Winning Space Race with Data Science

Emmanouil Spanoudakis

23.07.2023

Outline

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

Executive Summary

- Summary of methodologies
  - Data collection via API
  - Data collection via Web Scraping
  - Data wrangling
  - Explanatory data analysis with SQL
  - Explanatory data analysis with data visualization
  - Interactive Visual analytics and dashboard
  - Predictive analysis (machine learning)
- Summary of all results
  - Explanatory Data Analysis results
  - Interactive maps and dashboards
  - Predictive results derived from the data model

Introduction

- Project background and context
  - Companies are competing to make space travel affordable
  - Space X is the most cost effective with 62M cost per launch vs 165M of others
  - Space X is recovering and reusing first stage (large and expensive)
  - Aim is to have successful landing of first stage
  - As a data scientist of Space Y, the aim is to predict the price of each launch by gathering information about Space X
  - The goal of the project is based on the provided characteristics of a launch to predict if the first stage will land. If we can determine if the first stage will land, we can determine the cost of a launch
- Problems you want to find answers
  - What are the characteristics of a launch which influence the successful/failure of a landing?
  - What is the relationship between the launch variables and how do they influence the success or the failure of the first stage landing?
  - Which are the optimal conditions in order to achieve the best landing success rate?

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - SpaceX REST API
  - Web scraping from Wikipedia (`List of Falcon 9 and Falcon Heavy launches`)
- Perform data wrangling
  - Calculate the number of launches on each site
  - Calculate the number and occurrence of each orbit
  - One-hot encoding for the classification variable
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Standardize data, train_test_split, train the model, determine best model and output confusion matrix

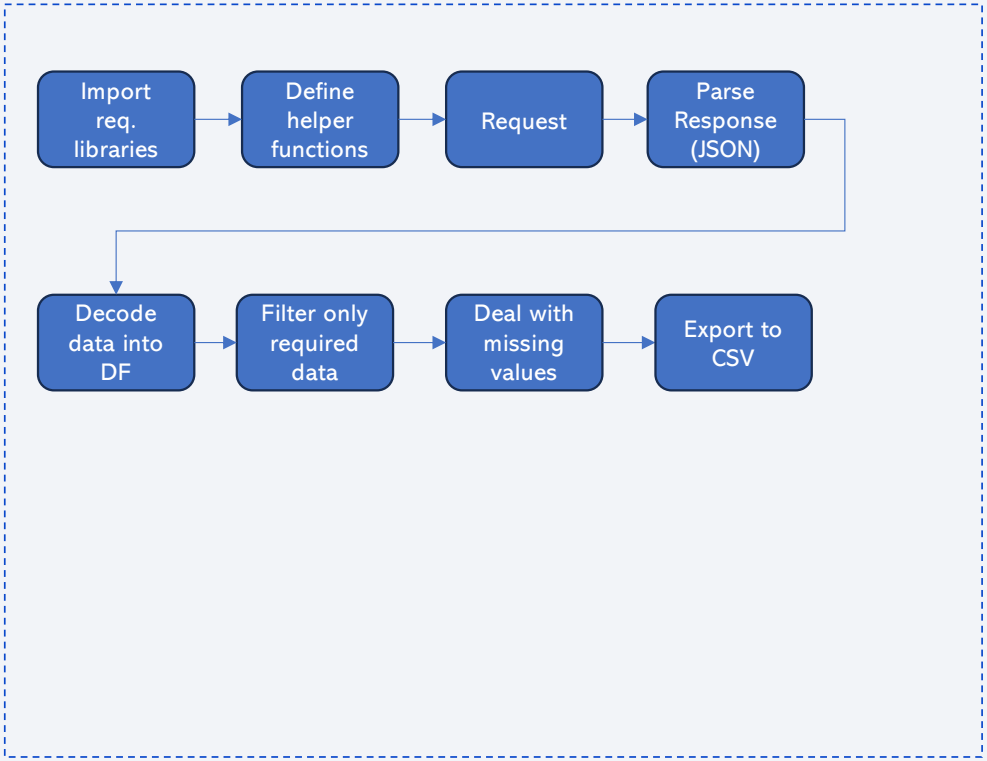
Data Collection

- Space X launch data was collected by Rest API and Web Scrapping
- Rest API
  - Different end points available, we will use “api.spacexdata.com/v4/launches/past”
- Web Scrapping
  - Wikipedia

Data Collection – SpaceX API

Process:

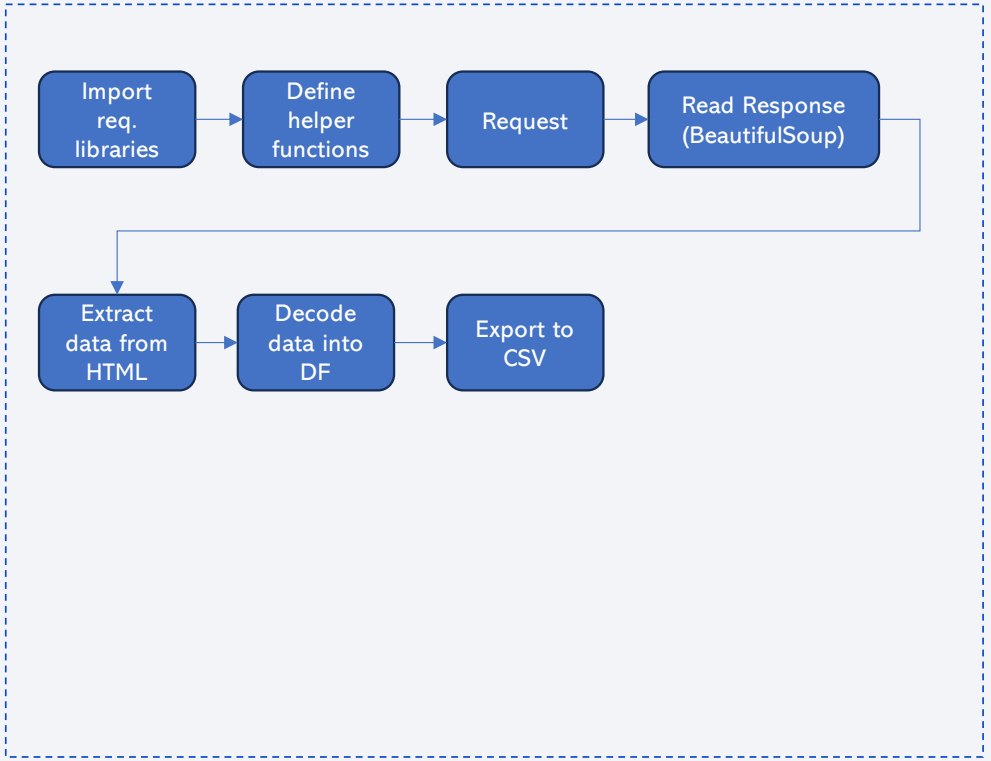
- Space X provides a public API with data
- Access the api to retrieve the data
- Transform and Clean the data
- Export to CSV



More details available in the following GitHub URL:
https://github.com/manSpan/ADScienceCapstone/blob/02951d7456064548f65a48267495ee62f4890f0b/jupyter-labs-spacex-data-collection-api.ipynb

