IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Mobarak Hossain
Feb 14 2023

Outline

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

Executive Summary

- Summary of methodologies
  Data Collection
  Data Wrangling
  Exploratory Data analysis with Data Visualization
  Exploratory Data Analysis with SQL
  Building an Interactive map with Folium
  Building a dashboard with Plotly dash
  Machine learning prediction
- Summary of all results
  Result from Exploratory Data Analysis
  Interactive visual analytics and dashboard
  Predictive analysis

Introduction

- Project background and context
  Commercial space age is here, companies are working on different projects and investing more on research and development to make space travel affordable. Among them, most successful one is SpaceX. Sending multiple manned mission to space and doing it inexpensively compared to other companies. They claim Falcon 9 rocket launches to be around 62 million dollars compared to other companies costs upwards of 165 million dollars because SpaceX can reuse the first stage of the launch.
- Problems you want to find answers
  Task is to determine the price of each launch. If we can determine if first stage will land, then we can determine the cost of the launch. I will be doing it by gathering information about SpaceX and by creating dashboard for the team. Will also determine if SpaceX will reuse the first stage.
  Will use different variables like payload mass, launch site, number of flights and orbit to check If it will affect the landing and will use predictive analysis.

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - SpaceX data was collected using SpaceX Rest API
  - Using Web scraping from Wikipedia
- Perform data wrangling
  - Combining complex data sets and making them more easier to analyze
  - Fixing the missing values
  - Representing categorical data using integer or float dummy numbers- one hot encoding
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash

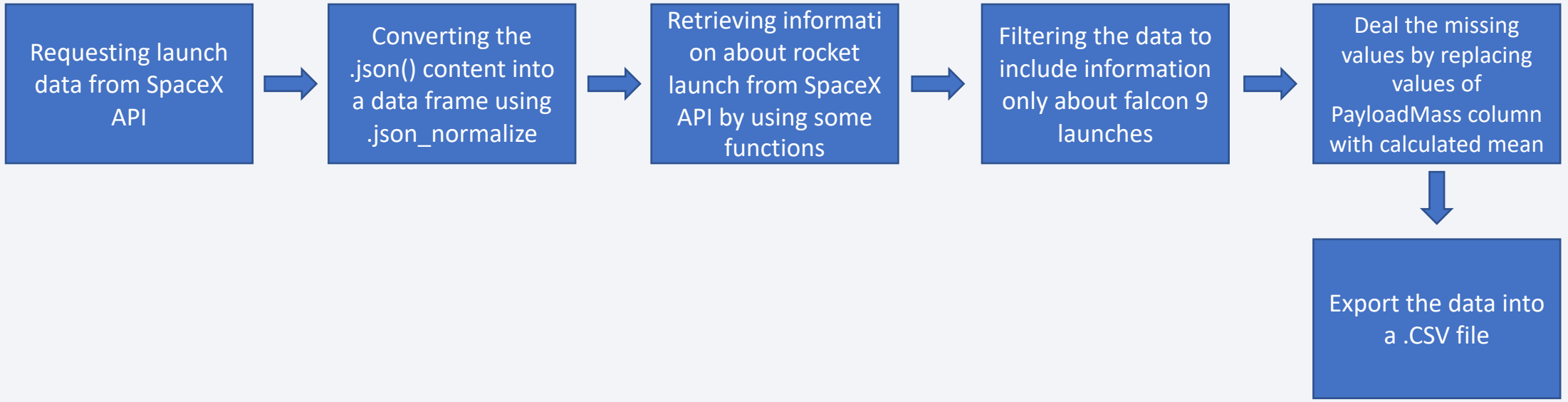
Methodology

- Perform predictive analysis using classification models
    - The collected data were divided into training and test sets. Logistic regression, Classification trees, SVM have been built . Then we used different parameters to find the accuracy of each model.

Data Collection

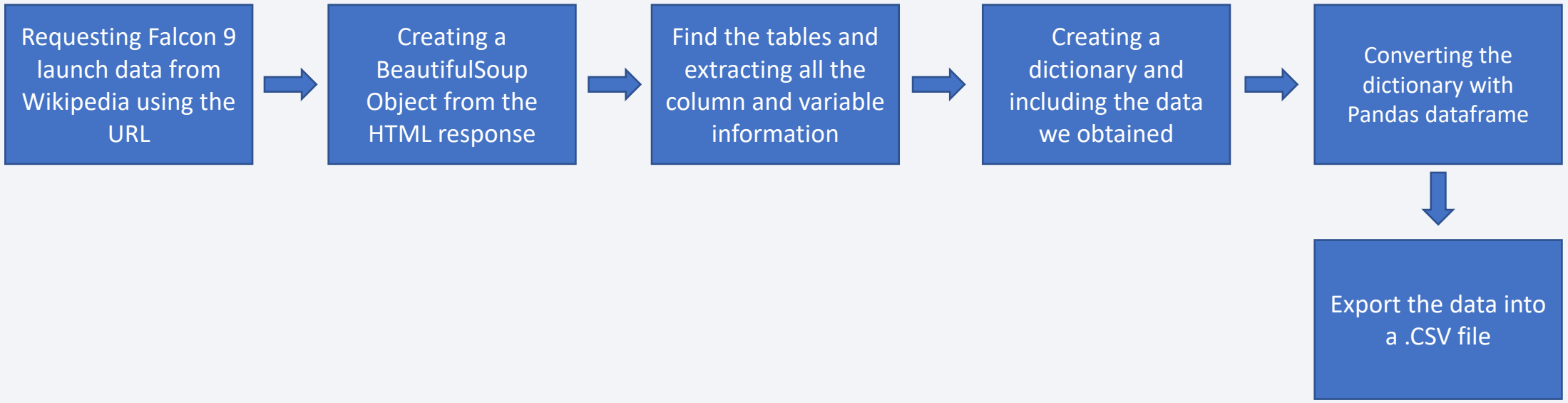
- Data Collection process involved gathering multiple information on variables of interest using API requests from SpaceX REST API and by using Web scraping from SpaceX's Wikipedia.
- We obtained the following columns by using Web scraping from SpaceX's Wikipedia: Payloadmass, Launch site, flight number, Orbit, Launch outcome, version booster, date, time, booster landing.
- Data that were obtained by using SpaceX REST API:
  Payloadmass, flightnumber, Launchsite, outcome, flights, longitude, latitude, reused count, serial, gridfins.

Data Collection – SpaceX API



- GitHub URL: Collecting the Data using SpaceX API

Data Collection - Scraping



- GitHub URL: Data Collection - Web Scraping

Data Wrangling

- We performed some Exploratory Data Analysis to find some patterns in the date and then combined the data in a more accessible way for training supervised models.

- GitHub URL: Data Wrangling

We determine the training labels by performing exploratory data analysis

Calculated the number of launches on each site

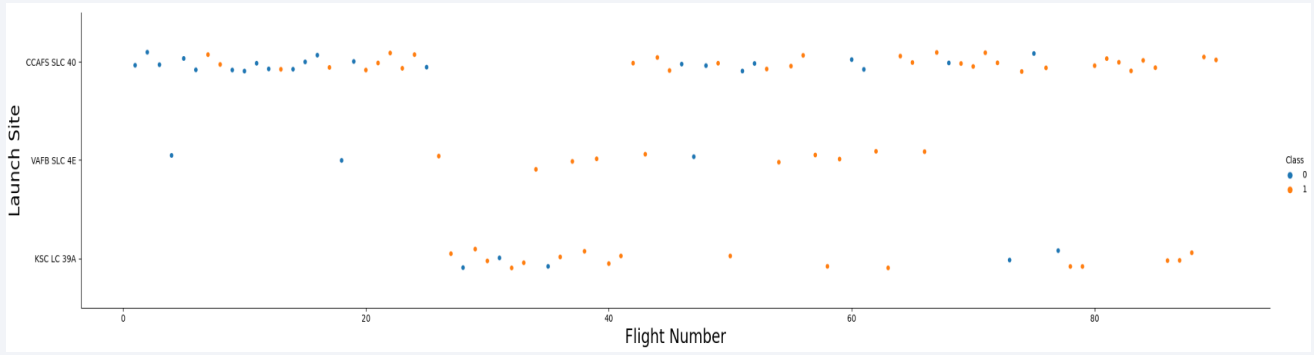
Calculated the number and occurrences of each Orbit

