IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Panagiotis Stenos
13/10/2022

Outline

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

Executive Summary

- Summary of methodologies
  - Data collection
  - Data wrangling
  - EDA with data visualization
  - EDA with SQL
  - Building an interactive map with Folium
  - Building an interactive map with Plotly Dash
  - Predictive analysis (Classification)
- Summary of all results
  - EDA results
  - Interactive analysis
  - Predictive analysis

Introduction

- Project context
  - Space missions are becoming cheaper and cheaper because the advanced technological framework has allowed for the development of rockets whose first stage is reusable. The ability to reuse the first stage can save more than 100 million dollars and thus has a significant impact on space travel and exploration!
  - SpaceX advertises that Falcon 9 rocket launches will cost only 62 million dollars as its first stage can be reused
- Aim
  - The aim of the project is to find the to what extent (the probability) the first stage of rockets can be recovered in future launches using exploratory data analysis on historical space missions.

Section 1

Methodology

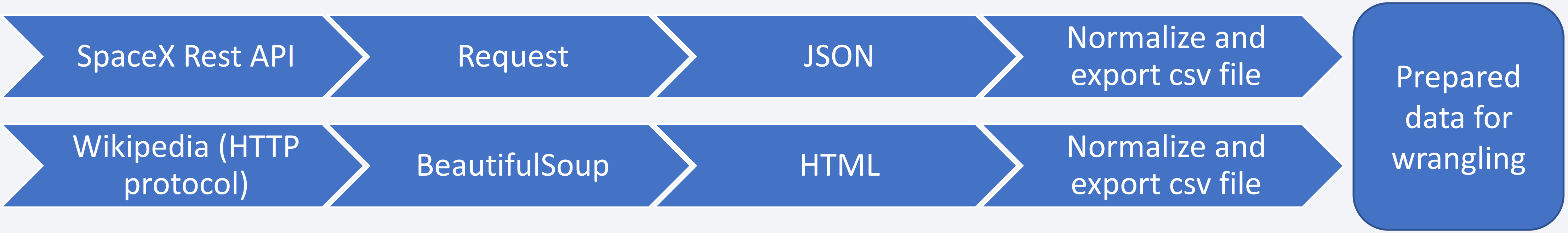
Methodology

Executive Summary

- Data collection methodology:
  - Request to the SpaceX Rest API
  - Web scraping Heavy Launches Records from Wikipedia
- Perform data wrangling
  - Data Cleaning and Preprocessing using Pandas
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Available dataset was used to train, validate and evaluate the models
  - Best classifier was determined and was further evaluated using the available dataset

Data Collection

- Data for SpaceX missions were obtained through the SpaceX Rest API and by Web scrapping heavy launches records from Wikipedia
- Rest API contained information like the date, launch site name, outcome and location of the launch as well as features like the version and payload mass of the rocket.
- Another source for collecting Falcon 9 data was Wikipedia. Data was extracted through Wikipedia using web scraping



Data Collection – SpaceX API

GitHub Link:

https://github.com/panstenos/IBM-CAPSTONE/blob/main/IBM%20M10W1%20Data%20Collection%20API.ipynb

Use requests.get method to communicate with the API

```
spacex_url="https://api.spacexdata.com/v4/launches/past"
response = requests.get(spacex_url)
```

Use json_normalize to convert response.json to DataFrame

```
response = requests.get(static_json_url)
data = pd.json_normalize(response.json())
```

Clean the Data

```
getBoosterVersion(data)
getLaunchSite(data)
getPayloadData(data)
getCoreData(data)
```

Combine the columns into a 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}
```

Manipulate the DataFrame and export it as a csv file

```
data_falcon9 = data_falcon9[data_falcon9['BoosterVersion']=='Falcon 9']
data_falcon9.to_csv('dataset_part\_1.csv', index=False)
```

Data Collection - Scraping

GitHub Link:

https://github.com/panstenos/IBM-CAPSTONE/blob/main/IBM%20M1OW1%20Data%20Collection%20Web%20Scraping.ipynb

Use requests and BeautifulSoup to parse the webpage

```
data = requests.get(static_url).text
soup = BeautifulSoup(data, 'html5lib')
```

Find all the tables

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

Get the column names in the interested table

```
column_names = []
temp = soup.find_all('th')
for x in range(len(temp)):
    try:
        name = extract_column_from_header(temp[x])
        if (name is not None and len(name)>0):
            column_names.append(name)
    except:
        pass
```

Combine the columns into a Dictionary

```
launch_dict= dict.fromkeys(column_names)
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'] = []
launch_dict['Version Booster']=[]
launch_dict['Booster landing']=[]
launch_dict['Date']=[]
launch_dict['Time']=[]
```

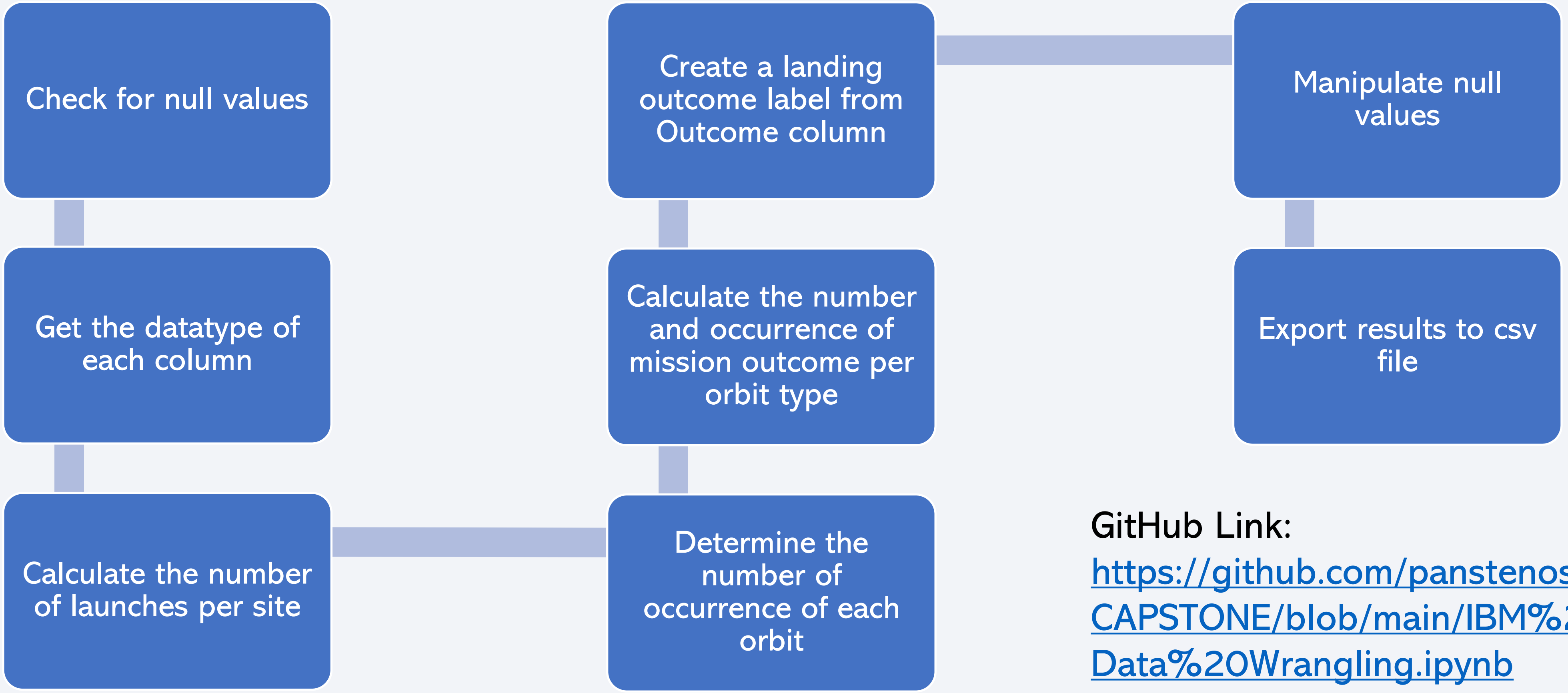
Append the data to the keys

