zomato

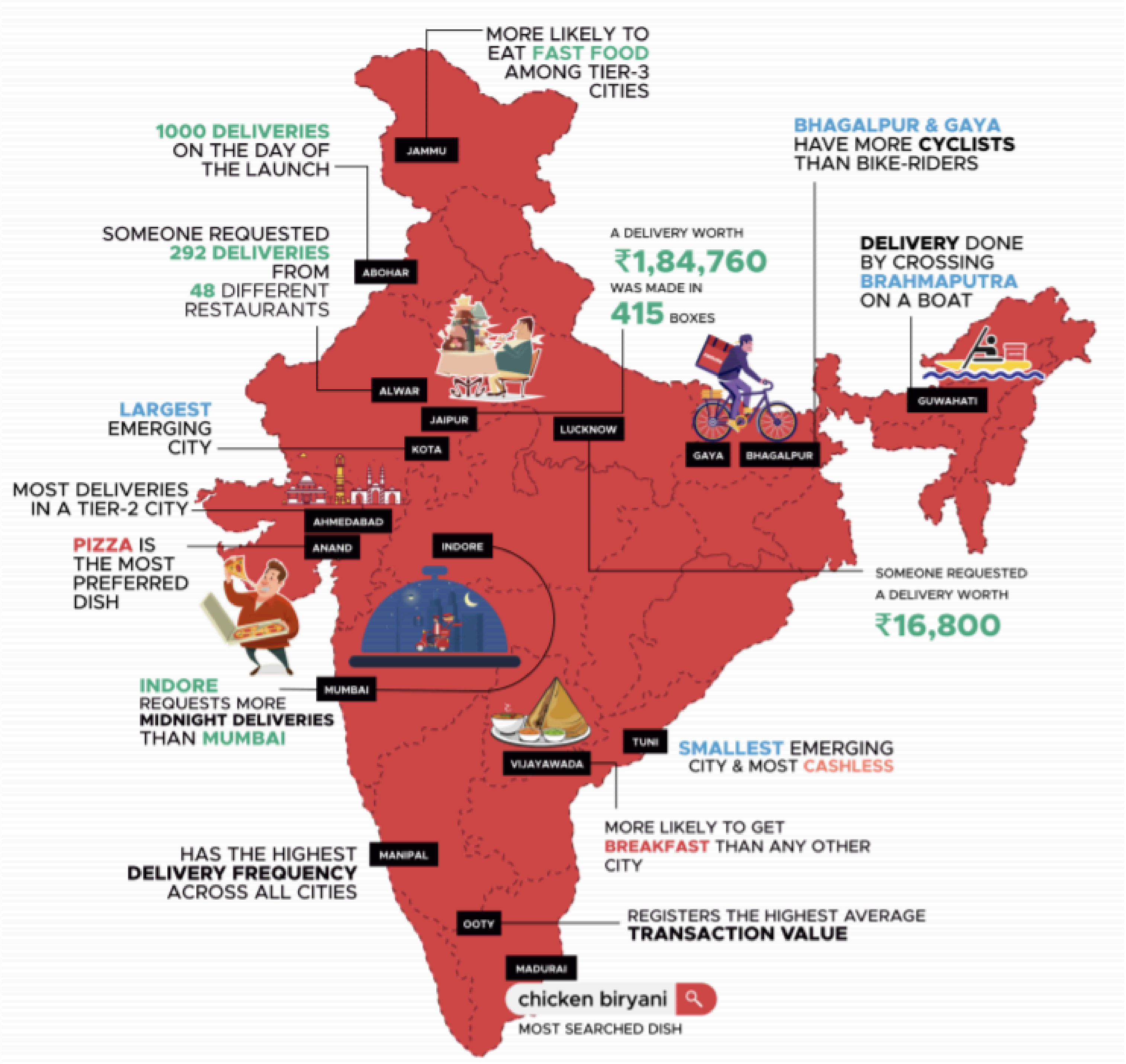
EXPLORATORY DATA ANALYSIS

DESIGNED BY: MAHAD AHMED

"INSIGHTS INTO DINING TRENDS AND CONSUMER PREFERENCES"

INTRODUCTION

The objective is to uncover patterns and trends in dining preferences across different Indian cities to inform targeted marketing strategies.



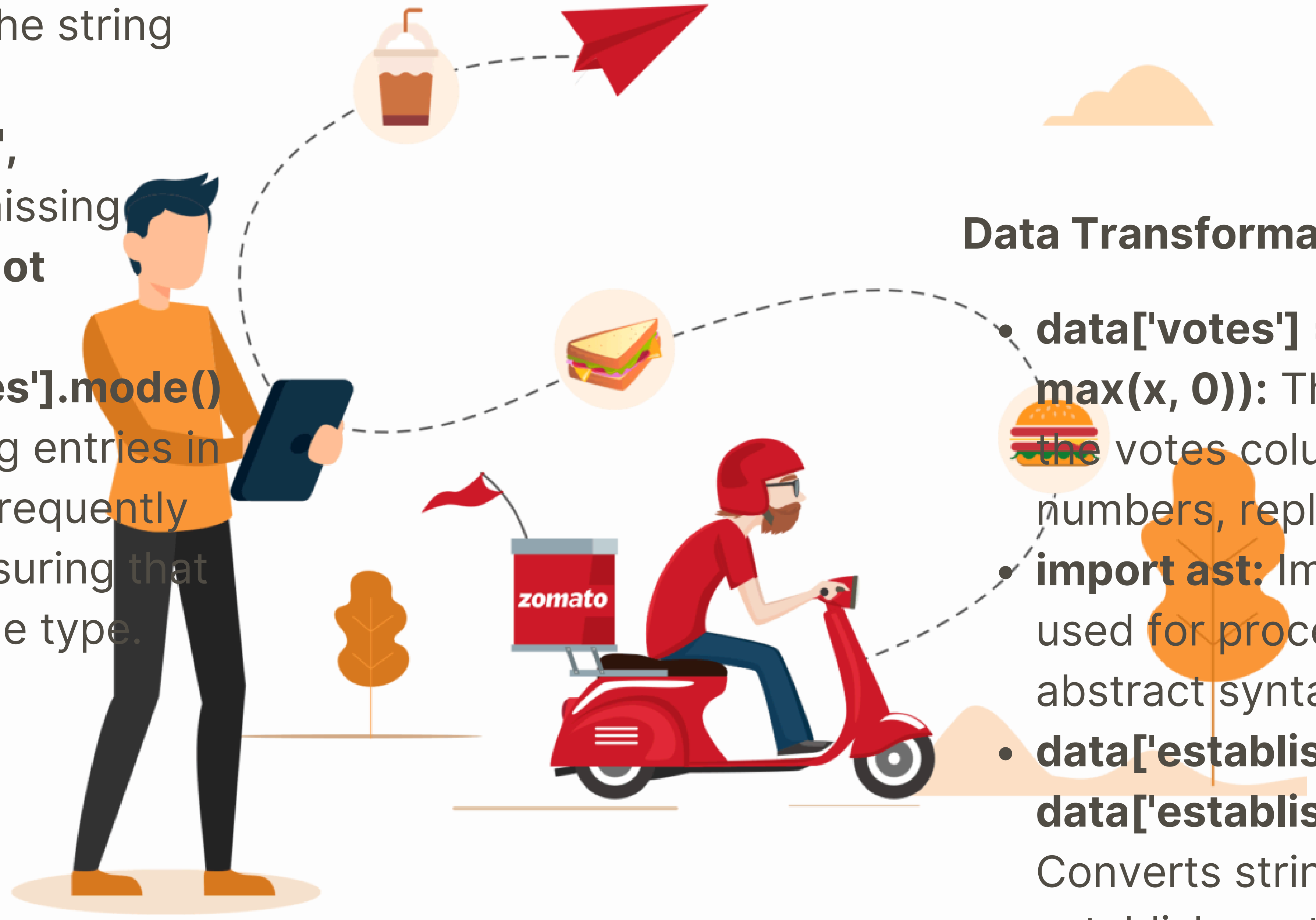
DATASET COMPOSITION

- The dataset comprises 211,944 records, each representing a restaurant.
- It includes restaurants from various cities across India, making it geographically diverse.
- Unique identifier for each restaurant.
- Classifies the type of dining establishment, such as Quick Bites, Casual Dining, Cafe, etc.
- Includes **address**, **city**, **locality**, which provide detailed location information.
- **latitude** and **longitude** fields for mapping and spatial analysis.
- Types of cuisines offered by the restaurant, e.g., Indian, Italian.
- Typical hours of operation.

