IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

<Siddhesh Kotwal>
<4th November 2023>

Outline

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

Executive Summary

- Summary of methodologies

Prediction of the landing of first stage of the Falcon-9 SpaceX rocket begin with the data collection through SpaceX's launch dataset and Wikipedia, then the insights were drawn by visualizing and plotting various features against each other and finally by selecting some predictive variables machine learning techniques were used to train a model which can predict whether or not the first stage of Falcon-9 will land successfully to ensure further cost reduction.

- Summary of all results

while performing analysis on the features of the dataset by Data Visualization, I found out that the payload of the rocket was one of the major predictor for our successful landing, also the success of the rocket was dependent on the orbit of the launch and the Flight number. By training the Machine learning model by various techniques I carried out the accuracy testing of each model and choose the one which was giving the highest accuracy of predicting the successful landing of Falcon-9.

Introduction

- Project background and context

I will predict if the Falcon 9 first stage will land successfully. 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 due to the fact that SpaceX can reuse the first stage.

- Problems you want to find answers

Whether or not the first stage of the Falcon-9 rocket will land successfully, and further the answer to this question will result in finding the answer to the question of the cost of the next launch due to the prevention of the first stage which costs the most ranging the rocket prize to 150 million dollars and without the first stage it costs around 60 million dollars. So, the prediction of the first stage land will reduce the further cost and we can make better choices based on these predictions which will benefit the company.

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Data was collected through the official spaceX website by REST-APIs and by Webscraping from wikipedia page where launch records were mentioned in tables.
- Perform data wrangling
  - Values were counted to check if the data is not biased and then the class column was made to hold the success and failure record.
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Classification Models were made by using scikit-learn Library and various accuracy tests were conducted to select the best one to predict.

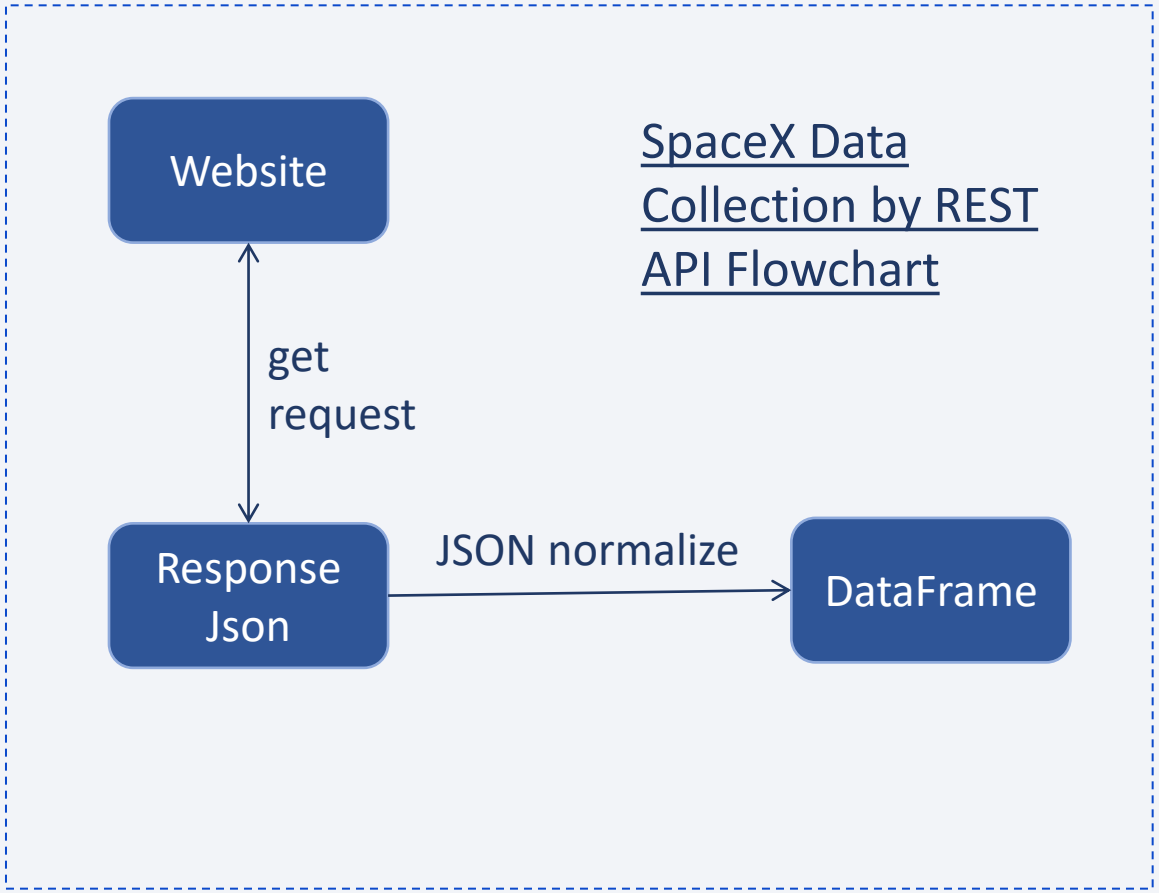
Data Collection

- Describe how data sets were collected.
  - Data sets were collected through SpaceX's website REST-APIs which returned Json package which was then read and converted into DataFrame by using pandas module
  - Data was also collected through the Wikipedia page where SpaceX's Launch information was displayed using the Tables and Which were read using Webscraping by using Beautiful Soup Package Library and finally converted into Pandas DataFrame.
- You need to present your data collection process use key phrases and flowcharts
  - URL was accessed by get method from requests library and the response was collected using the object response which collected the json format data and which was converted into dataframe by using json_normalize
  - Data was scrapped from the wikipedia webpage using beautiful soup object and the requests response was passed to soup object which then converted the raw table data into pandas dataframe.

Data Collection – SpaceX API

- Website was accessed through URL using get request from the requests library and the response was collected as JSON file
- Then converted into pandas data frame by normalizing the JSON data from response using json_normailize from json library

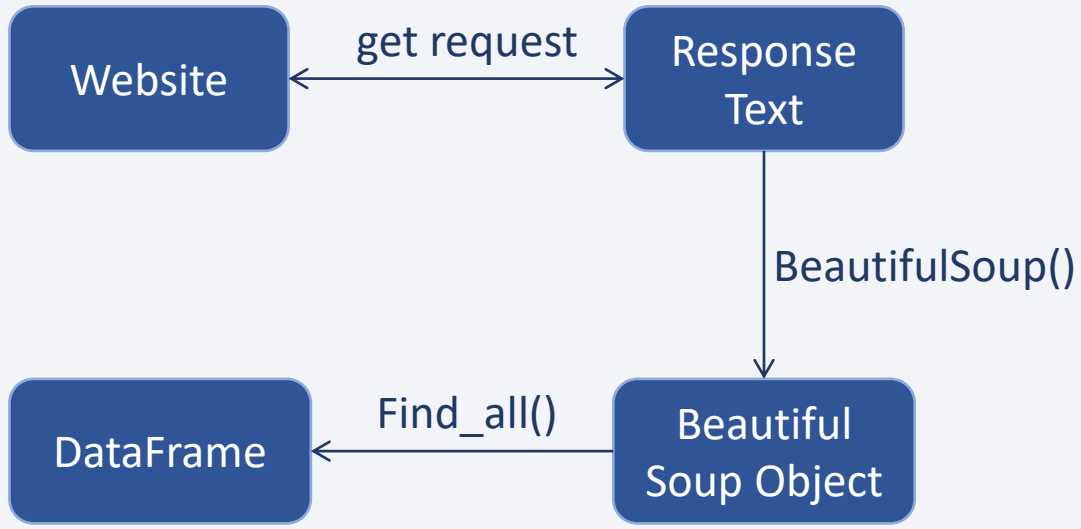
- Data Collection through REST API



Data Collection - Scraping

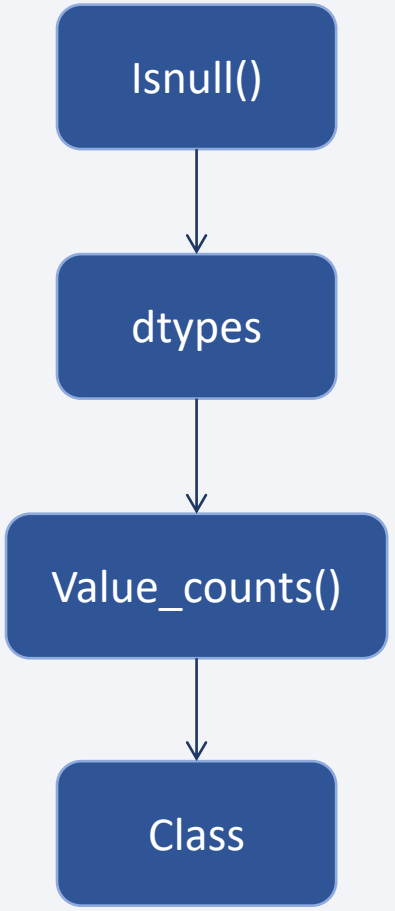
- We will access the website through URL using requests get method and extract the text (HTML Text) from the website.
- Then we will use BeautifulSoup() object to filter out the text by extracting tables and relevant html tags by using find_all() function finally converting it into Pandas Dataframe.
- Data Collection by Webscraping

SpaceX Data Collection by Webscraping Flowchart



Data Wrangling

- First of all, I checked for any null values in the dataset if so, were filled by the mean of the respective column except column of the landing pad which was necessary to know if landing pad was used or not.
- Next the data types were checked according to the data in the columns respectively.
- Then all the categorical values were counted to see if the dataset is biased or not
- Finally, the Class column was added at last representing the success and failure of the landing of rocket
- Data Wrangling



EDA with Data Visualization

1. I used catplot() to plot frequencies of the categorical variables with respect numerical variables.
2. I found out that with increase in flight number the payload mass increased. I also found out that the success rate at launch site VAFB SLC 4E was more as compare to the other launch sites.
3. By using scatterplot() and catplot() I found out that at VAFB SLC 4E launch site there are no rockets launched for heavy payload mass. And heavy payload mass landing were successful at Polar, LEO and ISS orbits.
4. By plotting Bar Chart we found out the success rate of the landing with respect to various orbits of launch.
5. Finally I plotted the success rate with respect to the time line of the company showcasing the increase in success rate with time.

