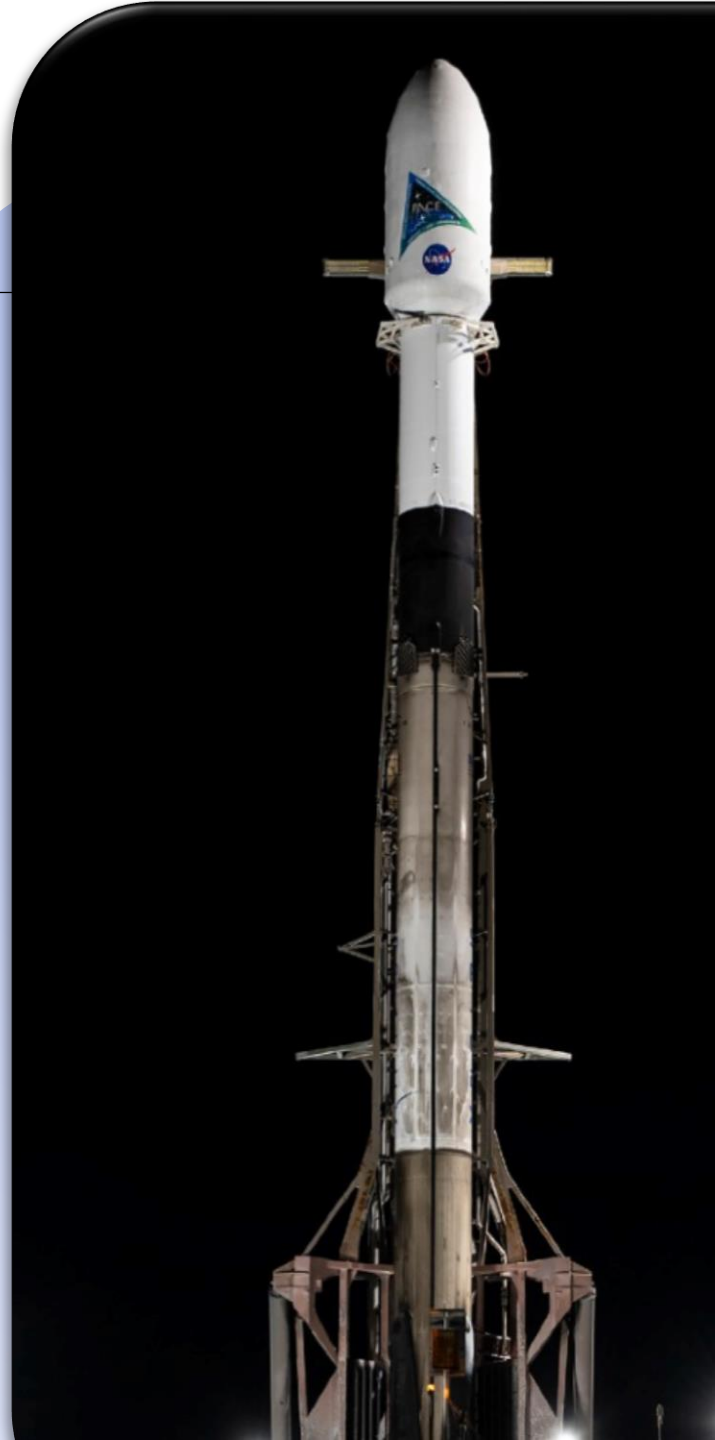
Winning the space through Data Science

Pabasara Oneth Weragalaarachchi

12th February 2024

Contents



Executive Summary

- **Summary of the methodologies**
  - ✓ Data collection through the SpaceX REST API
  - ✓ Data collection with the web scraping technique
  - ✓ Data Wrangling
  - ✓ Exploratory Data Analysis with SQL
  - ✓ Exploratory Data Analysis with Data Visualization
  - ✓ Interactive Visual Analytics with Folium

- **Summary of all results**
  - ✓ Exploratory Data Analysis result
  - ✓ Interactive analytics in screenshots
  - ✓ Predictive Analytics result

Introduction

SpaceX, is a leader in the space industry. SpaceX is known for its ambitious goals of reducing the cost of space travel and eventually enabling the colonization of other planets, particularly Mars. The company has developed a series of innovative rockets and spacecraft, notably the Falcon 1, Falcon 9, and Falcon Heavy rockets, as well as the Dragon spacecraft. Additionally, SpaceX developed a system to reuse the first stage of Falcon 9 rockets and they have reduced the cost per launch ($62 million). Other companies spend about $165 million per launch. By determining if the first stage of Falcon 9 will land or not we can determine the cost per launch. To do this, we use public data and machine learning models to predict if SpaceX can reuse the first stage.

- Problems to find answers
  - What factors affect a successful land?
  - Operating conditions for a successful land.
  - The interaction of the conditions for a successful land.

Methodology

Methodology

- Collection of the data through the SpaceX REST API and by web scraping through Wikipedia.
- Wrangling of data by filtering the data, handling missing values, and applying one hot encoding to prepare the data collected for analysis and modeling.
- Explore the data using SQL and other data visualization techniques.
- Visualize the data using Plotly Dash and Folium.
- Perform predictive analysis using classification models.

Data Collection

- The data was collected from various sources
  - ✓ Data was collected by using a GET request to the SpaceX API.
  - ✓ After that, I decoded the response as a JSON using the .json function call and turned it into pandas dataframe by using the .json.normalize() function.
  - ✓ I then cleaned the data, checked for missing values, and filled in missing values where necessary.
  - ✓ Lastly, I performed a web scrape from Wikipedia for Falcon 9 launch records with Beautiful Soup.
  - ✓ The objective was to extract the launch records as an HTML table, parse the table, and convert it into pandas dataframe for future analysis.

Data Collection – SpaceX API

- I used the GET request to collect, clean the requested data and some basic data wrangling and formatting to the obtained data through the SpaceX API.

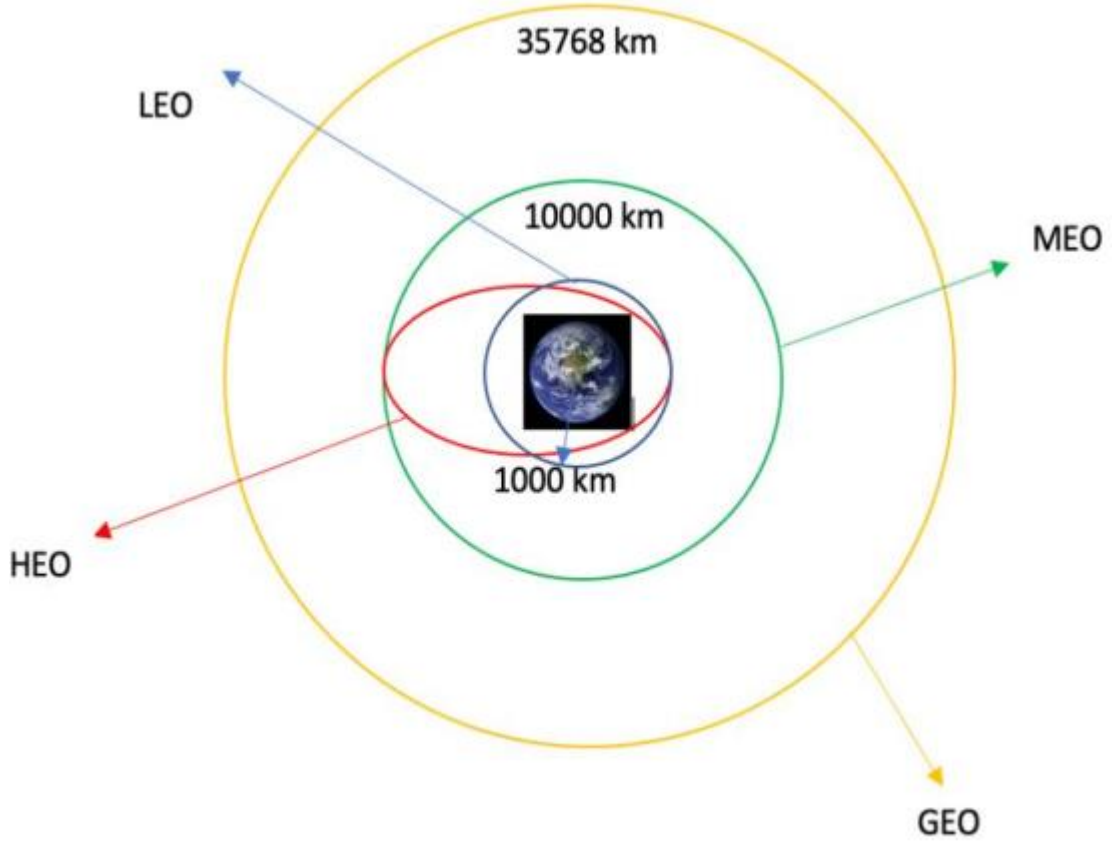
- This link to the notebook is https://github.com/OnethP/IBM_Data_Science_Capstone_Project/blob/main/jupyter-labs-spacex-data-collection-api.ipynb

