

Winning Space Race with Data Science

<Kathrine Zhang>
<11-28-2022>



Outline

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

Executive Summary

Summary of methodologies

1. Data collection
2. Data wrangling
3. Exploratory Data Analysis with Data Visualization
4. Exploratory Data Analysis with SQL –
5. Building an interactive map with Folium
6. Building a Dashboard with Plotly Dash
7. Predictive analysis (Classification)

Summary of all results

1. Exploratory Data Analysis results
2. Interactive analytics in screenshots
3. Predictive analysis results

Introduction

SpaceX is a revolutionary company who has disrupted the space industry by offering rocket launches specifically Falcon 9 as low as 62 million dollars; while other providers cost upward of 165 million dollars each. Most of this saving thanks to SpaceX's astounding idea to reuse the first stage of the launch by re-landing the rocket to be used on the next mission. Repeating this process will make the price even further. As a data scientist of a startup rivaling SpaceX, the goal of this project is to create the machine learning pipeline to predict the landing outcome of the first stage in the future. This project is crucial in identifying the right price to bid against SpaceX for a rocket launch.

The problems included:

- Identifying all factors that influence the landing outcome.
- The relationship between each variables and how it is affecting the outcome.
- The best condition 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 scrapping from Wikipedia
- Perform data wrangling
 - Data was processed using one-hot encoding for 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
 - How to build, tune, evaluate classification models

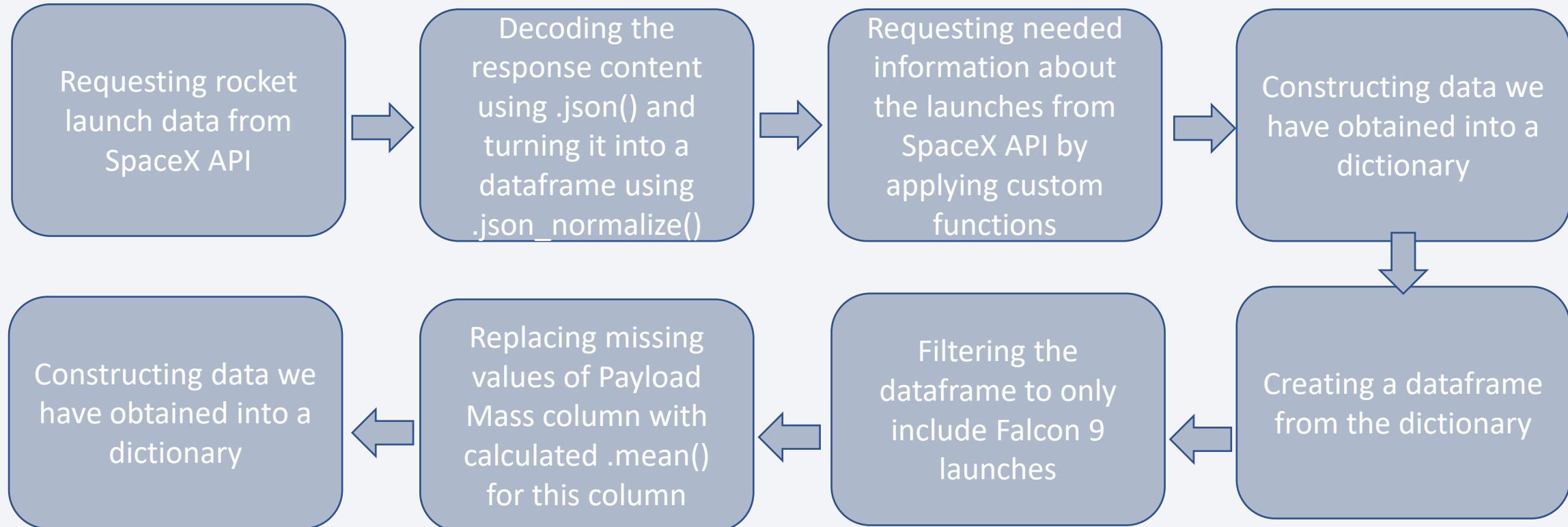
Data Collection

Data collection is the process of gathering and measuring information on targeted variables in an established system, which then enables one to answer relevant questions and evaluate outcomes. As mentioned, the dataset was collected by REST API and Web Scrapping from Wikipedia

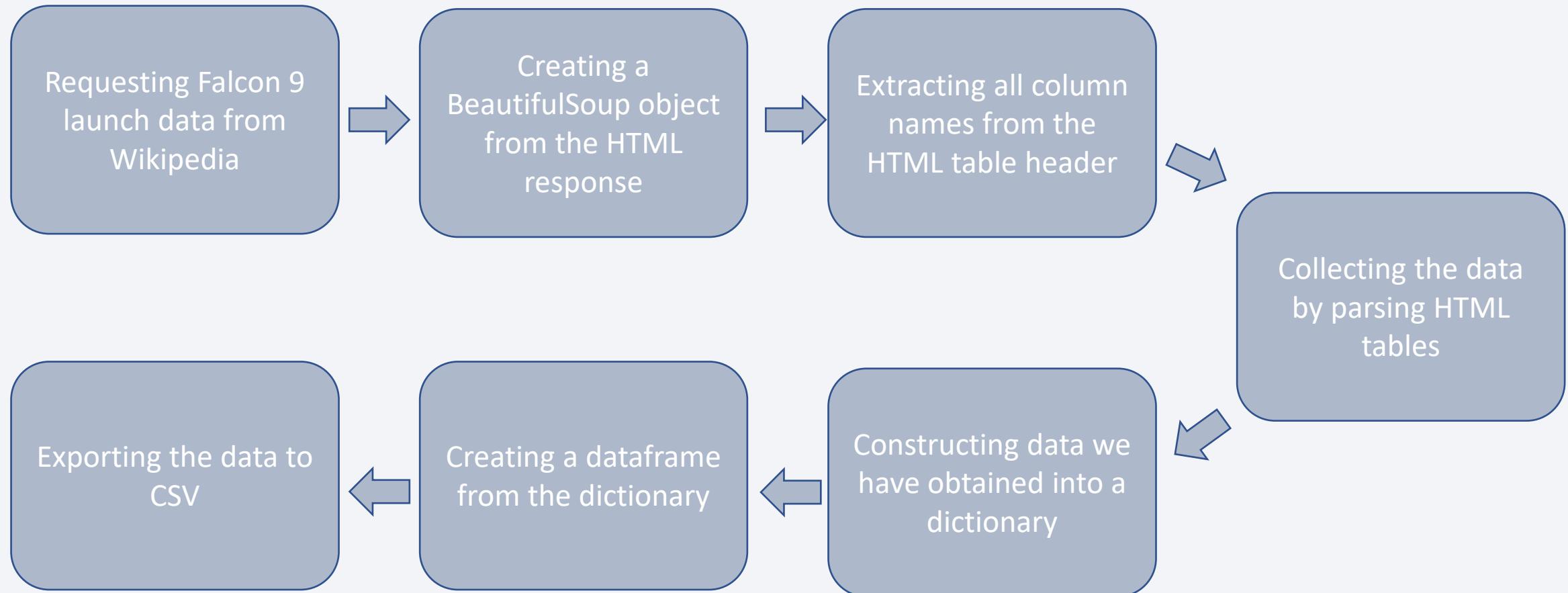
For REST API, its started by using the get request. Then, we decoded the response content as Json and turn it into a pandas dataframe using `json_normalize()`. We then cleaned the data, checked for missing values and fill with whatever needed.

For web scrapping, we will use the BeautifulSoup to extract the launch records as HTML table, parse the table and convert it to a pandas dataframe for further analysis

Data Collection – SpaceX API



Data Collection – Web Scraping



Data Wrangling

Perform exploratory Data Analysis and determine Training Labels

Calculate the number of launches on each site

Calculate the number and occurrence of each orbit

Calculate the number and occurrence of mission outcome per orbit type

Create a landing outcome label from Outcome column

Exporting the data to CSV

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 mainly convert those outcomes into Training Labels with “1” means the booster successfully landed, “0” means it was unsuccessful.

EDA with Data Visualization

Data Wrangling is the process of cleaning and unifying messy and complex data sets for easy access and Exploratory Data Analysis (EDA).

We will first calculate the number of launches on each site, then calculate the number and occurrence of mission outcome per orbit type.

We then create a landing outcome label from the outcome column. This will make it easier for further analysis, visualization, and ML. Lastly, we will export the result to a CSV.

EDA with SQL

- Using SQL, we had performed many queries to get better understanding of the dataset, Ex:
- - Displaying the names of the launch sites.
- - Displaying 5 records where launch sites begin with the string 'CCA'.
- - Displaying the total payload mass carried by booster launched by NASA (CRS).
- - Displaying the average payload mass carried by booster version F9 v1.1.
- - Listing the date when the first successful landing outcome in ground pad was achieved.
- - Listing the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000.
- - Listing the total number of successful and failure mission outcomes.
- - Listing the names of the booster_versions which have carried the maximum payload mass.
- - Listing the failed landing_outcomes in drone ship, their booster versions, and launch sites names for in year 2015.
- - Rank the count of landing outcomes or success between the date 2010-06-04 and 2017-03-20, in descending order.