Data Collection - Scraping

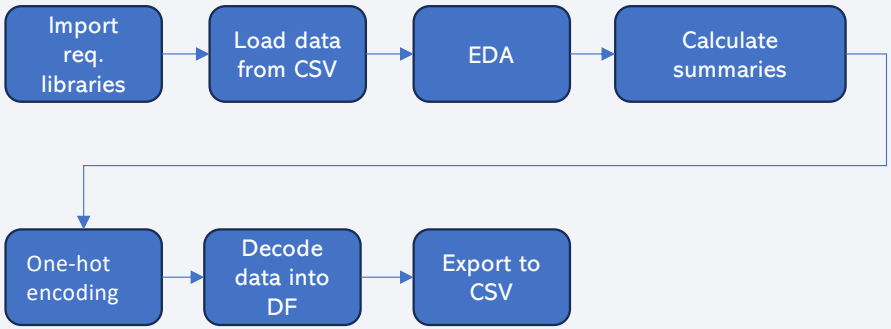
- Retrieve data regarding Falcon 9 historical launch records from a Wikipedia page titled 'List of Falcon 9 and Falcon Heavy launches'
- Extract data from the response and decode it into a dataframe
- Export data in CSV



GitHub URL of the completed web scraping notebook:
https://github.com/manSpan/ADScienceCapstone/blob/02951d7456064548f65a48267495ee62f4890f0b/jupyter-labs-webscraping.ipynb

Data Wrangling

- Perform exploratory Data Analysis
- Calculate summaries
- Determine Training Labels
- One-hot encoding for the classification variable



GitHub URL of the completed data wrangling related notebooks:
https://github.com/manSpan/ADScienceCapstone/blob/02951d7456064548f65a48267495ee62f4890f0b/labs-jupyter-spacex-data_wrangling_jupyterlite.jupyterlite.ipynb

EDA with Data Visualization

- Different types of graphs were leveraged
- Scatter graphs: To identify correlations between variables
- Bar graphs: to visualize the relationship between numeric and categorical values
- Line graphs: to visualize trends

GitHub URL of the completed EDA with data visualization notebook:
https://github.com/manSpan/ADScienceCapstone/blob/02951d7456064548f65a48267495ee62f4890f0b/jupyter-labs-eda-dataviz.ipynb.jupyterlite.ipynb

EDA with SQL

- A number of SQL queries were performed to analyze the dataset
  - Display the names of the unique launch sites
  - 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 achieved
  - 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 (with 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 successful landing_outcomes between the date 04-06-2010 and 20-03-2017 in descending order.
- The exact queries are available here: https://github.com/manSpan/ADScienceCapstone/blob/02951d7456064548f65a48267495ee62f4890f0b/jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

- A number of maps with several map objects such as markers, circles, lines, were created and created and added to a folium map in order to:
  - Mark all launch sites on a map and the number of launches (using Markers)
  - Mark the success/failed launches for each site on the map (using colored labeled markers)
  - Calculate the distances between a launch site to its proximities (using lines)
- This was done to investigate the impact a location has to the success / failure of a launch

GitHub URL of your completed interactive map with Folium map:
https://github.com/manSpan/ADScienceCapstone/blob/02951d7456064548f65a48267495ee62f4890f0b/labs-jupyter-spacex-data_wrangling_jupyterlite.jupyterlite.ipynb

Build a Dashboard with Plotly Dash

The following graphs and plots were build in a Plotly dashboard

- Percentage of launces per launch site
- Plot to visualize the success/failure based on the site, the launch size and the Booster version.
- These visualizations help to detect patterns with regards to the location, the launch size and the booster version

GitHub URL of your completed Plotly Dash lab:
https://github.com/manSpan/ADScienceCapstone/blob/02951d7456064548f65a48267495ee62f4890f0b/spacex_dash_app.py

Predictive Analysis (Classification)

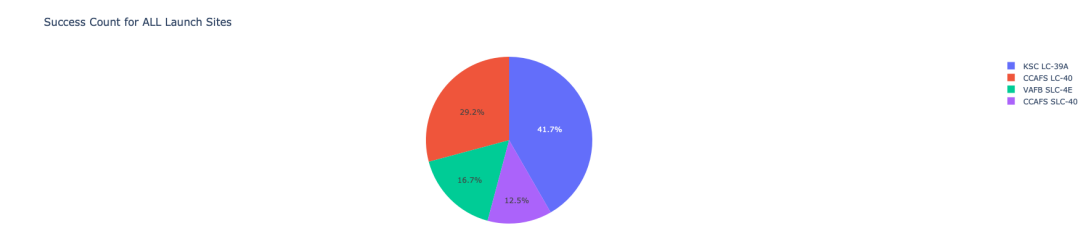
- Four classification models were evaluated
  - Logistic regression
  - Support vector machine
  - K nearest neighbors
  - Decision Tree Classifier
- The approach steps

GitHub URL of the predictive analysis lab:
https://github.com/manSpan/ADScienceCapstone/blob/02951d7456064548f65a48267495ee62f4890f0b/SpaceX_Machine_Learning_Prediction_Part_5.jupyterlite.ipynb

Exploratory data analysis results

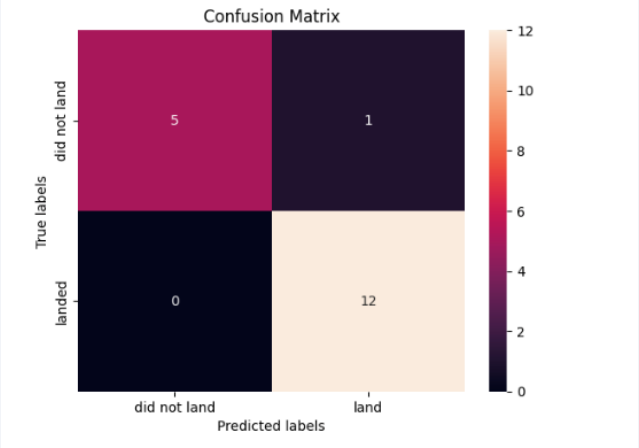
- Space X has 4 launch sites
- KSC LC-39A is the most successful site
- CCAFS SLC-40 is the less successful site
- Certain Orbit types have 100% success rates
- Initial launches were mostly failures for CCAFS SLC 40
- Success is increasing with the years

