IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Nurullah Kuş
06/07/2024

Outline

Executive Summary

- Summary of methodologies
  - Data Collection through API
  - Data Collection with Web Scraping
  - Data Wrangling
  - Exploratory Data Analysis with SQL
  - Exploratory Data Analysis with Data Visualization
  - Interactive Visual Analytics with Folium
  - Machine Learning Prediction
- Summary of all results
  - Exploratory Data Analysis
  - Interactive Dashboard
  - Predictive Analytics Evaluation

Introduction

- **Project background and context**

SpaceX advertises Falcon 9 rocket launches on its website with a cost of 62 million dollars;

other providers cost upwards 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. Using that info we can try to provide an insight using machine learning.

- **Problems you want to find answers**
- Success factors to determine the rocket landing
- What are the relations between the parameters that affects the result of landing?

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Used SpaceX API and scraped wikipedia data
- Perform data wrangling
  - One-hot encoding was applied to categorical features
- 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

- Data is collected accessing the SpaceX API.
- Data is decoded using .json() function and turned it into a pandas dataframe using .json_normalize().
- After that data cleaning process is done. Here Data is checked for missing values. Some missing values are replaced with NaN.
- In Wikipedia source, web scraping is performed for Falcon 9 launch records with BeautifulSoup.
- Extracted data is parsed as HTML table and converted to a pandas dataframe for future analysis.

Data Collection – SpaceX API

- Data was collected using a GET request to the SpaceX API, the requested data was cleaned, and basic data wrangling and formatting were performed

- https://github.com/nurullah44/IBM_Capstone/blob/main/jupyter-labs-spacex-data-collection-api.ipynb

```
In [8]:  spacex_url="https://api.spacexdata.com/v4/launches/past"

In [9]:  response = requests.get(spacex_url)

In [14]: # Use json_normalize meethod to convert the json result into a dataframe
         data = pd.json_normalize(response.json())
```

```
# Calculate the mean value of PayloadMass column
mean_value = data_falcon9["PayloadMass"].mean()

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

Data Collection - Scraping

- Web scraping was applied to extract Falcon 9 launch records using BeautifulSoup. The table was parsed and converted into a pandas DataFrame

- https://github.com/nurullah44/IBM_Capstone/blob/main/jupyter-labs-webscraping.ipynb

```
# use requests.get() method with the provided static_url
response = requests.get(static_url)
# assign the response to a object
print(response.status_code)
```

```
200
```

```
# Use BeautifulSoup() to create a BeautifulSoup object from a response text content
soup = BeautifulSoup(response.text, 'html.parser')
```

```
column_names = []
html_th = first_launch_table.find_all('th')

for th in html_th :
    if extract_column_from_header(th) is not None and len(extract_column_from_header(th)) > 0 :
        column_names.append(extract_column_from_header(th))
```

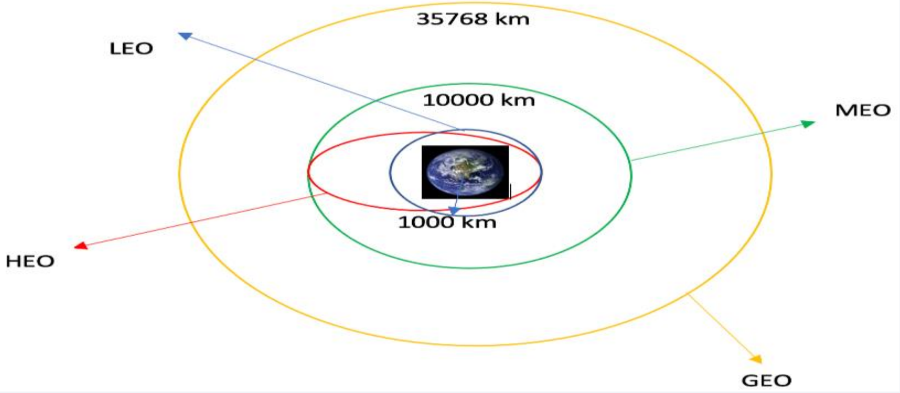
Data Wrangling

- Exploratory data analysis was performed to determine the training labels. The number of launches at each site and the number and occurrence of each orbit were calculated. A landing outcome label was created from the outcome column, and the results were exported to CSV.

- https://github.com/nurullah44/IBM_Capstone/blob/main/labs-jupyter-spacex-Data%20wrangling.ipynb

```
# Apply value_counts() on column LaunchSite
df["LaunchSite"].value_counts()
```

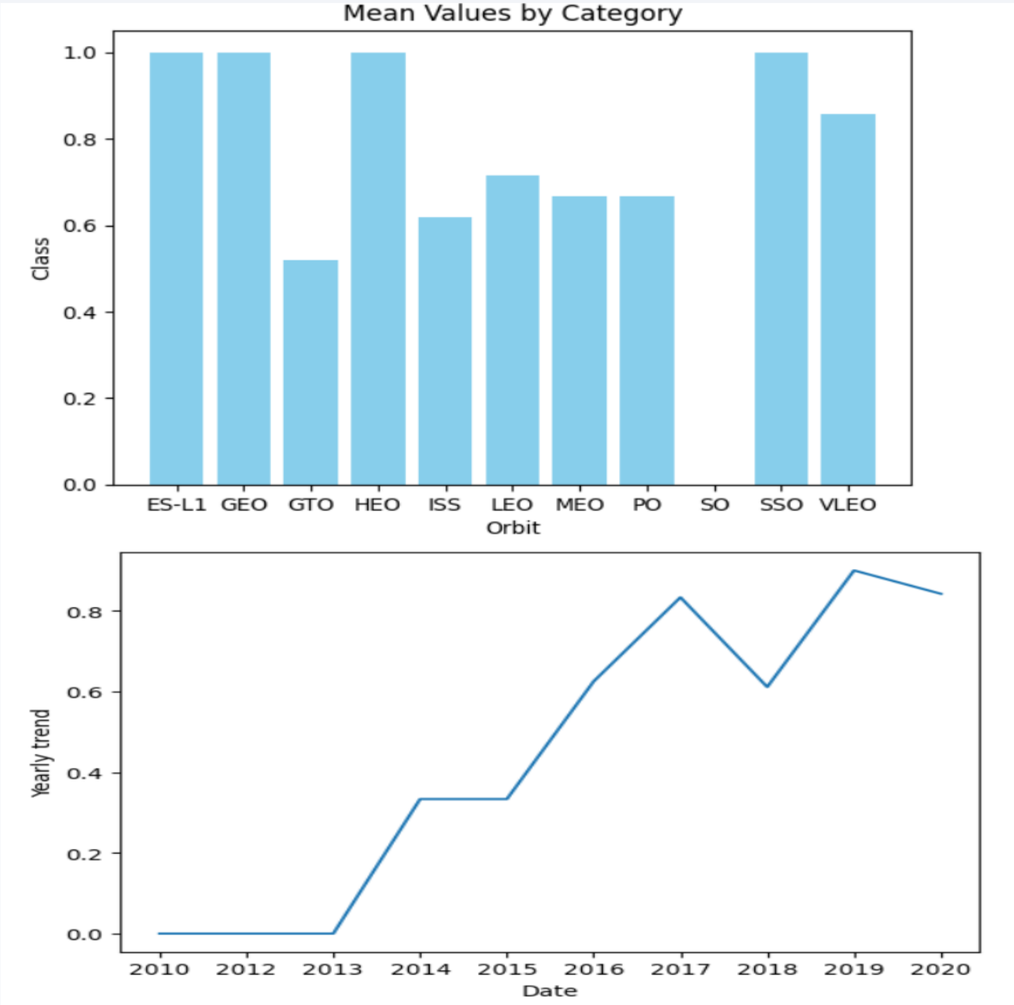
```
CCAFS SLC 40    55
KSC LC 39A      22
VAFB SLC 4E     13
Name: LaunchSite, dtype: int64
```