- Data Visualization

EDA with SQL

- Displayed the names of the unique launch sites in the space mission. And Displayed 5 records where launch sites begin with the string 'CCA'
- Displayed the total payload mass carried by boosters launched by NASA (CRS). Displayed average payload mass carried by booster version F9 v1.1. Listed the date when the first succesful landing outcome in ground pad was acheived.
- Listed the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000. And Listed the names of the booster_versions which have carried the maximum payload mass. Using a subquery
- Listed the total number of successful and failure mission outcomes
- Listed the records which will display the month names, failure landing_outcomes in drone ship ,booster versions, launch_site for the months in year 2015.
- Ranked 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.

- Exploratory Data Analysis using SQL

Build an Interactive Map with Folium

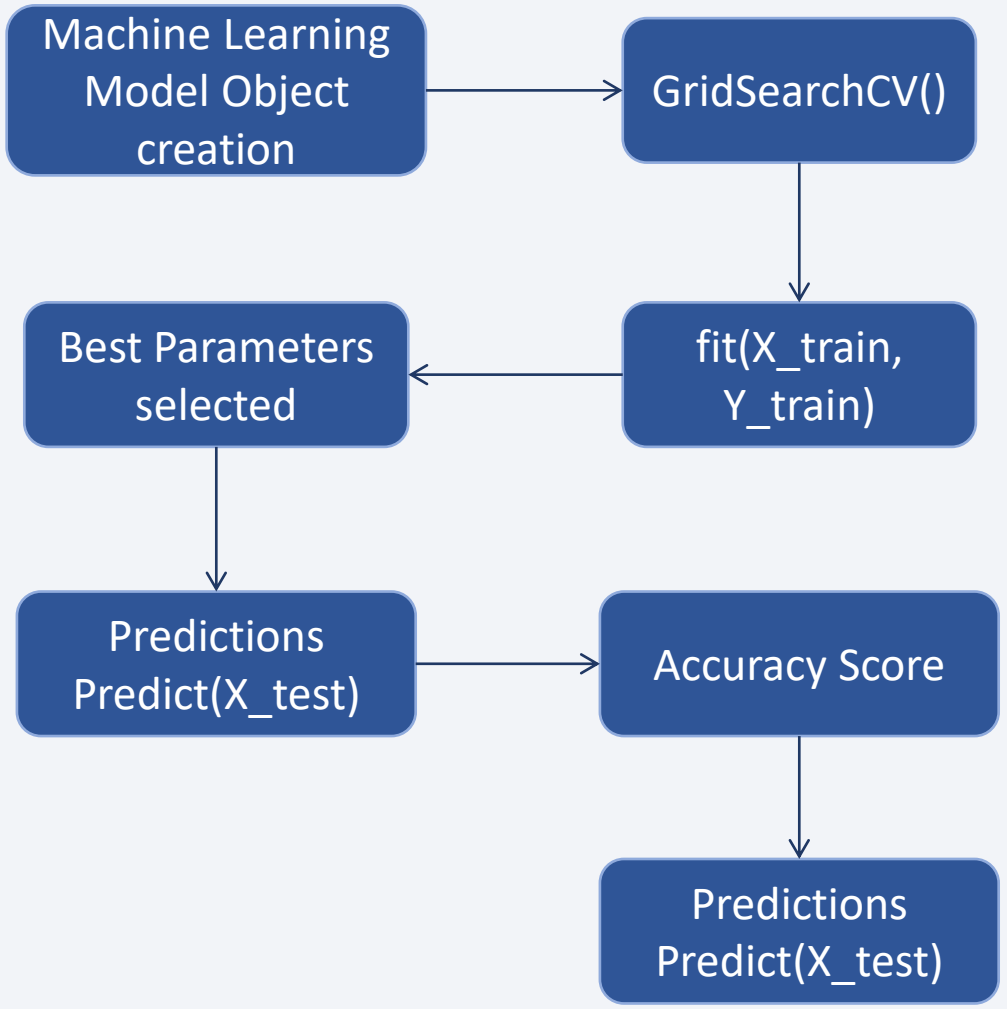
- I have marked various markers on the Map for various purposes, First of all, I have used Circle marker to mark the Launch sites of the Rockets with the icon displaying short name of the launch site, also Popup has been added to those markers displaying complete name of the launch site.
- Further I have used marker cluster representing the total number of launches according to the launch site by adding a child marker to the Map.
- Then separate failure and success launches popups are added with respect to each specific launch site. Then I have added MousePosition to detect the latitude and longitude of the point where the mouse is pointing. All of these above markers were added to visualize the launch sites and its characteristics on the Map.
- Finally, I have added distance Line to the Map representing the distance between nearest Railways, Highways and Coastlines, etc. All of this Lines were added to see if the surrounding can have effect on the failure of the landing of rockets.
- Map representing the Launch Site and its features

Build a Dashboard with Plotly Dash

- In SpaceX Launch Records Dashboard Application I have plotted Pie graphs and scatter plots with interaction of dropdown and slider.
- I have used dropdown menu with options for launch sites which plots Pie chart to find success rate of landings with respect to launch sites and the next plot contains a slider to change the payload range to plot the class vs payload scatter plot.
- Here we can understand that the Highest success rate is on the KSC LC-39A launch site and we understand that with increase in payload size the success rate decreases.
- Dashboard with Plotly Dash

Predictive Analysis (Classification)

- In Predictive Analysis, first of all I have standardized the dataset using StandardScaler(), then I have separated the dataset into training and testing sets.
- Further I have used Logistic Regression, Support Vector Machines, Decision Trees and K Nearest Neighbor Techniques to train the model and test its accuracy with each model to find the best fit.
- To select the best hyper parameters for each of the Technique we are using GridSearchCV() with parameters of each ML Technique.



Predictive Analysis (Classification)

- Process for each model follows as first we create an object of the Machine Learning technique then we use GridSearchCV to obtain the best hyper parameters for that model.
- Further, we train the model with training dataset then measure its accuracy score, and predict the result for testing dataset and plot confusion matrix to plot the accuracy of Model.
- Finally by measuring each Model's accuracy we came to the conclusion of best Machine learning model for this dataset, and it was Decision tree with accuracy greater than 93% (with score of 0.94).
- Predictive Analysis