Data Collection - Scraping

- I applied web scraping to web scrape Falcon 9 launch records with BeautifulSoup.
- I converted the table into a pandas dataframe.
- The link to my notebook is https://github.com/OnethP/IBM_Data_Science_Capstone_Project/blob/main/jupyter-labs-webscraping.ipynb

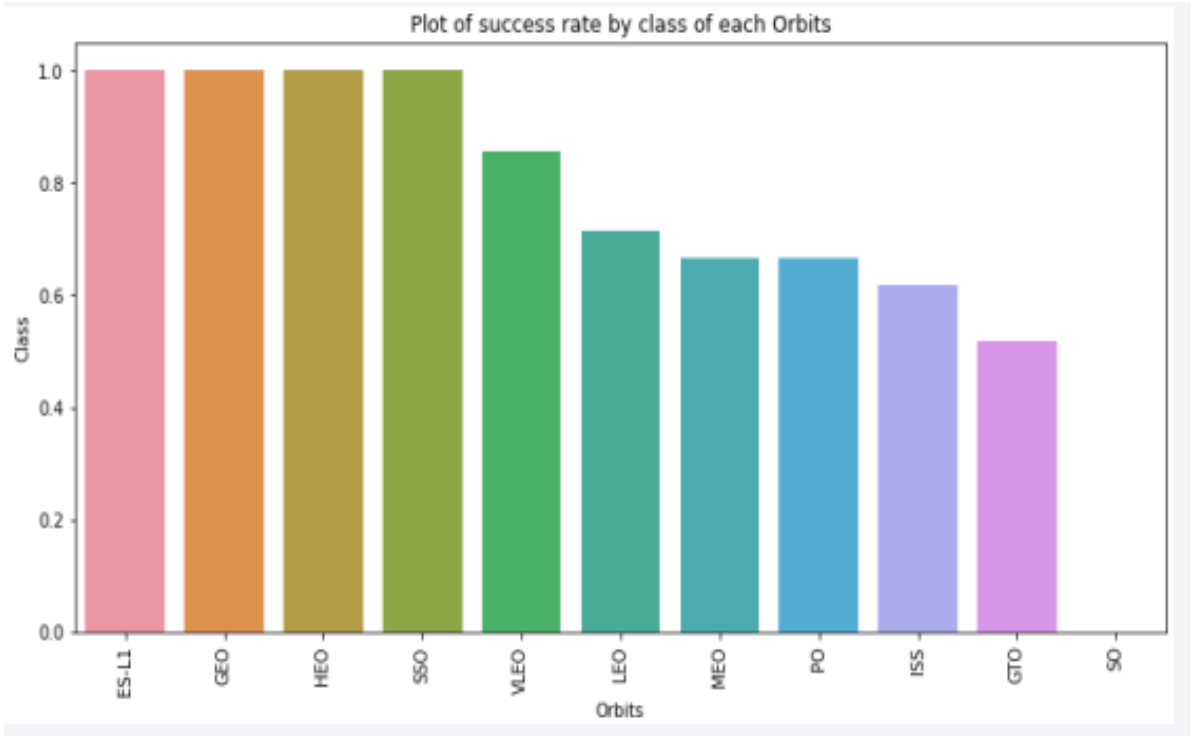
Data Wrangling

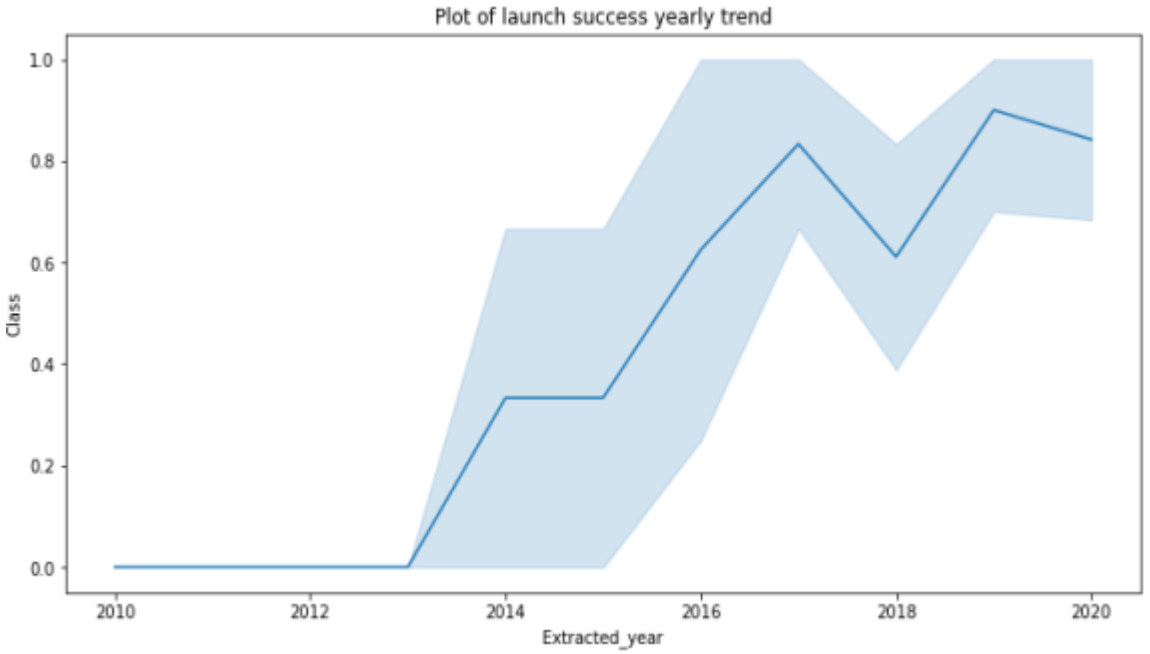
- I performed exploratory data analysis and determined the training labels.
- I calculated the number of launches at each specific site, and the number of occurrences of each different orbit.
- I created the landing outcome label from the outcome column and exported it into a CSV file.
- The link to the notebook is https://github.com/OnethP/IBM_Data_Science_Capstone_Project/blob/main/labs-jupyter-spacex-Data%20wrangling.ipynb



EDA with Data Visualization

- I explored the data by visualizing the relationship between flight number and launch site, the success rate of each orbit type, flight number, payload and launch site, and the launch success yearly trend.





The link to the notebook is https://github.com/OnethP/IBM_Data_Science_Capstone_Project/blob/main/jupyter-labs-eda-dataviz.ipynb.jupyterlite.ipynb

EDA with SQL

- I loaded the SpaceX data into a PostgreSQL database.
- I applied EDA with SQL to get insight from the data. I wrote queries to find out for example
  - The total number of failed and successful mission outcomes.
  - The total payload mass carried by boosters of NASA.
  - The names of the unique rocket launch sites in the space mission.
  - The failed landing attempts in drone ship, launch site name and the booster version.

- The link to the notebook is https://github.com/OnethP/IBM_Data_Science_Capstone_Project/blob/main/jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an interactive map with Folium

- I marked all launch sites and lines to mark the success or failure of each launch site and map objects like markers, circles, and lines into the folium map.
- By the colored marker clusters, I identified which launch sites mentioned have a high success rate.
- I assigned launch outcomes (0 for failure and 1 for success)
- I calculated the distance between a launch site and the public places around it. Like
  - ❖ The distance from cities
  - ❖ Are the launch sites near public places and coastal areas?

