IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Libby Lin
2024 Feb

Outline

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

Executive Summary

- Summary of methodologies
  - Gather raw data from the SpaceX website and the SpaceX API
  - Do EDA on the raw data.
  - Build Machine Learning models to predict Success or Failure in the landing outcome
- Summary of all results
  - Plotly Dash Dashboard built
  - Geospatial analysis complete
  - Predictive Machine Learning models built and compared

Introduction

- Project background and context
  - SpaceX Falcon rockets has a track record of success and failures, during the landing.
- Problems you want to find answers
  - I want to find out the combination of factors and predict the outcome for future SpaceX missions

Section 1

Methodology

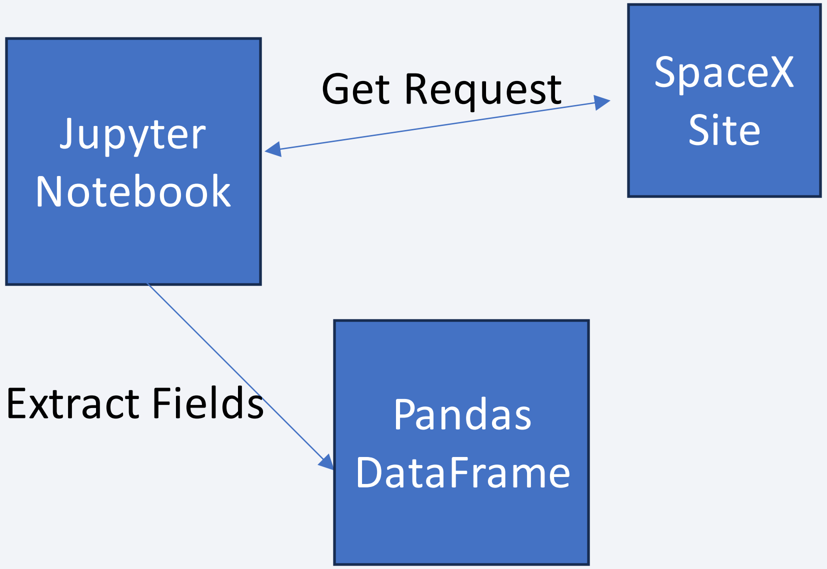
Methodology

Executive Summary

- Data collection methodology:
  - The source data was taken from the SpaceX Website: using the API as well as BeautifulSoup Web Scraping
- Perform data wrangling
  - Analysed nulls, and basic stats of the data
- Perform exploratory data analysis (EDA) using visualization and SQL
  - Load into a sqllite database and visualize graphs
- Perform interactive visual analytics using Folium and Plotly Dash
  - Built a Plotly dashboard
  - Used Folium to analyse the geospatial information
- Perform predictive analysis using classification models
  - Built Logistic Regresion, SVM Decision Tree, KNN models

Data Collection

- Describe how data sets were collected.
  - Webscraped the SpaceX website and the SpaceX API using HTTP GET request
  - Used custom logic to extract fields
  - Appended the rows into a Pandas Dataframe

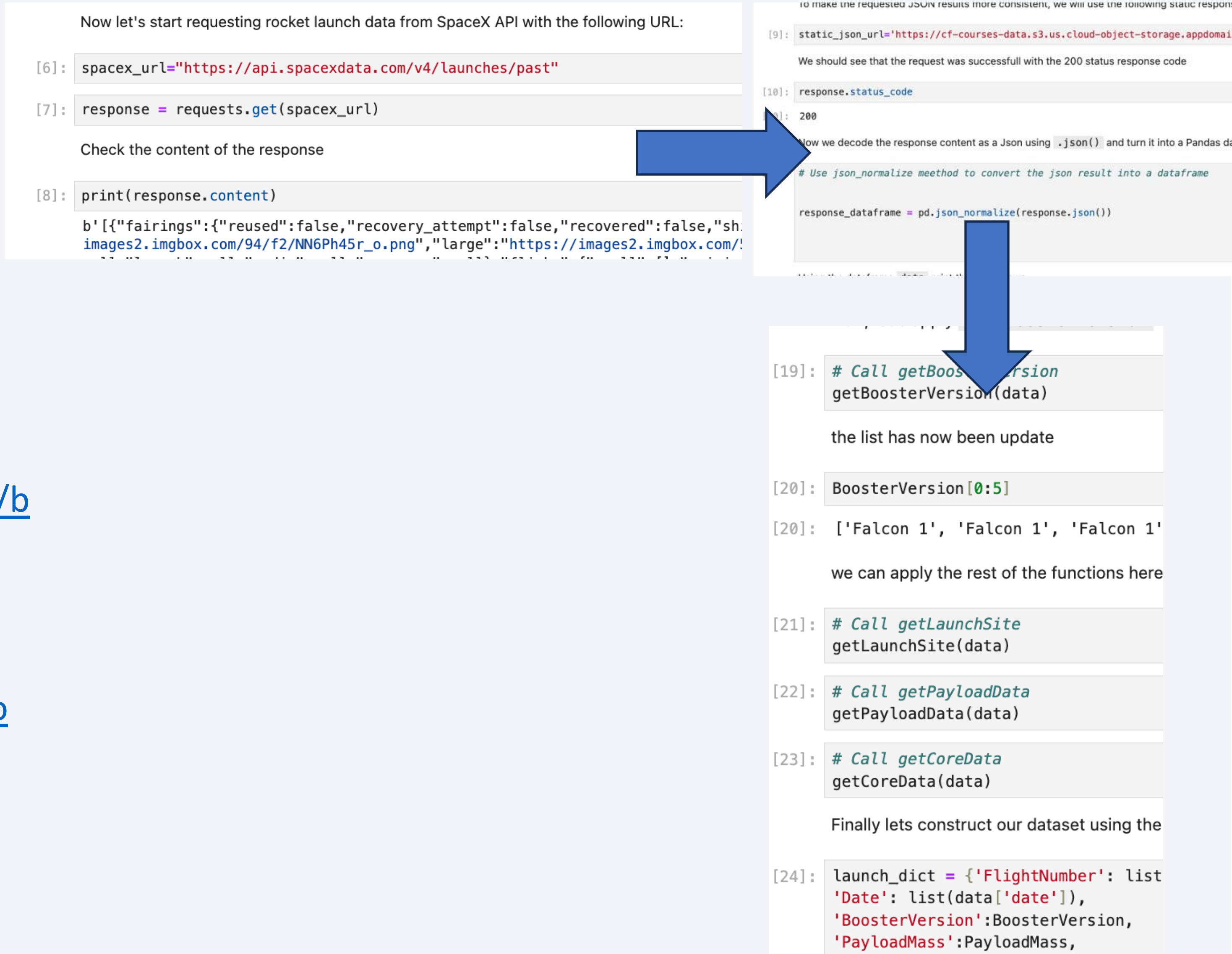


Data Collection – SpaceX API

- Present your data collection with SpaceX REST calls using key phrases and flowcharts
- Call API
- Extract Fields
- Build Dataframe

- Add the GitHub URL: https://github.com/hippopond/testrepo/blob/main/jupyter-labs-spacex-data-collection-api_submit.ipynb

- Repo: https://github.com/hippopond/testrepo



Data Collection - Scraping

- Use BeautifulSoup and Find_all tables
- For loop through each row and extract
- Output to DataFrame
- Add the GitHub: https://github.com/hippopond/testrepo/blob/main/jupyter-labs-webscraping_submit.ipynb

In [9]:
```
# Use the find_all function in the BeautifulSoup object, with element
# Assign the result to a list called `html_tables`

html_tables = spacex_beautifulsoup.find_all("table")
```

Starting from the third table is our target table contains the actual launch records.

