IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Patrick Zhang
30 January 2024

Outline

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

Executive Summary

- Summary of methodologies
  - Data Collection with APIs and Web Scrapping
  - Data Wrangling
  - Exploratory Data Analysis (EDA) with SQL and Data Visualization
  - Interactive Maps using Folium
  - Interactive Dashboard using Plotly Dash
  - Predictive Analysis using Machine Learning
- Summary of all results
  - EDA results
  - Interactive maps and dashboard
  - Predictive results

Introduction

- Project background and context
  - SpaceX advertises Falcon 9 rocket launches on its website, with a cost of 62 million dollars; other providers cost upward of 165 million dollars each. Much of the savings is because SpaceX can reuse the first stage. Therefore, if we can determine if the first stage will land, we can determine the cost of a launch. In this project, we will predict if the Falcon 9 first stage will land successfully.
- Problems you want to find answers
  - What characterizes a successful or failed landing?
  - Are there any relationships between rocket variables that determine the outcome of a landing?
  - What must be done to achieve the best landing success rate?

Section 1

Methodology

Methodology

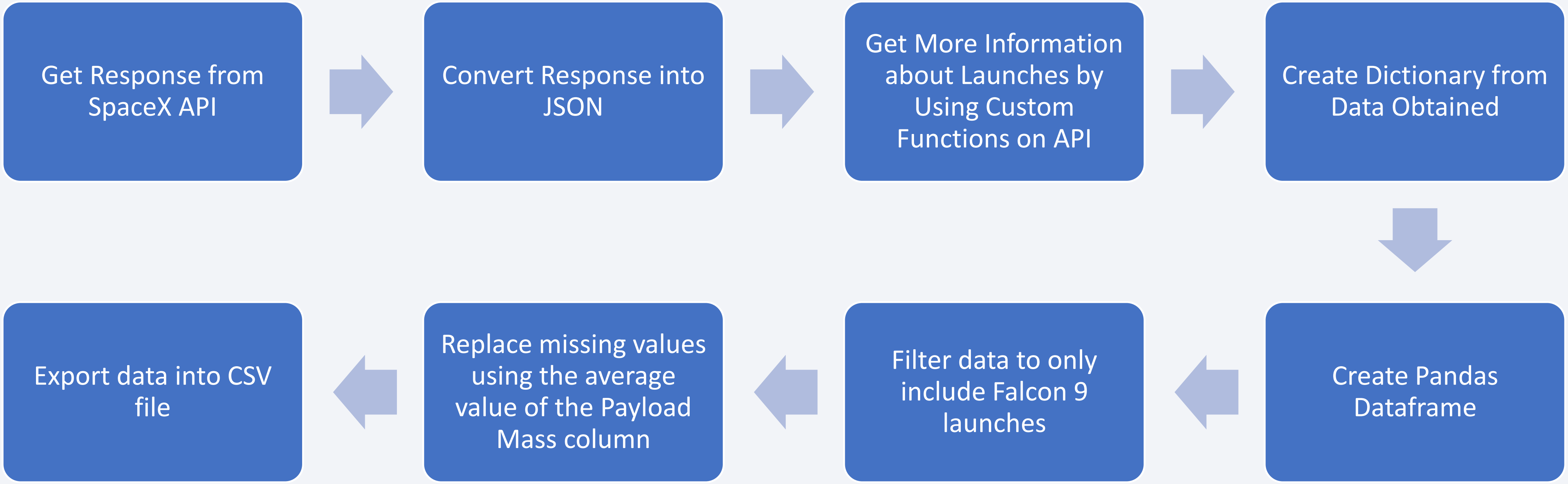
Executive Summary

- Data collection methodology:
  - Data was collected using the SpaceX API and web scrapping from Wikipedia.
- Perform data wrangling
  - Irrelevant columns were dropped
  - One hot encoding applied to categorical variables.
- 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

- Data was collected using the SpaceX API and web scrapping from Wikipedia.
- The first step was collecting data from the SpaceX API.
  - Generate a response using the GET request
  - Decode the response as a JSON and convert into a Pandas dataframe.
  - The data was cleaned by checking for missing values and filling in the missing values where necessary.
- After collecting the data from the API, web scrapping was performed to collect Falcon 9 historical launch data from Wikipedia.
  -

Data Collection – SpaceX API



Get Response from SpaceX API

Convert Response into JSON

Get More Information about Launches by Using Custom Functions on API

Create Dictionary from Data Obtained

Create Pandas Dataframe

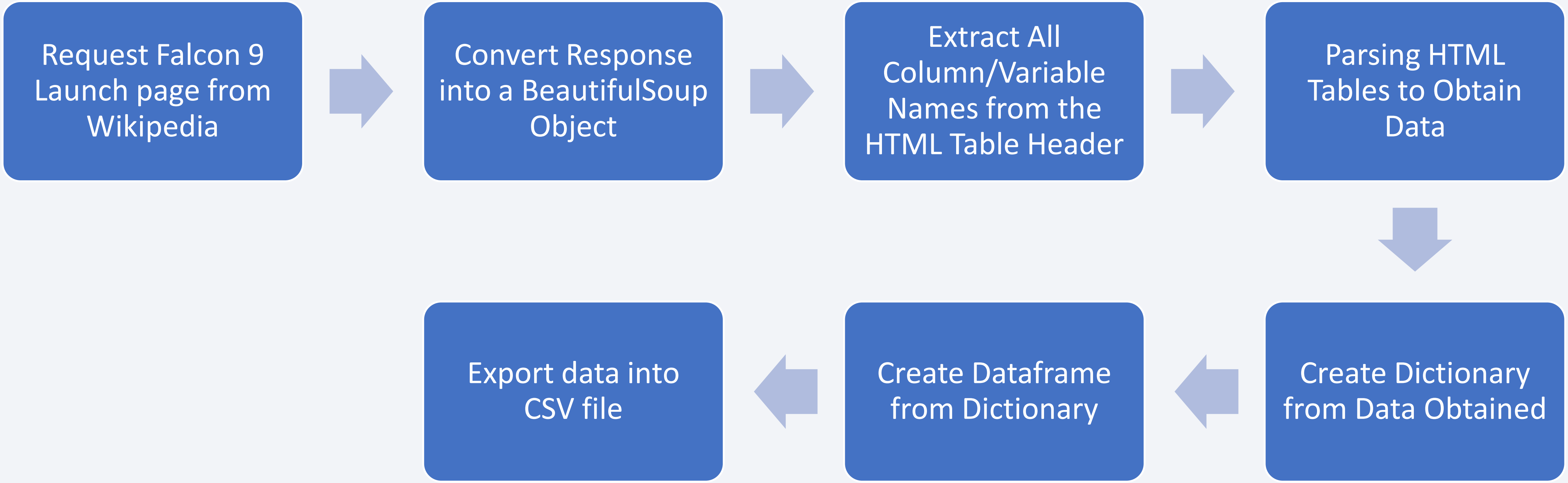
Filter data to only include Falcon 9 launches

Replace missing values using the average value of the Payload Mass column

Export data into CSV file

GitHub link to notebook

Data Collection - Scraping



Request Falcon 9 Launch page from Wikipedia

Convert Response into a BeautifulSoup Object

Extract All Column/Variable Names from the HTML Table Header

Parsing HTML Tables to Obtain Data

Create Dictionary from Data Obtained

Create Dataframe from Dictionary

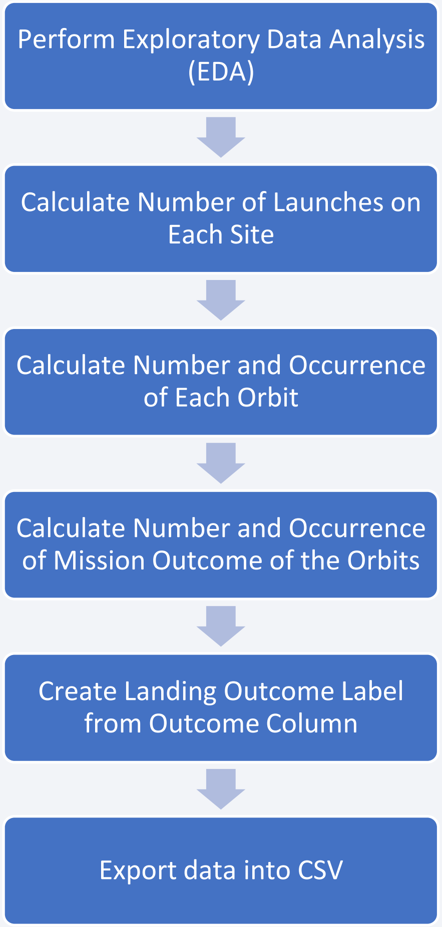
Export data into CSV file

GitHub link to notebook

Data Wrangling

- In the dataset, there are some cases where the booster landing was unsuccessful.
  - True Ocean, True RTLS, and True ASDS indicate a successful mission
  - False Ocean, False RTLS, and False ASDS indicate a failed mission
- Strings were transformed into categorical variables.
  - 1 indicates a successful mission
  - 0 indicates a failed mission

[GitHub link to notebook]



Perform Exploratory Data Analysis (EDA)

Calculate Number of Launches on Each Site

Calculate Number and Occurrence of Each Orbit

Calculate Number and Occurrence of Mission Outcome of the Orbits

Create Landing Outcome Label from Outcome Column

Export data into CSV

EDA with Data Visualization

- The following charts were plotted
  - Flight Number vs. Payload (Scatterplot)
  - Flight Number vs. Launch Site (Scatterplot)
  - Payload vs. Launch Site (Scatterplot)
  - Success Rate vs. Orbit (Bar Graph)
  - Flight Number vs. Orbit Type (Scatterplot)
  - Payload vs. Orbit Type (Scatterplot)
  - Success Yearly Trend (Line Graph)

GitHub link to notebook

- Scatterplots show relationships between variables, known as correlation.
- Bar graphs show relationships between numerical and categorical variables
- Line graphs show trends over time

EDA with SQL

- The following queries were performed:
  - Displaying the names of the unique launch sites in the space mission
  - Displaying 5 records where launch sites begin with the string 'CCA’
  - Displaying the total payload mass carried by boosters launched by NASA (CRS)
  - Displaying average payload mass carried by booster version F9 v1.1
  - Listing the date when the first successful landing outcome in ground pad was achieved.
  - Listing the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000
  - Listing the total number of successful and failure mission outcomes
  - Listing the names of the booster_versions which have carried the maximum payload mass
  - Listing the records which will display the month names, failure landing_outcomes in drone ship ,booster versions, launch_site for the months in year 2015.
  - Ranking 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.

GitHub link to notebook

Build an Interactive Map with Folium

- All launch sites were marked
- Objects such as circles and lines were added to the map to indicate the success or failure of the launches for each launch site
  - Red colored markers indicated failed launches while green colored markers indicated successful launches
  - Lines indicated distances from one launch site to its proximities like a highway, railway, and coastline.

