IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Avijeet Paul
25th November 2023

Outline

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

Executive Summary

- Utilizing the SpaceX API and Wikipedia data, I compiled information on rocket landings and established a 'class' label to distinguish successful landings. Employing SQL, visualizations, folium maps, and dashboards, I delved into data exploration. Extracted pertinent features, converted categorical variables to binary through one-hot encoding, standardized the data, and optimized machine learning models via GridSearchCV for enhanced performance. The accuracy scores of all models were visually presented for comprehensive analysis.
- I developed four machine learning models—Logistic Regression, Support Vector Machine, Decision Tree Classifier, and K Nearest Neighbors—all yielding comparable results, boasting an accuracy rate of approximately 83.33%. Notably, each model exhibited a tendency to over-predict successful landings. Enhancing model accuracy and robustness would require augmenting the dataset to provide a more comprehensive foundation for analysis and determination.

Introduction

Project background and context

- Commercial Space Travel is what the future holds .
- Space X has the best pricing of $62 million vs. where competition is priced at $165 million USD.
- SpaceX is able to reuse the first stage of the rocket launched into orbit
- Space Y wants to compete with Space X

Problem:

- 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:
  - Take data from SpaceX public APIs and from Wikipedia pages
- Perform data wrangling
  - Added relevant data and removed the irrelevant onces
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Made 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.

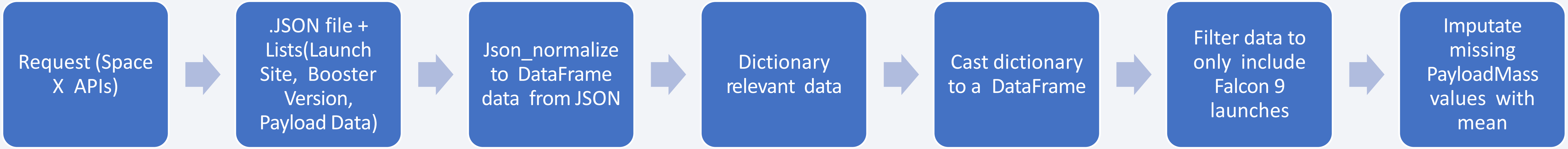
Space X API Data Columns:

- 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



Request (Space X APIs)

.JSON file + Lists(Launch Site, Booster Version, Payload Data)

Json_normalize to DataFrame data from JSON

Dictionary relevant data

Cast dictionary to a DataFrame

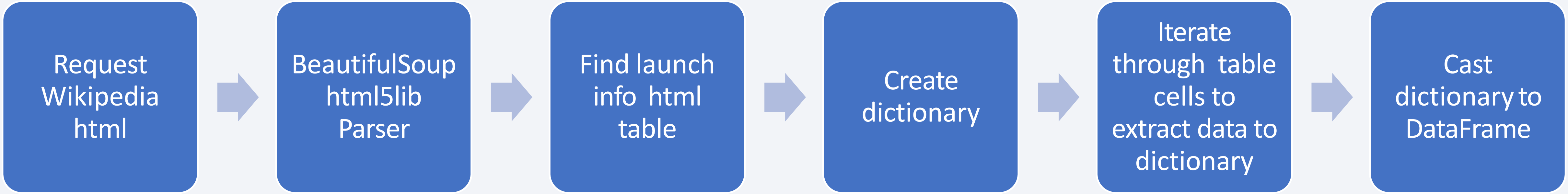
Filter data to only include Falcon 9 launches

Imputate missing PayloadMass values with mean

Github URL:

- https://github.com/AvijeetPaul/IBM-Data-Science-Capstone-Project/blob/main/Week%201/jupyter-labs-spacex-data-collection-api.ipynb

Data Collection - Scraping



Request Wikipedia html

BeautifulSoup html5lib Parser

Find launch info html table

Create dictionary

Iterate through table cells to extract data to dictionary

Cast dictionary to DataFrame

Github URL:

- https://github.com/AvijeetPaul/IBM-Data-Science-Capstone-Project/blob/main/Week%201/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.

<u>Value Mapping:</u>

- 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/AvijeetPaul/IBM-Data-Science-Capstone-Project/blob/main/Week%201/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. Success Rate, 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/AvijeetPaul/IBM-Data-Science-Capstone-Project/blob/main/Week%202/jupyter-labs-eda-dataviz.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

<u>Github URL:</u>

- https://github.com/AvijeetPaul/IBM-Data-Science-Capstone-Project/blob/main/Week%202/jupyter-labs-eda-sql-coursera_sqllite%20(1).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/AvijeetPaul/IBM-Data-Science-Capstone-Project/blob/main/Week%203/lab_jupyter_launch_site_location.ipynb

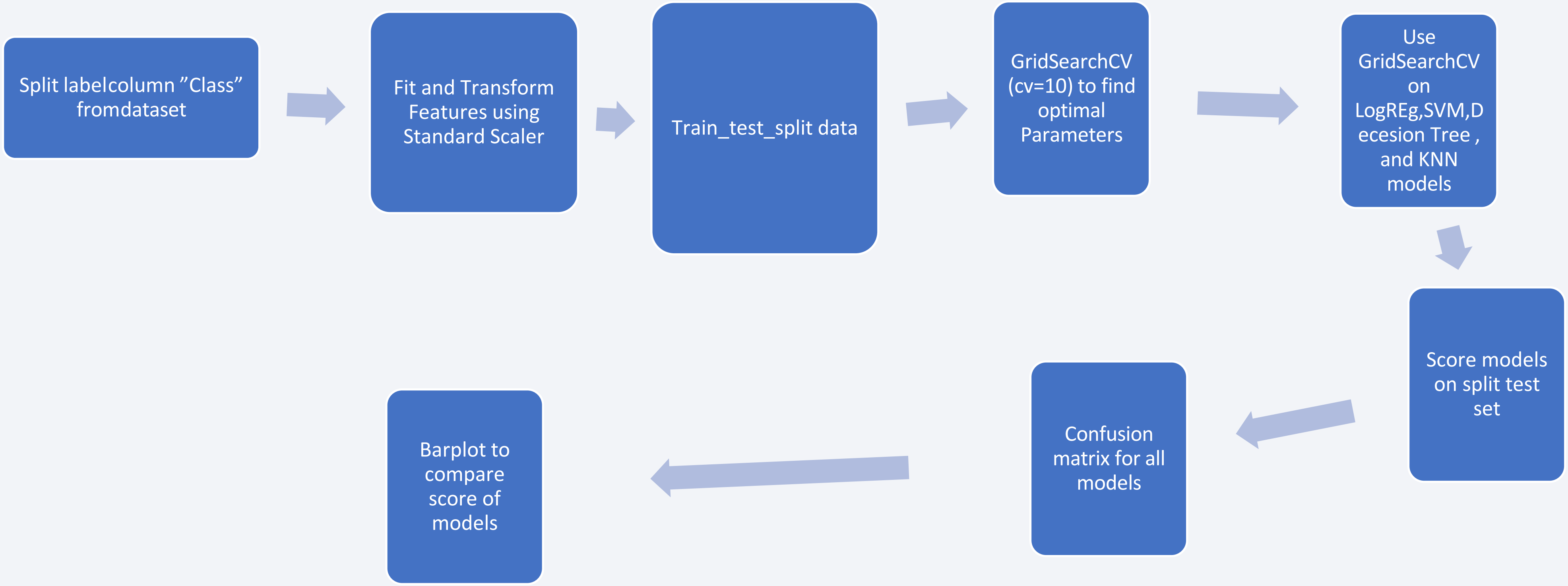
Build a Dashboard with Plotly Dash

- 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/AvijeetPaul/IBM-Data-Science-Capstone-Project/blob/main/Week%203/spacex_dash.py

Predictive Analysis (Classification)



Split labelcolumn ”Class” fromdataset

Fit and Transform Features using Standard Scaler

Train_test_split data

GridSearchCV (cv=10) to find optimal Parameters

Use GridSearchCV on LogREg,SVM,Decesion Tree , and KNN models

Score models on split test set

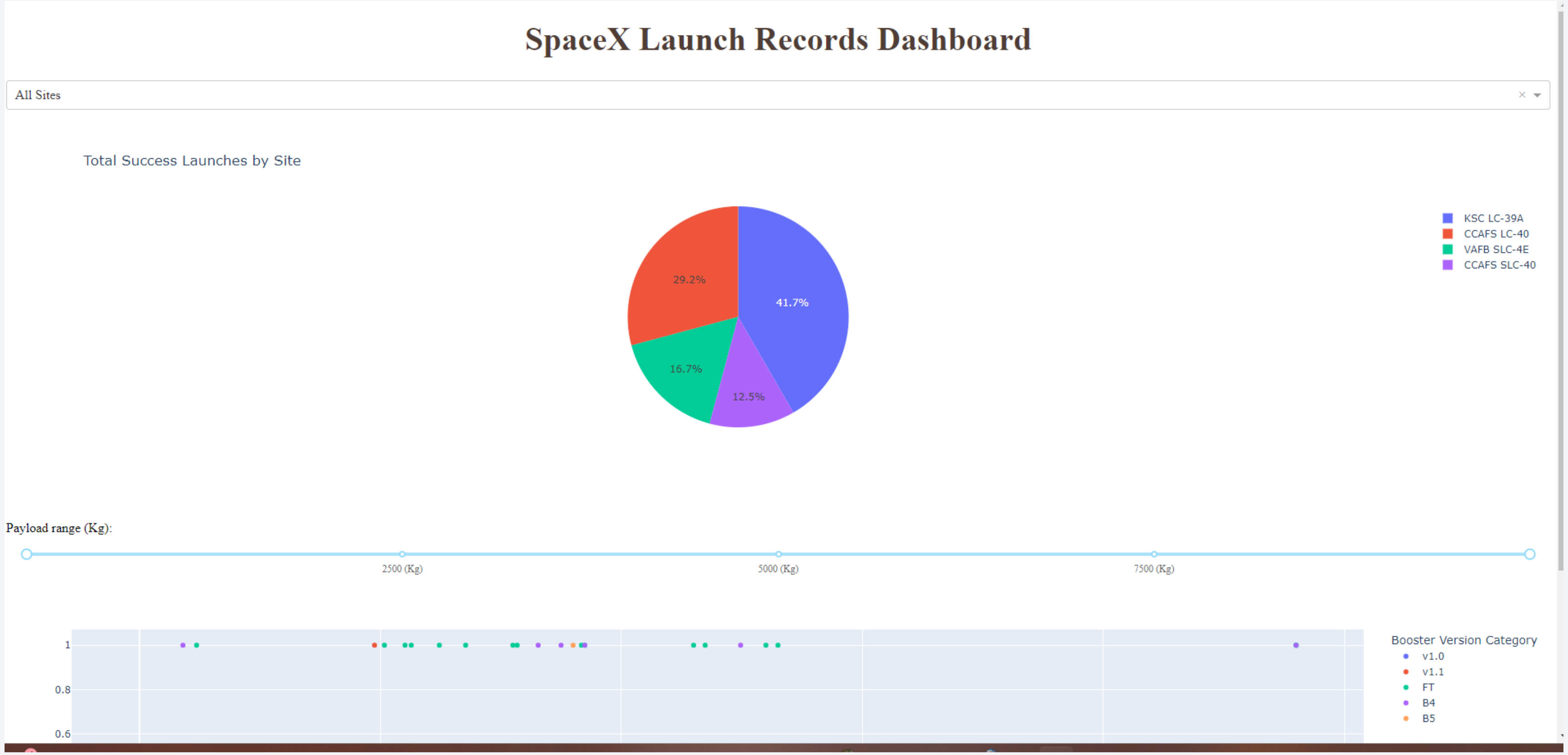
Confusion matrix for all models

Barplot to compare score of models

Github URL:

