IBM Developer
SKILLS NETWORK

# Winning Space Race with Data Science

Anas Wicaksono
July, $22^{nd}$ 2022

# Outline

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

# Executive Summary

- Summary of methodologies
- Summary of all results

# Introduction

- Project background and context
  - Space X is the most successful company on Earth and Space, and whats more important is Space X make it even more affordable. The company offer this product only for 62 million, much more affordable than the competitor 165 millions each, because Space X with their Falcon 9 can reuse their rockets. If we can predict it better, wheter if the first stage will land or not, we can determine the cost of the launch. We are going to make a predictive model if Space X will reuse it or not
- Problems you want to find answers
  - How do each variables affect the success rate of the first stage landing?
  - Does the success rate increase over the years ?
  - What is the best machine learning algorithm for this case?

Section 1

# Methodology

# Methodology

## Executive Summary

- Data collection methodology:
  - SpaceX Rest API
  - Webscraping from Wikipedia
- Perform data wrangling
  - Data filtering
  - Dealing with missing values
  - Using one hot encoding to transform categorical data into numerical one
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Building, tuning, and evaluation of classification models to ensure the best results

# Data Collection

- Data collection process involved a combination of API from SpaceX REST API and Web Scraping data from a table in SpaceX's Wikipedia entry
- We use both sources in order to complete information about the launches for a more accurate analysis
  - Data Columns are obtained using SpaceX REST API
    - FlightNumber, Date, BoosterVersion, Payloadmass, Orbit, LaunchSite, Outcome, Flights, GridFins, Reused, Legs, LandingPad, Block, ReusedCount, Serial, Longitude, Latitude
  - Data Columns are obtained by using Wikipedia Web Scraping
    - Flight No, Launch site, Payload, Payloadmass, Orbit, Customer, Launch Outcome, Version Booster, Booster Landing, Date, Time

# Data Collection – SpaceX API

Requesting rocket launch data via SpaceX API

Transform response content using .json() and turning it into a dataframe using .json_normalize()

Request information needed about the launches from SpaceX API by applying custom functions

Constructing data we have obtained into a dictionary

Exporting the result into csv

Filling missing values of PayLoadMass column with mean from this column

Filtering the dataframe to only include Falcon 9 launches

Creating a dataframe from the dictionary

GITHUB : https://github.com/anaswick/coursera_data_science_capstone_project

# Data Collection - Scraping

Requesting Falcon 9 launch data from Wikipedia

Creating a BeautifulSoup objeck from the HTML response

Extracting all column names from the HTML table header

Collecting the data by parsing HTML tables

Constructing data we have obtained into a dictionary

Creating a dataframe from the dictionary

Exporting the data to CSV

GITHUB : https://github.com/anaswick/coursera_data_science_capstone_project

# Data Wrangling

Perform Exploratory Data Analysis (EDA) and determine training labels

Calculate the number of launches on each site

Calculate the number and occurrence of each orbit

Calculate the number and occurrence of mission outcome per orbit type

Create a landing outcome lable from Outcome column

Exporting data to CSV

GITHUB : https://github.com/anaswick/coursera_data_science_capstone_project

# EDA with Data Visualization

- I plot these columns into charts :
  - Flight number vs Payload Mass, Flight Number vs Launch Site, Payload Mass vs Launch Site, Orbit Type vs Success Rate, Flight Number vs Orbit Type, Payload Mass vs Orbit Type, Success Rate Yearly Trend
- Scatter plot show the relationshop between variables. If a relationship exists they could be used for machine learning model
- Bar charts show comparisons between categories. The goal is to show the relationship between the specific categories being compared and measured value.
- Using line chart can show the data trend over time

GITHUB : https://github.com/anaswick/coursera_data_science_capstone_project

# EDA with SQL

Performed SQL queries:
• Displaying the names of the unique launch sites in the space mission
• Displaying 5 records where launch sites begin with the string ‘CCA'
• Displaying the total payload mass carried by boosters launched by NASA (CRS)
• Displaying average payload mass carried by booster version F9 v1.1
• Listing the date when the first successful landing outcome in ground pad was achieved
• Listing the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000
• Listing the total number of successful and failure mission outcomes
• Listing the names of the booster versions which have carried the maximum payload mass
• Listing the failed landing outcomes in drone ship, their booster versions and launch site names for the months in year 2015
• Ranking 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 : https://github.com/anaswick/coursera_data_science_capstone_project

# Build an Interactive Map with Folium

**Markers of all Launch Sites:**
- Added Marker with Circle, Popup Label and Text Label of NASA Johnson Space Center using its latitude and longitude coordinates as a start location.
- Added Markers with Circle, Popup Label and Text Label of all Launch Sites using their latitude and longitude coordinates to show their geographical locations and proximity to Equator and coasts.

**Coloured Markers of the launch outcomes for each Launch Site:**
- Added coloured Markers of success (Green) and failed (Red) launches using Marker Cluster to identify which launch sites have relatively high success rates.

**Distances between a Launch Site to its proximities:**
- Added coloured Lines to show distances between the Launch Site KSC LC-39A (as an example) and its proximities like Railway, Highway, Coastline and Closest City

GITHUB : [https://github.com/anaswick/coursera_data_science_capstone_project](https://github.com/anaswick/coursera_data_science_capstone_project)

# Build a Dashboard with Plotly Dash

**Launch Sites Dropdown List:**
- Added a dropdown list to enable Launch Site selection.

**Pie Chart showing Success Launches (All Sites/Certain Site):**
- Added a pie chart to show the total successful launches count for all sites and the Success vs. Failed counts for the site, if a specific Launch Site was selected.

