IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Murat ÇELEBI
25.03.2023

Outline

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

Executive Summary

- Summary of methodologies
  - Data Collection,
  - Web scraping,
  - EDA,
  - Site Location
  - Prediction with Machine Learning
- Summary of all results
  - Accuracy Score: 0.83

Introduction

- Project background and context
  - Project is an end-to-end data science project where we are investigating whether the first stage landing of the Spacex Falcon 9 rocket will be successful.
- Problems you want to find answers
  - Launched rockets are a huge cost element. Therefore, companies have placed the cost item on the top of their success.
  - The first question we need to answer in this project will be:
  - Will the first stage of a launched rocket land successfully again?

Section 1

Methodology

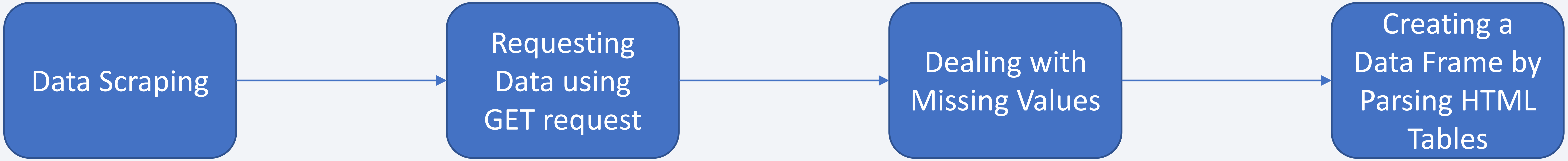
Methodology

Executive Summary

- Data collection methodology:
  - Describe how data was collected
- Perform data wrangling
  - Describe how data was processed
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - How to build, tune, evaluate classification models

Data Collection

- Describe how data sets were collected.
    - I gathered the data by performing a fetch request from Wikipedia to the SpaceX API.
- You need to present your data collection process use key phrases and flowcharts



Data Collection – SpaceX API

Github URL:

https://l24.im/HnD

https://l24.im/3qO8L

Data Collection - Scraping

- Requesting the Falcon 9 Launch Wiki page from its URL
- Extracting all column/variable names from HTML table header
- Creating a dataframe by parsing initialization HTML tables
- Github URL:
- https://l24.im/2SPp

Web Scraping Result Response object

```
response = requests.get(static_url).text
```

Normalize

```
pd.json_normalize(requests.get(static_url))
```

A BeautifulSoup object from the HTML response

```
soup = BeautifulSoup(response)
```

Extracting column names from HTML table header

```
html_tables = soup.find_all(name = 'table')
```

Data Wrangling

- We start by uploading our dataset. Then we perform a missing value analysis on the data set. Then we label the target variable as 0 and 1.

```
df.head(10)
```

```
df.isnull().sum()/df.count()*100
```

```
df.dtypes
```

```
df['LaunchSite'].value_counts()
```

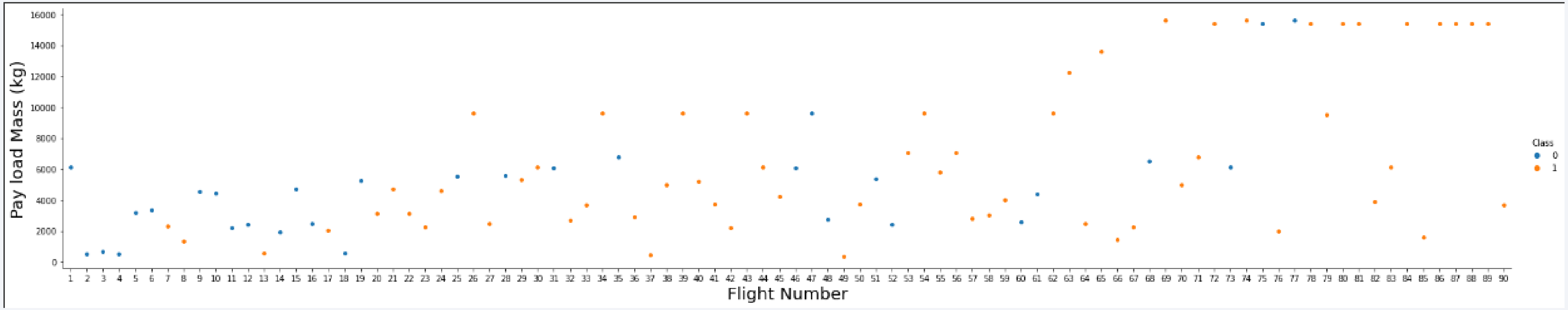
```
df['Orbit'].value_counts()
```

```
landing_class= []

for i in range(len(df["Outcome"])):
    if df["Outcome"][i] in bad_outcomes:
        landing_class.append(0)
    else:
        landing_class.append(1)

print(len(landing_class))
print(landing_class)
```

Github URL: https://l24.im/RJ7r

EDA with Data Visualization

- He designed the catplot chart to visualize the weight of the weight in each flight unit and whether, depending on the series of flights, the rocket at that weight did not successfully land on its first step.



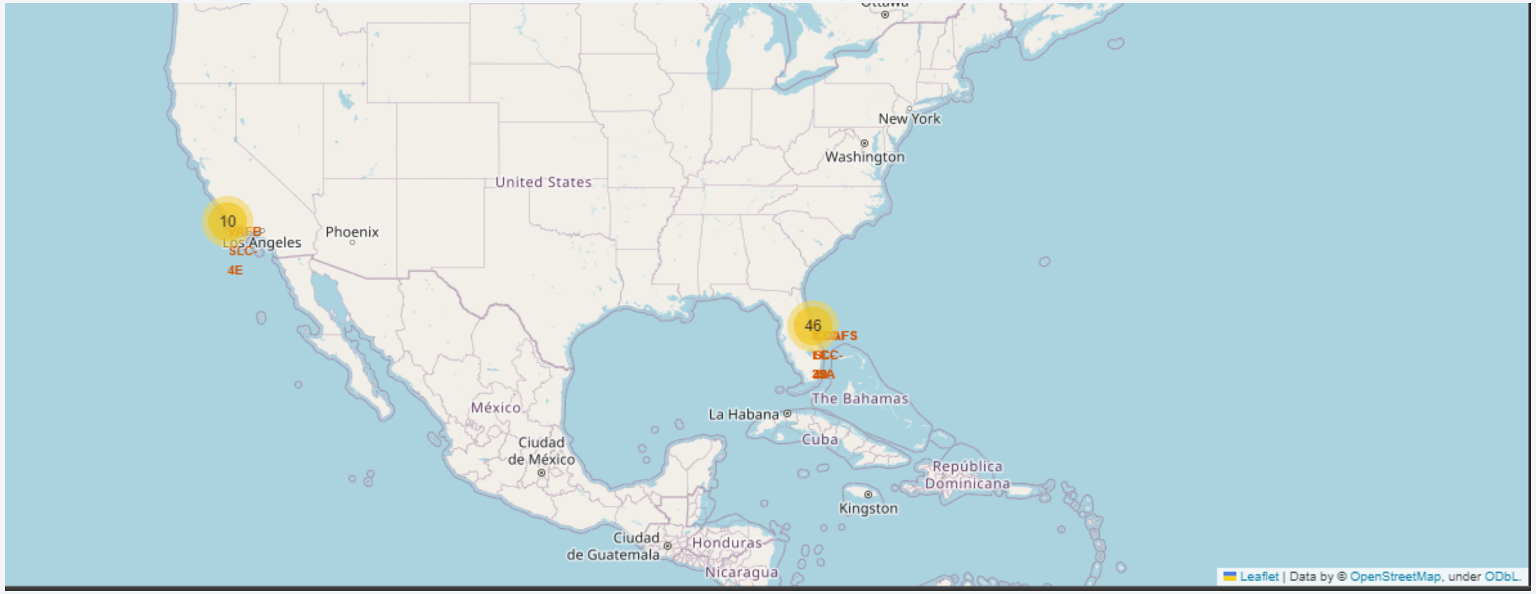
- Github URL: https://l24.im/NnQPD6

EDA with SQL

- The names of the unique launch sites in the space mission have been questioned.
- Questioning the total mass of payload carried by boosters launched by NASA (CRS).
- The date on which the first successful landing result was obtained on the ground pad was questioned.
- The total number of successful and unsuccessful mission results was questioned.
- Github URL: https://l24.im/12j