In [10]:
```
# Let's print the third table and check its content
first_launch_table = html_tables[2]
print(first_launch_table)
```

```
<table class="wikitable plainrowheaders collapsible" style="width: 100%;
```

In [1]:
```
soup = spacex_beautifulsoup

extracted_row = 0
#Extract each table
for table_number,table in enumerate(soup.find_all
   # get table row
   # pdb.set_trace()
    for rows in table.find_all("tr"):
        #check to see if first table heading is a
        if rows.th:
            if rows.th.string:
                flight_number=rows.th.string.stri
                flag=flight_number.isdigit()
        else:
            flag=False
        #get table element
        row=rows.find_all('td')
        #if it is number save cells in a dictonar
        if flag:
            extracted_row += 1
            # Flight Number value
            # TODO: Append the flight_number into
            #print(flight_number)
            launch_dict['Flight No.'].append(flig
            datatimelist=date_time(row[0])

            # Date value
```

Data Wrangling

- Data Wrangling
  - Calculate Launches from each site
  - Calculate Launches for each Orbit
  - Determine and append Outcome labels

- Add the GitHub URL: https://github.com/hippopond/testrepo/blob/main/labs-jupyter-spacex-Data%20wrangling_submit.ipynb

Use the method `value_counts()` on the column

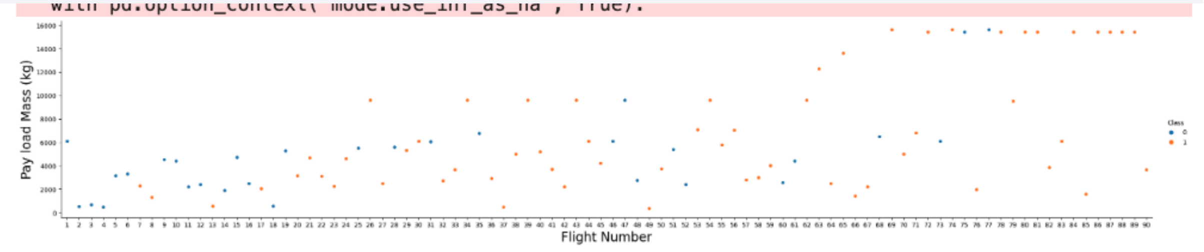
```
[5]: # Apply value_counts() on column LaunchSite

     df['LaunchSite'].value_counts()

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

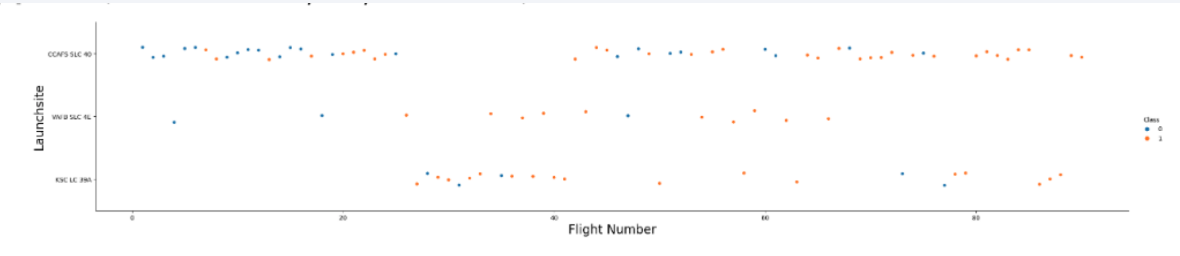
```
[6]: # Apply value_counts

     df['Orbit'].value_co

[6]: GTO       27
     ISS       21
     VLEO      14
     PO         9
     LEO        7
     SSO        5
     MEO        3
     ES-L1      1
     HEO        1
     SO         1
     GEO        1
     Name: O   , dtype:
```

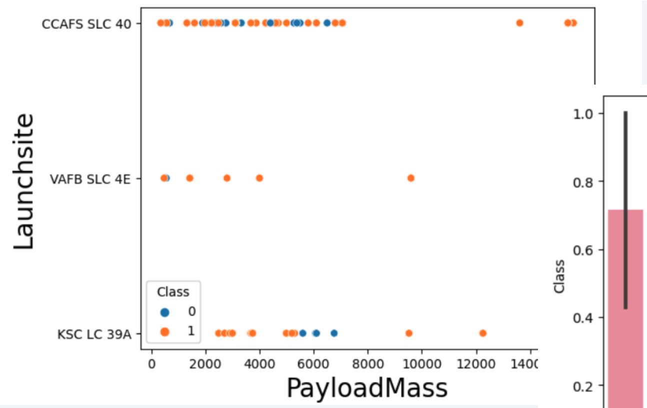
```
[12]: # landing_class = 0 if       outcome
      # landing_class = 1 o        e

      landing_class = []

      for dfindex, dfrow in df.iterrows():
          print(dfrow['Outcome'])
          print(bad_outcomes)
          if (dfrow['Outcome'] in bad_outcomes):
              # print("Bad")
              landing_class.append(0)
          else:
              # print("good")
              landing_class.append(1)

