IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Pumulo Sikaneta
May 15th, 2025

Outline

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

Executive Summary

- Summary of methodologies
  - Data Collection
  - Data Wrangling
  - Exploratory Data Analysis with Data Visualization
  - Exploratory Data Analysis with SQL
  - Interactive Visual Analytics and Dashboard
    Predictive Analysis
- Summary of all results
  - Exploratory Data Analysis Results
  - Interactive Analysis (Screenshots and links)
  - Predictive Analysis Results

Introduction

Project background and context

The commercial space landscape is rapidly evolving, with companies like Virgin Galactic, Rocket Lab, and Blue Origin contributing to diverse market segments. SpaceX's pioneering reusable Falcon 9 technology has yielded significant launch cost efficiencies (approximately $62M versus the industry average of $165M+), underpinning their success in critical areas such as ISS logistics, satellite internet deployment (Starlink), and human spaceflight.

For "Space Y" to effectively compete, a deep understanding of the factors influencing SpaceX's first-stage recovery – the key to their cost advantage – is paramount. This capstone project will employ advanced data analytics and machine learning techniques to analyze publicly available data and develop a predictive model for SpaceX's reuse decisions. This analysis will directly inform Space Y's strategic pricing initiatives.

Problems you want to find answers

1. What is the correlation between mission parameters (e.g., payload mass, launch site, flight number, orbit) and the probability of successful first-stage landing?
2. Is there a discernible trend of increasing landing success rates over time for SpaceX? What is the most effective binary classification algorithm for predicting landing outcomes in this context?

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - To prepare the acquired data for robust analysis, a comprehensive data wrangling phase was undertaken. This encompassed: the application of filtering techniques to isolate pertinent data subsets, the systematic management of missing values through appropriate imputation or removal strategies, and the implementation of one-hot encoding to convert categorical features into a numerical representation suitable for binary classification modeling.
- Perform data wrangling
  - The data wrangling process involved critical steps to ensure data quality and suitability for machine learning. This included filtering irrelevant data points, strategically addressing missing values to maintain analytical integrity, and employing one-hot encoding to transform categorical variables into a format compatible with binary classification algorithms.

Methodology (Continued)

Executive Summary

- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
    - Following data preparation, predictive analysis was performed to determine the likelihood of successful first-stage landings. This involved a systematic process of building, tuning, and evaluating various classification models, including algorithms like logistic regression and random forests, chosen for their suitability in binary classification. Rigorous hyperparameter tuning, employing techniques such as grid search, was conducted to optimize model performance, with metrics like precision and AUC closely monitored. The tuned models were then evaluated using cross-validation to ensure robust generalization. The final model selection was based on a comprehensive assessment of predictive accuracy, stability, and interpretability, ultimately aiming to provide a reliable prediction of first-stage landing success

Data Collection

To gather comprehensive data on SpaceX Falcon 9 launches, a dual approach was employed, combining the structured data available through the SpaceX REST API with supplementary information obtained via web scraping from Wikipedia. This combined methodology ensured a complete dataset for detailed analysis.

Data Collection – SpaceX API

Data was requested from the SpaceX API using a series of key phrase-based calls.

The response content was decoded using .json() and transformed into a Pandas DataFrame using .json_normalize().

Custom functions were applied to extract specific information about each launch.

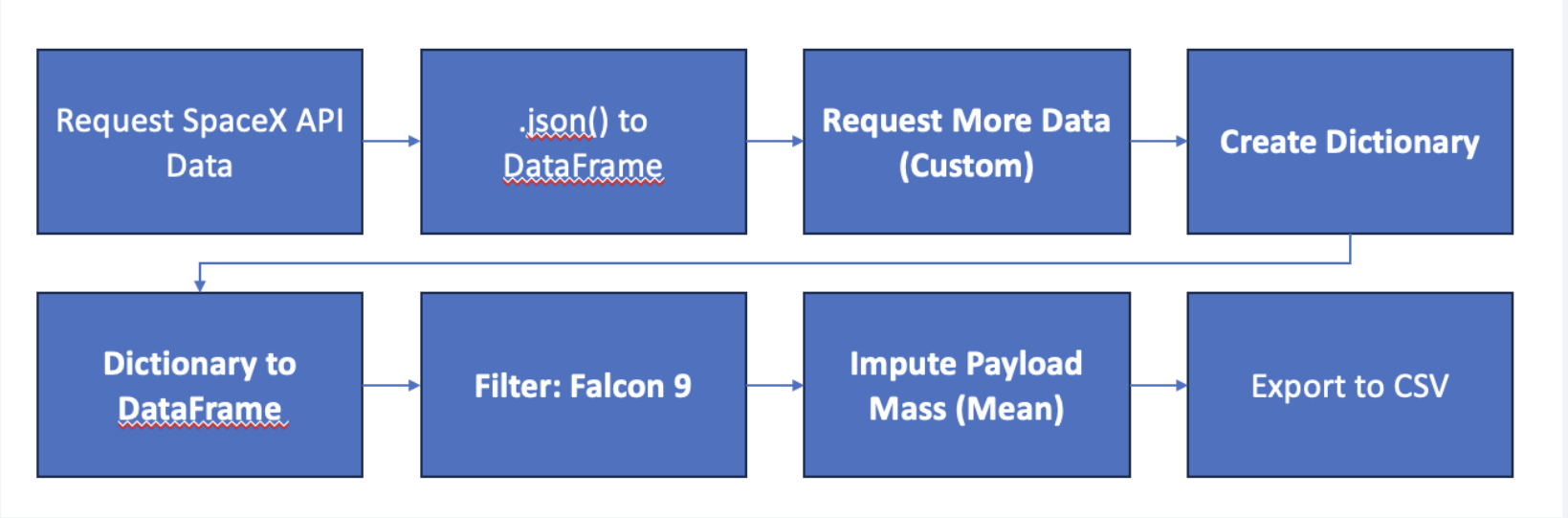
The DataFrame was filtered to include only Falcon 9 launches.

The extracted data was structured into a dictionary and then used to create a DataFrame.

Missing values in the PayloadMass column were replaced with the calculated mean for that column.

The resulting DataFrame was exported to a CSV file.

- For a detailed view of the code and implementation, please refer to the following GitHub notebook: SpaceX API Data Collection Notebook

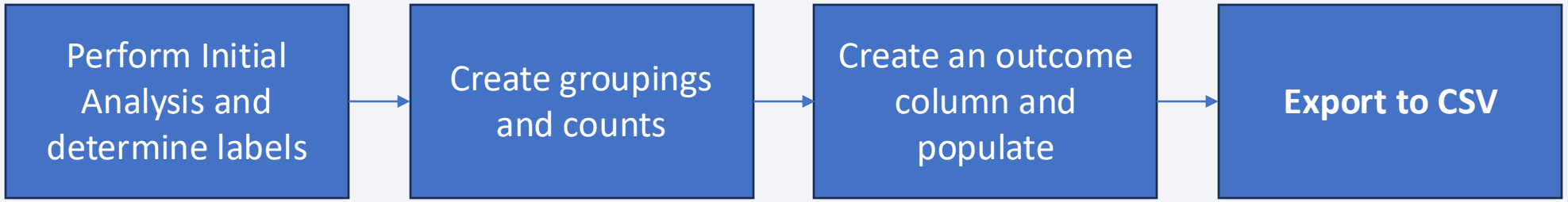


Data Collection - Scraping

- The process involved web scraping Falcon 9 historical launch records from the Wikipedia page "List of Falcon 9 and Falcon Heavy launches." BeautifulSoup was used to extract the relevant HTML table containing the launch records. Finally, BeautifulSoup and Pandas worked together to parse this table and convert it into a structured Pandas DataFrame.
- For a detailed view of the code and implementation, please refer to the following GitHub notebook: Web Scraping Lab

