IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

José Joel García Alicea

June 2023

Outline

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

Executive Summary

In the rapidly evolving era of affordable space travel, where companies like SpaceX are revolutionizing the industry, the race is on for a new player, Space Y, to challenge the status quo. Imagine harnessing the power of data science to predict whether SpaceX's first stage will be reused, enabling Space Y to determine launch prices and compete with the titan of space exploration. This is the captivating journey our Capstone Project embarks on, unravelling the secrets of rocket reusability and propelling Space Y into the forefront of the commercial space age.

Methodologies

- Data collection from API and web scraping.
- Data wrangling.
- Exploratory data analysis (EDA) using SQL, Pandas and Matplotlib.
- Interactive visual analytics and dashboard with Folium and Ploty Dash.
- Predictive analysis (classification).

Results

- The best hyperparameters for Logistic Regression, SVM, Decision Trees and KNN classifiers.
- The method that best performs using test data.

Introduction

The objectives and approach of this Capstone project is focused on predicting the first stage for rocket launches, with aim of determining the price of each launch for a new rocket company, Space Y. This project involves gathering information about SpaceX, the leading player in the commercial space industry, and utilizing data science techniques to predict weather the first stage will be reusable or not.

- The primary objective of this Capstone is to develop a machine learning model that can accurately predict the reusability of the first stage of for SpaceX rocket launches. By leveraging public information and training the model, the project seeks to provide insights into weather SpaceX will be able to reuse the first stage, enabling Space Y to determine the cost of each launch and compete with SpaceX in the commercial space sector.

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
    - Data was collected using SpaceX API and web scraping.
- Perform data wrangling
    - The data processing involved converting the initial JSON object and HTML table formats into a Pandas dataframe, enabling streamlined visualization and analysis of the data.
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
    - To build, tune, evaluate classification models for predictive analysis, I employed machine learning techniques to predict the successful landings of SpaceX's first stage rockets.

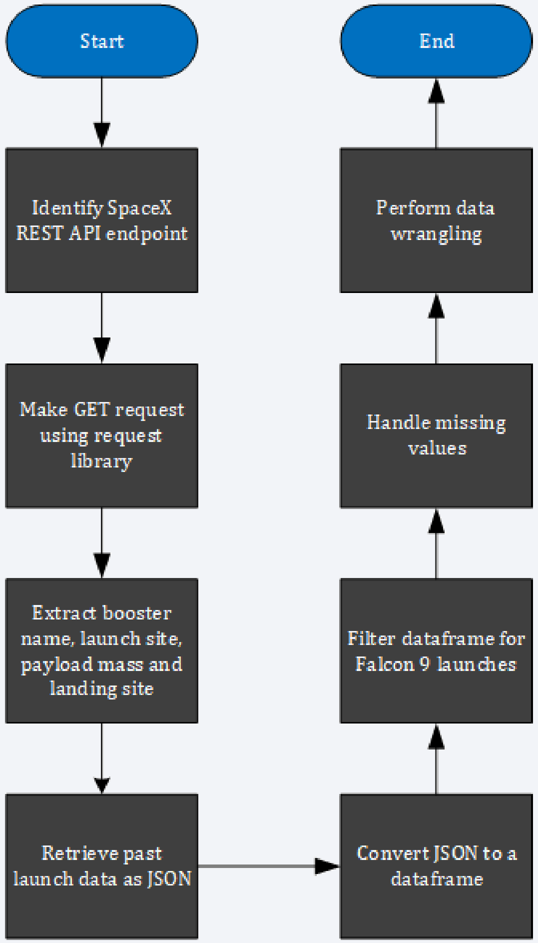
Data Collection

The data collection process involved employing diverse methodologies to gather relevant information in a comprehensive manner.

- Data collection was conducted by leveraging GET requests to access the SpaceX API.
- We decoded the response content as JSON using .json() and transformed it into a Pandas dataframe using .json_normalize() for streamline analysis.
- We conducted web scraping of Falcon 9 launch records from Wikipedia using BeautifulSoup, a Python library know for its web parsing capabilities.
- The objective encompassed extracting launch records from Wikipedia as an HTML table, parsing the table content, and subsequently transforming it into a Pandas dataframe for subsequent analysis and exploration.

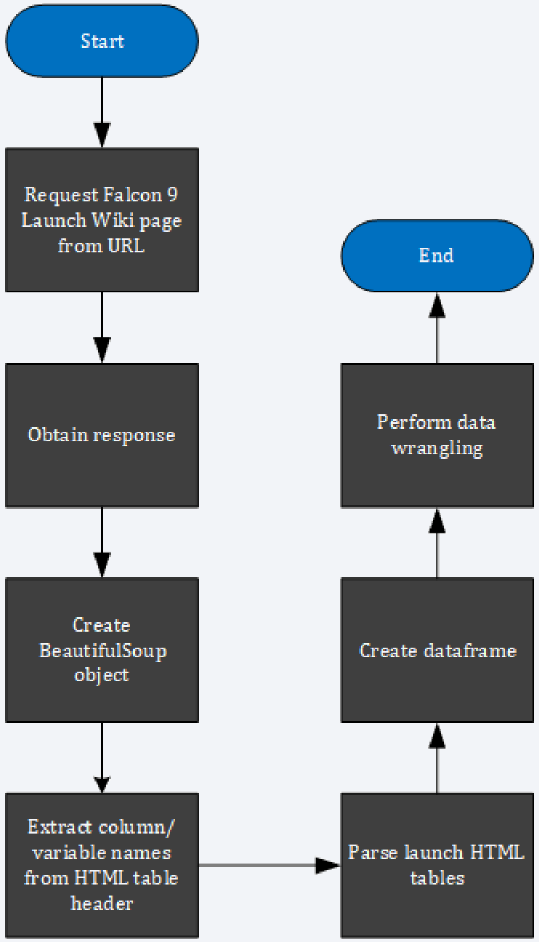
Data Collection – SpaceX API

- https://github.com/cronosenergy/IBM-DATA-SCIENCE-CAPSTONE-PROJECT/blob/dba909772649dda2f760a24bde528e6585c51374/IBM%20DATA%20SCIENCE%20CAPSTONE%20PART%201%20(SPACE%20X%20DATA%20COLLECTION%20API).ipynb



Data Collection - Scraping

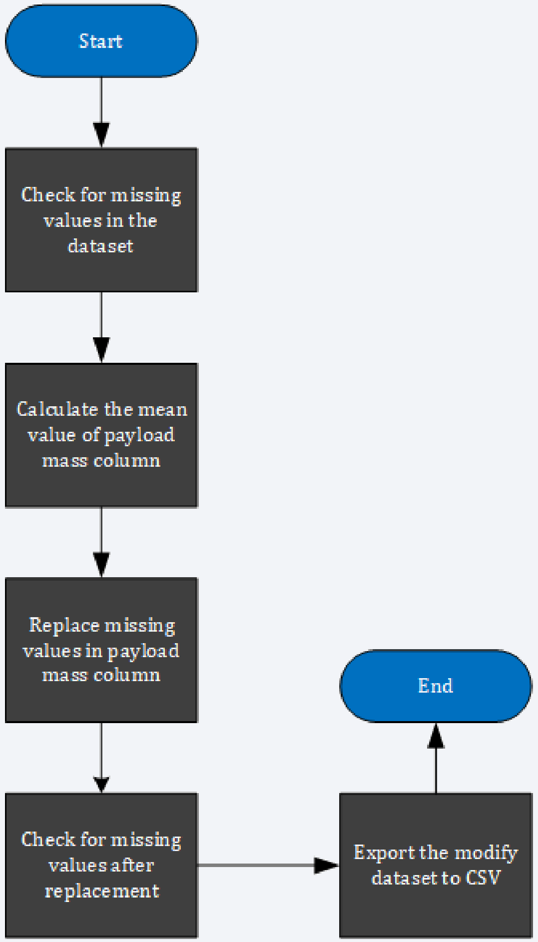
- https://github.com/cronosenergy/IBM-DATA-SCIENCE-CAPSTONE-PROJECT/blob/dba909772649dda2f760a24bde528e6585c51374/IBM%20DATA%20SCIENCE%20CAPSTONE%20PART%201%20(SPACE%20X%20DATA%20COLLECTION%20API).ipynb



Data Wrangling

We conducted exploratory data analysis to gain insights into the dataset and determined the training labels. This involved calculating the number of launches at each site, as well as the frequency and occurrence of different orbits. Additionally, we created a landing outcome label based on the outcome column. The resulting analysis was exported to a CSV file for further utilization.

- https://github.com/cronosenergy/IBM-DATA-SCIENCE-CAPSTONE-PROJECT/blob/dba909772649dda2f760a24bde528e6585c51374/IBM%20DATA%20SCIENCE%20CAPSTONE%20PART%20II%20(SPACE%20X%20DATA%20WRANGLING).ipynb



EDA with Data Visualization

1. Flight Number vs. Payload Mass:

- Scatter plot used to visualize the relationship.
- Flight Number on the x-axis and Payload Mass on the y-axis.
- Used to analyse the impact of Flight Number and Payload Mass on first stage landing success.

