Executive Summary

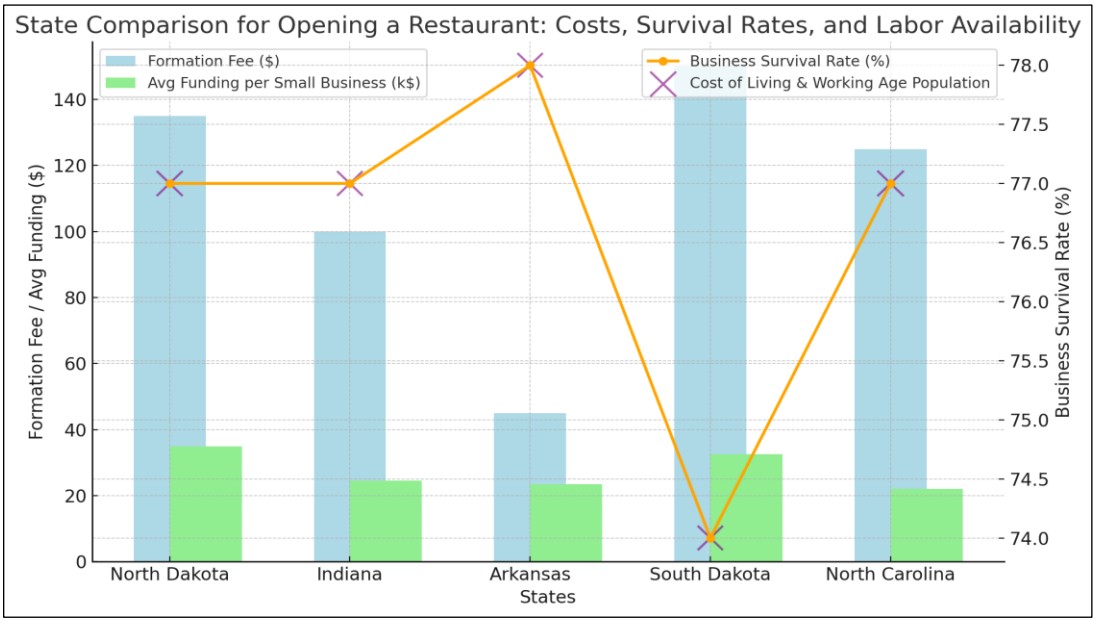
The goal of our team is to conduct market research for identifying counties in different states to open a new restaurant chain. The client has come with a specific request to open a smart restaurant which has chef-bots (meals prepared by robots), automated queue management systems and demand prediction systems. Given the client's specific request, we hired Forbes (reference here) to conduct primary research with an aim of identifying states which have the highest restaurant business growth potential.

Deep Dives and Analysis

State Level Restaurant Business Statistics

We used the forbes data to understand the state level dynamic of restaurant industry and further to drill down and evaluate a Overall rank of the state using the following parameters - Business Survival Rate, Avg Funding Per Small Business, Cost of Living Index, Business Costs and Climate, Financial Accessibility and Economy. We analyzed the data for the 50 states of US and defined the states-in-scope, basis the primary research following are the observations -