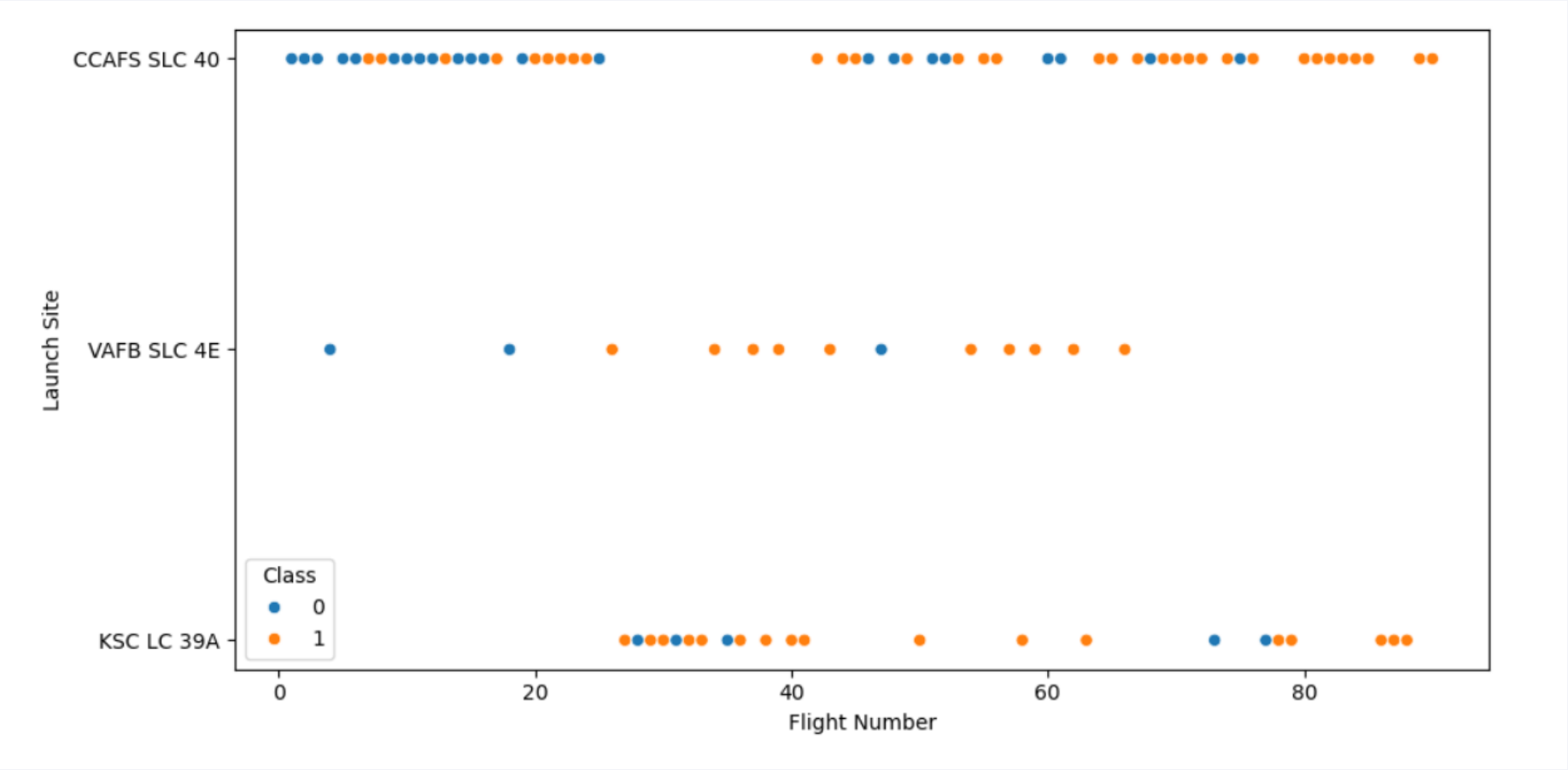
Results

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

Section 2

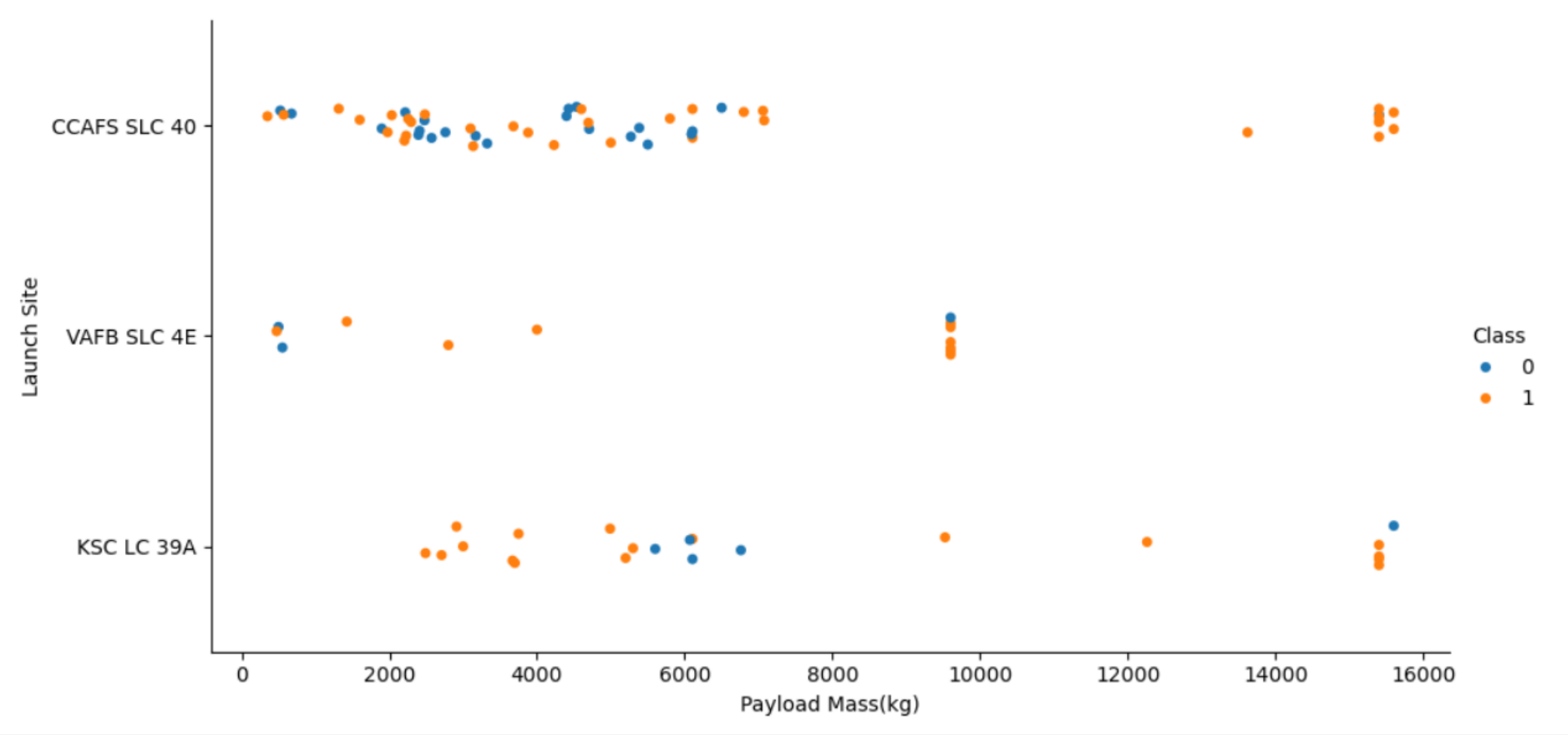
Insights drawn from EDA

Flight Number vs. Launch Site



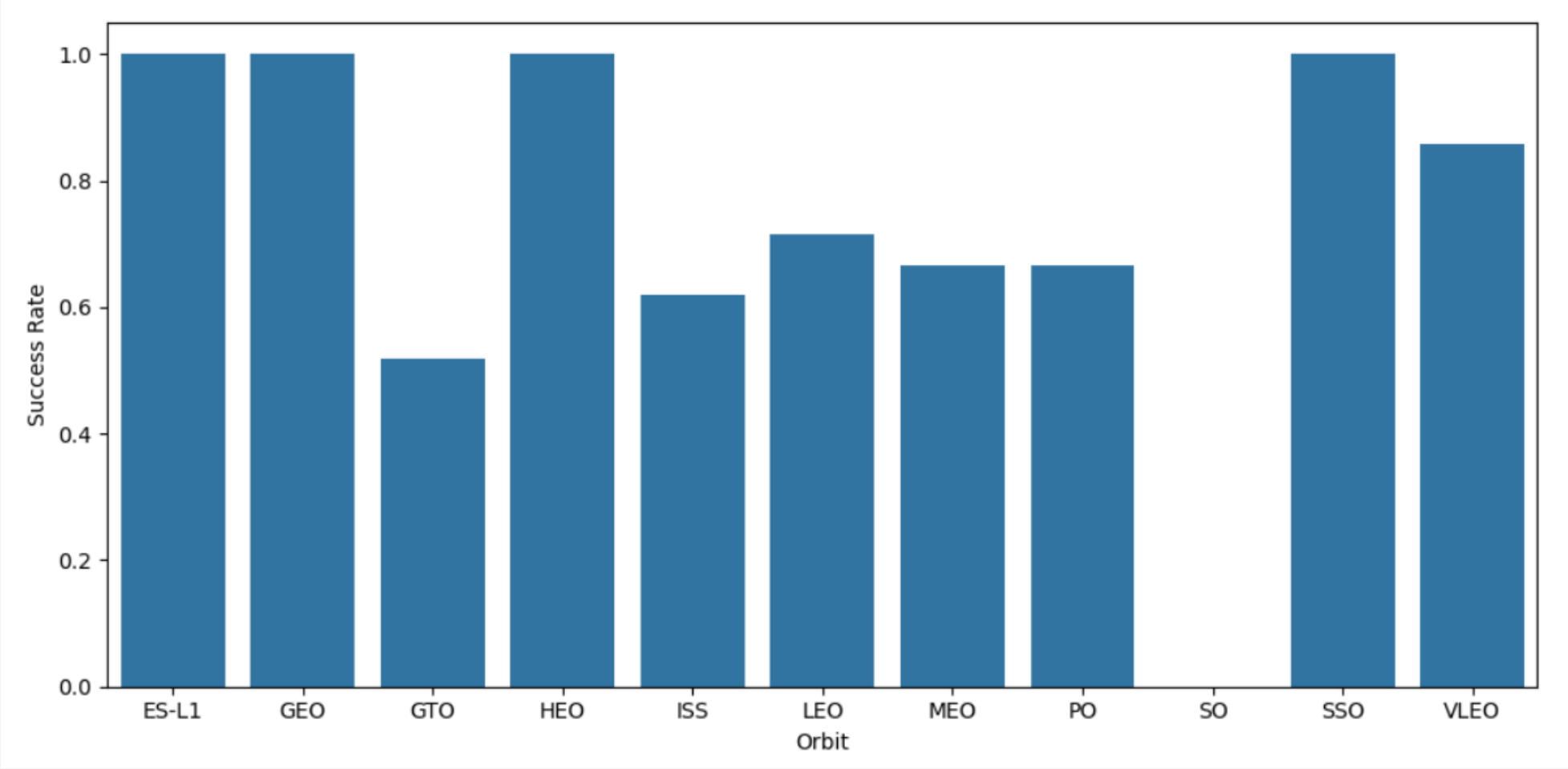
- Here, we can see that success rate is more predictable at launch site VAFB SLC 4E than CCAFS SLC 40 where it is hard to predict the outcome.

Payload vs. Launch Site



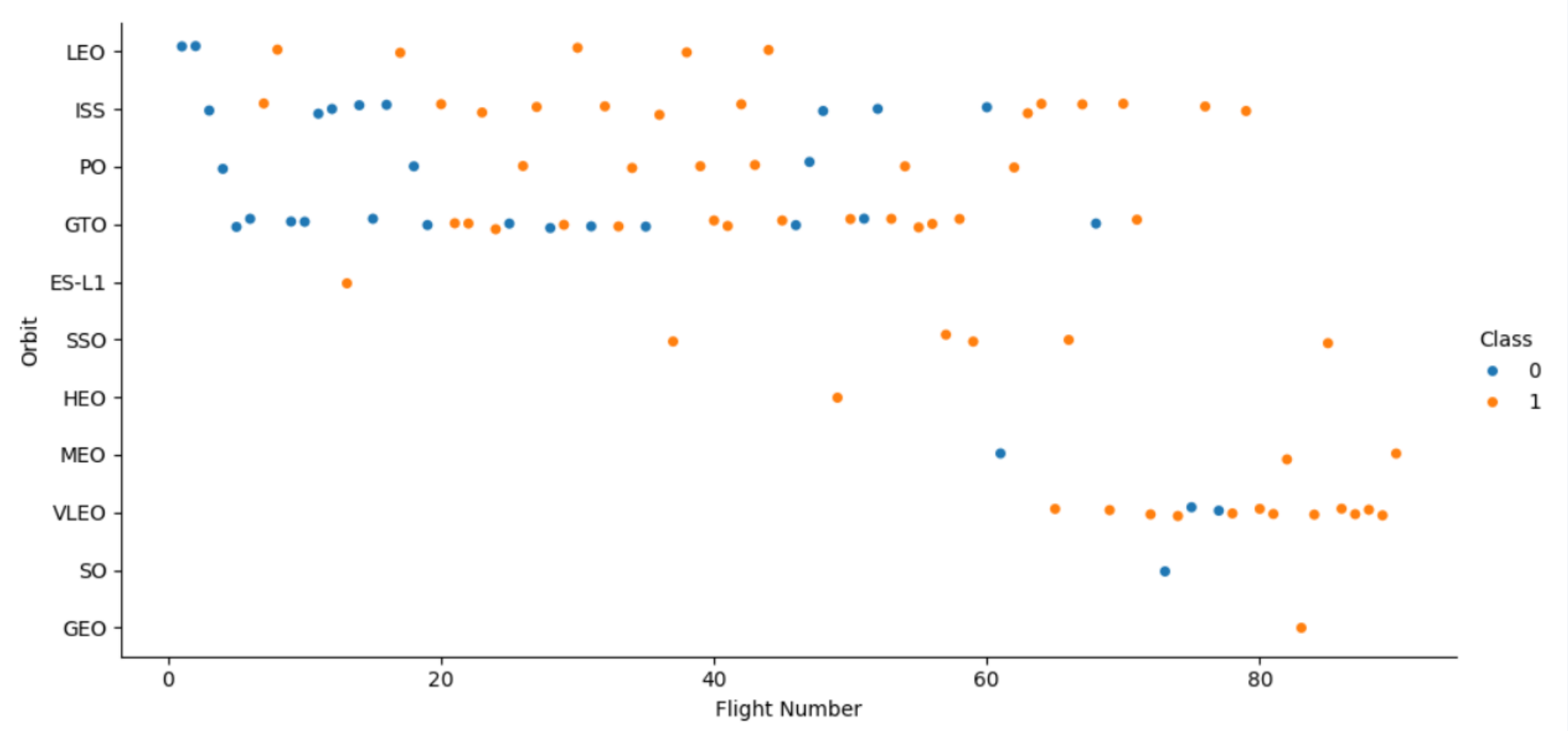
- Here we get the insights that heavy payload launches are not considered at launch site VAFB SLC 4E and are also less on other launch sites

