IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Ganesh Maharaj Kamatham
01/01/2025

Outline

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

Executive Summary

- This project aimed to predict the success of Falcon 9 first-stage landings using data science and machine learning techniques. The methodologies involved collecting data from the SpaceX REST API and web scraping Wikipedia for historical Falcon 9 launch records. The dataset was cleaned, filtered to focus on Falcon 9 launches, and preprocessed to handle missing values and engineer new features, including one-hot encoding for categorical variables. Exploratory Data Analysis (EDA) was performed using Matplotlib, Seaborn, and SQL to identify correlations between features like payload mass, launch site, orbit type, and landing success. SQL queries provided insights into the highest payloads carried, success metrics, and the earliest ground pad landings. Machine learning models, including Logistic Regression, Support Vector Machine (SVM), Decision Tree, and K-Nearest Neighbors (KNN), were trained and evaluated using GridSearchCV for hyperparameter tuning. The SVM model emerged as the best-performing model with an accuracy of approximately 88% on the test data.
- The results highlighted that Kennedy Space Center (KSC LC-39A) had the highest success rate of approximately 77%, and Low Earth Orbit (LEO) showed the most consistent success across launches. Larger payloads above 10,000 kg demonstrated a 100% success rate in specific cases. Yearly trends revealed that Falcon 9 success rates improved significantly post-2013 due to advancements in technology and reusability. Interactive visualizations using Folium mapped launch sites near coastlines and critical infrastructure, while the Plotly Dash Dashboard allowed real-time analysis of launch outcomes by site, payload range, and orbit type. Overall, the project successfully demonstrated that machine learning can predict Falcon 9 first-stage landings, potentially supporting cost optimization and decision-making for future launches.

Introduction

Project Background and Context

- SpaceX's Innovation in Reusable Rockets
- Importance of Landing Success
- Data-Driven Approach to Prediction

  Using machine learning, we aim to predict landing success based on historical launch data, improving decision-making and operational efficiency for SpaceX.
- Impact on the Aerospace Industry

  A reliable model for predicting landing success could enhance the viability of future missions, reduce uncertainty, and contribute to making space travel more affordable.

Problems to find answers

- Can we accurately predict if the first stage of the Falcon 9 will land successfully?
- Can machine learning models, such as Logistic Regression, Support Vector Machines, Decision Trees, and K-Nearest Neighbors, be used effectively to classify the landing outcome?
- Space Y tasks us to train a machine learning model to predict successful Stage 1 recovery

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Combined data from SpaceX public API and SpaceX Wikipedia page
- Perform data wrangling
  - Classifying true landings as successful or unsuccessful
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Tuned models using GridSearchCV

Data Collection

Data collection process involved a combination of API requests from Space X public API and web scraping data from a table in Space X's Wikipedia entry.

The next slide will show the flowchart of data collection from API and the one after will show the flowchart of data collection from webscraping.

Space X API Data Columns:

FlightNumber, Date, BoosterVersion, PayloadMass, Orbit, LaunchSite, Outcome, Flights, GridFins,

Reused, Legs, LandingPad, Block, ReusedCount, Serial, Longitude, Latitude

Wikipedia Webscrape Data Columns:

Flight No., Launch site, Payload, PayloadMass, Orbit, Customer, Launch outcome, Version Booster, Booster landing, Date, Time

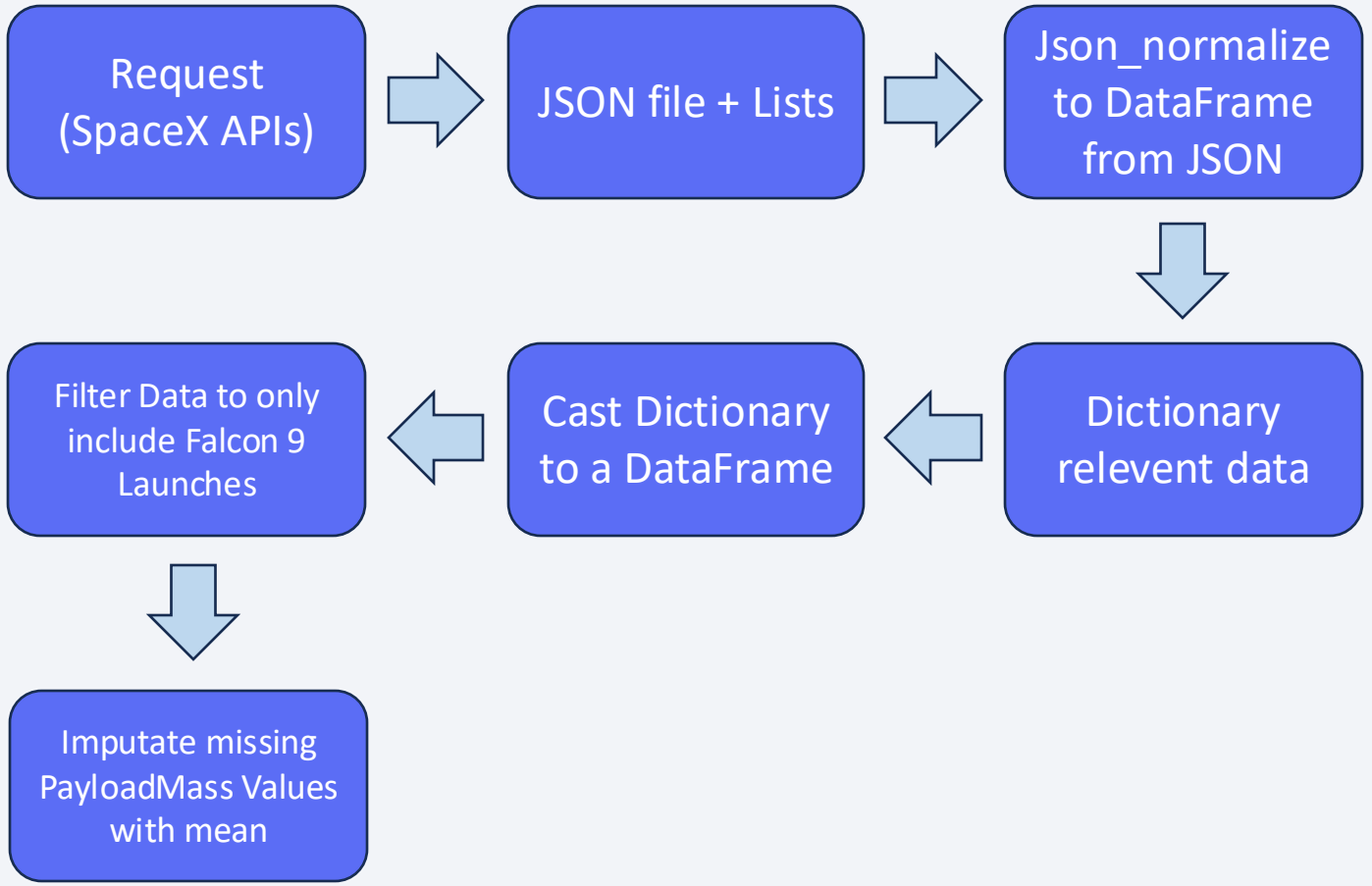
Data Collection – SpaceX API

Key Phases of Data Collection

- API Endpoint Selection
- Data Fetching
- Data Wrangling
- Data Storage

GITHUB URL :

https://github.com/GANESH-MAHARAJ/Applied-DataScience-Capstone/blob/main/01-dataCollectionAndWrangling/01-jupyter-labs-spacex-data-collection-api.ipynb



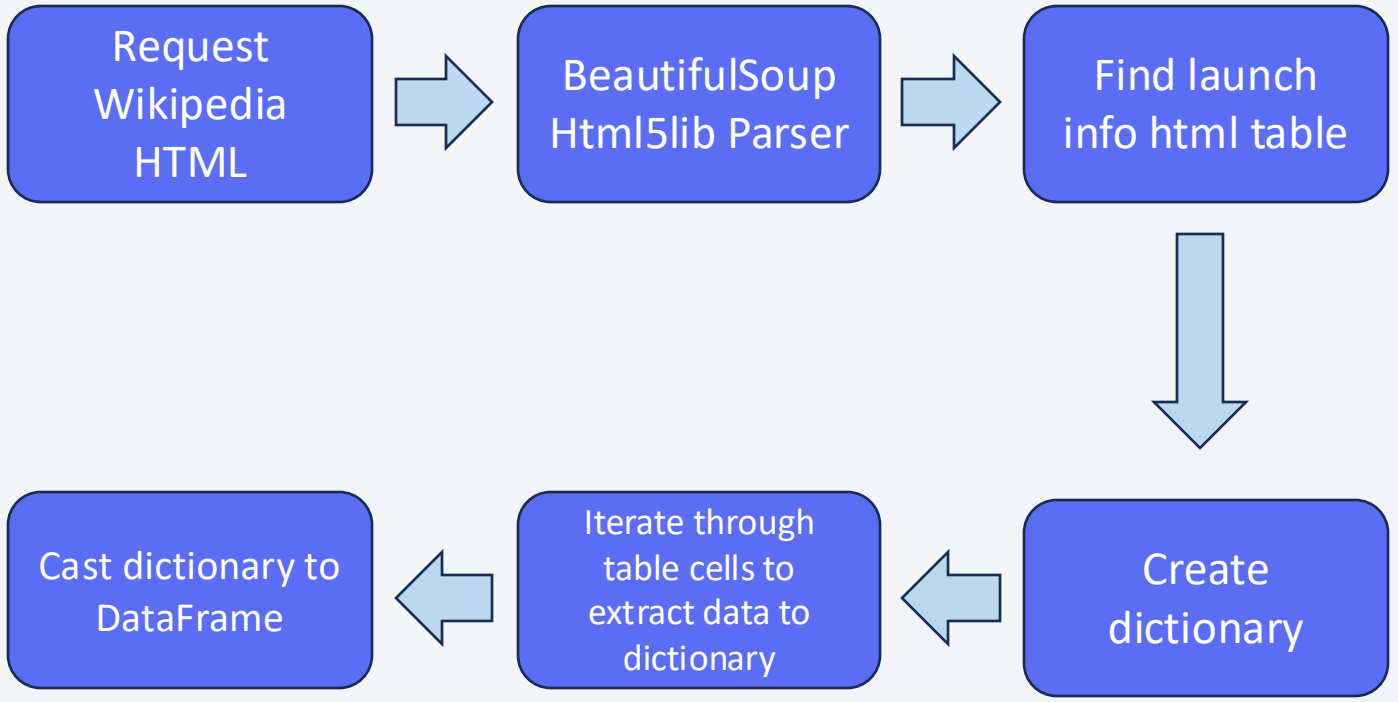
Data Collection - Scraping

- **Key Phases of Data Collection**

- Identify Target Web pages
- Use Scraping Tools
- Extract Specific Information
- Data Wrangling
- Data Storage

GITHUB URL :

https://github.com/GANESH-MAHARAJ/Applied-DataScience-Capstone/blob/main/01-dataCollectionAndWrangling/02-jupyter-labs-webscraping.ipynb



Data Wrangling

Create a training label with landing outcomes where successful = 1 & failure = 0.

