Assignment

Flight Price:

Q1. Load the flight price dataset and examine its dimensions. How many rows and columns does the
dataset have?

Ans:the dimensions of the flight price dataset, you would typically use a programming language such as Python along with libraries like Pandas. Unfortunately, I don't have access to specific datasets or the ability to execute code that requires external data. However, I can provide you with a general code template that you can use in your Python environment.

Assuming you have a CSV file containing the flight price data, here's an example using Python and Pandas:

python

Copy code

```
import        as


            'your_dataset.csv'







print f"The dataset has {rows} rows and {columns} columns."
```

Make sure to replace 'your_dataset.csv' with the actual file path or URL of your flight price dataset. Running this code will load the dataset and print the number of rows and columns it contains.

If you encounter any issues or have specific details about the dataset (e.g., the format or structure), feel free to provide more information for further assistance.

Q2. What is the distribution of flight prices in the dataset? Create a histogram to visualize the distribution.

Ans:To analyze the distribution of flight prices in the dataset and create a histogram for visualization, you can use Python along with libraries such as Pandas and Matplotlib. Here's an example code template:

python

Copy code

```
import        as

import                   as




            'your_dataset.csv'










                            'Price'
```

```
                    10  6
                             30        'skyblue'            'black'
          'Distribution of Flight Prices'
           'Flight Price'
           'Frequency'
              'y'            '--'        0.7
```

Make sure to replace 'your_dataset.csv' with the actual file path or URL of your flight price dataset. This code will load the dataset, extract the 'Price' column, and create a histogram to visualize the distribution of flight prices.

Adjust the number of bins in the `plt.hist()` function to control the granularity of the histogram. Running this code will provide insights into the spread and frequency distribution of flight prices in your dataset.

Q3. What is the range of prices in the dataset? What is the minimum and maximum price?

Ans:To find the range of prices in the flight price dataset and determine the minimum and maximum prices, you can use Python along with Pandas. Here's an example code snippet:

python

Copy code

```
import        as
```

```
            'your_dataset.csv'





                            'Price'


                          min

                          max


print f"The minimum price is: {min_price}"

print f"The maximum price is: {max_price}"
```

Replace 'your_dataset.csv' with the actual file path or URL of your flight price dataset. This code will load the dataset, extract the 'Price' column, and calculate the minimum and maximum prices.

Running this code will provide you with the minimum and maximum prices, giving you the range of prices in your flight price dataset.

Q4. How does the price of flights vary by airline? Create a boxplot to compare the prices of different

airlines.

Ans:To analyze how the price of flights varies by airline and create a boxplot for comparison, you can use Python along with Pandas and Seaborn. Here's an example code template:

python

Copy code

```
import        as
import         as
import                   as


            'your_dataset.csv'





                    12  8
              'Airline'    'Price'                            'viridis'
          'Flight Prices by Airline'
           'Airline'
           'Price'
                    45     'right'
```

Replace 'your_dataset.csv' with the actual file path or URL of your flight price dataset. This code will load the dataset and create a boxplot to visually compare the prices of different airlines.

The boxplot will display the distribution of flight prices for each airline, showing the median, quartiles, and potential outliers. Adjust the size, color palette, and other parameters based on your preferences.

Running this code will provide insights into how flight prices vary among different airlines in your dataset.

Q5. Are there any outliers in the dataset? Identify any potential outliers using a boxplot and describe how
they may impact your analysis.
Ans:To identify potential outliers in the flight price dataset and visualize them using a boxplot, you can use Python with libraries like Pandas and Seaborn. Outliers can have an impact on statistical analyses, so it's important to identify and understand them. Here's an example code template:

python

Copy code

```
import        as
import         as
import                   as


            'your_dataset.csv'





                    10  6
              'Price'                          'skyblue'
          'Boxplot of Flight Prices'
           'Flight Price'
```

Replace 'your_dataset.csv' with the actual file path or URL of your flight price dataset. This code will load the dataset and create a boxplot to visualize the distribution of flight prices and identify potential outliers.

In the boxplot, outliers are typically represented as points beyond the whiskers of the box. Points beyond these whiskers are considered as potential outliers. The impact of outliers on your analysis depends on the context and the specific analysis you're conducting. Outliers can influence measures such as mean and standard deviation, and they may need special attention or treatment depending on the goals of your analysis.

