IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Waltter Valdez
23/06/2024

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 map with Folium
  - Dashboard with Plotly Dash
  - Predictive analysis
- Summary of all results
  - Exploratory Data Analysis results
  - Interactive maps and dashboard
  - Predictive analysis results

Introduction

Project background and context

- The aim of this project is to predict if the Falcon 9 first stage will successfully land. SpaceX, the most successful company of the commercial space age, has made space travel more affordable by reusing rocket stages. On its website, SpaceX advertises the cost of a Falcon 9 rocket launch at 62 million dollars, compared to other providers who charge upwards of 165 million dollars each. This significant price difference is attributed to SpaceX's ability to reuse the first stage of their rockets.
- By accurately predicting whether the Falcon 9 first stage will land successfully, we can estimate the cost of a launch. This information is crucial for companies that aim to compete with SpaceX in the rocket launch market. Utilizing public information and advanced machine learning models, this project will predict the likelihood of SpaceX reusing the first stage, thereby determining the potential cost savings.

Introduction

Questions to be answered

- What are the main variables that affect the success of the first stage landing?
- Does the rate of successful landings is improving over the years?
- What is the best prediction algorithm for this task?

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Data was collected in two ways, using SapceX REST API and web scraping from Wikipedia.
- Perform data wrangling
  - Filtering the data and dropping unnecessary columns
  - Dealing with missing values
  - One Hot Encoding to prepare the data for the models
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Building, tuning and evaluation of models to choose the best one

Data Collection

The process involves combining the data from the SapceX REST API and Wikipedia Web Scrapping.

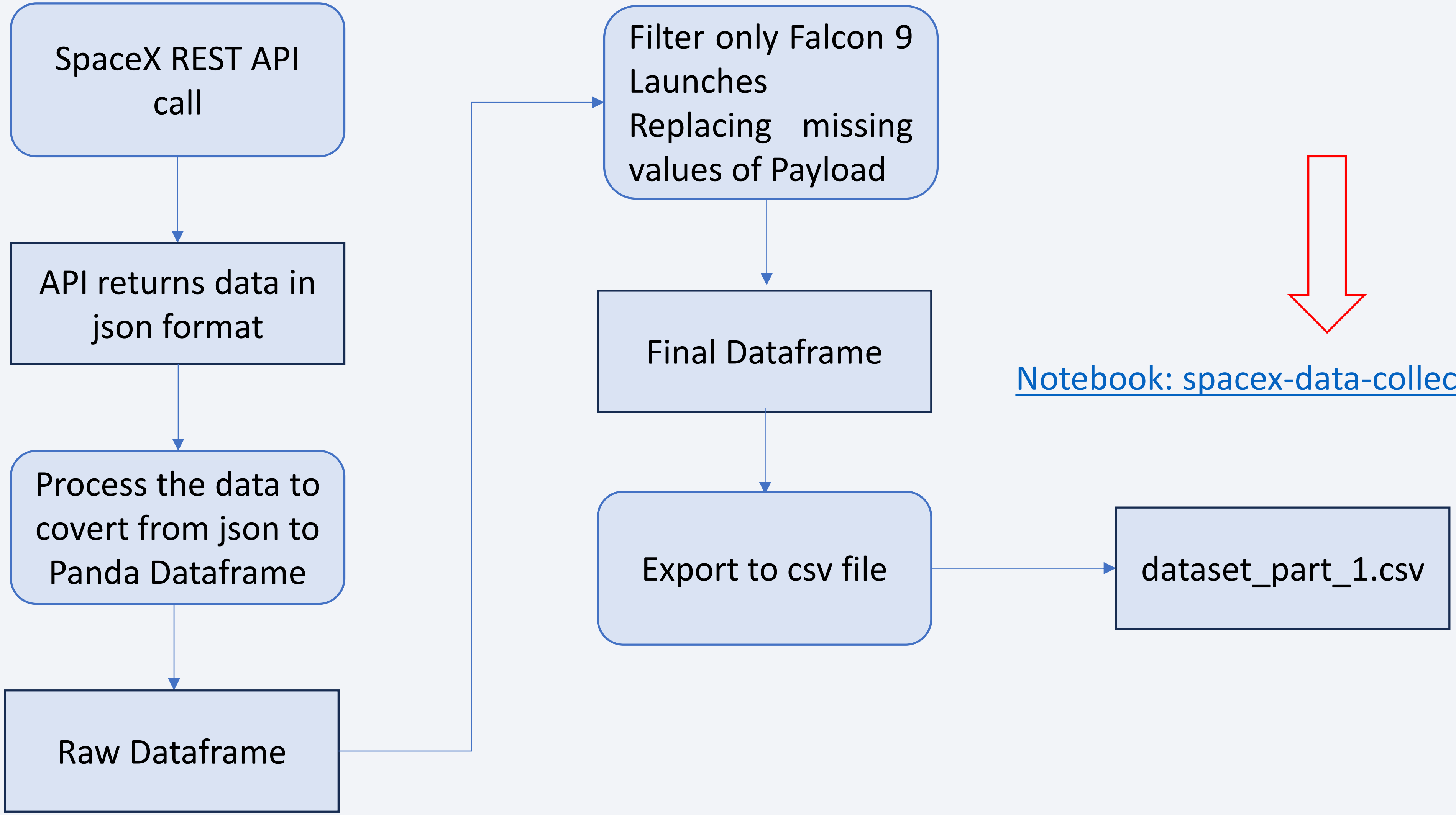
The information from SapceX REST API:

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

The information from Wikipedia Web Scrapping:

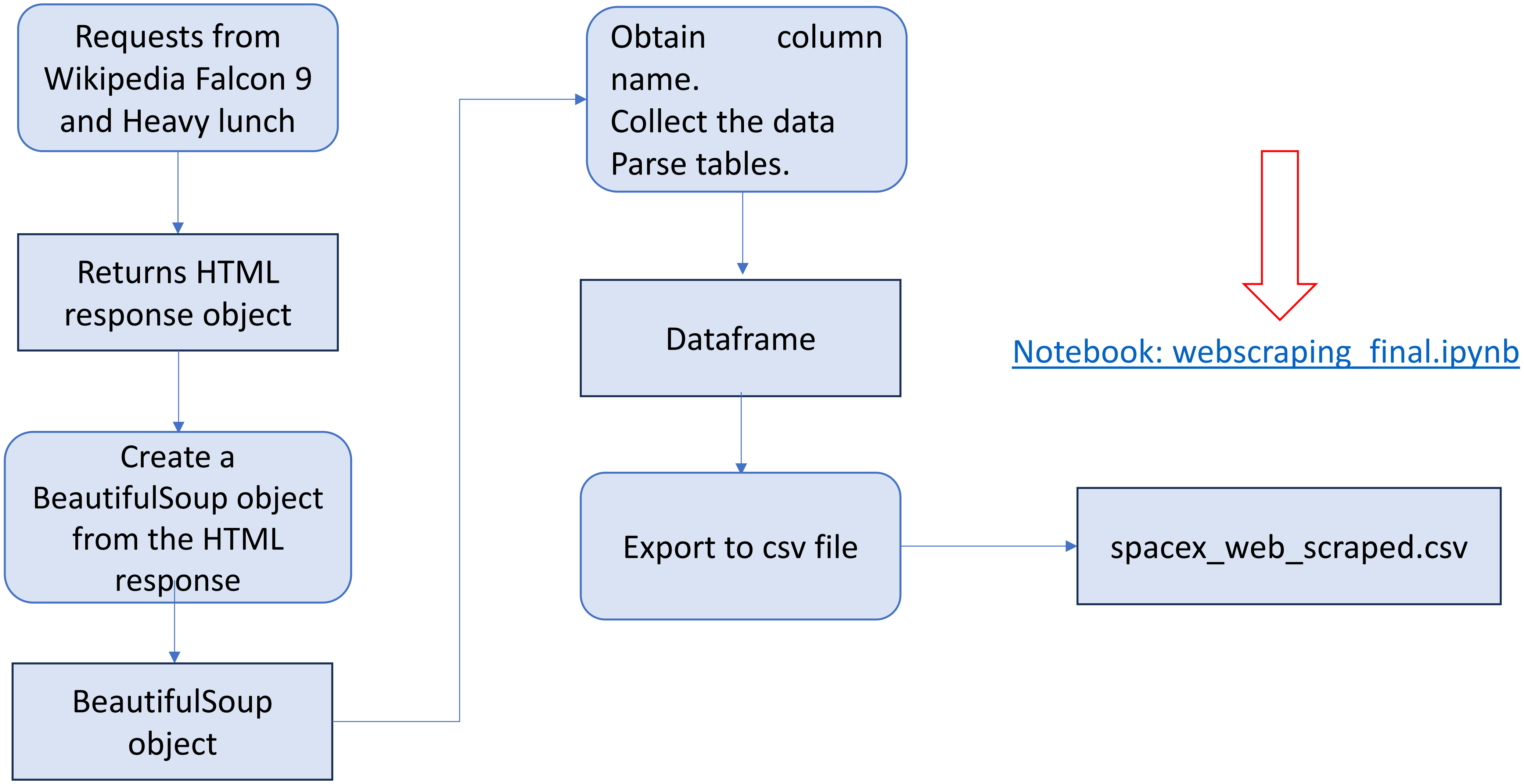
Flight No., Launch site, Payload, PayloadMass, Orbit, Customer, Launch outcome, Version Booster, Booster landing, Date, Time

