IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Vikas Prasad
3/27/2022

Table of Contents

Executive Summary

- Summary of methodologies
  1. Data Collection
  2. Data Wrangling
  3. EDA With Data Visualization
  4. EDA With SQL
  5. Building an Interactive Map with Folium
  6. Building a Dashboard with Plotly Dash
  7. Predictive Analysis and Classification
- Summary of all results
  1. Exploratory Data Analysis results
  2. Interactive Analysis Demo
  3. Predictive Analysis Results

Introduction

Project Background and Context

- This problem concerns SpaceX, and in specific the Falcon 9 Launch; SpaceX the main problem being predicted is whether or not the first stage will be completed successfully. SpaceX spends 62 million dollars on advertising the Falcon 9 launch on their website while other providers cost upwards of 165 million each. Therefore, if we can determine if the first stage will land, we can determine the cost of a launch. The information can be used if an alternate company wants to bid against SpaceX for a rocket launch.

Problems you want to find answers

- What influences if the rocket will land successfully?
- The effect each relationship with certain rocket variables will impact in determining the success rate of a successful landing?
- What conditions does SpaceX have to achieve to get the best results and ensure the best rocket landing success rate?

Section 1

Methodology

Methodology

- Data collection methodology:
  - Web Scraping from websites (ex. Wikipedia)
  - SpaceX REST API
- Perform data wrangling
  - Transforming data for Machine Learning → One Hot encoding data fields for machine learning and dropping irrelevant columns
- Perform exploratory data analysis (EDA) using visualization and SQ
  - Transforming data for Machine Learning → One Hot encoding data fields for machine learning and dropping irrelevant column
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models

Data Collection

- I worked with SpaceX launch data gathered form the SpaceX REST API → this API gives data about launches, including information about the rocket used, payload delivered, launch and landing specifications, and landing outcome.
- Webscraping Wikipedia using Beautiful Soup was another way I gained data for the project. The data was then normalized into flat data files such as a .csv file.

Data Collection – SpaceX API

- 1. Getting response from API
- 2. Converting to json()
- 3. Clean data using custom functions
- 4. Assign list to dictionary and data frame
- 5. Export data frame

[Github Link to Notebook:]()

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

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

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

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

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

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

```
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}
```

```
data_falcon9.to_csv('dataset_part\_1.csv', index=False)
```

Data Collection - Scraping

- 1. Getting response from HTML
- 2. Creating Beautiful Soup
- 3. Finding tables and getting column names
- 4. Creation of Dictionary
- 5. Appending data to keys
- 6. Converting dictionary to data frame and exporting

[Github Link to Notebook]()

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

```
Soup = BeautifulSoup(response.text, 'html.parser')
```

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

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

```
launch_dict= dict.fromkeys(column_names)
del launch_dict['Date and time ( )']
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']=[]
```

```
extracted_row = 0
for table_number,table in enumerate(So
    for rows in table.find_all("tr"):
```

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

Data Wrangling Background

From examining the data set, there are multiple occurrences where the booster did not land successfully. Sometimes, attempts were made but they failed; True and False Ocean means that mission successfully landed or unsuccessfully landed to a specific region in the ocean respectively. True RTLS and False RTLS means the mission successfully or unsuccessfully landed to a ground pad respectively. True ASDS and False ASDS means the mission successfully landed or unsuccessfully landed on a drone ship respectively. The value 1 refers to the fact that the booster was successfully and the value 0 means it was unsuccessful.

Data Wrangling Process

Perform Exploratory Data Analysis (EDA) on the dataset

Calculate the number of launches at each site

Calculate the number of occurrences at each orbit

Calculate the number and occurrence of mission outcome per orbit type

Create a landing outcome label from Outcome column

Work out success rate for every landing in the data set

Export data as a .csv file

Github Link to Notebook

EDA with Data Visualization

Charts Used:

- Scatter Plots: Flight Number v. Payload Mass, Flight Number v. Launch Site, Payload v. Launch Site, Orbit v. Flight Number, Payload v. Orbit Type, Orbit v. Payload Mass

Scatter plots portray how much one variable is being affected by another.

- Bar Graphs: Mean v. Orbit

A bar graph helps visualize the differences between groups; they help show the large changes in data over time.

- Line Graph: Success Rate v. Year

Line graphs are beneficial in that they show the data variables and trends clearly

[Link to Notebook on GitHub]()

EDA with SQL

[Link to Notebook on GitHub]

SQL Queries performed:

- Displaying the names of the unique launch sites
- Displaying 5 records where launch sites begin with “KSC”
- Displaying the total payload mass carried by boosters launched by NASA
- Displaying average payload mass carried by booster version F9 v1.1
- Listing the date where the successful landing outcome in drone ship was achieved
- Listing the names of the boosters which have success in ground pad and have payload mass greater than 4000 but less than 6000.
- Listing the total number of failed and successful mission outcomes
- Listing the names of the booster versions which have carried the maximum payload mass
- Listing the records which will display the month names, successful landing outcomes in ground pad, booster versions, launch sites in 2017
- Ranking the count of successful landing outcomes between June 4th 2010 and March 3rd 2017 in descending order

Build an Interactive Map with Folium

Visualize the Launch Data in an Interactive Map:

- We took the latitude and longitude coordinates at each launch site and added a circle marker around each launch site with an appropriate label

Assigned the values 0 and 1 to failures and successes for the dataframe launch_outcomes column. There are red and green markers respectively on the interactive map that designate the failures and successes.

Github url to Notebook:

Build a Dashboard with Plotly Dash

Shows scatter plots portraying the relationship between the variables to enhance user understanding of the data.

Pie Charts are shown to portray the total launches for each of the launch sites and all the launch sites in general.

Github Link:

Predictive Analysis (Classification)

BUILDING MODEL

- Load dataset into NumPy and Pandas
- Transform Data
- Split data into training and test splits
- Check how many data samples there are
- Decide the machine learning algorithm for the scenario
- Set parameters and algorithms to GridSearchCV
- Fit datasets into the GridSearchCV objects and train the data set

EVALUATING MODEL

- Check accuracy for each model
- Get tuned hyperparameters for each type of algorithms
- Plot confusion matrix

IMPROVING MODEL

- Feature Engineering
- Algorithm Tuning

FINDING THE BEST PERFORMING CLASSIFICATION MODEL

- The model with the best accuracy score wins the best performing model
- The other algorithms and their accuracy scores are also listed in the notebook for comparison purposes.

