IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Henry Etheridge
18th September 2022

Outline

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

Executive Summary

- Summary of methodologies
  - Data collection using APIs and web scraping
  - Wrangling the data to work with analysis techniques
  - Exploratory data analysis
  - Interactive tools using Folium and Plotly
  - Predictive analysis
- Summary of all results
  - Launch site proximity analysis
  - Launch probability by site and payload mass
  - Predictive analysis outcome

Introduction

- Project background and context
  - SpaceX provides among the cheapest rocket launches using the Falcon 9 rocket. This is mainly due to the ability to reuse the first stage of the rocket, which does most of the work in getting the payload to space. As a competing rocket company, we aim to determine the price of a SpaceX rocket launch. A big factor of this cost is whether the first stage can be reused after a launch. Hence, we aim to develop a machine learning model using available data to determine whether the first stage will land safely after a given launch.
- Problems you want to find answers
  - What is the cost of a SpaceX rocket launch?
  - Will SpaceX reuse the first stage of a rocket launch?
  - What is the probability of the first stage of the rocket landing successfully, such that it can be reused?
  - What factors affect the probability of the first stage landing successfully?

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Data was collected in two ways; from the SpaceX website using an API, and scraped from Wikipedia
- Perform data wrangling
  - Landing outcomes were converted to binary values, 1 = success, 0 = failure
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - How to build, tune, evaluate classification models

Data Collection

- The outcome of Falcon 9 heavy rocket launches were gathered through two sources; from the SpaceX web using an API, and from Wikipedia by web scraping.
- API: use a get request to the SpaceX API to get a list of JSON objects, each corresponding to a launch event. This is then converted to a Pandas dataframe using json.normalize(). This was then cleaned, including dealing with missing values.
- Web scraping: Using BeautifulSoup, launch records were extracted from a Wikipedia table as a HTML table. This was then converted to a dataframe.

Data Collection – SpaceX API

- Launch data is collected from the SpaceX website using the SpaceX REST API using a get request.

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

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

- The list of JSON objects is converted to a Pandas dataframe using json.normalize()

```
# Use json_normalize meethod to convert the json result into a dataframe
response = requests.get(static_json_url)
data = pd.json_normalize(response.json())
```

- The variables are checked, and missing values are filled with either Nan or 0

```
# Lets take a subset of our dataframe keeping only the features we want and the flight number, and date_utc.
data = data[['rocket', 'payloads', 'launchpad', 'cores', 'flight_number', 'date_utc']]

# We will remove rows with multiple cores because those are falcon rockets with 2 extra rocket boosters and rows that have
data = data[data['cores'].map(len)==1]
data = data[data['payloads'].map(len)==1]

# Since payloads and cores are lists of size 1 we will also extract the single value in the list and replace the feature.
data['cores'] = data['cores'].map(lambda x : x[0])
data['payloads'] = data['payloads'].map(lambda x : x[0])

# We also want to convert the date_utc to a datetime datatype and then extracting the date leaving the time
data['date'] = pd.to_datetime(data['date_utc']).dt.date

# Using the date we will restrict the dates of the launches
data = data[data['date'] <= datetime.date(2020, 11, 13)]
```

- Code - https://github.com/HenryEtheridge/DataScienceFinalProject/blob/main/Data%20Collection%20API.ipynb

Data Collection - Scraping

- Web scrape Falcon 9 launch records from Wikipedia as a HTML table using BeautifulSoup.
- Parse the table and convert it to a Pandas dataframe for further data wrangling.
- Code - https://github.com/HenryEtheridge/DataScienceFinalProject/blob/main/Web%20Scraping.ipynb

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

```
# use requests.get() method with the provided static_url
# assign the response to a object
response = requests.get(static_url)
```

Create a `BeautifulSoup` object from the HTML `response`

```
# Use BeautifulSoup() to create a BeautifulSoup object from a response text content
soup = BeautifulSoup(response.text)
```

Data Wrangling

- Data were processed to categorize all launches as either a “success” in which the first stage landed safely, and “unsuccessful” such that the first stage did not land. This allows the result to be saved as a binary output (1 for success, 0 for failure) which is useful for future data analysis and machine learning techniques as the target variable.

- Code - https://github.com/HenryEtheridge/DataScienceFinalProject/blob/main/Data%20Wrangling.ipynb

```
# landing_class = 0 if bad_outcome
# landing_class = 1 otherwise

landing_class = ~df.Outcome.isin(bad_outcomes)*1

landing_class
```

EDA with Data Visualization

- The following charts were plotted to visualize any relationship between two variables. The primary goal here was to determine which features had the largest affect on the success rate to be used in further analysis and models.
  - Payload mass and flight number as a scatter chart
  - Flight number and launch site as a scatter chart
  - Payload and launch site as a scatter chart
  - Success rate orb each possible orbit as a bar chart
  - Orbit and flight number, colored by success or failure as a scatter chart
  - Payload and orbit type, colored by success or failure as a scatter chart
  - Probability of success against launch year as a line chart
- Code - https://github.com/HenryEtheridge/DataScienceFinalProject/blob/main/Visualisation.ipynb

EDA with SQL

- SQL queries were performed to further explore the dataset, and identify
  - The names of the launch sites
  - 5 records where the launch sites begin with the string “CCA”
  - The total payload mass carried by NASA boosters
  - The average payload mass carried by booster version F9 v1.1
  - The date when the first successful landing outcome in ground pad occurred
  - The names of boosters which have successful drone ship landings and payload mass between 4000 and 6000
  - The total number of successful and unsuccessful mission outcomes
  - The names of the booster versions which have carried the maximum payload mass
  - The failed landing outcomes in droneship landings, their booster versions and the launch site names in the 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
- Code - https://github.com/HenryEtheridge/DataScienceFinalProject/blob/main/SQL%20Exploratory%20Analysis.ipynb

Build an Interactive Map with Folium

- We added circles and markers to identify the locations of launch sites, labelled with the site name. This allows us to identify that launch sites are near the coasts and towards the equator.
- The markers are colored by whether the launch was a success or failure in a marker cluster.
- We then calculated the distance from these launch sites to various features to identify whether launch sites are near railways, highways, coastlines or cities.
- Code - https://github.com/HenryEtheridge/DataScienceFinalProject/blob/main/Interactive%20Visual%20Analytics%20(1).ipynb