**Slider of Payload Mass Range:**
- Added a slider to select Payload range.

**Scatter Chart of Payload Mass vs. Success Rate for the different Booster Versions:**
- Added a scatter chart to show the correlation between Payload and Launch Success

GITHUB : https://github.com/anaswick/coursera_data_science_capstone_project

# Predictive Analysis (Classification)

Creating a NumPy array from the column ‘Class’ in data

Standardize the data with StandardScaler, then fit and transforform it

Split the data into training and test data set with train_test_split function

Create a GridSearchCV object with cv=10 to find the best parameter

Apply GridSearchCV on Logistic Regression, SVM, Decistion Tree, and KNN

Calculating the accuracy on the test dataset using .score() for all models

Examine the result using confusion matrix

Determine the best algorithm by jaccard_score and F1_score metrics

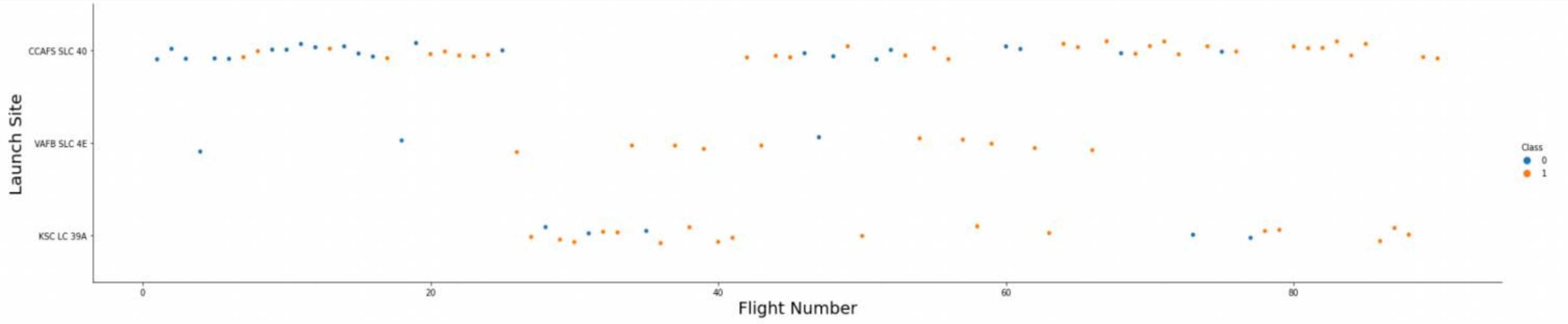
# Results

- Exploratory data analysis results
- Interactive analytics demo in screenshots
- Predictive analysis results

Section 2

# Insights drawn from EDA

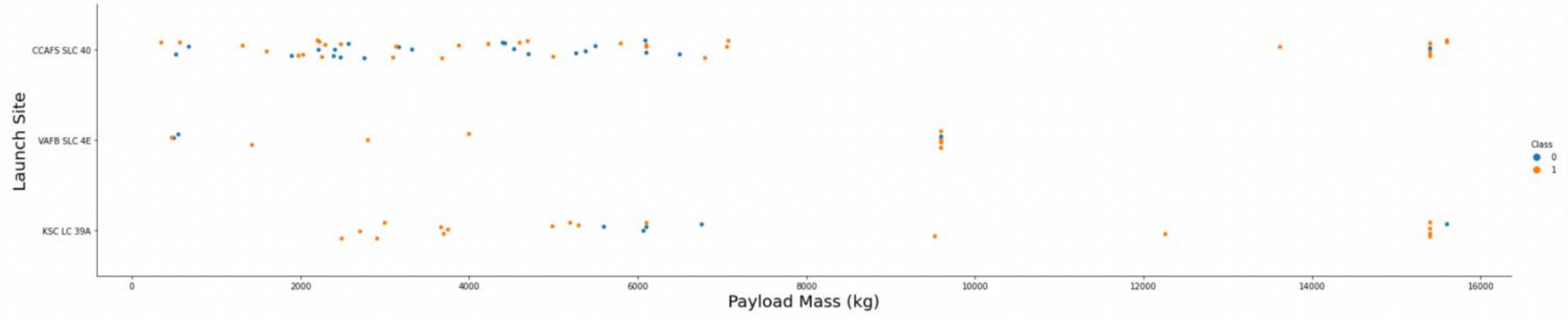
# Flight Number vs. Launch Site




Explanation:

- The earliest flights all failed while the latest flights all succeeded.
- The CCAFS SLC 40 launch site has about a half of all launches.
- VAFB SLC 4E and KSC LC 39A have higher success rates.
- It can be assumed that each new launch has a higher rate of success

# Payload vs. Launch Site



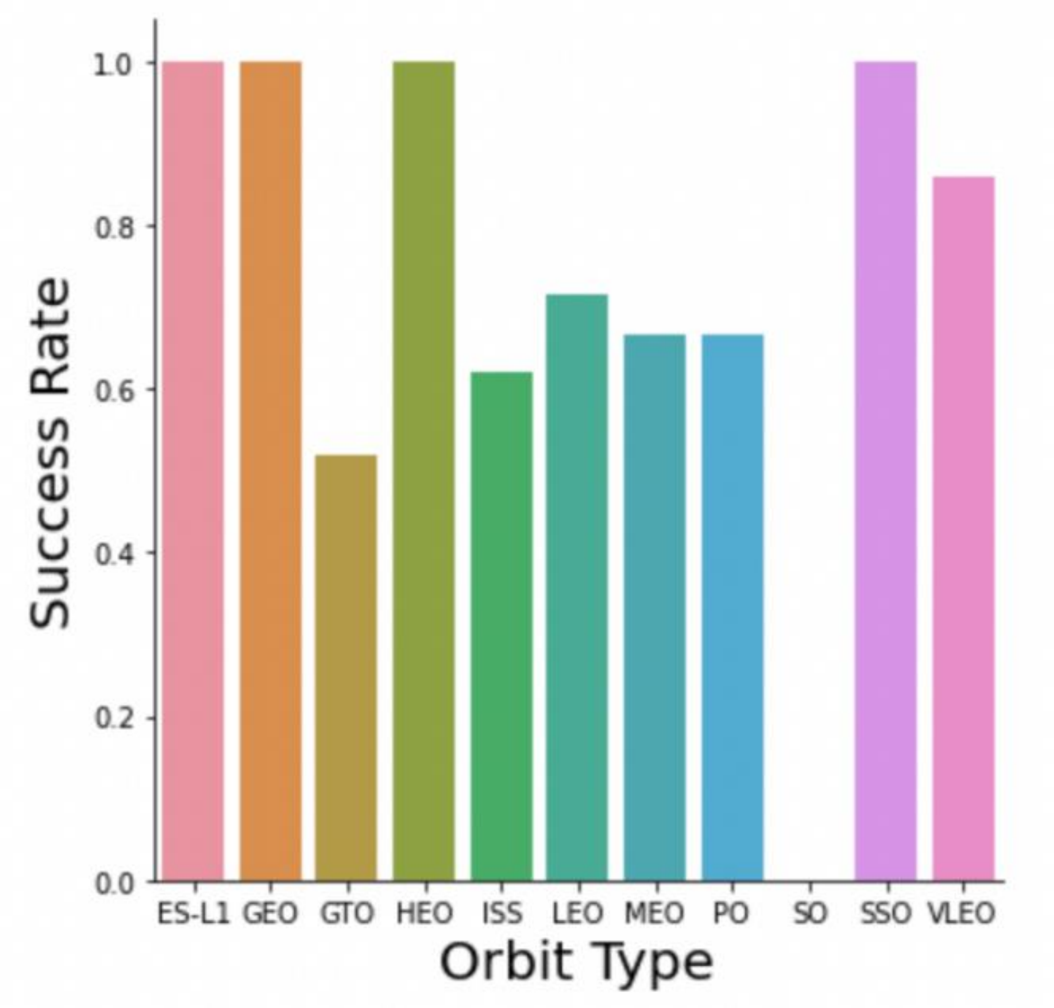

Explanation:

• For every launch site the higher the payload mass, the higher the success rate.

