FALCON 9: FIRST STAGE LANDING AND REUSABILITY

IKENNA OGBUANU
28/08/2023

Table of Contents

- Executive Summary........................ 3
- Introduction........................4
- Methodology........................5
    - Data Collection Steps – SpaceX API Call ....7 - 8
    - Data Collection - Scraping 9
    - Data Wrangling - Steps 10
    - EDA with Data Visualization 11
    - EDA with SQL 12
    - Map Analysis with Folium 13
    - Plotly Dashboards 14
- Predictive Analysis........................15
- Results ....16 – 38
- Conclusion........................39
- Recommendations........................40
- Appendix........................41

Executive Summary

SpaceX is the world's leading provider of launch services and is proud to be the first private company to have delivered astronauts to and from the International Space Station (ISS), and the first and only company to complete an all-civilian crewed mission to orbit in a cost-effective manner. SpaceX can do this because the rocket launches are relatively inexpensive. This is achievable through the Falcon 9 Heavy-lift Launch Vehicle.

The Falcon 9 is a reusable, two-stage rocket designed and manufactured by SpaceX for the reliable and safe transport of people and payloads into Earth orbit and beyond. Falcon 9 is the world's first orbital class reusable rocket. Reusability allows SpaceX to re-fly the most expensive parts of the rocket, which in turn drives down the cost of space access.

Our aim is to investigate the various factors surrounding the SpaceX program and predict if the Falcon 9 first stage will land successfully. This insights derived can then be used as a model for improving the odds for an alternate company bidding against the SpaceX monopoly for a rocket launch.

Introduction

This research attempts to identify and predict the factors required for a successful rocket landing. As a starting point we will need to collect data, as much and relevant as possible, from various sources such as Wikipedia and SpaceX websites using **API calls** and **Web-scraping techniques**, and **running Database queries using SQL scripts** .

After the raw data has been collected, we will improve on its quality by performing **data wrangling** and start drilling deeper into the processed data. In order to gain further insights into the data, we shall be applying some basic **statistical analysis and visualizations** to see how variables might be related to each other.

Furthermore, We will drill down into finer levels of detail by splitting the data into groups defined by **categorical variables** which will be used to build, evaluate, and refine **predictive ML models** for discovering more exciting insights.

The result of the **EDA** analysis and **Data Visualizations** should provide the following insights:

- Benefits of situating most launch sites at the equator and coastal regions.
- Maximum Payload for a successful launch outcome.
- Orbital launch counts and success/failure rates.
- Landing sites with a higher success rate.
- Which ML model was the best performing model and is able to accurately predict outcomes.

METHODOLOGY

Methodology

- ❑ **Collect** data using SpaceX REST API and web scraping techniques
- ❑ **Wrangle** data to create success/fail outcome variable
- ❑ **Explore** data with data visualization techniques, considering the following factors: payload, launch site, flight number and yearly trend
- ❑ **Analyze** the data with SQL, calculating the following statistics: total payload, payload
- ❑ range for successful launches, and total # of successful and failed outcomes
- ❑ **Explore** launch site success rates and proximity to geographical markers
- ❑ **Visualize** the launch sites with the most success and successful payload ranges
- ❑ **Build Models** to predict landing outcomes using logistic regression

Data Collection Steps – SpaceX API Call

- ❑ **Import** “requests” library
- ❑ **Define** custom CALL Functions to get API datasets: getBoosterVersion, getLaunchSite, getPayloadData, getCoreData.
- ❑ **Request data** from SpaceX API (launch data): spacex_url =“https://api.spacexdata.com/v4/launches/past” response = requests.get(spacex_url)
- ❑ **Decode response** using .json() and convert to a dataframe using .json_normalize()
- ❑ **Request information** about the launches from SpaceX API using custom get functions
- ❑ **Create dictionary** from the data
- ❑ **Create dataframe** from the dictionary
- ❑ **Filter dataframe** to contain only Falcon 9 launches
- ❑ **Replace missing values** of Payload Mass with calculated .mean()
- ❑ **Export data** to csv file

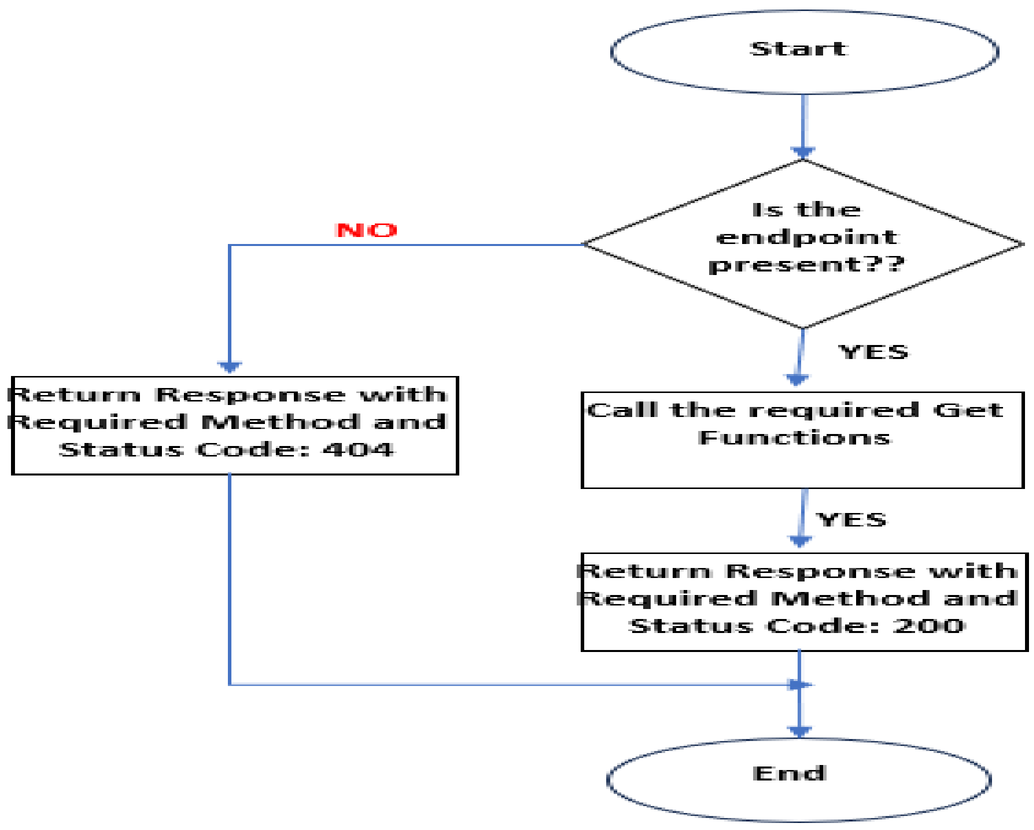
Data Collection – SpaceX API Call

API Call

- ❑ spacex_url=https://api.spacexdata.com/v4/launches/past
- ❑ response = requests.get(spacex_url)
- ❑ response.status_code: 200
- ❑ print(response.content)

Git URL:

https://github.com/Hunting007/SpaceX_Capstone/blob/main/CAPSTONE-spacex-data-collection-api.ipynb



Data Collection - Scraping

- ❑ **Request data** (Falcon 9 launch data) from Wikipedia
- ❑ **Create BeautifulSoup object** from HTML response
- ❑ **Extract column names** from HTML table header
- ❑ **Collect data** from parsing HTML tables
- ❑ **Create dictionary** from the data
- ❑ **Create dataframe** from the dictionary
- ❑ **Export data** to csv file

Git URL

https://github.com/Hunting007/SpaceX_Capstone/blob/main/CAPSTONE-labs-webscraping.ipynb

Data Wrangling - Steps

Data was analyzed from extracted dataset in .csv format using the steps below:

- ❑ Identified and calculated the percentage of the missing values in each attribute to determine the impact of missing data on the overall dataset.
- ❑ Replaced missing data with the column mean.
- ❑ Determined which columns are numerical or categorical, and set the appropriate type.
- ❑ Determined the number of launches on each site and the number and occurrence of each orbit.
- ❑ Calculated the number and occurences of mission outcomes of the orbits and determined their landing outcomes as binary.
- ❑ Create a landing outcome label from Outcome column and parse the outcomes into Binary Classes of 1's and 0's. Successful = 1 and Failed = 0

Git URL: https://github.com/Hunting007/SpaceX_Capstone/blob/main/CAPSTONE-spacex-Data%20wrangling.ipynb

EDA with Data Visualization

Charts

- ❑ Flight Number vs. Payload: Shows the relation between the frequency of flights and the successful landings of the 1st stage. Secondly, the more massive the payload, the less likely the first stage will return.
- ❑ Flight Number vs. Launch Site: Depicts the success rates of different launch sites. CCAFS LC-40, has a success rate of 60 %, while KSC LC-39A and VAFB SLC 4E has a success rate of 77%.
- ❑ Payload Mass (kg) vs. Launch Site: This shows that rockets carrying a heavy payload mass(greater than 10000) have not been launched from the VAFB-SLC launchsite.
- ❑ Payload Mass (kg) vs. Orbit type: This shows that with heavy payloads the successful landing or positive landing rate are more for Polar, LEO and ISS.
- ❑ Launch Success Yearly Trend: It is observed that the success rate since 2013 kept increasing till 2020 which is as a result of the increase in flights.

