IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Ann Elliott
20th June 2022

Outline

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

Executive Summary

Methodology

- Data collection through API
- Data collection by Web Scraping
- Data Wrangling
- Data analysis with SQL
- Data analysis through Data Visualisation
- Interactive Visual Analytics – Folium
- Machine Learning Prediction

Summary of all results

- Exploratory Data analysis results
- Interactive Analytics in screenshots
- Predictive Analytics – Results from Machine Learning Lab

Introduction

Project background and Context

SpaceX is an aerospace company that offers rocket launches. They found that by retrieving the boosters used during launch, and reusing them for subsequent launches they could launch at a considerably lower price than its competitors. Launching a SpaceX Falcon 9 rocket can cost as little as 62 million dollars while other providers cost upwards of 165 million dollars.

A competitor wants to create a machine learning pipeline to predict the landing outcome of this first launch stage. The goal is to predict if the first stage will land successfully. This will help to establish right price to bid against SpaceX for a rocket launch.

Possible problems:

- Identify factors that can influence the landing outcome.
- The relationships between variables and how the outcome can be affected.
- Identify the conditions needed to increase the probability of successful landing.

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Data was collected using SpaceX REST API and web scraping from Wikipedia
- Perform data wrangling:
  - One-hot encoding was applied to categorical features
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models

Data Collection

Data collection

- Data collection was done by performing a get request to the SpaceX API.

Clean data

- Next, we decoded the response content as a Json using .json() function call and turn it into a pandas dataframe using .json_normalize().
- We then cleaned the data, checked for missing values and filled in missing values where necessary.

Create Dataframe

- In addition, we performed web scraping from Wikipedia for Falcon 9 launch records with BeautifulSoup.
- The objective was to extract the launch records as an HTML table, parse the table and convert it to a Pandas dataframe for analysis.

Data Collection – SpaceX API

1. Get request for rocket launch data using API

In [6]:
```
spacex_url="https://api.spacexdata.com/v4/launches/past"
```

In [7]:
```
response = requests.get(spacex_url)
```

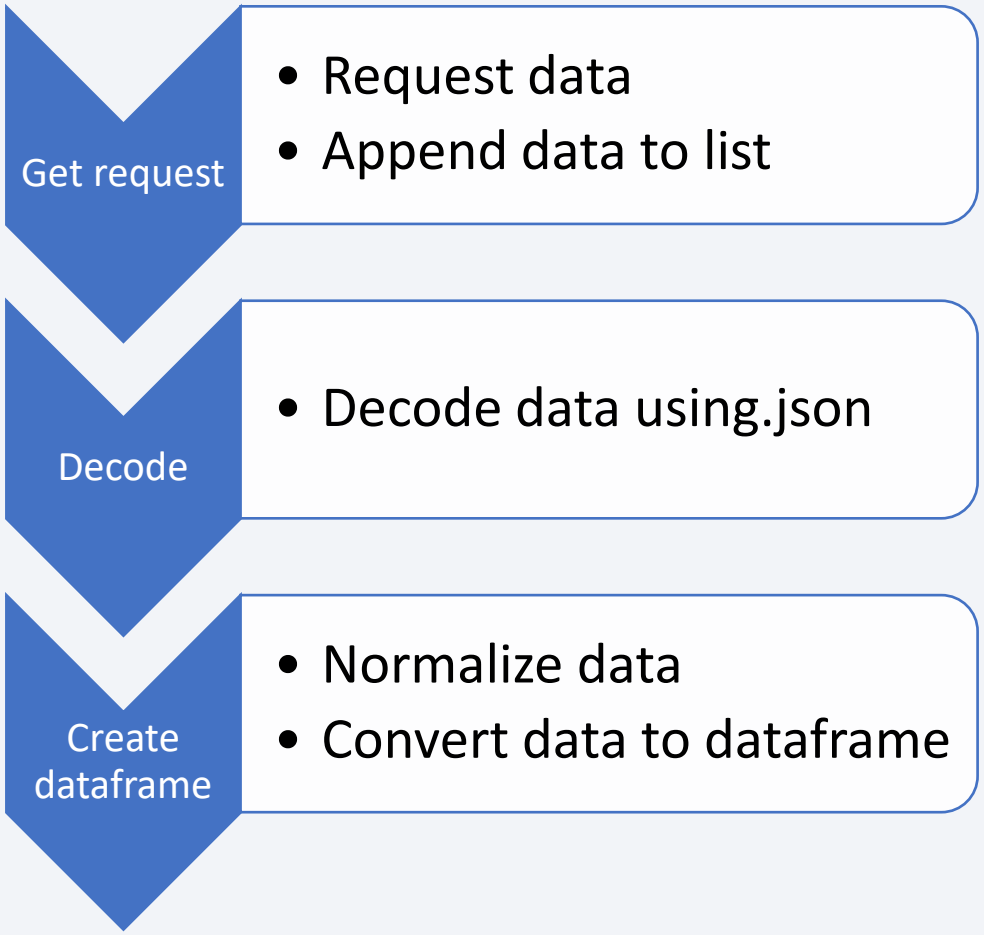
2. Use json_normalize method to convert json result to dataframe

In [12]:
```
# Use json_normalize method to convert the json result into a dataframe

# decode response content as json
static_json_df = res.json()
```

In [13]:
```
# apply json_normalize
data = pd.json_normalize(static_json_df)
```

3. We then performed data cleaning and filling in the missing values

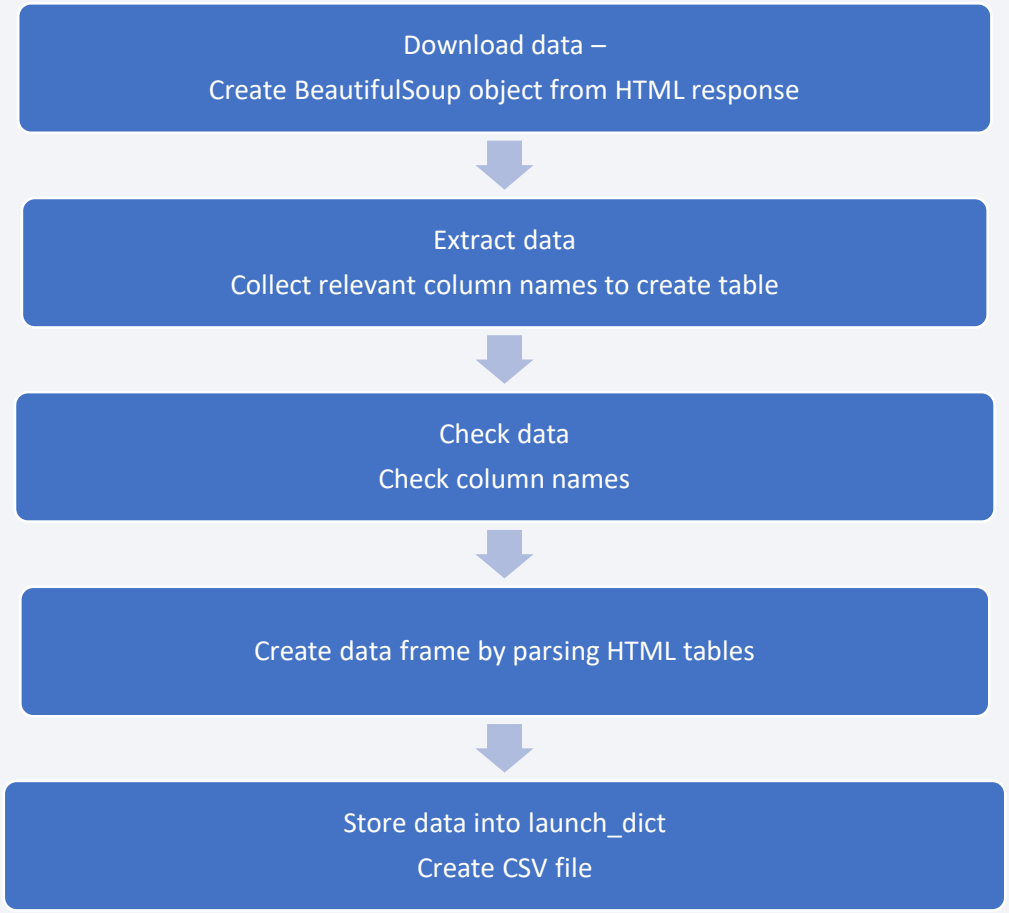
In [30]:
```
rows = data_falcon9['PayloadMass'].values.tolist()[0]

df_rows = pd.DataFrame(rows)
df_rows = df_rows.replace(np.nan, PayloadMass)

data_falcon9['PayloadMass'][0] = df_rows.values
data_falcon9
```



https://github.com/MAElliottWilms/FinalProjectSpaceY/blob/master/jupyter-labs-spacex-data-collection-api_Completed.ipynb

Data Collection – Web Scraping

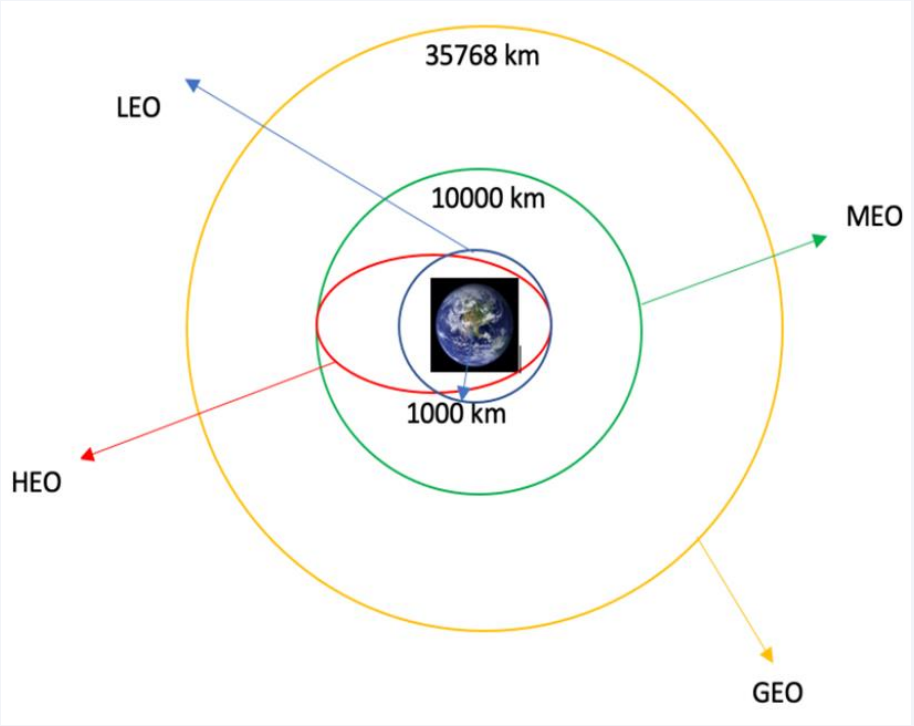
- We used web scraping to parse Falcon 9 launch records with BeautifulSoup
- We parsed the table and converted it into a pandas dataframe.



