IBM Developer
SKILLS NETWORK

# Winning Space Race with Data Science

João Batista de Camargo Junior
01/19/2022

# Outline

# 1. Executive Summary

The objective of this project is to build, tune and evaluate a ML model to predict if SpaceX first stage will land successfully. With this model, will be possible to determine if this first stage can be reused and provide information to determine the price of each launch to compete against SpaceX. To achieve this goal, initially data was collected using SpaceX REST API and web scraping related Wiki pages. Then data wrangling and Exploratory Data Analysis (EDA) was performed, using visualization (Pandas, Matplotlib, Folium and Plotly Dash) and SQL. The last methodological step was conduct predictive analysis using classification models, and to this end four classification models (Logistic Regression, SVM, Decision Tree and KNN) were built, tuned, and evaluated. The main result of this project shows that all models have the same accuracy, but Logistic Regression Model was elected for the sake of processing savings. Also, it was observed that the success rate improves when the amount of flight grows in a specific site, and less payload mass (below 4000 kg) has better success probability in all sites. The success rate since 2013 kept increasing till 2020, and site KSC LC-39A had more successful launches than other sites. CCAFS SLC-40 has the lowest rate of successful launches.

# 2. Introduction

- Companies are making space travel affordable for everyone, and the most successful of them is SpaceX.

- One reason SpaceX is successful is because their rocket launches are less expensive, since the first stage is reused. Therefore, if one can determine if the first stage will land, it is possible to determine the cost of a launch.

- Space Y is a new rocket company that would like to compete with SpaceX founded by Billionaire industrialist Allon Musk. In this project, the goal is to find answer to this question: will the SpaceX first stage land successfully and, therefore, can be reused?

- With this answer, will be possible to determine the price of each launch and compete against SpaceX. Data will be gathered from public sources to predict whether SpaceX will reuse the first stage.

# Methodology

# 3. Methodology

## Executive Summary

- Data collection methodology:
  - Data was collected using SpaceX REST API and web scraping related Wiki pages
- Perform data wrangling
  - Data was processed using an API, Sampling Data, and Dealing with Nulls
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Four classification models (Logistic Regression, SVM, Decision Tree and KNN) were built, tuned and evaluated

# 3.1 Data Collection

Data Collection

- First step of the project was Data Collection.

- Data was collected using SpaceX REST API. This API provided data about launches, including rocket used, payload delivered, launch specifications, landing specifications, and landing outcome.

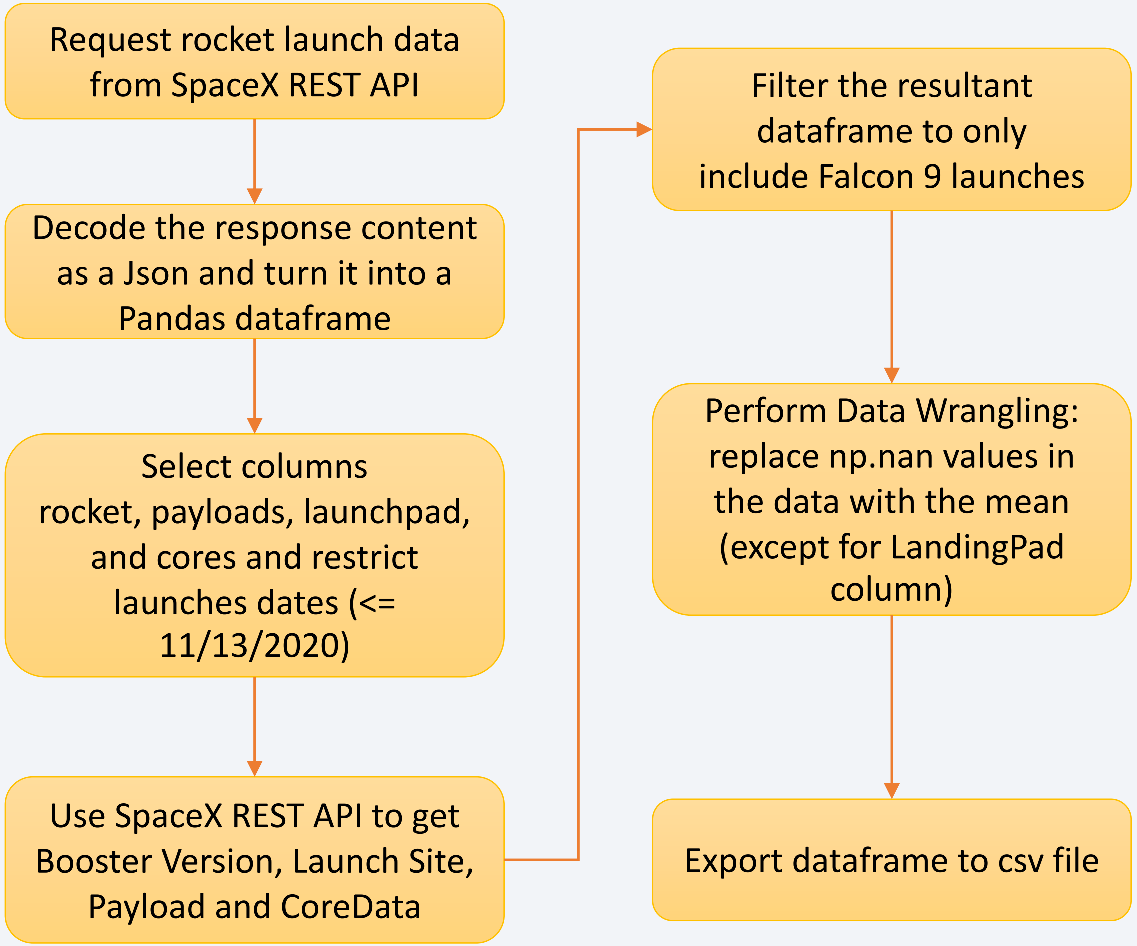
- Through web scrape in HTML pages, specifically using Python BeautifulSoup package, valuable Falcon 9 launch records were also obtained.

- All collected data was converted into Pandas data frames for further visualization and analysis.

# 3.1.1 Data Collection – SpaceX API

- Using SpaceX REST API, data was retrieved according flowchart
- Filters were applied and nulls were replaced to keep only useful data
- Final result was converted into csv file

- GitHub URL (https://github.com/jbcjr/IBM-Applied-Data-Science-Capstone/blob/main/1%20-%20Data%20Collection%20API%20Lab.ipynb)



# 3.1.1 Data Collection – SpaceX API

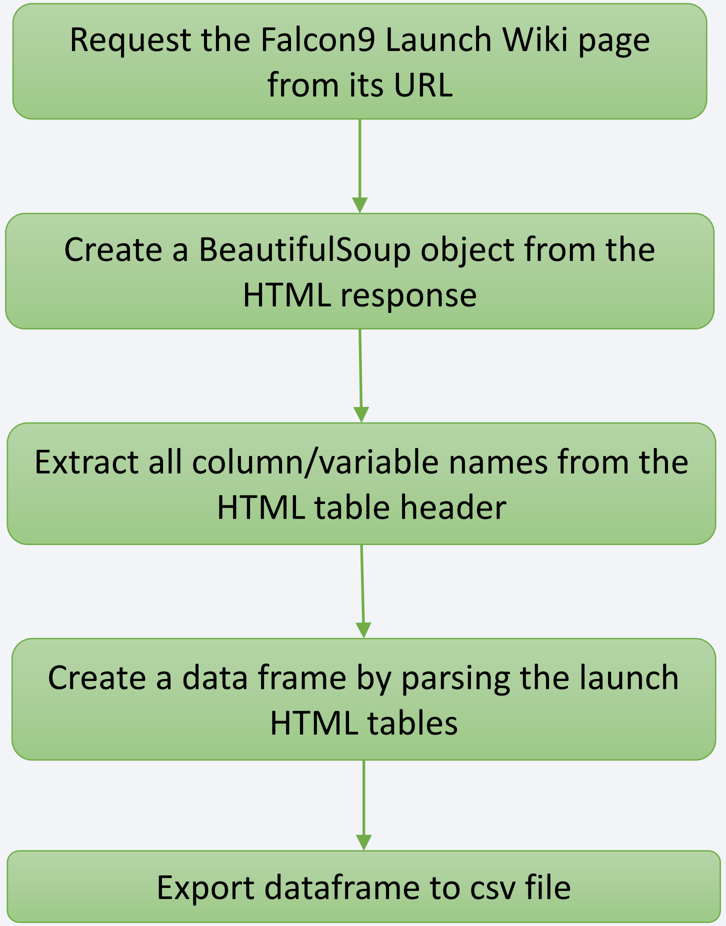
- Final csv file contains Flight Number, Date of flight, Booster Version (always Falcon 9), Payload Mass, Orbit, Launch Site, Outcome, quantity of Flights, whether gridfins were used, wheter legs were used, the landing pad used, the block of the core (number used to separate version of cores), number of times the specific core has been reused, serial of the core, Longitude and Latitude

| | FlightNumber | Date | BoosterVersion | PayloadMass | Orbit | LaunchSite | Outcome | Flights | GridFins | Reused | Legs | LandingPad | Block | ReusedCount | Serial | Longitude | Latitude |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 4 | 1 | 2010-06-04 | Falcon 9 | 6123.547647 | LEO | CCSFS SLC 40 | None None | 1 | False | False | False | None | 1.0 | 0 | B0003 | -80.577366 | 28.561857 |
| 5 | 2 | 2012-05-22 | Falcon 9 | 525.000000 | LEO | CCSFS SLC 40 | None None | 1 | False | False | False | None | 1.0 | 0 | B0005 | -80.577366 | 28.561857 |
| 6 | 3 | 2013-03-01 | Falcon 9 | 677.000000 | ISS | CCSFS SLC 40 | None None | 1 | False | False | False | None | 1.0 | 0 | B0007 | -80.577366 | 28.561857 |
| 7 | 4 | 2013-09-29 | Falcon 9 | 500.000000 | PO | VAFB SLC 4E | False Ocean | 1 | False | False | False | None | 1.0 | 0 | B1003 | -120.610829 | 34.632093 |
| 8 | 5 | 2013-12-03 | Falcon 9 | 3170.000000 | GTO | CCSFS SLC 40 | None None | 1 | False | False | False | None | 1.0 | 0 | B1004 | -80.577366 | 28.561857 |
| 9 | 6 | 2014-01-06 | Falcon 9 | 3325.000000 | GTO | CCSFS SLC 40 | None None | 1 | False | False | False | None | 1.0 | 0 | B1005 | -80.577366 | 28.561857 |
| 10 | 7 | 2014-04-18 | Falcon 9 | 2296.000000 | ISS | CCSFS SLC 40 | True Ocean | 1 | False | False | True | None | 1.0 | 0 | B1006 | -80.577366 | 28.561857 |
| 11 | 8 | 2014-07-14 | Falcon 9 | 1316.000000 | LEO | CCSFS SLC 40 | True Ocean | 1 | False | False | True | None | 1.0 | 0 | B1007 | -80.577366 | 28.561857 |
| 12 | 9 | 2014-08-05 | Falcon 9 | 4535.000000 | GTO | CCSFS SLC 40 | None None | 1 | False | False | False | None | 1.0 | 0 | B1008 | -80.577366 | 28.561857 |

