IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Ashish Lotangane
15 June 2023

Outline

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

Executive Summary

Reusability is key and critical aspect for any business. Our company make best of reusability to determine the price of each launch

The commercial space age is here, companies are making space travel affordable for everyone.
Virgin Galactic is providing suborbital spaceflights. Rocket Lab is a small satellite provider. Blue Origin manufactures sub-orbital and orbital reusable rockets. Perhaps the most successful is SpaceX.
In competition with these companies our company SPACE Y wants to make the space affordable for everyone.

Summary of methodologies

- Data Collection with API and Web scraping
- Data Wrangling
- Exploratory Data Analysis (EDA) with SQL and visualization
- Interactive Visual Analytics with Folium and Plotly Dash
- Predictive Analysis (Classification)

Summary of all results

- The best Hyperparameters for Logistic regression, SVM, Decision Tree and KNN classifiers
- The method that performs best using test data

Introduction

- SPACE Y that would like to compete with SpaceX founded by Billionaire industrialist Allon Musk
- Our objective is to
  - Use data analysis and machine learnings techniques to determine the price of each launch.
  - To determine reusability of first stage
- Ultimately SPACE Y is here to compete in the commercial space industry, making it affordable and inexpensive.

Section 1

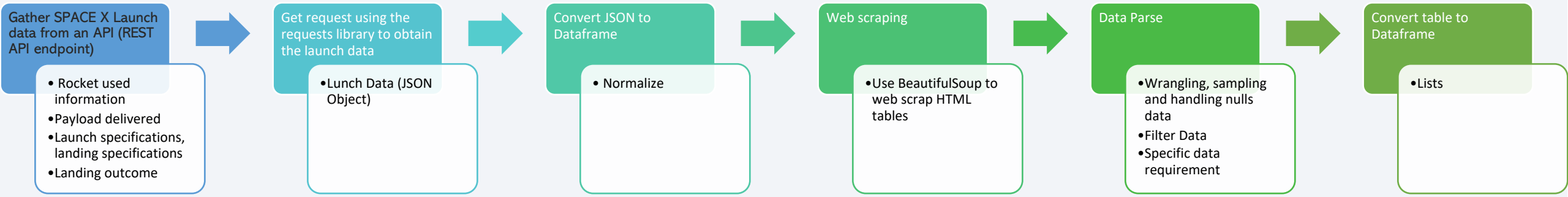
Methodology

Methodology

Executive Summary

- Data collection methodology:
  - SpaceX REST API and Web scraping from wiki pages used for data gathering
- Perform data wrangling
  - Collected data in JSON object and HTML tables converted into dataframe for analysis and visualization
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - To determine if the first stage of Falcon 9 will land successfully using machine learning

Data Collection

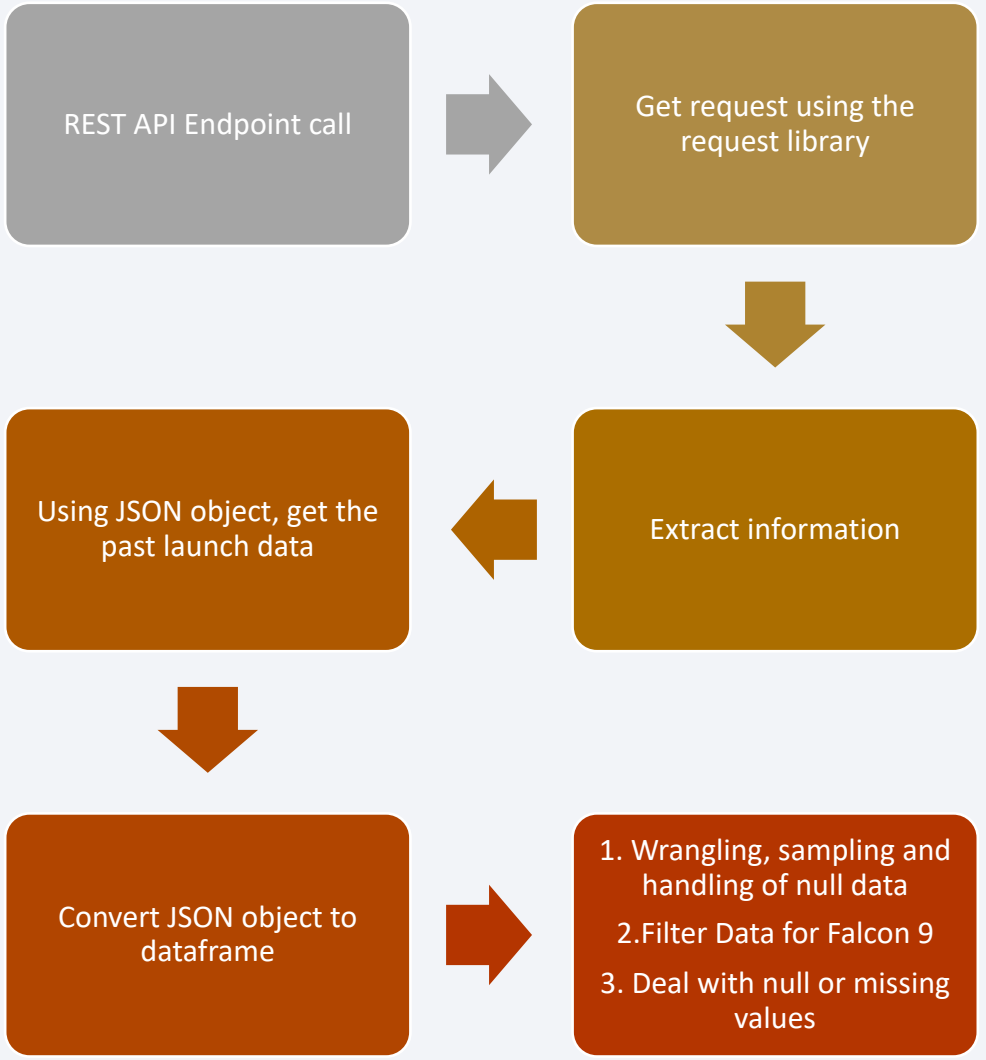


Data Collection – SpaceX API

GitHub URL :

Data Collection Notebook

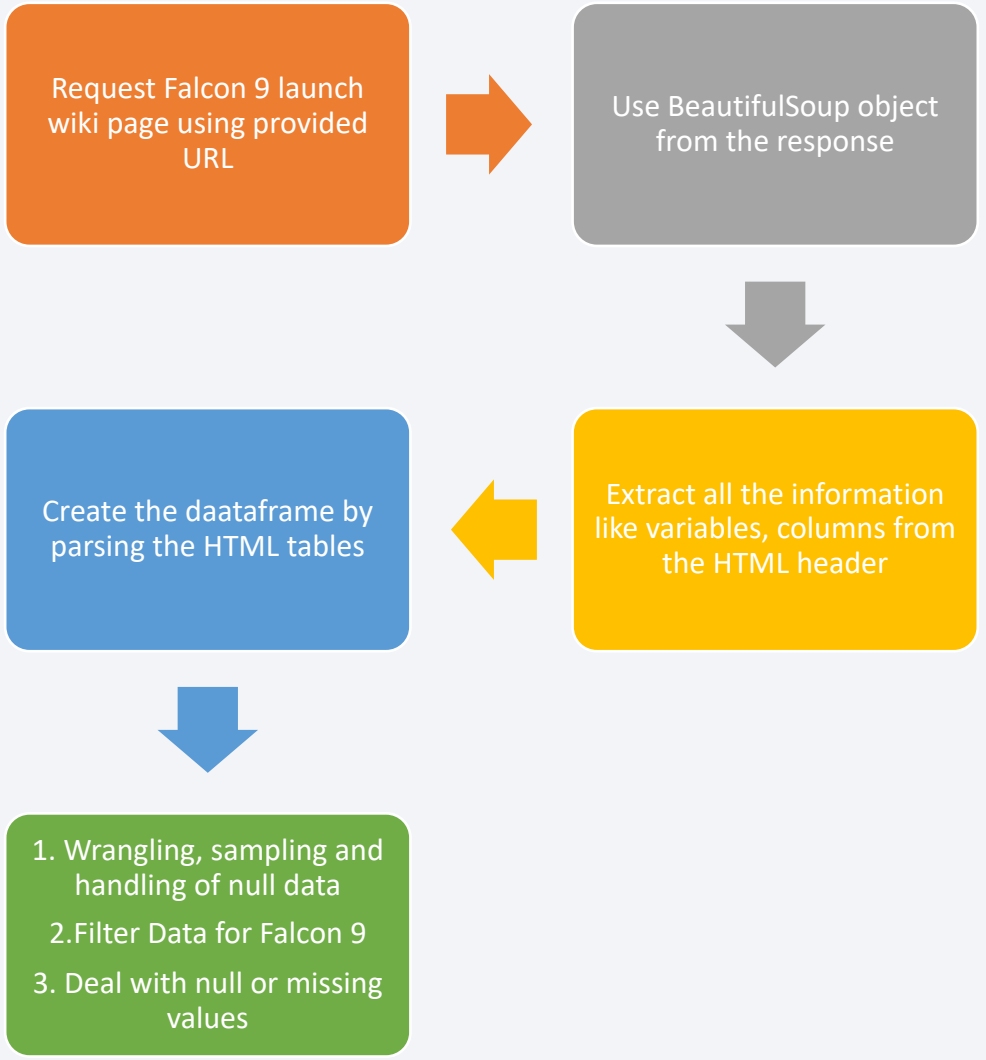
- Data collection with SpaceX REST calls



