IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Daniel Silva

January 11, 2024

Outline

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

Executive Summary

Summary of methodologies

Data collection

  - SpaceX Rest API
  - Web sacrapping from Wikipedia
- Data Wrangling
- Exploratory Data Analysis (EDA)
- Data Visualization
  - Folium (maps)
  - Plotly Dash (dashboard)
- Predictive Analysis
  - Classification models

Summary of all results

- Exploratory Data Analysis (EDA) Results
- Dynamic Visualizations (screenshots)
- Machine Learning Model Results for Predictive Analysis

Introduction

SpaceX's cost-saving innovation involves reusing the first stage of the launch, leading to substantial savings.

This project aims to analyze historical SpaceX launch data, focusing on forecasting Falcon 9's first stage landing outcomes.

Machine Learning not only enhance understanding of landing success but also provide valuable insights into potential launch costs.

The acquired information holds strategic significance for competing companies bidding against SpaceX for upcoming rocket launches.

Introduction

Motivation of the project

Solving common issues in rocket landings is crucial.

Understanding the factors influencing successful rocket landings is essential.

Identifying and achieving optimal conditions to ensure the best success rate in rocket landings.

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Data from Space X was obtained from 2 sources:
    - Space X API (https://api.spacexdata.com/v4/rockets/)
    - Web Scraping (https://en.wikipedia.org/wiki/List_of_Falcon/_9/_and_Falcon_Heavy_launches
- Perform data wrangling
  - Collected data was enriched by creating a landing outcome label based on outcome data after summarizing and analyzing features
- Perform exploratory data analysis (EDA) using visualization and SQL

Methodology

Executive Summary

- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Data that was collected until this step were normalized, divided in training and test data sets and evaluated by four different classification models, being the accuracy of each model evaluated using different combinations of parameters.

Data Collection

- SpaceX REST API: https://api.spacexdata.com/v4/launchpads/
  - SpaceX REST API provides comprehensive data on launches, including details about the rocket, payload information, specifications for launch and landing, and the outcomes of the landing.

- Web Scrapping:

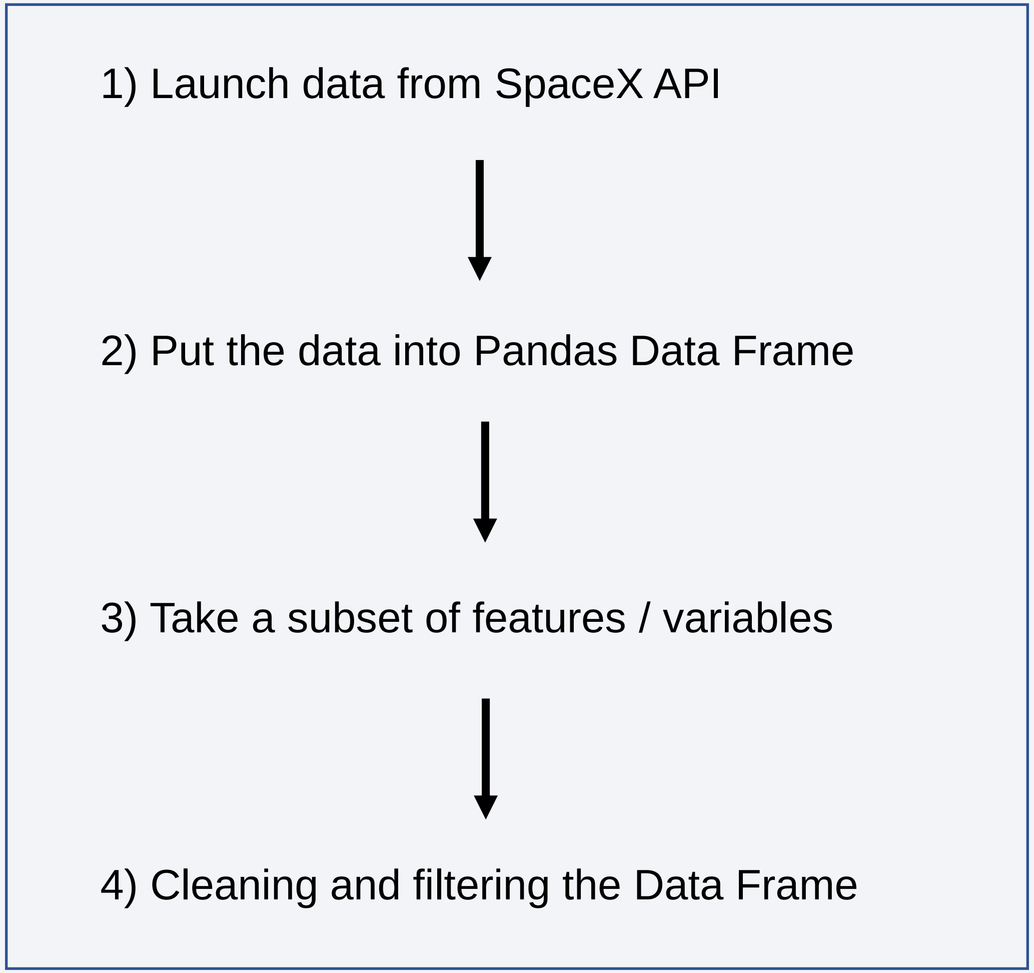
  List of Falcon 9 and Falcon Heavy launches – Wikipedia

  - Using Python's package - beautiful soup HTML tables from Wikipedia page were retrieved

Data Collection – SpaceX API

- SpaceX offers a public API where data can be obtained
- This API was used according to the flowchart beside

Source code:
https://github.com/Bombjack88/Applied-Data-Science-Capstone/blob/main/week%201%20-%20Collecting%20the%20data.ipynb



1) Launch data from SpaceX API

2) Put the data into Pandas Data Frame

3) Take a subset of features / variables

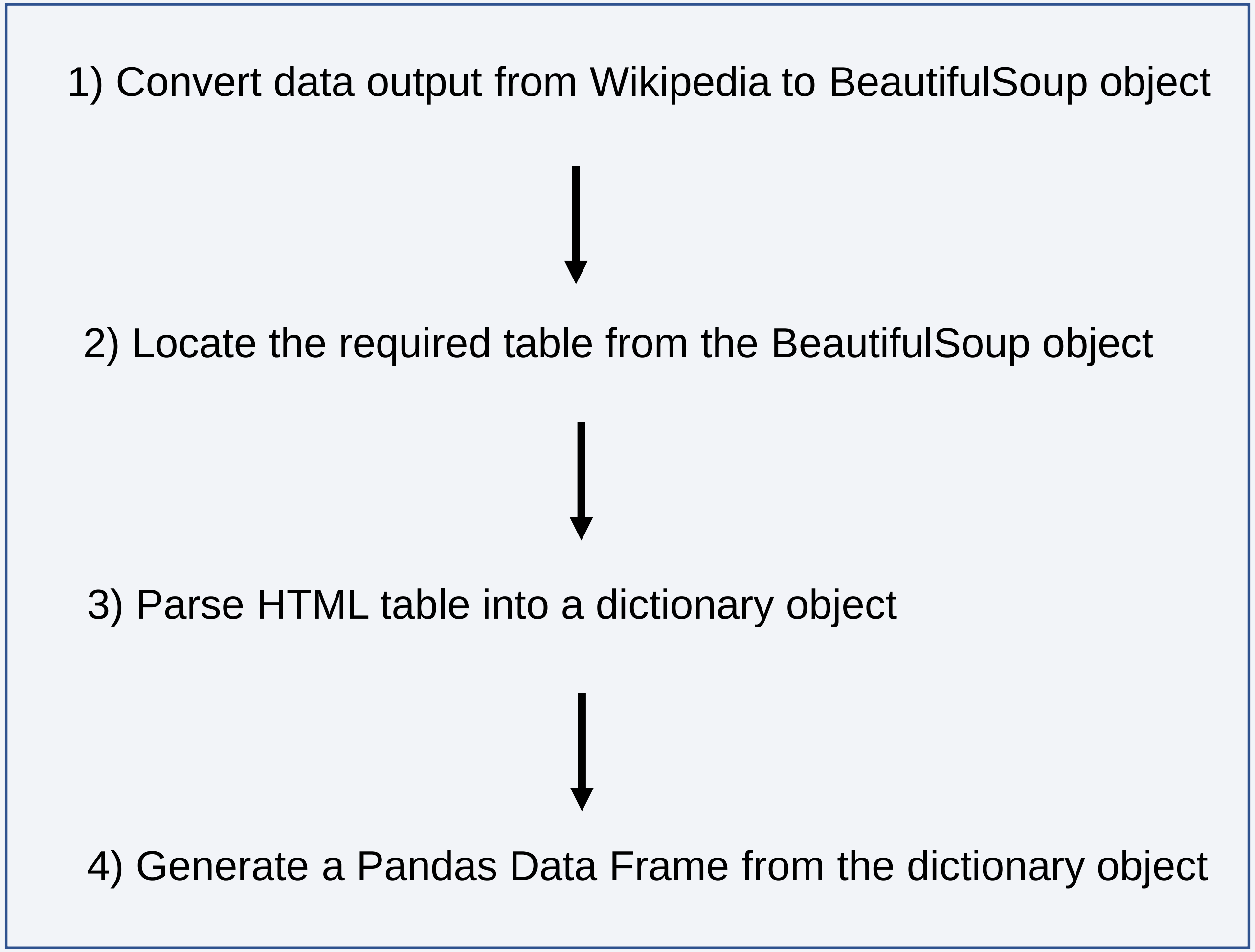
4) Cleaning and filtering the Data Frame

Data Collection - Scraping

- Data was collected via web scraping from:

[List of Falcon 9 and Falcon Heavy launches - Wikipedia]

Source code: https://github.com/Bombjack88/Applied-Data-Science-Capstone/blob/main/week%201.1%20-%20Web%20scraping%20Falcon%209%20and%20Falcon%20Heavy%20Launches%20Records%20from%20Wikipedia.ipynb



1) Convert data output from Wikipedia to BeautifulSoup object

2) Locate the required table from the BeautifulSoup object

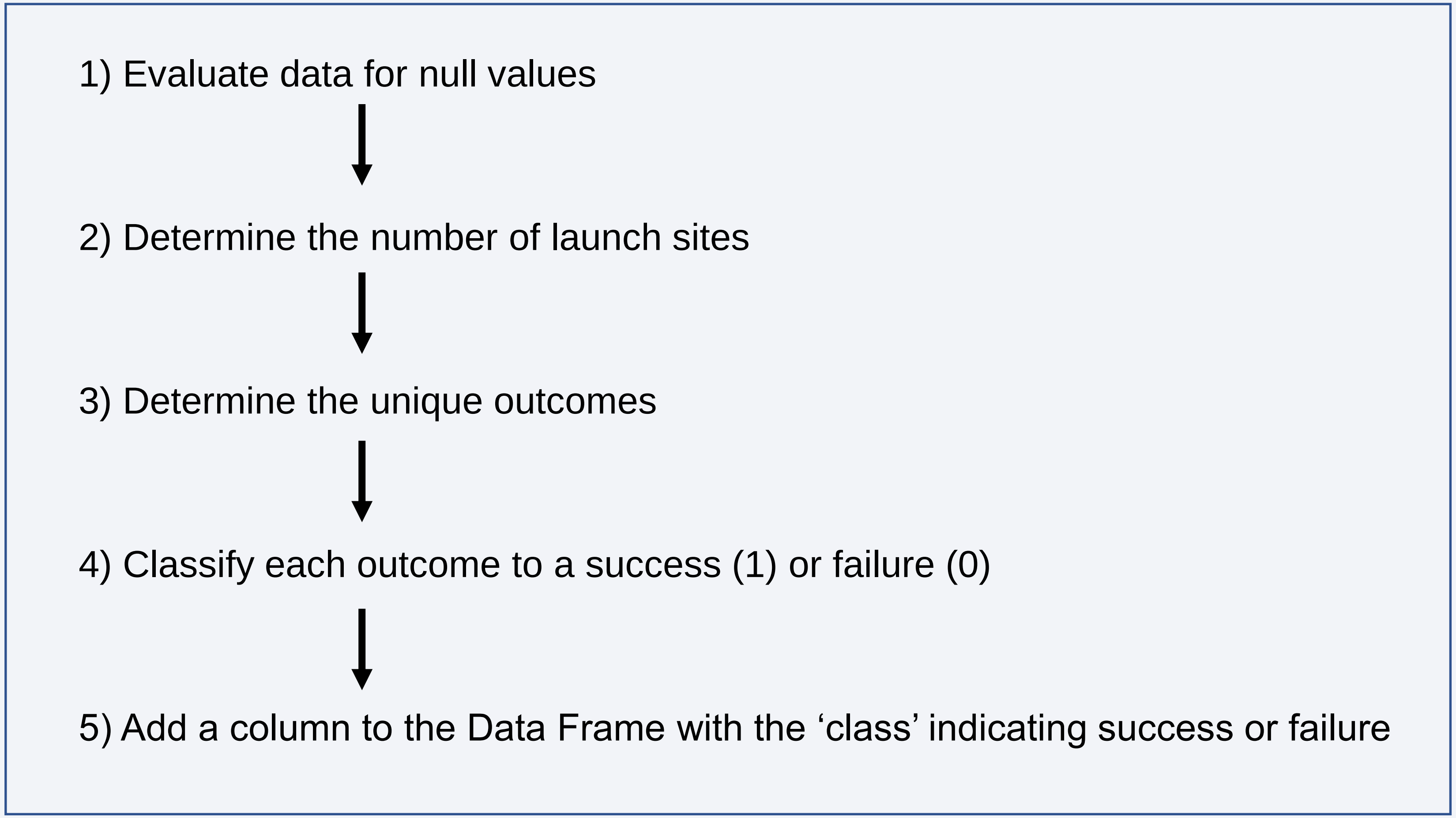
3) Parse HTML table into a dictionary object