Chart 1

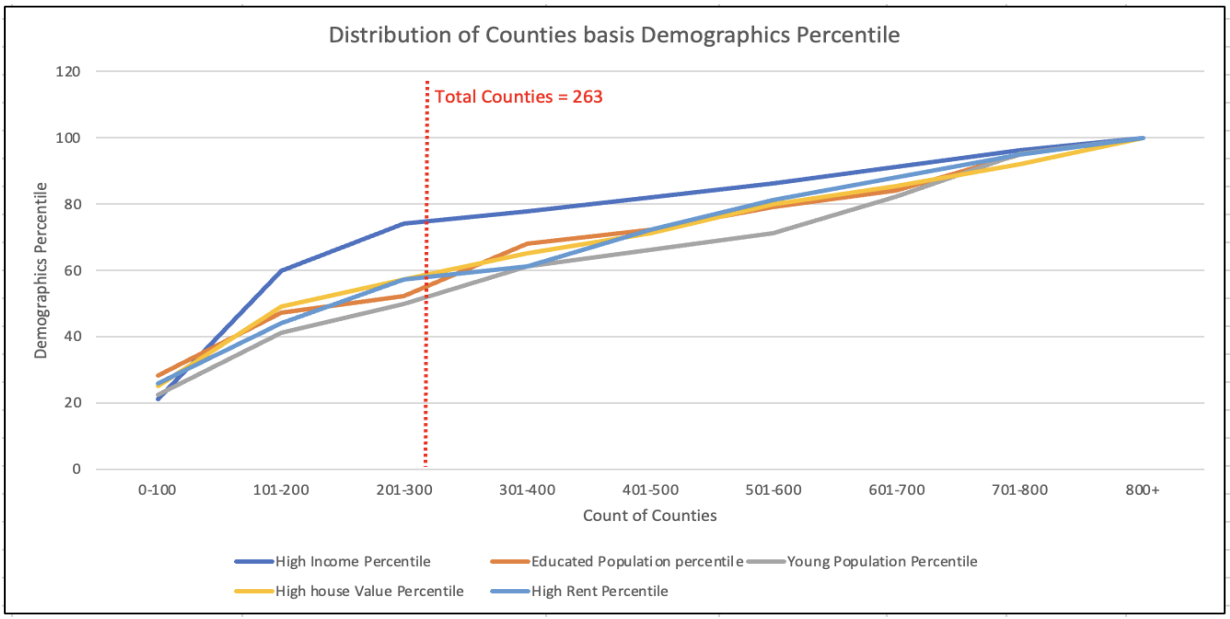


- **Formation Fee - Arkansas** has the lowest formation fee at $45, making it the most affordable state to start a business. **South Dakota** is the most expensive at $150, which may require more upfront capital.
- **Avg Funding per Small Business - North Dakota** provides the highest average funding at $34,894, making it ideal for businesses seeking substantial financial support. **North Carolina** offers the least funding at $22,019, which may limit capital availability.
- **Business Survival Rate - Arkansas** leads with the highest business survival rate at 78% indicating strong long-term prospects for new businesses. **South Dakota** has the lowest survival rate at 74%, which may pose a higher risk.
- **Cost of Living – Arkansas** has the lowest cost of living (0.89), providing an affordable environment for both businesses and employees. **North Carolina** has the highest cost of living (0.96), which may impact operating costs
- **Working Age Population - North Carolina** has the largest working-age population at 65%, offering the best labor availability for new businesses. **South Dakota** has the lowest percentage of the working-age population at 62%, potentially limiting the labor pool.

County Level Restaurant Statistics

We analyzed data of different counties to correlate with population with a.) income > $75K, b) Age group between 15 and 65 years, c) Education of high school and above, d) high monthly rental (>$1000), e.) High house value (> $100K). Further, we used percentile ranks of these demographic parameters to identify counties suitable for investing in new restaurants. As per the output of percentile rank logic (show in Diagram 2), **we observed that 263 counties (top 20%ile) contribute to 80% of the demographic parameters.**

Diagram 2



We refined the list of 263 counties to develop a more viable launch plan. Using a weighted moving average model with the following weights: high income population (0.3), high house value (0.2), high rent (0.2), young population (0.1), total restaurants (0.1), and mean dwell time (0.1), **we narrowed the list to the top 10 counties** with the highest potential for opening a smart restaurant as shown in Table 1.