Git URL:

https://github.com/Hunting007/SpaceX_Capstone/blob/main/CAPSTONE-eda-dataviz.ipynb

EDA with SQL

Queries

- Names of unique launch sites using “DISTINCT” feature.
- 5 records where launch site begins with ‘CCA’ using ‘CCA%’.
- Total payload mass carried by boosters launched by NASA (CRS) using “WHERE” clause.
- Average payload mass carried by booster version F9 v1.1. using AVG()
- Date of first successful landing on ground pad
- Boosters which had success landing on drone ship and have payload mass “BETWEEN 4,000 and 6,000”.
- Total number of successful and failed missions using “GROUP BY”
- Names of booster versions which have carried the max payload using “MAX()”
- Failed landing outcomes on drone ship, booster version and launch site for the months in the year 2015
- Count of landing outcomes between 2010-06-04 and 2017-03-20 using WHERE, GROUP BY and ORDER BY(desc)

https://github.com/Hunting007/SpaceX_Capstone/blob/main/CAPSTONE_SpaceX-eda-sql-coursera.ipynb

Map Analysis with Folium

Geofencing of Launch Sites

- ❑ Blue geofence at Johnson Space Center's coordinate with a popup label showing its name using its latitude and longitude.
- ❑ Red geofences at all launch sites coordinates with a popup label showing its name using its name using its latitude and longitude coordinates.

POIs of Launch Outcomes

- ❑ Successful Launch (green POI marker) and unsuccessful Launch (red POI marker).

Proximity Measurements Between Launch Site and Landmarks

- ❑ Added colored lines to show distance between launch site CCAFS SLC- 40 and its proximity to the nearest landmark e.g coastline, railway, highway, and city.

Git URL:

- https://github.com/Hunting007/SpaceX_Capstone/blob/main/CAPSTONE_launch_site_location.ipynb

Plotly Dashboards

Dropdown option contains the list of Launch Sites

- ❑ Allows the user to select all launch sites or a certain launch site.

Pie Chart Shows Successful Launches

- ❑ Allows the user to see successful and unsuccessful launches as a percent of the total.

Slicer button for Payload Mass Range

- ❑ Allows the user to select payload mass range.

Scatter Chart Showing Payload Mass vs. Success Rate by Booster Version

- ❑ Allows the user to see the correlation between Payload and Launch Success.

Predictive Analysis

STEPS

- **Create** NumPy array from the Class column
- **Standardize** the data with StandardScaler. Fit and transform the data.
- **Split** the data using train_test_split
- **Create** a GridSearchCV object with cv=10 for parameter optimization.
- **Apply** GridSearchCV on different algorithms: Logistic Regression (LogisticRegression()), Support Vector Machine (SVC()), Decision Tree (DecisionTreeClassifier()), K-Nearest Neighbor (KNeighborsClassifier())
- **Calculate** accuracy on the test data using .score() for all models.
- **Assess** the confusion matrix for all models.
- **Identify** the best model using Jaccard_Score, F1_Score and Accuracy.

Git URL:

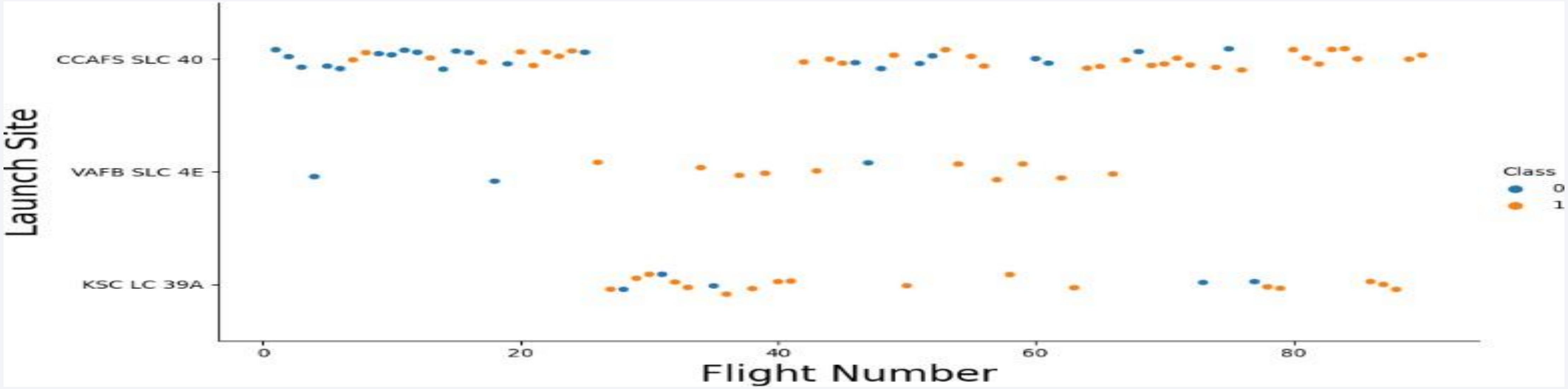
https://github.com/Hunting007/SpaceX_Capstone/blob/main/CAPSTONE-SpaceX_Machine_Learning_Prediction_Part_5.jupyterlite%20(2).ipynb

INSIGHTS DRAWN FROM EDA

Flight Number vs. Launch Site

Exploratory Data Analysis

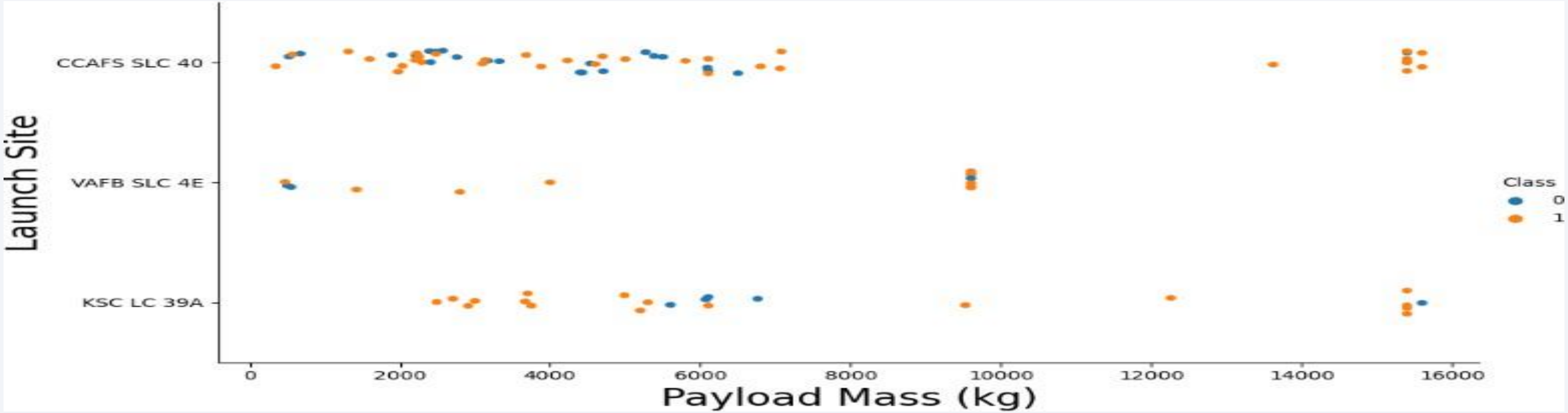
- Earlier flights had a lower success rate (blue = fail), but over time, the success rate improved significantly (orange = success), as the number of flight increased.
- Around 50% of launches are from CCAFS SLC 40 launch site while VAFB SLC 4E and KSC LC 39A have higher success rates
- We can infer that new launches have had higher success rate resulting from lessons learnt from previous failures.