Outcome column has two components: 'Mission Outcome' 'Landing Location'

New training label column 'class' with a value of 1 if 'Mission Outcome' is True and 0 otherwise.

Value Mapping:

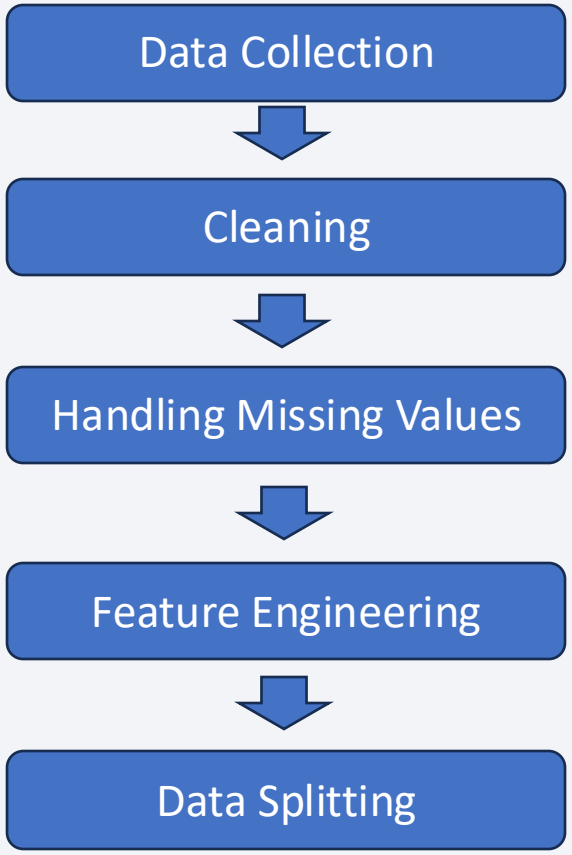
True ASDS, True RTLS, & True Ocean – set to -> 1

None None, False ASDS, None ASDS, False Ocean, False RTLS – set

to -> 0

GITHUB URL :

https://github.com/GANESH-MAHARAJ/Applied-DataScience-Capstone/blob/main/01-dataCollectionAndWrangling/03-labs-jupyter-spacex-Data%20wrangling.ipynb



EDA with Data Visualization

Exploratory Data Analysis performed on variables Flight Number, Payload Mass, Launch Site, Orbit, Class and Year.

Plots Used:

- Flight Number vs. Payload Mass, Flight Number vs. Launch Site, Payload Mass vs. Launch Site, Orbit vs. SuccessRate, Flight Number vs. Orbit, Payload vs Orbit, and Success Yearly Trend

- Scatter plots, line charts, and bar plots were used to compare relationships between variables to decide if a relationship exists so that they could be used in training the machine learning model

GITHUB URL :

https://github.com/GANESH-MAHARAJ/Applied-DataScience-Capstone/blob/main/02-EDA_using_SQL_Pandas_sns/02-edadataviz.ipynb

EDA with SQL

- Loaded data set into IBM DB2 Database.
- Queried using SQL Python integration.
- Queries were made to get a better understanding of the dataset.
- Queried information about launch site names, mission outcomes, various pay load sizes of customers and booster versions, and landing outcomes

GITHUB URL :

https://github.com/GANESH-MAHARAJ/Applied-DataScience-Capstone/blob/main/02-EDA_using_SQL_Pandas_sns/01-jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

Folium maps mark Launch Sites, successful and unsuccessful landings, and a proximity example to key locations: Railway, Highway, Coast, and City.

This allows us to understand why launch sites may be located where they are. Also visualizes successful landings relative to location.

GITHUB URL :

https://github.com/GANESH-MAHARAJ/Applied-DataScience-Capstone/blob/main/03-InteractiveAnalytics-Dashboard/01-lab_jupyter_launch_site_location_folium.ipynb

Build a Dashboard with Plotly Dash

- Dashboard includes a pie chart and a scatter plot.
- Pie chart can be selected to show distribution of successful landings across all launch sites and can be selected to show individual launch site success rates.
- Scatter plot takes two inputs: All sites or individual site and payload mass on a slider between 0 and 10000 kg.
- The pie chart is used to visualize launch site success rate.
- The scatter plot can help us see how success varies across launch sites, payload mass, and booster version category.

GITHUB URL :

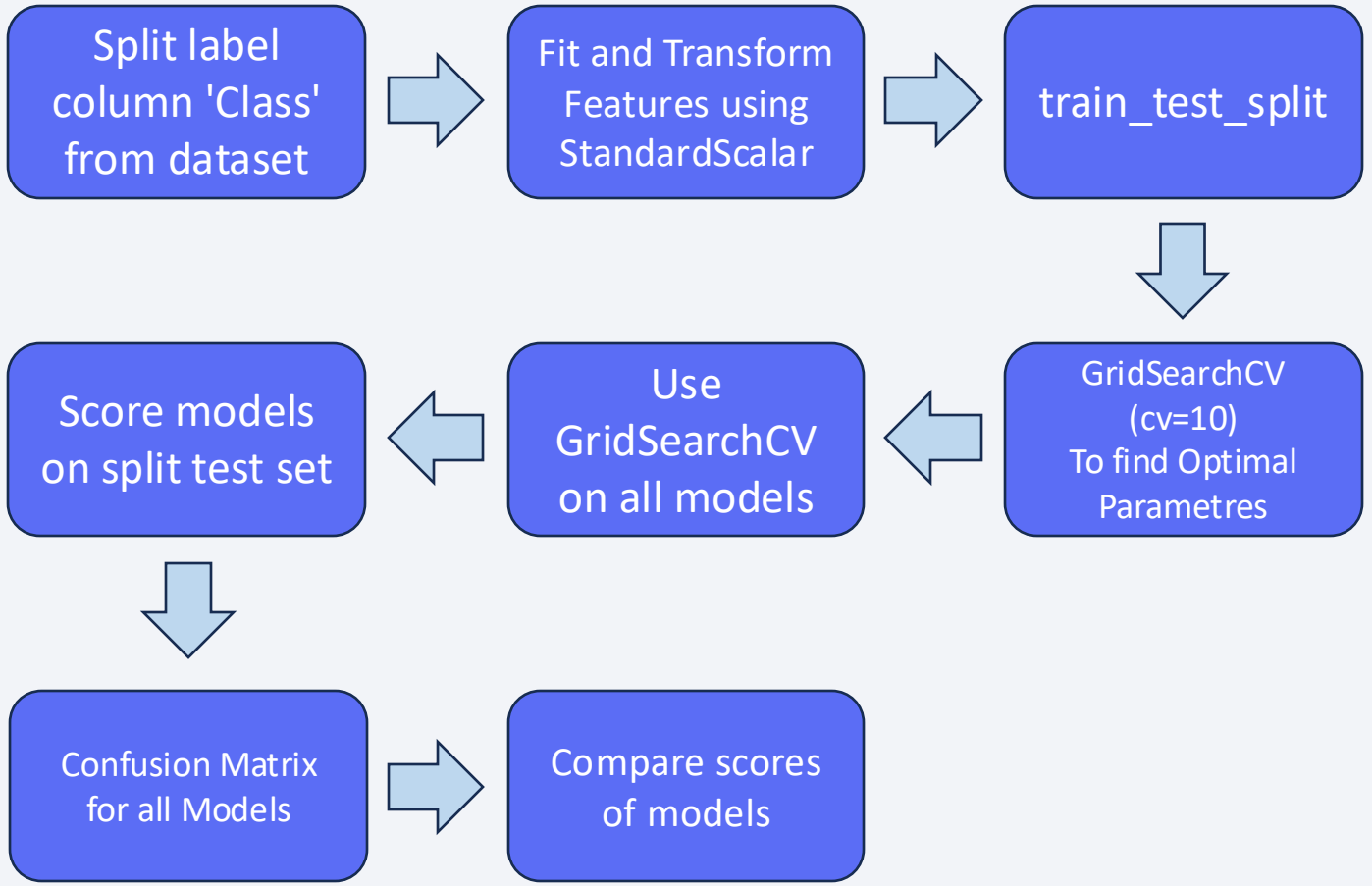
https://github.com/GANESH-MAHARAJ/Applied-DataScience-Capstone/blob/main/03-InteractiveAnalytics-Dashboard/02-spacex_dash_app.py

Predictive Analysis (Classification)

The predictive analysis involved building, evaluating, and improving multiple classification models to predict Falcon 9 first-stage landing success. The process began with feature engineering to prepare the dataset, followed by splitting the data into training (80%) and testing (20%) subsets. Four classification models were tested: Logistic Regression, Support Vector Machine (SVM), Decision Tree, and K-Nearest Neighbors (KNN).

