Predicting Falcon 9 First Stage Landing Success

A Data Science Capstone Project.
(In fulfilment of IBM Data Science Professional Certificate)

Presented By:

Akinola Aanu Olufemi

Data Scientist

aanu.akinolaa@gmail.com

Project Overview

This presentation is structured to provide a comprehensive overview of the capstone project on predicting Falcon 9 first stage landing success using various data science methodologies. The detailed content, along with visual illustrations, will ensure a clear understanding of the project's objectives, methodologies, and outcomes. The inclusion of code snippets and interactive elements will engage the management audience and showcase the practical implementation of data science techniques.

OBJECTIVE:

The primary objective of the SpaceX Launch Records Dashboard project is to develop a comprehensive and interactive data visualization tool that enables users to explore and analyze SpaceX launch records. The dashboard is designed to provide insightful and actionable information related to launch success rates, payload masses, and booster versions

Expected Outcome:

The successful completion of this project will result in a visually appealing and interactive SpaceX Launch Records Dashboard. Users will be able to gain valuable insights into launch success patterns, explore the impact of payload mass on launches, and understand the distribution of success across different launch sites. The project aims to contribute to informed decision-making and analysis within the space exploration community, catering to both enthusiasts and professionals alike.

Table of Content

Data Collection and Data Wrangling

Data collection involves obtaining the Spacex lauch data from kaggle, and data wrangling focuses on cleaning and preprocessing the data by handling missing values and performing necessary feature engineering. The data for this project is collected from from Wikipedia hosting falcon9 launch details

Output of the code generated the data collected in next slide

Check complete code in GITHUB

```
import pandas as pd

spacex_df = pd.read_csv("spacex_launch_dash.csv")

spacex_df
```

In [4]:
```
import pandas as pd
spacex_df = pd.read_csv("spacex_launch_dash.csv")
spacex_df
```

Out[4]:

	Unnamed: 0	Flight Number	Launch Site	class	Payload Mass (kg)	Booster Version	Booster Version Category
0	0	1	CCAFS LC-40	0	0.00	F9 v1.0 B0003	v1.0
1	1	2	CCAFS LC-40	0	0.00	F9 v1.0 B0004	v1.0
2	2	3	CCAFS LC-40	0	525.00	F9 v1.0 B0005	v1.0
3	3	4	CCAFS LC-40	0	500.00	F9 v1.0 B0006	v1.0
4	4	5	CCAFS LC-40	0	677.00	F9 v1.0 B0007	v1.0
5	5	7	CCAFS LC-40	0	3170.00	F9 v1.1	v1.1
6	6	8	CCAFS LC-40	0	3325.00	F9 v1.1	v1.1
7	7	9	CCAFS LC-40	0	2296.00	F9 v1.1	v1.1
8	8	10	CCAFS LC-40	0	1316.00	F9 v1.1	v1.1
9	9	11	CCAFS LC-40	0	4535.00	F9 v1.1	v1.1
10	10	12	CCAFS LC-40	0	4428.00	F9 v1.1 B1011	v1.1
11	11	13	CCAFS LC-40	0	2216.00	F9 v1.1 B1010	v1.1
12	12	14	CCAFS LC-40	0	2395.00	F9 v1.1 B1012	v1.1

```python
# Save the cleaned data to a new CSV file
cleaned_file_path = "cleaned_data.csv"
df.to_csv(cleaned_file_path, index=False)
print(f"\nCleaned data saved to {cleaned_file_path}")
```

```
Initial Data Overview:
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 56 entries, 0 to 55
Data columns (total 7 columns):
 #   Column                    Non-Null Count  Dtype
---  ------                    --------------  -----
 0   Unnamed: 0                56 non-null     int64
 1   Flight Number             56 non-null     int64
 2   Launch Site               56 non-null     object
 3   class                     56 non-null     int64
 4   Payload Mass (kg)         56 non-null     float64
 5   Booster Version           56 non-null     object
 6   Booster Version Category  56 non-null     object
dtypes: float64(1), int64(3), object(3)
memory usage: 3.2+ KB
None

Cleaned and Transformed Data:
   Flight Number  Launch Site  class  Payload Mass (kg)  Booster Version  \
0              1            0      0                0.0               36
1              2            0      0                0.0               37
2              3            0      0              525.0               38
3              4            0      0              500.0               39
4              5            0      0              677.0               40

   Booster Version Category
0                         3
1                         3
2                         3
3                         3
4                         3

Cleaned data saved to cleaned_data.csv
```