Payload vs. Launch Site

Exploratory Data Analysis

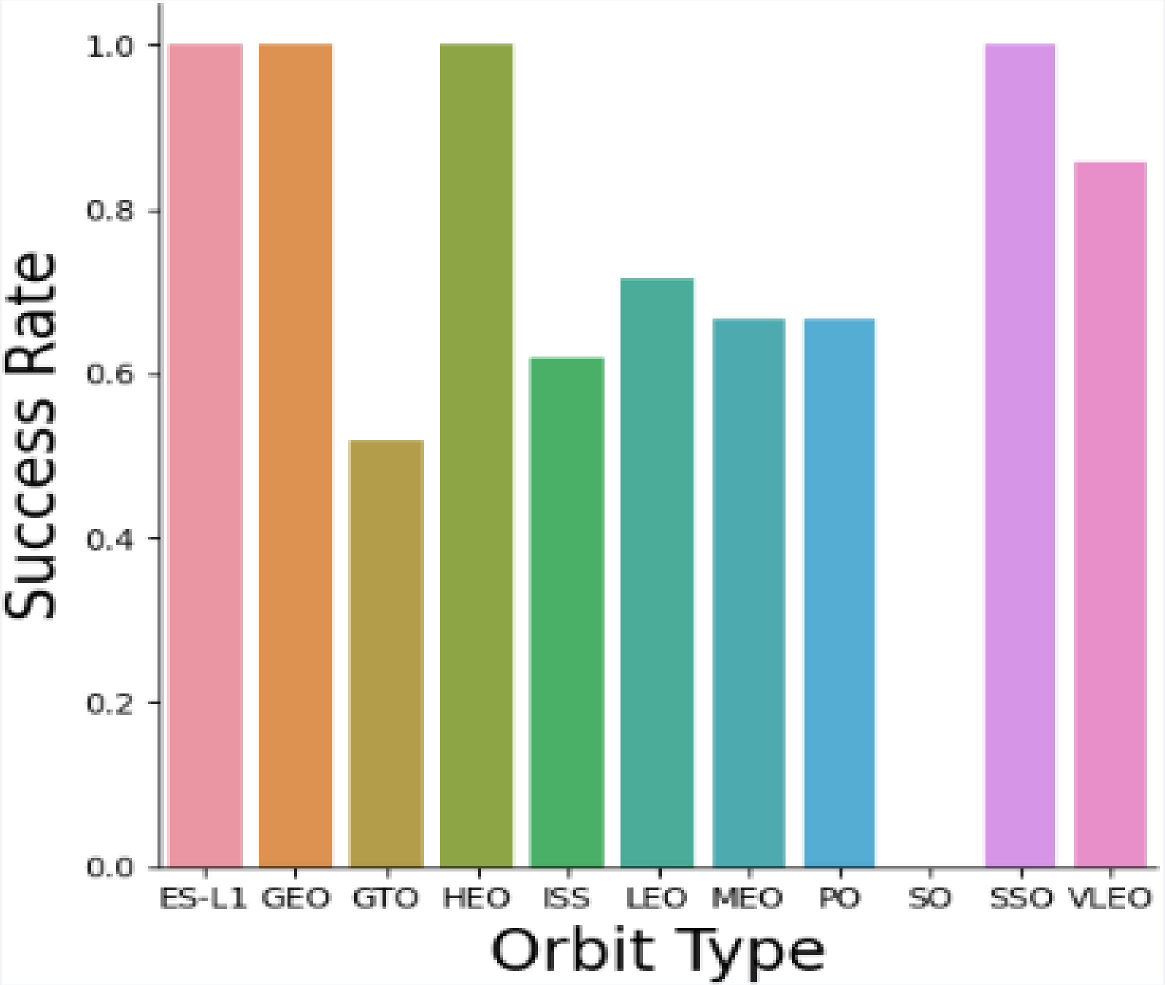
- We observe that the higher the payload mass (kg), the higher the success rate. Therefore, most rocket launches with a payload greater than 7,000 kg are seen to be successful.
- KSC LC 39A has a 100% success rate for launches less than 6,000 kg.
- VAFB SKC 4E has not launched anything greater than ~10,000 kg



Success Rate vs. Orbit Type

Bar chart of Orbital Success Rate of each orbit type.

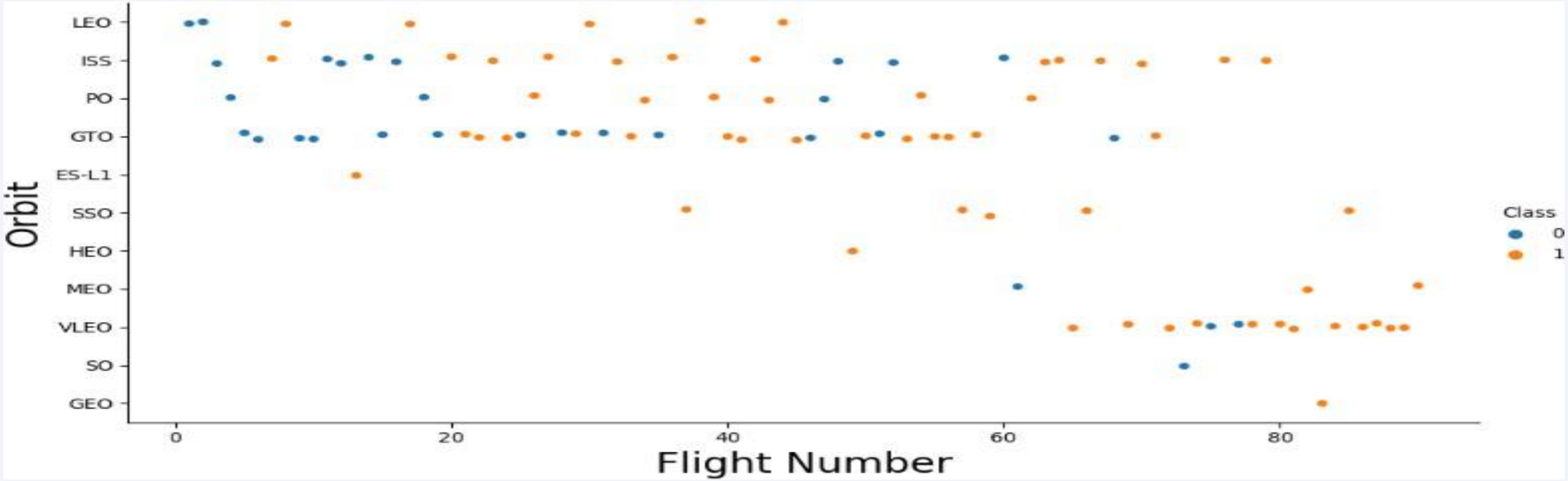
- 100% Success Rate: ES-L1, GEO, HEO and SSO
- 50%-80% Success Rate: GTO, ISS, LEO, MEO, PO
- 0% Success Rate: SO



Flight Number vs. Orbit Type

Exploratory Data Analysis

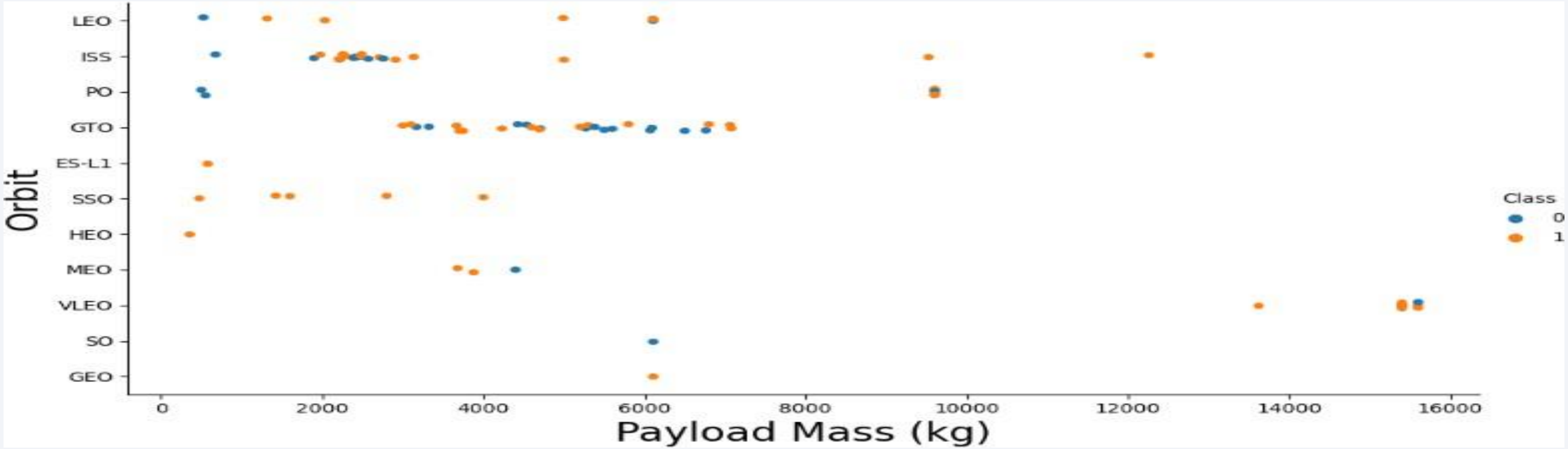
- The success rate typically increases with the number of flights for each orbit
- This relationship is highly apparent for the LEO orbit, however the GTO orbit, does not follow this trend



