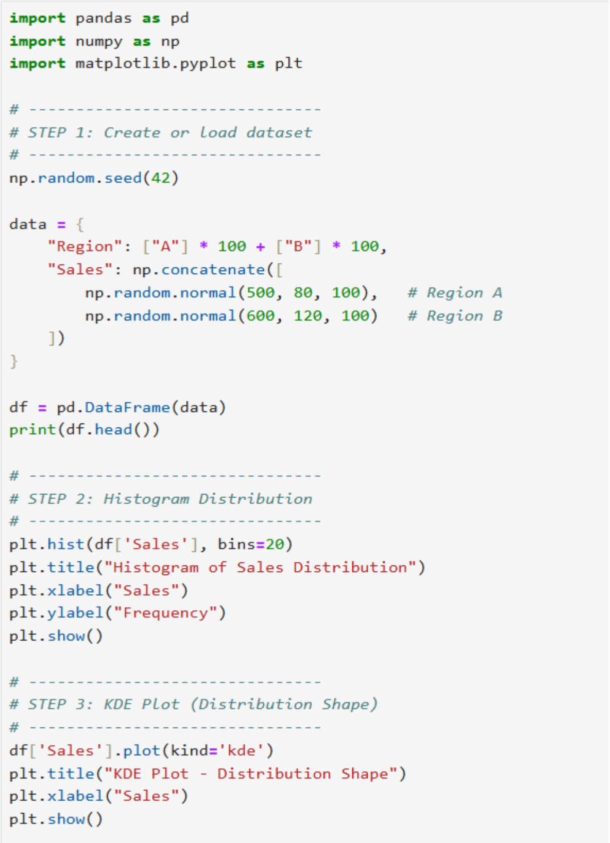
PROJECT -2 Statistical plots & distribution analysis

NAME: SUHEAL AHMAD

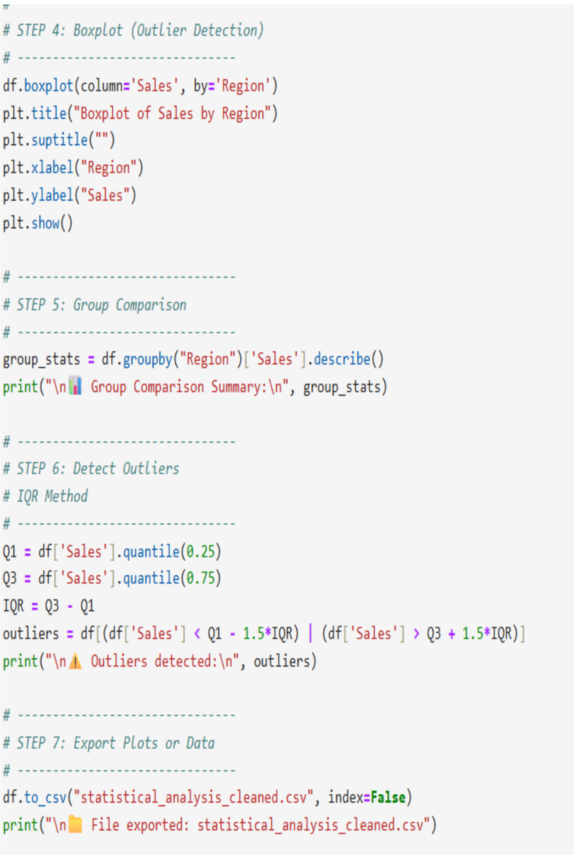
Objective:

To analyse data distributions using statistical plots like histograms, KDE, and boxplots, compare categories, detect outliers, and interpret the spread, skewness, and variation.

CODES:

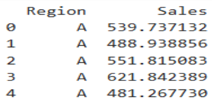


```python
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

# ------------------------------
# STEP 1: Create or load dataset
# ------------------------------
np.random.seed(42)

data = {
    "Region": ["A"] * 100 + ["B"] * 100,
    "Sales": np.concatenate([
        np.random.normal(500, 80, 100),   # Region A
        np.random.normal(600, 120, 100)   # Region B
    ])
}

df = pd.DataFrame(data)
print(df.head())

# ------------------------------
# STEP 2: Histogram Distribution
# ------------------------------
plt.hist(df['Sales'], bins=20)
plt.title("Histogram of Sales Distribution")
plt.xlabel("Sales")
plt.ylabel("Frequency")
plt.show()

# ------------------------------
# STEP 3: KDE Plot (Distribution Shape)
# ------------------------------
df['Sales'].plot(kind='kde')
plt.title("KDE Plot - Distribution Shape")
plt.xlabel("Sales")
plt.show()
```

