IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Sripathi M
18-Jan-2026

Outline

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

Executive Summary

- Summary of methodologies – Comprehensive data science tools to predict success of first-stage rocket landings
  - Working with data – data collection, data wrangling, exploratory data analysis (EDA)
  - Visualization of data – interactive visual analytics and dashboard
  - Predictive modeling – develop and fine tune classification models
  - Key Python libraries – Numpy, Pandas, Matplotlib, Seaborn, Plotly, Folium, Scikit-learn
- Summary of all results
  - Safety: Geographic analysis confirmed that most launch sites are strategically located near coastlines and away from major population areas
  - Launch Success Trends: Landing success rates have shown a significant upward trend over time
  - Site Performance: KSC LC-39A was identified as the launch site with the highest success rate
  - Orbit & Payload Impact: Specific orbits like ES-L1, GEO, HEO, and SSO achieved a 100% success rate

Introduction

- Project background and context
  - To predict if the Falcon 9 first stage will land successfully
  - Falcon 9 rocket launches cost $62 million, much of the savings is because the first stage ca be reused
  - Therefore determining the first stage landing, we can determine the cost of a launch
- Problems you want to find answers
  - What variables (e.g., payload mass, launch site, orbit type, flight number) most influence the success rate of a landing?
  - Does the success rate of landings improve over the years with experience?
  - Which machine learning algorithm provides the best prediction accuracy for landing outcomes?

Section 1

Methodology

Methodology

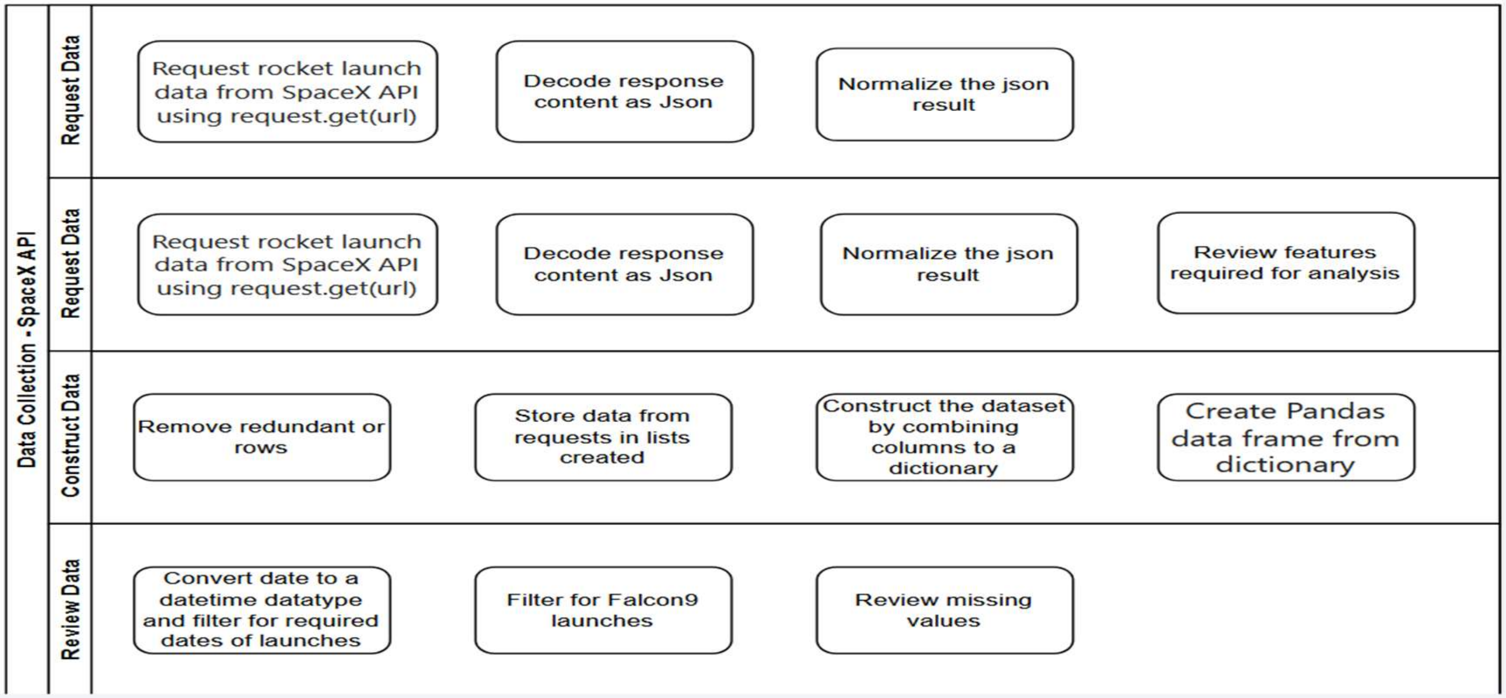
Executive Summary

- Data collection methodology:
    - Data sourced from both primary (SpaceX REST API) and secondary (Wikipedia's list of Falcon 9 launches) sources, ensuring a comprehensive dataset
- Perform data wrangling
    - Handling of missing values, standardization, statistical analysis, data visualization, feature engineering
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
    - Comparison and evaluating accuracy and score among 4 models including tuning using GridSearchCV and visualizing with confusion matrices

Data Collection

- Source of data collection to ensure a comprehensive dataset
  - Data sourced from primary (original) sources SpaceX REST API using HTTP requests, from which launch data was programmatically extracted
  - A secondary source, Wikipedia's list of Falcon 9 launches, was web scraped using Python libraries to gather supplementary details not available via the API
  - Merged both datasets into one single data frame and filtered for Falcon9 launches

Data Collection – SpaceX API



Data Collection - SpaceX API

Request Data

Request rocket launch data from SpaceX API using request.get(url)

Decode response content as Json

Normalize the json result

Request Data

Request rocket launch data from SpaceX API using request.get(url)

Decode response content as Json

Normalize the json result

Review features required for analysis

Construct Data

Remove redundant or rows

Store data from requests in lists created

Construct the dataset by combining columns to a dictionary

Create Pandas data frame from dictionary

Review Data

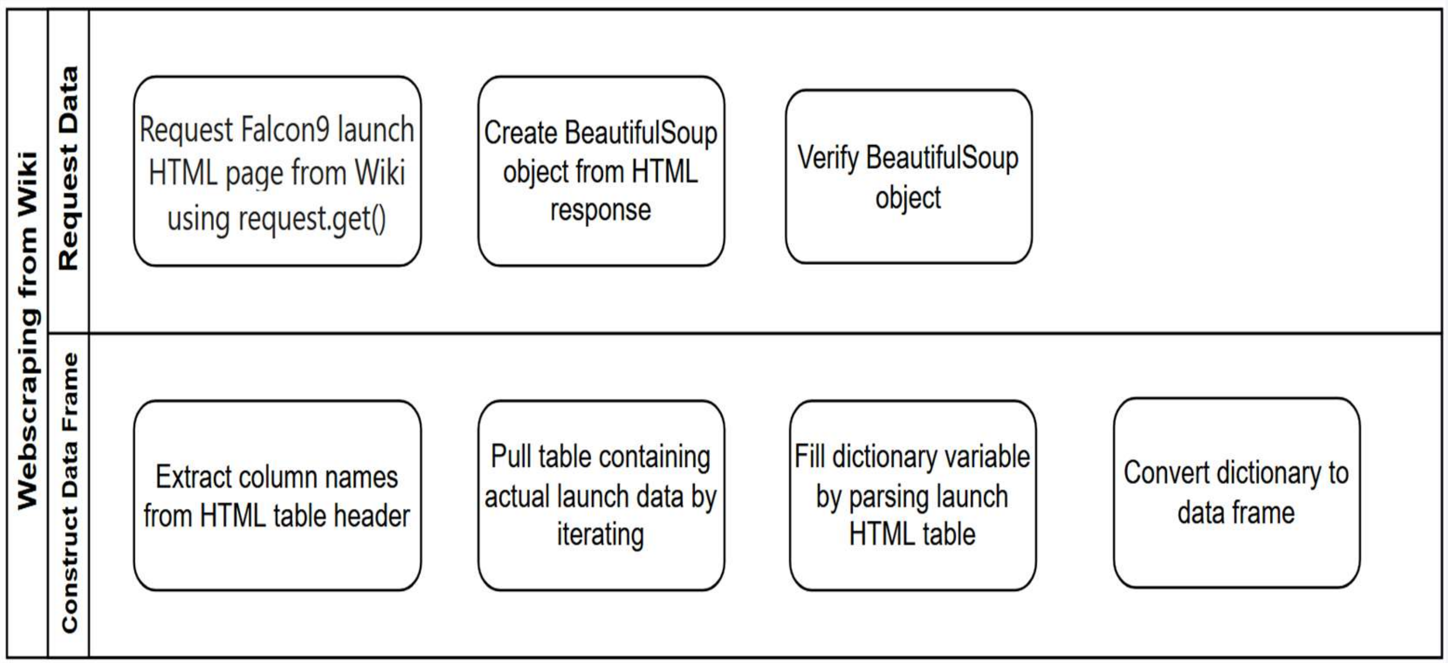
Convert date to a datetime datatype and filter for required dates of launches

Filter for Falcon9 launches

Review missing values

GitHub URL of SpaceX API calls → https://github.com/sripathi747/Capstone-Project/tree/main/Data%20Collection_Web%20Scraping_Wrangling

Data Collection - Scraping



Webscraping from Wiki

Request Data

Request Falcon9 launch HTML page from Wiki using request.get()

Create BeautifulSoup object from HTML response

Verify BeautifulSoup object

Construct Data Frame

Extract column names from HTML table header

Pull table containing actual launch data by iterating

Fill dictionary variable by parsing launch HTML table

Convert dictionary to data frame

GitHub URL of Webscraping → https://github.com/sripathi747/Capstone-Project/tree/main/Data%20Collection_Web%20Scraping_Wrangling

Data Wrangling

- Identified numerical and categorical variables
  - Missing values reviewed for each column
- Defined predicting variable called landing class variable consisting of 0 (failure) or 1 (otherwise)
- Site launches data
  - Reviewed counts of launches from each launch site that is not a geostationary
  - Outcome of launches from each site
  - Success rate based on successful launches
- GitHub URL of data wrangling → https://github.com/sripathi747/Capstone-Project/tree/main/Data%20Collection_Web%20Scraping_Wrangling

EDA with Data Visualization

- Flight Number vs Payload Mass plot revealed that even with higher payload, the first stage returns successfully
- Scatter plot of Flight Number vs Launch Site and Payload Mass vs Launch Site grouped by success or failure class shows that CCAFS SLC 40 site has maximum launches with high success rate, for both high and low payload
- Bar chart of Success rate vs Orbit shows that ES-L1, GEO, HEO and SSO orbits have high average success rates
- Scatterplot of Flight Number vs Orbit reveals 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.
- Scatterplot of Year vs Success Rate shows that success rate has an increasing trend, reflecting learnings from past launches
- GitHub URL of EDA with data visualization → https://github.com/sripathi747/Capstone-Project/tree/main/Exploratory%20Data%20Analysis

EDA with SQL

- Extracting of data using DISTINCT and 5 records from the SPACEXTBL
- Querying for total and average payload mass for a customer and booster version
- Identifying first successful landing outcome in ground pad
- Extracting data of successful and failed mission outcomes
- Identifying boosters with success in drone ship and payload mass between 4000 and 6000
- Querying for booster version with maximum payload mass
- Extracting data for year 2015 that had failed landing outcomes in drone ship
- GitHub URL of EDA with SQL → https://github.com/sripathi747/Capstone-Project/tree/main/Exploratory%20Data%20Analysis

Build an Interactive Map with Folium

- Folium Circles have been added as a circle area with text label to highlight launch site with specific coordinates
- Folium Markers are added as tear drop to identify exactly the launch sites coordinates
- Circles and markers are used to mark the launch outcomes for each site
- To identify each site with outcomes, marker_cluster() is utilized for marking
- Summarize what map objects such as markers, circles, lines, etc. you created and added to a folium map
- Using calculate_distance() between launch site coordinates and coordinates of areas of interest such as coasts, railways, etc to find the distance
- GitHub URL of interactive map with Folium map → https://github.com/sripathi747/Capstone-Project/tree/main/Interactive%20Visual%20Analytics_Dashboard_Prediction

