
# Winning Space Race with Data Science

<Name>
<Date>

# Outline

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

# Executive Summary

- Data received from the SpaceX API was collected, cleaned, wrangled and used to find the best prediction model for predicting the result of Falcon 9 landing
- The model performing best using test data is Logistic Regression model (R^2=0.89)

# Introduction

- Space X advertises Falcon 9 rocket launches on its website with a cost of 62 million dollars; other providers cost upward of 165 million dollars each, much of the savings is because Space X can reuse the first stage.
- We decided to create a machine learning pipeline to predict if the first stage will land using the SpaceX data about launch sites, landing sites, orbitas, payload mass etc.

Section 1

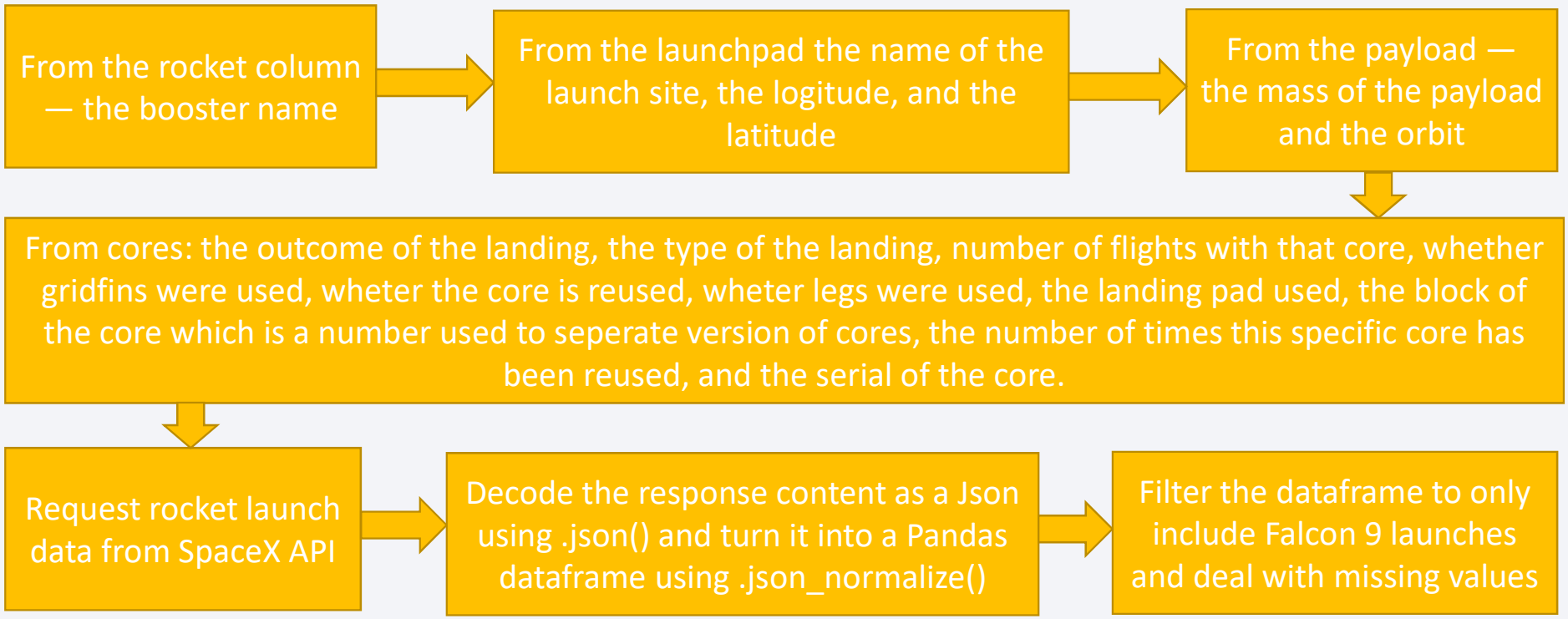
# Methodology

# Methodology

## Executive Summary

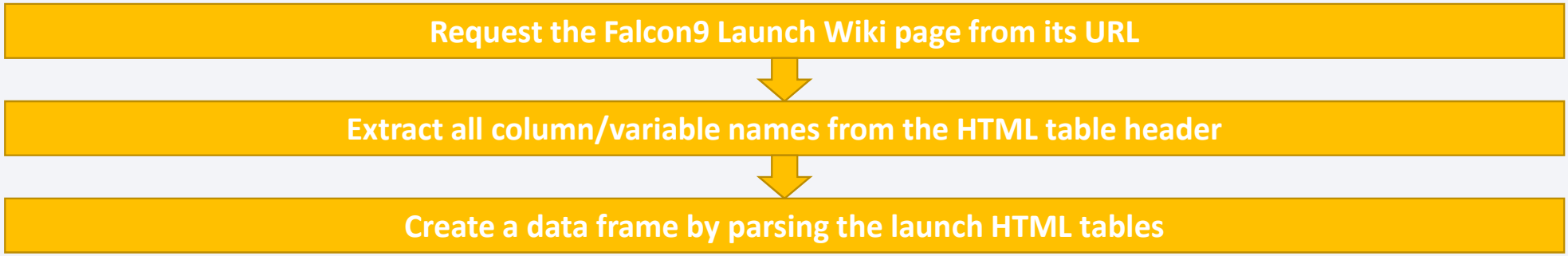
- Data collection methodology:
  - Data was requested to the SpaceX API and webscraped from wikipedia
- Perform data wrangling
  - Class column was added to data with 1 value if landing succeed and 0 if landing failed
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Models were build using pandas, numpy, matplotlib an sklearn libraries; from sklearn, LogisticRegression, SVC, DecisionTree and Kneighbours classification models were used; GridSearchCV was used to find best parameters for them; preprocessing and train_test_split were used to prepare the data for training and testing models

# Data Collection – SpaceX API



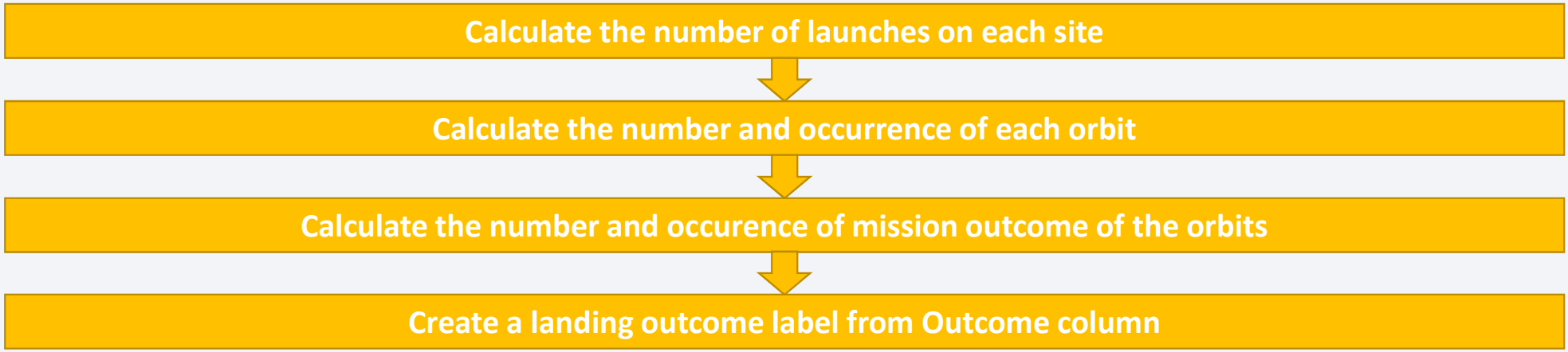

https://github.com/TheArtistFX/Capstone/blob/main/jupyter-labs-spacex-data-collection-api.ipynb

