
# Winning Space Race with Data Science

Yousef Ayman
01/07/2023

# Outline


| | |
|---|---|
| P3 | • Executive Summary |
| P4 | • Introduction |
| P6 | • Methodology |
| P16 | • Results |
| P17 | • EDA with Visualization |
| P23 | • EDA with SQL |
| P30 | • Interactive Maps with Folium |
| P35 | • Plotly Dash Dashboard |
| P39 | • Predictive Analytics |
| P41 | • Conclusion |

# Executive Summary

## Summary of Methodologies

The research attempts to identify the factors for a successful rocket landing. To make this determination, the following methodologies where used:

- **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
- **Explore** launch site success rates and proximity to geographical markers
- **Analyze** the data with SQL, calculating the following statistics: total payload, payload range for successful launches, and total # of successful and failed outcomes
- **Visualize** the launch sites with the most success and successful payload ranges
- **Build Models** to predict landing outcomes using logistic regression, support vector machine (SVM), decision tree and K-nearest neighbor (KNN)

## Results

### Exploratory Data Analysis:

- Launch success has improved over time
- KSC LC-39A has the highest success rate among landing sites
- Orbits ES-L1, GEO, HEO, and SSO have a 100% success rate

### Visualization/Analytics:

- Most launch sites are near the equator, and all are close to the coast

### Predictive Analytics:

- All models performed similarly on the test set. The decision tree model slightly outperformed

# Introduction

**Background**

SpaceX, a leader in the space industry, strives to make space travel affordable for everyone. Its accomplishments include sending spacecraft to the international space station, launching a satellite constellation that provides internet access and sending manned missions to space.

SpaceX can do this because the rocket launches are relatively inexpensive ($62 million per launch) due to its novel reuse of the first stage of its Falcon 9 rocket. Other providers, which are not able to reuse the first stage, cost upwards of $165 million each.

By determining if the first stage will land, we can determine the price of the launch. To do this, we can use public data and machine learning models to predict whether SpaceX –or a competing company –can reuse the first stage.

**Explore**

- How payload mass, launch site, number of flights, and orbits affect first-stage landing success
- Rate of successful landings over time
- Best predictive model for successful landing (binary classification)

Section 1

# Methodology

# Methodology


## Steps

- **Collect** data using SpaceX REST API and web scraping techniques
- **Wrangle** data – by filtering the data, handling missing values and applying one hot encoding – to prepare the data for analysis and modeling
- **Explore** data via EDA with SQL and data visualization techniques
- **Visualize** the data using Folium and Plotly Dash
- **Build Models** to predict landing outcomes using classification models. Tune and evaluate models to find best model and parameters

# Data Collection - Web Scraping


## Steps

- **Request data** from SpaceX API (rocket launch data)
- **Decode response** using .json() and convert to a dataframe using .json_normalize()
- **Request information** about the launches from SpaceX API using custom 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 - Web Scraping


## Steps

- **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

# Data Wrangling

**Steps**

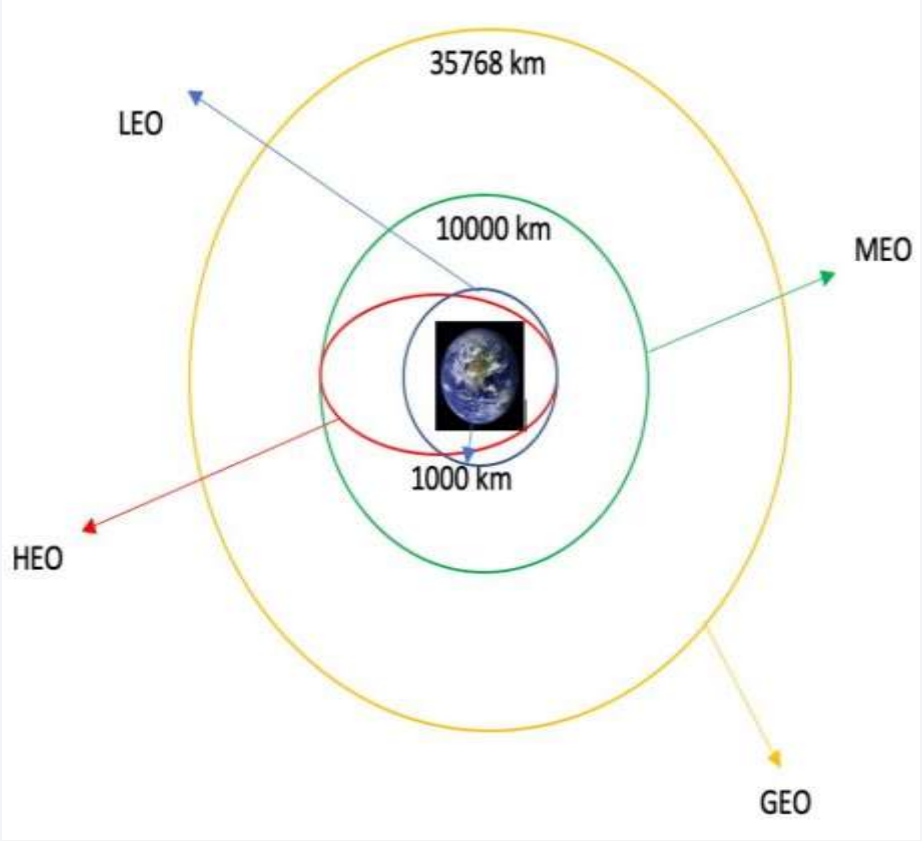
- **Perform EDA** and determine data labels
- **Calculate:**
  - #of launches for each site
  - #and occurrence of orbit
  - #and occurrence of mission outcome per orbit type]
- **Create binary** landing outcome column (dependent variable)
- **Export data** to csv file

**Landing Outcome**

- Landing was not always successful
- **True Ocean:** mission outcome had a successful landing to a specific region of the ocean

**Landing Outcome Cont.**

- **False Ocean:** represented an unsuccessful landing to a specific region of ocean
- **True RTLS:** meant the mission had a successful landing on a ground pad
- **False RTLS:** represented an unsuccessful landing on a ground pad
- **True ASDS:** meant the mission outcome had a successful landing on a drone ship
- **False ASDS:** represented an unsuccessful landing on drone ship
- **Outcomes converted** into 1 for a successful landing and 0 for an unsuccessful landing



# EDA with Data Visualization


## Charts

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

## Analysis

- **View relationship** by using **scatter plots**. The variables could be useful for machine learning if a relationship exists
- **Show comparisons** among discrete categories with **bar charts**. Bar charts
- show the relationships among the categories and a measured value.

# EDA with SQL


**Display**

- Names of unique launch sites
- 5 records where 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.

**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)

# 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** (**green**) and **unsuccessful** (**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**

# Build a Dashboard with Plotly Dash

**Dropdown List with Launch Sites**

- Allow user to select all launch sites or a certain launch site

**Pie Chart Showing Successful Launches**

- Allow user to see successful and unsuccessful launches as a percent of the total

