IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Nthabiseng Maahlo
11 March 2024

Outline

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

Executive Summary

- **Summary of methodologies**

Data was collected using the SpaceX REST API and webscrapping the Wikipedia page “List of falcon 9 and falcon heavy launches”. The requested data was then cleaned and converted to pandas dataframe. Data wrangling was then performed to determine the label for training supervised models. Exploratory data analysis was performed using data visualization and SQL. Built an interactive map with Folium and a dashboard with plotly dash. Built a machine learning pipeline using classification to predict if the first stage of the falcon 9 will land successfully.

- **Summary of all results**

From the EDA, the following features were selected and used in success prediction; Flight Number, Payload Mass, Orbit, Launch Site, Flights, Grid Fins, Reused, Legs, Landing Pad, Block, Reused Count, Serial. The logistic regression, SVM and KNN models all have the highest accuracy of 0.833 leaving the Decision tree as the worst performing model.

Introduction

- **Project background and context**

This project's task is to predict if the Falcon 9 first stage will land successfully. This information can be used by another company which would want to bid against SpaceX for a rocket launch. This is because most providers rocket launches cost 165 million dollars or more each while SpaceX advertises Falcon 9 rocket launches on its website, with a cost of 62 million dollars. The SpaceX rocket launch is cheaper because they reuse the first stage so if we can predict if the first stage will land successfully, we can determine if SpaceX will attempt to land a rocket or not.

