IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Manisha Barse
November 16, 2023

Outline

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

Executive Summary

- Summary of methodologies
  - Data Collection through WebScraping and REST API queries.
  - Data Wrangling to classify launches based on success and
  - Exploratory Data Analysis with SQL and Data Visualization
  - Build an interactive map to analyze the launch site proximity with Folium
  - Build a dashboard to analyze launch records interactively using Plotly Dash
  - Machine Learning to predict landing outcome using SVM, Classification Trees, Logistic Regression
- Summary of all results
  - Data analysis results, data visuals, interactive dashboards
  - Predictive model analysis results

Introduction

AIM: To predict if the Falcon 9 first stage will land successfully.

- 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.

Problems you want to find answers

- Determine if first stage of SpaceX Falcon9 will result in successful landing
- Impact of various parameters/variables like payload mass, launch site, booster version on outcome.
- Finding correlations between launch sites and success rates.

Section 1

Methodology

Methodology

Data collection	• SpaceX REST API • Webscraping Falcon9 and Falcon Heavy launch records from Wiki
Data wrangling	• Filtering data, handling missing values, • Apply One Hot Encoding for data fields for Machine Learning.
Exploratory data analysis	• Perform exploratory data analysis (EDA) using visualization and SQL • Plotting scatter, bar graphs to explore relationships between variables
Visualize	• Perform interactive visual analytics using Folium and Plotly Dash
Perform predictive analysis	• Build models, using classification models. • Tune and evaluate to find best model and parameters

Data Collection

SpaceX API

- Request data from SpaceX API
- Decode response using .json() then df using .json_normalize()
- Request information
- Create dictionary from data then df from it
- Filter df to contain Falcon9 launches
- Replace missing values of Payload by mean
- Export data to csv

Webscrapping

- Request Falcon9 launch data from Wiki
- Create BeautifulSoup object from HTML response
- Extract column names form HTML table header
- Collect data by parsing HTML tables
- Create dictionary from data
- Create dataframe from dictionary
- Export data to csv

Data Collection – SpaceX API

- Used get response to SpaceX API to collect, clean the requested data, performed some data wrangling

- Here is the GitHub URL of the completed SpaceX API calls notebook:

  [Github Link]

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

```python
# Use json_normalize meethod to convert the json result into a dataframe
df = response.json()
data = pd.json_normalize(df)
```

```python
# Call getBoosterVersion
getBoosterVersion(data)
```

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

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

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

```python
data_falcon9 = data[data['BoosterVersion'] != 'Falcon 1']
data_falcon9.loc[:,'FlightNumber'] = list(range(1, data_falcon9.shape[0]+1))
```

```python
# Calculate the mean value of PayloadMass column
mean_payload_mass = data_falcon9['PayloadMass'].mean()

# Replace the np.nan values with its mean value
data_falcon9['PayloadMass'].replace(np.nan, mean_payload_mass, inplace=True)
```

```python
data_falcon9.to_csv('dataset_part_1.csv', index=False)
```

Data Collection - Scraping

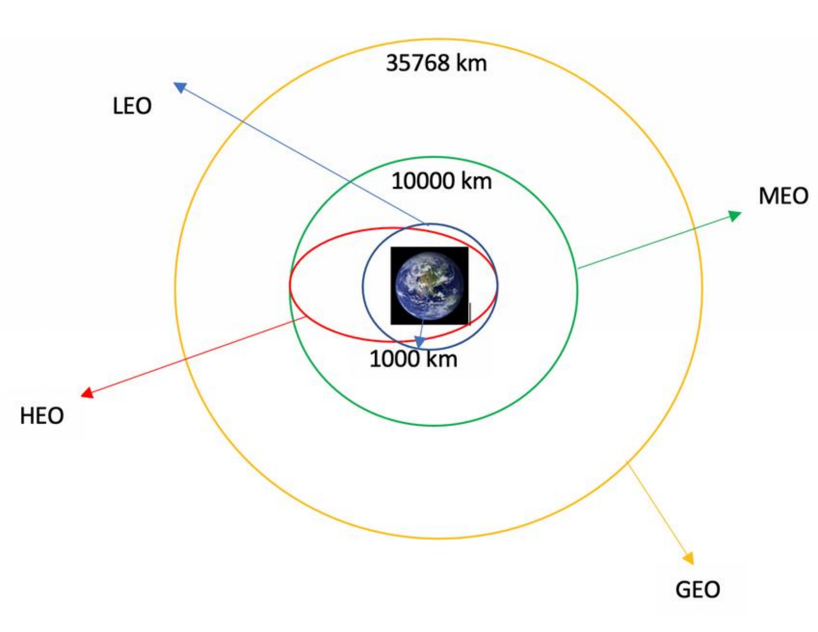
- Applied Webscrapping to Falcon9 launch data with BeautifulSoup
- Parsed html table data and converted to pandas dataframe
- Here is the GitHub URL of the completed web scraping notebook,

