WINNING SPACE RACE WITH DATA SCIENCE

Ehab Elgamal
April 2023

Outline

- Executive Summary
- Introduction
- Methodology
- Results
- Conclusions

EXECUTIVE SUMMARY

- **Methodology**
  - Data collection
  - Data wrangling
  - EDA with data visualization
  - EDA with SQL
  - Building an interactive map with Folium
  - Building a dashboard with Plotly Dash
  - Predictive analysis (Classification)

- **Results**
  - Insights from EDA with Visualization
  - Insights from EDA with SQL
  - Insights from Launch Sites Proximities Analysis
  - Insights from Dashboard
  - Results of Predictive Analysis

INTRODUCTION

- SPACE X 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.

- Prediction if the first stage would land allows the cost of a launch could be determined.

- The task of this project was to predict if the FALCON 9 first stage would land successfully.

METHODOLOGY

Data Collection

- SPACE X launch data was collected from SPACE X REST-API that revealed data about launches, rocket used, payload delivered, launch specifications, landing specifications & landing outcome.

- SPACE X REST-API endpoint used was that started with

  api.spacexdata.com/v4/launches/past

- Python BeautifulSoup package was used to webscrape some HTML tables that contained valuable FALCON 9 launch records in the related WIKI pages.

SPACE X REST-API

1. Requesting Response from API:

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

2. Converting Response to .json File:

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

3. Applying custom functions to extract information from launch data:

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

4. Construction of the dataset:

```
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}
```

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

5. Filtering & exporting to csv file:

```
data_falcon9 = df[df['BoosterVersion'] != 'Falcon 1']
data_falcon9.to_csv('dataset_part_1.csv', index=False)
```

https://github.com/Bob-Gemihood/IBM_capstone/blob/main/01.capstone_data_collection.ipynb

Webscraping

1. Requesting FALCON 9 launch wiki page from its URL:

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

2. Creating BeautifulSoup object:

```
launch_soup = BeautifulSoup(launch_data, "html.parser")
```

3. Finding tables:

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

first_launch_table = html_tables[2]
print(first_launch_table)
```

4. Extracting column names:

```
headers = first_launch_table.find_all('th')
for header in headers:
    name = extract_column_from_header(header)
    if name is not None and len(name) > 0:
        column_names.append(name)
```

5. Creating a dictionary:

```
launch_dict= dict.fromkeys(column_names)

# Remove an irrelvant column
del launch_dict['Date and time ( )']
del launch_dict['Customer']
# Let's initial the launch_dict with each value to be an empty list
launch_dict['Flight No.'] = []
launch_dict['Launch site'] = []
launch_dict['Payload'] = []
launch_dict['Payload mass'] = []
launch_dict['Orbit'] = []

launch_dict['Launch outcome'] = []
# Added some new columns
launch_dict['Version Booster']=[]
launch_dict['Booster landing']=[]
launch_dict['Date']=[]
launch_dict['Time']=[]
```

6. Filling dictionary with launch records:

```
launch_dict['Flight No.'].append(flight_number)
launch_dict['Date'].append(date)
launch dict['Time'].append(time)
launch_dict['Launch site'].append(bv)
launch_dict['Payload'].append(payload)
launch_dict['Payload mass'].append(payload_mass)
launch dict['Orbit'].append(orbit)
launch_dict['Launch outcome'].append(launch_outcome)
launch_dict['Booster landing'].append(booster_landing)
```

7. Creating dataframe & exporting to csv file:

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

https://github.com/Bob-Gemihood/IBM_capstone/blob/main/02.capstone_webscraping.ipynb

Data Wrangling

- Using the collected dataset, the attributes Flight Number, Date, Booster Version, Payload Mass, Orbit, Launch Site, Outcome, Flights, Grid Fins, Reused, Legs, Landing pad, Block, Reused Count, Serial, Longitude & Latitude of launch were reviewed.

- The column Outcome indicated if first stage successfully landed, there was 8 of them with True meaning that booster landed successfully to a drone ship & False meaning that it was unsuccessful.

- Landing Outcome was converted to classes 0 & 1 using one hot encoding with 0 denoting bad outcome & 1 denoting good outcome.

Data Wrangling

1. Checking null values:

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

2. Calculating the number of launches on each site:

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

3. Calculating the number & occurance of each orbit:

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

4. Calculating the number & occurance of mission outcome per orbit type:

```
landing_outcomes = df['Outcome'].value_counts()
print(landing_outcomes)
for i,outcome in enumerate(landing_outcomes.keys()):
    print(i,outcome)
```

5. Creating a set of outcomes where the second stage did not land successfully:

```
bad_outcomes=set(landing_outcomes.keys()[[1,3,5,6,7]])
bad_outcomes
```

6. Creating a landing outcome label from outcome column:

```
landing_class = []

for element in (df['Outcome']):
    if element in bad_outcomes:
        landing_class.append(0)
    else:
        landing_class.append(1)
```

7. Determining the success rate:

