CAPSTONE PROJECT IBM DATA SCIENCE

Mohammad Varrel Bramasta

EXECUTIVE SUMMARY METHODOLOGIES

01 Data collection

02 Data wrangling

03 Exploratory Data Analysis with Data Visualization

04 Exploratory Data Analysis with SQL

05 Building an interactive map with Folium

06 Building a Dashboard with Plotly Dash

07 Predictive analysis (Classification)

Introduction

SpaceX has achieved great success in the commercial space industry by making space travel more affordable. One of its major achievements is the development of the Falcon 9 rocket, which is listed at $62 million on its website. This is significantly lower compared to the launch costs of other providers which can reach $165 million. This significant saving is mainly due to SpaceX's ability to reuse the first stage of its rocket. By understanding and predicting whether the Falcon 9 first stage will land successfully, we can plan the launch costs. Therefore, by utilizing the available information and machine learning techniques, we can predict the likelihood of SpaceX's first stage being reused.

Data Collection

IBM Data Science

FLOW OF DATA COLLECTION

Begin by querying the SpaceX API to obtain data related to rocket launches.

Decode the response from the API using the .json() method.

Utilize .json_normalize() to flatten the JSON response and convert it into a structured dataframe.

Generate a dataframe from the organized dictionary to facilitate data analysis.

Structure the extracted information into a dictionary format to facilitate further processing.

Extract the specific details about the launches by applying

Filter the dataframe to retain only the launches involving the Falcon 9 rocket.

Address any missing values in the Payload Mass column by replacing them with the mean of that column, calculated using .mean().

Save the cleaned and processed dataframe to a CSV file for further use and analysis.

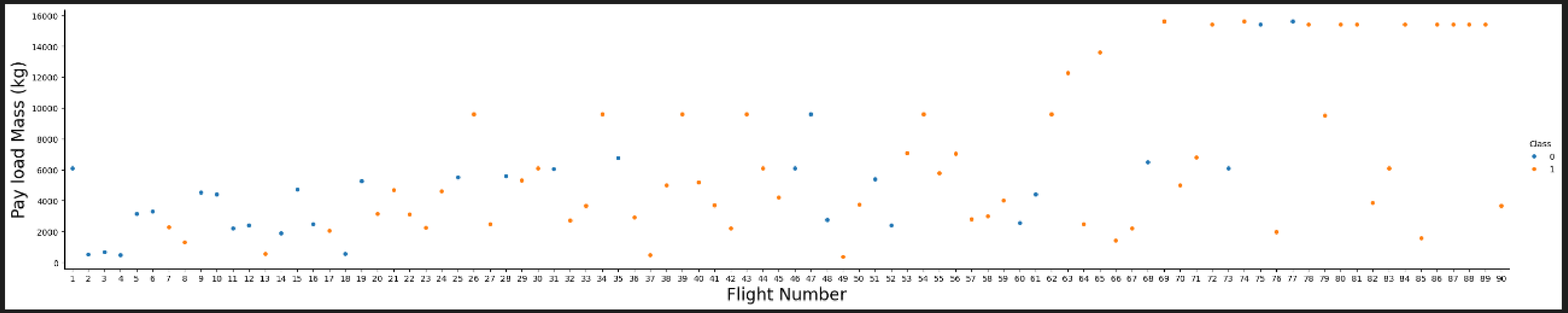
EDA with data visualization

IBM Data Science

Flight Number vs Payload Mass

```python
sns.catplot(y="PayloadMass", x="FlightNumber", hue="Class", data=df, aspect = 5)
plt.xlabel("Flight Number",fontsize=20)
plt.ylabel("Pay load Mass (kg)",fontsize=20)
plt.show()
```