Build a Dashboard with Plotly Dash

- I made an interactive dashboard with the Plotly Dash
- I plotted some pie charts showing the number of total launches by specific launch sites.
- I plotted a scatter chart to show to relationship between the payload and outcome for different boosters.

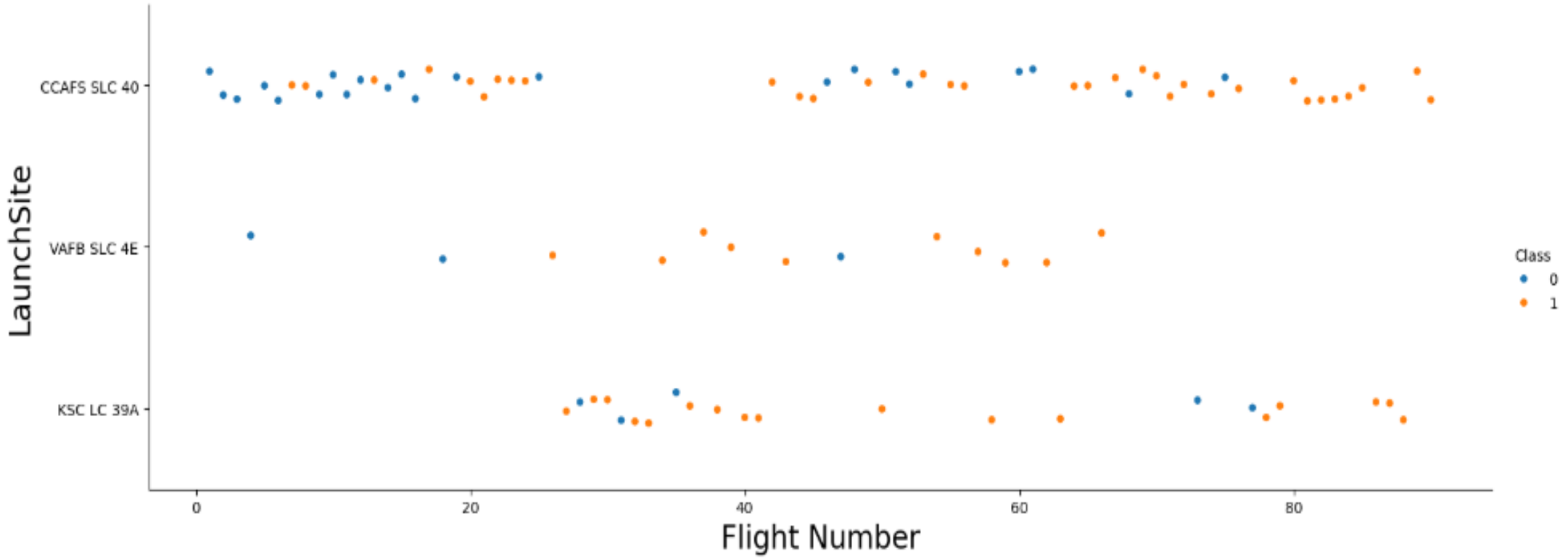
Predictive Analysis

- I loaded the data using pandas and Numpy, transformed it, and split it into training and testing
- I built various machine learning models and tuned them to various hyperparameters using GridSearchCV
- I found the most suitable classification model.
- The link to the notebook is https://github.com/OnethP/IBM_Data_Science_Capstone_Project/blob/main/SpaceX_Machine_Learning_Prediction_Part_5.jupyterlite.ipynb

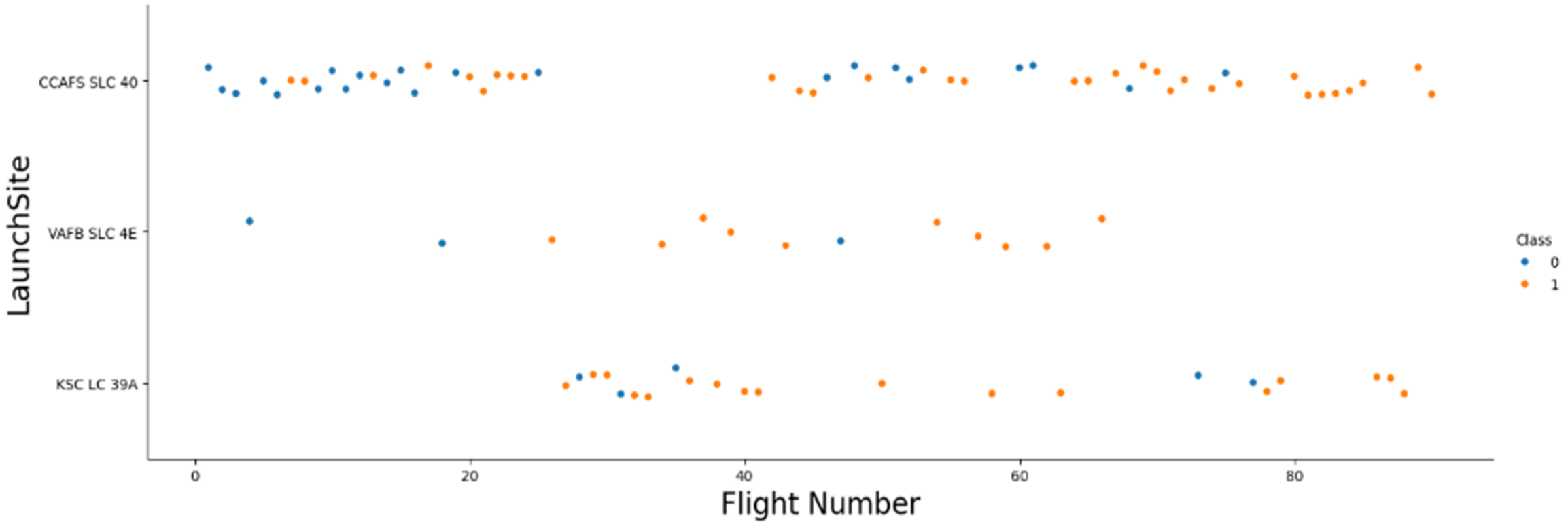
Results from EDA

Flight Number vs. Launch Site

- From this scatter plot, we can see that if the flight amount at a launch site is high, the greater the success rate at a launch site.