Table 1:

County	State	Adult Population	Educated population	High Income Pop	High Rental Pop	High house Value Pop	Restaurant Visiting Pop	Avg Total Dwell Time	# Total Restaurants	County_Potential_Business_Score
Wake County	NC	387176	590549	225741	80656	249691	1535156	31.8392969	801	0.98045977
Mecklenburg County	NC	412222	562919	196366	99022	219157	1479226	31.4301482	830	0.975095785
Marion County	IN	354593	403316	123884	29982	151221	1175327	30.9636248	787	0.969731801
Forsyth County	NC	125224	171790	52753	9005	75350	664890	31.7974957	291	0.957471264
Hamilton County	IN	105622	181425	76497	15505	90198	483707	28.3854069	269	0.949808429
Guilford County	NC	184801	246893	76643	13270	100417	784871	24.4784084	410	0.948659004
St. Joseph County	IN	93561	115346	35891	3483	46837	354150	38.4912485	216	0.94789272
Durham County	NC	124432	164467	55611	25357	67289	320058	30.4106509	214	0.946360153
Cumberland County	NC	134802	142789	37006	12801	46135	502563	29.5742063	318	0.941762452
Allen County	IN	125715	160048	53657	3983	69810	479970	26.775653	269	0.939846743

Underserved Areas

We identified that not all potential counties are covered using the above apporachcatered to outlying counties such as Case and Tippecanoe where dwell time is high but total restaurants are

low by only focusing on counties with high income and educated population. The list can be considered as launch plan for phase 2.

county_name	state_	young_pop	high_inc_pop	educated_pop	high_rent_pop	high_house_value_pop	visitors_to_restaurant	total_dwell_timing	total_restaurants	rank
Cass County	ND	75089	33975	90668	7670	37126	181054	42.28427788	138	11
Tippecanoe County	IN	91055	24286	74023	5573	31327	320976	36.97491039	141	16
Orange County	NC	58392	26542	75601	8890	31158	116577	32.2593985	67	23
Catawba County	NC	47693	21623	67632	1407	33627	326940	33.98674912	143	25
Brunswick County	NC	29678	22844	73580	2970	40903	122531	36.6595092	83	29
Harnett County	NC	48082	16210	56524	3751	22890	158454	35.73913043	76	32
Hancock County	IN	23542	15023	35674	1135	21286	92354	32.48717949	50	48
Ward County	ND	28575	12597	29255	3892	14397	79809	69.18981481	55	49
Chatham County	NC	17733	14153	38998	1251	20341	50201	36.2027972	36	51
Warrick County	IN	18168	12887	29990	727	16436	57462	45.03097345	29	57

