IBM Developer
SKILLS NETWORK

DATA SCIENCE CAPSTONE PROJECT
SPACE X

S.O. / 16.01.2022

Outline

- Executive Summary
- Introduction
- Methodology
- Results
- Conclusion

Executive Summary

- Summary of methodologies
    - ❑ Data collection, exploring, wrangling
    - ❑ Exploratory data analysis
        - ❑ Visualization (matplotlib, seaborn)
        - ❑ SQL
    - ❑ Interactive map with Folium
    - ❑ Dashboard with Plotly Dash
    - ❑ Machine learning models for predictive analysis
- Summary of results
    - ❑ Exploratory data analysis results
    - ❑ Interactive visualization results
    - ❑ Machine learning model prediction results

Introduction

- Project background

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.

The aim of this project is to predict if the Falcon 9 first stage will land successfully.

- What problems do we want to solve?

What are the parameters that affect launch success rate?

How does location affect the launch success rate given the information on previous launch site locations?

What are the optimum conditions for predicting successful rocket landing rate?

SECTION 1

METHODOLOGY

Methodology

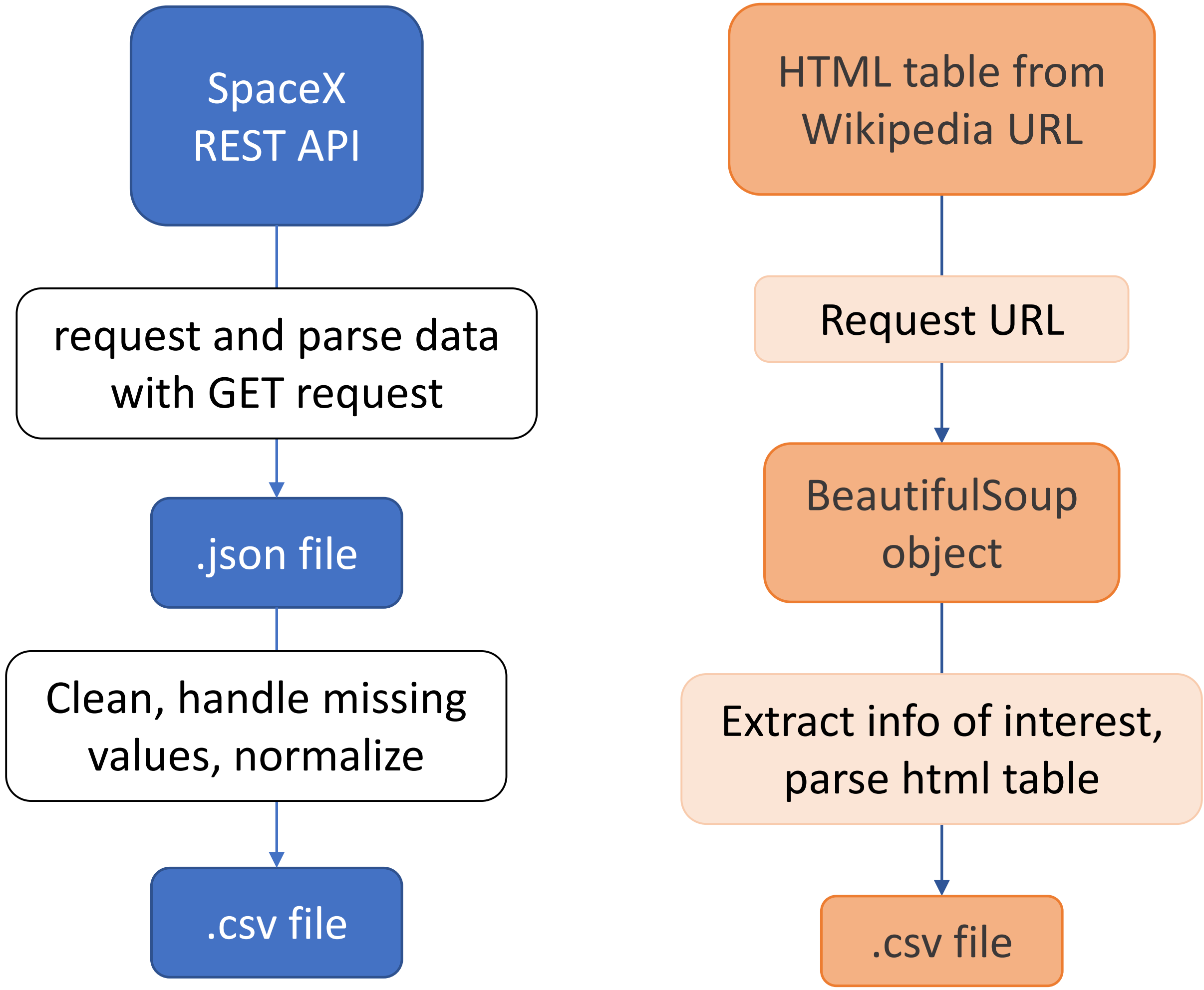
Executive Summary

- Data collection methodology:
  - SpaceX REST
  - Web scrapping (https://en.wikipedia.org/wiki/List_of_Falcon_9_and_Falcon_Heavy_launches)
- Perform data wrangling
  - Find patterns in the data, apply one hot encoding data fields, choose labels for ML training models
- 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

- We gathered data from SpaceX REST API, and obtained information such as rocket, payload mass, launch site used, outcome of landing, type of landing, number of flights from the core, landing pads used etc.

- We used information from Wikipedia page titled ”List of Falcon 9 and Falcon Heavy Launches” via web scraping to collect historical launch records.



Data Collection – SpaceX API

https://github.com/selenozkan/IBM_Capstone/blob/master/Data%20Collection%20API.ipynb

Request data from API

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

```
response = requests.get(spacex_url)
```

Convert to .json format

```
data = pd.json_normalize(response.json())
```

Clean, organize data with custom functions

```
# Call getLaunchSite
getLaunchSite(data)
```

```
# Call getPayloadData
getPayloadData(data)
```

```
# Call getCoreData
getCoreData(data)
```

Create dictionary

```
launch_dict = {'FlightNumber': list(data['flight_number']),
'Date': list(data['date']),
'BoosterVersion':BoosterVersion,
'PayloadMass':PayloadMass,
'Orbit':Orbit,
'LaunchSite':LaunchSite,
'Outcome':Outcome,
'Flights':Flights,
'GridFins':GridFins,
'Reused':Reused,
'Legs':Legs,
'LandingPad':LandingPad,
'Block':Block,
'ReusedCount':ReusedCount,
'Serial':Serial,
'Longitude': Longitude,
'Latitude': Latitude}
```

Create dataframe from dictionary

```
# Create a data from launch_dict
new_df = pd.DataFrame.from_dict(launch_dict)
```

Data Collection - Scraping

https://github.com/selenozkan/IBM_Capstone/blob/master/Data%20Collection%20with%20Web%20Scraping.ipynb

Request data from HTML and create BeautifulSoup object

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

```
soup = BeautifulSoup(response.content, 'html.parser')
```