Build an Interactive Map with Folium

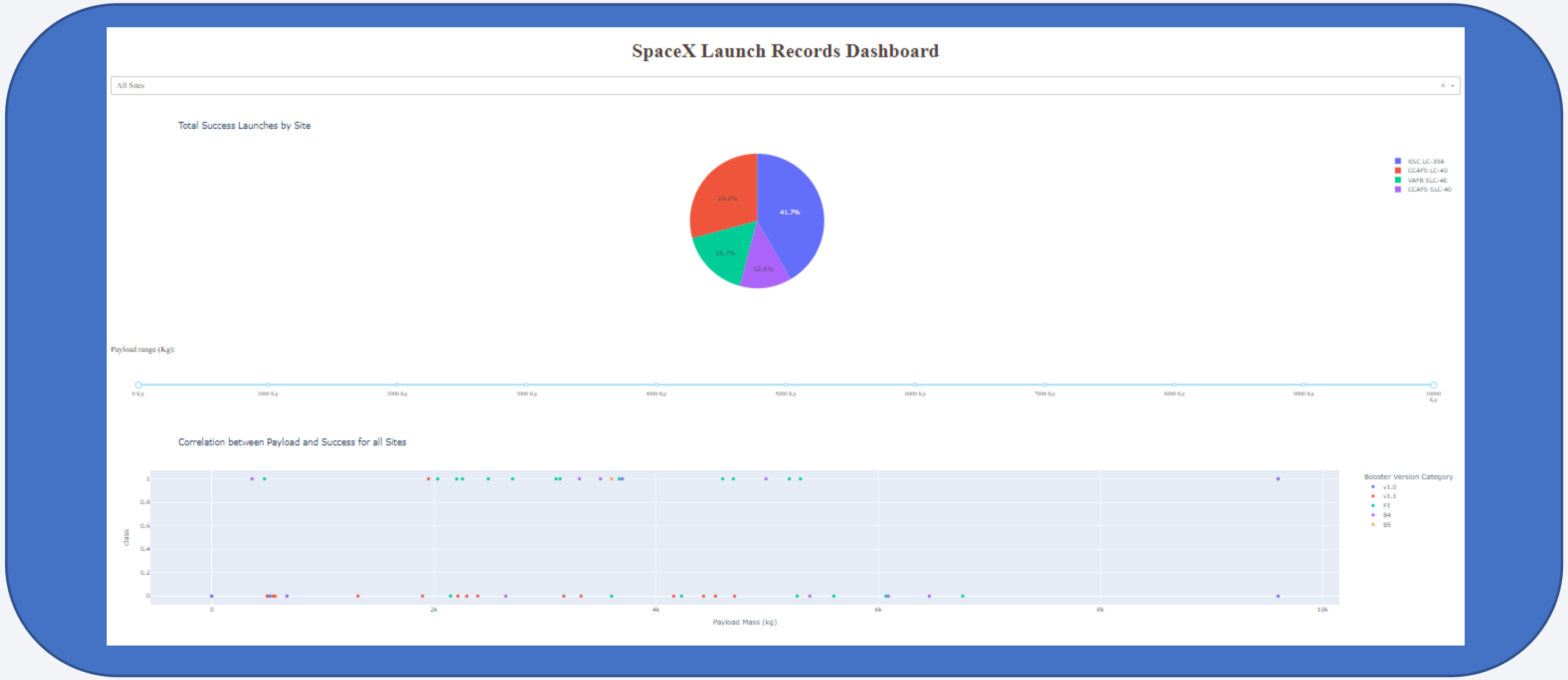
- I added markers to the leaf map, and all other objects to find launch pads on the world map and perform analysis.



- Github URL: https://l24.im/AE0e

Build a Dashboard with Plotly Dash

- The distribution of graphs and drawings in our data and the drawings on the Dashboard have been retained for live display in summary.



- Github URL: https://l24.im/yad

Predictive Analysis (Classification)

- I used five or four classification algorithms in the project and evaluated the results with Accuracy Score.

Loading the dataset → Splitting the dataset into X_train, X_test, Y_train, Y_test → Performing hyperparameter optimization for each algorithm → The success of each algorithm was evaluated with the Accuracy Score.

- Github URL:
- https://l24.im/thVYZ1

Results

- Exploratory data analysis results
- In Exploratory Data Analysis, it was observed that missing values, correlation between variables and successful first landings increased more over the years.

- Predictive analysis results

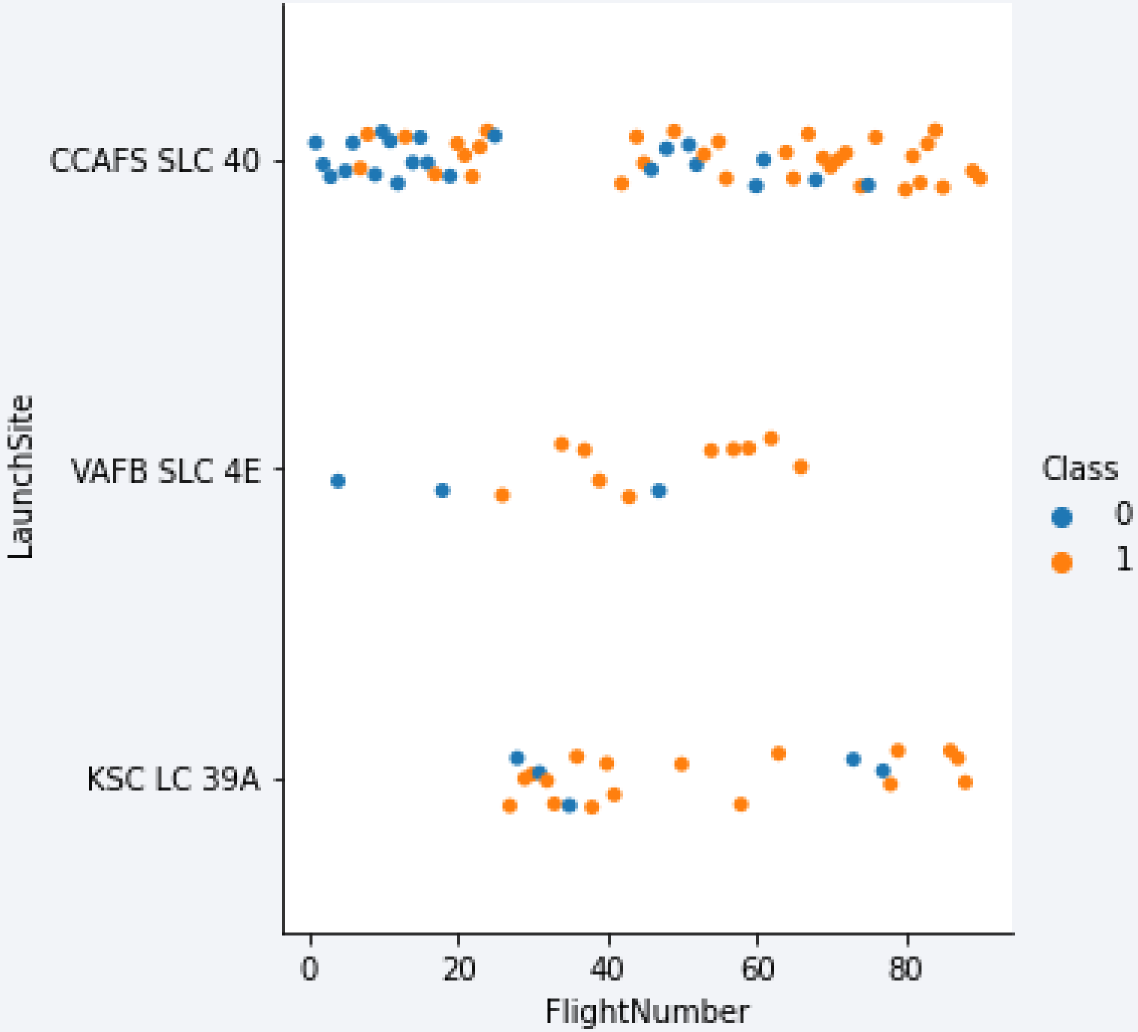
	Model	Accuracy Score (Hyperparameter Optimization)
0	Logistic Regression	0.833333
1	Support Vector Machine	0.833333
2	Decision Tree	0.888889
3	K Nearest Neighbor	0.833333

