IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

<Name>
<Date>

Outline

- Executive Summary
- Introduction
- Methodology
- Results
- Conclusion

Executive Summary

- **Summary of methodologies**
  - Data collection using SpaceX API
  - Data collection using Web Scraping from Wikipedia
  - Data Wrangling
  - Exploratory Data Analysis using SQL queries
  - Exploratory Data Analysis using Plots/Visualizations
  - Interactive Location Maps using Folium
  - Dashboard with Plotly Dash
  - Prediction using Machine Learning
- **Summary of all results**
- Results from Exploratory Data Analysis
- Insights from Interactive Visualizations
- Predictive Analytics

Introduction

- **Project background and context**

Space X 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 because Space X can reuse the first stage. If we can determine if the first stage will land, we can determine the cost of a launch. This information can be used if an alternate company wants to bid against space X for a rocket launch. In this lab, you will create a machine learning pipeline to predict if the first stage will land given the data from the preceding labs.

- **Problems you want to find answers**
    - Factors that determine if the rocket will land successfully.
    - Interaction amongst various variables that determine the success rate of a successful landing.
    - Operating conditions that needs to be in place to ensure a successful landing program.
    - A reliable method for accurately predicting the outcomes of a successful launch.

Section 1

Methodology

Methodology

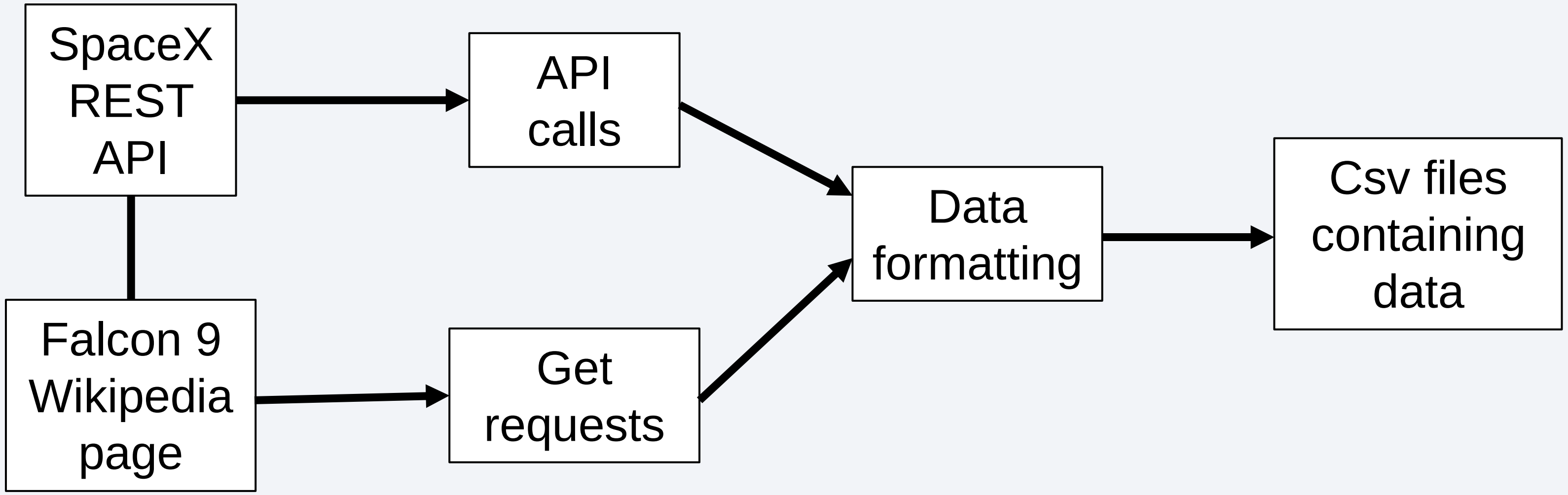
Executive Summary

- Data collection methodology:
  - Data was collected using python from SpaceX API and web scraping from wikipedia
- Perform data wrangling
  - Data was processed to remove null values and other preprocessing calculations, and the target variable was determined
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Models were built using sklearn, hyperparameters were tuned using GridSearchCV and evaluated using cross-validation (CV) score.

Data Collection

Two mutually-exclusive sources were used to collect the data:

- SpaceX REST API (using API calls)
- Falcon 9 Wikipedia page (using Web scraping tools)



Data Collection – SpaceX API

Key Details

- **Overall objective:** Use SpaceX REST API to extract the Falcon9 data and store it in cvs format.
- **Tools used:** Python notebook and associated API tools.
- **URL of the SpaceX API:** https://api.spacexdata.com/v4/launches/past
- **GitHub URL of the SpaceX data collection notebook:** https://github.com/PSCPU/Falcon-9-launch/blob/main/Data%20Collection%20API.ipynb

Get request Falcon9 Launch data from SpaceX API

↓

Decode the response as json and convert to dataframe

↓

Use API again to get information about the launches using the IDs for each launch from rocket, payload, launch and cores columns

↓

Create empty lists and store the data from requests to lists

↓

Combine the columns names (keys) and the associated data (values) into a dictionary

↓

Create the pandas dataframe from the dictionary

↓

Save the dataframe to a csv file

Data Collection - Scraping

Key Details

- **Overall objective:** Scrape the data from target table on Falcon 9 Wikipedia page and store it in cvs format.
- **Tools used:** Python notebook and associated web scraping tools.
- **Scraped Falcon 9 Wikipedia page** https://en.wikipedia.org/wiki/List_of_Falcon/_9/_and_Falcon_Heavy_launches
- **GitHub URL of the web scraping notebook:** https://github.com/PSCPU/Falcon-9-launch/blob/main/Data%20Collection%20with%20Web%20Scraping.ipynb