Appendix

Query 1 – Finalized Query to compute Top 10 Counties for opening restaurant.

```
WITH b AS (
-- Get distinct brand IDs for restaurants
SELECT DISTINCT safegraph_brand_id
FROM `mod-group-13-project.safegraph.brands`
WHERE sub_category IN ('Full-Service Restaurants', 'Limited-Service Restaurants')
),
v AS (
-- Aggregate visit data at the county level, including state_fips
SELECT SUBSTRING(poi_cbg, 1, 2) AS state_fips, -- Extract the state_fips from poi_cbg
SUBSTRING(poi_cbg, 3, 3) AS county_fips,
SUM(raw_visit_counts) AS total_visits,
AVG(median_dwell) AS avg_dwell,
COUNT(DISTINCT safegraph_place_id) AS total_restaurants
FROM `mod-group-13-project.safegraph.visits`
WHERE safegraph_brand_ids IN (SELECT safegraph_brand_id FROM b)
GROUP BY state_fips, county_fips
),
d AS (
-- Aggregate demographic data at the county level, including state_fips
SELECT SUBSTRING(cbg, 1, 2) AS state_fips, -- Extract the state_fips from cbg
SUBSTRING(cbg, 3, 3) AS county_fips,
SUM(`pop_m_15-17` + `pop_m_18-19` + `pop_m_20` + `pop_m_21` + `pop_m_22-24` +
`pop_m_25-29` + `pop_m_30-34` + `pop_m_35-39` +
`pop_f_15-17` + `pop_f_18-19` + `pop_f_20` + `pop_f_21` + `pop_f_22-24` +
`pop_f_25-29` + `pop_f_30-34` + `pop_f_35-39`) AS total_pop_15_to_39,
SUM(`edu_ged` + `edu_coll_lt1` + `edu_coll_gte1` + `edu_coll_assoc` + `edu_coll_bach`
+
`edu_coll_mast` + `edu_coll_prof` + `edu_coll_doc`) AS
total_pop_with_edu_greater_than_hs,
```

```
SUM(`inc_75-100` + `inc_100-125` + `inc_125-150` + `inc_150-200` + `inc_gte200`) AS
total_inc_greater_than_75K,
SUM(`rent_1000-1250` + `rent_1250-1500` + `rent_1500-1999` + `rent_2000-2500` +
`rent_2500-3000` + `rent_3000-3500` + `rent_gte3500`) AS total_rent_greater_than_1K,
SUM(`value_100-125` + `value_125-150` + `value_150-175` + `value_175-200` +
`value_200-250` + `value_250-300` + `value_300-400` + `value_400-500` +
`value_500-750` + `value_750-1000` + `value_1000-1500` + `value_1500-2000` +
`value_gte2000`) AS total_house_value_greater_than_100K
FROM `mod-group-13-project.safegraph.cbg_demographics`
GROUP BY state_fips, county_fips
),
f AS (
-- Include state_fips in the county mapping
SELECT DISTINCT state, state_fips, county_fips, county
FROM `mod-group-13-project.safegraph.cbg_fips`
),
r AS (
SELECT `Overall Ranking`, `State`, `State Code`, `Business Costs`, `Business Climate`,
`Financial Accessibility`, `Economy`, `Workforce`, `Weighted Average Rank`, `Scaled
Score`
FROM `mod-group-13-project.safegraph.restaurant_ranks_state_wise`
),
combined_data AS (
-- Join aggregated data
SELECT f.county AS county_name,
f.state AS state_name,
f.state_fips,
f.county_fips,
d.total_pop_15_to_39 AS young_pop,
d.total_pop_with_edu_greater_than_hs AS edu_pop,
d.total_inc_greater_than_75K AS high_inc_pop,
d.total_pop_with_edu_greater_than_hs AS educated_pop,
d.total_rent_greater_than_1K AS high_rent_pop,
d.total_house_value_greater_than_100K AS high_house_value_pop,
v.total_visits AS visitors_to_restaurant,
v.avg_dwell AS total_dwell_timing,
v.total_restaurants,
r.`Overall Ranking`,
r.`Business Costs`,
r.`Business Climate`,
r.`Weighted Average Rank`,
r.`Financial Accessibility`,
r.`Economy`,
r.`Workforce`,
r.`Scaled Score`
FROM v
JOIN d ON v.state_fips = d.state_fips AND v.county_fips = d.county_fips
JOIN f ON v.state_fips = f.state_fips AND v.county_fips = f.county_fips
JOIN r ON f.state = r.`State Code`
```

```
where r.`Overall Ranking` between 1 and 5
)
-- Final query: Calculate percentiles and add the weighted_restaurant_potential_score
SELECT *,
PERCENT_RANK() OVER (ORDER BY young_pop) AS young_pop_percentile,
PERCENT_RANK() OVER (ORDER BY edu_pop) AS edu_pop_percentile,
PERCENT_RANK() OVER (ORDER BY high_inc_pop) AS high_inc_pop_percentile,
PERCENT_RANK() OVER (ORDER BY educated_pop) AS educated_pop_percentile,
PERCENT_RANK() OVER (ORDER BY high_rent_pop) AS high_rent_pop_percentile,
PERCENT_RANK() OVER (ORDER BY high_house_value_pop) AS
high_house_value_pop_percentile,
PERCENT_RANK() OVER (ORDER BY visitors_to_restaurant) AS
visitors_to_restaurant_percentile,
PERCENT_RANK() OVER (ORDER BY total_dwell_timing) AS total_dwell_timing_percentile,
PERCENT_RANK() OVER (ORDER BY total_restaurants) AS total_restaurants_percentile,
-- Add the weighted restaurant potential score
(PERCENT_RANK() OVER (ORDER BY young_pop) * 0.2 +
PERCENT_RANK() OVER (ORDER BY high_inc_pop) * 0.3 +
PERCENT_RANK() OVER (ORDER BY educated_pop) * 0.3 +
PERCENT_RANK() OVER (ORDER BY visitors_to_restaurant) * 0.1 +
PERCENT_RANK() OVER (ORDER BY total_dwell_timing) * 0.1) AS
weighted_restaurant_potential_score
FROM combined_data
ORDER BY weighted_restaurant_potential_score DESC
LIMIT 10;
```