Success Rate vs. Orbit Type



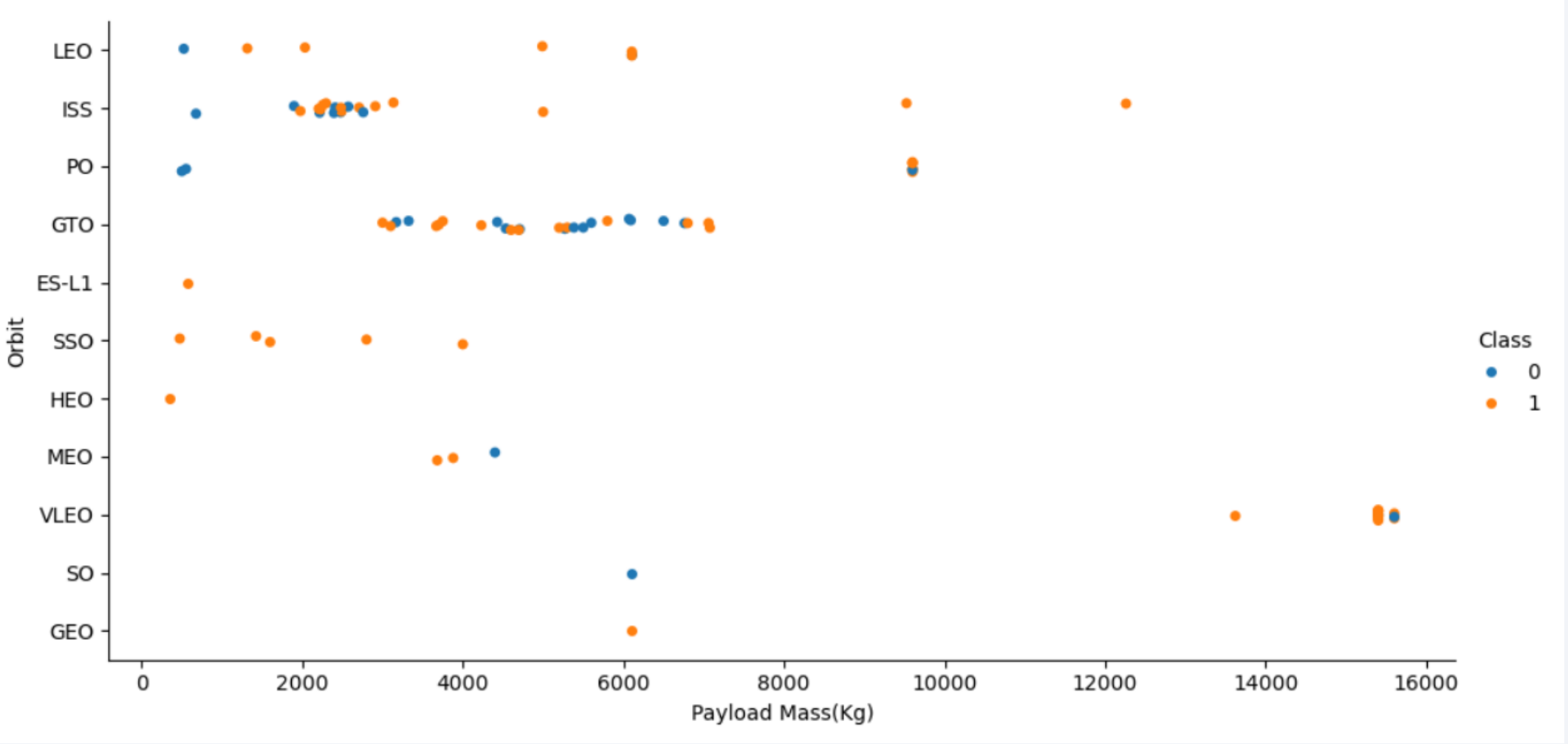
- We can see the less success rate of landing in the orbits GTO, SO and ISS and High success rate in the orbits SSO, ES-L1, GEO and HEO.

Flight Number vs. Orbit Type



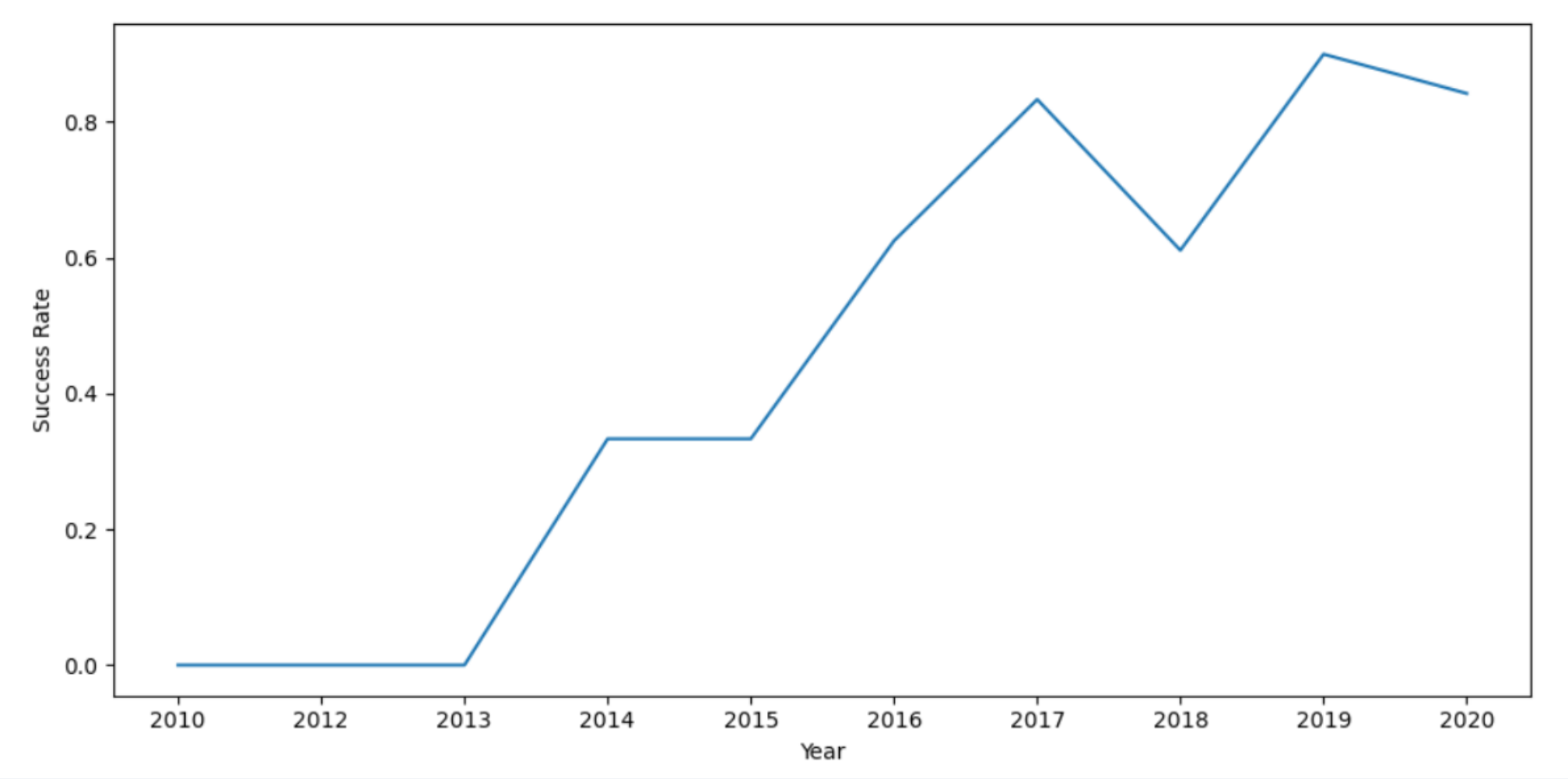
- We can see there were no launches in many of the orbits in early stages of the company. Also we can see there is high chances of success in the LEO orbit and its unpredictable in GTO orbit

Payload vs. Orbit Type



Here we see with heavy payloads the successful landing or positive landing rate are more for Polar, LEO and ISS. However for GTO we cannot distinguish this well as both positive landing rate and negative landing rates are present

Launch Success Yearly Trend



With increase in timeline the success rate of landings of first stage of rockets increases from 2013 to 2020

All Launch Site Names

- Display the names of the unique sites in the space mission
- Selecting Launch sites by Distinct gives unique values in that column from the table. And the result contains CCAFS LC-40, VAFB SLC-4E, KSC LC-39A and CCAFS SLC-40.

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

```
[9]: %sql SELECT DISTINCT(Launch_Site) FROM SPACEXTABLE
```

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

Done.

```
[9]: ,,,,,,,,,,,,,,,,,,,,
```

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

Launch Site Names Begin with 'CCA'

- Find 5 records where launch sites begin with `CCA`
- Queried complete records with the condition being launch site starts with CCA using Like keyword and limit upto 5 records

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

```
[11]: %sql SELECT * FROM SPACEXTABLE WHERE Launch_Site LIKE 'CCA%' LIMIT 5;
```

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

[11]: ''

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

Total Payload Mass

- Calculate the total payload carried by boosters from NASA
- Used sum() function to calculate total in the column payload_mass_kg while the condition being the customer is NASA used like keyword because of uncertainty of the customer word starting or ending with some other word but contains NASA. So, the total payload mass is 107010 Kgs.

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

```
[16]: %sql SELECT SUM(PAYLOAD_MASS__KG_) as Total_Payload_Mass_in_Kg FROM SPACEXTABLE WHERE Customer LIKE '%NASA%';
 * sqlite:///my_data1.db
Done.
[16]: ,,,,,,,,,,
```

Total_Payload_Mass_in_Kg
107010

Average Payload Mass by F9 v1.1

- Calculate the average payload mass carried by booster version F9 v1.1
- Used avg() function to calculate average and round() to round of the average of the payload mass in kg with condition being the booster version is F9 v1.1. So, the average payload mass is 2535.0 Kg.

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

```
[18]: %sql SELECT ROUND(AVG(PAYLOAD_MASS__KG_)) as Average_Payload_Mass FROM SPACEXTABLE WHERE Booster_Version LIKE '%F9 v1.1%';
 * sqlite:///my_data1.db
Done.
[18]: ,,,,,,,,,,
```

Average_Payload_Mass
2535.0

First Successful Ground Landing Date

- Find the dates of the first successful landing outcome on ground pad
- Used min() function to find the lowest date while the condition being landing outcome to be something around ground pad and mission outcome being success. So, the first landing on ground occurred on 2015-12-22.