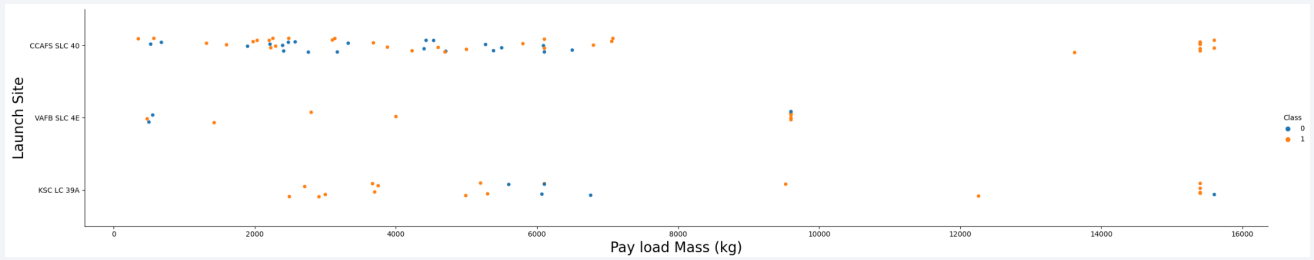
Calculated the number and occurrences of mission outcome per orbit type

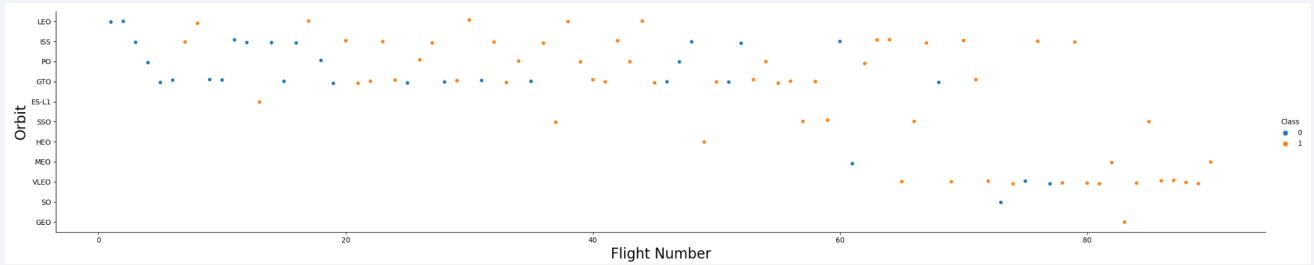
Created a landing outcome label

Exporting the date to .CSV format

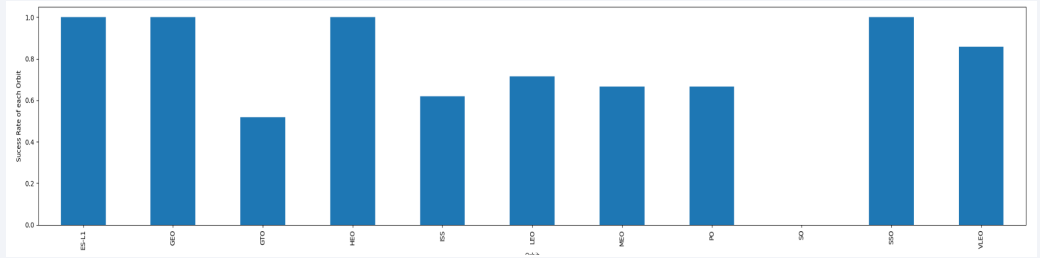
EDA with Data Visualization

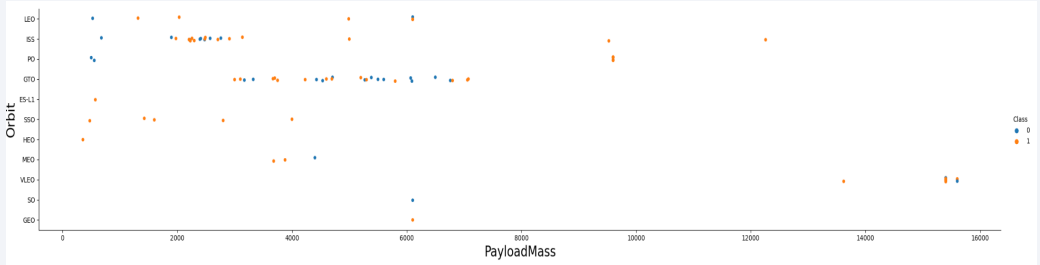






Scatter plot and barplots were used to visualize the relationship between flight number and launch site, payload mass and launch site, Success rate of each orbit and Orbit, Flight number and Orbit, Orbit and Payloadmass





- GitHub URL: EDA with Data Visualization

EDA with SQL

- SQL Queries Performed:

  1. Displaying the names of the unique launch sites in the space mission
  2. Displaying 5 records where launch sites begin with the string 'CCA'
  3. Displaying the total payload mass carried by boosters launched by NASA (CRS)
  4. Displaying average payload mass carried by booster version F9 v1.1
  5. Listing the date when the first succesful landing outcome in ground pad was acheived.
  6. Listing the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000
  7. Listing the total number of successful and failure mission outcomes
  8. Listing the names of the booster_versions which have carried the maximum payload mass by using a subquery
  9. 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.
  10. Ranking the count of successful landing_outcomes between the date 04-06-2010 and 20-03-2017 in descending order

- GitHub URL: EDA with SQL

Build an Interactive Map with Folium

Added marker with circle, pop up labels using its latitude and longitude coordinates where markers indicate launch sites, circles indicate coordinates and lines indicates the distance between two sites.

Colored markers of the launch outcomes with green being the marker of successful launch and red being a failed launch. We have used marker cluster to identify sites with high success rates.

We have also added colored line to show the distance between coordinates

- GitHub URL: Interactive Map with Folium

Build a Dashboard with Plotly Dash

- We have added a Launch site drop down component to enable multiple launch site selection

- Added a call back function to render success pie chart based on the selection
  This pie chart shows the total successful launch count for all sites and success vs failed launch count for individual selection
- We have added a Range slider to select payload

- Added a call back function to show success rate vs payload mass
  A scatterplot was created to show the correlation between the two

- GitHub URL: Dashboard with Plotly Dash

Predictive Analysis (Classification)

1.Created a numpy array from the column class by applying to_numpy()