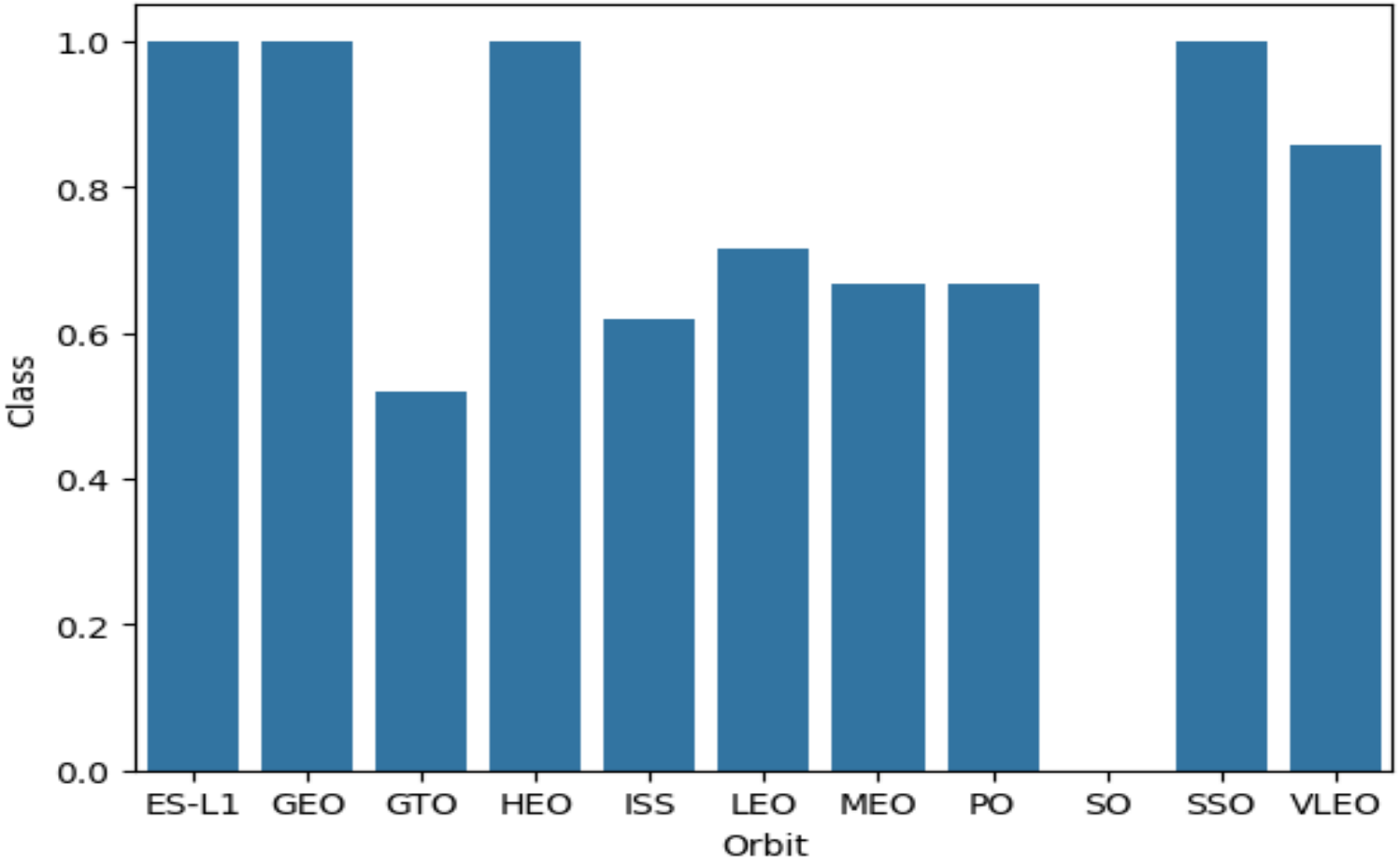


Payload vs. Launch Site



If the payload mass is bigger in the launch site CCAFS SLC 40 the success rate of the rocket is also high.

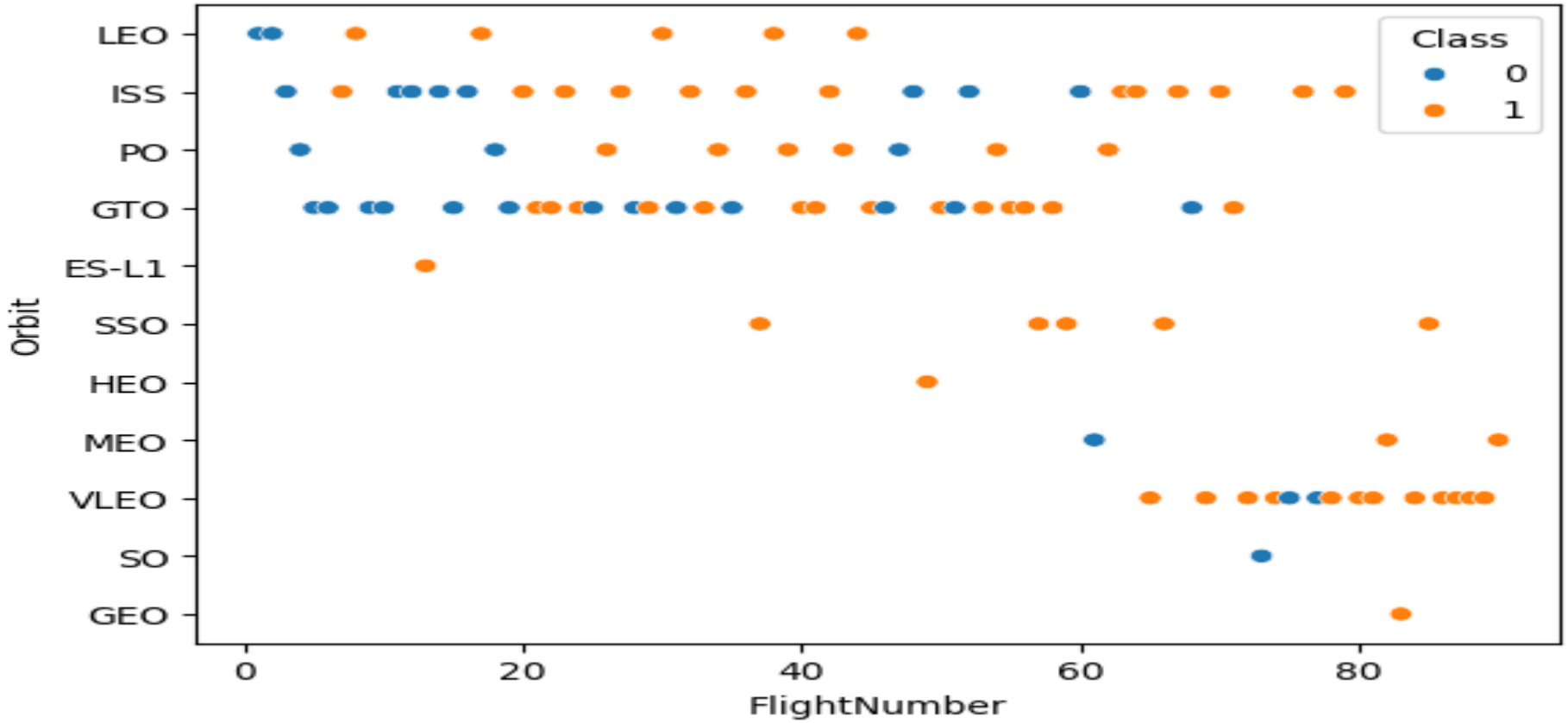
Success Rate vs. Orbit Type



- From the plot mentioned, we can conclude that ES-L1, GEO, HEO, SSO, and VLEO had the most success rate.

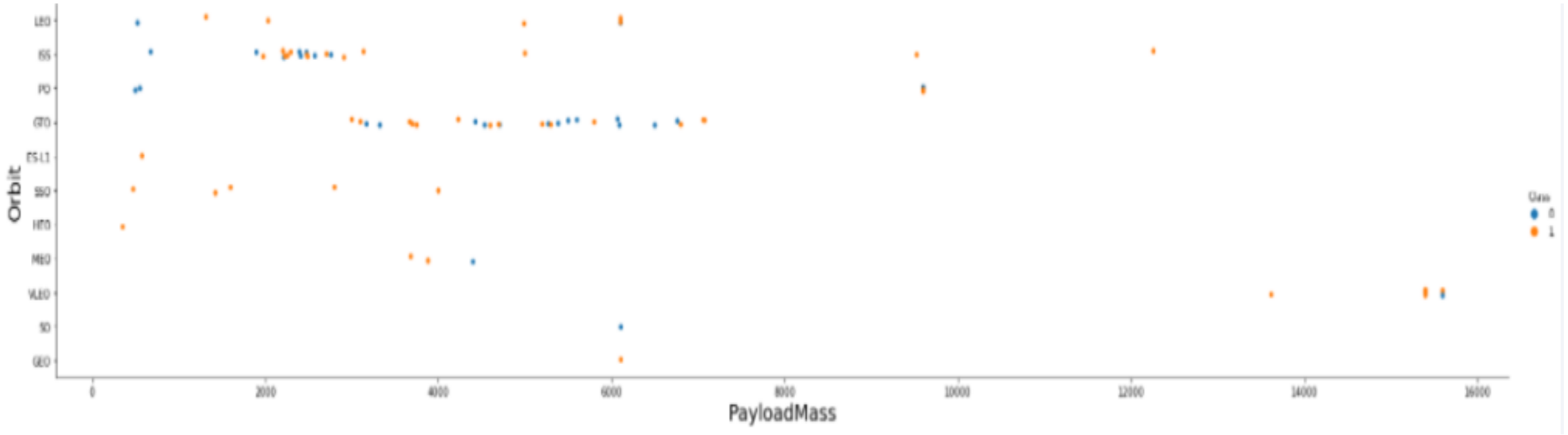
Flight number vs. Orbit Type

- The below-mentioned scatter chart shows Flight number vs. Orbit Type. We can observe that the LEO orbit success is related with number of flights but in the GTO orbit there is no relationship between orbit type and flight number.



