IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Claudio Reibaud
08/05/2024

Outline

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

Executive Summary

- **Summary of Methodologies:**
  - The project utilized a combination of web scraping, data cleaning, exploratory data analysis, interactive visual analytics, predictive analysis, and SQL querying techniques to achieve its objectives.
  - Each methodology contributed to a deeper understanding of SpaceX Falcon 9 launch data and provided valuable insights for predicting first-stage landing success.
- **Summary of Results:**
  - Overall, the project successfully explored and analyzed SpaceX Falcon 9 launch data, uncovered key insights, and developed predictive models for first-stage landing success.
  - The results contribute to a better understanding of factors influencing launch success rates and have practical implications for cost estimation and decision-making in the space industry.

Introduction

- **Project Background and Context:**
    - SpaceX, founded by Elon Musk, has revolutionized the space industry with its Falcon 9 rocket, which is known for its ability to reuse the first stage, significantly reducing the cost of launches.
    - The success of Falcon 9 missions depends on various factors, including the successful landing of the first stage.
    - Understanding the determinants of successful first-stage landings is crucial for estimating the cost of launches accurately and making informed decisions in the space industry.
- **Problems to Find Answers:**
    - **Predicting First-Stage Landing Success:** The main problem we aimed to address was predicting whether the first stage of a Falcon 9 rocket would land successfully or not.
    - **Cost Estimation for Launches:** By accurately predicting first-stage landing success, we could determine the cost of launches more precisely. This information is vital for SpaceX and other companies bidding for rocket launch contracts.
    - **Exploring Launch Data Patterns:** Additionally, we wanted to explore patterns and insights in SpaceX Falcon 9 launch data to understand factors influencing launch success rates.
    - **Geographical Analysis of Launch Sites:** Another problem we wanted to tackle was analyzing the geographical distribution of launch sites and their proximity to successful landing locations.
    - **Comparing Predictive Models:** We also aimed to compare different predictive models, including Support Vector Machines (SVM), Classification Trees, and Logistic Regression, to determine which one performs best for predicting first-stage landing success.

Section 1

Methodology

Methodology

Executive Summary

- **Data Collection Methodology:**
  - **Web Scraping:** We collected Falcon 9 launch records from a Wikipedia page titled "List of Falcon 9 and Falcon Heavy launches" using web scraping techniques. The BeautifulSoup library in Python was used to extract the HTML table containing the launch records.
- **Data Wrangling and Processing:**
  - **Cleaning and Formatting:** The scraped data underwent cleaning and formatting to ensure consistency and integrity. This involved handling missing values, converting data types, and standardizing formats.
- **Exploratory Data Analysis (EDA):**
  - **Visualization:** EDA was performed using Matplotlib and Seaborn libraries to visualize the data. We created various charts, such as histograms, scatter plots, and box plots, to explore distributions, relationships, and patterns in the data.
  - **SQL Queries:** In addition to visualization, SQL queries were executed to further analyze the SpaceX dataset. This involved extracting relevant information from the dataset using SQL statements to gain deeper insights.
- **Interactive Visual Analytics:**
  - **Folium:** Interactive visual analytics were conducted using Folium, a Python library for creating interactive maps. We marked all launch sites on a map and analyzed launch success/failure rates for each site.
  - **Plotly Dash:** Further interactive visual analytics were performed using Plotly Dash, a Python framework for building interactive web-based dashboards. This allowed us to create dynamic visualizations and explore the data interactively.

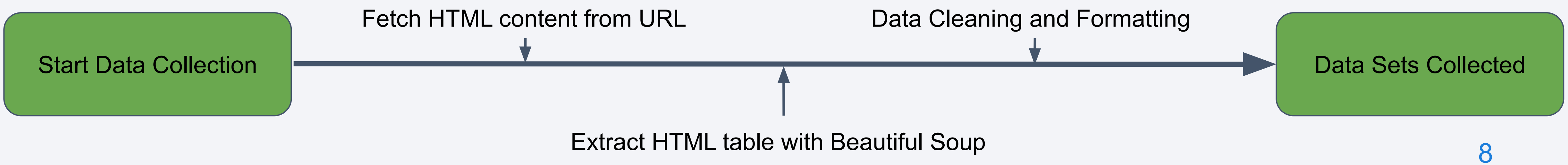
Methodology

Executive Summary

- **Predictive Analysis:**
  - **Classification Models:** We built predictive models using classification algorithms such as Support Vector Machines (SVM), Classification Trees, and Logistic Regression.
  - **Model Building:** The data was split into training and test sets, and the classification models were trained on the training data.
  - **Model Tuning:** Hyperparameter tuning techniques, such as grid search or randomized search, were used to find the optimal hyperparameters for each model.
  - **Model Evaluation:** The trained models were evaluated using metrics such as accuracy, precision, recall, and F1-score. Cross-validation techniques were employed to ensure robust evaluation.

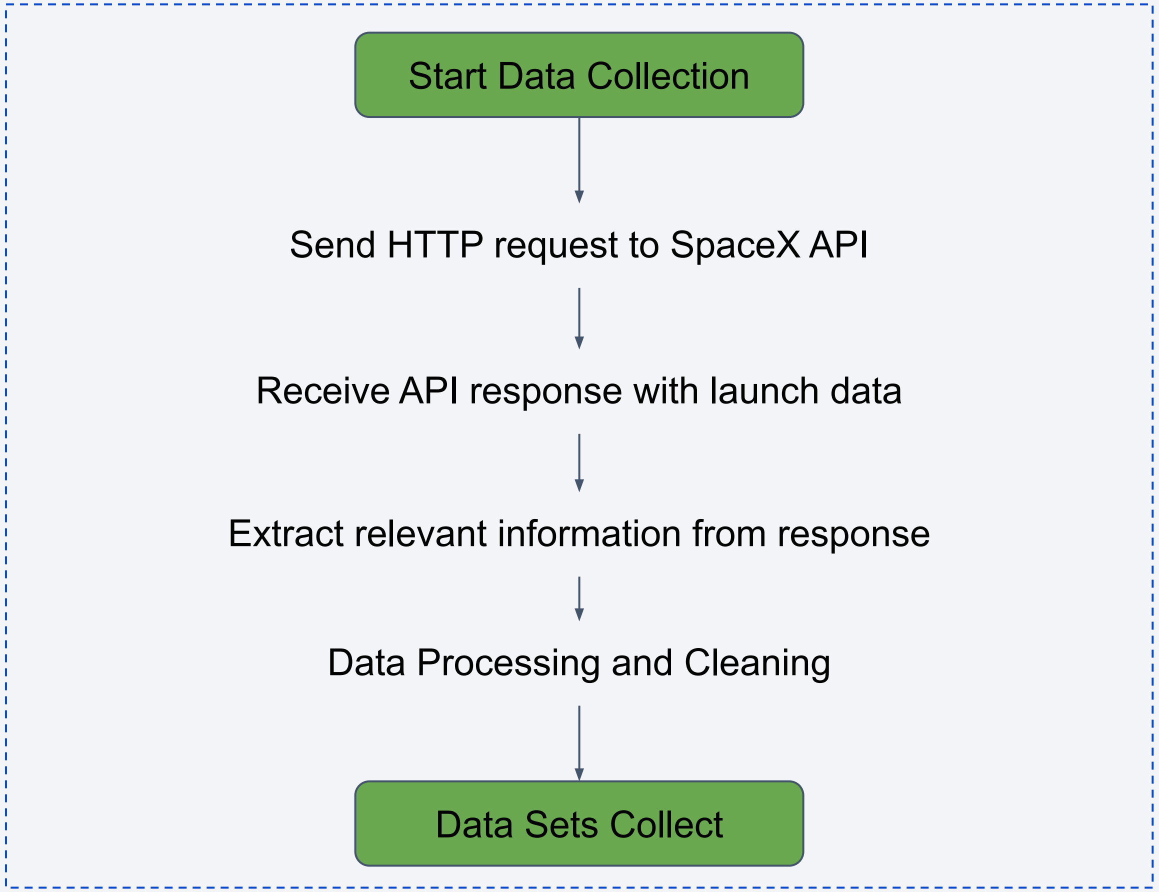
Data Collection

- **Data Collection Methodology:**
  - **Web Scraping with BeautifulSoup:**
    - Utilized BeautifulSoup library to extract Falcon 9 launch records HTML table from Wikipedia page.
    - Used Python's requests module to fetch the HTML content of the Wikipedia page.
    - Employed BeautifulSoup's find and find_all methods to locate and extract the desired HTML table.
- **Data Cleaning and Formatting:**
  - Identified and handled missing values, inconsistencies, and errors in the scraped data.
  - Converted data types to ensure consistency and integrity.
  - Standardized formats for uniformity and ease of analysis.