Build a Dashboard with Plotly Dash

- Interactive dashboard built – pie chart for success/failure rate and scatter plot for payload vs success
- Pie chart enables to view success vs failure rate of each site or all sites together through the drop-down feature developed
- Scatter plot helps to see how payload mass effect outcome of success that is grouped by the booster version
    - Slider is added in the interactive plot to select the payload range
    - Scatter plot can be viewed for all sites together or each site separately
- GitHub URL of Plotly Dash → https://github.com/sripathi747/Capstone-Project/tree/main/Interactive%20Visual%20Analytics_Dashboard_Prediction

Predictive Analysis (Classification) – 1

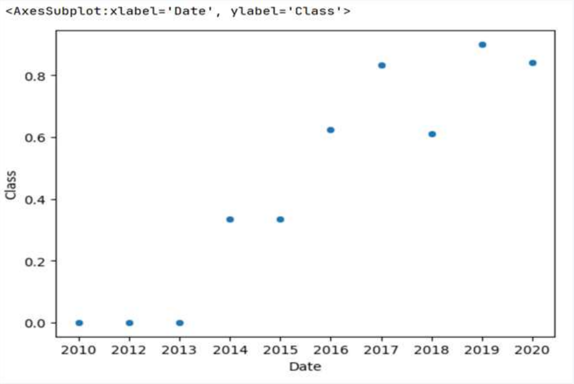
- Feature Engineering
  - Based on the data visualization in EDA, features that will be used for predicting outcome are identified
  - Using get_dummies(), OneHotEncoder is applied to categorical variables (columns)
- Data Preparation
  - Loaded data is segregated into data X for features that are independent variables (features) and data Y for dependent variable (class)
  - Standardize data in X using fit_transform()
  - Split training and test data in the ratio 80:20
- GitHub URL of predictive analysis → https://github.com/sripathi747/Capstone-Project/tree/main/Interactive%20Visual%20Analytics_Dashboard_Prediction

Predictive Analysis (Classification) – 2

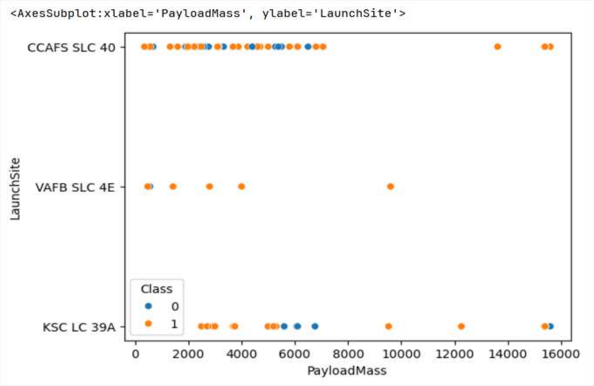
- Models used for prediction – Logistic Regression, Support Vector Machine, Decision Tree Classifier, K Nearest Neighbor
- Hyperparameter tuning using GridSearchCV()
  - Used to automate process of finding the optimal settings for the above ML models to achieve best performance
  - cv = 10 is specified to use 10-fold cross validation where the data is partitioned into 10 equal sized buckets and one-fold is held out as validation set while remaining 9 sets are used for training
  - Across the 10 iterations, cross validation score is calculated based on the average of the 10 scores
- Across all 4 models – scores on test data is found and confusion matrix plotted to identify false positives and true accuracy

Results – EDA

- Exploratory data analysis results
  - Total of 90 Falcon9 launches between 04-Jun-2010 and 05-Nov-2020 and success rate of 67%
  - 41 mission outcomes successfully landed to a drone ship
  - ES-L1, GEO, HEO and SSO orbits have high average success rates
  - CCAFS SLC 40 site has maximum launches with high success rate, for both high and low payload
  - Success rate has an increasing trend, reflecting learnings from past launches

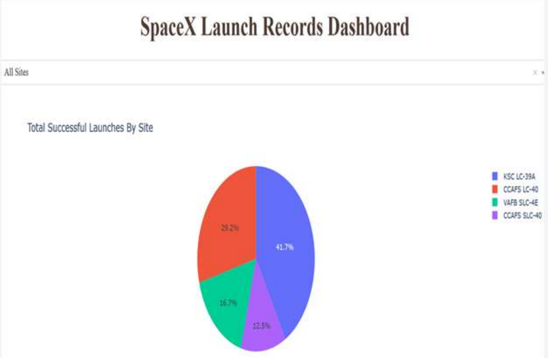


```
<AxesSubplot:xlabel='Date', ylabel='Class'>
```



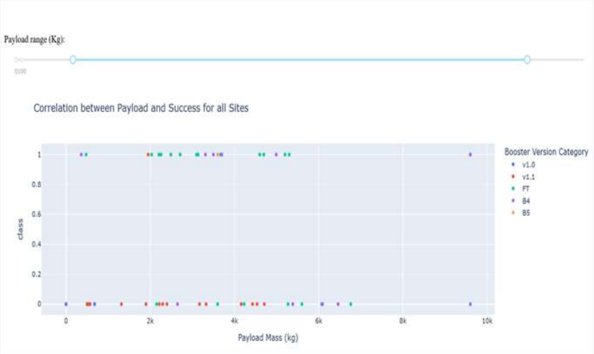
```
<AxesSubplot:xlabel='PayloadMass', ylabel='LaunchSite'>
```

Results – Interactive Analytics

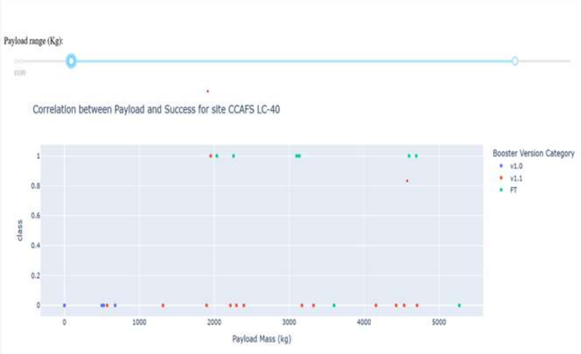


SpaceX Launch Records Dashboard

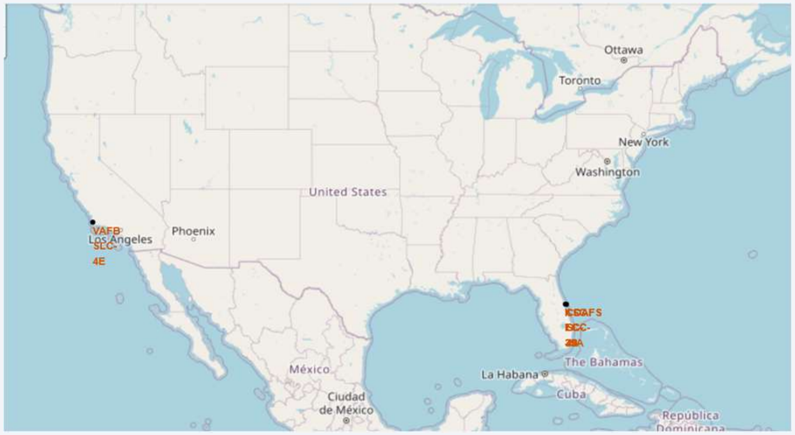
All Sites

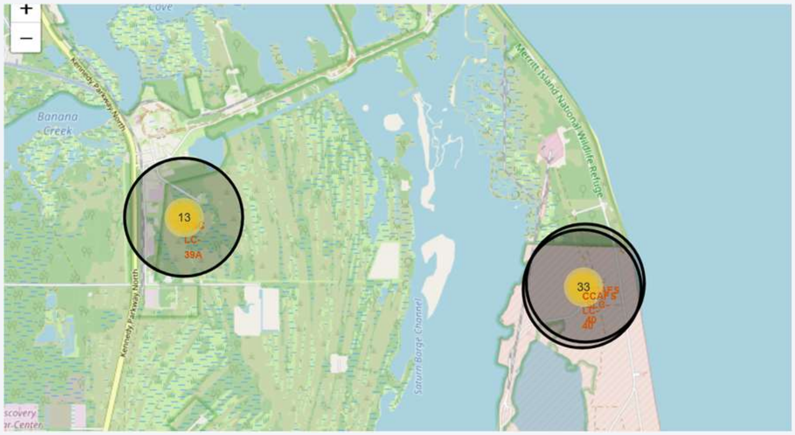


Payload range (Kg):



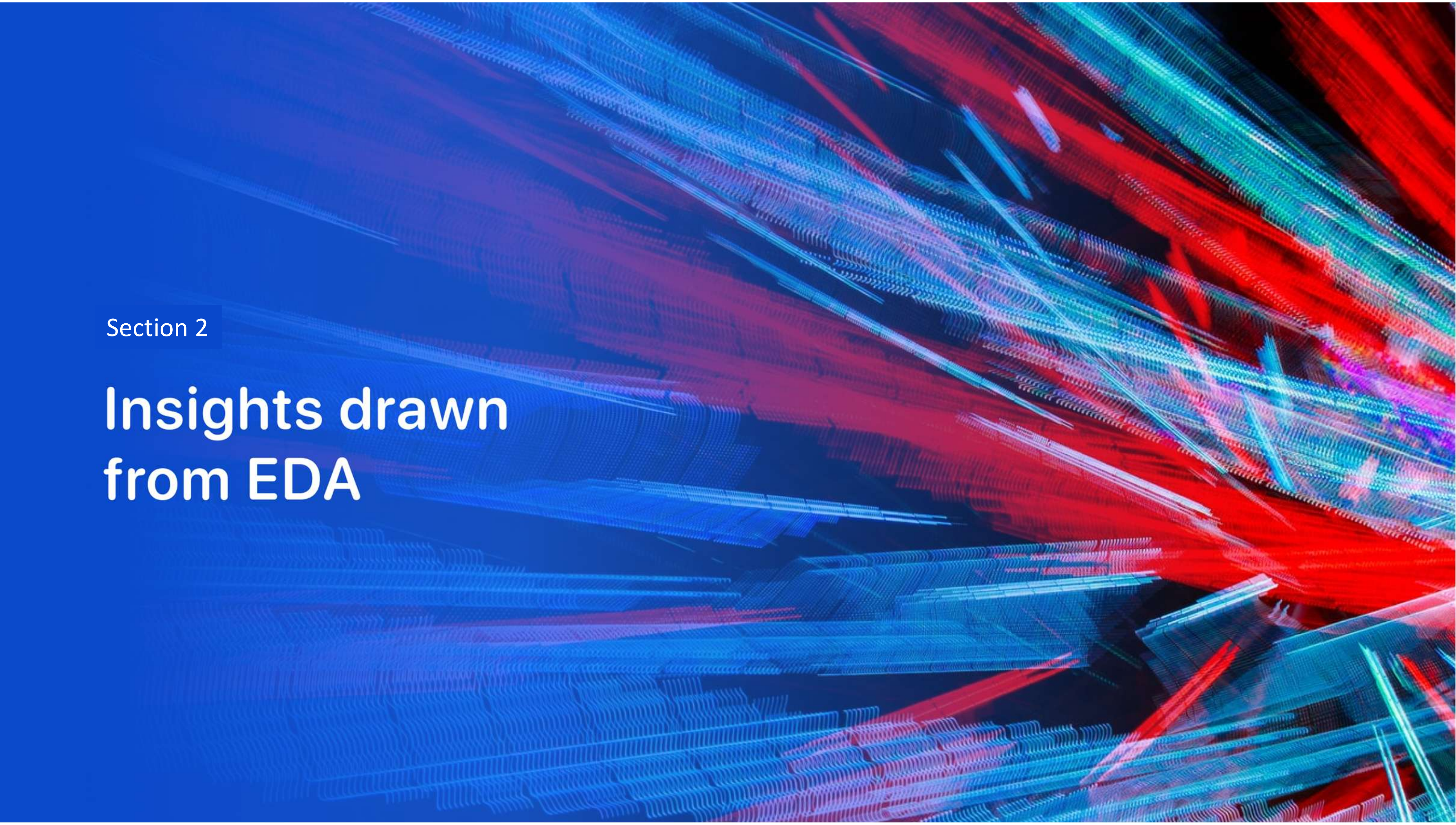
Payload range (Kg):





