Winning Space Race with Data Science

Korkrid Kyle Akepanidtaworn
April 2025

Outline

- Executive Summary
- Introduction
- Methodology
- Results
- Conclusion

Executive Summary

- Summary of Methodologies
  - In this capstone project, we will predict if the SpaceX Falcon 9 first stage will land successfully using several machine learning classification algorithms. The main steps in this project include:
    1. Data collection, wrangling, and formatting
    2. Exploratory data analysis
    3. Interactive data visualization
    4. Machine learning prediction
- Summary of All Results
  - Our analysis indicates that certain features of the rocket launches are correlated with the outcomes, specifically whether they are successful or not. We have also determined that a decision tree may be the most effective machine learning algorithm to predict the successful landing of the Falcon 9 first stage. This conclusion is based on the observed data and the performance of various algorithms. By utilizing this approach, we can potentially improve the accuracy of our predictions and enhance the overall success rate of the launches.

Introduction

- In this capstone, we will predict if the Falcon 9 first stage will land successfully. SpaceX advertises Falcon 9 rocket launches on its website with a cost of 62 million dollars; other providers cost upward of 165 million dollars each, much of the savings is because SpaceX can reuse the first stage. Therefore, if we can determine if the first stage will land, we can determine the cost of a launch. This information can be used if an alternate company wants to bid against SpaceX for a rocket launch.
- Many unsuccessful landings are intentional. Occasionally, SpaceX conducts a controlled landing in the ocean. The primary question we aim to address is whether the first stage of a Falcon 9 rocket will land successfully, given a set of features such as payload mass, orbit type, and launch site.

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Describe how data was collected
- Perform data wrangling
  - Describe how data was processed
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - How to build, tune, evaluate classification models

Data Collection

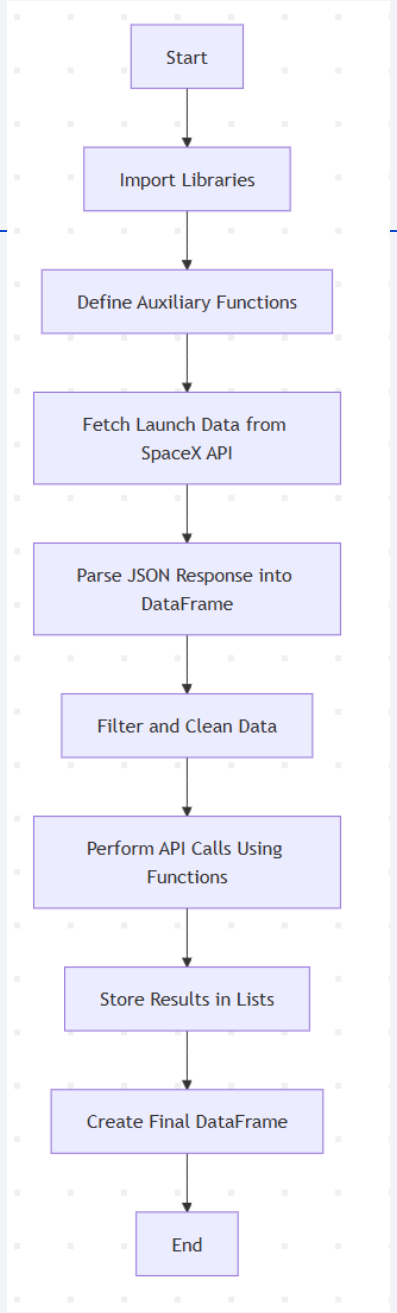
- I am making a GET request to the SpaceX API and will perform some basic data wrangling and formatting. First, I clean the requested data. To ensure the JSON results are consistent, we will use a predefined static response object for this project. Next, we decode the response content as JSON using the .json() method and convert it into a Pandas dataframe with .json_normalize(). This approach helps in maintaining the uniformity of the data structure.

DTSA-5841-IBM-Applied-DS-Capstone/module1/jupyter-labs-spacex-data-collection-api-answer.ipynb at main · korkridake/DTSA-5841-IBM-Applied-DS-Capstone

Data Collection – SpaceX API

To present the data collection process using SpaceX REST API calls, I will use key phrases to describe the steps and flowcharts to visualize the flow of data collection.

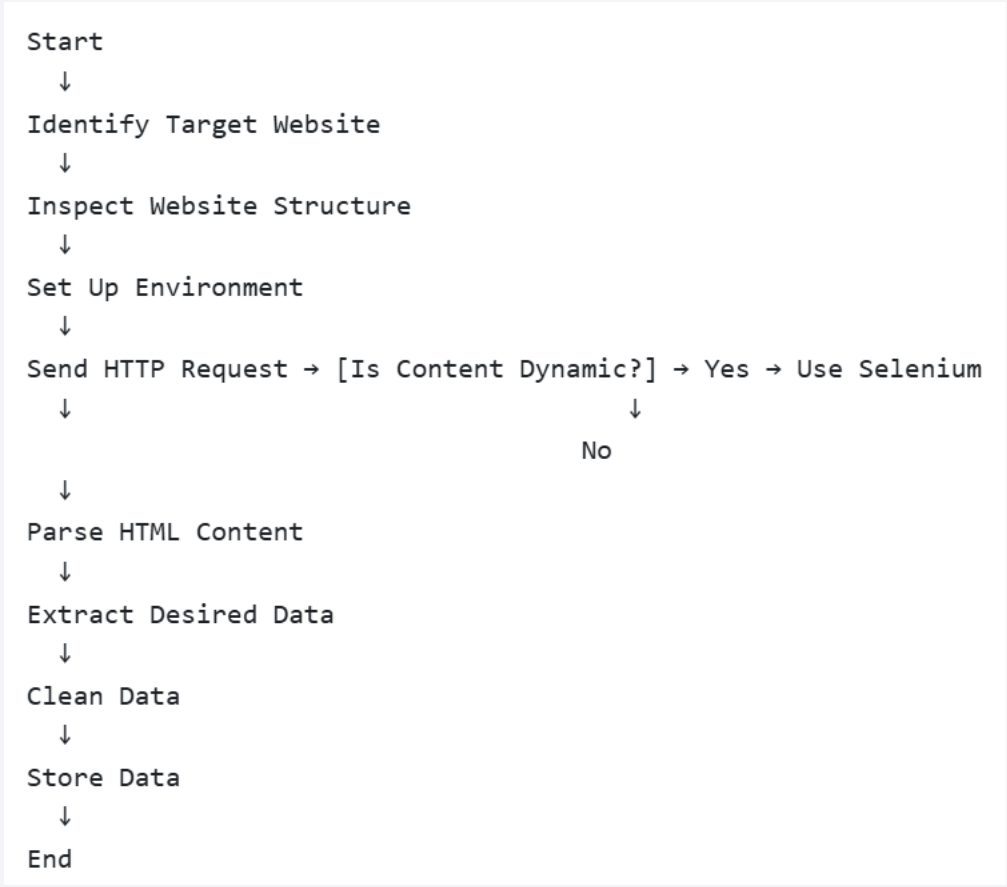
1. Use Python libraries (requests, pandas, numpy) for HTTP requests, data manipulation, and calculations.
2. Create reusable functions to call SpaceX REST API endpoints for specific data.
3. Use the SpaceX API endpoint for past launches: https://api.spacexdata.com/v4/launches/past and parse the response JSON and convert it into a pandas DataFrame.
4. Process Data. Filter rows with single payloads and single cores for simplicity. Extract relevant columns like rocket, payloads, launchpad, cores, etc.
5. Populate Data Using API Calls.
6. Combine the processed data in lists into a structured pandas DataFrame for further analysis.



DTSA-5841-IBM-Applied-DS-Capstone/module1/jupyter-labs-spacex-data-collection-api-answer.ipynb at main · korkridake/DTSA-5841-IBM-Applied-DS-Capstone

Data Collection - Scraping

- Identify Target Website: Select the website and data to scrape.
- Inspect Website Structure: Use browser developer tools to understand the HTML structure.
- Set Up Environment: Install libraries like BeautifulSoup, Selenium, or Scrapy.
- Send HTTP Request: Use libraries like requests to fetch the web page.
- Parse HTML Content: Use parsers like html.parser or lxml to extract specific HTML tags.
- Data Extraction: Identify and extract the desired data elements (e.g., text, links, images).
- Data Cleaning: Process and clean the raw extracted data.
- Store Data: Save the data into a structured format (e.g., CSV, JSON, database).
- Handle Dynamic Content: Use tools like Selenium for JavaScript-heavy pages.
- Respect Ethical Guidelines: Comply with the website's terms of service and robots.txt.