{'None ASDS', 'False Ocean', 'False RTLS',
```

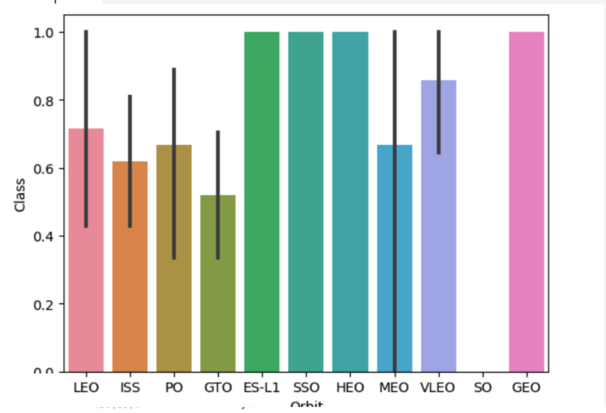
EDA with Data Visualization

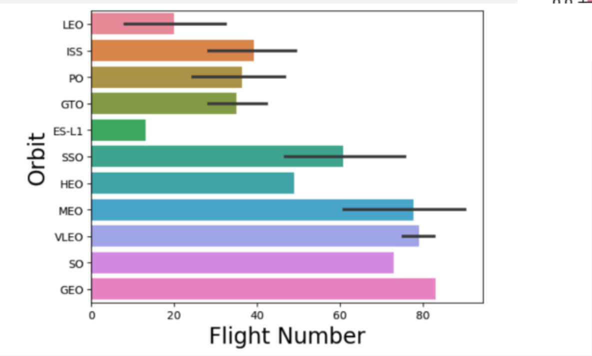
- Summarize what charts:
  - Payload Mass and Flight Number and Outcome
  - Launch Site, Flight Number and Outcome
  - Payload, Launch Site and Outcomes
  - Success Rate of Each Orbit Type
  - Flight Number and Orbit Type
  - Yearly Trend
- Add the GitHub URL:https://github.com/hippopond/testrepo/blob yter-labs-eda-dataviz_submit.ipynb

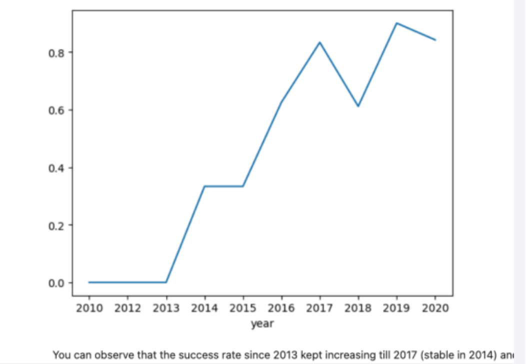








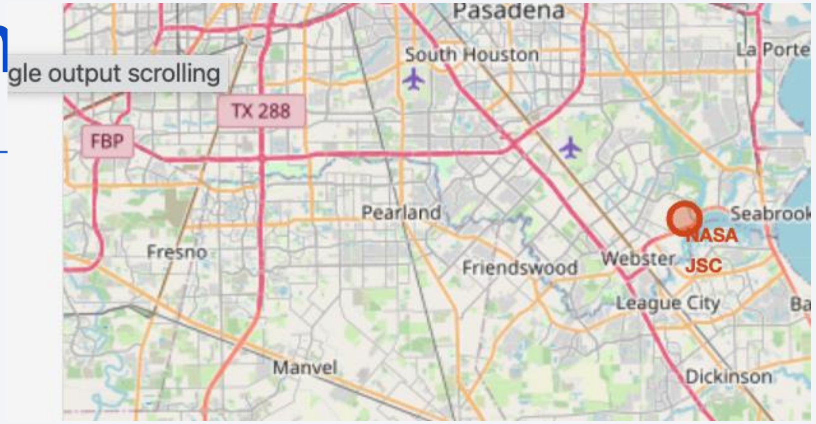




EDA with SQL

- Using bullet point format, summarize the SQL queries you performed
    - Create Table
    - Select all from the SpaceX Table
    - Get all the rows from the site that starts with CCA
    - Get the total payload of the customer that starts with NASA
    - Get the average Payload of Booster F9 v1.1
    - Get the earliest successful launch
    - Get the Booster Version of where it was successful and the payload was between 4000 and 6000
    - Get the count by landing outcomes
    - Get the Booster Versions which carried the highest payloads
    - Get the records that will display the month names in 2015 of launches
    - See the landing outcomes by descending order between 2010 June 4th and 2-17 March 20th
- Add the GitHub URL: https://github.com/hippopond/testrepo/blob/main/jupyter-labs-eda-sql-coursera_sqllite_submit.ipynb

Build an Interactive Map with Folium



- Added in the Markers to mark Nasa, the launch sites
  - This helps to see whether they are near the equator or the coast etc.
  - Utilized Folium Circles and Folium Markers
- Add the GitHub URL: https://github.com/hippopond/testrepo/blob/main/lab_jupyter_launch_site_location_submit.ipynb

Build a Dashboard with Plotly Dash

- Summarize what plots/graphs and interactions you have added to a dashboard
  - A dashboard that shows the outcomes by sites
  - A slider – to adjust the payload mass
- This enables me to examine the launch sites at different payloads