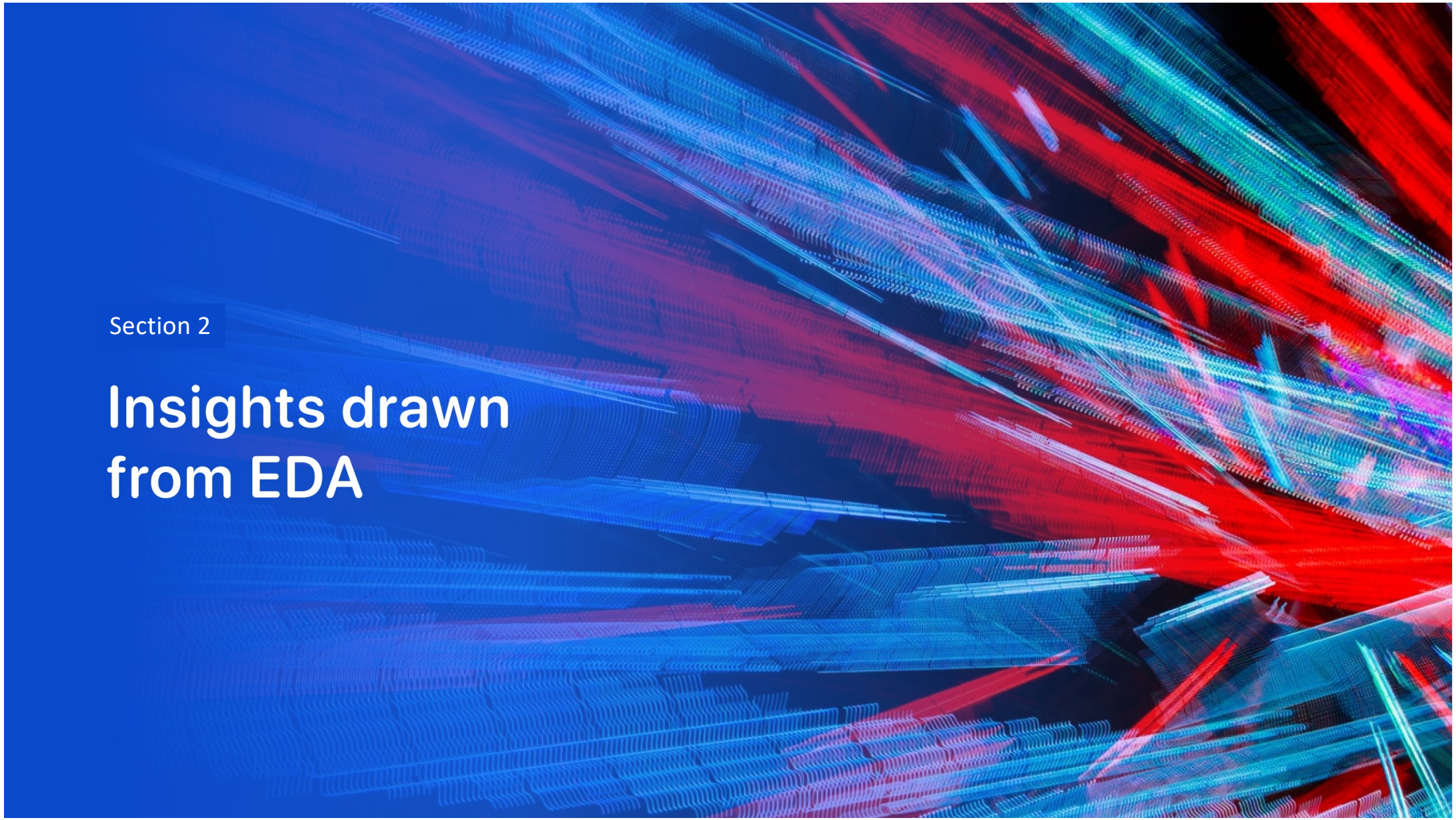
Exploratory data analysis results



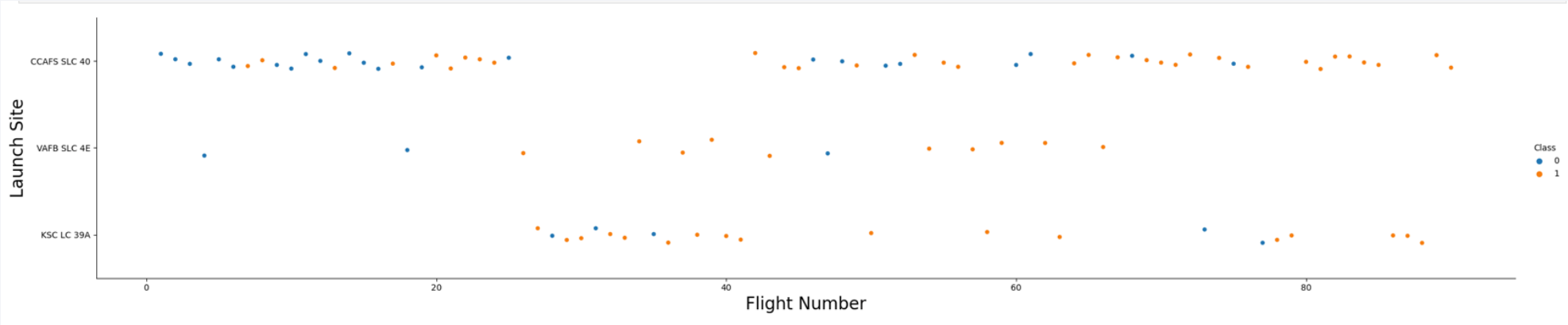
Predictive analysis results

- All models have similar accuracy
- The best performing model is the Decision Tree
- The confusion matrix is similar for all models



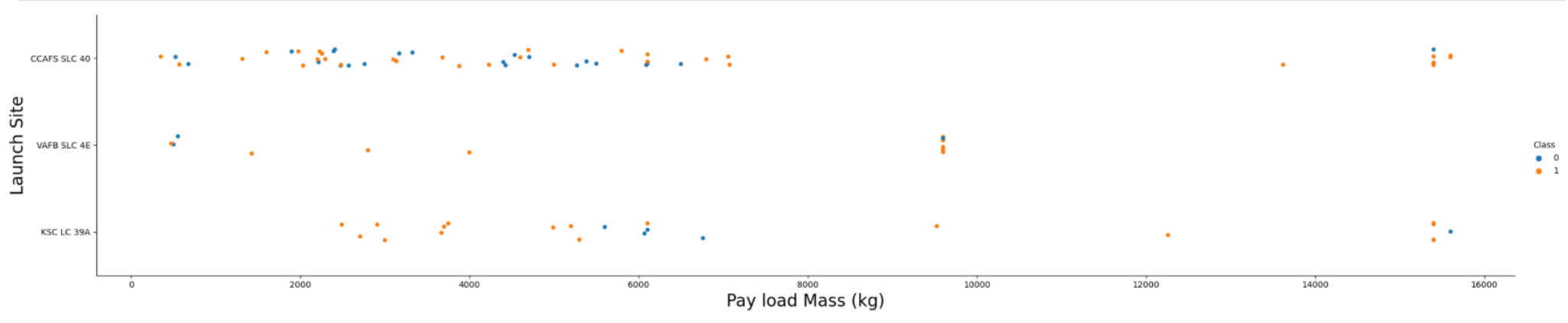

Section 2
Insights drawn from EDA

Flight Number vs. Launch Site



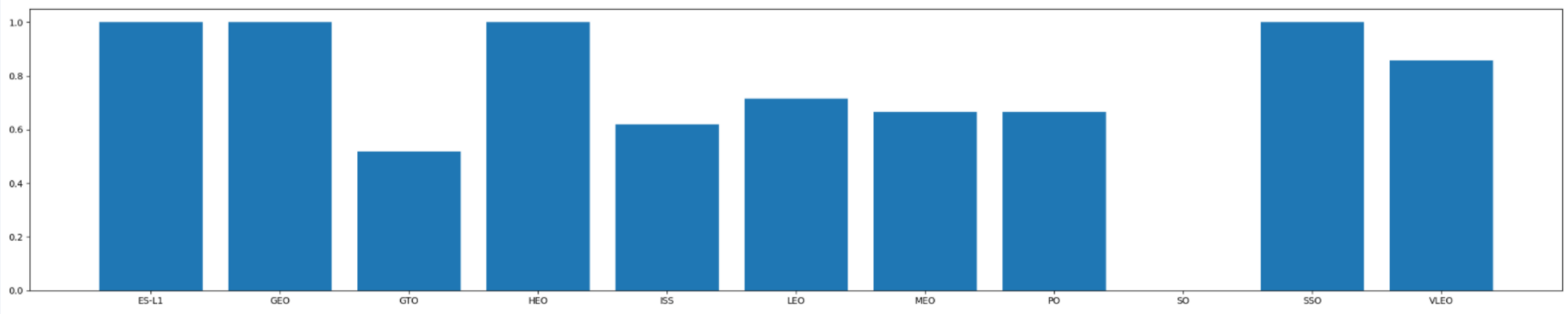
- There is a trend of increasing successful rate with the increase of Flight number
- The initial attempts for CCAFS SLC 40 were mostly failures

