IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Sam Hoffman

7/13/23

Outline

- Executive Summary
- Introduction
- Methodology
- Results
- Conclusion

Executive Summary

- Summary of methodologies. All work was performed using Python with Jupyter Notebooks and the noted libraries.
  - Data gathering: Data was scraped using Requests and parsed using BeautifulSoup, data was also gathered using the SpaceX Rest API.
  - Data wrangling and cleaning: Data was cleaned for outliers, improper datatypes, and other issues using Pandas and Numpy.
  - Exploratory Analysis: Conducted using SQL magic commands, Pandas, & Matplotlib.
  - Data visualizations: Visualizations were generated using Plotly Express, Dash, Seaborn, Folium, & Matplotlib.
  - Machine Learning Analysis: Conducted using Logistic Regression, Decision Trees, K-Nearest Neighbors, and Support Vector Machines from the Scikitlearn library.
- Summary of all results.
  - Several factors impact whether a flight was able to have a successful landing and recovery of the first stage.
  - We found that later flights have been more successful than earlier flights.
  - Payload appears to have little to no impact on a successful recovery of the first stage.

Introduction

Project background and context

- The space industry has seen remarkable advancements in recent years, with private companies playing a significant role in revolutionizing space exploration. One such company at the forefront of this innovation is SpaceX, led by entrepreneur Elon Musk. SpaceX's Falcon 9 rocket has gained recognition for its groundbreaking ability to land and be reused, significantly reducing the cost of space missions.
- SpaceX's Falcon 9 rocket is comprised of two stages: the first stage and the second stage. The first stage is responsible for the initial boost during launch and is designed to be reusable, while the second stage carries the payload to its intended destination. By reusing the first stage, SpaceX has successfully disrupted the traditional space industry model, where rockets were predominantly disposable, leading to astronomical costs for each launch.

Introduction

- In this capstone project, our objective is to predict the landing success of the Falcon 9 first stage. By accurately determining whether the first stage will land successfully, we can estimate the cost of each launch. This information is vital for potential competitors who wish to bid against SpaceX for rocket launches, as they need to factor in the cost savings achieved through the reuse of the first stage.

Section 1

Methodology

Methodology

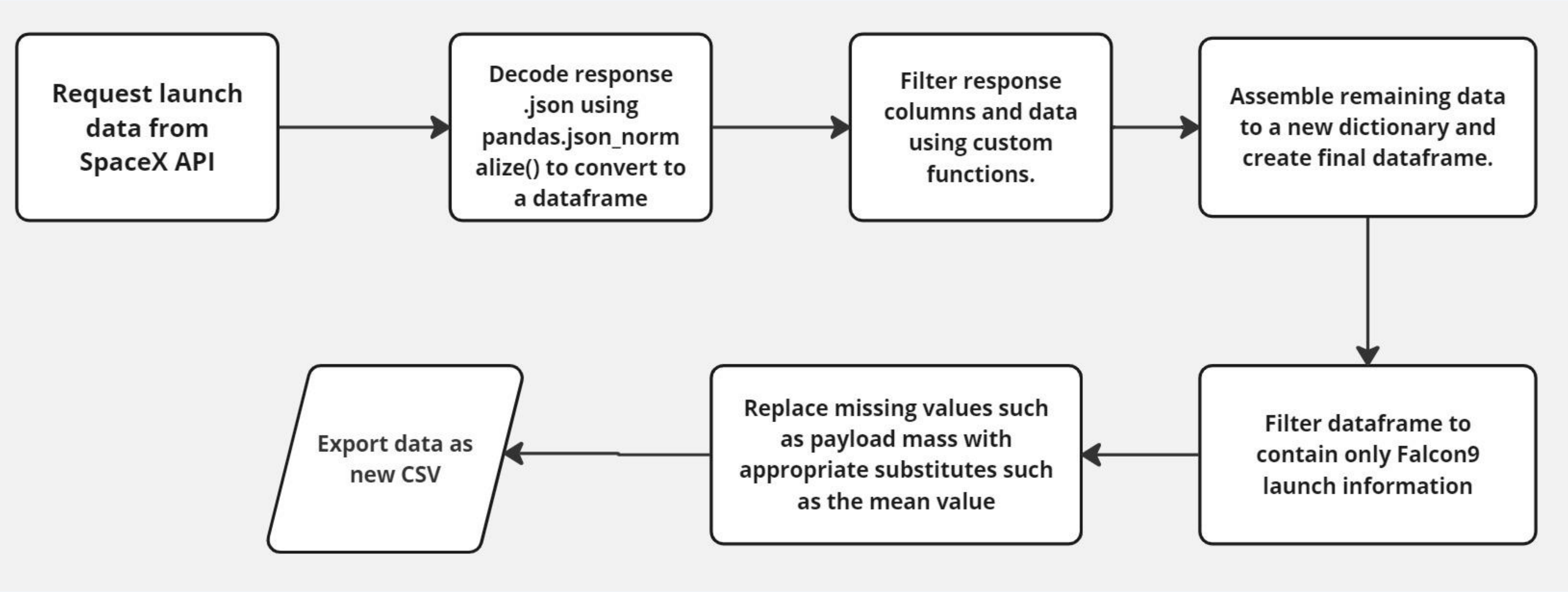
Executive Summary

- Data collection methodology:
  - Using SpaceX Rest API
  - Using web scraping from Wikipedia
- Perform data wrangling
  - Filtering the data, dealing with missing values, employing One Hot Encoding to prepare categorical data.
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - How to build, tune, evaluate classification models

Data Collection

- The data collection process involved using API requests from SpaceX's REST API and web scraping applicable data found here
- Data needed to be gathered using both methods to create a complete dataset.

Data Collection – SpaceX API



Request launch data from SpaceX API

Decode response .json using pandas.json_normalize() to convert to a dataframe

Filter response columns and data using custom functions.

Assemble remaining data to a new dictionary and create final dataframe.

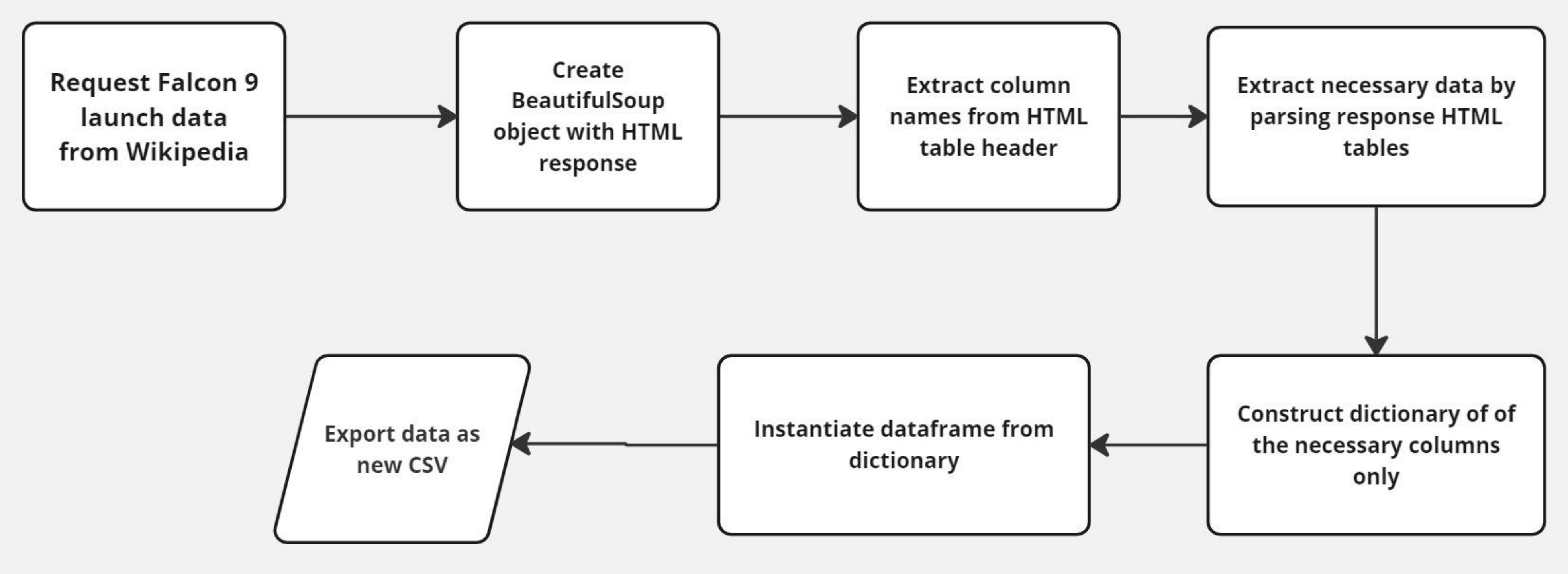
Filter dataframe to contain only Falcon9 launch information

Replace missing values such as payload mass with appropriate substitutes such as the mean value

Export data as new CSV

Notebook found here: https://github.com/hoffman1sr/IBMCapstoneProject/blob/main/jupyter-labs-spacex-data-collection-api%20(1).ipynb

Data Collection - Scraping


Request Falcon 9 launch data from Wikipedia
Create BeautifulSoup object with HTML response
Extract column names from HTML table header
Extract necessary data by parsing response HTML tables
Construct dictionary of of the necessary columns only
Instantiate dataframe from dictionary
Export data as new CSV

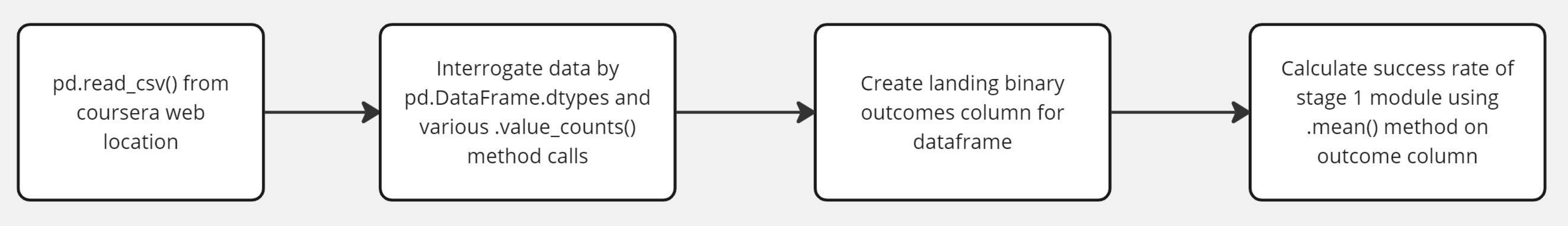
- Notebook found here: https://github.com/hoffman1sr/IBMCapstoneProject/blob/main/jupyter-labs-webscraping.ipynb

Data Wrangling

- Some initial exploratory data analysis (EDA) was performed to find things like:

1. Number of launches from the different sites
2. number of successful landings based on orbit types
3. Success rate of landing types

Notebook found here: https://github.com/hoffman1sr/IBMCapstoneProject/blob/main/labs-jupyter-spacex-data_wrangling_jupyterlite.jupyterlite.ipynb



pd.read_csv() from coursera web location

Interrogate data by pd.DataFrame.dtypes and various .value_counts() method calls

Create landing binary outcomes column for dataframe

Calculate success rate of stage 1 module using .mean() method on outcome column

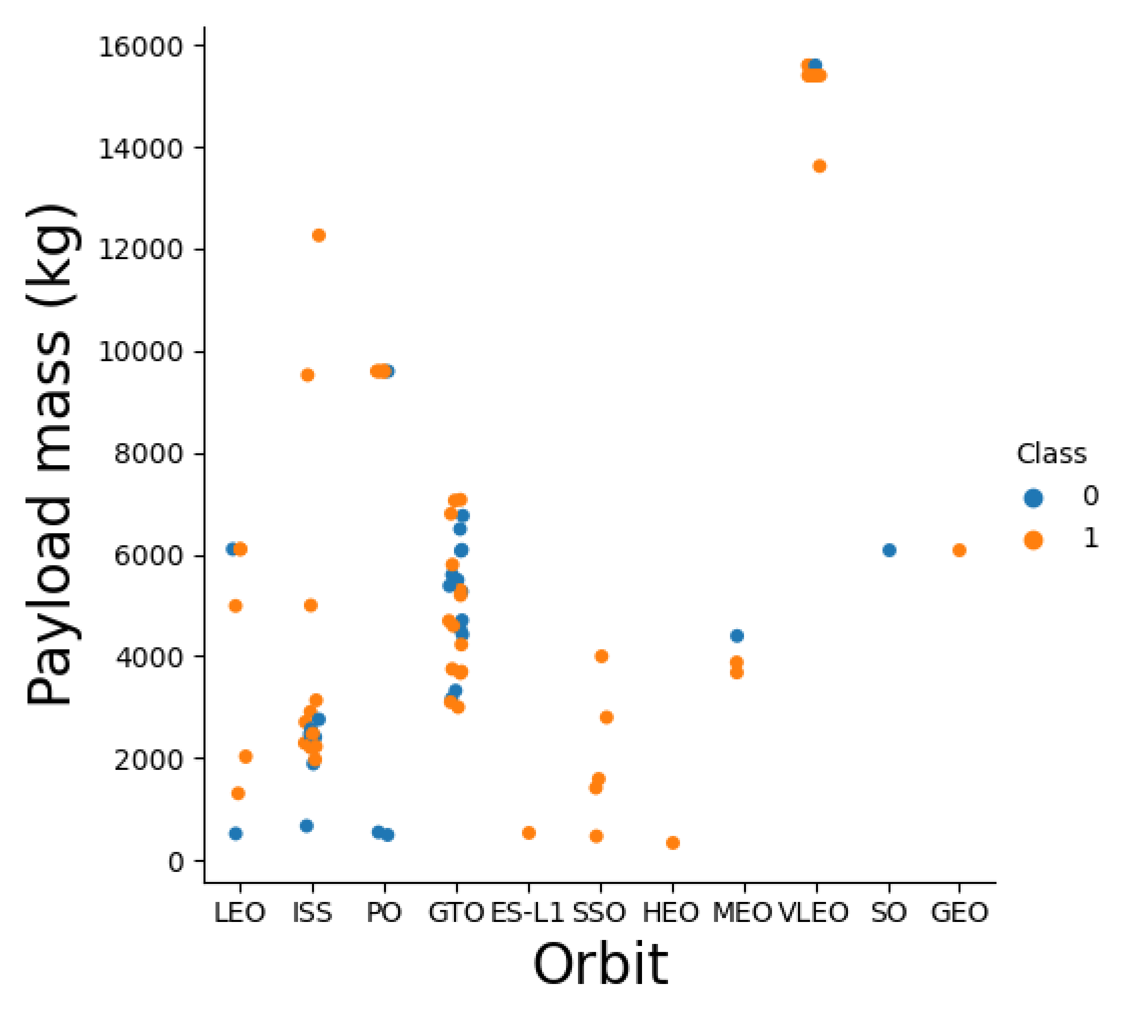
EDA with Data Visualization

Several plots were created during EDA. The plots created to investigate and visualize the relationships between 3 key flight factors:

1. Flight number (first flight, second flight, … nth flights)
2. Payload mass (kg)
3. Achieved orbit

Notebook here:

https://github.com/hoffman1sr/IBMCapstoneProject/blob/main/jupyter-labs-eda-dataviz.ipynb.jupyterlite.ipynb



EDA with SQL

Several SQL queries were performed for further EDA, including:

- Individual launch sites
- Total aggregated and average payload mass launched by key characteristics
- Ranking landing outcomes by number of occurrences as shown here:

```
%sql select Landing_Outcome, count(*) as counts from SPACEXTBL where DATE BETWEEN '04-06-2010' AND '20-03-2017’ group by Landing_Outcome order by counts desc;
```

Notebook available here:

https://github.com/hoffman1sr/IBMCapstoneProject/blob/main/jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

Folium was used for geospatial visualization.

