Data Science Capstone Project

SPACEX Launch Data

Isabela T.

27 March 2024

Outline

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

Executive Summary

- Summary of methodologies
  - Data collection
  - Data wrangling
  - Exploratory data analysis (EDA)
  - Data visualization
  - Predictive analysis
- Summary of all results
  - EDA results
  - Interactive analysis results
  - Predictive analysis results

Introduction

- **Space Exploration Technologies Corporation (SpaceX)** is a company that designs, manufactures and launches advanced rockets and spacecraft
- SpaceX advertises its **Falcon 9 reusable, two-stage rocket** launches on its website with a cost of 62 million dollars, while other providers cost upward of 165 million dollars each
- If we can determine if the first stage of Falcon 9 will land, we can determine the cost of a launch
- This is a relevant information for a company that wants to bid against SpaceX for a rocket launch

- **Objective: predict the landing outcome of the first stage of Falcon 9 in the future**
- **Questions:**
  - **What factors influence the landing outcome?**
  - **Which set of conditions would ensure mission success?**

Section 1

Methodology

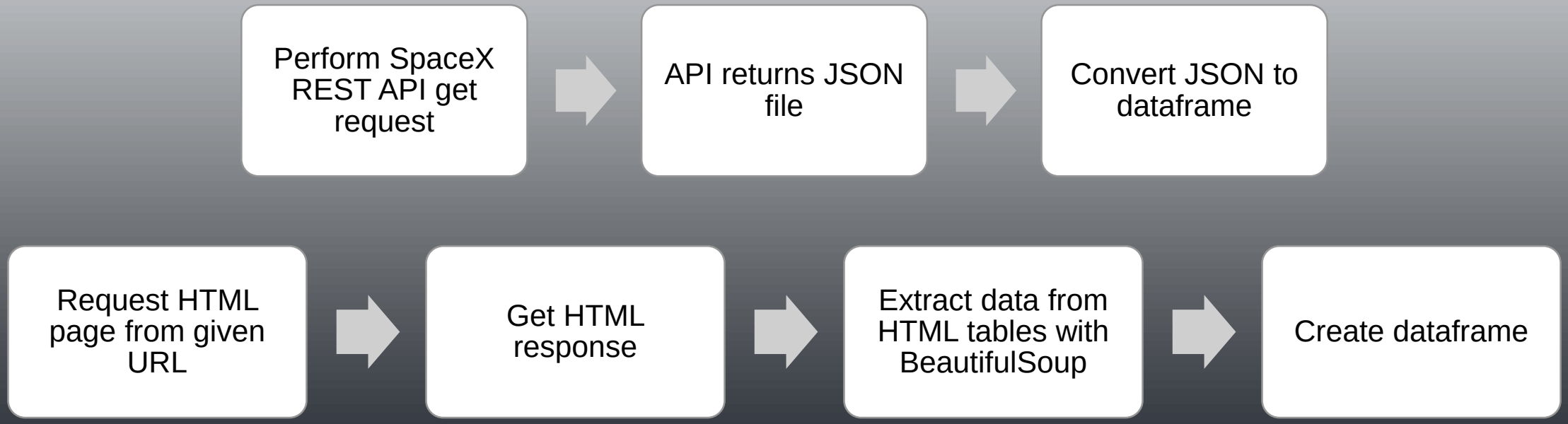
Methodology

1. Perform data collection and data wrangling using:
   - SpaceX REST API
   - Web scraping (BeautifulSoup)
2. Perform EDA using:
   - SQL
   - Pandas and NumPy
   - Visualization tools (Matplotlib and Seaborn)
3. Perform interactive visual analytics using:
   - Folium
   - Plotly Dash
4. Perform predictive analysis using:
   - classification models (logistic regression, support vector machine, decision tree, and k-nearest neighbors)

Data Collection

- SpaceX launch data were collected using the **SpaceX REST API** and Python package **BeautifulSoup** (which was used to web scrape HTML tables that contain Falcon 9 launch records)

SpaceX API endpoint: https://api.spacexdata.com/v4/launches/past
Falcon 9 launch data URL: https://en.wikipedia.org/w/index.php?title=List_of_Falcon_9_and_Falcon_Heavy_launches&oldid=1027686922



Data Collection – SpaceX API

1. Request SpaceX launch data using the GET request

```
spacex_url = "https://api.spacexdata.com/v4/launches/past"
```

```
response = requests.get(spacex_url)
```

2. Decode response content as a JSON and turn it into a dataframe

```
data = pd.json_normalize(response.json())
```

3. Perform data cleaning and replace missing values
    a) Filter the dataframe to include only Falcon 9 launches

```
data_falcon9 = data[data['BoosterVersion'] != 'Falcon 1']
```

    b) Fill the missing values

```
# Calculate the mean value of PayloadMass column
avg_payloadmass = data_falcon9['PayloadMass'].astype('float').mean(axis=0)
# Replace the np.nan values with its mean value
data_falcon9['PayloadMass'].replace(np.nan, avg_payloadmass, inplace=True)
```

GitHub URL: https://github.com/elathr/AppliedDataScienceCapstone/blob/main/Data_Collection_SpaceX_API.ipynb

Data Collection – Web Scraping

1. Request Falcon 9 Wiki page using the HTTP GET method

```
static_url = "https://en.wikipedia.org/w/index.php?title=List_of_Falcon_9_and_Falcon_Heavy_launches&oldid=1027686922"
```

```
data = requests.get(static_url).text
```

2. Create BeautifulSoup object from the HTML response

```
soup = BeautifulSoup(data, 'html.parser')
```

3. Extract all column/variable names from the HTML table header

```
html_tables = soup.find_all('table')
first_launch_table = html_tables[2]
column_names = []
for cols in first_launch_table.find_all('th'):
    col_name = extract_column_from_header(cols)
    if col_name is not None and len(col_name) > 0:
        column_names.append(col_name)
```

4. Create dictionary with keys from the extracted column names and then convert it into a dataframe

GitHub URL: https://github.com/elathr/AppliedDataScienceCapstone/blob/main/Data_Collection_Web_Scraping.ipynb

Data Wrangling

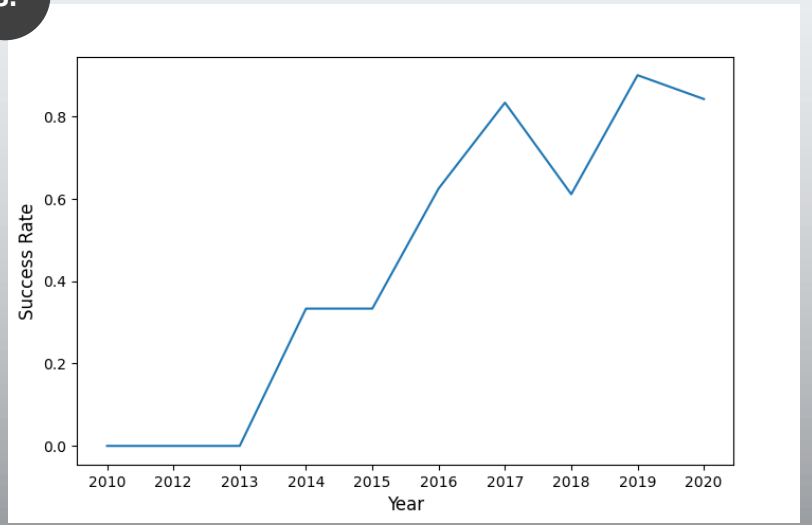
- The column **Outcome** in the dataframe indicates if the first stage successfully landed
- There are 8 different landing outcome labels: True Ocean, False Ocean, True RTLS, False RTLS, etc., which denote mission success (or failure) and landing type
- These landing outcomes were converted into training labels 0 and 1, where
  - 0 denotes a bad outcome (the booster did not land)
  - 1 denotes a good outcome (the booster did land)

GitHub URL: https://github.com/elathr/AppliedDataScienceCapstone/blob/main/Data_Wrangling.ipynb

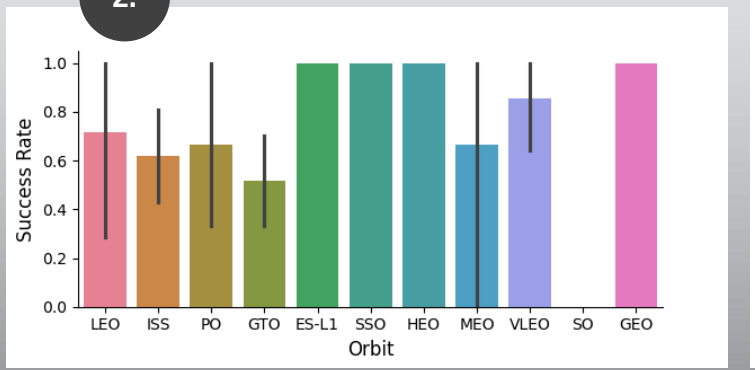
EDA with Data Visualization

1. Scatter plots:
   - Payload Mass vs. Flight Number
   - Launch Site vs. Flight Number
   - Launch Site vs. Payload Mass
   - Orbit vs. Flight Number
   - Orbit vs. Payload Mass
2. Bar plot:
   - Success Rate vs. Orbit
3. Line plot:
   - Success Rate vs. Year

