IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Piotr Palacz
08.03.2023

Outline

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

Executive Summary

Summary of methodologies:

- Data Collection – SpaceX API
- Data Collection – Scraping
- Data Wrangling
- EDA with Data Visualization
- EDA with SQL
- Build an Interactive Map with Folium
- Build a Dashboard with Plotly Dash
- Predictive Analysis (Classification)

Summary of all results:

- Exploratory data analysis results
- Interactive analytics in screenshots
- Predictive Analytics resault

Introduction

In this project, we 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 because SpaceX can reuse the first stage. Therefore 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 SpaceX for a rocket launch.

Section 1

Methodology

Methodology

Executive Summary

- Data was extracted from the API issued by SpaceX and Wikipedia.
- The extracted data were filtered to show only Falcon 9 launches, then missing payload mass data were replaced with an average value of payload mass
- Exploratory data analysis (EDA) using visualization of correlation between parameters and SQL
- Interactive visual analytics using Folium and Plotly Dash
- Predictive analysis using classification models and attempting to select the best model

Data Collection

Data was collected from Space X Rest API and webscraping from Wikipedia

Space X API

Request and parse the SpaceX launch data using the GET reques → Decode the response content as a Json → Filter the dataframe to only include Falcon 9 launches → Missing payload mass data were replaced with an average value of payload mass

Wikipedia

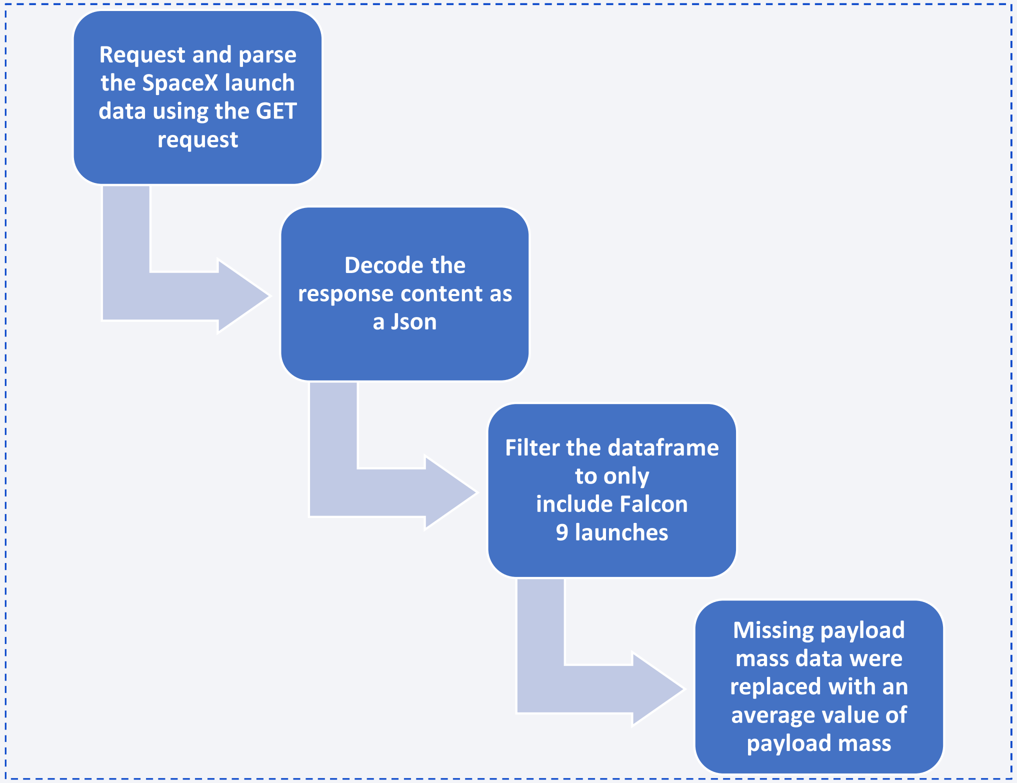
Request the Falcon9 Launch Wiki page from its URL → Decode the response content with a BeautyfulSoup → Extract all column/variable names from the HTML table header → Create a data frame by parsing the launch HTML tables

Data Collection – SpaceX API

First we Request and parse the SpaceX launch data using the GET request function, then we decode the response content as a Json. At the end we filtered the dataframe to only include Falcon 9 launches and replaced missing payload mass data with an average value of payload mass

To check the notebook:

[GitHub URL]

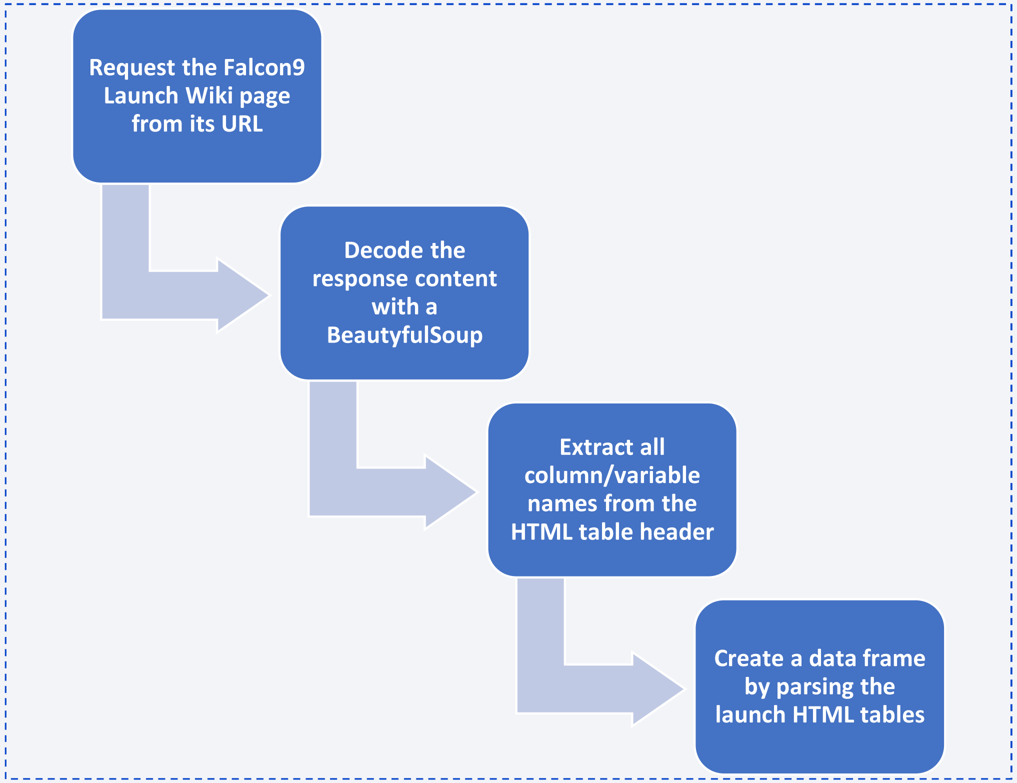


Data Collection - Scraping

First we request the Falocon 9 data from Wikipedia page from its URL, then we decode the response content using BeautyfulSoup library. At the end we extract all column/variable names from the HTML table header and create a data frame.