# 3.1.2 Data Collection - Scraping

- Using web scrape, data was retrieved according flowchart
- All columns and variables names were extracted from the table header
- After parsing the launch tables, a dataframe was created and converted into csv file

- GitHub URL (https://github.com/jbcjr/IBM-Applied-Data-Science-Capstone/blob/main/2%20-%20Data%20Collection%20with%20Web%20Scraping.ipynb)

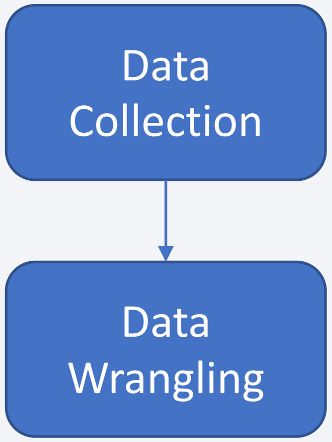


# 3.1.2 Data Collection - Scraping

- Final csv file contains Flight Number, Launch Site, Payload, Payload Mass, Orbit, Customer, Launch Outcome, Version Booster, Booster Landing, Date and Time

| | Flight No. | Launch site | Payload | Payload mass | Orbit | Customer | Launch outcome | Version Booster | Booster landing | Date | Time |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | CCAFS | Dragon Spacecraft Qualification Unit | 0 | LEO | SpaceX | Success | F9 v1.0B0003.1 | Failure | 4 June 2010 | 18:45 |
| 1 | 2 | CCAFS | Dragon | 0 | LEO | NASA | Success | F9 v1.0B0004.1 | Failure | 8 December 2010 | 15:43 |
| 2 | 3 | CCAFS | Dragon | 525 kg | LEO | NASA | Success | F9 v1.0B0005.1 | No attempt | 22 May 2012 | 07:44 |
| 3 | 4 | CCAFS | SpaceX CRS-1 | 4,700 kg | LEO | NASA | Success | F9 v1.0B0006.1 | No attempt | 8 October 2012 | 00:35 |
| 4 | 5 | CCAFS | SpaceX CRS-2 | 4,877 kg | LEO | NASA | Success | F9 v1.0B0007.1 | No attempt | 1 March 2013 | 15:10 |
| 5 | 6 | VAFB | CASSIOPE | 500 kg | Polar orbit | MDA | Success | F9 v1.1B1003 | Uncontrolled | 29 September 2013 | 16:00 |
| 6 | 7 | CCAFS | SES-8 | 3,170 kg | GTO | SES | Success | F9 v1.1 | No attempt | 3 December 2013 | 22:41 |
| 7 | 8 | CCAFS | Thaicom 6 | 3,325 kg | GTO | Thaicom | Success | F9 v1.1 | No attempt | 6 January 2014 | 22:06 |
| 8 | 9 | Cape Canaveral | SpaceX CRS-3 | 2,296 kg | LEO | NASA | Success | F9 v1.1 | Controlled | 18 April 2014 | 19:25 |

# 3.2 Data Wrangling

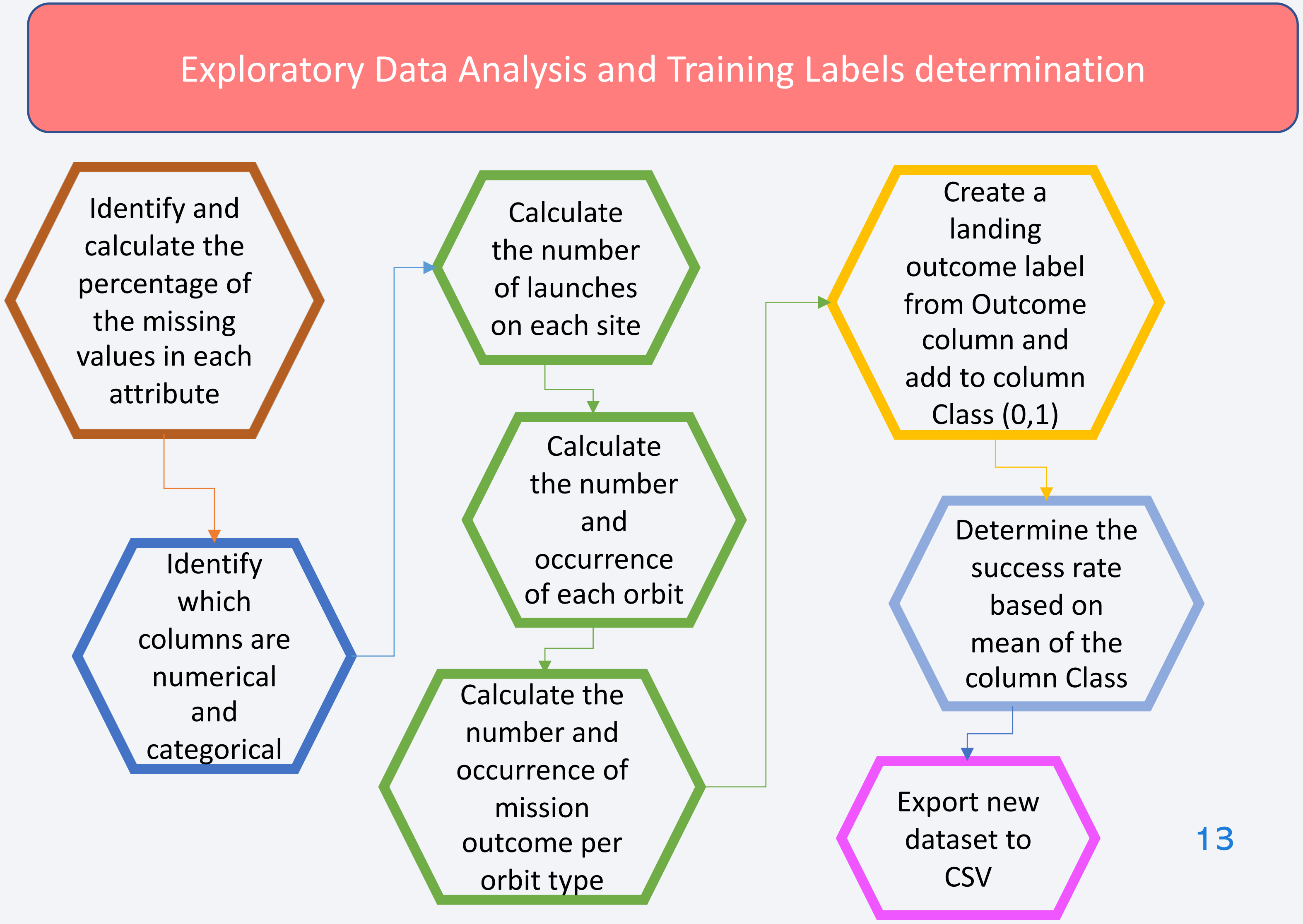



- Data Wrangling was the second step of the project. In the collected data, there are several different cases where the booster did not land successfully. For example:
  - True Ocean: the mission outcome was successfully landed to a specific region of the ocean;
  - False Ocean: the mission outcome was unsuccessfully landed to a specific region of the ocean;
  - True RTLS: the mission outcome was successfully landed to a ground pad;
  - False RTLS: the mission outcome was unsuccessfully landed to a ground pad;
  - True ASDS: the mission outcome was successfully landed on a drone ship;
  - False ASDS: the mission outcome was unsuccessfully landed on a drone ship.

- The goal here is to convert those outcomes into Training Labels with 1 means the booster successfully landed and 0 means it was unsuccessful.

# 3.2 Data Wrangling

- The process for Data Wrangling is described in the flowchart
- GitHub URL (https://github.com/jbcjr/IBM-Applied-Data-Science-Capstone/blob/main/3%20-%20Data%20Wrangling%20-%20EDA.ipynb)

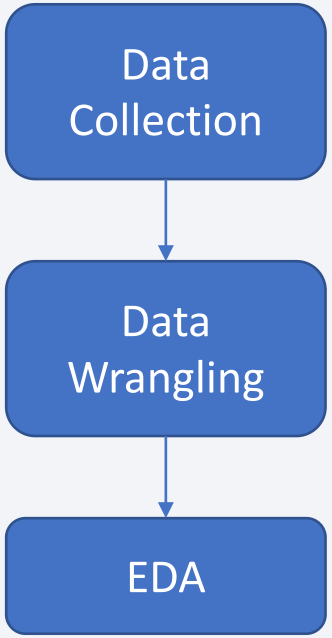


# 3.3 Exploratory Data Analysis (EDA)

- Third part of the project was a complete Exploratory Data Analysis using Pandas, Matplotlib and SQL Queries. For EDA, Folium Maps and a Dashboard with Plotly Dash were also built.

- Initially EDA was performed with Data Visualization. For this purpose, Scatter Graphs, Bar Graph and Line Graph were plotted.

- Scatter Graph was chosen to data visualization because it shows how much one variable affects another (correlation). The Bar Graph, in the other hand, facilitates to compare data from different groups. Finally, Line Graph was chosen to to demonstrate data variation over time.



# 3.3.1 EDA with Data Visualization

- Graphs plotted

**Scatter Graphs**

FlightNumber vs PayloadMass
FlightNumber vs LaunchSite
Payload vs Launch Site
FlightNumber vs Orbit type
Payload vs Orbit type

**Bar Graph**

Relationship between success rate of each orbit type

**Line Graph**

Launch success yearly trend

After visualize data, was possible to obtain some preliminary insights about how each important variable would affect the success rate. Therefore, features that will be used in success prediction in next steps of the project were selected and dummy variables were created to categorical columns.