Data Wrangling

1. Data Loading and Initial Analysis: The SpaceX dataset was loaded using pandas from a CSV file. The first 10 rows were displayed to understand the data structure, including attributes like FlightNumber, Date, Orbit, Outcome, etc.
2. Handling Missing Values: The percentage of missing values in each column was calculated. Columns like LandingPad had a significant percentage of missing values (~28.89%).
3. Data Exploration: The number of launches per site (LaunchSite) and occurrences of different orbits (Orbit) were calculated using value_counts().
4. Outcome Analysis: The Outcome column was analyzed to classify mission results (e.g., True ASDS, False Ocean.)
5. Label Creation: A binary classification label, Class, was generated.

DTSA-5841-IBM-Applied-DS-Capstone/module1/labs-jupyter-spacex-Data wrangling-answer.ipynb at main · korkridake/DTSA-5841-IBM-Applied-DS-Capstone

EDA with Data Visualization

- Libraries or modules used include pandas and numpy.
- Functions from these libraries, such as value_counts(), are utilized to extract basic information from the collected data. This information encompasses the following:
  - The number of launches at each launch site
  - The frequency of each orbit
  - The number and frequency of each mission outcome

EDA with SQL

- The framework utilized for this project is IBM DB2.
- We employed the ibm_db libraries or modules.Data is queried using SQL to address several questions, such as:
    - Identifying the names of the unique launch sites in the space mission.
    - Calculating the total payload mass carried by boosters launched by NASA (CRS).
    - Determining the average payload mass carried by booster version F9 v1.1
- The SQL statements and functions used include SELECT, DISTINCT, AS, FROM, WHERE, LIMIT, LIKE, SUM(), AVG(), MIN(), BETWEEN, COUNT(), and YEAR().

Build an Interactive Map with Folium

The Folium library is used to:

- Mark all launch sites on a map
- Mark the succeeded launches and failed launches for each site on the map
- Mark the distances between a launch site to its proximities such as the nearest city, railway, or highway

These are done using functions from folium such as add_child() and folium plugins which include MarkerCluster, MousePosition, and DivIcon.

Example: A folium map showing the succeeded launches and failed launches for a specific launch site. If we zoom in on one of the launch site, we can see green and red tags. Each green tag represents a successful launch while each red tag represents a failed launch.

Build a Dashboard with Plotly Dash

- Functions from Dash are utilized to create an interactive website that allows us to adjust inputs through a dropdown menu and a range slider. The site features a pie chart and a scatterplot to display the following interactive elements:
  - The total success launches from each launch site
  - The correlation between payload mass and mission outcome (success or failure) for each launch site
- The application is launched on a terminal on the IBM Skills Network website.

Predictive Analysis (Classification)

1. Standardize the data using preprocessing.StandardScaler() from sklearn.
2. Split the data into training and test sets using train_test_split from sklearn.model_selection.
3. Create machine learning models:
    1. Logistic regression with LogisticRegression from sklearn.linear_model.
    2. Support vector machine (SVM) with SVC from sklearn.svm.
    3. Decision tree with DecisionTreeClassifier from sklearn.tree.
    4. K nearest neighbors (KNN) with KNeighborsClassifier from sklearn.neighbors.
4. Fit the models on the training set. Use GridSearchCV from sklearn.model_selection to find the best hyperparameters. Evaluate models using accuracy scores and confusion matrix from sklearn.metrics. All models have the same accuracy score and confusion matrix on the test set. Therefore, they are ranked by their GridSearchCV best scores.