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

Hint:Use min function

[16]:
```
%%sql

SELECT MIN(Date) as First_Successful_Ground_Pad_Landing FROM SPACEXTABLE WHERE Landing_Outcome
    LIKE '%Ground pad%' AND Mission_Outcome = 'Success';
```

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

[16]:

First_Successful_Ground_Pad_Landing
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

- List the names of boosters which have successfully landed on drone ship and had payload mass greater than 4000 but less than 6000
- This result shows that the particular booster versions which follows the conditions are all F9 FT B… with some variations which are successful on landing on drone ship

List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000

[17]:
```
%%sql

SELECT Booster_Version FROM SPACEXTABLE WHERE Landing_Outcome LIKE '%drone ship%' AND
    Mission_Outcome = 'Success' AND PAYLOAD_MASS__KG_ BETWEEN 4000 AND 6000;
```

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

[17]:

Booster_Version
F9 FT B1020
F9 FT B1022
F9 FT B1026
F9 FT B1021.2
F9 FT B1031.2

Total Number of Successful and Failure Mission Outcomes

- Calculate the total number of successful and failure mission outcomes
- In this query result we can see that there is only one failure in rocket landings and 100 successful landings. I have used cases for mission outcome for success and failure and finally summed up both, to get the total success and failure landings.

List the total number of successful and failure mission outcomes

[15]:
```
%%sql
SELECT
    SUM(CASE WHEN Mission_Outcome LIKE '%success%' THEN 1 ELSE 0 END) as Success,
    SUM(CASE WHEN Mission_Outcome LIKE '%failure%' THEN 1 ELSE 0 END) as Failure
FROM SPACEXTABLE;
```

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

[15]:

Success	Failure
100	1

Boosters Carried Maximum Payload

- List the names of the booster which have carried the maximum payload mass
- Used subquery to get the boosters which were used to carry heavy payloads.

List the names of the booster_versions which have carried the maximum payload mass. Use a subquery

[50]:
```
%%sql

SELECT Booster_Version FROM SPACEXTABLE WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTABLE);
```

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

[50]:

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

2015 Launch Records

- List the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015

List the records which will display the month names, failure landing_outcomes in drone ship ,booster versions, launch_site for the months in year 2015.

Note: SQLLite does not support monthnames. So you need to use substr(Date, 6,2) as month to get the months and substr(Date,0,5)='2015' for year.

[59]:
```
%%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'
        ELSE 'Unknown'
    END as Month,
    (SELECT Landing_Outcome FROM SPACEXTABLE WHERE Landing_Outcome LIKE '%failure (drone ship)%') as Failure_Landing_outcomes,
    Booster_Version,
    Launch_Site
FROM SPACEXTABLE
WHERE substr(Date, 0, 5) = '2015';
```

2015 Launch Records

- Result of the previous query
- This result shows the record containing the Month, Booster version and Launch sites where drone ship landings failed in the year 2015.

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

[59]:

Month	Failure_Landing_outcomes	Booster_Version	Launch_Site
October	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
November	Failure (drone ship)	F9 v1.1 B1013	CCAFS LC-40
February	Failure (drone ship)	F9 v1.1 B1014	CCAFS LC-40
April	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40
April	Failure (drone ship)	F9 v1.1 B1016	CCAFS LC-40
June	Failure (drone ship)	F9 v1.1 B1018	CCAFS LC-40
December	Failure (drone ship)	F9 FT B1019	CCAFS LC-40

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

- Rank 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
- Here we can have ranked the landing outcomes and we can see ground pad landings are mostly successful while drone ship landing's success to failure ratio is zero.

Rank 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.

[19]:
```
%%sql

SELECT Landing_Outcome, COUNT(Landing_Outcome) as outcome_count FROM SPACEXTABLE WHERE
    Date BETWEEN '2010-06-04' AND '2017-03-20' GROUP BY Landing_Outcome ORDER BY outcome_count DESC;
```

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

[19]:

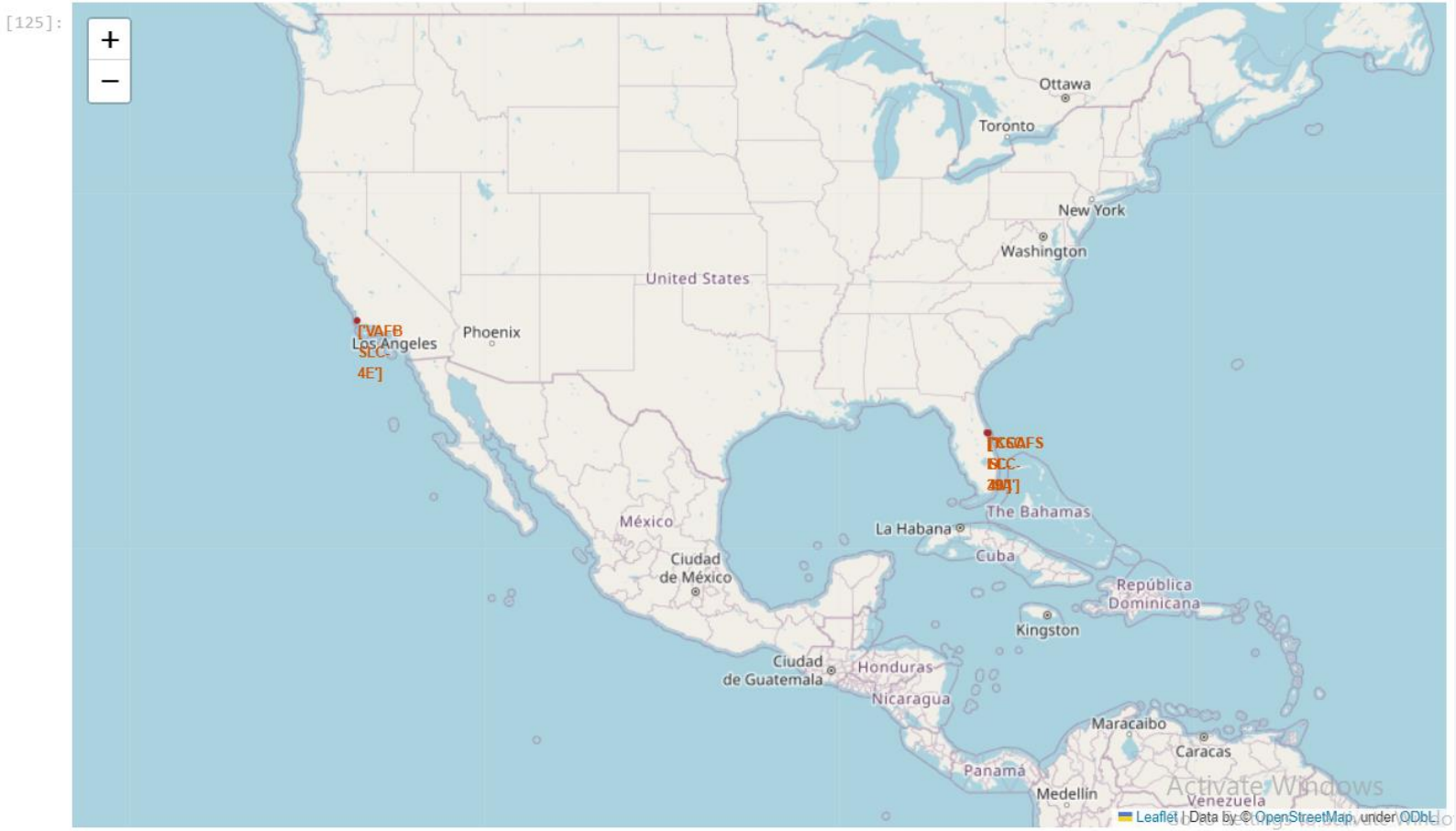
Landing_Outcome	outcome_count
No attempt	10
Success (ground pad)	5
Success (drone ship)	5
Failure (drone ship)	5
Controlled (ocean)	3
Uncontrolled (ocean)	2
Precluded (drone ship)	1
Failure (parachute)	1