```
xtracted_row = 0
Extract each table
or table_number,table in enumerate(soup.find_all('table',"wikitable plainrowheaders collapsible")):
  # get table row
   for rows in table.find_all("tr"):
       #check to see if first table heading is as number corresponding to launch a number
       if rows.th:
           if rows.th.string:
               flight_number=rows.th.string.strip()
               flag=flight_number.isdigit()
       else:
           flag=False
```

Obtain DataFrame and export it as a csv file

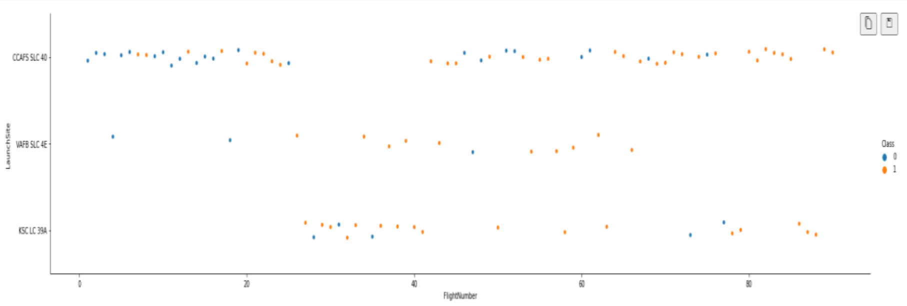
```
df=pd.DataFrame(launch_dict)
df.to_csv('spacex_web_scraped.csv', index=False)
```

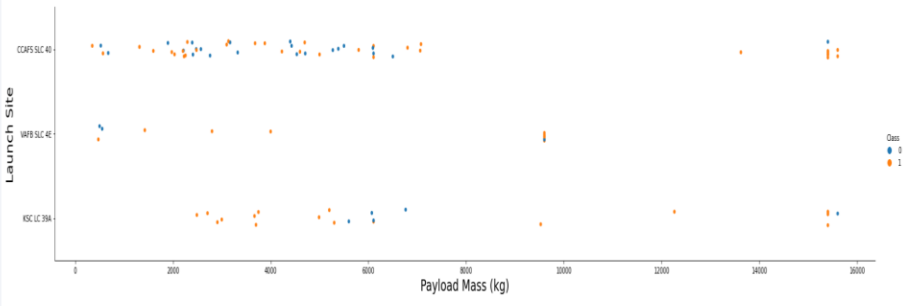
Data Wrangling

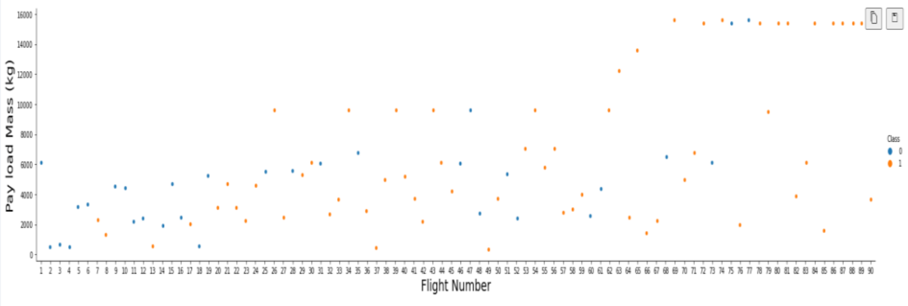


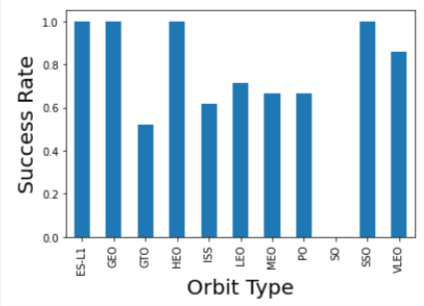
GitHub Link: https://github.com/panstenos/IBM-CAPSTONE/blob/main/IBM%20M10W1%20Data%20Wrangling.ipynb

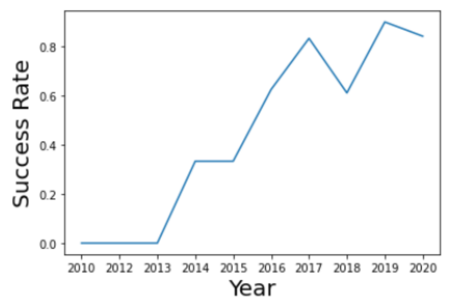
EDA with Data Visualization











GitHub Link: https://github.com/panstenos/IBM-CAPSTONE/blob/main/IBM%20M10W2%20EDA%20with%20Visualization.ipynb

EDA with SQL

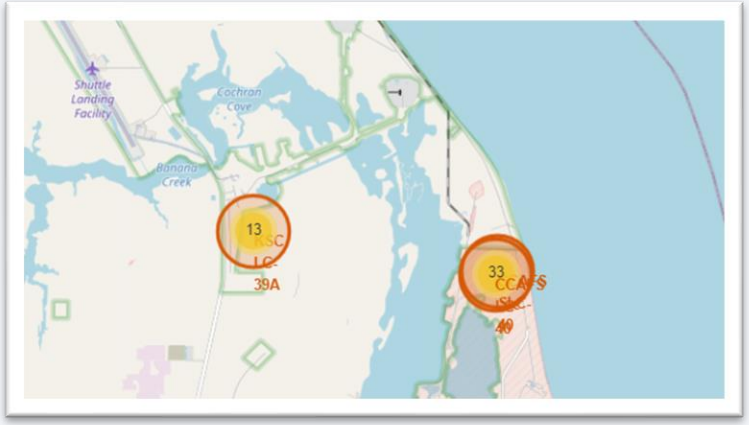
- SQL queries performed:
  - Display the names of the unique launch sites in the space mission
  - Display 5 records where launch sites begin with the string ‘CCA’
  - Display the total payload mass carried by boosters launched by NASA (CRS)
  - Display average payload mass carried by booster version 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. Use a subquery