Results

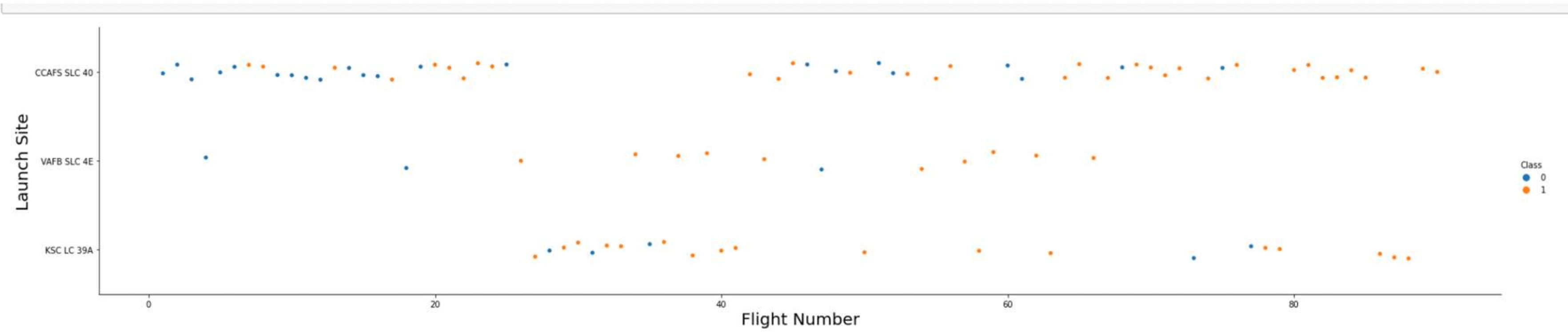
- Exploratory data analysis results
- Interactive analytics demo in screenshots
- Predictive analysis results

