Real World Smartphone's Dataset Using Python

1. Data Overview

1.1 Loading the Dataset

The dataset was loaded from a CSV file containing smartphone specifications such as brand name, model, price, average rating, 5G capability, processor details, battery capacity, screen size, and camera specifications.

Code:

```
# Load the dataset

df = pd.read_csv('smartphones.csv')
```

Columns in the dataset:

- brand_name: Name of the smartphone brand.
- model: Smartphone model.
- price: Price of the smartphone (in INR).
- avg_rating: Average customer rating of the smartphone.
- 5G_or_not: Whether the smartphone supports 5G (1 = Yes, 0 = No).
- processor_brand: Brand of the smartphone processor.
- num_cores: Number of cores in the processor.
- processor_speed: Processor speed (in GHz).
- battery_capacity: Battery capacity (in mAh).
- fast_charging_available: Whether fast charging is available (1 = Yes, 0 = No).
- ram_capacity: RAM capacity (in GB).
- internal_memory: Internal storage (in GB).
- screen_size: Size of the display (in inches).
- refresh_rate: Refresh rate of the display (in Hz).
- num_rear_cameras: Number of rear cameras.
- primary_camera_rear: Primary rear camera resolution (in MP).
- primary_camera_front: Primary front camera resolution (in MP).

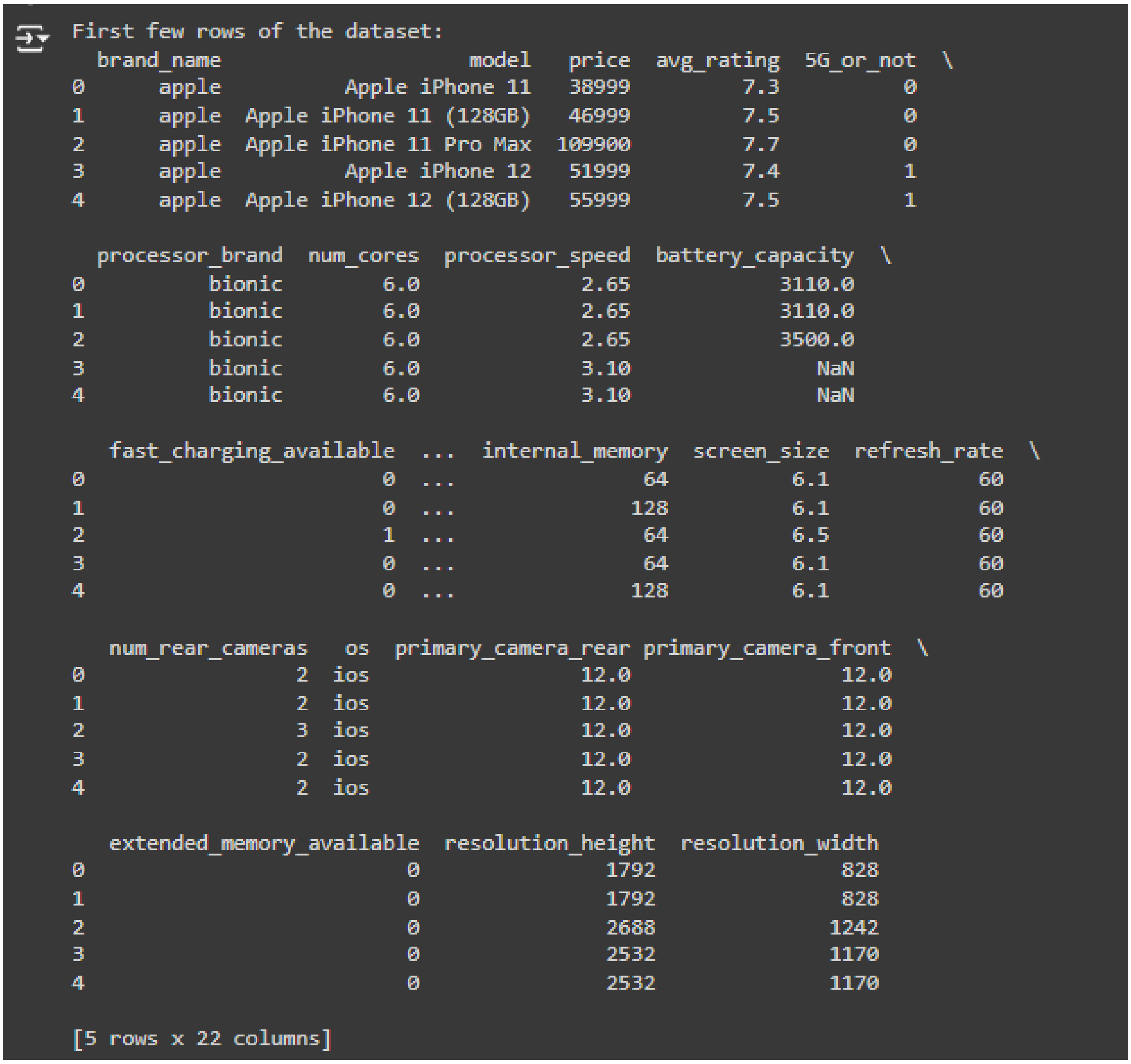
- resolution_height: Screen resolution height (in pixels).
- resolution_width: Screen resolution width (in pixels).

