

swiggy-data-analysis

August 10, 2024

```
[15]: import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import mysql.connector
import numpy as np

db = mysql.connector.connect(host = "localhost",
                             username = "root",
                             password = "1012",
                             database = "swiggy")

cur = db.cursor()
```

1. Question: Calculate the average delivery time for each food type across all cities and rank them from the fastest to the slowest delivery time.

```
[15]: query = """SELECT Food_type, round(AVG(Delivery_time),2) AS Avg_Delivery_Time
FROM swiggy
GROUP BY Food_type
ORDER BY Avg_Delivery_Time ASC"""
cur.execute(query)
data = cur.fetchall()
df = pd.DataFrame(data, columns = ["Food_Type", "Avg Delivery time"])
df
```

```
[15]:
```

	Food_Type	Avg Delivery time
0	Beverages,Chinese,Punjabi	22.00
1	Desserts Ice Cream Chaat	24.00
2	Punjabi,Thalis,Combo,North Indian,Mughlai,Guja...	24.00
3	Biryani,Tandoor,Chinese,Indian,Kebabs,Desserts	24.50
4	Street Food,Desserts,Beverages	25.00
...
3729	Italian,Chinese,Continental	95.00
3730	Chinese,Biryani,Tandoor	95.00
3731	North Indian,Mughlai,Chinese,Biryani	95.00
3732	Fast Food,Combo,Juices	96.00

3733 Indian,Rajasthani,Punjabi,Thalis,Desserts,Beve...

96.00

[3734 rows x 2 columns]

2. Question: Identify the area with the most expensive average price per dish for a specific food type (e.g., “Chinese”).

```
[14]: query = """SELECT Area, round(AVG(Price),2) AS Avg_Price
FROM swiggy
WHERE Food_type = 'Chinese'
GROUP BY Area
ORDER BY Avg_Price DESC
LIMIT 1"""
cur.execute(query)
data = cur.fetchall()
df = pd.DataFrame(data, columns = ["Area","Avg Priced"])
df
```

```
[14]:      Area Avg Priced
0  Bandra Area    2000.00
```

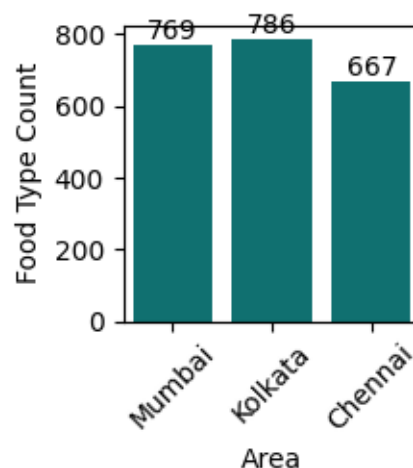
3. Question: Determine which city has the highest average rating for restaurants that have a delivery time of less than 30 minutes.

```
[13]: query = """SELECT City, AVG(Avg_ratings) AS Avg_Rating
FROM swiggy
WHERE Delivery_time < 30
GROUP BY City
ORDER BY Avg_Rating DESC
LIMIT 1;"""
cur.execute(query)
data = cur.fetchall()
df = pd.DataFrame(data, columns = ["City","Avg Rating"])
df
```

```
[13]:      City Avg Rating
0  Chennai     4.65
```

4 4. Question: Find the top 3 cities with the most diverse range of food types available.

```
[7]: query = """SELECT City, COUNT(DISTINCT Food_type) AS Food_Type_Count
FROM swiggy
GROUP BY City
ORDER BY Food_Type_Count DESC
LIMIT 3"""
cur.execute(query)
data = cur.fetchall()
df = pd.DataFrame(data, columns = ["Area", "Food Type Count"])
df = df.sort_values(by = "Area", ascending = False)
plt.figure(figsize = (2,2))
ax = sns.barplot(x = df["Area"], y = df["Food Type Count"], color = "#008080")
ax.bar_label(ax.containers[0])
plt.xticks(rotation = 45)
plt.show()
```



5 5. Question: List the top 10 restaurants that have the highest total ratings but an average rating of less than 4.0.

```
[8]: query = """SELECT Restaurant, City, Avg_ratings, Total_ratings
FROM swiggy
WHERE Avg_ratings < 4.0
ORDER BY Total_ratings DESC
LIMIT 10"""
cur.execute(query)
data = cur.fetchall()
```

```
df = pd.DataFrame(data, columns = ["Restaurant", "City", "Avg_ratings",  
    ↪ "Total_ratings"])  
df
```

```
[8]:
```