Section 2

Insights drawn from EDA

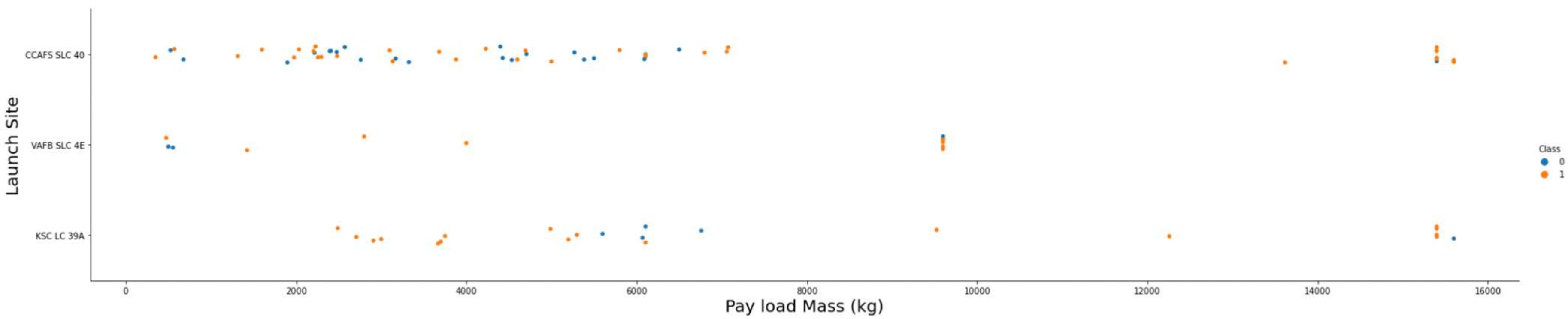
Flight Number vs. Launch Site

The more amount of launches at a site, the greater chance of success



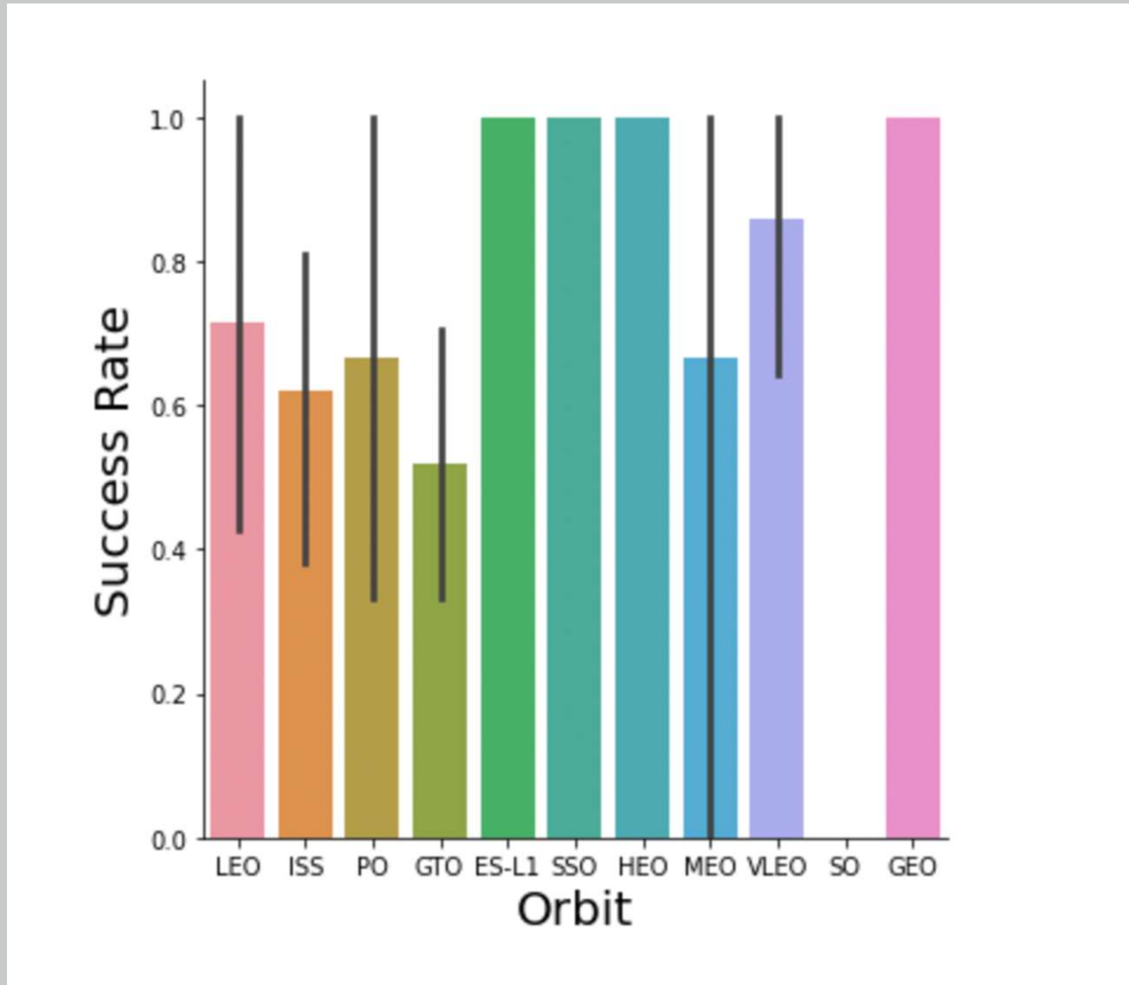
Payload vs. Launch Site

The greater the payload mass, the higher probability of success



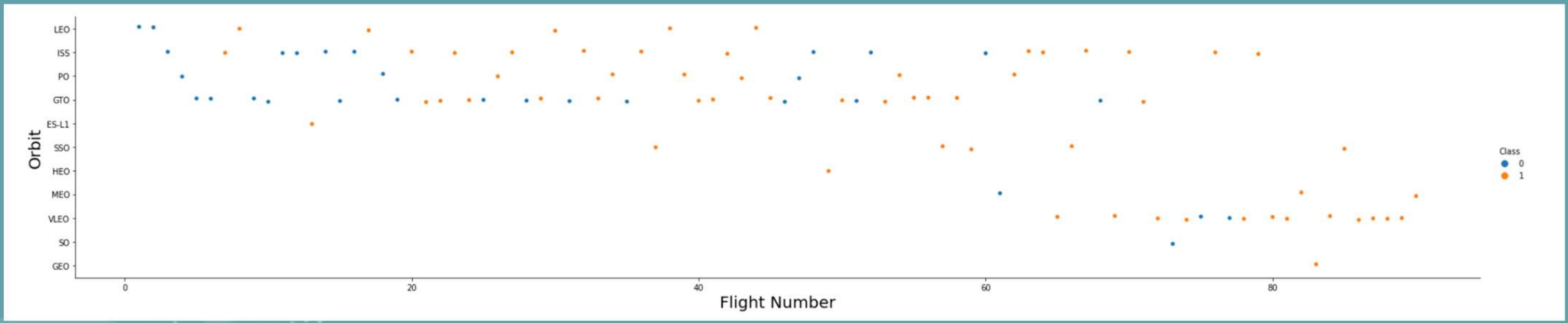
Success Rate vs. Orbit Type

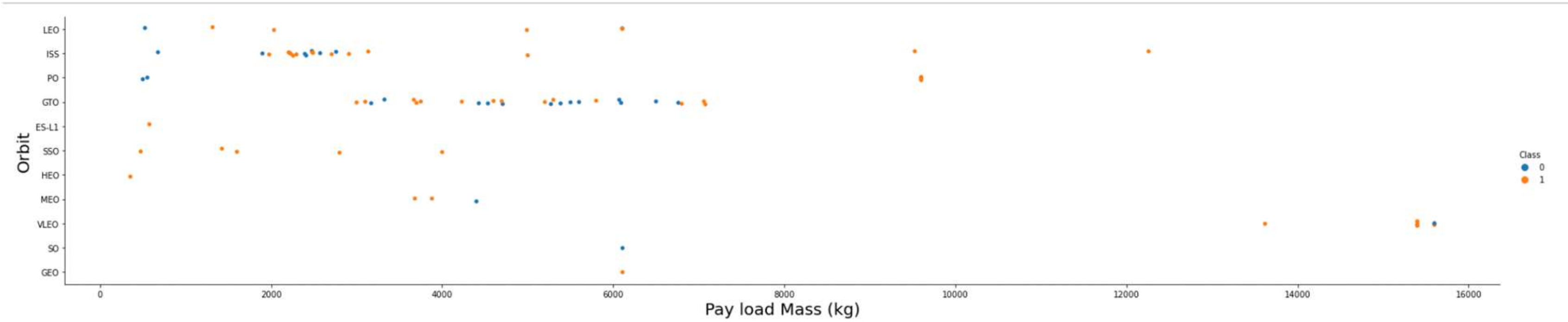
The orbits, ES-L1, SSO, HEO, and GEO have the greatest success rates



Flight Number vs. Orbit Type

Show a scatter point of Flight number vs. You should see that in the LEO orbit the Success appears related to the number of flights; on the other hand, there seems to be no relationship between flight number when in GTO orbit.



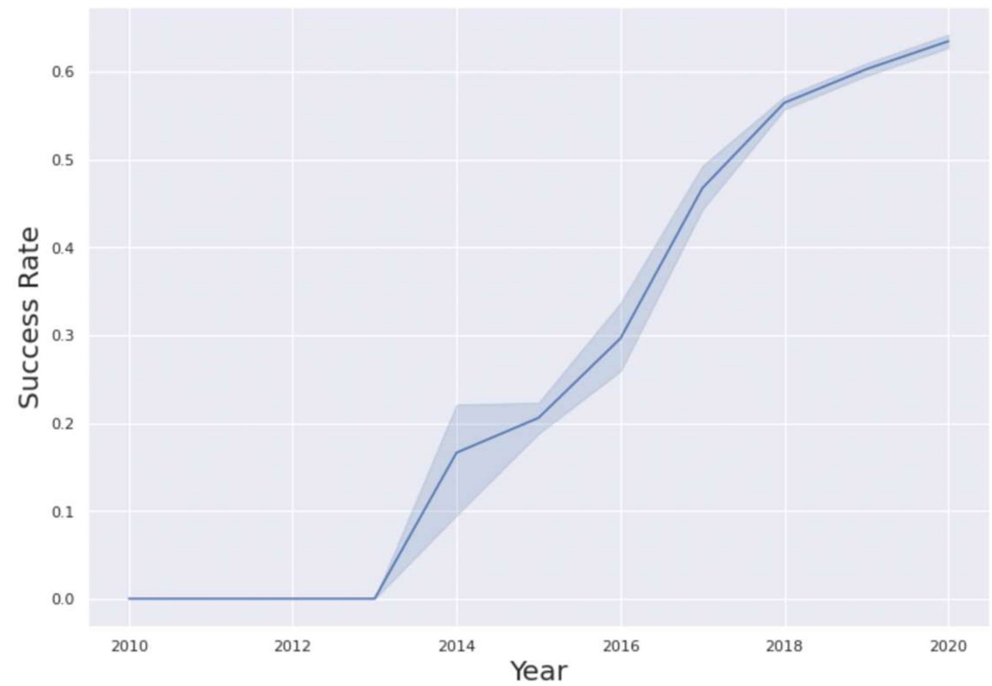


Payload vs. Orbit Type

With heavy payloads the successful landing or positive landing rate are more for Polar, LEO, and ISS.

