IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Tong Jing Ning
24/ 09/ 2024

Outline

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

Executive Summary

- Summary of methodologies
  - SpaceX Falcon 9 Launches data collection through web GET requests from SpaceX V4 API
  - SpaceX Falcon 9 Launches data collection through web scraping using BeautifulSoup4
  - SpaceX Falcon 9 Launches data wrangling with Python Pandas
  - SpaceX Falcon 9 Launches Exploratory Data analysis with SQL, Pandas, Matplotlib / Seaborn
  - SpaceX Falcon 9 Launch Sites analysis with Folium (Interactive Map)
  - SpaceX Falcon 9 Launch Sites analysis with Plotly Dash
  - SpaceX Falcon 9 Launch Prediction with Machine Learning
- Summary of all results
  - EDA Results
  - Interactive Visual Analytics and Dashboard
  - Classification Predictive Analysis

Introduction

- Project Background and Context

SpaceX advertises Falcon 9 rocket launches on its website with a cost of 62 million dollars each. Much of this saving is because SpaceX can reuse the first stage. Therefore, if the first stage can be determined to land, we can determine the cost of a launch. This will be useful in business setting, for example when an alternate company wants to bid against SpaceX for a rocket launch.

- Problems you want to find answers

Predict if the Falcon 9 first stage will land successfully using historical data from Falcon 9 rocket launches advertised on its website

Section 1

Methodology

Methodology

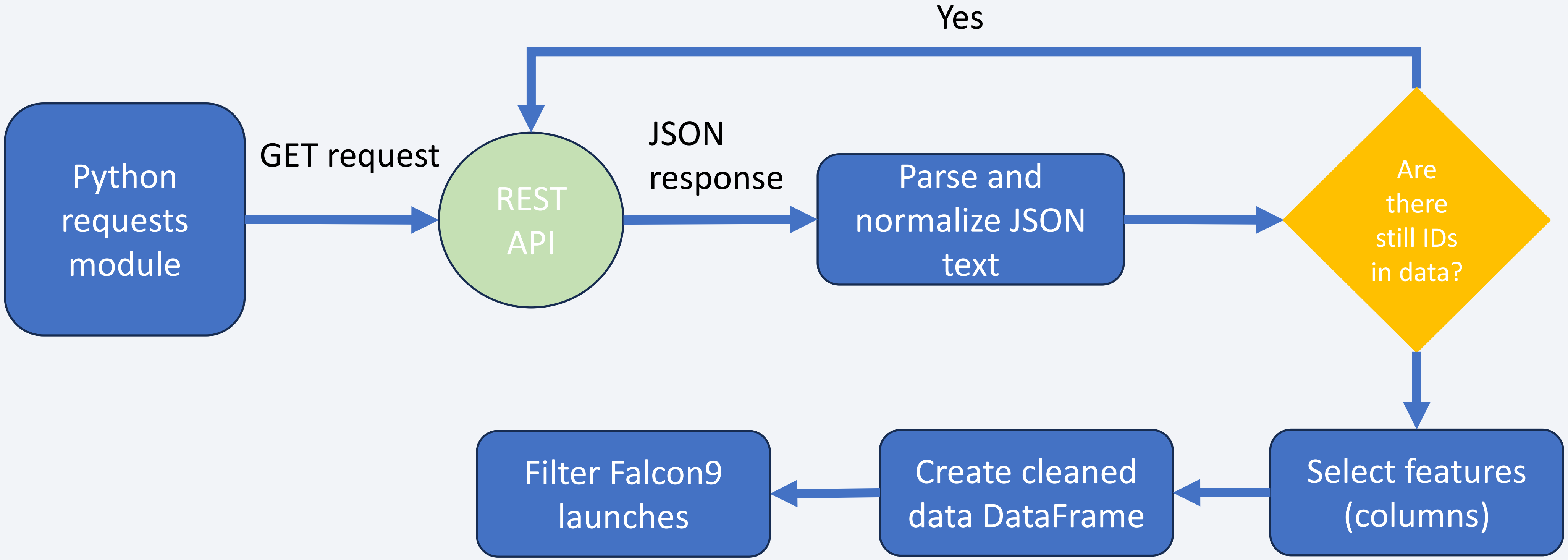
Executive Summary

- Data collection methodology:
  - Describe how data was collected
- Perform data wrangling
  - Describe how data was processed
- 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

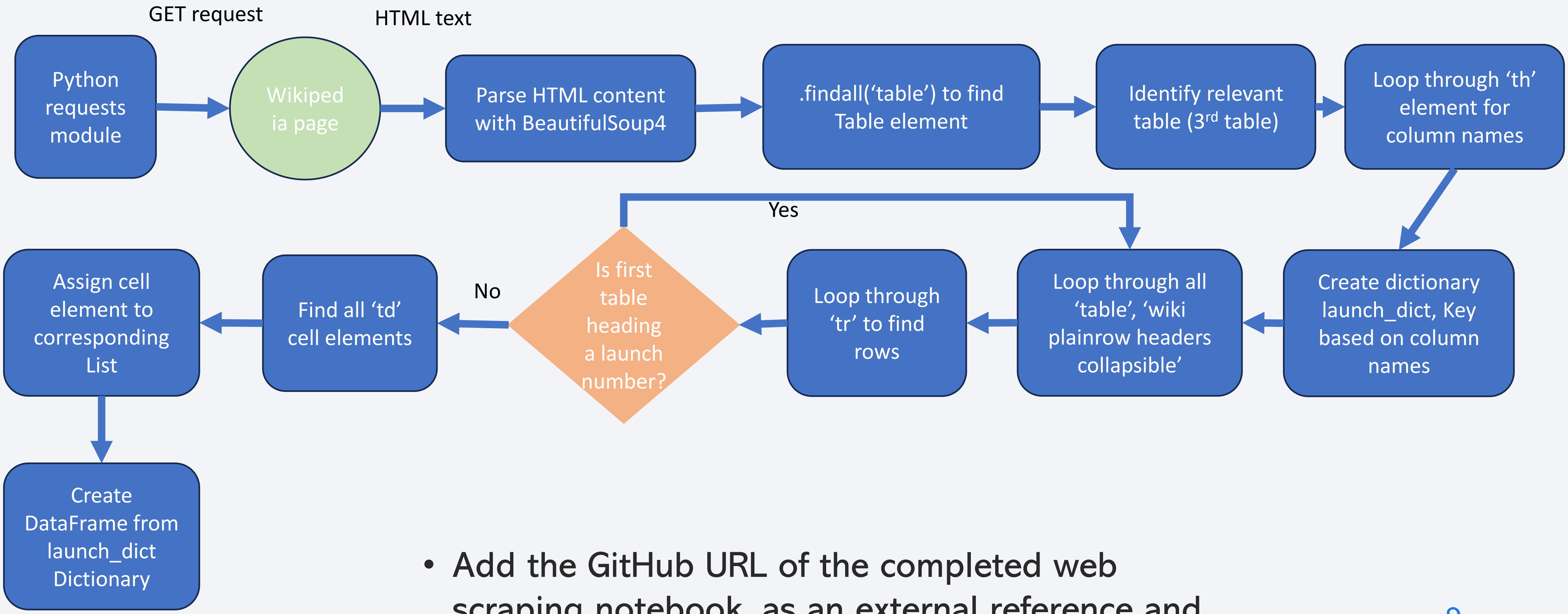
- Datasets comes from a variety of sources, namely the SpaceX API and Wikipedia which stores a table of past SpaceX launches.
- First, using the RESTful API of SpaceX, a GET request was made to the SpaceX API using the Python requests library. Upon successful status, a JSON text response was received. This was then parsed and normalized into a Pandas dataframe. These are all natively available through Python libraries.
- In terms of web scraping, a Wikipedia page titled "List of Falcon 9 and Falcon Heavy launches" was selected as it has launch records stored in a HTML table. BeautifulSoup4 Python library is a package suitable to extract information from HTML files. The HTML text was thus parsed and preprocessed into Pandas dataframe.

Data Collection – SpaceX API


Yes
Python requests module
GET request
REST API
JSON response
Parse and normalize JSON text
Are there still IDs in data?
Select features (columns)
Create cleaned data DataFrame
Filter Falcon9 launches

https://github.com/syriessw/DataScienceProjects/blob/main/FinalProjects/Capstone/SpaceX%20Falcon9%20Launches%20Prediction%20Part1.ipynb

Data Collection - Scraping


GET request
HTML text
Python requests module
Wikiped ia page
Parse HTML content with BeautifulSoup4
.findall('table') to find Table element
Identify relevant table (3rd table)
Loop through 'th' element for column names
Create dictionary launch_dict, Key based on column names
Loop through all 'table', 'wiki plainrow headers collapsible'
Loop through 'tr' to find rows
Is first table heading a launch number?
Yes
No
Find all 'td' cell elements
Assign cell element to corresponding List
Create DataFrame from launch_dict Dictionary

- Add the GitHub URL of the completed web scraping notebook, as an external reference and peer-review purpose

Data Wrangling

- Using Pandas, there are native functions we can use. Using ‘isnull().sum() allow a quick look at which columns contain empty (NULL) values. These were ‘PayloadMass’ and ‘LandingPad’.
- From research, we understand that ‘LandingPad’ can be absent during launches, so NULL values are valid.
- For ‘PayloadMass’, there should not be missing values. Thus, we calculate the mean value of ‘PayloadMass’ and substitute all NULL values with the mean.
- To identify which features would be useful, we took a look at the number of ‘LaunchSites’ and ‘Orbit’ with value_counts.
- Finally, the objective is to predict if the SpaceX launch will succeed or fail, which is equivalent to the ‘Outcome’ column. To simplify the target variable for prediction, we will recategorize into binary [0, 1] where 0 = Fail and 1 = Success.
- All ‘False’ and ‘None’ prefix in ‘Outcome’ tables were used to generate ‘0’ in a new column ‘Class’. Remaining values were filled with ‘1’.

https://github.com/syriessw/DataScienceProjects/blob/main/FinalProjects/Capstone/SpaceX%20Falcon9%20Launches%20Prediction%20Part3.ipynb

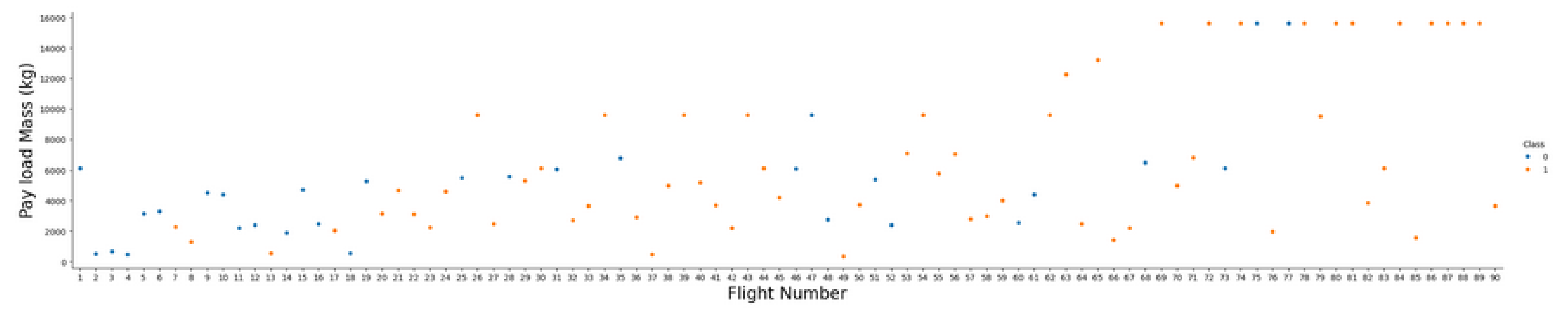
EDA with Data Visualization

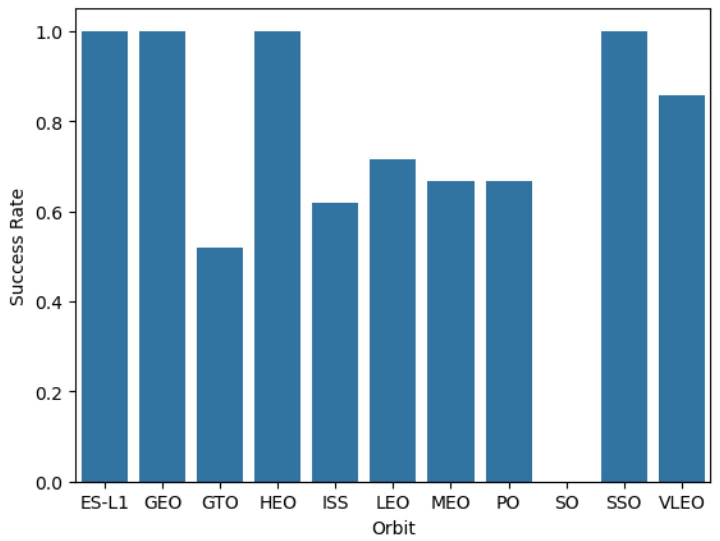
- EDA with MatplotLib / Seaborn
- Scatter plots to show relationship between two variables
- Bar chart to show breakdown of categorical variable
- Line chart to show trends across time (Year)
  - Scatterplot of FlightNumber vs Payload
    - The chart tells us as Flight number increases, the first stage is more likely to succeed. Similarly, at heavier Payloads, the flights were more successful.
  - Scatterplot of FlightNumber vs LaunchSite
    - The chart tells us as Flight number increases, the number of successes increases. We can also see that the most successful launch site is KSC LC 39A