DATA CLEANING & PREPRATION(|)

Fill Missing Values:

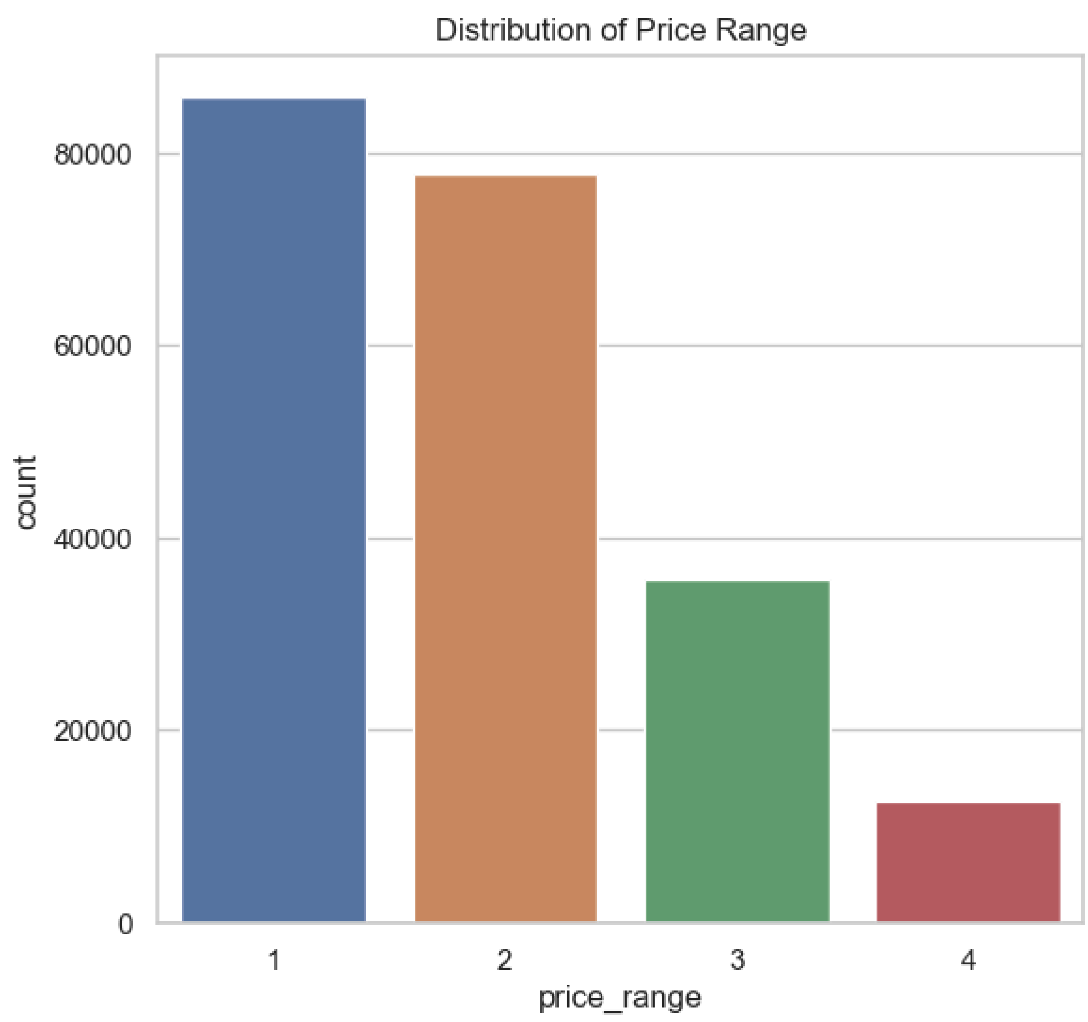
- **data['address'].fillna('Unknown', inplace=True):** This line replaces any missing values in the address column with the string **'Unknown'**.
- **data['timings'].fillna('Not available', inplace=True):** This line replaces missing values in the timings column with **'Not available'**.
- **data['cuisines'].fillna(data['cuisines'].mode()[0], inplace=True):** This fills missing entries in the cuisines column with the most frequently occurring cuisine in the dataset, ensuring that no restaurant is left without a cuisine type.

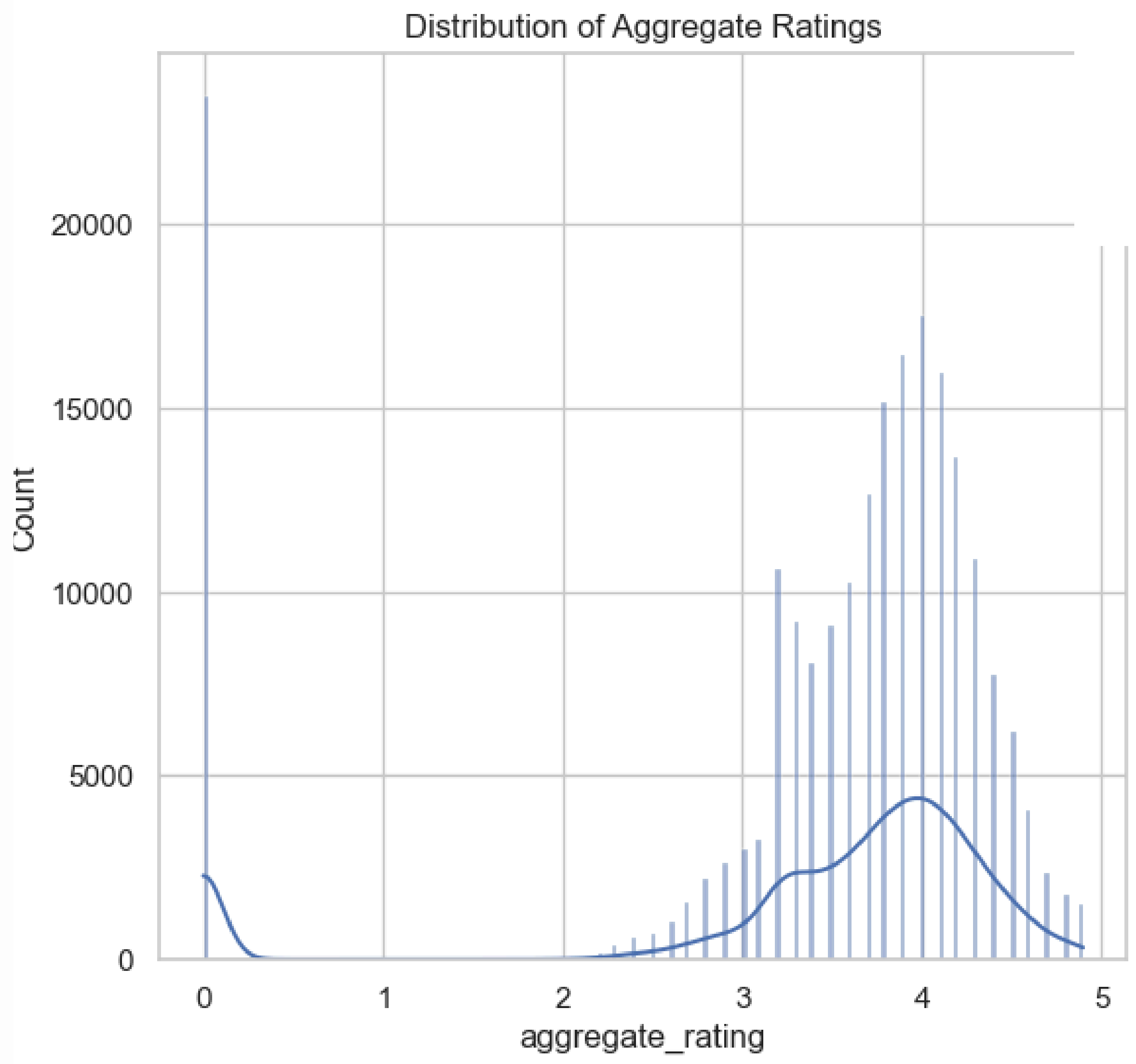


Data Transformation:

- **data['votes'] = data['votes'].apply(lambda x: max(x, 0)):** This lambda function ensures that the votes column doesn't contain any negative numbers, replacing them with zero if present.
- **import ast:** Imports the ast module, which is used for processing trees of the Python abstract syntax grammar.
- **data['establishment'] = data['establishment'].apply(ast.literal_eval):** Converts string representations of lists in the establishment column back into actual Python list objects using ast.literal_eval.