**Slider of Payload Mass Range**

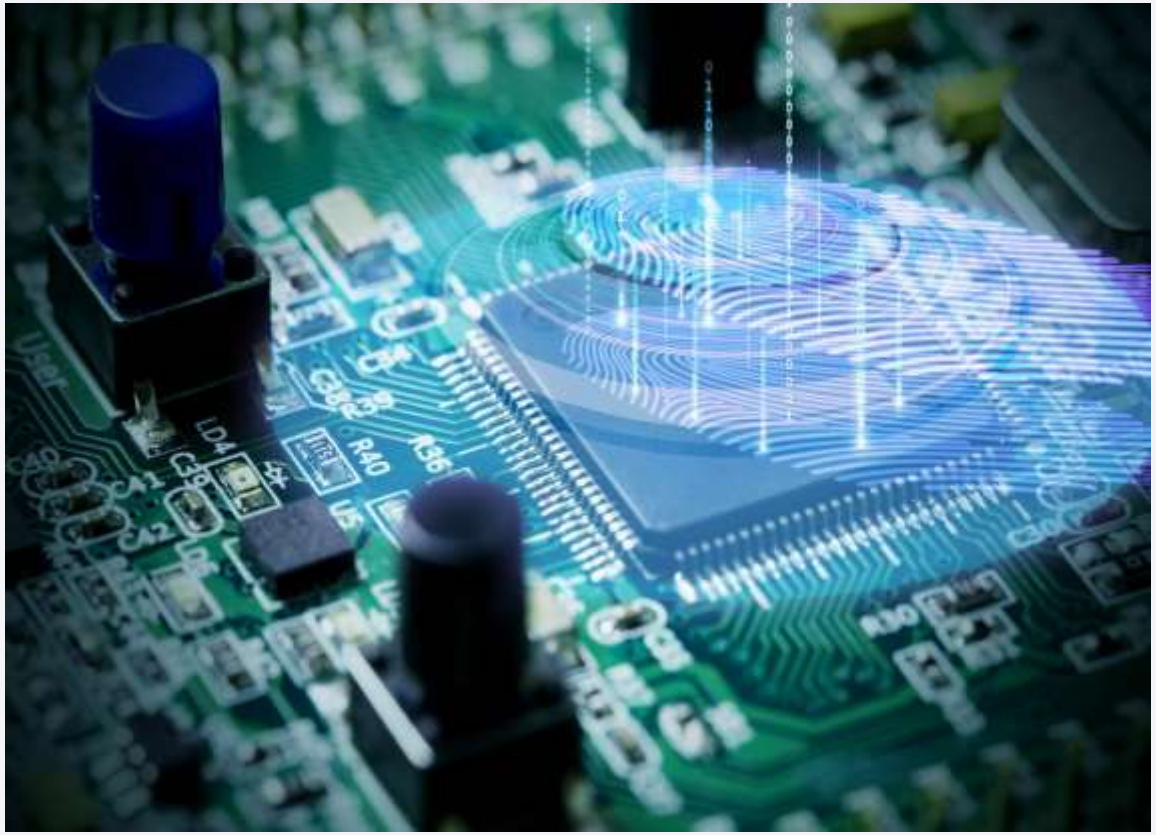
- Allow user to select payload mass range

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

- Allow user to see the correlation between Payload and Launch Success

# Predictive Analysis (Classification)

| | |
|---|---|
| Create | Create NumPy array from the Class column |
| Standardize | Standardize the data with StandardScaler. Fit and transform the data. |
| Split | Split the data using train_test_split |
| Create | Create a GridSearchCV object with cv=10 for parameter optimization |
| Apply | Apply GridSearchCV on different algorithms: logistic regression, support vector machine, decision tree and K-Nearest Neighbour |
| Calculate | Calculate accuracy on the test data using .score() for all models |
| Assess | Assess the confusion matrix for all models |
| Identify | Identify the best model using F1_Score and Accuracy |



# Results

**Exploratory Data Analysis**

- Launch success has improved over time
- KSC LC-39A has the highest success rate among landing sites
- Orbits ES-L1, GEO, HEO and SSO have a 100% success rate

**Visual Analytics**

Most launch sites are near the equator, and all are close to the coast

Launch sites are far enough away from anything a failed launch can damage (city, highway, railway), while still close enough to bring people and material to support launch activities

**Predictive Analytics**

Decision Tree model is the best predictive model for the dataset

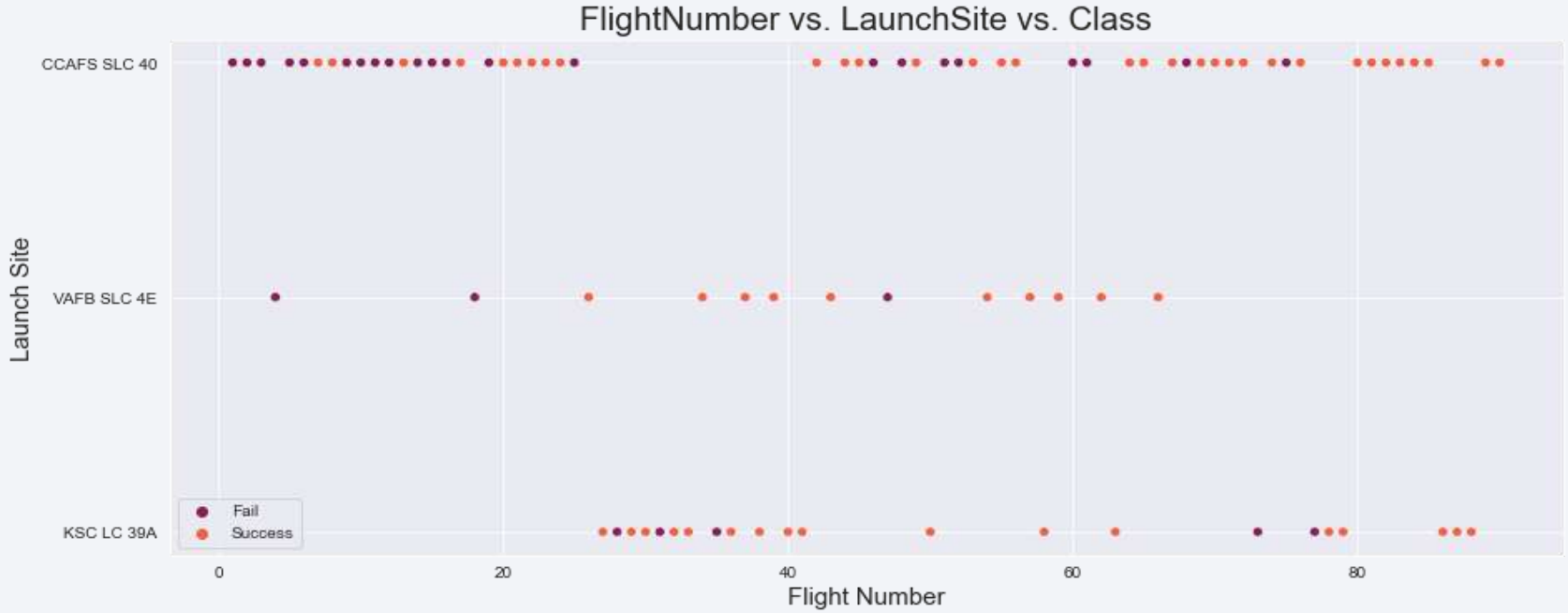
Section 2

# Insights drawn from EDA

# Flight Number vs. Launch Site

**Exploratory Data Analysis**

- **Earlier flights** had a **lower success rate** (**purple = fail**)
- **Later flights** had a **higher success rate** (**orange = success**)
- Around half of launches were from CCAFS SLC 40 launch site
- VAFB SLC 4E and KSC LC 39A have higher success rates
- We can infer that new launches have a higher success rate