To check the notebook:

[GitHub URL]



Data Wrangling

We filtered the dataframe to only include Falcon 9 launches and replaced missing payload mass data with an average value of payload mass using the NumPY library.

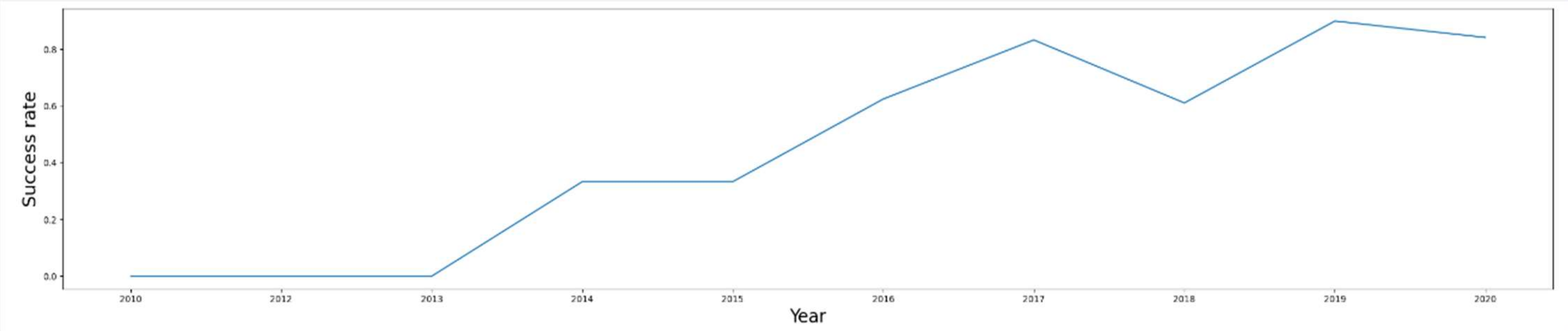
```
# Calculate the mean value of PayloadMass column
Mean_PLM = data_falcon9.PayloadMass.mean()

# Replace the np.nan values with its mean value
data_falcon9['PayloadMass'] = data_falcon9['PayloadMass'].replace(np.nan, Mean_PLM)
data_falcon9.isnull().sum()
```

To check the notebook:

[GitHub URL]

EDA with Data Visualization

Using dot plots, it is visualised how the relationships between specific data influence launch success. In addition, thanks to the visualisation, we can easily see that the success of the launch is increasing year on year.



To check the notebook:

[GitHub URL]

EDA with SQL

- **The total payload mass carried by boosters launched by NASA (CRS) was 45596 kg**
- **Average payload mass carried by booster version F9 v1.1 was 2928.4 kg**
- **First succesful landing outcome in ground pad was achieved in 01-05-2017**
- **F9 FT B1022, F9 FT B1026, F9 FT B1021.2, F9 FT B1031.2 was boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000**
- **It was 100 successful mission outcomes and only 1 failure**
- **in year 2015 were two lunches:**

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

To check the notebook:

[GitHub URL]

Build an Interactive Map with Folium

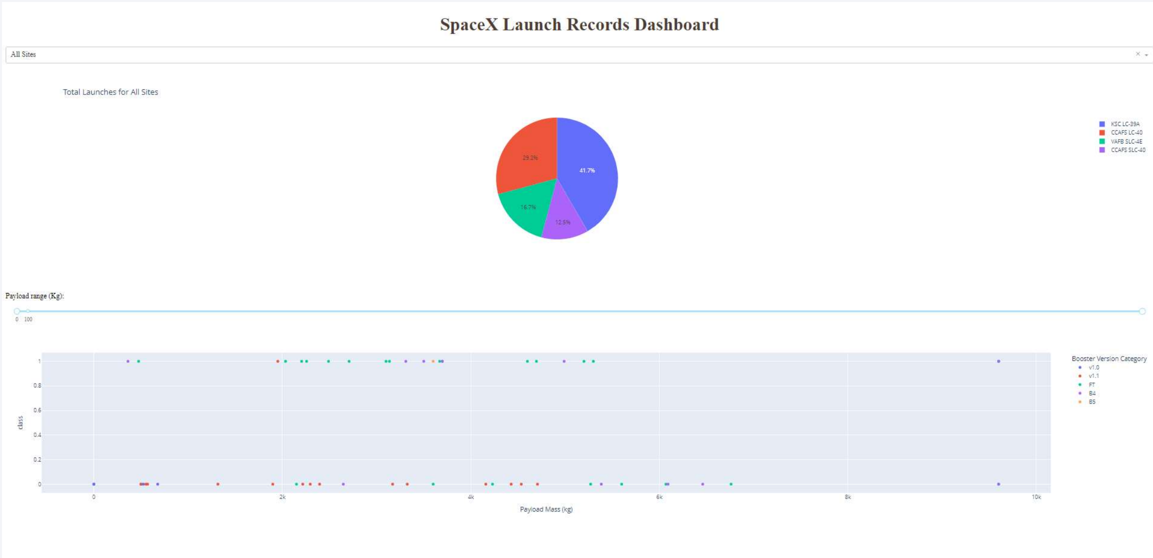
On map was marked all launch sites, the success/failed launches for each site and was calculated the distances between a launch site to its proximities. We use folium Circle to add a highlighted circle area with a text label on a specific coordinate. Next, we explore and analyze the proximities of launch sites. We mark down a point on the closest coastline using MousePosition and calculate the distance between the coastline point and the launch site, then draw a PolyLine between a launch site to the selected coastline point.

To check the notebook:

[GitHub URL]

Build a Dashboard with Plotly Dash

We bulit a Dashbord including menu for selecting lunch site, pie charts displaying success rate, scatter chart displaying lunch site, payload mass and success/failure. This dashbord was bulit for find lunch site with highest success rate and payload range with highest lunch success rate

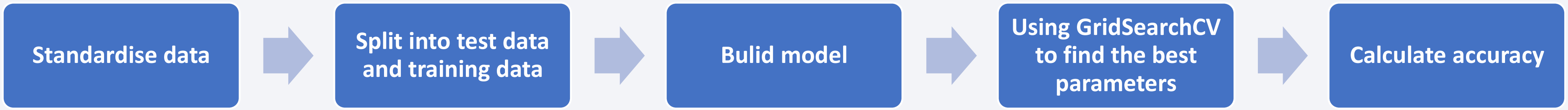


To check the dashboard:

[GitHub URL]

Predictive Analysis (Classification)

Using the standardscaler library, the variables used in the analysis were standardised. the dataset was split into test data and training data. 4 machine learning models were then created: logistic regression, support vector machine, decision tree classifier and k nearest neighbours. For each model, tuning was carried out using GridSearchCV to find the best parameters. For each model, the accuracy and For each model, the accuracy was calculated and the best model was selected on the basis of this accuracy.



