IBM Developer
SKILLS NETWORK

# Winning Space Race with Data Science

Miguel Alzate

February 6, 2024

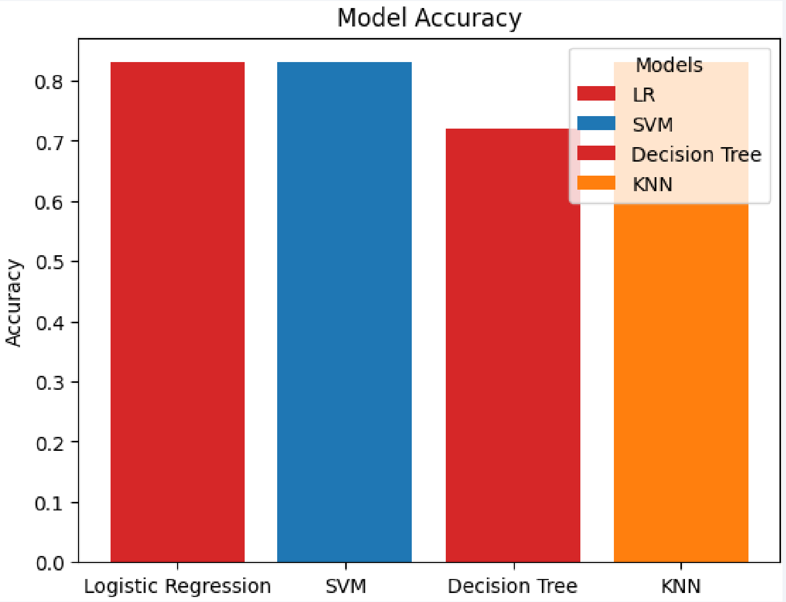
# Outline

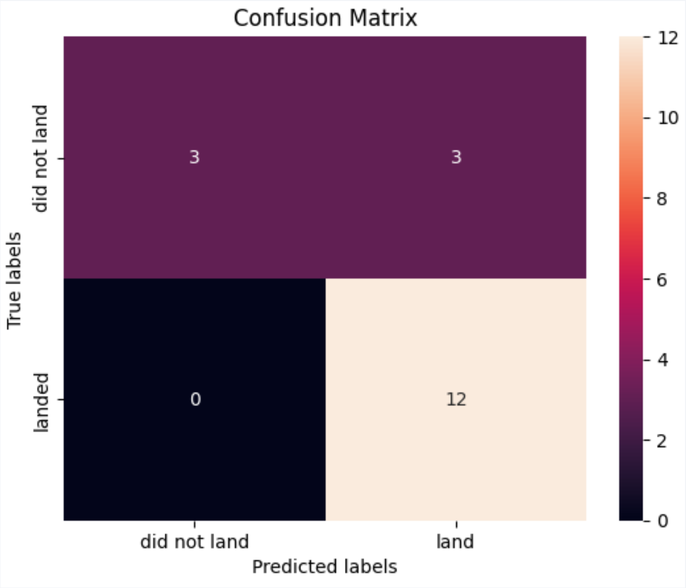
- Executive Summary
- Introduction
- Methodology
- Results
- Conclusion
- Appendix

# Executive Summary

- Methodologies: Data collection, Data wrangling, exploratory data analysis (EDA) using visualization and SQL

For predicting the outcome of a mission: Logistic Regression, SVM and KNN are appropriate models as they yielded an accuracy score of 0.83%





# Introduction

The first stage of a rocket can be recovered fully, by landing it with as the first stage falls into the ground. This new technology has made space rockets least expensive.

Aerospatial physics is a complicated subject. Data Science offers an alternative way of predicting the outcome of a mission using data sets of previous missions and analyzing the variables that influence the outcome of the mission.

This saves a lot of time and effort. The question that we are trying to answer is if the outcome of a mission can be predicted with data! Instead of using complicated rocket physics

Section 1

# Methodology

# Methodology

## Executive Summary

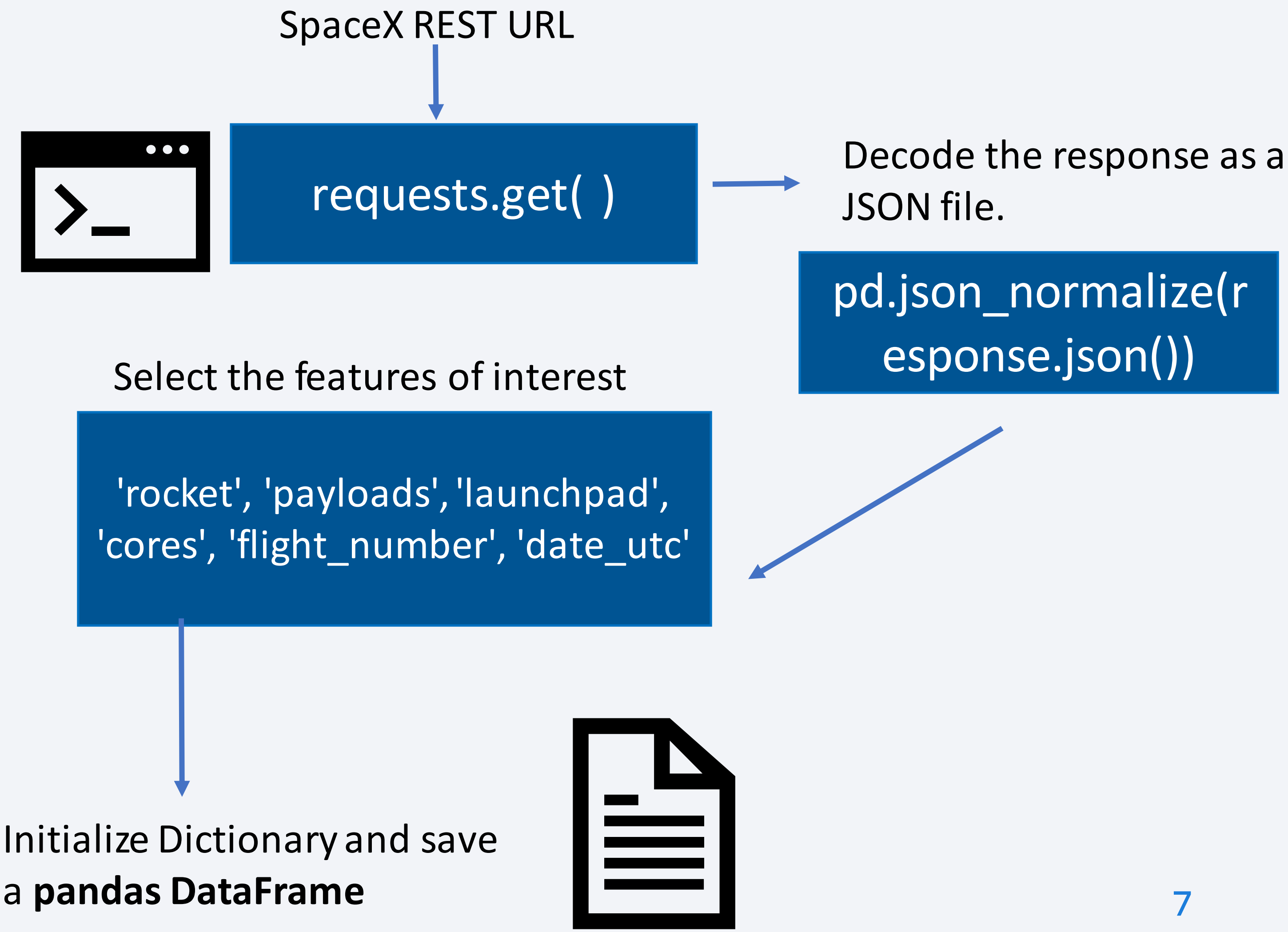
- Data collection methodology:
    - Space X API and web scrapping methods used to obtain the data
- Perform data wrangling
    - Cleaning the data: Remove Null values and add new Outcome column
- Exploratory data analysis (EDA) using visualization and SQL
    - Revealing data insights to choose the variables that have a stronger influence over the outcome of a mission
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
    - Several machine learning models were tested to seek the highest accuracy possible for predictions

# Data Collection – SpaceX API

- SpaceX REST: Python's library "Request"

r-spacex/**SpaceX-API**

- GitHub URL of the Jupyter Notebook for the API calls: https://github.com/Miguel88Alzate/DataScience/blob/main/Data%20Collection%20API%20Lab.ipynb



SpaceX REST URL

requests.get( )

Decode the response as a JSON file.

pd.json_normalize(response.json())

Select the features of interest

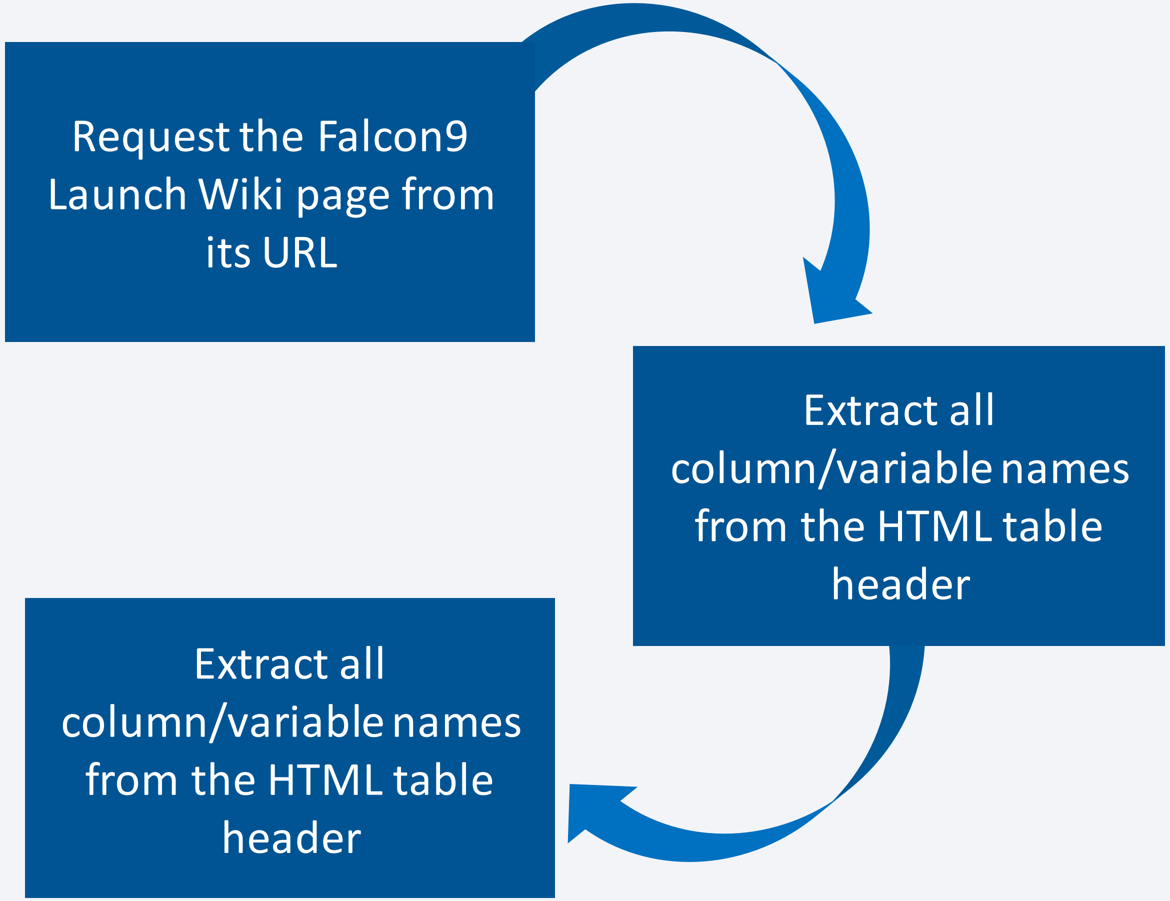
'rocket', 'payloads', 'launchpad', 'cores', 'flight_number', 'date_utc'

Initialize Dictionary and save a **pandas DataFrame**

# Data Collection - Scraping

- Web scraping process using **Beautiful Soup** and Falcon 9 and Falcon Heavy Launches **Records from Wikipedia**

