AJEENKYA
D Y PATIL UNIVERSITY
THE INNOVATION UNIVERSITY

A
MINI PROJECT REPORT ON

“Analysis of the Used Bike Market”

FOR

Term Work Examination

Bachelors of Computer Application in Artificial Intelligence and Machine Learning – (BCA-AIML)

Year 2024-2025

Ajeenkya DY Patil University, Pune

-Submitted By-

Mr. Shreyash Patil

Under the guidance of

Prof. Vivek More

Ajeenkya DY Patil University

D Y Patil Knowledge City,
Charholi Bk. Via Lohegaon,
Pune - 412105
Maharashtra (India)

.

Date: / / 2025

CERTIFICATE

This is to certified that___Shreyash Patil____
A student's of **BCA(AIML) Sem-4** URN No 2023-B-07042005A
has Successfully Completed the Dashboard Report On

"Analysis of the Used Bike Market"

As per the requirement of
Ajeenkya DY Patil University, Pune was carried out under my supervision.
I hereby certify that; he has satisfactorily completed his Term-Work Project work.

Place: - Pune

Examiner

INDEX

Introduction

The pre-owned two-wheeler market in India has seen significant growth in recent years, fueled by increasing demand for affordable transportation and the rising cost of new vehicles. This dataset, titled **"BIKE DETAILS.csv,"** provides comprehensive information about various used bikes that have been listed for resale. It includes essential attributes such as the bike name, selling price, manufacturing year, seller type (individual or dealer), ownership history, kilometers driven, and the original ex-showroom price (where available). These variables play a crucial role in determining a bike's market value and help both buyers and sellers make informed decisions.

Understanding used bike trends through data-driven insights can offer a valuable perspective into customer preferences, popular models, and price fluctuations based on usage and ownership. By analyzing the data, one can uncover how much value different bikes retain over time, how seller types affect pricing, and what role mileage and ownership history play in resale value.

This dataset is not only useful for individual buyers and sellers but also for businesses, financial analysts, and researchers looking to study the dynamics of the second-hand automobile market. It offers a solid foundation for conducting exploratory data analysis, predictive modeling, and market strategy development in the two-wheeler segment.

Objectives

The key objectives of the **"BIKE DETAILS.csv"** dataset are as follows:

1. **Price Analysis and Depreciation Study**:
   Compare selling prices with the original ex-showroom prices to assess how different bikes depreciate over time, and identify which brands or models retain value better.

2. **Ownership and Usage Insights**:
   Evaluate the impact of ownership history (first, second, third owner, etc.) and kilometers driven on the resale value, offering insights into what buyers prioritize.

3. **Identify High-Demand Bikes**:
   Discover which bikes are most frequently listed and sold in the market, indicating popularity and brand value.

4. **Assist Decision-Makers**:
   Provide data-driven insights that can help buyers, sellers, and dealers make informed decisions, negotiate better prices, or optimize their listings.

5. **Contribute to Market Research**:
   Serve as a foundational dataset for researchers and analysts studying the two-wheeler market in emerging economies like India.

Methodology and Aproach

To extract meaningful insights from the **"BIKE DETAILS.csv"** dataset, the following methodology will be adopted:

1. **Data Collection and Loading**:
   The dataset is sourced from a CSV file containing details of 1,061 used bikes, including variables such as name, selling price, year, seller type, owner, kilometers driven, and ex-showroom price.

2. **Data Cleaning and Preprocessing**:
   - Handle missing values, particularly in the ex_showroom_price column.
   - Convert relevant columns to appropriate data types (e.g., date, numeric).
   - Standardize categorical values (e.g., seller type and owner categories).
   - Remove or correct any obvious anomalies (e.g., negative prices, future years).

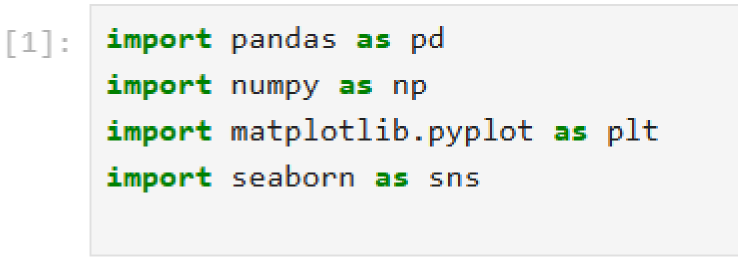
3. **Exploratory Data Analysis (EDA)**:
   - Analyze distributions of key variables like selling price, year of manufacture, and km driven.
   - Identify correlations between features using statistical plots and correlation matrices.
   - Group data by bike model, year, and seller type to reveal patterns and trends.

4. **Visualization**:
   - Use graphs such as bar charts, histograms, box plots, and scatter plots to visually present findings.
   - Generate comparative insights across various groups (e.g., individual vs. dealer sellers).

5. **Interpretation and Reporting**:
   - Summarize key insights and trends.
   - Make actionable recommendations for stakeholders such as buyers, sellers, and dealerships.