  - List the records which will display the month names, failure landing_outcomes in drone ship ,booster versions, launch_site for the months in year 2015
  - Rank the count of successful landing_outcomes between the date 04-06-2010 and 20-03-2017 in descending order

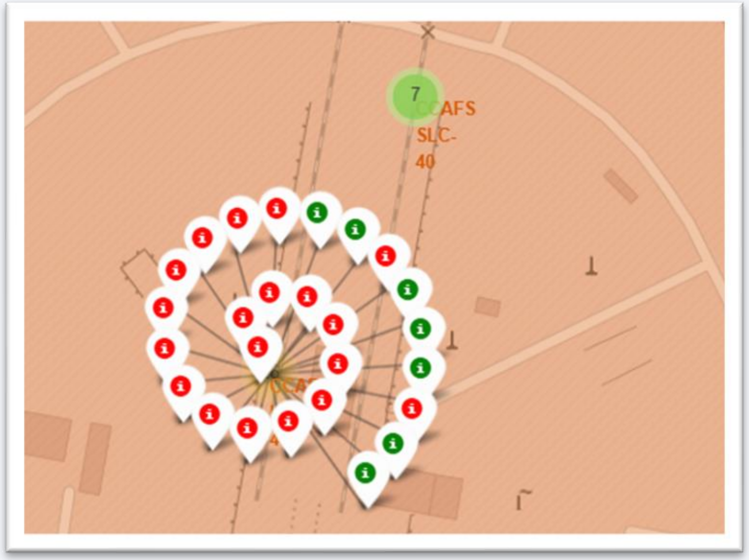
GitHub Link: https://github.com/panstenos/IBM-CAPSTONE/blob/main/IBM%20M10W2%20EDA%20with%20SQL.ipynb

Build an Interactive Map with Folium

- Markers used to build Folium Maps
  - Circle marker >> locate the coordinates
  - Text label >> name the locations
  - Marker clusters >> to simplify the map containing many markers
  - Color labeled markers >> reflected the outcome of the mission

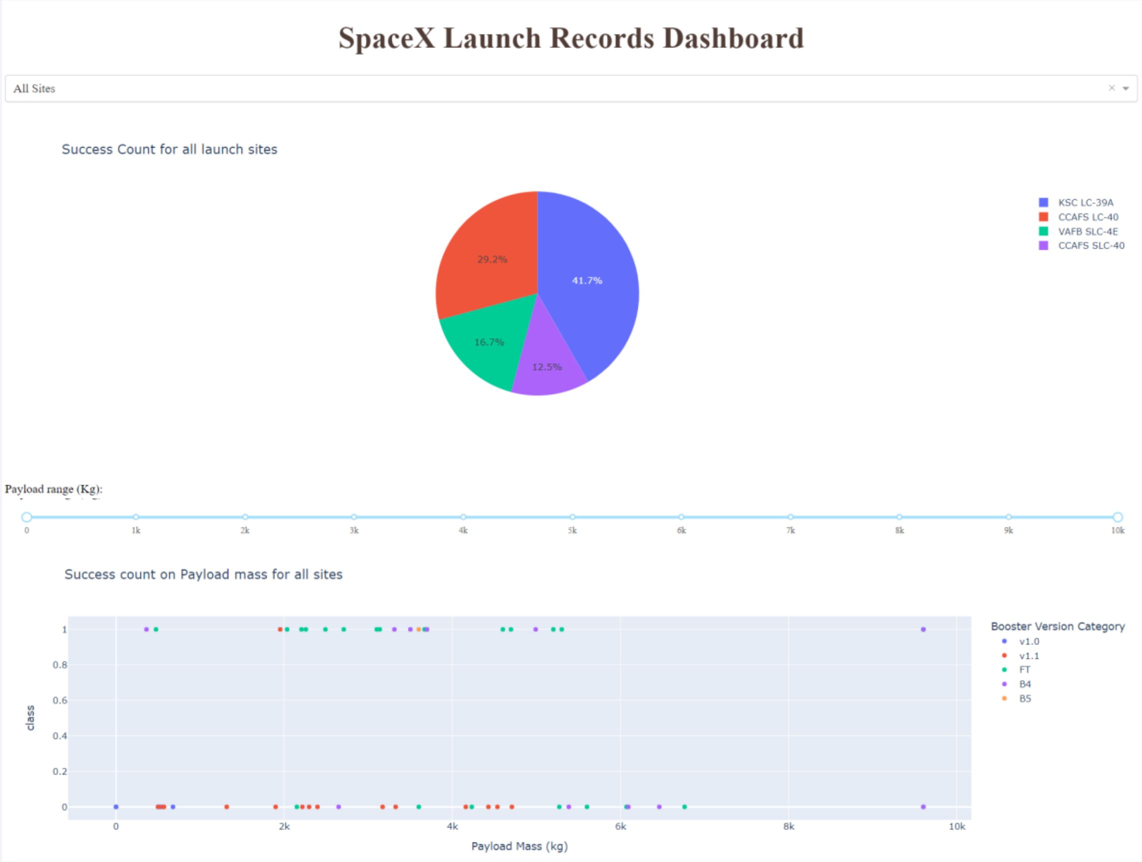
GitHub Url: https://github.com/panstenos/IBM-CAPSTONE/blob/main/IBM%20M10W2%20EDA%20with%20Visualization.ipynb





Build a Dashboard with Plotly Dash

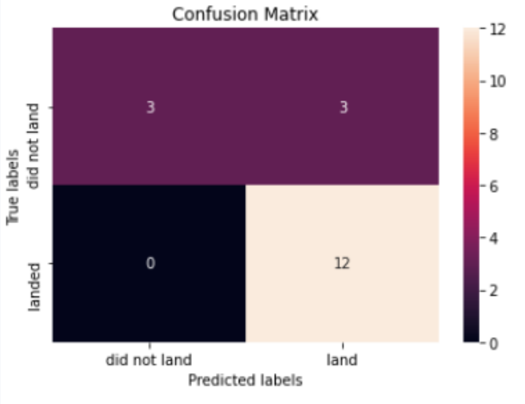
- Pie chart containing successful launches grouped by launch site
- Scatter plot of Payload Mass vs outcome of launches colored by Booster Version category

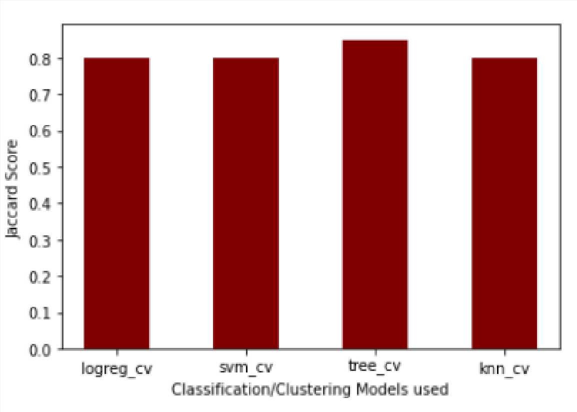


GitHub Url: https://github.com/panstenos/IBM-CAPSTONE/blob/main/IBM%20M10W3%20Dash.ipynb

Predictive Analysis (Classification)

- Machine Learning models were used to predict the outcome of future missions
- SVM, KNN, Logistic Regression and Tree models were built for that purpose and were ~83% accurate





GitHub URL: https://github.com/panstenos/IBM-CAPSTONE/blob/main/IBM%20M10W4%20Machine%20Learning%20Prediction.ipynb