Results – Predictive Analysis

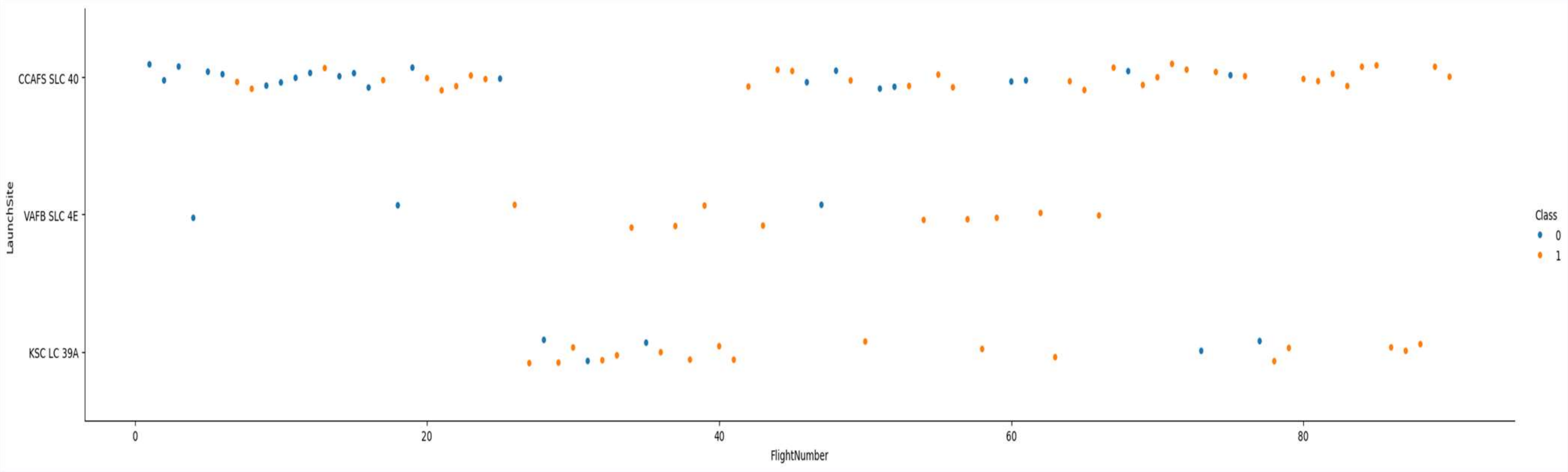
- Predictive analysis results
  - 20% data used for testing
  - Sigmoid kernel has the best result on the validation dataset for Support Vector Machine model
  - Accuracy of all 4 models based on test data = 83.33%
  - Of the 4 models, Decision Tree Classifier performed best for prediction based on confusion matrix