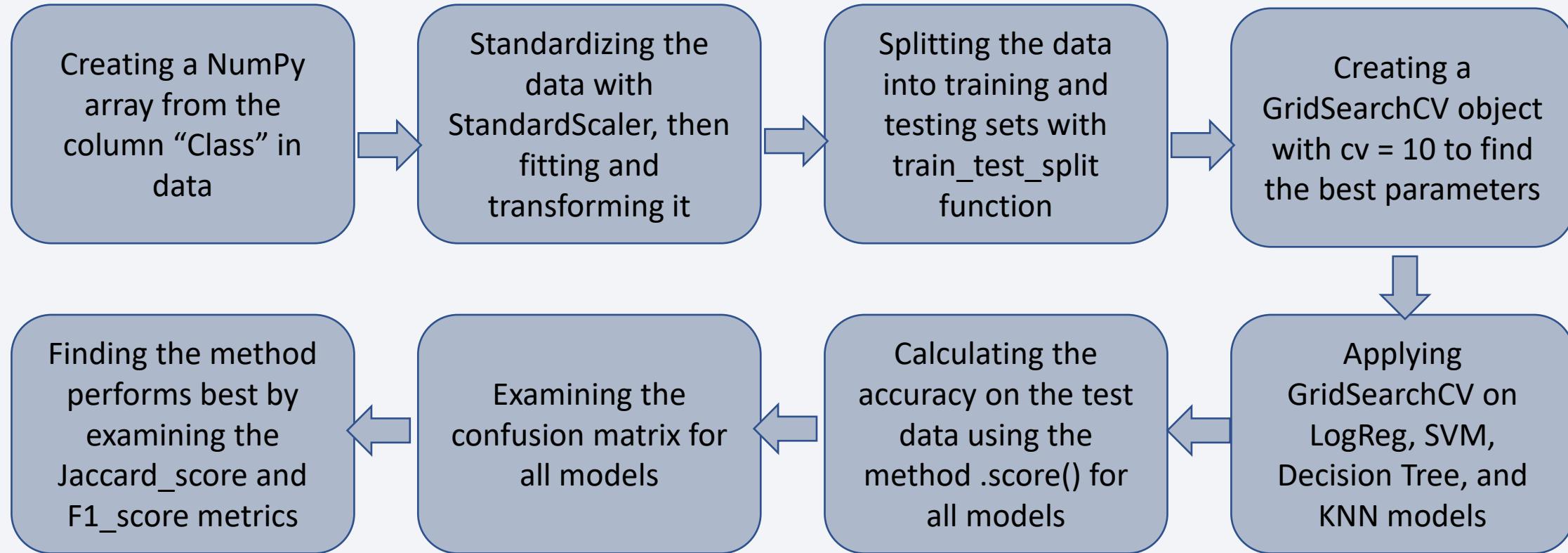
Build an Interactive Map with Folium

- To visualize the launch data into an interactive map. We took the latitude and longitude
- coordinates at each launch site and added a circle marker around each launch site with a
- label of the name of the launch site.
- We then assigned the dataframe `launch_outcomes(failure,success)` to classes 0 and 1 with
- **Red** and **Green** markers on the map in `MarkerCluster()`.
- We then used the Haversine's formula to calculated the distance of the launch sites to
- various landmark to find answer to the questions of:
- • How close the launch sites with railways, highways and coastlines?
- • How close the launch sites with nearby cities?

Build a Dashboard with Plotly Dash

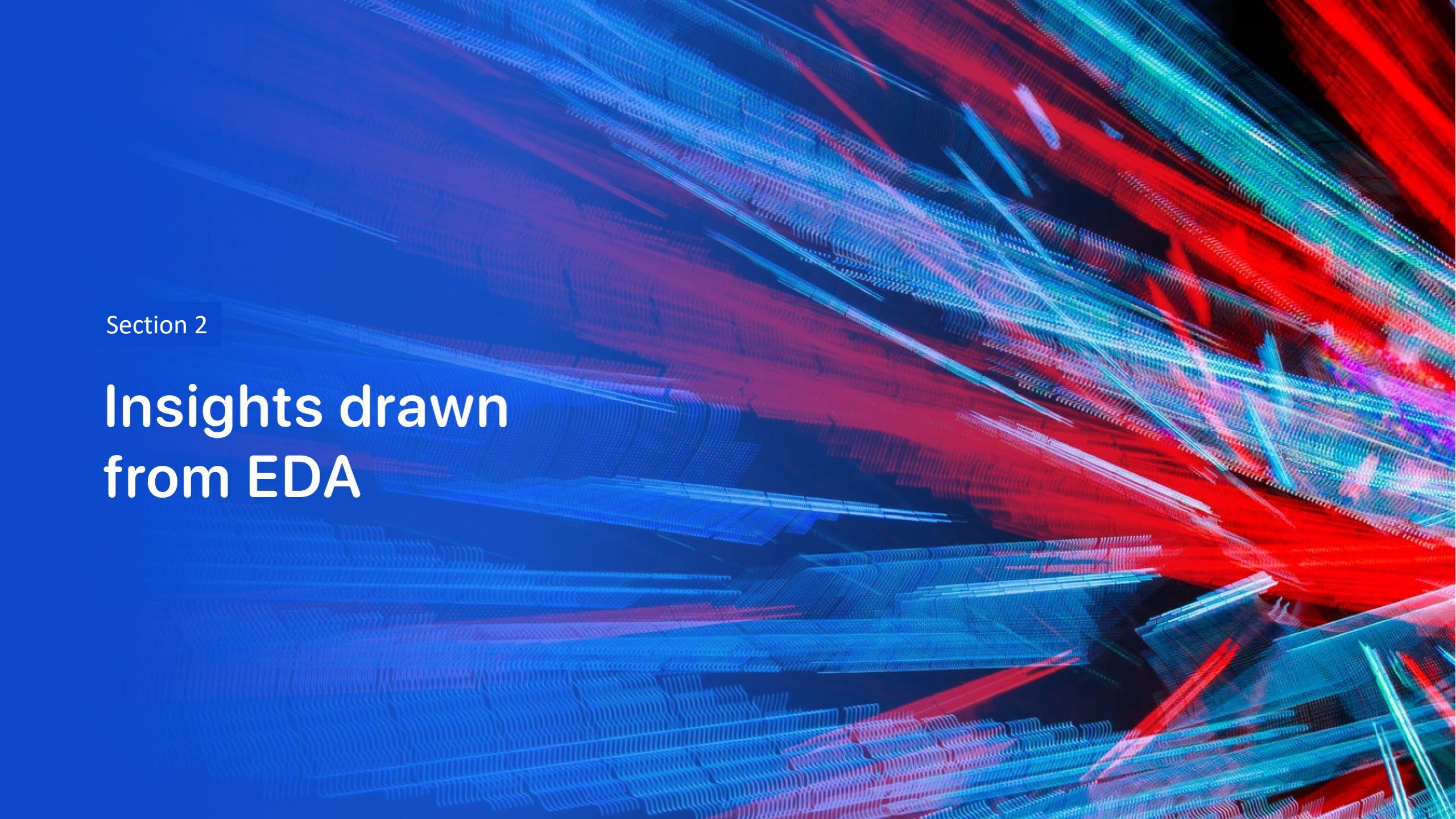
- We built an interactive dashboard with Plotly dash which allowing the user to play around with the data as they need.
- We plotted pie charts showing the total launches by a certain sites.
- We then plotted scatter graph showing the relationship with Outcome and Payload Mass (Kg) for the different booster version.