https://github.com/MAElliottWilms/FinalProjectSpaceY/blob/master/jupyter-labs-webscraping_completed.ipynb

Data Wrangling

- A dataset was loaded from CSV and checked for missing values
- Datatypes were corrected where necessary
- We used the method value_counts() to check the number of launches at each site
- By using value_count() the number and occurrence of each orbit were calculated and the number of landing_outcomes determined.
- Using the Outcome column, we created a list to show Landing_class to show whether the first stage landed successfully (1) or unsuccessfully (0).
- We used Use df[“Class”].mean() to determine success rate and exported the file to CSV

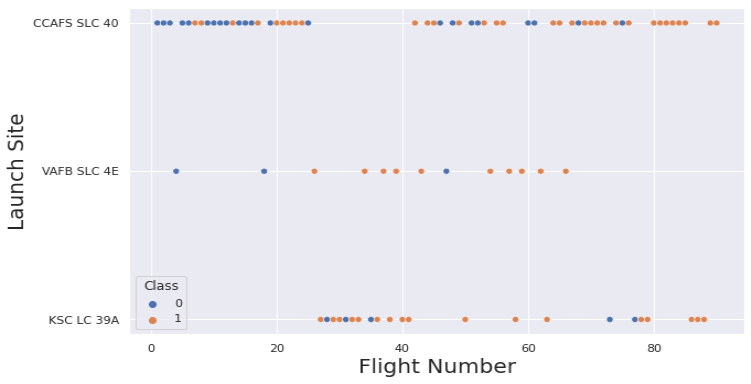


- https://github.com/MAElliottWilms/FinalProjectSpaceY/blob/master/labs-jupyter-spacex-Data%20wrangling_Complete.ipynb

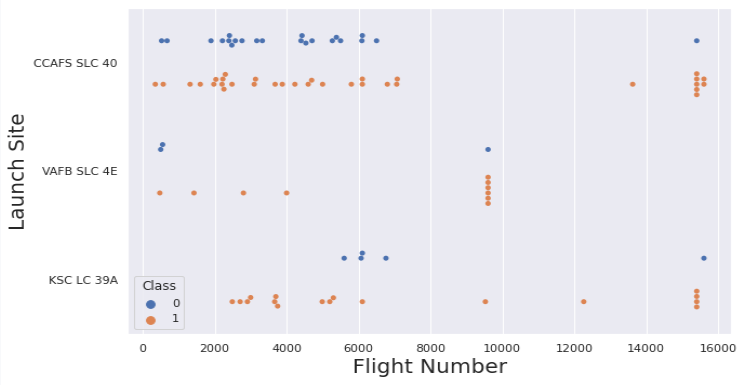
EDA with Data Visualization

We created the following scatter point charts to visualize the following relationships.

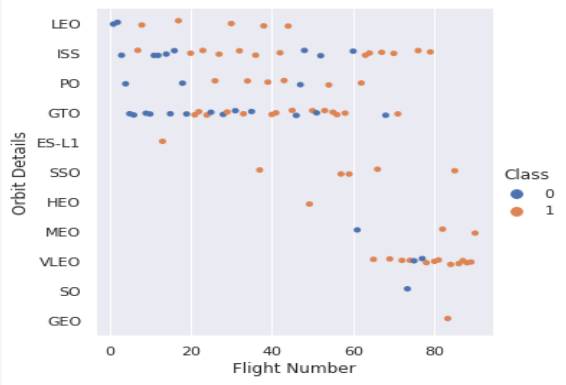
Flight number and launch site



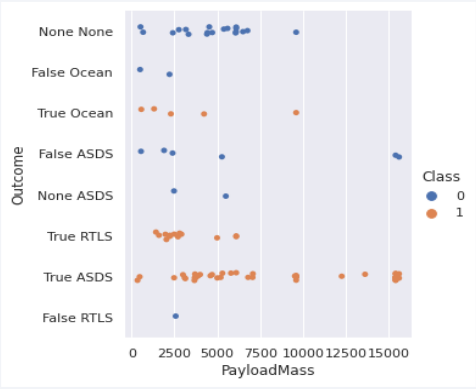
Payload and launch site



Flight number and orbit type



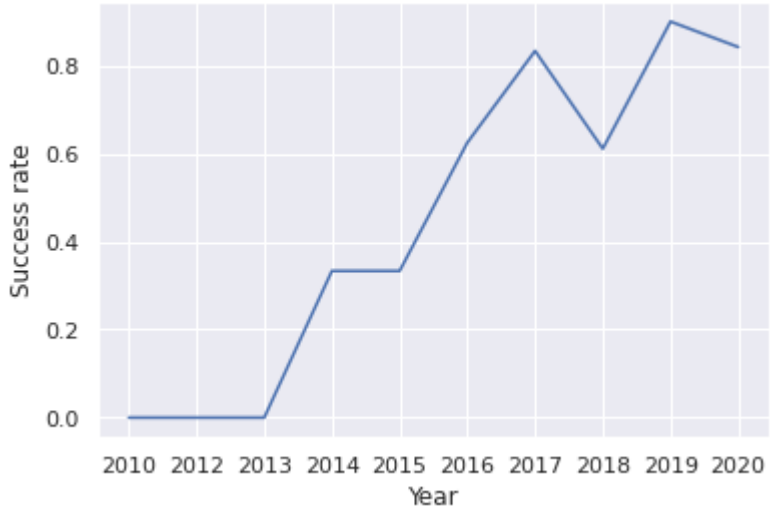
Payload and orbit type



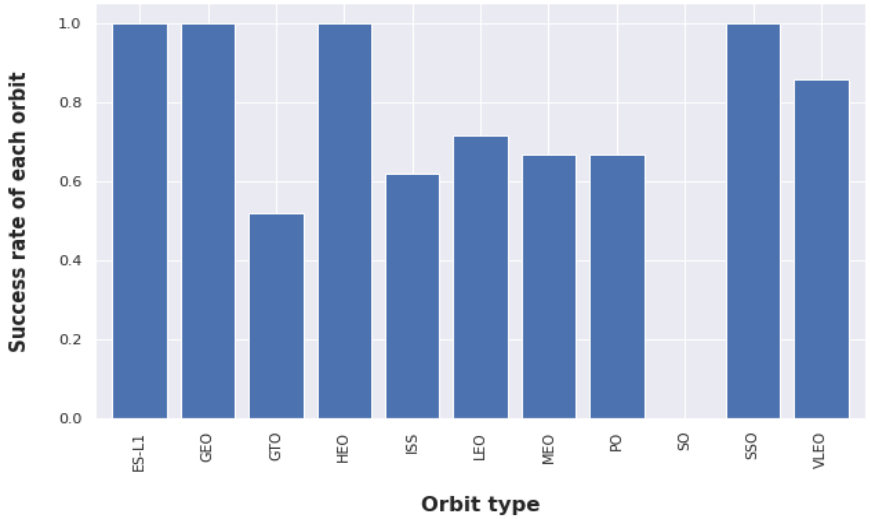
https://github.com/MAElliottWilms/FinalProjectSpaceY/blob/master/EDA%20with%20data%20visualisation2_Complete.ipynb

EDA with Data Visualization

Line chart to visualize the launch success yearly trend



Bar chart to visualize success rate of each orbit type



Feature Engineering was used in success prediction in the future module by created the dummyvariables to categorical columns.

We see that as the flight number increases, the first stage is more likely to land successfully. The payload mass is also important; it seems the higher mass of the payload, the less likely the first part will return successfully.

https://github.com/MAElliottWilms/FinalProjectSpaceY/blob/master/EDA%20with%20data%20visualisation2_Complete.ipynb

EDA with SQL

We performed the following queries in SQL:

- Display the names of the launch sites.
- Display 5 records where launch sites begin with the string ‘KSC’.
- Display the total payload mass carried by booster launched by NASA (CRS).
- Display the average payload mass carried by booster version F9 v1.1.
- List the date when the first successful landing outcome in drone ship 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.
- List the records which will display the month names, succesful landing_outcomes in ground pad ,booster versions, launch_site for the months in year 2017
- Rank the count of landing outcomes or success between the date 2010-06-04 and 2017-03-20, in descending order.

https://github.com/MAElliottWilms/FinalProjectSpaceY/blob/master/jupyter-labs-eda-sql-edx_complete.ipynb

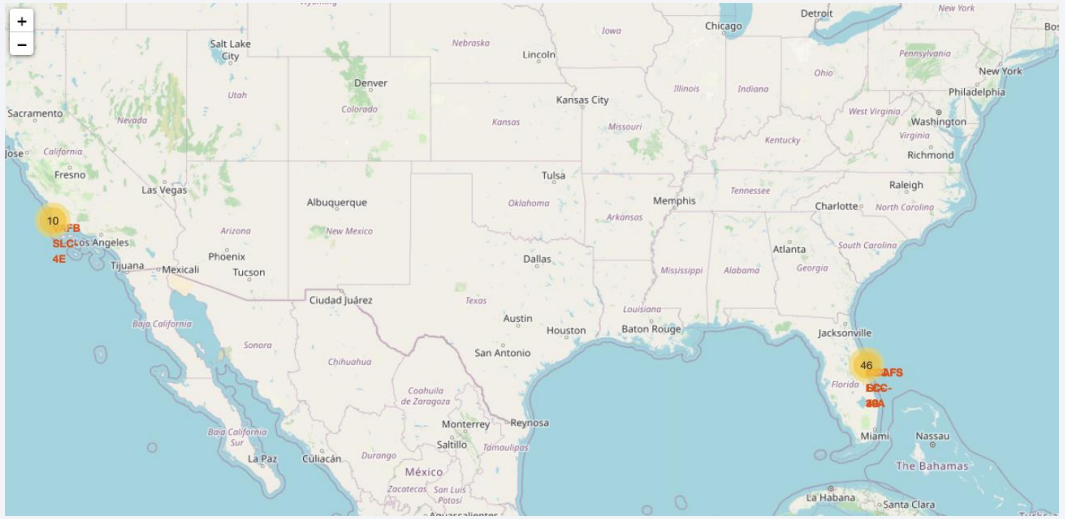
Build an Interactive Map with Folium

Launch success may depend on the location and proximities of a launch site. Finding an optimal location for building a launch site involves many factors and we could analyze the existing launch site locations.

We took the latitude and longitude coordinates of each launch site, we added a folium Map objects, and added circle markers with a label of the name of the launch site.

We marked the success/failed launches for each site on the map using dataframe launch_outcomes(failure,success) to classes 0 and 1 with Red and **Green** markers on the map in a MarkerCluster() object.