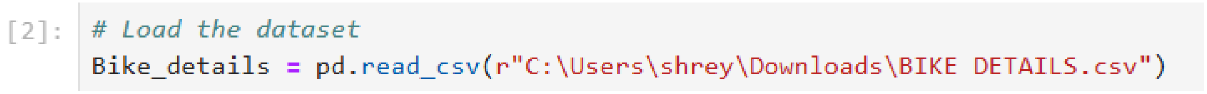
Implementation & Code

1.Importing necessary libraries-



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

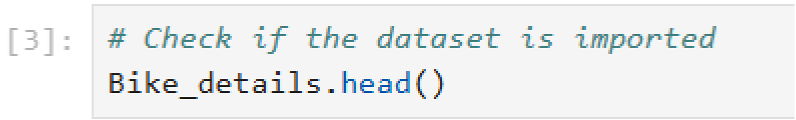
2.Importing Data



```
[2]: # Load the dataset
     Bike_details = pd.read_csv(r"C:\Users\shrey\Downloads\BIKE DETAILS.csv")
```

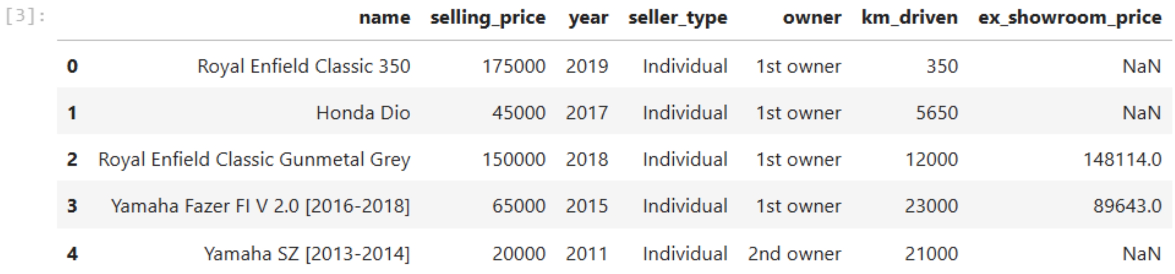
The dataset is read into a DataFrame using Pandas. This serves as the foundation for the subsequent analysis.

3. Check if the dataset is imported



```
[3]:  # Check if the dataset is imported
      Bike_details.head()
```

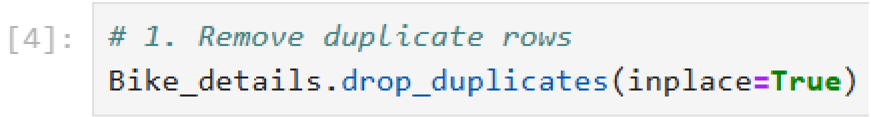
3.Output



[3]:

	name	selling_price	year	seller_type	owner	km_driven	ex_showroom_price
0	Royal Enfield Classic 350	175000	2019	Individual	1st owner	350	NaN
1	Honda Dio	45000	2017	Individual	1st owner	5650	NaN
2	Royal Enfield Classic Gunmetal Grey	150000	2018	Individual	1st owner	12000	148114.0
3	Yamaha Fazer FI V 2.0 [2016-2018]	65000	2015	Individual	1st owner	23000	89643.0
4	Yamaha SZ [2013-2014]	20000	2011	Individual	2nd owner	21000	NaN

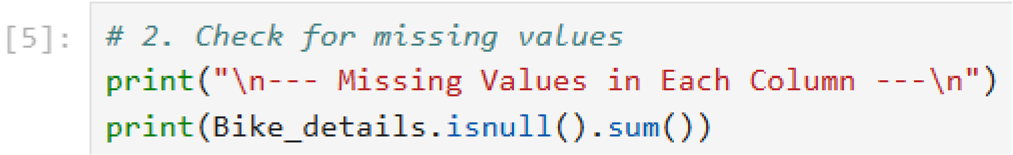
4.Removing Duplicates



```
[4]:  # 1. Remove duplicate rows
      Bike_details.drop_duplicates(inplace=True)
```

Duplicate rows may occur due to errors in data entry or merging. Removing duplicates ensures that each data point is unique and the analysis is not biased by repetition.

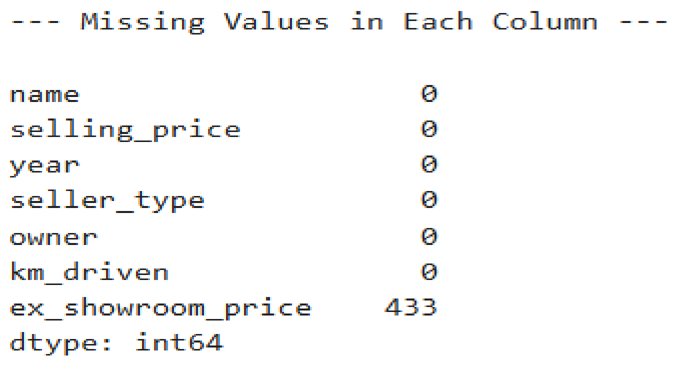
5.Checking for missing values



```
[5]:  # 2. Check for missing values
      print("\n--- Missing Values in Each Column ---\n")
      print(Bike_details.isnull().sum())
```

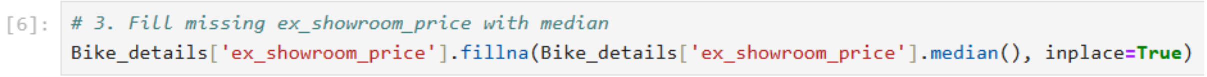
This command displays the total number of missing values in each column. Understanding which columns contain null values is crucial for applying proper imputation strategies or dropping irrelevant data.