4) Generate a Pandas Data Frame from the dictionary object

Data Wrangling

This flowchart demonstrates the process we undertook when wrangling the data for this project

Source code: https://github.com/Bombjack88/Applied-Data-Science-Capstone/blob/main/week%201.2%20-%20Space%20X%20%20Falcon%209%20First%20Stage%20Landing%20Prediction%20-%20Data%20wrangling%20.ipynb



1) Evaluate data for null values

2) Determine the number of launch sites

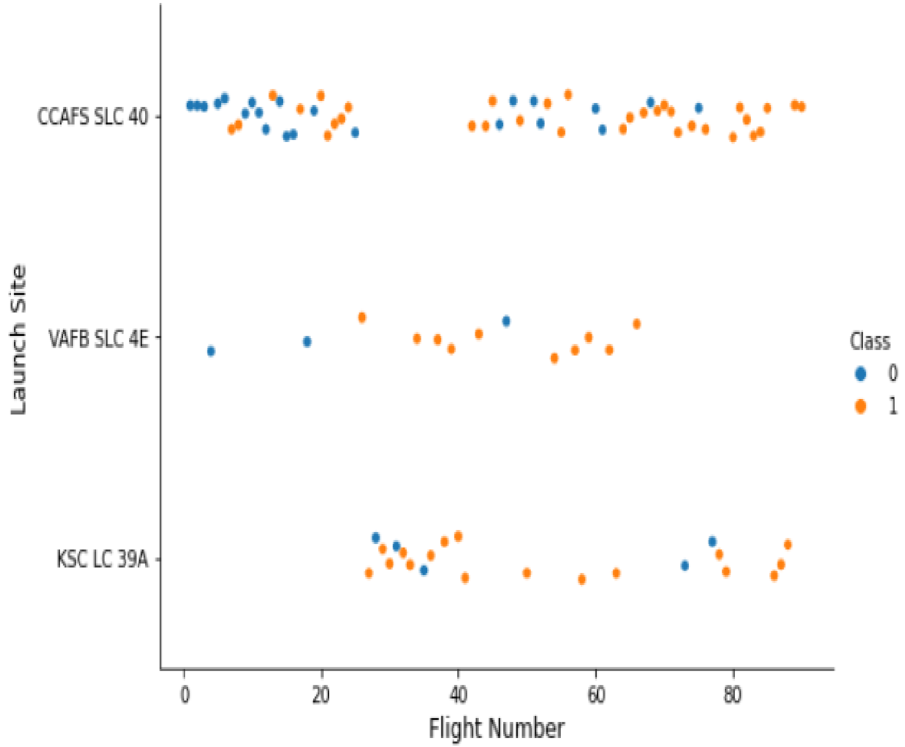
3) Determine the unique outcomes

4) Classify each outcome to a success (1) or failure (0)

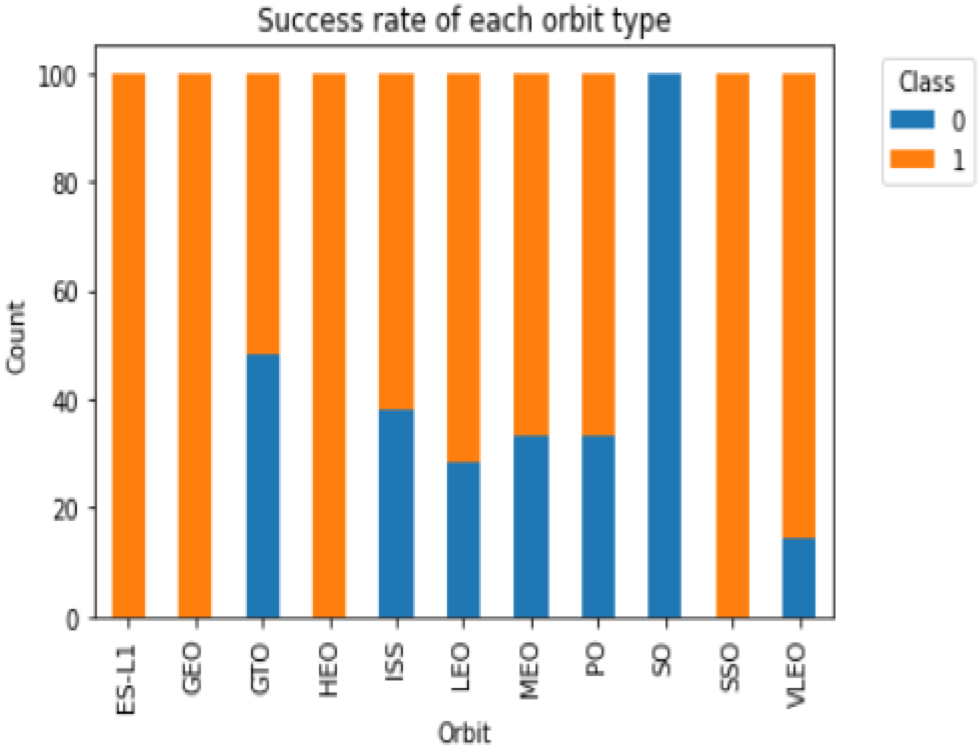
5) Add a column to the Data Frame with the ‘class’ indicating success or failure

EDA with Data Visualization

Graph 1: Flight Number vs. Launch Site (color: class variable)



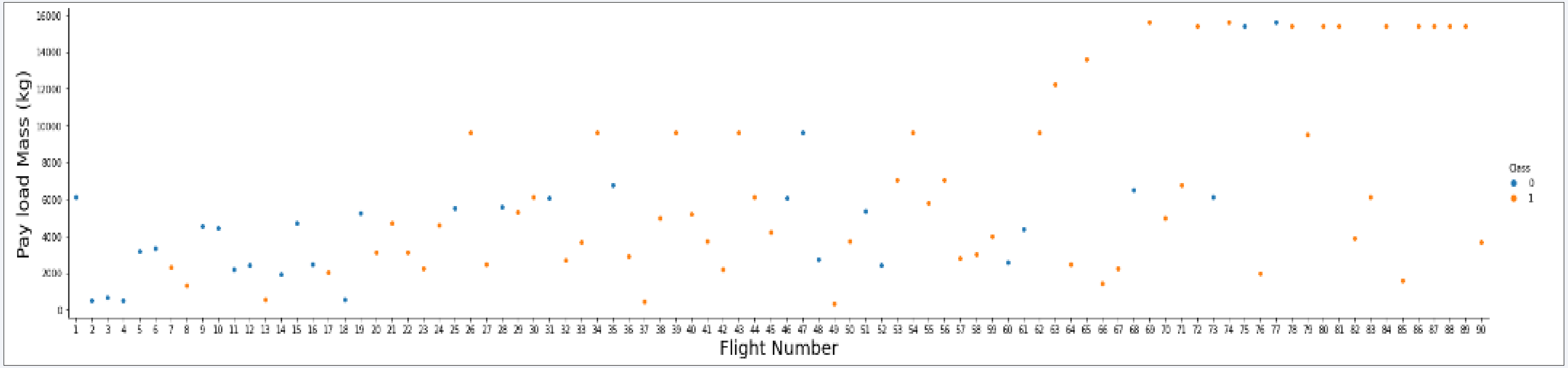
Graph 2: Success rate of each Orbit Type



Source code:
https://github.com/Bombjack88/Applied-Data-Science-Capstone/blob/main/week%202.1%20-%20Assignment%20-%20Exploring%20and%20Preparing%C2%A0Data.ipynb

EDA with Data Visualization

Graph 3: Flight Number vs. Payload Mass (color: class variable)



Other analysis:

Payload vs. Launch Site

Flight Number vs. Orbit Type

Success Rate Over Years

Flight Number vs. Orbit Type

Payload vs. Orbit Type

Main Conclusions: 1) Rate of successful landing increases with Flight Number; 2) Some ‘Orbit Types’ have highest probabilities of making a successful landing; 3) Success rate increased over the years.

EDA with SQL

- The following SQL queries were performed:
  - Names of the unique launch sites in the space mission;
  - Top 5 launch sites whose name begin with the string 'CCA’;
  - Total payload mass carried by boosters launched by NASA (CRS);
  - Average payload mass carried by booster version F9 v1.1;
  - Date when the first successful landing outcome in ground pad was achieved;
  - Names of the boosters which have success in drone ship and have payload mass between 4000 and 6000 kg;
  - Total number of successful and failure mission outcomes;
  - Names of the booster versions which have carried the maximum payload mass;
  - Failed landing outcomes in drone ship, their booster versions, and launch site names for in year 2015;
  - Rank of the count of landing outcomes (such as Failure (drone ship) or Success (ground pad)) between two dates.

Source code: https://github.com/Bombjack88/Applied-Data-Science-Capstone/blob/main/week%202%20-%20%20Assignment%20-%20SQL%20Notebook%20for%20Peer%20Assignment.ipynb

Build an Interactive Map with Folium

- We initialize a Folium map and add Circles around NASA John Space Centre (which acts as the center to our map). Also, we add red circles around the launch sites which are present in the states of Florida and California. These Circles are named after the location they enclose.
- Then we create a markercluster – to group makers close to each other.
- The come the markers, which represent the launch site coordinates and are represented by colors Red (Unsuccessful Landing) and Green (Successful Landing).
- Another element being used is PolyLine, to represent the distance between a particular launch site and nearby structures (railroad, highway, city)