We added a MousePosition to get coordinates based on the mouse position. We then calculated the distance of the launch sites from various landmarks such as railways, highways, and coastlines and drew PolyLines to show the distance.



https://github.com/MAElliottWilms/FinalProjectSpaceY/blob/master/Interactive%20Visual%20Analytics%20with%20Folium_Completed.ipynb

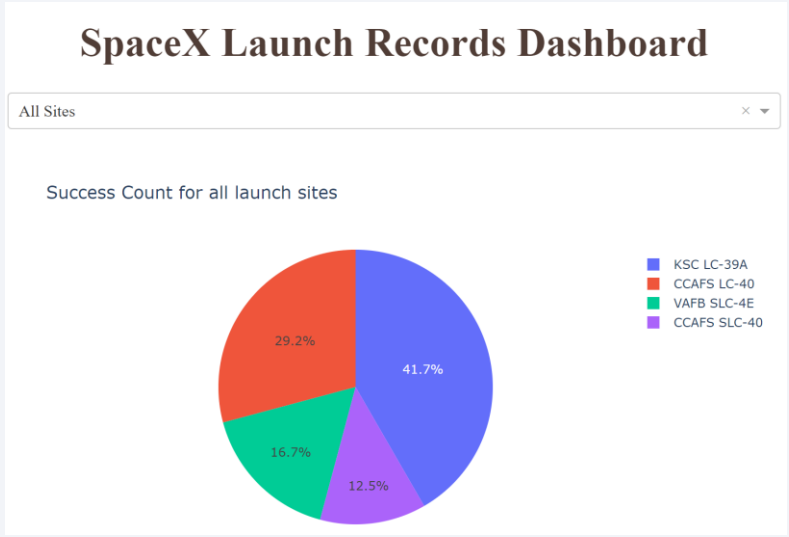
Build a Dashboard with Plotly Dash

We built an interactive dashboard with the following:

Pie chart – Success Count

To visualize the success count for all launch sites

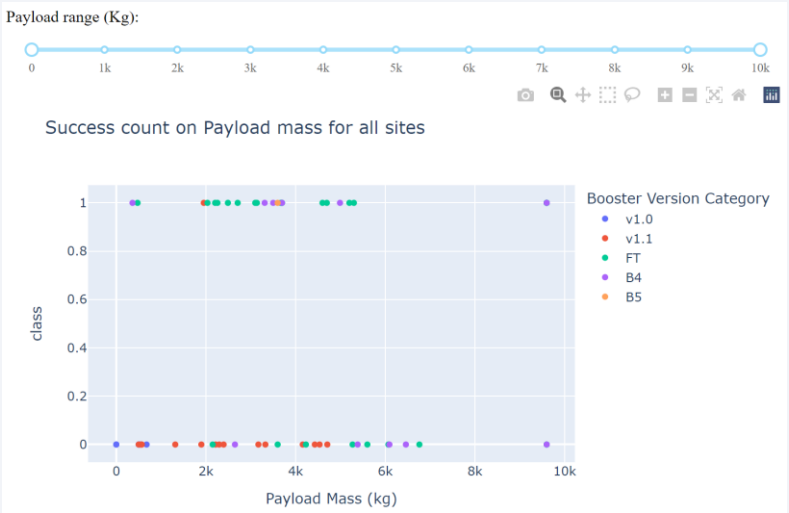
Includes drop down menu to select just one or all launch sites



Scatter graph with a slide scale

This shows the success rate depending on the payload range selected on the sliding scale.

It also shows which booster was used.



https://github.com/MAElliottWilms/FinalProjectSpaceY/blob/master/spacex_dash_app.py

Predictive Analysis (Classification)

Build model

- Load the dataset into NumPy and Pandas
- Transform the data
- Split into training and testing sets

Evaluate model

- Create a logistic regression object then create a GridSearchCV object
- Fit the object to find the best parameters

Improve model

- Compare accuracy by checking the data
- Get tuned hyperparameters for each type of algorithm: Logistic regression object
- Plot the Confusion matrix.

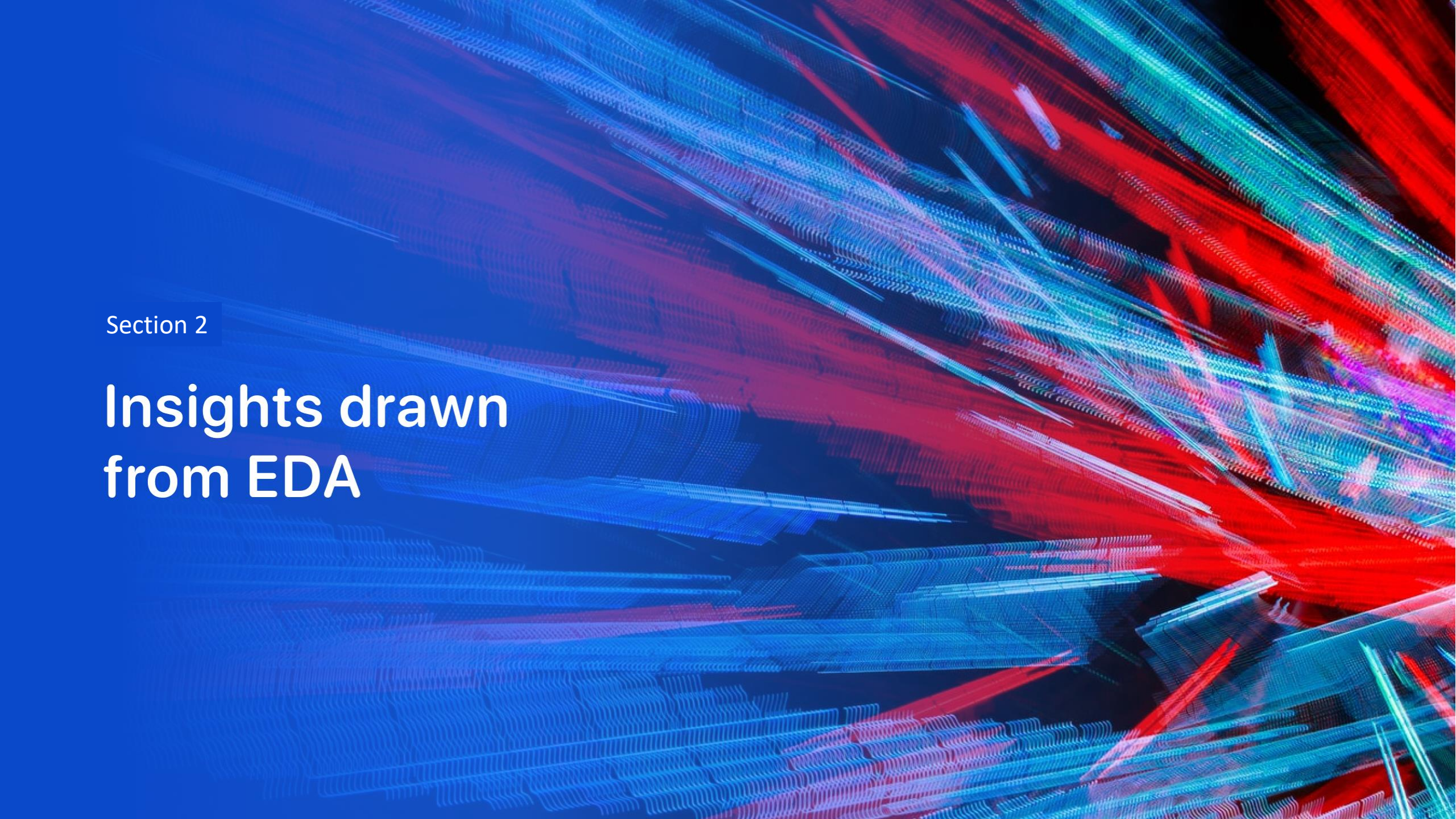
Find best model

- We used accuracy as the metric for our model
- Use Feature engineering and Algorithm tuning
- The model with the best accuracy score is the best performing model

https://github.com/MAElliottWilms/FinalProjectSpaceY/blob/master/Machine%20Learning_complete.ipynb

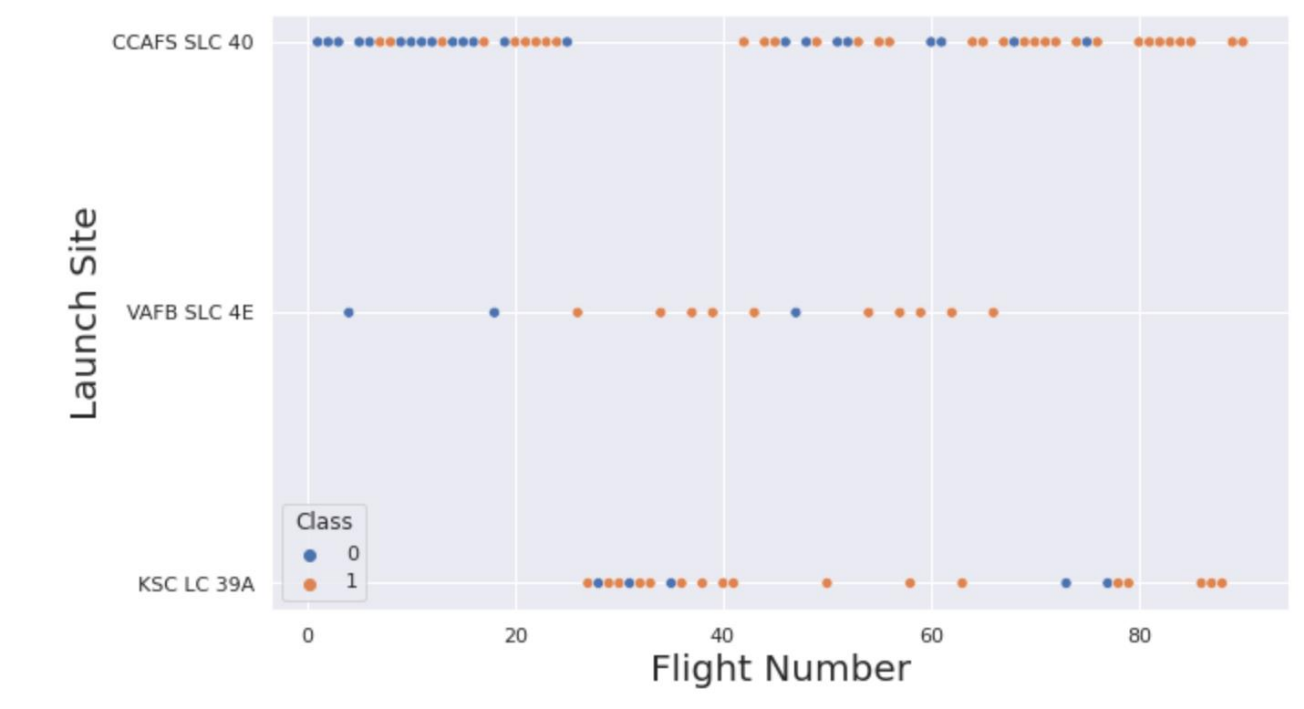
Results

- 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 of Flight Number vs. Launch Site

