IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Sheldon Gordon
July 12, 2024

Outline

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

Executive Summary

This project aims to enhance the predictive capabilities of a private space launch company by determining the factors that influence the successful landing of the Falcon 9 first stage. Leveraging extensive data collection from sources such as the SpaceX API and historical launch records from Wikipedia, we performed rigorous data wrangling and exploratory data analysis (EDA) to ensure data quality and uncover critical insights. Key variables affecting landing success were identified and analyzed through SQL querying and feature engineering, leading to a deeper understanding of the launch dynamics.

Machine learning techniques, including SVM, classification trees, and logistic regression, were employed to build robust predictive models. An interactive dashboard application, developed using Plotly Dash, provides real-time visual analytics, enabling stakeholders to explore launch data dynamically. The analysis revealed that optimizing launch site selection and understanding payload characteristics significantly impact success rates. These insights offer strategic guidance for future launches, potentially reducing costs and increasing the competitive advantage of the company in the space launch market.

Introduction

Project Background and Context

In the rapidly evolving aerospace industry, private space launch companies are at the forefront of innovation, striving to reduce costs and increase the reliability of space missions. Among these companies, SpaceX stands out for its pioneering achievements, including the successful reuse of the Falcon 9 first stage. By enabling the first stage to land safely and be reused, SpaceX has significantly lowered the cost of rocket launches, offering a competitive price of $62 million per launch compared to other providers charging upwards of $165 million. This cost reduction is a critical factor for potential clients considering bids for space launch services.

As a data scientist for a private space launch company, your role in this project is to harness data science techniques to predict the success of Falcon 9 first stage landings. This involves collecting, processing, and analyzing data from various sources to build predictive models that can forecast landing outcomes. By improving the accuracy of these predictions, the company can enhance its competitive edge, optimize launch operations, and better inform strategic decisions.

Introduction Cont'd

Problems to Be Addressed

The primary objective of this project is to determine the factors that influence the successful landing of the Falcon 9 first stage. Specifically, the project aims to address the following problems:

1. **Data Collection and Quality Improvement**: How can we collect relevant and comprehensive data from multiple sources, and what methods should be employed to ensure the data is clean, accurate, and suitable for analysis?
2. **Exploratory Data Analysis**: What patterns and relationships can be identified within the collected data, and how do different variables, such as payload mass and launch site location, impact the success of the first stage landing?
3. **Predictive Modeling**: Which machine learning models are most effective in predicting the landing success of the Falcon 9 first stage, and what are the key features that these models rely on?
4. **Interactive Visualization and Insights**: How can we develop an interactive dashboard that allows stakeholders to explore launch data dynamically and gain actionable insights? What geographical and operational factors contribute to higher success rates, and how can these insights inform future launch strategies?

By addressing these questions, the project seeks to provide a comprehensive analysis of the factors affecting Falcon 9 first stage landings, ultimately contributing to the company's ability to conduct more cost-effective and reliable space missions.

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Data was collected from multiple sources, including the SpaceX API and historical launch records from Wikipedia. Data was retrieved using API requests and web scraping techniques with BeautifulSoup.
- Perform data wrangling
  - Data wrangling techniques such as cleaning, formatting, and converting data into a suitable structure for analysis were applied.
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash

Methodology

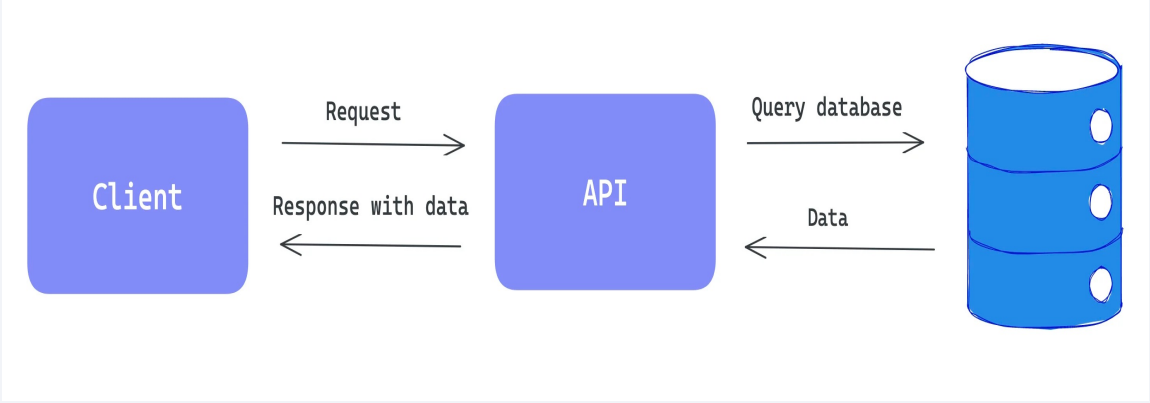
Executive Summary

- Perform predictive analysis using classification models
    - Key steps included splitting the data into training and testing sets, standardizing the data, and applying algorithms such as Support Vector Machines (SVM), classification trees, and logistic regression. Hyperparameter tuning was performed to optimize model performance, and the best-performing model was selected based on its accuracy and predictive power.

Data Collection – SpaceX API

API Request to SpaceX:

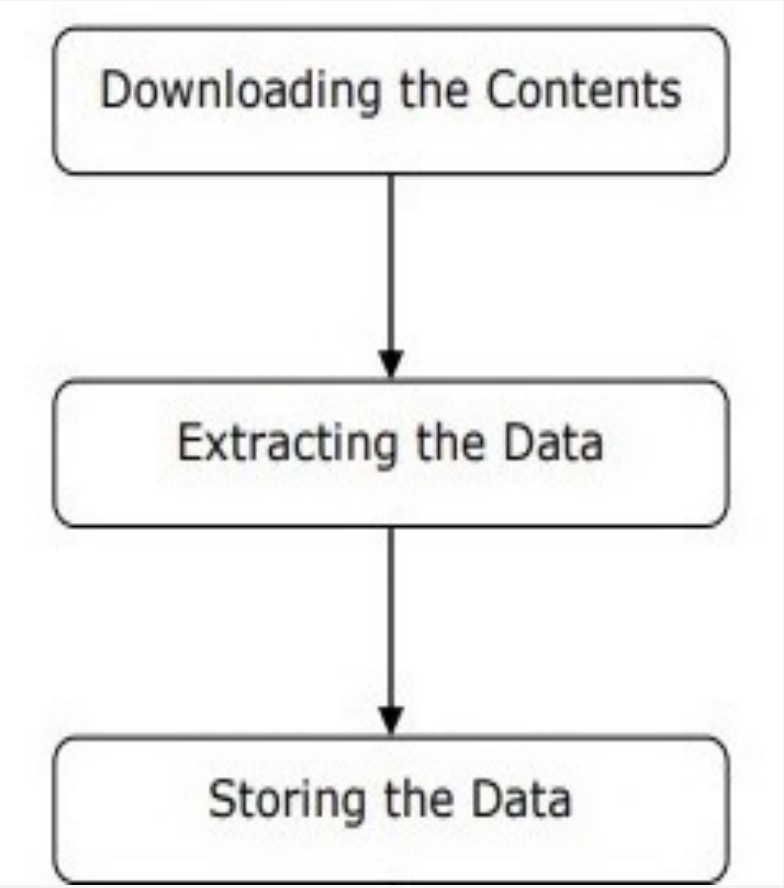
- **Send GET requests to the SpaceX API**: We start by making GET requests to the SpaceX API to collect data on Falcon 9 launches.
- **Retrieve data on Falcon 9 launches**: The API responses contain detailed information about each launch in JSON format.
- **Store data in JSON format**: The collected data is stored in JSON format for further processing.
- https://github.com/sheldongordon4/Data-Science-Capstone-IBM/blob/main/1.%20spacex-data-collection-api.ipynb



Data Collection - Scraping

Web Scraping Wikipedia:

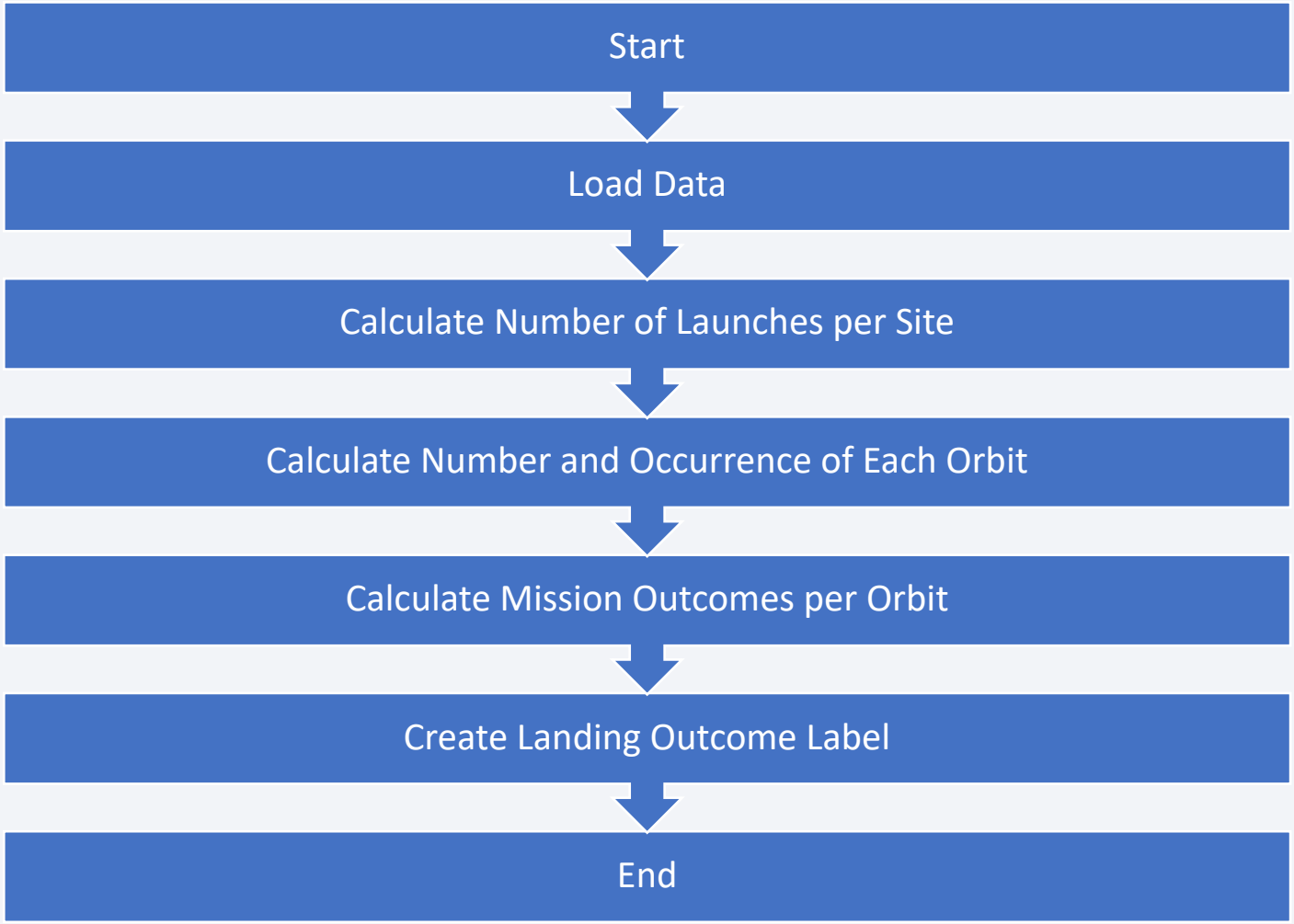
- **Identify target URL for Falcon 9 and Falcon Heavy launches**: We target the specific Wikipedia page containing the launch records.
- **Use BeautifulSoup to parse HTML content**: We use the BeautifulSoup library to parse the HTML content of the webpage.
- **Extract launch records table**: The HTML table containing the launch records is extracted.
- **Convert HTML table to Pandas DataFrame**: The extracted table is then converted into a Pandas DataFrame for further analysis.
- https://github.com/sheldongordon4/Data-Science-Capstone-IBM/blob/main/2.%20web_scraping.ipynb



