IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Mohammad O. Farooq
4/23/2023

Outline

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

Executive Summary

Data Collection
- From the SpaceX API
- Through Web Scraping

Data Wrangling
- Perform Exploratory Data Analysis
- Training Labels

Exploratory Data Analysis
- EDA through the use of SQL queries

Data Visualization
- EDA through the use of Data Visualization
- Feature Engineering

Visual Analytics
- Interactive map with Folium
- Dashboard with Plotly

Machine Learning
- Classification Models
- Predictive Analysis

- When predicting the best classification model, there were two models out of the four which yielded the same accuracy score and would be good to use for future machine learning predictions.

Introduction

SpaceX is the only private company ever to return a spacecraft from low-earth orbit, which it first accomplished in December 2010.

Falcon 9 rocket launches are advertised on SpaceX's 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.

What we want to accomplish:

- Utilize web scraping, exploratory data analysis, data visualization, and machine learning to predict whether Falcon 9 will land successfully.
- Determine the cost of a launch, which can be useful information for if an alternate company wants to bid against SpaceX for a rocket launch.

Section 1

Methodology

Methodology

- Collect the data:
  - Open Source SpaceX REST API
  - Web Scraping from HTML Table on Falcon 9 Launches Wikipedia page
- Data wrangling
  - Describe how data was processed
- Perform exploratory data analysis (EDA) using visualization and SQL queries
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - How to build, tune, evaluate classification models
  - Choosing the best predictive model (most accurate)

Data Collection

API

Data was first collected from an open source SpaceX REST API.

That data was then cleaned to compose our final dataset.

Web Scraping

Extracted the HTML table of Falcon 9 launch records from the Wikipedia page

The extracted data was parsed and a new DataFrame was created.

Data Collection – SpaceX API

- **After requesting the data, we cleaned the data using a series of functions.**
- **We then used that filtered data to create a new dataset, which we exported to CSV for the next step**

Launch data was requested from the SpaceX API

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

response = requests.get(spacex_url)
```

Decoded the response as a Json and turned it into a Pandas dataframe

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

Created a dictionary using lists containing stored data that was requested using the API (e.g. BoosterVersion from 'rocket' column)

Created a pandas data frame from the above dictionary

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

Filtered the dataframe to show only Falcon 9 launches

```
data_falcon9 = data_ld[data_ld['BoosterVersion'] != 'Falcon 1']
```

Replaced the missing payload values with the column's mean value

```
avg_pm = data_falcon9["PayloadMass"].mean()

data_falcon9["PayloadMass"] = data_falcon9["PayloadMass"].replace(np.nan, avg_pm)
```

https://github.com/mofarooq3/IBM_Course/blob/main/SpaceX%20Capstone%20Project/1%20-%20Data%20Collection%20API.ipynb

Data Collection - Scraping

- After requesting the data from the URL, we created a BeautifulSoup object so we could properly parse the data
- We then created a DataFrame from the extracted data before finally exporting it to CSV for the next step

Requested Falcon 9 Wikipedia page from URL

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

Created a BeautifulSoup object from the above response

```
soup = BeautifulSoup(response.text, 'lxml')
```

Extracted all column names from the HTML table and created a dictionary with keys from extracted column names

Converted dictionary to DataFrame and exported it to CSV

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

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

https://github.com/mofarooq3/IBM_Course/blob/main/SpaceX%20Capstone%20Project/2%20-%20Data%20Collection%20w_%20Web%20Scraping.ipynb

Data Wrangling

- **When calculating the number of launches, orbit, and mission outcome, we created a dataset for the mission outcomes**
- **That dataset was used to identify all of the failed landings and create a new dataset that would be the source of the training labels**
- **The success rate was determined to be 66.67%**

Calculated % of missing values for each attribute

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

Calculated the number of launches on each site, number and occurrence of each orbit, and number and occurrence of mission outcome per orbit type

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

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

```
landing_outcomes = df["Outcome"].value_counts()
landing_outcomes
```

Created a dataset populated by all variations of unsuccessful landings

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

{'False ASDS', 'False Ocean', 'False RTLS', 'None ASDS', 'None None'}
```

Created training labels for landing outcomes based on the dataset 'bad_outcomes'

Determined success rate and exported to CSV

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

0.6666666666666666
```

https://github.com/mofarooq3/IBM_Course/blob/main/SpaceX%20Capstone%20Project/3%20-%20Data%20Wrangling.ipynb

EDA with Data Visualization

- Scatter Plots were created to determine correlation between the variables and be able to assist with the future prediction. They also included whether each flight was a success/failure. Among these scatter plots were the following:
  - Flight Number vs. Payload Mass
  - Flight Number vs. Launch Site
  - Payload Mass vs. Launch Site
  - Flight Number vs. Orbit Type
  - Payload Mass vs. Orbit Type
- Year vs. Success Rate (line graph) was created to observe the trend for successful/failed landings over the years
- Orbit Type vs. Success Rate (bar graph) was created to determine whether the type of orbit has an effect on whether the rocket lands successfully.

https://github.com/mofarooq3/IBM_Course/blob/main/SpaceX%20Capstone%20Project/5%20-%20EDA%20Data%20Viz.ipynb

EDA with SQL

- After connecting to the database, queries were run for the following:
  - Unique launch site names in the space mission
  - 5 records where the launch site begins with "CCA"
  - Total payload mass carried by boosters launched by NASA (CRS)
  - Average payload mass carried by Booster Version F9 v1.1
  - Date when the first successful ground pad landing was achieved
  - List booster versions with success in drone ship landing and a payload mass between 4000-6000kg
  - Total number of successful and failed landings
  - List booster versions that carried the maximum payload mass
  - List the records displaying month, year, landing outcome, booster version, and launch site for the year 2015
  - Rank the amount of successful landing outcomes between 04-06-2010 and 20-03-2017 in descending order

https://github.com/mofarooq3/IBM_Course/blob/main/SpaceX%20Capstone%20Project/4%20-%20EDA%20with%20SQL.ipynb

Build an Interactive Map with Folium

<u>Visualization of SpaceX Launch Sites</u>

Features added to the map include:

- Circle markers around each launch site
- Marker cluster of landing outcomes for each launch site
- Lines from the VAFB SLC-4E launch site to nearby features: the nearest coastline (Pacific Ocean), railway (Coast Line), highway (CA 246), and city (Lompoc).

The proximity of features to the VAFB SLC-4E launch site were added to analyze the location strategy and metrics used when deciding where to launch from.

By zooming in to a launch site, we can click on the marker cluster and see how many launches were successful and how many were failed from that specific site.

https://github.com/mofarooq3/IBM_Course/blob/main/SpaceX%20Capstone%20Project/6%20-%20Location%20Analysis%20with%20Folium.ipynb

Build a Dashboard with Plotly Dash

Interactive Plotly dashboard with two main data visualization elements: pie chart & scatter plot.

- There is a dropdown menu at the top to specify a launch site or select all sites
- There is a slider above the scatter plot in order to focus the x-axis to a certain range of payload mass in order to analyze correlations with varied data ranges.

Pie Chart:

- Displays the success rate of all launch sites, or a specific launch site if indicated in the dropdown menu above
- Added to be able to compare the success rates across launch sites

Scatter Plot:

- Payload Mass (kg) vs. Success Rate
- Used to illustrate whether there is any correlation between payload and success
- Can specify a payload range from the slider above to more closely analyze a specific range
- The legend on the right shows which Booster Version Category each point represents, enabling us to identify which category had higher success rates.

https://github.com/mofarooq3/IBM_Course/blob/main/SpaceX%20Capstone%20Project/7%20spacex_dash_app.py

Predictive Analysis (Classification)

- All four classification models were executed and tested against the test data:
  - •Logistic Regression & SVM were the most accurate models, yielding 83.33% accuracy
  - •Decision Tree was the least accurate model, yielding 72.22% accuracy

Created a NumPy array from the column 'Class'