• Most of the launches with payload mass over 7000 kg were successful.

• KSC LC 39A has a 100% success rate for payload mass under 5500 kg too

# Success Rate vs. Orbit Type




Explanation:

• Orbits with 100% success rate:

- ES-L1, GEO, HEO, SSO

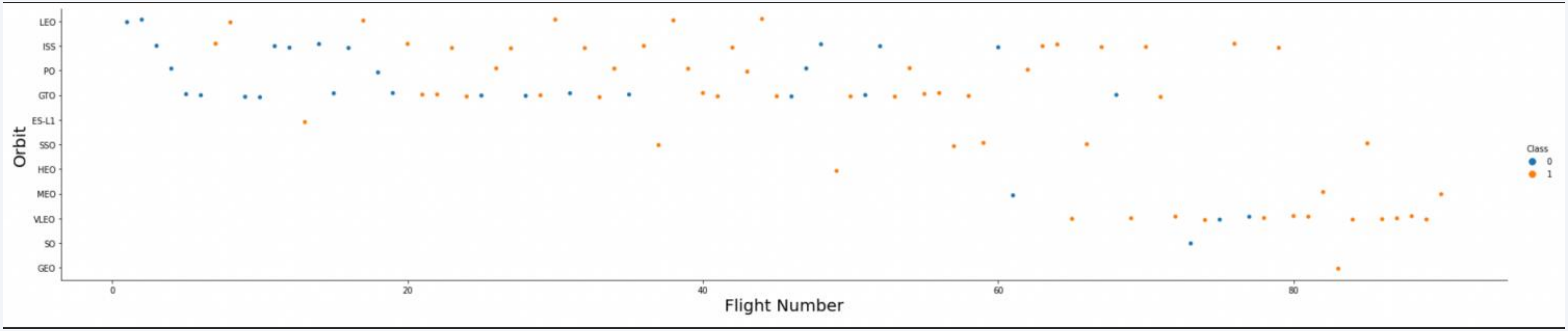
• Orbits with 0% success rate:

- SO

• Orbits with success rate between 50% and 85%:

- GTO, ISS, LEO, MEO, PO

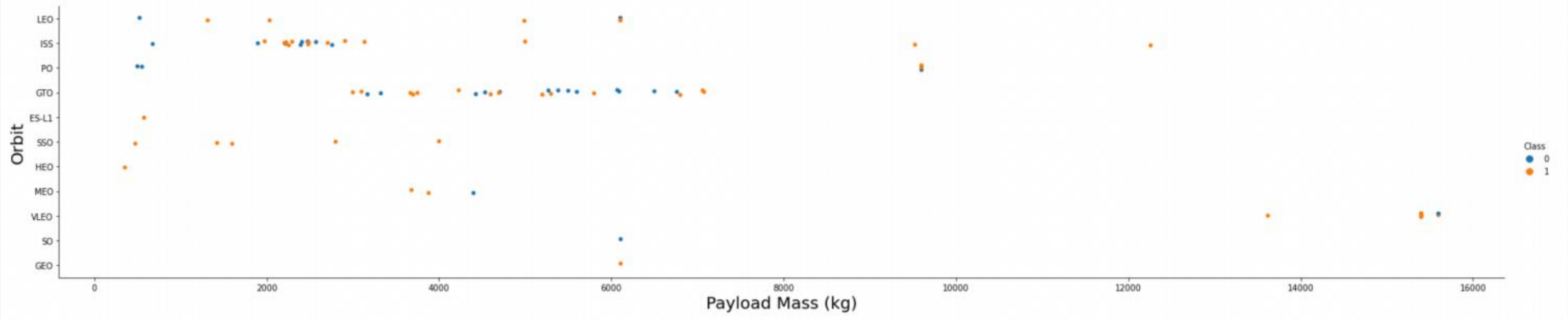
# Flight Number vs. Orbit Type




Explanation:

• 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

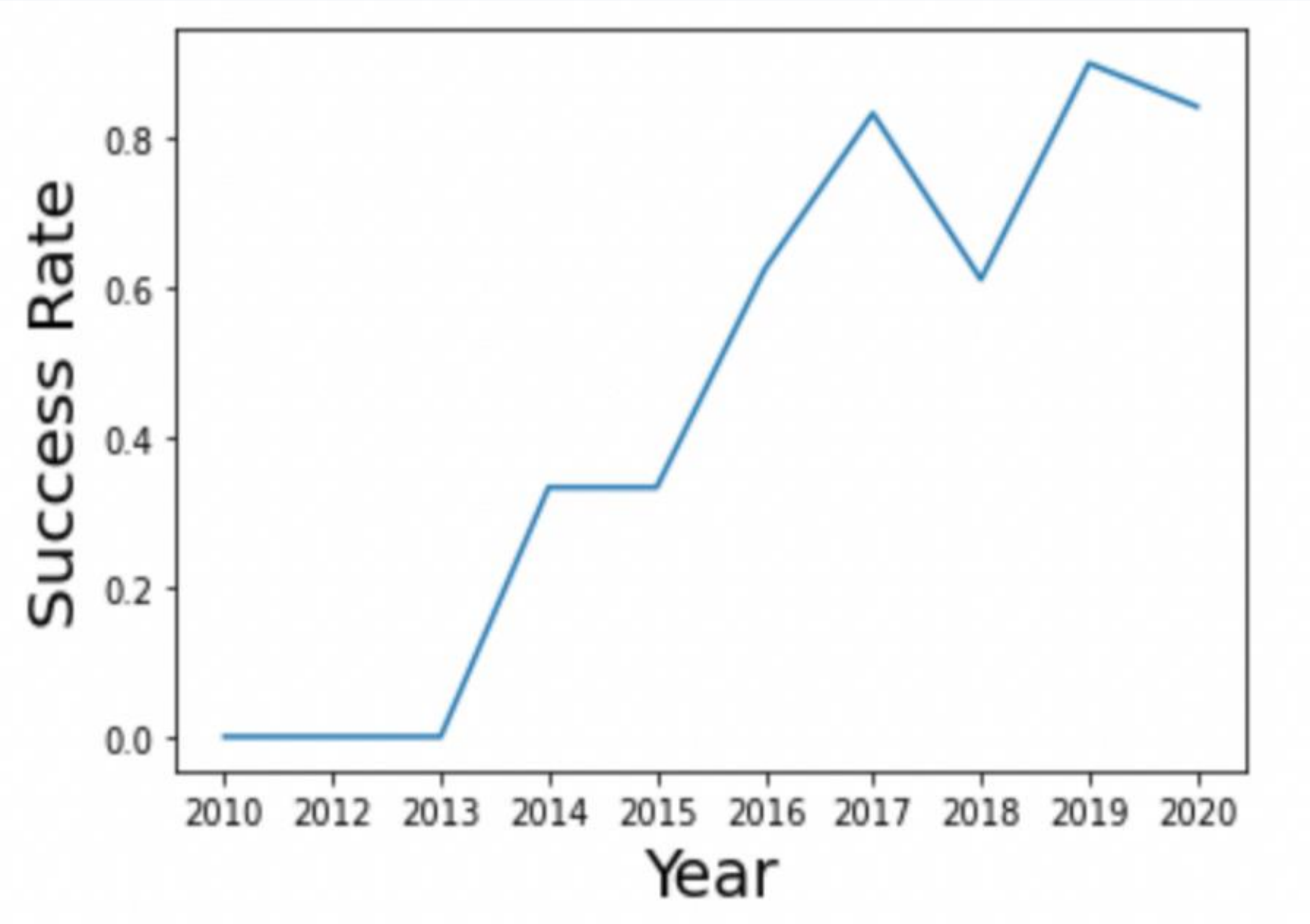
# Payload vs. Orbit Type




Explanation:

• Heavy payloads have a negative influence on GTO orbits and positive on GTO and Polar LEO (ISS) orbits.

# Launch Success Yearly Trend




Explanation:

- The success rate keeps increasing till 2020

# All Launch Site Names

```
In [4]: %sql select distinct launch_site from SPACEXDATASET;
```

```
 * ibm_db_sa://wzf08322:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/bludb
Done.
```

Out[4]:

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

# Launch Site Names Begin with 'CCA'

```
In [5]: %sql select * from SPACEXDATASET where launch_site like 'CCA%' limit 5;

 * ibm_db_sa://wzf08322:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/bludb
Done.
```

Out[5]:

| 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

```
In [6]: %sql select sum(payload_mass__kg_) as total_payload_mass from SPACEXDATASET where customer = 'NASA (CRS)';

 * ibm_db_sa://wzf08322:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/bludb
Done.

Out[6]:
```

| total_payload_mass |
| --- |
| 45596 |

# Average Payload Mass by F9 v1.1

In [7]:
```
%sql select avg(payload_mass__kg_) as average_payload_mass from SPACEXDATASET where booster_version like '%F9 v1.1%';
```

```
 * ibm_db_sa://wzf08322:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/bludb
Done.
```

Out[7]:

| average_payload_mass |
|---|
| 2534 |

# First Successful Ground Landing Date

```
In [8]: %sql select min(date) as first_successful_landing from SPACEXDATASET where landing__outcome = 'Success (ground pad)';

 * ibm_db_sa://wzf08322:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/bludb
Done.

Out[8]:
```

| first_successful_landing |
|---|
| 2015-12-22 |

# Successful Drone Ship Landing with Payload between 4000 and 6000

```
In [9]: %sql select booster_version from SPACEXDATASET where landing__outcome = 'Success (drone ship)' and payload_mass__kg_ between 4
        000 and 6000;

 * ibm_db_sa://wzf08322:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/bludb
Done.
```

Out[9]:

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

# Total Number of Successful and Failure Mission Outcomes

```
In [10]: %sql select mission_outcome, count(*) as total_number from SPACEXDATASET group by mission_outcome;

 * ibm_db_sa://wzf08322:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/bludb
Done.
Out[10]:
```

| mission_outcome | total_number |
|---|---|
| Failure (in flight) | 1 |
| Success | 99 |
| Success (payload status unclear) | 1 |

# Boosters Carried Maximum Payload

```
In [11]: %sql select booster_version from SPACEXDATASET where payload_mass__kg_ = (select max(payload_mass__kg_) from SPACEXDATASET);

 * ibm_db_sa://wzf08322:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/bludb
Done.
```

Out[11]:

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

# 2015 Launch Records

```
In [12]: %%sql select monthname(date) as month, date, booster_version, launch_site, landing__outcome from SPACEXDATASET
               where landing__outcome = 'Failure (drone ship)' and year(date)=2015;

 * ibm_db_sa://wzf08322:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/bludb
Done.
```

Out[12]:

| MONTH | DATE | booster_version | launch_site | landing__outcome |
|---|---|---|---|---|
| January | 2015-01-10 | F9 v1.1 B1012 | CCAFS LC-40 | Failure (drone ship) |
| April | 2015-04-14 | F9 v1.1 B1015 | CCAFS LC-40 | Failure (drone ship) |

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

In [13]:
```
%%sql select landing__outcome, count(*) as count_outcomes from SPACEXDATASET
      where date between '2010-06-04' and '2017-03-20'
      group by landing__outcome
      order by count_outcomes desc;
```

```
 * ibm_db_sa://wzf08322:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/bludb
Done.
```

Out[13]:

| landing__outcome | count_outcomes |
|---|---|
| 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 |