- https://github.com/AvijeetPaul/IBM-Data-Science-Capstone-Project/blob/main/Week%204/SpaceX_Machine_Learning_Prediction_Part_5.jupyterlite.ipynb

Results



SpaceX Launch Records Dashboard

All Sites

This is a preview of the Plotly dashboard. The following sides will show the results of EDA with visualization, EDA with SQL, Interactive Map with Folium, and finally the results of our model with about 83% accuracy.

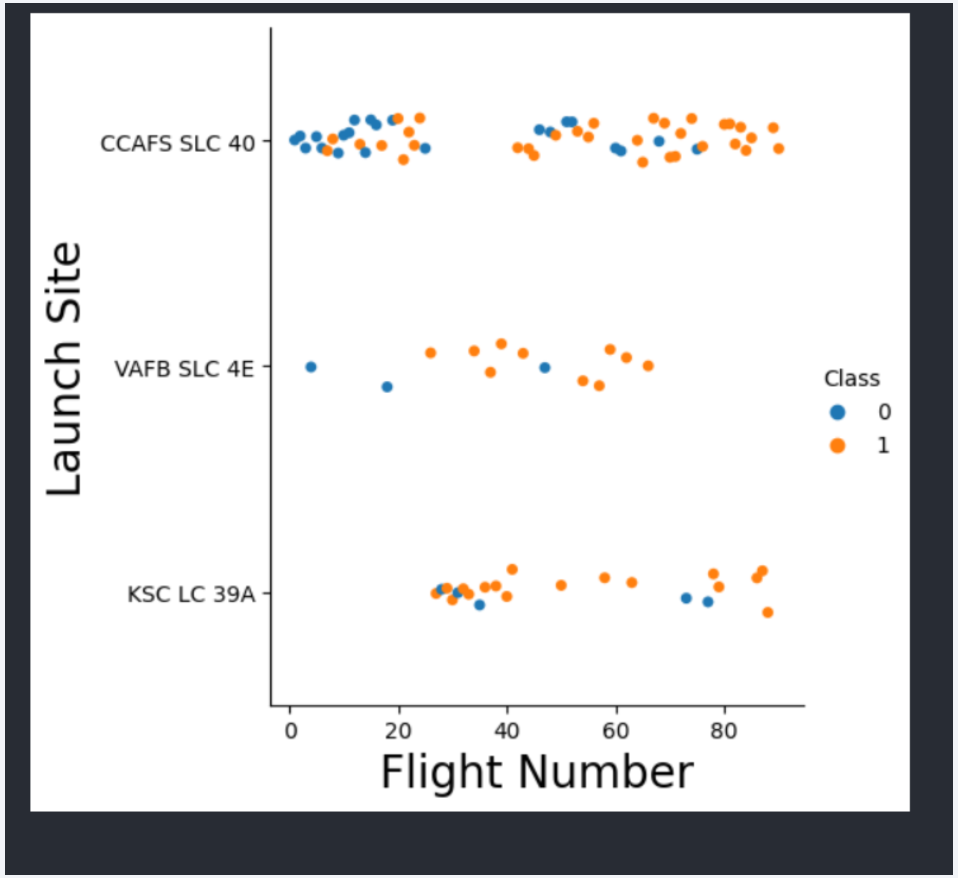
Section 2

Insights drawn from EDA

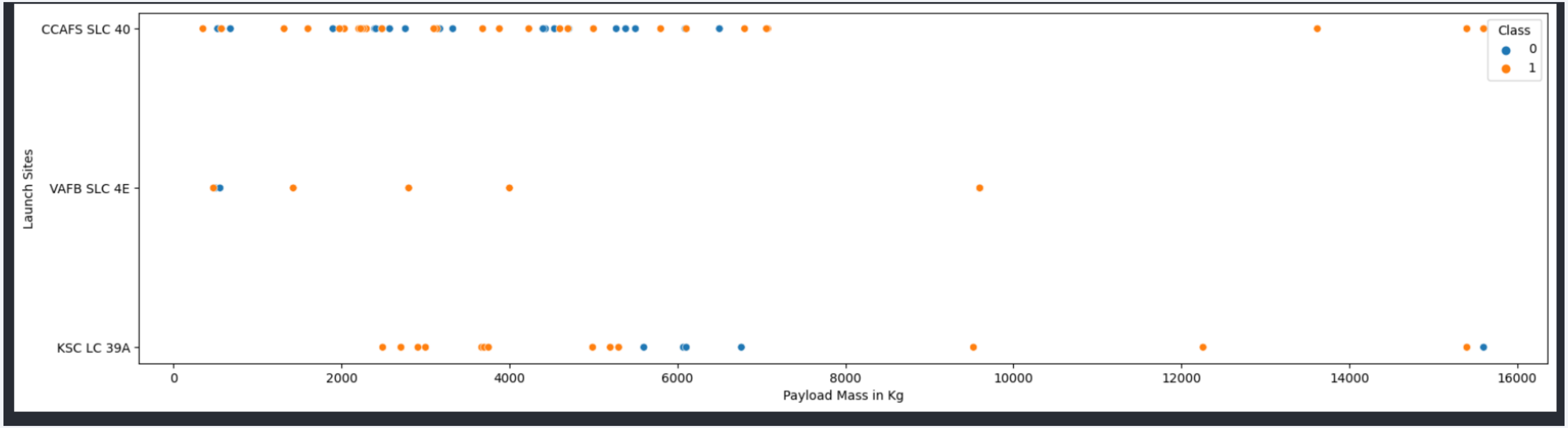
Flight Number vs. Launch Site

- Blue indicates unsuccessful launch; red indicates successful launch.

- 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

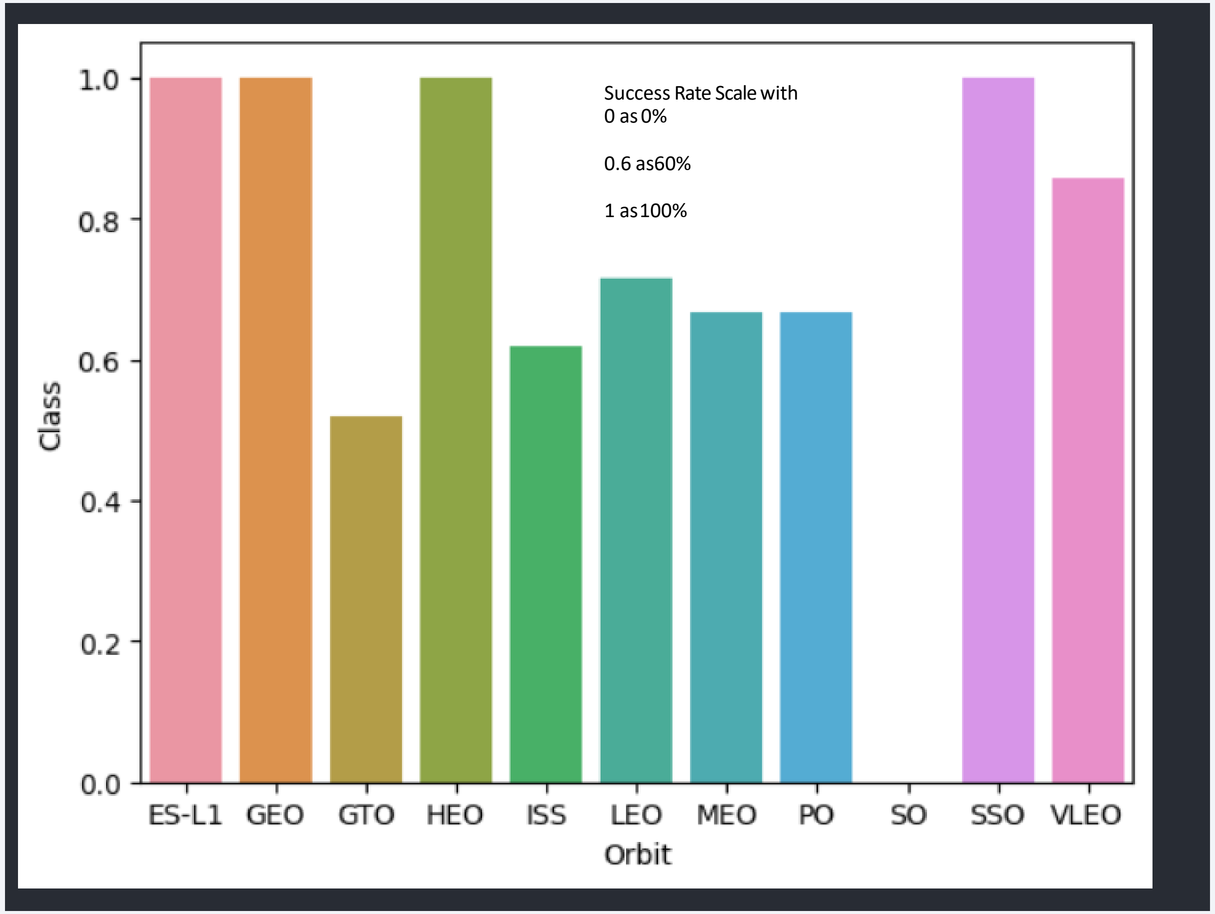


- Blue indicates unsuccessful launch; red indicates successful launch.