```
Y =  data['Class'].to_numpy()
```

Standardized the data in X

```
transform = preprocessing.StandardScaler()
X = transform.fit_transform(X)
```

Split the data into training and testing sets

```
X_train, X_test, Y_train, Y_test = train_test_split(X,Y,test_size = 0.2, random_state = 0)
```

Ran the following classification models:

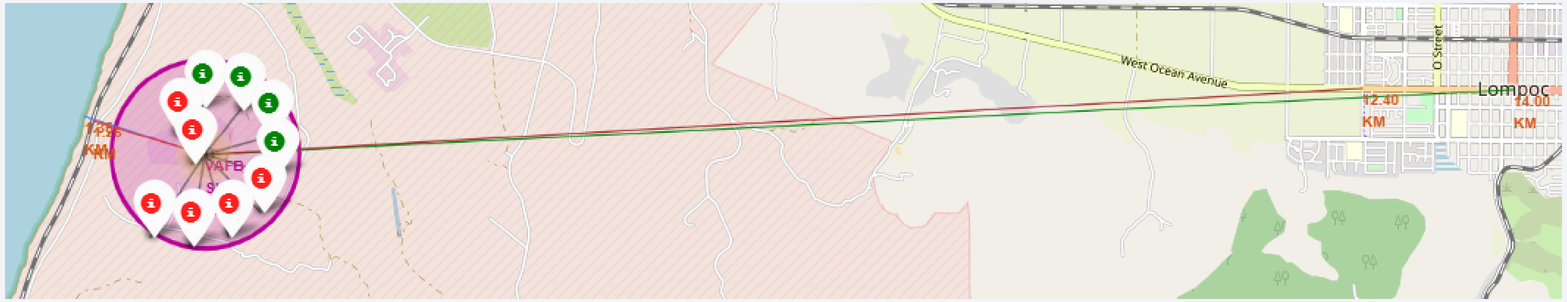
Logistic Regression model	K Nearest Neighbors model
Support Vector Machine model	Decision Tree Classifier model

Tested the accuracy of all models against the test data and created Confusion Matrices for each

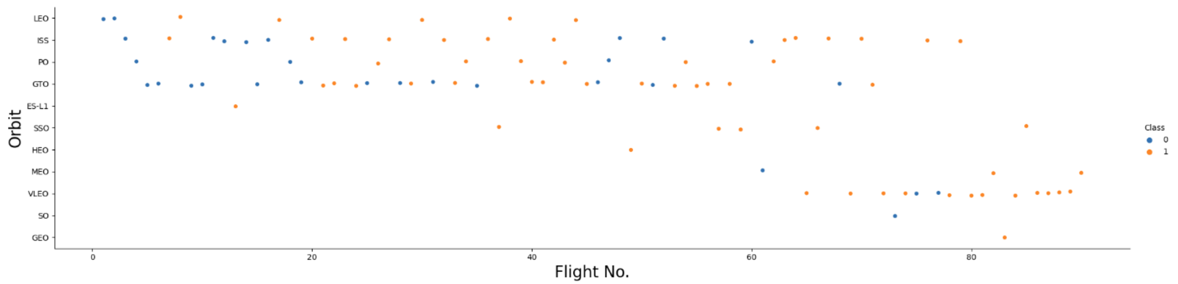
https://github.com/mofarooq3/IBM_Course/blob/main/SpaceX%20Capstone%20Project/8%20-%20Machine%20Learning%20Prediction.ipynb

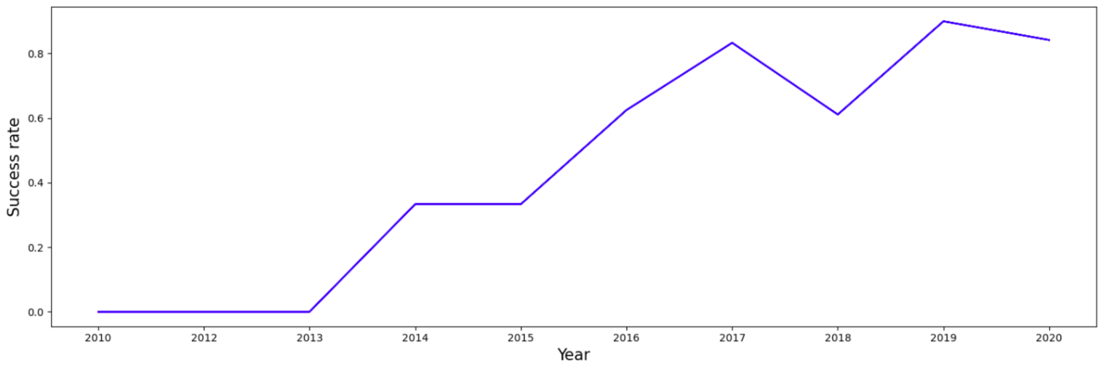
Results

All map features illustrated here:

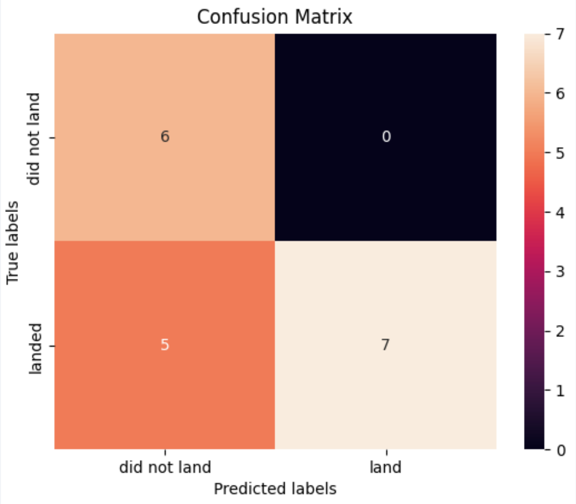


Orbit Type not always correlated with number of flights:





Success rate increased with time:

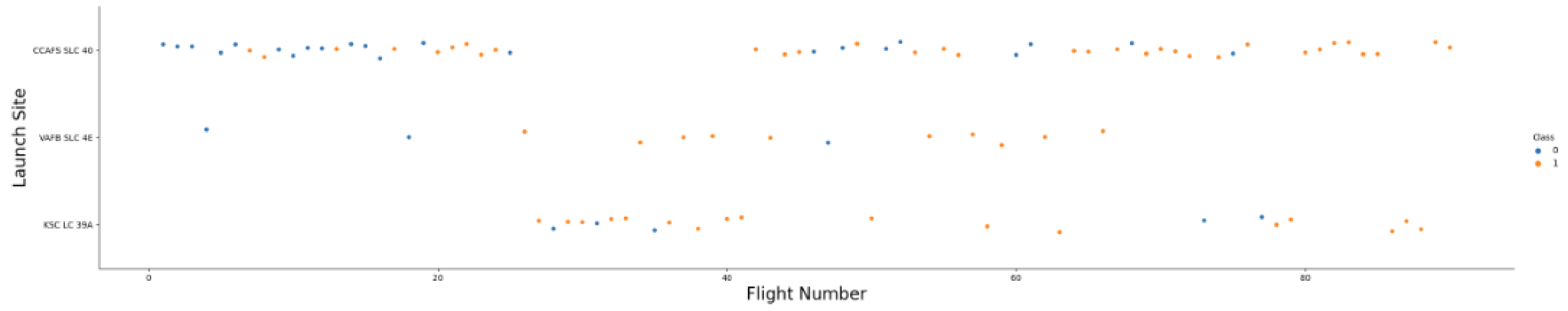


Confusion Matrix of Decision Tree Classifier (least accurate):

Section 2

Insights drawn from EDA

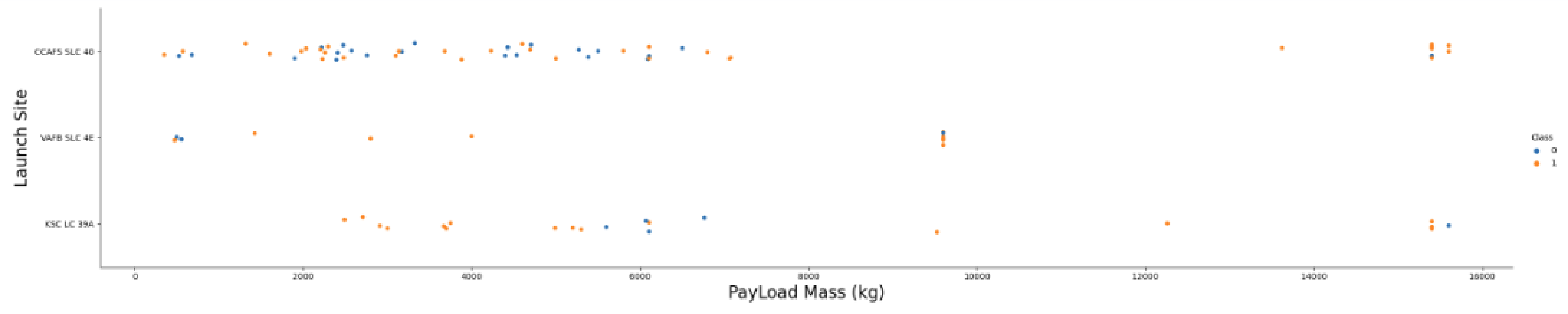
Flight Number vs. Launch Site



This scatter plot shows us the amount of flights that were taken from the three launch sites, and whether each launch resulted in a successful or failed landing. Looking at this scatter plot, we can notice a few things:

- **CCAFS SLC 40 had the most flights, but the lowest rate of successful landings**
- **VAFB SLC 4E had the least flights, but with a slightly higher success rate than CCAFS SLC 40**
- **KSC LC 39A had more flights and a slightly higher success rate than VAFB SLC 4E.**

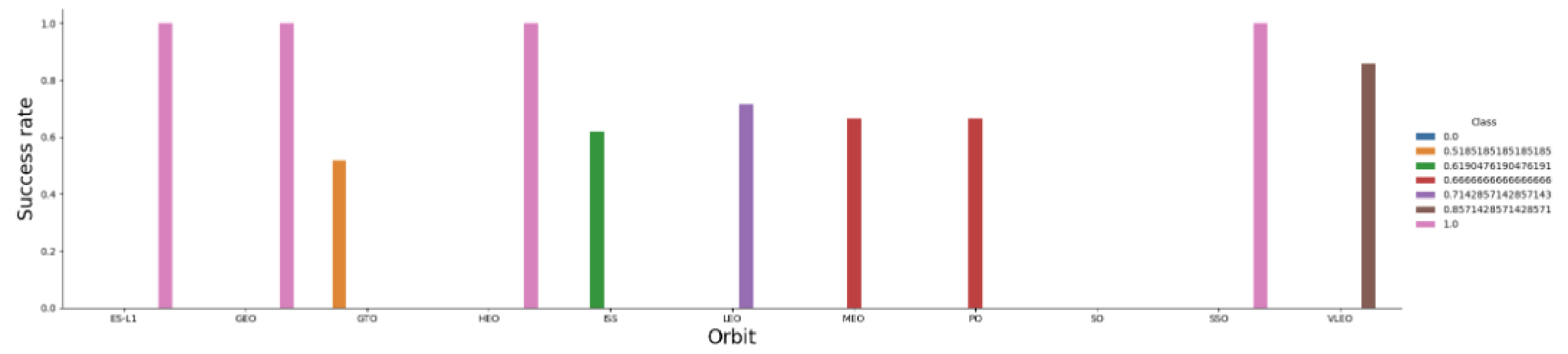
Payload vs. Launch Site



This scatter plot now gives us information on the Payload Mass of each flight against the launch sites, while keeping the determination of successful vs. failed landings. From this scatter plot, we can see:

- **The VAFB SLC 4E launch site had no launches with a payload of over 10,000kg**
- **The KSC LC 39A launch site had no launches with a payload under ~2300kg**
- **The rate of success doesn't have a very strong correlation with the payload mass, because the outcome of each of the landings was varied throughout the scatter plot.**

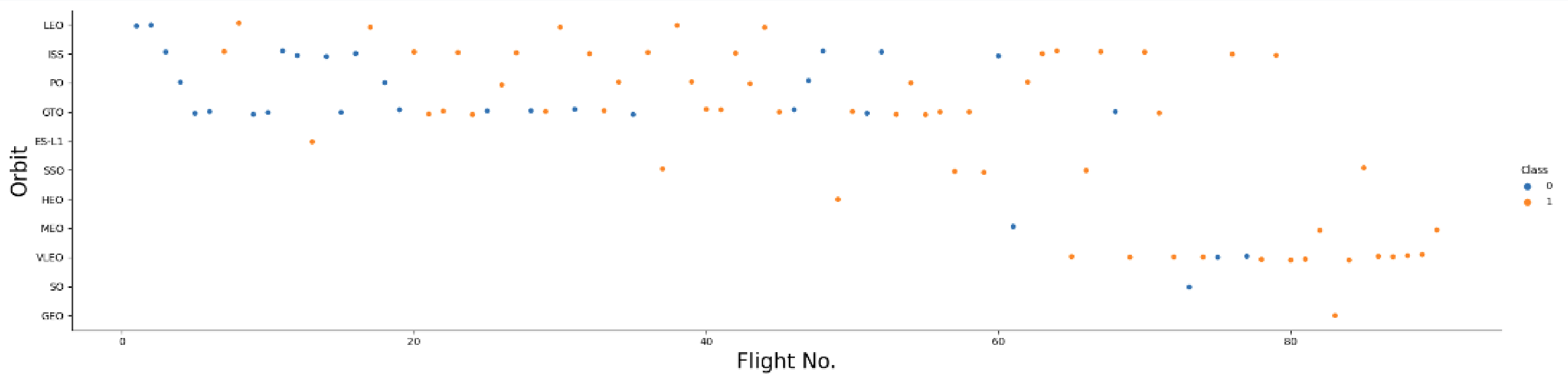
Success Rate vs. Orbit Type



This bar graph shows us the success rate of flights within certain Orbit Types, and it is also color coded with a legend on the right to make visual analysis of the graph easier. We can note that:

- **ES-L1, GEO, HEO, and SSO Orbit Types have the highest success rate**
  - ***ES-L1, GEO, and HEO only have 1 flight each, so there's not much credibility to the success rate* (shown on next slide)**