Section 3

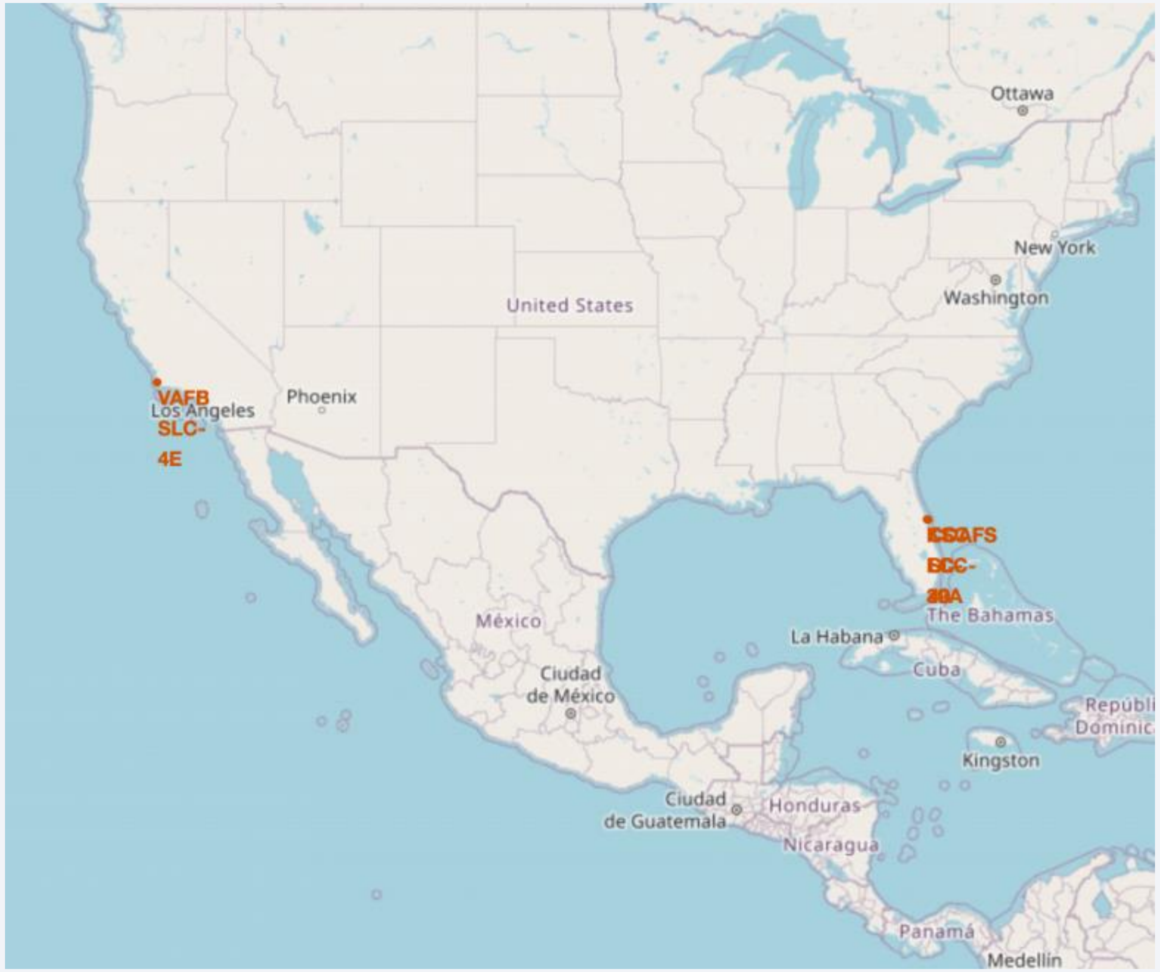
# Launch Sites Proximities Analysis

# All Launch Site markers on Global Map

Explanation:

• Most of Launch sites are in proximity to the Equator line. The land is moving faster at the equator than any other place on the surface of the Earth. Anything on the surface of the Earth at the equator is already moving at 1670 km/hour. If a ship is launched from the equator it goes up into space, and it is also moving around the Earth at the same speed it was moving before launching. This is because of inertia. This speed will help the spacecraft keep up a good enough speed to stay in orbit.

• All launch sites are in very close proximity to the coast, while launching rockets towards the ocean it minimises the risk of having any debris dropping or exploding near people.