DATA CLEANING & PREPRATION(||)





Feature Engineering:

- **data['number_of_highlights'] = data['highlights'].apply(len):** Creates a new column number_of_highlights by calculating the length of each list in the highlights column, effectively counting the number of highlights per restaurant.

Visualizations:

- **fig, axes = plt.subplots(1, 2, figsize=(14, 6)):** Creates a figure with two subplots side-by-side.
- **sns.histplot(data['aggregate_rating'], kde=True, ax=axes[0])**: Plots a histogram of **aggregate_rating** with a kernel density estimate to show the distribution shape.
- **axes[0].set_title('Distribution of Aggregate Ratings')**: Sets the title for the first subplot.
- **sns.countplot(x='price_range', data=data, ax=axes[1])**: Creates a count plot for the **price_range** column to show the frequency of each price range category.
- **axes[1].set_title('Distribution of Price Range')**: Sets the title for the second subplot.
- **plt.show()**: Displays the plots.

EXPLORATORY DATA ANALYSIS (|)

Statistical Analysis:

1. **Central Tendency:**
   - central_tendency = data[['aggregate_rating', 'average_cost_for_two', 'number_of_highlights']].agg(['mean', 'median'])
2. **Dispersion:**
   - dispersion = data[['aggregate_rating', 'average_cost_for_two', 'number_of_highlights']].agg(['std', 'min', 'max'])
3. **Shape:**
   - shape = data[['aggregate_rating', 'average_cost_for_two', 'number_of_highlights']].agg(['skew', 'kurt'])

Correlation:

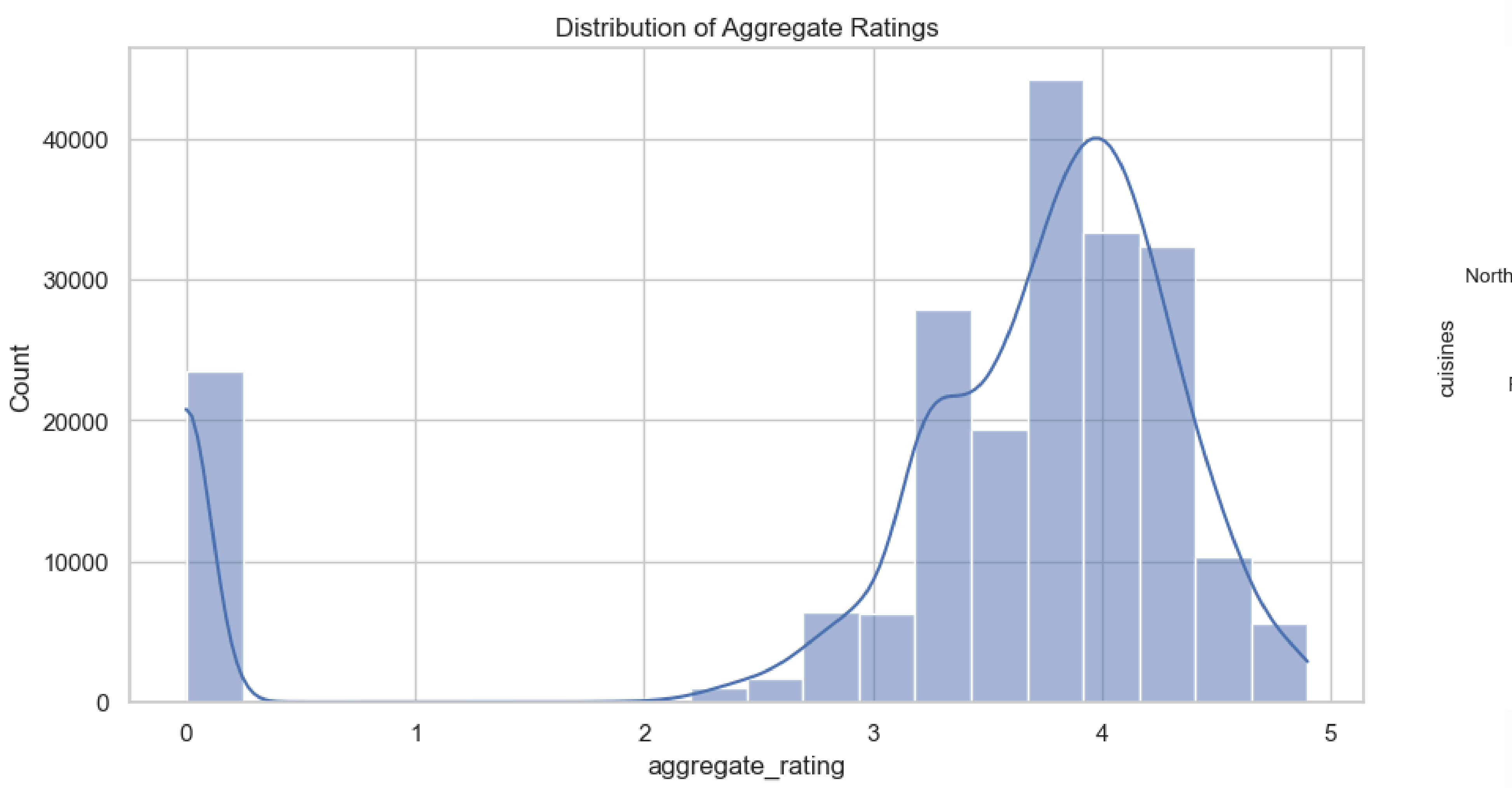
- correlation_matrix = data[['aggregate_rating', 'average_cost_for_two', 'votes', 'number_of_highlights']].corr()