- GitHub URL of the web scraping notebook: https://github.com/Miguel88Alzate/DataScience/blob/main/Web%20scraping.ipynb



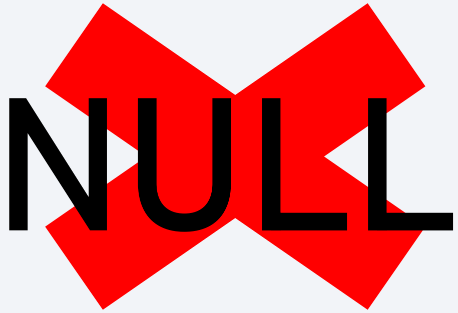
Request the Falcon9 Launch Wiki page from its URL

Extract all column/variable names from the HTML table header

Extract all column/variable names from the HTML table header

# Data Wrangling

- The Null values of the extracted data set were removed and replacing them with the mean of the data in each respective column. A new column 'Class' was added to represent the landing outcomes in binary



## Outcomes Column

Class (0 Failure outcome, 1 successful outcome)

GitHub URL of the Data Wrangling notebook: https://github.com/Miguel88Alzate/DataScience/blob/main/Data%20Wrangling.ipynb

# EDA with SQL

## SQL queries you performed:

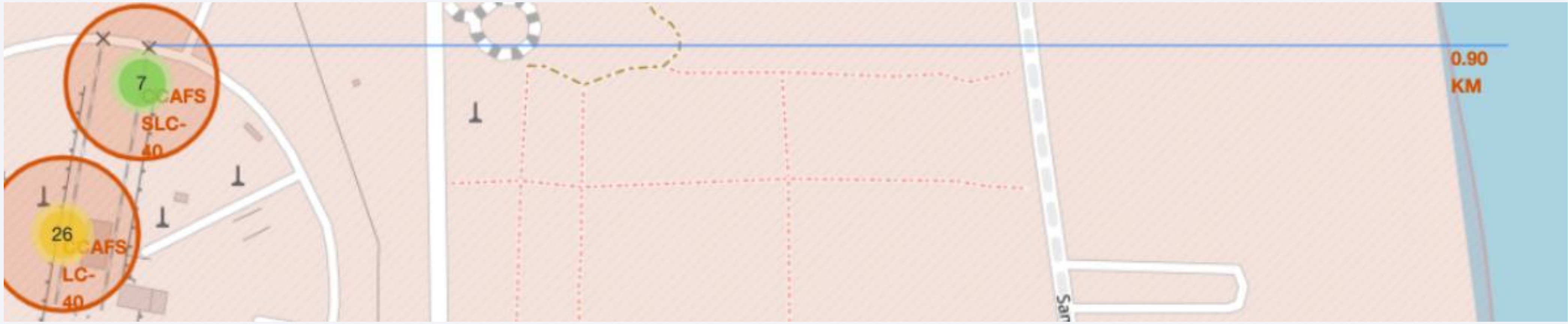
- Names of the unique launch sites in the space mission
- 5 records where launch sites begin with the string 'CCA'
- Total payload mass carried by boosters launched by NASA (CRS)
- Average payload mass carried by booster version F9 v1.1
- The date when the first succesful landing outcome in ground pad was achieved.
- Names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000
- Total number of successful and failure mission outcome
- Names of the booster_versions which have carried the maximum payload mass.
- Records which will display the month names, failure landing_outcomes in drone ship ,booster versions, launch_site for the months in year 2015.
- 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 of the SQL notebook:** https://github.com/Miguel88Alzate/DataScience/blob/main/EDA%20with%20SQL.ipynb

# Interactive Map with Folium

Folium Objects:

- Markers: Select a specific point on the map, as the different launch site
- Circles: Display circle of a specific radius in a geographic location
- Lines: Draw lines to connect points in the map

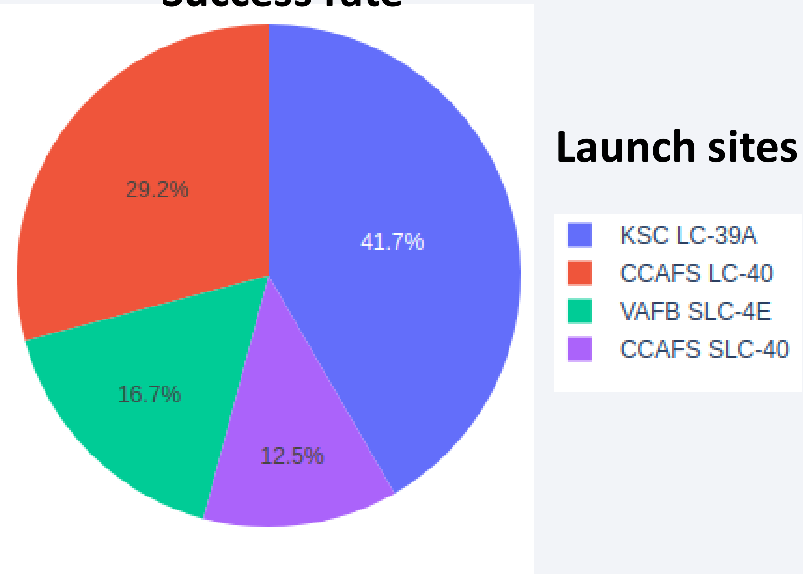


GitHub URL of the Folium notebook: https://github.com/Miguel88Alzate/DataScience/blob/main/Folium.ipynb

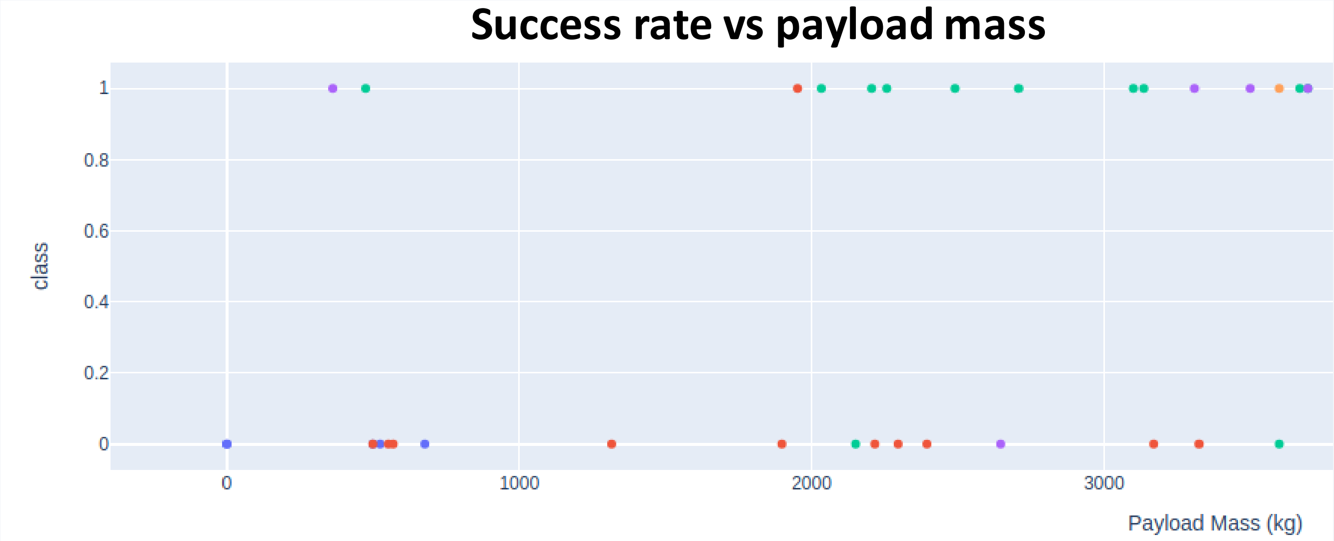
# Build a Dashboard with Plotly Dash

- Main plots: Pie chart displaying the launch sites success rate with the option of selecting an individual launch site. Scatter plot, allows user to check how the Payload mass affect the success rate in each launch site

Success rate



Launch sites



Success rate vs payload mass

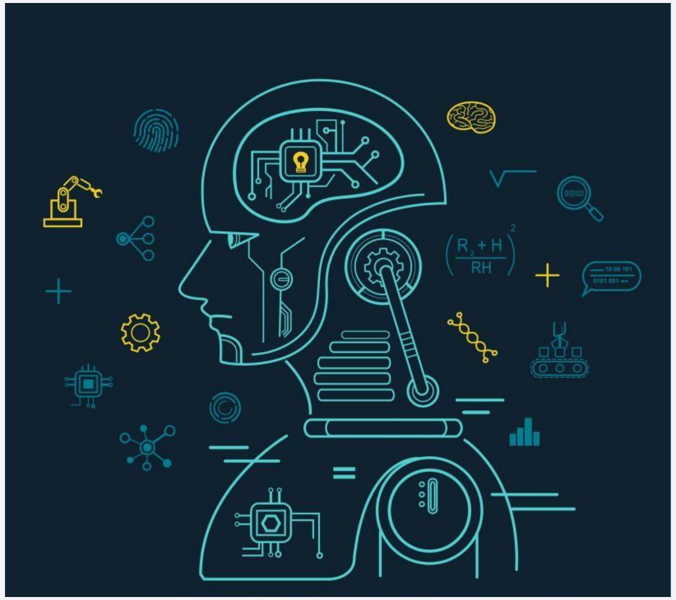
GitHub URL of the Dash lab: https://github.com/Miguel88Alzate/DataScience/blob/main/spacex_dash_app.py

# Predictive Analysis (Classification)

Before testing the models, the data set was **split into train and test sets** with a proportion of **20% for testing**. For all models, the **Sckit Learn Grid Search** was implemented to find the parameters that yielded the highest accuracy.

Models used:

- Logistic Regression
- Support Vector Machine (SVM)
- Decision Tree
- K nearest-neighbors



GitHub URL of the Machine Learning notebook: https://github.com/Miguel88Alzate/DataScience/blob/main/Machine%20Learning%20Prediction%20lab.ipynb