GitHub URL (https://github.com/jbcjr/IBM-Applied-Data-Science-Capstone/blob/main/5%20-%20EDA%20using%20Pandas%20and%20Matplotlib.ipynb)

# 3.3.2 EDA with SQL

- EDA was also performed with SQL queries
- SQL queries performed were:
  - Display the names of the unique launch sites in the space mission
  - Display 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 acheived

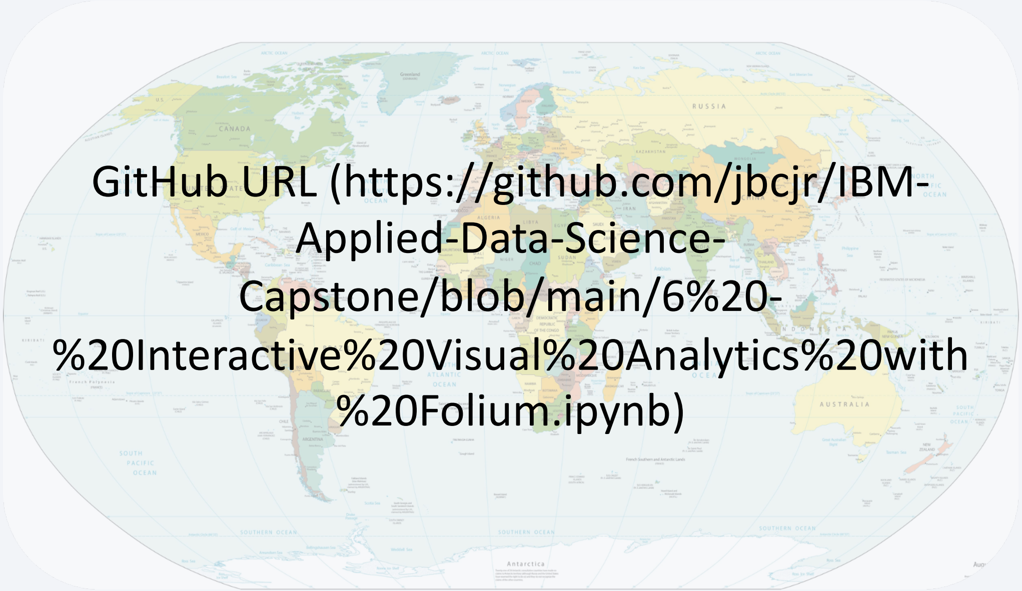
# 3.3.2 EDA with SQL

- Another SQL queries performed were:
  - List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000
  - List the total number of successful and failure mission outcomes
  - List the names of the booster versions which have carried the maximum payload mass using a subquery
  - List the failed landing outcomes in drone ship, their booster versions, and launch site names for year 2015
  - Rank the count of landing outcomes (such as Failure (drone ship) or Success (ground pad)) between the date 2010-06-04 and 2017-03-20, in descending order

GitHub URL (https://github.com/jbcjr/IBM-Applied-Data-Science-Capstone/blob/main/4%20-%20EDA%20with%20SQL.ipynb)

# 3.3.3 Build an Interactive Map with Folium

- Based on the Latitude and Longitude coordinates of each launch site, a Circle Marker and the name of the Launch site was added around each one in a Folium map.



GitHub URL (https://github.com/jbcjr/IBM-Applied-Data-Science-Capstone/blob/main/6%20-%20Interactive%20Visual%20Analytics%20with%20Folium.ipynb)

- Next, the success/failed launches for each site were marked on the map using a MarkerCluster and colors Green / Red based on the Class column.

- Finally, distances from each Launch Site to several Landmarks were calculated and noted with a Marker, a PolyLine and the respective label.

# 3.3.4 Build a Dashboard with Plotly Dash

- A Dashboard with Plotly Dash was created for better data exploration
- Graphs and interactions added:

**Pie Chart demonstrating Total Success Launches**

User can choose a specific Launch Site or all sites

**Scatter Graph demonstrating the correlation between Payload and Success**

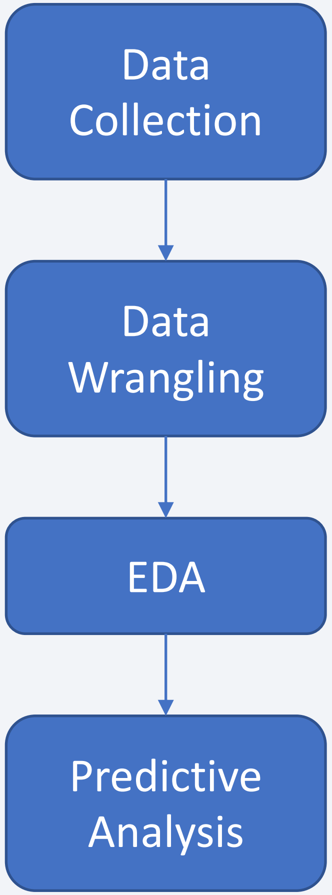
User can choose a specific Launch Site or all sites
User can choose the Payload range (between 0 and 10000)

GitHub URL (https://github.com/jbcjr/IBM-Applied-Data-Science-Capstone/blob/main/7%20-%20SpaceX%20Dashboard%20with%20Plotly%20Dash.py)

# 3.3.4 Build a Dashboard with Plotly Dash

- Pie Chart was chosen because it can easily demonstrate the success rate, allowing visual comparation
- Defining a specific site, it is possible to compare success rate proportionally
- Defining all sites, it is possible to compare success rate between Launch Sites
- Scatter Graph was chosen because it is the best visual form to represent correlations between two variables
- Defining a specific site or all sites, it is possible to verify the specific correlations, including Booster version
- Defining the payload range, it is possible to filter the data for specific payloads

# 3.4 Predictive Analysis (Classification)




Project last step was Predictive Analysis

Four classification models were built, evaluated and improved. After that, the best performing classification model was found

GitHub URL (https://github.com/jbcjr/IBM-Applied-Data-Science-Capstone/blob/main/8%20-%20Machine%20Learning%20Prediction.ipynb)

# 3.4 Predictive Analysis (Classification) process

Load data into Pandas dataframe

Create a NumPy array from the column Class in data

Standardize the data

Split data into training and test data sets

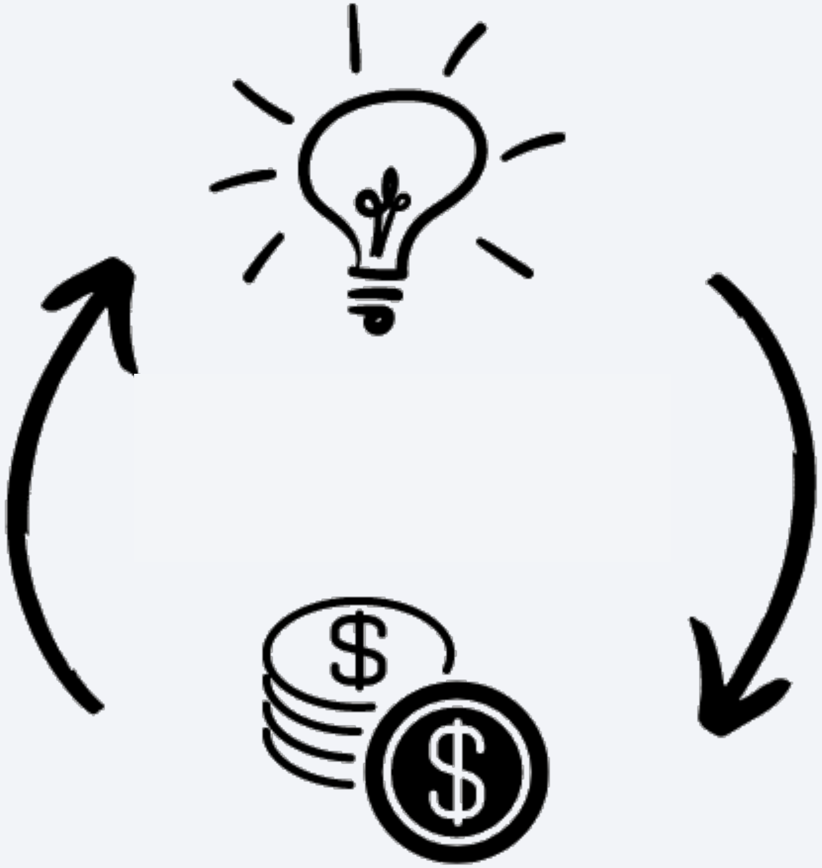
| | | | |
|---|---|---|---|
| Create a logistic regression object | Create a vector machine object | Create a tree classifier object | Create a k nearest neighbor object |
| Create a GridSearchCV object | Create a GridSearchCV object | Create a GridSearchCV object | Create a GridSearchCV object |
| Fit the object to find the best parameters | Fit the object to find the best parameters | Fit the object to find the best parameters | Fit the object to find the best parameters |
| Calculate the accuracy on test data | Calculate the accuracy on test data | Calculate the accuracy on test data | Calculate the accuracy on test data |
| Plot the Confusion Matrix | Plot the Confusion Matrix | Plot the Confusion Matrix | Plot the Confusion Matrix |

Determine which classification model performs best

# Results

# 4. Results

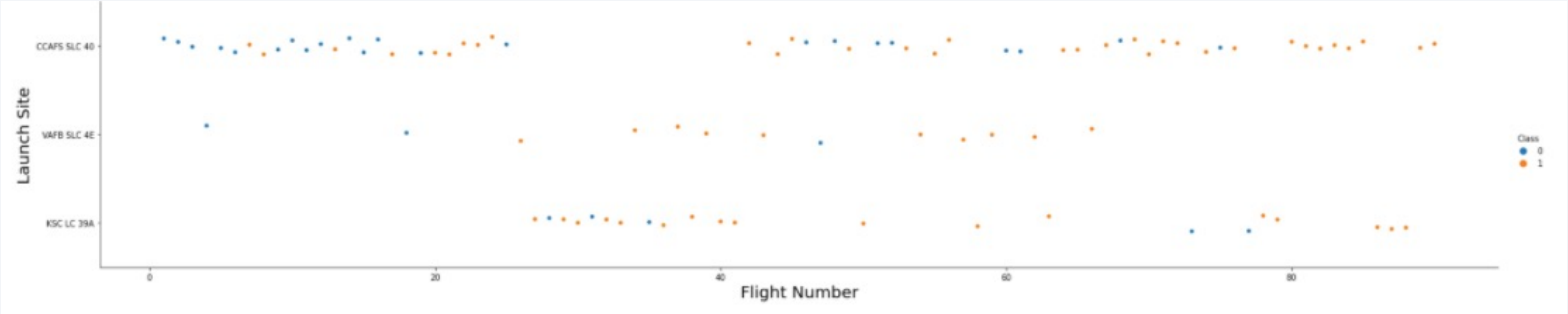
- This section will show the project results
- It is divided into:
  - 4.1 Insights drawn from EDA
  - 4.2 Launch Sites Proximities Analysis
  - 4.3 Build a Dashboard with Plotly Dash
  - 4.4 Predictive Analysis (Classification)