	Restaurant	City	Avg_ratings	Total_ratings
0	Mehfil	Hyderabad	3.9	10000
1	Bawarchi	Hyderabad	3.9	10000
2	Kolkata@99	Kolkata	3.8	5000
3	Cafe 555 & Aqeeq Restaurant	Hyderabad	3.9	5000
4	Capital Multi Cuisine Restaurant	Hyderabad	3.8	5000
5	Zaika	Mumbai	3.9	1000
6	Royal Tiffin Centre	Hyderabad	3.8	1000
7	Pakwan	Delhi	3.7	1000
8	Shri Swami Samartha Pure Veg	Pune	3.7	1000
9	Wah Ji Wah (Budhvihar)	Delhi	3.8	1000

6 Question 6: Find the Average Price and Delivery Time for Each Food Type in Each City, and Rank the Results.

```
[11]: query = """WITH Avg_Price_Time AS (  
    SELECT City, Food_type, Round(AVG(Price),2) AS Avg_Price,  
    ↪ROUND(AVG(Delivery_time),2) AS Avg_Delivery_Time  
    FROM Swiggy  
    GROUP BY City, Food_Type)  
SELECT City, Food_type, Avg_Price, Avg_Delivery_Time,  
    RANK() OVER (PARTITION BY City ORDER BY Avg_Price DESC) AS Price_Rank,  
    RANK() OVER (PARTITION BY City ORDER BY Avg_Delivery_Time ASC) AS  
    ↪Delivery_Rank  
FROM Avg_Price_Time;"""  
cur.execute(query)  
data = cur.fetchall()  
df = pd.DataFrame(data, columns = ["City", "Food_type", "Avg_price",  
    ↪ "Avg_Delivery_Time", "Price_Rank", "Delivery_Rank"])  
df
```

```
[11]:
```

	City	Food_type	Avg_price \
0	Ahmedabad	Ice Cream,Ice Cream Cakes,Desserts	250.00
1	Ahmedabad	Punjabi,Indian,Chinese	450.00
2	Ahmedabad	Juices,Beverages,Desserts	200.00
3	Ahmedabad	South Indian,Indian,Snacks,Combo,Thalis,Desserts	150.00
4	Ahmedabad	Indian,Chinese,Fast Food	200.00
...
5139	Surat	Chinese,Pizzas	200.00
5140	Surat	Combo,Thalis,Desserts,Indian,Biryani	250.00
5141	Surat	Pan-Asian,Burmese,Chinese,Sushi	200.00

5142	Surat	Gujarati,Indian,Fast Food,Pizzas,Beverages	300.00
5143	Surat	Thalis,Snacks,Indian,Combo	150.00

	Avg_Delivery_Time	Price_Rank	Delivery_Rank
0	25.00	210	1
1	26.00	62	2
2	27.00	267	3
3	27.00	366	3
4	27.00	267	3
...
5139	71.00	189	297
5140	74.00	137	299
5141	75.00	189	300
5142	77.00	67	301
5143	82.00	276	302

[5144 rows x 6 columns]

7 Question 7: Identify the Restaurants with Price Variations Greater than 20% Between Cities.

```
[12]: query = """WITH Price_Variation AS (
        SELECT Restaurant, MAX(Price) AS Max_Price, MIN(Price) AS Min_Price,
               round((MAX(Price) - MIN(Price)) / MIN(Price) * 100,2) AS
        ↪Price_Diff_Percent
        FROM swiggy
        GROUP BY Restaurant)
SELECT Restaurant, Max_Price, Min_Price, Price_Diff_Percent
FROM Price_Variation
WHERE Price_Diff_Percent > 20"""
cur.execute(query)
data = cur.fetchall()
df = pd.DataFrame(data, columns = ["Restaurant", "Max_Price", "Min_Price",
↪"Price_Diff_Percent"])
df
```

```
[12]:
```

	Restaurant	Max_Price	Min_Price	\
0	Treat	800	350	
1	Bangaliana	300	200	
2	Aalishan Restaurant & Caterer	290	200	
3	Kitchen Of Joy	250	200	
4	Chinese Pavilion	850	400	
..	
178	Creamy Heaven	300	150	
179	Kutchi King	200	100	
180	Dairy Don	200	120	