Payload vs Orbit Type

Exploratory Data Analysis

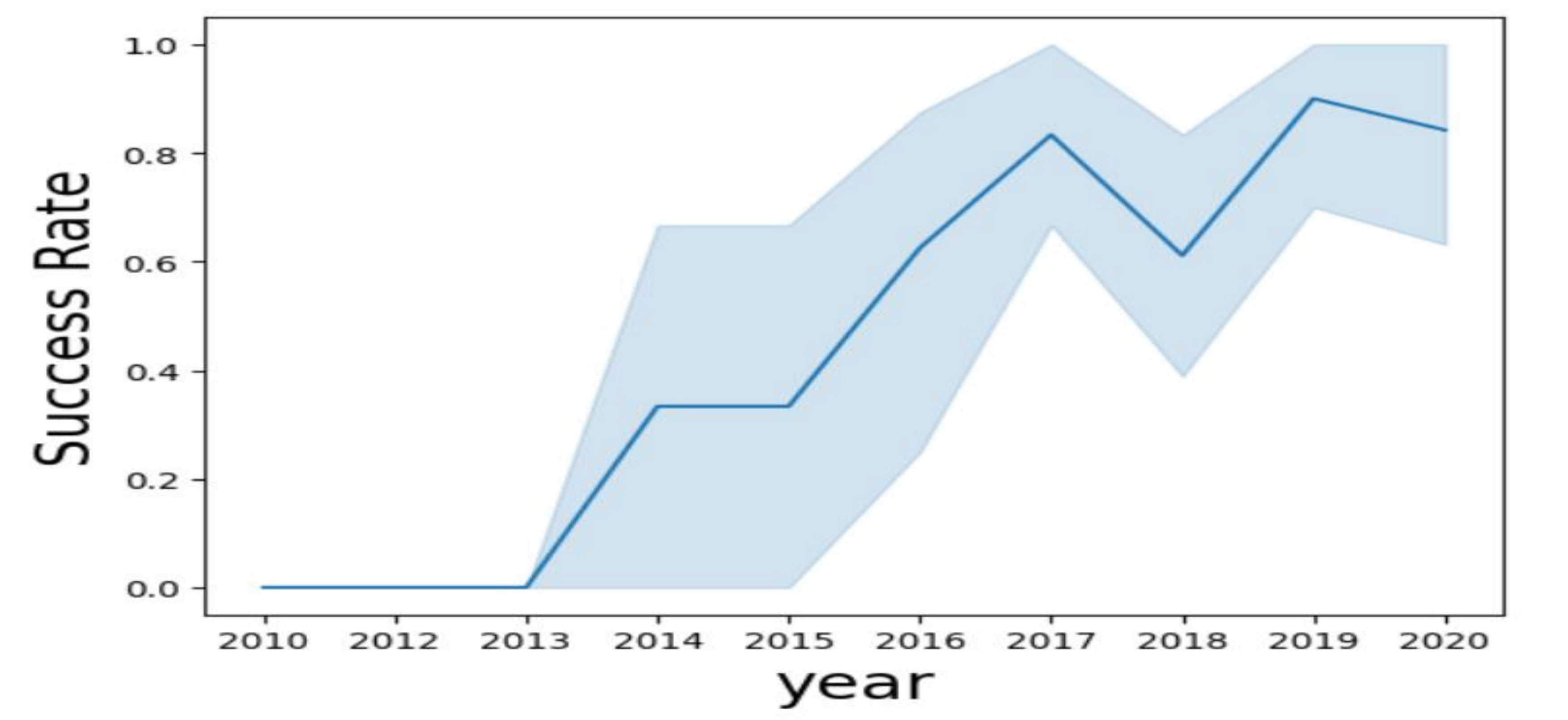
- Heavier payloads are better with LEO, ISS and PO orbits. The reason for this could be because of the close proximity to the earth, and shorter flight time.
- The GTO orbit has mixed results with heavier payloads. Hence, further analysis is needed to arrive at a definite conclusion.



Launch Success Yearly Trend

Exploratory Data Analysis

- The success rate steadily improved from 2013-2020 with a slight dip in 2018.



All Launch Site Names and Sites Beginning with CCA

- Find the names of the Unique launch sites
  - %%sql
  - SELECT DISTINCT(launch_site) FROM SPACEXDATA;

Result: CCAFS LC-40, CCAFS SLC-40,

KSC LC-39A, VAFB SLC-4E

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

```
%%sql
SELECT DISTINCT(launch_site) FROM SPACEXDATA;
```

```
 * postgresql://postgres:***@localhost/SpaceX_Research
4 rows affected.
```

5]:

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

Find 5 records where launch sites begin with `CCA`

- %%sql
- select * from spacexdata where launch_site like ('CCA%')
- limit 5;

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

```
%%sql
select * from spacexdata where launch_site like ('CCA%')
limit 5;
```

```
 * postgresql://postgres:***@localhost/SpaceX_Research
5 rows affected.
```

5]:

date	time_(utc)	booster_version	launch_site	payload	payload_m
2010-04-06	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	
2010-08-12	15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight C1, two CubeSats, barrel of Brouere cheese	
2012-05-22	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	
2012-08-10	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	
2013-01-03	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	

Total Payload Mass & Average Payload Mass by F9 v1.1

- **Calculate the total payload carried by boosters from NASA**
  - %%sql
  - select SUM(PAYLOAD_MASS_KG) as "Total Payload" FROM SPACEXDATA WHERE CUSTOMER = 'NASA (CRS)';

  Total Payload = 45,596KG

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

```
%%sql

select SUM(PAYLOAD_MASS_KG) as "Total Payload" FROM SPACEXDATA WHERE CUSTOMER = 'NASA (CRS)';

 * postgresql://postgres:***@localhost/SpaceX_Research
1 rows affected.
```

7]:

Total Payload
45596

- **Calculate the average payload mass carried by booster version F9 v1.1**
  - %%sql
  - SELECT ROUND(AVG(PAYLOAD_MASS_KG),2) as "Average Payload" FROM SPACEXDATA WHERE BOOSTER_VERSION = 'F9 v1.1';

  Average Payload = 2,928.40KG

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

```
%%sql
SELECT ROUND(AVG(PAYLOAD_MASS_KG),2) as "Average Payload" FROM SPACEXDATA WHERE BOOSTER_VERSION = 'F9 v1.1';

 * postgresql://postgres:***@localhost/SpaceX_Research
1 rows affected.
```

]:

Average Payload
2928.40

First Successful Ground Landing Date

1st Successful Landing in Ground Pad

- 12/22/2015

```
%%sql
SELECT MIN(DATE) as "First Successful Landing" FROM SPACEXDATA WHERE LANDING_OUTCOME = 'Success (ground pad)'
```

```
 * postgresql://postgres:***@localhost/SpaceX_Research
1 rows affected.
```

First Successful Landing
2015-12-22

Total Number of Successful and Failed Mission Outcomes

- 1 Failure in Flight
- 99 Success
- 1 Success (payload status unclear)

```
%sql SELECT MISSION_OUTCOME, COUNT(*) as total_number FROM SPACEXDATA GROUP BY MISSION_OUTCOME;
```

```
 * postgresql://postgres:***@localhost/SpaceX_Research
4 rows affected.
```

mission_outcome	total_number
Success (payload status unclear)	1
Success	98
Success	1
Failure (in flight)	1

Booster Drone Ship Landing

- Booster mass greater than 4,000 but less than 6,000
- JSCAT-14, JSCAT-16, SES-10, SES-11 / EchoStar 105

List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000

```
%%sql
SELECT PAYLOAD, PAYLOAD_MASS_KG, LANDING_OUTCOME FROM SPACEXDATA WHERE LANDING_OUTCOME LIKE 'Success (drone ship)'
AND payload_mass_kg BETWEEN 4000 AND 6000;
```

```
 * postgresql://postgres:***@localhost/SpaceX_Research
4 rows affected.
```

payload	payload_mass_kg	landing_outcome
JCSAT-14	4696	Success (drone ship)
JCSAT-16	4600	Success (drone ship)
SES-10	5300	Success (drone ship)
SES-11 / EchoStar 105	5200	Success (drone ship)

Boosters Carried Maximum Payload

List of Boosters that lifted maximum weight

- 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

```
%%sql
SELECT BOOSTER_VERSION FROM SPACEXDATA WHERE
PAYLOAD_MASS_KG = (SELECT MAX(PAYLOAD_MASS_KG) FROM SPACEXDATA);
```

```
 * postgresql://postgres:***@localhost/SpaceX_Research
12 rows affected.
```