2. Standardizing the date with StandardScalar and then transforming it

3. We split the data into training and testing set

4. Creating a GridSearchCV object and finding the best parameter from the dictionary

5. Calculated the accuracy of the test data using score()

6. Creating a GridsearchCV object and applying on Decision trees Logreg, SVM, KNN models

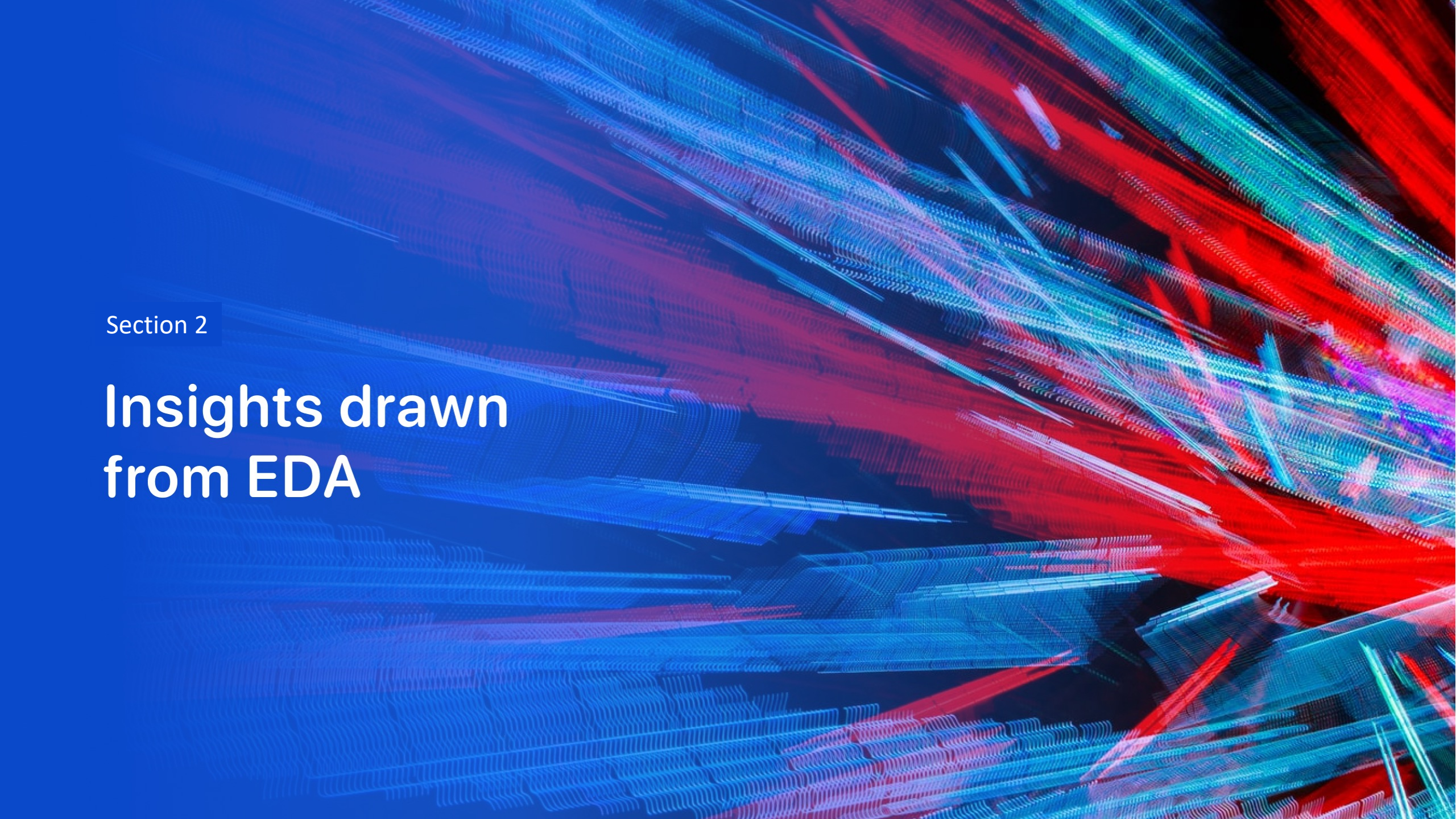
7.Examining the confusion matrix for all models

8. Finding the method that performs the best

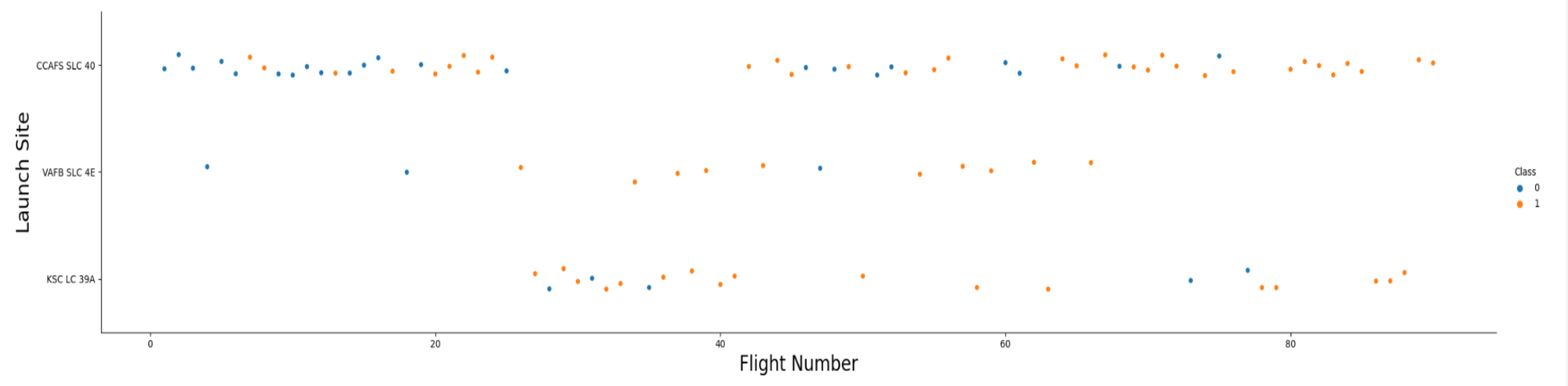
- GitHub URL: Machine Learning prediction

Results

- SpaceX uses 4 different launch sites
- The first successful landing was in 2015
- Low weighted payloads had better performance
- Landing outcomes got better with years
- KSC LC 39A had the most successful launches among all the sites
- After doing the predictive analysis, we can see that Decision tree classifier is the best model to predict successful landing with an accuracy over 87%

