IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

<Miguel Laplana Marin>
<23/05/2022>

Table of Contents

Executive Summary

- **Summary of methodologies**
  - Data collection
  - Data wrangling
  - EDA with data visualization
  - EDA with SQL
  - Interactive map with Folium
  - Dashboard with Plotly Dash
  - Predictive analysis by classification model
- **Summary of all results**
  - EDA results
  - Interactive analytics
  - Predictive analysis

Introduction

The objective of this project is to use data in order to predict whether the SpaceX's 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 saving is because SpaceX can reuse the first stage. Therefore if we can determine if the first stage will land, we would be able to use this info in order determine the cost of next launches.

Section 1

Methodology

Methodology

Executive Summary

- **Data collection methodology:**
  - SpaceX Rest API / Web Scraping from Wikipedia
- **Perform data wrangling:**
  - Identify the categorical/numerical columns. Replace NULL values
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models

Data Collection

- Data was collected using request method to the SpaceX Rest API and also performing web scraping against wikipedia

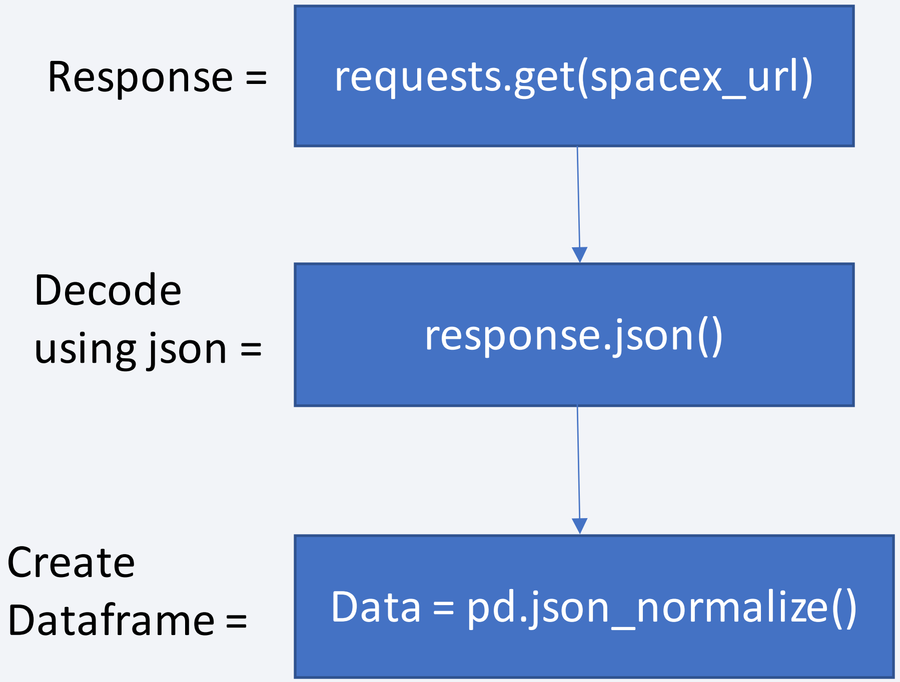
Now let's start requesting rocket launch data from SpaceX API with the following URL:

```
[6]: spacex_url="https://api.spacexdata.com/v4/launches/past"
```

```
[7]: response = requests.get(spacex_url)
```

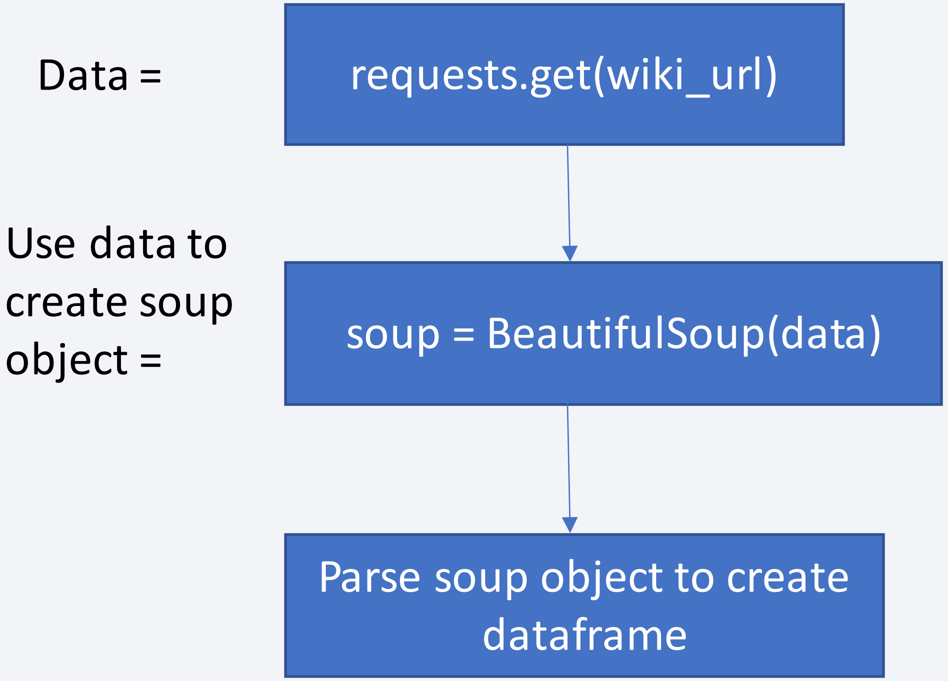
Data Collection – SpaceX API

- Collecting data using SpaceX REST calls
- [jupyter-labs-spacex-data-collection]



Data Collection - Scraping

- And we perform webscraping in the wikipedia page
- [jupyter-labs-webscraping]



Data Wrangling

- [labs-jupyter-spacex-Data-wrangling]()
- Checking missing values -> identifying categorical/numerical columns -> calculate number of lunches per site -> calculate number per orbit -> number and occurrence of mission per orbit -> landing outcome label -> Create a landing outcome label from Outcome column

```
0        None None
1        None None
2        None None
3       False Ocean
4        None None
            ...
85       True ASDS
86       True ASDS
87       True ASDS
88       True ASDS
89       True ASDS
```

Out[22]:

	Class
0	0
1	0
2	0
3	0
4	0
5	0
6	1
7	1

EDA with Data Visualization

- Scatter plots and Bar graphs for different features to see the relation
- FlightNumber vs PayloadMass
- PayloadMass vs LaunchSite
- Orbit vs Success Rate
- FlightNumber vs Orbit
- Orbit vs PayloadMass
- Year vs Success rate
- jupyter-labs-eda-datavisualization

EDA with SQL

SQL Queries:

- 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
- List the date when the first successful landing outcome in ground pad was achieved
- List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000
- List the total number of successful and failure mission outcomes
- List the names of the booster versions which have carried the maximum payload mass
- List the failed landing outcomes in drone ship, their booster versions, and launch site names for in year 2015
- 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
- jupyter-labs-eda-sql-

Build an Interactive Map with Folium

Interactive map created with folium, added circles and markers to make visualization useful and simple for the client. Used green markers if the land was successful , red if they failed

Build a Dashboard with Plotly Dash

Interactive dashboard created using dash.