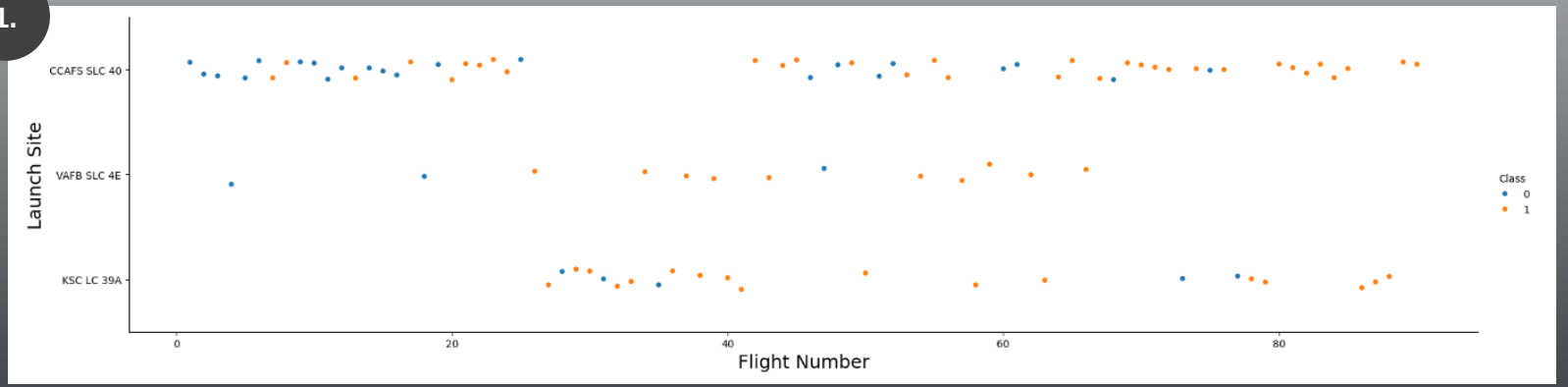
3.



2.



1.



GitHub URL: https://github.com/elathr/AppliedDataScienceCapstone/blob/main/EDA_with_Data_Visualization.ipynb

EDA with SQL

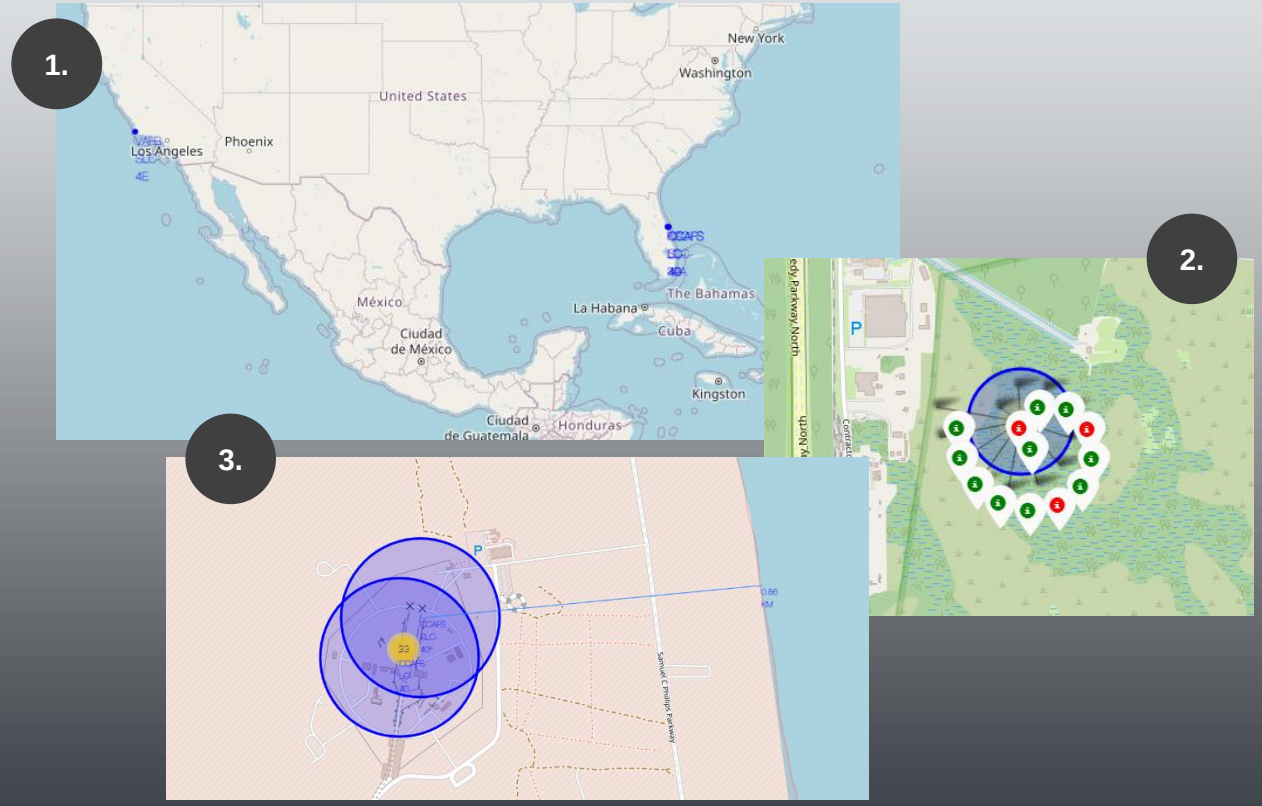
- The SpaceX launch dataset was analyzed using SQL to obtain some valuable insights from the data
- The following information were extracted from the dataset:
  - Names of unique launch sites in the space mission
  - Records where launch sites begin with the string "CCA"
  - Total payload mass carried by boosters launched by NASA (CRS)
  - Average payload mass carried by booster version F9 v1.1
  - Date when the first successful 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 failed missions
  - Booster versions which have carried the maximum payload mass
  - Records which display the month names, failure landing outcomes in drone ship, booster versions, and launch sites for the months in year 2015
  - Total number of landing outcomes (for each type of outcome) between the date 2010/06/04 and 2017/03/20

GitHub URL: https://github.com/elathr/AppliedDataScienceCapstone/blob/main/EDA_with_SQL.ipynb

Build an Interactive Map with Folium

- The launch success rate may depend on the location and proximities of a launch site, which can be easily visualized on a map using the Python library Folium

1. Mark the locations of launch sites using map objects such as markers, circles, and lines
2. Identify which launch sites have high success rate by using the color-labeled marker clusters
3. Calculate the distance between a launch site and its proximities (e.g., railways, highways and coastline)



GitHub URL: https://github.com/elathr/AppliedDataScienceCapstone/blob/main/Interactive_Visual_Analytics_with_Folium.ipynb

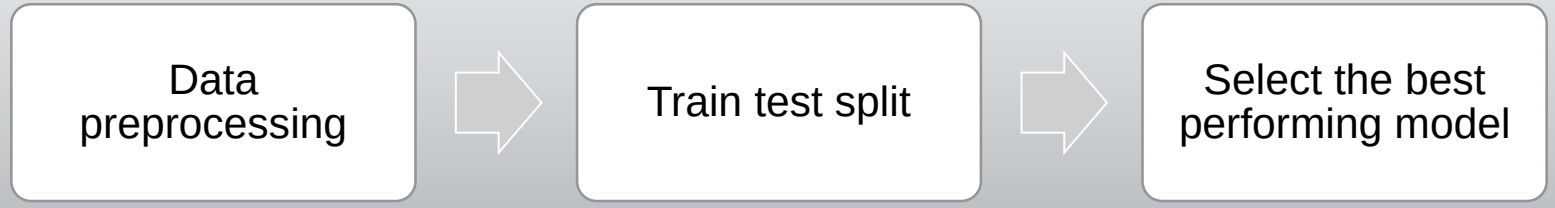
Build a Dashboard with Plotly Dash

- Plotly Dash application was created for users to perform interactive visual analytics on SpaceX launch data in real-time
- The dashboard contains the following elements:
  - **Pie chart** showing the total launches for selected launch site
  - **Scatter plot** showing the relationship between Payload mass (in kg) and Mission outcome for different booster versions