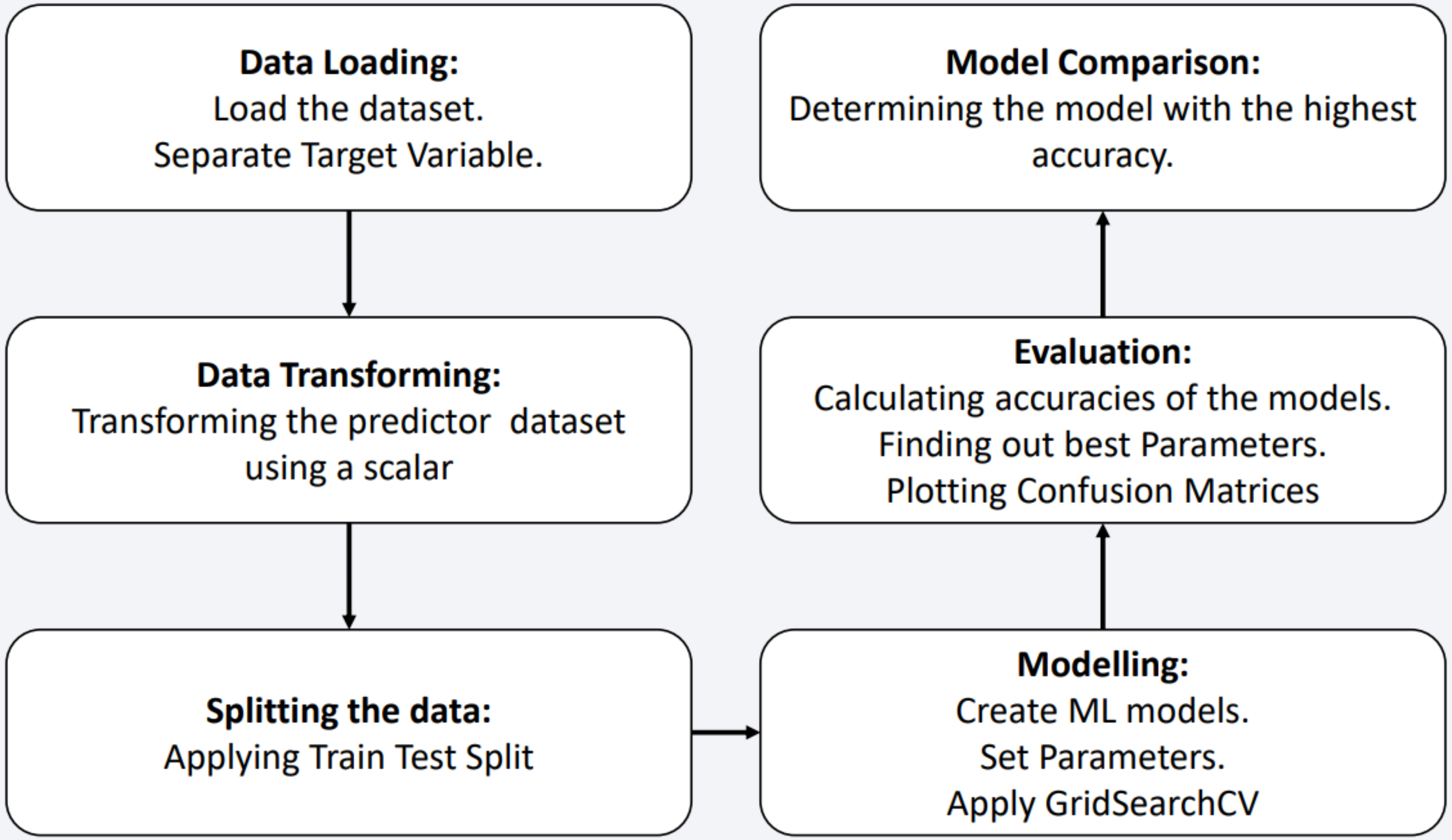
Source code:
https://github.com/Bombjack88/Applied-Data-Science-Capstone/blob/main/week%203%20-%20Launch%20Sites%20Locations%20Analysis%20with%20Folium.ipynb

Build a Dashboard with Plotly Dash

- Dashboarding has been done using Plotly. There's a dropdown menu allowing to select launch sites with default value as all sites. This dropdown interacts with the pie chart placed just below it. Which renders a pie chart containing pieces with values equal to individual number of launch sites records (when drop is at default value or All sites).
- In the case of the launch site being selected, the pie chart displays outcomes of landing.
- A Range slider at the bottom allows to control the payload mass which is plotted underneath it with the help of a scatter plot.
- The launch selected in the dropdown is also communicated to the bottommost scatter plot.
- These visualizations have been created mainly to make them dynamic and help gather more in-depth insights.

Source code:
https://github.com/Bombjack88/Applied-Data-Science-Capstone/blob/main/week%203.1%20-%20Spacex_dash_app.py

Predictive Analysis (Classification)



Data Loading:
Load the dataset.
Separate Target Variable.

Data Transforming:
Transforming the predictor dataset using a scalar

Splitting the data:
Applying Train Test Split

Modelling:
Create ML models.
Set Parameters.
Apply GridSearchCV

Evaluation:
Calculating accuracies of the models.
Finding out best Parameters.
Plotting Confusion Matrices

Model Comparison:
Determining the model with the highest accuracy.

Source code: https://github.com/Bombjack88/Applied-Data-Science-Capstone/blob/main/week%204%20-%20Assignment%20-%20%20Machine%20Learning%20Prediction.ipynb

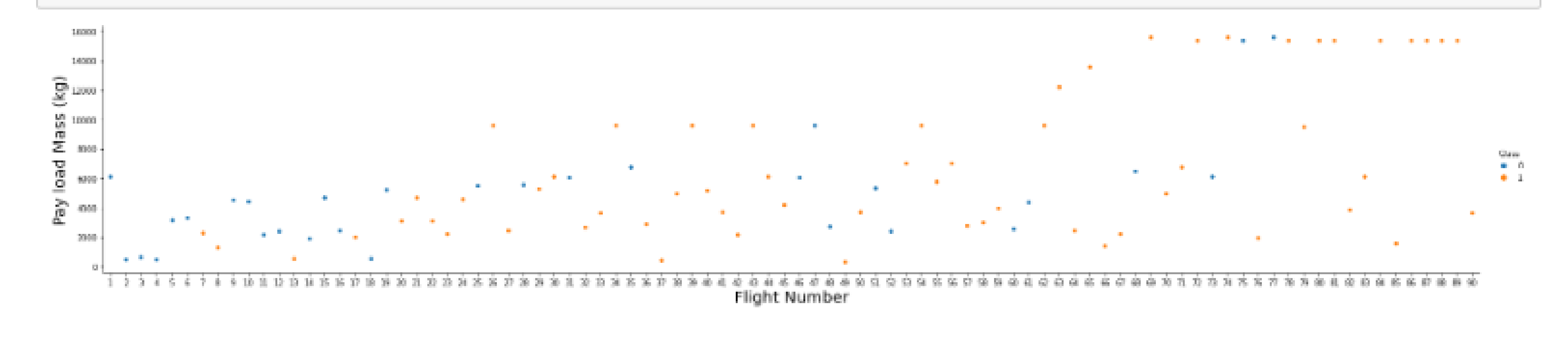
Results

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

Section 2

Insights drawn from EDA

Flight Number vs. Launch Site

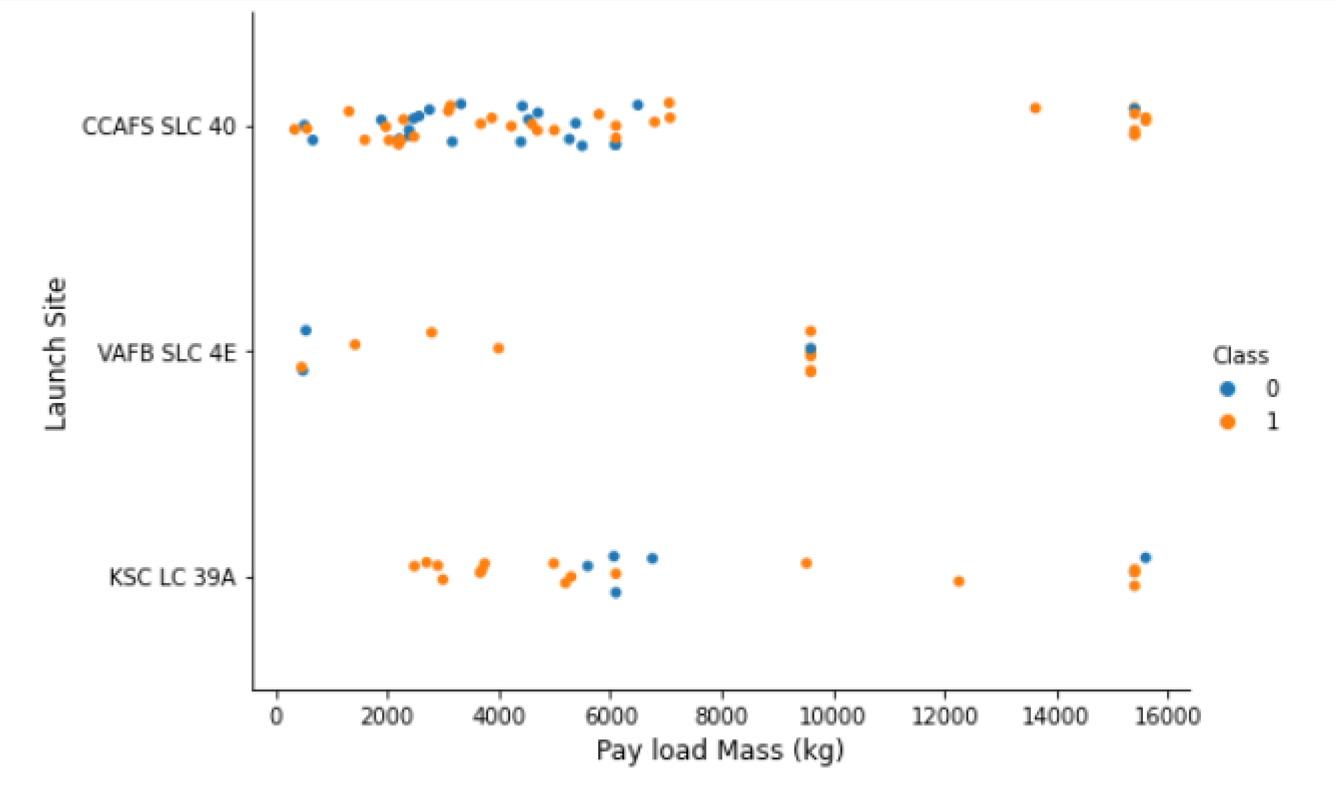


Main Conclusions:

- According to the plot above, it’s possible to verify that the best launch site nowadays is CCAF5 SLC 40 where most of recent launches were successful. In second place VAFB SLC 4E and third place KSC LC 39A;

- It’s also possible to see that the general success rate improved over time. Higher the flight number, higher the successful rate.

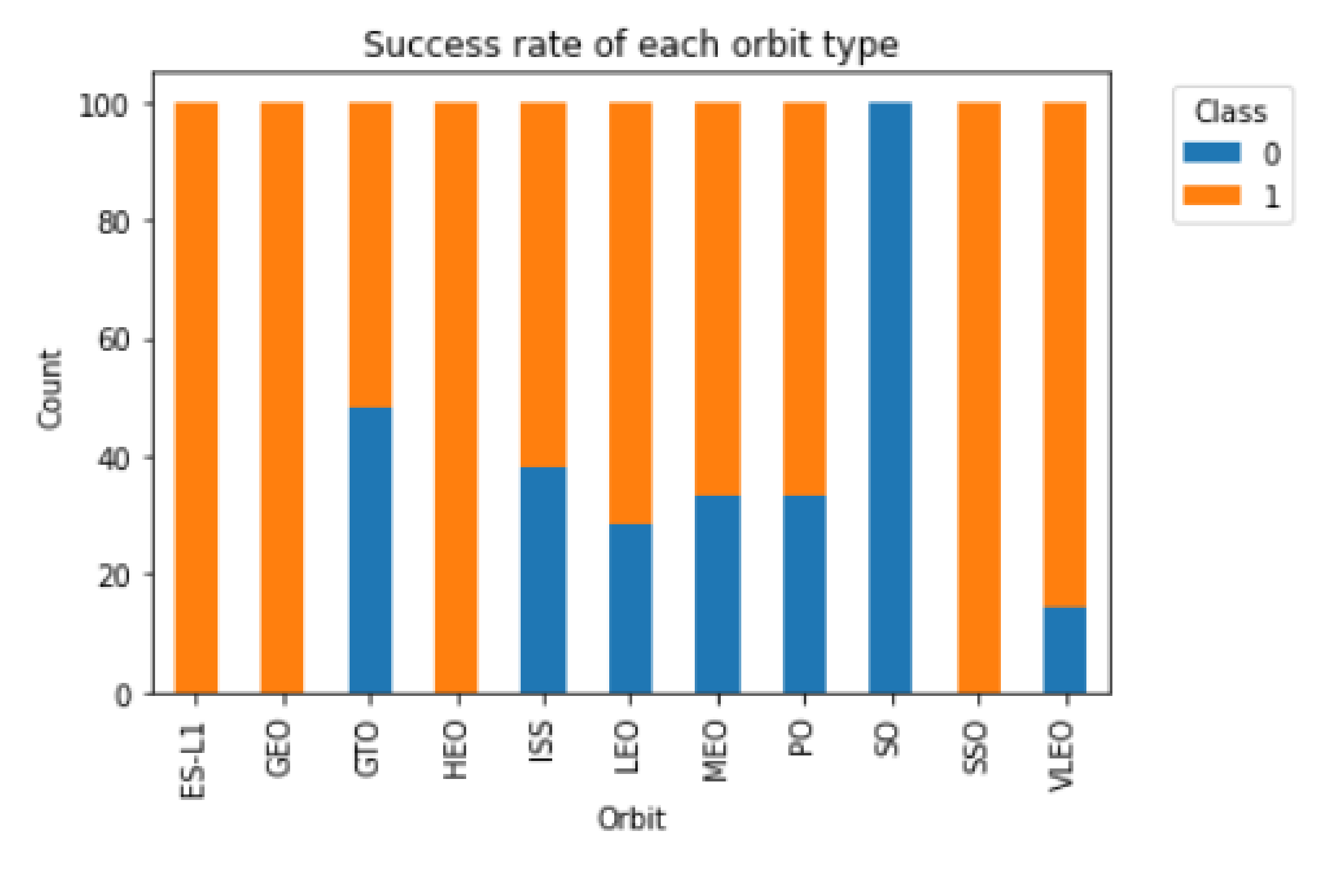
Payload vs. Launch Site



Main Conclusions:

- Payloads over 9,000kg have excellent success rate;
- Consequently, lower payloads have more chances of being a failure;
- Payloads over 12,000kg seems to be possible only on CCAFS SLC 40 and KSC LC 39A launch sites.