- 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

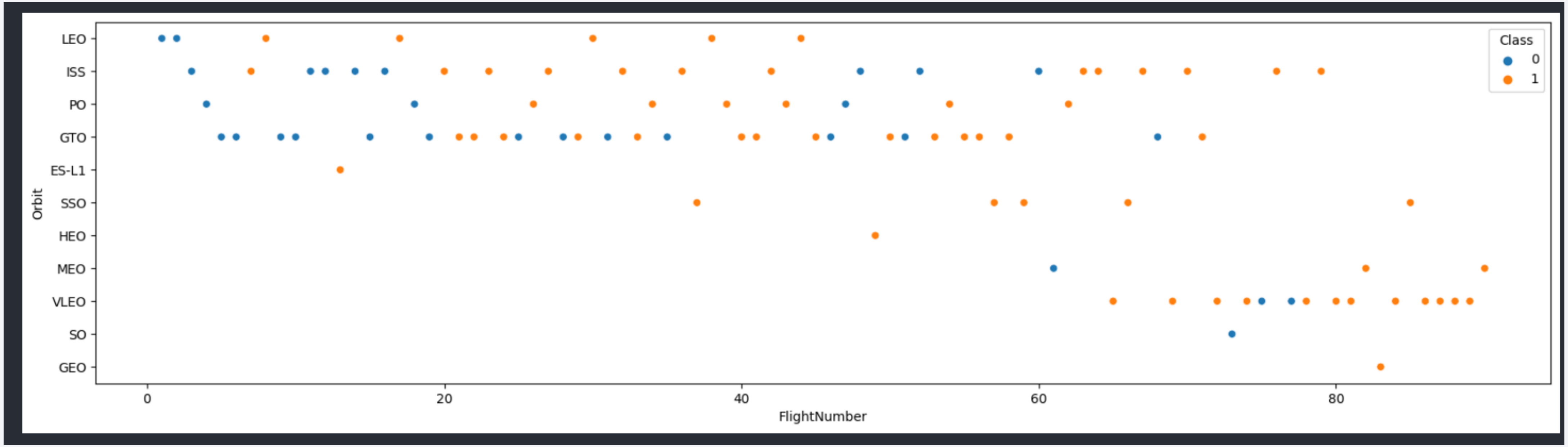


Success Rate Scale with
0 as 0%

0.6 as60%

1 as100%

Flight Number vs. Orbit Type

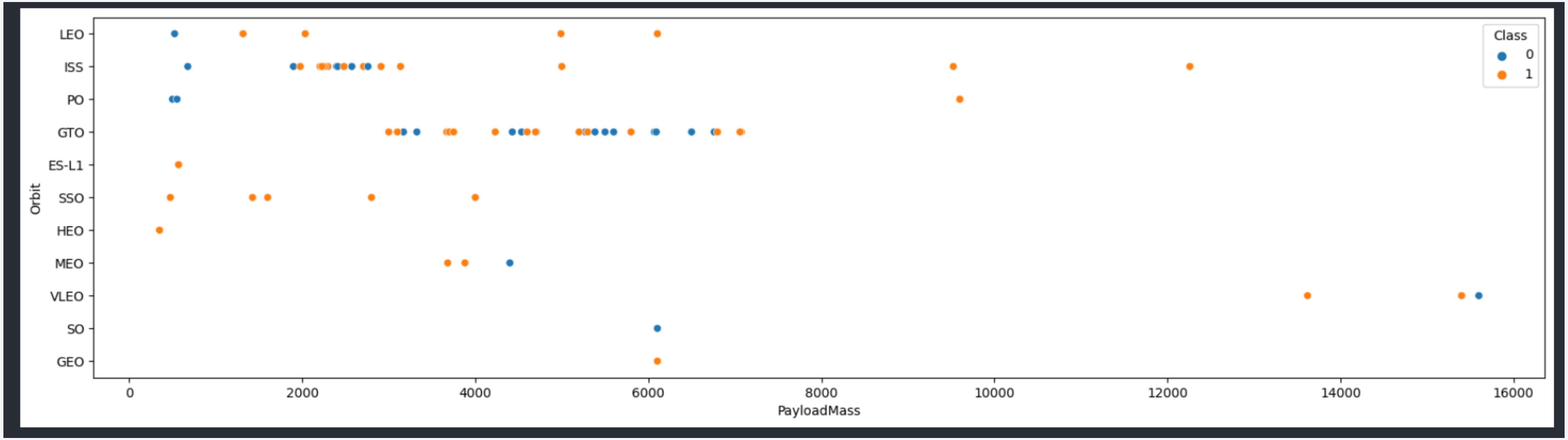


- Blue indicates unsuccessful launch; red indicates successful launch.