Build a Dashboard with Plotly Dash

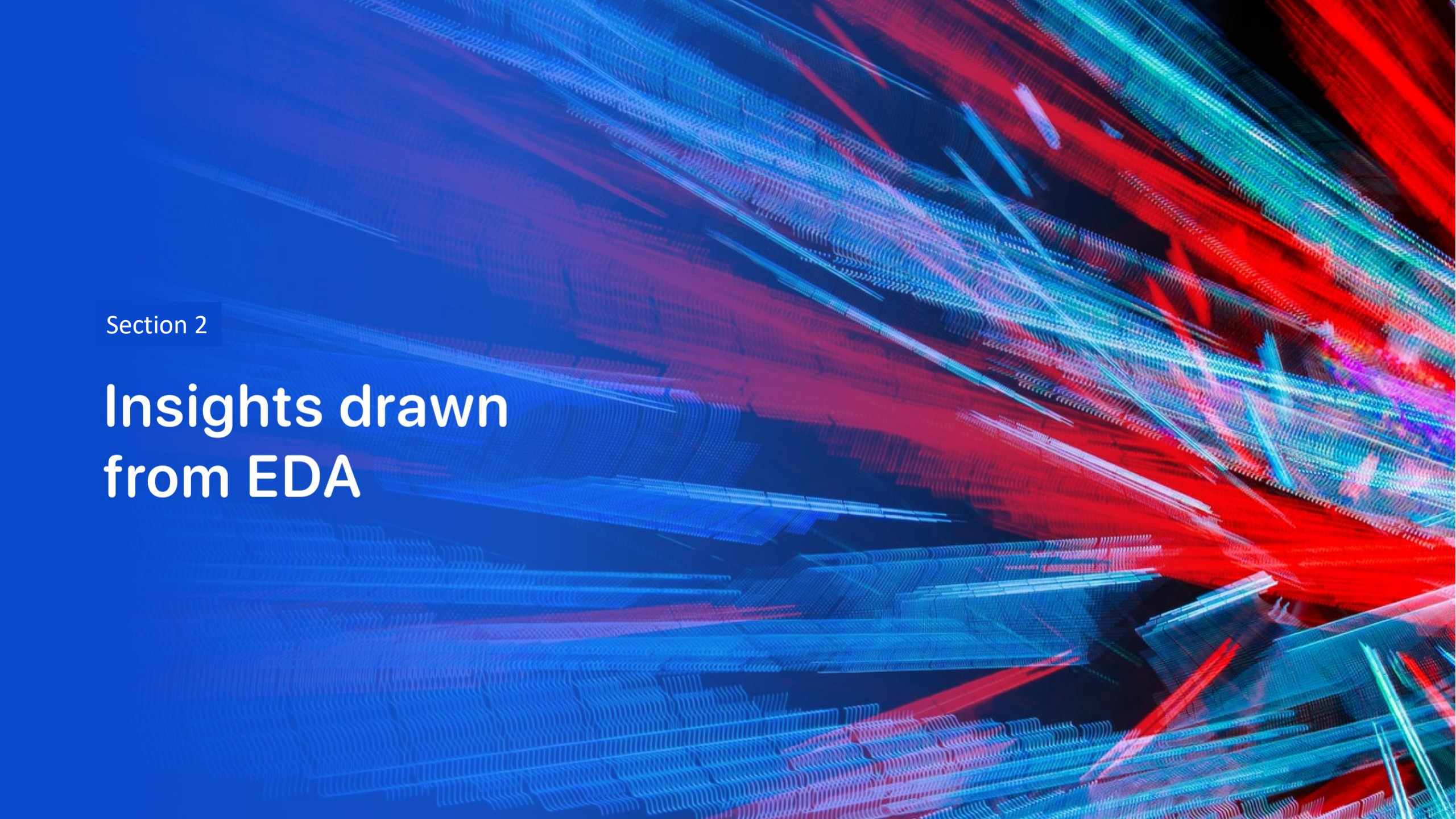
- We created visualizations to answer 5 questions
  - Which site has the largest successful launches?
  - Which site has the highest launch success rate?
  - Which payload range has the highest/lowest success rate?
  - Which F9 boosted version has the highest success rate?
- For this, we created a dropdown box to allow for the launch site/s and payload range to be selected.
- We added a Pie chart showing the total launches per site and success rate for a single site and a scatter chart of outcome and payload mass for the different booster versions.

Predictive Analysis (Classification)

- The data was converted from a Pandas dataframe to a Numpy matrix, standardized using the preprocessing standard scaler and then split into a training set and a test set.
- We then trained a variety of machine learning models, including logistic regression, support vector machine, decision tree classifier and k-nearest neighbors.
- The accuracy of these models is tested using the test data set to calculate the best_score parameter as well as a confusion matrix.
- The models are optimized through a grid search method across the hyperparameter space.
- The best performing model will be the model with the best accuracy after optimization.
- Code - https://github.com/HenryEtheridge/DataScienceFinalProject/blob/main/Machine%20Learning%20Prediction%20(1).ipynb