Extract table and info from html

```
cells = first_launch_table.find_all('th')

#print(cells)
for row in cells:
    #print(row)
    name=extract_column_from_header(row)
    #print(name)
    if name is not None and len(name) > 0:
        column_names.append(name)
```

Create dictionary

```
launch_dict= dict.fromkeys(column_names)

# Remove an irrelvant column
del launch_dict['Date and time ( )']

# Let's initial the launch_dict with eac
launch_dict['Flight No.'] = []
launch_dict['Launch site'] = []
launch_dict['Payload'] = []
launch_dict['Payload mass'] = []
launch_dict['Orbit'] = []
launch_dict['Customer'] = []
launch_dict['Launch outcome'] = []
# Added some new columns
launch_dict['Version Booster']=[]
launch_dict['Booster landing']=[]
launch_dict['Date']=[]
launch_dict['Time']=[]
```

Fill dictionary with info from html table

```
extracted_row = 0
#Extract each table
for table_number,table in enumerate(soup.find_all('table',"wikitable plainrowheaders coll
apsible")):
   # get table row
    for rows in table.find_all("tr"):
        #check to see if first table heading is as number corresponding to launch a numbe
r
        if rows.th:
            if rows.th.string:
                flight_number=rows.th.string.strip()
                flag=flight_number.isdigit()
        else:
            flag=False
        #get table element
        row=rows.find_all('td')
        #if it is number save cells in a dictonary
        if flag:
            extracted_row += 1
            # Flight Number value
```

Create dataframe from dictionary

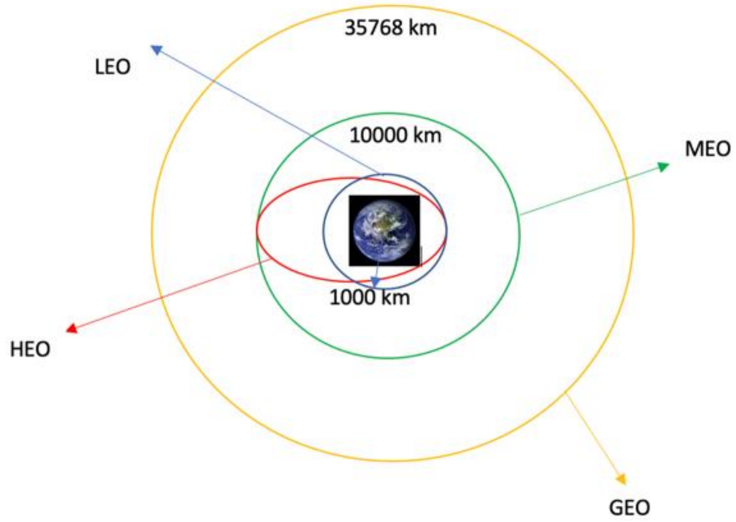
```
df=pd.DataFrame(launch_dict)
df
```

https://github.com/selenozkan/IBM_Capstone/blob/master/EDA.ipynb

Data Wrangling

- In the data set, there are several different cases where the booster did not land successfully. Sometimes a landing was attempted but failed due to an accident; for example,
  - True Ocean : the mission outcome was successfully landed to a specific region of the ocean while
  - False Ocean : the mission outcome was unsuccessfully landed to a specific region of the ocean.
  - True RTLS : the mission outcome was successfully landed to a ground pad
  - False RTLS : the mission outcome was unsuccessfully landed to a ground pad.
  - True ASDS : the mission outcome was successfully landed on a drone ship
  - False ASDS : the mission outcome was unsuccessfully landed on a drone ship.
- We performed exploratory data analysis and calculated the following:
  - Calculated the number of launches at each site
  - Calculated the number and occurrence of each orbit
  - Created the binary landing outcome labels where 1 means successful landing

    0 means unsuccessful landing of the booster.



https://github.com/selenozkan/IBM_Capstone/blob/master/EDA%20with%20Data%20Visualization.ipynb

EDA with Data Visualization

- We performed data visualization to see various relationships

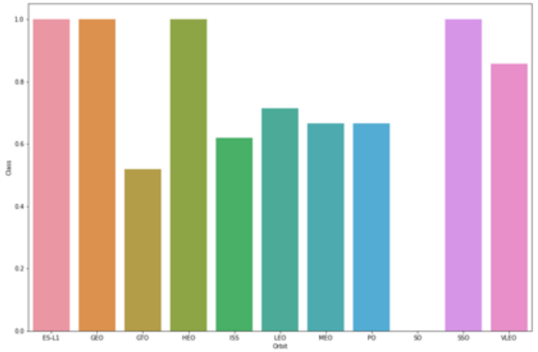
Scatter plot

- Flight number – payload mass
- Flight number – launch site
- Flight number - orbit
- Payload – launch site
- Payload – orbit type
- Orbit – payload mass

These plots will allow us to see the correlation between two numerical values, and how much they are dependent on each other.

Bar plot

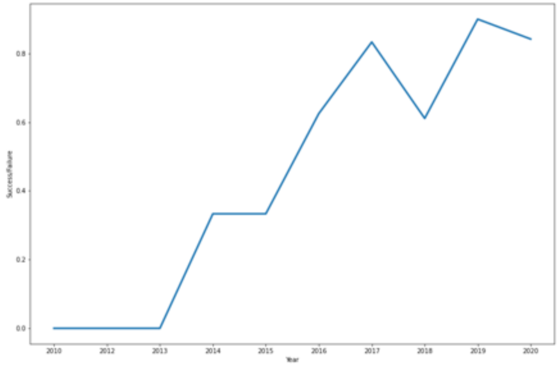
- Mean - orbit



These plots are useful for making comparisons between groups of categorical data

Line plot

- Success rate – years



These plots will allow us to see the trends between two variables clearly.

https://github.com/selenozkan/IBM_Capstone/blob/master/EDA%20with%20SQL.ipynb

EDA with SQL

- We used SQL to get insight from the data and used the following queries to understand the data better:

- 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 acheived
- 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

https://github.com/selenozkan/IBM_Capstone/blob/master/Interactive%20Visual%20Analytics%20with%20Folium%20lab.ipynb

Build an Interactive Map with Folium

The overall aim of creating interactive maps is to understand better the role of launch sites.

