IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Austin
<Date>

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 a Dashboard with Plotly Dash
  - Predictive analysis (Classification)

Summary of all results

Exploratory data analysis results

Interactive analytics demo in screenshots

Predictive analysis results

Introduction

- Project background and context

Space X has advertised their Falcon 9 rocket launches to be inexpensive as compared to other business to be approximately 62 million dollars. With the determination the first stage of the aircraft's landing successfully, the company will then be able to determine the expenses for a launch. With the presentation of such information, it can also be used in the hind set of expenses but also the company's other sources such as man-power, time, and availability to produce future projects. The goal of the project is to create a machine learning pipeline to be able to predict if the first stage of the Falcon 9 to be successful and considerations of additional factors to be recognized alongside such success.

- Problems you want to find answers

1. Review the interaction amongst various features of the aircraft that determine the success rate of a successful landing.
2. Consider of operational conditions that may be required to ensure successful landing.
3. Determine and review what factors will affect and ensure the Falcon 9 a successful landing.

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Data was collected via SpaceX's API and web scrapping from Wikipedia sources.
- Perform data wrangling
  - One-hot encoding was applied to implement categorical features and review of items
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - How to build, tune, evaluate classification models

Data Collection

- The data was collected through various methods, such as:
  - Use of requests to SpaceX API
    - Get Booster Version
    - Get Launch Site Data
    - Get Payload Data
    - Get Core Data
- Webscraping of SpaceX Wikipedia Page
  - HTML requests (HTTP-Get)
  - BeautiflSoup (Python package)
  - Extraction of column names from HTML tables, parsing table and conversion to panda dataframes

Data Collection – SpaceX API

- Use of GET request to SpaceX API interface website to access collection of data, parse of data and extraction of necessary data via implementation of data wrangling and formatting for Falcon 9 alone due to case study.

- Access to notebook: https://github.com/JAustin5/IBM-Data-Science-Capstone/blob/main/jupyter-labs-spacex-data-collection-api.ipynb

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

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

Check the content of the response

```
#print(response.content)
```

You should see the response contains massive information about SpaceX launches. Next, let's try to discover some more relevant information for this project.

Task 1: Request and parse the SpaceX launch data using the GET request

To make the requested JSON results more consistent, we will use the following static response object for this project:

```
static_json_url='https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DS0321EN-SkillsNetwork/datasets/API
```

We should see that the request was successfull with the 200 status response code

```
response.status_code
```

```
200
```

Now we decode the response content as a Json using `.json()` and turn it into a Pandas dataframe using `.json_normalize()`

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

Using the dataframe `data` print the first 5 rows

```
# Get the head of the dataframe
data.head(5)
```

Data Collection - Scraping

- Applied webscrapping practices to retrieve Falcon 9 launch records via HTTP requests to SpaceX Wikipedia and implementation of parsing data into Panda library data frame with BeautifulSoup.

- Access to notebook: https://github.com/JAustin5/IBM-Data-Science-Capstone/blob/main/jupyter-labs-webscraping.ipynb

First, let's perform an HTTP GET method to request the Falcon9 Launch HTML page, as an HTTP response.

In [6]:
```
# use requests.get() method with the provided static_url
# assign the response to a object
req = requests.get(static_url)
data = req.text
```

Create a `BeautifulSoup` object from the HTML `response`

In [8]:
```
# Use BeautifulSoup() to create a BeautifulSoup object from a response text content
soup = BeautifulSoup(data, "html.parser")
```

Print the page title to verify if the `BeautifulSoup` object was created properly

In [9]:
```
# Use soup.title attribute
print(soup.title)
```

```
<title>List of Falcon 9 and Falcon Heavy launches - Wikipedia</title>
```

TASK 2: Extract all column/variable names from the HTML table header

Next, we want to collect all relevant column names from the HTML table header

Let's try to find all tables on the wiki page first. If you need to refresh your memory about `BeautifulSoup`, please check the external reference link towards the end of this lab

In [12]:
```
# 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')
```

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

In [17]:
```
# Let's print the third table and check its content
first_launch_table = html_tables[2]
#print(first_launch_table)
```

You should able to see the columns names embedded in the table header elements `<th>` as follows:

Data Wrangling

- Implementation of exploratory data analysis and determining training labels for best fit.
- Calculation of number of launches at each site alongside number and occurrences of each orbits.
- Dealt with any missing values with column payload mass, and replaced such values (as are essential) with mean values
- Creation of landing outcome labels from outcome column and exporting results to csv after practice.
- Link to notebook: https://github.com/JAustin5/IBM-Data-Science-Capstone/blob/main/labs-jupyter-spacex-Data%20wrangling.ipynb

Use the method `.value_counts()` to determine the number and occurrence of each orbit in the column `Orbit`

```
# Apply value_counts on Orbit column
df['Orbit'].value_counts()
```

```
GTO      27
ISS      21
VLEO     14
PO        9
LEO       7
SSO       5
MEO       3
ES-L1     1
HEO       1
SO        1
GEO       1
Name: Orbit, dtype: int64
```

TASK 3: Calculate the number and occurence of mission outcome of the orbits

Use the method `.value_counts()` on the column `Outcome` to determine the number of `landing_outcomes`.Then assign it to a variable landing_outcomes.

```
# landing_outcomes = values on Outcome column
landing_outcomes = df['Outcome'].value_counts()
landing_outcomes
```

```
True ASDS      41
None None      19
True RTLS      14
False ASDS      6
True Ocean      5
False Ocean     2
None ASDS       2
False RTLS      1
Name: Outcome, dtype: int64
```