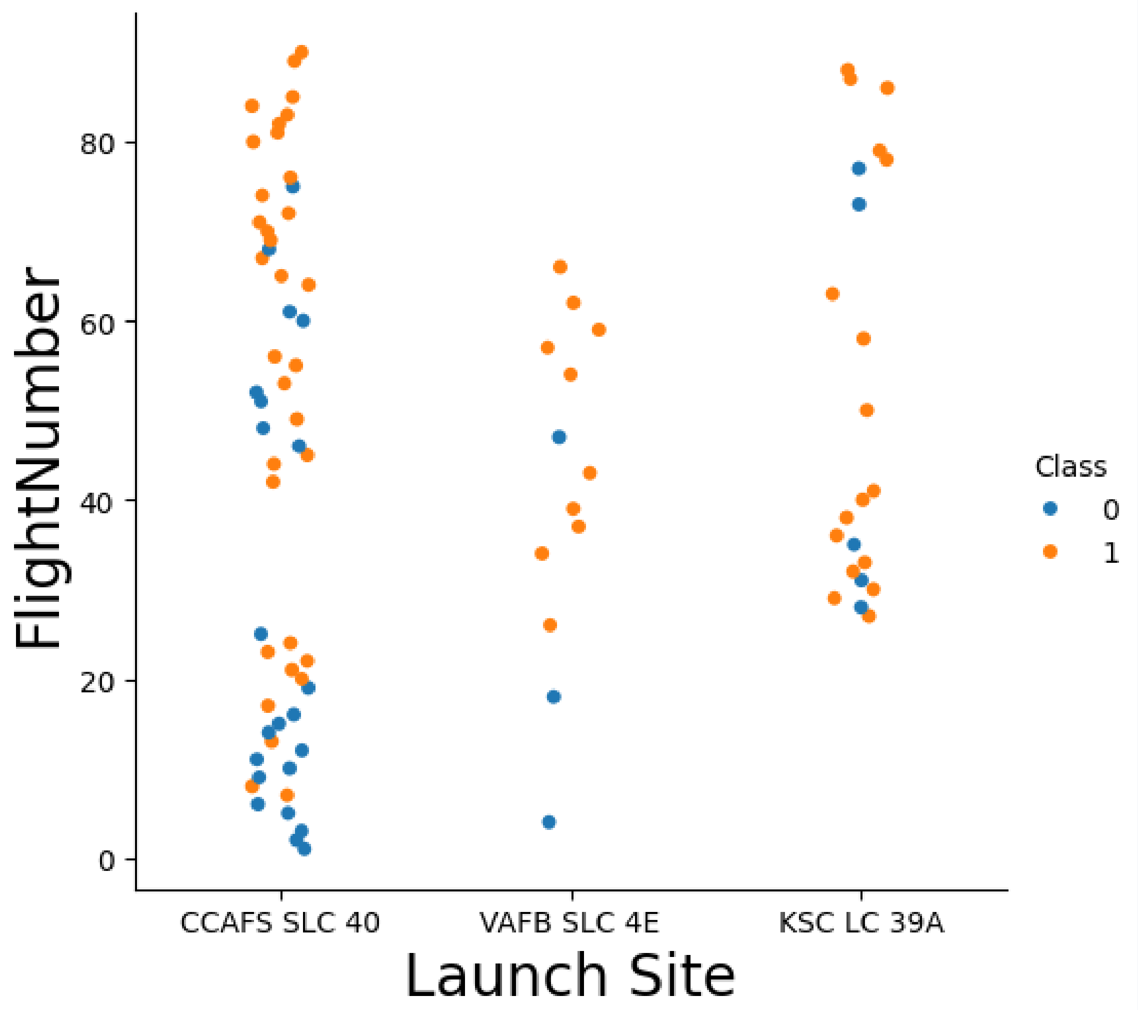
# Results

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

Section 2

# Insights drawn from EDA

# Flight Number vs. Launch Site



## Scatter plot Flight Number vs Launch Site

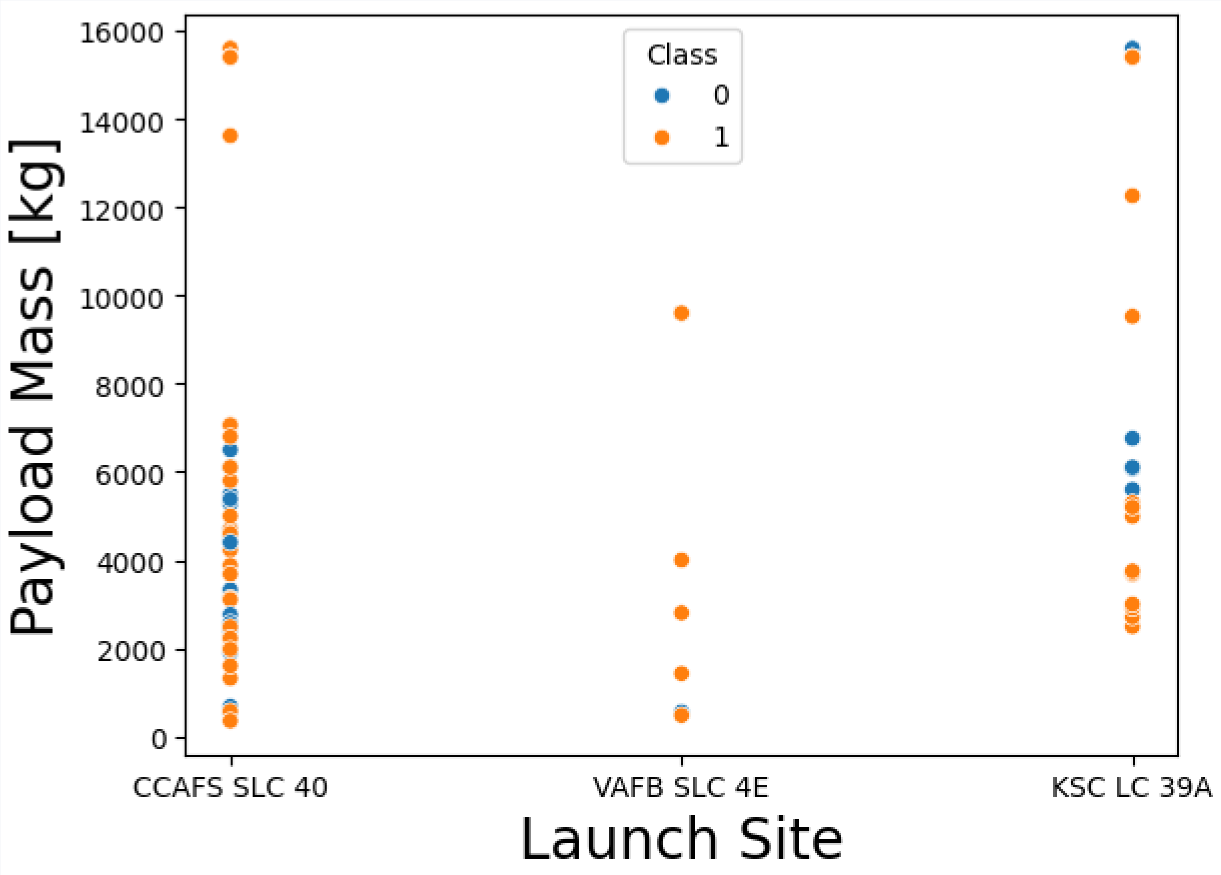
General tendency indicates that increase in flight number yields a successful mission outcome

CCAFS SLC 40: More common launch site. A lot of launches performed that get better with flight number

VAFB SLC 4E: Least common launch site. Two abnormal unsuccessful missions

KSC LC 39: Have the highest success rate. Last missions' outcome were successful

# Payload vs. Launch Site



## Scatter plot Payload vs Launch Site

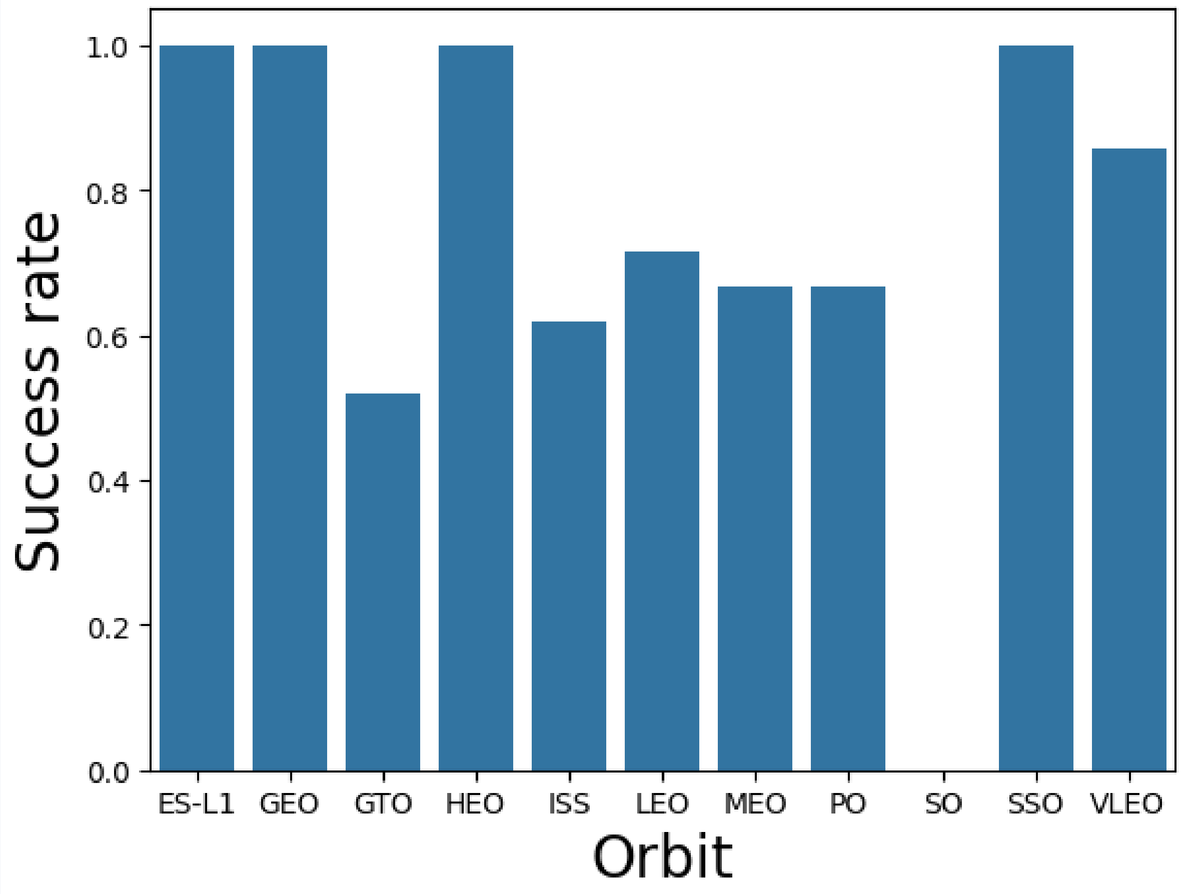
General tendency indicates that in the high payloads the missions' outputs were successful except for an abnormal data point at KSC LC 39A

CCAFS SLC 40: More common launch site. A lot of launches, unsuccessful launches stopped at around 6500 payload mass, above the missions were successful

VAFB SLC 4E: Highest success rate, but also not a lot of launches

KSC LC 39: Last launch with a high payload was unsuccessful

# Success Rate vs. Orbit Type



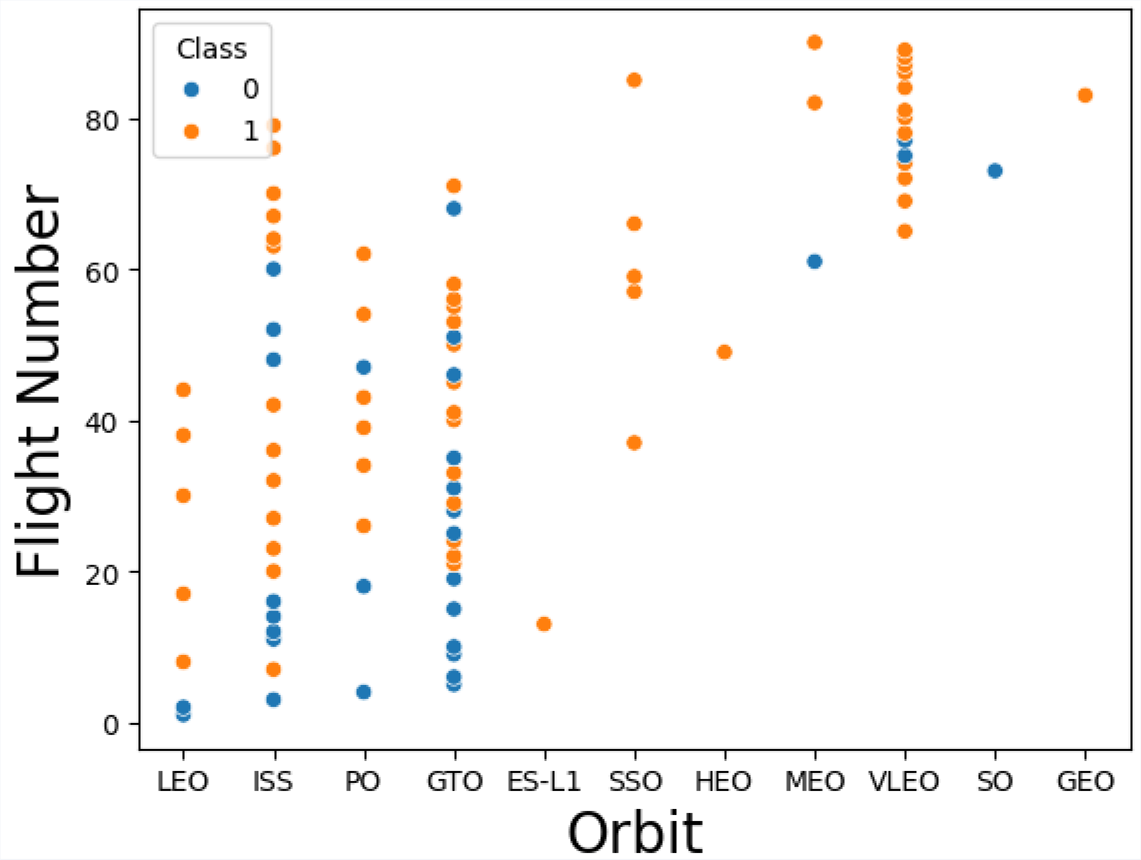
## Bar plot Success rate vs Orbit Type

Success rate for ES-L1, GEO, HEO and SSO orbits had 100% success rate, but also only one launch was attempted, except for SSO (4 attempts)

The second lowest success rate was around 50% and belong to the GTO orbit, associated with the difficulty of getting payloads to the GTO orbit

The lowest success (0 %) rate belongs to SO orbit. Reason: Only 1 attempt was made for that orbit type, which resulted in failure.

