Winning Space Race with Data Science

- Cherry O'Connell

- 01 June 2023

Outline

- Executive Summary
- Introduction
- Methodology
- Results
- Conclusion

Executive Summary

- Summary of methodologies
    1. Data Collection through API and web scraping
    2. Data Wrangling
    3. Exploratory Analysis with SQL and Data Visualisation
    4. Interactive Visual Analytics with Folium
    5. Machine Learning Prediction
- Summary of all results
    1. Exploratory Data Analysis results
    2. Interactive Analytics results/screenshots
    3. Machine Learning Predictions

• Introduction

The Space Exploration Technologies Corporation, most commonly known as SpaceX, manufactures spacecraft, launchers and satellite communications. SpaceX advertises Falcon 9 rocket launches on its website, with a cost of 62 million dollars; other providers cost upward of 165 million dollars each, much of the savings is because SpaceX can reuse the first stage. Therefore if we can determine if the first stage will land, we can determine the cost of a launch. This information can be used if an alternate company wants to bid against SpaceX for a rocket launch.

So instead of using rocket science to determine if the first stage will land successfully, we will train a machine learning model and use public information to predict if SpaceX will reuse the first stage. Other questions we aim to answer on this project is how variables such as launch site or location, number of flights, etc. and their relationship with each other affect the success or failure of the first stage landing, and what's the best machine learning algorithm to use among others.

- Section 1

Methodology

Methodology

- Executive Summary
- Data collection methodology:
  - Data was collected using SpaceX API and web scraping Wikipedia
- Perform data wrangling
  - Data was filtered and missing values were dealt with, and data was processed using one-note encoding
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium
- Perform predictive analysis using classification models
  - How to build, tune, evaluate classification models

• Data Collection

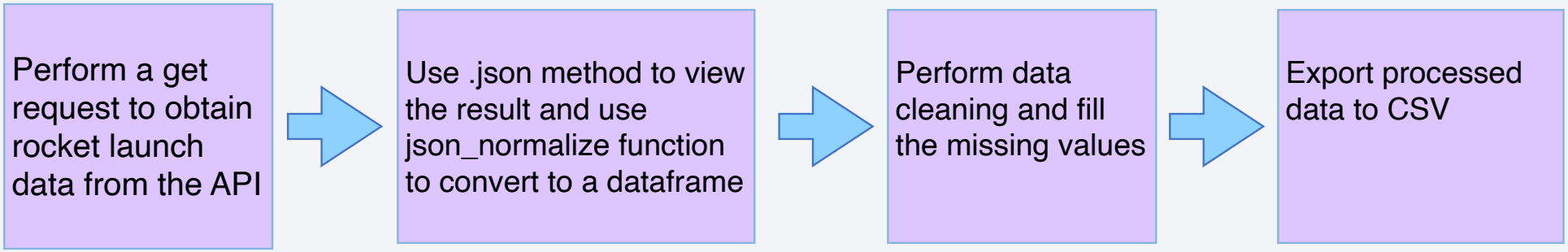
As a first step, we need to gather data that is needed to answer our questions or solve the problems presented in this project as mentioned in the Introduction.

There are two processes used in this project:

SpaceX REST API where we initialised a request, received the response content and then transformed it into our Pandas dataframe.

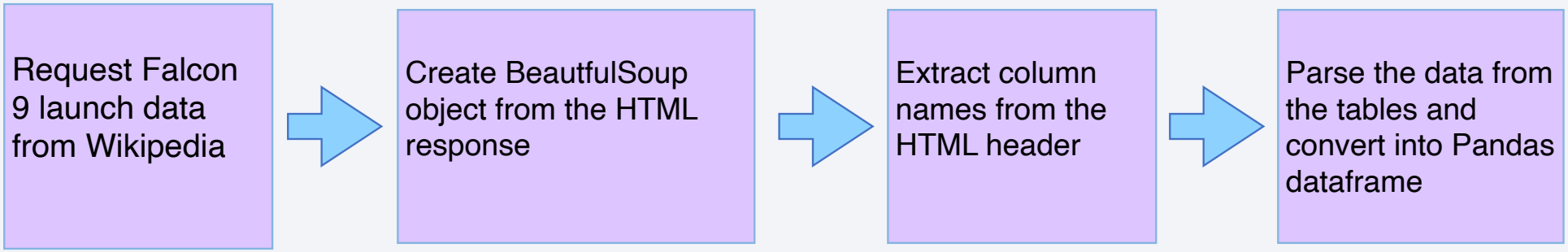
Wikipedia web scraping where we used the Python package, BeautifulSoup to web scrape some web tables that contain valuable Falcon 9 launch records. We then parsed the data from those tables and converted them into a Pandas dataframe for further visualisation and analysis.

