IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Judah Qin
July 2024

Outline

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

Executive Summary

- Summary of methodologies
- Summary of all results

Introduction

- Project background and context

SpaceX's rocket launches are relatively inexpensive with Falcon 9 rocket launches at a cost of 62 million dollars. This is because SpaceX can reuse the first stage.

- Problems you want to find answers

Therefore, we'd like to predict if the first stage will land, then we can determine the cost of a launch

Our goal is to use this data to predict whether SpaceX will attempt to land a rocket or not.

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Data is gathered from an API, specifically the SpaceX REST API. This API will give us data about launches, including information about the rocket used, payload delivered, launch specifications, landing specifications, and landing outcome.
- Perform data wrangling
  - The data collected is in form of a JSON object and HTML tables, after that the data is converted into a Pandas dataframe for visualization and analysis.
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Use of machine learning to determine if the first stage of Falcon 9 will land successfully

Data Collection

- Describe how data sets were collected.
- You need to present your data collection process use key phrases and flowcharts

The data was gathered from the SpaceX REST API and web scraped from wiki pages

1. Get past launch data as a JSON endpoint SpaceX REST API
2. Get request using the requests library
3. Get past launch data as a JSON object
4. Convert the JSON to a dataframe objects
5. Web scraping Falcon 9 launch records
6. Use BeautifulSoup to web scrape Parse data from HTML tables
7. Parse data from tables
8. Convert tables into a DataFrame

Notebook link

Data Collection – SpaceX API

- Collect and make sure the data is in the correct format from an API

SpaceX REST API endpoint → Get request using the requests library → Extract information about booster object name, launch site, payload mass, and landing site → Get the past launch data as a JSON object → Convert the JSON to a dataframe → Filter the dataframe to only include Falcon 9 launches → Deal with Missing Values → Data Wrangling

[Notebook link]

Data Collection - Scraping

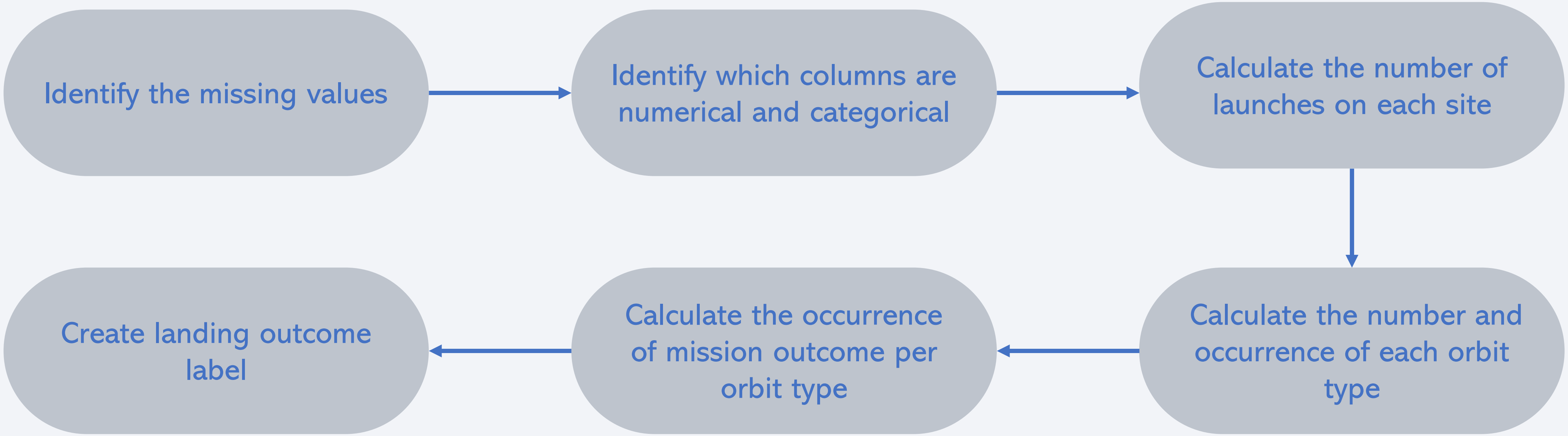
- Perform web scraping to collect Falcon 9 historical launch records from Wikipedia page

1. Request the Falcon9 Launch Wiki page object from its URL
2. BeautifulSoup object from the response
3. Extract column/variable names from the HTML table header
4. Create a data frame by parsing the launch HTML tables
5. Data Wrangling

Notebook link

Data Wrangling

- Perform Exploratory Data Analysis (EDA) to find patterns in the data and determine what would be the label for training supervised models



The variable represents the classification outcome of each launch.
Zero means the first stage did not land successfully; one means the first stage landed successfully.

[Notebook link:]

EDA with Data Visualization

Summary of charts that were plotted:

- Catplot to visualize the relationship between Flight Number and Payload.
- Catplot to visualize the relationship between Flight Number and Launch Site.
- Catplot to visualize the relationship between Payload and Launch Site.
- Bar chart to visualize the relationship between success rate of each Orbit type.
- Catplot to visualize the relationship between Flight Number and Orbit type.
- Catplot to visualize the relationship between Payload and Orbit type.
- Line chart to visualize the launch success yearly trend.

EDA with Visualization notebook

EDA with SQL

SQL queries performed:

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

SELECT DISTINCT(launch_site) FROM SPACEXTBL;

- Display 5 records where launch sites begin with the string

'CCA': SELECT * FROM SPACEXTBL WHERE launch_site LIKE 'CCA%' LIMIT 5;

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

SELECT SUM(payload_mass_kg_) AS TOTAL_PAYLOAD_MASS FROM SPACEXTBL WHERE customer='NASA (CRS)';

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

SELECT AVG(payload_mass_kg) AS AVG_PAYLOAD_MASS FROM SPACEXTBL WHERE booster_version='F9 v1.1';

- List the date when the first successful landing outcome on the ground pad was achieved:

SELECT MIN(DATE) AS first_successful_landing FROM SPACEXTBL WHERE landing_outcome='Success (ground pad)';

SQL notebook

Build an Interactive Map with Folium

Summary of map objects that were created and added to the Folium map:

- folium.Circle and folium.Marker to add a highlighted circle area with a text label on a specific coordinate for each launch site on the site map.
- MarkerCluster object for simplifying a map containing many markers having the same coordinate.
- MousePosition on the map to get coordinates for a mouse over a point on the map.
- folium.PolyLine object to draw a line between a launch site to its closest city, railway, and highway.

Folium notebook

Build a Dashboard with Plotly Dash

Summary of plots/graphs and interactions that were added to the dashboard to perform interactive visual analytics on SpaceX launch data in real-time.

This dashboard application contains input components such as a dropdown list and a range slider to interact with a pie chart and a scatter point chart.

- A launch Site Drop-down Input Component. There are four different launch sites and a dropdown menu lets us select different launch sites.
- A callback function to render success-pie-chart based on the selected site dropdown. The general idea of this callback function is to get the selected launch site from the site-dropdown and render a pie chart visualizing launch success counts.
- A range Slider to Select Payload. The Slider is to be able to easily select different payload ranges and see if we can identify some visual patterns. Interactive Dashboard with Plotly Dash

Dashboard Notebook

Predictive Analysis (Classification)

Summary of the model development process used to predict if the first stage will land given the data from the preceding labs:

- Creation of a NumPy array from the column Class in data.
- Data standardization. - Use of the function train_test_split to split the data X and Y into training and test data.
- Searching for the best Hyperparameters for Logistic Regression, SVM, Decision Tree, and KNN classifiers.
- Searching for the method that performs best using test data.