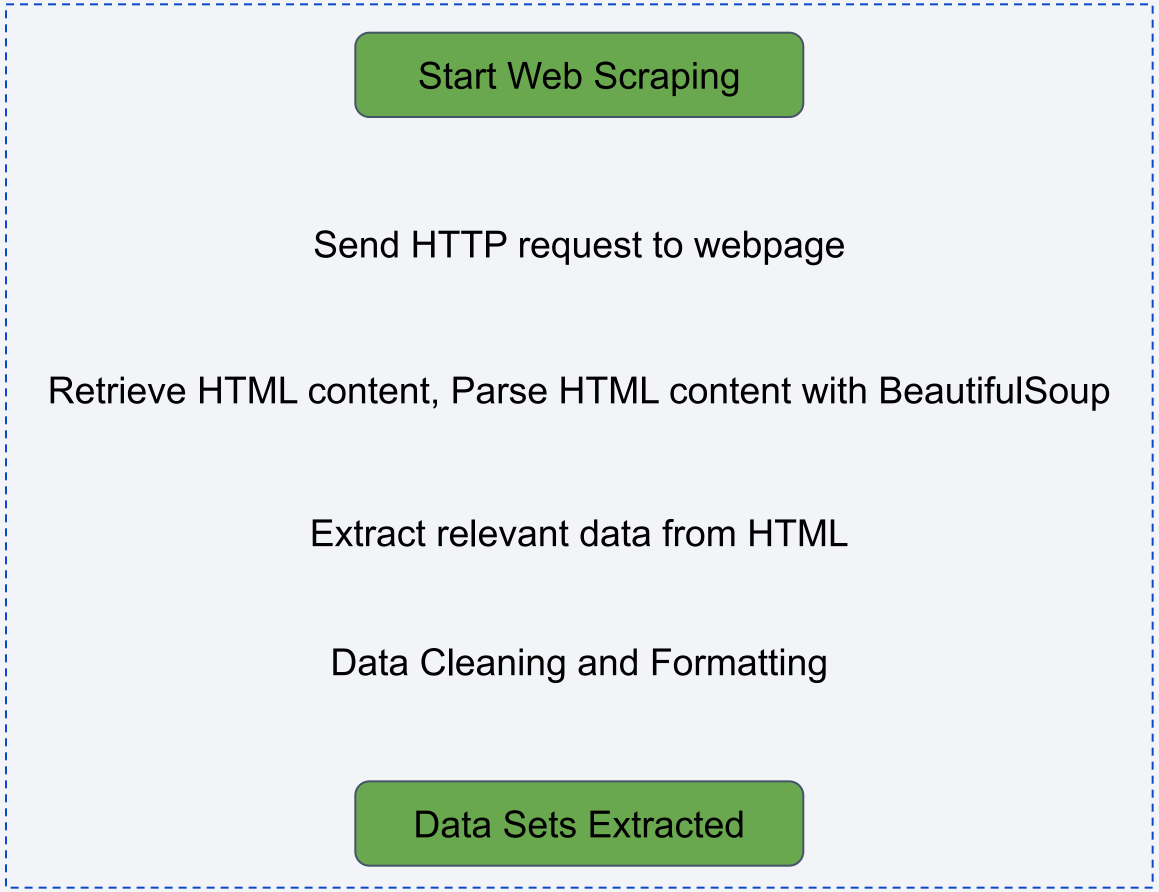
Data Collection – SpaceX API

- **SpaceX API Calls:**
  - Utilized SpaceX's REST API to fetch data related to Falcon 9 launches.
  - Sent HTTP requests to SpaceX API endpoints to retrieve information such as launch details, outcomes, and landing success/failure.
- **Data Processing:**
  - Extracted relevant information from the API responses, including launch dates, payloads, launch outcomes, and landing outcomes.
  - Ensured data consistency and integrity by handling missing values and standardizing formats.



Data Collection - Scraping

- **BeautifulSoup for HTML Parsing:**
  - Utilized the BeautifulSoup library in Python for parsing HTML content.
  - Extracted relevant data from the HTML structure of the target webpage.
  - Requests Module for HTTP Requests:
  - Used Python's requests module to send HTTP requests to the target webpage.
  - Retrieved the HTML content of the webpage for parsing.
- **Data Extraction and Cleaning:**
  - Located and extracted specific HTML elements containing the desired data, such as tables or text.
  - Cleaned and formatted the extracted data to ensure consistency and integrity



Data Wrangling

- **Data Wrangling Methodology:**
  - **Data Cleaning:**
    - Identified and handled missing values, inconsistencies, and errors in the dataset.
    - Removed duplicates and irrelevant columns to streamline the data.
  - **Data Transformation:**
    - Converted data types to ensure consistency and compatibility.
    - Standardized formats for uniformity and ease of analysis.
  - **Feature Engineering:**
    - Derived new features from existing ones to enhance the predictive power of the dataset.
    - Applied domain knowledge to create meaningful features for analysis.

EDA with Data Visualization

- **Charts Plotted in Exploratory Data Analysis (EDA):**
  - **Histograms:**
    - Used to visualize the distribution of numerical variables such as payload mass, flight number, and launch success.
    - Helpful for understanding the spread and central tendency of the data.
  - **Scatter Plots:**
    - Employed to explore relationships between numerical variables, such as payload mass vs. launch success.
    - Useful for identifying correlations and patterns in the data.
  - **Bar Plots:**
    - Utilized to compare categorical variables, such as launch sites and mission outcomes.
    - Effective for visualizing frequency distributions and making comparisons between categories.
  - **Line Plots:**
    - Used to visualize trends over time, such as the number of launches per year.
    - Helpful for identifying temporal patterns and trends in the data.
  - **Box Plots:**
    - Employed to visualize the distribution of numerical variables across different categories, such as launch outcomes by launch site.
    - Useful for detecting outliers and understanding variability within categories.

EDA with SQL

- **Summary of SQL Queries:**
  - Aggregated launch outcomes by launch site to calculate the number of successful and failed launches at each site.
  - Calculated the success rate for each launch site by dividing the number of successful launches by the total number of launches.
  - Analyzed the distribution of launch outcomes by mission outcome type to understand the proportion of successful and failed landings.
  - Examined the relationship between payload mass and launch success by calculating the average payload mass for successful and failed launches.
  - Investigated the distribution of launch outcomes by booster landing outcome to identify trends in landing success.

Build an Interactive Map with Folium

- **Map Objects Added to Folium Map:**
  - **Markers:**
    - Added markers to represent launch sites on the map.
    - Each marker indicates the location of a launch site, making it easier to visualize the geographical distribution of launch sites.
  - **Circles:**
    - Created circles to highlight the proximity of launch sites to successful landing locations.
    - The radius of each circle represents the distance between a launch site and its proximities, providing insights into the accessibility of successful landing locations.
  - **Lines:**
    - Drew lines to connect launch sites with successful landing locations.
    - Visualized the trajectory of Falcon 9 launches and landing locations, helping to identify spatial patterns and correlations.
- **Reasons for Adding Map Objects:**
  - **Markers:**
    - Added markers to provide a visual representation of launch sites, allowing users to easily identify their locations on the map.
  - **Circles:**
    - Circles were added to visually represent the proximity of launch sites to successful landing locations, enabling users to assess the spatial relationship between launch sites and landing zones.
  - **Lines:**
    - Lines were drawn to illustrate the trajectory of Falcon 9 launches and their connections to successful landing locations, aiding in the analysis of launch trajectories and spatial patterns.

Build a Dashboard with Plotly Dash

- **Plots/Graphs and Interactions Added to Plotly Dash Dashboard:**
  - **Bar Charts:**
    - Added bar charts to visualize the distribution of launch outcomes by different categories, such as launch sites, mission outcomes, and booster landing outcomes.
    - Enables users to compare the frequency of successful and failed launches across different categories and identify patterns or trends.
  - **Pie Charts:**
    - Included pie charts to show the proportion of successful and failed launches relative to total launches.
    - Provides a clear visualization of launch success rates and allows users to quickly assess overall mission outcomes.
  - **Interactive Dropdowns:**
    - Implemented interactive dropdown menus to allow users to select specific categories or filters for the plotted data.
    - Enhances user interactivity and enables dynamic exploration of launch data based on user preferences.
  - **Hover Interactions:**
    - Enabled hover interactions on the plots to display additional information, such as launch dates, mission names, and success/failure counts.
    - Provides users with detailed insights into individual data points and enhances the interpretability of the plots.
- **Reasons for Adding Plots and Interactions:**
  - **Bar Charts and Pie Charts:**
    - Added to provide visual representations of launch outcomes and success rates across different categories, helping users understand the distribution and trends in launch data.
  - **Interactive Dropdowns:**
    - Implemented to facilitate user-driven exploration of the data by allowing them to select specific categories or filters for visualization, enhancing user engagement and flexibility.
  - **Hover Interactions:**
    - Enabled to provide users with detailed information about individual data points, such as launch dates and mission names, improving the interpretability and usability of the plots.

Predictive Analysis (Classification)

- **Model Development Process:**
  - **Data Preprocessing:**
    - Prepared the dataset by encoding categorical variables, scaling numerical features, and splitting the data into training and testing sets.
  - **Model Building:**
    - Implemented classification models including Support Vector Machines (SVM), Classification Trees, and Logistic Regression.
    - Trained each model on the training data using default hyperparameters.
  - **Model Evaluation:**
    - Evaluated the performance of each model using metrics such as accuracy, precision, recall, and F1-score on the test data.
    - Employed cross-validation techniques to ensure robust evaluation and minimize overfitting.
  - **Model Improvement:**
    - Conducted hyperparameter tuning for each model using techniques such as grid search or randomized search.
    - Fine-tuned hyperparameters to optimize model performance and generalization ability.
  - **Finding the Best Performing Model:**
    - Compared the performance of different models based on evaluation metrics and selected the best performing model.
    - Identified the model with the highest accuracy, precision, recall, or F1-score as the best performing model.

Results

- **Exploratory Data Analysis (EDA) Results:**
  - Explored the distribution of key variables such as payload mass, flight number, and launch success/failure.
  - Investigated relationships between variables using visualizations like histograms, scatter plots, and box plots.
  - Analyzed the distribution of launch outcomes by different categories such as launch sites, mission outcomes, and booster landing outcomes.
  - Identified trends and patterns in the data to gain insights into SpaceX Falcon 9 launch data.
- **Interactive Analytics Demo in Screenshots:**
  - Created an interactive map using Folium to visualize the geographical distribution of launch sites and successful landing locations.
  - Implemented interactive features such as markers, circles, and lines to enhance user engagement and exploration of launch data.
  - Captured screenshots demonstrating the interactivity and functionality of the interactive analytics demo.
- **Predictive Analysis Results:**
  - Built and evaluated classification models using algorithms like Support Vector Machines (SVM), Classification Trees, and Logistic Regression.
  - Tuned hyperparameters to optimize model performance and generalization ability.
  - Selected the best performing model based on evaluation metrics such as accuracy, precision, recall, and F1-score.