Launch Success Yearly Trend

From the line plot, we can determine that the success rate has been increasing each year since 2013.



All Launch Site Names

Unique launch sites:

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

SQL Query:

Select DISTINCT launch_site from tblSpaceX

This will show all the unique values in the launch_site column in the table tblSpaceX

Launch Site Names Begin with 'CCA'

Query: Select TOP 5 * from tblSpaceX WHERE Launch_Site LIKE 'KSC%'

The KSC% ensures that the launch site must begin with KSC and the top 5 ensures that it will only show 5 records

	Date	Time_UTC	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS_KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
0	19-02-2017	2021-07-02 14:39:00.0000000	F9 FT B1031.1	KSC LC-39A	SpaceX CRS-10	2490	LEO (ISS)	NASA (CRS)	Success	Success (ground pad)
1	16-03-2017	2021-07-02 06:00:00.0000000	F9 FT B1030	KSC LC-39A	EchoStar 23	5600	GTO	EchoStar	Success	No attempt
2	30-03-2017	2021-07-02 22:27:00.0000000	F9 FT B1021.2	KSC LC-39A	SES-10	5300	GTO	SES	Success	Success (drone ship)
3	01-05-2017	2021-07-02 11:15:00.0000000	F9 FT B1032.1	KSC LC-39A	NROL-76	5300	LEO	NRO	Success	Success (ground pad)
4	15-05-2017	2021-07-02 23:21:00.0000000	F9 FT B1034	KSC LC-39A	Inmarsat-5 F4	6070	GTO	Inmarsat	Success	No attempt

Total Payload Mass

Total Payload Mass → 45596

SQL Query: select SUM(PAYLOAD_MASS_KG_) TotalPayloadMass from tblSpaceX WHERE Customer = ‘NASA (CRS)’, TotalPayloadMass

Query Explanation: The sum function ensures that the sum of the column will be shown, and the where clauses filters the data so the only values that re included will have customer equal to NASA (CRS).

Average Payload Mass

Average Payload Mass → 2928

SQL Query: select AVG(PAYLOAD_MASS_KG_) TotalPayloadMass from tblSpaceX WHERE Booster_Version = ‘F9 v1.1’

Query Explanation: The average function gets the average of the column and the where clause filters to only include the booster version.

First Successful Ground Landing Date

The first successful ground landing date was 06-05-2016

SQL Query: select MIN(Date) SLO from tblSpaceX where Landing_Outcome = “Success (Drone Ship)”

Query Explanation: The min function ensures that the date will be the earliest possible occurrence, thus the first successful. Moreover, the where clause filters the data so only successes are in looked at.

Successful Drone Ship Landing with Payload between 4000 and 6000

- SQL Query: select Booster_Version from tblSpaceX where Landing_Outcome = ‘Success (ground pad)’ AND Payload_MASS_KG_ > 4000 AND Payload_MASS_KG_ < 6000
- Explanation: Selecting only booster_versions where the conditions are it is a success, and the payload mass is between 4000 and 6000 exclusive.

	Date which first Successful landing outcome in drone ship was acheived.
0	F9 FT B1032.1
1	F9 B4 B1040.1
2	F9 B4 B1043.1

Total Number of Successful and Failure Mission Outcomes

- Total successful outcomes → 100
- Total failure mission outcomes → 1

SQL Query:

SELECT(SELECT Count(Mission_Outcome) from tblSpaceX WHERE Mission_Outcome LIKE '%Success%') as Successful_Outcomes,

(SELECT Count(Mission_Outcome) from tblSpaceX where Mission_Outcome like '%Failure%') as Failure_Outcomes

Explanation: The subqueries failure and success have to be in the String

Boosters Carried Maximum Payload

- Select DISTINCT Booster_Version, MAX(PAYLOAD_MASS_KG_) AS [Maximum Payload Mass] from tblSpaceX GROUP BY Booster_Version ORDER BY [Maximum Payload Mass] DESC
- This will only show unique values in the Booster_Version column due to the distinct keyword and will portrayed in descending order by maximum payload mass.

	Booster_Version	Maximum Payload Mass
0	F9 B5 B1048.4	15600
1	F9 B5 B1048.5	15600
2	F9 B5 B1049.4	15600
3	F9 B5 B1049.5	15600
4	F9 B5 B1049.7	15600
...	...	...
92	F9 v1.1 B1003	500
93	F9 FT B1038.1	475
94	F9 B4 B1045.1	362
95	F9 v1.0 B0003	0
96	F9 v1.0 B0004	0

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

- SQL Query: SELECT COUNT(Landing_Outcome) FROM tblSpaceX WHERE (Landing_Outcome LIKE ‘%Success%’) AND (Date > ‘04-06-2010’) AND (Date < ‘20-03-2017’)
- SQL Query Explanation: The AND keywords signify conditions, which is the boundaries of the date, and the count keyword counts the amount of occurrences where the success string appears.