• Data Collection – SpaceX API



GitHub URL: https://github.com/cheroconnell/Applied-Data-Science-Capstone/blob/main/01.%20jupyter-labs-spacex-data-collection-api.ipynb

• Data Collection - Scraping



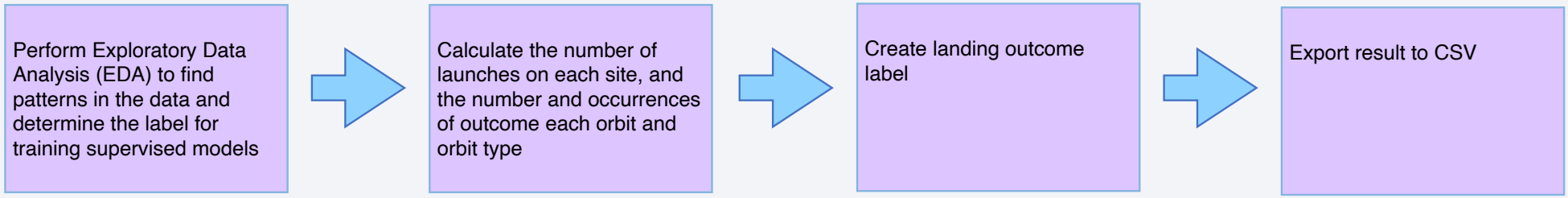
GitHub URL: https://github.com/cheroconnell/Applied-Data-Science-Capstone/blob/main/02.%20jupyter-labs-webscraping.ipynb

• Data Wrangling

In data wrangling, we will perform some Exploratory Data Analysis (EDA) to find some patterns in the data and determine what would be the label for training supervised models.

In the data set, there are several different cases where the booster did not land successfully. Sometimes a landing was attempted but failed due to an accident; for example, True Ocean means the mission outcome was successfully landed to a specific region of the ocean while False Ocean means the mission outcome was unsuccessfully landed to a specific region of the ocean. True RTLS means the mission outcome was successfully landed to a ground pad False RTLS means the mission outcome was unsuccessfully landed to a ground pad.True ASDS means the mission outcome was successfully landed on a drone ship False ASDS means the mission outcome was unsuccessfully landed on a drone ship.

We will mainly convert those outcomes into Training Labels with 1 means the booster successfully landed 0 means it was unsuccessful.



GitHub URL: https://github.com/cheroconnell/Applied-Data-Science-Capstone/blob/main/02.%20jupyter-labs-webscraping.ipynb

• EDA with Data Visualization

Scatter plots, bar charts and line charts were plotted. We plotted graphs to find the relationship between the attributes between, for example, flight number and payload mass, flight number and launch site, payload mass and launch site, orbit type and success rate, among others.

We used scatter plots to find and display relationship between variables. These are useful in machine learning as they can be used to identify the relationship between features and the target variable, outliers and can be used to visualise the performance of our model.

Bar charts on the other hand, are used to compare our categorical data, show the distribution of data and identify trends.

Lastly, we used line charts to track changes over time and also identify trends

GitHub URL: https://github.com/cheroconnell/Applied-Data-Science-Capstone/blob/main/05.%20jupyter-labs-eda-dataviz.ipynb.jupyterlite.ipynb

• EDA with SQL

With Exploratory Analysis using SQL, we queried our dataset to and completed the following tasks:

- Display the names of the unique launch sites in the space mission
- Display 5 records where launch sites begin with the string 'CCA'
- Display the total payload mass carried by boosters launched by NASA (CRS)
- Display average payload mass carried by booster version F9 v1.1
- List the date when the first 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. Use a subquery
- List the records which will display the month names, failure landing_outcomes in drone ship ,booster versions, launch_site for the months in year 2015.
- Rank the count of successful landing_outcomes between the date 04-06-2010 and 20-03-2017 in descending order

GitHub URL: https://github.com/cheroconnell/Applied-Data-Science-Capstone/blob/main/04.%20jupyter-labs-eda-sql-coursera_sqllite.ipynb

• Build an Interactive Map with Folium

First, we marked all sites on a map:
- We added our circle as the marker, pop up label and text label of NASA Johnson Space Center using the latitude and longitude coordinates at each launch site

Secondly, we marked the success/failed launches for each site on the map:
- We assigned green marker if it is a success launch and red marker if it is a failure.

And lastly, we calculated the distances between a launch site to its proximities:
- We calculated distances of various and nearest landmarks near our chosen launch site. Landmarks such as the nearest highway, railway, coastline and city.

GitHub URL: https://github.com/cheroconnell/Applied-Data-Science-Capstone/blob/main/06.%20jupyter_launch_site_location.jupyterlite-2.ipynb

• Build a Dashboard with Plotly Dash

First, added a dropdown list to enable Launch Site selection.

Then, we added a pie chart to show the total successful launches count for all sites where if a specific launch site is selected, we showed the Success vs Failed counts for the site.

Afterwards, we added a slider to select the payload range.

And lastly, we added a scatter chart to show the correlation between payload and launch success.

GitHub URL: https://github.com/cheroconnell/Applied-Data-Science-Capstone/blob/main/07.%20spacex_dash_app.py

• Predictive Analysis (Classification)

In Predictive Analysis, we first performed Exploratory Data Analysis, determine our training labels and and the method that performs the best using the test data.

1. Load the data and create a NumPy array from the column Class in data

2. Standardise the data in X then reassign it to the variable X

3. Split the data X and Y into training and test data. Set the parameter test_size to 0.2 and random_state to 2

4. Create a GridSearchCV object with cv = 10. Fit the object to find the best parameters from the dictionary parameters.

5. Apply GridSearchCV object with cv = 10 on logreg_cv (Logistic Regression), svm_cv (Support Vector Machine), tree_cv (Decision Tree), and knn_cv (K-Nearest Neighbor)