Section 3

Launch Sites Proximities Analysis

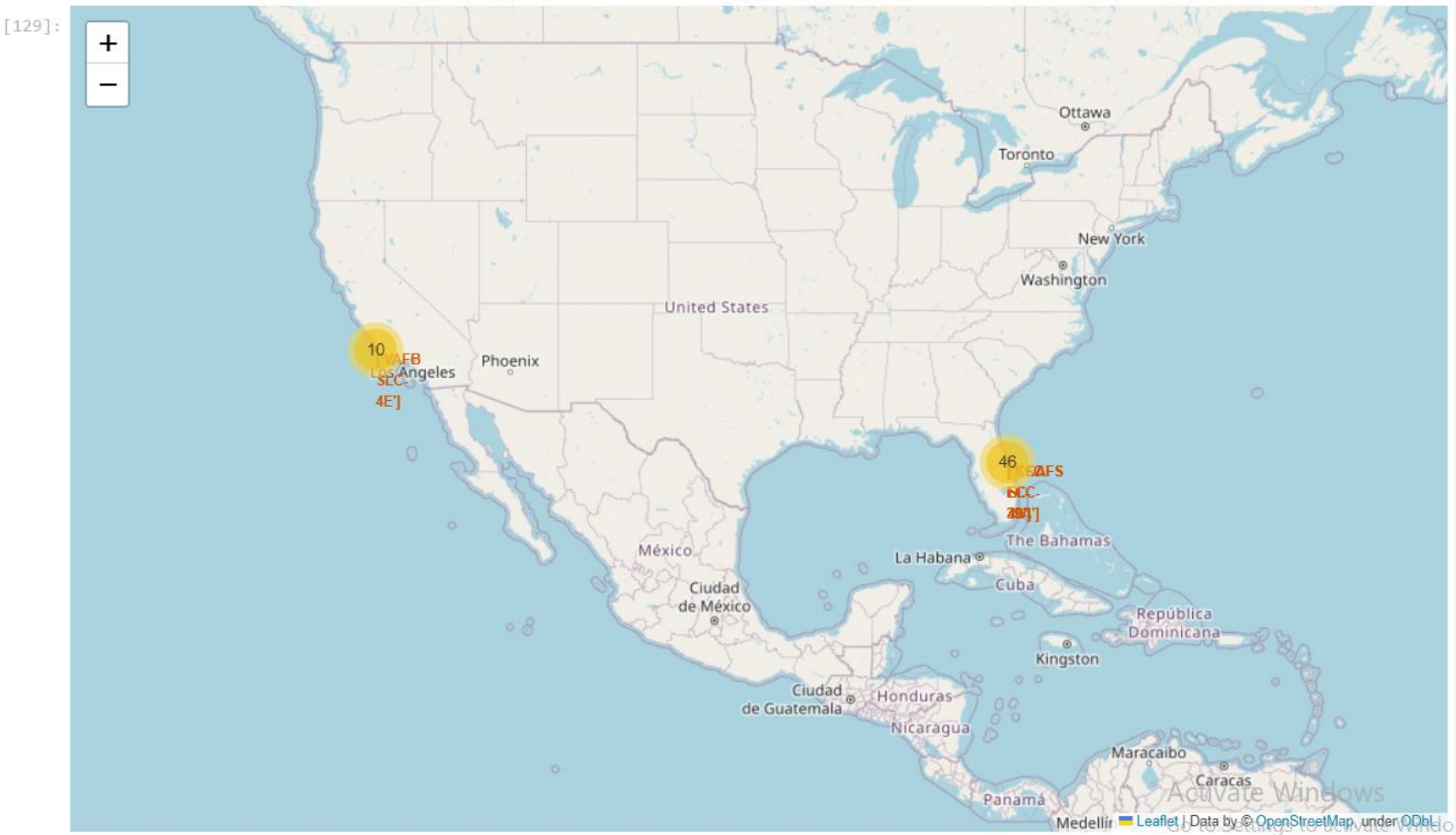
SpaceX Launch Sites

- These are the 4 launch SpaceX launch sites marked using circle markers with their name we can see three sites are very close to each other. Although 2 of them overlap meaning they are the same which was renamed later on. And the one is far away from else.



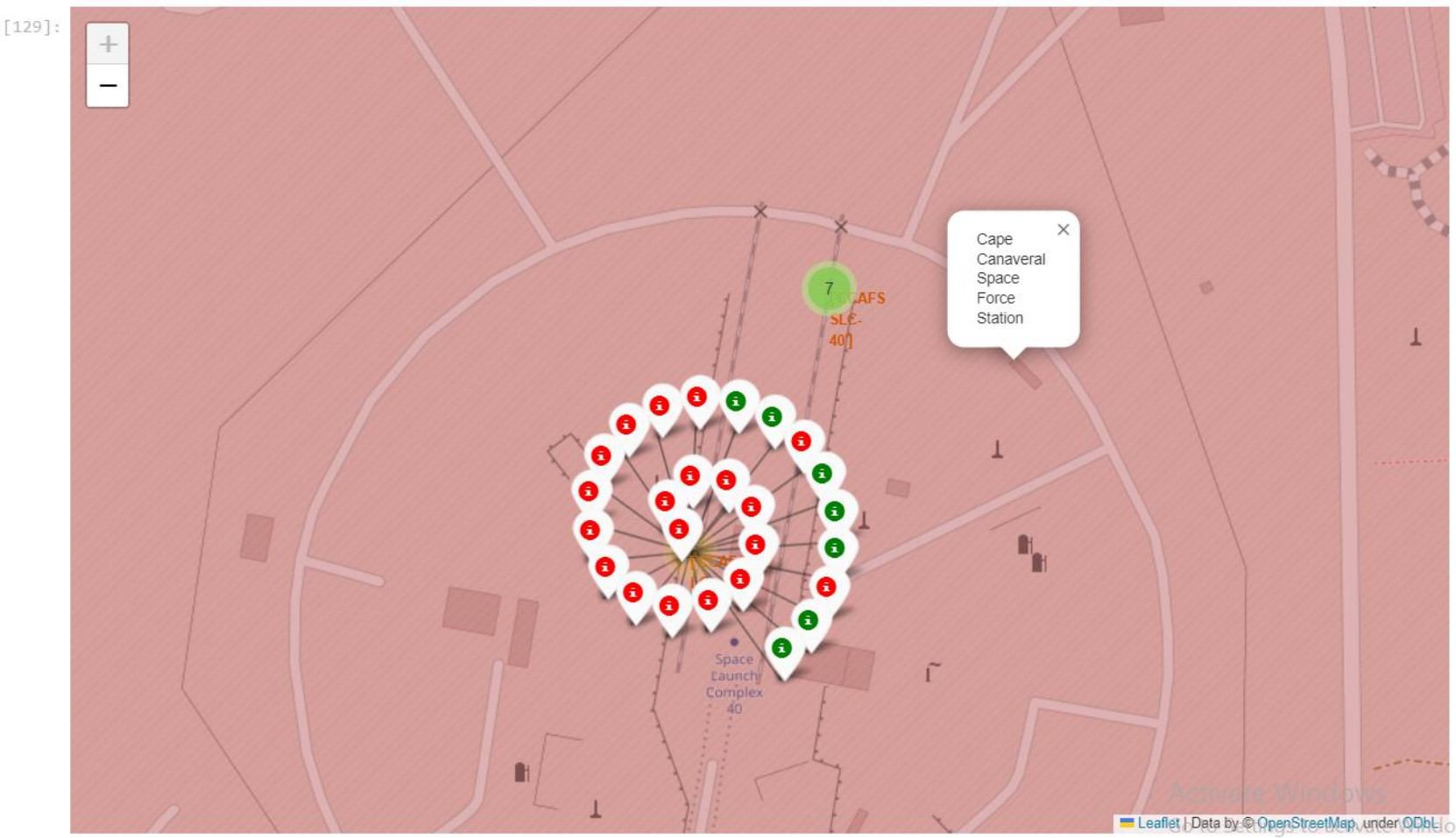
Number of Launches specific to each site

- Here we have used marker clusters representing the number of launches from each of the marked sites respectively. By zooming in they diverge according to each site.



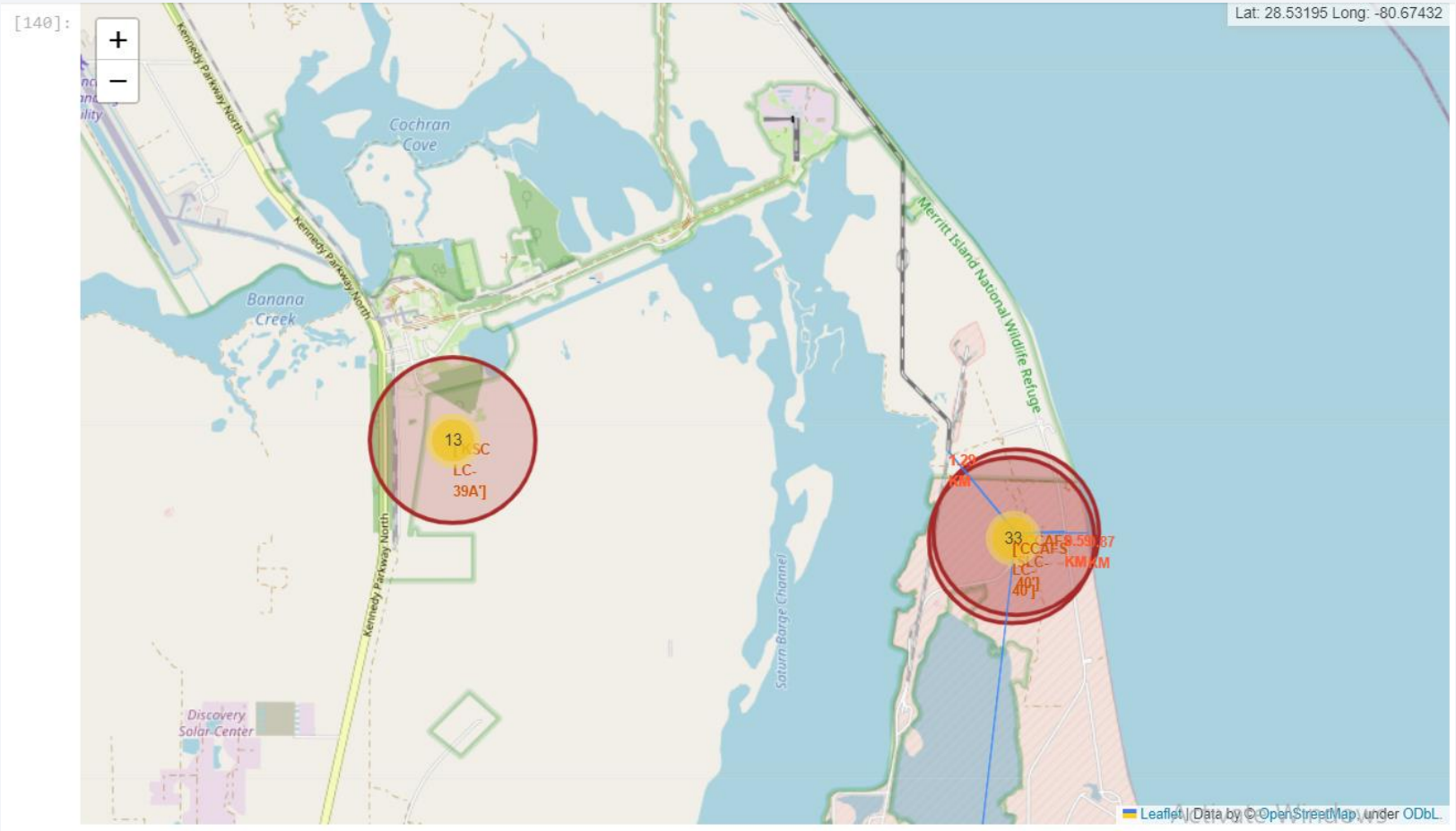
Success and Failure of landings for each site