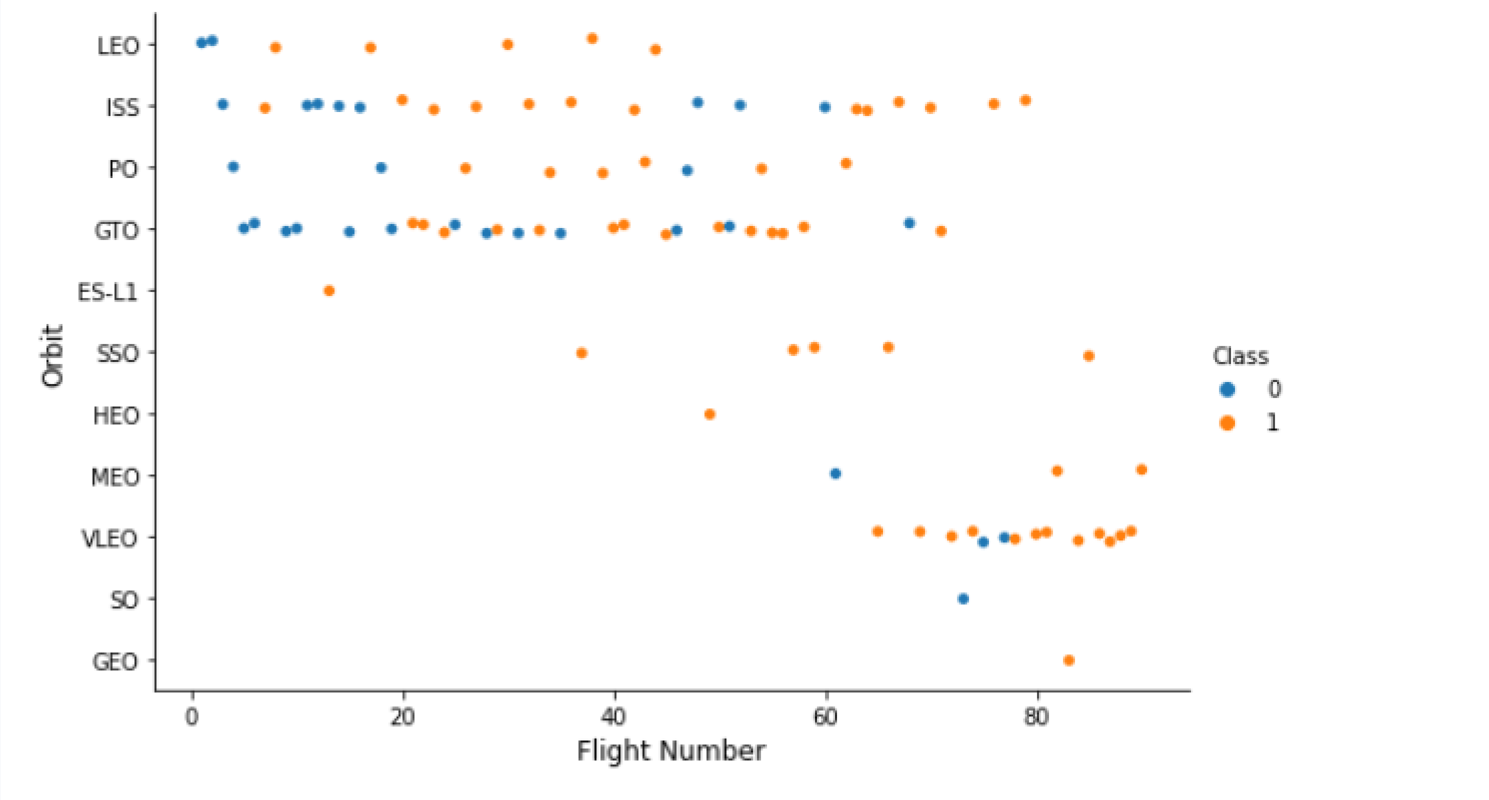
Success Rate vs. Orbit Type



Main Conclusions:

- Orbits ES-L1, GEO, HEO, SSO have the highest success rates;
- Orbit SO has not had a single successful outcome

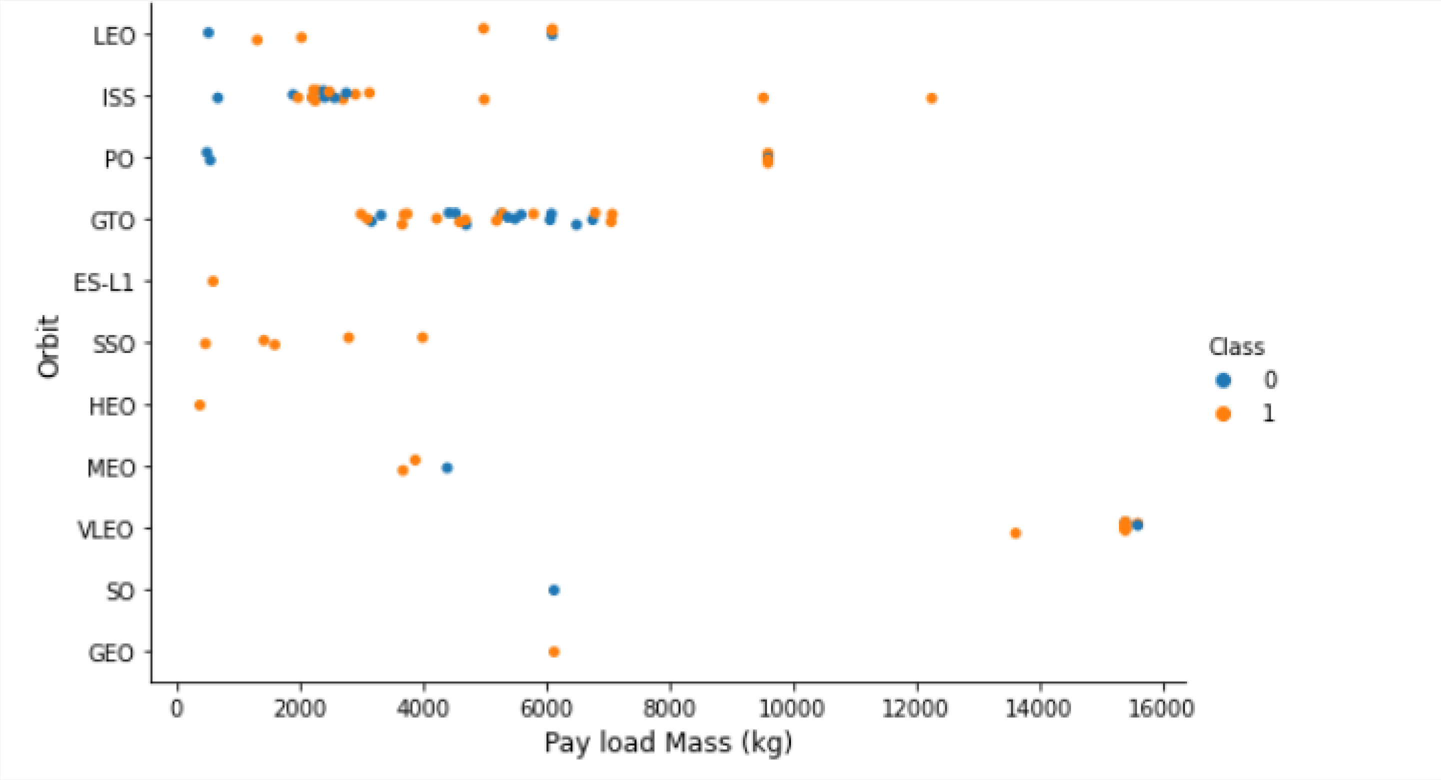
Flight Number vs. Orbit Type



Main Conclusions:

- Apparently, there is no relation between payload and success rate to orbit GTO;
- ISS orbit has the widest range of payload and a good rate of success;
- There are few launches to the orbits SO and GEO.

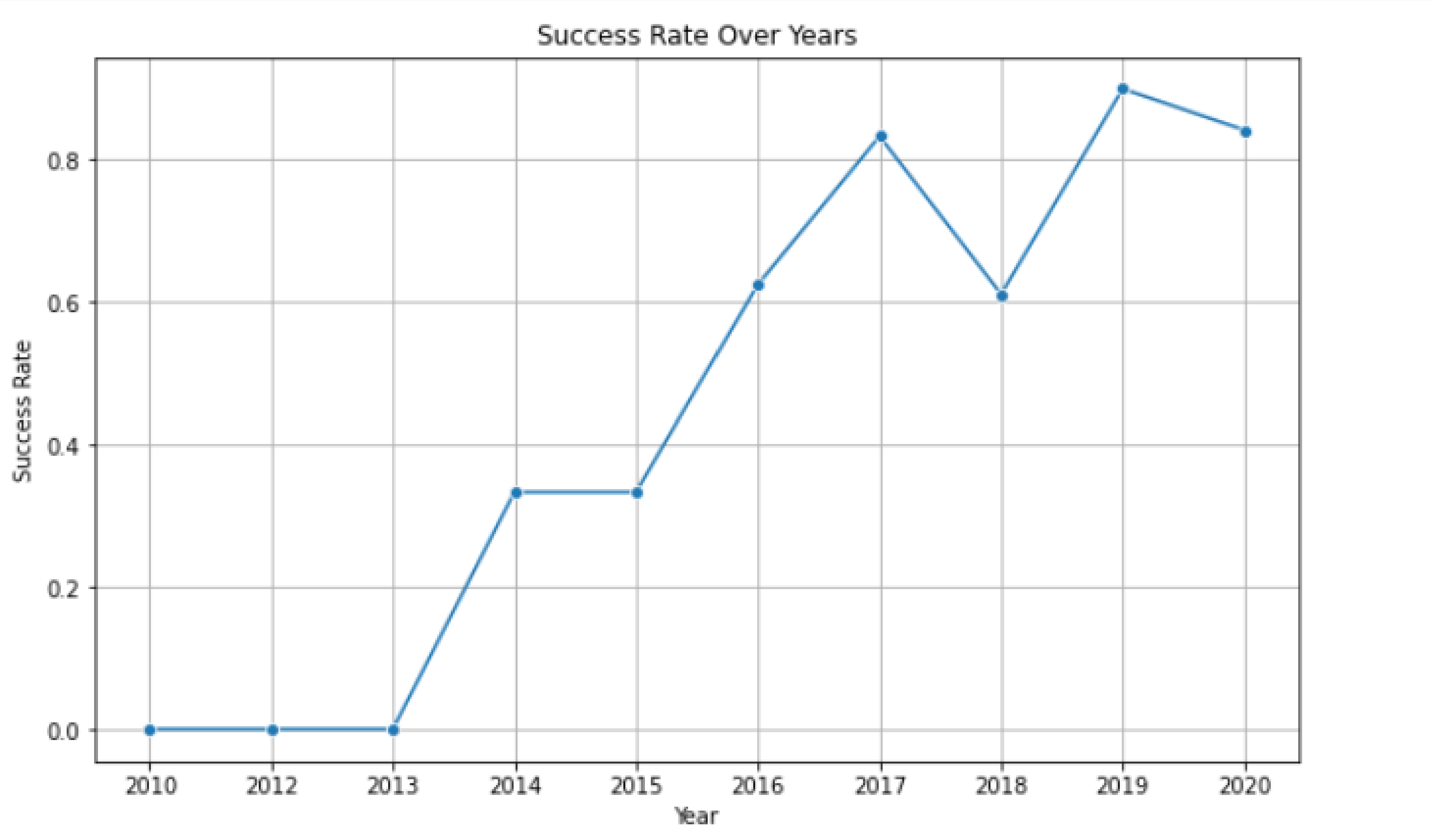
Payload vs. Orbit Type



Main Conclusions:

- Apparently, there is no relation between payload and success rate to orbit GTO;
- ISS orbit has the widest range of payload and a good rate of success;
- There are few launches to the orbits SO and GEO.