- 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

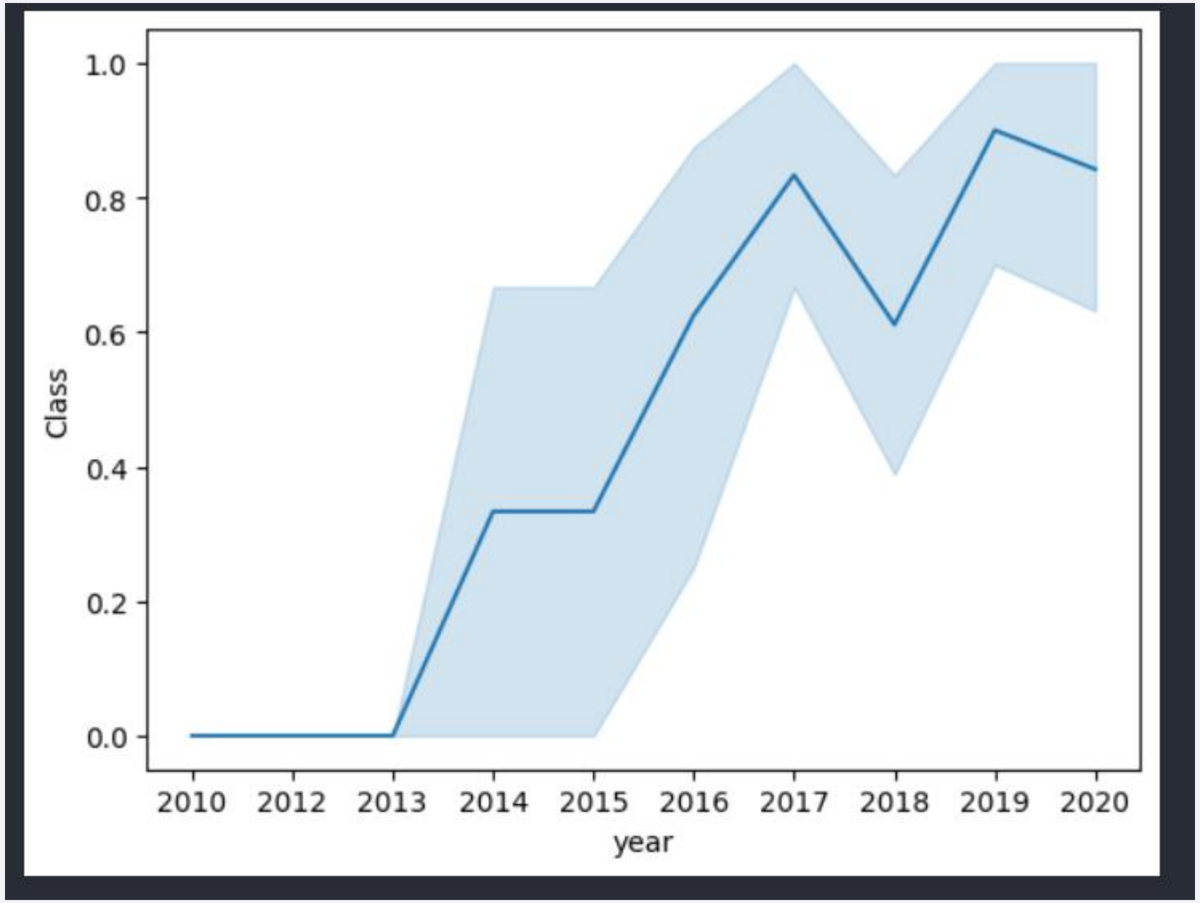
Payload vs. Orbit Type



- 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%

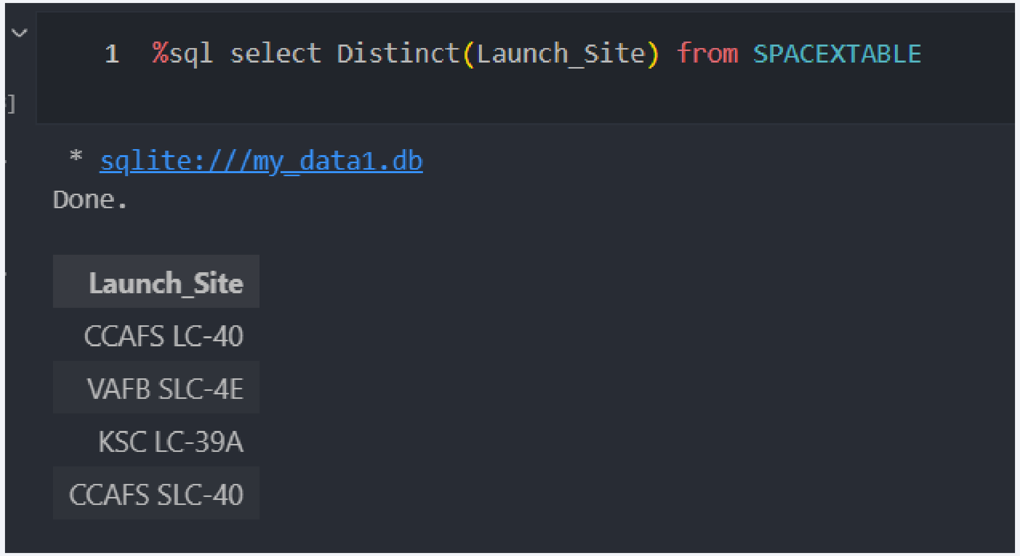


All Launch Site Names

- 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

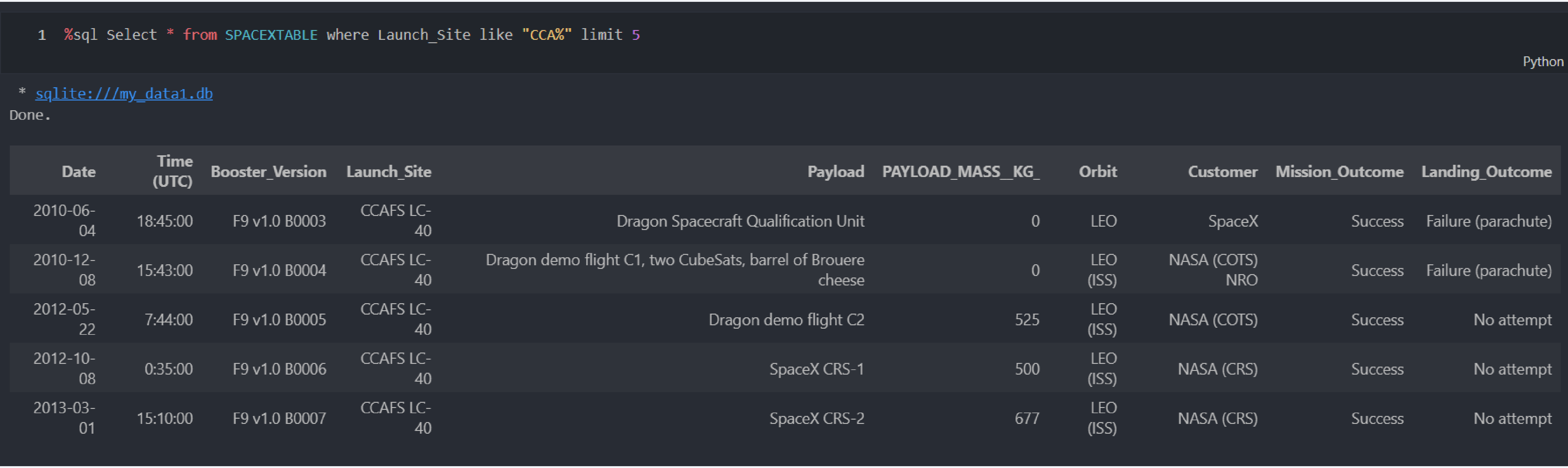


```
%sql select Distinct(Launch_Site) from SPACEXTABLE
```

```
 * 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'

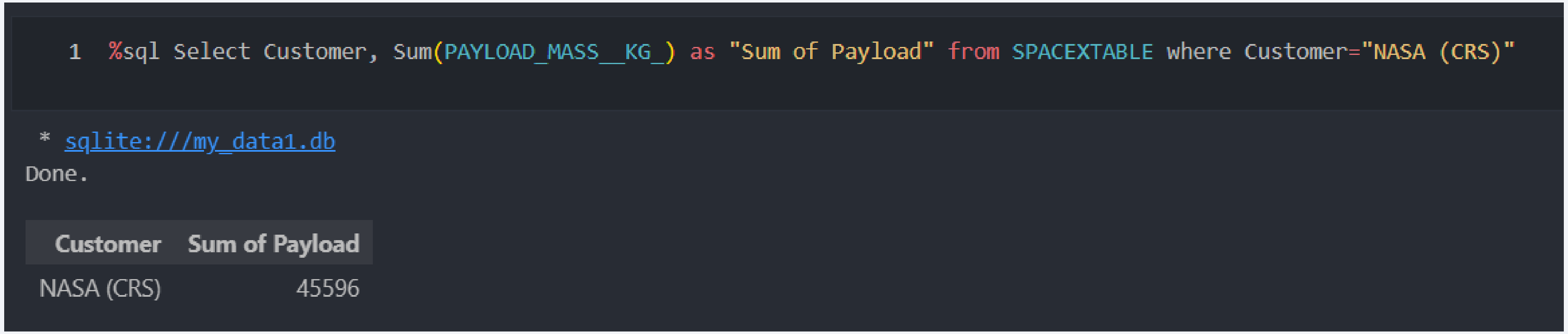


```
%sql Select * from SPACEXTABLE where Launch_Site like "CCA%" limit 5
```

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

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
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

Total Payload Mass



```
%sql Select Customer, Sum(PAYLOAD_MASS__KG_) as "Sum of Payload" from SPACEXTABLE where Customer="NASA (CRS)"
```

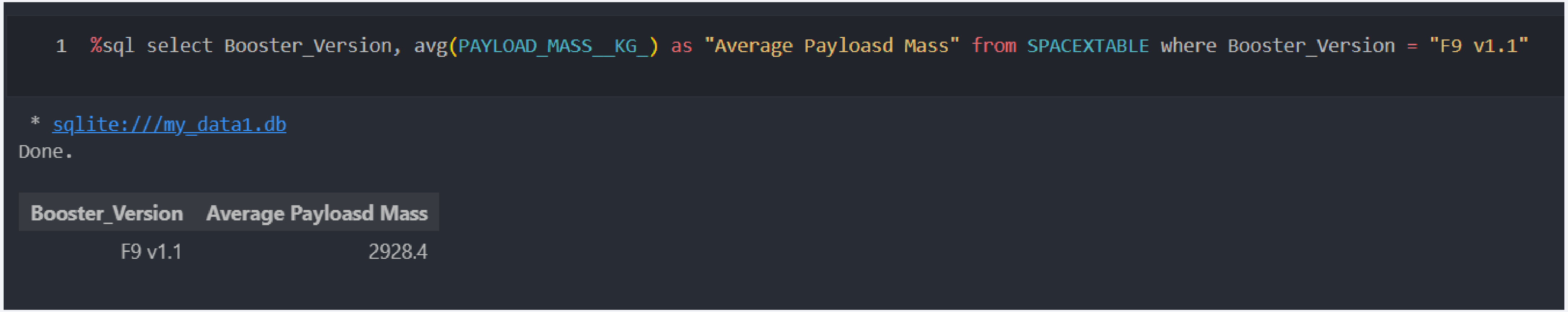
```
 * sqlite:///my_data1.db
Done.
```

Customer	Sum of Payload
NASA (CRS)	45596

- 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

- 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



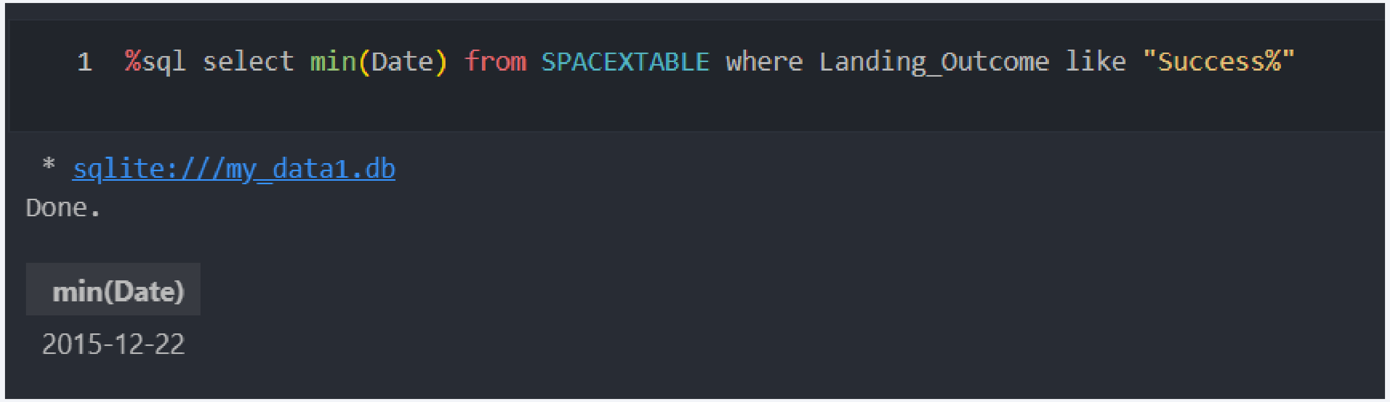
```
%sql select Booster_Version, avg(PAYLOAD_MASS__KG_) as "Average Payloasd Mass" from SPACEXTABLE where Booster_Version = "F9 v1.1"
```

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

Booster_Version	Average Payloasd Mass
F9 v1.1	2928.4

First Successful Ground Landing Date

- 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.



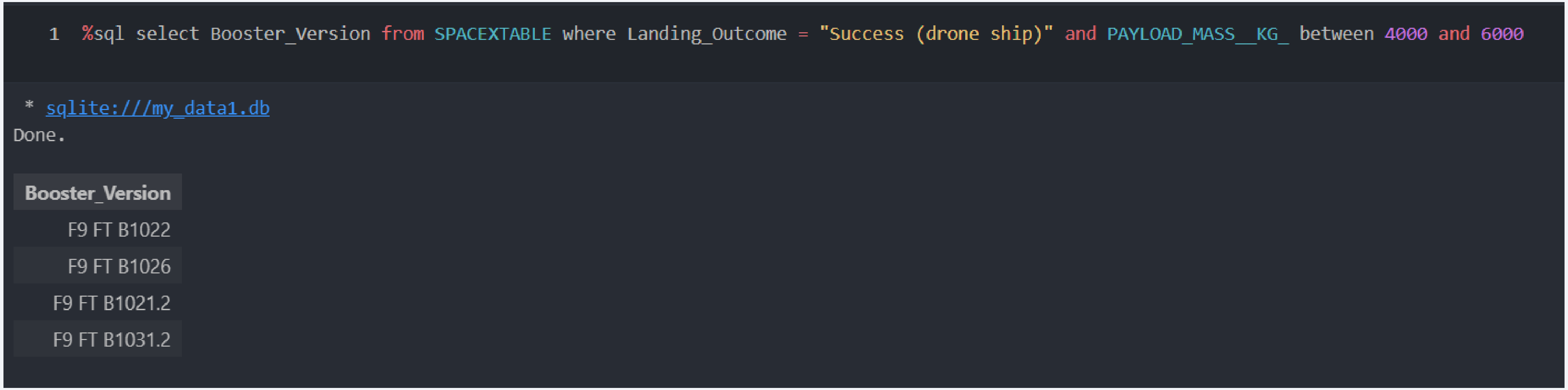
```
%sql select min(Date) from SPACEXTABLE where Landing_Outcome like "Success%"
```

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

min(Date)
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

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



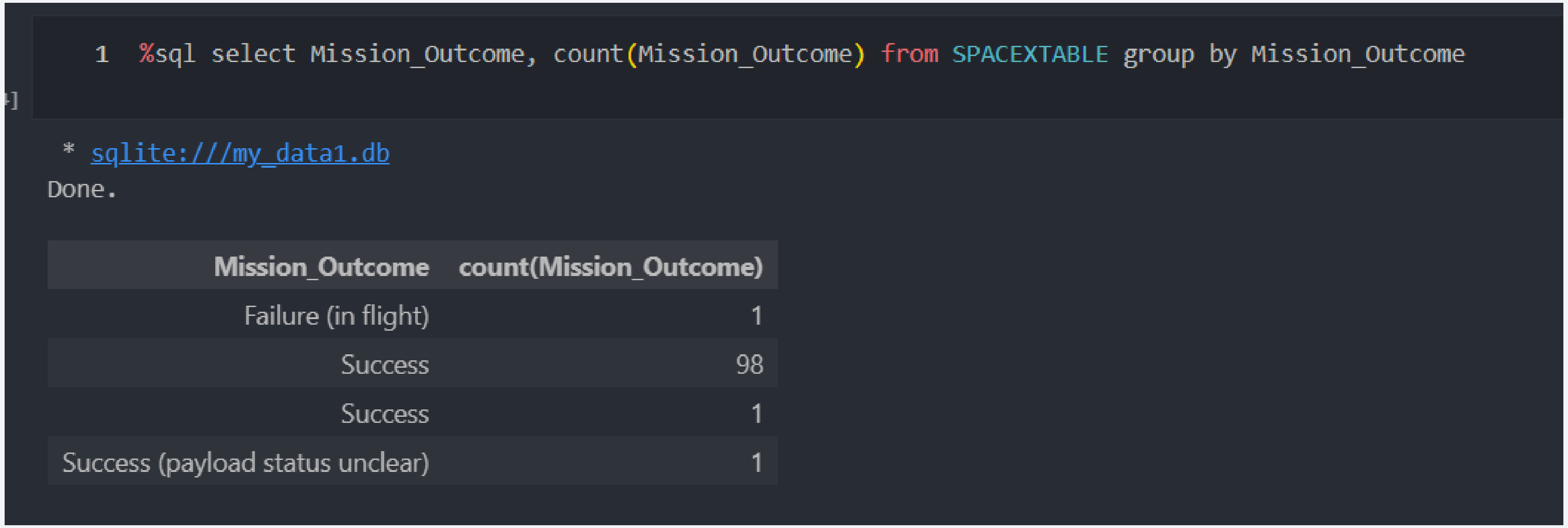
```
%sql select Booster_Version from SPACEXTABLE where Landing_Outcome = "Success (drone ship)" and PAYLOAD_MASS__KG_ between 4000 and 6000
```