Data Collection - Scraping

GitHub URL– Data collection - scraping

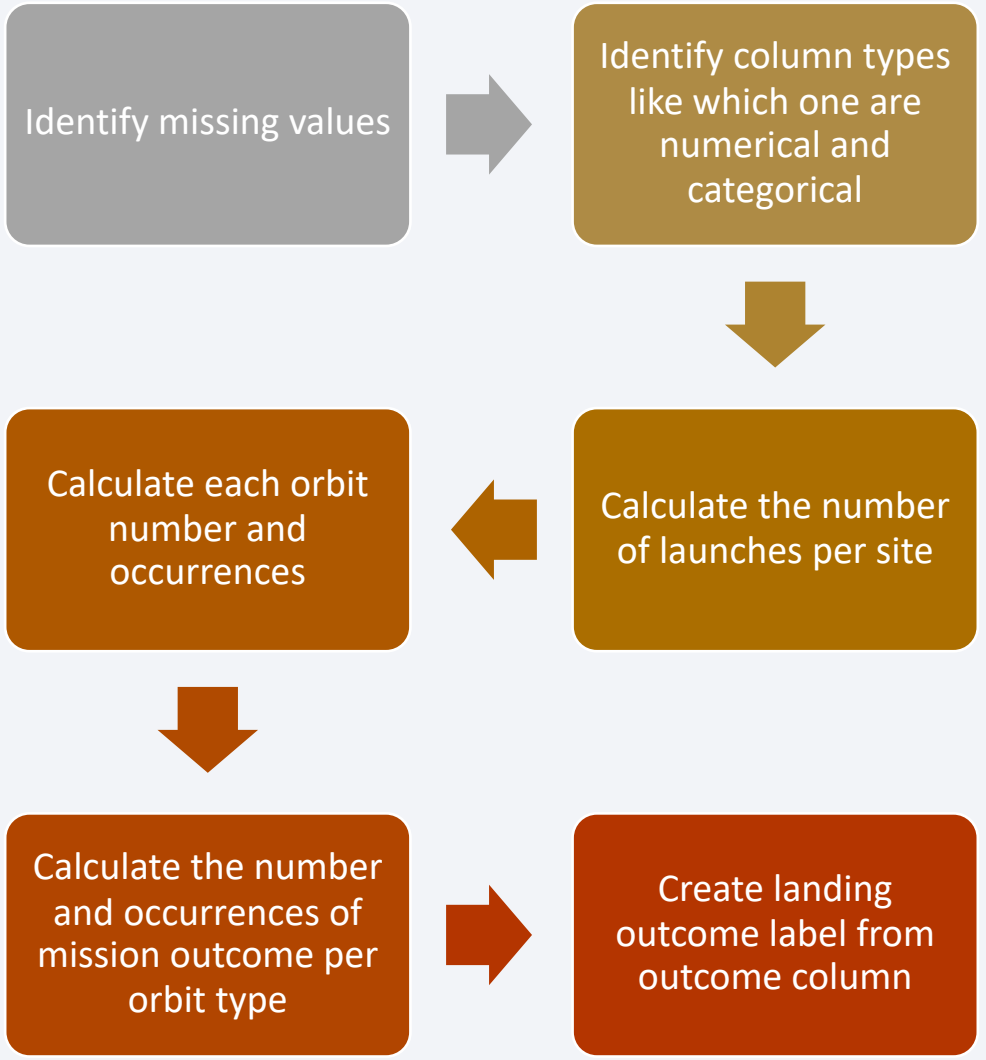
- Web scraping process



Data Wrangling

GitHub URL – Data Wrangling

- To find data patterns, perform Exploratory Data Analysis (EDA)



EDA with Data Visualization

GitHub URL– EDA with visualization

- Summarize of charts plotted
  - Catplot to visualize the relationship between
    - Flight number and Payload
    - Flight number and launch
    - Payload and Launch site
    - Flight number and orbit type
    - Payload and orbit type
  - Bar Chart to visualize the relationship between success rate of each orbit type
  - Line chart to visualize the launch success trend (yearly)

EDA with SQL

GitHub URL–
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_) FROM SPACEXTBL WHERE CUSTOMER='NASA (CRS)'
    ```

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

    ```
    SELECT AVG(PAYLOAD_MASS__KG_) FROM SPACEXTBL WHERE BOOSTER_VERSION='F9 v1.1'
    ```

  - **List the date when the first successful landing outcome in ground pad was achieved**

    ```
    SELECT min(DATE) FROM SPACEXTBL WHERE LANDING_OUTCOME='Success (ground pad)'
    ```

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

    ```
    SELECT BOOSTER_VERSION FROM SPACEXTBL WHERE PAYLOAD_MASS__KG_ between 4000 and 6000 AND
    LANDING_OUTCOME='Success (drone ship)'
    ```

EDA with SQL (Continued)

GitHub URL–
EDA with SQL

- **List the total number of successful and failure mission outcomes**

```
SELECT COUNT(*) FROM SPACEXTBL WHERE MISSION_OUTCOME LIKE '%Success%' OR MISSION_OUTCOME LIKE '%Failure%’
```

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

```
SELECT BOOSTER_VERSION FROM SPACEXTBL WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTBL)
```

- **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.**

```
SELECT LANDING_OUTCOME, BOOSTER_VERSION, LAUNCH_SITE FROM SPACEXTBL WHERE Landing_Outcome = 'Failure (drone ship)' AND CAST(SUBSTR(DATE, 7, 4) AS integer) = 2015 ’
```

- **Rank the count of successful landing_outcomes between the date 04-06-2010 and 20-03-2017 in descending order.**

```
SELECT LANDING_OUTCOME, COUNT(LANDING_OUTCOME) AS TOTAL_NUMBER \ FROM SPACEXTBL \ WHERE DATE BETWEEN '04-06-2010' AND '20-03-2017' \ GROUP BY LANDING_OUTCOME \ ORDER BY TOTAL_NUMBER DESC
```

Build an Interactive Map with Folium

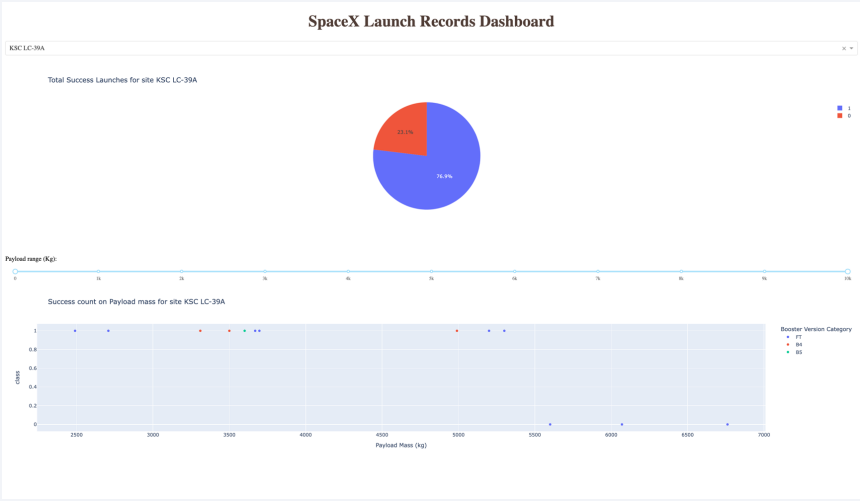
GitHub URL– Interactive_Map_with_Folium

- Summarize what map objects
  - folium.Circle and folium.Marker - To add highlighted circle area with text label on a specific coordinate for each launch site
  - MousePosition - To get coordinate for a mouse point
  - MarkerCluster – To simplify map containing many markers with same coordinate
  - Folium.PolyLine – To draw a line between launch site to its closest city, railway and highway

Build a Dashboard with Plotly Dash

GitHub URL– Dashboard_with_plotly

- Plotly Dash to perform interactive visual analytics on SpaceX launch data in real-time.
  - Added a Launch Site Drop-down Input Component
  - Added a callback function to render success-pie-chart based on selected site dropdown
  - Added a Range Slider to Select Payload
  - Added a callback function to render the success-payload-scatter-chart scatter plot



- Dashboard helps in finding
  - Which site has the largest successful launches?
  - Which site has the highest launch success rate?
  - Which payload range(s) has the highest launch success rate?
  - Which payload range(s) has the lowest launch success rate?
  - Which F9 Booster version (v1.0, v1.1, FT, B4, B5, etc.) has the highest launch success rate?

Predictive Analysis (Classification)

GitHub URL–
Predictive_Analysis

- Summary of the model development process
  - NumPy array creation from the class in data
  - Data Standardization
  - Train_test_Split to split the data X and Y into training and test data
  - Searching for the best Hyperparameters for logistics regression, SVM, decision Tree and KNN classifiers
  - Searching for method that performs best using data

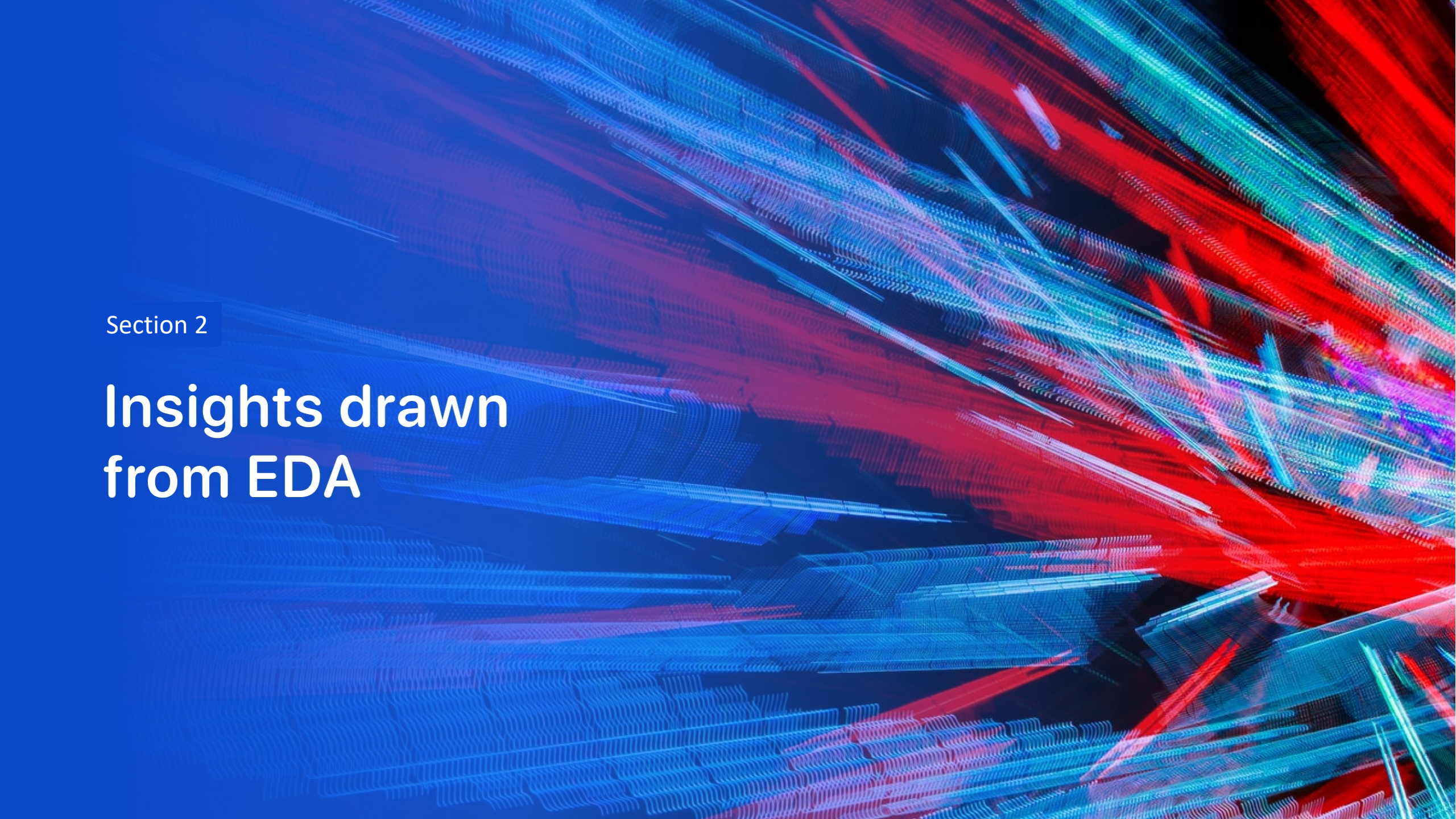
Predictive Analysis (Classification)