Includes dropdown to select launch and range-slider in order to select the range of the payload mass. Shows an interactive pie chart of success rate for the launch sites and an scatter plot to show the relation of outcomes and payload in that range

spacex_dash_app

Predictive Analysis (Classification)

Create model to perform prediction.

- Get column of results to train the data ->
- Normalize our data (X)
- Split data into train and test sets
- Create multiple models like decision tree, linear regression.. Use gridSearch to find best hyperparameters
- Calculate accuracy of models with test data

SpaceXmachineLearningPrediction

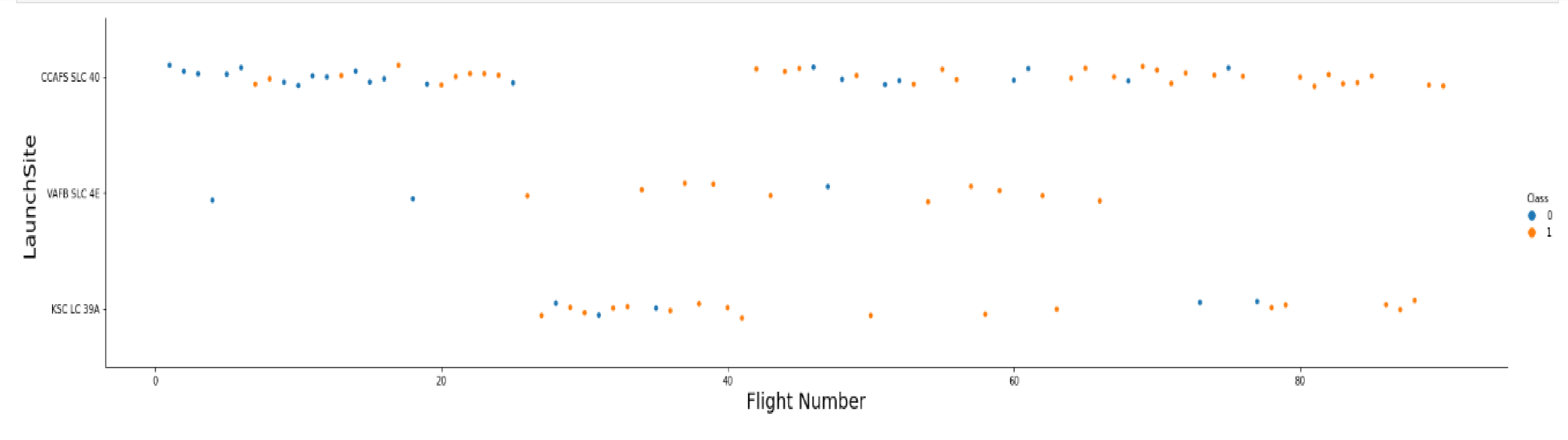
Results

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

Section 2

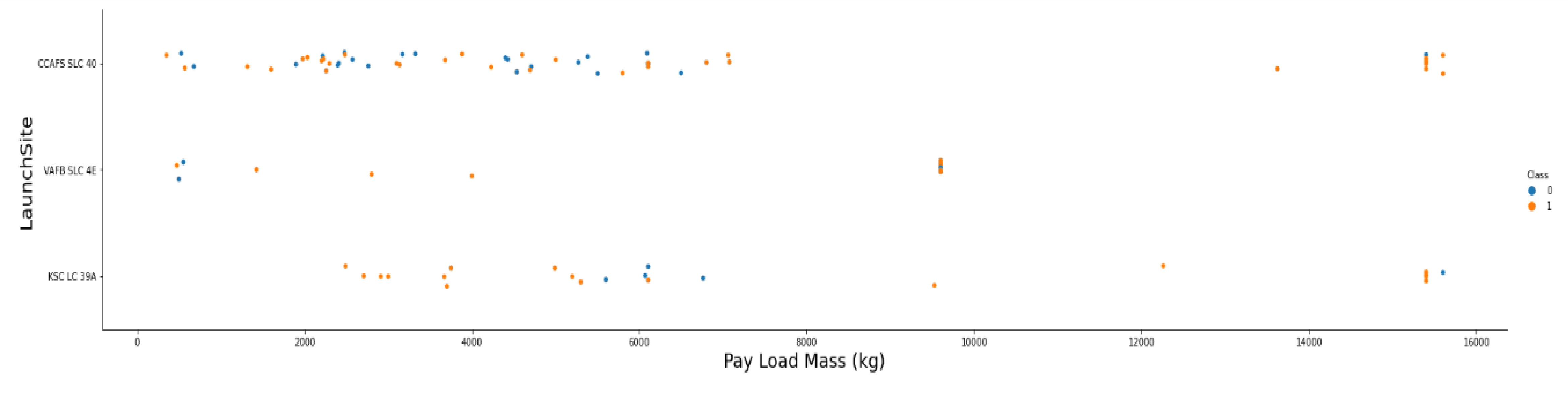
Insights drawn from EDA

Flight Number vs. Launch Site



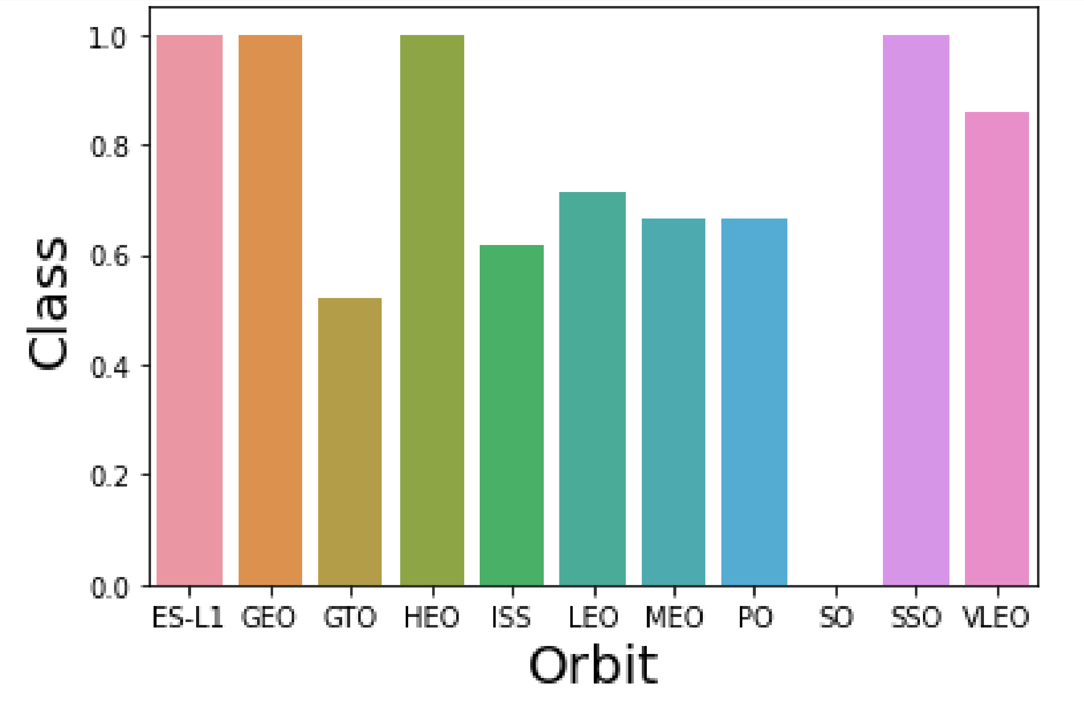
We see that different launch sites have different success rates

Payload vs. Launch Site



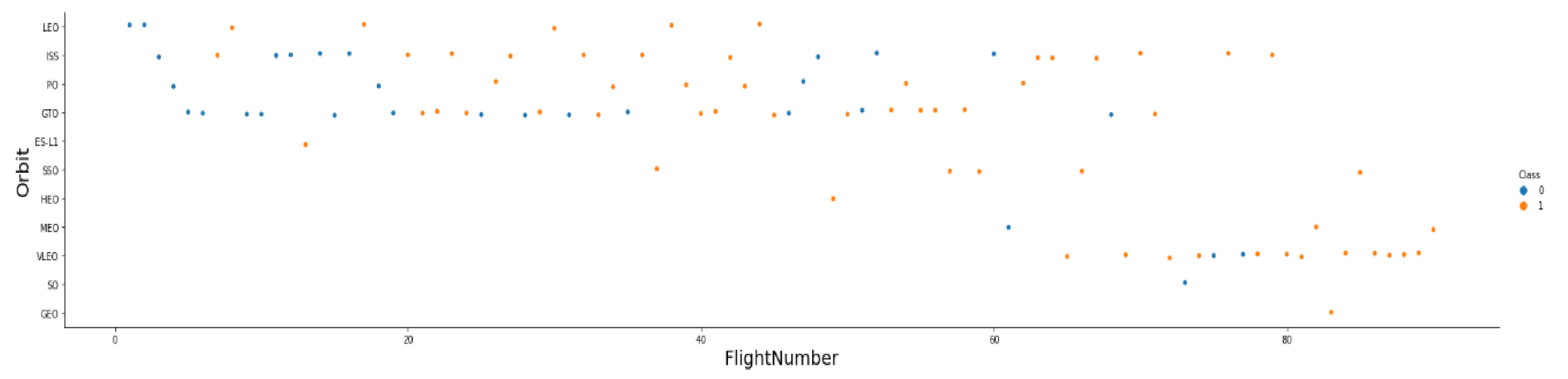
We see that in VAFB-SLC launch site there are no rockets launched for heavy payload mass(greater than 10000).

Success Rate vs. Orbit Type



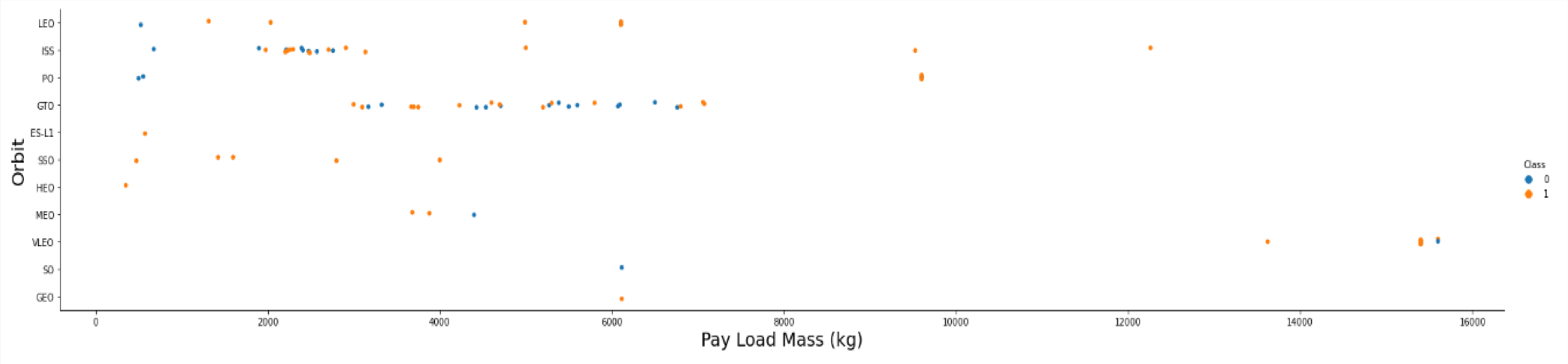
There is greater success rate in Orbits like ES-l1, GEO, HEO, or SSO