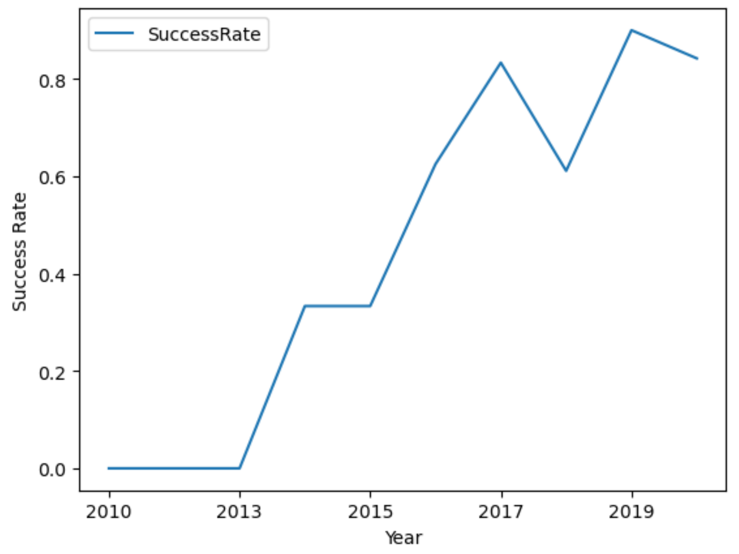
- Scatterplot of PayloadMass vs Launchsite
  - The chart tells us that for VAFB-SLC launch site, no rockets launched for heavy payload mass.
- Bar charts on categorical variable (Orbit)
  - Highest success rates orbits are – ES-L1, GEO, HEO and SSO.
- Scatterplot of FlightNumber and Orbit Type
  - In LEO orbit, the success seems to be related to number of flights. In GTO orbit, we cannot see any relationship
- Line chart of Average success rate vs Year
  - Show the trend of mean success rate as the year goes. After 2013, the success rate keeps increasing.

https://github.com/syriessw/DataScienceProjects/blob/main/FinalProjects/Capstone/SpaceX%20Falcon9%20Launches%20Prediction%20Part5.ipynb

EDA with Data Visualization (Cont)







https://github.com/syriessw/DataScienceProjects/blob/main/FinalProjects/Capstone/SpaceX%20Falcon9%20Launches%20Prediction%20Part5.ipynb

EDA with SQL

- Get all distinct Launch Sites
- Find 5 launch records beginning with ‘CCA’ as launch site
- Find Total payload carried by boosters by NASA (CRS)
- Find average payload mass carried by booster version F9 v1.1
- Find the first successful landing outcome in ground pad
- Find the names of boosters that have successes in drone ship and have payload mass >= 4000 and <= 6000
- List total number of successful and failure mission outcomes
- List name of booster_versions that have carried the maximum payload
- List records which will display the month names, failure landing outcomes in drone ship, booster versions, launch site for the months in 2015.
- Rank the count of landing outcome between 2010-06-04 and 2017-03-20 in descending order

https://github.com/syriessw/DataScienceProjects/blob/main/FinalProjects/Capstone/SpaceX%20Falcon9%20Launches%20Prediction%20Part4.ipynb

Build an Interactive Map with Folium

- Summarize what map objects such as markers, circles, lines, etc. you created and added to a folium map
- In order to identify the launch sites, a marker popup and circle was created at each launch site.
- Further markers in a marker cluster was added for all flights taken, with color red to represent failure, and green to represent success. These would be clustered around launch sites.
- There are also additional markers added for the geographical features (coastline, railway, highway and nearest city) to the CCAFS SLC-40 launch site. A line is drawn from each feature to the launch site to show the absolute distance to the point. This helps us to explain if there is any rationale behind the location of the launch sites. We determine that launch sites must be far away from city, but near enough to railway, coastline and highway.

https://github.com/syriessw/DataScienceProjects/blob/main/FinalProjects/Capstone/SpaceX%20Falcon9%20Launches%20Prediction%20Part6.ipynb

Build a Dashboard with Plotly Dash

- The Dashboard contains an interactable dropdown for options of All launch sites or a specific one, and a slider for PayloadMass.
- There is a pie chart to showcase:
  - Successful launches of all sites
  - Distribution of launches (success and failure) of each site (determined by drop down)
- There is a scatter plot to showcase:
  - Launches (Class) vs PayloadMass by Booster Version category (filterable by payload mass)
- The pie chart tells us which site has the largest successes and which sites has the highest success rate
- From the scatter plot, we can find out which payload range has the highest / lowest success, which booster version has the highest success rates and what range of payload affects the success rate.

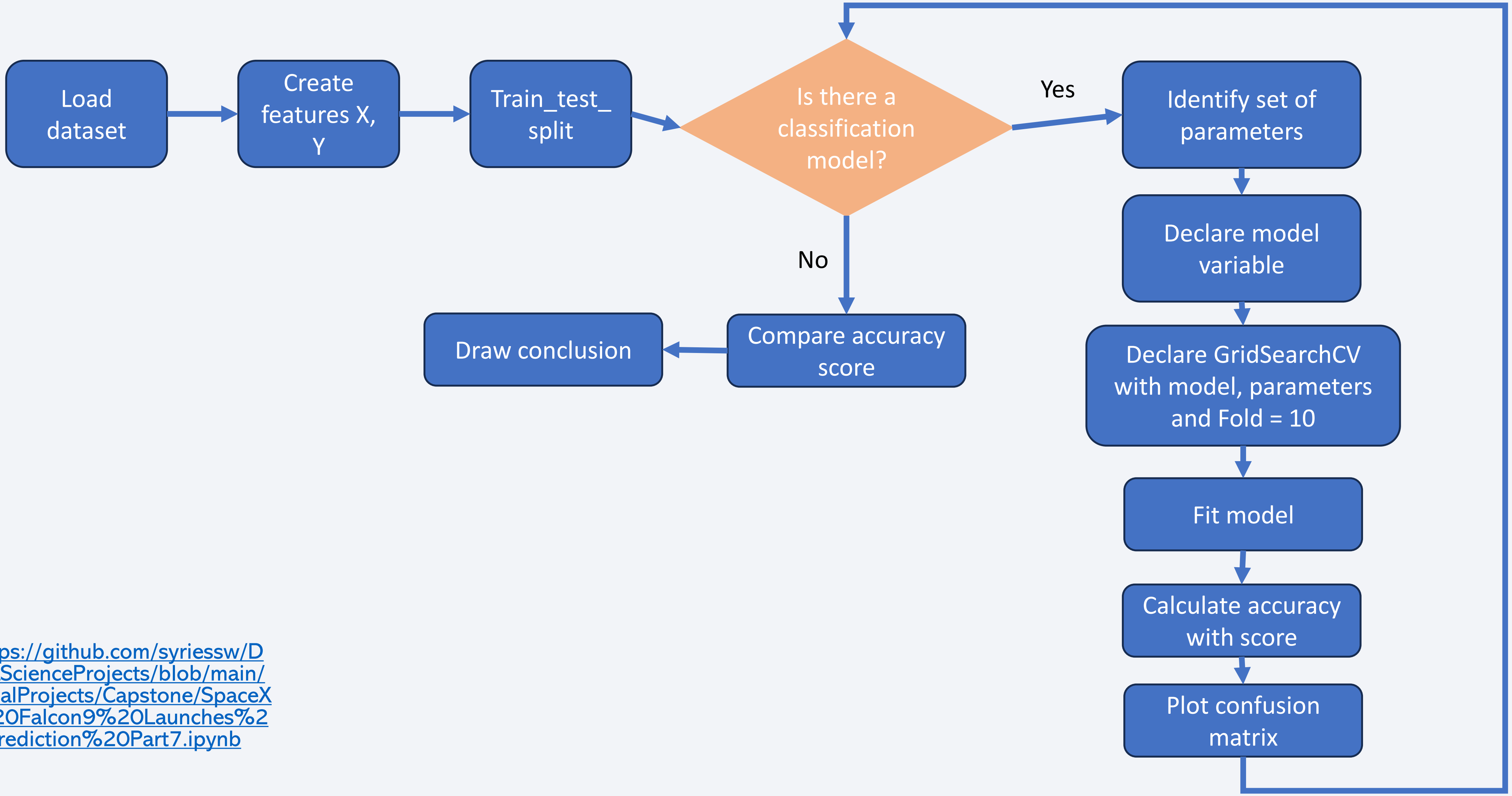
https://github.com/syriessw/DataScienceProjects/blob/main/FinalProjects/Capstone/spacex_dash_app.py

Predictive Analysis (Classification)

- After identifying the features, we being the analysis phase. We start off by declaring the dependant variables (X) and the target variable (Y). We will utilize sklearn machine learning library for the models.
- Use train_test_split function to split the dataset into training and test samples. We retain 20% of dataset for testing.
- Identify the classification models for training and evaluating
  - Logistic Regression
  - Support Vector Machine
  - Decision Tree Classifier
  - K-Nearest Neighbor
- Identify the parameters to implement to search for hyperparameters
- For each classification model
  - use GridSearch to find the best hyperparameters
  - Fit best parameters model with training data
  - Predict yhat variable with best model
  - Calculate accuracy, and observe the confusion matrix
- Compare the accuracy and confusion matrix of all models

https://github.com/syriessw/DataScienceProjects/blob/main/FinalProjects/Capstone/SpaceX%20Falcon9%20Launches%20Prediction%20Part7.ipynb

Predictive Analysis (Classification) (Cont)



Load dataset

Create features X, Y

Train_test_split

Is there a classification model?

Yes

Identify set of parameters

Declare model variable

Declare GridSearchCV with model, parameters and Fold = 10

Fit model

Calculate accuracy with score

Plot confusion matrix

No

Compare accuracy score

Draw conclusion

https://github.com/syriessw/DataScienceProjects/blob/main/FinalProjects/Capstone/SpaceX%20Falcon9%20Launches%20Prediction%20Part7.ipynb

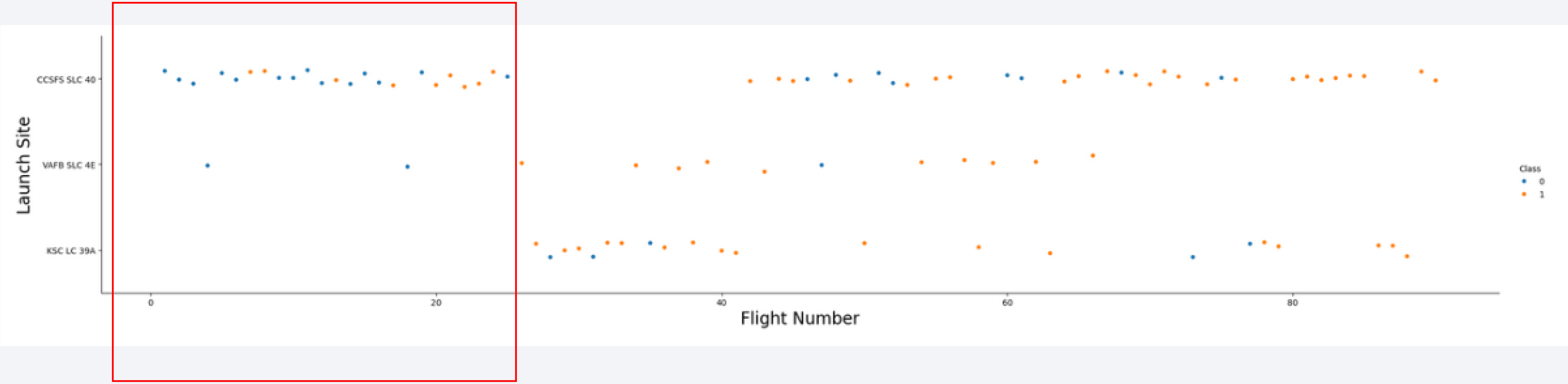
Results

- From EDA we found that:
  - There is a relationship between PayloadMass, LaunchSites and Orbit
  - The success rate of the launches started going up after 2013.
- From the Interactive Folium map, we found that:
  - Launch sites have to be close to coastline, railway and highway, but far away from cities.
- From the Interactive Plotly Dash dashboard, we found that:
  - The most successful launch site is KSC LC-39A
  - The highest success rate launch site is CCAFS LC-40
  - The heavier the payload, the more the number of successes
- From our evaluation of the different classification algorithm:
  - We find that all of them are similar in accuracy to predict whether the launches can succeed or fail.