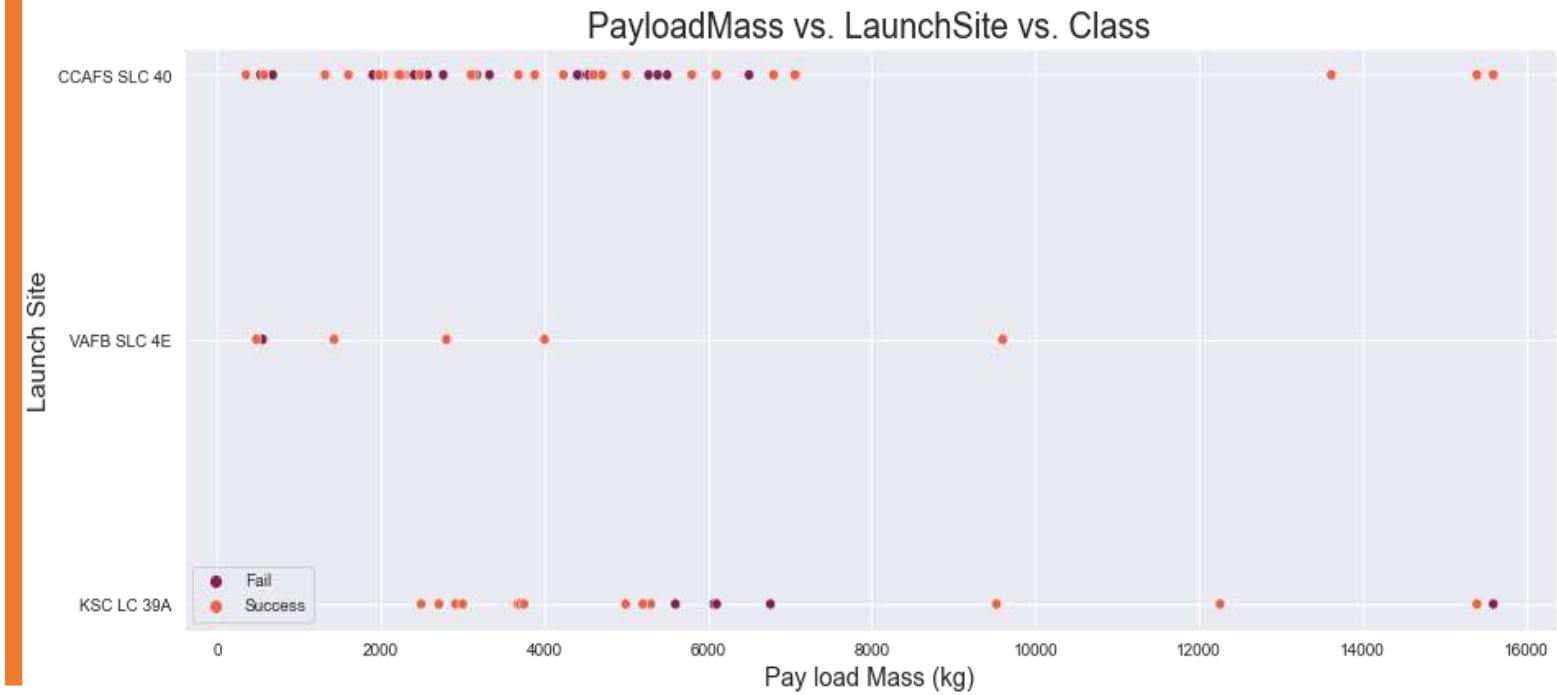


# Payload vs. Launch Site

**Exploratory Data Analysis**

- Typically, the **higher** the **payload mass** (kg), the **higher** the **success rate**
- Most launces with a payload greater than 7,000 kg were successful
- KSC LC 39A has a 100% success rate for launches less than 5,500 kg
- VAFB SKC 4E has not launched anything greater than ~10,000 kg

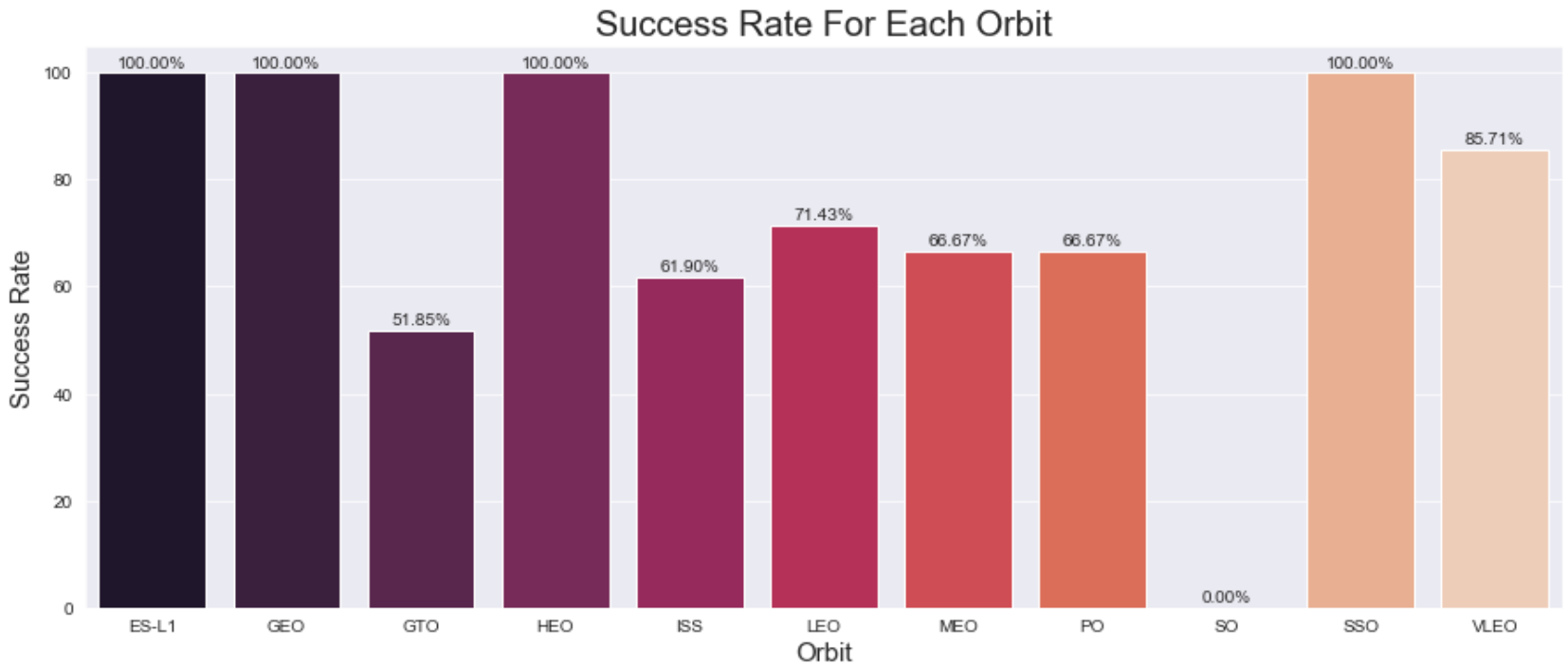
The greater the payload mass for launch site CCAFS SLC 40 the higher the success rate for the rocket.



# Success Rate vs. Orbit Type

**Exploratory Data Analysis**

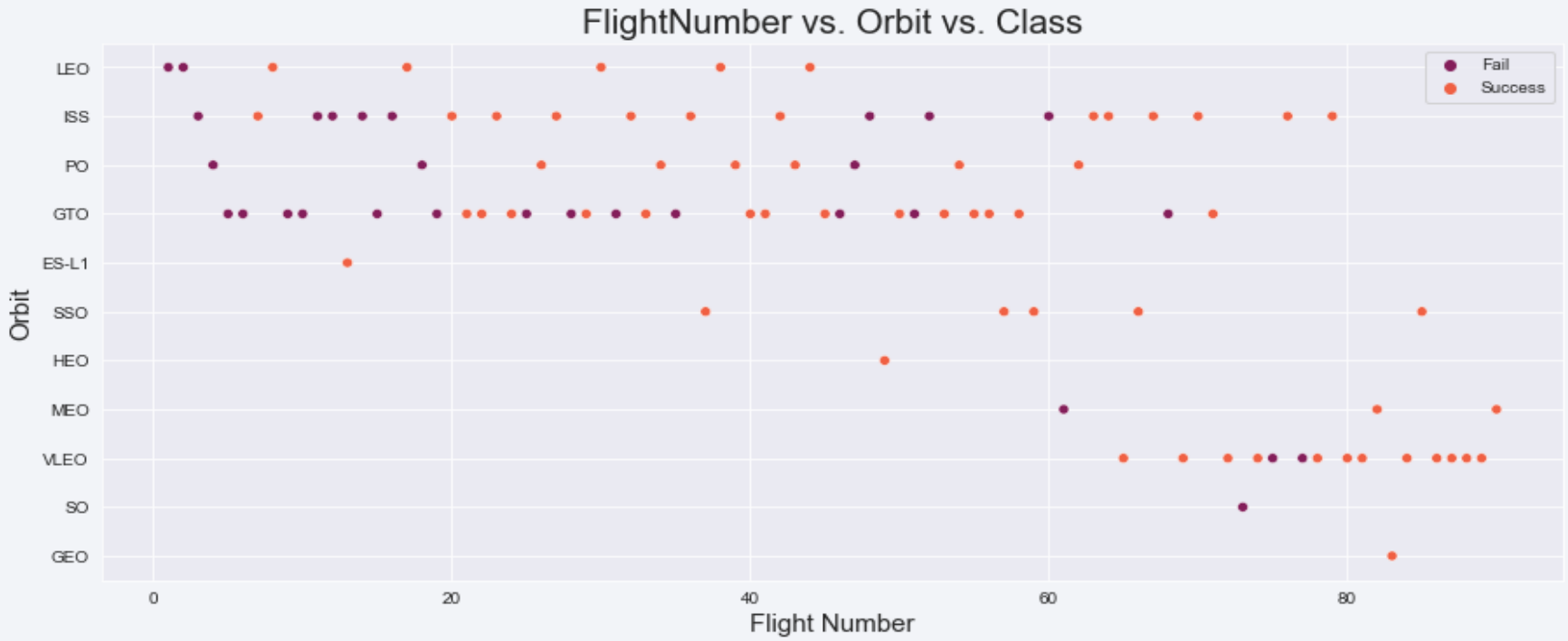
- **100%Success Rate**: ES-L1, GEO, HEO and SSO
- **50%-80%Success Rate**: GTO, ISS, LEO, MEO, PO
- **0%Success Rate**: SO