Payload vs. Launch Site



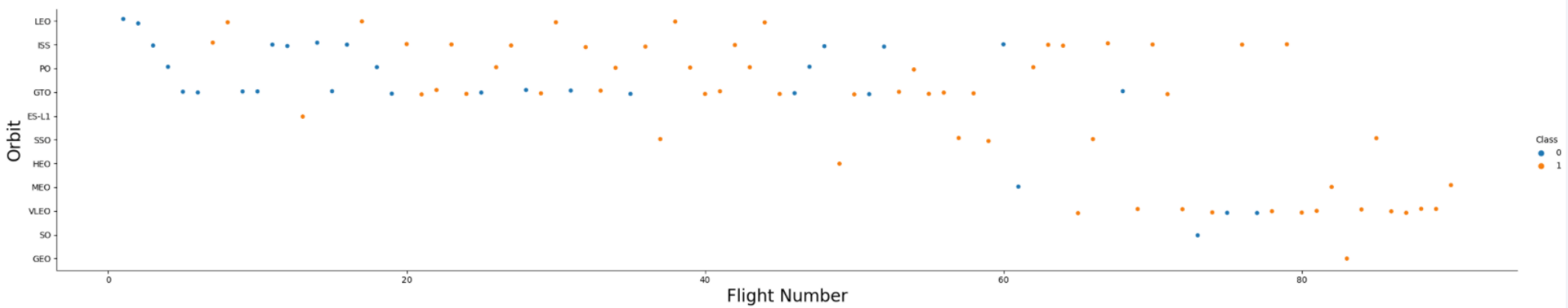
- Payloads above 7000Kg tend to have higher success rate
- KSC has more success for lighter and heavier payloads while it has a higher failure rate around 6000kg
- There's a different behavior per site

Success Rate vs. Orbit Type



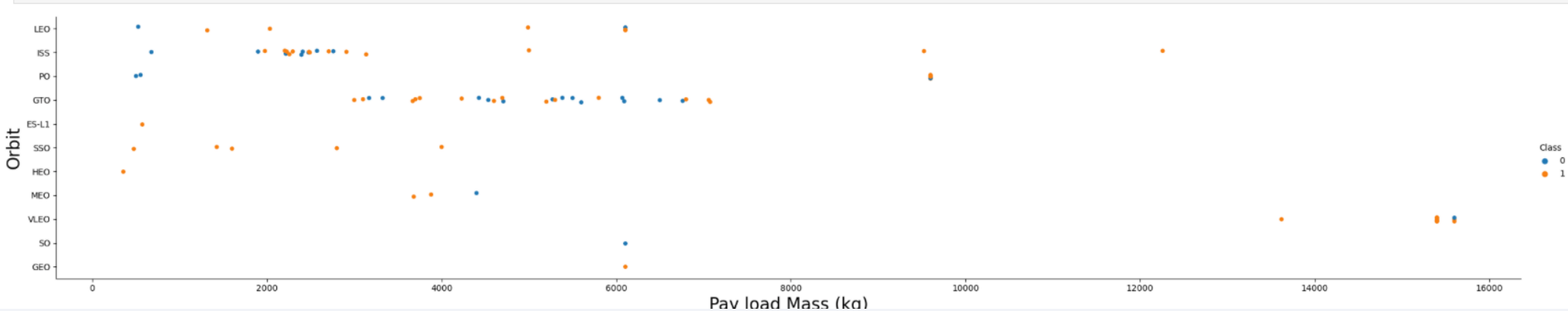
- Certain Orbit types have 100% success rates
- SO has 0%
- May be misleading if we don't evaluate the number of samples per orbit type

Flight Number vs. Orbit Type



- For LEO orbit the Success appears related to the number of flights
- It seems to be no relationship between flight number when in GTO orbit.
- Failures in the initial flights

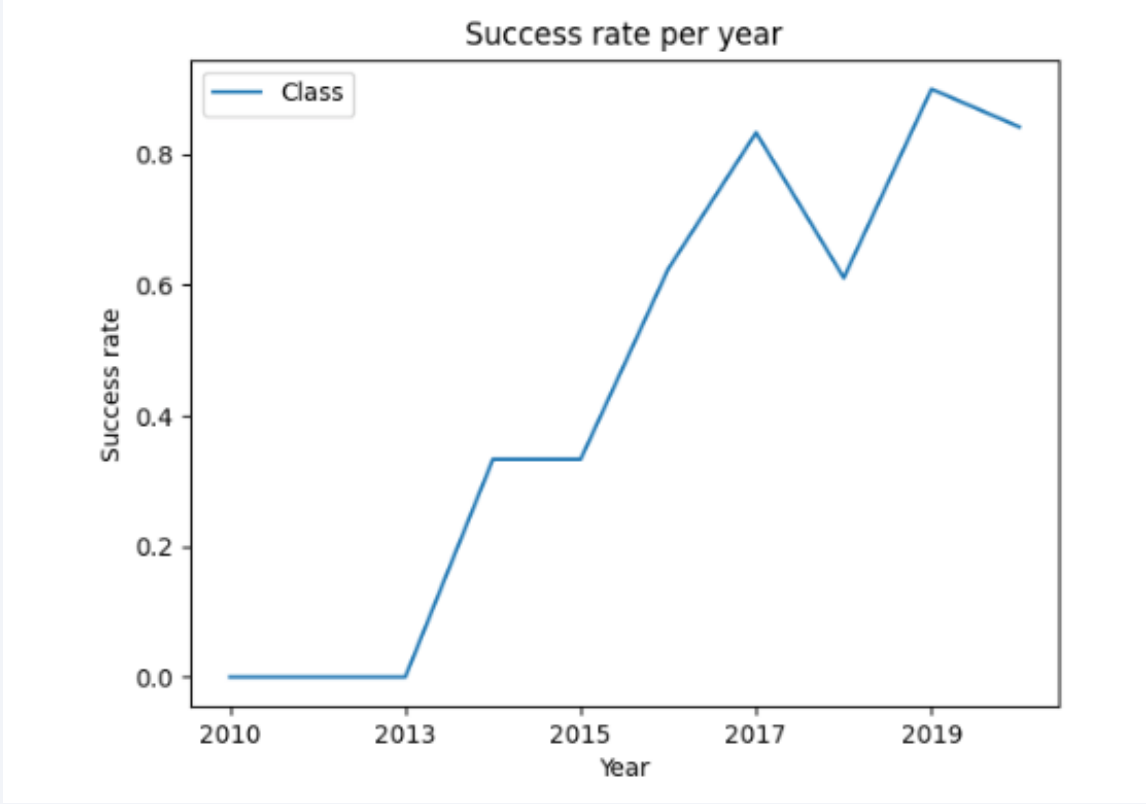
Payload vs. Orbit Type



- With heavy payloads the successful landing or positive landing rate are more for Polar, LEO and ISS.
- For GTO we cannot distinguish this well as both positive landing rate and negative landing (unsuccessful mission) are there.