# Data Collection - Scraping



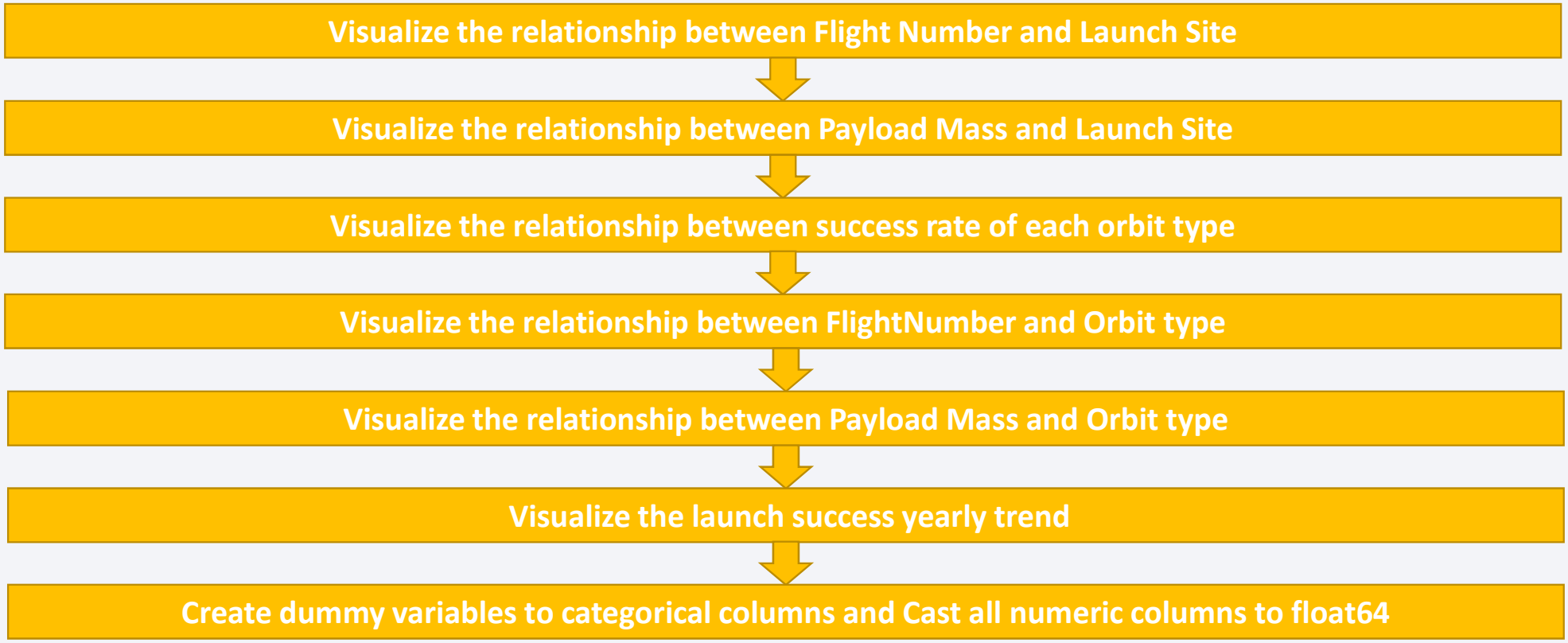

https://github.com/TheArtistFX/Capstone/blob/main/jupyter-labs-webscraping.ipynb

# Data Wrangling




https://github.com/TheArtistFX/Capstone/blob/main/labs-jupyter-spacex-Data%20wrangling.ipynb

# EDA with Data Visualization




https://github.com/TheArtistFX/Capstone/blob/main/edadataviz.ipynb

# EDA with SQL

- 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 succesful landing outcome in ground pad was acheived.
- 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. Use a subquery
- List the records which will display the month names, failure landing_outcomes in drone ship ,booster versions, launch_site for the months in 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.

https://github.com/TheArtistFX/Capstone/blob/main/jupyter-labs-eda-sql-coursera_sqllite.ipynb

# Build an Interactive Map with Folium

- Were marked: all launch sites, the success/failed launches for each site on the map, the distances between all sites with its proximities 9coastline, railroad etc.)
- URL: https://github.com/TheArtistFX/Capstone/blob/main/lab_jupyter_launch_site_location.ipynb

# Build a Dashboard with Plotly Dash

Were added:

- a dropdown list to enable Launch Site selection;
- a pie chart to show the total successful launches count for all sites;
- a slider to select payload range;
- a scatter chart to show the correlation between payload and launch success.

GitHub URL:
https://github.com/TheArtistFX/Capstone/blob/main/spacex_dash_app.py

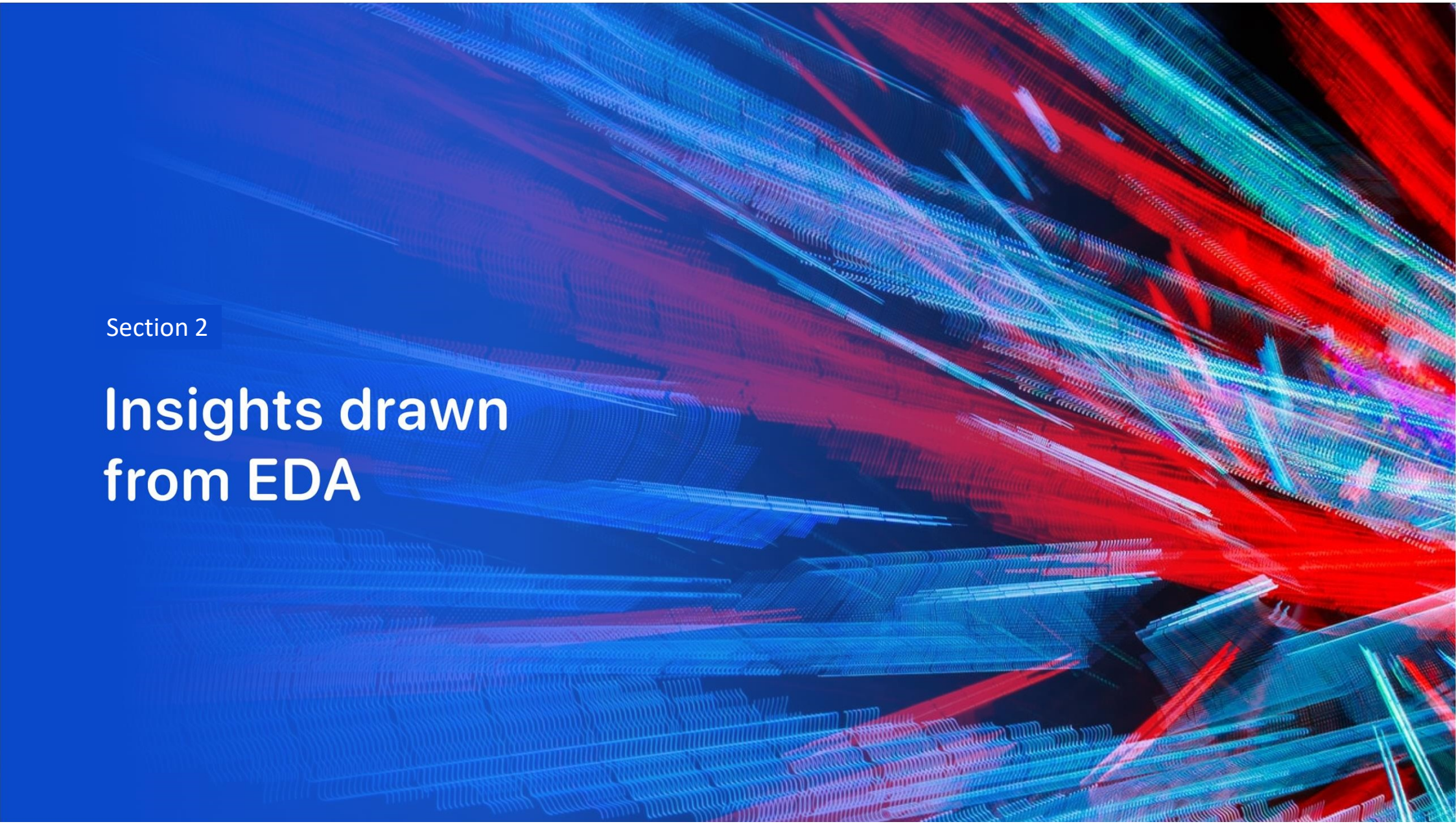
# Predictive Analysis (Classification)

