IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Pedro Carvalho
26/10/2025

Outline

- Executive Summary
- Introduction
- Methodology
- Results
- Conclusion
- Appendix

Executive Summary

- Collected data from the public SpaceX API and SpaceX Wikipedia page. Created a target column called **‘class’**, which indicates whether a launch was successfully landed. The data was explored using SQL queries, visualizations, Folium maps, and interactive dashboards. Relevant columns were selected as features, and all categorical variables were converted to binary format using one-hot encoding. The data was standardized, and **GridSearchCV** was used to identify the optimal hyperparameters for the machine learning models. Model performance was evaluated and visualized using accuracy scores.
- Four machine learning models were developed: **Logistic Regression**, **Support Vector Machine**, **Decision Tree Classifier**, and **K-Nearest Neighbors**. All models achieved similar results, with an average accuracy of approximately **83.33%**. However, all models tended to overpredict successful landings. Additional data and further optimization are required to improve model accuracy and reliability.

Introduction

Background

The commercial space age has arrived, with private companies leading innovation in space exploration and transportation. **SpaceX** currently dominates the industry, offering launch services at a significantly lower cost (**$62 million** compared to the industry average of **$165 million USD**).

This cost advantage is largely attributed to SpaceX's ability to **recover and reuse the first stage (Stage 1) of its rockets**, drastically reducing production and operational expenses.

To stay competitive, **SpaceY** aims to develop similar reusable rocket technology and reduce launch costs.

Problem Statement

SpaceY has tasked our team with developing a **machine learning model** capable of **predicting the successful recovery of Stage 1**.

Accurate predictions will help the company make data-driven decisions, optimize launch operations, and improve the likelihood of successful recoveries in future missions.

Section 1

Methodology

Methodology

Executive Summary

- Data was collected from the public SpaceX API and the SpaceX Wikipedia page. A target column, **'class'**, was created to label successful landings. The data was explored using SQL queries, visualizations, Folium maps, and interactive dashboards. Relevant columns were selected to serve as features. All categorical variables were converted to binary format using one-hot encoding. The data was then standardized, and **GridSearchCV** was applied to determine the best parameters for the machine learning models. The accuracy scores of all models were visualized for comparison.

- Four machine learning models were developed: **Logistic Regression, Support Vector Machine (SVM), Decision Tree Classifier,** and **K-Nearest Neighbors (KNN)**. Each model achieved a similar accuracy rate of approximately **83.33%**. However, all models tended to overpredict successful landings. Additional data would be beneficial to improve model selection and overall accuracy.

Data Collection

The data collection process combined **API requests** from the public **SpaceX API** and **web scraping** from the **SpaceX Wikipedia page**.

These two sources were used to gather comprehensive launch and landing data for model training and analysis.

- The **next slide** presents the **flowchart of data collection from the SpaceX API**.
- The **following slide** illustrates the **flowchart of data collection through web scraping**.

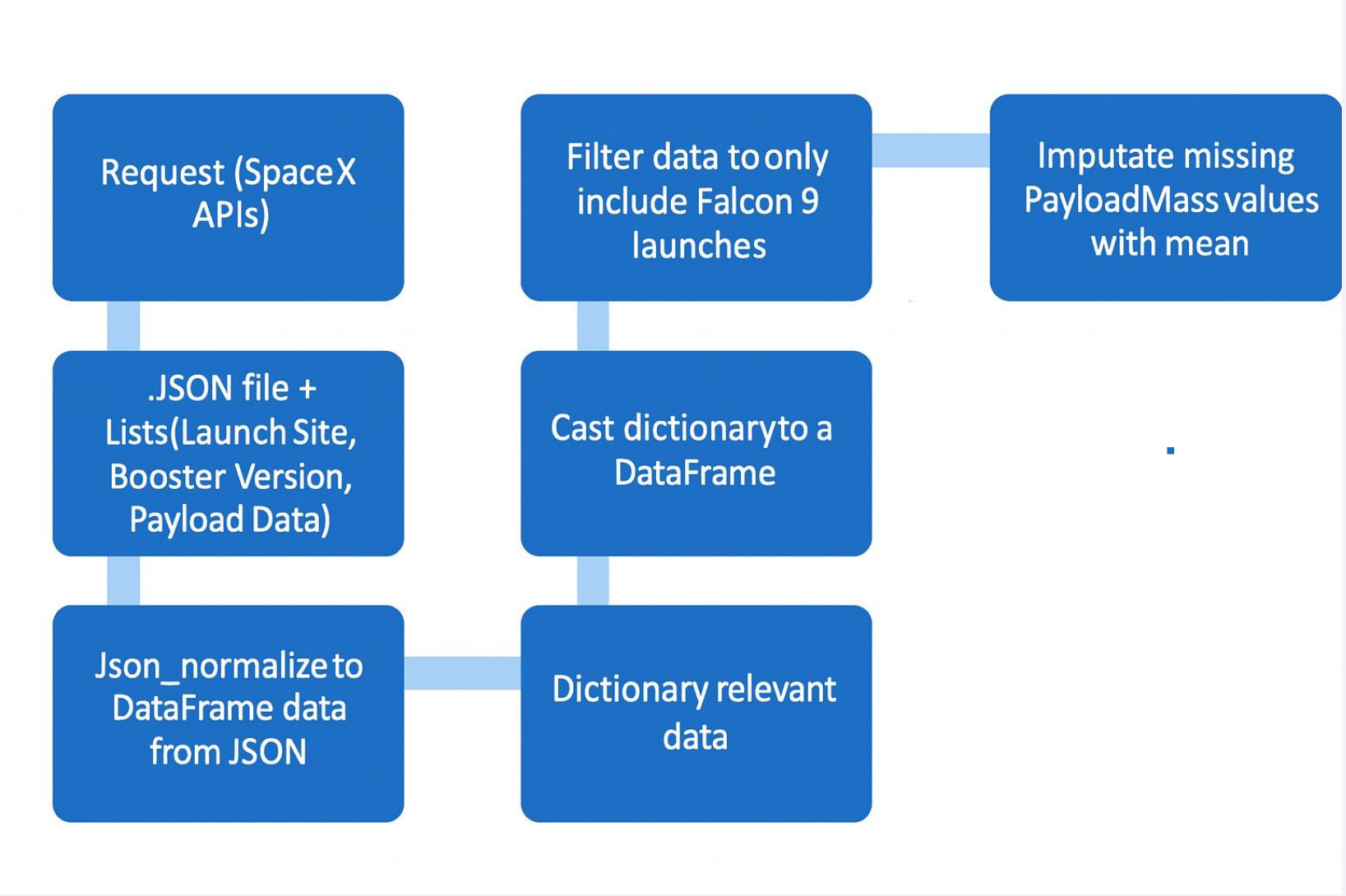
SpaceX API Data Columns

- **FlightNumber**, **Date**, **BoosterVersion**, **PayloadMass**, **Orbit**, **LaunchSite**, **Outcome**, **Flights**, **GridFins**, **Reused**, **Legs**, **LandingPad**, **Block**, **ReusedCount**, **Serial**, **Longitude**, **Latitude**