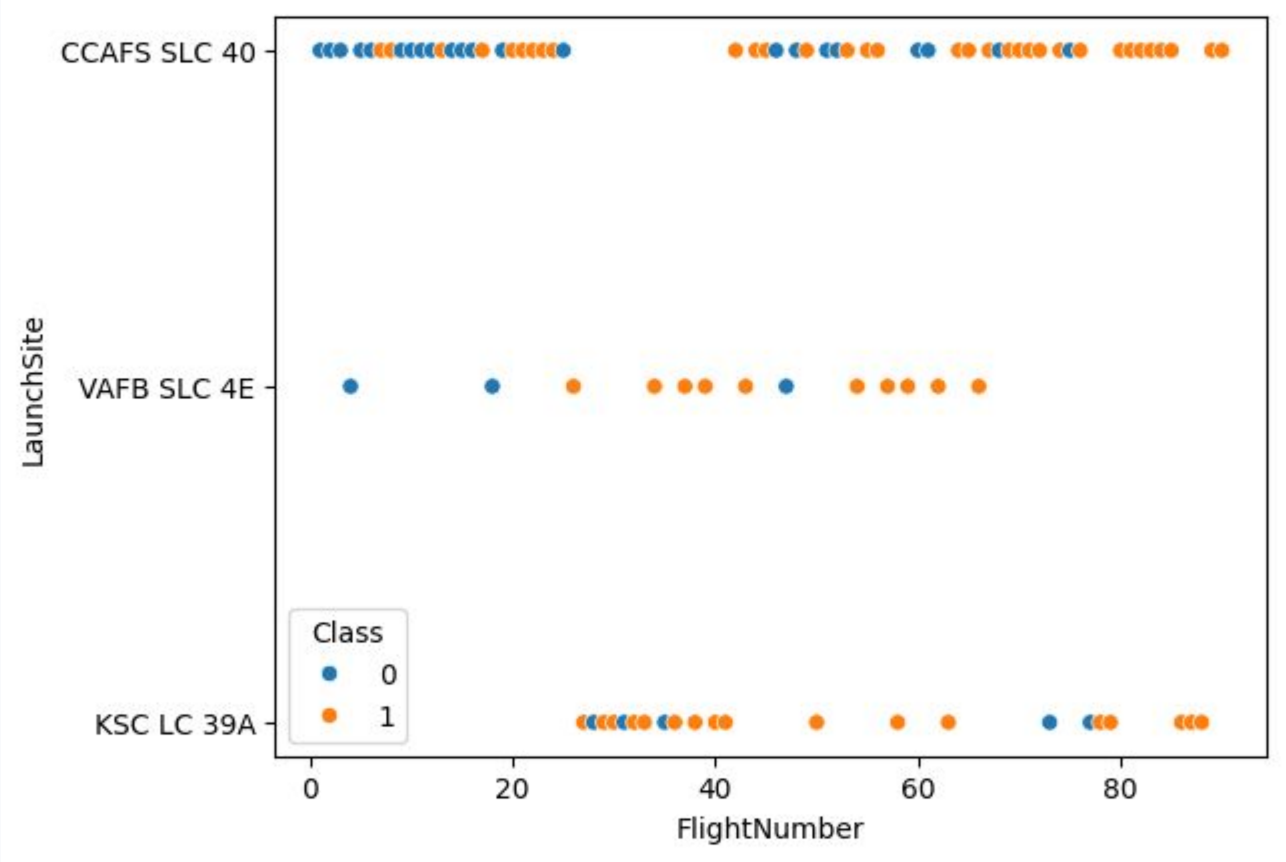
Section 2

Insights drawn from EDA

Flight Number vs. Launch Site

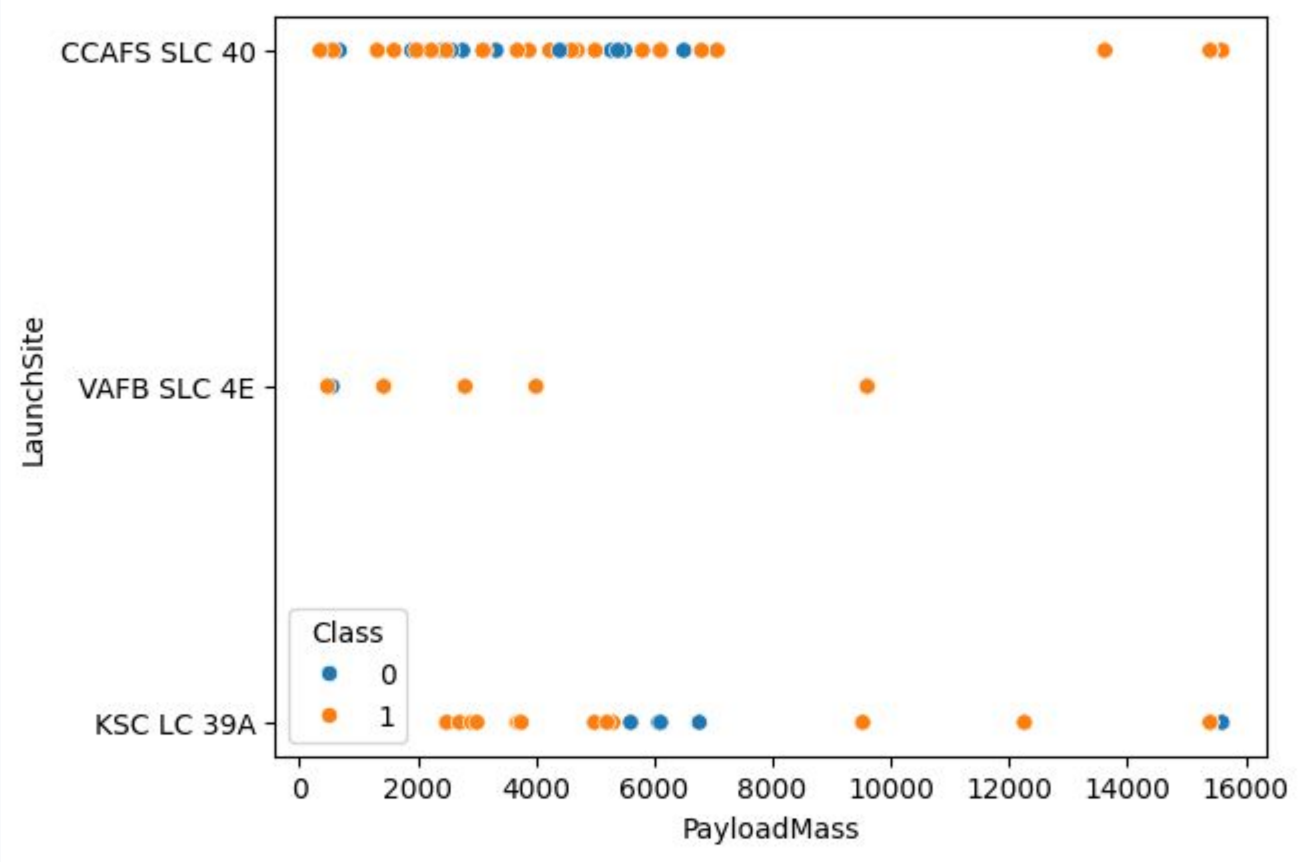
- **Y axis (vertical) - Launch sites :**
  - **CCAFS SLC 40 (top):** This launch site has a large amount of dots, indicating many flights. The points are mixed between classes 0 (blue) and 1 (orange).
  - **VAFB SLC 4E (middle):** This site also shows a substantial number of flights. Both classes are present but there seems to be a predominance of class 0.
  - K**SC LC 39A (bottom):** This site has fewer points compared to the other two, with a notable presence of class 1, especially in the higher flight numbers.
- **X axis (horizontal) - Flight number :**
  - The axis starts at 0 and goes beyond 100. Points are scattered along the axis, representing different flights.
- **Data points:**
  - Each point represents a specific flight at a specific launch site. The colour of the point (blue for class 0 and orange for class 1) indicates the class to which the flight belongs.
- **Legend:**
  - Located at the bottom left, it clarifies that the blue dots correspond to class 0, so a fail and the orange dots to class 1, so a success.

Conclusion: We can assume that CCAFS SLC 40 and KSC LC 39A are the two launch site that can be more successful because they have a lot of launch (more than 15 not like VAFB SLC 4E) and more than 50% of success. This success is more relevant than the 76.9% success at VAFB SLC 4E (10/13).



Payload vs. Launch Site

- **Y axis (vertical) - Launch sites :**
  - **CCAFS SLC 40 (top):** This launch site has a large amount of dots, indicating many payload mass. The points are mixed between classes 0 (blue) and 1 (orange).
  - **VAFB SLC 4E (middle):** This site has fewer points compared to the other two, with a notable presence of class 1, especially in the higher flight numbers.
  - KSC LC 39A **(bottom):** This site also shows a substantial number of flights. Both classes are present but there seems to be a predominance of class 1.
- **X axis (horizontal) - Payload Mass:**
  - The axis starts at 0 and goes beyond 16000. Points are scattered along the axis, representing different payload mass.
- **Data points:**
  - Each point represents a specific payload mass at a specific launch site. The colour of the point (blue for class 0 and orange for class 1) indicates the class to which the flight belongs.
- **Legend:**
  - Located at the bottom left, it clarifies that the blue dots correspond to class 0, so a fail and the orange dots to class 1, so a success.

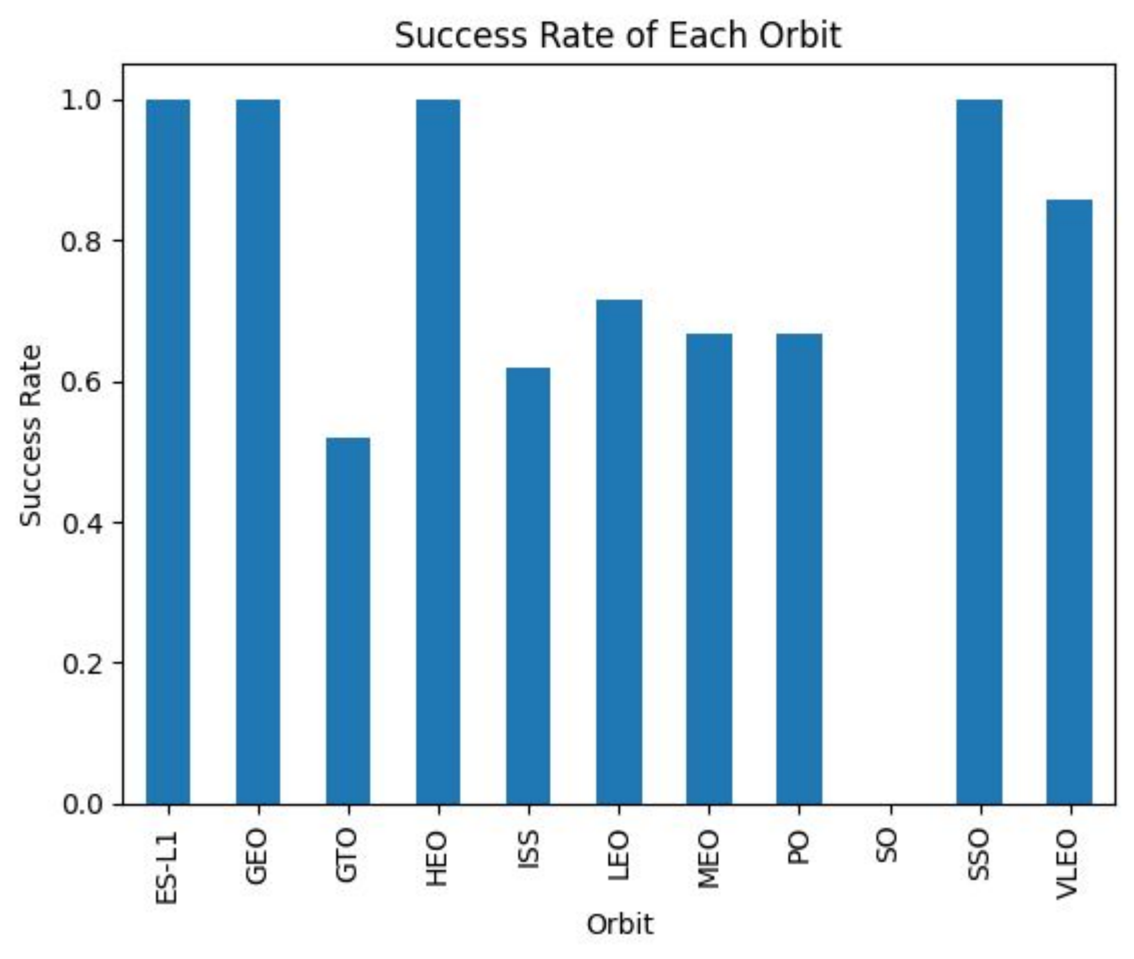


Conclusion: We can assume that there are more success for payload mass that are less than 8000.