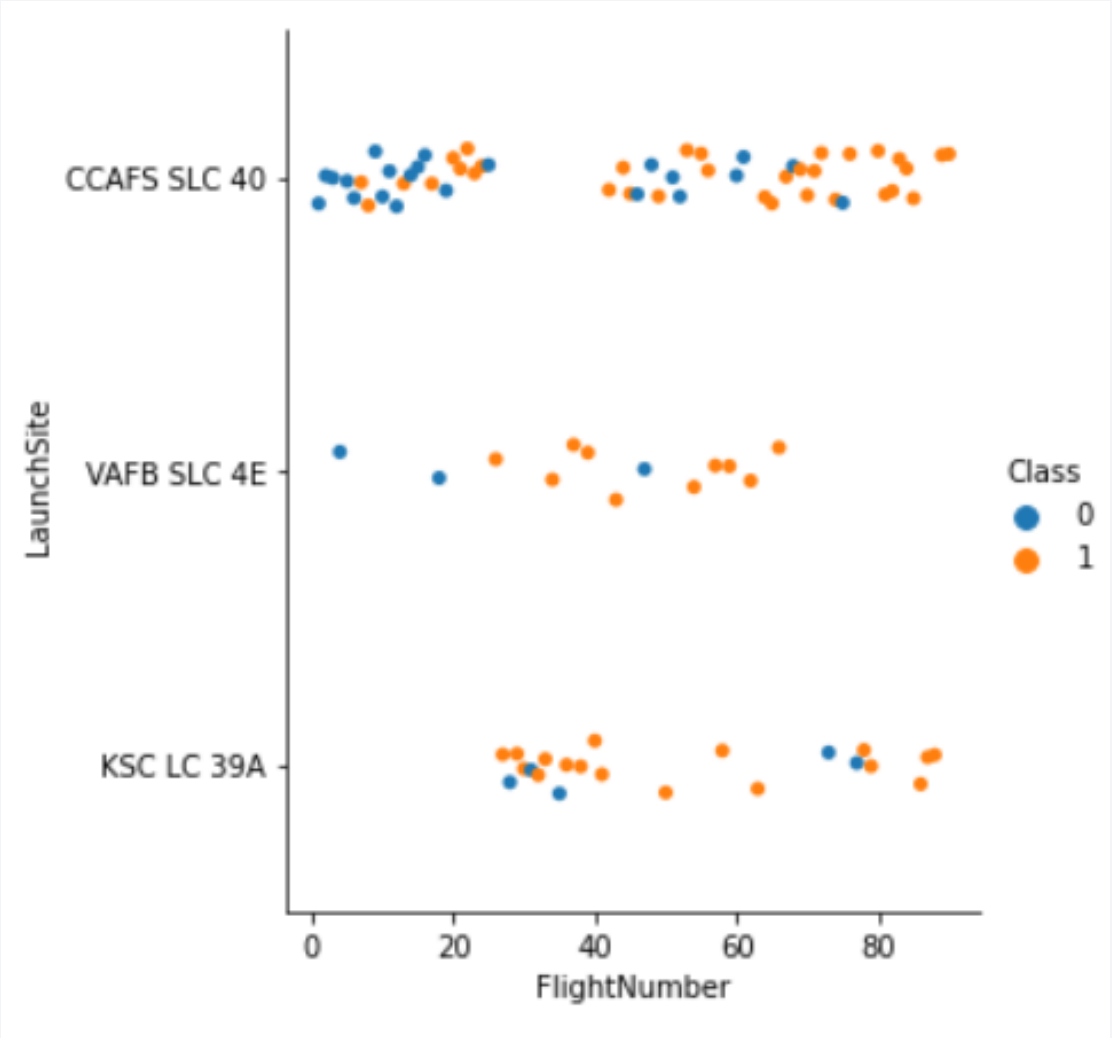
Results

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


Section 2
Insights drawn from EDA

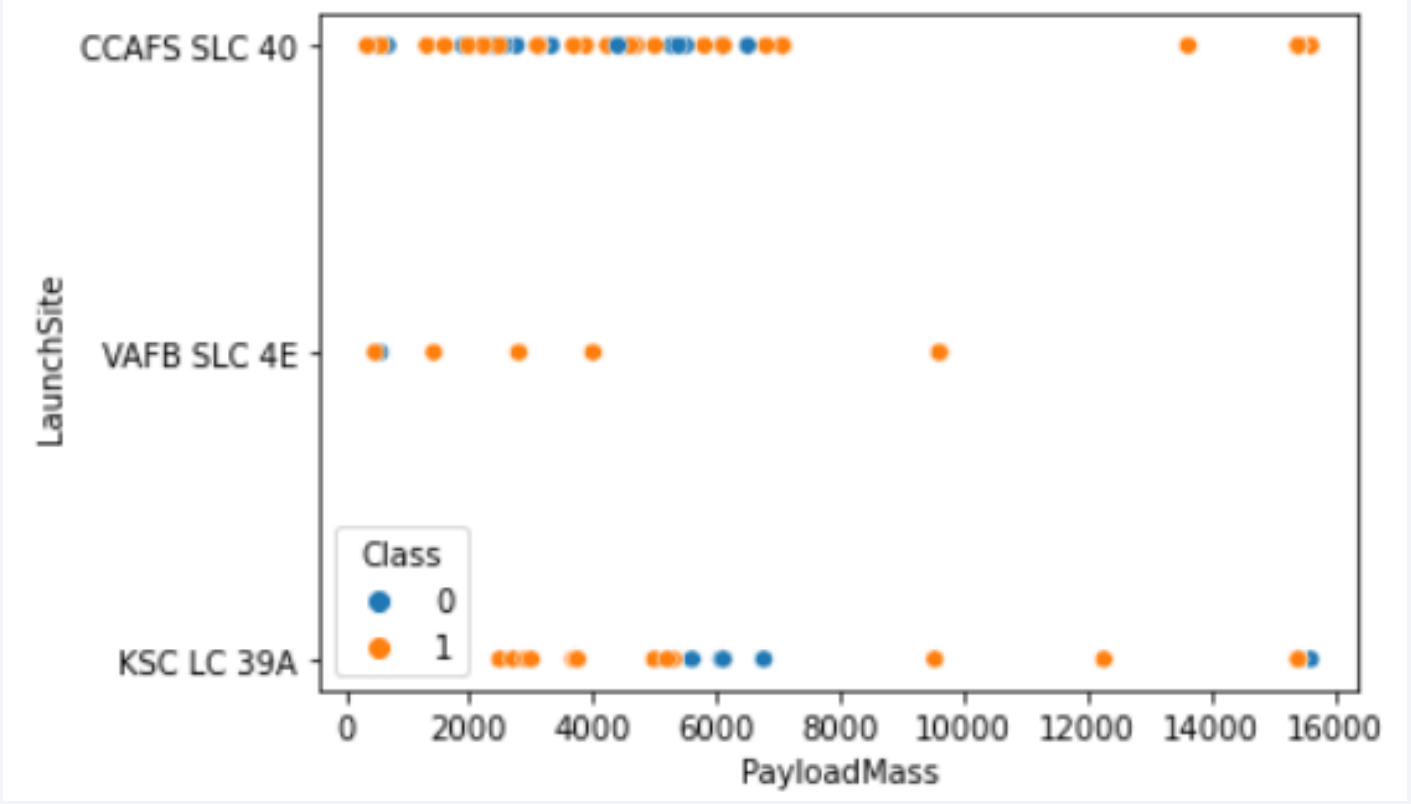
Flight Number vs. Launch Site

- A scatter plot of Flight Number vs. Launch Site. The colour relates to a successful (1) or unsuccessful (0) first stage landing.
- We can see that different sites have been used at different rates, with KSC only beginning launches at roughly flight number 25.
- The probability of success increases with increasing flight number.



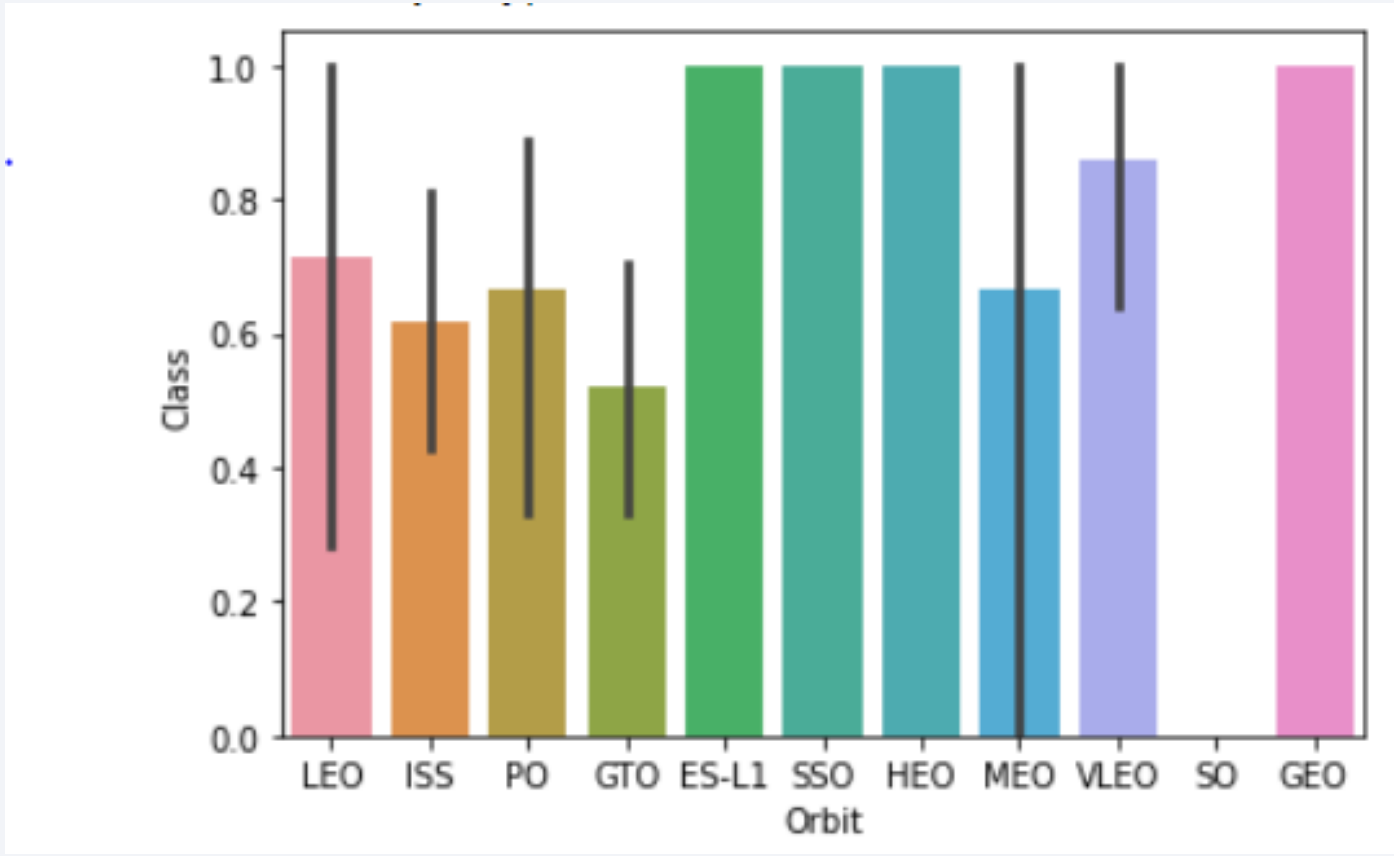
Payload vs. Launch Site

- A scatter plot of Payload vs. Launch Site. The colour relates to a successful (1) or unsuccessful (0) first stage landing.
- Most launches have used smaller payloads.
- No clear relationship between payload mass and success or launch site.