5.Output



```
--- Missing Values in Each Column ---

name                   0
selling_price          0
year                   0
seller_type            0
owner                  0
km_driven              0
ex_showroom_price    433
dtype: int64
```

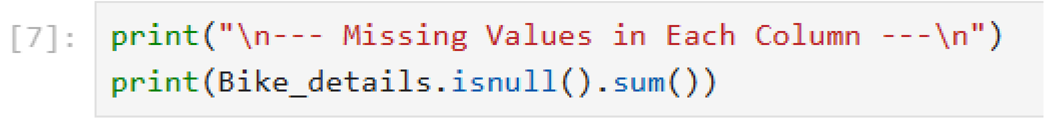
6.Filling missing ex_showroom_price with Median



```
[6]: # 3. Fill missing ex_showroom_price with median
     Bike_details['ex_showroom_price'].fillna(Bike_details['ex_showroom_price'].median(), inplace=True)
```

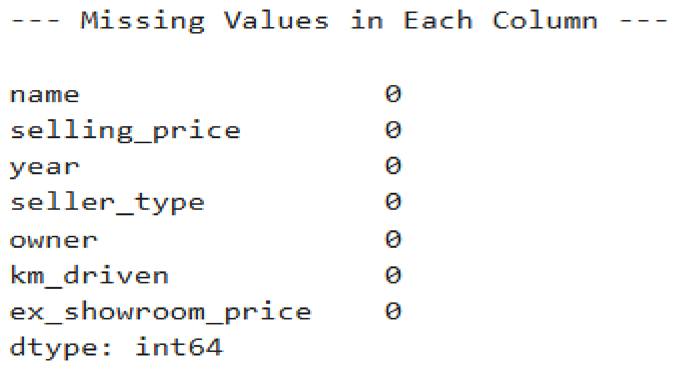
The missing values in the ex_showroom_price column are replaced with the median. Median is used instead of mean to reduce the influence of outliers in pricing.

7.Rechecking missing values



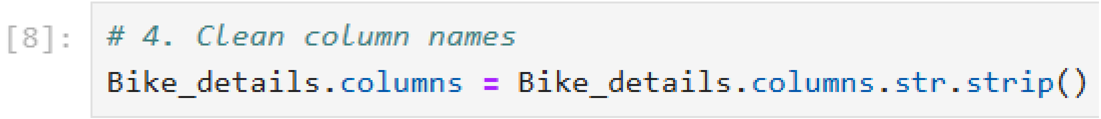
```
[7]: print("\n--- Missing Values in Each Column ---\n")
     print(Bike_details.isnull().sum())
```

7.Output



```
--- Missing Values in Each Column ---

name                 0
selling_price        0
year                 0
seller_type          0
owner                0
km_driven            0
ex_showroom_price    0
dtype: int64
```

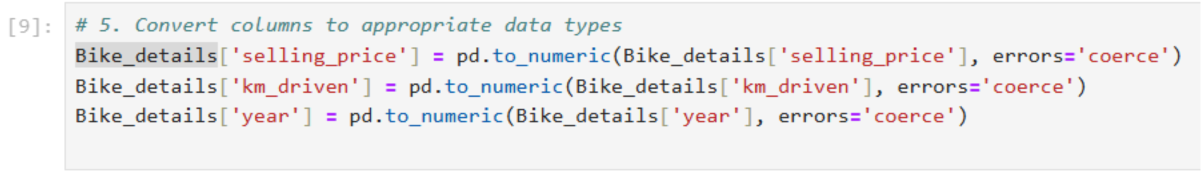
8.Cleaning column names



```
[8]: # 4. Clean column names
     Bike_details.columns = Bike_details.columns.str.strip()
```

This removes any leading or trailing whitespace in column names, which can cause bugs when referencing columns.

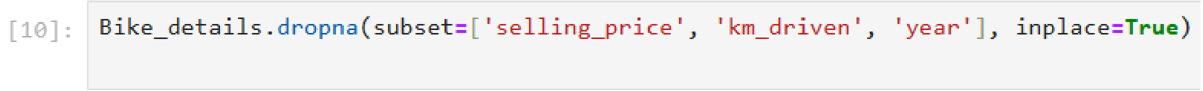
9.Converting Data types



```
[9]: # 5. Convert columns to appropriate data types
     Bike_details['selling_price'] = pd.to_numeric(Bike_details['selling_price'], errors='coerce')
     Bike_details['km_driven'] = pd.to_numeric(Bike_details['km_driven'], errors='coerce')
     Bike_details['year'] = pd.to_numeric(Bike_details['year'], errors='coerce')
```

Ensures that key fields like price, kilometers driven, and year are treated as numeric values. Invalid entries are converted to NaN, which can then be cleaned up.