4.1

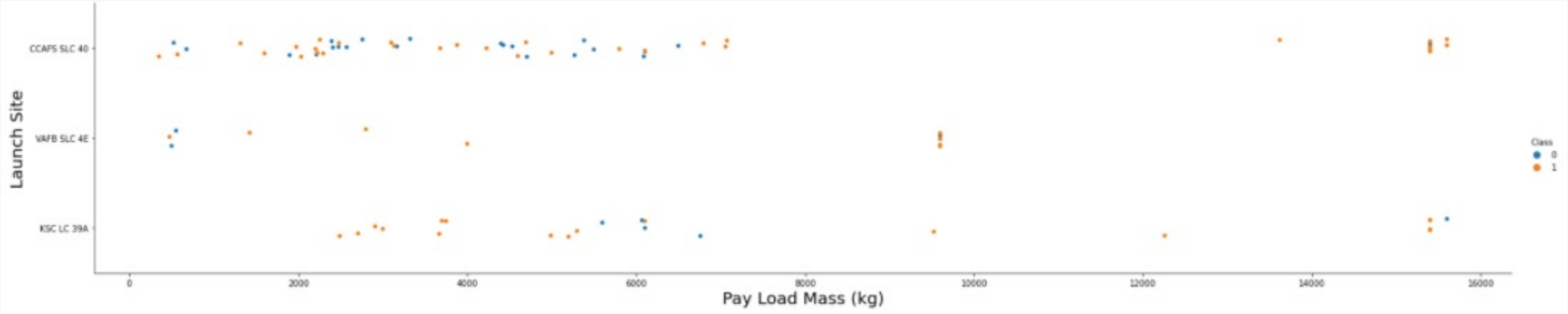
# Insights drawn from EDA

# Flight Number vs. Launch Site



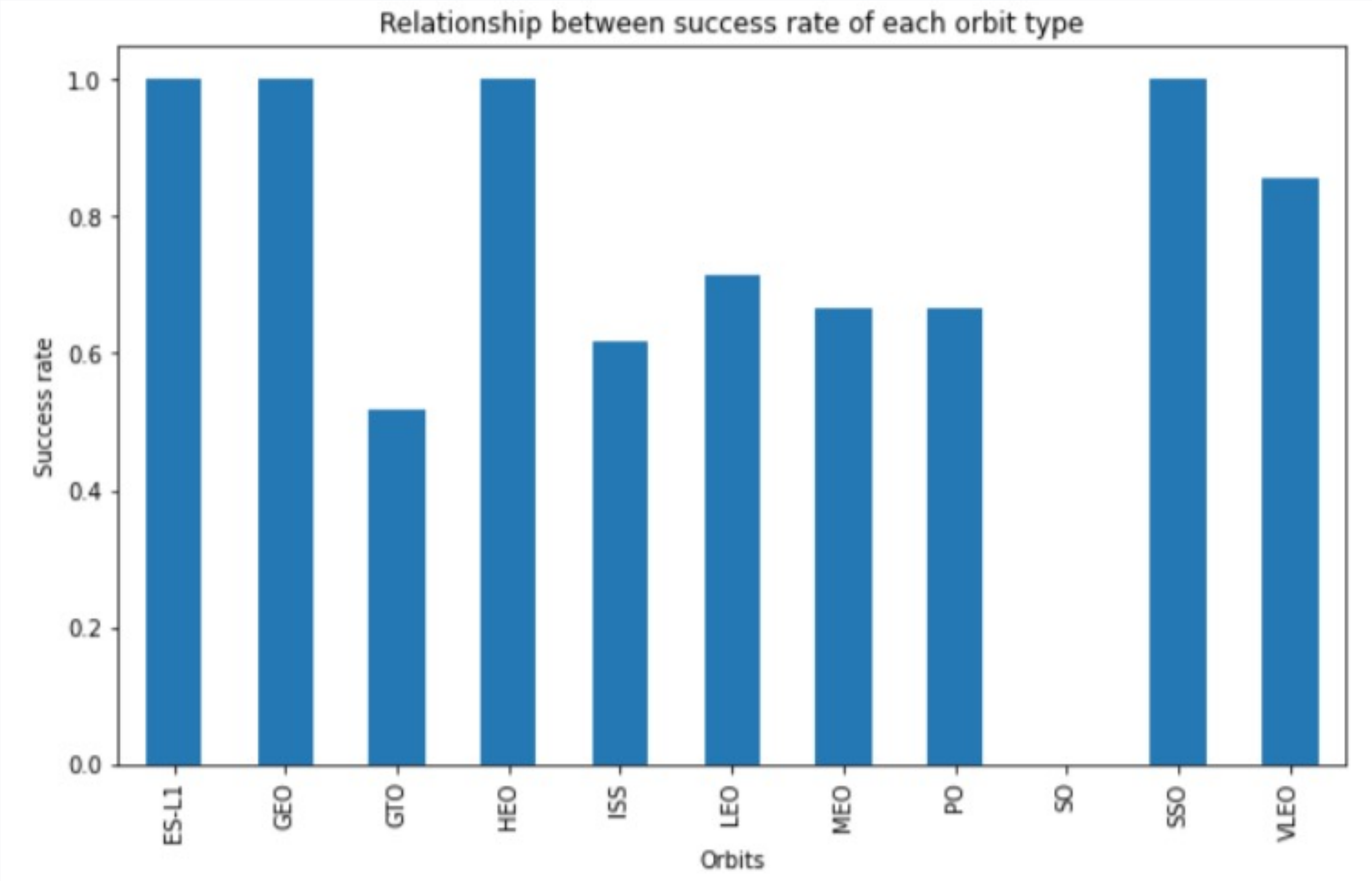

In this Scatter Graph, it can be perceived that the success rate improves when the amount of flight grows. Therefore, CCAFS SLC 40 success rate grow up with more Flight Numbers.

# Payload vs. Launch Site



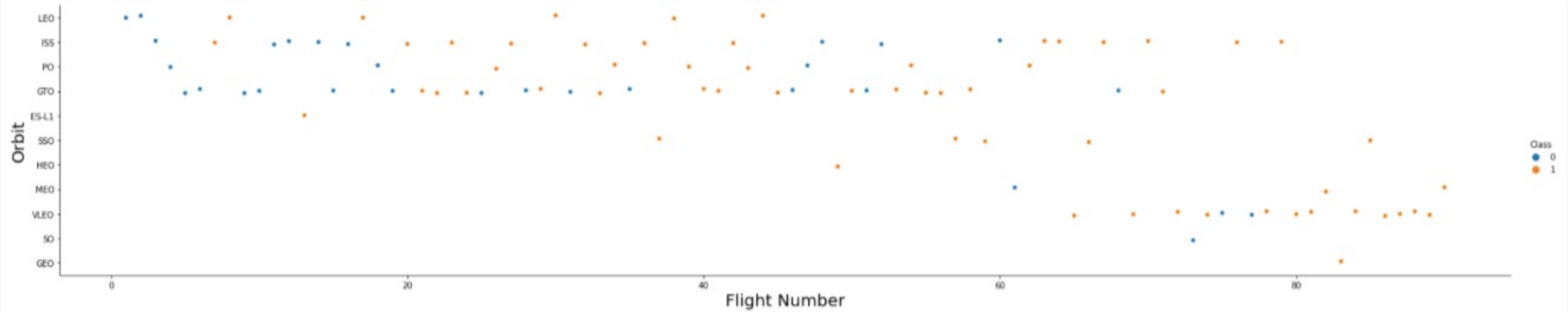

This Graph shows that less Payload Mass (below 4000 kg) has better success probability in all sites. Launch Site CCAFS SLC 40 has more success than other sites with Payload Mass above 13000 kg. For VAFB-SLC Launch Site there are no rockets launched for heavy payload mass (above 10000 kg).

# Success rate vs. Orbit Type



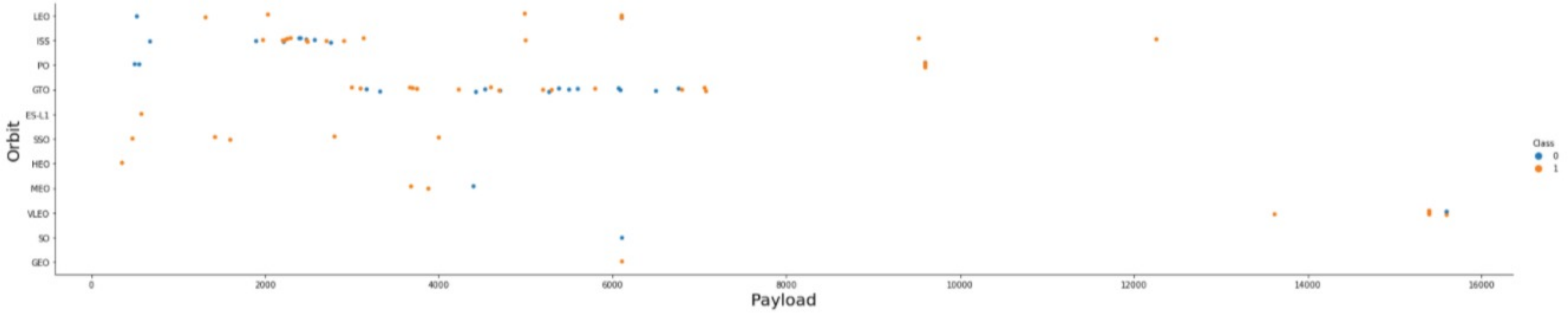

Analyzing the bar graph, it is identified that ES-L1, GEO, HEO and SSO orbits have the highest success rate. GTO has the lowest success rate, and SO has no launches.

# Flight Number vs. Orbit Type



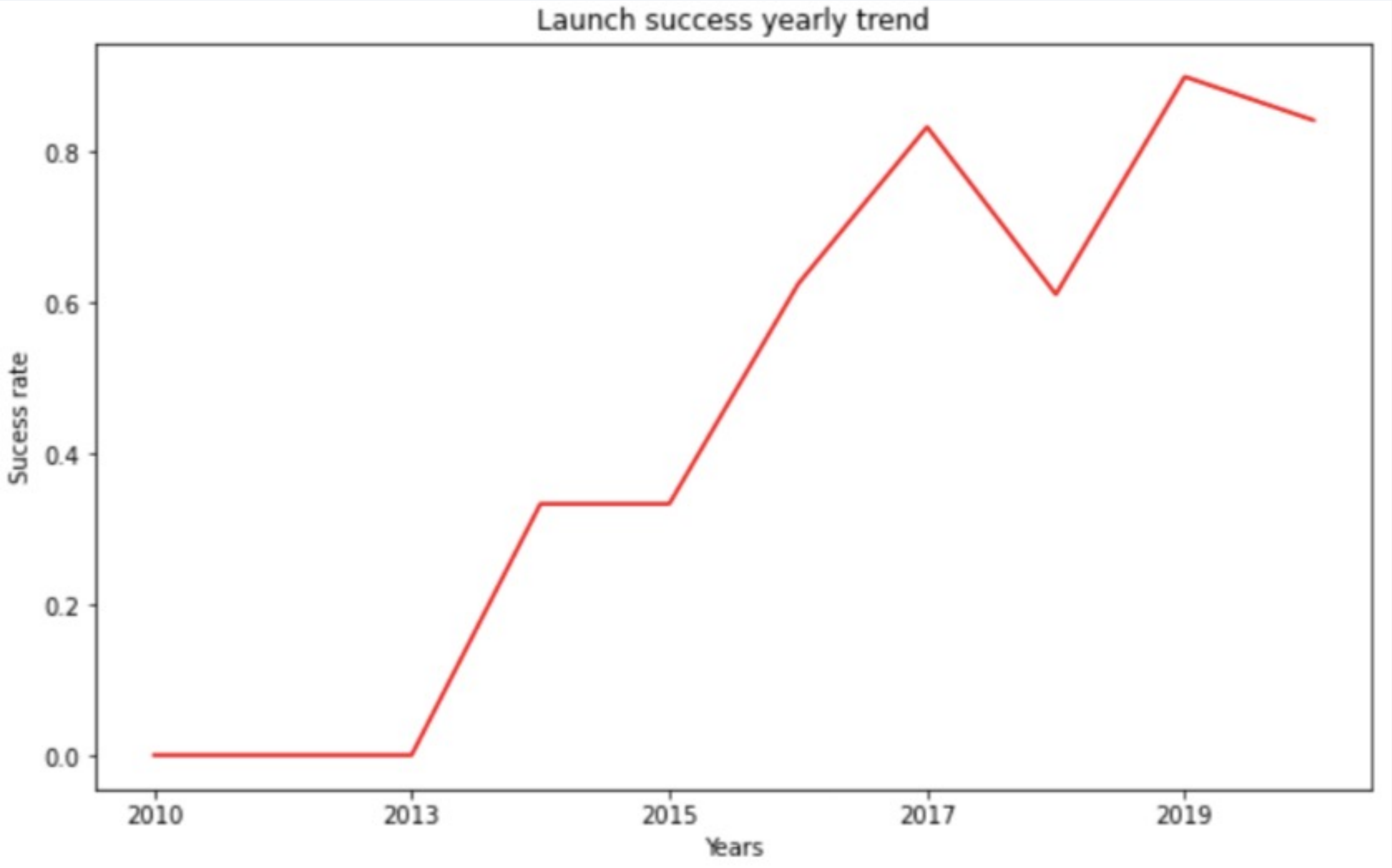

In the Graph is perceived that in the LEO orbit the success appears related to the number of flights. On the other hand, there seems to be no relationship between Flight Number when in GTO orbit. VLEO has more success rate with greater Flight Numbers.