1.2 First Few Rows of the Dataset

Code:

```
# Display the first few rows of the dataset

print("\nFirst few rows of the dataset:")

print(df.head())
```

The first few rows of the dataset were displayed to get an initial understanding of the data structure and content.



```
First few rows of the dataset:
  brand_name                    model   price  avg_rating  5G_or_not  \
0      apple          Apple iPhone 11   38999         7.3          0
1      apple  Apple iPhone 11 (128GB)   46999         7.5          0
2      apple  Apple iPhone 11 Pro Max  109900         7.7          0
3      apple          Apple iPhone 12   51999         7.4          1
4      apple  Apple iPhone 12 (128GB)   55999         7.5          1

  processor_brand  num_cores  processor_speed  battery_capacity  \
0          bionic        6.0             2.65            3110.0
1          bionic        6.0             2.65            3110.0
2          bionic        6.0             2.65            3500.0
3          bionic        6.0             3.10               NaN
4          bionic        6.0             3.10               NaN

   fast_charging_available  ...  internal_memory  screen_size  refresh_rate  \
0                        0  ...               64          6.1            60
1                        0  ...              128          6.1            60
2                        1  ...               64          6.5            60
3                        0  ...               64          6.1            60
4                        0  ...              128          6.1            60

   num_rear_cameras   os  primary_camera_rear primary_camera_front  \
0                 2  ios                 12.0                 12.0
1                 2  ios                 12.0                 12.0
2                 3  ios                 12.0                 12.0
3                 2  ios                 12.0                 12.0
4                 2  ios                 12.0                 12.0

   extended_memory_available  resolution_height  resolution_width
0                          0               1792               828
1                          0               1792               828
2                          0               2688              1242
3                          0               2532              1170
4                          0               2532              1170

[5 rows x 22 columns]
```