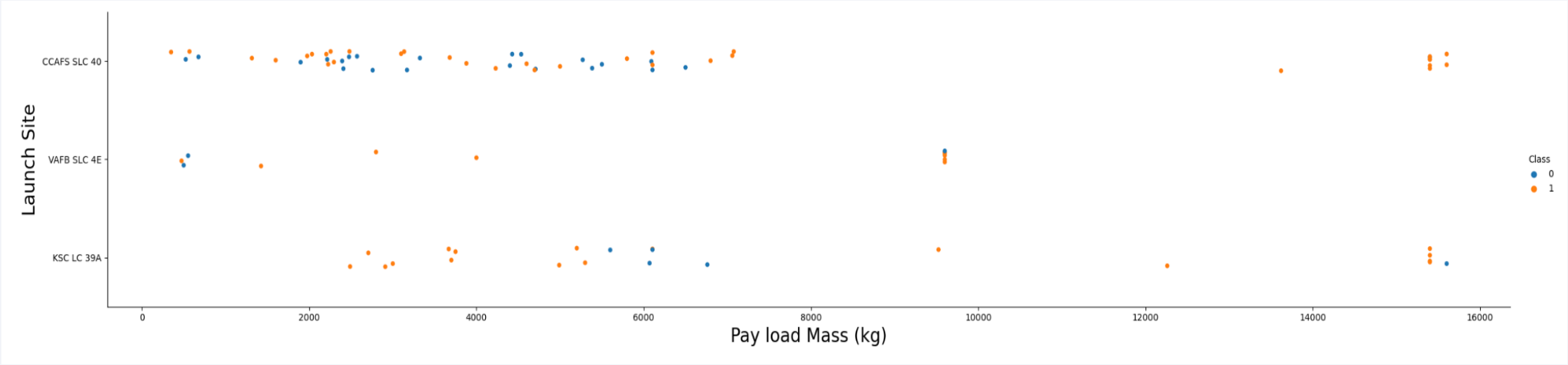

Section 2
Insights drawn from EDA

Flight Number vs. Launch Site



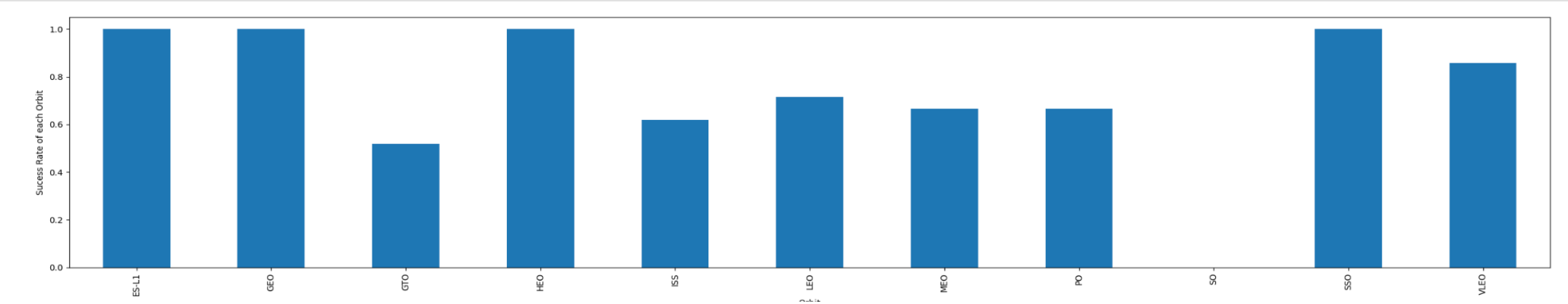
CCAFS SLC 40 was the site with earlier launches and we can see that CCAFS SLC 40 was the best launch site as it is significantly higher than other launch site
We can also see that VAFB SLC 4E was the site with fewer launches

Payload vs. Launch Site



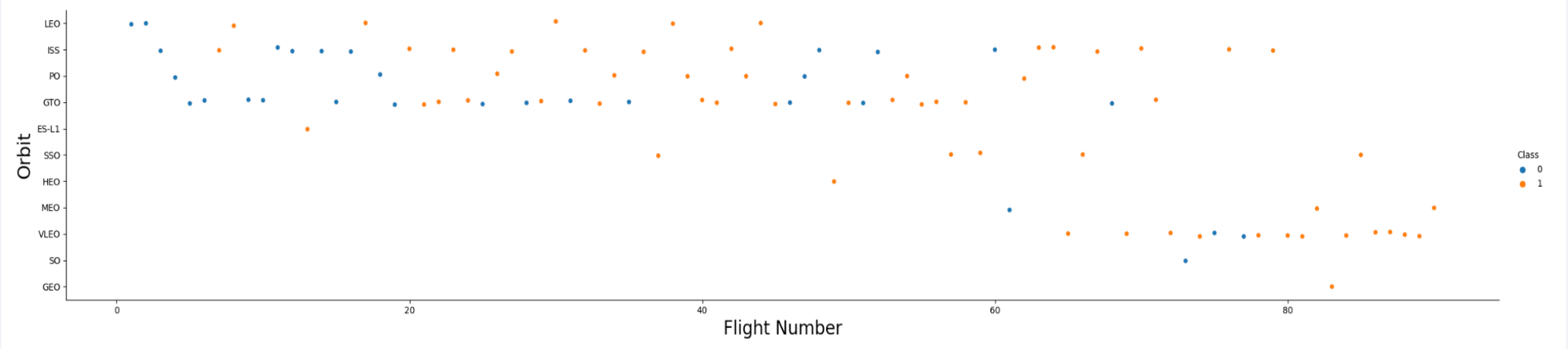
We can see that VAFB SLC 4E has lower payload mass launches compared to other sites

Success Rate vs. Orbit Type



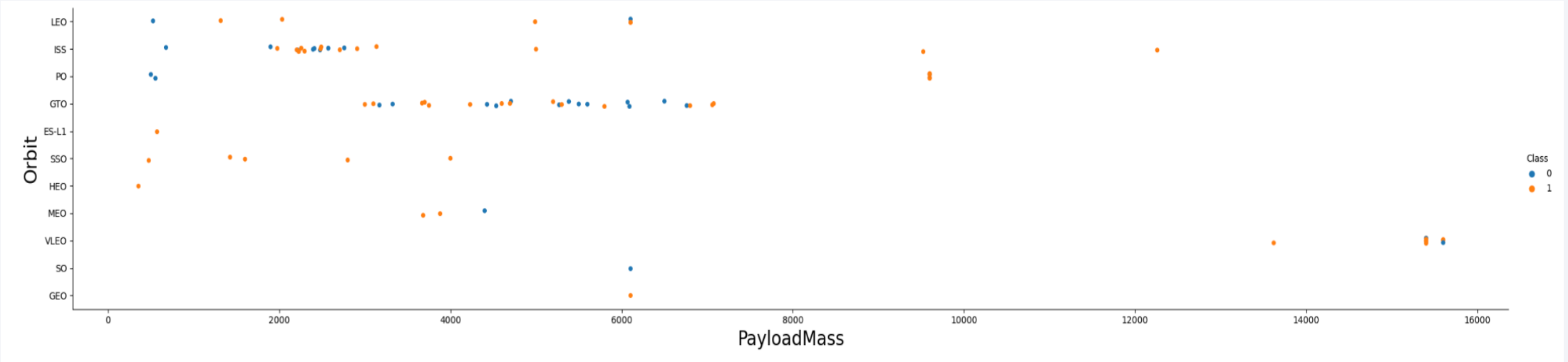
Orbit type ES-L1, GEO, HEO and SSO were the one with highest success rate

Flight Number vs. Orbit Type



We can see that success rate increased over time for all Orbits

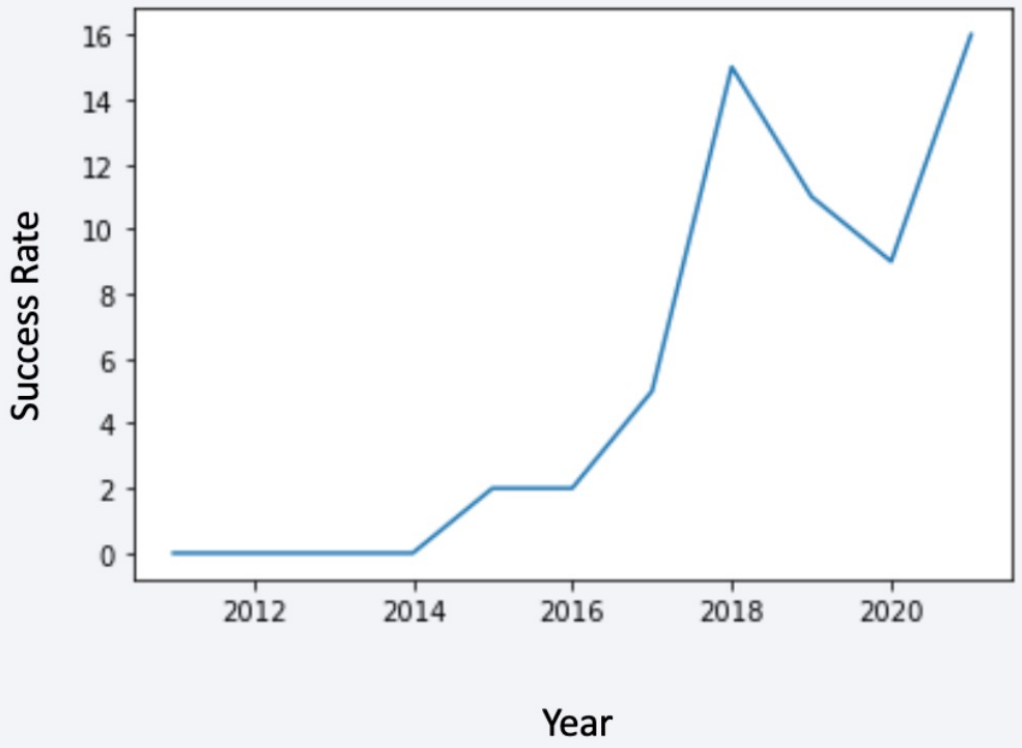
Payload vs. Orbit Type



We can see a strong correlation between ISS and payload mass around 2000

Launch Success Yearly Trend

We can see that launch success rate started to see a significant increase after 2013. This could be due to more advancement in research and development