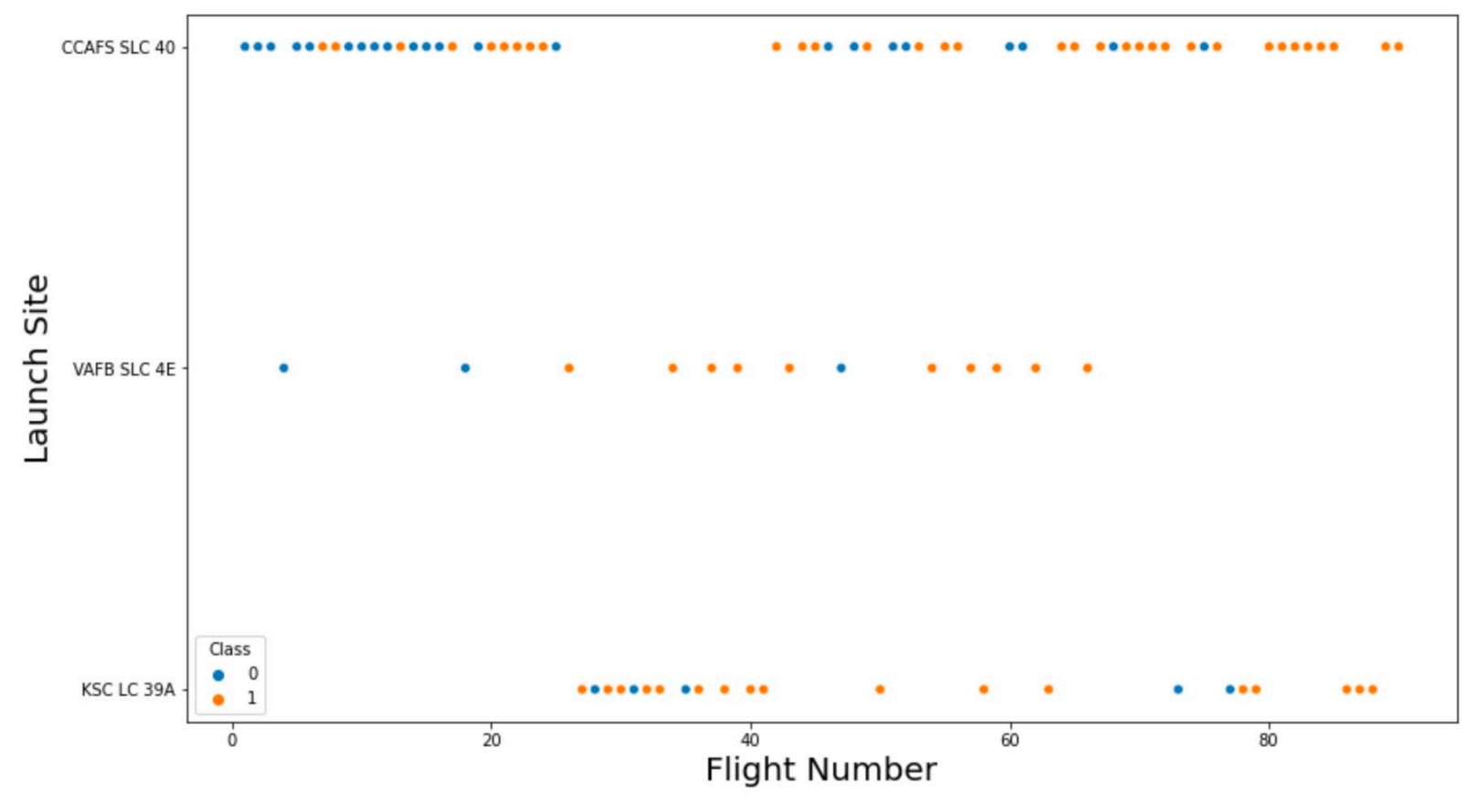
Results

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

Section 2

Insights drawn from EDA

Flight Number vs. Launch Site

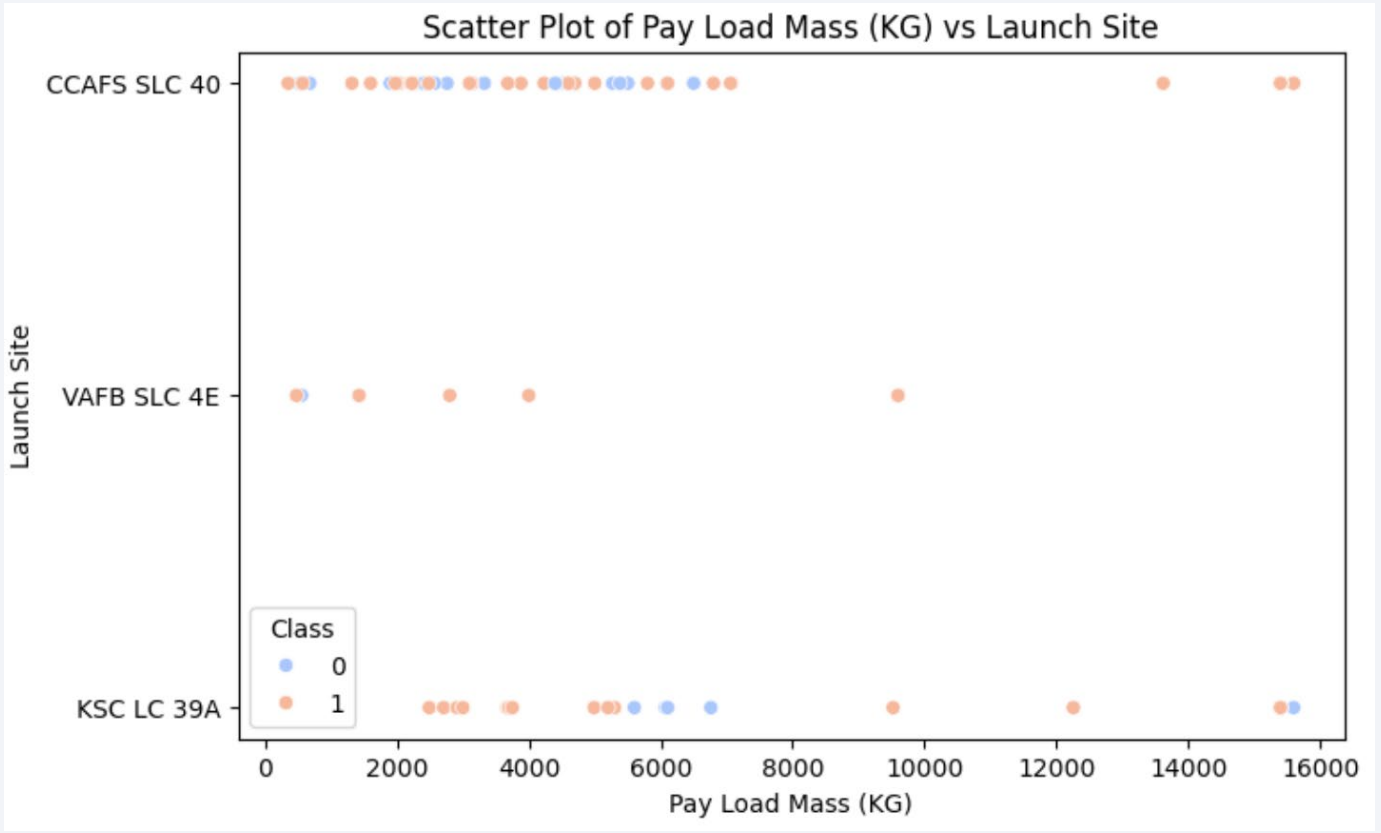


- The x-axis represents the flight number, and the y-axis represents the launch site.
- Each point in the scatterplot represents a single flight.
- The colour of the points represents the class of the flight, with blue for class 0 and orange for class 1.

DTSA-5841-IBM-Applied-DS-Capstone/module2/edadataviz-answer.ipynb at main · korkridake/DTSA-5841-IBM-Applied-DS-Capstone

Payload vs. Launch Site

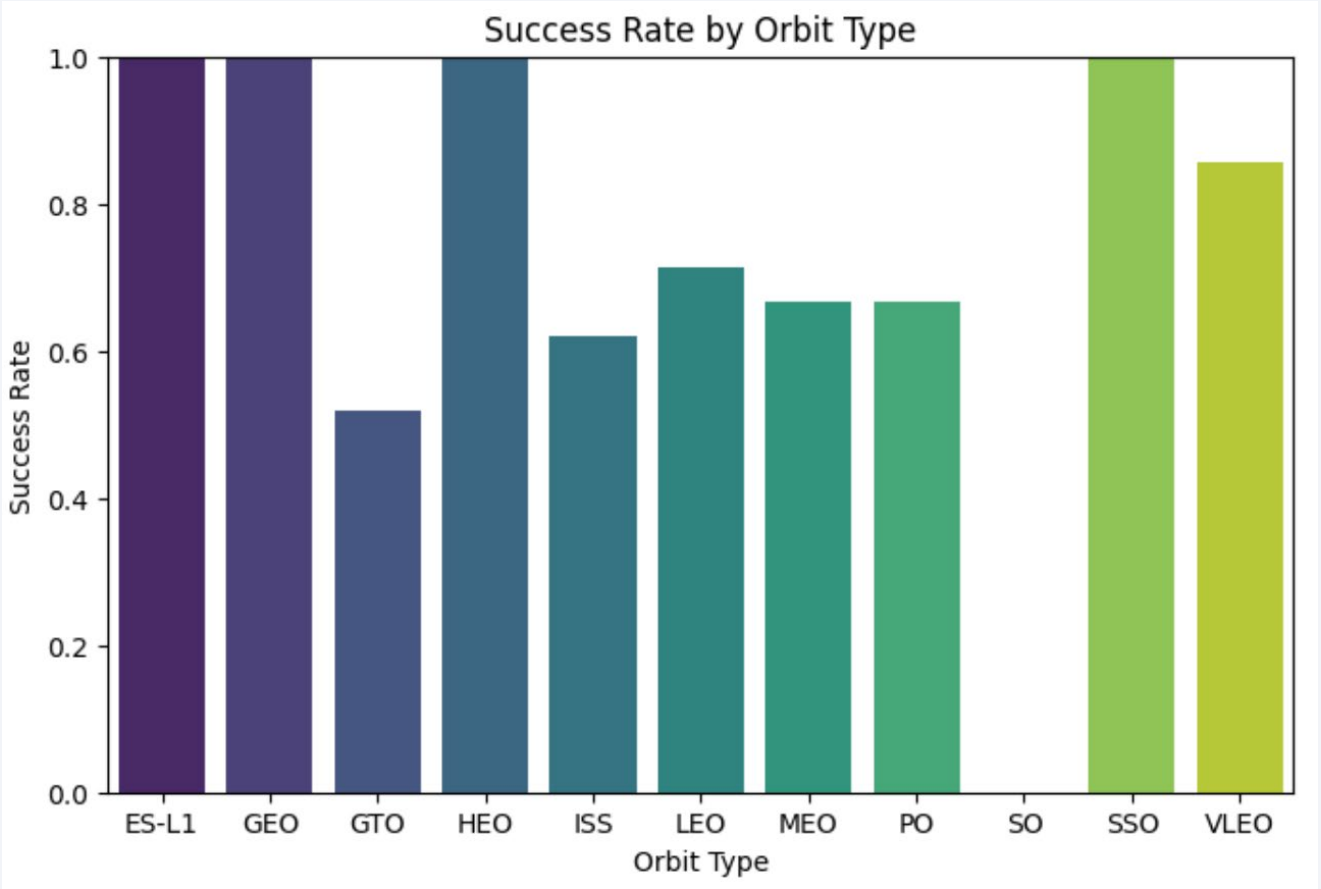
- We also want to observe if there is any relationship between launch sites and their payload mass.
- For the VAFB-SLC launchsite there are no rockets launched for heavypayload mass (greater than 10000).