Data Collection – SpaceX API

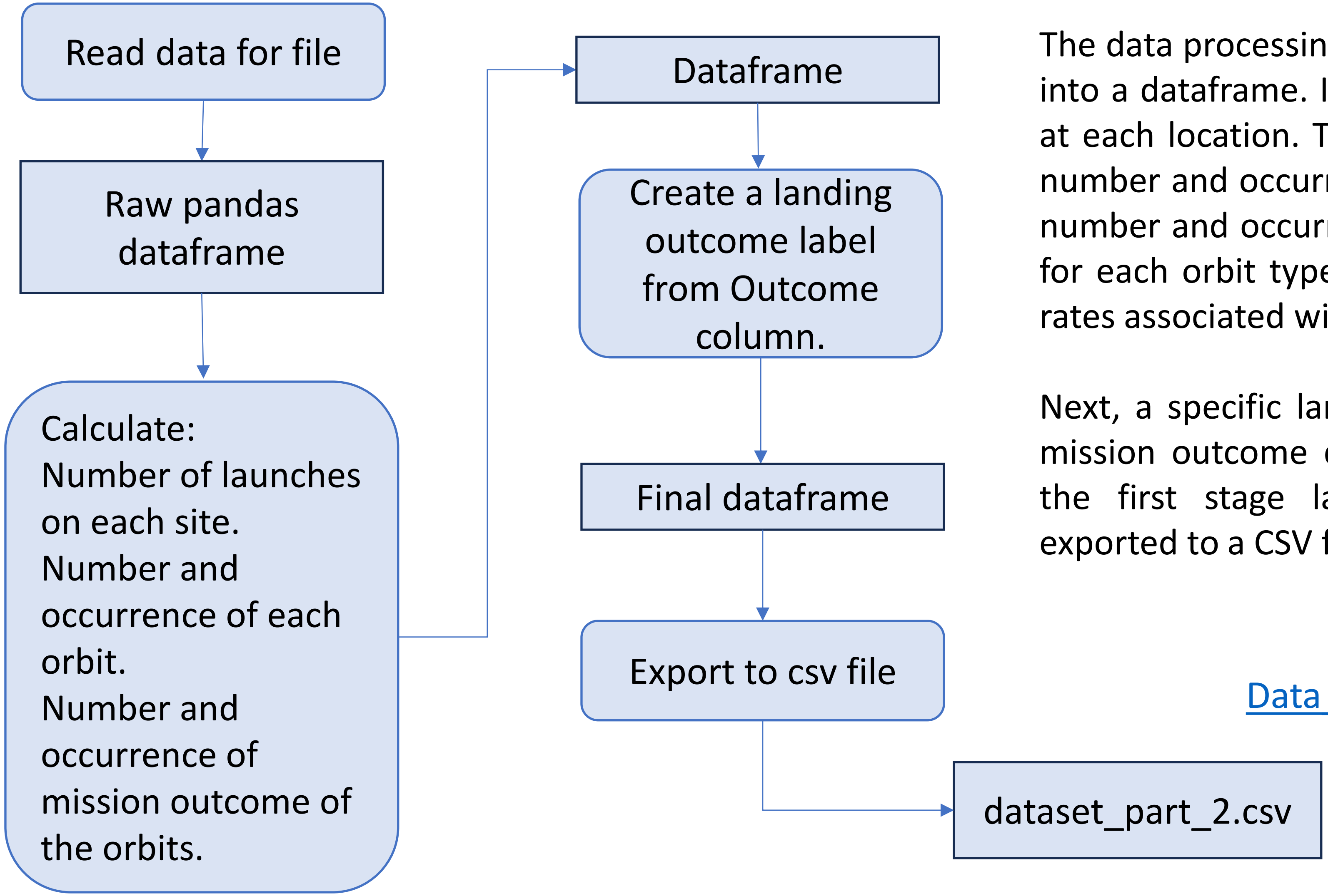


[Notebook: spacex-data-collection-api_final.ipynb]

Data Collection – Web Scraping



Data Wrangling



The data processing workflow begins by loading the data file into a dataframe. It then calculates the number of launches at each location. The dataset is then analyzed to count the number and occurrence of each orbit type. Additionally, the number and occurrence of mission outcomes are calculated for each orbit type, which helps to understand the success rates associated with different orbits.

Next, a specific landing outcome label is created from the mission outcome data, indicating the success or failure of the first stage landing. Finally, the processed data is exported to a CSV file.

[Notebook: Data_wrangling_final.ipynb]

EDA with Data Visualization

Scatter Plots

They are used to illustrate the relationship between variables. This relationship is called **correlation**. Graphs: Flight Number vs. Payload Mass, Flight Number vs. Launch Site, Payload Mass vs. Launch Site, Flight Number vs. Orbit Type and Payload Mass vs Orbit Type

Bar Charts

They are utilized to present comparisons between discrete categories on the basis of a measured value. Graph: Orbit Type vs. Success Rate.

Line charts

The are employed to illustrate trends in data over time (time series). Graph: Success Rate Yearly Trend.

Notebook: eda_dataviz_final.ipynb

EDA with SQL

Performed SQL queries:

- Display the names of the unique launch sites in the space mission.
- Display 5 records where launch sites begin with the string ‘CCA’.
- Display the total payload mass carried by boosters launched by NASA (CRS).
- Display average payload mass carried by booster version F9 v1.1.
- List the date when the first successful landing outcome in ground pad was achieved.
- List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000.
- List the total number of successful and failure mission outcomes.
- List the names of the booster versions which have carried the maximum payload mass.
- List the failed landing outcomes in drone ship, their booster versions, and launch site names for the months in the year 2015.
- Rank the count of landing outcomes (such as Failure (drone ship) or Success (ground pad)) between the dates 2010-06-04 and 2017-03-20 in descending order.

[Notebook: eda-sql-coursera_sqllite_final.ipynb]

Build an Interactive Map with Folium (1/2)

Map Objects Created and Added to a Folium Map

- **NASA Johnson Space Center**:
  - Red Circle: Added a red circle at NASA Johnson Space Center's coordinates.
  - Label: Included a label showing the name of the NASA Johnson Space Center.
- **Launch Sites**:
  - Red Circles: Added red circles at each launch site's coordinates.
  - Labels: Included labels showing the names of the launch sites.
  - Grouped Points: Used MarkerCluster to group points and display multiple and different information for the same coordinates.
- **Landing Outcomes**:
  - Markers: Added markers to show successful (green) and unsuccessful (red) landings.
  - Icons: Used folium Icons to differentiate between successful and unsuccessful landings.
- **Distance Markers and Lines**:
  - Markers: Added markers to show distances between launch sites and key locations such as railways, highways, coastlines, and cities.
  - Lines: Plotted lines between launch sites and these key locations to visualize distances.

Build an Interactive Map with Folium (1/2)

Reasons for Adding These Objects

- Understanding Geographical Locations: By adding circles and markers at the NASA Johnson Space Center and various launch sites, the map provides a clear visual representation of their geographical locations and proximity to significant landmarks.

- Identifying Success Rates: The colored markers (green for success, red for failure) help in quickly identifying which launch sites have higher success rates for landings, thus making it easier to analyze the performance of different sites.

- Visualizing Proximity and Accessibility: The distance markers and lines plotted between launch sites and nearby key locations (railways, highways, coastlines, cities) help in understanding the accessibility and logistical considerations related to each launch site.

[Notebook: launch_site_location_final.ipynb](#)

Build a Dashboard with Plotly Dash

Plots/Graphs and Interactions Added to the Dashboard

Dropdown Component: Allows selection of specific or all launch sites. Added with `dash_core_components.Dropdown`.

Pie Chart: Displays total successful and failed launches for the selected site. Created with `plotly.express.pie`.

Range Slider: Enables selection of payload mass range. Added with `dash_core_components.RangeSlider`.

Scatter Plot: Shows the relationship between payload mass and launch success. Created with `plotly.express.scatter`.

Reasons for Adding These Plots and Interactions

Dropdown Component: Facilitates site-specific or overall data analysis.

Pie Chart: Visualizes launch success and failure rates.

Range Slider: Filters data by payload mass for detailed analysis.

Scatter Plot: Highlights correlation between payload mass and launch success.

Script: spacex_dash_app.py

Predictive Analysis (Classification) (1/3)

Data Preparation:

- Load Dataset: Loaded the dataset into the environment.
- Normalize Data: Standardized the data using `StandardScaler`, fitting and transforming it.
- Split Data: Divided the data into training and testing sets using the `train_test_split` function.

Model Preparation:

- Select Algorithms: Chose machine learning algorithms: Logistic Regression, SVM, Decision Tree, and KNN.
- Set Parameters: Configured parameters for each algorithm using `GridSearchCV` with `cv=10`.
- Train Models: Trained the models using the training dataset with the `GridSearchCV` object.

Model Evaluation:

- Get Best Hyperparameters: Identified the best hyperparameters for each model.
- Compute Accuracy: Calculated the accuracy of each model on the test dataset using the `.score()` method.
- Plot Confusion Matrix: Examined the confusion matrix for all models to evaluate their performance.