To check the notebook:

[GitHub URL]

Results

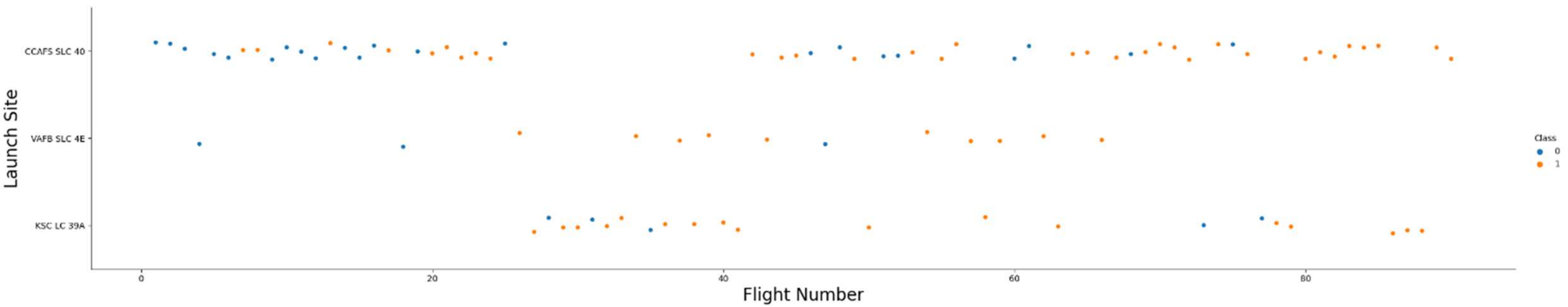
- Exploratory data analysis results
- Interactive analytics in screenshots
- Predictive Analytics resault

Section 2

Insights drawn from EDA

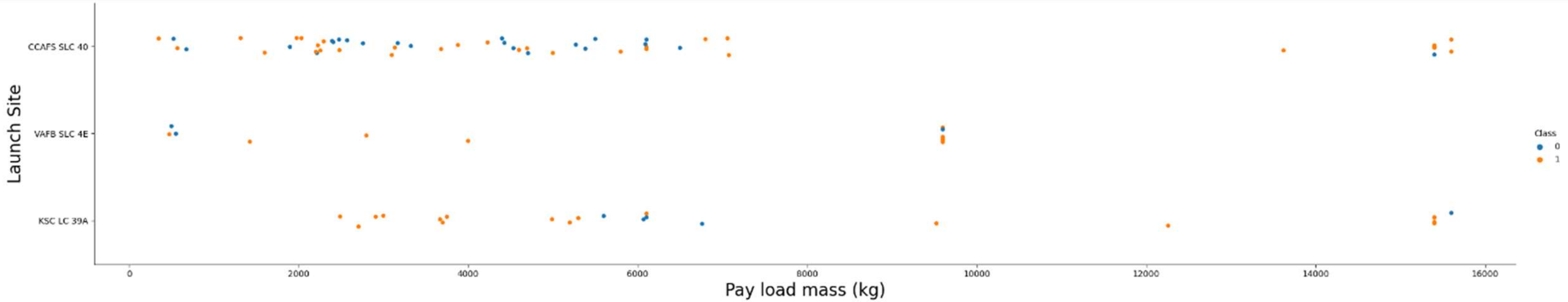
Flight Number vs. Launch Site

In the graph, we can see that the higher the flight number, the greater the probability of success. This means that the rockets got better and better over time.



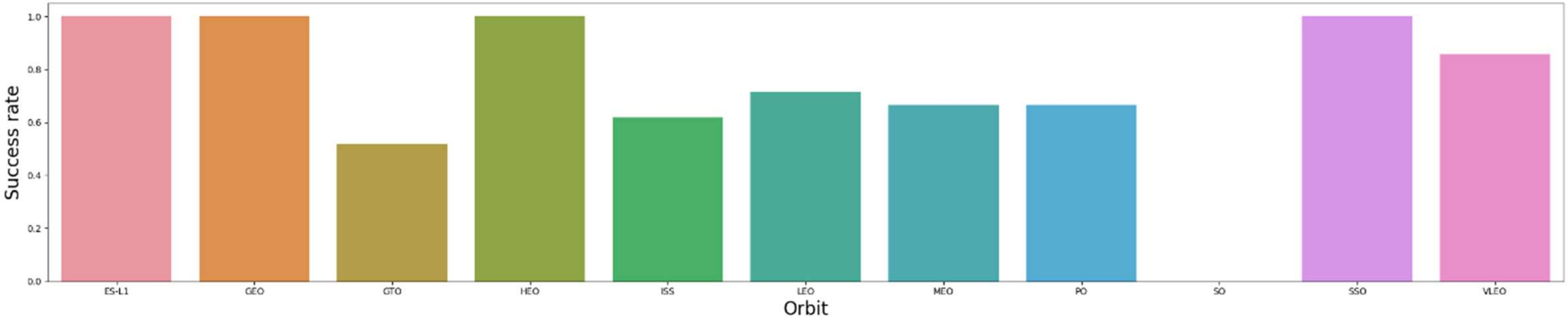
Payload vs. Launch Site

In the graph, we can see that the greather pay load mass, the greater the probability of success.



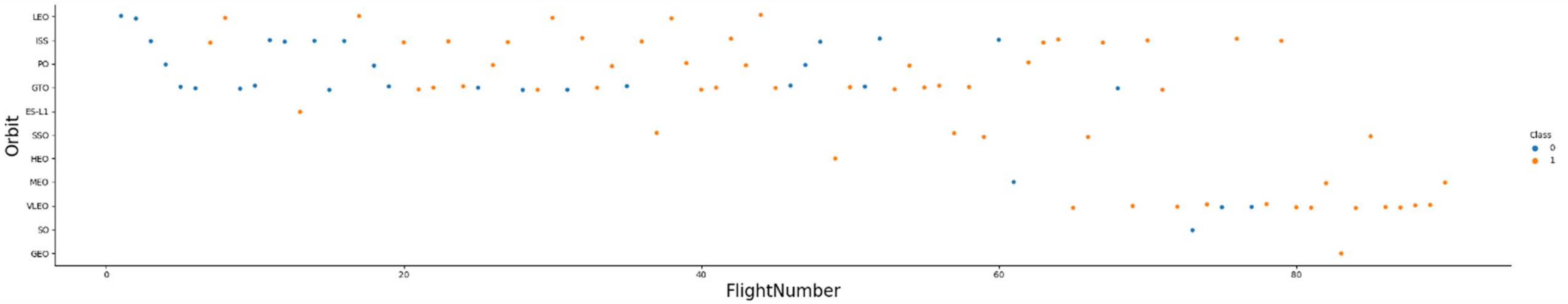
Success Rate vs. Orbit Type

We can see that orbits ES-L1, GEO, HEO, SSO have 100% success rate. Also orbit type SO have 100% failure.