Success Rate vs. Orbit Type

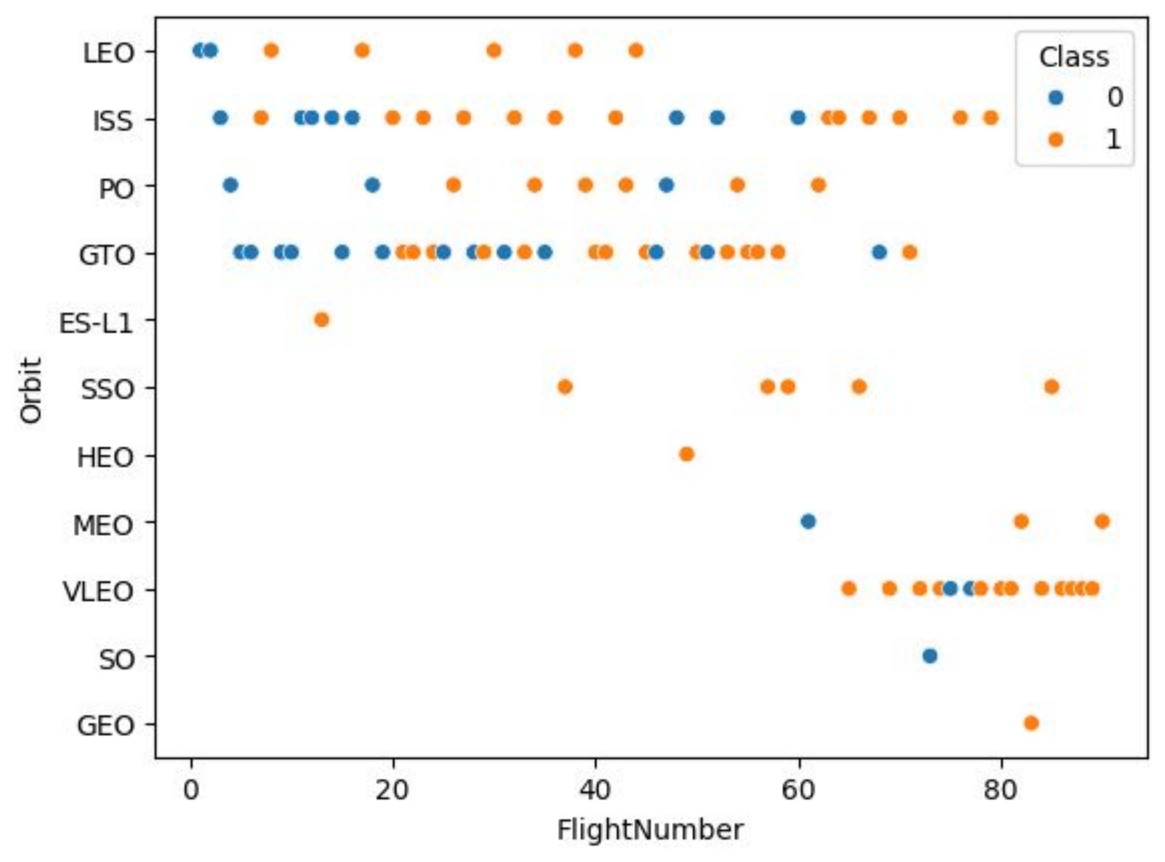
- **Y axis (vertical) - Success Rate :**
  - The axis starts at 0 and goes beyond 1.0. This represent 0% to 100%.
- **X axis (horizontal) - Orbit:**
  - **ES-L1:** 100% success.
  - **GEO:** 100% success.
  - **GTO:** 55% success.
  - **HEO:** 100% success.
  - **ISS:** 60% success.
  - **LEO:** 70% success.
  - **MEO:** 65% success.
  - **PO:** 65% success.
  - **SO:** 0% success.
  - **SSO:** 100% success.
  - **VLEO:** 85% success.
- **Data bar:**
  - Each bar represents the success rate of an orbit.

Conclusion: We can assume that ES-L1, GEO, HEO, SSO are the most successful orbit.



Flight Number vs. Orbit Type

- **Y axis (vertical) - Orbit :**
  - **ES-L1:** 100% success. That 1/1 success.
  - **GEO:** 100% success. That 1/1 success.
  - **GTO:** 51.85% success. That 14/27 success.
  - **HEO:** 100% success. That 1/1 success.
  - **ISS:** 61.9% success. That 13/21 success.
  - **LEO:** 71.4% success. That 5/7 success.
  - **MEO:** 66.67% success. That 2/3 success.
  - **PO:** 66.67% success. That 6/9 success.
  - **SO:** 0% success. That 0/1 success.
  - **SSO:** 100% success. That 5/5 success.
  - **VLEO:** 85.7% success. That 12/14 success.
- **X axis (horizontal) - Flight number :**
  - The axis starts at 0 and goes beyond 100. Points are scattered along the axis, representing different flights.
- **Data point:**
  - Each point represents a specific fight at a specific orbit. The colour of the point (blue for class 0 and orange for class 1) indicates the class to which the flight belongs.
- **Legend:**
  - Located at the top right, it clarifies that the blue dots correspond to class 0, so a fail and the orange dots to class 1, so a success.

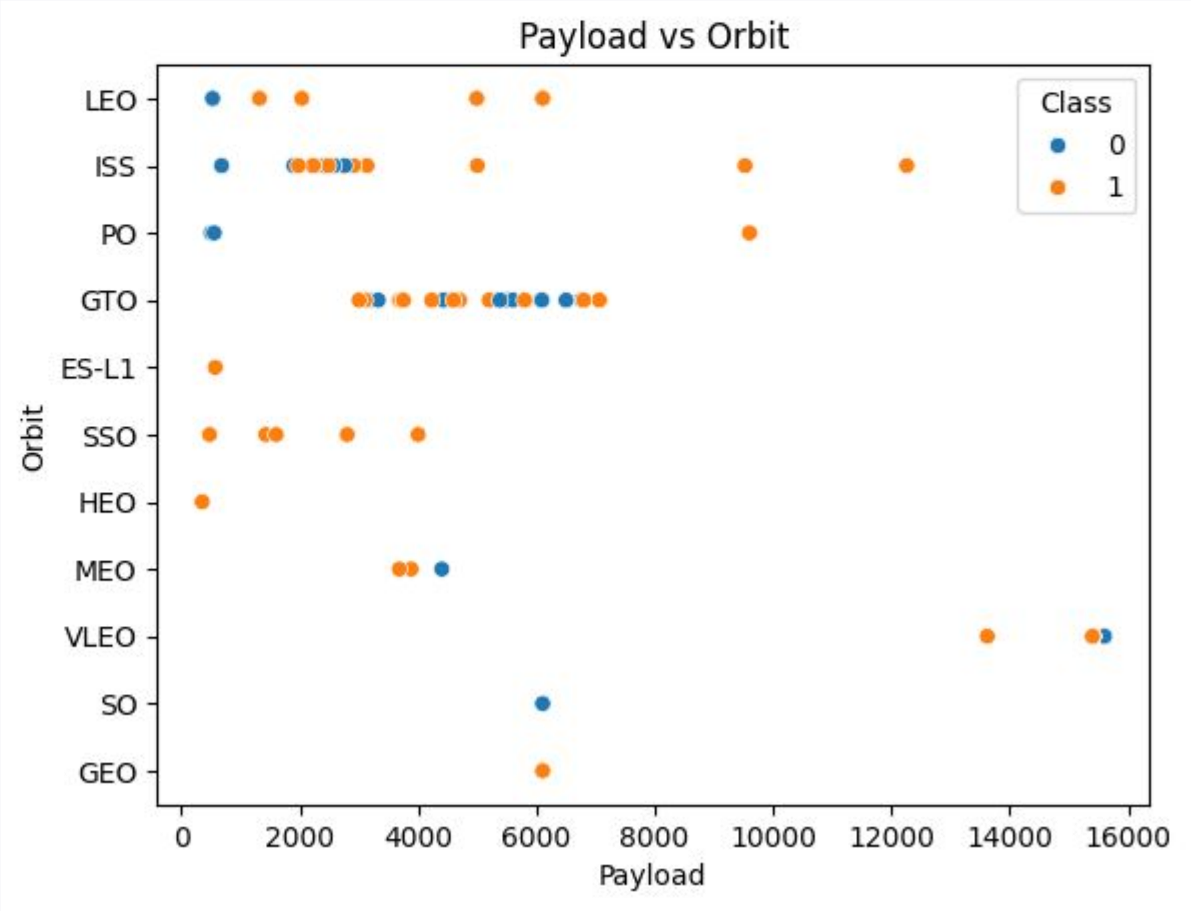


Conclusion: We can assume that ISS, GTO, VLEO are the most accurated orbit because they were more fight. So we can say that GTO and VLEO are the most successful.

Payload vs. Orbit Type

- **Y axis (vertical) - Orbit :**
  - This axis represents the different orbits.
- **X axis (horizontal) - Payload Mass:**
  - The axis starts at 0 and goes beyond 16000. Points are scattered along the axis, representing different payload mass.
- **Data point:**
  - Each point represents a specific payload at a specific orbit. The colour of the point (blue for class 0 and orange for class 1) indicates the class to which the flight belongs.
- **Legend:**
  - Located at the top right, it clarifies that the blue dots correspond to class 0, so a fail and the orange dots to class 1, so a success.