2. Launch Site Analysis:

- Bar chart used to compare success rates among different launch sites.
- Launch sites on the x-axis and success rate on the y-axis.
- Helps identify variations in success rates across different launch sites

EDA with Data Visualization

3. Flight Number vs. Launch Site:

- Scatter plot used to visualize the relationship.

- Flight Number on the x-axis and Launch Site on the y-axis.

- Intended to explore any relationship between Flight Number and Launch Site.

- Conclusion: No clear pattern observed in the relationship between Flight Number and Launch Site

4. Flight Number vs. Orbit Type:

- Scatter plot used to visualize the relationship.

- Flight Number on the x-axis and Orbit on the y-axis.

- Intended to examine the relationship between Flight Number and Orbit Type.

- Conclusion: In the LEO (Low Earth Orbit) category, the success rate appears to be related to the number of flights. However, no clear relationship between flight number and orbit type is observed in the GTO (Geostationary Transfer Orbit).

EDA with Data Visualization

5. Success Rate by Orbit:

- Bar chart used to represent success rates for each orbit type.
- Orbits on the x-axis and success rate on the y-axis.
- Helps identify the orbits with high success rates.

6. Flight Number vs. Orbit:

- Scatter plot used to visualize the relationship.
- Flight Number on the x-axis and Orbit type on the y-axis.
- Examines the relationship between Flight Number and Orbit type.

EDA with Data Visualization

7. Payload vs. Orbit:

- Scatter plot used to visualize the relationship.
- Payload Mass on the x-axis and Orbit type on the y-axis.
- Explores the impact of Payload Mass on success rates for different orbits.

8. Annual Success Rate Trend:

- Line chart used to show the average success rate trend over the years.
- Years on the x-axis and success rate on the y-axis.
- Illustrates the overall trend in success rates over time.

- https://github.com/cronosenergy/IBM-DATA-SCIENCE-CAPSTONE-PROJECT/blob/dba909772649dda2f760a24bde528e6585c51374/IBM%20DATA%20SCIENCE%20CAPSTONE%20PART%20IV%20(EDA%20DATA%20VISUALIZATION).ipynb

EDA with SQL

1. Retrieve the names of the unique launch sites in the space mission.

- SQL Query: SELECT DISTINCT launch_site FROM SPACEXTBL;

2. Retrieve 5 records where the launch sites begin with the string 'CCA'.

- SQL Query: SELECT * FROM SPACEXTBL WHERE launch_site LIKE 'CCA%' LIMIT 5;

3. Display the total payload mass carried by boosters launched by NASA (CRS).

- SQL Query: SELECT SUM(payload_mass__kg_) AS total_payload_mass FROM SPACEXTBL WHERE customer = 'NASA (CRS)';

EDA with SQL

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

- SQL Query: SELECT AVG(payload_mass__kg_) AS average_payload_mass FROM SPACEXTBL WHERE booster_version = 'F9 v1.1';

5. List the date when the first successful landing outcome in ground pad was achieved.

- SQL Query: SELECT MIN(date) AS first_successful_landing_date FROM SPACEXTBL WHERE landing_outcome LIKE 'Success (ground pad)%';

6. List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000.

- SQL Query: SELECT booster_serial FROM SPACEXTBL WHERE landing_outcome LIKE 'Success (drone ship)%' AND payload_mass__kg_ > 4000 AND payload_mass__kg_ < 6000;

EDA with SQL

6. List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000.

- SQL Query: SELECT booster_serial FROM SPACEXTBL WHERE landing_outcome LIKE 'Success (drone ship)%' AND payload_mass__kg_ > 4000 AND payload_mass__kg_ < 6000;

7. List the total number of successful and failure mission outcomes.

- SQL Query: SELECT mission_outcome, COUNT(*) AS count FROM SPACEXTBL GROUP BY mission_outcome;

8. List the names of the booster versions which have carried the maximum payload mass. (Using a subquery)

- SQL Query: SELECT booster_version FROM SPACEXTBL WHERE payload_mass__kg_ = (SELECT MAX(payload_mass__kg_) FROM SPACEXTBL);

EDA with SQL

9. List the records that display the month names, failure landing outcomes in drone ship, booster versions, and launch sites for the months in the year 2015.

- SQL Query: SELECT substr(Date, 4, 2) AS month, landing_outcome, booster_version, launch_site FROM SPACEXTBL WHERE substr(Date, 7, 4) = '2015' AND landing_outcome LIKE 'Failure (drone ship)%’;

10. Rank the count of successful landing outcomes between the date 04-06-2010 and 20-03-2017 in descending order.

- SELECT landing_outcome, COUNT(*) AS count FROM SPACEXTBL WHERE date BETWEEN '04-06-2010' AND '20-03-2017' AND landing_outcome LIKE 'Success%' GROUP BY landing_outcome ORDER BY count DESC;

- https://github.com/cronosenergy/IBM-DATA-SCIENCE-CAPSTONE-PROJECT/blob/dba909772649dda2f760a24bde528e6585c51374/IBM%20DATA%20SCIENCE%20CAPSTONE%20PART%20III%20(SPACE%20X%20EDA%20SQL).ipynb

Build an Interactive Map with Folium

Below is a comprehensive summary of the map objects that were crafted integrated into the Folium map:

- Circles: A blue circle was added to the map for each launch site. These circles represent a radius of 1000 meters around each launch site
- Markers: A marker was added to the map for each launch site. These markers display the name of the launch site and are represented by a small, orange-colored text icon.
- Marker Cluster: All the markers representing launch sites were grouped together using a MarkerCluster object. This cluster visually groups nearby markers together when zoomed out on the map.
- Marker Colors: Each launch site marker was assigned a color based on the success or failure of the launches from that site. Successful launches are represented by green markers, while failed launches are represented by red markers.

- https://github.com/cronosenergy/IBM-DATA-SCIENCE-CAPSTONE-PROJECT/blob/dba909772649dda2f760a24bde528e6585c51374/IBM%20DATA%20SCIENCE%20CAPSTONE%20PART%20V%20(SPACE%20X%20LAUNCH%20SITE%20LOCATION).ipynb

Build a Dashboard with Plotly Dash

The SpaceX Launch Records Dashboard is a data visualization application that provides interactive plots and controls to explore and analyse SpaceX launch data. The dashboard includes a dropdown list for selecting launch sites, a pie chart displaying success counts, a payload range slider, and a scatter chart illustrating the correlation between payload mass and launch success. These components enable users to gain insights into launch records, compare site-specific performance, and examine the relationship between payload characteristics and mission outcomes.

- Dropdown List: The dropdown list, with the id 'site-dropdown', allows the selection of a launch site. The inclusion of this interaction enables the user to filter the data based on the launch site and view site-specific visualizations.

- Pie Chart: The 'success-pie-chart' graph shows the total count of successful launches for all sites or for a specific site if chosen from the dropdown list. This plot provides an overview of the success rates and allows for easy comparison between different launch sites. It is included to provide a high-level understanding of launch success.

Build a Dashboard with Plotly Dash

- Payload Range Slider: The range slider, with the id 'payload-slider', enables the selection of a payload mass range. By adjusting the slider, the user can filter the data based on payload mass, allowing for more granular analysis. This interaction is included to explore the correlation between payload mass and launch success.

- Scatter Chart: The 'success-payload-scatter-chart' displays a scatter plot depicting the correlation between payload mass and launch success. The plot considers the filtered data based on the selected launch site and payload range. Additionally, the scatter plot distinguishes different booster version categories using colors. This visualization helps identify any patterns or trends between payload mass and launch success, considering the specified criteria.

• https://github.com/cronosenergy/IBM-DATA-SCIENCE-CAPSTONE-PROJECT/blob/dba909772649dda2f760a24bde528e6585c51374/IBM%20DATA%20SCIENCE%20CAPSTONE%20PART%20VI%20(SPACEX%20LAUNCH%20RECORDS%20DASHBOARDS).ipynb

Predictive Analysis (Classification)

In this machine learning prediction assignment, I followed a systematic approach to build, evaluate, improve, and identify the best performing classification model, employing techniques such as data preprocessing, model training, evaluation, and model improvement through hyperparameter tuning.

1. Data Preprocessing:

- Loaded the dataset using the pd.read_csv() function.
- Performed data cleaning by dropping any missing values using df.dropna().
- Encoded categorical variables using pd.get_dummies() to convert them into numerical values.
- Split the dataset into input features (X) and the target variable (y).

Predictive Analysis (Classification)

2. Data Split:

- Split the dataset into training and testing sets using train_test_split() from scikit-learn.
- Specified a test size of 0.2, meaning 20% of the data will be used for testing, and the remaining 80% for training.

3. Model Training and Evaluation:

- Chose the Random Forest Classifier as the initial model by creating an instance of RandomForestClassifier() from scikit-learn.
- Trained the model on the training data using the fit() method.
- Evaluated the model's performance on the testing data using accuracy score, precision score, recall score, and F1 score.

Predictive Analysis (Classification)

4. Model Improvement:

- Analyzed the model's performance metrics to identify areas for improvement.

- Considered different strategies to improve the model, such as adjusting hyperparameters or trying different algorithms.