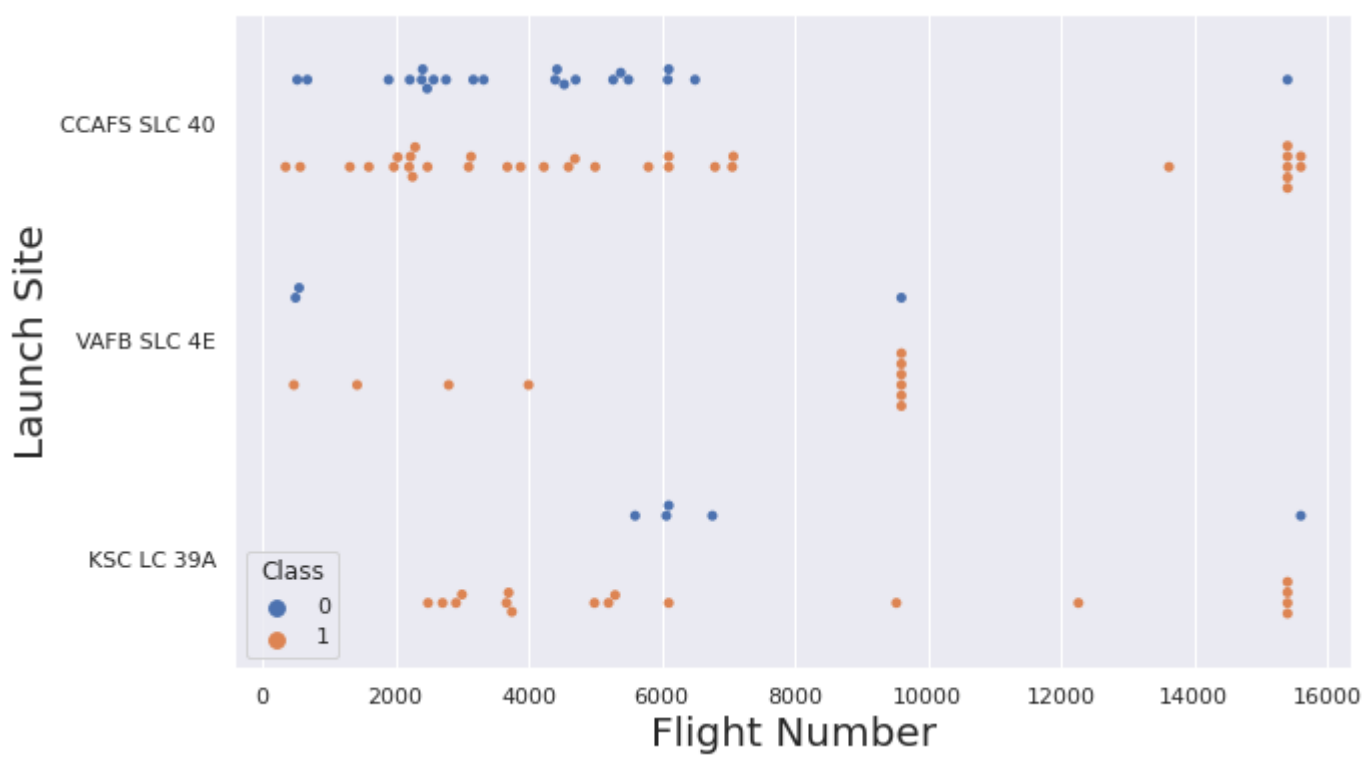


This scatter plot shows that the success rate increases the more flights are launched per launch site.

Site CCAFS SLC 40 shows the least pattern in this.

Payload vs. Launch Site

Scatter plot of Payload vs. Launch Site



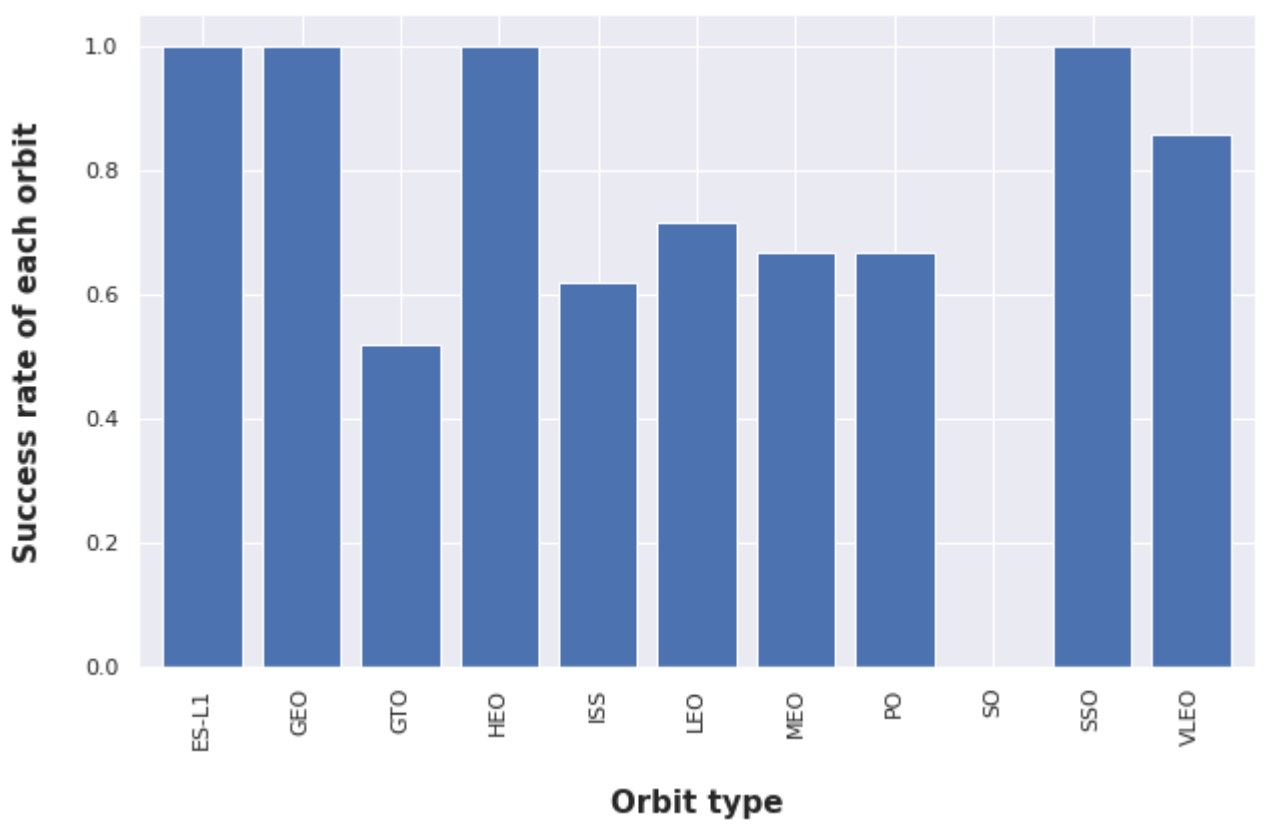
This scatter plot shows that when the payload mass around 3000 kg is the probability of success will be greatly increased. It also shows high success with payloads above 15000 kg

For the VAFB-SLC launch site there are no rockets launched for heavy payload mass (greater than 10000 kg).

The majority of payloads with lower mass have been launched from CCAFH SLC 40.

Success Rate vs. Orbit Type

Bar chart with Success rate of each orbit type



This figure depicts the likelihood of the orbits to influence the landing outcomes.

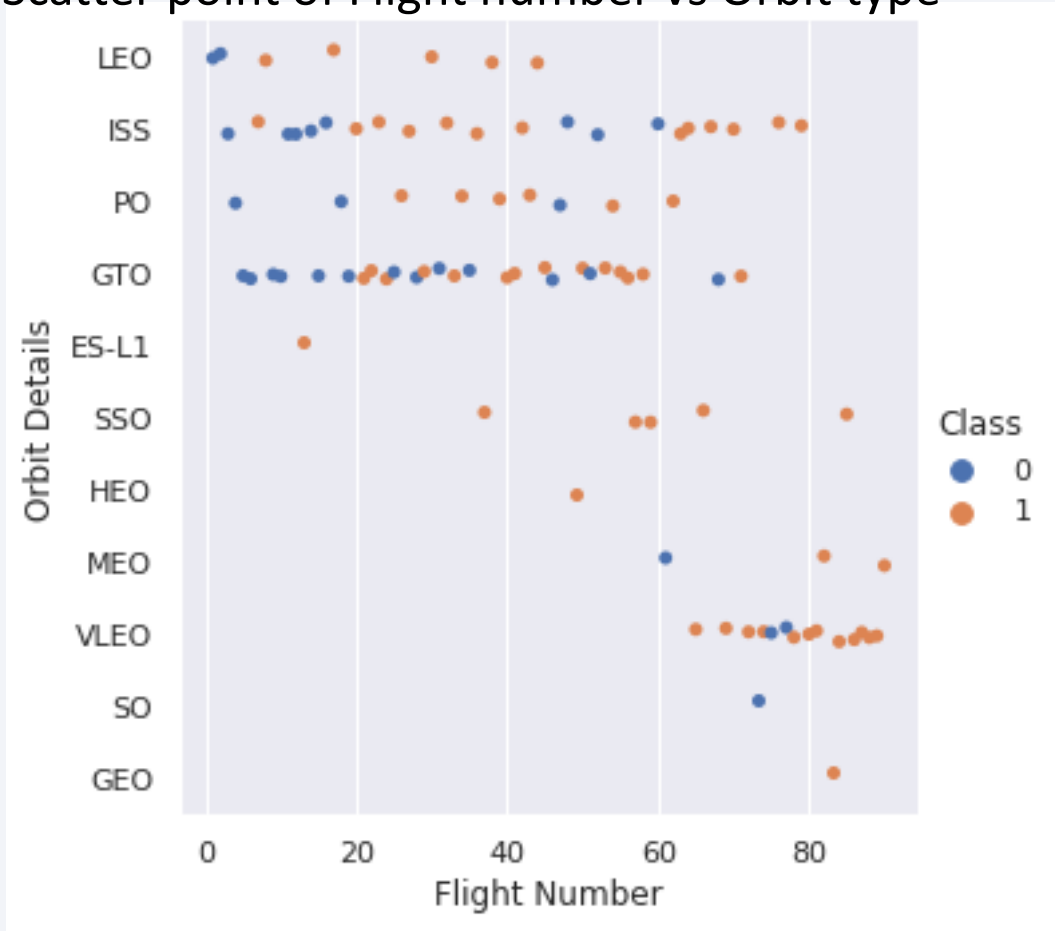
Some orbits have 100% success rate such as SSO, HEO, GEO AND ES-L1.