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

Launch Records and Landing Outcomes between 2010 - 2017

- 2015 LAUNCH RECORDS

  Showing two failed landings, booster version, and launch site for the months of April and October 2015.

```
%%sql
SELECT EXTRACT('MONTH' FROM Date) AS MONTH,BOOSTER_VERSION, LAUNCH_SITE, Landing_Outcome
FROM SPACEXDATA where Landing_Outcome = 'Failure (drone ship)'
AND EXTRACT(YEAR FROM DATE) ='2015';
```

```
 * postgresql://postgres:***@localhost/SpaceX_Research
2 rows affected.
```

month	booster_version	launch_site	landing_outcome
10	F9 v1.1 B1012	CCAFS LC-40	Failure (drone ship)
4	F9 v1.1 B1015	CCAFS LC-40	Failure (drone ship)

- Landings between 2010 – 2017

  Showing count of landing outcomes between 2010-06-04 and 2017-03-20 in descending order.

```
%%sql
SELECT Landing_Outcome, count(*) as Outcomes FROM SPACEXDATA
WHERE DATE BETWEEN '2010-06-04' AND '2017-03-20'
GROUP BY Landing_Outcome
ORDER BY Outcomes DESC;
```

```
 * postgresql://postgres:***@localhost/SpaceX_Research
8 rows affected.
```

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

LAUNCH SITE PROXIMITIES

Launch Sites

Launch sites are situated based on the following factors:

1. **Proximity to the Ocean:**

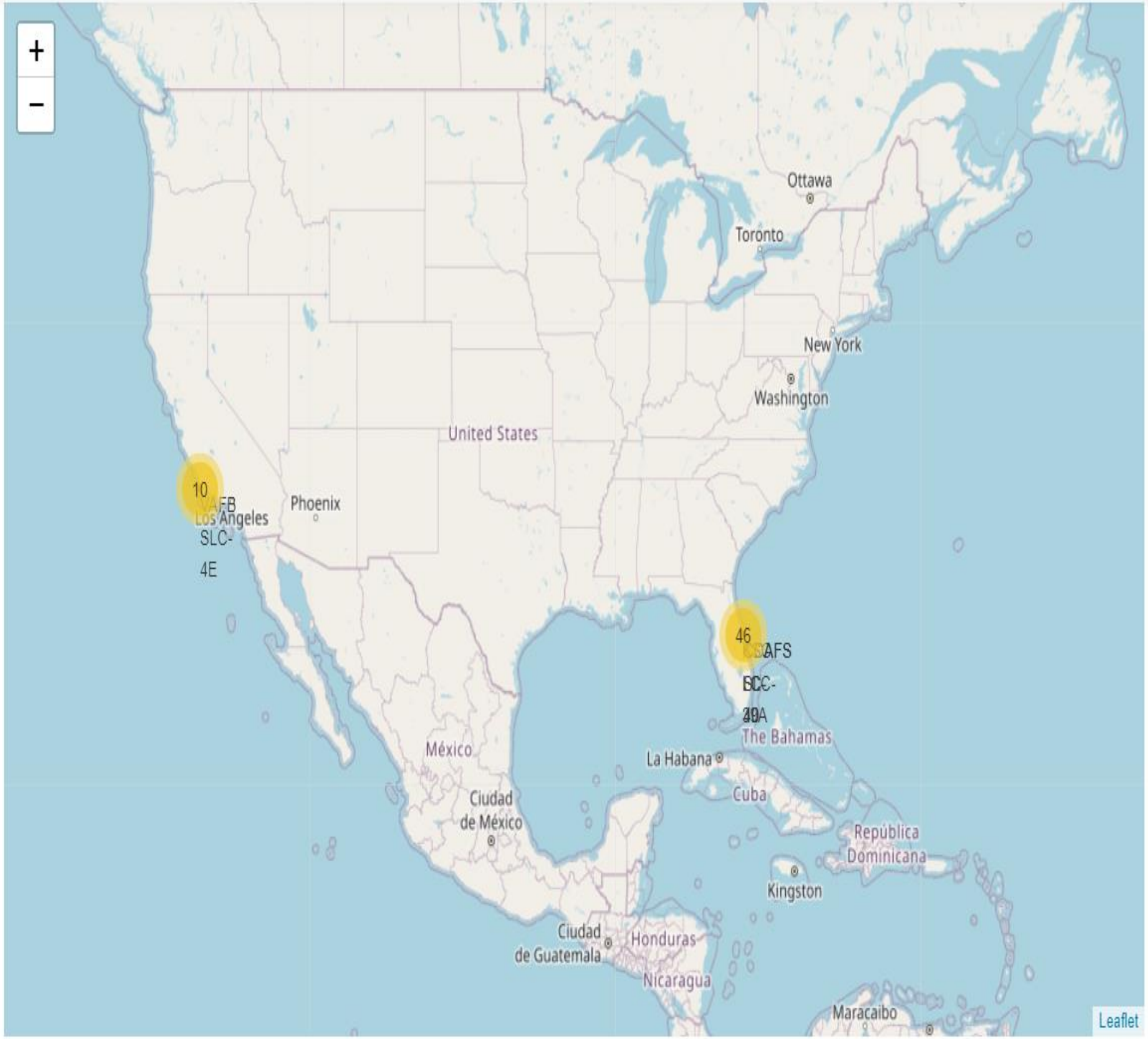
The coasts and ocean provides a safe trajectory over unpopulated areas, and in the event of a booster malfunction, the rocket can be put down safely in the ocean. Cape Canaveral, Fl. and Vandenberg AFB, CA are located on the Atlantic and Pacific coasts respectively.

2. **Speed boost from the Earth's rotation:**

SPACEX takes advantage of Earth's natural rotation by launching toward the east from Cape Canaveral, Fl.

The speed at which the Earth rotates at Cape Canaveral is roughly 914 mph, helping to give rockets some extra speed to reach their destination resulting is less fuel burnt.

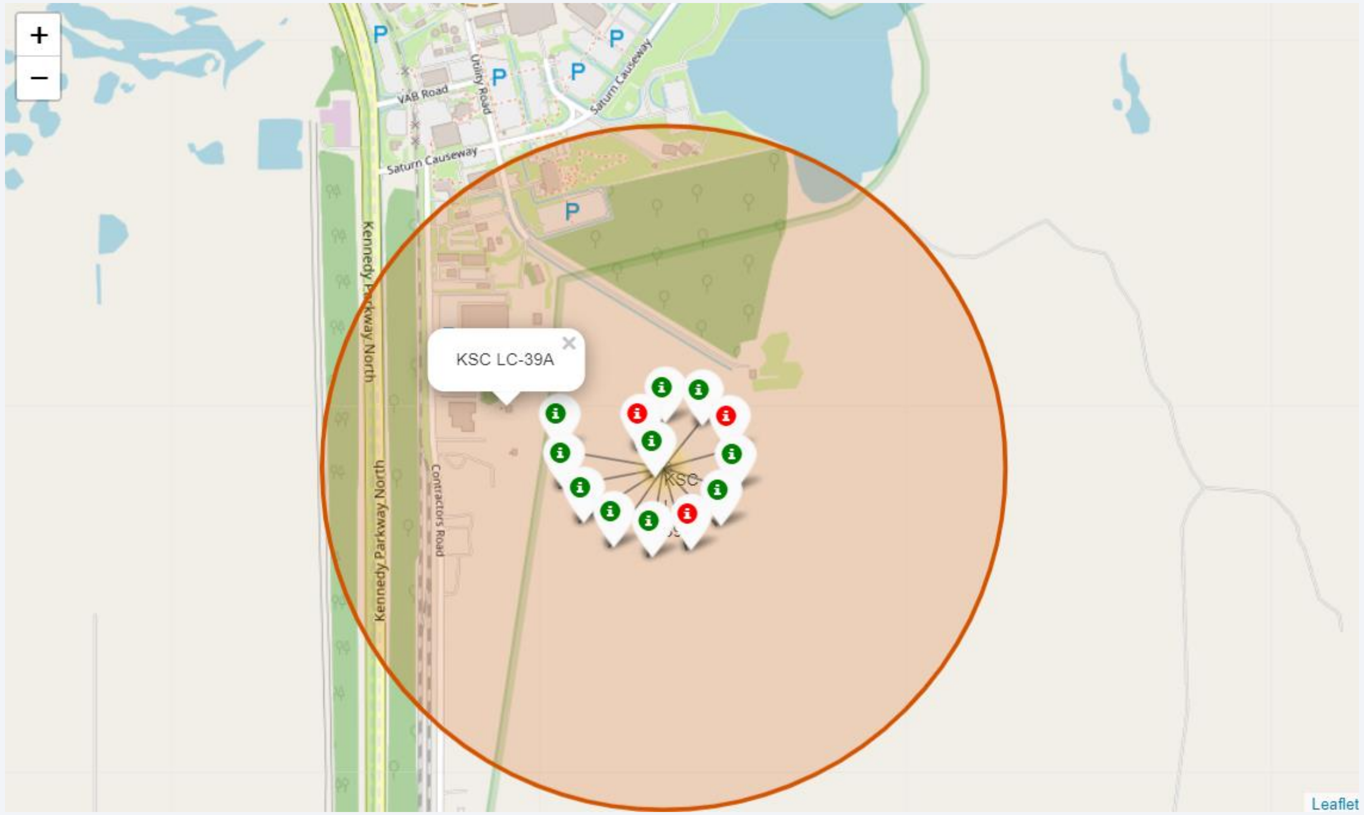
These combined factors are why Florida has been the preferred site for rocket launches lately with **46** launches at Cape Canaveral compared to just **10** launches at Vandenberg, CA.



