Day 31

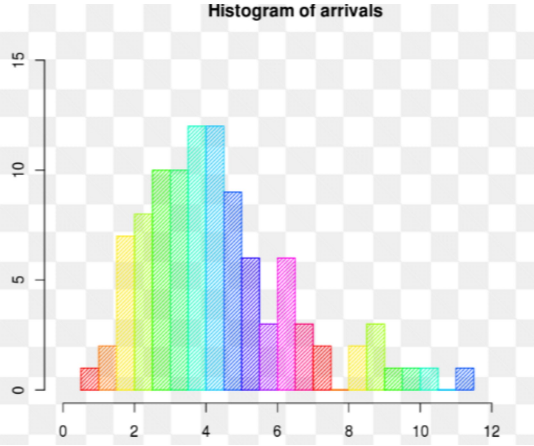
Exploratory Data Analysis (EDA):

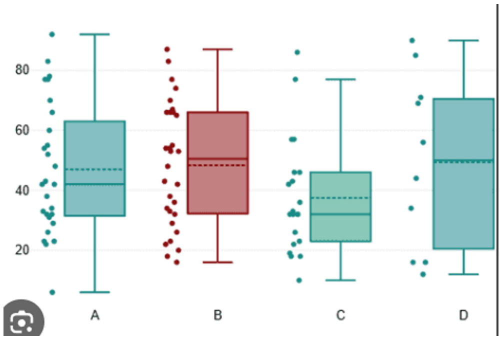
It involves analyzing and visualizing data to understand its key characteristics, uncover patterns, and identify relationships between variables refers to the method of studying and exploring record sets to apprehend their predominant traits, discover patterns, locate outliers, and identify relationships between variables.

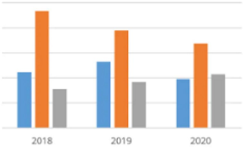
Types of EDA:

1. **Univariate Analysis**:
   Univariate analysis focuses on a single variable to understand its internal structure. It is primarily concerned with describing the data and finding patterns existing in a single feature. Some common techniques:

- **Histograms**: Used to visualize the distribution of a variable.



- **Box plots**: Useful for detecting outliers and understanding the spread and skewness of the data.
- 

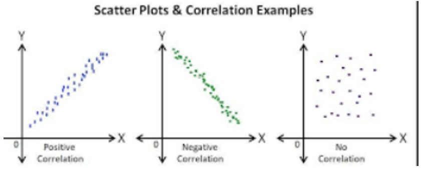
- **Bar charts**: Employed for categorical data to show the frequency of each category.
- 

- **Summary statistics**: Calculations like mean, median, mode, variance, and standard deviation that describe the central tendency and dispersion of the data.

2. Bivariate Analysis

Bivariate evaluation involves exploring the connection between variables. It enables find associations, correlations, and dependencies between pairs of variables. Bivariate analysis is a crucial form of exploratory data analysis that examines the relationship between two variables. Some key techniques used in bivariate analysis:

- **Scatter Plots:** These are one of the most common tools used in bivariate analysis. A scatter plot helps visualize the relationship between two continuous variables.



- **Correlation Coefficient**: This statistical measure (often Pearson's correlation coefficient for linear relationships) quantifies the degree to which two variables are related.
- **Cross-tabulation**: Also known as contingency tables, cross-tabulation is used to analyze the relationship between two categorical variables. It shows the frequency distribution of categories of one variable in rows and the other in columns, which helps in understanding the relationship between the two variables.
- **Line Graphs**: In the context of time series data, line graphs can be used to compare two variables over time. This helps in identifying trends, cycles, or patterns that emerge in the interaction of the variables over the specified period.

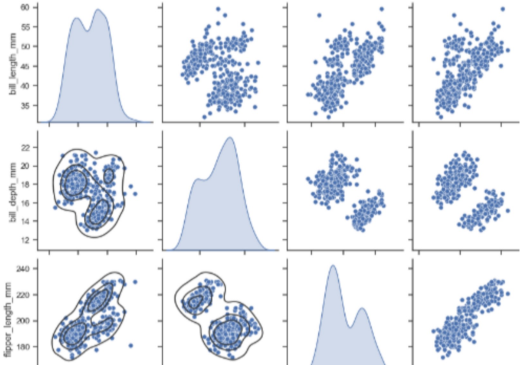
90
80
70
60
50
40
30
20
10
0

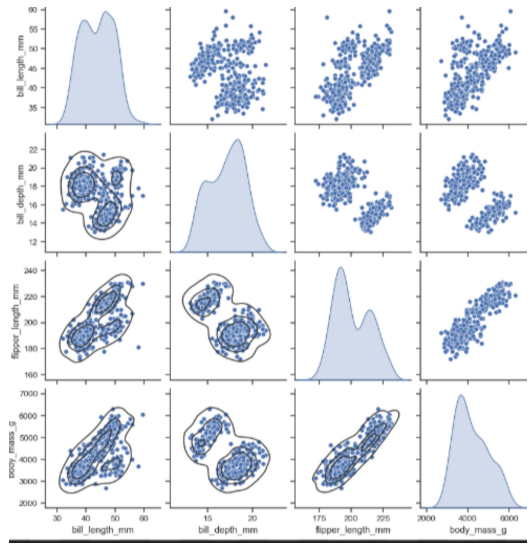
- **Covariance**: Covariance is a measure used to determine how much two random variables change together. However, it is sensitive to the scale of the variables, so it's often supplemented by the correlation coefficient for a more standardized assessment of the relationship.

3. Multivariate Analysis

Multivariate analysis examines the relationships between two or more variables in the dataset. It aims to understand how variables interact with one another, which is crucial for most statistical modeling techniques. Techniques include:

- **Pair plots**: Visualize relationships across several variables simultaneously to capture a comprehensive view of potential interactions.



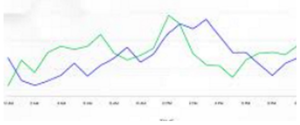


- **Principal Component Analysis (PCA)**: A dimensionality reduction technique used to reduce the dimensionality of large datasets, while preserving as much variance as possible.

Specialized EDA Techniques

- **Spaial Analysis**: For geographical data, using maps and spatial plotting to understand the geographical distribution of variables.

- **Text Analysis**: Involves techniques like word clouds, frequency distributions, and sentiment analysis to explore text data.
- **Time Series Analysis:** This type of analysis is mainly applied to statistics sets that have a temporal component. Time collection evaluation entails inspecting and modeling styles, traits, and seasonality inside the statistics through the years. Techniques like line plots, autocorrelation analysis, transferring averages, and ARIMA (AutoRegressive Integrated Moving Average) fashions are generally utilized in time series analysis.



Python Libraries

- **Pandas**: Provides extensive functions for data manipulation and analysis, including data structure handling and time series functionality.
- **Matplotlib**: A plotting library for creating static, interactive, and animated visualizations in Python.
- **Seaborn**: Built on top of Matplotlib, it provides a high-level interface for drawing attractive and informative statistical graphics.
- **Plotly**: An interactive graphing library for making interactive plots and offers more sophisticated visualization capabilities.