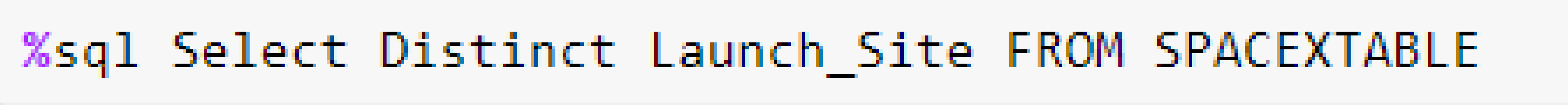
Launch Success Yearly Trend



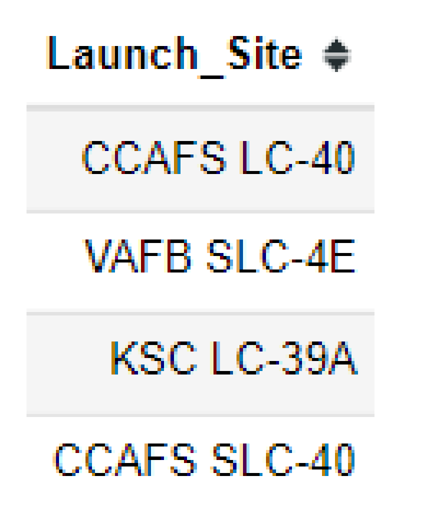
Main Conclusions:

- The success rate was practically zero for almost 4 years.
- After that it has been steadily increasing with the peek observed in the year 2019.

All Launch Site Names



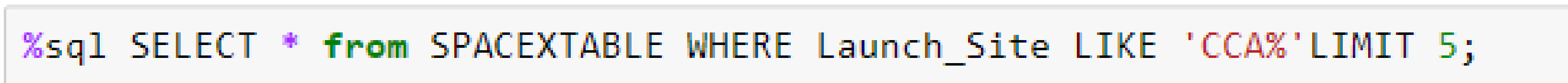
```
%sql Select Distinct Launch_Site FROM SPACEXTABLE
```



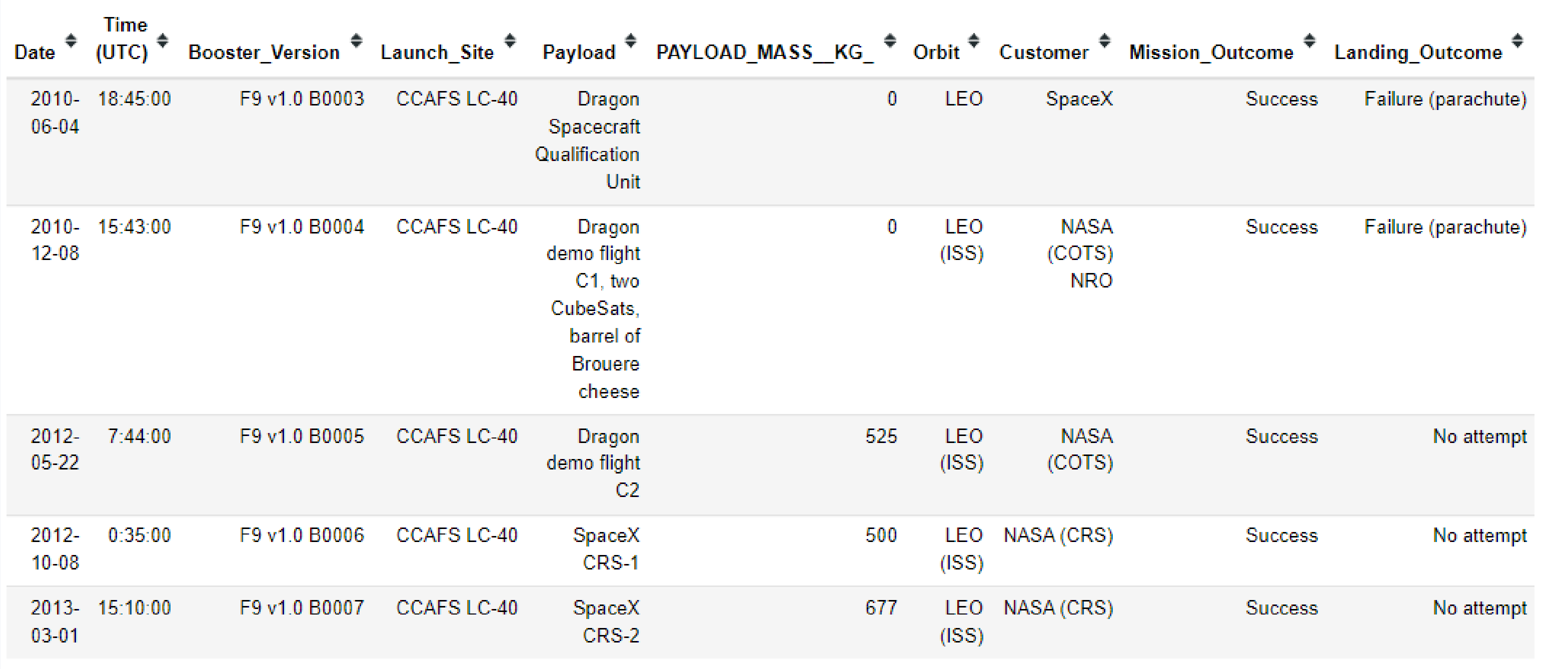
Launch_Site
CCAFS LC-40
VAFB SLC-4E
KSC LC-39A
CCAFS SLC-40

Using the word DISTINCT in the query means that it will only show Unique values in the Launch_Site column from SPACEXTABLE

Launch Site Names Begin with 'CCA'



```
%sql SELECT * from SPACEXTABLE WHERE Launch_Site LIKE 'CCA%'LIMIT 5;
```

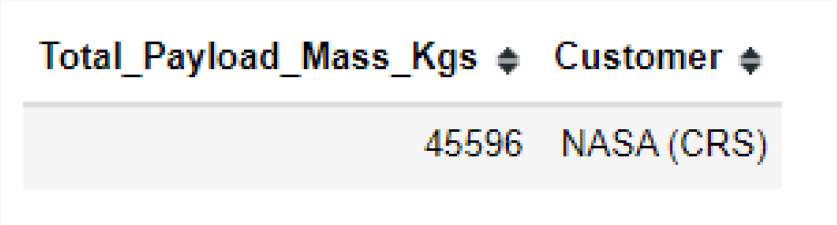


Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
2010-12-08	15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight 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

The % is what allows for matching for all strings containing CCA, regardless of whatever it follows.

Total Payload Mass

```
%sql SELECT SUM(PAYLOAD_MASS__KG_) as Total_Payload_Mass_Kgs, Customer FROM SPACEXTABLE
WHERE Customer = 'NASA (CRS)' ORDER BY CUSTOMER
```

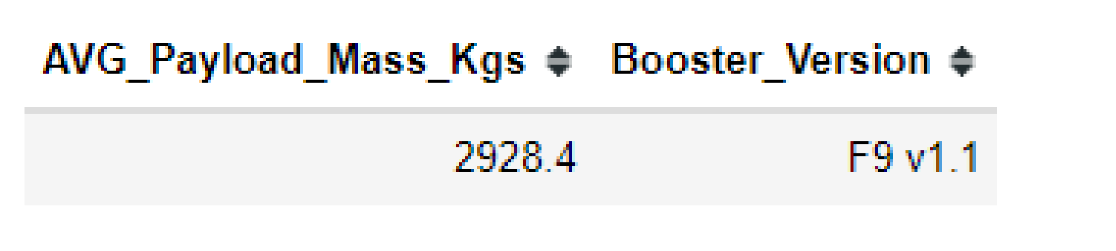


Total_Payload_Mass_Kgs ⬍	Customer ⬍
45596	NASA (CRS)

SUM() provides the cumulative addition of Payload_Mass_Kg_ for NASA (CRS) customers and displays it has Total_Payload_Mass

Average Payload Mass by F9 v1.1

```
%sql SELECT AVG(PAYLOAD_MASS__KG_) as AVG_Payload_Mass_Kgs, Booster_Version FROM SPACEXTABLE
WHERE Booster_Version ='F9 v1.1' ORDER BY Booster_Version
```

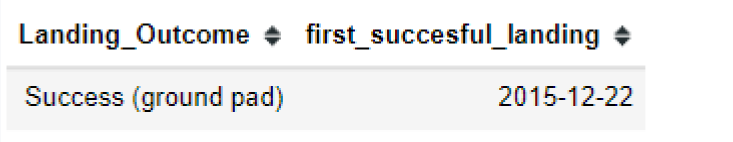


AVG_Payload_Mass_Kgs	Booster_Version
2928.4	F9 v1.1

This SQL query calculates the average payload mass for the booster version 'F9 v1.1' in the SPACEXTABLE dataset.

First Successful Ground Landing Date

```
%sql SELECT Landing_Outcome, MIN(DATE) AS first_succesful_landing FROM SPACEXTABLE WHERE
Landing_Outcome='Success (ground pad)' GROUP BY Landing_Outcome
```

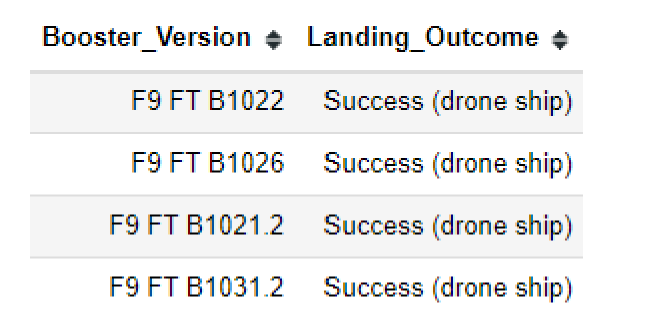


Landing_Outcome ⬍	first_succesful_landing ⬍
Success (ground pad)	2015-12-22

Selects the oldest year from the table when a landing was successful on the ground pad.

Successful Drone Ship Landing with Payload between 4000 and 6000

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

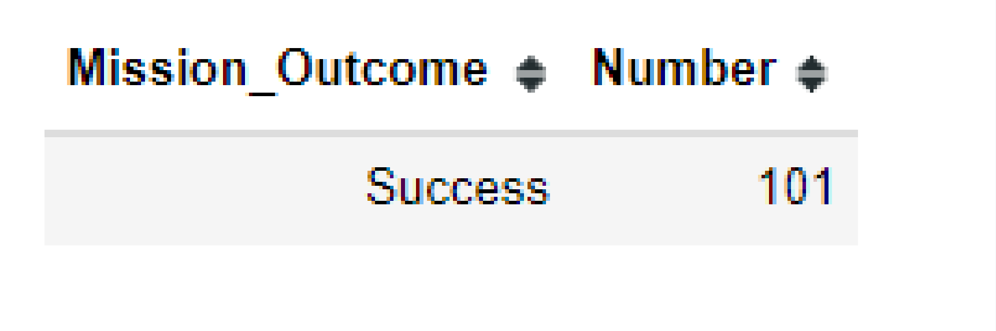


Booster_Version	Landing_Outcome
F9 FT B1022	Success (drone ship)
F9 FT B1026	Success (drone ship)
F9 FT B1021.2	Success (drone ship)
F9 FT B1031.2	Success (drone ship)

Selects the booster version from the table for which landing has been successfully done on drone ship and the payload mass was been 4000 and 6000.