Predictive Analysis (Classification) (2/3)

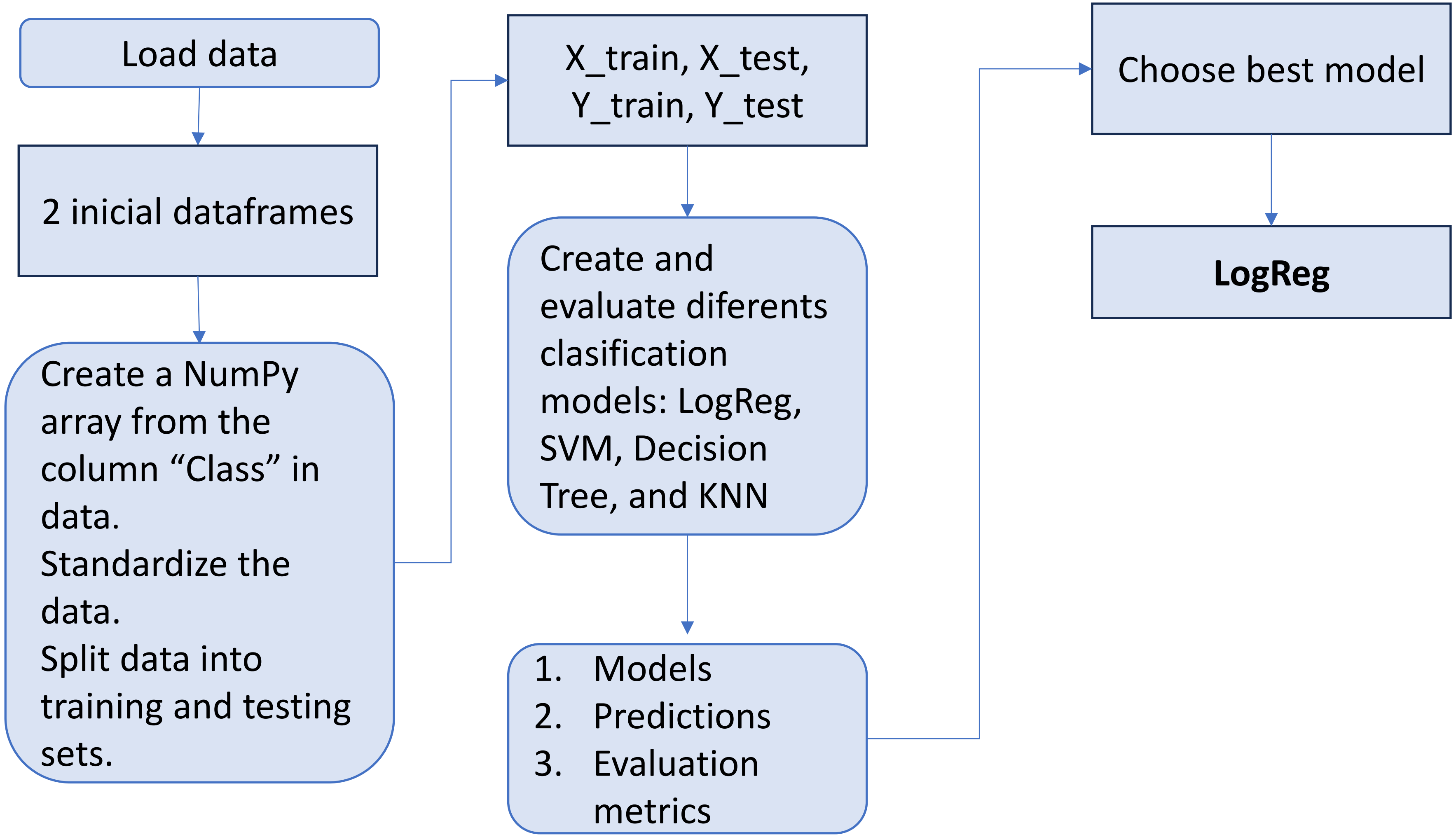
Model Comparison:

- Compare Models: Compared the models based on their accuracy scores.
- Select Best Model: Determined the best performing model by examining the Jaccard score and F1 score metrics, choosing the one with the highest accuracy.

This process ensures the identification of the most effective classification model through systematic preparation, training, evaluation, and comparison.

Notebook: SpaceX_Machine_Learning_Prediction_Part_5_final.ipynb

Predictive Analysis (Classification) (3/3)



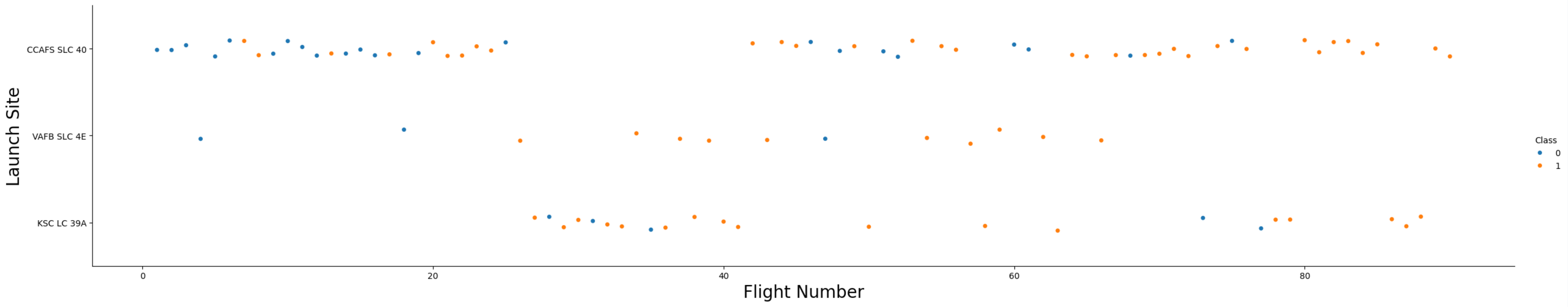
Results

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

Section 2

Insights drawn from EDA

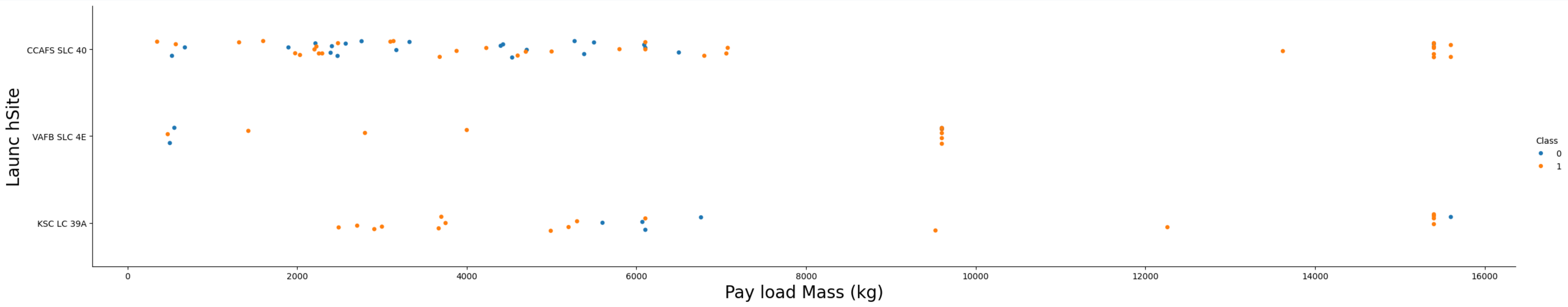
Flight Number vs. Launch Site



- **Early Flights**: The earliest flights, represented by the points on the left side of the plot, predominantly failed, as indicated by the blue markers.
- **Recent Flights**: The most recent flights, on the right side of the plot, mostly succeeded, as shown by the orange markers. This suggests an improvement in success rates over time.
- **Success Rate Trends**: It can be inferred that each new launch has a higher probability of success, indicating continuous improvement in SpaceX's launch processes.
- **CCAFS SLC 40**: This site has the most launches. The success rate improves over time, as indicated by the increasing number of orange markers.
- **VAFB SLC 4E**: This site has fewer launches but shows a relatively high success rate with many orange markers.
- **KSC LC 39A**: This site also shows high success rates, particularly for payloads under 5500 kg, where it has a 100% success rate.

In summary, KSC LC 39A and VAFB SLC 4E show higher success rates compared to CCAFS SLC 40. This suggests that these sites may have better conditions or processes leading to more successful launches.