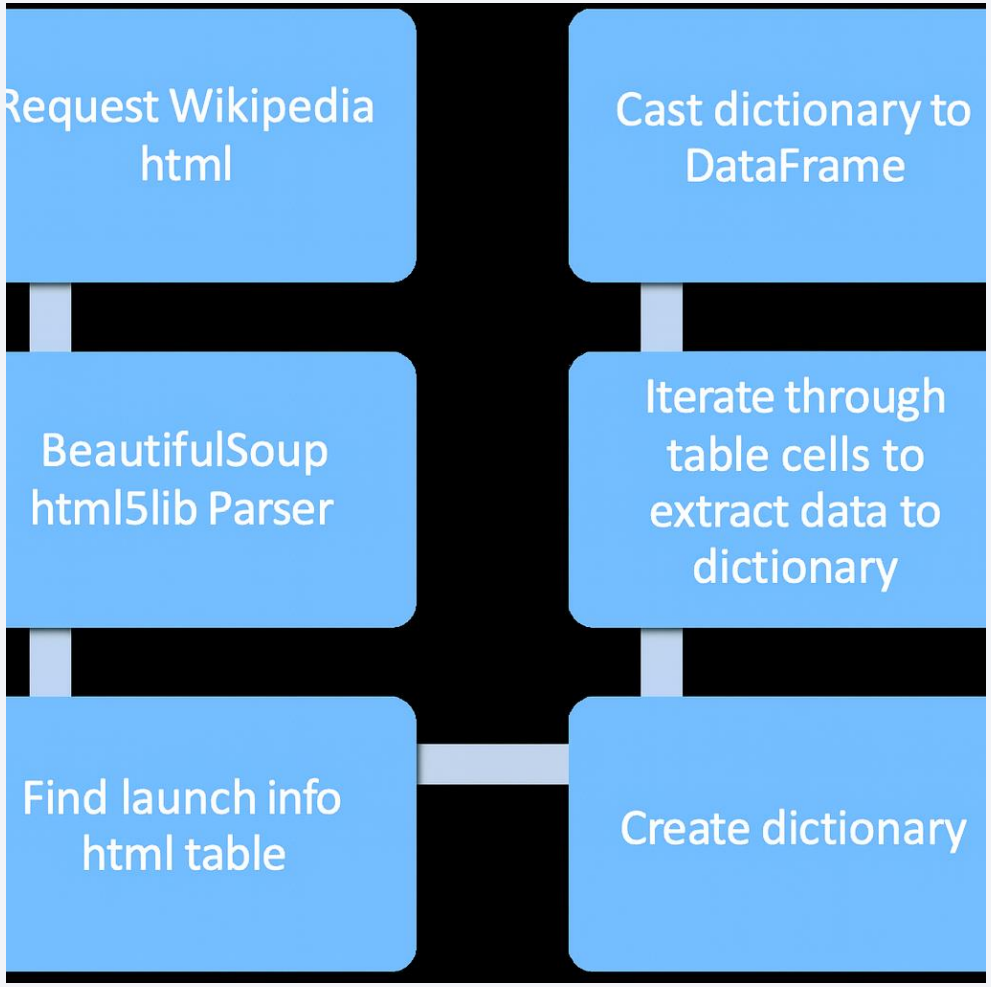
Wikipedia Web-Scraped Data Columns

- **Flight No.**, **Launch Site**, **Payload**, **PayloadMass**, **Orbit**, **Customer**, **Launch Outcome**, **Version Booster**, **Booster Landing**, **Date**, **Time**

Data Collection – SpaceX API



Data Collection - Scraping



Data Wrangling

The **Outcome** column consists of two components: **‘Mission Outcome’** and **‘Landing Location’**.

A new column named **‘class’** was generated, assigning a value of **1** if the **Mission Outcome** was successful (**True**) and **0** otherwise.

Value Mapping:

- True ASDS, True RTLS, and True Ocean → **1 (Successful Landing)**
- None None, False ASDS, None ASDS, False Ocean, and False RTLS → **0 (Failed Landing)**

EDA with Data Visualization

Plots Used:

- Flight Number vs. Payload Mass
- Flight Number vs. Launch Site
- Payload Mass vs. Launch Site
- Orbit vs. Success Rate
- Flight Number vs. Orbit
- Payload vs. Orbit
- Success Yearly Trend

Visualization Techniques:

- Scatter plots, line charts, and bar plots were used to analyze and compare relationships between variables.
- The goal was to identify significant correlations that could inform feature selection and improve the performance of the machine learning models.

EDA with SQL

- The dataset was **loaded into an IBM DB2 Database** and explored using **SQL integrated with Python**.
  - SQL queries were executed to gain deeper insights into the dataset, including:
  - Launch site names
  - Mission outcomes
  - Payload sizes across different customers and booster versions
  - Landing outcomes
- This process provided a clearer understanding of the data and supported feature selection for the machine learning models.

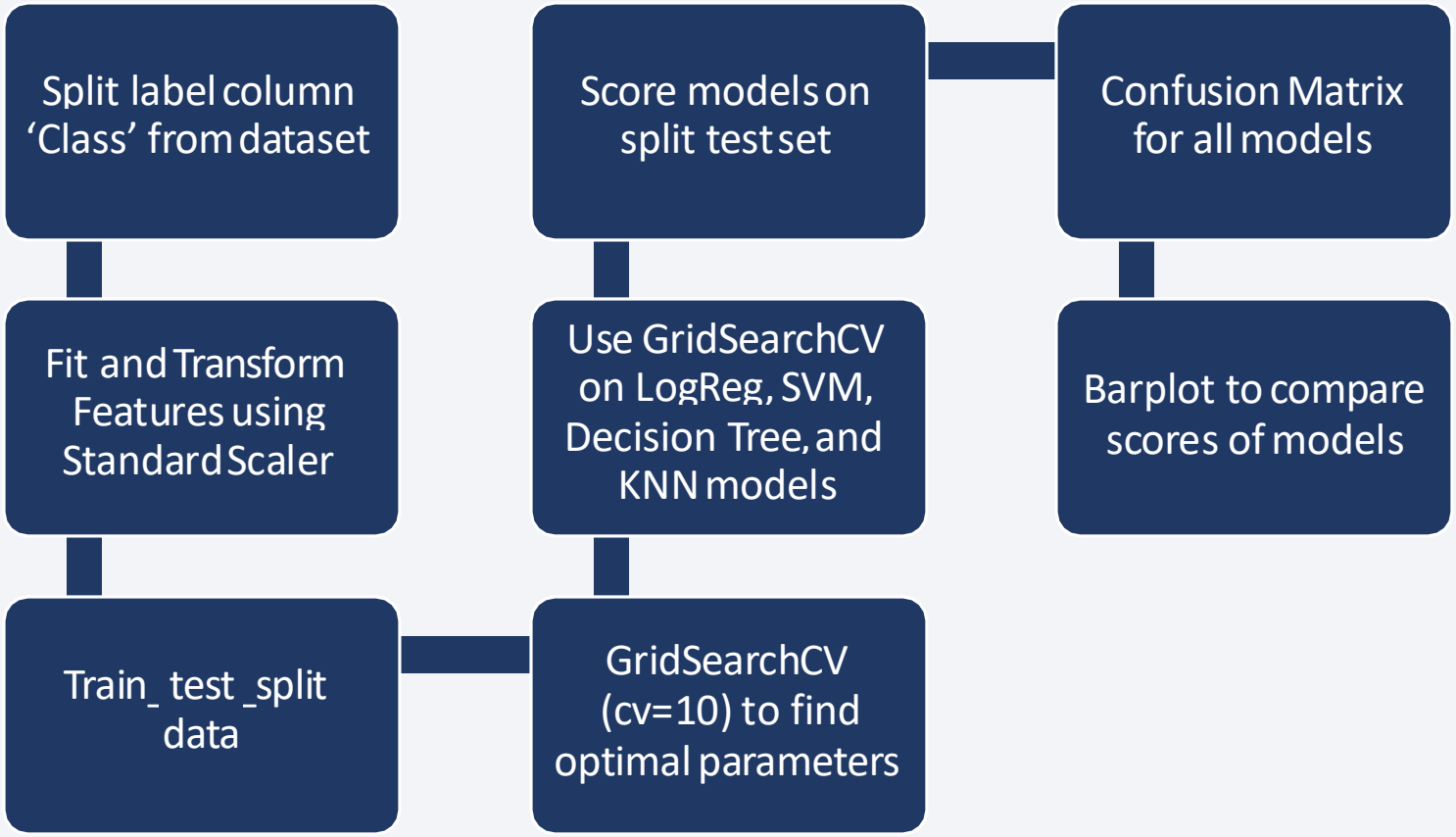
Build an Interactive Map with Folium

- Folium maps mark Launch Sites, successful and unsuccessful landings, and a proximity example to key locations: Railway, Highway, Coast, and City.
- This allows us to understand why launch sites may be located where they are. Also visualizes successful landings relative to location.

