1. ***Importing Libraries***

**import pandas as pd**: Imports the Pandas library, which is used for data manipulation and analysis. The alias pd is a common convention.

**import numpy as np**: Imports the NumPy library, which provides support for large, multi-dimensional arrays and matrices, along with a collection of mathematical functions to operate on these arrays. The alias np is also a common convention.

**from sklearn.model\_selection import train\_test\_split**: Imports the train\_test\_split function from Scikit-Learn (sklearn), which is a machine learning library. This function is used to split your dataset into training and testing sets.

**from sklearn.preprocessing import StandardScaler**: Imports the StandardScaler class from Scikit-Learn, which is used to standardize features by removing the mean and scaling to unit variance.

**from sklearn.linear\_model import LogisticRegression**: Imports the LogisticRegression class from Scikit-Learn, which is a simple linear model used for binary classification.

**from sklearn.metrics import accuracy\_score, confusion\_matrix, classification\_report**: Imports functions for evaluating the performance of your machine learning model.

accuracy\_score: Calculates the accuracy of the model.

confusion\_matrix: Creates a matrix that shows the counts of true positives, true negatives, false positives, and false negatives.

classification\_report: Provides a detailed summary of the model’s precision, recall, F1-score, and support.

**import matplotlib.pyplot as plt**: Imports Matplotlib’s plotting functions with the alias plt. Matplotlib is a plotting library used for creating static, interactive, and animated visualizations in Python.

**import seaborn as sns**: Imports Seaborn, a statistical data visualization library based on Matplotlib. It provides a high-level interface for drawing attractive and informative statistical graphics.

1. ***Loading the Dataset***

**url = "https://raw.githubusercontent.com/mwaskom/seaborn-data/master/titanic.csv"**: Defines the URL of the Titanic dataset, which is hosted on GitHub.

**df = pd.read\_csv(url)**: Loads the dataset from the URL into a Pandas DataFrame called df. A DataFrame is a 2-dimensional, labeled data structure similar to a table in a database or an Excel spreadsheet.

1. ***Displaying the First Few Rows of the Data***

**print("Original DataFrame:")**: Prints a title to describe the output that follows.

**print(df.\_\_\_\_)** Prints the first 5 rows of the DataFrame. This is a quick way to get an overview of what the dataset looks like.

1. ***Handling Missing Values***

**df['\_\_\_\_'].fillna(df['\_\_\_\_'].median(), inplace=True)**: Fills missing values in the age column with the median \_\_\_\_. fillna() is a Pandas method that replaces missing data (NaN values). The median() function calculates the middle value of the column. inplace=True means that the changes are made directly to the original DataFrame.

**df['\_\_\_\_\_'].fillna(df['\_\_\_\_\_'].mode()[0], inplace=True)**: Fills missing values in the \_\_\_\_\_\_ column with the most common value (mode). mode() returns the most frequent value, and [0] selects the first mode in case of multiple modes. Again, inplace=True modifies the DataFrame directly.

1. ***Dropping Irrelevant Columns***

**df.drop(columns=[\_\_\_, \_\_\_\_, \_\_\_\_, \_\_\_\_,\_\_\_\_\_\_,\_\_\_\_\_\_], inplace=True)**: Drops the specified columns from the DataFrame because they are not relevant for the analysis or modeling. drop() is a Pandas method used to remove rows or columns from a DataFrame. inplace=True applies the change directly to the DataFrame

1. ***One-Hot Encoding of Categorical Variables***

**df = pd.get\_dummies(df, columns=['\_\_\_\_', '\_\_\_\_'], drop\_first=True)**: Converts categorical variables (\_\_\_\_ and \_\_\_\_\_) into numerical format using **one-hot encoding**. get\_dummies() creates binary columns (0 or 1) for each category. The drop\_first=True parameter avoids *multicollinearity* by dropping the first category in each column.

1. ***Splitting the Data into Features and Target***

Scikit-learn

>>>

1. ***Splitting the Data into Training and Testing Sets***

Scikit-learn

>>>

***Feature Scaling***

scaler = StandardScaler(): Initializes a StandardScaler object, which standardizes features by removing the mean and scaling to unit variance.

>>>

***Making Predictions on the Test Data***

>>>

***Evaluating the Model***

Accuracy ---

Conf\_matrix ----

Class\_report ----

>>>