Flight Number vs. Orbit Type



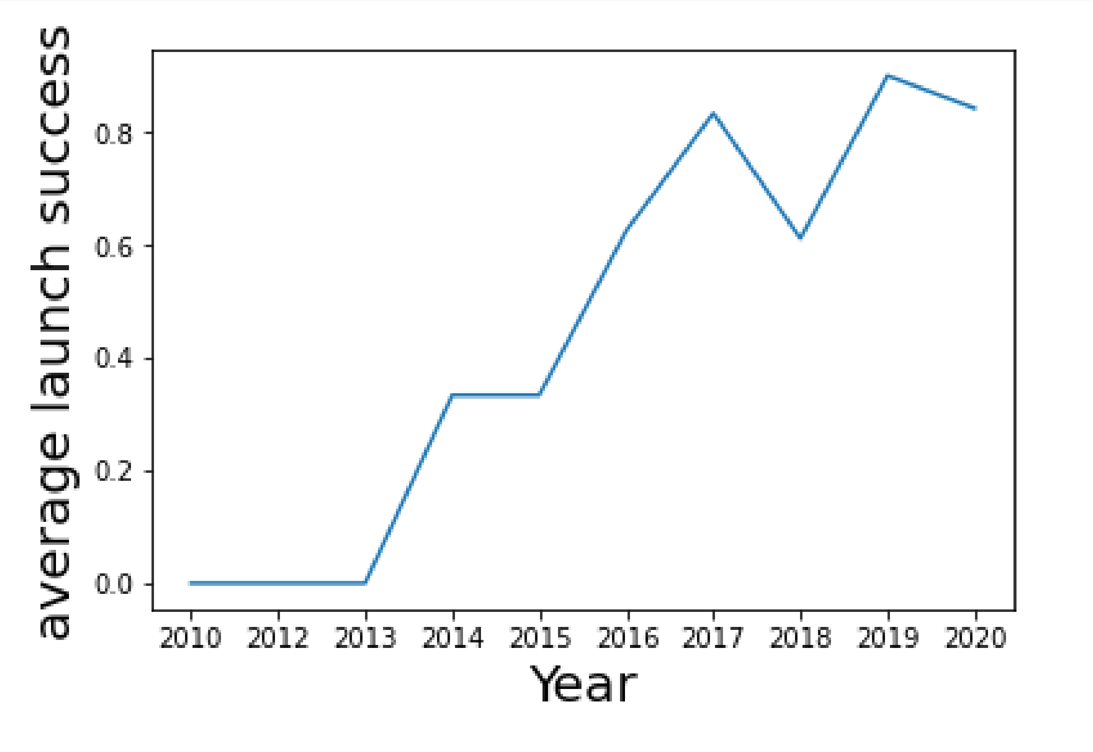
In the LEO orbit the Success appears related to the number of flights; on the other hand, there seems to be no relationship between flight number when in GTO orbit.

Payload vs. Orbit Type



With heavy payloads the successful landing or positive landing rate are more for Polar,LEO and ISS.

Launch Success Yearly Trend



We can see that average launch success has increased since 2013 till 2020

All Launch Site Names

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

```
[9]: %sql select distinct(launch_site) from spacexdataset
```

```
 * ibm_db_sa://phv23334:***@0c77d6f2-5da9-48a9-81f8-86b520b8
Done.
```

[9]:

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

Launch Site Names Begin with 'CCA'

Task 2

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

[11]:
```
%sql select * from spacexdataset where launch_site like 'CCA%' limit 5
```

```
 * ibm_db_sa://phv23334:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/bludb
Done.
```

[11]:

DATE	time__utc_	booster_version	launch_site	payload	payload_mass__kg_	orbit	customer	mission_outcome	landing__outcome
2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
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	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
2012-10-08	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
2013-03-01	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

Total Payload Mass

Task 3

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

```
[17]: %sql select sum(payload_mass__kg_) from spacexdataset where customer like 'NASA (CRS)'

 * ibm_db_sa://phv23334:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.da
Done.
```

[17]:

1
45596

Average Payload Mass by F9 v1.1

Task 4

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

[25]:
```
%sql select AVG(payload_mass__kg_) from spacexdataset where booster_version like 'F9 v1.1%'
```

```
 * ibm_db_sa://phv23334:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databas
Done.
```

[25]:

1
2534

First Successful Ground Landing Date

Task 5

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

Hint:Use min function

```
%sql select min(DATE) from spacexdataset where landing__outcome like 'Success (ground pad)'
```

```
 * ibm_db_sa://phv23334:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databas
Done.
```

1
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 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

```
%sql select distinct(booster_version) from spacexdataset where (landing__outcome like 'Success (drone ship)') and (payload_mass__kg_ > 4000 and payload_mass__kg_ < 6000)
```

```
 * ibm_db_sa://phv23334:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/bludb
Done.
```

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

Total Number of Successful and Failure Mission Outcomes

Task 7

List the total number of successful and failure mission outcomes

```
%sql select count(*) from spacexdataset where mission_outcome like '%Failure%'
```

```
 * ibm_db_sa://phv23334:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8
Done.
```

1
1

```
%sql select count(*) from spacexdataset where mission_outcome like '%Success%'
```

```
 * ibm_db_sa://phv23334:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8
Done.
```

1
100

Boosters Carried Maximum Payload

Task 8

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

```
%sql select distinct(booster_version) from spacexdataset where payload_mass__kg_ = (select max(payload_mass__kg_) from spacexdataset)
```

```
 * ibm_db_sa://phv23334:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/bludb
Done.
```

booster_version
F9 B5 B1048.4
F9 B5 B1048.5
F9 B5 B1049.4
F9 B5 B1049.5
F9 B5 B1049.7
F9 B5 B1051.3
F9 B5 B1051.4
F9 B5 B1051.6
F9 B5 B1056.4
F9 B5 B1058.3
F9 B5 B1060.2
F9 B5 B1060.3

2015 Launch Records

Task 9

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

```
%sql select booster_version, launch_site, landing__outcome from spacexdataset where landing__outcome like 'Failure (drone ship)' and DATE like '2015%'
```

```
 * ibm_db_sa://phv23334:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/bludb
Done.
```

booster_version	launch_site	landing__outcome
F9 v1.1 B1012	CCAFS LC-40	Failure (drone ship)
F9 v1.1 B1015	CCAFS LC-40	Failure (drone ship)

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

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

```
%sql select landing__outcome, count(landing__outcome) from spacexdataset where DATE between '2010-06-04' and '2017-03-20' and landing__outcome like 'Success%' group by landing__outcome
```

```
 * ibm_db_sa://phv23334:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/bludb
Done.
```