Section 2

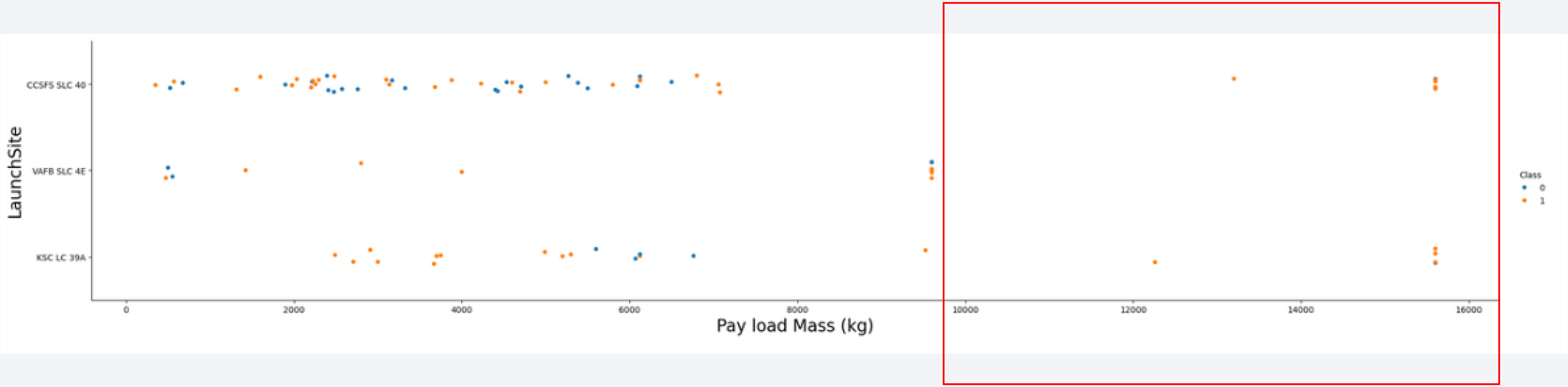
Insights drawn from EDA

Flight Number vs. Launch Site



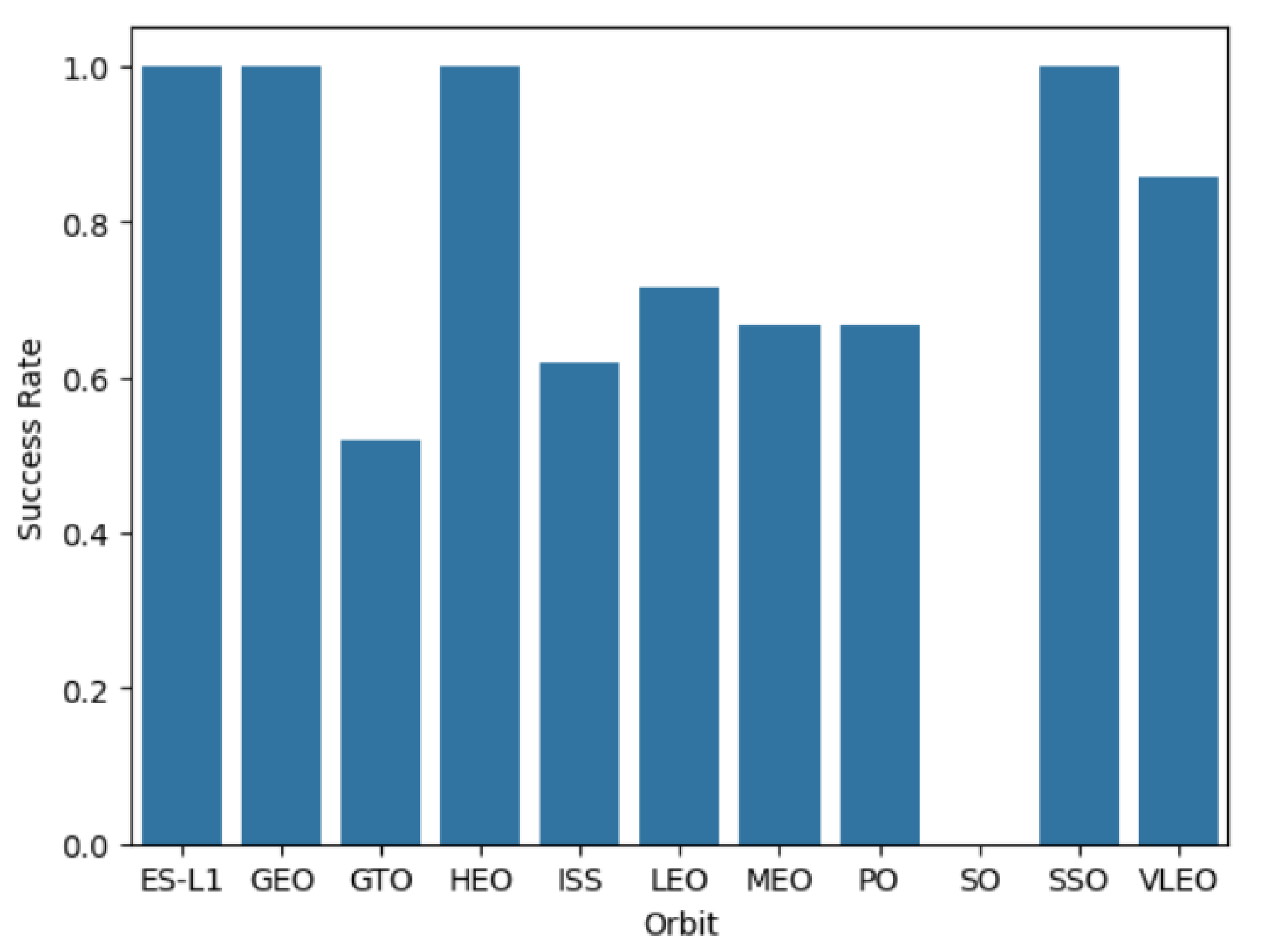
- Site launches at VAFB SLC 4E and KSC LC 39A were rarely/not used for the first few launches until Flight Number 20 onwards. As the Flight Number goes up, the number of successes increase. The ratio of Failures to Successes are better for VAFB SLC 4E and KSC LC 39A compared to CCSFS SLC 40 launch site. However, the usage of CCSFS SLC is consistent throughout all the Flight Numbers except around Flight Numbers 25 to 40, where majority of the launches are at KSC LC 39A")

Payload vs. Launch Site



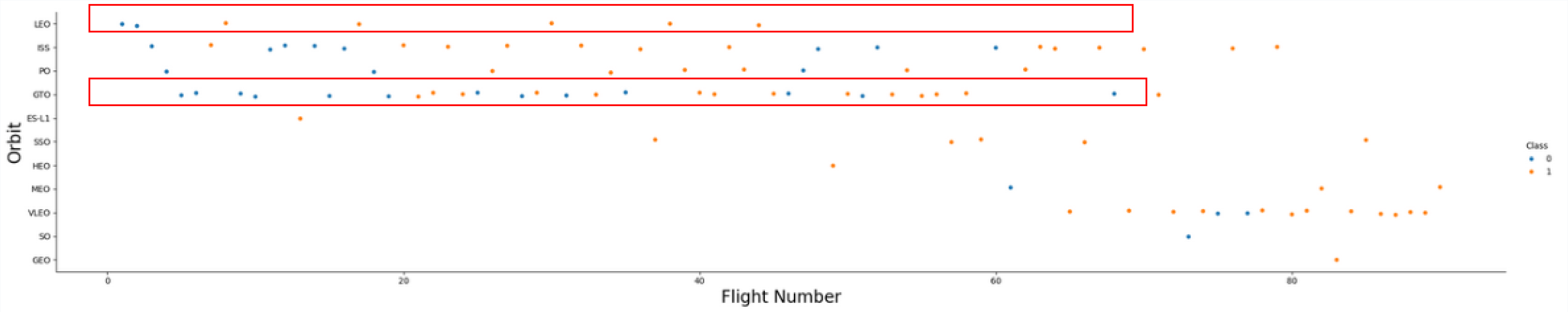
- Payloads greater than 10,000kg were the most successful across all launch sites.
- CCSFS SLC 40 carried the greatest variance of payloads.

Success Rate vs. Orbit Type



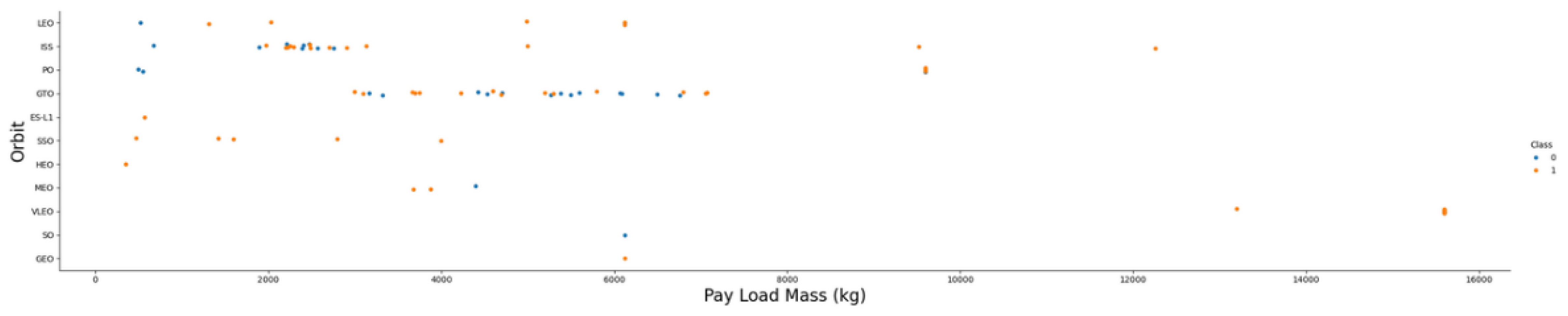
- Highest success rates are:
  - ES-L1
  - GEO
  - HEO
  - SSO

Flight Number vs. Orbit Type



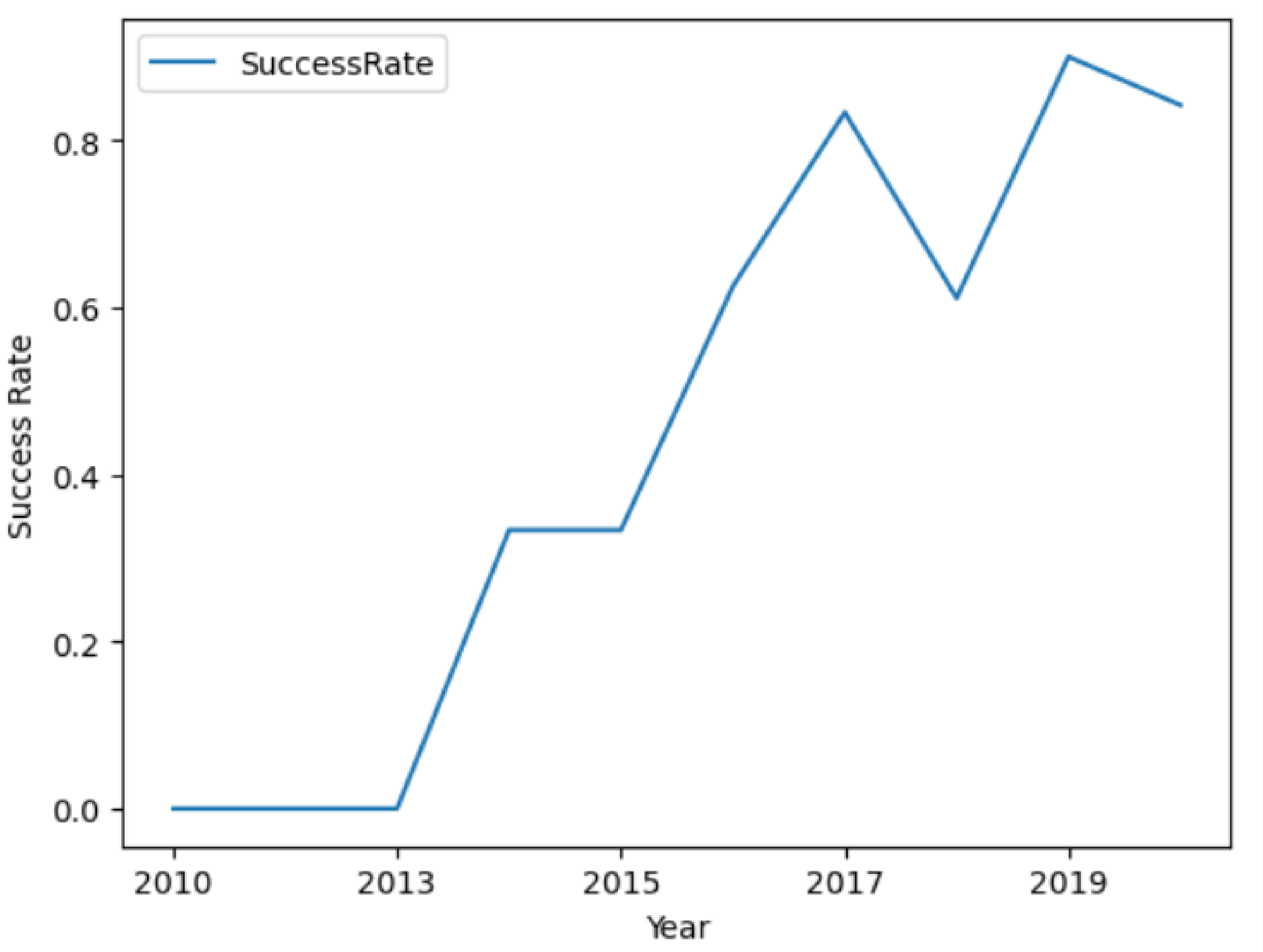
- LEO Orbit, as flight number increases, success increases
- GTO orbit there is no direct relationship between flight number and success rate

Payload vs. Orbit Type



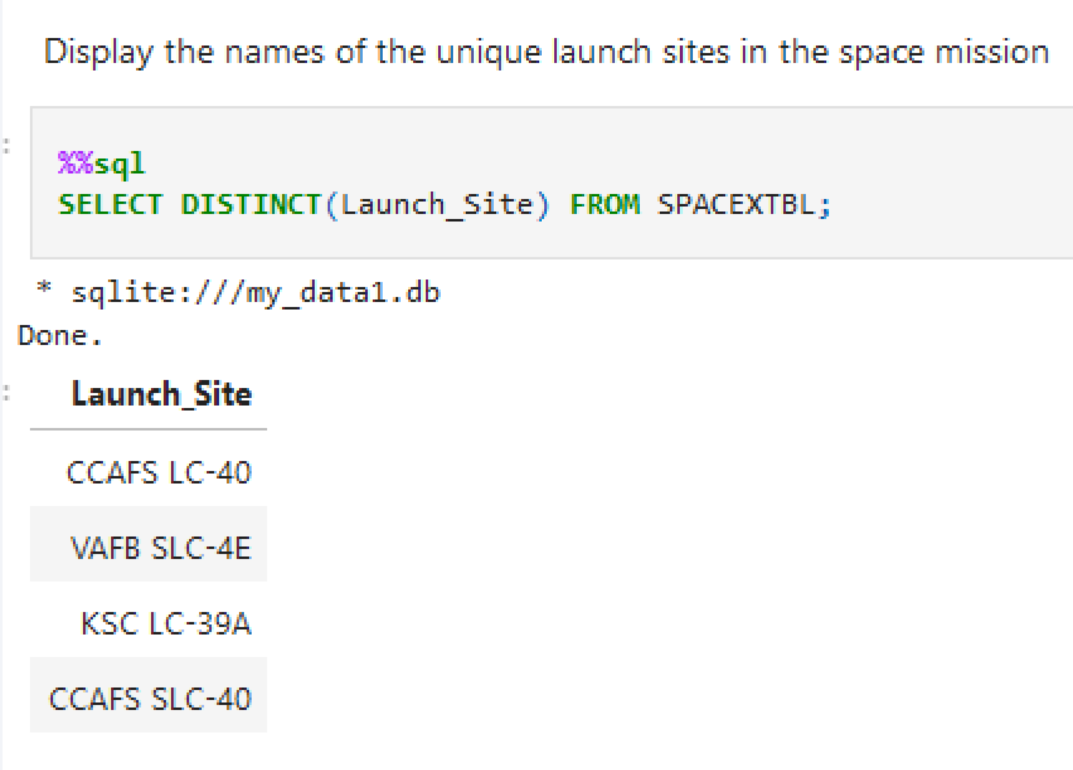
- With heavy payloads the successful landing or positive landing rate are more for Polar, LEO and ISS.
- However, for GTO, it's difficult to distinguish between successful and unsuccessful landings as both outcomes are present

Launch Success Yearly Trend



- From 2013, the success rate increase, except for 2018 where there is a dip before it follows the general trend of increasing in 2019