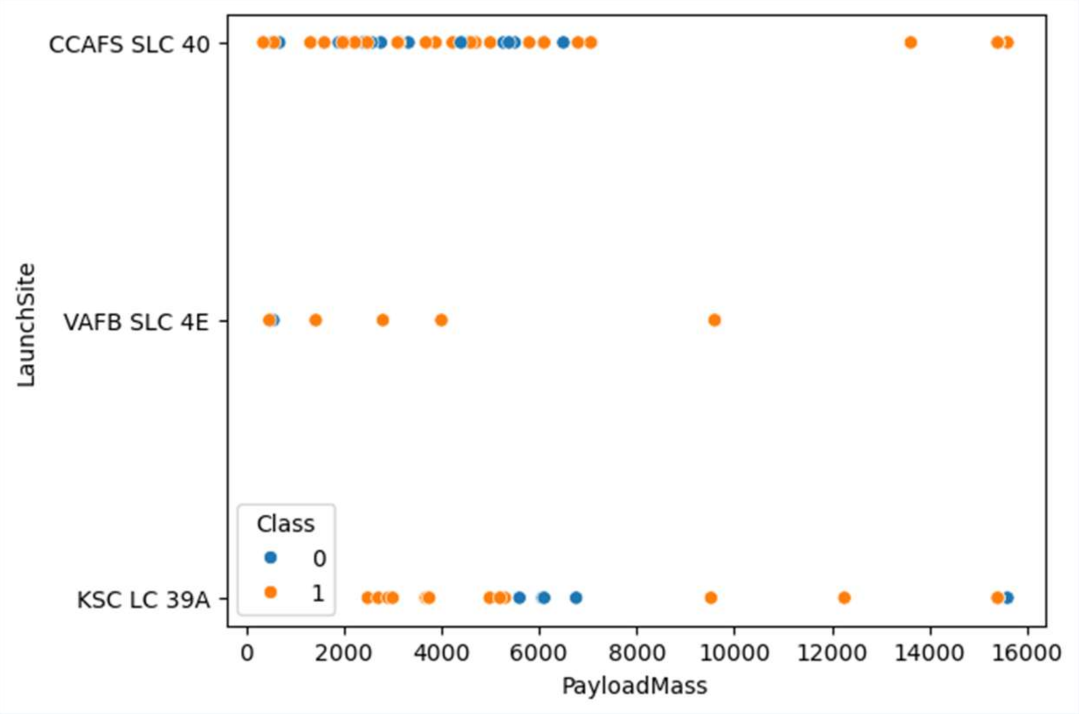
Section 2

Insights drawn from EDA

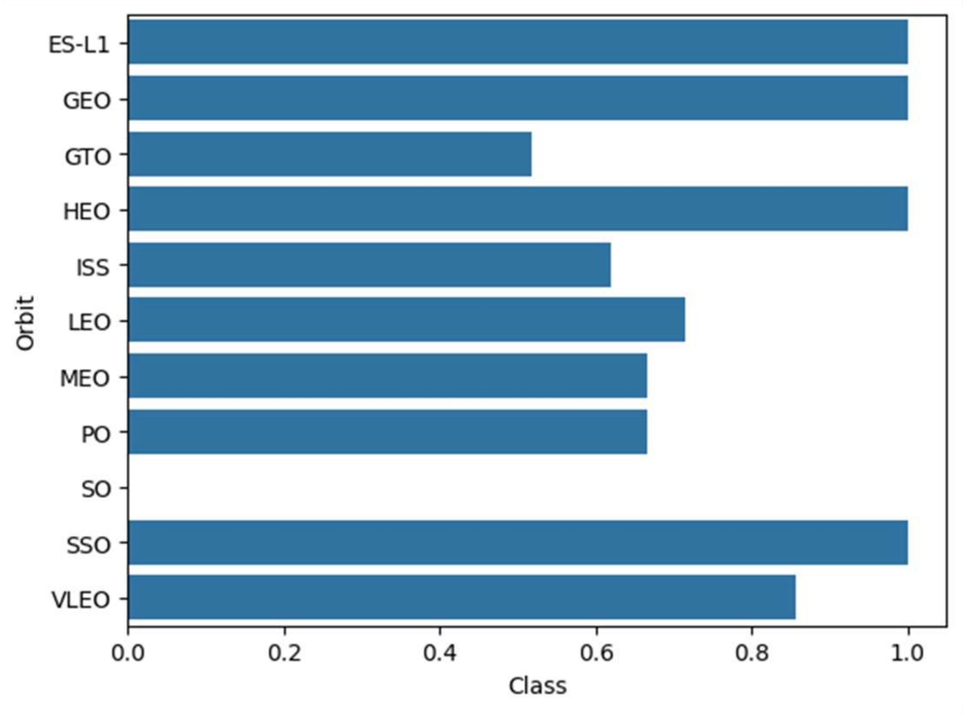
Flight Number vs. Launch Site



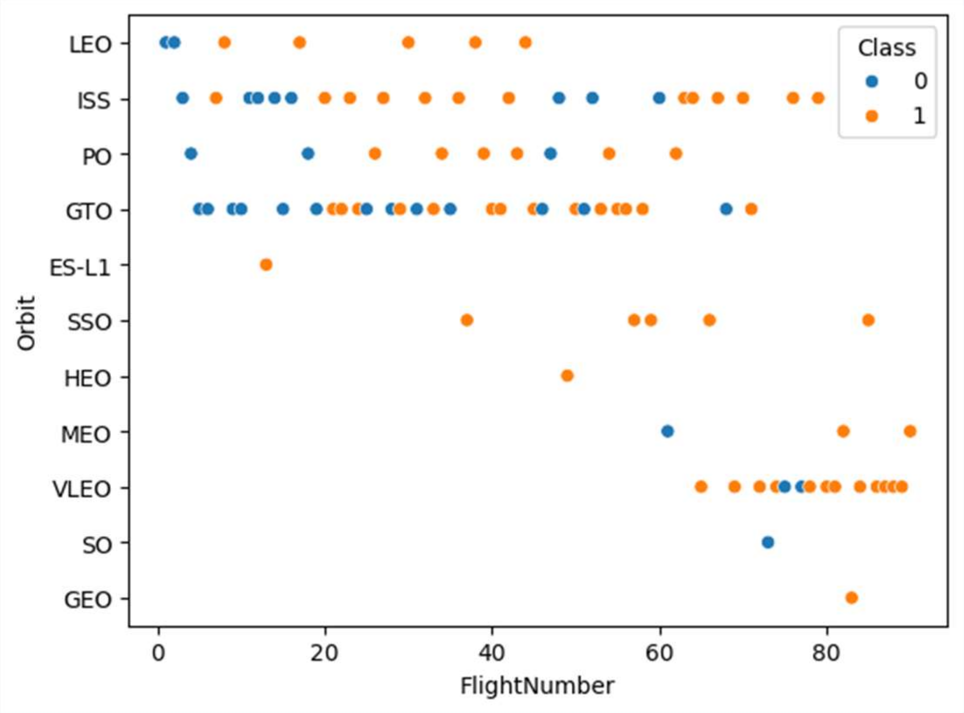
Payload vs. Launch Site



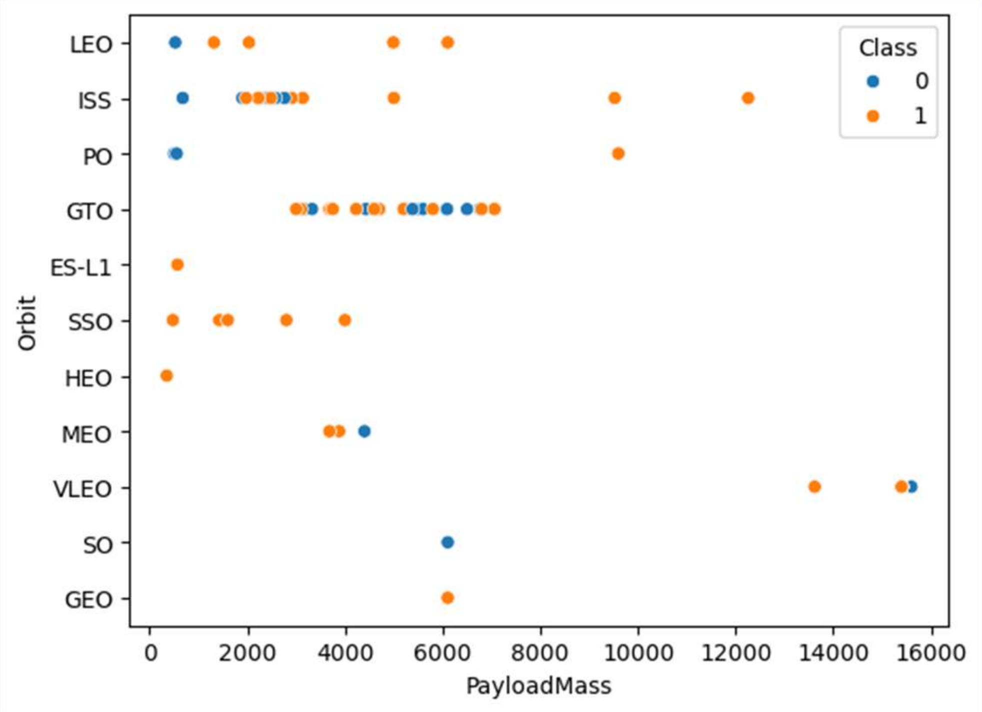
Success Rate vs. Orbit Type



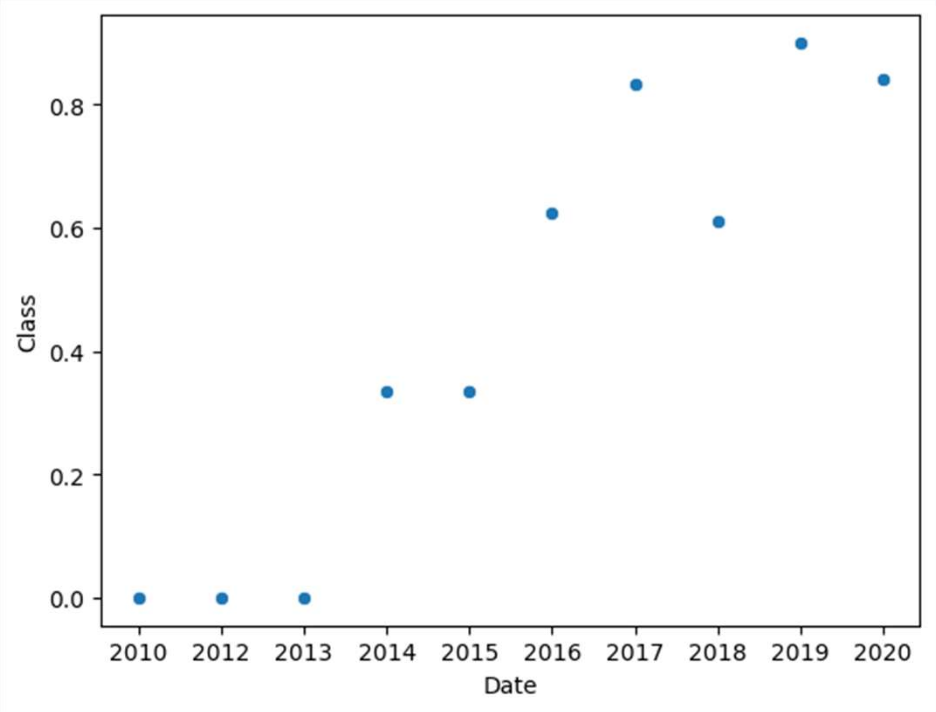
Flight Number vs. Orbit Type



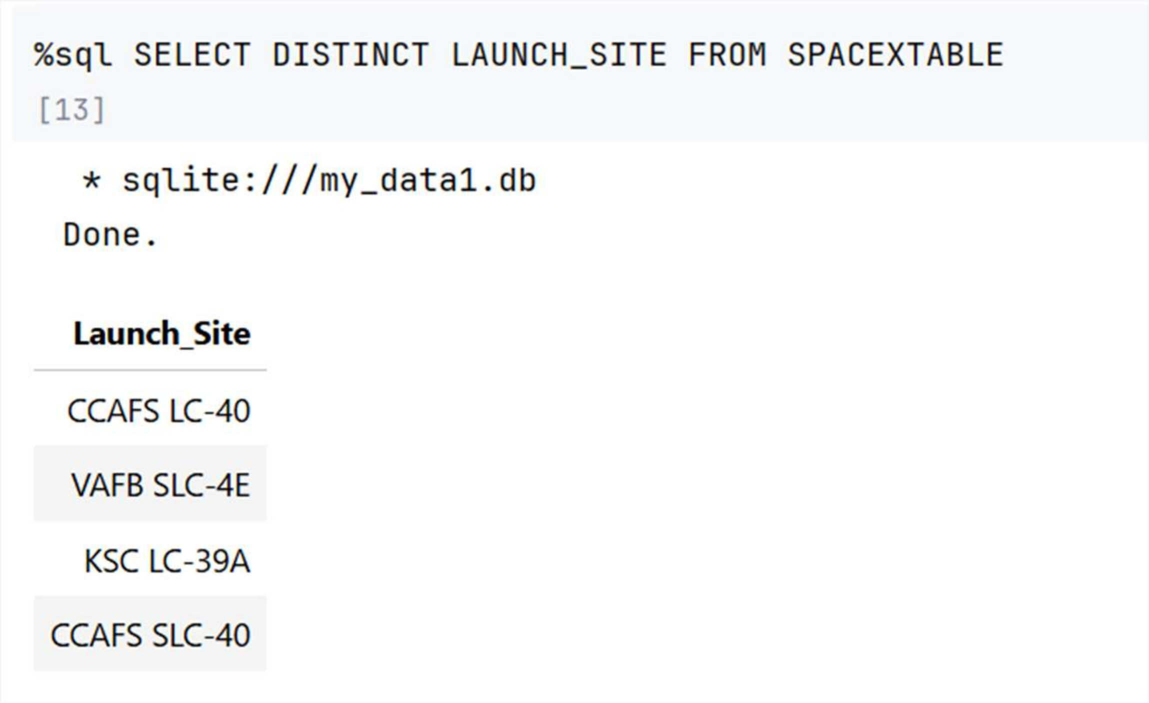
Payload vs. Orbit Type



Launch Success Yearly Trend



All Launch Site Names

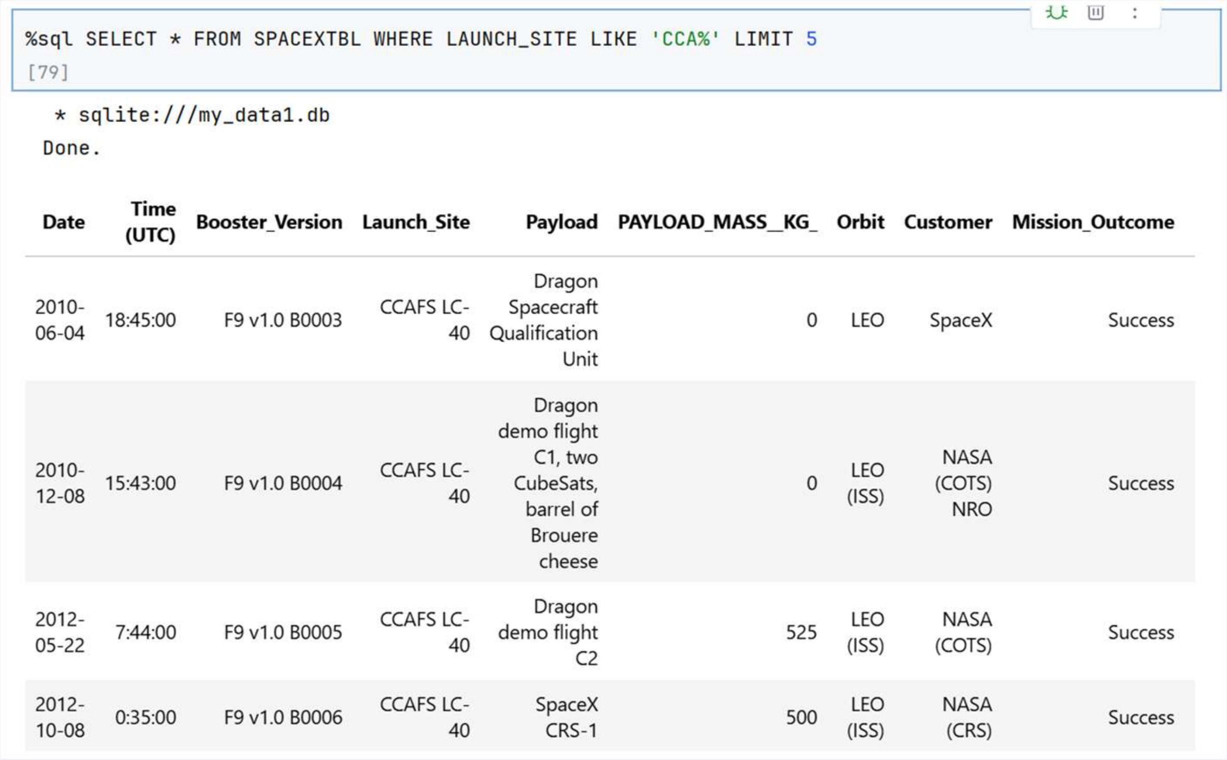


```
%sql SELECT DISTINCT LAUNCH_SITE FROM SPACEXTABLE
[13]
```

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

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

Launch Site Names Begin with 'CCA'

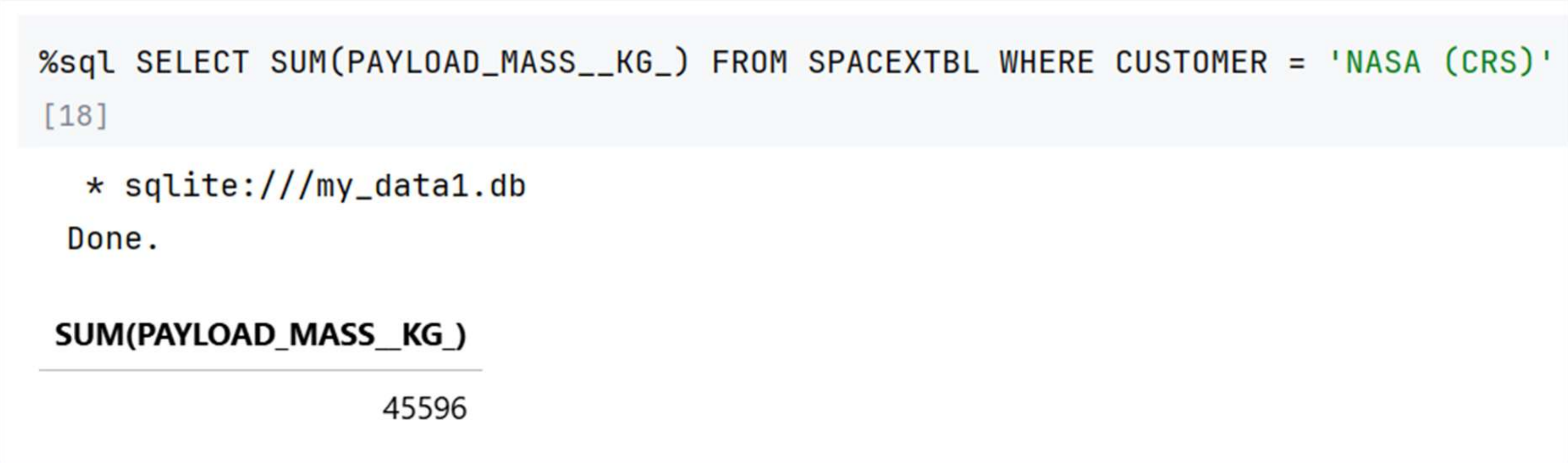


```
%sql SELECT * FROM SPACEXTBL WHERE LAUNCH_SITE LIKE 'CCA%' LIMIT 5
[79]
```

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

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome
2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success
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
2012-05-22	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success
2012-10-08	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success

Total Payload Mass

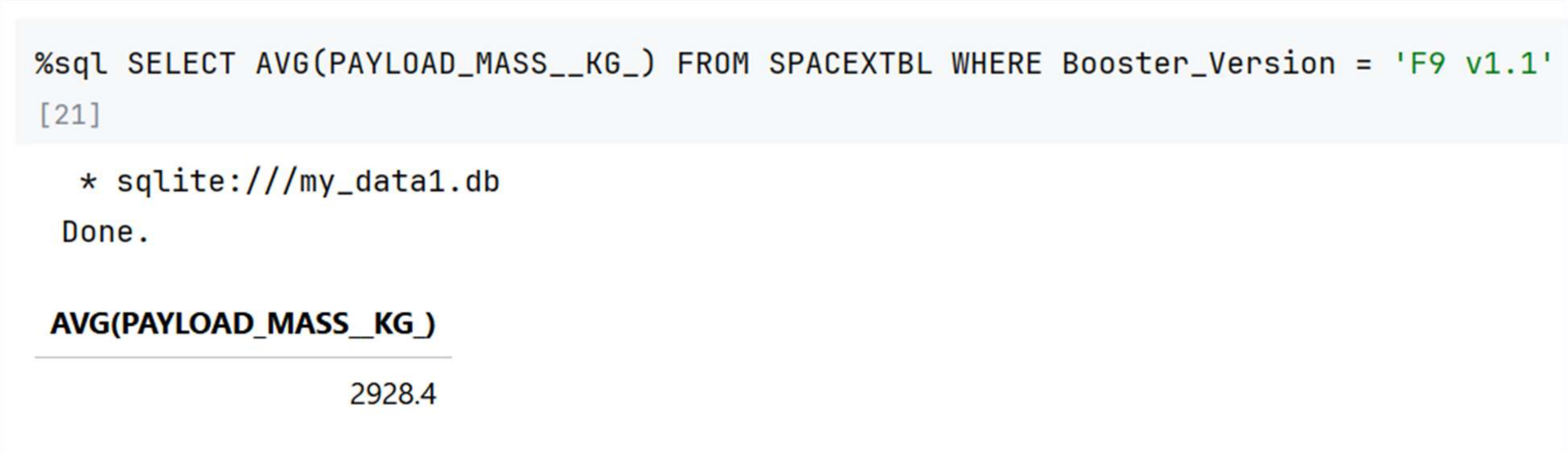


```
%sql SELECT SUM(PAYLOAD_MASS__KG_) FROM SPACEXTBL WHERE CUSTOMER = 'NASA (CRS)'
[18]
```