SO orbit produced a 0% rate of success.

However, GEO, SO, HEO and ES-L1 have only one result which means we need more data before we can draw any conclusions.

Flight Number vs. Orbit Type

Scatter point of Flight number vs Orbit type



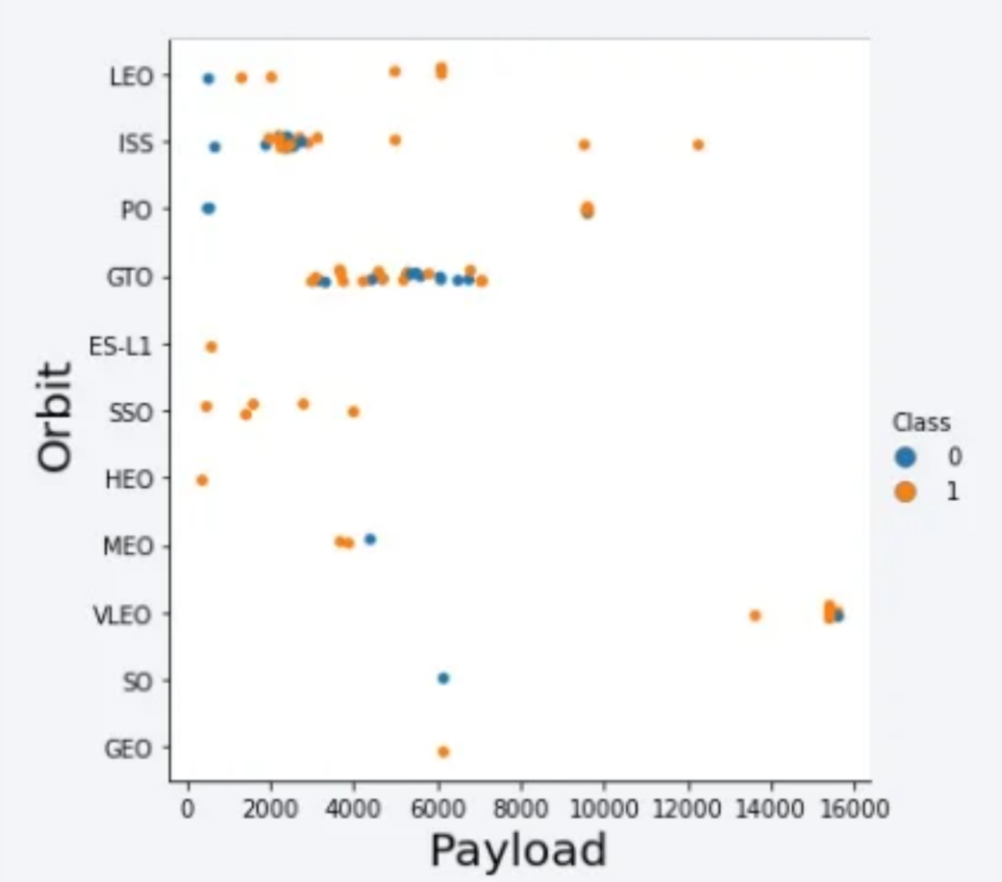
This scatter plot shows that generally, the larger number of flights launched from each orbit, the greater the success rate (especially the VLEO orbit).

GTO orbit shows no relationship between the both attributes.

Orbits that only have 1 occurrence should also be excluded from the above statement as more data is needed.

Payload vs. Orbit Type

Scatter point of Payload vs. Orbit type



ISS - A payload of 2000 led to the most success for this orbit. Payloads over this amount were also successful.

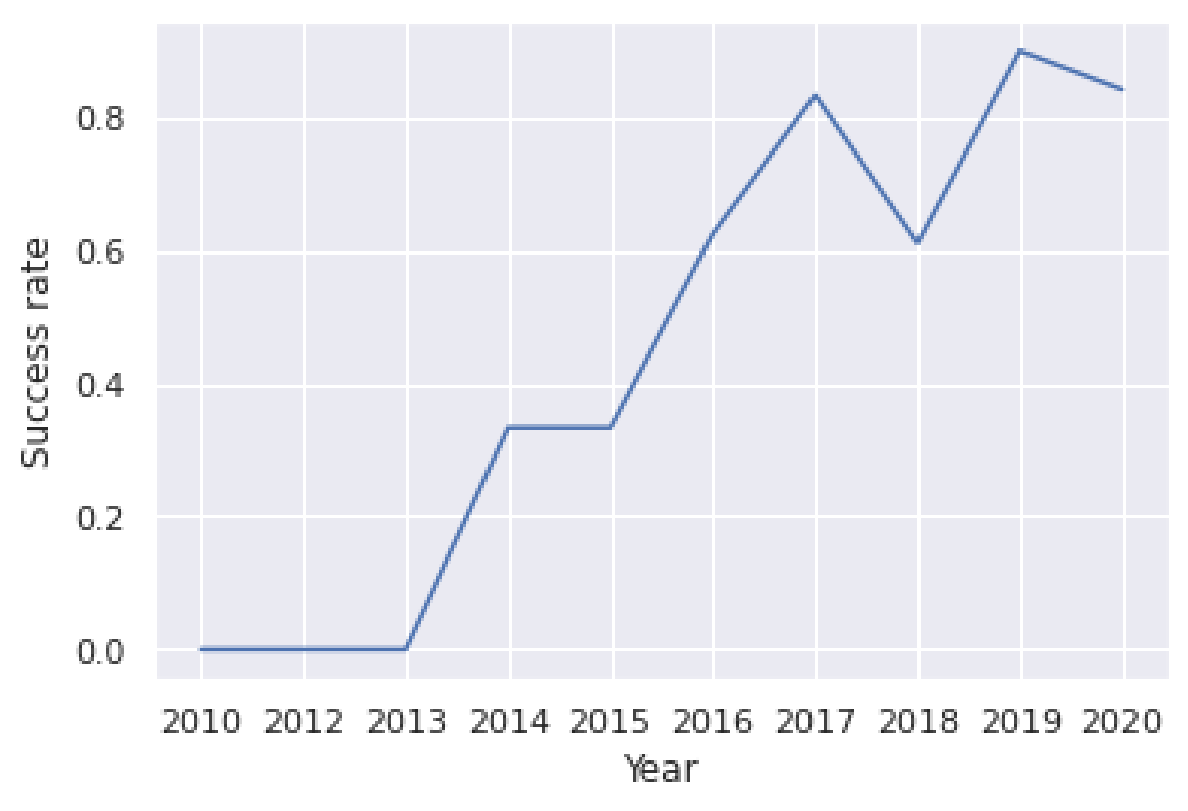
GTO - A payload of 3-4000 gave the most success, but payloads over 5000 are largely unsuccessful.

VLEO – Payloads of over 13000 are generally successful but there is no data for lower mass.

SSO – has a consistent rate of success with payloads up to 5000.

Launch Success - Yearly Trend

Line Chart of Yearly average success rate



This graph displays a growing success rate from the year 2013 until 2020.

All Launch Site Names

Names of unique launch sites

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

We used the key word DISTINCT to show only unique launch sites from the SpaceX data

%sql SELECT DISTINT LAUNCH_SITE as “Launc_Sites” FROM SPACEXTBL;

Launch Site Names Begin with 'KSC'

5 records where launch sites' names start with `KSC` were selected from the dataset.

```
%sql SELECT LAUNCH_SITE FROM SPACEXTBL WHERE LAUNCH_SITE LIKE 'KSC%' LIMIT 5;
```

```
 * ibm_db_sa://dds32186:***@824dfd4d-99de-440d-9991-629c01b3832d.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:30119/bludb
Done.
```

launch_site
KSC LC-39A
KSC LC-39A
KSC LC-39A
KSC LC-39A
KSC LC-39A

Total Payload Mass

The total payload carried by boosters from NASA was **45596 kg** using the query below:

```
%sql SELECT SUM (PAYLOAD_MASS__kg_) FROM SPACEXTBL WHERE CUSTOMER = 'NASA(CRS)' ;
```

```
 * ibm_db_sa://dds32186:***@824dfd4d-99de-440d-9991-629c01b3832d.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:30119/bludb
Done.
```

Average Payload Mass by F9 v1.1

The average payload mass carried by booster version F9 v1.1 is 2928 kg

```
%sql SELECT AVG (PAYLOAD_MASS__KG_) FROM SPACEXTBL WHERE BOOSTER_VERSION = 'F9 v1.1';
```

```
 * ibm_db_sa://dds32186:***@824dfd4d-99de-440d-9991-629c01b3832d.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:30119/bludb
Done.
```

1
2928

First Successful Ground Landing Date

The min() function was used to find the result of the first landing.

The date of the first successful landing outcome on a ground pad was on:
22nd December 2015

```
%sql SELECT MIN (DATE) AS FirstSuccessfullanding FROM SPACEXTBL WHERE Landing_Outcome LIKE 'Success (ground pad)'
```

Successful Drone Ship Landings - Payload between 4000 and 6000

The boosters which successfully landed on a drone ship and had a payload mass greater than 4000 but less than 6000 are listed here.

The WHERE clause was used to filter for boosters which have successfully landed on the drone ship, and the AND condition applied to determine successful landing with payload mass greater than 4000 but less than 6000.

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

```
%sql select BOOSTER_VERSION from SPACEXTBL where LANDING__OUTCOME='Success (drone ship)'  AND PAYLOAD_MASS__KG_ BETWEEN 4000 AND 6000
```

Total Number of Successful and Failure Mission Outcomes

Number of Successful Missions: 100

Number of Failed Missions: 1

‘%’ used to include all types of success or failure,

Successful Mission	Failed Mission
100	1

```
%sql SELECT COUNT MISSION_OUTCOME AS "successful mission" FROM SPACEXTBL WHERE MISSION_OUTCOME LIKE 'Success%'
%sql SELECT COUNT MISSION_OUTCOME AS "failed mission" FROM SPACEXTBL WHERE MISSION_OUTCOME LIKE 'Fail%'
%sql SELECT sum(case when MISSION_OUTCOME LIKE '%Success%' then 1 else 0 end) AS "Successful Mission", \
    sum(case when MISSION_OUTCOME LIKE '%Fail%' then 1 else 0 end) AS "Failed Mission" \
FROM SPACEXTBL;
```

Boosters Carried Maximum Payload

The boosters carrying the maximum payload were determined using a subquery in the WHERE clause and the MAX() function.

boosterversion
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

2017 Launch Records

This table displays succesful mission_outcomes, booster versions, and launch_sites for the months in year 2017

Note: Landing outcomes did not work for me so I used Mission outcome instead.