Exploratory Data Analysis (EDA)

Exploratory Data Analysis (EDA) involves analyzing and visualizing the main characteristics of a dataset such as Matplotlib, and Seaborn. our sample here is based on using Pandas

Interactive Visual Analytics

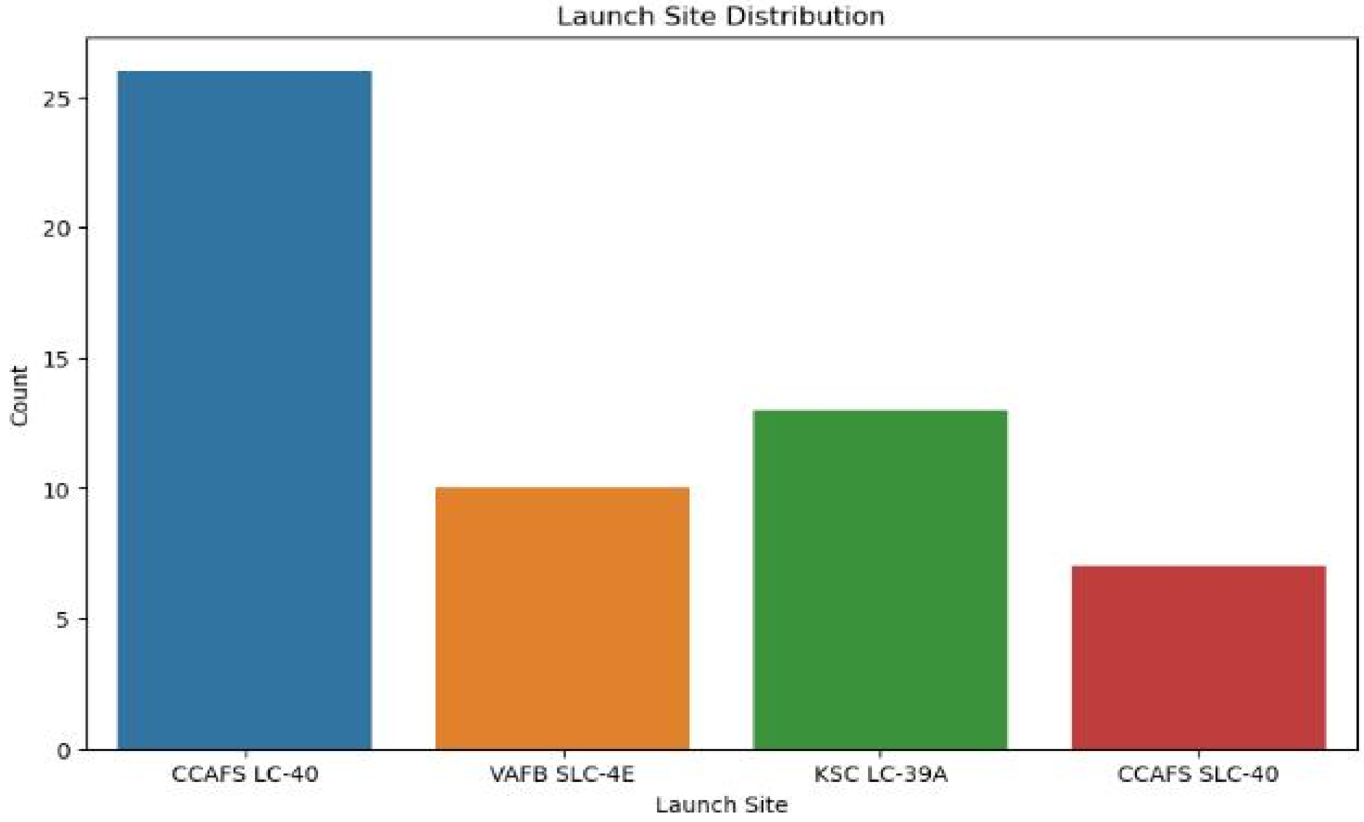
Define a layout with dropdowns, sliders, and graphs for interactive visual analytic s