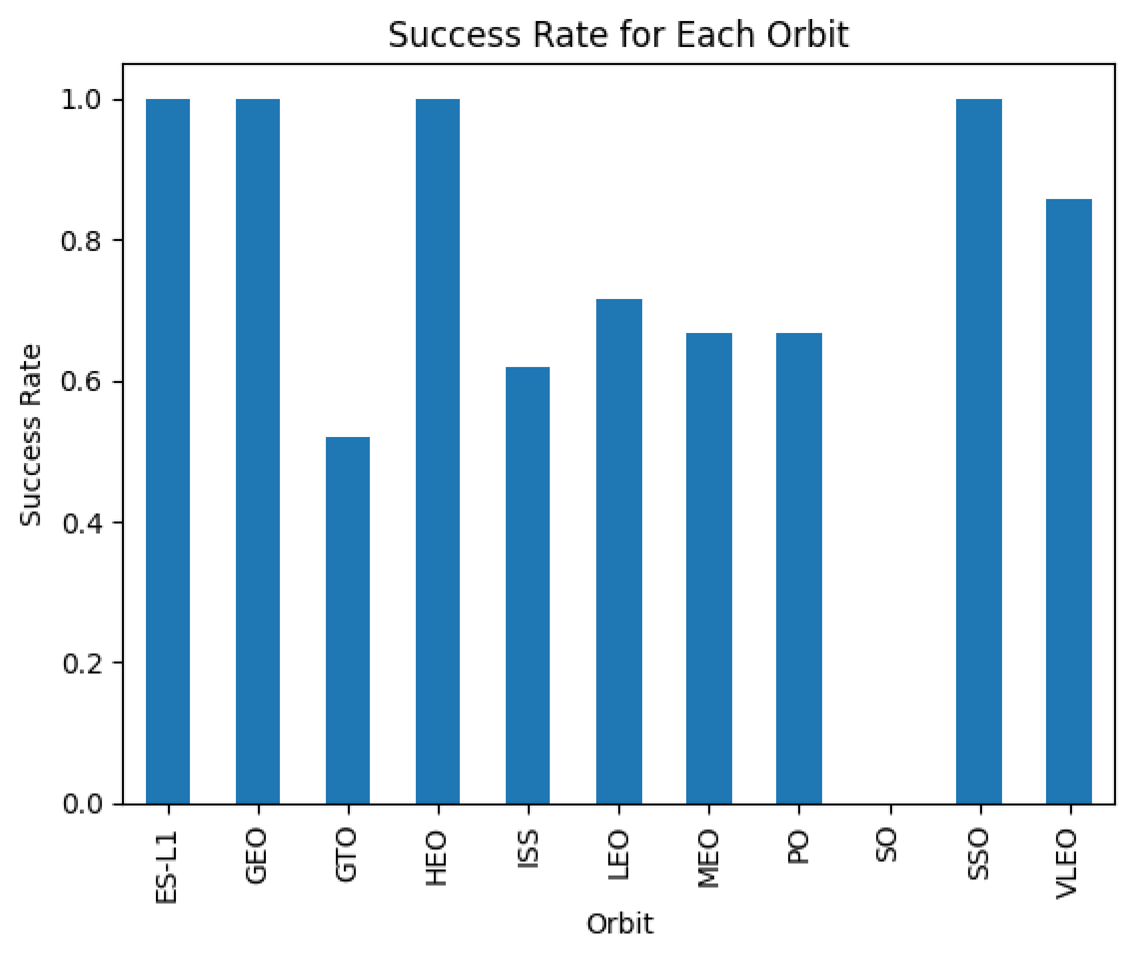
Payload vs. Launch Site



- **Higher Payload Mass and Success Rate:** For every launch site, higher payload mass tends to correlate with higher success rates. Most launches with payload mass over 7000 kg were successful.

- **KSC LC 39A has a 100% success rate for payload mass under 5500 kg too.**

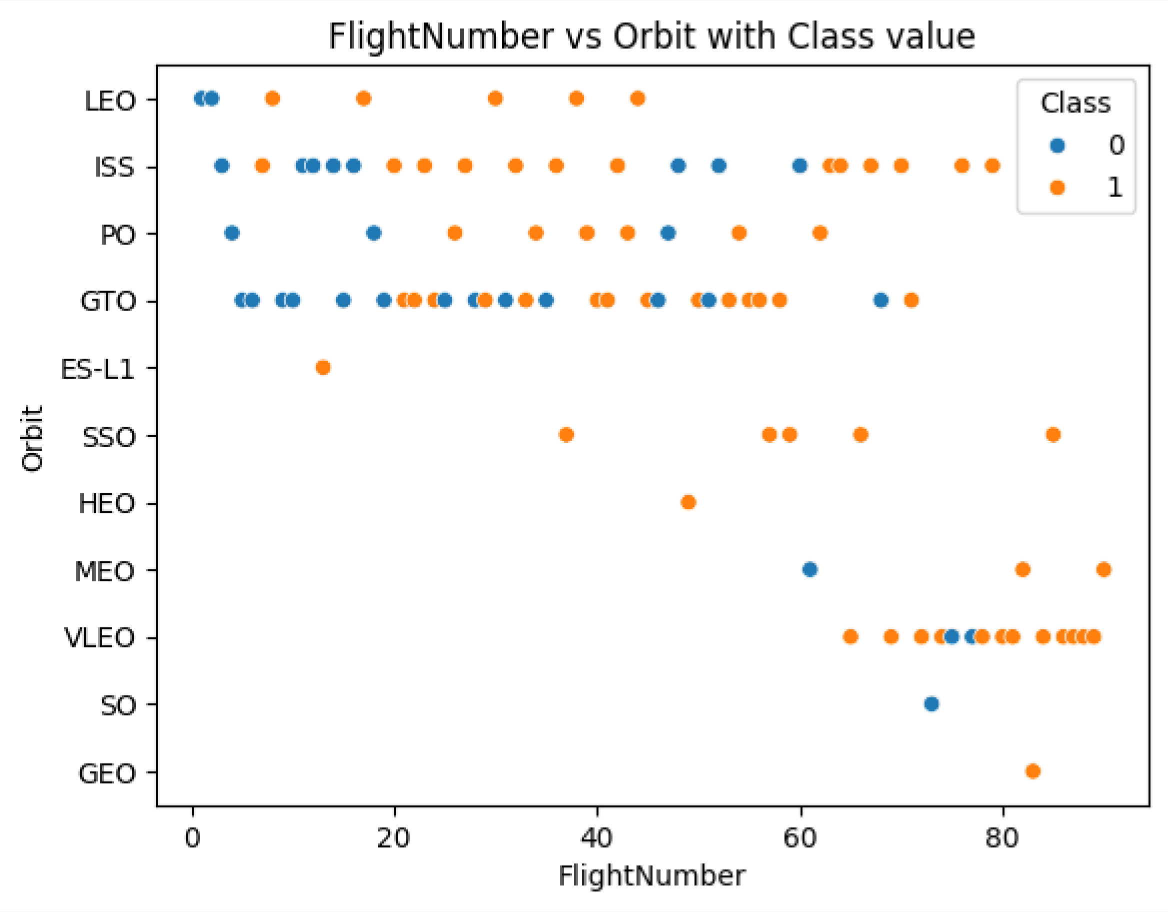
Success Rate vs. Orbit Type



The bar chart displays the success rates for different orbit types:

- **Orbits with 100% Success Rate**:ES-L1, GEO, HEO, SSO: These orbits have achieved a perfect success rate in launches.
- **Orbit with 0% Success Rate**: SO, This orbit has not had any successful launches.
- **Orbits with Success Rates Between 50% and 85%**: GTO, ISS, LEO, MEO, PO, These orbits have moderate success rates, ranging from 50% to 85%.
- This plot allows for a clear comparison of success rates across various orbit types, highlighting the orbits where SpaceX has consistently achieved successful launches and those where there is room for improvement.

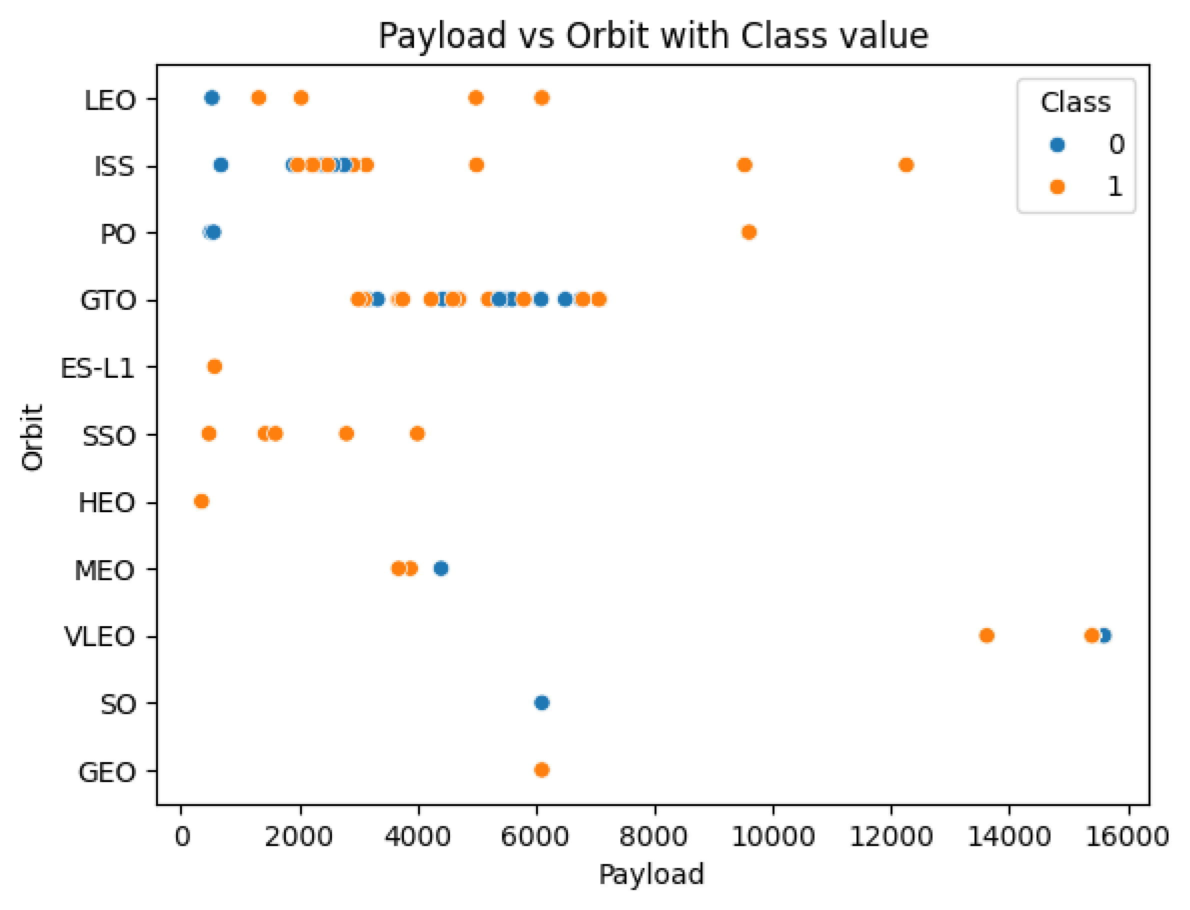
Flight Number vs. Orbit Type



The scatter plot shows the relationship between flight number and orbit type, with points colored to indicate the success or failure of the launch.

- **LEO Orbit**: Success appears to be related to the number of flights, with an increasing success rate as the number of flights grows.
- **GTO Orbit**: There seems to be no clear relationship between the flight number and success rate, with successes and failures distributed across the flight numbers.
- **SSO and HEO Orbits**: These orbits show a high success rate, which may be attributed to the knowledge and experience gained from previous launches in other orbits.