- Interactive Maps with Folium were created to mark launch sites.
- Markers were used to show the successful or failed launches on each site
- Launch sites with high success rates were shown with color-labeled markers
- The distances between launch sites and their proximities were calculated

By doing so, we were able to answer the following questions:

- Are launch sites in close proximity to railways, highways or coastlines?
- Do launch sites keep certain distance from the cities?

https://github.com/selenozkan/IBM_Capstone/blob/master/dashboard.py

Build a Dashboard with Plotly Dash

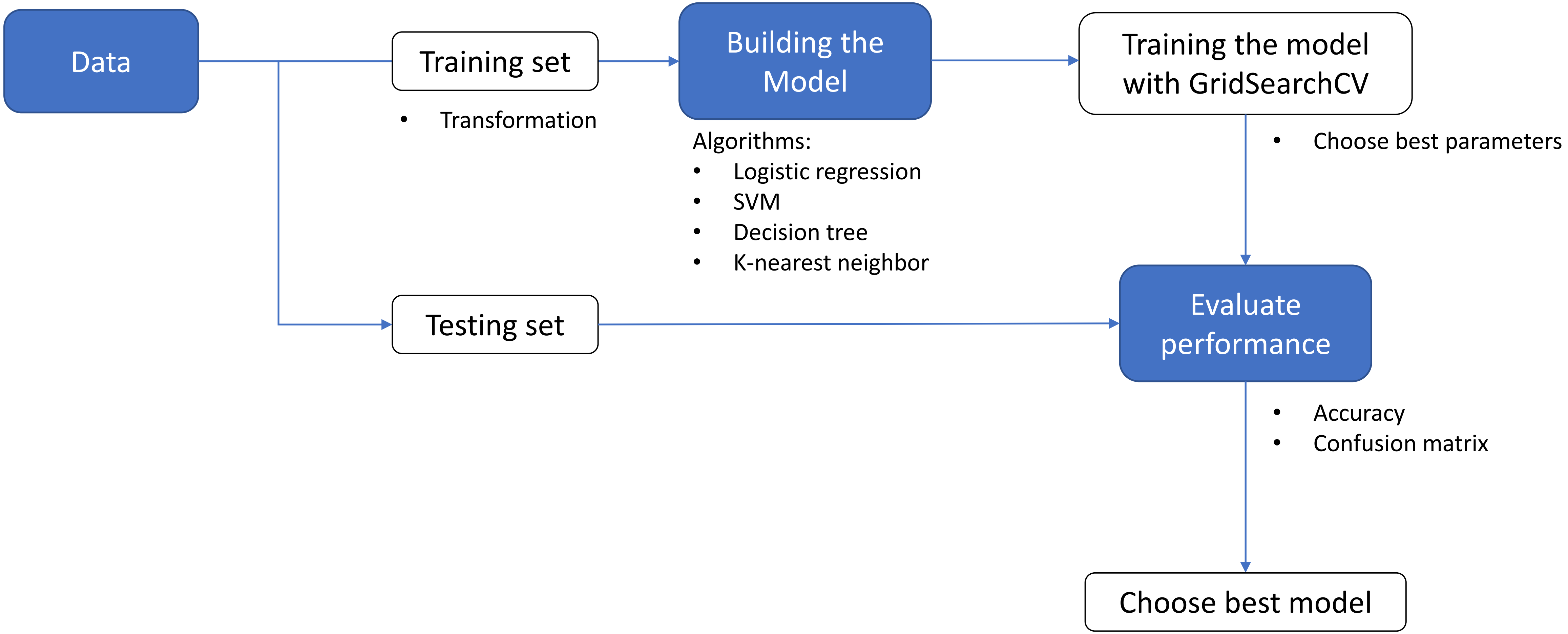
We built an interactive dashboard with Plotly dash

- Pie charts were created to show the total launches by a certain cite or all sites
    - Relative proportions of multiple classes of data were shown
    - Size of circle is made proportional to the total quantity

- Scatter plots were created to show the relationship between outcome and payload mass (kg) for the different booster versions
    - Relationship between variables were shown
    - The range of data flow can be easily determined

https://github.com/selenozkan/IBM_Capstone/blob/master/Machine%20Learning%20Prediction.ipynb

Predictive Analysis (Classification)