```
 * 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

- 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.

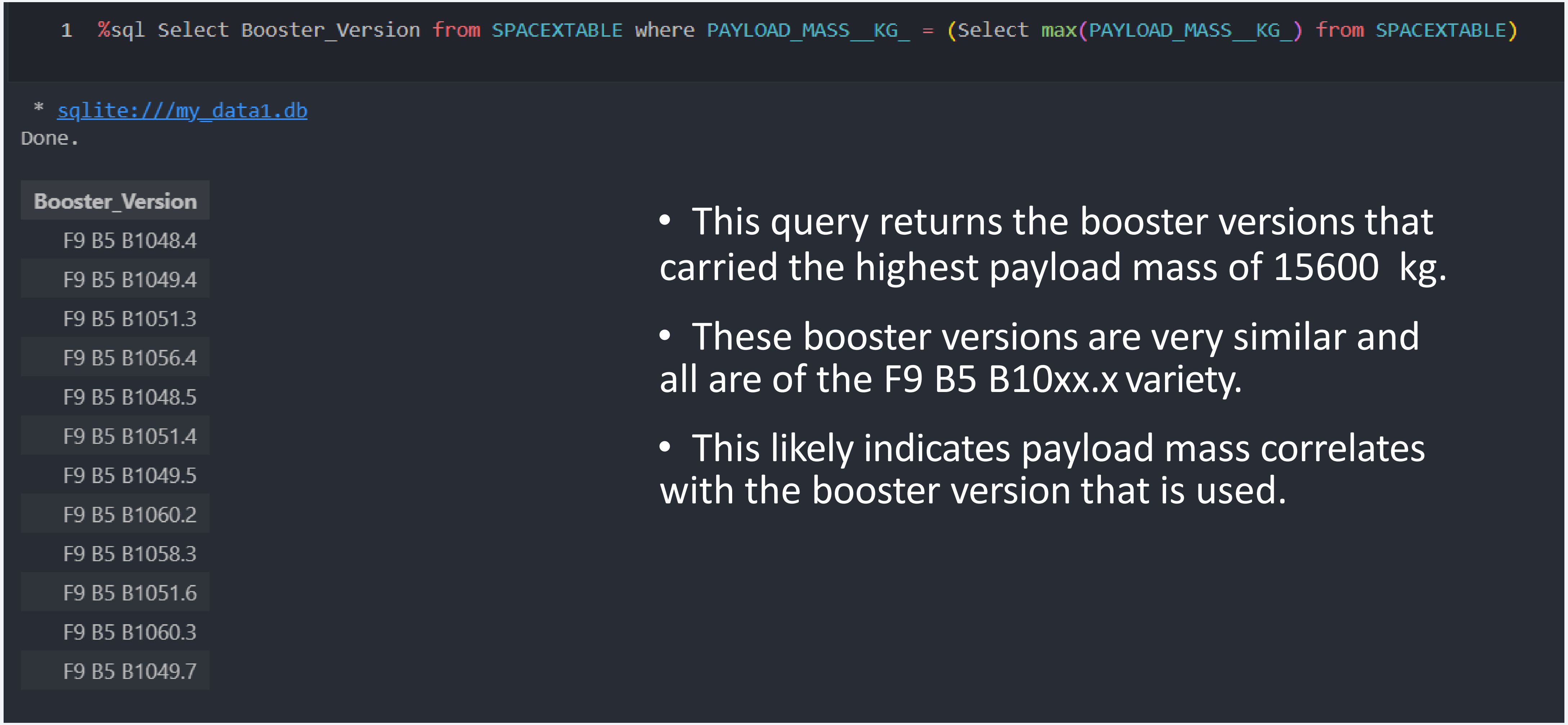


```
%sql select Mission_Outcome, count(Mission_Outcome) from SPACEXTABLE group by Mission_Outcome
```

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

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

Boosters Carried Maximum Payload



```
%sql Select Booster_Version from SPACEXTABLE where PAYLOAD_MASS__KG_ = (Select max(PAYLOAD_MASS__KG_) from SPACEXTABLE)
```

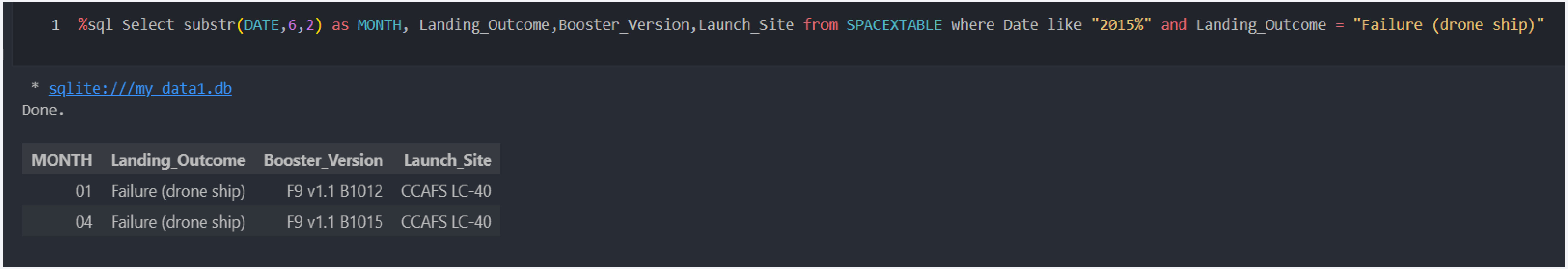
```
 * sqlite:///my_data1.db
Done.
```

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

- This query returns the Month, Landing Outcome, Booster Version, Payload Mass (kg), and Launch site of 2015 launches where stage 1 failed to land on a drone ship.
- There were two such occurrences.

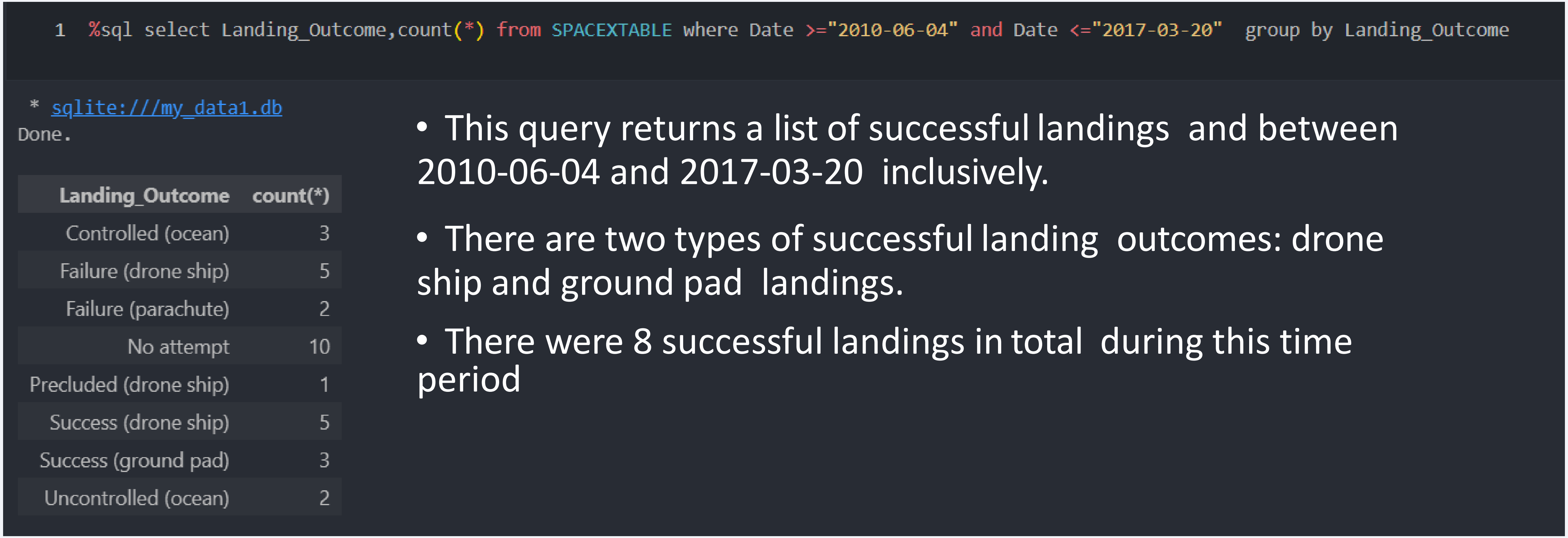


```
%sql Select substr(DATE,6,2) as MONTH, Landing_Outcome,Booster_Version,Launch_Site from SPACEXTABLE where Date like "2015%" and Landing_Outcome = "Failure (drone ship)"
```