- Markers, circles, polylines, and clusters were all used to mark points of interest.
- Markers were used to indicate point locations like launch sites.
- Clusters were used to indicate groups of events like the multiple launches which may occur from a single site over time.
- Polylines were used for calculating and approximating rough distances visually to key sites like the ocean, roadways, and railways.

Notebook here:

https://github.com/hoffman1sr/IBMCapstoneProject/blob/main/lab_jupyter_launch_site_location.jupyterlite.ipynb

Build a Dashboard with Plotly Dash

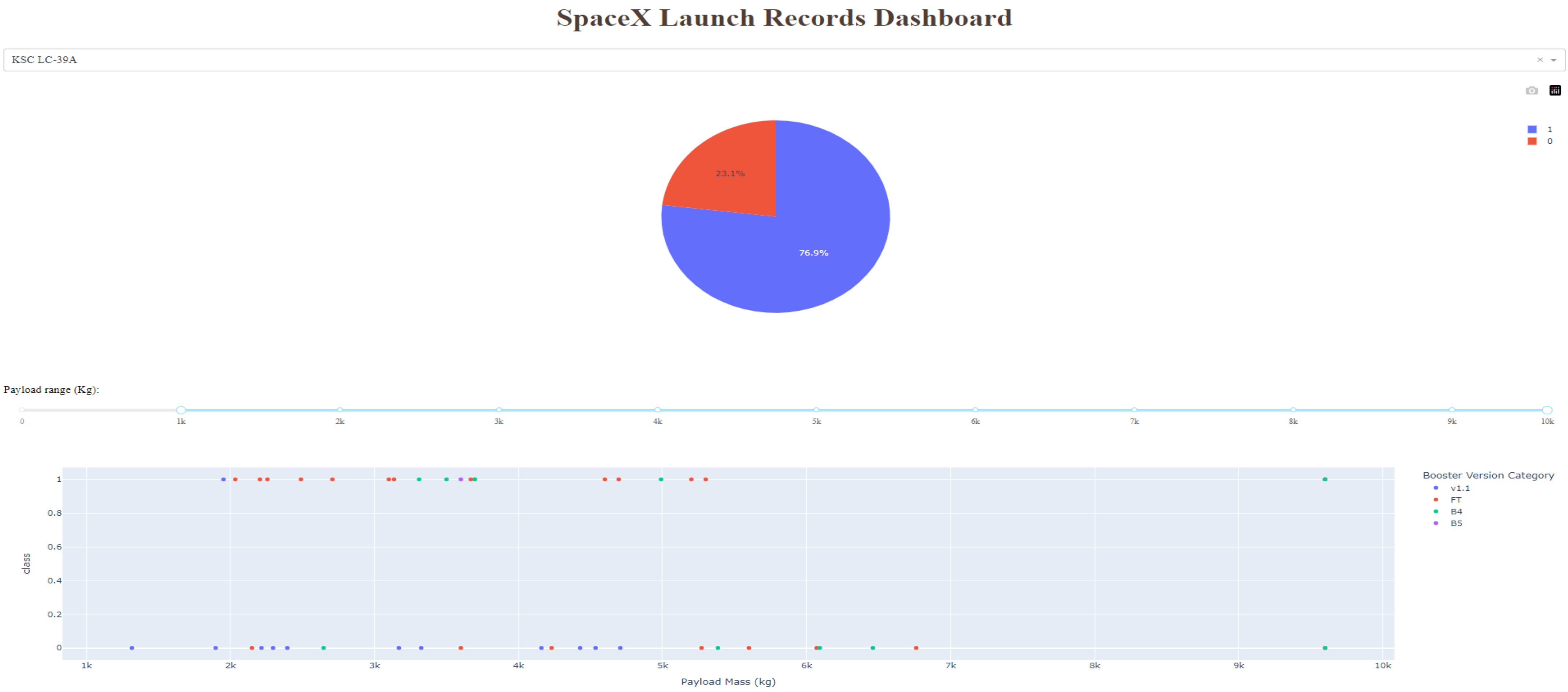
- A Plotly Dash dashboard was created to explore and present results. Two main plots were used:

  1. A pie chart with a drop down menu to select launch locations. The chart displays the proportion of launches which were either successful landings (class = 1) or not by each launch location.

  2. A scatter plot which allows us to visualize the relationship between successful landings, payload mass, and booster version. The mass displayed can be modified using a slider above the plot.

- Add the GitHub URL of your completed Plotly Dash lab, as an external reference and peer-review purpose

Build a Dashboard with Plotly Dash



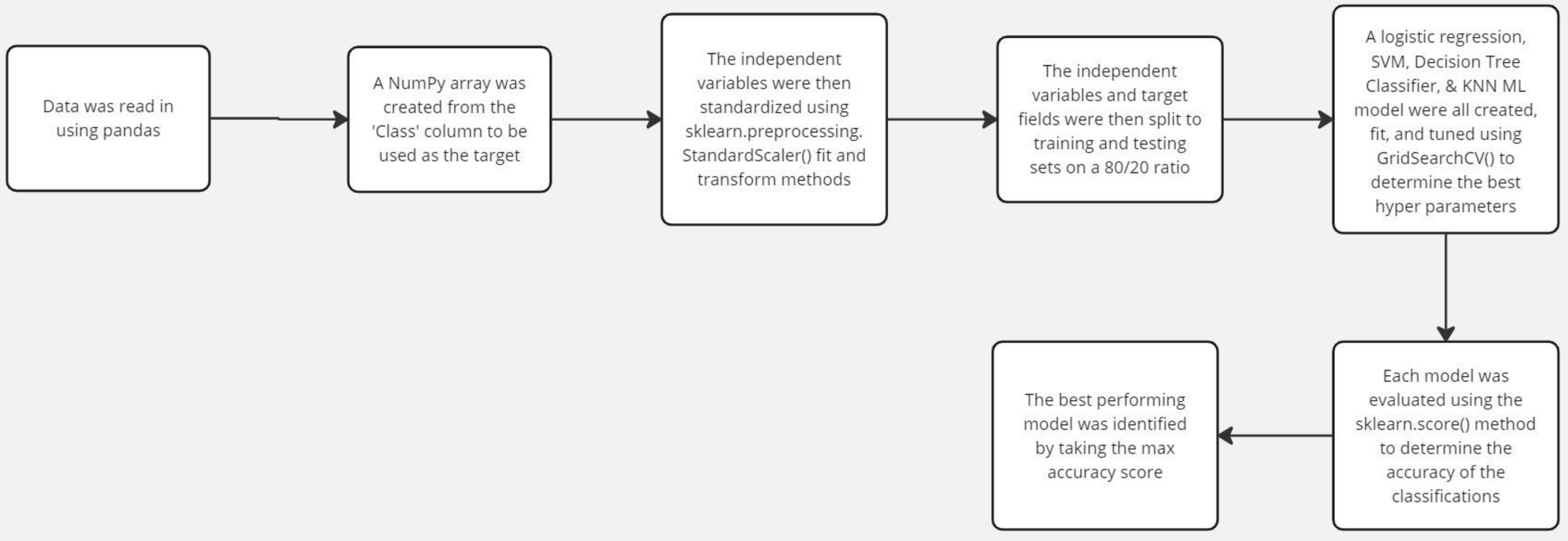
SpaceX Launch Records Dashboard

KSC LC-39A

Payload range (Kg):

Predictive Analysis (Classification)

- Several ML classification models were built and tested to determine how to best accurately predict successful landing outcomes on the data.



Data was read in using pandas

A NumPy array was created from the 'Class' column to be used as the target

The independent variables were then standardized using sklearn.preprocessing. StandardScaler() fit and transform methods

The independent variables and target fields were then split to training and testing sets on a 80/20 ratio

A logistic regression, SVM, Decision Tree Classifier, & KNN ML model were all created, fit, and tuned using GridSearchCV() to determine the best hyper parameters

Each model was evaluated using the sklearn.score() method to determine the accuracy of the classifications

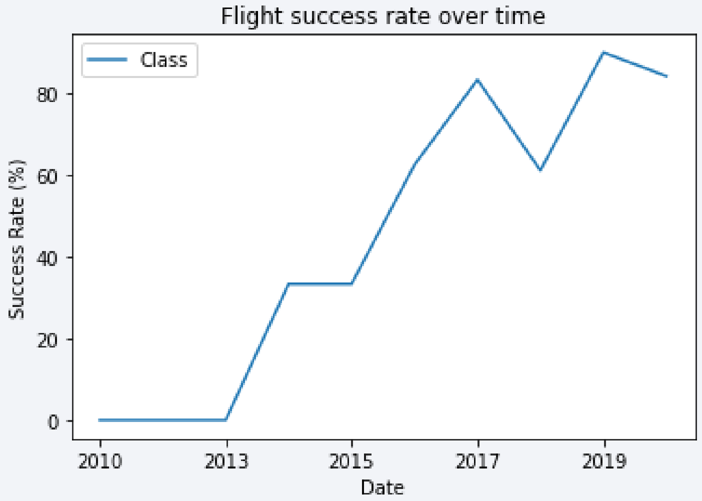
The best performing model was identified by taking the max accuracy score

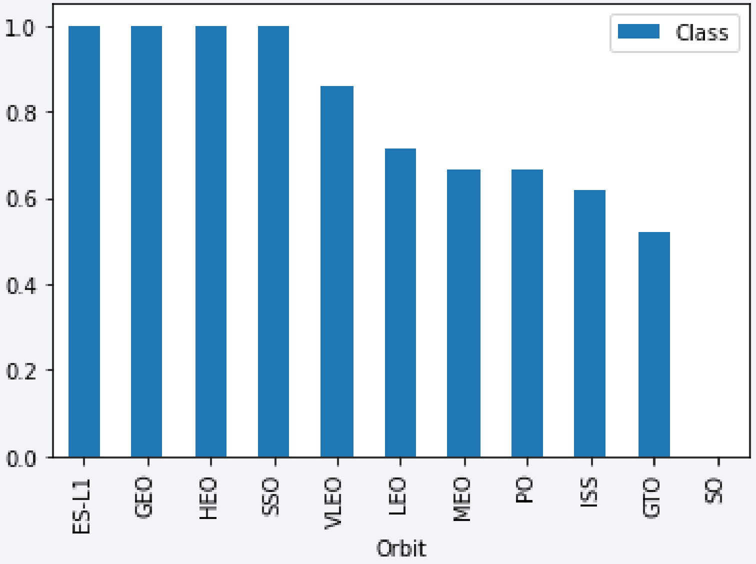
Notebook available here: https://github.com/hoffman1sr/IBMCapstoneProject/blob/main/SpaceX_Machine_Learning_Prediction_Part_5.jupyterlite.ipynb

Results

Several results presented during exploratory data analysis:

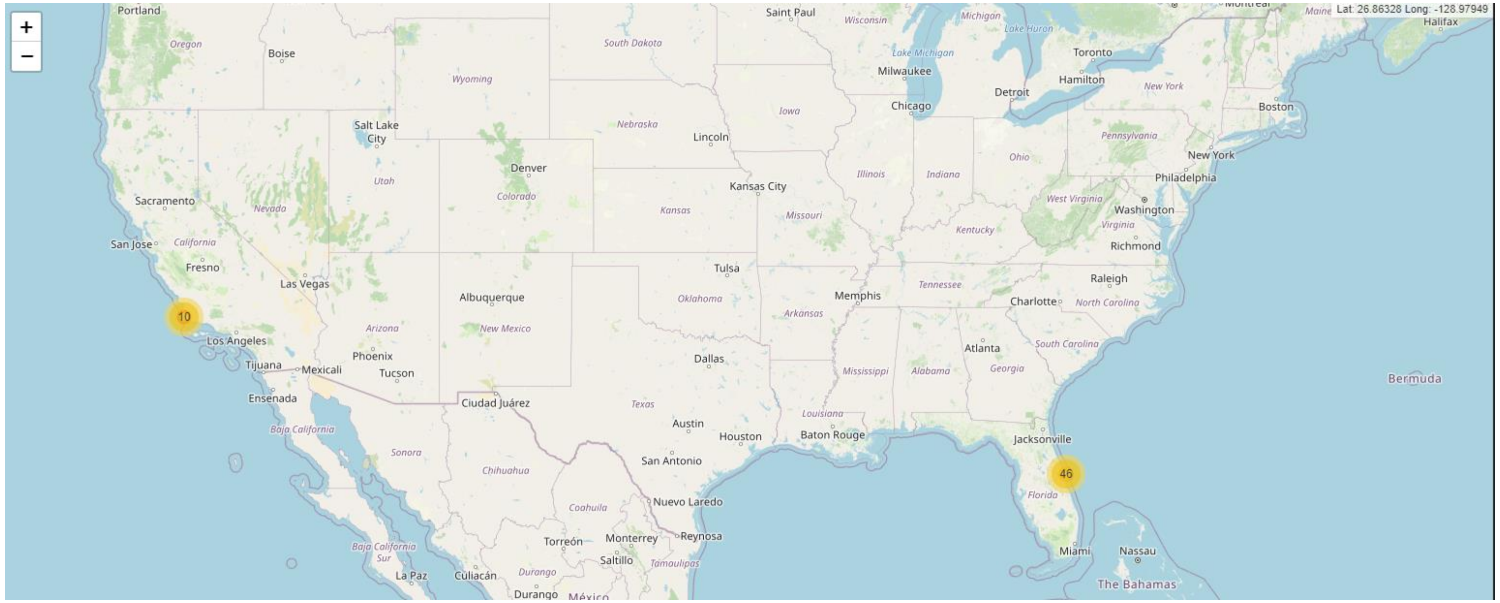
- Space X has used a total of three unique launch sites
- Payload mass on flights with the Falcon 9 booster has ranged from 350 to 15,600 kg
- There is a negative correlation between the payload mass and the likelihood of the first stage to return
- Flight success rate has increased over time
- ES-L1, GEO, HEO, & SSO orbits have a 100% success rate