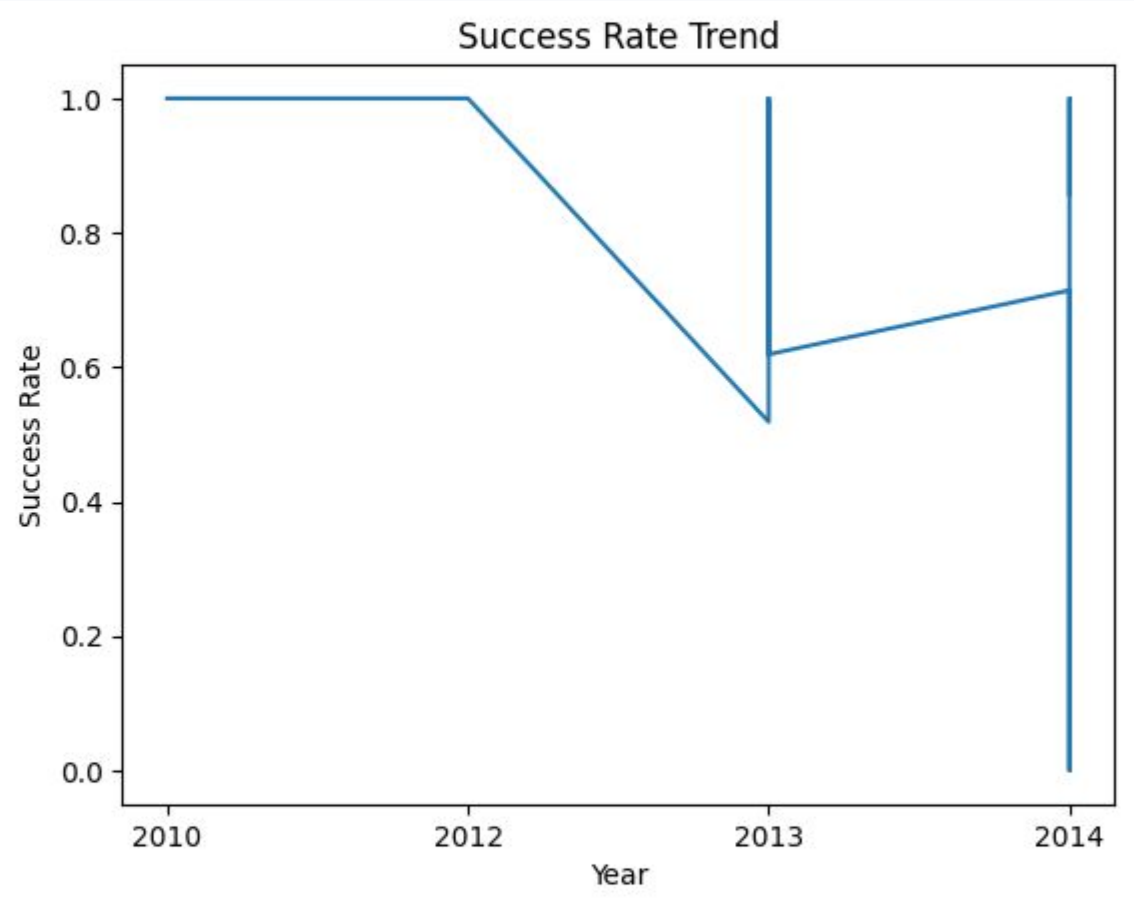
Conclusion: We can assume that between 2000 and 8000 payload mass the success is maximum.



Launch Success Yearly Trend

- **Y axis (vertical) - success Rate :**
  - The axis starts at 0 and goes beyond 1.0. This represent 0% to 100%.
- **X axis (horizontal) - Years:**
  - The axis starts at 2010 and goes beyond 2014.
- **Data line:**
  - The line represents increase or decrease of the success depending on the year.

Conclusion: We can assume that between 2010 and 2012 the success rate was at 100% between 2012 and 2013 they lost 50% and they increased between 2013 and 2014 form 50% to 70%. The two max in 2013 and 2014 are irrelevant it seems that there are null data.



All Launch Site Names

- **Launch Site:**
  - **CCAFS SLC-40 or CCAFS LC-40:**
    - Space Launch Complex 40(SLC-40), sometimes pronounced Slick Forty and previously Launch Complex 40 (LC-40) is a launch pad for rockets located at the north end of Cape Canaveral Space Force Station, Florida.
  - **VAFB SLC-4E:**
    - Space Launch Complex 4 (SLC-4) is a launch and landing site at Vandenberg Space Force Base, California, U.S. It has two pads, both of which are used by SpaceX for Falcon 9, one for launch operations, and the other as Landing Zone 4 (LZ-4) for SpaceX landings.
  - **KSC LC-39A:**
    - Launch Complex 39A (LC-39A) is the first of Launch Complex 39's three launch pads, located at NASA's Kennedy Space Center in Merritt Island, Florida. The pad, along with Launch Complex 39B, was first designed to accommodate the Saturn V launch vehicle. Typically used to launch NASA's crewed spaceflight missions since the late 1960s, the pad was leased by SpaceX and has been modified to support their launch vehicles.

Launch Site Names Begin with 'CCA'

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

- **Date:** The date (in UTC) when the launch took place.
- **Time (UTC):** The time (in UTC) when the launch occurred.
- **Booster_Version:** The version of the Falcon 9 booster used for the launch.
- **Launch_Site:** The launch site from which the Falcon 9 was launched (e.g., CCAFS LC-40).
- **Payload:** The payload carried by the Falcon 9 rocket.
- **PAYLOAD_MASS__KG_:** The mass of the payload carried by the Falcon 9 rocket in kilograms.
- **Orbit:** The orbit in which the payload was deployed (e.g., Low Earth Orbit - LEO).
- **Customer:** The customer or organization that commissioned the mission.
- **Mission_Outcome:** The outcome of the mission, indicating whether it was successful or not.
- **Landing_Outcome:** The outcome of the booster landing attempt, indicating whether it was successful, unsuccessful, or if there was no attempt made.

Total Payload Mass

- **Query:**
  - **SELECT SUM(PAYLOAD_MASS__KG_) AS Total_Payload_Mass FROM SPACEXTABLE WHERE Customer = 'NASA (CRS)'**

- **SELECT SUM(PAYLOAD_MASS__KG_) AS Total_Payload_Mass:**
  - This part of the query selects the sum of the **"PAYLOAD_MASS__KG_"** column from the **SPACEXTABLE**.
  - The "AS Total_Payload_Mass" alias renames the resulting column to **"Total_Payload_Mass"** for clarity.
- **FROM SPACEXTABLE:**
  - Specifies the table from which the data will be retrieved, in this case, **SPACEXTABLE**.
- **WHERE Customer = 'NASA (CRS)':**
  - Filters the data to include only rows where the **"Customer"** column equals **'NASA (CRS)'**.
  - This condition restricts the data to missions under the CRS program commissioned by NASA.

- The total payload carried by boosters from NASA is 45596 KG.

Average Payload Mass by F9 v1.1

- **Query:**
  - **SELECT AVG(PAYLOAD_MASS__KG_) AS Average_Payload_Mass FROM SPACEXTABLE WHERE Booster_Version = 'F9 v1.1'**

- **SELECT AVG(PAYLOAD_MASS__KG_) AS Average_Payload_Mass:**
  - This part of the query selects the average of the **"PAYLOAD_MASS__KG_"** column from the **SPACEXTABLE**.
  - The **"AS Average_Payload_Mass"** alias renames the resulting column to **"Average_Payload_Mass"** for clarity.
- **FROM SPACEXTABLE:**
  - Specifies the table from which the data will be retrieved, in this case, SPACEXTABLE.
- **WHERE Booster_Version = 'F9 v1.1':**
  - Filters the data to include only rows where the **"Booster_Version"** column equals **'F9 v1.1'**.
  - This condition restricts the data to launches that utilized Falcon 9 version 1.1 boosters.

- The average payload mass carried by booster version F9 v1.1 is 2928.4 KG.

First Successful Ground Landing Date

- **Query:**
  - **SELECT MIN(Date) AS First_Successful_Landing_Date FROM SPACEXTABLE WHERE Landing_Outcome = 'Success (ground pad)'**

- **SELECT MIN(Date) AS First_Successful_Landing_Date:**
  - This part of the query selects the minimum (earliest) date from the **"Date"** column of the **SPACEXTABLE**.
  - The **"AS First_Successful_Landing_Date"** alias renames the resulting column to **"First_Successful_Landing_Date"** for clarity.
- **FROM SPACEXTABLE:**
  - Specifies the table from which the data will be retrieved, in this case, **SPACEXTABLE**.
- **WHERE Landing_Outcome = 'Success (ground pad)'**:
  - Filters the data to include only rows where the **"Landing_Outcome"** column equals 'Success (ground pad)'.
  - This condition restricts the data to successful landings on ground pads.

- The dates of the first successful landing outcome on ground pad is 2015-12-22.

Successful Drone Ship Landing with Payload between 4000 and 6000

- **Query:**
  - **SELECT Booster_Version FROM SPACEXTABLE WHERE Landing_Outcome = 'Success (drone ship)' AND PAYLOAD_MASS__KG_ > 4000 AND PAYLOAD_MASS__KG_ < 6000**

- **SELECT Booster_Version:**
  - This part of the query selects the values from the **"Booster_Version"** column of the **SPACEXTABLE.**
- **FROM SPACEXTABLE:**
  - Specifies the table from which the data will be retrieved, in this case, **SPACEXTABLE.**
- **WHERE Landing_Outcome = 'Success (drone ship)':**
  - Filters the data to include only rows where the **"Landing_Outcome"** column equals **'Success (drone ship)'.**
  - This condition restricts the data to successful landings on a drone ship.
- **AND PAYLOAD_MASS__KG_ > 4000 AND PAYLOAD_MASS__KG_ < 6000:**
  - Further filters the data to include only rows where the **"PAYLOAD_MASS__KG_"** column falls within the range of 4000 to 6000 kilograms.
  - This condition restricts the data to payloads within the specified mass range.