HTTP GET method: request the Falcon9 Launch HTML page

↓

Beautifulsoup object from the Wikipedia page

↓

find all tables on the wiki page

↓

Check the contents of the target table

↓

Extract the column names of the target table

↓

Create a dictionary with keys from the extracted column names

↓

Fill dictionary with launch records extracted from table rows

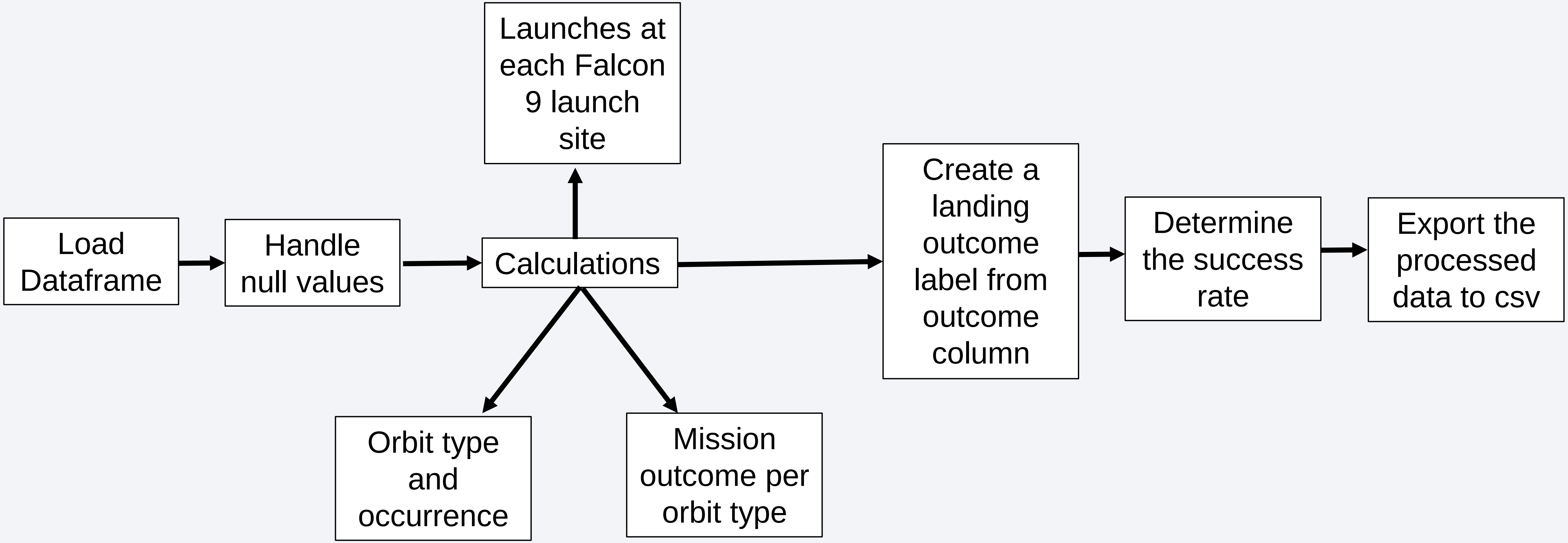
↓

Create the pandas dataframe from the dictionary

↓

Save the dataframe to a csv file

Data Wrangling



GitHub URL: https://github.com/PSCPU/Falcon-9-launch/blob/main/EDA.ipynb

EDA with Data Visualization

Following charts were used for visualizing the relationships between different feature variables:

1. Scatterplots for discrete variables:

   a. Flight number versus - payload mass

   - launch site

   - orbit type

   c. Launch site versus payload mass

   d. Payload versus orbit type

2. Bar plot for success rate versus orbit type (one continuous and one discrete variable)

3. Line plot for success rate versus year (periodically incremental variable)

GitHub URL: https://github.com/PSCPU/Falcon-9-launch/blob/main/EDA-Visualization.ipynb

EDA with SQL

SQL queries were performed to accomplish the following tasks:

- Display the names of the unique launch sites in the space mission
- Display 5 records where launch sites begin with the string 'CCA'
- Display the total payload mass carried by boosters launched by NASA (CRS)
- Display average payload mass carried by booster version F9 v1.1
- Date when the first succesful landing outcome in ground pad was achieved.
- Boosters with success in drone ship and payload > 4000 but < 6000
- Total number of successful and failure mission outcomes
- Names of the booster versions which have carried the maximum payload mass
- Month names, failure landing in drone ship, booster versions, launch_site for 2015.
- Rank the successful landing outcomes between 04-06-2010 and 20-03-2017 in descending order.

GitHub URL: https://github.com/PSCPU/Falcon-9-launch/blob/main/EDA-SQL.ipynb

Build an Interactive Map with Folium

Following map objects were created and added to the folium map:

– Circle (for a highlighted circle area on a specific coordinate)

– Marker (to mark each launch site on the site map)

– Marker clusters (to indicate many launch records with the exact same location)

– MousePosition (to get coordinate for a mouse over a point on the map)

– PolyLine (to plot distance lines of launch sites from the proximitines)

GitHub URL: https://github.com/PSCPU/Falcon-9-launch/blob/main/Folium-maps.ipynb

Build a Dashboard with Plotly Dash

Following plots/graphs and interactions have been added to a dashboard:

– Dropdown list (to enable Launch Site selection)

– Pie chart (to show the total successful launches count for all sites)

– Scatter chart (to show the correlation between payload and launch success)

GitHub URL: https://github.com/PSCPU/Falcon-9-launch/blob/main/spacex_dash_app.py

Predictive Analysis (Classification)

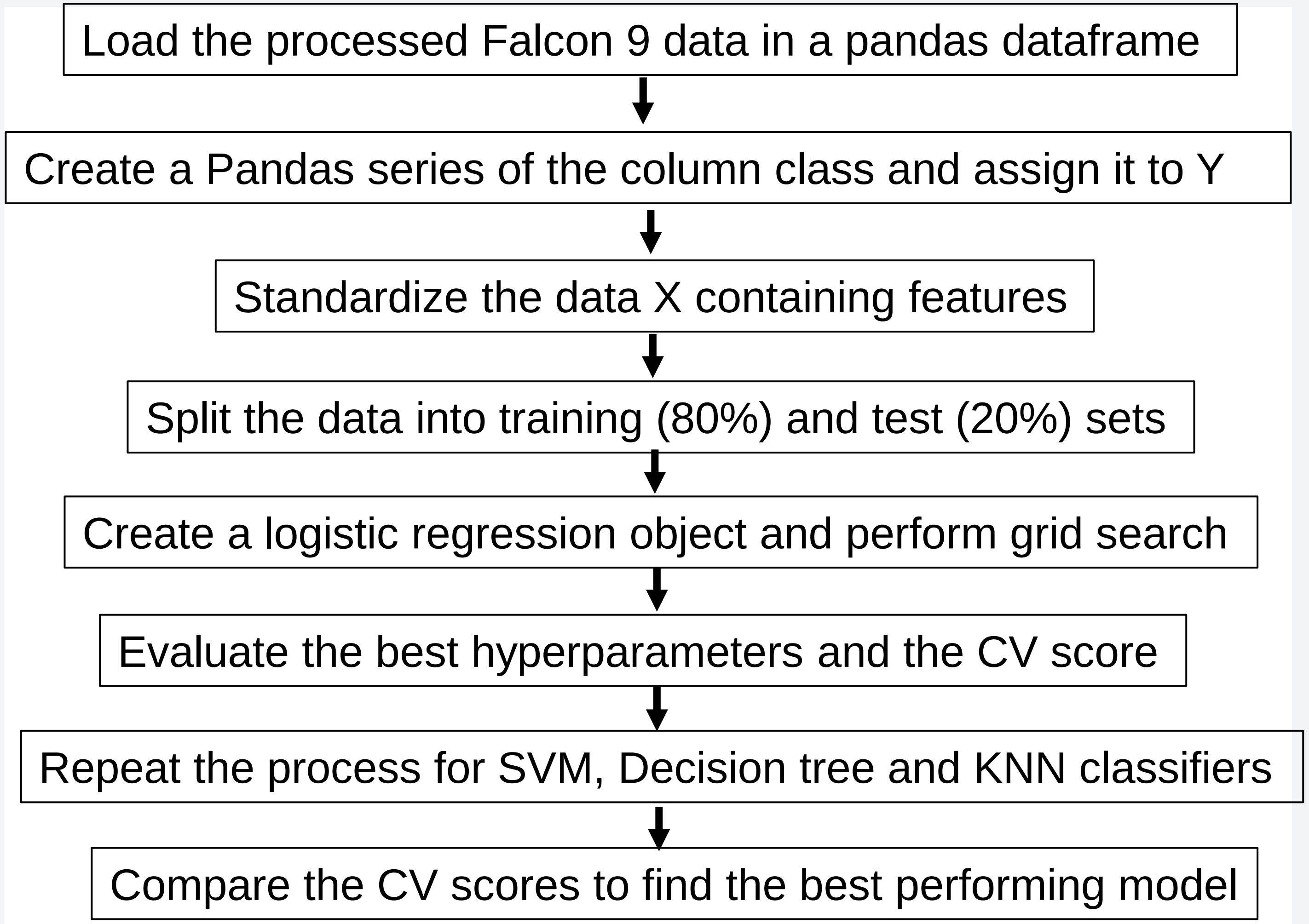
- Performance of the following four classification methods were compared after performing hyperparameter optimization:

– Logistic Regression

– Support Vector Machines (SVM)

– Decision Tree Classifiers

– K-Nearest Neighbor (KNN)



GitHub URL: https://github.com/PSCPU/Falcon-9-launch/blob/main/spacex_dash_app.py

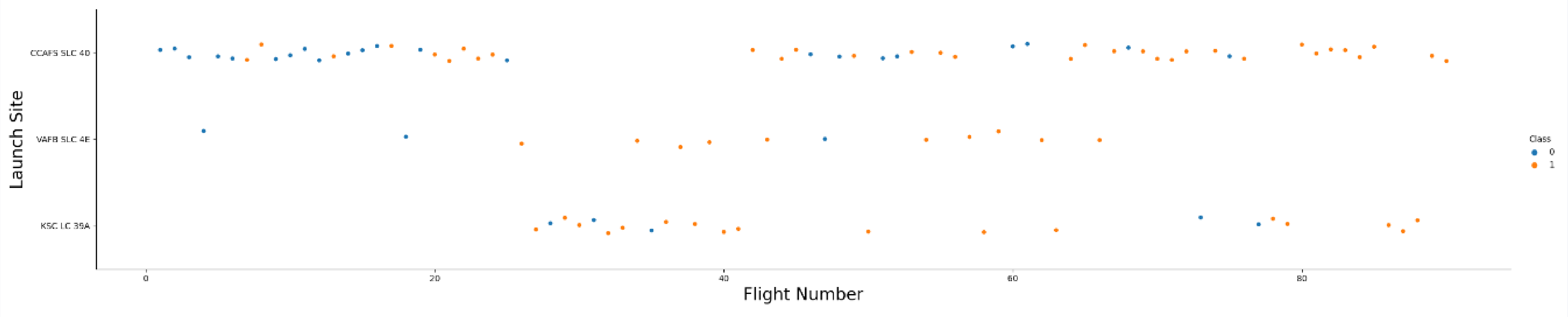
Results

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

Section 2

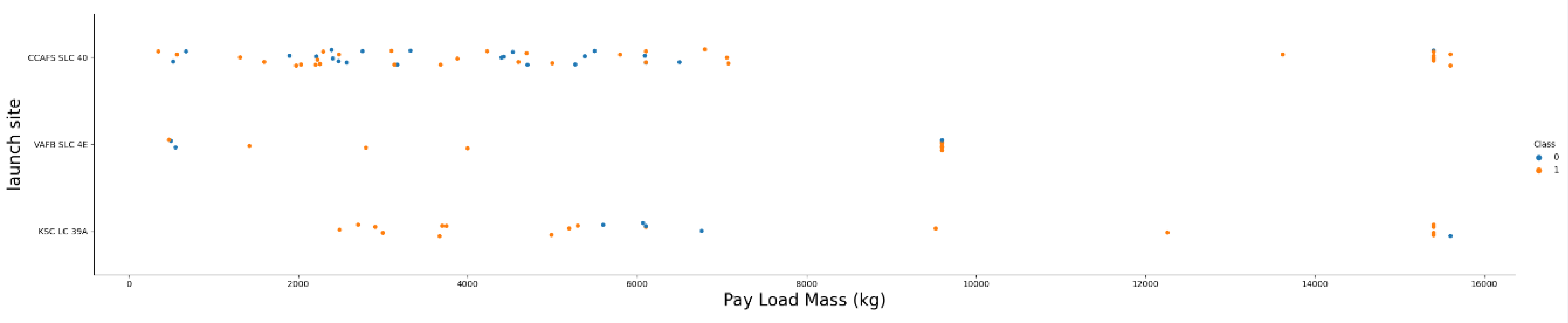
Insights drawn from EDA

Flight Number vs. Launch Site



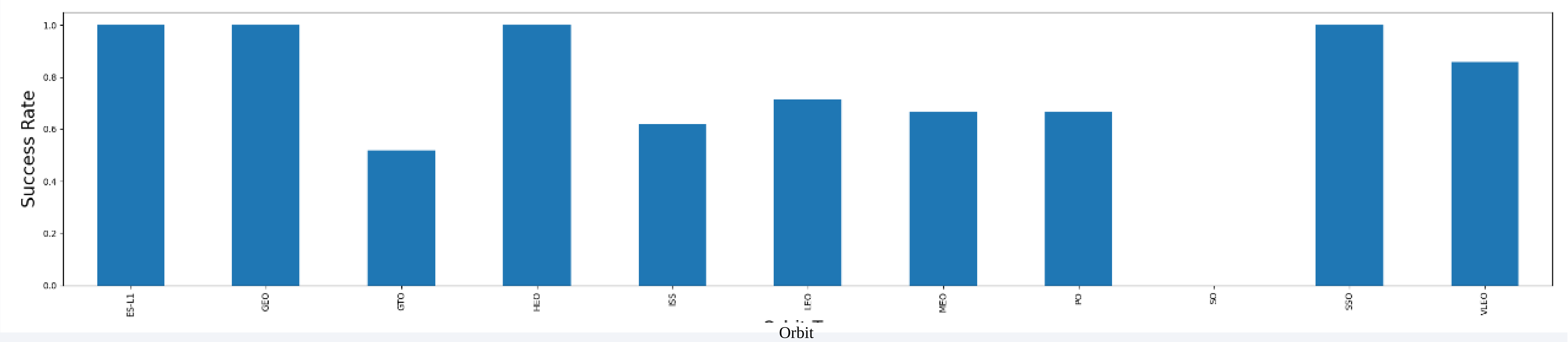
- Launches from CCAFS SLC-40 station are significantly higher than other stations.