# Flight Number vs. Orbit Type

**Exploratory Data Analysis**

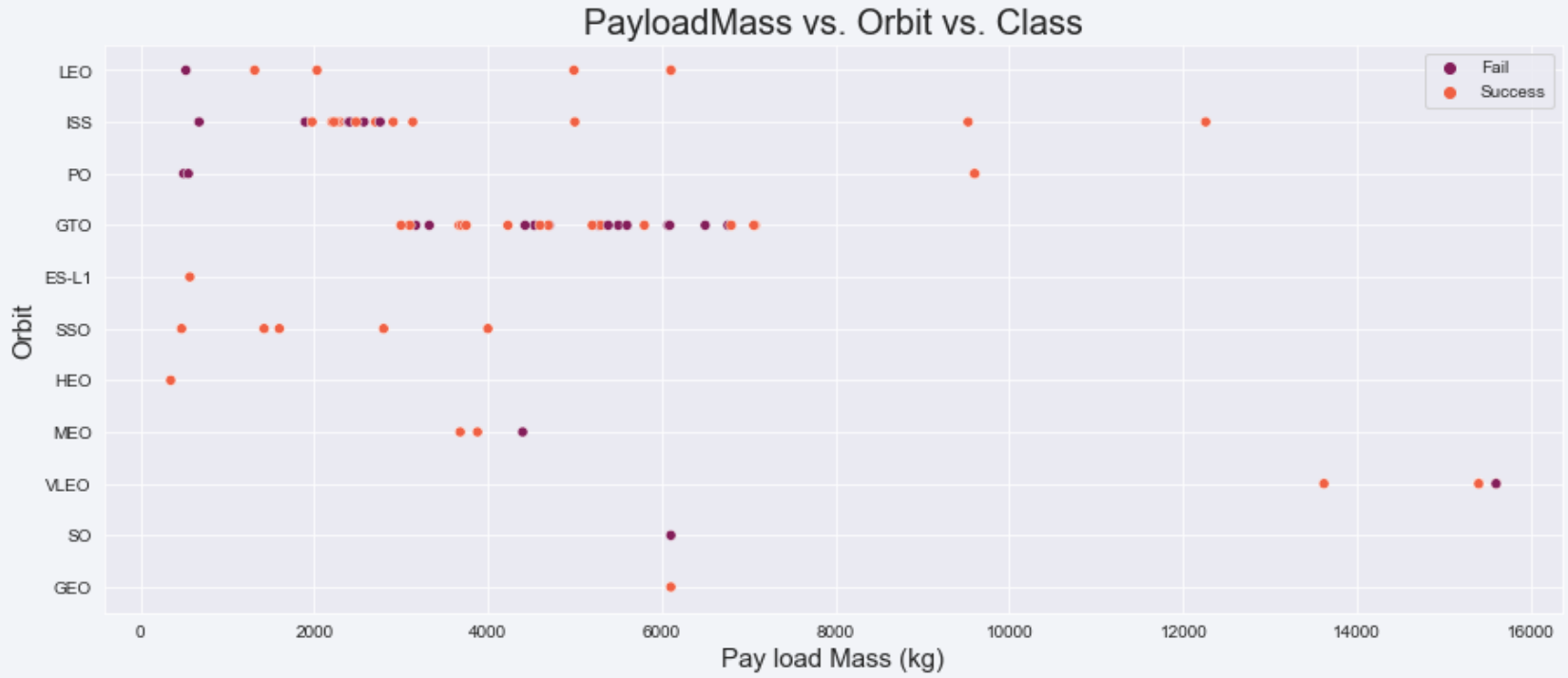
- The success rate typically increases with the number of flights for each orbit
- This relationship is highly apparent for the LEO orbit
- The GTO orbit, however, does not follow this trend



# Payload vs. Orbit Type

**Exploratory Data Analysis**

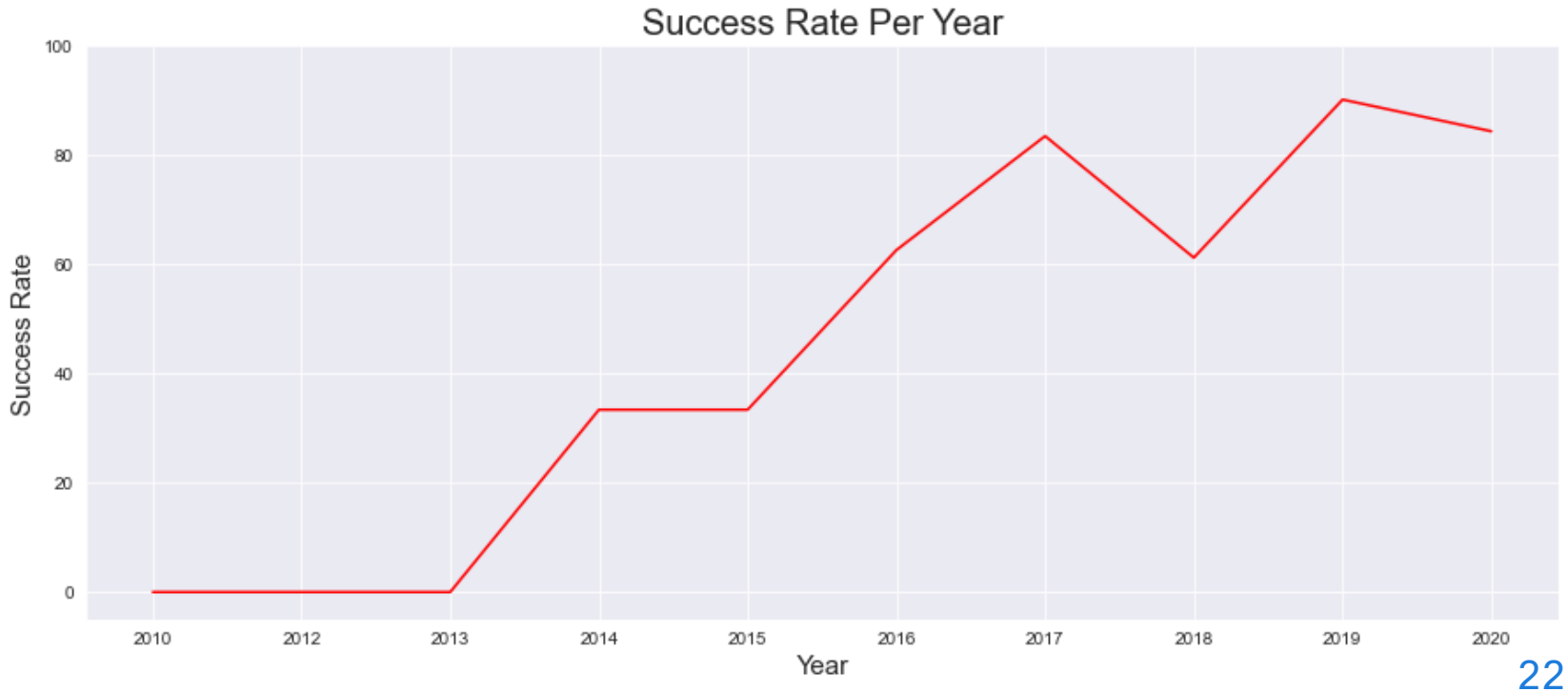
- Heavy payloads are better with LEO, ISS and PO orbits
- The GTO orbit has mixed success with heavier payloads



# Launch Success Yearly Trend

**Exploratory Data Analysis**

- The success rate improved from 2013-2017 and 2018-2019
- The success rate decreased from 2017-2018 and from 2019-2020
- Overall, the success rate has improved since 2013



# Launch Site Information

**Launch Site Names**

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

**Landing Outcome Cont.**

```
In [7]: %%sql
        SELECT DISTINCT Launch_Site
        FROM SPACEXTBL;
```

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

Out[7]:

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

**Records with Launch Site Starting with CCA**

- Displaying 5 records below

```
In [8]: %%sql
        SELECT *
        FROM SPACEXTBL
        WHERE Launch_Site like 'CCA%'
        LIMIT 5;
```

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

Out[8]:

| Date | Time (UTC) | Booster_Version | Launch_Site | Payload | PAYLOAD_MASS__KG_ | Orbit | Customer | Mission_Outcome | Landing_Outcome |
|---|---|---|---|---|---|---|---|---|---|
| 06/04/2010 | 18:45:00 | F9 v1.0 B0003 | CCAFS LC-40 | Dragon Spacecraft Qualification Unit | 0.0 | LEO | SpaceX | Success | Failure (parachute) |
| 12/08/2010 | 15:43:00 | F9 v1.0 B0004 | CCAFS LC-40 | Dragon demo flight C1, two CubeSats, barrel of Brouere cheese | 0.0 | LEO (ISS) | NASA (COTS) NRO | Success | Failure (parachute) |
| 22/05/2012 | 7:44:00 | F9 v1.0 B0005 | CCAFS LC-40 | Dragon demo flight C2 | 525.0 | LEO (ISS) | NASA (COTS) | Success | No attempt |
| 10/08/2012 | 0:35:00 | F9 v1.0 B0006 | CCAFS LC-40 | SpaceX CRS-1 | 500.0 | LEO (ISS) | NASA (CRS) | Success | No attempt |
| 03/01/2013 | 15:10:00 | F9 v1.0 B0007 | CCAFS LC-40 | SpaceX CRS-2 | 677.0 | LEO (ISS) | NASA (CRS) | Success | No attempt |

# Payload Mass

**Total Payload Mass**

- **45,596 kg** (total) carried by boosters launched by NASA (CRS)

```
In [9]: %%sql
        SELECT SUM(PAYLOAD_MASS__KG_) as total_mass
        FROM SPACEXTBL
        WHERE Customer = 'NASA (CRS)';
 * sqlite:///my_data1.db
Done.
```

Out[9]:

| total_mass |
|---|
| 45596.0 |

**Average Payload Mass**

- **2,928 kg** (average) carried by booster version F9 v1.1

```
In [10]: %%sql
         SELECT avg(PAYLOAD_MASS__KG_) as avg_mass
         FROM SPACEXTBL
         WHERE Booster_Version = 'F9 v1.1';
 * sqlite:///my_data1.db
Done.
```

Out[10]:

| avg_mass |
|---|
| 2928.4 |

# Landing & Mission Info

**1st Successful Landing in Ground Pad**

- 22/12/2015

In [26]:
```
%%sql
SELECT MAX(Date) as Date
FROM SPACEXTBL
WHERE Landing_Outcome = 'Success (ground pad)';
```

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

Out[26]:

| Date |
|---|
| 22/12/2015 |

**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

In [30]:
```
%%sql
SELECT DISTINCT Booster_Version
FROM SPACEXTBL
WHERE Landing_Outcome = 'Success (drone ship)'
AND (PAYLOAD_MASS__KG_ > 4000 AND PAYLOAD_MASS__KG_ < 6000);
```

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

Out[30]:

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

**Total Number of Successful and Failed Mission Outcomes**

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

In [31]:
```
%%sql
SELECT Mission_Outcome, COUNT(*) as Count
FROM SPACEXTBL
GROUP BY Mission_Outcome;
```

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

Out[31]:

| Mission_Outcome | Count |
|---|---|
| None | 898 |
| Failure (in flight) | 1 |
| Success | 98 |
| Success | 1 |
| Success (payload status unclear) | 1 |

# Boosters Carried Maximum Payload

- Determined the booster that have carried the maximum payload using a subquery in the **WHERE** clause and the **MAX()** function.

**Carrying Max Payload**

- 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

In [14]:
```
%%sql
SELECT Booster_Version
FROM SPACEXTBL
WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTBL);
```

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

Out[14]:

| 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 |

# Failed Landings on Drone Ship

**In 2015**

- Showing month, date, booster version, launch site and landing outcome

In [15]:
```
%%sql
SELECT substr(Date,4,2) as month, Date ,Booster_Version, Launch_Site, Landing_Outcome
FROM SPACEXTBL
WHERE Landing_Outcome = 'Failure (drone ship)' and substr(Date,7,4)='2015';
```

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

Out[15]:

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

# Count of Successful Landings

**Ranked Descending**

- Count of landing outcomes between 2010-06-04 and 2017-03-20 in descending order

```
In [16]: %%sql
SELECT Landing_Outcome, COUNT(*) AS QTY
FROM SPACEXTBL
WHERE Date BETWEEN '04-06-2010' AND '20-03-2017'
GROUP BY Landing_Outcome
ORDER BY QTY DESC;
```

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

Out[16]:

| Landing_Outcome | QTY |
|---|---|
| Success | 20 |
| No attempt | 10 |
| Success (drone ship) | 8 |
| Success (ground pad) | 7 |
| Failure (drone ship) | 3 |
| Failure | 3 |
| Failure (parachute) | 2 |
| Controlled (ocean) | 2 |
| No attempt | 1 |

- Selected Landing outcomes and the **COUNT** of landing outcomes from the data and used the **WHERE** clause to filter for landing outcomes and used **AND** to get the date between 2010-06-04 to 2010-03-20.
- Applied the **GROUP BY** clause to group the landing outcomes and the **ORDER BY** clause to order the grouped landing outcome in descending order.