Section 2

Insights drawn from EDA

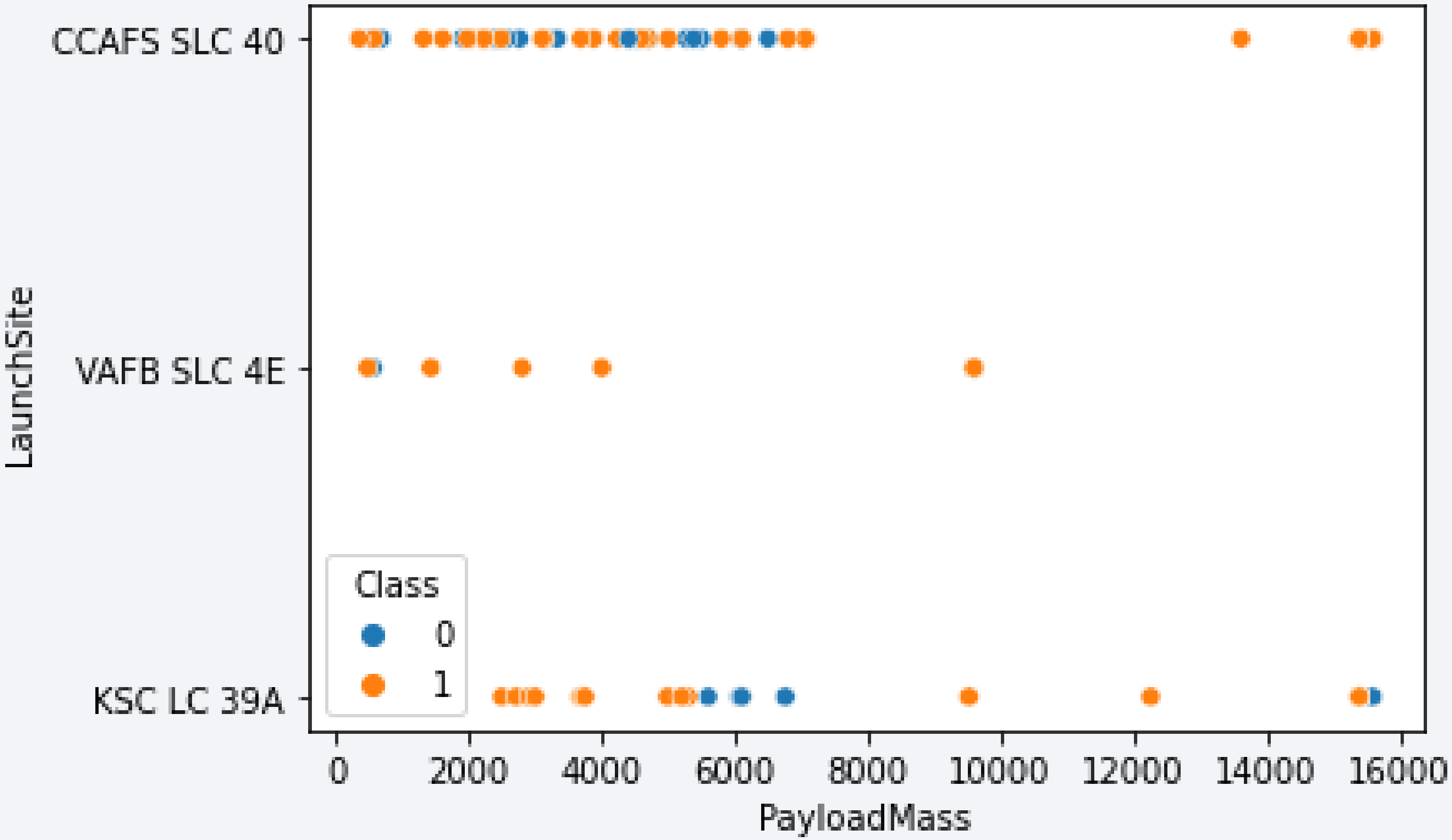
Flight Number vs. Launch Site

- LaunchSite: CCAF5 SLC 40 seems to have more failed landings.
- LaunchSite: KSC LC 39A has more successful landings.



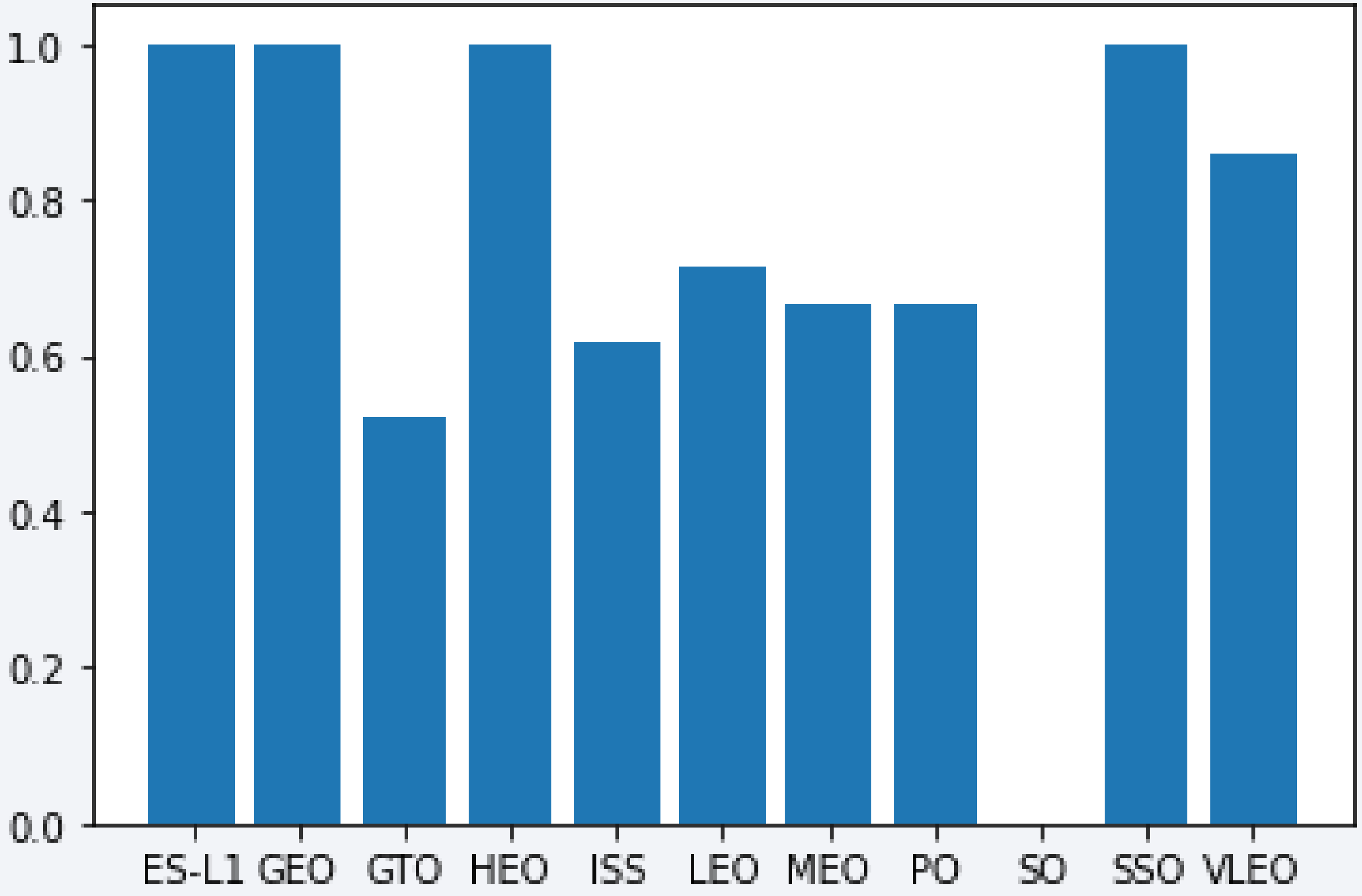
Payload vs. Launch Site

- Flights with payload masses between 2000 and 4000 appear mostly from LaoundchSite: CCAFS SLC 40. And it is observed that most of the flights from here make successful landings.