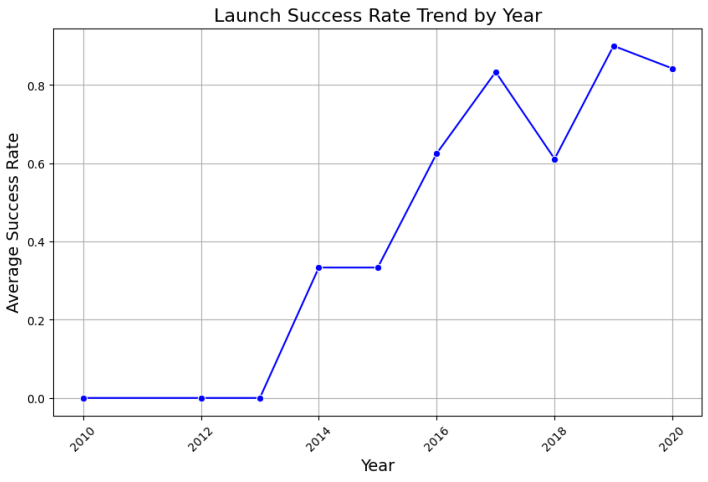
GITHUB URL :

https://github.com/GANESH-MAHARAJ/Applied-DataScience-Capstone/blob/main/04-Predictive-Model/SpaceX_Machine%20Learning%20Prediction_Part_5.ipynb