- Applied hyperparameter tuning to the Random Forest model using GridSearchCV from scikit-learn to find the best combination of hyperparameters.

- Repeated the training and evaluation process with the improved model.

5. Best Model Selection:

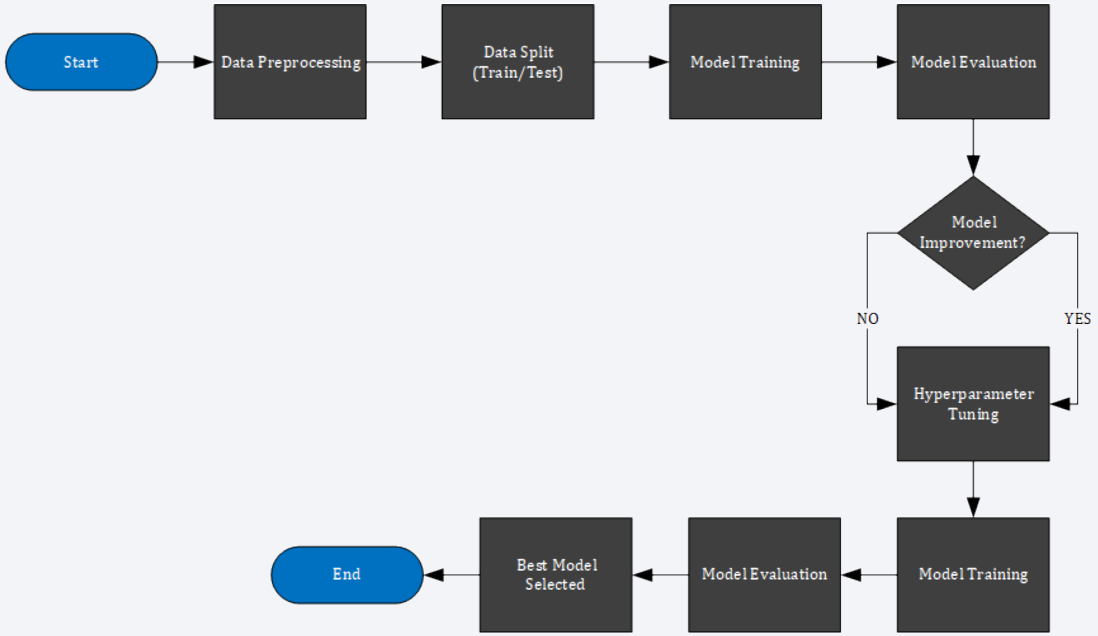
- Compared the performance of different models based on evaluation metrics such as accuracy, precision, recall, and F1 score.

- Selected the model with the best overall performance on the testing set, considering the highest scores across the metrics.

Predictive Analysis (Classification)

In this machine learning prediction assignment, I followed a systematic approach to build, evaluate, improve, and identify the best performing classification model, employing techniques such as data preprocessing, model training, evaluation, and model improvement through hyperparameter tuning.

Flowchart:



- https://github.com/cronosenergy/IBM-DATA-SCIENCE-CAPSTONE-PROJECT/blob/dba909772649dda2f760a24bde528e6585c51374/IBM%20DATA%20SCIENCE%20CAPSTONE%20PART%20VII%20(SPACEX%20MACHINE%20LEARNING%20PREDICTION).ipynb

Results

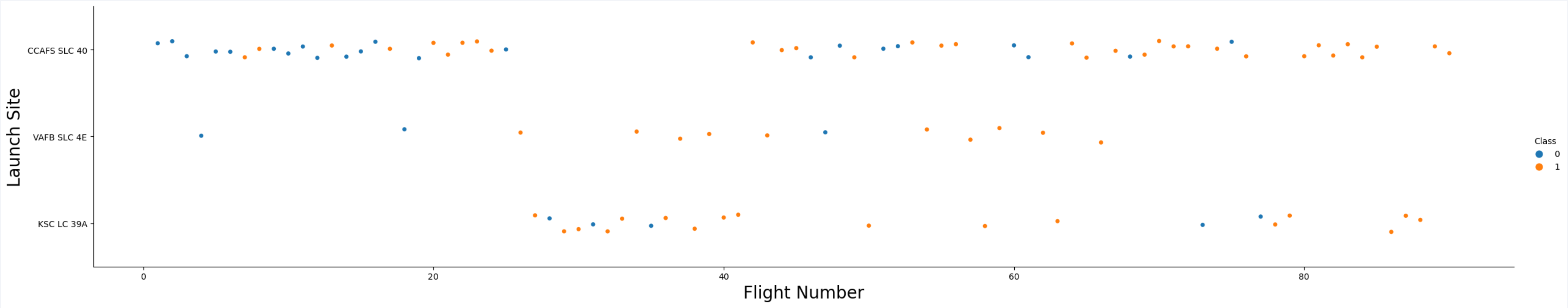
- Exploratory data analysis results
- Interactive analytics demo in screenshots
- Predictive analysis results

Section 2

Insights drawn from EDA

Flight Number vs. Launch Site

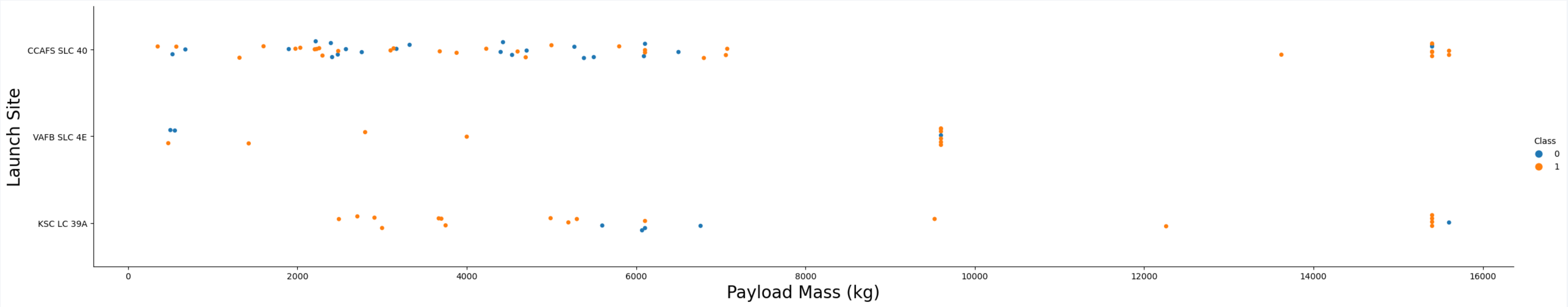
- Launch Site CCAFS SLC 40 exhibits the highest number of launches among all sites.
- VAFB SLC 4E and KSC LC 39A demonstrate a higher successful rate, despite representing only one third of the total launches.
- Remarkably, there has been a notable increase in the success rate for every launch site particularly for CCAFS SLC 40, which are concentrates the majority of launches.



Payload vs. Launch Site

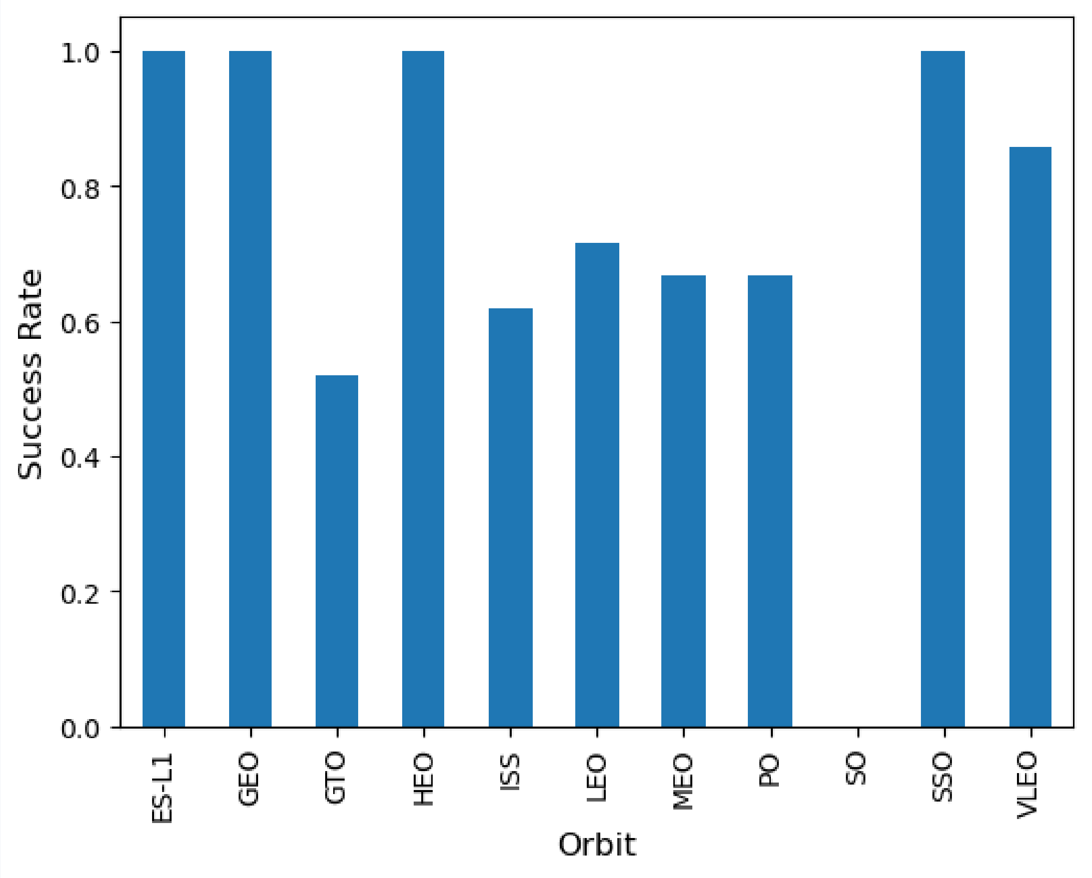
The following data provides key insights into the launch sites and their corresponding payload capacities.