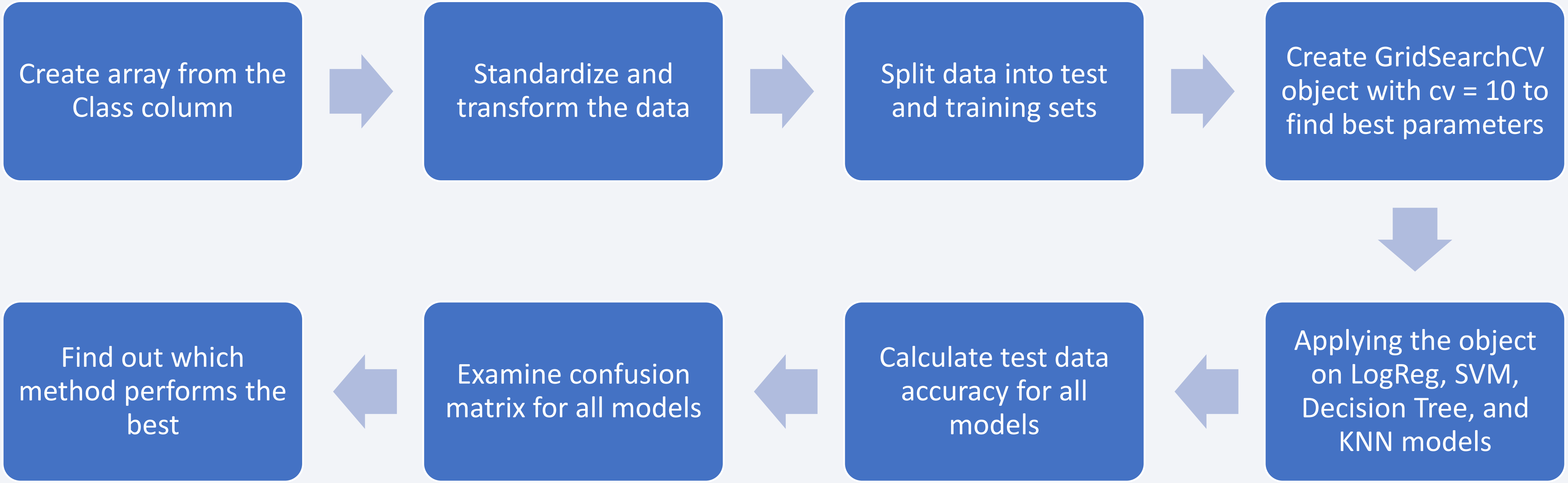
GitHub link to notebook

Build a Dashboard with Plotly Dash

- The interactive dashboard contained a dropdown menu, pie chart, slider and scatterplot.
- The dropdown menu allows users to choose all launch sites or a particular one
- A pie chart shows the total number of the successful launches for all sites.
  - If a particular site is chosen, it shows the total number successful launches and the total number of failed launches
- A slider allows users to choose the payload mass in a range.
- A scatterplot shows the correlation between payload mass and successful launches.

GitHub link to Code

Predictive Analysis (Classification)



Create array from the Class column

Standardize and transform the data

Split data into test and training sets

Create GridSearchCV object with cv = 10 to find best parameters

Applying the object on LogReg, SVM, Decision Tree, and KNN models

Calculate test data accuracy for all models

Examine confusion matrix for all models

Find out which method performs the best

[GitHub link to notebook]

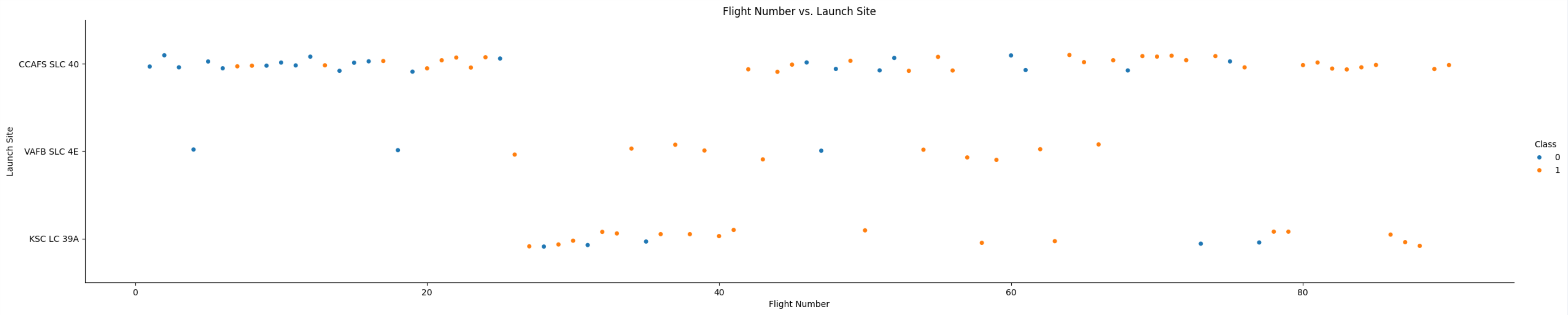
Results

- Exploratory data analysis results
- Interactive analytics demo in screenshots
- Predictive analysis results

Section 2

Insights drawn from EDA

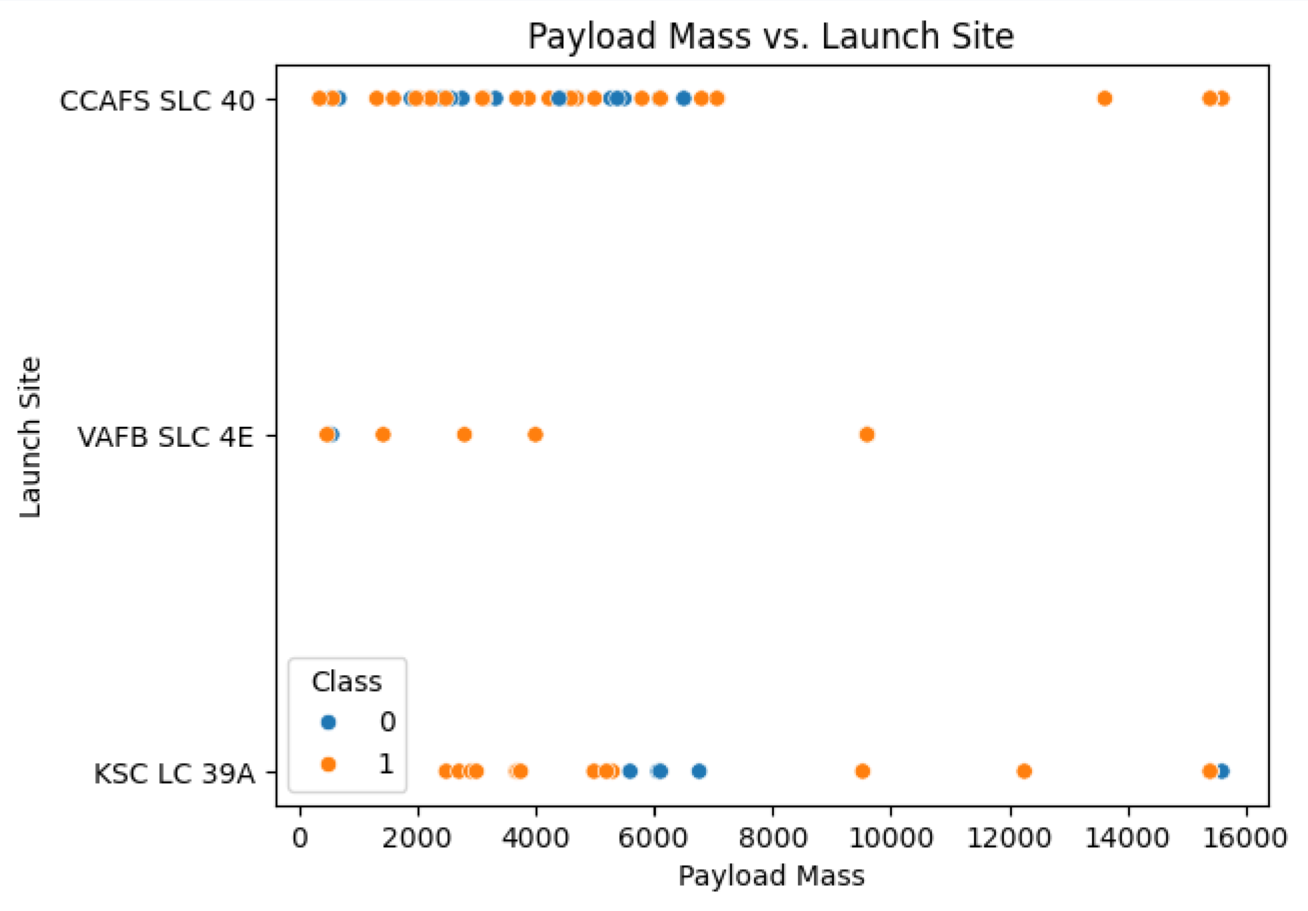
Flight Number vs. Launch Site



Based on the graph, we can observe the following:

- Earlier flights mostly failed while later flights mostly succeeded
- VARB SLC 4E and KSC LC 39A have higher success rates

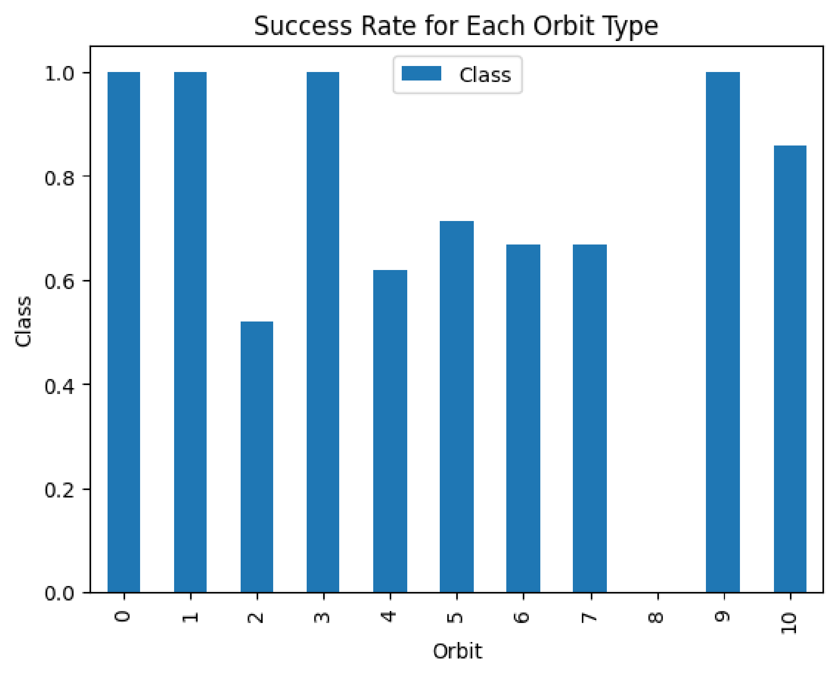
Payload vs. Launch Site

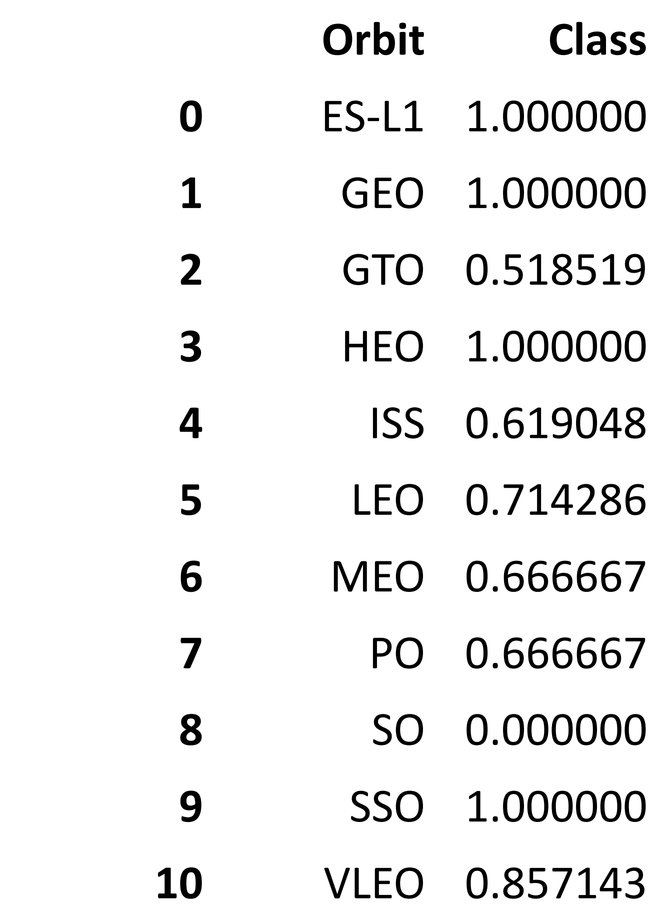


Based on the graph, we can observe the following:

- Most launches with a payload mass of 7000 kg tend to succeed.
- KSC LC 39A has a 100% success rate for payload masses under 5000 kg

Success Rate vs. Orbit Type

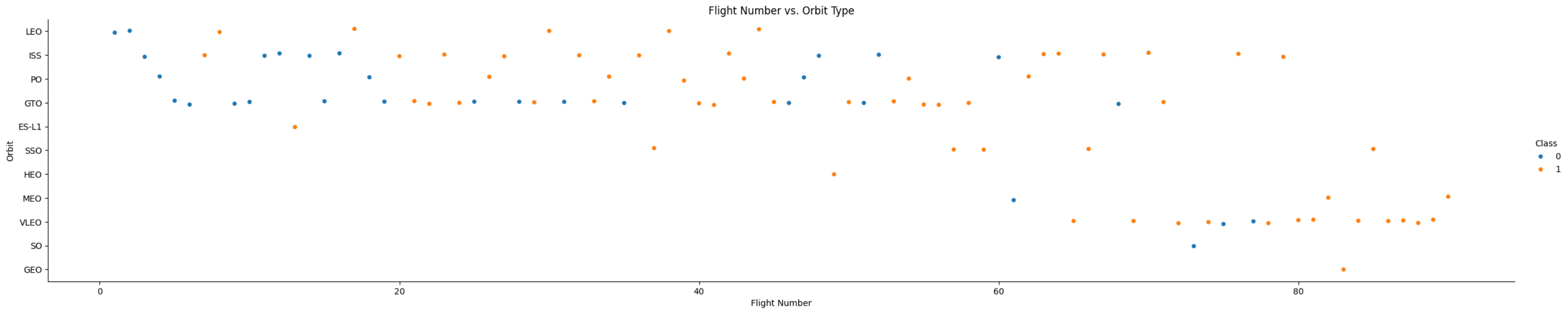

Success Rate for Each Orbit Type



	Orbit	Class
0	ES-L1	1.000000
1	GEO	1.000000
2	GTO	0.518519
3	HEO	1.000000
4	ISS	0.619048
5	LEO	0.714286
6	MEO	0.666667
7	PO	0.666667
8	SO	0.000000
9	SSO	1.000000
10	VLEO	0.857143

From the graph, ES-L1, GEO, HEO, and SSO have the best success rate.

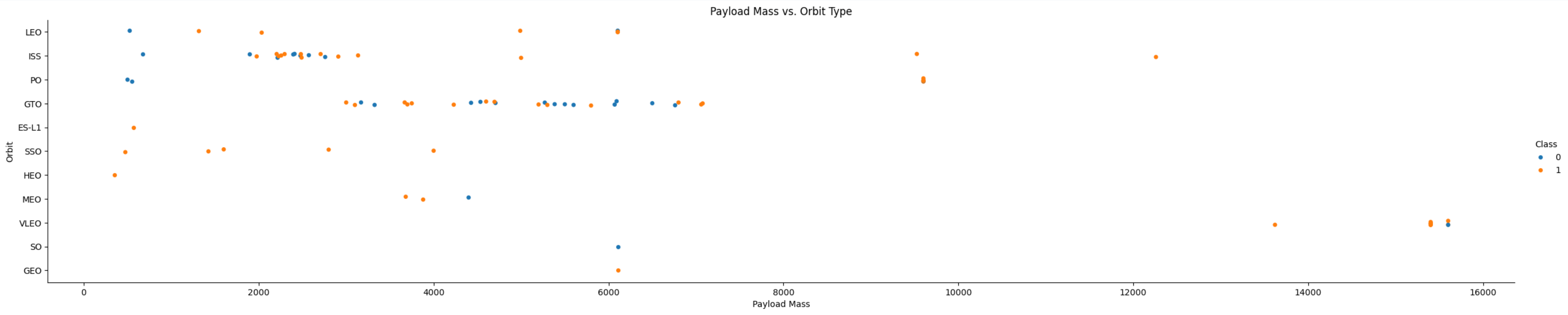
Flight Number vs. Orbit Type



Based on the graph, we can observe the following:

- There’s a relationship between the LEO orbit and the number of flights
- There’s no relationship between the GTO orbit and the number of flights

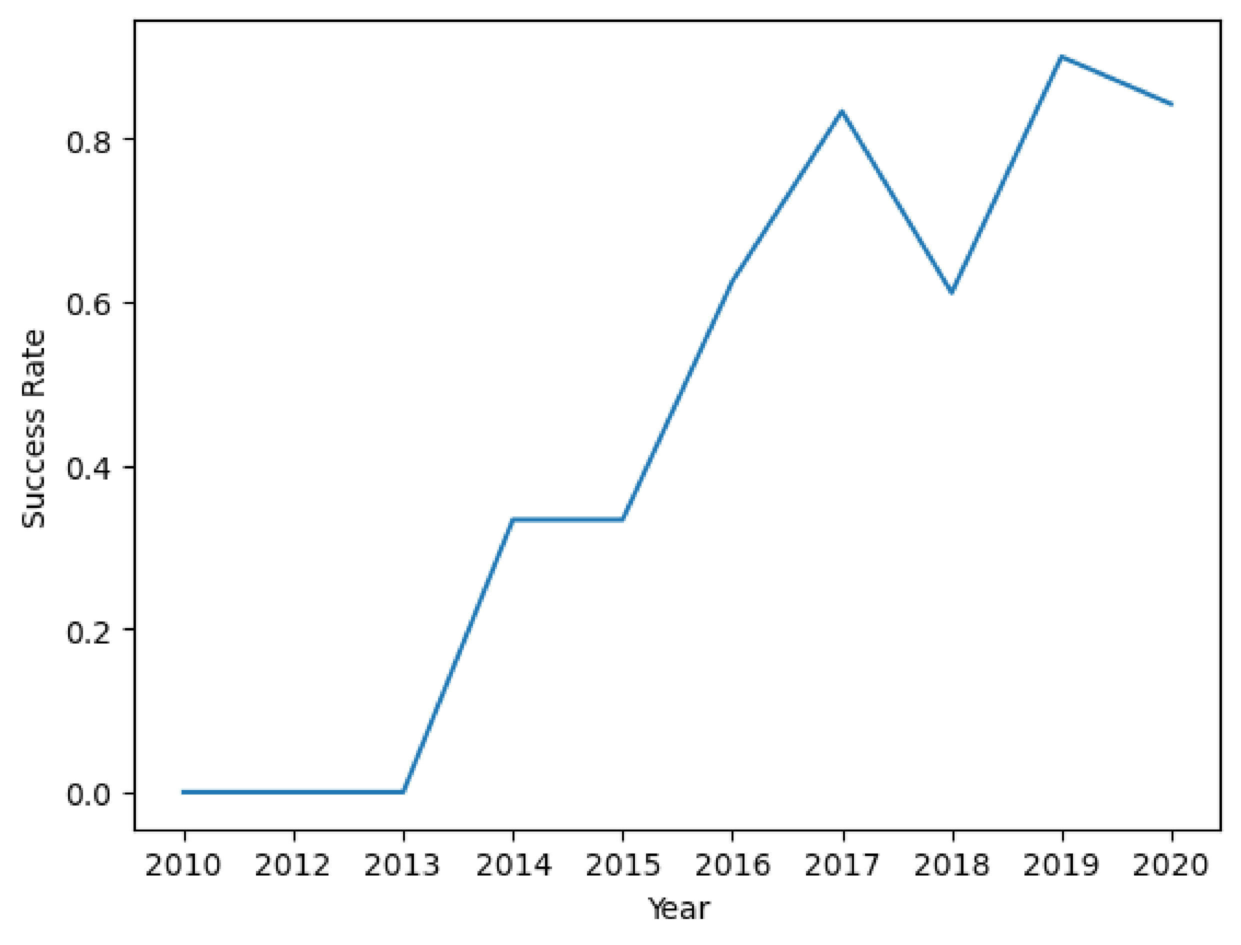
Payload vs. Orbit Type



Based on the graph, we can observe the following:

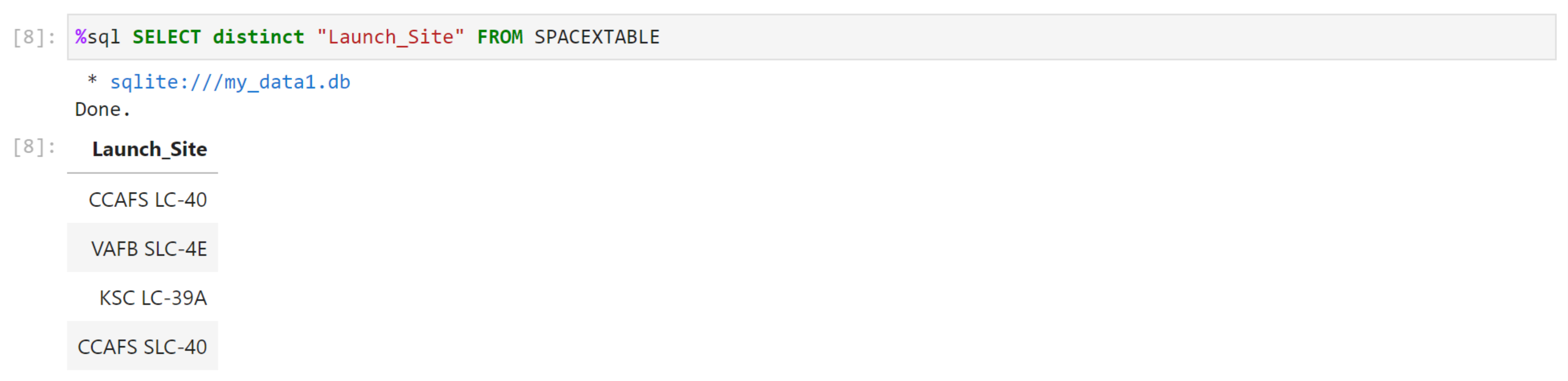
- Heavy payloads have a negative influence on the GTO orbit
- Have payloads have a positive influence on the PO, LEO, and ISS orbits

Launch Success Yearly Trend



Since 2013, the success rate has increased overall.