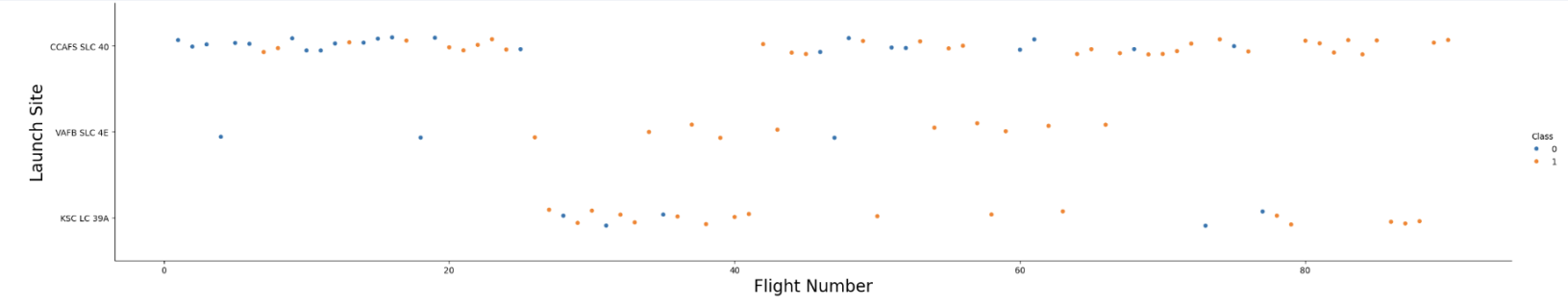
Data Wrangling

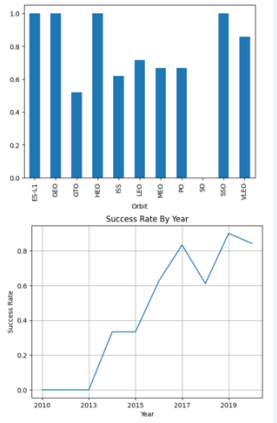
- In the dataset, several mission outcomes indicated unsuccessful booster landings. Specifically, "False Ocean," "False RTLS," and "False ASDS" denoted unsuccessful landing attempts in the ocean, at a ground pad, and on a drone ship, respectively. Conversely, "True Ocean," "True RTLS," and "True ASDS" indicated successful landings in those locations. These outcomes were converted into training labels: "1" for successful booster landings and "0" for unsuccessful ones.
- For a detailed view of the code and implementation, please refer to the following GitHub notebook: Data Wrangling Notebook



EDA with Data Visualization

- A thorough data analysis and feature engineering process was conducted using Pandas and Matplotlib. This involved in-depth exploratory data analysis, careful data preparation, and the creation of clear visualizations to highlight key findings. Key chart types created were Scatter Plots, Line Plots and Bar Charts
- For a detailed view of the code and implementation, please refer to the following GitHub notebook: EDA with DV Notebook





EDA with SQL

- Conducted a comprehensive analysis of the SpaceX dataset. First, the dataset was loaded into a Db2 database. Then, SQL queries were executed to extract key insights and answer assignment questions including:
  - Names of the unique launch sites in the space mission
  - Top 5 launch sites whose name begin with the string 'CCA'
  - Total payload mass carried by boosters launched by NASA (CRS)
  - Average payload mass carried by booster version F9 v1.1
  - Date when the first successful landing outcome in ground pad was achieved
  - Names of the boosters which have success in drone ship and have payload mass between 4000 and 6000 kg
  - Total number of successful and failure mission outcomes
  - Names of the booster versions which have carried the maximum payload mass
  - Failed landing outcomes in drone ship, their booster versions, and launch site names for in year 2015
  - Rank of the count of landing outcomes (such as Failure (drone ship) or Success (ground pad)) between the date 2010-06-04 and 2017-03-20.
- This analysis provided valuable information for understanding SpaceX launch activities and mission outcomes.
- For a detailed view of the code and implementation, please refer to the following GitHub notebook: EDA with SQL Notebook

Build an Interactive Map with Folium

- The launch sites were mapped, and key spatial relationships were visualized. Specifically, NASA Johnson Space Center was marked with a circle, popup label, and text label, using its latitude and longitude. All launch sites were similarly marked to illustrate their geographical locations and proximity to the Equator and coasts. Launch outcomes were also visualized, with colored markers (green for success, red for failure) and marker clusters to highlight launch site success rates. Finally, distances between Launch Site KSC LC-39A and nearby features such as railways, highways, coastlines, and the closest city were displayed using colored lines.
- For a detailed view of the code and implementation, please refer to the following GitHub notebook: Interactive Map Notebook

Build a Dashboard with Plotly Dash

Created an interactive dashboard with analytical capabilities. The following elements were added:

- Launch Sites Dropdown List: A dropdown list was added to allow users to select specific launch sites. This enables focused analysis on individual locations.
- Pie Chart showing Success Launches (All Sites/Certain Site): A pie chart was included to provide a clear view of launch success. When no site is selected, it shows the total successful launch count for all sites. If a specific launch site is selected from the dropdown, the pie chart displays the success versus failure counts for that particular site. This allows for easy comparison of overall success rates and site-specific performance.
- Slider of Payload Mass Range: A slider was added to enable users to filter the data by payload mass range. This allows for exploration of how payload mass affects launch outcomes.
- Scatter Chart of Payload Mass vs. Success Rate for the different Booster Versions: A scatter chart was added to visualize the relationship between payload mass and launch success rate, broken down by booster version. This helps identify potential correlations and performance differences across booster types.

- For a detailed view of the code and implementation, please refer to the following GitHub notebook: Interactive Dashboard Notebook

Predictive Analysis (Classification)

Predictive analysis was conducted using the following approach:

Exploratory Data Analysis (EDA) and Feature Engineering: Initial data exploration was conducted, and training labels were determined.

Feature Engineering: A column for the class variable was created.

Data Preprocessing: The data was standardized.

Data Splitting: The data was split into training and test sets.

Model Selection and Training:

Support Vector Machine (SVM)

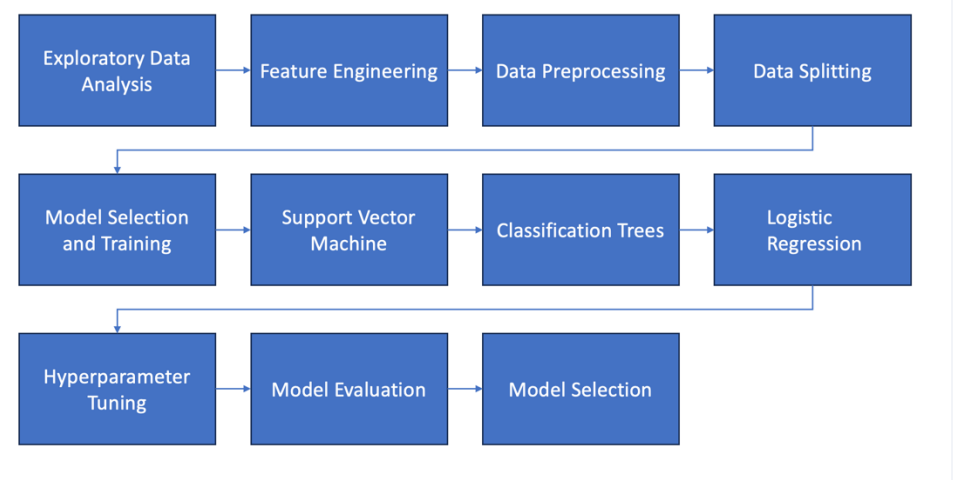
Classification Trees

Logistic Regression

Hyperparameter Tuning: Optimal hyperparameters were found for each model.

Model Evaluation: The performance of each model was evaluated using the test data.