```
df["Class"].mean()
```

https://github.com/Bob-Gemihood/IBM_capstone/blob/main/03.capstone_data_Wrangling.ipynb

EDA with Data Visulaization

- Some attributes were used to determine if first stage can be reused.
- It was observed that success rate since 2013 has improved.
- Different launch sites had different success rates, as a result, they could be used to help determining if first stage would land successfully.
- It was noticed that CCAFS LC-40 had a success rate of 60% while KSC LC-39A and VAFB SLC 4E had a success rate of around 77%.
- When the result of the landing outcomes were overlayed as a colour it was noticed that CCAFS LC-40 had a success rate of 60% but if the mass was above 10,000 kg the success rate was 100%.

https://github.com/Bob-Gemihood/IBM_capstone/blob/main/05.capstone_EDA_visualization.ipynb

EDA with SQL

- SQL queries performed include:
  - Displaying the names of the unique launch sites in the space mission.
  - Displaying 5 records where launch sites began with the string ‘CCA’.
  - Displaying the total payload mass carried by boosters launched by NASA (CRS)
  - Displaying average payload mass carried by booster version F9 v1.1.
  - Listing the date when the first successful landing outcome in ground pad was achieved.
  - Listing the names of the boosters which had success in drone ship and had payload mass > 4000 but < 6000.
  - Listing the total number of successful & failure mission outcomes.
  - Listing the names of booster_versions which had carried the maximum payload mass using a subquery.
  - Listing the records that displayed the month names, failure landing_outcomes in drone ship, booster versions, launch_site for the months in year 2015.
  - Ranking the count of successful landing_outcomes between the date 04-06-2010 and 20-03-2017 in descending order.

https://github.com/Bob-Gemihood/IBM_capstone/blob/main/04.capstome_sqlite.ipynb

Building an Interactive Map with Folium

- Launch site locations & their close proximities were marked by creation & adding folium.Circle and folium.Marker for each launch site on the site map.
- Markers for all launch records were added so that If a launch was successful (class=1), a green marker was used and if a launch was failed, a red marker (class=0) was used.
- A Polyline between a launch site to a selected coastline point, raliway, highway and a city map symbol were added.
- The target was to explain how an optimal launch site was chosen.

https://github.com/Bob-Gemihood/IBM_capstone/blob/main/06.capstone_maps.ipynb

Building a Dashboard with Plotly Dash

- A dashboard application was built with Plotly Dash package.