GitHub URL: https://github.com/elathr/AppliedDataScienceCapstone/blob/main/spacex_dash_app.py

Predictive Analysis (Classification)

- Objective: build a ML pipeline to predict whether first stage of Falcon 9 will land successfully



Building the model

- Load data
- Standardize data
- Split data into training and test datasets
- Decide what type of ML algorithm to use
- Build different models and tune different hyperparameters using GridSearchCV

Evaluating the model

- Calculate model accuracy on test data
- Plot confusion matrix

Improving the model

- Increase model accuracy with feature engineering and hyperparameter tuning techniques

Finding the best model

- Compare accuracy scores for different models, and select the model with the highest accuracy

GitHub URL: https://github.com/elathr/AppliedDataScienceCapstone/blob/main/Machine_Learning_Prediction.ipynb

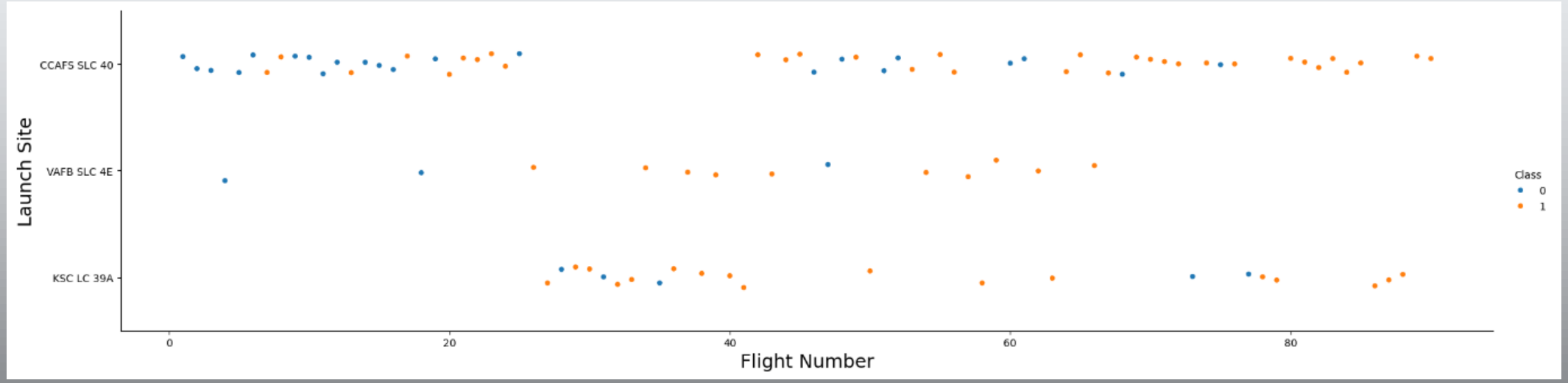
Results

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

Section 2

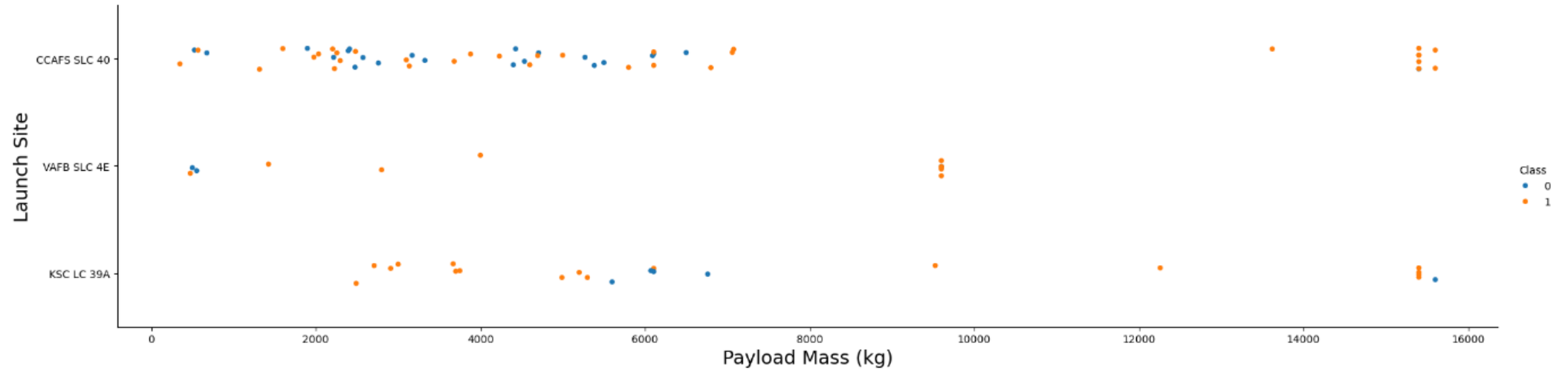
Insights Drawn from EDA

Flight Number vs. Launch Site



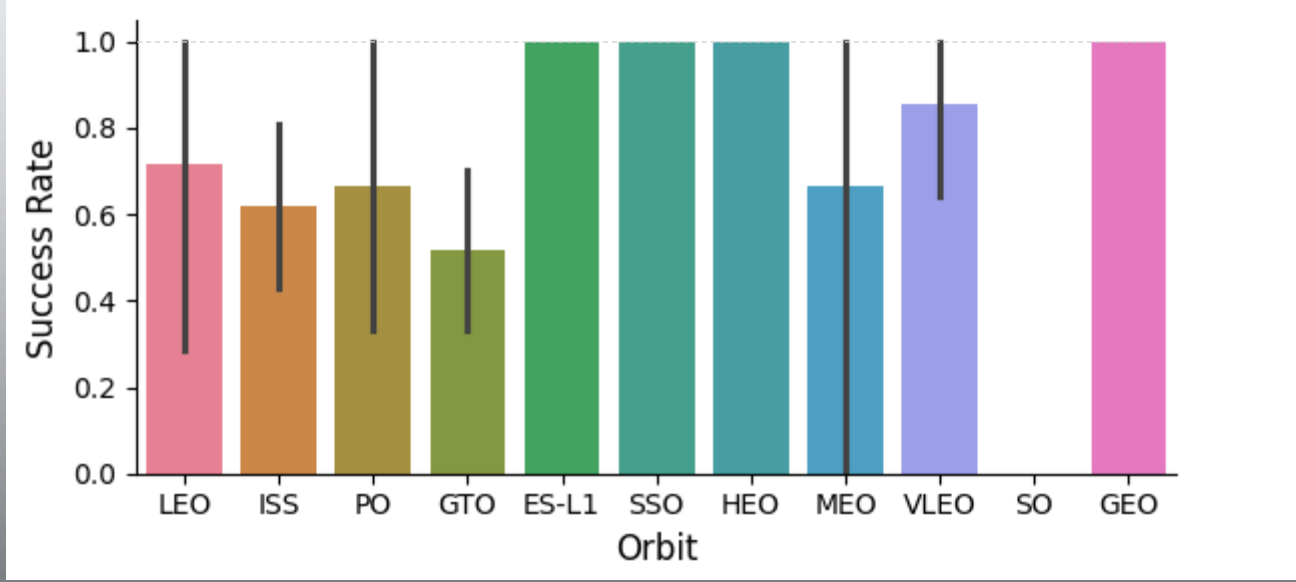
- The launch success rate increases with the number of flights

Payload vs. Launch Site



- Rockets with payload masses above 9000 kg have a high success rate
- It seems that the rockets with payload masses above 10000 kg could only be launched from the sites CCAFS SLC 40 and KSC LC 39A