DTSA-5841-IBM-Applied-DS-Capstone/module2/edadataviz-answer.ipynb at main · korkridake/DTSA-5841-IBM-Applied-DS-Capstone

Success Rate vs. Orbit Type

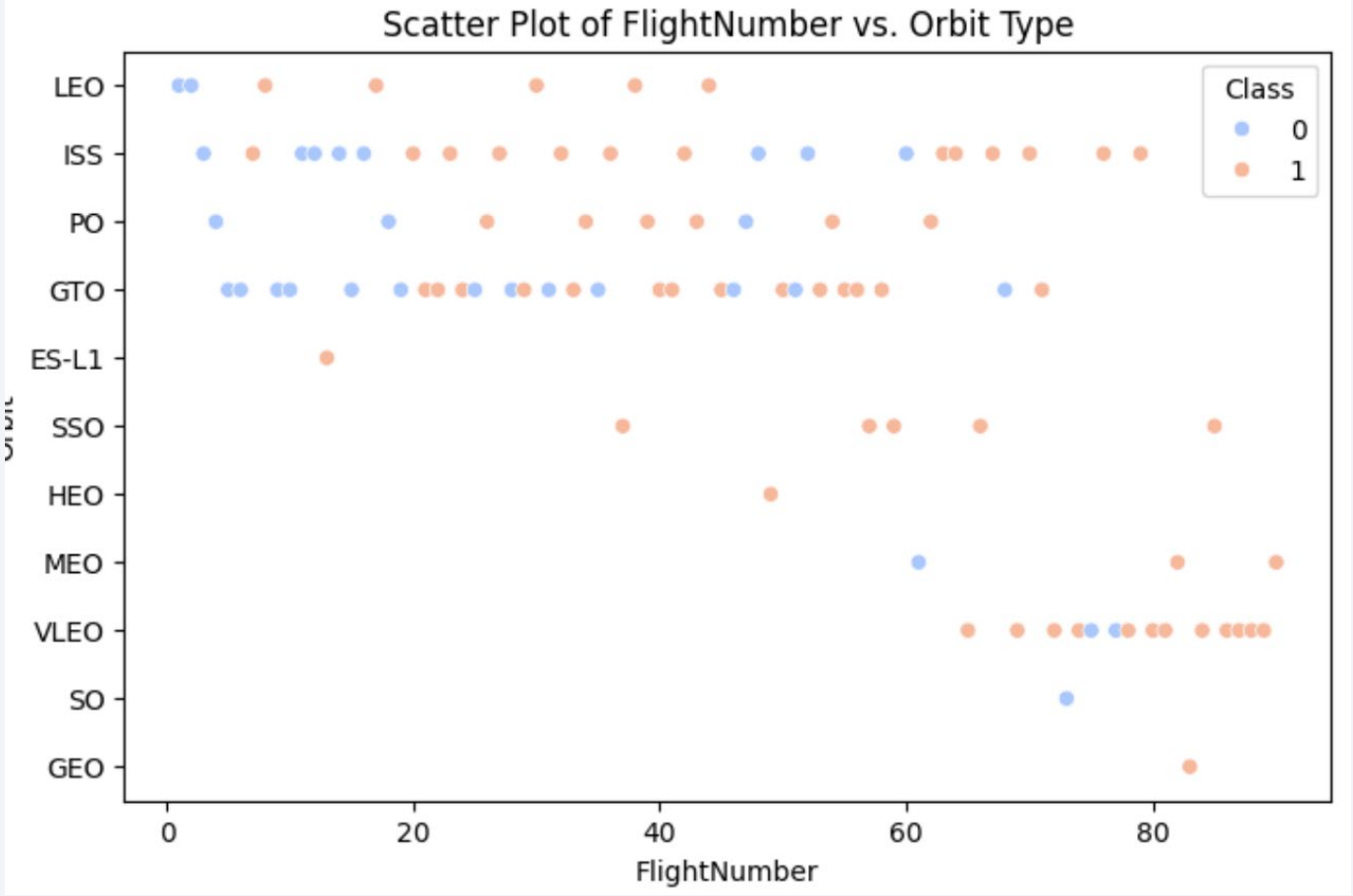
- We want to visually check if there are any relationship between success rate and orbit type.
- The chart highlights that certain orbit types might be inherently more complex or prone to failure during launch or initial orbital insertion phases (e.g., GTO).
- The high success rates for orbits like GEO and SSO could be attributed to well-established launch procedures and potentially more stable orbital mechanics once achieved.



DTSA-5841-IBM-Applied-DS-Capstone/module2/edadataviz-answer.ipynb at main · korkridake/DTSA-5841-IBM-Applied-DS-Capstone

Flight Number vs. Orbit Type

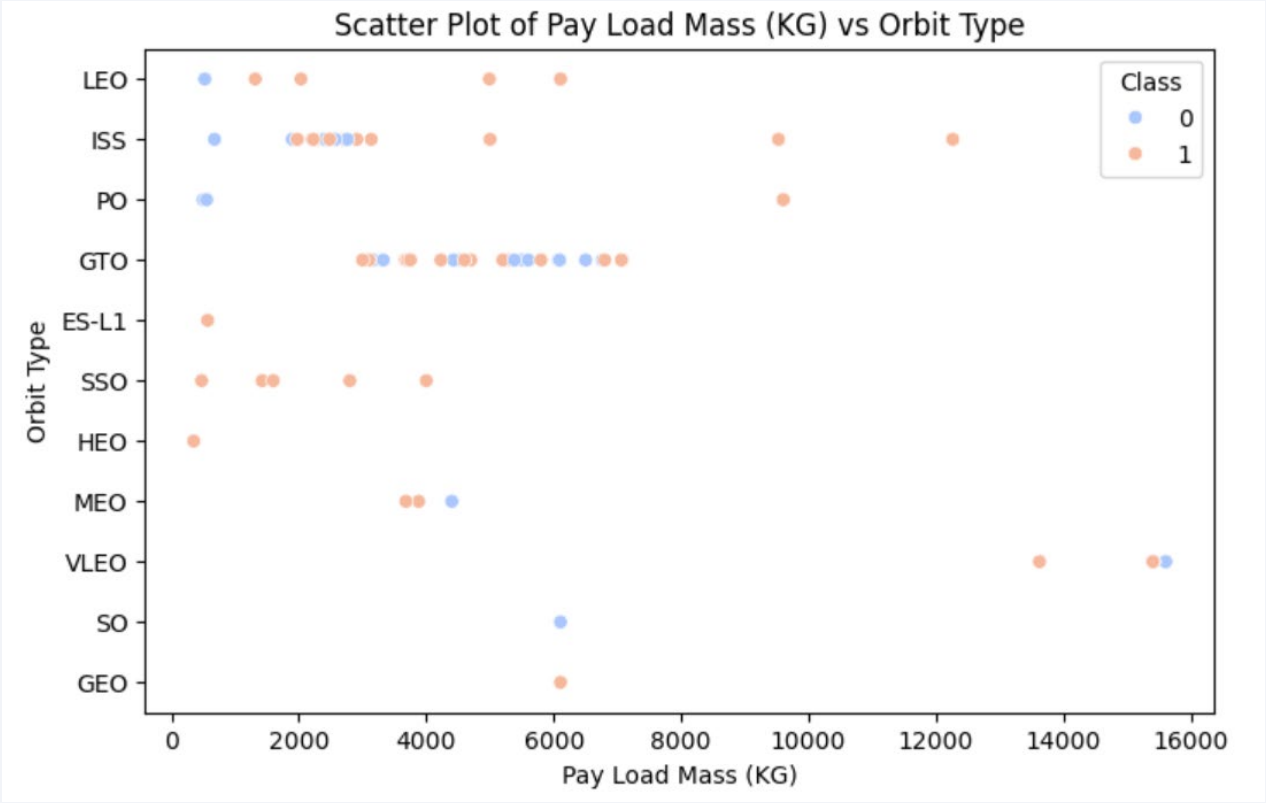
- For each orbit, we want to see if there is any relationship between FlightNumber and Orbit type.
- I observe that in the LEO orbit, success seems to be related to the number of flights. Conversely, in the GTO orbit, there appears to be no relationship between flight number and success.



DTSA-5841-IBM-Applied-DS-Capstone/module2/edadataviz-answer.ipynb at main · korkridake/DTSA-5841-IBM-Applied-DS-Capstone