Predictive Analysis (Classification)



Results

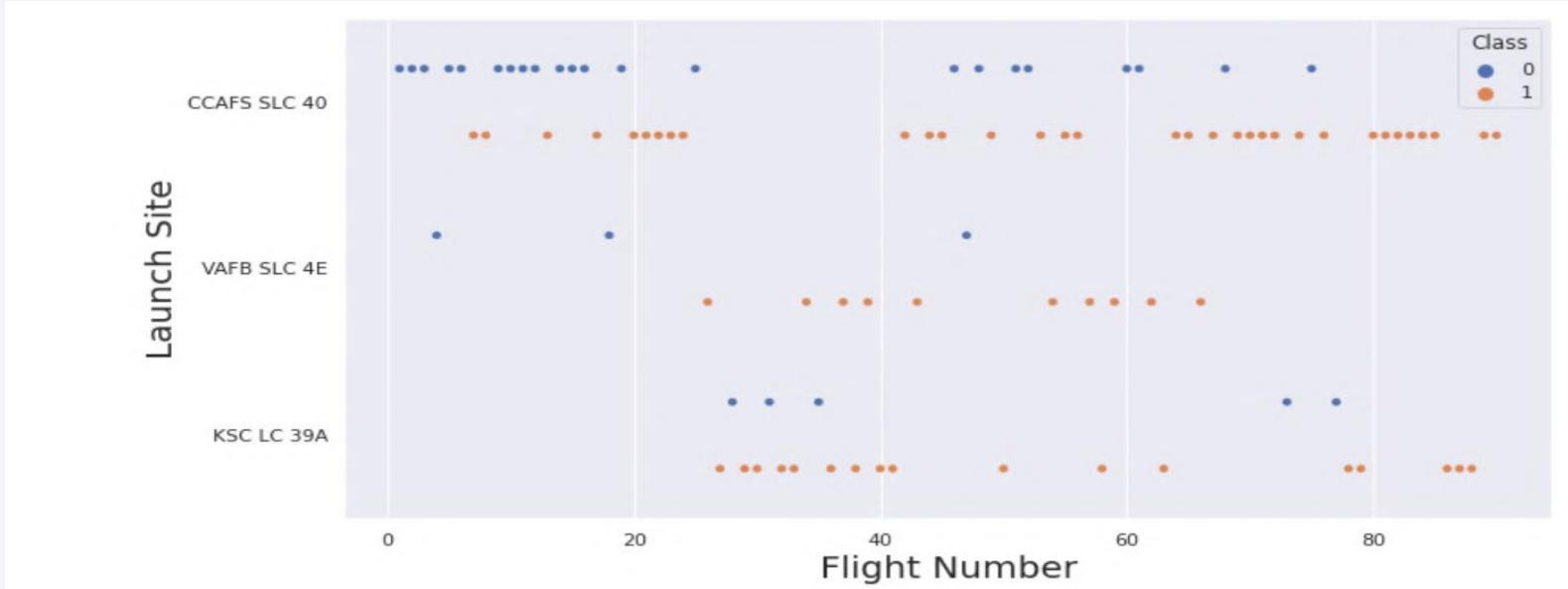
- Exploratory data analysis results
- Interactive analytics demo in screenshots
- Predictive analysis results

The background of the slide features a complex, abstract digital visualization. It consists of numerous thin, glowing lines that create a sense of depth and motion. The lines are primarily blue and red, with some green and purple highlights. They form a grid-like structure that curves and twists across the frame, resembling a 3D wireframe or a microscopic view of a complex system. The overall effect is futuristic and dynamic.

Section 2

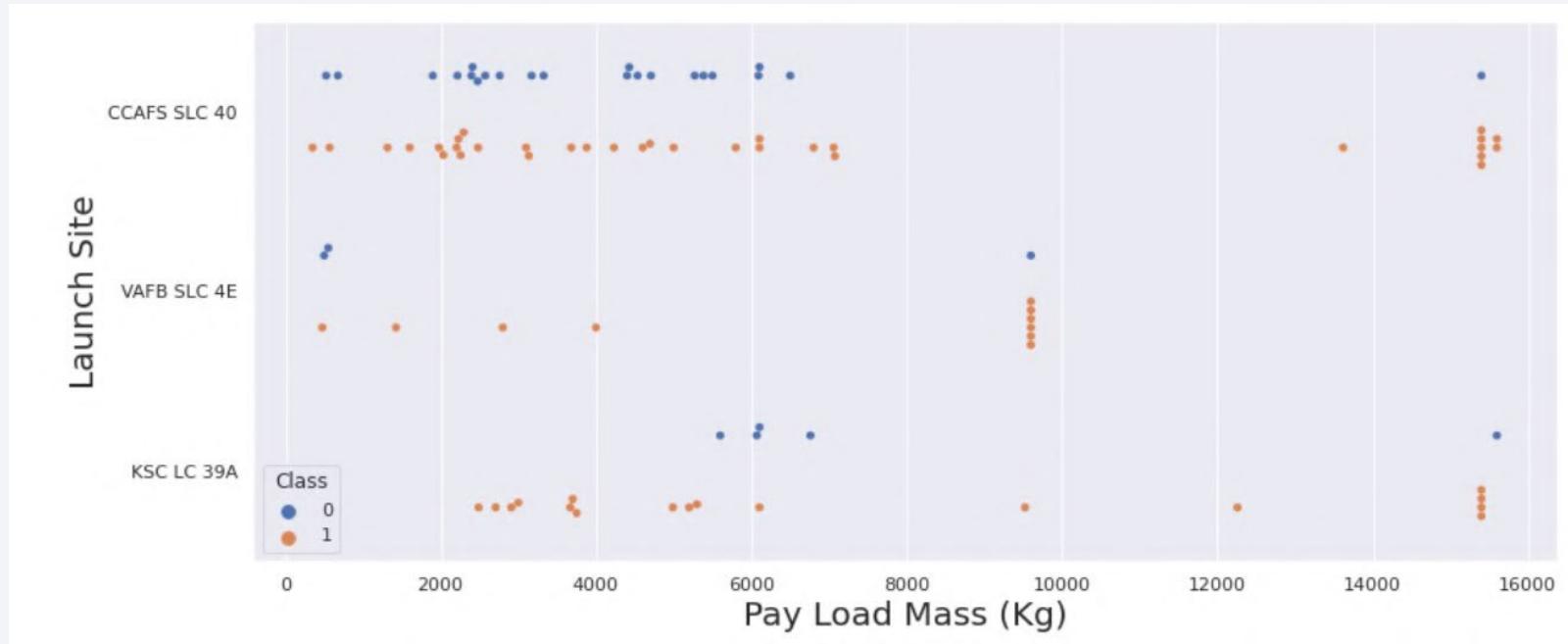
Insights drawn from EDA

Flight Number vs. Launch Site



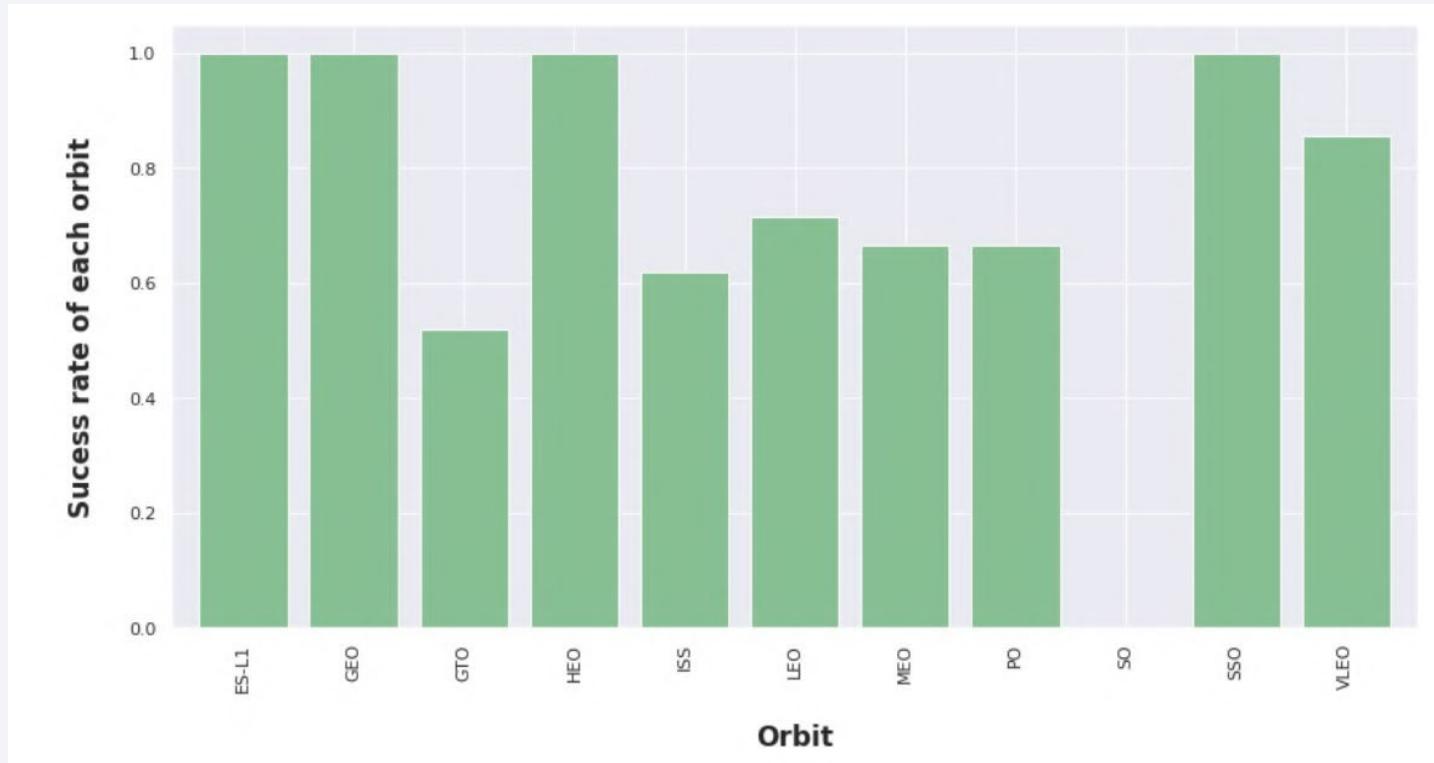
- The earliest flights all failed while the latest flights all succeeded.
- The CCAFS SLC 40 launch site has about a half of all launches.
- VAFB SLC 4E and KSC LC 39A have higher success rates.
- It can be assumed that each new launch has a higher rate of success.