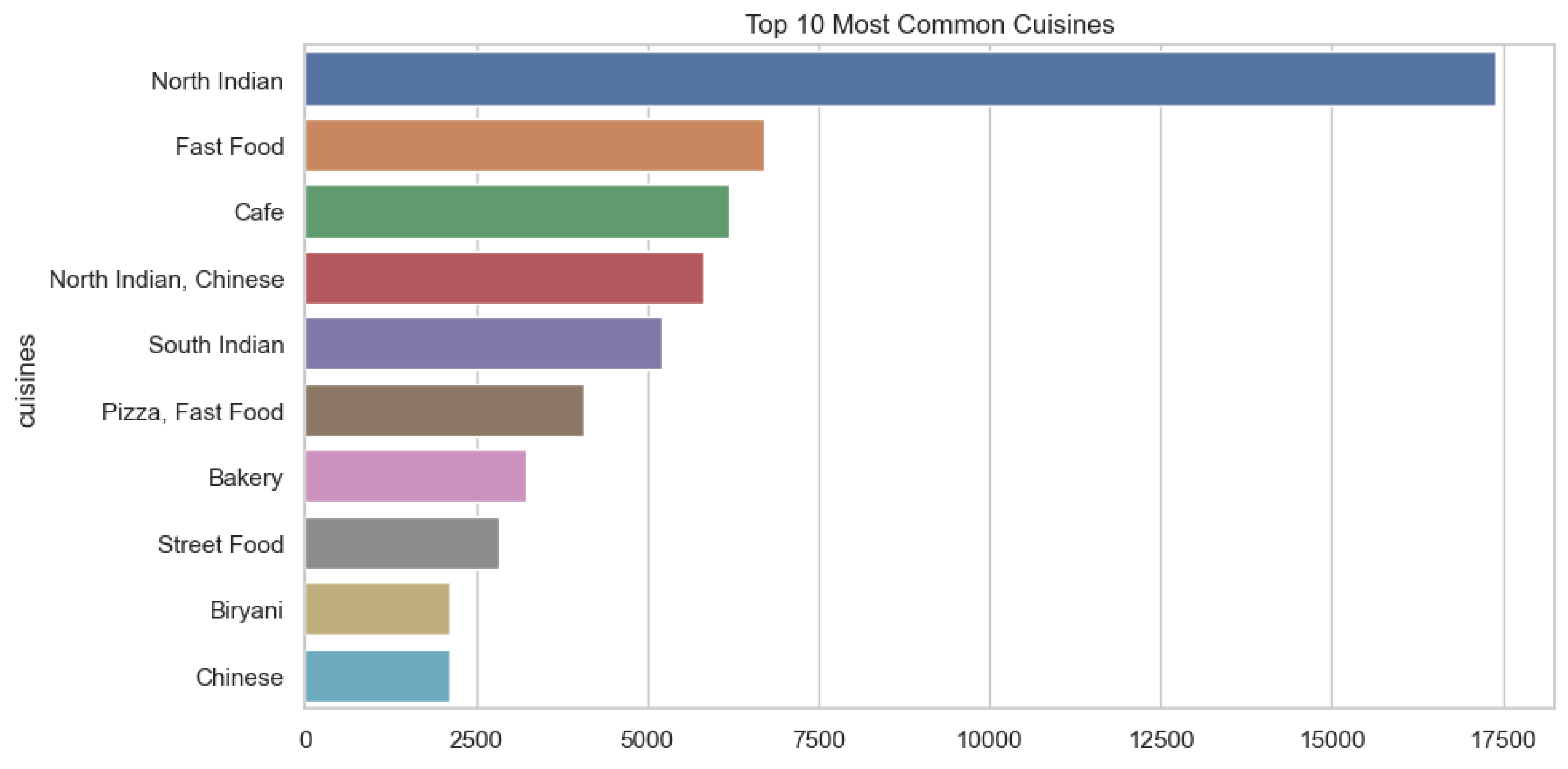
Heatmap of Correlation Matrix:

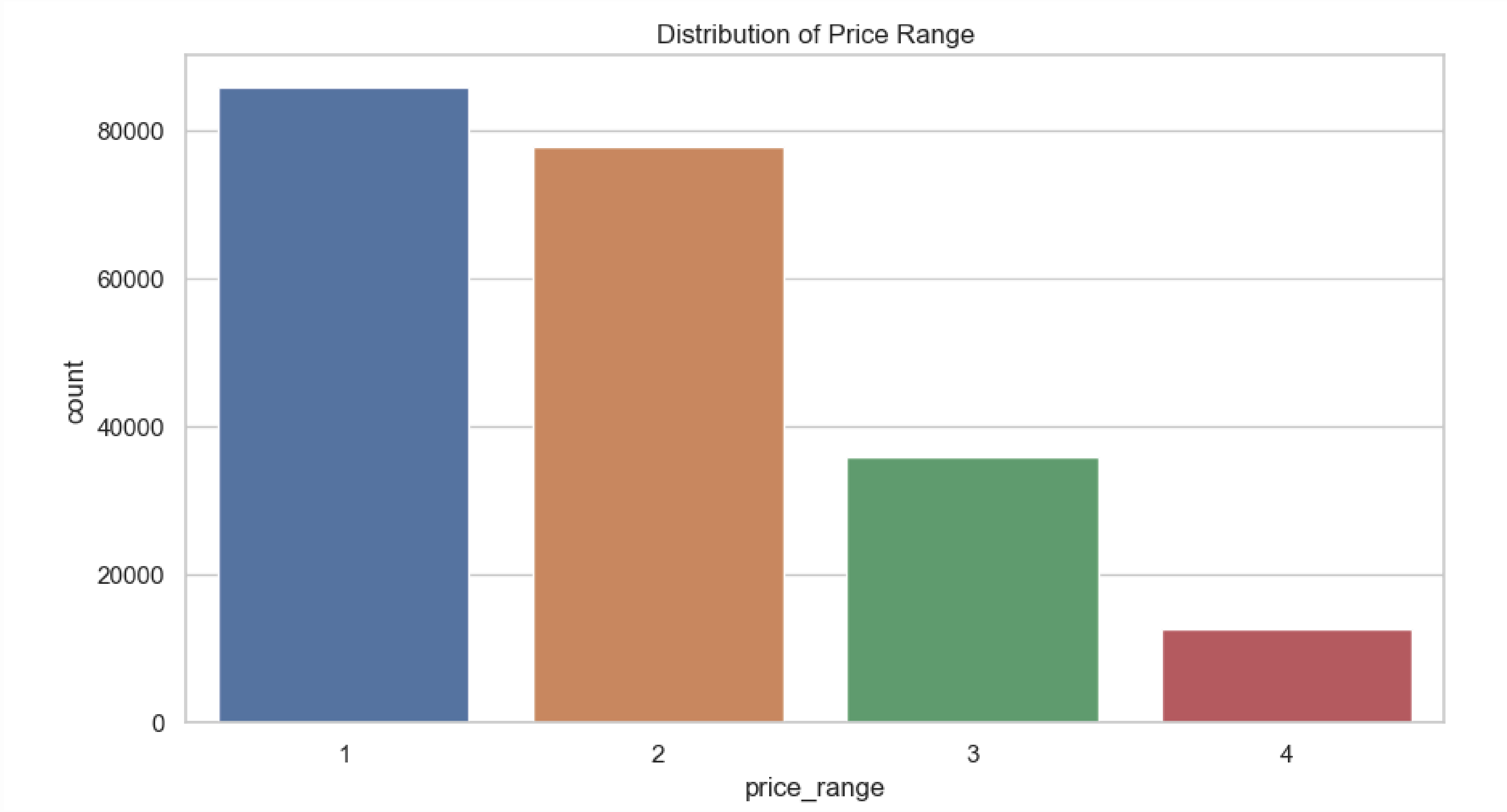
- plt.figure(figsize=(8, 6))
- sns.heatmap(correlation_matrix, annot=True, fmt=".2f", cmap='coolwarm')
- plt.title('Correlation Matrix')

