from google.colab import files

# Upload the Titanic CSV file

uploaded = files.upload()

# Import Libraries

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

import seaborn as sns

# Load dataset

df = pd.read\_csv('Titanic-Dataset.csv')

print("First few rows of the dataset:")

print(df.head())

# Basic info and missing values

import pandas as pd

# Load your data into df

file\_path = '/content/drive/My Drive/Titanic-Dataset.csv'

df = pd.read\_csv(file\_path)

print("\nBasic info:")

print(df.info())

print("\nMissing values per column:")

print(df.isnull().sum())

print("\nColumns after cleaning names:")

print(df.columns.tolist())

# Generate Summary Statistics

print("\nSummary Statistics:")

print(df.describe(include='all')) # Include all columns

from google.colab import drive

import os

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

import numpy as np

# Mount Google Drive

drive.mount('/content/drive')

# Load your data

file\_path = '/content/drive/My Drive/Titanic-Dataset.csv' # Update with actual path

df = pd.read\_csv(file\_path)

# Create directory to save plots

save\_dir = '/content/drive/My Drive/titanic\_2'

os.makedirs(save\_dir, exist\_ok=True)

# Select numeric columns

numeric\_cols = df.select\_dtypes(include=np.number).columns

# Plot and save each numeric column

for col in numeric\_cols:

plt.figure(figsize=(8, 4))

sns.histplot(df[col].dropna(), kde=True, bins=30)

plt.title(f'Distribution of {col}')

plt.xlabel(col)

plt.ylabel('Frequency')

# Save and show plot

file\_path = os.path.join(save\_dir, f'{col}\_distribution.png')

plt.savefig(file\_path, bbox\_inches='tight')

plt.show()

plt.close()

# Boxplots to Detect Outliers

#file\_path = '/content/drive/My Drive/Titanic-Dataset.csv'

#save\_dir = '/content/drive/My Drive/titanic\_2/Boxplots'

for col in numeric\_cols:

plt.figure(figsize=(8, 4))

sns.boxplot(x=df[col])

plt.title(f'Boxplot of {col}')

plt.show()

file\_path = os.path.join(save\_dir, f'{col}\_Boxplot.png')

plt.savefig(file\_path, bbox\_inches='tight')

plt.show()

plt.close()

#Correlation Matrix and Heatmap

corr = df[numeric\_cols].corr()

plt.figure(figsize=(10, 8))

sns.heatmap(corr, annot=True, cmap='coolwarm', fmt='.2f')

plt.title('Correlation Heatmap')

plt.show()

file\_path = os.path.join(save\_dir, f'{col}\_CorreleationHeatmap.png')

plt.savefig(file\_path, bbox\_inches='tight')

plt.show()

plt.close()

# Pairplot for Feature Relationships

sns.pairplot(df[numeric\_cols])

plt.suptitle('Pairplot of Numerical Features', y=1.02)

plt.show()

file\_path = os.path.join(save\_dir, f'{col}\_Pairplot.png')

plt.savefig(file\_path, bbox\_inches='tight')

plt.show()

plt.close()

if 'Survived' in df.columns:

plt.figure(figsize=(8, 5))

sns.countplot(x='Survived', data=df)

plt.title('Survival Count (0 = Died, 1 = Survived)')

plt.xlabel('Survived')

plt.ylabel('Count')

plt.show()

file\_path = os.path.join(save\_dir, f'{col}\_Survived.png')

plt.savefig(file\_path, bbox\_inches='tight')

plt.show()

plt.close()

# Survival Rate by Categorical Variables (if present)

categorical\_cols = df.select\_dtypes(include='object').columns

for col in categorical\_cols:

if 'Survived' in df.columns:

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

sns.barplot(x=col, y='Survived', data=df, ci=None)

plt.title(f'Survival Rate by {col}')

plt.xticks(rotation=45)

plt.show()

file\_path = os.path.join(save\_dir, f'{col}\_SurvivalRate.png')

plt.savefig(file\_path, bbox\_inches='tight')

plt.show()

plt.close()