- The dashboard contained input components such as dropdown list and a range slider to interact with a pie chart and a scatter point chart.
- The target was to find more insights from the SPACE X dataset more easily than with static graphs.

https://github.com/Bob-Gemihood/IBM_capstone/blob/main/07.capstone_dashboard.ipynb

Predictive Analysis (Classification)

- Machine Learning pipeline was built to predict if first stage of the FALCON 9 would land successfully. It included:
  - Preprocessing: allowing to standardize the data.
  - Train_test_split: allowing to split the data into training & testing subsets.
  - Training the model & performing Grid Search: allowing to to find the hyperparameters that allow a given algorithm to perform best. Using the hyperparameter values, the model with the best accuracy would be determined using the training data.
  - Different classification algorithms were tested including Logistic regression, SVM, Decision Tree classifier & KNN.
  - Finally the confusion matrix would be created.

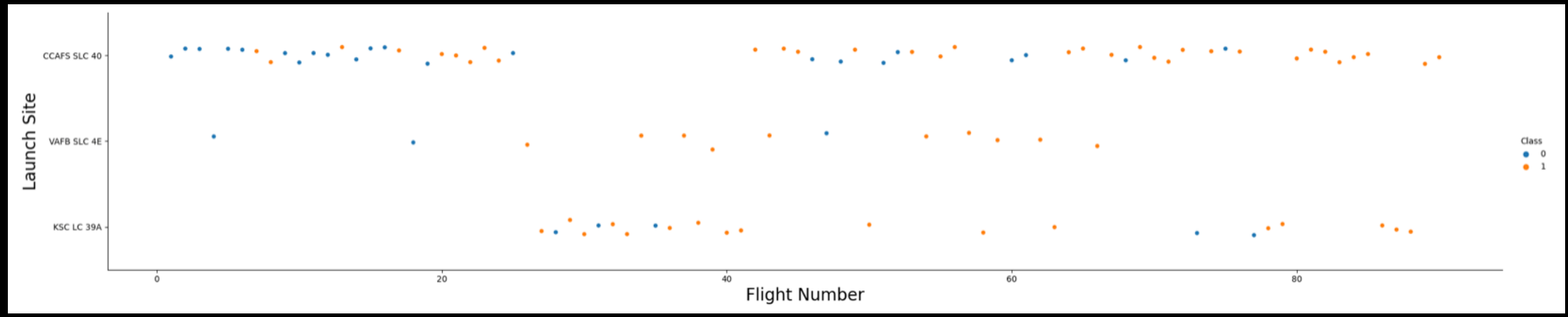
https://github.com/Bob-Gemihood/IBM_capstone/blob/main/08.capstone_ML.ipynb

RESULTS

- Insights from EDA with Visualization
- Insights from EDA with SQL
- Insights from Launch Sites Proximities Analysis
- Insights from Dashboard
- Results of Predictive Analysis

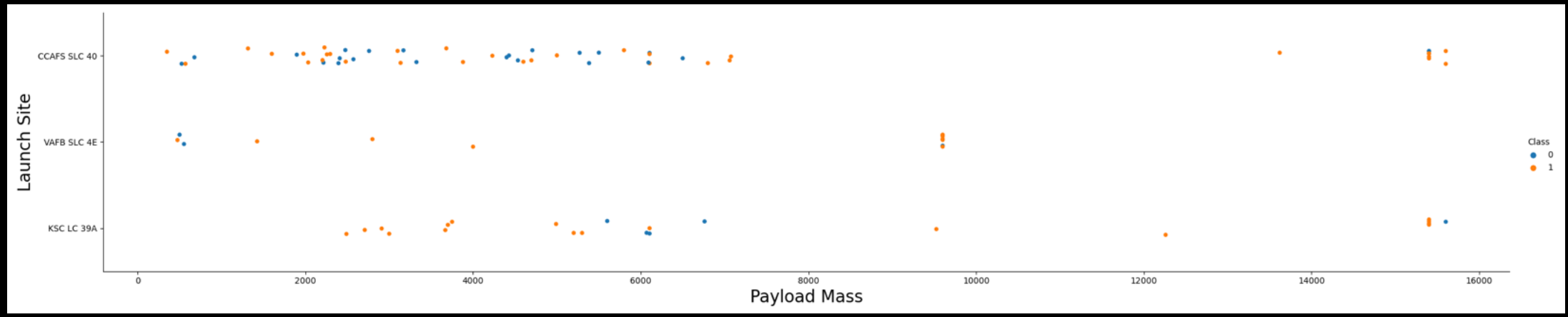
INSIGHTS FROM EDA WITH VISUALIZATION

Flight Number vs. Launch Site



- It could be clearly noticed that launches from the launch site CCAFS SLC 40 are significantly higher compared to the other sites.

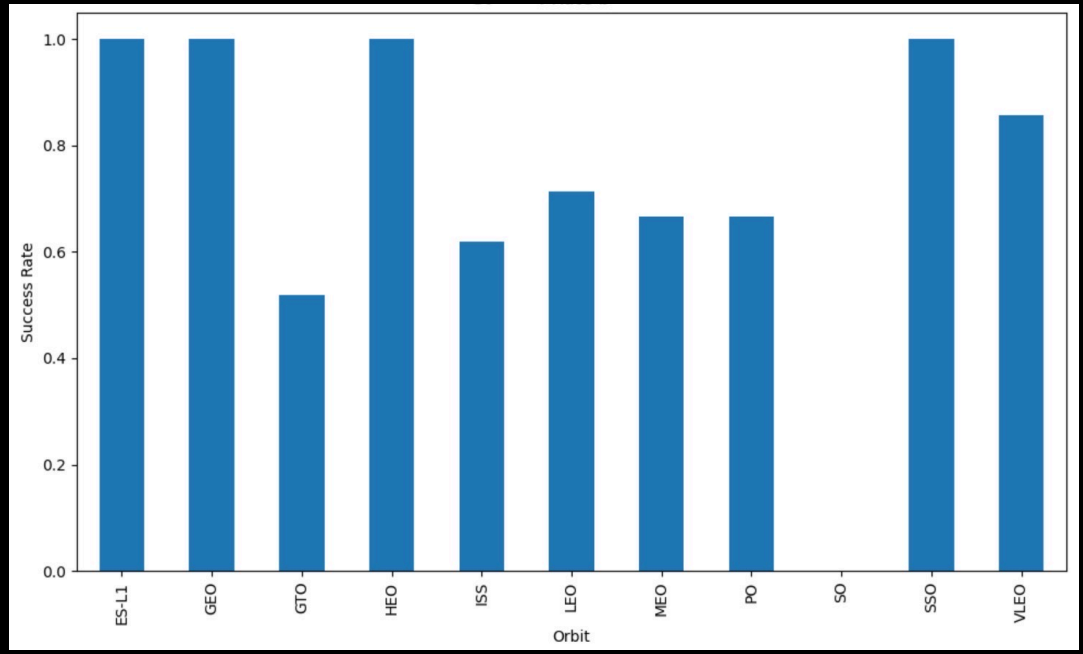
Payload Mass vs. Launch Site



- It is noticable that for the VAFB SLC 4E launch site there were no rockets launched for load mass > 10000.
- The majority of launches for lower load masses were from the launch site CCAFS SLC 40.

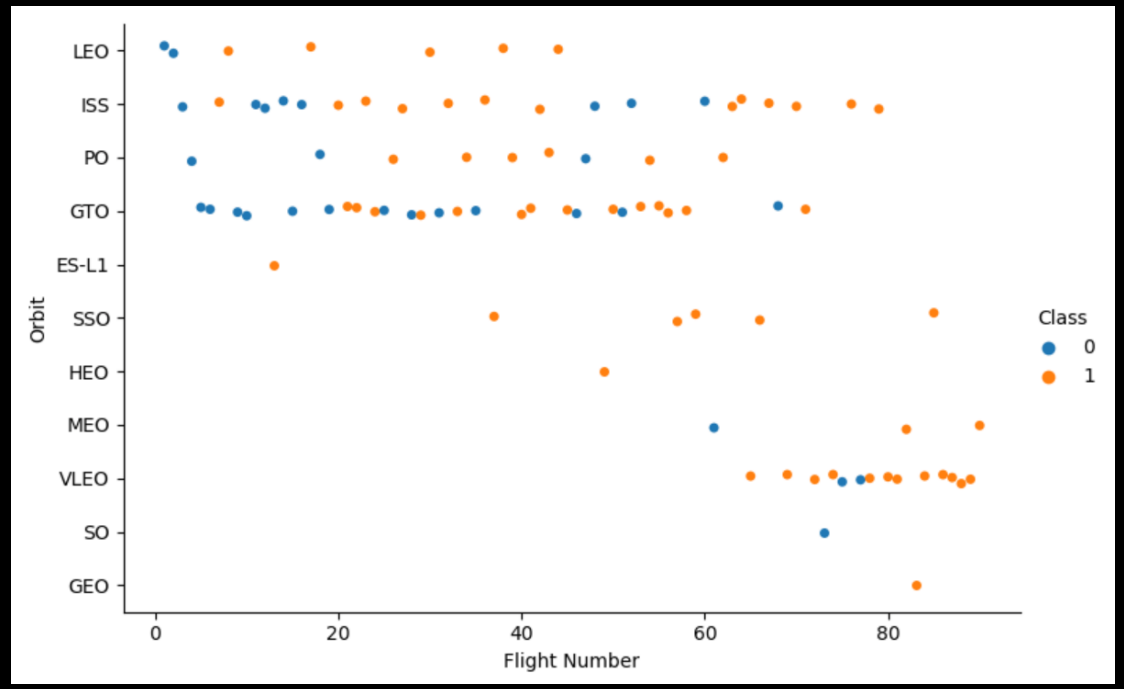
Success Rate vs. Orbit Type

- The orbit types ES-L1, GEO, HEO and SSO had the highest success rates.

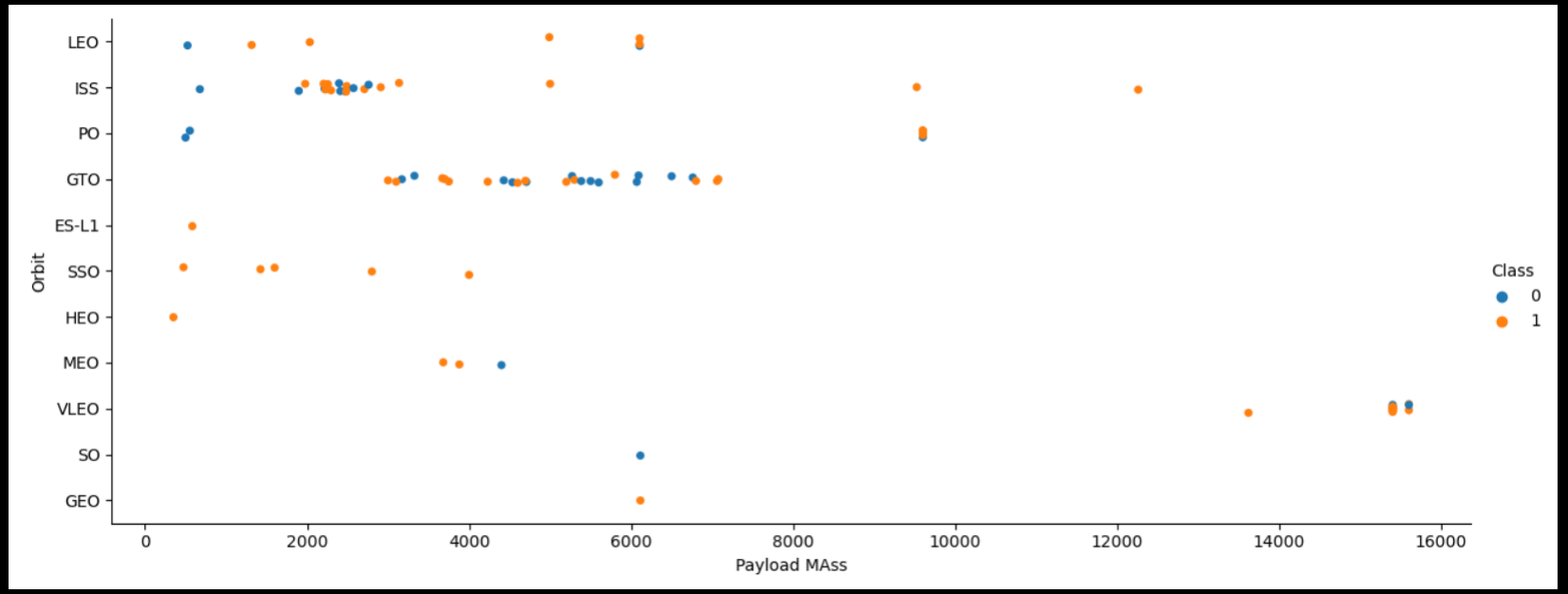


Flight Number vs. Orbit Type

- LEO orbit success was related to the number of flights.
- No relationship between flight number & CTO orbit success.
- More VLEO orbit launches were noticed in the recent most of which were successful.



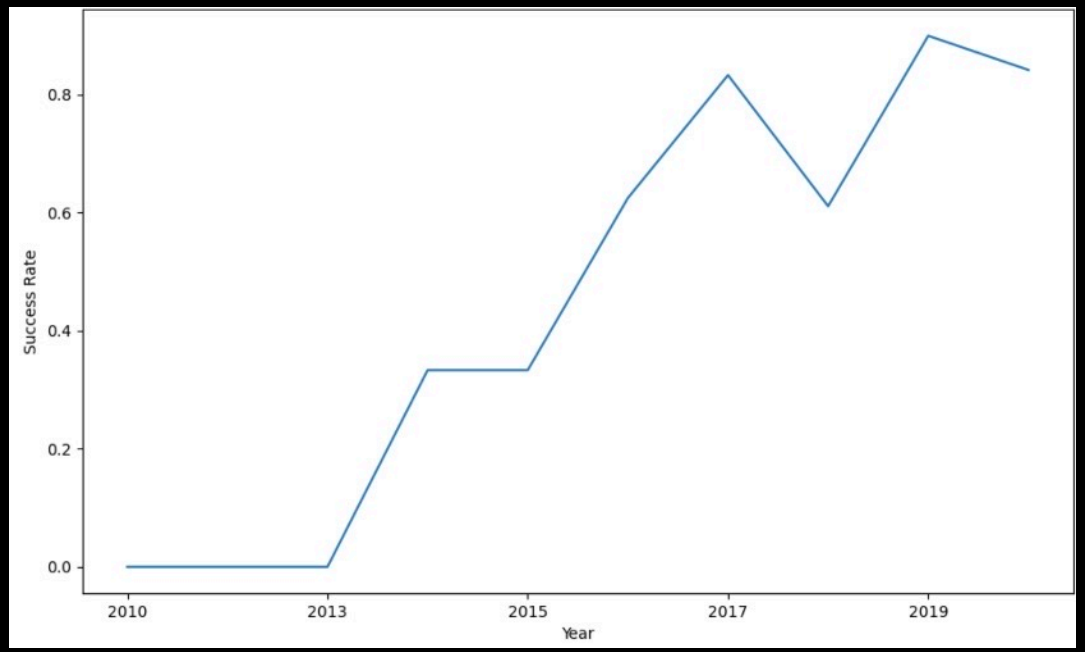
Payload Mass vs. Orbit Type



- It was observed that successful landing rates were more for orbits LEO and ISS with heavy payload masses

Launch Success Yearly Trend

- Launch success rate since 2013 kept increasing till 2020.



INSIGHTS FROM EDA WITH SQL

All Launch Site Names