- The launch site VAFB SLC 4E does not accommodate launches with a payload mass exceeding 10,000 kg.
- At KSC LC 39A launch site, no rockets have been launched with a payload mass below 2500 kg.
- CCAFS SLC launch site has launched rockets with a payload mass below 7,500 kg and above 13,000 kg, but not within the range of 7,500 kg to 13,000 kg.



Success Rate vs. Orbit Type

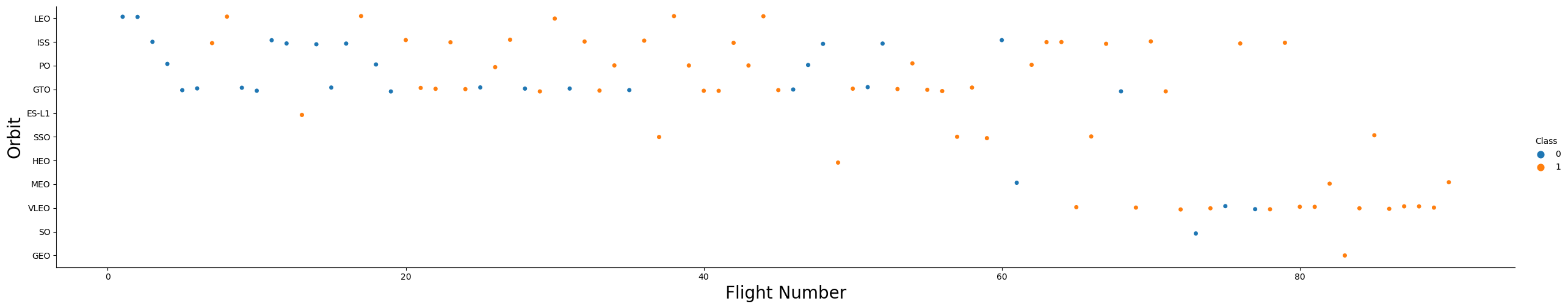
- Orbit types ES-L1, GEO, HEO, and SSO demonstrate the highest success rates.



Flight Number vs. Orbit Type

The relationship between flight number and orbit type reveals interesting insights, shedding light on the distribution and patterns of launches across various orbits.

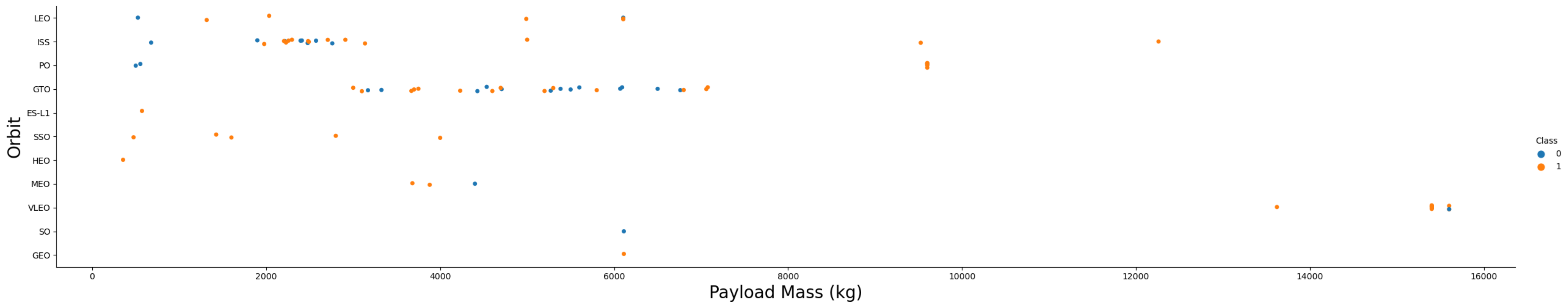
- As anticipated, the initial series of launches experienced a higher rate of failures. However, following the first 40 launches, significant improvements were observed, resulting in a 50 percent reduction in unsuccessful landings.
- The GTO and ISS orbits exhibit a notable concentration of launches, accompanied by the lowest ratio of successful landings.



Payload vs. Orbit Type

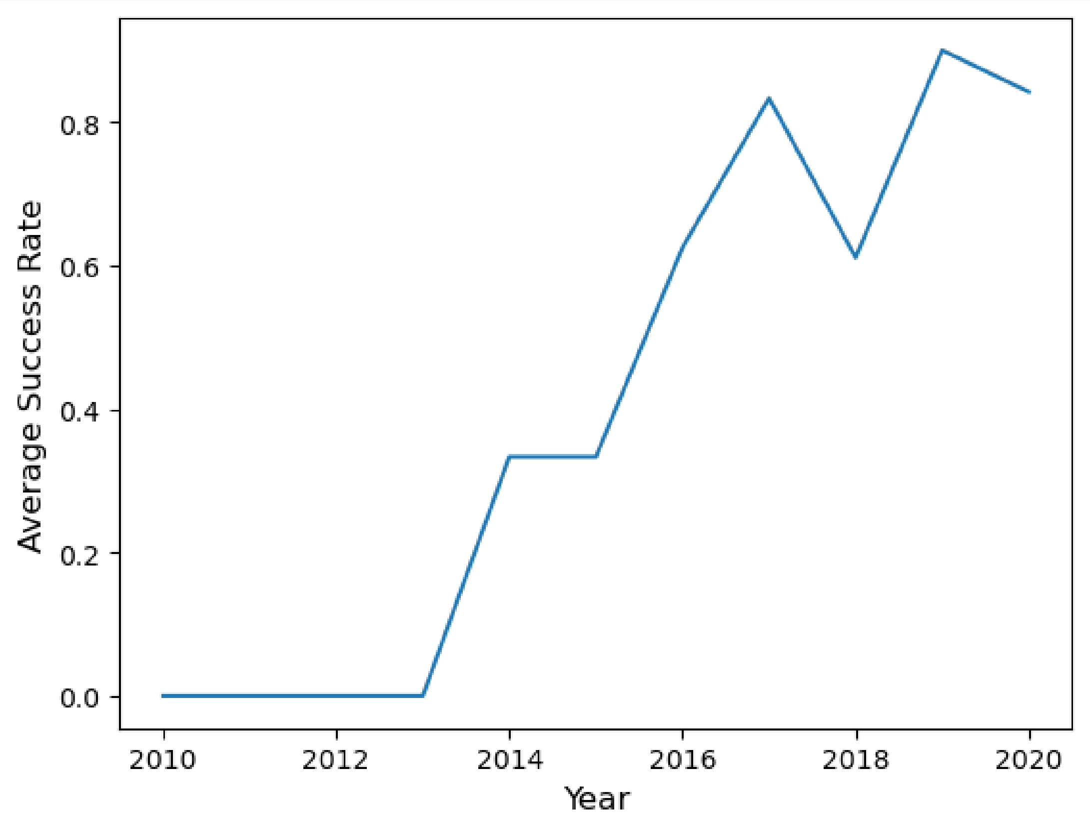
The relationship between payload mass and orbit type plays a crucial role in determining mission feasibility and success rates.

- There appears to be a discernible threshold for payload mass around 7,600kg, as only a few launches, less than 10 in total, surpass this limit.
- Heavier payloads, particularly for Polar Orbit (PO), Low Earth Orbit (LEO), and International Space Station (ISS) missions, exhibit higher rates of successful landings.



Launch Success Yearly Trend

The success rate of launches has exhibited a consistent upward trend from 2013 to 2020, reflecting the continuous improvement and advancements in the field of space exploration.



All Launch Site Names

The query "SELECT DISTINCT launch_site FROM SPACEXTBL;" retrieves the unique launch site names from the "SPACEXTBL" table in the database. The result of this query will display the names of the launch sites involved in the space missions. By using the "DISTINCT" keyword, duplicate entries of launch site names are eliminated, ensuring that only unique names are returned in the result set.