Code for Visualization/Bar Chat

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
spacex_df = pd.read_csv("spacex_launch_dash.csv")
plt.figure(figsize=(10, 6))
sns.countplot(x="Launch Site", data=spacex_df)
plt.title("Launch Site Distribution")
plt.xlabel("Launch Site")
plt.ylabel("Count")
plt.show()
```

In [15]:

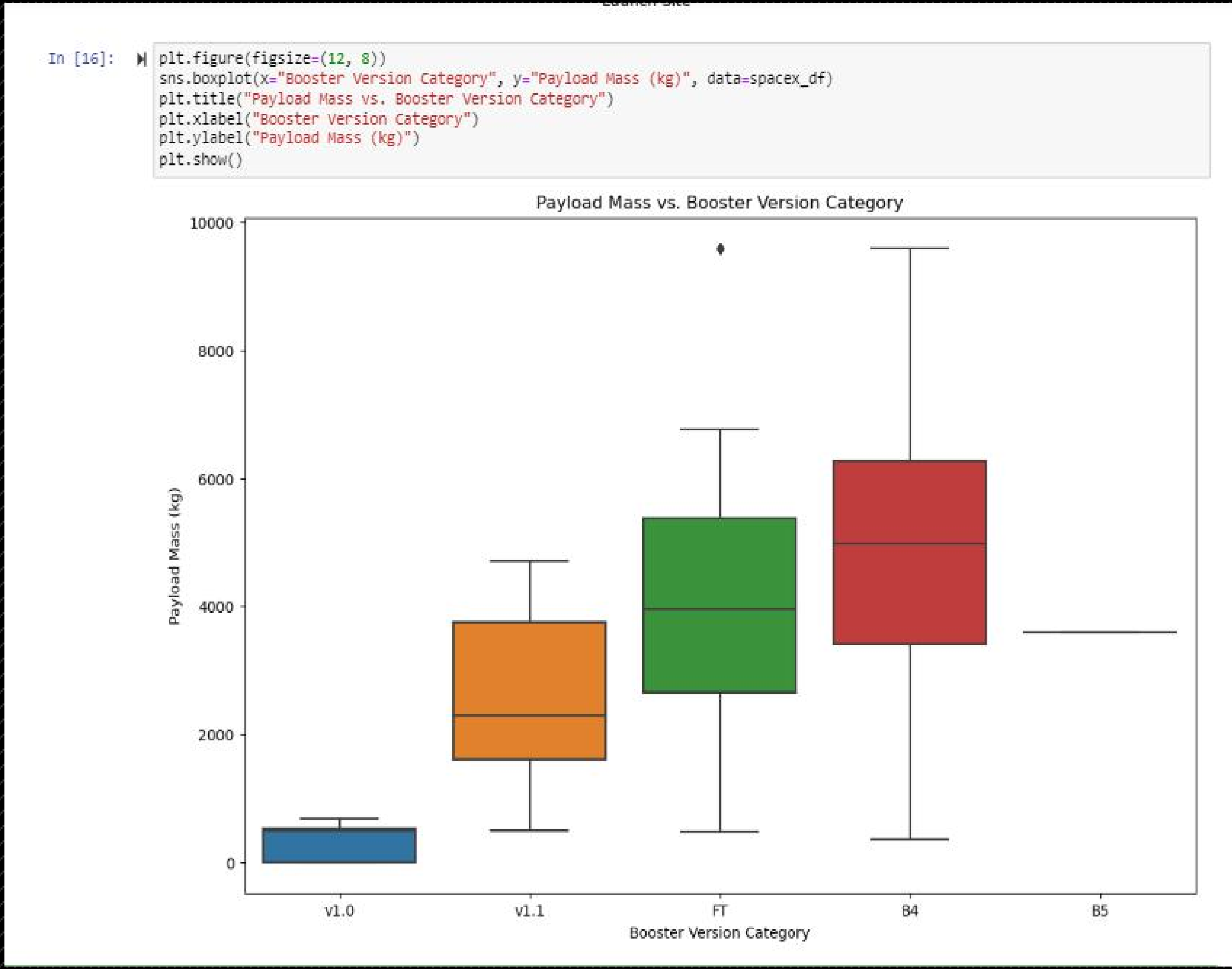