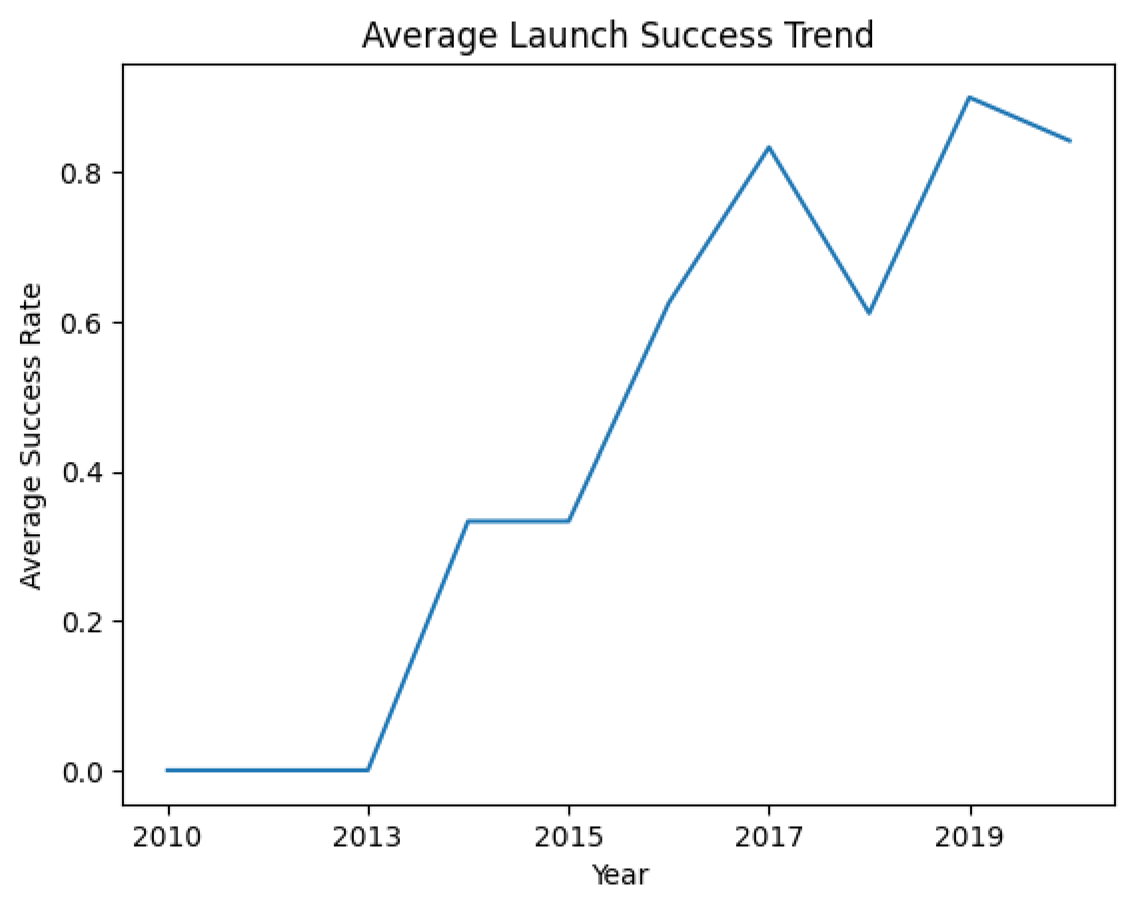
Payload vs. Orbit Type



- **LEO Orbit**: Heavier payloads tend to improve the success rate for launches in the LEO orbit, as indicated by the prevalence of orange markers for higher payloads.
- **GTO Orbit**: For the GTO orbit, success is more likely with lighter payloads, as shown by the higher number of successful markers at lower payload masses.
- **ISS Orbit:** Similar to LEO, heavier payloads in the ISS orbit seem to have a positive influence on the success rate.

In summary, the plot suggests that payload weight significantly influences launch success rates in different orbits. Heavier payloads improve success rates in LEO and ISS orbits, while lighter payloads are more favorable for success in GTO orbits.

Launch Success Yearly Trend



The chart displays the average launch success rate of SpaceX rockets over time, from 2010 to 2020.

- **2010 to 2013**: The success rate remained very low, indicating that early launches had a high failure rate.
- **2013 to 2015**: There was a noticeable increase in the success rate, reflecting improvements in launch technology and processes.
- **2015 to 2018**: The success rate continued to rise sharply, reaching over 60%, showcasing significant advancements and increased reliability in SpaceX launches.
- **2018 to 2020**: The success rate peaked at over 80%, although there was a slight dip before maintaining a high success rate.

In summary, since 2013, SpaceX has seen a steady increase in its rocket launch success rate, demonstrating ongoing improvements and growing expertise in their launch operations.

All Launch Site Names

```
%sql SELECT DISTINCT "LAUNCH_SITE" FROM SPACEXTBL
```

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

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

- This demonstrates how the DISTINCT keyword can be used to filter out duplicate records and display only the unique entries from a specified column.
- The query outputs the names of the unique launch sites involved in SpaceX missions

Launch Site Names Begin with 'CCA'

```
[ ] %sql SELECT * FROM SPACEXTBL WHERE "LAUNCH_SITE" LIKE '%CCA%' LIMIT 5

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

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	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
2012-10-08	0:35:00	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

- **Filtering with LIKE Clause**: The WHERE clause followed by the LIKE clause filters launch sites that contain the substring 'CCA'.
- **LIMIT Clause**: The LIMIT 5 clause restricts the result to the first 5 records that meet the filtering criteria.
- This query effectively filters and displays a subset of records where the launch site name includes 'CCA', providing a quick overview of relevant launch details.

Total Payload Mass

```
[ ]  %sql SELECT SUM("PAYLOAD_MASS__KG_") FROM SPACEXTBL WHERE "CUSTOMER" = 'NASA (CRS)'

  * sqlite:///my_data1.db
 Done.
 SUM("PAYLOAD_MASS__KG_")
 45596
```

- **SUM Function**: The SUM function is used to calculate the total payload mass.
- **Filtering with WHERE Clause**: The WHERE clause filters records where the customer is 'NASA (CRS)'.
- The output displays the total payload mass, which is 45,596 kg. This query effectively summarizes the payload mass for all missions conducted by NASA (CRS), providing a single value that represents the total weight carried by these specific launches.

Average Payload Mass by F9 v1.1

```
%sql SELECT AVG("PAYLOAD_MASS__KG_") FROM SPACEXTBL WHERE "Booster_Version" LIKE '%F9 v1.1%'
```

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

AVG("PAYLOAD_MASS__KG_")
2534.6666666666665

- **AVG Function**: The AVG function is used to calculate the average payload mass.
- **Filtering with LIKE Clause**: The WHERE clause followed by the LIKE clause filters records where the booster version contains the substring 'F9 v1.1'.
- The output displays the average payload mass, which is approximately 2534.67 kg. This query effectively calculates and displays the average weight carried by the booster version F9 v1.1, providing insight into the typical payload capacity for this specific booster.

First Successful Ground Landing Date

```
%sql SELECT MIN(DATE) FROM SPACEXTBL WHERE "Landing_Outcome"  LIKE '%Success%

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

MIN(DATE)
2015-12-22

- **Filtering with LIKE Clause**: The WHERE clause filters the dataset to include only records where the landing outcome contains the word 'Success’.
- **MIN Function**: The MIN function is used to select the record with the oldest (earliest) date among the filtered records.
- The output displays the date of the first successful landing outcome, which is December 22, 2015.

Successful Drone Ship Landing with Payload between 4000 and 6000

```
%sql SELECT "Booster_Version" FROM SPACEXTBL WHERE "Landing_Outcome" = 'Success (drone ship)' AND "PAYLOAD_MASS__KG_" BETWEEN 4000 AND 6000
```

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

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

- **Filtering with WHERE and AND Clauses**: The WHERE clause filters the dataset to include only records where the landing outcome is 'Success (drone ship)'. The AND clause further filters these records to include only those with a payload mass between 4000 and 6000 kg.
- This query effectively identifies the specific booster versions that have successfully landed on a drone ship while carrying payloads within the specified weight range.

Total Number of Successful and Failure Mission Outcomes

```
%%sql SELECT (SELECT COUNT("MISSION_OUTCOME") FROM SPACEXTBL WHERE "MISSION_OUTCOME" LIKE '%Success%') AS SUCCESS,
(SELECT COUNT("MISSION_OUTCOME") FROM SPACEXTBL WHERE "MISSION_OUTCOME" LIKE '%Failure%') AS FAILURE
```

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

SUCCESS	FAILURE
100	1

- **Subqueries**: The query contains two subqueries:
  - The first subquery counts the number of records where the mission outcome contains the word 'Success'.
  - The second subquery counts the number of records where the mission outcome contains the word 'Failure'.
- **COUNT Function**: The COUNT function is used to count the number of records that meet the specified criteria.
- **Filtering with LIKE Clause**: The WHERE clause followed by the LIKE clause filters the mission outcomes to include only those that contain 'Success' or 'Failure'.
- The output displays the total number of successful missions (100) and the total number of failed missions (1). This query effectively summarizes the mission outcomes, providing a clear count of successes and failures.

Boosters Carried Maximum Payload

```
%%sql SELECT DISTINCT "BOOSTER_VERSION" FROM SPACEXTBL
WHERE "PAYLOAD_MASS__KG_" = (SELECT max("PAYLOAD_MASS__KG_") FROM SPACEXTBL)
```

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

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

- Subquery with MAX Function: The subquery calculates the maximum payload mass using the MAX function.
- Main Query with Subquery Result: The main query uses the result of the subquery to filter records where the payload mass equals the maximum value.
- DISTINCT Keyword: The SELECT DISTINCT clause ensures that only unique booster versions are returned.
- This query effectively identifies and lists all unique booster versions that have carried the heaviest payloads, providing insight into the boosters capable of handling the largest payloads.