# Payload vs. Orbit Type




This Scatter Graph shows that, with heavy payloads, the successful landing or positive landing rate are higher for Polar, LEO and ISS orbits. With light payloads, the success rate is better for ES-L1 and SSO. However, is not possible to determine a pattern for GTO, since both positive landing rate and negative landing happens independent of payload.

# Launch Success Yearly Trend




The Line Graph shows that the success rate since 2013 kept increasing till 2020, with a slight decrease in 2018

# All Launch Site Names

Query

```
Select UNIQUE(launch_site) from SPACEXDATASET
```

Result

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

Explanation

The word UNIQUE in the query allows to retrieve Launch Sites names without repetition, i.e., the query will retrieve unique values from column launch_site.

# Launch Site Names Begin with 'CCA'

## Query

```
Select * from SPACEXDATASET
    where launch_site like 'CCA%'
    limit 5
```

## Explanation

To show records where launch sites begin with ‘CCA’ was used ‘like 'CCA%’. Using ‘limit 5’, only five records are selected.

## Result

| 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 | 07:44:00 | F9 v1.0 B0005 | CCAFS LC-40 | Dragon demo flight C2 | 525 | LEO (ISS) | NASA (COTS) | Success | No attempt |
| 2012-10-08 | 00:35:00 | 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

Query

```
Select SUM(payload_mass__kg_) as TOTAL_PAYLOAD from SPACEXDATASET
    where customer like '%NASA (CRS)%'
```

Result

| total_payload |
| --- |
| 48213 |

Explanation

Function SUM calculates the total amount in column payload_mass__kg_. Using ‘like '%NASA (CRS)%’, this sum is made only for customer NASA (CRS), i.e., whenever in that column it has the name NASA (CRS).

# Average Payload Mass by F9 v1.1

Query

```
Select AVG(payload_mass__kg_) as AVG_PAYLOAD from SPACEXDATASET
    where booster_version = 'F9 v1.1'
```

Result

| avg_payload |
| --- |
| 2928 |

Explanation

Function AVG calculates the average value in column payload_mass__kg_. The where clause selects only the average payload mass carried by booster version F9 v1.1.

# First Successful Ground Landing Date

Query

```
Select MIN(DATE) as DATE from SPACEXDATASET
    where landing__outcome = 'Success (ground pad)'
```

Result

| DATE |
| --- |
| 2015-12-22 |

Explanation

The MIN(DATE) function allows to select the minimal date from column DATE, while where clause filter the result for the first successful landing outcome on ground pad.

# Successful Drone Ship Landing with Payload between 4000 and 6000

Query

```
Select UNIQUE(booster_version) as BOOSTERS from SPACEXDATASET
    where landing__outcome = 'Success (drone ship)' and
    payload_mass__kg_ between 4000 and 6000
```

Result

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

Explanation

Using UNIQUE word, booster versions are selected without repetition. The where clause filters the results to only boosters that successfully landed on drone ship and had payload mass greater than 4000 but less than 6000.

# Total Number of Successful and Failure Mission Outcomes

Query

```
Select * from
(Select COUNT(mission_outcome) as Success from SPACEXDATASET where mission_outcome like '%Success%'),
(Select COUNT(mission_outcome) as Failure from SPACEXDATASET where mission_outcome like '%Failure%')
```

Result

| success | failure |
|---|---|
| 100 | 1 |

Explanation

Using subqueries, initially success mission outcomes were calculated, counting all records with the Success notation. Then failure outcomes were calculated counting all records with the Failure notation.

# Boosters Carried Maximum Payload

Query

```
Select booster_version
    from SPACEXDATASET
    where payload_mass__kg_ =
        (Select MAX(payload_mass__kg_)
            from SPACEXDATASET)
```

Explanation

Using a subquery, the maximum value for payload_mas__kg was defined. Then, all booster version with this payload were retrieved with the where clause.

Result

| booster_version |
|---|
| F9 B5 B1048.4 |
| F9 B5 B1048.5 |
| F9 B5 B1049.4 |
| F9 B5 B1049.5 |
| F9 B5 B1049.7 |
| F9 B5 B1051.3 |
| F9 B5 B1051.4 |
| F9 B5 B1051.6 |
| F9 B5 B1056.4 |
| F9 B5 B1058.3 |
| F9 B5 B1060.2 |
| F9 B5 B1060.3 |

# 2015 Launch Records

Query

```
Select booster_version, launch_site, landing__outcome from SPACEXDATASET
where landing__outcome = 'Failure (drone ship)' and
      YEAR(DATE) = 2015
```

Result

| booster_version | launch_site | landing__outcome |
|---|---|---|
| F9 v1.1 B1012 | CCAFS LC-40 | Failure (drone ship) |
| F9 v1.1 B1015 | CCAFS LC-40 | Failure (drone ship) |

Explanation

The where clause allows to retrieve failed landing_outcomes in drone ship for year 2015. Their respective booster versions and launch site names were also retrieved.

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

Query

```
Select landing__outcome, COUNT(landing__outcome)
as COUNT from SPACEXDATASET
where DATE between '2010-06-04' and '2017-03-20'
group by landing__outcome
ORDER BY COUNT(landing__outcome) DESC
```

Explanation

COUNT function counts how many landing outcomes it has for each category. Where clause filters date between 2010-06-04 and 2017-03-20. The GROUP BY function groups the results, and 'ORDER BY COUNT(landing__outcome) DESC' allows to shows results in descending order.

Result

| landing__outcome | COUNT |
|---|---|
| No attempt | 10 |
| Failure (drone ship) | 5 |
| Success (drone ship) | 5 |
| Controlled (ocean) | 3 |
| Success (ground pad) | 3 |
| Failure (parachute) | 2 |
| Uncontrolled (ocean) | 2 |
| Precluded (drone ship) | 1 |

4.2

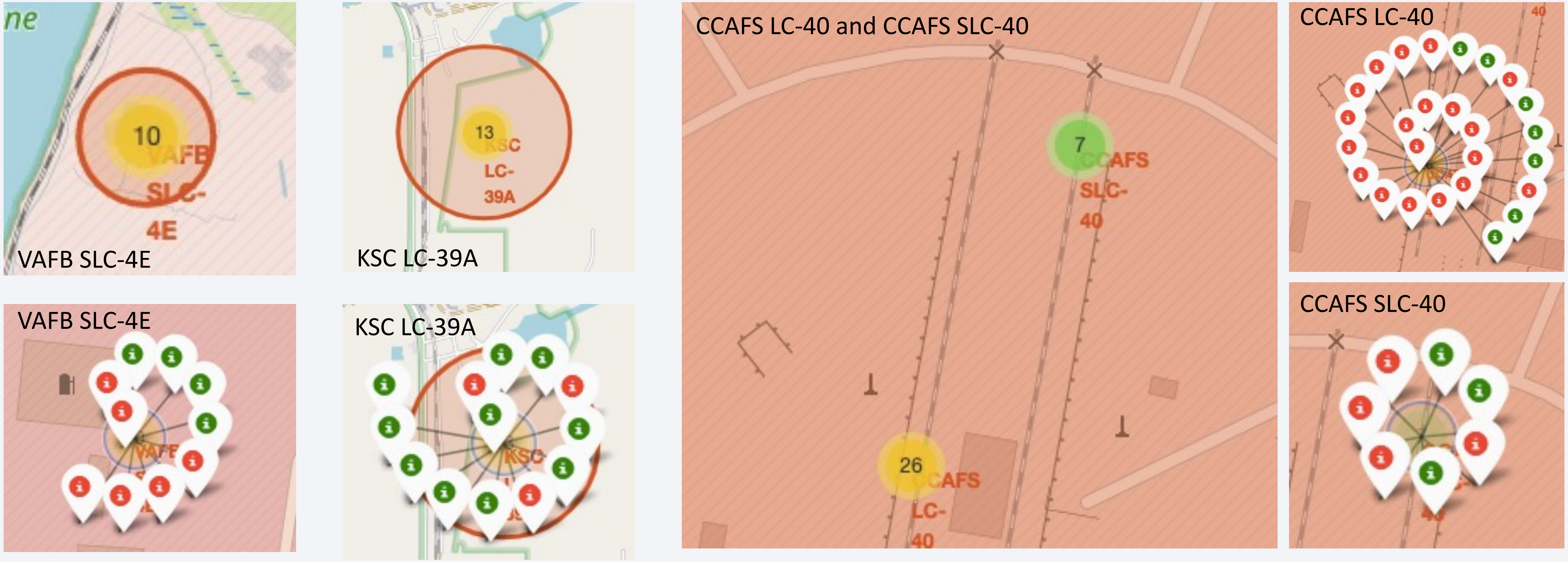
# Launch Sites Proximities Analysis

# All Launch Sites markers


This map indicates that all Launch Sites are in US coasts (Florida and California)

The Equator line is depicted in blue, showing that VAFB SLC-4E is the more distant site from the Equator line