Query 2 – To fetch data for identifying underserved areas

```
WITH b AS (
-- Get distinct brand IDs for restaurants
SELECT DISTINCT safegraph_brand_id
FROM `mod-group-13-project.safegraph.brands`
WHERE sub_category IN ('Full-Service Restaurants', 'Limited-Service Restaurants')
),
v AS (
-- Aggregate visit data at the county level, including state_fips
SELECT SUBSTRING(poi_cbg, 1, 2) AS state_fips, -- Extract the state_fips from poi_cbg
SUBSTRING(poi_cbg, 3, 3) AS county_fips,
SUM(raw_visit_counts) AS total_visits,
AVG(median_dwell) AS avg_dwell,
COUNT(DISTINCT safegraph_place_id) AS total_restaurants
FROM `mod-group-13-project.safegraph.visits`
WHERE safegraph_brand_ids IN (SELECT safegraph_brand_id FROM b)
GROUP BY state_fips, county_fips
),
d AS (
-- Aggregate demographic data at the county level, including state_fips
SELECT SUBSTRING(cbg, 1, 2) AS state_fips, -- Extract the state_fips from cbg
SUBSTRING(cbg, 3, 3) AS county_fips,
```

```
SUM(`pop_m_15-17` + `pop_m_18-19` + `pop_m_20` + `pop_m_21` + `pop_m_22-24` +
`pop_m_25-29` + `pop_m_30-34` + `pop_m_35-39` +
`pop_f_15-17` + `pop_f_18-19` + `pop_f_20` + `pop_f_21` + `pop_f_22-24` +
`pop_f_25-29` + `pop_f_30-34` + `pop_f_35-39`) AS total_pop_15_to_39,
SUM(`edu_ged` + `edu_coll_lt1` + `edu_coll_gte1` + `edu_coll_assoc` + `edu_coll_bach`
+
`edu_coll_mast` + `edu_coll_prof` + `edu_coll_doc`) AS
total_pop_with_edu_greater_than_hs,
SUM(`inc_75-100` + `inc_100-125` + `inc_125-150` + `inc_150-200` + `inc_gte200`) AS
total_inc_greater_than_75K,
SUM(`rent_1000-1250` + `rent_1250-1500` + `rent_1500-1999` + `rent_2000-2500` +
`rent_2500-3000` + `rent_3000-3500` + `rent_gte3500`) AS total_rent_greater_than_1K,
SUM(`value_100-125` + `value_125-150` + `value_150-175` + `value_175-200` +
`value_200-250` + `value_250-300` + `value_300-400` + `value_400-500` +
`value_500-750` + `value_750-1000` + `value_1000-1500` + `value_1500-2000` +
`value_gte2000`) AS total_house_value_greater_than_100K
FROM `mod-group-13-project.safegraph.cbg_demographics`
GROUP BY state_fips, county_fips
),
f AS (
-- Include state_fips in the county mapping
SELECT DISTINCT state, state_fips, county_fips, county
FROM `mod-group-13-project.safegraph.cbg_fips`
),
r AS (
SELECT `Overall Ranking`, `State`, `State Code`, `Business Costs`, `Business Climate`,
`Financial Accessibility`, `Economy`, `Workforce`, `Weighted Average Rank`, `Scaled
Score`