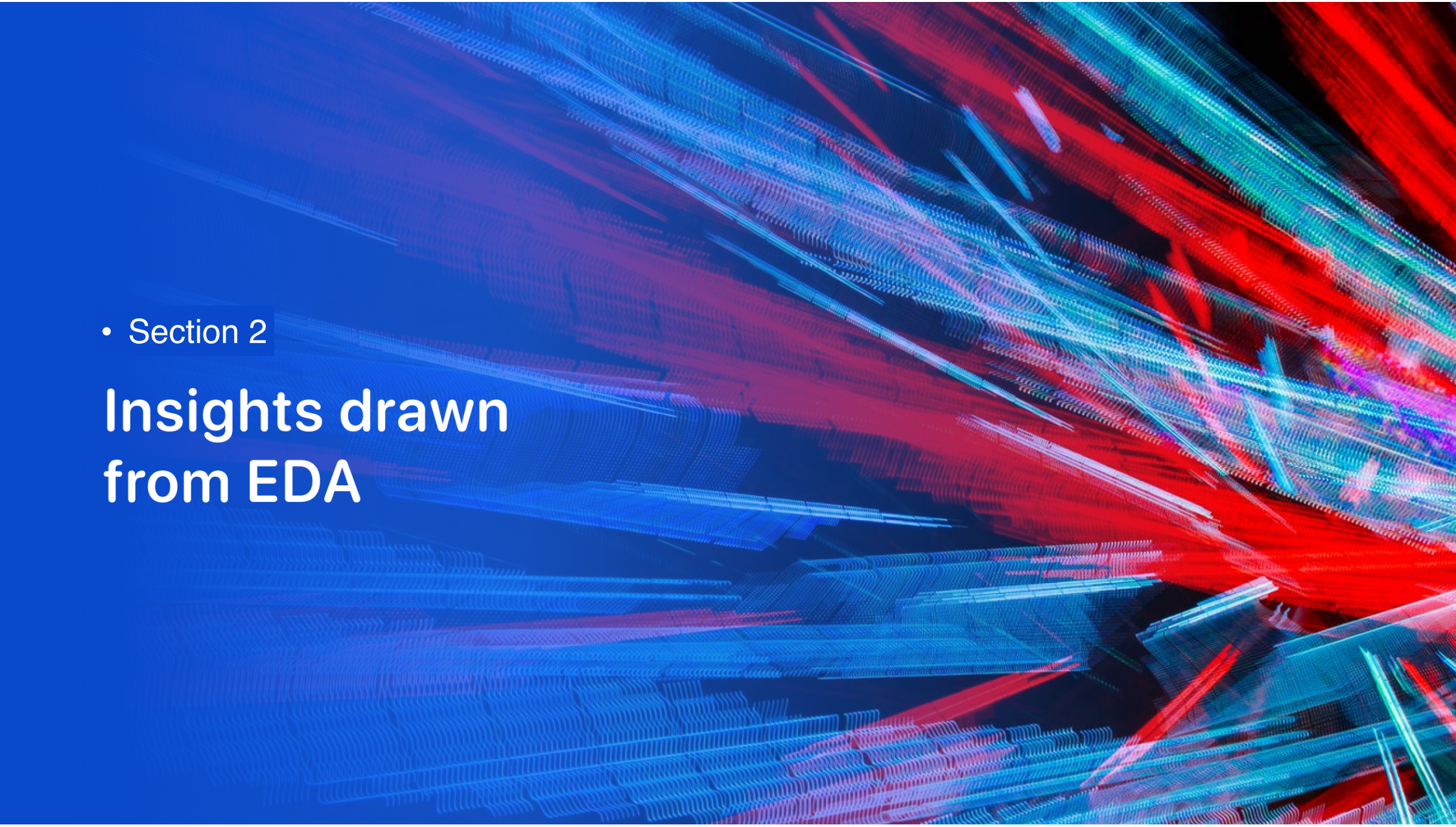
6. Calculate the accuracy on the test data

7. Find the method performs best

GitHub URL: https://github.com/cheroconnell/Applied-Data-Science-Capstone/blob/main/01.%20jupyter-labs-spacex-data-collection-api.ipynb

Results

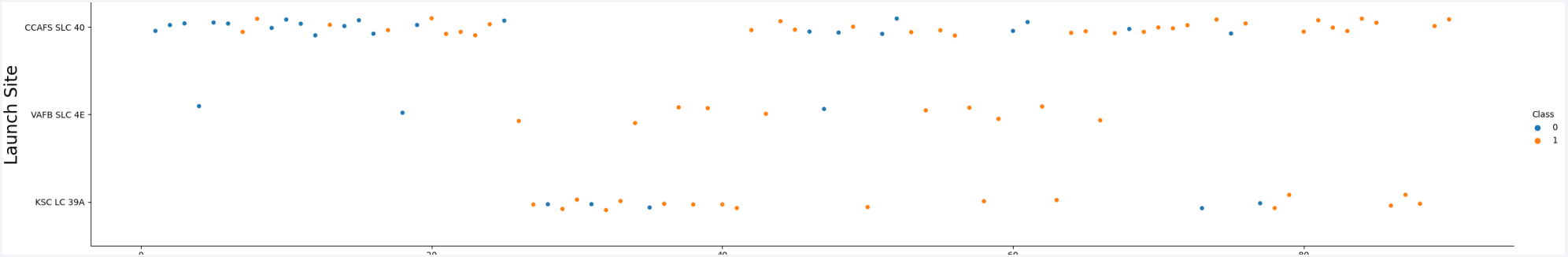
- Exploratory data analysis results
- Interactive analytics demo in screenshots
- Predictive analysis results