```python
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
spacex_df = pd.read_csv("spacex_launch_dash.csv")
plt.figure(figsize=(10, 6))
sns.countplot(x="Launch Site", data=spacex_df)
plt.title("Launch Site Distribution")
plt.xlabel("Launch Site")
plt.ylabel("Count")
plt.show()
```



In [ ]:

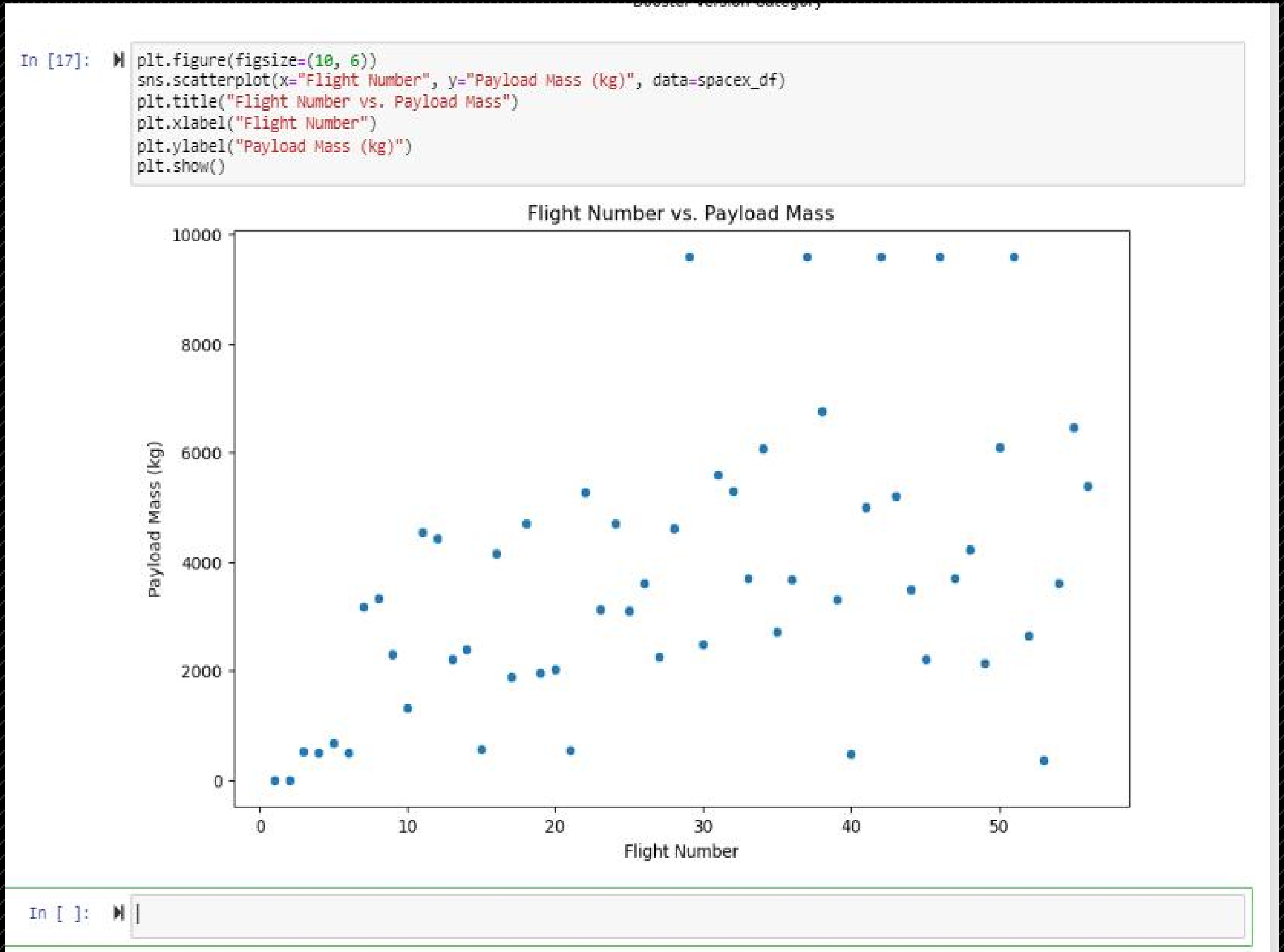
Code for Visualization/Boxplot