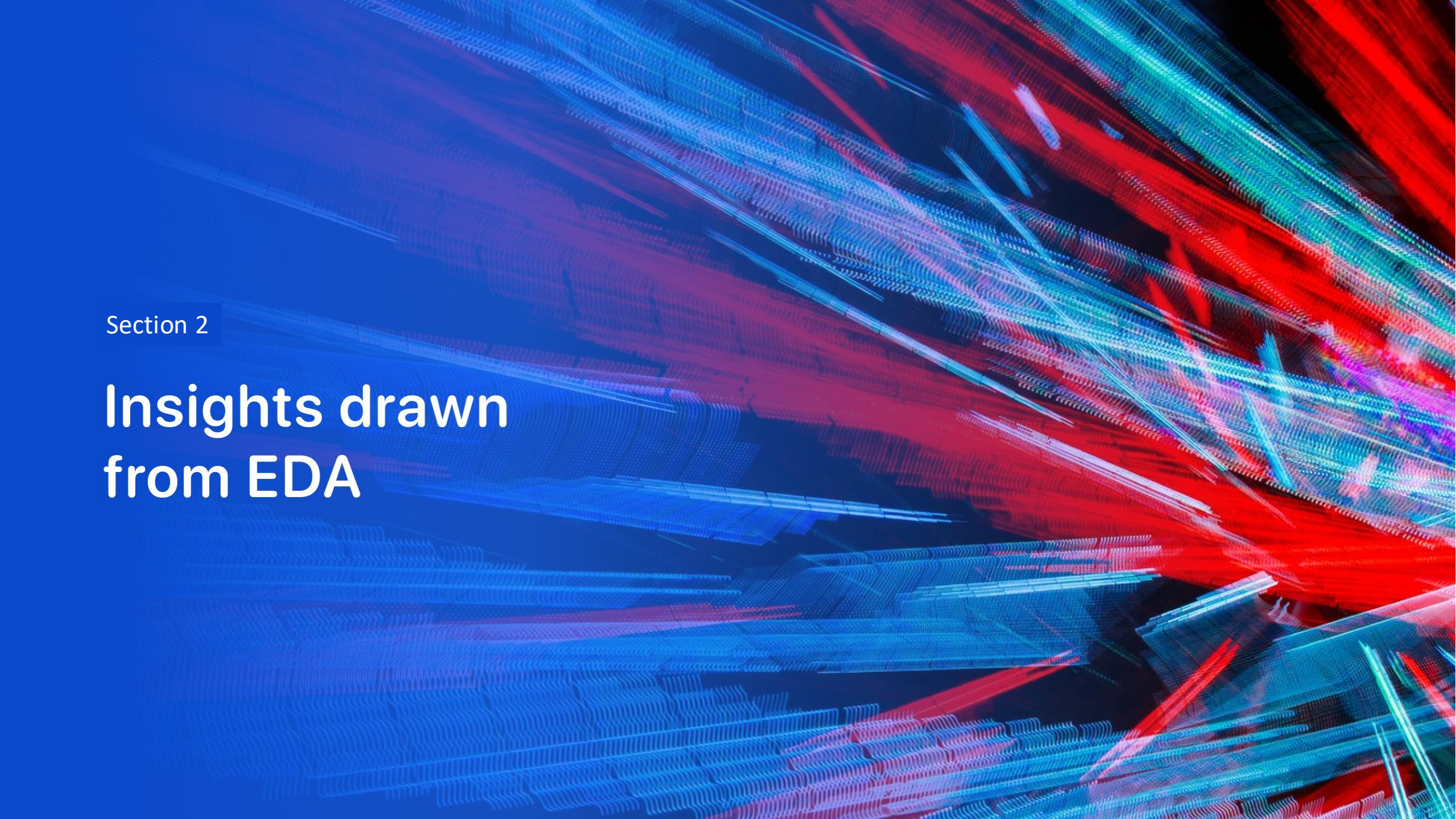
Model Selection: The best-performing model was selected based on the evaluation metrics.



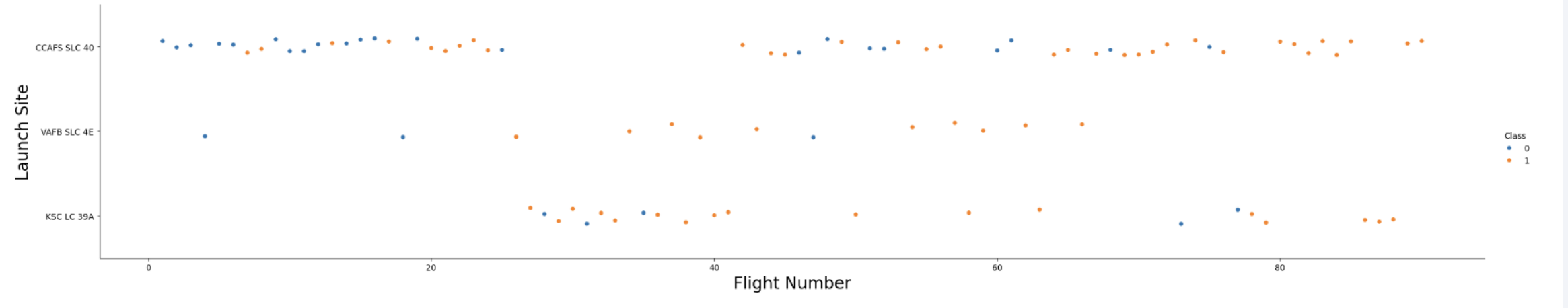
For a detailed view of the code and implementation, please refer to the following GitHub notebook: Predictive Analysis Notebook

Results

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


Section 2
Insights drawn from EDA

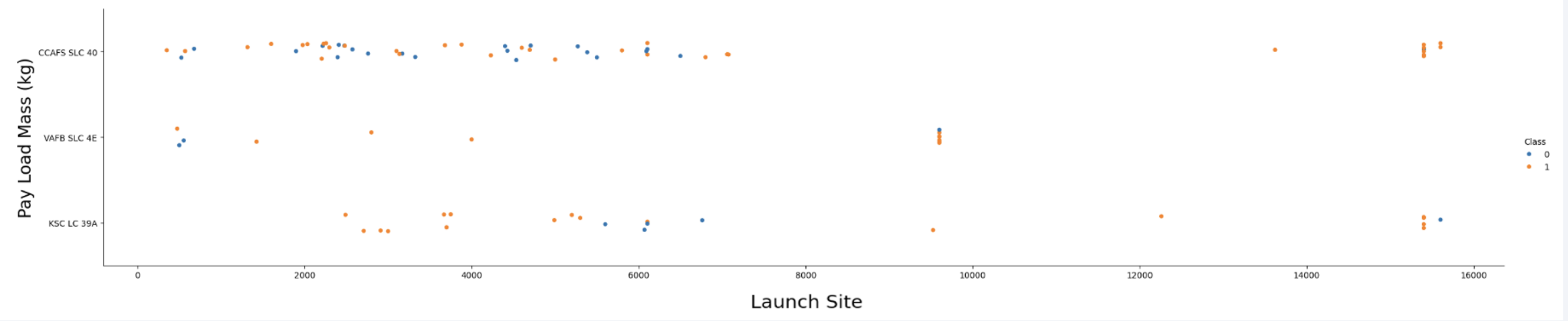
Flight Number vs. Launch Site



Here's a summary of key observations from the launch data:

- Earlier flights exhibited a higher failure rate, while more recent flights demonstrated a higher success rate.
- The CCAFS SLC 40 launch site accounts for approximately half of all launches.
- VAFB SLC 4E and KSC LC 39A launch sites show higher success rates.
- There's a general trend of increasing success rates with newer launches