Data Wrangling

- **Calculate Number of Launches per Site**
  - Group data by launch site and count number of launches per site.
- **Calculate Number and Occurrence of Each Orbit**
  - Group data by orbit type and count number of occurrences for each orbit.
- **Calculate Mission Outcomes per Orbit**
  - Group data by orbit type and mission outcome and count number of occurrences for each mission outcome.
- **Create Landing Outcome Label**
  - Define successful and unsuccessful landing outcomes and create a new column for landing outcome labels based on existing outcome data.
- https://github.com/sheldongordon4/Data-Science-Capstone-IBM/blob/main/3.%20spacex-Data%20wrangling.ipynb

Data Wrangling Flowchart



EDA with Data Visualization

1. Visualization of relationship between Flight number and launch site (bar & scatter)
2. Visualization of relationship between payload and launch site (bar & scatter)
3. Visualization of relationship between success rate and each orbit (bar & scatter)
4. Visualization of relationship between flight number and orbit type (bar & scatter)
5. Visualization of relationship between payload and orbit type (bar & scatter)
6. Visualization of yearly launch success trend

- https://github.com/sheldongordon4/Data-Science-Capstone-IBM/blob/main/5.%20edadataviz.ipynb

EDA with SQL

1. Display the names of the unique launch sites in the space mission
2. Display 5 records where launch sites begin with the string ‘CCA’
3. Display the total payload mass carried by boosters launched by NASA (CRS)
4. Display average payload mass carried by booster version F9 v1.1
5. List the date when the first successful landing outcome in ground pad was achieved
6. List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000
7. List the total number of successful and failure mission outcomes

EDA with SQL

8. List the names of the booster versions which have carried maximum payload mass
9. List the records which will display the month names, failure landing outcomes in drone ship, booster versions, launch site for the months in year 2015
10. Rank the count of landing outcomes (such as Failure (drone ship) or Success (ground pad)) between the date 2010-06-04 and 2017-03-20, in descending order

- https://github.com/sheldongordon4/Data-Science-Capstone-IBM/blob/main/4.%20eda-sql_sqllite.ipynb

Build an Interactive Map with Folium

- Generated initial site map with coordinates and zoom
- Created and added folium circle and folium marker for each launch site to be able to distinctly identify them on the map
- Created a new column in data frame caller “marker color” to store color based on class, with green being success and red being failure
- Marked the success/failed launches for each site on the map using a marker cluster to enhance map readability and appearance
- Calculated the distance between launch site and the nearest coastline, highway and city respectively
- Created and added folium marker to map and drew a line from launch site to each of the above to visualize and analyze proximities
- https://github.com/sheldongordon4/Data-Science-Capstone-IBM/blob/main/6.%20launch_site_location.ipynb

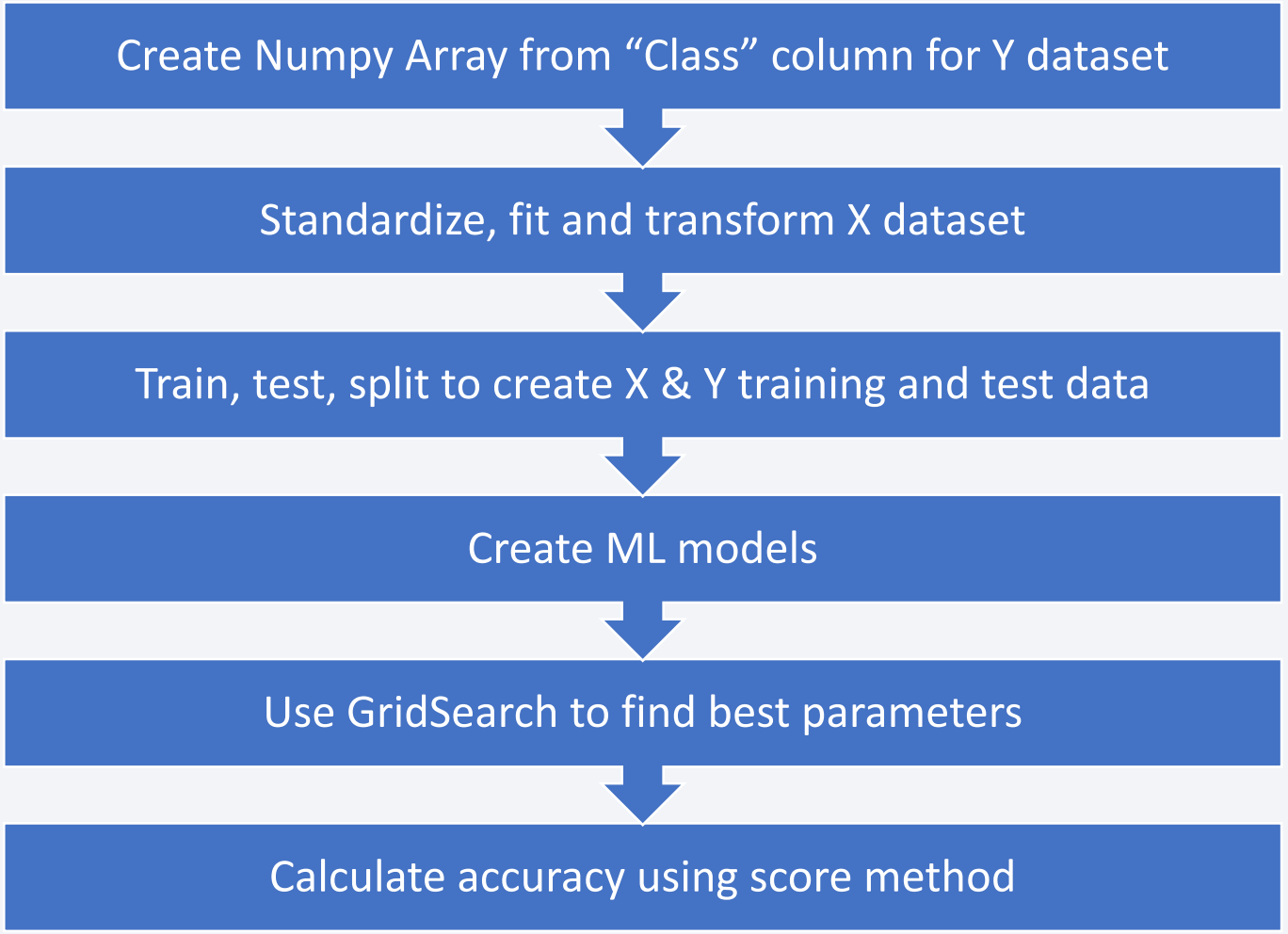
Build a Dashboard with Plotly Dash

- Built a dashboard with dropdown options for each launch site (or all)
- Displayed a pie chart showing success and failure percentage of each site to show the breakdown of launches for each site as well as a summary chart of percentage launch success contributed by each site when “All” is selected
- Displayed a slider control scatter plot of success and booster version category with slider controlling payload mass to show the booster version category success rate based on payload mass
- https://github.com/sheldongordon4/Data-Science-Capstone-IBM/blob/main/7.%20spacex_dash_app.py

Predictive Analysis (Classification)

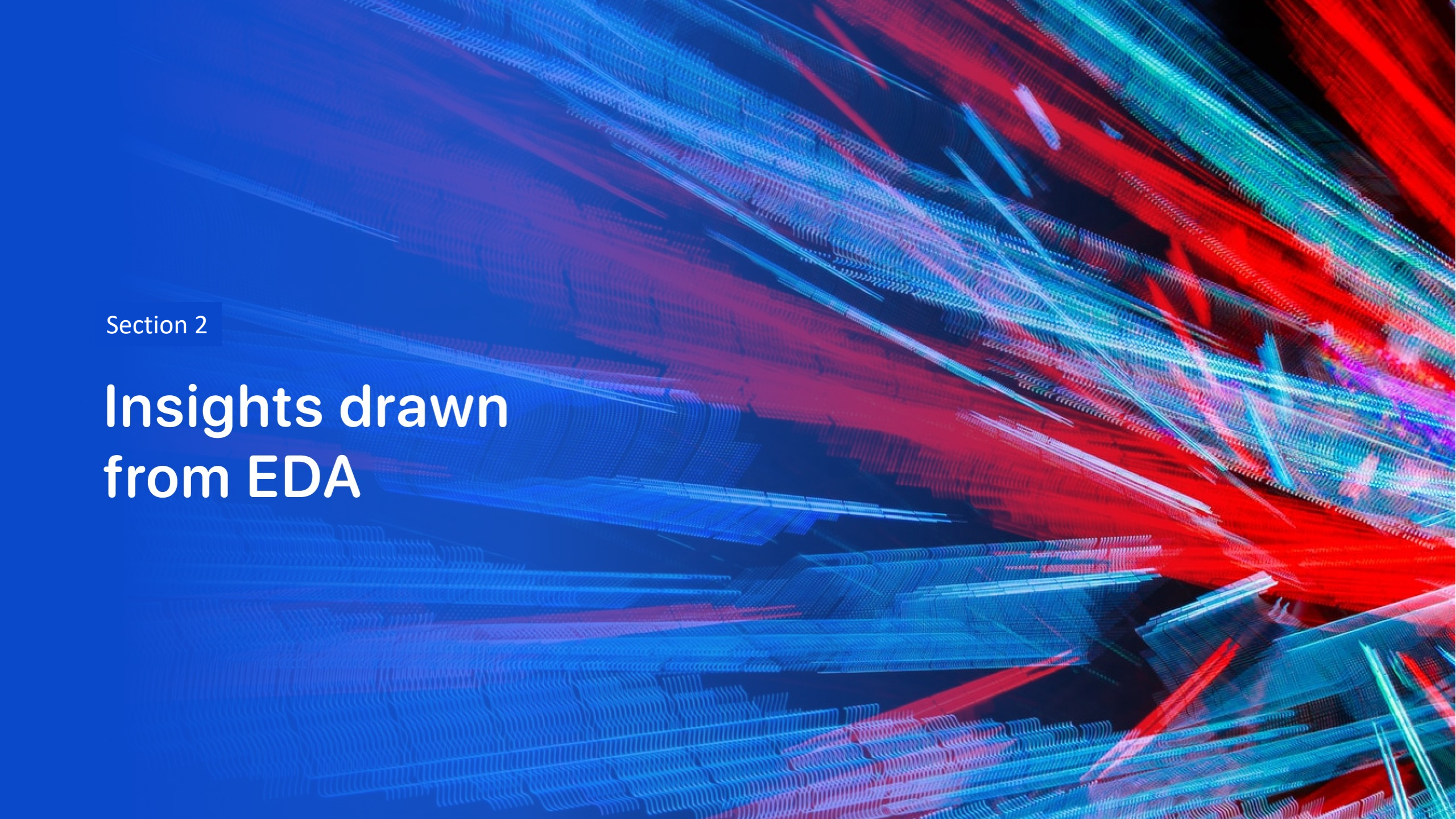
- Created a Numpy array from the “Class” column of the data
- Standardized, fit and transformed X data
- Used the function train_test_split to create X and Y training and test datasets
- Created logistic regression, support vector machine, decision tree and k nearest neighbor models
- Used GridSearch to find the best parameters for each model as well as used the score method to calculate accuracy of each
- https://github.com/sheldongordon4/Data-Science-Capstone-IBM/blob/main/8.%20SpaceX_Machine%20Learning%20Prediction.ipynb