Payload vs. Launch Site



This scatter plot shows once the pay load mass is greater than 7000kg, the probability of the success rate will be highly increased. However, there is no clear pattern to say the launch site is dependent to the pay load mass for the success rate.

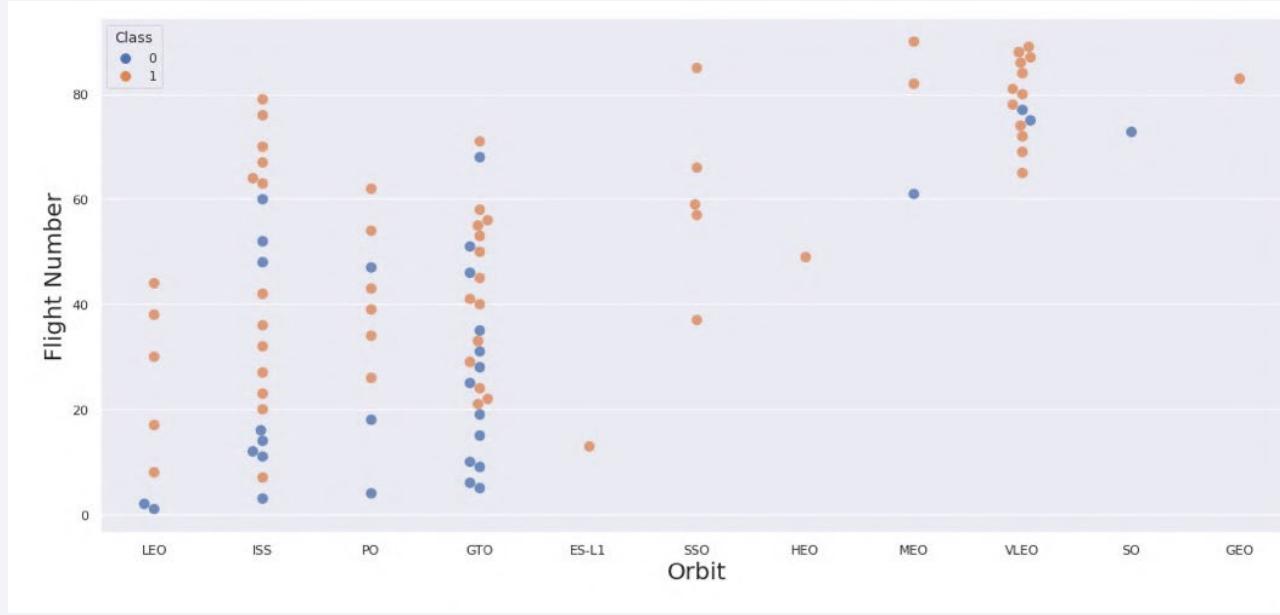
Success Rate vs. Orbit Type



This figure depicted the possibility of the orbits to influences the landing outcomes as some orbits has 100% success rate such as SSO, HEO, GEO AND ES-L1 while SO orbit produced 0% rate of success.

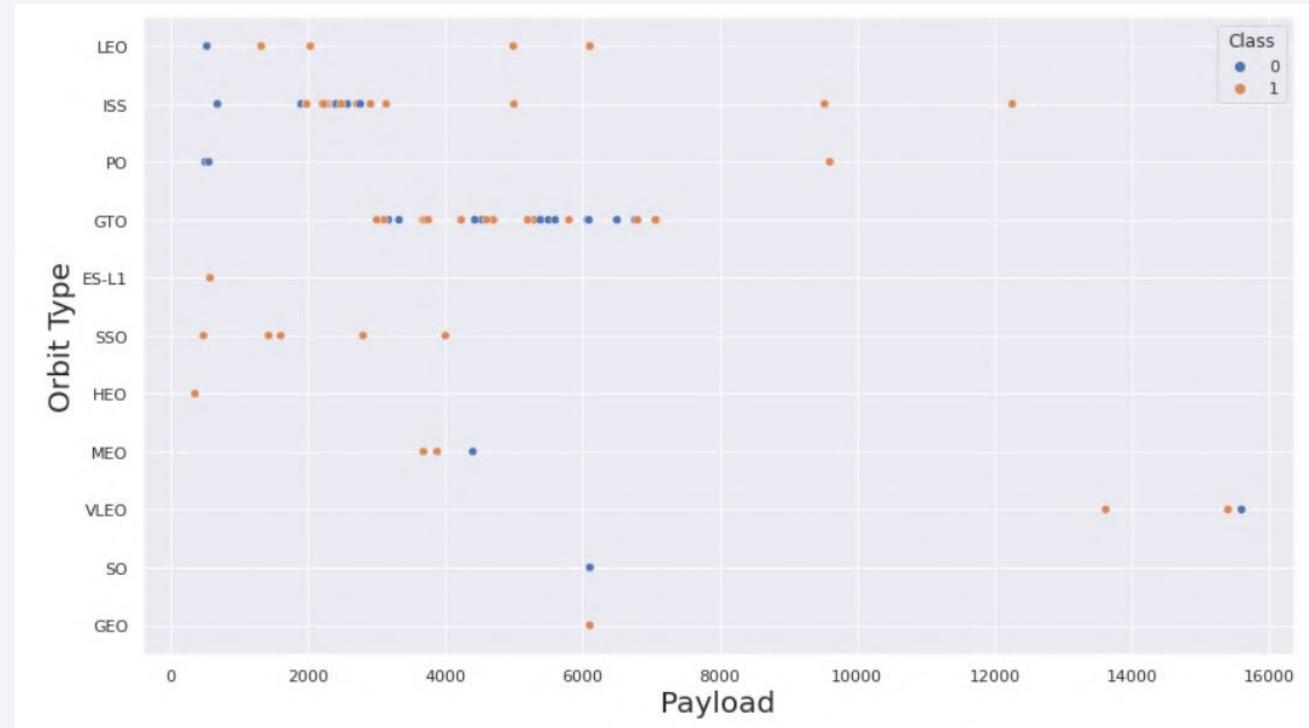
However, deeper analysis show that some of this orbits has only 1 occurrence such as GEO, SO, HEO and ES-L1 which mean this data need more dataset to see pattern or trend before we draw any conclusion.

Flight Number vs. Orbit Type



This scatter plot shows that generally, the larger the flight number on each orbits, the greater the success rate (especially LEO orbit) except for GTO orbit which depicts no relationship between both attributes.
Orbit that only has 1 occurrence should also be excluded from above statement as it's needed more dataset.

Payload vs. Orbit Type



Heavier payload has positive impact on LEO, ISS and PO orbit. However, it has negative impact on MEO and VLEO orbit.