```
# landing_class = 0 if bad_outcome
# landing_class = 1 otherwise
landing_class = []
for outcome in df["Outcome"]:
    if outcome in bad_outcomes:
        landing_class.append(0)
    else :
        landing_class.append(1)
```

EDA with Data Visualization

- The data was explored by visualizing the relationship between flight number and launch site, payload and launch site, success rate of each orbit type, flight number and orbit type, and the yearly trend of launch success.
- https://github.com/nurullah44/IBM_Capstone/blob/main/edadatavisualization-spacex.ipynb



EDA with SQL

- The names of unique launch sites in the space mission.
- The total payload mass carried by boosters launched by NASA (CRS)
- The average payload mass carried by booster version F9 v1.1
- The total number of successful and failure mission outcomes
- date when the first successful landing outcome in ground pad was achieved
- The failed landing outcomes in drone ship, their booster version and launch site names.

- https://github.com/nurullah44/IBM_Capstone/blob/main/jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

- All launch sites were marked, and map objects such as markers, circles, and lines were added to indicate the success or failure of launches for each site on the folium map.
- The launch outcomes were assigned to classes 0 and 1 to see the successful landings and failure ones
- Using color-labeled marker clusters, launch sites with relatively high success rates were identified
- The distance was calculated to the proximities to understand the geographic location of the launch sites.

- https://github.com/nurullah44/IBM_Capstone/blob/main/lab_jupyter_launch_site_location.ipynb

Build a Dashboard with Plotly Dash

- An interactive dashboard was built using Plotly Dash
- Pie charts were plotted to show the total launches by certain sites.
- We plotted scatter graph showing the relationship with Outcome and Payload Mass (Kg) for the different booster version
- The goal was to find the successful landings using payload mass and looking the overall success rate.
- https://github.com/nurullah44/IBM_Capstone/blob/main/spacex_dash_app.py

Predictive Analysis (Classification)

- The data was loaded using NumPy and pandas, transformed, and split into training and testing sets
- Different machine learning models were built and hyperparameters were tuned using GridSearchCV
- Accuracy was used as the metric for evaluating the models. The model was improved through feature engineering and algorithm tuning

```
31]:
models = { "KNN":knn_cv.best_score_ ,
           "DecisionTree":tree_cv.best_score_,
           "SVM" : svm_cv.best_score_,
           "LogReg" : logreg_cv.best_score_ }
best_model = max(models, key = models.get)
best_score = models[best_model]
print(f'The best model is: {best_model} with score of {best_score} ')
#print("accuracy :",knn_cv.best_score_)
#print("accuracy :",tree_cv.best_score_)
#print("accuracy :",svm_cv.best_score_)
#print("accuracy :",logreg_cv.best_score_)