```
%sql SELECT DISTINCT "Launch_Site" From SPACEXTBL;
```

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

- Four unique launch site names were detected.

Launch Site Names Began with ‘CCA’

```
%sql SELECT * From SPACEXTBL WHERE "Launch_Site" LIKE "CCA%" LIMIT 5;
```

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

```
%%sql SELECT SUM("PAYLOAD_MASS__KG_") AS SUM
FROM SPACEXTBL WHERE "Customer" = "NASA (CRS)";
```

SUM
45596

- Calculated total payload carried by boosters from NASA.

Average Payload Mass by F9 v1.1

```
%%sql SELECT AVG("PAYLOAD_MASS__KG_") AS AVG
FROM SPACEXTBL WHERE "BOOSTER_VERSION" = "F9 v1.1";
```

AVG
2928.4

First Successful Ground Landing Date

```
%%sql SELECT min(Date) AS Successful_Date
FROM SPACEXTBL
WHERE "Landing _Outcome" = "Success (ground pad)";
```

Successful_Date
2015-12-22

- The date of the first successful landing outcome on ground pad.

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

- Names of the boosters that had successfully landed on drone ships and had payload mass between 4000 and 6000.

Total Number of Successful and Failure Mission Outcome

```
%%sql SELECT "Mission_outcome", COUNT(Mission_Outcome) AS "COUNT"
FROM SPACEXTBL
WHERE "Mission_outcome" LIKE "Success%" OR "Mission_outcome" LIKE "Failure%"
GROUP BY "Mission_outcome";
```

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

Boosters Carried Maximum Load

```
%%sql SELECT "Booster_Version" FROM SPACEXTBL
WHERE "PAYLOAD_MASS__KG_" IN
(SELECT MAX("PAYLOAD_MASS__KG_") AS MAX FROM SPACEXTBL);
```

- Names of boosters that had carried the maximum payload mass using subquery.

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