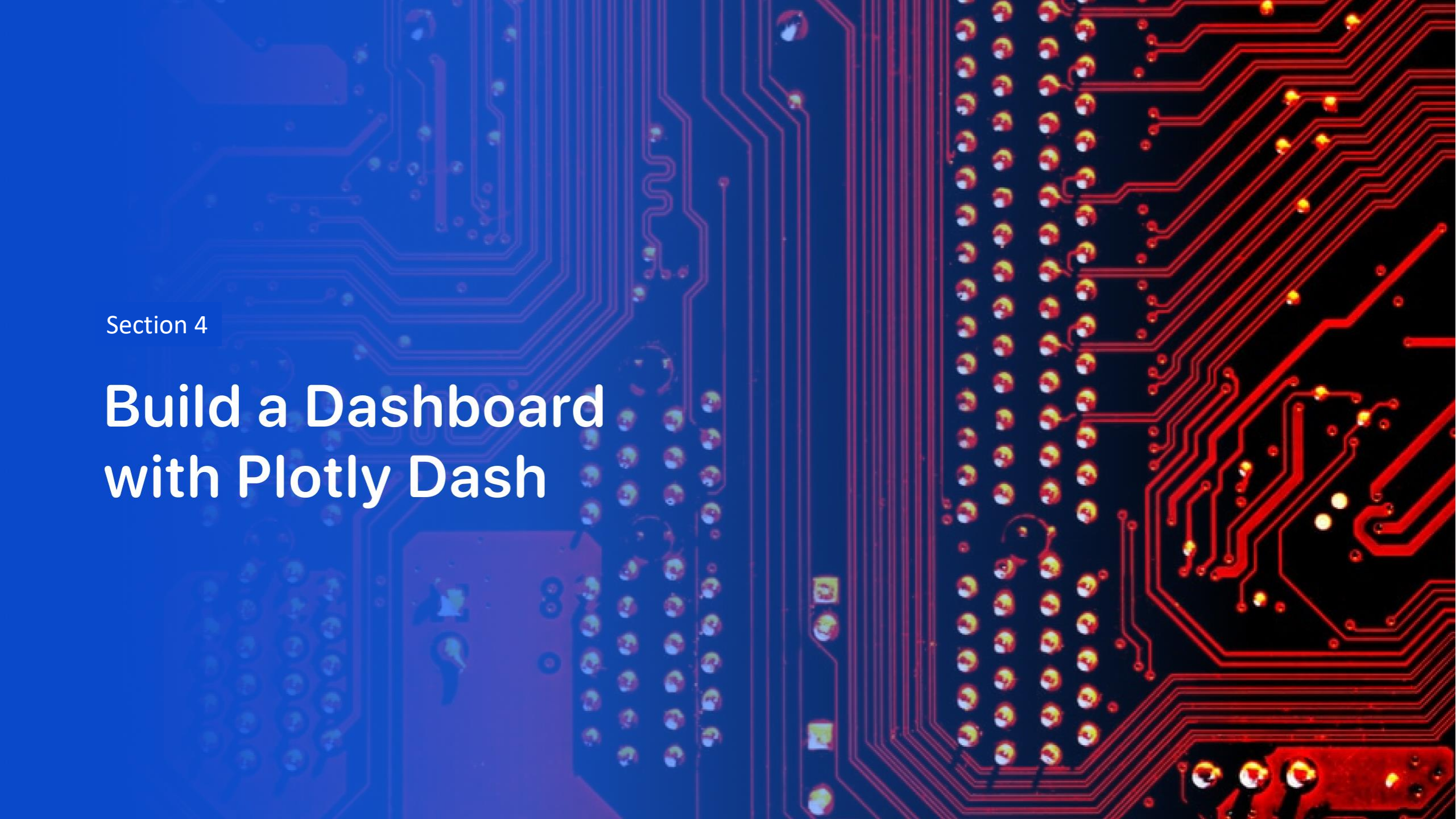
- Here we can see the color labeled markers representing success as green and failure as red of rocket landings and also popups for complete name of the launch sites.



Launch Site’s close Proximities

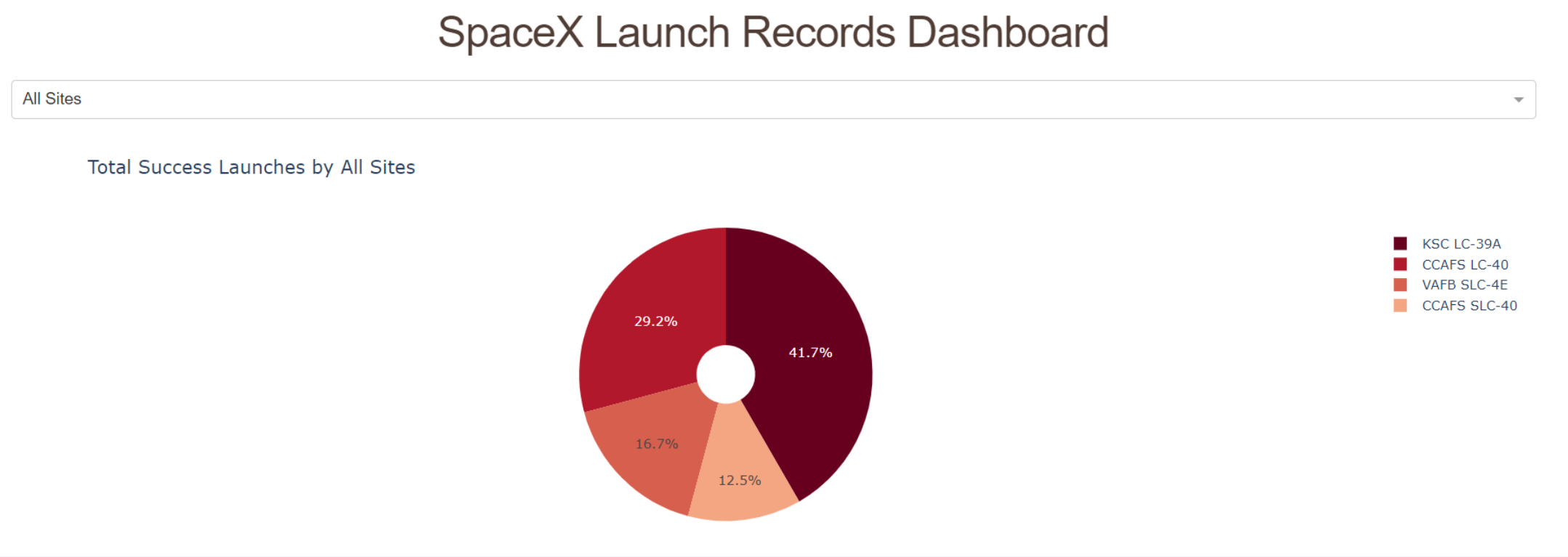
- Here we can see the distance between the launch site and all of the close proximities like coastal lines, railways, highways is calculated and marked by a blue line and also railways, highways and cities are marked with a unique symbols.




Section 4
Build a Dashboard with Plotly Dash

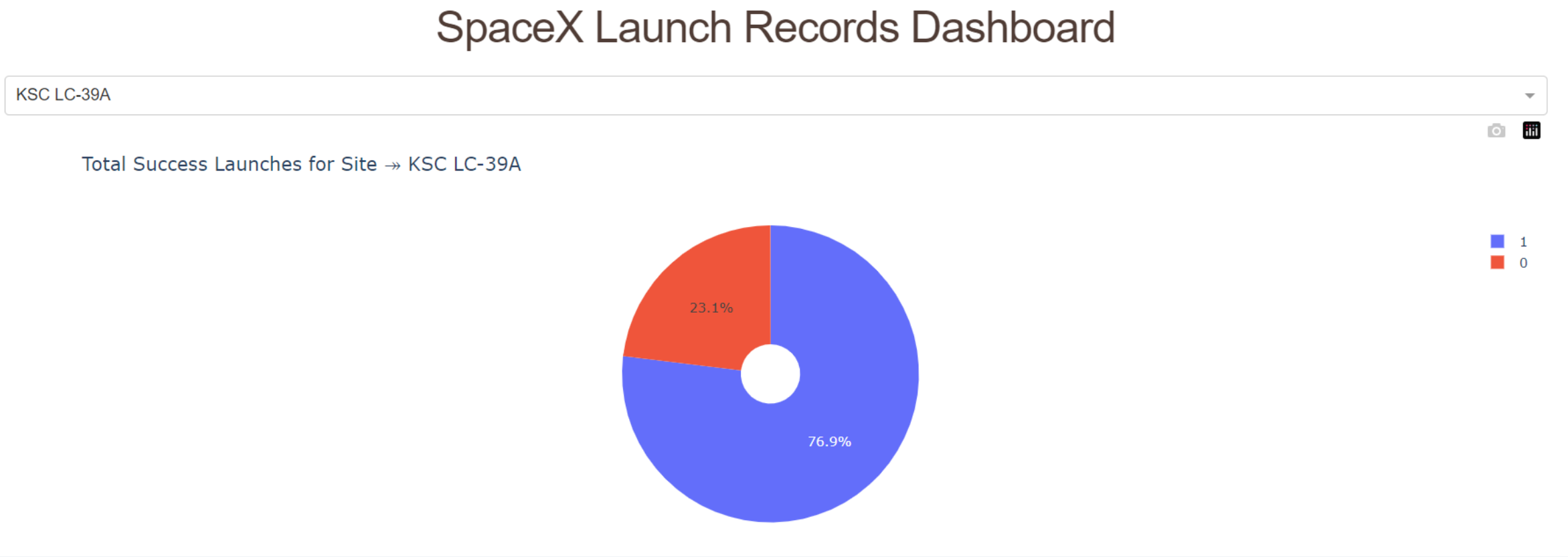
Total Success launches by All Sites

- Total success launch rates of All sites are plotted using pie chart. We can see the highest success rate of launches is at site KSC LC-39A and the lowest at site CCAFS SLC-40.



Highest success rate launch site

- Highest success launch ratio is plotted using a pie chart with the success rate of 76.9% at the site KSC LC-39A.



Scatter plot of Payload v/s Class for various payloads and booster versions used

- This scatter plot shows that payload range between 0kg to 3000kg has higher success rate than further it decreases as payload increases. And the booster version FT has the highest success rate.