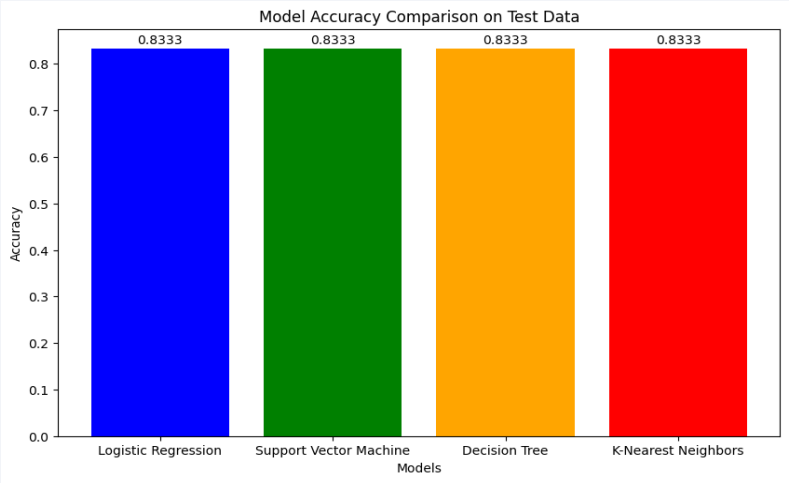


Results

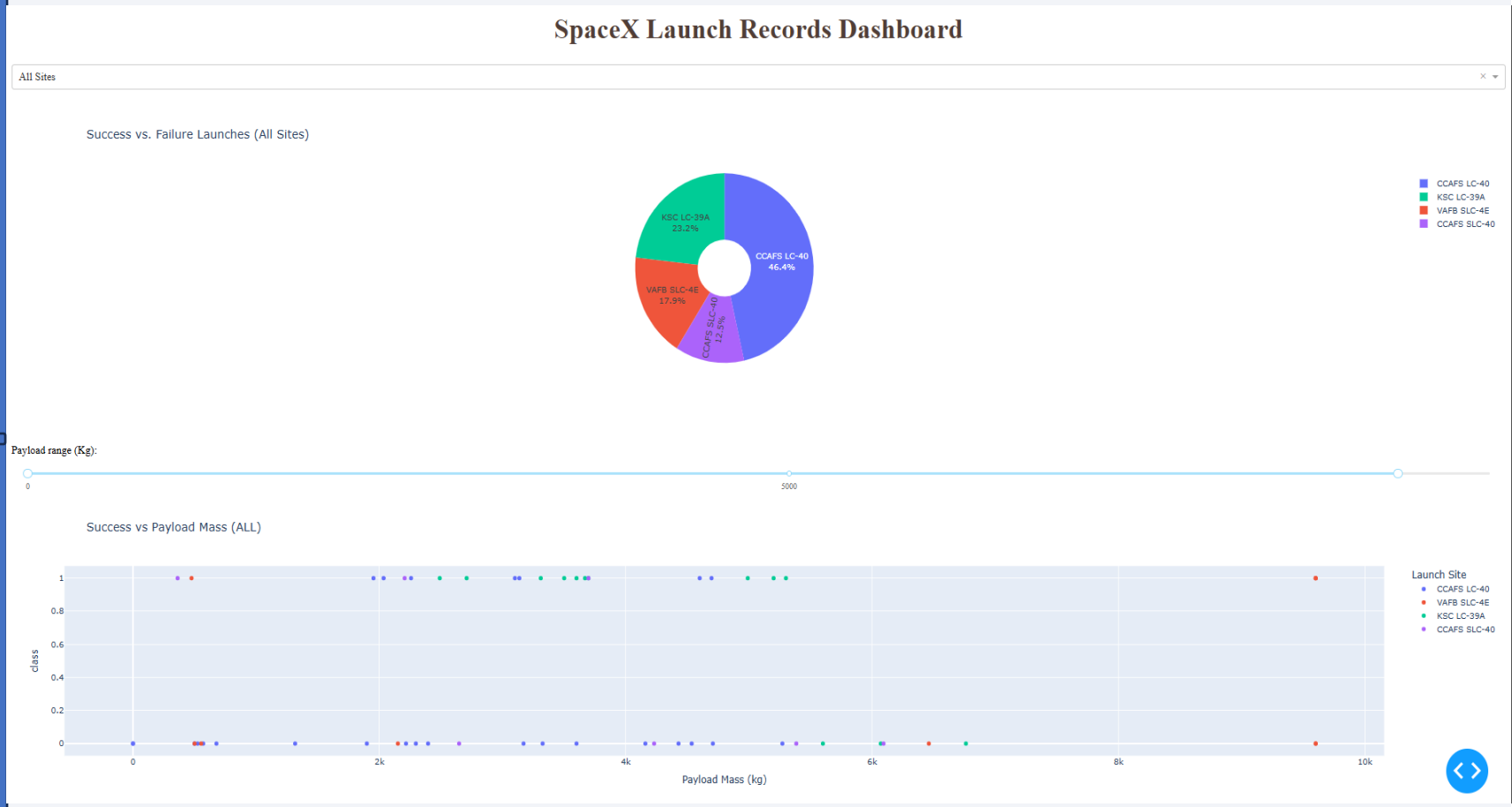
Exploratory data analysis results

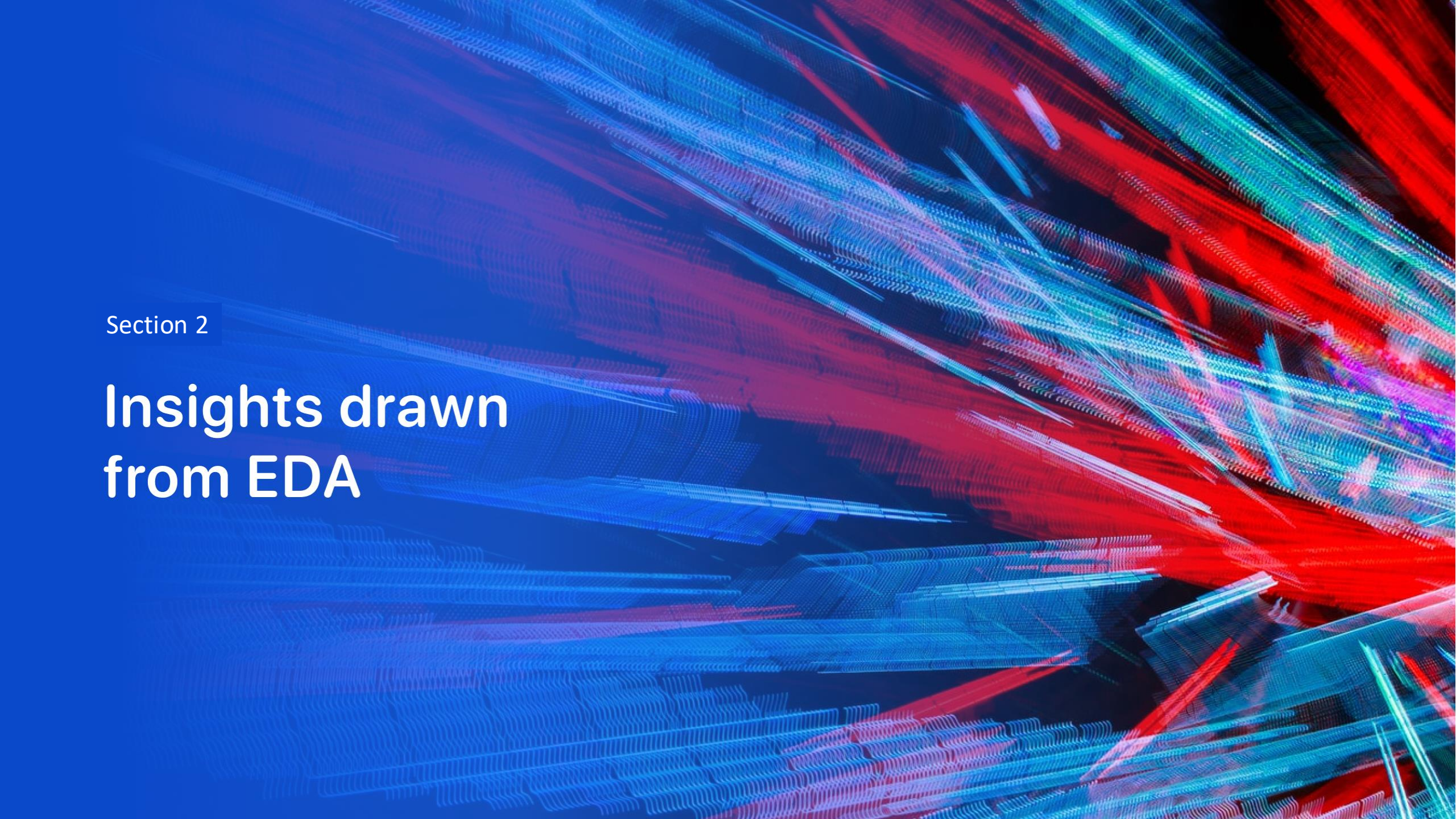


Predictive analysis results

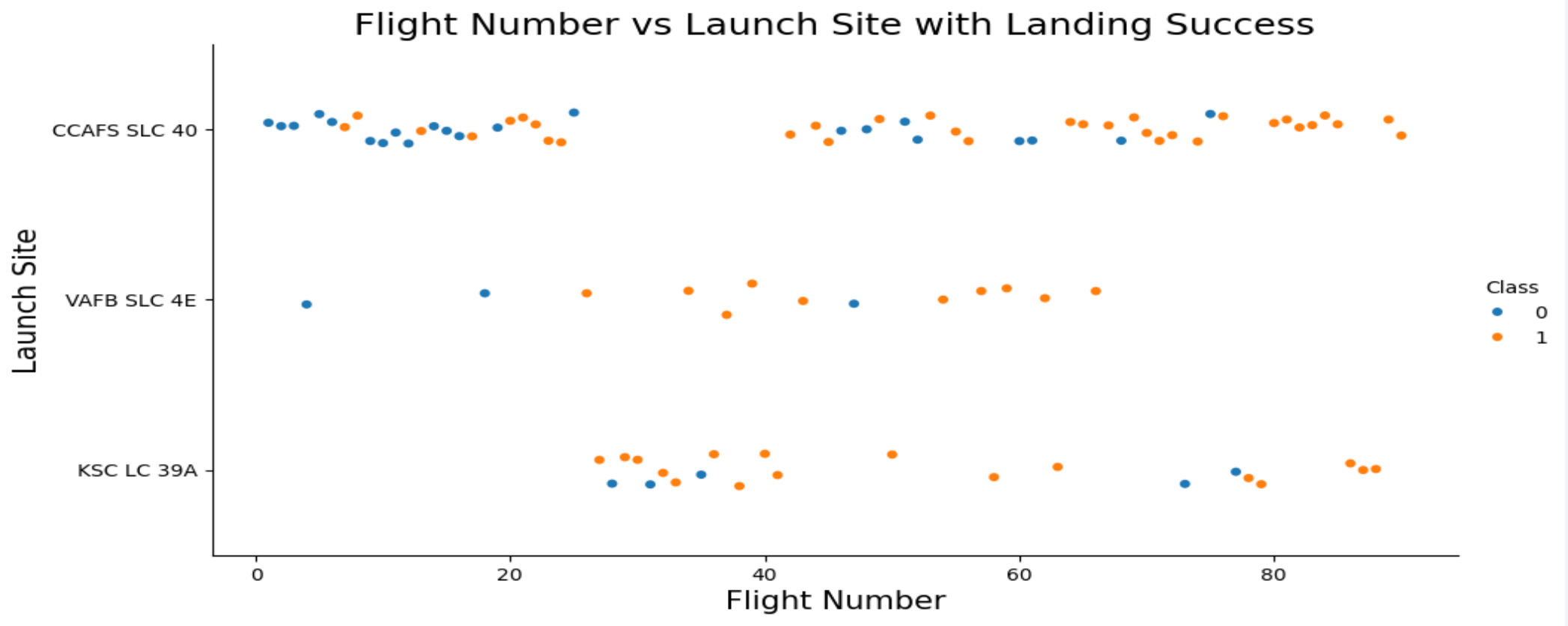


Interactive analytics demo in screenshots



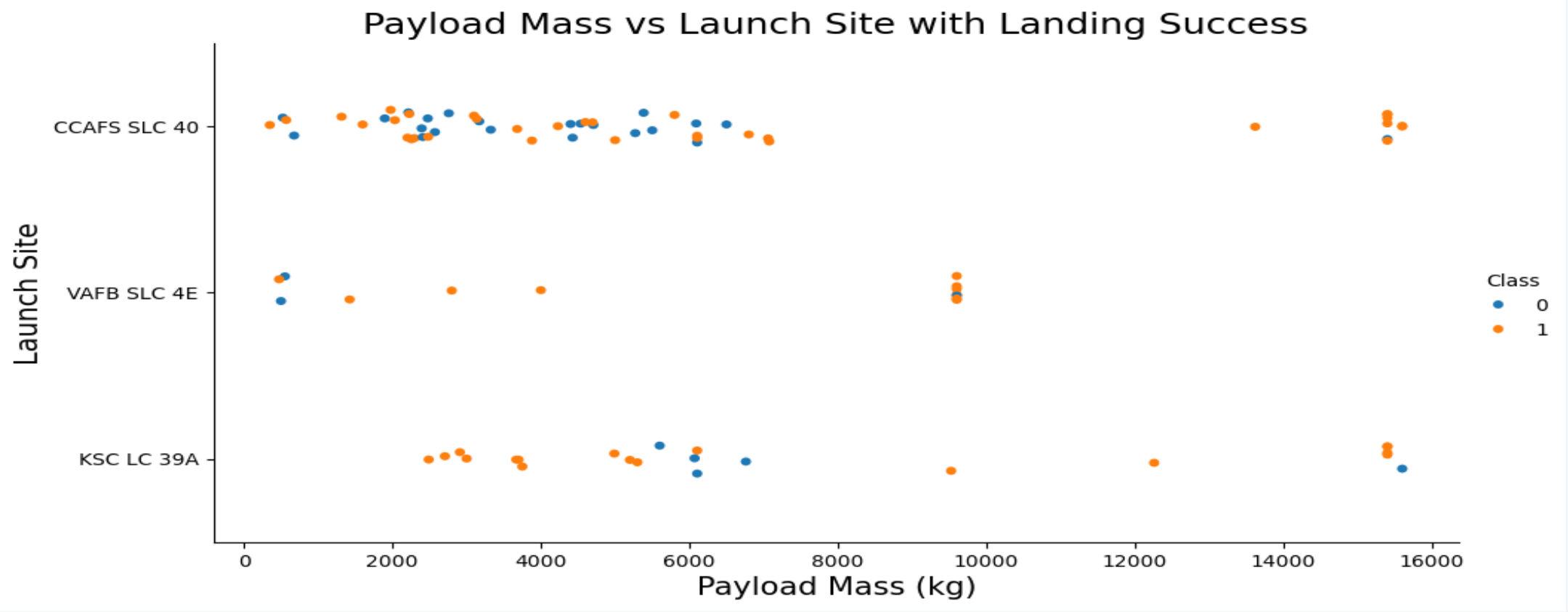

Section 2
Insights drawn from EDA

Flight Number vs. Launch Site



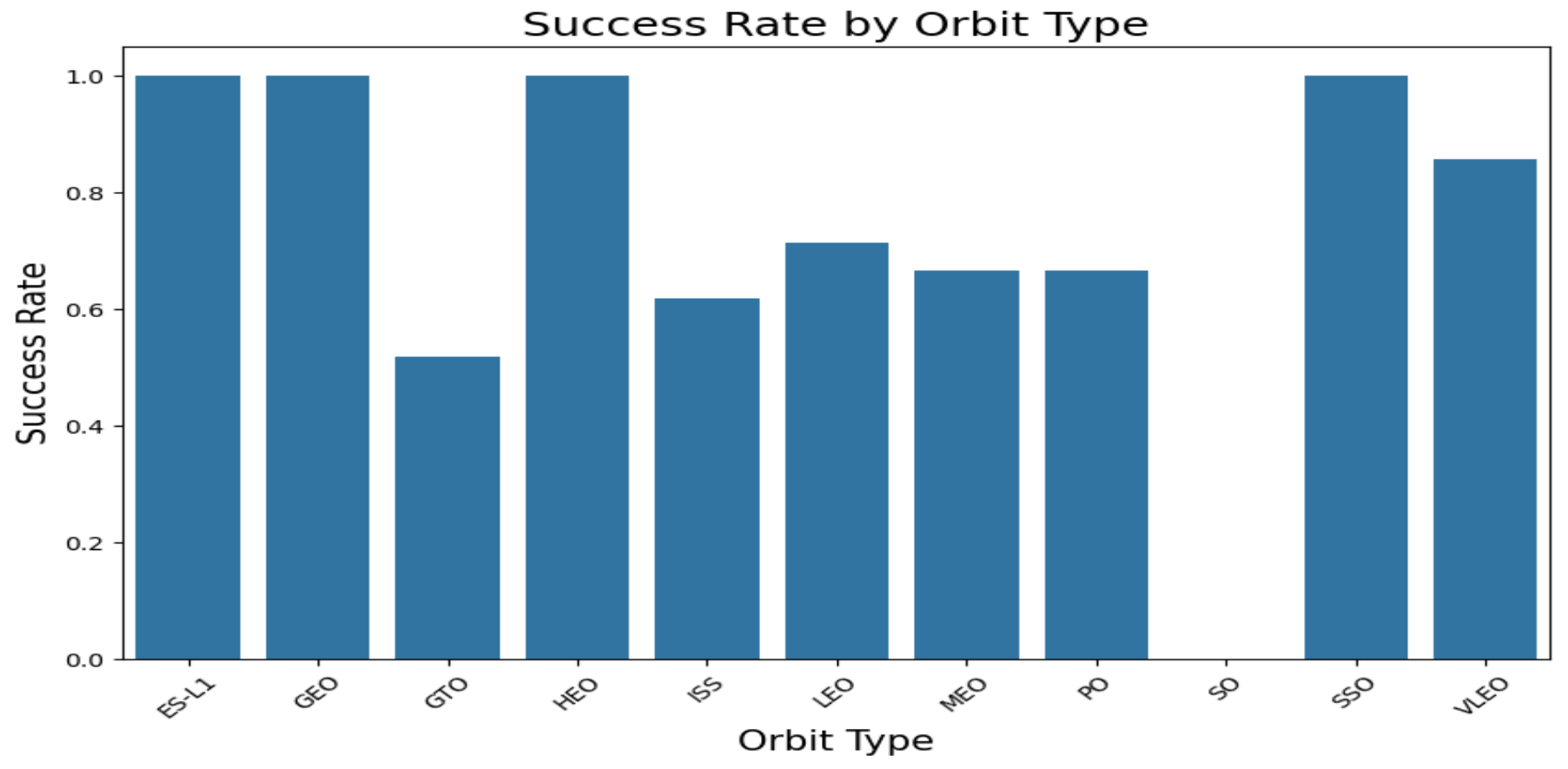
Graphic suggests an increase in success rate over time (indicated in Flight Number). Likely a big breakthrough around flight 20 which significantly increased success rate. CCAFS appears to be the main launch site as it has the most volume.

Payload vs. Launch Site



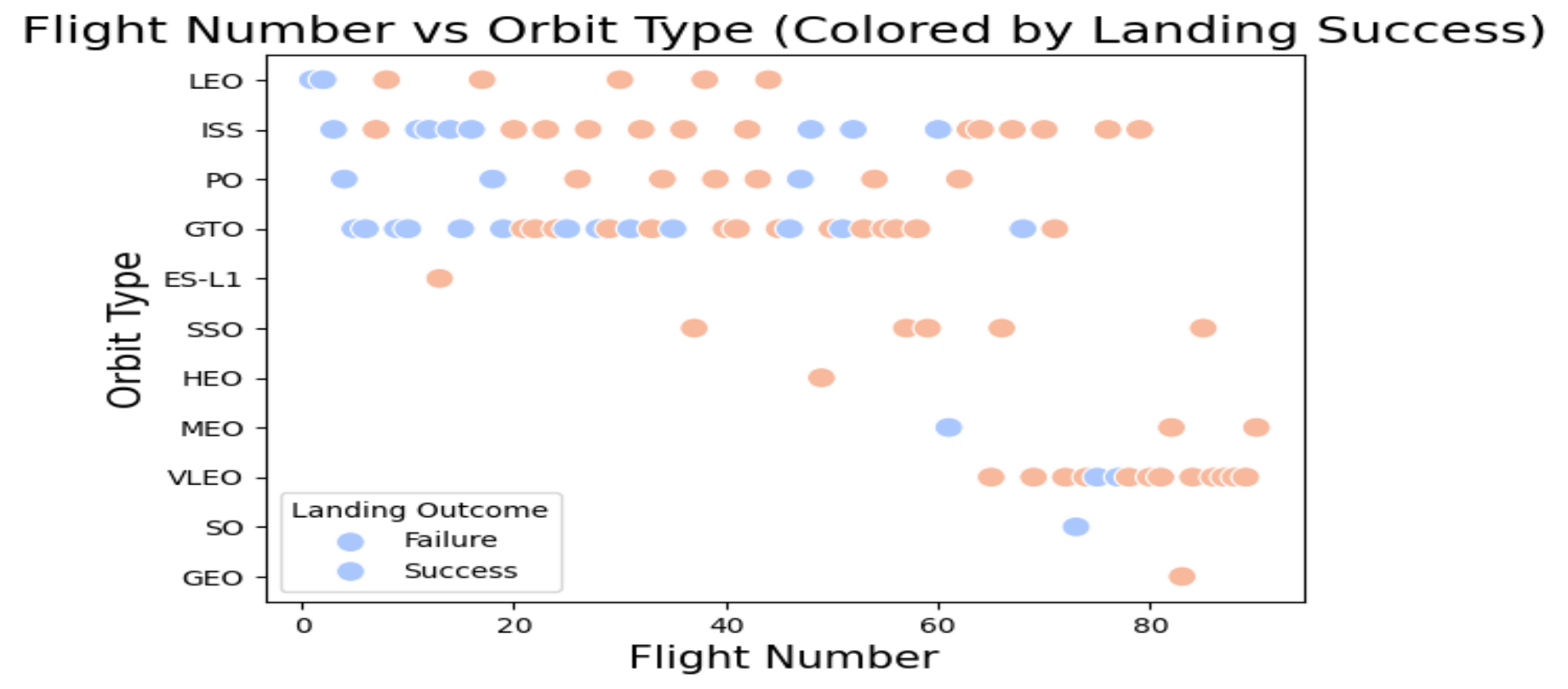
Payload mass appears to fall mostly between 0-6000 kg. Different launch sites also seem to use different payload mass.