Machine Learning Prediction notebook

Results

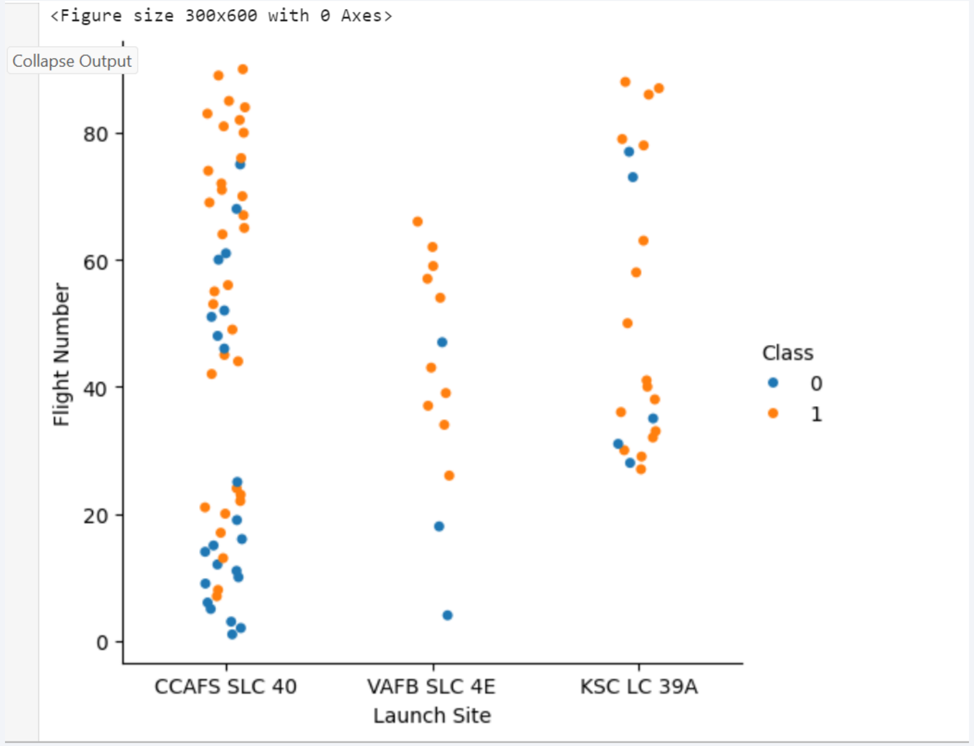
- Exploratory data analysis results
- Interactive analytics demo in screenshots
- Predictive analysis results

Section 2

Insights drawn from EDA

Flight Number vs. Launch Site

- Show a scatter plot of Flight Number vs. Launch Site

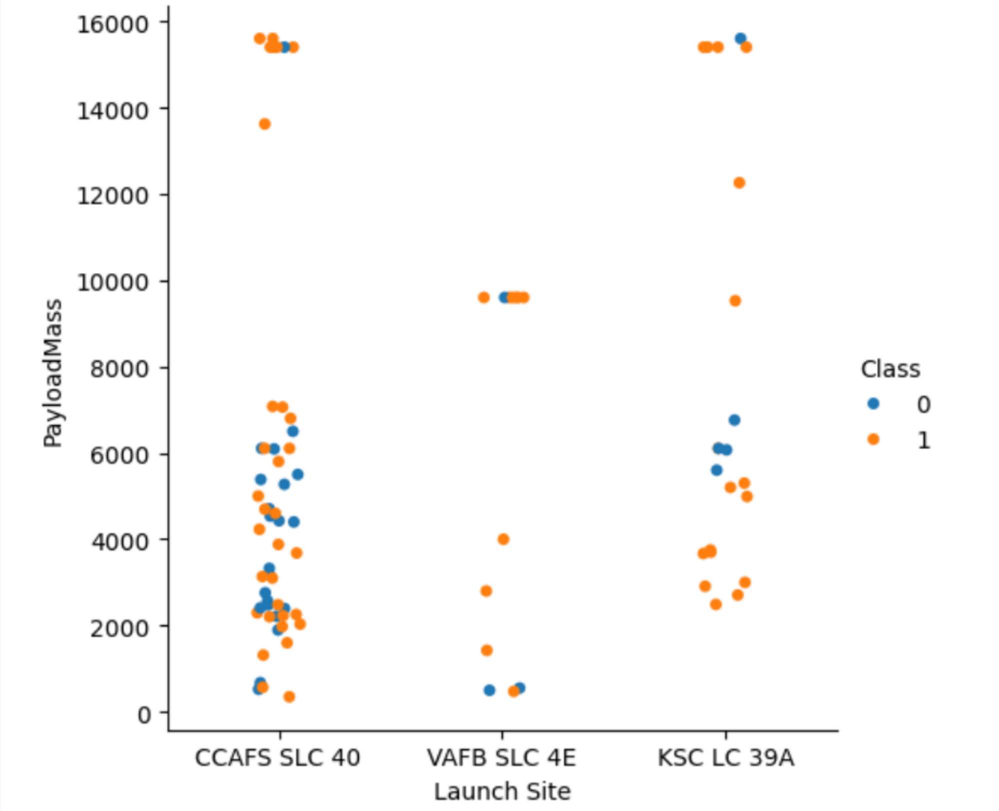


- With time the successful rate has increased for every Launch Site, especially for CCAFS SLC 40, where are concentrated the majority of the launches.

- VAFB SLC 4E and KSC LC 39A have a higher successful rate but represent one third of the total launches.

Payload vs. Launch Site

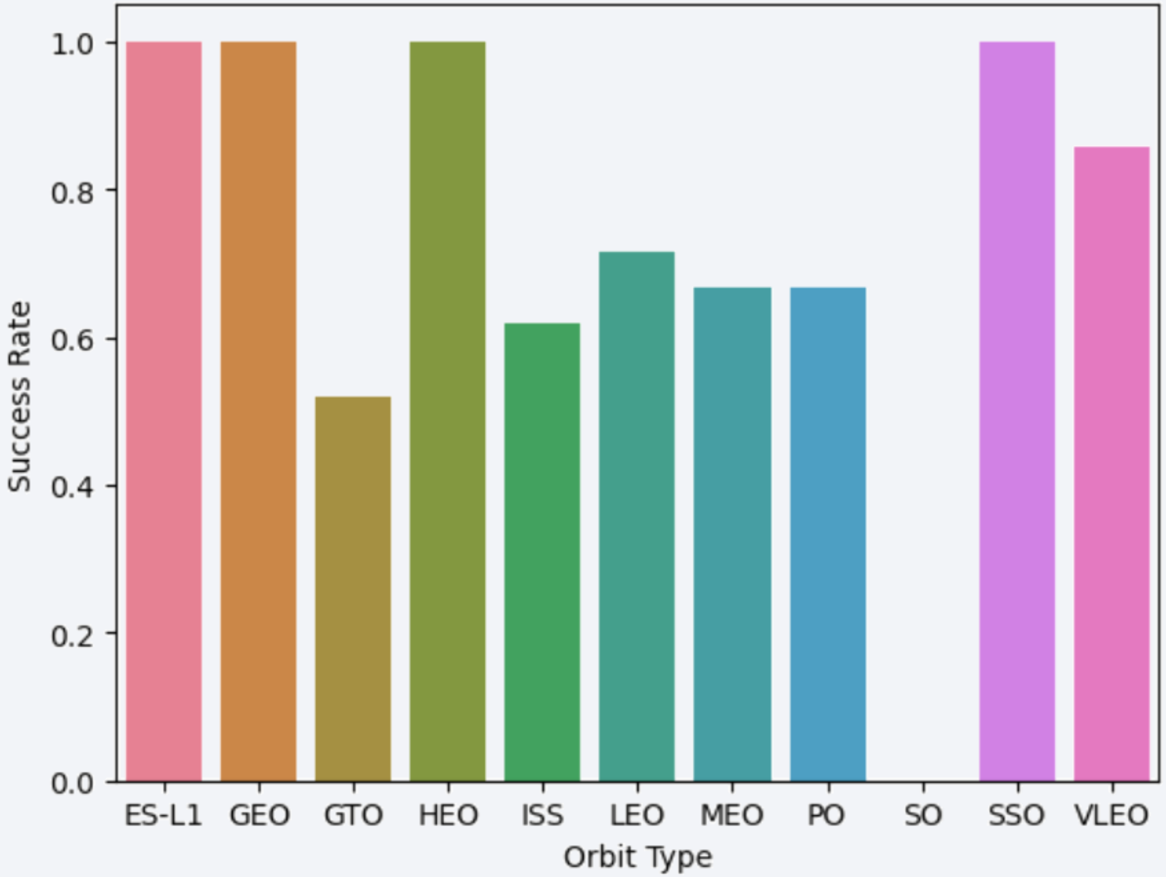
- Show a scatter plot of Payload vs. Launch Site