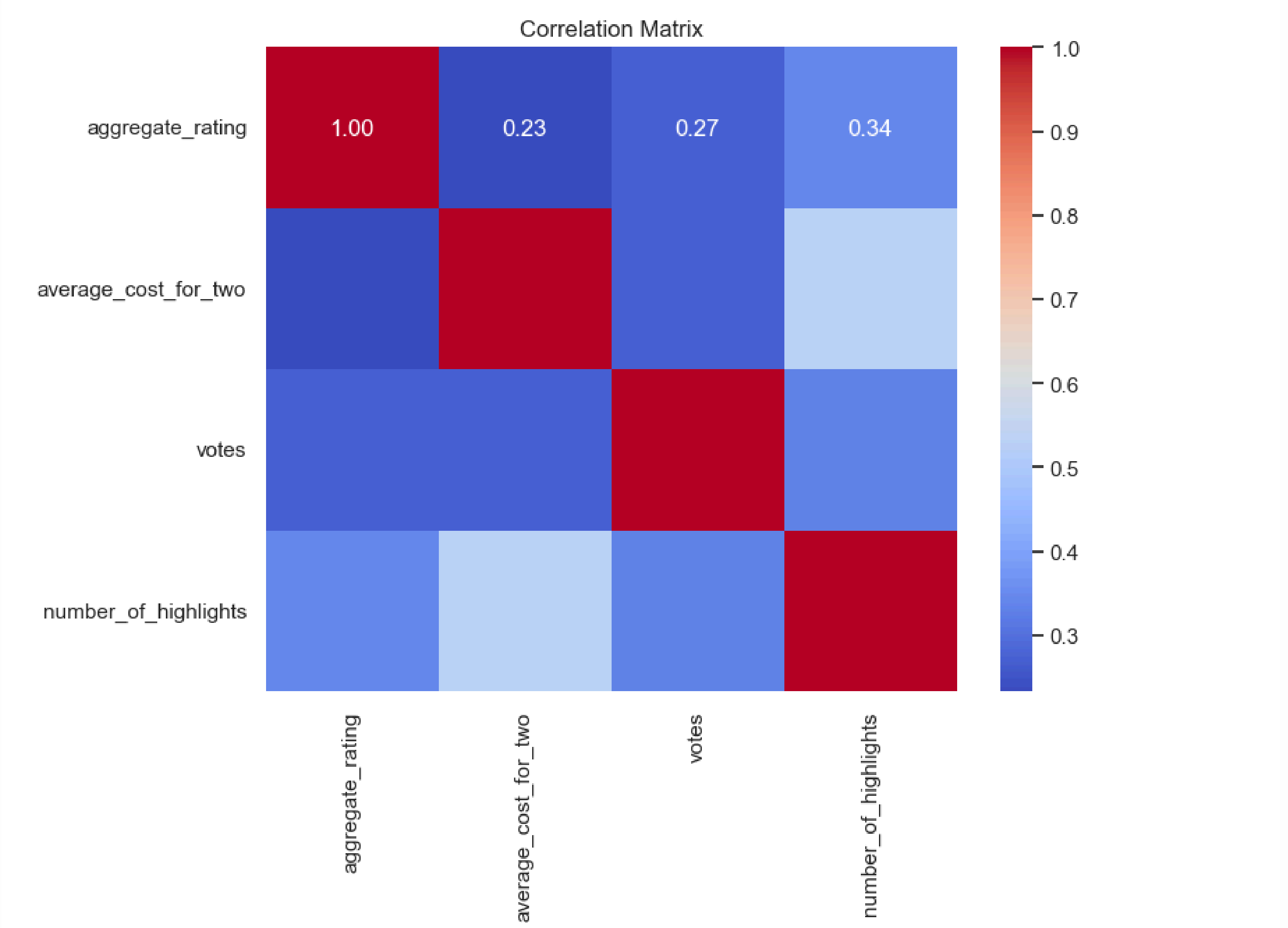
Visualization:

1. Histogram of Aggregate Ratings:
   - sns.histplot(data['aggregate_rating'], bins=20, kde=True, ax=axes[0])
   - axes[0].set_title('Distribution of Aggregate Ratings')
2. Count Plot of Price Range:
   - sns.countplot(x='price_range', data=data, ax=axes[1])
   - axes[1].set_title('Distribution of Price Range')
3. Bar Plot of Top Cuisines:
   - top_cuisines = data['cuisines'].value_counts().head(10)
   - sns.barplot(x=top_cuisines.values, y=top_cuisines.index, ax=axes[2])
   - axes[2].set_title('Top 10 Most Common Cuisines')
   - plt.tight_layout()


Distribution of Aggregate Ratings
Count
40000
30000
20000
10000
0
0
1
2
3
4
5
aggregate_rating


Top 10 Most Common Cuisines
cuisines
North Indian
Fast Food
Cafe
North Indian, Chinese
South Indian
Pizza, Fast Food
Bakery
Street Food
Biryani
Chinese
0
2500
5000
7500
10000
12500
15000
17500


Distribution of Price Range
count
80000
60000
40000
20000
0
1
2
3
4
price_range


Correlation Matrix
aggregate_rating
average_cost_for_two
votes
number_of_highlights
1.00
0.23
0.27
0.34
aggregate_rating
average_cost_for_two
votes
number_of_highlights
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3

EXPLORATORY DATA ANALYSIS (||)

Data Grouping and Aggregation:

1. **Group and Aggregate Data by City:**
   - city_group = data.groupby('city').agg(...)
     - 'res_id': 'count'
     - 'average_cost_for_two': 'mean':
     - 'aggregate_rating': 'mean':
     - 'number_of_highlights': 'mean':

   - .rename(columns={'res_id': 'number_of_restaurants'}):
   - .sort_values(by='number_of_restaurants', ascending=False):
1. **Extract Top 10 Cities:**
   - top_cities = city_group.head(10):

Unique Characteristics per City:

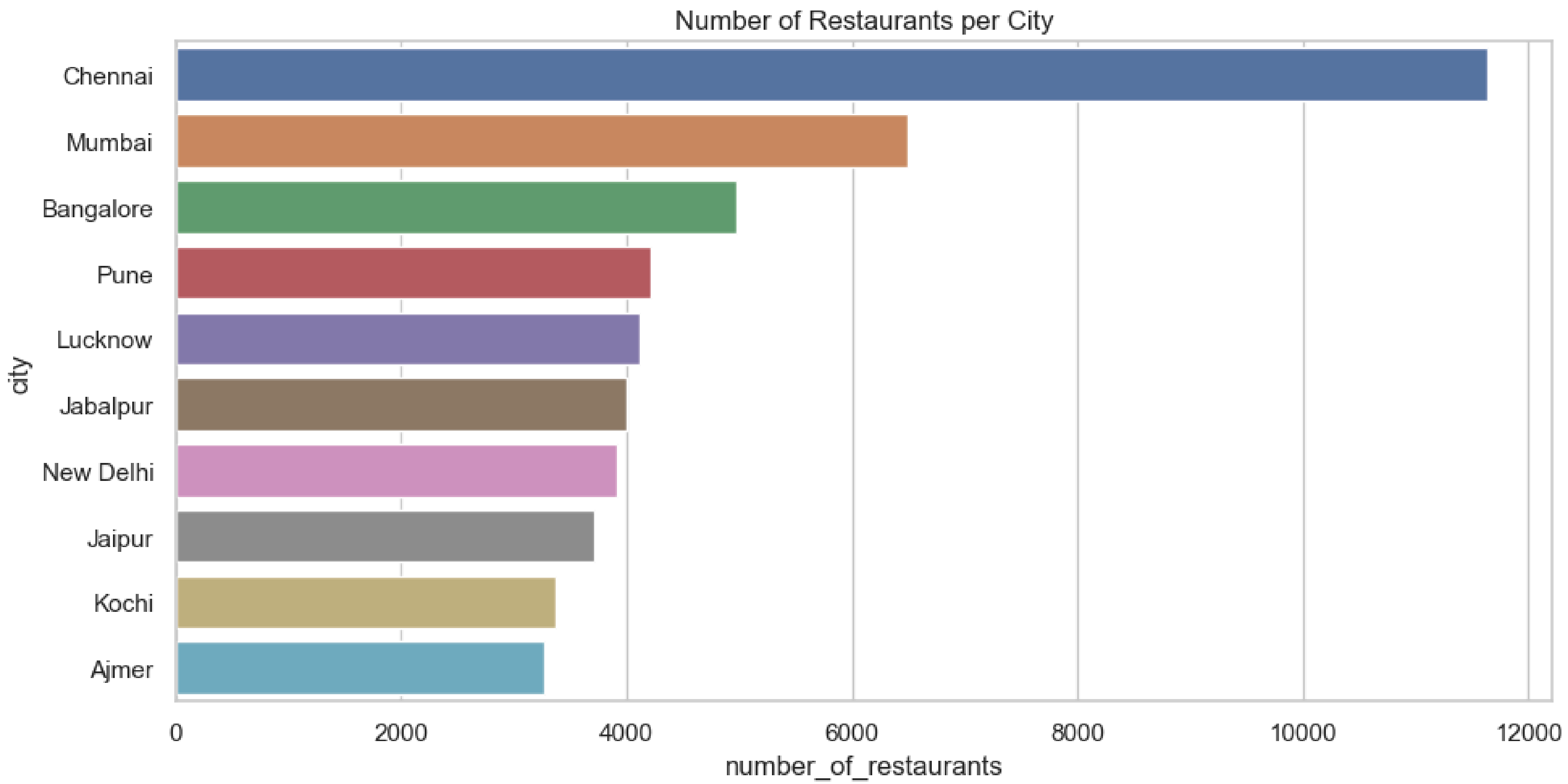
1. **Calculate Mode of Cuisines and Highlights:**
- city_characteristics = data.groupby('city').agg(...):
2. **Display Characteristics:**
- city_characteristics.head(10):