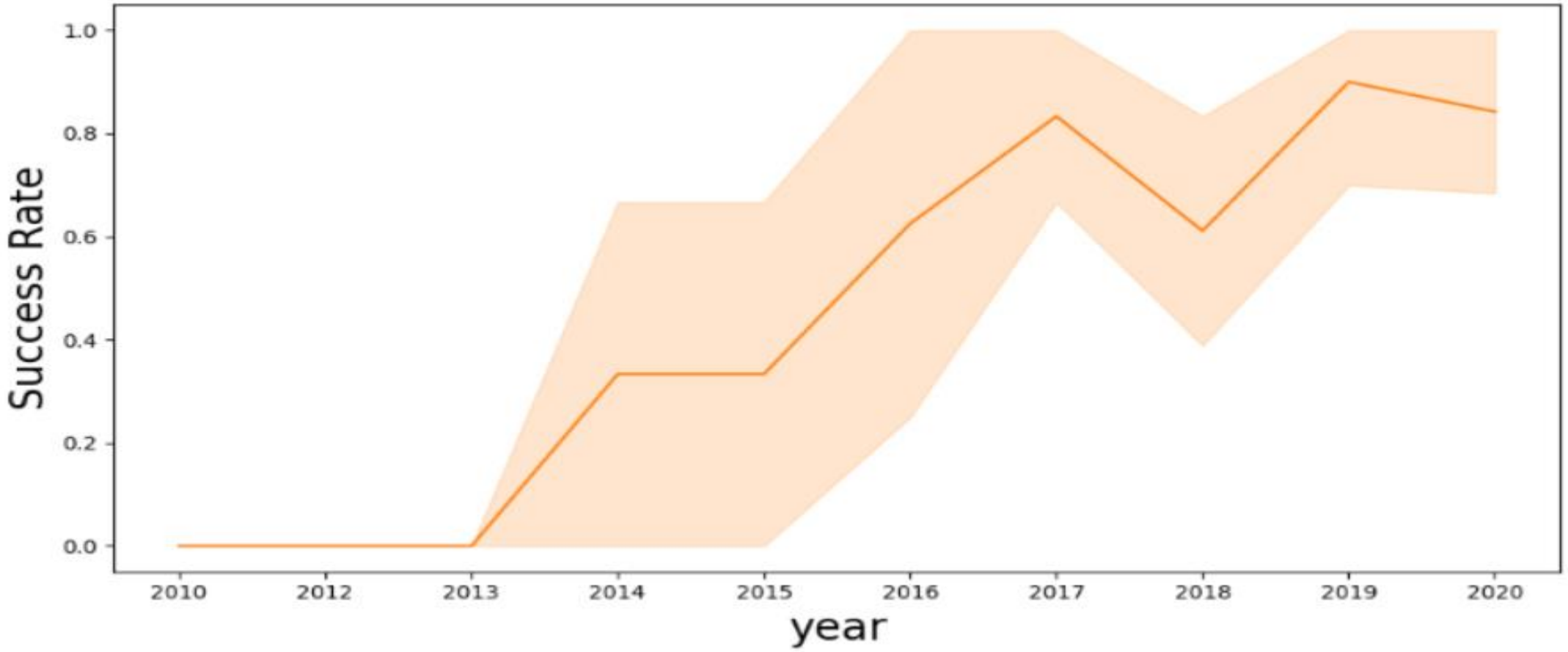
Payload vs. Orbit Type

- We can observe that successful landings are more with heavy payloads for PO, LEO, and ISS orbits.



Launch Success over time

- From this chart we can observe that the success rate kept increasing from 2013 to 2020.



Launch site names

- I used the **DISTINCT** keyword to show only unique launch sites from the Data.

```
In [9]: sql SELECT DISTINCT LAUNCH_SITE FROM SPACEXTBL ORDER BY 1;
```

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

Out[9]:

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

Records with launch sites starting CCA

- Displaying the 5 records where launch sites begin with ‘CCA’

```
In [10]: sql SELECT * FROM SPACEXTBL WHERE LAUNCH_SITE LIKE 'CCA%' LIMIT 5;
 * sqlite:///my_data1.db
Done.
```

Out[10]:

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

Payload Mass

Total Payload Mass

- Boosters produced by NASA (CRS) carried **45,596 kg (total)**

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

```
In [11]: sql SELECT SUM (PAYLOAD_MASS__KG_) FROM SPACEXTBL WHERE CUSTOMER='NASA (CRS)'
 * sqlite:///my_data1.db
Done.
```

Out[11]:

SUM (PAYLOAD_MASS__KG_)
45596

Average Payload Mass

- F9 v1.1 boosters carried **2,534 kg (Average)**

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

```
In [12]: sql SELECT AVG(PAYLOAD_MASS__kg_) FROM SPACEXTBL WHERE booster_version LIKE 'F9 v1.1%'
 * sqlite:///my_data1.db
Done.
```

Out[12]:

AVG(PAYLOAD_MASS__kg_)
2534.6666666666665

1st successful landing on ground pad

- The first successful ground pa landing was recorded on the 22nd December 2015.

```
In [13]: sql SELECT MIN(DATE) FROM SPACEXTBL WHERE "Landing_Outcome" = 'Success (ground pad)'
 * sqlite:///my_data1.db
Done.
```

Out[13]:

MIN(DATE)
2015-12-22

Successful drone ship landing with payload 4000 - 6000

```
In [14]: %sql select BOOSTER_VERSION from SPACEXTBL where PAYLOAD_MASS__KG_ BETWEEN 4000 AND 6000
 * sqlite:///my_data1.db
Done.
```

Out[14]:

Booster_Version
F9 v1.1
F9 v1.1 B1011
F9 v1.1 B1014
F9 v1.1 B1016
F9 FT B1020
F9 FT B1022
F9 FT B1026
F9 FT B1030
F9 FT B1021.2
F9 FT B1032.1
F9 B4 B1040.1
F9 FT B1031.2
F9 B4 B1043.1
F9 FT B1032.2
F9 B4 B1040.2
F9 B5 B1046.2
F9 B5 B1047.2
F9 B5 B1046.3
F9 B5B1054
F9 B5 B1048.3
F9 B5 B1051.2
F9 B5B1060.1
F9 B5 B1058.2
F9 B5B1062.1

- These booster successfully landed onto a drone ship with a payload of greater than 4000 but less than 6000.

Count of successful landings

- Number of landings between 04-06-2010 until 20-03-2017 in descending order.

```
%sql SELECT [Landing _Outcome], count(*) as count_outcomes \
FROM SPACEXTBL \
WHERE DATE between '04-06-2010' and '20-03-2017' group by [Landing _Outcome] order by count_outcomes DESC;
```

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

Landing _Outcome	count_outcomes
Success	20
No attempt	10
Success (drone ship)	8
Success (ground pad)	6
Failure (drone ship)	4
Failure	3
Controlled (ocean)	3
Failure (parachute)	2
No attempt	1