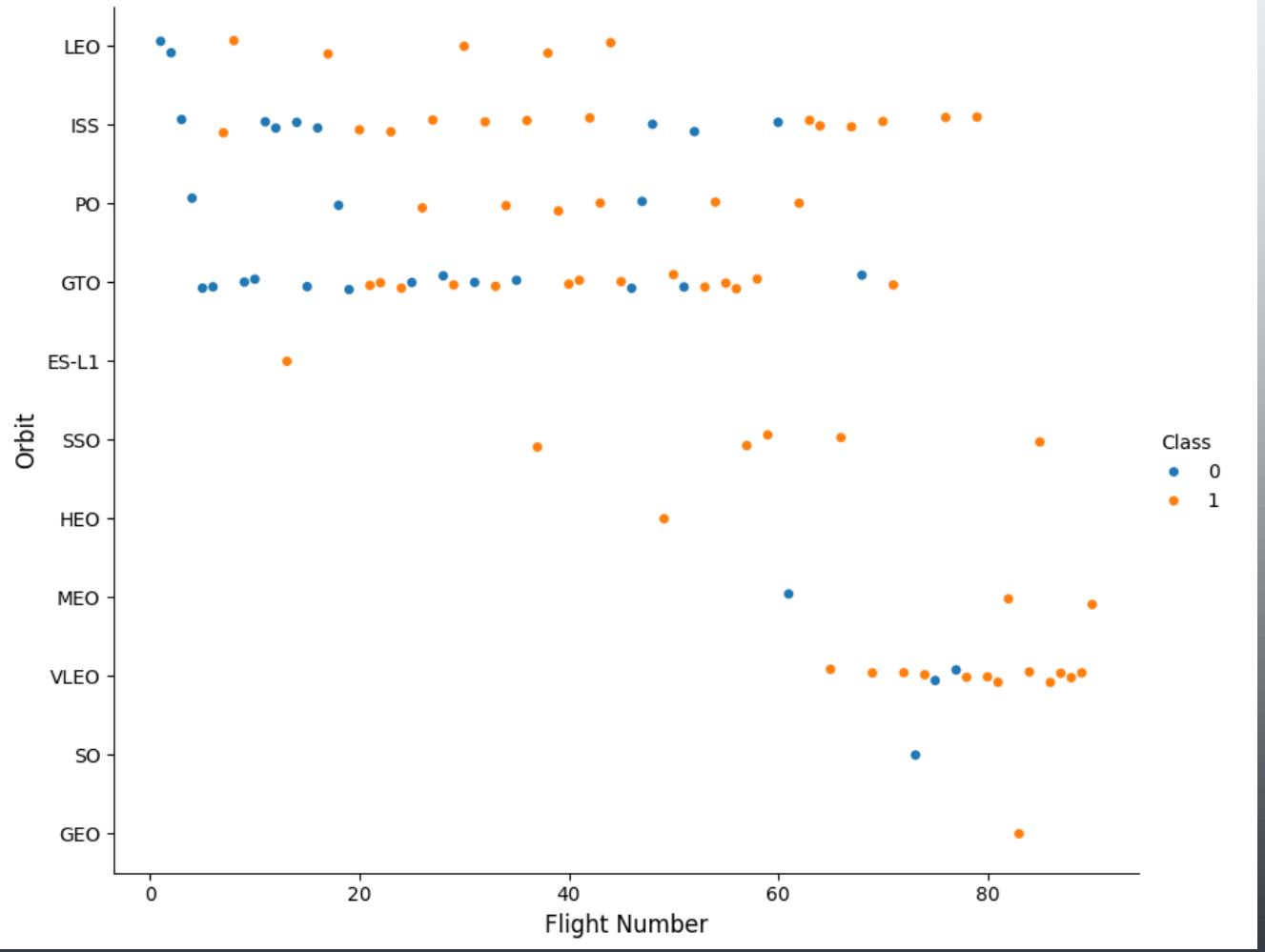
Success Rate vs. Orbit Type



- Orbits with the highest success rate (~100%) are ES-L1, SSO, HEO, and GEO

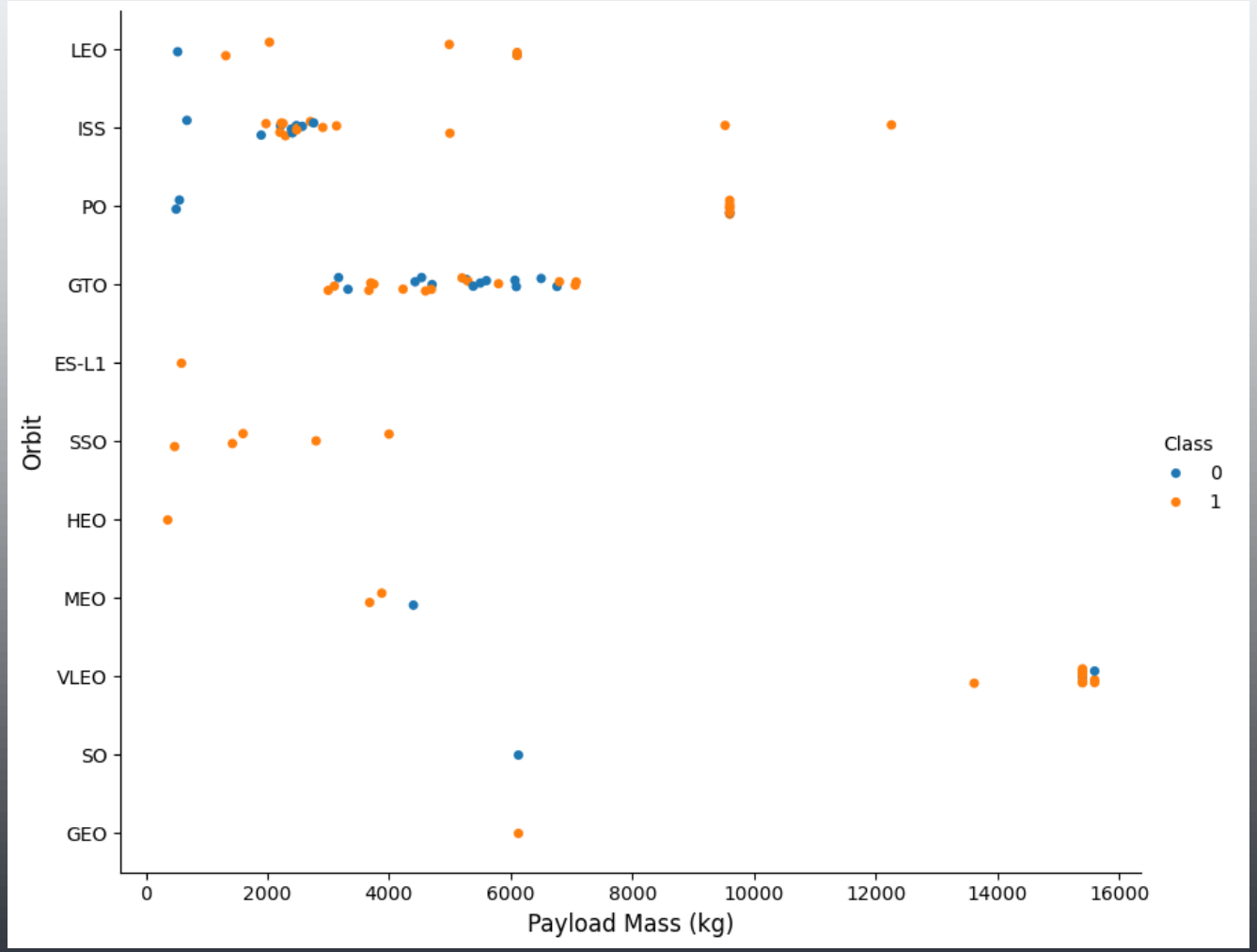
Flight Number vs. Orbit Type

- The data for the LEO orbit show that the success rate is increasing with the number of flights
- For GTO, there is no correlation between the number of flights and the success rate
- For ES-L1, HEO, SO, and GEO there is not enough data to draw a conclusion

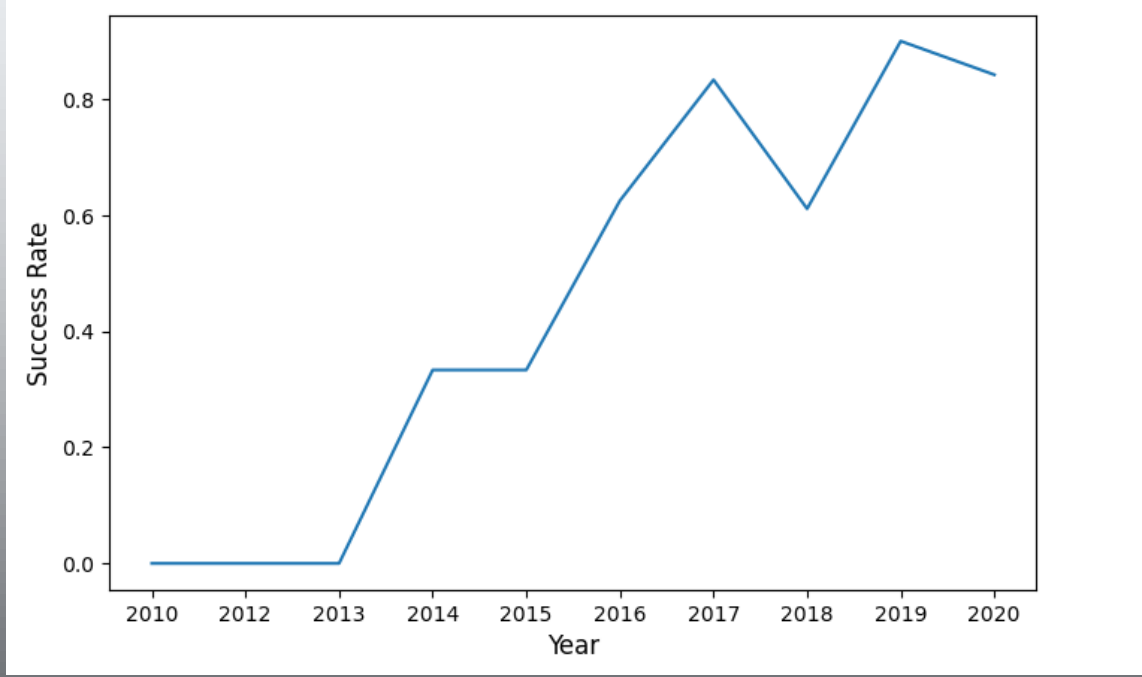


Payload vs. Orbit Type

- The orbits LEO, PO, and ISS have a high success rate with heavy payloads
- For GTO, there is no correlation between the payload mass and the success rate
- For ES-L1, HEO, SO, and GEO there is not enough data to draw a conclusion