All Launch Site Names



Display the names of the unique launch sites in the space mission

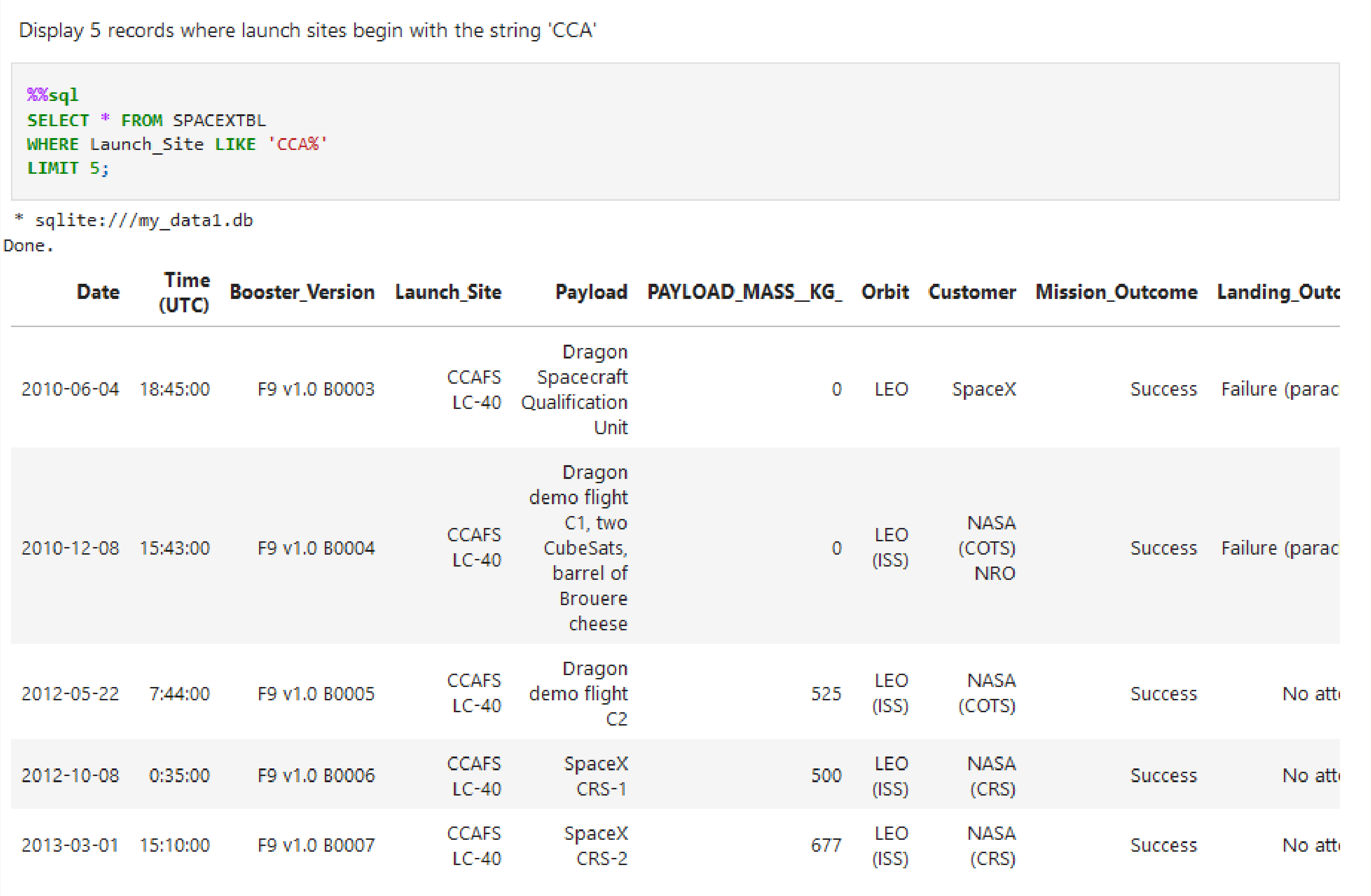
```
%%sql
SELECT DISTINCT(Launch_Site) FROM SPACEXTBL;
```

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

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

- Use DISTINCT to only pick each instance of a Launch Site in the column ‘Launch_Site’ from the table SPACEXTBL

Launch Site Names Begin with 'CCA'



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

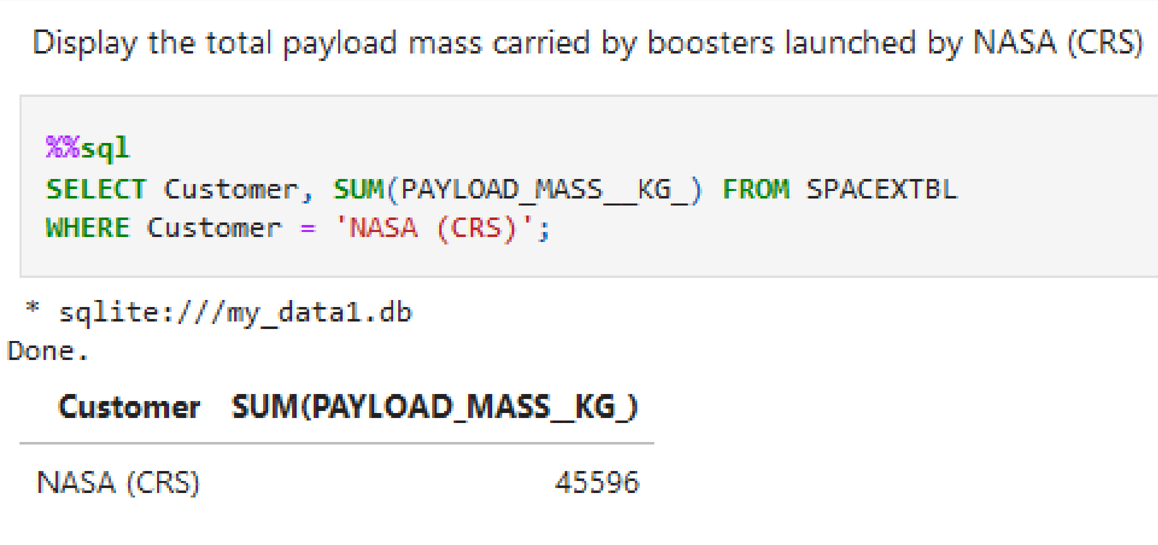
```
%%sql
SELECT * FROM SPACEXTBL
WHERE Launch_Site LIKE 'CCA%'
LIMIT 5;
```

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

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outc
2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parac
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 (parac
2012-05-22	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No att
2012-10-08	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No att
2013-03-01	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No att

- Use WHERE to filter the results
- Use ‘LIKE’ for text comparison and ‘%’ as a placeholder to indicate any text character. Hence ‘CCA%’ means a string beginning specifically with ‘CCA’ followed by any other character
- Use LIMIT to only return 5 records

Total Payload Mass



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

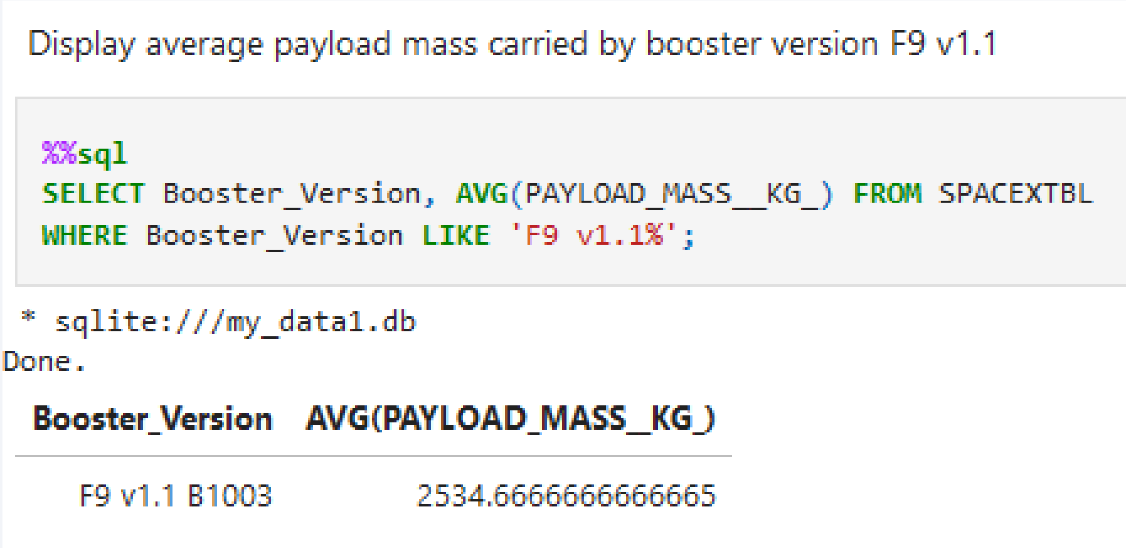
```
%%sql
SELECT Customer, SUM(PAYLOAD_MASS__KG_) FROM SPACEXTBL
WHERE Customer = 'NASA (CRS)';
```

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

Customer	SUM(PAYLOAD_MASS__KG_)
NASA (CRS)	45596

- Use SUM to aggregate all values under the column PAYLOAD_MASS__KG
- Use WHERE to specify by ‘NASA (CRS)’

Average Payload Mass by F9 v1.1



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

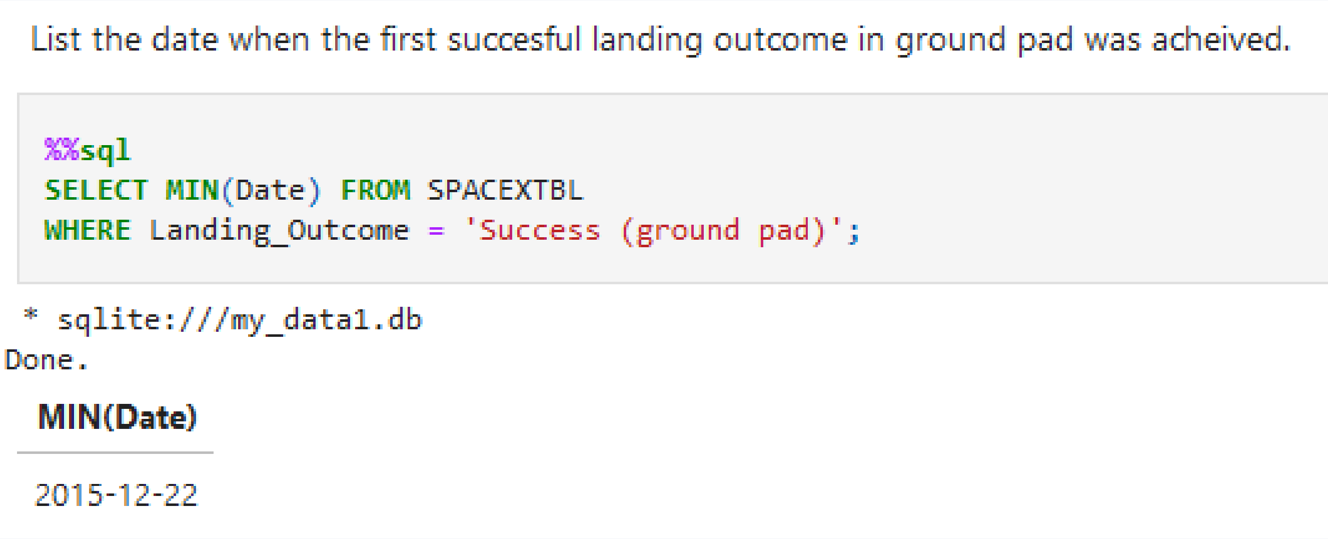
```
%%sql
SELECT Booster_Version, AVG(PAYLOAD_MASS__KG_) FROM SPACEXTBL
WHERE Booster_Version LIKE 'F9 v1.1%';
```

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

Booster_Version	AVG(PAYLOAD_MASS__KG_)
F9 v1.1 B1003	2534.6666666666665

- Use AVG to aggregate and calculate mean of PAYLOAD_MASS__KG_
- Use WHERE to filter the result

First Successful Ground Landing Date



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

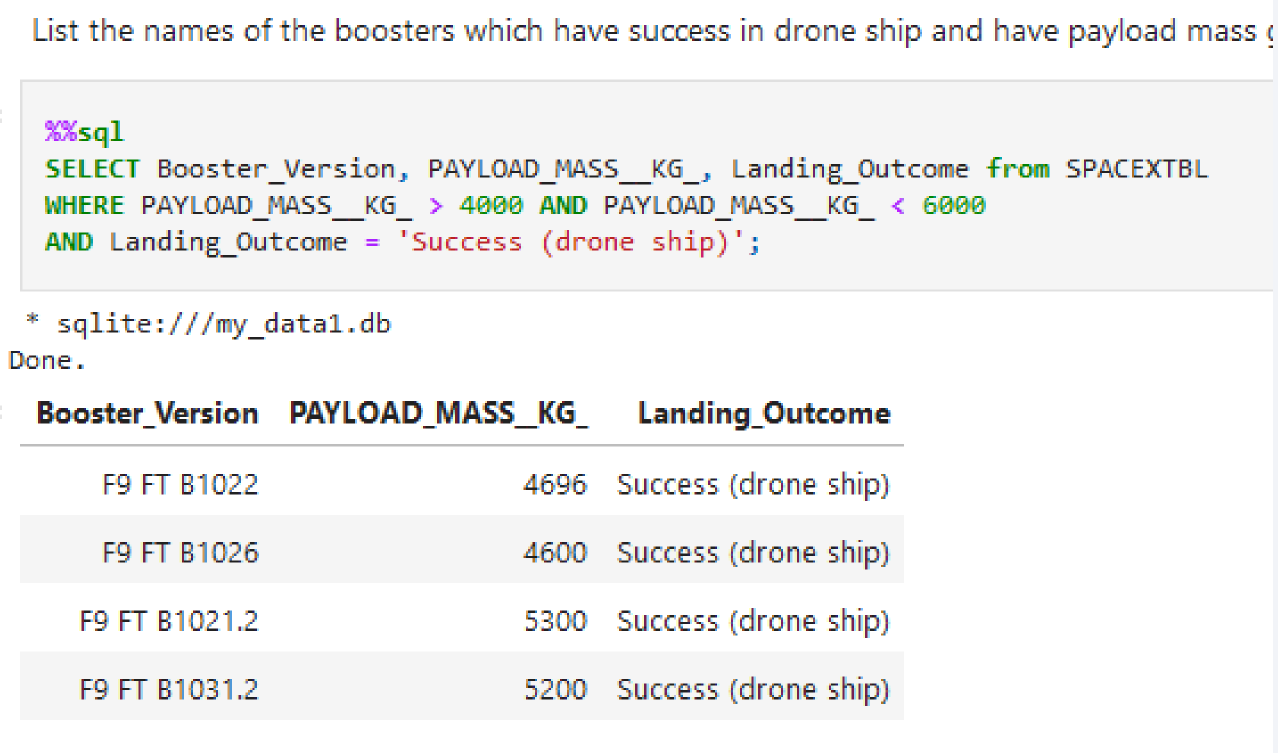
```
%%sql
SELECT MIN(Date) FROM SPACEXTBL
WHERE Landing_Outcome = 'Success (ground pad)';
```