[Github Link]()

```
# use requests.get() method with the provided static_url
# assign the response to a object
response = requests.get(static_url)

# Use BeautifulSoup() to create a BeautifulSoup object from a response text content
soup = BeautifulSoup(response.text, 'html.parser')

# Use the find_all function in the BeautifulSoup object, with element type `table`
# Assign the result to a list called `html_tables`
html_tables = soup.find_all('table')

launch_dict= dict.fromkeys(column_names)

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

# 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['Customer'] = []
launch_dict['Launch outcome'] = []
# Added some new columns
launch_dict['Version Booster']=[]
launch_dict['Booster landing']=[]
launch_dict['Date']=[]
launch_dict['Time']=[]
df= pd.DataFrame({ key:pd.Series(value) for key, value in launch_dict.items() })

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

Data Wrangling

- Performed Exploratory data analysis and determined training labels.
- Calculated
  - # of launches at each site
  - # and occurrences of each orbit
  - # and occurrences of mission outcome per orbit type
- Created landing outcome label from Outcome column
- Export data as csv
- Worked out success rate for every landing in the dataset
- Here is the [Github Link]()



EDA with Data Visualization

Charts:

- Scatter plots
  - Flight Number vs. Payload
  - Flight Number vs. Launch Site
  - Payload Mass (kg) vs. Launch Site
  - Payload Mass (kg) vs. Orbit type
  - Orbit vs. Flight Number
  - Orbit vs. Payload Mass (kg)

  uses dots to represent values for two different numeric variables and show their relationship.

- Bar Charts
  - Mean vs. Orbit

  compare categories, display summary values, and understand relationships between categorical variables.

- Line Graph
  - Success rate vs. Year

  a graphical representation of information that changes over time.

- Here is the GitHub Link of EDA with visualization

EDA with SQL

- Loaded SpaceX dataset into a PostgreSQL

Queries:

- The names of unique launch sites in the space mission.
- The total payload mass carried by boosters launched by NASA (CRS)
- The average payload mass carried by booster version F9 v1.1
- The total number of successful and failure mission outcomes
- The failed landing outcomes in drone ship, their booster version and launch site names.

List:

- Date of first successful landing on ground pad
- Names of boosters which had success landing on drone ship and have payload mass greater than 4,000 but less than 6,000