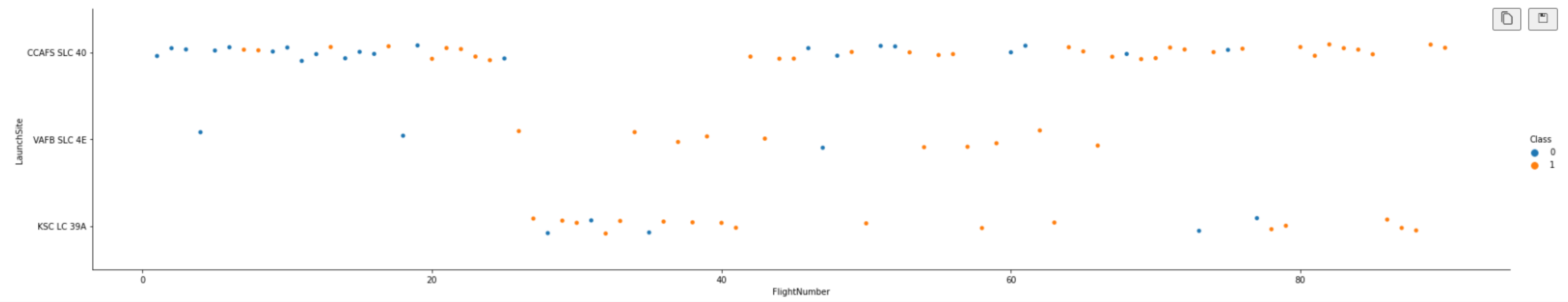
Results

- Heavy load mass rockets were found more likely to be retrieved that lighter rockets
- KSC LC 39A had the most successful number of launches between all sites
- Orbits GEO, HEO, SSO and ES-L1 have the best success rate
- The success rate of retrievable 1st stage rocket launches increases over time and is expected to exceed 85% in the following years

Section 2

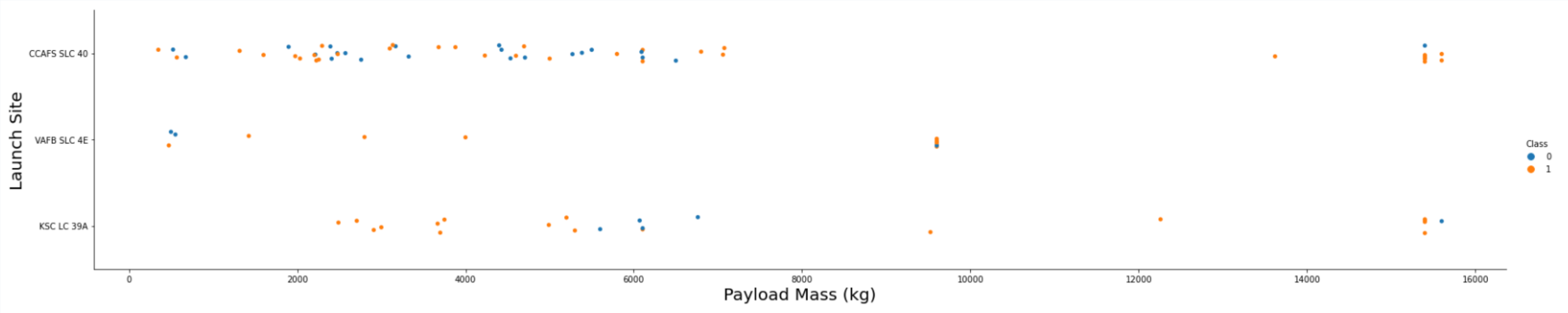
Insights drawn from EDA

Flight Number vs. Launch Site



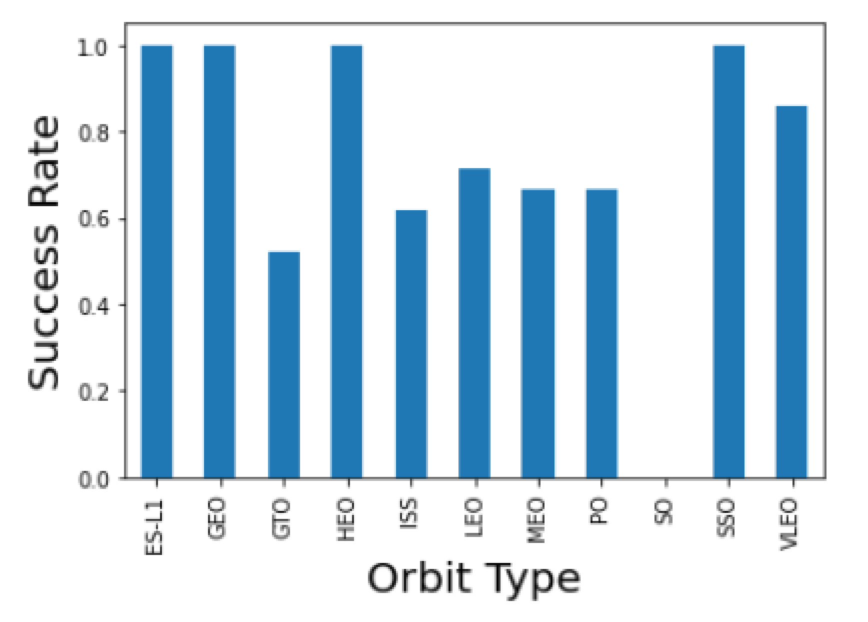
- Scatter plot of Flight number against Launch site
- As the scatter plot indicates, the Launch site with the best success rate was KSC LC 39A

Payload vs. Launch Site



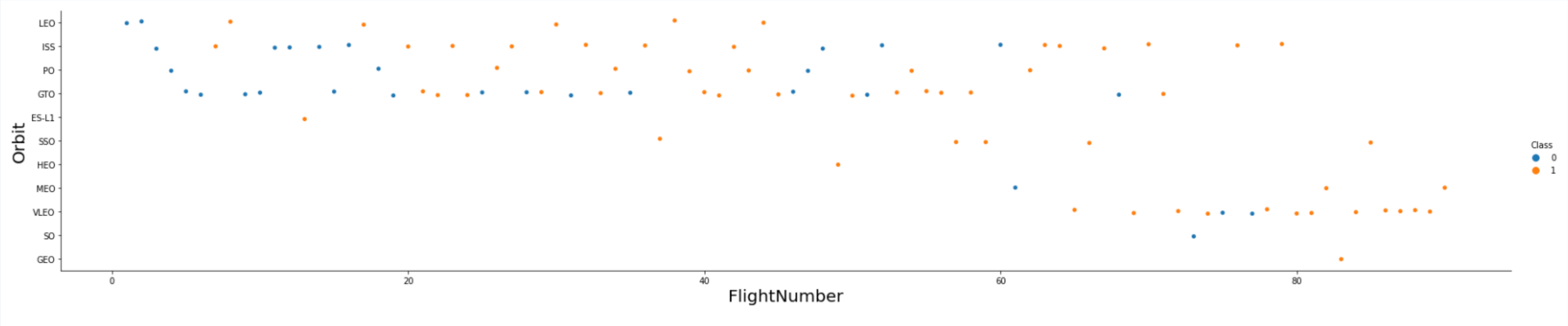
- Scatter plot of Payload mass (kg) against Launch site
- As the scatter plot indicates, heavier rockets' first stage are more likely to be recovered than the ones of lighter rockets

Success Rate vs. Orbit Type



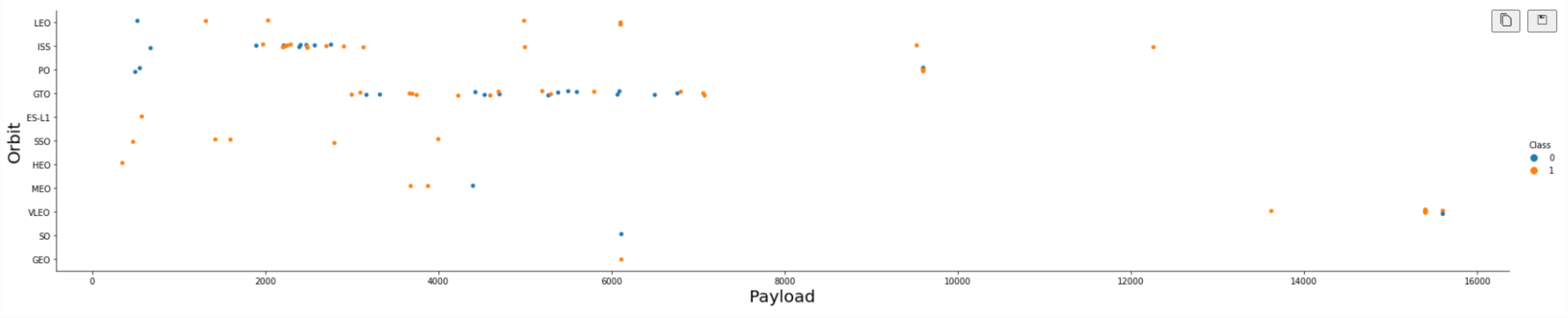
- Bar plot of Orbit type against Success rate
- Orbits GEO, HEO, SSO and ES-L1 have the best Success rate

Flight Number vs. Orbit Type



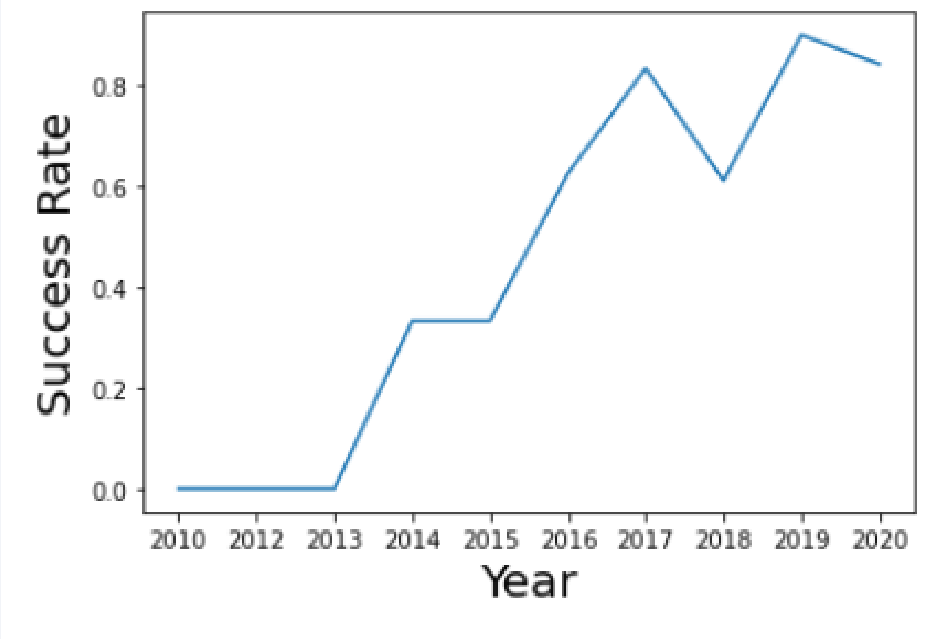
- Scatter plot of Flight number against Orbit

Payload vs. Orbit Type



- Scatter plot of Payload against Orbit

Launch Success Yearly Trend



- Line plot of Year vs success rate
- As shown, yearly success rate is expected to increase as years pass

All Launch Site Names