GitHub URL–
Predictive_Analysis

- Summary of the model development process
  - Created a NumPy array from the column Class in data, by applying the method to_numpy() then assigned it to the variable Y, made sure the output is a Pandas series (only one bracket df['name of column']).
  - Standardized the data in X then reassigned it to the variable X using the transform provided
  - Used the function train_test_split to split the data X and Y into training and test data. Set the parameter test_size to 0.2 and random_state to 2. The training data and test data was assigned
  - Created a logistic regression object then created a GridSearchCV object logreg_cv with cv = 10. Fitted the object to find the best parameters from the dictionary parameters.
  - Calculated the accuracy on the test data using the method score
  - Create a support vector machine object then create a GridSearchCV object svm_cv with cv - 10. Fitted the object to find the best parameters from the dictionary parameters.
  - Calculated the accuracy on the test data using the method score
  - Created a decision tree classifier object then create a GridSearchCV object tree_cv with cv = 10. Fitted the object to find the best parameters from the dictionary parameters.
  - Calculate the accuracy of tree_cv on the test data using the method score
  - Created a k nearest neighbors object then create a GridSearchCV object knn_cv with cv = 10. Fit the object to find the best parameters from the dictionary parameters.
  - Calculated the accuracy of knn_cv on the test data using the method score
  - Found the method performs best

Results

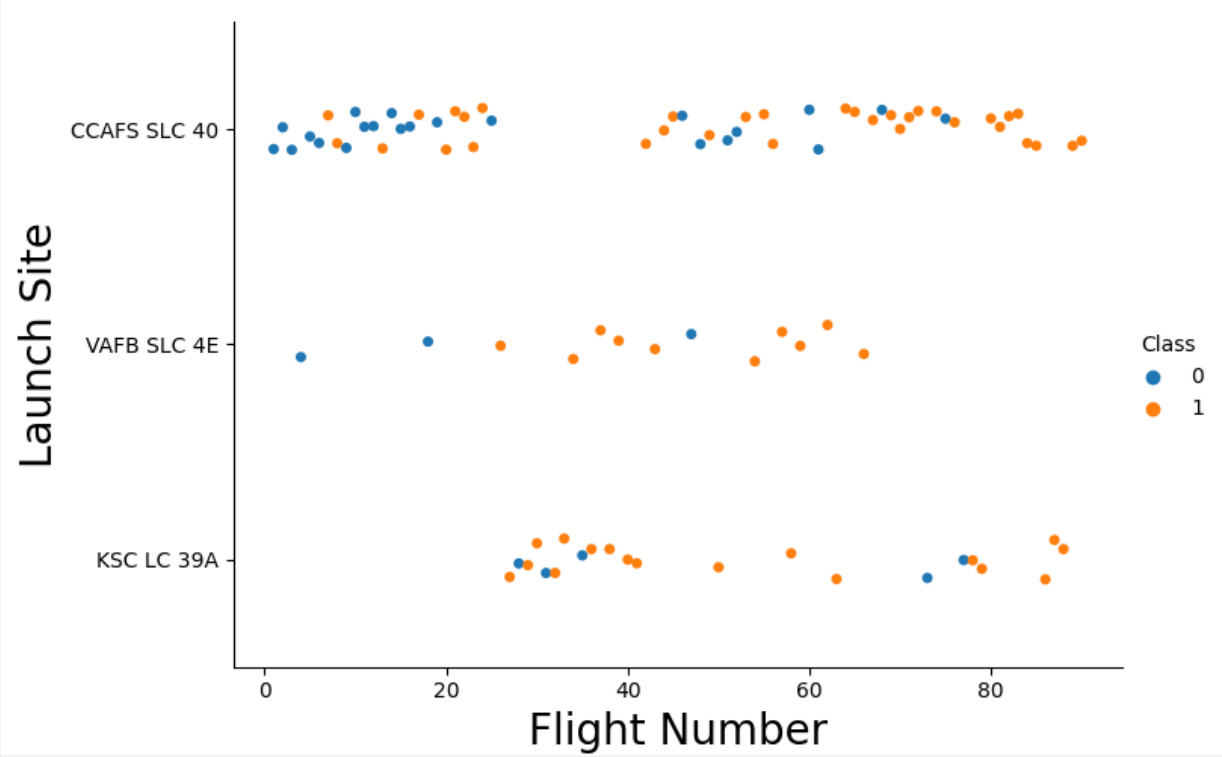
- Exploratory data analysis results
- Interactive analytics demo in screenshots
- Predictive analysis results


Section 2
Insights drawn from EDA

Flight Number vs. Launch Site

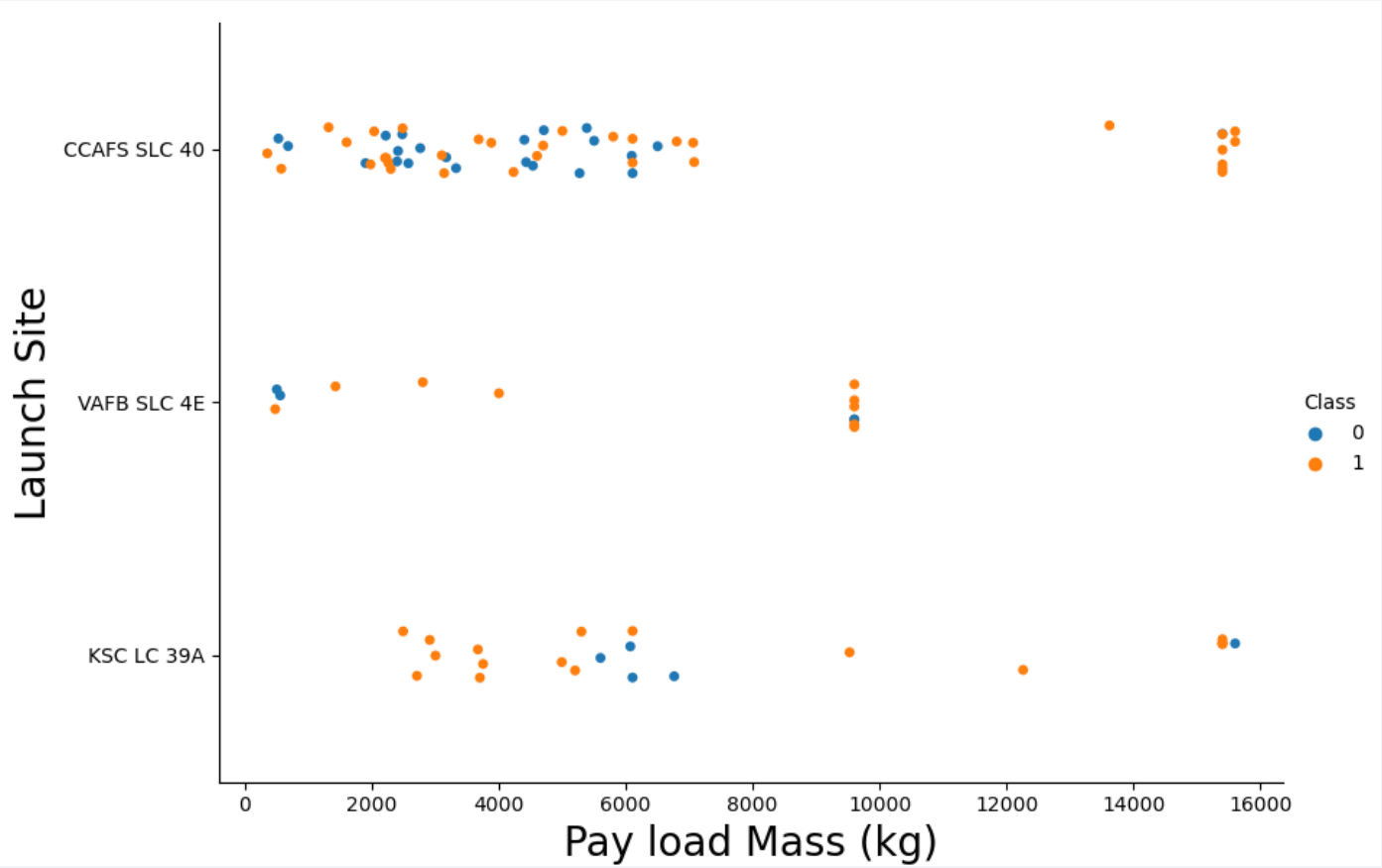
We see that different launch sites have different success rates.

CCAFS LC-40, has a success rate of 60 %, while KSC LC-39A and VAFB SLC 4E has a success rate of 77%.



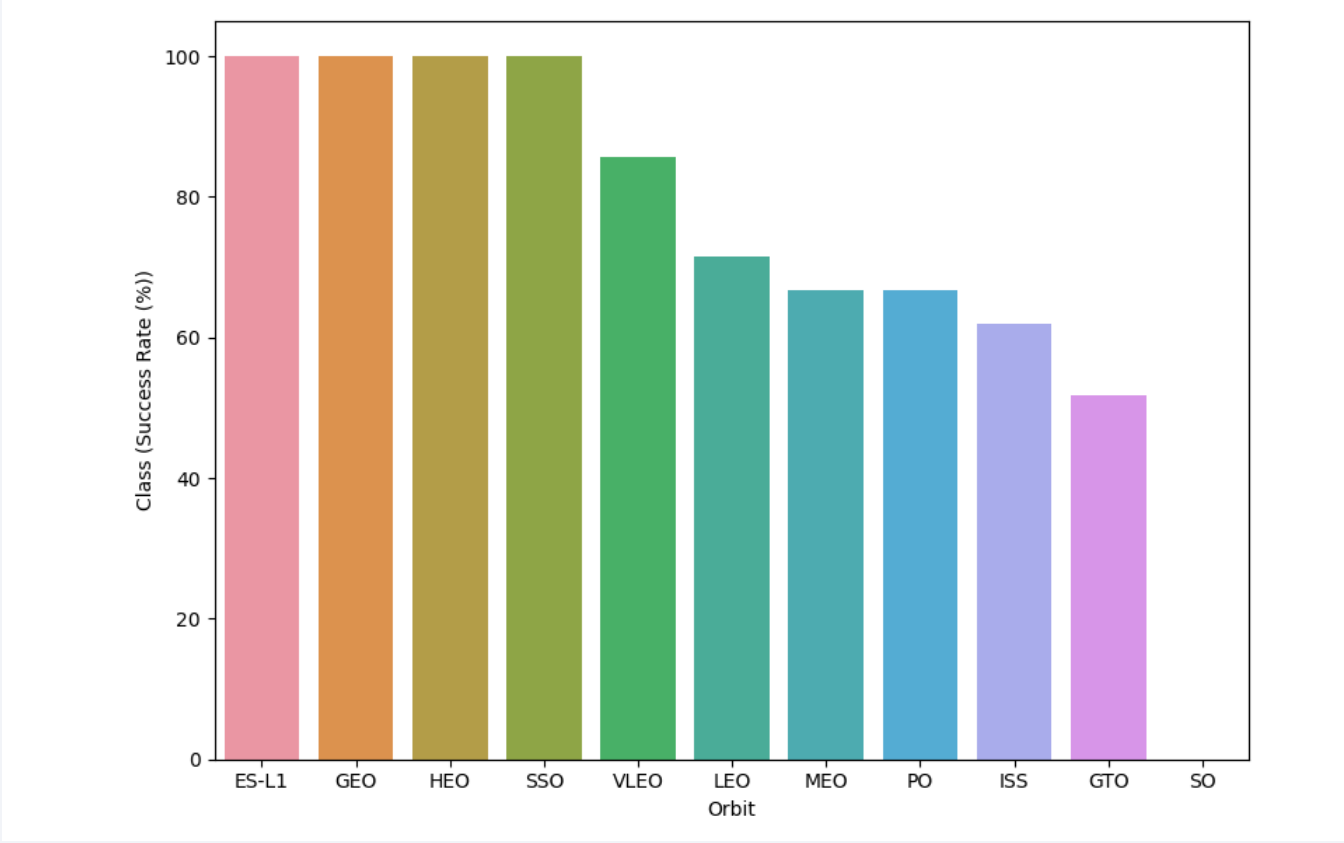
Payload vs. Launch Site

- We observed there is relationship between launch sites and their payload mass.
- VAFB-SLC launchsite, there are no rockets launched for heavypayload mass (greater than 10000).
- CCAFS SLC launchsite there are rockets less than 5500kg and more than 13000kg payloads but not in between