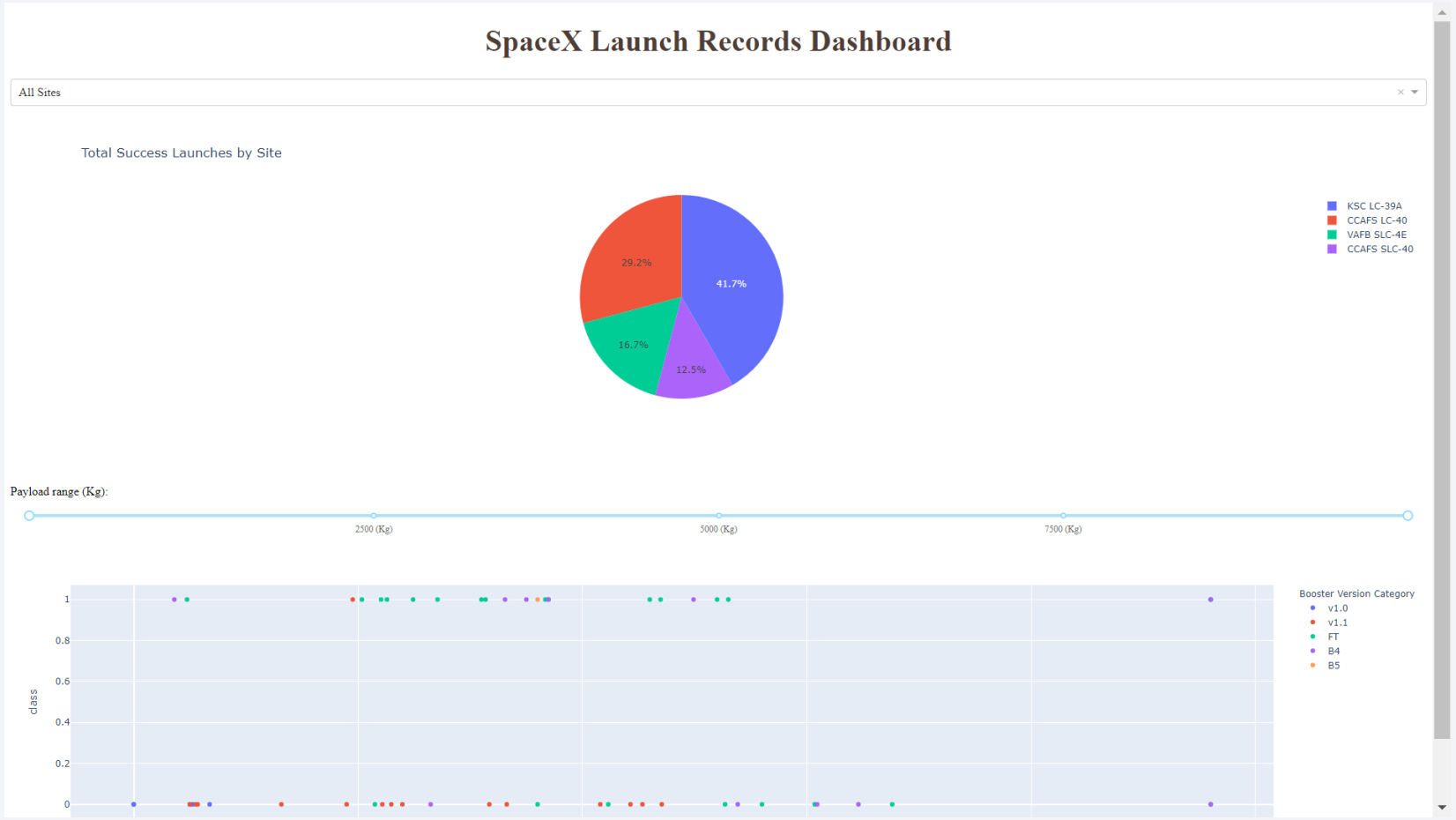
Build a Dashboard with Plotly Dash

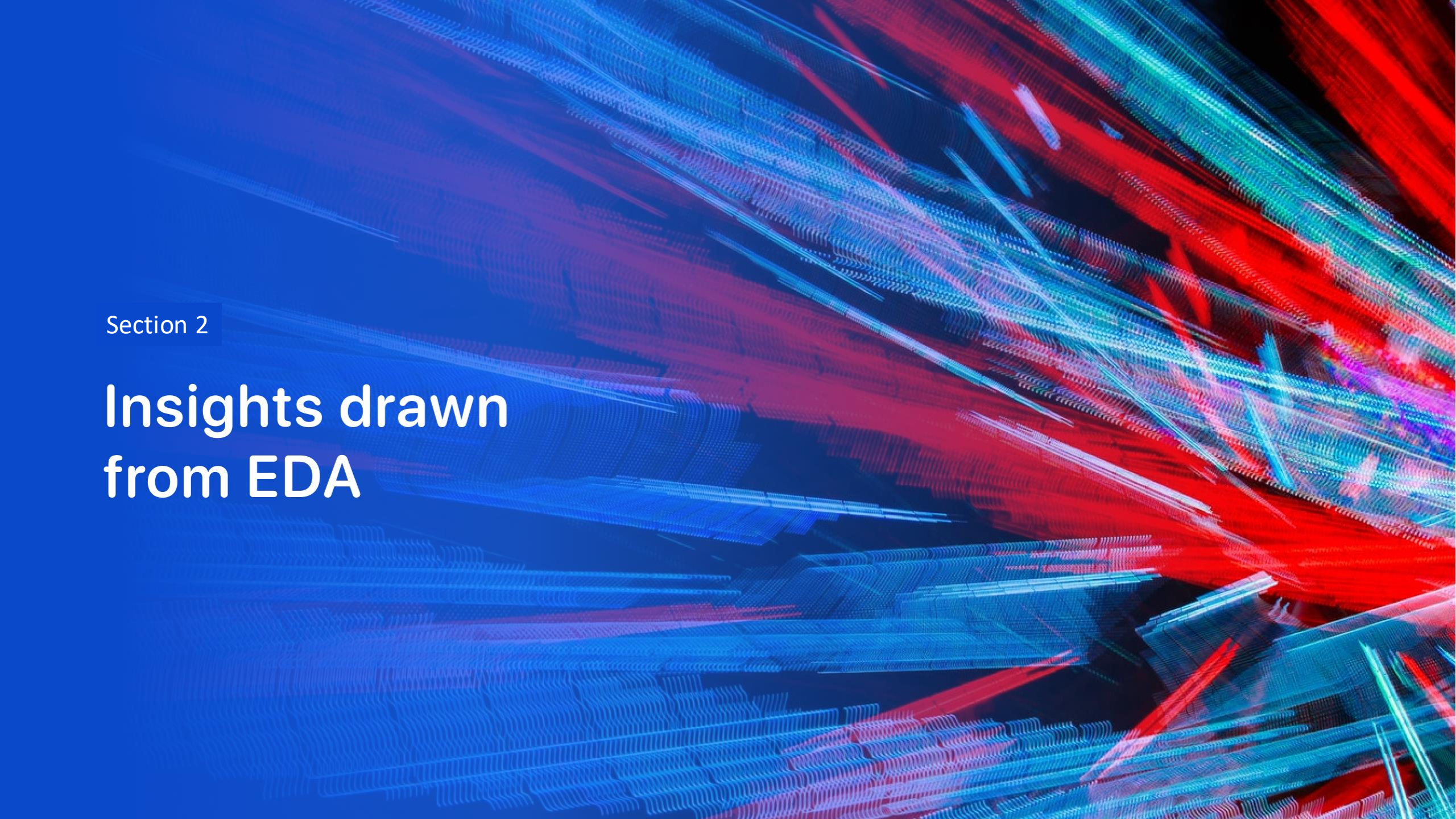
- The dashboard includes a **pie chart** and a **scatter plot**.
- The **pie chart** displays the distribution of successful landings across all launch sites. It can also be filtered to show the success rate for an individual launch site.
- The **scatter plot** allows users to select either all sites or a specific site, and adjust the payload mass using a slider ranging from 0 to 10,000 kg.
- The **pie chart** visualizes the launch site success rates, while the **scatter plot** helps analyze how success varies with launch site, payload mass, and booster version category.