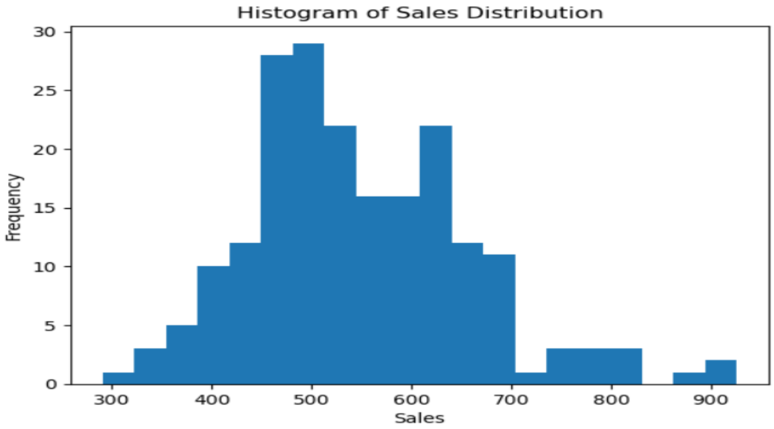


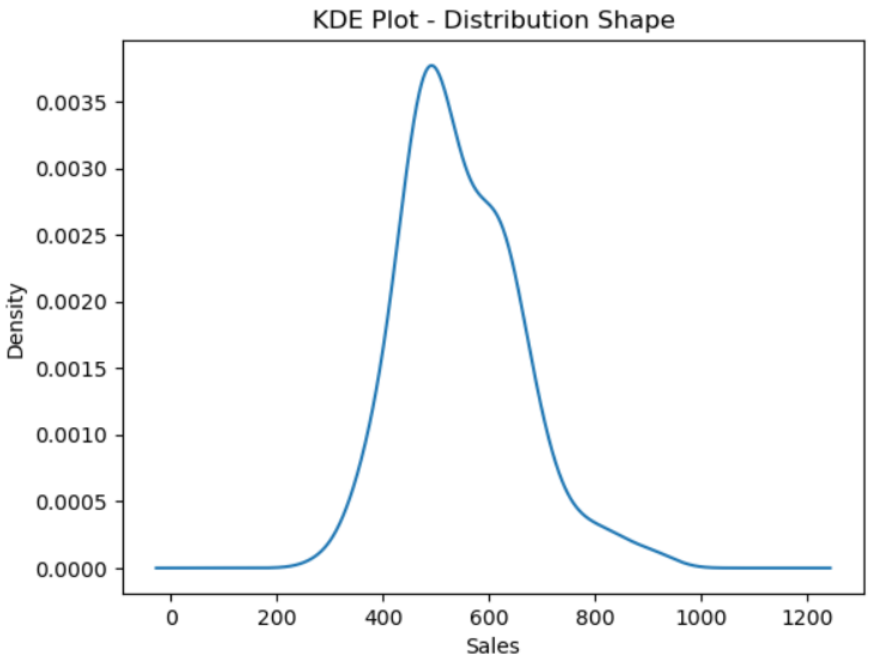
```python
#
# STEP 4: Boxplot (Outlier Detection)
# ------------------------------
df.boxplot(column='Sales', by='Region')
plt.title("Boxplot of Sales by Region")
plt.suptitle("")
plt.xlabel("Region")
plt.ylabel("Sales")
plt.show()

# ------------------------------
# STEP 5: Group Comparison
# ------------------------------
group_stats = df.groupby("Region")['Sales'].describe()
print("\n📊 Group Comparison Summary:\n", group_stats)

# ------------------------------
# STEP 6: Detect Outliers
# IQR Method
# ------------------------------
Q1 = df['Sales'].quantile(0.25)
Q3 = df['Sales'].quantile(0.75)
IQR = Q3 - Q1
outliers = df[(df['Sales'] < Q1 - 1.5*IQR) | (df['Sales'] > Q3 + 1.5*IQR)]
print("\n⚠ Outliers detected:\n", outliers)

# ------------------------------
# STEP 7: Export Plots or Data
# ------------------------------
df.to_csv("statistical_analysis_cleaned.csv", index=False)
print("\n📁 File exported: statistical_analysis_cleaned.csv")
```