- **GTO has the lowest success rate, though it has the most flights**

Flight Number vs. Orbit Type

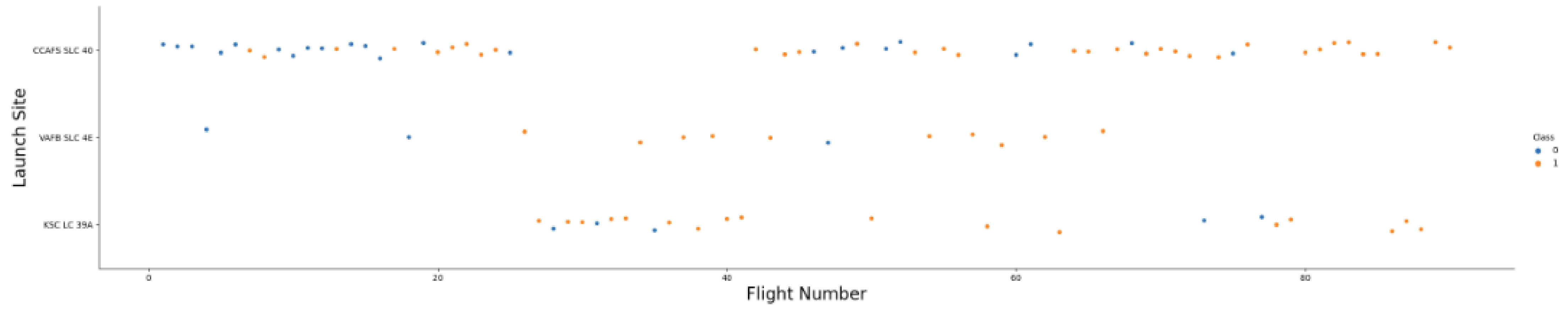


This scatter plot shows us the amount of flights taken for each orbit type.

Though this visual also indicates whether each launch had a successful or failed landing, the success rate would be better extracted from the bar graph in the previous slide. From this scatter plot, we can see:

- **GTO had the most flights, followed by ISS**
- **ES-L1, HEO, and GEO only have 1 flight each**
- **SSO has 5 flights, all resulting in a successful landing**

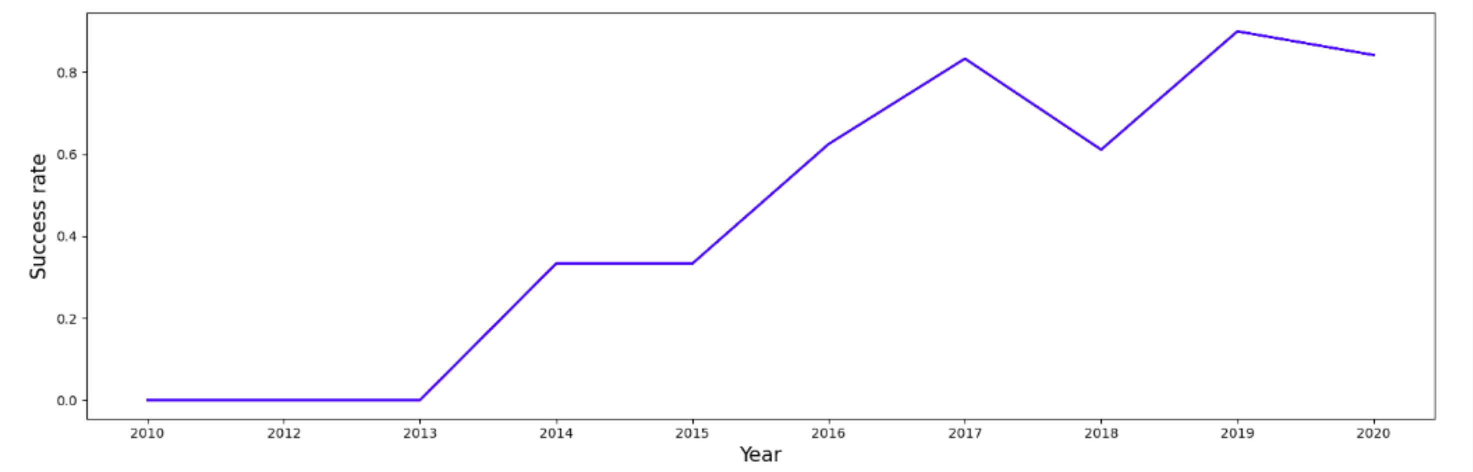
Payload vs. Orbit Type



This scatter plot shows us the amount of flights that were taken from the three launch sites, and whether each launch resulted in a successful or failed landing. Looking at this scatter plot, we can notice a few things:

- **CCAFS SLC 40 had the most flights, but the lowest rate of successful landings**
- **VAFB SLC 4E had the least flights, but with a slightly higher success rate than CCAFS SLC 40**
- **KSC LC 39A had more flights and a slightly higher success rate than VAFB SLC 4E.**

Launch Success Yearly Trend



The above line graph shows how the success rate was affected over years:

- **Success rate began increasing in 2013 and continued increasing until 2019**
- **Success rate dropped in 2018 before going back up the next year**

All Launch Site Names