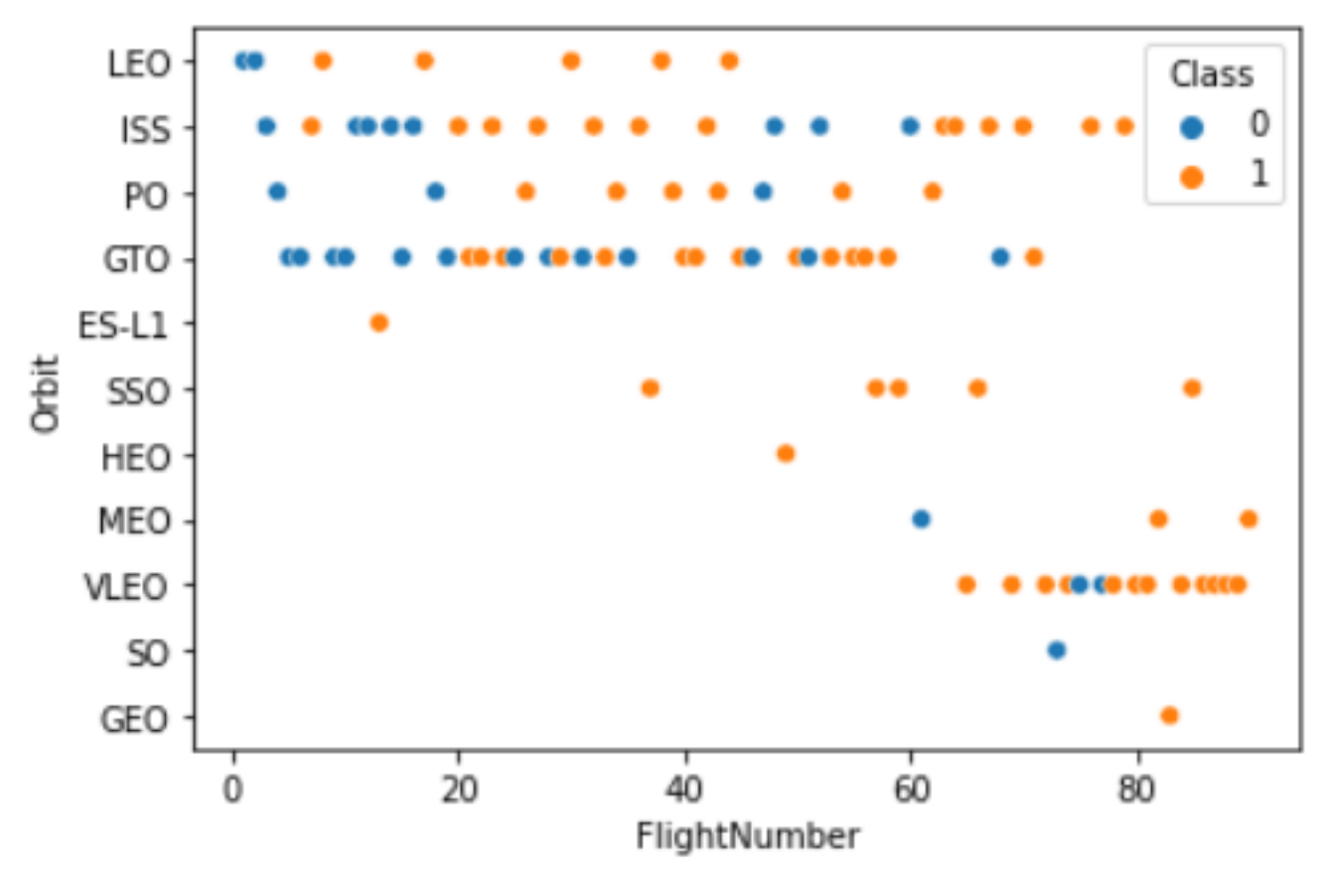
Success Rate vs. Orbit Type

- A bar chart for the success rate (Class) of each orbit type.
- Likely a relationship between orbit type and success rate, although with large uncertainty.
- Several orbit types have 100% success rate, and one has 0%, however this has no demonstration of sample size.



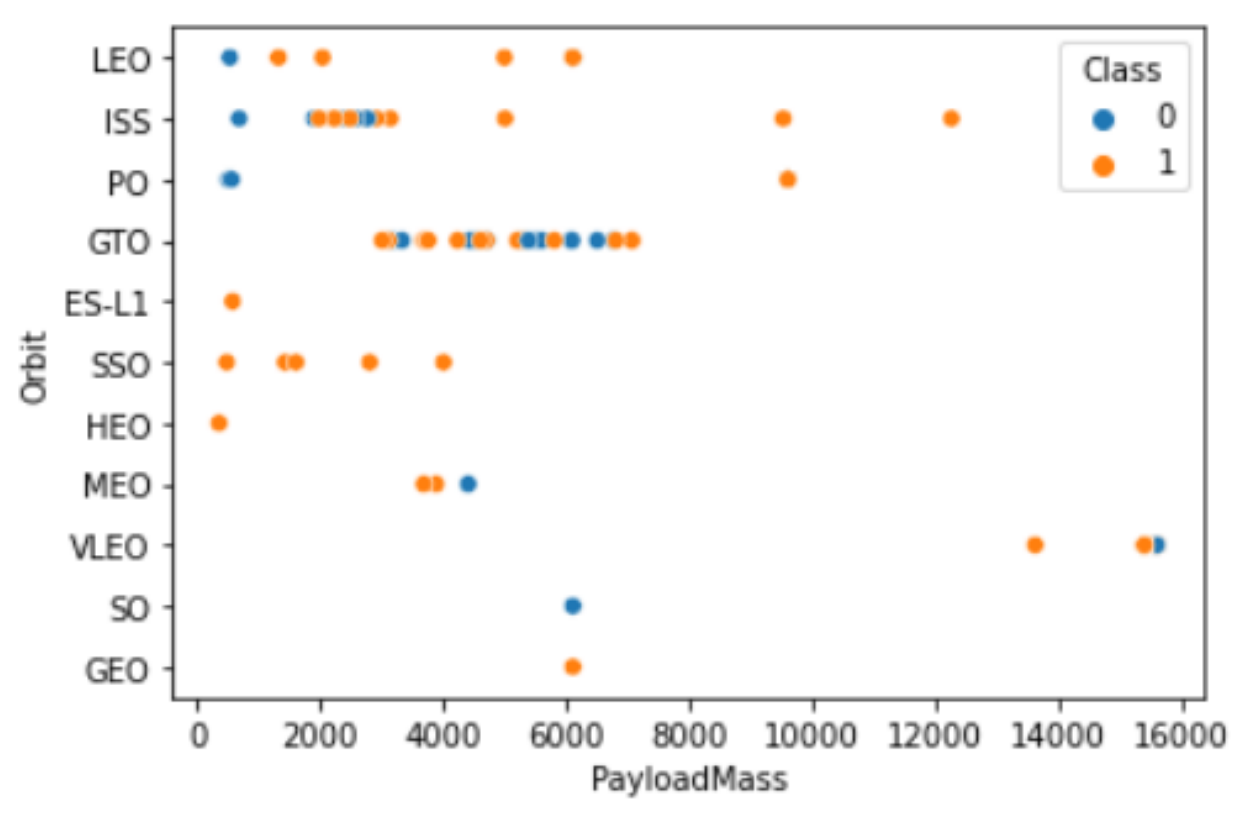
Flight Number vs. Orbit Type

- A scatter plot of Flight number vs. Orbit type, with class as colour.
- There is an increasing variety of orbit type with increasing flight number, however several types, e.g. LEO, have not been performed for many launches.
- Success rate strongly varies with orbit type and with flight number