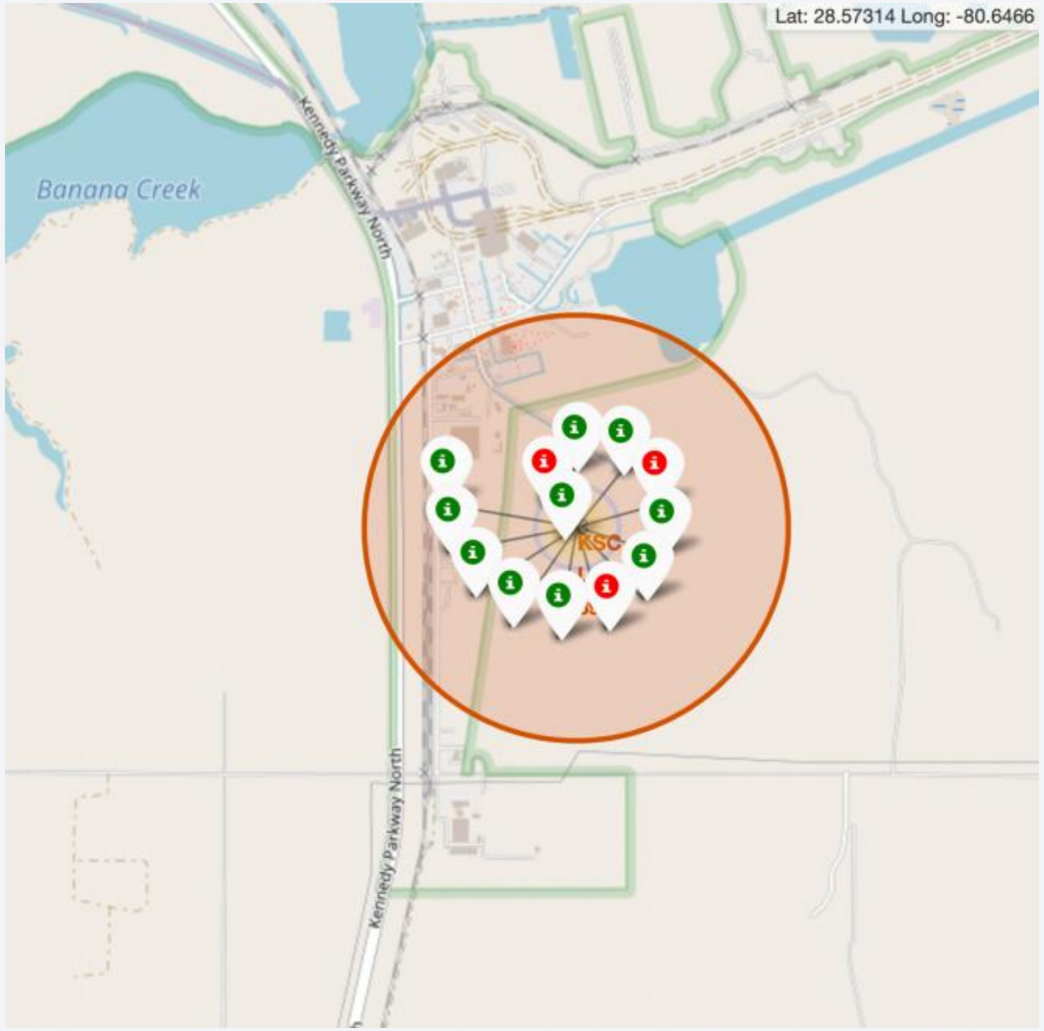
# Color Labled Launch Record on the Map

Explanation:

• From the colour-labeled markers we should be able to easily identify which launch sites have relatively high success rates.

- Green Marker = Successful Launch
- Red Marker = Failed Launch

• Launch Site KSC LC-39A has a very high Success Rate.



# Distance from The Launch Site KSC LC-39A to its Proximities

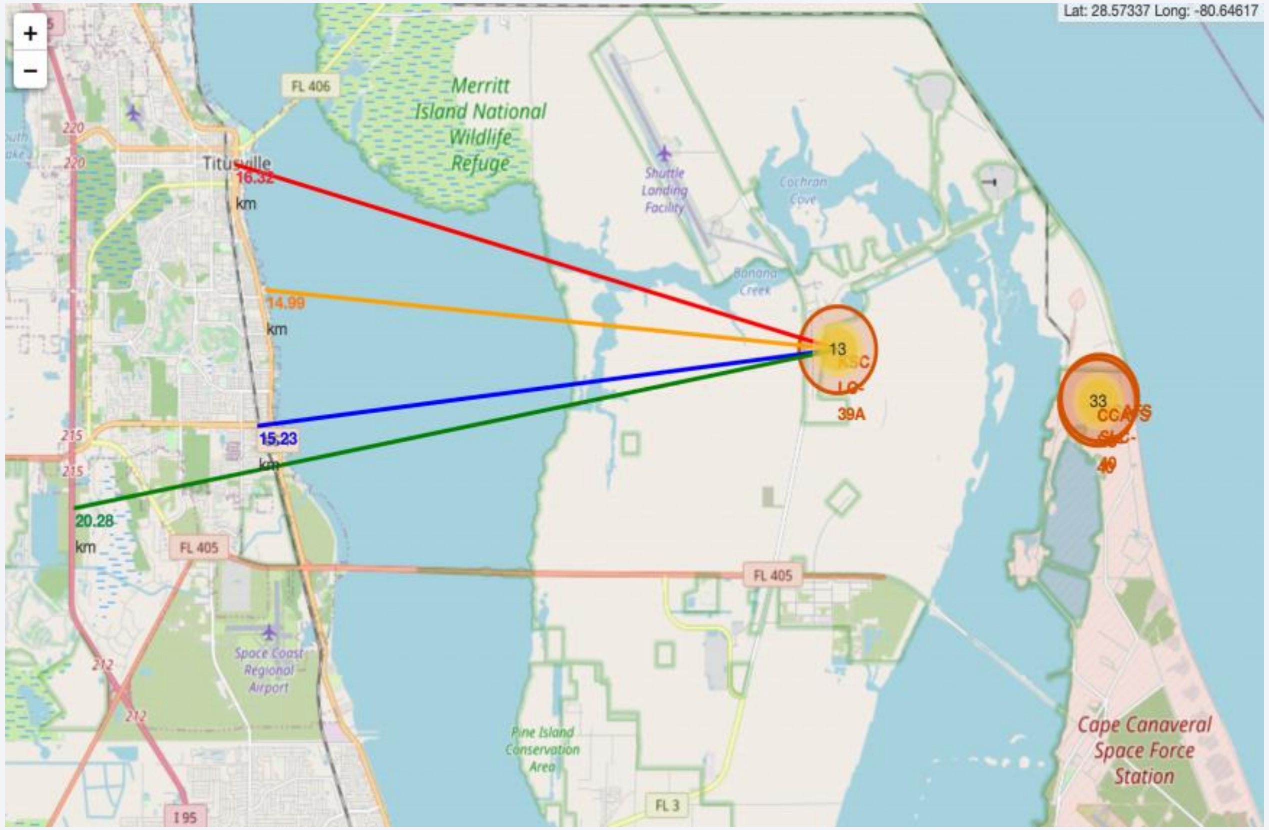
Explanation:

• From the visual analysis of the launch site KSC LC-39A we can clearly see that it is:

- relative close to railway (15.23 km)
- relative close to highway (20.28 km)
- relative close to coastline (14.99 km)

• Also the launch site KSC LC-39A is relative close to its closest city Titusville (16.32 km).