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

Total Number of Successful and Failure Mission Outcomes

- **Query:**
  - **SELECT Mission_Outcome, COUNT(*) AS Total_Count FROM SPACEXTABLE GROUP BY Mission_Outcome**

- **SELECT Mission_Outcome, COUNT(*) AS Total_Count:**
  - This part of the query selects the **"Mission_Outcome"** column and calculates the count of occurrences for each unique mission outcome in the **SPACEXTABLE**.
  - The **"AS Total_Count"** alias renames the resulting count column to "Total_Count" for clarity.
- **FROM SPACEXTABLE:**
  - Specifies the table from which the data will be retrieved, in this case, **SPACEXTABLE**.
- **GROUP BY Mission_Outcome:**
  - Groups the results based on the values in the **"Mission_Outcome"** column.
  - This groups the data into subsets where each subset contains rows with the same mission outcome.

Mission_Outcome	Total_Count
Failure (in flight)	1
Success	98
Success	1
Success (payload status unclear)	1

Boosters Carried Maximum Payload

- **Query:**
  - **SELECT Booster_Version FROM SPACEXTABLE WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTABLE)**

- **SELECT Booster_Version:**
  - This part of the query selects the values from the **"Booster_Version"** column of the **SPACEXTABLE**.
- **FROM SPACEXTABLE:**
  - Specifies the table from which the data will be retrieved, in this case, **SPACEXTABLE**.
- **WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTABLE):**
  - Filters the data to include only rows where the **"PAYLOAD_MASS__KG_"** column equals the maximum payload mass found in the entire SPACEXTABLE.
  - The subquery **"(SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTABLE)"** calculates the maximum payload mass from the **SPACEXTABLE**, and the outer query selects the rows where the payload mass matches this maximum value.

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

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

- **SELECT CASE ... END AS Month:**
    - This part of the query uses the CASE statement to extract the month from the "Date" column and map it to the corresponding month name.
    - It creates a new column named "Month" to store the mapped month names.
- **Landing_Outcome, Booster_Version, Launch_Site:**
    - These columns are selected directly to be included in the result set.
    - The query retrieves the landing outcome, booster version, and launch site information for the selected Falcon 9 launches.
- **FROM SPACEXTABLE:**
    - Specifies the table from which the data will be retrieved, in this case, SPACEXTABLE.
- **WHERE substr(Date, 0, 5) = '2015':**
    - Filters the data to include only rows where the year portion of the "Date" column equals '2015'.
- **AND Landing_Outcome LIKE '%Failure%' AND Landing_Outcome LIKE '%drone ship%':**
    - Further filters the data to include only rows where the landing outcome contains the word 'Failure' and the phrase 'drone ship'.
    - This condition restricts the data to missions with a landing outcome indicating failure on a drone ship.

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

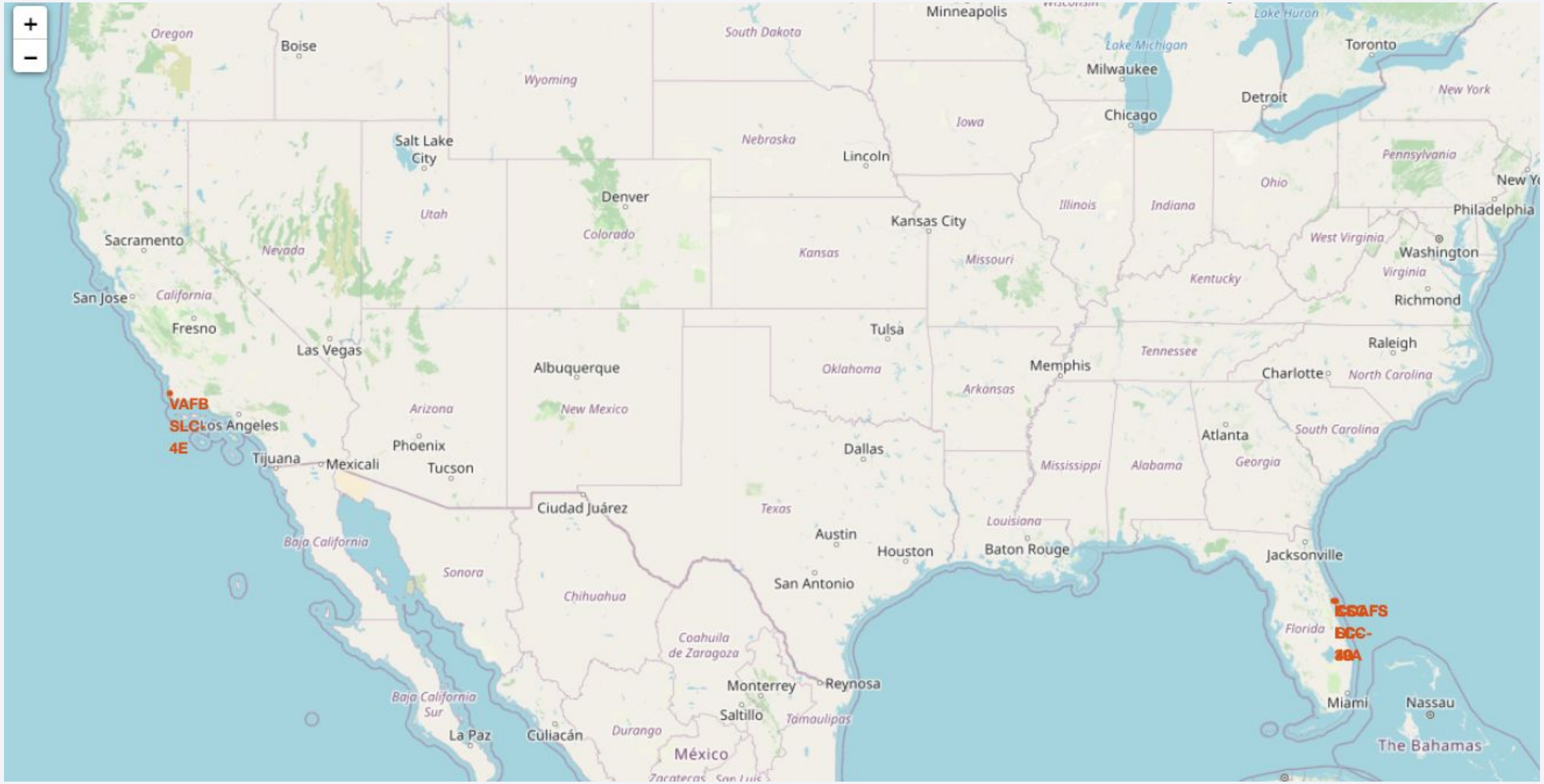
- **SELECT Landing_Outcome, COUNT(*) AS Count:**
  - This part of the query selects the "Landing_Outcome" column and calculates the count of occurrences for each unique landing outcome in the SPACEXTABLE.
  - The "AS Count" alias renames the resulting count column to "Count" for clarity.
- **FROM SPACEXTABLE:**
  - Specifies the table from which the data will be retrieved, in this case, SPACEXTABLE.
- **WHERE Date >= '2010-06-04' AND Date <= '2017-03-20':**
  - Filters the data to include only rows where the "Date" column falls within the specified date range.
  - This condition restricts the data to SpaceX Falcon 9 launches that occurred between June 4, 2010, and March 20, 2017.
- **GROUP BY Landing_Outcome:**
  - Groups the results based on the values in the "Landing_Outcome" column.
  - This groups the data into subsets where each subset contains rows with the same landing outcome.
- **ORDER BY Count DESC:**
  - Orders the results in descending order based on the count of occurrences.
  - This arranges the landing outcomes with the highest count first in the result set.