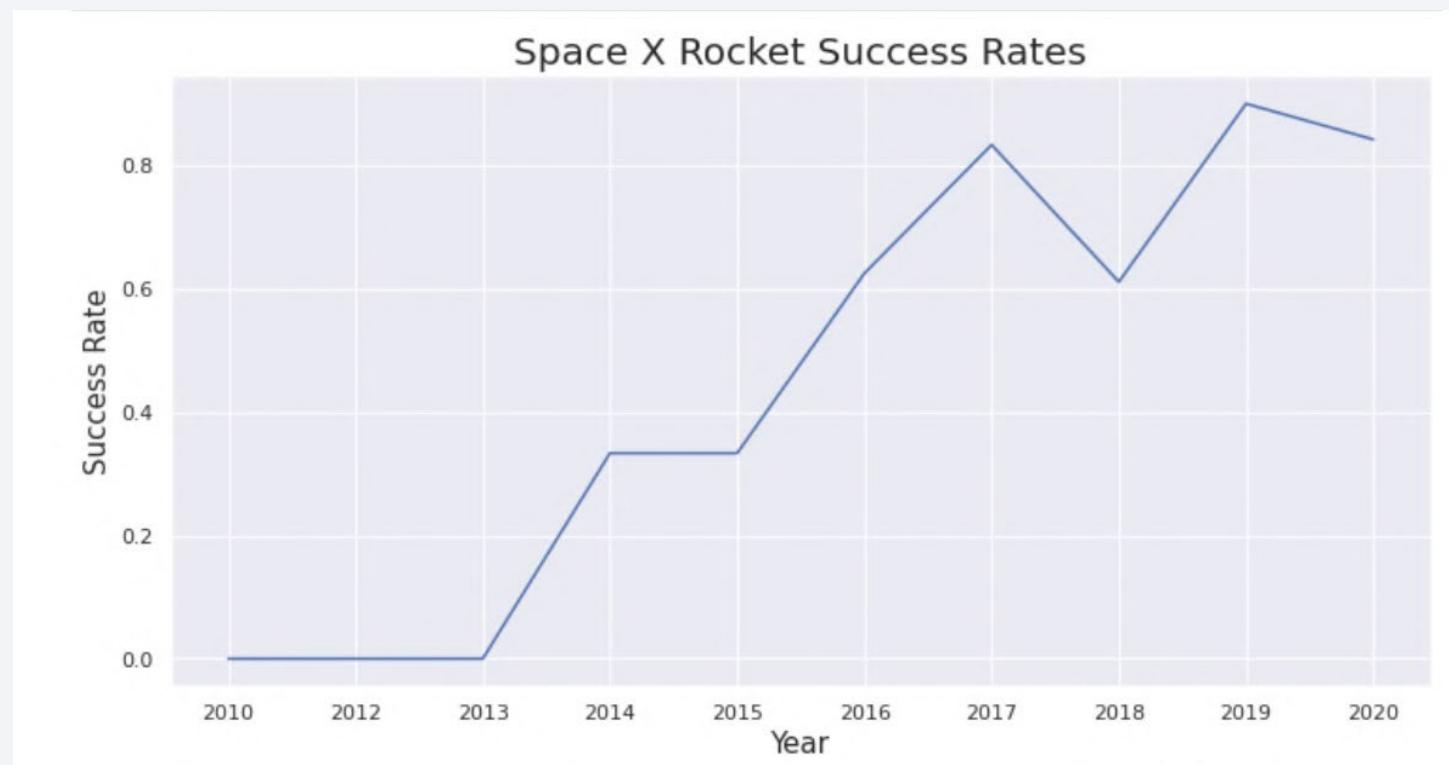
GTO orbit seem to depict no relation between the attributes.

Meanwhile, again, SO, GEO and HEO orbit need more dataset to see any pattern or trend.

Launch Success Yearly Trend

This figure clearly depicted an increasing trend from the year 2013 until 2020. If this trend continues for the next year onward.

The success rate will steadily increase until reaching 1/100% success rate.



All Launch Site Names

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

In [8]: %sql SELECT DISTINCT LAUNCH_SITE as "Launch_Sites" FROM SPACEXTBL;

* ibm_db_sa://xgb78019:***@b0aebb68-94fa-46ec-a1fc-1c999edb6187.c3n41cmd0nqnrk39u98g.databases.appdomain.cloud:31249/bludb
Done.

Out[8]: Launch_Sites
        CCAFS LC-40
        CCAFS SLC-40
        KSC LC-39A
        VAFB SLC-4E
```

Explanation: • Displaying the names of the unique launch sites in the space mission.

Launch Site Names Begin with 'CCA'

```
In [9]: %sql SELECT LAUNCH_SITE FROM SPACEXTBL WHERE LAUNCH_SITE LIKE 'CCA%' LIMIT 5;  
* ibm_db_sa://xgb78019:***@b0aebb68-94fa-46ec-a1fc-1c999edb6187.c3n41cmd0nqnrk39u98g.databases.appdomain.cloud:31249/bludb  
Done.  
Out[9]: launch_site  
CCAFS LC-40  
CCAFS LC-40  
CCAFS LC-40  
CCAFS LC-40  
CCAFS LC-40
```

Explanation: • Displaying 5 launch sites begin with the string 'CCA'.

Total Payload Mass

We calculated the total payload carried by boosters from NASA as 45596 using the query below

```
In [11]: %sql select sum(payload_mass_kg_) as total_payload_mass from SPACEXTBL where customer = 'NASA (CRS)';  
* ibm_db_sa://xgb78019:***@b0aebb68-94fa-46ec-a1fc-1c999edb6187.c3n41cmd0nqnrk39u98g.databases.appdomain.cloud:31249/bludb  
Done.  
Out[11]: total_payload_mass  
45596
```

Average Payload Mass by F9 v1.1

We calculated the average payload mass carried by booster version F9 v1.1 as 2928.4

```
In [12]: %sql SELECT AVG (PAYLOAD_MASS__KG_) FROM SPACEXTBL WHERE BOOSTER_VERSION = 'F9 v1.1';  
* ibm_db_sa://xgb78019:***@b0aebb68-94fa-46ec-a1fc-1c999edb6187.c3n41cmd0nqnrk39u98g.databases.appdomain.cloud:31249/bludb  
Done.  
Out[12]: 1  
2928
```

First Successful Ground Landing Date

We use the min() function to find the result

We observed that the dates of the first successful landing outcome on ground pad was 22nd December 2015

```
In [15]: %sql select min(date) as first_successful_landing from SPACEXTBL where landing__outcome = 'Success (ground pad)';

* ibm_db_sa://xgb78019:***@b0aebb68-94fa-46ec-a1fc-1c999edb6187.c3n41cmd0nqnrk39u98g.databases.appdomain.cloud:31249/bludb
Done.

Out[15]: first_successful_landing
2015-12-22
```

Successful Drone Ship Landing with Payload between 4000 and 6000

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

```
In [17]: %sql select booster_version from SPACEXTBL where landing__outcome = 'Success (drone ship)' and payload_mass_kg_ between 4000 and 6000;  
* ibm_db_sa://xgb78019:***@b0aebb68-94fa-46ec-a1fc-1c999edb6187.c3n41cmd0nqnrk39u98g.databases.appdomain.cloud:31249/bludb  
Done.  
Out[17]: booster_version  
F9 FT B1022  
F9 FT B1026  
F9 FT B1021.2  
F9 FT B1031.2
```

Total Number of Successful and Failure Mission Outcomes

List the total number of successful and failure mission outcomes

In [21]: `%sql select mission_outcome, count(*) as total_number from SPACEXTBL group by mission_outcome;`
* ibm_db_sa://xgb78019:***@b0aebb68-94fa-46ec-a1fc-1c999edb6187.c3n41cmd0nqnrk39u98g.databases.appdomain.cloud:31249/bludb
Done.

Out[21]:

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

Boosters Carried Maximum Payload

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

In [25]:

```
%sql select booster_version from SPACEXTBL where payload_mass_kg_ = (select max(payload_mass_kg_) from SPACEXTBL);
```

```
* ibm_db_sa://xgb78019:***@b0aebb68-94fa-46ec-a1fc-1c999edb6187.c3n41cmd0nqnrk39u98g.databases.appdomain.cloud:31249/bludb  
Done.
```

Out[25]: **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

We determined the booster that have carried the maximum payload using a subquery in the WHERE clause and the MAX() function

2015 Launch Records

We used a combinations of the WHERE clause, LIKE, AND, and BETWEEN conditions to filter for failed landing outcomes in drone ship, their booster versions, and launch site names for year 2015

In [26]:

```
%%sql select monthname(date) as month, date, booster_version, launch_site, landing__outcome from SPACEXTBL  
where landing__outcome = 'Failure (drone ship)' and year(date)=2015;
```

```
* ibm_db_sa://xgb78019:***@b0aebb68-94fa-46ec-a1fc-1c999edb6187.c3n41cmd0nqnrk39u98g.databases.appdomain.cloud:31249/bludb  
Done.
```

Out[26]:

MONTH	DATE	booster_version	launch_site	landing__outcome
October	2015-10-01	F9 v1.1 B1012	CCAFS LC-40	Failure (drone ship)
April	2015-04-14	F9 v1.1 B1015	CCAFS LC-40	Failure (drone ship)

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

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

In [27]:

```
%sql select landing_outcome, count(*) as count_outcomes from SPACEXTBL  
      where date between '2010-06-04' and '2017-03-20'  
      group by landing_outcome  
      order by count_outcomes desc;
```