Launch Outcomes

Launch Outcome At Each Launch Site:

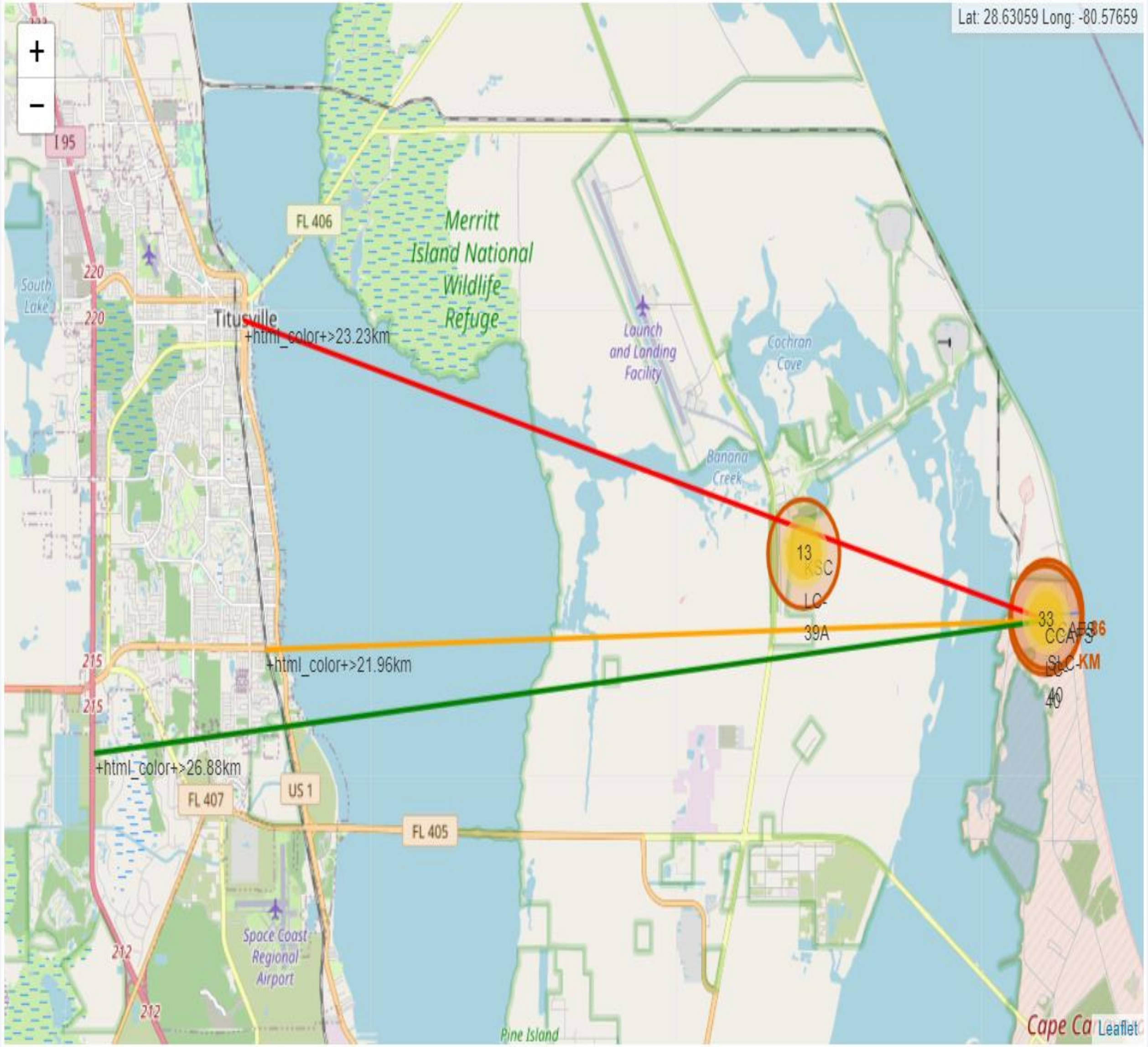
- Green markers for successful launches
- Red markers for unsuccessful launches
- Launch site **KSC LC-39A** has a **10/13 success rate** (76.9%)



Launch Site Proximity To Landmarks

CCAFS SLC-40 is situated at the following proximities

- .86 km from nearest coastline
- 21.96 km from nearest railway
- 23.23 km from nearest city
- 26.88 km from nearest highway
- **Coasts**: provides a safe environment for the recovery of spent boosters/1st stage OR failed launches.
- **Safety / Security:** needs to be an exclusion zone around the launch site to keep unauthorized people away and keep people safe.
- **Infrastructure**: Railway and road networks provide a means of supplying the launch site with materials and personnel to support space exploration.




DASHBOARDS & ANALYTICS

Distribution of Launch Success by Site

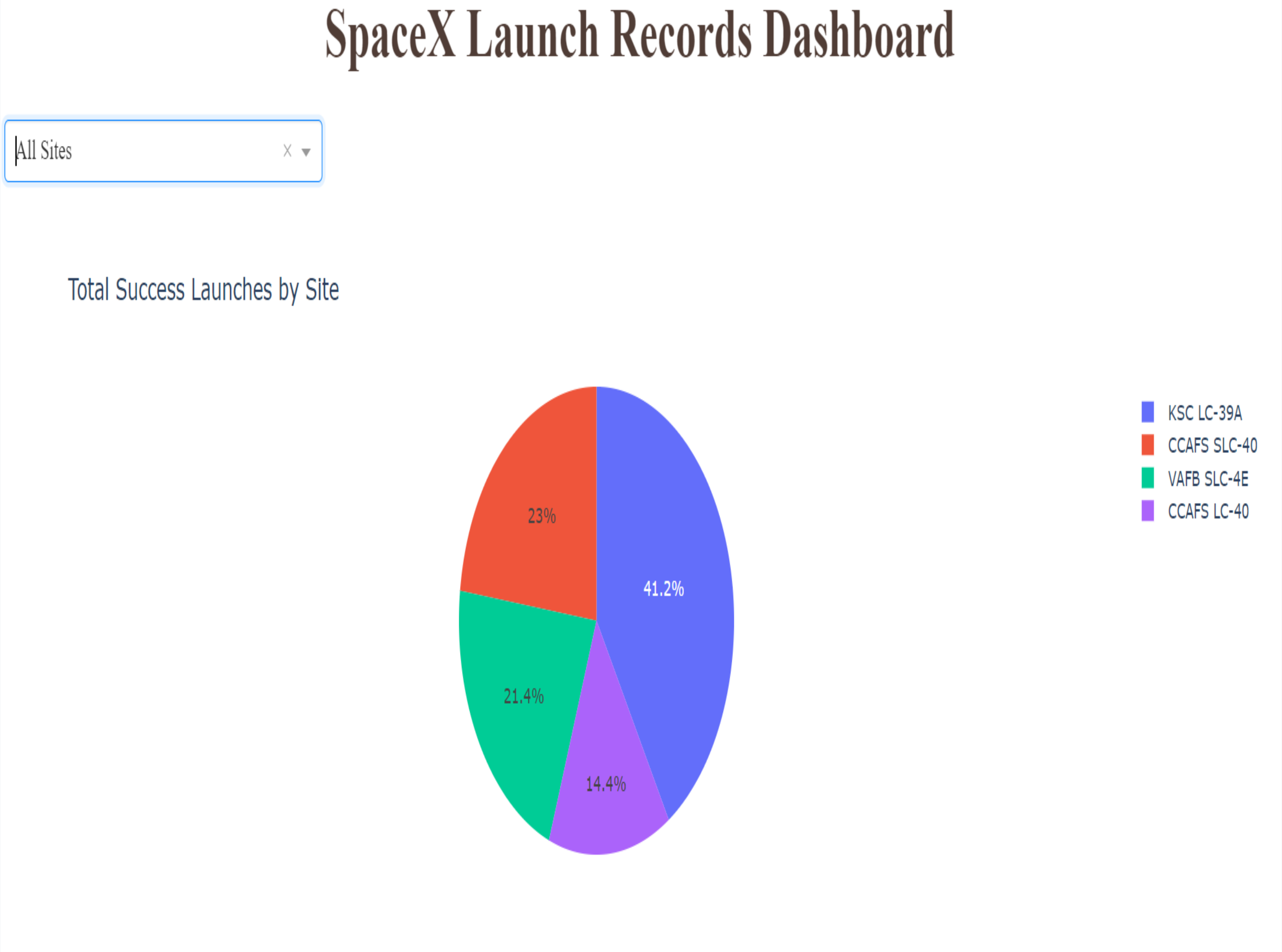
Rocket Launches on the East coast have an overall success rate of 78.6% while Vandenberg Launch Site on the west coast has 21.4% positive outcome.