• Section 2

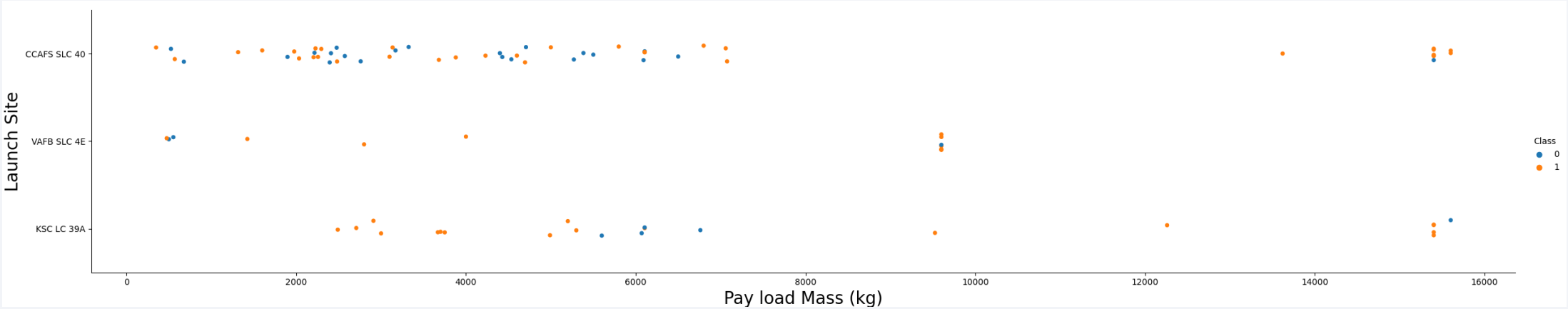
Insights drawn from EDA

• Flight Number vs. Launch Site



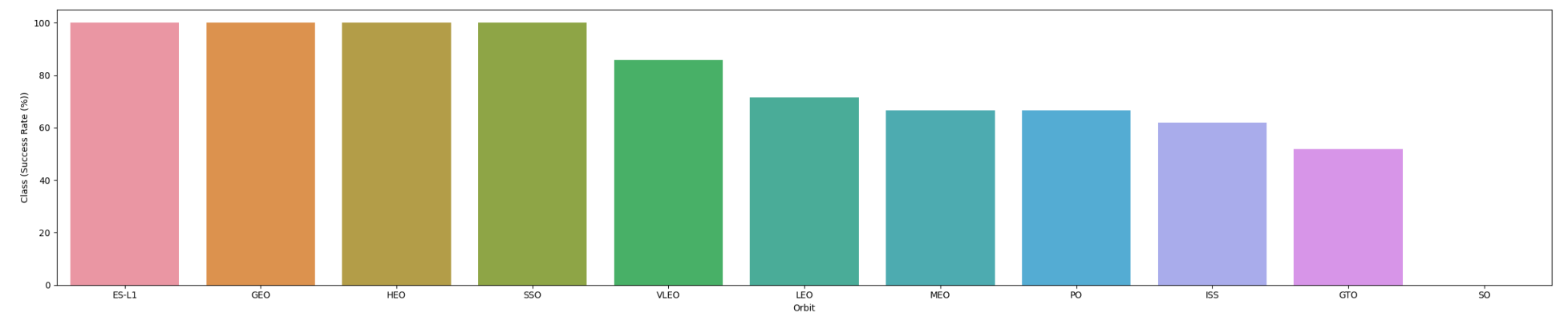
- We can see that different launch sites have different success rates.
- Blue dots mean failed flight, and orange dots mean successful flight. In this case, the earlier flights had a lower success rates and the latest flights are more successful.
- Most flights are from the CCAFS SLC 40 launch site which is in Florida, USA (east coast)
- The least number of flights launched are launched from VAFB SLC 4E site located in California, USA (west coast)
- We can assume that the newer flights had higher success rate

• Payload vs. Launch Site



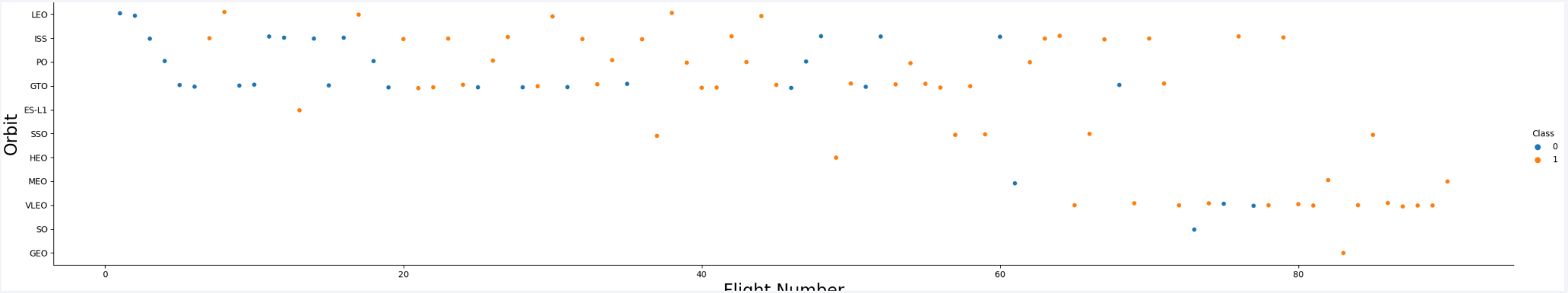
- Most launches with 8,000 pay load mass (kg) are successful
- VAFB SKC 4E have not launched a flight with pay load mass over 10,000 kg
- Nine out of 12 launches exceeding pay load mass of 12,000 kg are successful