181		Cake Factory	500	100
182	Gourmet Ice Cream Cakes By Baskin Robbins		400	200

	Price_Diff_Percent
0	128.57
1	50.00
2	45.00
3	25.00
4	112.50
..	...
178	100.00
179	100.00
180	66.67
181	400.00
182	100.00

[183 rows x 4 columns]

8 Question 8: Track the Top 5 Restaurants with the Highest Ratings Over Time.

```
[16]: query = """WITH Top_Restaurants AS (
        SELECT Restaurant, MAX(Avg_Ratings) AS Max_Rating
        FROM swiggy
        GROUP BY Restaurant
        ORDER BY Max_Rating DESC
        LIMIT 5)
        SELECT S.Restaurant, S.Area, S.City, S.Avg_Ratings
        FROM swiggy S
        JOIN Top_Restaurants T ON S.Restaurant = T.Restaurant
        ORDER BY S.Restaurant, S.Avg_Ratings DESC"""
cur.execute(query)
data = cur.fetchall()
df = pd.DataFrame(data, columns = ["Restaurant", "Area", "City", "Avg_ratings"])
df
```

	Restaurant	Area	City	Avg_ratings
0	Fat Tiger	Nsp	Delhi	5.0
1	Nivala Apka Apna	Andheri East	Mumbai	5.0
2	Shree Krishna Juice Bar	Shivajinagar	Pune	5.0
3	Wallonia Waffle Co.	Himayath Nagar	Hyderabad	5.0
4	Zorro - Milkshakes And Thickshakes	Himayath Nagar	Hyderabad	5.0

9 Question 9: Detailed Evaluation of Food Type Preferences and Their Impact on Delivery Time, Price, and Ratings Across Multiple Areas

```
[23]: query = """WITH Area_Food_Type_Popularity AS (
                                -- Calculate the popularity of each food_
                                ↪type in each area based on total ratings
                                SELECT Area, Food_type,
                                    SUM(Total_ratings) AS Food_type_Popularity,
                                    AVG(Price) AS Avg_Price,
                                    AVG(Delivery_time) AS Avg_Delivery_Time,
                                    AVG(Avg_ratings) AS Avg_Rating
                                FROM swiggy
                                GROUP BY Area, Food_type),
Food_Type_Rankings AS (
                                -- Rank food types within each area based_
                                ↪on popularity, delivery time, and ratings
                                SELECT Area, Food_type, Food_Type_Popularity, Avg_Price, Avg_Delivery_Time,
                                ↪Avg_Rating,
                                    RANK() OVER (PARTITION BY Area ORDER BY Food_Type_Popularity DESC)
                                ↪AS Popularity_Rank,
                                    RANK() OVER (PARTITION BY Area ORDER BY Avg_Delivery_Time ASC) AS
                                ↪Delivery_Time_Rank,
                                    RANK() OVER (PARTITION BY Area ORDER BY Avg_Rating DESC) AS
                                ↪Rating_Rank
                                FROM Area_Food_Type_Popularity),
Top_Food_Types_Per_Area AS (
                                -- Select the top food type for each area_
                                ↪based on a combined ranking score
                                SELECT Area, Food_type,
                                    Popularity_Rank + Delivery_Time_Rank + Rating_Rank AS Combined_Rank,
                                    Food_Type_Popularity, Avg_Price, Avg_Delivery_Time, Avg_Rating
                                FROM Food_Type_Rankings
                                WHERE Popularity_Rank <= 3
                                ORDER BY Area, Combined_Rank ASC),
Top_Restaurants_Per_Food_Type AS (
                                -- Identify the top-performing restaurant_
                                ↪for each food type in each area
                                SELECT S.Restaurant, S.Area, S.Food_type, S.Price, S.Delivery_Time, S.
                                ↪Avg_Ratings,
                                    RANK() OVER (PARTITION BY S.Area, S.Food_Type ORDER BY S.Avg_Ratings
                                ↪DESC) AS Restaurant_Rank
                                FROM Swiggy S
                                JOIN Top_Food_Types_Per_Area T ON S.Area = T.Area AND S.Food_type = T.
                                ↪Food_type)
SELECT Restaurant, Area, Food_type, Price, Delivery_Time, Avg_Ratings
```

```

FROM Top_Restaurants_Per_Food_Type
WHERE Restaurant_Rank = 1
ORDER BY Area, Food_type, Avg_Ratings DESC"""
cur.execute(query)
data = cur.fetchall()
df = pd.DataFrame(data, columns = ["Restaurant", "Area", "Food_type", "Price", "Delivery_time", "Avg_ratings"])
df