EDA with Data Visualization

- Charts used:
  - Flight Number vs. Launch Site: Shows relationship of flight number and launch sites to lead to review for the best success rate
  - Payload Mass vs. Launch Site: Shows relationship of payload mass and launch site to which present best fit for success rate
  - Success Rate vs. Orbit Type: Shows relationship of orbit types that have certain success rates; indicating which is the best to consider for a successful landing
  - Flight Number vs. Orbit Type: Shows relationship of the success rate for certain flight numbers and the types of orbits
  - Payload Mass vs. Orbit Type: Shows relationship of the success rate for payload mass clusters and orbit types
  - Success rate vs. Year: Shows correlation between success development over time
- Notebook: https://github.com/JAustin5/IBM-Data-Science-Capstone/blob/main/edadataviz%20(2).ipynb

EDA with SQL

- SQL queries performed:
  - Extraction of list of all launch sites
  - Display of names of unique launch sites in the space mission
  - Display of 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 names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000
  - List total number of successful and failure mission outcomes
  - List names of the booster_version which have carried the maximum payload mass
  - List the records of failure landing_outcomes in drone ship, booster versions, launch_site for the months in year 2015
  - Rank count of landing outomces (Failure (drone ship) or Success (ground pad)) between the date 2010-06-04 and 2017-03-20, in descending order.
- Notebook: https://github.com/JAustin5/IBM-Data-Science-Capstone/blob/main/jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

- Creation of maps marked with launch sites and added objects such as markers, circles, lines to mark successful or failed launches for each site.
- Assigned feature launch outcomes as 0 (failure) or 1 (success)
- Used color-labeled marker clusters to easier identify launch sites level of success rate, i.e., higher success rates
- Calculation of distance between launch site as proximities near other objects such as:
  - Launch sites that may be near railways, highways and coastlines
  - Launch site location in relation to distance from cities
- Notebook: https://github.com/JAustin5/IBM-Data-Science-Capstone/blob/main/lab_jupyter_launch_site_location.ipynb

Build a Dashboard with Plotly Dash

- Input elements:
  - Dropdown list for allowance of selection of launch site
  - RangeSlider for selecting the payload mass
- Output elements:
  - Pie chart: show success rate for each launch site, or showing number of successful landing outcomes (if all sites selected)
  - Scatter plot: display of success or failure by payload and booster version
- Notebook: https://github.com/JAustin5/IBM-Data-Science-Capstone/blob/main/spacex_dash_app.py

Predictive Analysis (Classification)

- Pre-processing
  - Onehot-encoding for categorical features
  - Split data into dependent and independent variables and train and test data
  - Scale data with StandardScaler
- Model Building
  - Logistic Regression
  - Support Vector Machine
  - Decision Tree
  - K-Nearest Neighbor
- Optimization
  - Use of GridSearch for optimizing the models based on their hyperparameters
- Evaluation
  - Use of accuracy of GridSearch for selection of best parameter
  - Use score to compare each classification method

Notebook
https://github.com/JAustin5/IBM-Data-Science-Capstone/blob/main/SpaceX_Machine%20Learning%20Prediction_Part_5.ipynb

Results

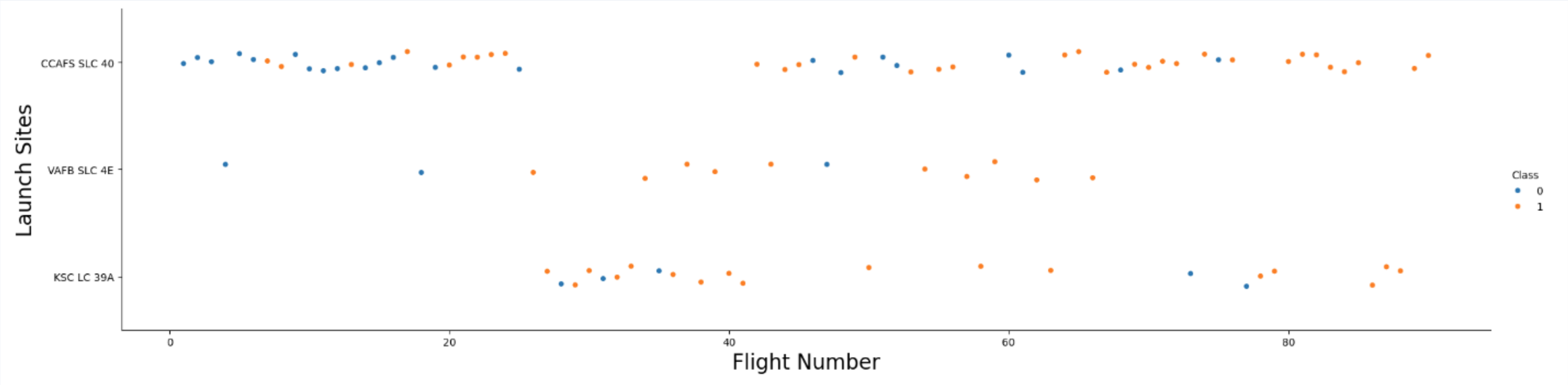
- Exploratory data analysis results
  - Launch success rate increases over time
  - Higher success rate for those with higher orbits
- Interactive analytics demo in screenshots
  - Higher success rate for higher payload mass
  - Low success rate for earlier booster versions
  - Higher success rate for Kennedy Space center and those recently started at Cape Canaveral
- Predictive analysis results
  - Best prediction results with Logistic Regression and Support Vector Machine (SVM)