```
plt.figure(figsize=(12, 8))

sns.boxplot(x=″Booster Version
Category″, y=″Payload Mass (kg)″,
data=spacex_df)

plt.title(″Payload Mass vs. Booster
Version Category″)

plt.xlabel(″Booster Version
Category″)

plt.ylabel(″Payload Mass (kg)″)

plt.show()
```



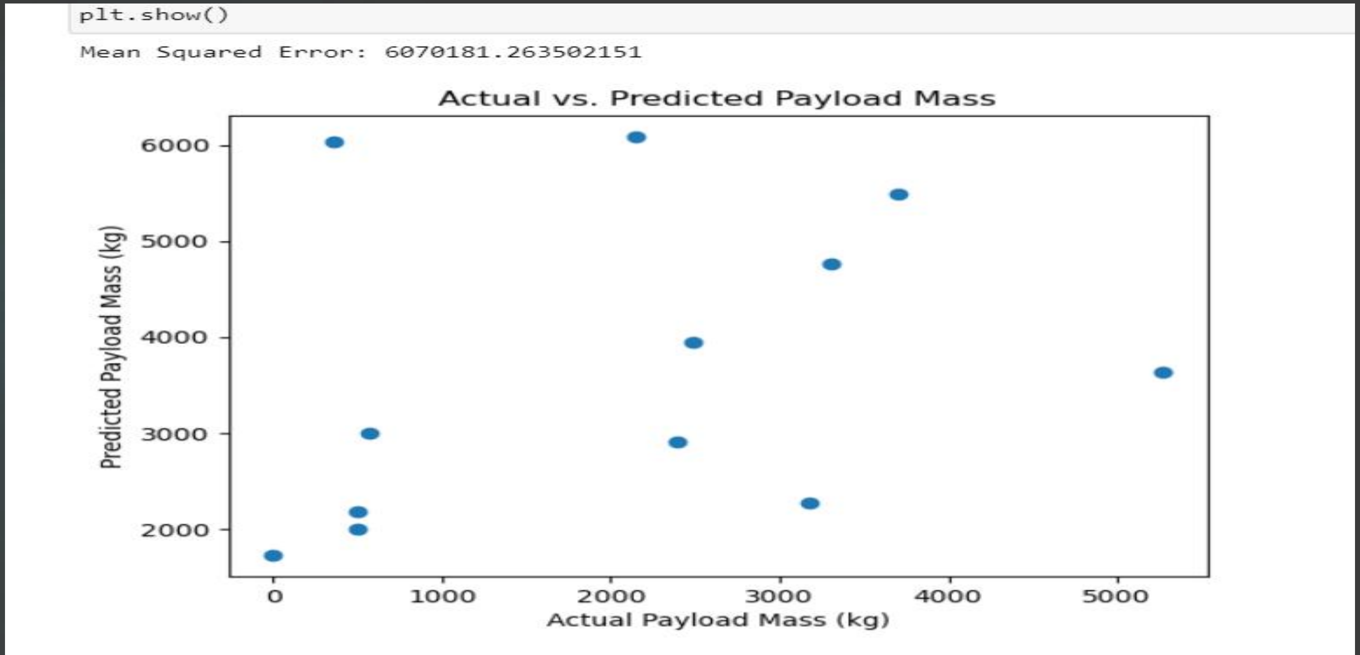
Code for Visualization/scatterplot