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

MONTH	Landing_Outcome	Booster_Version	Launch_Site
01	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



```
%sql select Landing_Outcome,count(*) from SPACEXTABLE where Date >="2010-06-04" and Date <="2017-03-20"  group by Landing_Outcome
```

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

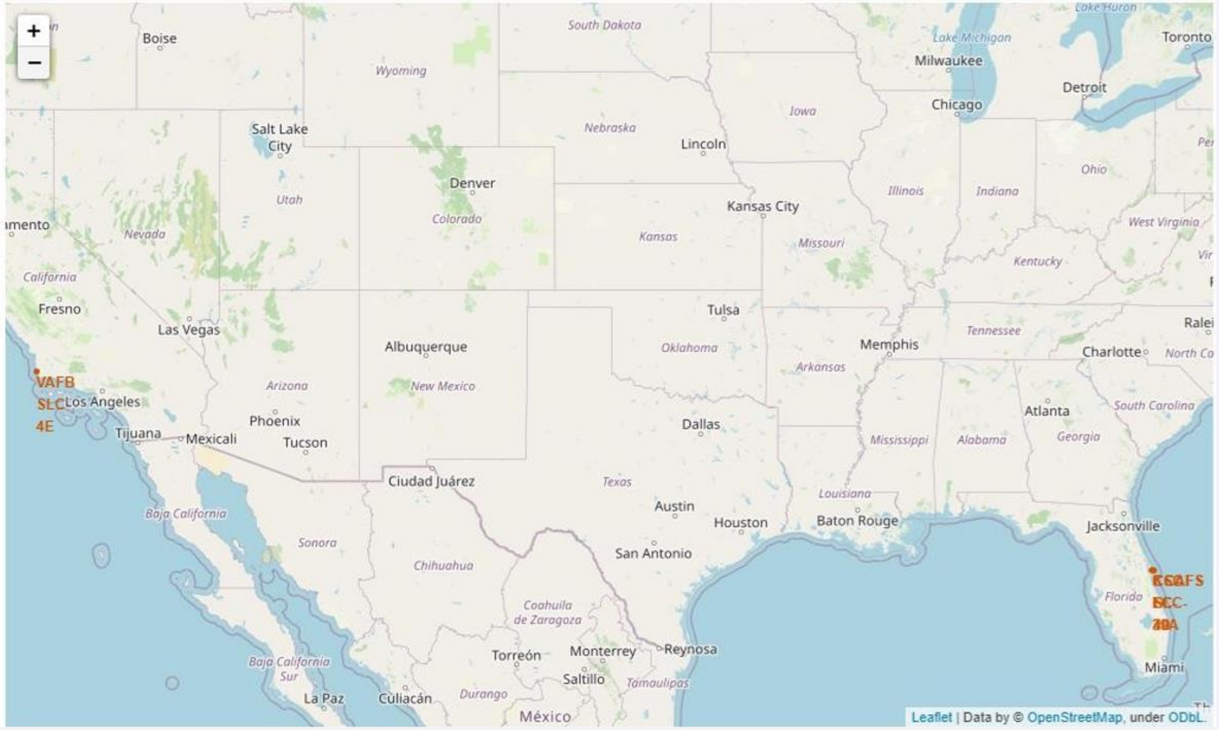
Landing_Outcome	count(*)
Controlled (ocean)	3
Failure (drone ship)	5
Failure (parachute)	2
No attempt	10
Precluded (drone ship)	1
Success (drone ship)	5
Success (ground pad)	3
Uncontrolled (ocean)	2

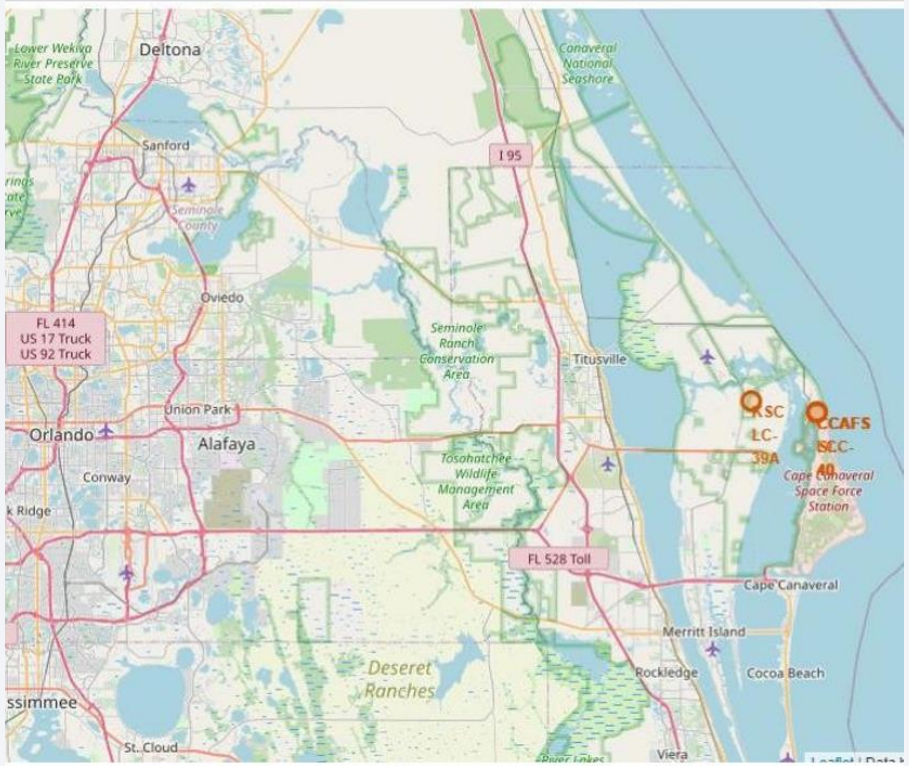
- This query returns a list of successful landings and between 2010-06-04 and 2017-03-20 inclusively.
- There are two types of successful landing outcomes: drone ship and ground pad landings.
- There were 8 successful landings in total during this time period

Section 3

Launch Sites Proximities Analysis

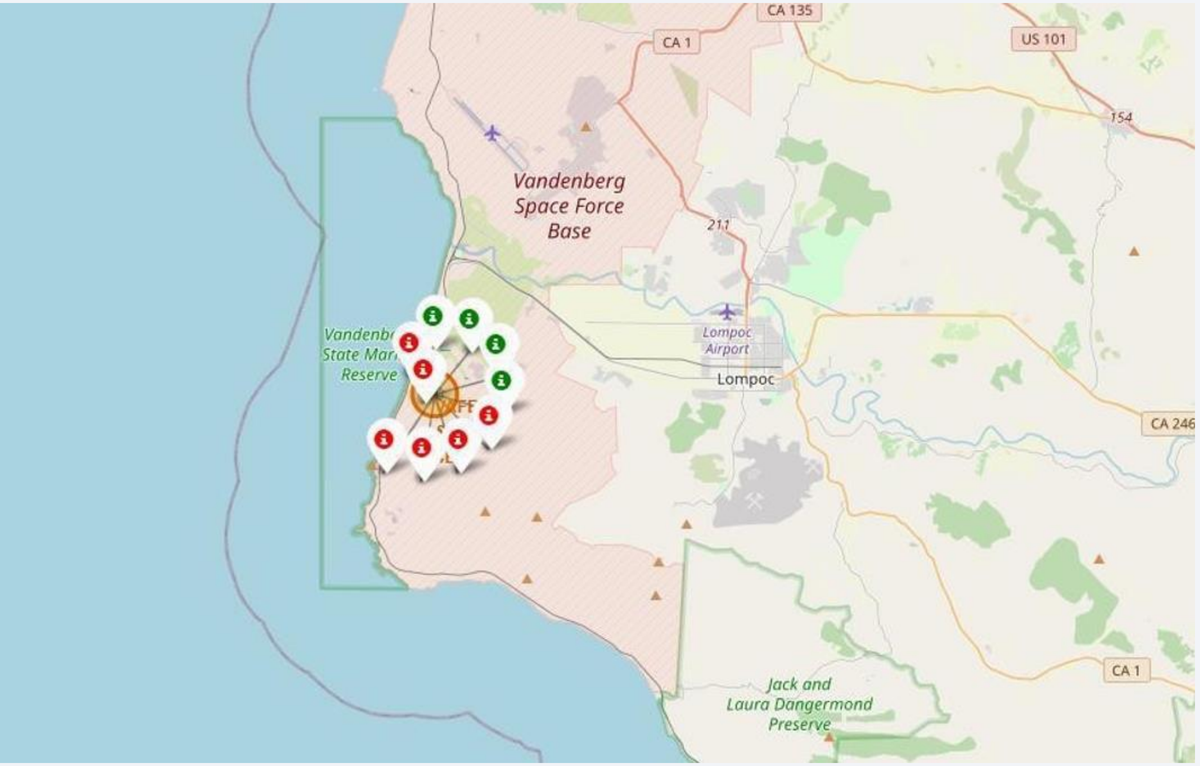
Launch Site Locations





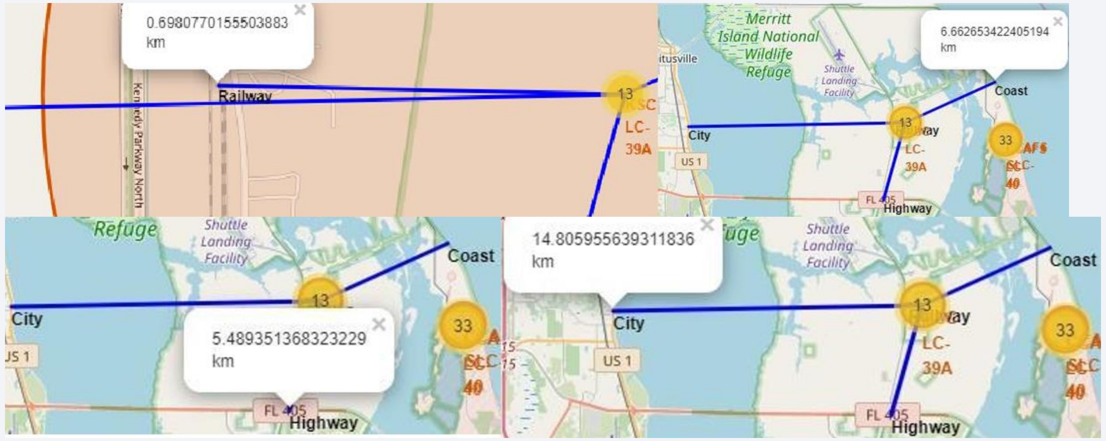
The left map shows all launch sites relative US map. The right map shows the two Florida launch sites since they are very close to each other. All launch sites are near the ocean.