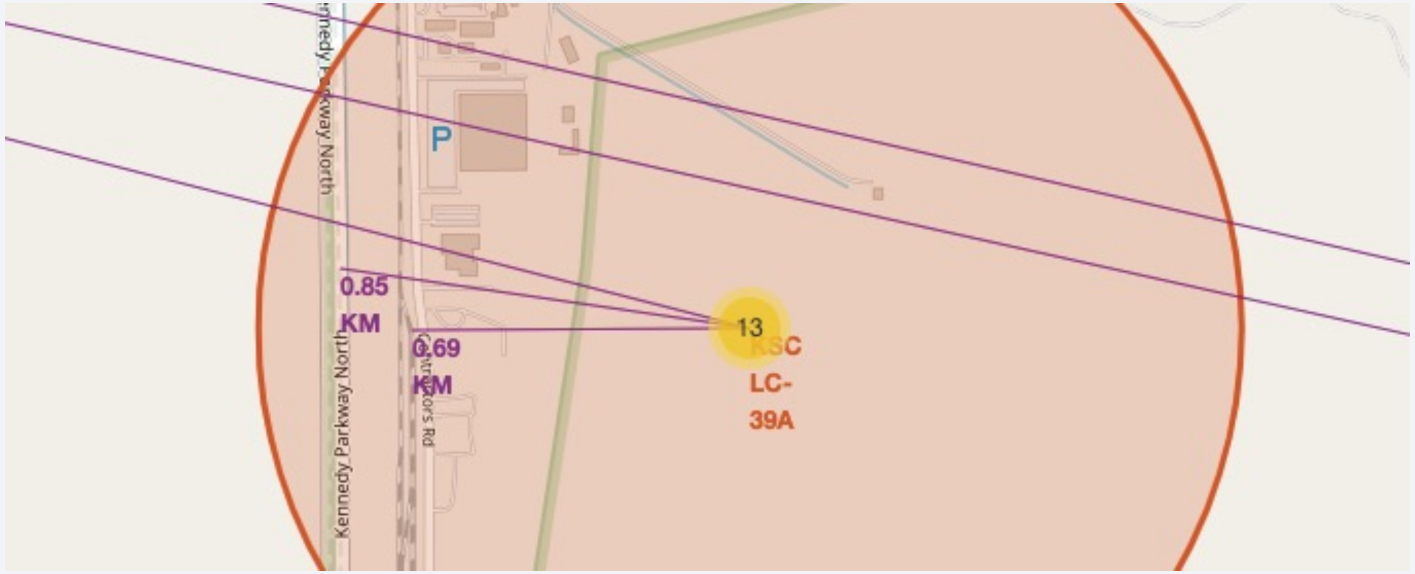
# Successful and Failure launches by site

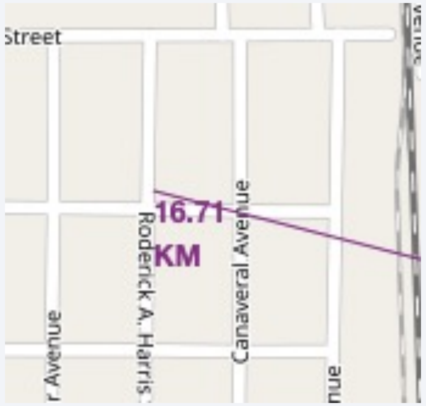



The screenshots demonstrate the launches quantity by site and successful and failure launches. Green Marker shows successful launches and Red Marker shows Failures. It is possible to observe that KSC LC-39A had more successful launches than other sites.

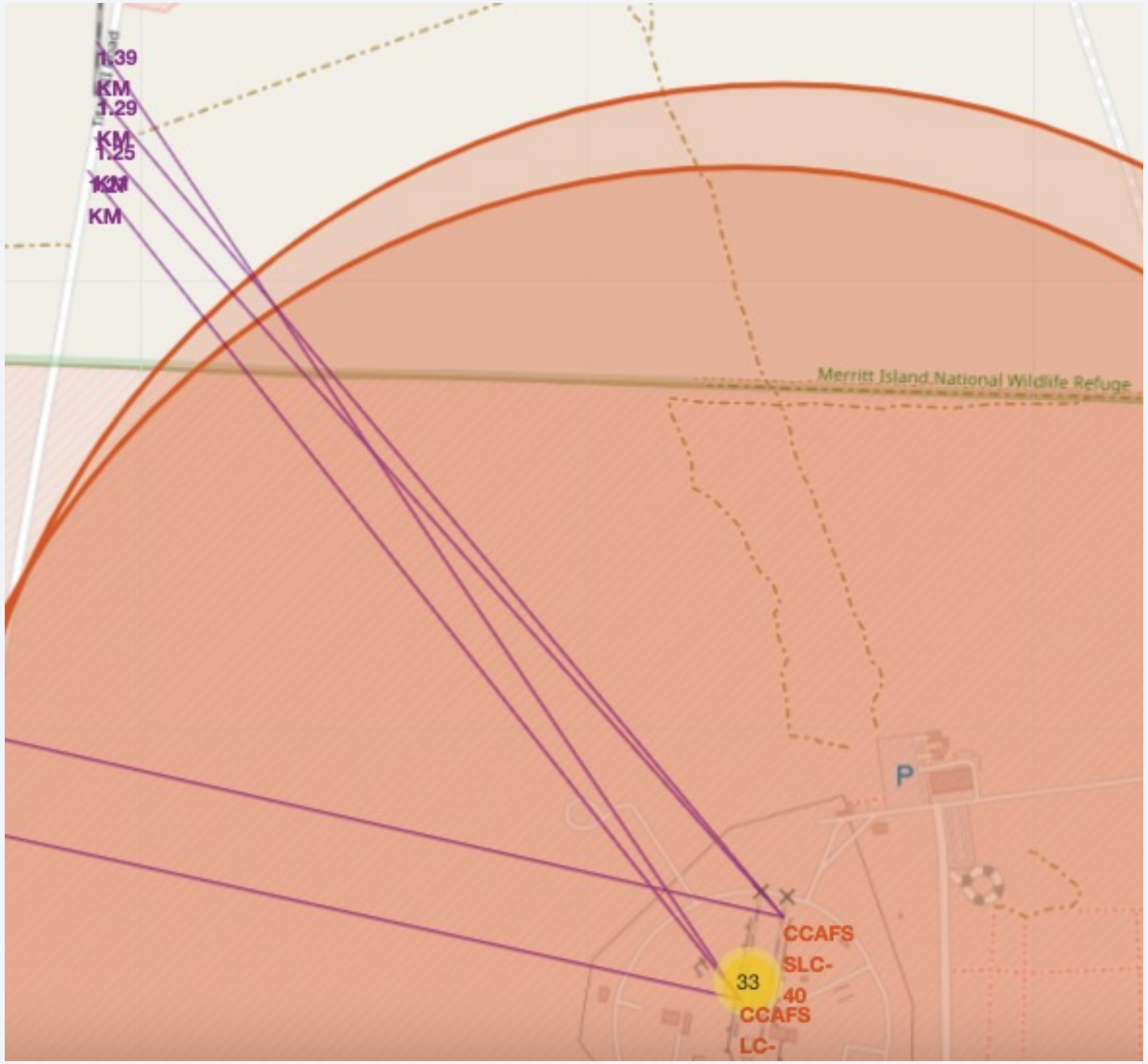
# VAFB SLC-4E and KSC LC-39A distances to landmarks

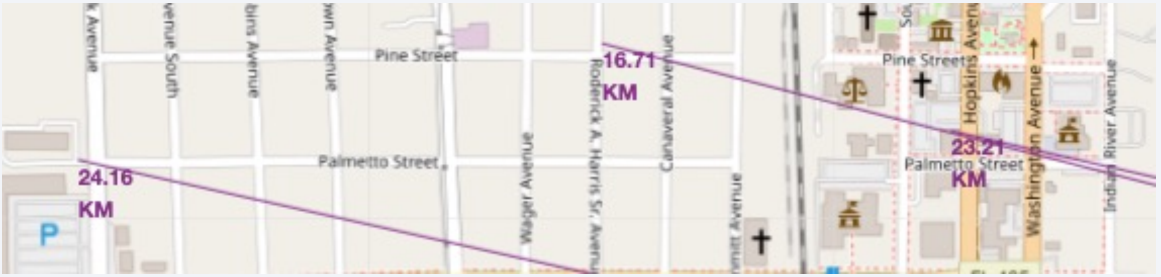




# CCAFS LC-40 and CCAFS SLC-40 distances to landmarks



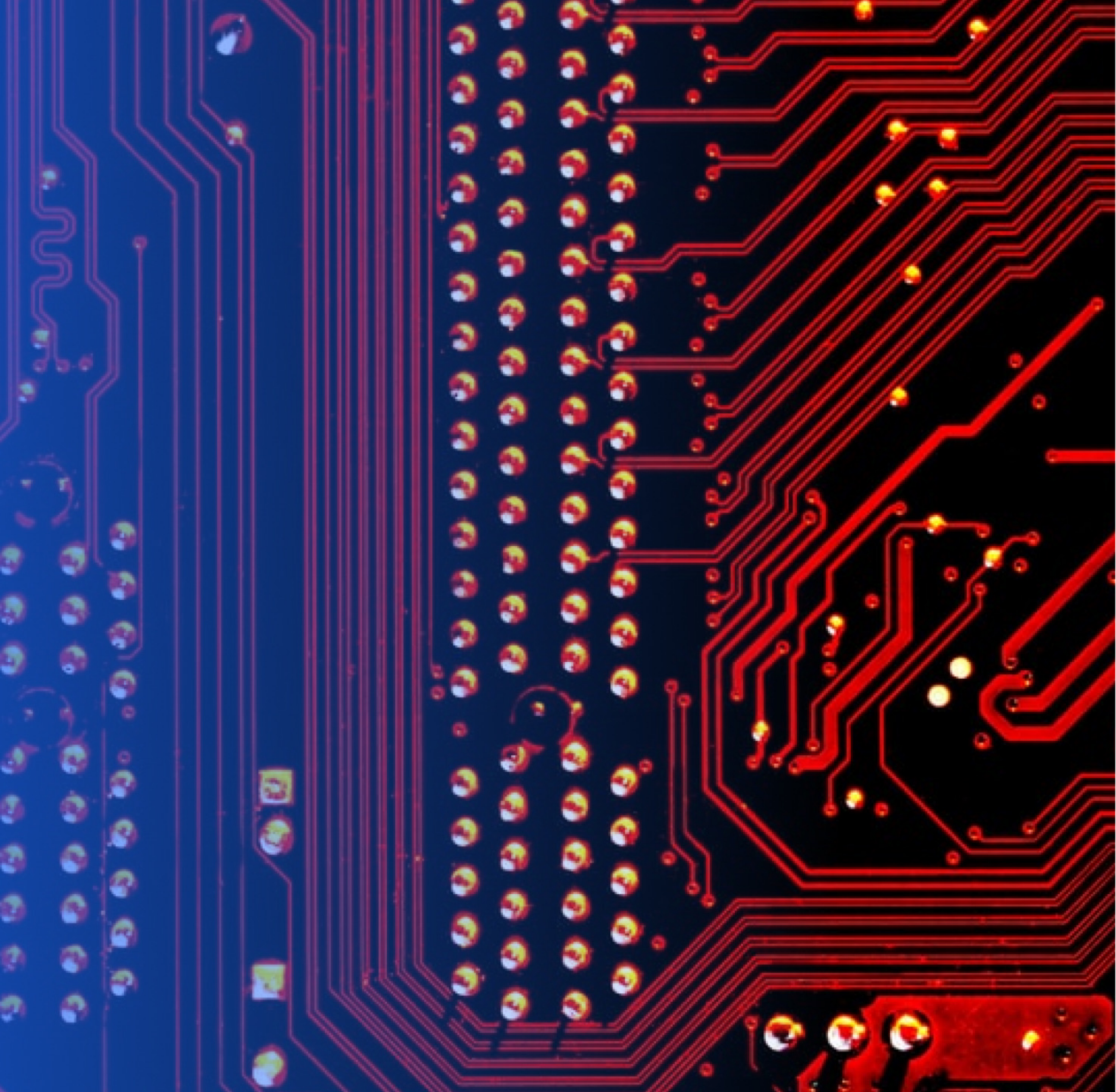

All launch sites are in close proximity to railways. VAFB SLC-4E is 1.26 Km from a railway, while KSC LC-39A is 0.69 Km to a railway. CCAFS LC-40 and CCAFS SLC-40 are 1.29 Km close to a railway.