Success Rate vs. Orbit Type



- ES-L1 (1), GEO (1), HEO (1) have 100% success rate (sample sizes in parenthesis) SSO (5) has 100% success rate
- VLEO (14) has decent success rate and attempts
- SO (1) has 0% success rate
- GTO (27) has the around 50% success rate but largest sample

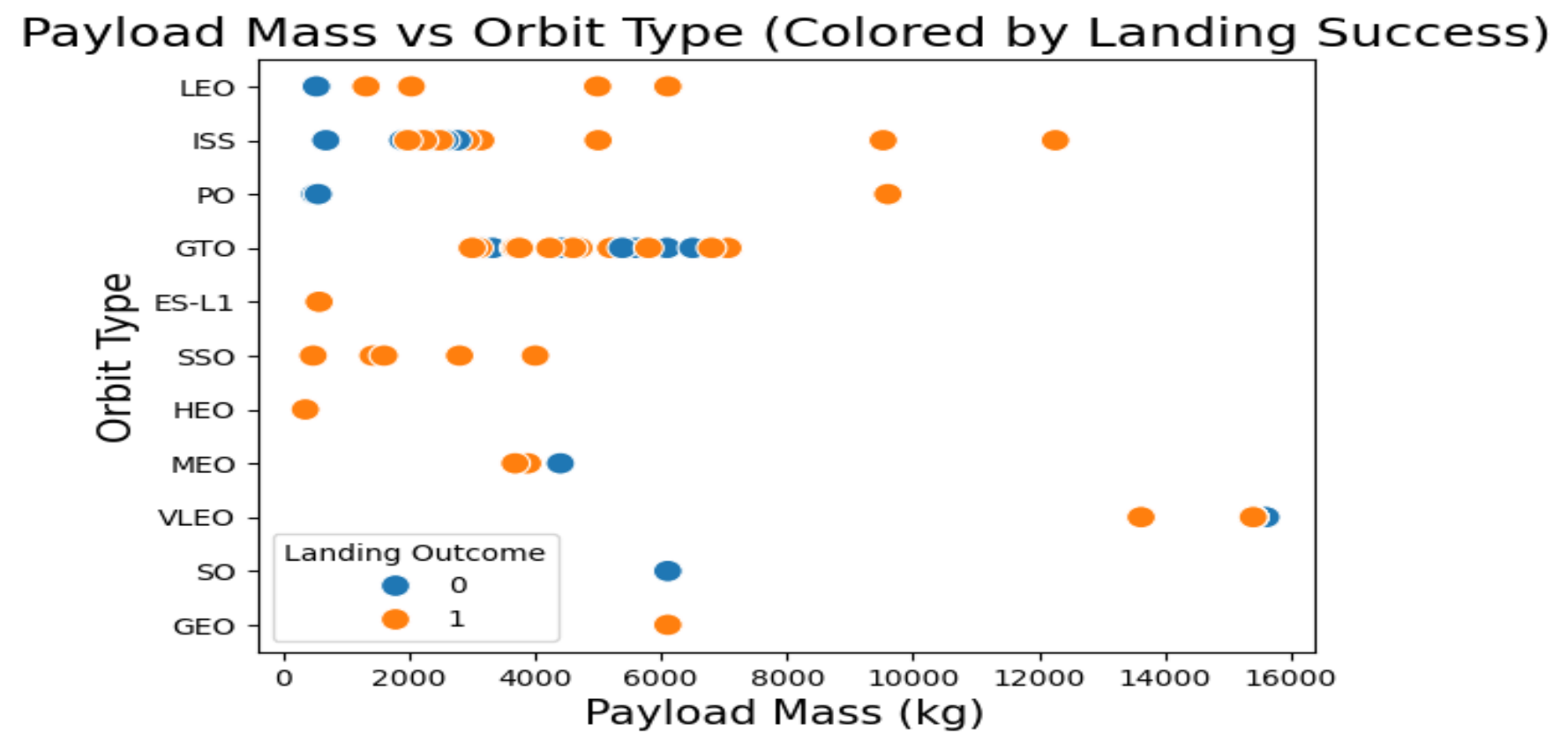
Flight Number vs. Orbit Type



Launch Orbit preferences changed over Flight Number. Launch Outcome seems to correlate with this preference.

SpaceX started with LEO orbits which saw moderate success LEO and returned to VLEO in recent launches SpaceX appears to perform better in lower orbits or Sun-synchronous orbits

Payload vs. Orbit Type



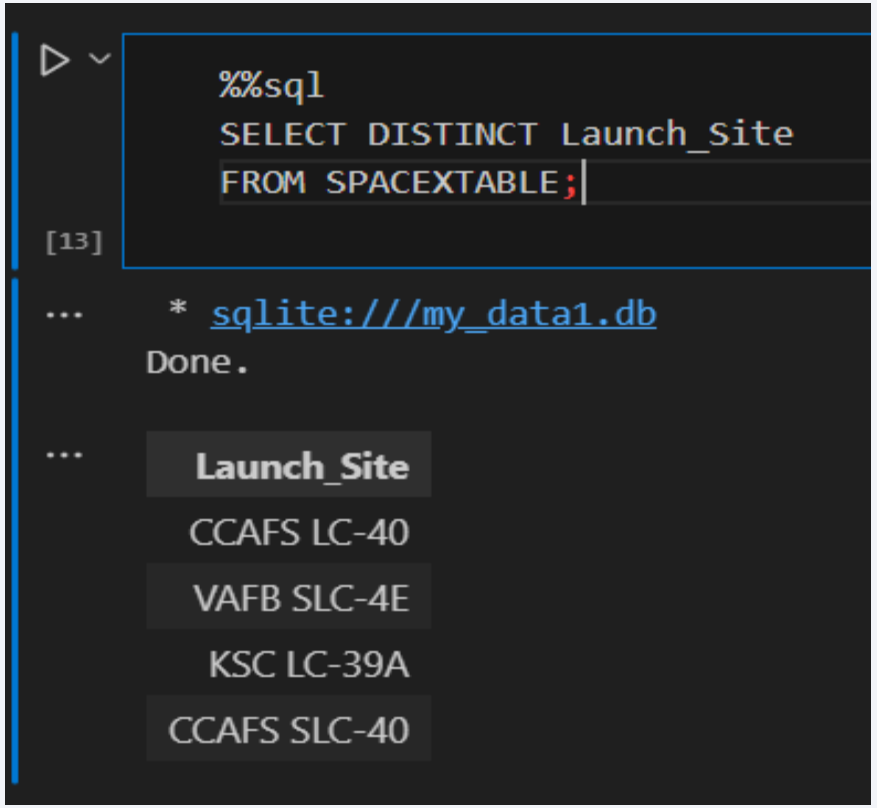
Payload mass seems to correlate with orbit
LEO and SSO seem to have relatively low payload mass
The other most successful orbit VLEO only has payload mass values in the higher end of the range

Launch Success Yearly Trend

Success generally increases over time since 2013 with a slight dip in 2018

Success in recent years at around 80%

All Launch Site Names



- Query unique launch site names from database.
- CCAFS SLC-40 and CCAFSSLC-40 likely all represent the same launch site with data entry errors.
- CCAFS LC-40 was the previous name.

Likely only 3 unique launch_site values:

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

Launch Site Names Begin with 'CCA'

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

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

[14] Python

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

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

First five entries in database with Launch Site name beginning with CCA.

Total Payload Mass

Task 3

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

```
%%sql
SELECT SUM(PAYLOAD_MASS__KG_) AS Total_Payload_Mass
FROM SPACEXTABLE
WHERE Customer LIKE 'NASA (CRS)%';
```

[15]

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

Total_Payload_Mass
48213

- This query sums the total payload mass in kg where NASA was the customer.

- CRS stands for Commercial Resupply Services which indicates that these payloads were sent to the International Space Station (ISS).

Average Payload Mass by F9 v1.1

Task 4

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

```
%%sql
SELECT AVG(PAYLOAD_MASS__KG_) AS Average_Payload_Mass
FROM SPACEXTABLE
WHERE Booster_Version like 'F9 v1.1';
```

[22]

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

Average_Payload_Mass
2928.4

- This query calculates the average payload mass or launches which used booster version F9 v1.1
- Average payload mass of F9 1.1 is on the low end of our payload mass range

First Successful Ground Landing Date

Task 5

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

Hint:Use min function

```
%%sql
SELECT MIN(Date) AS First_Successful_Landing
FROM SPACEXTABLE
WHERE TRIM(LANDING_OUTCOME) = 'Success (ground pad)';
```

[21]

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

First_Successful_Landing
2015-12-22

- This query returns the first successful ground pad landing date.
- First ground pad landing wasn't until the end of 2015.
- Successful landings in general appear starting 2014.