- In VAFB-SLC launch site there are no rockets launched for heavy payload mass (greater than 10000 kg).
- In KSC LC launch site there are no rockets launched for lower payload mass (less than 2500 kg).
- CCAFS SLC has launched rockets less than 7500 kg and more than 13000 kg payload mass but not in between.

Success Rate vs. Orbit Type

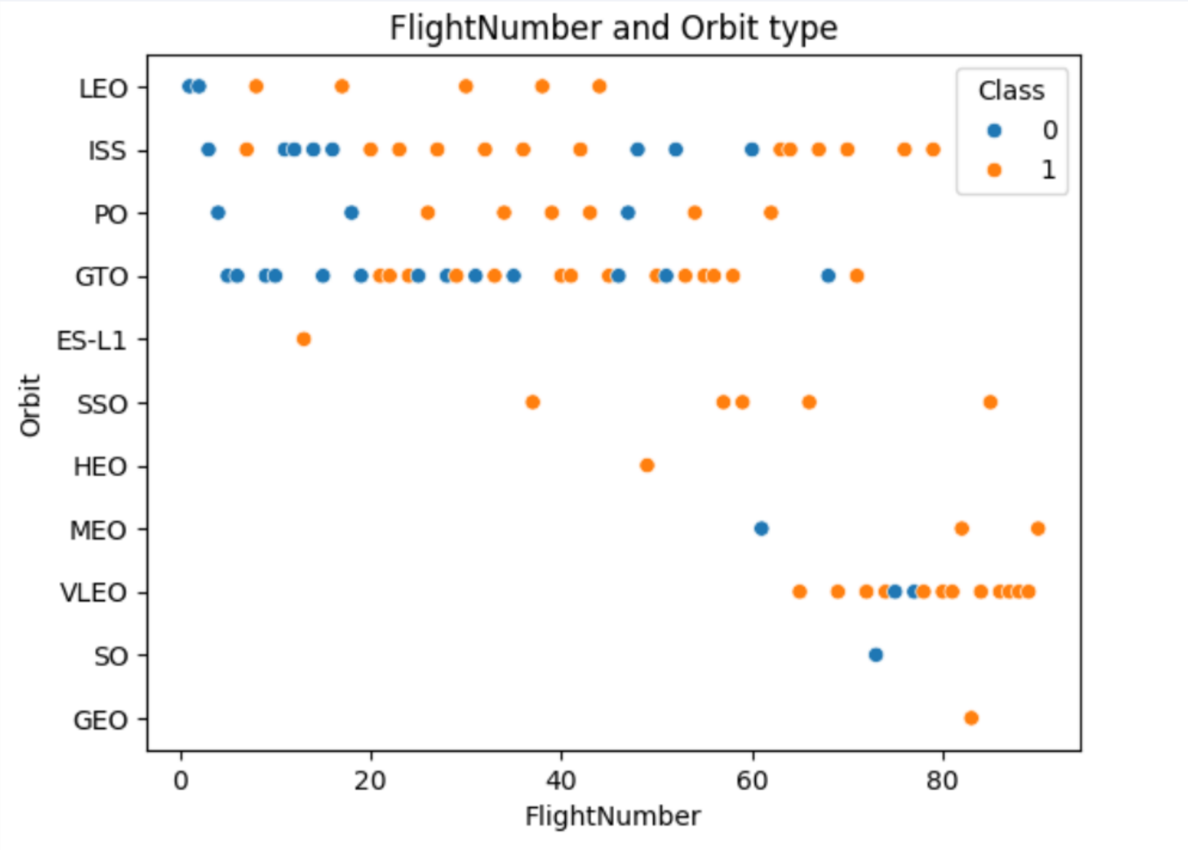
- Show a bar chart for the success rate of each orbit type



- The 4 Orbit types (ESL1 GEO HEO SSO) have the best success rate. But how many type attempts are per orbit type?
- The bar chart must be interpreted with the number of launches per orbit type.

Flight Number vs. Orbit Type

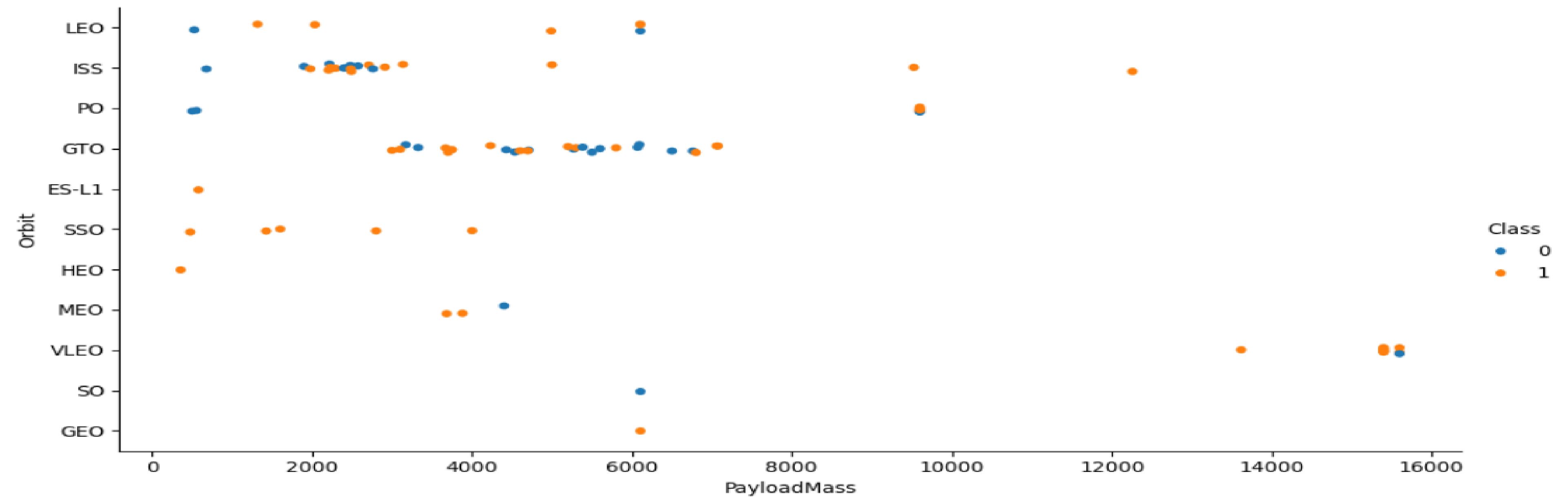
- Show a scatter point of Flight number vs. Orbit type



- As expected, there are more failures at the beginning of the series of launches, but after the first 40 launches, the ratio improves by reducing 50 percent of unsuccessful landings.
- GTO and ISS orbits have the higher concentration of launches with the lowest ratio of successful landings.
- The orbits with higher successful rates have one or just a few number of launches.