Predictive Analysis (Classification)

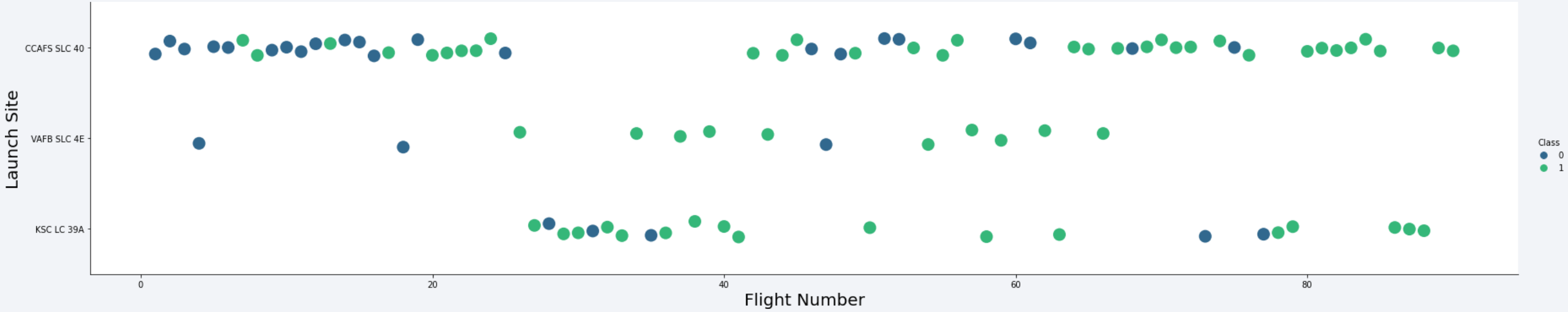


Results


SpaceX Launch Records Dashboard
All Sites
Total Success Launches by Site
29.2%
41.7%
16.7%
12.5%
KSC LC-39A
CCAFS LC-40
VAFB SLC-4E
CCAFS SLC-40
Payload range (Kg):
2500 (Kg)
5000 (Kg)
7500 (Kg)
class
1
0.8
0.6
0.4
0.2
0
Booster Version Category
v1.0
v1.1
FT
B4
B5