Display the names of the unique launch sites in the space mission

```
In [12]: %sql SELECT DISTINCT launch_site FROM SPACEXTBL;

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

Out[12]:

Launch_Site
CCAFS LC-40
VAFB SLC-4E
KSC LC-39A
CCAFS SLC-40
None

Launch Site Names Begin with 'CCA'

This query uses the "LIKE" operator with the pattern 'CCA%' in the "WHERE" clause to match launch sites that start with 'CCA'. The '%' symbol is a wildcard character that represents any number of characters. By executing this query, you can obtain the details of the first five records where the launch sites satisfy the specified condition, allowing you to analyse specific launches associated with launch sites starting with 'CCA'.

Display 5 records where launch sites begin with the string 'CCA'

```
In [14]: %sql SELECT * FROM SPACEXTBL WHERE launch_site LIKE 'CCA%' LIMIT 5;

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

Out[14]:

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
06/04/2010	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0.0	LEO	SpaceX	Success	Failure (parachute)
12/08/2010	15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight C1, two CubeSats, barrel of Brouere cheese	0.0	LEO (ISS)	NASA (COTS) NRO	Success	Failure (parachute)
22/05/2012	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525.0	LEO (ISS)	NASA (COTS)	Success	No attempt
10/08/2012	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500.0	LEO (ISS)	NASA (CRS)	Success	No attempt
03/01/2013	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677.0	LEO (ISS)	NASA (CRS)	Success	No attempt

Total Payload Mass

- By executing this query, it calculates the sum of the "payload_mass__kg_" column for all records where the customer is 'NASA (CRS)'. The result returned is the total payload mass carried by boosters launched by NASA under the CRS program.

Display the total payload mass carried by boosters launched by NASA (CRS)

```
In [15]: %sql SELECT SUM(payload_mass__kg_) AS total_payload_mass FROM SPACEXTBL WHERE customer = 'NASA (CRS)';

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

Out[15]:

total_payload_mass
45596.0

Average Payload Mass by F9 v1.1

By executing this query, it calculates the average payload mass for boosters of version 'F9 v1.1' by selecting the corresponding records from the table and applying the average function to the "payload_mass__kg_" column. The result returned is the average payload mass, which provides insight into the typical payload capacity of boosters with that specific version.

Display average payload mass carried by booster version F9 v1.1

```
In [16]: %sql SELECT AVG(payload_mass__kg_) AS average_payload_mass FROM SPACEXTBL WHERE booster_version = 'F9 v1.1';

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

Out[16]:

average_payload_mass
2928.4

First Successful Ground Landing Date

By executing this query, it identifies the earliest date when a successful landing outcome on a ground pad was achieved. The query searches the "landing_outcome" column for records that have a landing outcome starting with "Success (ground pad)". It then selects the minimum date from those records and renames it as "first_successful_landing_date". The result returned is the date of the first successful landing on a ground pad. This query helps to determine the initial milestone when a successful landing on a ground pad was accomplished in the space missions.

List the date when the first succesful landing outcome in ground pad was acheived.

Hint:Use min function

```
In [17]: %sql SELECT MIN(date) AS first_successful_landing_date FROM SPACEXTBL WHERE landing_outcome LIKE 'Success (ground pad)%';

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

Out[17]:

first_successful_landing_date
01/08/2018

Successful Drone Ship Landing with Payload between 4000 and 6000

By executing this query, it identifies the earliest date when a successful landing outcome on a ground pad was achieved. The query searches the "landing_outcome" column for records that have a landing outcome starting with "Success (ground pad)". It then selects the minimum date from those records and renames it as "first_successful_landing_date". The result returned is the date of the first successful landing on a ground pad. This query helps to determine the initial milestone when a successful landing on a ground pad was accomplished in the space missions.

List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000

In [13]:
```
%%sql
SELECT booster_version
FROM SPACEXTBL
WHERE payload_mass__kg_ = (
    SELECT MAX(payload_mass__kg_)
    FROM SPACEXTBL
    WHERE landing_outcome = 'Success (drone ship)'
        AND payload_mass__kg_ BETWEEN 4000 AND 6000
)
```

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

Out[13]:

Booster_Version
F9 FT B1021.2
F9 FT B1032.1
F9 B5 B1047.2

Total Number of Successful and Failure Mission Outcomes

By executing this query, it retrieves the mission outcomes recorded in the "mission_outcome" column of the SPACEXTBL table. The query then groups the records based on each unique mission outcome and calculates the count of occurrences for each outcome using the COUNT(*) function. The result returned is a summary of the total count for each mission outcome. This query provides an overview of the distribution of mission outcomes in the dataset, allowing for analysis and understanding of the frequency of different mission outcomes.

List the total number of successful and failure mission outcomes

In [14]:
```
%sql SELECT mission_outcome, COUNT(*) AS TOTAL FROM SPACEXTBL GROUP BY mission_outcome;
```

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

Out[14]:

Mission_Outcome	TOTAL
None	898
Failure (in flight)	1
Success	98
Success	1
Success (payload status unclear)	1

Boosters Carried Maximum Payload

By executing this query, it retrieves the distinct booster versions from the SPACEXTBL table. The query includes a subquery that calculates the maximum value of the "payload_mass__kg_" column from the same table. The outer query then filters the records to only include booster versions that have the maximum payload mass. The result returned is a list of unique booster versions associated with the highest payload mass recorded in the dataset. This query helps identify the specific booster versions that achieved the maximum payload mass, providing insights into the capabilities of different booster versions.

List the names of the booster_versions which have carried the maximum payload mass. Use a subquery

In [18]:

```
%%sql

SELECT DISTINCT booster_version
FROM SPACEXTBL
WHERE payload_mass__kg_ = (
    SELECT MAX(payload_mass__kg_)
    FROM SPACEXTBL
);
```

```
 * 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

By executing this query, it extracts the substring from the "Date" column to retrieve the month portion and aliases it as "month". The query filters the records based on the condition that the substring from the 7th position to the 10th position of the "Date" column should be '2015'. It further filters the records to include only those with a landing outcome that contains the phrase 'Failure (drone ship)'. The columns "landing_outcome", "booster_version", and "launch_site" are selected for display. This query helps identify the specific months in 2015 when there were failures in landing on the drone ship, along with the corresponding booster version and launch site.

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.

Note: SQLLite does not support monthnames. So you need to use substr(Date, 4, 2) as month to get the months and substr(Date,7,4)='2015' for year.

In [16]:
```
%sql SELECT substr(Date, 4, 2) AS month, landing_outcome, booster_version, launch_site FROM SPACEXTBL WHERE substr(Date, 7, 4) =
```

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

Out[16]:

month	Landing_Outcome	Booster_Version	Launch_Site
10	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
04	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

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

By executing this query, it retrieves the landing outcomes from the "landing_outcome" column and calculates the count of each unique landing outcome using the COUNT(*) function. The query filters the records based on the condition that the "date" falls between '04-06-2010' and '20-03-2017'. It further filters the records to include only those with a landing outcome that starts with 'Success'. The results are then grouped by the landing outcome and ordered in descending order based on the count. This query provides a summary of the count of successful landing outcomes within the specified date range, helping to analyze the frequency of successful landings during that period.

Rank the count of successful landing_outcomes between the date 04-06-2010 and 20-03-2017 in descending order.

In [17]:
```
%sql SELECT landing_outcome, COUNT(*) AS count FROM SPACEXTBL WHERE date BETWEEN '04-06-2010' AND '20-03-2017' AND landing_outcom
```

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

Out[17]:

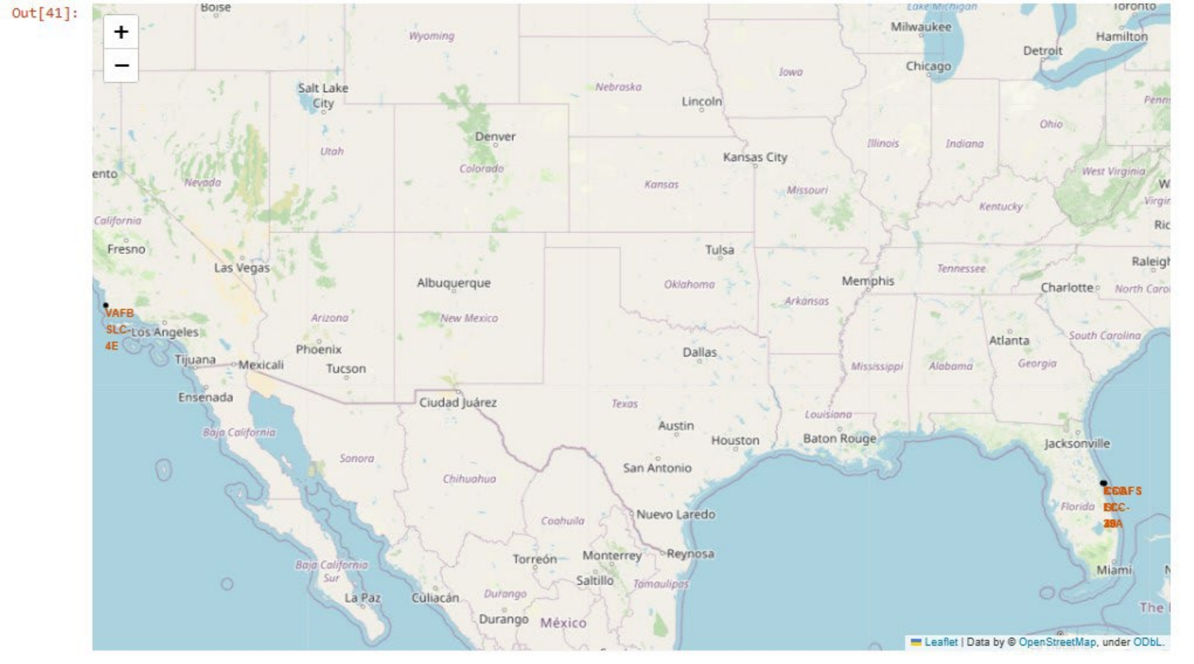
Landing_Outcome	count
Success	20
Success (drone ship)	8
Success (ground pad)	7