All Launch Site Names



```
[8]: %sql SELECT distinct "Launch_Site" FROM SPACEXTABLE
```

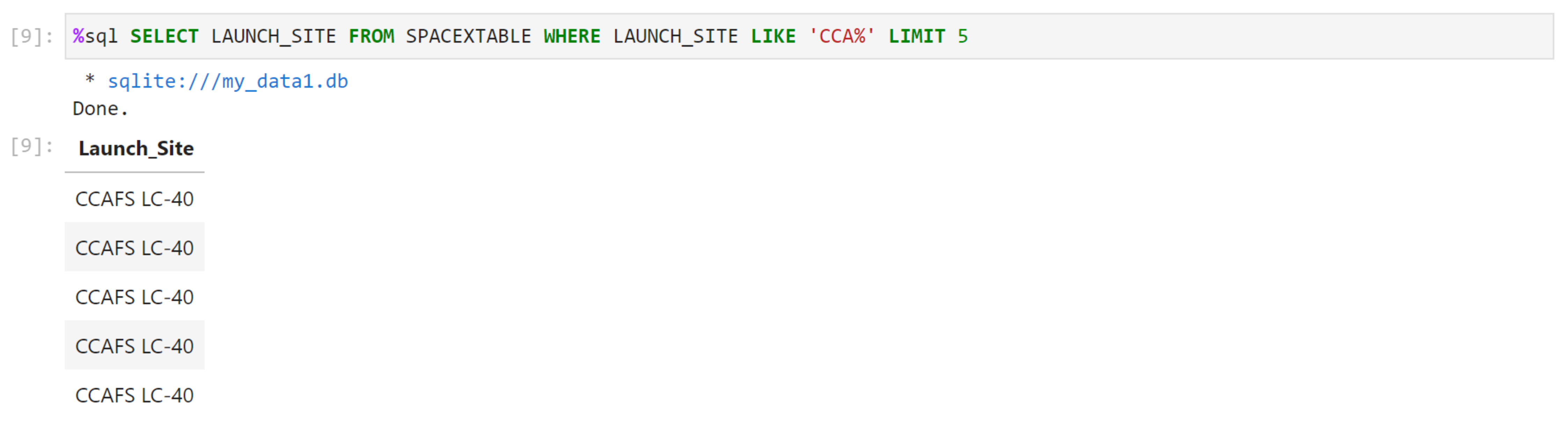
```
 * sqlite:///my_data1.db
Done.
```

[8]:

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

By using the DISTINCT clause, it only shows unique launch sites.

Launch Site Names Begin with 'CCA'



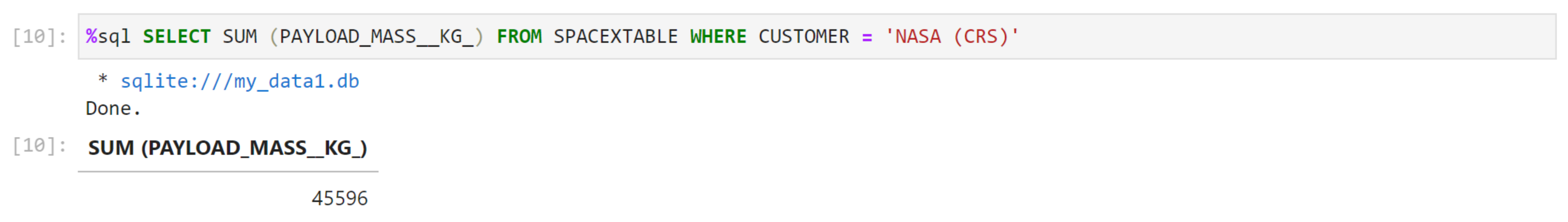
```
[9]: %sql SELECT LAUNCH_SITE FROM SPACEXTABLE WHERE LAUNCH_SITE LIKE 'CCA%' LIMIT 5

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

[9]:

Launch_Site
CCAFS LC-40
CCAFS LC-40
CCAFS LC-40
CCAFS LC-40
CCAFS LC-40

When using the WHERE clause along with the LIKE clause, we filter the launch site names that contain the substring ‘CCA’. By using the LIMIT clause, we only see 5 records.

Total Payload Mass



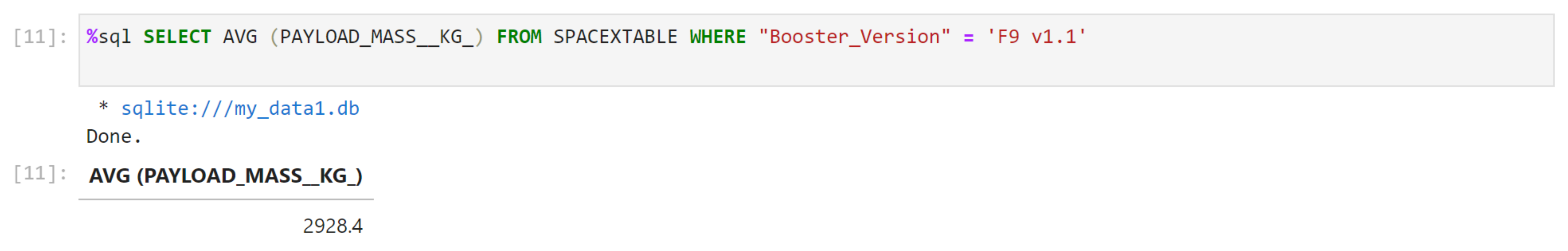
```
[10]: %sql SELECT SUM (PAYLOAD_MASS__KG_) FROM SPACEXTABLE WHERE CUSTOMER = 'NASA (CRS)'
```

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

```
[10]: SUM (PAYLOAD_MASS__KG_)
                        45596
```

This query shows the sum of the payload masses for the customer, NASA (CRS).

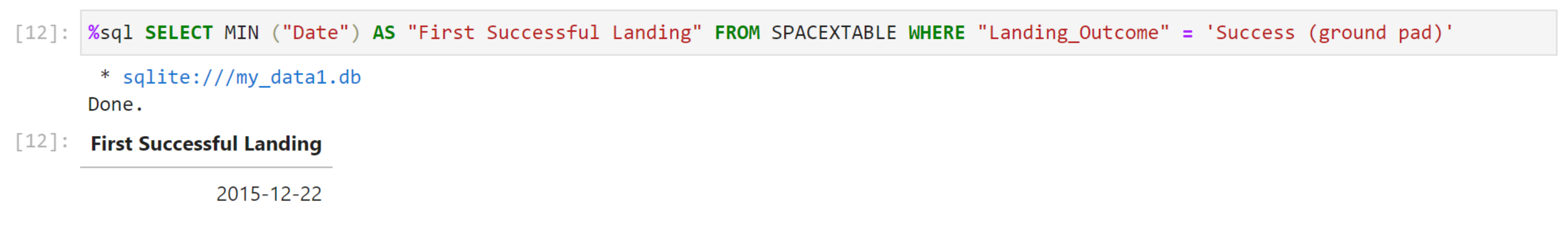
Average Payload Mass by F9 v1.1



```
[11]: %sql SELECT AVG (PAYLOAD_MASS__KG_) FROM SPACEXTABLE WHERE "Booster_Version" = 'F9 v1.1'

       * sqlite:///my_data1.db
      Done.

[11]: AVG (PAYLOAD_MASS__KG_)
                       2928.4
```

This query shows the average payload mass carried by the Booster Version F9 v1.1

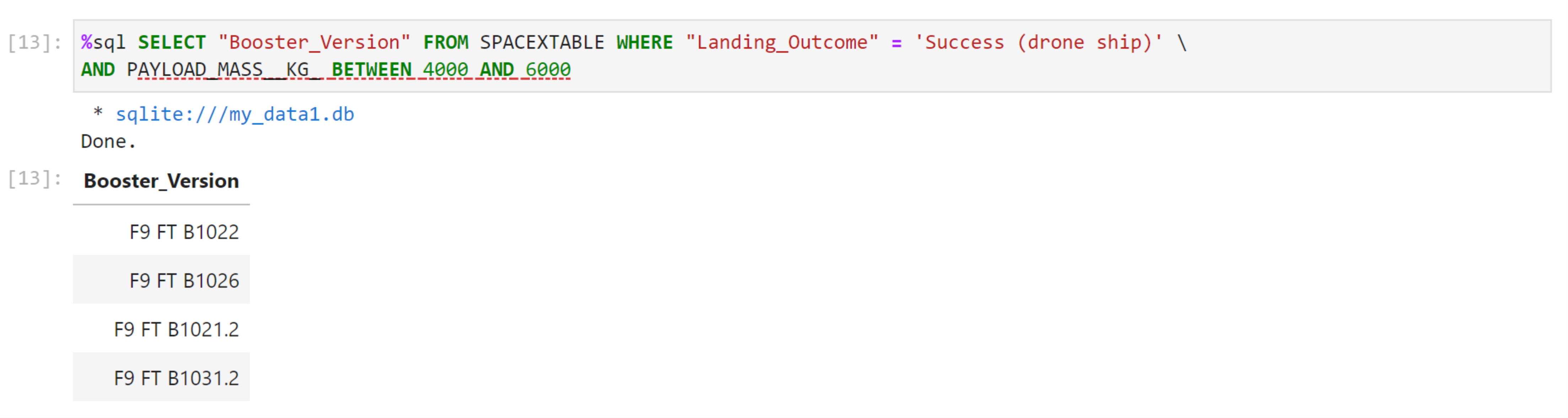
First Successful Ground Landing Date



```
[12]: %sql SELECT MIN ("Date") AS "First Successful Landing" FROM SPACEXTABLE WHERE "Landing_Outcome" = 'Success (ground pad)'

       * sqlite:///my_data1.db
      Done.

[12]: First Successful Landing
                    2015-12-22
```

The records are being filtered using the WHERE clause to show only successful landings in ground pad. Using the MIN clause gets the record with the oldest date.

Successful Drone Ship Landing with Payload between 4000 and 6000



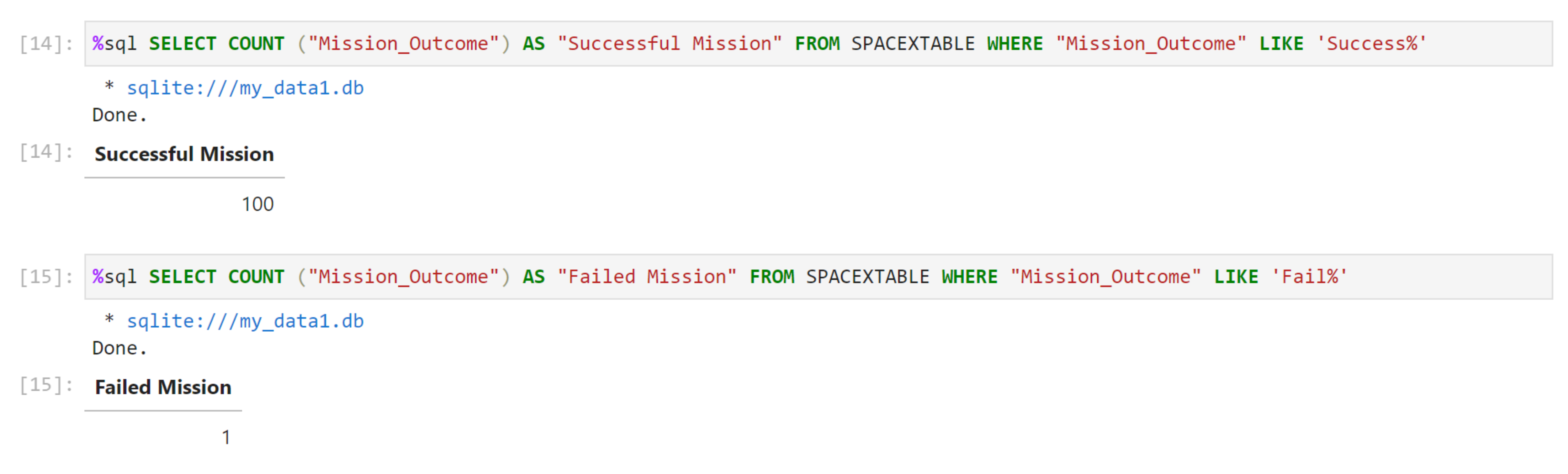
```
[13]: %sql SELECT "Booster_Version" FROM SPACEXTABLE WHERE "Landing_Outcome" = 'Success (drone ship)' \
      AND PAYLOAD_MASS__KG_ BETWEEN 4000 AND 6000

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

[13]:

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

The records are being filtered using the WHERE clause to show only successful landings in drone ship. Additionally, the BETWEEN clause was used to filter landings with payload masses between 4000 and 6000.