The best model is: DecisionTree with score of 0.8732142857142856
```

- https://github.com/nurullah44/IBM_Capstone/blob/main/SpaceX_Machine%20Learning%20Prediction.ipynb

Results

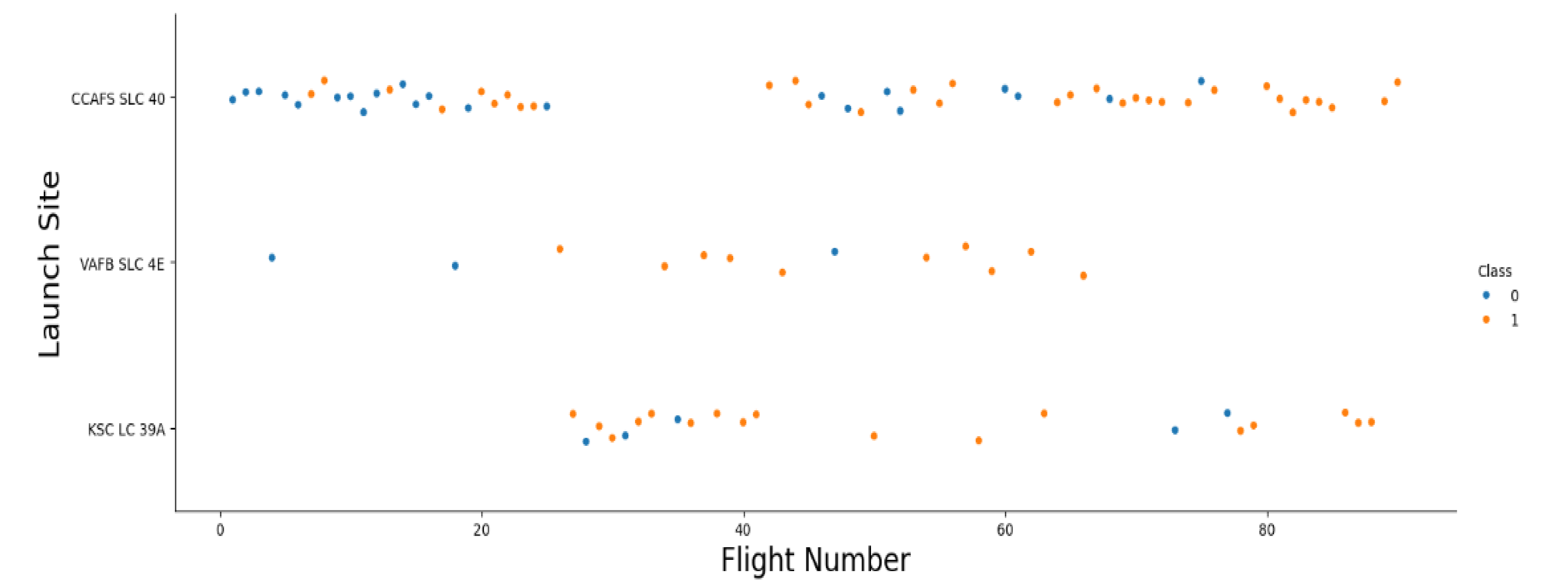
- Exploratory data analysis results
- Interactive analytics demo in screenshots
- Predictive analysis results

Section 2

Insights drawn from EDA

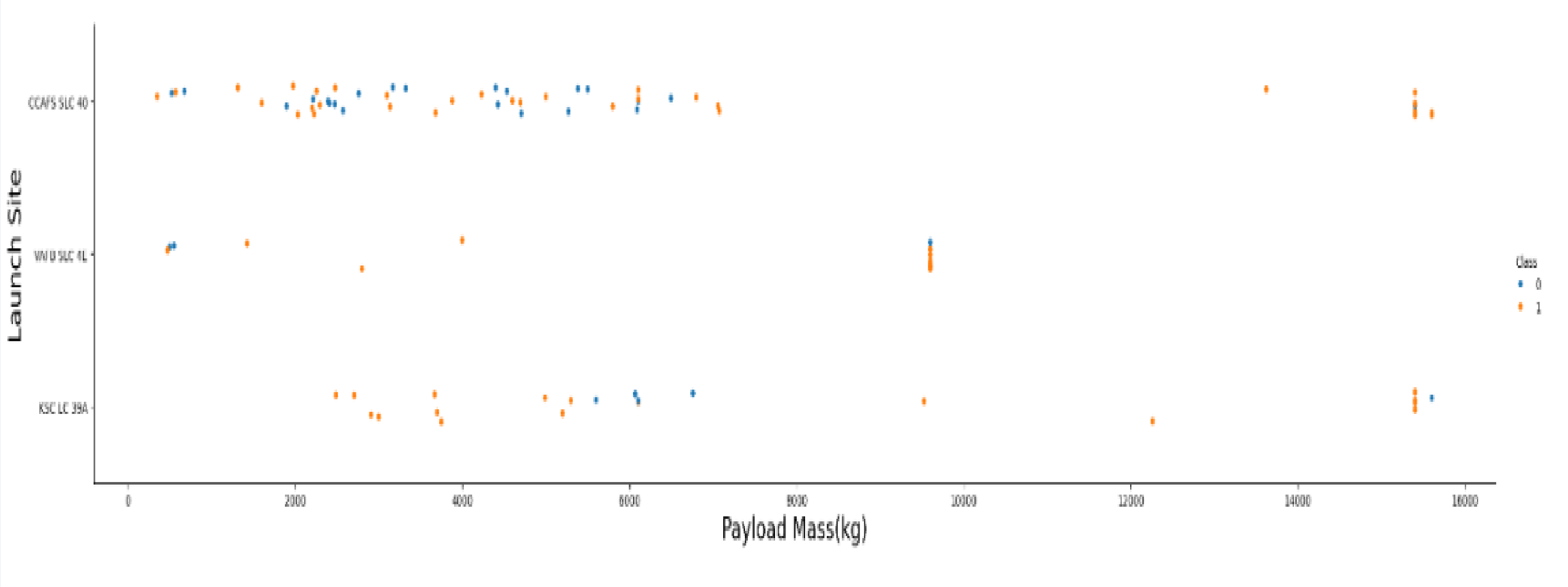
Flight Number vs. Launch Site

- The plot revealed that an increase in the number of flights at a launch site corresponds to a higher success rate at that location



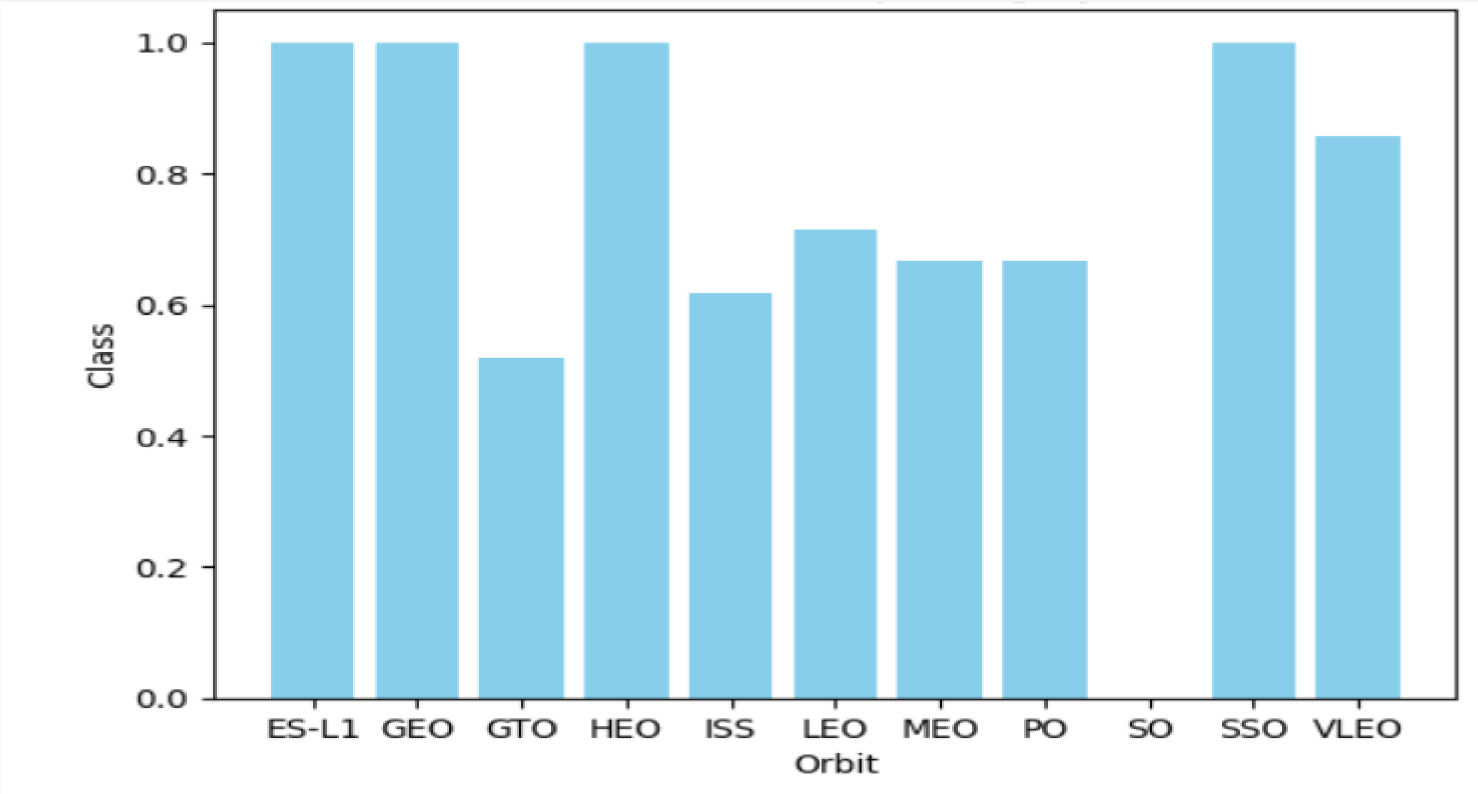
Payload vs. Launch Site

- It is mostly distrubuted low weighted .
- For CCAFS SLC 40 heavier the payload higher the success rate for the rocket
- There is no heavy weighted landing for VAFB-SLC . Max is 10000 kg



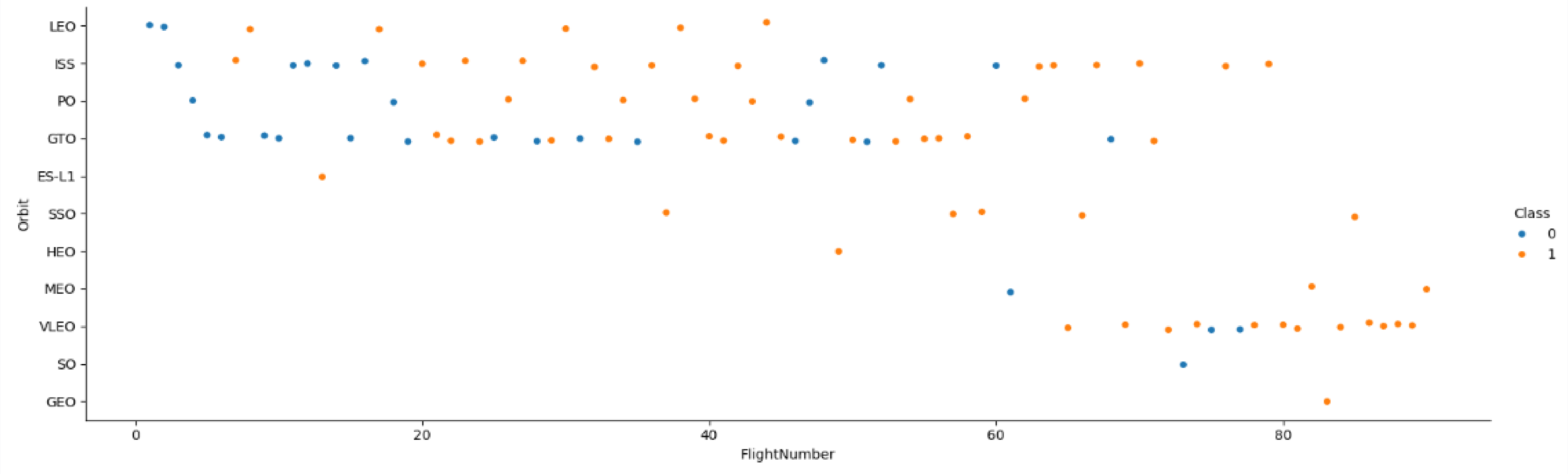
Success Rate vs. Orbit Type

- From the bar chart, it is seen that ES-L1, GEO, HEO, SSO have the most success rate.



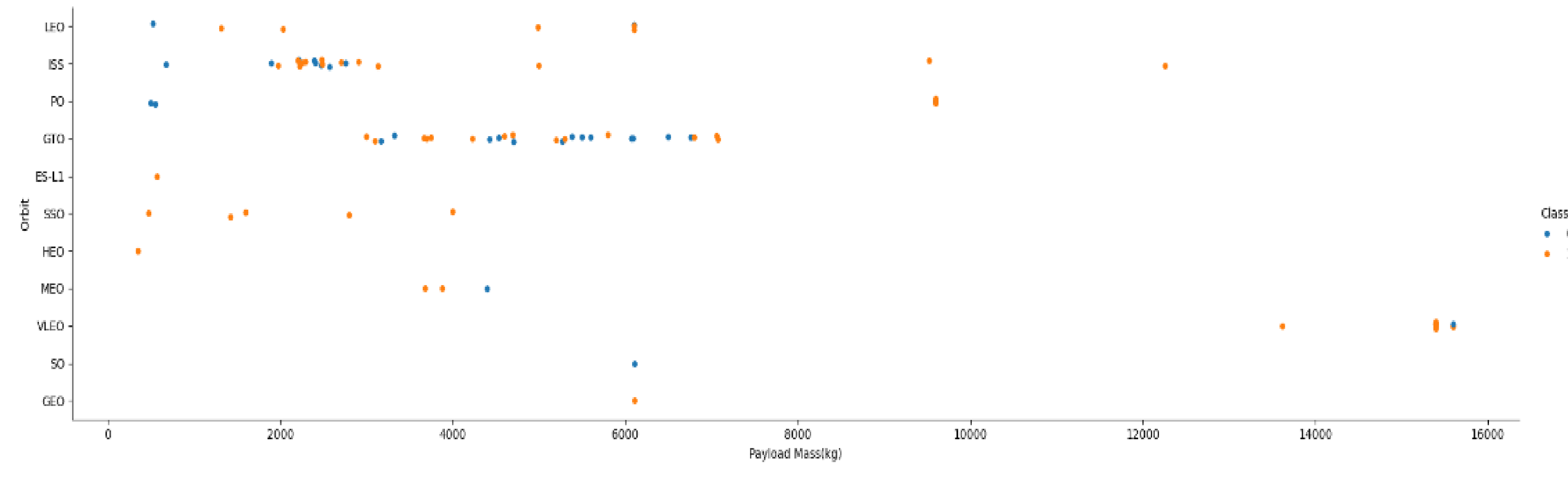
Flight Number vs. Orbit Type