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

Section 3

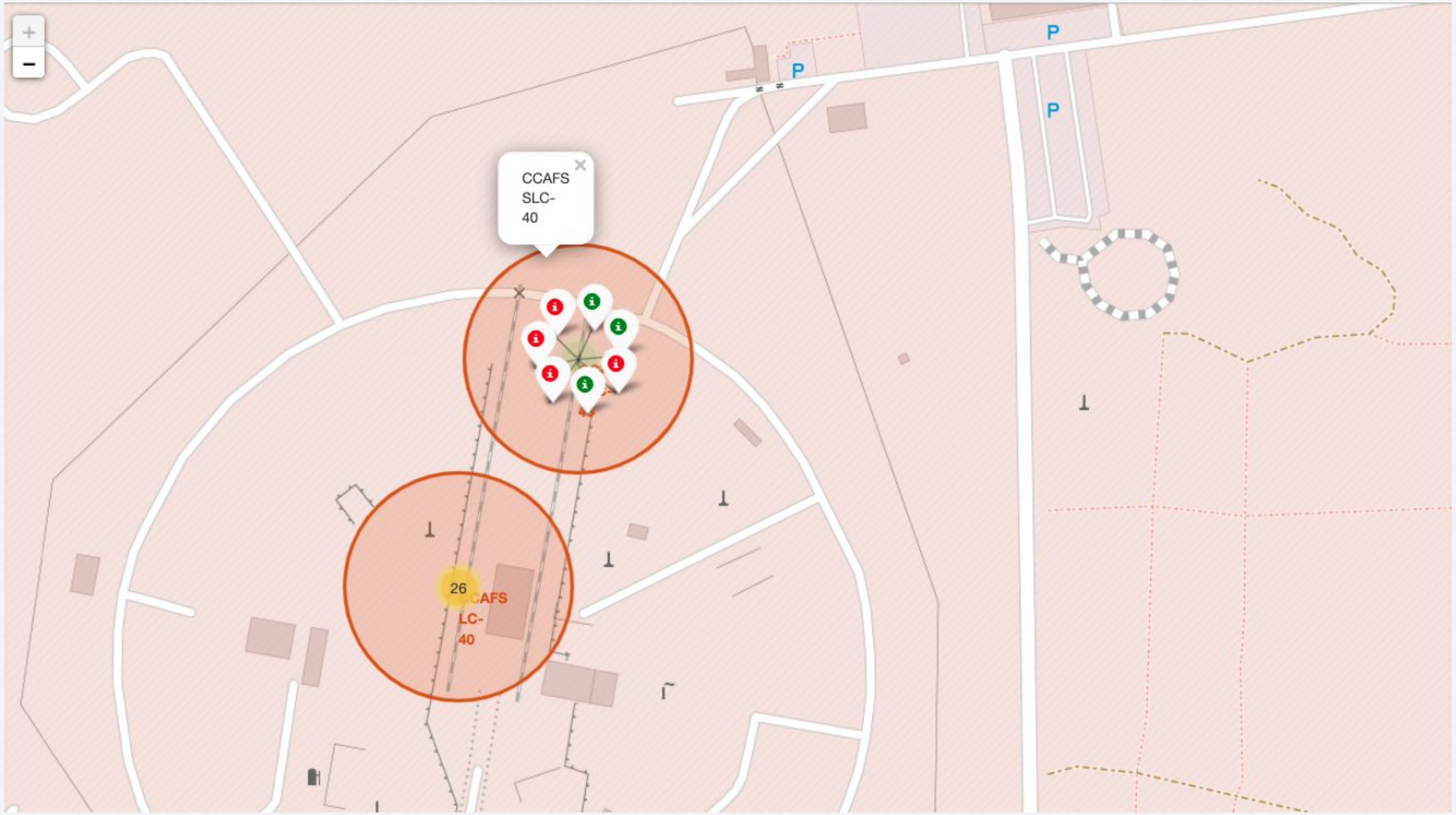
Launch Sites Proximities Analysis

Folium map Launch sites



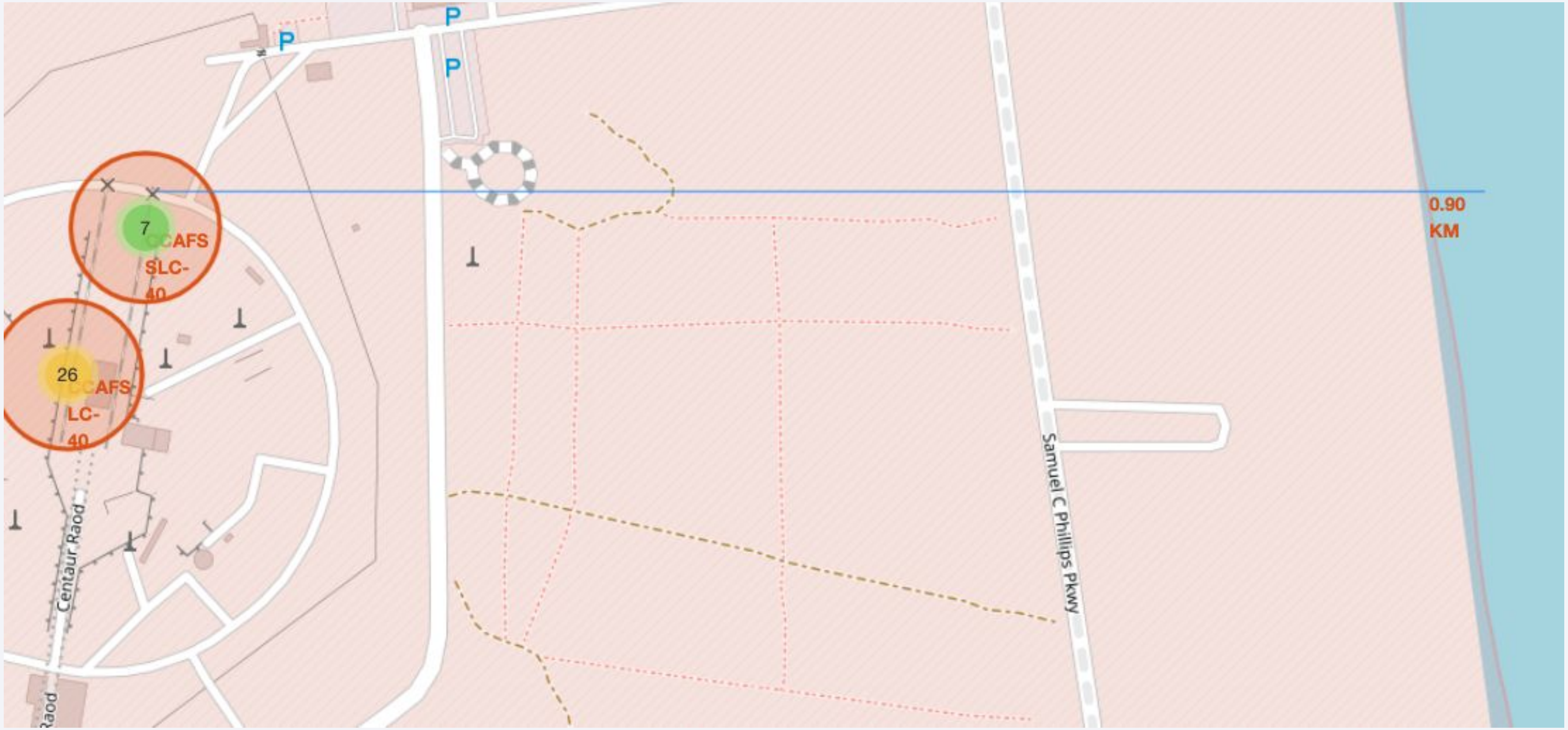
You can find the 3 launch sites

Folium map Success rate Launch Site



You can see 3 success launch that are in green for the launch site CCAFS SLC-40

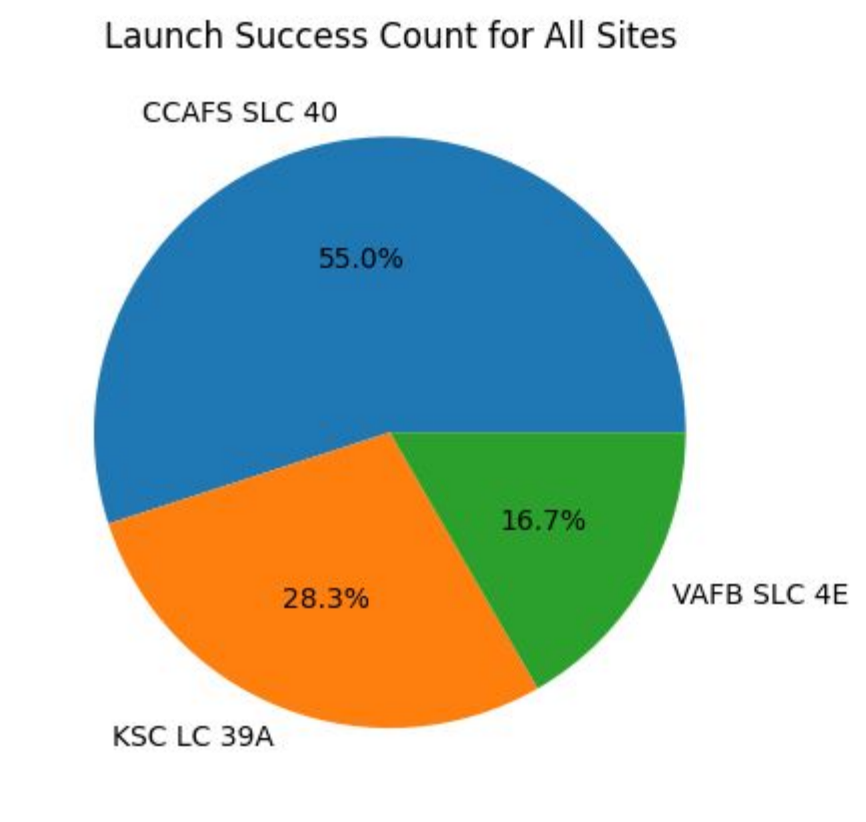
Folium map distantce launch



Section 4

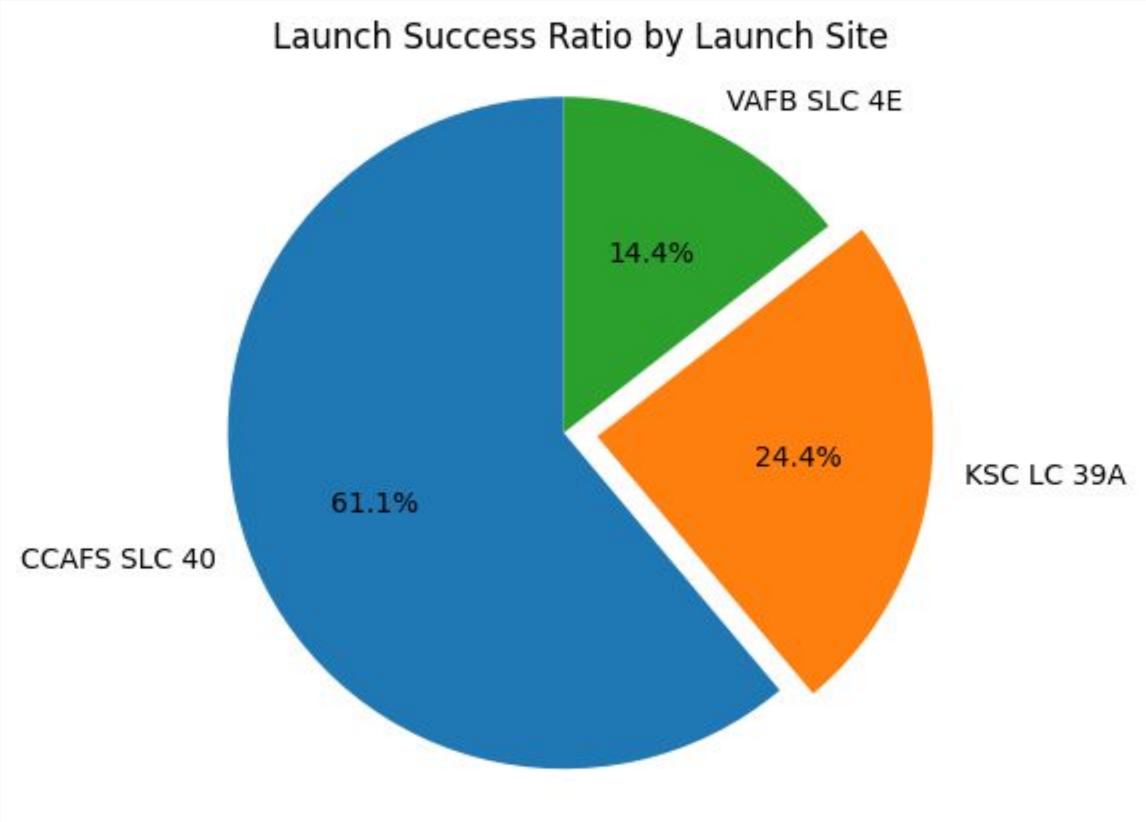
Build a Dashboard with Plotly Dash

Launch Success Count for all sites



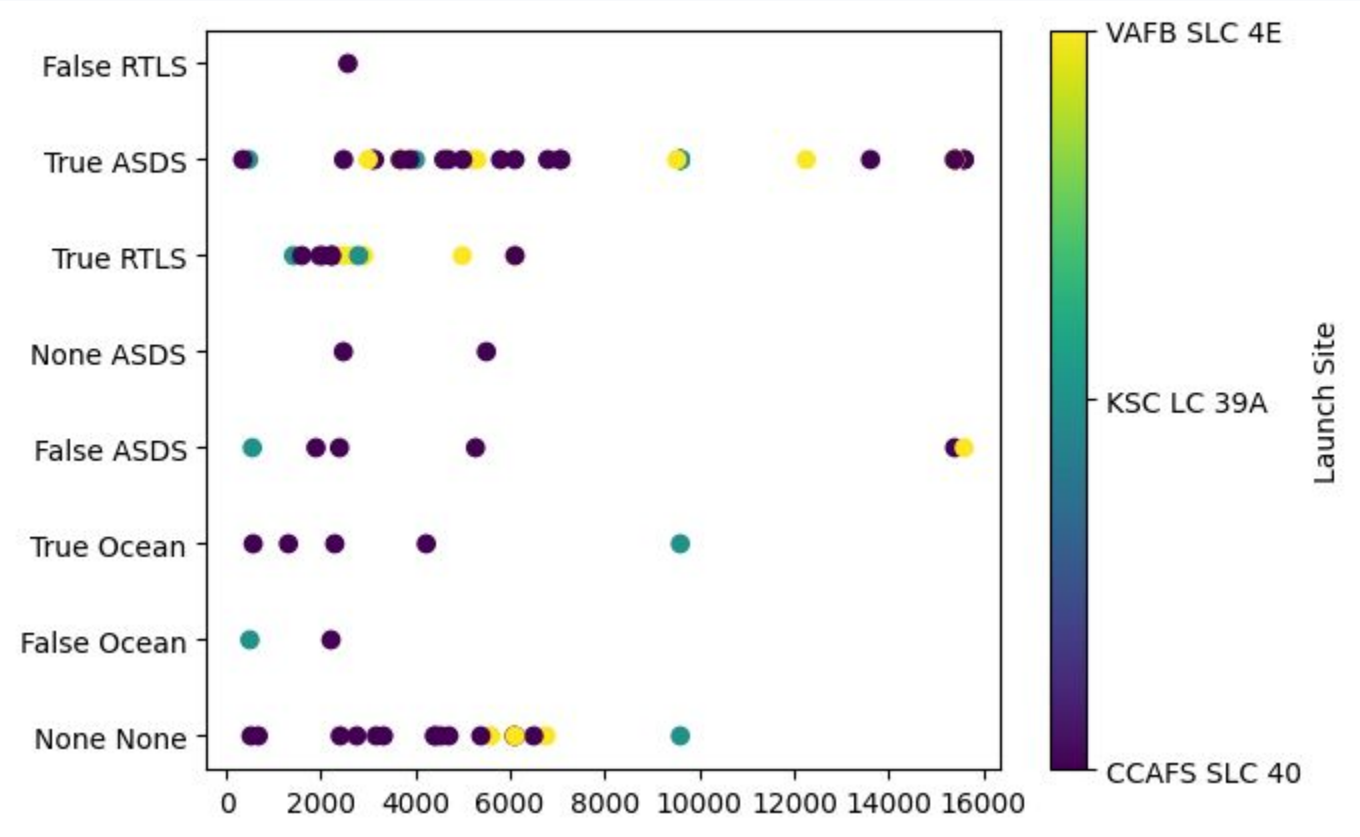
We can says that CCAFS SLC 40 as more launch success by count than the other sites.

Launch Success Ration by Launch Site



We can says that CCAFS SLC 40 as more launch success by ratio than the other sites.