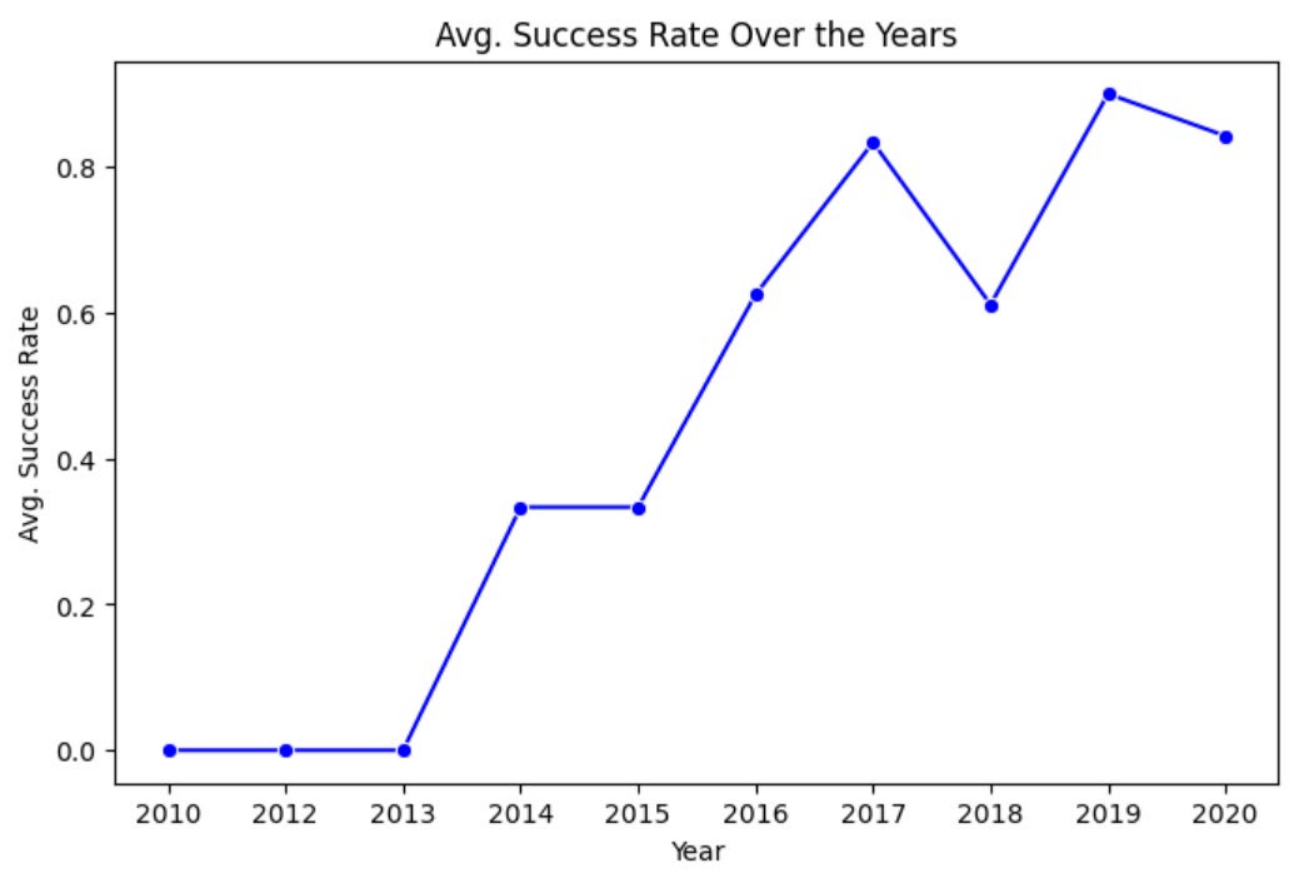
Payload vs. Orbit Type

- Similarly, we can plot the Payload Mass vs. Orbit scatter point charts to reveal the relationship between Payload Mass and Orbit type.
- With heavy payloads the successful landing or positive landing rate are more for Polar,LEO and ISS.
- However, for GTO, it's difficult to distinguish between successful and unsuccessful landings as both outcomes are present.



DTSA-5841-IBM-Applied-DS-Capstone/module2/edadataviz-answer.ipynb at main · korkridake/DTSA-5841-IBM-Applied-DS-Capstone

Launch Success Yearly Trend



- I can plot a line chart of yearly average success rate. I observe that the sucess rate since 2013 kept increasing till 2020

DTSA-5841-IBM-Applied-DS-Capstone/module2/edadataviz-answer.ipynb at main · korkridake/DTSA-5841-IBM-Applied-DS-Capstone

All Launch Site Names

- I display the names of the unique launch sites in the space mission

%sql SELECT DISTINCT LAUNCH_SITE as "Launch_Sites" FROM SPACEXTABLE;

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

- **CCAFS LC-40:** This refers to Cape Canaveral Space Force Station Launch Complex 40 located at Cape Canaveral Space Force Station in Florida. It was initially used for Titan rockets and later leased by SpaceX. It is now a high-volume launch site for SpaceX's Falcon 9 rockets

- **VAFB SLC-4E:** This refers to Vandenberg Space Force Base Space Launch Complex 4 East located at Vandenberg Space Force Base in California.

- **KSC LC-39A:** This refers to Kennedy Space Center Launch Complex 39A located at NASA's Kennedy Space Center in Florida.

DTSA-5841-IBM-Applied-DS-Capstone/module2/edadataviz-answer.ipynb at main · korkridake/DTSA-5841-IBM-Applied-DS-Capstone

Launch Site Names Begin with 'CCA'

%sql SELECT * FROM SPACEXTABLE WHERE LAUNCH_SITE LIKE "CCA%" LIMIT 5;

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
2010-06-04	18:45:00	F9 v1.0 B0003	**CCAFS LC-40**	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
2010-12-08	15:43:00	F9 v1.0 B0004	**CCAFS LC-40**	Dragon demo flight C1, two CubeSats, barrel of Brouere cheese	0	LEO (ISS)	NASA (COTS) NRO	Success	Failure (parachute)
2012-05-22	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

Total Payload Mass

- %sql SELECT SUM(PAYLOAD_MASS__KG_) AS "The Total Payload Mass Carried by Boosters Launched by NASA (CRS)" FROM SPACEXTABLE WHERE Customer = "NASA (CRS)";
- The total payload mass carried by boosters launched by NASA (CRS) is 45,596.

Average Payload Mass by F9 v1.1

- %sql SELECT AVG(PAYLOAD_MASS__KG_) FROM SPACEXTABLE WHERE Booster_Version LIKE "F9 v1.1“
- We calculate the average payload mass carried by booster version F9 v1.1 to be 2,928.4.

First Successful Ground Landing Date

- %sql SELECT DISTINCT Landing_Outcome FROM SPACEXTABLE
- %sql SELECT MIN(DATE) FROM SPACEXTBL WHERE Landing_Outcome = "Success (ground pad)";

```
In [27]: %sql SELECT MIN(DATE) FROM SPACEXTBL WHERE Landing_Outcome = "Success (ground pad)";
 * sqlite:///my_data1.db
Done.
Out[27]: MIN(DATE)
         2015-12-22
```

Successful Drone Ship Landing with Payload between 4000 and 6000

- List the names of boosters which have successfully landed on drone ship and had payload mass greater than 4000 but less than 6000
- %sql SELECT Booster_Version FROM SPACEXTBL WHERE Landing_Outcome = "Success (drone ship)" AND PAYLOAD_MASS__KG_ BETWEEN 4000 AND 6000;

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

Total Number of Successful and Failure Mission Outcomes

- %sql SELECT DISTINCT Mission_Outcome FROM SPACEXTABLE
- %sql SELECT COUNT(Mission_Outcome) FROM SPACEXTBL WHERE Mission_Outcome LIKE "%Success%";
- %sql SELECT COUNT(Mission_Outcome) FROM SPACEXTBL WHERE Mission_Outcome LIKE "%Failure%";

```
%sql SELECT COUNT(Mission_Outcome) FROM SPACEXTBL WHERE Mission_Outcome LIKE "%Success%";
```

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

COUNT(Mission_Outcome)
100

```
%sql SELECT COUNT(Mission_Outcome) FROM SPACEXTBL WHERE Mission_Outcome LIKE "%Failure%";
```

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

COUNT(Mission_Outcome)
1

Boosters Carried Maximum Payload

- %sql SELECT DISTINCT Booster_Version FROM SPACEXTBL WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTBL) ORDER BY Booster_Version ASC;

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

2015 Launch Records

- List the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015
- %sql SELECT DATE, substr(Date, 6,2) AS MONTHNAME, Booster_Version, Launch_Site FROM SPACEXTBL WHERE substr(Date,0,5)='2015' AND Landing_Outcome = "Failure (drone ship)";

Date	MONTHNAME	Booster_Version	Launch_Site
2015-01-10	01	F9 v1.1 B1012	CCAFS LC-40
2015-04-14	04	F9 v1.1 B1015	CCAFS LC-40