- Below we can see a positive correlation between the flight number and success rate for LEO orbit. For others, it is not clearly correlated.



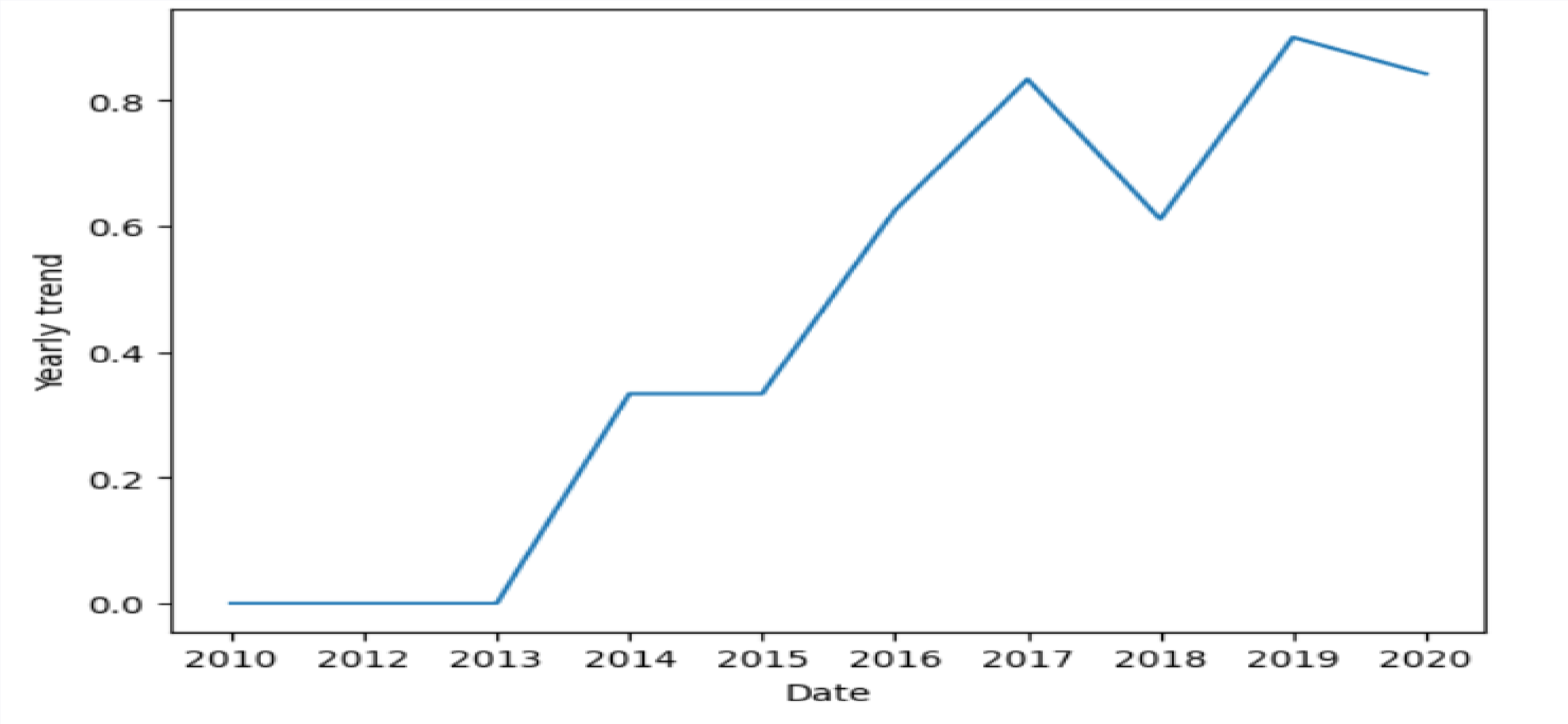
Payload vs. Orbit Type

- According to this plot we can see the increase in the success for LEO ,ISS and PO



Launch Success Yearly Trend

There is an increase in success rate between 2013 and 2020



All Launch Site Names

- it is obtained using Distinct Keyword in sql

In [15]:
```
%sql select DISTINCT "Launch_Site" from SPACEXTABLE
```

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

Out[15]:

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

Launch Site Names Begin with 'CCA'

- The query to print out the Launch Site Names begin with CCA is below.

In [37]:
```
%sql select * from SPACEXTABLE  where "Launch_Site" LIKE 'CCA%' limit 5
```

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

Out[37]:

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	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
2012-10-08	0:35:00	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

- Total is 45596 KG when we use below query