All Launch Site Names

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

In [29]:
```
%sql select distinct launch_site from SPACEXTBL;
```

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

Out[29]:

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

We got 4 launch sites by selecting unique occurrences from the launch sites

Launch Site Names Begin with 'CCA'

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

In [30]:
```
%sql select * from SPACEXTBL where launch_site like 'CCA%' limit 5;
```

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

Out[30]:

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing _Outcome
04-06-2010	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
08-12-2010	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)
22-05-2012	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
08-10-2012	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
01-03-2013	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

Total Payload Mass

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

In [31]:
```
%sql select sum(payload_mass__kg_) as total_payload_mass from SPACEXTBL where customer = 'NASA (CRS)';
```

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

Out[31]:

total_payload_mass
45596

We calculated it by getting the sum of payload mass where customer code is in NASA (CRS)

Average Payload Mass by F9 v1.1

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

In [32]:
```
%sql select avg(payload_mass__kg_) as average_payload_mass from SPACEXTBL where booster_version like '%F9 v1.1%';
```

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

Out[32]:

average_payload_mass
2534.6666666666665

Displaying the average payload mass carried by booster version F9 v1. 1 which is 2534.66

First Successful Ground Landing Date

```
In [63]: %sql SELECT MIN("DATE") FROM SPACEXTBL WHERE "Landing _Outcome" LIKE '%Success%';

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

Out[63]:

MIN("DATE")
01-05-2017

We filtered the data and found the first successful ground landing date to be 01-5-2017

Successful Drone Ship Landing with Payload between 4000 and 6000

In [64]:

```
%sql SELECT "BOOSTER_VERSION" FROM SPACEXTBL WHERE "LANDING _OUTCOME" = 'Success (drone ship)' \
AND "PAYLOAD_MASS__KG_" > 4000 AND "PAYLOAD_MASS__KG_" < 6000;
```

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

Out[64]:

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

We selected the booster version between payload mass of 4000 kg and 6000 kg that had successful landing

Total Number of Successful and Failure Mission Outcomes

In [65]:
```
%sql SELECT (SELECT COUNT("MISSION_OUTCOME") FROM SPACEXTBL WHERE "MISSION_OUTCOME" LIKE '%Success%') AS SUCCESS, \
(SELECT COUNT("MISSION_OUTCOME") FROM SPACEXTBL WHERE "MISSION_OUTCOME" LIKE '%Failure%') AS FAILURE
```

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

Out[65]:

SUCCESS	FAILURE
100	1

We found the total number of successful outcomes to be around 100 and failed outcomes to be 1

Boosters Carried Maximum Payload

In [46]:
```
%sql select booster_version from SPACEXTBL where payload_mass__kg_ = (select max(payload_mass__kg_) from SPACEXTBL);
```

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

Out[46]:

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

We found the booster versions that carried the maximum payload mass

2015 Launch Records

In [66]:

```
%sql SELECT substr("DATE", 4, 2) AS MONTH, "BOOSTER_VERSION", "LAUNCH_SITE" FROM SPACEXTBL\
WHERE "LANDING _OUTCOME" = 'Failure (drone ship)' and substr("DATE",7,4) = '2015'
```

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

Out[66]:

MONTH	Booster_Version	Launch_Site
01	F9 v1.1 B1012	CCAFS LC-40
04	F9 v1.1 B1015	CCAFS LC-40

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

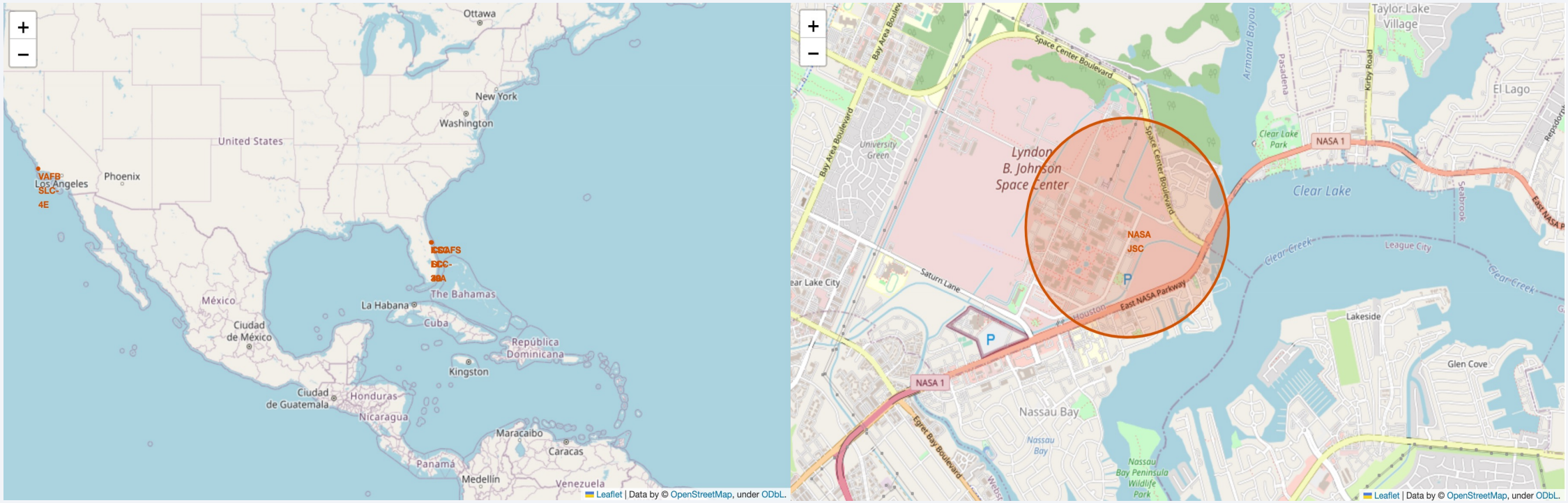
In [50]:

```
%%sql select landing__outcome, count(*) as count_outcomes from SPACEXTBL
      where date between '2010-06-04' and '2017-03-20'
      group by landing__outcome
      order by count_outcomes desc;
```

Ranking all the record between 2010-06-04 and 2017-03-20

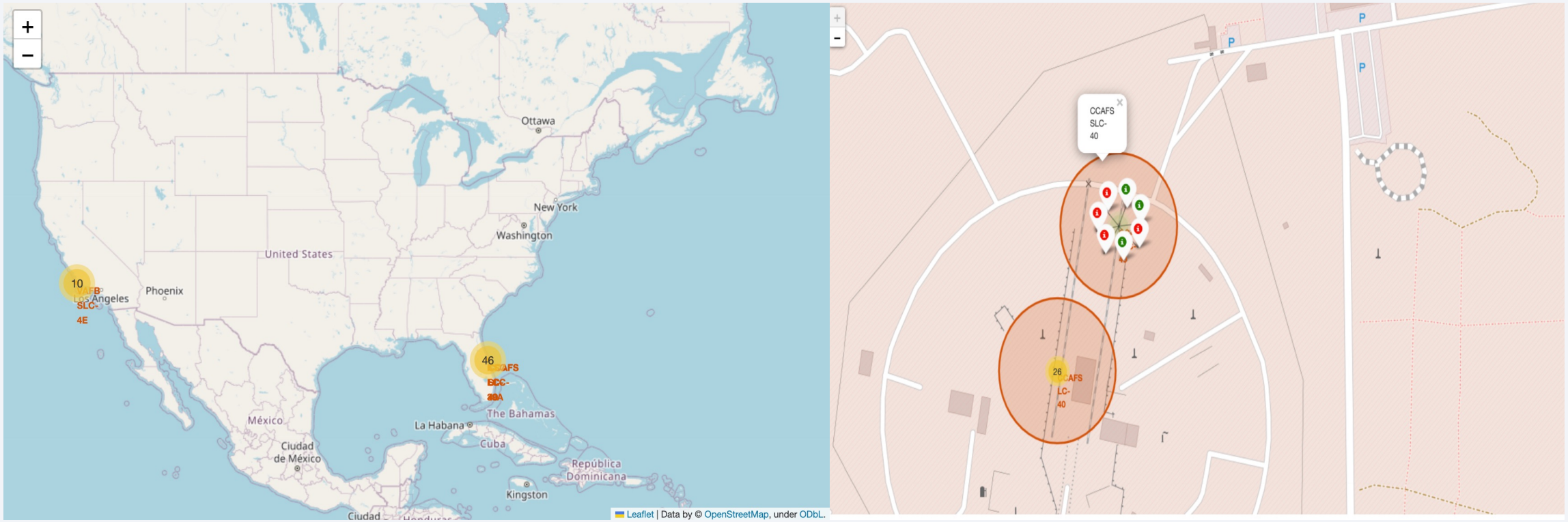

Section 3
Launch Sites
Proximities Analysis

All Launch Sites on a Map



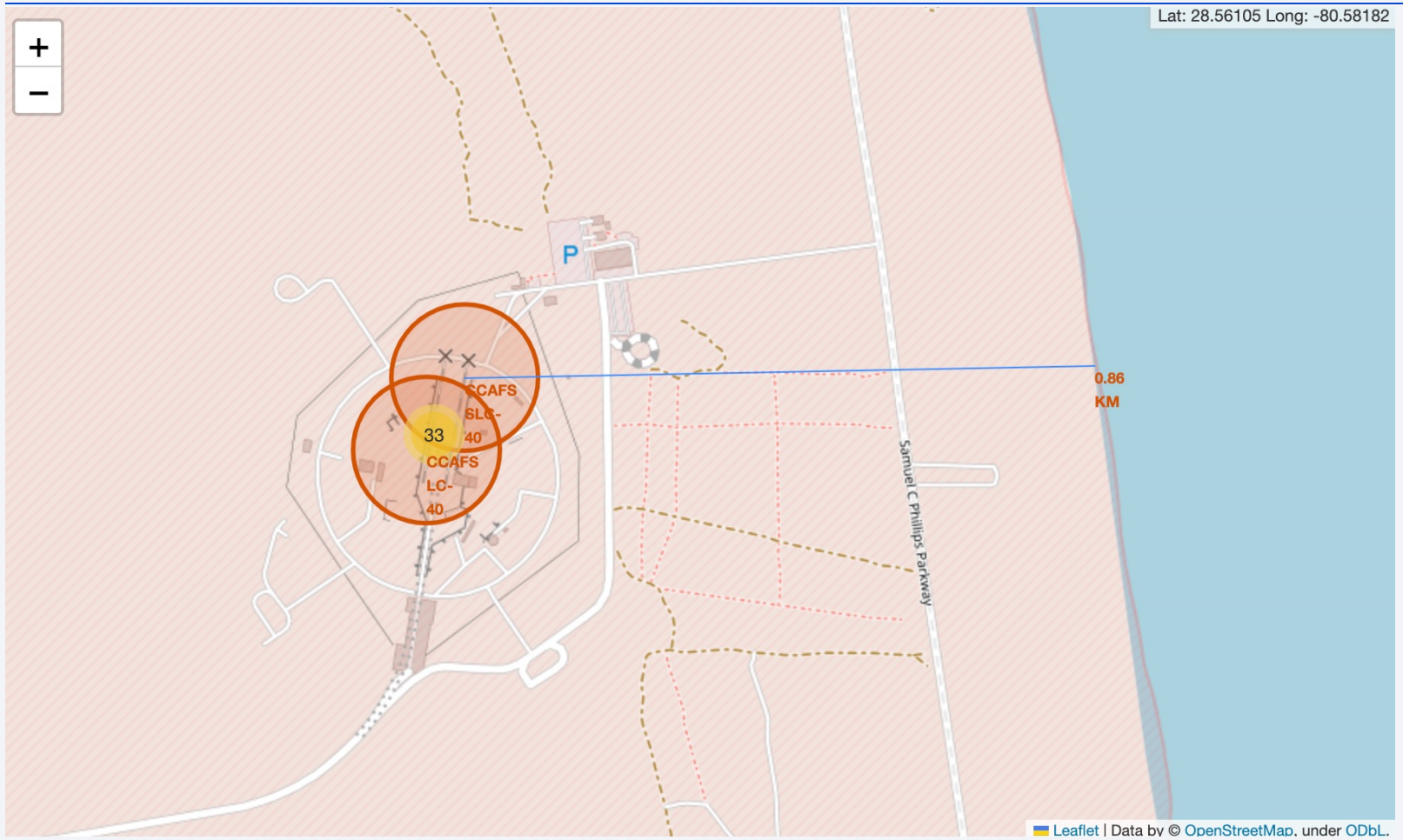
We can see all the launch sites marked on the Map

Success and failed launches for each site on the map

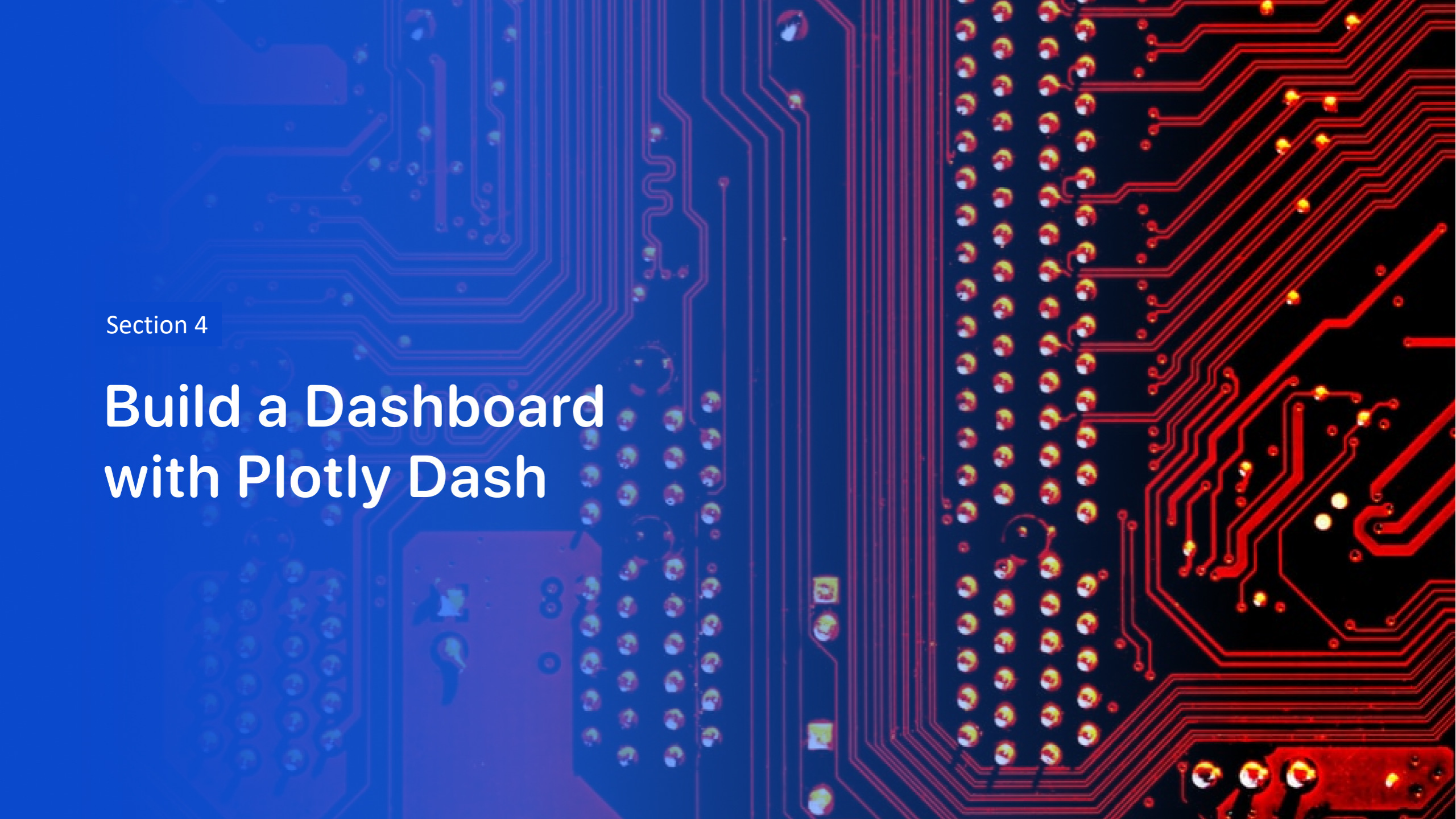


SpaceX launches from different sites and we can see it on the map as a cluster