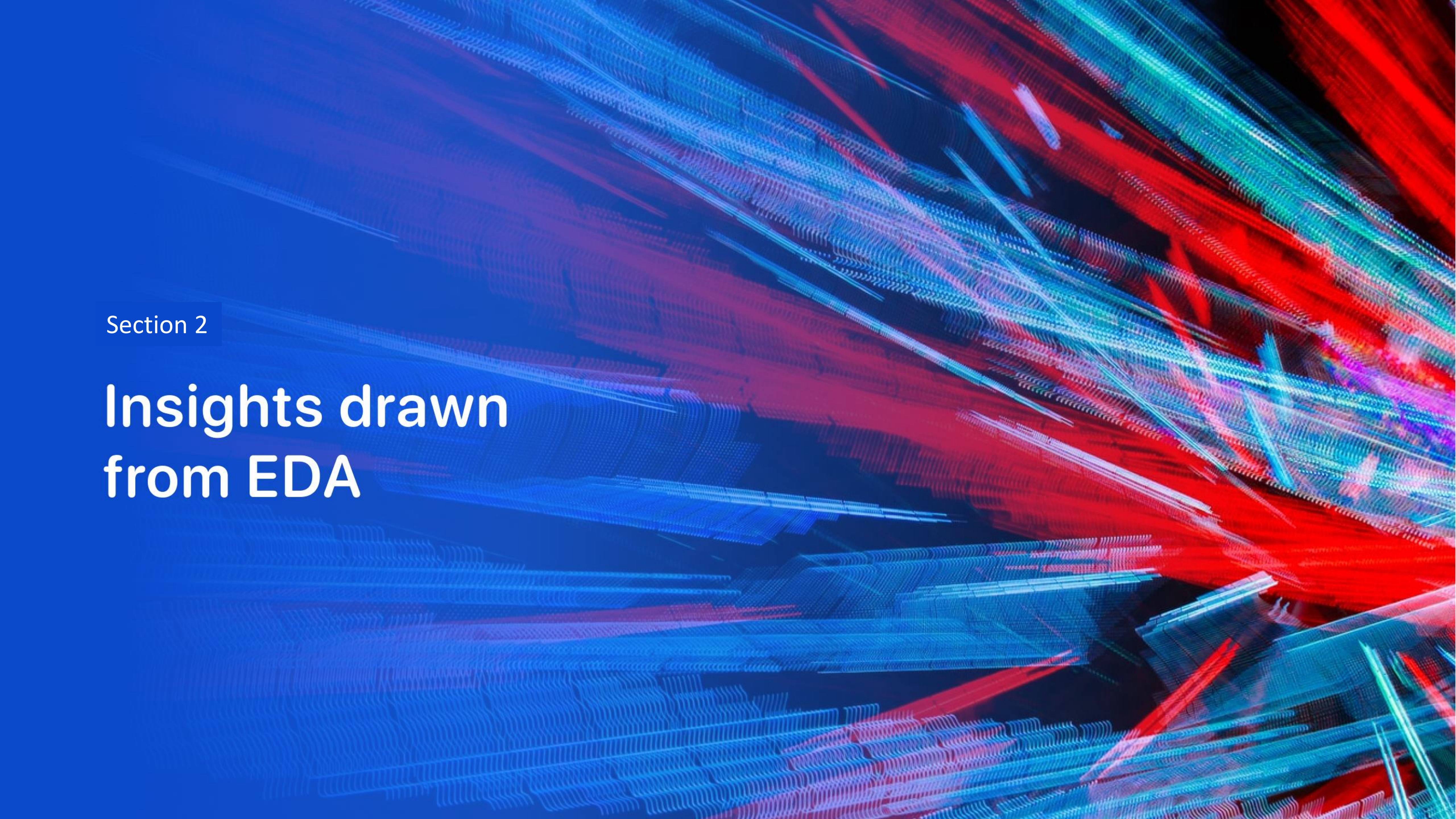
Results

- Flights were launched from the states of Florida & California.
- Majority of flights (46) were launched in Florida across 3 sites, with only 10 in California.
- All flights were near the coast.



Section 2

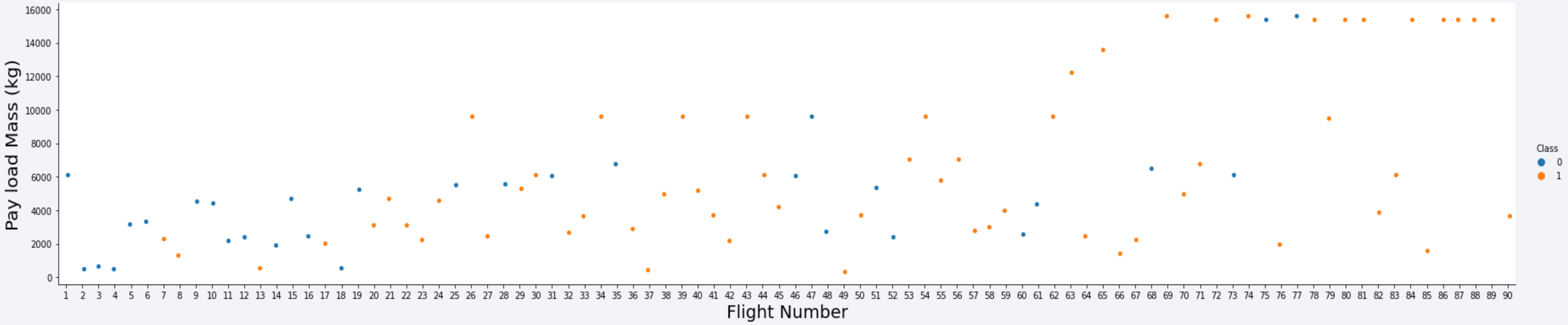
Insights drawn from EDA



Flight Number vs. Launch Site

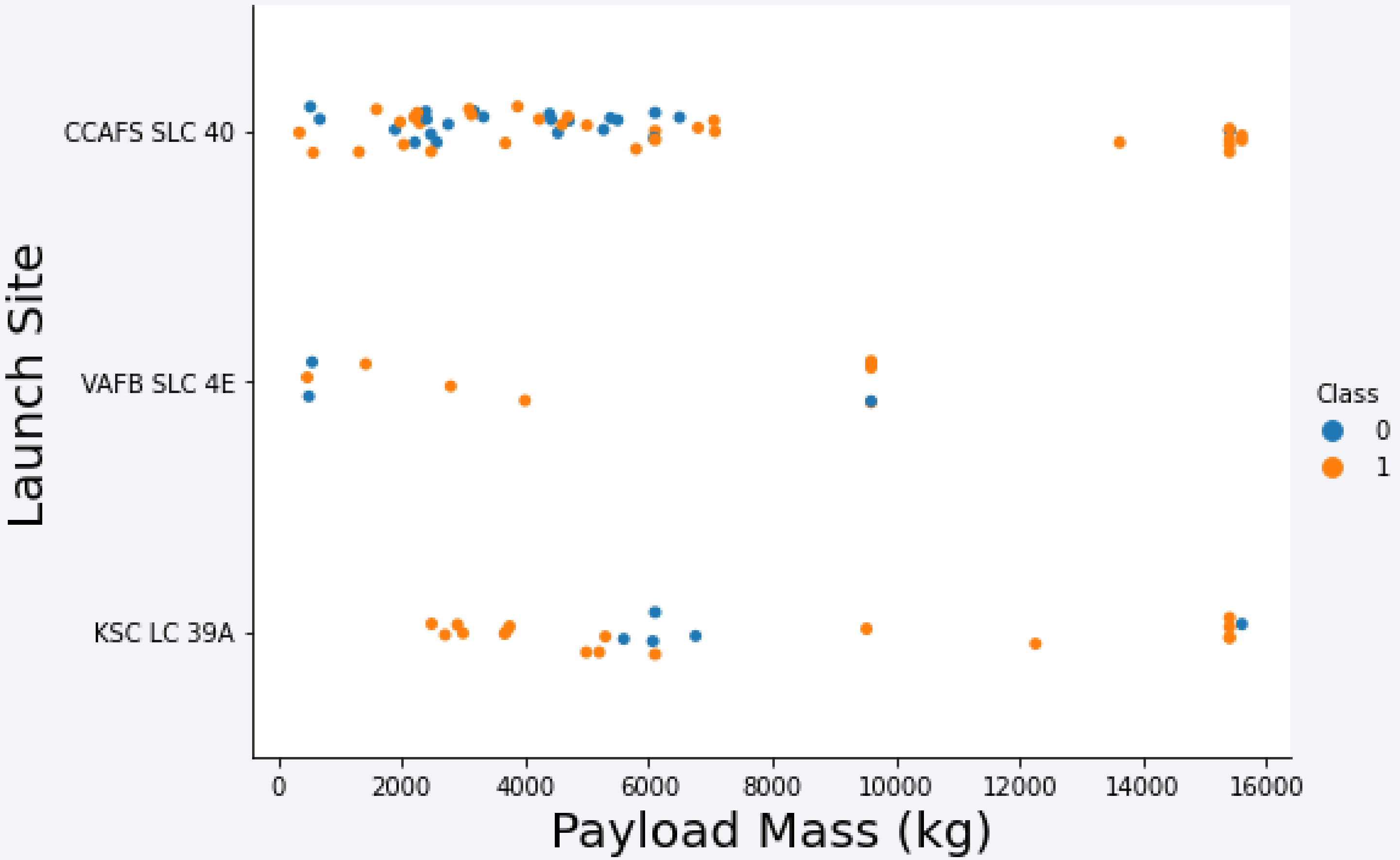
The figure here displays a scatter plot of flight number vs Payload. A few interesting things to note here:

1. There is a positive correlation between mass and flight number over time
2. First stage landings have become more successful (Class 1) over time



Payload vs. Launch Site

- There does not appear to be any strong correlation between launch site and successful stage 1 landings.
- However, it can be observed that more massive launches do not take place at the VAFB SLC 4E launch site.

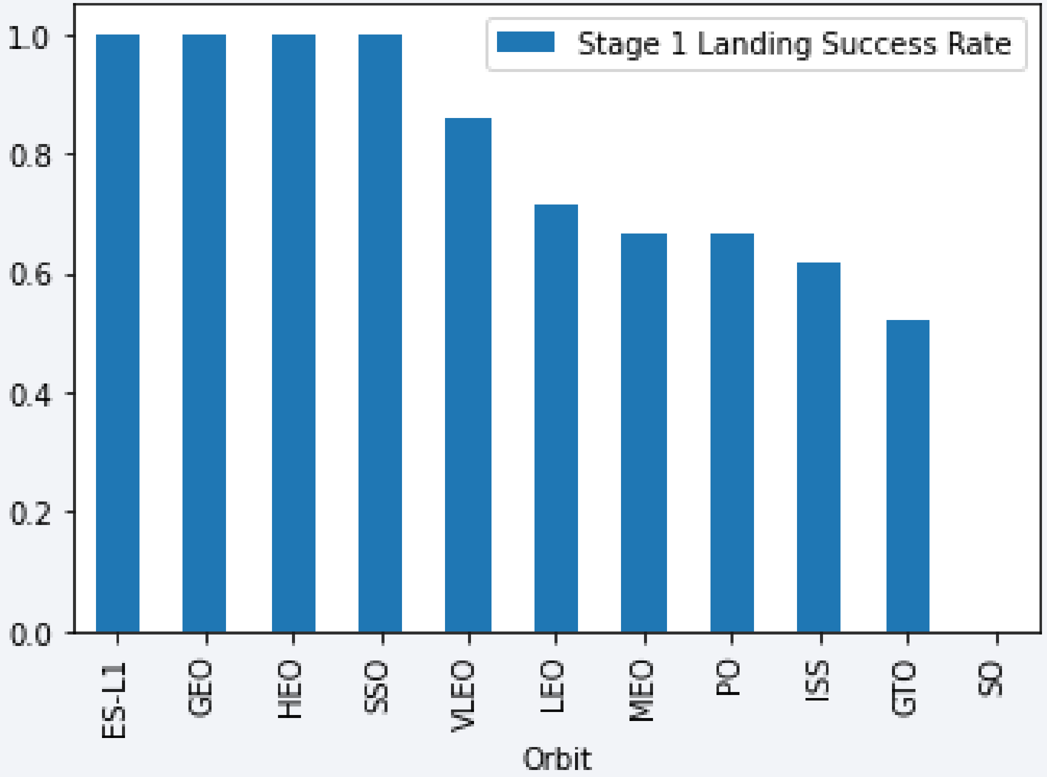


Success Rate vs. Orbit Type

- There are 4 orbit types which SpaceX has had a 100% first stage landing success rate with:

1. ES-L1
2. GEO
3. HEO
4. SSO

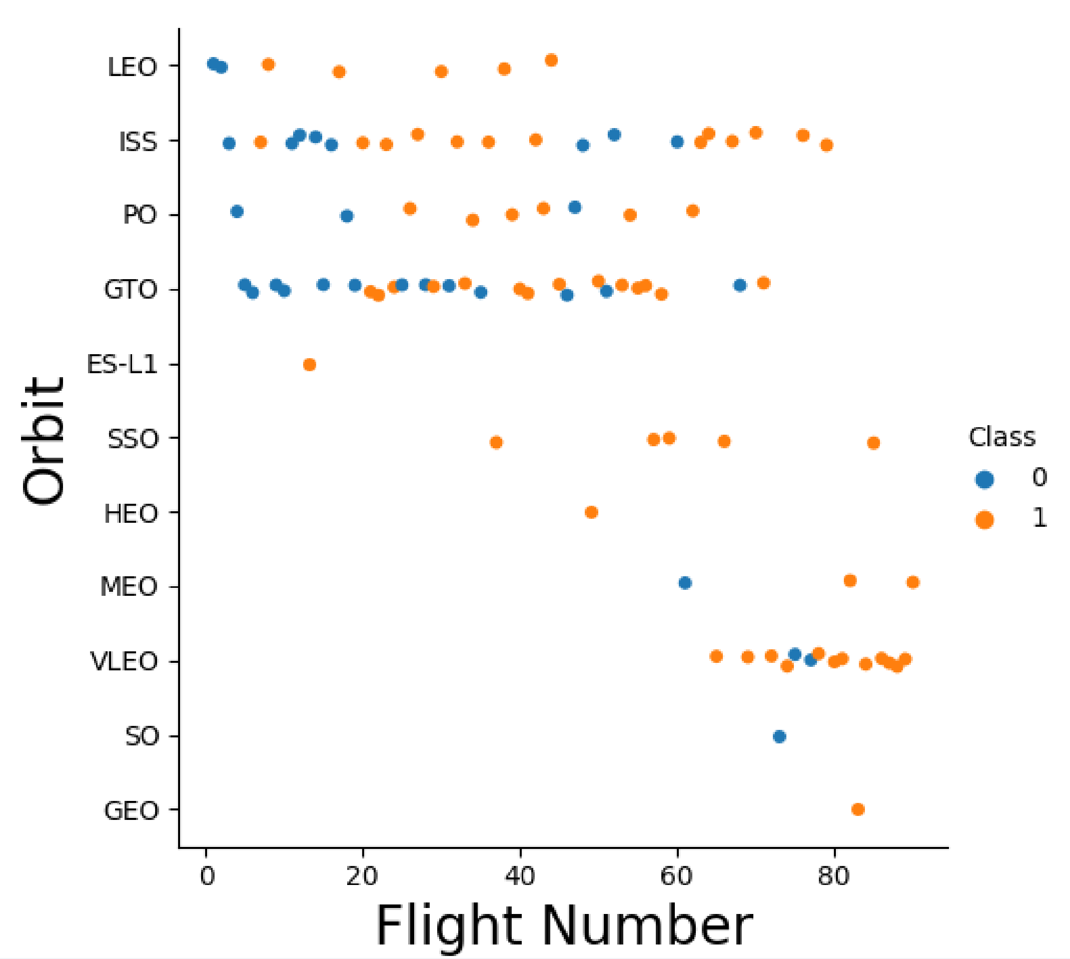
- The SO orbit has not had a successful landing, though there have only been 4 flights to the SO orbit.



Flight Number vs. Orbit Type

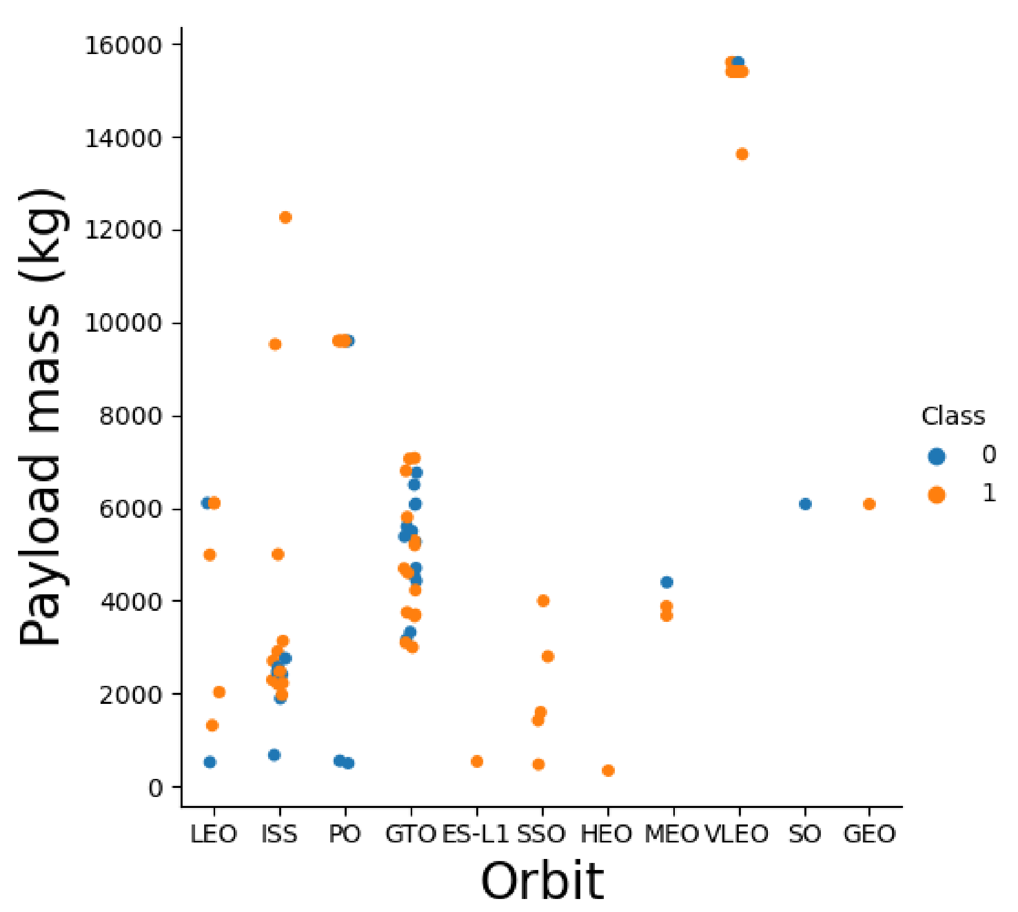
- Most flights, especially earlier flights have been to: LEO, ISS, PO, & GTO orbits.

- It is difficult to say if the high success rates of SSO, ES-L1, HEO, and GEO orbits noted on the last slide are related to the orbit, or being later flight numbers which have higher success rates.



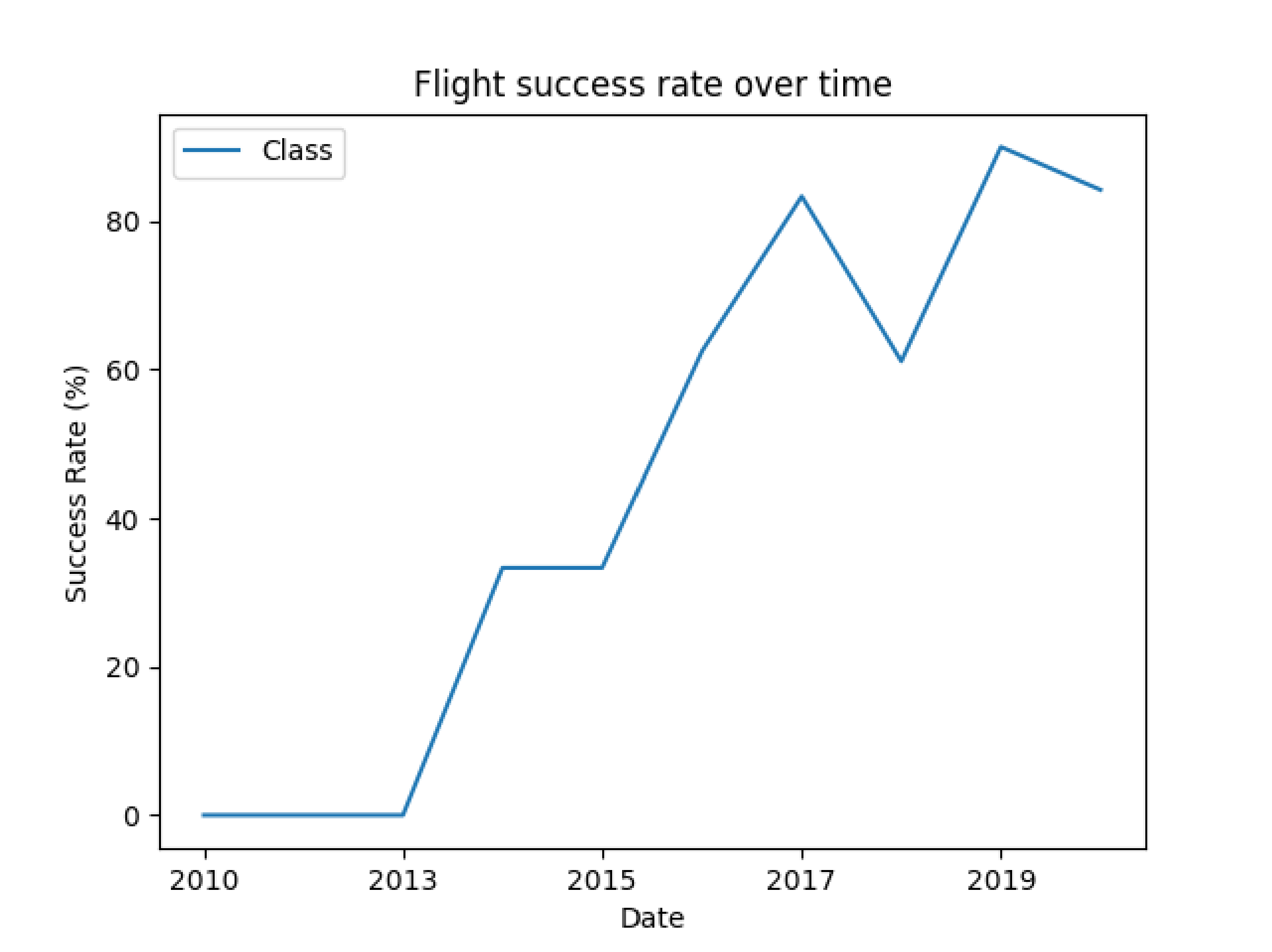
Payload vs. Orbit Type

- SO, Polar, LEO, and ISS Orbits have been more successful with heavier payloads.
- For GTO, payload seems to have no / little correlation between payload and landing success.



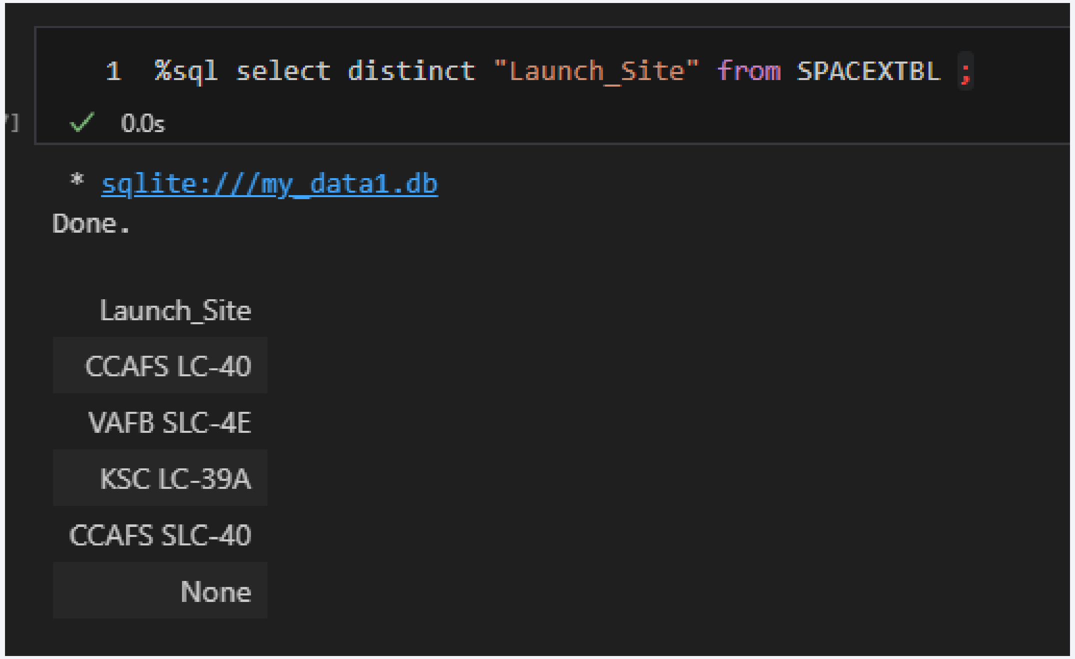
Launch Success Yearly Trend

- Success rates of each flight has increased over time.
- Notably there was not a successful Falcon 9 first stage landing from 2010 – 2013.
- Rate increased steadily with some variation from 2013-2020.