Section 2

Insights drawn from EDA

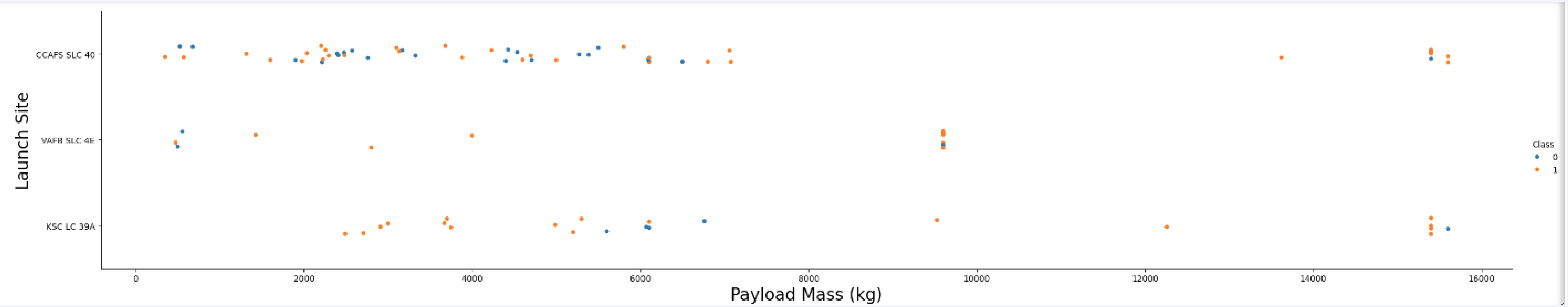
Flight Number vs. Launch Site

- From the plot, the larger the flight amount at a launch site, the greater the success rate at a launch site will be.



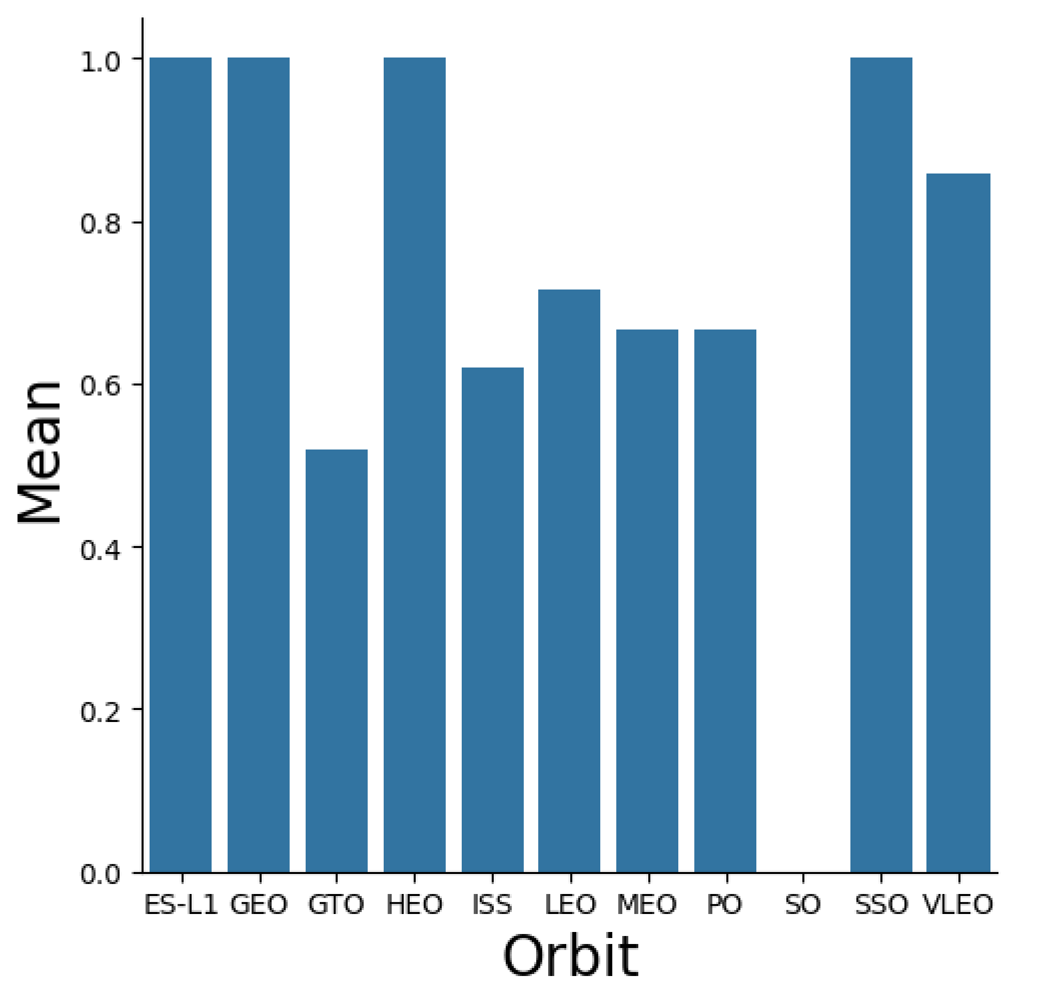
Payload vs. Launch Site

- After reviewing the data, the greater the payload mass for launch site for CCAFS SLC 40 the higher the success rate for the rocket is.