```
* ibm_db_sa://xgb78019:***@b0aeabb68-94fa-46ec-a1fc-1c999edb6187.c3n41cmd0nqnrk39u98g.databases.appdomain.cloud:31249/bludb  
Done.
```

Out[27]:

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

We selected Landing outcomes and the COUNT of landing outcomes from the data and used the WHERE clause to filter for landing outcomes BETWEEN 2010-06-04 to 2010-03-20.

We applied the GROUP BY clause to group the landing outcomes and the ORDER BY clause to order the grouped landing outcome in descending order

The background of the slide is a photograph taken from space at night. It shows the curvature of the Earth against the dark void of space. City lights are visible as numerous small white and yellow dots, primarily concentrated in the lower right quadrant where the United States and Mexico would be. In the upper left quadrant, the green and blue glow of the aurora borealis (Northern Lights) is visible in the upper atmosphere.

Section 3

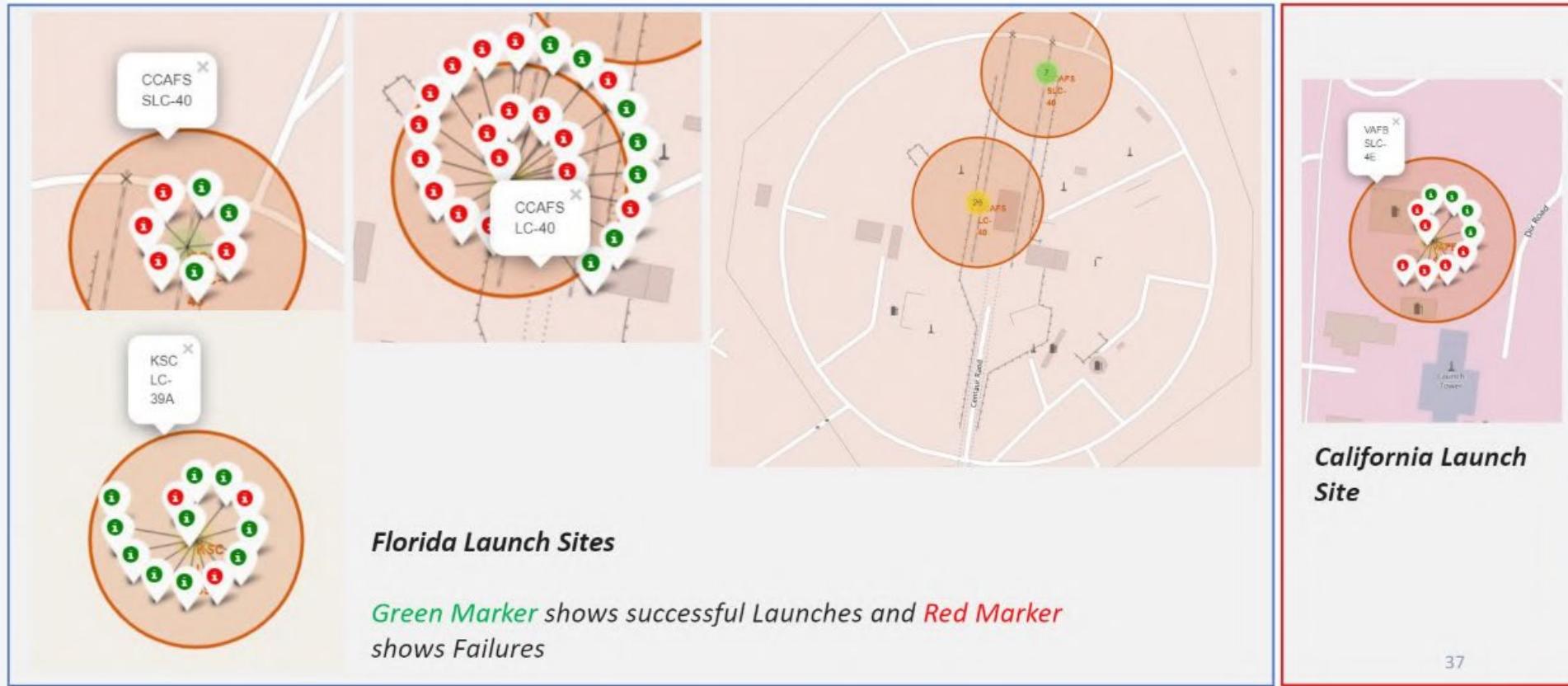
Launch Sites Proximities Analysis

Locations of Launch Sites

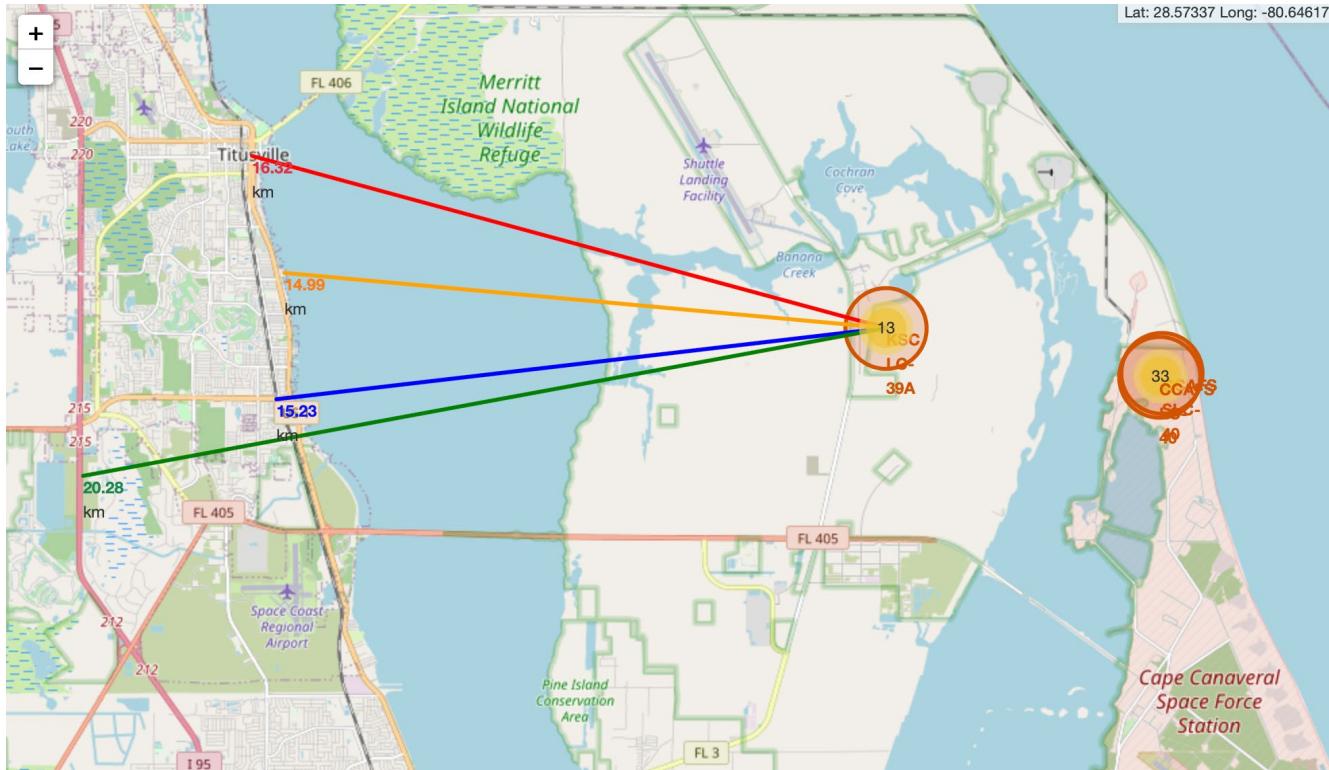


We can see that all the SpaceX launch sites are located inside the United States

Markers showing launch sites with color labels



Distance from the launch site KSC LC-39A to its proximities

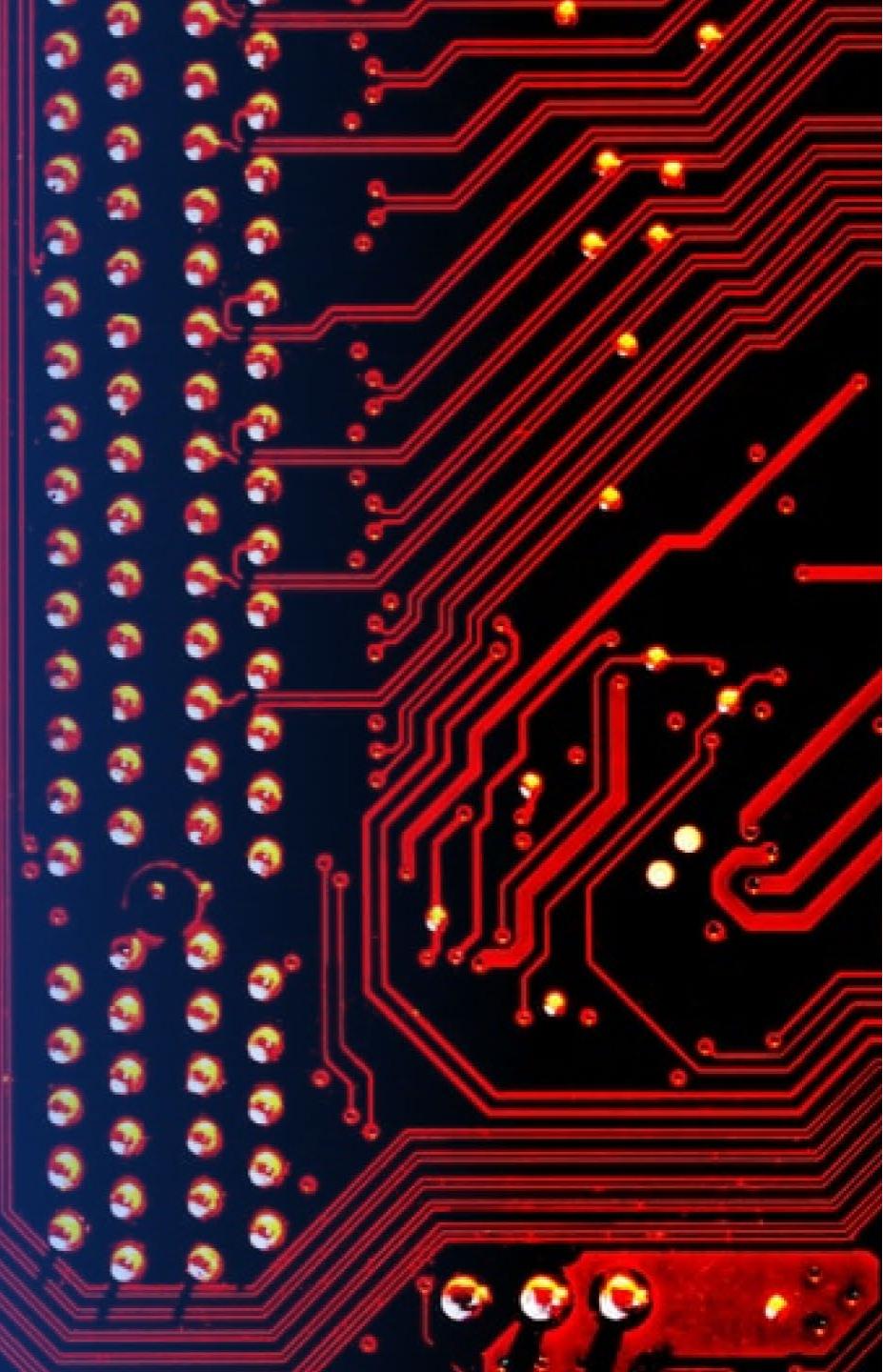


Explanation:

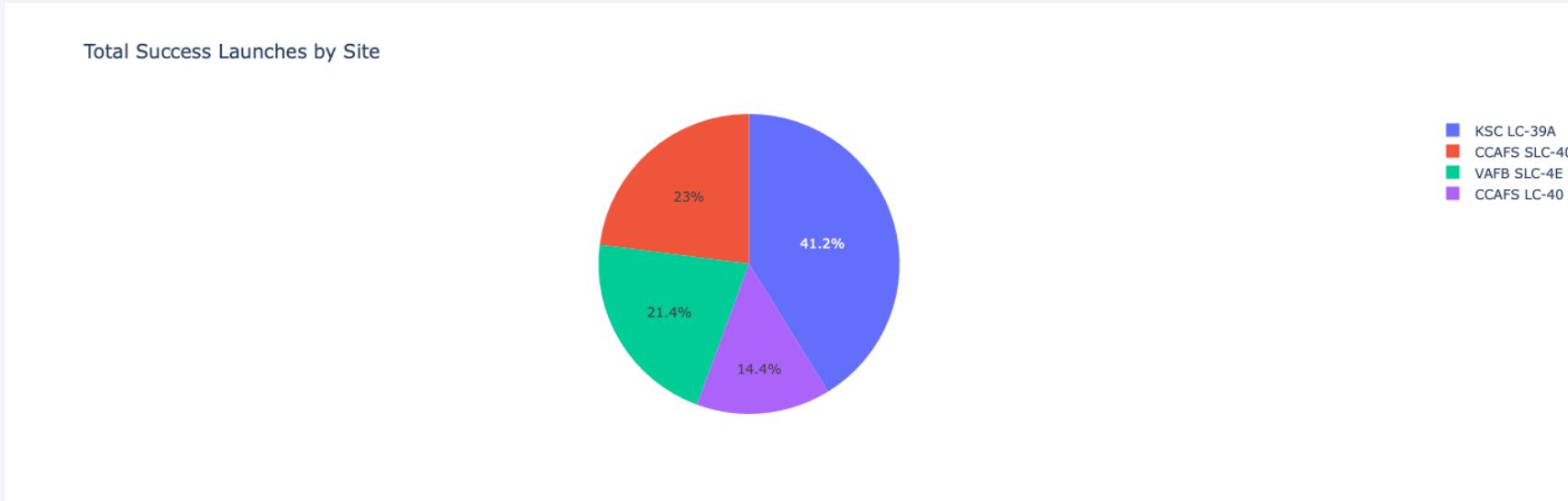
- From the visual analysis of the launch site KSC LC-39A we can clearly see that it is:-
 - relative close to railway (15.23 km) -