Launch Success Yearly Trend

- The success rate since 2013 kept increasing till 2020 with an exception for 2018



All Launch Site Names

Task 1

Display the names of the unique launch sites in the space mission

```
%%sql

SELECT distinct("Launch_Site") FROM SPACEXTBL;
```

```
 * sqlite:///my_data1.db
Done.
```

Launch_Site
CCAFS LC-40
VAFB SLC-4E
KSC LC-39A
CCAFS SLC-40
None

- Leveraging the distinct keyword to avoid duplicates in the names

Launch Site Names Begin with 'CCA'

Task 2

Display 5 records where launch sites begin with the string 'CCA'

```
%%sql

SELECT *
  FROM SPACEXTBL
 WHERE "Launch_Site" LIKE 'CCA%' LIMIT 5
```

```
 * sqlite:///my_data1.db
Done.
```

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
06/04/2010	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0.0	LEO	SpaceX	Success	Failure (parachute)
12/08/2010	15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight C1, two CubeSats, barrel of Brouere cheese	0.0	LEO (ISS)	NASA (COTS) NRO	Success	Failure (parachute)
22/05/2012	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525.0	LEO (ISS)	NASA (COTS)	Success	No attempt
10/08/2012	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500.0	LEO (ISS)	NASA (CRS)	Success	No attempt
03/01/2013	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677.0	LEO (ISS)	NASA (CRS)	Success	No attempt

- Limiting the results to only 5 with the Limit keyword

Total Payload Mass

Task 3

Display the total payload mass carried by boosters launched by NASA (CRS)

```
%%sql

SELECT SUM(PAYLOAD_MASS__KG_)
  FROM SPACEXTBL
 WHERE Customer = 'NASA (CRS)'
```

```
 * sqlite:///my_data1.db
Done.
```

SUM(PAYLOAD_MASS__KG_)
45596.0

- Use SUM to calculate the total

Average Payload Mass by F9 v1.1

Task 4

Display average payload mass carried by booster version F9 v1.1

```
%%sql

SELECT AVG(PAYLOAD_MASS__KG_)
  FROM SPACEXTBL
 WHERE "Booster_Version" = 'F9 v1.1'
```

```
 * sqlite:///my_data1.db
Done.
```

AVG(PAYLOAD_MASS__KG_)
2928.4

- Function AVG calculates the average

First Successful Ground Landing Date

List the date when the first succesful landing outcome in ground pad was acheived.

Hint:Use min function

```
%%sql

SELECT MIN(Date)
  FROM SPACEXTBL
 WHERE "Landing_Outcome" = 'Success (ground pad)'
```

```
 * sqlite:///my_data1.db
Done.
```

MIN(Date)
01/08/2018

- Using min to get the earliest date

Successful Drone Ship Landing with Payload between 4000 and 6000

List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000

```
%%sql

SELECT distinct("Booster_Version")
  FROM SPACEXTBL
 WHERE "PAYLOAD_MASS__KG_" > 4000 AND "PAYLOAD_MASS__KG_" < 6000
       AND "Landing_Outcome" = 'Success (drone ship)'
```

```
 * sqlite:///my_data1.db
Done.
```

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

- Using where to filter the desired payload mass

Total Number of Successful and Failure Mission Outcomes

List the total number of successful and failure mission outcomes

```
%%sql

SELECT count(*), "Mission_Outcome"
  FROM SPACEXTBL
 GROUP By "Mission_Outcome"
```

```
 * sqlite:///my_data1.db
Done.
```

count(*)	Mission_Outcome
898	None
1	Failure (in flight)
98	Success
1	Success
1	Success (payload status unclear)

- Using aggregate query to count the number of rows per Mission Outcome

Boosters Carried Maximum Payload

- Use a nested query to find the maximum payload mass and then distinct to get the unique booster versions

Task 8

List the names of the booster_versions which have carried the maximum payload mass. Use a subquery

```
%%sql

SELECT Distinct("Booster_Version")
  FROM SPACEXTBL
 WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_)
                            FROM SPACEXTBL)
```

```
 * sqlite:///my_data1.db
Done.
```

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

- Using substring to get the respective date parts, case to translate the number to a string

Task 9

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.

Note: SQLLite does not support monthnames. So you need to use substr(Date, 4, 2) as month to get the months and substr(Date,7,4)='2015' for year.

```
%%sql

SELECT CASE
            WHEN substr(Date, 4, 2) = '01' THEN 'January'
            WHEN substr(Date, 4, 2) = '02' THEN 'February'
            WHEN substr(Date, 4, 2) = '03' THEN 'March'
            WHEN substr(Date, 4, 2) = '04' THEN 'April'
            WHEN substr(Date, 4, 2) = '05' THEN 'May'
            WHEN substr(Date, 4, 2) = '06' THEN 'June'
            WHEN substr(Date, 4, 2) = '07' THEN 'July'
            WHEN substr(Date, 4, 2) = '08' THEN 'August'
            WHEN substr(Date, 4, 2) = '09' THEN 'September'
            WHEN substr(Date, 4, 2) = '10' THEN 'October'
            WHEN substr(Date, 4, 2) = '11' THEN 'November'
            WHEN substr(Date, 4, 2) = '12' THEN 'December'
        ELSE 'Ooops'
        END AS "Month",
       "Landing_Outcome",  "Booster_Version" , "Launch_Site"
 FROM SPACEXTBL
WHERE  substr(Date,7,4)='2015'
      AND "Landing_Outcome" = 'Failure (drone ship)'
```

```
 * sqlite:///my_data1.db
Done.
```

Month	Landing_Outcome	Booster_Version	Launch_Site
October	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
April	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

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

Task 10

Rank the count of successful landing_outcomes between the date 04-06-2010 and 20-03-2017 in descending order.

```
%%sql

SELECT count(*) ,"Landing_Outcome"
  FROM SPACEXTBL
 WHERE "Landing_Outcome" LIKE 'Success%'
       AND DATE > '04-06-2010' AND DATE <'20-03-2017'
GROUP BY "Landing_Outcome"
ORDER BY count(*) DESC
```