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

SUM(PAYLOAD_MASS__KG_)
45596

Average Payload Mass by F9 v1.1

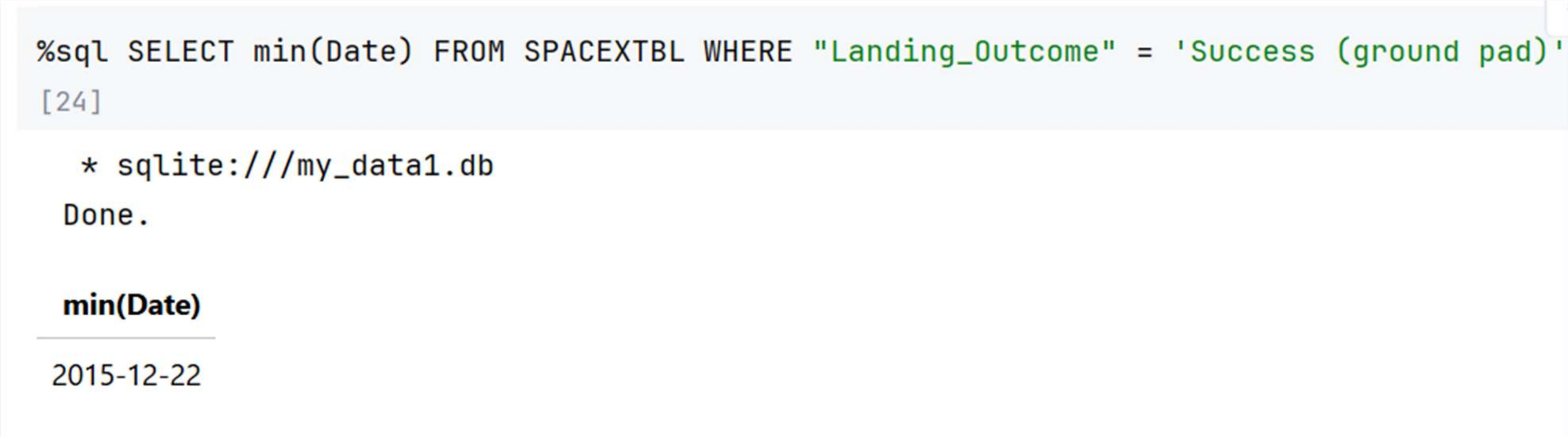


```
%sql SELECT AVG(PAYLOAD_MASS__KG_) FROM SPACEXTBL WHERE Booster_Version = 'F9 v1.1'
[21]
```

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

AVG(PAYLOAD_MASS__KG_)
                2928.4
```

First Successful Ground Landing Date

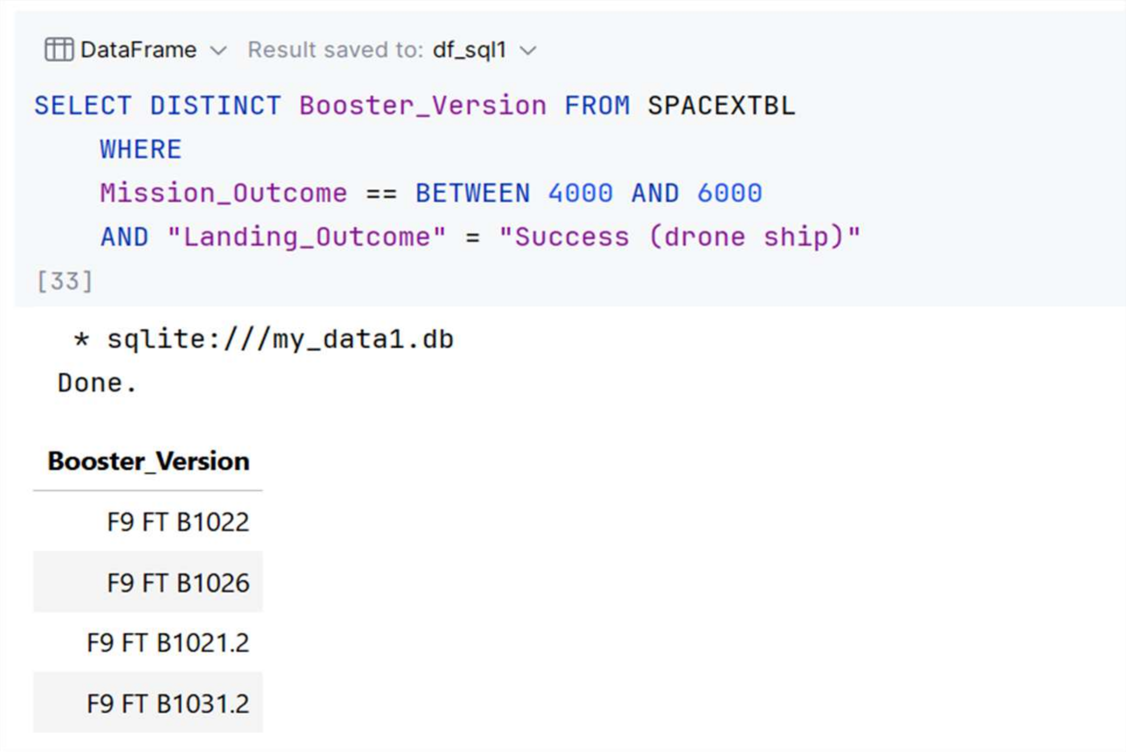


```
%sql SELECT min(Date) FROM SPACEXTBL WHERE "Landing_Outcome" = 'Success (ground pad)'
[24]
```

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

min(Date)
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000



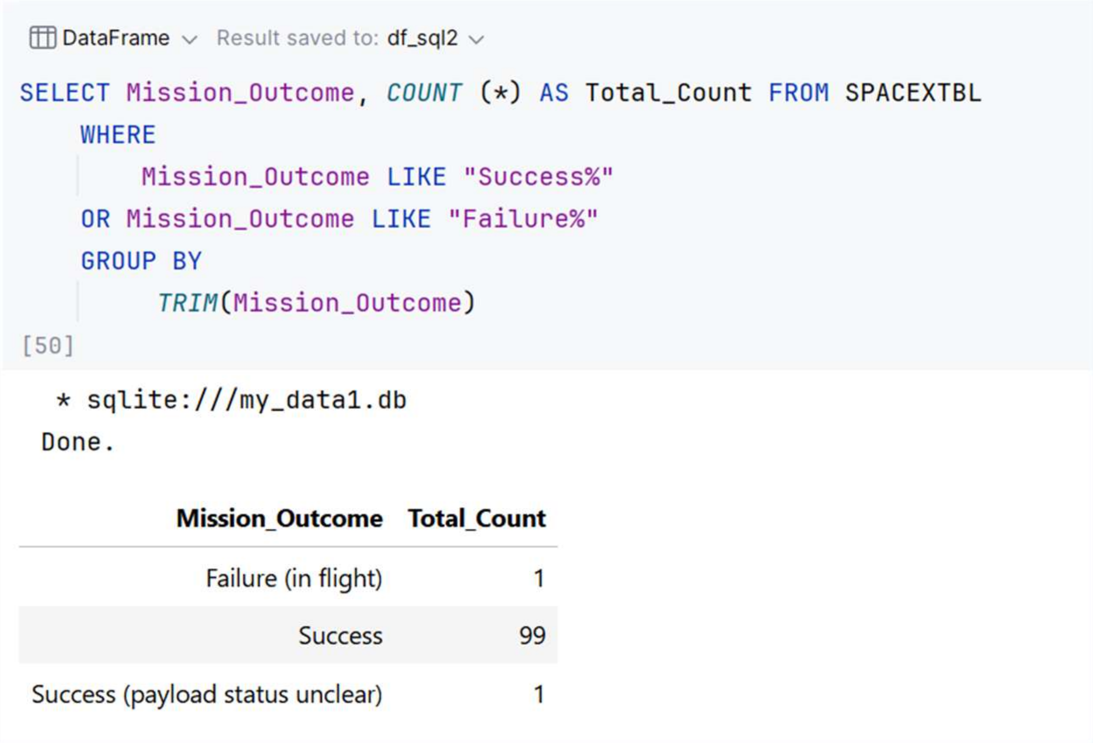
DataFrame Result saved to: df_sql1

```
SELECT DISTINCT Booster_Version FROM SPACEXTBL
    WHERE
    Mission_Outcome == BETWEEN 4000 AND 6000
    AND "Landing_Outcome" = "Success (drone ship)"
[33]
```

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

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

Total Number of Successful and Failure Mission Outcomes

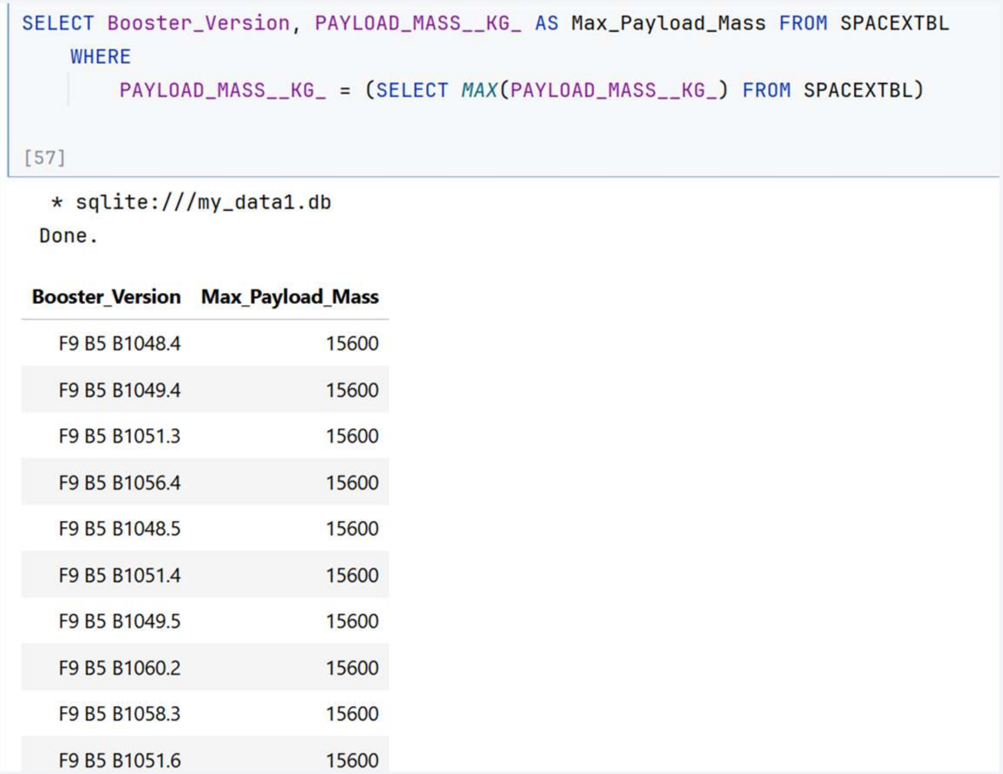