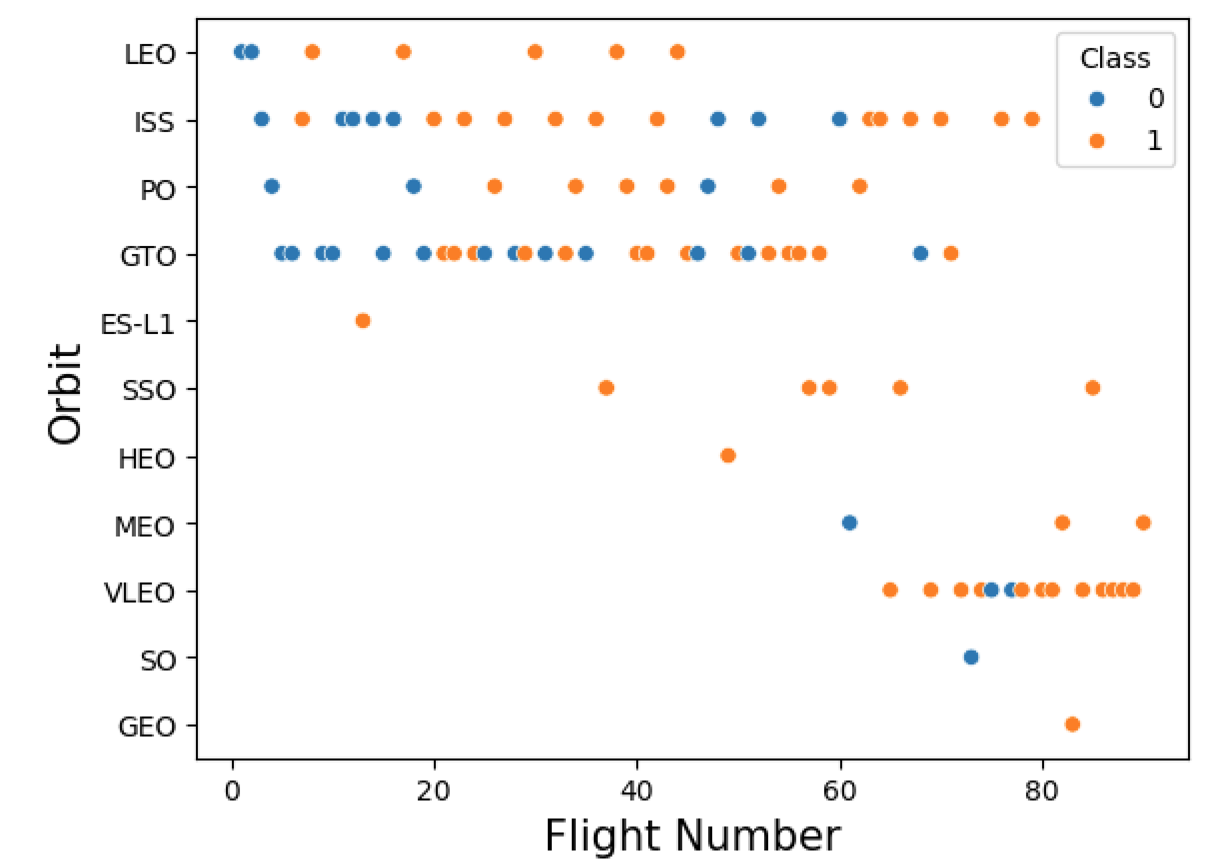
Success Rate vs. Orbit Type

- From this bar chart's display, one can infer that ES-L1, GEO, HEO, SSO had the most successful rate (can also include VLEO).