All Launch Site Names

- 4 total launch sites were used.
- SQL EDA code shown here. All query's performed using SQL magic commands within a Jupyter notebook environment.



```
%sql select distinct "Launch_Site" from SPACEXTBL ;
```

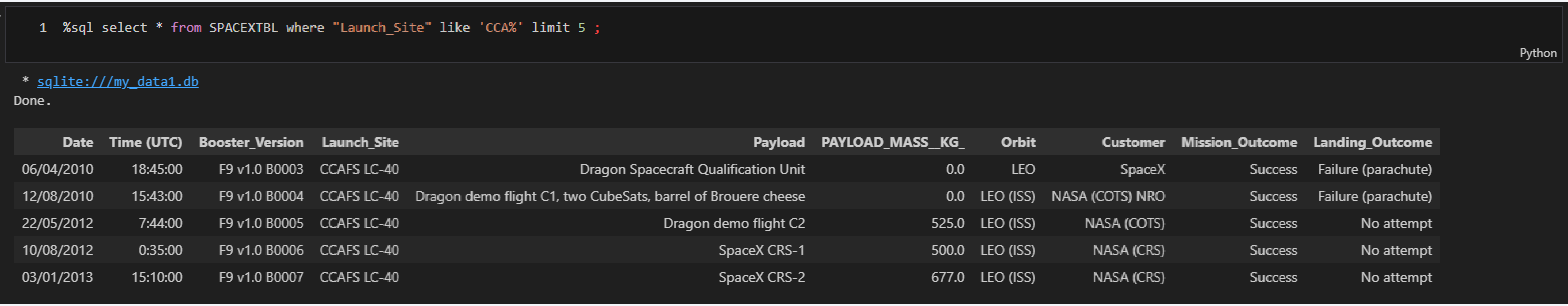
✓ 0.0s

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

Launch_Site
CCAFS LC-40
VAFB SLC-4E
KSC LC-39A
CCAFS SLC-40
None

Launch Site Names Begin with 'CCA'

- The first 5 flights from either CCAFS – LC40 or CCAFS – SLC40 occurred from the LC-40 site.
    - Query shown below.



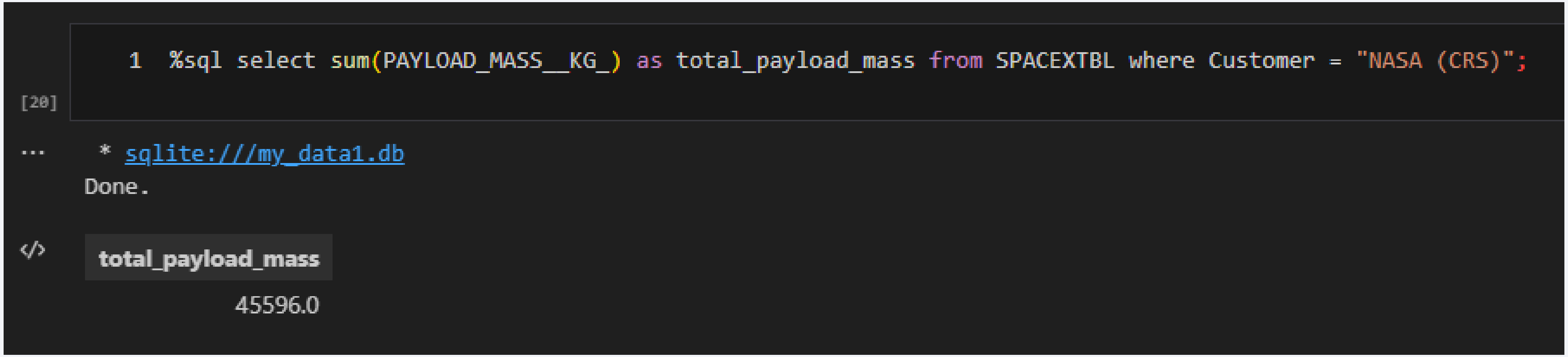
```
%sql select * from SPACEXTBL where "Launch_Site" like 'CCA%' limit 5 ;
```

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

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
06/04/2010	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0.0	LEO	SpaceX	Success	Failure (parachute)
12/08/2010	15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight C1, two CubeSats, barrel of Brouere cheese	0.0	LEO (ISS)	NASA (COTS) NRO	Success	Failure (parachute)
22/05/2012	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525.0	LEO (ISS)	NASA (COTS)	Success	No attempt
10/08/2012	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500.0	LEO (ISS)	NASA (CRS)	Success	No attempt
03/01/2013	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677.0	LEO (ISS)	NASA (CRS)	Success	No attempt

Total Payload Mass

- A total payload mass of 45,596 kgs have been carried for NASA specific flights.
- Sum() is used to aggregate the data identified using the where clause.



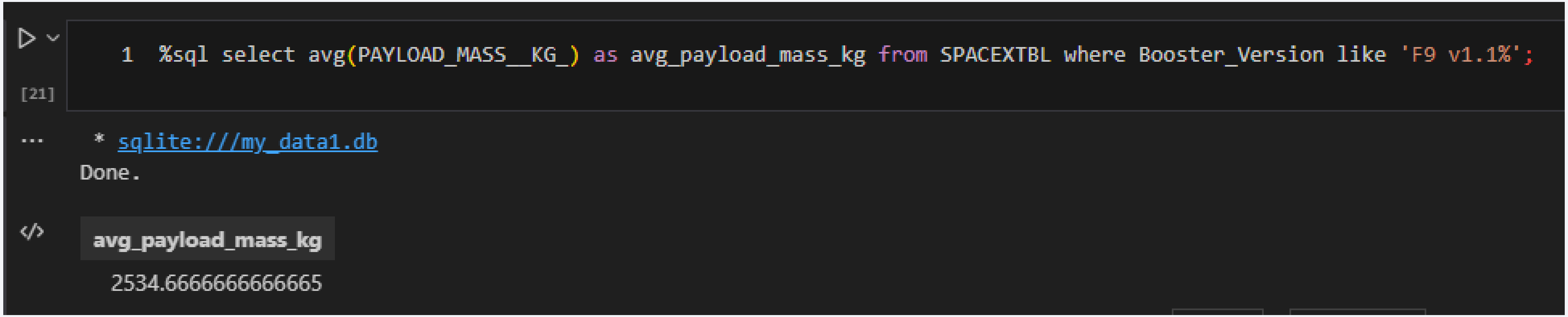
```
%sql select sum(PAYLOAD_MASS__KG_) as total_payload_mass from SPACEXTBL where Customer = "NASA (CRS)";
```

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

total_payload_mass
45596.0

Average Payload Mass by F9 v1.1

- The Falcon9 v1.1 booster had an average payload mass of approximately 2535 kg / flight.
- ‘Like’ is applied with a wild card to identify all booster version names which start with “F9 v1.1” regardless of following string characters.



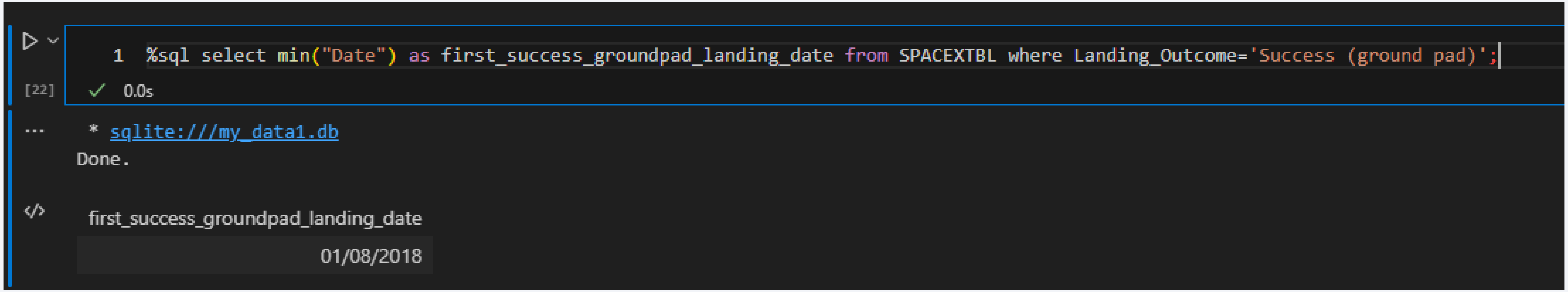
```
%sql select avg(PAYLOAD_MASS__KG_) as avg_payload_mass_kg from SPACEXTBL where Booster_Version like 'F9 v1.1%';
```

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

avg_payload_mass_kg
2534.6666666666665

First Successful Ground Landing Date

- The first successful SpaceX ground pad landing occurred on January 8th, 2018
- Min() function used to identify the earliest date identified within the data matching the where clause filter of successful ground pad landings.



```
%sql select min("Date") as first_success_groundpad_landing_date from SPACEXTBL where Landing_Outcome='Success (ground pad)';
```

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

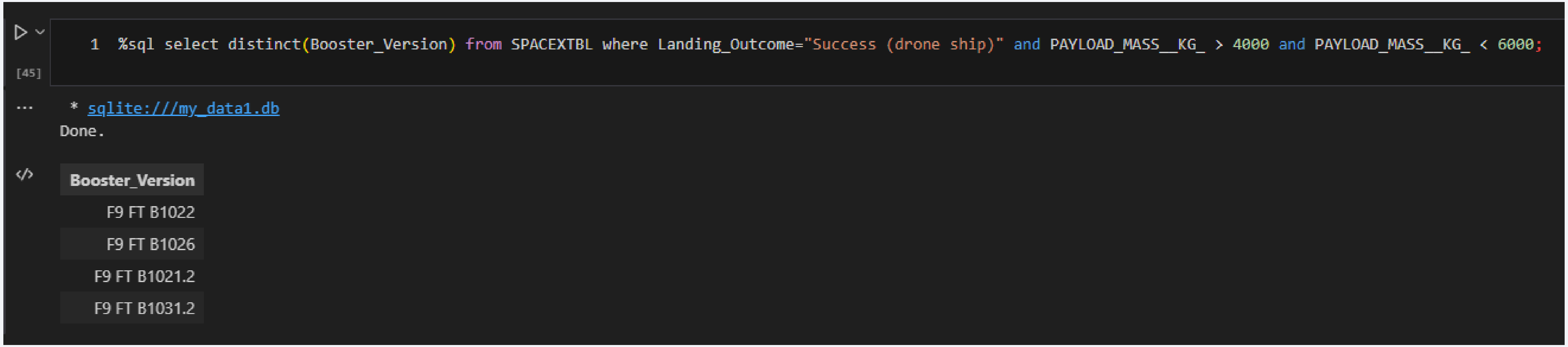
first_success_groundpad_landing_date
01/08/2018

Successful Drone Ship Landing with Payload between 4000 and 6000

- 4 boosters have had successful drone ship landings with a payload mass of:

  4000kg < payload mass < 6000kg

- Distinct() used to identify only the booster names which have had successful drone ship landings and meet the pay load mass requirements.



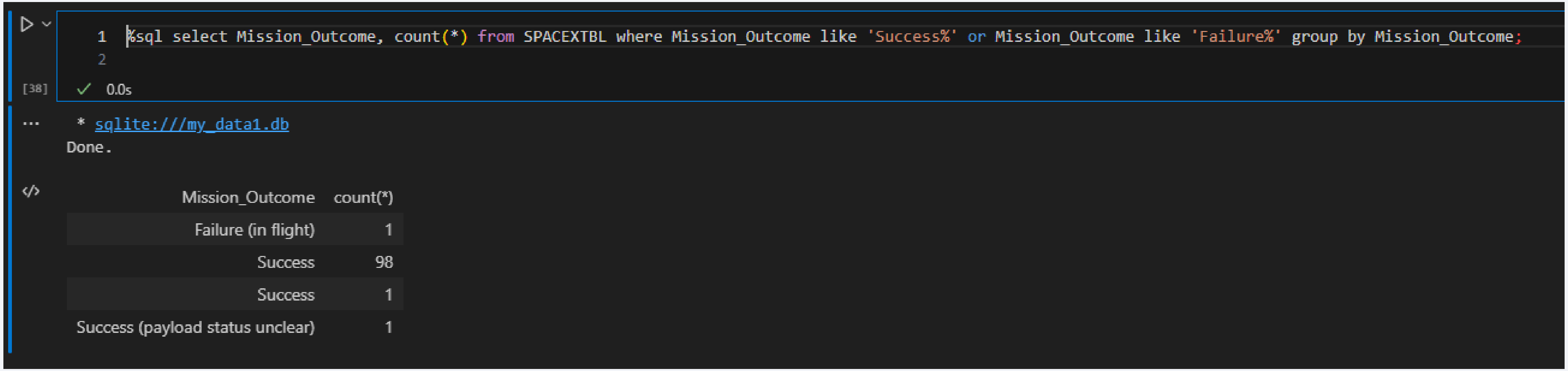
```
%sql select distinct(Booster_Version) from SPACEXTBL where Landing_Outcome="Success (drone ship)" and PAYLOAD_MASS__KG_ > 4000 and PAYLOAD_MASS__KG_ < 6000;
```

```
 * 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

- SpaceX experienced only 1 failed mission, while there were 100 successful mission outcomes.
- A wild card % character is used with an or operator to identify all outcomes starting with either “Success” or “Failure”.



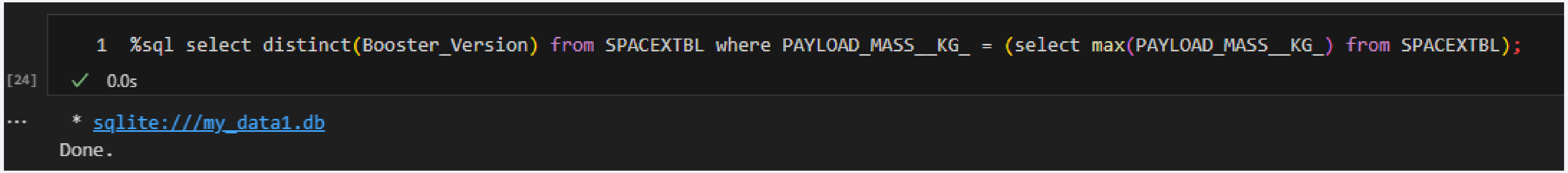
```
%sql select Mission_Outcome, count(*) from SPACEXTBL where Mission_Outcome like 'Success%' or Mission_Outcome like 'Failure%' group by Mission_Outcome;

```

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

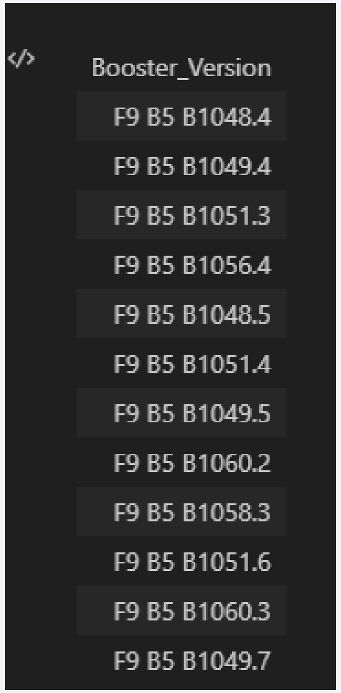
Mission_Outcome	count(*)
Failure (in flight)	1
Success	98
Success	1
Success (payload status unclear)	1

Boosters Carried Maximum Payload

- 12 boosters have carried the max payload of ~2535 kg. They are listed below.
- A subquery is used to first identify the max payload value. This is then used in the parent where clause.



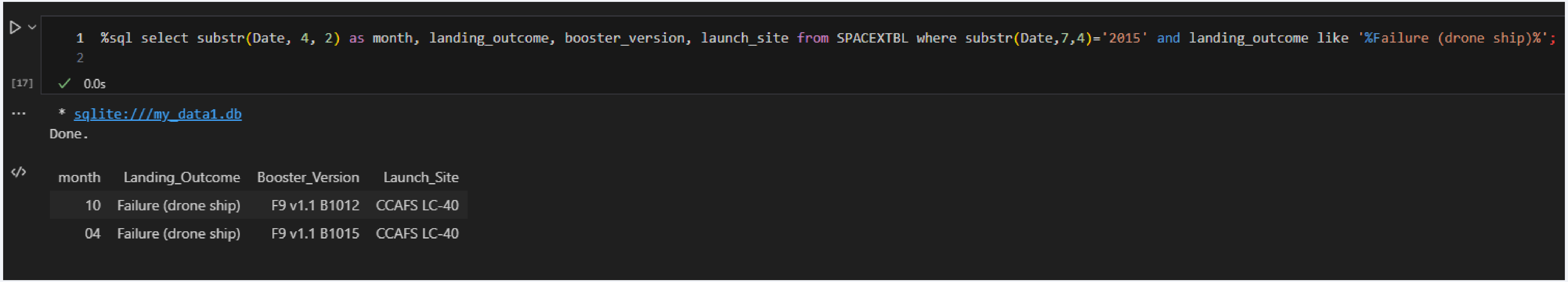
```
%sql select distinct(Booster_Version) from SPACEXTBL where PAYLOAD_MASS__KG_ = (select max(PAYLOAD_MASS__KG_) from SPACEXTBL);
[24] ✓ 0.0s
...  * sqlite:///my_data1.db
    Done.
```



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

2015 Launch Records

- There were two failed landing outcomes in 2015. The month, booster, and launch site for each are listed below.
- SQLite substr(string, start, length) function was used for parsing str formatted dates as SQLite does not support month names.
- SQLite is 1 indexed. As a result, substr('Hello World', 1, 5) = "Hello" rather than "ello ".