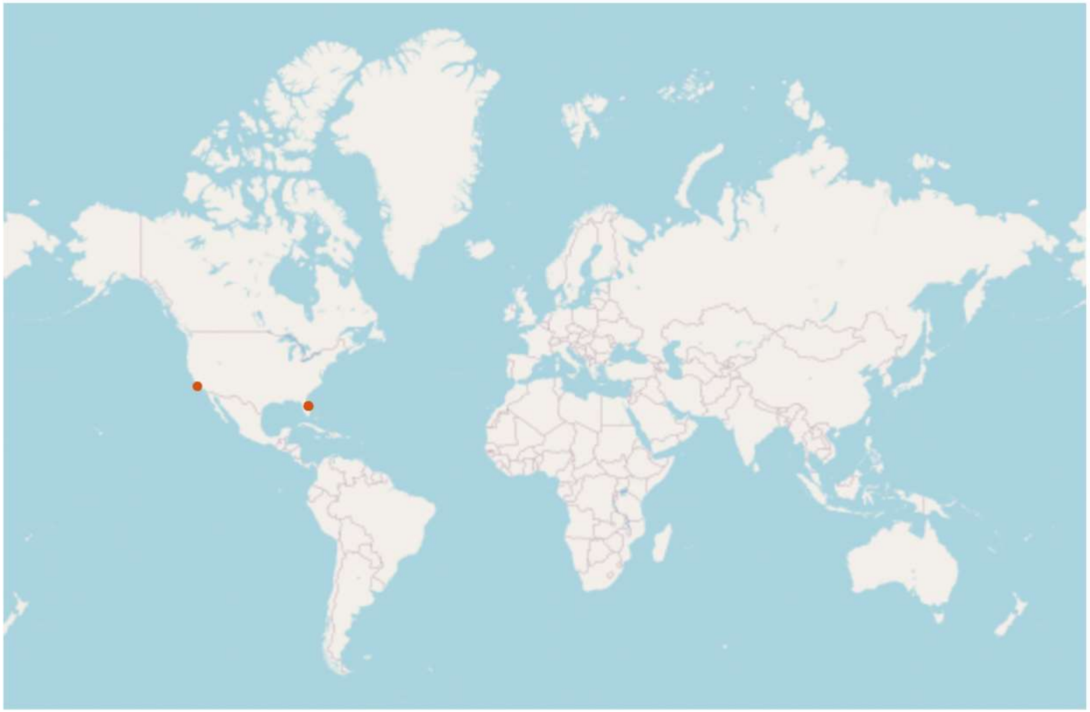
	Successful Landing Outcomes Between 2010-06-04 and 2017-03-20
0	34

Section 4

Launch Sites Proximities Analysis

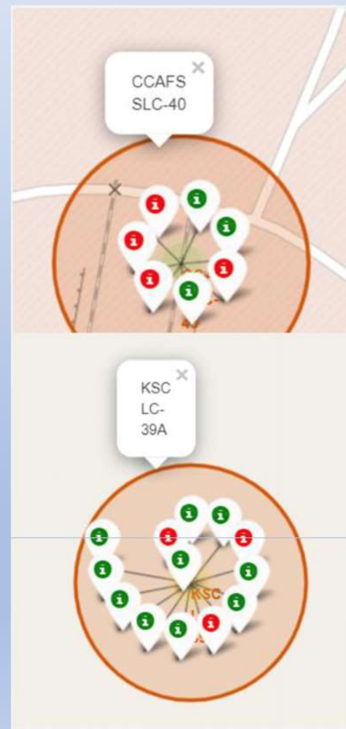
Global Map Markers

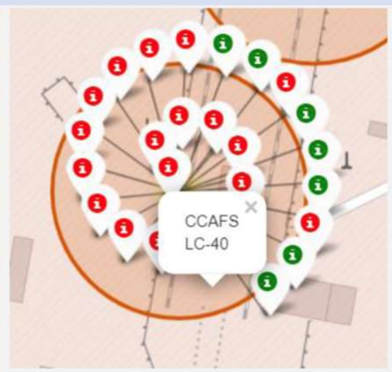
- The launch areas are found in the states of California and Florida, in the United States of America