- Add the GitHub URL of your completed Plotly Dash lab: https://github.com/hippopond/testrepo/blob/main/spacex_dash_app_submit.py

Predictive Analysis (Classification)

- Summarize:
  - Model Development:
    - Loaded the data into a DataFrame, Standardized it by Fit and Transform
    - Converted to Float, and split it to Train and Test sets of independent and dependent variables.
    - Then I ran the fit a hyper parameter tuning of Log Regression, Decision Tress, SVM and KNN
  - Evaluated the Models by reviewing the confusion matrix and scores.
- Add the GitHub URL: https://github.com/hippopond/testrepo/blob/main/SpaceX_Machine_Learning_Prediction_Part_5.jupyterlite_submit.ipynb

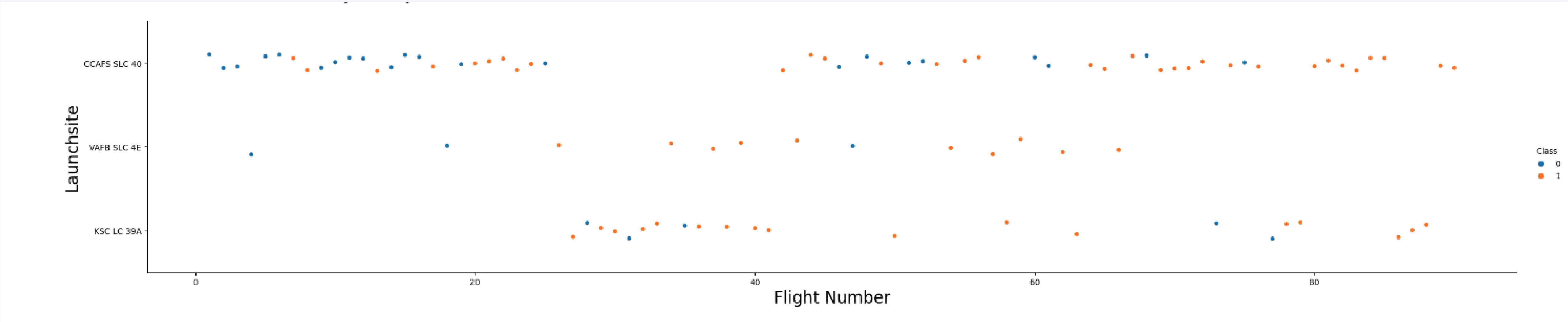
Results

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

Section 2

Insights drawn from EDA

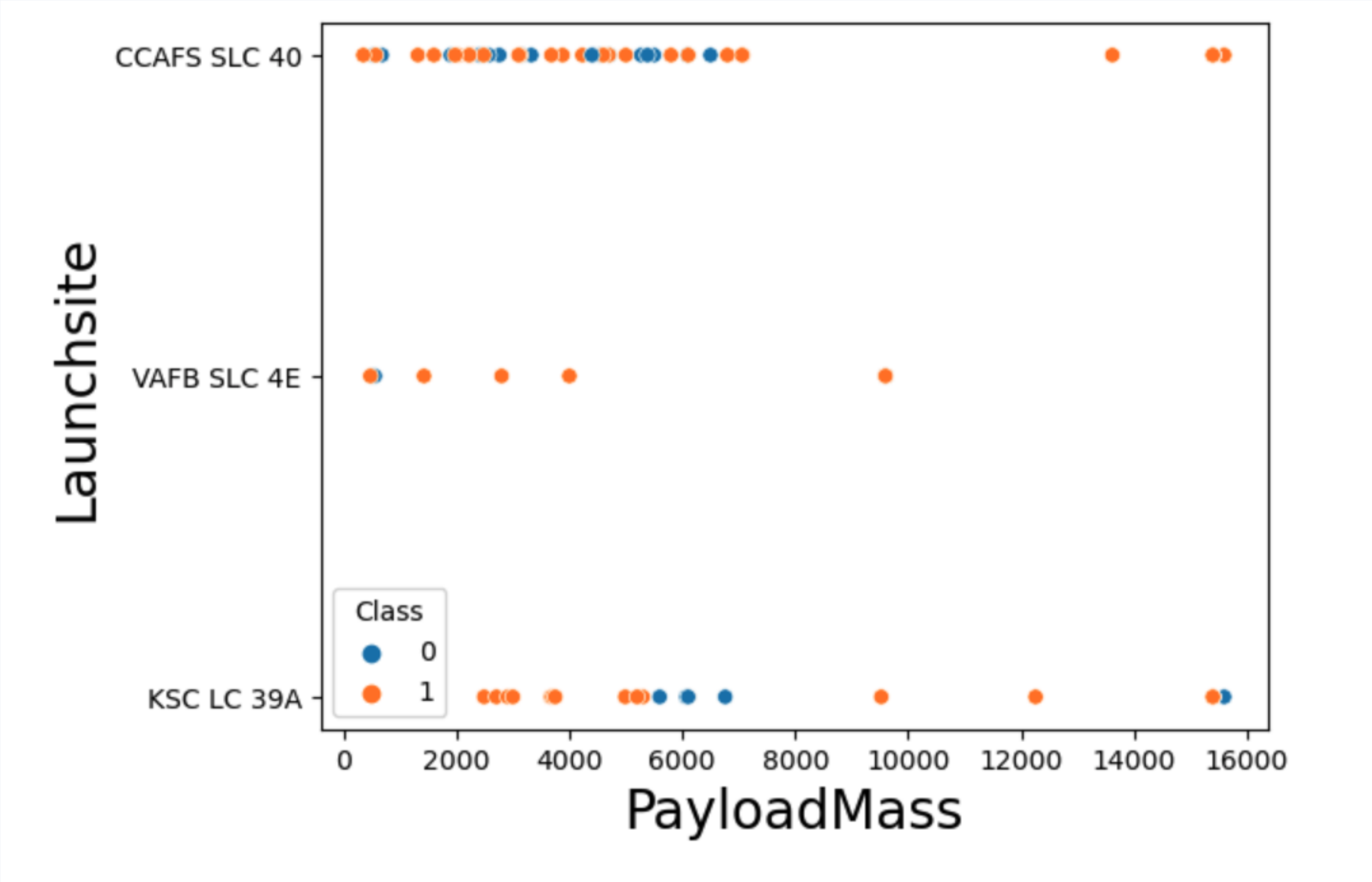
Flight Number vs. Launch Site



We see that as the Flights increase the successes increased. On all sites

Payload vs. Launch Site

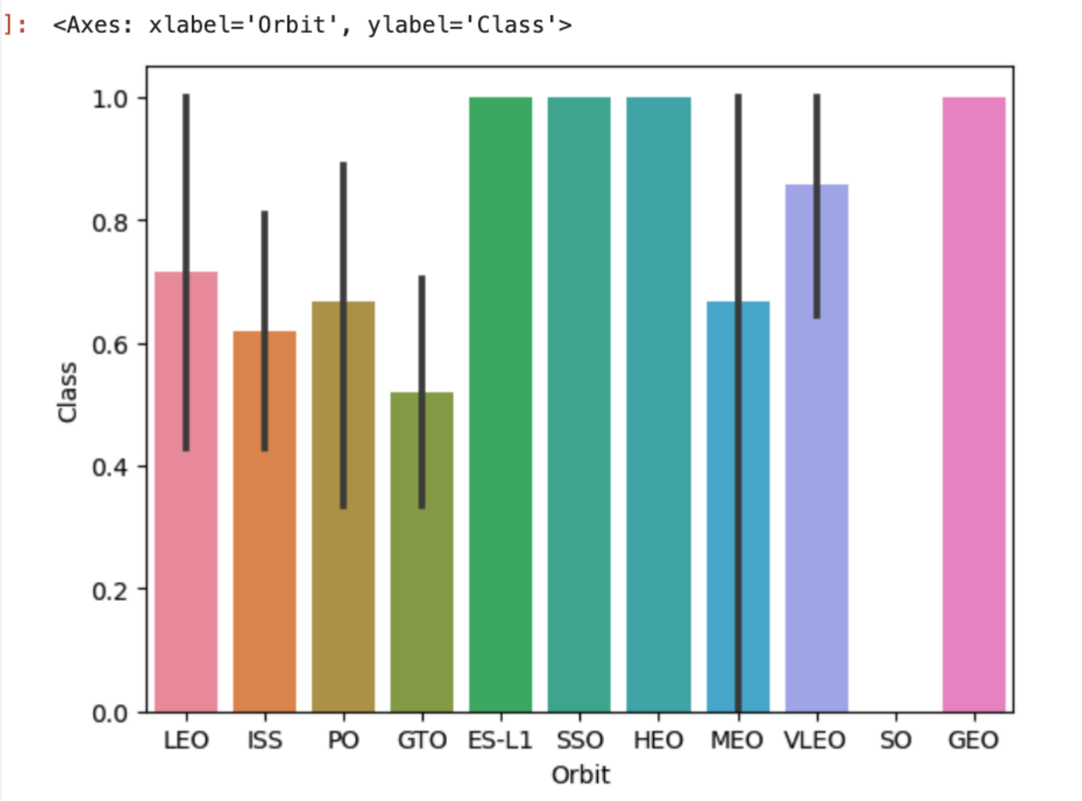
- VAFB site has the best success rate.
- All sites do better at the higher payloads (with 1 failure at KSC)
- CCAFS has spotty record at the lower masses



Success Rate vs. Orbit Type

- Show a bar chart for the success rate of each orbit type

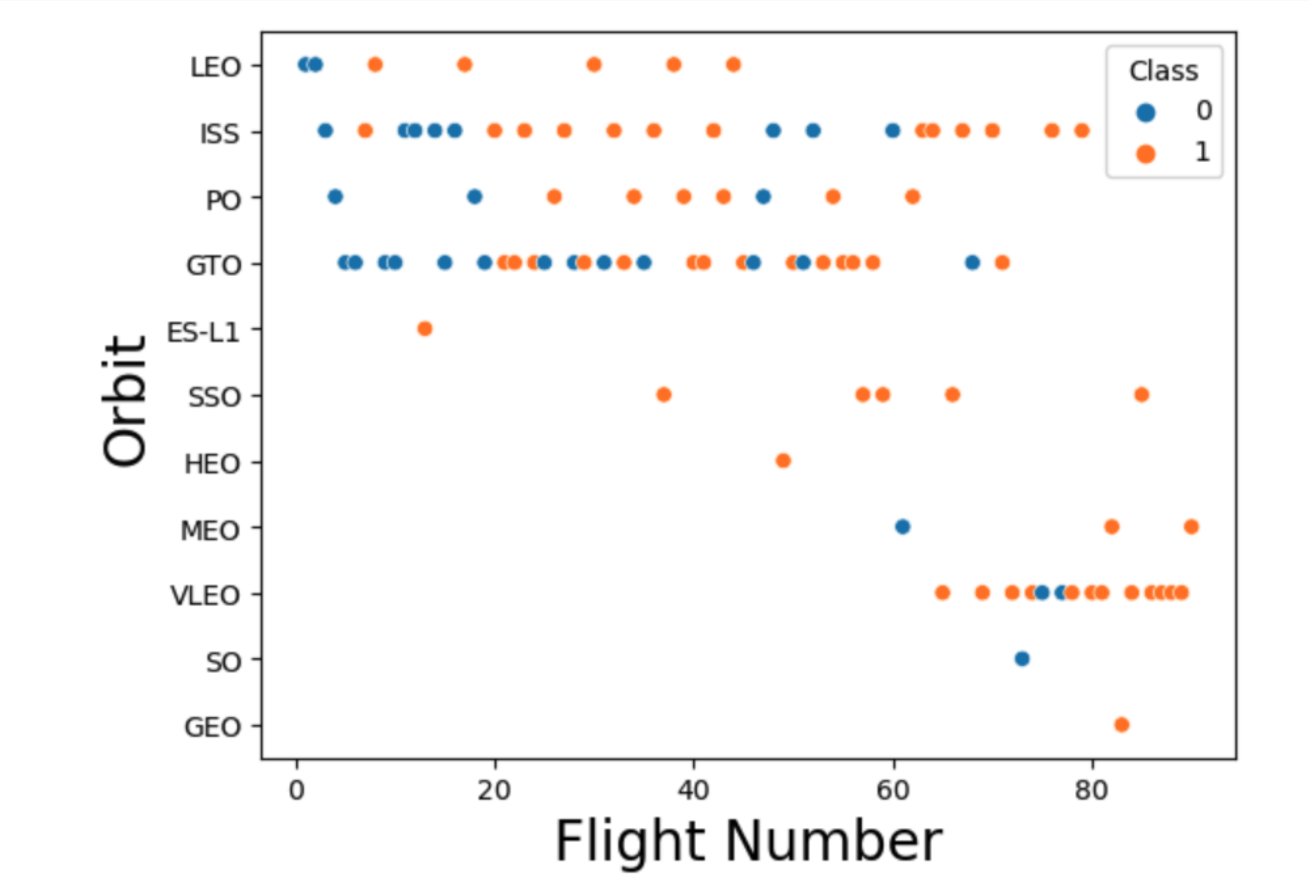
- ES-L1, SSO, HEO has the highest success rate. While GTO is bad



Flight Number vs. Orbit Type

- Show a scatter point of Flight number vs. Orbit type

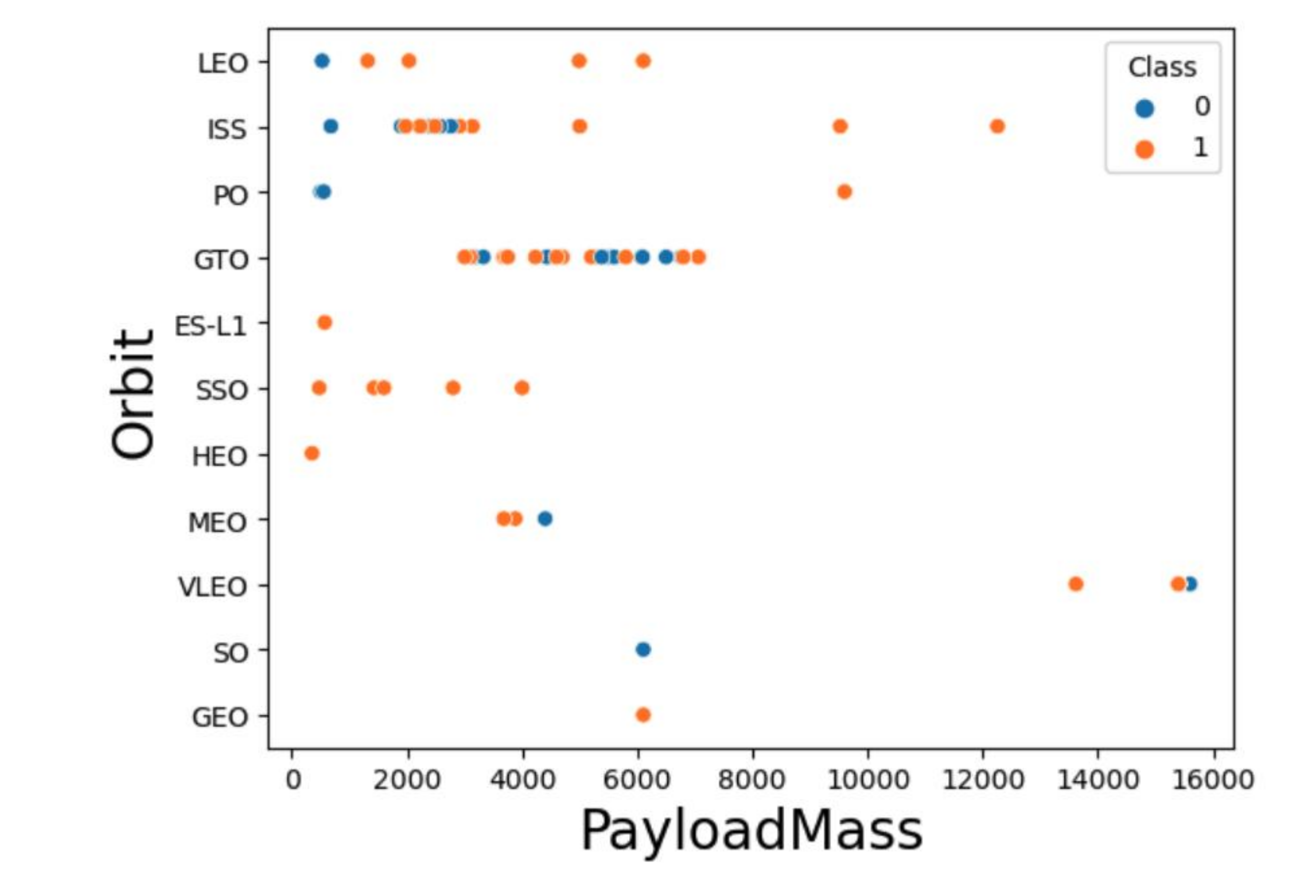
- The color shows the outcome class of the launches and we see that we only went to some MEO, VLEO, SO, GEO orbits at the later flights



Payload vs. Orbit Type

- Show a scatter point of payload vs. orbit type

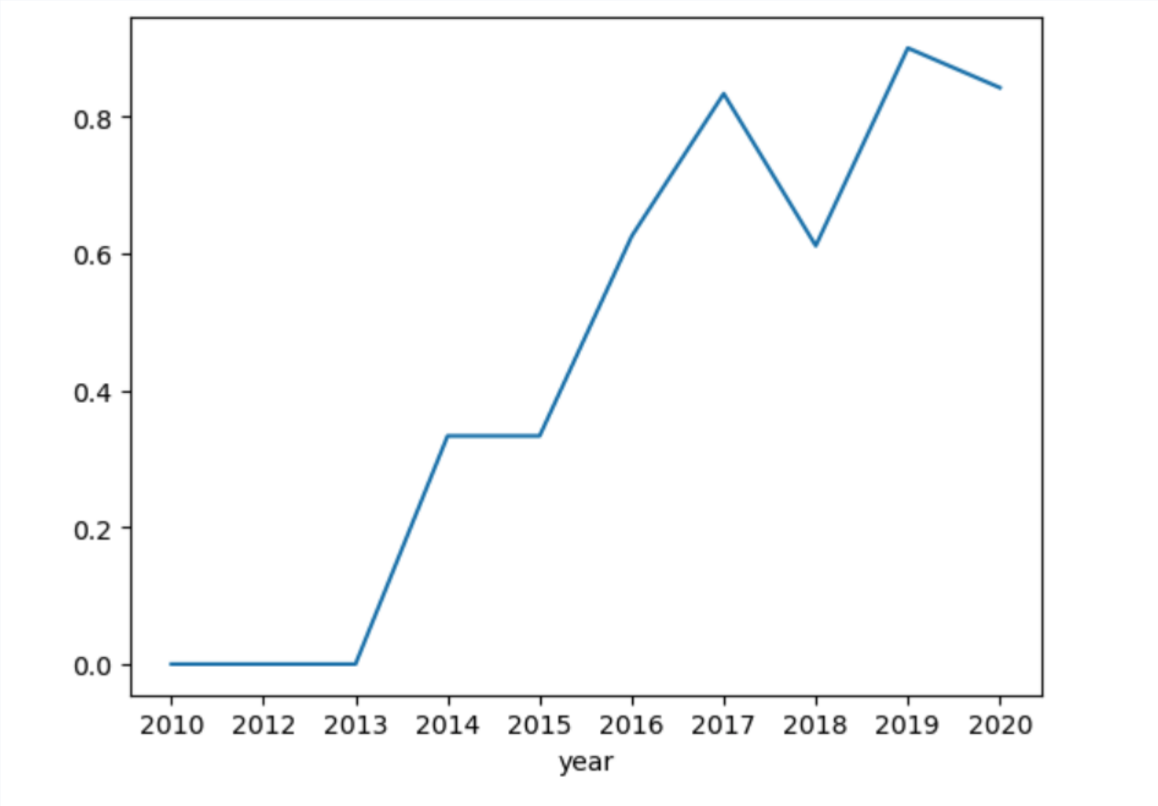
- At higher payloads – the success is better for ISS and PO



Launch Success Yearly Trend

- Show a line chart of yearly average success rate

- As time goes on, the teams gets better !



All Launch Site Names

- Find the names of the unique launch sites
- There are 4 unique launch sites

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

[13]:

```
%sql select distinct Launch_Site from SPACEXTABLE
```

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

[13]:

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

Launch Site Names Begin with 'CCA'

- Here is the Results

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

[9]:

```
%sql select * from SPACEXTABLE WHERE Launch_Site LIKE 'CCA%' limit 5
```

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

[9]:

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Missi
2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	
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	
2012-05-22	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	
2012-10-08	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	
2013-03-01	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	

oggle output scrolling

Total Payload Mass

- Calculate the total payload carried by boosters from NASA
- Total payload mass is 99980KG

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

```
%sql select SUM(Payload_mass__KG_) from SPACEXTABLE WHERE  Customer LIKE 'NASA%';
```

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

SUM(Payload_mass__KG_)
99980

Average Payload Mass by F9 v1.1

- Calculate the average payload mass carried by booster version F9 v1.1
- 2928.4 KG is the average load by booster F9 v 1.1

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

[11]:

```
%sql select AVG(Payload_mass__KG_) from SPACEXTABLE WHERE  Booster_Version =  'F9 v1.1';
```

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

[11]:

AVG(Payload_mass__KG_)
2928.4

First Successful Ground Landing Date

- Find the dates of the first successful landing outcome on ground pad
- 2018 July 22 is the first landing outcome

```
]:  %sql select MIN(Date) from SPACEXTABLE WHERE  Landing_Outcome = 'Success';
```

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

]:

MIN(Date)
2018-07-22

Successful Drone Ship Landing with Payload between 4000 and 6000

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

- Here is the list of boosters

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 Booster_Version from SPACEXTABLE WHERE  Landing_Outcome = 'Success' and Payload_mass__KG_ BETWEEN 4000 and 600
```

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

Booster_Version
F9 B5 B1046.2
F9 B5 B1047.2
F9 B5 B1046.3
F9 B5 B1048.3
F9 B5 B1051.2
F9 B5B1060.1
F9 B5 B1058.2
F9 B5B1062.1

Total Number of Successful and Failure Mission Outcomes

- Calculate the total number of successful and failure mission outcomes

21 no attempts, there are many different types of outcomes

List the total number of successful and failure mission outcomes

```
%sql select Landing_Outcome, count(*) from SPACEXTABLE Group By Landing_Outcome;
```

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

Landing_Outcome	count(*)
Controlled (ocean)	5
Failure	3
Failure (drone ship)	5
Failure (parachute)	2
No attempt	21
No attempt	1
Precluded (drone ship)	1
Success	38
Success (drone ship)	14
Success (ground pad)	9
Uncontrolled (ocean)	2

Boosters Carried Maximum Payload

- List the names of the booster which have carried the maximum payload mass
- Used a sub query to find the maximum and the list of boosters is listed here

[17]:

```
%sql SELECT Booster_Version FROM SPACEXTABLE WHERE Payload_mass__KG_ = \
(SELECT MAX(Payload_mass__KG_) AS MAX_LOAD FROM SPACEXTABLE);
```

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

[17]:

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

- List the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015
- Here's 2 instances of failed drone ship landings in 2015:

```
9]:
#%sql Select substr(Date, 6, 2) from Spacextable where substr(Date,0,5) = '2015';

%sql select booster_version, launch_site from spacextable where substr(Date,0,5) = '2015' \
and Landing_outcome like '%fail%drone%';
```

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

9]:

Booster_Version	Launch_Site
F9 v1.1 B1012	CCAFS LC-40
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

- No attempts was the highest, followed by 5 Successes

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.