```
%sql SELECT DISTINCT Launch_Site from SPACEXTBL;
```

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

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

Using SQL queries, we were able to learn a lot about the data.

Here, we can see that there are 4 distinct launch sites that were extracted from the launch records.

Launch Site Names Begin with 'CCA'

```
%sql SELECT * from SPACEXTBL where Launch_Site like 'CCA%' LIMIT 5;
```

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

Next, we executed a query to see 5 launch records in which the Launch Site began with 'CCA'

Total Payload Mass

```
%sql SELECT SUM(Payload_Mass__KG_) as Total_PL from SPACEXTBL where Customer like 'NASA (CRS)';
```

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

Total_PL
45596

Executed a SQL query to calculate the total Payload Mass (kg), which was 45,596kg

Average Payload Mass by F9 v1.1

```
%sql SELECT AVG(Payload_Mass__KG_) as Avg_PL from SPACEXTBL where Booster_Version like 'F9 v1.1';
```

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

Avg_PL
2928.4

Executed a SQL query to calculate the average of the Payload Masses (kg) carried by Booster F9 v1.1, which was 2,928.4kg

First Successful Ground Landing Date

```
%sql SELECT min(Date) as First_Success from SPACEXTBL where "Landing _Outcome" like "Success (ground pad)";
```

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

First_Success
01-05-2017

Determine the date of the flight that resulted in the first successful ground landing: 01/02/2017

Successful Drone Ship Landing with Payload between 4000 and 6000

** %sql SELECT Booster_Version, Payload_Mass__KG_, "Landing _Outcome" from SPACEXTBL where "Landing _Outcome" = 'Success (drone ship)' and Payload_Mass__KG_ between 4000 and 6000; **

```
%sql SELECT Booster_Version, Payload_Mass__KG_, "Landing _Outcome" from SPACEXTBL where "Landing _Outcome" = 'Success (drone ship)' and Payload_Mass__
```

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

Booster_Version	PAYLOAD_MASS__KG_	Landing _Outcome
F9 FT B1022	4696	Success (drone ship)
F9 FT B1026	4600	Success (drone ship)
F9 FT B1021.2	5300	Success (drone ship)
F9 FT B1031.2	5200	Success (drone ship)

Show which Boosters have success in drone ship landings with a payload mass between 4000-6000kg

Total Number of Successful and Failure Mission Outcomes

```
%sql SELECT Mission_Outcome, Count(*) as Total from SPACEXTBL group by Mission_Outcome;
```

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

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

Show how many flights were successful and how many were failures

Boosters Carried Maximum Payload

```
%sql SELECT Booster_Version from SPACEXTBL where Payload_Mass__KG_ = (SELECT MAX(Payload_Mass__KG_) from SPACEXTBL);
```

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

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

Using a subquery to list out which of the Boosters have carried the Maximum Payload Mass (kg)

2015 Launch Records

** %sql SELECT substr(Date, 4, 2) as Month, substr(Date, 7, 4) as Year, "Landing _Outcome", Booster_Version, Launch_Site from SPACEXTBL where "Landing _Outcome" like 'Failure (drone ship)' and Year='2015'; **

```
%sql SELECT substr(Date, 4, 2) as Month, substr(Date, 7, 4) as Year, "Landing _Outcome", Booster_Version, Launch_Site from SPACEXTBL where "Landing _O
```

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

Month	Year	Landing _Outcome	Booster_Version	Launch_Site
01	2015	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
04	2015	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

For the data of failed launch records from the year 2015, output a dataset showing the month, year, landing outcome, booster version, and launch site

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

** %sql SELECT "Landing _Outcome", Count(*) as Total_Sc from SPACEXTBL where "Landing _Outcome" like 'Success%' and Date between '04-06-2010' and '20-03-2017' group by "Landing _Outcome" order by Total_Sc DESC; **