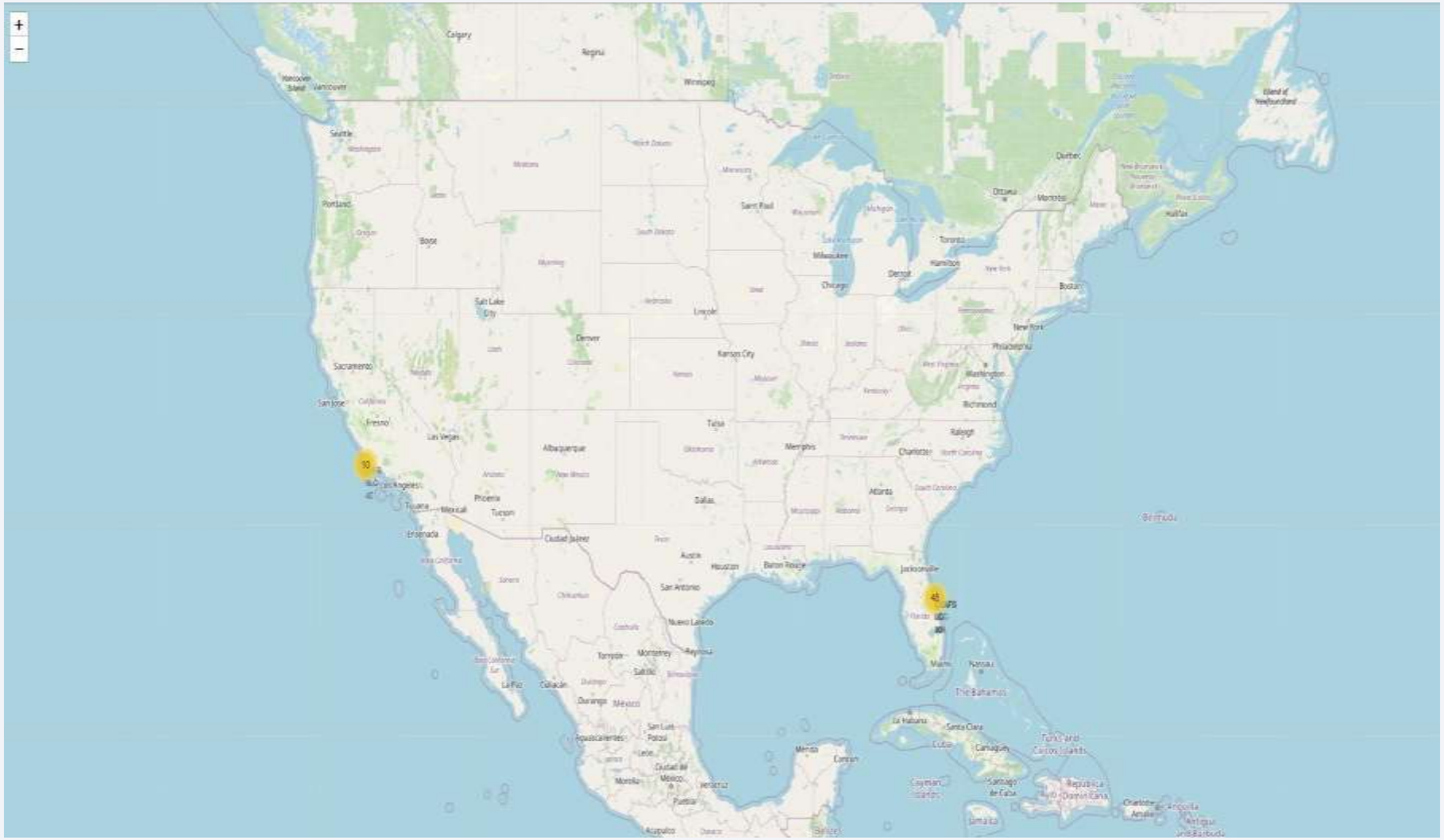
Section 4

# Launch Sites Proximities Analysis

# Launch Sites

**With Markers**

- **Near Equator**: the closer the launch site to the equator, the **easier** it is **to launch** to equatorial orbit, and the more help you get from Earth's rotation for a prograde orbit. Rockets launched from sites near the equator get an **additional natural boost** - due to the rotational speed of earth - that **helps save the cost** of putting in extra fuel and boosters.



# Launch Outcomes

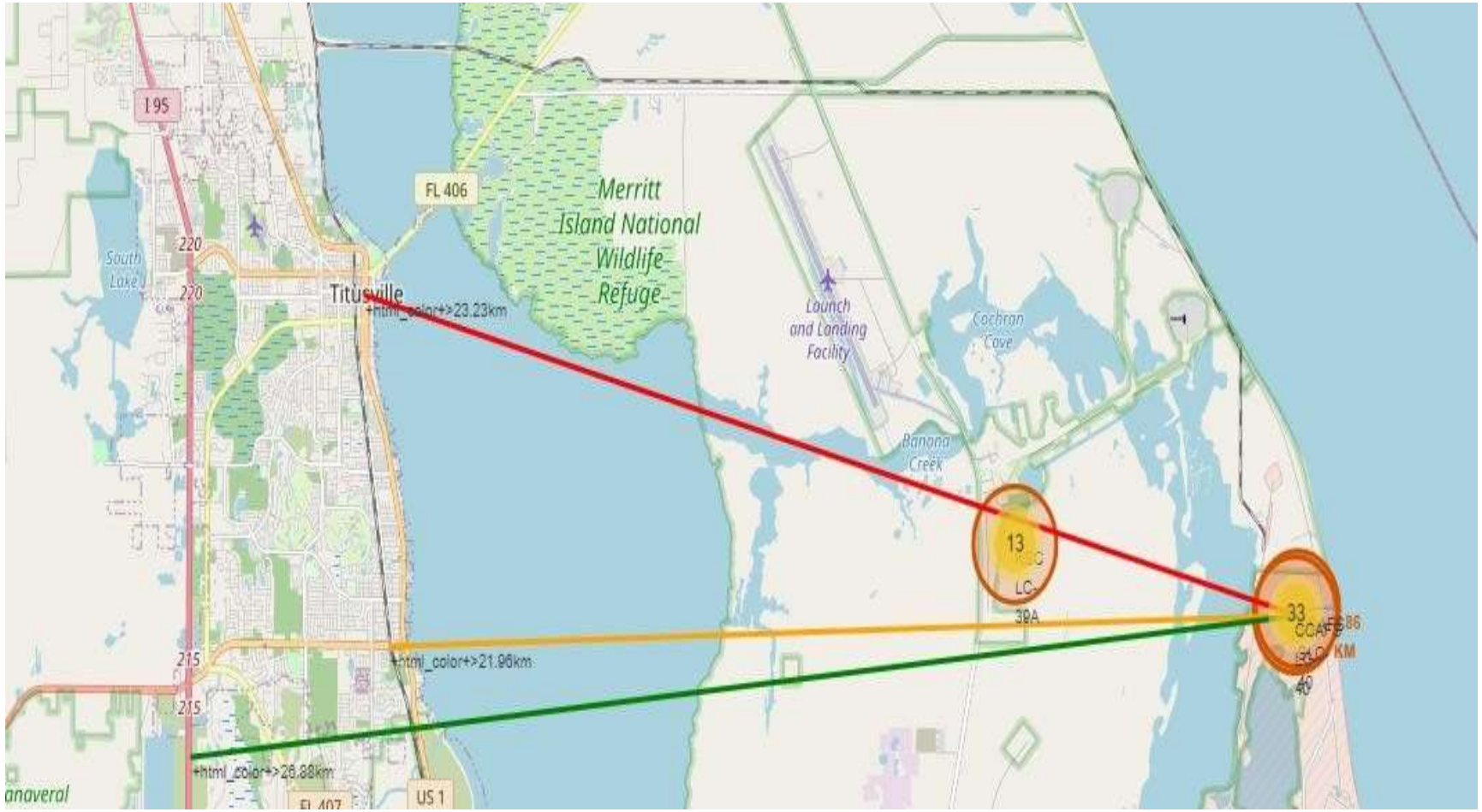
**At Each Launch Site**

- **Outcomes**:
- **Green** markers for successful launches
- **Red** markers for unsuccessful launches
- Launch site **CCAFS SLC-40** has a **3/7 success rate** (**42.9%**)

# Distance to Proximities

**CCAFS SLC-40**

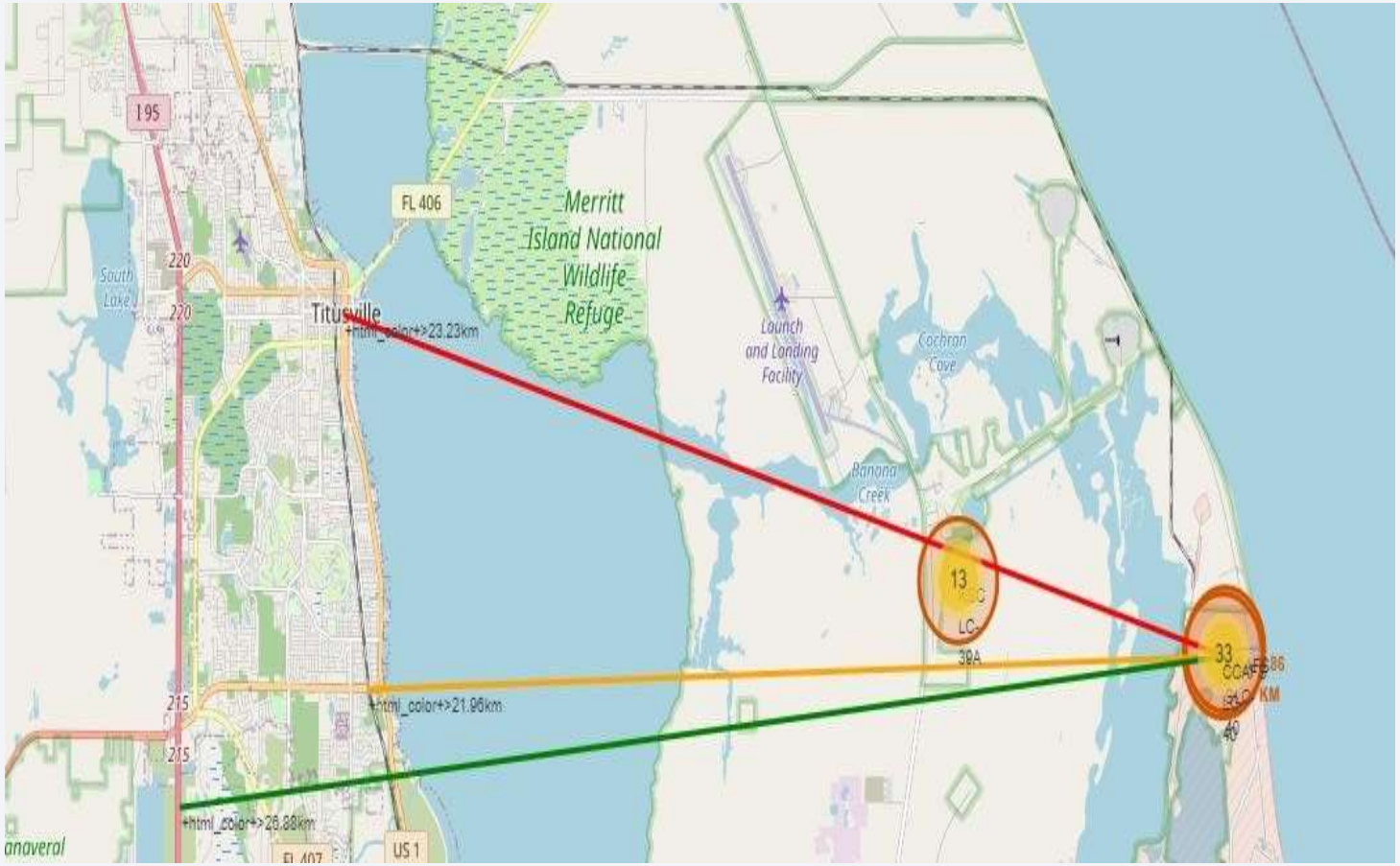
- **.86 km** from nearest coastline
- **21.96 km** from nearest railway
- **23.23 km** from nearest city
- **26.88 km** from nearest highway