[38]:

```
%sql select count(*), Landing_Outcome from spacextable where date between '2010-6-04' and '2017-03-20' \
group by landing_outcome ORDER BY count(*) DESC;
```

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

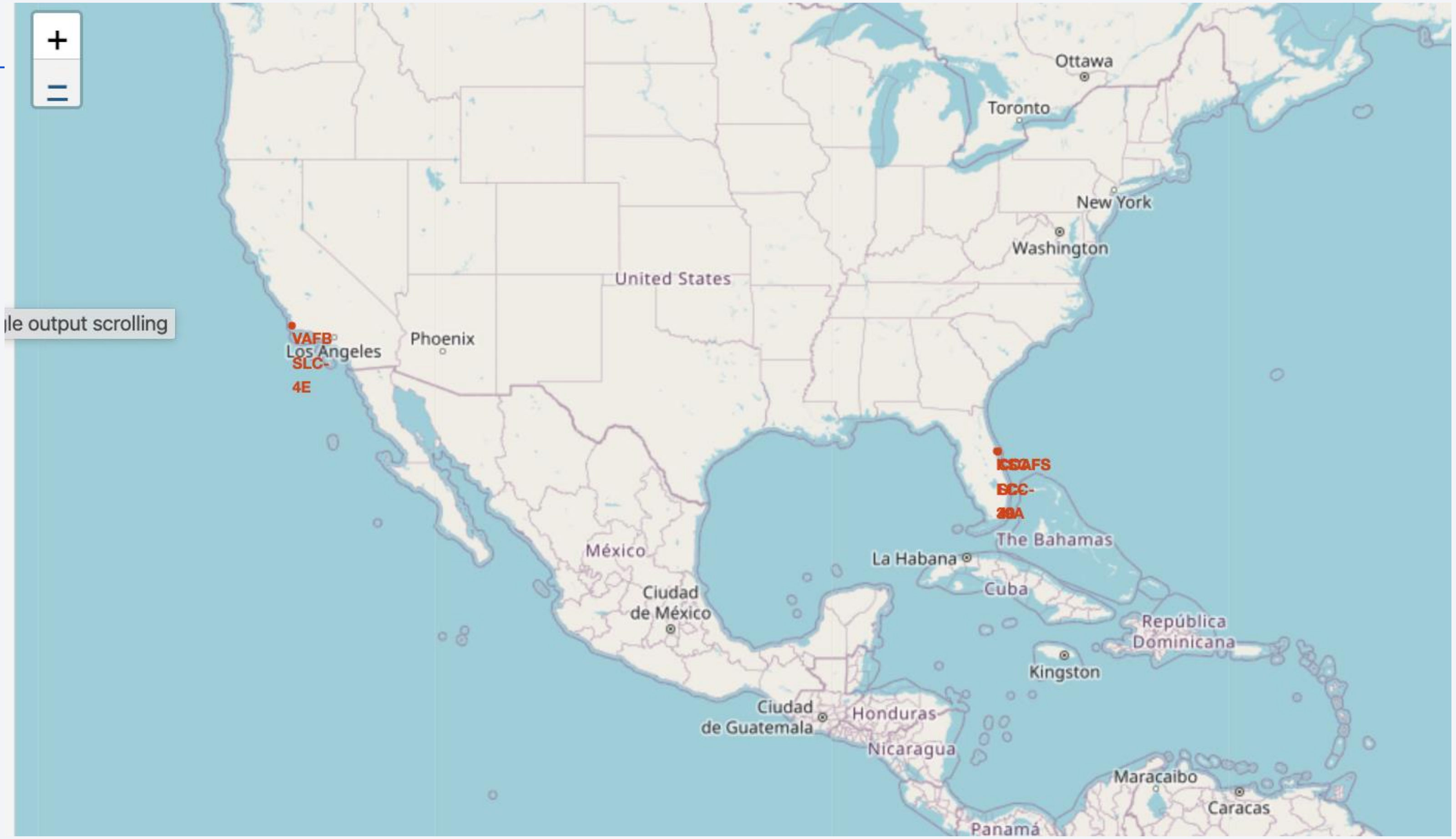
[38]:

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

Section 3

Launch Sites Proximities Analysis

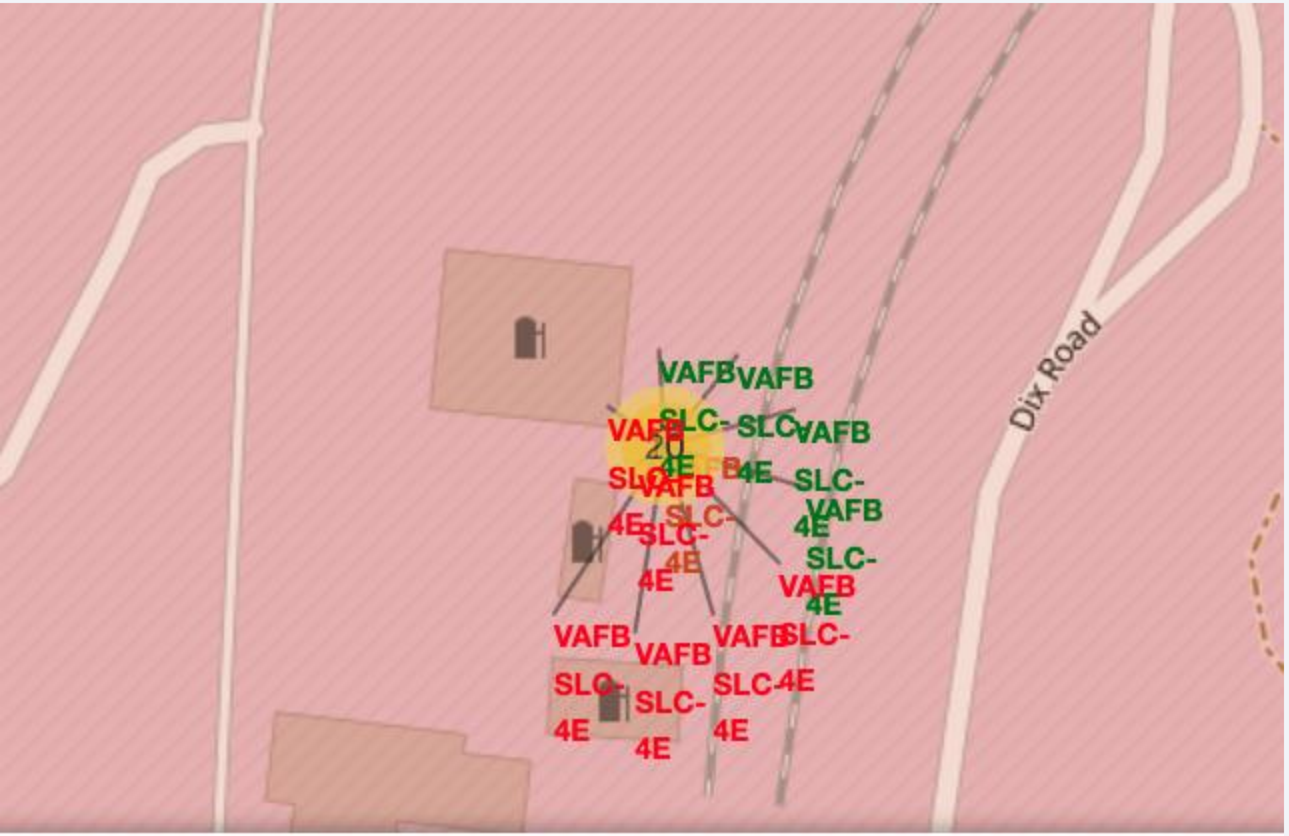
SpaceX Launch Sites



The launch sites are all near the coast and as south as possible

Success and Failure by site

- CCAFS



- VAFB

Proximities to various geographical points

- About 0.9KM to the Coast

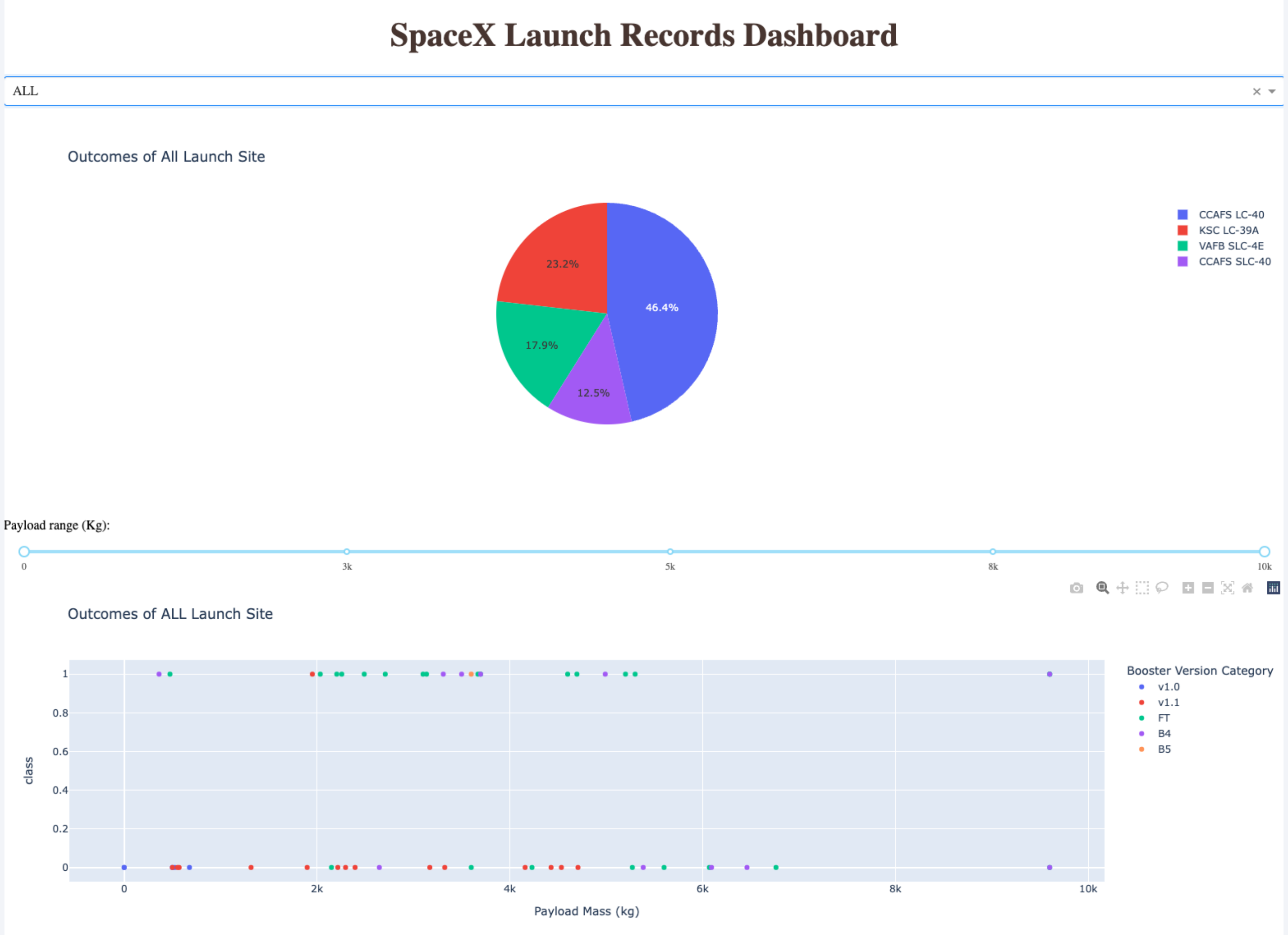
Near transportation in order for better logistics