Total Number of Successful and Failure Mission Outcomes

```
%sql SELECT Mission_Outcome, COUNT(Mission_Outcome) AS Number FROM SPACEXTABLE ORDER BY
Mission_Outcome
```

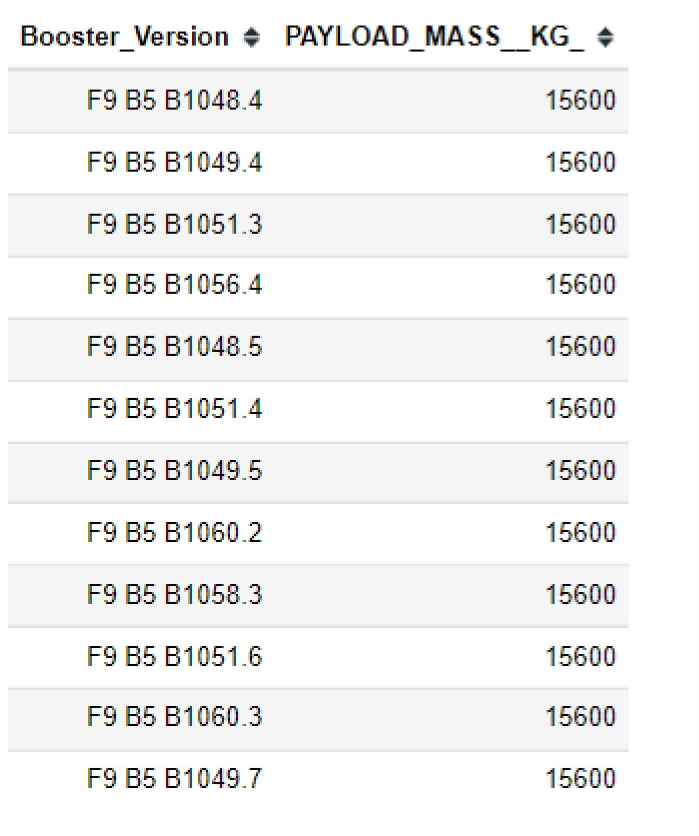


Mission_Outcome	Number
Success	101

Query returns counts of mission outcomes in for the launches grouped by success and failure.

Boosters Carried Maximum Payload

%sql SELECT Booster_Version, PAYLOAD_MASS__KG_ FROM SPACEXTABLE WHERE PAYLOAD_MASS__KG_ IN (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTABLE)

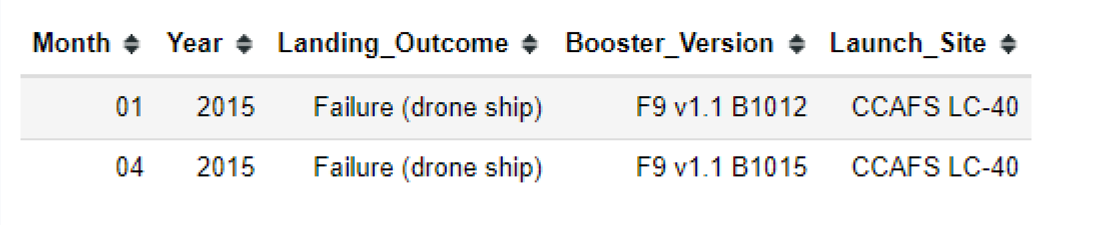


Booster_Version	PAYLOAD_MASS__KG_
F9 B5 B1048.4	15600
F9 B5 B1049.4	15600
F9 B5 B1051.3	15600
F9 B5 B1056.4	15600
F9 B5 B1048.5	15600
F9 B5 B1051.4	15600
F9 B5 B1049.5	15600
F9 B5 B1060.2	15600
F9 B5 B1058.3	15600
F9 B5 B1051.6	15600
F9 B5 B1060.3	15600
F9 B5 B1049.7	15600

A subquery is used to select boosters with max payloads.

2015 Launch Records

%sql SELECT substr(Date, 6,2) as Month, substr(Date,0,5) as Year, Landing_Outcome, Booster_Version, Launch_site FROM SPACEXTABLE WHERE Landing_Outcome='Failure (drone ship)' and substr(Date,0,5)='2015'

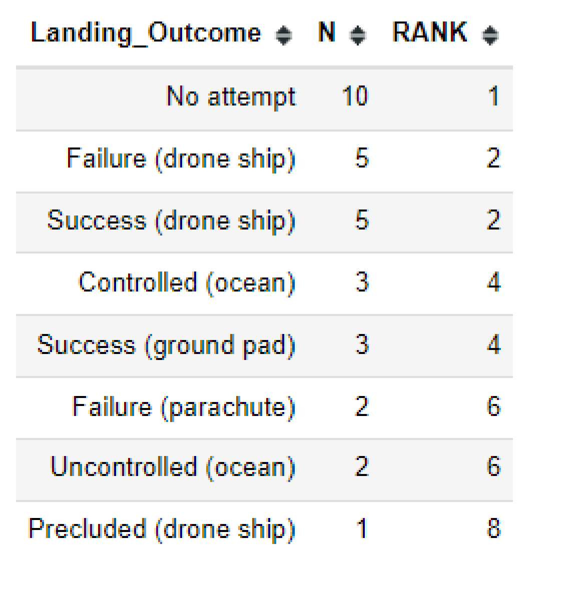


Month	Year	Landing_Outcome	Booster_Version	Launch_Site
01	2015	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
04	2015	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

This query returned month, booster version, launch site, and additionally, Landing outcome.

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

```
%sql SELECT Landing_Outcome, COUNT(Landing_Outcome) AS N, RANK() OVER ( ORDER BY
COUNT(Landing_Outcome)DESC ) as RANK FROM SPACEXTBL WHERE (Date BETWEEN '2010-06-04' AND '2017-03-20')
GROUP BY Landing_Outcome
```



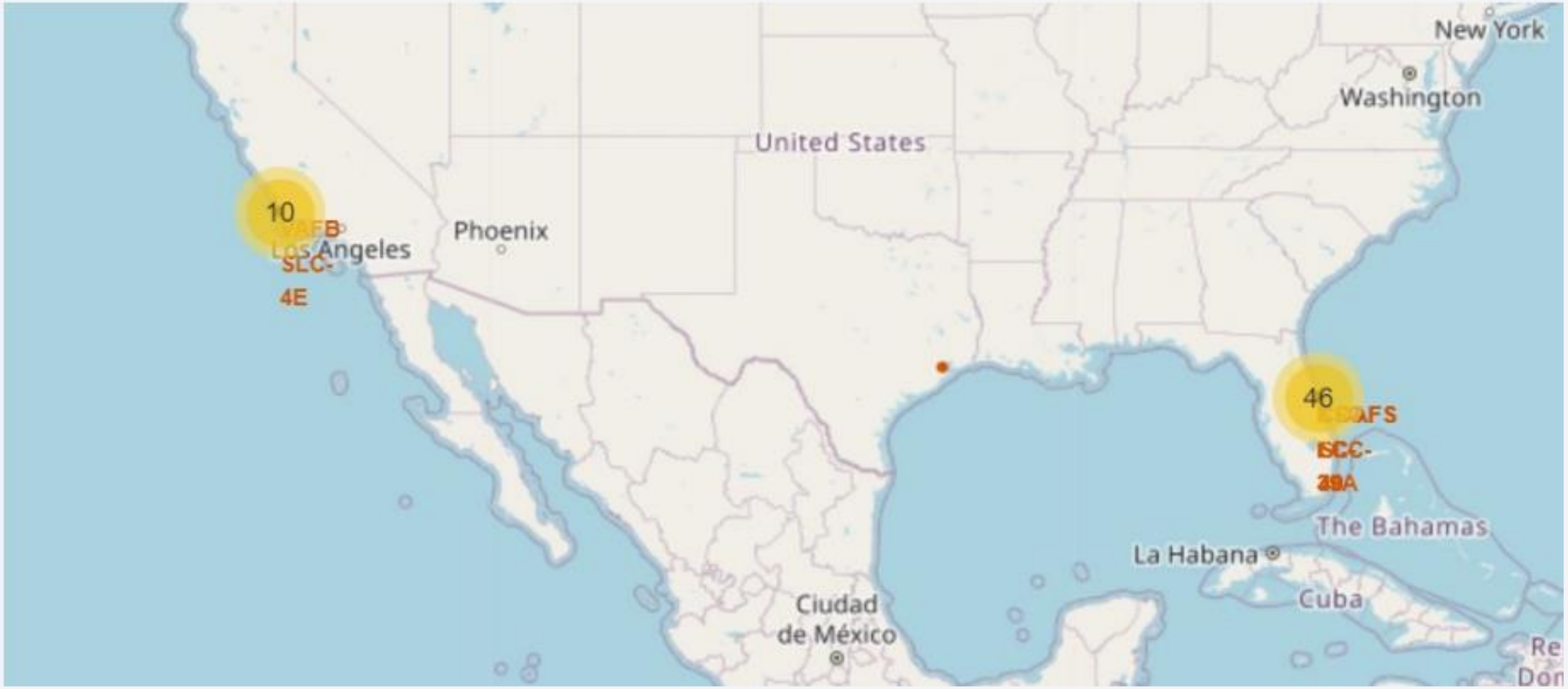
Landing_Outcome ⬍	N ⬍	RANK ⬍
No attempt	10	1
Failure (drone ship)	5	2
Success (drone ship)	5	2
Controlled (ocean)	3	4
Success (ground pad)	3	4
Failure (parachute)	2	6
Uncontrolled (ocean)	2	6
Precluded (drone ship)	1	8

Selects Landing outcomes as distinct values and ranks them according to their magnitude.