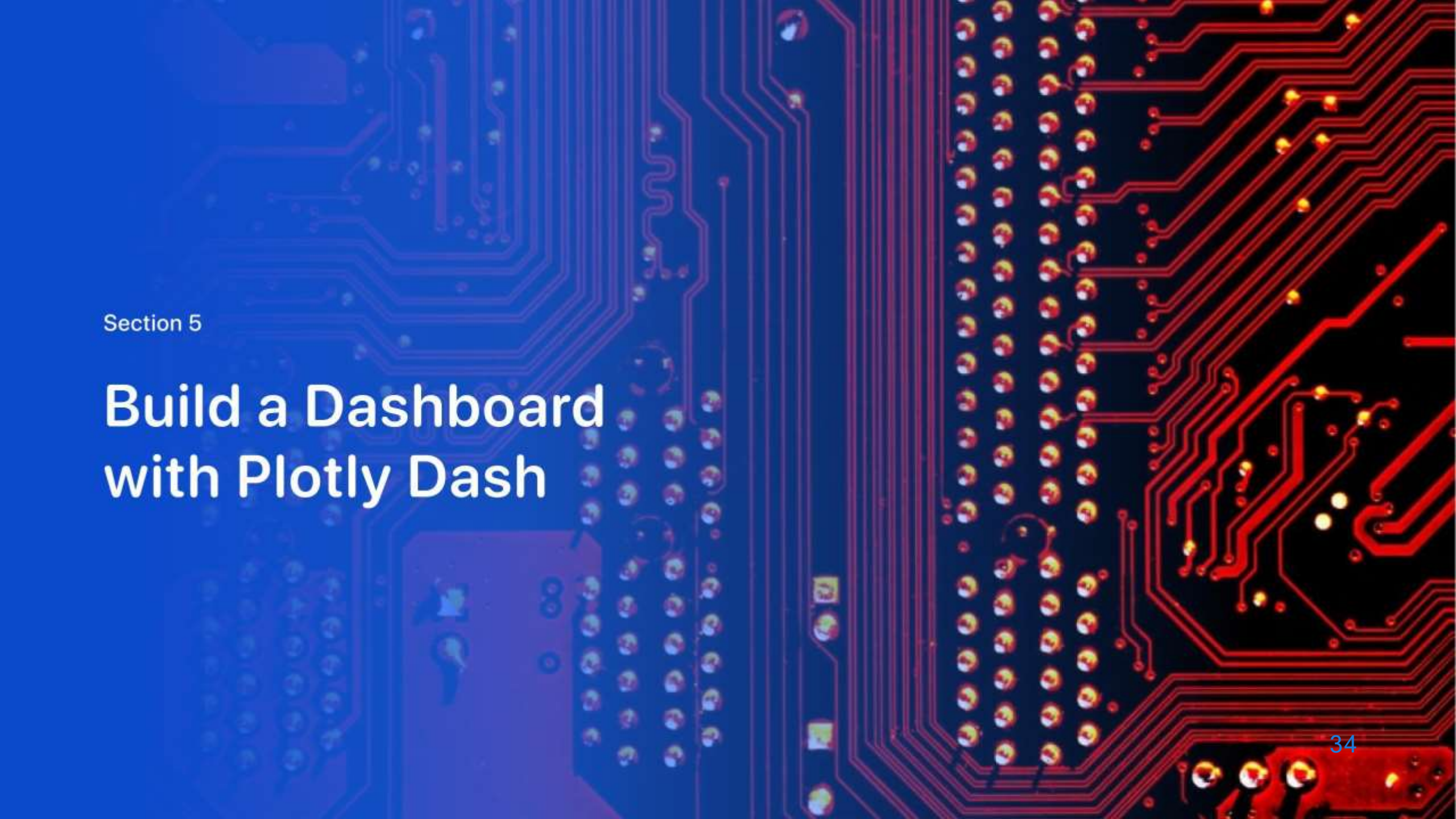


# Distance to Proximities

**CCAFS SLC-40**

- **Coasts**: help ensure that spent stages dropped along the launch path or failed launches don't fall on people or property.
- **Safety / Security:** needs to be an exclusion zone around the launch site to keep unauthorized people away and keep people safe.
- **Transportation/Infrastructure and Cities**: need to be away from anything a failed launch can damage, but still close enough to roads/rails/docks to be able to bring people and material to or from it in support of launch activities.