```
%%sql SELECT substr(Date,4,2) AS MONTH,
"Landing _Outcome","Booster_Version","Launch_Site"
FROM SPACEXTBL
WHERE substr(Date,7,4)='2015'
AND "Landing _Outcome" LIKE "Failure (Drone Ship)%"
ORDER BY Date
```

MONTH	Landing _Outcome	Booster_Version	Launch_Site
01	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
04	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

- List of the failed landing outcomes in drone ships, their booster versions and launch site names in year 2015.

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

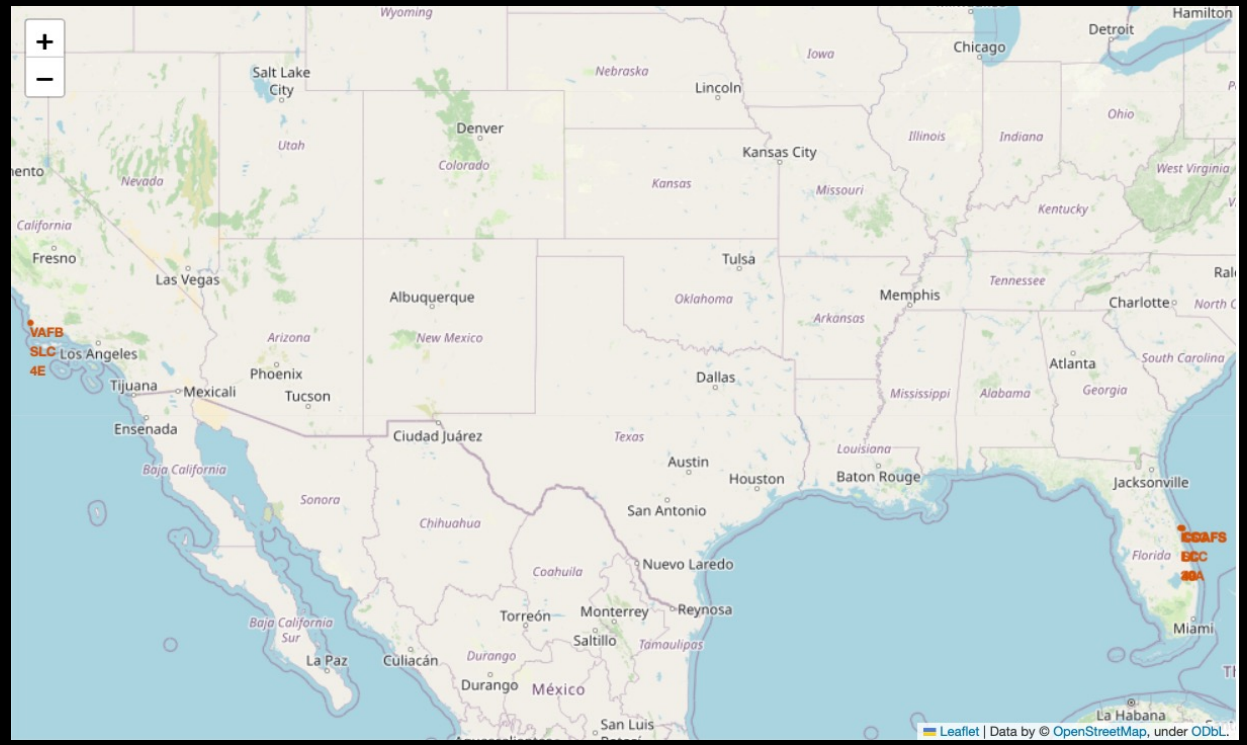
```
%%sql SELECT "Landing _Outcome", COUNT("Landing _Outcome") AS "Count", "Date"
FROM SPACEXTBL
WHERE "Date" Between '04-06-2010' AND '20-03-2017'
AND "Landing _Outcome" LIKE "Success (Ground Pad)%"