```
plt.figure(figsize=(10, 6))
sns.scatterplot(x="Flight Number", y="Payload Mass (kg)", data=spacex_df)
plt.title("Flight Number vs. Payload Mass")
plt.xlabel("Flight Number")
plt.ylabel("Payload Mass (kg)")
plt.show()
```



Predictive Analysis Methodology

Predictive analysis involves predicting future trends or outcomes based on historical data. For this example, let's consider a simple predictive analysis using linear regression to predict the Payload Mass based on other variables in the dataset. the code is hosted on GITHUB due to the line volume.

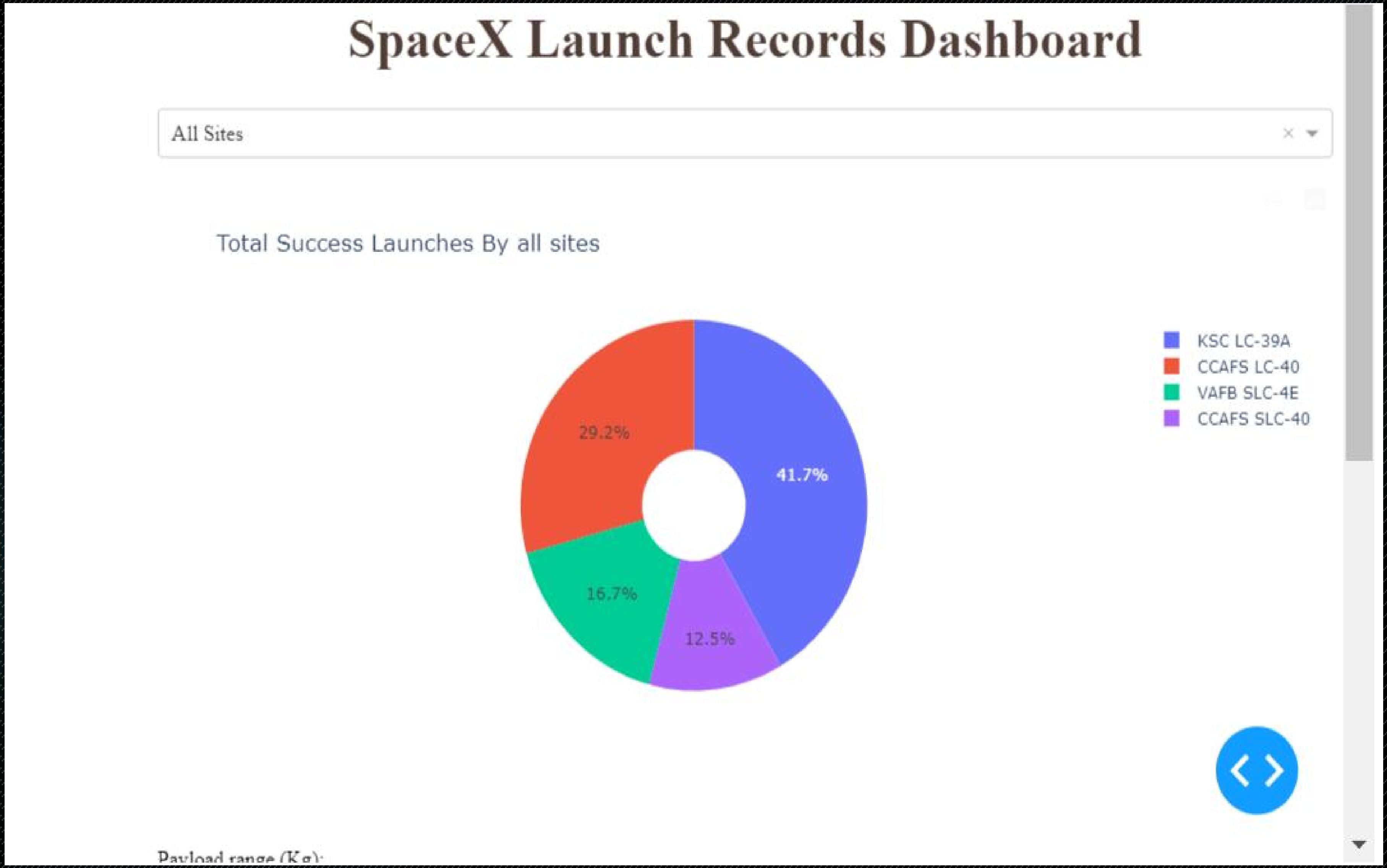


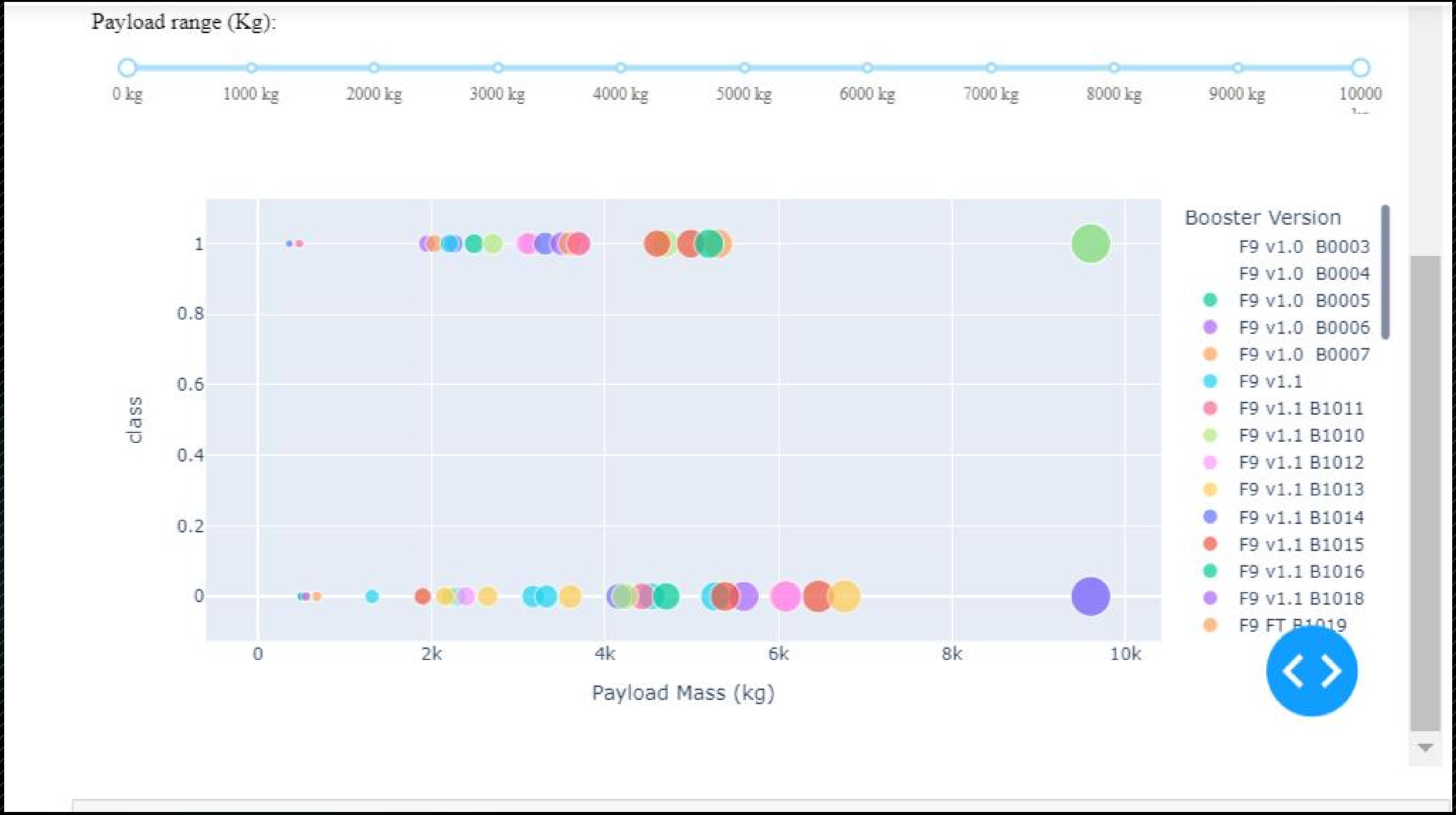
Web Scraping

Web scraping to get SpaceX launch data from Wikipedia

https://en.wikipedia.org/wiki/List_of_Falcon_9_and_Falcon_Heavy_launches

Web scraping is employed to gather additional data on Falcon 9 launches. The code demonstrates the extraction of launch schedule information from a website using BeautifulSoup and requests.