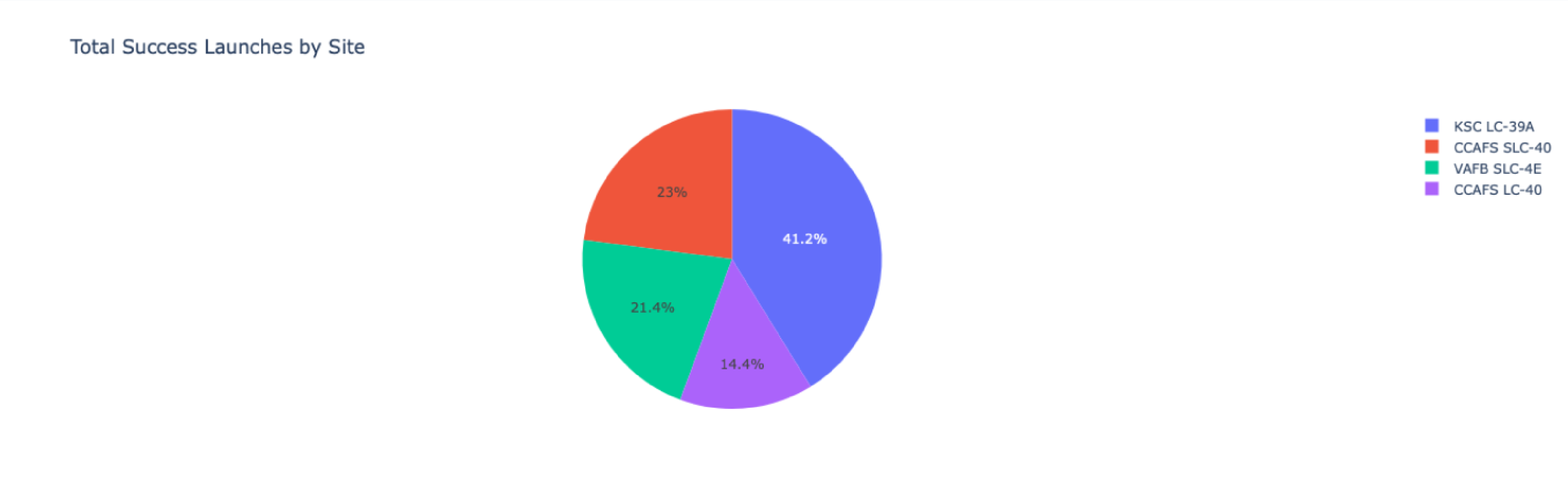
• Failed rocket with its high speed can cover distances like 15-20 km in few seconds. It could be potentially dangerous to populated areas.



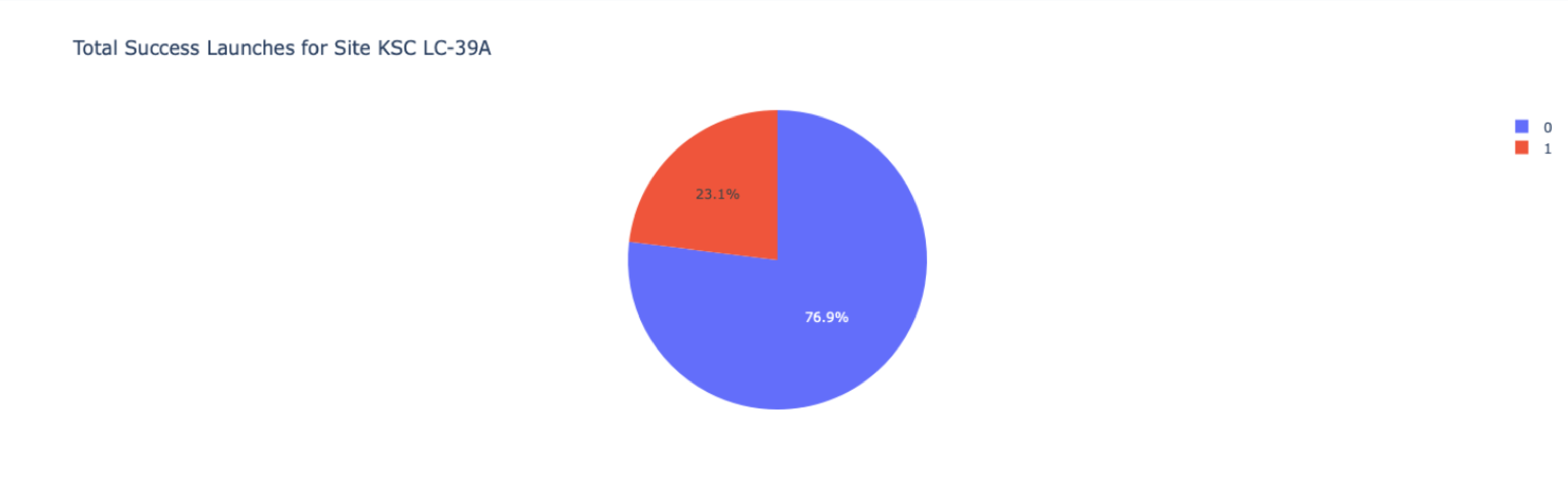
Section 4

# Build a Dashboard with Plotly Dash

# Top Success Launches by Site



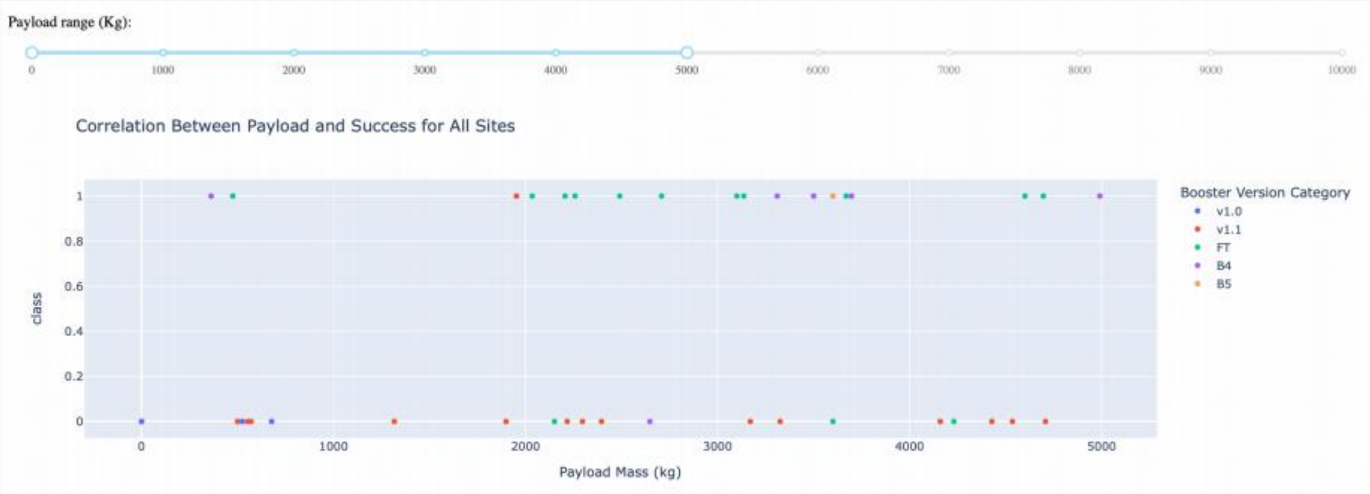
# Launch Site with Highest Success Ratio