```
In [29]: %sql select SUM("PAYLOAD_MASS__KG_") as total  from SPACEXTABLE where Customer like "NASA (CRS)"

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

Out[29]:

total
45596

Average Payload Mass by F9 v1.1

- Average query is 2279.8 KG

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

In [30]:
```
%sql select AVG("PAYLOAD_MASS__KG_") as total  from SPACEXTABLE where Customer like "NASA (CRS)"
```

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

Out[30]:

total
2279.8

First Successful Ground Landing Date

- First Successful Landing Outcome on ground pad was 22 nd December 2015

```
[8]: %sql select min(Date) as SuccessDate   from SPACEXTABLE where "Landing_Outcome" like "Success (ground pad)"

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

[8]:

SuccessDate
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

- Below Query is to see Booster versions that had been successful Drone Ship Landing between 4000 kg and 6000 kg. We see that it is 5 of

In [38]:
```
%sql select "Booster_Version" from SPACEXTABLE where "Landing_Outcome" LIKE "%drone ship%" and ("PAYLOAD_MASS__KG_" > 4000 a
```

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

Out[38]:

Booster_Version
F9 FT B1020
F9 FT B1022
F9 FT B1026
F9 FT B1021.2
F9 FT B1031.2

Total Number of Successful and Failure Mission Outcomes

- Here we can see the Mission outcome as success or failure. When we compare it to the landing outcome it is really high.

In [41]:
```
%sql select "Mission_Outcome", COUNT(*) as value_counts from SPACEXTABLE group by "Mission_Outcome"
```

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

Out[41]:

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

Boosters Carried Maximum Payload

- Here we listed the Booster versions that have the max payload as kg.