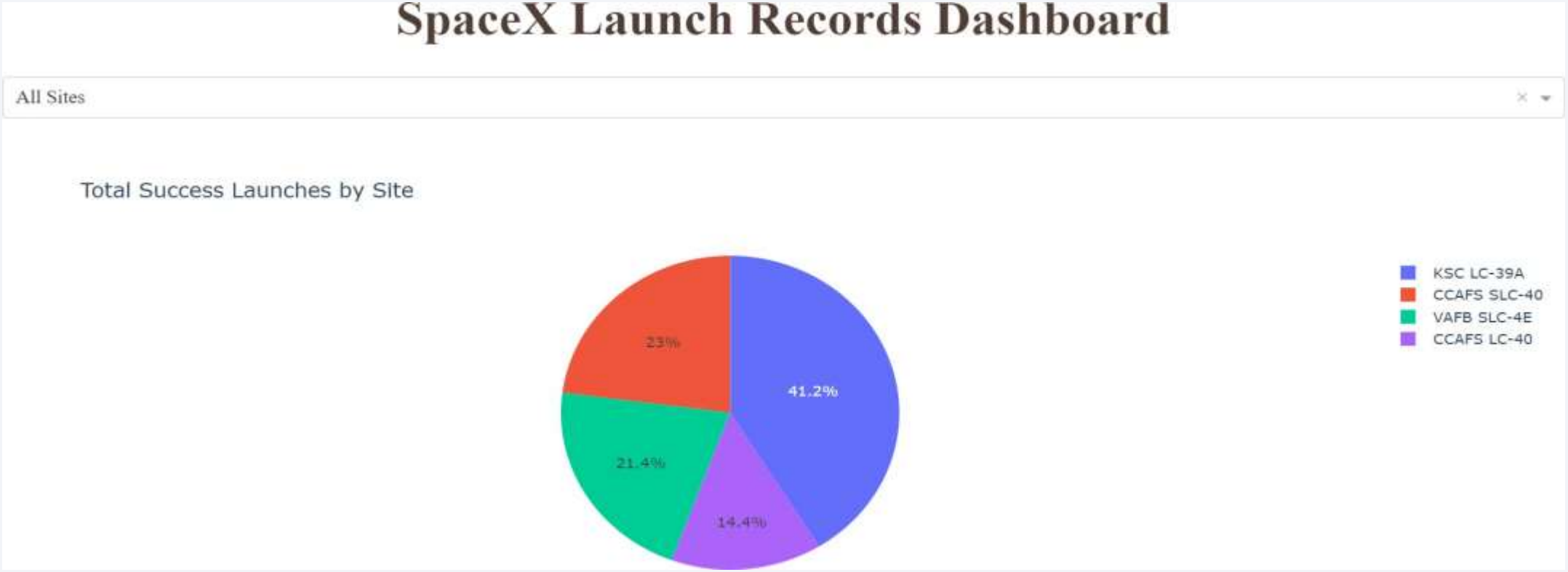



Section 5
Build a Dashboard with Plotly Dash

# Launch Success by Site

**Success as Percent of Total**

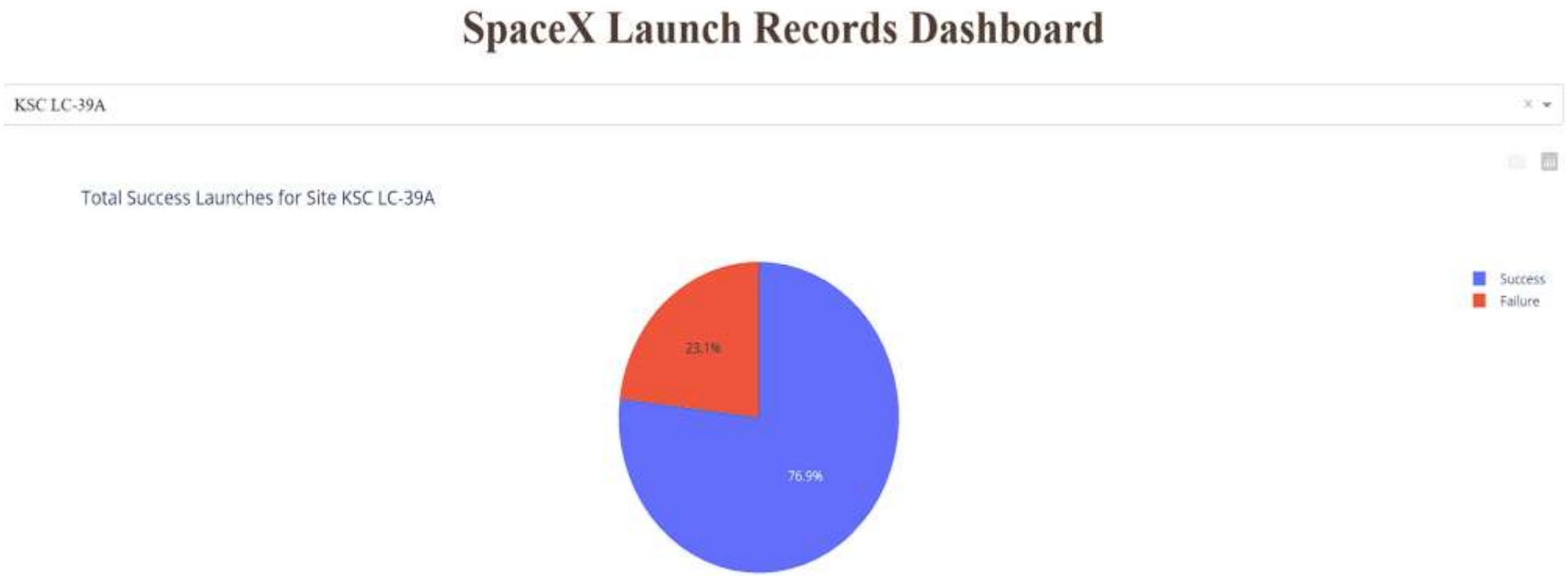
- **KSC LC-39A** has the **most successful launches** amongst launch sites (**41.2%**)



# Launch Success (KSC LC-29A)

**Success as Percent of Total**

- **KSC LC-39A** has the **highest success rate** amongst launch sites (**76.9%**)
- 10 successful launches and 3 failed launches



# Payload Mass and Success

**By Booster Version**

- **Payloads between 2,000 kg** and **5,000 kg** have the **highest success rate**
- 1 indicating successful outcome and 0 indicating an unsuccessful outcome

Section 6

# Predictive Analysis (Classification)

# 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_ on training data

In [27]:
```
def get_best_model(models):
    """Prints the best model"""
    best_model, best_score = None, 0

    for model in models:
        if model.best_score_ > best_score:
            best_score = model.best_score_
            best_model = model
    print(f"""The Best Model is "{type(best_model.estimator).__name__}" with accuracy = {best_model.best_score_ * 100:.2f}%  \n
            With the following parameters: {best_model.best_params_}""")

models = [logreg_cv, svm_cv, tree_cv, knn_cv]
get_best_model(models)
```

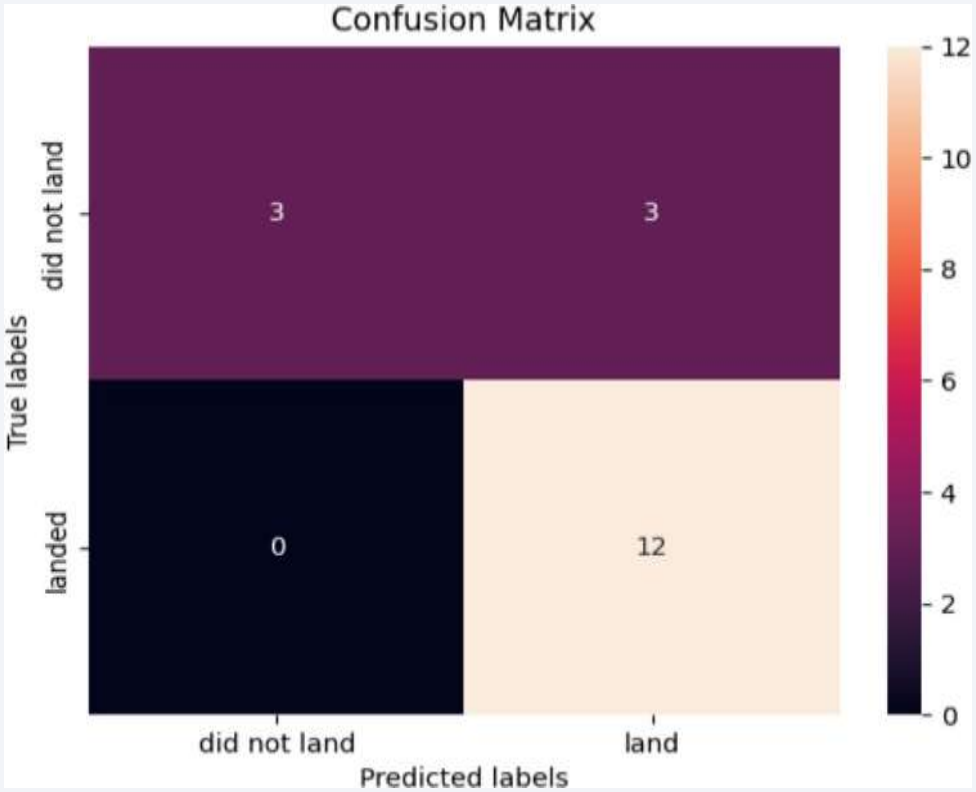
```
The Best Model is "DecisionTreeClassifier" with accuracy = 87.68%

            With the following parameters: {'criterion': 'gini', 'max_depth': 8, 'max_features': 'auto', 'min_samples_leaf': 2,
'min_samples_split': 10, 'splitter': 'best'}
```

# Confusion Matrices

**Performance Summary**

- A confusion matrix summarizes the performance of a classification algorithm
- All the confusion matrices were identical
- The fact that there are false positives (Type 1 error) is not good
- 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

**Research**

- **Model Performance**: The models performed similarly on the test set with the decision tree model slightly outperforming on training
- **Equator**: Most of the launch sites are near the equator for an additional natural boost - due to the rotational speed of earth - which helps save the cost of putting in extra fuel and boosters
- **Coast**: All the launch sites are close to the coast
- **Launch Success**: Increases over time
- **KSC LC-39A**: Has the highest success rate among launch sites. Has a 100% success rate for launches less than 5,500 kg
- **Orbits**: ES-L1, GEO, HEO, and SSO have a 100% success rate
- **Payload Mass**: Across all launch sites, the higher the payload mass (kg), the higher the success rate

**Things to Consider**

- **Dataset**: A larger dataset will help build on the predictive analytics results to help understand if the findings can be generalizable to a larger data set
- **Feature Analysis / PCA**: Additional feature analysis or principal component analysis should be conducted to see if it can help improve accuracy
- **XGBoost**: Is a powerful model which was not utilized in this study. It would
  be interesting to see if it outperforms the other classification models


Thank you!