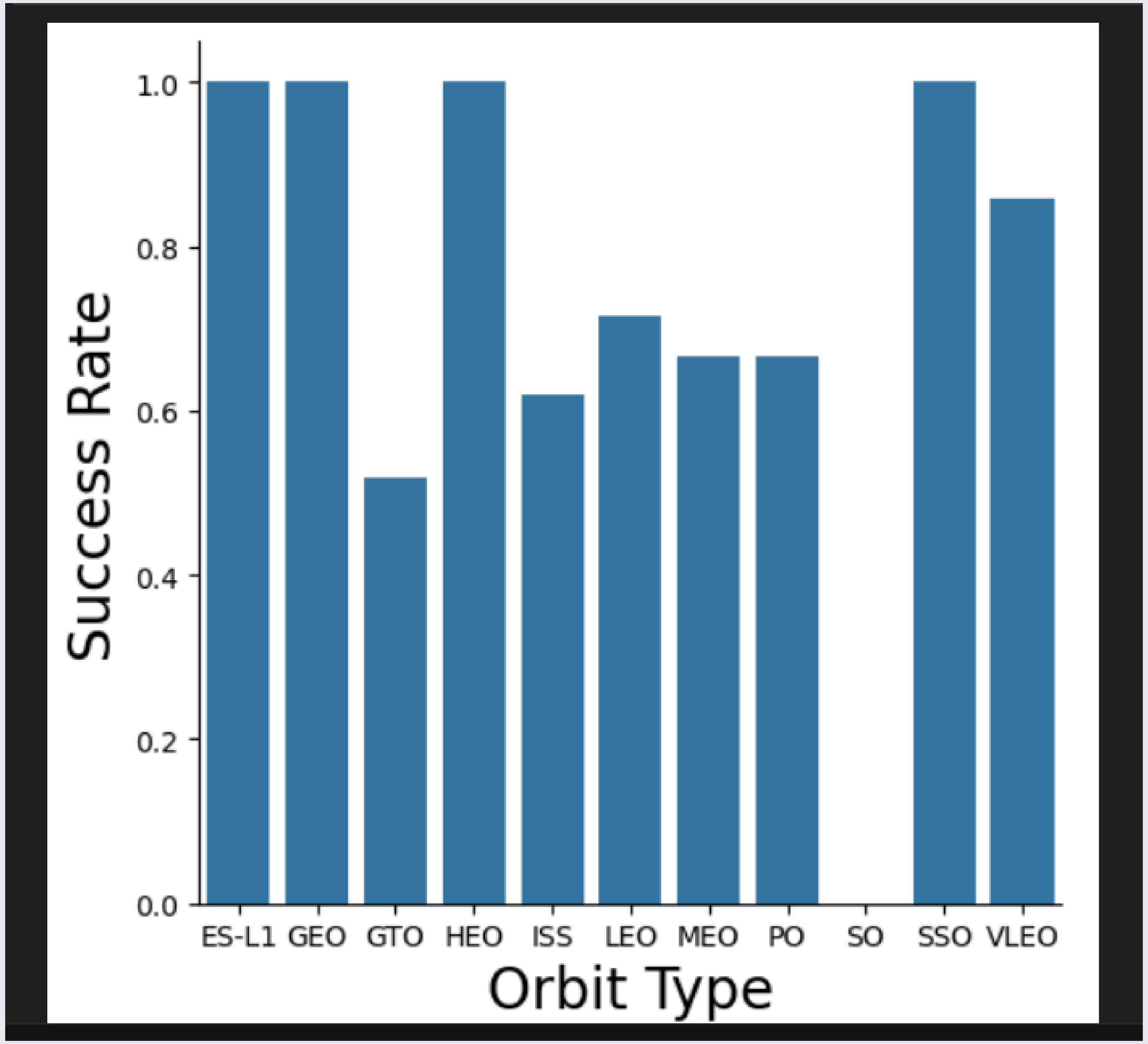
✓ 2.1s Python



METODELOGY:

In this analysis, I apply the scatter plot methodology to visualize SpaceX rocket launch data. The scatter plot allows for a comparison between two key variables: flight number and payload mass. Using this visualization, I can graphically assess the relationship between flight number and payload mass, identify patterns or trends, and see how payload mass changes with flight number.

Bar Chart : Success Rate VS Orbit Type



METODELOGY:

In this analysis, I use the bar chart methodology to compare between the success rate and the orbit type. The bar chart allows a clear visualization of the comparison between different types of orbits, with each bar representing the launch success rate for each orbit type. This method helps in evaluating and comparing how successful the rocket launches are in reaching the various types of targeted orbits.

Predictive analysis

IBM Data Science

FLOW OF PREDICTIVE CLASSFICIATION ANALYSIS

Begin by extracting the "Class" column from the dataset and converting it into a NumPy array.

Apply standardization to the data using StandardScaler.

Split the dataset into training and testing sets using the train_test_split function.

Create a GridSearchCV object with 10-fold cross-validation (cv=10).

Apply GridSearchCV to several models, including Logistic Regression (LogReg), Support Vector Machine (SVM),etc

Compute the accuracy of each model on the test set using the .score() method.

Examine the confusion matrix for each model to understand classification errors and the performance in predicting the correct and incorrect classes.

Assess model performance using additional metrics such as Jaccard_score and F1_score.