Section 3

Launch Sites Proximities Analysis

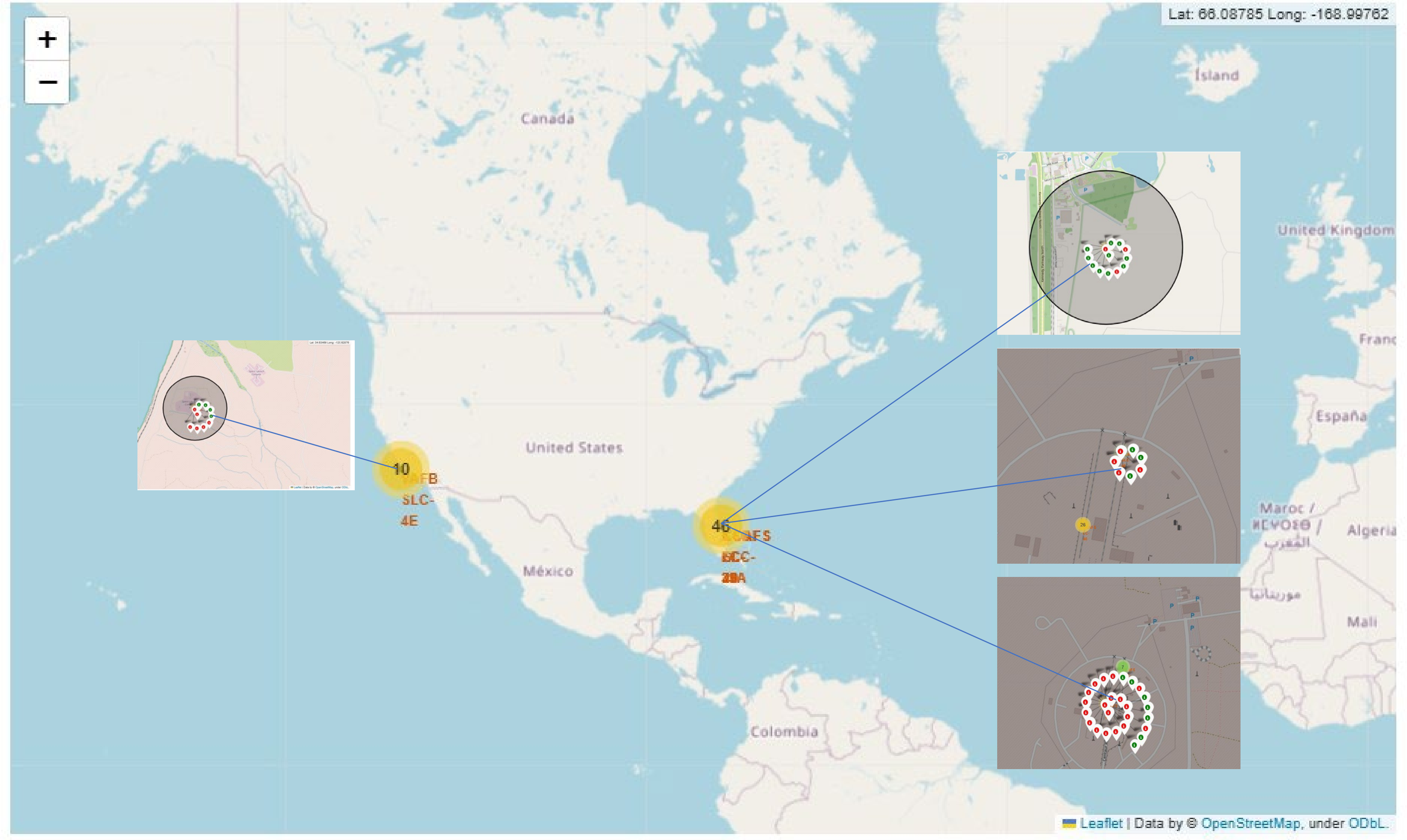
Launch Sites Map

The launch sites are strategically located in close proximity to coastal areas, primarily to ensure safety considerations. However, they are also carefully situated within reasonable distances from road and rail networks.



Success/Failed Launches for Each Site

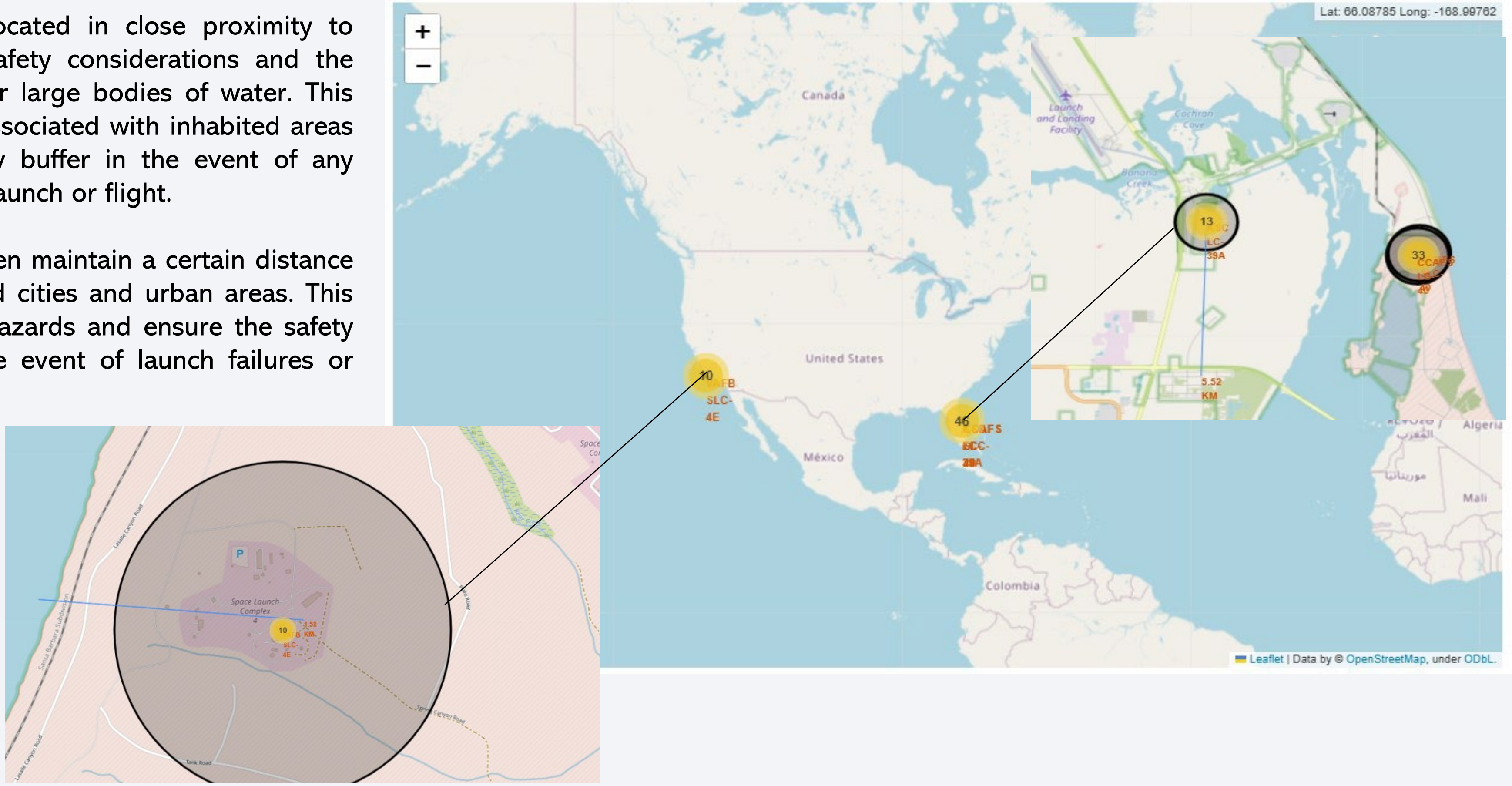
Green markers denote successful outcomes, while red markers signify failures.



Launch Site To Its Proximities

Launch sites are typically located in close proximity to coastlines due to various safety considerations and the ability to launch rockets over large bodies of water. This reduces the potential risks associated with inhabited areas and provides a larger safety buffer in the event of any unforeseen incidents during launch or flight.

Furthermore, launch sites often maintain a certain distance away from densely populated cities and urban areas. This helps to mitigate potential hazards and ensure the safety of nearby populations in the event of launch failures or accidents.