They are also close to highways. VAFB SLC-4E is 1.18 Km from Lasalle Canyon Road, and KSC LC-39A is 0,85 Km to Kennedy Parkway North. CCAFS LC-40 and CCAFS SLC-40 are 1.25 Km to Titan III Road.

All launch sites are in close proximity to coastline, with VAFB SLC-4E being 1.35 km from the coastline.

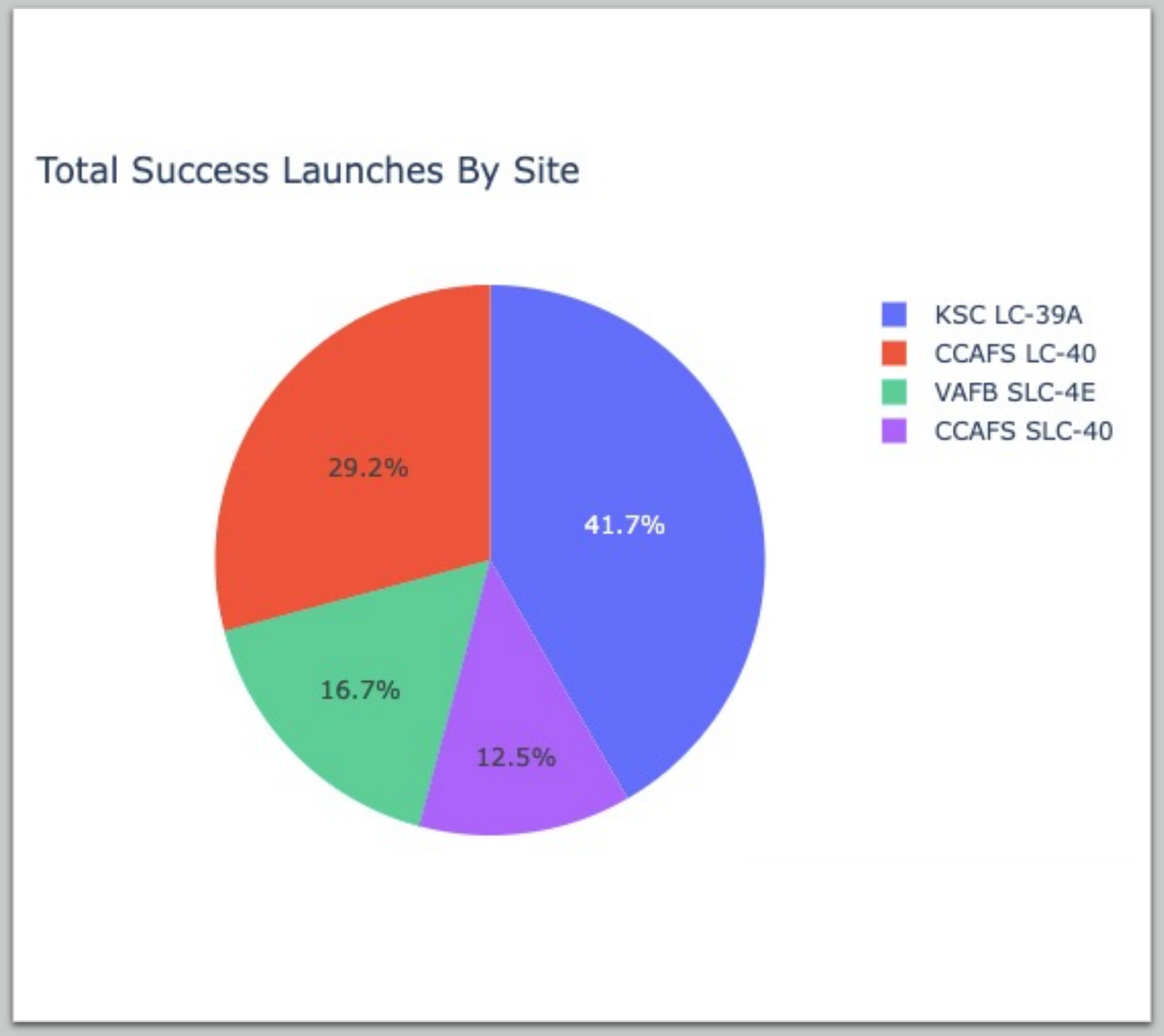
They keep certain distance away from cities. VAFB SLC-4E is 13.99 KM from Lompoc, and KSC LC-39A is 16,71 KM from Titusville. CCAFS LC-40 and CCAFS SLC-40 are 23.21 Km and 24.16 Km from Titusville.

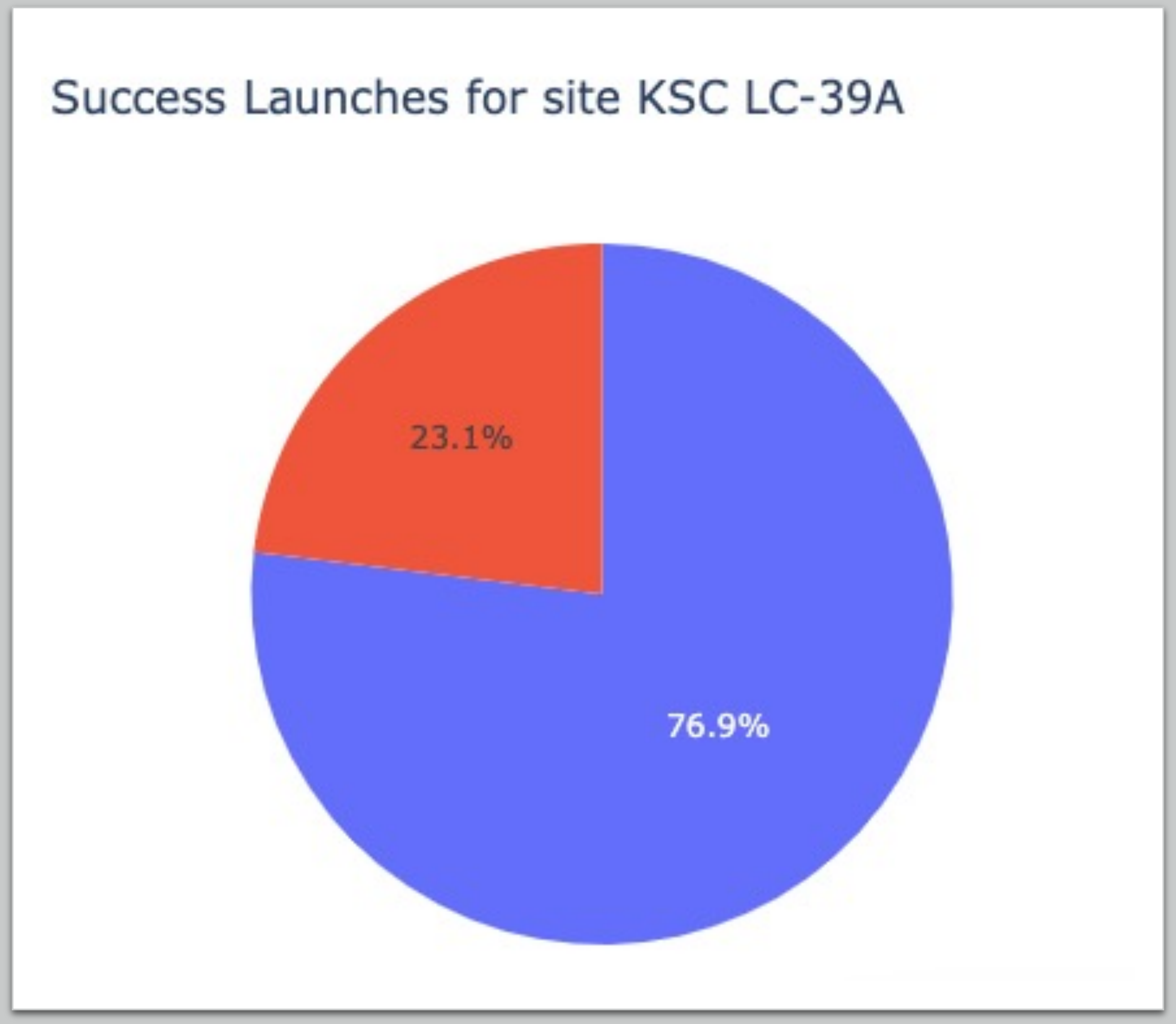
## 4.3
# Build a Dashboard with Plotly Dash



# Total Success Launches by Site

- In this Pie Chart from the Dashboard built it is possible to observe that KSC LC-39A has the greatest rate of successful launches.
- On the other hand, CCAFS SLC-40 has the lowest rate of successful launches.



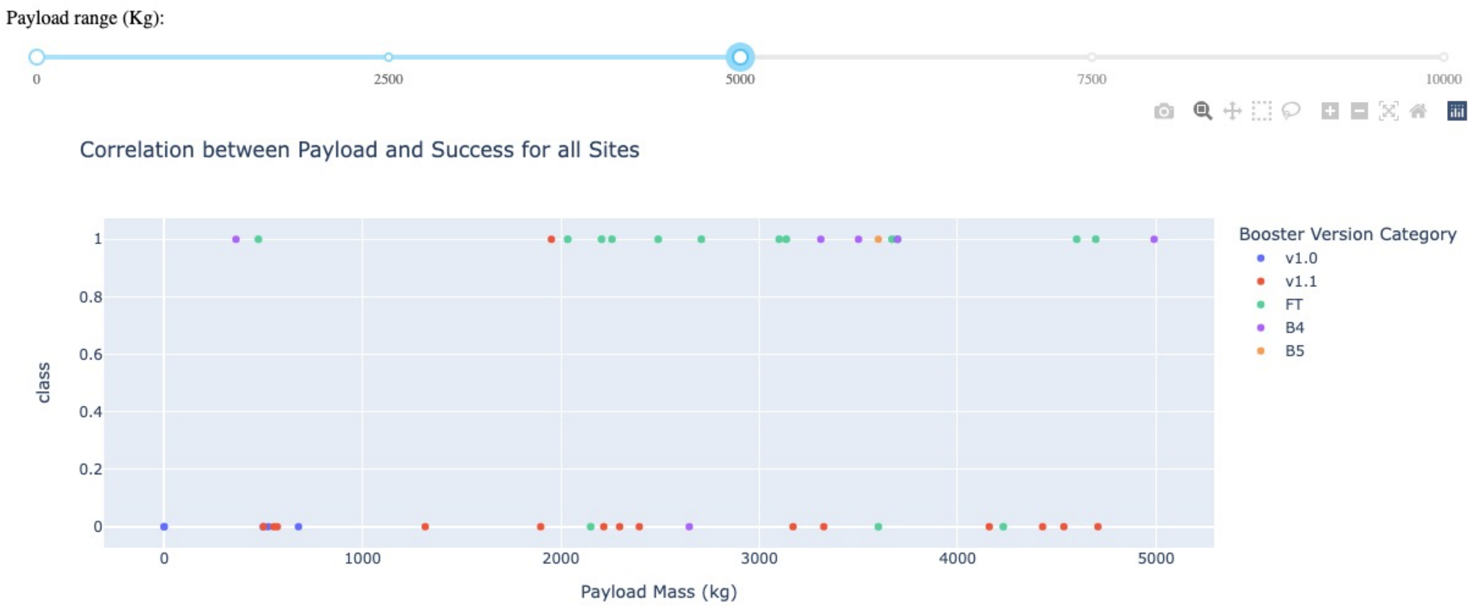



# Launch Site with highest launch success ratio

- Detailing the launch site with highest launch success ratio (KSC LC-39A), the rate of successful launches is 76,9%.