Month	Mission_Outcome	Booster_Version	Launch_Site
01	Success	F9 FT B1029.1	VAFB SLC-4E
02	Success	F9 FT B1031.1	KSC LC-39A
03	Success	F9 FT B1030	KSC LC-39A
03	Success	F9 FT B1021.2	KSC LC-39A
05	Success	F9 FT B1032.1	KSC LC-39A
05	Success	F9 FT B1034	KSC LC-39A
06	Success	F9 FT B1035.1	KSC LC-39A
06	Success	F9 FT B1029.2	KSC LC-39A
06	Success	F9 FT B1036.1	VAFB SLC-4E
07	Success	F9 FT B1037	KSC LC-39A
08	Success	F9 B4 B1039.1	KSC LC-39A
08	Success	F9 FT B1038.1	VAFB SLC-4E
09	Success	F9 B4 B1040.1	KSC LC-39A
10	Success	F9 B4 B1041.1	VAFB SLC-4E
10	Success	F9 FT B1031.2	KSC LC-39A
10	Success	F9 B4 B1042.1	KSC LC-39A
12	Success	F9 FT B1035.2	CCAFS SLC-40
12	Success	F9 FT B1036.2	VAFB SLC-4E

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

I selected landing outcome based on “success” between the dates provided, these were grouped by type and the count presented in descending order.

```
%sql SELECT LANDINGOUTCOME, COUNT(*) AS COUNT_LAUNCHES FROM SPACEXTBL

WHERE DATE BETWEEN '2010-06-04' AND '2017-03-20'

AND LANDINGOUTCOME LIKE 'Success%'

GROUP BY LANDING_OUTCOME ORDER BY COUNT_LAUNCHES DESC;
```

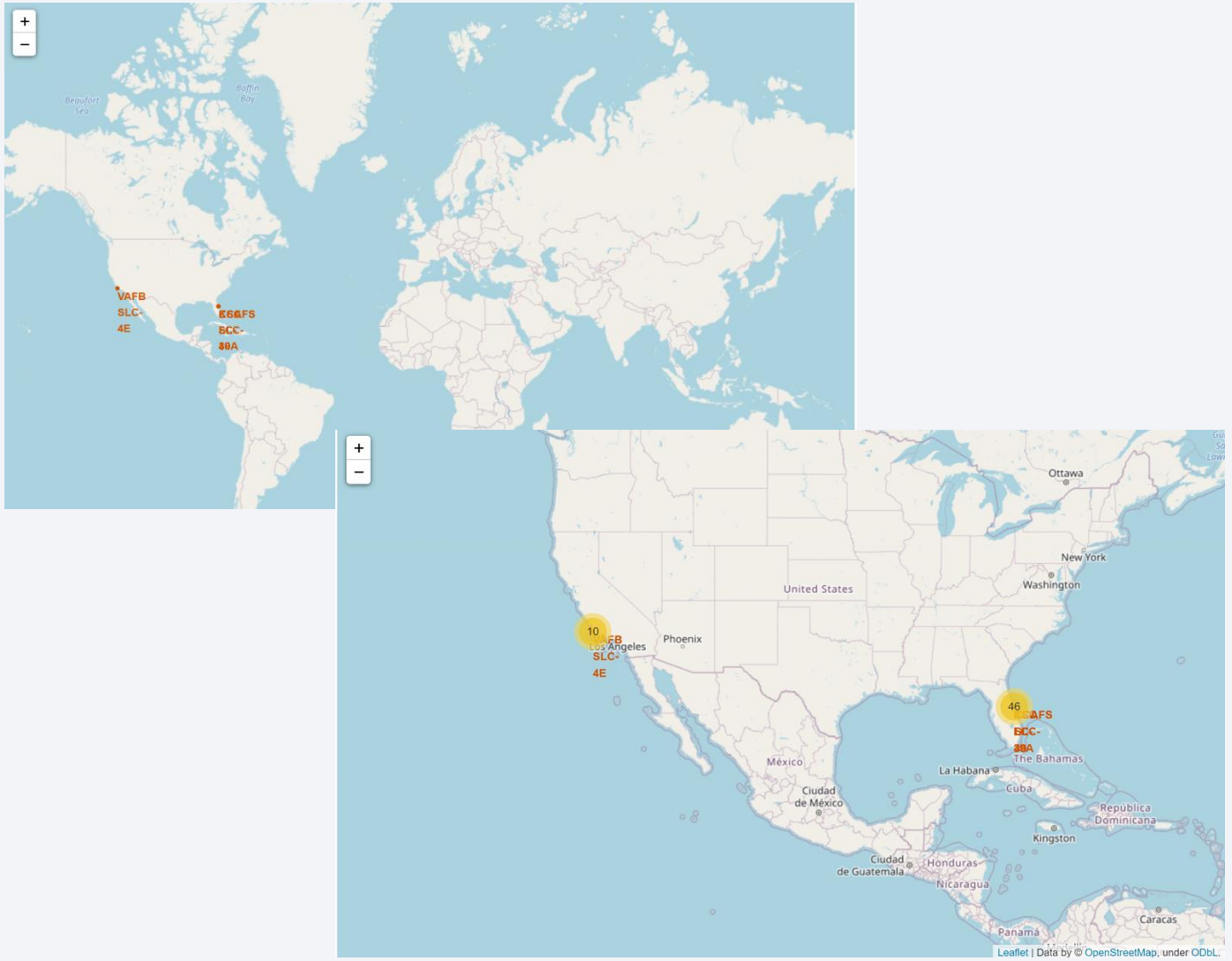

Section 3
Launch Sites
Proximities Analysis

Global map with SpaceX launch locations

We can see that all the SpaceX launch sites are in the USA.

All launch sites are close to the Equator.

All launch sites are close to coastlines.

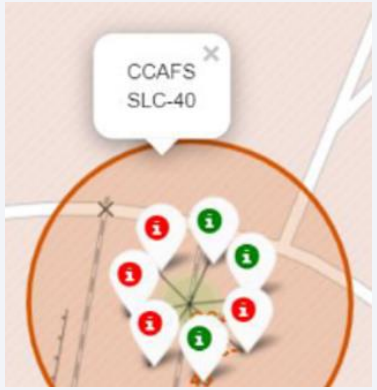


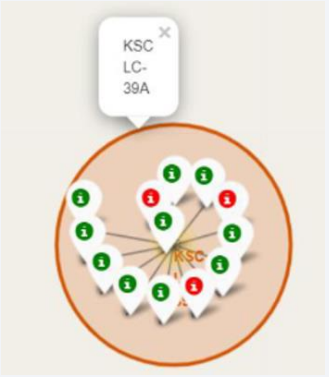
Map showing launch outcomes

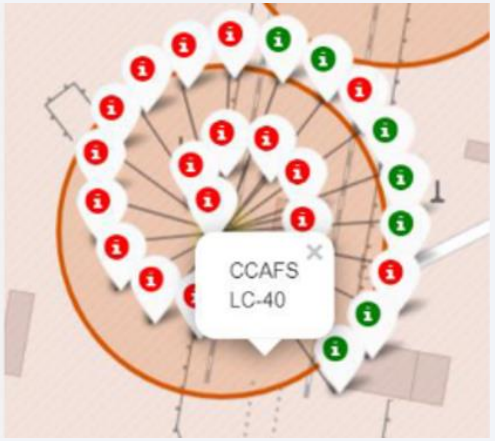
This map shows launch outcomes at different launch sites.

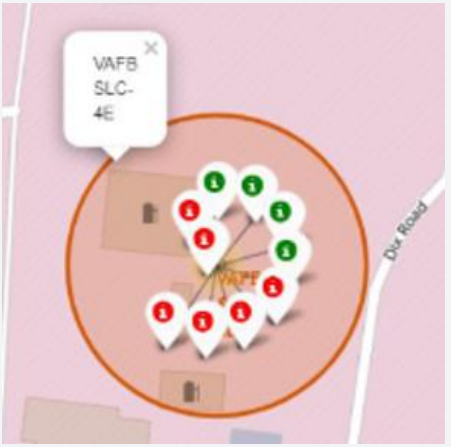
Successful launches are marked in green and unsuccessful in red.

In these examples we can see that KSC LC 39A has the highest rate of successful outcomes.