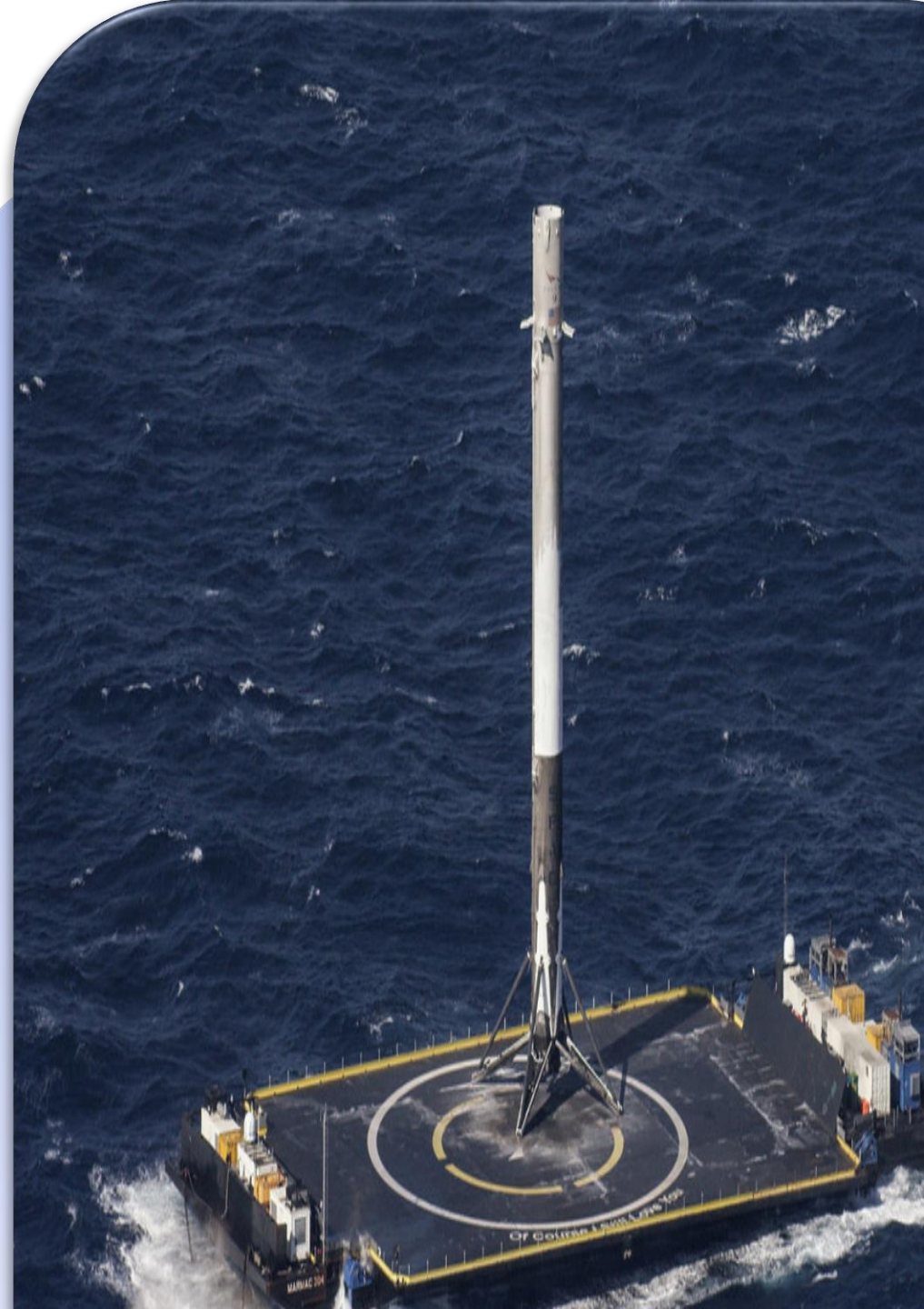
Failed landings on Drone ship

- Showing the booster version, Launch site, landing outcome, date, and month.

```
%sql SELECT substr(Date,4,2) as month, DATE,BOOSTER_VERSION, LAUNCH_SITE, [Landing _Outcome] \
FROM SPACEXTBL \
where [Landing _Outcome] = 'Failure (drone ship)' and substr(Date,7,4)='2015';
```

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

month	Date	Booster_Version	Launch_Site	Landing _Outcome
01	10-01-2015	F9 v1.1 B1012	CCAFS LC-40	Failure (drone ship)
04	14-04-2015	F9 v1.1 B1015	CCAFS LC-40	Failure (drone ship)



Boosters Maximum Payload

- These are the boosters that carry the maximum payload mass.

In [21]:
```
sql SELECT BOOSTER_VERSION,PAYLOAD_MASS__KG_ FROM SPACEXTBL WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXT
```

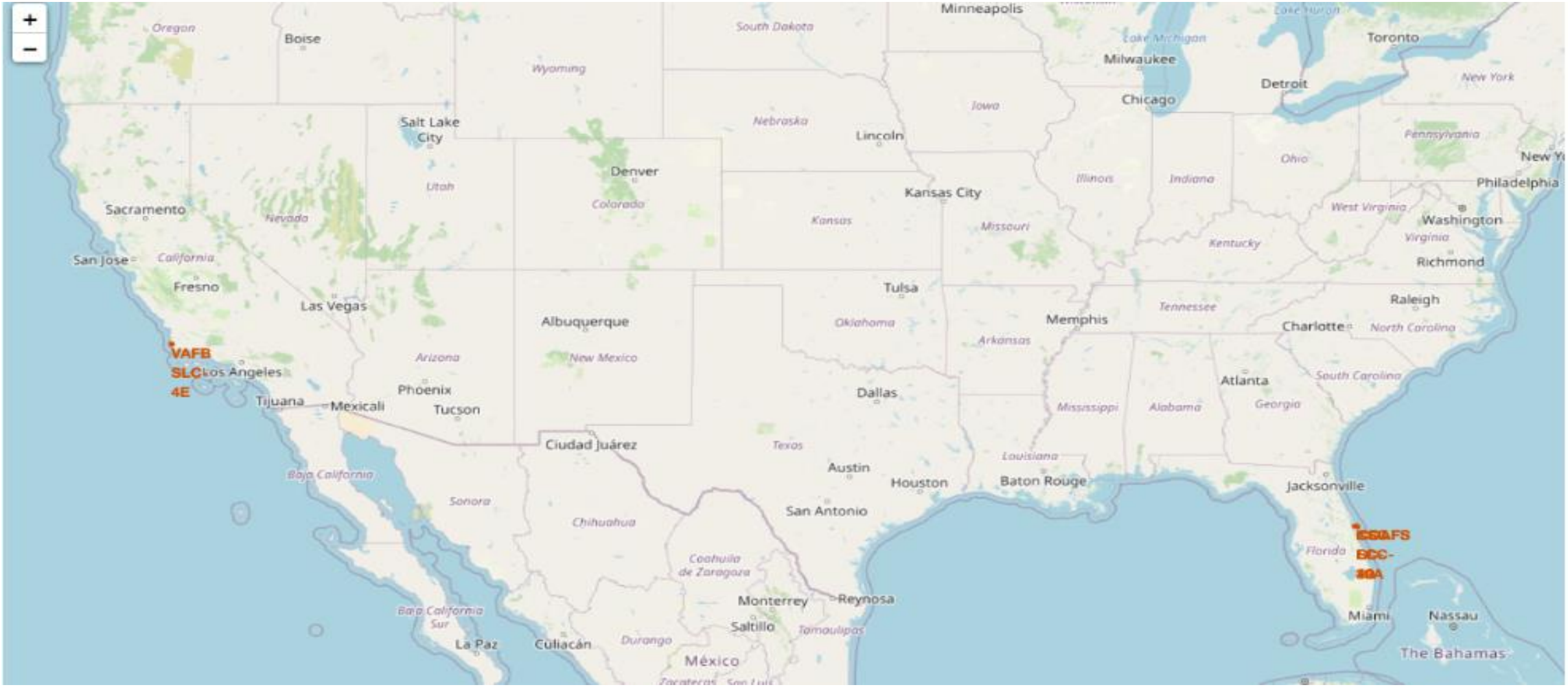
```
 * sqlite:///my_data1.db
Done.
```

Out[21]:

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

Launch site analysis

Launch sites



- **We can see that the SpaceX launch stations in the USA in located on the Florida and California coasts.**