# Flight Number vs. Orbit Type



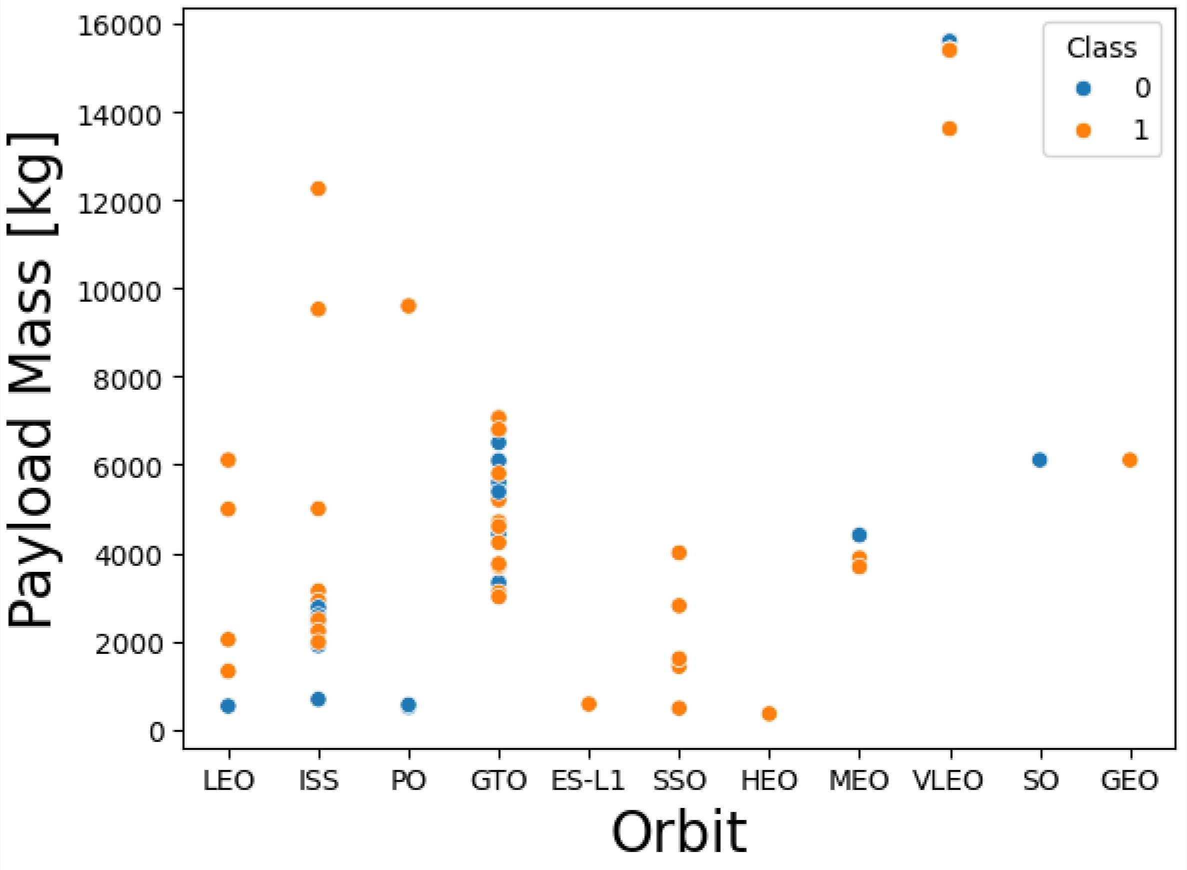
## Scatter plot Flight number vs Orbit Type

The scatter plot shows that when the flight number increases, the common tendency for the orbit type is to yield a successful outcome.

The orbits, ES-L1, SSO HEO, SO and GEO had a few attempts ranging from 1 to 4

The orbits LEO, ISS, PO, GTO and VLEO had more attempts, where GTO orbit presents irregular successful mission outcomes

# Payload vs. Orbit Type



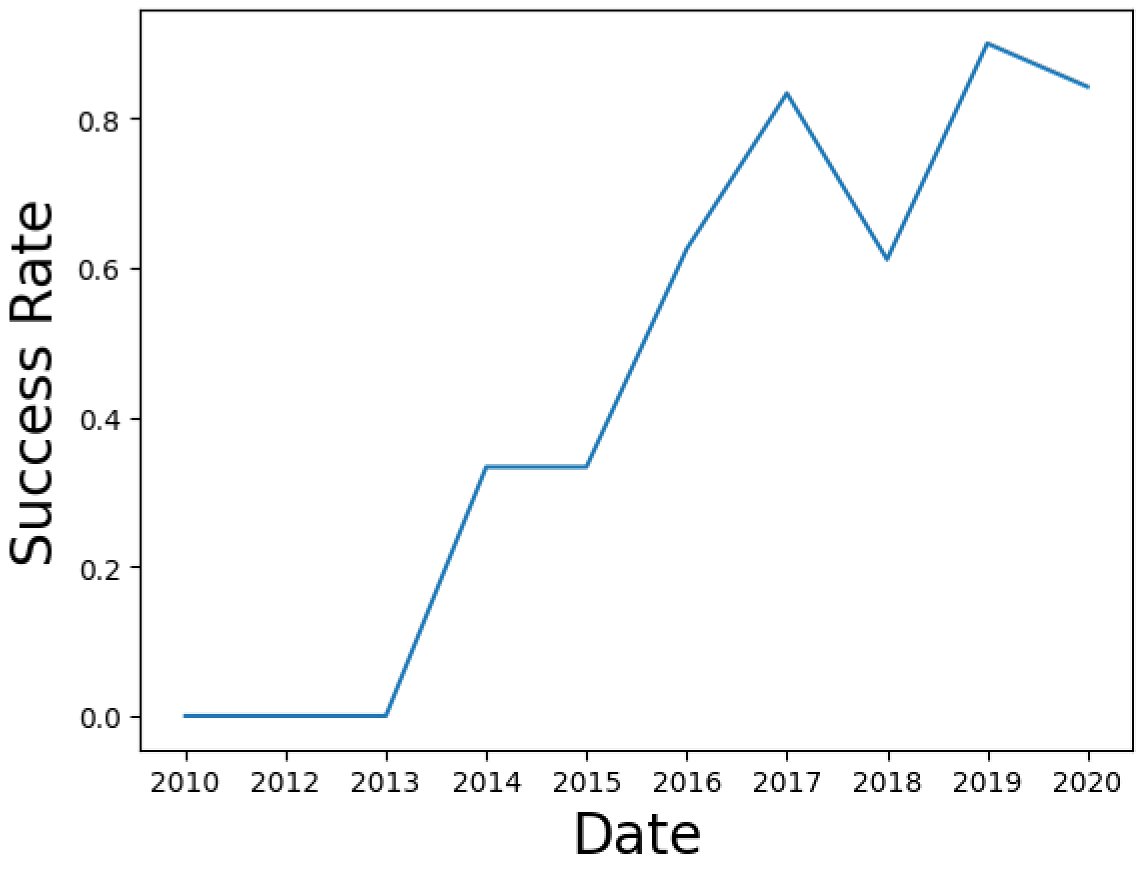
## Scatter plot Payload mass vs Orbit Type

As the payload mass increases, the success rate increases for LEO, ISS and SSO orbits.

GTO orbits had problems yielding a success outcome regularly with payload masses above 5000 kg

MEO orbit payload mass of around 16000 kg had a failure mission outcome out of 3 attempts with payload masses above 13000 kg

# Launch Success Yearly Trend



## Line plot Success rate trend

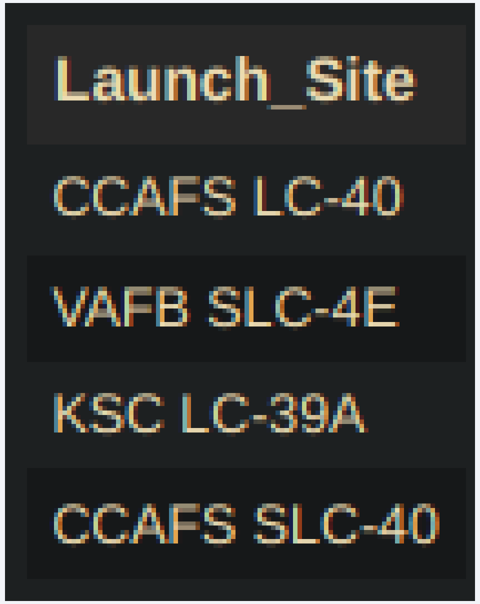
The trend shows an increase in the success rate from 2010 to 2020. The graphs shows that the trend is not a perfect line, but sometimes the success rate can get lower. Nonetheless, the general tend indicates that the success rate will keep increasing

# All Launch Site Names

**QUERY**: %sql select distinct Launch_Site from SPACEXTBL

Select from the column 'Launch_Site' of the SPACEXTBL Data base the distinct launch sites

**RESULT**:



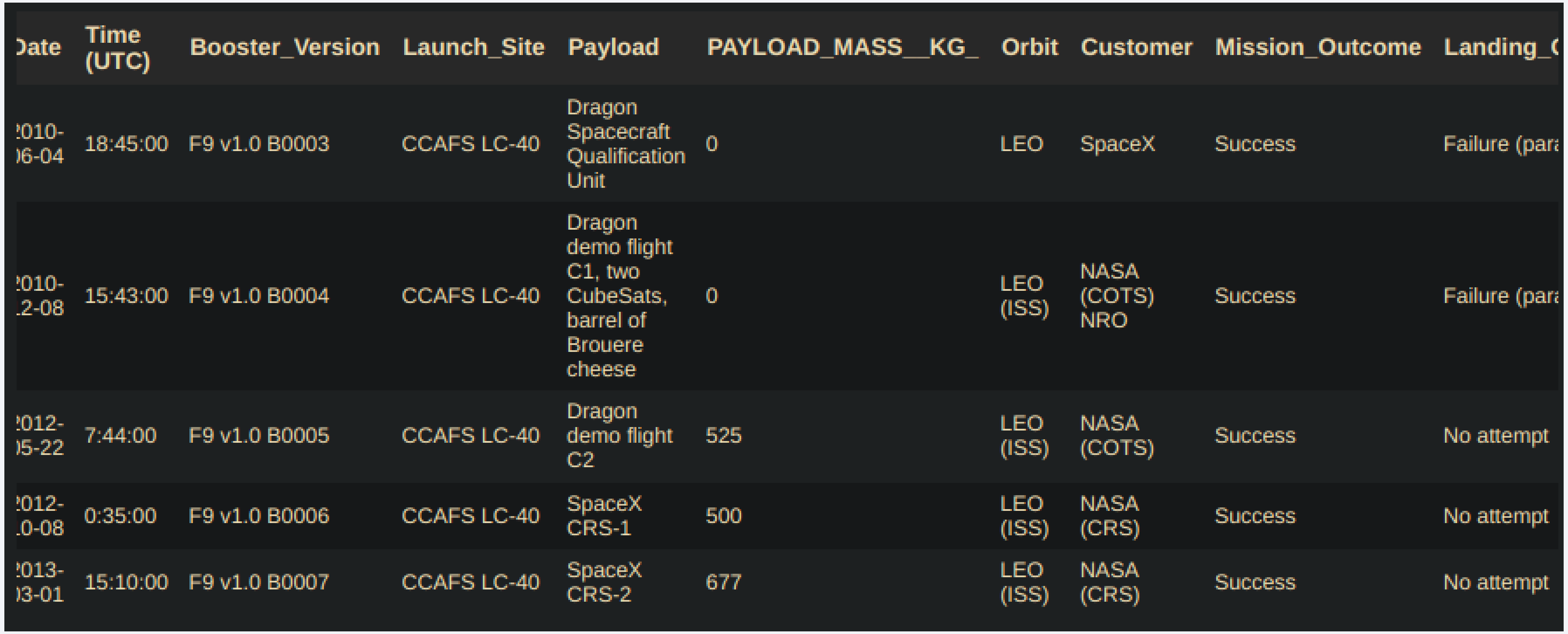
| Launch_Site |
|---|
| CCAFS LC-40 |
| VAFB SLC-4E |
| KSC LC-39A |
| CCAFS SLC-40 |