Predictive Analysis Flowchart



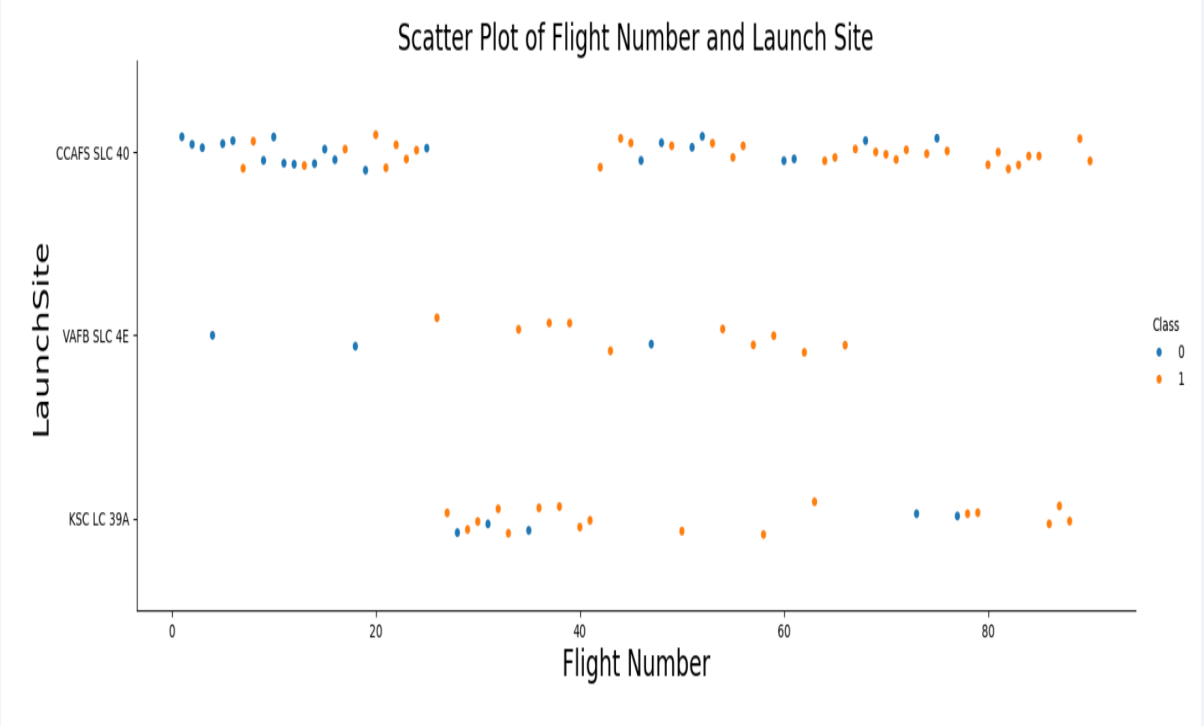
Results

- Exploratory data analysis results
- Interactive analytics demo in screenshots
- Predictive analysis results


Section 2
Insights drawn from EDA

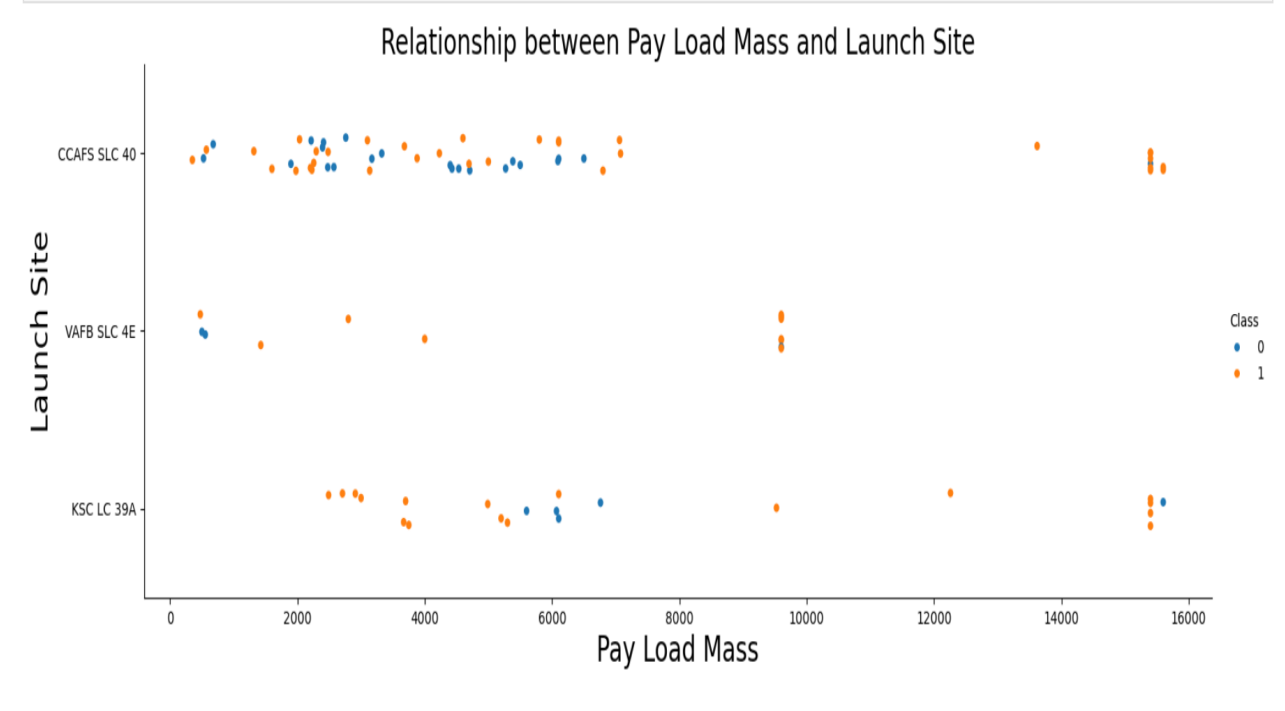
Flight Number vs. Launch Site

- Launch success generally improved with the number of launches
- The CCAFS SLC-40 site had most of the initial launches and as such a higher failure rate compared with other sites, especially for early launches as most of the learning was done here
- Subsequent launches were more successful for all sites thereafter

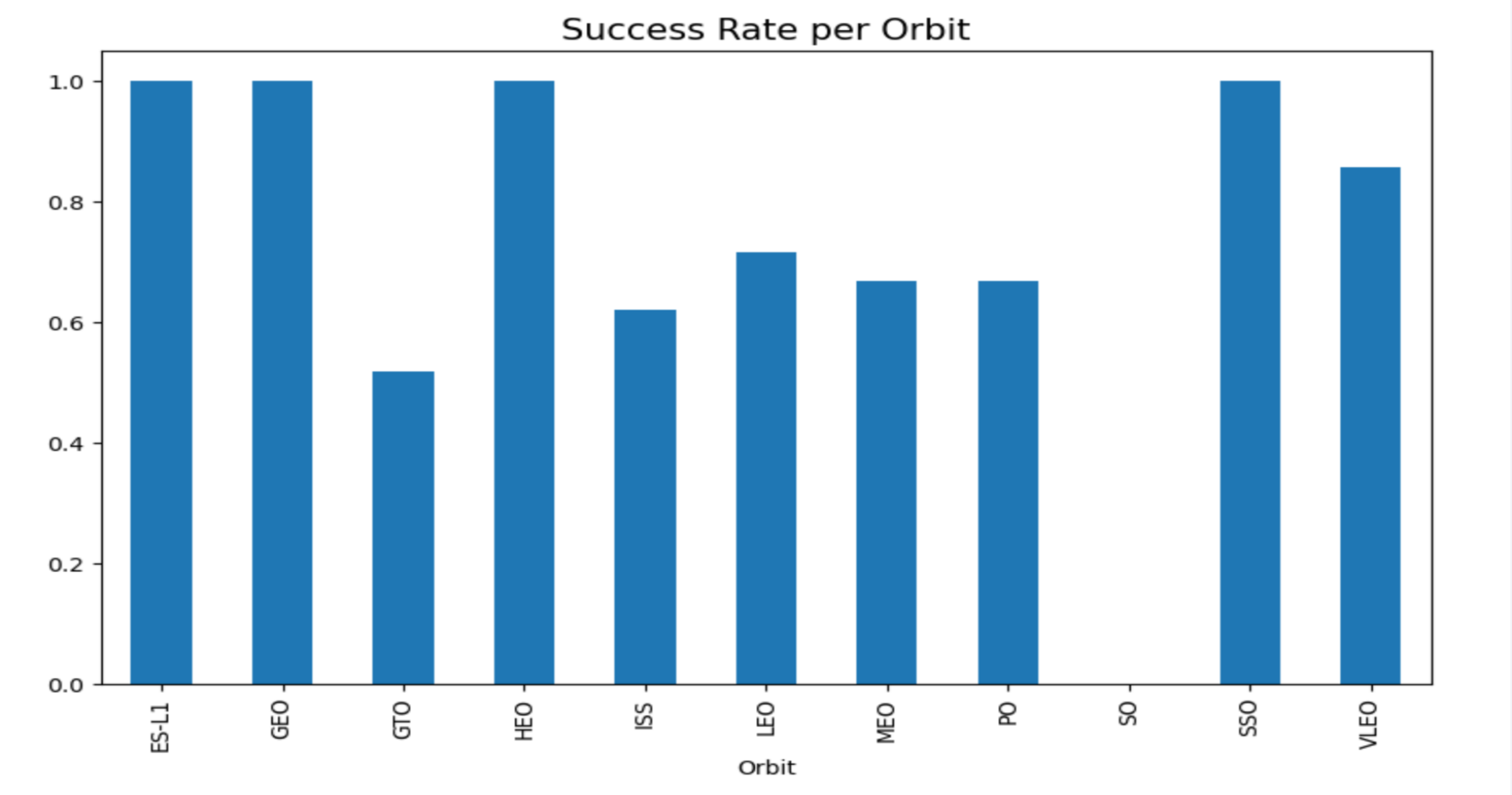


Payload vs. Launch Site

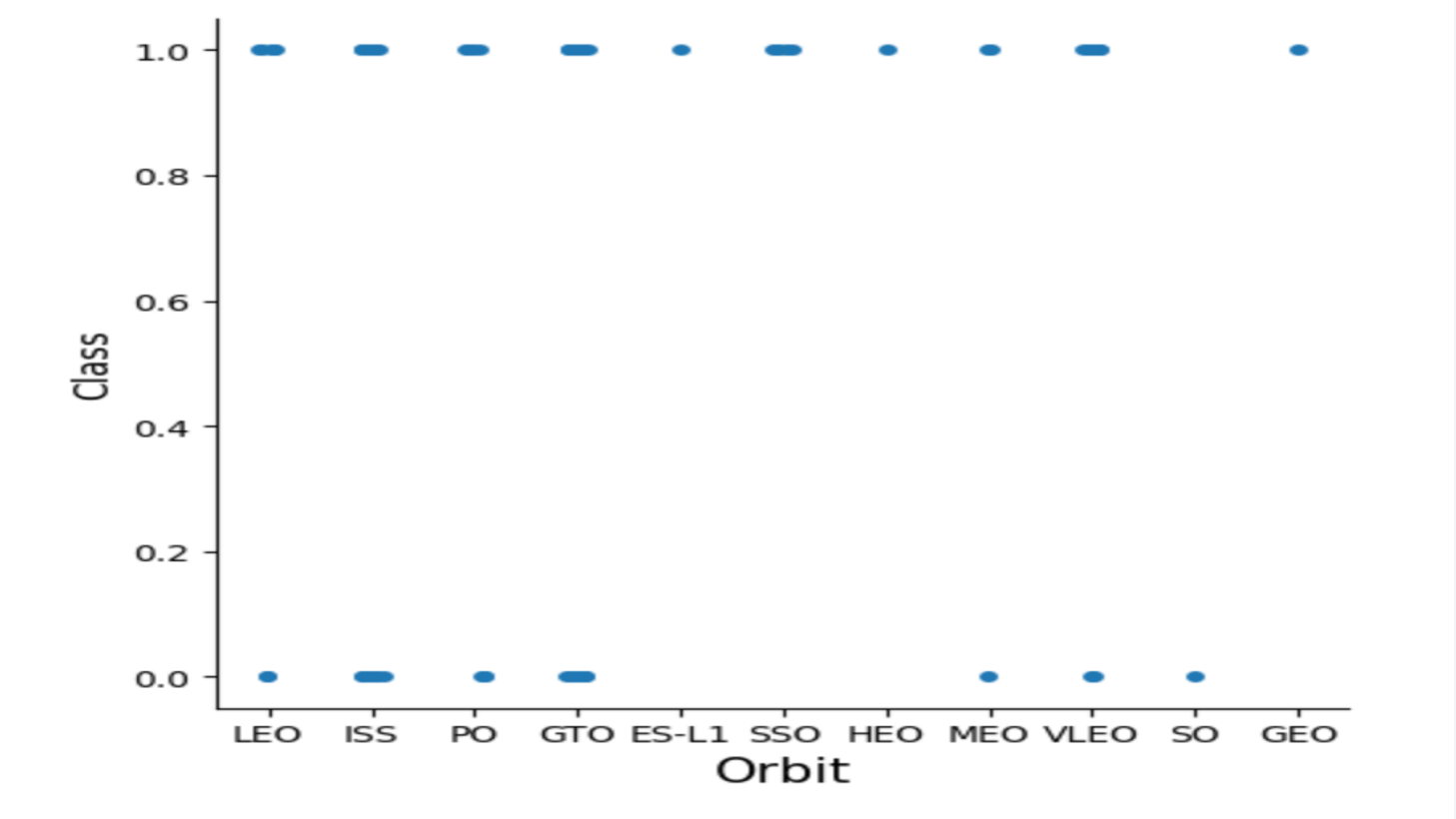
- Launch success is highest for payload masses greater than 7000 kg
- The CCAFS SLC-40 had the greatest payload range
- The other sites (VAFB SLC-4E and KSC LC-39A) had higher success rates at various payload masses