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

MIN(Date)
2015-12-22

- Use MIN( ) function to calculate the smallest value from the Date column
- Use WHERE to filter the result

Successful Drone Ship Landing with Payload between 4000 and 6000



List the names of the boosters which have success in drone ship and have payload mass

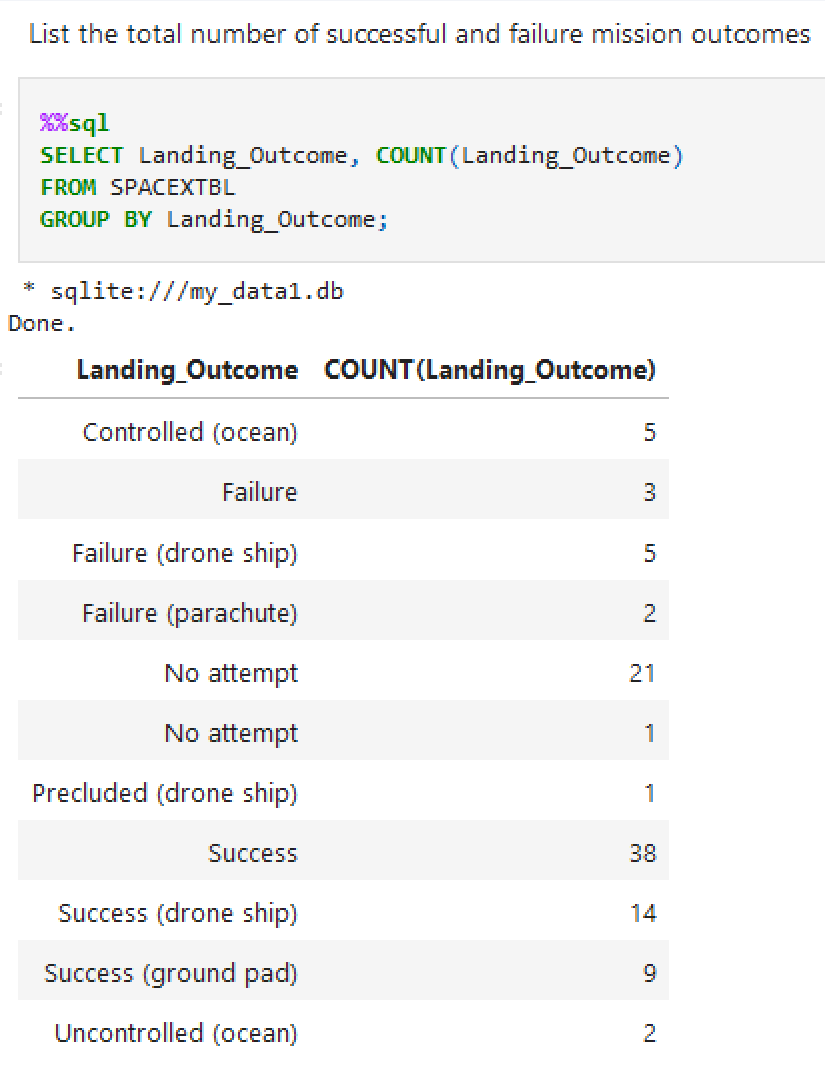
```
%%sql
SELECT Booster_Version, PAYLOAD_MASS__KG_, Landing_Outcome from SPACEXTBL
WHERE PAYLOAD_MASS__KG_ > 4000 AND PAYLOAD_MASS__KG_ < 6000
AND Landing_Outcome = 'Success (drone ship)';
```

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

Booster_Version	PAYLOAD_MASS__KG_	Landing_Outcome
F9 FT B1022	4696	Success (drone ship)
F9 FT B1026	4600	Success (drone ship)
F9 FT B1021.2	5300	Success (drone ship)
F9 FT B1031.2	5200	Success (drone ship)

- Use WHERE to filter result
- Use > to indicate greater than
- Use AND to indicate condition1 and condition2 must both equate to TRUE
- < to indicate smaller than

Total Number of Successful and Failure Mission Outcomes



List the total number of successful and failure mission outcomes

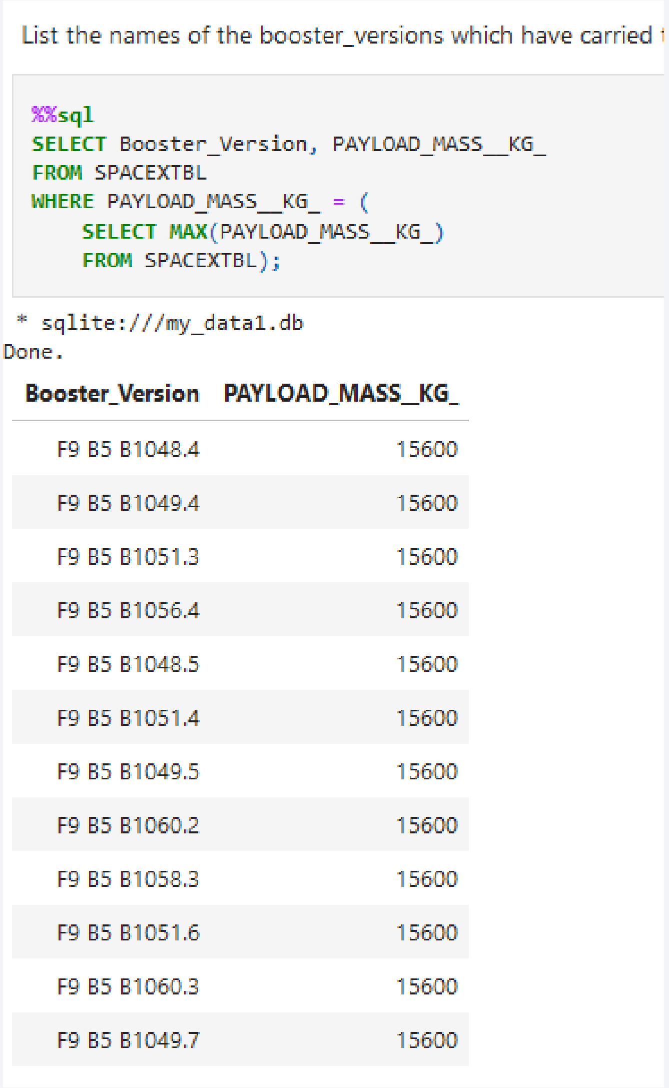
```
%%sql
SELECT Landing_Outcome, COUNT(Landing_Outcome)
FROM SPACEXTBL
GROUP BY Landing_Outcome;
```

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

Landing_Outcome	COUNT(Landing_Outcome)
Controlled (ocean)	5
Failure	3
Failure (drone ship)	5
Failure (parachute)	2
No attempt	21
No attempt	1
Precluded (drone ship)	1
Success	38
Success (drone ship)	14
Success (ground pad)	9
Uncontrolled (ocean)	2

- Use COUNT( ) function to obtain the number of values in the column Landing_Outcome
- Use GROUP BY to collapse/ group up all similar values into one row

Boosters Carried Maximum Payload



List the names of the booster_versions which have carried

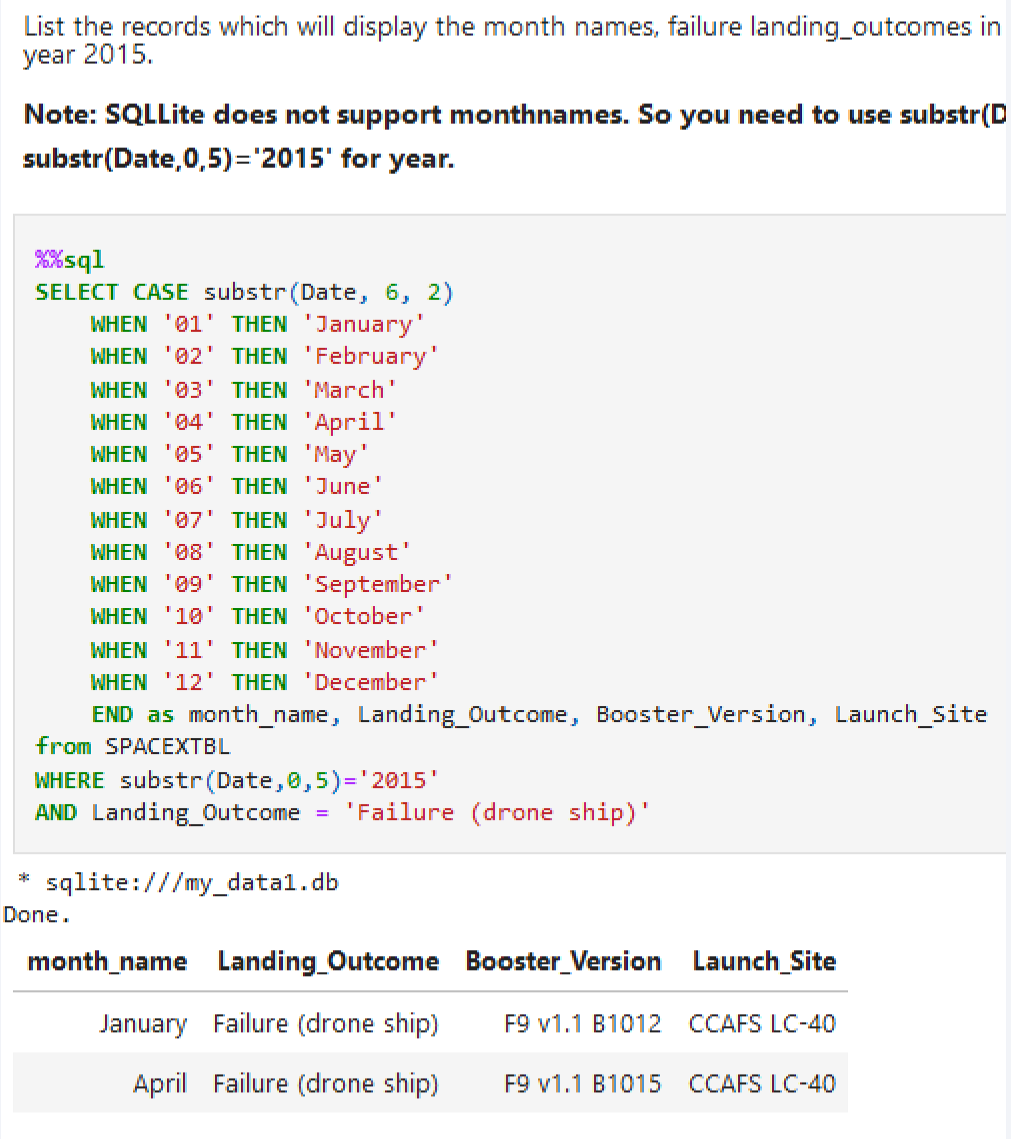
```
%%sql
SELECT Booster_Version, PAYLOAD_MASS__KG_
FROM SPACEXTBL
WHERE PAYLOAD_MASS__KG_ = (
    SELECT MAX(PAYLOAD_MASS__KG_)
    FROM SPACEXTBL);
```

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

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

- Use WHERE to filter result
- For the equivalent condition, use parantheses on another SQL query for a subquery
- Use MAX( ) function to obtain the largest value in the column PAYLOAD_MASS__KG_

2015 Launch Records



List the records which will display the month names, failure landing_outcomes in year 2015.

Note: SQLLite does not support monthnames. So you need to use substr(D substr(Date,0,5)='2015' for year.

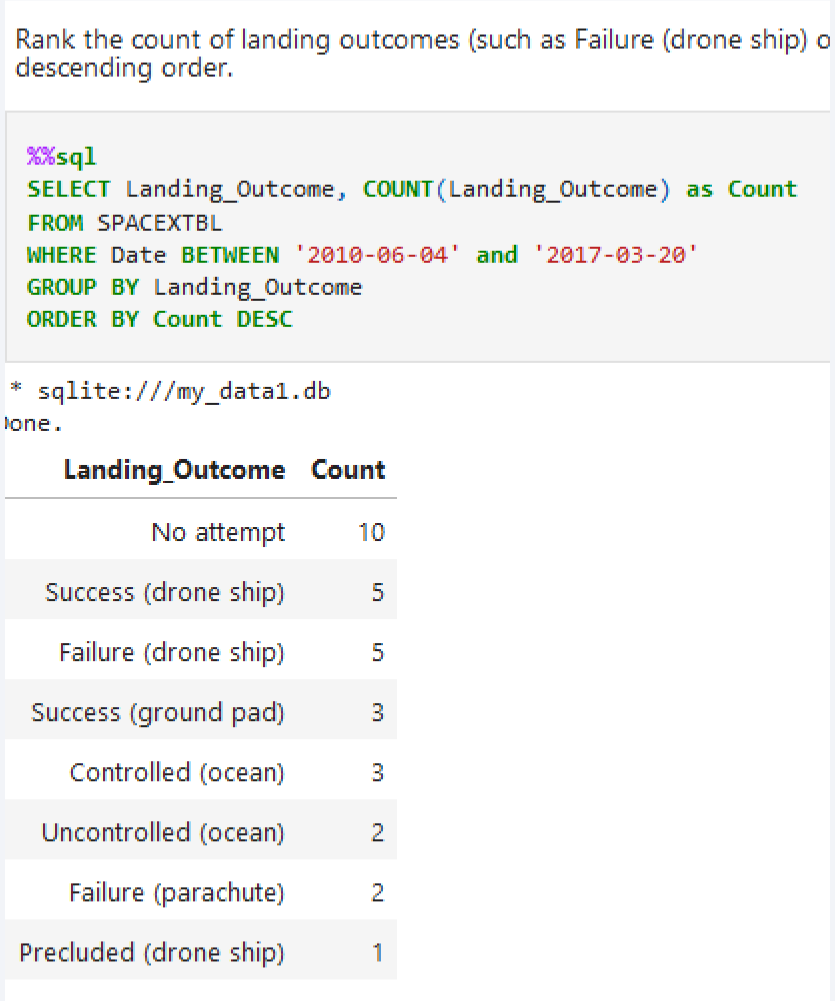
```
%%sql
SELECT CASE substr(Date, 6, 2)
    WHEN '01' THEN 'January'
    WHEN '02' THEN 'February'
    WHEN '03' THEN 'March'
    WHEN '04' THEN 'April'
    WHEN '05' THEN 'May'
    WHEN '06' THEN 'June'
    WHEN '07' THEN 'July'
    WHEN '08' THEN 'August'
    WHEN '09' THEN 'September'
    WHEN '10'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

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

month_name	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

- Use CASE, WHEN, THEN to do series of IF ELSE condition to replace numerical Month as long word Month Name

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



Rank the count of landing outcomes (such as Failure (drone ship) o
descending order.