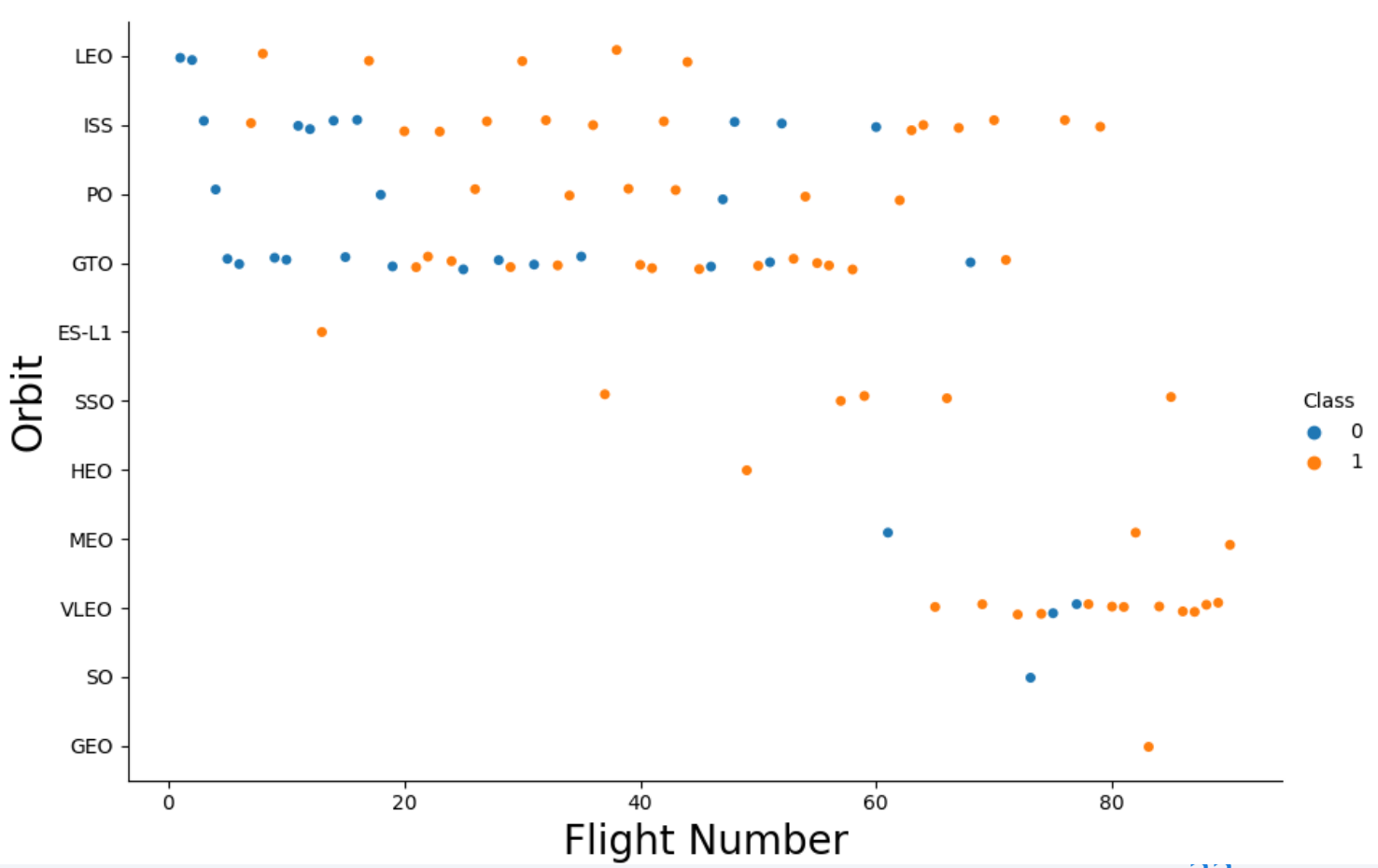
Success Rate vs. Orbit Type

- First 4 orbits types has the best successful rate (ES-L1, GEO, HEO and SSO) compared to GTO



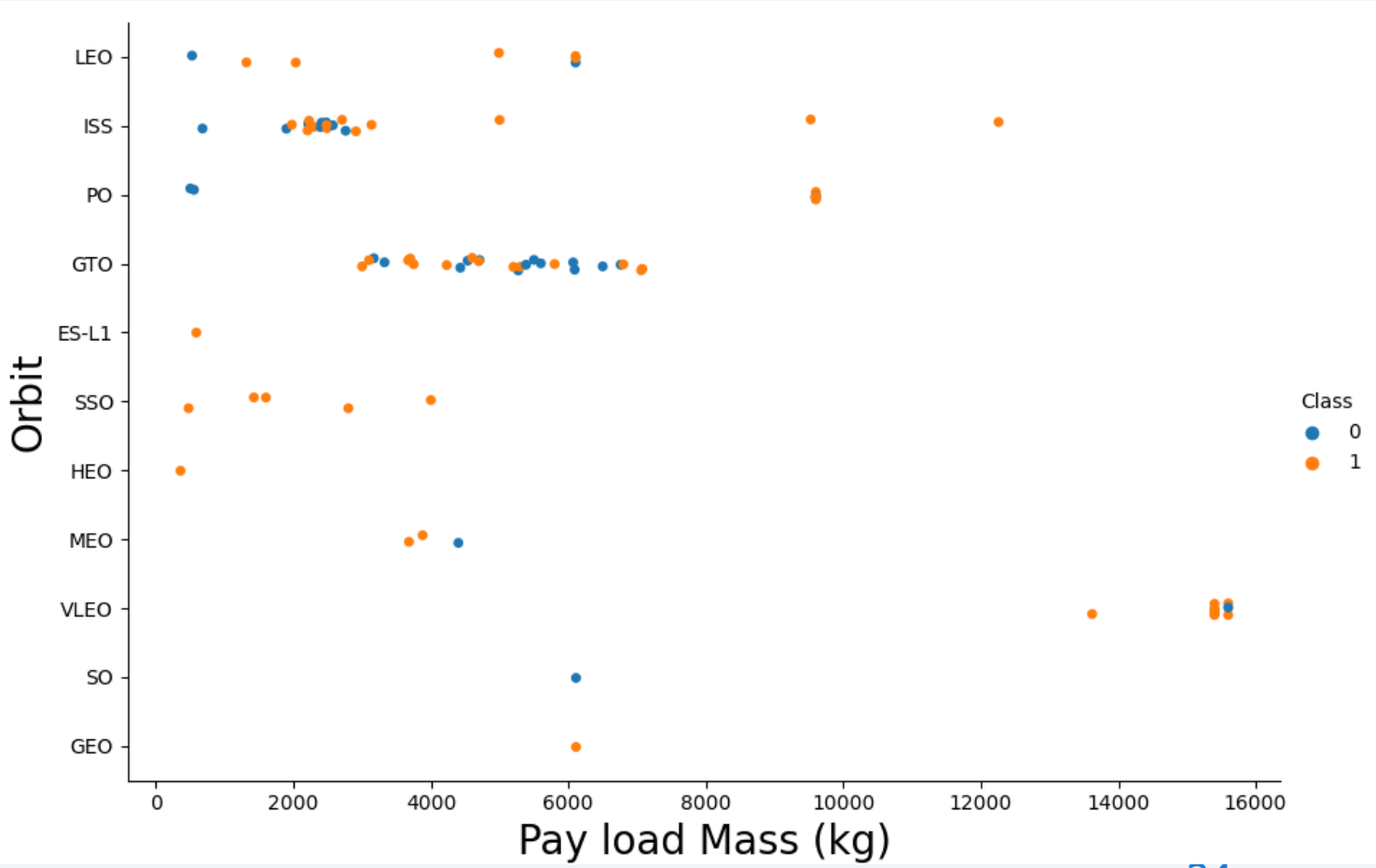
Flight Number vs. Orbit Type

- For LEO orbit the Success appears related to the number of flights; on the other hand, there seems to be no relationship between flight number when in GTO orbit
- The orbits with higher successful rate has one or few number of launches
- There are more failures at beginning however success ratio improves over the period