OR "Landing _Outcome" LIKE "Failure (Drone Ship)%"
GROUP BY "Landing _Outcome"
ORDER BY "Date" DESC;
```

Landing _Outcome	Count	Date
Success (ground pad)	6	18-07-2016
Failure (drone ship)	5	10-01-2015

- Ranking the count of landing outcomes (Failure (drone ship) or Success (ground pad)) between the date 2010-06-04 and 2017-03-20.

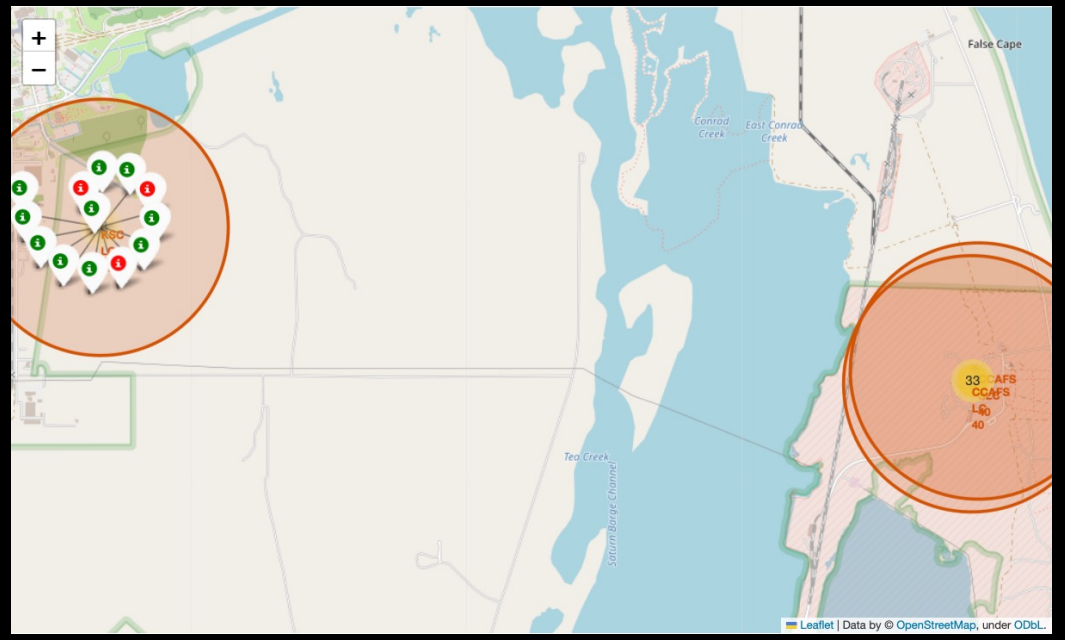
INSIGHTS FROM LAUNCH SITES PROXIMITIES ANALYSIS

All Launch Sites



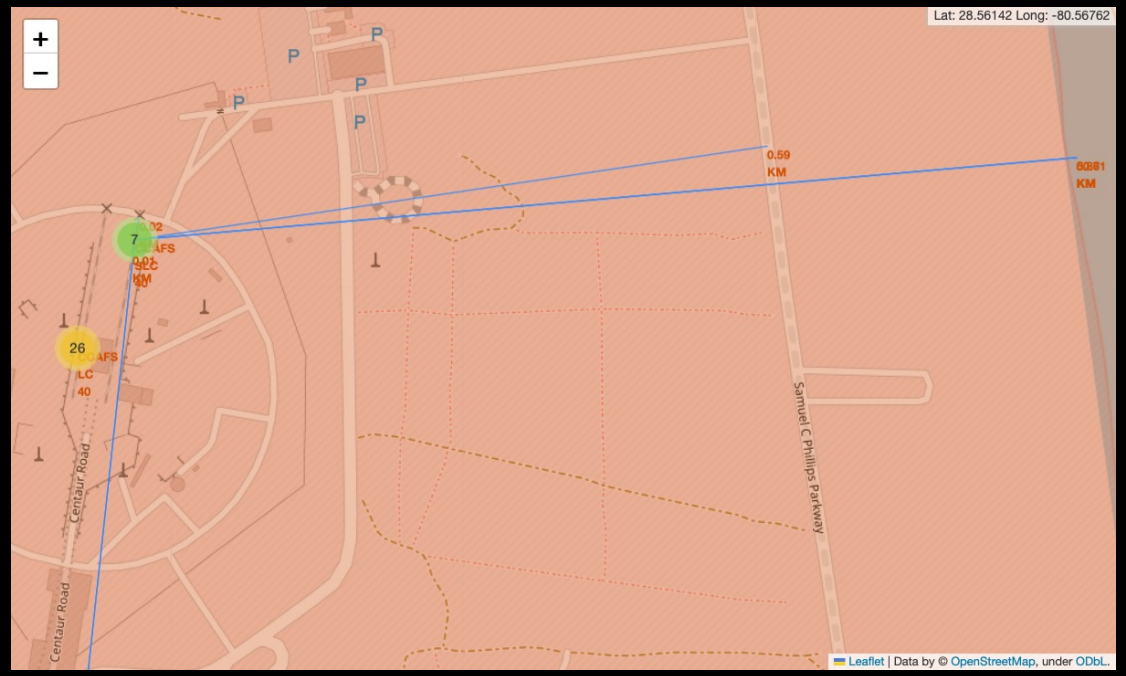
- A folium.Circle and folium.Marker for each launch site were added on the site map.

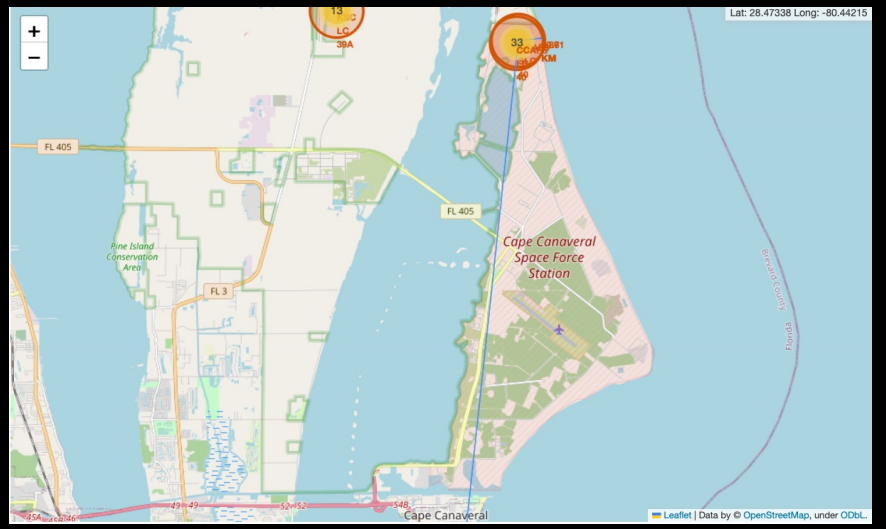
Success/Failed Launches for Each Site

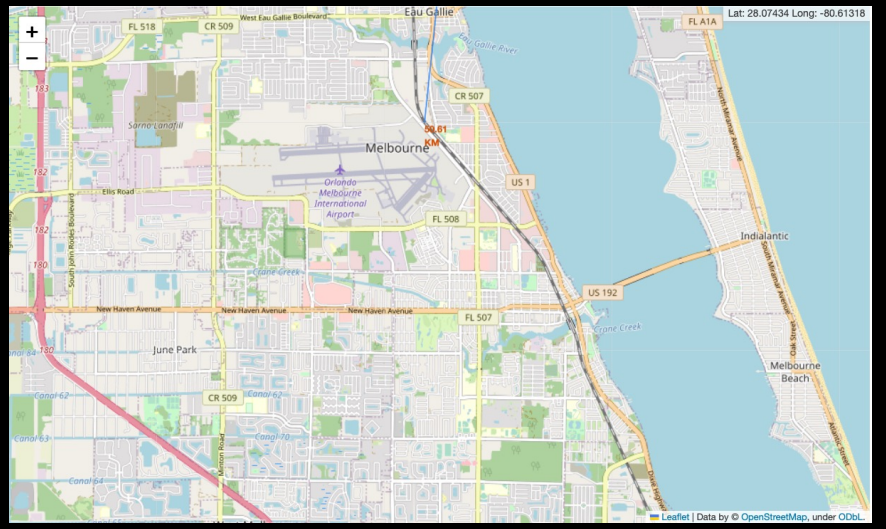


- A marker_cluster was created and a color labeled folium.Marker for each launch site was added to the marker_cluster with green color for class 1 and red color for class 0.