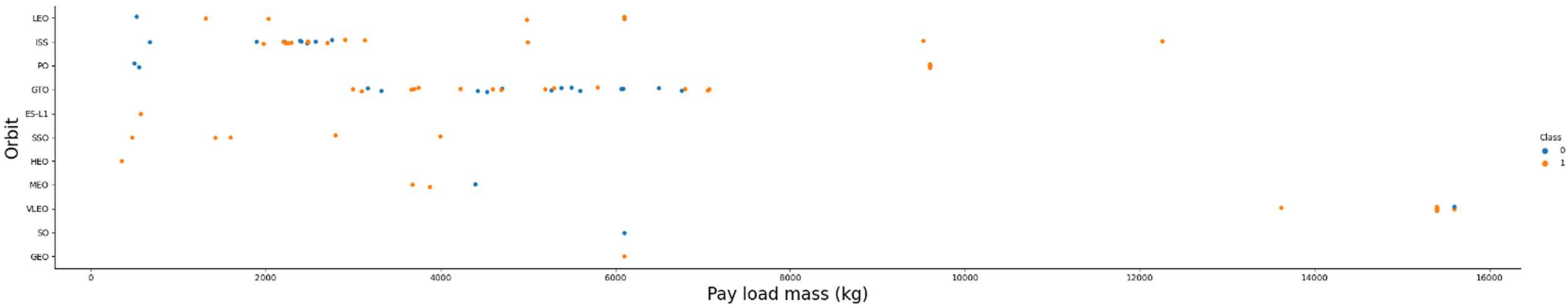
Flight Number vs. Orbit Type

In the graph, we can see that GTO orbits are relatively rarely successful, in contrast to VLEO orbits where success is frequent.



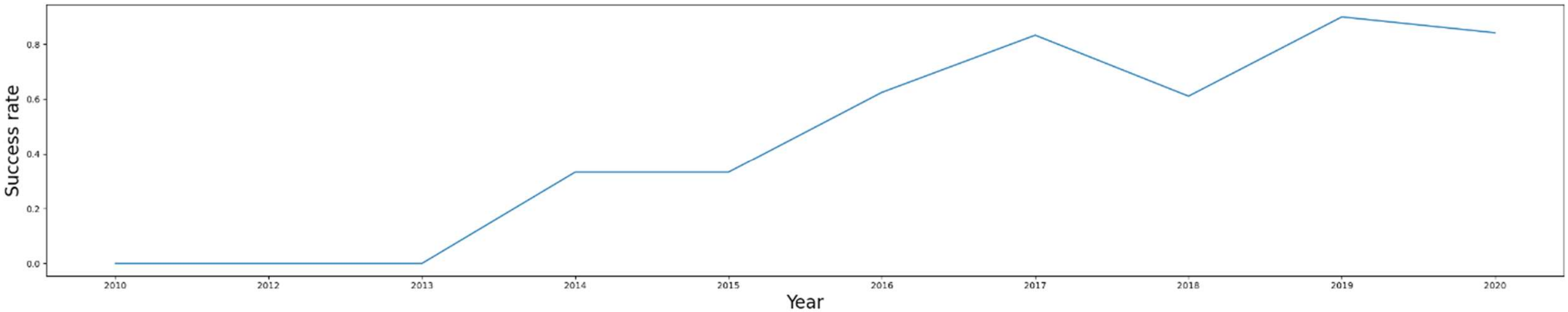
Payload vs. Orbit Type

On the plot we can see that heavy paylooads for PO and LEO orbits increase the success rate.



Launch Success Yearly Trend

Thanks to the visualisation, we can easily see that the success of the launch is increasing year on year.



All Launch Site Names

There are 4 distinct lunch site names

```
%sql SELECT distinct Launch_Site FROM SPACEXTBL
```

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

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

Launch Site Names Begin with 'CCA'

5 records where launch sites name starts with `CCA`

```
%%sql SELECT Launch_Site FROM SPACEXTBL
        WHERE Launch_Site LIKE "CCA%"
        LIMIT 5
```

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

Launch_Site
CCAFS LC-40
CCAFS LC-40
CCAFS LC-40
CCAFS LC-40
CCAFS LC-40

Total Payload Mass

The total payload carried by boosters from NASA

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

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

SUM(PAYLOAD_MASS__KG_)
45596

Average Payload Mass by F9 v1.1

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

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

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

AVG(PAYLOAD_MASS__KG_)
2928.4

First Successful Ground Landing Date

The date of the first successful landing outcome on ground pad

```
%%sql SELECT MIN(Date) FROM SPACEXTBL
    WHERE [Landing _Outcome] = "Success (ground pad)"
```

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

MIN(Date)
01-05-2017

Successful Drone Ship Landing with Payload between 4000 and 6000

The names of boosters which have successfully landed on drone ship and had payload mass greater than 4000 but less than 6000

```
%%sql SELECT Booster_Version FROM SPACEXTBL
    WHERE [Landing _Outcome] = "Success (drone ship)"
    AND PAYLOAD_MASS__KG_ BETWEEN 4000 AND 6000
```

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

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

Total Number of Successful and Failure Mission Outcomes

The total number of successful and failure mission outcomes

```
%%sql SELECT Mission_Outcome, COUNT(Mission_Outcome) FROM SPACEXTBL
        GROUP BY Mission_Outcome
```

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

Mission_Outcome	COUNT(Mission_Outcome)
Failure (in flight)	1
Success	98
Success	1
Success (payload status unclear)	1

Boosters Carried Maximum Payload

The names of the booster which have carried the maximum payload mass

```
%%sql SELECT Booster_Version FROM SPACEXTBL
        WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTBL)
```

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

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

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

```
%%sql SELECT SUBSTRING(Date, 4, 2) AS Month, [Landing _Outcome], Booster_Version, Launch_Site
  FROM SPACEXTBL
  WHERE [Landing _Outcome] = 'Failure (drone ship)'
  AND SUBSTRING(Date,7,4) = '2015'
```

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

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

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

```
%%sql SELECT [Landing _Outcome], COUNT([Landing _Outcome]) AS
  FROM SPACEXTBL
  WHERE [Landing _Outcome] LIKE '%Success%'
  AND Date BETWEEN '04-06-2010' AND '20-03-2017'
  GROUP BY [Landing _Outcome]
  ORDER BY
```