```
%sql SELECT "Landing _Outcome", Count(*) as Total_Sc from SPACEXTBL where "Landing _Outcome" like 'Success%' and Date between '04-06-2010' and '20-03-
```

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

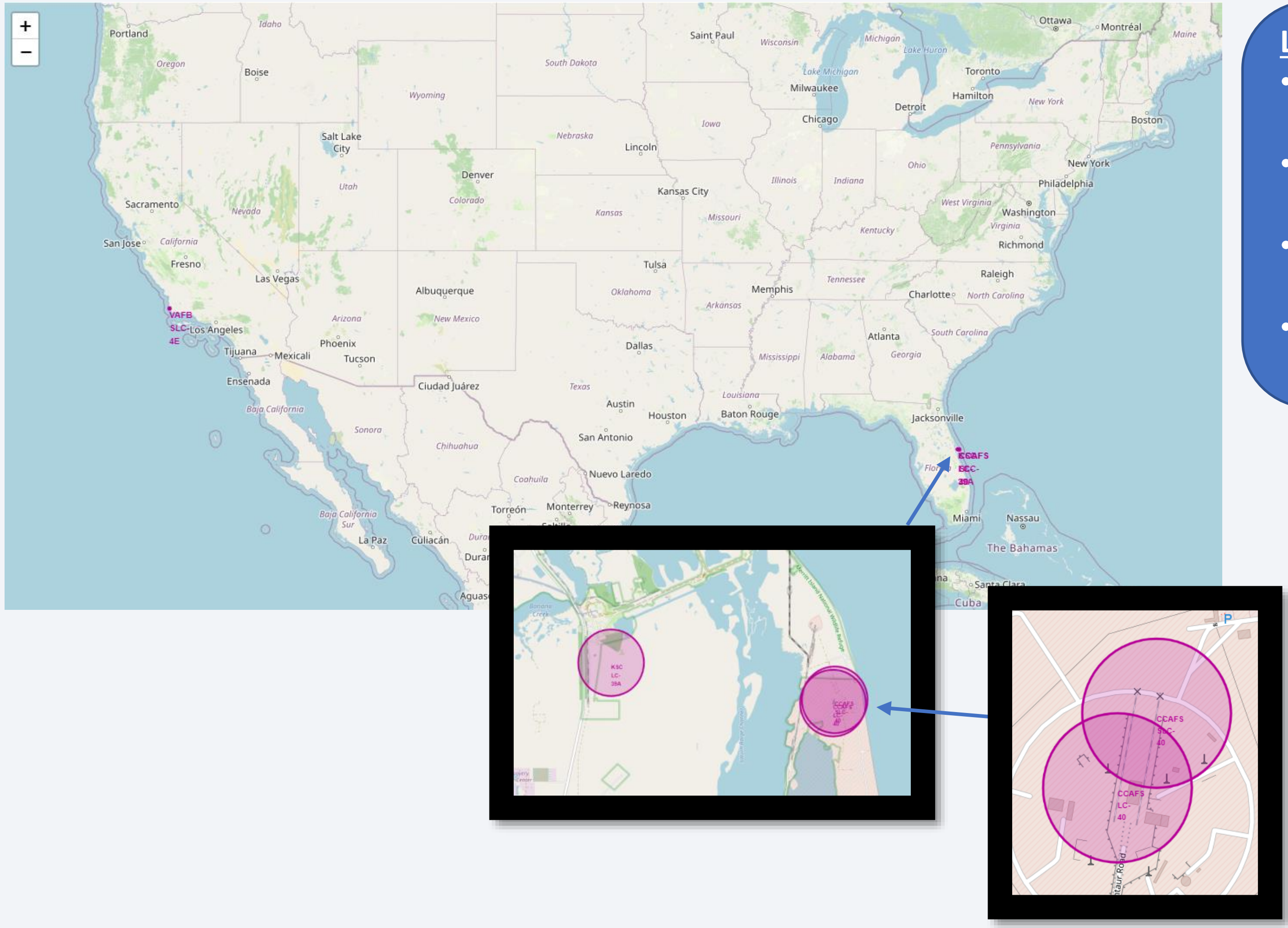
Landing _Outcome	Total_Sc
Success	20
Success (drone ship)	8
Success (ground pad)	6

From 04-06-2010 to 20-03-2017, rank the successful landing outcomes in descending order

Section 3

Launch Sites Proximities Analysis

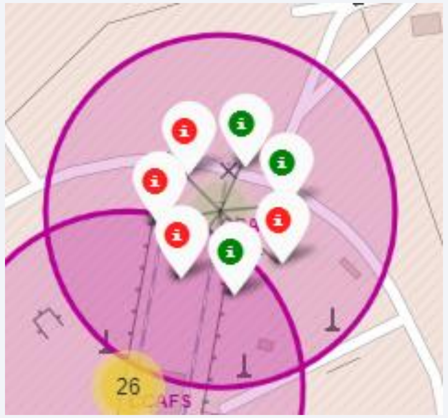
All Launch Sites



Launch Sites (East to West):

- Cape Canaveral Air Force Station Space Launch Complex 40 (CCAFS SLC-40) - Florida
- Cape Canaveral Air Force Station Launch Complex 40 (CCAFS LC-40) - Florida
- Kennedy Space Center Launch Complex 39A (KSC LC-39A) - Florida
- Vandenberg Air Force Base Space Launch Complex 4 East (VAFB SLC-4E) - California

Success/Failure of Launches by Site

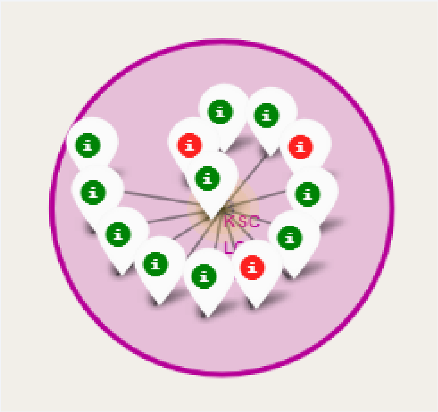


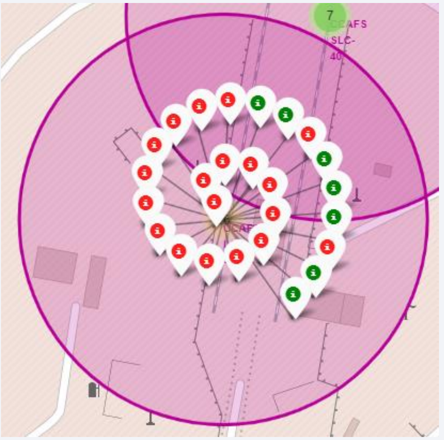
CCAFS SLC-40

Failure: 4
Success: 3

KSC LC-39A

Failure: 3
Success: 10





CCAFS LC-40

Failure: 19
Success: 7

VAFB SLC-4E

Failure: 6
Success: 4

Launch Site Proximity to other Features

VAFB SLC-4E

Nearest coastline: Pacific Ocean	1.35km
Nearest city: Lompoc, CA	14.00km
Nearest railway: Coast Line – Santa Barbara Subdivision	1.26km
Nearest highway: CA 246	12.40km

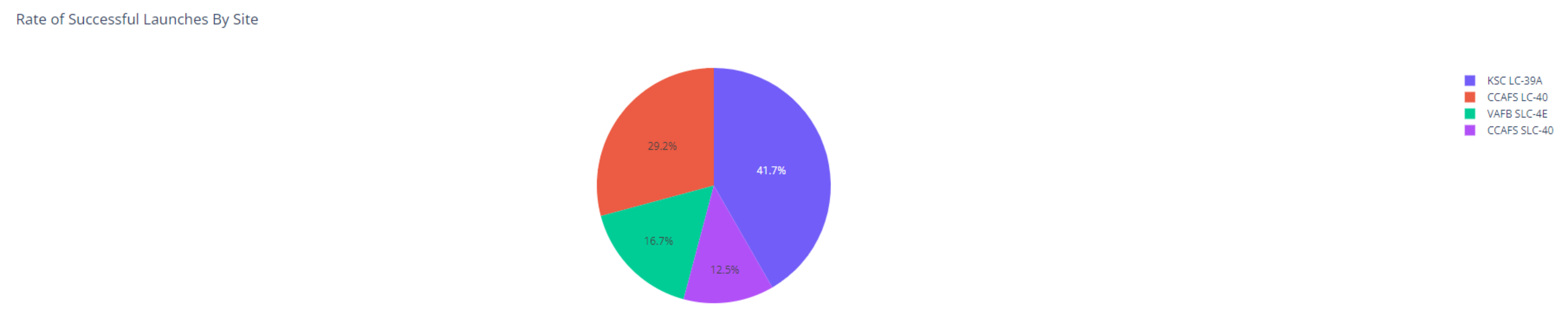
The VAFB launch site is not far from the nearest railway or coastline, almost equally distant from both at 1.26km & 1.35km respectively.