- Total number of successful and failed missions
- Names of booster versions which have carried the max payload
- Failed landing outcomes on drone ship, their booster version and launch site for the months in the year 2015
- Count of landing outcomes between 2010-06-04 and 2017-03-20 (desc) 2023

- Here is the GitHub Link of completed EDA with SQL notebook

Build an Interactive Map with Folium

- Markers Indicating Launch Sites
  - Added blue circle at NASA Johnson Space Center's coordinate with a popup label showing its name using its latitude and longitude coordinates.
  - Added red circles at all launch sites coordinates with a popup label showing its name using its name using its latitude and longitude coordinates.
- Colored Markers of Launch Outcomes
  - Added colored markers of successful (1, green) and unsuccessful (0, red) launches at each launch site to show which launch sites have high success rates.
- Distances Between a Launch Site to Proximities
  - Added colored lines to show distance between launch site CCAFS SLC- 40 and its proximity to the nearest coastline, railway, highway, and city.
  - Answered questions like: - are launch sites near coastlines? Yes
    - are launch sites near railways? No
    - are launch sites near highways? No
    - do these keep certain distance from cities? Yes

- Here is the [GitHub Link]of your completed interactive map with Folium map

Build a Dashboard with Plotly Dash

- Built an interactive dashboard with Plotly dash
- Created Dropdown menu with launch sites to allow user to select all or a certain launch site.
- Plotted pie charts showing the total launches by a certain sites to see successful or unsuccessful launches as a percent of the total.
- Plotted scatter graph showing the correlation with Outcome and Payload Mass (Kg) for the different booster version.
- Here is the GitHub Link of your completed Plotly Dash lab

Predictive Analysis (Classification)

Building Model

- Load our dataset into numpy and pandas transform data
- Split our data into training and test data sets
- Check how many test samples we have
- Decide which type of machine learning algorithms we want to use
- Set our parameters and algorithms to GridSearchCV
- Fit our datasets into the GridSearchCV objects and train our dataset.

- 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
  - In the notebook there is a dictionary of algorithms with scores at the bottom of the notebook