```
SELECT Mission_Outcome, COUNT (*) AS Total_Count FROM SPACEXTBL
    WHERE
        Mission_Outcome LIKE "Success%"
    OR Mission_Outcome LIKE "Failure%"
    GROUP BY
         TRIM(Mission_Outcome)
[50]
```

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

Mission_Outcome	Total_Count
Failure (in flight)	1
Success	99
Success (payload status unclear)	1

Boosters Carried Maximum Payload



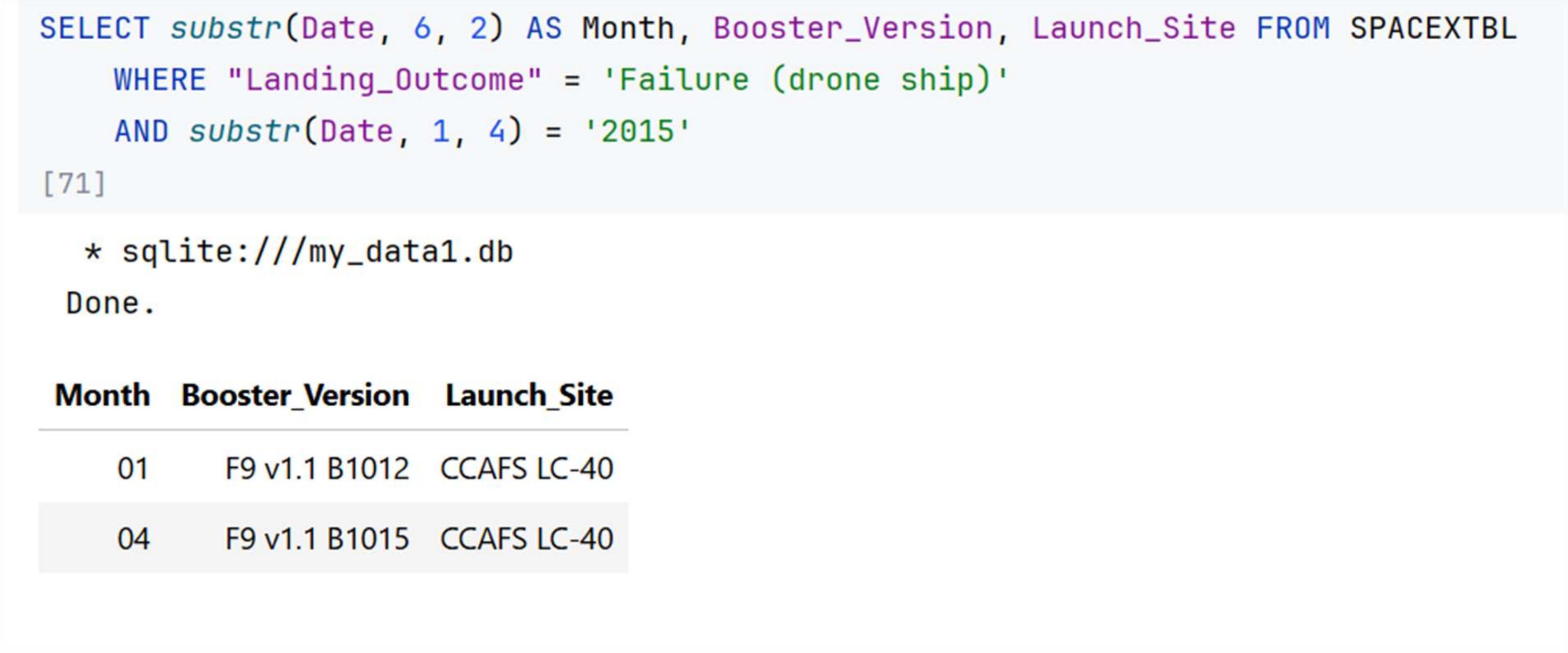
```
SELECT Booster_Version, PAYLOAD_MASS__KG_ AS Max_Payload_Mass FROM SPACEXTBL
    WHERE
        PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTBL)

[57]
```

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

Booster_Version	Max_Payload_Mass
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

2015 Launch Records

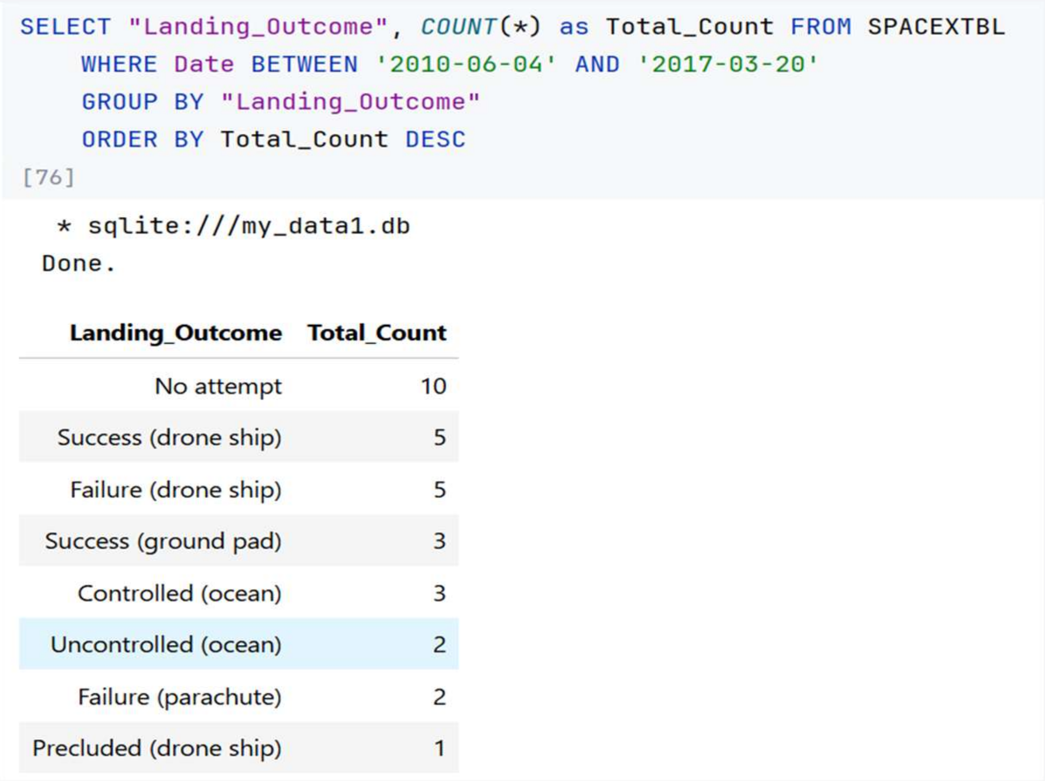


```
SELECT substr(Date, 6, 2) AS Month, Booster_Version, Launch_Site FROM SPACEXTBL
    WHERE "Landing_Outcome" = 'Failure (drone ship)'
    AND substr(Date, 1, 4) = '2015'
[71]
```

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

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

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