1. Create a NumPy array from the column Class in data, by applying the method to_numpy() then assign it to the variable Y.
2. Standardize the data in X then reassign it to the variable X using the transform provided below.
3. Use the function train_test_split to split the data X and Y into training and test data. Set the parameter test_size to 0.2 and random_state to 2.
4. Create a logistic regression object then create a GridSearchCV object with cv = 10. Fit the object to find the best parameters from the dictionary parameters.
5. Repeat 4 for SVM, Decision tree and K-nearest neighbour.
6. Calculate R^2 score for each model.
7. Find the method performs best.

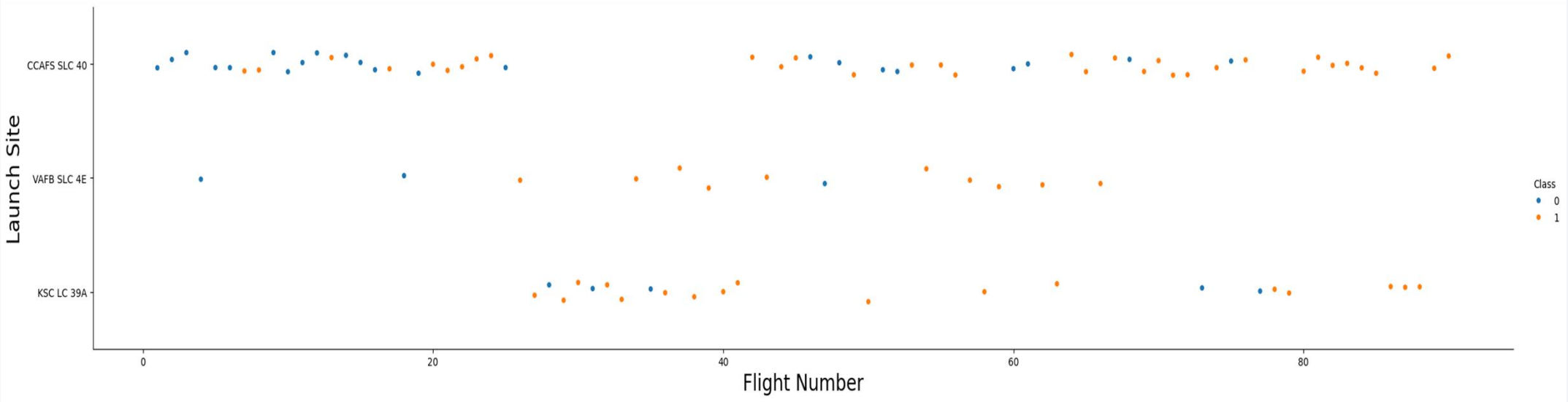
- URL: https://github.com/TheArtistFX/Capstone/blob/main/SpaceX_Machine%20Learning%20Prediction_Part_5.ipynb

# Results

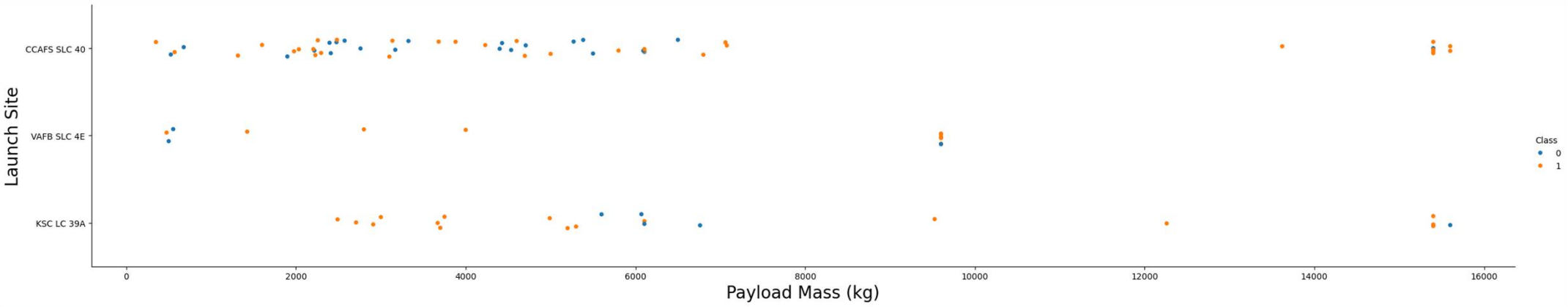
- for the VAFB-SLC launchsite there are no rockets launched for heavy payload mass (greater than 10000);
- ES-L1, GEO, HEO and SSO orbits have 100% success rate;
- in the LEO orbit, success seems to be related to the number of flights. Conversely, in the GTO orbit, there appears to be no relationship between flight number and success;
- with heavy payloads the successful landing or positive landing rate are more for Polar,LEO and ISS;
- the sucess rate since 2013 kept increasing till 2020.
- Linear Regression model performed best to predict landing results