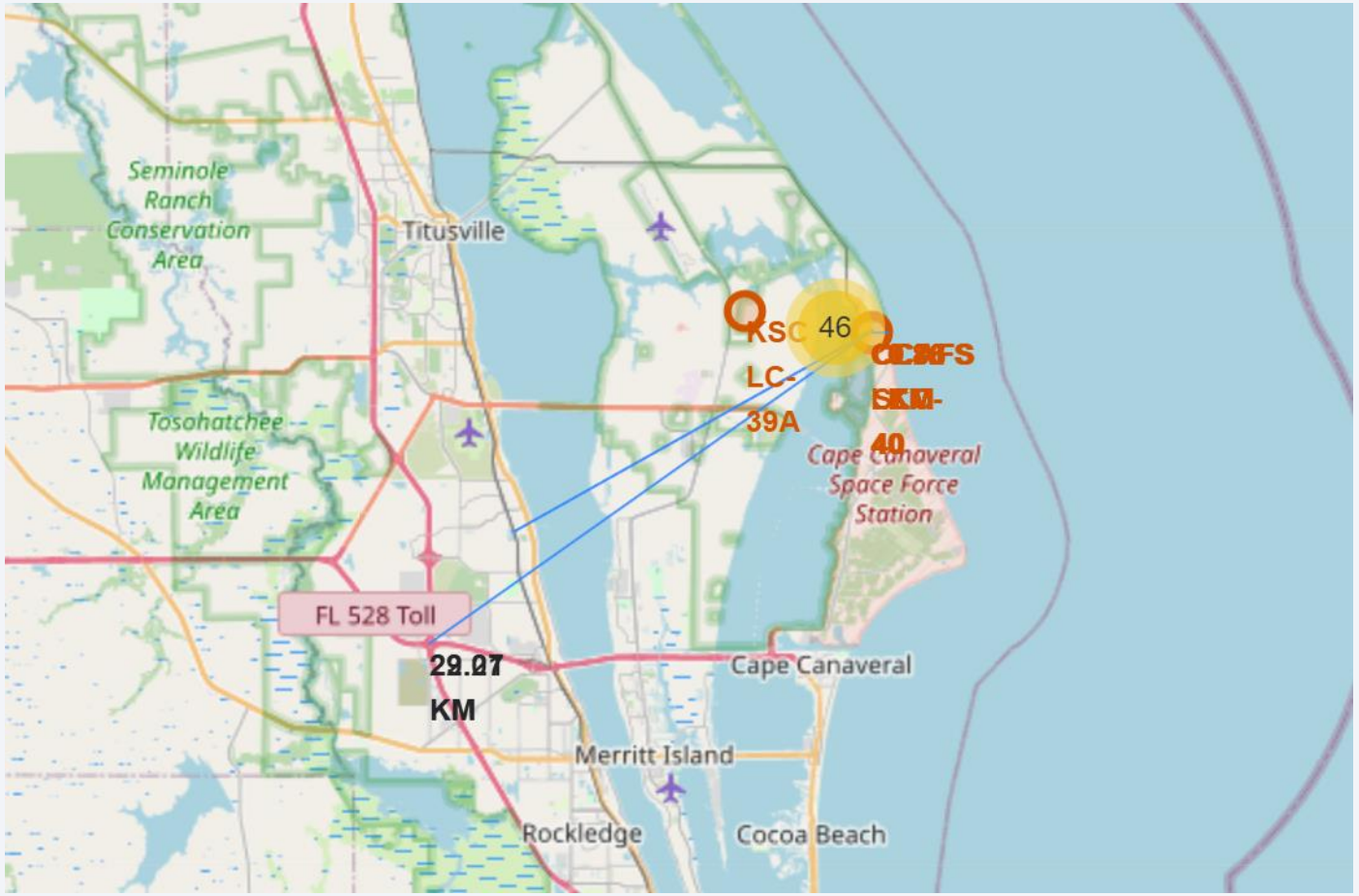


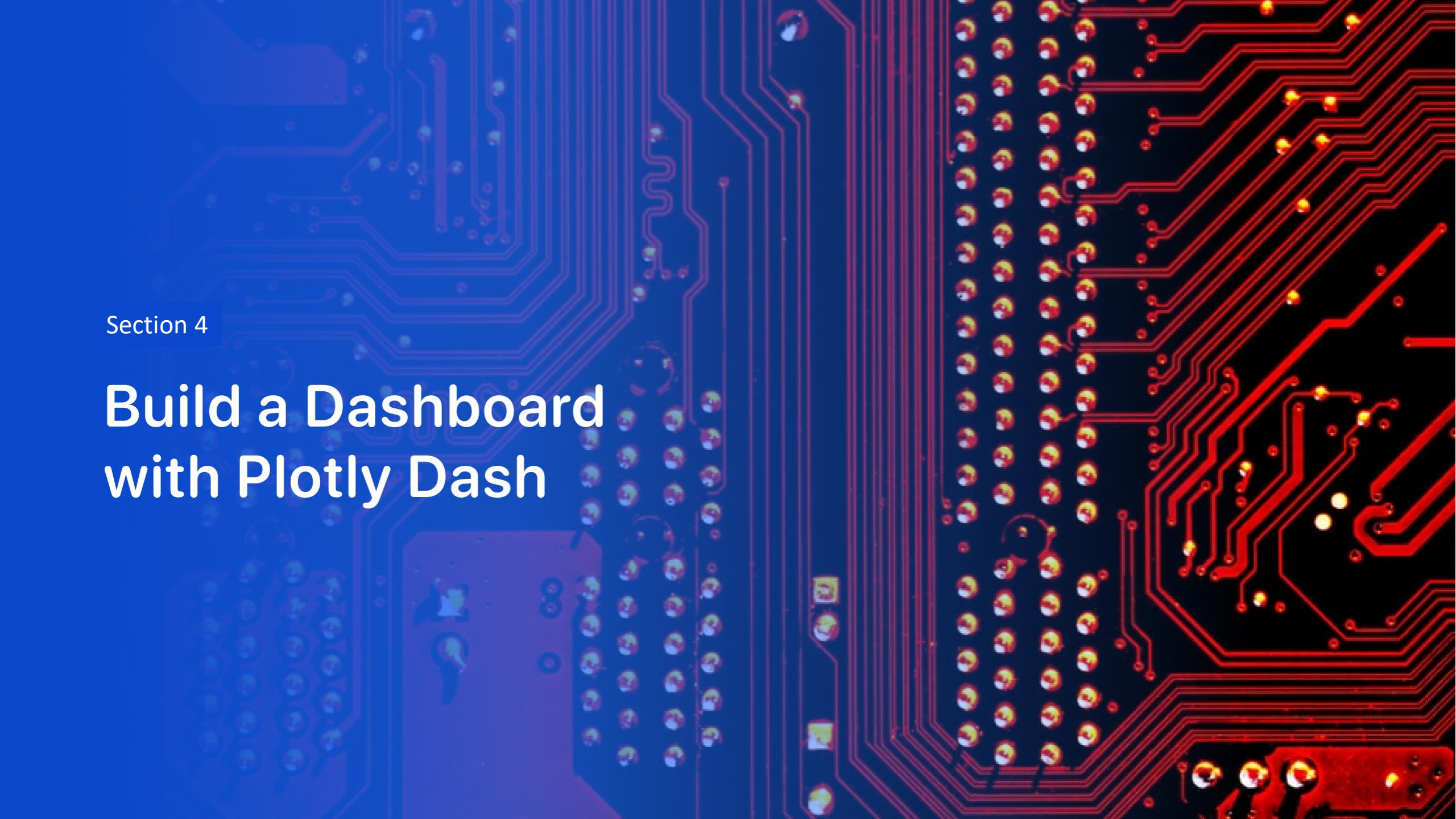
Map showing proximity to landmarks

This map shows the distance from the launchsite to the nearest highway and nearest railroad.

Launch sites are generally close to a coastline and are a certain distance away from cities.

Launch sites are not always close to railways or highways.

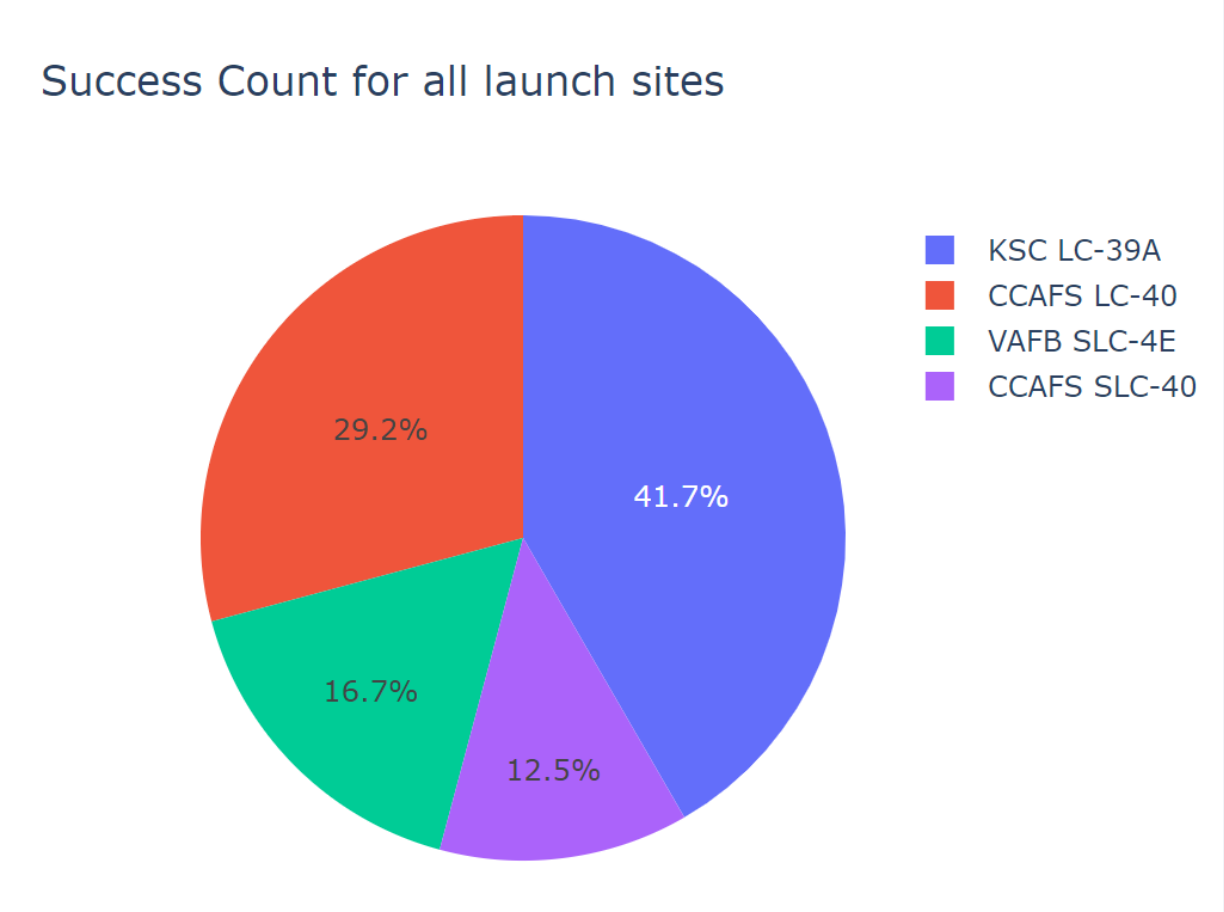



Section 4
Build a Dashboard with Plotly Dash

Success rate (%) per launch site

We can see that launch site KSC LC-39A has the greatest success rate percentage.