```
%%sql
SELECT Landing_Outcome, COUNT(Landing_Outcome) as Count
FROM SPACEXTBL
WHERE Date BETWEEN '2010-06-04' and '2017-03-20'
GROUP BY Landing_Outcome
ORDER BY Count DESC
```

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

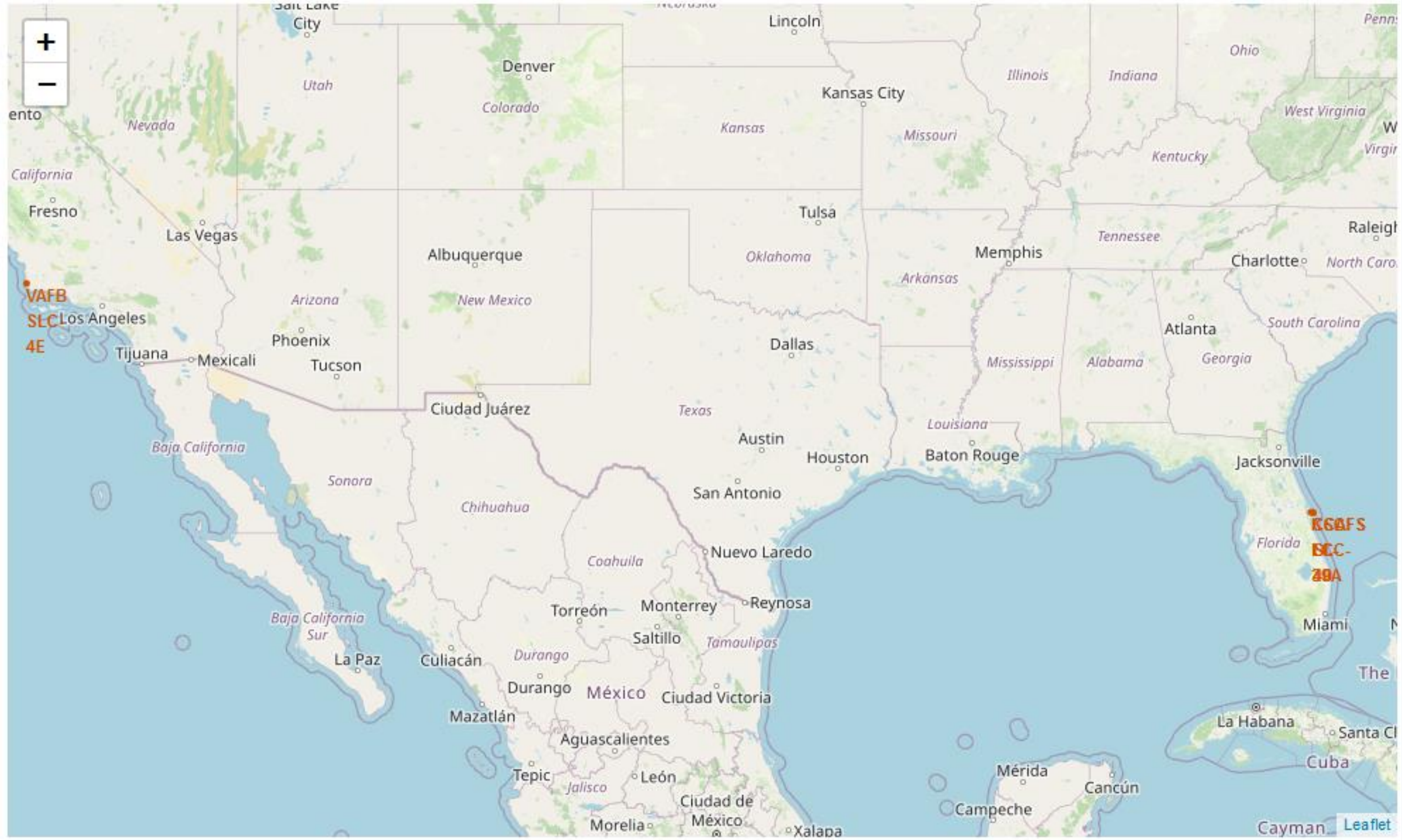
Landing_Outcome	Count
No attempt	10
Success (drone ship)	5
Failure (drone ship)	5
Success (ground pad)	3
Controlled (ocean)	3
Uncontrolled (ocean)	2
Failure (parachute)	2
Precluded (drone ship)	1

- Use COUNT ( ) function with AS to provide an alias on the aggregated value
- Use GROUP BY to group up each unique value of Landing_Outcome
- Use ORDER BY with DESC to sort the result rows by given column (Count) in descending order

Section 3

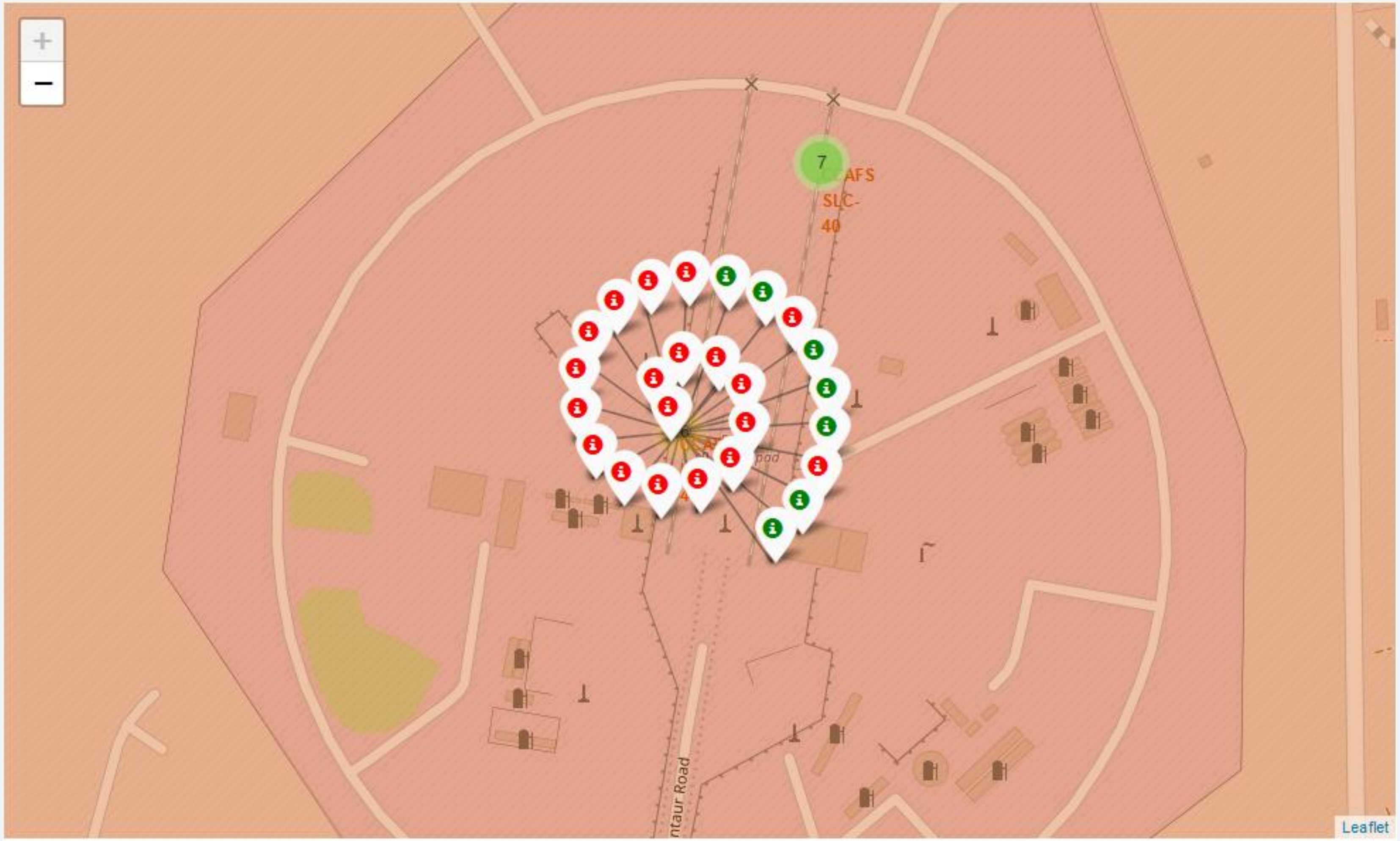
Launch Sites Proximities Analysis

SpaceX Launch Sites on United States



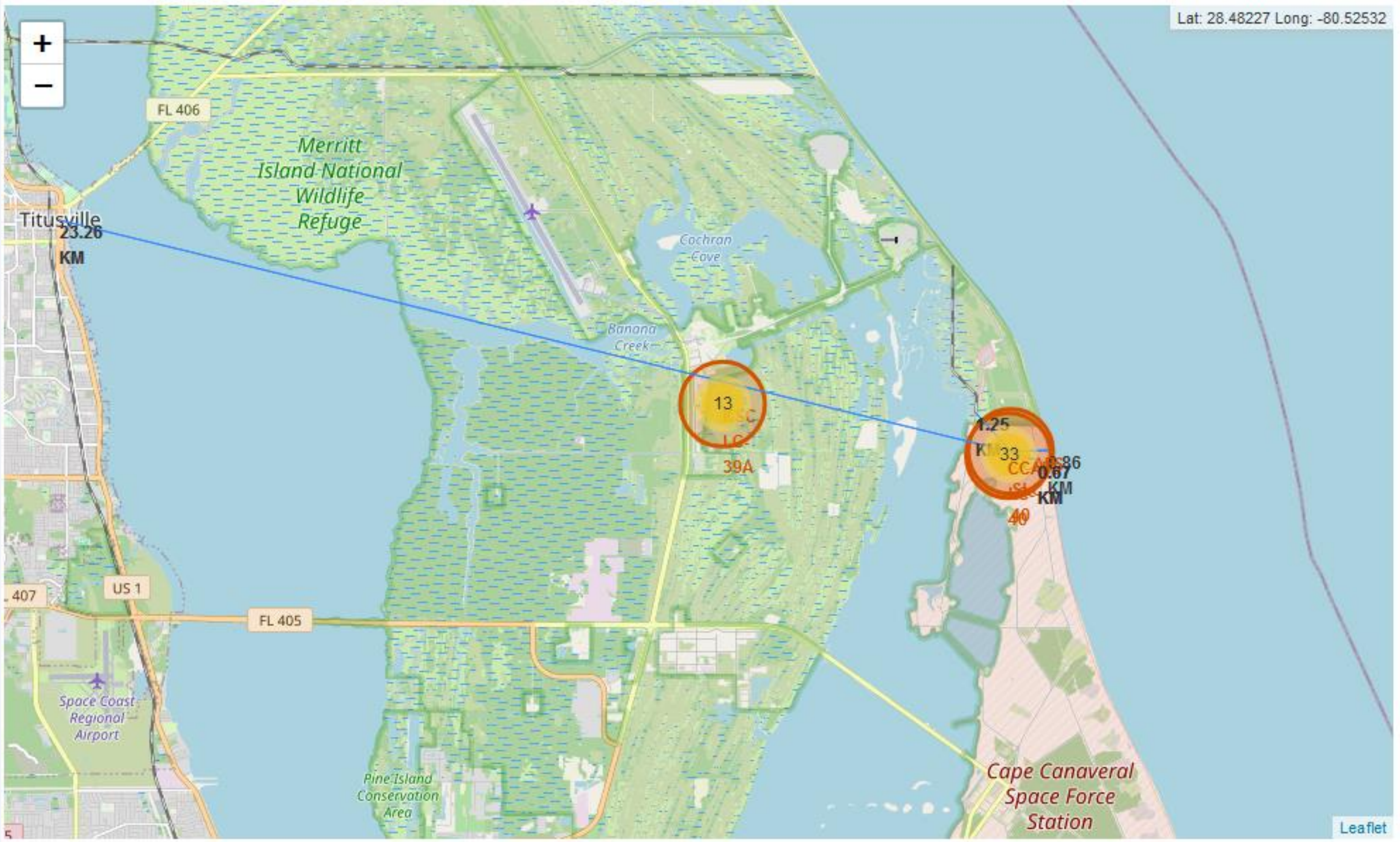
- The Launch sites are found on the coasts of United States
- We have 1 in the West coast of US near Santa Maria
- We have 3 in the East Coast near Meritt Island

SpaceX Launch Sites Success Rates

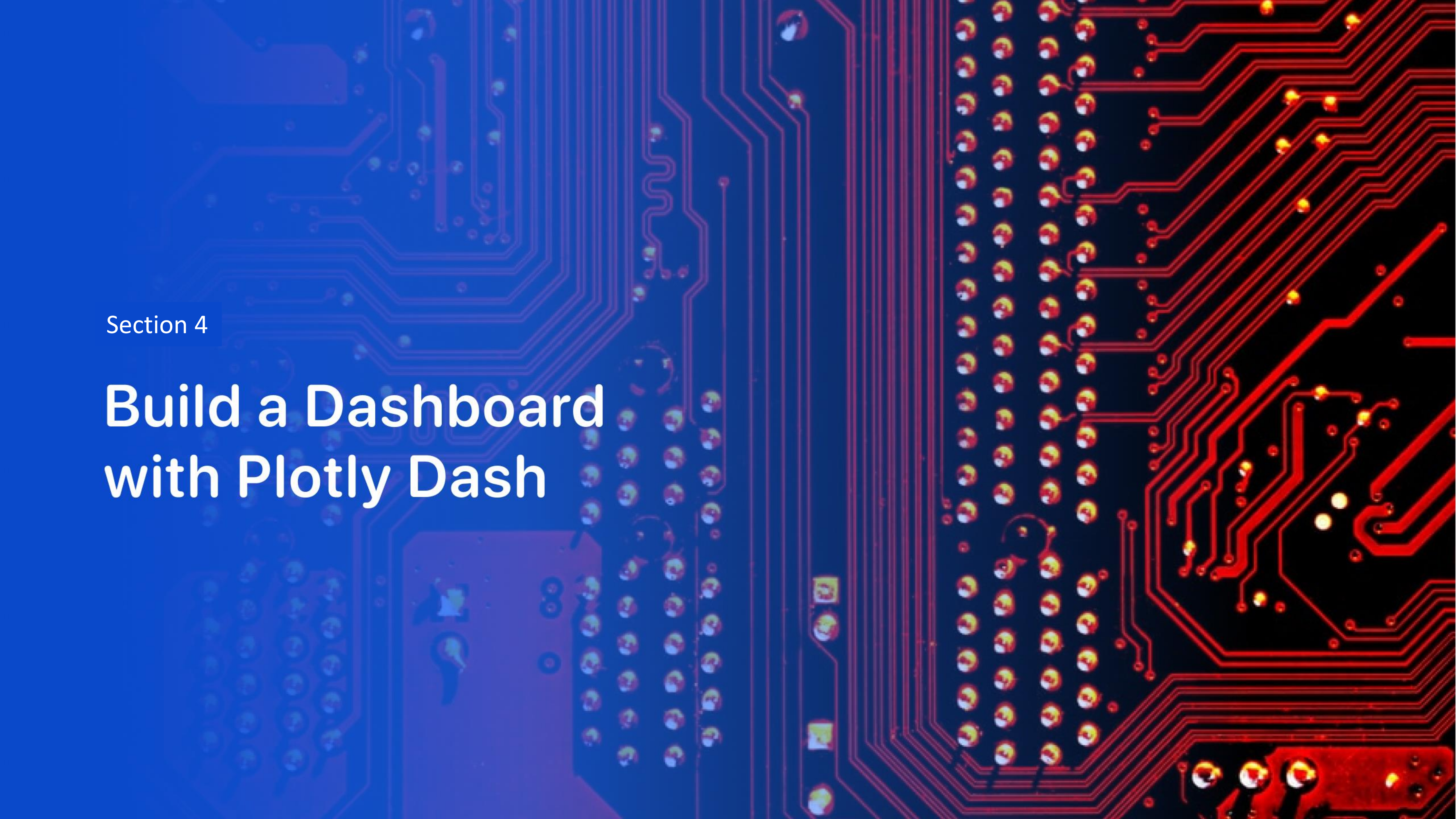


- We can see that at the CCAFS SLC-40 Launch Site, there is a greater number of failed launches compared to successful launches.
- Namely:
  - 19 Fail
  - 7 Success

Launch Site Distances to Geographical Landmarks


Lat: 28.48227 Long: -80.52532
+
−
FL 406
Merritt Island National Wildlife Refuge
Titusville
23.26 KM
Cochran Cove
Banana Creek
13
LC-39A
1.25 KM
33
0.86 KM
0.67 KM
40
407
US 1
FL 405
Space Coast Regional Airport
Pine Island Conservation Area
Cape Canaveral Space Force Station
Leaflet