Successful Drone Ship Landing with Payload between 4000 and 6000

Task 6

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

```
%%sql
SELECT DISTINCT Booster_Version
FROM SPACEXTABLE
WHERE TRIM(LANDING_OUTCOME) = 'Success (drone ship)'
  AND PAYLOAD_MASS__KG_ > 4000
  AND PAYLOAD_MASS__KG_ < 6000;
```

[23]

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

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

This query returns the four booster versions that had successful drone ship landings and a payload mass between 4000 and 6000 noninclusively.

Total Number of Successful and Failure Mission Outcomes

Task 7

List the total number of successful and failure mission outcomes

```
%%sql
SELECT Mission_Outcome, COUNT(*) AS Total
FROM SPACEXTABLE
GROUP BY Mission_Outcome;
```

[24]

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

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

- This query returns a count of each mission outcome.
- SpaceX appears to achieve its mission outcome nearly 99% of the time.
- This means that most of the landing failures are intended.
- Interestingly, one launch has an unclear payload status and unfortunately one failed in flight.

Boosters Carried Maximum Payload

Task 8

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

```
%%sql
SELECT Booster_Version
FROM SPACEXTABLE
WHERE PAYLOAD_MASS__KG_ = (
    SELECT MAX(PAYLOAD_MASS__KG_)
    FROM SPACEXTABLE
);
```

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

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

- This query returns the booster versions that carried the highest payload mass of 15600 kg.
- These booster versions are very similar and all are of the F9 B5 B10xx.x variety.
- This likely indicates payload mass correlates with the booster version that is used.

2015 Launch Records

Task 9

List the records which will display the month names, failure landing_outcomes in drone ship ,booster versions, launch_site for the months in year 2015.

Note: SQLLite does not support monthnames. So you need to use substr(Date, 6,2) as month to get the months and substr(Date,0,5)='2015' for year.

```
%%sql
SELECT
    CASE
        WHEN substr(Date, 6, 2) = '01' THEN 'January'
        WHEN substr(Date, 6, 2) = '02' THEN 'February'
        WHEN substr(Date, 6, 2) = '03' THEN 'March'
        WHEN substr(Date, 6, 2) = '04' THEN 'April'
        WHEN substr(Date, 6, 2) = '05' THEN 'May'
        WHEN substr(Date, 6, 2) = '06' THEN 'June'
        WHEN substr(Date, 6, 2) = '07' THEN 'July'
        WHEN substr(Date, 6, 2) = '08' THEN 'August'
        WHEN substr(Date, 6, 2) = '09' THEN 'September'
        WHEN substr(Date, 6, 2) = '10' THEN 'October'
        WHEN substr(Date, 6, 2) = '11' THEN 'November'
        WHEN substr(Date, 6, 2) = '12' THEN 'December'
    END AS Month,
    LANDING_OUTCOME,
    Booster_Version,
    Launch_Site
FROM SPACEXTABLE
WHERE substr(Date, 0, 5) = '2015'
AND TRIM(LANDING_OUTCOME) = 'Failure (drone ship)';
```

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

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

- This query returns the Month, Landing Outcome, Booster Version and Launch site of 2015 launches where stage 1 failed to land on a drone ship.

- There were two such occurrences.