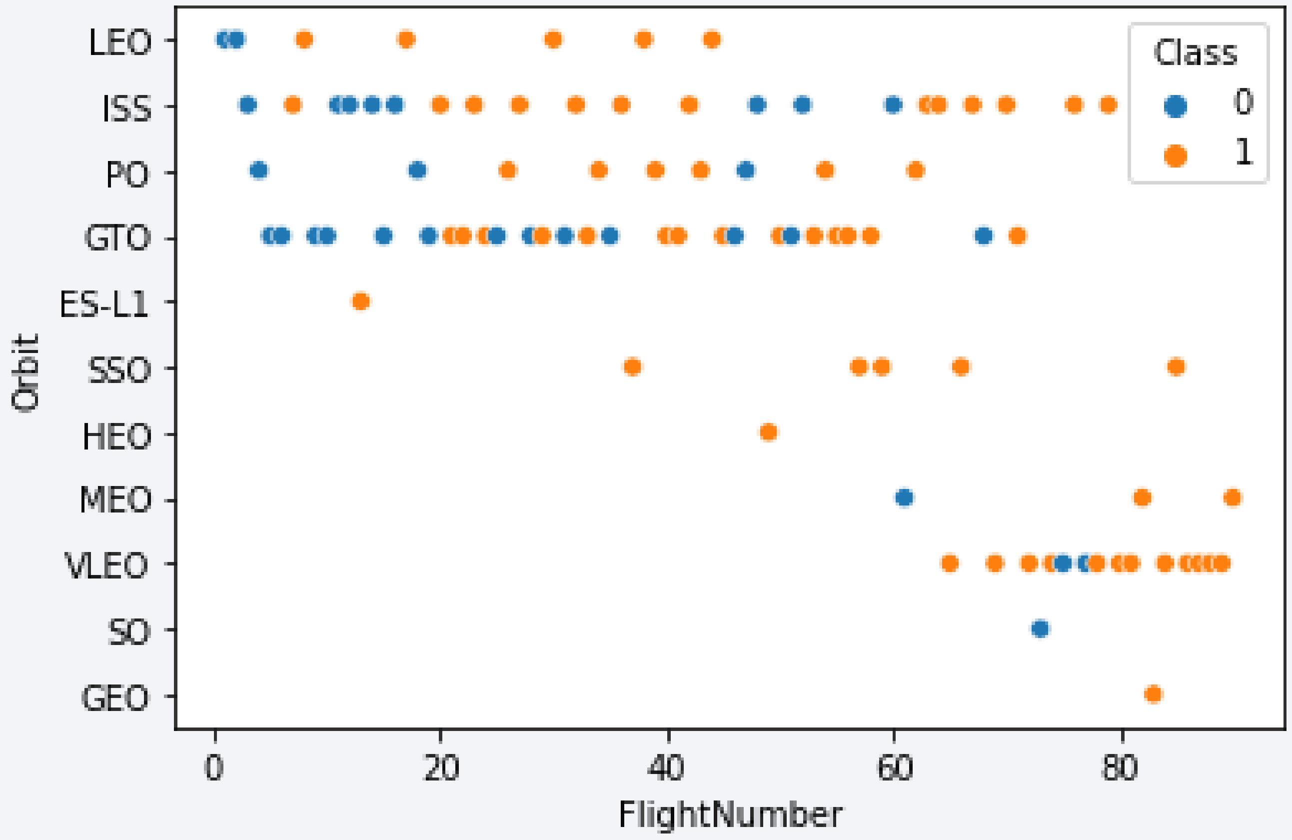
Success Rate vs. Orbit Type

- Orbit Type: All flights with ES-L1, GEO, HEO and SSO appear to have a successful first landing.



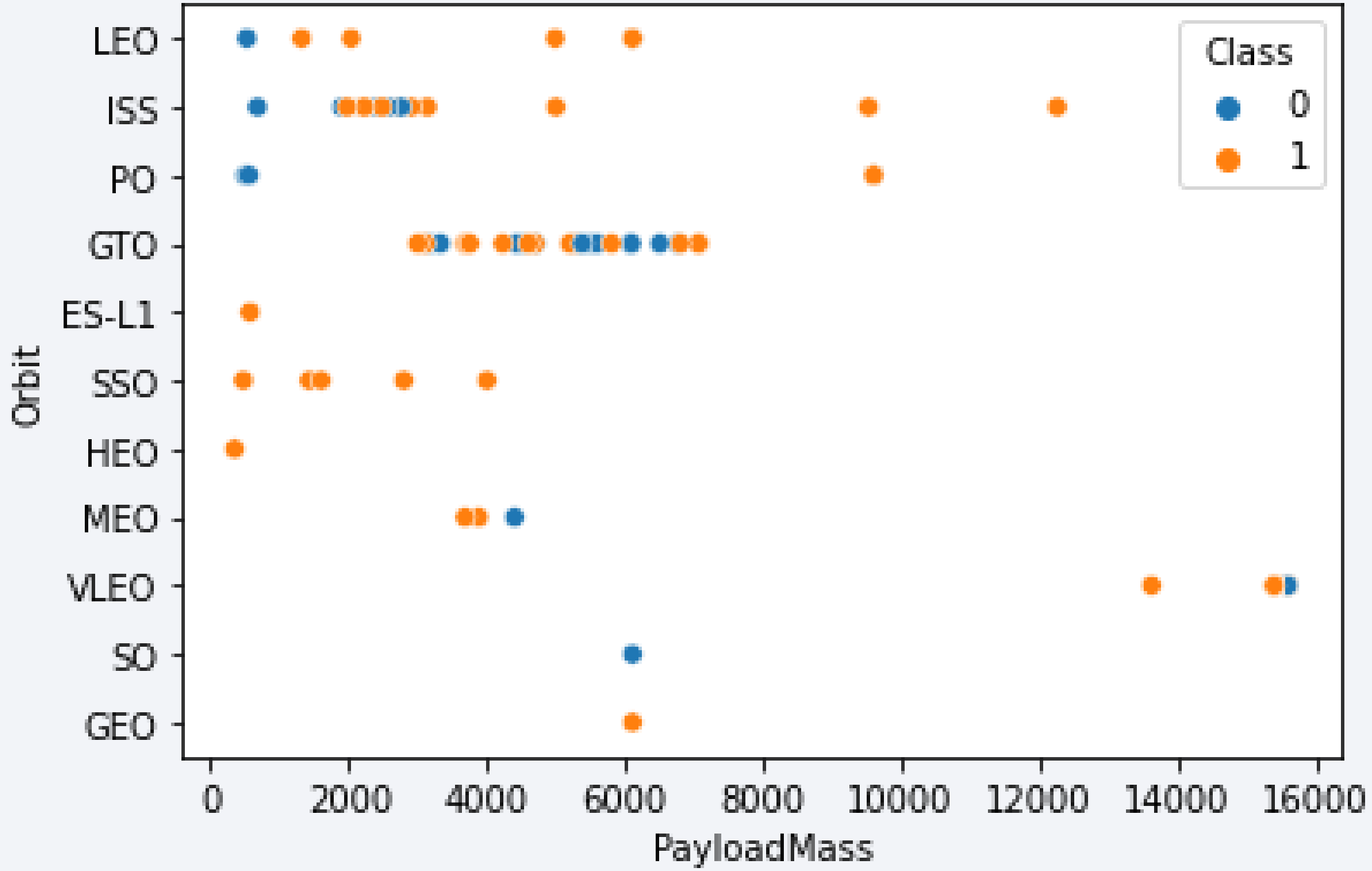
Flight Number vs. Orbit Type

- It is seen that the number of successful landings increases as the flight numbers increase. At the same time, almost none of the first 50 flight number launches took place in orbits of the Orbit type (ES-L1, SSO, HEO, MEO, VLEO, SO, GEO).



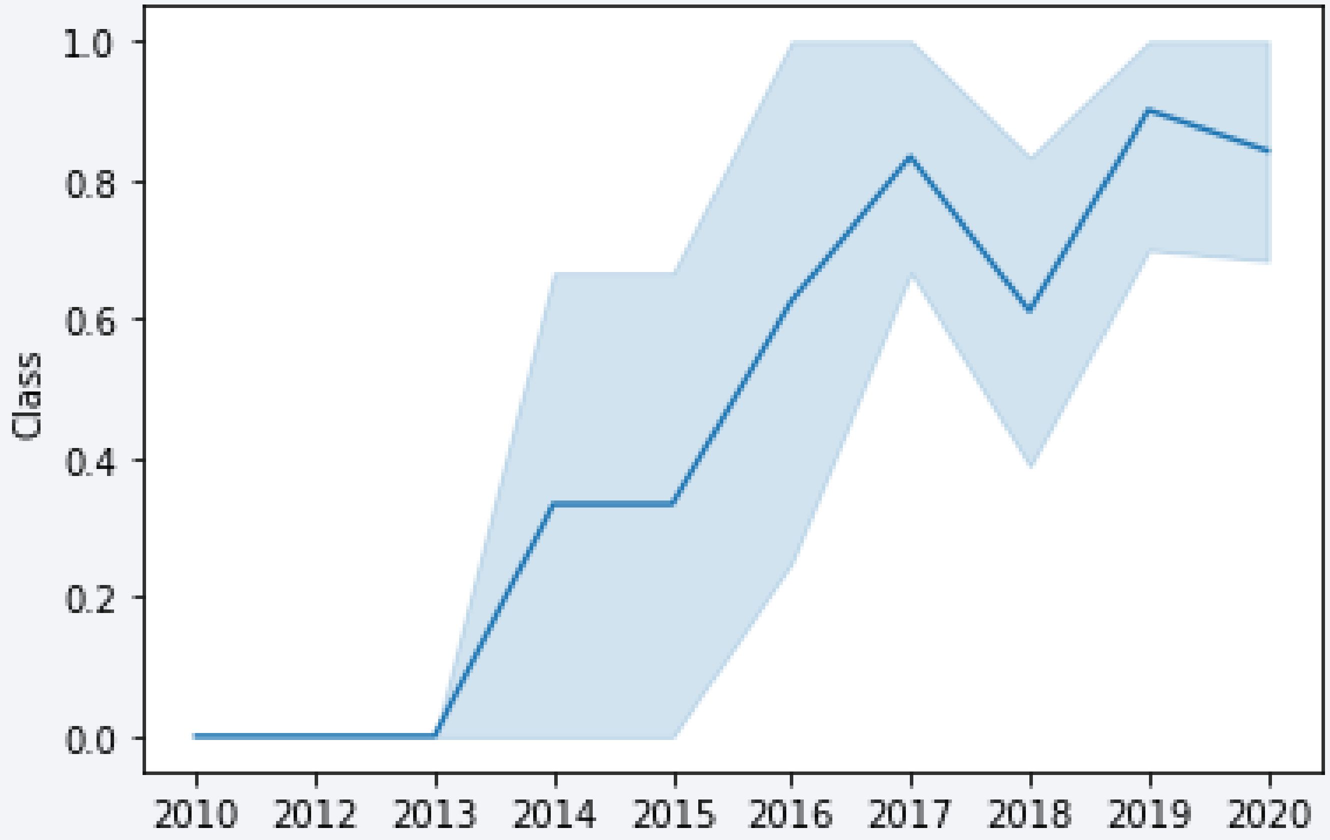
Payload vs. Orbit Type

- It is seen that as the Payload Mass increases, the launches decrease.
- At the same time, all launches of the payload mass between 4000 and 8000 were made at the point of the Orbit-type GTO.