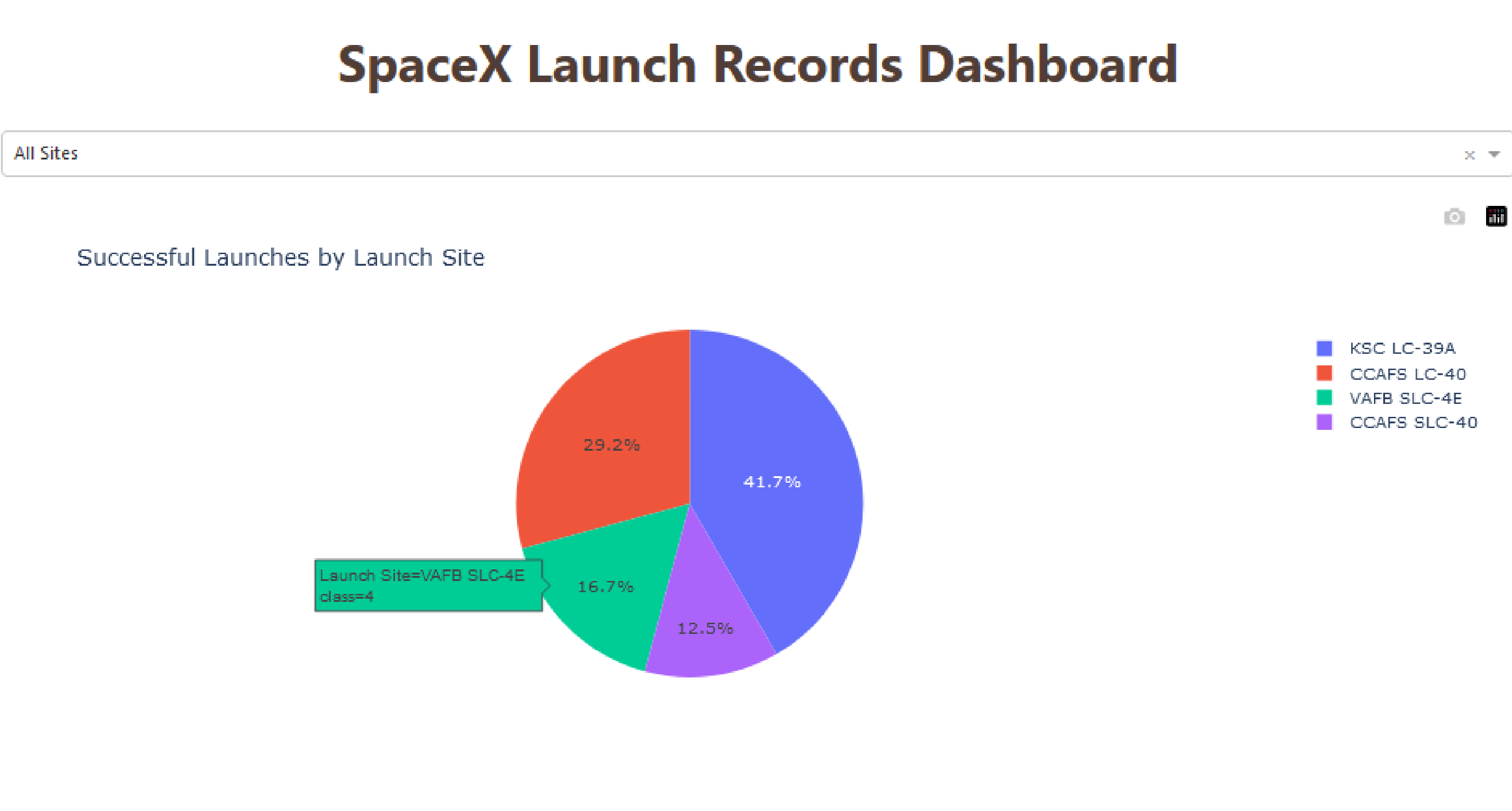
- We can see there are 4 lines respectively for Coastline, Railway, Highway and City
- The distance from the City to the launch site is the furthest out of all of the landmarks



Section 4

Build a Dashboard with Plotly Dash

Successful Launches by Launch Site



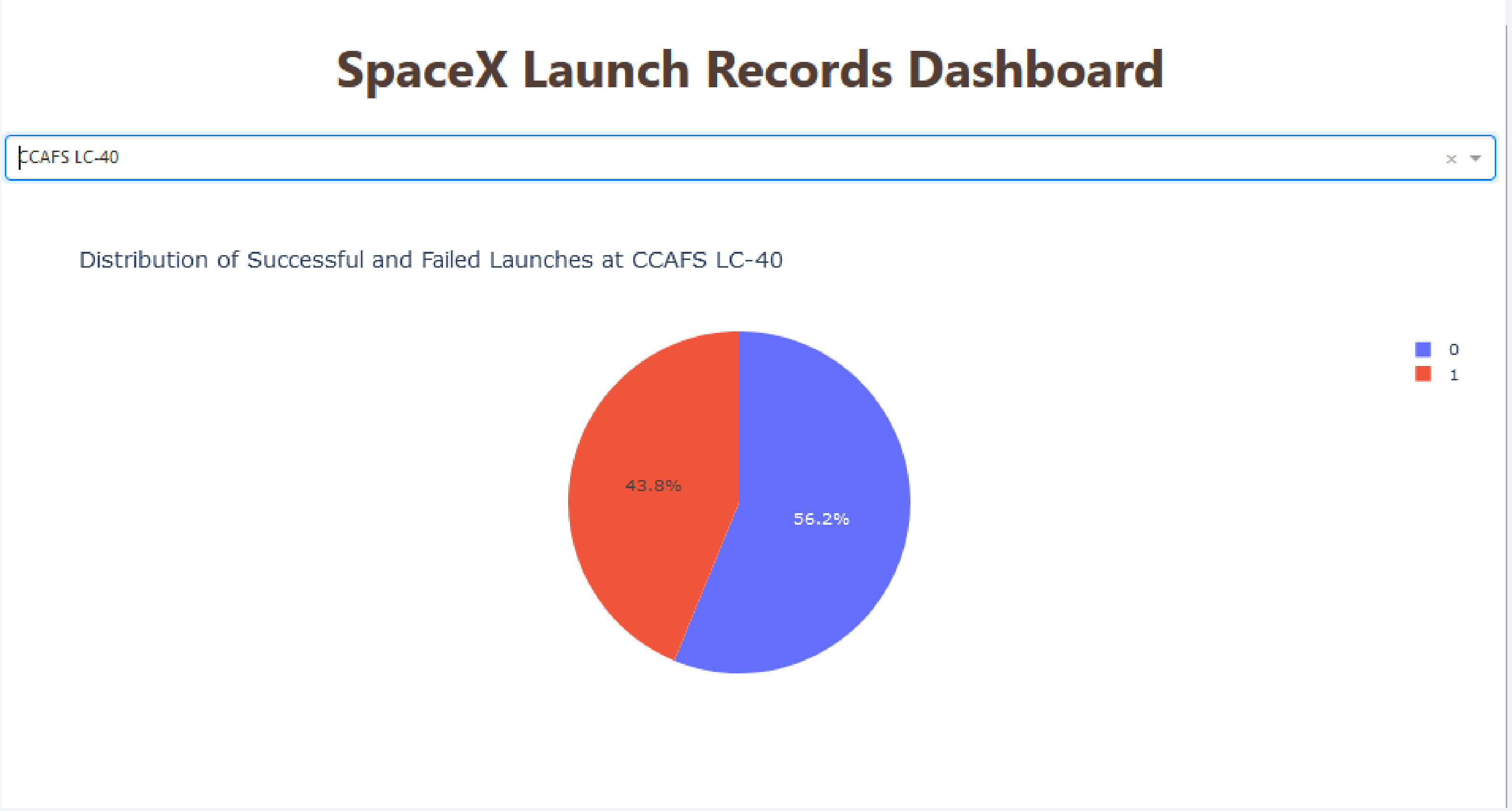
SpaceX Launch Records Dashboard

All Sites

Successful Launches by Launch Site

- The pie chart shows the percentage of each launch site based on their numerical count over total records
- When hovered over, you can see the Launch site, and the number count of success (class = 1)

Highest Launch success rate Launch Site



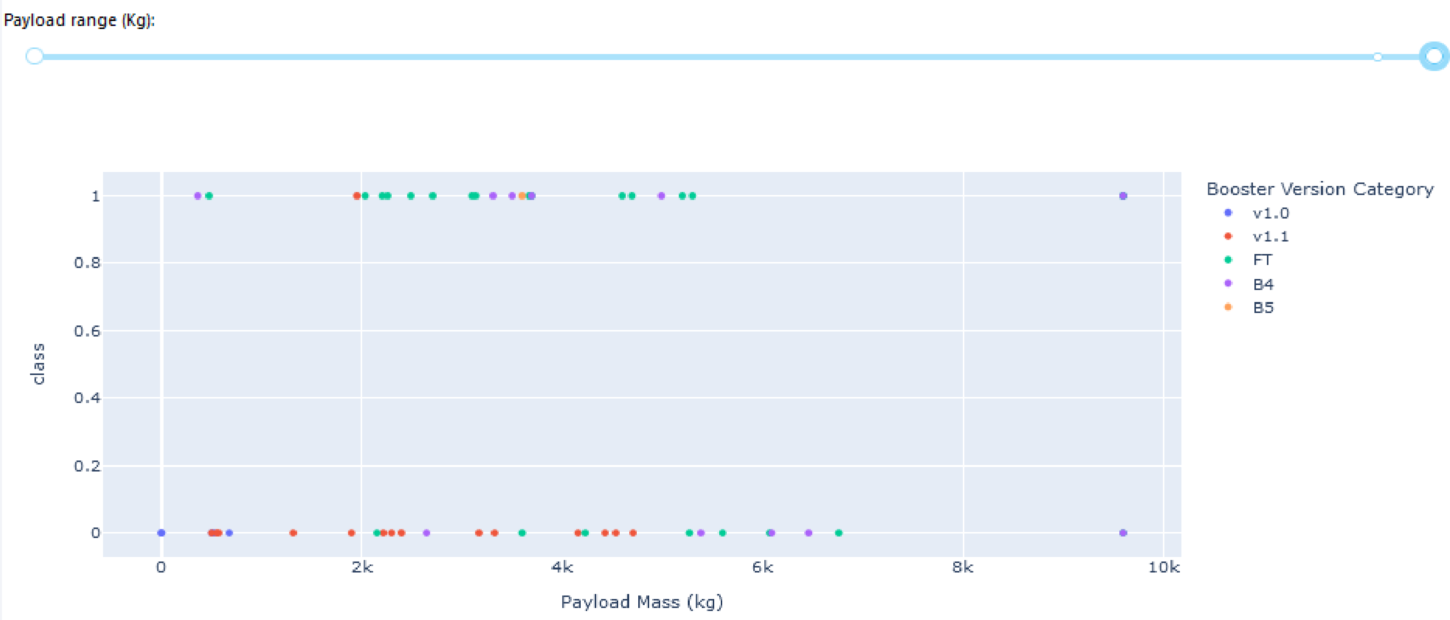
SpaceX Launch Records Dashboard

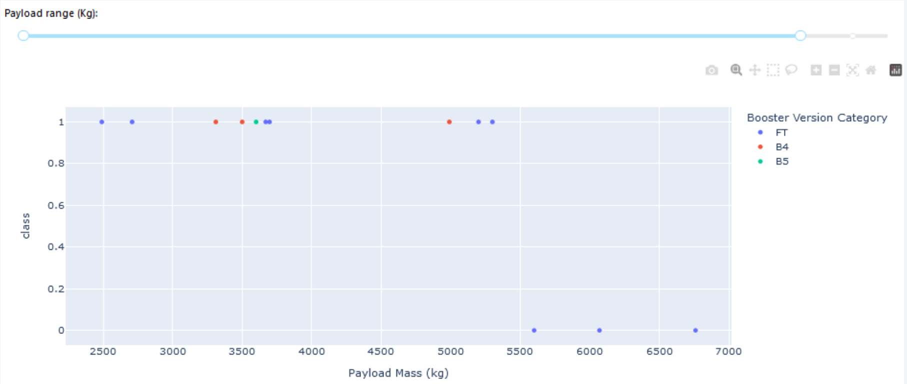
CCAFS LC-40

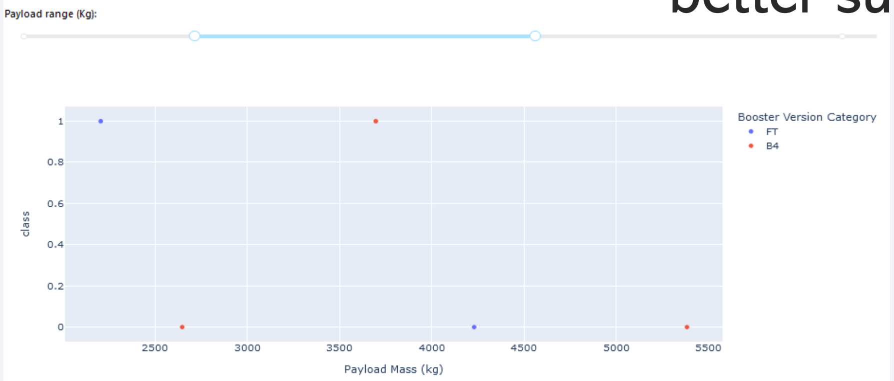
Distribution of Successful and Failed Launches at CCAFS LC-40

- The dropdown allows filtering of any specific Launch Site
- The piechart will then be updated to look at all records (class =0 or 1) filtered by the launch site
- We see here that CCAFS LC-40 has the highest percentage at 43.8%.

Launch Outcome vs Payload Mass







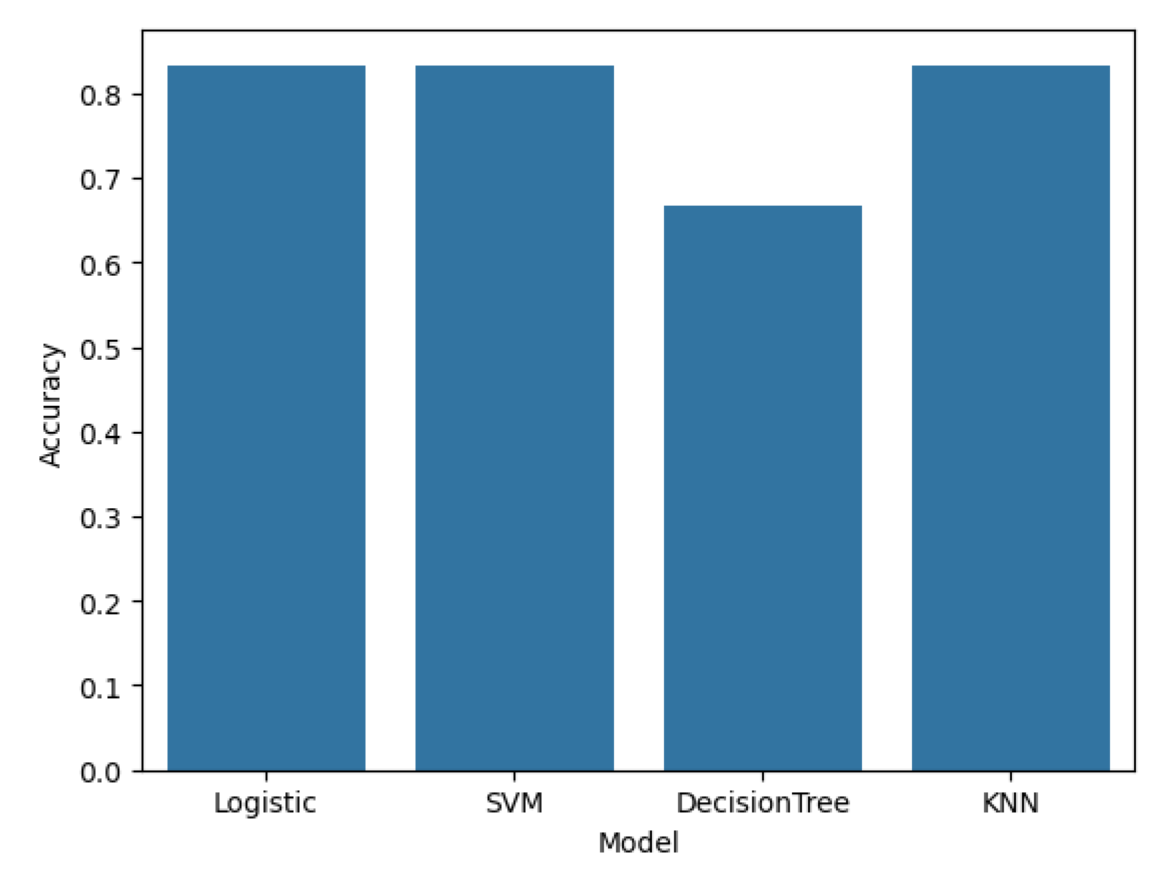
- The Payload range slider allows a closer look by restricting the Payload Mass to the given min and max value
- From the screenshot, we can see that between 2k to 5k payload mass yield better success results

Section 5

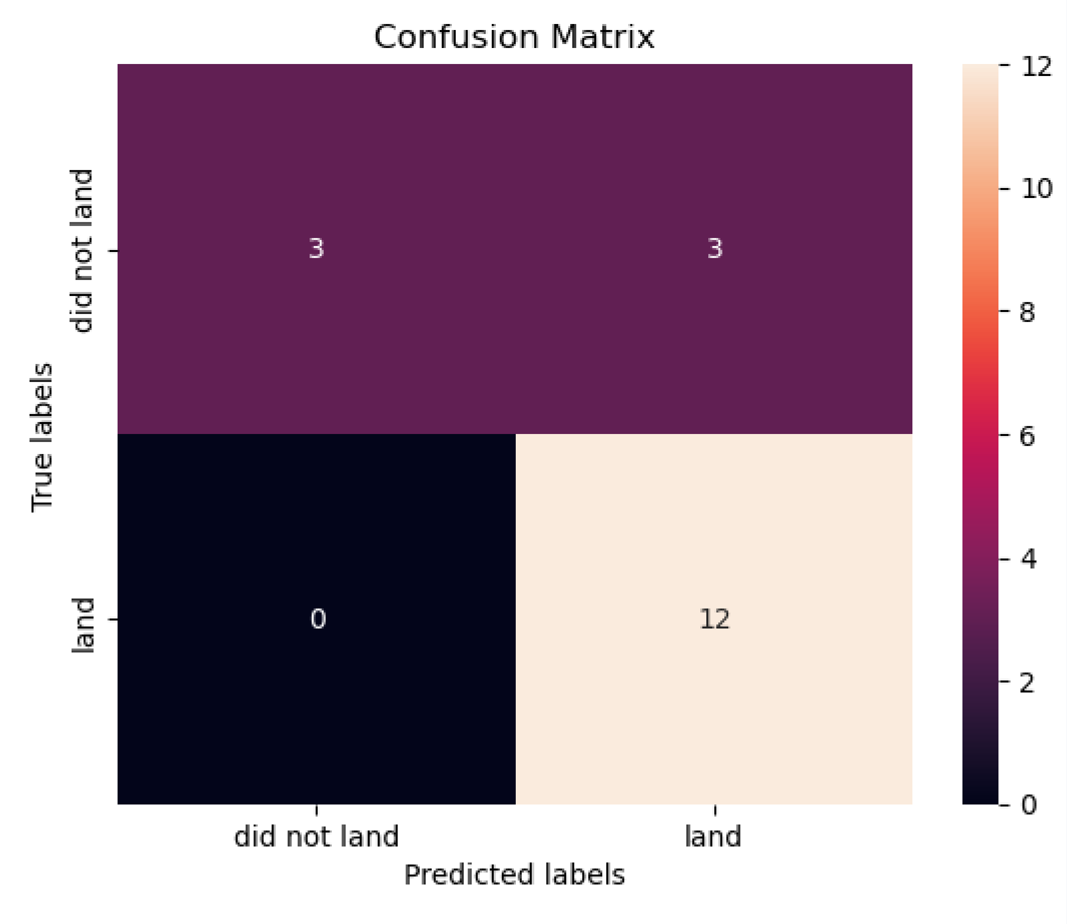
Predictive Analysis (Classification)

Classification Accuracy

- All the models have similar accuracy except for Decision Tree at 0.8333