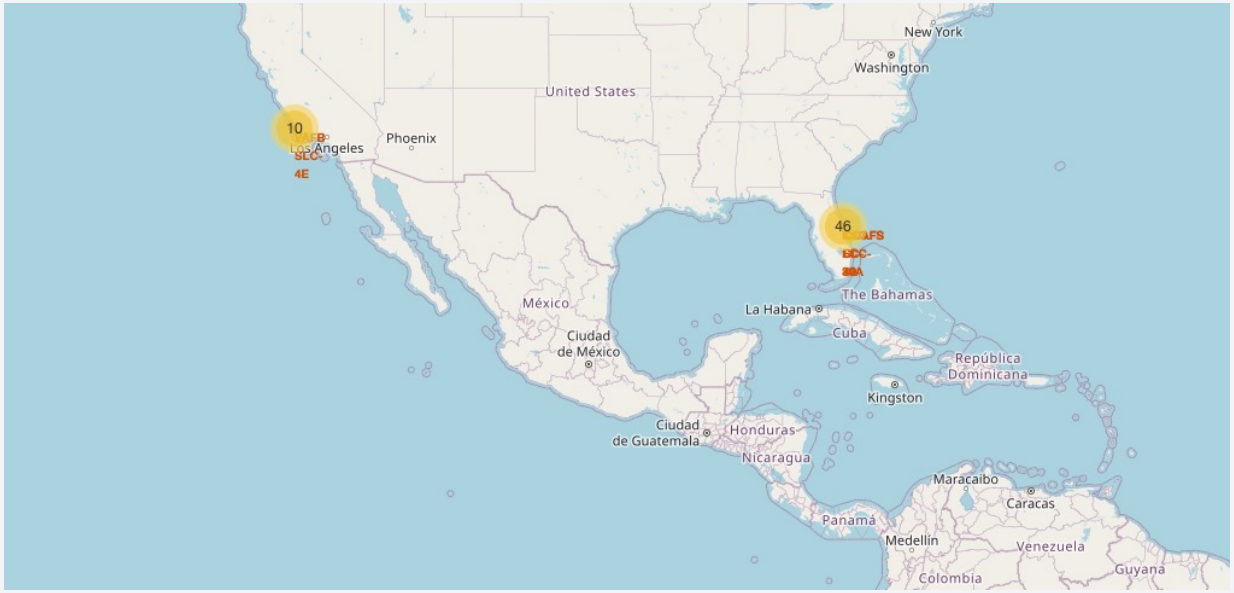
```
 * sqlite:///my_data1.db
Done.
```

count(*)	Landing_Outcome
20	Success
8	Success (drone ship)
7	Success (ground pad)

- Group by landing outcome and counting each result per outcome. Finally sorting desc to achieve the descending order

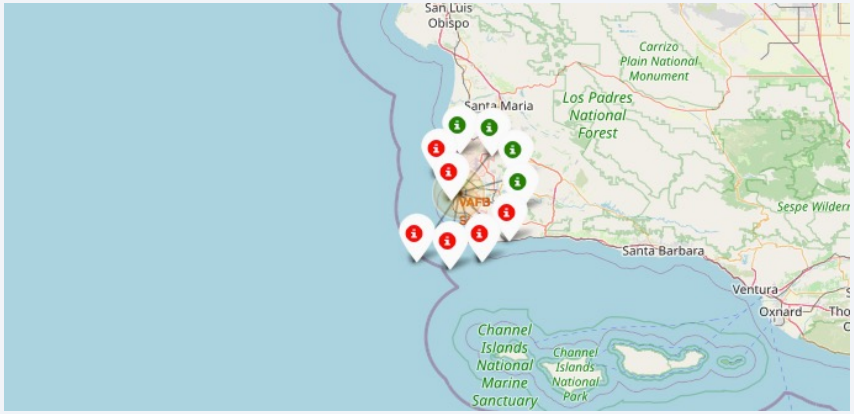

Section 3
Launch Sites
Proximities Analysis

Launch Sites locations and num of launches



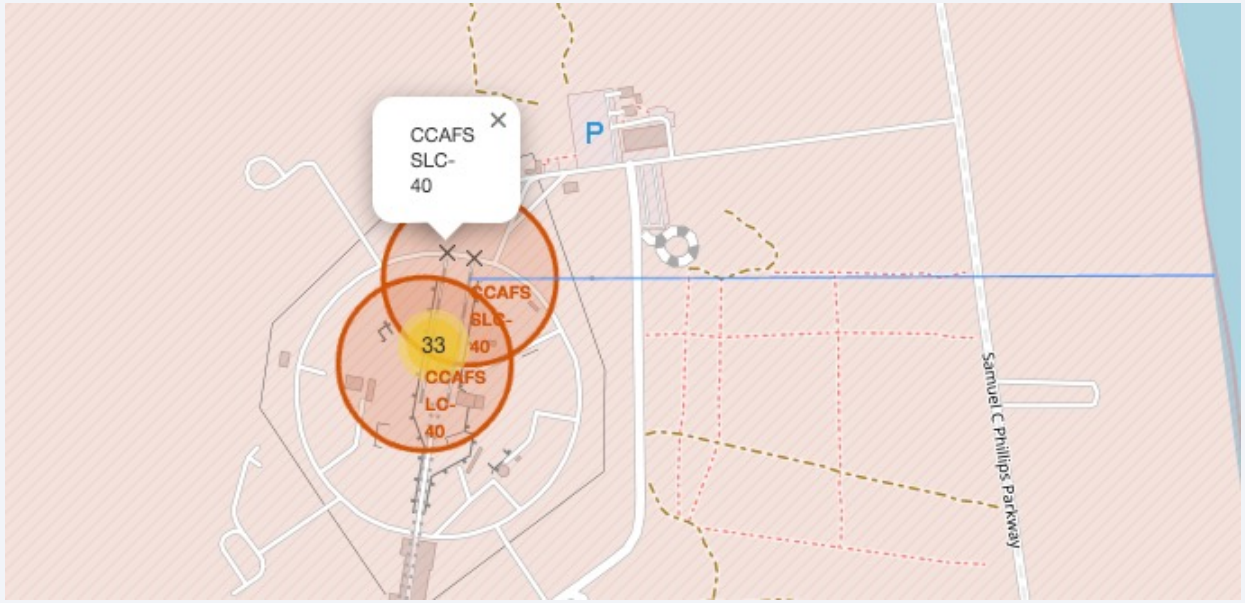
- The markers indicate the Launch site locations
- The numbers indicate the number of launches per location
- Launch sites are next to the sea and also close to big cities