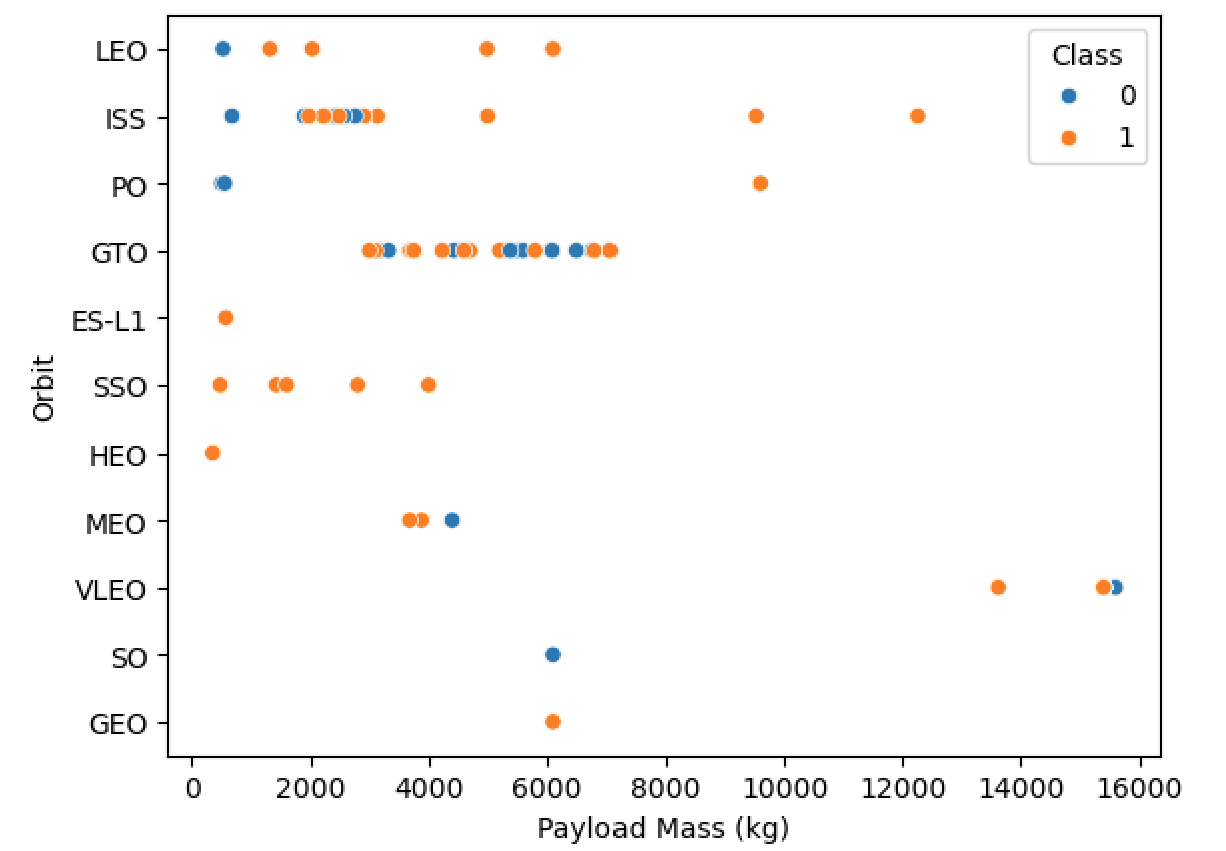
Flight Number vs. Orbit Type

- The scatter plot presents the LEO orbit has a well success number of flights as opposed to the GEO orbit that has no relationship between flight numbers and the orbit to represent it to be a success.



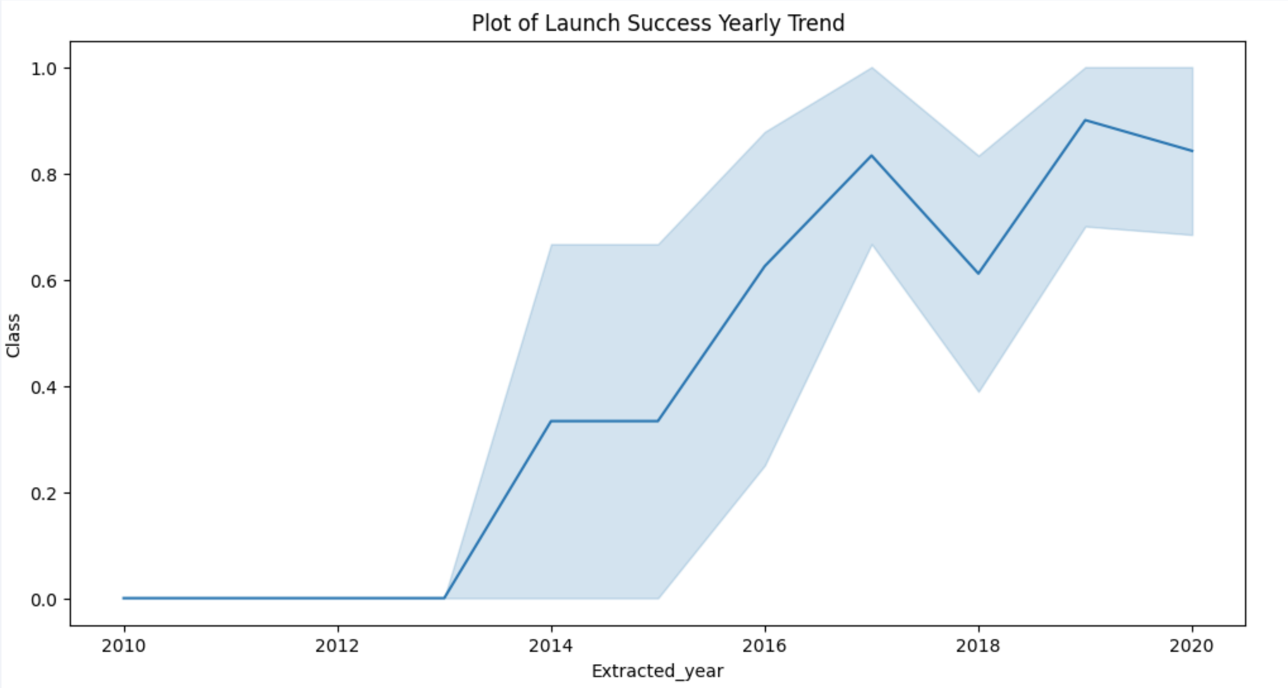
Payload vs. Orbit Type

- One can see that with heavy payloads, the successful landing is more common within the PO, LEO and ISS orbits.



Launch Success Yearly Trend

- One can see that the progression of success rates have progress since 2013 with a continuous successes till 2020.