Launch Success Yearly Trend

- It is very clear that as the years increase, the success rates increase.
- And none of the rockets launched in the first three years seem to have made their first successful landing.



All Launch Site Names

- It can be seen that there are four initiation points.

```
1  %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'

- It seems that launch sites have only five starting points with CCA.

```
[ ]  1  %sql SELECT * FROM SPACEXTBL WHERE "Launch_Site" LIKE '%CCA%' LIMIT 5
```

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

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

Total Payload Mass

- The total mass of payload carried by boosters launched by NASA (CRS) appears to be 45596 kg.

```
%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 the booster version F9 v1.1 appears to be 2534.6666666666665 kg.

```
[ ]  1  %sql SELECT AVG("PAYLOAD_MASS__KG_") FROM SPACEXTBL WHERE "Booster_Version" LIKE '%F9 v1.1%'

 * sqlite:///my_data1.db
Done.
AVG("PAYLOAD_MASS__KG_")
2534.6666666666665
```

First Successful Ground Landing Date

- We see that the first successful landing was on the first day of May 2017.

```
[ ]   1   %sql SELECT MIN("DATE") FROM SPACEXTBL WHERE "Landing _Outcome" LIKE '%Success%'

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

MIN("DATE")
01-05-2017

Successful Drone Ship Landing with Payload between 4000 and 6000

- Rockets that landed successfully on your drone ship:
- F9 FT B1022,
- F9 FT B1026,
- F9 FT B1021.2,
- F9 FT B1031.2

```
%sql SELECT "BOOSTER_VERSION" FROM SPACEXTBL WHERE "LANDING _OUTCOME" = 'Success (drone ship)' \
AND "PAYLOAD_MASS__KG_" > 4000 \
AND "PAYLOAD_MASS__KG_" < 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

- We see that the total successful landing is 100 and the total failed landing is 1.

```
%sql SELECT \
(SELECT COUNT("MISSION_OUTCOME") FROM SPACEXTBL WHERE "MISSION_OUTCOME" LIKE '%Success%') AS SUCCESS, \
(SELECT COUNT("MISSION_OUTCOME") FROM SPACEXTBL WHERE "MISSION_OUTCOME" LIKE '%Failure%') AS FAILURE
```

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

SUCCESS	FAILURE
100	1

Boosters Carried Maximum Payload

- Below we see the names of the launchers carrying the maximum load mass.

```
[ ]  1  %sql SELECT DISTINCT "BOOSTER_VERSION" FROM SPACEXTBL \
     2  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

- In 2015, we see the results of the rockets that landed unsuccessfully.