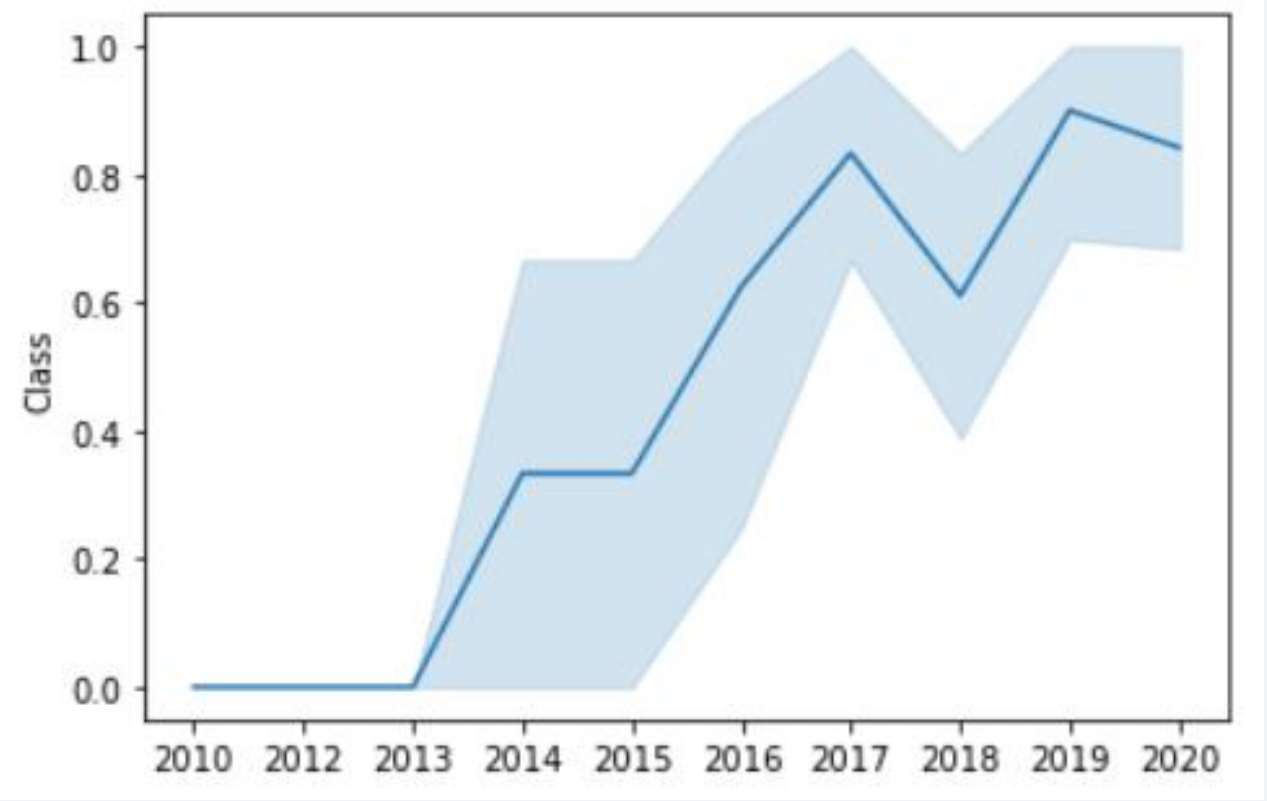
Payload vs. Orbit Type

- A scatter plot of payload vs. orbit type, with colour as class.
- Certain ranges of payload mass have only been used for certain orbits. E.g., GTO only between 3k and 8k mass.
- Lighter masses appear to have a lower success rate within a target orbit.



Launch Success Yearly Trend

- A line chart of yearly average success rate
- Strong positive relationship between success and year.
- No successes until 2014, increases to about 80% in 2017. Minimal change observed over the next few years.



All Launch Site Names

- An SQL query was used to list the unique launch sites used in the space mission
- Unique was used to filter the list of site names in the SPACEXTBL to return only unique entries

```
%%sql
SELECT unique(launch_site) FROM SPACEXTBL
```

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

Launch Site Names Begin with 'CCA'

- SQL query to find 5 launches from sites beginning with CCA
- Uses LIKE with % to match site names starting with CCA and then anything afterwards.
- LIMIT limits the returned items to the requisite number.

```
%%sql
SELECT * FROM SPACEXTBL
WHERE launch_site LIKE 'CCA%'
LIMIT 5
```

DATE	time_utc_	booster_version	launch_site	payload
2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit
2010-12-08	15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight C1, two CubeSats, barrel of Brouere cheese
2012-05-22	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2
2012-10-08	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1
2013-03-01	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2

Total Payload Mass

- SQL query to return the total payload mass carried by all boosters launched by NASA within the dataset – 45596.
- Uses SUM to aggregate all results into one total.
- WHERE clause filters to only those where the customer is NASA.

```
%%sql
SELECT sum(payload_mass__kg_) FROM SPACEXTBL
WHERE customer = 'NASA (CRS)'
```

```
45596
```

Average Payload Mass by F9 v1.1

- SQL query to return the average payload mass carried by F9 v1.1 boosters – 2928.
- Uses AVG to aggregate all results into one total.
- WHERE clause filters to only those where the booster version is F9 v1.1.

```
%%sql
SELECT AVG(payload_mass__kg_) FROM SPACEXTBL
WHERE booster_version = 'F9 v1.1'
```

```
2928
```

First Successful Ground Landing Date

- SQL query to return the first successful ground landing data – 2015-12-22.
- MIN function returns only the earliest date.
- WHERE clause filters to only successful ground pad landings.

```
%%sql
SELECT min(DATE) FROM SPACEXTBL
WHERE landing__outcome = 'Success (ground pad)'
```

2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

- SQL query to identify booster versions which have successfully landed on a drone ship with payload mass between 4000 and 6000.
- Filters for payload mass BETWEEN 4000 and 6000, landing outcome is success on a drone ship.
- Returns a UNIQUE list of booster versions that fulfil the above criteria

```
%%sql
select unique(booster_version) from SPACEXTBL
WHERE landing__outcome = 'Success (drone ship)'
AND
payload_mass__kg_ BETWEEN 4000 and 6000
```

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

Total Number of Successful and Failure Mission Outcomes

- SQL query to return the total number of successful (100, although 1 with unclear payload status) and failure (1) mission outcomes.
- Returns the count of mission outcomes by using the GROUP BY clause over the mission outcomes.

mission_outcome	2
Failure (in flight)	1
Success	99
Success (payload status unclear)	1

```
%%sql
SELECT mission_outcome, count(mission_outcome) FROM SPACEXTBL
GROUP BY mission_outcome
```

Boosters Carried Maximum Payload

- SQL query to return booster versions which have carried the maximum payload.
- Uses a subquery to filter out launches where the payload mass is not the MAX.
- Returns a UNIQUE list of boosters that are not filtered out.

```
%%sql
SELECT unique(booster_version) from SPACEXTBL
WHERE payload_mass__kg_ =
        (SELECT max(payload_mass__kg_) from SPACEXTBL)
```

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

2015 Launch Records

- SQL query to return the failed landing outcomes in 2015.
- Uses WHERE to match the YEAR of the date to 2015, and the mission outcome to NOT success.

```
%%sql
SELECT DATE, booster_version, launch_site, landing__outcome from SPACEXTBL
WHERE YEAR(DATE) = '2015' AND NOT mission_outcome = 'Success'
```

DATE	booster_version	launch_site	landing__outcome
2015-06-28	F9 v1.1 B1018	CCAFS LC-40	Precluded (drone ship)

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

- SQL query returning an ordered list of the count of landing outcomes between 2010-06-04 and 2017-03-20, in descending order.
- Uses WHERE to include only dates within this range. Groups by landing outcome, and then returns the count of each outcome ordered into descending order.

```
%%sql
SELECT landing__outcome, count(landing__outcome) FROM SPACEXTBL
WHERE DATE between '2010-06-04' AND '2017-03-20'
GROUP BY landing__outcome
ORDER BY count(landing__outcome) DESC
```

landing__outcome	2
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

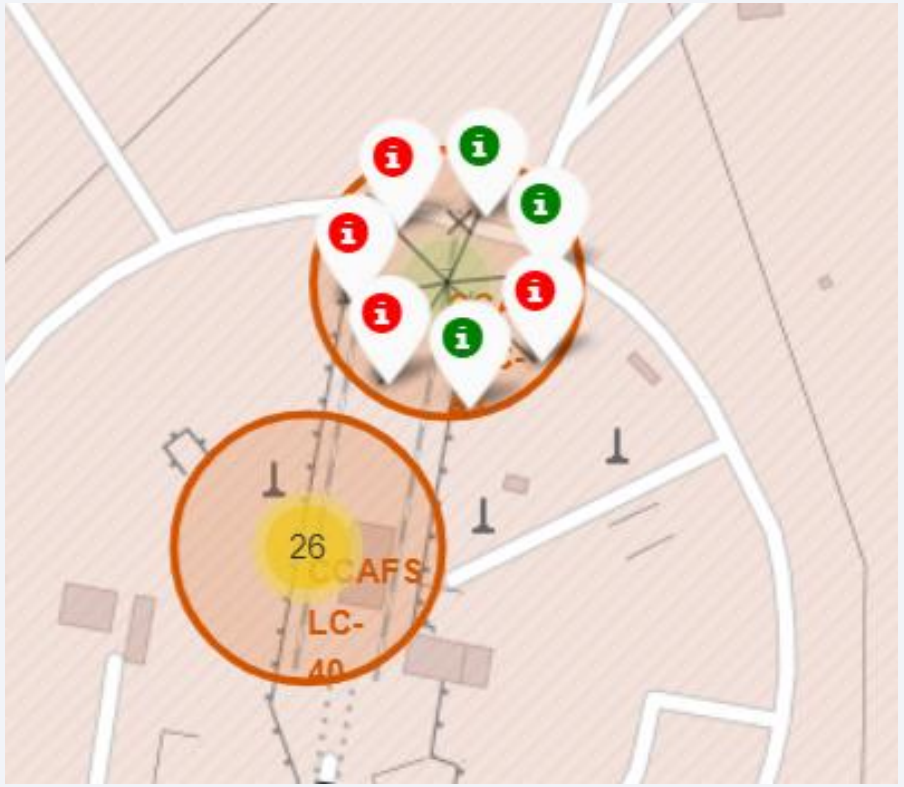
Launch Sites Proximities Analysis

Location of Launch Sites

- A map showing the location of SpaceX launch sites used within the dataset.
- Launch sites are all at coastal sites within the USA.