Section 2
Insights drawn from EDA

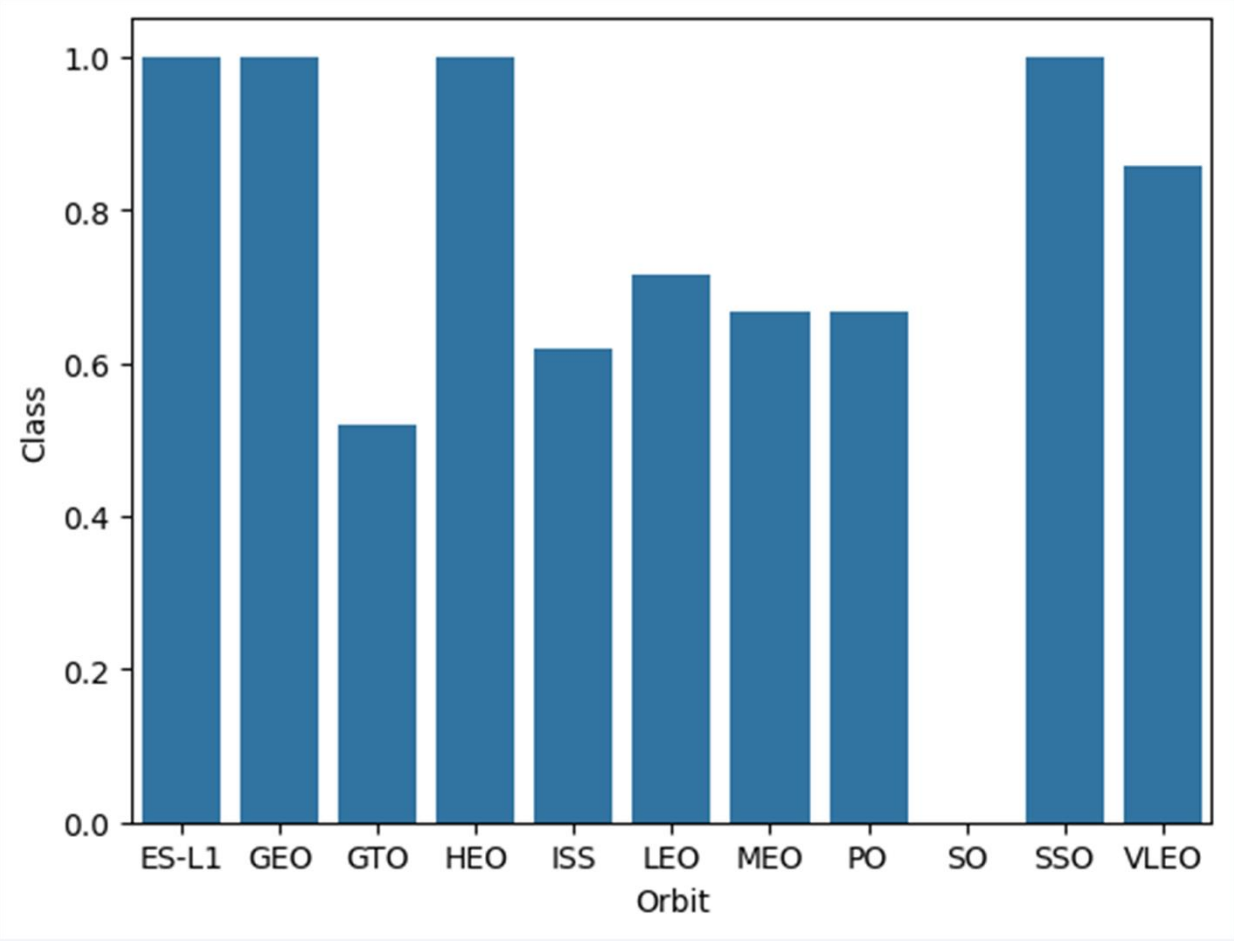
# Flight Number vs. Launch Site



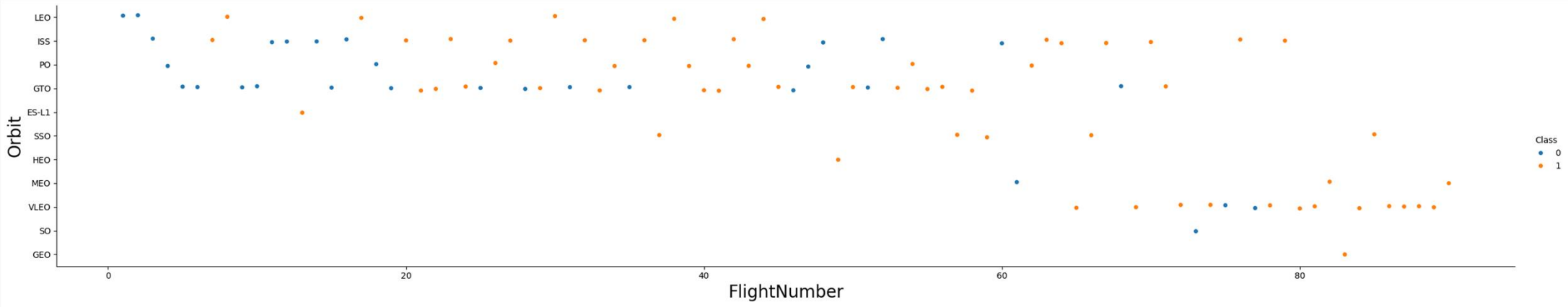
# Payload vs. Launch Site



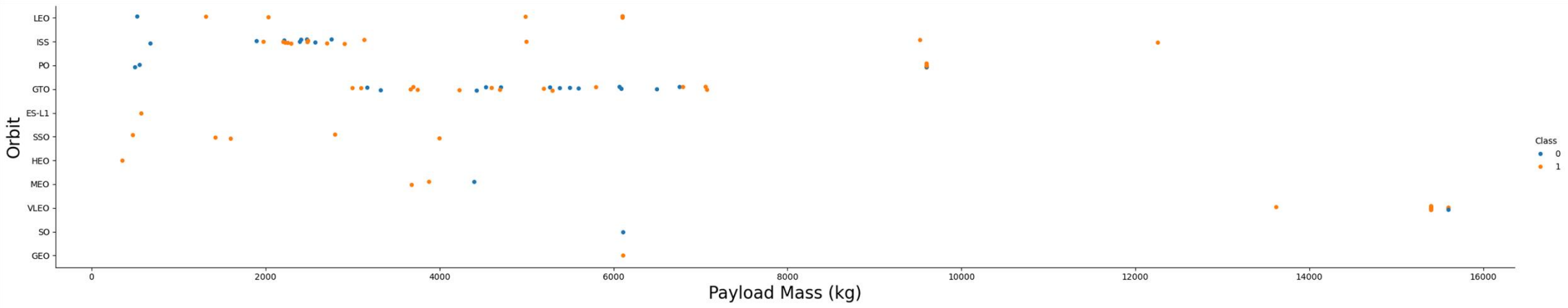
# Success Rate vs. Orbit Type



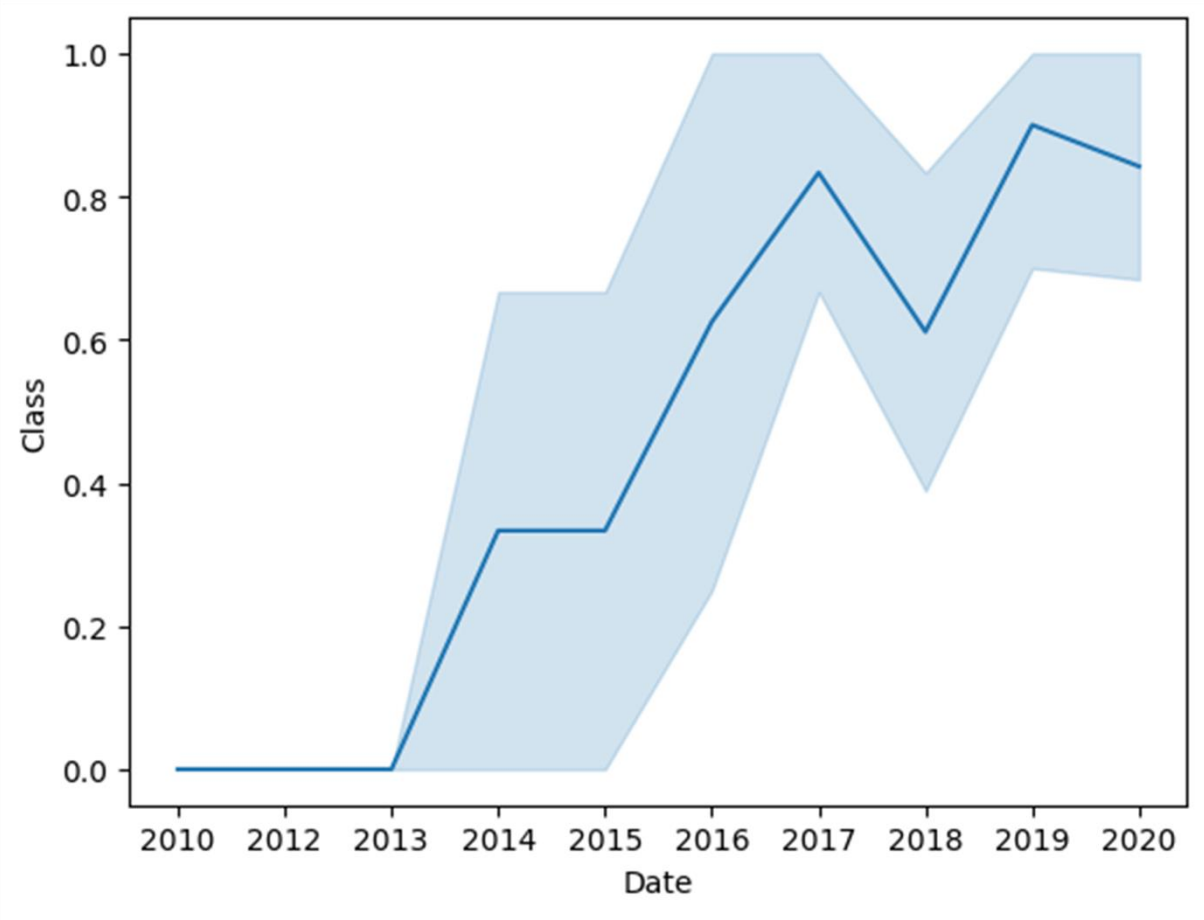
# Flight Number vs. Orbit Type



# Payload vs. Orbit Type



# Launch Success Yearly Trend



# All Launch Site Names

**%sql** SELECT DISTINCT Launch_Site FROM SPACEXTABLE;

Result:

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

# Launch Site Names Begin with 'CCA'

**%sql** SELECT * FROM SPACEXTABLE WHERE Launch_Site LIKE 'CCA%' LIMIT 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 | 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

**%sql** SELECT SUM(Payload_Mass__kg_) AS Total_Payload_Mass FROM SPACEXTABLE WHERE Customer = 'NASA (CRS)';

Result:

| Total_Payload_Mass |
| --- |
| 45596 |

# Average Payload Mass by F9 v1.1

**%sql** SELECT AVG(Payload_Mass__kg_) AS Average_Payload_Mass FROM SPACEXTABLE WHERE Booster_Version = 'F9 v1.1';

Result:

| Average_Payload_Mass |
|---|
| 2928.4 |

# First Successful Ground Landing Date

**%sql** SELECT MIN(Date) AS First_Successful_Landing FROM SPACEXTABLE WHERE Landing_Outcome = 'Success (ground pad)';

Result:

| First_Successful_Landing |
| --- |
| 2015-12-22 |

## Successful Drone Ship Landing with Payload between 4000 and 6000

**%sql** SELECT Booster_Version FROM SPACEXTABLE WHERE Landing_Outcome = 'Success (drone ship)' AND Payload_Mass__kg_ > 4000 AND Payload_Mass__kg_ < 6000;

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

**%sql** SELECT Landing_Outcome, COUNT(*) AS Outcome_Count FROM SPACEXTABLE GROUP BY Landing_Outcome;

Result:

| Landing_Outcome | Outcome_Count |
|---|---|
| Controlled (ocean) | 5 |
| Failure | 3 |
| Failure (drone ship) | 5 |
| Failure (parachute) | 2 |
| No attempt | 21 |
| No attempt | 1 |
| Precluded (drone ship) | 1 |
| Success | 38 |
| Success (drone ship) | 14 |
| Success (ground pad) | 9 |
| Uncontrolled (ocean) | 2 |

# Boosters Carried Maximum Payload

**%sql** SELECT Booster_Version FROM SPACEXTABLE WHERE Payload_Mass__kg_ = (SELECT MAX(Payload_Mass__kg_) FROM SPACEXTABLE);

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

**%sql** SELECT substr(Date, 6, 2) AS Month, Booster_Version, Launch_Site, Landing_Outcome FROM SPACEXTABLE WHERE substr(Date, 1, 4) = '2015' AND Landing_Outcome = 'Failure (drone ship)';

Result:

| Month | Booster_Version | Launch_Site | Landing_Outcome |
|---|---|---|---|
| 01 | F9 v1.1 B1012 | CCAFS LC-40 | Failure (drone ship) |
| 04 | F9 v1.1 B1015 | CCAFS LC-40 | Failure (drone ship) |

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

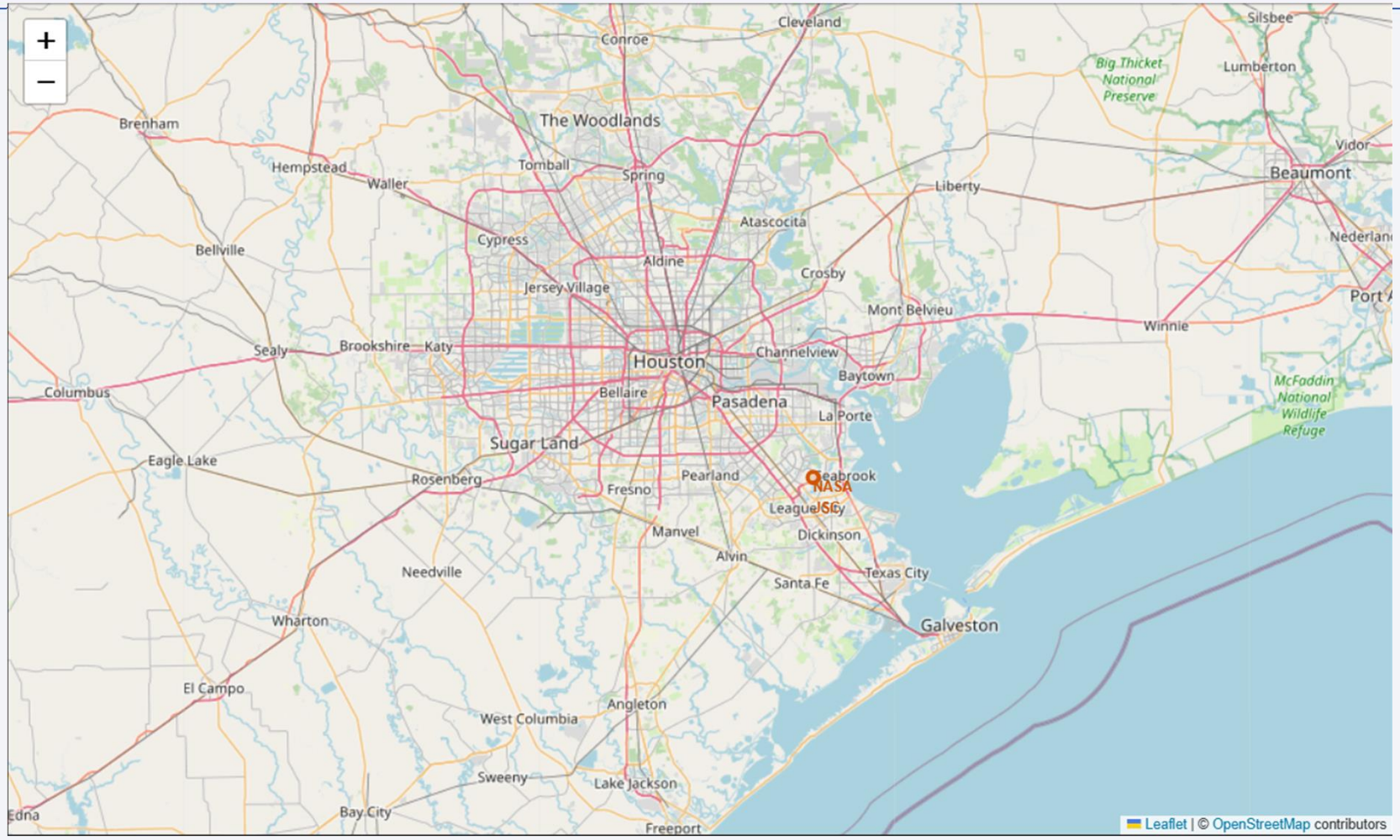
**%sql** SELECT Landing_Outcome, COUNT(*) AS Outcome_Count FROM SPACEXTABLE WHERE Date BETWEEN '2010-06-04' AND '2017-03-20' GROUP BY Landing_Outcome ORDER BY Outcome_Count DESC;