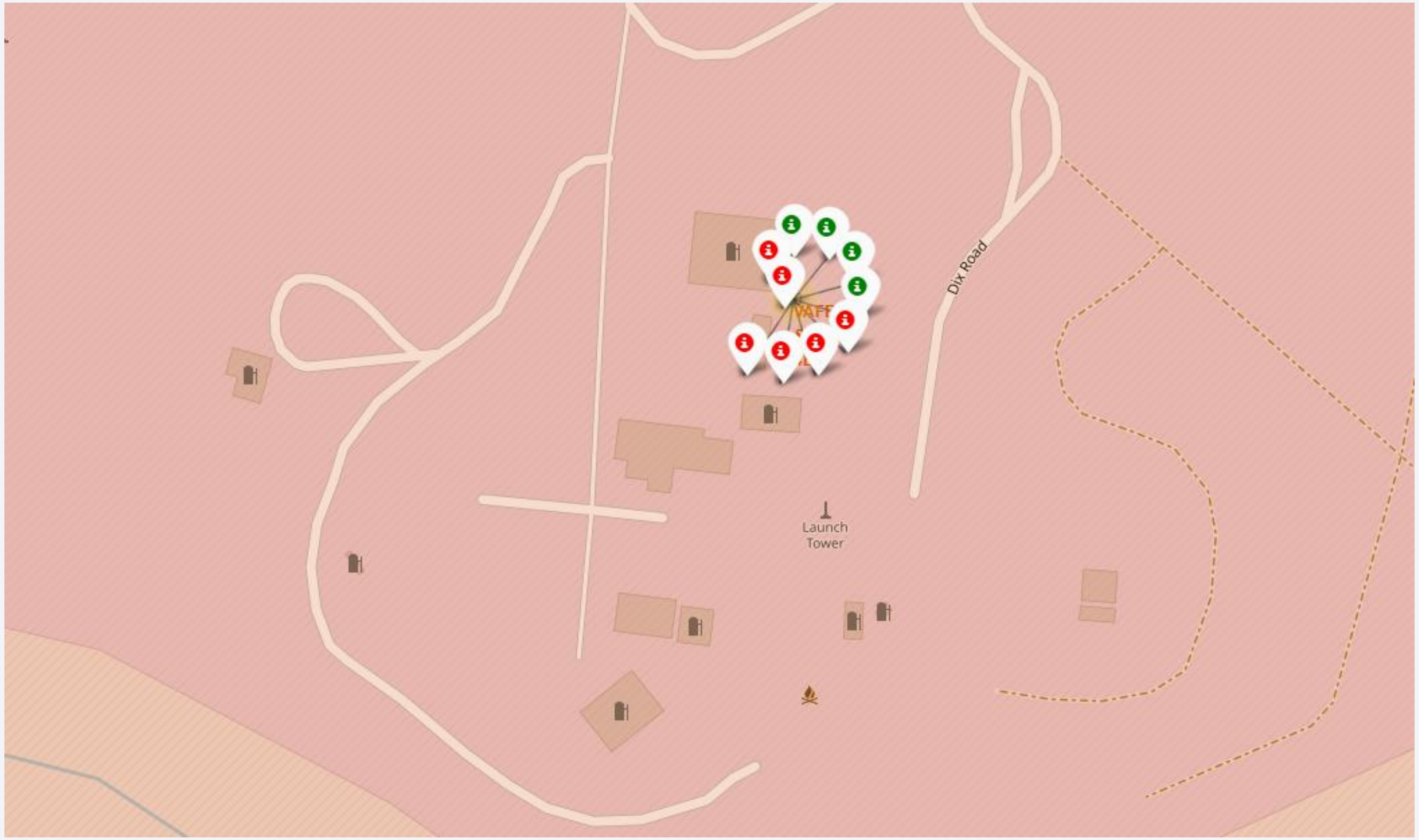
landing__outcome	2
Success (drone ship)	5
Success (ground pad)	3

Section 3

Launch Sites Proximities Analysis

<Folium Map Screenshot 1>

<Folium Map Screenshot 2>



<Folium Map Screenshot 3>

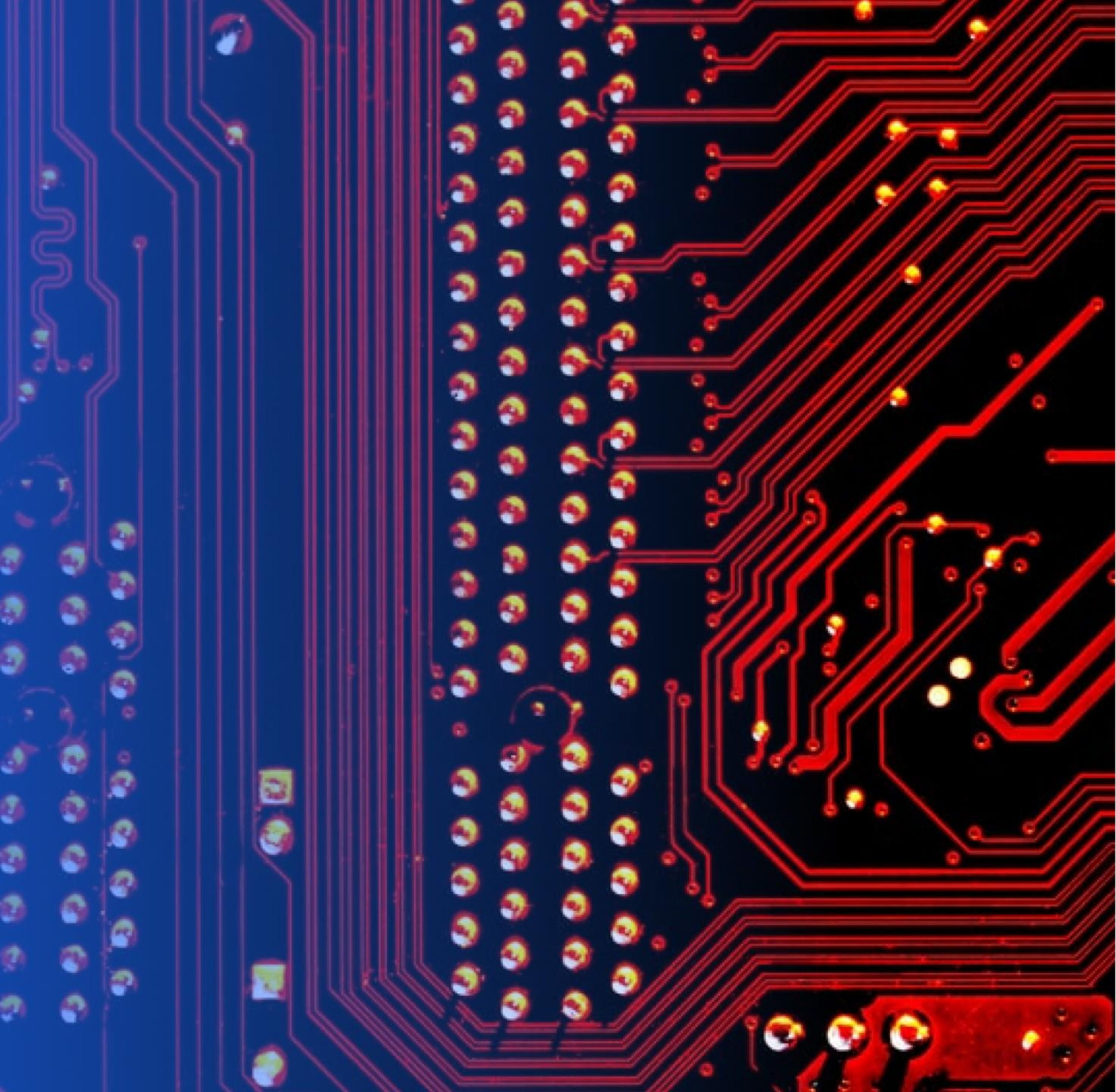
- Replace <Folium map screenshot 3> title with an appropriate title