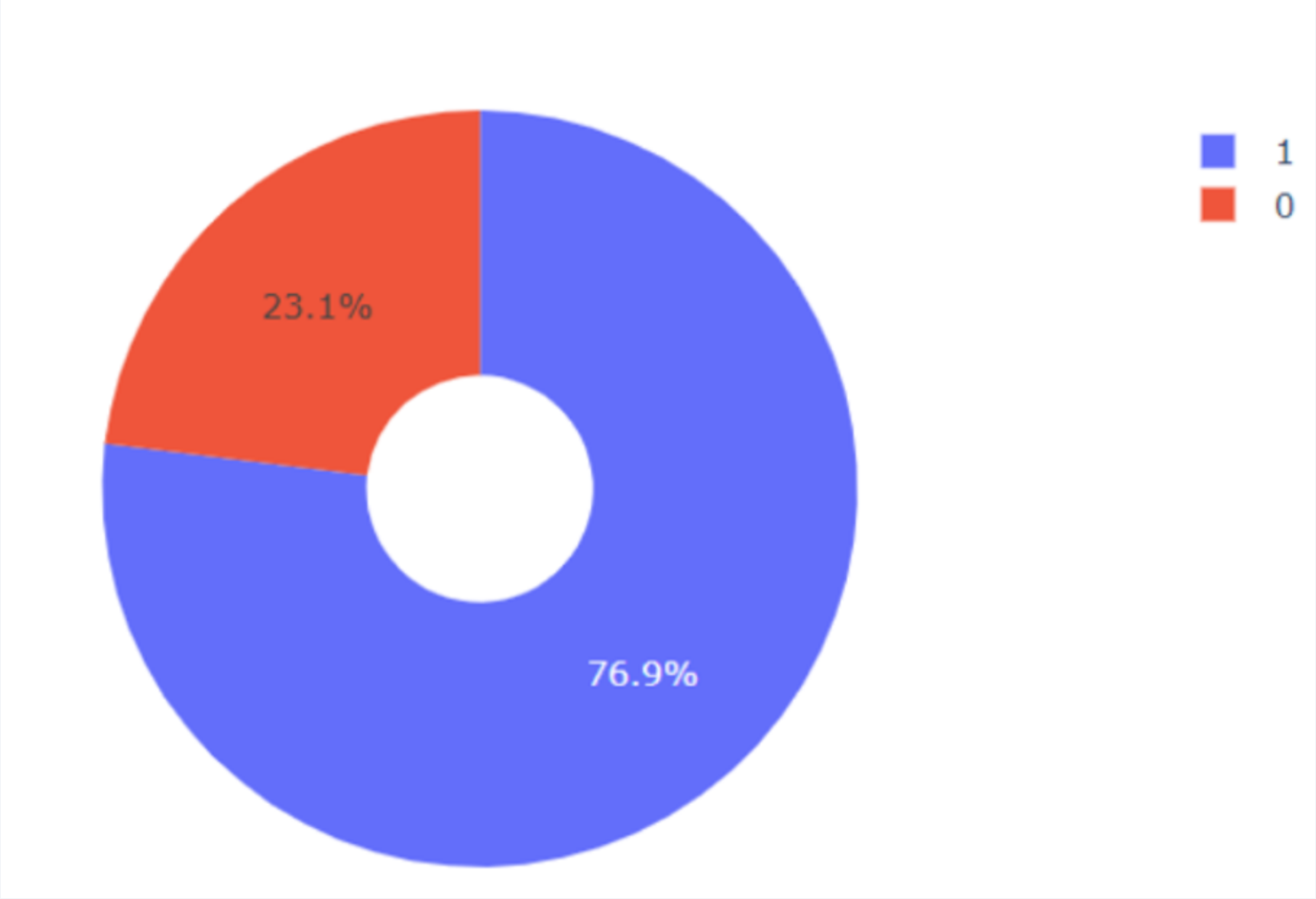
CCAFS SLC-40 has the lowest success rate percentage.



Launch site with highest launch success rate

KSC LC-39A has the greatest launch success rate percentage of 76.9%.

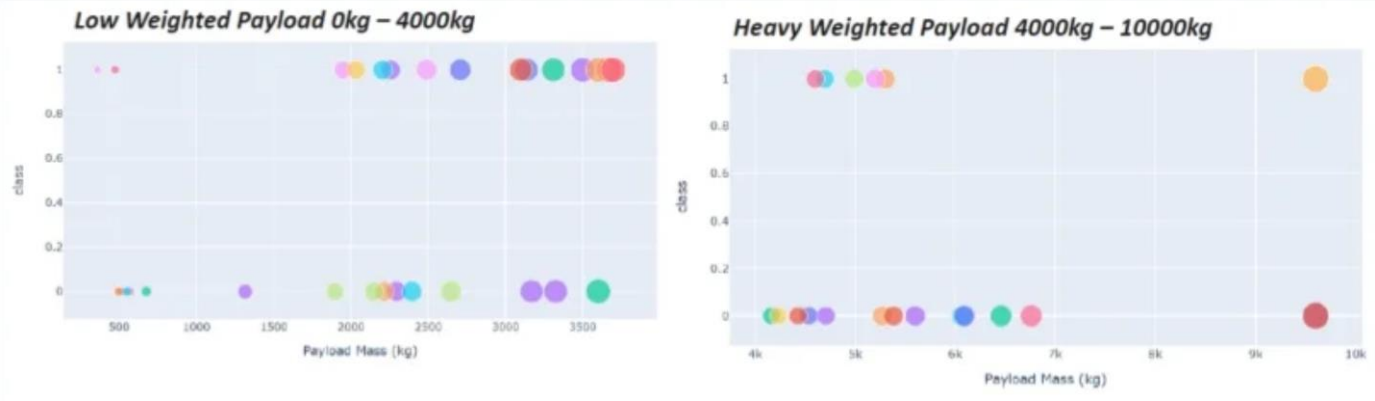
The failure rate is 23.1%.

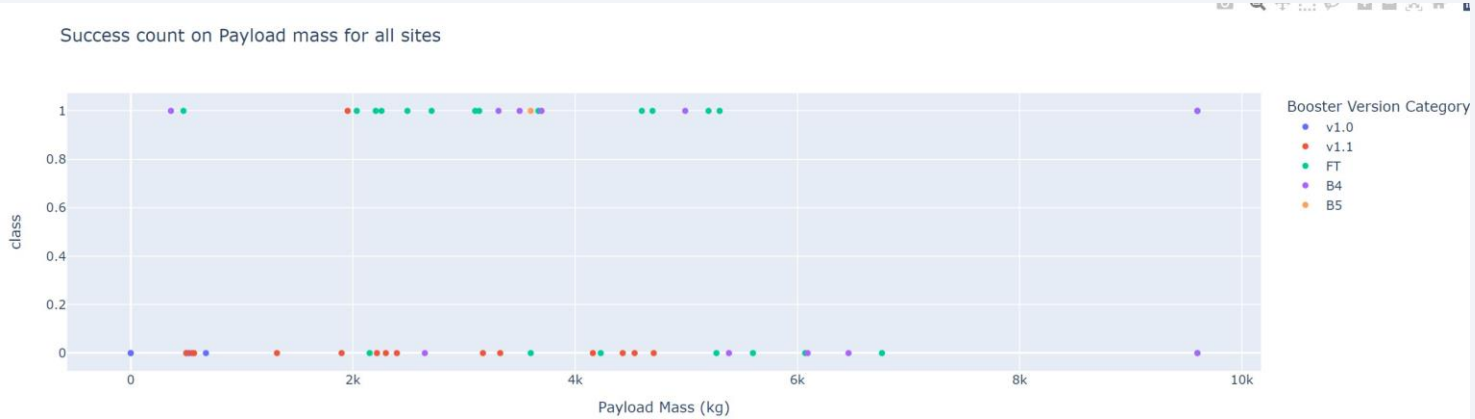


Payload related to Launch outcome

The success rate for low payloads is higher than the heavier payloads and the most successful payload range is around 3000 kg.

The B4 booster is successful with all payloads.



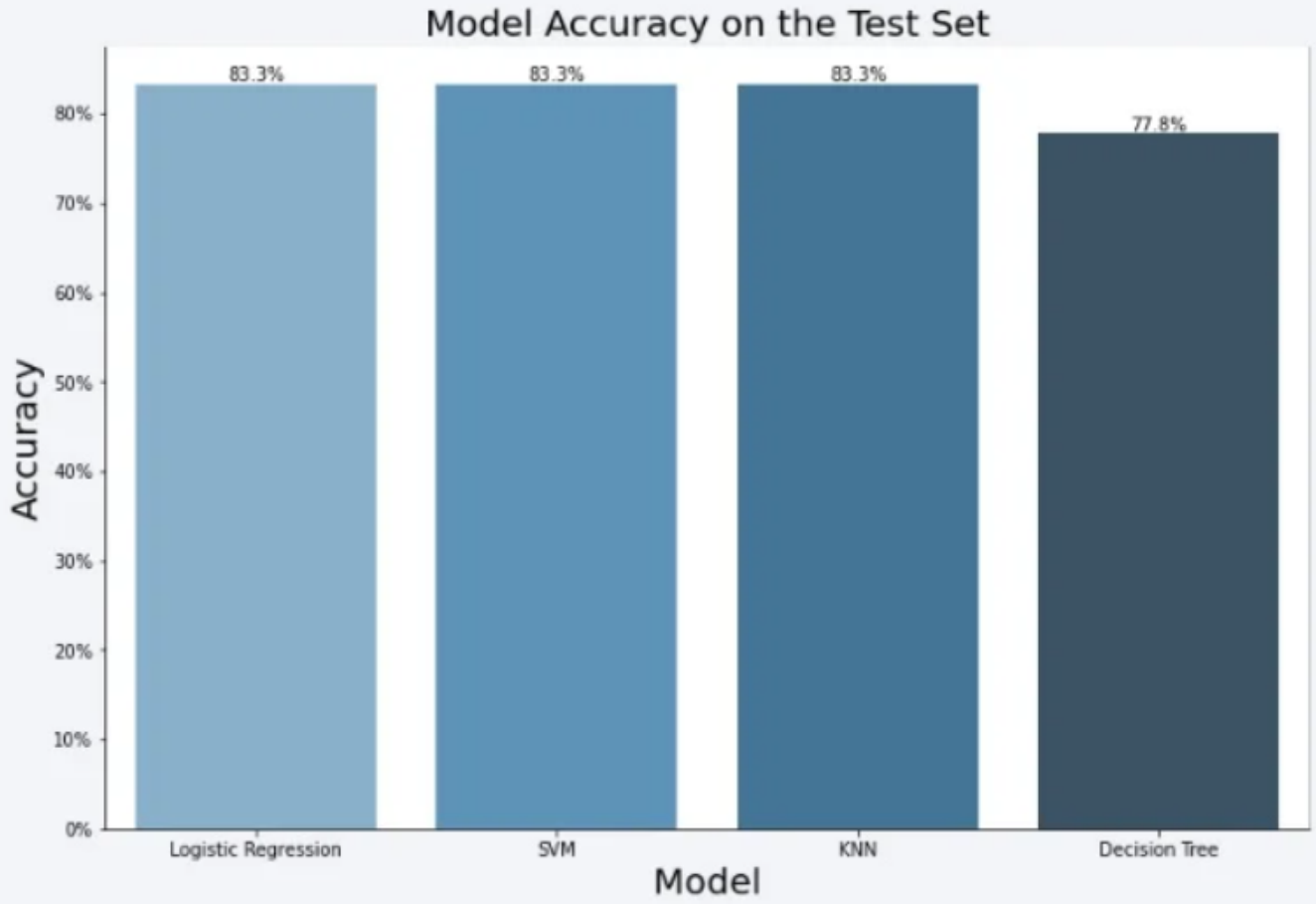


Section 5

Predictive Analysis (Classification)

Classification Accuracy

The DecisionTree Algorithm is the model with the highest classification accuracy of 88.75%



Confusion Matrix

The confusion matrix of the decision tree classifier show that it can distinguish between the different classes. The main problem is that false positives which are unsuccessful landings marked wrongly as successful.

Conclusions

We can conclude that:

The Decision Tree Classifier Algorithm is the best Machine Learning approach for this dataset.

The low weighted payloads (which define as 4000kg and below) resulted in more successful launches.

From the year 2013 to 2020, the success rate of SpaceX launches has continued to increase.

Launch site KSC LC-39A has most successful launches of any sites (76.9%).

SSO orbit has the highest success rate.

Launch sites are generally near the equator and coastlines.

Appendix

- Github repository: https://github.com/MAElliottWilms/FinalProjectSpaceY

Thank you!