Launch Site and Proximities



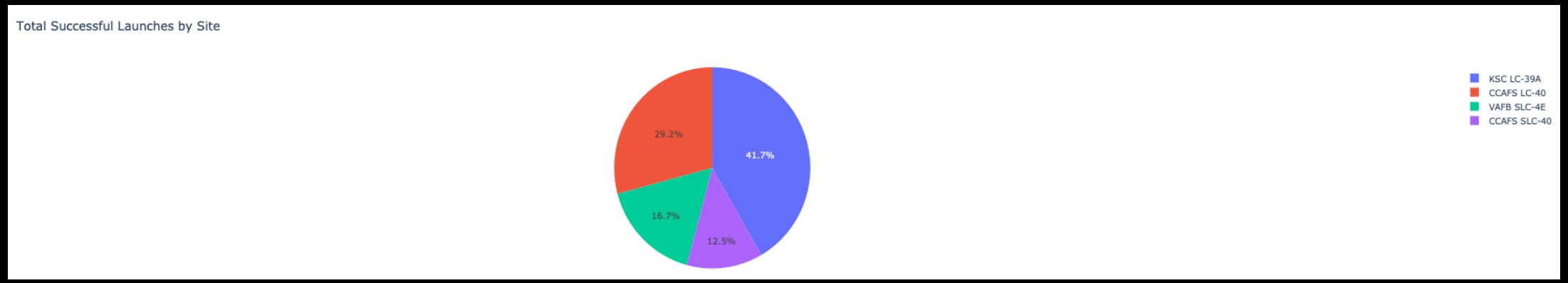




- Lines were drawn between one launch site and its closest coastline point, railway, highway and city.

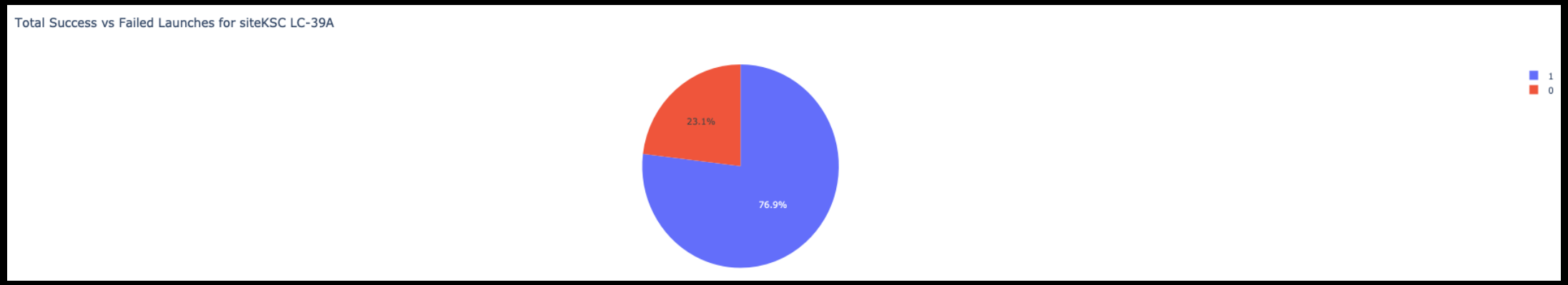
INSIGHTS FROM DASHBOARD

Launch Success Count for All Sites



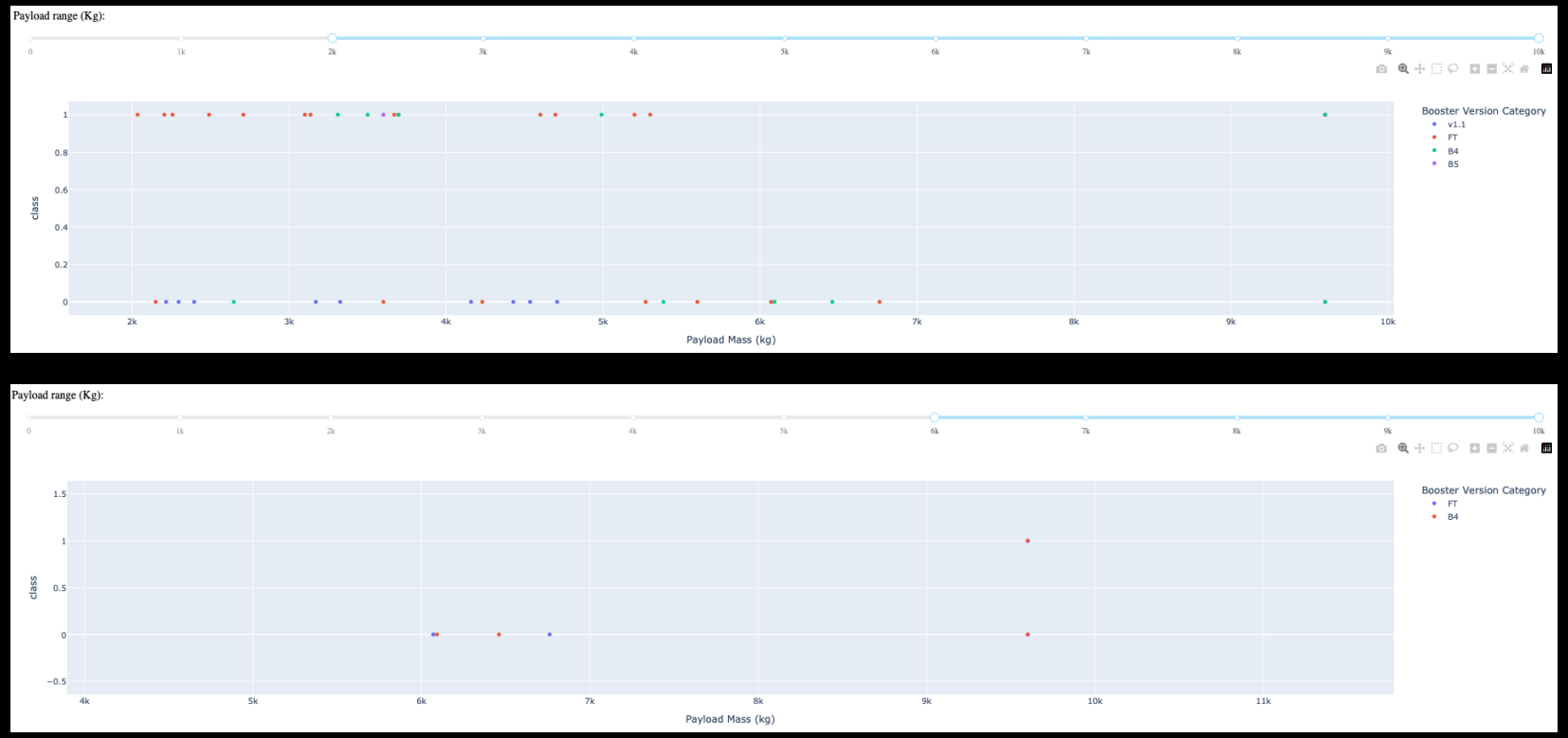
- KSC LC-39A launch site had the most successful launches compared to all the other sites.

Launch Site with The Highest Success Rate



- KSC LC-39A launch site achieved 76.9 % success rate compared to 23.1 % failure rate.

Payload vs. Launch Outcome Scatter Plot

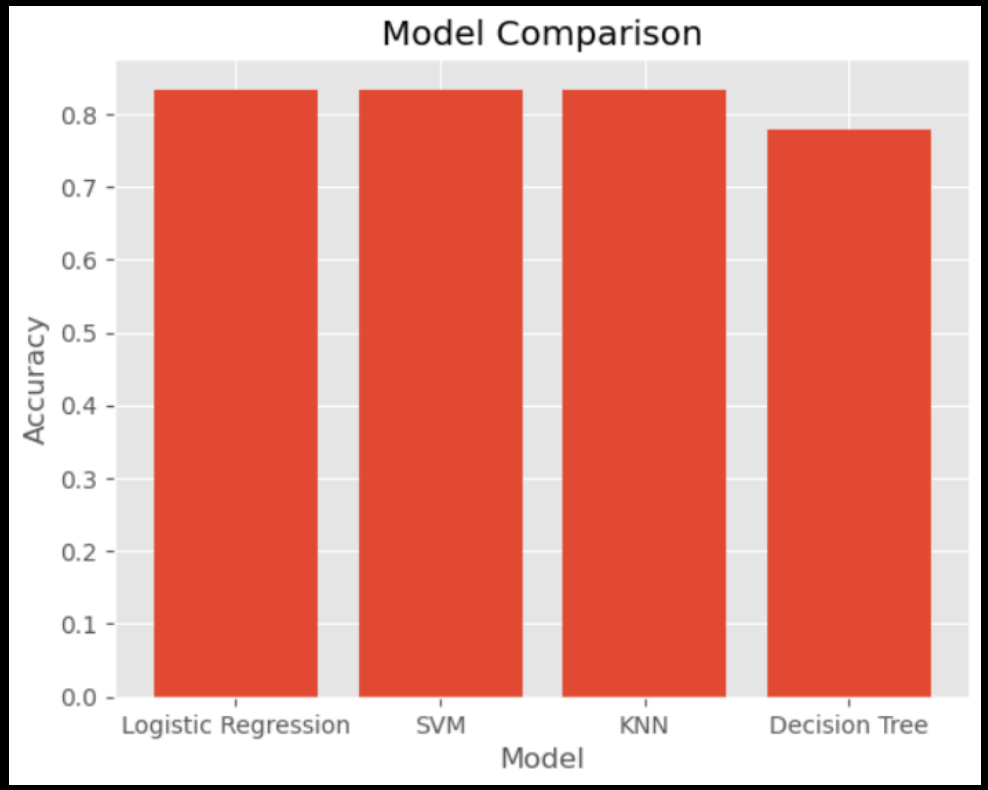


- It was noticed that the success rate was higher with lighter weighted payloads.