```
%%sql
SELECT launch_site, COUNT(launch_site) AS "COUNT" FROM SPACEXTBL GROUP BY launch_site;
```

```
 * ibm_db_sa://jdf13642:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.data
Done.
```

launch_site	COUNT
CCAFS LC-40	26
CCAFS SLC-40	34
KSC LC-39A	25
VAFB SLC-4E	16

Launch Site Names Begin with 'CCA'

```
%%sql
SELECT * FROM SPACEXTBL WHERE launch_site LIKE '%CCA%' LIMIT 5;
```

```
 * ibm_db_sa://jdf13642:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/BLUDB
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 Qualificatio	0	LEO	SpaceX	Success	Failure (parachute)
08-12-2010	15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight C1, two Cub	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

```
%%sql

SELECT SUM(payload_mass__kg_) AS "TOTAL MASS (kg)" FROM SPACEXTBL
WHERE customer LIKE 'NASA (CRS)';
```

```
 * ibm_db_sa://jdf13642:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2i
Done.
```

TOTAL MASS (kg)
45596

Average Payload Mass by F9 v1.1

```
%%sql

SELECT TRUNCATE(AVG(payload_mass__kg_),0) AS "AVERAGE PAYLOAD MASS (kg)" FROM SPACEXTBL
WHERE booster_version LIKE '%F9 v1.1%'
```

```
 * ibm_db_sa://jdf13642:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.datab
Done.
```

AVERAGE PAYLOAD MASS (kg)
2534

First Successful Ground Landing Date

```
%%sql

SELECT MIN(DATE) FROM SPACEXTBL WHERE landing__outcome LIKE 'Success';
```

```
 * ibm_db_sa://jdf13642:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l0
Done.
```

1
22-12-2015

Successful Drone Ship Landing with Payload between 4000 and 6000

```
%%sql

SELECT booster_version FROM SPACEXTBL
WHERE payload_mass__kg_ BETWEEN 4000 AND 6000
AND landing__outcome = 'Success (drone ship)'
GROUP BY booster_version;
```

✓ 0.8s

```
 * ibm_db_sa://jdf13642:***@0c77d6f2-5da9-48a9-81f
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

```
%%sql
SELECT COUNT(mission_outcome) AS "NUMBER OF SUCCESSFUL AND FAILURE MISSION OUTCOMES" FROM SPACEXTBL
WHERE mission_outcome LIKE '%Success%'
OR mission_outcome = 'Failure (in flight)'
```

0.6s

```
 * ibm_db_sa://jdf13642:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdoma:
Done.
```

NUMBER OF SUCCESSFUL AND FAILURE MISSION OUTCOMES
101

Boosters Carried Maximum Payload

```
%%sql
SELECT DISTINCT BOOSTER_VERSION FROM SPACEXTBL
WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTBL);
```

✓ 0.7s

```
 * ibm_db_sa://jdf13642:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kc
Done.
```

booster_version
F9 B4 B1041.2
F9 B4 B1041.1
F9 B5 B1049.2
F9 B5B1048.1
F9 FT B1036.2
F9 FT B1029.1
F9 FT B1036.1

2015 Launch Records

```
%%sql
SELECT LANDING__OUTCOME, BOOSTER_VERSION, LAUNCH_SITE
FROM SPACEXTBL
WHERE landing__outcome = 'Failure (drone ship)'
    AND DATE LIKE '%2015%';
```

✓ 0.8s