FROM `mod-group-13-project.safegraph.restaurant_ranks_state_wise`
),
combined_data AS (
-- Join aggregated data
SELECT f.county AS county_name,
f.state AS state_name,
f.state_fips,
f.county_fips,
d.total_pop_15_to_39 AS young_pop,
d.total_pop_with_edu_greater_than_hs AS edu_pop,
d.total_inc_greater_than_75K AS high_inc_pop,
d.total_pop_with_edu_greater_than_hs AS educated_pop,
d.total_rent_greater_than_1K AS high_rent_pop,
d.total_house_value_greater_than_100K AS high_house_value_pop,
v.total_visits AS visitors_to_restaurant,
v.avg_dwell AS total_dwell_timing,
v.total_restaurants,
r.`Overall Ranking`,
r.`Business Costs`,
r.`Business Climate`,
r.`Weighted Average Rank`,
```

```
r.`Financial Accessibility`,
r.`Economy`,
r.`Workforce`,
r.`Scaled Score`
FROM v
JOIN d ON v.state_fips = d.state_fips AND v.county_fips = d.county_fips
JOIN f ON v.state_fips = f.state_fips AND v.county_fips = f.county_fips
JOIN r ON f.state = r.`State Code`
),
percentile_data AS (
-- Calculate percentiles for all key metrics
SELECT *,
PERCENT_RANK() OVER (ORDER BY total_dwell_timing) AS dwell_time_percentile,
PERCENT_RANK() OVER (ORDER BY total_restaurants) AS restaurant_percentile,
PERCENT_RANK() OVER (ORDER BY young_pop) AS young_pop_percentile,
PERCENT_RANK() OVER (ORDER BY high_inc_pop) AS high_inc_pop_percentile,
PERCENT_RANK() OVER (ORDER BY educated_pop) AS educated_pop_percentile,
PERCENT_RANK() OVER (ORDER BY visitors_to_restaurant) AS
visitors_to_restaurant_percentile
FROM combined_data
),
ranked_data AS (
-- Calculate the weighted score and rank counties
SELECT *,
(young_pop_percentile * 0.2 +
high_inc_pop_percentile * 0.3 +
educated_pop_percentile * 0.3 +
visitors_to_restaurant_percentile * 0.1 +
dwell_time_percentile * 0.1) AS weighted_restaurant_potential_score,
ROW_NUMBER() OVER (ORDER BY (young_pop_percentile * 0.2 +
high_inc_pop_percentile * 0.3 +
educated_pop_percentile * 0.3 +
visitors_to_restaurant_percentile * 0.1 +
dwell_time_percentile * 0.1) DESC) AS rank
FROM percentile_data
)
-- Final query: Filter for counties where dwell time is high, restaurant count is low,
excluding top 10
SELECT *
FROM ranked_data
WHERE rank > 10 -- Exclude top 10 counties
AND dwell_time_percentile > 0.8 -- High dwell time (above 80th percentile)
AND restaurant_percentile < 0.5 -- Low restaurant count (below 50th percentile)
ORDER BY dwell_time_percentile DESC, restaurant_percentile ASC;
```