```
In [42]: %sql select "Booster_Version" from SPACEXTABLE where "PAYLOAD_MASS__KG_" = ( SELECT MAX("PAYLOAD_MASS__KG_") FROM SPACEXTABL
```

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

Out[42]:

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

- Here we see the quick summary of the 2015 records using specific parameters.

In [48]:
```
%sql select substr(Date, 6,2) as month ,"Landing_Outcome","Booster_Version","launch_site" from SPACEXTABLE where substr(Date
```

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

Out[48]:

month	Landing_Outcome	Booster_Version	Launch_Site
01	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
02	Controlled (ocean)	F9 v1.1 B1013	CCAFS LC-40
03	No attempt	F9 v1.1 B1014	CCAFS LC-40
04	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40
04	No attempt	F9 v1.1 B1016	CCAFS LC-40
06	Precluded (drone ship)	F9 v1.1 B1018	CCAFS LC-40
12	Success (ground pad)	F9 FT B1019	CCAFS LC-40

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

- We see the descended order of the rank of the Landing outcome between 2010 and 2017.

[11]:
```
%sql select "Landing_Outcome",count(*) as rank from SPACEXTABLE  WHERE Date BETWEEN '2010-06-04' AND '2017-03-20' GROUP BY "Landing_Outcome" o
```

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

[11]:

Landing_Outcome	rank
No attempt	10
Success (drone ship)	5
Failure (drone ship)	5
Success (ground pad)	3
Controlled (ocean)	3
Uncontrolled (ocean)	2
Failure (parachute)	2
Precluded (drone ship)	1

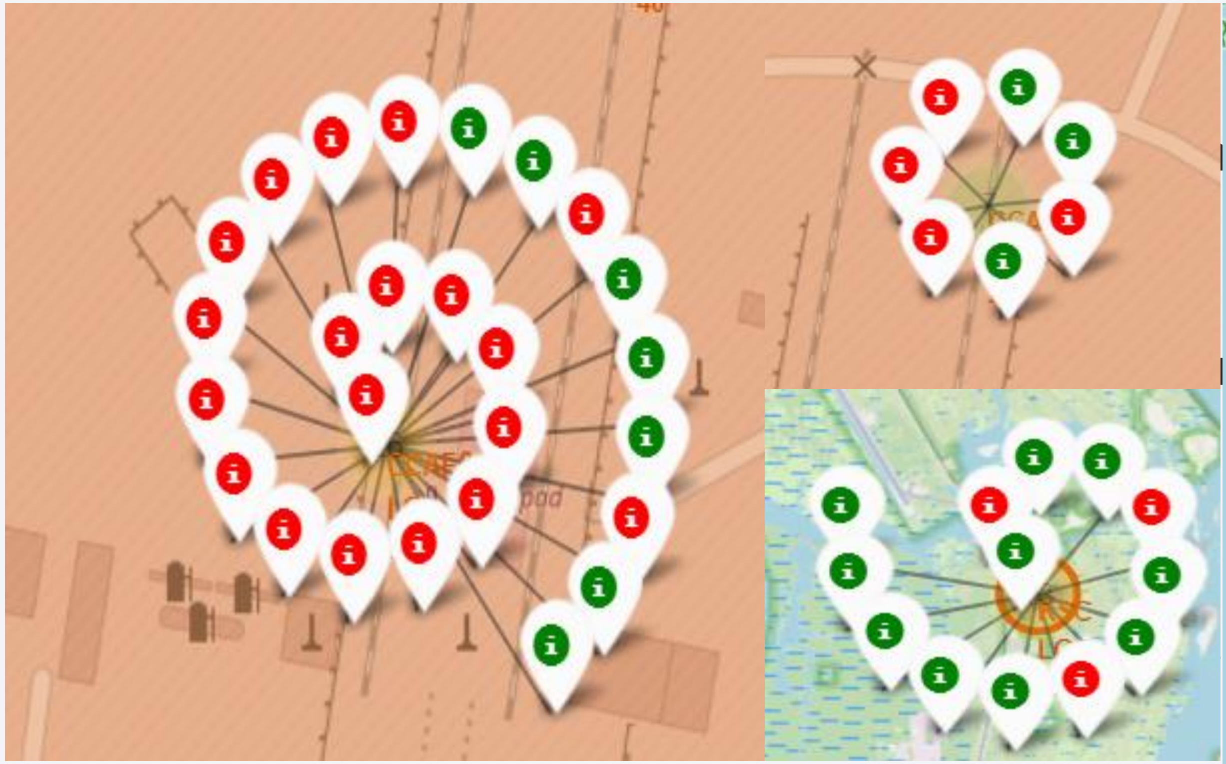
Section 3

Launch Sites Proximities Analysis

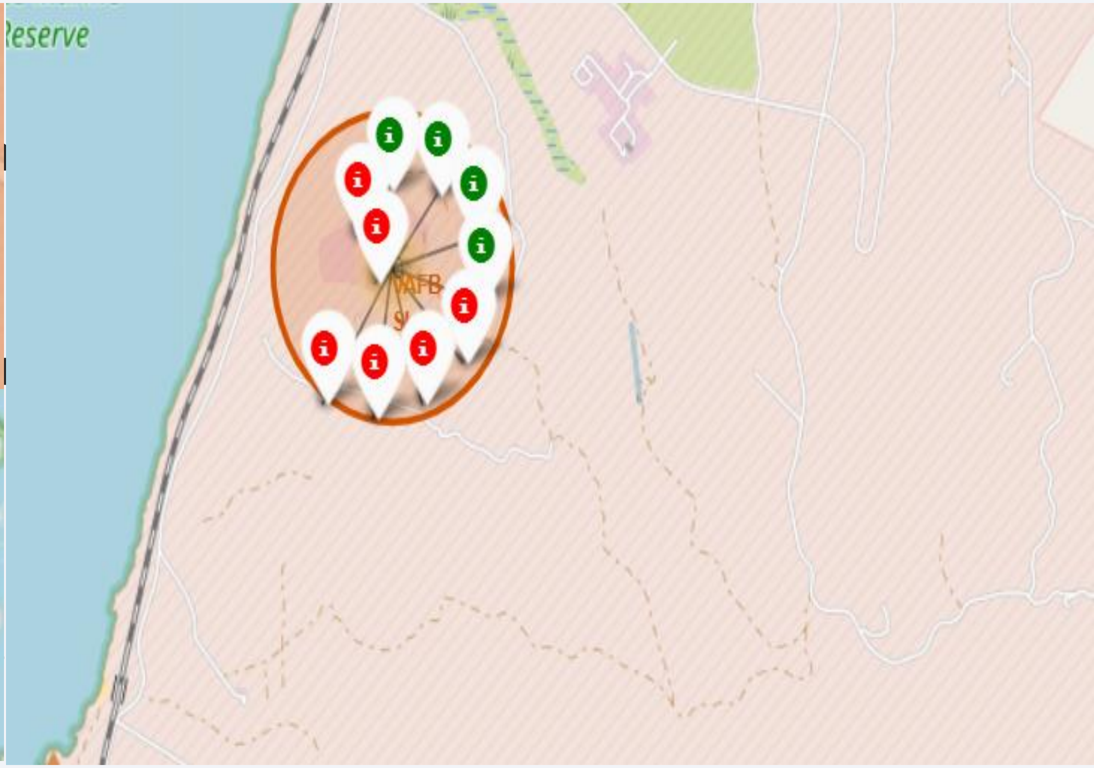
Launch sites in a Global Mab

WE SEE THAT IT IS ONLY FROM USA CLOSE TO THE EQUATOR LINE

Landing outcome of the Launch Sites By Location



FLORIDA



CALIFORNIA

Green markers are success
and Red markers are failure

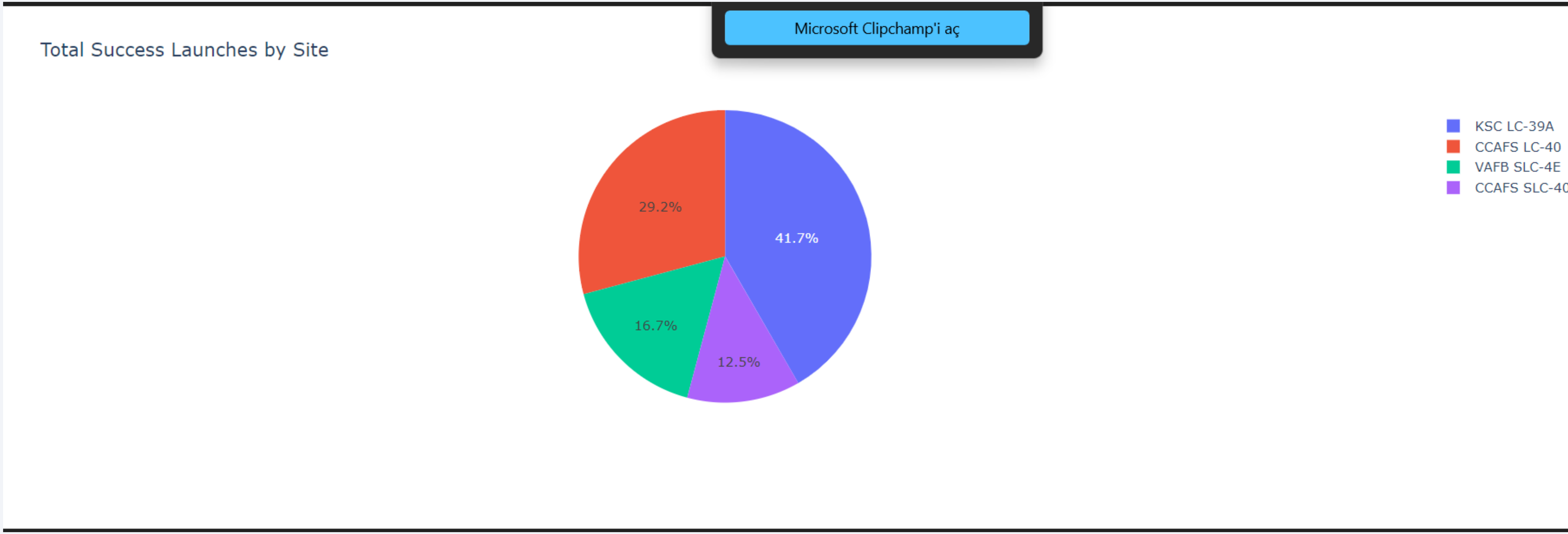
Distance from Launch sites to the Promities

- Here we can see that it is close to the coast and further than the city centers

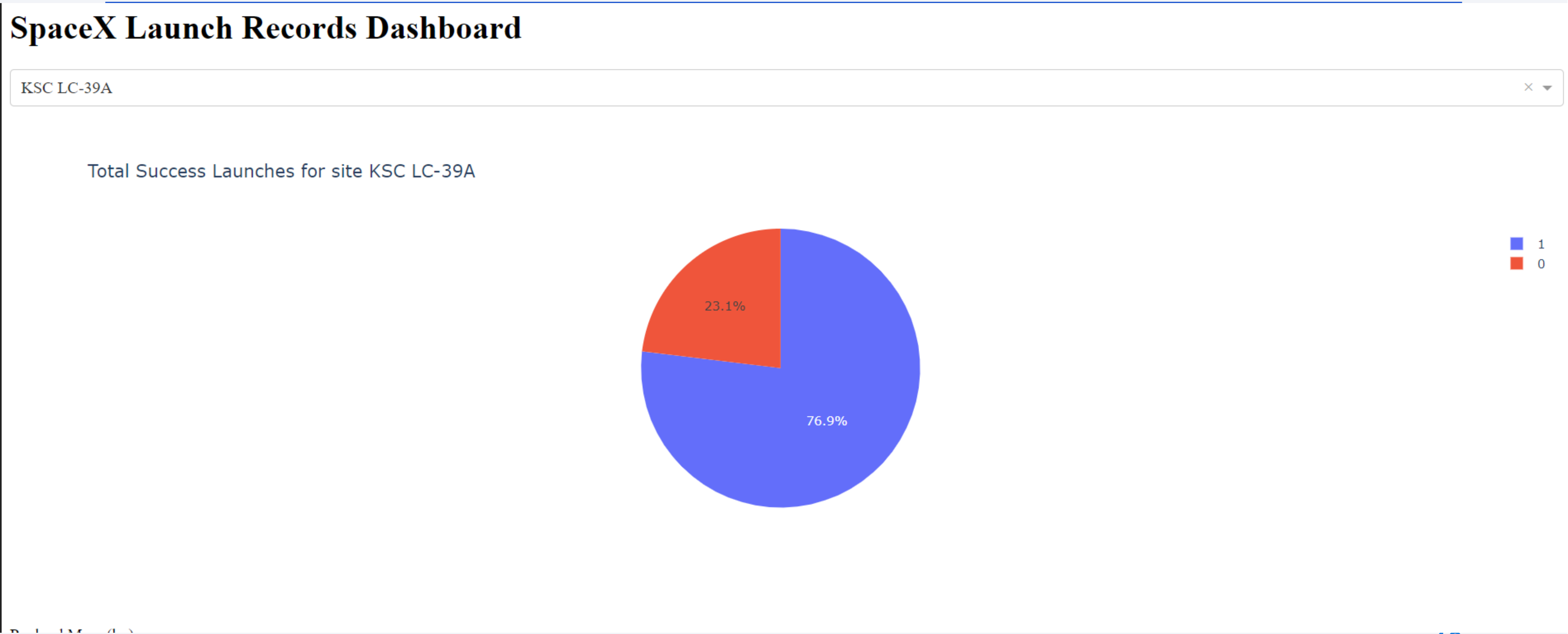
Section 4

Build a Dashboard with Plotly Dash

Total Success Launches by Site

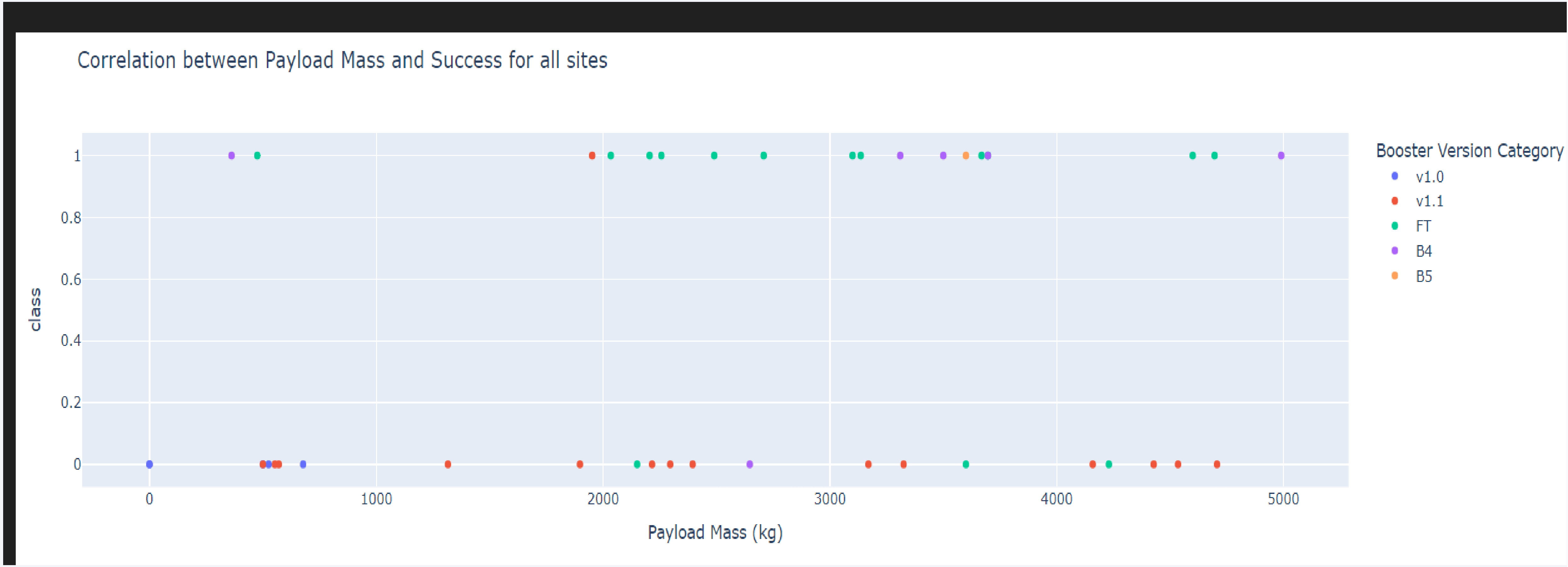