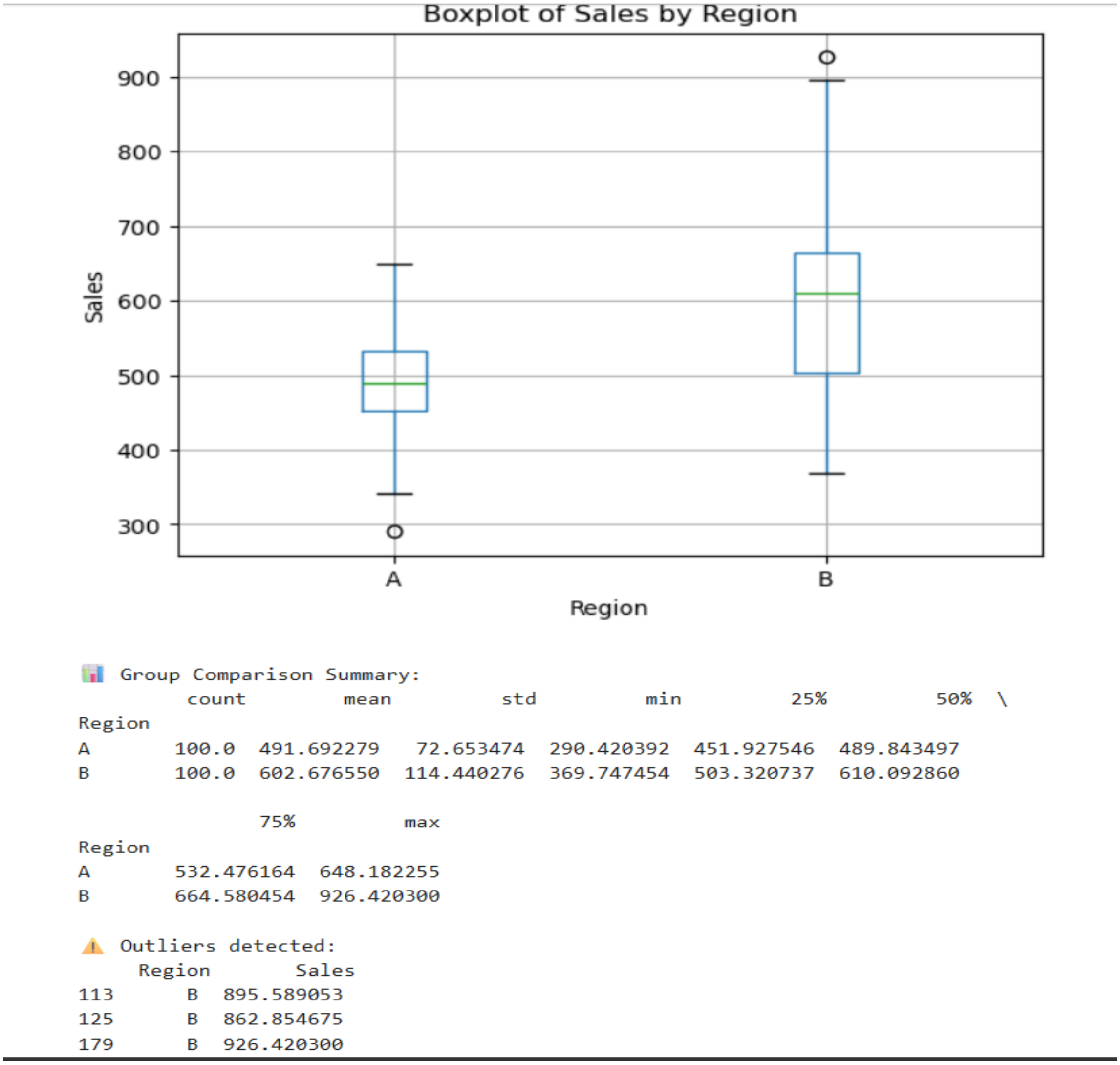
RESULTS:



```
  Region       Sales
0      A  539.737132
1      A  488.938856
2      A  551.815083
3      A  621.842389
4      A  481.267730
```







```
📊 Group Comparison Summary:
        count        mean         std         min         25%         50%  \
Region
A       100.0  491.692279   72.653474  290.420392  451.927546  489.843497
B       100.0  602.676550  114.440276  369.747454  503.320737  610.092860

               75%         max
Region
A       532.476164  648.182255
B       664.580454  926.420300

⚠️ Outliers detected:
    Region       Sales
113      B  895.589053
125      B  862.854675
179      B  926.420300
```

Result Interpretation:

- *The histogram and KDE plot show the distribution shape for sales.*
- *Region A has smaller spread and less variation.*
- *Region B shows higher sales but more spread (higher standard deviation).*
- *Boxplot indicates outliers and Region B is more skewed.*
- *The distribution for Region A is closer to normal, while Region B is more stretched.*

Conclusion:

The statistical distribution analysis provided valuable insights into the structure, spread, and behaviour of the dataset. By visualizing the data with histograms, KDE plots, and boxplots, it became easier to understand how each variable was distributed and whether the data leaned towards a symmetrical, skewed, or multimodal pattern. The histogram and KDE visualizations clearly highlighted where most data points were concentrated, helping identify dominant trends within the dataset, while boxplots helped identify outliers, variability, and the median range of each numerical feature.

A comparison of distributions between groups (e.g., Region A vs Region B) helped in identifying performance gaps, differences in central tendencies, and overall spread. This step revealed whether groups were similar, significantly different, or influenced by external factors. Outlier detection further refined the findings by identifying extreme values that could distort the results if not handled properly. Observing skewness helped determine whether data transformation or normalization might be needed before moving into deeper analysis or predictive modelling.

Overall, this project demonstrates how statistical plots play a crucial role in data understanding before any advanced analysis or machine learning is performed. The visual exploration provided clarity on data quality, distribution shape, consistency, and group-level comparison. These observations form a strong foundation for future decisions such as feature engineering, model selection, or business-level insights. In conclusion, the project successfully highlights the importance of visualization in discovering patterns, spotting anomalies, and making data-driven decisions with confidence.