- Explore the generated folium map and show the screenshot of a selected launch site to its proximities such as railway, highway, coastline, with distance calculated and displayed

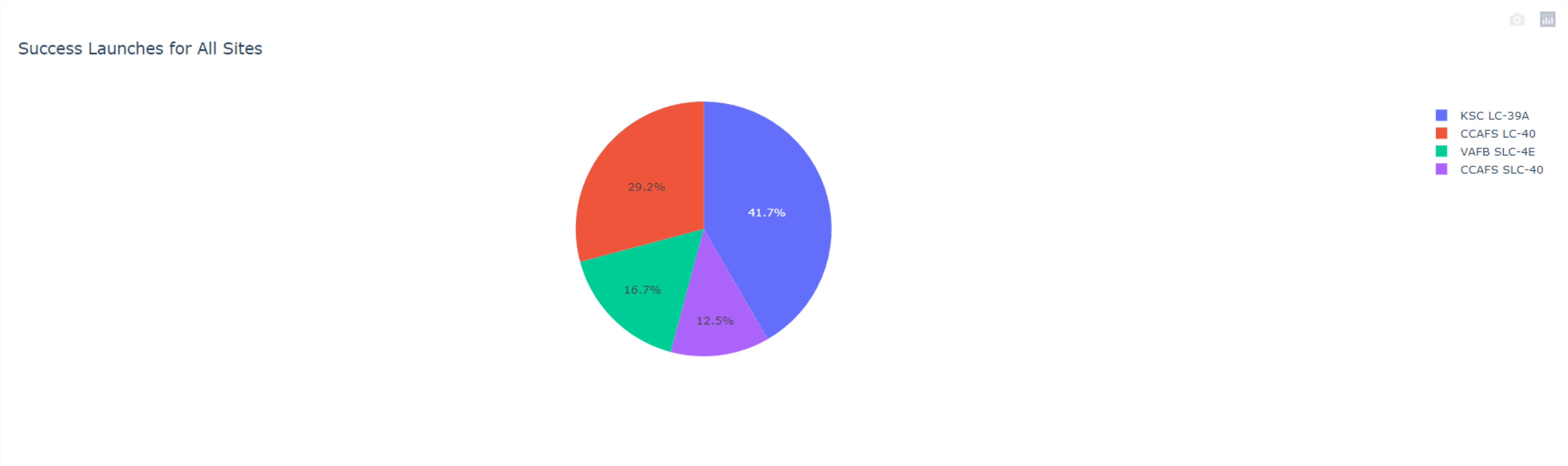
- Explain the important elements and findings on the screenshot