# Launch Site Names Begin with 'CCA'

**QUERY**: %sql select*from SPACEXTBL where Launch_Site like 'CCA%' limit 5

Select all columns from the SPACEXTBL Data base where the launch site name starts with 'CCA'

**RESULT**:



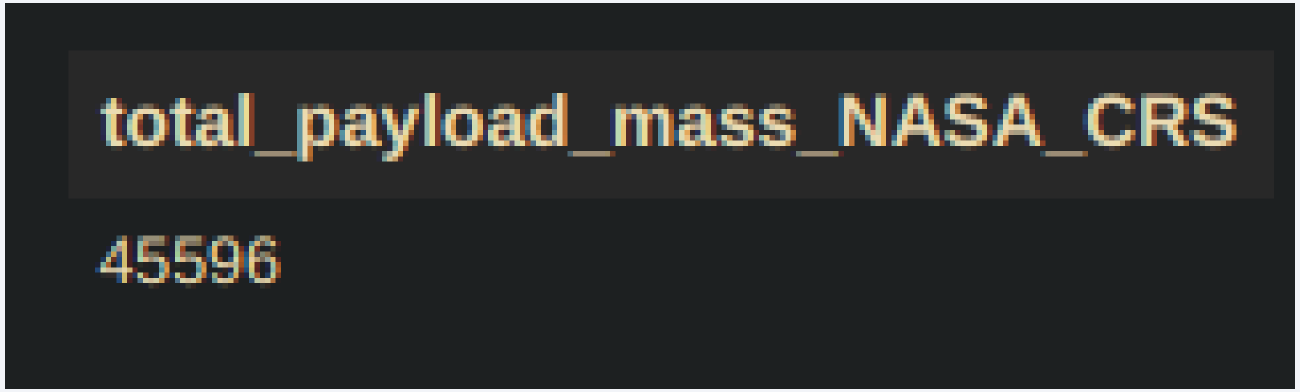
| Date | Time (UTC) | Booster_Version | Launch_Site | Payload | PAYLOAD_MASS__KG_ | Orbit | Customer | Mission_Outcome | Landing_( |
|---|---|---|---|---|---|---|---|---|---|
| 2010-06-04 | 18:45:00 | F9 v1.0 B0003 | CCAFS LC-40 | Dragon Spacecraft Qualification Unit | 0 | LEO | SpaceX | Success | Failure (para |
| 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 (para |
| 2012-05-22 | 7:44:00 | F9 v1.0 B0005 | CCAFS LC-40 | Dragon demo flight C2 | 525 | LEO (ISS) | NASA (COTS) | Success | No attempt |
| 2012-10-08 | 0:35:00 | 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**: %sql select sum(PAYLOAD_MASS__KG_) as total_payload_mass_NASA_CRS from SPACEXTBL where Customer = 'NASA (CRS)'

Sum the payload column where the customer is 'NASA (CRS)' values and renamed it

**RESULT**:



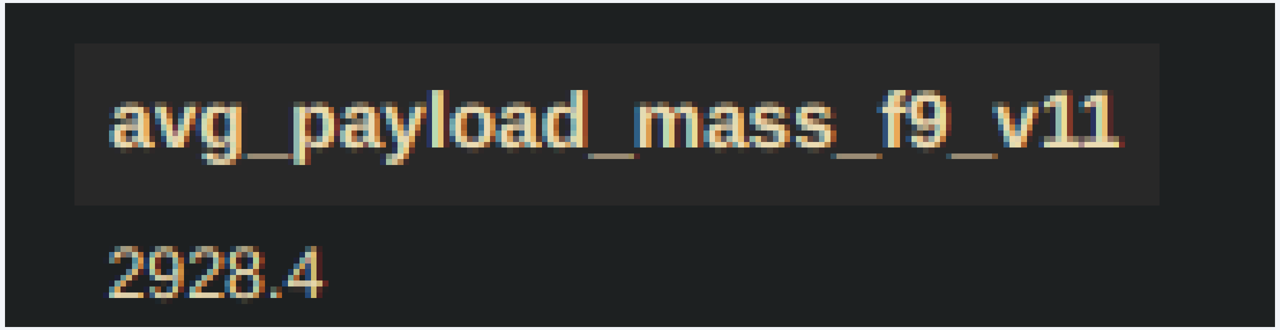
```
total_payload_mass_NASA_CRS
45596
```

# Average Payload Mass by F9 v1.1

**QUERY**: %sql select avg(PAYLOAD_MASS__KG_) as avg_payload_mass_f9_v11 from SPACEXTBL where Booster_Version = 'F9 v1.1'

Average the payload mass column values where the booster version is 'F9 v1.1'

**RESULT**:



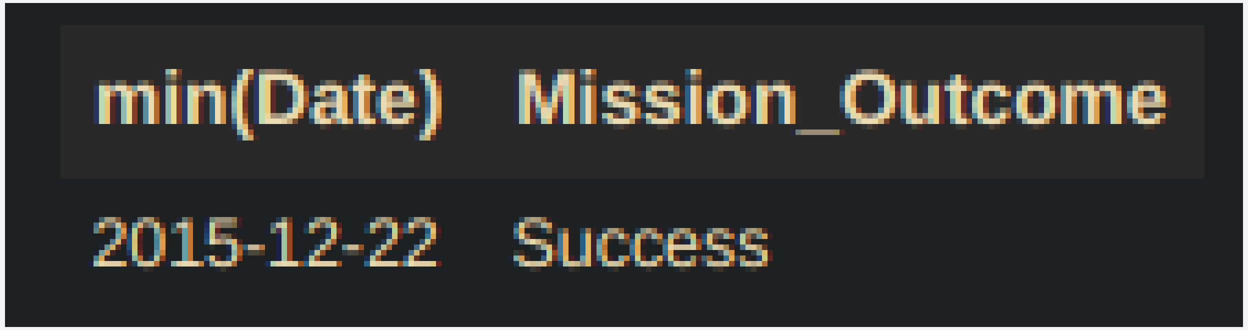
```
avg_payload_mass_f9_v11
2928.4
```

# First Successful Ground Landing Date

**QUERY**: %sql select min(Date),Mission_Outcome from SPACEXTBL where Landing_Outcome = 'Success (ground pad)'

Use the min function to find the minimum date value, to get the first successul landing outcome

**RESULT**:



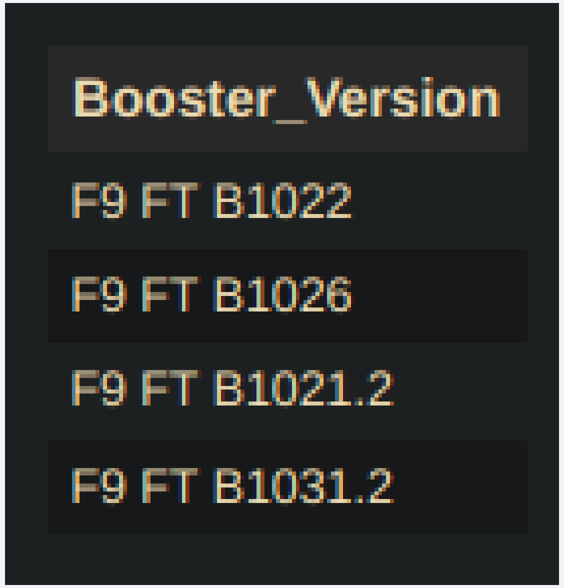
| min(Date) | Mission_Outcome |
|---|---|
| 2015-12-22 | Success |

# Successful Drone Ship Landing with Payload between 4000 and 6000

**QUERY**: %sql select Booster_Version from SPACEXTBL where PAYLOAD_MASS__KG_ between 4000 and 6000 and Landing_Outcome = 'Success (drone ship)'

Select the successful landing boosters that carried a payload between 4000 kg and 6000 kg

**RESULT**:



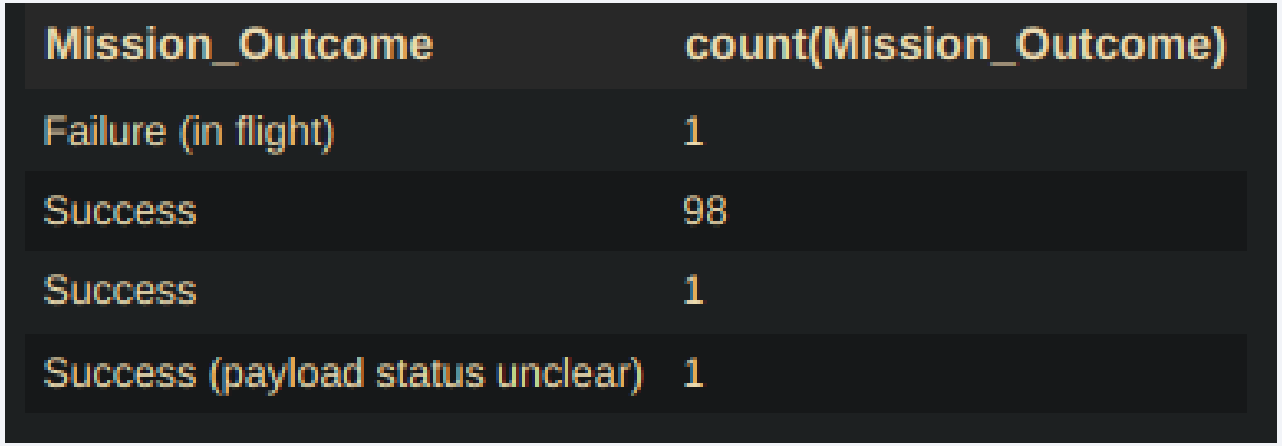
| Booster_Version |
|---|
| F9 FT B1022 |
| F9 FT B1026 |
| F9 FT B1021.2 |
| F9 FT B1031.2 |

# Total Number of Successful and Failure Mission Outcomes

**QUERY**: %sql select Mission_Outcome, count(Mission_Outcome) from SPACEXTBL group by Mission_Outcome

Count the number of missions outcomes

**RESULT**:



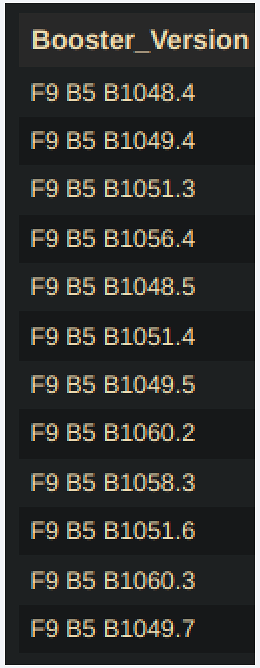
| Mission_Outcome | count(Mission_Outcome) |
|---|---|
| Failure (in flight) | 1 |
| Success | 98 |
| Success | 1 |
| Success (payload status unclear) | 1 |

# Boosters Carried Maximum Payload

**QUERY**: %sql select distinct Booster_Version from SPACEXTBL where PAYLOAD_MASS__KG_ = (select max(PAYLOAD_MASS__KG_) from SPACEXTBL)