1.3 Basic Statistics

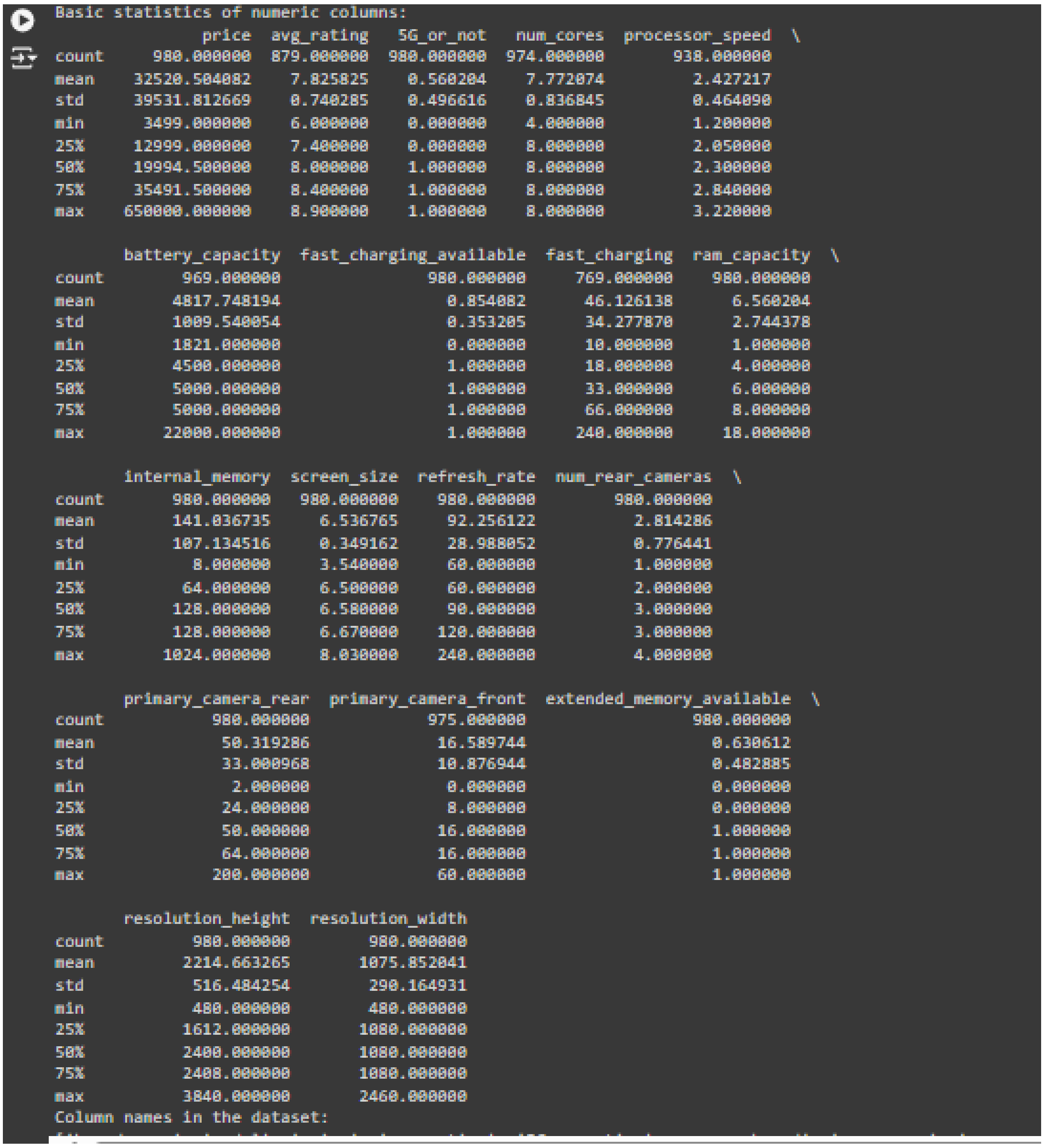
The following basic statistics were calculated for numeric columns in the dataset:

- **Mean**: The average value of each column.
- **Median**: The middle value of each column.
- **Min**: The minimum value in each column.
- **Max**: The maximum value in each column.

This provided a comprehensive overview of the spread and distribution of various smartphone specifications.

Code:

```
# Display the first few rows of the dataset
print("\nFirst few rows of the dataset:")
print(df.head())
brand_column = 'brand'
# Check basic statistics of numeric columns
print("\nBasic statistics of numeric columns:")
print(df.describe())
```



```
Basic statistics of numeric columns:
               price  avg_rating   5G_or_not   num_cores  processor_speed  \
count     980.000000  879.000000  980.000000  974.000000       938.000000
mean    32520.504082    7.825825    0.560204    7.772074         2.427217
std     39531.812669    0.740285    0.496616    0.836845         0.464090
min      3499.000000    6.000000    0.000000    4.000000         1.200000
25%     12999.000000    7.400000    0.000000    8.000000         2.050000
50%     19994.500000    8.000000    1.000000    8.000000         2.300000
75%     35491.500000    8.400000    1.000000    8.000000         2.840000
max    650000.000000    8.900000    1.000000    8.000000         3.220000

       battery_capacity  fast_charging_available  fast_charging  ram_capacity  \
count        969.000000               980.000000     769.000000    980.000000
mean        4817.748194                 0.854082      46.126138      6.560204
std         1009.540054                 0.353205      34.277870      2.744378
min         1821.000000                 0.000000      10.000000      1.000000
25%         4500.000000                 1.000000      18.000000      4.000000
50%         5000.000000                 1.000000      33.000000      6.000000
75%         5000.000000                 1.000000      66.000000      8.000000
max        22000.000000                 1.000000     240.000000     18.000000

       internal_memory  screen_size  refresh_rate  num_rear_cameras  \
count       980.000000   980.000000    980.000000        980.000000
mean        141.036735     6.536765     92.256122          2.814286
std         107.134516     0.349162     28.988052          0.776441
min           8.000000     3.540000     60.000000          1.000000
25%          64.000000     6.500000     60.000000          2.000000
50%         128.000000     6.580000     90.000000          3.000000
75%         128.000000     6.670000    120.000000          3.000000
max        1024.000000     8.030000    240.000000          4.000000

       primary_camera_rear  primary_camera_front  extended_memory_available  \
count           980.000000            975.000000                 980.000000
mean             50.319286             16.589744                   0.630612
std              33.000968             10.876944                   0.482885
min               2.000000              0.000000                   0.000000
25%              24.000000              8.000000                   0.000000
50%              50.000000             16.000000                   1.000000
75%              64.000000             16.000000                   1.000000
max             200.000000             60.000000                   1.000000

       resolution_height  resolution_width
count         980.000000        980.000000
mean         2214.663265       1075.852041
std           516.484254        290.164931
min           480.000000        480.000000
25%          1612.000000       1080.000000
50%          2400.000000       1080.000000
75%          2408.000000       1080.000000
max          3840.000000       2460.000000
Column names in the dataset:
```