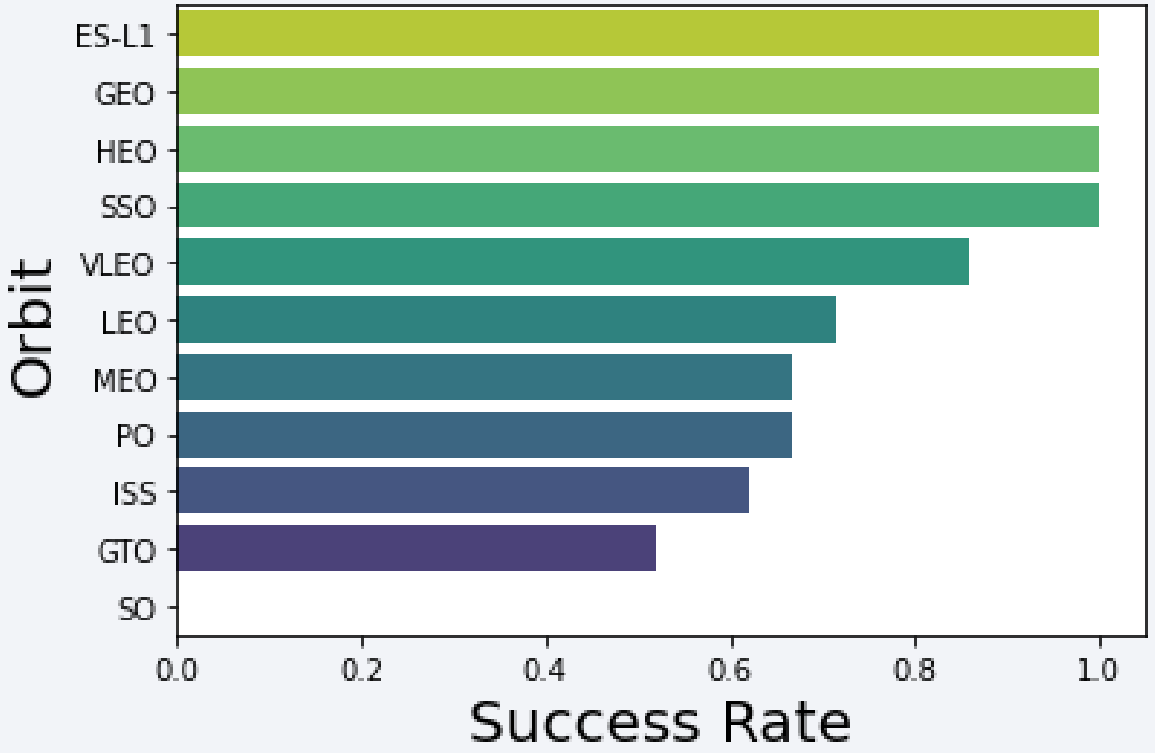

Section 2
Insights drawn from EDA

Flight Number vs. Launch Site

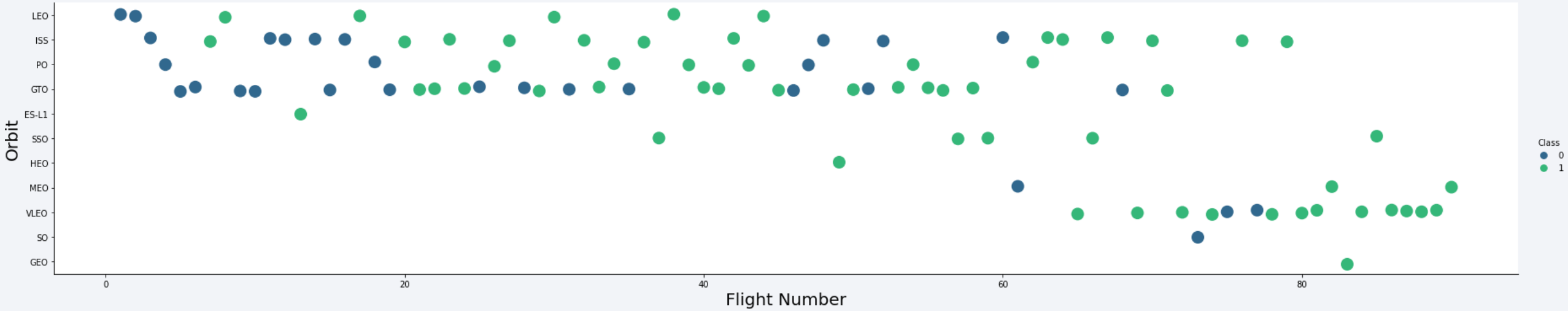


Payload vs. Launch Site

Success Rate vs. Orbit Type

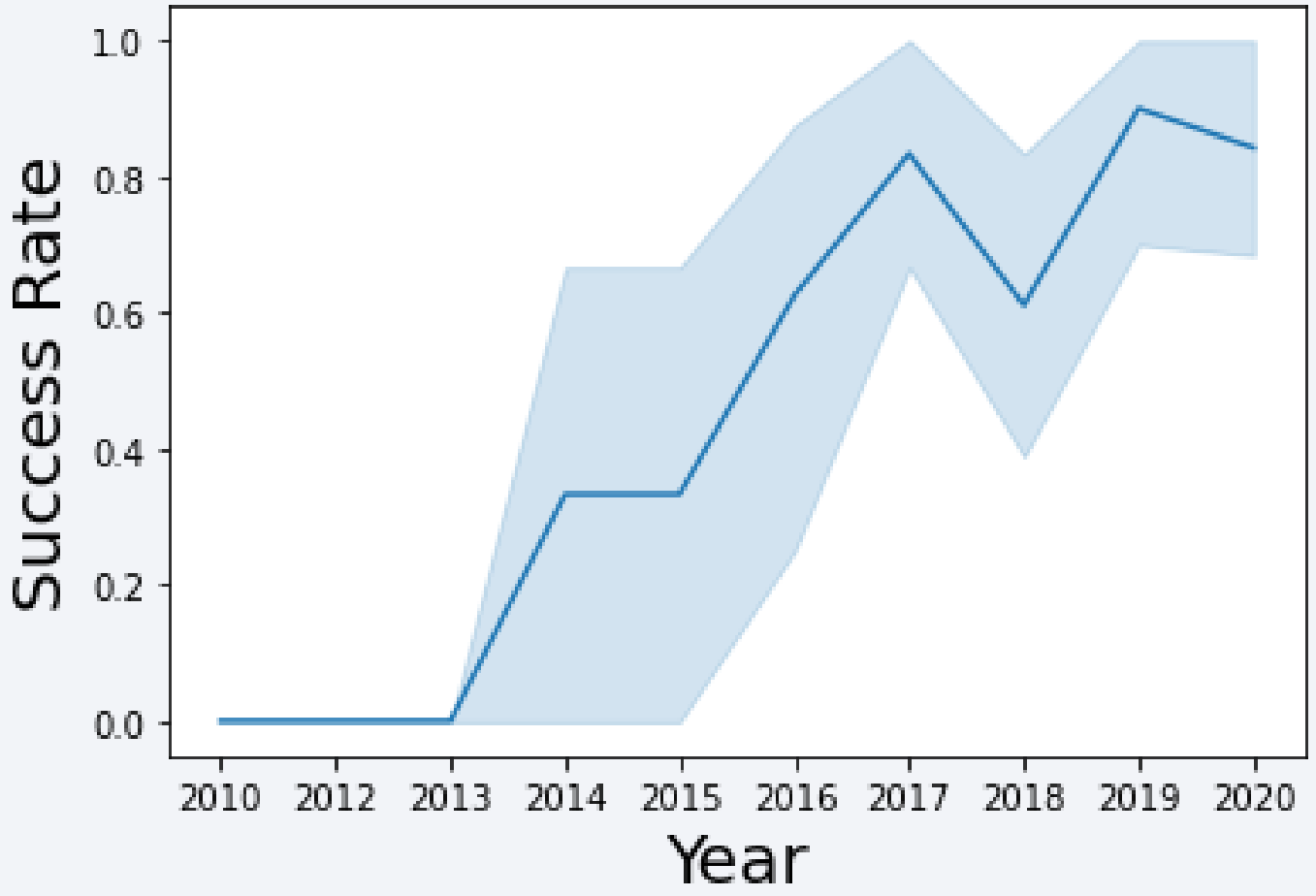


Flight Number vs. Orbit Type



Payload vs. Orbit Type

Launch Success Yearly Trend



All Launch Site Names

```
In [4]: %%sql
        SELECT UNIQUE LAUNCH_SITE
        FROM SPACEXDATASET;

         * ibm_db_sa://ftb12020:***@0c77d6f2
        Done.
```

Out[4]:

launch_site
CCAFS LC-40
CCAFS SLC-40
CCAFSSLC-40
KSC LC-39A
VAFB SLC-4E

Launch Site Names Begin with 'CCA'

In [5]:
```
%%sql
SELECT *
FROM SPACEXDATASET
WHERE LAUNCH_SITE LIKE 'CCA%'
LIMIT 5;
```

```
 * ibm_db_sa://ftb12020:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/bludb
Done.
```

Out[5]:

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

Total Payload Mass

```
%%sql
SELECT SUM(PAYLOAD_MASS__KG_) AS SUM_PAYLOAD_MASS_KG
FROM SPACEXDATASET
WHERE CUSTOMER = 'NASA (CRS)';
```

```
 * ibm_db_sa://ftb12020:***@0c77d6f2-5da9-48a9-81f8-86
Done.
```

sum_payload_mass_kg
45596

Average Payload Mass by F9 v1.1

```
%%sql
SELECT AVG(PAYLOAD_MASS__KG_) AS AVG_PAYLOAD_MASS_KG
FROM SPACEXDATASET
WHERE booster_version = 'F9 v1.1'
```

```
 * ibm_db_sa://ftb12020:***@0c77d6f2-5da9-48a9-81f8-8
Done.
```

avg_payload_mass_kg
2928

First Successful Ground Landing Date

```
%%sql
SELECT MIN(DATE) AS FIRST_SUCCESS
FROM SPACEXDATASET
WHERE landing__outcome = 'Success (ground pad)';
```

```
 * ibm_db_sa://ftb12020:***@0c77d6f2-5da9-48a9-81
Done.
```

first_success
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

```
%%sql
SELECT booster_version
FROM SPACEXDATASET
WHERE landing__outcome = 'Success (drone ship)' AND payload_mass__kg_ BETWEEN 4001 AND 5999;
```

```
 * ibm_db_sa://ftb12020:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.database
Done.
```

booster_version
F9 FT B1022
F9 FT B1026
F9 FT B1021.2
F9 FT B1031.2

Total Number of Successful and Failure Mission Outcomes

```
%%sql
SELECT mission_outcome, COUNT(*) AS no_outcome
FROM SPACEXDATASET
GROUP BY mission_outcome;
```

```
 * ibm_db_sa://ftb12020:***@0c77d6f2-5da9-48a9-
Done.
```

mission_outcome	no_outcome
Failure (in flight)	1
Success	99
Success (payload status unclear)	1