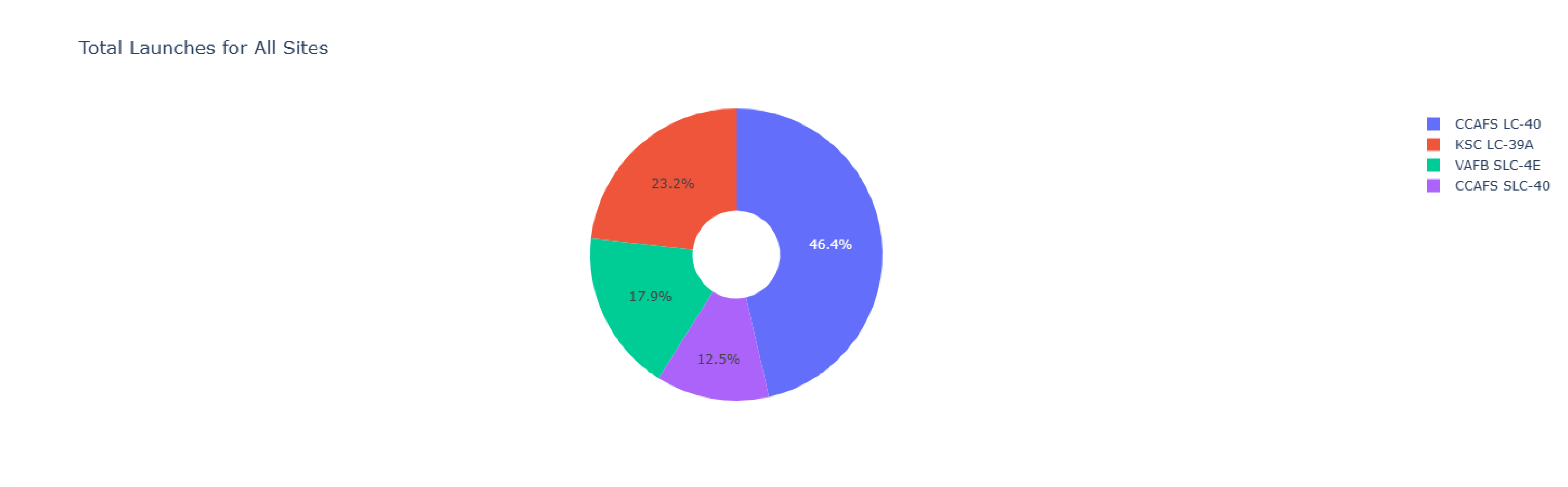


Section 4

Build a Dashboard with Plotly Dash

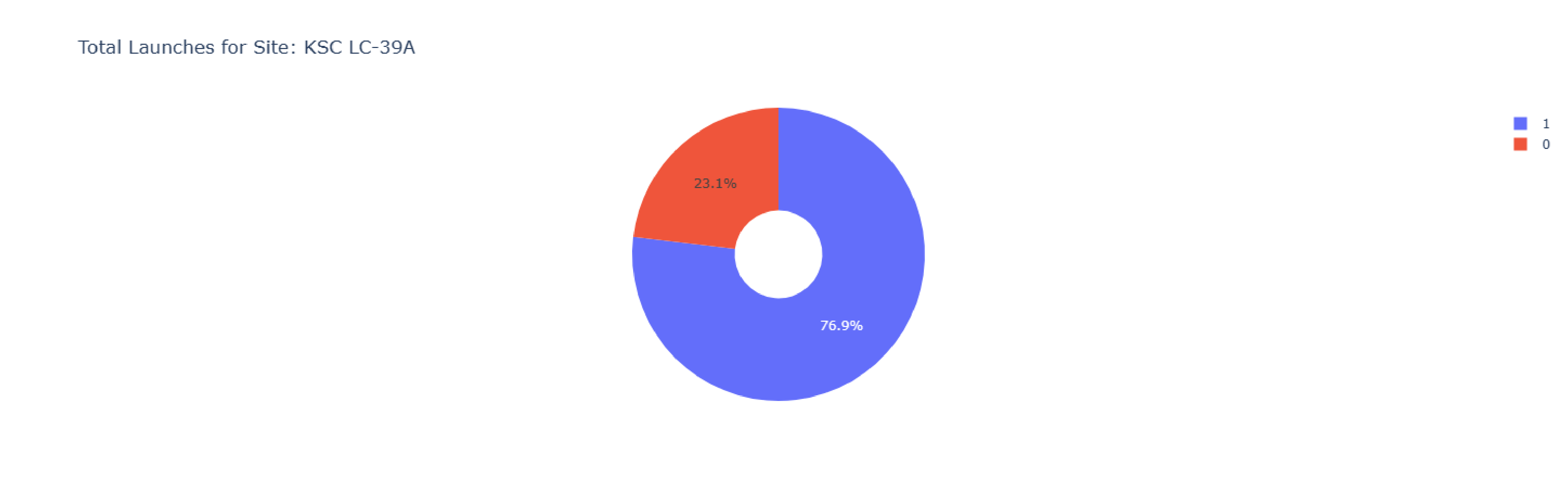
Launch Success Statistics Across All Sites

Launch site CCAF LC-40 has established itself as the foremost launch site based on its exceptional performance and track record.



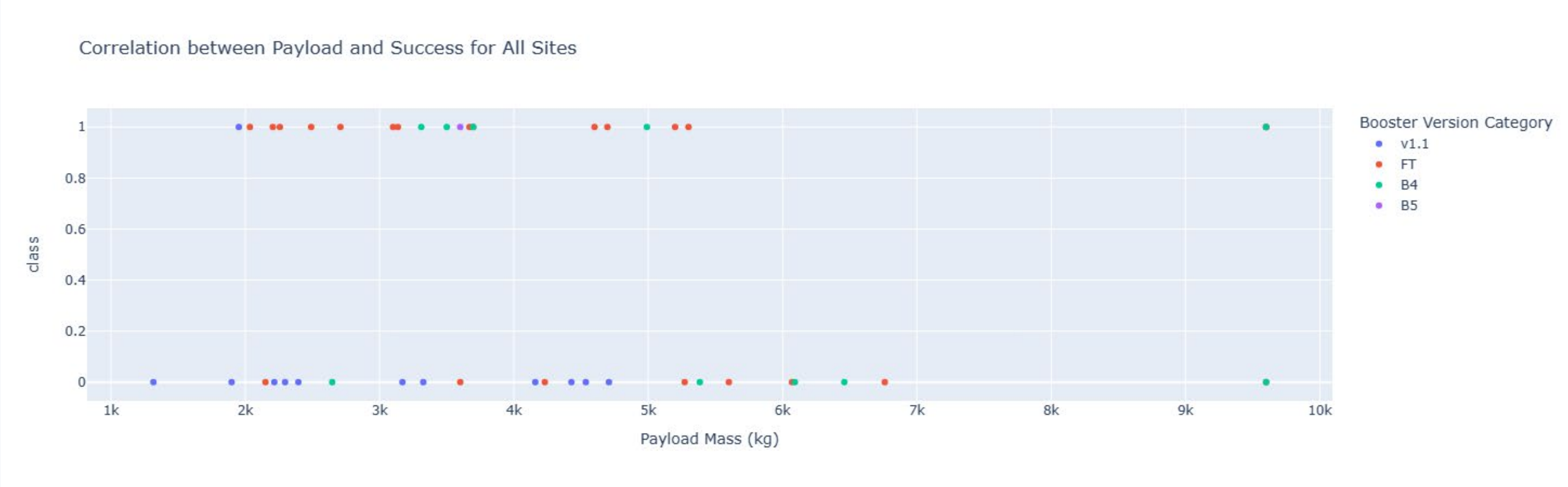
Top Performing Launch Site Based on Launch Success Ratio

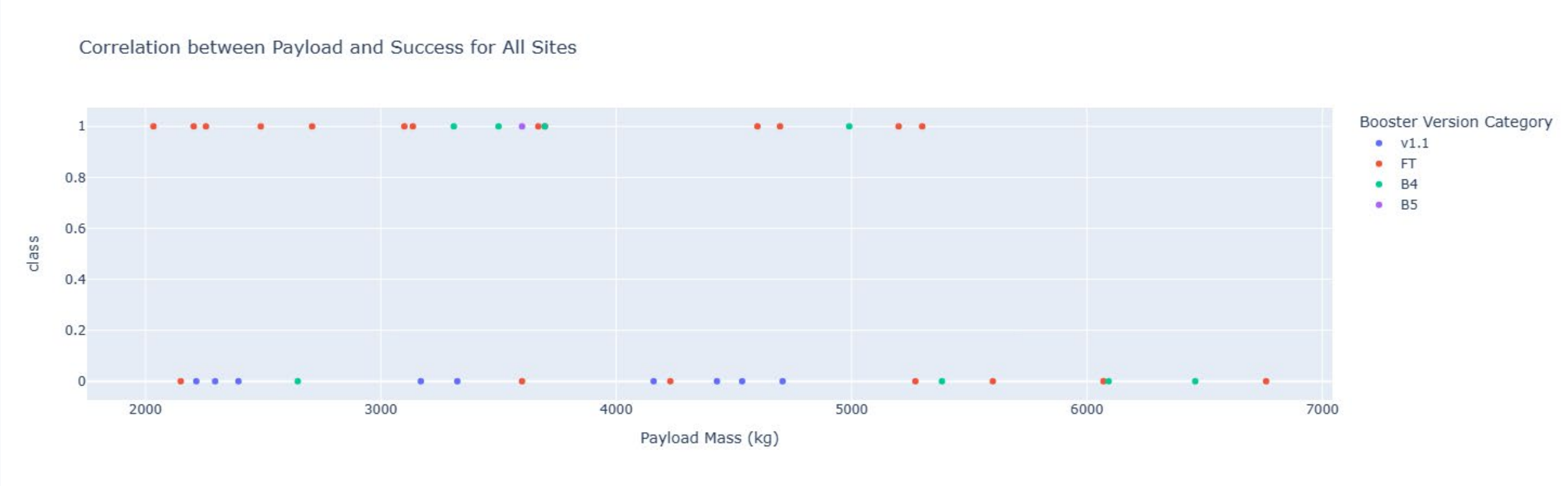
Launch site KSC LC-39A has demonstrated the highest rate of successful launches. .