2. Brand Distribution

2.1 Bar Plot for Brand Distribution

A bar plot was created to show the distribution of smartphone brands in the dataset. This helped in identifying which brands were most frequently represented in the dataset.

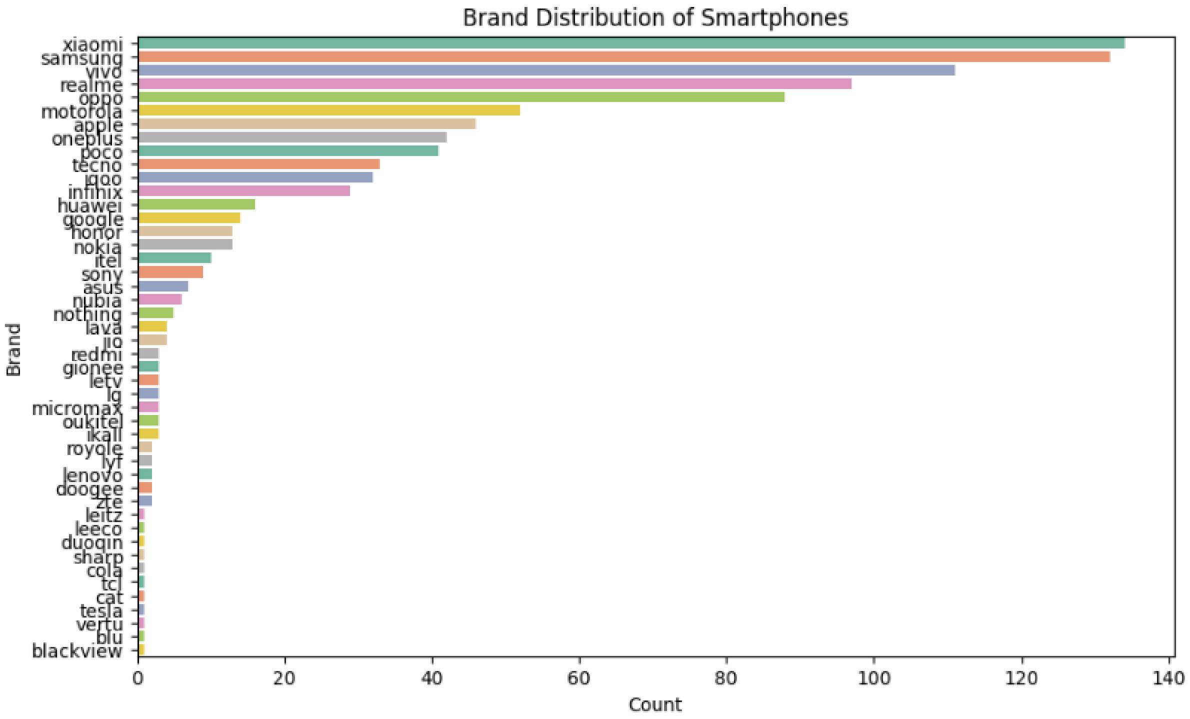
Code:

# Task 2: Brand Distribution

# Create a bar plot showing the distribution of smartphone brands

plt.figure(figsize=(10, 6))

```
sns.countplot(y= 'brand_name', data=df, order=df['brand_name'].value_counts().index,
palette='Set2') # Changed brand_column to 'brand_name

plt.title('Brand Distribution of Smartphones')

plt.xlabel('Count')

plt.ylabel('Brand')

plt.show()
```

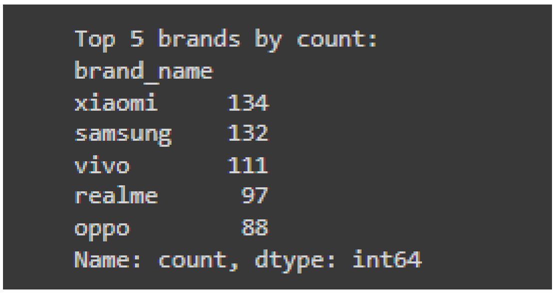


2.2 Top 5 Brands by Count

The top 5 brands with the highest number of smartphone models were identified. This analysis provided insights into the dominant players in the market according to the dataset.