Launch locations and outcome

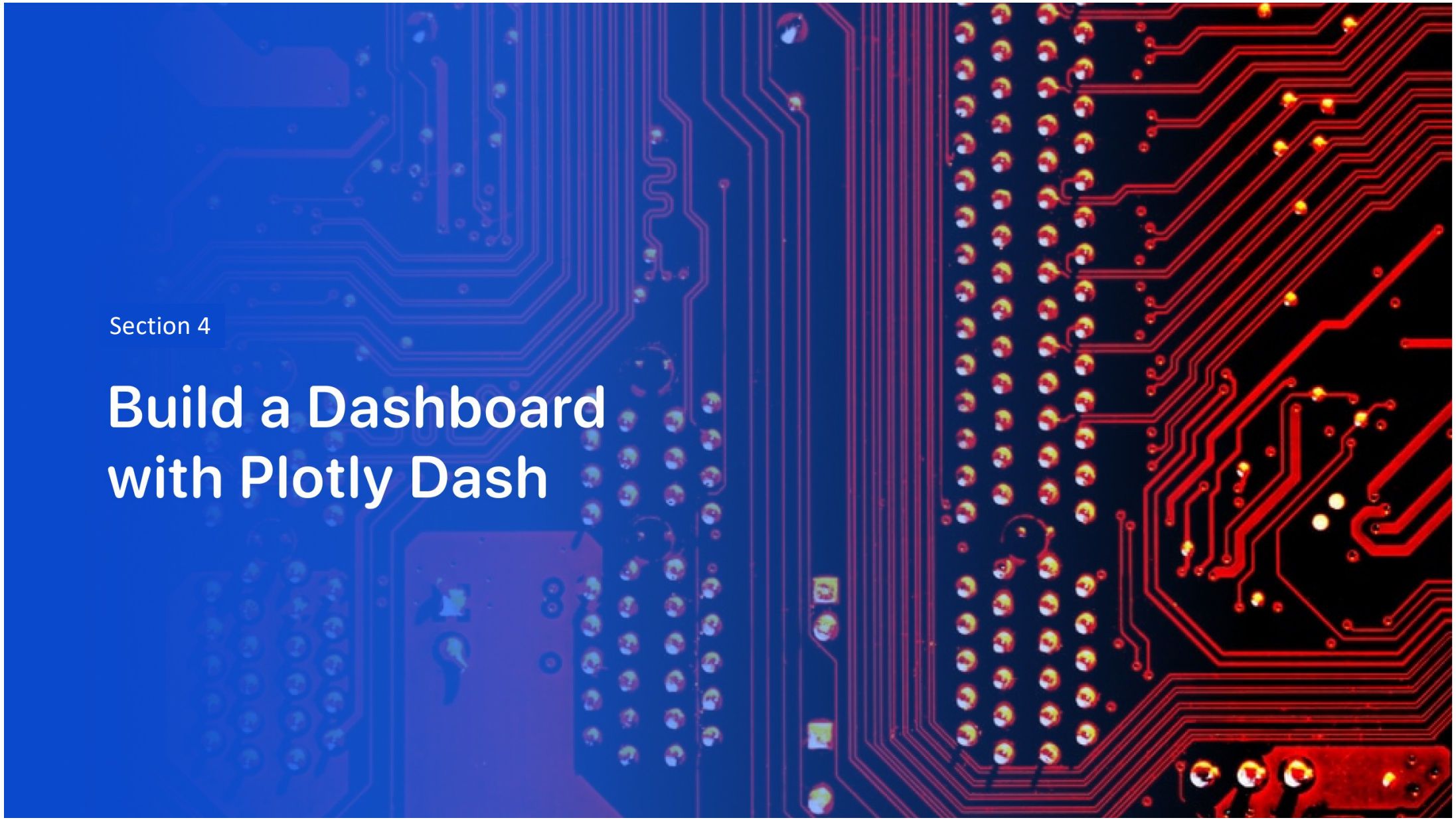


In the above maps we can visualize the launch locations and the success/failure for each launch based on the marker color

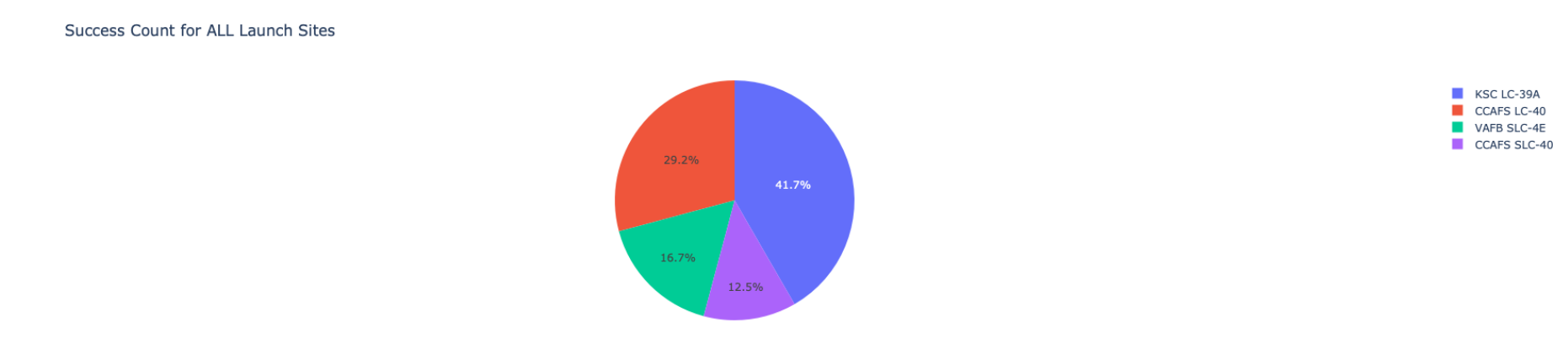
Launch site proximities



- The launch site CCAFS SLC-40 and the distance to the coast

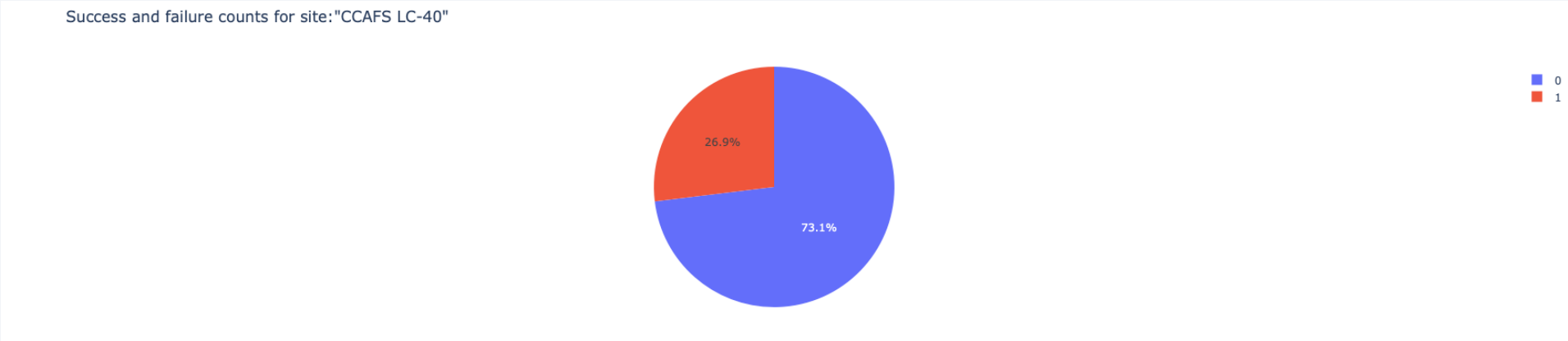

Section 4
Build a Dashboard with Plotly Dash

Launch success for all sites



- KSC LC-39A is the most successful site
- CCAFS SLC-40 is the less successful site

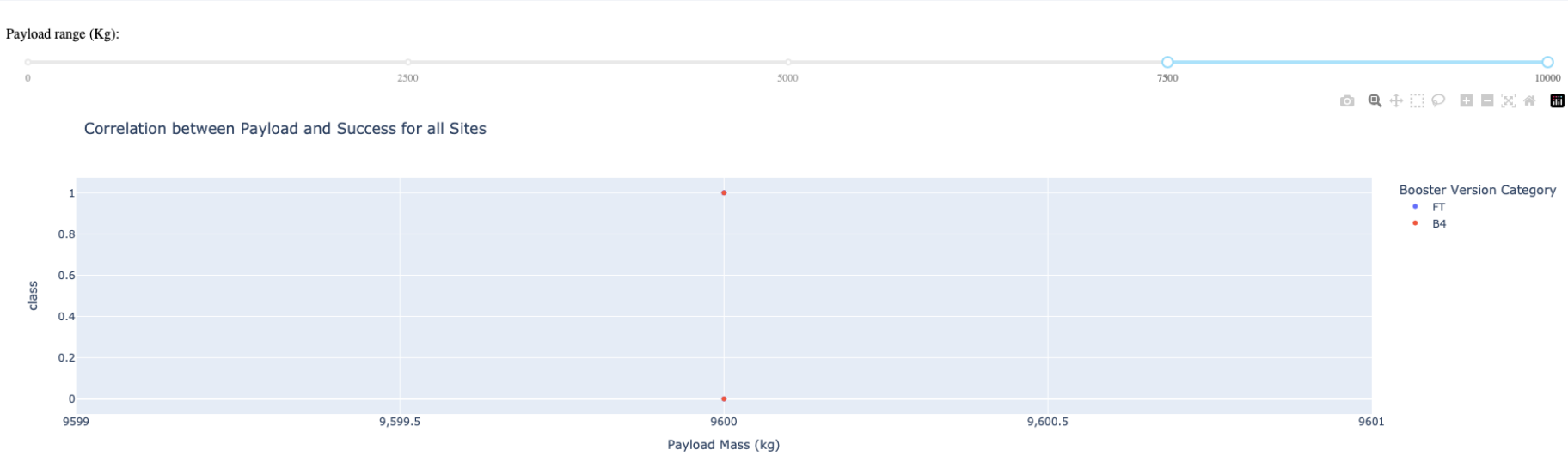
Success vs Failure Ratio for CCAFS LC-40