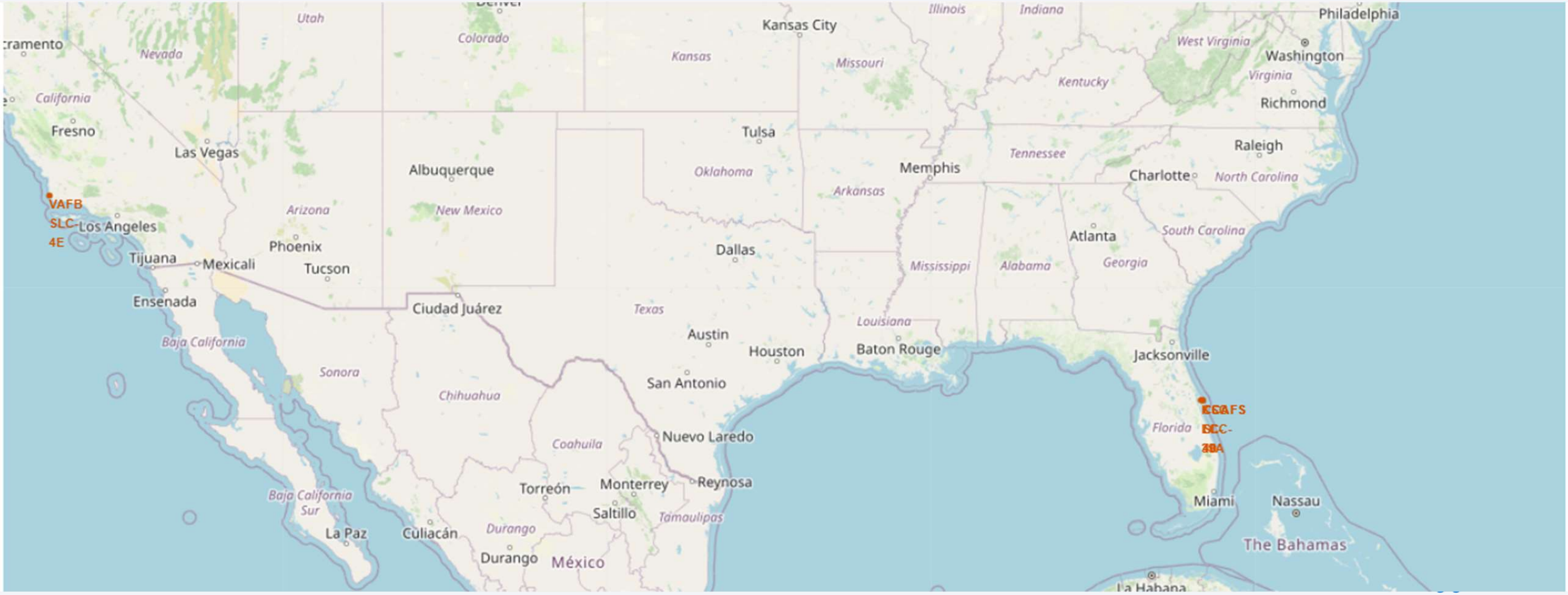
```
 * sqlite:///my_data1.db
Done.
```

Landing _Outcome	COUNT([Landing _Outcome])
Success	20
Success (drone ship)	8
Success (ground pad)	6