- This is with GridSearchCV with scoring='accuracy'



Confusion Matrix



- Examining the confusion matrix, we see that the SVM can distinguish between the different classes. We see that it is similar to Logistic Regression with the same number of false positives.

- Overview:

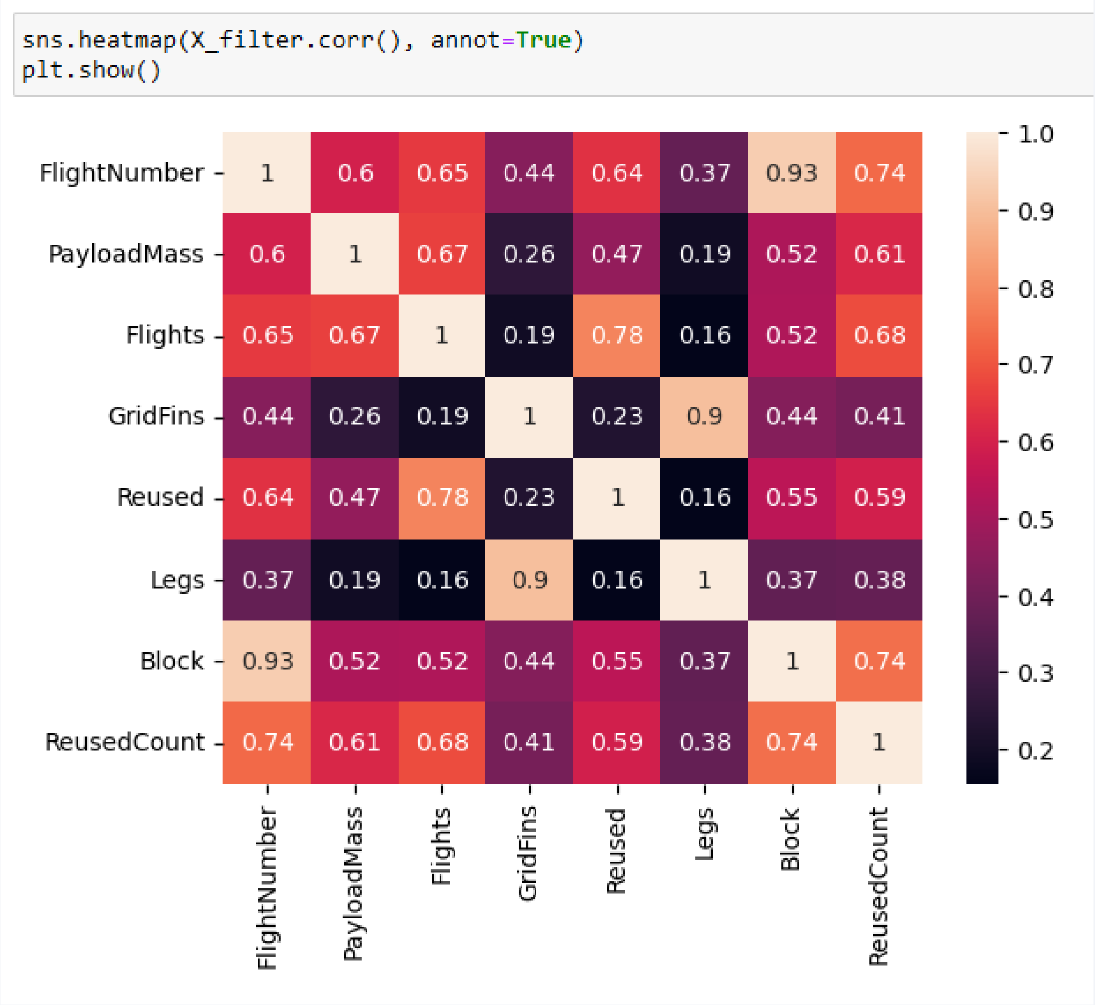
- True Positive - 12 (True label is landed, Predicted labels is also landed)

- False Positives - 3 (True label is not landed, Predicted labels is landed)

Conclusions

- Our selected prediction model is SVM which has a 0.83 accuracy score against our testing data
- We were able to determine the following features are dependant variables through EDA:
  - Launch Site
  - Flight Number
  - Payload Mass
  - Orbit
- We did not explore all the different relationships of the other dependent variables we used such as Reused, Block, GridFins etc

Appendix



```
sns.heatmap(X_filter.corr(), annot=True)
plt.show()
```

	FlightNumber	PayloadMass	Flights	GridFins	Reused	Legs	Block	ReusedCount
FlightNumber	1	0.6	0.65	0.44	0.64	0.37	0.93	0.74
PayloadMass	0.6	1	0.67	0.26	0.47	0.19	0.52	0.61
Flights	0.65	0.67	1	0.19	0.78	0.16	0.52	0.68
GridFins	0.44	0.26	0.19	1	0.23	0.9	0.44	0.41
Reused	0.64	0.47	0.78	0.23	1	0.16	0.55	0.59
Legs	0.37	0.19	0.16	0.9	0.16	1	0.37	0.38
Block	0.93	0.52	0.52	0.44	0.55	0.37	1	0.74
ReusedCount	0.74	0.61	0.68	0.41	0.59	0.38	0.74	1

- Correlation of Boolean and continuous variables
- We can see that there are weak relation between GridFins and Flights and Reused
- We see strong relationship for ReusedCount and FlightNumber and ReusedCount and Flights.

Thank you!