10. Drop Rows with Missing Essential Data



```
[10]: Bike_details.dropna(subset=['selling_price', 'km_driven', 'year'], inplace=True)
```

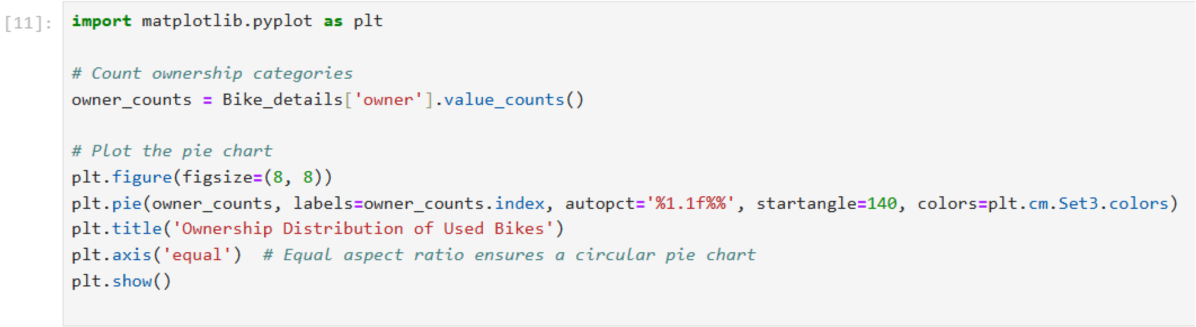
Any rows missing crucial values needed for analysis are removed to maintain the integrity of further processing.

Results and Visualizations

1.Pie Chart

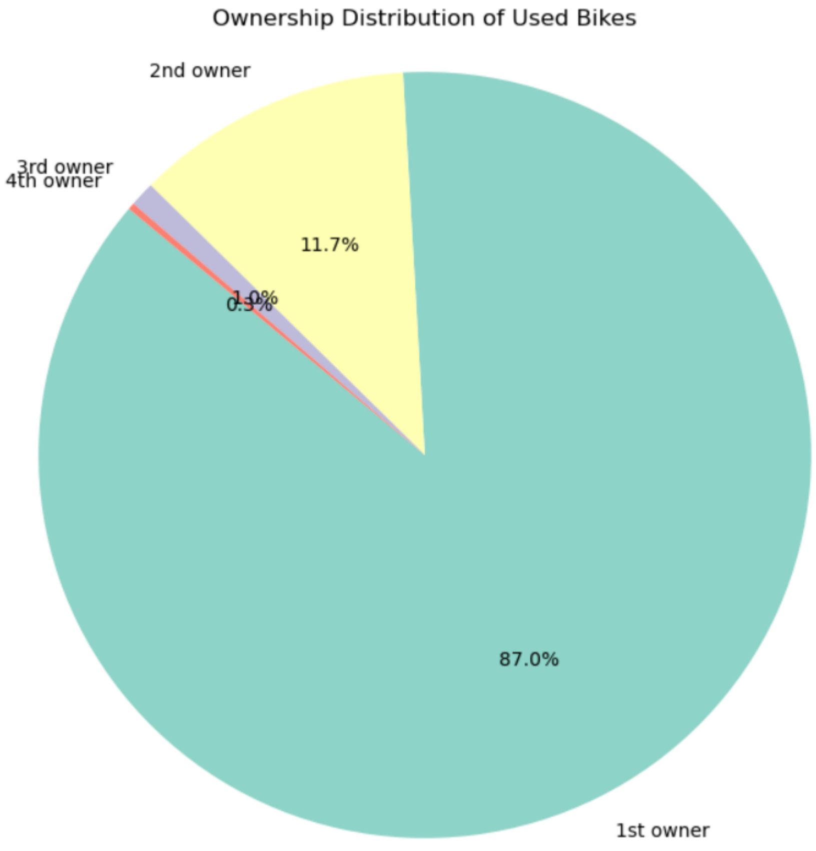
Ownership Distribution: 1st , 2nd, 3rd and 4th bike owners

Code



```
[11]: import matplotlib.pyplot as plt

      # Count ownership categories
      owner_counts = Bike_details['owner'].value_counts()

      # Plot the pie chart
      plt.figure(figsize=(8, 8))
      plt.pie(owner_counts, labels=owner_counts.index, autopct='%1.1f%%', startangle=140, colors=plt.cm.Set3.colors)
      plt.title('Ownership Distribution of Used Bikes')
      plt.axis('equal')  # Equal aspect ratio ensures a circular pie chart
      plt.show()
```