Launch Success Yearly Trend



- The success rate increased gradually from 0 in 2013 to almost 100% in 2020

All Launch Site Names

SQL query

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

Query result

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

- The SELECT DISTINCT statement is used to return only distinct values of launch site

Launch Site Names Begin with 'CCA'

SQL query

```
%sql SELECT * \
     FROM SPACEXTABLE \
     WHERE `Launch_Site` LIKE 'CCA%' \
     LIMIT 5;
```

- The wildcard character % is used to find values that start with CCA
- The LIMIT clause is used to specify the number of records to return (5)

Query 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	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 query

```
%sql SELECT `Booster_Version`, SUM(`PAYLOAD_MASS__KG_`) AS TOTAL_PAYLOAD_MASS \
     FROM SPACEXTABLE \
     WHERE `Customer` = 'NASA (CRS)' \
     GROUP BY `Booster_Version` \
     ORDER BY TOTAL_PAYLOAD_MASS;
```

Query result

Booster_Version	TOTAL_PAYLOAD_MASS
F9 v1.0 B0006	500
F9 v1.0 B0007	677
F9 v1.1 B1015	1898
F9 v1.1 B1018	1952
F9 B5 B1059.2	1977
F9 FT B1035.2	2205
F9 v1.1 B1010	2216
F9 FT B1025.1	2257
F9 B5 B1056.2	2268
F9 v1.1	2296
F9 v1.1 B1012	2395
F9 FT B1031.1	2490
F9 B5B1056.1	2495
F9 B5B1050	2500
F9 B4 B1039.2	2647
F9 B4 B1045.2	2697
F9 FT B1035.1	2708
F9 B5 B1058.4	2972
F9 FT B1021.1	3136
F9 B4 B1039.1	3310

- The SUM() function is used with the GROUP BY clause to return the total payload mass for each booster

Average Payload Mass by F9 v1.1

SQL query

```
%sql SELECT AVG(`PAYLOAD_MASS__KG_`) AS AVERAGE_PAYLOAD_MASS \
     FROM SPACEXTABLE \
     WHERE `Booster_Version` = 'F9 v1.1';
```

Query result

AVERAGE_PAYLOAD_MASS
2928.4

- The AVG() function is used to return the average payload mass carried by the booster version F9 v1.1

First Successful Ground Landing Date

SQL query

```
%sql SELECT `Date` \
     FROM SPACEXTABLE \
     WHERE `Landing_Outcome` = 'Success (ground pad)' \
     ORDER BY `Date`\
     LIMIT 1;

#%sql SELECT MIN(`Date`) \
#     FROM SPACEXTABLE \
#     WHERE `Landing_Outcome` = 'Success (ground pad)' \
#     ORDER BY `Date`;
```

Query result

Date
2015-12-22

- The ORDER BY keyword is used with the LIMIT clause to return the date of the first successful landing outcome on ground pad
- The MIN() function can be used as well

Successful Drone Ship Landing with Payload between 4000 and 6000

SQL query

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

Query result

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

- The WHERE clause is used with the AND operator to filter records and return the boosters that have successfully landed on the drone ship and had a payload mass greater than 4000, but less than 6000 kg

Total Number of Successful and Unsuccessful Mission Outcomes

SQL query

```
%sql SELECT (SELECT COUNT(`Mission_Outcome`) \
     FROM SPACEXTABLE \
     WHERE `Mission_Outcome` LIKE 'Success%') AS SUCCESS, \
     (SELECT COUNT(`Mission_Outcome`) \
     FROM SPACEXTABLE \
     WHERE `Mission_Outcome` LIKE 'Failure%') AS FAILURE;
```

Query result

SUCCESS	FAILURE
100	1

- Two subqueries are used: the first one returns the total number of successful mission outcomes, and the second one returns the total number of unsuccessful mission outcomes

Boosters Carried Maximum Payload

SQL query

```
%sql SELECT `Booster_Version` \
     FROM SPACEXTABLE \
     WHERE `PAYLOAD_MASS__KG_` = (SELECT MAX(`PAYLOAD_MASS__KG_`) FROM SPACEXTABLE);
# max. payload mass is 15600 kg
```

Query 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

- The MAX() function is used to return the largest payload mass
- The subquery is used because the WHERE keyword cannot be used with aggregate functions

2015 Launch Records

SQL query

```
%sql SELECT SUBSTR(`Date`, 6, 2) AS Month, SUBSTR(`Date`, 0, 5) AS Year, `Booster_Version`, `Landing_Outcome`, `Launch_Site` \
     FROM SPACEXTABLE \
     WHERE Year = '2015' AND `Landing_Outcome` = 'Failure (drone ship)';
```

Query result

Month	Year	Booster_Version	Landing_Outcome	Launch_Site
01	2015	F9 v1.1 B1012	Failure (drone ship)	CCAFS LC-40
04	2015	F9 v1.1 B1015	Failure (drone ship)	CCAFS LC-40

- The SUBSTR() function is used to extract the year and month from a column Date

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

SQL query

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

Query result

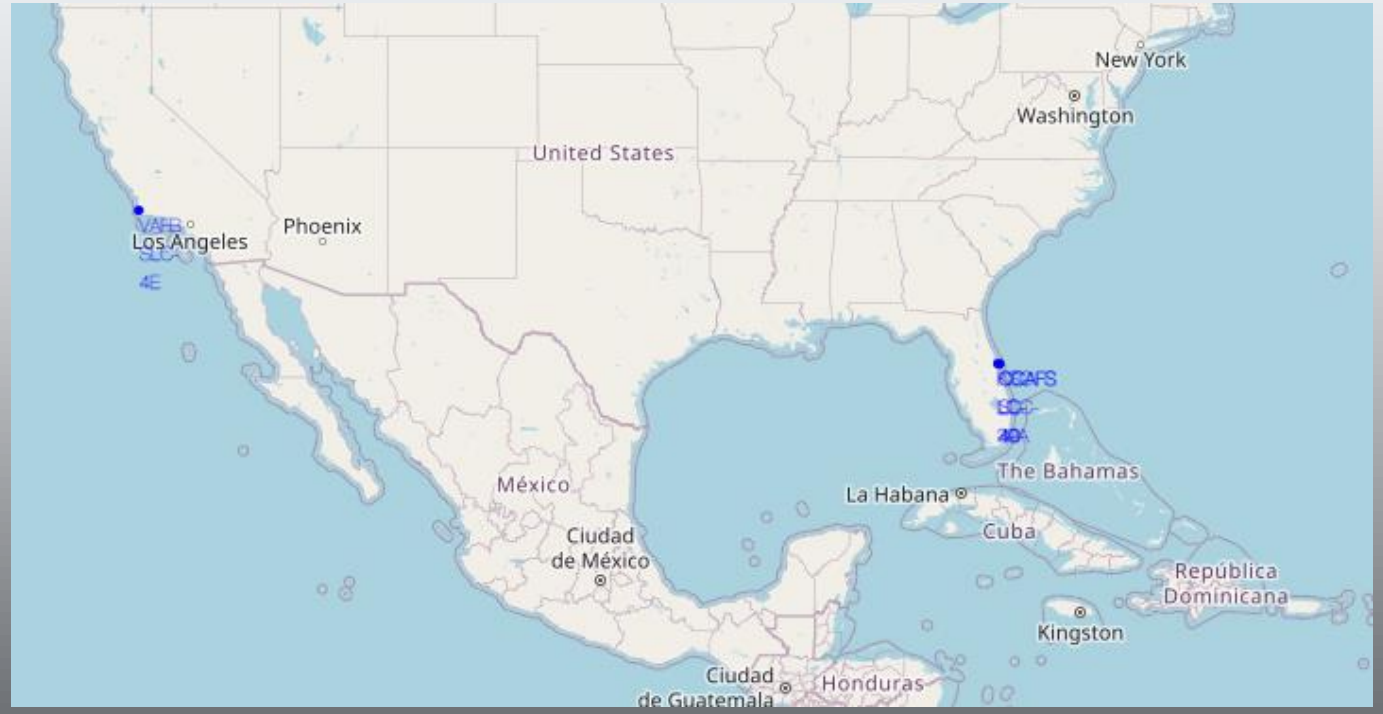
Landing_Outcome	TOTAL_NUMBER
No attempt	10
Success (drone ship)	5
Failure (drone ship)	5
Success (ground pad)	3
Controlled (ocean)	3
Uncontrolled (ocean)	2
Failure (parachute)	2
Precluded (drone ship)	1