Success Rate vs. Orbit Type

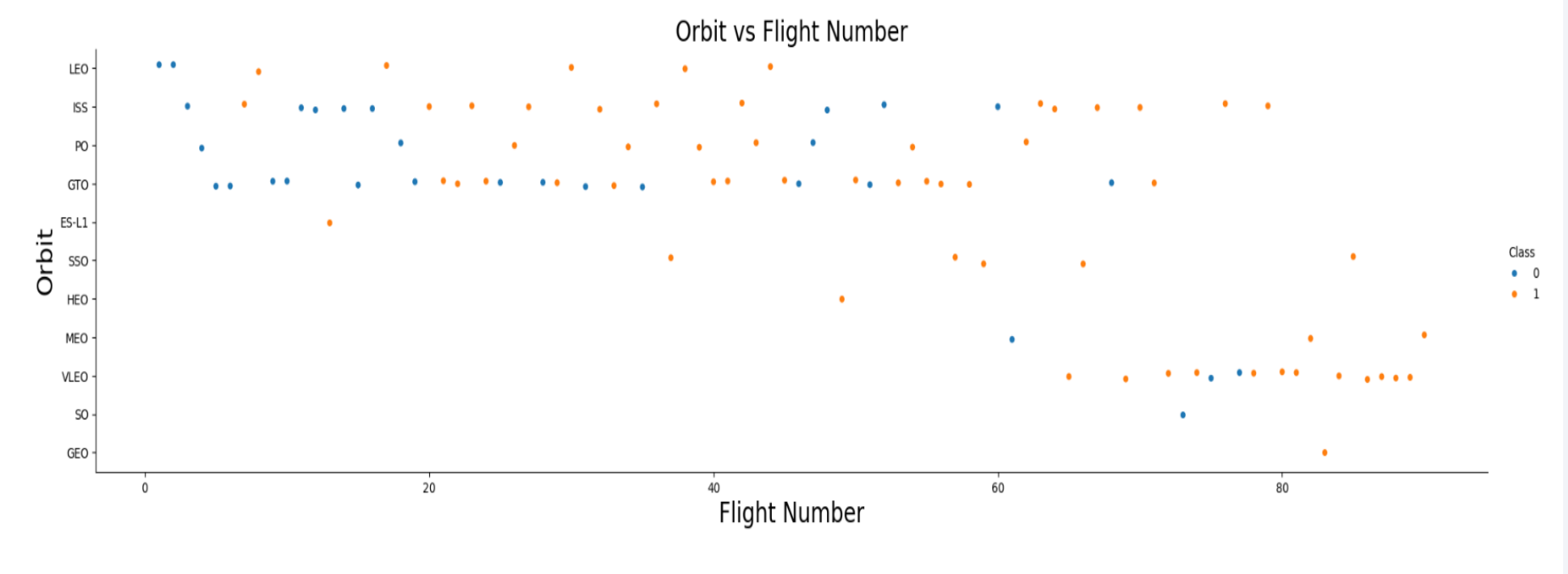


Success Rate vs. Orbit Type



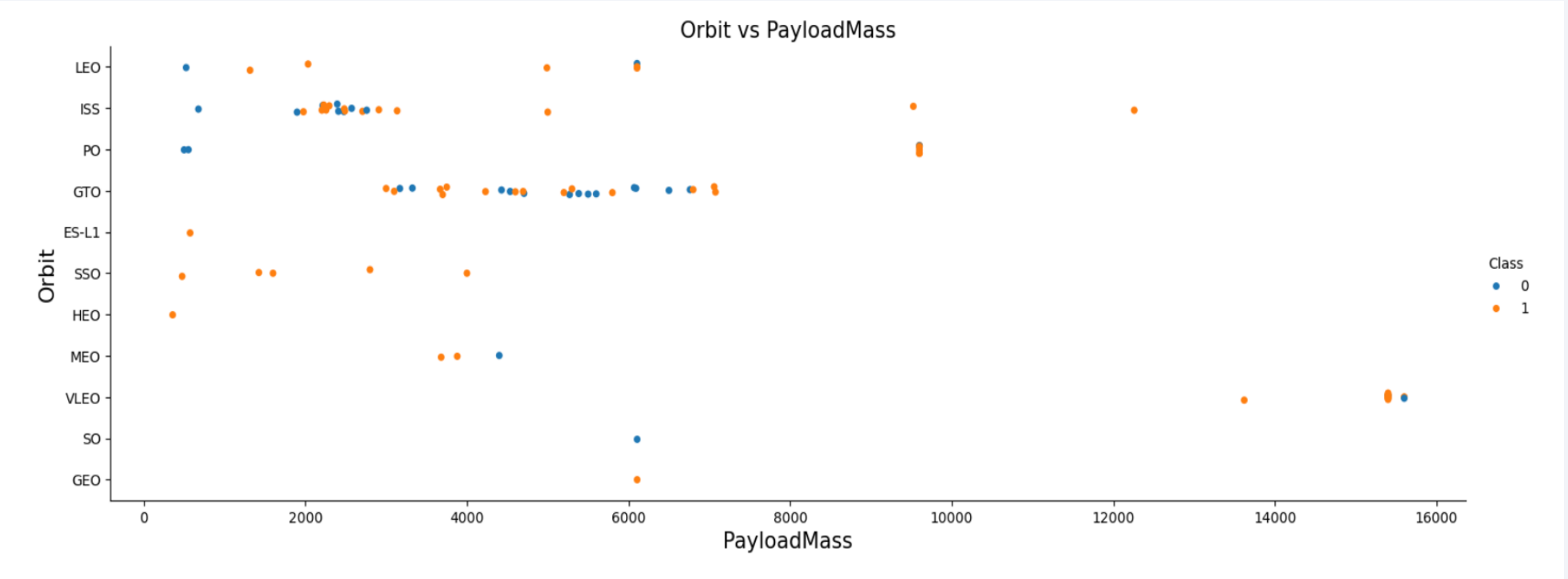
ES-L1, SSO, HEO and GEO orbit types have 100% success rates, while all others have mixed rates of success.

Flight Number vs. Orbit Type



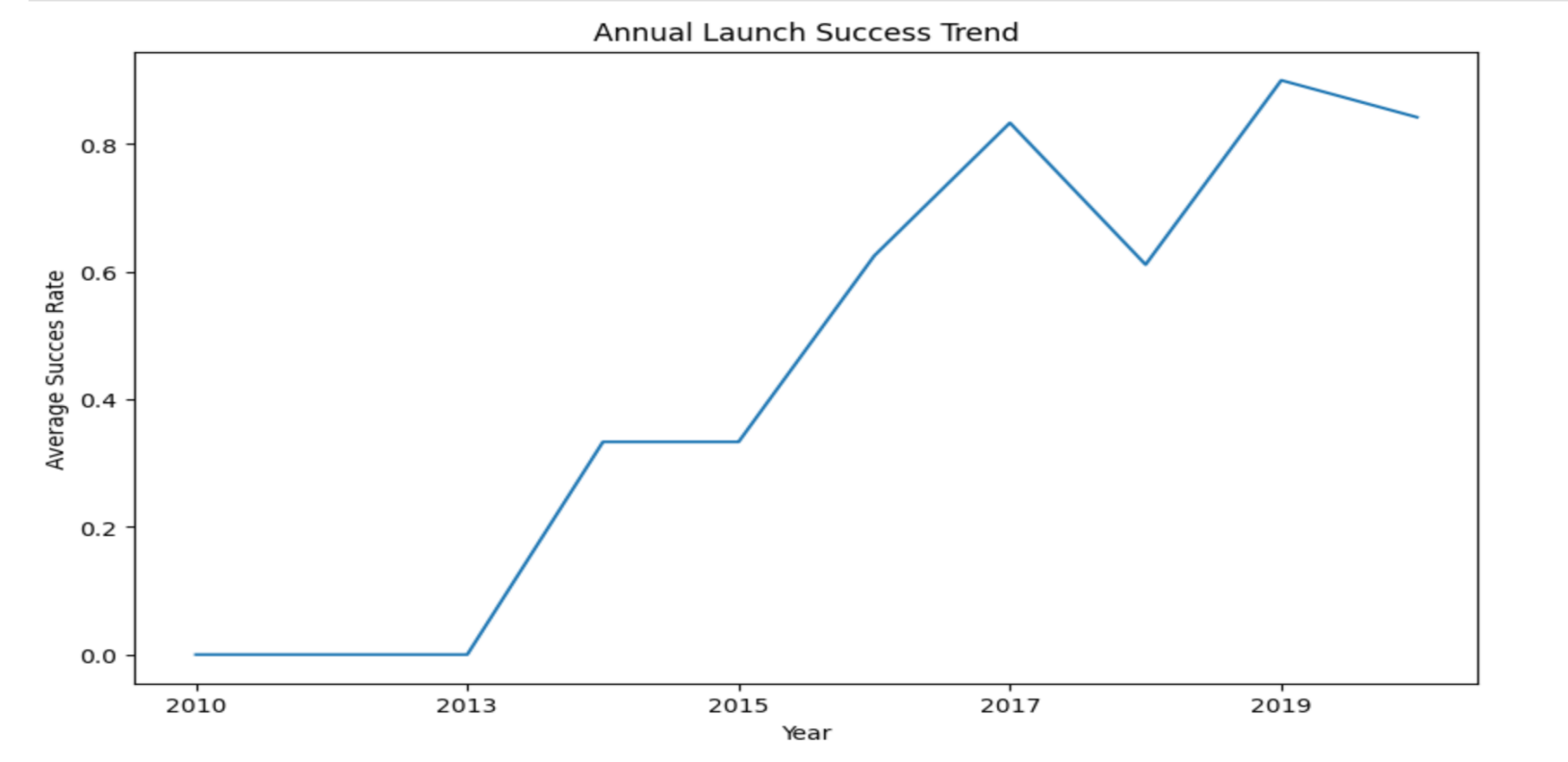
- GEO, ES-L1 and HEO each have a single, successful launch
- SSO has had a total of five launches with 100% success
- ISS and VLEO have both a high number of launches as well as high success rates
- LEO had the first two launches fail but all five subsequent launches have been successful

Payload vs. Orbit Type



- ISS, PO and VLEO have had successful launches with payloads greater than 7000 kg
- SSO has had 100% launch success rate, however, with payloads lower than 5000 kg
- GTO has the highest payload range with mixed success rate

Launch Success Yearly Trend



- Launch success rate has had a general upward trend from 2010 to 2020 with a slight dip in 2018, which was reversed in 2019.
- Success rate increased from zero to 85% in 2020 with a high of about 90% in 2019.