All Launch Site Names

- The names of the unique launch sites

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

Launch Site Names Begin with 'CCA'

- 5 records where launch sites begin with `CCA`

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
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	Success	Failure (parachute)
2012-05-22	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
2012-10-08	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
2013-03-01	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

Total Payload Mass

- The calculated total payload carried by boosters from NASA

sum
45596

Average Payload Mass by F9 v1.1

- Calculation of the average payload mass carried by booster version F9 v1.1

Average
2928.4

First Successful Ground Landing Date

- The date(s) of the first successful landing outcome on ground pad

Date
2010-06-04

Successful Drone Ship Landing with Payload between 4000 and 6000

- List of the name of the boosters of which had been successful in drone ship and had a payload mass greater than 4000 but less than 6000

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

Total Number of Successful and Failure Mission Outcomes

- The calculation of the total number of successful and failure mission outcomes

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

Boosters Carried Maximum Payload

- The list of names of the booster which have carried the maximum payload mass

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

- The list of failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015

Month	Date	Landing_Outcome	Booster_Version	Launch_Site
01	2015-01-10	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
04	2015-04-14	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

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

- The rank count of landing outcomes (such as Failure (drone ship) or Success (ground pad)) between the date 2010-06-04 and 2017-03-20, presented in descending order

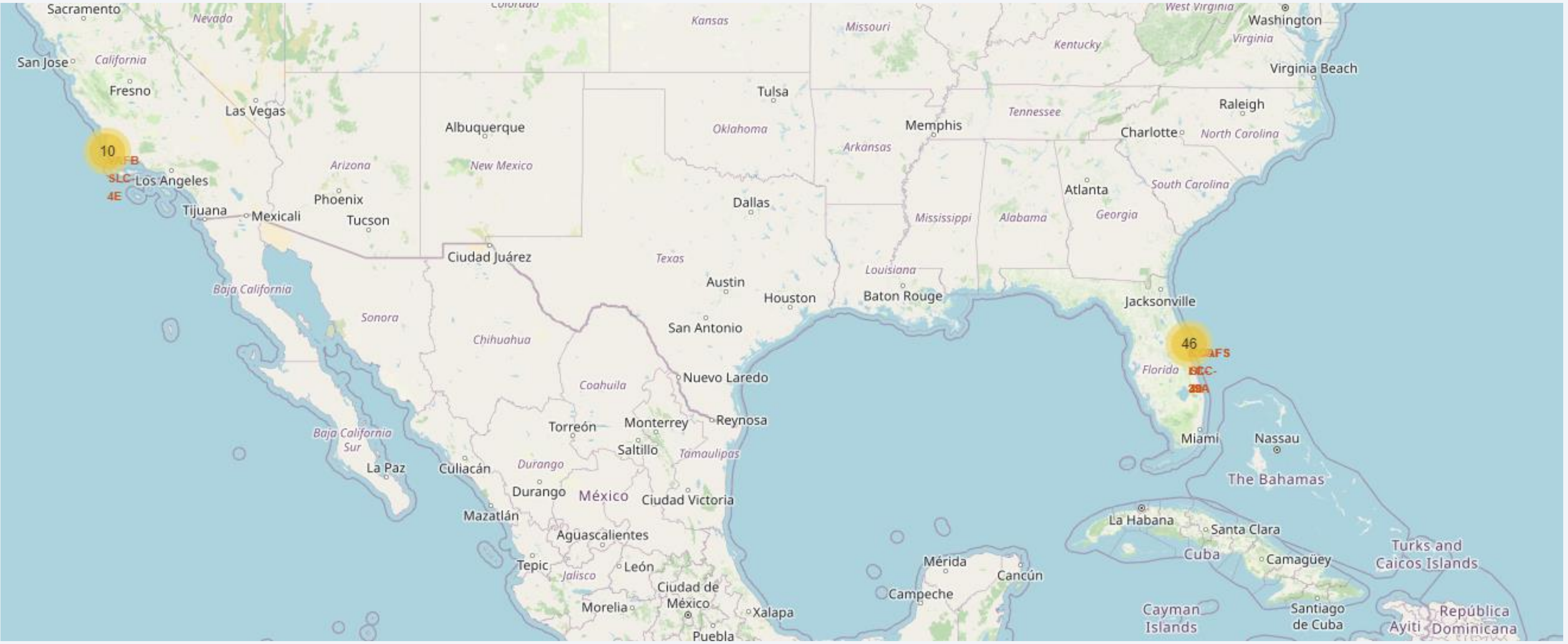
Landing_Outcome	count_c
No attempt	10
Success (drone ship)	5
Failure (drone ship)	5
Success (ground pad)	3
Controlled (ocean)	3
Uncontrolled (ocean)	2
Failure (parachute)	2
Precluded (drone ship)	1

Section 3

Launch Sites Proximities Analysis

Folium Map: Generating Markers and Circles

- The generated map is to present the launch sites with markers and coordinate circles for these sites



Folium Map: Color-Labeled Launch Outcomes

- This map presents the launch site locations having successful (green/1) or failed (red/0) launches per launch site.