Color Coded Launch Marker

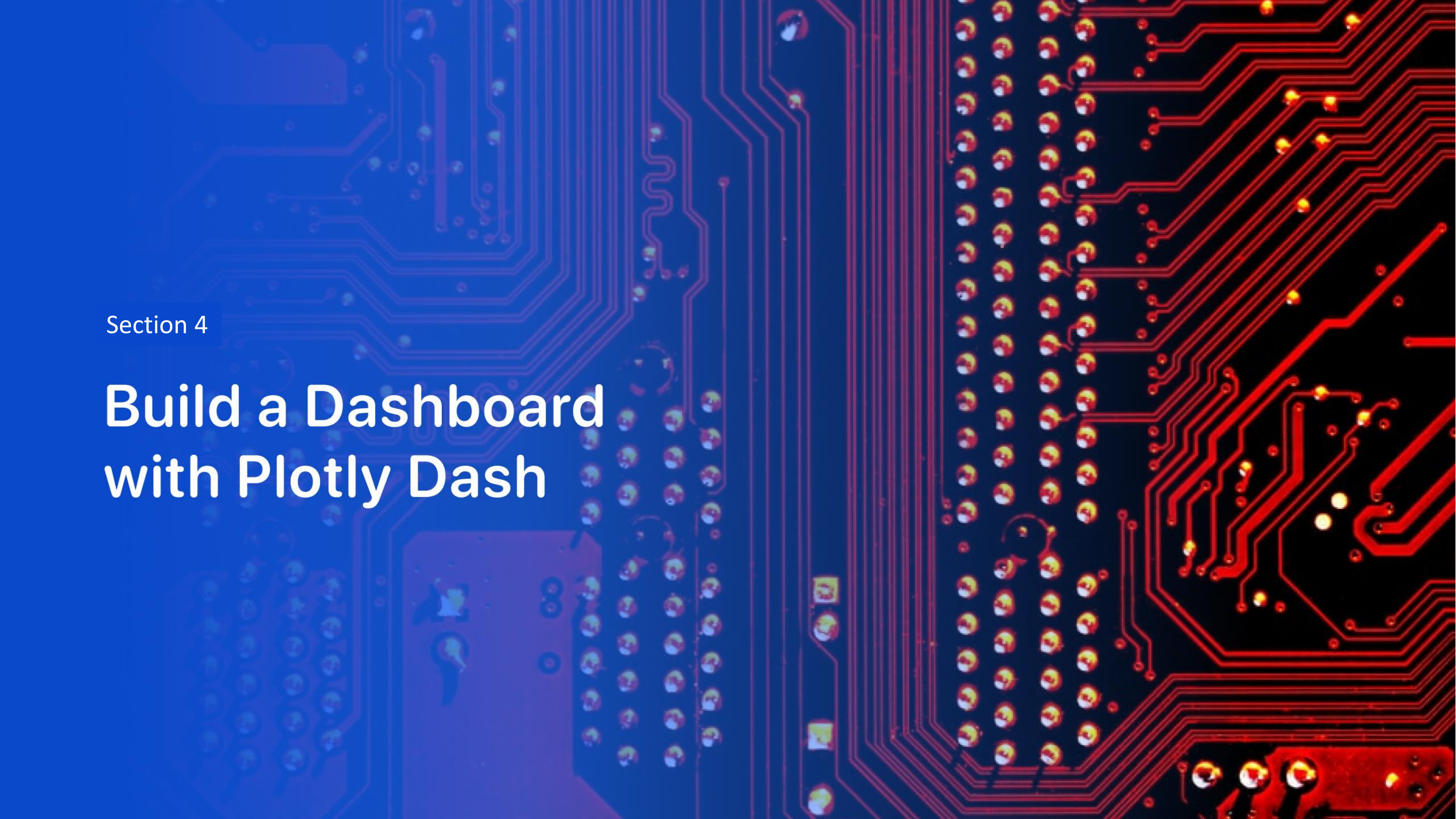


Clusters on Folium map can be clicked on to display each successful landing (green icon) and failed landing (red icon). In this example VAFB SLC-4E shows 4 successful landings and 6 failed landings.

Key Location Proximity



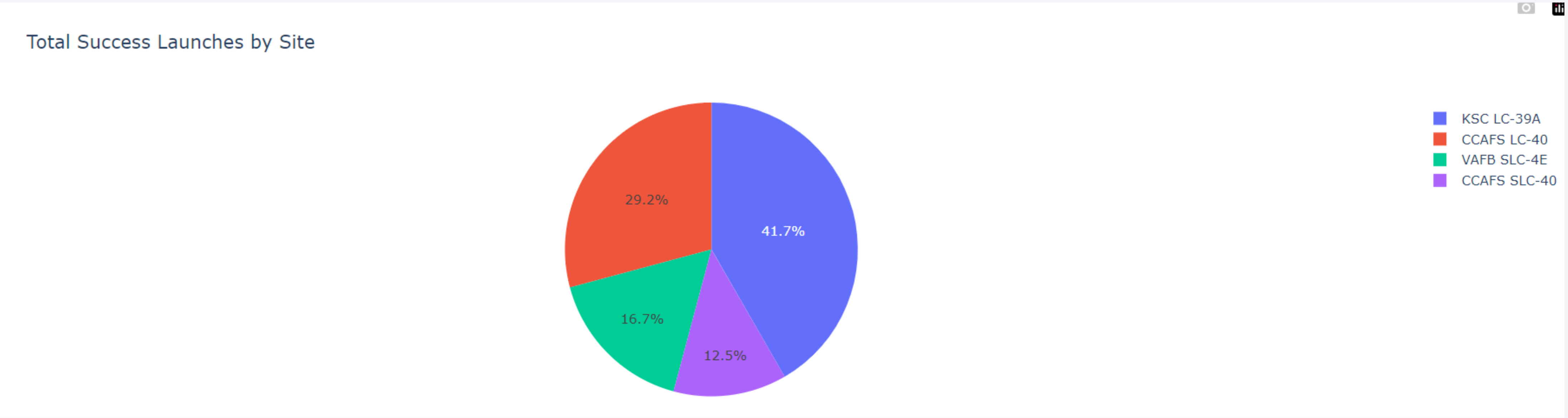
Using KSC LC-39A as an example, launch sites are very close to railways for large part and supply transportation. Launch sites are close to highways for human and supply transport. Launch sites are also close to coasts and relatively far from cities so that launch failures can land in the sea to avoid rockets falling on densely populated areas.