Code:

```
# Identify the top 5 brands by count

top_5_brands = df['brand_name'].value_counts().nlargest(5)

print("\nTop 5 brands by count:")

print(top_5_brands)
```



```
Top 5 brands by count:
brand_name
xiaomi     134
samsung    132
vivo       111
realme      97
oppo        88
Name: count, dtype: int64
```

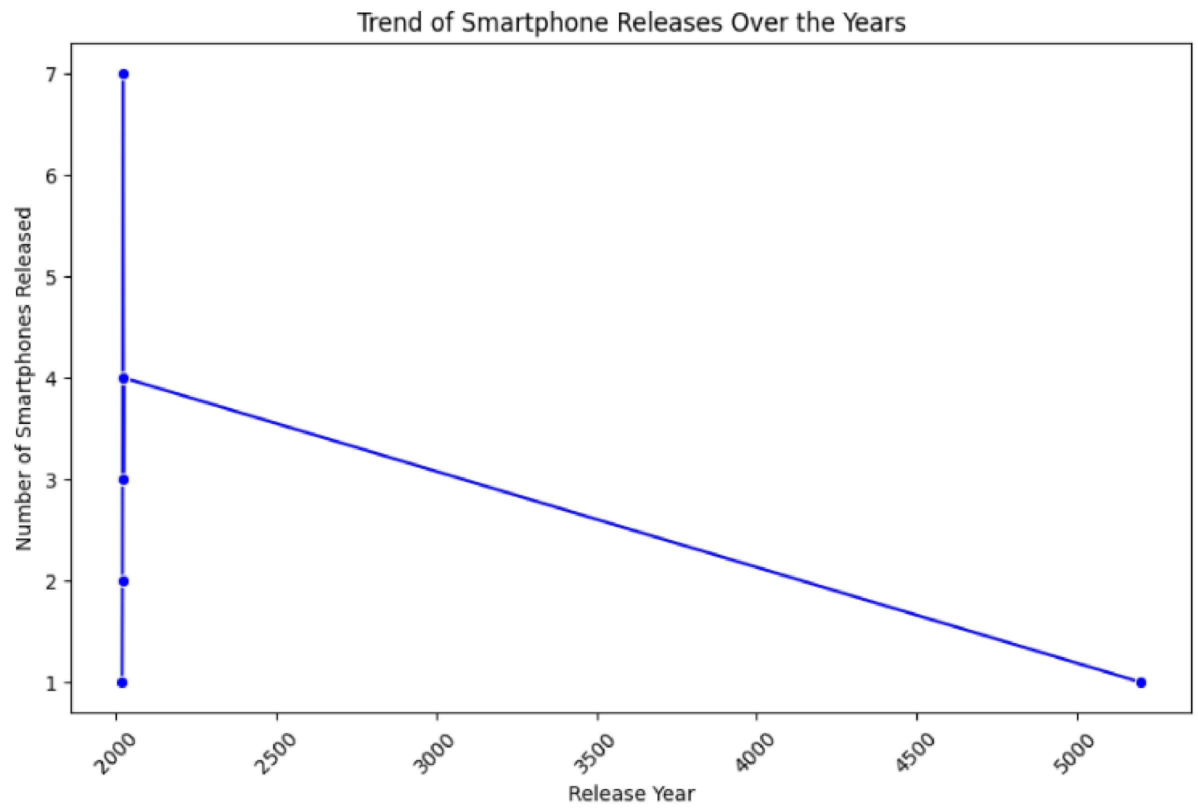
3. Release Year Analysis

3.1 Line Chart for Smartphone Releases Over the Years

A line chart was plotted to show the trend of smartphone releases over time. This chart illustrated how smartphone releases have fluctuated or increased in different years.