Payload vs. Launch Site

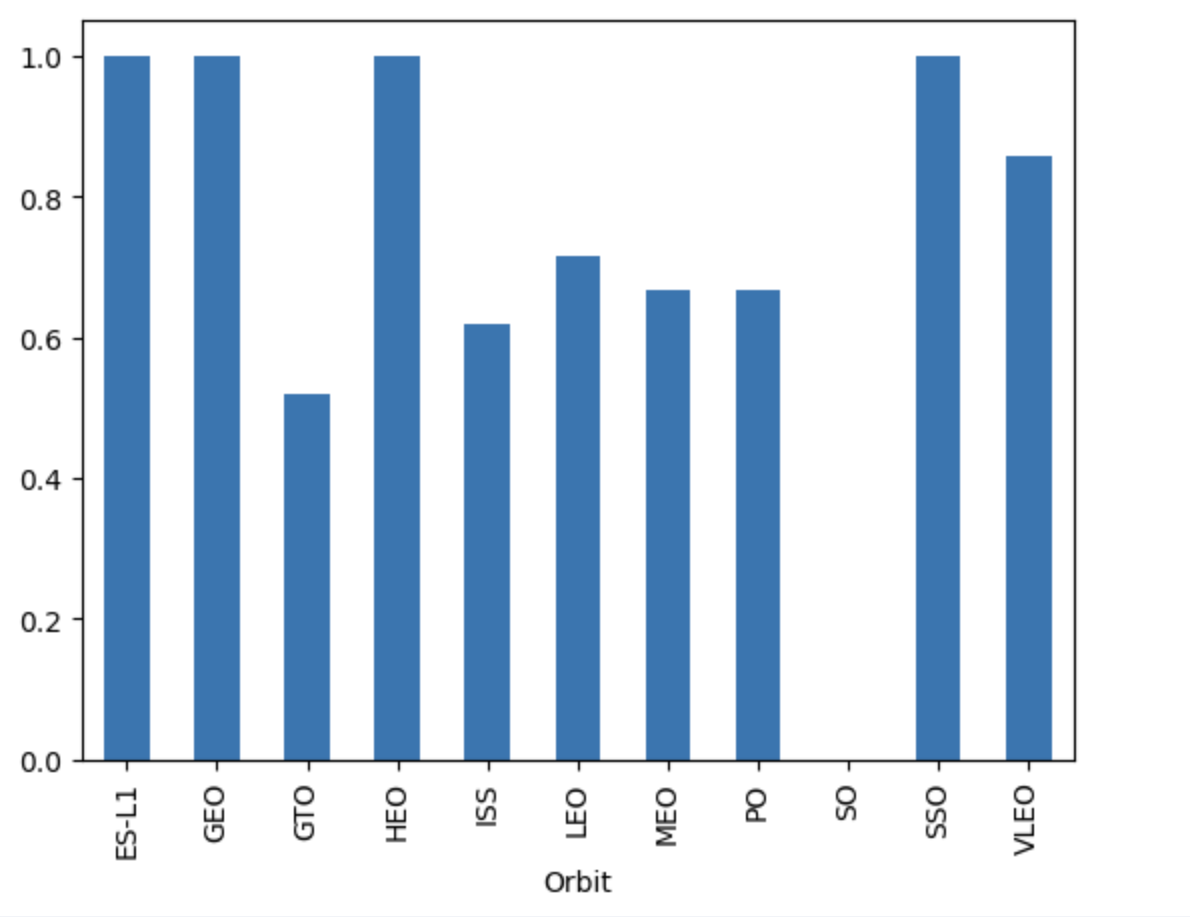


- For every launch site the higher the payload mass, the higher the success rate.
- Most of the launches with payload mass over 7000 kg were successful.
- KSC LC 39A has a 100% success rate for payload mass under 5500 kg too.

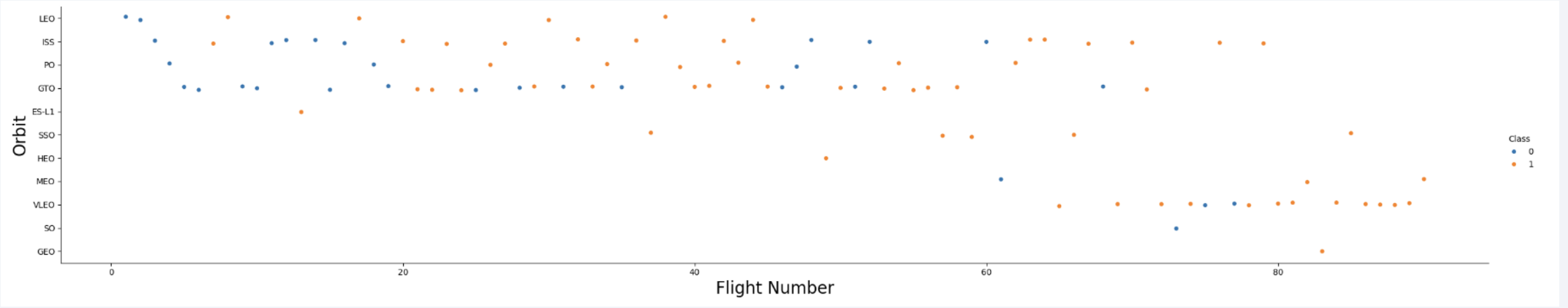
Success Rate vs. Orbit Type

Orbit Summary:

- Orbits with 100% success rate:
  - ES-L1
  - GEO
  - HEO
  - SSO
- Orbits with 0% success rate:
  - SO
- Orbits with success rate between 50% and 85%:
  - GTO
  - ISS,
  - LEO
  - MEO
  - PO
  - VLEO



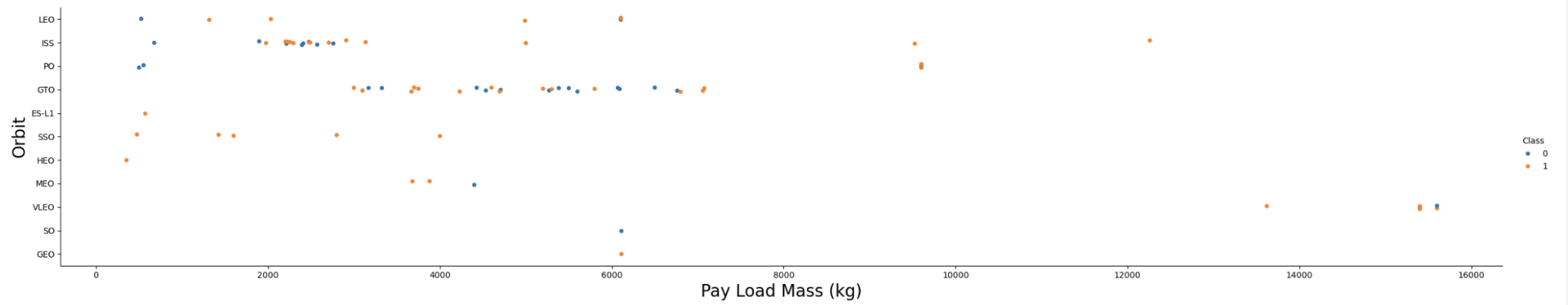
Flight Number vs. Orbit Type



Orbit Trends:

- In LEO orbits, success appears to correlate with the number of flights.
- In GTO orbits, there seems to be no relationship between flight number and success.