Result:

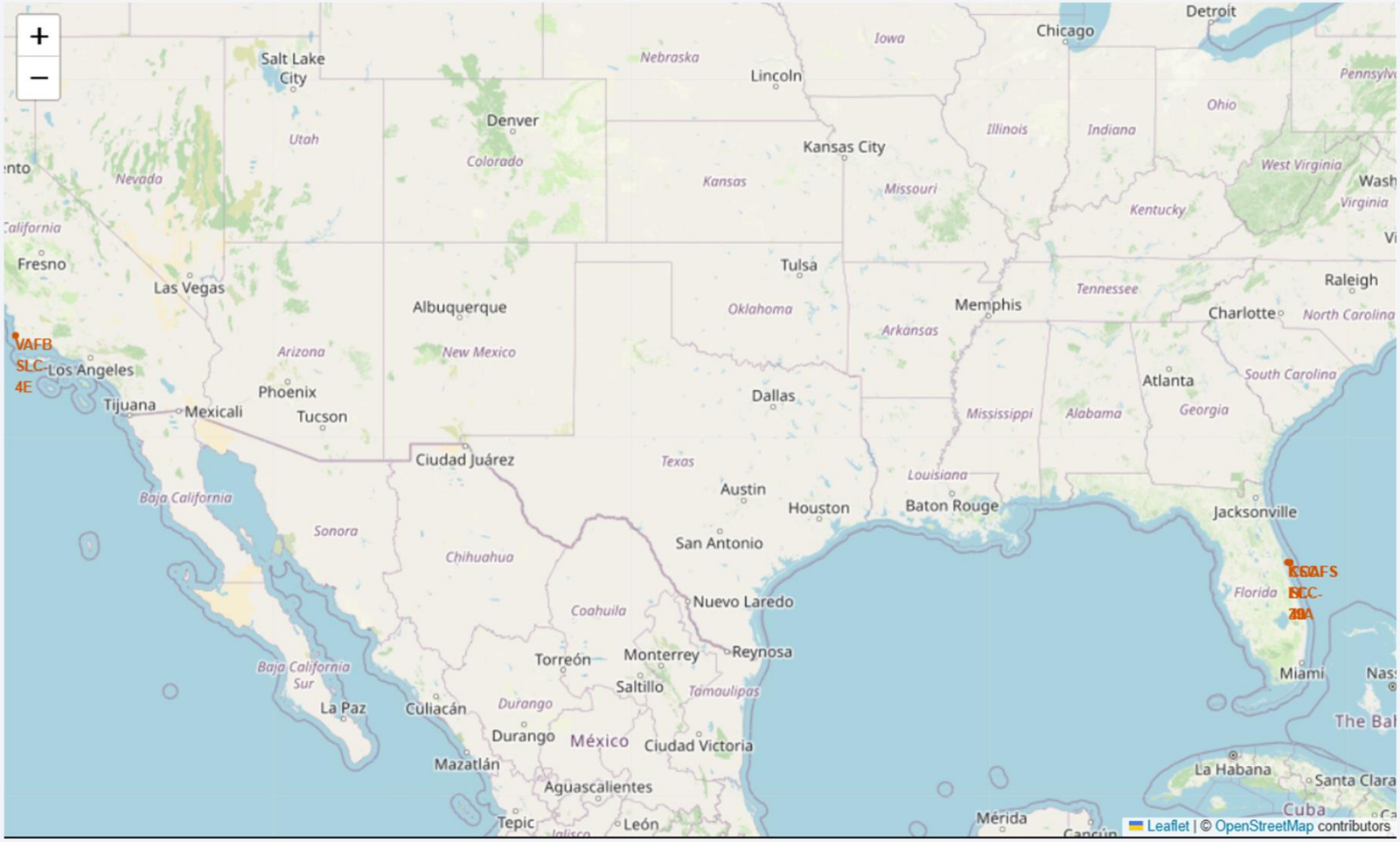
| Landing_Outcome | Outcome_Count |
|---|---|
| No attempt | 10 |
| Success (drone ship) | 5 |
| Failure (drone ship) | 5 |
| Success (ground pad) | 3 |
| Controlled (ocean) | 3 |
| Uncontrolled (ocean) | 2 |
| Failure (parachute) | 2 |
| Precluded (drone ship) | 1 |