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

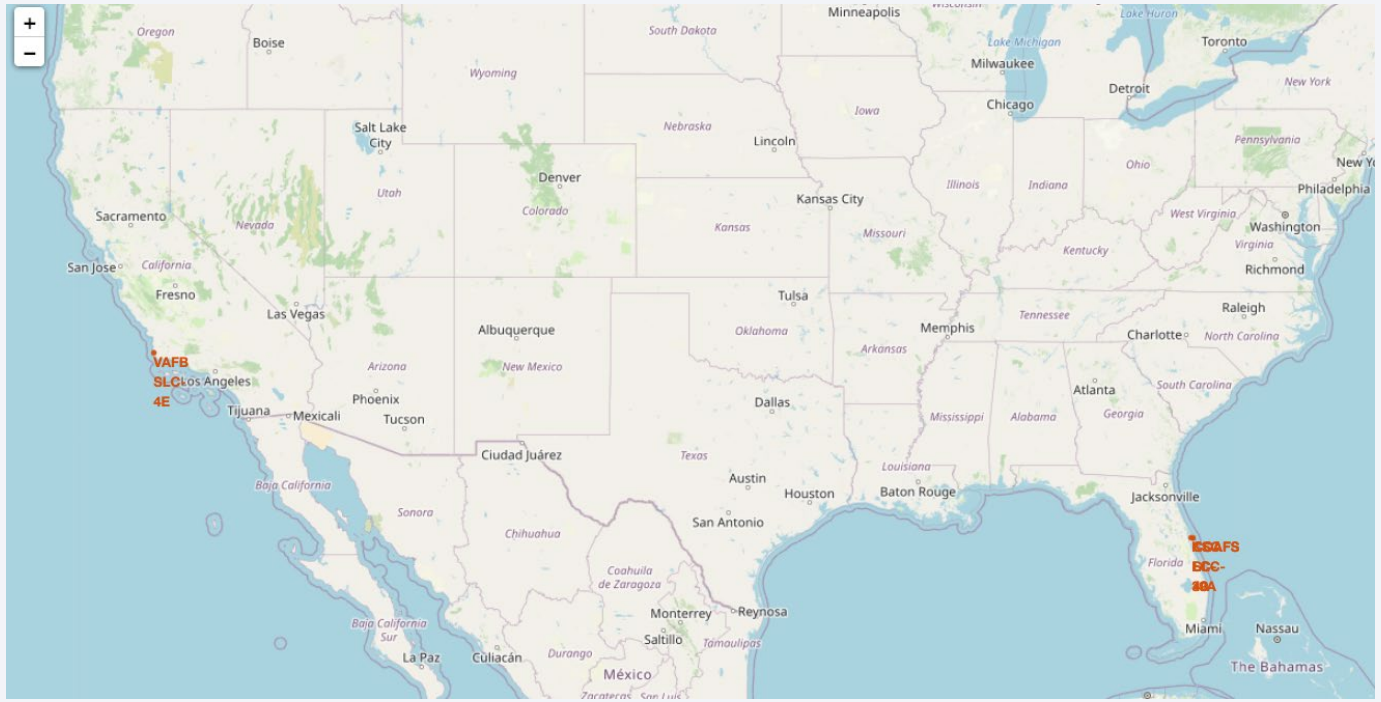
- Rank the count of landing outcomes (such as Failure (drone ship) or Success (ground pad)) between the date 2010-06-04 and 2017-03-20, in descending order.
- %sql SELECT Landing_Outcome, COUNT(Landing_Outcome) FROM SPACEXTBL WHERE Date BETWEEN "2010-06-04" and "2017-03-20" GROUP BY Landing_Outcome ORDER BY COUNT(Landing_Outcome) DESC;

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


Section 3
Launch Sites
Proximities Analysis

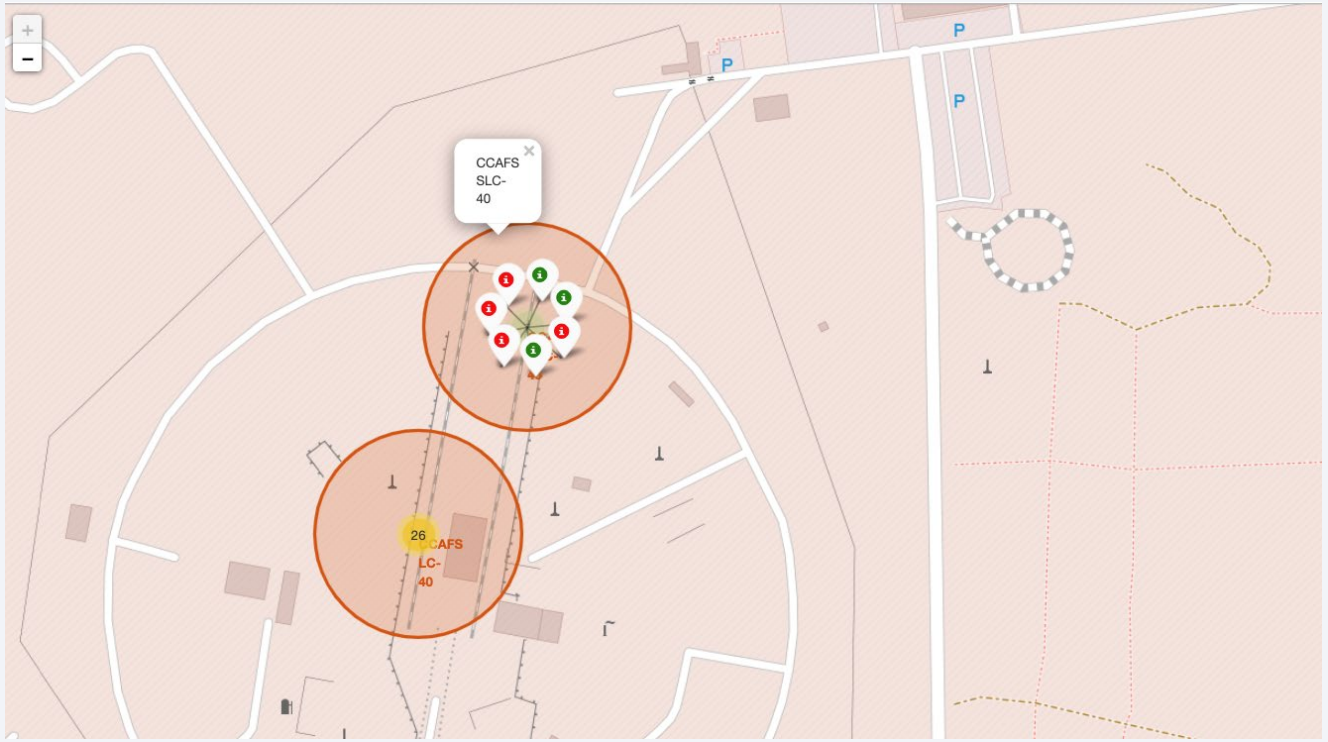
Folium I: All Launch Sites on Map

- Are all launch sites in proximity to the Equator line? No. The launch sites shown on this map (California and Florida) are located significantly far from the Equator. Proximity to the Equator is beneficial for eastward launches to gain extra velocity from Earth's rotation and for reaching geostationary orbit efficiently. However, it's not a strict requirement for all launch sites, especially those intended for polar or other types of orbits.
- Are all launch sites in very close proximity to the coast? Yes, based on the labels visible in this image. Both the California and Florida sites appear to be located very close to the coastline. Coastal locations are often preferred for launch sites for safety reasons, as they allow for over-water trajectories in case of launch failures, minimizing risks to populated areas.