After visualizing potential outliers, you may consider further investigation to determine if they are genuine data points or if there are data quality issues. Deciding whether to exclude, transform, or handle outliers in a specific way depends on the nature of the data and the goals of your analysis.

Q6. You are working for a travel agency, and your boss has asked you to analyze the Flight Price dataset
to identify the peak travel season. What features would you analyze to identify the peak season, and how
would you present your findings to your boss?
Ans:To identify the peak travel season in the Flight Price dataset, you can analyze several features that might provide insights into when demand and prices are highest. Here are some features to consider and ways to present your findings to your boss:

Features to Analyze:

Date/Time Features:

- Analyze the distribution of flight prices over different months, days of the week, and times of the day.
- Consider creating additional features, such as 'month,' 'day_of_week,' and 'hour_of_day.'

Seasonal Patterns:

- Explore seasonal trends by grouping data into quarters or seasons (spring, summer, fall, winter).
- Examine how flight prices vary during holidays and special events.

Duration of Flight:

- Investigate whether the length of the flight influences pricing.
- Check if there are specific periods when long-haul or short-haul flights are more expensive.

Advance Booking Period:

- Analyze how prices change based on the advance booking period (number of days between booking and departure).

Airline-Specific Trends:

- Compare how different airlines adjust their prices during peak and off-peak seasons.
- Identify if specific airlines offer lower prices during certain times.

Analysis Steps:

Descriptive Statistics:

- Present summary statistics for key features, such as mean prices, price ranges, and price distributions.

Visualizations:

- Create line charts or bar plots to visualize the average flight prices over months, days of the week, or times of the day.
- Use boxplots or violin plots to highlight seasonal patterns and identify outliers.

Statistical Tests:

- Conduct statistical tests to assess whether there are significant differences in flight prices between different periods or seasons.

Correlation Analysis:

- Examine the correlation between features like advance booking period and flight prices to identify patterns.

Presentation:

Executive Summary:

- Provide a brief executive summary outlining the goal of the analysis and the importance of identifying peak travel seasons.

Key Findings:

- Summarize key findings from the analysis, highlighting the features that strongly correlate with peak travel seasons.

Visual Representations:

- Present visualizations like line charts, bar plots, and boxplots to convey trends and patterns.
- Include visuals that compare different months, days of the week, or seasons.

Recommendations:

- Offer recommendations based on your analysis, such as suggestions for promotional campaigns, targeted marketing, or strategic pricing adjustments during peak seasons.

Limitations and Caveats:

- Acknowledge any limitations of the analysis, such as data availability, potential confounding factors, or external influences.

Next Steps:

- Propose next steps, such as further analysis, data collection, or monitoring to refine strategies over time.

By conducting a thorough analysis and presenting your findings in a clear and visually appealing manner, you can provide valuable insights to help your travel agency make informed decisions about when to offer promotions, optimize pricing, and allocate resources during peak travel seasons.

Q7. You are a data analyst for a flight booking website, and you have been asked to analyze the Flight
Price dataset to identify any trends in flight prices. What features would you analyze to identify these
trends, and what visualizations would you use to present your findings to your team?
Ans:As a data analyst for a flight booking website, analyzing the Flight Price dataset to identify trends in flight prices involves exploring various features and using visualizations to present your findings. Here are key features to analyze and suggested visualizations:

Features to Analyze:

Temporal Features:

- Date/Time: Analyze trends based on months, days of the week, and times of the day.
- Seasonal Patterns: Explore how flight prices vary across different seasons.

Advance Booking:

- Investigate trends related to the advance booking period (number of days between booking and departure).

Airline-Specific Trends:

- Examine how different airlines adjust their prices over time.

Flight Duration:

- Analyze trends related to flight duration and how it correlates with pricing.

Categorical Features:

- Explore how flight prices vary based on other categorical features like the airline, source airport, destination airport, or stops.

Holiday and Event Impact:

- Investigate how prices fluctuate during holidays and major events.

Visualizations:

Time Series Plots:

- Use line charts to show trends over time, such as monthly or weekly variations in average flight prices.
- Overlay multiple lines to compare trends for different airlines.

Seasonal Plots:

- Create seasonal plots to visualize price changes throughout the year, highlighting peaks and troughs.