• Success Rate vs. Orbit Type



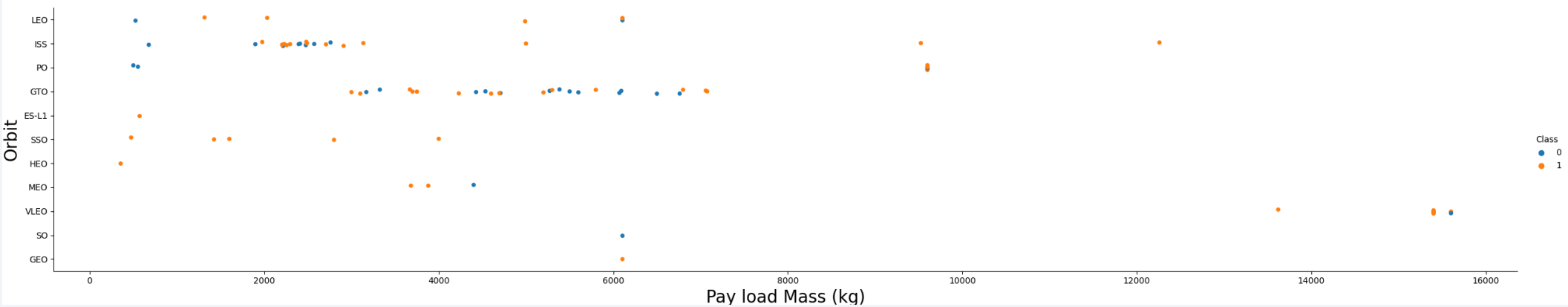
- There are 4 orbit types with 100% success rate - ES-L1, GEO (Geostationary orbit), HEO, (High earth orbit) and SSO (Sun-synchronous orbit)
- One orbit type (SO) has 0% success rate
- Six orbit types have success rates of 50%-85%

• Flight Number vs. Orbit Type



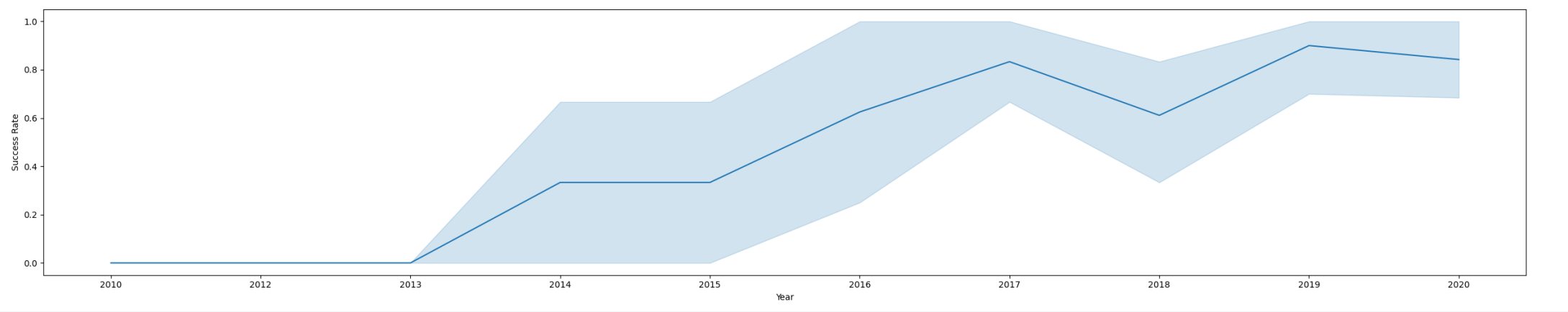
- Our plot shows that the success rate typically increases with the number of flights for each orbit.
- This is true specially for LEO orbit

• Payload vs. Orbit Type



- Heavy pay load mass (kg) have positive impact or better with three orbits LEO, ISS and PO orbit types
- SSO orbit types are all successful regardless of pay load mass

• Launch Success Yearly Trend



- Success rate of launches from 2013 until 2017 are in upward, and downward for a year but bounced back and 2019 had surpassed the success rate from 2017
- Overall, the rate of success each year has improved from 2013

• All Launch Site Names

```
In [7]: %sql SELECT DISTINCT LAUNCH_SITE AS "Launch_sites" FROM SPACEXTBL;
 * sqlite:///my_data1.db
Done.
```

Out[7]:

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

These are the names of all launch sites in the space mission which was a result from querying the dataset using SQL

• Launch Site Names Begin with 'CCA'

In [8]:
```
%sql SELECT * FROM 'SPACEXTBL' WHERE Launch_Site LIKE 'CCA%' LIMIT 5;
```

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

Out[8]:

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Lanc
06/04/2010	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0.0	LEO	SpaceX	Success	Failu
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	Failu
22/05/2012	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525.0	LEO (ISS)	NASA (COTS)	Success	
10/08/2012	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500.0	LEO (ISS)	NASA (CRS)	Success	
03/01/2013	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677.0	LEO (ISS)	NASA (CRS)	Success	

These are 5 records where launch sites begin with CCA which was a result from querying the dataset using SQL as shown above

• Total Payload Mass

```
In [9]: %sql SELECT SUM(PAYLOAD_MASS__KG_) AS "Payload Mass Kg", Customer FROM SPACEXTBL WHERE Customer = 'NASA (CRS)';
 * sqlite:///my_data1.db
Done.
```

Out[9]:

Payload Mass Kg	Customer
45596.0	NASA (CRS)

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

• Average Payload Mass by F9 v1.1

```
In [10]: %sql SELECT AVG(PAYLOAD_MASS__KG_) AS "Payload Mass Kg", Customer, Booster_version FROM SPACEXTBL WHERE Booster_V
 * sqlite:///my_data1.db
Done.
```