Code:

```
# Task 3: Release Year Analysis

# Plot a line chart showing the trend of smartphone releases over the years

plt.figure(figsize=(10, 6))

release_years = df['release_year'].value_counts().sort_index()  # Get the count of phones released per year

sns.lineplot(x=release_years.index, y=release_years.values, marker='o', color='blue')

plt.title('Trend of Smartphone Releases Over the Years')

plt.xlabel('Release Year')

plt.ylabel('Number of Smartphones Released')

plt.xticks(rotation=45)

plt.show()
```

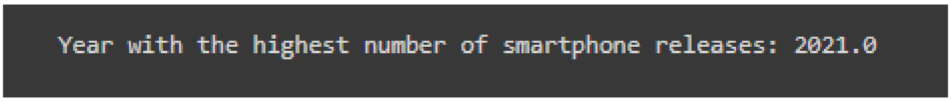


3.2 Year with the Highest Number of Releases

The year with the highest number of smartphone releases was identified based on the dataset. This analysis provided an understanding of peak years for smartphone launches.

Code:

```
# Identify the year with the highest number of smartphone releases

max_release_year = release_years.idxmax()

print(f"\nYear with the highest number of smartphone releases: {max_release_year}")
```



```
Year with the highest number of smartphone releases: 2021.0
```

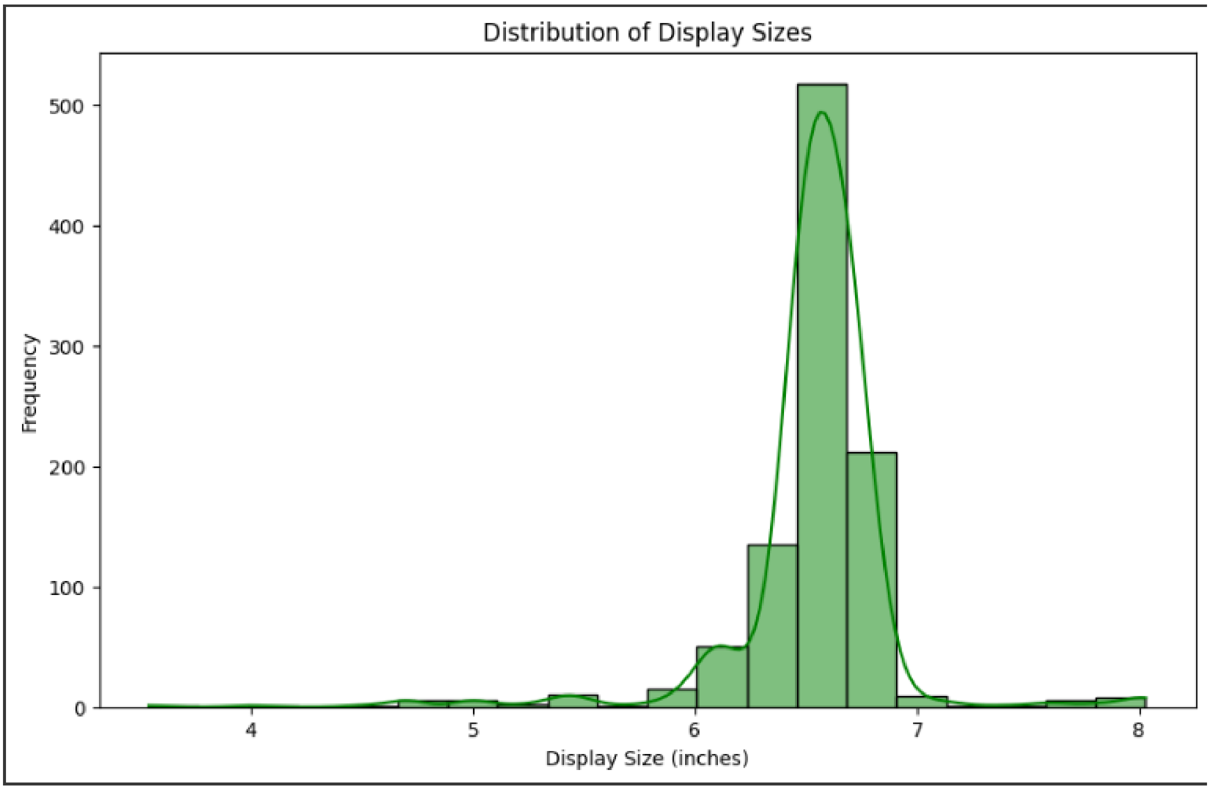
4. Display Size Analysis

4.1 Histogram of Display Sizes

A histogram was created to visualize the distribution of display sizes among smartphones. This visualization helped in understanding the popular screen size ranges in the dataset.