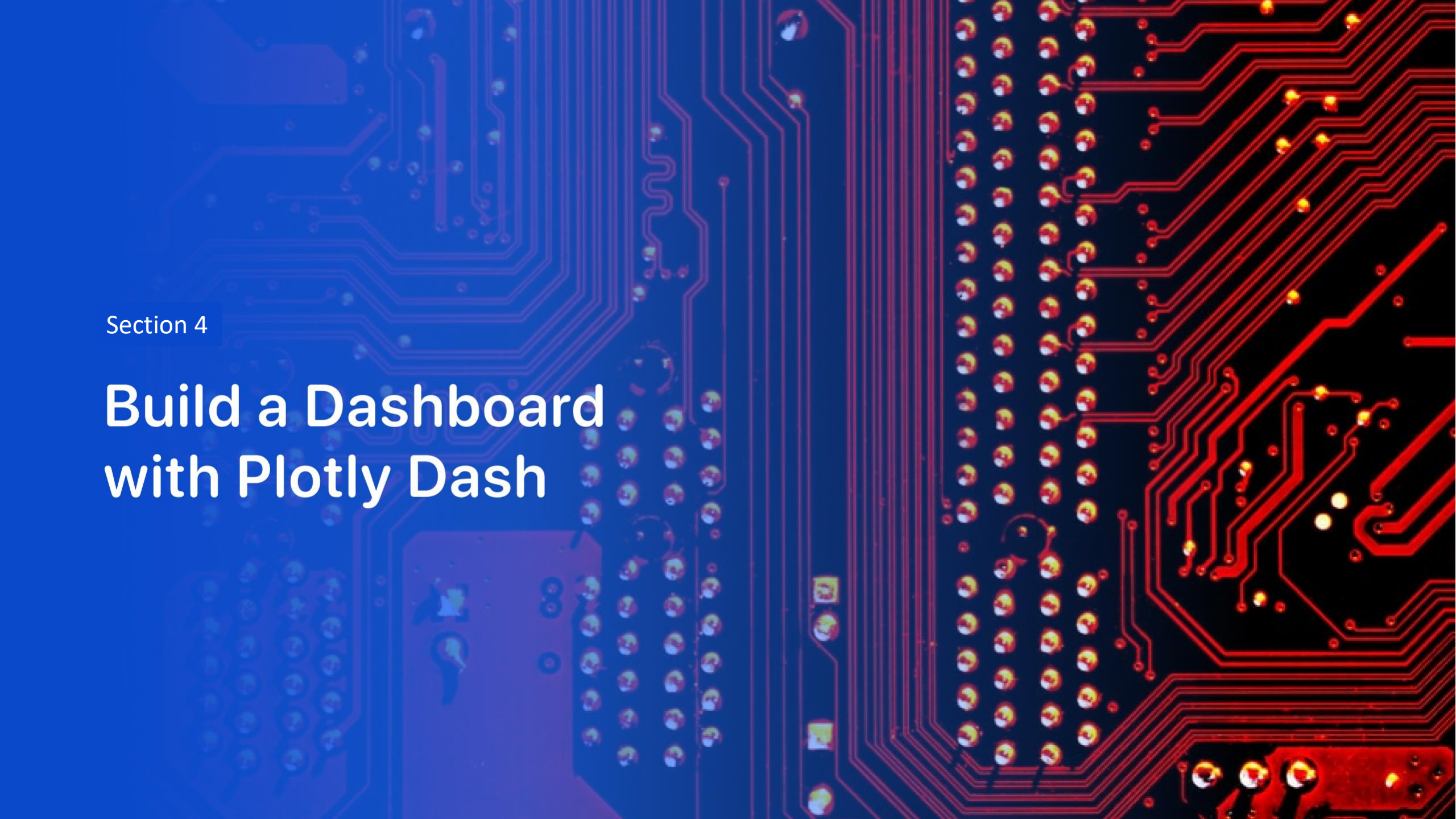
Mapped launch sites with success labels

- An example of a launch site, CCAFS SLC-40, with markers representing an individual launch and whether it was successful (green) or unsuccessful (red).
- The orange circles each represent a single launch site. For the lower site, the number is a cluster that denotes the total number of launches. For the upper site, the cluster has been selected and each launch is shown individually.



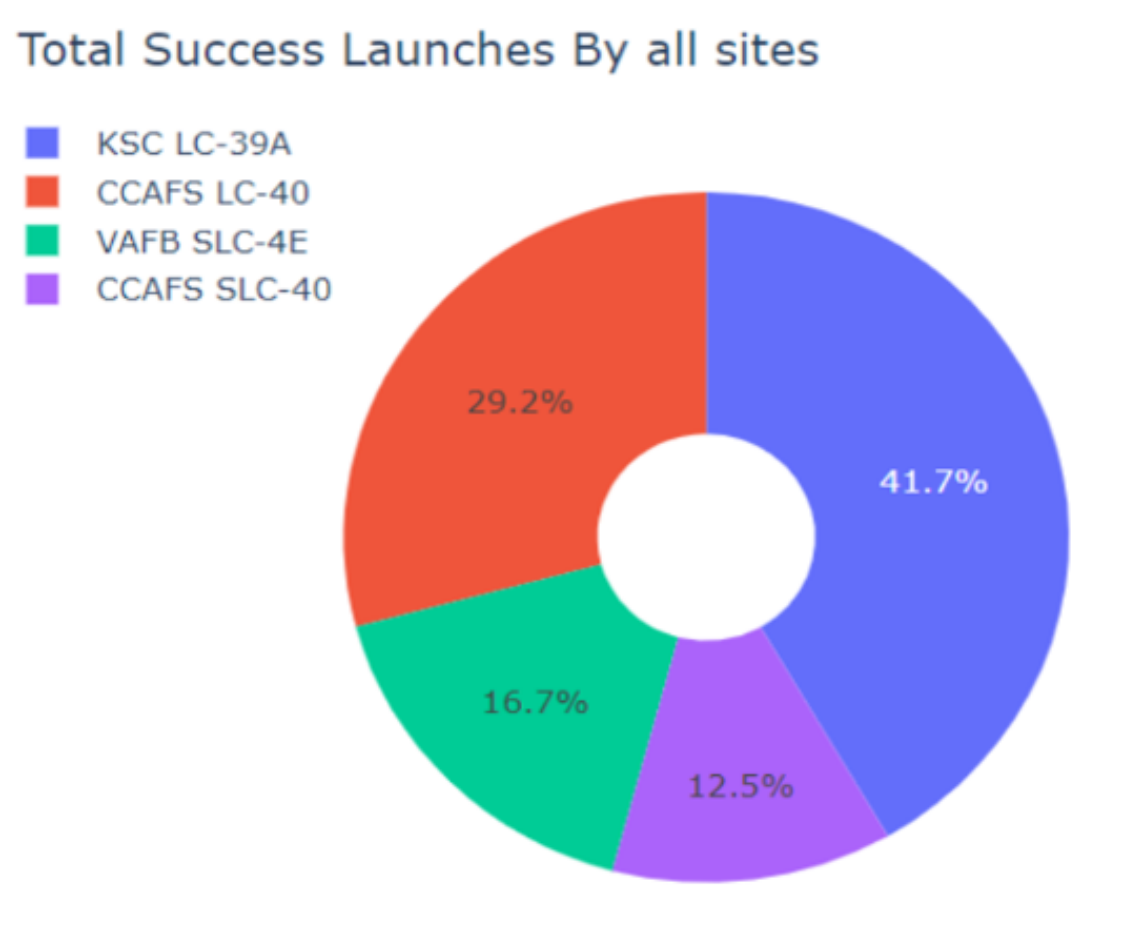
Distance between launch site and landmarks

- Distance between the CCAFS SLC-40 launch site and two neighbouring features; the city Orlando and the nearest highway.
- The city is approx. 80 km away, while the highway is approx. 30 km away.
- This uses a polyline to mark the distance using the coordinates of the features as the end points. The label is added using a marker with the calculated distance.