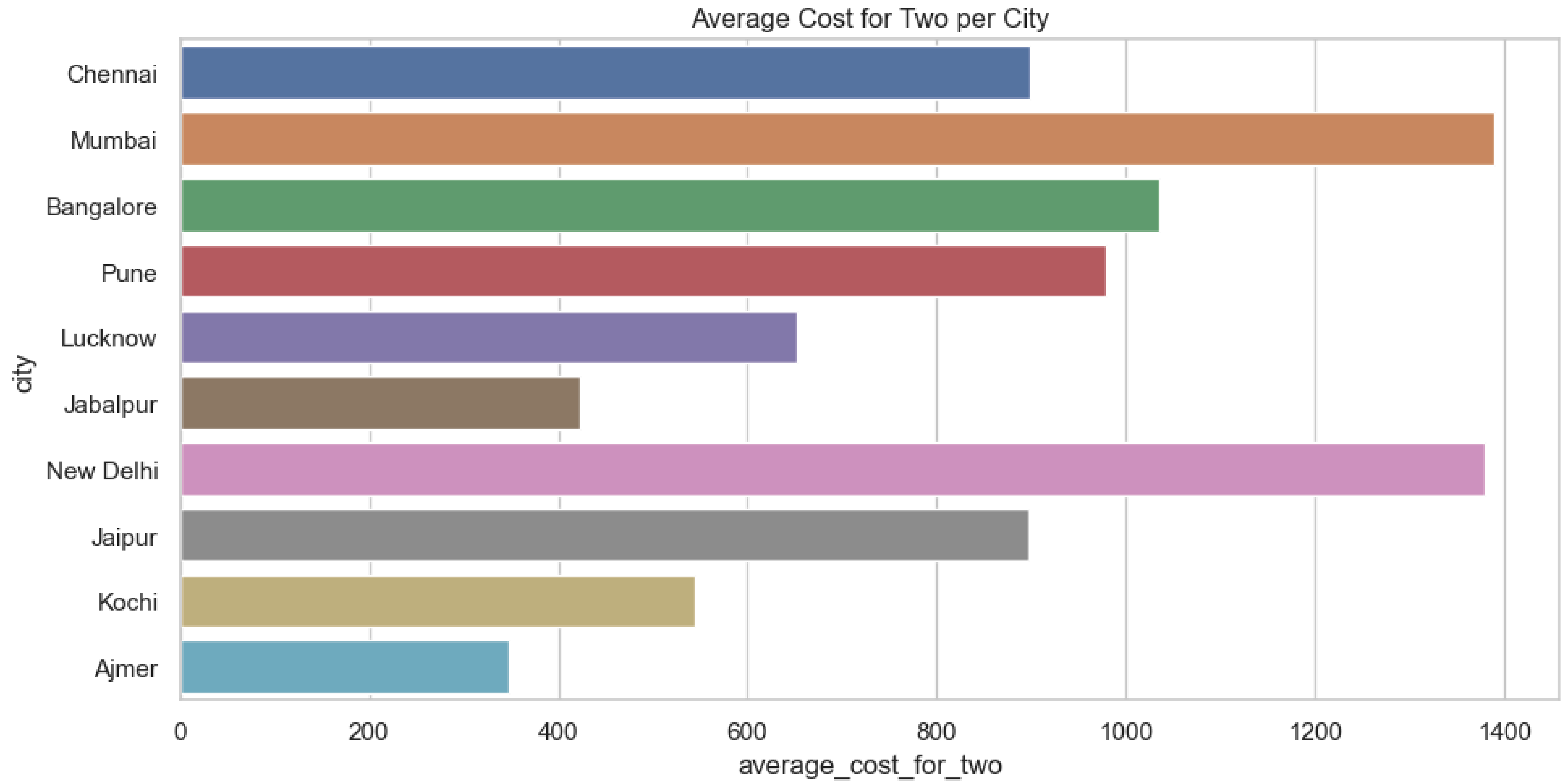
Visualization of City Data

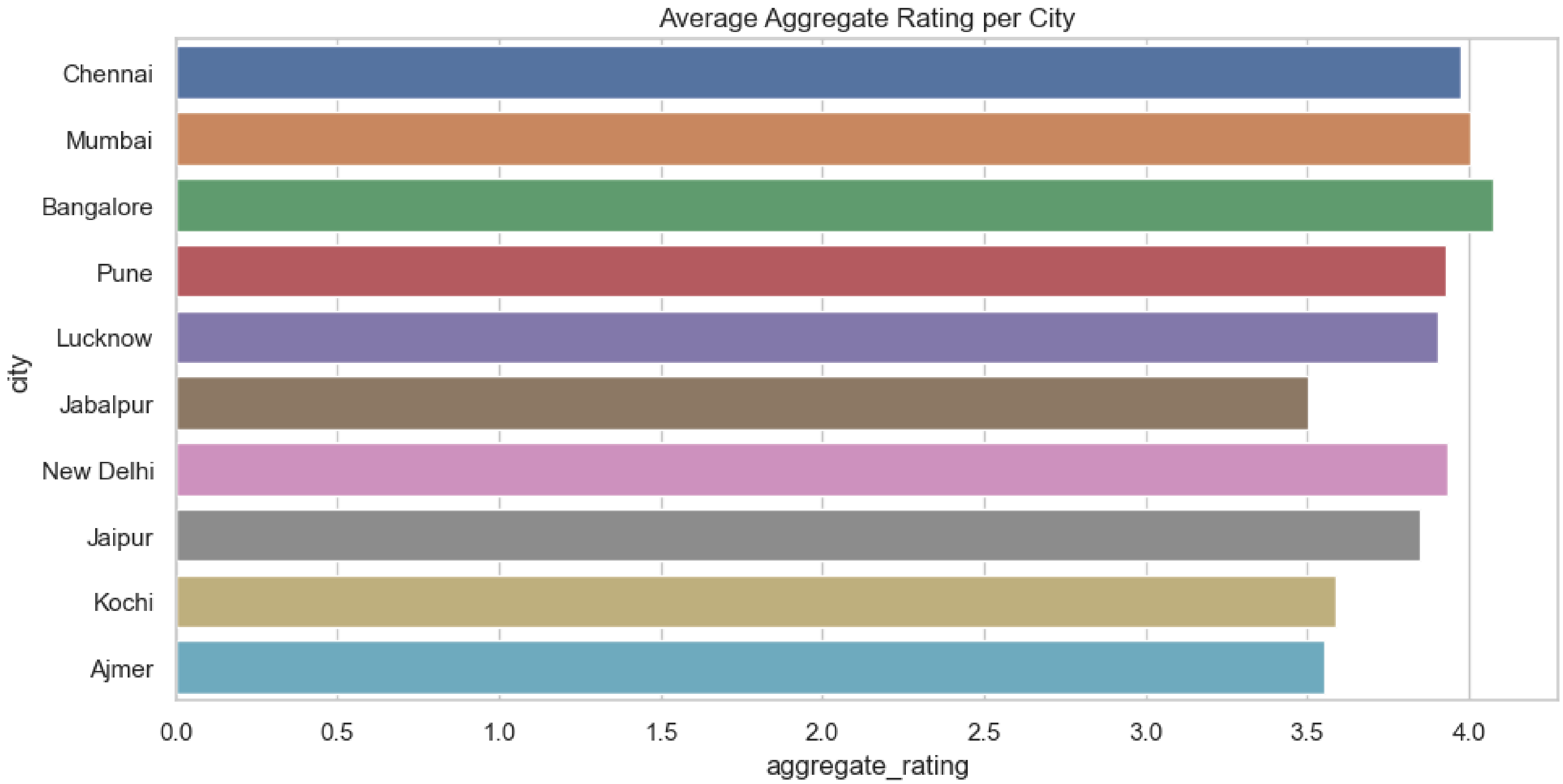
Setup Plot:
- fig, ax = plt.subplots(3, 1, figsize=(10, 15)):