Section 4

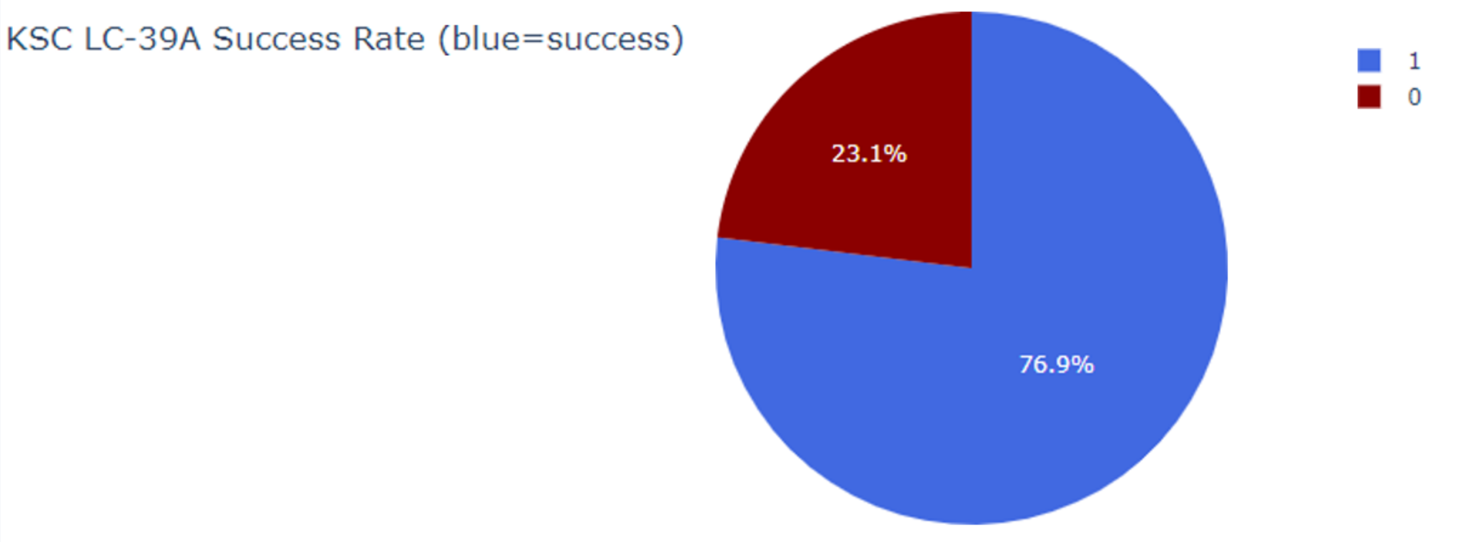
Build a Dashboard with Plotly Dash

Successful Launches Across Launch Sites



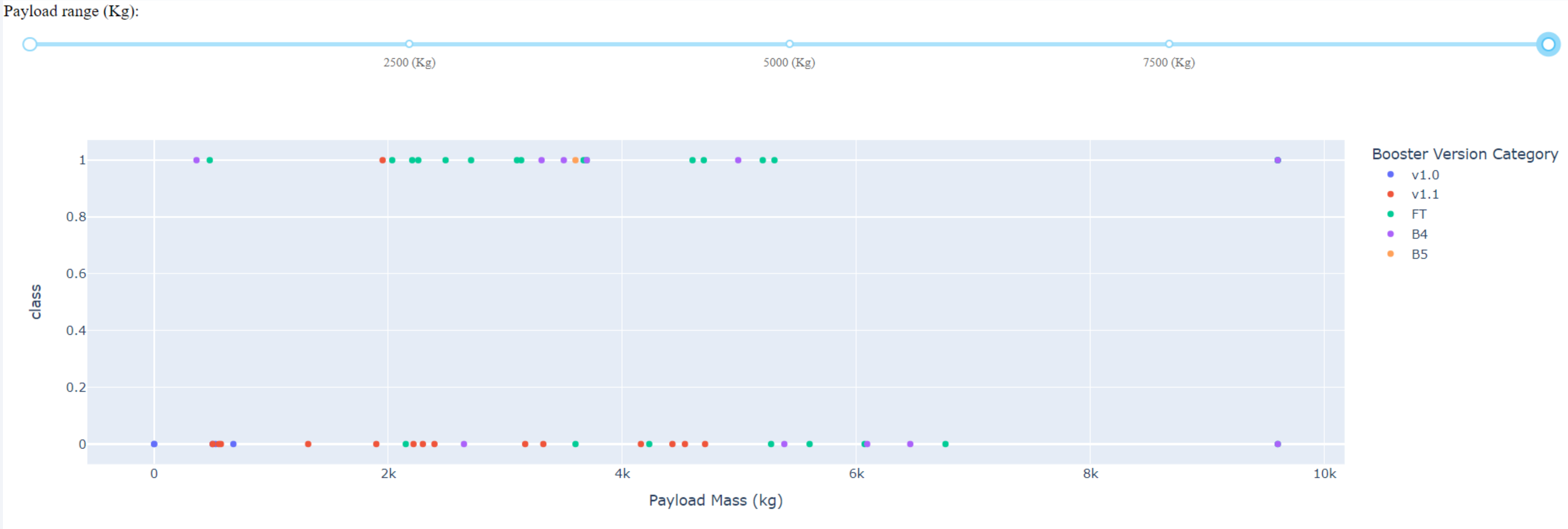
This is the distribution of successful landings across all launch sites. CCAFS LC-40 is the old name of CCAFS SLC-40 so CCAFS and KSC have the same amount of successful landings, but a majority of the successful landings where performed before the name change. VAFB has the smallest share of successful 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 Category



Payload range (Kg):

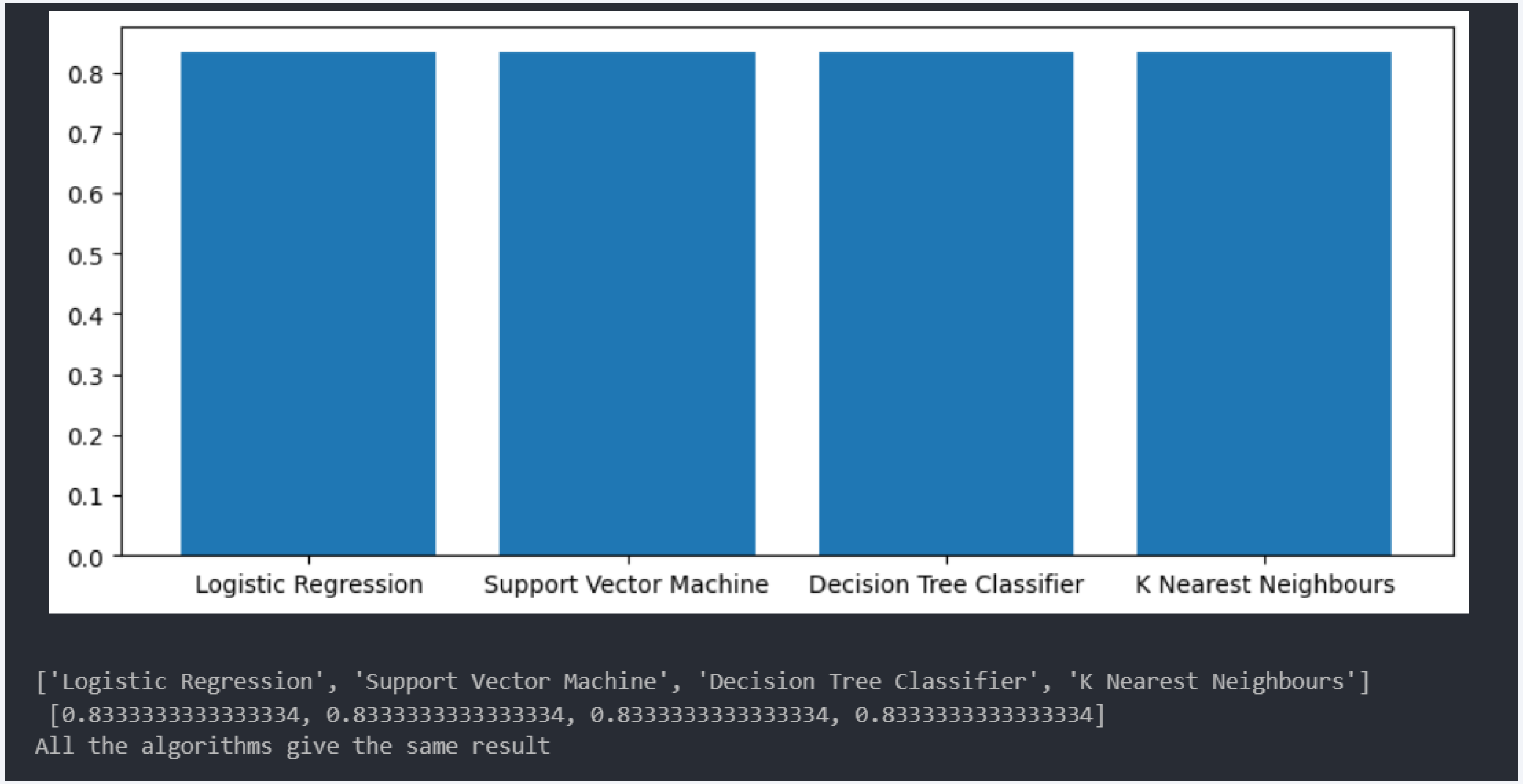
Plotly dashboard has a Payload range selector. However, this is set from 0-10000 instead of the max Payload of 15600. Class indicates 1 for successful landing and 0 for failure. Scatter plot also accounts for booster version category in color and number of launches in point size. In this particular range of 0-6000, interestingly there are two failed landings with payloads of zero kg.

Section 5

Predictive Analysis (Classification)

Classification Accuracy

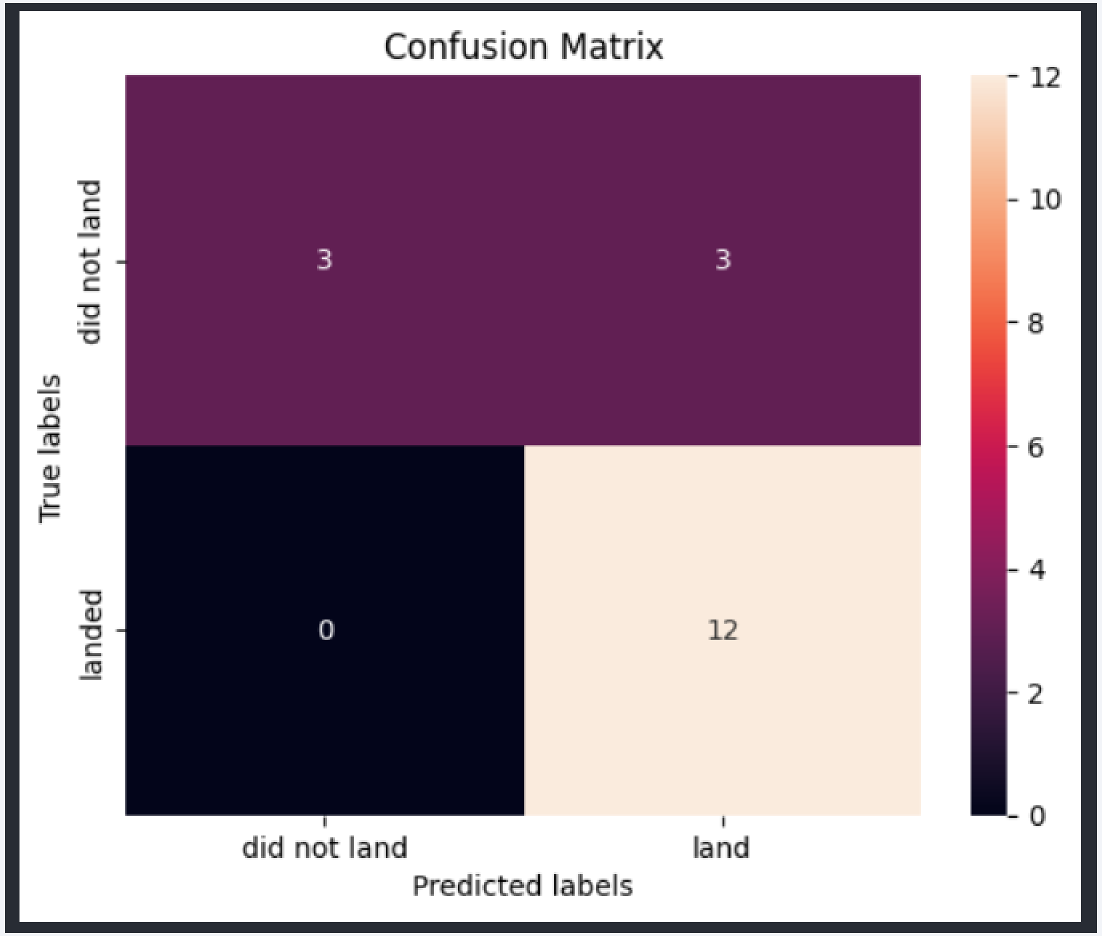
- All models had virtually the same accuracy on the test set at 83.33% accuracy.
- It should be noted that test size is small at only sample size of 18.
- This can cause large variance in accuracy results, such as those in Decision Tree model in repeated runs.
- We likely need more data to determine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



```
['Logistic Regression', 'Support Vector Machine', 'Decision Tree Classifier', 'K Nearest Neighbours']
 [0.8333333333333334, 0.8333333333333334, 0.8333333333333334, 0.8333333333333334]
All the algorithms give the same result
```

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 repository url:

- https://github.com/AvijeetPaul/IBM-Data-Science-Capstone-Project

Instructors:

- *Rav Ahuja, Alex Aklson, Aije Egwaikhide, Svetlana Levitan, Romeo Kienzler, Polong Lin, Joseph Santarcangelo, Azim Hirjani, Hima Vasudevan, Saishruthi Swaminathan, Saeed Aghabozorgi, Yan Luo*

Special Thanks to All Instructors:

- https://www.coursera.org/professional-certificates/ibm-data-science?#instructors

Thank you!