```
%sql SELECT substr("DATE", 4, 2) AS MONTH, "BOOSTER_VERSION", "LAUNCH_SITE" \
FROM SPACEXTBL\
WHERE "LANDING _OUTCOME" = 'Failure (drone ship)' and substr("DATE",7,4) = '2015'
```

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

MONTH	Booster_Version	Launch_Site
01	F9 v1.1 B1012	CCAFS LC-40
04	F9 v1.1 B1015	CCAFS LC-40

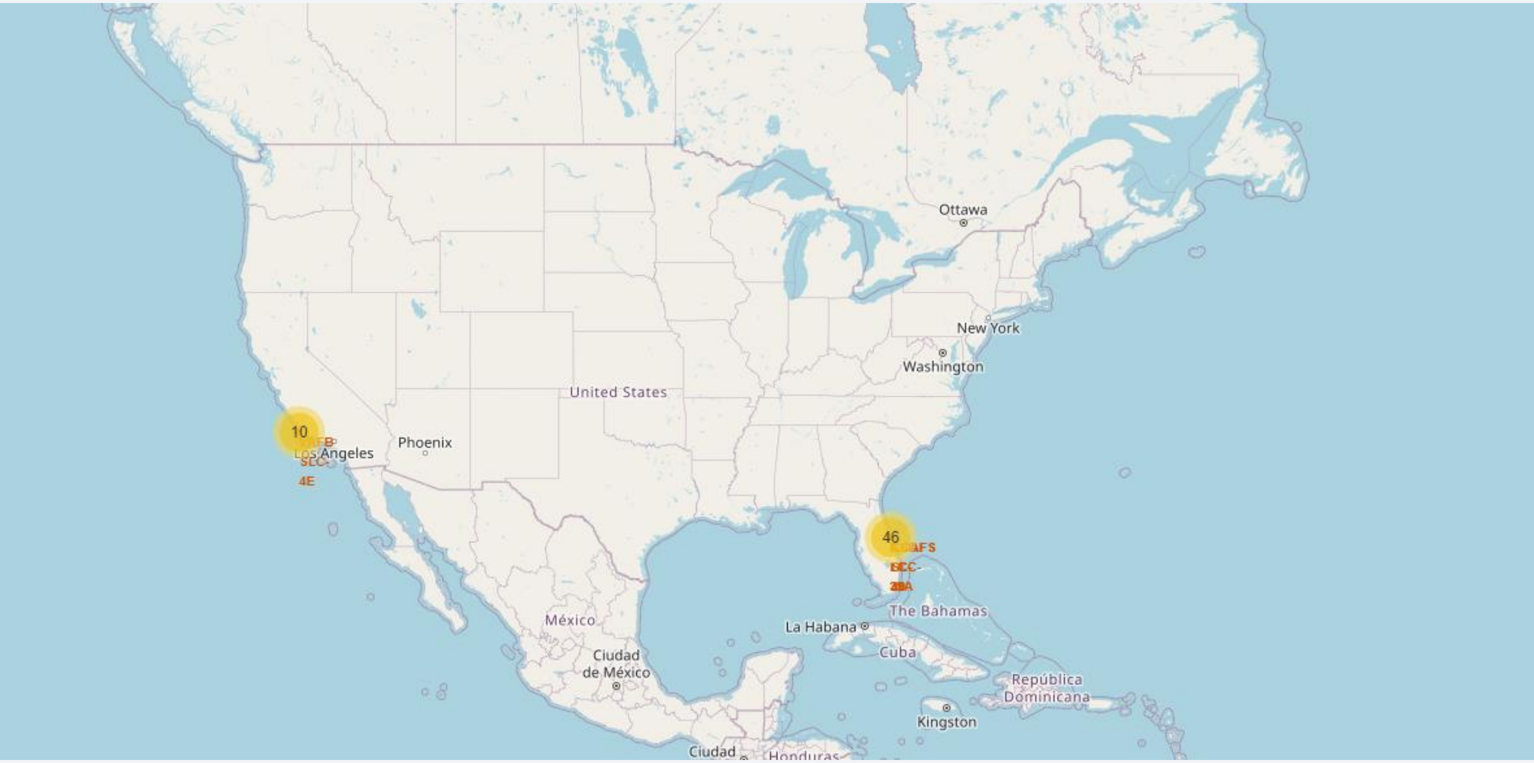
Rank Landing Outcomes Between 2010-06-04 and 2017-03-20

Section 3

Launch Sites Proximities Analysis

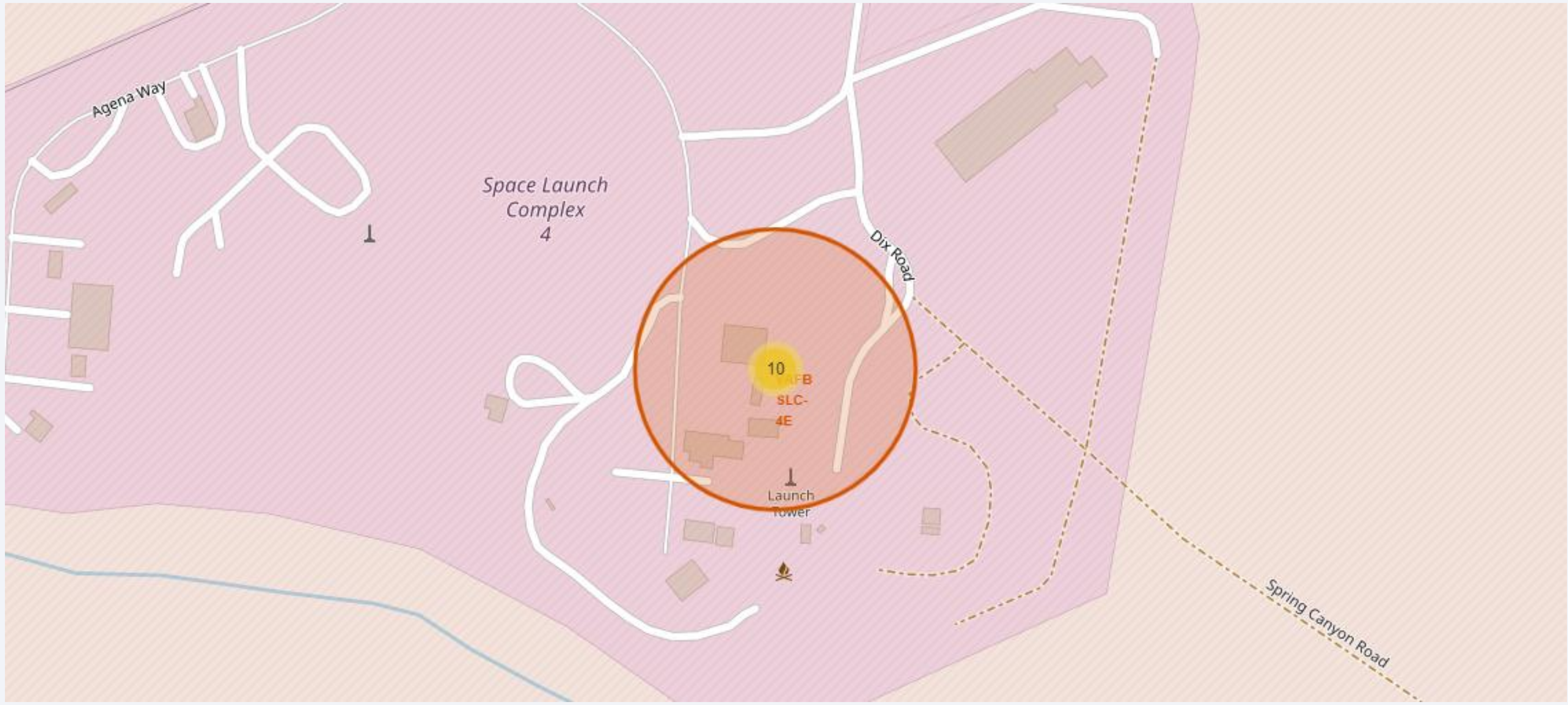
Site Map

- The map is as follows when a marker is added at each launch point in the dataset.



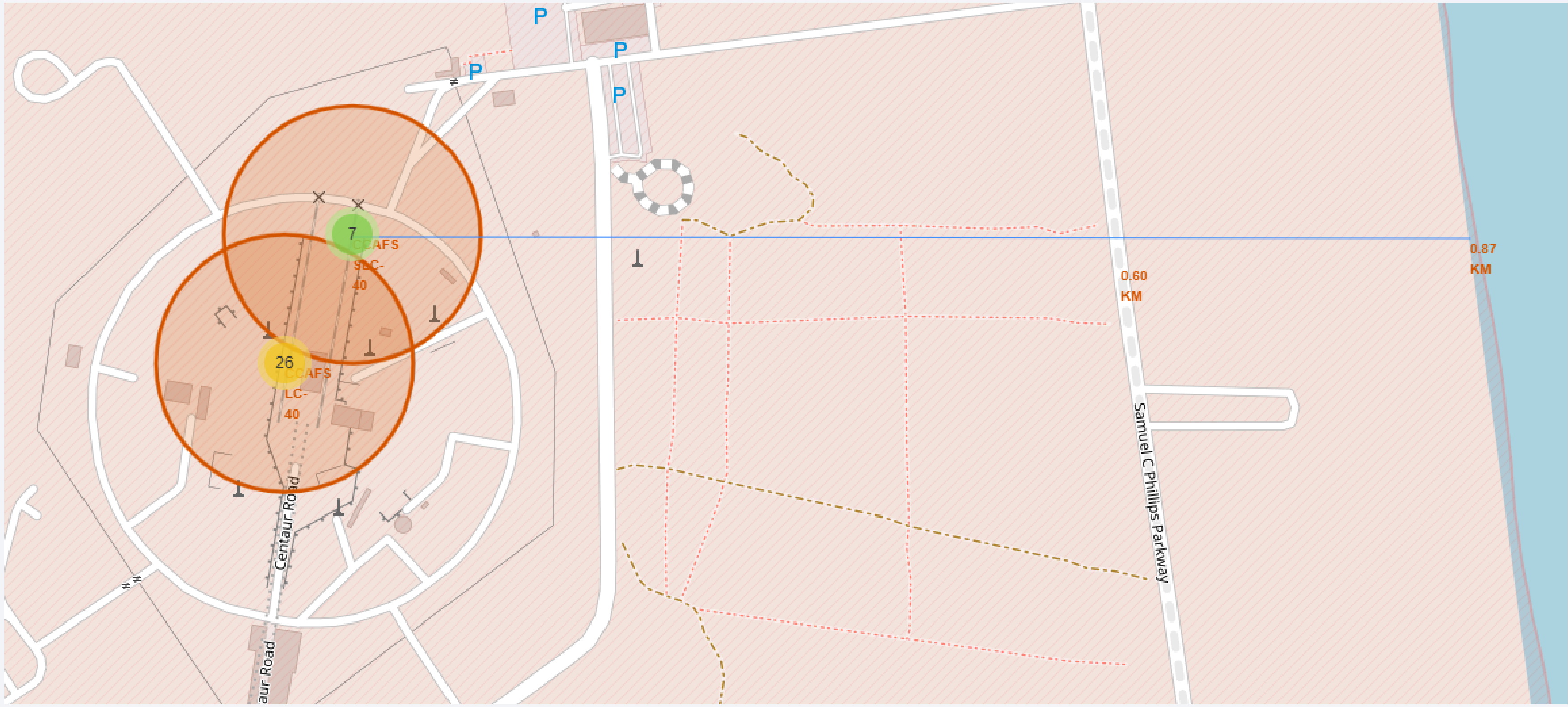
Color Labels

- We see that it is labeled with colored labels.



Adding Distances to the Map