Bar Plots for Each Metric:
1. **Number of Restaurants:**
- sns.barplot(x=top_cities['number_of_restaurants'], y=top_cities.index, ax=ax[0]):
- ax[0].set_title('Number of Restaurants per City'):
2. **Average Cost for Two:**
- sns.barplot(x=top_cities['average_cost_for_two'], y=top_cities.index, ax=ax[1]):
- ax[1].set_title('Average Cost for Two per City'):
3. **Average Aggregate Rating:**
- sns.barplot(x=top_cities['aggregate_rating'], y=top_cities.index, ax=ax[2]):
- ax[2].set_title('Average Aggregate Rating per City'):

Layout Adjustment:
- plt.tight_layout():


Number of Restaurants per City
city
Chennai
Mumbai
Bangalore
Pune
Lucknow
Jabalpur
New Delhi
Jaipur
Kochi
Ajmer
0
2000
4000
6000
8000
10000
12000
number_of_restaurants


Average Cost for Two per City
city
Chennai
Mumbai
Bangalore
Pune
Lucknow
Jabalpur
New Delhi
Jaipur
Kochi
Ajmer
0
200
400
600
800
1000
1200
1400
average_cost_for_two


Average Aggregate Rating per City
city
Chennai
Mumbai
Bangalore
Pune
Lucknow
Jabalpur
New Delhi
Jaipur
Kochi
Ajmer
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
aggregate_rating

EXPLORATORY DATA ANALYSIS (|||)

1. Analyzing Popular Cuisines in Different Regions:

- **Grouping and Counting Cuisines:**
  - city_cuisine_counts = data.groupby(['city', 'cuisines']).size().reset_index(name='count'):
- **Identifying Most Popular Cuisine per City:**
  - most_popular_cuisine_per_city = city_cuisine_counts.groupby('city').apply(lambda x: x.loc[x['count'].idxmax()]):

2. Visualizing Relationships Between Ratings, Price, and Popularity:

1. **Scatter Plot: Cost vs. Rating with Price Range Hue:**
- plt.figure(figsize=(10, 6)):
- sns.scatterplot(...):
- plt.title(...), plt.xlabel(...), plt.ylabel(...), plt.legend(...):

2. **Scatter Plot: Votes vs. Rating:**
- A similar setup for another scatter plot, this time plotting the number of votes against the aggregate rating. This plot aims to visualize whether more popular (higher voted) restaurants tend to have higher ratings, with the assumption that votes indicate popularity.

3. Correlation Analysis:

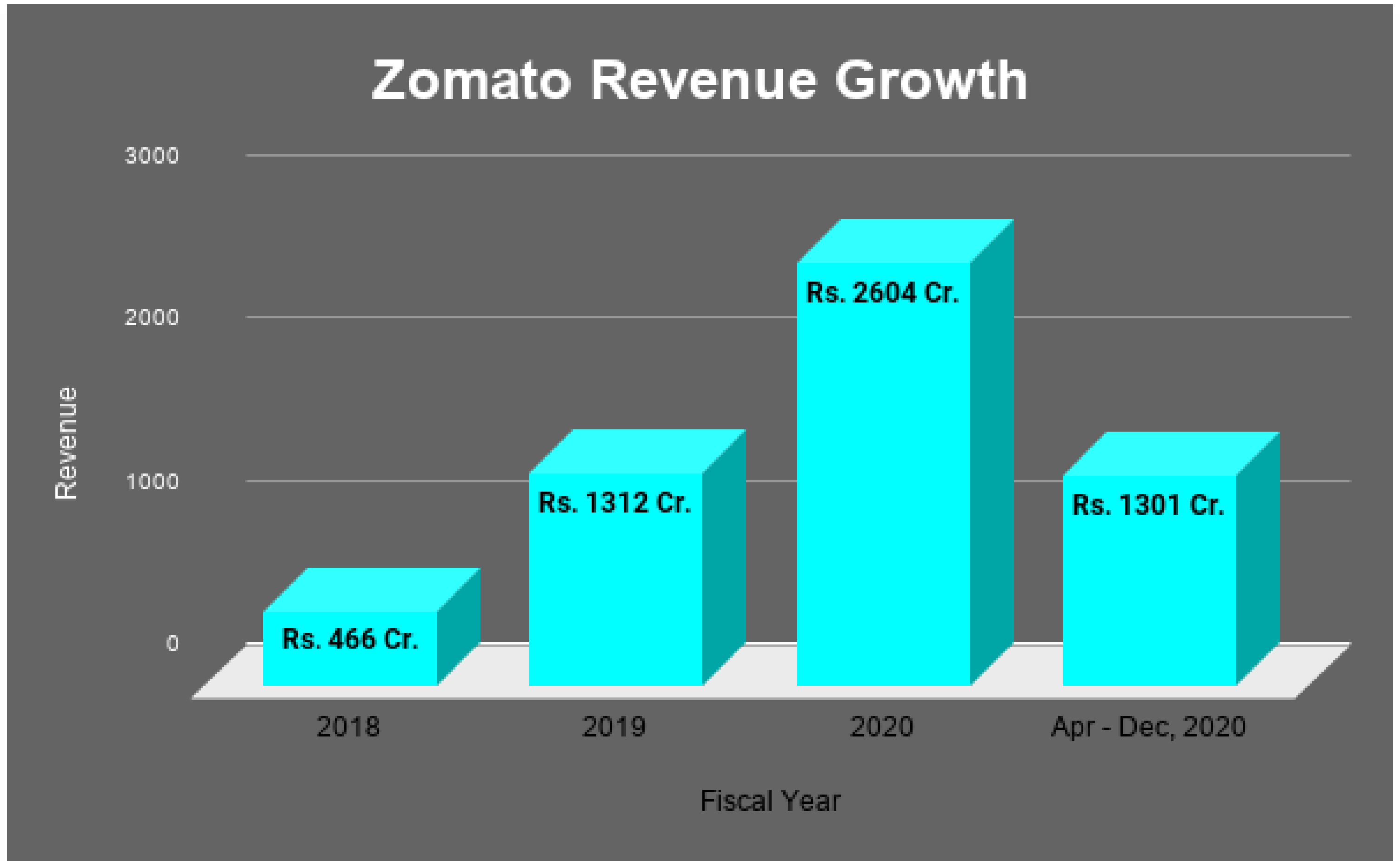
- correlation_matrix = data[['aggregate_rating', 'average_cost_for_two', 'votes']].corr():
- print(correlation_matrix):