Boosters Carried Maximum Payload

```
%%sql
SELECT booster_version, PAYLOAD_MASS__KG_
FROM SPACEXDATASET
WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXDATASET);
```

```
 * ibm_db_sa://ftb12020:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1
Done.
```

booster_version	payload_mass__kg_
F9 B5 B1048.4	15600
F9 B5 B1049.4	15600
F9 B5 B1051.3	15600
F9 B5 B1056.4	15600
F9 B5 B1048.5	15600
F9 B5 B1051.4	15600
F9 B5 B1049.5	15600
F9 B5 B1060.2	15600
F9 B5 B1058.3	15600
F9 B5 B1051.6	15600
F9 B5 B1060.3	15600
F9 B5 B1049.7	15600

2015 Launch Records

```
%%sql
SELECT MONTHNAME(DATE) AS MONTH, landing__outcome, booster_version, PAYLOAD_MASS__KG_, launch_site
FROM SPACEXDATASET
WHERE landing__outcome = 'Failure (drone ship)' AND YEAR(DATE) = 2015;
```

```
 * ibm_db_sa://ftb12020:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.app
Done.
```

MONTH	landing__outcome	booster_version	payload_mass__kg_	launch_site
January	Failure (drone ship)	F9 v1.1 B1012	2395	CCAFS LC-40
April	Failure (drone ship)	F9 v1.1 B1015	1898	CCAFS LC-40

Rank Landing Outcomes Between 2010-06-04 and 2017-03-20

```
%%sql
SELECT landing__outcome, COUNT(*) AS no_outcome
FROM SPACEXDATASET
WHERE landing__outcome LIKE 'Succes%' AND DATE BETWEEN '2010-06-04' AND '2017-03-20'
GROUP BY landing__outcome
ORDER BY no_outcome DESC;
```

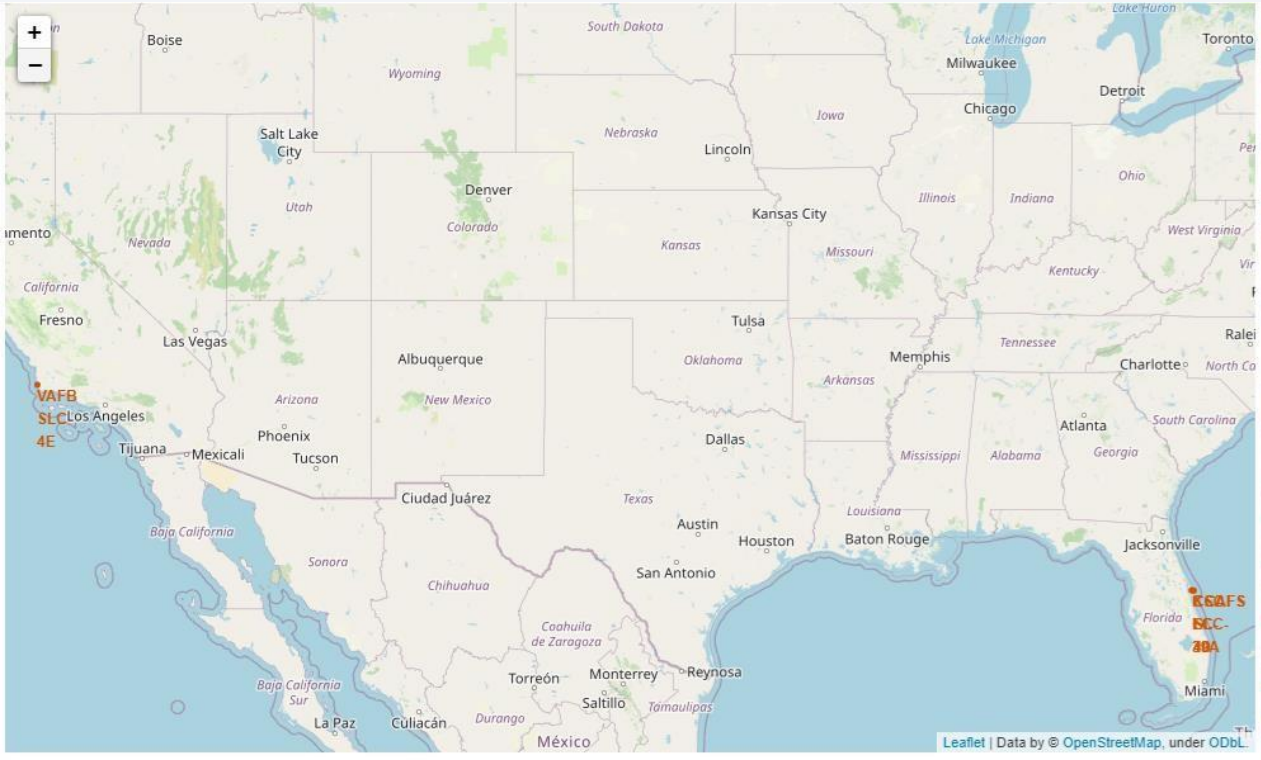
```
 * ibm_db_sa://ftb12020:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg
Done.
```

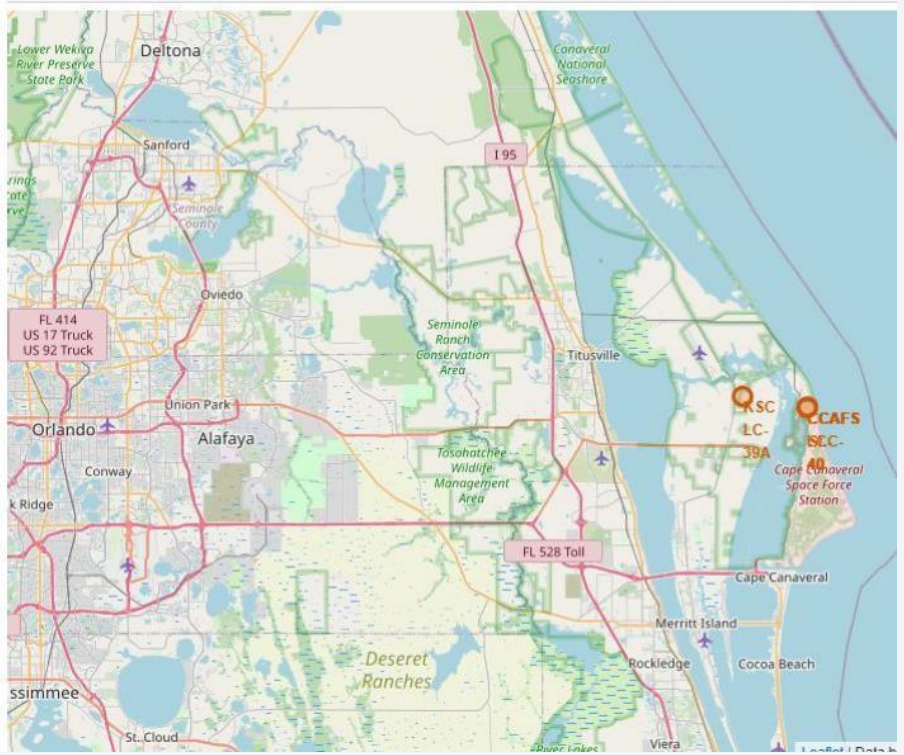
landing__outcome	no_outcome
Success (drone ship)	5
Success (ground pad)	3