Payload vs. Launch Site



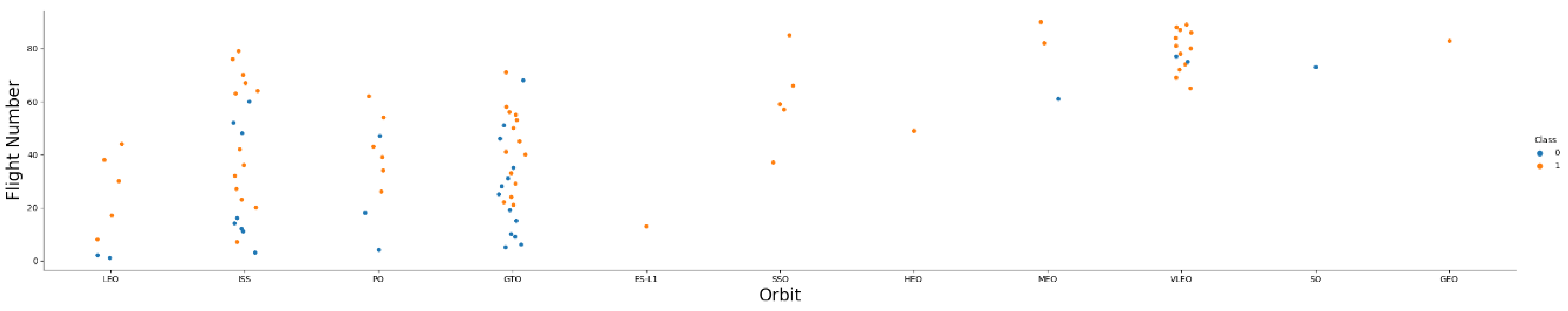
- Very high payloads are not launched at VAFS-SLC-4E station.
- The proportion of high payload mass launches is significantly smaller than low or moderate payload mass.

Success Rate vs. Orbit Type



- Four orbits have almost 100% success rate.
- One orbit has 0% success rate.

Flight Number vs. Orbit Type

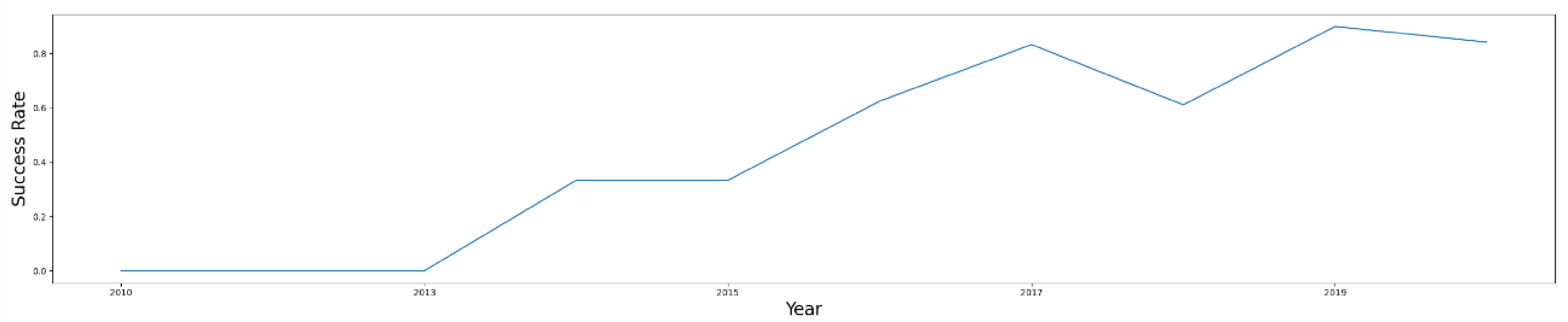


- For LEO orbit the Success appears related to the number of flights.
- On the other hand, there seems to be no relationship between flight number when in GTO orbit.

Payload vs. Orbit Type

- VLEO orbit carries the high pay load mass.
- Low payload mass is generally carried by LSS
- Moderate payload mass is generally carried by GTO
- LSS and GTO together carry majority of the medium and low payload masses

Launch Success Yearly Trend



- Barring a small dip in 2018 and 2020, the yearly average success rate has risen over the years.
- The highest success rate of close to 90% was observed in 2019.

All Launch Site Names

Query:

```
%sql SELECT DISTINCT Launch_Site FROM SPACEXTBL;
```

Result:

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

- Four launch sites were used in total

Launch Site Names Begin with 'CCA'

Query:

```
%sql SELECT * from SPACEXTBL WHERE Launch_Site like 'CCA%' LIMIT 5
```

Result:

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

- All these older launches from CCAFS-LC-40 used the LEO orbit type.
- Some of these earlier launches did not attempt landing.