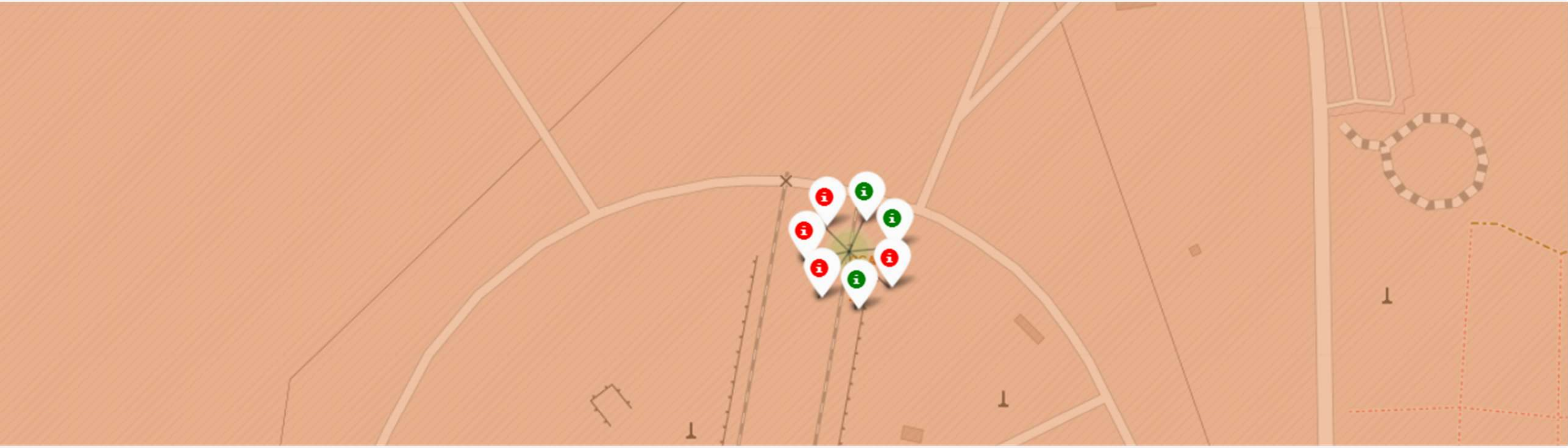
Section 3

Launch Sites Proximities Analysis

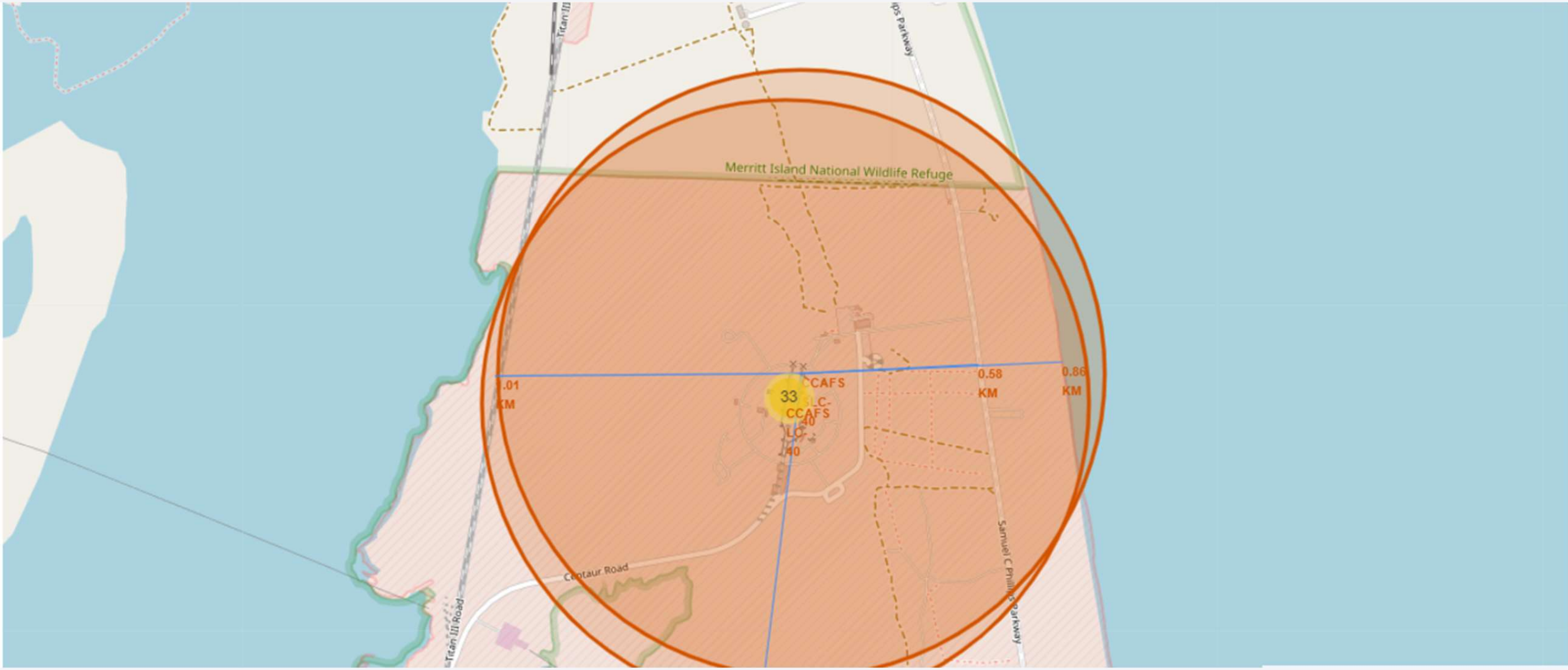
Lunch sites locations

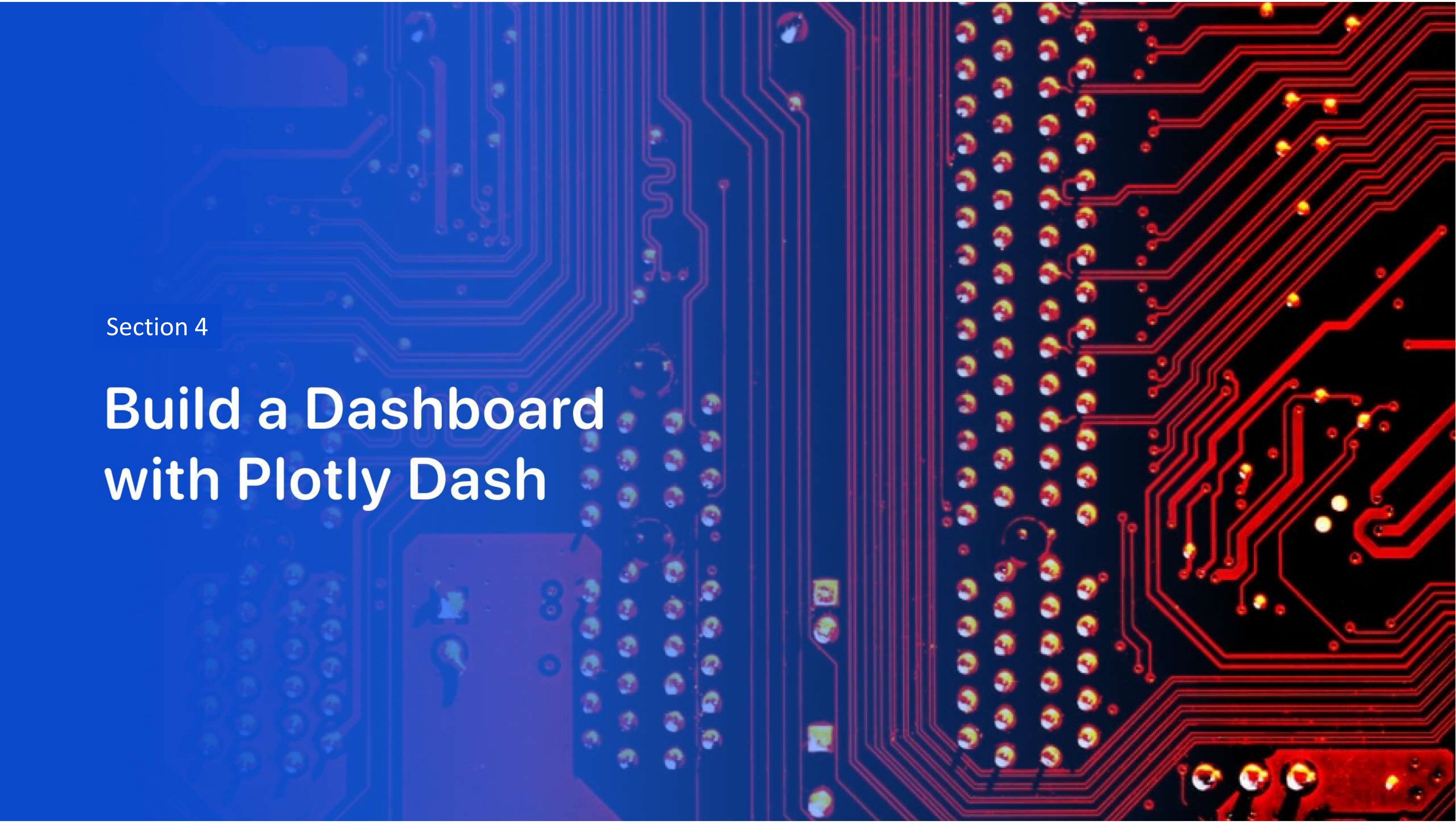


Lunch sites locations- colored success lunches



Lunch sites locations- distance

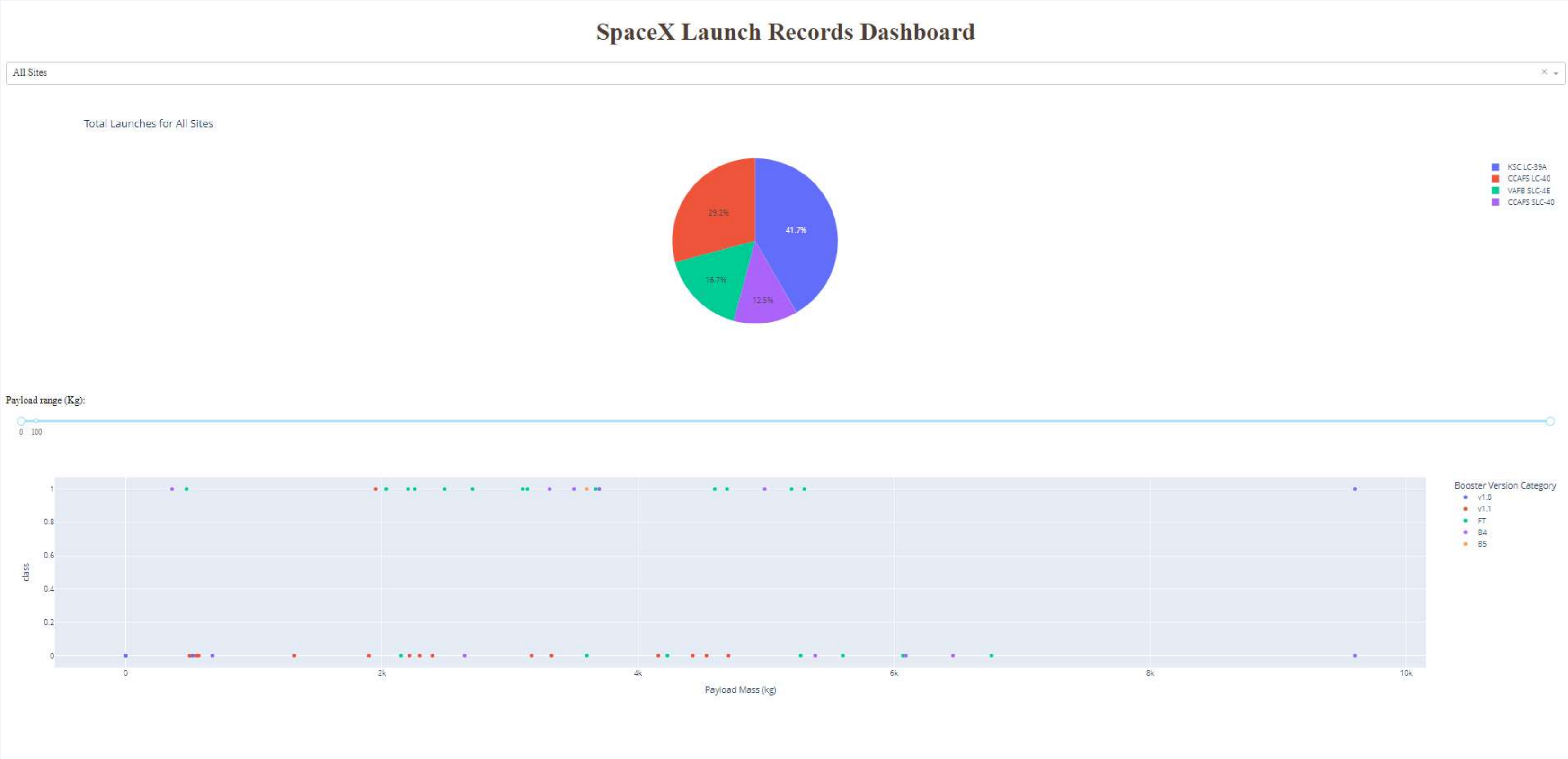




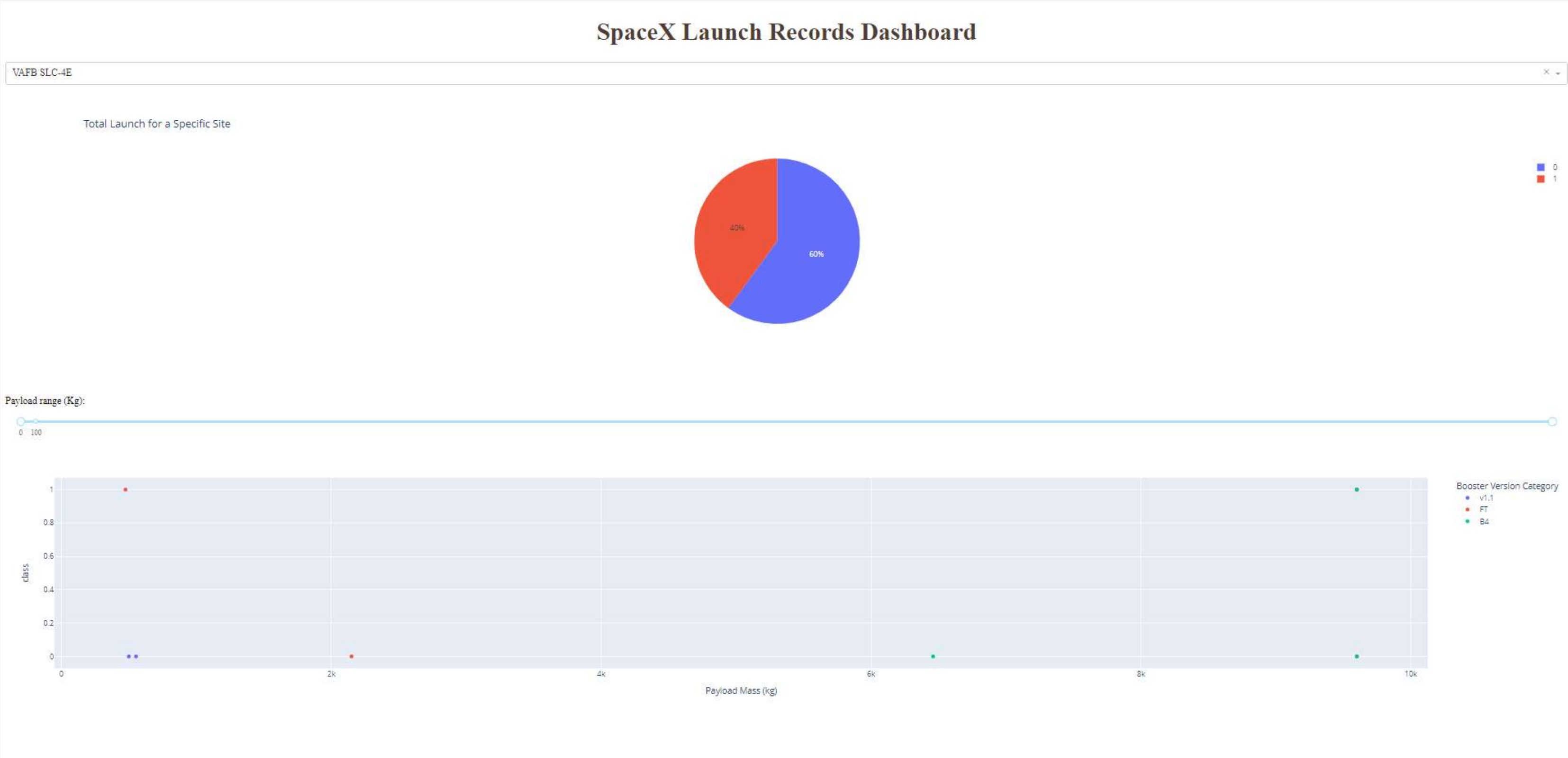
Section 4

Build a Dashboard with Plotly Dash