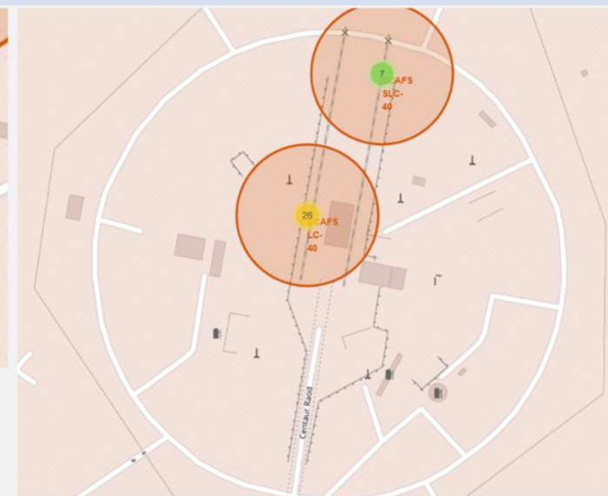


Color Labelled Marks

California Launch Site

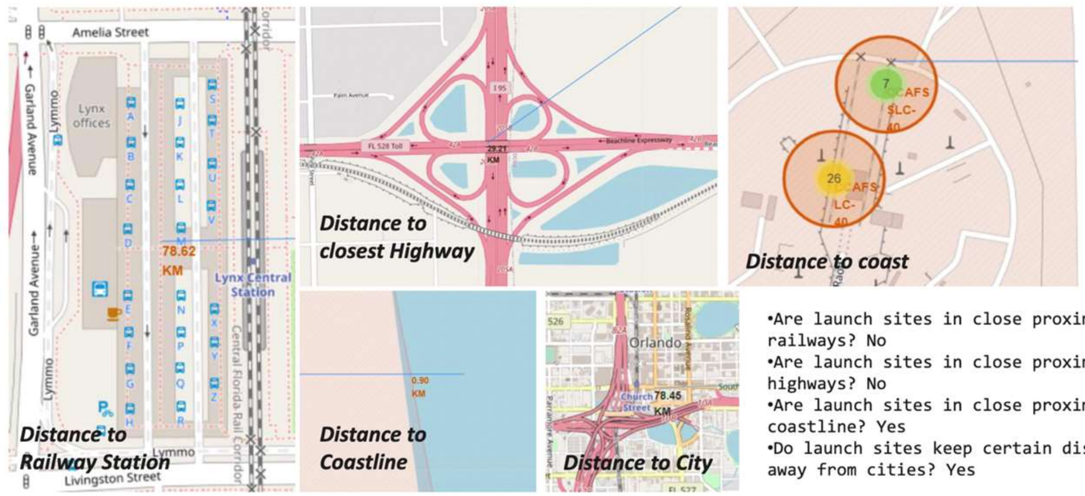






Florida Launch Sites

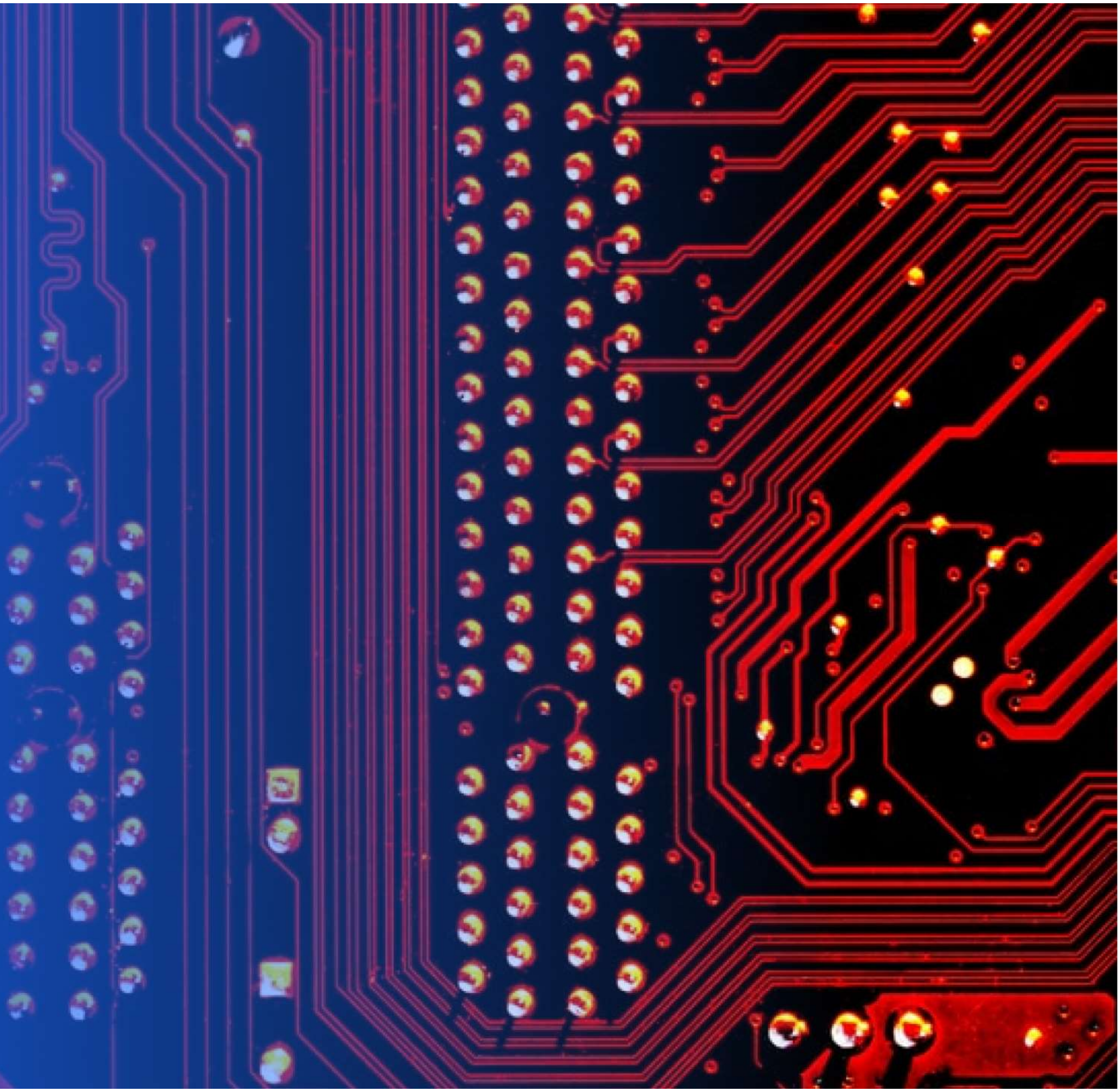
Launch Sites distance to ladnmarks



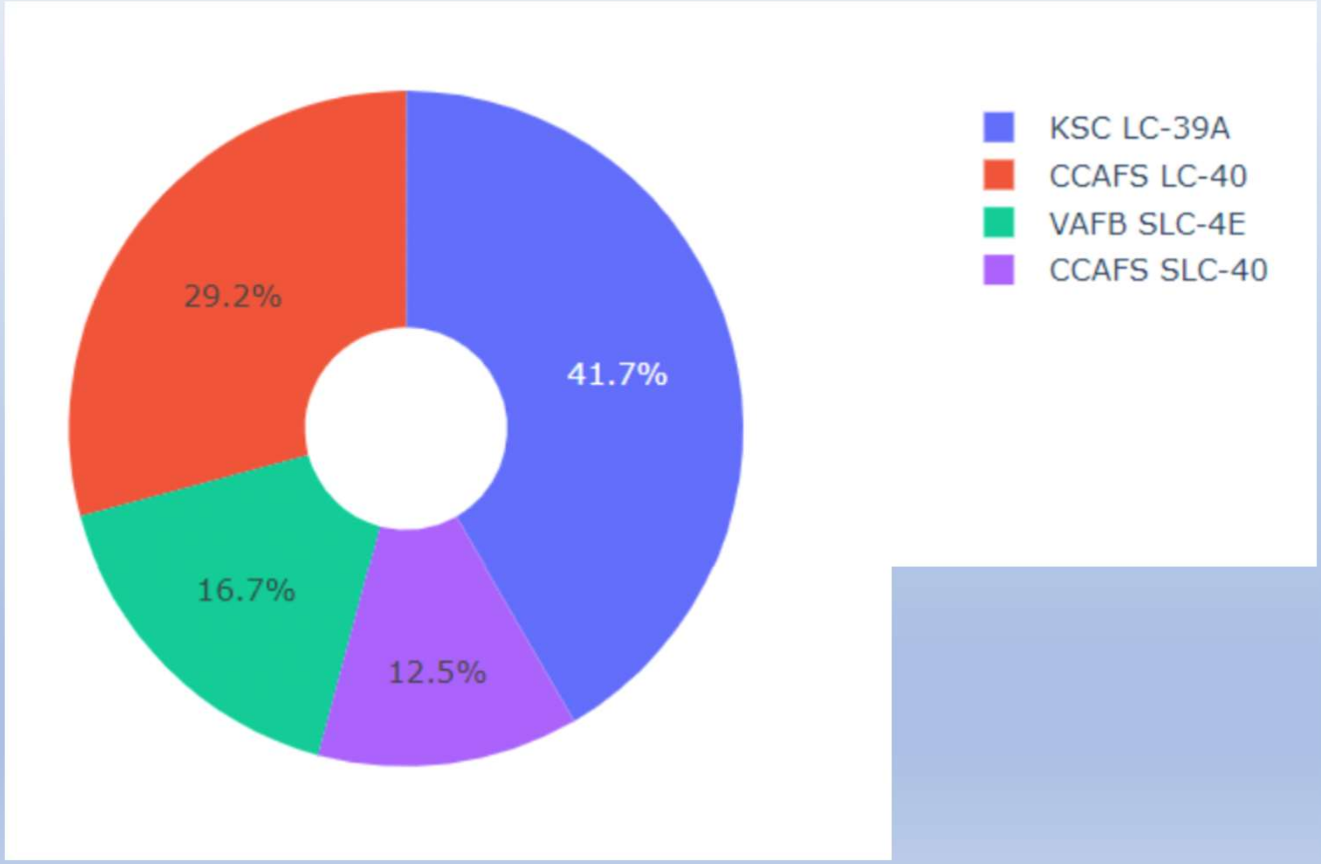
•Are launch sites in close proximity to railways? No
•Are launch sites in close proximity to highways? No
•Are launch sites in close proximity to coastline? Yes
•Do launch sites keep certain distance away from cities? Yes