Boxplots or Violin Plots:

- Display boxplots or violin plots to show the distribution of flight prices for different categorical variables, such as airlines or the number of stops.

Scatter Plots:

- Use scatter plots to explore relationships between variables, such as the relationship between the advance booking period and flight prices.

Heatmaps:

- Create heatmaps to visualize average flight prices based on combinations of features, providing a comprehensive view of trends.

Regression Analysis:

- Perform regression analysis to identify any linear relationships between independent variables and flight prices.

Interactive Dashboards:

- Build interactive dashboards using tools like Tableau or Plotly to allow your team to explore trends dynamically.

Presentation:

Introduction:

- Start with an introduction outlining the purpose of the analysis and the importance of understanding trends in flight prices.

Key Findings:

- Summarize the key trends and patterns identified during the analysis.

Visualizations:

- Showcase visualizations that best represent the identified trends, providing clear and insightful visuals.

Insights and Interpretations:

- Provide interpretations of the visualizations, explaining the implications of trends on pricing strategy.

Recommendations:

- Offer actionable recommendations based on the identified trends, suggesting potential strategies to optimize pricing or marketing efforts.

Interactive Exploration:

- If applicable, share interactive dashboards or tools to allow your team to explore trends and patterns on their own.

Conclusion:

- Conclude with a summary of key takeaways and any future steps or analyses that may be beneficial.

By conducting a comprehensive analysis and presenting your findings in a visually compelling manner, you can help your team make informed decisions regarding pricing strategies, marketing campaigns, and customer engagement.

Q8. You are a data scientist working for an airline company, and you have been asked to analyze the
Flight Price dataset to identify the factors that affect flight prices. What features would you analyze to
identify these factors, and how would you present your findings to the management team?
Ans:As a data scientist working for an airline company, analyzing the Flight Price dataset to identify factors that affect flight prices involves exploring various features and presenting insights to the management team. Here's a systematic approach:

Features to Analyze:

Date/Time Features:

- Month, Day of the Week: Analyze how flight prices vary across different months and days of the week.
- Time of the Day: Explore price fluctuations based on departure and arrival times.

Advance Booking:

- Booking Period: Investigate the impact of the advance booking period on flight prices.

Airline-Specific Factors:

- Airline: Examine how different airlines price their flights.
- Flight Duration: Analyze whether longer or shorter flights are associated with specific pricing strategies.

Route and Stops:

- Source and Destination Airports: Explore how the choice of airports influences prices.
- Number of Stops: Investigate how the number of stops affects pricing.

Categorical Features:

- Travel Class: Analyze variations in pricing for different travel classes (e.g., economy, business).

External Factors:

- Holidays and Special Events: Investigate if flight prices spike during holidays or major events.

Analysis Techniques:

Descriptive Statistics:

- Calculate summary statistics for key features, such as mean, median, and standard deviation.

Correlation Analysis:

- Perform correlation analysis to identify relationships between features and flight prices.

Regression Analysis:

- Conduct regression analysis to quantify the impact of different factors on flight prices.

Visualization Techniques:

- Use line charts, bar plots, boxplots, and scatter plots to visually represent trends and patterns in the data.

Feature Importance:

- Implement machine learning algorithms (e.g., random forests) to assess feature importance in predicting flight prices.

Presentation to Management:

Introduction:

- Start with an introduction to the analysis, outlining the objective of identifying factors affecting flight prices.

Key Findings:

- Summarize the key findings, highlighting the most influential factors affecting flight prices.

Visualizations:

- Showcase visualizations that effectively represent the identified trends and relationships.
- Use visually appealing charts to convey insights.

Interpretations and Insights:

- Provide interpretations of the visualizations, explaining the impact of each identified factor on flight prices.

Recommendations:

- Offer actionable recommendations based on the analysis, such as potential pricing strategies, marketing initiatives, or operational adjustments.

Model Performance (if applicable):

- If machine learning models were employed, provide insights into model performance and the most significant features.

Limitations and Considerations:

- Discuss any limitations or considerations related to the analysis, emphasizing transparency and potential areas for further exploration.

Interactive Dashboards (if applicable):

- Consider developing interactive dashboards or tools to allow management to explore the data and findings in more detail.

Conclusion:

- Conclude with a summary of key takeaways and the potential impact of implementing suggested strategies.