Classification

[31] ✓ 0.1s

	LogReg	SVM	Tree	KNN
Jaccard_Score	0.833333	0.845070	0.796875	0.819444
F1_Score	0.909091	0.916031	0.886957	0.900763
Accuracy	0.866667	0.877778	0.855556	0.855556

METODELOGY:

In this analysis, I employed four classification algorithms–Logistic Regression, Support Vector Machine (SVM), Decision Tree, and K-Nearest Neighbors (KNN)–to evaluate their performance using various metrics. The models were assessed based on their Jaccard Score, F1 Score, and Accuracy. The results revealed the following performance:

- Logistic Regression achieved a Jaccard Score of 0.833, an F1 Score of 0.909, and an Accuracy of 0.867.
- SVM performed slightly better with a Jaccard Score of 0.845, an F1 Score of 0.916, and an Accuracy of 0.878.
- Decision Tree yielded a Jaccard Score of 0.797, an F1 Score of 0.887, and an Accuracy of 0.856.
- KNN showed a Jaccard Score of 0.819, an F1 Score of 0.901, and an Accuracy of 0.856.

These metrics provide a comprehensive comparison of how each model handles classification tasks, with SVM demonstrating the highest overall performance across the evaluated criteria.

EDA with SQL

IBM Data Science

Launch site names begin with CCA

DATE	time__utc_	booster_version	launch_site	payload	payload_mass__kg_	orbit	customer	mission_outcome	landing__outcome
2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
2010-12-08	15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight C1, two CubeSats, barrel of Brouere cheese	0	LEO (ISS)	NASA (COTS) NRO	Success	Failure (parachute)
2012-05-22	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
2012-10-08	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
2013-03-01	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

METODELOGY:

Using this query :

select * from SPACEXDATASET where launch site like ' CCN ' limit 5;

Display 5 rows of table

Displaying the total payload mass

total_payload_mass
45596

METODELOGY:

Using this query :

select sum(payload_mass_kg) as total_payload_mass from SPACEXDATASET where customer = ‘NASA (CRS) ‘;

Showing the total payload mass transported by boosters launched for NASA's CRS missions.

Displaying average payload mass

average_payload_mass
2534

METODELOGY:

Using this query :

select avg(payload mass kg_) as average_payload mass from SPACEXDATASET where booster_version like '%F9 v1.1%';

Showing the average payload mass transported by the booster version F9 v1.1.