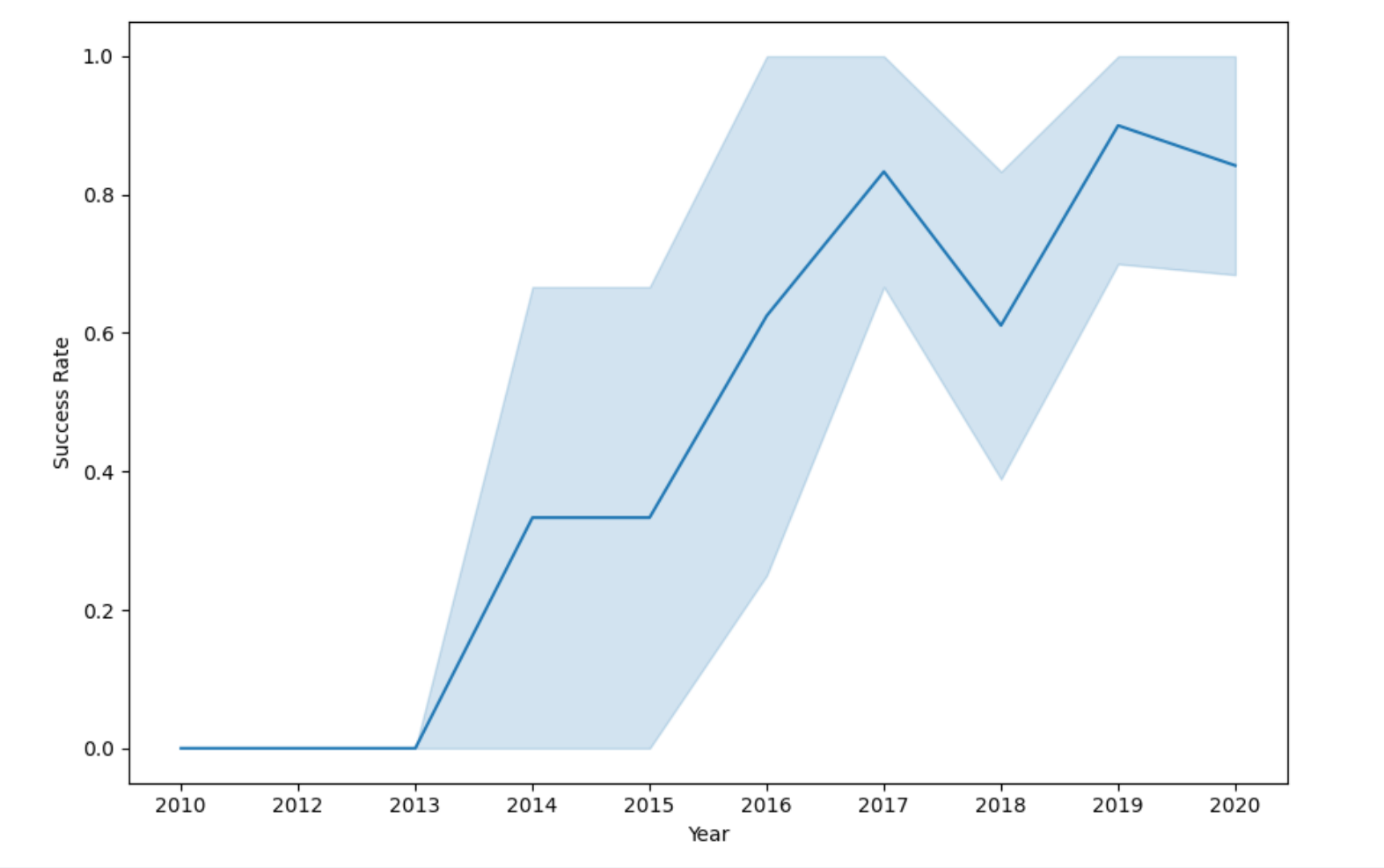
Payload vs. Orbit Type

- 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(unsuccessful mission) are both there here.



Launch Success Yearly Trend

- Success rate shown drastically increasing since 2013 till 2020



All Launch Site Names

- There are four unique launch sites in the space mission – SQL using DISTINCT

```
In [7]: %sql SELECT DISTINCT LAUNCH_SITE FROM SPACEXTBL;
 * sqlite:///my_data1.db
Done.
```

Out[7]:

Launch_Site
CCAFS LC-40
VAFB SLC-4E
KSC LC-39A
CCAFS SLC-40
None

Launch Site Names Begin with 'CCA'

- Where, Like and Limit used to get 5 launch sites with string CCA

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

In [8]:
```
%sql SELECT * FROM SPACEXTBL WHERE LAUNCH_SITE LIKE 'CCA%' LIMIT 5
```

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

Out[8]:

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Lan
06/04/2010	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0.0	LEO	SpaceX	Success	Fai
12/08/2010	15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight C1, two CubeSats, barrel of Brouere cheese	0.0	LEO (ISS)	NASA (COTS) NRO	Success	Fai
22/05/2012	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525.0	LEO (ISS)	NASA (COTS)	Success	
10/08/2012	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500.0	LEO (ISS)	NASA (CRS)	Success	
03/01/2013	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677.0	LEO (ISS)	NASA (CRS)	Success	

Total Payload Mass

- Total payload mass carried by boosters launched by NASA (CRS) - SUM function with WHERE clause

Task 3

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

```
In [9]: %sql SELECT SUM(PAYLOAD_MASS__KG_) FROM SPACEXTBL WHERE CUSTOMER='NASA (CRS)'
 * sqlite:///my_data1.db
Done.
```

Out[9]:

SUM(PAYLOAD_MASS__KG_)
45596.0

Average Payload Mass by F9 v1.1

- Average payload mass carried by booster using F9 V1.1 – AVG function

Task 4

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

```
In [10]: %sql SELECT AVG(PAYLOAD_MASS__KG_) FROM SPACEXTBL WHERE BOOSTER_VERSION='F9 v1.1'
```

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

Out[10]:

AVG(PAYLOAD_MASS__KG_)
2928.4

First Successful Ground Landing Date

- First successful landing outcome using MIN function

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

Hint:Use min function

```
In [19]: %sql SELECT min(DATE) FROM SPACEXTBL WHERE LANDING_OUTCOME='Success (ground pad)'
```

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

Out[19]:

min(DATE)
01/08/2018

Successful Drone Ship Landing with Payload between 4000 and 6000

- Using BETWEEN, WHERE and AND operator, names of the boosters which have success in drone ship and have payload mass greater than 4000 and 6000 derived

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

In [20]:
```
%sql SELECT BOOSTER_VERSION FROM SPACEXTBL WHERE PAYLOAD_MASS__KG_ between 4000 and 6000 AND LANDING_OUTCOME='Suc
```

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

Out[20]:

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

Total Number of Successful and Failure Mission Outcomes

- Total number of successful and failure mission outcomes using COUNT and GROUP BY

List the total number of successful and failure mission outcomes

In [13]:
```
%sql SELECT COUNT(*) FROM SPACEXTBL WHERE MISSION_OUTCOME LIKE '%Success%' OR MISSION_OUTCOME LIKE '%Failure%'
```

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

Out[13]:

COUNT(*)
101

Boosters Carried Maximum Payload

- Names of boosters which carried maximum payload using COUNT and GROUP BY function

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

```
In [14]: %sql SELECT BOOSTER_VERSION FROM SPACEXTBL WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTB
 * sqlite:///my_data1.db
Done.
```

Out[14]:

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

- For year 2015, the failure landing outcomes in drone ship, booster versions and launch site derived using LIKE

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, 4, 2) as month to get the months and substr(Date,7,4)='2015' for year.

In [40]:
```
%sql SELECT LANDING_OUTCOME, BOOSTER_VERSION, LAUNCH_SITE FROM SPACEXTBL WHERE Landing_Outcome = 'Failure (drone
```

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

Out[40]:

Landing_Outcome	Booster_Version	Launch_Site
Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

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

- Count of successful landing between the dates derived using GROUP BY, COUNT and HAVING

Rank the count of successful landing_outcomes between the date 04-06-2010 and 20-03-2017 in descending order.

In [41]:
```
%sql SELECT LANDING_OUTCOME, COUNT(LANDING_OUTCOME) AS TOTAL_NUMBER \
FROM SPACEXTBL \
WHERE DATE BETWEEN '04-06-2010' AND '20-03-2017' \
GROUP BY LANDING_OUTCOME \
ORDER BY TOTAL_NUMBER DESC
```

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

Out[41]:

Landing_Outcome	TOTAL_NUMBER
Success	20
No attempt	10
Success (drone ship)	8
Success (ground pad)	7
Failure (drone ship)	3
Failure	3
Failure (parachute)	2
Controlled (ocean)	2
No attempt	1