All Launch Site Names

Find the names of the unique launch sites

- Use “distinct” clause to ensure that unique sites were displayed

In [11]:
```
%sql SELECT DISTINCT "Launch_Site" FROM SPACEXTABLE
```

```
 * sqlite:///my_data1.db
Done.
```

Out[11]:

Launch_Site
CCAFS LC-40
VAFB SLC-4E
KSC LC-39A
CCAFS SLC-40

Launch Site Names Begin with 'CCA'

Find 5 records where launch sites begin with `CCA`

- Use the “like” clause to specify sites that have ‘CCA’ at the beginning of their names

In [12]:
```
%sql SELECT * FROM SPACEXTABLE WHERE "Launch_Site" LIKE 'CCA%' LIMIT 5
```

```
 * sqlite:///my_data1.db
Done.
```

Out[12]:

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer
2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX
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
2012-05-22	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)
2012-10-08	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)
2013-03-01	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)

Total Payload Mass

Calculate the total payload carried by boosters from NASA

- Use the “sum” function to calculate total as well as the “like” clause to get all sites with “NASA” in their name

```
In [13]: %sql SELECT SUM(PAYLOAD_MASS__KG_) FROM SPACEXTABLE WHERE Customer LIKE 'NASA%'
 * sqlite:///my_data1.db
Done.
Out[13]: SUM(PAYLOAD_MASS__KG_)
---------------------------
                       99980
```

Average Payload Mass by F9 v1.1

Calculate the average payload mass carried by booster version F9 v1.1

- Use the “average” function to calculate average payload as well as the “where” clause to single out the “F9 v1.1 booster version

```
In [14]: %sql SELECT AVG(PAYLOAD_MASS__KG_) FROM SPACEXTABLE WHERE Booster_Version = "F9 v1.1"
 * sqlite:///my_data1.db
Done.
Out[14]: AVG(PAYLOAD_MASS__KG_)
         2928.4
```

First Successful Ground Landing Date

Find the dates of the first successful landing outcome on ground pad

- Use the “min” function to specify the minimum or first date as well as the “like” clause to get landing outcome with “ground” in their name

```
In [15]: %sql SELECT MIN(Date) FROM SPACEXTABLE WHERE Landing_Outcome LIKE '%ground%'

 * sqlite:///my_data1.db
Done.

Out[15]: MIN(Date)
         2015-12-22
```

Successful Drone Ship Landing with Payload between 4000 and 6000

List the names of boosters which have successfully landed on drone ship and had payload mass greater than 4000 but less than 6000

Full query: %sql SELECT Booster_Version FROM SPACEXTABLE WHERE Landing_Outcome LIKE '%Success (drone%' AND PAYLOAD_MASS__KG_ > 4000

- Use the “where” clause to specify landing outcome as well as the “like” clause to limit search to those starting with “Success (drone)” and with payload greater than 4000 kg

```
In [16]: %sql SELECT Booster_Version FROM SPACEXTABLE WHERE Landing_Outcome LIKE '%Success (drone
 * sqlite:///my_data1.db
Done.
```

Out[16]:

Booster_Version
F9 FT B1022
F9 FT B1026
F9 FT B1029.1
F9 FT B1021.2
F9 FT B1036.1
F9 B4 B1041.1
F9 FT B1031.2

Total Number of Successful and Failure Mission Outcomes

Calculate the total number of successful and failure mission outcomes

- Use ”count” function as well as “like” clause to limit results to mission outcomes starting with “Suc”

```
In [17]: %sql SELECT COUNT(Mission_Outcome) FROM SPACEXTABLE WHERE Mission_Outcome LIKE 'Suc%'
 * sqlite:///my_data1.db
Done.
Out[17]: COUNT(Mission_Outcome)
                              100
```

Boosters Carried Maximum Payload

List the names of the booster which have carried the maximum payload mass

Full query: %sql Select BoosterVersion From SPACEXTABLE WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTABLE)

- Use a nested query of “max” function on payload mass in the “where” clause to specify the maximum payload mass

```
In [18]: %sql Select Booster_Version From SPACEXTABLE WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLO
 * sqlite:///my_data1.db
Done.
```

Out[18]:

Booster_Version
F9 B5 B1048.4
F9 B5 B1049.4
F9 B5 B1051.3
F9 B5 B1056.4
F9 B5 B1048.5
F9 B5 B1051.4
F9 B5 B1049.5
F9 B5 B1060.2
F9 B5 B1058.3
F9 B5 B1051.6
F9 B5 B1060.3
F9 B5 B1049.7

2015 Launch Records

List the failed landing outcomes in drone ship, their booster versions, and launch site names for in year 2015

Full query: %sql SELECT CASE substr(Date, 6, 2) WHEN '01' THEN 'January' WHEN '02' THEN 'February' WHEN '03' THEN 'March' WHEN '04' THEN 'April' WHEN '05' THEN 'May' WHEN '06' THEN 'June' WHEN '07' THEN 'July' WHEN '08' THEN 'August' WHEN '09' THEN 'September' WHEN '10' THEN 'October' WHEN '11' THEN 'November' WHEN '12' THEN 'December' END AS Month_Name, Landing_Outcome, Booster_Version, Launch_Site FROM SPACEXTABLE WHERE Landing_Outcome LIKE 'Failure (dro%' LIMIT 2

- Use the ”substr” function to get the month number from the ”date” column and use the “case” clause to assign the month name to the month number. Use “limit” clause to get the top 2 results

```
In [19]: %sql SELECT CASE substr(Date, 6, 2) WHEN '01' THEN 'January' WHEN '02' THEN 'February' W
 * sqlite:///my_data1.db
Done.
```

Out[19]:

Month_Name	Landing_Outcome	Booster_Version	Launch_Site
January	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
April	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

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

- Rank the count of landing outcomes (such as Failure (drone ship) or Success (ground pad)) between the date 2010-06-04 and 2017-03-20, in descending order
- **%sql** SELECT Landing_Outcome, COUNT(Landing_Outcome) AS outcome_count FROM SPACEXTABLE WHERE DATE BETWEEN '2010-06-04' AND '2017-03-20' GROUP BY Landing_Outcome ORDER BY outcome_count DESC;
- Use the “count” function to get the number of each outcome then use the ”between” clause to specify the time period. Finally, use “groupby” and “order by” to group and rank.

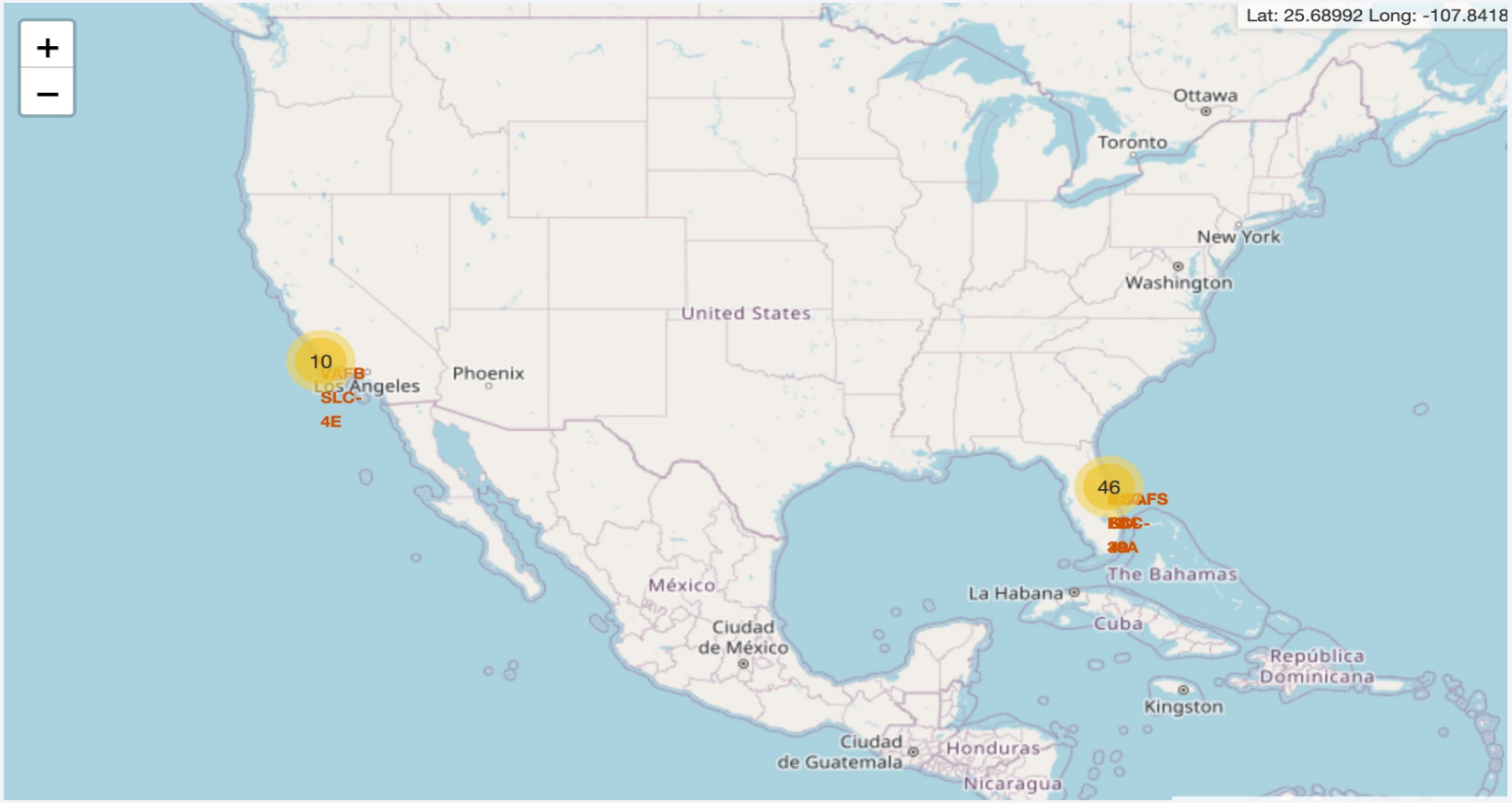
```
In [11]: %sql SELECT Landing_Outcome, COUNT(Landing_Outcome) AS outcome_count FROM SPACEXTABLE WH
 * sqlite:///my_data1.db
Done.
```

Out[11]:

Landing_Outcome	outcome_count
No attempt	10
Success (drone ship)	5
Failure (drone ship)	5
Success (ground pad)	3
Controlled (ocean)	3
Uncontrolled (ocean)	2
Failure (parachute)	2
Precluded (drone ship)	1