Payload vs. Orbit Type

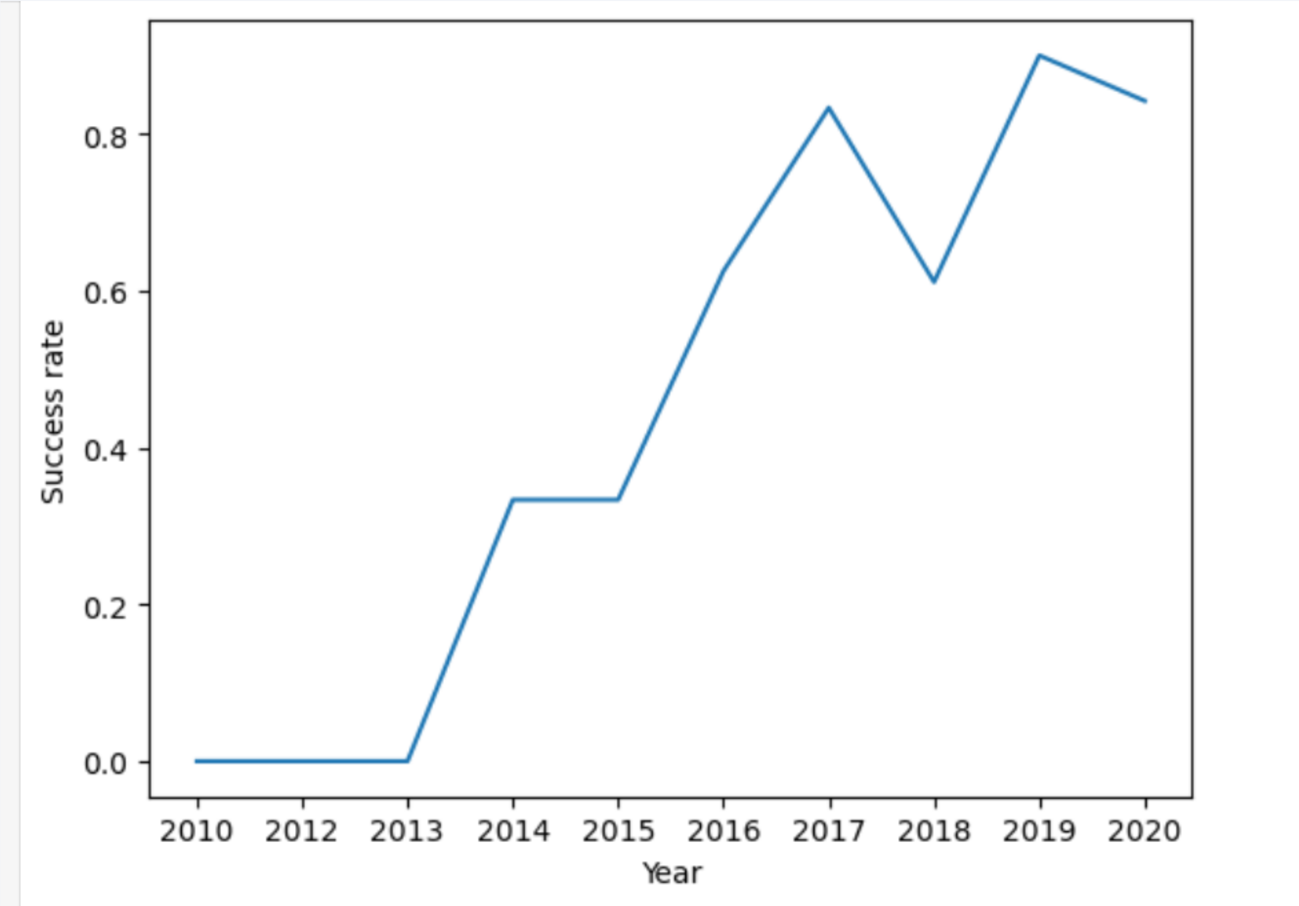
- Show a scatter point of payload vs. orbit type



- There is a visible limit of Payload around 7600 kg. Less than 10 launches exceed that limit.
- With heavy payloads, the successful landing rate is higher for Polar, LEO, and ISS.
- However, for GTO, we cannot distinguish this well as both positive landing rate and negative landing are both there.

Launch Success Yearly Trend

- Show a line chart of yearly average success rate



All Launch Site Names

- Find the names of the unique launch sites
- Present your query result with a short explanation here

```
[16]: %sql SELECT DISTINCT Launch_Site FROM SPACEXTABLE

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

[16]:

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`

[18]:
```
%sql SELECT * FROM SPACEXTABLE WHERE Launch_Site like 'CCA%' LIMIT 5
```

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

[18]:

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer
2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX
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
2012-05-22	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)
2012-10-08	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)
2013-03-01	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)

Total Payload Mass

- Calculate the total payload carried by boosters from NASA

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

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

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

[24]:

SUM (PAYLOAD_MASS__KG_)
45596

Average Payload Mass by F9 v1.1

- Calculate the average payload mass carried by booster version F9 v1.1

Task 4

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

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

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

[25]:

AVG (PAYLOAD_MASS__KG_)
2928.4

First Successful Ground Landing Date

- Find the dates of the first successful landing outcome on ground pad

Task 5

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

Hint:Use min function

```
[35]: %sql SELECT MIN (DATE) FROM SPACEXTABLE WHERE Landing_Outcome = 'Success (ground pad)'
```

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

[35]:

MIN (DATE)
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

Task 6

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

[33]:
```
%%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.
```

[33]:

Booster_Version
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

Task 7

List the total number of successful and failure mission outcomes

[47]:
```
%%sql SELECT
    COUNT(CASE WHEN Mission_Outcome = 'Success' THEN 1 END) AS 'Successful Mission',
    COUNT(CASE WHEN Mission_Outcome != 'Successful' THEN 1 END) AS 'Failure Mission'
    FROM SPACEXTABLE
```

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

[47]:

Successful Mission	Failure Mission
98	101

Boosters Carried Maximum Payload

- List the names of the booster which have carried the maximum payload mass

Task 8

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

[59]:
```
%%sql SELECT Booster_Version
    FROM SPACEXTABLE
    WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTABLE)
```

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

[59]:

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

Task 9

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.

[60]:
```
%%sql SELECT SUBSTR(Date, 6,2) AS 'Month', Landing_Outcome, Launch_Site
FROM SPACEXTABLE
WHERE SUBSTR(Date, 0,5) = '2015'
```

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

[60]:

Month	Landing_Outcome	Launch_Site
01	Failure (drone ship)	CCAFS LC-40
02	Controlled (ocean)	CCAFS LC-40
03	No attempt	CCAFS LC-40
04	Failure (drone ship)	CCAFS LC-40
04	No attempt	CCAFS LC-40
06	Precluded (drone ship)	CCAFS LC-40
12	Success (ground pad)	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

Task 10

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.

[72]:

```
%%sql SELECT Landing_Outcome, COUNT(Landing_Outcome)
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.
```

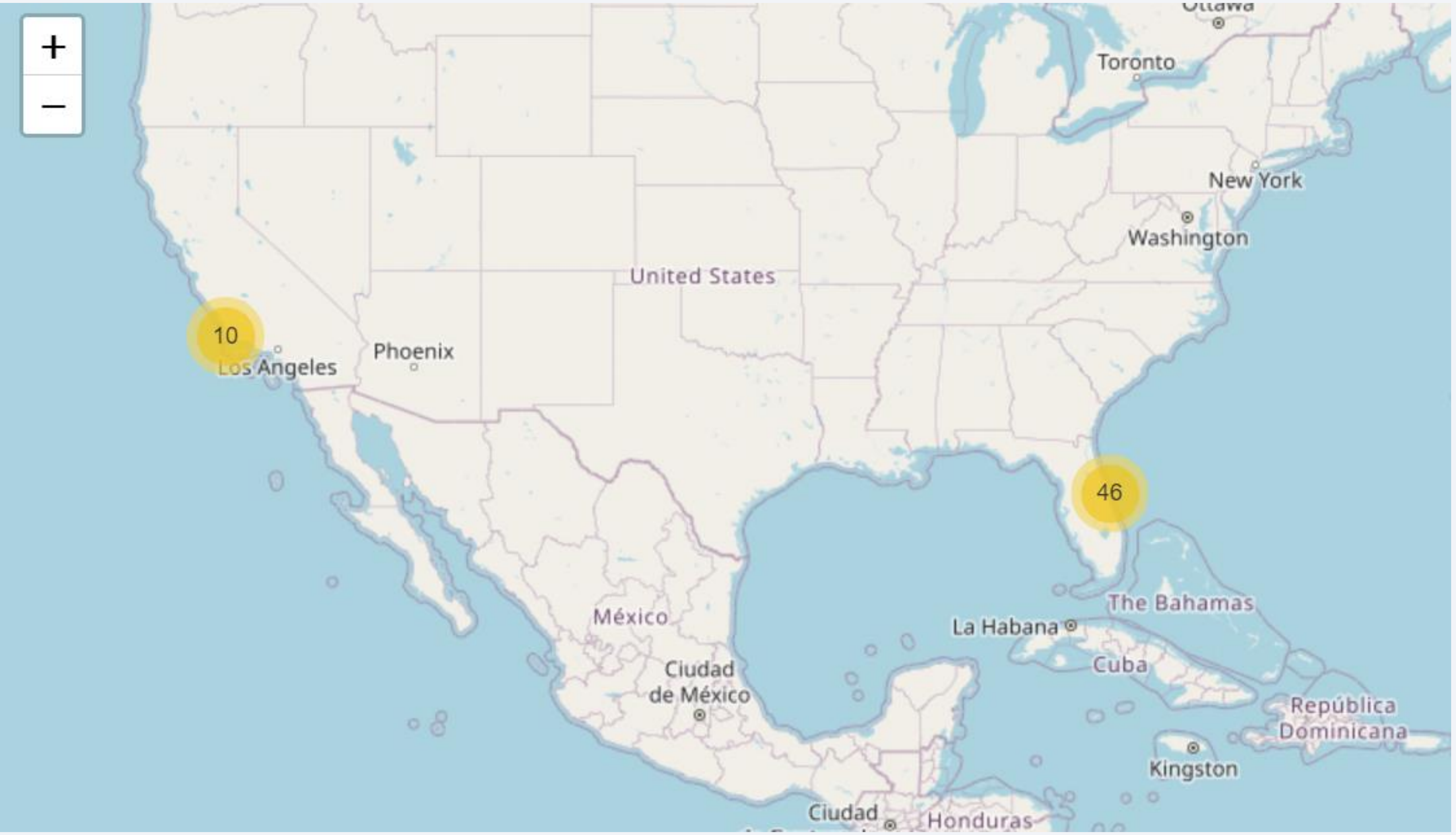
[72]:

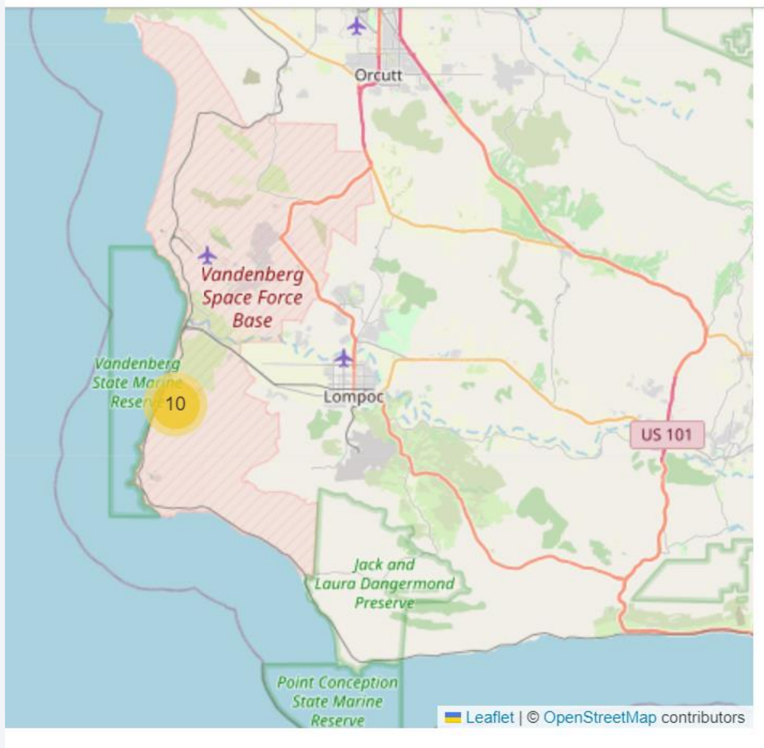
Landing_Outcome	COUNT(Landing_Outcome)
No attempt	10
Success (drone ship)	5
Failure (drone ship)	5
Success (ground pad)	3
Controlled (ocean)	3
Uncontrolled (ocean)	2
Failure (parachute)	2
Precluded (drone ship)	1

Section 3