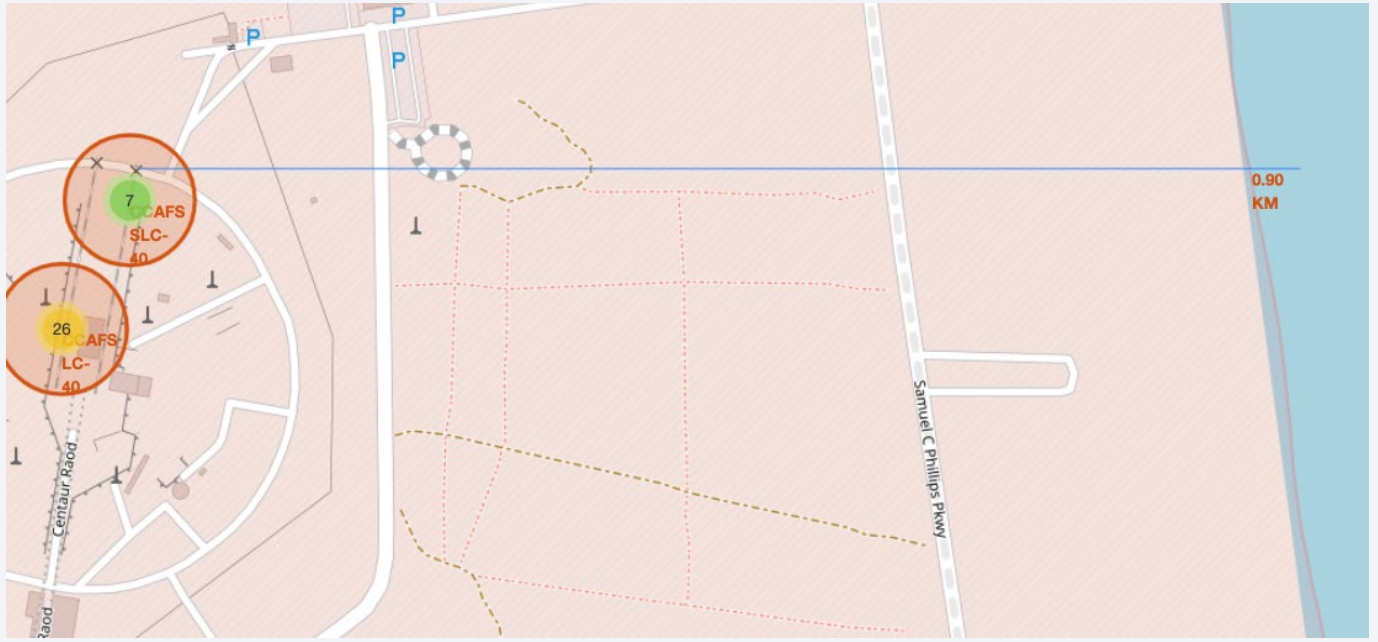
Folium II: The Success/Failed Launches for Each Site on the Map

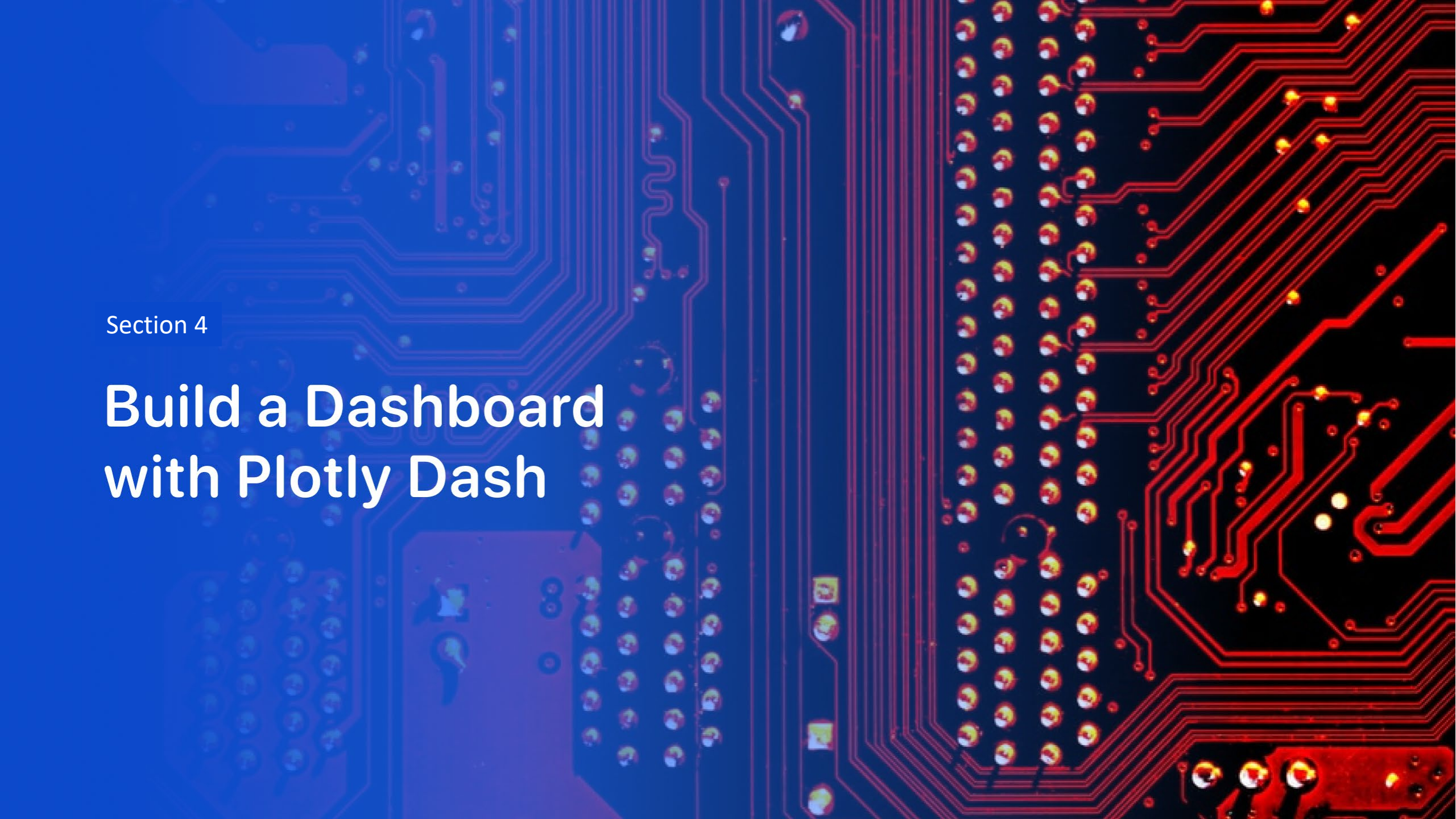
- If we take a closer look at one of the launch sites, we will notice green and red tags. Each green tag indicates a successful launch, while each red tag indicates a failed launch.
- The top cluster (labeled "CCAFS SLC-40"): This cluster contains a mix of green and red tags. There are more green tags than red tags. This indicates that this launch site has experienced both successful and failed launches, but the number of successful launches appears higher than the number of failed launches. Therefore, it has a relatively good success rate.
- The bottom cluster (labeled "AFS LC-40"): This cluster also contains both green and red tags. By visual inspection, it appears to have a similar or possibly slightly lower proportion of green tags compared to the top cluster. This indicates that this launch site also has a history of both successful and failed launches.



Folium III: Distances Between a launch site to its Proximities

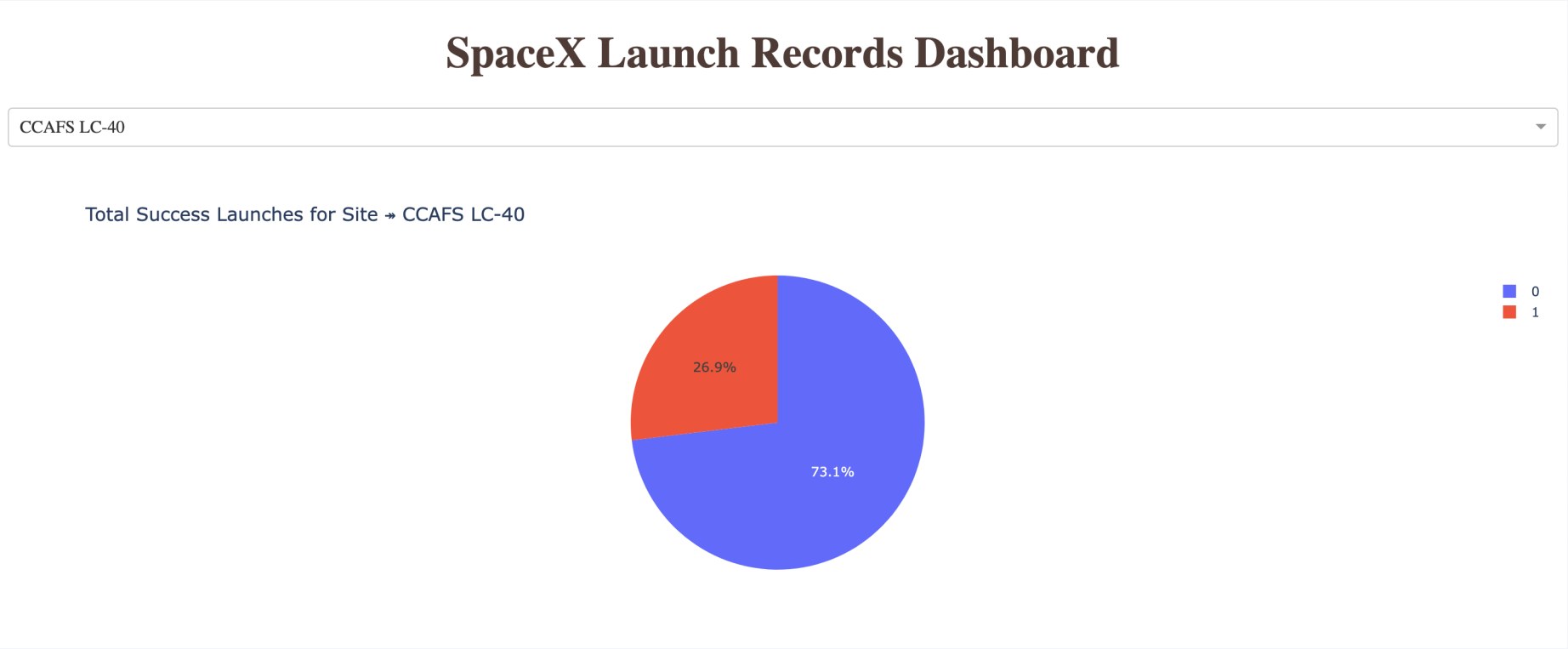
- Proximity to Highways: Yes
- Proximity to Coastline: Yes