- The BETWEEN operator is used to select dates within a given range

Section 3

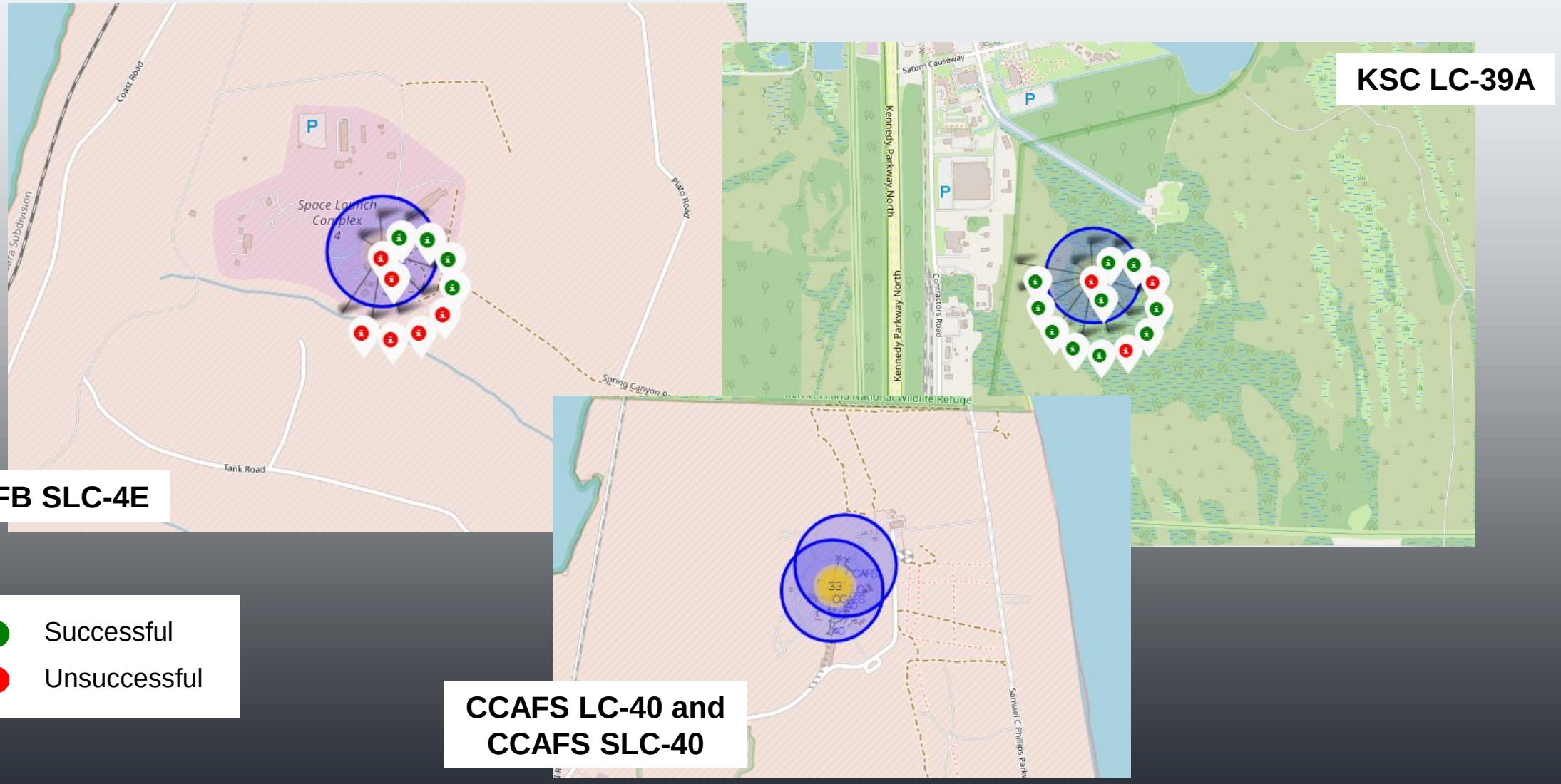
Launch Sites Proximities Analysis

Launch Sites



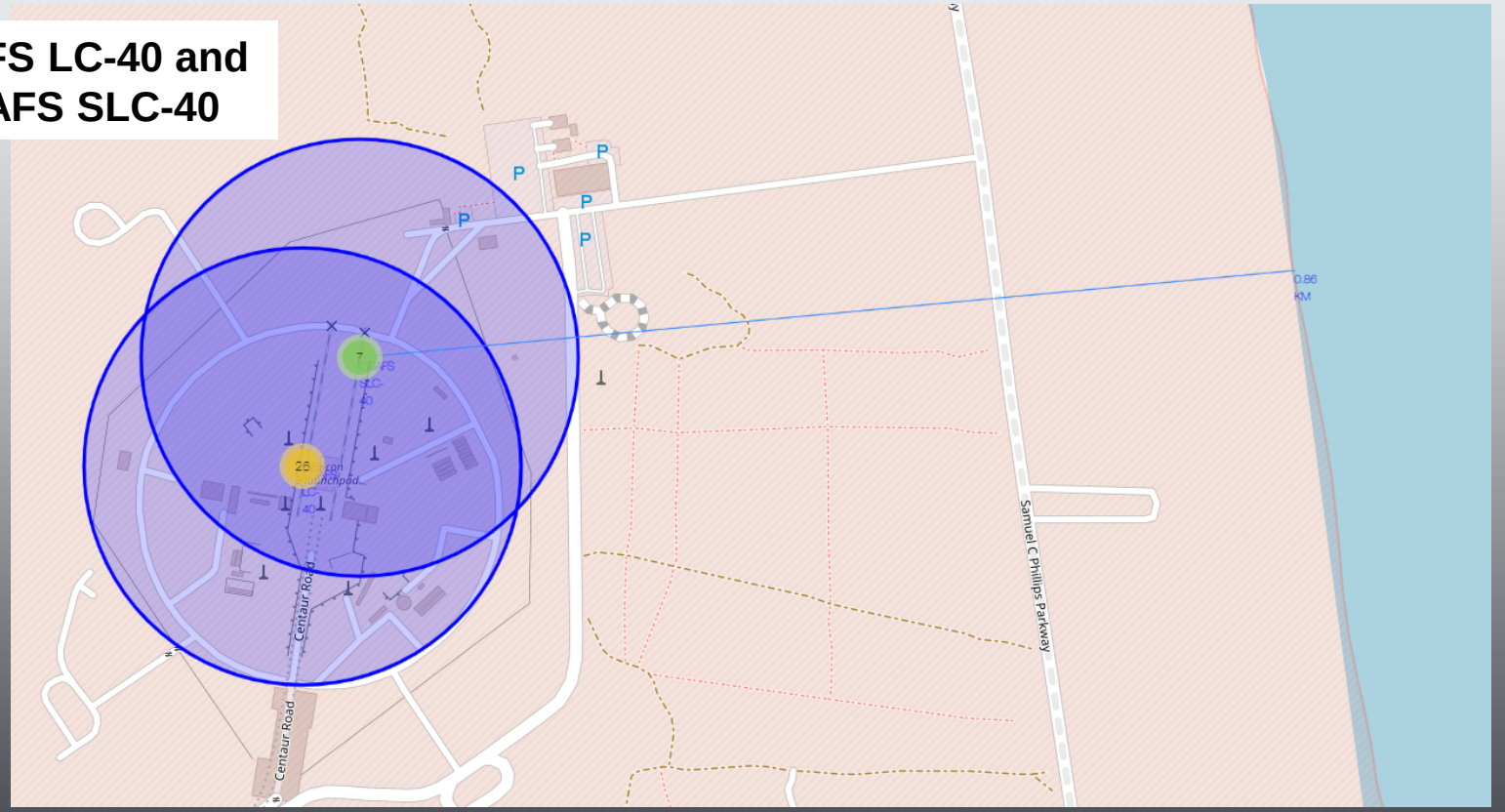
- The above map shows the four locations of the established SpaceX launch sites (marked by blue circles)