<u>**East Coast Launches:**</u>

KSC LC-39A site accounts **for 52%** of successful outcomes at **Cape Canaveral**.

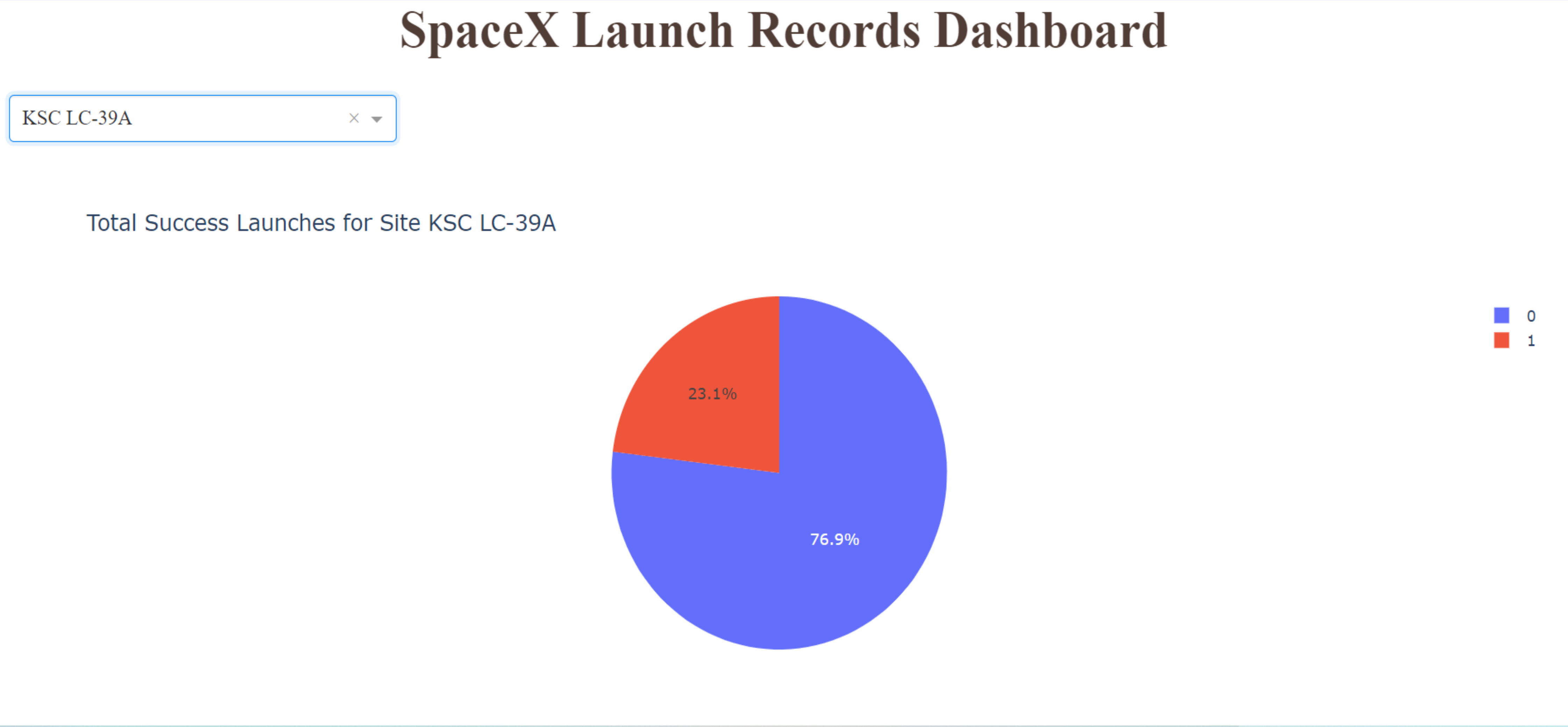
CCAFS SLC-40 site comes in second with **29%** of successful outcomes.

The data tell us that there is preference for the Cape Canaveral site because of its eastly position and proximity to the equator which guarantees a higher chance of success.



Launch Success to Failure Ratio – KSC LC-39A

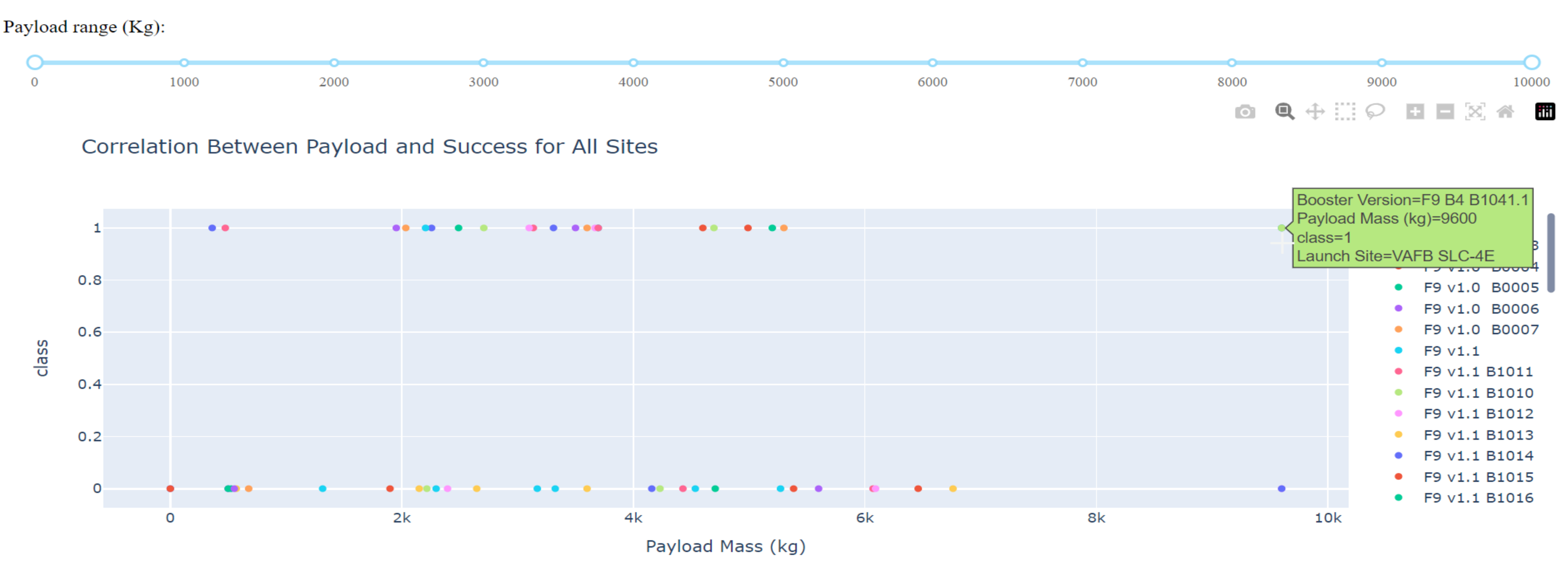
This shows that for every 10 rocket launches at any given time, 7 have successful outcomes while 3 either malfunctioned of failed totally.



Payload Mass vs Launch Outcomes

- Only one booster, F9 B4 B1041.1 successfully lifted a payload in excess of 9,500kg.
- The bulk of Payloads successfully launched is between 2000kg – 6000kg

Therefore, it is safe to assume that the recommended maximum payload should be capped at 6,000kg.