Total Number of Successful and Failure Mission Outcomes

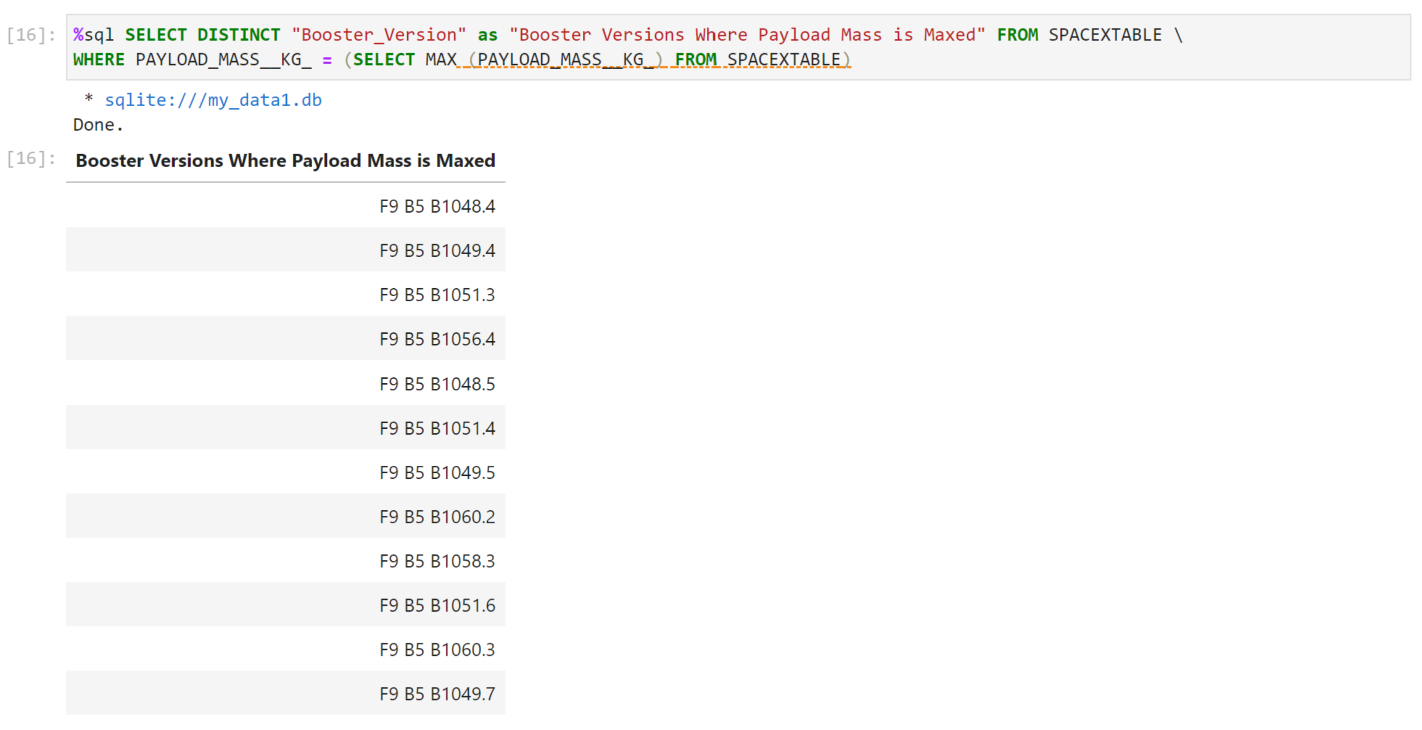


```
[14]: %sql SELECT COUNT ("Mission_Outcome") AS "Successful Mission" FROM SPACEXTABLE WHERE "Mission_Outcome" LIKE 'Success%'

 * sqlite:///my_data1.db
Done.

[14]: Successful Mission
                     100
```

```
[15]: %sql SELECT COUNT ("Mission_Outcome") AS "Failed Mission" FROM SPACEXTABLE WHERE "Mission_Outcome" LIKE 'Fail%'

 * sqlite:///my_data1.db
Done.

[15]: Failed Mission
                  1
```

The COUNT clause was used to get the total number of successful or failed missions. The LIKE clause and the ‘%’ wildcard was used to filter the outcomes where the word ‘success’ or ‘fail’ appeared.

Boosters Carried Maximum Payload



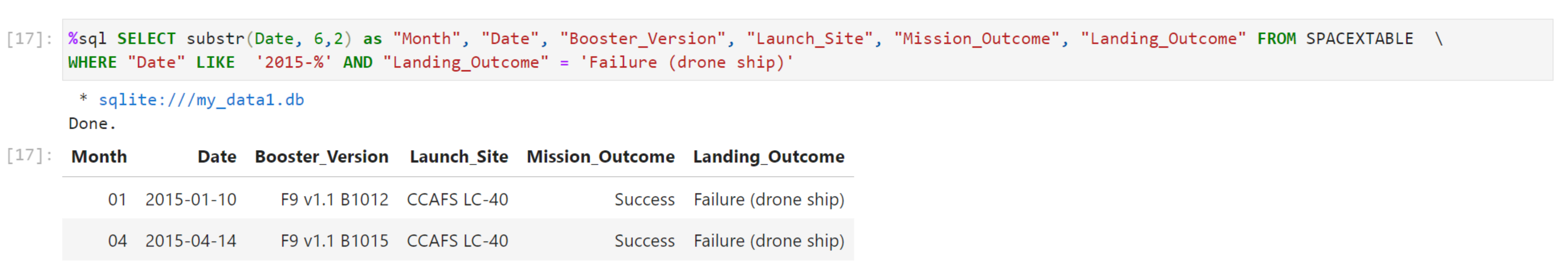
```
[16]: %sql SELECT DISTINCT "Booster_Version" as "Booster Versions Where Payload Mass is Maxed" FROM SPACEXTABLE \
      WHERE PAYLOAD_MASS__KG_ = (SELECT MAX (PAYLOAD_MASS__KG_) FROM SPACEXTABLE)

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

[16]:

Booster Versions Where Payload Mass is Maxed
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

The DISTINCT clause gets all the unique booster version names. The maximum payload mass is acquired using the MAX function and using it with the WHERE clause gets the booster versions that carried the maximum payload.

2015 Launch Records



```
[17]: %sql SELECT substr(Date, 6,2) as "Month", "Date", "Booster_Version", "Launch_Site", "Mission_Outcome", "Landing_Outcome" FROM SPACEXTABLE  \
      WHERE "Date" LIKE  '2015-%' AND "Landing_Outcome" = 'Failure (drone ship)'
```

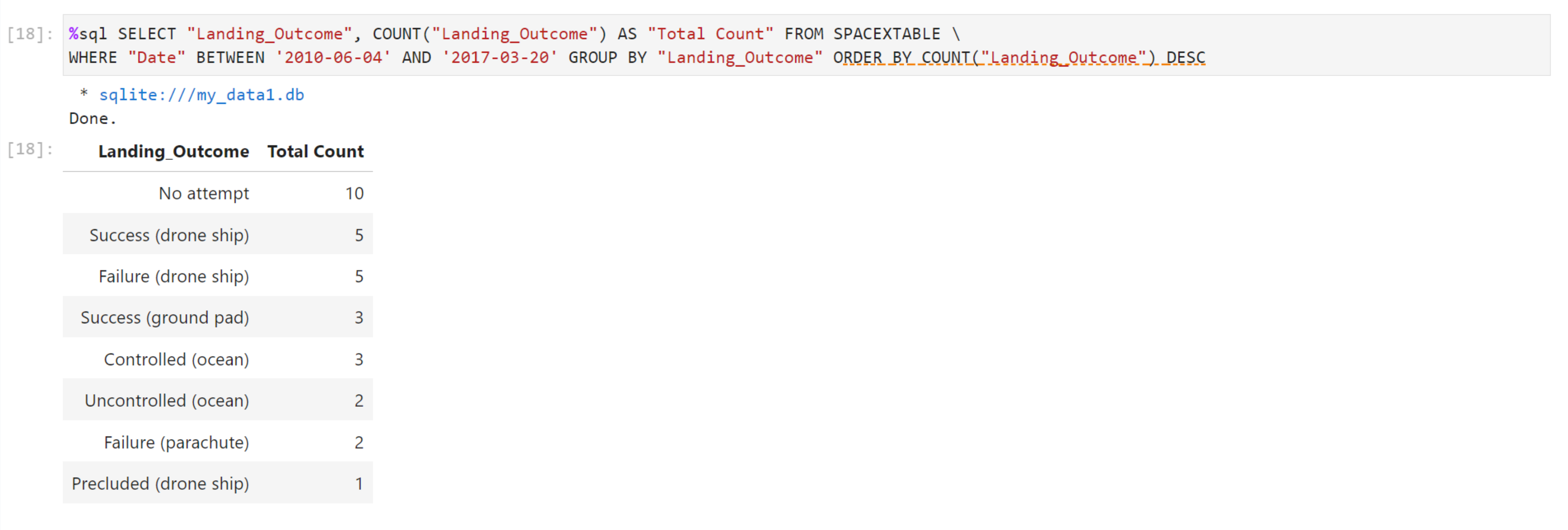
```
 * sqlite:///my_data1.db
Done.
```

[17]:

Month	Date	Booster_Version	Launch_Site	Mission_Outcome	Landing_Outcome
01	2015-01-10	F9 v1.1 B1012	CCAFS LC-40	Success	Failure (drone ship)
04	2015-04-14	F9 v1.1 B1015	CCAFS LC-40	Success	Failure (drone ship)

This query uses a combination of the WHERE, AND, and LIKE clauses to filter records for the year 2015 and get boosters and launch sites with successful missions, but failed landings in drone ship.

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



```
[18]: %sql SELECT "Landing_Outcome", COUNT("Landing_Outcome") AS "Total Count" FROM SPACEXTABLE \
      WHERE "Date" BETWEEN '2010-06-04' AND '2017-03-20' GROUP BY "Landing_Outcome" ORDER BY COUNT("Landing_Outcome") DESC