Payload vs. Orbit Type



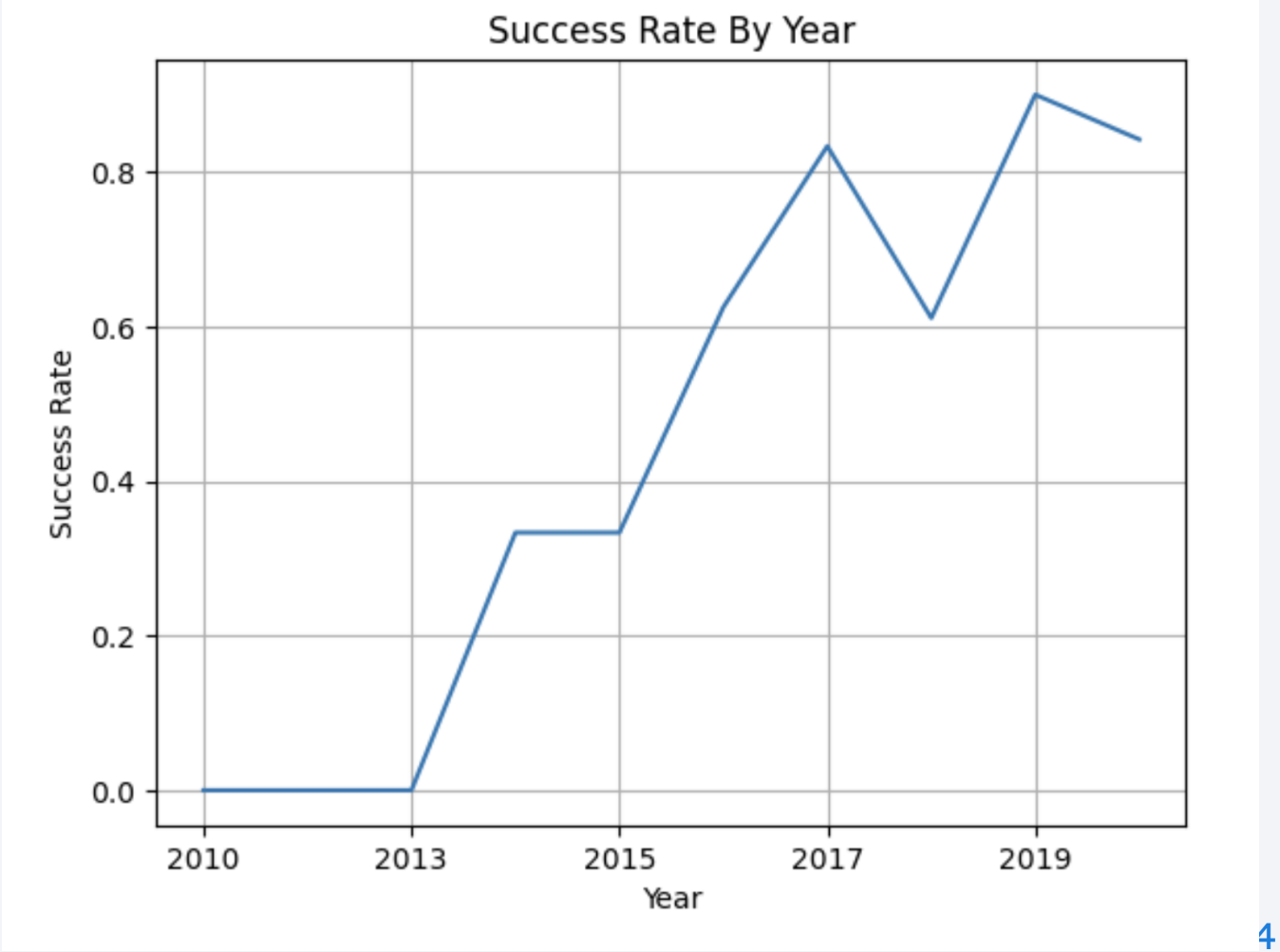
Payload Mass, Orbit Type and Landing Success:

- For launches with heavy payloads, successful landings are more frequent in Polar, LEO, and ISS orbits.
- In GTO orbits, successful and unsuccessful landings are more evenly distributed, making it difficult to discern a clear pattern.

Launch Success Yearly Trend

Trends in Success Rate:

- The success rate since 2013 increased until 2020.



All Launch Site Names

There are 4 distinct launch sites. The DISTINCT clause in the SQL statement returns all unique launch sites:

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

```
In [12]: %sql SELECT DISTINCT "Launch_Site" FROM SPACEXTABLE
 * sqlite:///my_data1.db
Done.
```

Out[12]:

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

Launch Site Names Begin with 'CCA'

Using the LIKE SQL clause we can filter by sites which start with ‘CCA’. The LIMIT 5 clause restricts the return values to 5 rows.

In [19]:
```
%sql SELECT * from SPACEXTABLE WHERE "Launch_Site" LIKE "CCA%" LIMIT 5
```

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

Out[19]:

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

Total Payload Mass

```
In [28]: %sql SELECT SUM("PAYLOAD_MASS__KG_") FROM SPACEXTABLE WHERE "Customer" LIKE "NASA (CRS)"
 * sqlite:///my_data1.db
Done.
Out[28]: SUM("PAYLOAD_MASS__KG_")
                                45596
```

The total payload mass in KG is 45,596. We use the SUM SQL clause to get the total of all values where the Customer is NASA

Average Payload Mass by F9 v1.1

```
In [29]: %sql SELECT AVG("PAYLOAD_MASS__KG_") FROM SPACEXTABLE WHERE "Booster_Version" LIKE "F9 v1.1"
 * sqlite:///my_data1.db
Done.
Out[29]: AVG("PAYLOAD_MASS__KG_")
         2928.4
```

The average payload mass in KG is 2,928.4. We use the AVG SQL clause to get the total of all values where the Booster_Version is ‘F9 v1.1’

First Successful Ground Landing Date

```
In [31]: %sql SELECT MIN("Date") FROM SPACEXTABLE WHERE "Landing_Outcome" LIKE "Success (ground pad)"
 * sqlite:///my_data1.db
Done.
Out[31]: MIN("Date")
         2015-12-22
```

The first successful landing date is 2015-12-22. We use the MIN SQL clause on the Date field where the Landing_Outcome is ‘Success’ to calculate this date.

Successful Drone Ship Landing with Payload between 4000 and 6000

```
In [32]: %sql SELECT DISTINCT "Booster_Version" FROM SPACEXTABLE WHERE "Landing_Outcome" LIKE "Success (drone ship)" AND "
 * sqlite:///my_data1.db
Done.
```

Out[32]:

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

Listing the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000. The WHERE clause is longer and contains AND key words to add additional criteria.

Total Number of Successful and Failure Mission Outcomes

```
In [34]: %sql SELECT COUNT("Mission_Outcome") FROM SPACEXTABLE WHERE "Mission_Outcome" LIKE "%Success%"
 * sqlite:///my_data1.db
Done.
Out[34]: COUNT("Mission_Outcome")
                              100

In [36]: %sql SELECT COUNT("Mission_Outcome") FROM SPACEXTABLE WHERE "Mission_Outcome" LIKE "%Fail%"
 * sqlite:///my_data1.db
Done.
Out[36]: COUNT("Mission_Outcome")
                                1
```

There were 100 successful mission outcomes and 1 unsuccessful mission outcomes. The SQL statements use the COUNT clause and then filter by Mission_Outcome

Boosters Carried Maximum Payload

```
In [40]: %sql SELECT DISTINCT("Booster_Version") FROM SPACEXTABLE WHERE "PAYLOAD_MASS__KG_" = (SELECT MAX("PAYLOAD_MASS__K
 * sqlite:///my_data1.db
Done.
```

Out[40]:

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

The following query is used to determine Boosters carrying the maximum payload:

SELECT DISTINCT("Booster_Version") FROM SPACEXTABLE WHERE "PAYLOAD_MASS__KG_" = (SELECT MAX("PAYLOAD_MASS__KG_") FROM SPACEXTABLE)

2015 Launch Records

```
In [43]: %sql SELECT substr("Date", 6,2) as "Month", "Booster_Version", "Launch_Site" FROM SPACEXTABLE WHERE "Landing_Out
 * sqlite:///my_data1.db
Done.
```

Out[43]:

Month	Booster_Version	Launch_Site
01	F9 v1.1 B1012	CCAFS LC-40
04	F9 v1.1 B1015	CCAFS LC-40

The following query displays the month names, booster versions, launch_site for failures for the months in year 2015:

SELECT substr("Date", 6,2) as "Month", "Booster_Version", "Launch_Site" FROM SPACEXTABLE WHERE "Landing_Outcome" LIKE "Failure (drone ship)" AND substr("Date",0,5)='2015'

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

```
SELECT COUNT("Landing_Outcome") FROM SPACEXTABLE
WHERE "DATE">"2010-06-04" AND "DATE"<"2017-03-20"
ORDER BY
```