Out[10]:

Payload Mass Kg	Customer	Booster_Version
2534.6666666666665	MDA	F9 v1.1 B1003

This shows the average payload mass carried by boosters version F9 V1.1

- First Successful Ground Landing Date

```
In [11]: %sql SELECT MIN(Date) FROM SPACEXTBL WHERE "Landing_Outcome" = "Success (ground pad)";
 * sqlite:///my_data1.db
Done.
Out[11]: MIN(Date)
         01/08/2018
```

This is the date when the first successful landing outcome in ground pad was achieved

• Total Number of Successful and Failure Mission Outcomes

```
In [17]: %sql SELECT "Mission_Outcome", COUNT("Mission_Outcome") AS Total FROM SPACEXTBL GROUP BY "Mission_Outcome";
 * sqlite:///my_data1.db
Done.
```

Out[17]:

Mission_Outcome	Total
None	0
Failure (in flight)	1
Success	98
Success	1
Success (payload status unclear)	1

This displays the total number of successful and failed mission outcomes

• Boosters Carried Maximum Payload

```
In [18]: %sql SELECT "Booster_Version",Payload, "PAYLOAD_MASS__KG_" FROM SPACEXTBL WHERE "PAYLOAD_MASS__KG_" = (SELECT MAX
 * sqlite:///my_data1.db
Done.
```

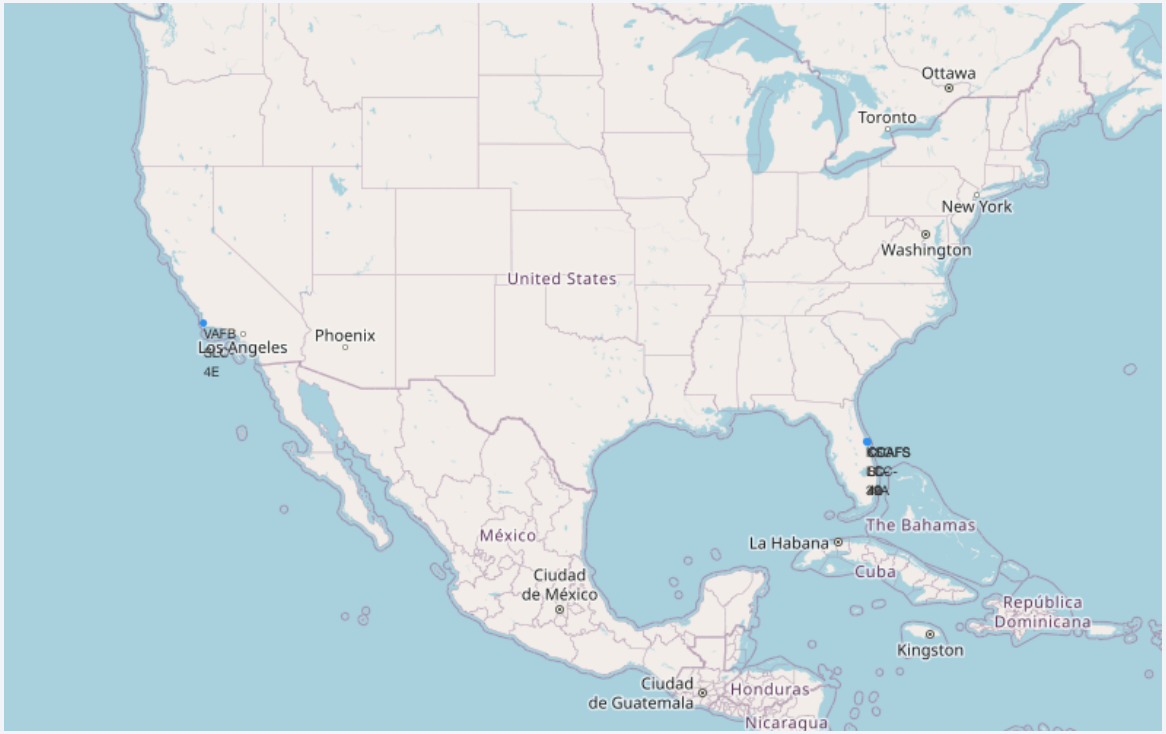
Out[18]:

Booster_Version	Payload	PAYLOAD_MASS__KG_
F9 B5 B1048.4	Starlink 1 v1.0, SpaceX CRS-19	15600.0
F9 B5 B1049.4	Starlink 2 v1.0, Crew Dragon in-flight abort test	15600.0
F9 B5 B1051.3	Starlink 3 v1.0, Starlink 4 v1.0	15600.0
F9 B5 B1056.4	Starlink 4 v1.0, SpaceX CRS-20	15600.0
F9 B5 B1048.5	Starlink 5 v1.0, Starlink 6 v1.0	15600.0
F9 B5 B1051.4	Starlink 6 v1.0, Crew Dragon Demo-2	15600.0
F9 B5 B1049.5	Starlink 7 v1.0, Starlink 8 v1.0	15600.0
F9 B5 B1060.2	Starlink 11 v1.0, Starlink 12 v1.0	15600.0
F9 B5 B1058.3	Starlink 12 v1.0, Starlink 13 v1.0	15600.0
F9 B5 B1051.6	Starlink 13 v1.0, Starlink 14 v1.0	15600.0
F9 B5 B1060.3	Starlink 14 v1.0, GPS III-04	15600.0
F9 B5 B1049.7	Starlink 15 v1.0, SpaceX CRS-21	15600.0

These are the names of the booster version which have carried a maximum payload mass