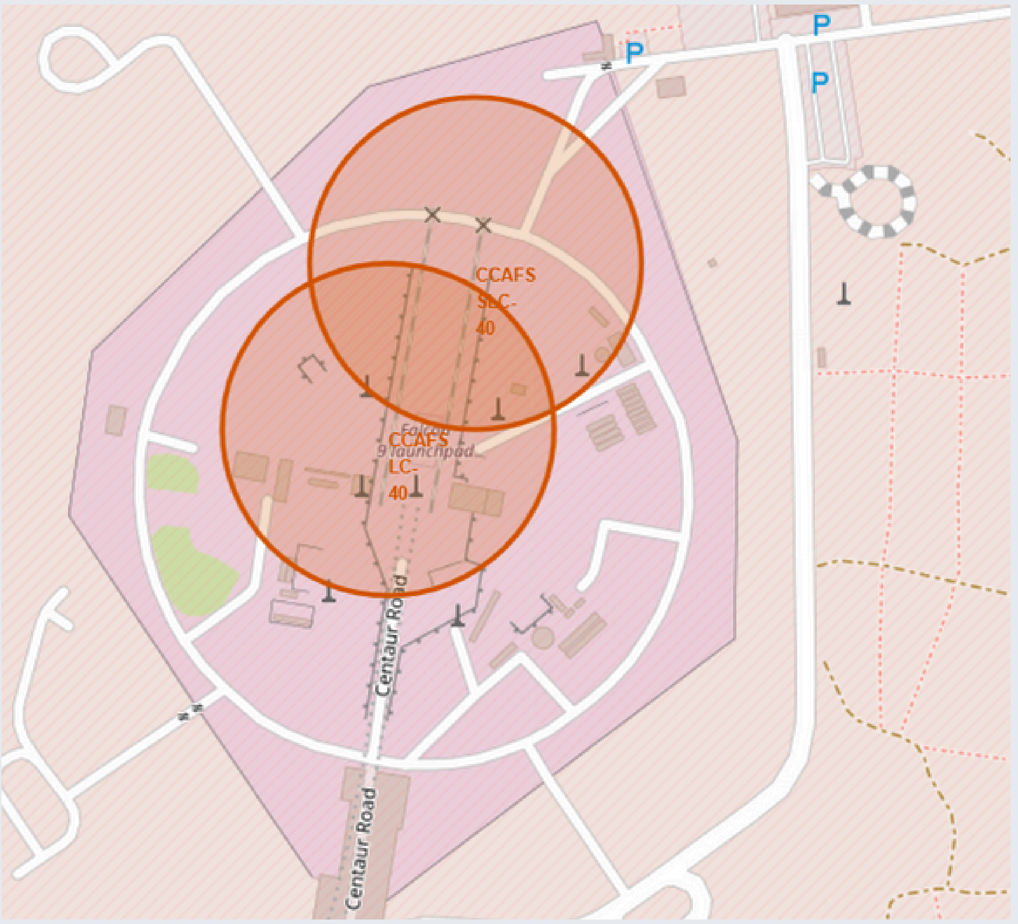
Mapping With Folium

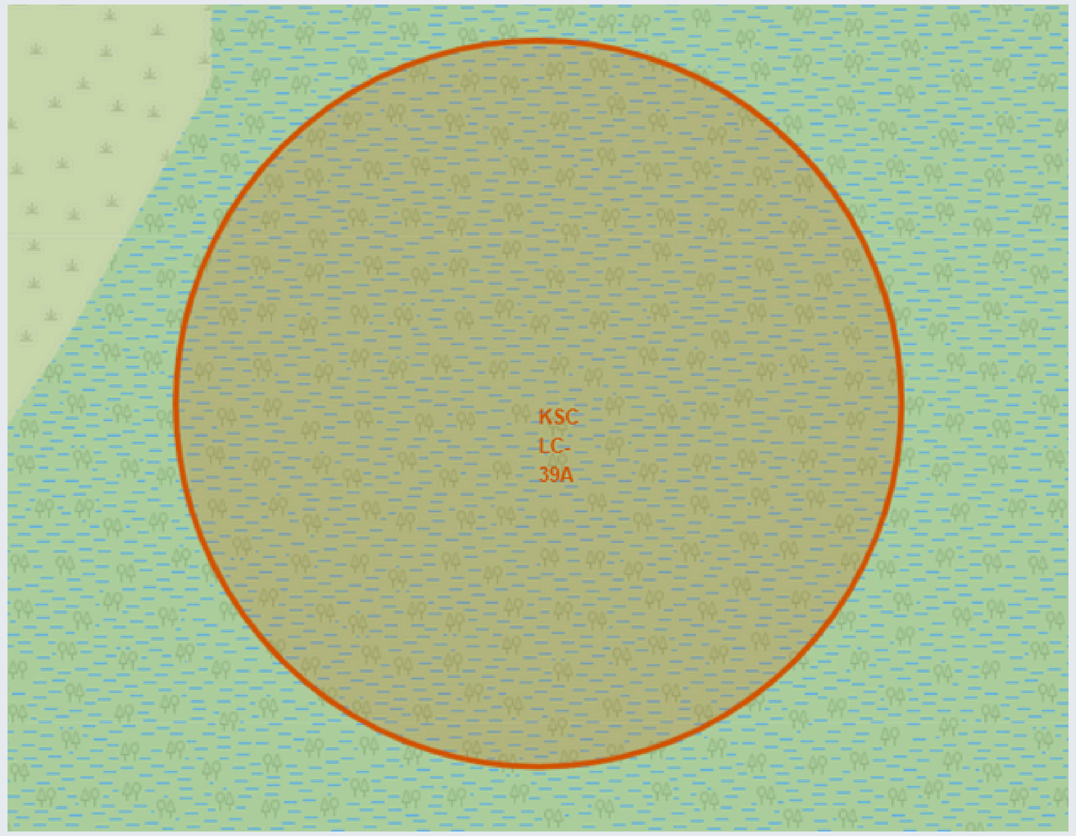
IBM Data Science

Metodelogy

• Folium maps are used to mark launch sites, along with indicators of successful and unsuccessful landings. Additionally, proximity to key locations such as railways, highways, coastlines, and cities is demonstrated.

• This helps to analyze the strategic placement of launch sites and provides a visual understanding of how location impacts the success of rocket landings.