- **Problem statement**

Predict whether Space X will attempt to land a rocket or not

Section 1

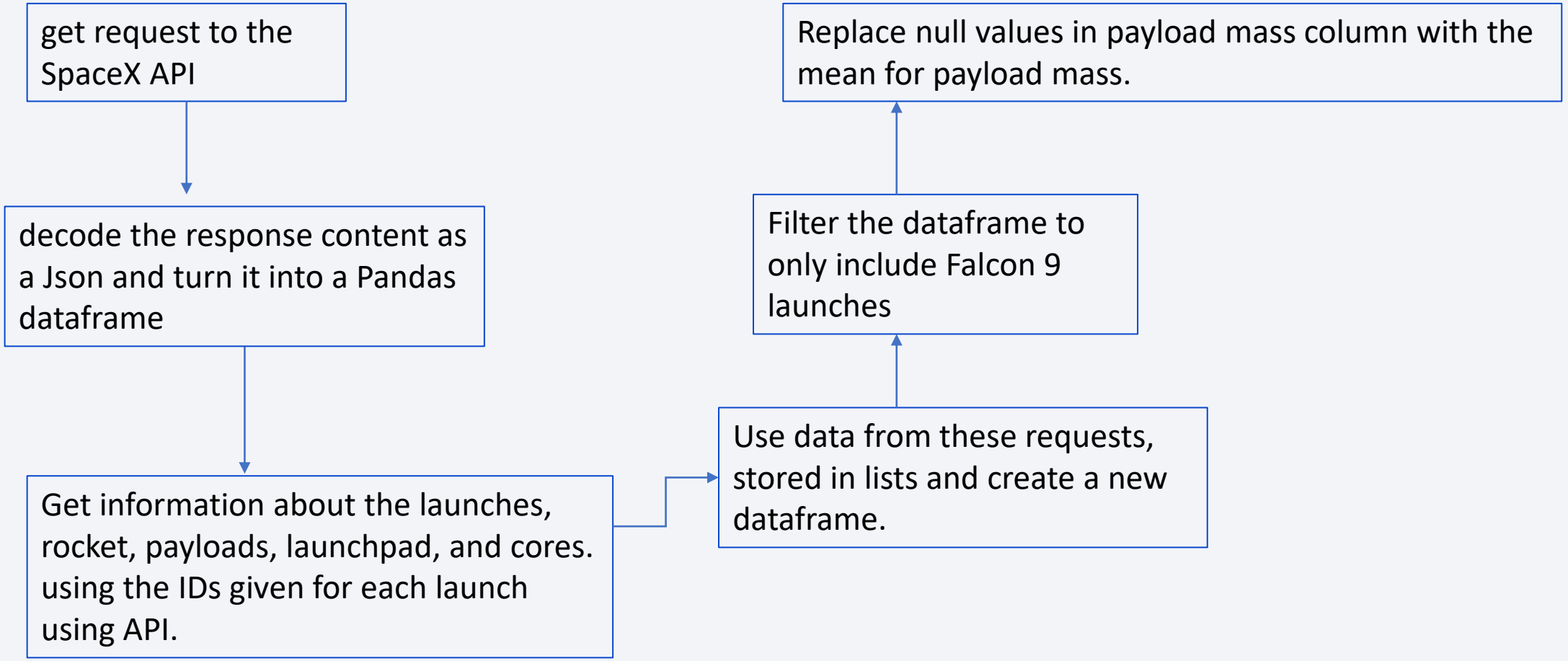
Methodology

Methodology

Executive Summary

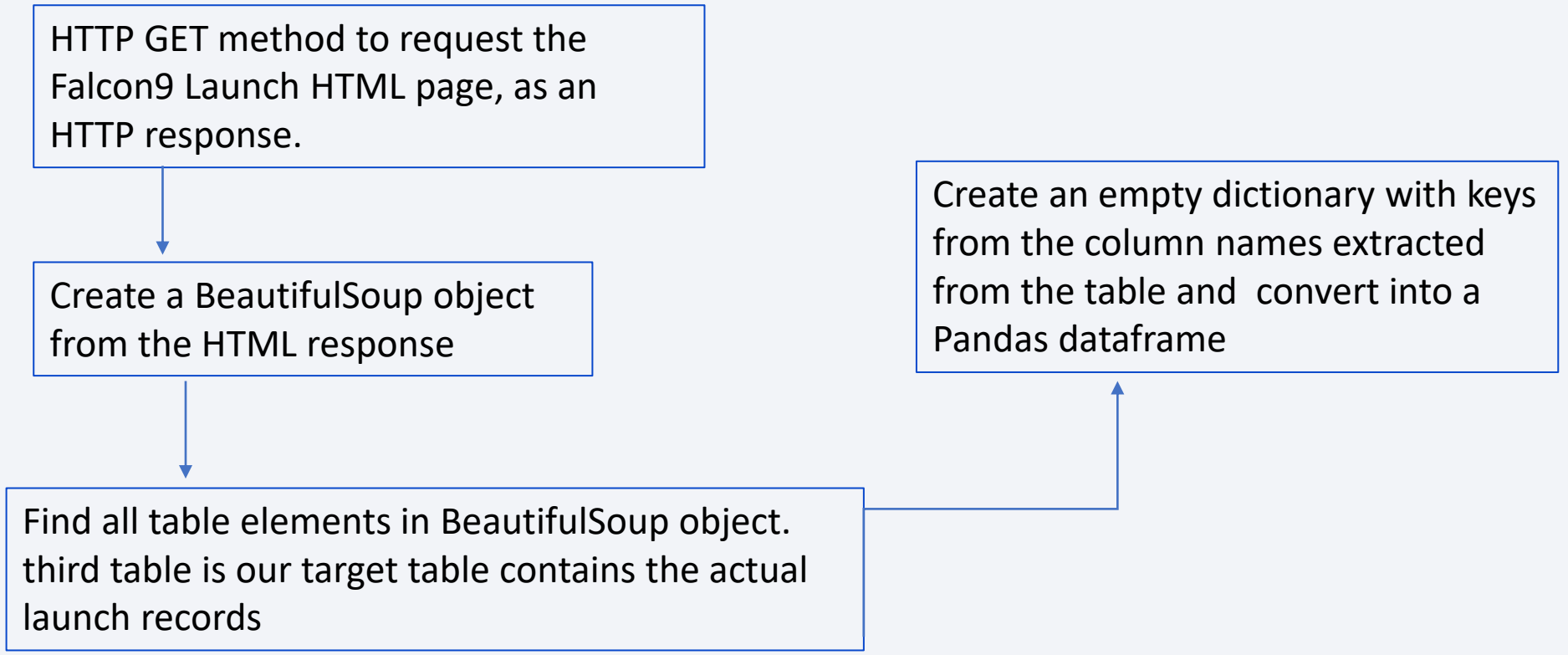
- Data collection methodology:
- Perform data wrangling
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models