Section 3

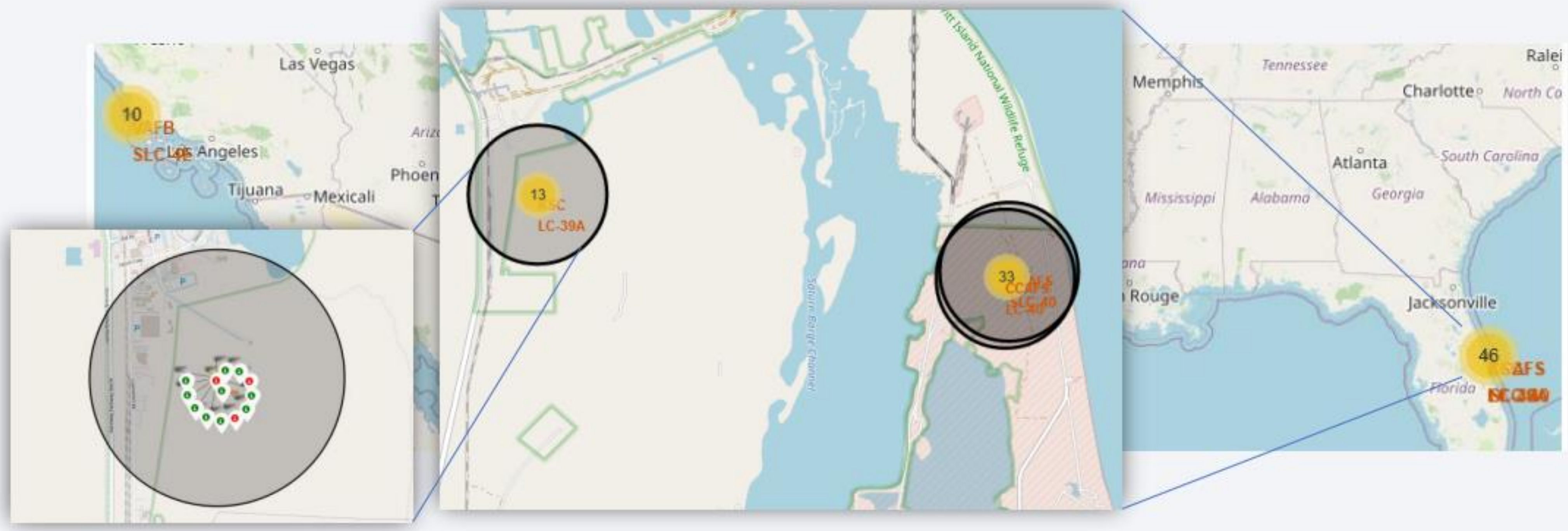
Launch Sites Proximities Analysis

All Launch Sites



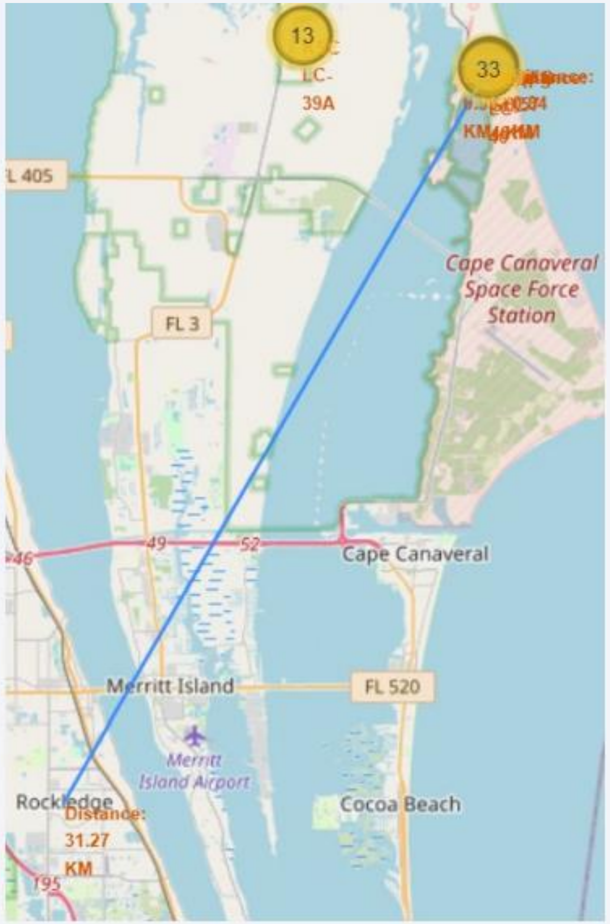
All the launch sites are in USA. Particularly in the states of Florida and California. Launch sites are significantly close to the coasts.

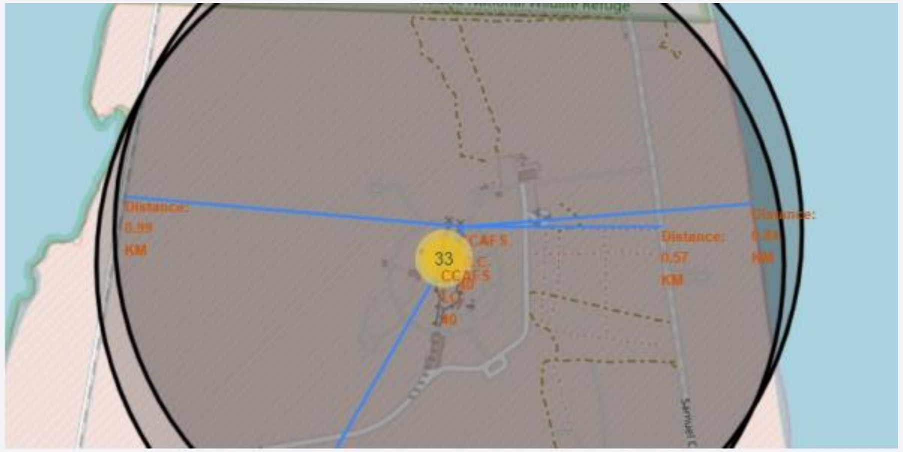
Color Labelled Launch Outcomes



Example of KSC LC-39A launch site launch outcomes.
Green markers indicate successful and red ones indicate failure.

Launch Site Proximity





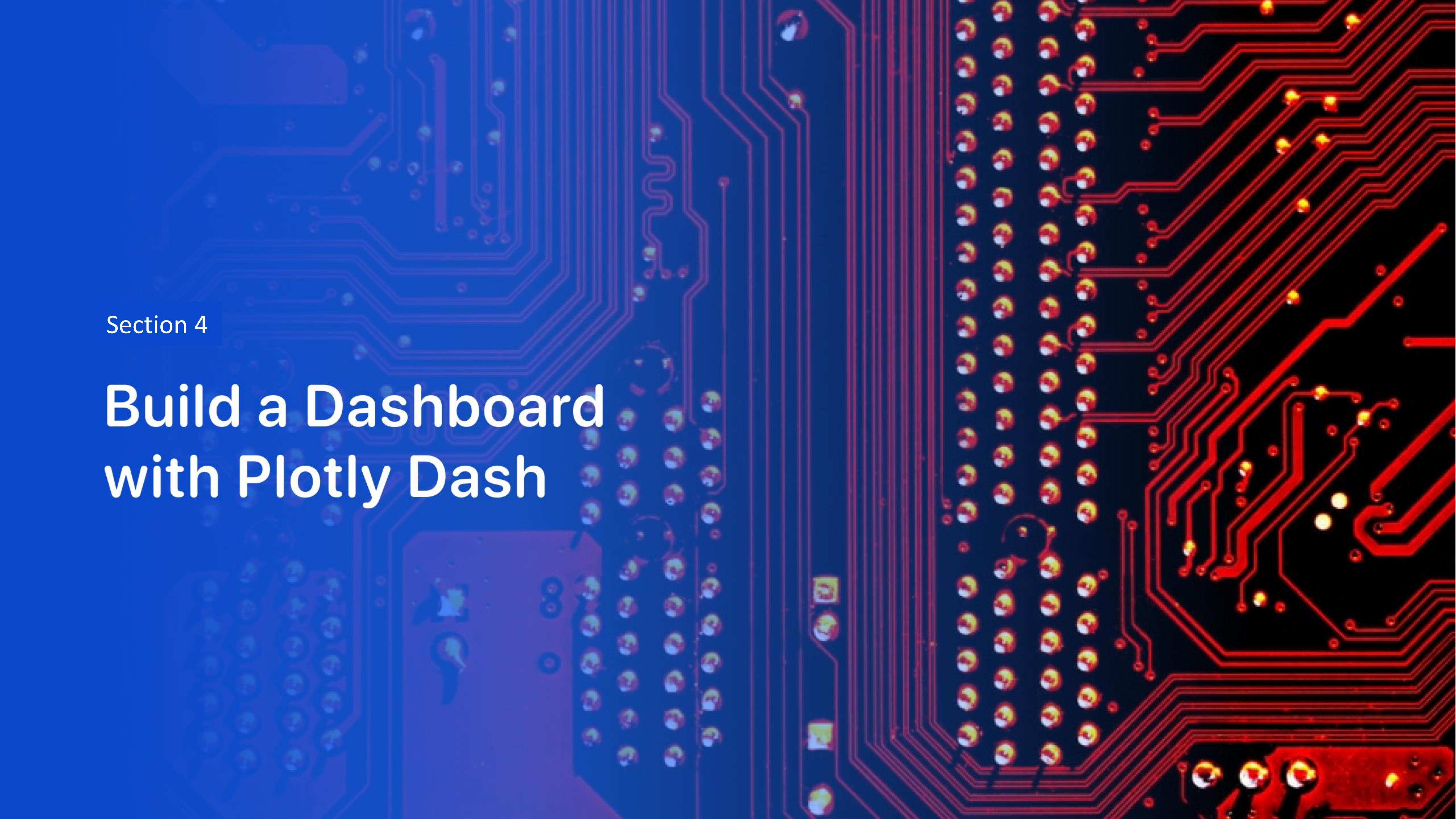
Launch sites are close to railroads and coastlines.

Launch sites aren’t necessarily close to highways. (Contrary to what’s observed in these images, the launch sites In California are not in close proximity with Highways)

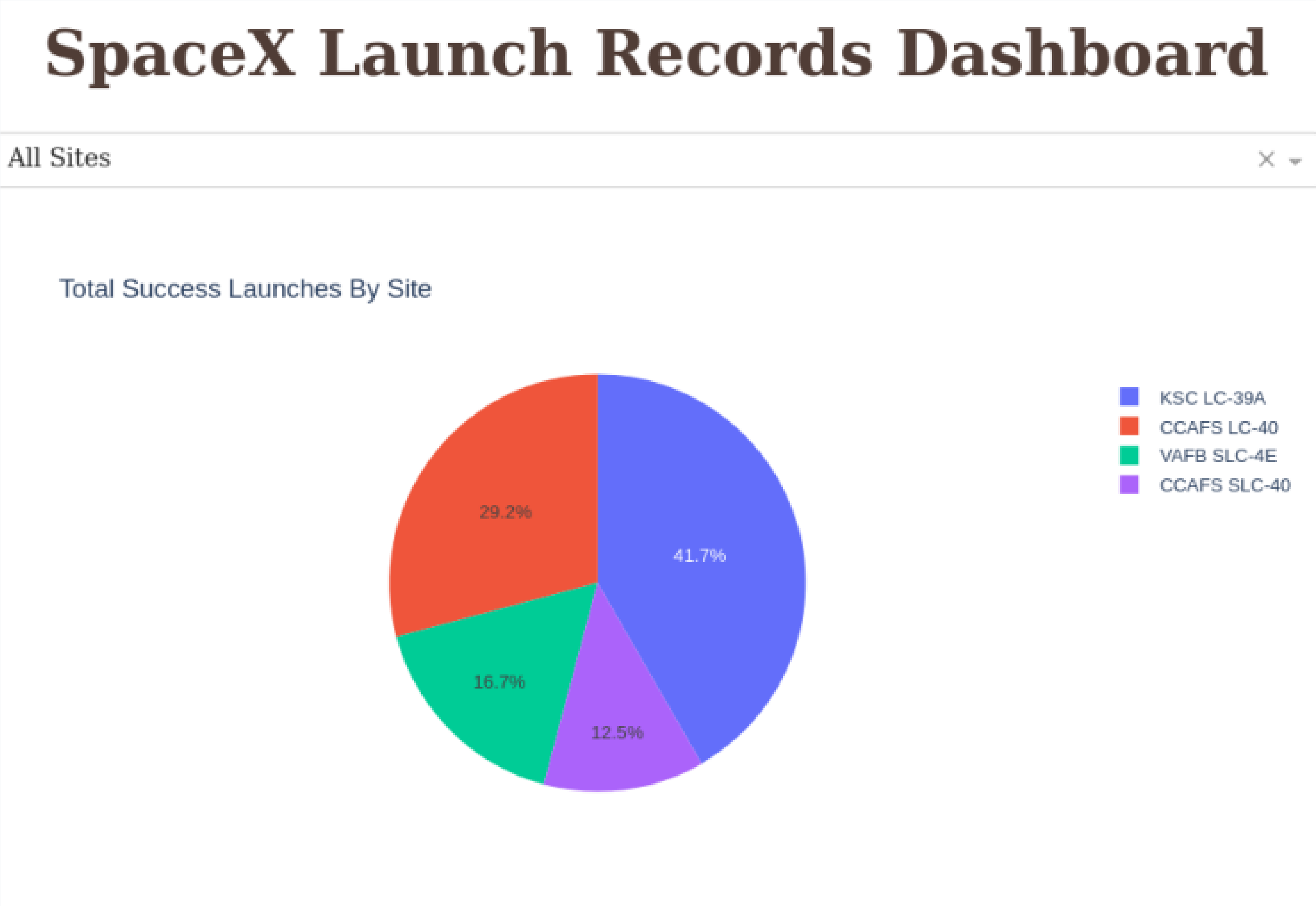
All Launch sites are considerably away from cities.