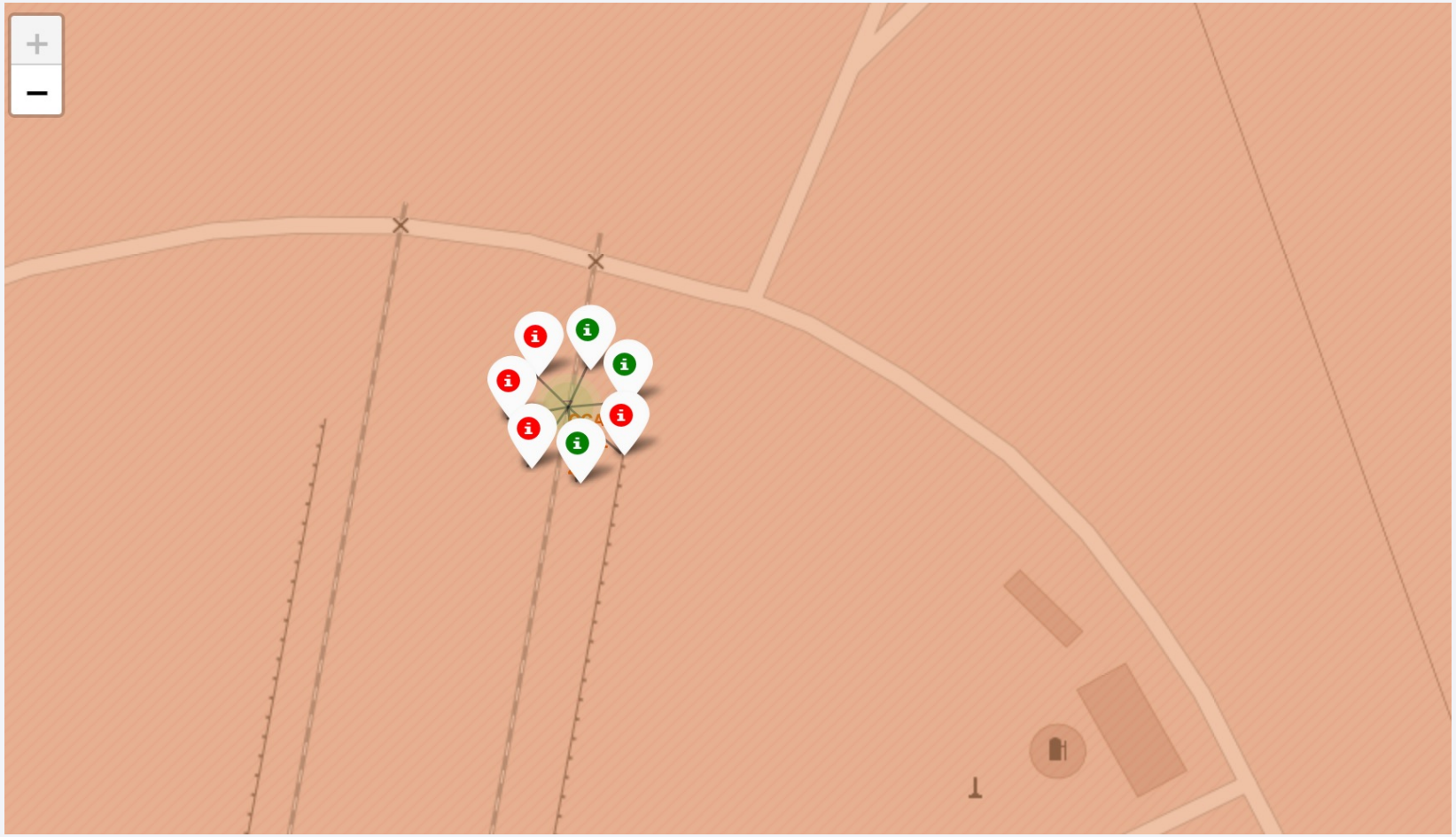

Section 3
Launch Sites
Proximities Analysis

Launch Site Map with Outcome Markers



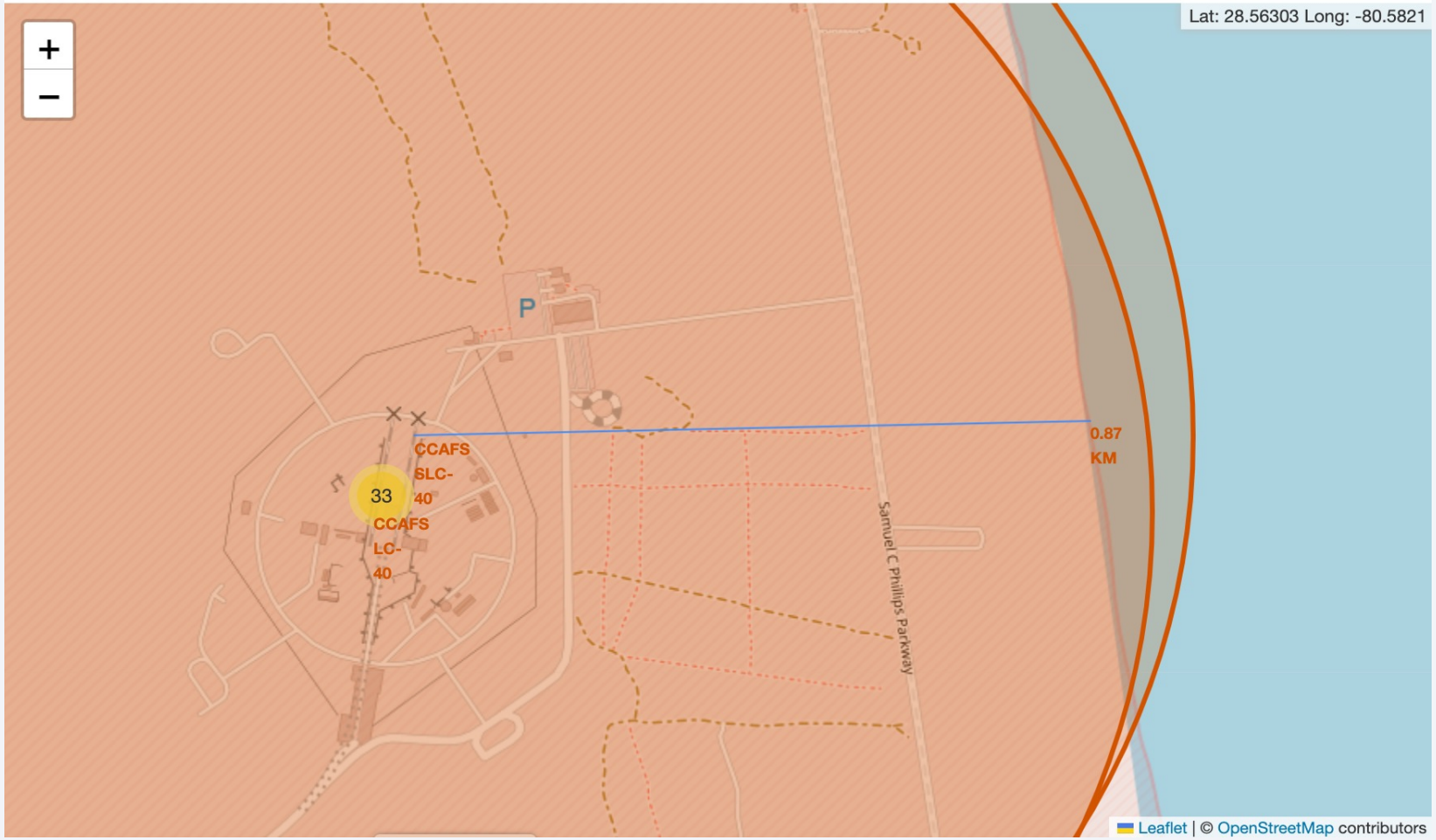
The map above shows the launch site markers as well marker cluster with the number of successful as well as failed launched for each site.

Launch Site Map with color-labelled outcomes



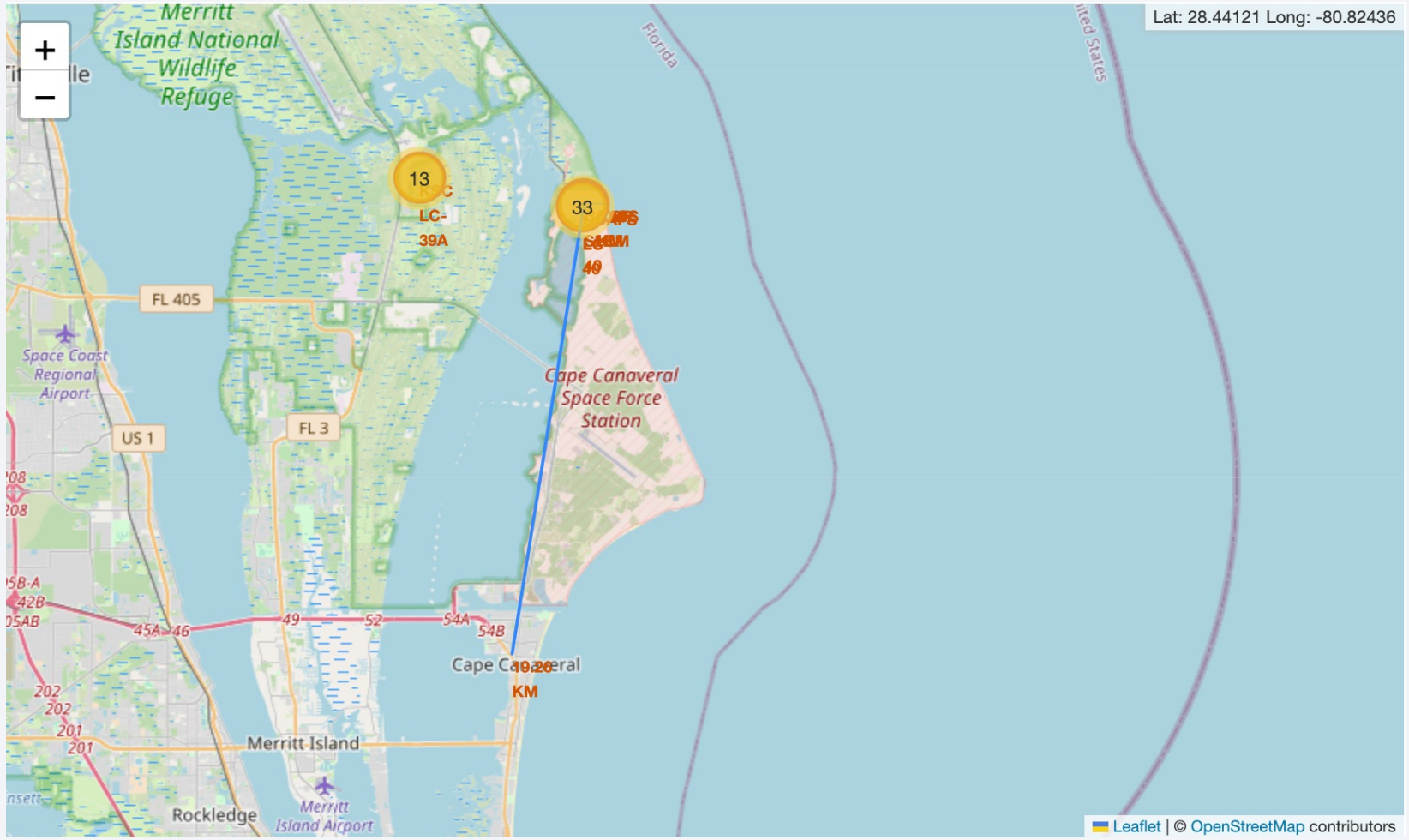
The map above shows the close-up view of a launch site with green markers for successful and red markers for failed launches.