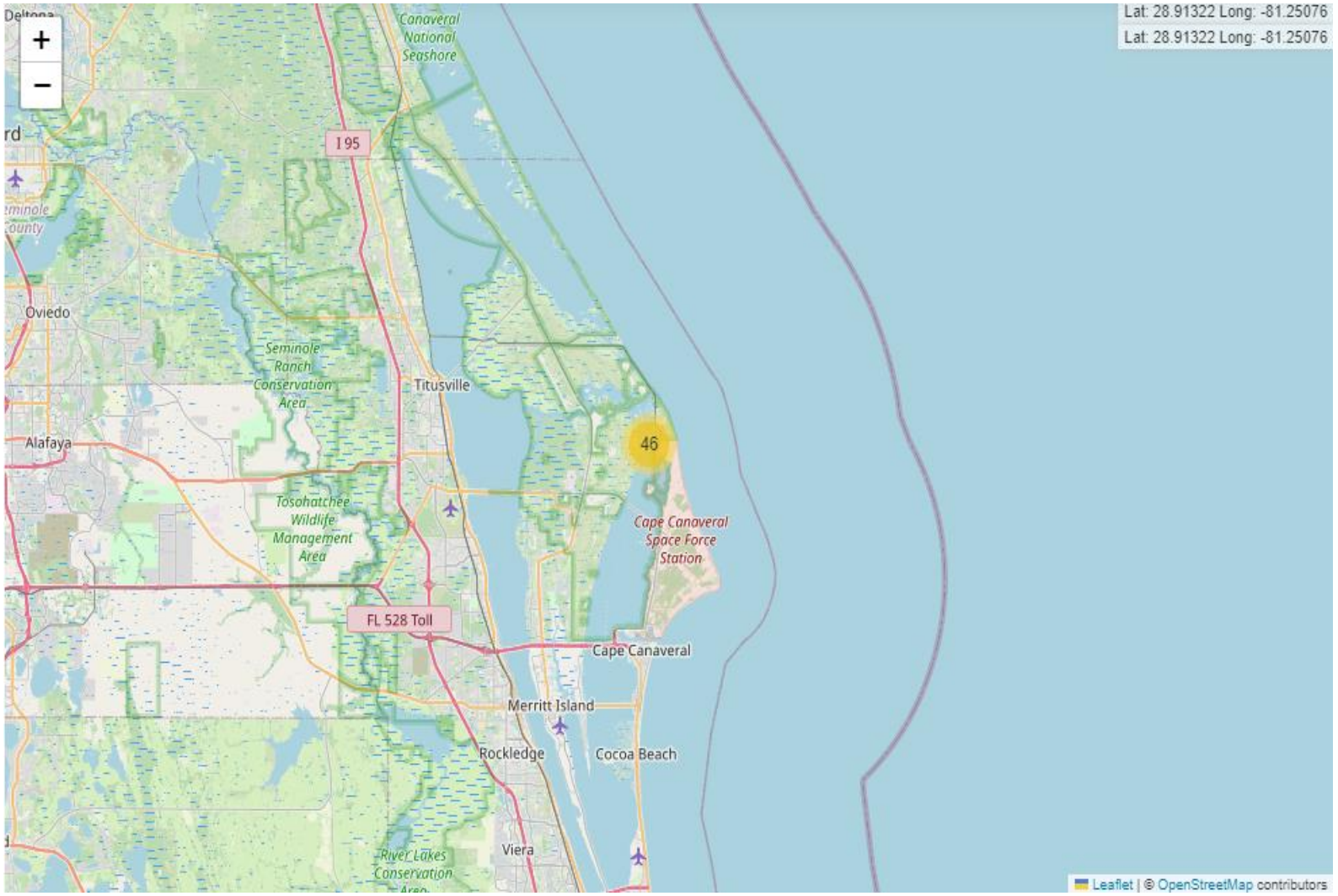
Launch outcomes – Color Labels

- **Green markers** are for successful launches.
- **Red markers** are for unsuccessful launches.
- Launch site **CCAFS SLC-40** has a **3/7 success rate**

Distance to proximities

CCAFS SLC-40

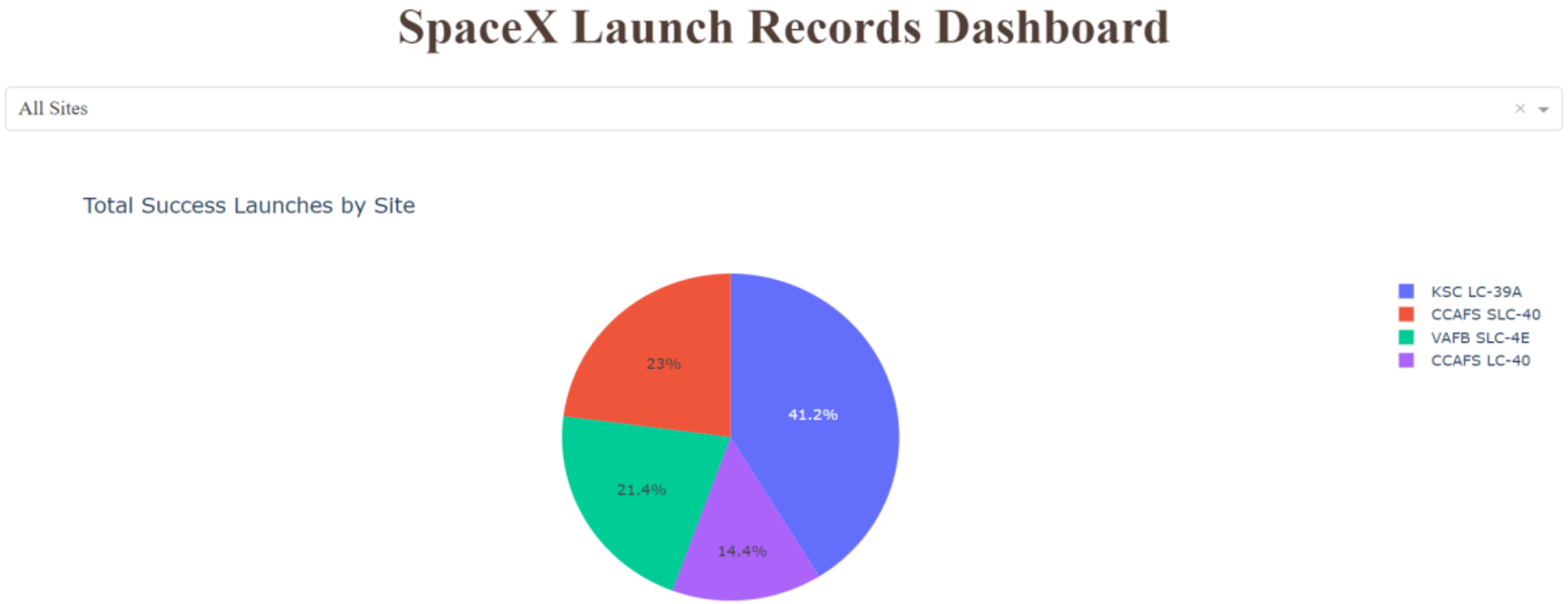
- **Coasts** – help prevent the falling of used stages or failures on people.
- **Cities and Infrastructure** – This must be away from the launch site to prevent damage to property in case of a failure but be close enough for railway tracks, and docks to bring materials and people to the launch site.