```
SELECT "Landing_Outcome", COUNT(*) as Total_Count FROM SPACEXTBL
    WHERE Date BETWEEN '2010-06-04' AND '2017-03-20'
    GROUP BY "Landing_Outcome"
    ORDER BY Total_Count DESC
[76]
```

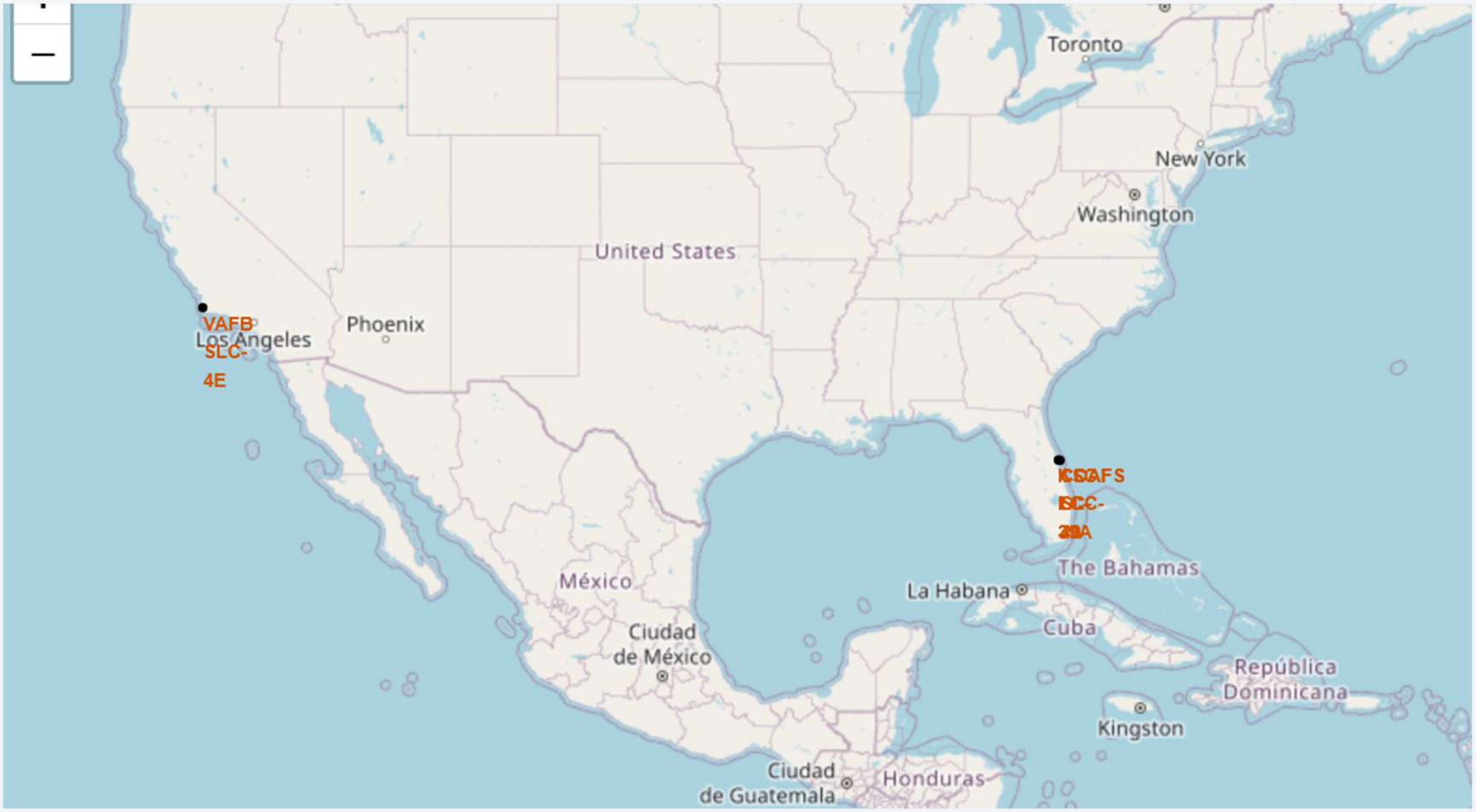
```
 * sqlite:///my_data1.db
Done.
```

Landing_Outcome	Total_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

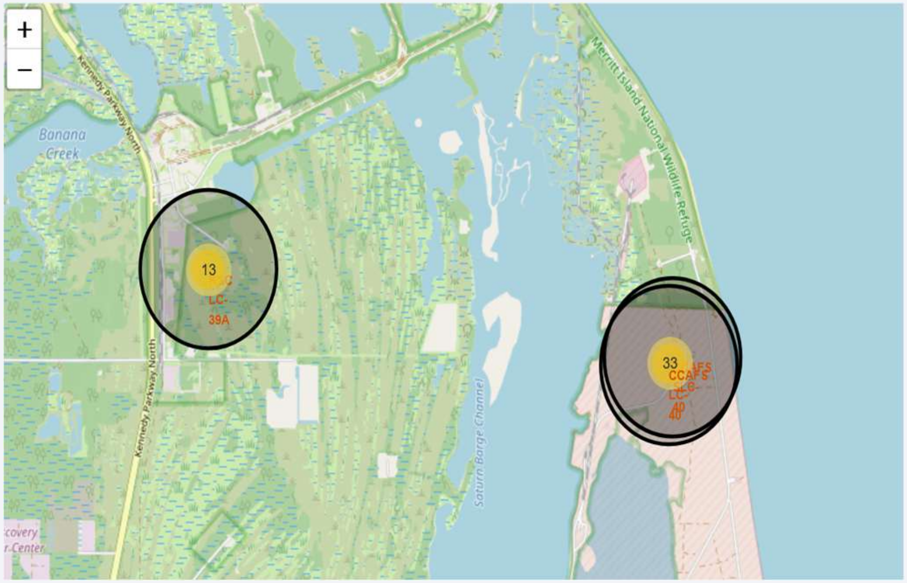
Section 3

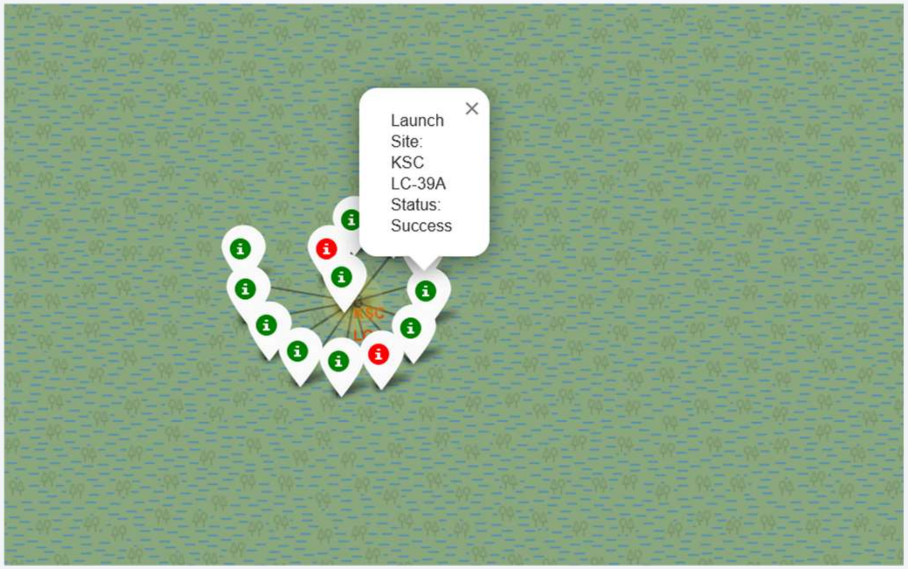
Launch Sites Proximities Analysis

Launch Sites of Falcon9

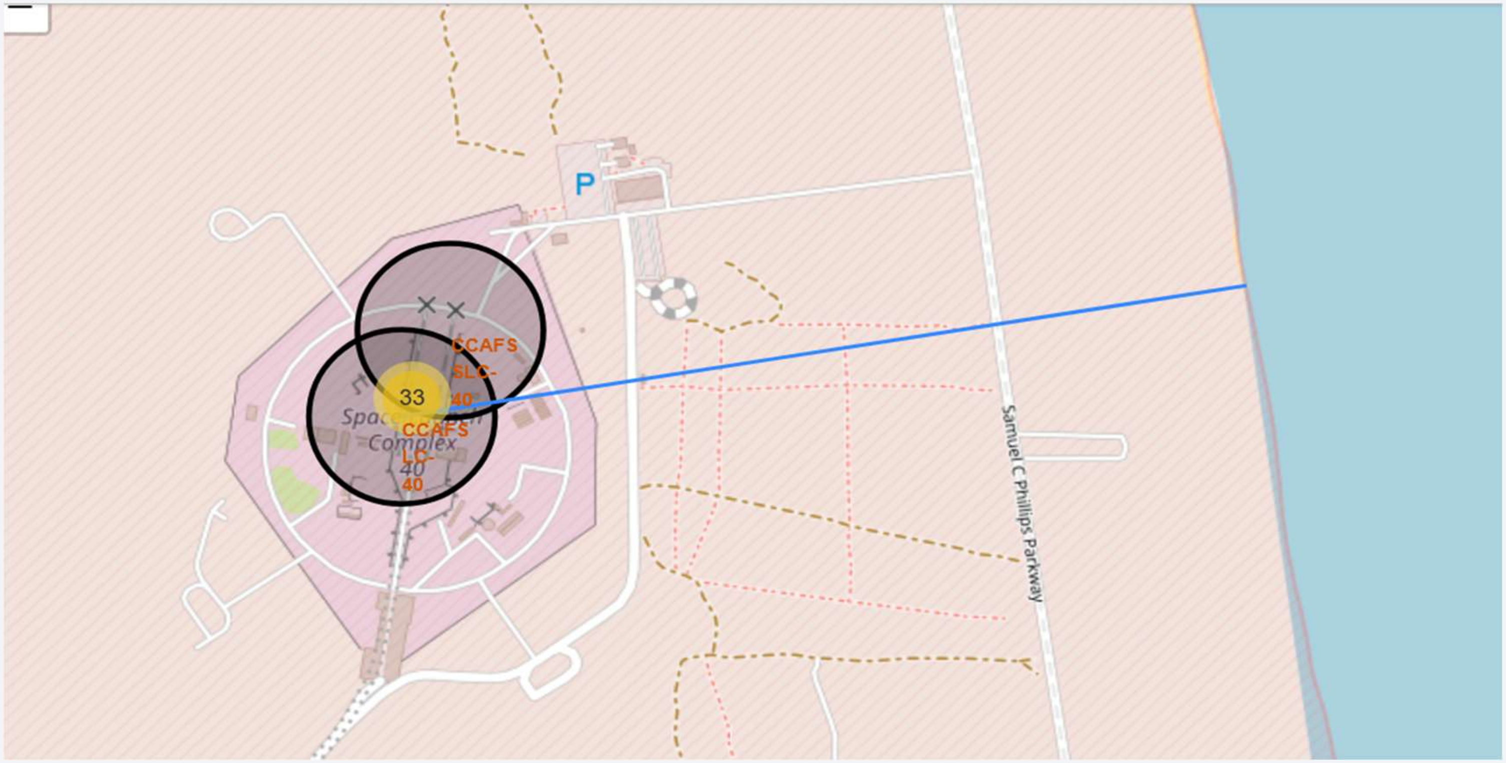


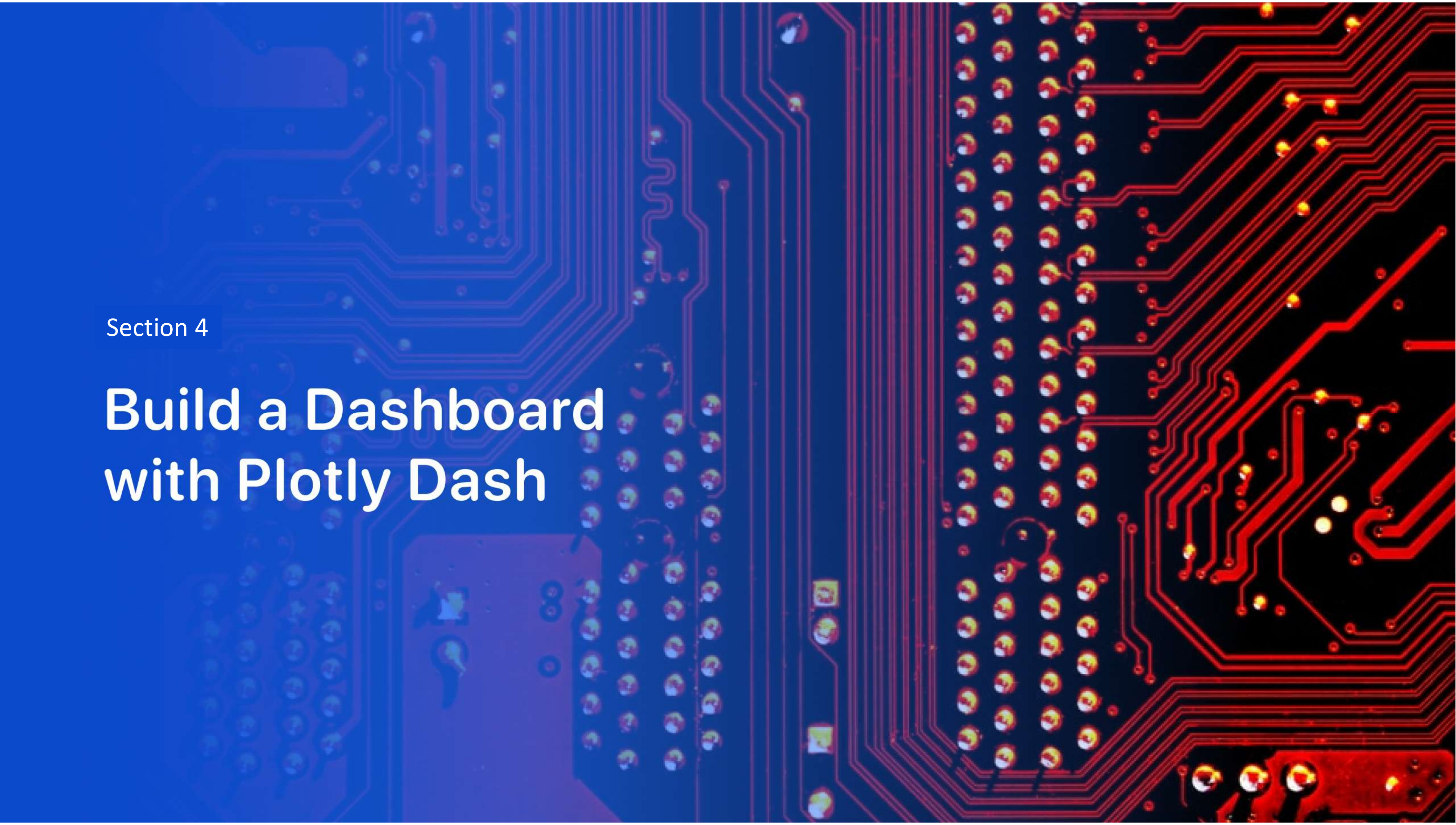
Eastern Launch Sites & KSCLC-39A Site Success/Failure Markings




Launch Site: KSC LC-39A Status: Success

CCAFS SLC-40 Site Proximity to Coast (Blue Line)

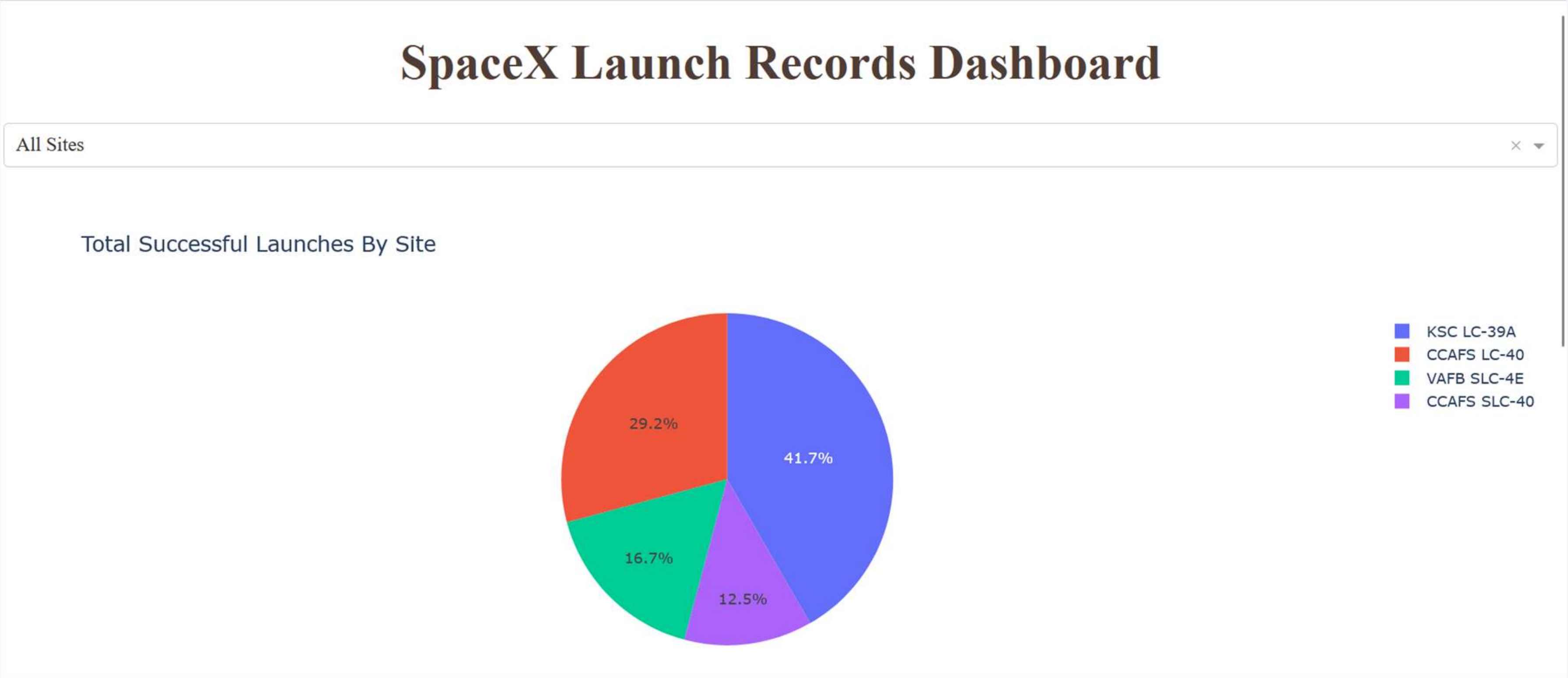




Section 4

Build a Dashboard with Plotly Dash

Success Rate of Launch – All Sites

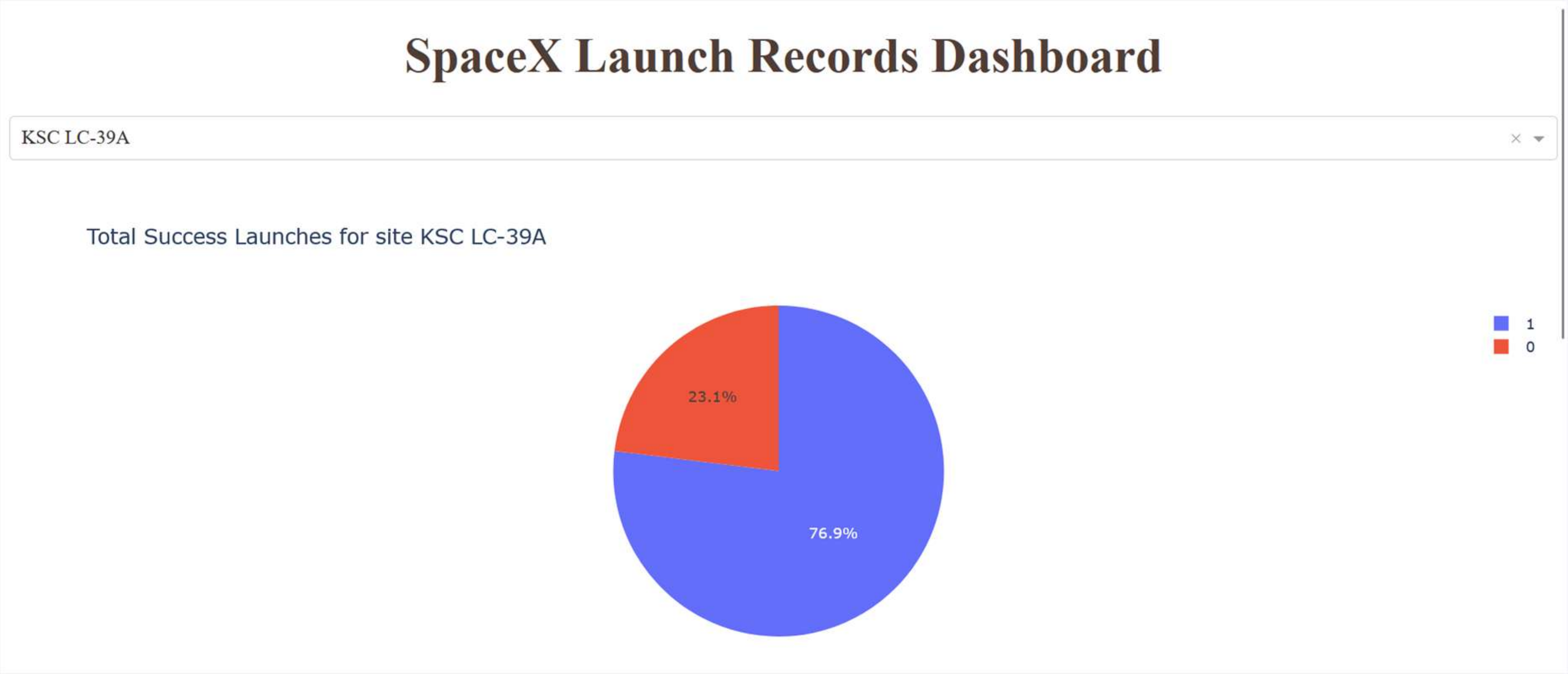


SpaceX Launch Records Dashboard