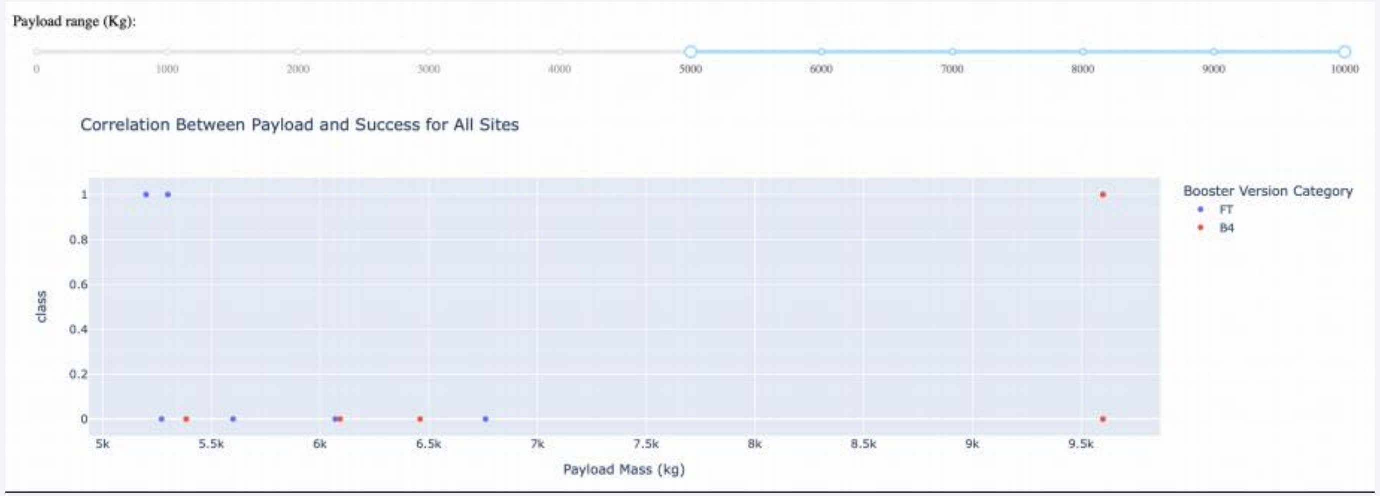


# <Dashboard Screenshot 3>

Explanation:

- The charts show that payloads between 2000 and 5500 kg have the highest success rate



Section 5

# Predictive Analysis (Classification)

# Classification Accuracy

Explanation:

• Based on the scores of the Test Set, we can not confirm which method performs best.

• Same Test Set scores may be due to the small test sample size (18 samples). Therefore, we tested all methods based on the whole Dataset.

• The scores of the whole Dataset confirm that the best model is the Decision Tree Model. This model has not only higher scores, but also the highest accuracy

Accuracy Score on Test Test

| | 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 |

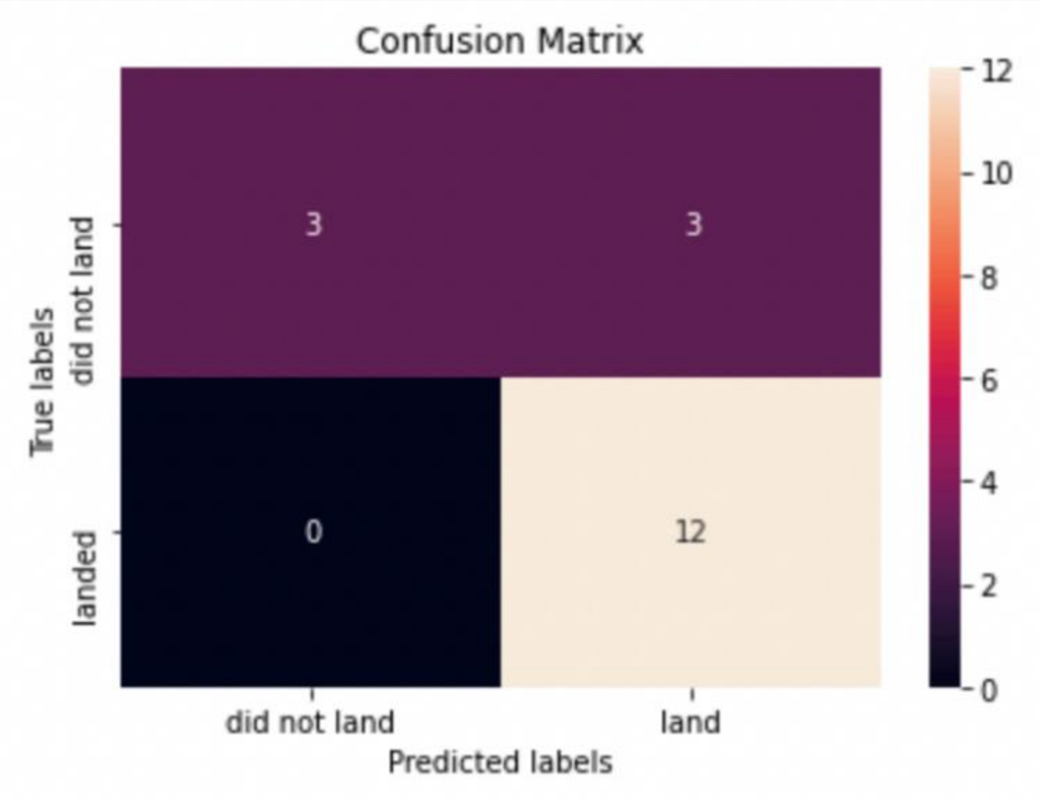
Accuracy Score on Entire Dataset

| | LogReg | SVM | Tree | KNN |
|---|---|---|---|---|
| **Jaccard_Score** | 0.833333 | 0.845070 | 0.882353 | 0.819444 |
| **F1_Score** | 0.909091 | 0.916031 | 0.937500 | 0.900763 |
| **Accuracy** | 0.866667 | 0.877778 | 0.911111 | 0.855556 |

# Confusion Matrix

Explanation:

• Examining the confusion matrix, we see that logistic regression can distinguish between the different classes. We see that the major problem is false positives



# Conclusions

- Decision Tree Model is the best algorithm for this dataset.
- Launches with a low payload mass show better results than launches with a larger payload mass.
- Most of launch sites are in proximity to the Equator line and all the sites are in very close proximity to the coast.
- The success rate of launches increases over the years.
- KSC LC-39A has the highest success rate of the launches from all the sites.
- Orbits ES-L1, GEO, HEO and SSO have 100% success rate.

# Appendix

# Thank you!