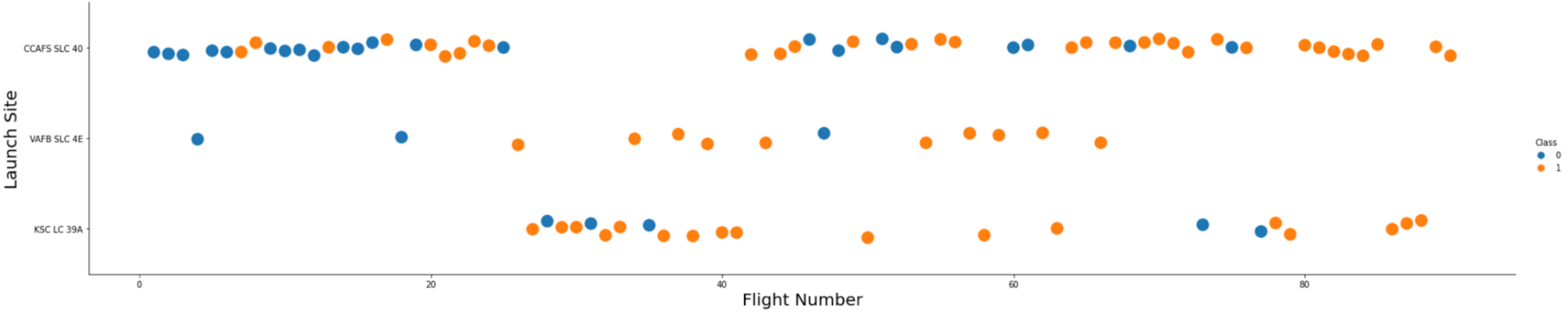
Results

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

SECTION 2

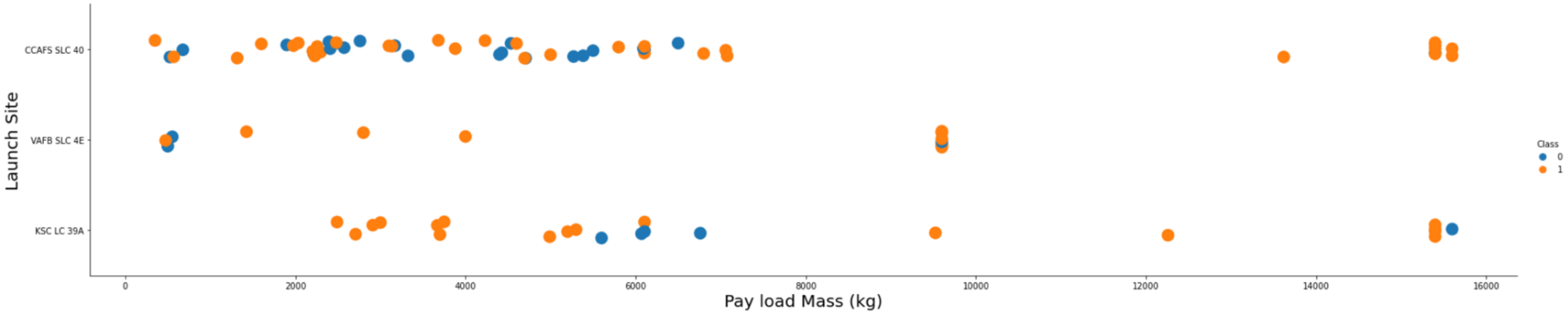
INSIGHTS DRAWN FROM EDA

Flight Number vs. Launch Site



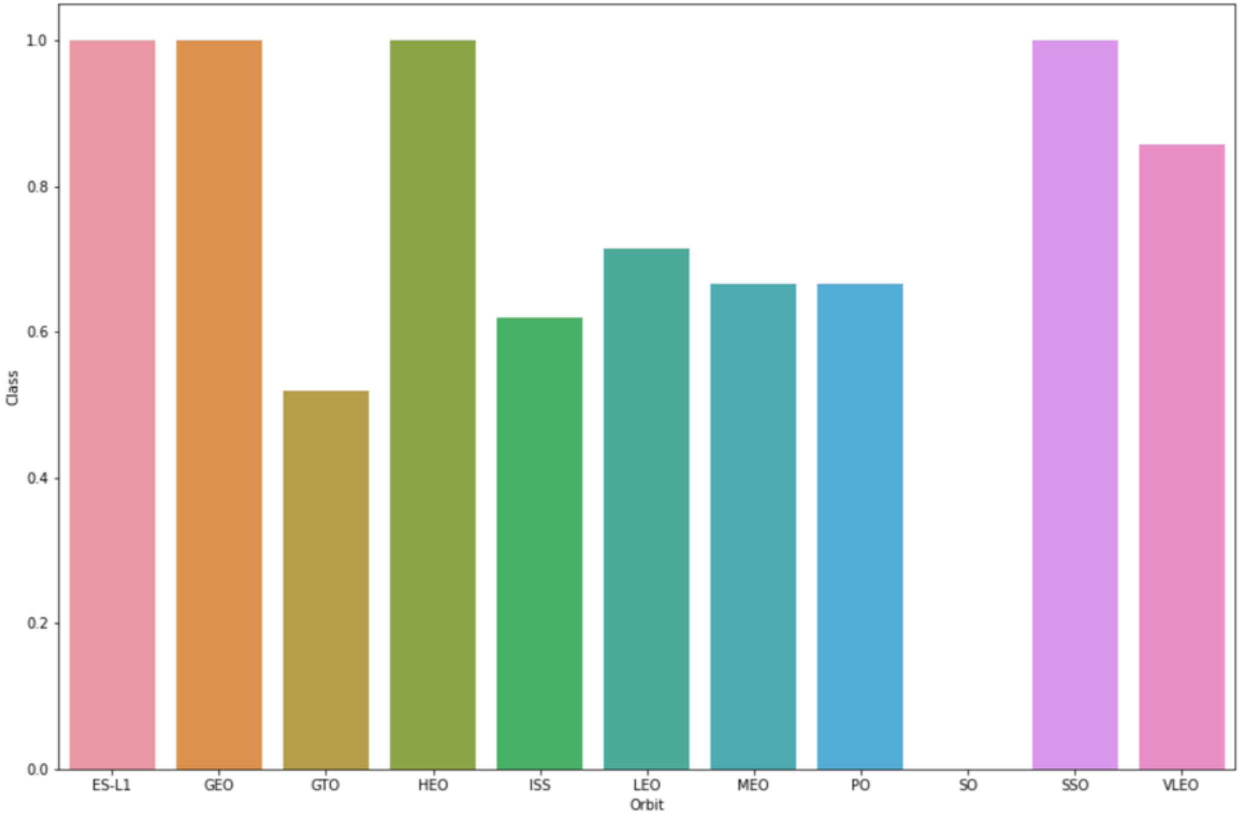
The higher the flight numbers for launch site CCAFS SLC 40 the higher success rate for the rocket

Payload vs. Launch Site



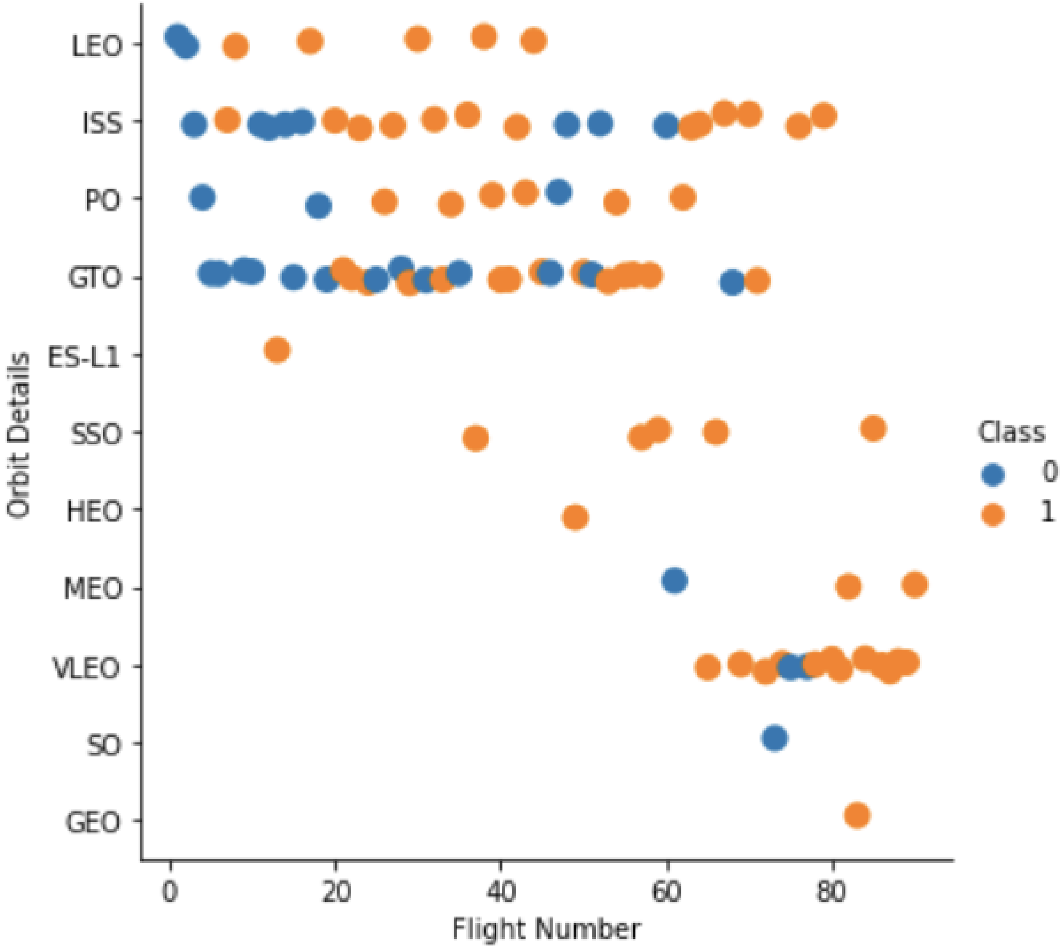
The greater the payload mass for launch site CCAFS SLC 40 the higher success rate for the rocket

Success Rate vs. Orbit Type



We can see from the bar plot that ES-L1, GEO, HEO, SSO, VLEO had the most success rate.