```

```

[23]:

```

	Restaurant	Area \
0	Shiv Shakti Fast Food	Akhbar Nagar Circle
1	The Hide Away Cafe	Begumpet
2	Malabar Point	26
3	Hari Super Sandwich	3Rd Block Jayanagar
4	The Tasty Bites	A Unit Of M/S Cohort Ruby Area
...
2312	Gopal Chinese	Yoginagar Society
2313	Eat N Joy Bakers	Yousufguda
2314	4In Natural Fruit Juice	Yousufguda
2315	Crispy Hub	Zamistanpur
2316	Snehitha Restaurant	Zamistanpur

	Food_type	Price	Delivery_time \
0	Gujarati, Fast Food	200	53
1	Chinese	480	67
2	Biryani, Kerala, South Indian, Thal	450	58
3	Fast Food, Chaat, Snacks, Pizzas, North Indian, Indian	100	42
4	Asian, Indian, Tandoor, Tibetan, Chinese	300	67
...
2312	Chinese	200	55
2313	Bakery	200	67
2314	Juices Healthy Food	250	69
2315	American, Pizzas, Continental	250	41
2316	South Indian, Indian	160	37

	Avg_ratings
0	2.9
1	2.9
2	4.2
3	4.0
4	4.2
...	...
2312	3.9
2313	4.1
2314	4.0
2315	3.8
2316	4.4

[2317 rows x 6 columns]

10 Question 10: In-Depth Analysis of Price and Delivery Time Trends for Each Food Type Across Multiple Cities

```
[22]: query = """WITH City_Food_Type_Stats AS (  
                                -- Compute average price, delivery_  
    ↪time, and average ratings for each food type in each city  
    SELECT City, Food_type,  
           AVG(Price) AS Avg_Price,  
           AVG(Delivery_Time) AS Avg_Delivery_Time,  
           AVG(Avg_Ratings) AS Avg_Rating,  
           COUNT(Restaurant) AS Restaurant_Count  
    FROM swiggy  
    GROUP BY City, Food_type),  
Price_Delivery_Correlation AS (  
                                -- Calculate the correlation between_  
    ↪price and delivery time for each food type in each city  
    SELECT City, Food_type,  
           SUM(Price * Delivery_Time) AS Sum_XY,  
           SUM(Price) AS Sum_X,  
           SUM(Delivery_Time) AS Sum_Y,  
           COUNT(*) AS n,  
           SUM(Price * Price) AS Sum_X2,  
           SUM(Delivery_Time * Delivery_Time) AS Sum_Y2  
    FROM swiggy  
    GROUP BY City, Food_type),  
Correlation_Calculation AS (  
    SELECT P.City, P.Food_Type,  
           (Sum_XY - (Sum_X * Sum_Y / n)) /  
           SQRT((Sum_X2 - (Sum_X * Sum_X / n)) * (Sum_Y2 - (Sum_Y * Sum_Y /_  
    ↪n))) AS Price_Delivery_Correlation  
    FROM Price_Delivery_Correlation P)  
SELECT C.City, C.Food_type, C.Avg_Price, C.Avg_Delivery_Time, C.Avg_Rating, C.  
    ↪Restaurant_Count,  
       Coalesce(R.Price_Delivery_Correlation, 0) AS Price_Delivery_Correlation  
FROM City_Food_Type_Stats C  
LEFT JOIN Correlation_Calculation R ON C.City = R.City AND C.Food_type = R.  
    ↪Food_type  
ORDER BY C.City, C.Food_type;"""  
cur.execute(query)  
data = cur.fetchall()  
df = pd.DataFrame(data, columns = ["City", "Food_type", "Avg_Price",_  
    ↪"Avg_Delivery_time", "Avg_Rating", "Restaurant_Count", "Price_Delivery_Correlation"])
```

df

```
[22]:
```

	City	Food_type	Avg_Price \
0	Ahmedabad	American	130.0000
1	Ahmedabad	American,Continental,Fast Food,Combo	500.0000
2	Ahmedabad	American,Continental,Fast Food,Desserts,Beverages	500.0000
3	Ahmedabad	American,Desserts,Beverages	600.0000
4	Ahmedabad	American,Fast Food	375.0000
...
5139	Surat	Thalis,North Indian,Chaat,Fast Food,Street Food	250.0000
5140	Surat	Thalis,Punjabi,Gujarati,Combo,Chinese,North In...	130.0000
5141	Surat	Thalis,Punjabi,North Indian,Biryani	150.0000
5142	Surat	Thalis,Punjabi,North Indian,Chinese,Snacks,Tan...	250.0000
5143	Surat	Thalis,Snacks,Indian,Combo	150.0000