Folium Map: Launch Site Proximities

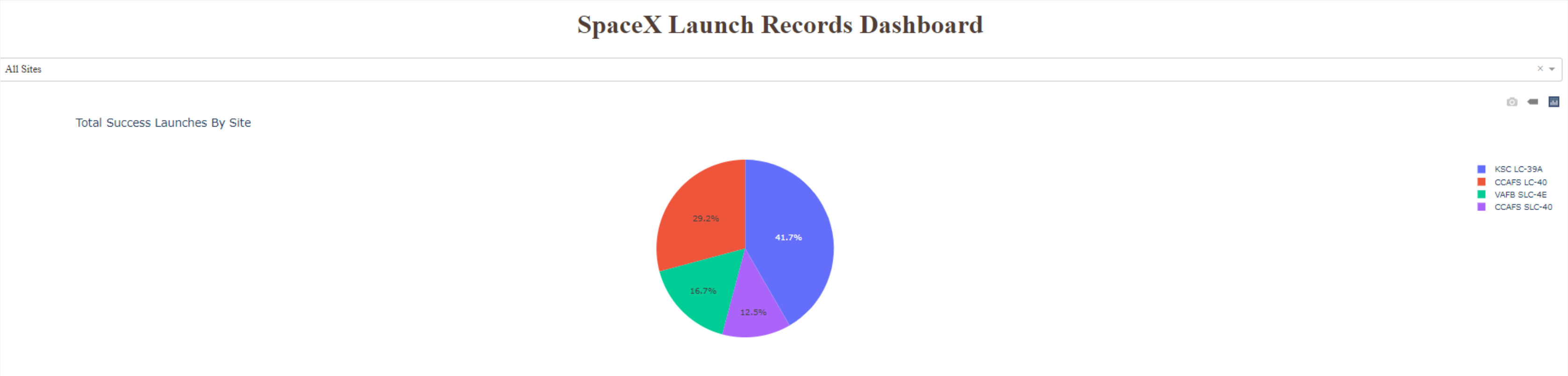
- The map shows a selected launch site and displaying its proximities such as railway, highway, coastline, with distance calculated

Section 4

Build a Dashboard with Plotly Dash

Launch Success (Pie Chart)

- Launch success count for all sites, represented in a Pie chart

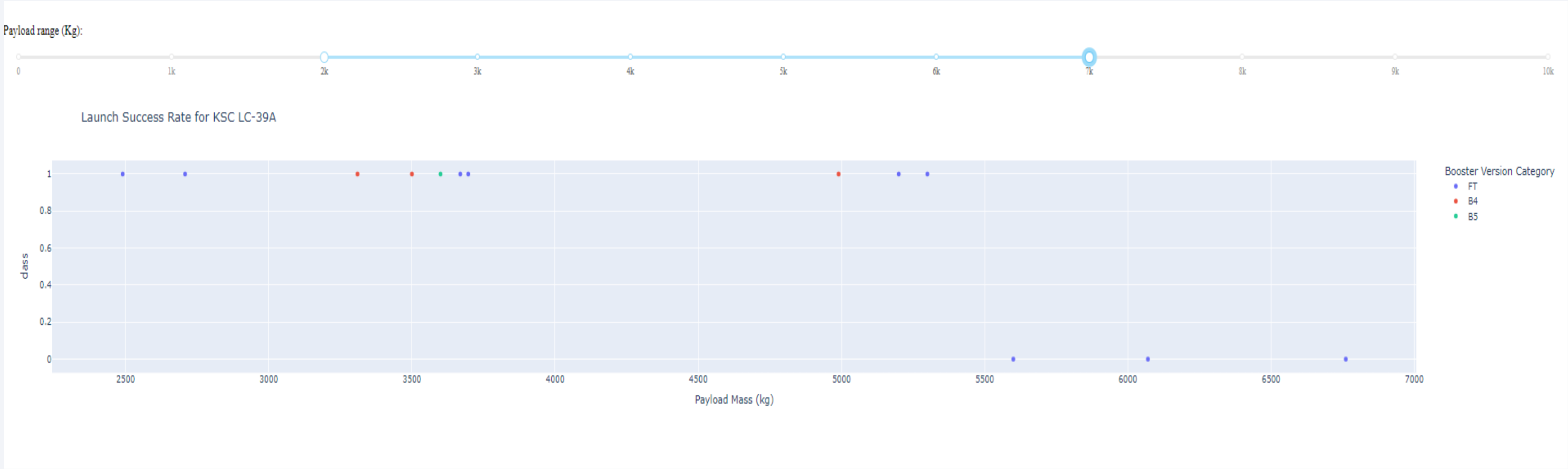


Highest Launch Success Ratio

- Pie chart for the launch site with highest launch success ratio is the KSC LC-39A

Payload vs. Launch Outcome for all sites

- Payload vs. Launch Outcome scatter plot for all sites, with different payload selected in the range slider