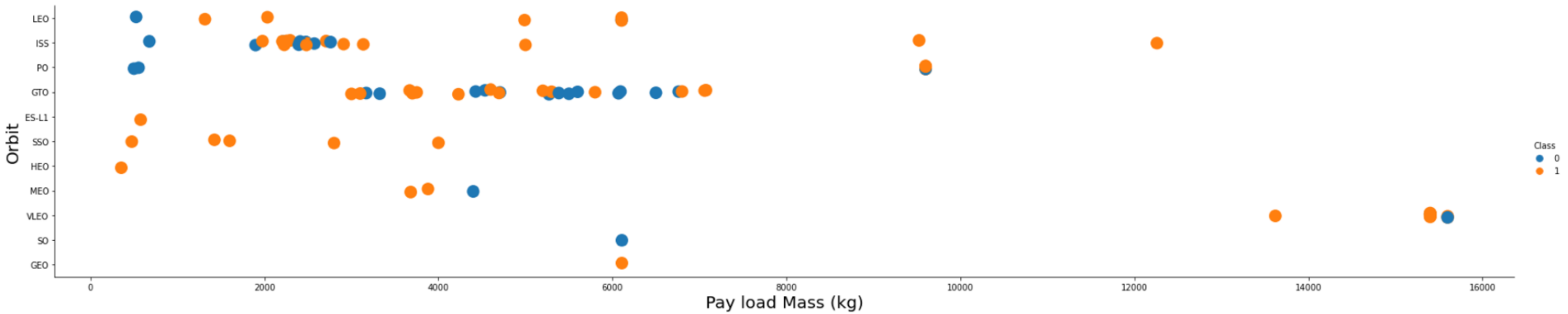
Flight Number vs. Orbit Type



In this plot, we can see that the success rate of LEO orbit depends on the number of flights.

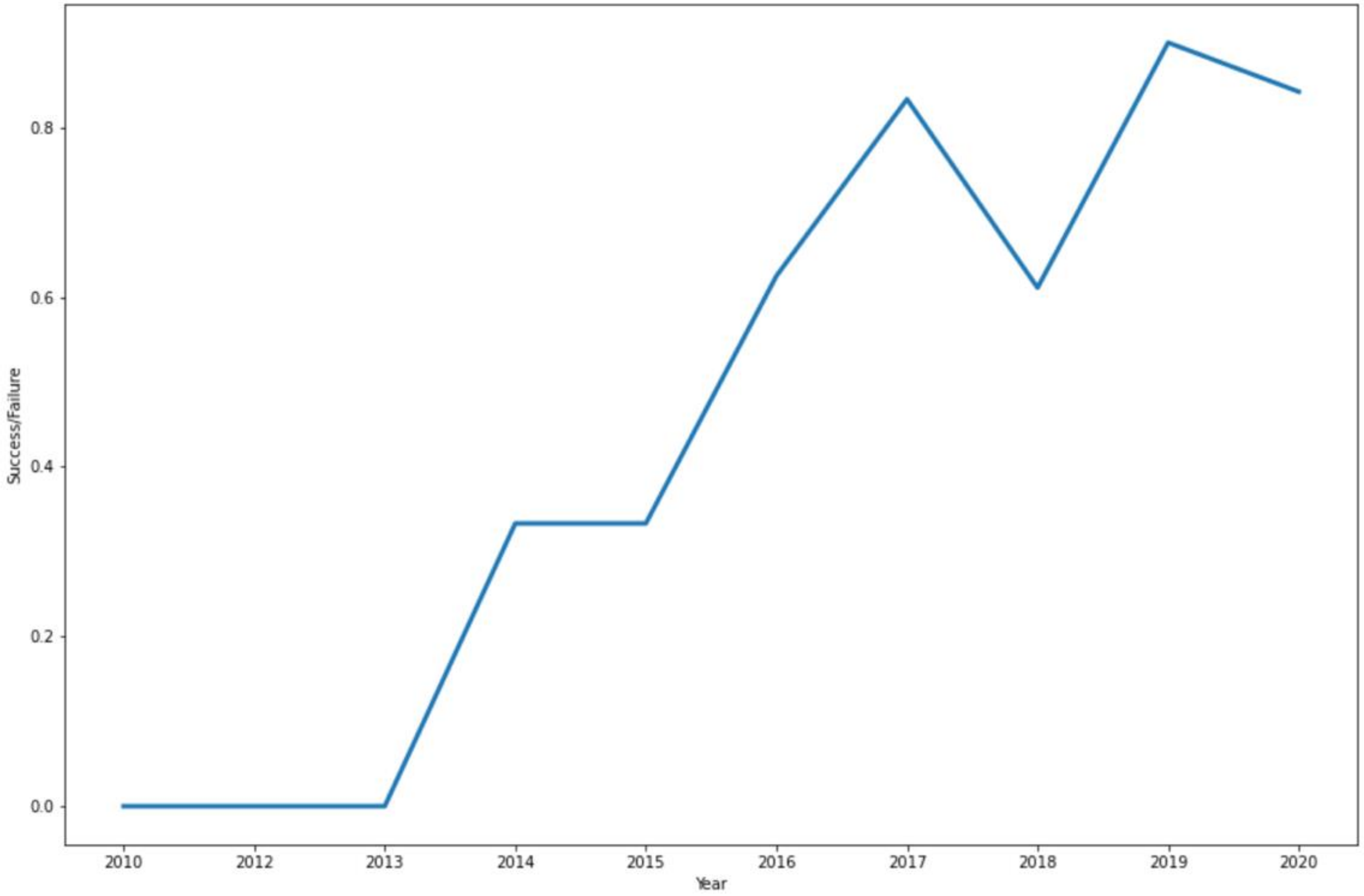
On the other hand, GTO orbit has no relation with the flight number.

Payload vs. Orbit Type



We can see from this plot that with heavy payloads, the rate of successful landing increases for LEO, ISS and PO. However for MEO, it has a negative influence.

Launch Success Yearly Trend



From the line plot, we can observe that success rate since 2013 kept on increasing till 2020

All Launch Site Names

```
%sql SELECT Unique(LAUNCH_SITE) from SPACEXTBL
```

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

We used the key word **DISTINCT** to show only unique launch sites from the SpaceX data.

Launch Site Names Begin with 'CCA'

```
%sql SELECT LAUNCH_SITE from SPACEXTBL where (LAUNCH_SITE) LIKE 'CCA%' LIMIT 5;
```

launch_site
CCAFS LC-40
CCAFS LC-40
CCAFS LC-40
CCAFS LC-40
CCAFS LC-40

We used the query above to display the 5 records where launch site begin with ‘CCA’

Total Payload Mass

```
%sql SELECT sum(PAYLOAD_MASS__KG_) as payloadmass from SPACEXTBL where CUSTOMER = 'NASA (CRS)'
```

payloadmass
45596

We used the query above to calculate the total payload carried by boosters from NASA as 45596

Average Payload Mass by F9 v1.1

```
%sql SELECT avg(PAYLOAD_MASS__KG_) as payloadmass from SPACEXTBL where Booster_Version = 'F9 v1.1'
```

payloadmass
2928

We used the query above to calculate the average payload mass carried by booster version F9 v1.1 as 2928

First Successful Ground Landing Date

```
%sql select min(DATE) from SPACEXTBL where LANDING__OUTCOME = 'Success (ground pad)'
```

1
2015-12-22

We used the query above to find the first successful ground landing date as December 22, 2015

Successful Drone Ship Landing with Payload between 4000 and 6000

```
%sql select BOOSTER_VERSION from SPACEXTBL where LANDING__OUTCOME='Success (drone ship)'
and PAYLOAD_MASS__KG_ BETWEEN 4000 and 6000
```

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

We used the query above using WHERE clause to filter for successfully landed drone ships and those that have payload mass between 4000 and 6000

Total Number of Successful and Failure Mission Outcomes

```
%sql select count(MISSION_OUTCOME) from SPACEXTBL where MISSION_OUTCOME LIKE '%Success%'
```

1
100

We used wildcard LIKE '%Success%' to find successful mission outcomes.

```
%sql select count(MISSION_OUTCOME) from SPACEXTBL where MISSION_OUTCOME LIKE '%Failure%'
```

1
1

We used wildcard LIKE '%Failure%' to find successful mission outcomes.