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

Task 10

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(*) AS Count
FROM SPACEXTABLE
WHERE Date BETWEEN '2010-06-04' AND '2017-03-20'
GROUP BY LANDING_OUTCOME
ORDER BY Count DESC;
```

[29]

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

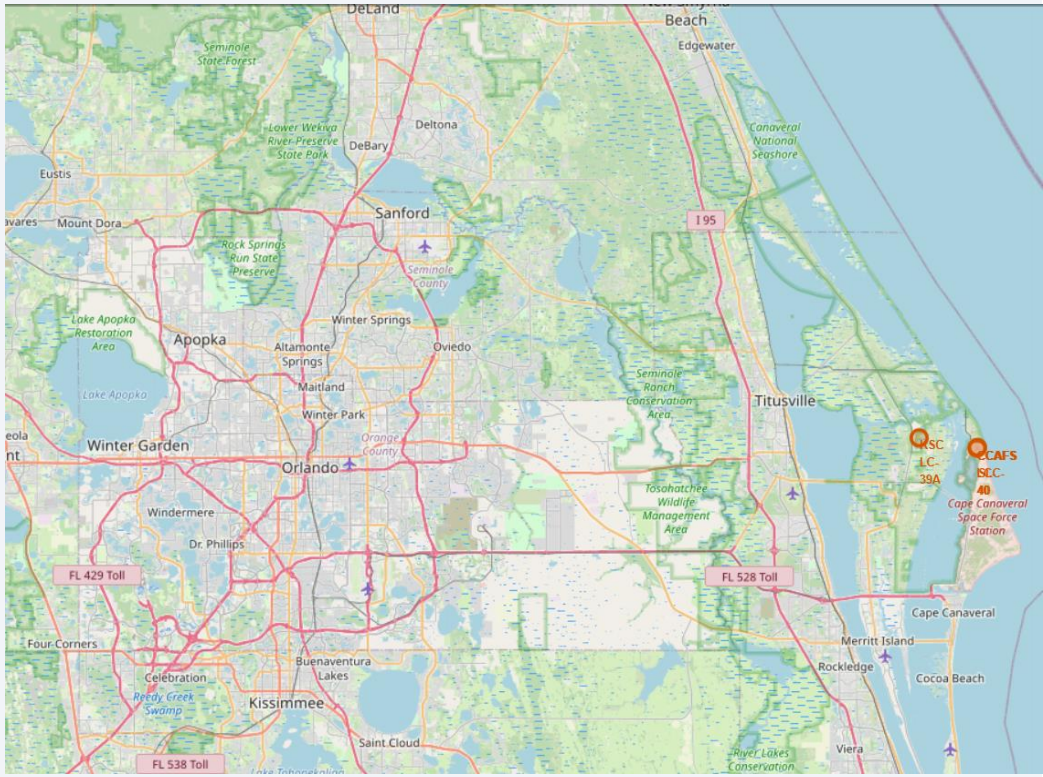
Landing_Outcome	Count
No attempt	10
Success (drone ship)	5
Failure (drone ship)	5
Success (ground pad)	3
Controlled (ocean)	3
Uncontrolled (ocean)	2
Failure (parachute)	2
Precluded (drone ship)	1

- This query returns a list of successful landings and between 2010-06-04 and 2017-03-20 inclusively.
- There are different types of landing outcomes

Section 3

Launch Sites Proximities Analysis

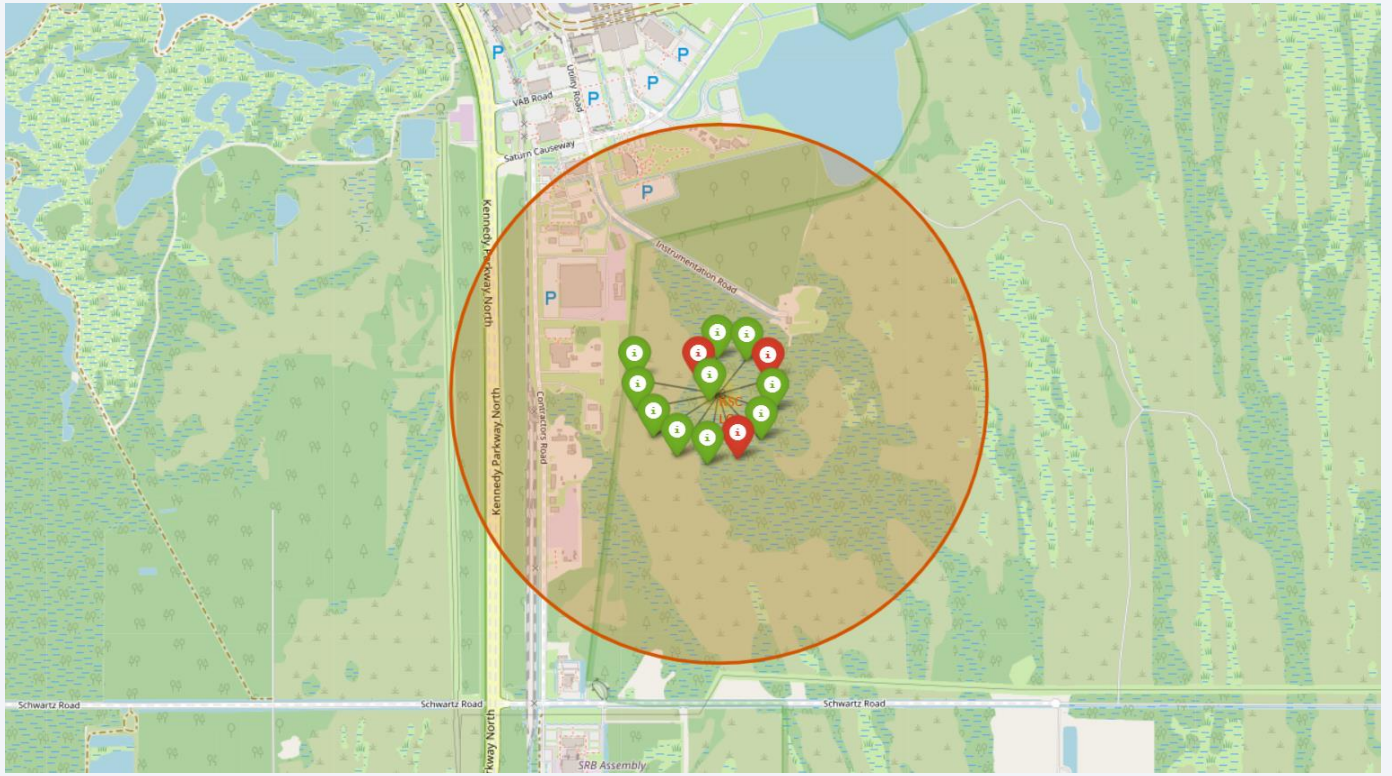
Launch Site Locations



- The left map shows all launch sites relative US map.

- The right map shows the two Florida launch sites since they are very close to each other. All launch sites are near the ocean.

Color-Coded Launch Markers

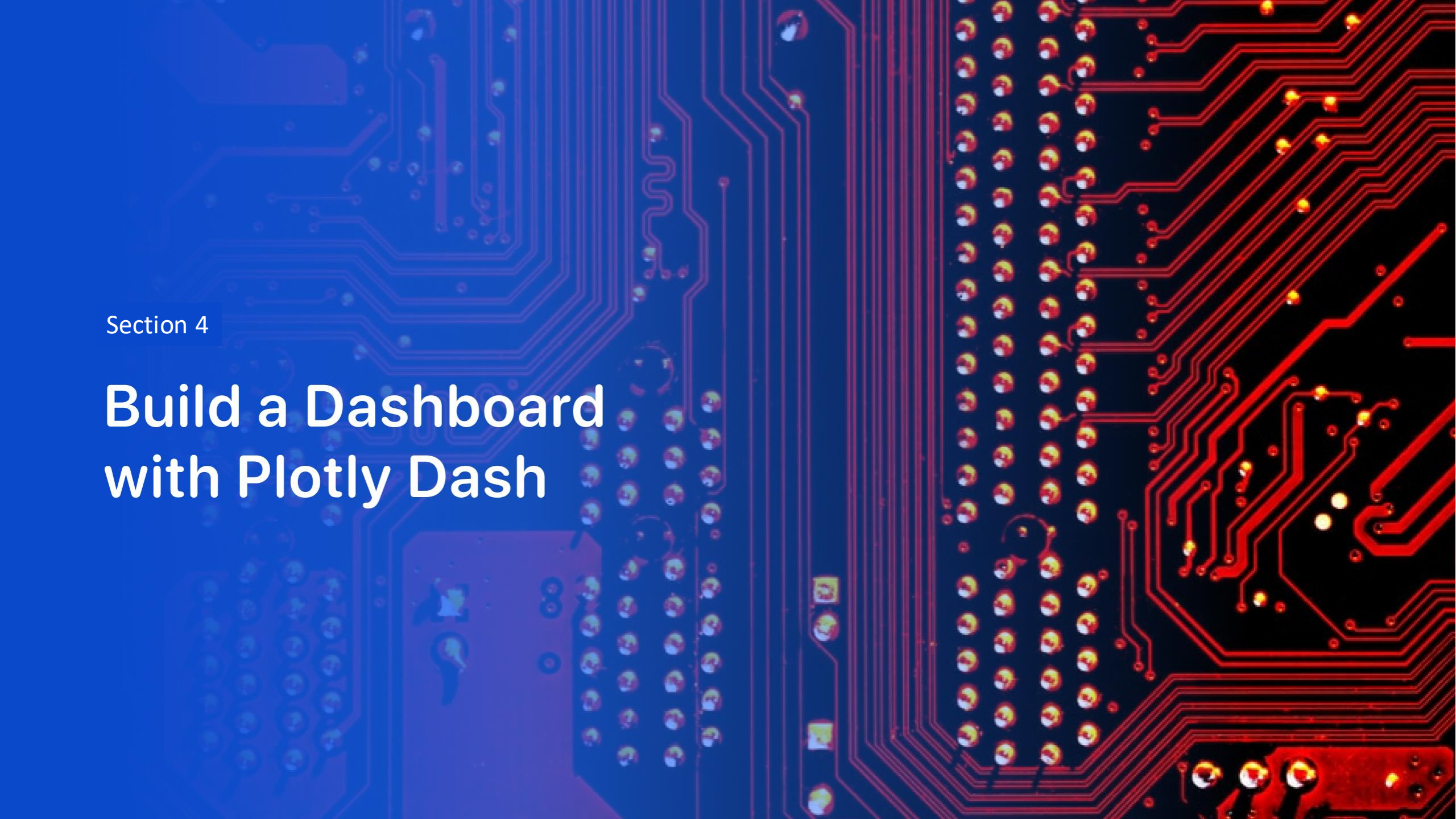


- Clusters on Folium map can be clicked on to display each successful landing (green icon) and failed
- landing (red icon). In this example KSC LC-39A shows 10 successful landings and 3 failed landings.

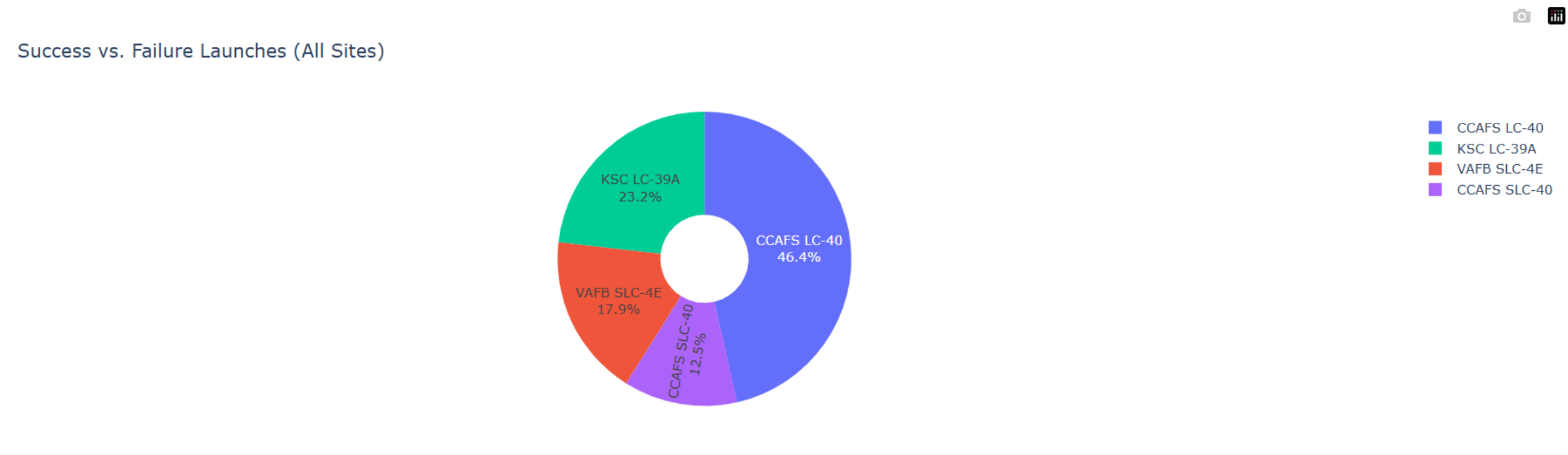
Key Location Proximities

Using CCAFS SLC-40 as an example,

- Launch sites are very close to railways (0.30 Km) for large part and supply transportation.
- Launch sites are close to highways (0.65 Km) for human and supply transport.
- Launch sites are also close to coasts (0.93 Km) and
- Relatively far from cities (23.11 Km) so that launch failures can land in the sea to avoid rockets falling on densely populated areas.

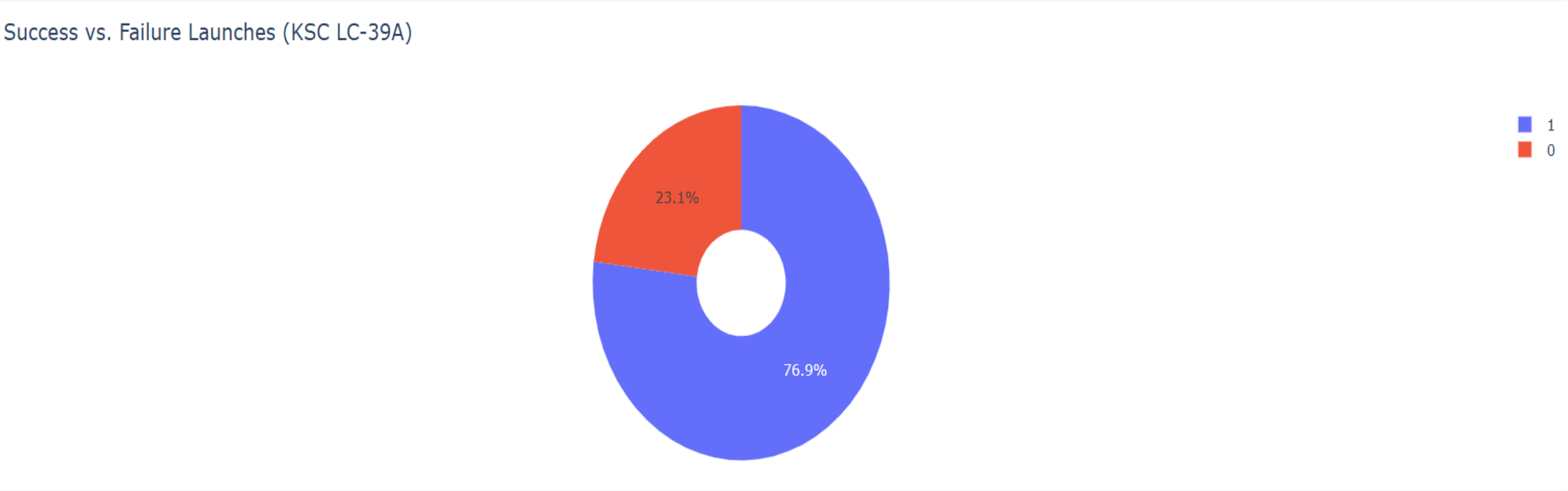

Section 4
Build a Dashboard with Plotly Dash

Successful Launches From across Launch Sites



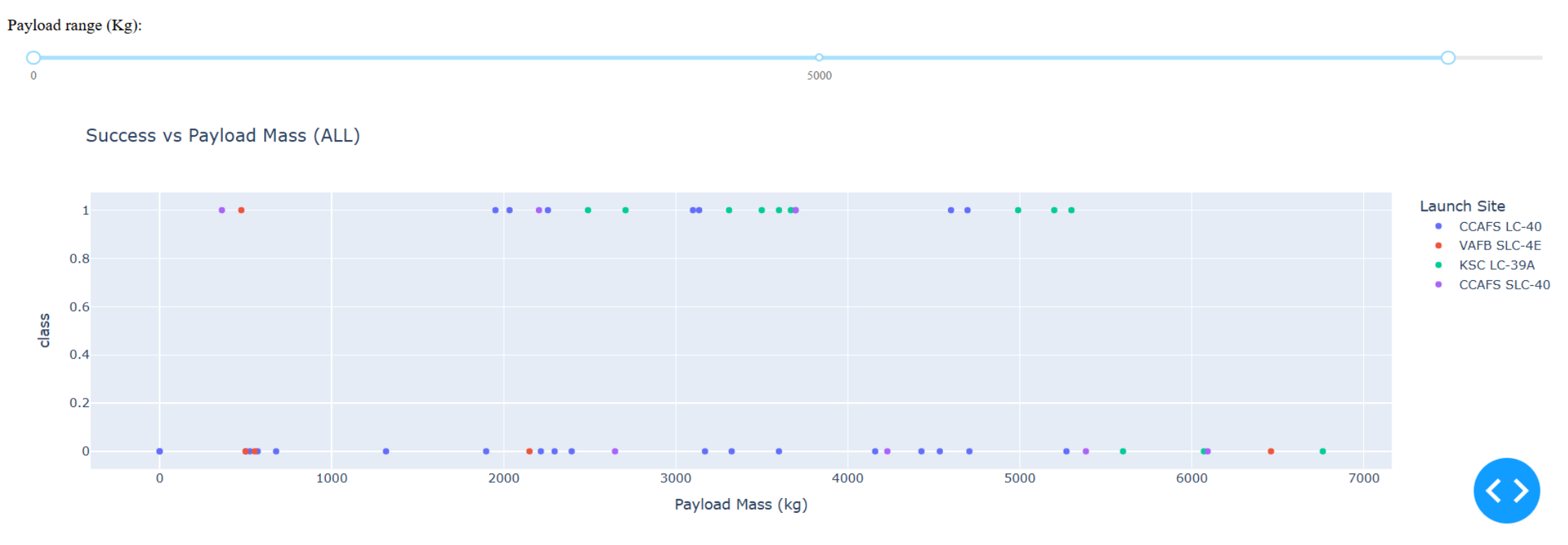
- This is the distribution of successful landings across all launch sites. CCAFS LC-40 is the old name of CCAFS SLC-40 so CCAFS has huge amount of successful landings, but a majority of the successful landings where performed before the name change.
- VAFB has the smallest share of successful landings. This may be due to smaller sample and increase in difficulty of launching in the west coast.