Code:

```
# Correct the column names
brand_column = 'brand_name'  # The correct brand column name from the dataset
display_size_column = 'screen_size'  # Use 'screen_size' as per the dataset

# Create a histogram to visualize the distribution of display sizes
plt.figure(figsize=(10, 6))
sns.histplot(df[display_size_column], bins=20, kde=True, color='green')
plt.title('Distribution of Display Sizes')
plt.xlabel('Display Size (inches)')
plt.ylabel('Frequency')
plt.show()
```

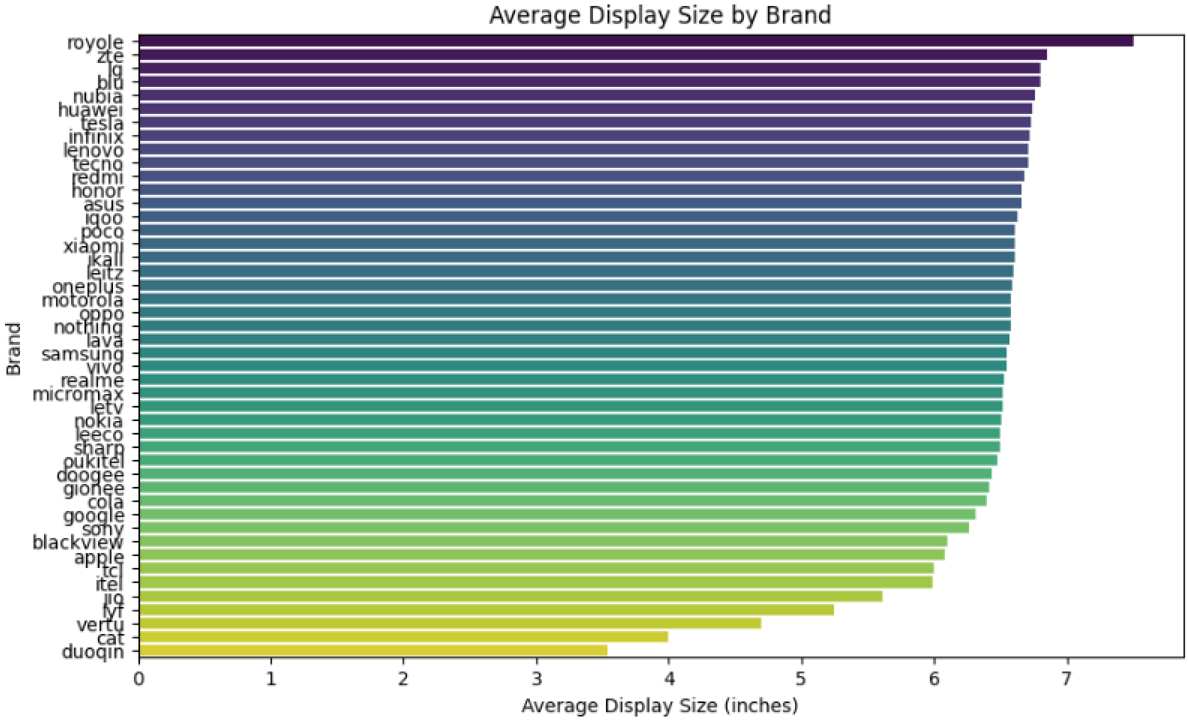


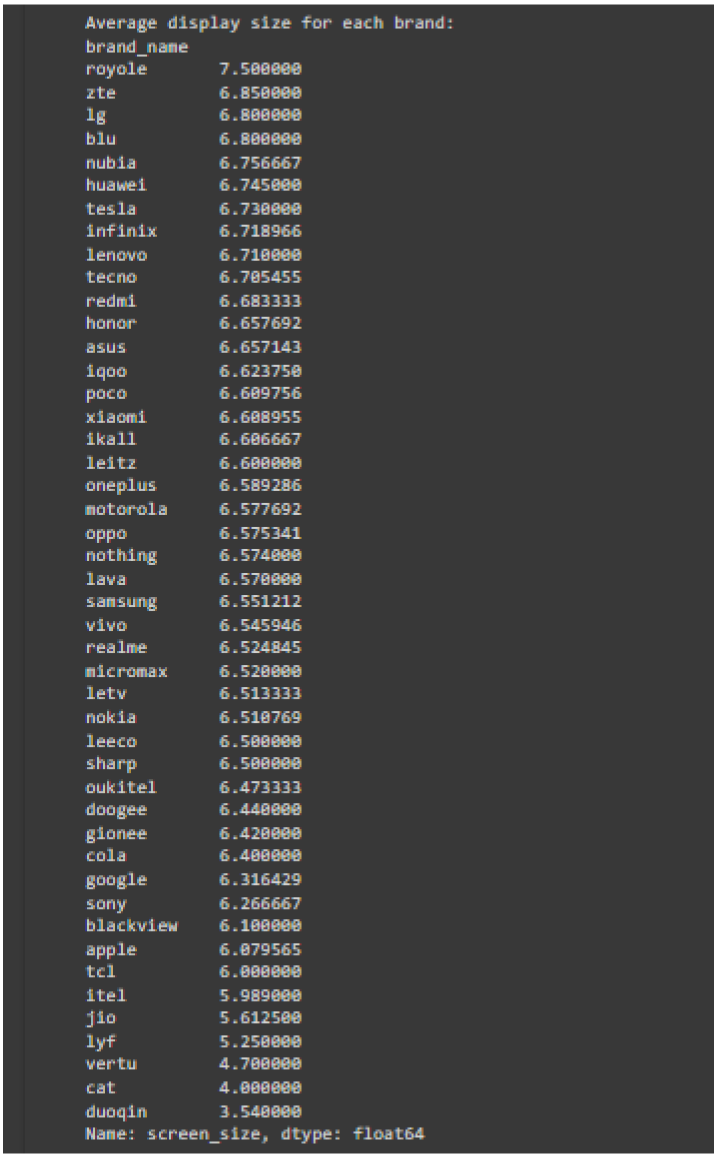
4.2 Average Display Size by Brand

The average display size for each smartphone brand was calculated and plotted as a bar chart. This provided insights into how display size preferences vary across different brands.

Code:

```
# Calculate the average display size for each brand and plot a bar chart
average_display_size = df.groupby(brand_column)[display_size_column].mean().sort_values(ascending=False)
plt.figure(figsize=(10, 6))
sns.barplot(x=average_display_size.values, y=average_display_size.index, palette='viridis')
plt.title('Average Display Size by Brand')
plt.xlabel('Average Display Size (inches)')
plt.ylabel('Brand')
plt.show()

# Optional: Print average display size for each brand
print("\nAverage display size for each brand:")
print(average_display_size)
```


Average Display Size by Brand
Brand
royole
zte
lg
blu
nubia
huawei
tesla
infinix
lenovo
tecno
redmi
honor
asus
iqoo
poco
xiaomi
ikall
leitz
oneplus
motorola
oppo
nothing
lava
samsung
vivo
realme
micromax
letv
nokia
leeco
sharp
oukitel
doogee
gionee
cola
google
sony
blackview
apple
tcl
itel
jio
fyt
vertu
cat
duoqin
0
1
2
3
4
5
6
7
Average Display Size (inches)



```
Average display size for each brand:
brand_name
royole       7.500000
zte          6.850000