- Only 23,1% of the launches in site KSC LC-39A ended in failure.

# Payload vs. Launch Outcome scatter plot




This Scatter Chart from the Dashboard built demonstrates the differences between payload and launch outcome. When light payloads are selected in the range slider (0-5000 kg), the successful rate increases, with FT booster being the most successful. When heavy payloads are selected in the range slider (5000-10000 kg), the successful rate decreases consistently.

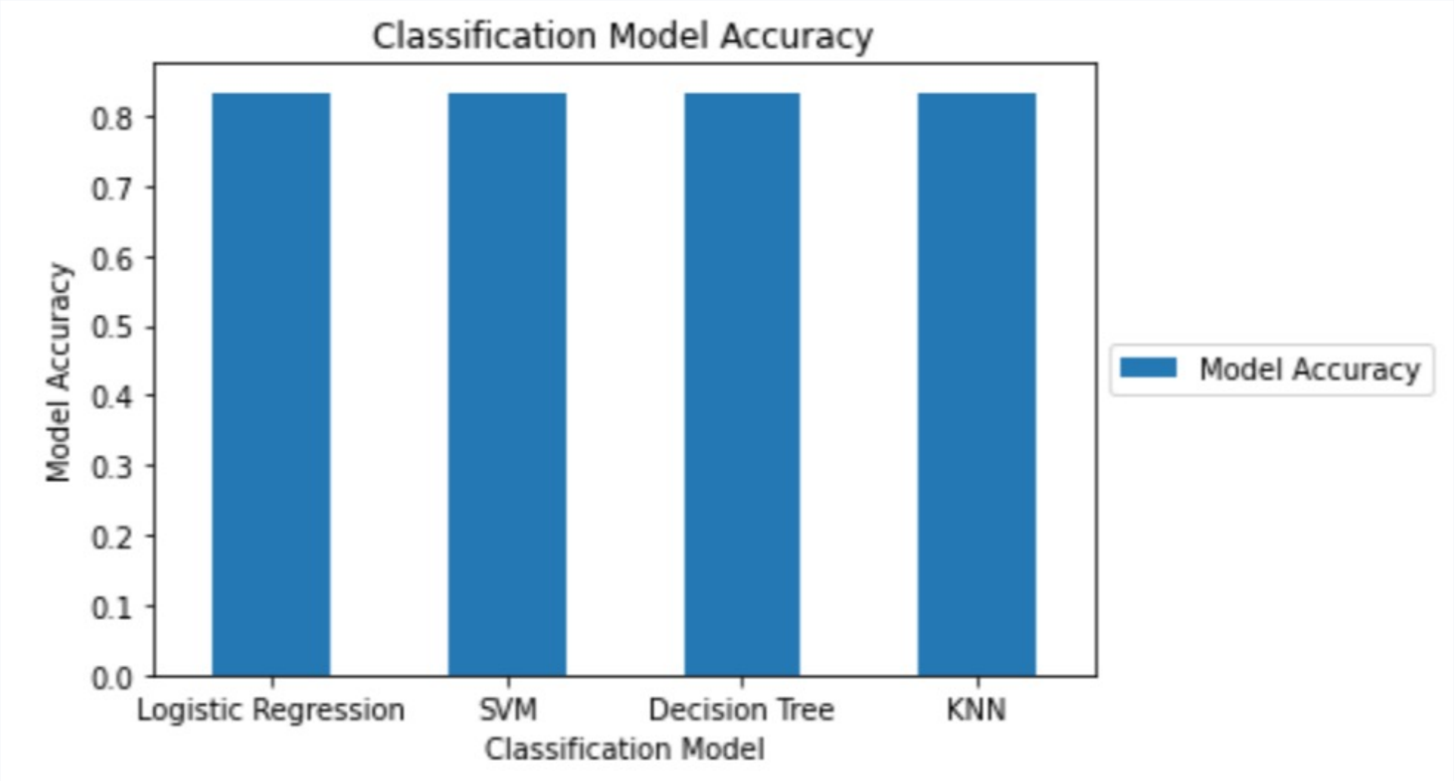
# 4.4
# Predictive Analysis (Classification)

# Classification Accuracy

- All four models built have the same accuracy with test data (0,83) as depicted in the Bar chart

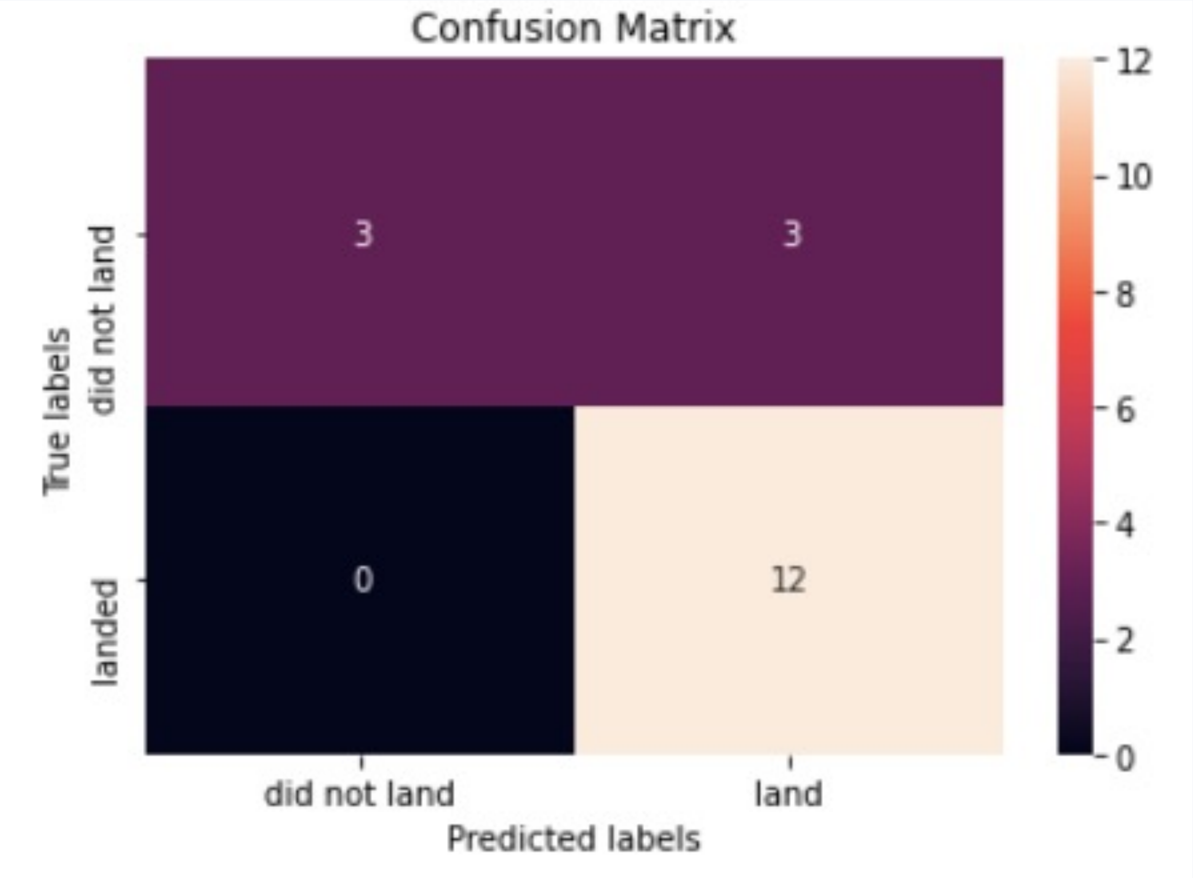
- Even trying F1 score, the accuracy result is the same for them (0,81)

| Algorithm | F1-score |
|---|---|
| KNN | 0.814815 |
| Decision Tree | 0.814815 |
| SVM | 0.814815 |
| LogisticRegression | 0.814815 |




Therefore, all of them could be considered the best model for this project. But in view of processing savings, the model elected is the Logistic Regression

# Confusion Matrix




- The confusion matrix for Logistic Regression Model shows that it can distinguish between the different classes, and the major problem is false positives (3)

- The model has a high rate of true positives (12), and a moderate rate of true negatives (3)

# Conclusions

- The Project objective was achieved, since was possible to build, tune and evaluate a ML model to determine if SpaceX first stage will land successfully.

- Based on this determination, it becomes possible to define if the first stage can be reused and determine the price of each launch to compete against SpaceX.

- The Logistic Regression model was defined as the best model to be used in production environment.

# Conclusions

The project made possible to conclude that:

- The success rate improves when the amount of flight grows in a specific site. For example, CCAFS SLC 40 success rate grow up with more Flight Numbers.

- Less Payload Mass (below 4000 kg) has better success probability in all sites.

- With heavy payloads, the successful landing rate is higher for Polar, LEO and ISS orbits. With light payloads, the success rate is better for ES-L1 and SSO.

# Conclusions

- ES-L1, GEO, HEO and SSO orbits have the highest success rate. GTO has the lowest success rate.

- In the LEO orbit, success appears related to the number of flights, while VLEO orbit has more success rate with greater Flight Numbers.

- The success rate since 2013 kept increasing till 2020, with a slight decrease in 2018.

# Conclusions

- All Launch Sites are in US coasts, specifically in Florida and California, and VAFB SLC-4E is the more distant site from the Equator line.
- All launch sites are near railways, highways, and coastline, but they keep certain distance away from cities.
- Site KSC LC-39A had more successful launches than other sites. CCAFS SLC-40 has the lowest rate of successful launches.

# Appendix

- Link to GitHub with all material, including notebooks, python code for Dashboard and this presentation: https://github.com/jbcjr/IBM-Applied-Data-Science-Capstone

# Thank you!