Data Collection – SpaceX API



- GitHub URL of the completed SpaceX API calls notebook: https://github.com/Nthabiseng551/Data-Science-Coursera/blob/main/jupyter-labs-spacex-data-collection-api.ipynb

Data Collection - Scraping



- GitHub URL of the completed web scraping notebook: https://github.com/Nthabiseng551/Data-Science-Coursera/blob/main/jupyter-labs-webscraping.ipynb

Data Wrangling

- **How data were processed**

Performed some Exploratory Data Analysis (EDA) to find some patterns in the data and determined the label for training supervised models. Converted launch outcomes in Training Labels, 1 means booster landed successfully and 0 means it was unsuccessfully. Used the function get_dummies to apply OneHotEncoder to the column Orbits, LaunchSite, LandingPad, and Serial.

- GitHub URL of the completed data wrangling related notebook: https://github.com/Nthabiseng551/Data-Science-Coursera/blob/main/labs-jupyter-spacex-Data%20wrangling.ipynb

EDA with Data Visualization (Summary of charts)

- Exploratory data analysis and feature engineering using matplotlib and pandas.
- We can plot out the Flight Number vs. Payload Mass catplot and overlay the outcome of the launch.
- We can plot out the Flight Number vs. Launch site scatterplot and overlay the outcome of the launch
- We also want to observe if there is any relationship between launch sites and their payload mass. Scatterplot
- we want to visually check if there are any relationship between success rate and orbit type. Bar chart
- For each orbit, we want to see if there is any relationship between Flight Number and Orbit type. Scatterplot
- we can plot the Payload vs. Orbit scatter point charts to reveal the relationship between Payload and Orbit type
- You can plot a line chart with x axis to be Year and y axis to be average success rate, to get the average launch success trend.
- GitHub URL of your completed EDA with data visualization notebook: https://github.com/Nthabiseng551/Data-Science-Coursera/blob/main/jupyter-labs-eda-dataviz.ipynb.jupyterlite.ipynb

EDA with SQL (Summary of SQL queries)

- 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
- 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 landing outcomes (such as Failure (drone ship) or Success (ground pad)) between the date 2010-06-04 and 2017-03-20, in descending order
- **Add the GitHub URL of your completed EDA with SQL notebook:** https://github.com/Nthabiseng551/Data-Science-Coursera/blob/main/jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

- **Summary of map objects added to a folium map**

Folium.Circle was used to add a highlighted circle area with a text label (folium.marker) on a specific coordinate to mark all launch sites. Red and green markers for each launch record were included to mark the success/failed launches for each site on the map using marker clusters. MousePosition was added on the map to get coordinate for a mouse over a point on the map. As such, while exploring the map, the coordinates of any points of interests (such as railway) are easily found. Marked down a point on the closest coastline using MousePosition and calculate the distance between the coastline point and the launch site. Drew a PolyLine between a launch site to the selected coastline point.

- GitHub URL of the completed interactive map with Folium map: https://github.com/Nthabiseng551/Data-Science-Coursera/blob/main/lab_jupyter_launch_site_location.jupyterlite%20(1).ipynb

Build a Dashboard with Plotly Dash

- **Summary of plots/graphs and interactions added to the dashboard**

Added a Launch Site Drop-down Input Component, a callback function to render success-pie-chart based on selected site dropdown, a Range Slider to Select Payload, and a callback function to render the success-payload-scatter-chart scatter plot

- GitHub URL of the completed Plotly Dash lab: https://github.com/Nthabiseng551/Data-Science-Coursera/blob/main/spacex-dash.py

Predictive Analysis (Classification)