Select the boosters that carried the maximum payload

**RESULT**:



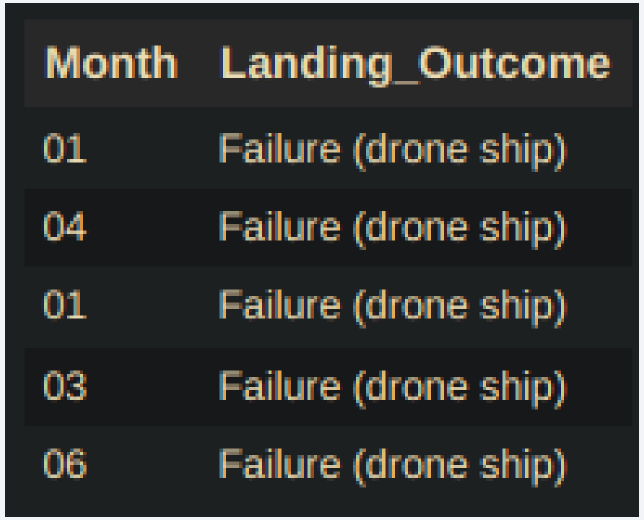
| Booster_Version |
|---|
| F9 B5 B1048.4 |
| F9 B5 B1049.4 |
| F9 B5 B1051.3 |
| F9 B5 B1056.4 |
| F9 B5 B1048.5 |
| F9 B5 B1051.4 |
| F9 B5 B1049.5 |
| F9 B5 B1060.2 |
| F9 B5 B1058.3 |
| F9 B5 B1051.6 |
| F9 B5 B1060.3 |
| F9 B5 B1049.7 |

# 2015 Launch Records

**QUERY**: %sql select substr(Date, 6,2) as Month,Landing_Outcome from SPACEXTBL where Landing_Outcome = 'Failure (drone ship)'

Select the months 2015 of where the mission outcomes were 'Failure (dron ship)'

**RESULT**:



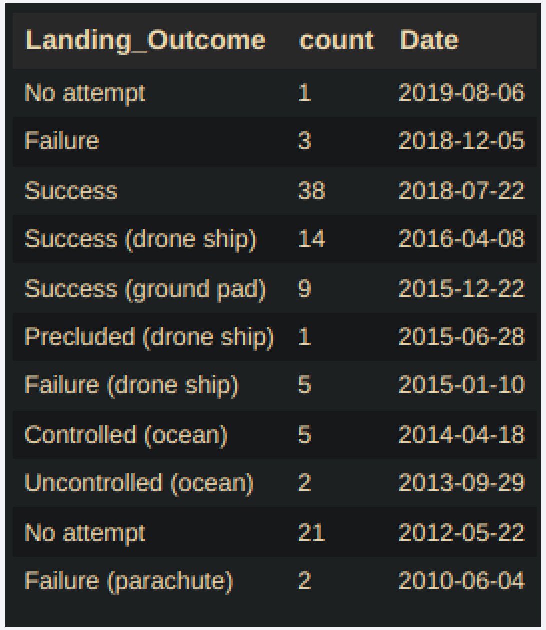
| Month | Landing_Outcome |
|---|---|
| 01 | Failure (drone ship) |
| 04 | Failure (drone ship) |
| 01 | Failure (drone ship) |
| 03 | Failure (drone ship) |
| 06 | Failure (drone ship) |

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

**QUERY**: %sql select Landing_Outcome, count(Landing_Outcome) as count,date from SPACEXTBL group by Landing_Outcome order by date desc

Rank the count of the outcomes by date

**RESULT**:



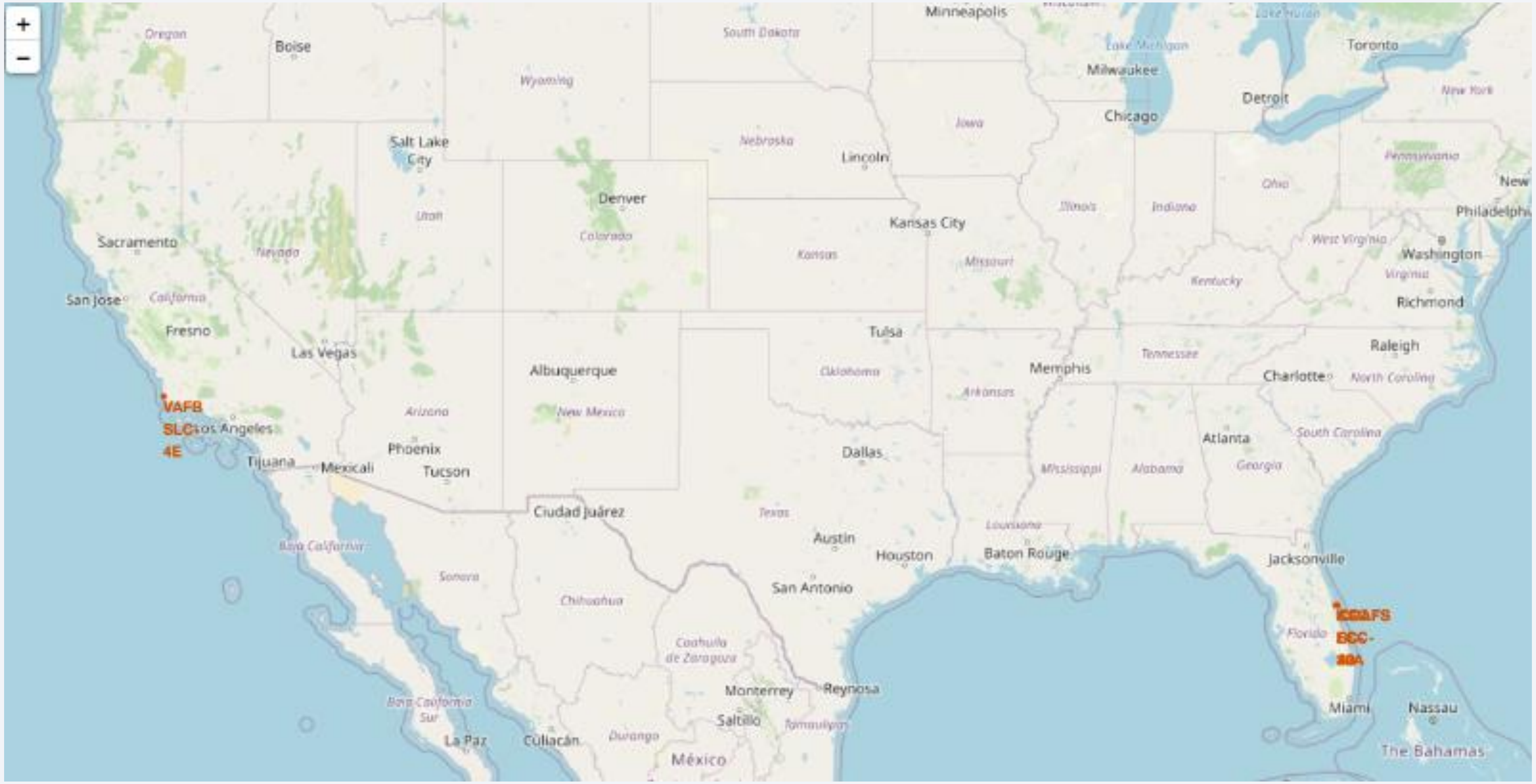
| Landing_Outcome | count | Date |
|---|---|---|
| No attempt | 1 | 2019-08-06 |
| Failure | 3 | 2018-12-05 |
| Success | 38 | 2018-07-22 |
| Success (drone ship) | 14 | 2016-04-08 |
| Success (ground pad) | 9 | 2015-12-22 |
| Precluded (drone ship) | 1 | 2015-06-28 |
| Failure (drone ship) | 5 | 2015-01-10 |
| Controlled (ocean) | 5 | 2014-04-18 |
| Uncontrolled (ocean) | 2 | 2013-09-29 |
| No attempt | 21 | 2012-05-22 |
| Failure (parachute) | 2 | 2010-06-04 |

Section 3

# Launch Sites Proximities Analysis

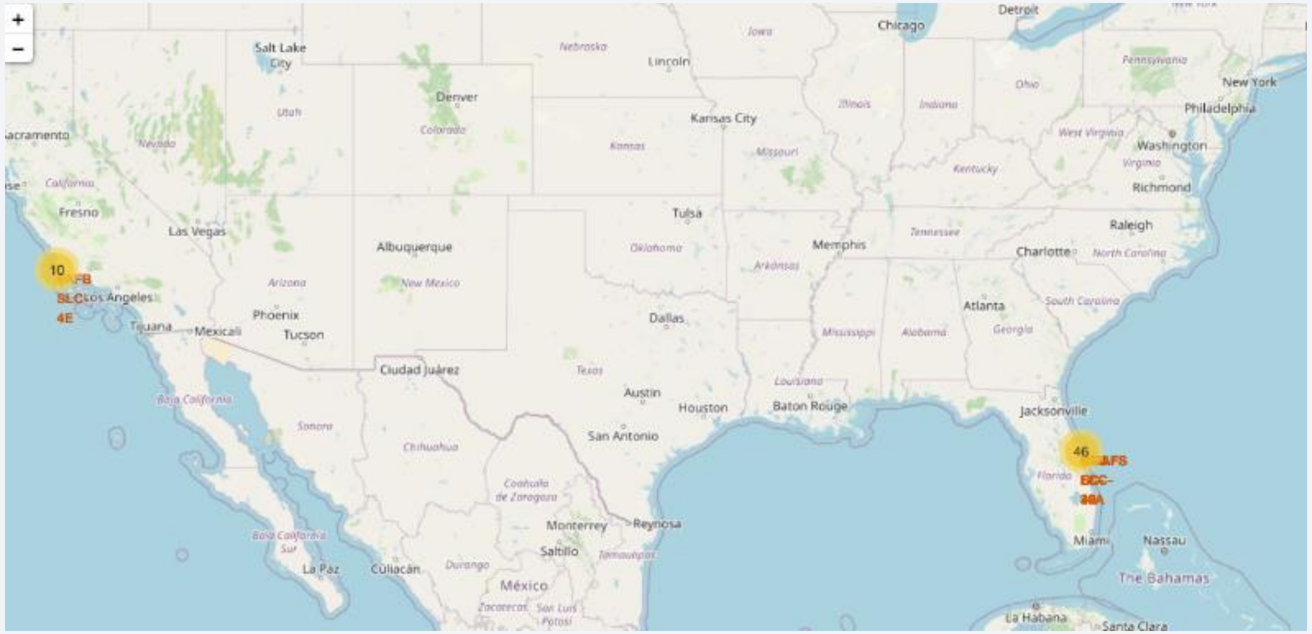
# Launch sites

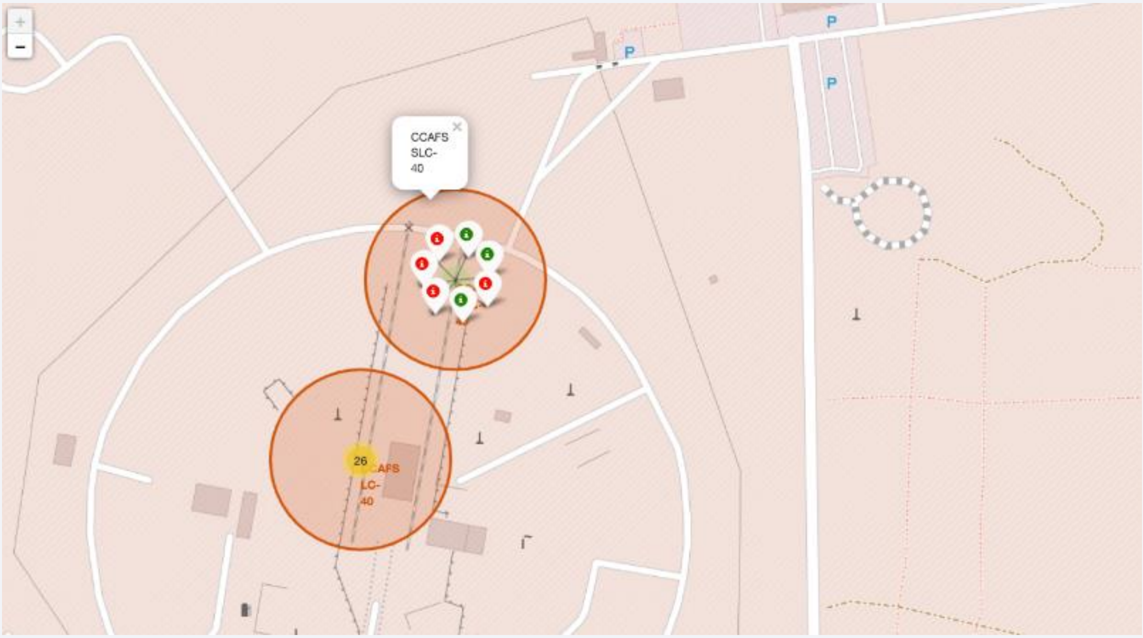
**Map displaying the locations of all launch sites**