- Here is the Github Link of completed predictive analysis lab

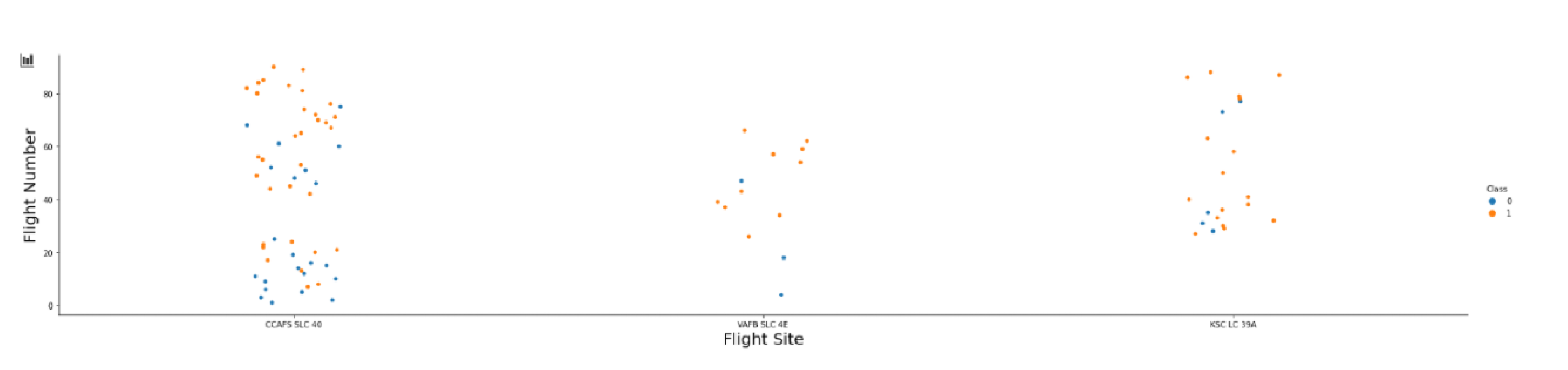
Results

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

Section 2

Insights drawn from EDA

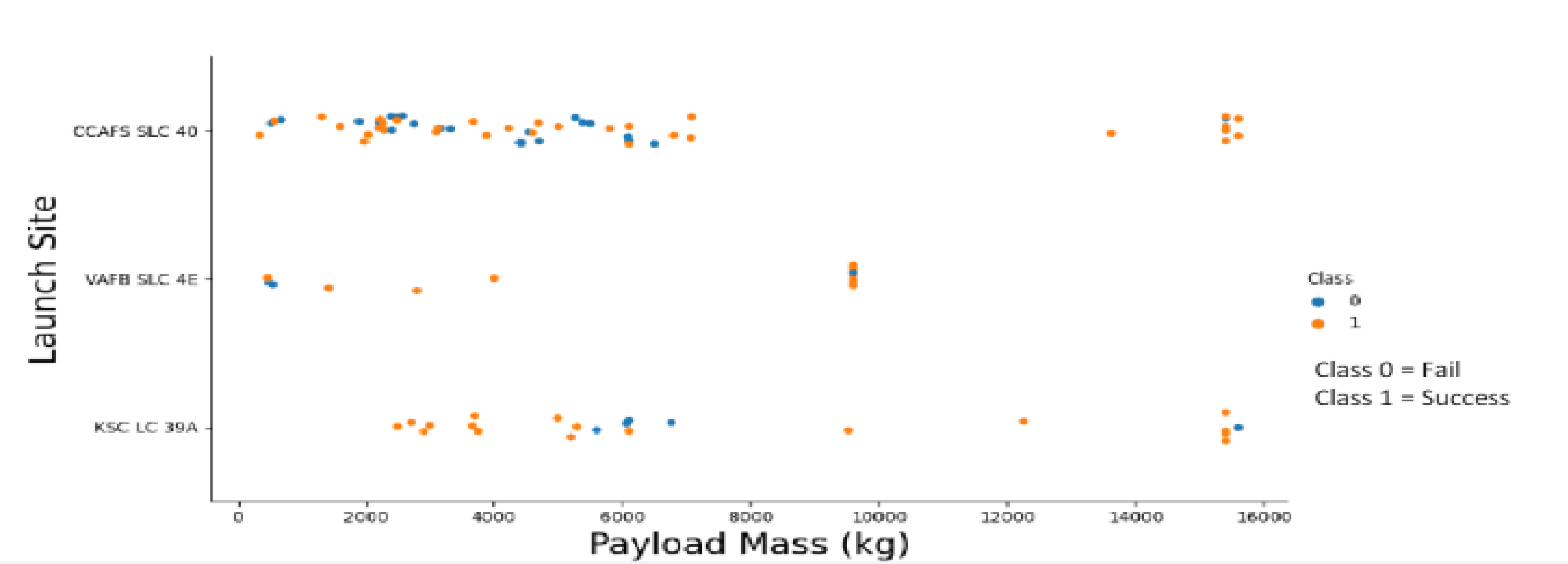
Flight Number vs. Launch Site



Earlier flights had a lower success rate (blue = fail) , Later flights had a higher success rate (orange = success)

The more amount of flights at a launch site -> greater the success at a launch site

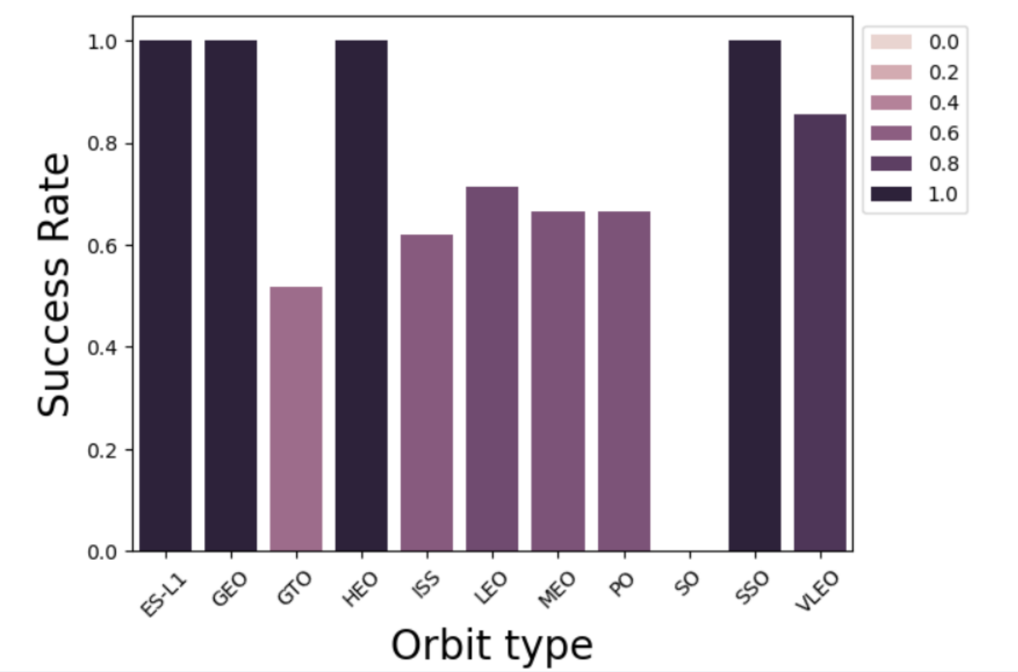
Payload vs. Launch Site



The greater the payload mass for Launch Site CCAFS SLC 40 -> the higher the success rate for the Rocket.

There is not quite a clear pattern to be found using this visualization to make a decision if the Launch Site is dependent on Payload Mass for a success launch.

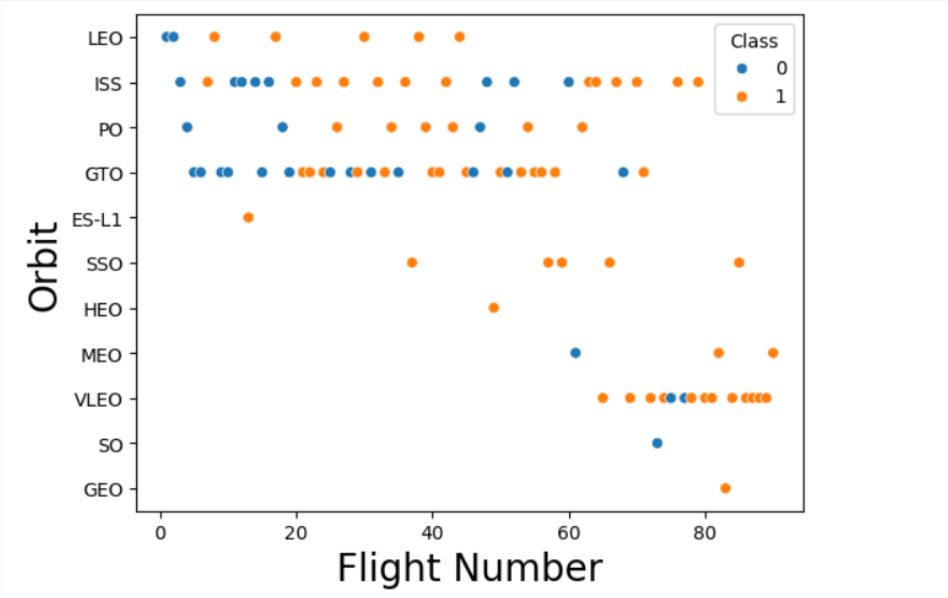
Success Rate vs. Orbit Type



Orbit ES-11, GEO, HEO, SSO have the best Success Rate

Orbit SO have 0 success rate.

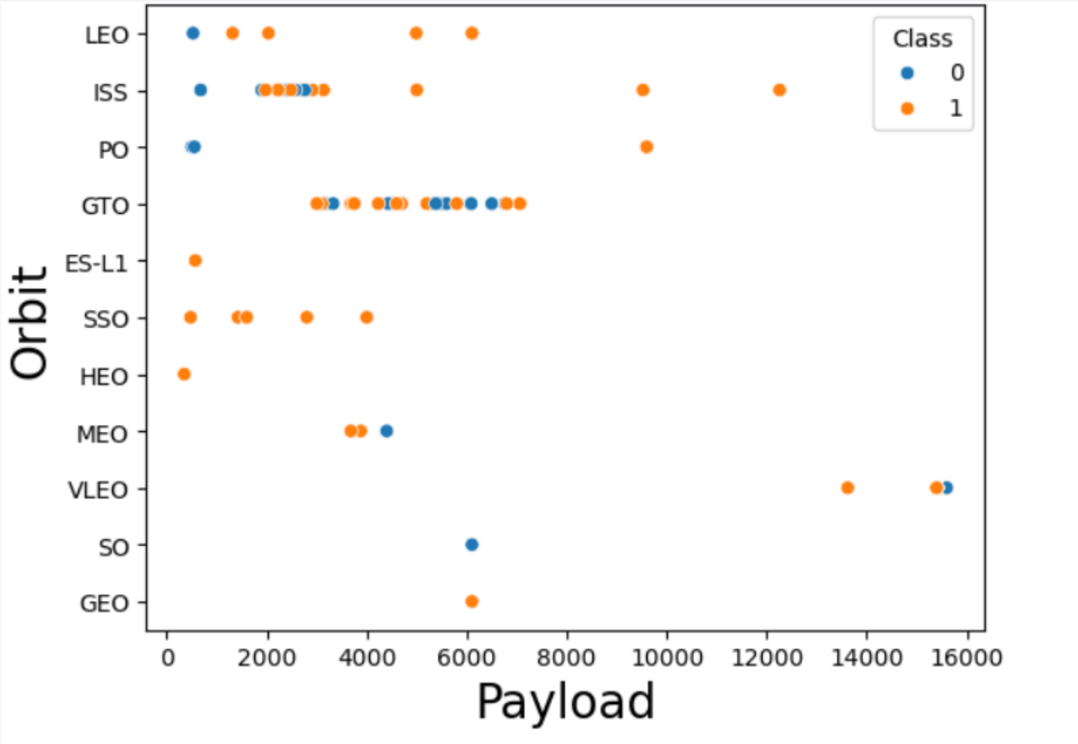
Flight Number vs. Orbit Type



In the LEO orbit the Success appears related to the number of flights

There seems to be no relationship between flight number when in GTO orbit.

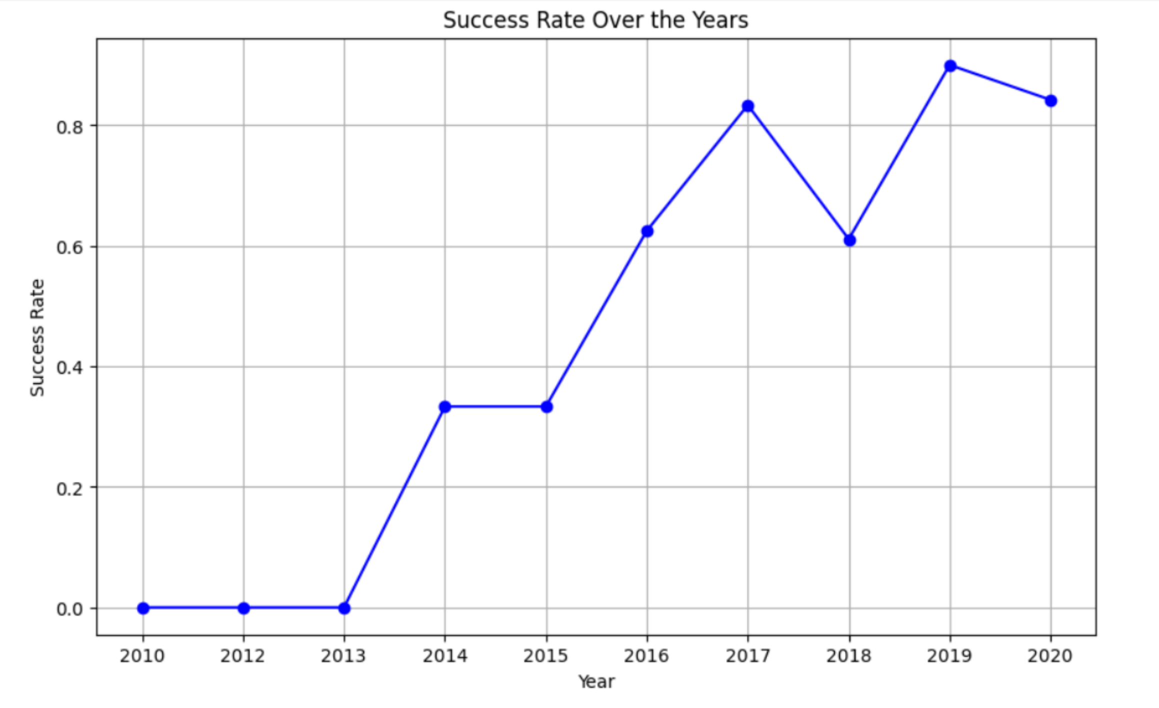
Payload vs. Orbit Type



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

For GTO, we cannot distinguish, both positive landing rate and negative landing(unsuccessful mission) present

Launch Success Yearly Trend



Success rate since 2013 kept increasing till 2020

All Launch Site Names

Task 1

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

In [8]:
```
%sql SELECT DISTINCT "Launch_Site" FROM SPACEXTBL;
```

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

Out[8]:

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

Launch Site Names Begin with 'CCA'

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

In [9]:
```
%sql SELECT * FROM SPACEXTABLE WHERE "Launch_Site" LIKE 'CCA%' LIMIT 5;
```

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

Out[9]:

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Ou
6/4/2010	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (para
12/8/2010	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 (para
22/05/2012	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No a
10/8/2012	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No a
3/1/2013	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No a

Total Payload Mass

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

In [10]:

```
%%sql
SELECT SUM("PAYLOAD_MASS__KG_") AS 'Total Payload Mass'
FROM SPACEXTABLE
WHERE "Customer" = 'NASA (CRS)';
```

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

Out[10]:

Total Payload Mass
45596

Average Payload Mass by F9 v1.1

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

In [11]:

```
%%sql
SELECT AVG("PAYLOAD_MASS__KG_") AS 'Average Payload Mass'
FROM SPACEXTABLE
WHERE "Booster_Version" = 'F9 v1.1';
```

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

Out[11]:

Average Payload Mass
2928.4

First Successful Ground Landing Date

```
In [12]: %%sql
SELECT MIN("Date") AS 'FirstSuccessfulLandingDate'
FROM SPACEXTABLE
WHERE "Landing_Outcome" = 'Success (ground pad)';
```

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

Out[12]:

FirstSuccessfulLandingDate
1/8/2018

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

In [13]:
```
%%sql
SELECT "Booster_Version" AS 'Booster_Name'
FROM SPACEXTABLE
WHERE "Landing_Outcome" = 'Success (drone ship)'
  AND "PAYLOAD_MASS__KG_" > 4000
  AND "PAYLOAD_MASS__KG_" < 6000;
```

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

Out[13]:

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

Total Number of Successful and Failure Mission Outcomes

- Calculate the total number of successful and failure mission outcomes

In [14]:
```
%%sql
SELECT "Mission_Outcome", COUNT(*) AS 'Outcome_Count'
FROM SPACEXTABLE
WHERE "Mission_Outcome" IN ('Success', 'Failure')
GROUP BY "Mission_Outcome";
```

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

Out[14]:

Mission_Outcome	Outcome_Count
Success	98

Boosters Carried Maximum Payload

- List the names of the booster which have carried the maximum payload mass

```
In [19]: %%sql
SELECT "Booster_Version" AS 'Booster_Name'
FROM SPACEXTABLE
WHERE "PAYLOAD_MASS__KG_" = (
    SELECT MAX("PAYLOAD_MASS__KG_") FROM SPACEXTABLE
);

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

Out[19]:

Booster_Name
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

In [25]:

```
%%sql
SELECT
    CASE substr(Date, 6, 2)
        WHEN '01' THEN 'January'
        WHEN '02' THEN 'February'
        WHEN '03' THEN 'March'
        WHEN '04' THEN 'April'
        WHEN '05' THEN 'May'
        WHEN '06' THEN 'June'
        WHEN '07' THEN 'July'
        WHEN '08' THEN 'August'
        WHEN '09' THEN 'September'
        WHEN '10' THEN 'October'
        WHEN '11' THEN 'November'
        WHEN '12' THEN 'December'
    END AS "Month",
    "Landing_Outcome",
    "Booster_Version",
    "Launch_Site"
FROM SPACEXTABLE
WHERE substr(Date, 0, 5) = '2015'
    AND "Landing_Outcome" = 'Failure (drone ship)';
```

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

Out[25]:

Month	Landing_Outcome	Booster_Version	Launch_Site

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

In [26]:
```
%%sql
SELECT "Landing_Outcome", COUNT(*) AS "OutcomeCount"
FROM SPACEXTABLE
WHERE "Date" BETWEEN '2010-06-04' AND '2017-03-20'
GROUP BY "Landing_Outcome"
ORDER BY "OutcomeCount" DESC;
```

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

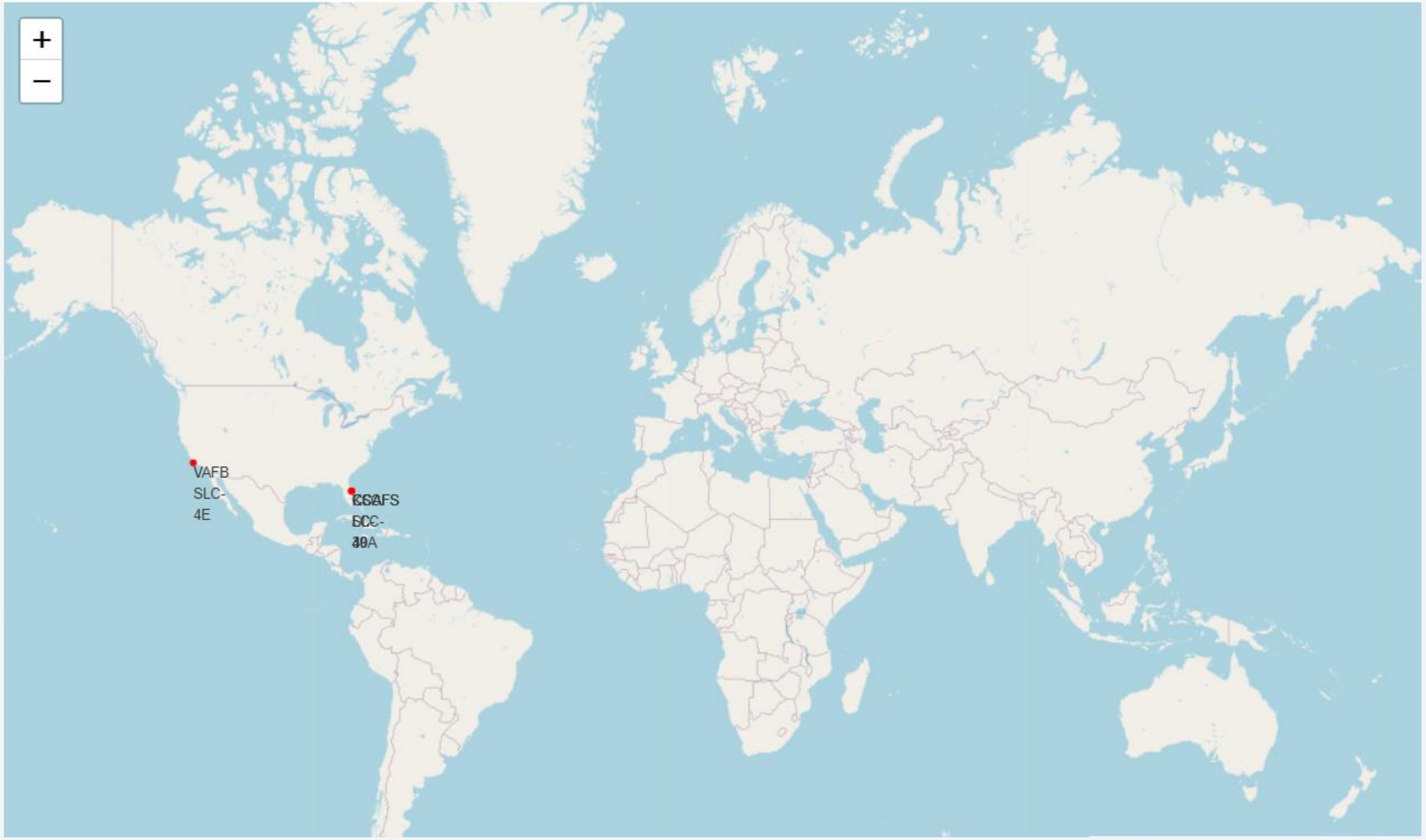
Out[26]:

Landing_Outcome	OutcomeCount

Section 3

Launch Sites Proximities Analysis

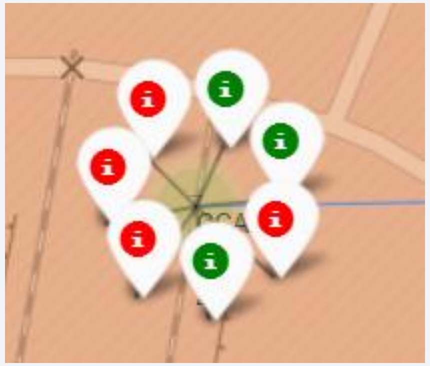
All Launch Sites Global Markers

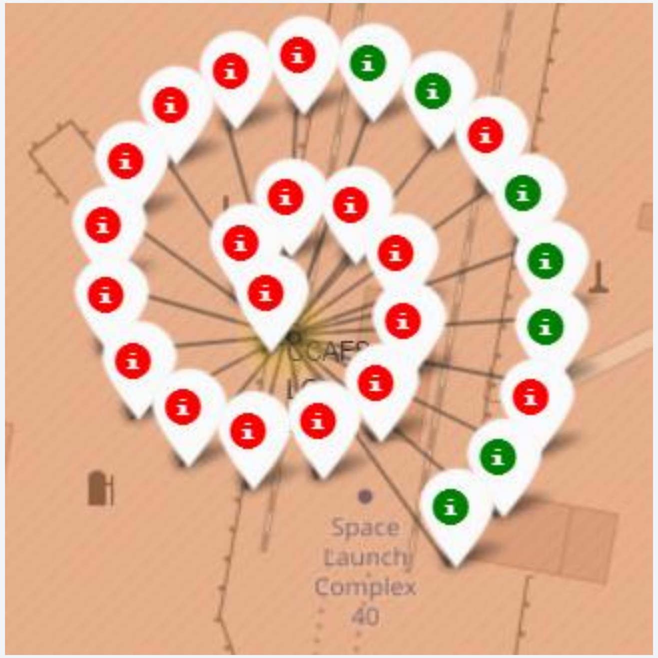


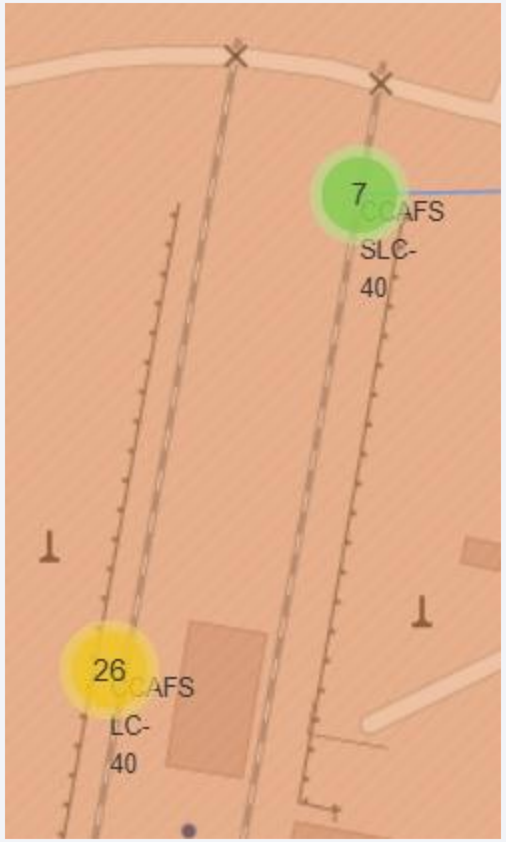
All sites located in the USA coasts- California and Florida

Colored Labelled Markers

CCAFS LC 40







KSC LC 39A

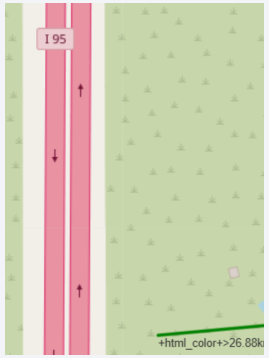
Florida Launch Sites

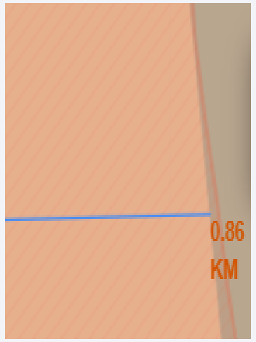
Green Marker -> successful launches
Red Marker -> Failure

VAFB SLC 4E

California Launch Sites

Launch Site distances to Landmarks





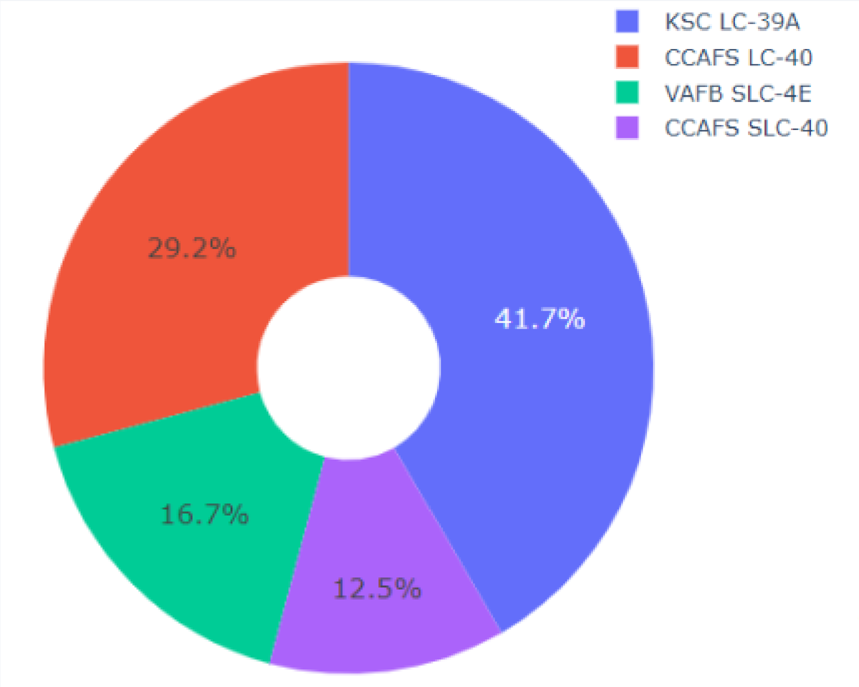
•Are launch sites in close proximity to railways? No
•Are launch sites in close proximity to highways? No
•Are launch sites in close proximity to coastline? Yes
•Do launch sites keep certain distance away from cities? Yes