By presenting a thorough analysis and actionable insights, you can help the management team make informed decisions regarding pricing strategies, marketing efforts, and customer engagement to optimize the overall business strategy.

Google Playstore:

Q9. Load the Google Playstore dataset and examine its dimensions. How many rows and columns does the dataset have?

Ans:including the Google Playstore dataset. However, I can provide you with a general code template in Python using Pandas that you can use to load and examine the dimensions of a dataset. If you have the Google Playstore dataset in a CSV or other compatible format, you can use the following code:

python

Copy code

```
import        as



            'your_dataset.csv'










print f"The dataset has {rows} rows and {columns} columns."
```

Make sure to replace 'your_dataset.csv' with the actual file path or URL of your Google Playstore dataset. Running this code will load the dataset and print the number of rows and columns it contains.

If you encounter any issues or have specific details about the dataset (e.g., the format or structure), feel free to provide more information for further assistance.

Q10. How does the rating of apps vary by category? Create a boxplot to compare the ratings of different
app categories.
Ans:To analyze how the ratings of apps vary by category in the Google Playstore dataset and create a boxplot for comparison, you can use Python along with Pandas and Seaborn. Here's an example code template:
python

Copy code

```
import        as
import         as
import                   as



            'your_dataset.csv'





                    12  8
              'Category'    'Rating'                                 'viridis'
          'App Ratings by Category'
           'App Category'
           'Rating'
                    45      'right'
```

Replace 'your_dataset.csv' with the actual file path or URL of your Google Playstore dataset. This code will load the dataset and create a boxplot to visually compare the ratings of different app categories.

The boxplot will display the distribution of app ratings for each category, showing the median, quartiles, and potential outliers. Adjust the size, color palette, and other parameters based on your preferences.

Running this code will provide insights into how app ratings vary among different categories in your Google Playstore dataset.

Q11. Are there any missing values in the dataset? Identify any missing values and describe how they may
impact your analysis.
Ans:To identify and handle missing values in the Google Playstore dataset, you can use Python along with Pandas. Here's a code template to check for missing values and provide insights on how they may impact your analysis:
python

Copy code

```
import        as



            'your_dataset.csv'





                                         sum


print "Missing Values:"
print


                                       len                    100


print "\nPercentage of Missing Values:"
print
```

Replace 'your_dataset.csv' with the actual file path or URL of your Google Playstore dataset. Running this code will display the count and percentage of missing values for each column in the dataset.

Impact on Analysis:

Incomplete Information: Missing values may result in incomplete information for certain apps or categories, impacting the accuracy of analyses that rely on complete data.
Biased Results: If missing values are not handled properly, analyses and visualizations may produce biased results, as they may not accurately represent the entire dataset.
Imputation Decisions: Decisions about how to handle missing values (e.g., imputation or removal) can influence the overall analysis and conclusions drawn from the data.
Data Quality: Identifying and addressing missing values is essential for maintaining data quality and ensuring that analyses are based on reliable information.

Depending on the extent and nature of missing values, you may choose to handle them through imputation techniques, removal of incomplete rows or columns, or other suitable methods. It's crucial to consider the specific goals of your analysis and the potential impact of missing values on the validity of your findings.

Q12. What is the relationship between the size of an app and its rating? Create a scatter plot to visualize
the relationship.

Ans:To explore the relationship between the size of an app and its rating in the Google Playstore dataset, you can use Python along with Pandas and Seaborn for a more visually appealing scatter plot. Here's an example code template:

python

Copy code

```
import        as
import         as
import                   as



            'your_dataset.csv'
```

```
                    10  6
                  'Size'    'Rating'                             0.5
      'blue'
          'Relationship between App Size and Rating'
           'App Size'
           'Rating'
         True
```

Replace 'your_dataset.csv' with the actual file path or URL of your Google Playstore dataset. This code will load the dataset and create a scatter plot using Seaborn to visualize the relationship between app size and rating.

Adjust the size, color, and other parameters based on your preferences. The `alpha` parameter controls the transparency of the points, and you can modify it to enhance visualization.