Section 4

Build a Dashboard with Plotly Dash

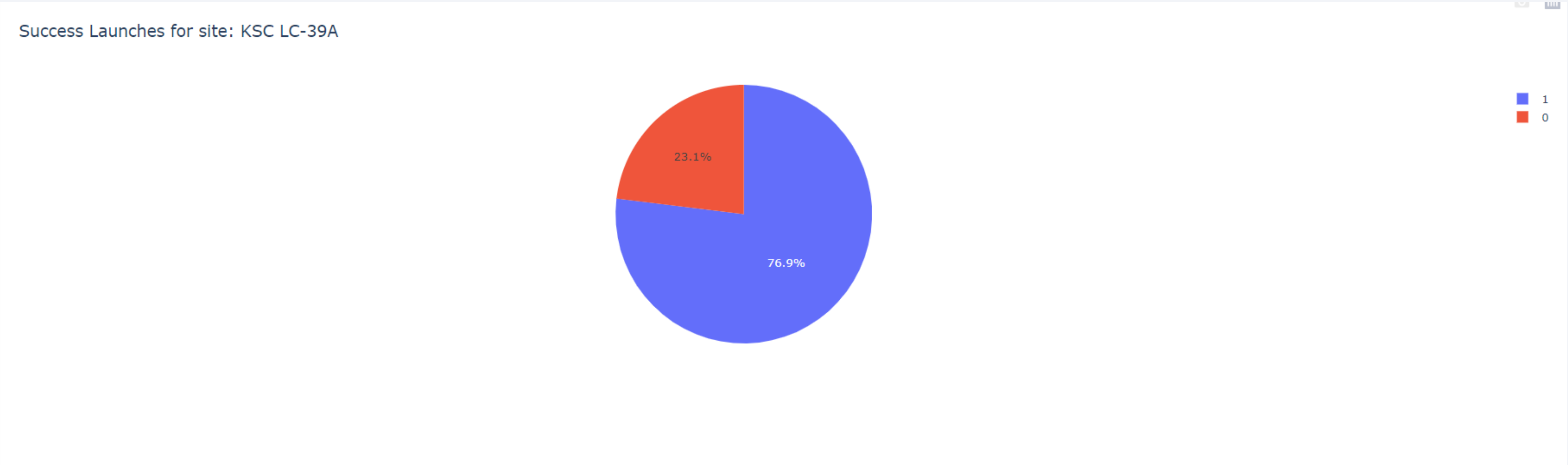


<Dashboard Screenshot 1>



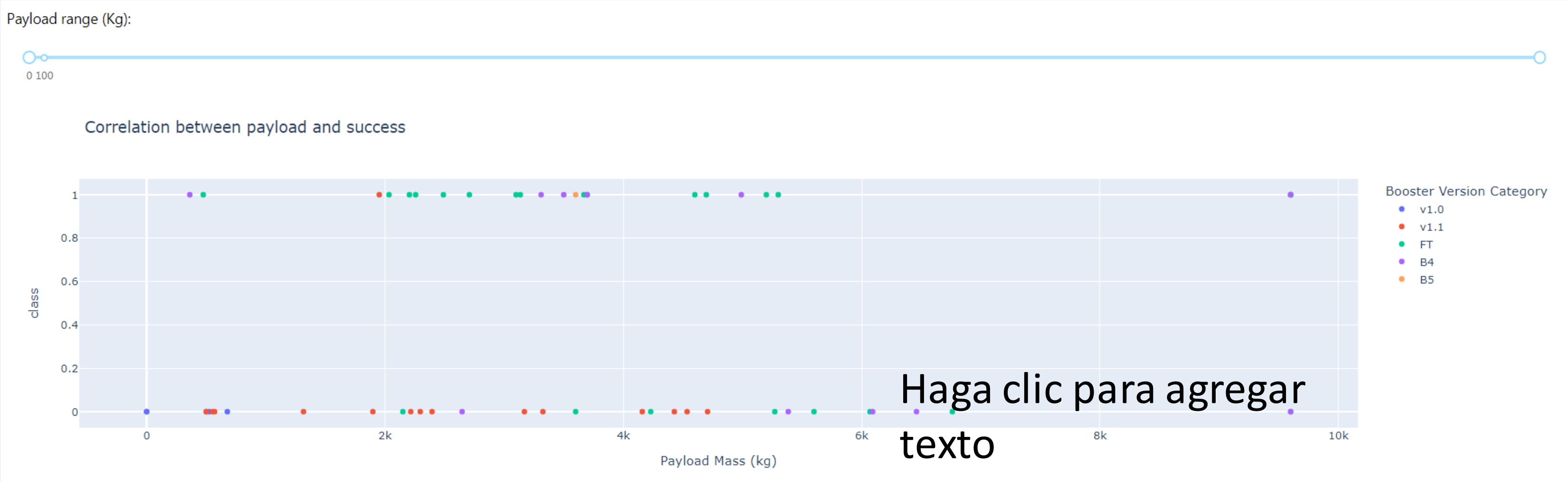
We can see KSC LC-39A site has the highest success while CCAFS SLC-40 has the lowest

<Dashboard Screenshot 2>

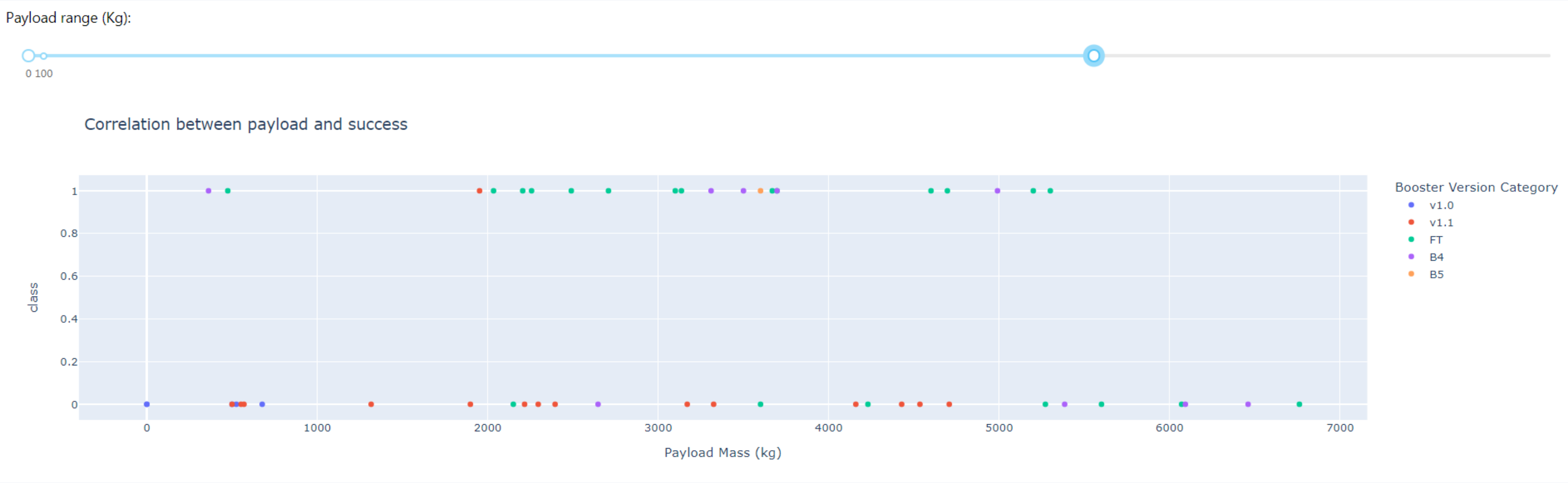


Success rate = 76.9% in the site KSC LC-39A which is the highest success rate site