Section 4
Build a Dashboard with Plotly Dash

Plotly Dash I: SpaceX Launch Records Dashboard

- The chart indicates that for the CCAFS LC-40 launch site:26.9% of the recorded launches were successful (Orange segment, labeled "1").73.1% of the recorded launches were failures (Blue segment, labeled "0").

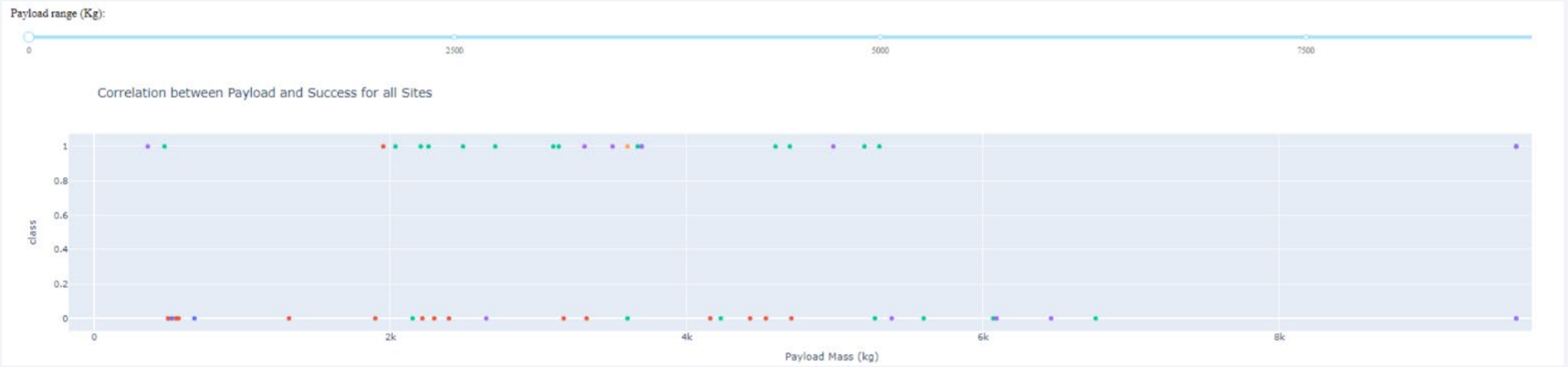


Plotly Dash II: Correlation Between Payload and Success for Site

- The plot suggests that for CCAFS LC-40, the booster version is a significant factor influencing success. The older v1.1 booster appears associated only with failures in this dataset view, while the FT booster shows mixed results but is responsible for all the successes depicted here. The relationship between payload mass and success isn't straightforward and seems intertwined with the booster technology used.

Plotly Dash III: Correlation Between Payload and Success for All Sites

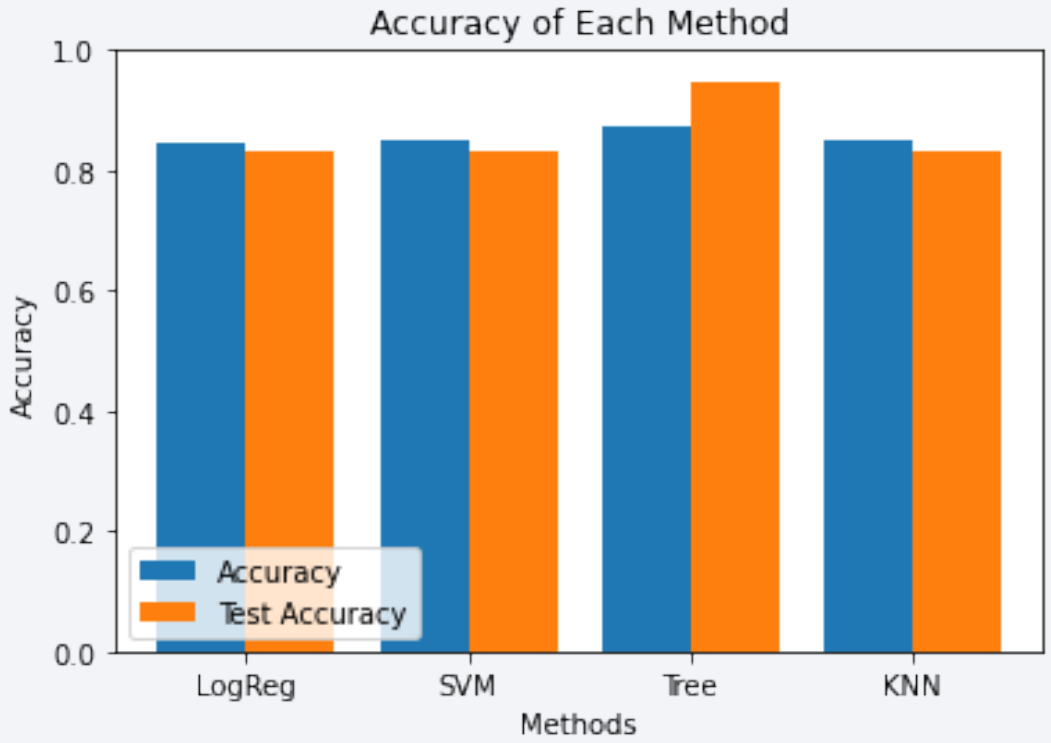
- The plot provides a high-level overview of launch success versus payload mass across all SpaceX sites. It suggests that while failures have occurred across many payload ranges, launches with the heaviest payloads have generally been successful overall. However, the lack of a legend prevents a deeper analysis of how different factors (like launch site or booster type, potentially represented by the colors) contribute to these outcomes.



Section 5

Predictive Analysis (Classification)

Classification Accuracy



- Tree method shows the highest accuracy: The "Tree" method (likely referring to a Decision Tree or Random Forest) has the highest accuracy on both the training data (represented by the blue bar) and the test data (represented by the orange bar) compared to the other methods (LogReg, SVM, and KNN).
- KNN has the lowest accuracy: The K-Nearest Neighbors (KNN) method exhibits the lowest accuracy on both the training and test datasets among the four methods compared.
- Overfitting in the Tree method: While the Tree method has the highest overall accuracy, there is a noticeable difference between its training accuracy (blue bar) and test accuracy (orange bar). The training accuracy is significantly higher than the test accuracy, suggesting potential overfitting. This means the model might be learning the training data too well, including noise, and thus performs relatively worse on unseen data.

Confusion Matrix

- Our best performing model is Decision Tree.
- The confusion matrix shows it effectively identifies "landed" cases with a perfect recall (100%) and high overall accuracy (94.4%). Its minor weakness is occasionally predicting a landing when one didn't occur, resulting in a precision of 92.3% for "landed" predictions and a specificity of 83.3% for the "did not land" class.

Conclusions

- Each characteristic of a Falcon 9 launch, including its payload mass and orbit type, can influence the mission outcome in various ways. For instance, the payload mass might determine the amount of fuel required, while the orbit type could affect the duration and trajectory of the mission. Understanding these factors is crucial for predicting and optimizing the success of each launch.
- With heavy payloads, the successful landing rate or positive landing rate is higher for orbit types such as Polar, LEO, and ISS.
- The Tree-based method achieves the best performance but shows signs of overfitting. LogReg and SVM offer more balanced performance with better generalization, while KNN has the weakest performance among the methods compared.

Thank you!