Boosters Carried Maximum Payload

```
%sql select BOOSTER_VERSION, PAYLOAD_MASS__KG_ from SPACEXTBL where PAYLOAD_MASS__KG_=(se
lect max(PAYLOAD_MASS__KG_) from SPACEXTBL)
```

booster_version	payload_mass__kg_
F9 B5 B1048.4	15600
F9 B5 B1049.4	15600
F9 B5 B1051.3	15600
F9 B5 B1056.4	15600
F9 B5 B1048.5	15600
F9 B5 B1051.4	15600
F9 B5 B1049.5	15600
F9 B5 B1060.2	15600
F9 B5 B1058.3	15600
F9 B5 B1051.6	15600
F9 B5 B1060.3	15600
F9 B5 B1049.7	15600

We used the query above to find the boosters that have carried the maximum payload using a subquery in the WHERE clause and MAX()

2015 Launch Records

```
%sql SELECT MONTH(DATE),MISSION_OUTCOME,BOOSTER_VERSION,LAUNCH_SITE FROM SPACEXTBL where
EXTRACT(YEAR FROM DATE)='2015' and MISSION_OUTCOME LIKE '%Failure%'
```

1	mission_outcome	booster_version	launch_site
6	Failure (in flight)	F9 v1.1 B1018	CCAFS LC-40

We used the query above with WHERE, LIKE and AND conditions to choose failed landing outcomes for year 2015

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

```
%sql SELECT LANDING__OUTCOME, count(LANDING__OUTCOME)  FROM SPACEXTBL WHERE DATE BETWEEN
'2010-06-04' AND '2017-03-20' GROUP BY LANDING__OUTCOME ORDER BY COUNT(LANDING__OUTCOME)
DESC
```

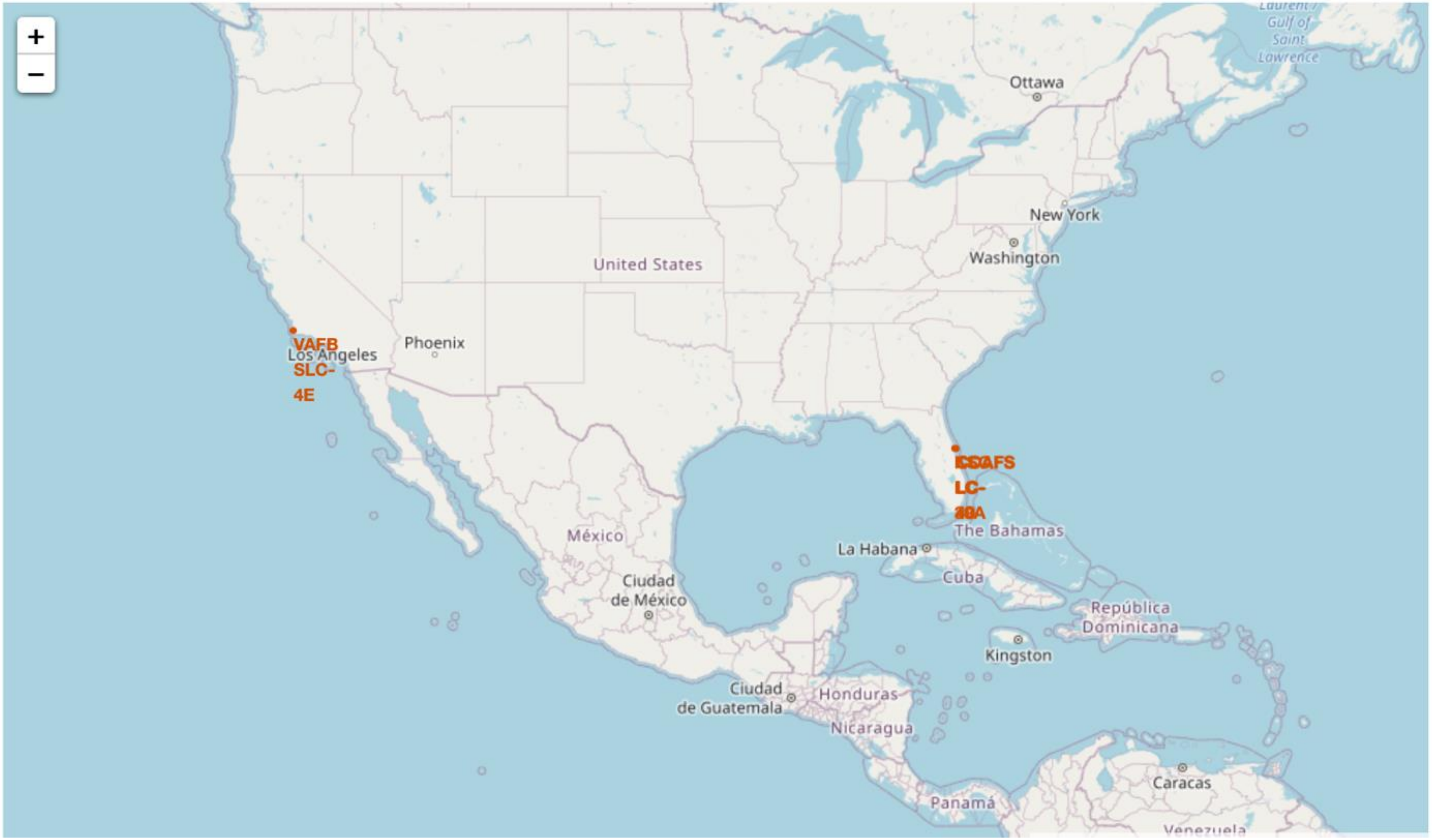
landing__outcome	2
No attempt	10
Failure (drone ship)	5
Success (drone ship)	5
Controlled (ocean)	3
Success (ground pad)	3
Failure (parachute)	2
Uncontrolled (ocean)	2
Precluded (drone ship)	1

We selected landing outcomes and the COUNT of landing outcomes between the two specified dates using WHERE and BETWEEN. We used GROUP BY on landing outcomes and ORDER BY on the landing outcomes in descending order with DESC.