<Dashboard Screenshot 3>



Haga clic para agregar texto

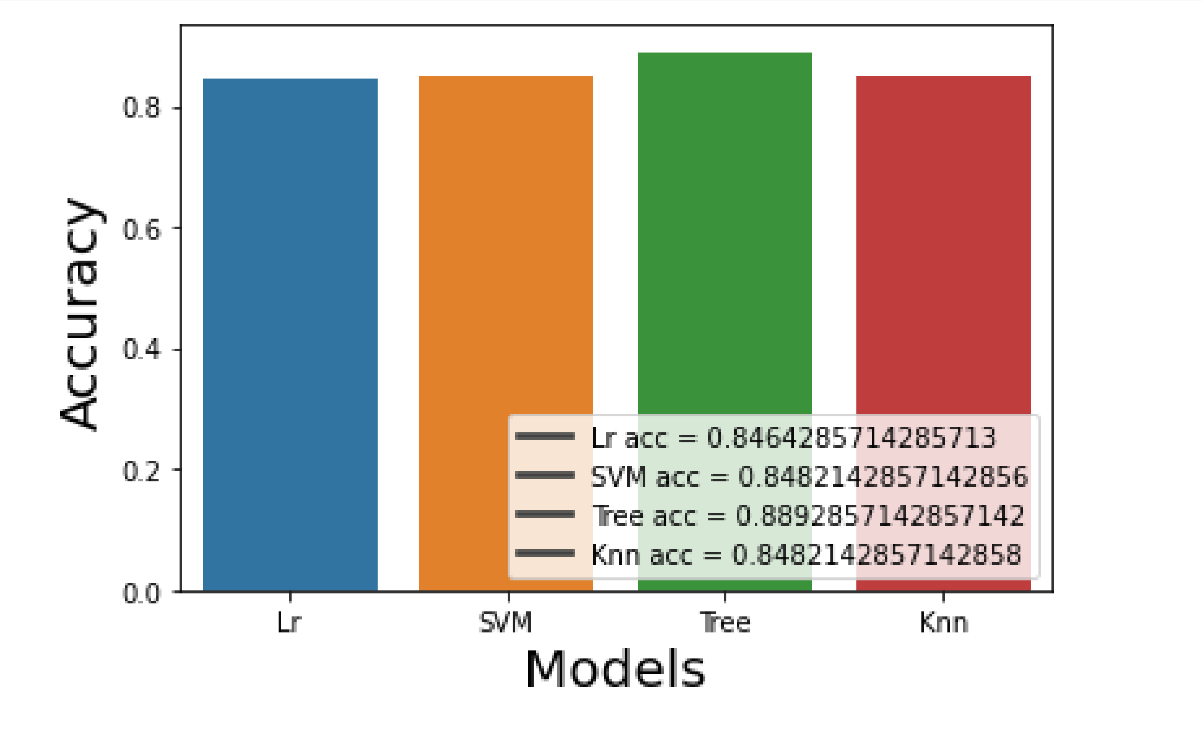


We can see that the FT booster and the payload mass in range (2000-4000) have the highest success rate.

Section 5

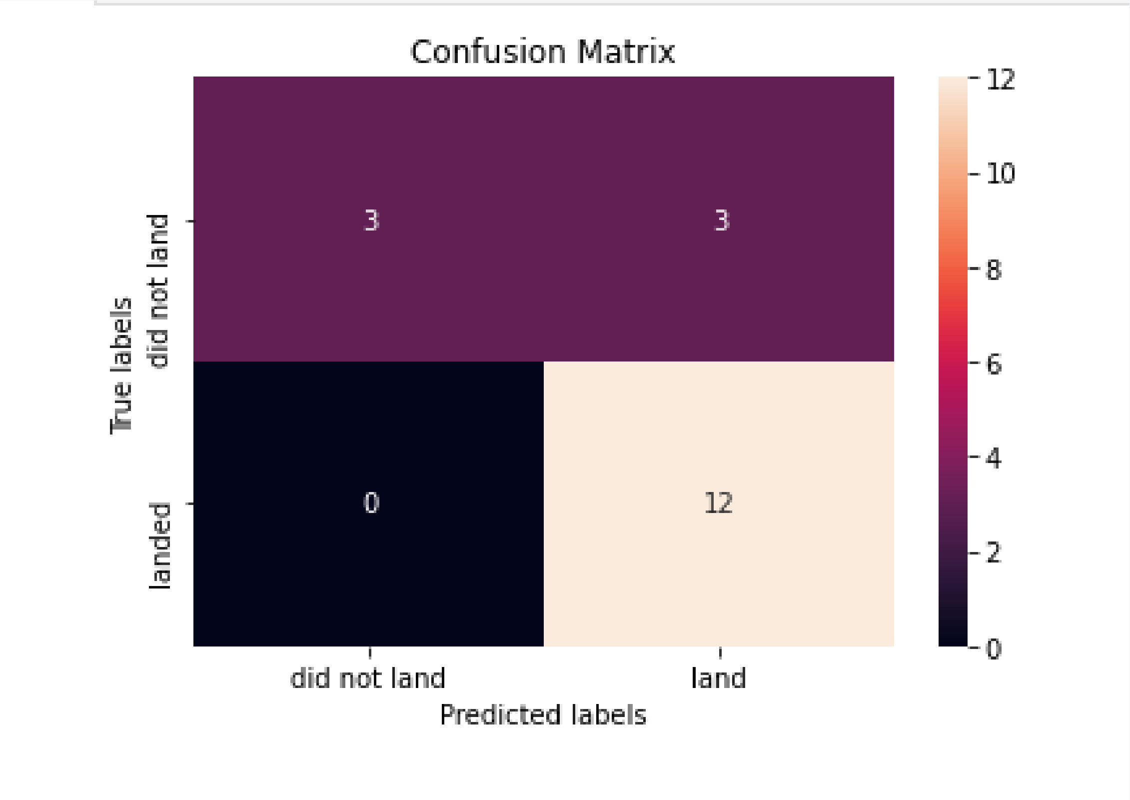
Predictive Analysis (Classification)

Classification Accuracy



We can see that decision Tree has the best accuracy

Confusion Matrix



Confusion Matrix for our decision Tree model, we see that most of our predictions are correct but we have a few false positives.

Conclusions

We have been able to create a good model that predicts the first stage landing of the rockets with great accuracy. The model that predicted better was a decision tree.

We have seen that heavier payloads perform better than the lighter ones, also that the success rate changes depending on factors like the different types of orbit or the launches sites.

Finally we can appreciate that the success rate on these landings have increased over the years since 2013

Appendix

Github link to the project files and information :

https://github.com/SKTPausanias/Applied-DataScience-Capstone

Link to the course:

https://www.coursera.org/learn/applied-data-science-capstone

Thank you!