```
%sql select substr(Date, 4, 2) as month, landing_outcome, booster_version, launch_site from SPACEXTBL where substr(Date,7,4)='2015' and landing_outcome like '%Failure (drone ship)%';
```

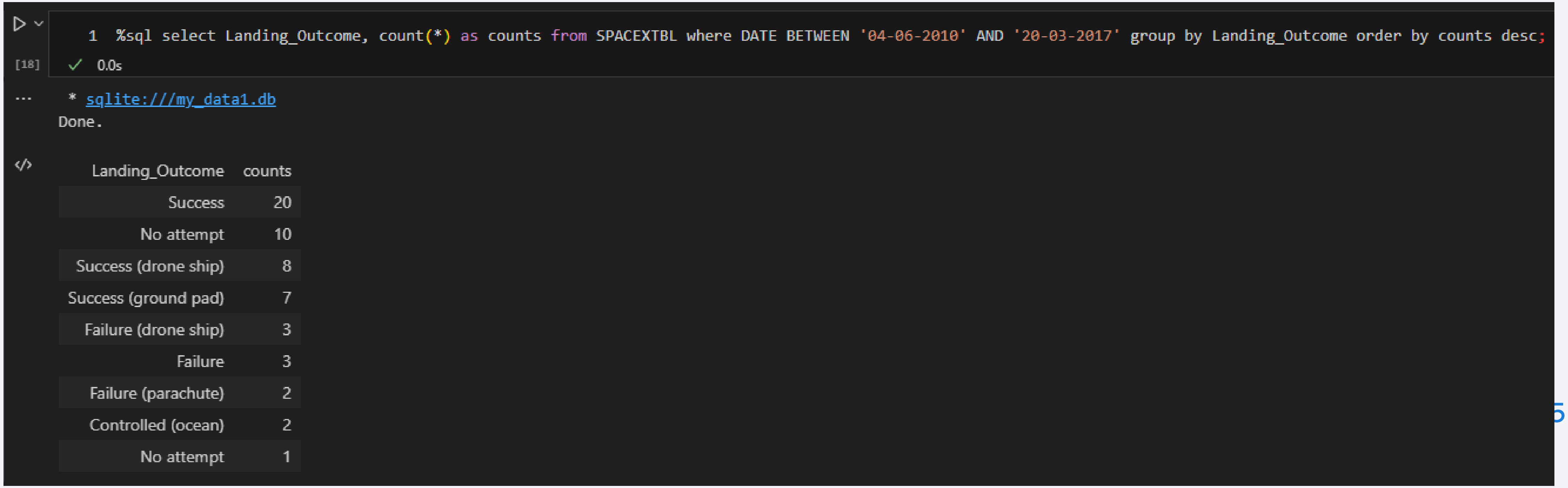
```
[17] ✓ 0.0s
```

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

month	Landing_Outcome	Booster_Version	Launch_Site
10	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
04	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

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

- The ranked 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 are listed below.
- Group by and order by counts used to organize the query results fitting the target date range.