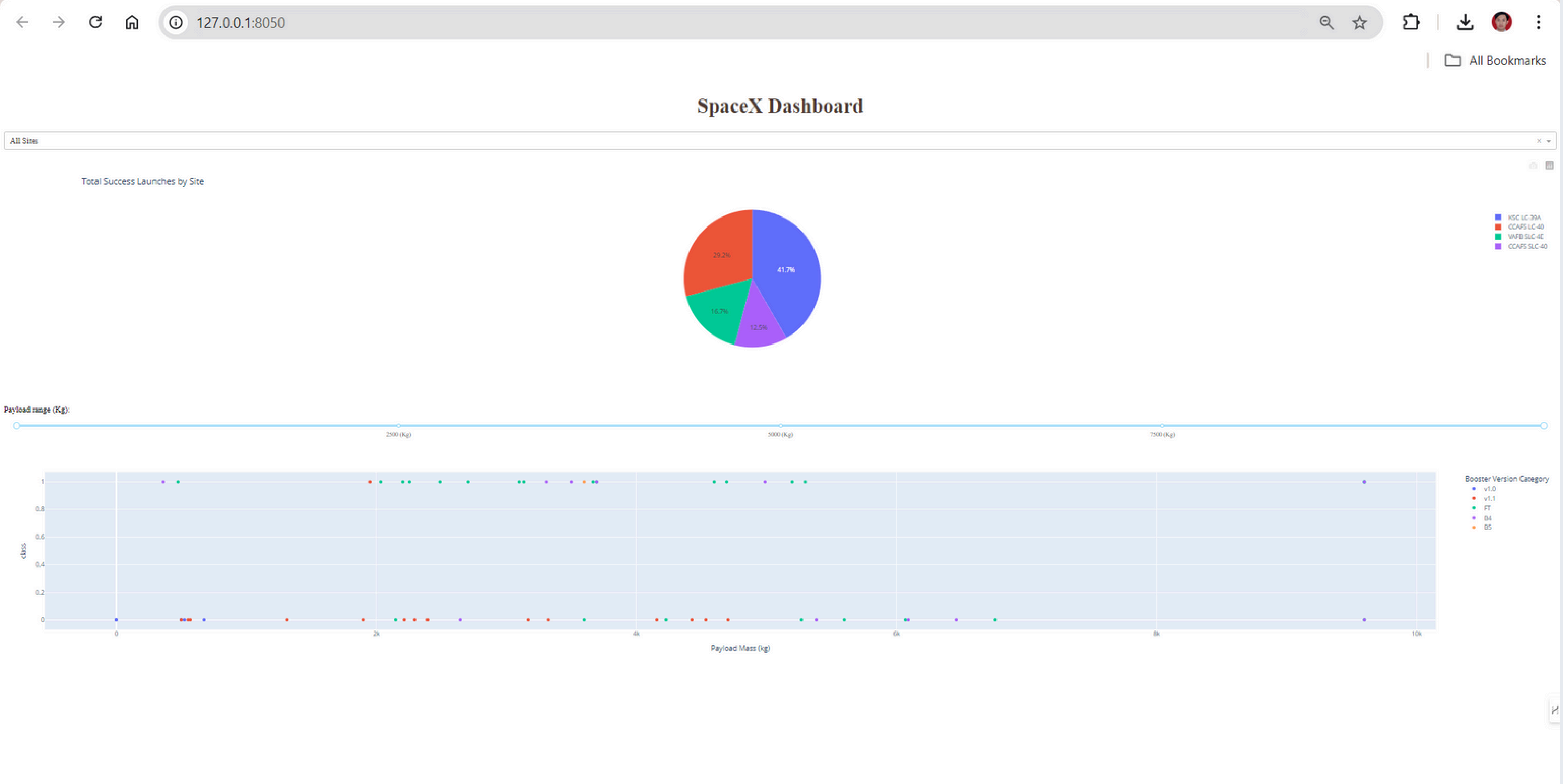




Dashboard With Ploty

IBM Data Science

Dashboard SpaceX



- The pie chart provides insights into the success rates of the launch sites.
- The scatter plot helps visualize how success rates vary based on different launch sites, payload mass, and booster version categories.

- The pie chart allows you to visualize the distribution of successful landings across all launch sites and can be filtered to display success rates for individual launch sites.
- The scatter plot takes two inputs: it allows you to select either all launch sites or a specific site and adjust the payload mass using a slider ranging from 0 to 10,000 kg.

CONCLUSION

- Several algorithms, including Logistic Regression, SVM, Decision Tree, and KNN, were employed to predict the success of SpaceX rocket launches.

- Folium maps were utilized to visualize launch sites and the success or failure of rocket landings.

- A scatter plot was used to analyze and visualize the relationship between flight number and payload mass, providing insights into how payload varies across different launches.

- Both pie charts and scatter plots were employed to visualize the distribution of success rates across various launch sites and to show how success varies based on payload mass and booster version category.

- GridSearchCV was applied to find the best parameters across different models like Logistic Regression, SVM, Decision Tree, and KNN.

Thank you

IBM Data Science