```
 * ibm_db_sa://jdf13642:***@0c77d6f2-5da9-48a9-81f8-86b520
Done.
```

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

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

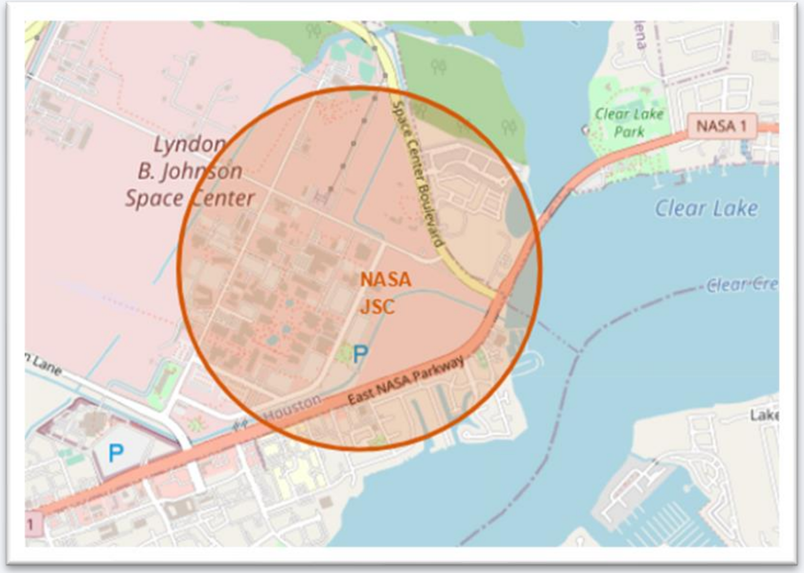
```
%%sql
SELECT landing__outcome, COUNT(landing__outcome) AS "TOTAL_NUMBER"
FROM SPACEXTBL
WHERE DATE BETWEEN '2010-06-04' AND '2017-03-20'
GROUP BY landing__outcome
ORDER BY TOTAL_NUMBER DESC
```

time__utc_	booster_version	launch_site	payload	payload_mass__kg_	orbit	customer	mission_outcome	landing__outcome
14:39:00	F9 FT B1031.1	KSC LC-39A	SpaceX CRS-10	2490	LEO (ISS)	NASA (CRS)	Success	Success (ground pad)
17:54:00	F9 FT B1029.1	VAFB SLC-4E	Iridium NEXT 1	9600	Polar LEO	Iridium Communications	Success	Success (drone ship)
05:26:00	F9 FT B1026	CCAFS LC-40	JCSAT-16	4600	GTO	SKY Perfect JSAT Group	Success	Success (drone ship)