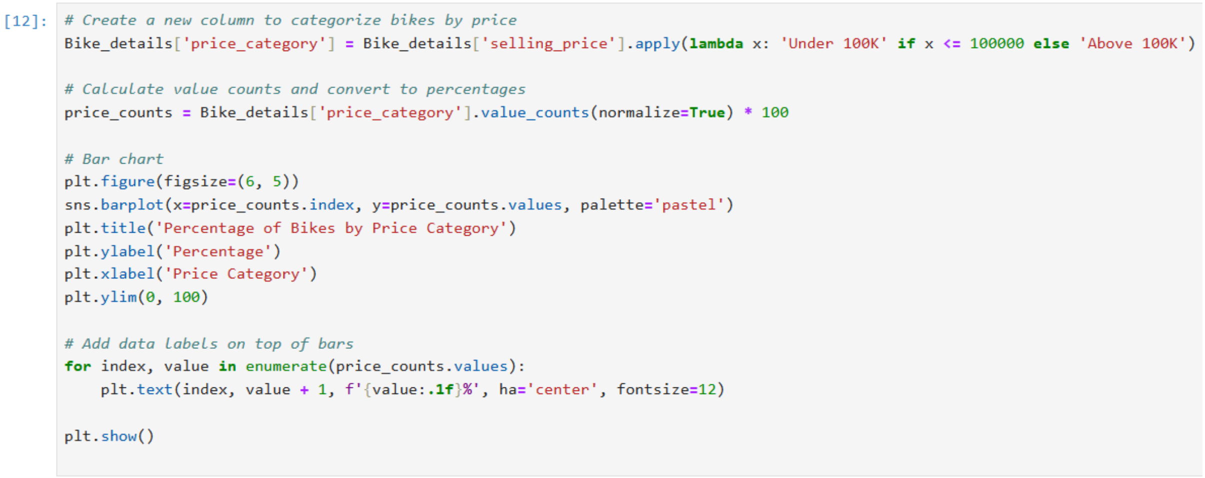
Output



2.Bar Chart

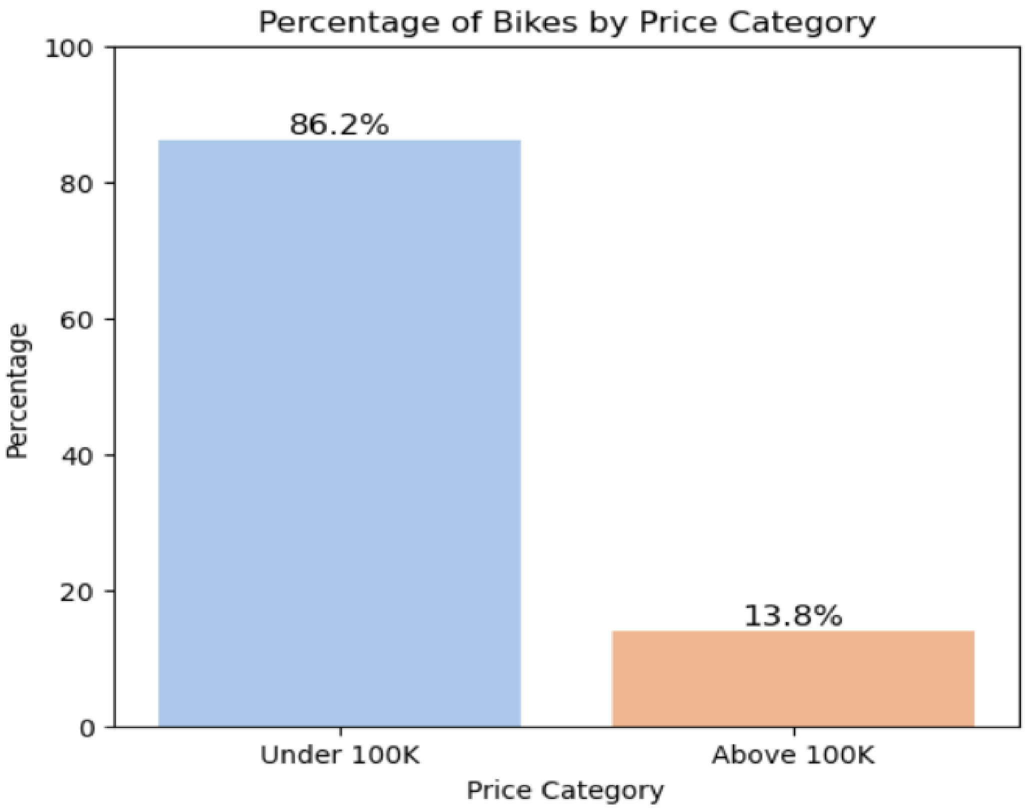
Percentage of Bikes by price category

Code



```
[12]: # Create a new column to categorize bikes by price
      Bike_details['price_category'] = Bike_details['selling_price'].apply(lambda x: 'Under 100K' if x <= 100000 else 'Above 100K')

      # Calculate value counts and convert to percentages
      price_counts = Bike_details['price_category'].value_counts(normalize=True) * 100

      # Bar chart
      plt.figure(figsize=(6, 5))
      sns.barplot(x=price_counts.index, y=price_counts.values, palette='pastel')
      plt.title('Percentage of Bikes by Price Category')
      plt.ylabel('Percentage')
      plt.xlabel('Price Category')
      plt.ylim(0, 100)

      # Add data labels on top of bars
      for index, value in enumerate(price_counts.values):
          plt.text(index, value + 1, f'{value:.1f}%', ha='center', fontsize=12)

      plt.show()
```