SECTION 3

LAUNCH SITES PROXIMITIES ANALYSIS

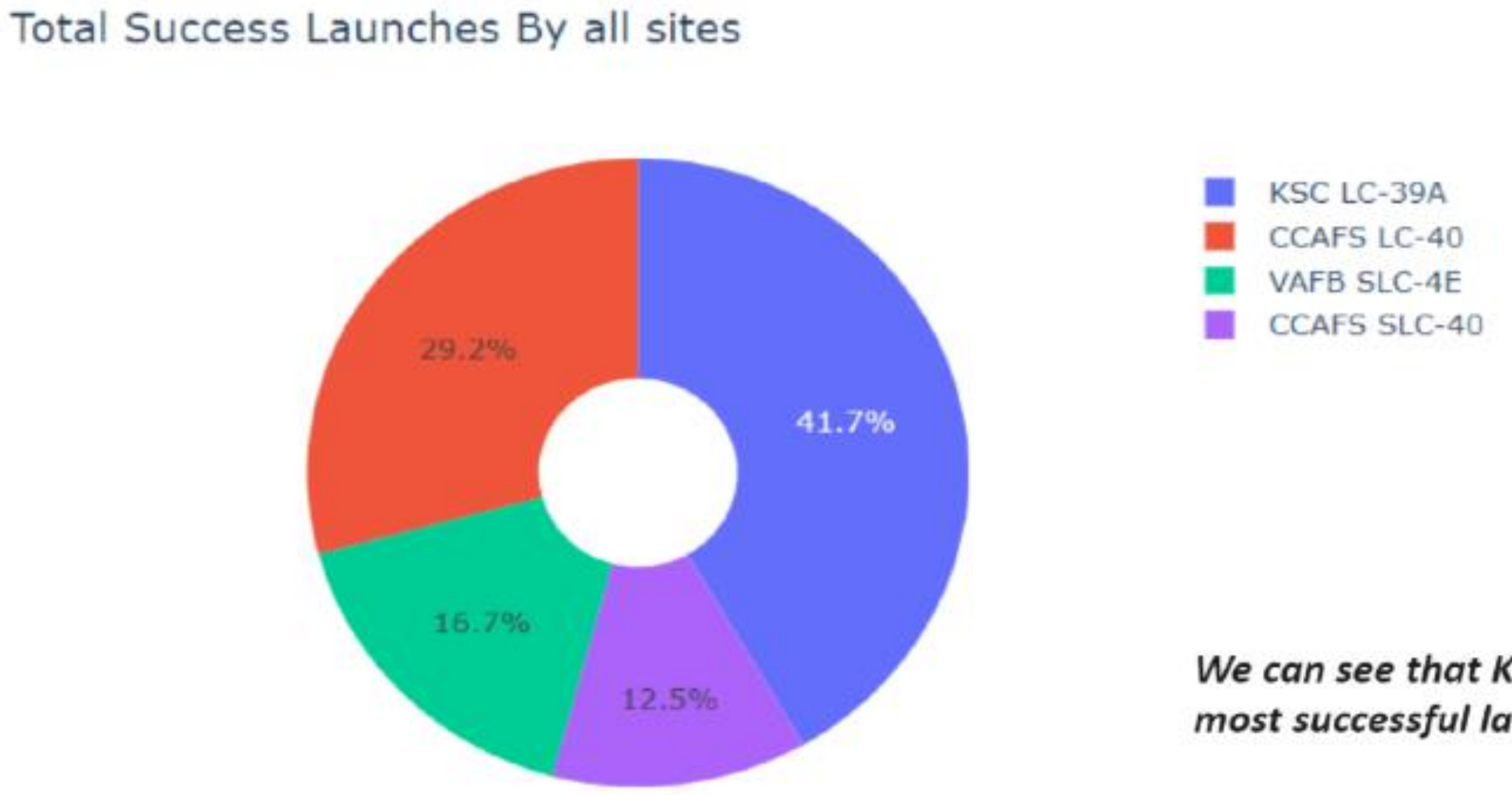
All launch sites global map markers



We see SpaceX launch sites are in the USA; Florida and California

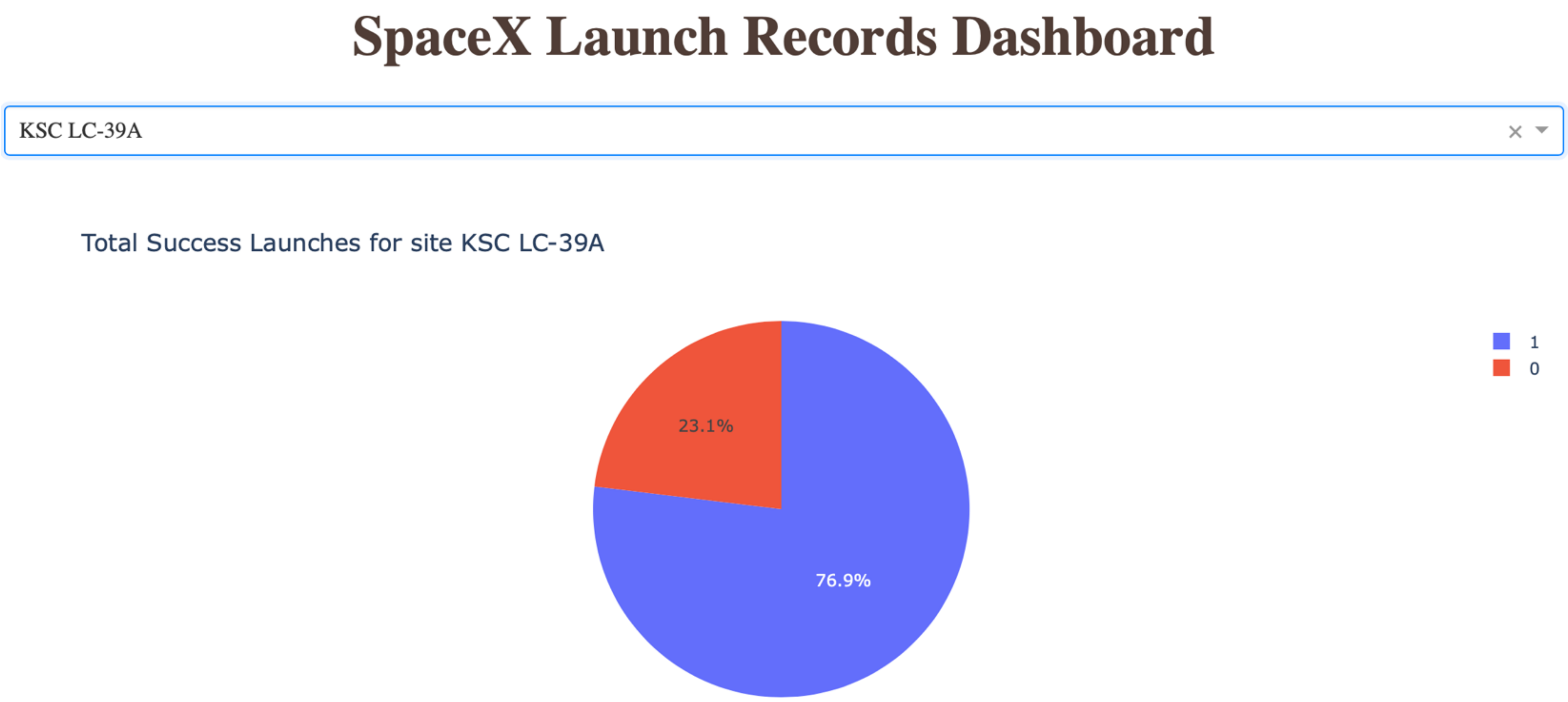
SECTION 4

DASHBOARD WITH PLOTLY DASH



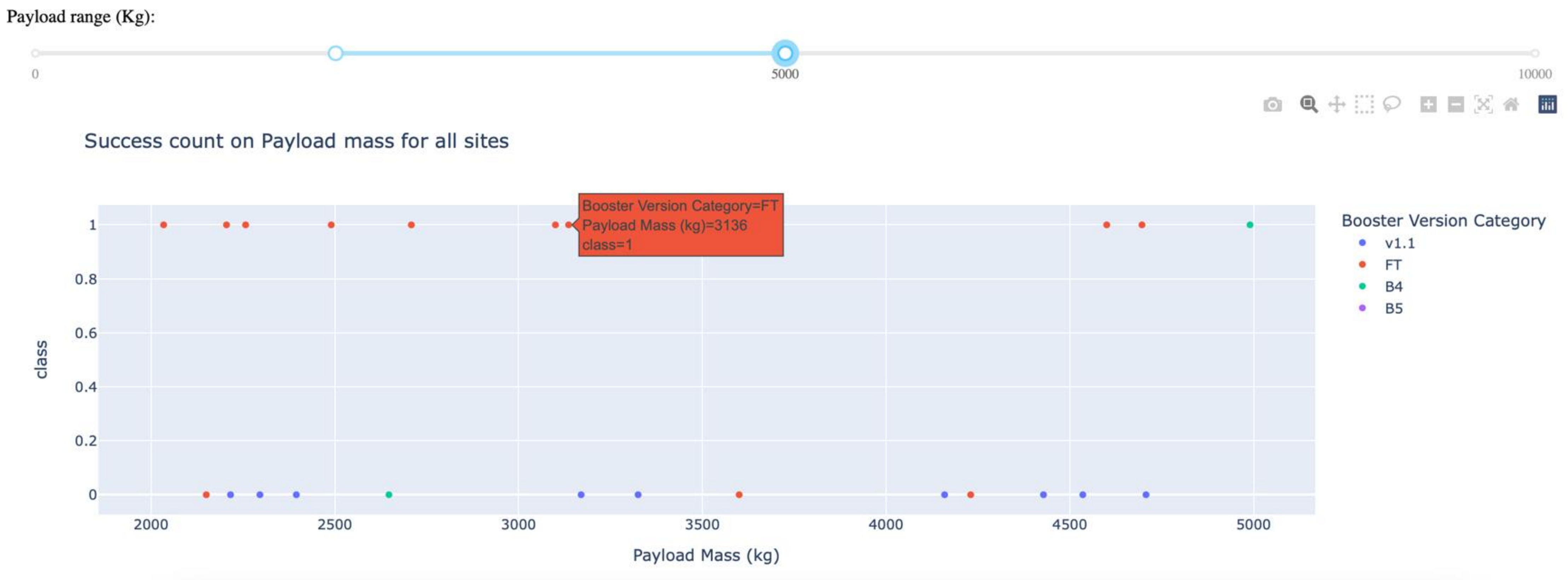
We can see that KSC LC-39A had the most successful launches from all the sites

Pie chart showing success percentage for KSC LC-39A



We see KSC LC-39A has 76.9% success rate and 23.1% failure rate.

Scatterplot of Payload vs LaunchOutcome



We see the distribution of success rates for different payload masses.

SECTION 5

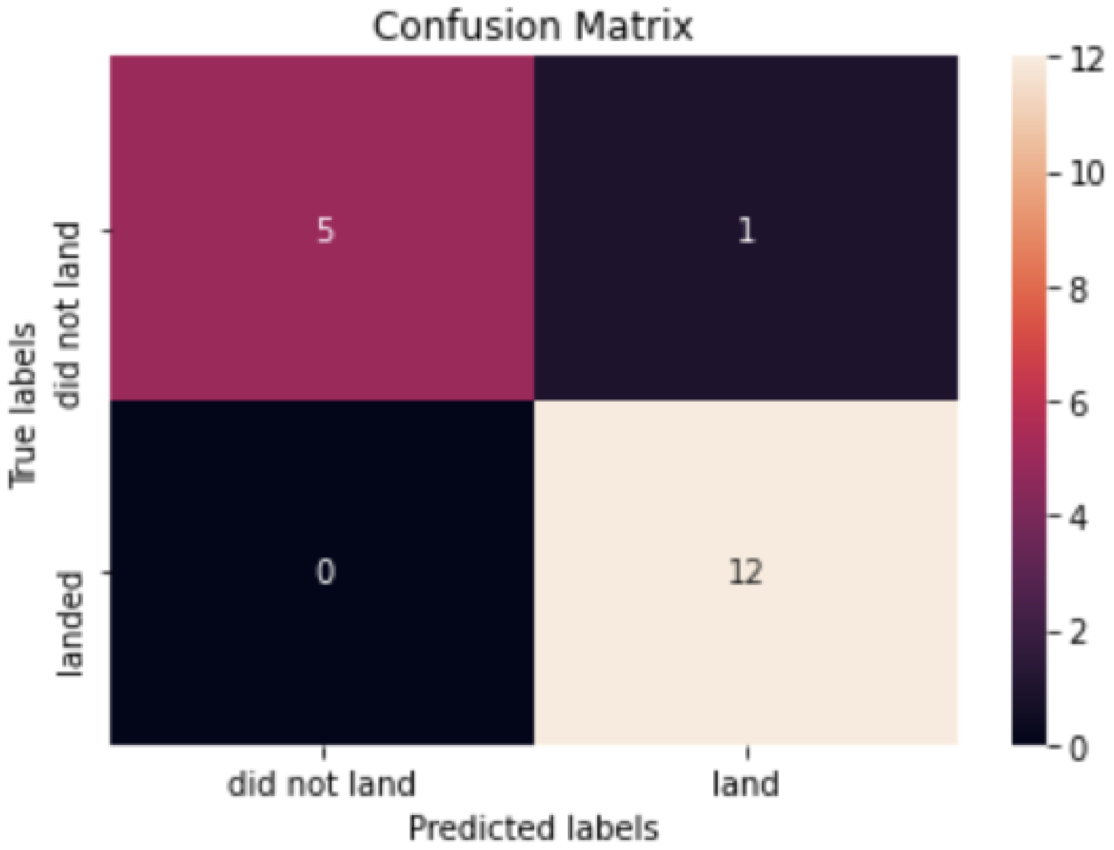
PREDICTIVE ANALYSIS (CLASSIFICATION)

Classification Accuracy

We see that all four models built have more or less same accuracy, however **Decision Tree model** has the highest score

Confusion Matrix

```
yhat = tree_cv.predict(X_test)
plot_confusion_matrix(Y_test,yhat)
```



The confusion matrix for the decision tree classifier shows that the classifier can distinguish between the different classes, with one False Positive and zero False Negative predictions.

Conclusions

- The larger the flight amount at a launch site, the greater the success rate at a launch site.
- Launch success rate started to increase in 2013 till 2020.
- Orbits ES-L1, GEO, HEO, SSO, VLEO have the most success rate.
- KSC LC-39A has the most successful launches of any sites.
- The Decision tree classifier is the best machine learning algorithm for this task.

IBM Developer
SKILLS NETWORK

Thank you