# Mission outcomes

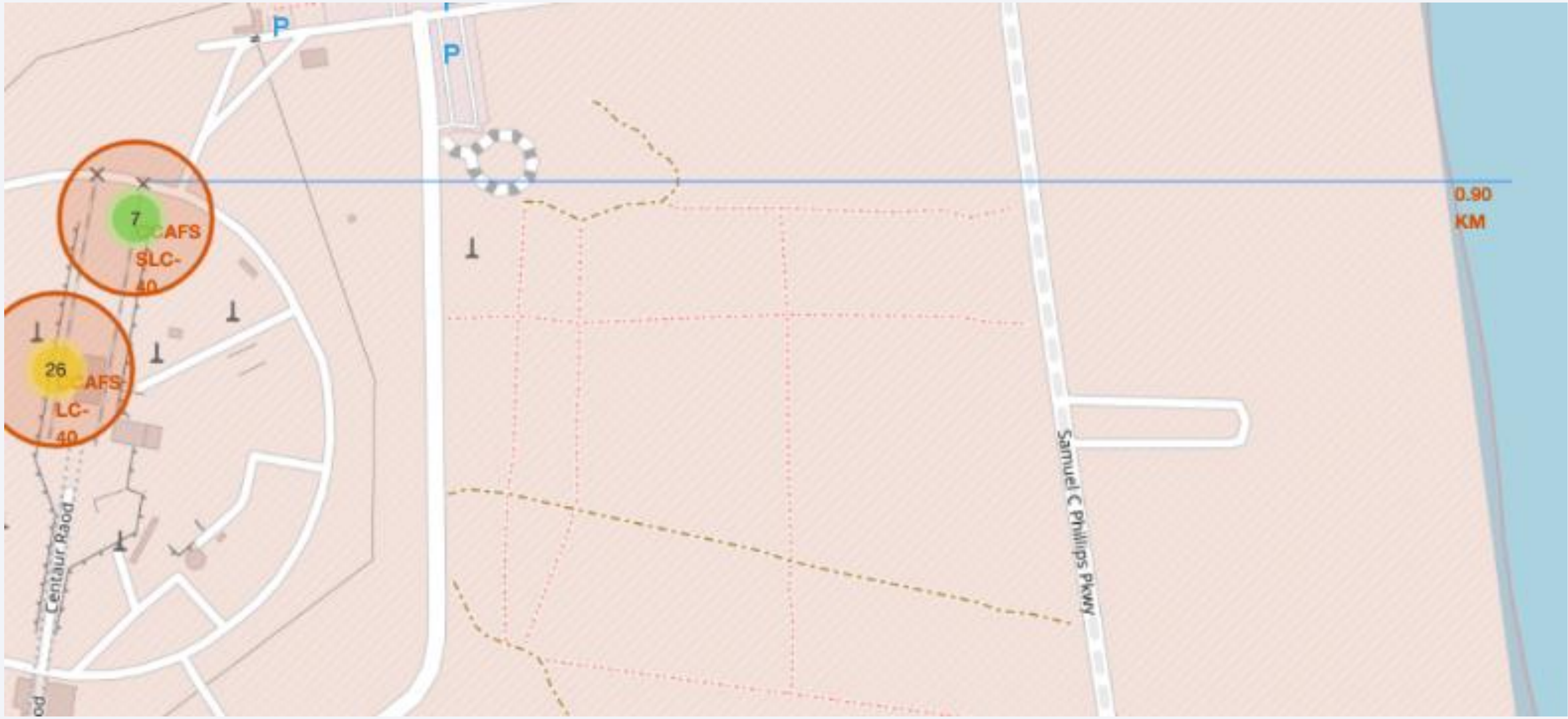
**Landing outcomes of each sites represented on the map**