Total Payload Mass

Query:

```
%sql SELECT SUM(PAYLOAD_MASS__KG_) FROM SPACEXTBL WHERE Customer='NASA (CRS)'
```

Result:

SUM(PAYLOAD_MASS__KG_)
45596

- Total of 45596 kg payload mass was carried by boosters launched by NASA (CRS)

Average Payload Mass by F9 v1.1

Query:

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

Result:

AVG(PAYLOAD_MASS__KG_)
2928.4

- An average of 2928.4 kg was carried by F9 v1.1 boosters

First Successful Ground Landing Date

Query:

```
%sql SELECT max(Date) as Date from SPACEXTBL WHERE `Landing _Outcome` like 'SUCCESS (ground pad)'
```

Result:

Date
22-12-2015

- The first successful ground landing occurred on 22 December 2015

Successful Drone Ship Landing with Payload between 4000 and 6000

Query:

```
%sql SELECT DISTINCT BOOSTER_VERSION FROM SPACEXTBL WHERE PAYLOAD_MASS__KG_ > 4000 AND PAYLOAD_MASS__KG_ < 6000 AND `Landing _Outcome` like 'SUCCESS
```

Result:

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

- Four F9 FT booster versions – B1022, B1026, B1021.2 and B.1031.2 fulfilled the query criteria

Total Number of Successful and Failure Mission Outcomes

Query:

```
%sql SELECT COUNT(MISSION_OUTCOME) as 'success/failure' FROM SPACEXTBL group by MISSION_OUTCOME like 'failure%' or 'Success%'
```

Result:

success/failure
100
1

- 100 successful and 1 failed missions occured

Boosters Carried Maximum Payload

Query:

```
%sql select DISTINCT Booster_Version FROM SPACEXTBL WHERE PAYLOAD_MASS__KG_=(select max(PAYLOAD_MASS__KG_) from SPACEXTBL)
```

Result:

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

- 12 distinct booster versions carried the maximum payload

2015 Launch Records

```
%sql SELECT substr(Date,4, 2) as Month, `Landing _Outcome`, Booster_Version, Launch_Site FROM SPACEXTBL WHERE `Landing _Outcome`='Failure (drone ship)
```

Month	Landing _Outcome	Booster_Version	Launch_Site
01	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
04	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

These are the records of failed landings on drone ships, along with the booster version and launch site

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

```
%sql SELECT COUNT(`Landing _Outcome`) from SPACEXTBL WHERE `Landing _Outcome` like 'Success%' AND DATE BETWEEN '04-06-2010' AND '20-03-2017'
```

COUNT(`Landing _Outcome`)
34

- 34 landing outcomes between the date 2010-06-04 and 2017-03-20.

Section 3

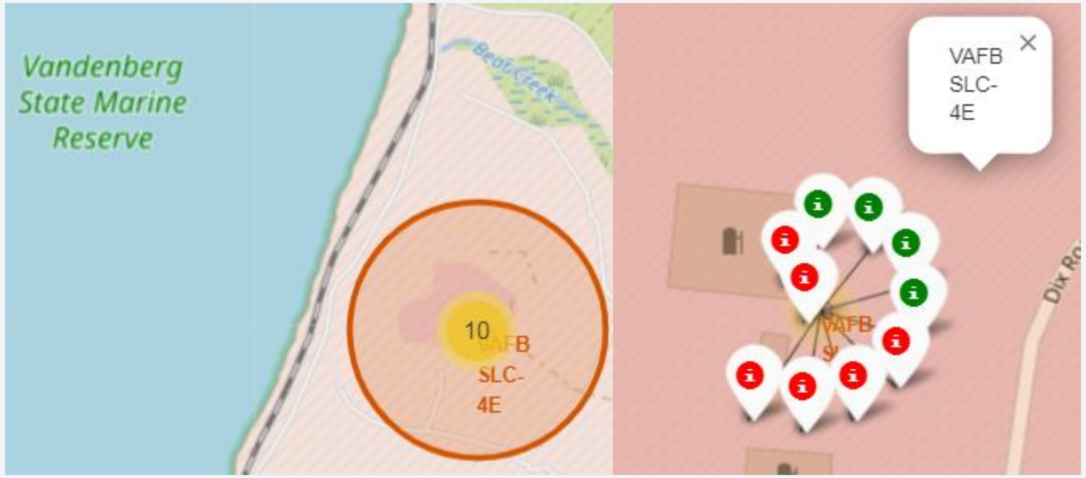
Launch Sites Proximities Analysis

Folium Map with marked Launch Sites

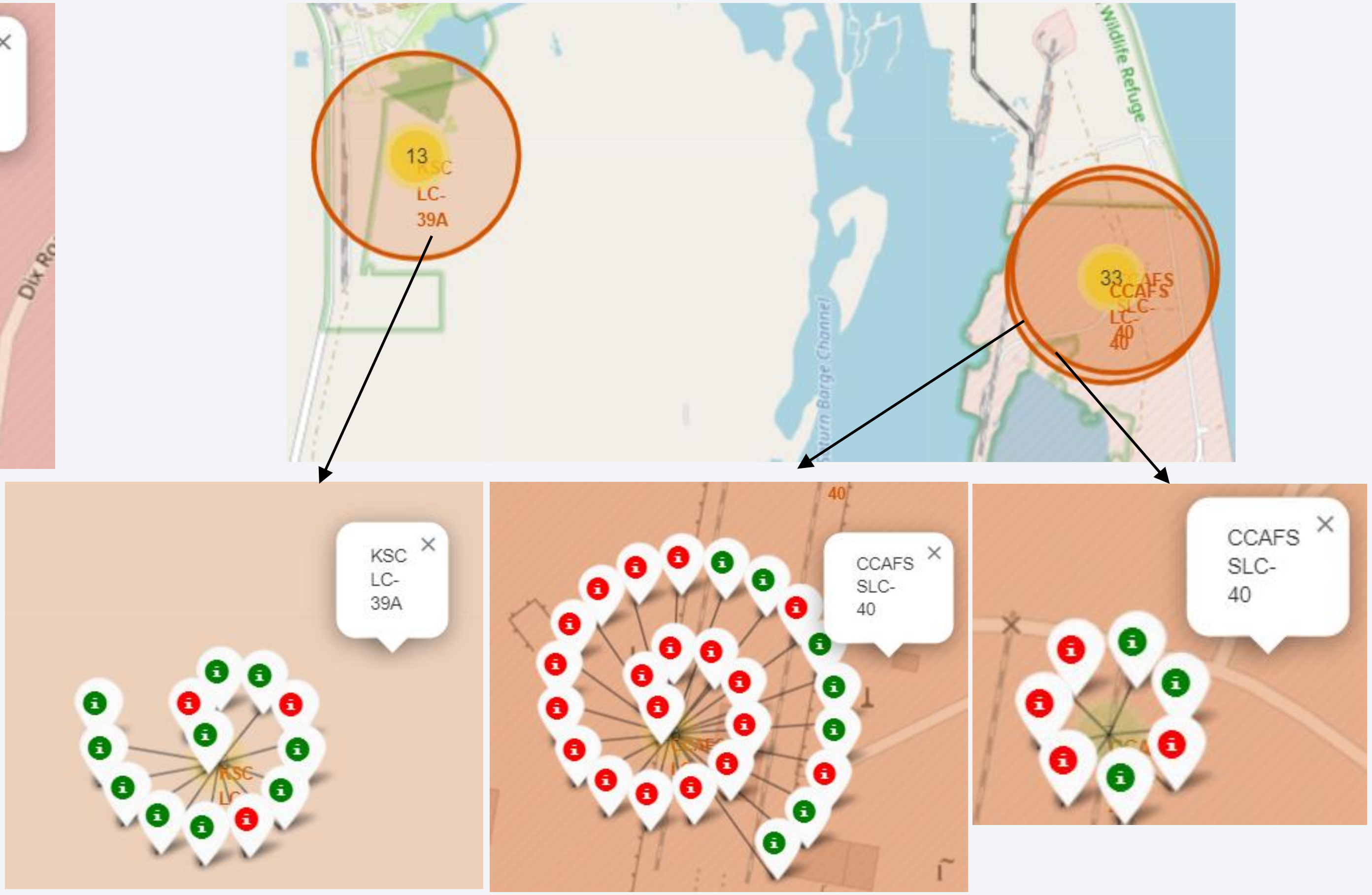
- Launch sites marked in orange on the Folium map