Most Successful Launch Site - KSC LC-39A



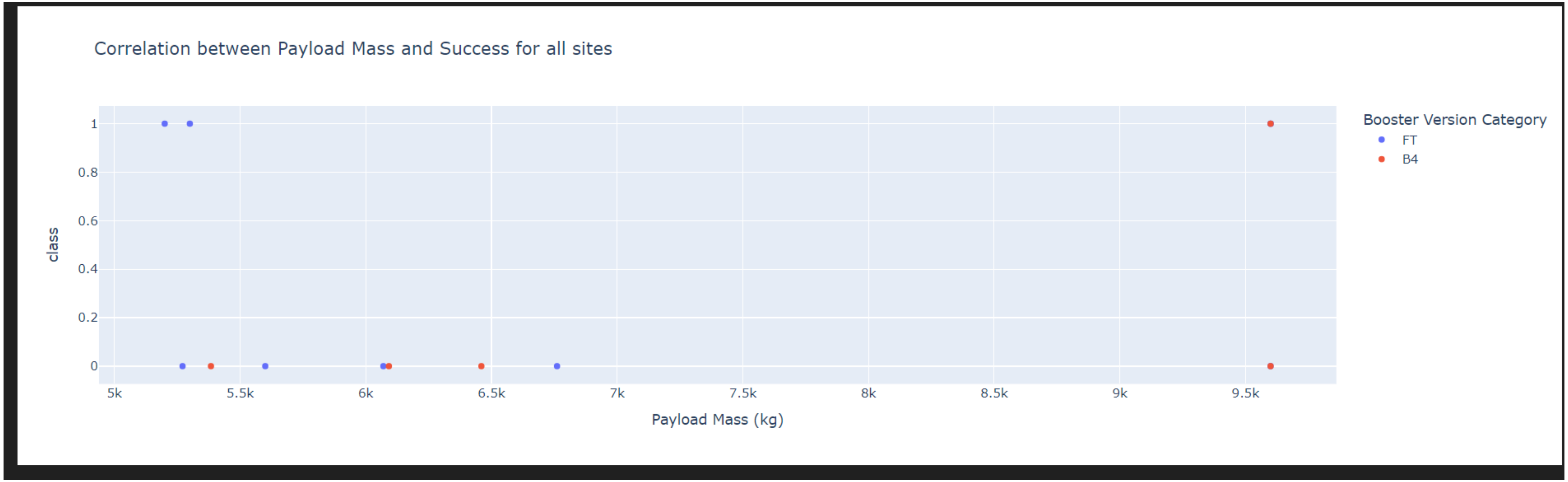
Correlation Between Payload Mass and Success for all Sites 0 - 5000 kg

- As you see when low weighted payload is relatively successful



Correlation Between Payload Mass and Success for all Sites 5000 - 10000 kg

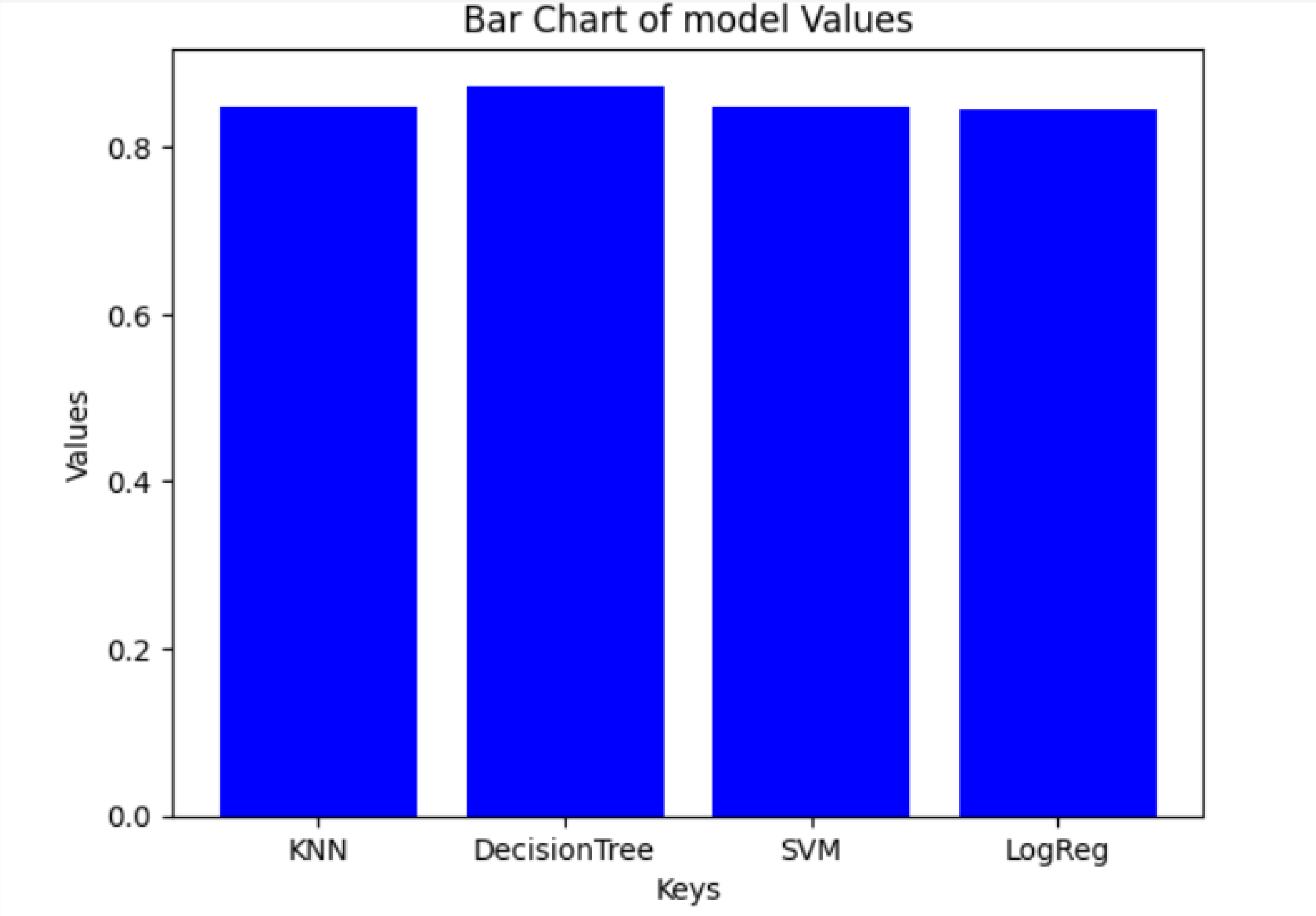
As you see when heavy weighted payloads are not successful when compare it to low weighted ones.



Section 5

Predictive Analysis (Classification)

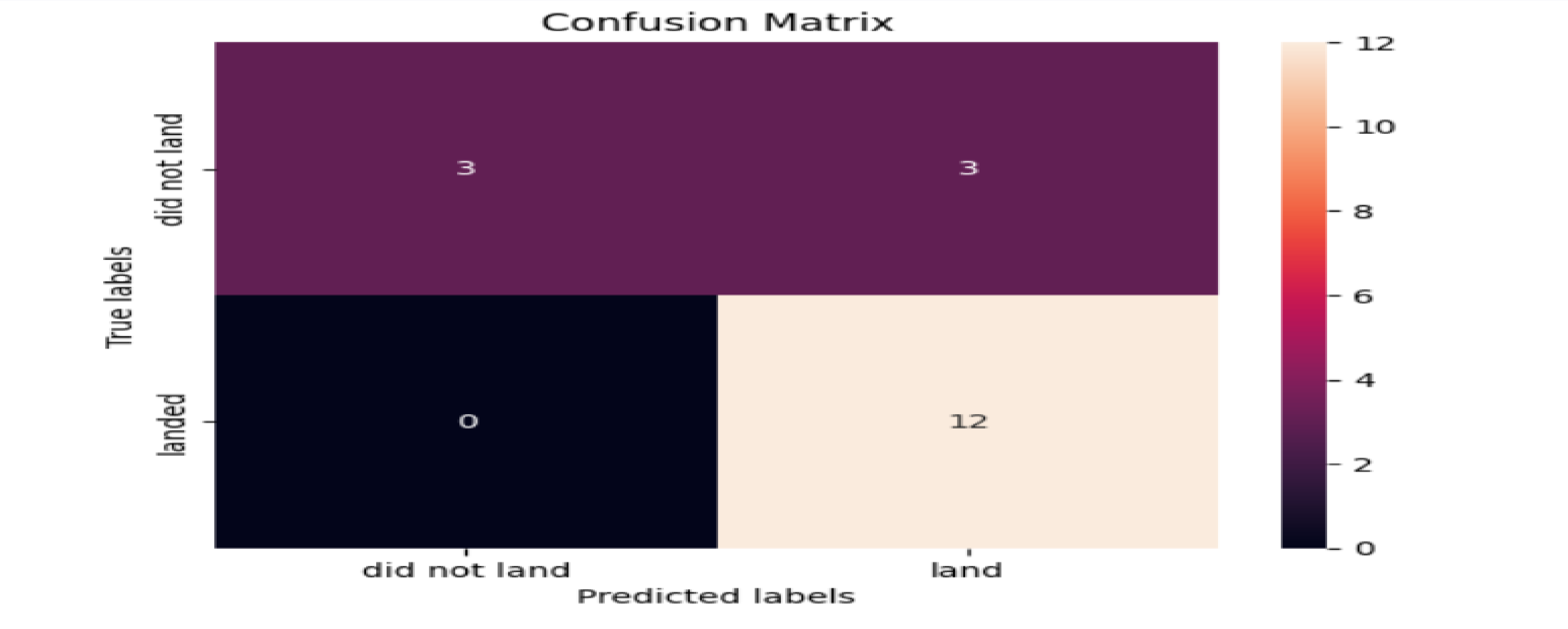
Classification Accuracy



Here we see that Decision tree is slightly better

Confusion Matrix

- We can see that it is good enough overall, Only problem is False Positives . 3 of them are predicted successful instead of failure



Conclusions

- There is a positive correlation between flight amount and succe ss rate.
- Launch success rate is seeming to increase in 2013 till 2020.
- Orbits ES-L1, GEO, HEO, SSO, VLEO had the most success rate.
- KSC LC-39A had the most successful launches of any sites overall.
- The Decision tree classifier is the best choice for this task.

Thank you!