Section 4
Build a Dashboard with Plotly Dash

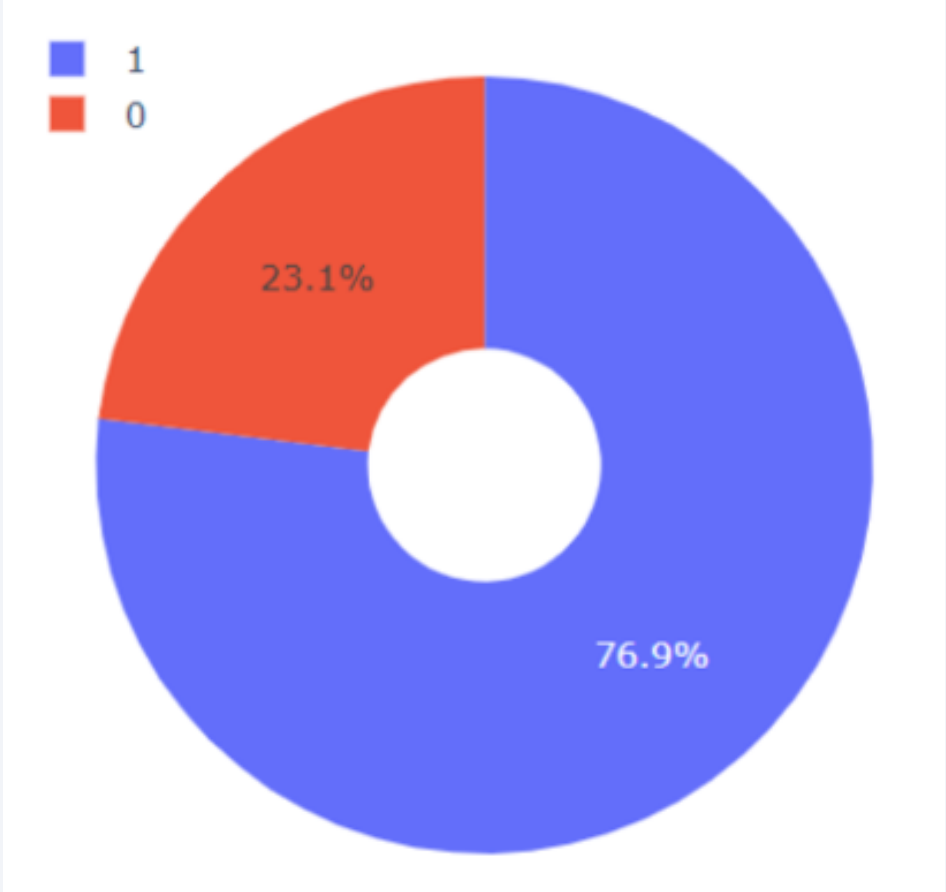
Distribution of Successful Launches by Launch Site

- Successful launches have occurred at 4 launch sites.
- The highest number of successful launces have been from KSC LC-39A. However, this does not show the probability of a successful launch.



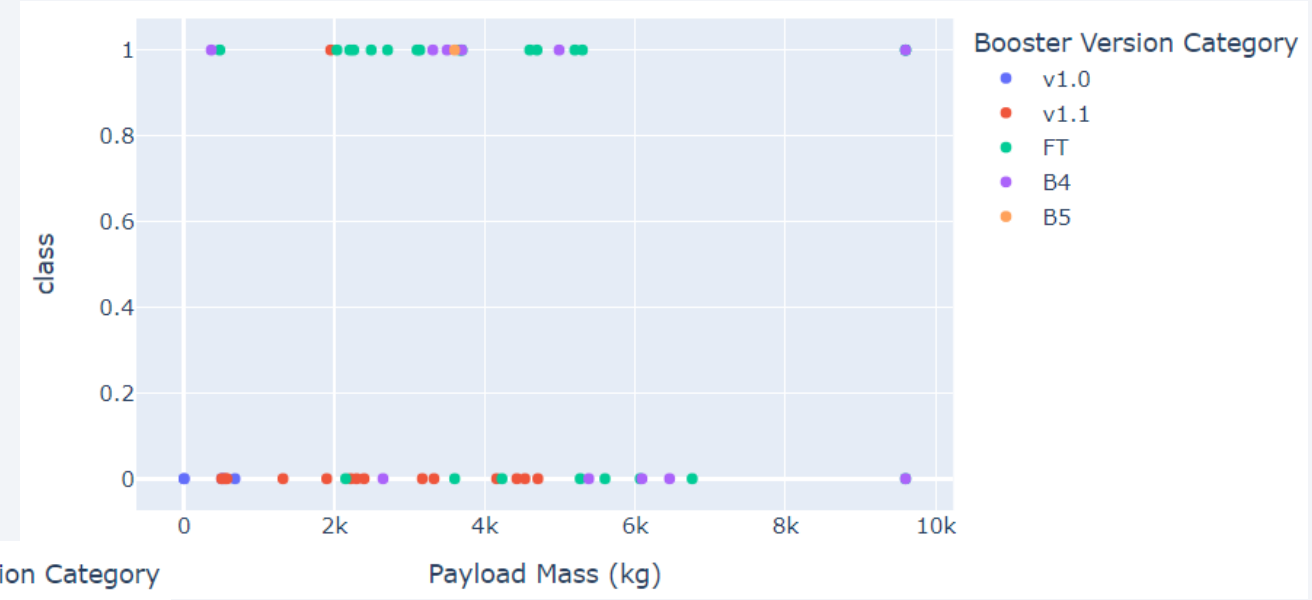
Probability of a Successful Launch From KSC LC-39A

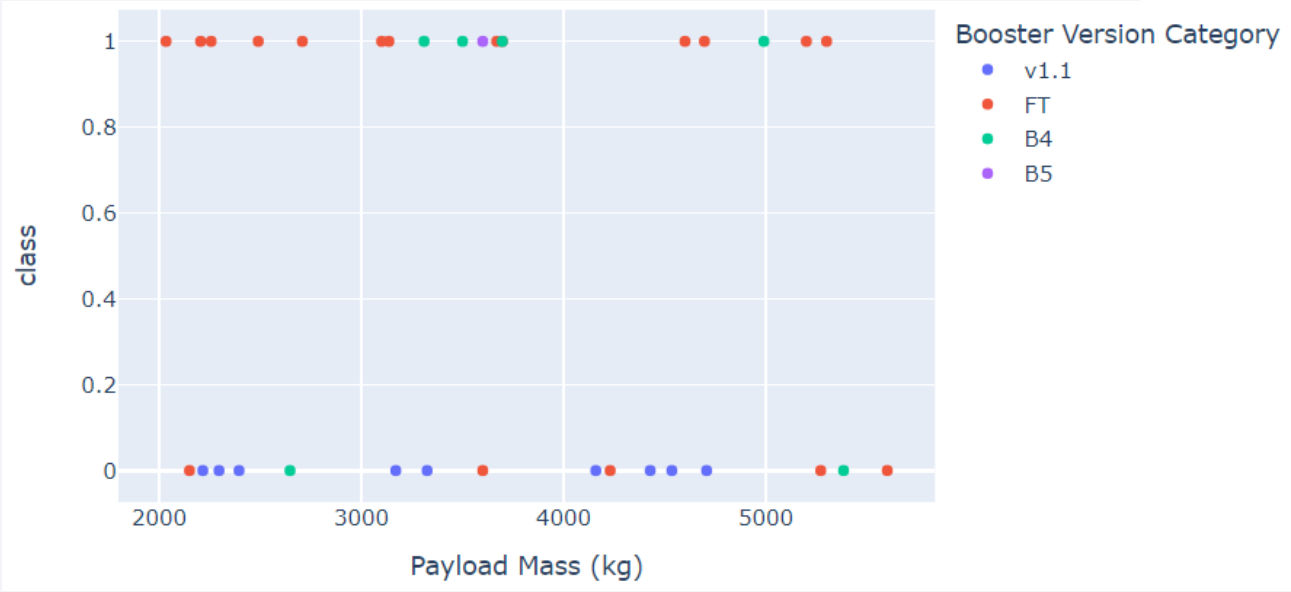
- The site with the highest number of successful launches had a successful launch percentage of 76.9%



Launch Outcome as a Function of Payload Mass

- Right: Scatter plot of success rate (1 = success, 0 = failure) across the entire range of payload masses.
- Below: Success rate for launches with payload mass between 2000 and 5500 kg.