Section 3

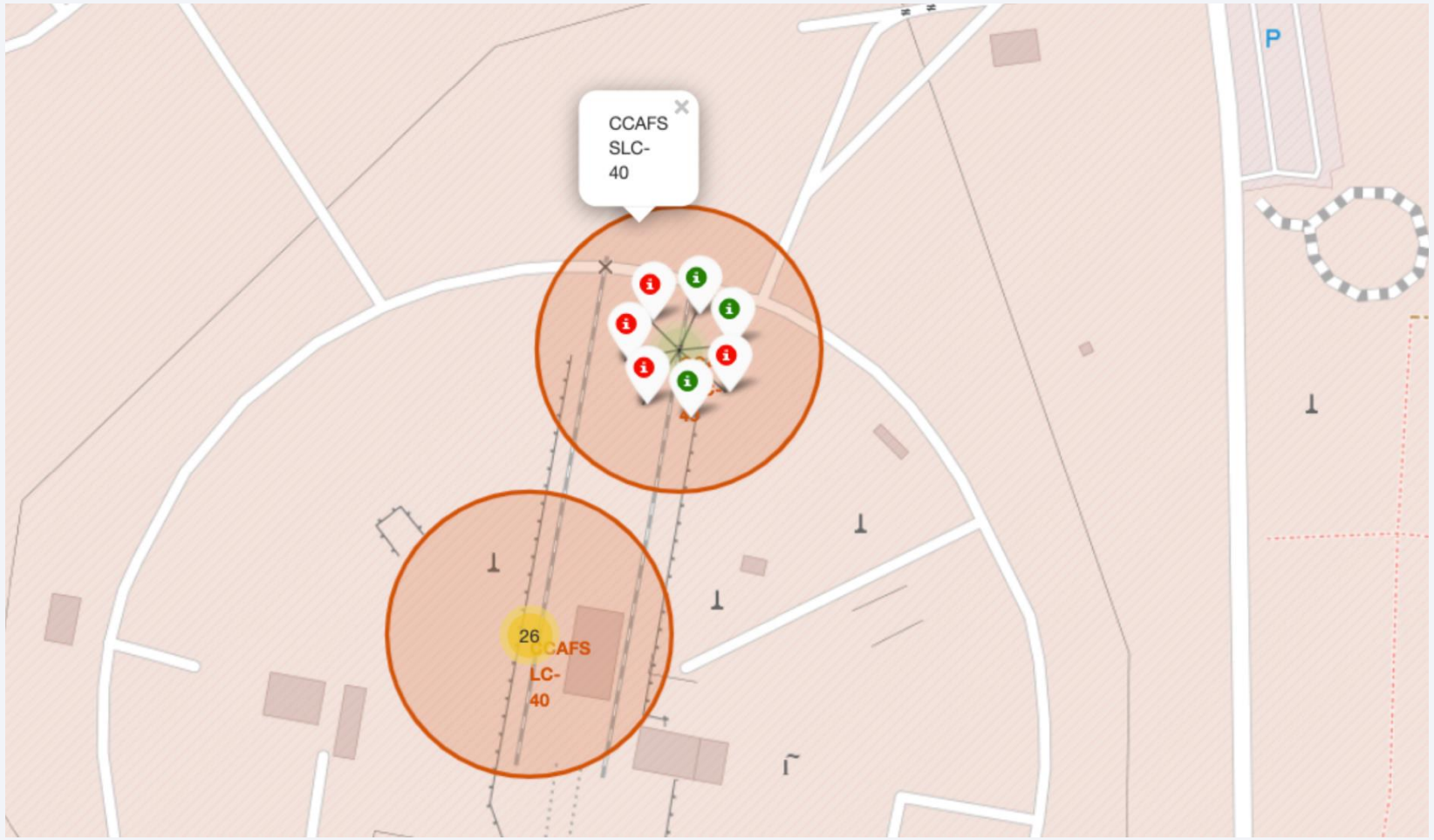
Launch Sites Proximities Analysis

All Launch Sites' Location Markers on a Global Map

Launch Site Selection Rationale:

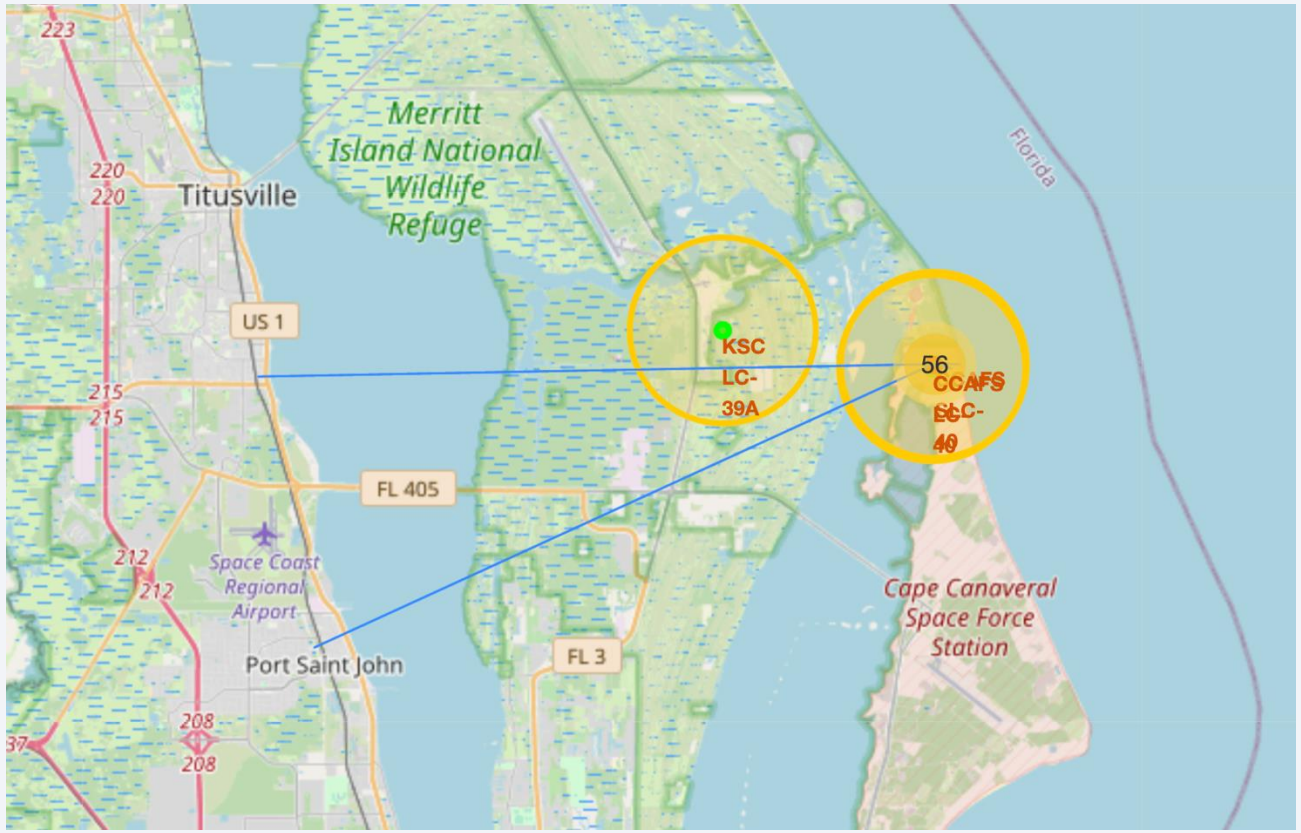
- Launch sites are primarily located near the Equator to leverage the Earth's rotational speed (approximately 1670 km/hour), which provides additional velocity for spacecraft and helps them achieve orbit.
- Sites are also positioned in close proximity to coastlines, enabling launches over the ocean to minimize the risk of debris or explosions affecting populated areas

Colour-labeled launch records on the map



Launch sites are color-coded (green for successful, red for failed launches) to visually indicate their relative success rates. CCAFS SLC-40 has a mixed success rate (3 successes vs 4 failures)