Section 3
Launch Sites
Proximities Analysis

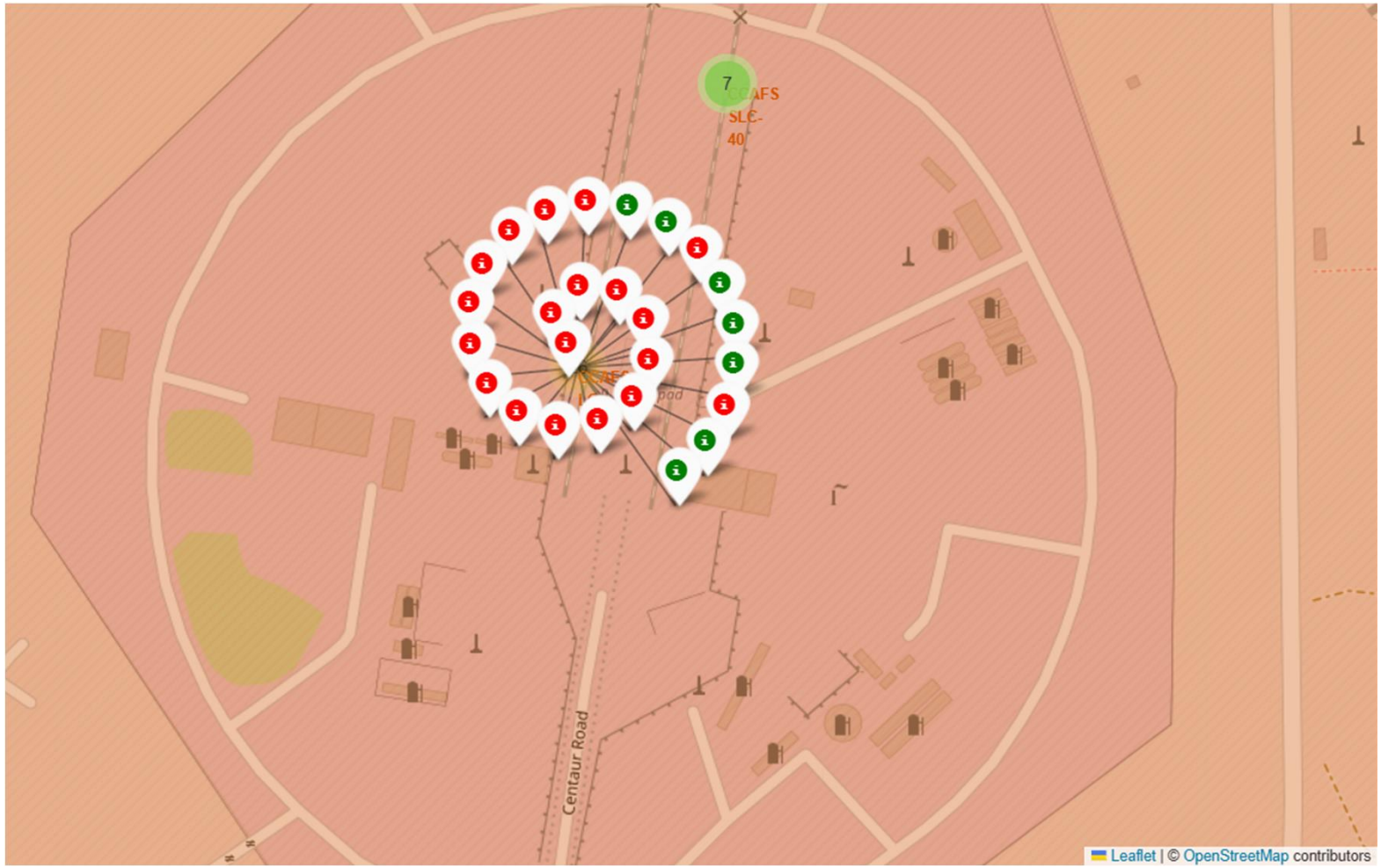
# NASA coordinates marked on map



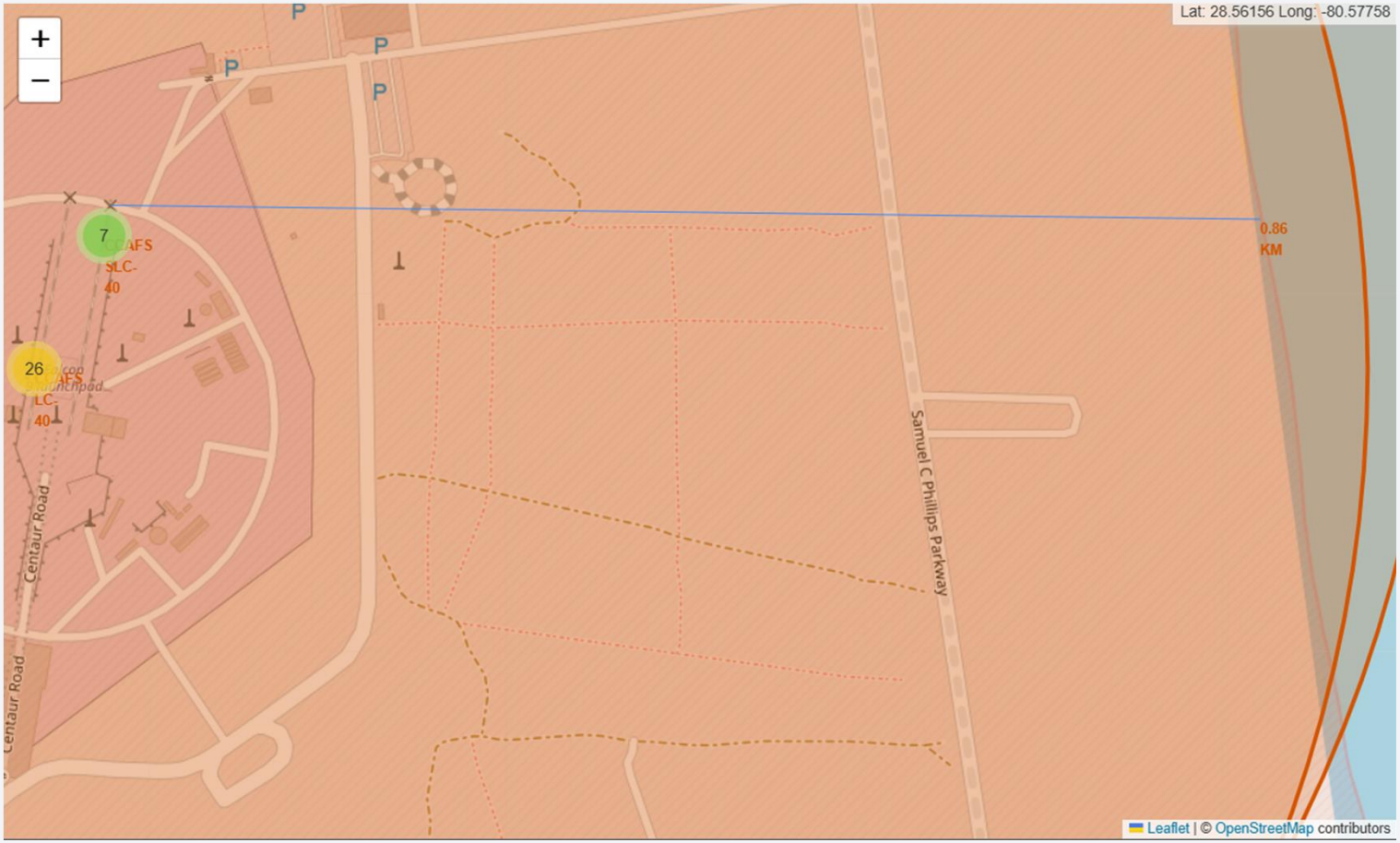
# All launch sites on a map

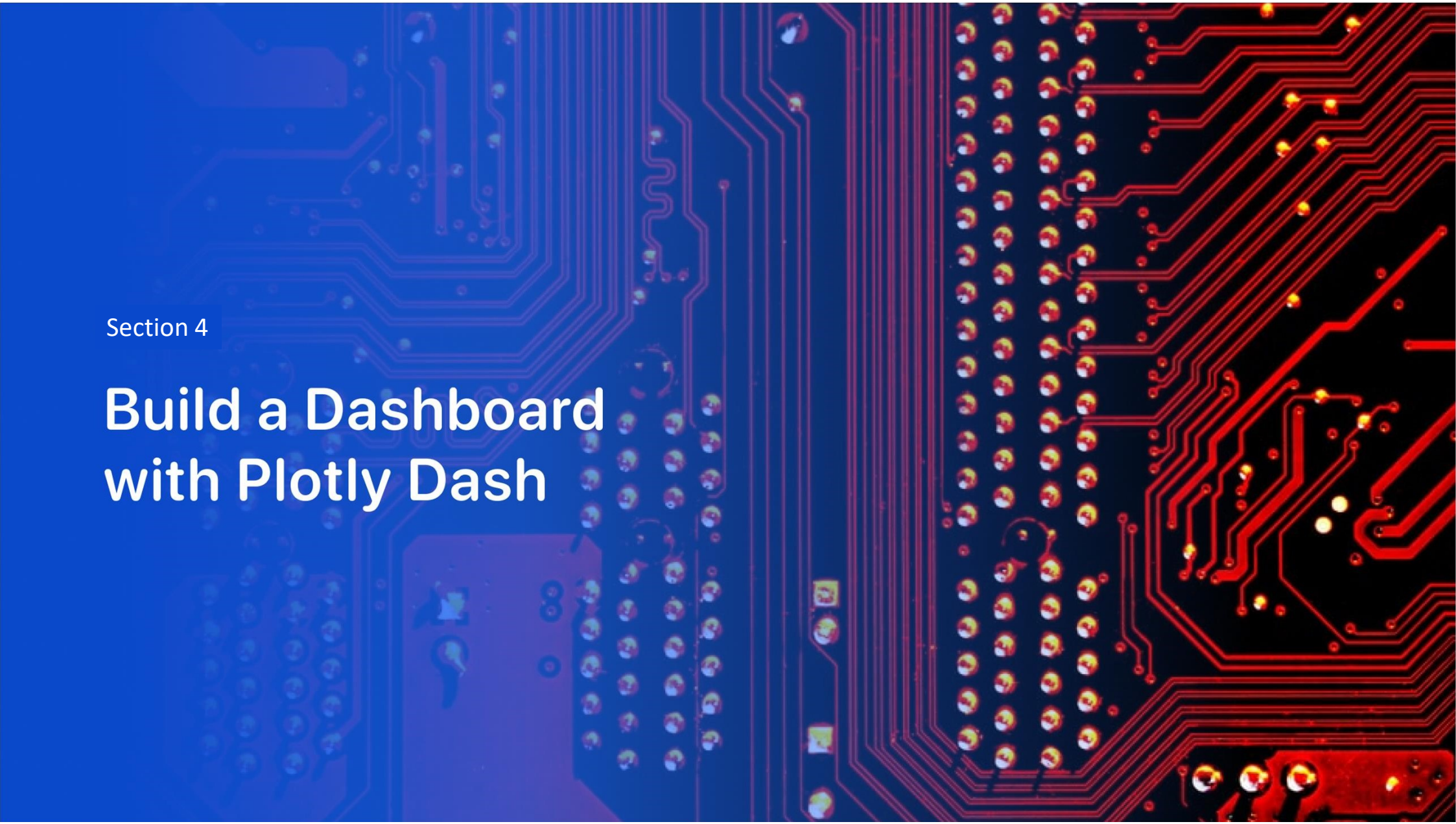


# Successful/Failed landings



# Distance from the coastline




Section 4
Build a Dashboard with Plotly Dash

# <Dashboard Screenshot 1>

- Replace <Dashboard screenshot 1> title with an appropriate title
- Show the screenshot of launch success count for all sites, in a piechart
- Explain the important elements and findings on the screenshot