Success/failed launches for each site on the map

California site

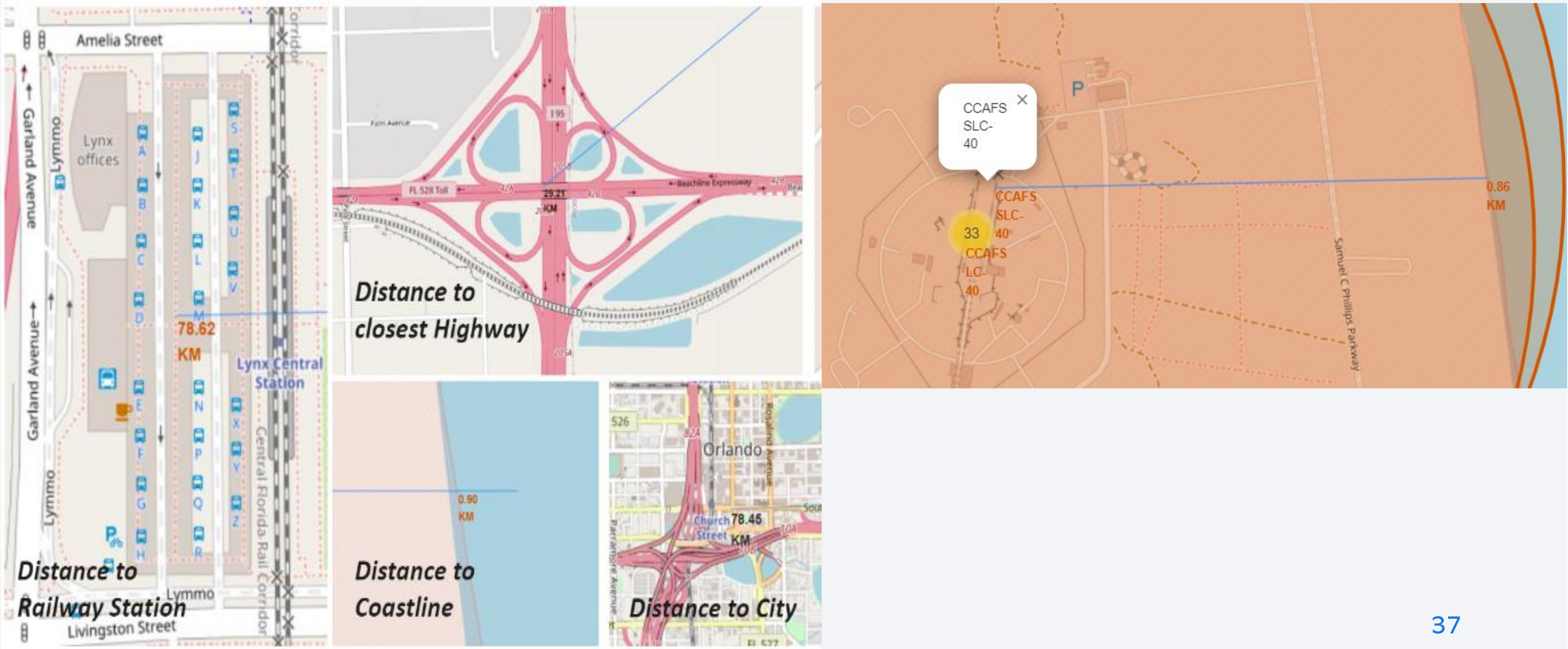


Florida sites



Red indicates successful launch, whereas green indicates failed launch

Proximity analysis using Folium

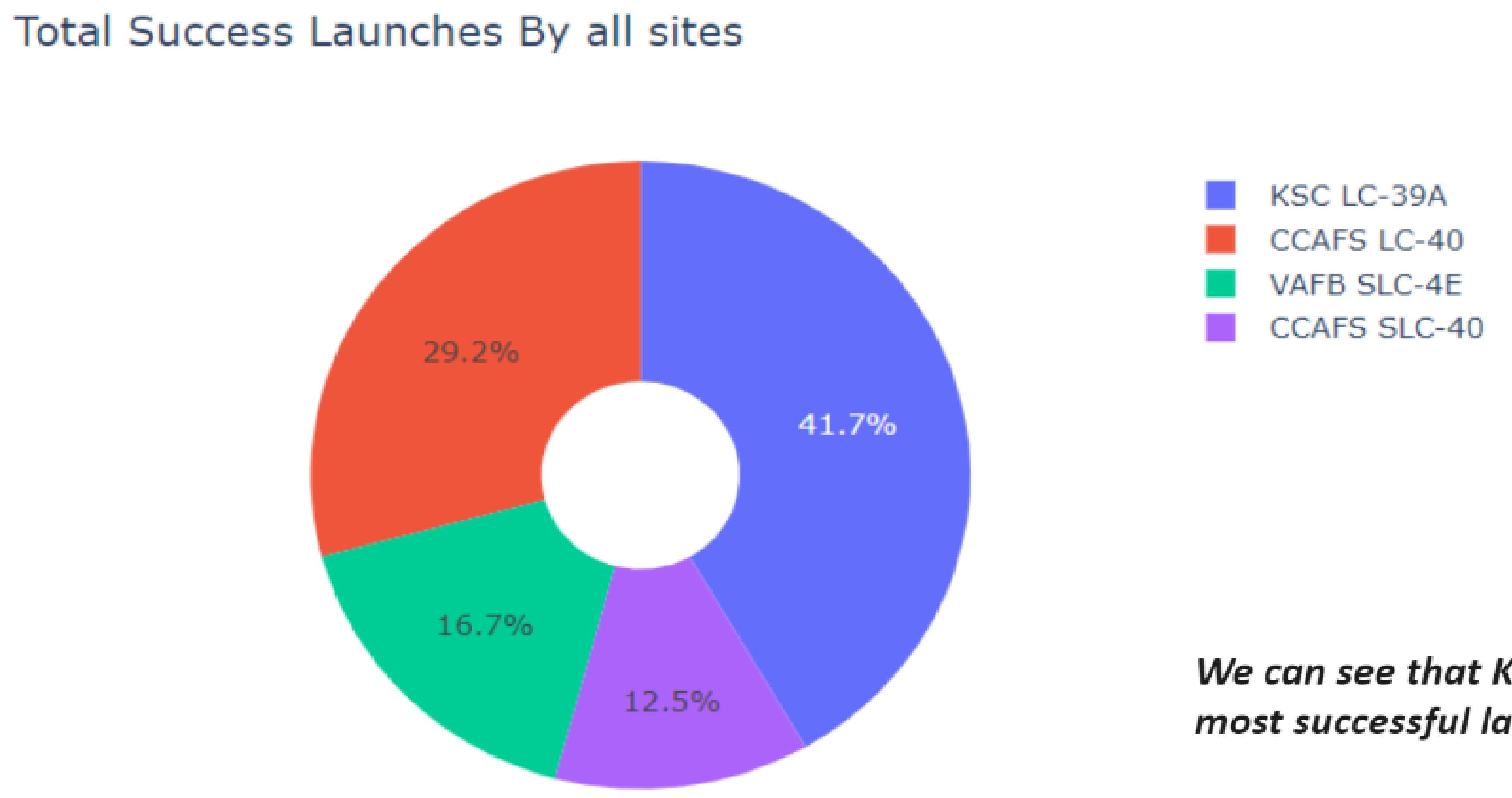


Launch sites are close to coastlines

Section 4

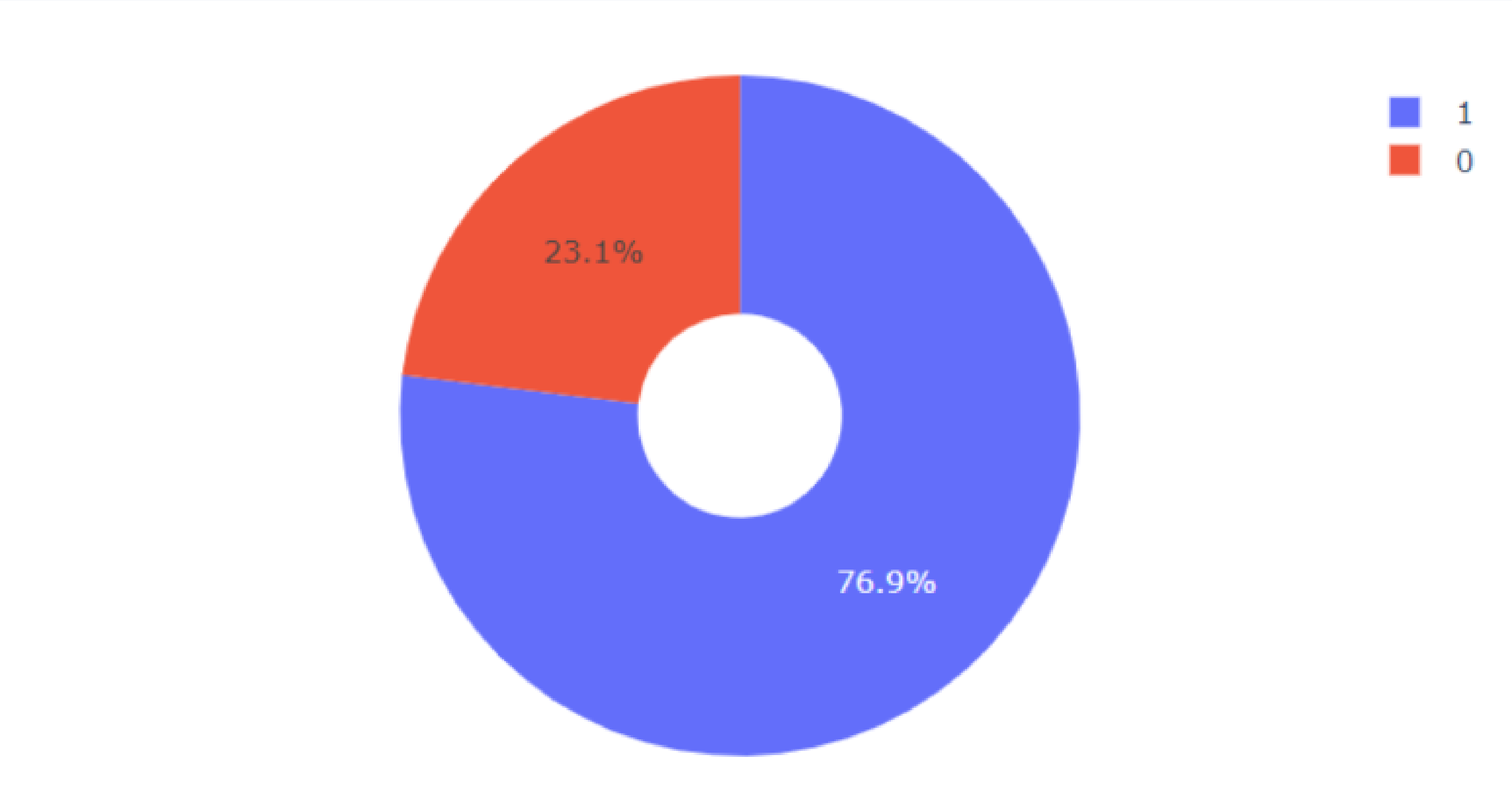
Build a Dashboard with Plotly Dash

Success percentage achieved by each launch site



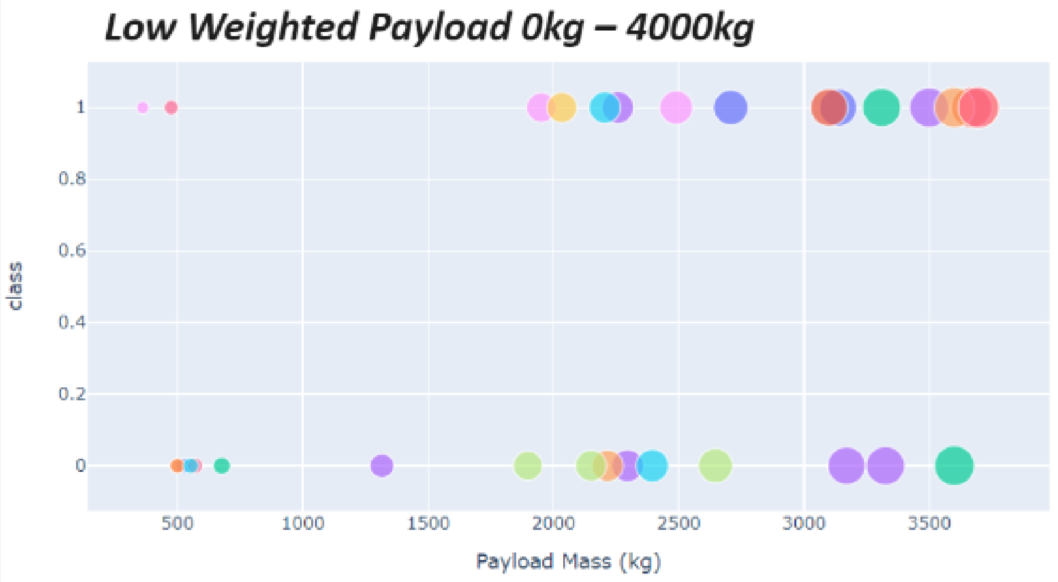
We can see that KSC LC-39A had the most successful launches from all the sites

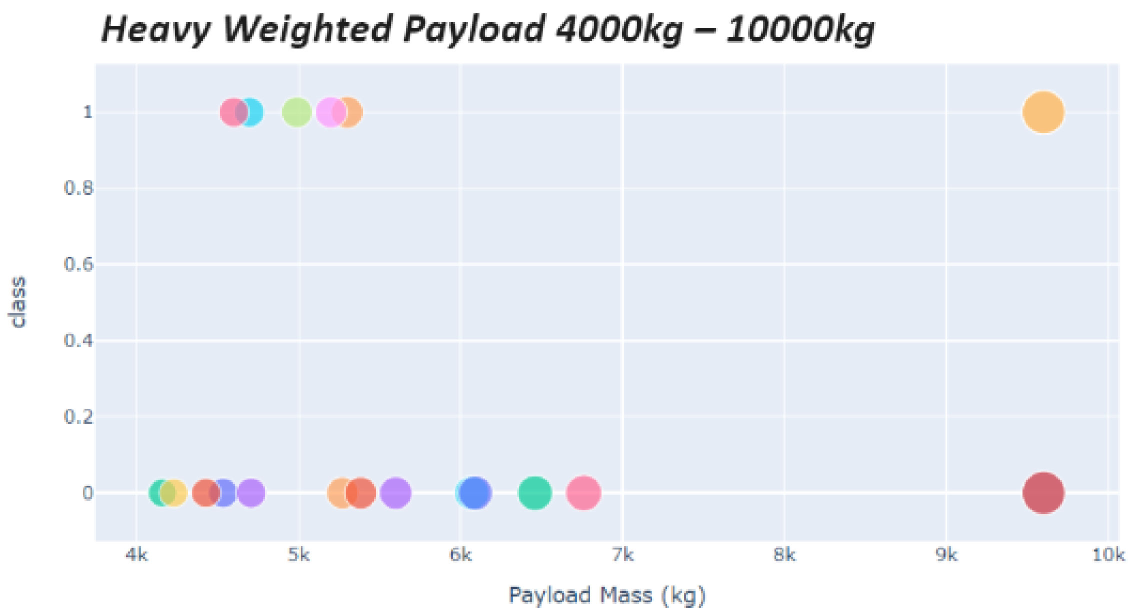
Launch site with the highest launch success ratio