 - relative close to highway (20.28 km) -
 - relative close to coastline (14.99 km)
- Also the launch site KSC LC-39A is relative close to its closest city Titusville (16.32 km).
- Failed rocket with its high speed can cover distances like 15-20 km in few seconds. It could be potentially dangerous to populated areas.

Section 4

Build a Dashboard with Plotly Dash

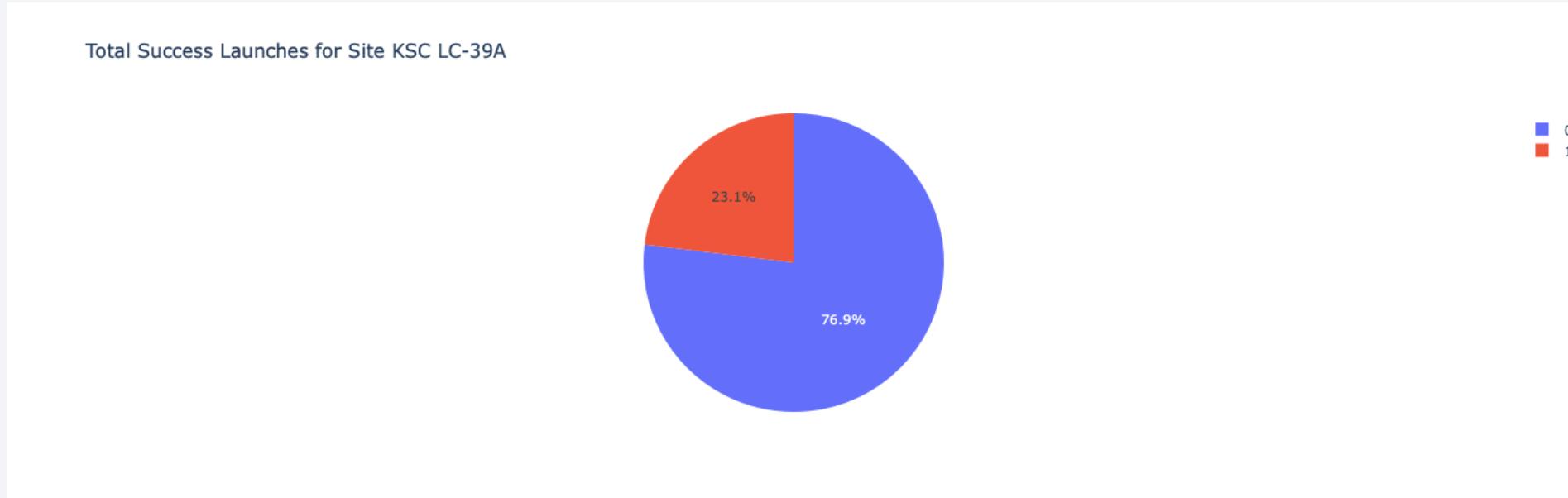


Lunch success count for all sites



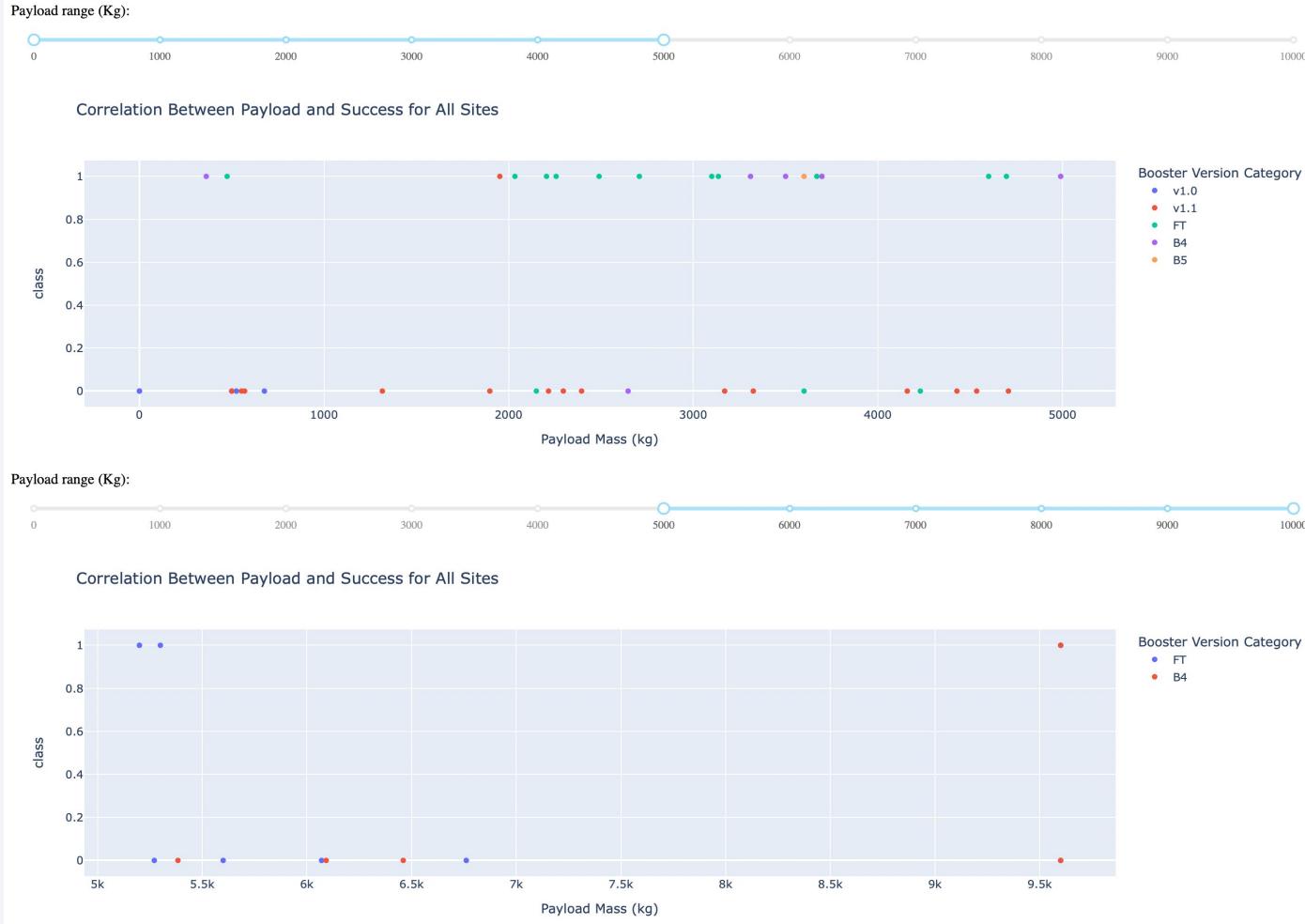
The chart clearly shows that from all the sites, KSC LC-39A has the most successful launches.

Launch site with highest launch success ratio



KSC LC-39A has the highest launch success rate (76.9%) with 10 successful and only 3 failed landings.

<Dashboard Screenshot 3>



The charts on the left shows that payloads between 2000 and 5500 kg have the highest success rate.

The background of the slide features a dynamic, abstract design. It consists of several thick, curved lines in shades of blue and yellow, creating a sense of motion and depth. The lines curve from the bottom left towards the top right, with some lines being more prominent than others. The overall effect is reminiscent of a tunnel or a high-speed journey through a digital space.

Section 5

Predictive Analysis (Classification)

Classification Accuracy

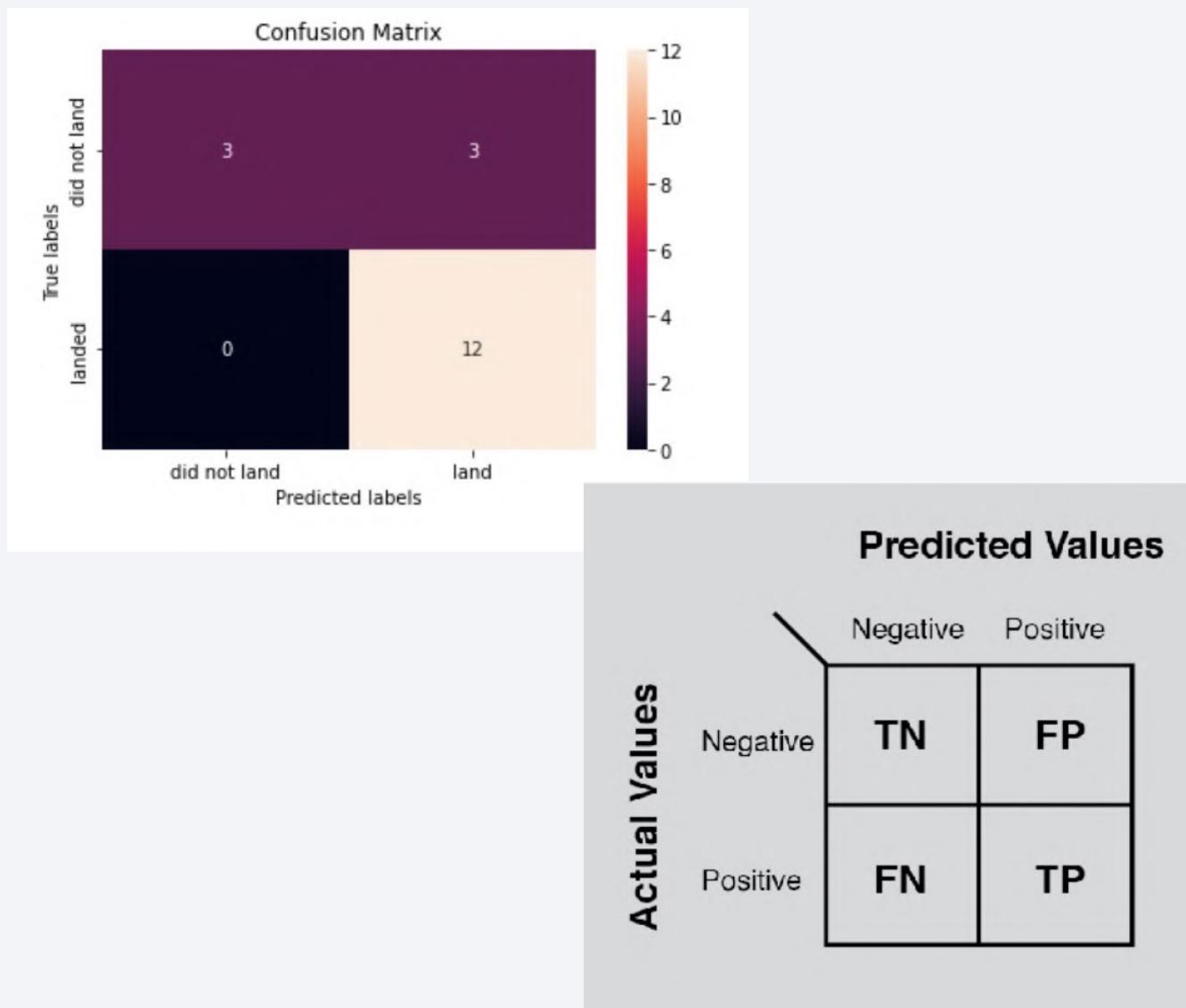
As we can see, by using the code as below: we could identify that the best algorithm to be the Tree Algorithm which have the highest classification accuracy.

Find the method performs best:

