being compared to, while not infringing on the possible success status. The reasoning here, being, that businesses that see more checkins on the weekend can indicate a higher likelihood of a social event, whereas if a business saw more checkins on the weekdays, it may indicate success, but could be less socially popular. Note that this is a generalization, and is just one proposed metric of several we’ve come up with, and should be used in parallel with others for support. Summary: Businesses with higher-than-average weekend check-ins can be classified as popular.

Using Queries, this can be done through first calculating the average check-in count for Fridays and Saturdays for each pair of zipcodes and business categories, joining with Business check-in, and lastly comparing the calculated average with the weekend check-ins count of each business, to get a list of businesses with their Friday/Saturday check-in count, average weekend check-in count, and whether the specific business is to be deemed popular or not.

**[See Figure 1-3 below for Queries/Descriptions/Steps]**.

Our group proposes to use one or possibly both explained methods of determining popularity in parallel to one another, so there isn’t a blanket assignment. We also may look into incorporating certain checkin-times with popularity, if our method of checking weekend foot-traffic to assign popularity is acceptable.

**[Figure 1-1]:**

First, the average num\_reviews and the average num\_checkins for each pair of zipcodes and business categories can be calculated:

SELECT b.zipcode, cat.category, AVG(b.num\_reviews) AS avg\_num\_reviews, AVG(b.num\_checkins) AS avg\_num\_checkins

FROM Business b

JOIN Categories cat ON b.id = cat.business

GROUP BY b.zipcode, cat.category;

Second, join with Business:

SELECT b.id AS business\_id, b.num\_reviews AS business\_num\_reviews, b.num\_checkins AS business\_num\_checkins, a.avg\_num\_reviews, a.avg\_num\_checkins

FROM Business b

JOIN Categories cat ON b.id = cat.business

JOIN (SELECT b.zipcode, cat.category, AVG(b.num\_reviews) AS avg\_num\_reviews, AVG(b.num\_checkins) AS avg\_num\_checkins

FROM Business b

JOIN Categories cat ON b.id = cat.business

GROUP BY b.zipcode, cat.category) a

ON b.zipcode = a.zipcode AND cat.category = a.category;

Lastly, compare the num\_reviews and num\_checkins of each business with the averages of the area, getting a list of businesses with their num\_reviews, num\_checkins, average num\_reviews, and average num\_checkins for each category, with a popularity status being assigned to assign the business as either popular or not popular. Note that in this, both the number of reviews (count) AND the number of check-ins must be above both the area/category average number of reviews AND average number of check-ins (not an OR operation here, in this metric for determining popularity):

SELECT business\_id, business\_num\_reviews, business\_num\_checkins, avg\_num\_reviews, avg\_num\_checkins,

CASE WHEN business\_num\_reviews > avg\_num\_reviews AND business\_num\_checkins > avg\_num\_checkins THEN 'Popular'

ELSE 'Not Popular'

END AS popularity\_status

FROM (SELECT b.id AS business\_id, b.num\_reviews AS business\_num\_reviews, b.num\_checkins AS business\_num\_checkins, a.avg\_num\_reviews, a.avg\_num\_checkins

FROM Business b

JOIN Categories cat ON b.id = cat.business

JOIN (SELECT b.zipcode, cat.category, AVG(b.num\_reviews) AS avg\_num\_reviews, AVG(b.num\_checkins) AS avg\_num\_checkins

FROM Business b

JOIN Categories cat ON b.id = cat.business

GROUP BY b.zipcode, cat.category) a

ON b.zipcode = a.zipcode AND cat.category = a.category) subquery;

**[Figure 1-2]:**

First, calculate average review\_rating for each pair of zipcode and business:

SELECT b.zipcode, cat.category, AVG(b.review\_rating) AS avg\_review\_rating

FROM Business b

JOIN Categories cat ON b.id = cat.business

GROUP BY b.zipcode, cat.category;

Second, Join with Business:

SELECT b.id AS business\_id, b.review\_rating AS business\_review\_rating, a.avg\_review\_rating

FROM Business b

JOIN Categories cat ON b.id = cat.business

JOIN (SELECT b.zipcode, cat.category, AVG(b.review\_rating) AS avg\_review\_rating

FROM Business b

JOIN Categories cat ON b.id = cat.business

GROUP BY b.zipcode, cat.category) a

ON b.zipcode = a.zipcode AND cat.category = a.category;

Third, compare calculated average with the review\_rating of each business, to get a list of businesses, their review\_ratings, the average review\_rating for the specific category and zipcode pair, and a determination of whether they are deemed successful or not successful from this:

SELECT business\_id, business\_review\_rating, avg\_review\_rating,

CASE WHEN business\_review\_rating > avg\_review\_rating THEN 'Successful Business'

ELSE 'Business is not Successful'

END AS success\_status

FROM (SELECT b.id AS business\_id, b.review\_rating AS business\_review\_rating, a.avg\_review\_rating

FROM Business b

JOIN Categories cat ON b.id = cat.business

JOIN (SELECT b.zipcode, cat.category, AVG(b.review\_rating) AS avg\_review\_rating

FROM Business b

JOIN Categories cat ON b.id = cat.business

GROUP BY b.zipcode, cat.category) a

ON b.zipcode = a.zipcode AND cat.category = a.category) subquery;

**[Figure 1-3]:**

First, calculate average check-in count for Fridays and Saturdays for each pair of zipcodes and business categories:

SELECT b.zipcode, cat.category,

AVG(CASE WHEN DAYOFWEEK(c.day) = 6 OR DAYOFWEEK(c.day) = 7 THEN c.count ELSE 0 END) AS avg\_weekend\_checkins

FROM Checkins c

JOIN Business b ON c.business = b.id

JOIN Categories cat ON b.id = cat.business

GROUP BY b.zipcode, cat.category;

Second, join with Business check-in:

SELECT b.id AS business\_id, c.count AS business\_checkins, a.avg\_weekend\_checkins

FROM Checkins c

JOIN Business b ON c.business = b.id

JOIN Categories cat ON b.id = cat.business

JOIN (SELECT b.zipcode, cat.category,

AVG(CASE WHEN DAYOFWEEK(c.day) = 6 OR DAYOFWEEK(c.day) = 7 THEN c.count ELSE 0 END) AS avg\_weekend\_checkins

FROM Checkins c

JOIN Business b ON c.business = b.id

JOIN Categories cat ON b.id = cat.business

GROUP BY b.zipcode, cat.category) a

ON b.zipcode = a.zipcode AND cat.category = a.category;

Third, compare the calculated average with the weekend check-ins count of each business, to get a list of businesses with their Friday/Saturday check-in count, average weekend check-in count, and whether the specific business is to be deemed popular or not.

SELECT business\_id, business\_checkins, avg\_weekend\_checkins,

CASE WHEN business\_checkins > avg\_weekend\_checkins THEN 'Popular Business'

ELSE 'Business is not Popular'

END AS popularity\_status

FROM (SELECT b.id AS business\_id, c.count AS business\_checkins, a.avg\_weekend\_checkins

FROM Checkins c

JOIN Business b ON c.business = b.id

JOIN Categories cat ON b.id = cat.business

JOIN (SELECT b.zipcode, cat.category,

AVG(CASE WHEN DAYOFWEEK(c.day) = 6 OR DAYOFWEEK(c.day) = 7 THEN c.count ELSE 0 END) AS avg\_weekend\_checkins

FROM Checkins c

JOIN Business b ON c.business = b.id

JOIN Categories cat ON b.id = cat.business

GROUP BY b.zipcode, cat.category) a

ON b.zipcode = a.zipcode AND cat.category = a.category) subquery;