Section 4

Build a Dashboard with Plotly Dash

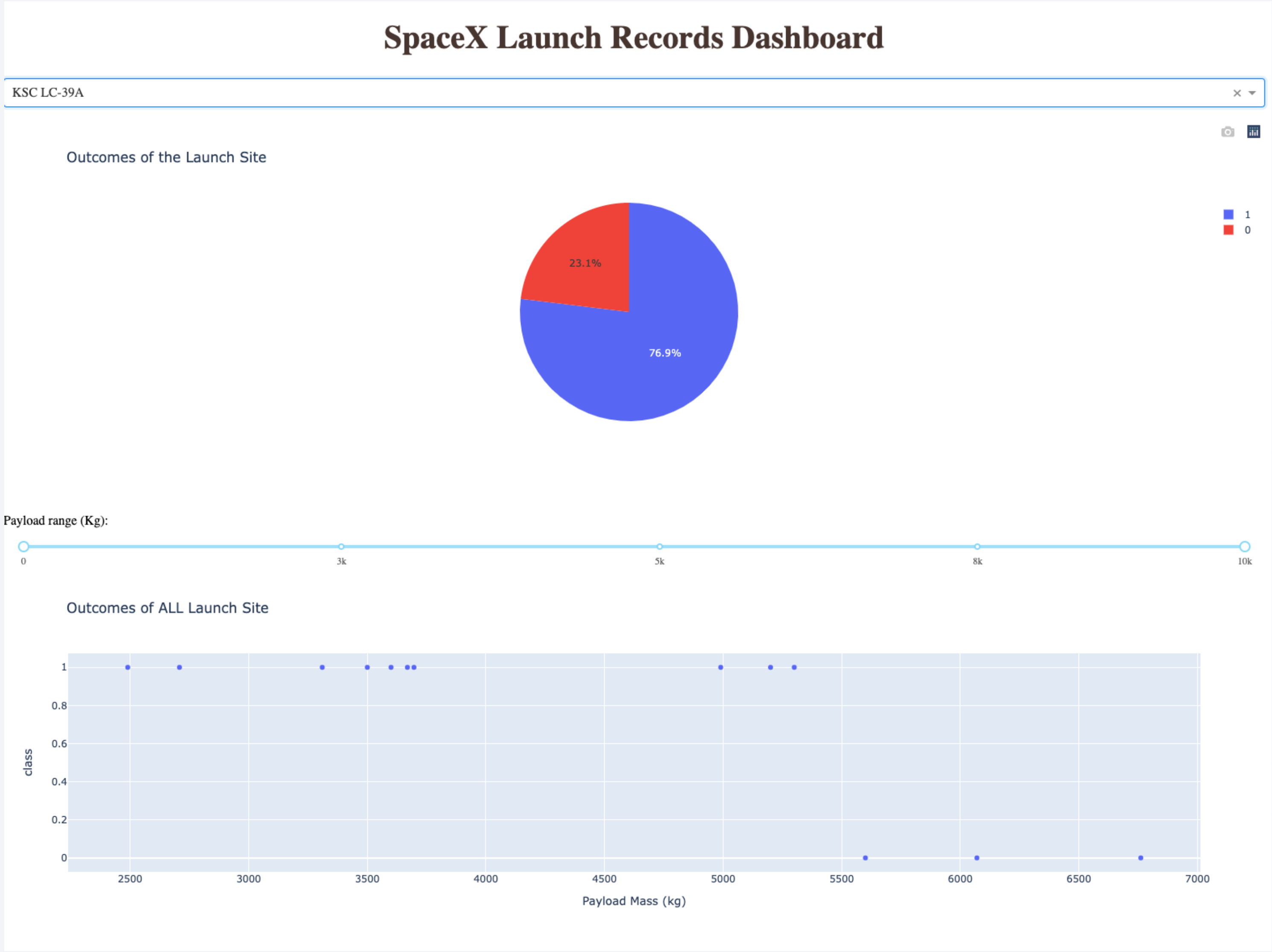
Space X Launch Records Dashboard

Payload Mass, Cass by Booster Version

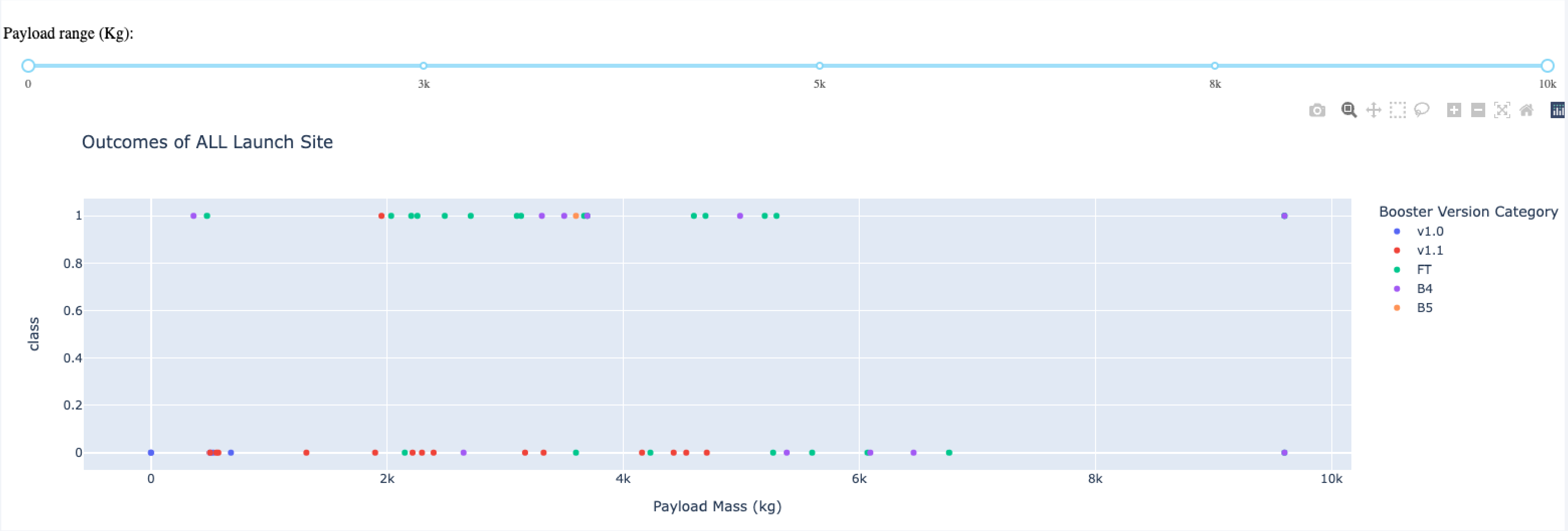


KSC LC-39A Launch site has the highest launch success

- 76.9% success rate
- The failures are at the payloads above 5500 KG



Payload vs Launch Outcomes for all sites



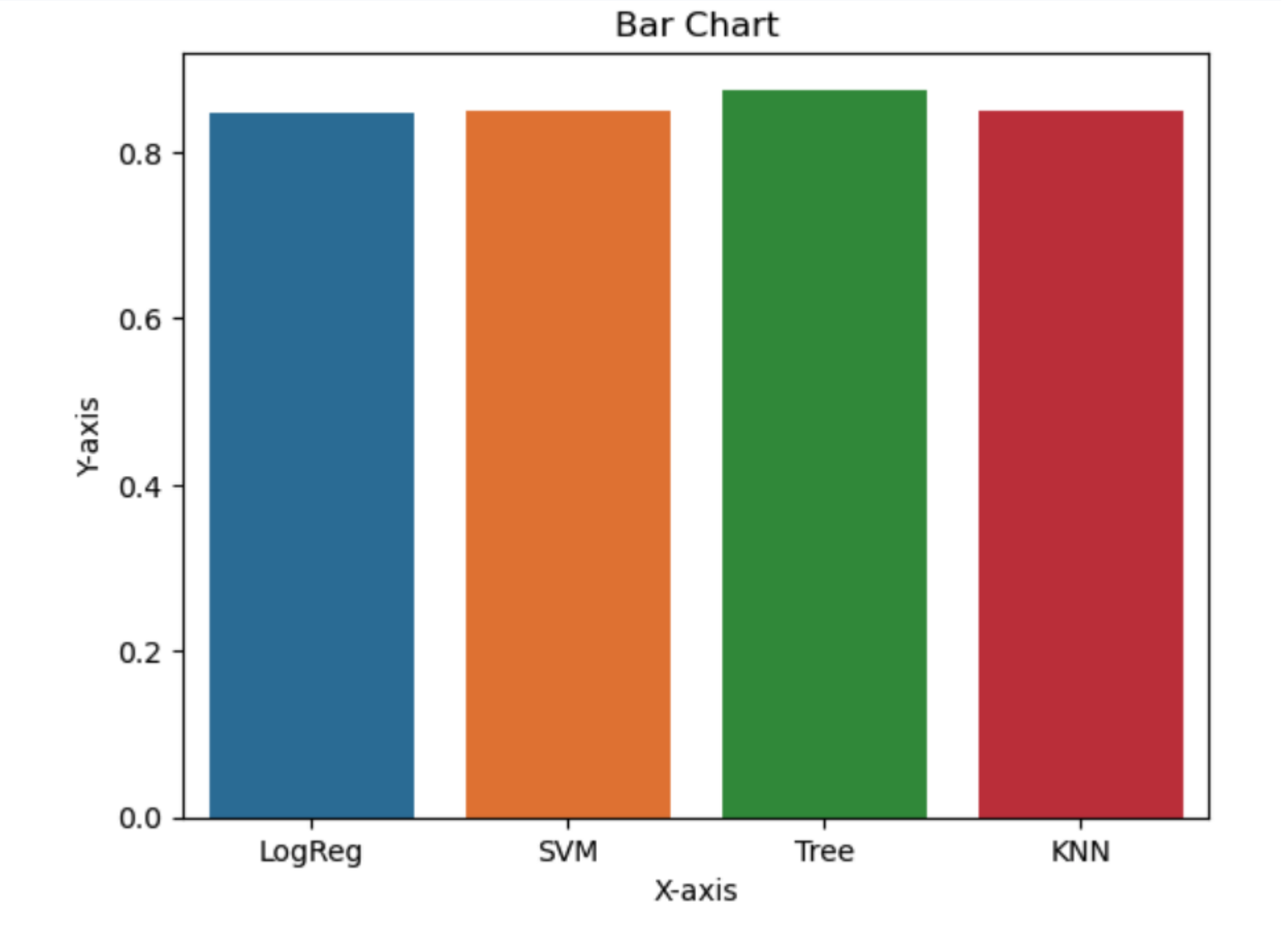
Section 5

Predictive Analysis (Classification)

Classification Accuracy

- Decision Tree Has the highest Accuracy of:

- 0.875



Confusion Matrix

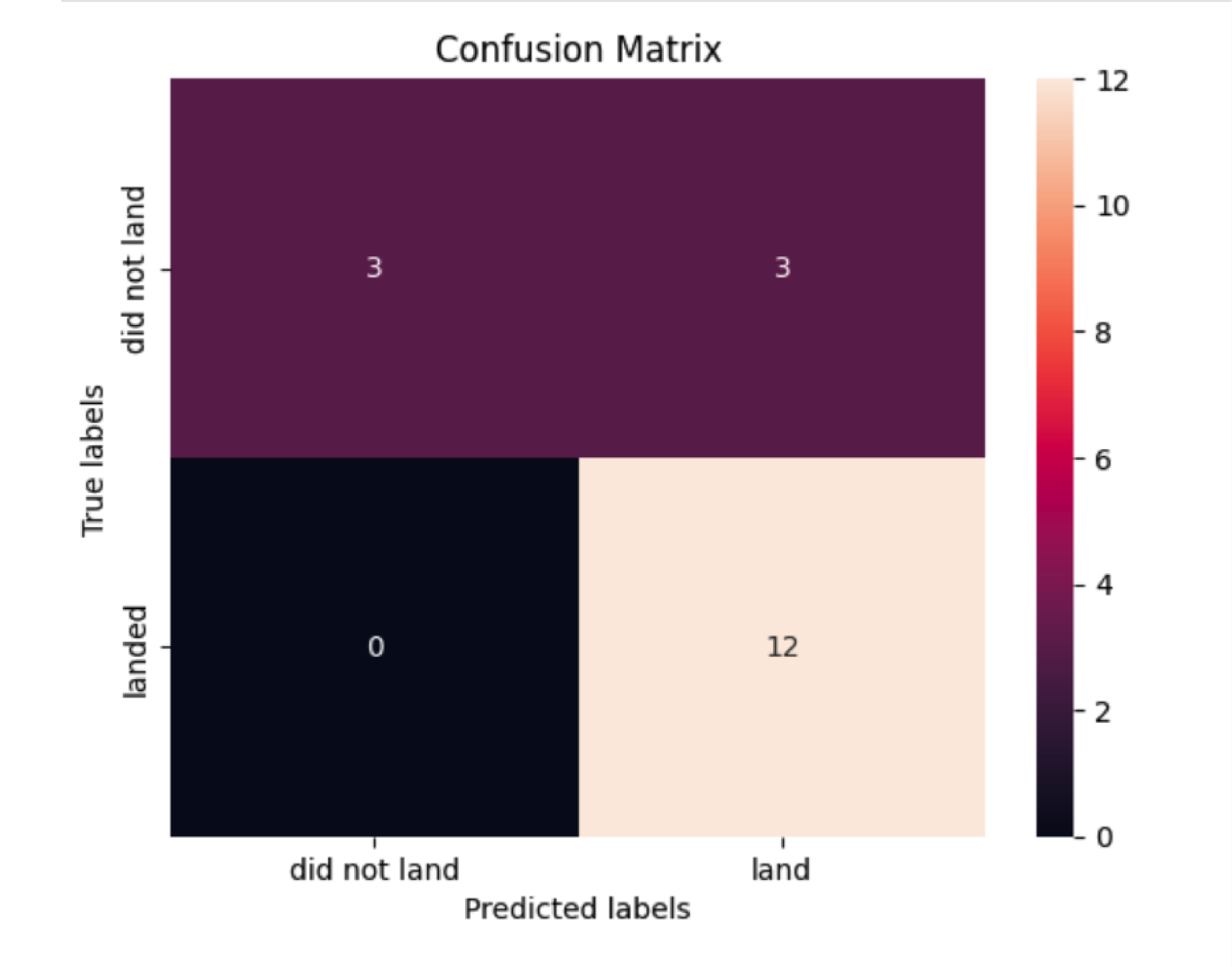
Decision Tree Confusion Matrix:

12 Landed as well as predicted – True Positive

3 Did not Land as predicted

- True Negative

3 – Predicted Land, but really did not – False Positive



Conclusions

- As time progresses, the organization learns and the success rate increases as time progresses
- The launch site: KSC LC 39A is the site with the highest success rate
- Decision Tree Produces the best predictive model of 0.875 accuracy

Appendix

- Include any relevant assets like Python code snippets, SQL queries, charts, Notebook outputs, or data sets that you may have created during this project

Thank you!