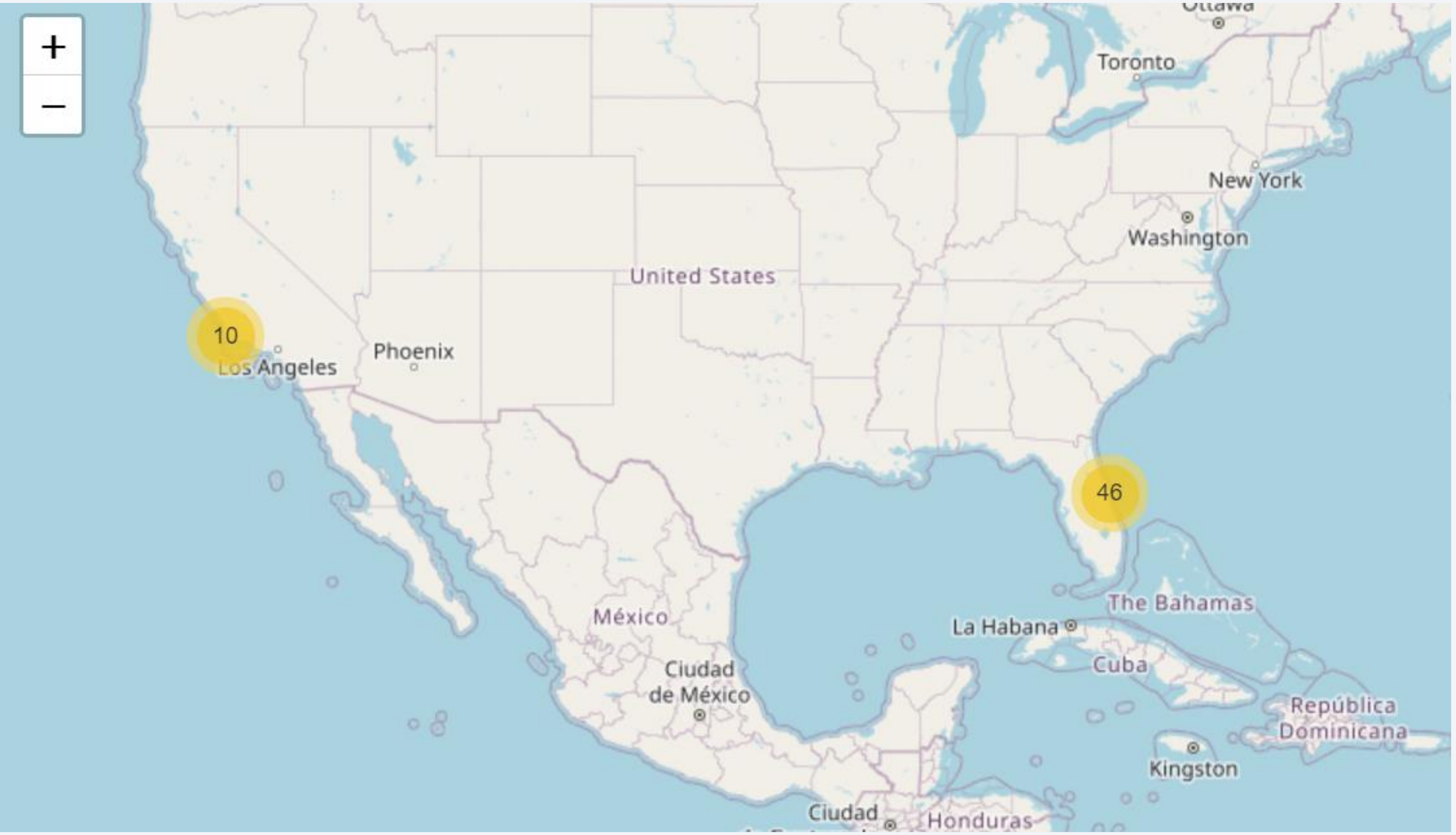
Launch Sites Proximities Analysis

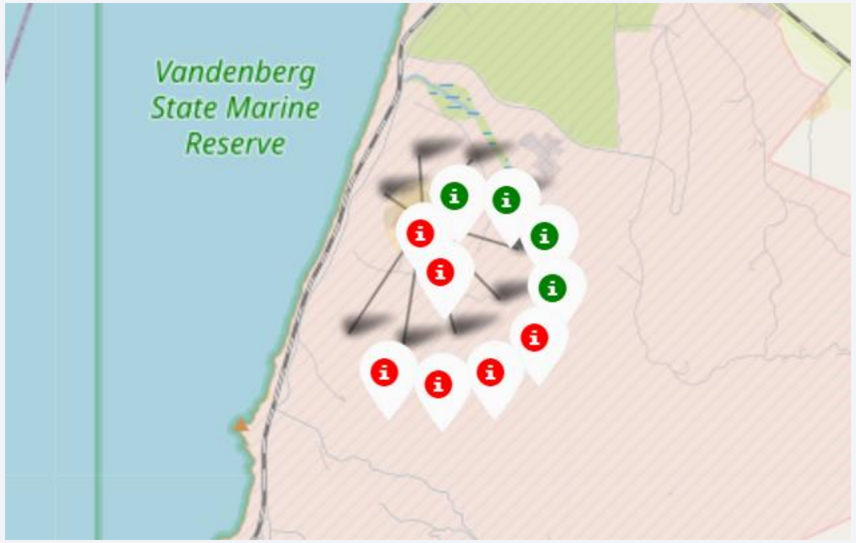
<Folium Map Screenshot 1>



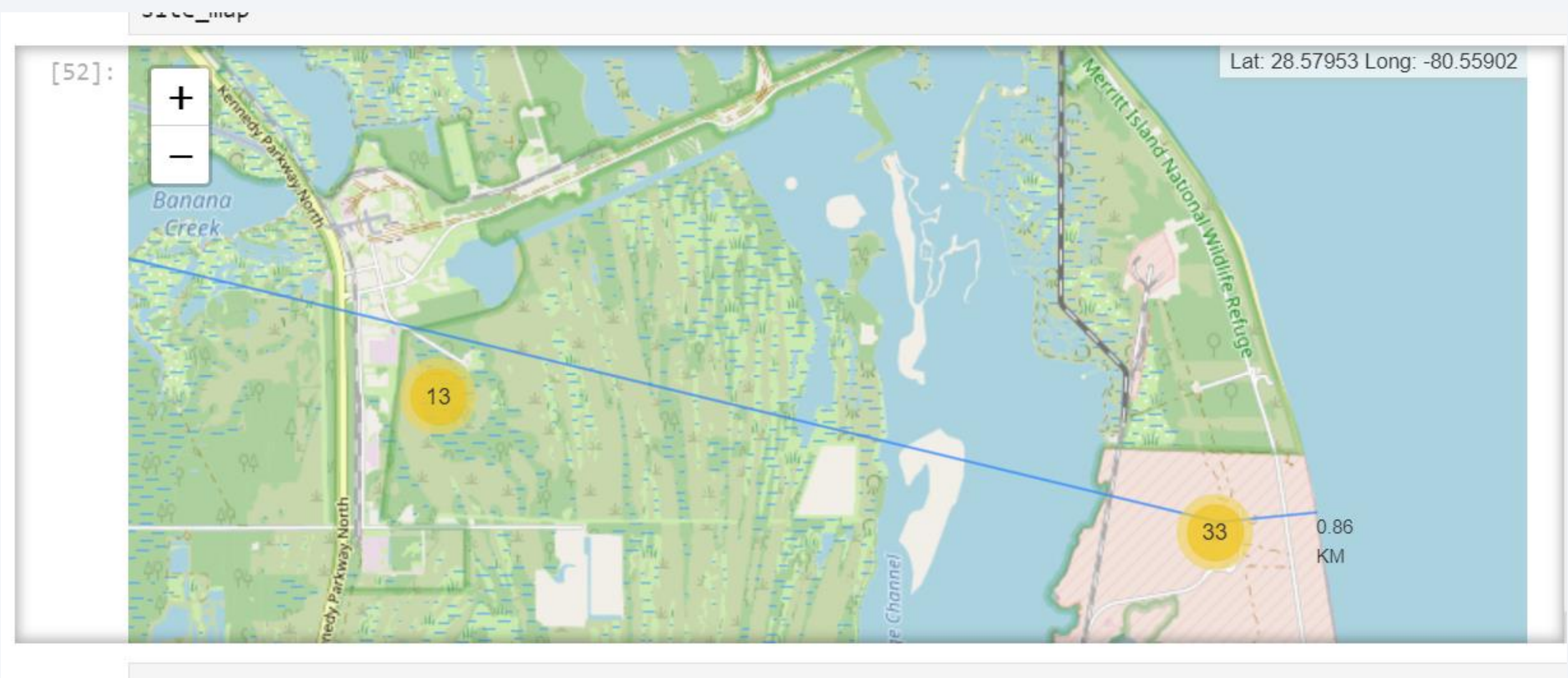


<Folium Map Screenshot 2>





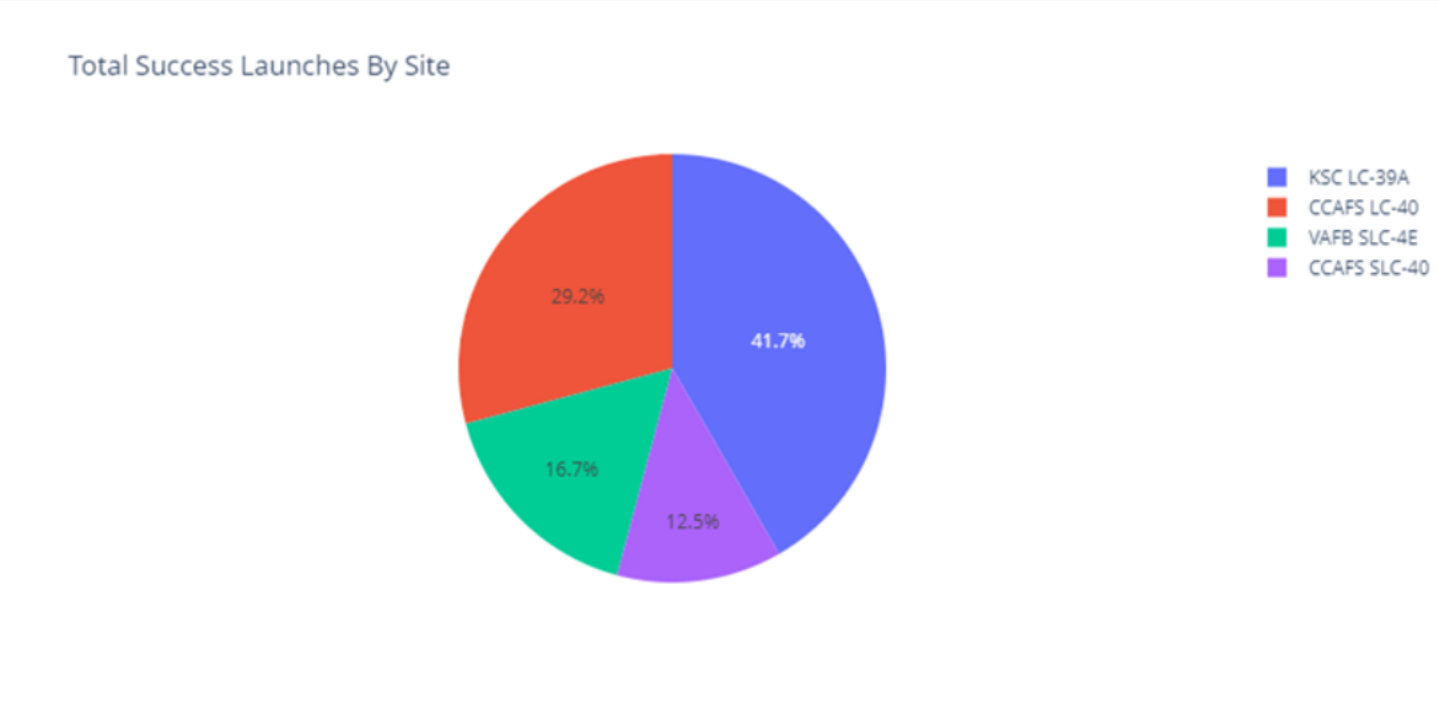
<Folium Map Screenshot 3>



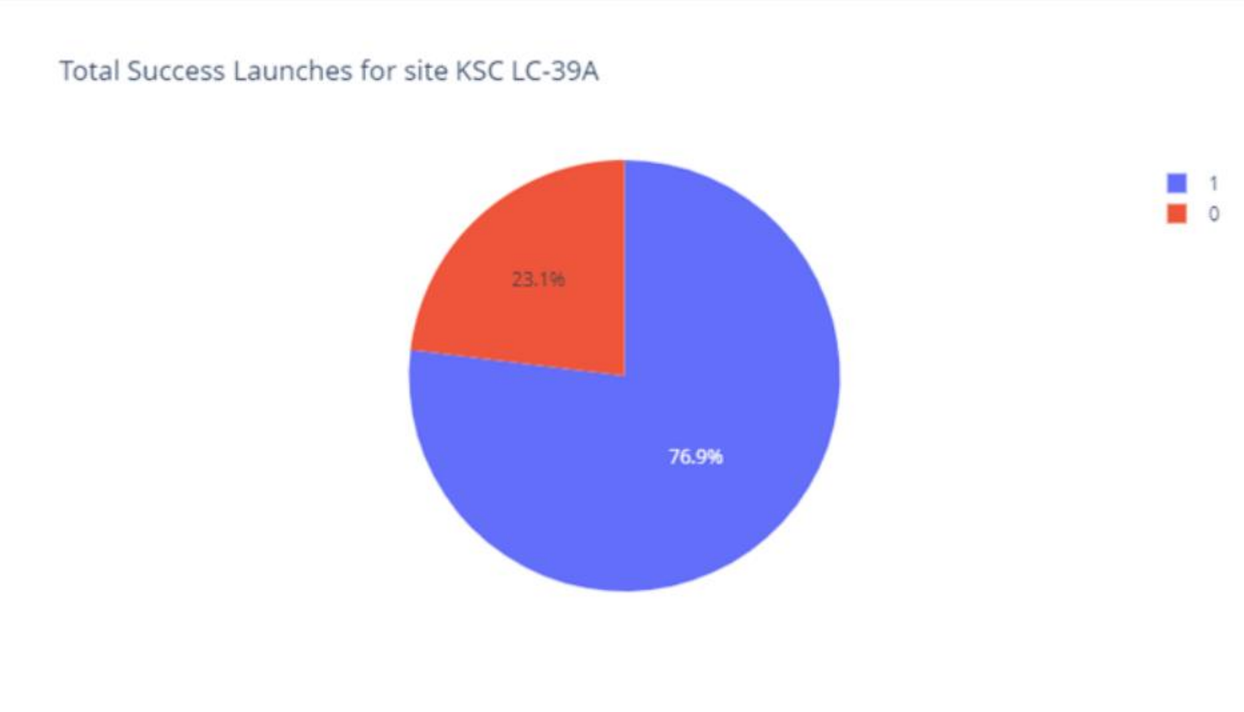
Section 4

Build a Dashboard with Plotly Dash

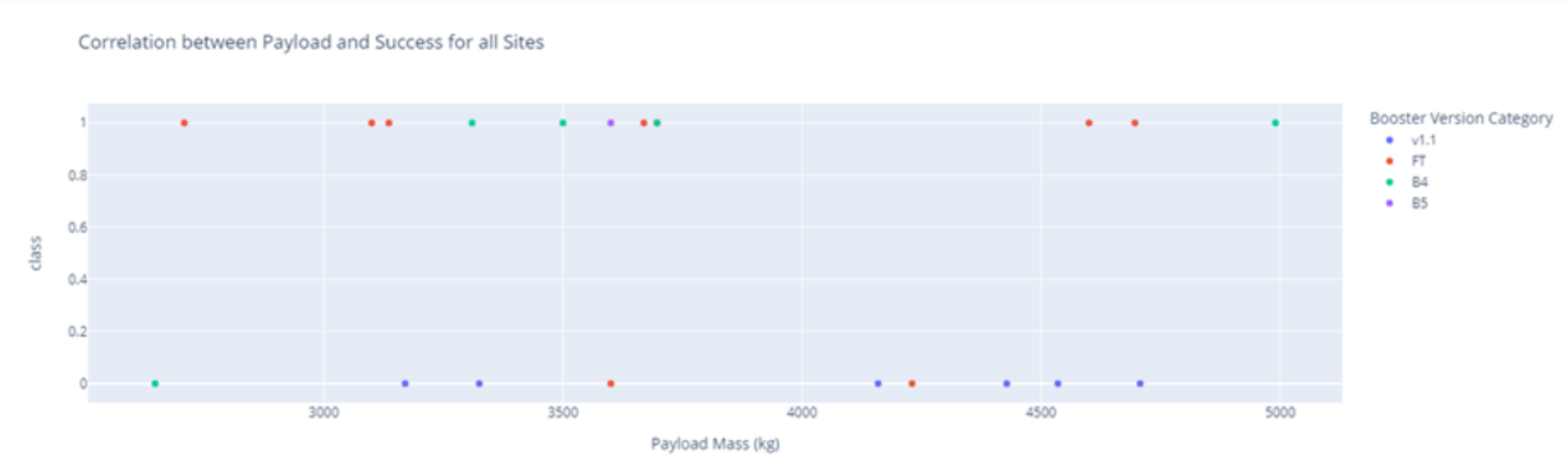
<Dashboard Screenshot 1>