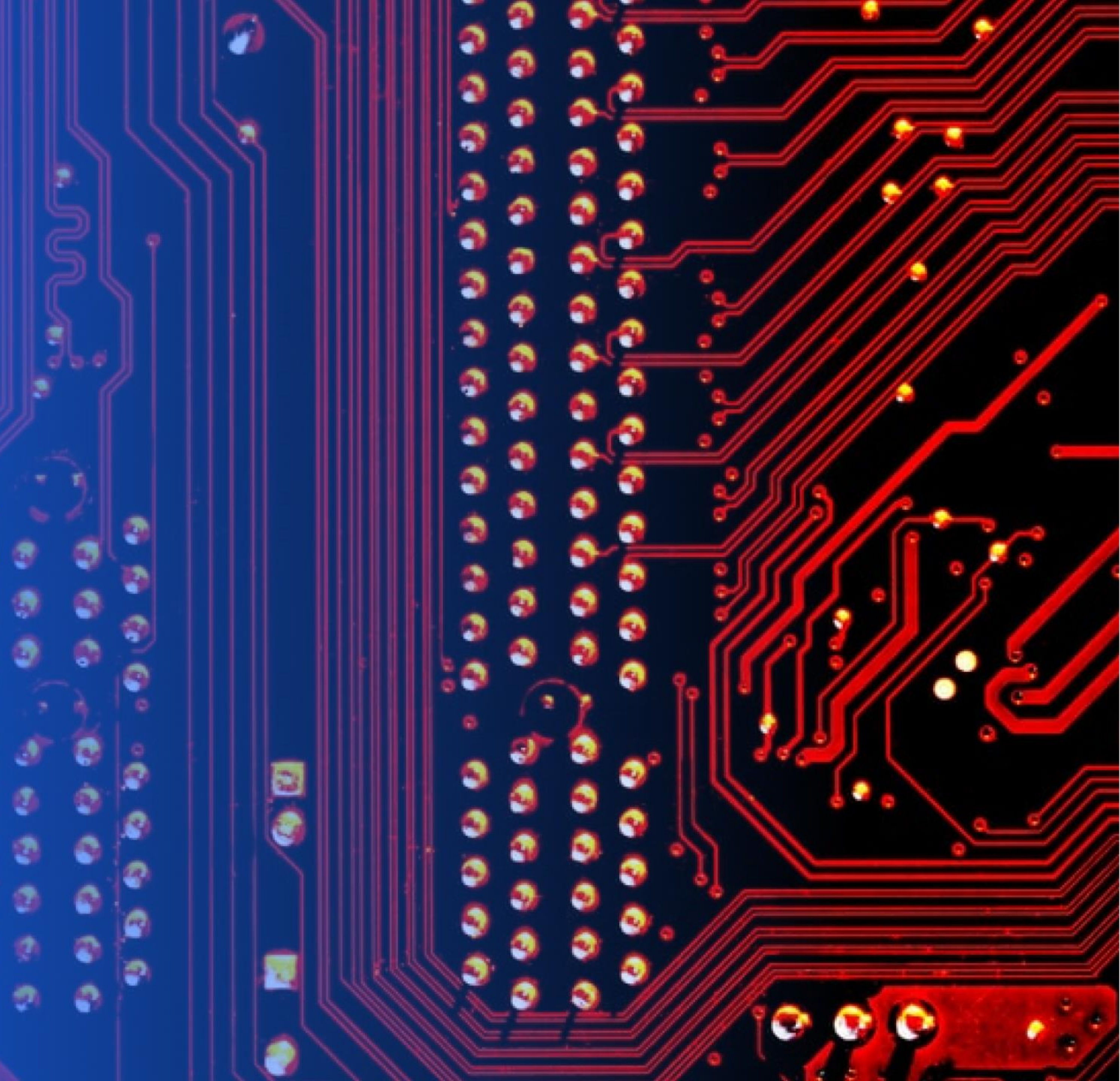
# Distance from sea

**Map showing the distance between launch sites a proximity, this case (the coast)**

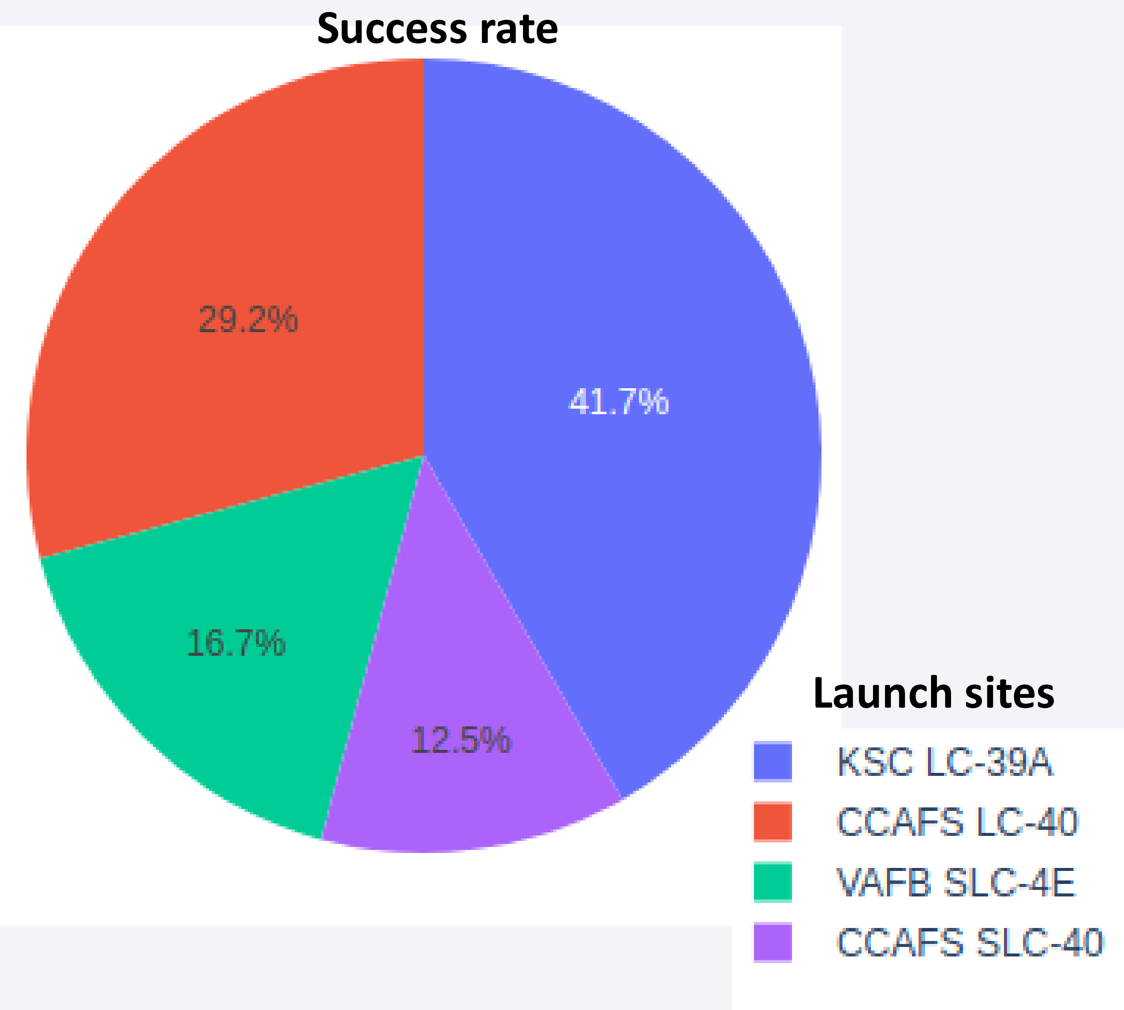


Section 4

# Build a Dashboard with Plotly Dash

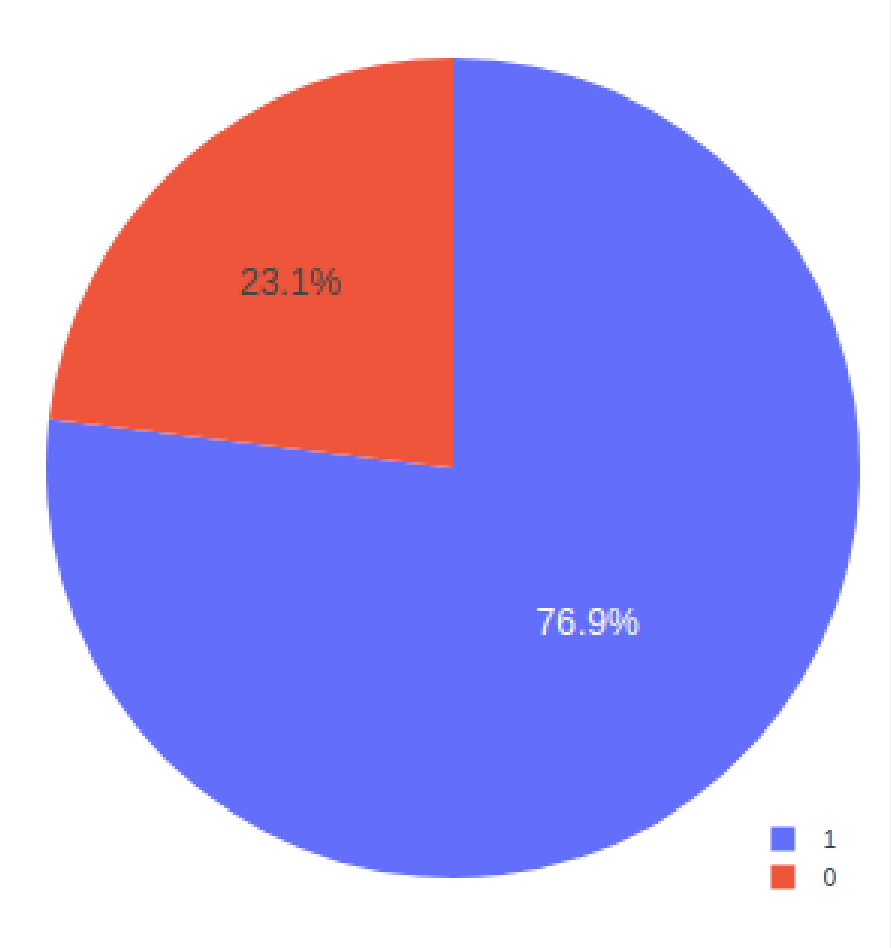


# Launch success for all sites



Pie chart showing the success rates of each launch site. KSC LC-39A has the highest success rate 41.7%, while VAFB SLC-4E has the lowest success rate 12.5 %

# KSC LC-39A Success rate



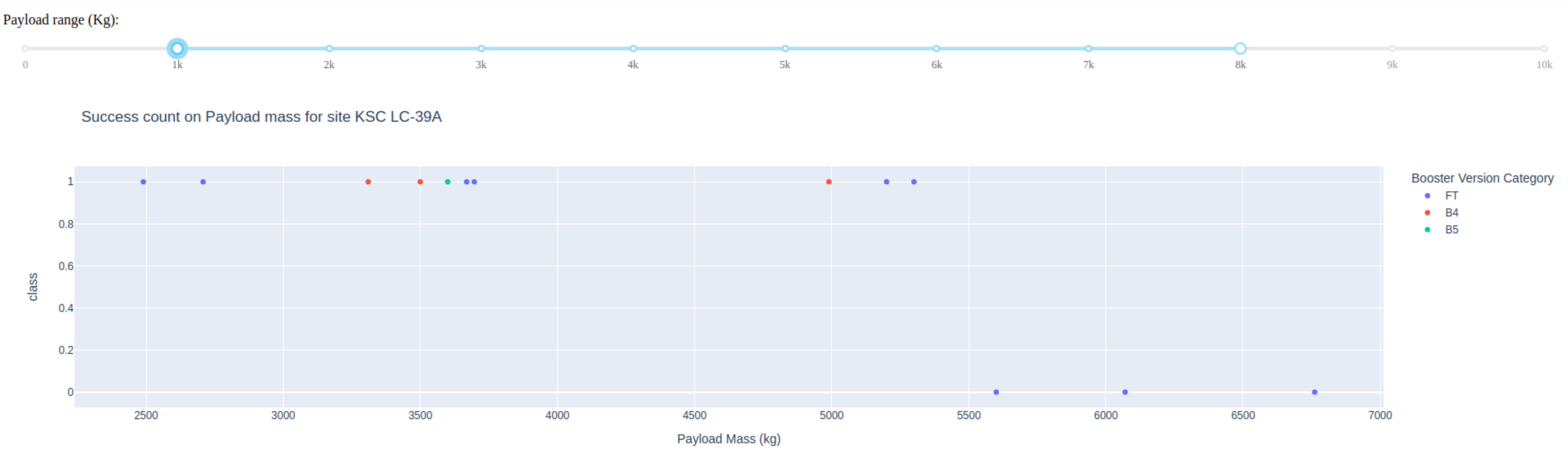
This pie chart shows the success rate of the KSC LC. 39A launching site

' 1 ' Means that the landing outcome was successful

' 0 ' Means that the landing outcme was unsuccessful

# Success rate vs Payload mass for boosters

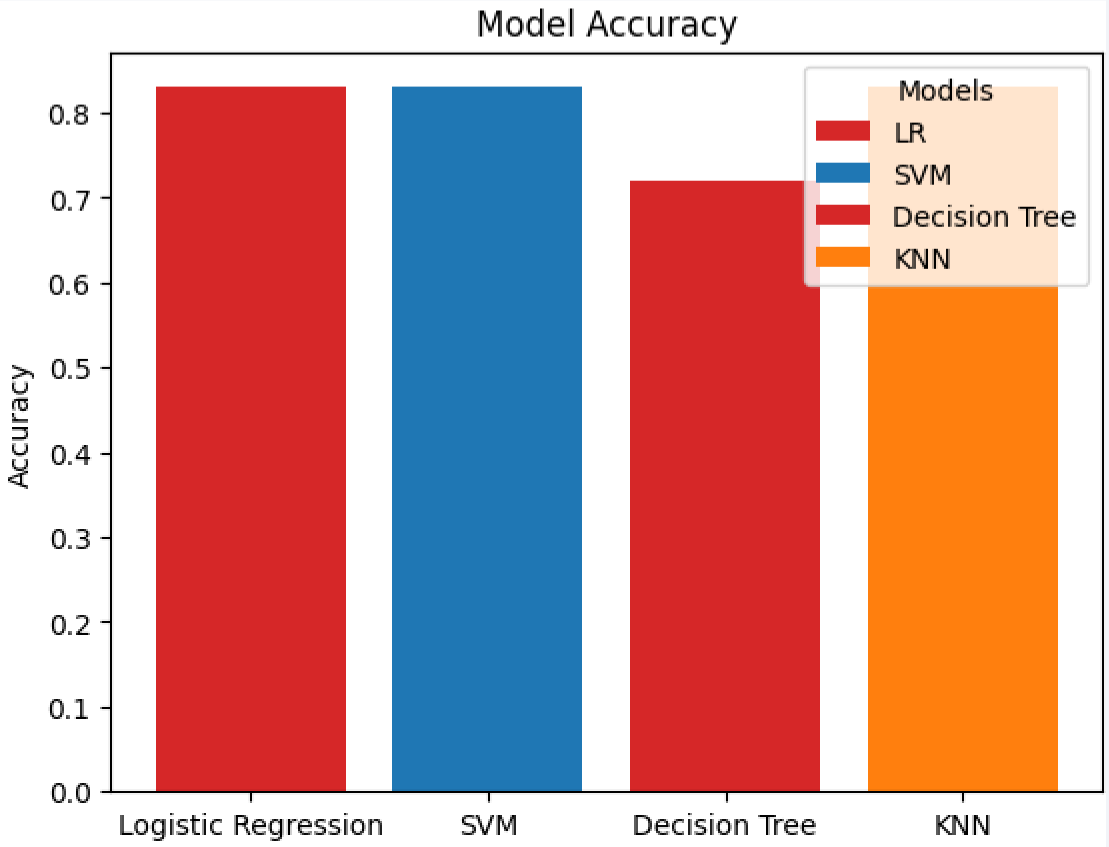
This scatter plot shows the success rate vs different payload masses that can be selected in the bar above the chart.



Section 5

# Predictive Analysis (Classification)

# Classification Accuracy

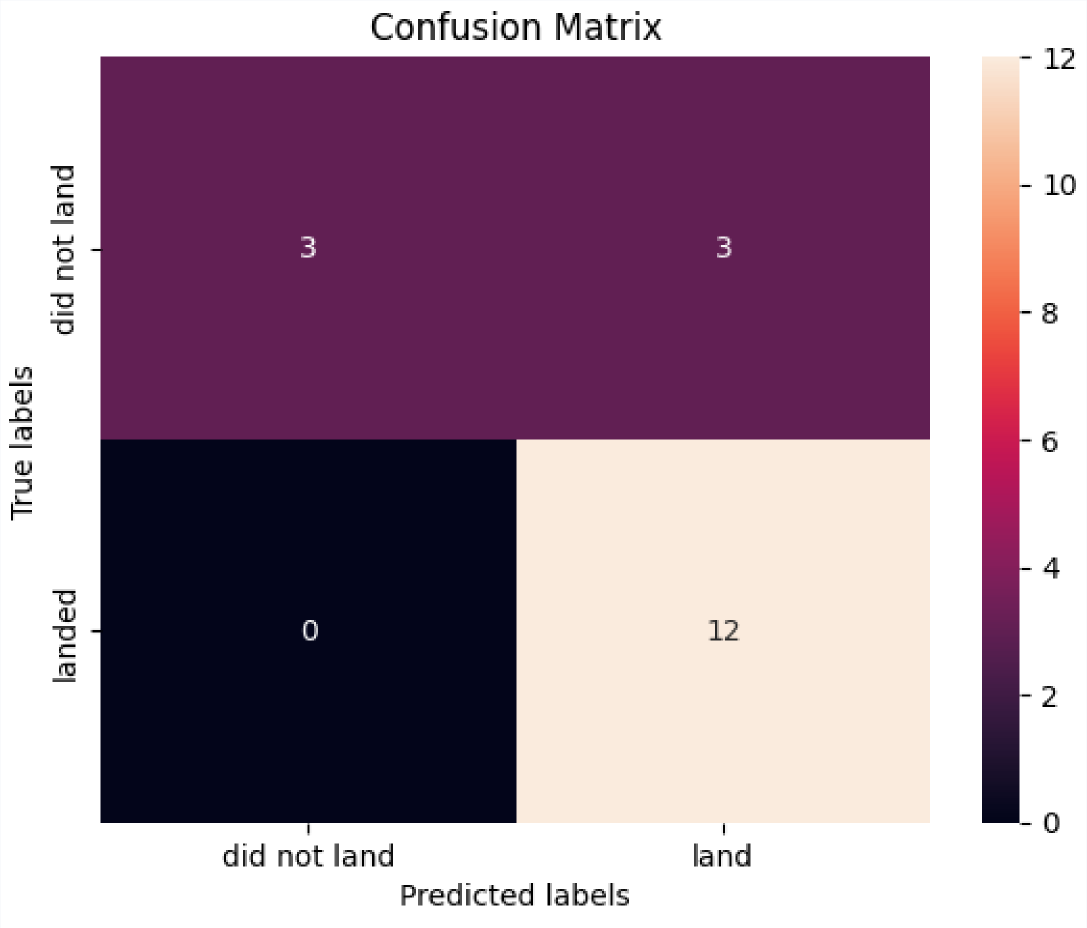


Logistic Regression, Support Vector Machine and K Nearest-Neighbors all yield the same accuracy which was 0.833. Decision Tree was left behind with an accuracy of 0.72

According to these results, the models mentioned above excluding decision trees, are suitable for making predictions on the outcome of a mission

# Confusion Matrix

**Logistic Regression, SVM and KNN confusion matrix**



As explained in the previous slide, those 3 models yielded the same accuracy, and their confusion matrix is the same.

The confusion matrix shows that models did not label incorrectly landed missions as failure, but label incorrectly 3 data points for 'did not land' as 'land'.

The rest of the predictions were done correctly by the 3 models

# Conclusions

- Is important to have data sources available to extract the desire data, as API or web scraping to collect the necessary data for developing the model
- Data wrangling is important as it helps clean and organize the data so the models can perform well. 'Garbage in, Garbage out'
- To explore the data is important, as it reveals important insights and relations between variables and helps decide which variables are important to predict the outcome of a mission
- Machine learning is a powerful tool to make predictions on a data. It saves time, as is not necessary to know about rocket physics to predict the outcome of a mission. The data itself can give us the answer.

Thank you!