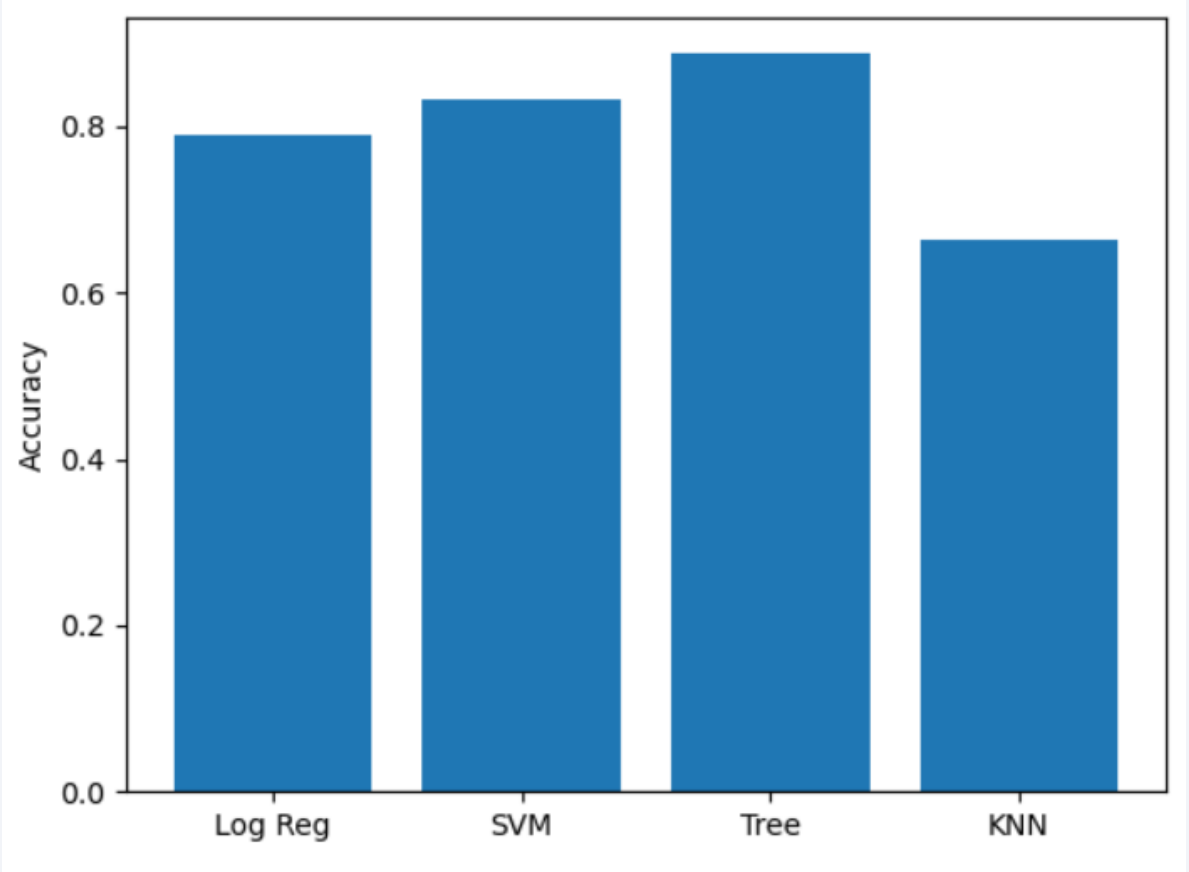
- We observe that the highest rate of successful launches is between 2000 and 5500 kg. Although there are few launches with payload masses above 6000 kg.

Section 5

Predictive Analysis (Classification)

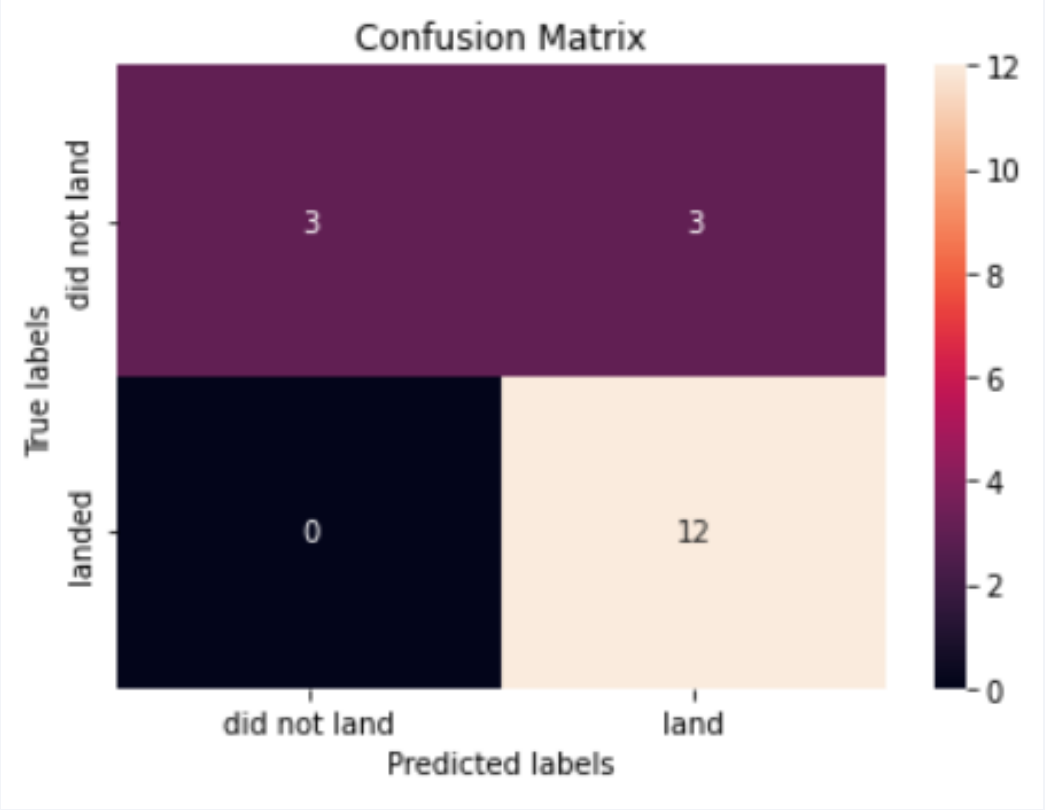
Classification Accuracy

- The best performing algorithm is Decision Tree classifier. Which correctly predicted 89% of the test set.



Confusion Matrix

- This is the confusion matrix for the Decision Tree Classifier.
- A confusion matrix displays the possible outcomes to a prediction – true positive and true negative where the model predicted correctly, and false positive and false negative where the model predicted incorrectly.
- We see this model only predicted 3 false positives, while the remaining 15 predictions were correct.



Conclusions

- The probability of a successful launch has dramatically increased between 2010 and 2020.
- Machine learning techniques can be used to achieve high accuracy predictions across multiple dimensions, that have unclear relationships with the target variable.
- Decision Tree Classifier resulted in the highest accuracy predictions within the dataset used.
- KNN had the worst accuracy.

Thank you!