Launch Site Proximity to Coastline

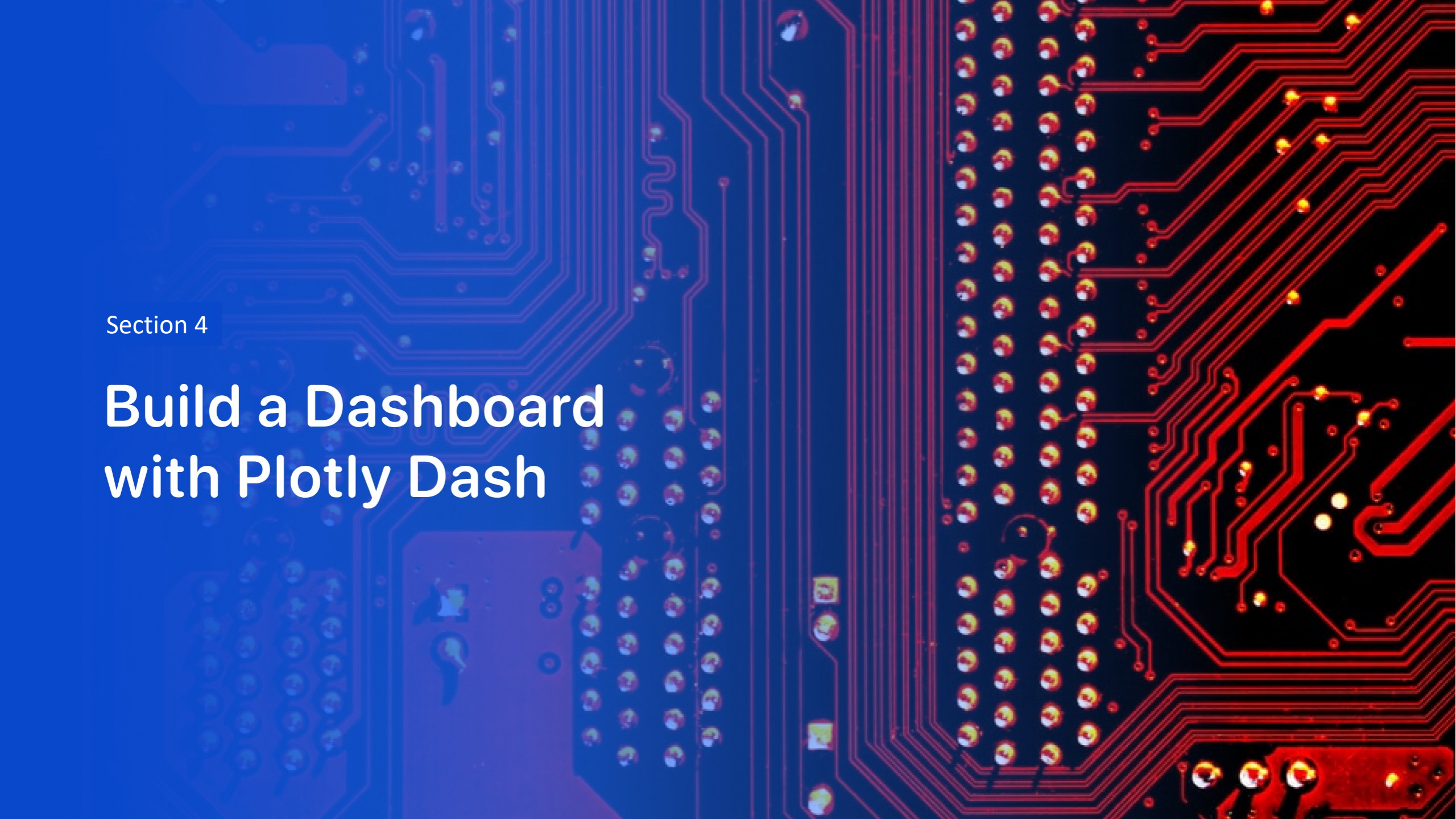


The map above shows the proximity of a launch site to the coastline, connected by a line and displaying the distance between the two points.

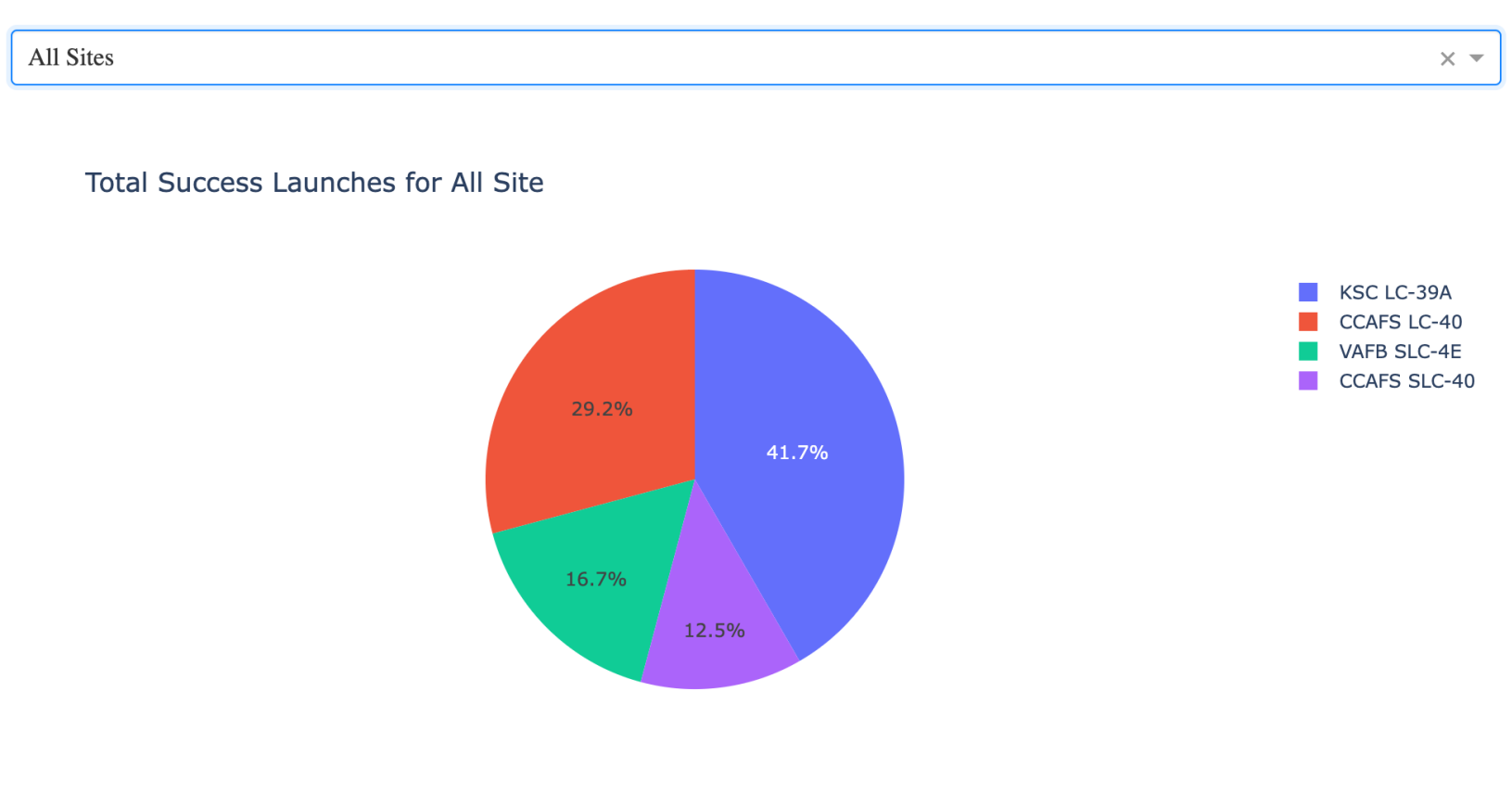
Launch Site Proximity to Nearest City



The map above shows the proximity of a launch site to the nearest city, connected by a line and displaying the distance between the two points.