Section 5

Build a Dashboard with Plotly Dash

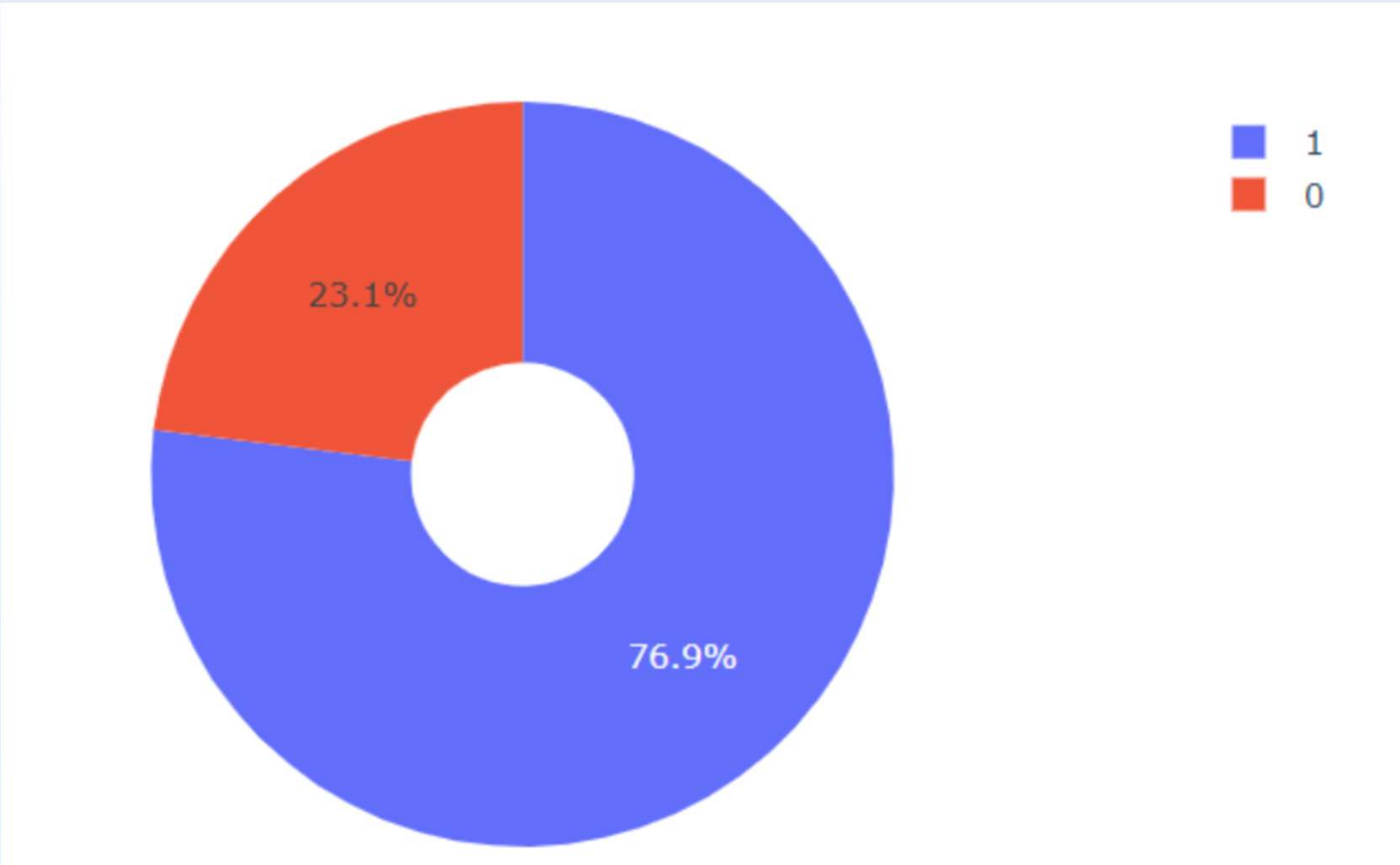


Success Percentage for each Launch Site



From this pie chart, we can determine that KSC LC-39A had the greatest percentage of successes.

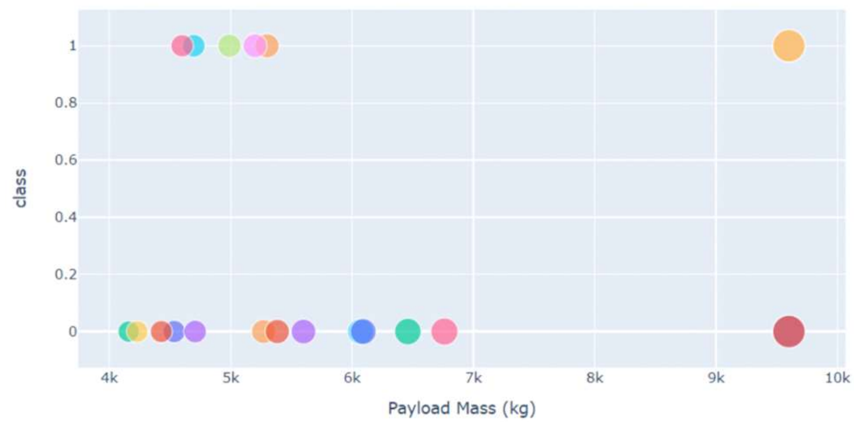
Success/Failure Percentages for Launch Site with the Most Successes