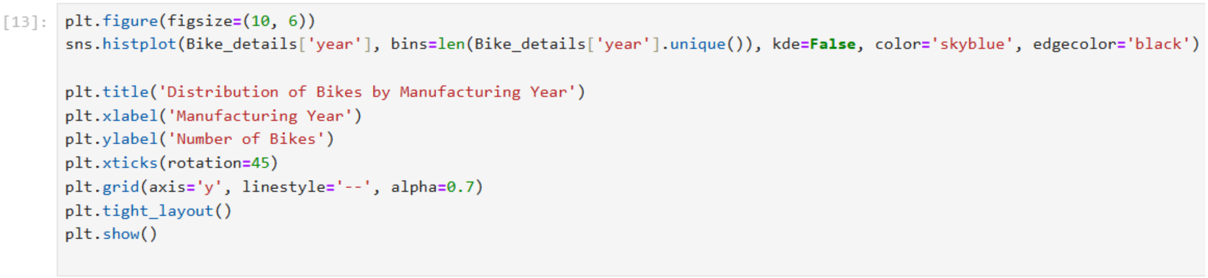
Output



3.Histogram

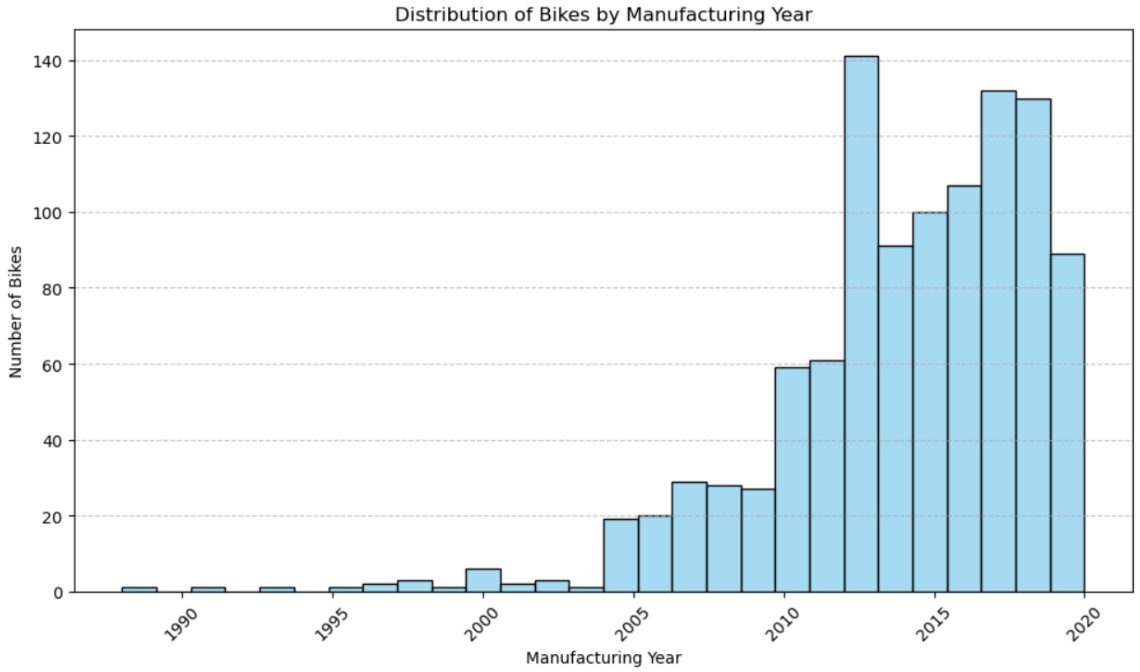
Distribution of bikes by Manufacturing Year

Code



```
[13]: plt.figure(figsize=(10, 6))
      sns.histplot(Bike_details['year'], bins=len(Bike_details['year'].unique()), kde=False, color='skyblue', edgecolor='black')

      plt.title('Distribution of Bikes by Manufacturing Year')
      plt.xlabel('Manufacturing Year')
      plt.ylabel('Number of Bikes')
      plt.xticks(rotation=45)
      plt.grid(axis='y', linestyle='--', alpha=0.7)
      plt.tight_layout()
      plt.show()
```