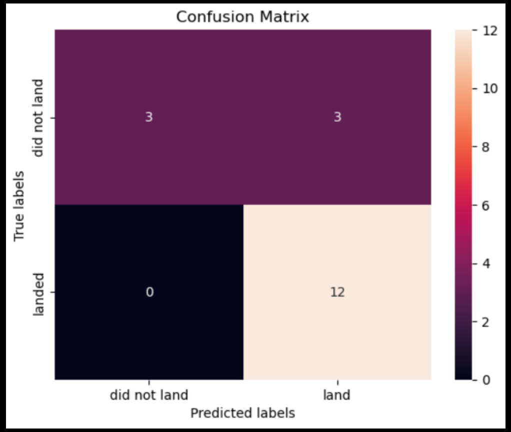
RESULTS OF PREDICTIVE ANALYSIS (CLASSIFICATION)

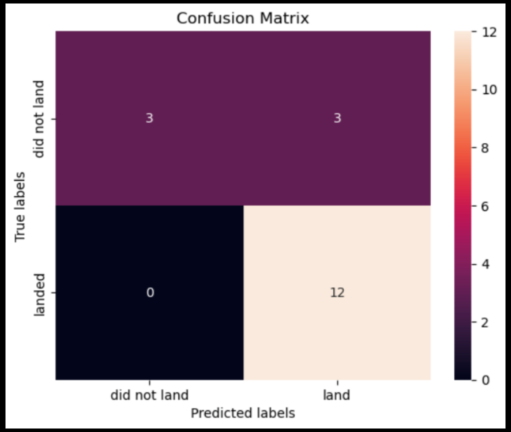
Classification Accuracy

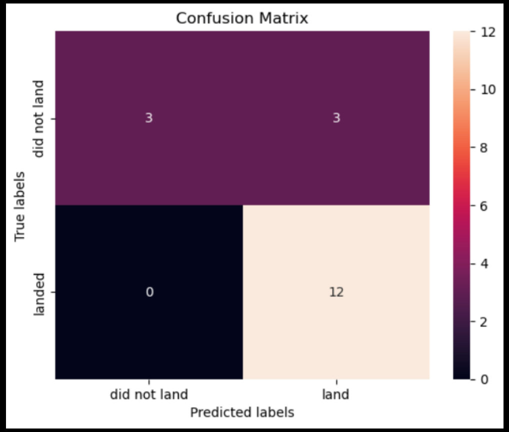
- Logistic Regression, SVM and KNN models had the same accuracy of 83.33 % compared to less accuracy of the Decision Tree classifier that had 77.78 % accuracy.

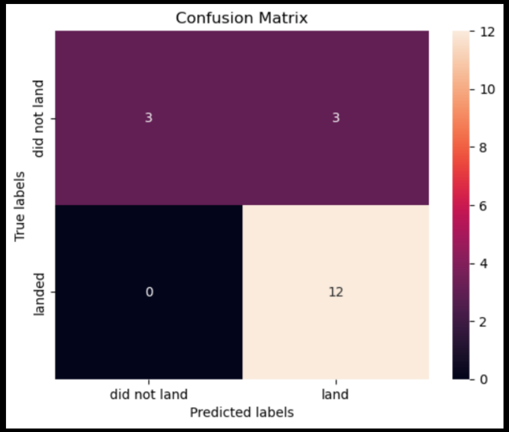


Confusion Matrix









CONCLUSIONS

- The orbit types ES-L1, GEO, HEO and SSO have the highest success rates.
- Low weighted payloads have higher success rate.
- Launch sites are chosen as far as possible from cities.
- KSC LC-39A launch site had the most successful launches compared to all the other sites.
- Logistic regression, SVM & KNN models are the best to predict the accuracy for this project.

THANK YOU