```
In [79]: algorithms = {'KNN':knn_cv.best_score_,'Tree':tree_cv.best_score_,'LogisticRegression':logreg_cv.best_score_}
bestalgorithm = max(algorithms, key=algorithms.get)
print('Best Algorithm is',bestalgorithm,'with a score of',algorithms[bestalgorithm])
if bestalgorithm == 'Tree':
    print('Best Params is :',tree_cv.best_params_)
if bestalgorithm == 'KNN':
    print('Best Params is :',knn_cv.best_params_)
if bestalgorithm == 'LogisticRegression':
    print('Best Params is :',logreg_cv.best_params_)

Best Algorithm is Tree with a score of 0.9
Best Params is : {'criterion': 'entropy', 'max_depth': 18, 'max_features': 'auto', 'min_samples_leaf': 4, 'min_samples_split': 10, 'splitter': 'random'}
```

Confusion Matrix



The confusion matrix for the decision tree classifier shows that the classifier can distinguish between the different classes. The major problem is the false positives .i.e., unsuccessful landing marked as successful landing by the classifier.

Conclusions

We can conclude that:

- The Tree Classifier Algorithm is the best Machine Learning approach for this dataset.
- The low weighted payloads (which define as 4000kg and below) performed better than the heavy weighted payloads.
- Starting from the year 2013, the success rate for SpaceX launches is increased, directly proportional time in years to 2020, which it will eventually perfect the launches in the future.
- KSC LC-39A have the most successful launches of any sites; 76.9%
- SSO orbit have the most success rate; 100% and more than 1 occurrence.

Thank you!