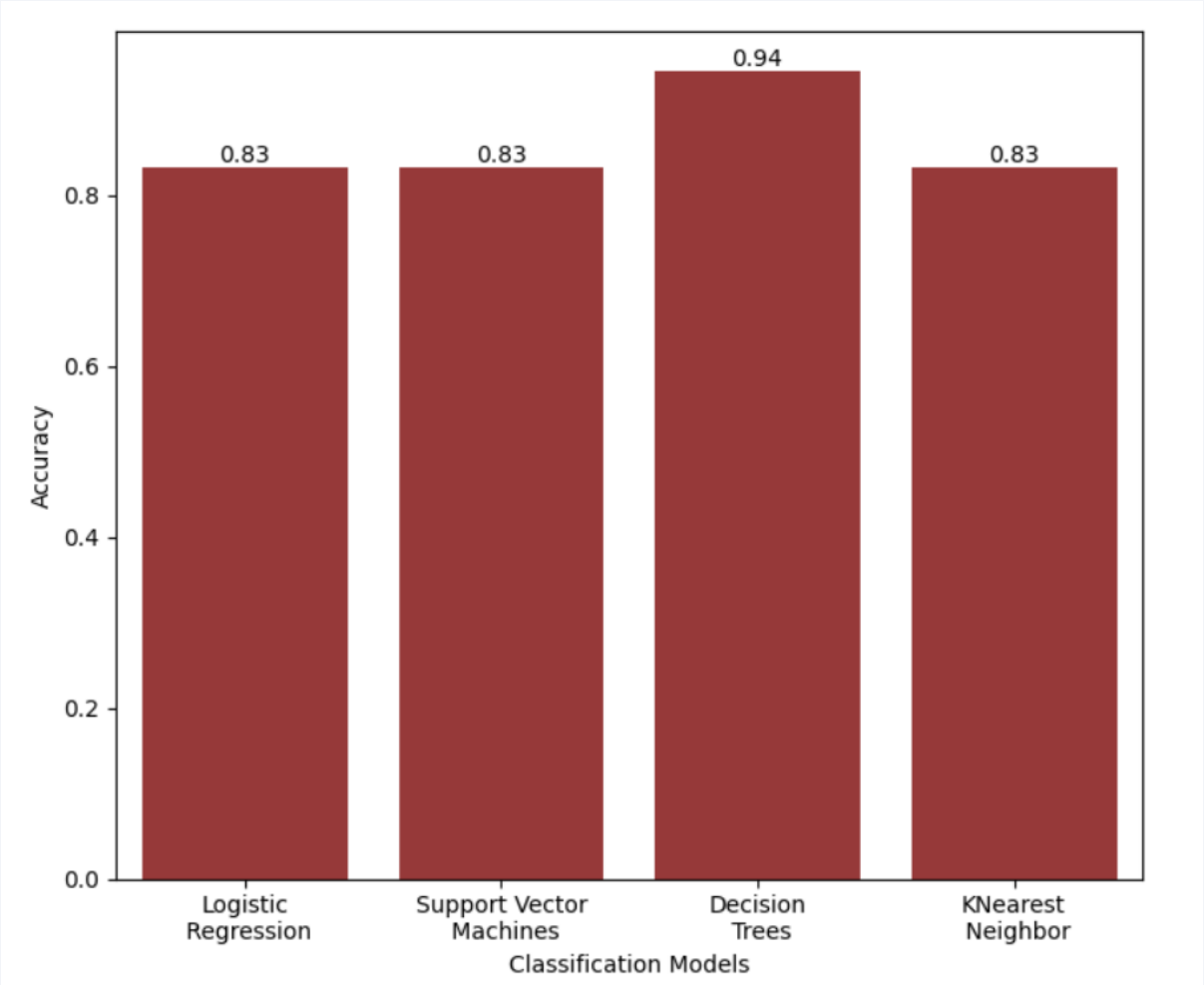
Section 5

Predictive Analysis (Classification)

Classification Accuracy

- Bar plot of all classification models trained and tested on the datasets, gives these accuracy scores.

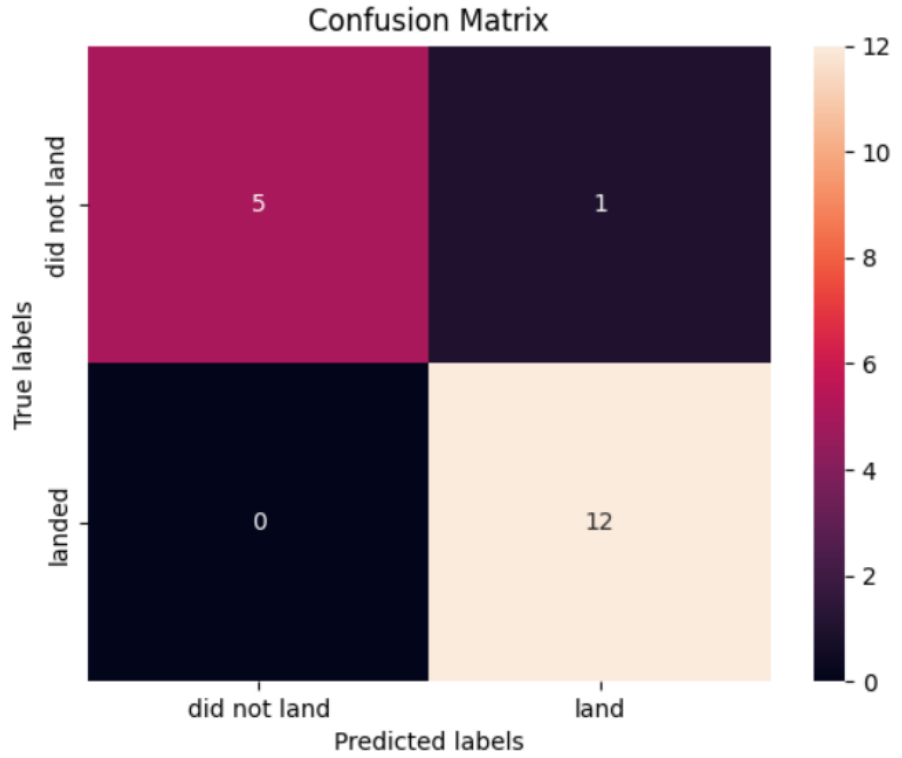
- Decision Trees is the Model with Highest accuracy score of 0.94 and all other Model’ s accuracy is same.



Confusion Matrix

- This is the confusion matrix of the testing dataset of class with the predicted classes by using **Decision Tree model**
- Which gave the accuracy of 94% and here we can see the prediction by the model for ‘did not land’ is 5 and only one was mistaken by the model as land
- And for ‘land’ all the observations were predicted correctly as 12. Overall this model performed very well.

```
[26]: yhat = tree_cv.predict(X_test)
      plot_confusion_matrix(Y_test,yhat)
```

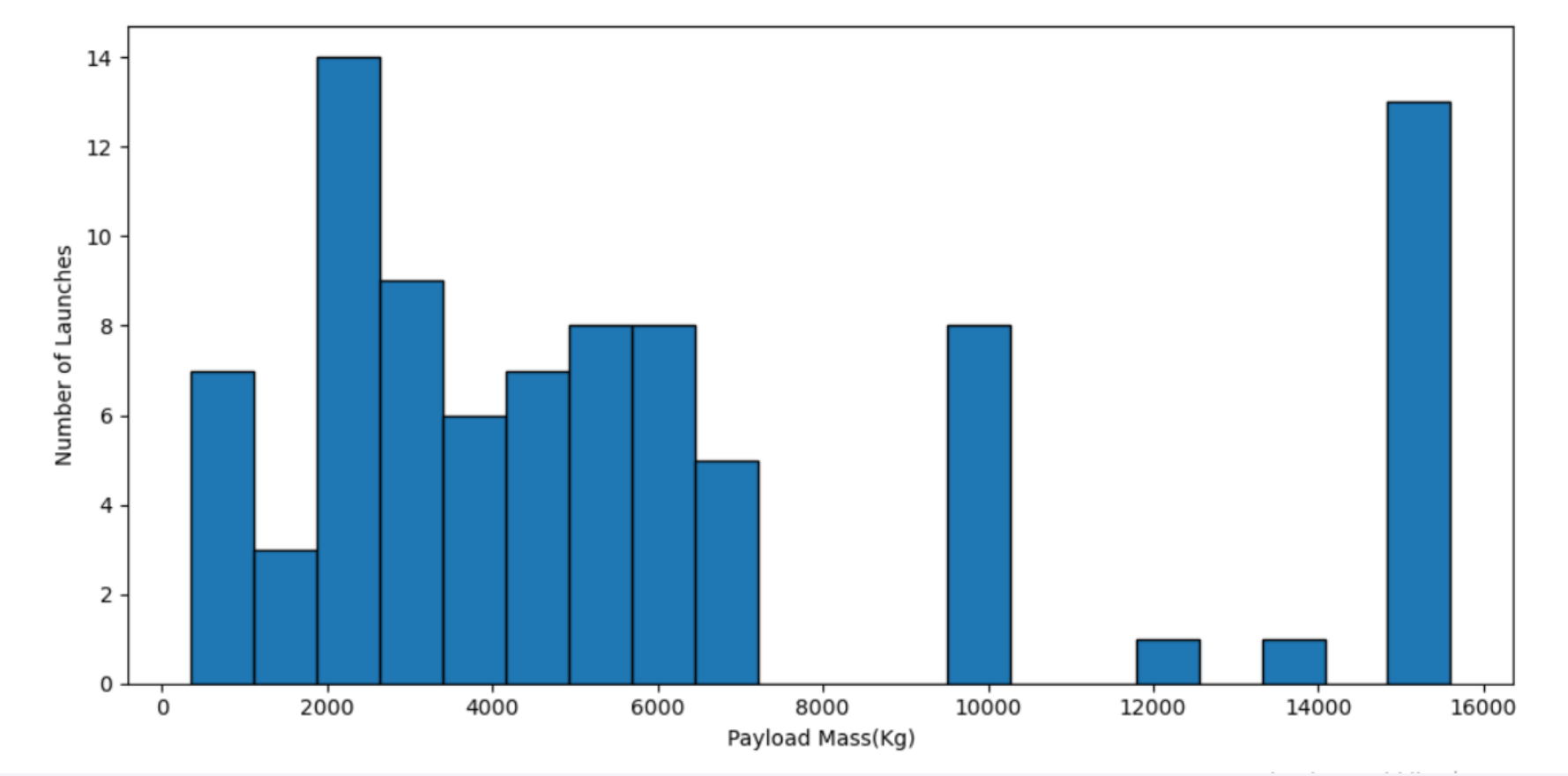


Conclusions

- Point 1: By Data Visualization we find out that payload mass, flight number, orbit type and launch sites are the predictive features for the success or failure of the launch.
- Point 2: By using SQL queries we found out various insights about the landing outcomes and mission outcomes based on the booster versions and payload mass.
- Point 3: By Marking the locations of the launch sites on Map we were able to understand the surrounding of the launch sites railways, coastal lines, highways and nearby cities which should be safe the impact of those surroundings on the launch. Overall understanding of the location of sites and representation of the successful and failed launches at each site
- Point 4: By building a Dashboard with plotly we got to know the success rate at each launch site and by interacting with the slider and dropdown we were able to create plots for various payloads and launch sites with different booster versions.
- Point 5: Finally by getting insights of all the features which can be great predictable variables we tested 4 classification models on the testing dataset and chosen the best one as decision Tree for the prediction of the landing of the First stage of Falcon-9 Rocket which was our goal, which will be further predictable variable for the cost of launch.

Appendix

- Histogram plots the payload mass distribution in Kg. We can see most launches contains payload mass of 2500 Kg and 15000 Kg. And there are no launches of the payload mass of around 8000 Kg.



Thank you!