- We see the distance of the launch point to the nearest shore.

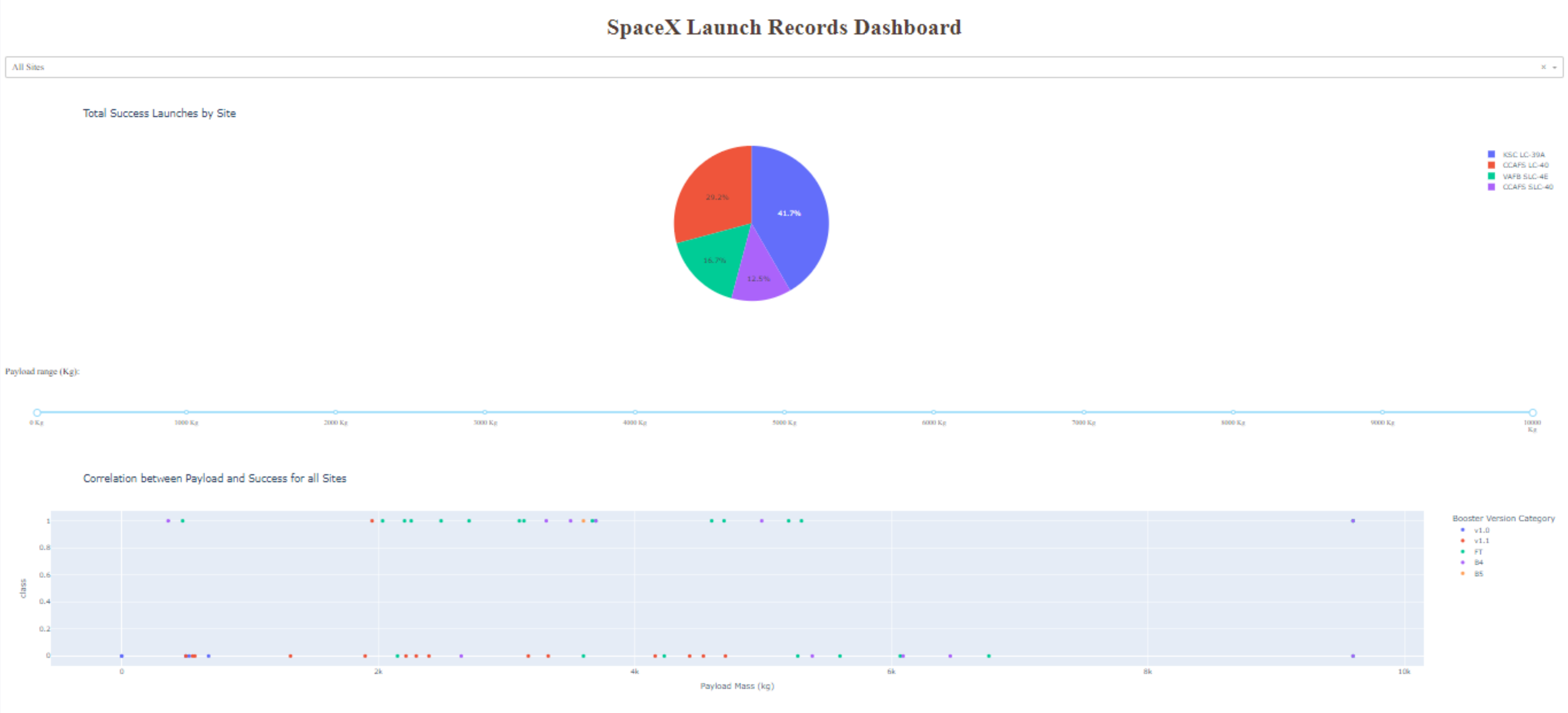


Section 4

Build a Dashboard with Plotly Dash

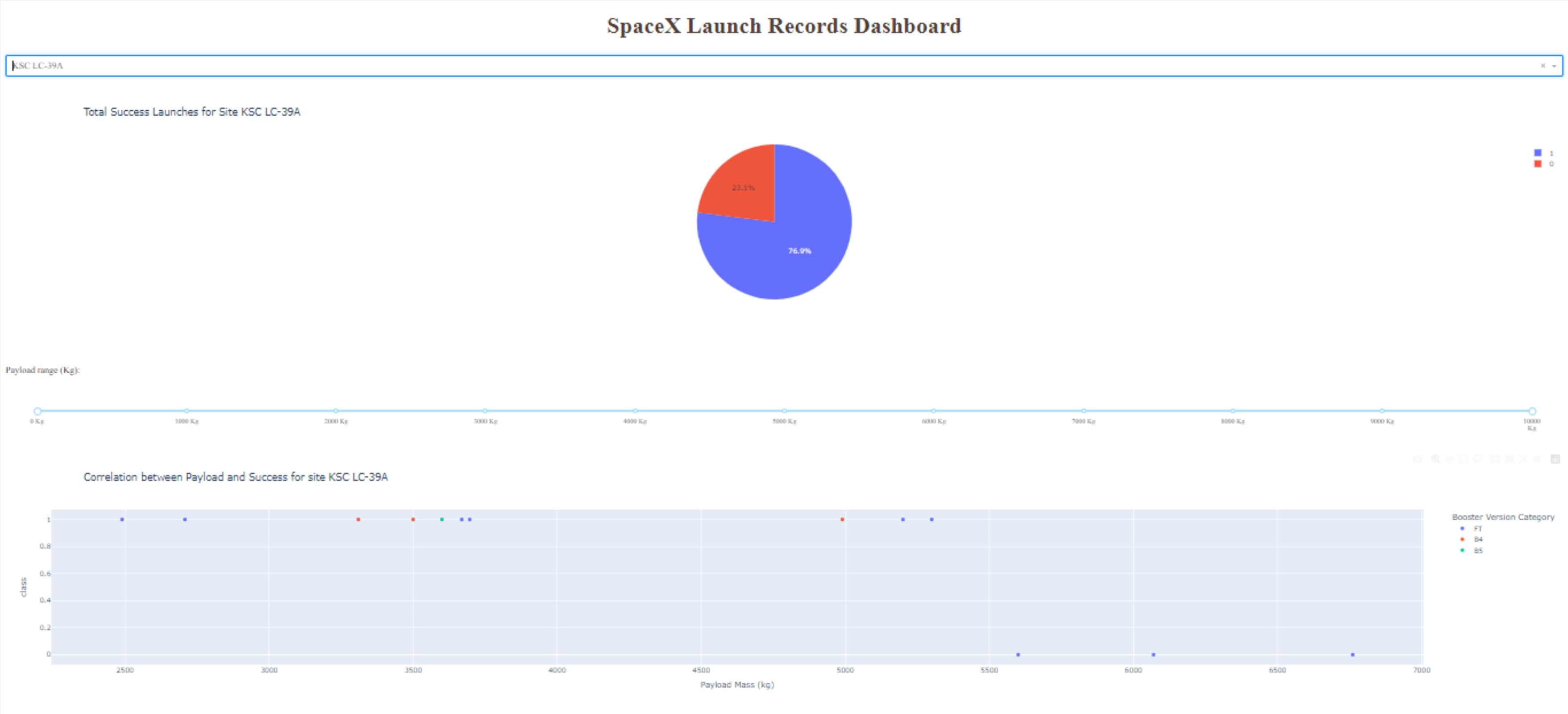
Tüm Siteler Panosu

- fdg



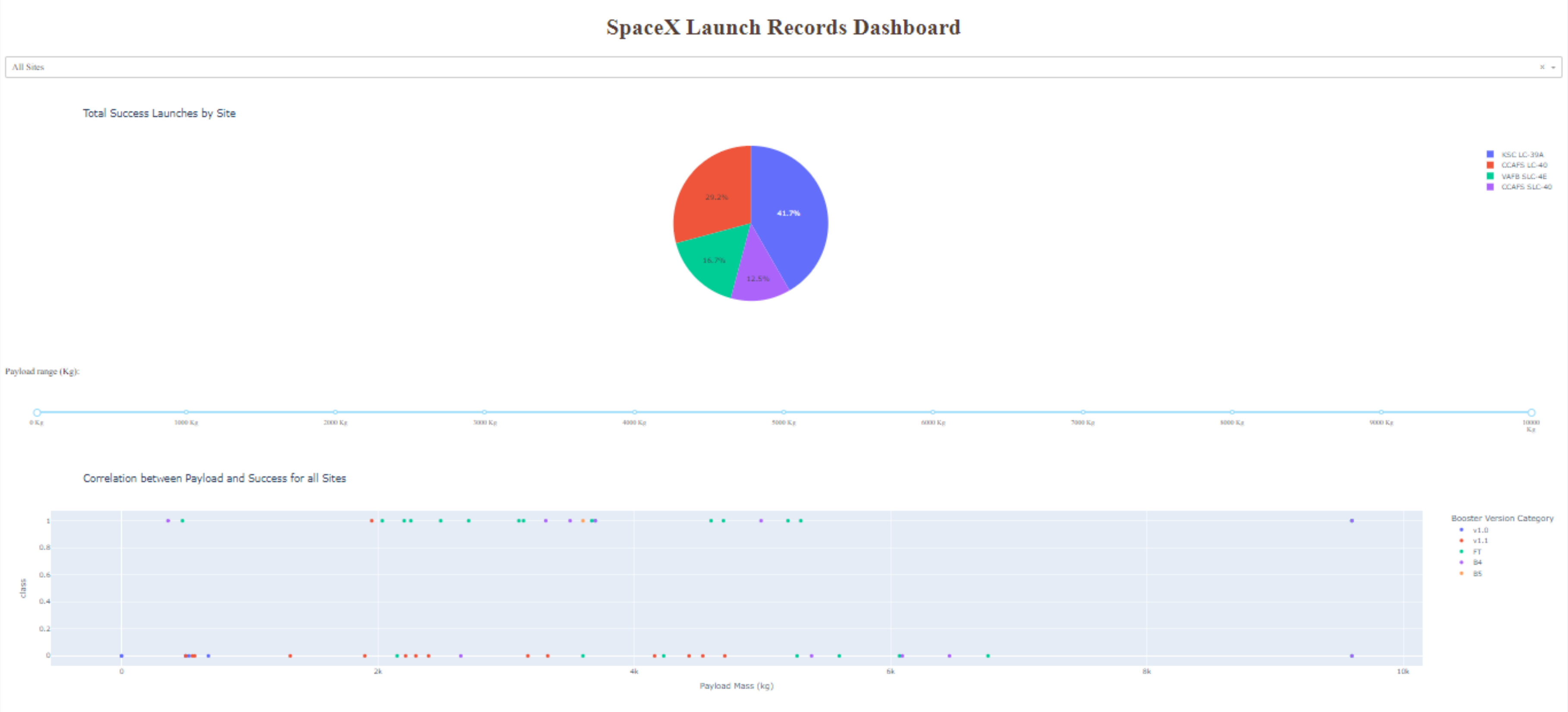
Highest Launch Success Rate

- In the Dashboard below, we see the point with the highest launch point.



All Cities Distribution Chart

- We see the scatterplot graph at different values.

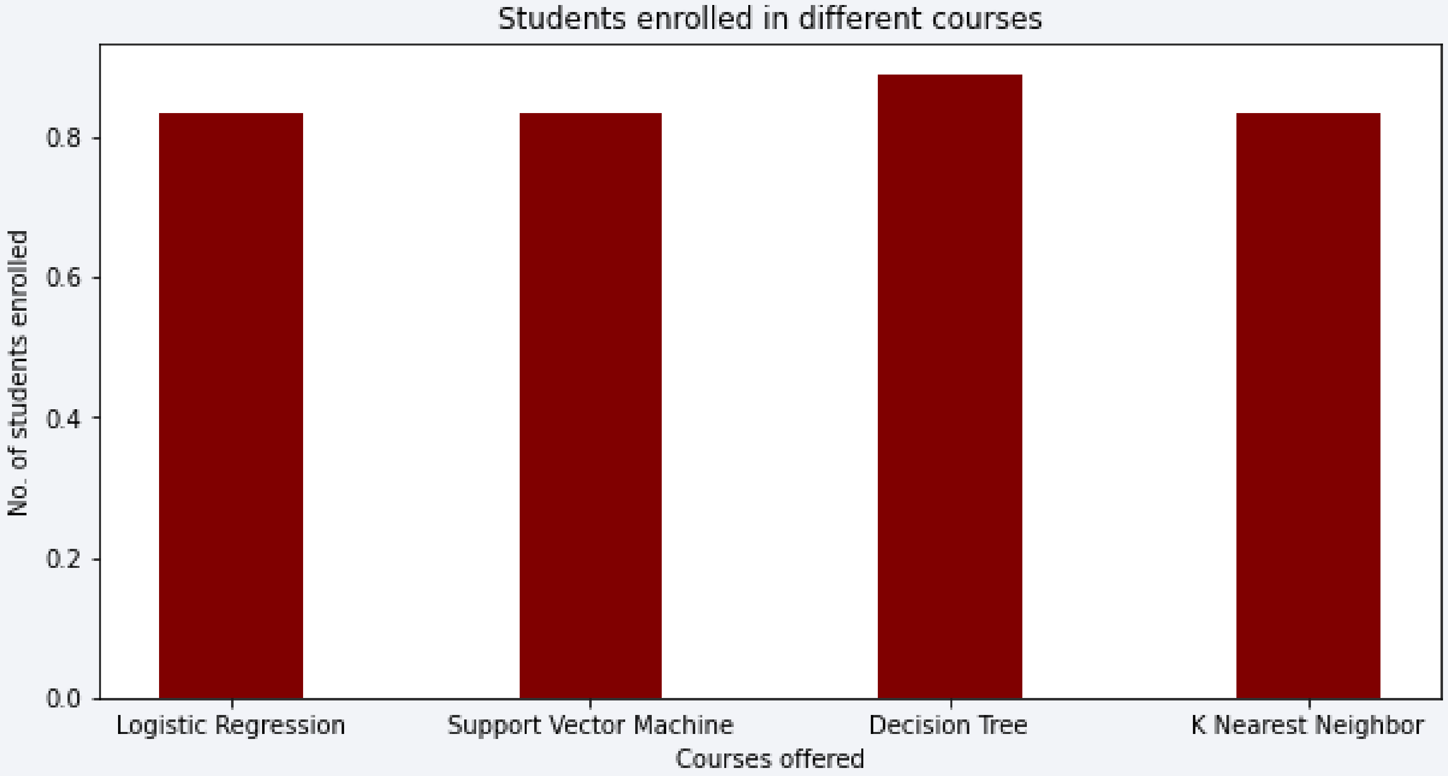


Section 5

Predictive Analysis (Classification)

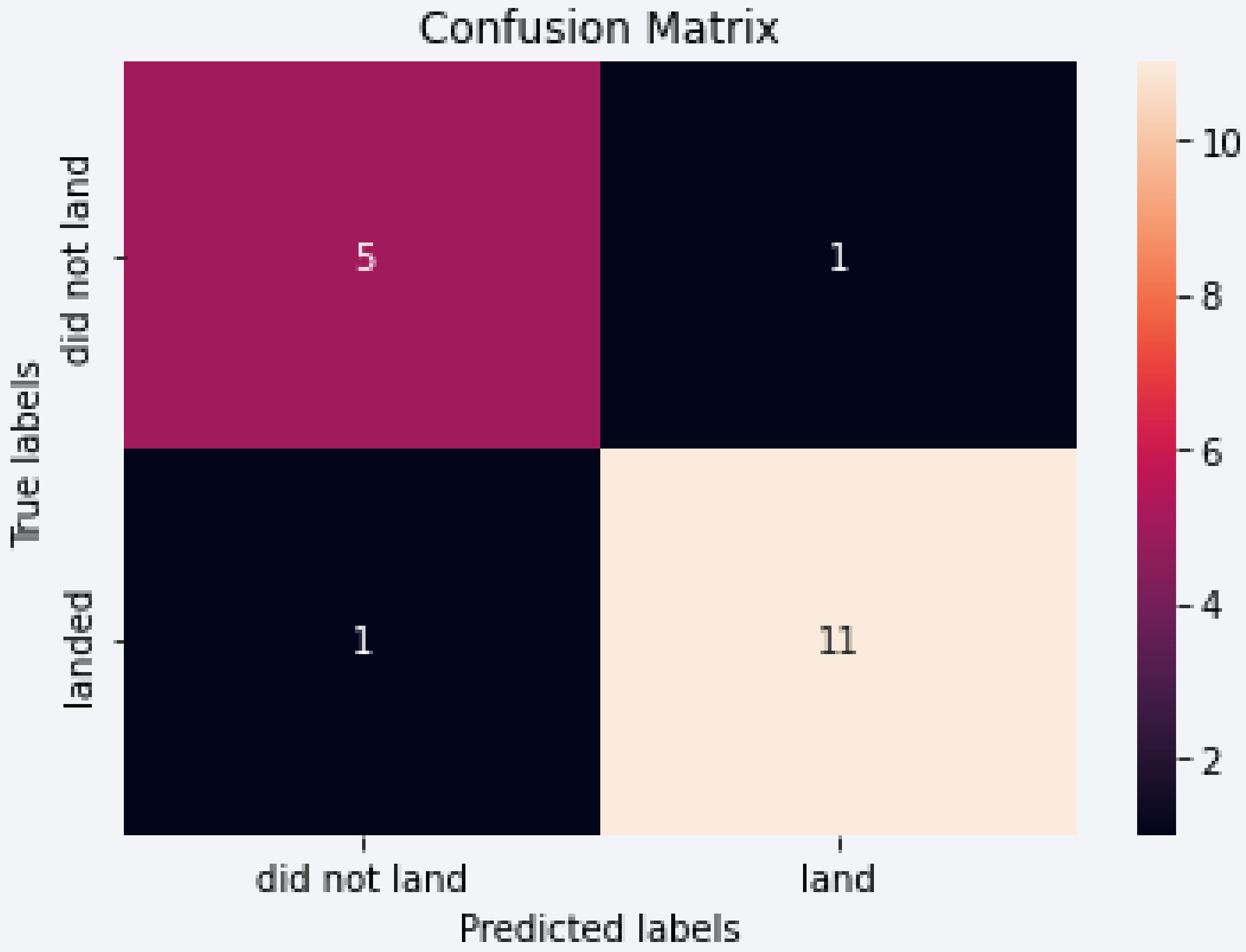
Classification Accuracy

- Best Model: Decision Tree



Confusion Matrix

- Best Model: Decision Tree
- Decision Tree Confusion Matrix

