IBM Data Science Capstone - SpaceX Launch Analysis

Introduction

This project focuses on analyzing SpaceX's Falcon 9 rocket launches. The goal is to predict whether the first stage of Falcon 9 will successfully land after launch, using data science techniques including data wrangling, exploratory data analysis (EDA), and machine learning models.

Data Collection

Data was collected from two sources:

1. SpaceX API - Information about all the launches such as the rocket used, payload, launch site, and success of the mission.
2. Web Scraping - Additional data about SpaceX launches was scraped from Wikipedia using BeautifulSoup.

Code for Data Collection:

```
import requests
import pandas as pd

# Fetch SpaceX launches data from API
url = "https://api.spacexdata.com/v4/launches"
response = requests.get(url)
data = response.json()

# Convert to DataFrame
```

```
df = pd.json_normalize(data)

df.head()
```

Data Wrangling

Data wrangling included handling missing values and extracting useful features from nested JSON fields.

For example, we extracted payload mass and rocket type from the API data to create features for machine learning.

Code for Data Wrangling:

```
# Handle missing values in 'payloads' and extract mass

df['payload_mass_kg'] = df['payloads'].apply(lambda x: x[0]['mass_kg'] if isinstance(x, list) and len(x) > 0 else None)


# Extract rocket type

df['rocket_type'] = df['rocket'].apply(lambda x: x['type'] if x is not None else None)
```

Exploratory Data Analysis (EDA)

EDA was performed to analyze the distribution of payloads, launch outcomes, and trends over time. Key findings

included that lighter payloads had a higher success rate, and certain launch sites had a higher number of successful

landings.

Code for EDA:

```
import matplotlib.pyplot as plt

import seaborn as sns
```

```
# Plot successful landings by launch site
plt.figure(figsize=(10, 6))
sns.countplot(x='launchpad', hue='success', data=df)
plt.title('Success by Launch Site')
plt.xticks(rotation=45)
plt.show()
```

Machine Learning Models

Several machine learning models were trained to predict the success of the landing. Models used included:

1. Logistic Regression
2. K-Nearest Neighbors (KNN)
3. Support Vector Machine (SVM)

Accuracy was used to evaluate the models, and Logistic Regression and SVM performed best.

Code for Logistic Regression Model:

```
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score

# Define features and target
X = df[['flight_number', 'payload_mass_kg', 'rocket_type']]
y = df['success']
```

```
# Train-test split

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)



# Train the Logistic Regression model

model = LogisticRegression()

model.fit(X_train, y_train)



# Make predictions and calculate accuracy

y_pred = model.predict(X_test)

accuracy = accuracy_score(y_test, y_pred)

print(f'Accuracy: {accuracy:.2f}')
```