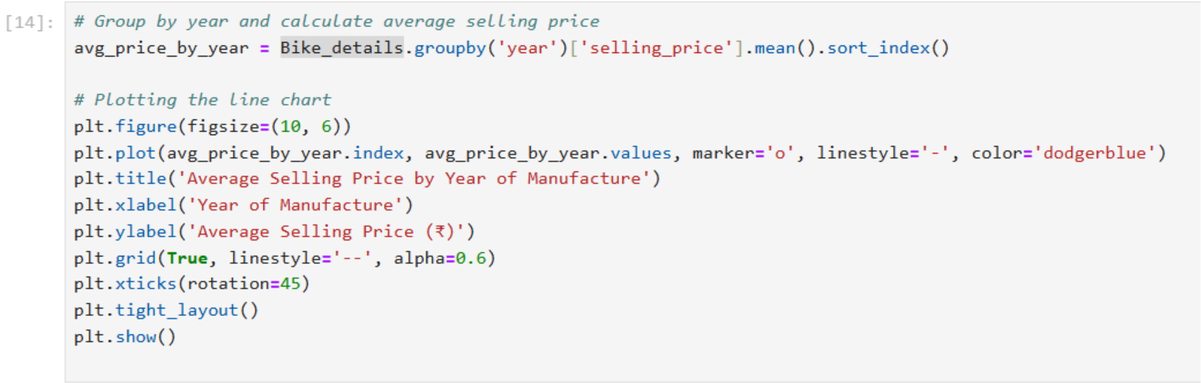
Output



4.Line Chart

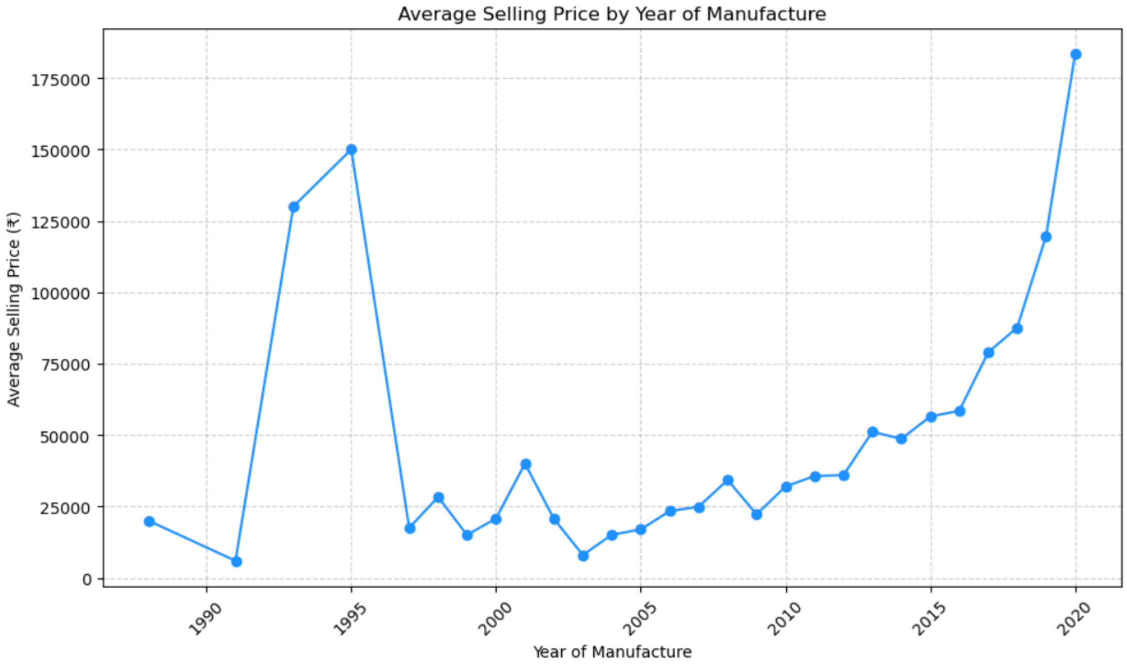
Average selling price by year of manufacture

Code



```
# Group by year and calculate average selling price
avg_price_by_year = Bike_details.groupby('year')['selling_price'].mean().sort_index()

# Plotting the line chart
plt.figure(figsize=(10, 6))
plt.plot(avg_price_by_year.index, avg_price_by_year.values, marker='o', linestyle='-', color='dodgerblue')
plt.title('Average Selling Price by Year of Manufacture')
plt.xlabel('Year of Manufacture')
plt.ylabel('Average Selling Price (₹)')
plt.grid(True, linestyle='--', alpha=0.6)
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()
```