• Section 3
Launch Sites
Proximities Analysis

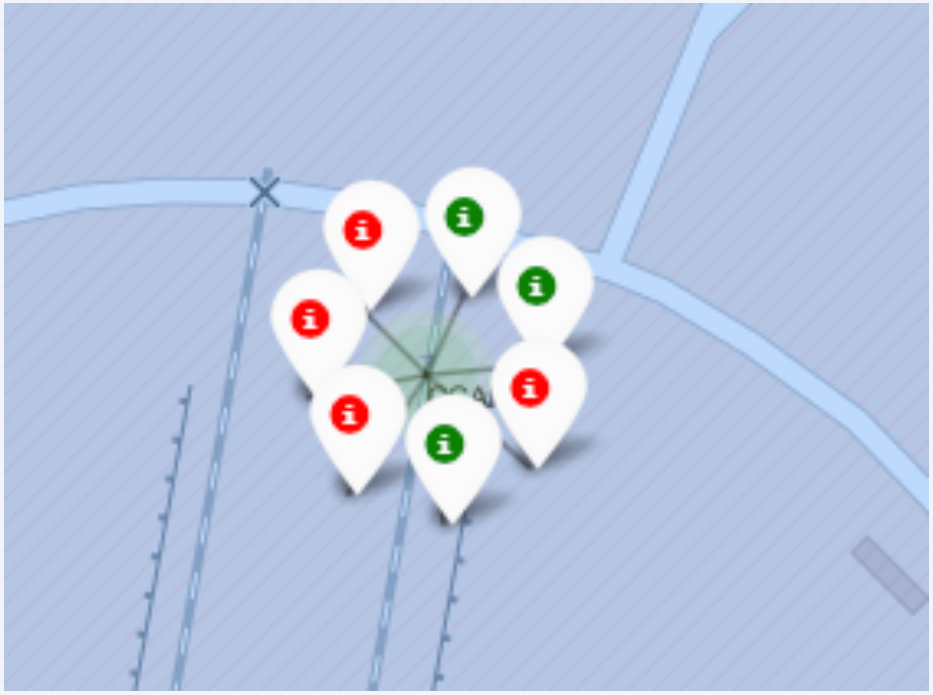
• Lauch Sites



Launch sites are in proximity to the equator which makes it easier to launch rockets to equatorial orbit.

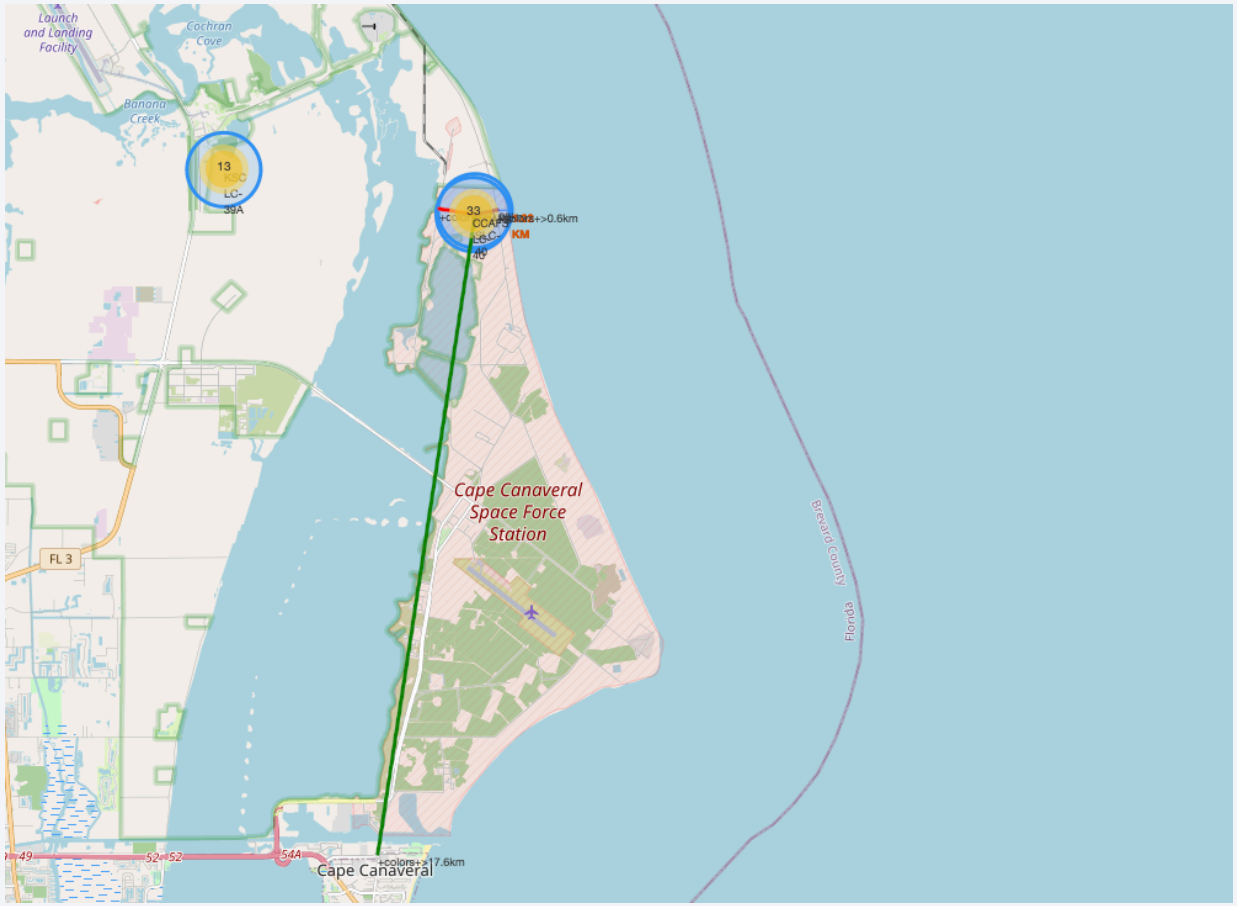
Also, the launch sites are near coastlines and other infrastructures such railways and highways, and but not too near the general population or cities.