2015 Launch Records

```
%%sql SELECT substr("DATE", 6, 2) AS MONTH, "Booster_Version", "Launch_Site" FROM SPACEXTBL
WHERE "Landing_Outcome" = 'Failure (drone ship)' and substr("DATE",0,5) = '2015'

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

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

- **Filtering with WHERE Clause**: The query filters records where the landing outcome is 'Failure (drone ship)' and the date is in the year 2015.
- **SUBSTR Function**: The substr function extracts parts of the date to get the month and year.
  - substr("DATE", 6, 2) AS MONTH extracts the month from the date.
  - substr("DATE", 0, 5) checks if the year is 2015.
- This query effectively lists the specific details of failed landings on a drone ship for the year 2015, providing insights into the months, booster versions, and launch sites associated with these failures.

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

```
%%sql SELECT Landing_Outcome, COUNT(*) AS Outcome_Count
FROM SPACEXTABLE
WHERE Date BETWEEN '2010-06-04' AND '2017-03-20'
GROUP BY Landing_Outcome
ORDER BY Outcome_Count DESC;
```

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

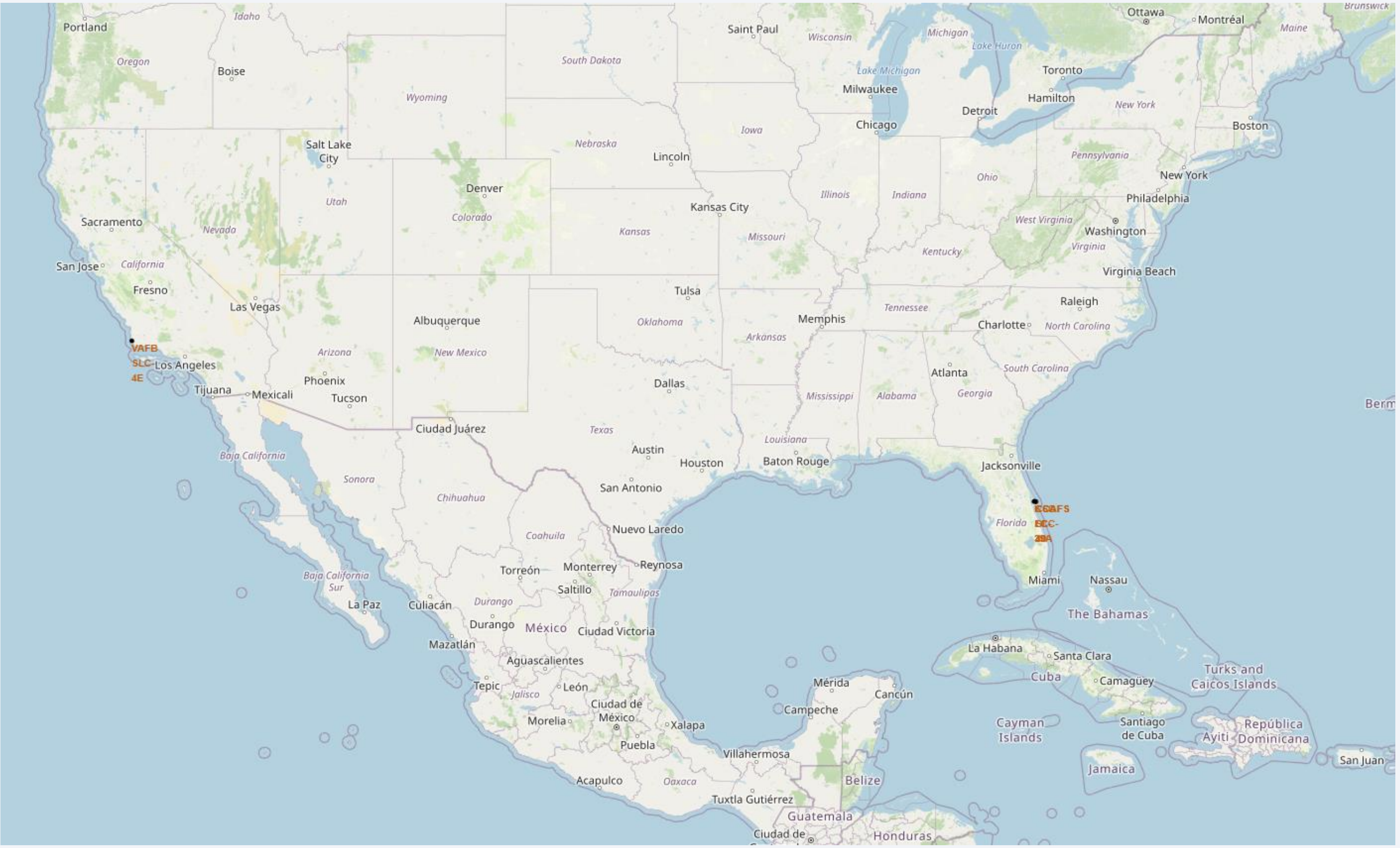
Landing_Outcome	Outcome_Count
No attempt	10
Success (drone ship)	5
Failure (drone ship)	5
Success (ground pad)	3
Controlled (ocean)	3
Uncontrolled (ocean)	2
Failure (parachute)	2
Precluded (drone ship)	1

- **Filtering with WHERE Clause**: The WHERE clause filters records to include only those with dates between June 4, 2010, and March 20, 2017.
- GROUP BY Clause: The GROUP BY clause groups the results by landing outcome.
- **COUNT Function**: The COUNT function counts the number of occurrences of each landing outcome.
- **ORDER BY Clause**: The ORDER BY Outcome_Count DESC clause sorts the results in descending order based on the count of each landing outcome.
- This query effectively ranks the landing outcomes by their frequency within the specified date range, providing a clear view of the most common and least common landing results during that period.

Section 3

Launch Sites Proximities Analysis

Folium Map – Launch sites locations

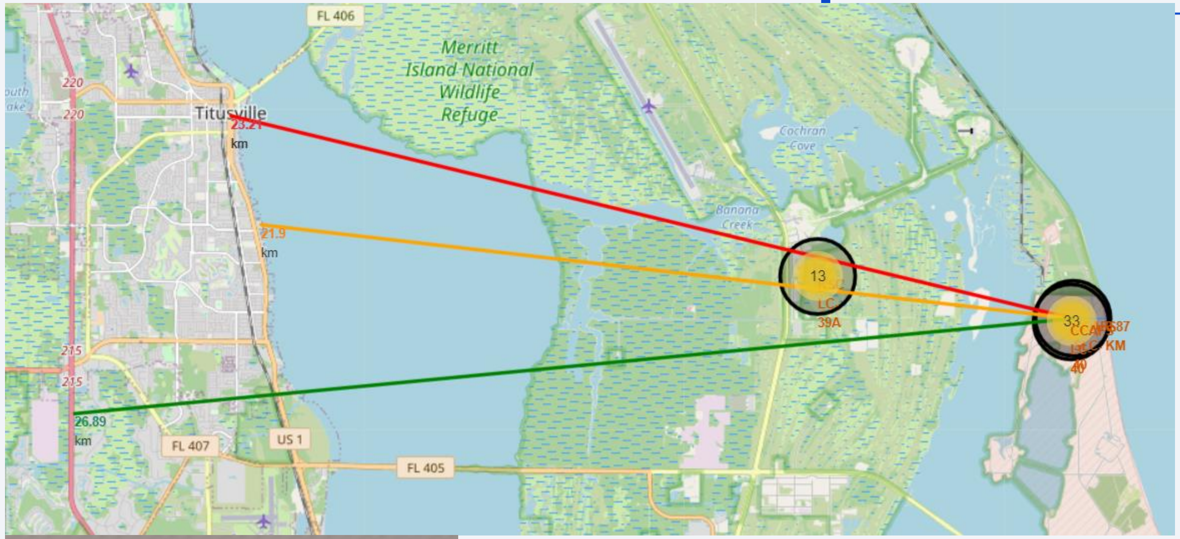


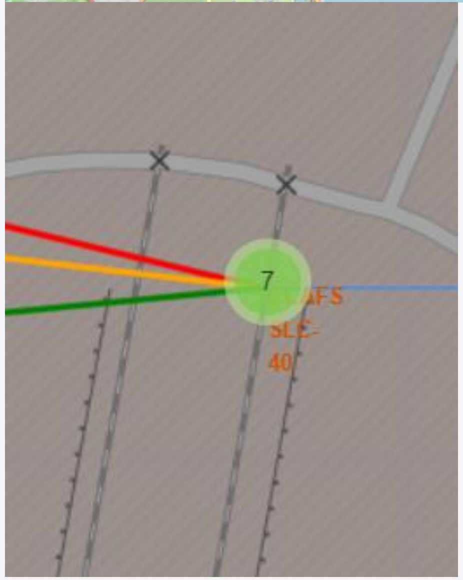