Distance from the launch site CCAFS SLC-40 to its proximities

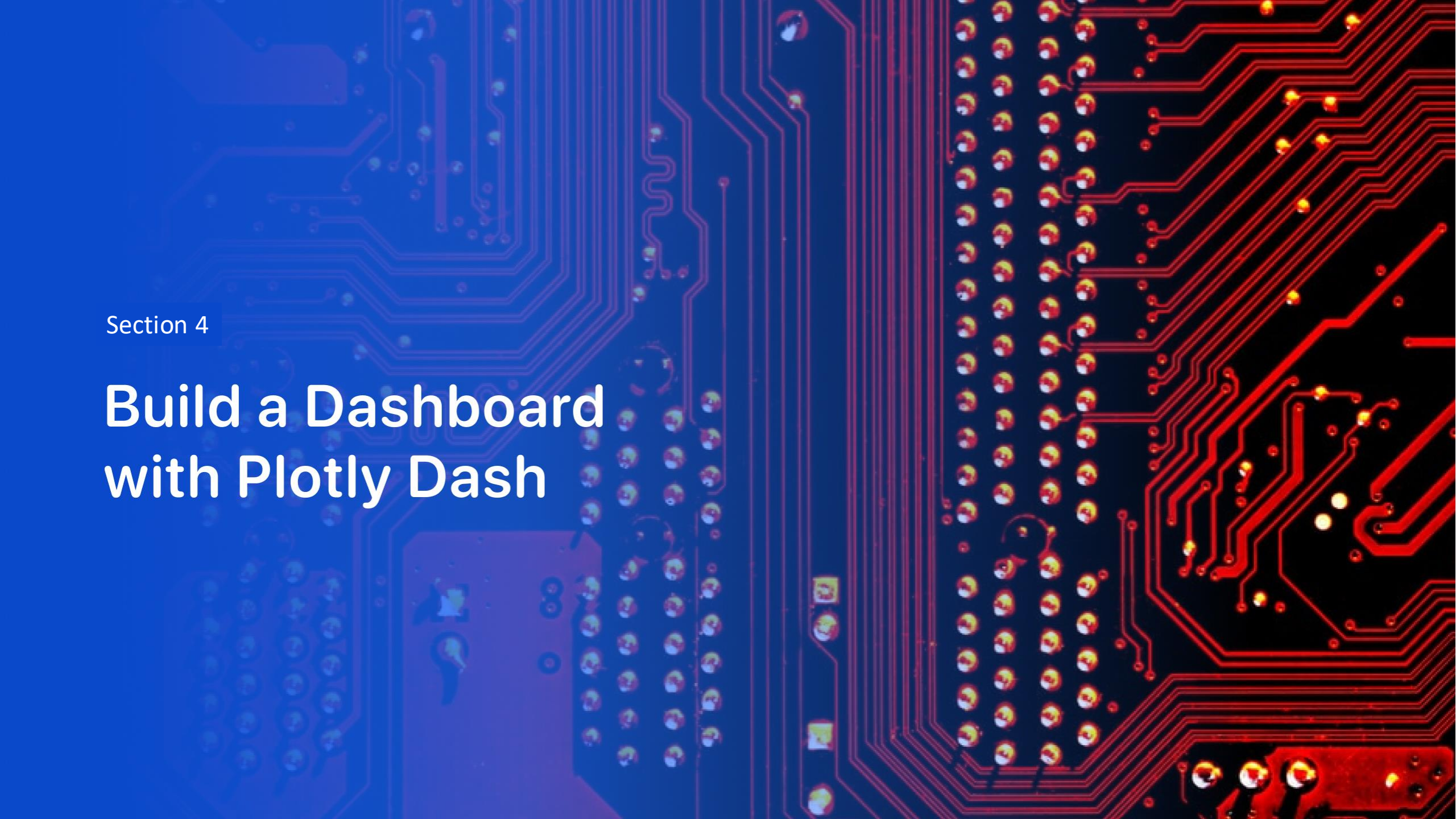


From the visual analysis of the launch site CCAFS SLC-40 we can clearly see that it is:

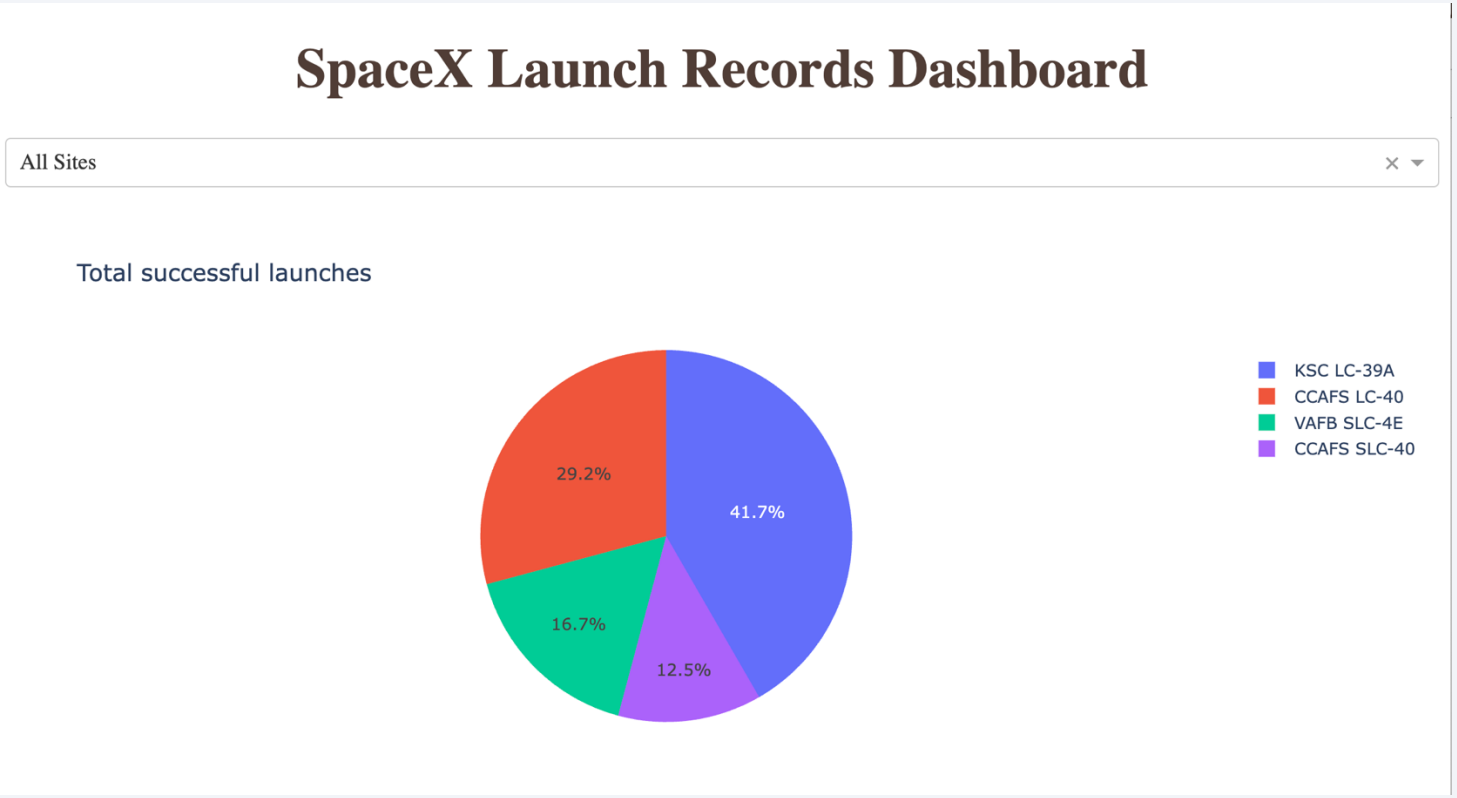
- relatively close to a railway
- relatively close to a highway
- very close to coastline.

The launch site CCAFS SLC-40 is relatively close to its closest city Port Saint John.

A failed rocket with its high speed can cover large distances in a few seconds. It can pose a danger to nearby populated areas.