```
%sql select Landing_Outcome, count(*) as counts from SPACEXTBL where DATE BETWEEN '04-06-2010' AND '20-03-2017' group by Landing_Outcome order by counts desc;
```

[18] ✓ 0.0s

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

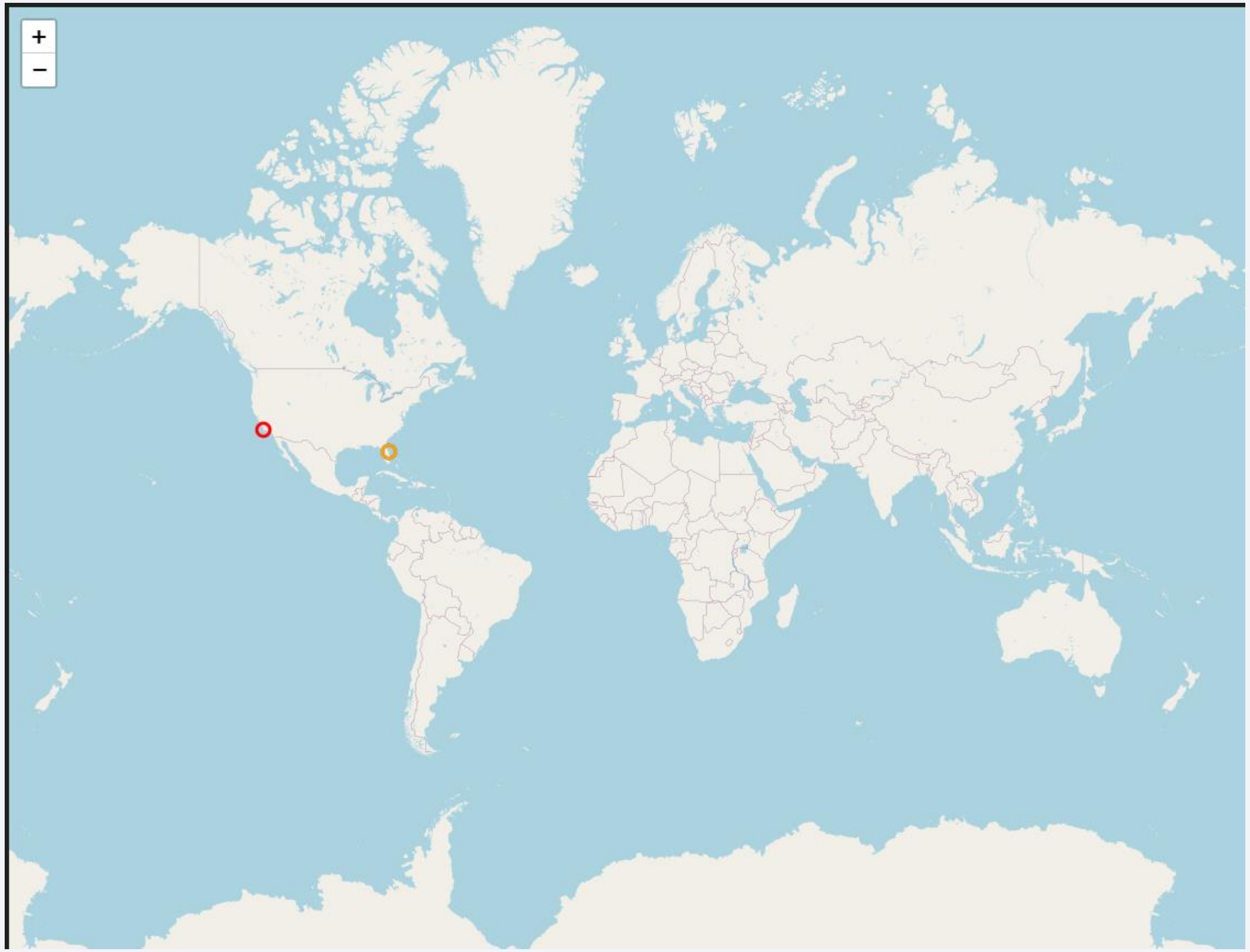
Landing_Outcome	counts
Success	20
No attempt	10
Success (drone ship)	8
Success (ground pad)	7
Failure (drone ship)	3
Failure	3
Failure (parachute)	2
Controlled (ocean)	2
No attempt	1

Section 3

Launch Sites Proximities Analysis

Global Launch Sites

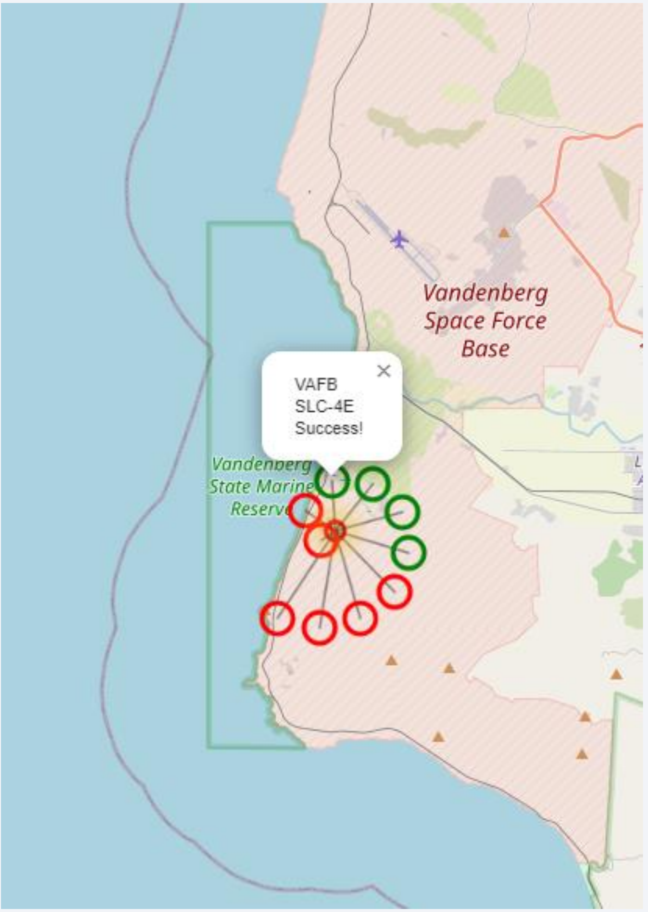
- All SpaceX launches occurred near American coast lines.
- Specifically, the launches occurred in Florida and California.



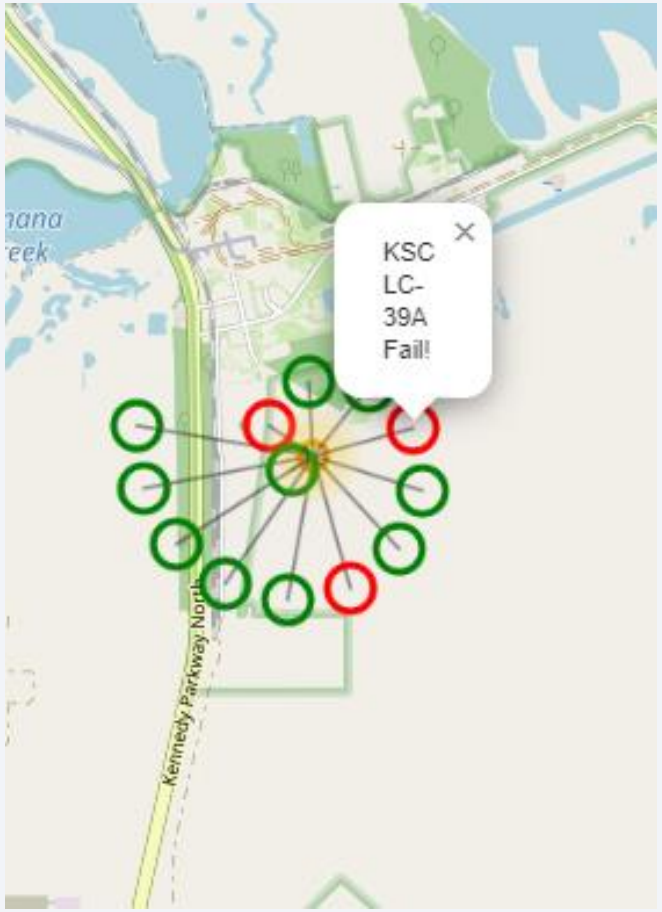
Launch Success by Location

- Successful launches marked in green.
- Failed launches marked in red.

VAFB SLC-4E (California)



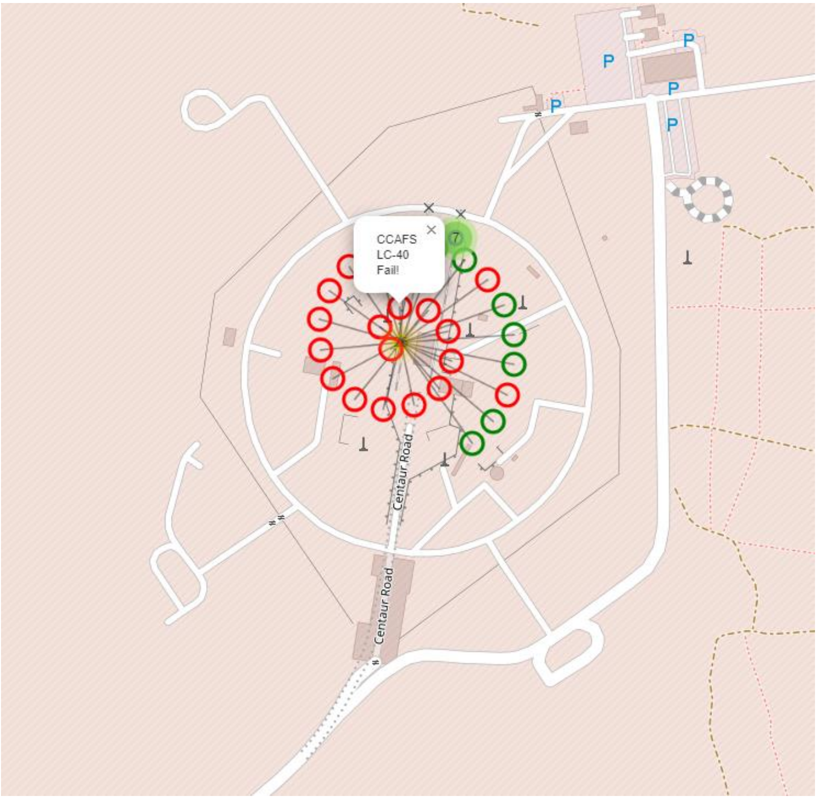
KSC LC-39A (Florida)



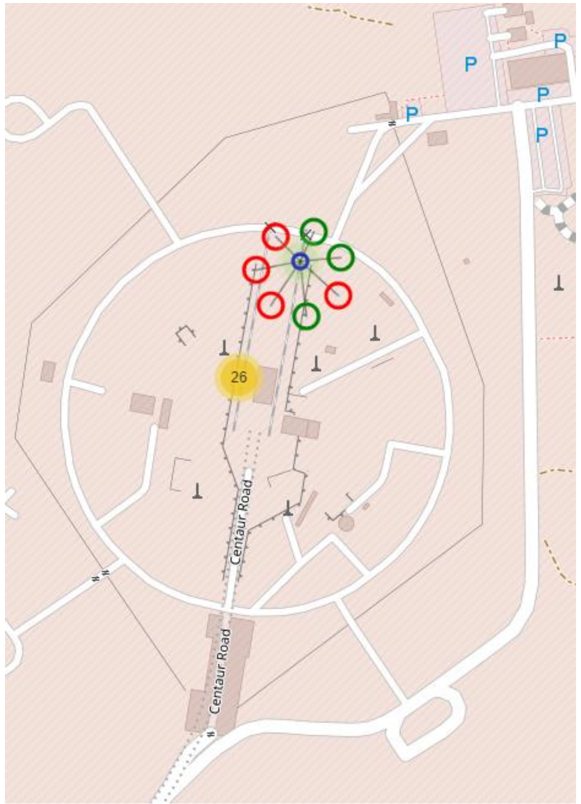
Launch Success by Location

- Successful launches marked in green.
- Failed launches marked in red.

CCAFS LC-40 (Florida)

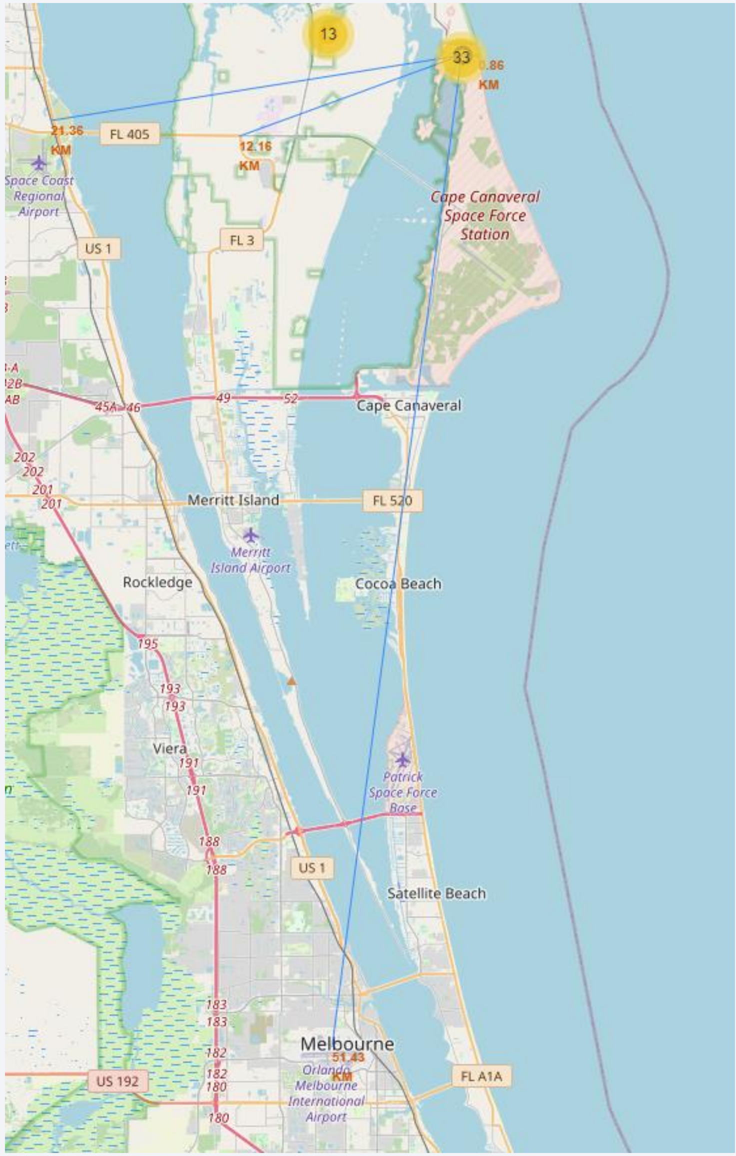


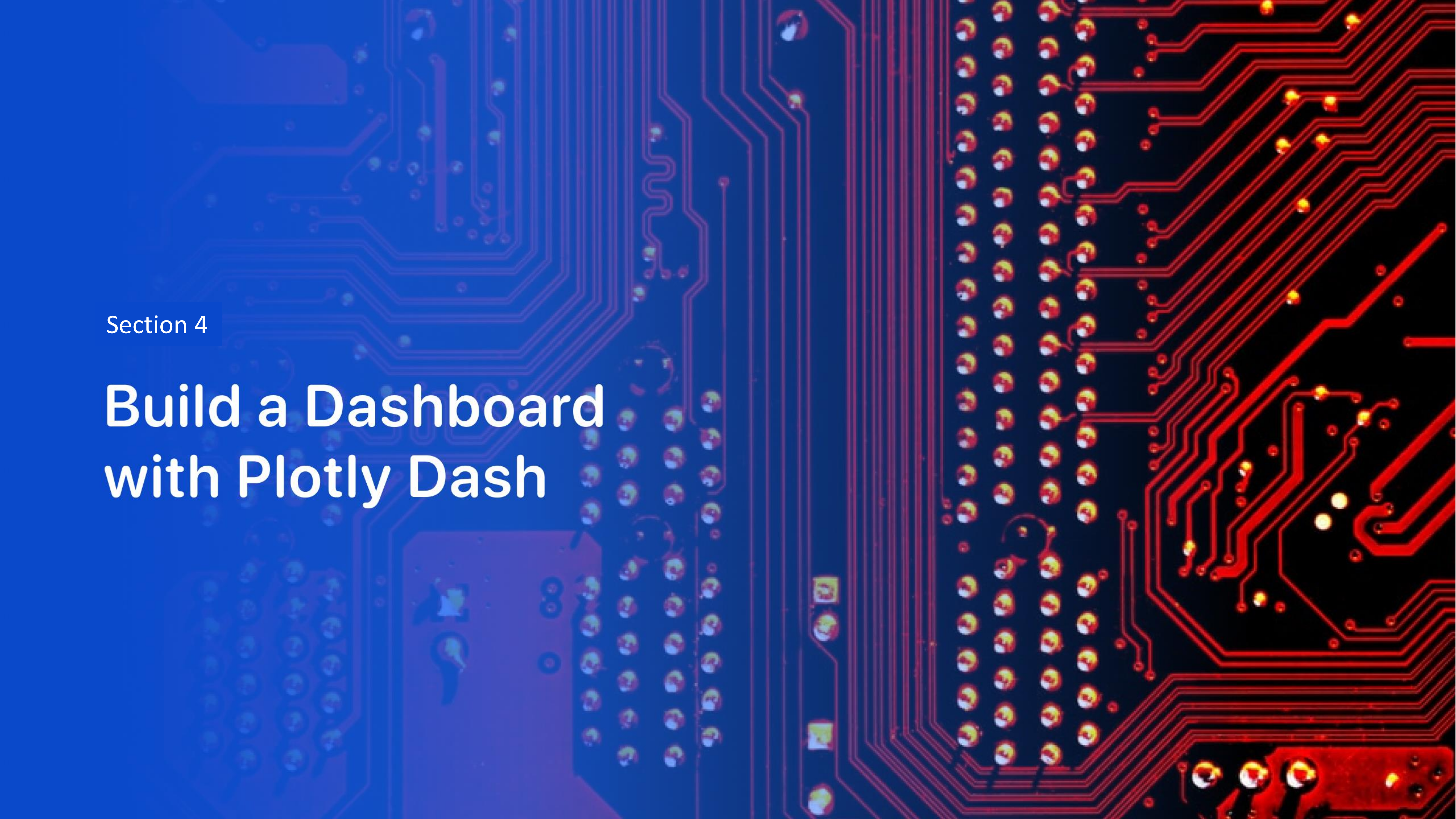
CCAFS SLC-40 (Florida)



Proximity to Infrastructure

- Launch Sites are strategically placed to maximize the distance between the site and key infrastructure like highways, railroads, and cities.
- The distance to water, where crafts could presumably safely crash land is minimized. See the distances to the right.
- The nearest major infrastructure is highway FL 405, roughly 12 kilometers away.



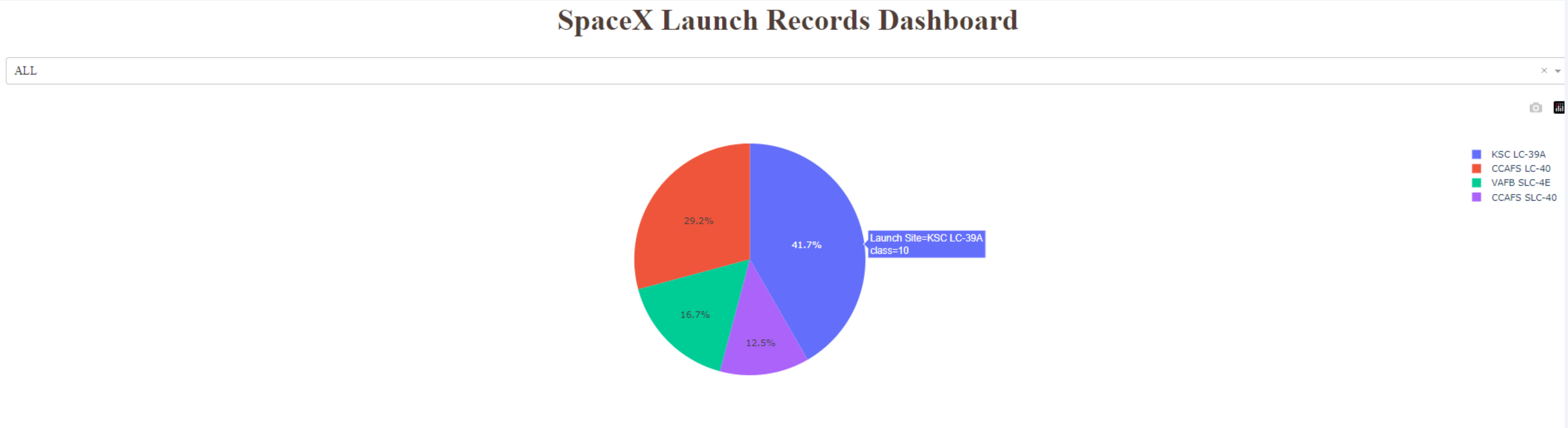


Section 4

Build a Dashboard with Plotly Dash

Successful Launches by Site

- The most successful launches by all sites occurred from KSC LC-39A with a total of 10 successes. This accounts for ~42% of all successes.