SpaceX Launch Records Dashboard
All Sites
Total Success Launches By all sites
29.2%
41.7%
16.7%
12.5%
KSC LC-39A
CCAFS LC-40
VAFB SLC-4E
CCAFS SLC-40
Payload range (Kg):


Payload range (Kg):
0 kg
1000 kg
2000 kg
3000 kg
4000 kg
5000 kg
6000 kg
7000 kg
8000 kg
9000 kg
10000
class
1
0.8
0.6
0.4
0.2
0
0
2k
4k
6k
8k
10k
Payload Mass (kg)
Booster Version
F9 v1.0 B0003
F9 v1.0 B0004
F9 v1.0 B0005
F9 v1.0 B0006
F9 v1.0 B0007
F9 v1.1
F9 v1.1 B1011
F9 v1.1 B1010
F9 v1.1 B1012
F9 v1.1 B1013
F9 v1.1 B1014
F9 v1.1 B1015
F9 v1.1 B1016
F9 v1.1 B1018

Executive Summary

The SpaceX Launch Records Dashboard provides a user-friendly interface for exploring and analyzing SpaceX launch data. It empowers users with the ability to make data-driven decisions, understand success patterns, and explore the impact of payload mass on launch outcomes. This tool is a valuable asset for SpaceX enthusiasts, analysts, and decision-makers in the space exploration community.