- KSC LC-39A achieved the highest success rate

<Dashboard Screenshot 3>





Low rated payload has higher success rate

Section 5

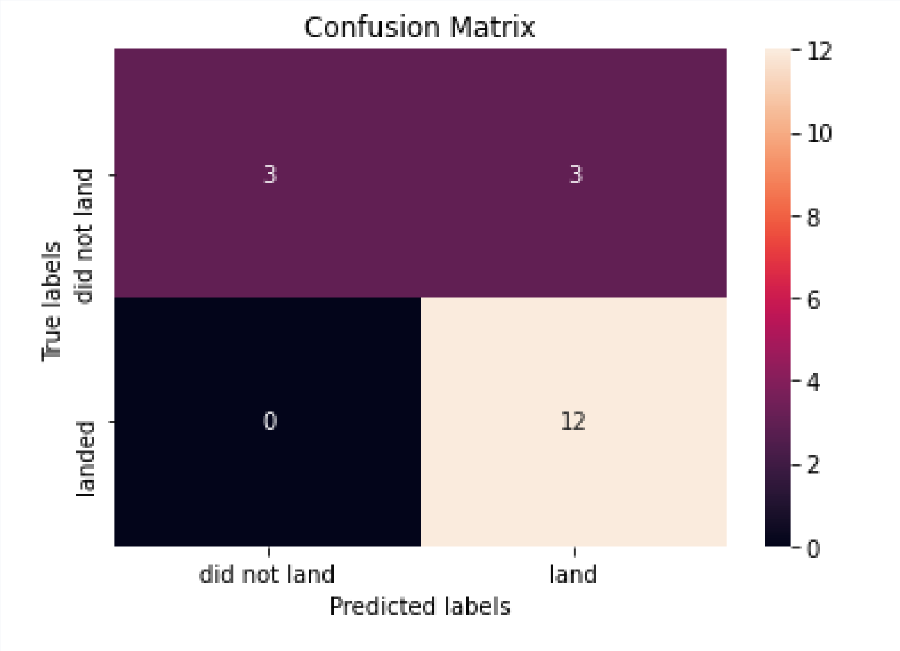
Predictive Analysis (Classification)

Classification Accuracy

```
Accuracy for Logistics Regression method: 0.8333333333333334
Accuracy for Support Vector Machine method: 0.8333333333333334
Accuracy for Decision tree method: 0.5555555555555556
Accuracy for K nearest neighbors method: 0.8333333333333334
Logistic Regression, Support Vector Machine and K-nearest neighbors perform equally well
```

```
Logistic Regression, Support Vector Machine and K-nearest neighbors perform equally well
```

Confusion Matrix



Decision tree classifier r can distinguish between the different classes. The major problem is the false positives.

Conclusions

- Logistic regression, support vector machines and decision tree classifiers provide equal accuracy, as evidenced from their cross-validation scores.
- KSC LC-39A achieved the highest success rate.
- Older launches from CCAFS-LC-40 used the LEO orbit type.
- Some of these earlier launches did not attempt landing.
- Very high payloads are not launched at VAFS-SLC-4E station.
- The proportion of high payload mass launches is significantly smaller than low or moderate payload mass.

Thank you!