# <Dashboard Screenshot 2>

- Replace <Dashboard screenshot 2> title with an appropriate title
- Show the screenshot of the piechart for the launch site with highest launch success ratio
- Explain the important elements and findings on the screenshot

# <Dashboard Screenshot 3>

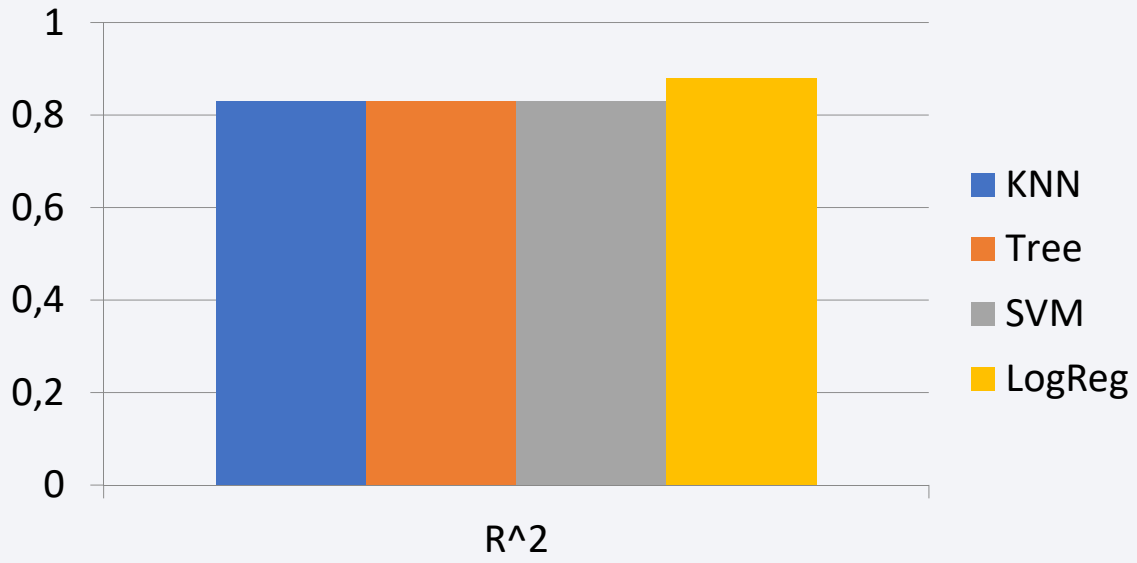
- Replace <Dashboard screenshot 3> title with an appropriate title
- Show screenshots of Payload vs. Launch Outcome scatter plot for all sites, with different payload selected in the range slider
- Explain the important elements and findings on the screenshot, such as which payload range or booster version have the largest success rate, etc.

Section 5

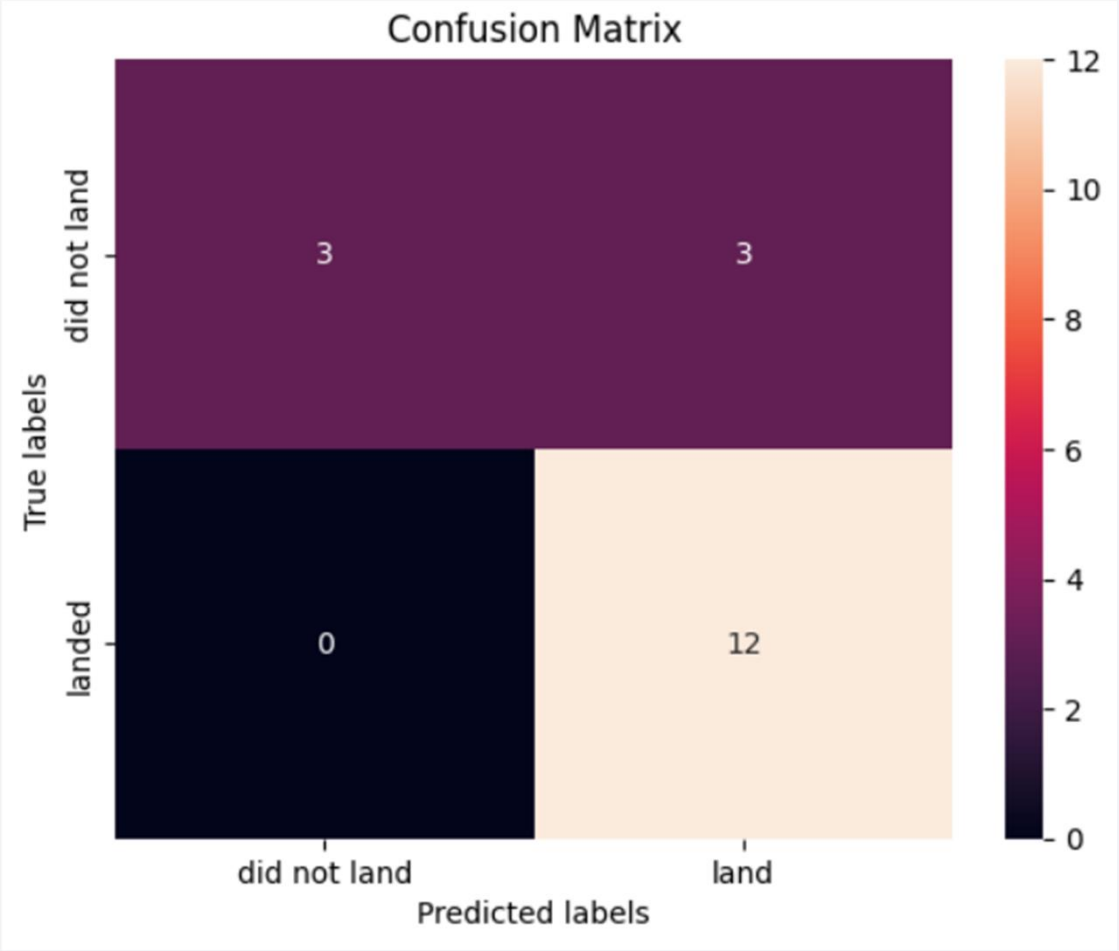
# Predictive Analysis (Classification)

# Classification Accuracy

- KNN has 0.8333333333333334 accuracy
- Decision tree has 0.8333333333333334 accuracy
- SVM has 0.8333333333333334 accuracy
- Logistic regression has **0.8888888888888888 accuracy**



# Confusion Matrix



# Conclusions

- Built LogReg model may be used to predict landing results
- Some orbits have 100% success rate, so it would be a good idea to use only them until modifying rockets
- From 2013 success rate increased, it means that NASA works to the right direction and we can build another model to predict the time when we will reach, for example, 90% success rate on landing
- Models have comparable accuracy, so it needs more data or new algorithmes to modify them and choose the best

# Thank you!