Correlation Between Payload and Launch Outcome

The booster version FT boasts the highest success rate within the payload range of 2000 - 5000kg, making it a top-performing choice in terms of reliability and successful payload deployments.



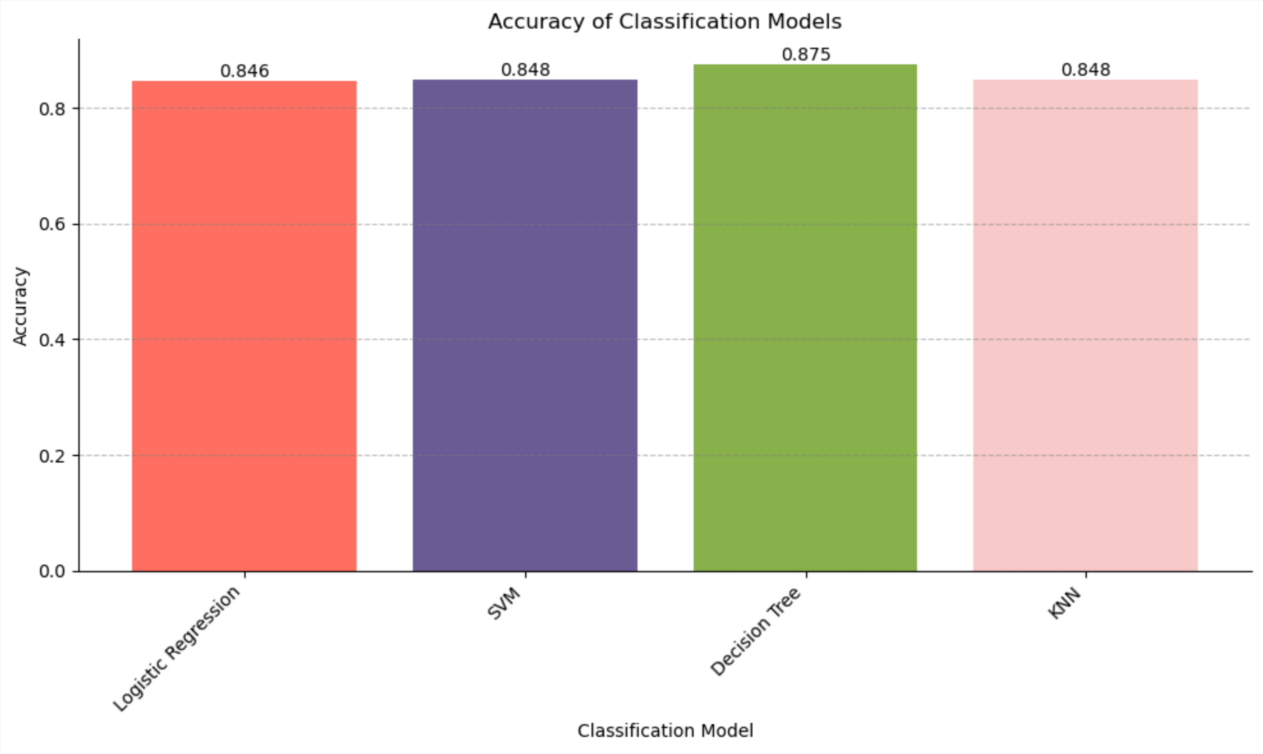


Section 5

Predictive Analysis (Classification)

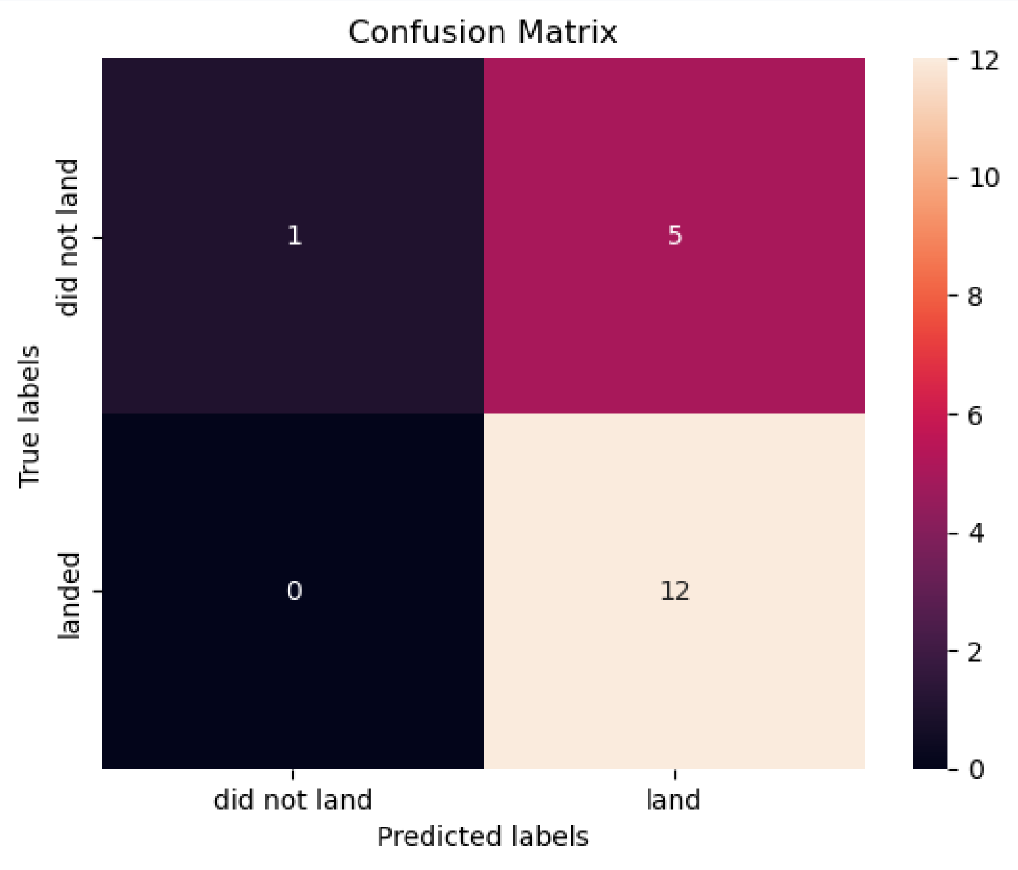
Classification Accuracy

Based on the evaluation results, the decision tree model exhibits the highest accuracy among the classification models.



Confusion Matrix

The Decision Tree Classifier demonstrates its high accuracy, as evidenced by the significant numbers of true positives and true negatives in the confusion matrix, outweighing the occurrences of false positives and false negatives.



Conclusions

- Best Launch Site: KSCLC-39A shows superior performance in mission success rates, safety, and reliability.
- Payload Weight: Launches above 7,000kg are associated with lower risks, indicating the importance of robust rocket systems.
- Improved Landing Outcomes: Successful landing outcomes have shown a positive trend over time due to process evolution and technological advancements.
- Decision Tree Classifier: Utilizing this model enables accurate prediction of successful landings, optimizing profits and mission success.

Appendix

- The project utilized a combination of exploratory data analysis (EDA), feature engineering, model selection, training, and evaluation to gain insights from the data and develop predictive models.
- During the EDA phase, a comprehensive exploration of the dataset was conducted, leveraging statistical measures, visualizations, and summary statistics to understand the data's underlying patterns and relationships.
- Feature engineering techniques were applied to preprocess the data, including standardization and the creation of new variables to enhance the models' predictive capabilities.
- Multiple classification models, such as Logistic Regression, Support Vector Machines (SVM), Decision Trees, and K-Nearest Neighbors (KNN), were trained using the scikit-learn library. Hyperparameter tuning was performed to identify the optimal combination of model parameters.
- Evaluation of the models' performance involved assessing accuracy, precision, recall, and utilizing the confusion matrix to visualize predictions. Based on these metrics, the Decision Tree Classifier emerged as the most accurate model for predicting successful landings and optimizing profitability.
- In conclusion, the project employed robust analytical techniques to extract valuable insights from the data, resulting in the identification of the Decision Tree Classifier as a reliable model for optimizing mission success rates and resource allocation.

Thank you!