Section 4

Build a Dashboard with Plotly Dash

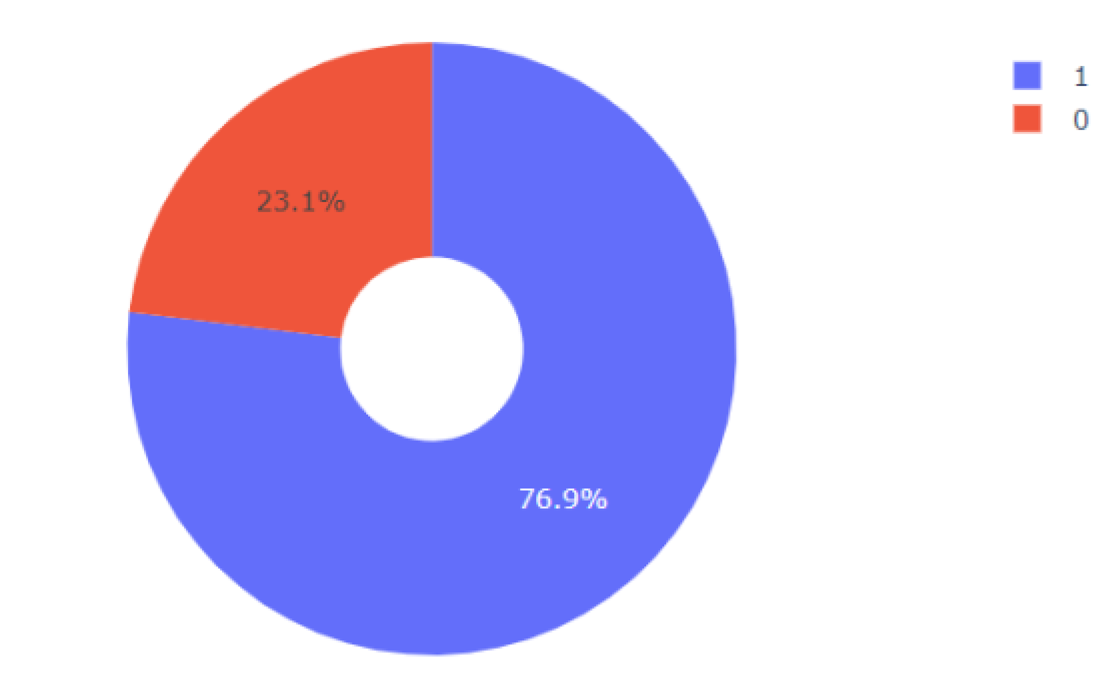
Dashboard Piechart

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



Dashboard

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



Section 5

Predictive Analysis (Classification)

Classification Accuracy

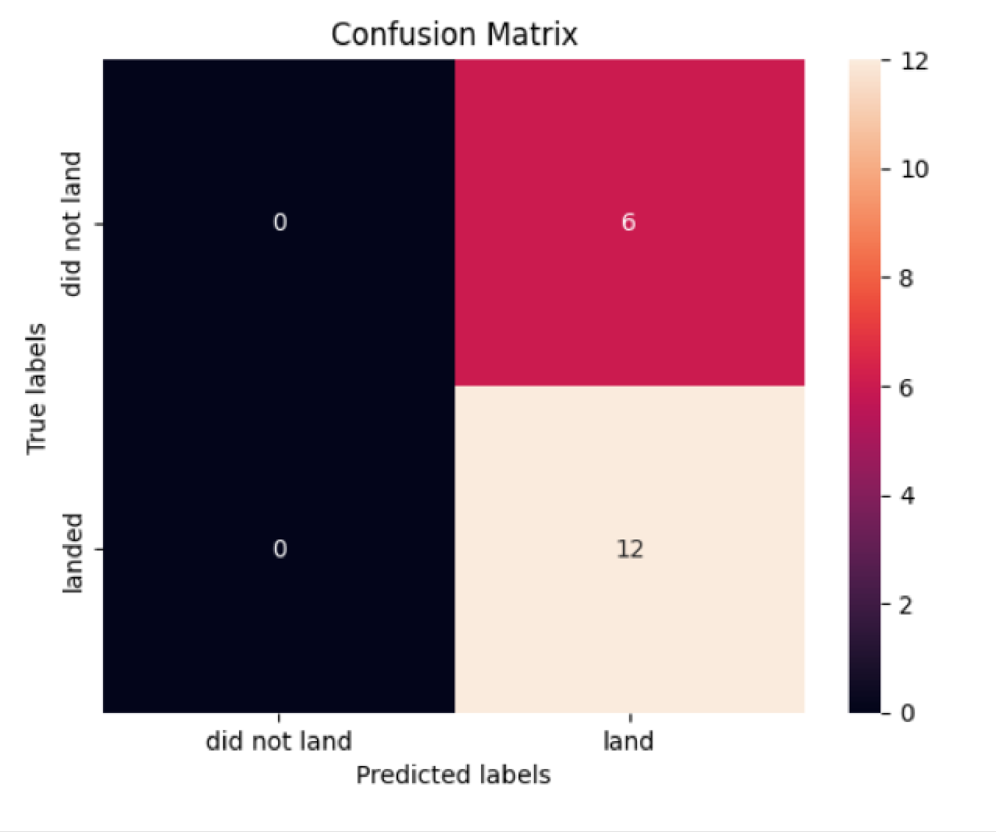
	ML Method	Accuracy (%)
0	SVM	84.821
1	Logistic Regression	84.642
2	KNN	84.821
3	Tree	89.107

Tree wins!

Confusion Matrix

- Confusion matrix of Tree algorithm
  - Tree can distinguish between different classes
  - There is a problem in false positives

		Predicted Class	
		P	N
Actual Class	P	True Positives (TP)	False Negatives (FN)
	N	False Positives (FP)	True Negatives (TN)



Conclusions

- Orbits GEO,HEO,SSO,ES-L1 has the best Success Rate
- KSC LC-39A had the most successful launches from all the sites
- Low weighted payloads performed better than the heavier payloads
- The success rates for SpaceX launches is directly proportional time in years.
- The Tree Classifier Algorithm is the best for Machine Learning algorithm for this dataset.

Thank you!