Section 4

Build a Dashboard with Plotly Dash



Successful Launches by Site

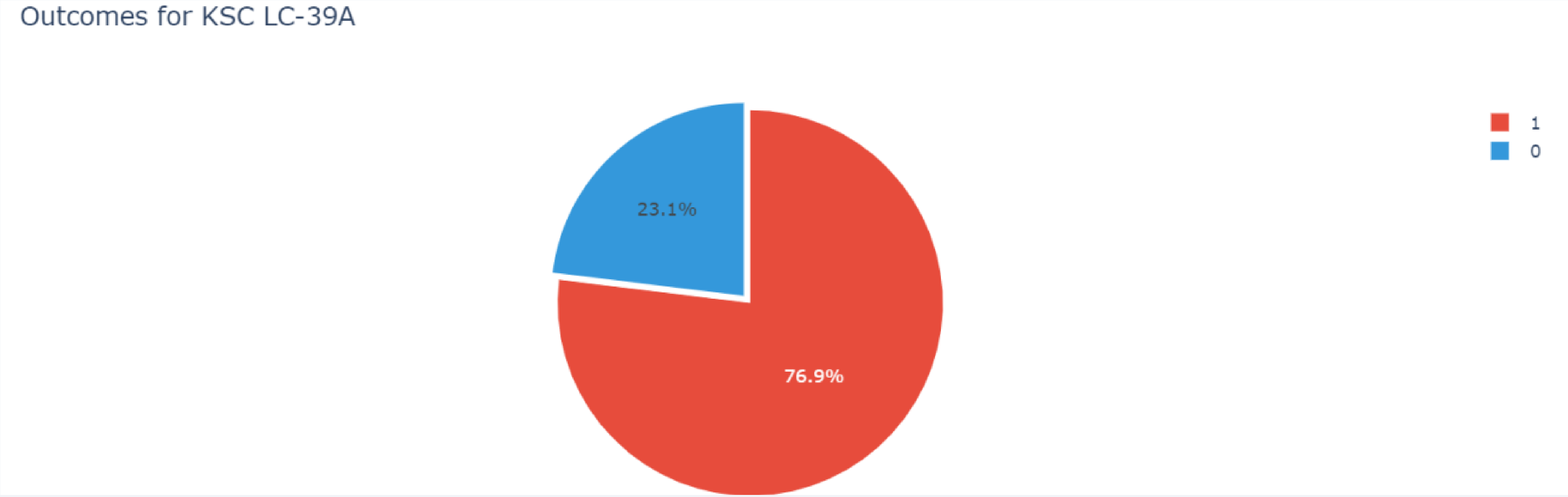


SpaceX Launch Records Dashboard

All Sites

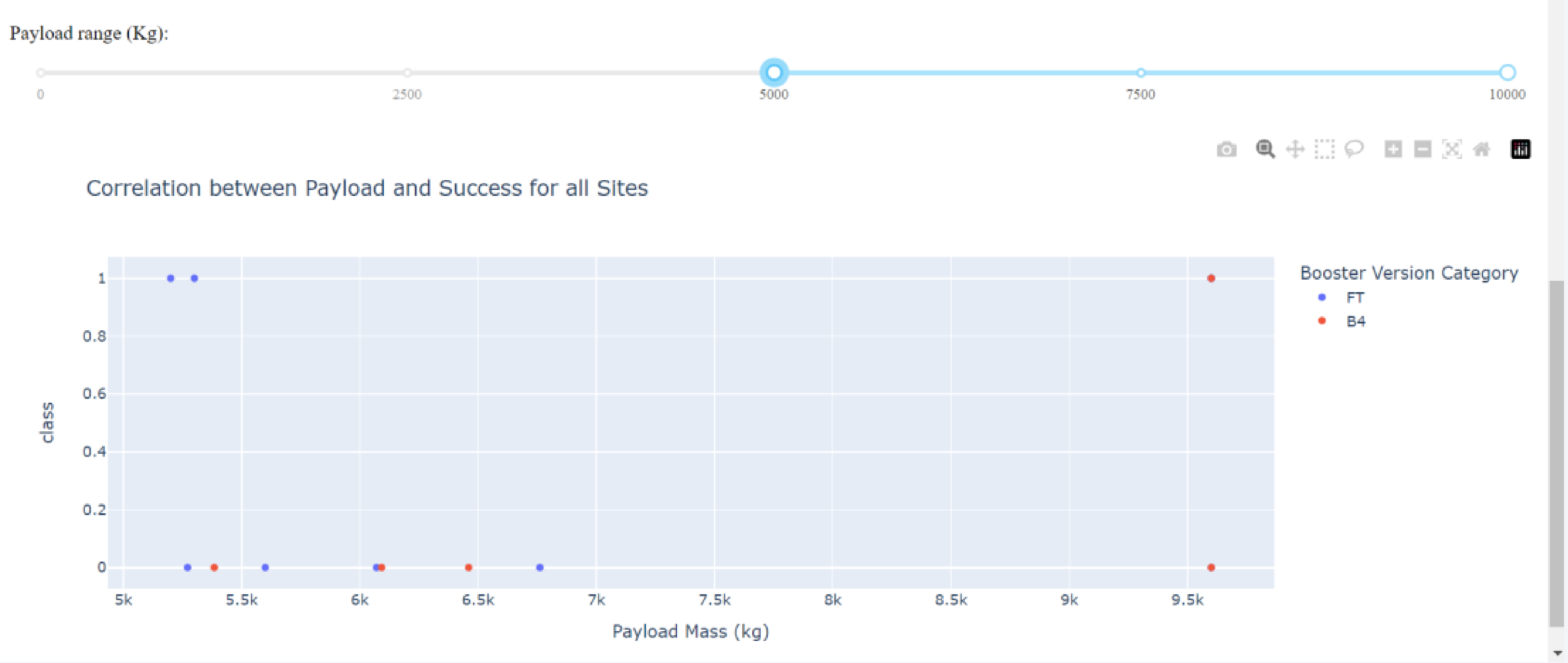
The place from where launches are done seems to be a very important factor of success of missions.

Launch Success Ratio for KSC LC-39A



This site has a success rate of more than 75%.

Payload Mass VS Launch Outcome Scatter Plot

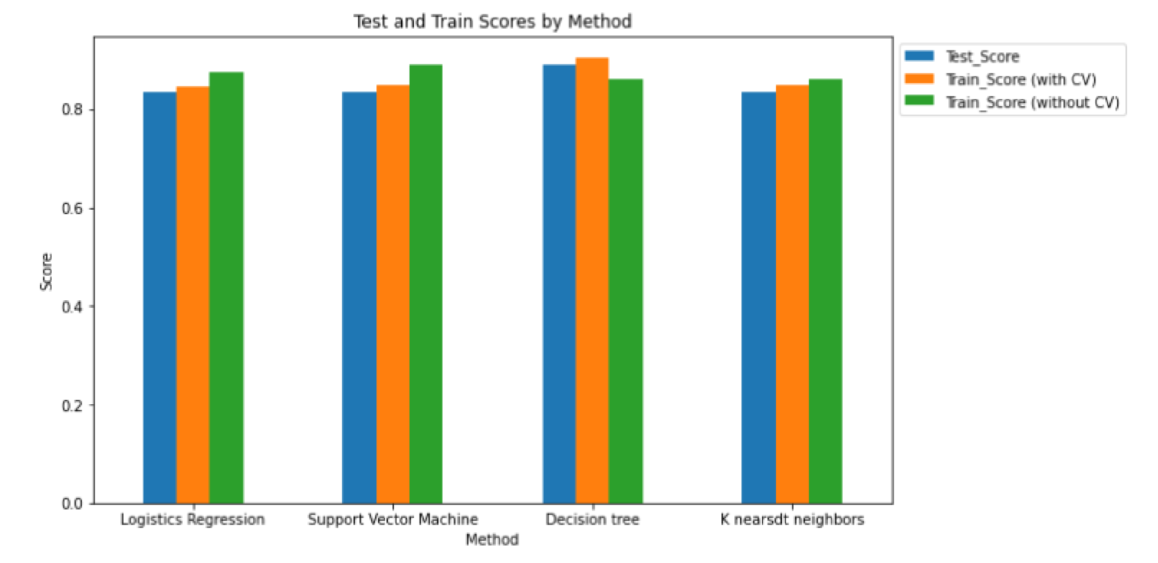


Payloads under 6,000kg and FT boosters are the most successful combination.

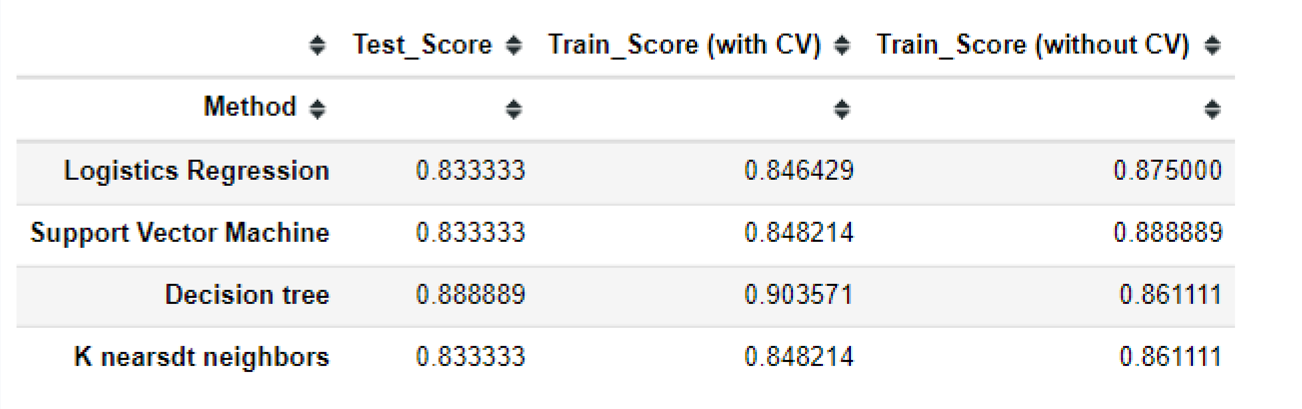
Section 5

Predictive Analysis (Classification)

Classification Accuracy

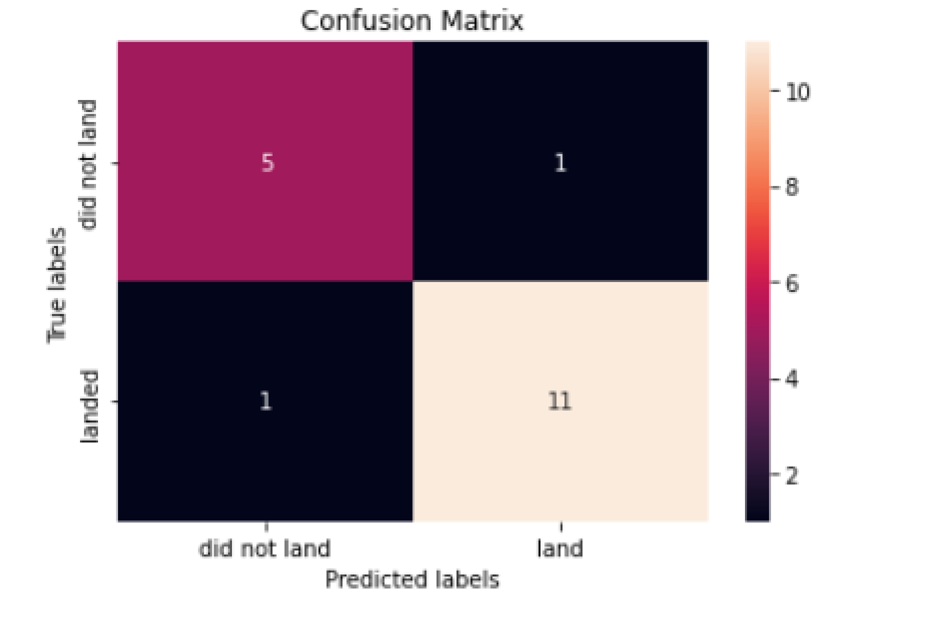


If we use the Test data and the Train data with cross-validation we choose Decision tree method. Using the Train data without cross-validation we will choose Support Vector Machine.



	Test_Score	Train_Score (with CV)	Train_Score (without CV)
Method			
Logistics Regression	0.833333	0.846429	0.875000
Support Vector Machine	0.833333	0.848214	0.888889
Decision tree	0.888889	0.903571	0.861111
K nearsdt neighbors	0.833333	0.848214	0.861111

Confusion Matrix of Decision Tree Classifier



Confusion matrix of Decision Tree Classifier proves its accuracy by showing the big numbers of true positive and true negative compared to the false ones.

Conclusions

- Different data sources were analyzed, refining conclusions along the process;
- The best launch site is KSC LC-39A;
- Launches above 7,000kg are less risky;
- Launch sites are general in close proximity with coastlines, railroads, highways, and are far away from cities;
- Although most of mission outcomes are successful, successful landing outcomes seem to improve over time, according the evolution of processes and rockets;
- Decision Tree Classifier can be used to predict successful landings and increase profits.

Thank you!