04:45:00	F9 FT B1025.1	CCAFS LC-40	SpaceX CRS-9	2257	LEO (ISS)	NASA (CRS)	Success	Success (ground pad)
[illegible]	[illegible]	CCAFS LC-	Thaicom 8	3100	GTO	Thaicom	Success	Success (drone ship)

Section 3

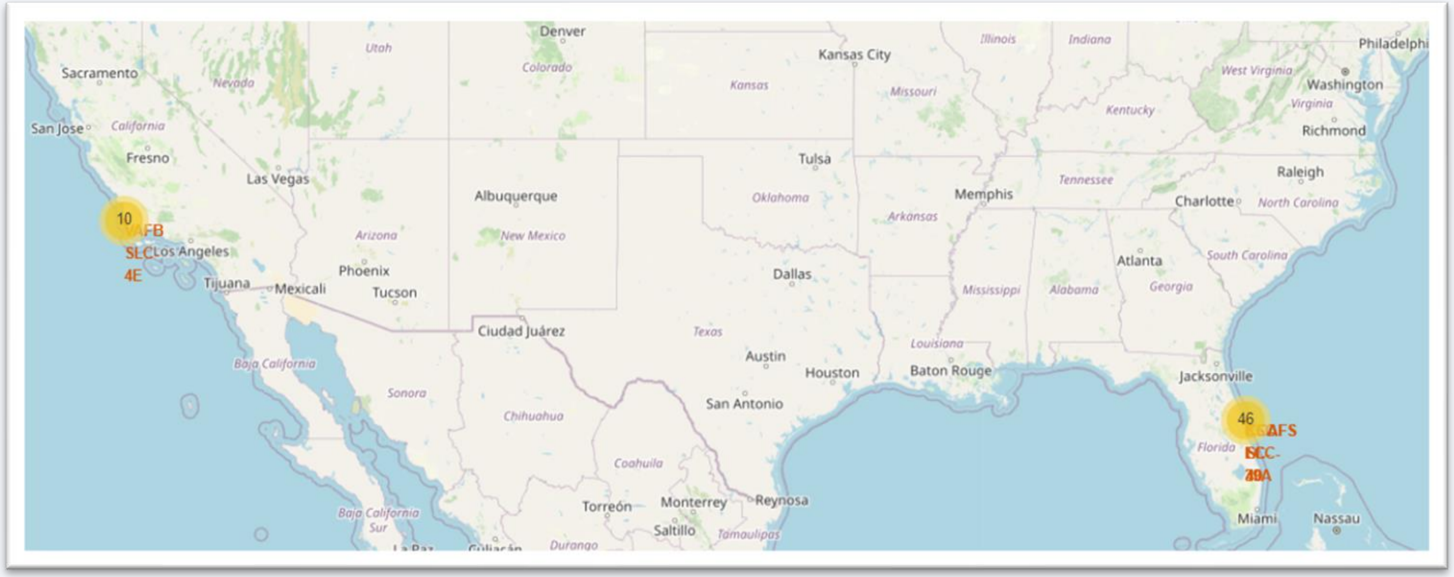
Launch Sites Proximities Analysis

Launch Sites

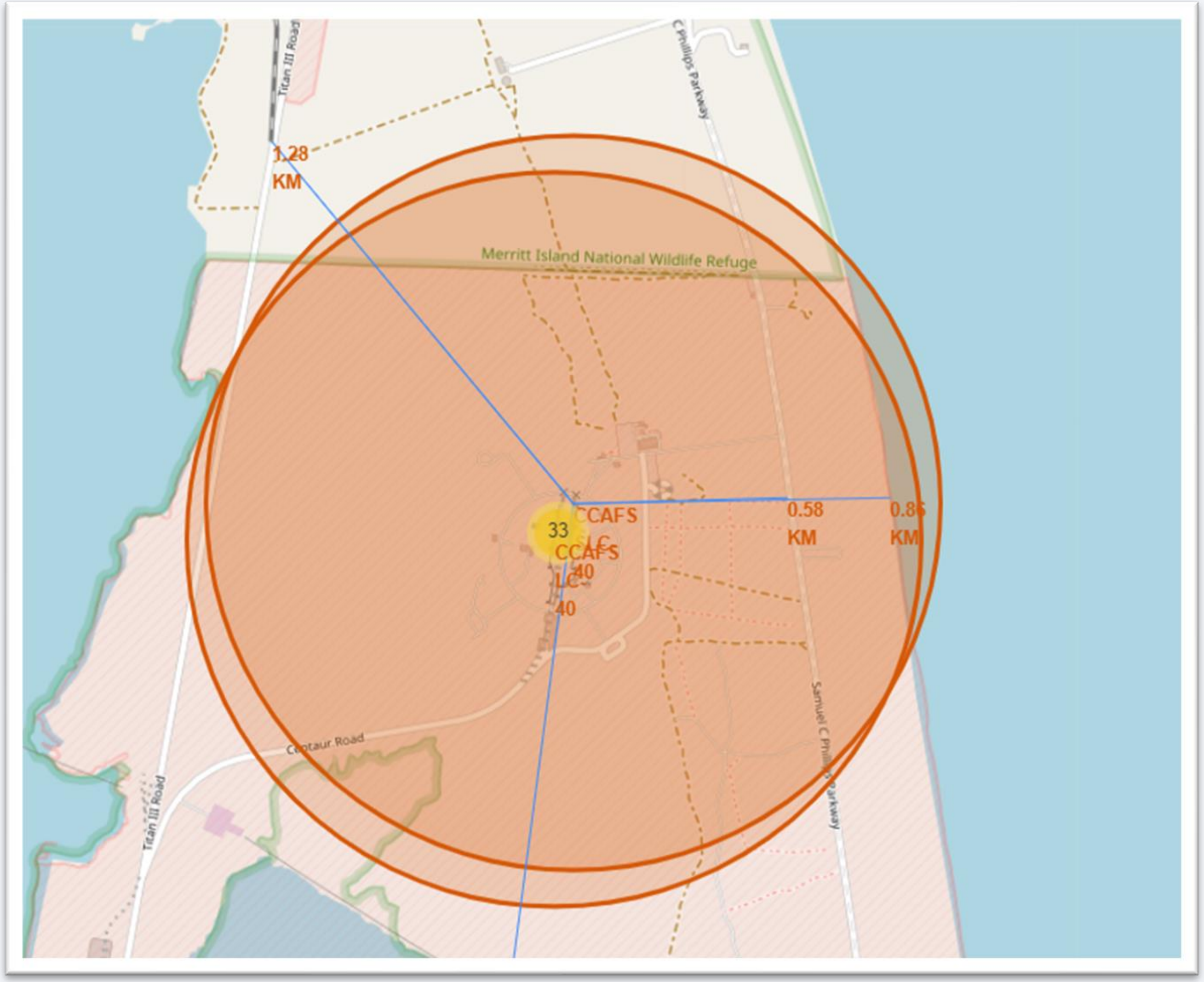


We can see that all the launch sites are to a very close proximity to the sea

Successful/Failed launches per launch site



Distances between launch sites and its proximities

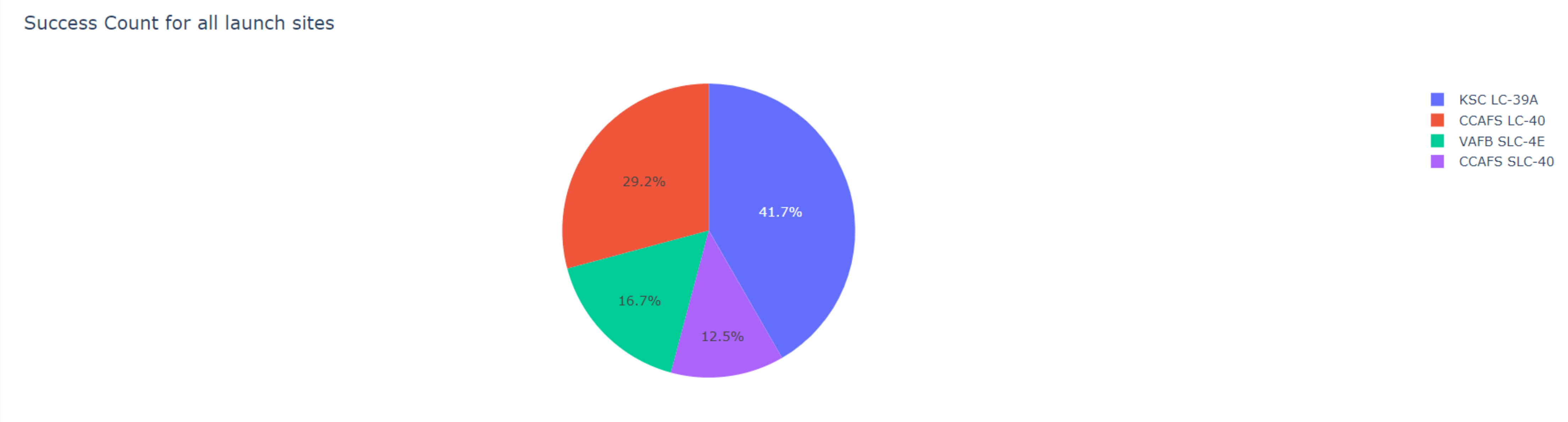


It is observed that launch sites are close to railways and highways and distant from cities where the population is dense

Section 4

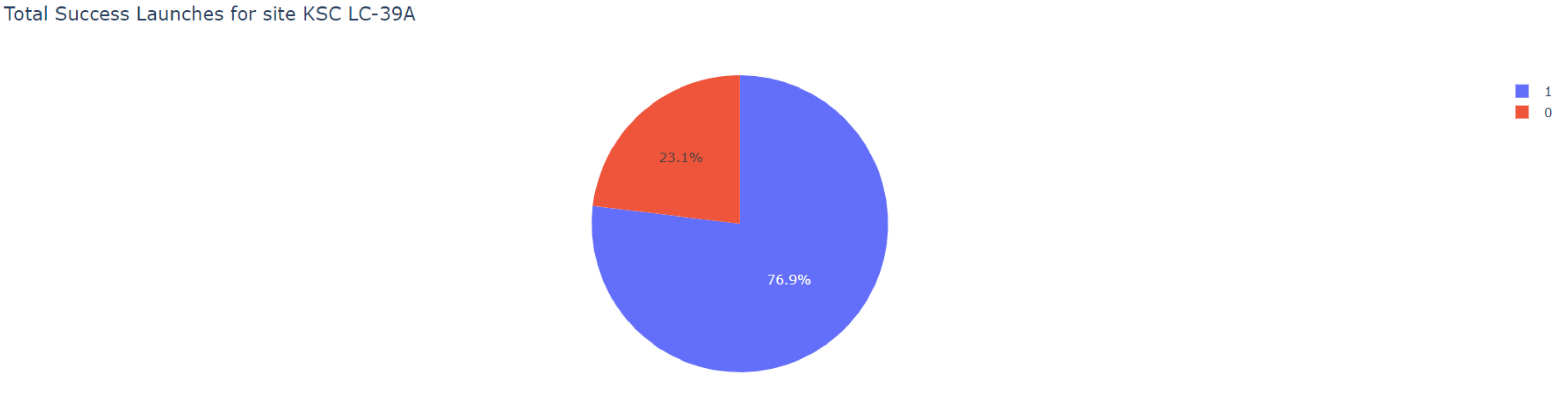
Build a Dashboard with Plotly Dash

Net success of launches by each site



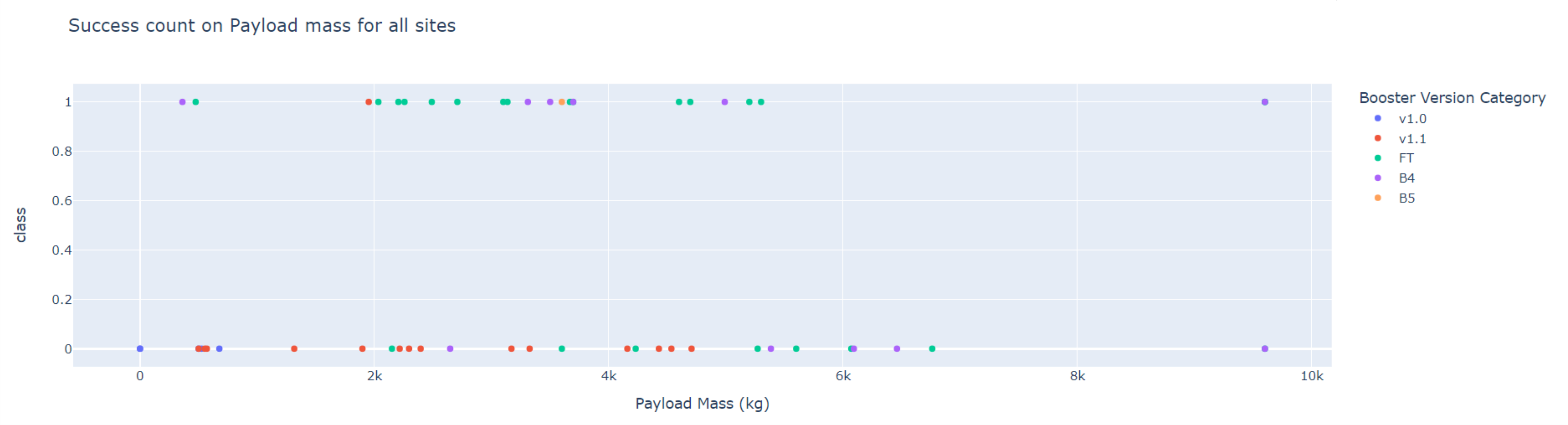
KSC LC-39A is the launch site with the most successful launches

Success rate by site