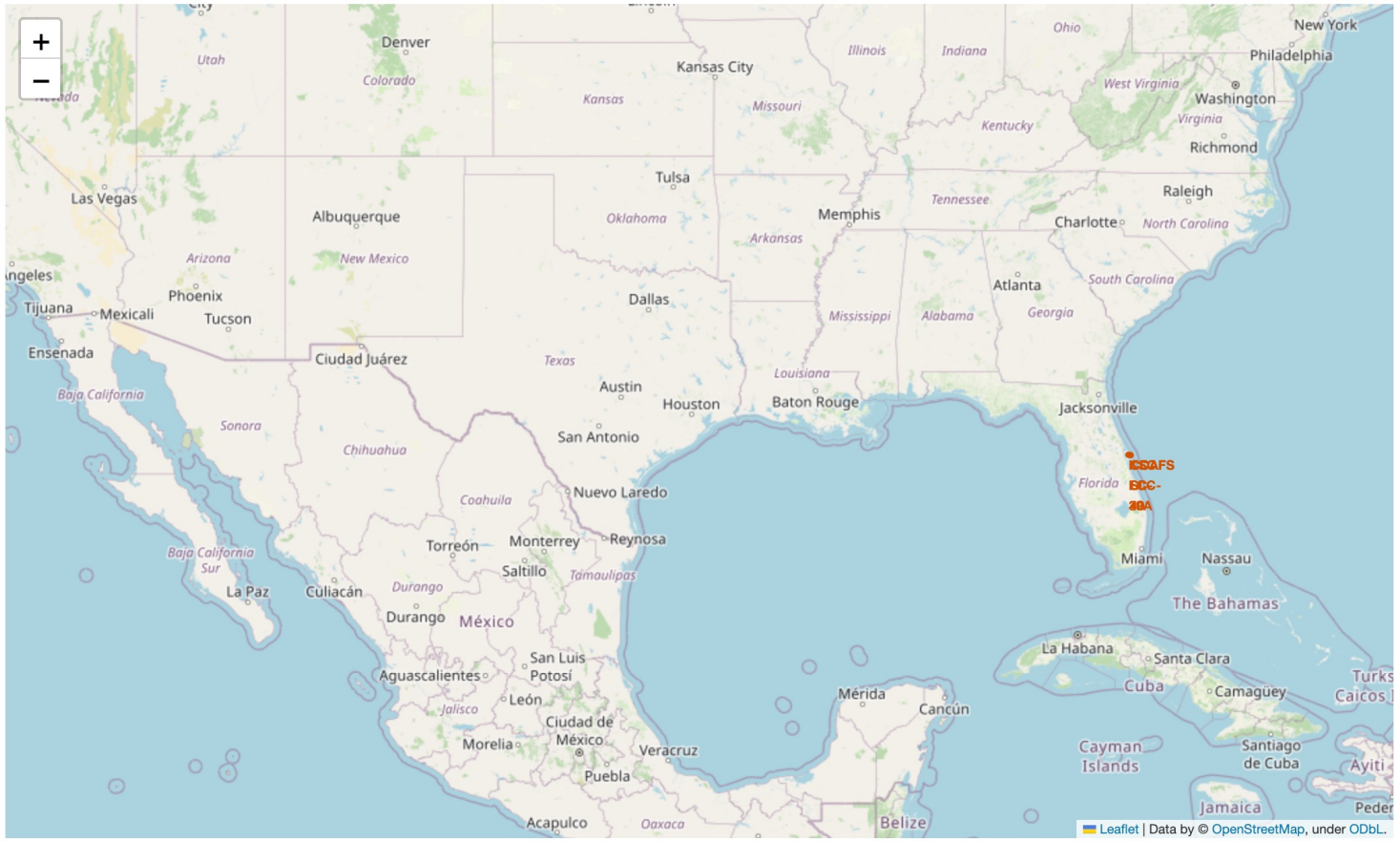

Section 3
Launch Sites
Proximities Analysis

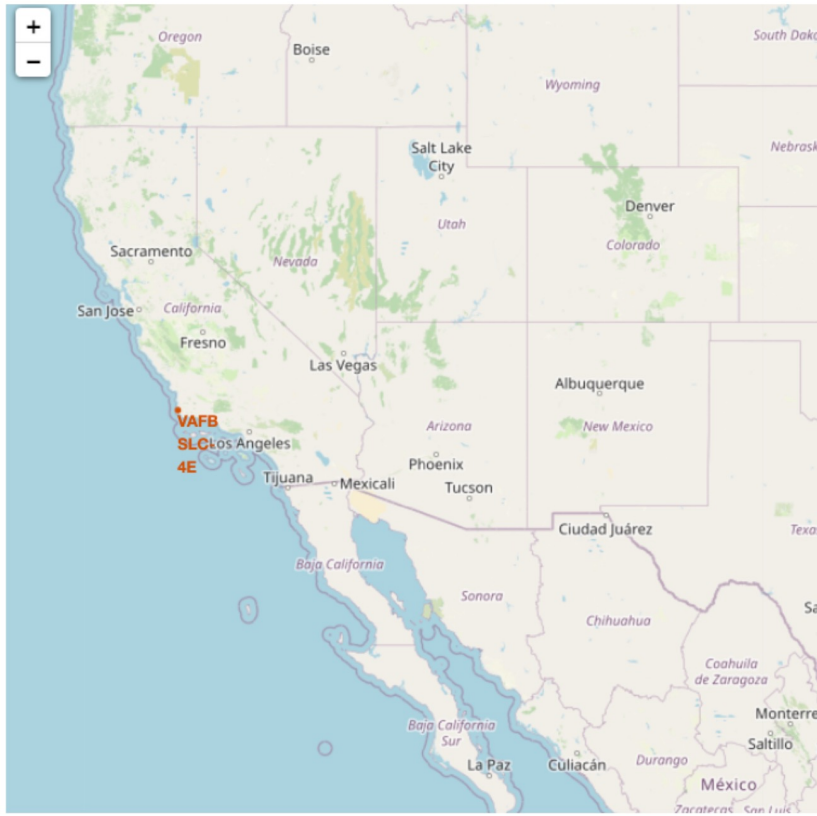
All Launch Sites

NBViewer Link to see the maps

- All launch sites are in close proximity to the coast and are in restricted area

Out[14]:

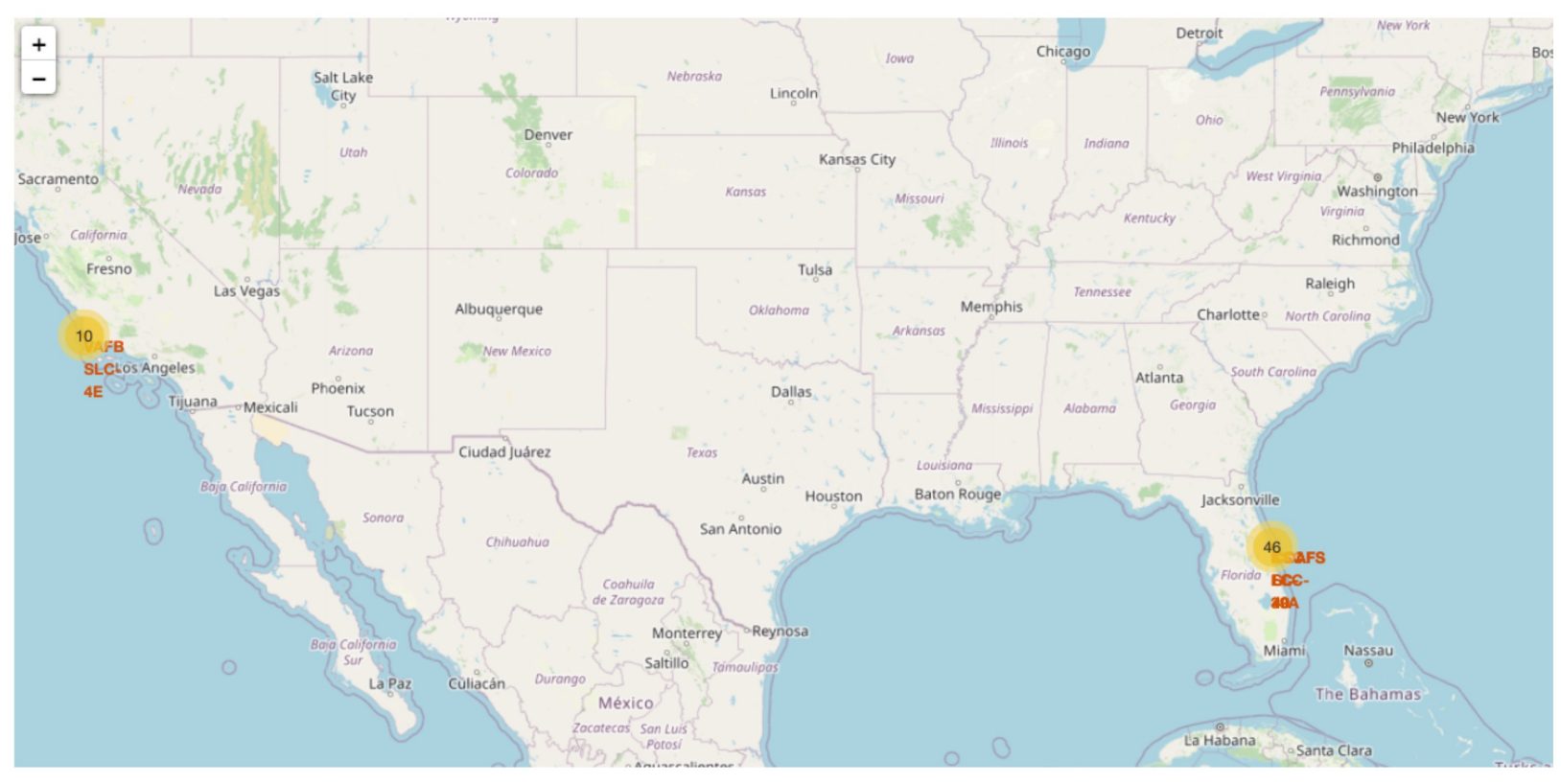


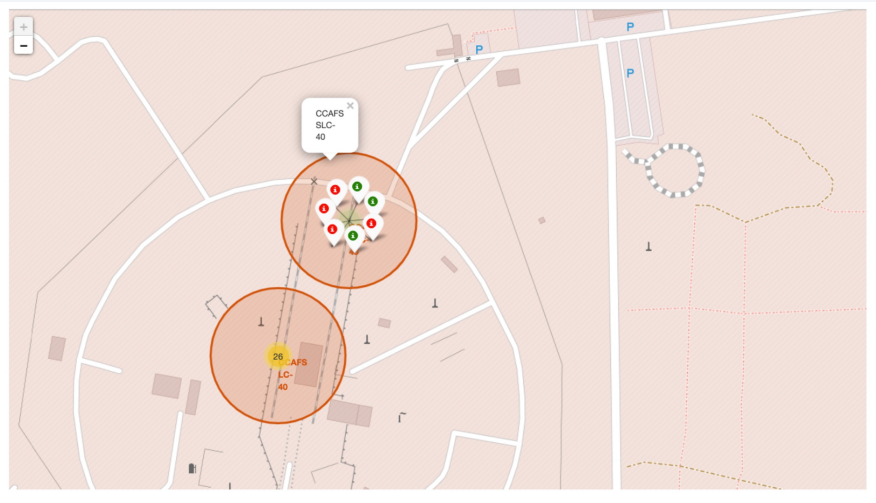


Success/Failed Launches for Each Site

NBViewer Link to see the maps

1. Clusters for every launch site
2. Green Marker if launch was successful and red marker if launch was failed

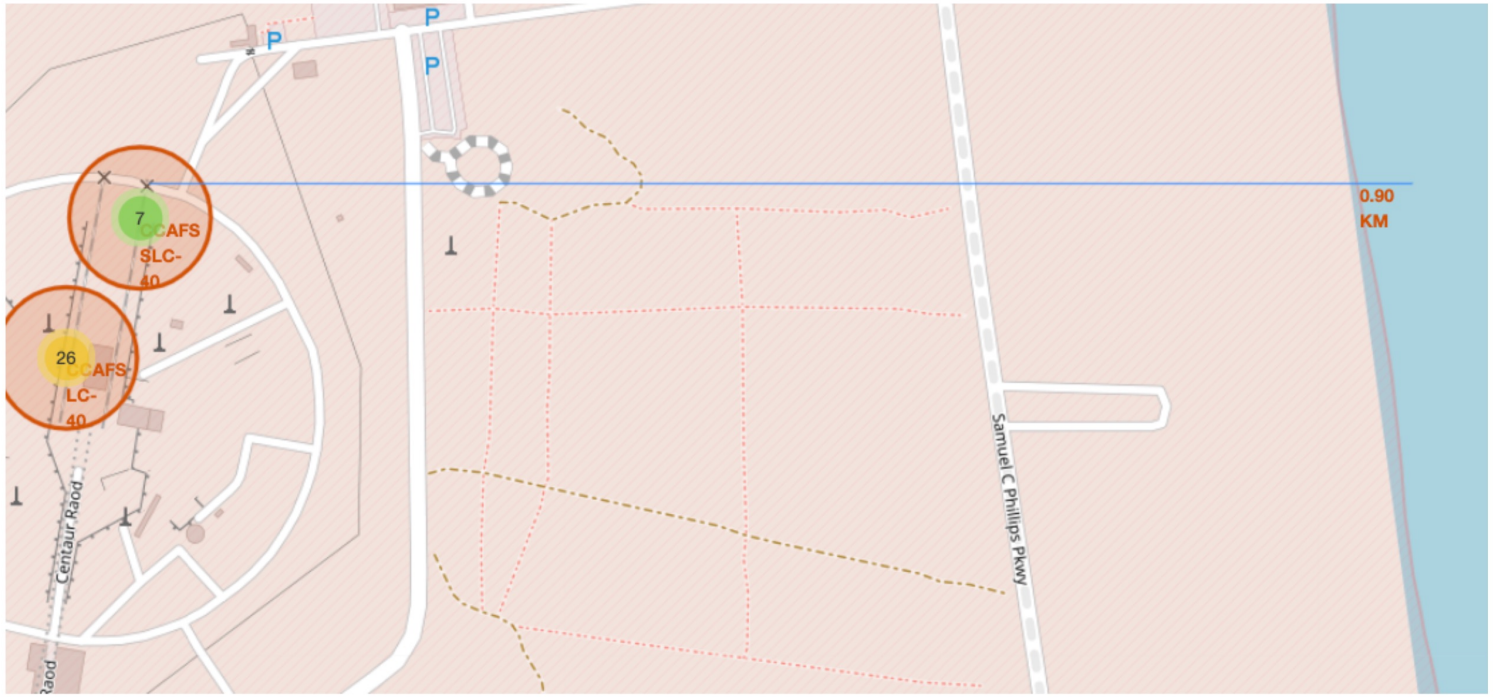




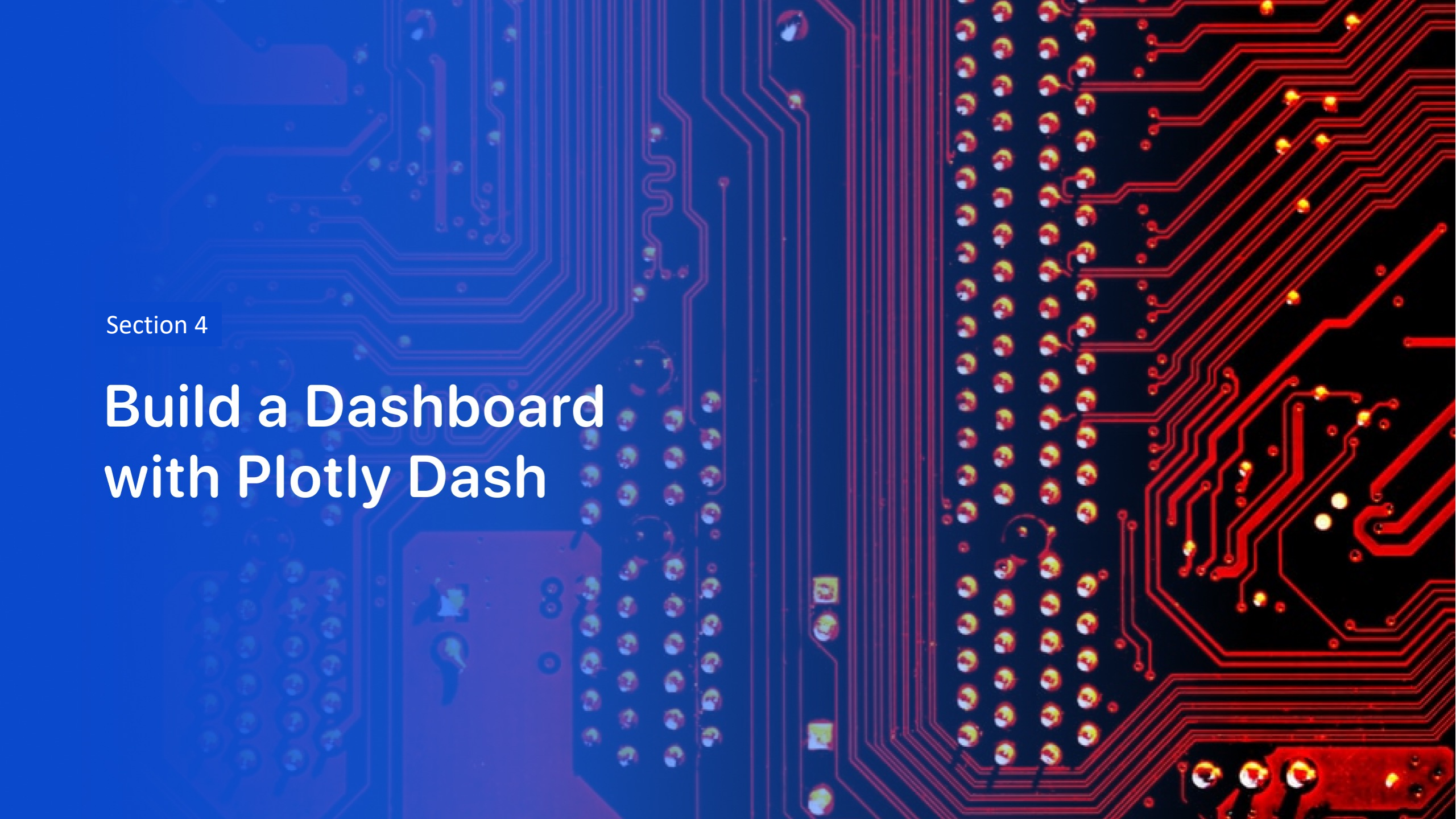
Launch Sites Proximities

NBViewer Link to see the maps

- Launch sites are near to coastlines, railways, roads and highways.



TODO: Similarly, you can draw a line betwee a launch site to its closest city, railway, highway, etc. You need to use `MousePosition` to find the their coordinates on the map first