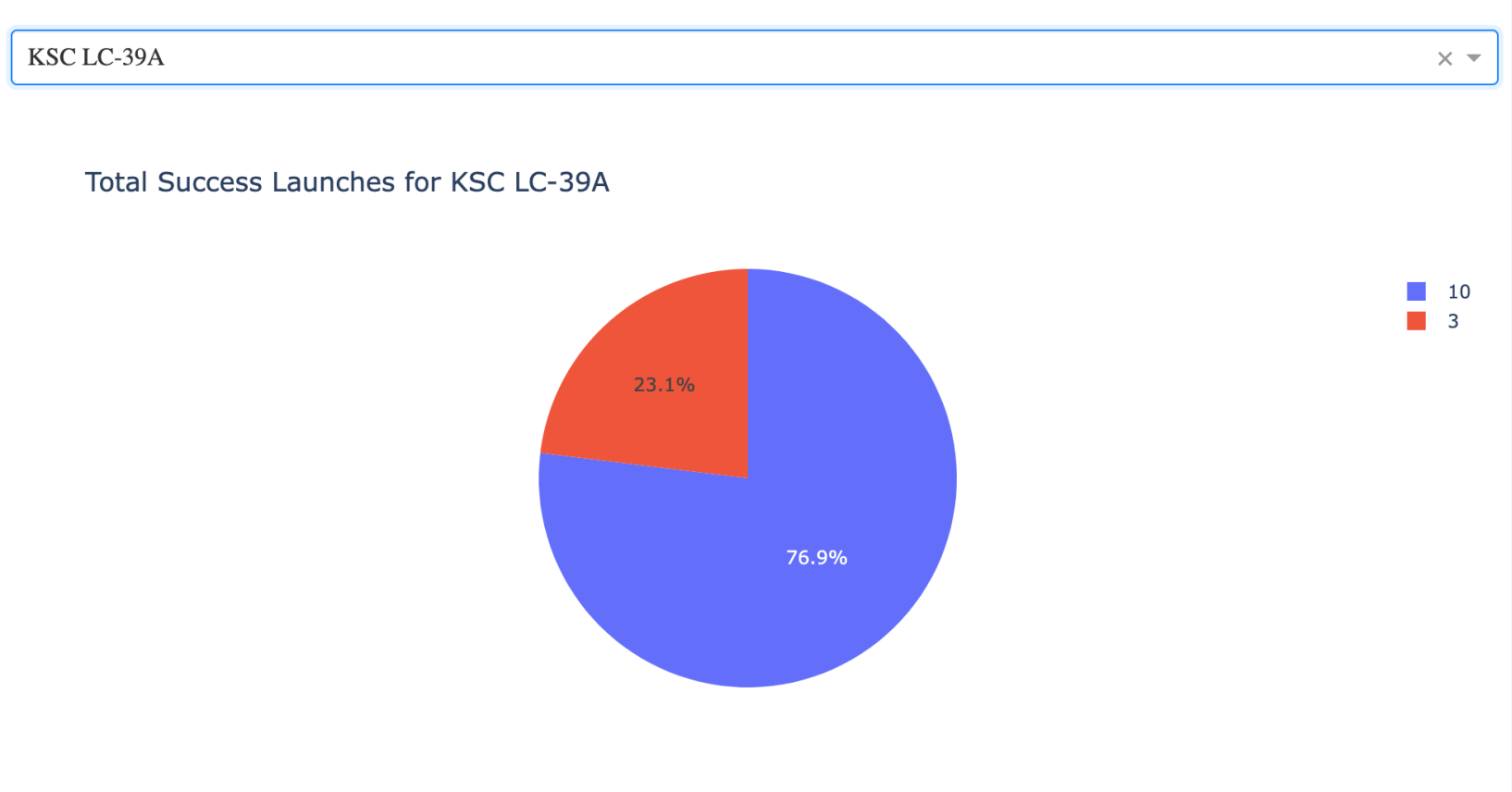

Section 4
Build a Dashboard with Plotly Dash

Pie Chart showing Success Contribution of All Sites



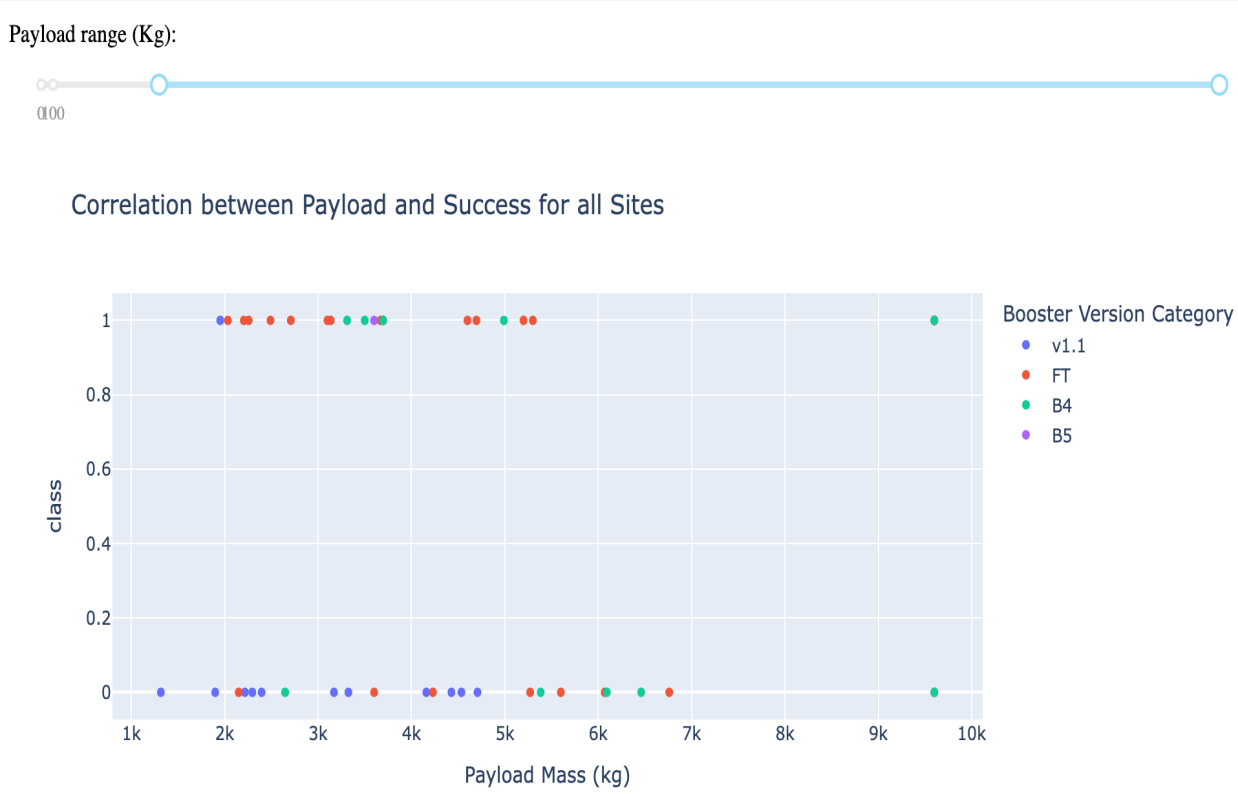
The pie chart above shows the contribution of all launch sites to total successful launches by SpaceX.

Pie Chart showing Site with Highest Success Ratio



The pie chart above shows the launch site (KSC LC-39A) with the highest launch success ratio.

Scatter Chart showing Payload Mass vs. Booster Version Success

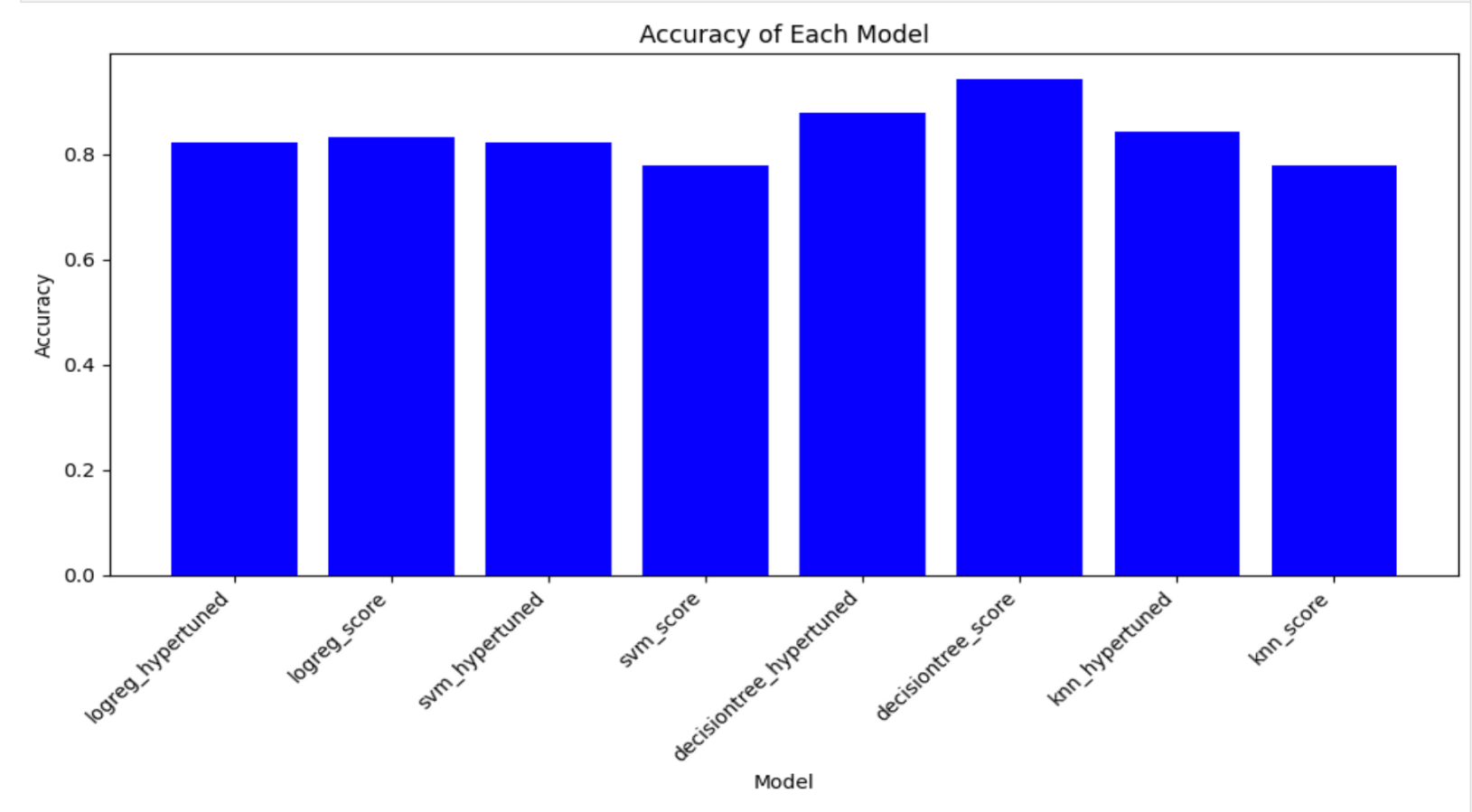


The scatter charts above shows payload range and success of each booster version at different payload masses.

Section 5

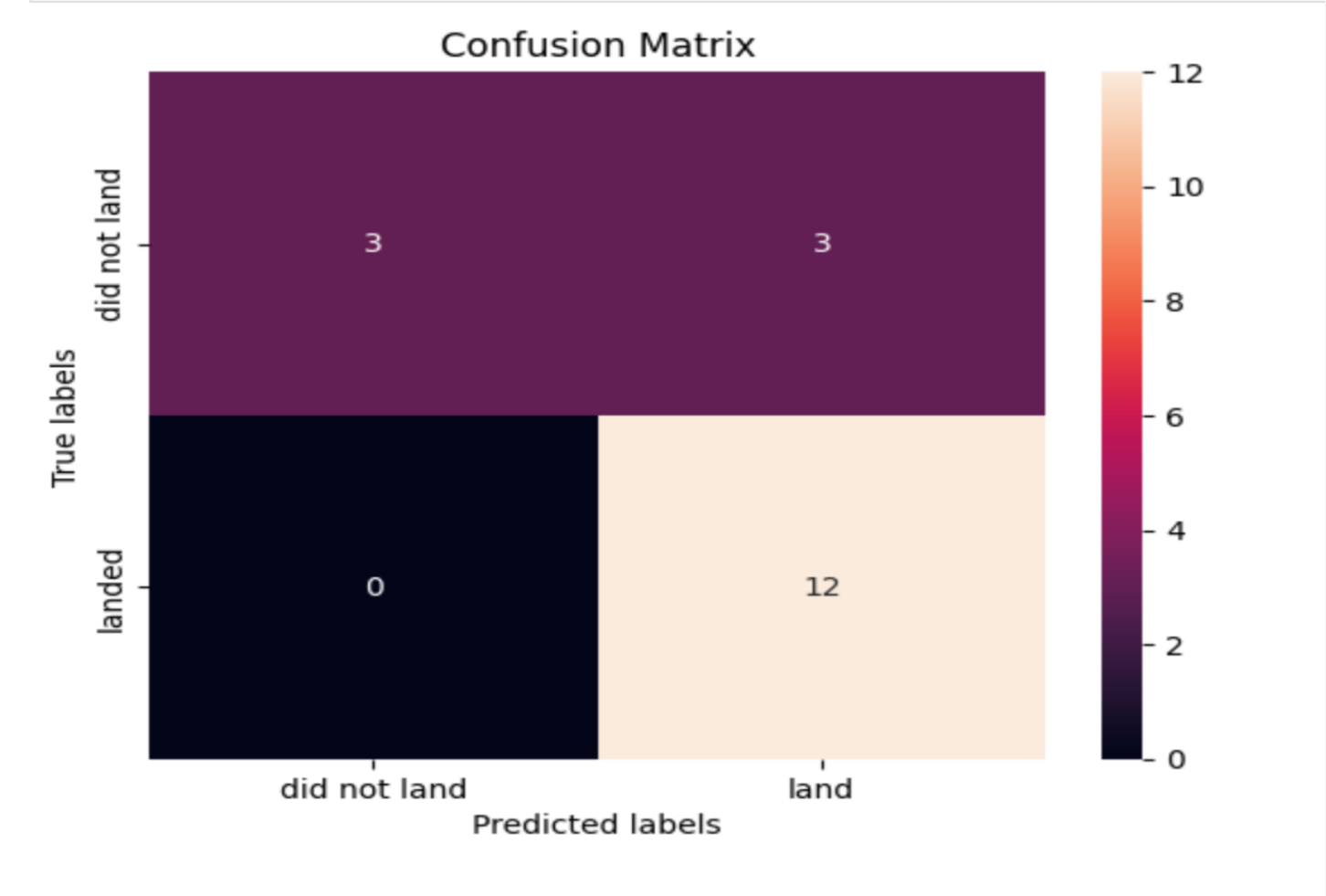
Predictive Analysis (Classification)

Classification Accuracy



The bar graph above shows the accuracy of each machine learning model that was developed with decision tree model performing the best.

Confusion Matrix of Decision Tree Model



The confusion matrix above shows 12 true positives (correctly predicted), 3 true negatives (correctly predicted), 3 false positives (incorrectly predicted) and 0 false negatives (incorrectly predicted).

Conclusions

- Launch success is generally greater for higher payload masses.
- Orbit type and booster version are also a very important factors in launch success.
- Success rate increased with time and number of flights due to constant learning and feedback.
- Launch sites are typically located several kilometers from cities while being close to coastlines.
- The most successful launch site is KSC LC-39A with a success rate of 76.9%.
- Using the decision tree model, we can predict, with an 88% accuracy the successful outcome of launches.

Appendix

Appendix

First, let's perform an HTTP GET method to request the Falcon9 Launch HTML page, as an HTTP response.

In [5]:
```
# use requests.get() method with the provided static_url
# assign the response to a object

data = requests.get(static_url).text
```

Create a `BeautifulSoup` object from the HTML `response`

In [6]:
```
# Use BeautifulSoup() to create a BeautifulSoup object from a response text content

soup = BeautifulSoup(data, 'html.parser')
```

Print the page title to verify if the `BeautifulSoup` object was created properly

In [7]:
```
# Use soup.title attribute

soup.title
```

Out[7]:
```
<title>List of Falcon 9 and Falcon Heavy launches - Wikipedia</title>
```

Web scarping snippet. Also, the url used for web scraping:

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

Appendix

The data contains several Space X launch facilities: Cape Canaveral Space Launch Complex 40 **VAFB SLC 4E** , Vandenberg Air Force Base Space Launch Complex 4E **(SLC-4E)**, Kennedy Space Center Launch Complex 39A **KSC LC 39A** .The location of each Launch Is placed in the column `LaunchSite`

Next, let's see the number of launches for each site.

Use the method `value_counts()` on the column `LaunchSite` to determine the number of launches on each site:

In [12]:
```
# Apply value_counts() on column LaunchSite

launch_sites = df['LaunchSite'].value_counts()

launch_sites
```

Out[12]:
```
CCAFS SLC 40    55
KSC LC 39A      22
VAFB SLC 4E     13
Name: LaunchSite, dtype: int64
```

Thank you!