- We see that Space X launch sites are located on the coast of the United States, minimizing the risk of potential failures or explosions.
- Most launch sites are located near the Equator. Launching from the equator utilizes the rotational speed due to inertia, helping spacecraft achieve and maintain orbital velocity.

Folium Map - Color-labeled launch outcomes

Green markers represent successful launches, while red markers indicate unsuccessful launches. These color-labeled markers make it easy to identify which launch sites have relatively high success rates.

Folium Map - distances between CCAFS SLC-40 launch site to its proximities



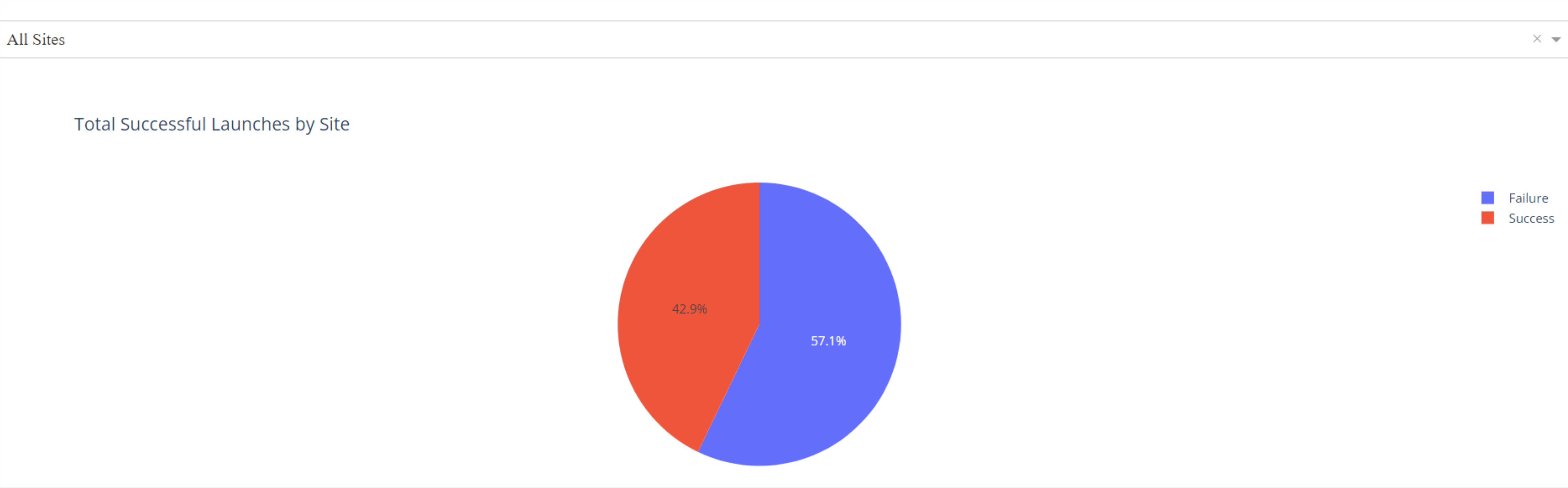


- CCAFS SL-C40 is near to: Railways, highways and coastline.
- CCAFS SL-C40 doesn't keep certain distance away from cities.

Section 4

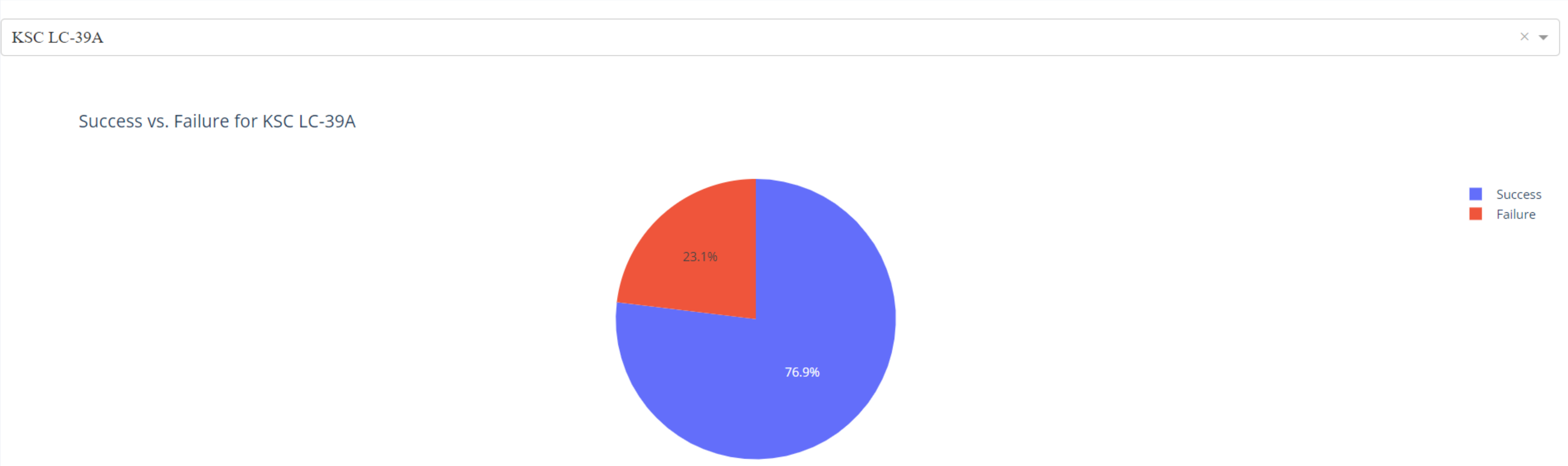
Build a Dashboard with Plotly Dash

Dashboard - Launch success count for all sites



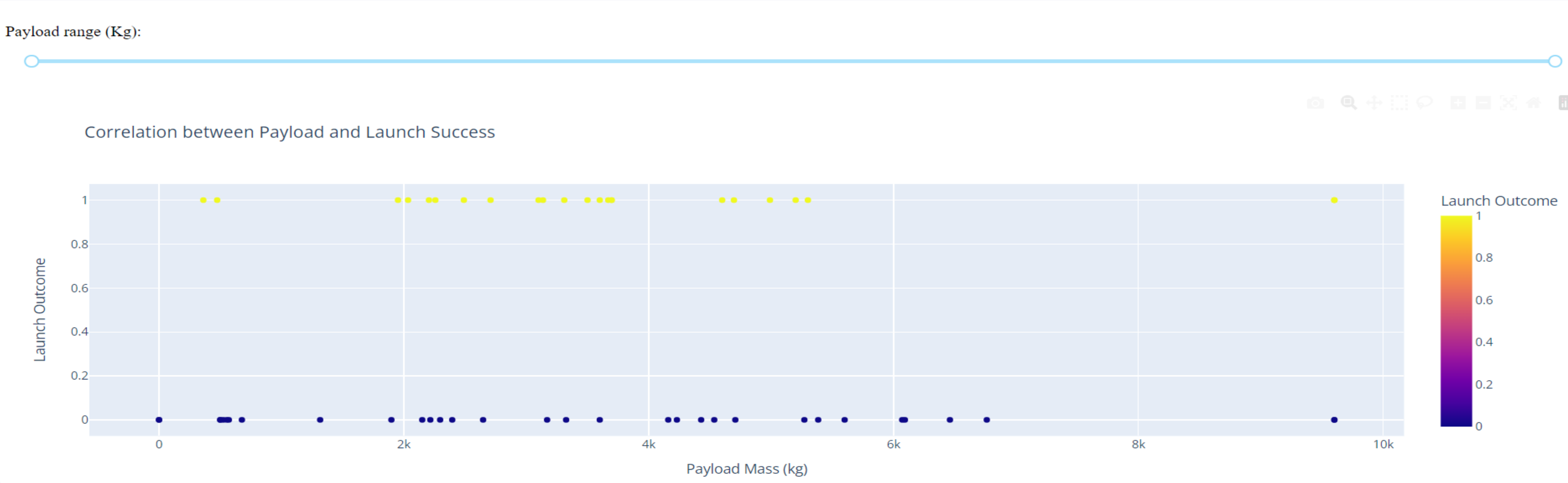
- Total Success: 57.1%
- Total Failure: 42.9%

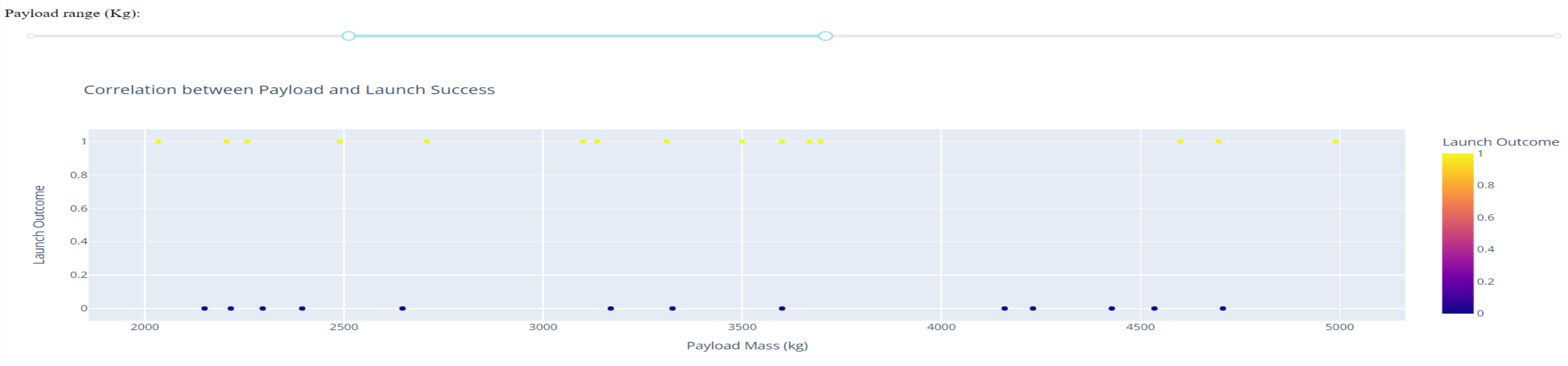
Dashboard - KSC LC-39A successful rate



KSC LC-39A has the highest launch success rate with 76.9% and the failure rate is 23.1%,

Dashboard - Payload vs. Launch Outcome scatter plot



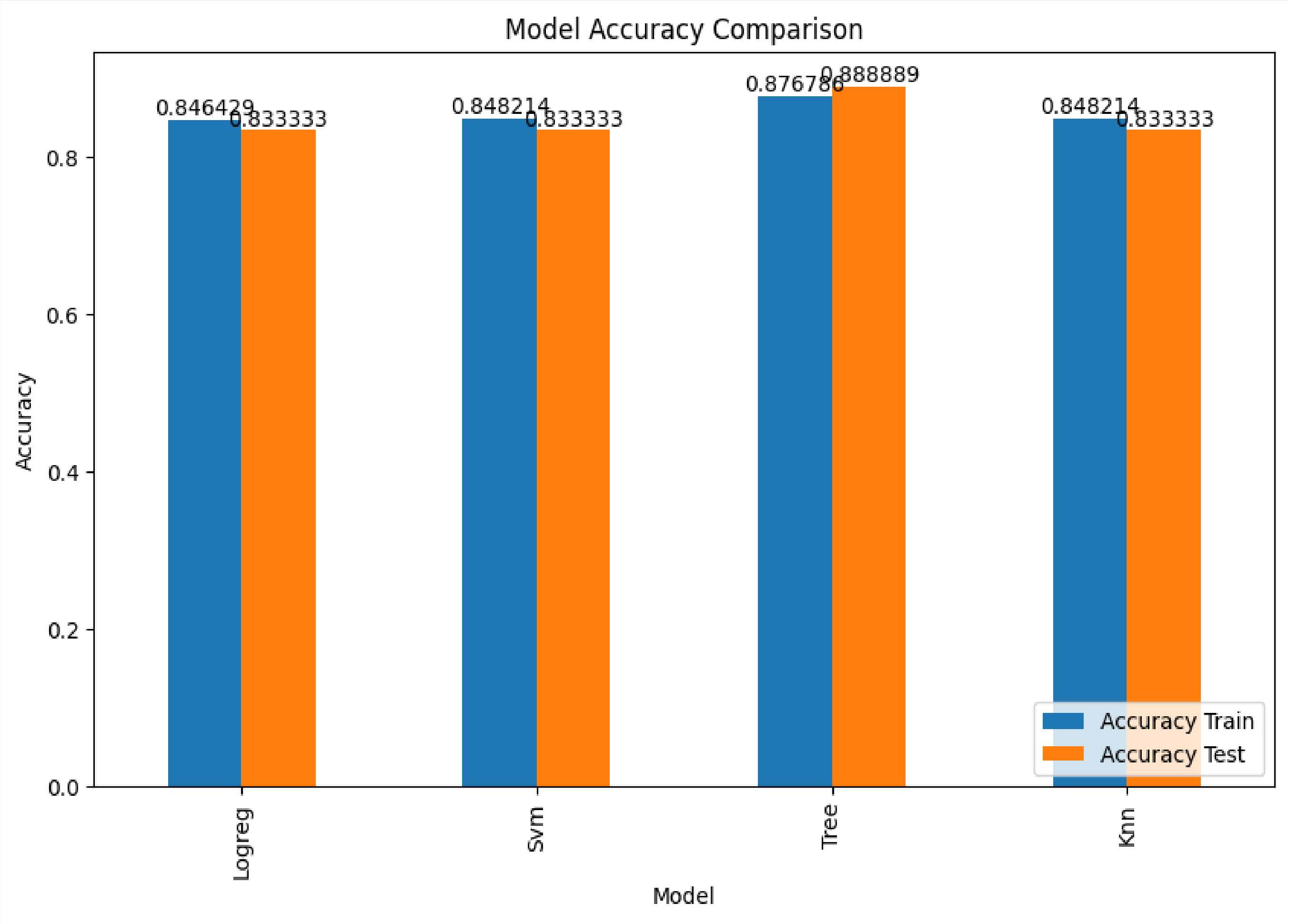


The charts show that payloads between 2000 and 5500 kg have the highest success rate.

Section 5

Predictive Analysis (Classification)

Classification Accuracy

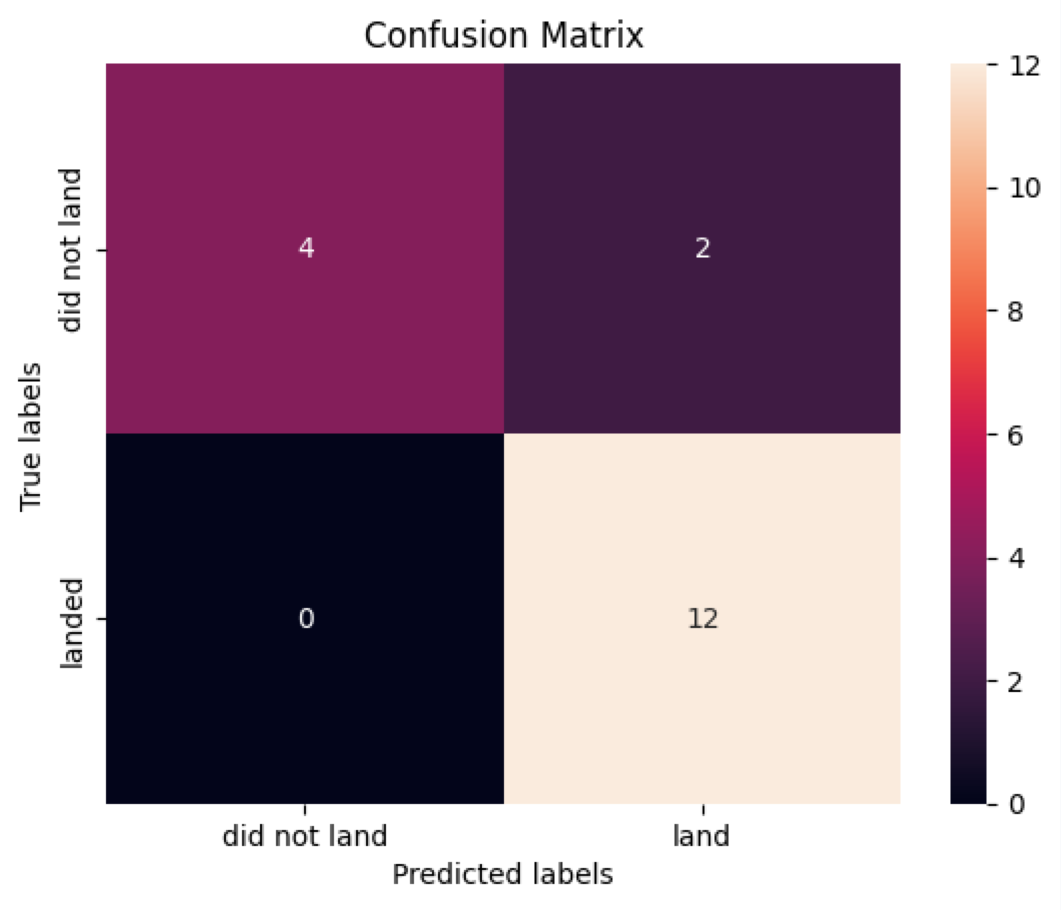


Key observations:

- **Logreg**: Training accuracy is 0.846429, and test accuracy is 0.833333.
- **SVM**: Training accuracy is 0.848214, and test accuracy is 0.833333.
- **Tree**: Training accuracy is 0.876786, and test accuracy is 0.888889, indicating the highest test accuracy among the models.
- **KNN**: Training accuracy is 0.848214, and test accuracy is 0.833333.

The Decision Tree model shows the highest accuracy on the test data, suggesting it might be the most effective model for this specific dataset. The other models have similar accuracy, with slight variations between training and test performance.

Confusion Matrix



Key observations from the matrix:

- **True Positive (landed and predicted landed):** 12 instances where the model correctly predicted the rocket would land.