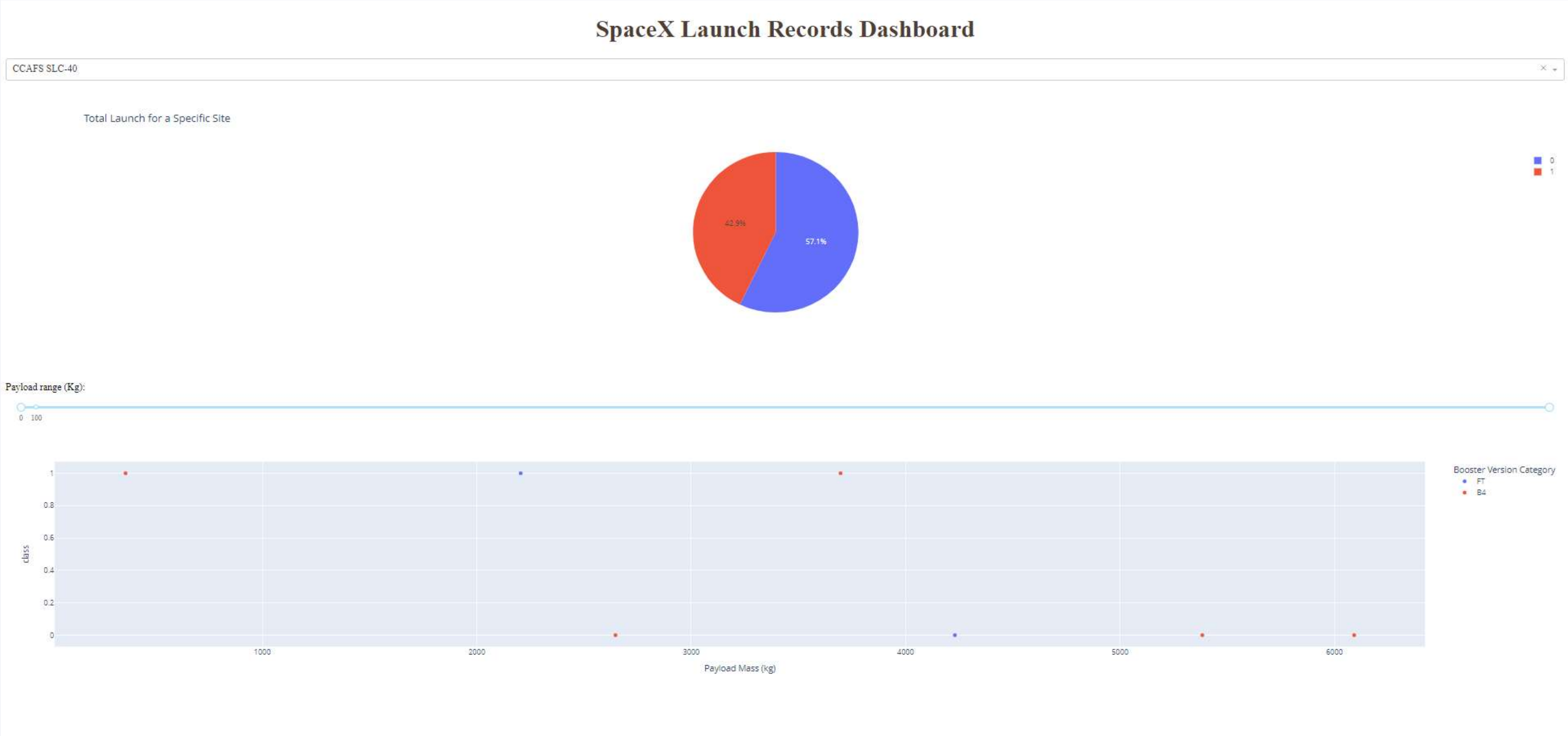
Dashboard- All sites



Dashboard- VAFB SLC- 4E



Dashboard- CCAFS SLC-40



Section 5

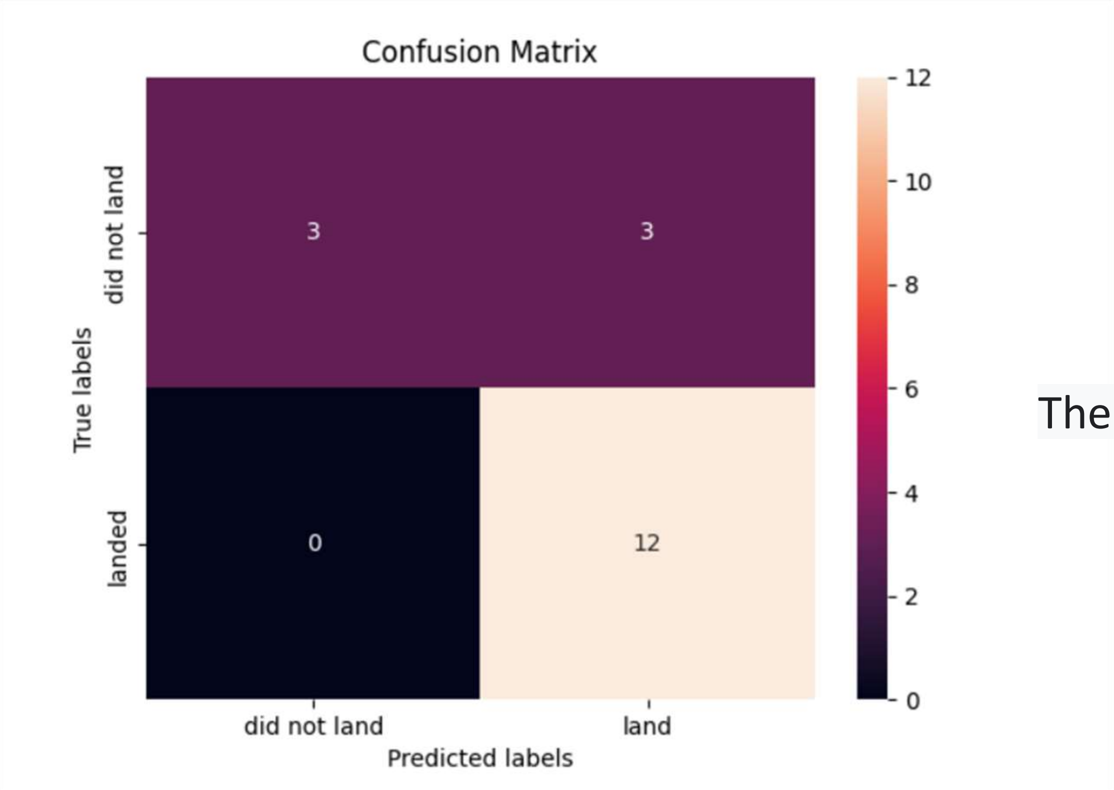
Predictive Analysis (Classification)

Classification Accuracy

The best accuracy has Decision Tree model with 0.89 accuracy. All other models have the same accuracy - 0.83.

	Method	Accuracy
0	LogisticRegression	0.833333
1	SVC	0.833333
2	DecisionTreeClassifier	0.888889
3	KNeighborsClassifier	0.833333

Confusion Matrix



The main problem is 3 false positives

Conclusions

- Lunch success rate increase year by year
- Orbits GEO, HEO, VLEO has the most success rate
- KSC LC-39A has the most succesful lunches
- The decision tree performs the best for this task

Appendix

- Pyhon code for dashboard CLICK


Thank you!