KSC LC-39A is the launch site with 76.9% success rate

Payload vs launch outcome

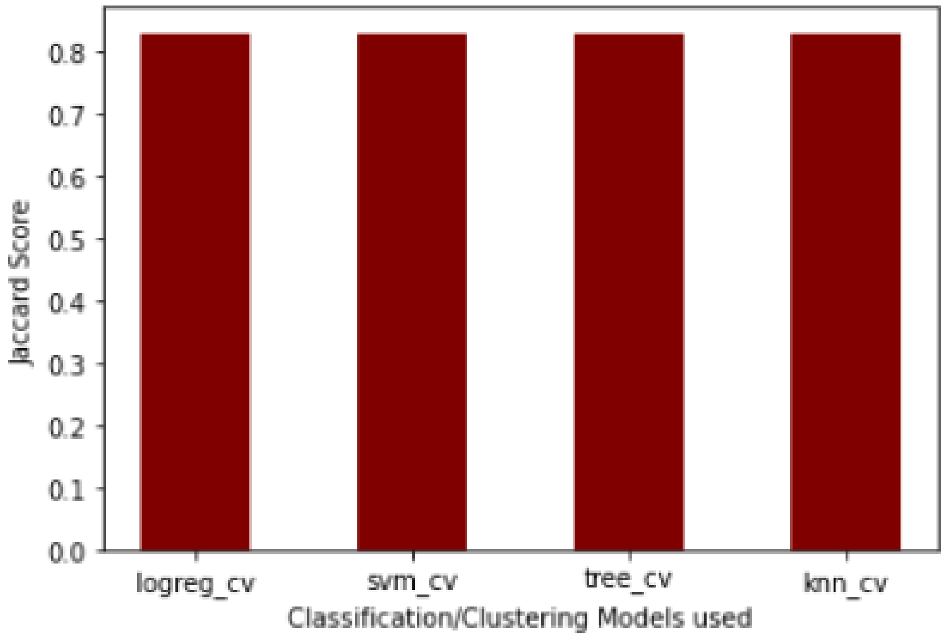


Section 5

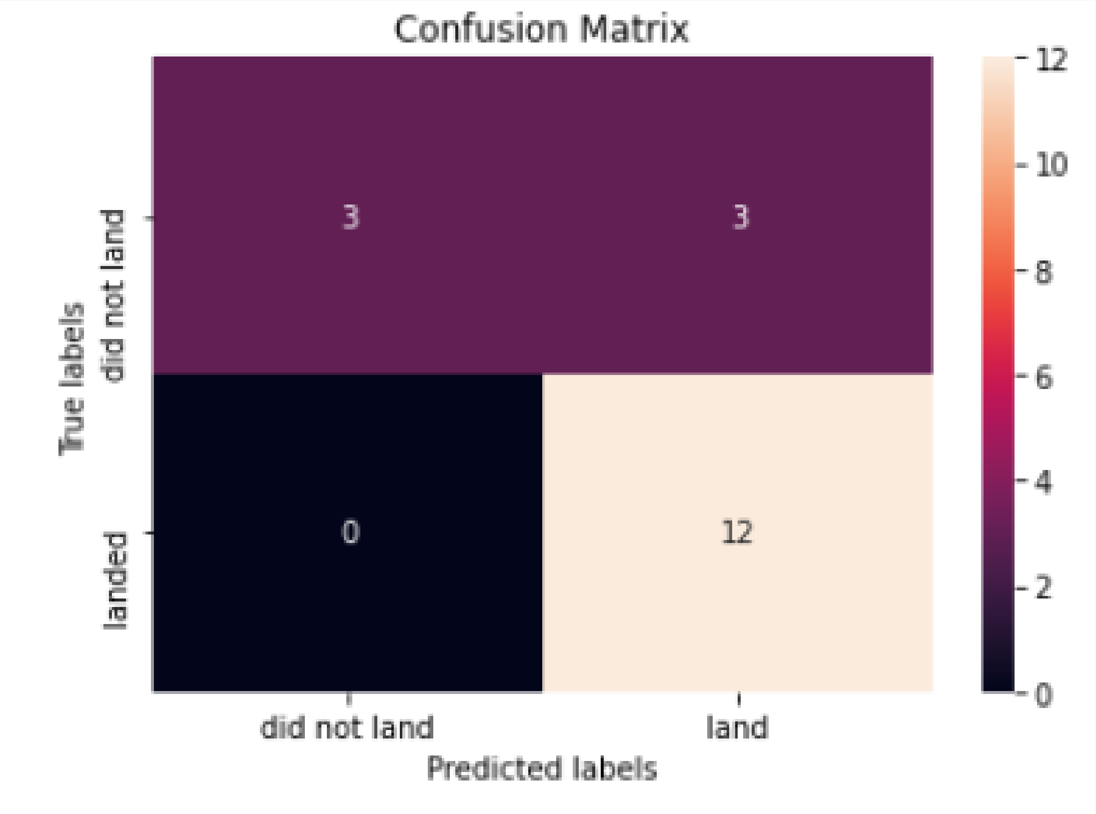
Predictive Analysis (Classification)

Classification Accuracy

```
logreg_cv accuracy score -> 0.83
svm_cv accuracy score -> 0.83
tree_cv accuracy score -> 0.83
knn_cv accuracy score -> 0.83
```



Confusion Matrix



Only 3 labels were mis predicted as ‘land’ where the actual outcome was ‘did not land’
This is the case of a false positive

Conclusions

- All models were very good in terms of prediction accuracy for this data set
- Heavy weighted payloads seemed to perform better that lighter ones (in terms of success rate)
- KSC LC 39A had the most successful number of launches between all sites with 76.9% success rate
- Orbits GEO, HEO, SSO and ES-L1 have the best success rate of all orbits

Thank you!