- **True Negative (did not land and predicted did not land):** 4 instances where the model correctly predicted the rocket would not land.
- **False Positive (did not land but predicted landed):** 2 instances where the model incorrectly predicted the rocket would land when it did not.
- **False Negative (landed but predicted did not land):** 0 instances where the model incorrectly predicted the rocket would not land when it actually did.

This indicates that the model performs well in predicting landings, with no false negatives and a small number of false positives.

Conclusions (1/2)

Based on the analysis of the dataset and the evaluation of various machine learning models, several key conclusions can be drawn:

- **Best Performing Model:** The Decision Tree model is identified as the best algorithm for this dataset, achieving the highest accuracy among the evaluated models. Despite similar test accuracies, the Decision Tree model's higher training accuracy makes it the preferred choice.
- **Launch Success Factors:**
  - Payload Mass: Launches with lower payload mass tend to have better success rates compared to those with larger payload masses.
  - Launch Site: Most launch sites are near the Equator and close to the coast, with KSC LC-39A having the highest success rate among all sites.
  - Orbit Type: Orbits such as ES-L1, GEO, HEO, and SSO show a 100% success rate, indicating their reliability for successful missions.
- **Trends Over Time:** The success rate of launches has increased over the years, likely due to accumulated knowledge and improvements in technology and processes.

Conclusions (2/2)

- **Confusion Matrix Insights:** The Decision Tree model shows strong performance with 12 true positives, 4 true negatives, 2 false positives, and 0 false negatives, indicating its effectiveness in predicting successful landings accurately.
- **Further Analysis:** While the current data provides significant insights, further investigation into atmospheric conditions or other relevant factors could help explain why certain launch sites perform better than others.

In summary, the Decision Tree model is the most effective for this dataset, and factors such as payload mass, launch site proximity to the Equator and coast, and specific orbits significantly influence the success rates of SpaceX missions.

Thank you!