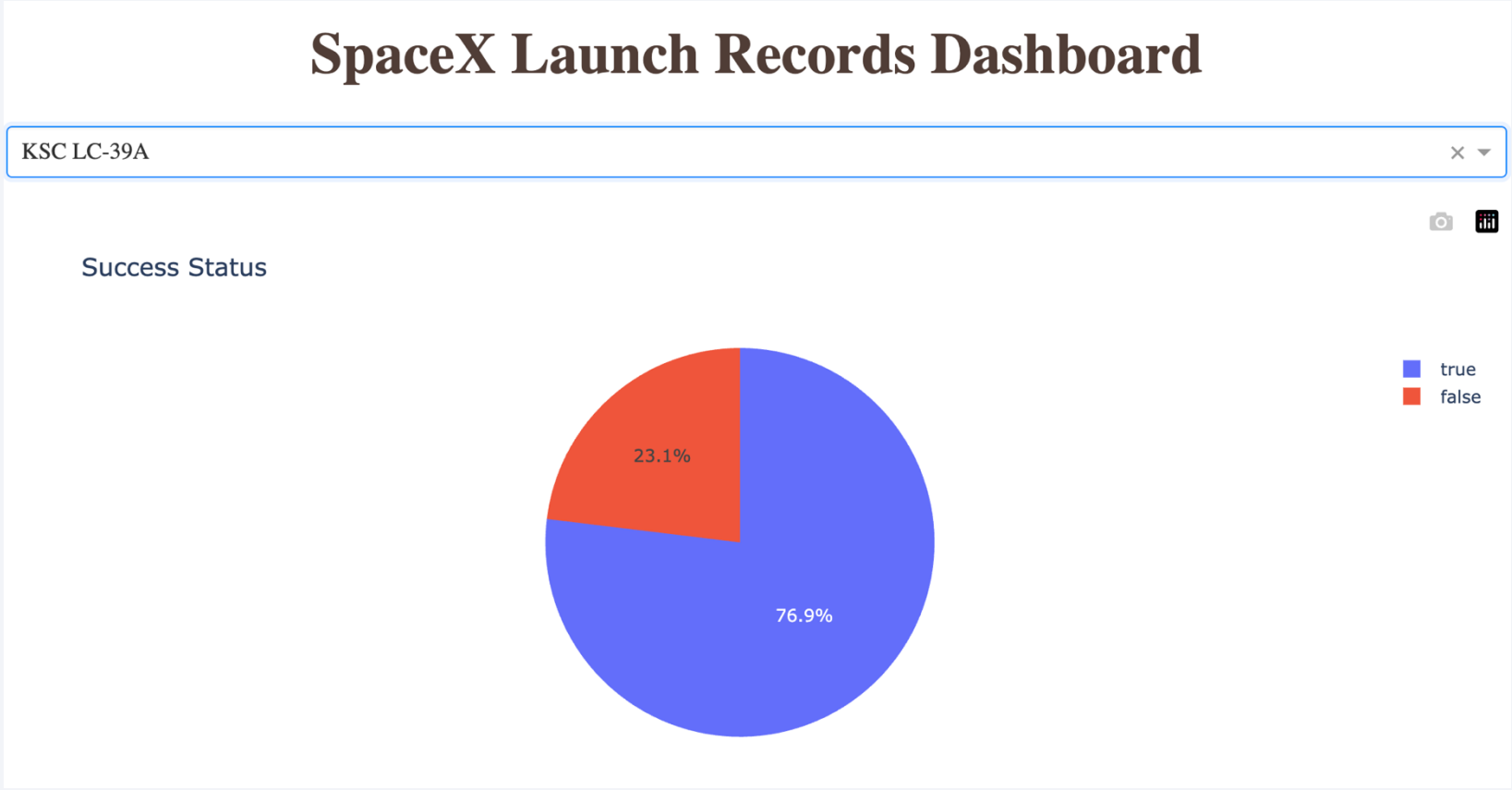

Section 4
Build a Dashboard with Plotly Dash

Launch success count for all sites



The chart indicates that KSC LC-39A has the most successful launches. CCAFS LC-40 has a lot of successful launches also.

Launch site with highest launch success ratio



KSC LC-39A has the highest launch success rate (73.1%) with 10 successful and 3 failed landings.

Payload Mass vs. Launch Outcome for all sites

The highest number of Successes occurred in the range 1800 Kg -5500 Kg,

Failures were slightly more prevalent in the lower payload range, but they did occur across the entire range

Section 5

Predictive Analysis (Classification)

Classification Accuracy

```
In [16]: logreg_cv.score(X_test, Y_test)
Out[16]: 0.8333333333333334
```

```
In [22]: svm_cv.score(X_test, Y_test)
Out[22]: 0.8333333333333334
```

```
In [29]: tree_cv.score(X_test, Y_test)
Out[29]: 0.8888888888888888
```

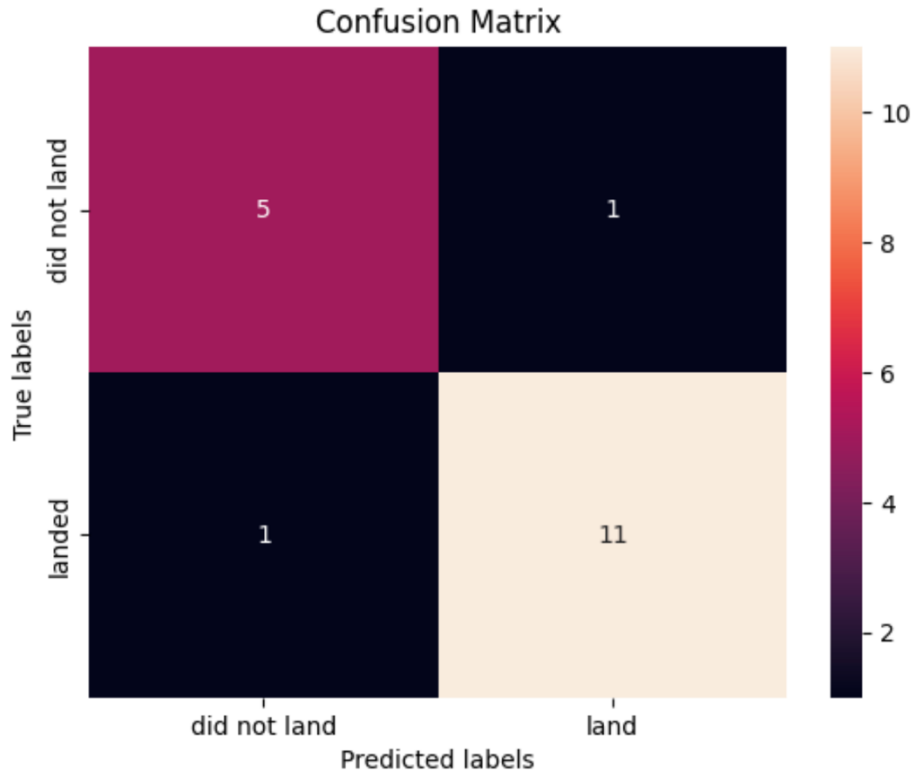
```
In [34]: knn_cv.score(X_test, Y_test)
Out[34]: 0.8333333333333334
```

The test set results did not definitively identify the best performing method, potentially due to the small sample size (18 samples).

Evaluation across the entire dataset revealed the Decision Tree Model as superior, exhibiting both higher scores and the highest accuracy.

Confusion Matrix

```
In [30]: yhat = tree_cv.predict(X_test)
         plot_confusion_matrix(Y_test,yhat)
```



The best performing model was the Decision Tree model.

Examining the confusion matrix for this model, we see that it can distinguish between the different classes.

It very accurately predicted the outcomes with only 2 outliers in the entire data set.

Conclusions

In summary, the Decision Tree Model is the most effective algorithm for this dataset.

- Launches with lower payload mass demonstrate better results than those with larger payloads.
- Most launch sites are located near the Equator and in close proximity to the coast.
- Launch success rates have shown a general increase over the years, with KSC LC-39A exhibiting the highest success rate among all sites.
- Orbits ES-L1, GEO, HEO, and SSO have a 100% success rate.

Further research is recommended to explore the factors contributing to these trends and to further optimize launch strategies accordingly.

Appendix

Please see linked github artifacts

- SpaceX API Data Collection Notebook
- Web Scraping Lab
- Data Wrangling Notebook
- EDA with DV Notebook
- EDA with SQL Notebook
- Interactive Map Notebook
- Interactive Dashboard Notebook
- Predictive Analysis Notebook

Thank you!