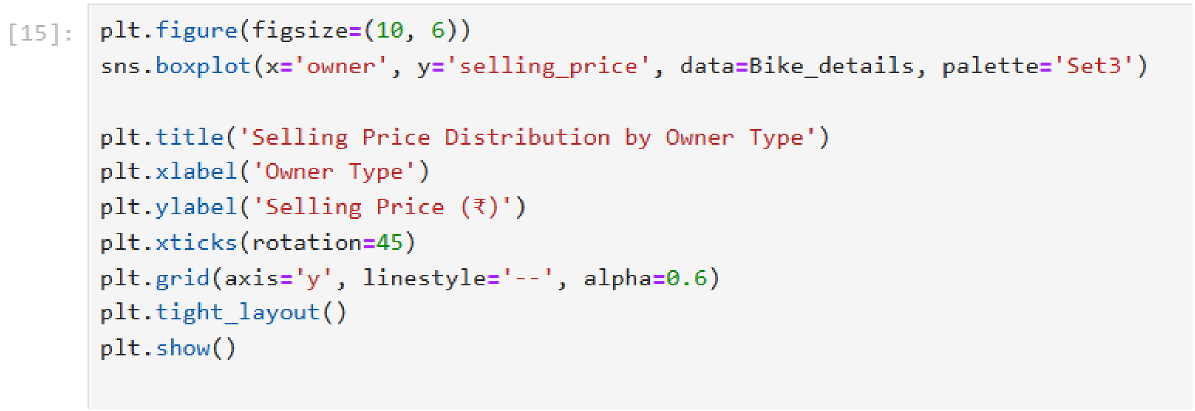
Output



5.Box Plot

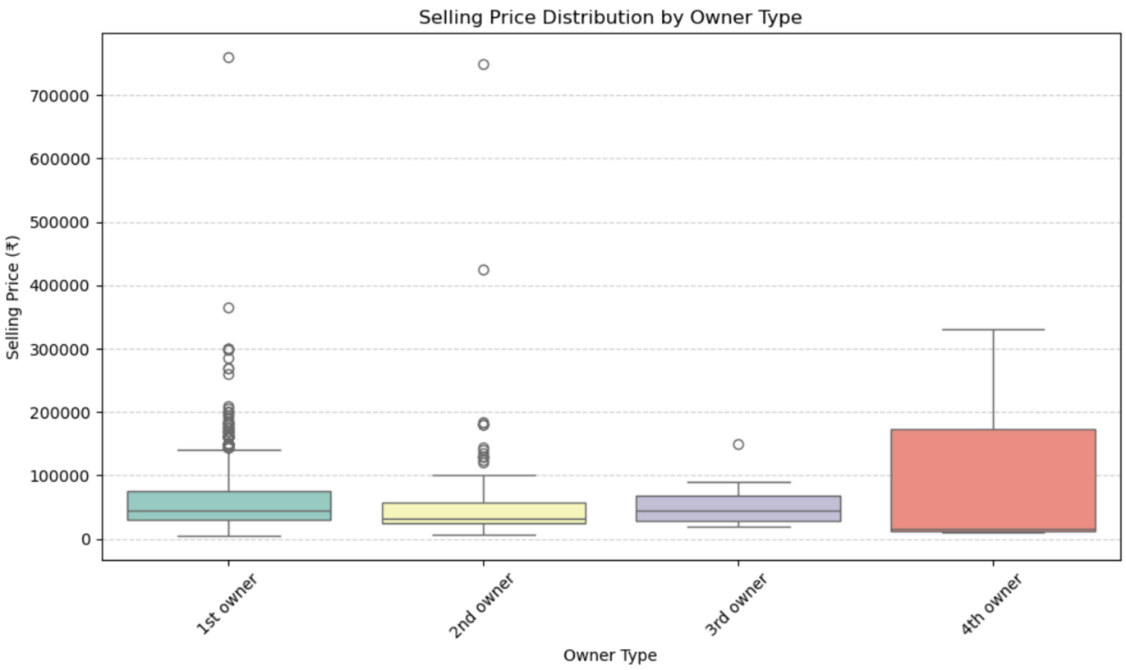
Selling price distribution by owner type

Code



```
[15]: plt.figure(figsize=(10, 6))
      sns.boxplot(x='owner', y='selling_price', data=Bike_details, palette='Set3')

      plt.title('Selling Price Distribution by Owner Type')
      plt.xlabel('Owner Type')
      plt.ylabel('Selling Price (₹)')
      plt.xticks(rotation=45)
      plt.grid(axis='y', linestyle='--', alpha=0.6)
      plt.tight_layout()
      plt.show()
```

Output



Conclusion

The analysis of the used bike dataset reveals several key insights into the resale bike market. A majority of the bikes fall into the price category below ₹100,000, indicating affordability and higher demand in that segment. Most bikes are sold by individual sellers, but dealers tend to list newer and sometimes higher-priced models. First-owner bikes command significantly higher resale values compared to second or third-owner bikes, highlighting the importance of ownership history. The data also shows that newer models (post-2010) dominate the listings, with a noticeable decline in older bikes, suggesting a preference for relatively modern vehicles. Fuel type and transmission play a role in pricing and popularity, with petrol and manual bikes being the most common. Visual comparisons using pie charts, bar graphs, and histograms provided a clear view of trends and consumer behavior. These insights can aid buyers, sellers, and dealers in making informed decisions in the used bike market.

Future Scope

The current analysis offers valuable insights into the used bike market, but there is ample scope for future enhancement. Incorporating machine learning techniques can enable predictive modeling to estimate resale prices based on features such as brand, age, kilometers driven, and ownership. Adding geographical data would allow for regional trend analysis, while time-series analysis could uncover seasonal variations in sales. Furthermore, integrating user reviews and ratings can offer qualitative insights into bike conditions and customer satisfaction. A comparative study between new and used bike markets could also highlight pricing gaps and depreciation rates. Expanding the dataset to include details like service history, insurance status, and accident records would provide a more comprehensive and accurate analysis. These improvements would enrich the overall understanding of market dynamics and support more informed decision-making for buyers, sellers, and dealers alike.