- Success rate is 73.1%

Payload vs. Launch Outcome scatter plot all sites

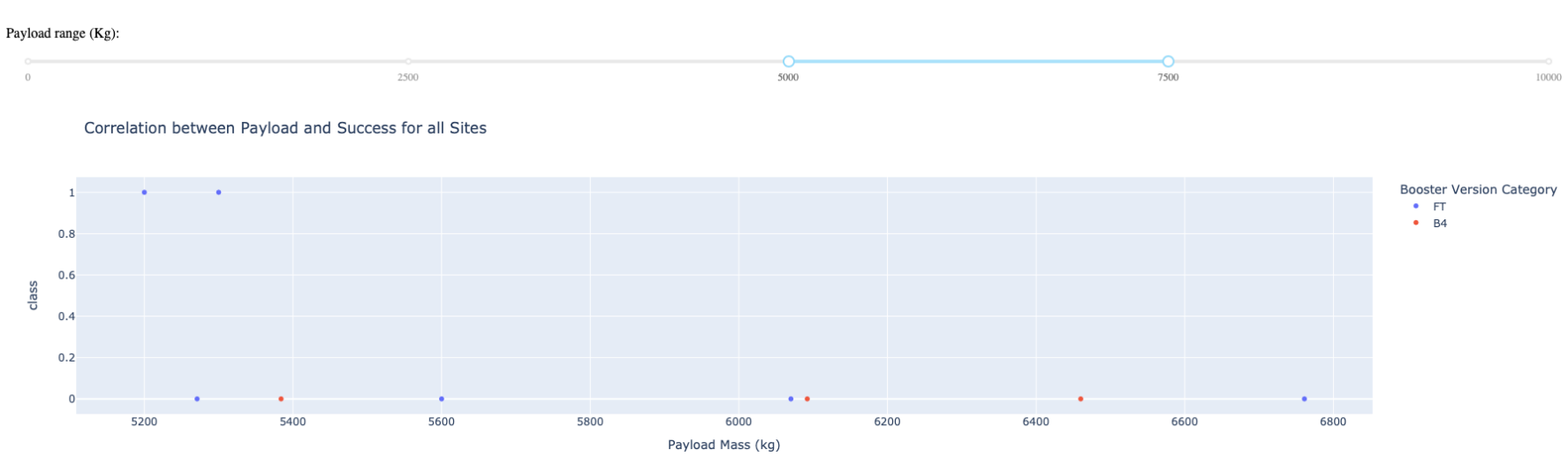
Payload 7500 Kg – 10000 Kg



- B4 has 50% success
- No data for FT

Payload vs. Launch Outcome scatter plot all sites (cont.)

Payload 5000 kg – 7500 kg



- B4 has 0% success
- FT has 50% success
- Overall we observe more failures in this range

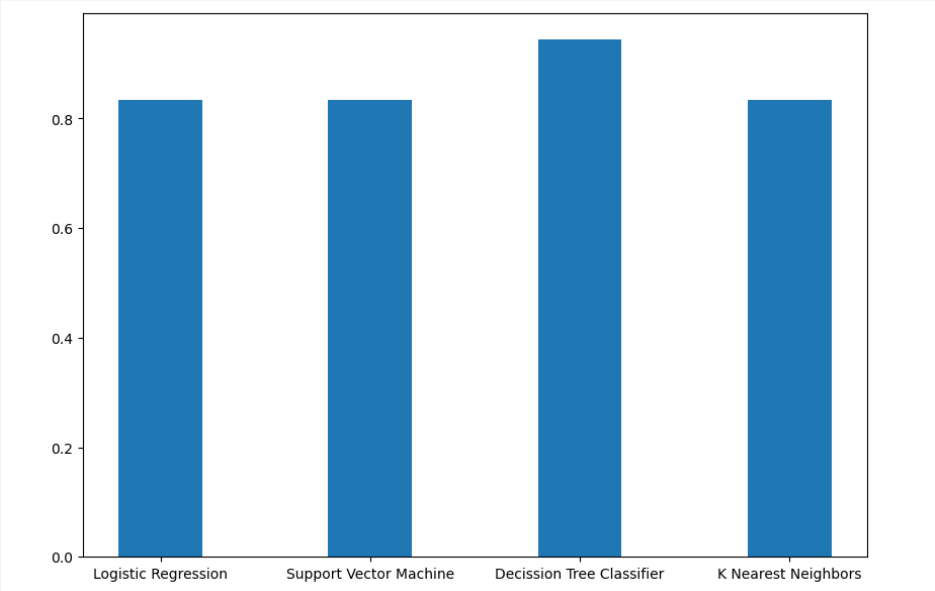
Section 5

Predictive Analysis (Classification)

Classification Accuracy

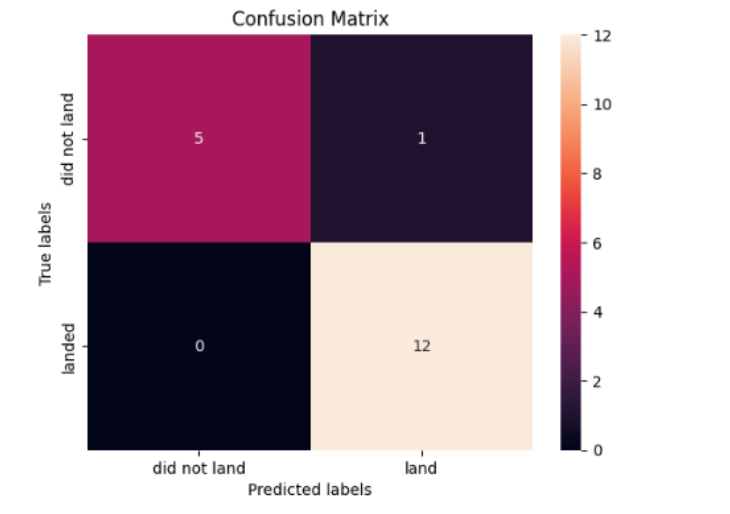
- The bar chart visualizes the model accuracy for all built classification models

- The model with the highest accuracy is the “Decision Tree Classifier”



Confusion Matrix Decision Tree Classifier

- The Best performing model is the Decision Tree Classifier
- Great performance in True positive and true negative
- Predicts false positive



Conclusions

- We can leverage existing open data to generate models in order to achieve our goal (Predict first stage landing)
- Several parameters can influence the outcome
  - Payload
  - Orbit type
  - Launch location
  - Number of launches
- Based on the existing data, we can leverage several models in order to predict based on the parameters, whether the first stage will land.
- All the models perform great with regards to True Positive and True Negative
- The optimal model to predict whether the first stage will land is the “Decision Tree Classifier”

Appendix

- Space X Rest API URL: https://api.spacexdata.com/v4
- Wikipedia page: https://en.wikipedia.org/wiki/List_of_Falcon_9_and_Falcon_Heavy_launches

Thank you!