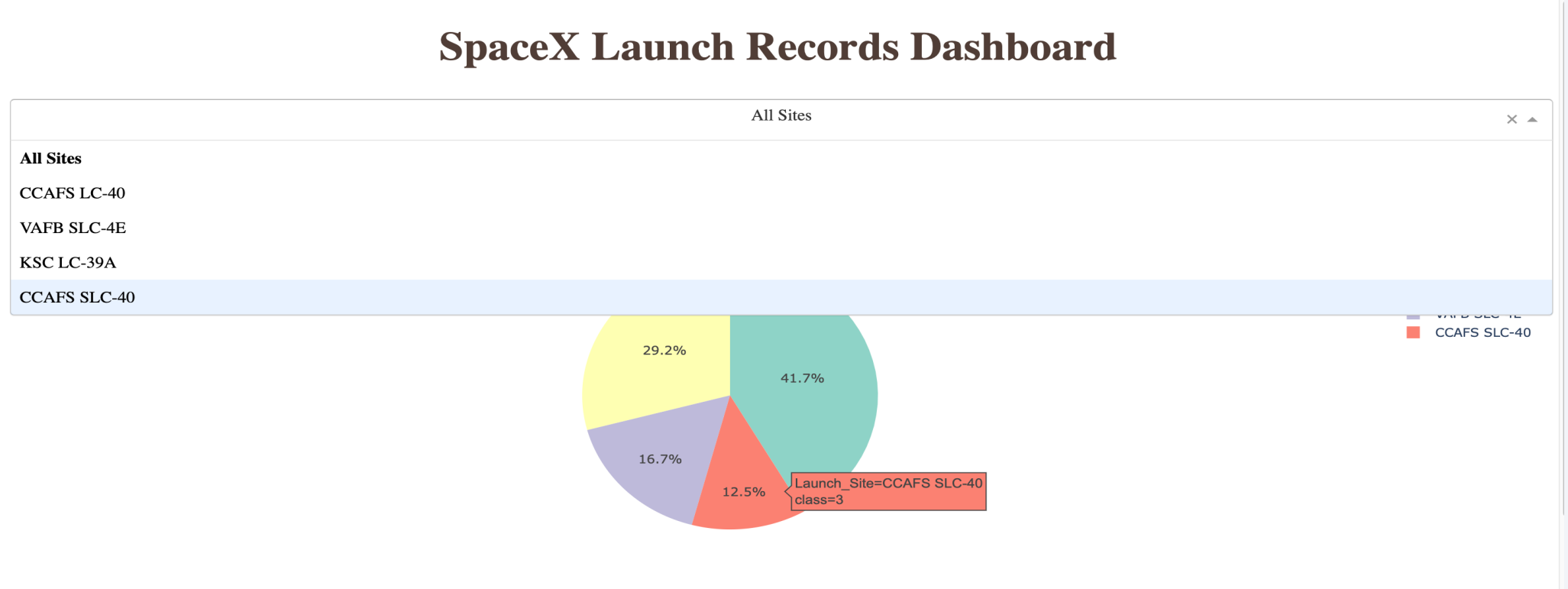
Distance between launch sites and its proximities



We can see the distance between the launch sites and its coastal points that is shown using a polyline

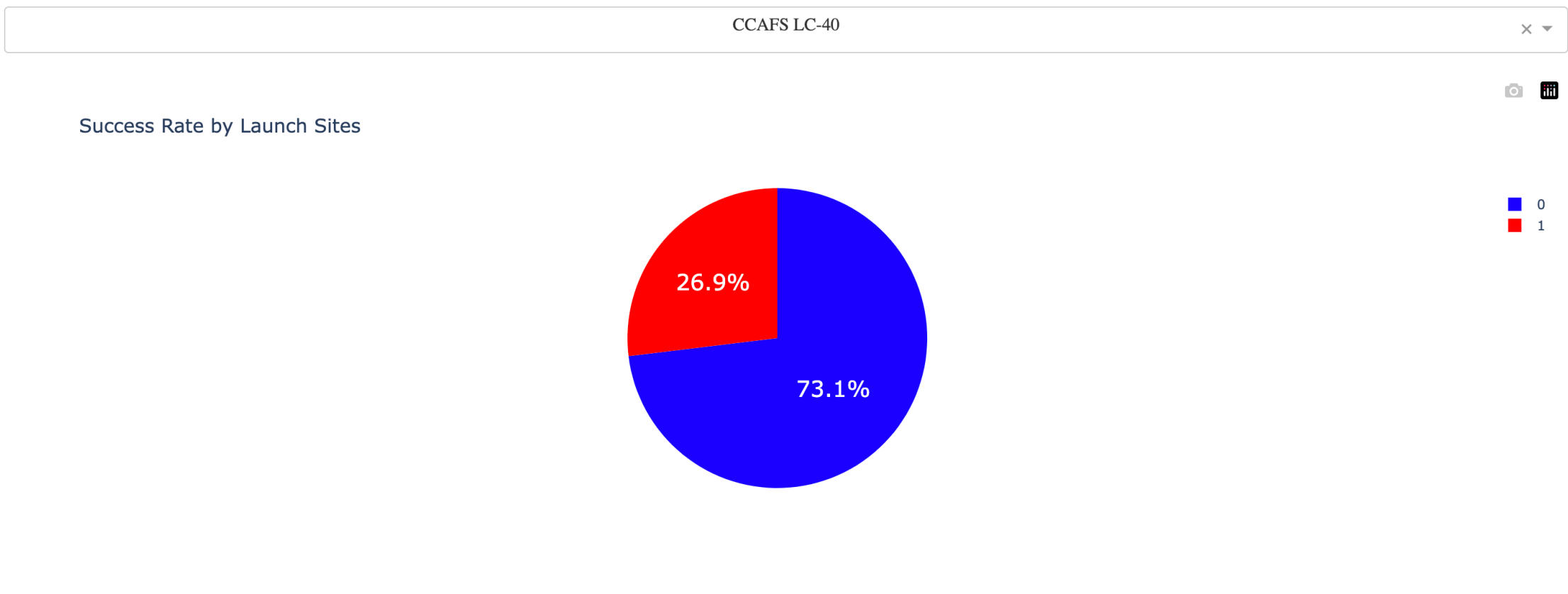

Section 4
Build a Dashboard with Plotly Dash

SpaceX Launch Dashboard



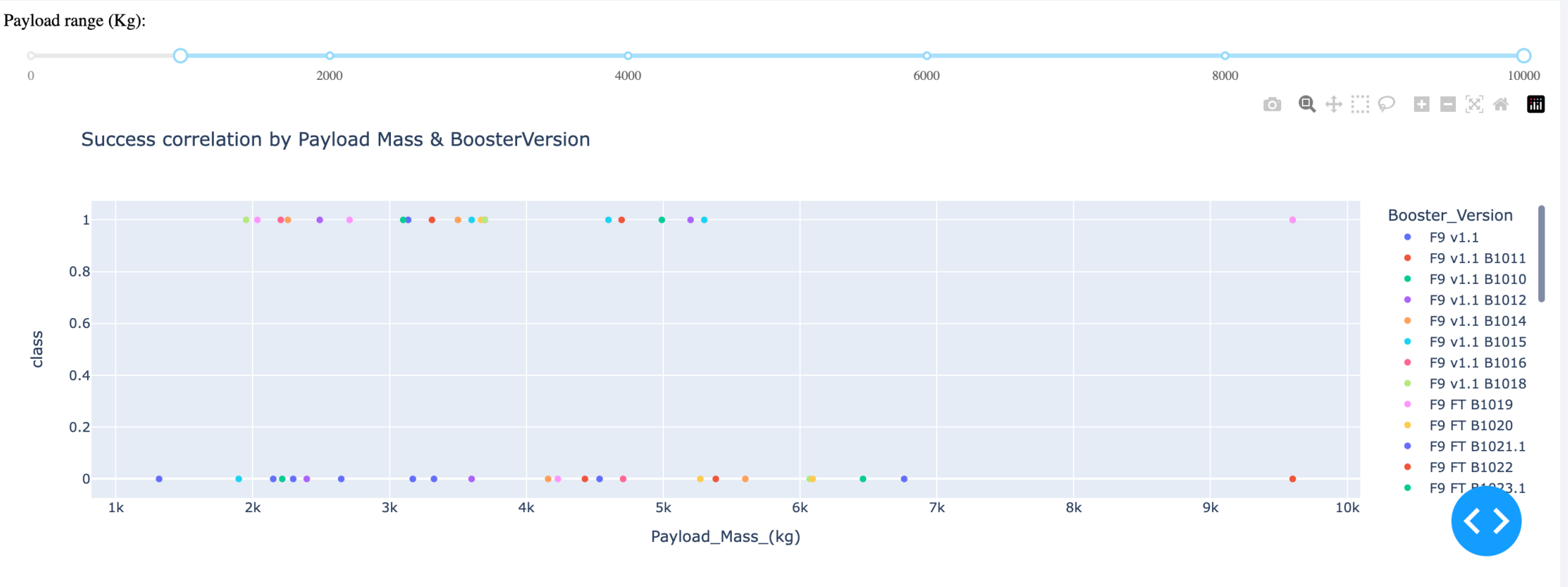
We can see the SpaceX launch record of all sites.

Success Rate by Launch Sites



We can see success rate of 73.1% and failure rate of 26.9% for CCAFS LC-40

Correlation by Payload Mass and BoosterVersion



Section 5

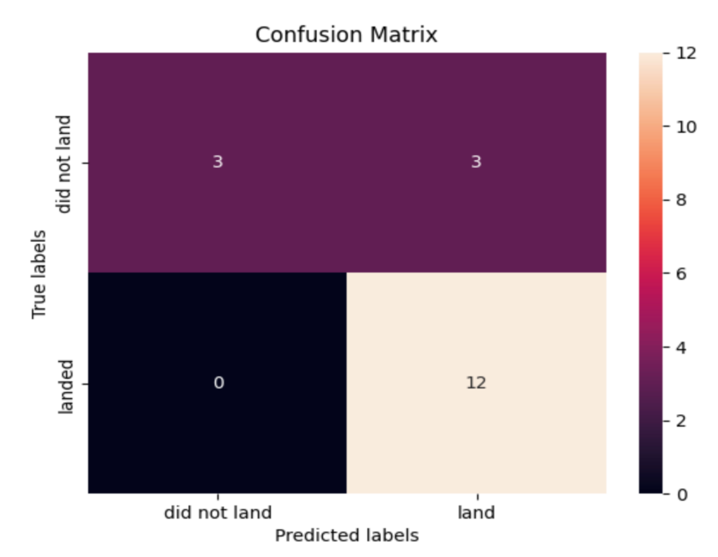
Predictive Analysis (Classification)