	Avg_Delivery_time	Avg_Rating	Restaurant_Count \
0	35.0000	2.90	1
1	44.0000	3.90	1
2	30.0000	4.40	1
3	35.0000	4.50	1
4	31.0000	4.25	2
...
5139	32.0000	4.30	1
5140	34.0000	3.50	1
5141	42.0000	3.50	1
5142	59.0000	3.90	1
5143	82.0000	3.60	1

	Price_Delivery_Correlation
0	0.0
1	0.0
2	0.0
3	0.0
4	-1.0
...	...
5139	0.0
5140	0.0
5141	0.0
5142	0.0
5143	0.0

[5144 rows x 7 columns]

11 Question 11: Comprehensive Evaluation of Restaurant Performance Based on Price, Delivery Time, and Ratings Across Different Areas

```
[21]: query = """WITH Area_Restaurant_Stats AS (
                                -- Compute average
                                price, delivery time, and ratings for each restaurant in each area
                                SELECT Area, Restaurant,
                                       AVG(Price) AS Avg_Price,
                                       AVG(Delivery_time) AS Avg_Delivery_Time,
                                       AVG(Avg_ratings) AS Avg_Rating,
                                       SUM(Total_ratings) AS Total_Ratings
                                FROM swiggy
                                GROUP BY Area, Restaurant),
                                Restaurant_Popularity AS (
                                -- Calculate the
                                popularity of each restaurant based on total ratings
                                SELECT Area, Restaurant,
                                       SUM(Total_Ratings) AS Total_Ratings
                                FROM swiggy
                                GROUP BY Area, Restaurant),
                                Avg_Ratings_Comparison AS (
                                -- Compare average
                                ratings of restaurants across different areas
                                SELECT A.Area, A.Restaurant, A.Avg_Price, A.Avg_Delivery_Time, A.Avg_Rating,
                                       R.Total_Ratings AS Restaurant_Popularity
                                FROM Area_Restaurant_Stats A
                                JOIN Restaurant_Popularity R ON A.Area = R.Area AND A.Restaurant = R.
                                Restaurant)
                                SELECT Area, Restaurant, Avg_Price, Avg_Delivery_Time, Avg_Rating,
                                Restaurant_Popularity
                                FROM Avg_Ratings_Comparison
                                ORDER BY Area, Restaurant_Popularity DESC;
                                """
cur.execute(query)
data = cur.fetchall()
df = pd.DataFrame(data, columns = ["Area", "Restaurant", "Avg_Price",
                                "Avg_Delivery_Time", "Avg_Rating", "Restaurant_Popularity"])
df
```

```
[21]:
```

	Area	Restaurant	Avg_Price \
0	Akhbar Nagar Circle	Shiv Shakti Fast Food	200.0000
1	Begumpet	The Hide Away Cafe	480.0000
2	26	Malabar Point	450.0000
3	3Rd Block Jayanagar	Hari Super Sandwich	100.0000
4	A Unit Of M/S Cohort Ruby Area	The Tasty Bites	300.0000

...
8668	Yoginagar Society	Gopal Chinese	200.0000
8669	Yousufguda	4In Natural Fruit Juice	250.0000
8670	Yousufguda	Eat N Joy Bakers	200.0000
8671	Zamistanpur	Crispy Hub	250.0000
8672	Zamistanpur	Snehitha Restaurant	160.0000

	Avg_Delivery_Time	Avg_Rating	Restaurant_Popularity
0	53.0000	2.9	80
1	67.0000	2.9	80
2	58.0000	4.2	100
3	42.0000	4.0	1000
4	67.0000	4.2	500
...
8668	55.0000	3.9	100
8669	69.0000	4.0	50
8670	67.0000	4.1	20
8671	41.0000	3.8	100
8672	37.0000	4.4	100

[8673 rows x 6 columns]