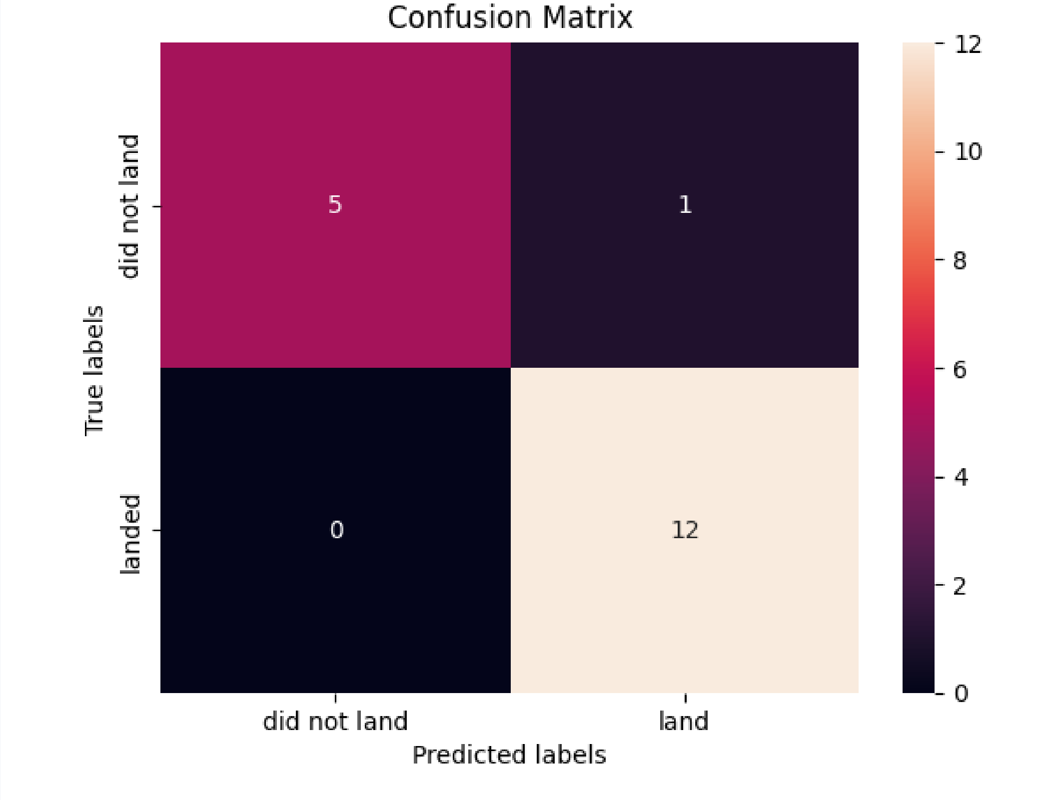


Section 5

Predictive Analysis (Classification)

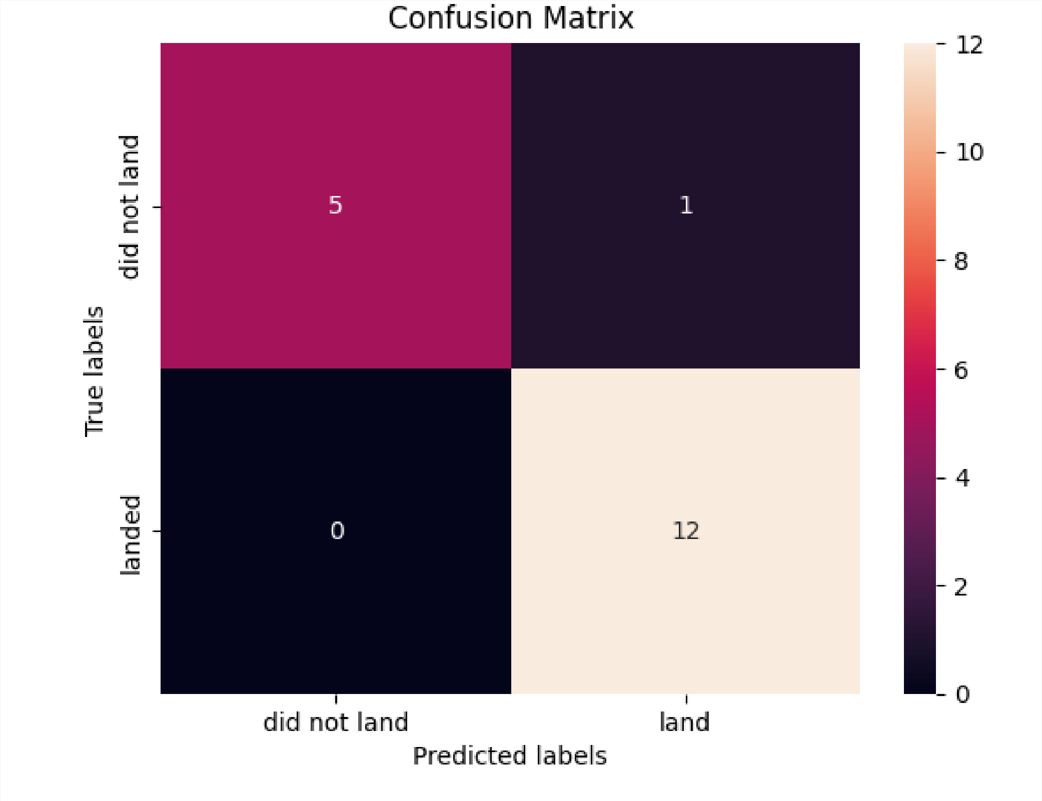
Classification Accuracy

- The model with the highest classification accuracy is Logistic Regression and SVM methods.



Confusion Matrix

- The confusion matrix of the best performing model was presented through the Logistic Regression and SVM methods.



Conclusions

- With the implementation of various comparisons with data, one can be able to determine and analyze the best resulting form and locations for the future Falcon 9 project.
- With observations through the data and several models, would be best to assume that the most successful launch site would be KSC LC-39A.
- The best method(s) to perform test on data in form of data analysis is Logistic Regression and SVM.

Appendix

- All relevant assets can be reviewed here: https://github.com/JAustin5/IBM-Data-Science-Capstone/tree/main

Thank you!