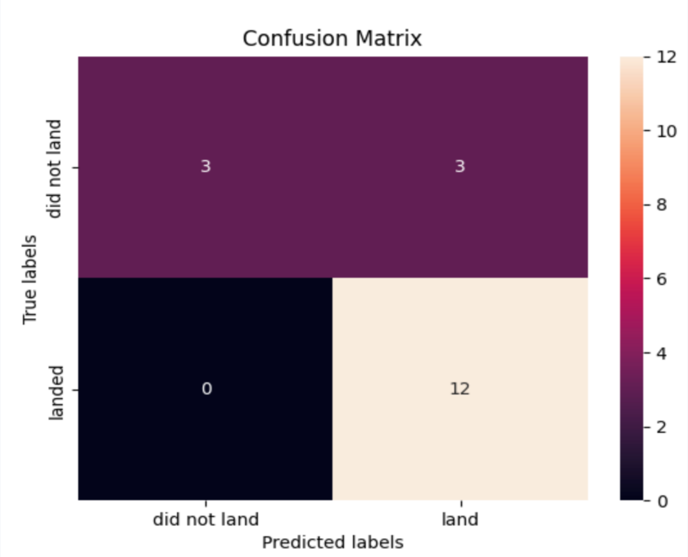
Classification Accuracy

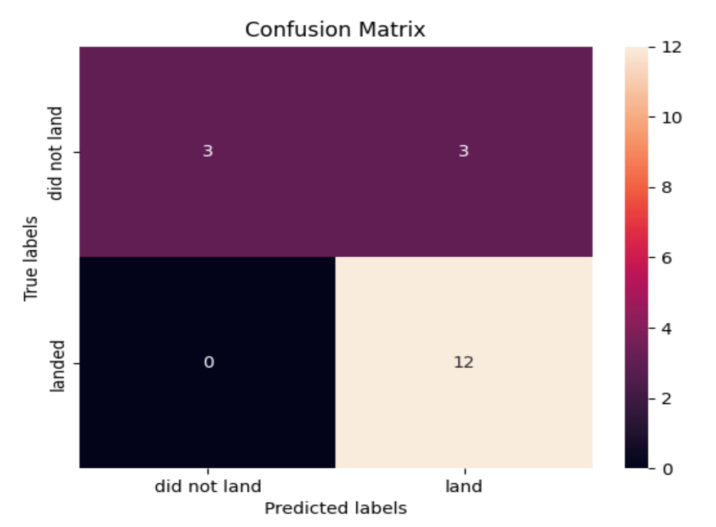
After training 4 different models, each had an accuracy rate of 83.33%

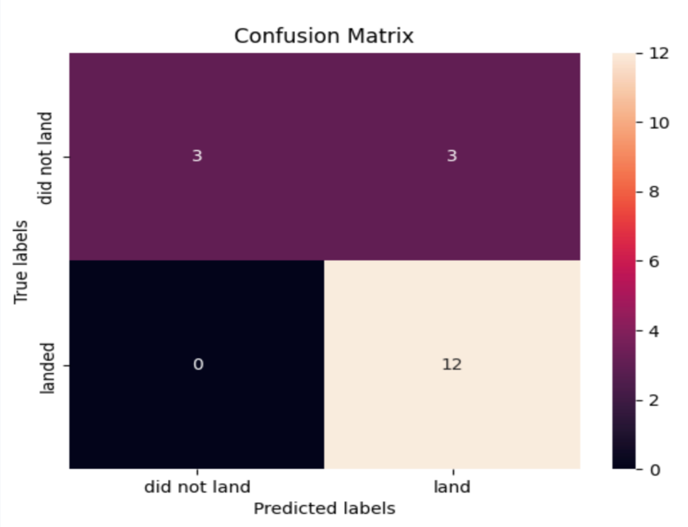
The method that performed the best was the Decision Tree with a score of 87.86%

Confusion Matrix









Conclusions

- Success rate of SpaceX launches increased relatively with time
- KSC LC-39A had the most successful launches
- Among the 4 models, Decision Tree classifier algorithm was the best fit in terms of profitability
- Orbit ES-L1 GEO, SSO has the highest success rate

Appendix

- GitHub Repository URL

Thank you!