       * sqlite:///my_data1.db
      Done.
[18]:
```

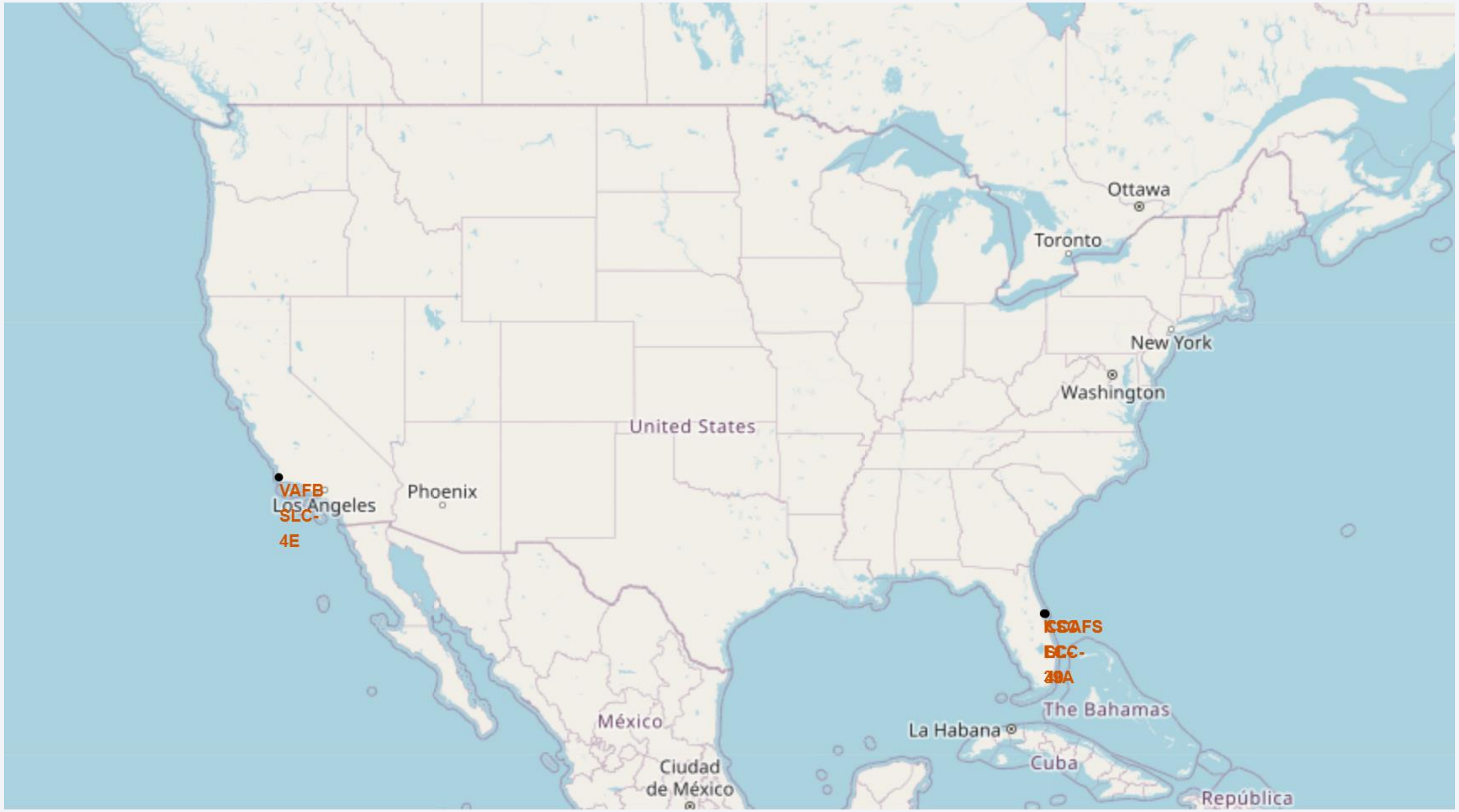
Landing_Outcome	Total Count
No attempt	10
Success (drone ship)	5
Failure (drone ship)	5
Success (ground pad)	3
Controlled (ocean)	3
Uncontrolled (ocean)	2
Failure (parachute)	2
Precluded (drone ship)	1

Using the BETWEEN clause filtered the records for landings between 2010-06-04 and 2017-03-20. All landing outcomes are grouped together by the GROUP BY clause and the ORDER BY COUNT orders the results in descending order by the total count of landings with that particular outcome.

Section 3

Launch Sites Proximities Analysis

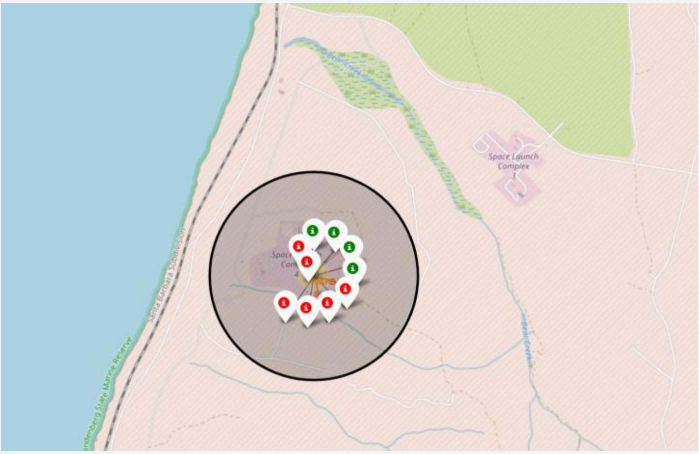
Launch Site Locations



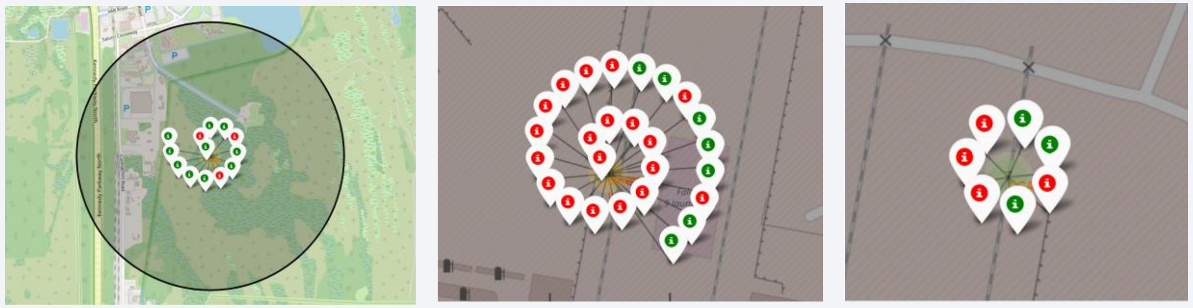
All the SpaceX launch sites are located on the coasts of Florida and California in the USA.

Colored Labeled Launch Records

California Launch Site



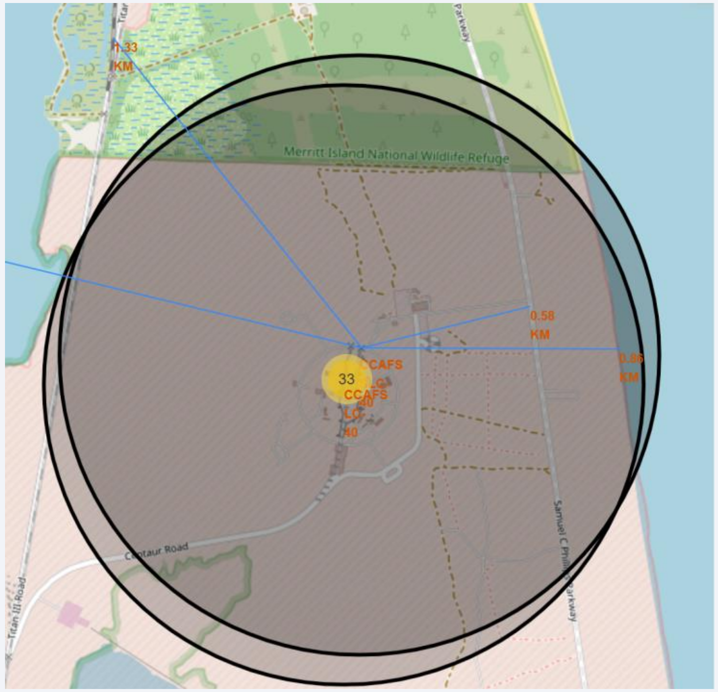
Florida Launch Sites

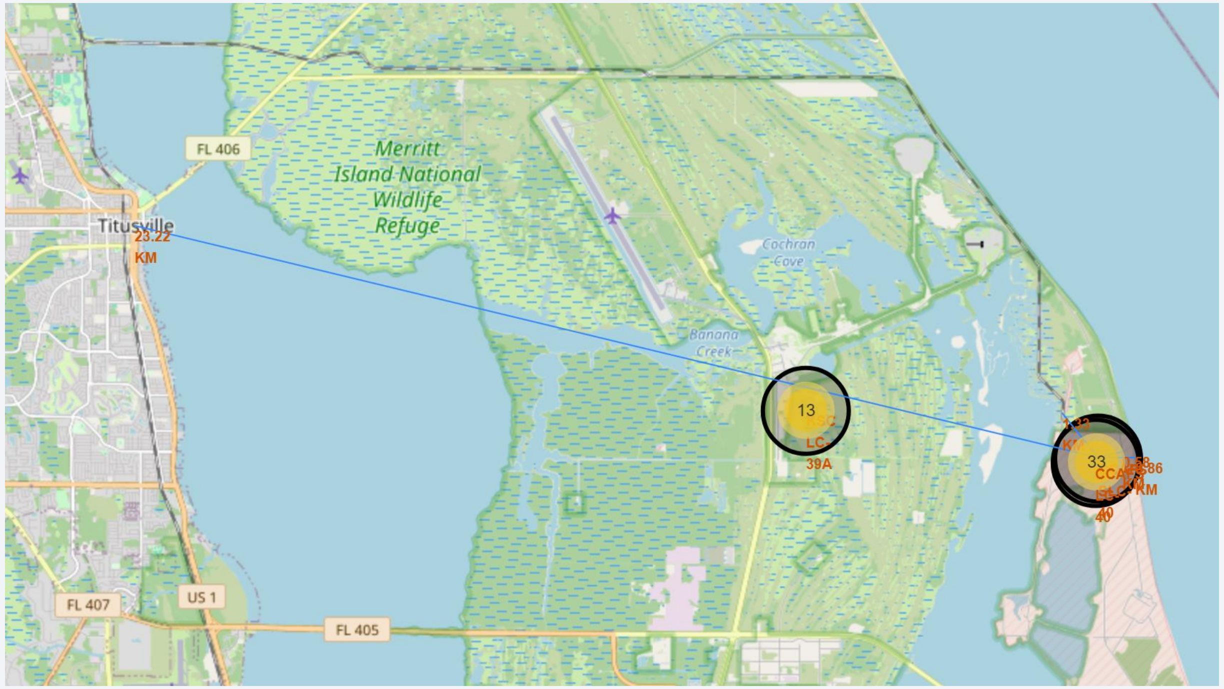


Based on the screenshots, we can observe the following:

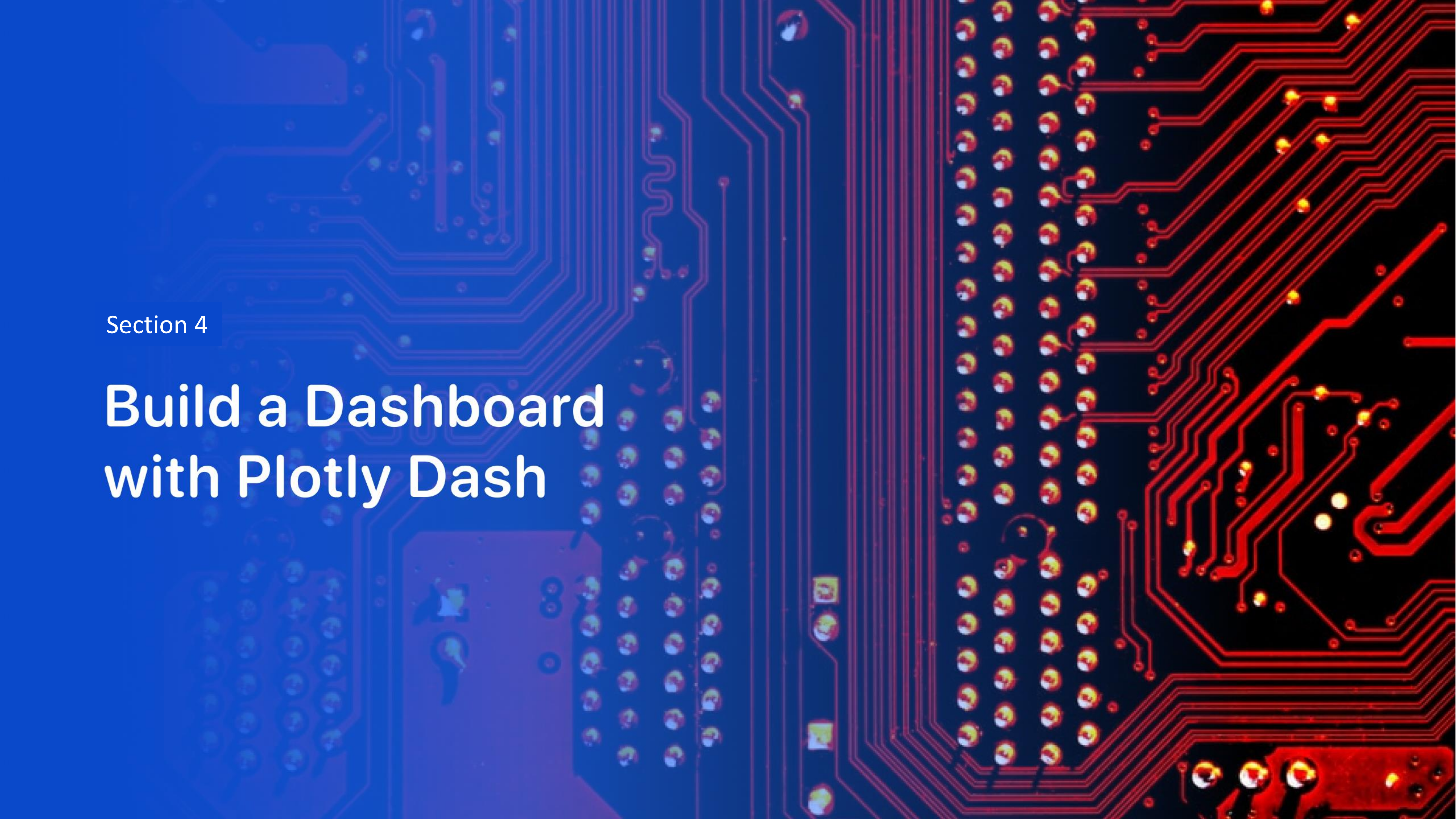
- Red indicates failed launches while green indicates successful launches.
- The KSC LC-39A launch site had the highest success rate