Color-labeled Launch Outcomes



Launch Site Distance to Coastline

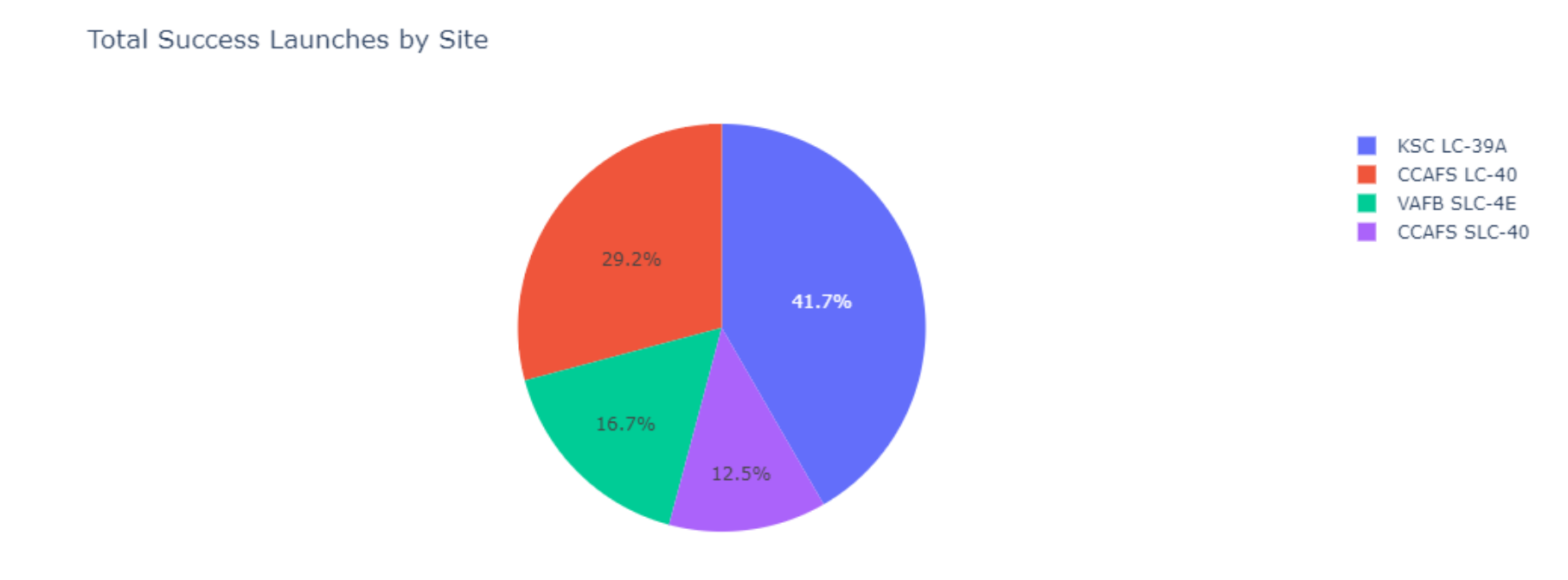
- Launch sites CCAFS LC-40 and CCAFS SLC-40 are situated near the coastline, at a distance of ~0.86 km



Section 4

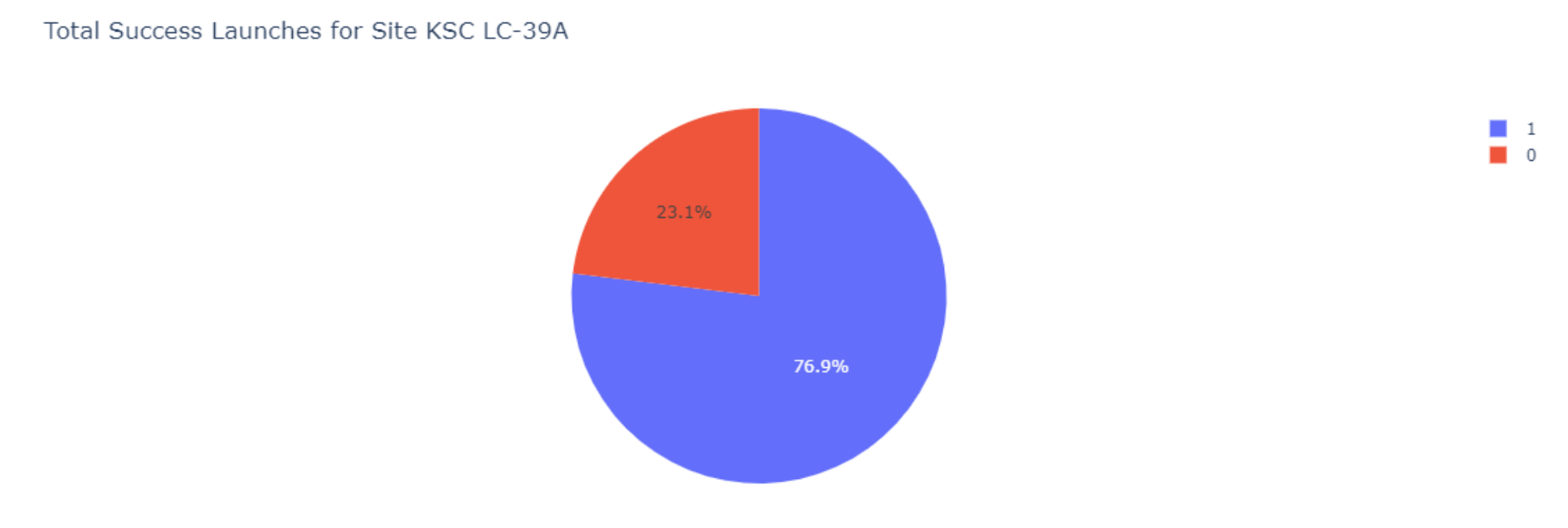
Build a Dashboard with Plotly Dash

Launch Success Rate by Site



- The launch site KSC LC-39A has the highest launch success rate

Launch Success Rate for KSC LC-39A



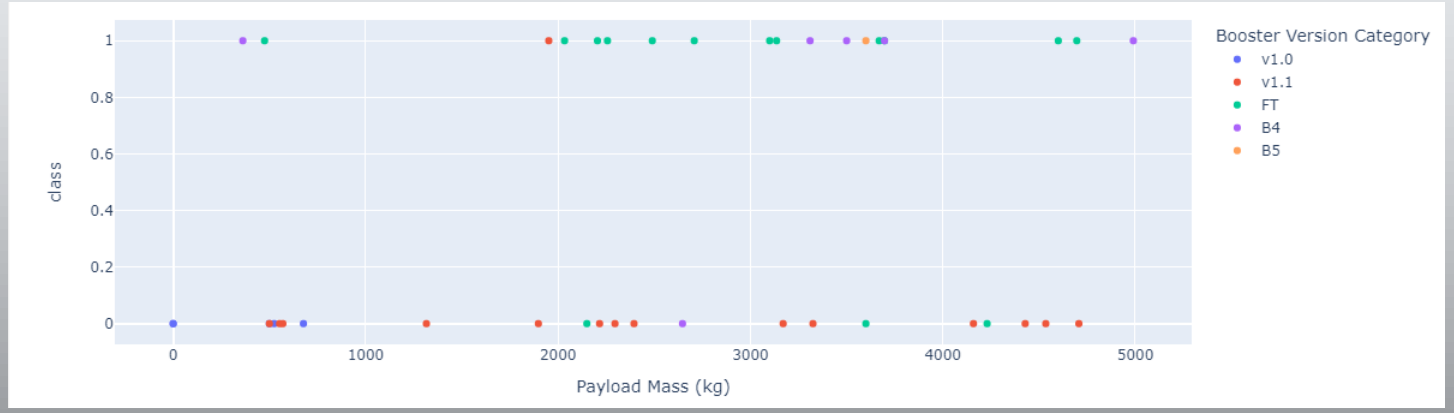
- 76.9% of all launches from this site were successful