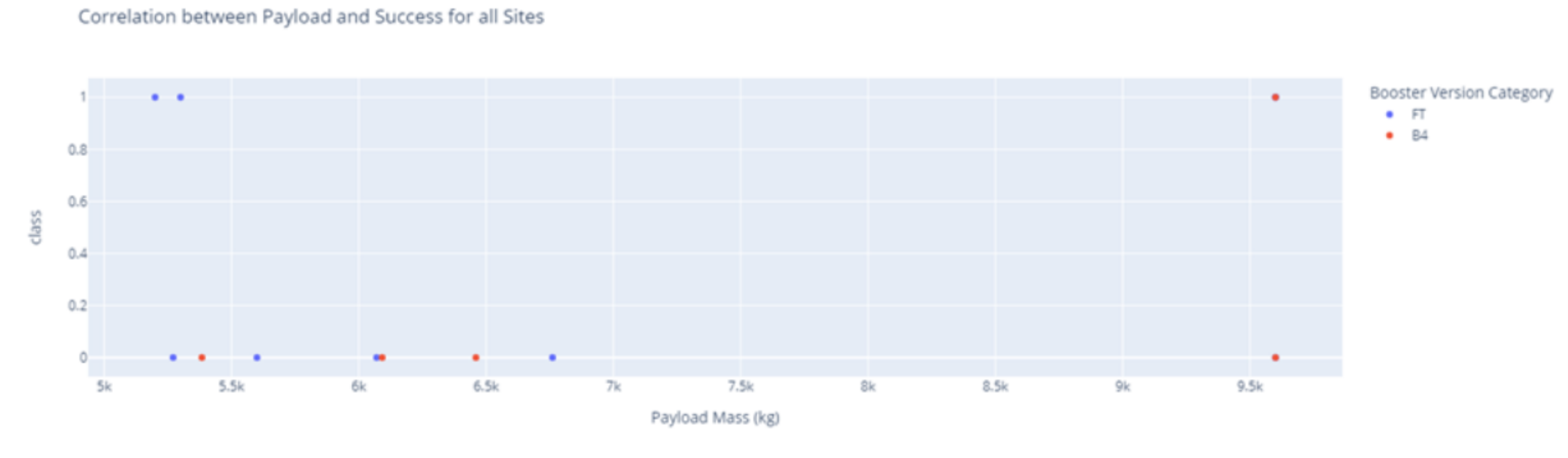


<Dashboard Screenshot 2>



<Dashboard Screenshot 3>



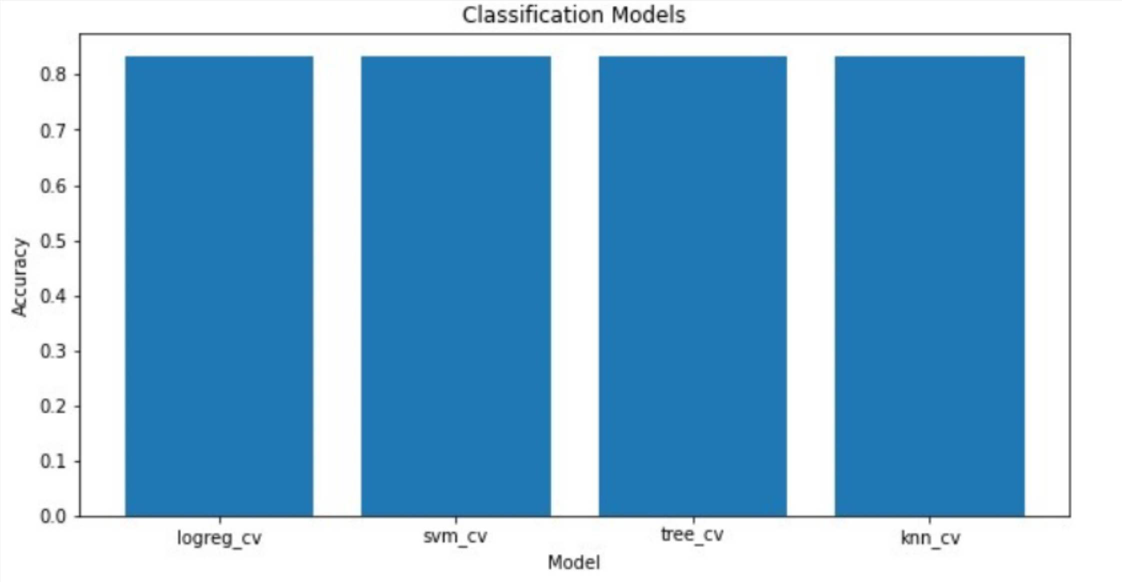


Section 5

Predictive Analysis (Classification)

Classification Accuracy

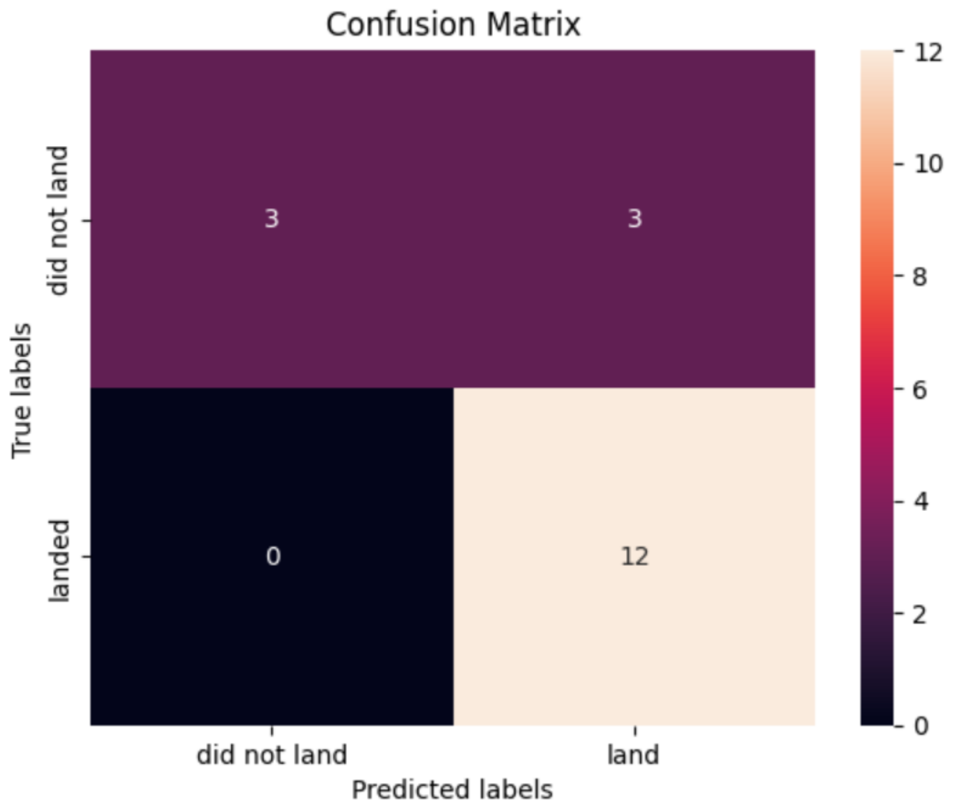
- The accuracy is the same for all models



Confusion Matrix

- The confusion matrix is the same for all models

```
[32]: yhat=logreg_cv.predict(X_test)
      plot_confusion_matrix(Y_test,yhat)
```



Conclusions

- As all the algorithms are giving the same accuracy, they all perform practically the same.
- By using our machine learning model, we can predict if the first stage of our competitor will land and determine the cost of a launch.

Appendix

- Include any relevant assets like Python code snippets, SQL queries, charts, Notebook outputs, or data sets that you may have created during this project

Thank you!