Section 3

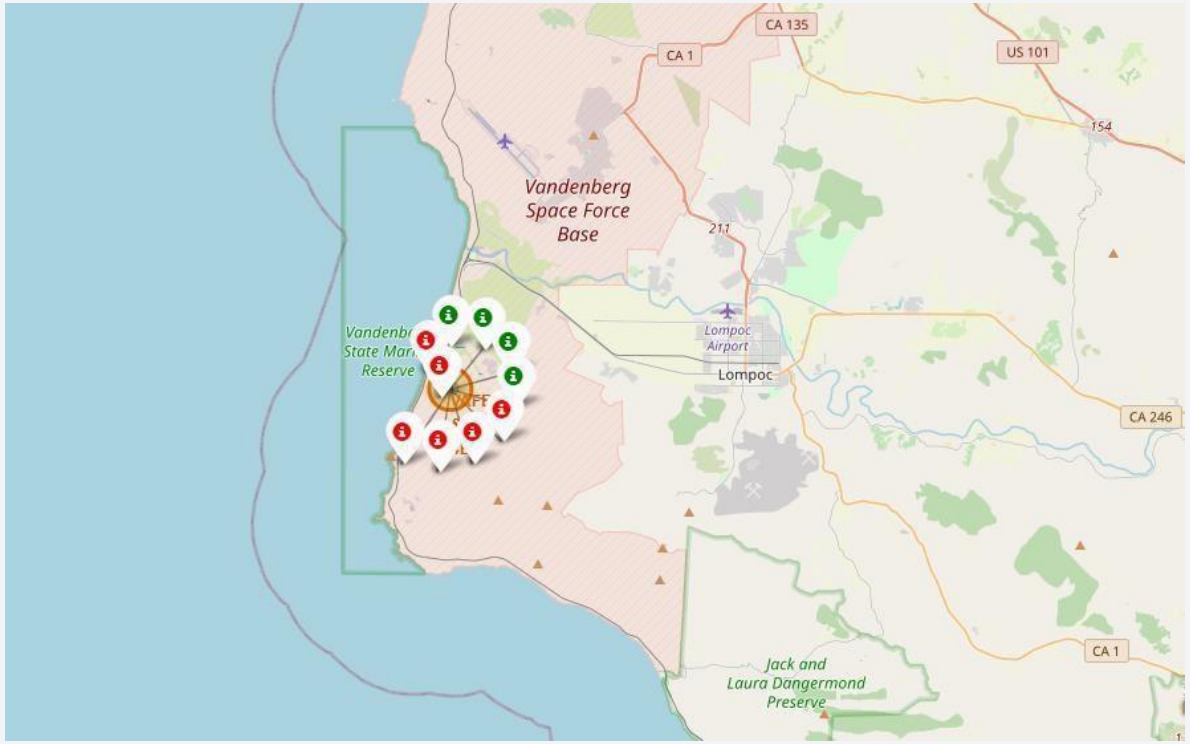
Launch Sites Proximities Analysis

<Folium Map Screenshot 1>

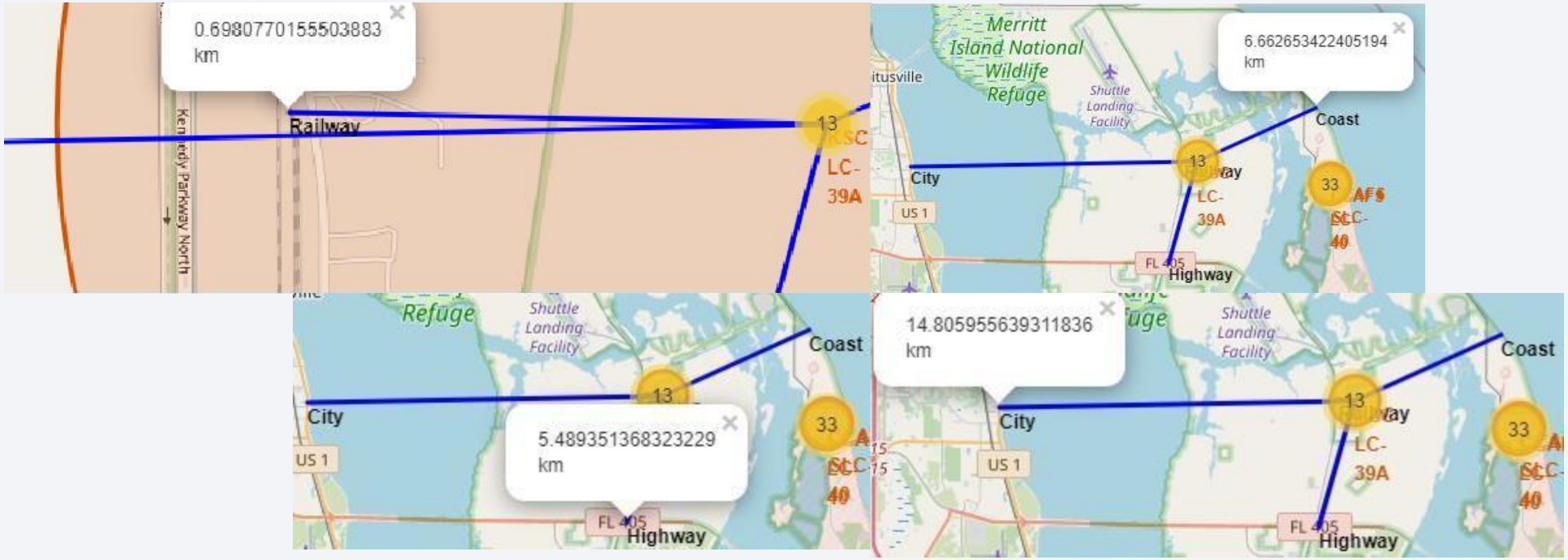


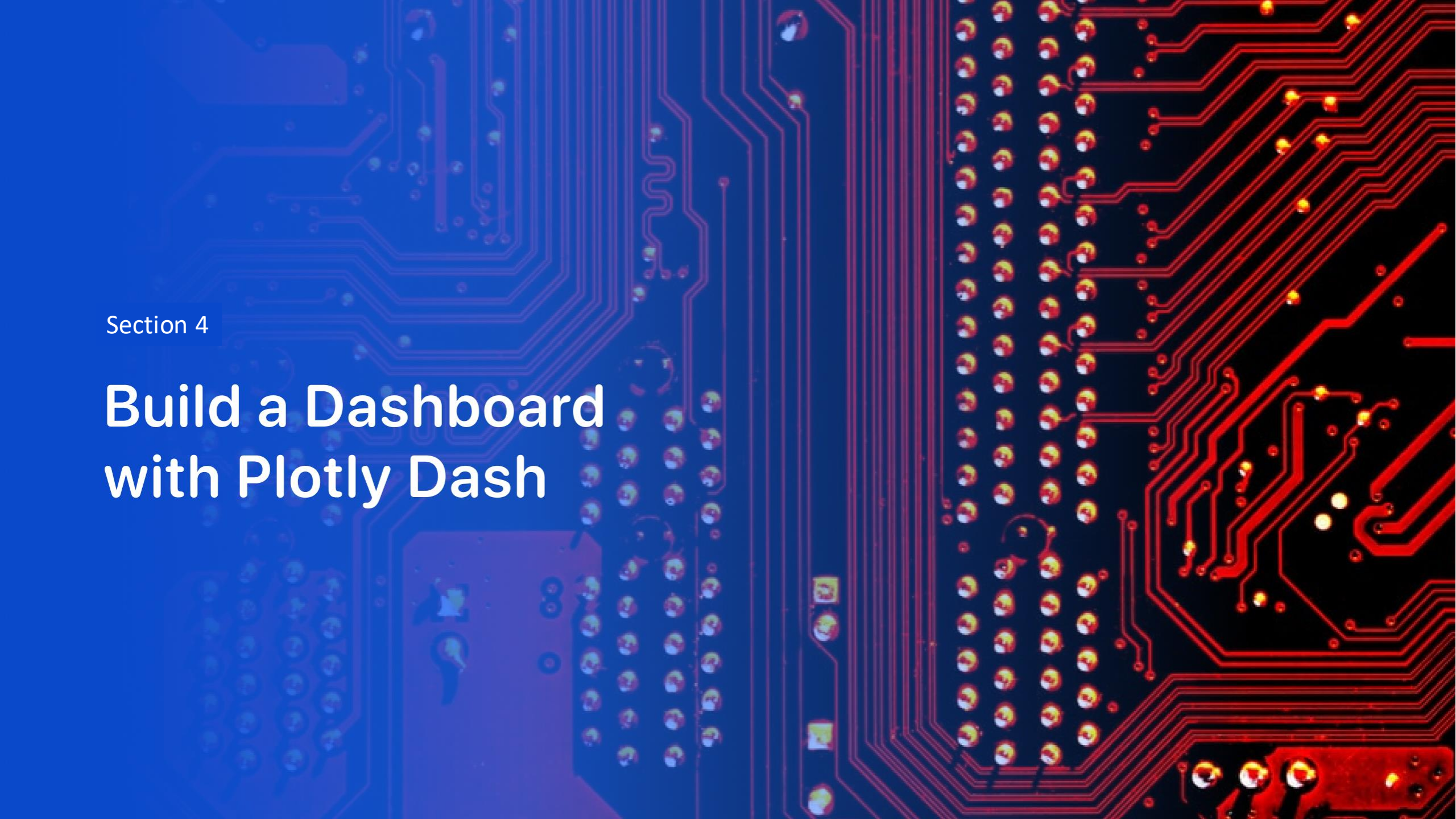


<Folium Map Screenshot 2>

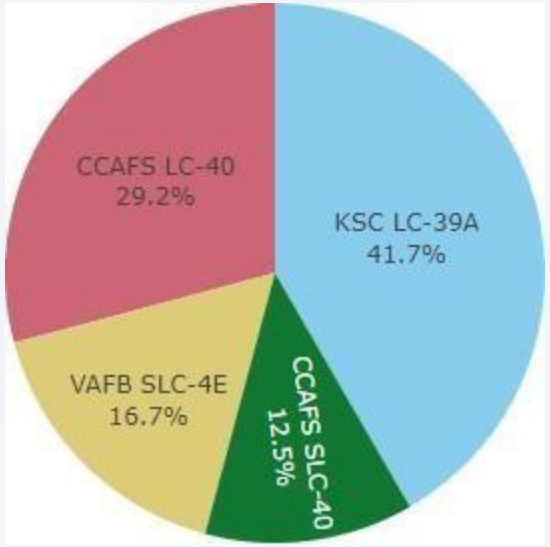


<Folium Map Screenshot 3>

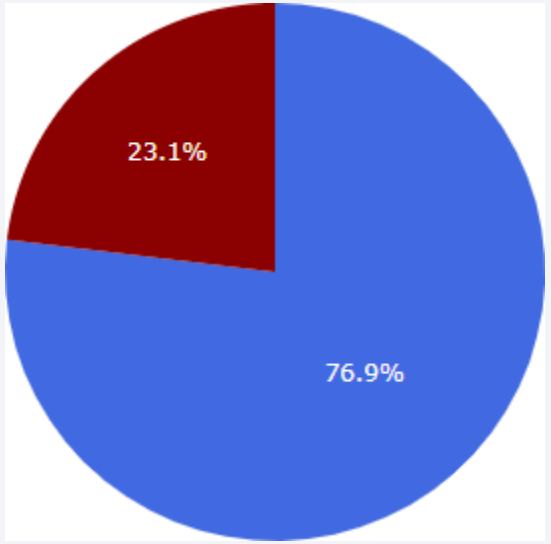



Section 4
Build a Dashboard with Plotly Dash

<Dashboard Screenshot 1>



<Dashboard Screenshot 2>

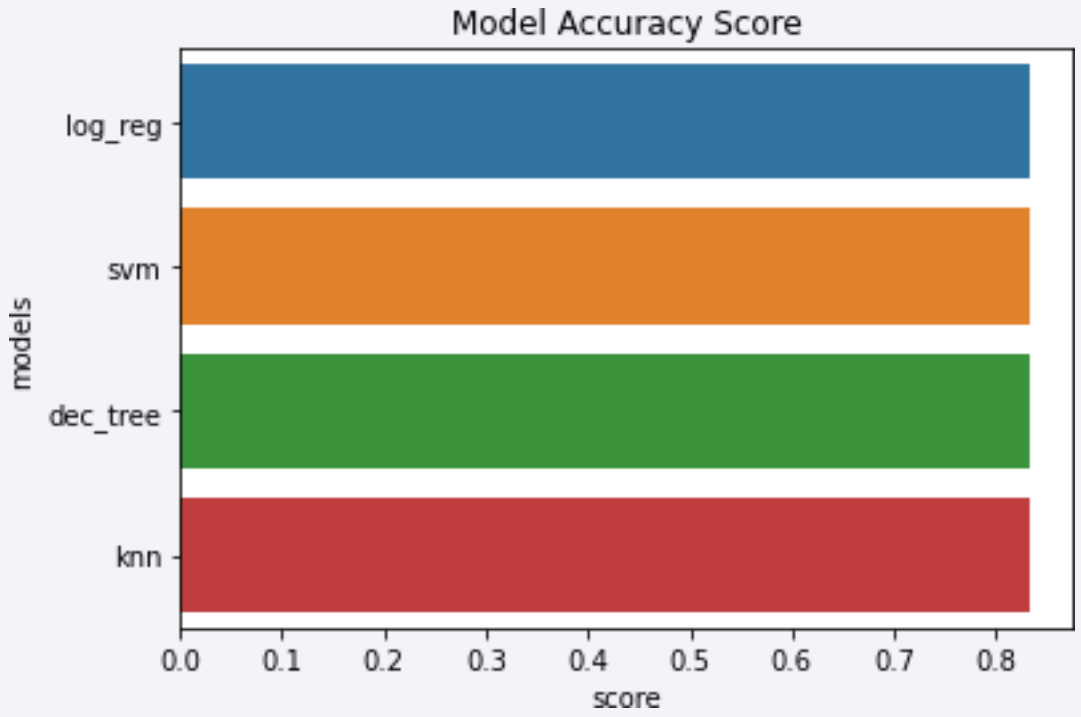


<Dashboard Screenshot 3>

Section 5

Predictive Analysis (Classification)

Classification Accuracy



Confusion Matrix

Conclusions

- Our task was to develop a **machine learning model** for **SpaceY**, a company aiming to compete with **SpaceX**.
- The goal of the model is to **predict whether the Stage 1 booster will successfully land**, potentially saving the company around **$100 million USD per launch**.

- We used data collected from a **public SpaceX API** and through **web scraping** of the SpaceX **Wikipedia page**. The data was **cleaned, labeled, and stored** in a **DB2 SQL database** for analysis.
- To visualize the insights, we developed an **interactive dashboard** featuring a pie chart and a scatter plot.
- Using this data, we trained a **machine learning model** that achieved an **accuracy of 83%**.
- This model enables **Allon Mask**, the CEO of SpaceY, to **predict with high confidence whether a launch's Stage 1 landing will be successful** before the launch takes place — helping to make informed go/no-go decisions and reduce financial risk.
- To further improve the model's performance and reliability, **collecting additional data** is recommended to help identify the best possible algorithm and enhance prediction accuracy.

Appendix

Thank you!