Payload vs. Launch Outcome

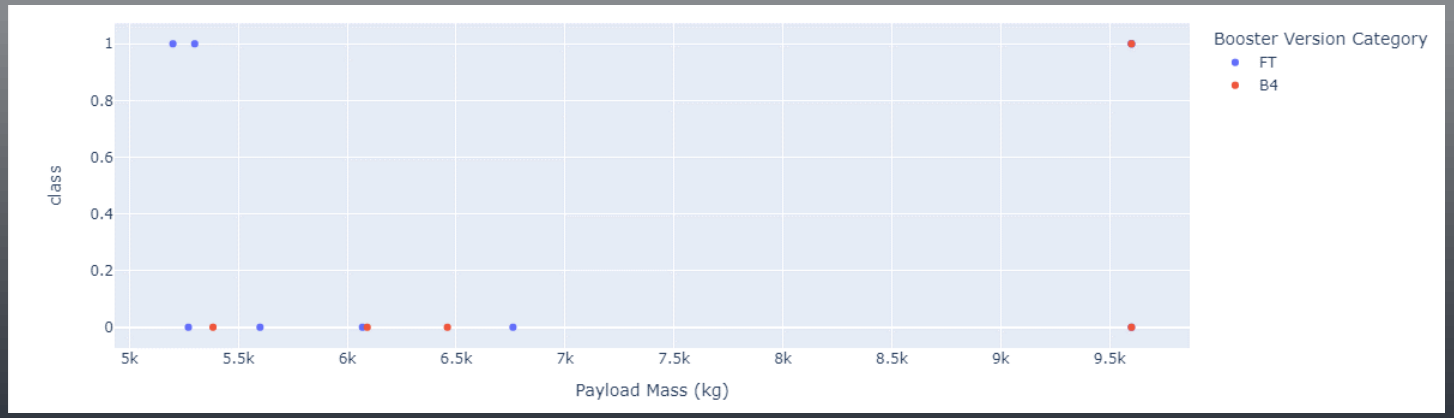
Relationship between Payload mass (kg) and Mission outcome for different booster versions, for all launch sites

- The Falcon 9 FT booster has the highest launch success rate

Payload range 0-5000 kg



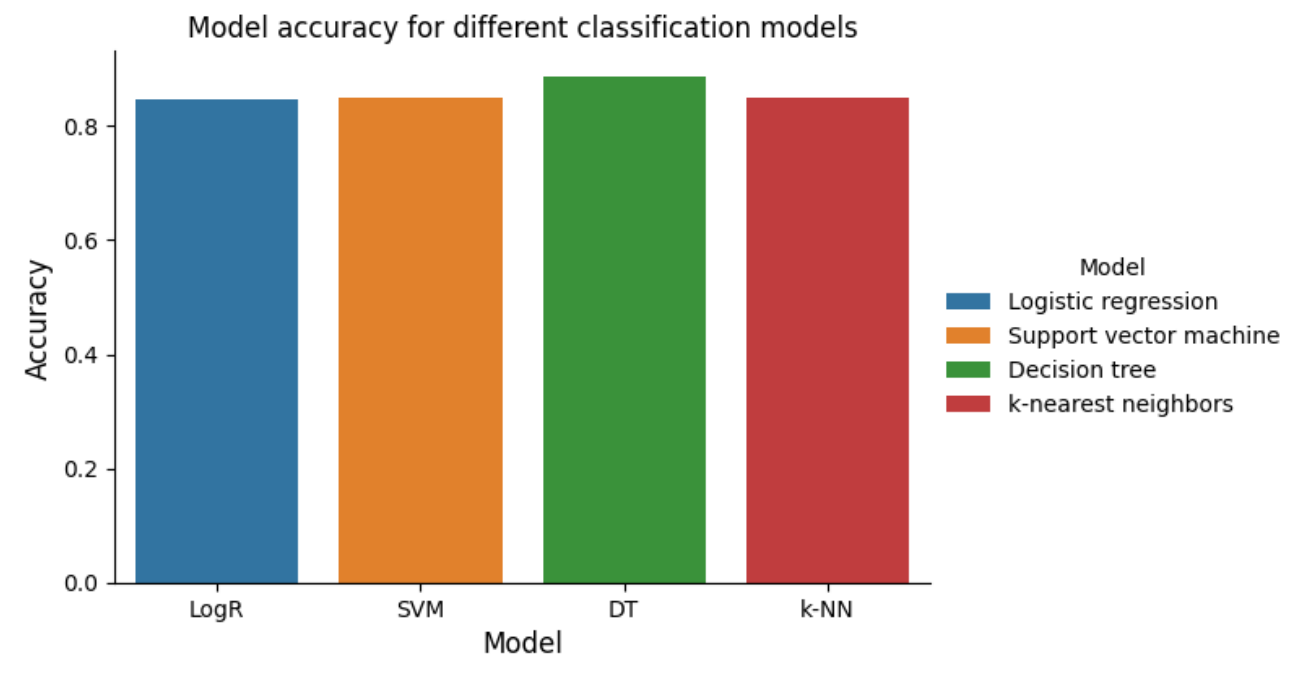
Payload range 5000-10000 kg



Section 5

Predictive Analysis (Classification)

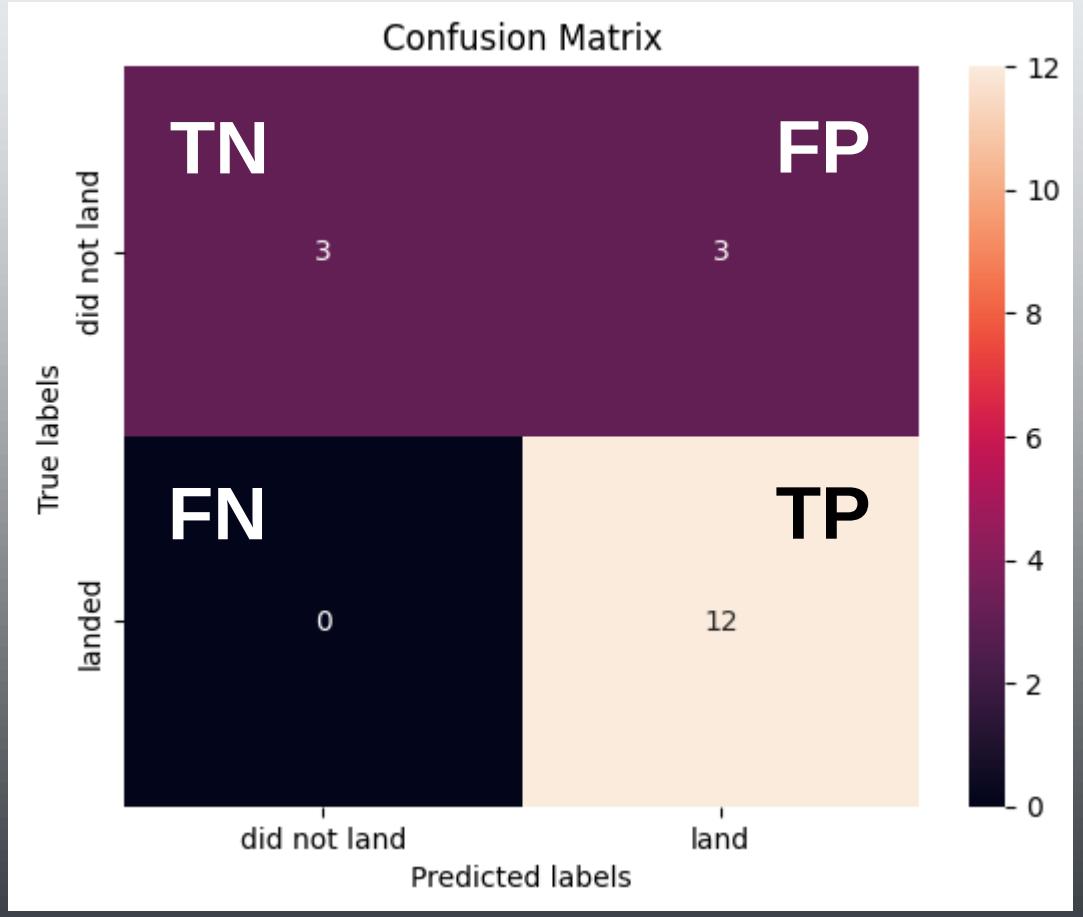
Classification Accuracy



- The decision tree classifier is the model with the highest classification accuracy of 89%, although the other models performed similarly (~85%)

Confusion Matrix

- A confusion matrix is used to assess the performance of a classification model on a set of test data
- From the confusion matrix, it can be seen that the classifier can distinguish between the different classes; the major problem is false positives (FP)



Conclusions

- The SpaceX launch success rate gradually increased over the years (starting from 2013)
- Orbits ES-L1, SSO, HEO, and GEO have a 100% success rate
- Kennedy Space Center Launch Complex 39A has the highest success rate of all launch sites
- The decision tree classifier is the best ML algorithm for this dataset

Thank you! :)