Query 3 - Query to fetch data for count of restaurants and places for a given CBG

```
SELECT v.poi_cbg,
COUNT(DISTINCT p.safegraph_place_id) AS num_places,
```

```
COUNT(DISTINCT p.safegraph_brand_ids) AS num_brands
FROM `mod-group-13-project.safegraph.visits` v
JOIN `mod-group-13-project.safegraph.places` p
ON v.safegraph_place_id = p.safegraph_place_id
GROUP BY v.poi_cbg
ORDER BY num_places DESC
```

Query 4 – To fetch forbes data -

```
SELECT `Overall Ranking`, `State`, `State Code`, `Business Costs`, `Business Climate`,
`Financial Accessibility`, `Economy`, `Workforce`, `Weighted Average Rank`, `Scaled
Score`
FROM `mod-group-13-project.safegraph.restaurant_ranks_state_wise`;
```

Query 4 – Query to fetch data for foot traffic in different geographic areas (CBGs) over time

```
SELECT v.poi_cbg,
EXTRACT(YEAR FROM v.date_range_start) AS year,
EXTRACT(MONTH FROM v.date_range_start) AS month,
SUM(v.raw_visit_counts) AS total_visits
FROM `mod-group-13-project.safegraph.visits` v
GROUP BY v.poi_cbg, year, month
ORDER BY total_visits DESC
LIMIT 100;
```

Query 5 – Query to fetch the average dwell time in each CBG by examining the bucketed dwell time ranges.

```
WITH dwell_time_sums AS (
SELECT v.poi_cbg,
SUM(CAST(IFNULL(JSON_EXTRACT_SCALAR(REPLACE(v.bucketed_dwell_times, '"<5"',
'"less_than_5"'), '$.less_than_5'), '0') AS INT64)) AS dwell_time_under_5,
SUM(CAST(IFNULL(JSON_EXTRACT_SCALAR(REPLACE(v.bucketed_dwell_times, '"5-20"',
'"five_to_twenty"'), '$.five_to_twenty'), '0') AS INT64)) AS dwell_time_5_to_20,
SUM(CAST(IFNULL(JSON_EXTRACT_SCALAR(REPLACE(v.bucketed_dwell_times, '"21-60"',
'"twenty_one_to_sixty"'), '$.twenty_one_to_sixty'), '0') AS INT64)) AS
dwell_time_21_to_60,
SUM(CAST(IFNULL(JSON_EXTRACT_SCALAR(REPLACE(v.bucketed_dwell_times, '"61-240"',
'"sixty_to_twoforty"'), '$.sixty_to_twoforty'), '0') AS INT64)) AS
dwell_time_61_to_240,
SUM(CAST(IFNULL(JSON_EXTRACT_SCALAR(REPLACE(v.bucketed_dwell_times, '">240"',
'"greater_than_240"'), '$.greater_than_240'), '0') AS INT64)) AS dwell_time_above_240
FROM `mod-group-13-project.safegraph.visits` v
GROUP BY v.poi_cbg
)
SELECT poi_cbg,
```

```
(dwell_time_under_5 + dwell_time_5_to_20 + dwell_time_21_to_60 + dwell_time_61_to_240
+ dwell_time_above_240) AS total_dwell_time,
dwell_time_under_5, dwell_time_5_to_20, dwell_time_21_to_60, dwell_time_61_to_240,
dwell_time_above_240
FROM dwell_time_sums
ORDER BY total_dwell_time DESC;
```

Query 4 – This query fetches data for foot traffic variation throughout the day in different regions

```
SELECT v.poi_cbg,
EXTRACT(HOUR FROM v.date_range_start) AS hour,
ARRAY(SELECT value FROM UNNEST(SPLIT(REPLACE(JSON_EXTRACT(v.popularity_by_hour, '$'),
'[', ''), ']')) AS value) AS traffic_by_hour
FROM `mod-group-13-project.safegraph.visits` v
LIMIT 100;
```

Query 6 – This query fetched data to identify regions with high traffic but few businesses

```
WITH traffic_and_business AS (
SELECT v.poi_cbg,
SUM(v.raw_visit_counts) AS total_visits,
COUNT(DISTINCT p.safegraph_place_id) AS num_places
FROM `mod-group-13-project.safegraph.visits` v
JOIN `mod-group-13-project.safegraph.places` p
ON v.safegraph_place_id = p.safegraph_place_id
GROUP BY v.poi_cbg
)
SELECT poi_cbg, total_visits, num_places
FROM traffic_and_business
WHERE num_places < 5 -- Low business presence
ORDER BY total_visits DESC
LIMIT 100;
```