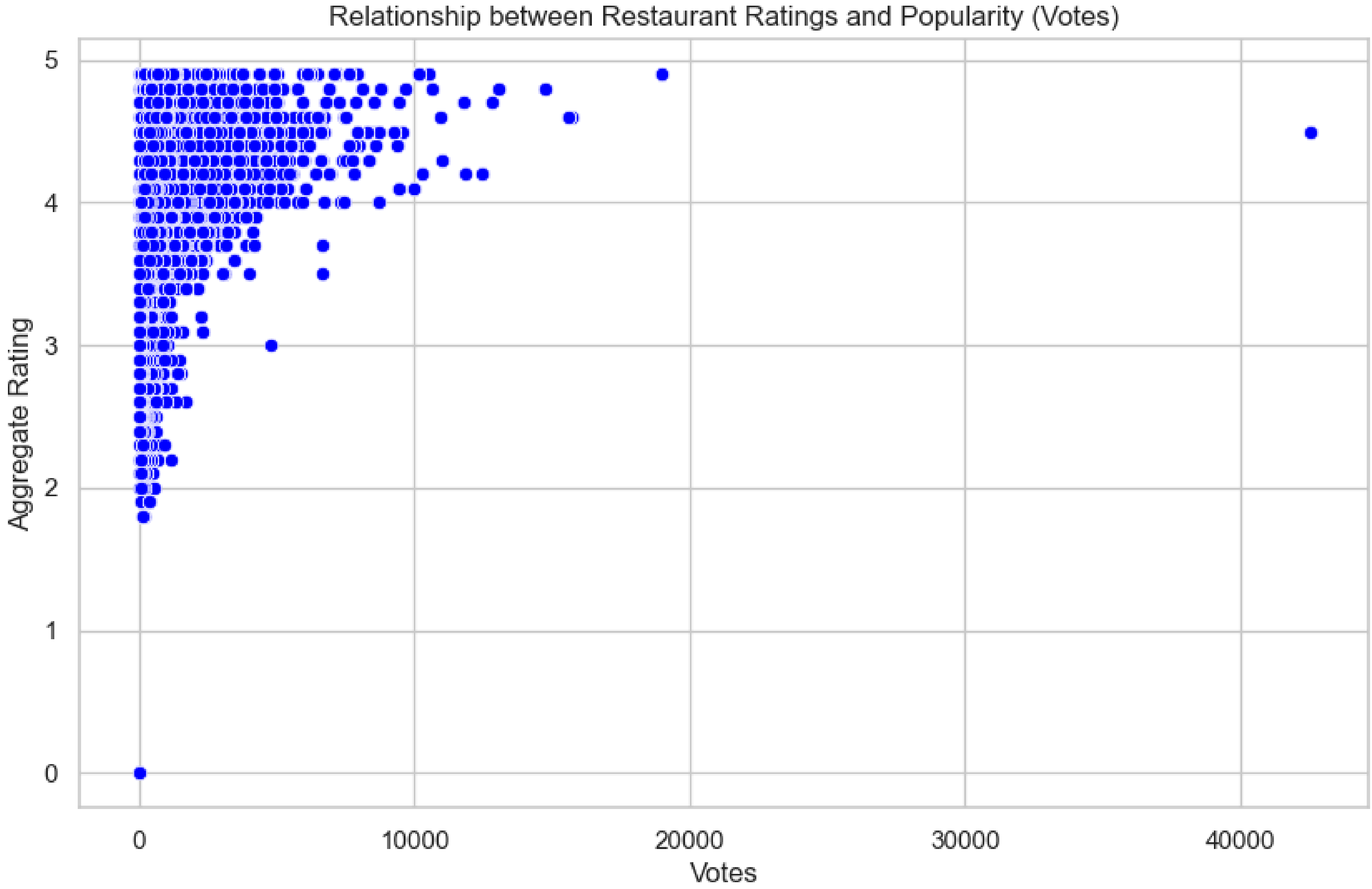
4. Displaying Most Popular Cuisine per City:

1. **Print Most Popular Cuisines:**
- print(most_popular_cuisine_per_city.head()):


Relationship between Restaurant Ratings, Price Range, and Popularity
Price Range
1
2
3
4
Aggregate Rating
0
1
2
3
4
5
0
5000
10000
15000
20000
25000
30000
Average Cost for Two


zomato


Zomato Revenue Growth
3000
2000
1000
0
Revenue
Rs. 466 Cr.
Rs. 1312 Cr.
Rs. 2604 Cr.
Rs. 1301 Cr.
2018
2019
2020
Apr - Dec, 2020
Fiscal Year


Relationship between Restaurant Ratings and Popularity (Votes)
Aggregate Rating
0
1
2
3
4
5
0
10000
20000
30000
40000
Votes

IMPLICATIONS FOR MARKETING STRATEGY

1. Targeting Strategy:

- **Regional Focus:**
  - → **North India (Delhi, Chandigarh):** Emphasize local cuisines like North Indian and street food.
  - → **South India (Chennai, Bangalore):** Focus on traditional South Indian eateries and modern cafes popular among young professionals.
  - → **West India (Mumbai, Pune):** Highlight seafood and fast-food options popular in these cosmopolitan areas.

- **Customer Segmentation:**
  - → **Youth and Students:** Promote budget-friendly quick bites and cafes with vibrant ambiances.
  - → **Working Professionals:** Focus on restaurants offering lunch specials, efficient service, and high ratings.
  - → **Families:** Highlight family-friendly restaurants with diverse menus and good safety ratings.

2. Differentiation Strategy:

- **Highlight USPs:** Promote unique dining experiences like rooftop dining, pet-friendly cafes, or restaurants offering organic and sustainable menu options.
- **Leverage Top Performers**: Focus on promoting restaurants with high ratings but lower visibility, marketing them as 'must-visit' spots.

IMPLICATIONS FOR MARKETING STRATEGY

3. Promotional Tactics:

- Discount & Offers:
  - → **Early Bird Specials:** Discounts for customers dining before peak hours
  - → **Weekday Discounts**: Special pricing on weekdays to boost footfall.
- Loyalty Program:
  - → **Zomato Gold Partnership:** Offer enhanced benefits for Zomato Gold members, like 1+1 on food or 2+2 on drinks.
  - → **Reward Points System:** Implement a points system where customers ca earn and redeem points based on their spending.
- Special Event:
  - → **Cuisine Days**: Host theme days like "Italian Cuisine Day" or "Punjabi Food Fest" to attract cuisine-specific aficionados.
  - → **Celebrity Chef Nights**: Organize special dinners curated by renowned chefs to attract high-end diners.

4. Technology & Engagement:

- App Feature:
  - → Push notifications for nearby restaurant promotions based on user location and past preferences.
  - → AR features to preview best-selling dishes when users point their camera at the menu.
- Social Media Campaigns:
  - → **Instagram Photo Contests:** Encourage users to post their meals with specific hashtags to win dining vouchers.
  - → **Facebook Events:** Create events for special dining nights or festivals, encouraging RSVPs and sharing.

CONCLUSION

Cuisine Preferences Vary by City:

- The analysis demonstrated significant variation in popular cuisines across different cities. This indicates that regional tastes and cultural differences strongly influence dining preferences. For instance, the most popular cuisine in each city reflects local culinary traditions or the demographic makeup of the area. Marketers and new restaurants can leverage this insight to tailor their menu offerings to align with local tastes.

Relationship Between Price and Ratings:

- The scatterplot analysis of restaurant ratings against average cost for two, colored by price range, suggests that higher-priced restaurants generally receive higher ratings. This could be attributed to the perception of quality associated with cost or the actual quality of food and service provided by higher-priced establishments. This relationship highlights a potential strategy for restaurants to position themselves in higher price brackets as a mark of quality, though this must be backed by genuine quality improvements to sustain customer satisfaction.

Popularity and Ratings:

- The positive correlation between the number of votes (popularity) and aggregate ratings indicates that more popular restaurants tend to have higher ratings. This relationship suggests that higher customer satisfaction can lead to increased popularity through word-of-mouth and repeat visits. Restaurants should focus on customer experience to boost their ratings and popularity, which in turn can attract more patrons.

Correlation Insights:

- The correlation analysis provided a clear understanding of the relationships among key numerical variables. Notably, while there is a positive correlation between average cost and ratings, the relationship between these factors and votes was less pronounced than expected. This indicates that while cost and quality (ratings) are significant, they are not the sole drivers of popularity

THANKYOU