The nearest highway would be the beginning of the CA 246 highway as you enter Lompoc, which happens to be the nearest city as well. However, both of these are reasonably farther away, being upwards of 12km away from the launch site. This is to minimize the risk of casualties in case of unforeseen accidents occurring with launch or landing.

Section 4

Build a Dashboard with Plotly Dash

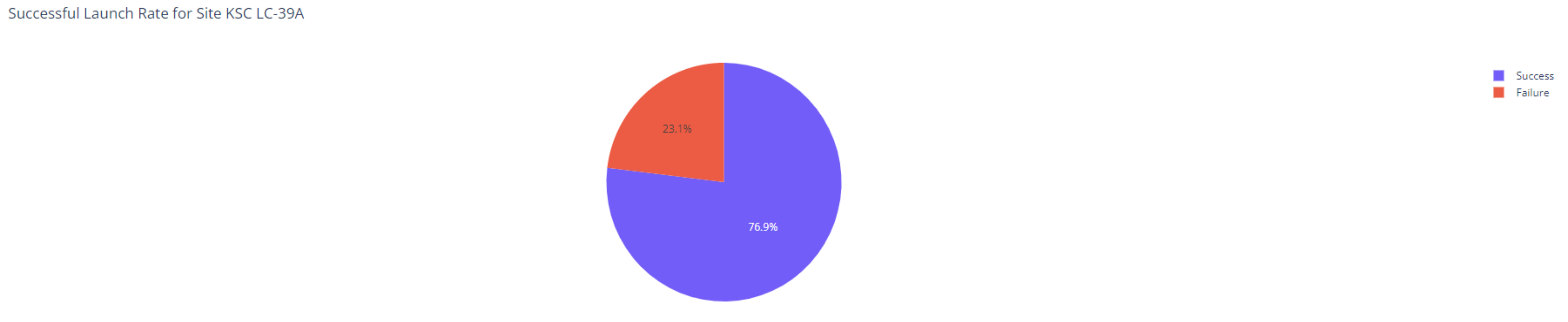
Rate of Successful Launches by Site



Percentage of successful launches for all launch sites:

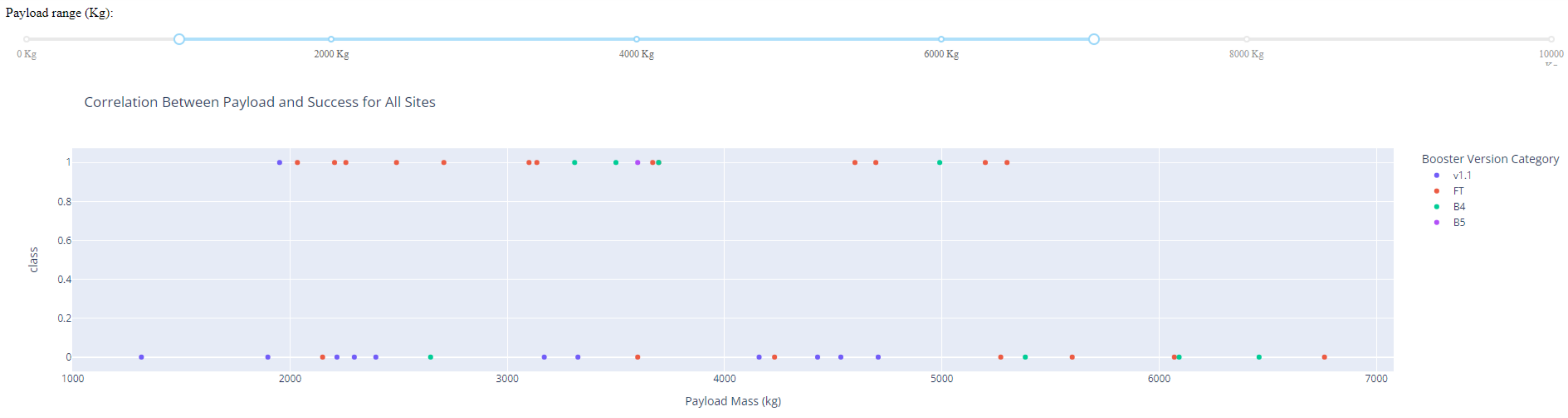
- KSC LC-39A had the most successful launches at 41.7%
- CCAFS SLC-40 had the least successful launches at 12.5%

Highest Launch Success Rate (KSC LC-39A)



Percentage of successful launches for the KSC LC-39A launch site:
- There was a 76.9% success rate with 10 successful launches
- There was a 23.1% failure rate with 3 successful launches

Correlation between Payload and Success Rate



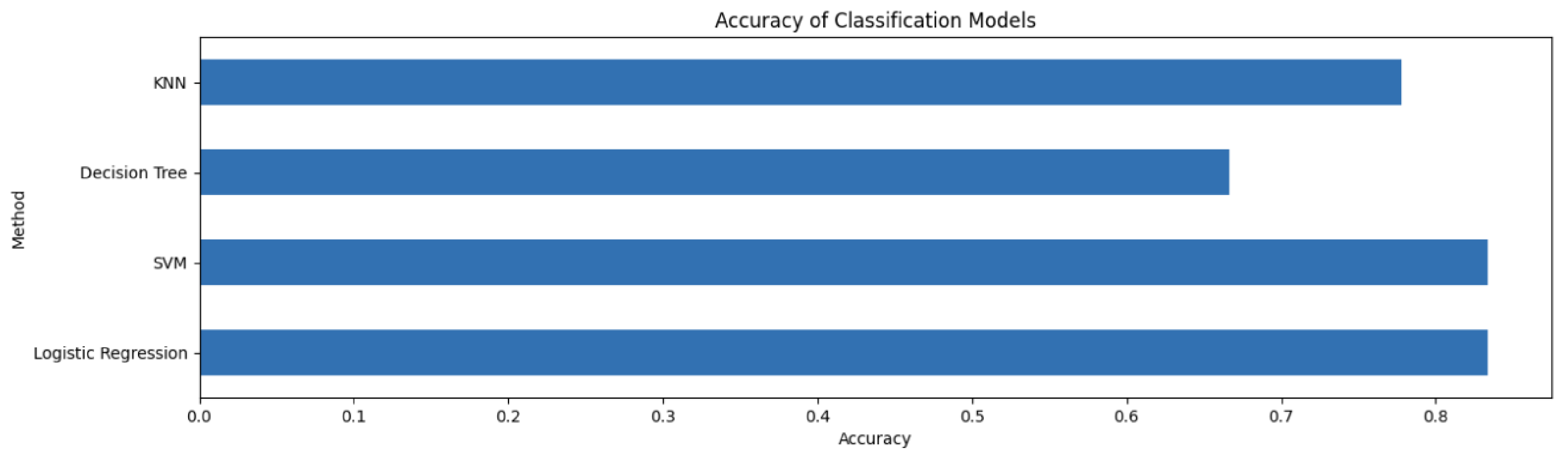
Correlation between Payload Mass (kg) and Success Rate for all sites from 1000-7000kg shown:

- Within this range, the FT Booster Version Category had the highest success rate
- Although most successful landings occurred within the 2000-5500kg range, there is little correlation between Payload Mass and Success Rate
- Little correlation – there were no successful launches from 5500-7000kg, but there were ~equal amounts of successful/failed landings from 2000-5500kg

Section 5

Predictive Analysis (Classification)

Classification Accuracy

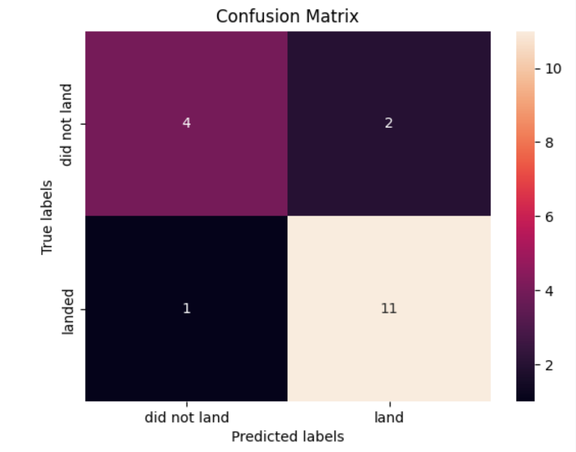


We can conclude from the bar graph that the **Logistic Regression** and **Simple Vector Machine** methods have the highest accuracy.

Confusion Matrix

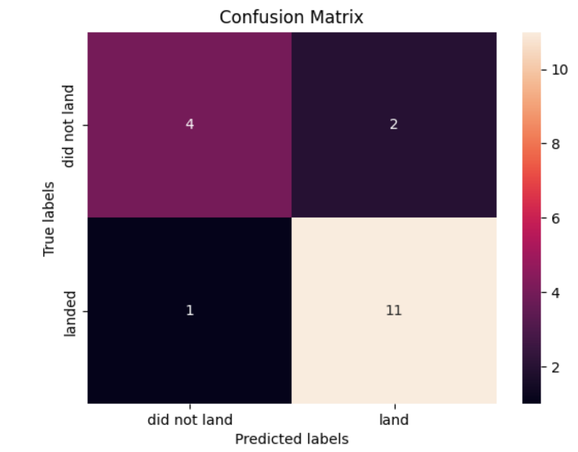
Logistic Regression (83.33%)