Query 7 – This query fetches data for conducting EDA for fetching restaurant datas and count of people (basis sex) visiting to these restaurants.

```
with b as (

select distinct safegraph_brand_id, brand_name, top_category, sub_category
FROM `mod-group-13-project.safegraph.brands` b
where sub_category in ('Full-Service Restaurants', 'Limited-Service Restaurants')
),
v as (
select safegraph_brand_ids, poi_cbg, sum(raw_visit_counts) as total_visits
from `mod-group-13-project.safegraph.visits` v
group by 1,2
),
```

```
d as (
select distinct cbg, sum(pop_m_total) as male, sum(pop_f_total) as female
from `mod-group-13-project.safegraph.cbg_demographics` d
group by 1
)

select d.cbg, b.brand_name, b.top_category, b.sub_category, v.total_visits, d.male,
d.female
from b join v on b.safegraph_brand_id = v.safegraph_brand_ids
join d on v.poi_cbg = d.cbg
```

Query 8 – Finalized Query which fetches the trend of traffic over different month and identifies areas with low business count but increase in traffic.

```
WITH traffic_trends AS (
-- Step 1: Calculate the trend in foot traffic over time for each CBG
SELECT p.safegraph_place_id,
v.poi_cbg,
v.date_range_start,
SUM(v.raw_visit_counts) AS total_visits,
EXTRACT(MONTH FROM v.date_range_start) AS month
FROM `mod-group-13-project.safegraph.visits` v
JOIN `mod-group-13-project.safegraph.places` p
ON v.safegraph_place_id = p.safegraph_place_id
GROUP BY p.safegraph_place_id, v.poi_cbg, month, v.date_range_start
),
business_density AS (
-- Step 2: Count the number of businesses in each geographic area (CBG) using the
NAICS code from brands
SELECT p.safegraph_place_id, v.poi_cbg,
COUNT(DISTINCT b.safegraph_brand_id) AS num_businesses
FROM `mod-group-13-project.safegraph.places` p
JOIN `mod-group-13-project.safegraph.brands` b
ON p.safegraph_brand_ids = b.safegraph_brand_id
JOIN `mod-group-13-project.safegraph.visits` v
ON p.safegraph_place_id = v.safegraph_place_id
GROUP BY p.safegraph_place_id, v.poi_cbg
),
recent_trends AS (
-- Step 3: Identify regions with increasing foot traffic
SELECT poi_cbg,
SUM(CASE WHEN month = 1 THEN total_visits ELSE 0 END) AS visits_Jan,
SUM(CASE WHEN month = 2 THEN total_visits ELSE 0 END) AS visits_Feb,
SUM(CASE WHEN month = 3 THEN total_visits ELSE 0 END) AS visits_Mar,
SUM(CASE WHEN month = 4 THEN total_visits ELSE 0 END) AS visits_Apr,
FROM traffic_trends
GROUP BY poi_cbg
```

```sql
HAVING visits_Apr > visits_Mar AND visits_Mar > visits_Feb and visits_Feb > visits_Jan
),
limited_business_areas AS (
-- Step 4: Filter regions with low business density (e.g., fewer than 5 businesses)
SELECT poi_cbg
FROM business_density
WHERE num_businesses < 5
)
-- Step 5: Combine results to get geographic areas with upward traffic trends but
limited business presence
SELECT distinct r.poi_cbg, r.visits_Jan, r.visits_Feb, r.visits_Mar, r.visits_Apr,
sum(bd.num_businesses)
FROM recent_trends r
JOIN limited_business_areas lba
ON r.poi_cbg = lba.poi_cbg
JOIN business_density bd
ON r.poi_cbg = bd.poi_cbg
group by 1,2,3,4,5
```