<Folium Map Screenshot 3>





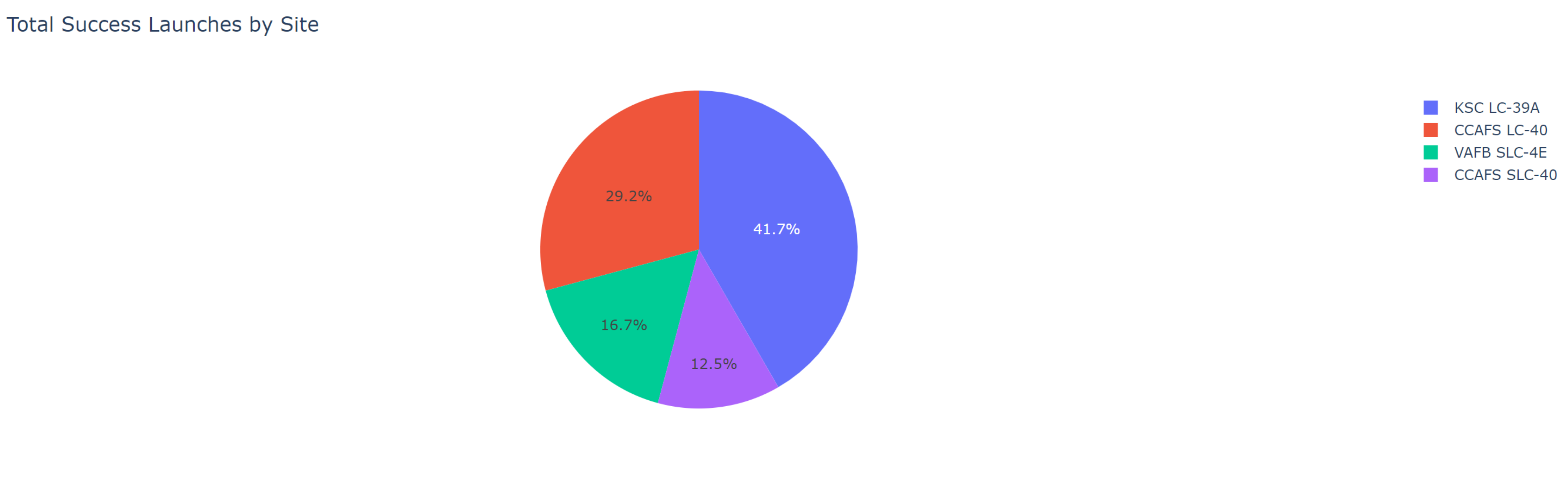
- Are launch sites in close proximity to railways? Yes
- Are launch sites in close proximity to highways? Yes
- Are launch sites in close proximity to coastline? Yes
- Do launch sites keep certain distance away from cities? No



Section 4

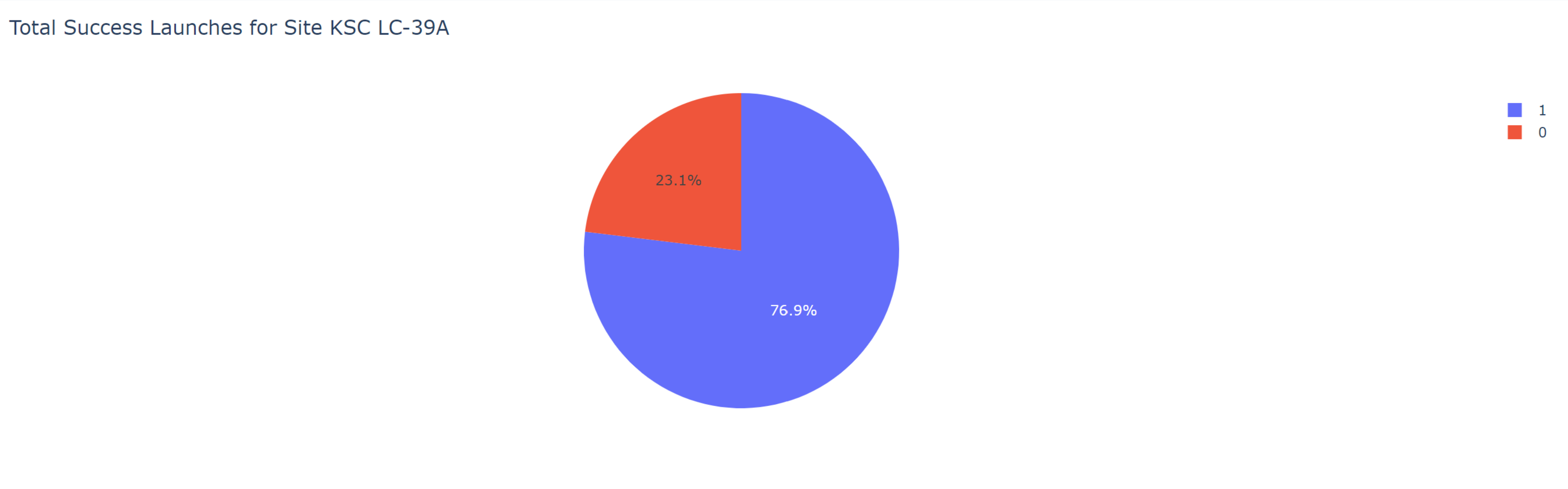
Build a Dashboard with Plotly Dash

Total Successful Launches by Launch Site



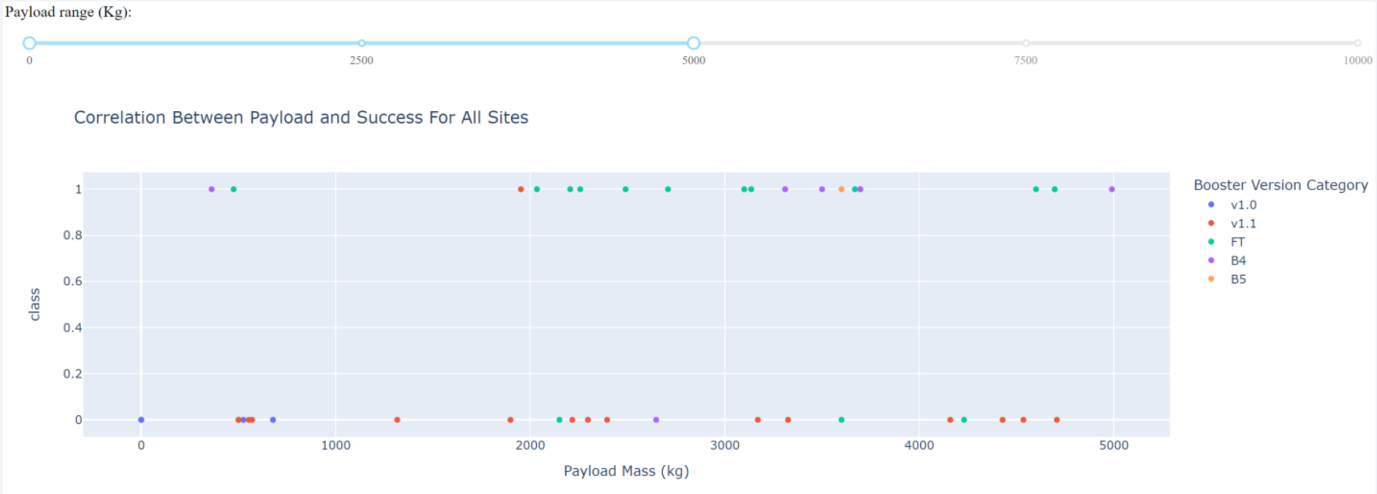
The chart shows the number of successful launches for all the launch sites. KSC LC-39A has the most successful launches.

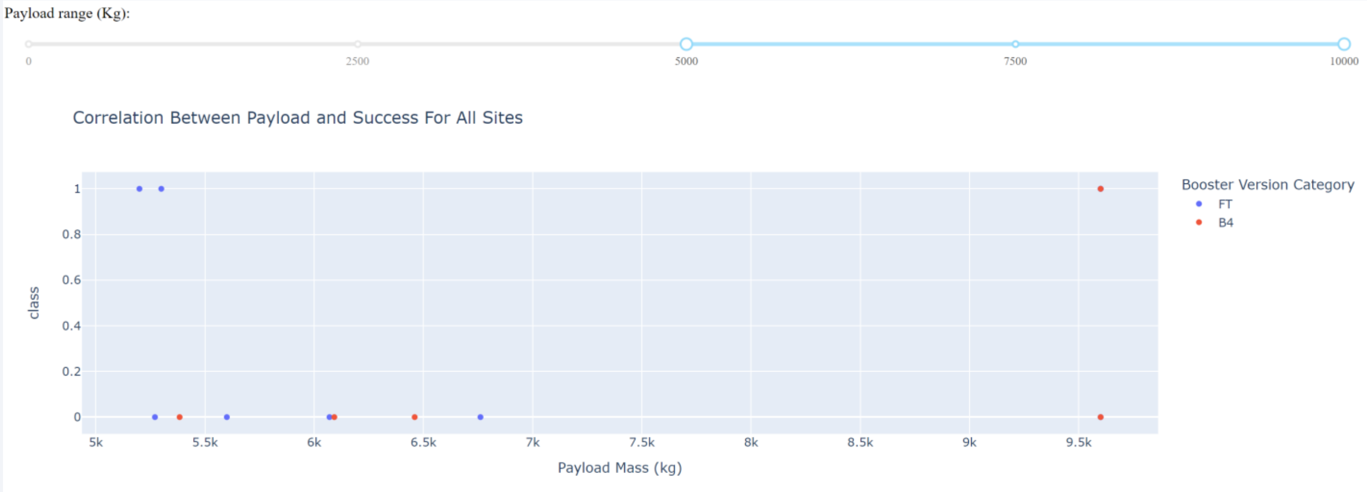
Launch Site with the Highest Launch Success Rate



The chart shows that KSC LC-39A has a 76.9% success rate and a 23.1% failure rate.

Payload Mass vs. Success Rate for All Launch Sites





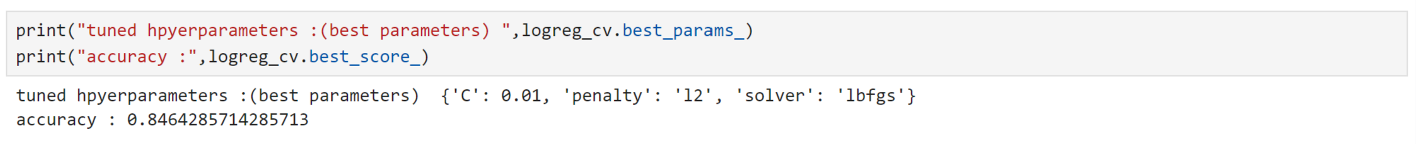
The scatterplot shows that light payloads have better success rates than heavy payloads.

Section 5

Predictive Analysis (Classification)

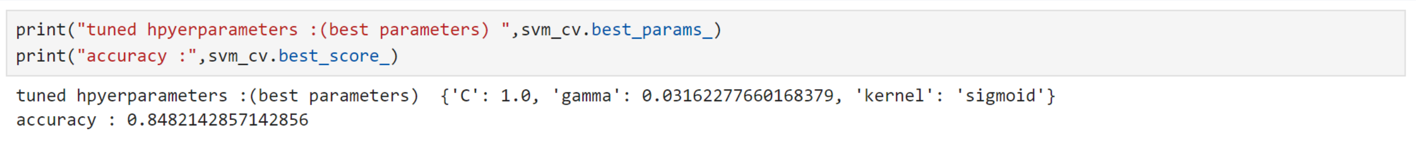
Classification Accuracy

LogReg



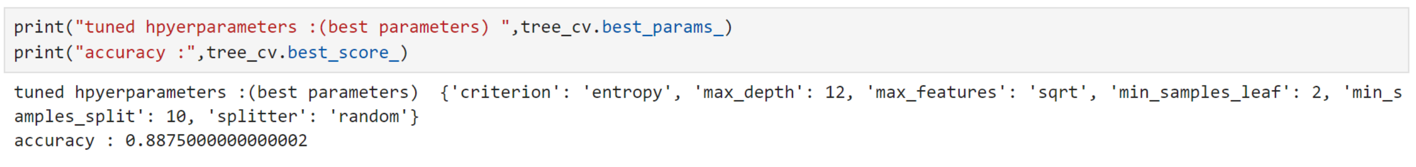
```
print("tuned hpyerparameters :(best parameters) ",logreg_cv.best_params_)
print("accuracy :",logreg_cv.best_score_)