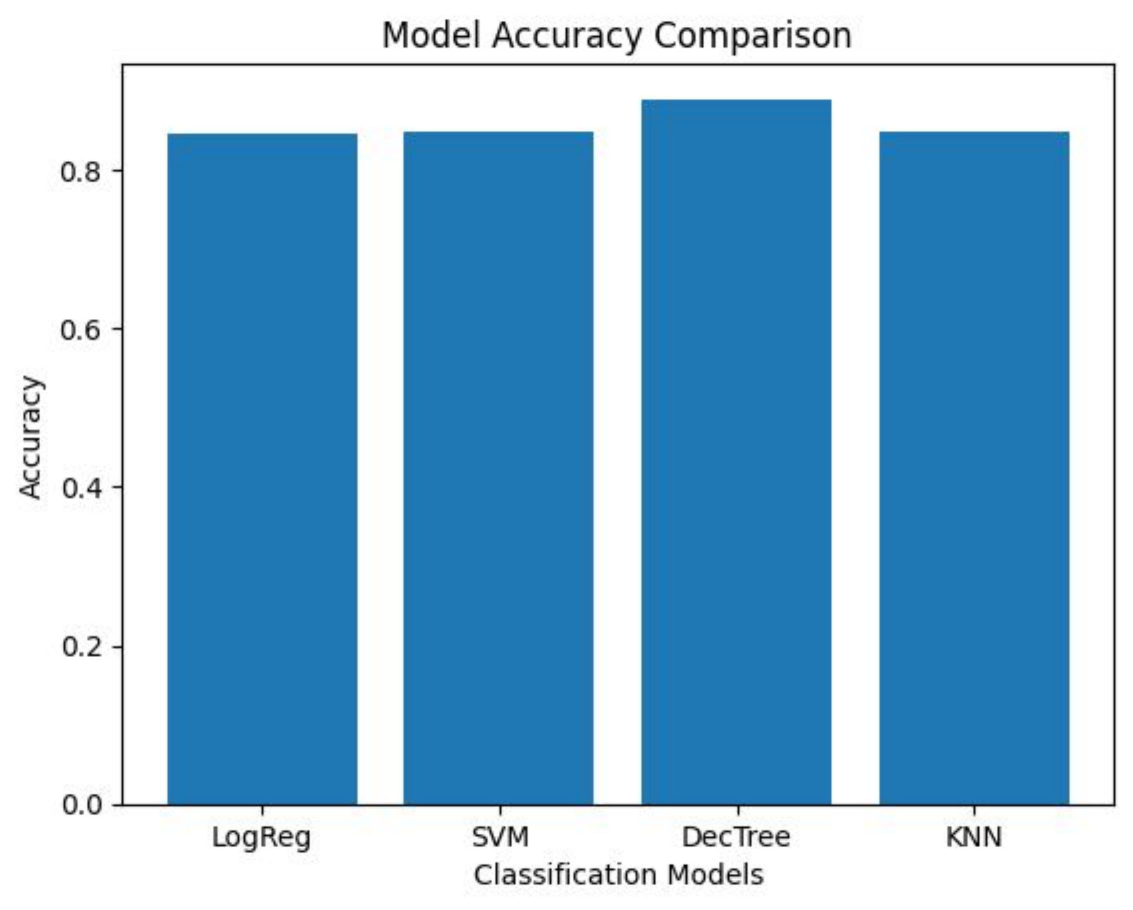
Payload vs. Launch Outcome scatter plot for all sites



Section 5

Predictive Analysis (Classification)

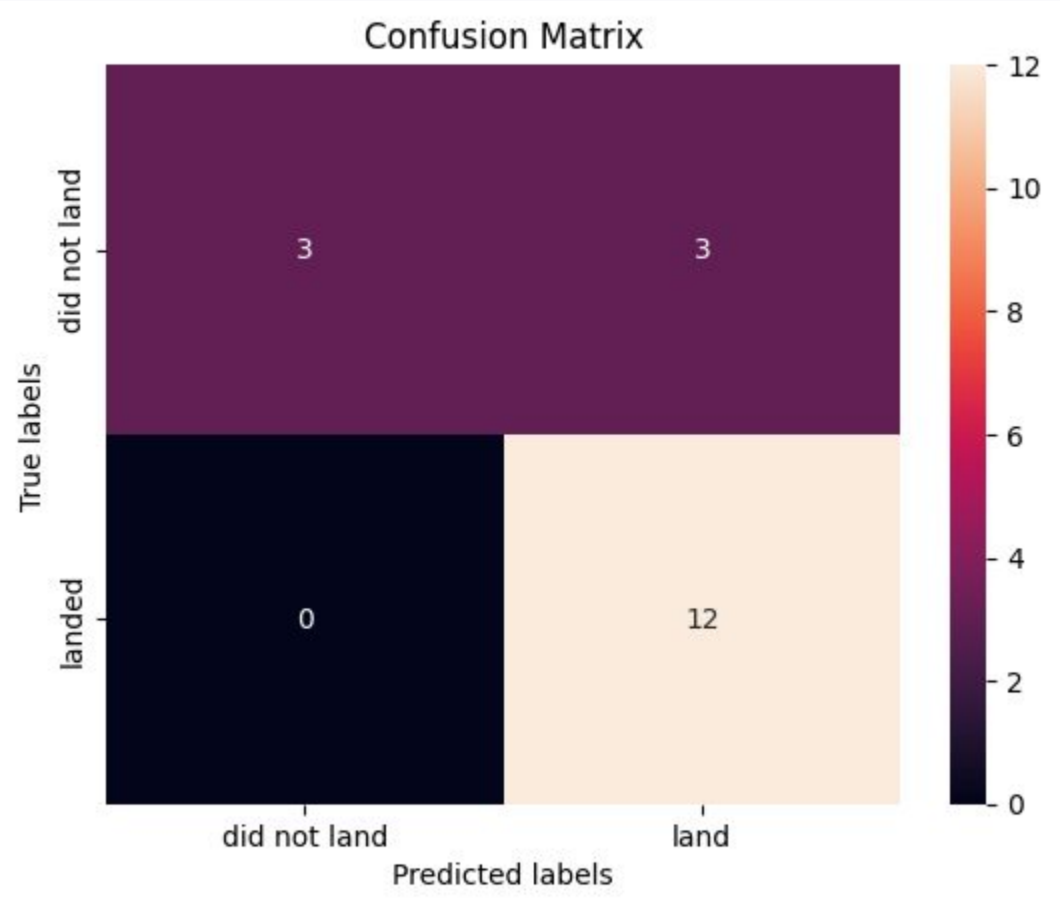
Classification Accuracy



The decision tree seems to be the most accurate model

Confusion Matrix

- **True Positives (TP):** 12 (Actual: Landed, Predicted: Landed)
- **False Positives (FP):** 0 (Actual: Did not land, Predicted: Landed)
- **False Negatives (FN):** 3 (Actual: Landed, Predicted: Did not land)
- **True Negatives (TN):** 3 (Actual: Did not land, Predicted: Did not land)



Conclusions

- In conclusion, based on the superior accuracy of the decision tree and the analysis of the confusion matrix, we can recommend the use of the decision tree as a predictive model for this specific task of predicting launch results.

Thank you!