• Success/Failed Site Launches



These are site launch markers that indicate whether a launch is a success (green) or fail (red) for a particular launch site which is in this case, CCAFS SLC-40 located in Florida, USA

• Launch Site and the closest city, railway, and highway



The chosen launch pad CCAFS SLC-40 in Florida, USA is at least 15km away from the nearest city. This is important to note that launch pads are not near the general population for safety reasons. On the other hand, there should be a nearby highway for easy access on the transportation of required equipment, services, and in case of emergency, etc. In this case, the nearest highway is just over half a kilometre. Same goes in case of railways which is under 1km.

• Section 4

Predictive Analysis (Classification)

• Classification Accuracy

```
In [33]:
algo = {'KNN':knn_cv.best_score_,'Decision Tree':tree_cv.best_score_,'Logistic Regression':logreg_cv.best_score_,
best_algo = max(algo, key= lambda x: algo[x])

print(best_algo, "method performed the best with a score of" ,algo[best_algo])
```

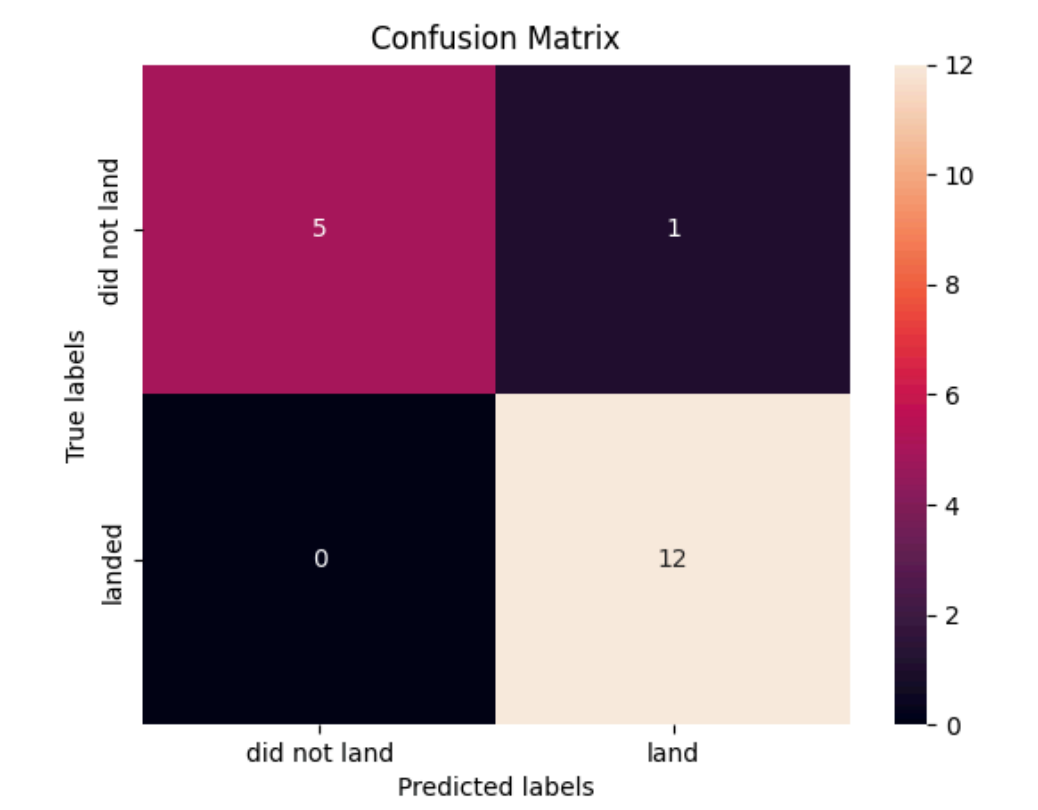
```
Decision Tree method performed the best with a score of 0.8857142857142858
```

```
In [34]:
algo
```

```
Out[34]: {'KNN': 0.8482142857142858,
 'Decision Tree': 0.8857142857142858,
 'Logistic Regression': 0.8464285714285713,
 'SVM': 0.8482142857142856}
```

All models have performed at about the same level, and the best one (Decision Tree) is just slightly better than the other three methods.

• Confusion Matrix



This is the confusion matrix for the accuracy of Decision Tree method on the test data. It is a summary of the performance of a classification algorithm.

• Conclusions

- We learned that different launch sites have different success rates.
- The earlier flights had a lower success rates and the latest flights are more successful.
- Most launches with 8,000 pay load mass (kg) are successful
- Launch sites are in proximity to the equator and are near coastlines and other infrastructures such railways and highways, and but not too near the general population or cities.
- There are 4 orbit types with 100% success rate - ES-L1, GEO (Geostationary orbit), HEO, (High earth orbit) and SSO (Sun-synchronous orbit), and that (SO) has 0% success rate
- Overall, the rate of success each year has improved from 2013
- And most importantly, Decision Tree method is the best algorithm for this data

Thank you!