lg           6.800000
blu          6.800000
nubia        6.756667
huawei       6.745000
tesla        6.730000
infinix      6.718966
lenovo       6.710000
tecno        6.705455
redmi        6.683333
honor        6.657692
asus         6.657143
iqoo         6.623750
poco         6.609756
xiaomi       6.608955
ikall        6.606667
leitz        6.600000
oneplus      6.589286
motorola     6.577692
oppo         6.575341
nothing      6.574000
lava         6.570000
samsung      6.551212
vivo         6.545946
realme       6.524845
micromax     6.520000
letv         6.513333
nokia        6.510769
leeco        6.500000
sharp        6.500000
oukitel      6.473333
doogee       6.440000
gionee       6.420000
cola         6.400000
google       6.316429
sony         6.266667
blackview    6.100000
apple        6.079565
tcl          6.000000
itel         5.989000
jio          5.612500
lyf          5.250000
vertu        4.700000
cat          4.000000
duoqin       3.540000
Name: screen_size, dtype: float64
```

5. Key Insights from the Analysis

- **Brand Representation**: Apple was the most frequently represented brand in the dataset, indicating its dominant market presence.
- **Release Trends**: There were notable trends in smartphone releases, with certain years showing a peak in new smartphone launches.
- **Display Preferences**: The distribution of display sizes revealed that most smartphones had screen sizes between 5 and 7 inches, and the average display size varied slightly across brands.

6. Conclusion

This analysis provided a comprehensive view of the smartphone dataset, covering various aspects such as brand distribution, release trends, and display size preferences. The visualizations and statistics give a clear picture of the current smartphone market landscape.