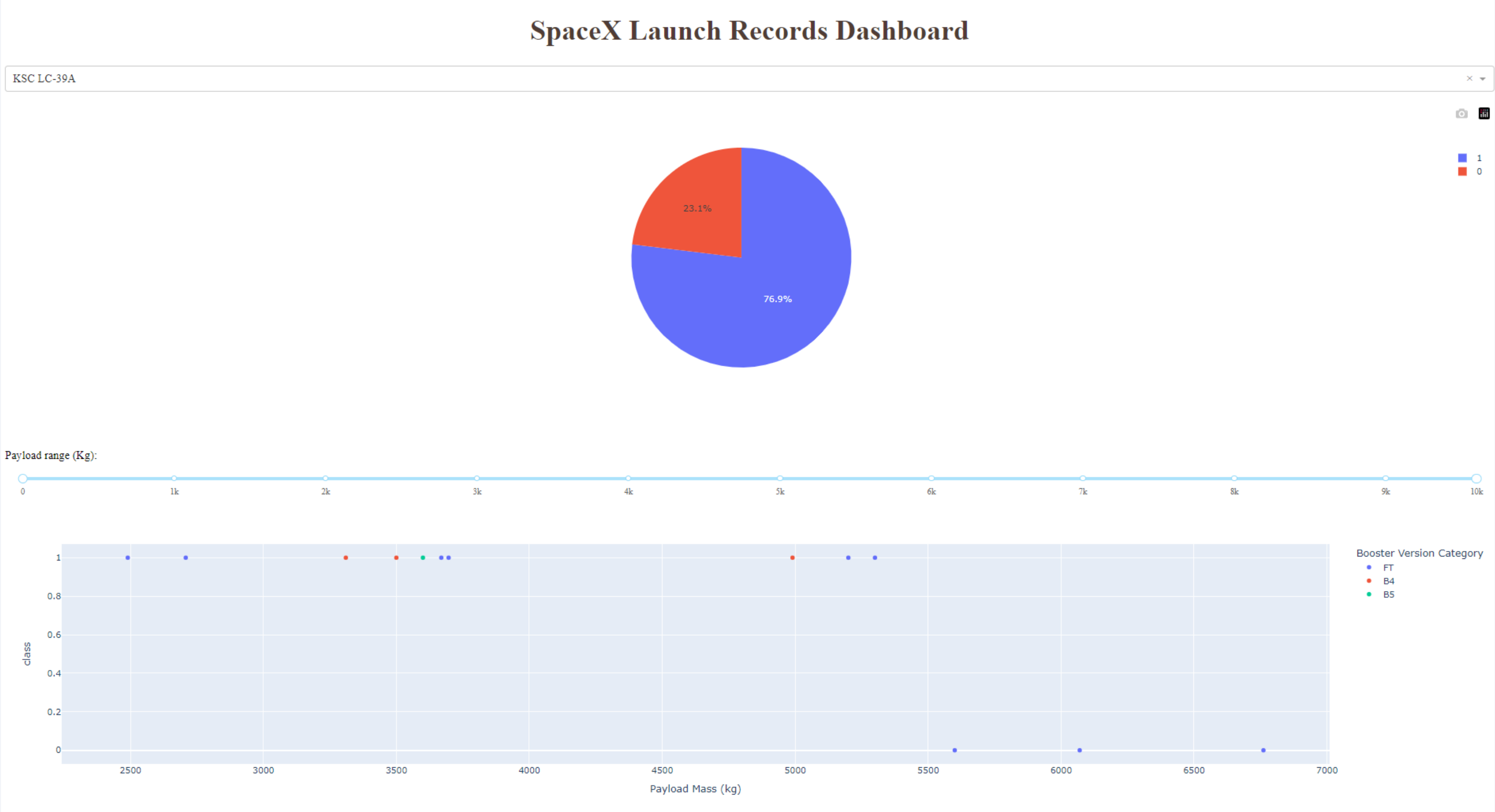


SpaceX Launch Records Dashboard

ALL

KSC LC-39A Success Rate

- The launch site KSC LC-39A has had the highest flight success rate, at ~77% successful flights.
- The only failures have come with payloads over 5500 kg.

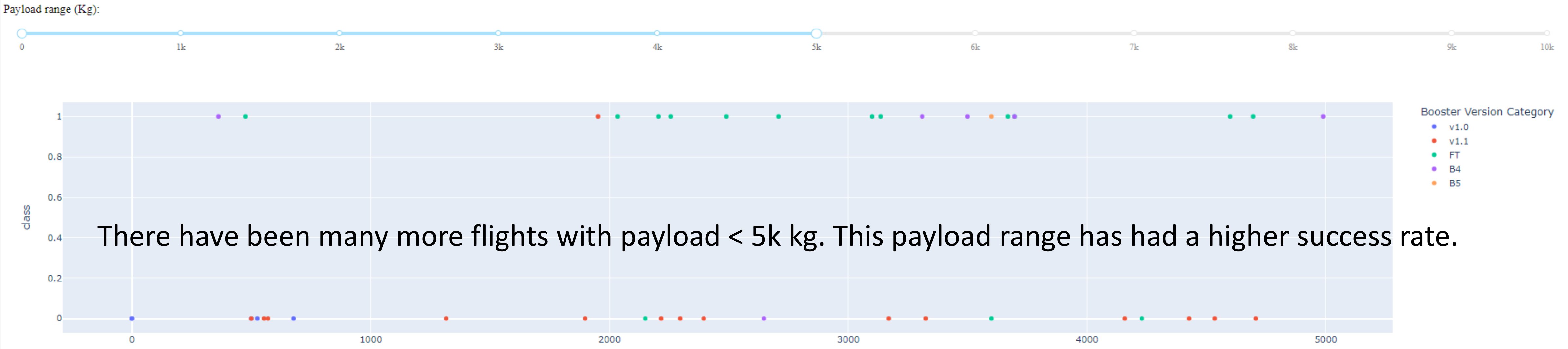


SpaceX Launch Records Dashboard

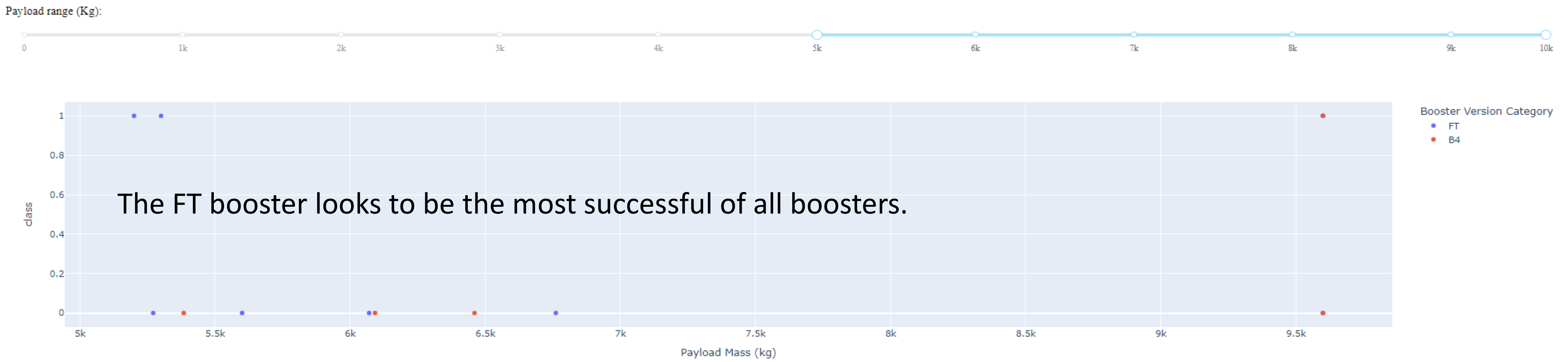
KSC LC-39A

Payload range (Kg):

Success by Booster and Payload



There have been many more flights with payload < 5k kg. This payload range has had a higher success rate.



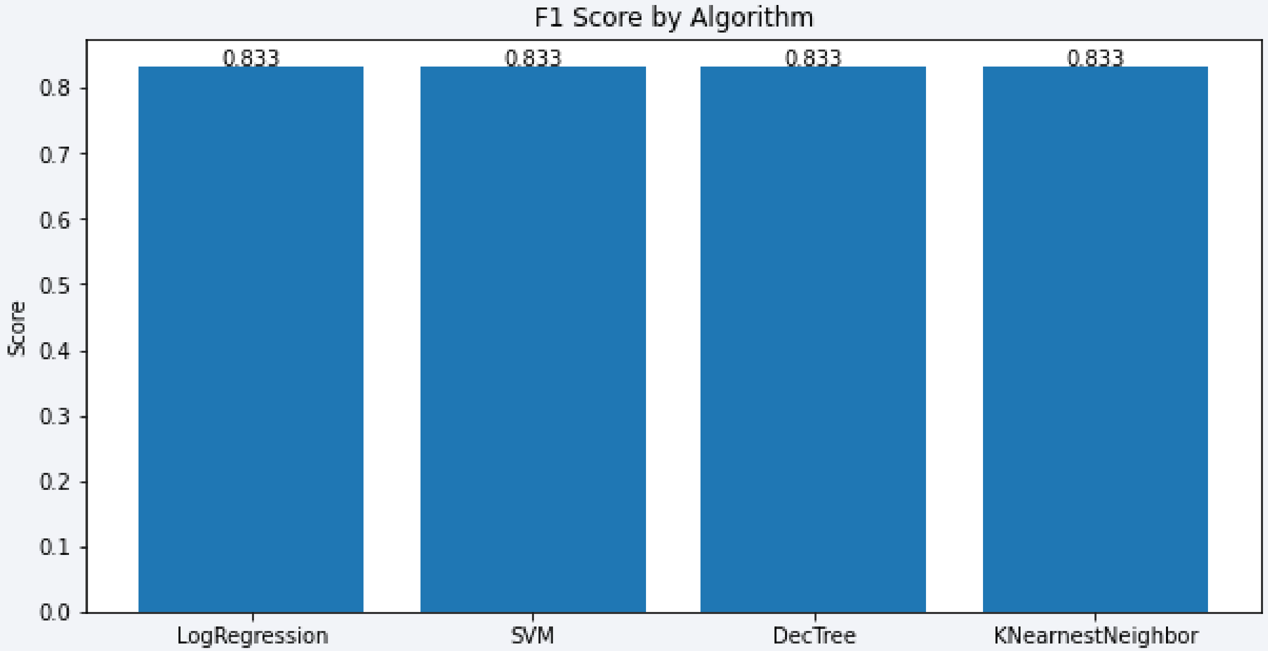
The FT booster looks to be the most successful of all boosters.

Section 5

Predictive Analysis (Classification)

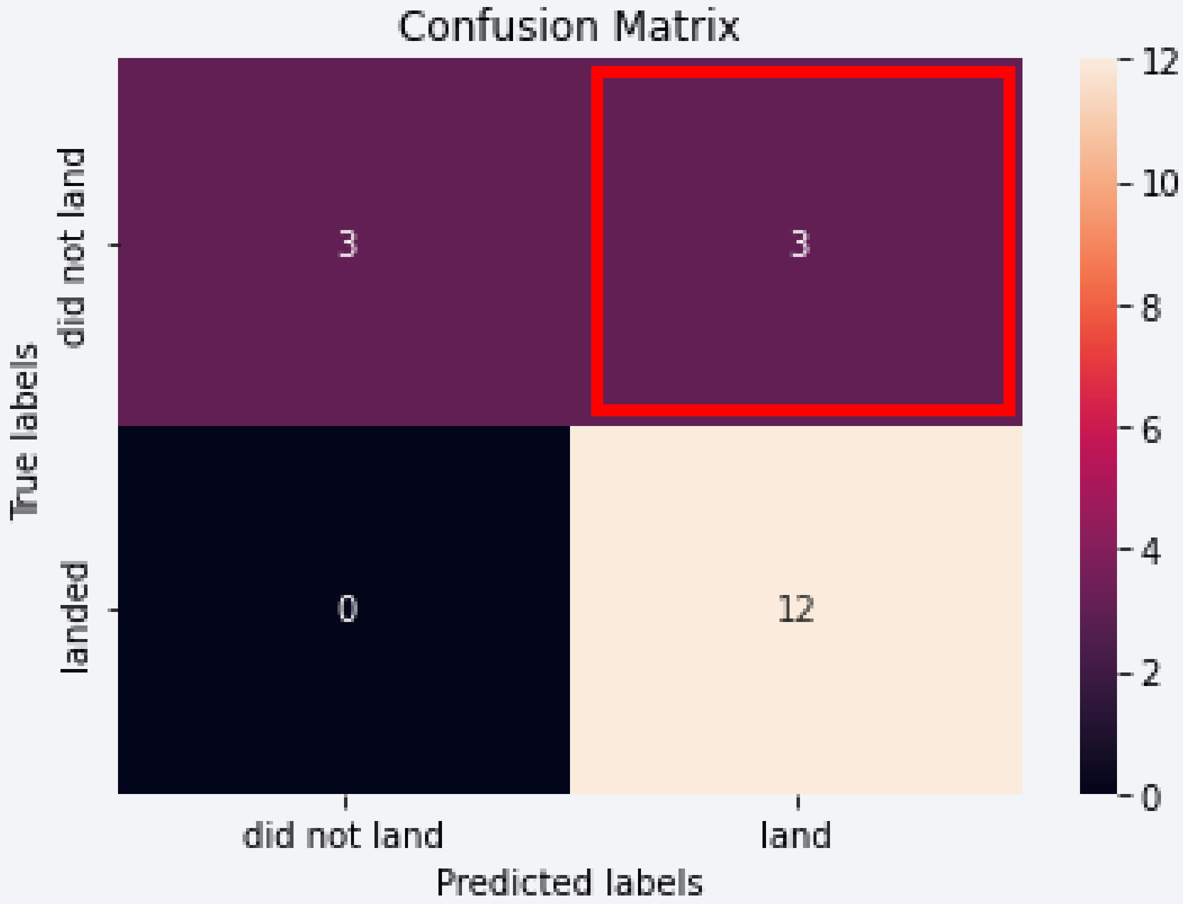
Classification Accuracy

- Model accuracy is similar regardless of the method used for classification.
- This may be explained by such a small dataset n=90.
- It would be recommended to use K Nearest Neighbor due to being non-parametric which may lead to less overfitting.



Confusion Matrix

- Each algorithm produced the same confusion matrix shown right.
- The biggest area for improvement with the predictions is eliminating false positives (flights which were predicted to land but did not, highlighted in red).
  - The model predicted 15 successful landings while only 12 true successful landings exist in the test data.



Conclusions

- SpaceX flight success has increased over time.
  - Rate increased as a trend from 2013 until the end of the data in 2020.
- KSC LC-39A has been the most successful SpaceX launch site.
- Payload mass has increased over time as SpaceX has become more experienced.
- There has not been an unsuccessful flight to ES-L1, GEO, HEO, or SSO orbits.
- It would be recommended to use K Nearest Neighbors to build the classification model.

Thank you!