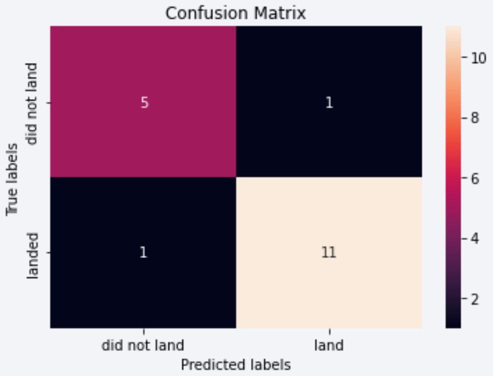


Conclusions

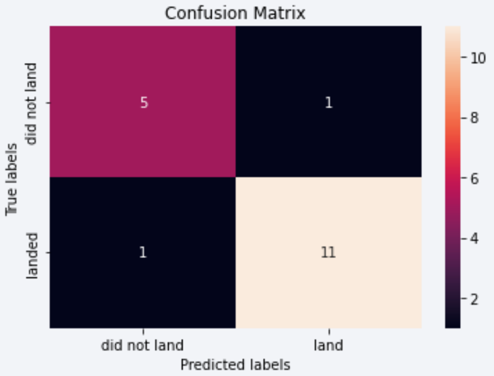
Logistic Regression

Support Vector Machine

Decision Tree



KNN



Appendix

```
model_name = ["Logistic Regression",
              "Support Vector Machine",
              "Decision Tree",
              "K Nearest Neighbor"]


ho_as = [logreg_ho, svm_ho, tree_ho, knn_ho]

model = pd.DataFrame({"Model" : model_name,
                      "Accuracy Score (Hyperparameter Optimization)": ho_as})
model
```

```
fig = plt.figure(figsize = (10, 5))

columns = model["Model"]
values = model["Accuracy Score (Hyperparameter Optimization)"]
# creating the bar plot
plt.bar(columns, values, color ='maroon',
        width = 0.4)

plt.show()
```

Thank you!