tuned hpyerparameters :(best parameters)  {'C': 0.01, 'penalty': 'l2', 'solver': 'lbfgs'}
accuracy : 0.8464285714285713
```

SVM



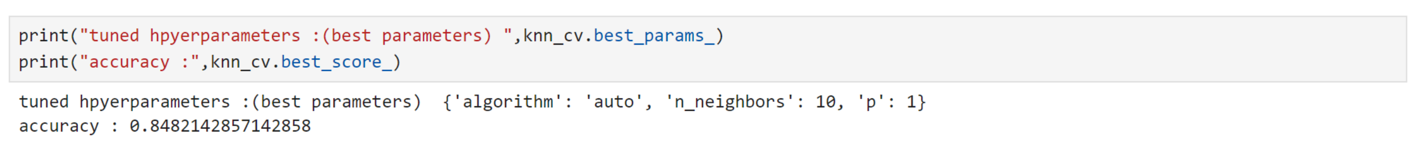
```
print("tuned hpyerparameters :(best parameters) ",svm_cv.best_params_)
print("accuracy :",svm_cv.best_score_)

tuned hpyerparameters :(best parameters)  {'C': 1.0, 'gamma': 0.03162277660168379, 'kernel': 'sigmoid'}
accuracy : 0.8482142857142856
```

Tree



```
print("tuned hpyerparameters :(best parameters) ",tree_cv.best_params_)
print("accuracy :",tree_cv.best_score_)

tuned hpyerparameters :(best parameters)  {'criterion': 'entropy', 'max_depth': 12, 'max_features': 'sqrt', 'min_samples_leaf': 2, 'min_s
amples_split': 10, 'splitter': 'random'}
accuracy : 0.8875000000000002
```

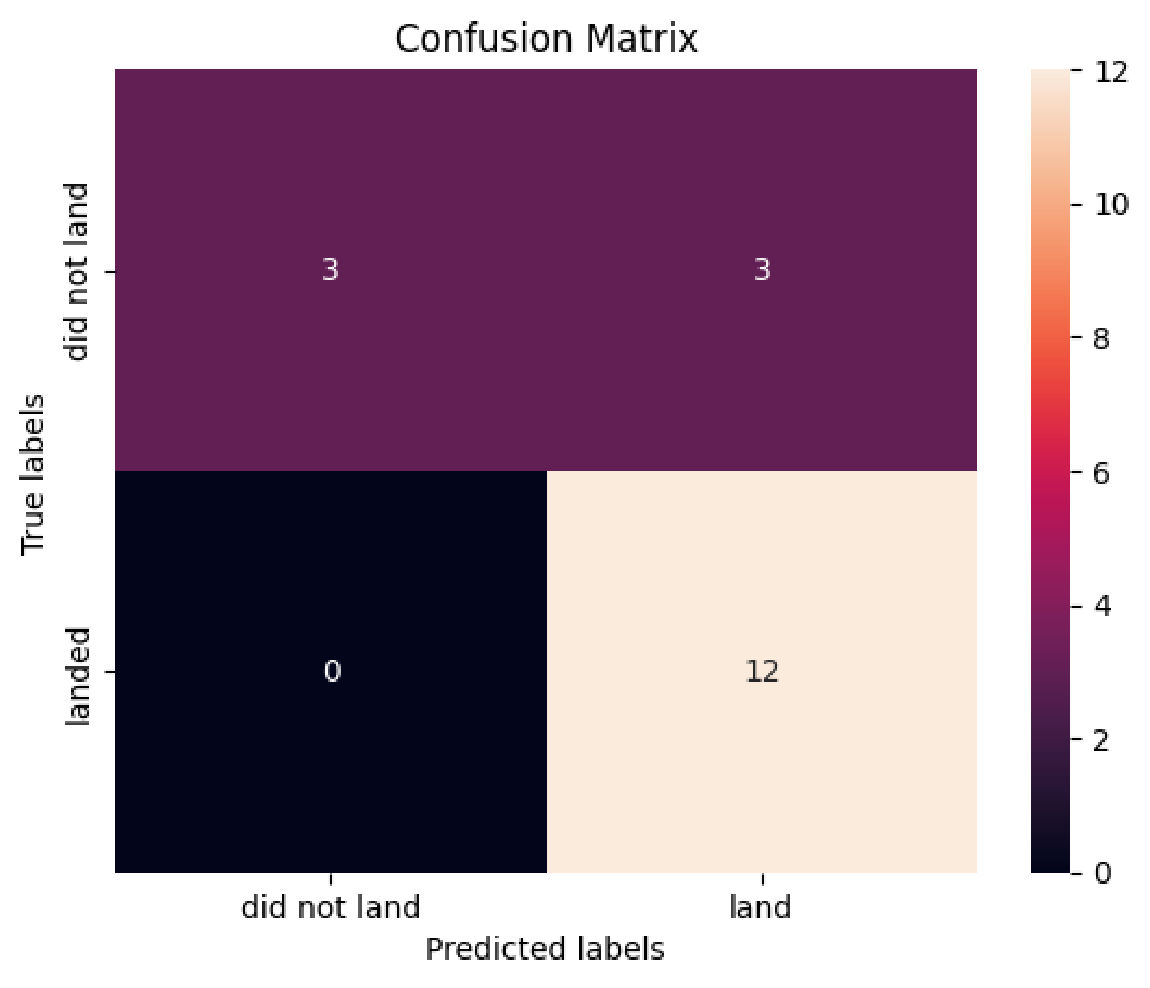
KNN



```
print("tuned hpyerparameters :(best parameters) ",knn_cv.best_params_)
print("accuracy :",knn_cv.best_score_)

tuned hpyerparameters :(best parameters)  {'algorithm': 'auto', 'n_neighbors': 10, 'p': 1}
accuracy : 0.8482142857142858
```

- The accuracy for the test data on all the models are the same.
- However, for the training data, the decision tree model had the highest accuracy.

Confusion Matrix



- The confusion matrix for the decision tree model clearly shows the classifier can distinguish between the different classes.
- The major problem comes from false positives. In this context, the classifier marks a failed landing as a successful landing.

Conclusions

- Factors such as orbit type, number of previous launches, and payload mass influence whether a space mission will be successful.
- Orbits with the best successful rates: ES-L1, GEO, HEO, and SSO.
- The successful rate started to increase in 2013.
  - We can assume that knowledge has been gained between launches, which in turn, increases the probability of a successful launch.
- KSC LC-39A was the launch site with the highest number of successful launches.
  - More information should be gathered as to why some launch sites are better than others.
- Light payloads perform better than heavy payloads.
- The decision tree model had the best train accuracy.

Thank you!