```
yhat=logreg_cv.predict(X_test)
plot_confusion_matrix(Y_test,yhat)
```



SVM (83.33%)

```
yhat=svm_cv.predict(X_test)
plot_confusion_matrix(Y_test,yhat)
```

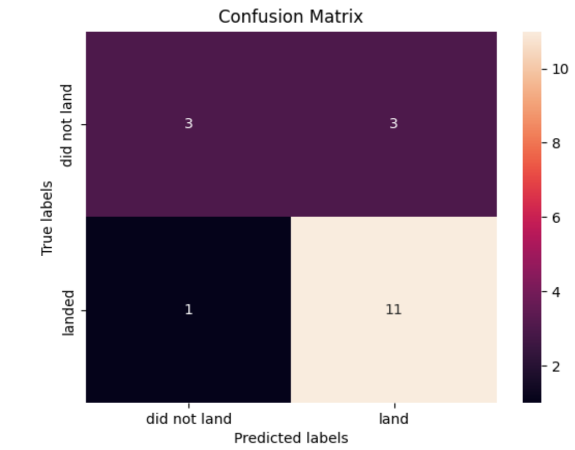


Most accurate

Less accurate

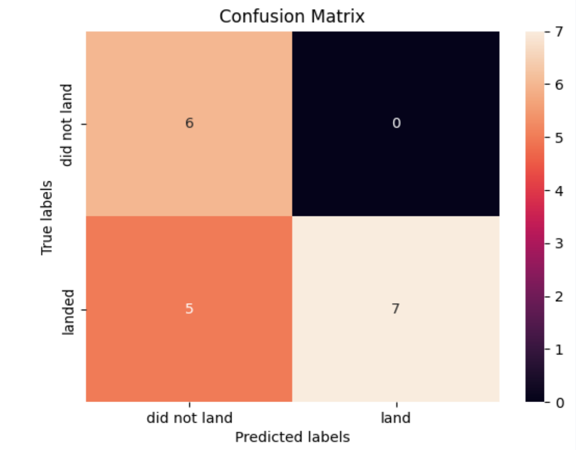
KNN (77.78%)

```
yhat = knn_cv.predict(X_test)
plot_confusion_matrix(Y_test,yhat)
```



Decision Tree (72.22%)

```
yhat = tree_cv.predict(X_test)
plot_confusion_matrix(Y_test,yhat)
```



Conclusions

Though the accuracy of the classification models were not very high, we were able to determine which of the models would be the best predictor with an accuracy score of 83.33% using both the Logistic Regression and SVM models.

Flights utilizing a GTO orbit have the highest rate of failure, in spite of the fact that the majority of launched flights were of that orbit type.

In regards to Launch Site data:

- CCAFS SLC-40 had the most launches
- KSC LC-39A had the most successful flights

In regards to Booster Version Categories:

- FT Booster Versions had the most success
- v1.1 Booster Versions had the most failures

Appendix

GitHub Links:

- Project Github- https://github.com/mofarooq3/IBM_Course/tree/main/SpaceX%20Capstone%20Project
- API- https://github.com/mofarooq3/IBM_Course/blob/main/SpaceX%20Capstone%20Project/1%20-%20Data%20Collection%20API.ipynb
- WebScraping- https://github.com/mofarooq3/IBM_Course/blob/main/SpaceX%20Capstone%20Project/2%20-%20Data%20Collection%20w_%20Web%20Scraping.ipynb
- Wrangling- https://github.com/mofarooq3/IBM_Course/blob/main/SpaceX%20Capstone%20Project/3%20-%20Data%20Wrangling.ipynb
- EDA w/ SQL- https://github.com/mofarooq3/IBM_Course/blob/main/SpaceX%20Capstone%20Project/4%20-%20EDA%20with%20SQL.ipynb
- Data Viz- https://github.com/mofarooq3/IBM_Course/blob/main/SpaceX%20Capstone%20Project/5%20-%20EDA%20Data%20Viz.ipynb
- Folium- https://github.com/mofarooq3/IBM_Course/blob/main/SpaceX%20Capstone%20Project/6%20-%20Location%20Analysis%20with%20Folium.ipynb
- Dashboard- https://github.com/mofarooq3/IBM_Course/blob/main/SpaceX%20Capstone%20Project/7%20spacex_dash_app.py
- ML Prediction- https://github.com/mofarooq3/IBM_Course/blob/main/SpaceX%20Capstone%20Project/8%20-%20Machine%20Learning%20Prediction.ipynb

Thank you!