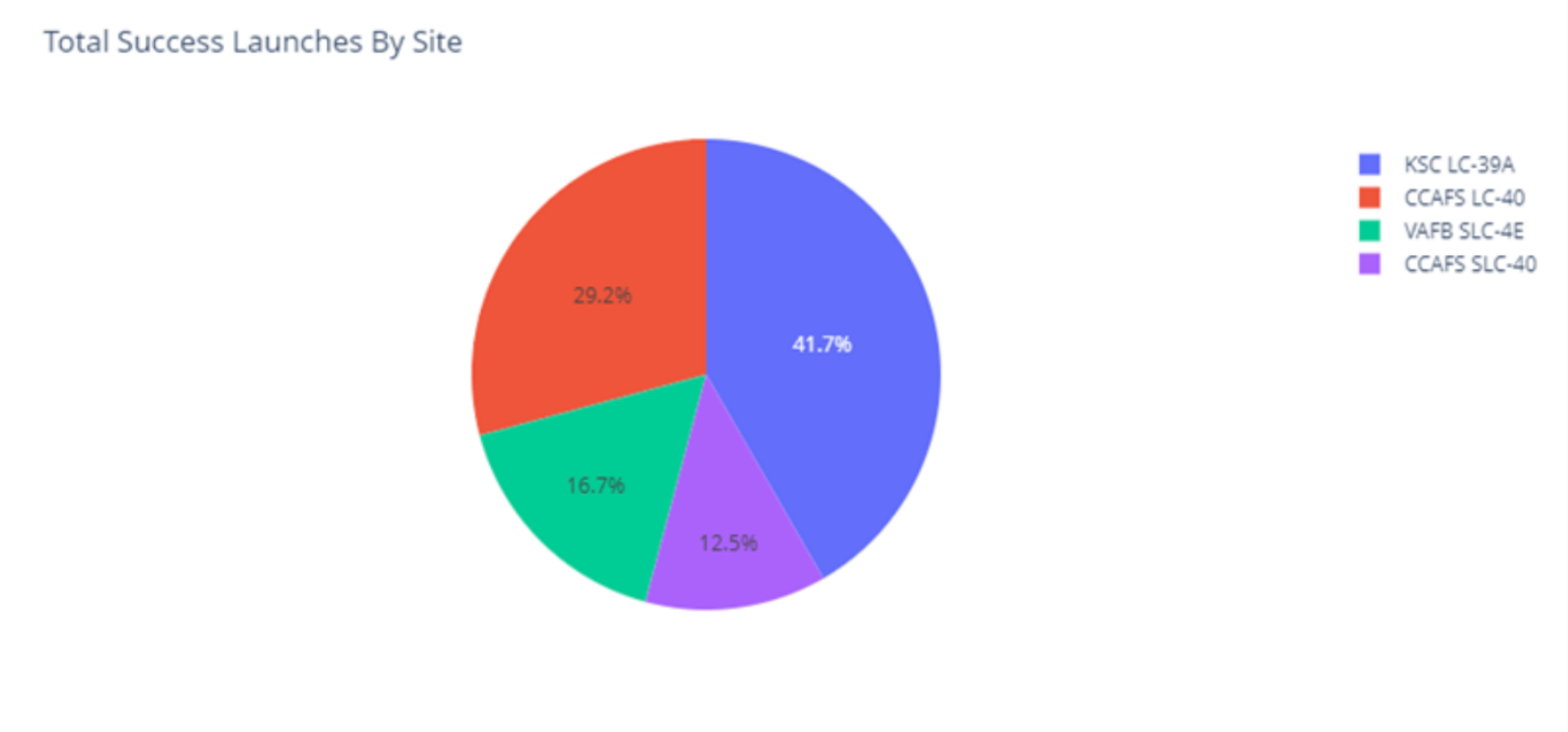


Section 4

Build a Dashboard with Plotly Dash

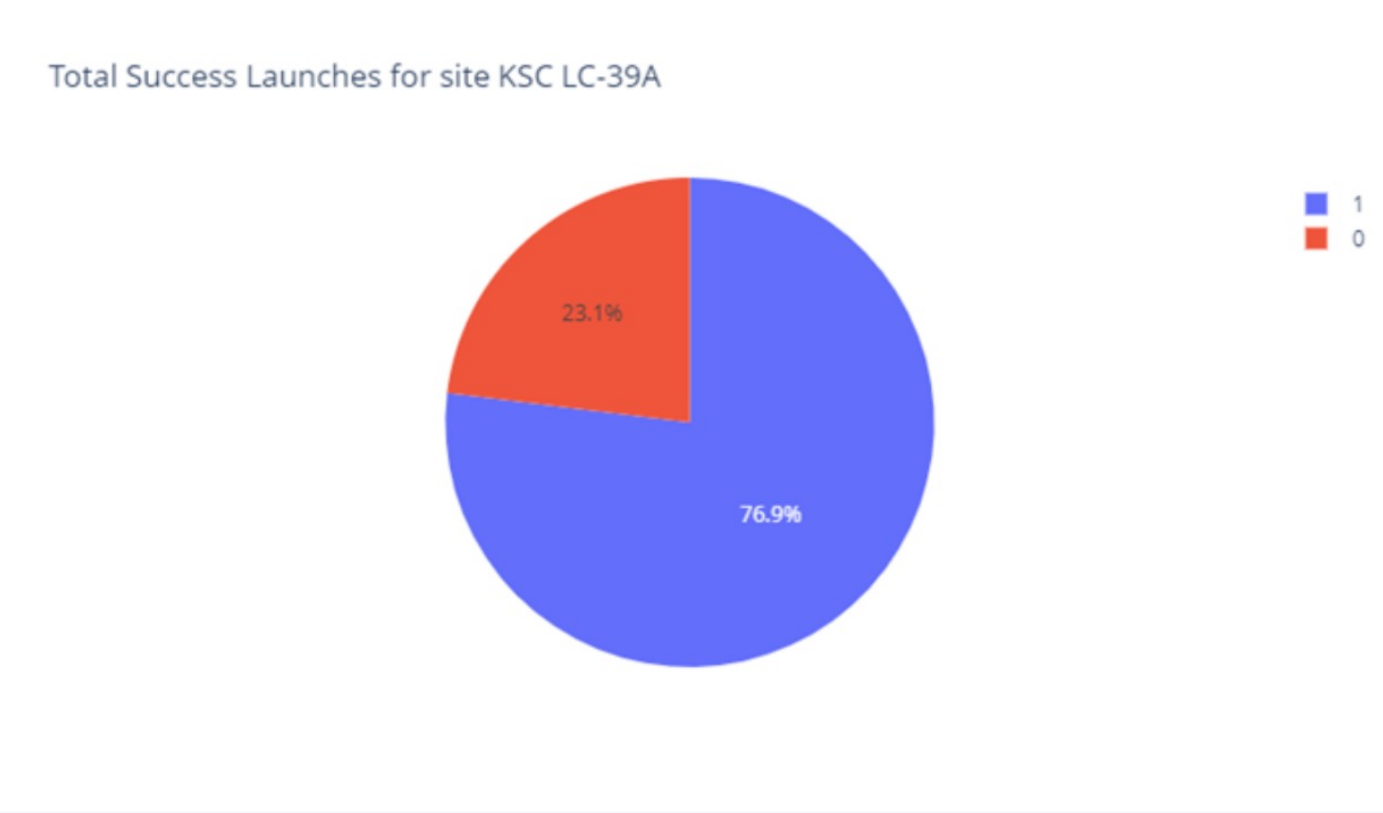
Total Success Launches by Sites

- KSC LC-39A is the site with higher success rate of launches



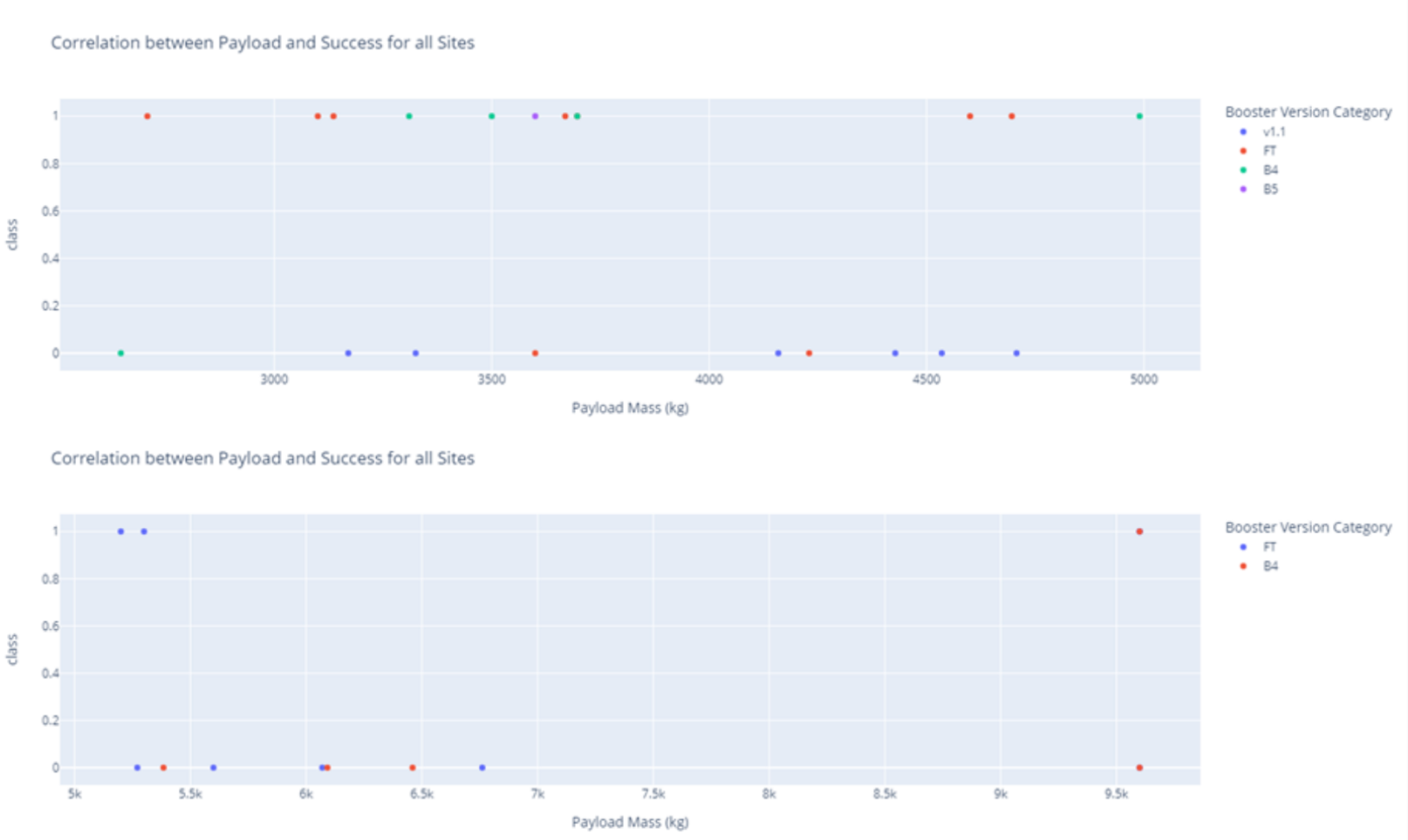
Total Launches for site KSC LC-39A

- KSC LC-39A total success launches with launch success rate of 77%



Payload vs launch outcome

- Correlation between payload and success

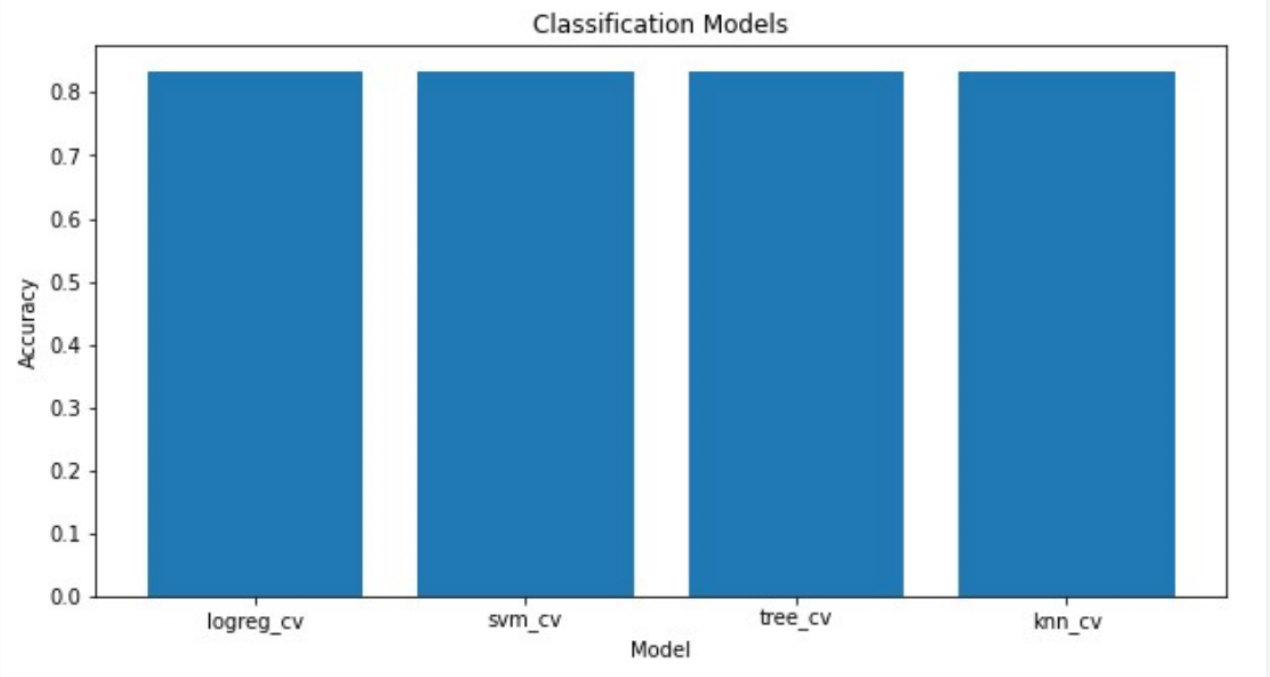


Section 5

Predictive Analysis (Classification)

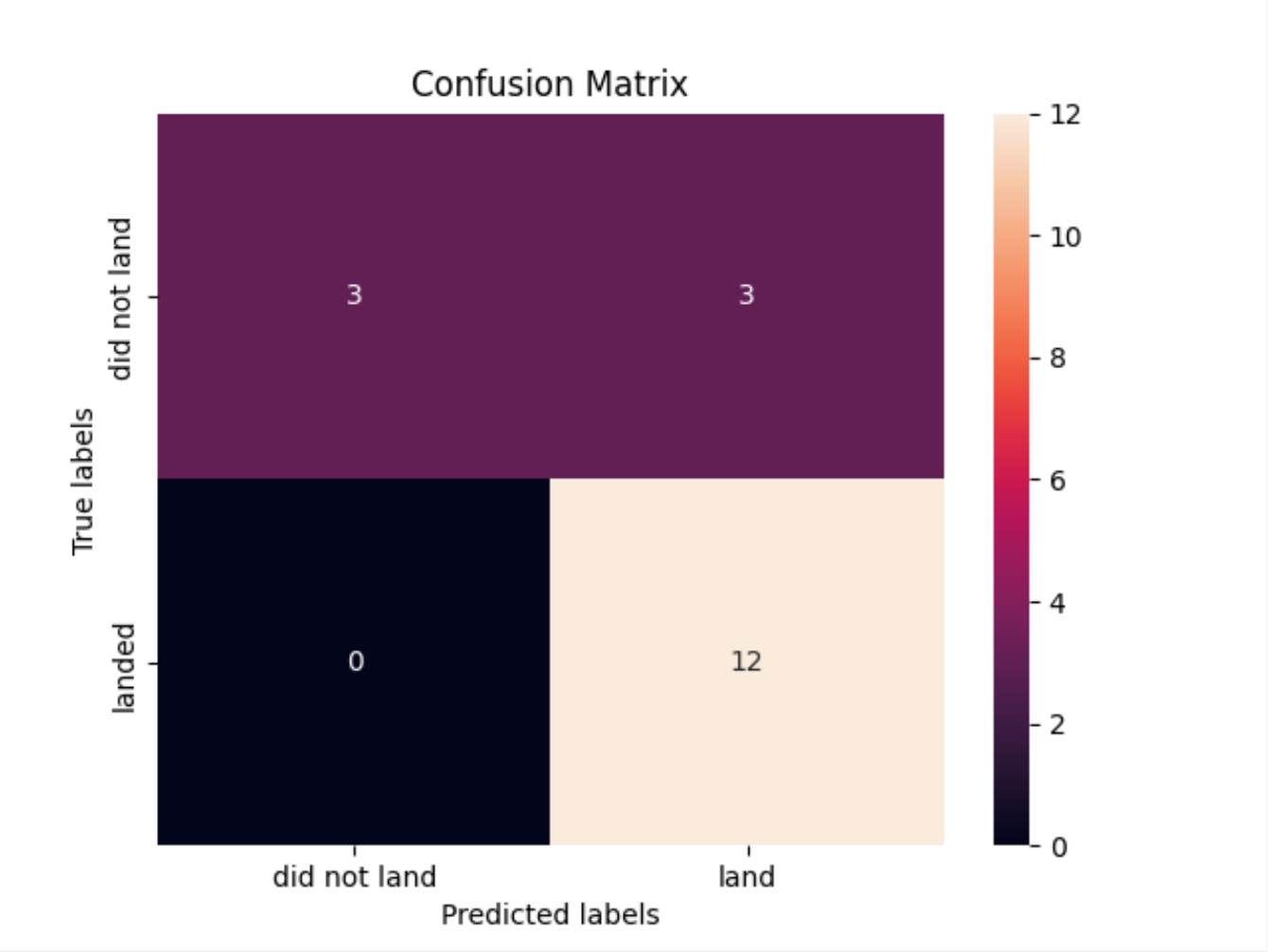
Classification Accuracy

- The accuracy is same with respect to all the models



Confusion Matrix

- The confusion matrix is same with respect to all the models



Conclusions

- Tune Hyperparameters giving best or same accuracy across models
- Data analysis and machine learnings techniques, we can determine the price of each launch. If the first stage land is successful, we gain the upper hand compare to our competitor
- Also reusability of first stage is determined with cost of a launch and its success of launch and landing

Appendix

- Refer to Github for more details on notebooks and scripts

Link

Thank you!