Highest Success Rate Launch Site



KSC LC-39A has the highest success rate with 10 successful landings and 3 failed landings.

PayLoad Mass VS Success VS Booster Version cat

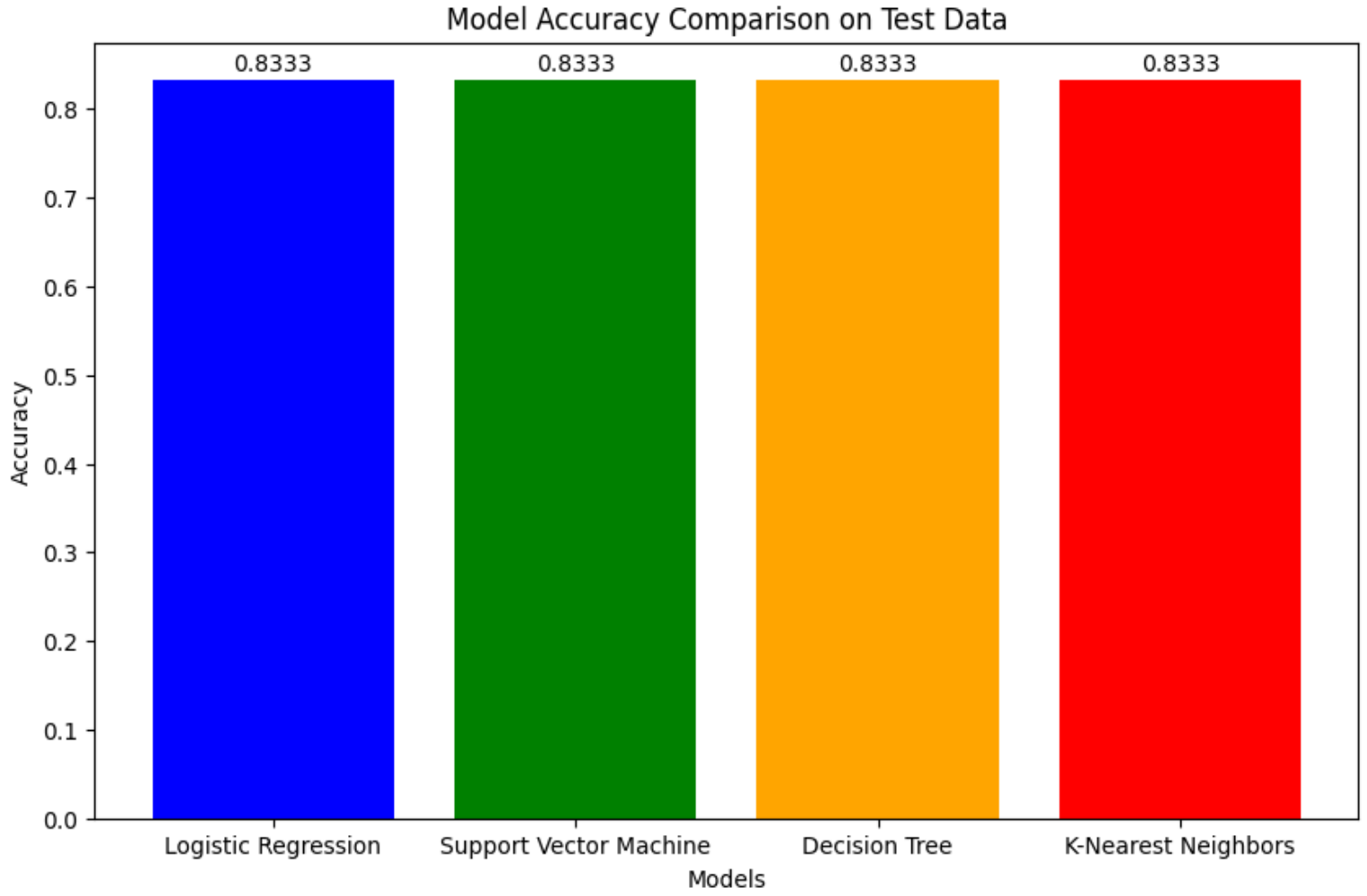


Plotly dashboard has a Payload range selector. However, this is set from 0-10000 instead of the max Payload of 15600. Class indicates 1 for successful landing and 0 for failure. Scatter plot also accounts for booster version category in color and number of launches in point size.

Section 5

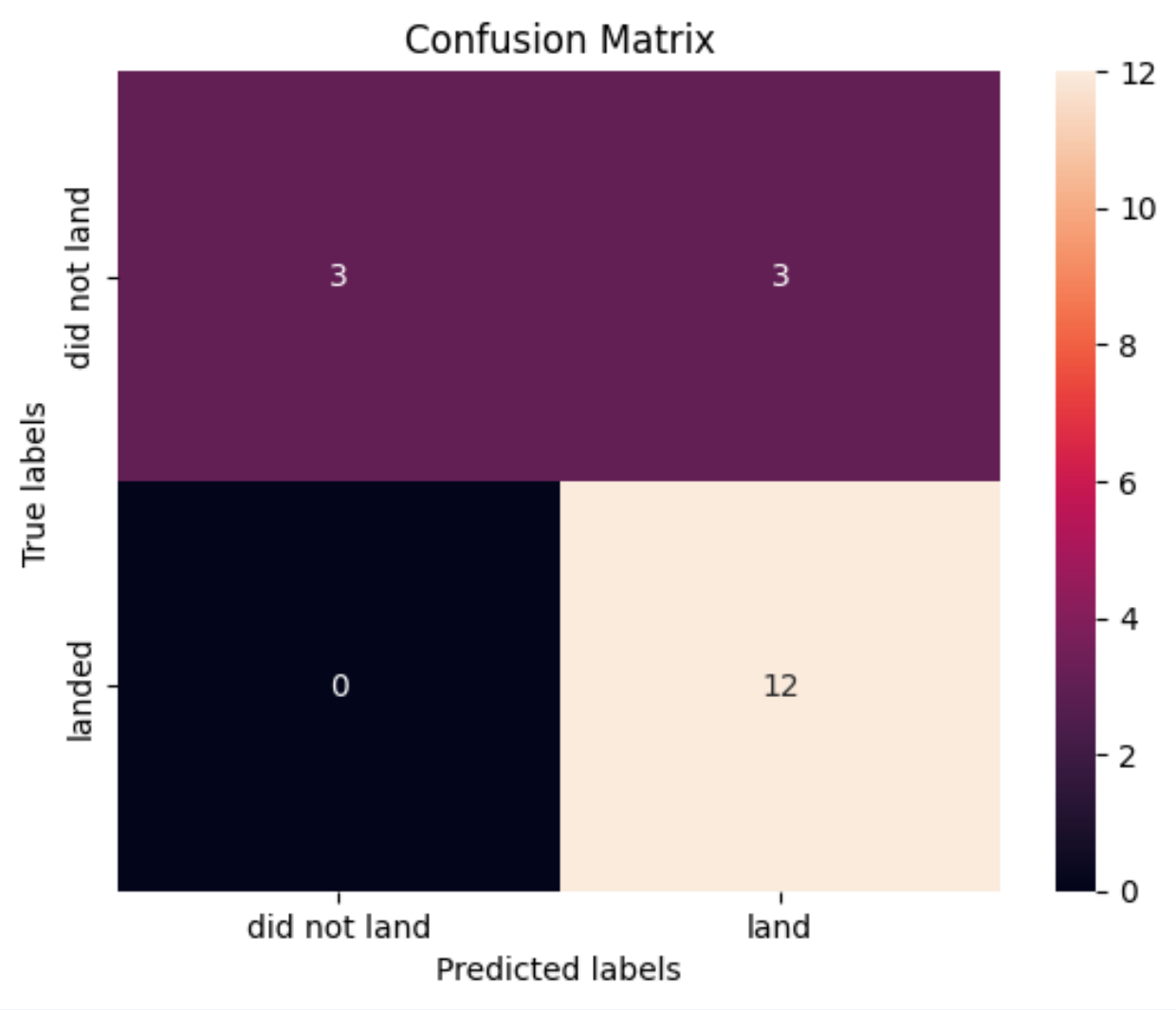
Predictive Analysis (Classification)

Classification Accuracy



- All models had virtually the same accuracy on the test set at 83.33% accuracy. It should be noted that test size is small at only sample size of 18.
- This can cause large variance in accuracy results, such as those in Decision Tree Classifier model in repeated runs.
- We likely need more data to determine the best model.

Confusion Matrix



- Since all models performed the same for the test set, the confusion matrix is the same across all models. The models predicted 12 successful landings when the true label was successful landing.
- The models predicted 3 unsuccessful landings when the true label was unsuccessful landing.
- The models predicted 3 successful landings when the true label was unsuccessful landings (false positives). Our models over predict successful landings.

Conclusions

Our task: to develop a machine learning model for Space Y who wants to bid against SpaceX

- The goal of model is to predict when Stage 1 will successfully land to save ~$100 million USD
- Used data from a public SpaceX API and web scraping SpaceX Wikipedia page
- Created data labels and stored data into a DB2 SQL database
- Created a dashboard for visualization
- We created a machine learning model with an accuracy of 83%
- Allon Mask of SpaceY can use this model to predict with relatively high accuracy whether a launch will have a successful Stage 1 landing before launch to determine whether the launch should be made or not
- If possible more data should be collected to better determine the best machine learning model and improve accuracy

Appendix

GITHUB Repo :

https://github.com/GANESH-MAHARAJ/Applied-DataScience-Capstone

Thank you!