Running this code will generate a scatter plot where each point represents an app, with the x-axis representing app size and the y-axis representing app ratings. The Seaborn scatter plot may provide additional features like better color handling and regression lines for a more comprehensive analysis of the relationship.
Q13. How does the type of app affect its price? Create a bar chart to compare average prices by app type.
Ans:To analyze how the type of app (free or paid) affects its price in the Google Playstore dataset and create a bar chart to compare average prices by app type, you can use Python with Pandas and Seaborn. Here's an example code using Seaborn:
python

Copy code

```
import       as
import        as
import                   as




            'your_dataset.csv'
```

```
                    10  6
              'Type'    'Price'                                  None          'Set2'
          'Average Prices by App Type'
           'App Type'
           'Average Price'
```

Replace 'your_dataset.csv' with the actual file path or URL of your Google Playstore dataset. This code will load the dataset, calculate the average prices for each app type, and create a bar chart using Seaborn to visualize the comparison.

Adjust the figure size, color palette, and other parameters based on your preferences. The resulting bar chart will display the average prices for free and paid apps, providing insights into how the type of app may affect its pricing.

Q14. What are the top 10 most popular apps in the dataset? Create a frequency table to identify the apps
with the highest number of installs.
Ans:To identify the top 10 most popular apps in the Google Playstore dataset based on the number of installs, you can use Python with Pandas. Here's an example code to create a frequency table and identify the apps with the highest number of installs:
python

Copy code

```
import        as



            'your_dataset.csv'






                       'App'  'Installs'  sum                          False



                                     10
print "Top 10 Most Popular Apps by Installs:"
print
```

Replace 'your_dataset.csv' with the actual file path or URL of your Google Playstore dataset. This code will load the dataset, create a frequency table by summing the installs for each app, and then display the top 10 most popular apps based on the total number of installs.

Adjust the code based on your specific dataset structure and column names if needed. Running this code will provide you with a list of the top 10 most popular apps along with their total installs.

Q15. A company wants to launch a new app on the Google Playstore and has asked you to analyze the
Google Playstore dataset to identify the most popular app categories. How would you approach this
task, and what features would you analyze to make recommendations to the company?
Ans:To identify the most popular app categories on the Google Playstore and provide recommendations to the company, you can follow these steps:

Approach:

Data Exploration:

- Begin by loading and exploring the Google Playstore dataset to understand its structure and contents.
- Examine the available features, data types, and the overall distribution of data.

Define "Popularity":

- Clearly define what "popularity" means in the context of the company's goals. Popularity can be measured by the number of installs, ratings, or other relevant metrics.

Identify Relevant Features:

- Determine which features in the dataset are relevant for assessing popularity and app categories.
- Key features may include 'Category', 'Installs', 'Rating', 'Reviews', 'Price', and 'Genre'.

Analyze App Categories:

- Group the dataset by the 'Category' feature and calculate relevant metrics (e.g., average installs, average ratings) for each category.

Visualizations:

- Create visualizations such as bar charts or pie charts to represent the distribution of installs or ratings across different app categories.

Statistical Analysis:

- Conduct statistical analysis to identify significant differences between app categories, helping to prioritize the most popular ones.

Consider Other Factors:

- Analyze other relevant factors that may impact popularity, such as app size, content rating, and price.

Recommendations:

- Based on the analysis, provide clear recommendations to the company regarding the most popular app categories.
- Consider suggesting categories with a high average number of installs, positive ratings, or other metrics aligned with the company's objectives.

Code Example:

python

Copy code

```
import        as
import         as
import                   as


            'your_dataset.csv'




                       'Category'  'Installs'                                       False


                    12  6

        'viridis'
                    90
          'Average Installs by App Category'
           'App Category'
           'Average Installs'
```

Replace 'your_dataset.csv' with the actual file path or URL of your Google Playstore dataset. This code provides a bar chart visualizing the average number of installs for each app category, helping to identify the most popular categories.

Recommendations:

- Consider User Engagement: High average installs and positive ratings may indicate user engagement, making these categories favorable for the company.
- Explore Niche Categories: Investigate categories with moderate installs but high engagement, as they may represent niche markets with less competition.
- Competitor Analysis: Examine popular apps in identified categories and assess the competitive landscape to identify opportunities and challenges.
- Adaptability: Consider the company's strengths, goals, and target audience when making recommendations. Certain categories may align better with the company's expertise or objectives.

By following this approach, the company can make informed decisions about the most popular app categories for their new app launch, maximizing the potential for success in the Google Playstore.