PREDICTIVE ANALYSIS

Classification Accuracy

- All the models performed at about the same level and had the same scores and accuracy. This is likely due to the small dataset. The Decision Tree model slightly outperformed the rest when looking at .best_score
- ***.best_score*** *is the average of all cv folds for a single combination of the parameters*

	LogReg	SVM	Tree	KNN
Jaccard_Score	0.800000	0.800000	0.800000	0.800000
F1_Score	0.888889	0.888889	0.888889	0.888889
Accuracy	0.833333	0.833333	0.833333	0.833333

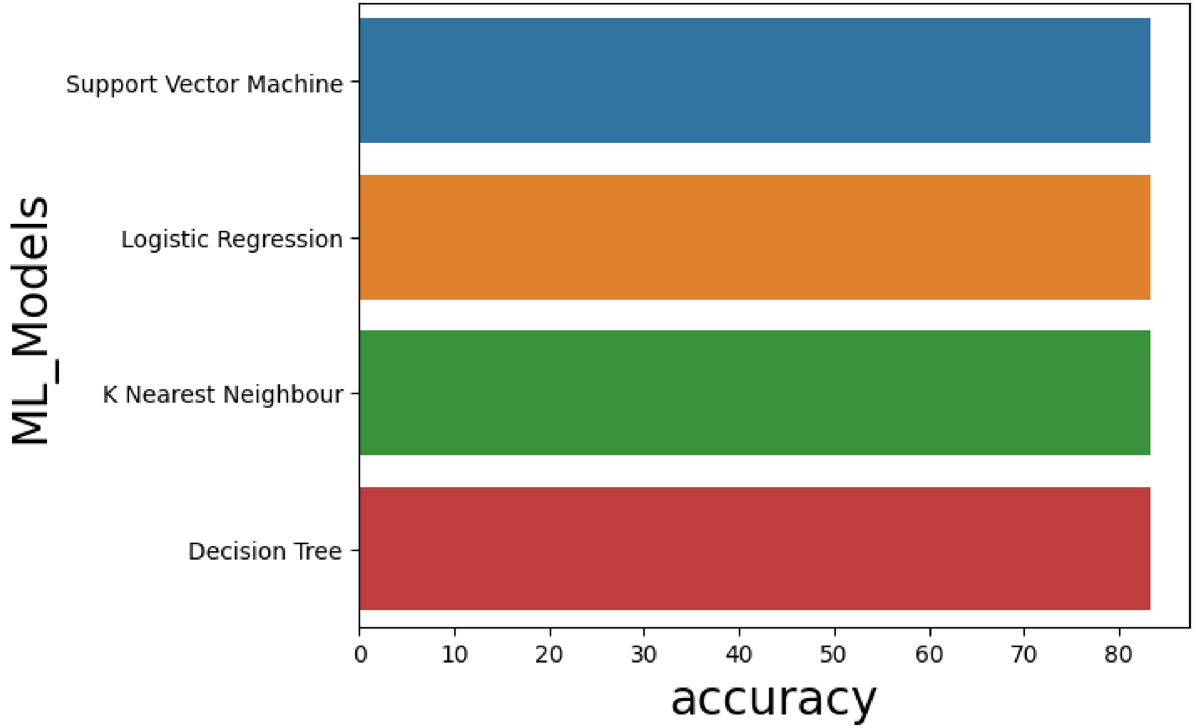
```
models = {'KNeighbors':knn_cv.best_score_,
              'DecisionTree':tree_cv.best_score_,
              'LogisticRegression':logreg_cv.best_score_,
              'SupportVector': svm_cv.best_score_}

bestalgorithm = max(models, key=models.get)
print('Best model is', bestalgorithm,'with a score of', models[bestalgorithm])
if bestalgorithm == 'DecisionTree':
    print('Best params is :', tree_cv.best_params_)
if bestalgorithm == 'KNeighbors':
    print('Best params is :', knn_cv.best_params_)
if bestalgorithm == 'LogisticRegression':
    print('Best params is :', logreg_cv.best_params_)
if bestalgorithm == 'SupportVector':
    print('Best params is :', svm_cv.best_params_)
```

```
Best model is DecisionTree with a score of 0.8875000000000002
Best params is : {'criterion': 'gini', 'max_depth': 8, 'max_features': 'sqrt', 'min_samples_leaf': 2, 'min_samples_split': 5, 'splitter': 'best'}
```

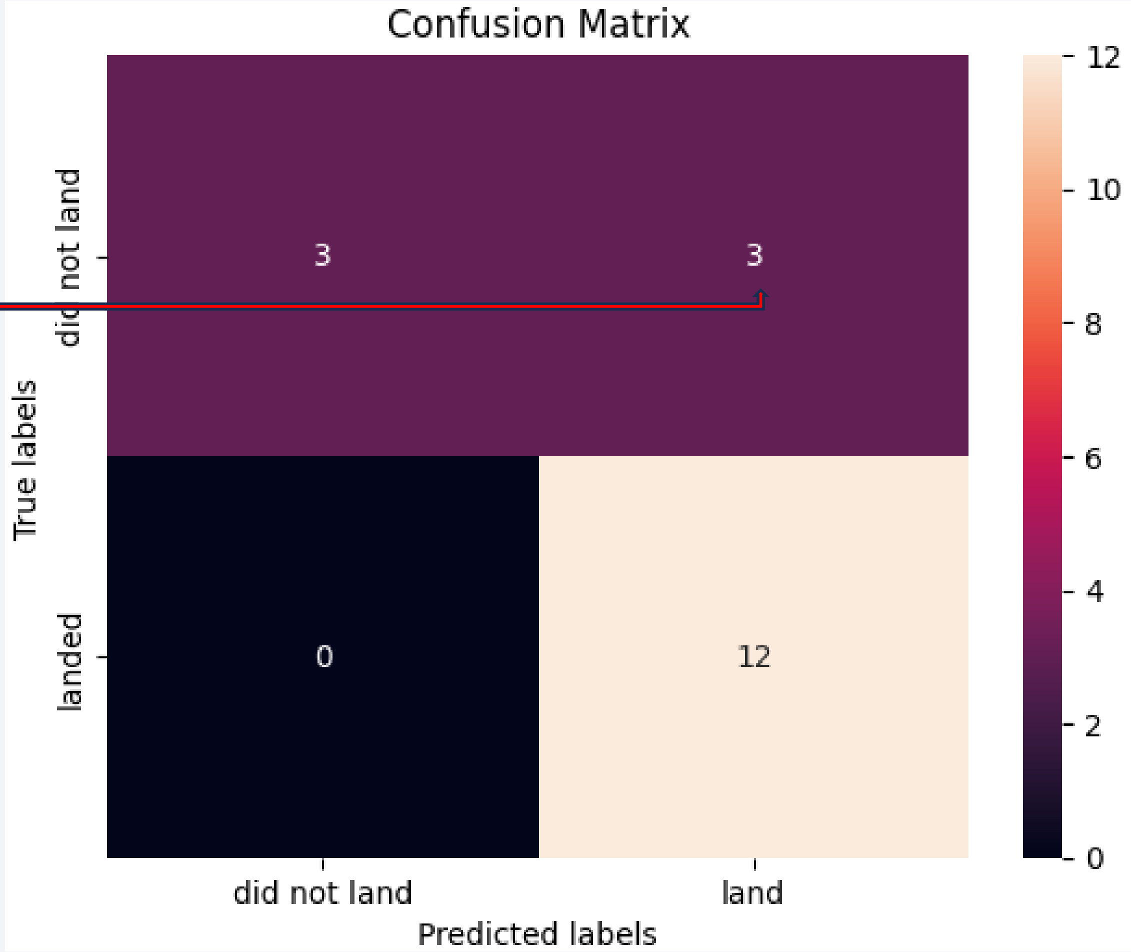
	ML Algorithm	Accuracy Score (%)
0	Support Vector Machine	83.333333
1	Logistic Regression	83.333333
2	K Nearest Neighbour	83.333333
3	Decision Tree	83.333333

```
g=sns.barplot(x= accuracy, y = method, data = pd.DataFrame(models))
plt.xlabel('accuracy',fontsize=20)
plt.ylabel('ML_Models',fontsize=20)
plt.show()
```



Confusion Matrix

- A confusion matrix summarizes the performance of a classification algorithm. All confusion matrices were identical.
- The fact that there are false positives (Type 1 error) is not a good indicator.
- Confusion Matrix Outputs:
  - 12 True positive
  - 3 True negative
  - **3 False positive**
  - 0 False Negative
- **Precision** = TP / (TP + FP)
  - 12 / 15 = .80
- **Recall** = TP / (TP + FN)
  - 12 / 12 = 1
- **F1 Score** = 2 * (Precision * Recall) / (Precision + Recall)
  - 2 * (.8 * 1) / (.8 + 1) = .89
- **Accuracy** = (TP + TN) / (TP + TN + FP + FN) = .833



Conclusions

All four models performed within the same range, however, the Decision Tree Model is recommended to be the best performing model within the range.

We also gained valuable insights on the impact of the following parameters on the successful mission outcome of a flight:

- **Launch Site Location**: Launch sites located on the eastern seaboard and close to the equator have a higher successful mission outcome which can be attributed to the shorter flight time and equatorial boost.
- **Mission Success**: Increases over time based on the frequency of flights.
- **KSC LC-39A Launch Site**: Has a 100% success rate for launches less than 5,500kg.
- **Orbits**: ES-L1, GEO, HEO, and SSO have a 100% success rate.
- **Payload Mass**: The higher the payload mass (kg), the higher the success rate.

Recommendations

In order to improve on the accuracy of the models, it is recommended that more data be included in the training samples.

Appendix

- Inserted is the plotly dash app .py code.
- https://github.com/Hunting007/SpaceX_Capstone/blob/main/Plotly_Dash_APP_SpaceX_Capstone.txt

Plotly_Dash_Code

Thank you!