Dashboard with Plotly Dash

Launch Success by Site

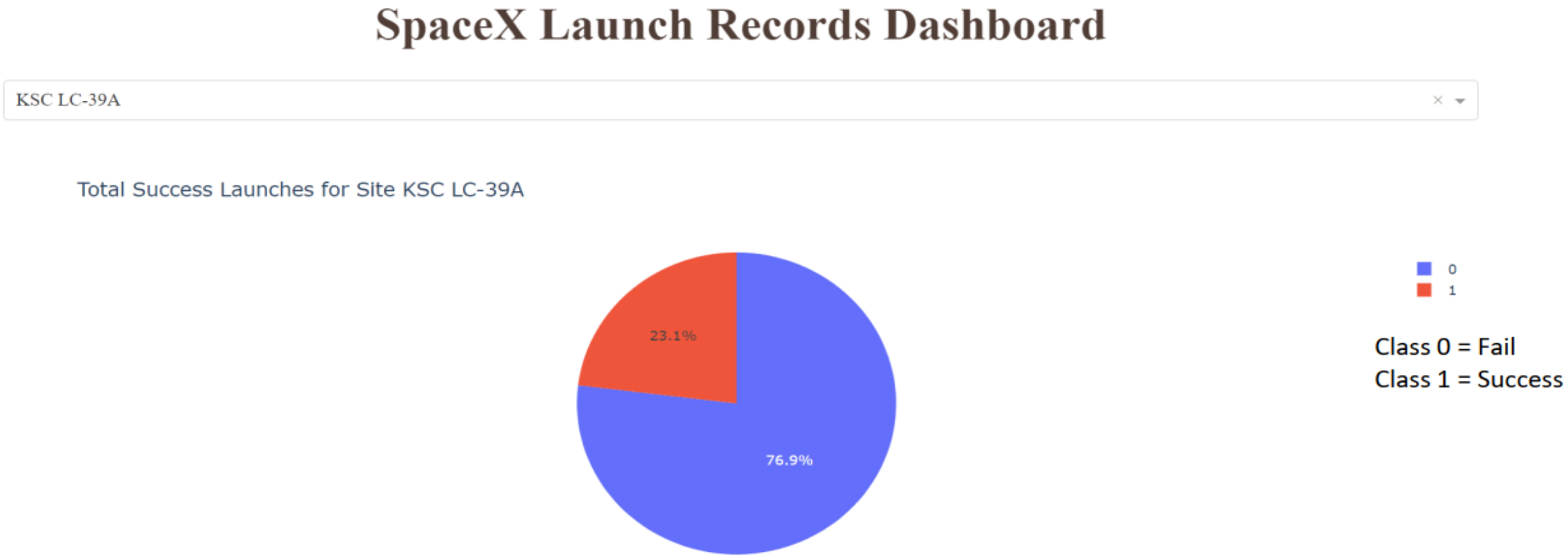
- **KSC LC-39A** has the **most successful launches** amongst the launch sites in this data.



Launch Success

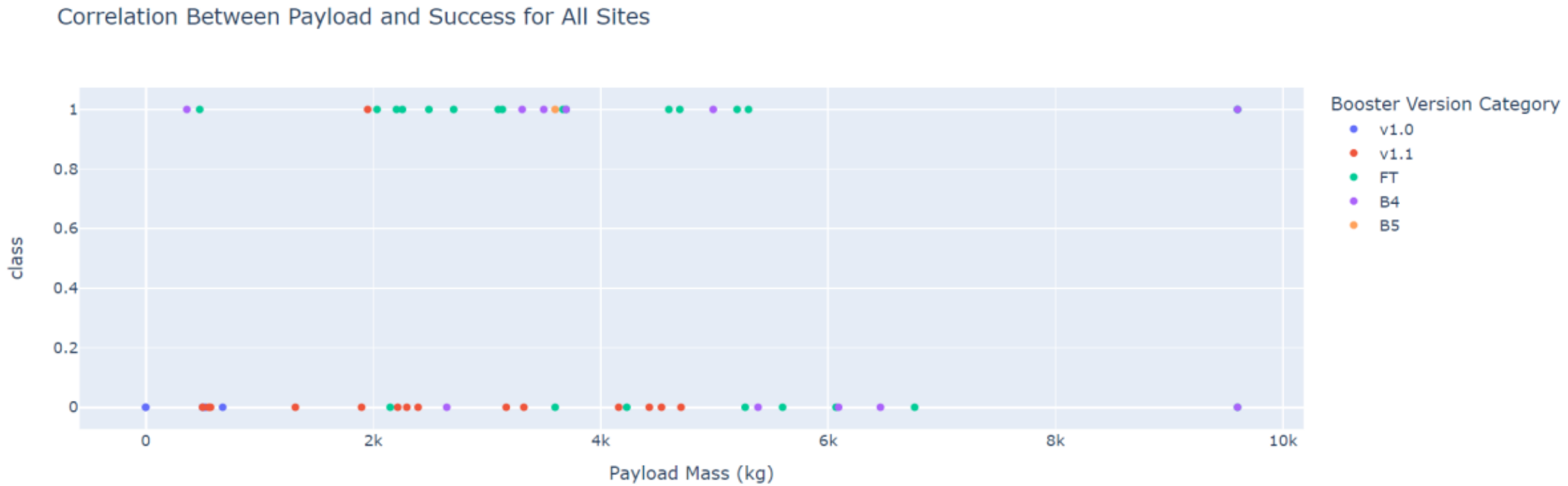
KSC LC-39A

- **KSC LC-39A** has the highest success rate among all the launch sites in the dataset.
- It has a success rate of **10/13** with **10 successful** and **3 unsuccessful** launches.



Success and Payload Mass

- Payloads between **2,000 kg** to **5,000 kg** have the highest success rate.
- 0 shows unsuccessful outcome while 1 shows successful outcome.



Predictive Analytics (Classification)

Classification Accuracy

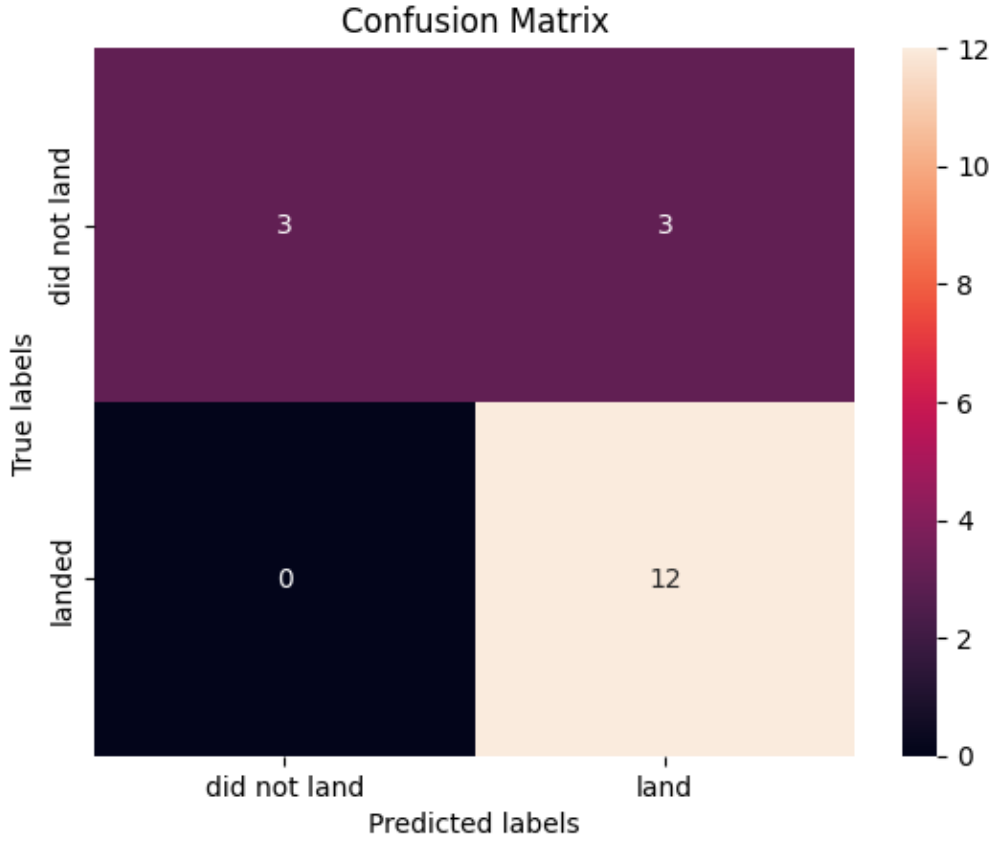
- The decision tree is the model which scored the highest classification Accuracy.

```
In [30]: tree_score = tree_cv.score(X_test, Y_test)

In [31]: print(f"Decision Tree - Accuracy using method score: {tree_score}")

         Decision Tree - Accuracy using method score: 0.8888888888888888
```

We can plot the confusion matrix

Confusion Matrices

- The confusion matrix made for the decision tree that it can distinguish between different classes. **An unsuccessful landing is marked as a successful landing** by the classifier. It is a big problem.



Conclusions

- Model Performance: The model is **slightly outperforming.**
- Launch success has increased over time.
- Across all launch sites if we **increase the Payload**, the **success rate is high.**
- Booster **KSC LC-39A** has a **100%** success rate for launches below **5,500 kg**.
- All the launch sites have been built near the coastal line.
- Orbits **ES-L1, SSO, GEO, and HEO** had the highest success rate.

Thank You!