All Sites

Total Successful Launches By Site

Pie Chart for Site with Highest Launch Success Ratio

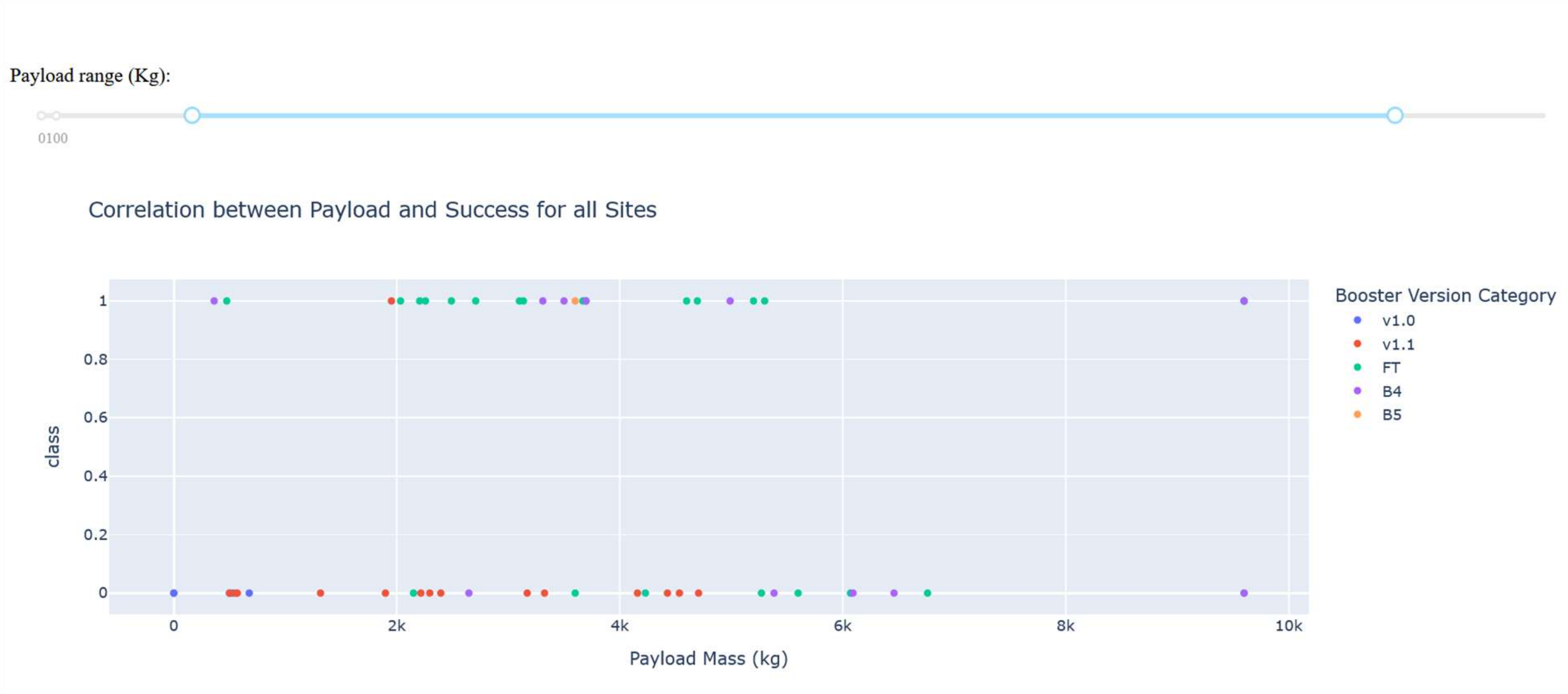


SpaceX Launch Records Dashboard

KSC LC-39A

Total Success Launches for site KSC LC-39A

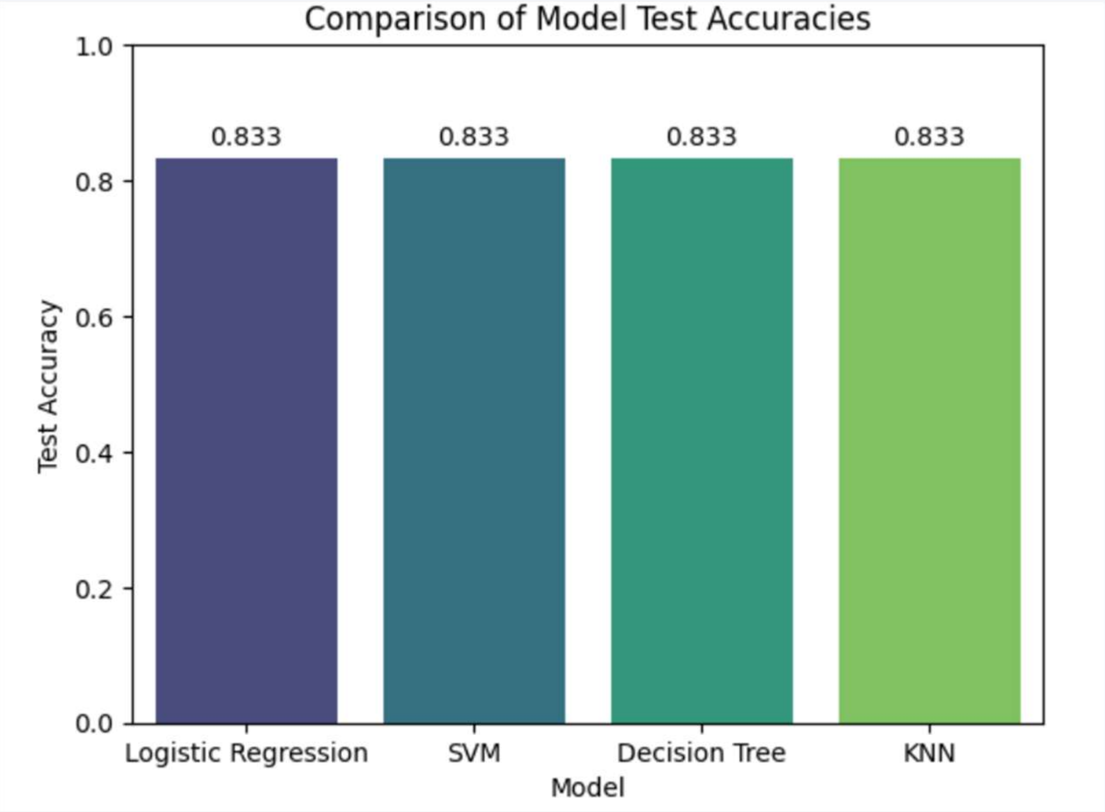
Scatter Plot for Payload vs Success – All Sites


Payload range (Kg):

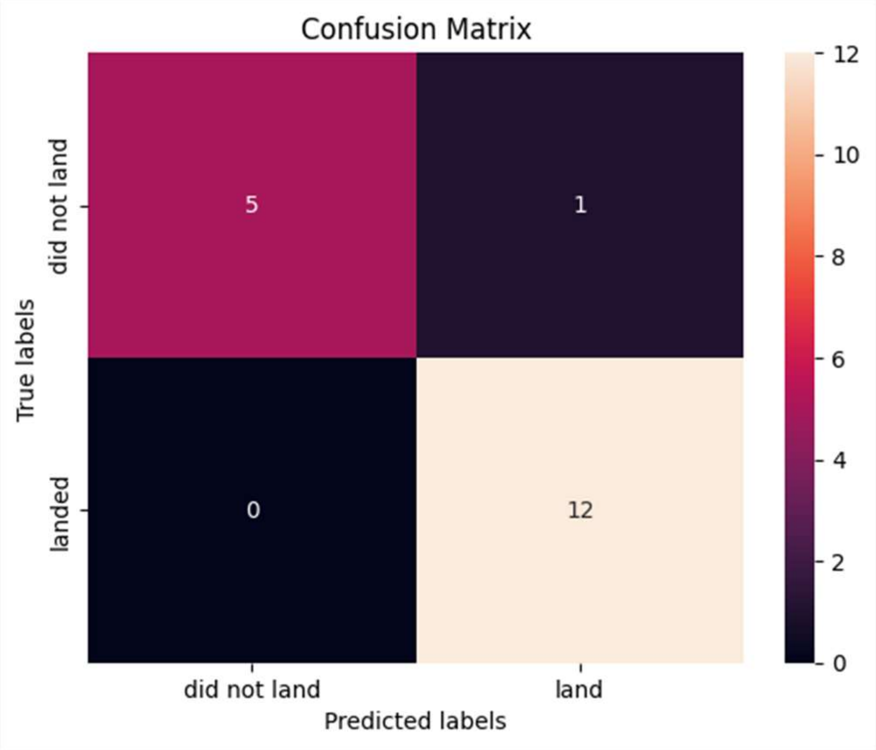
Section 5

Predictive Analysis (Classification)

Classification Accuracy



Confusion Matrix – Best Model Decision Tree Classifier



Conclusions

- Key features (Flight Number, Payload Mass, Launch Site, Orbit) provide a reliable approach for predicting launch success
- All four models score a high of 83.33% of test accuracy,
- Best model is Decision Tree Classifier as it has better reliability based on confusion matrix
- Accurate prediction of launch success can help in cost optimization since the first stage launcher can be reused

Thank you!

Appendix