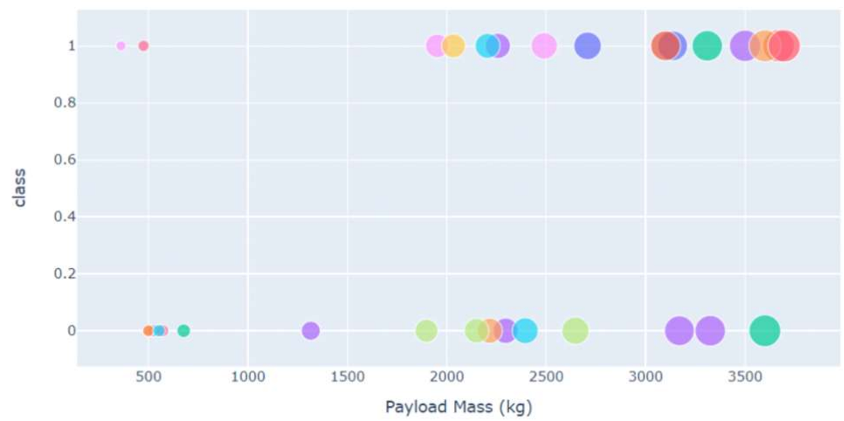
KSC LC-39A achieved a 76.9% success rate and a 23.1% failure rate

Payload Mass vs Launch Outcome scatter plots

- From these scatter plots, we can determine that the chance of success is greater for low weighted payloads compared to heavy weighted payloads.



Heavy Weighted Payload → 4000 to 10000



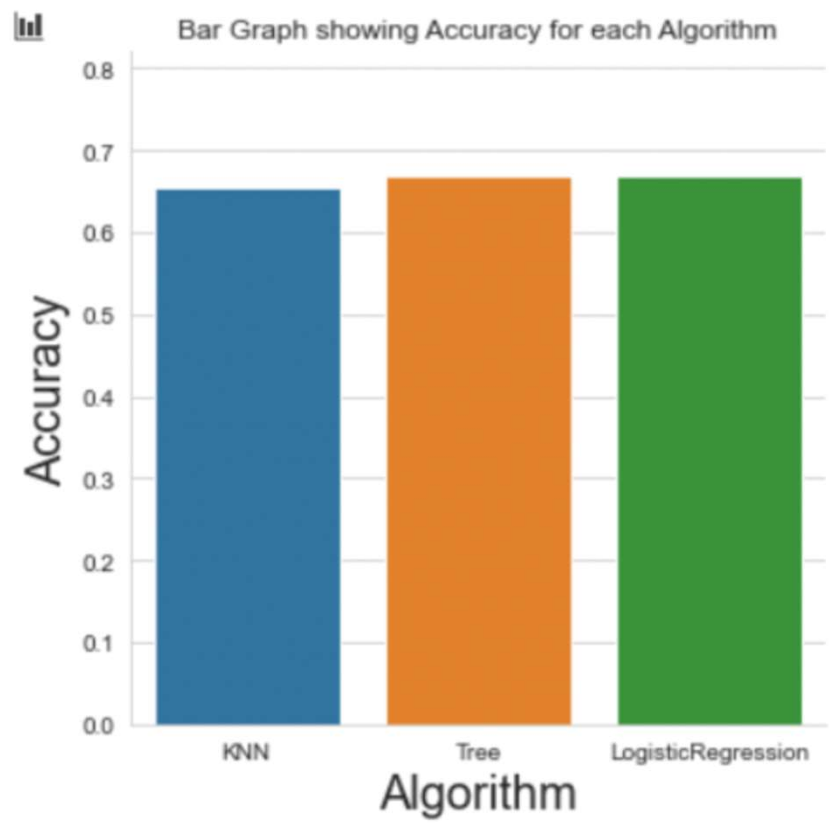
Low Weighted Payload → 0 to 4000

Section 6

Predictive Analysis (Classification)

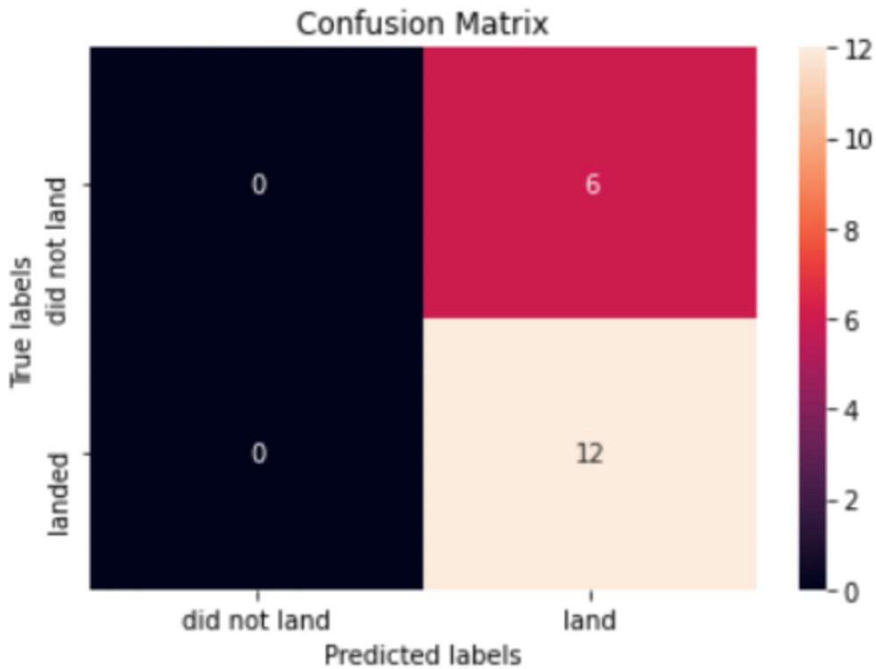
Classification Accuracy

- From the bar chart, the difference in accuracy is very small.
- However, the tree Algorithm is the best algorithm for this scenario.
- In addition, after selecting the best hyperparameters for the decision tree classifier using the validation data, 83.3% accuracy was reached.



Confusion Matrix

- Analyzing the confusion matrix, we can determine that the major problem that we are facing is false positives.



Conclusions

My Overall Conclusions:

- The best algorithm for machine learning for this particular dataset will be the Tree Classifier Algorithm.
- The launch site KSC LC-39A had the most successful launches.
- Low weighted payloads perform better than heavier payloads
- The success rate for SpaceX launches is directly proportional to time; in years, they will eventually perfect the launches
- Orbit ES-L1, GEO, HEO, and SSO have the best success rate


Thank you!