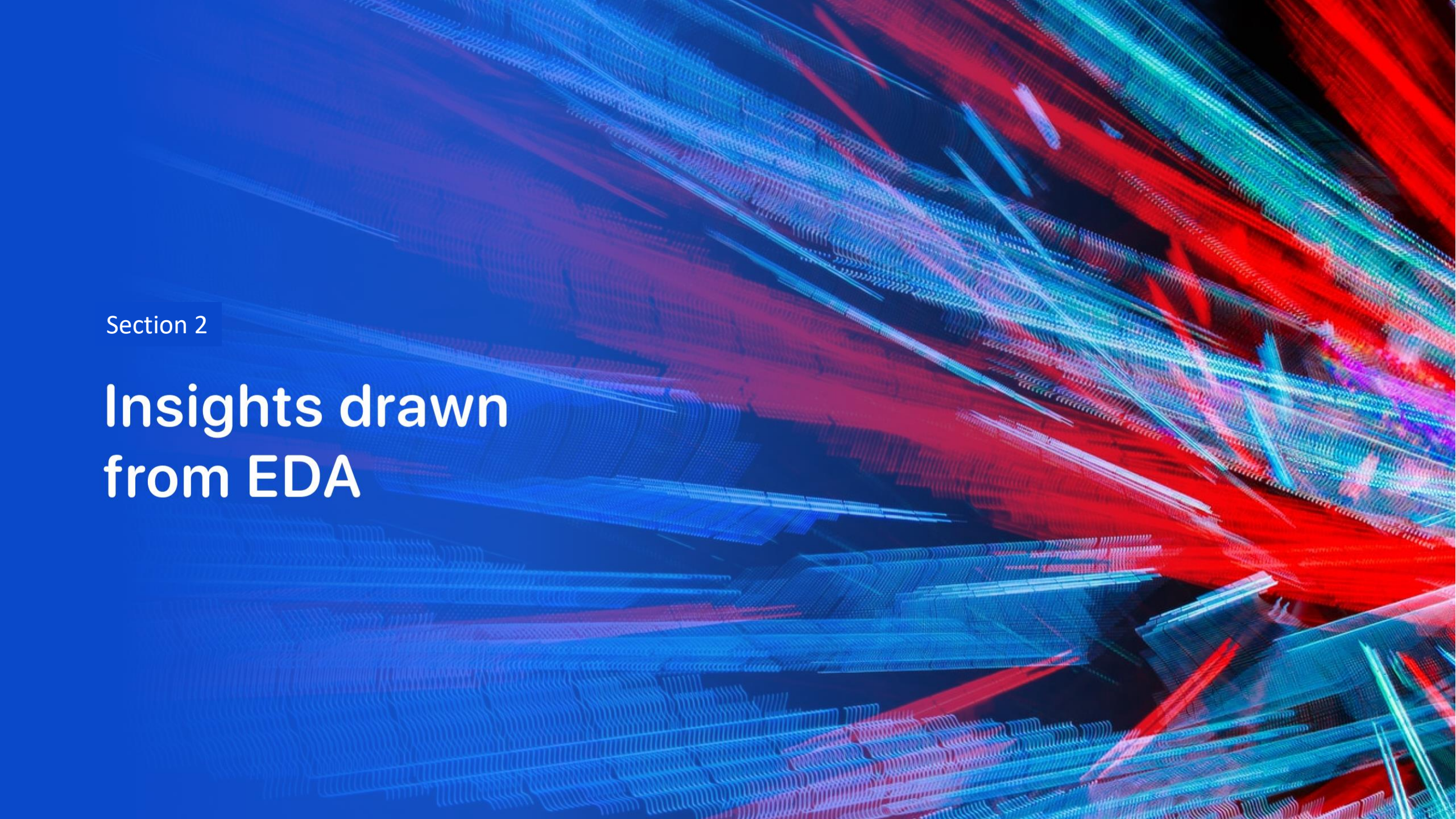
- **Summary of how classification models were built, tuned and evaluated.**

The building of a machine learning pipeline to predict if the first stage of the falcon 9 lands successfully included: Preprocessing, allowing us to standardize our data, and Train_test_split, allowing us to split our data into training and testing data, Models were trained and Grid Search was performed to find the hyperparameters that allow a given algorithm to perform best. Using the best hyperparameter values, the model with the best accuracy was determined using training data. The Logistic Regression, Support Vector machines, Decision Tree Classifier, and K-nearest neighbors were tested and a confusion matrix outputted for each.

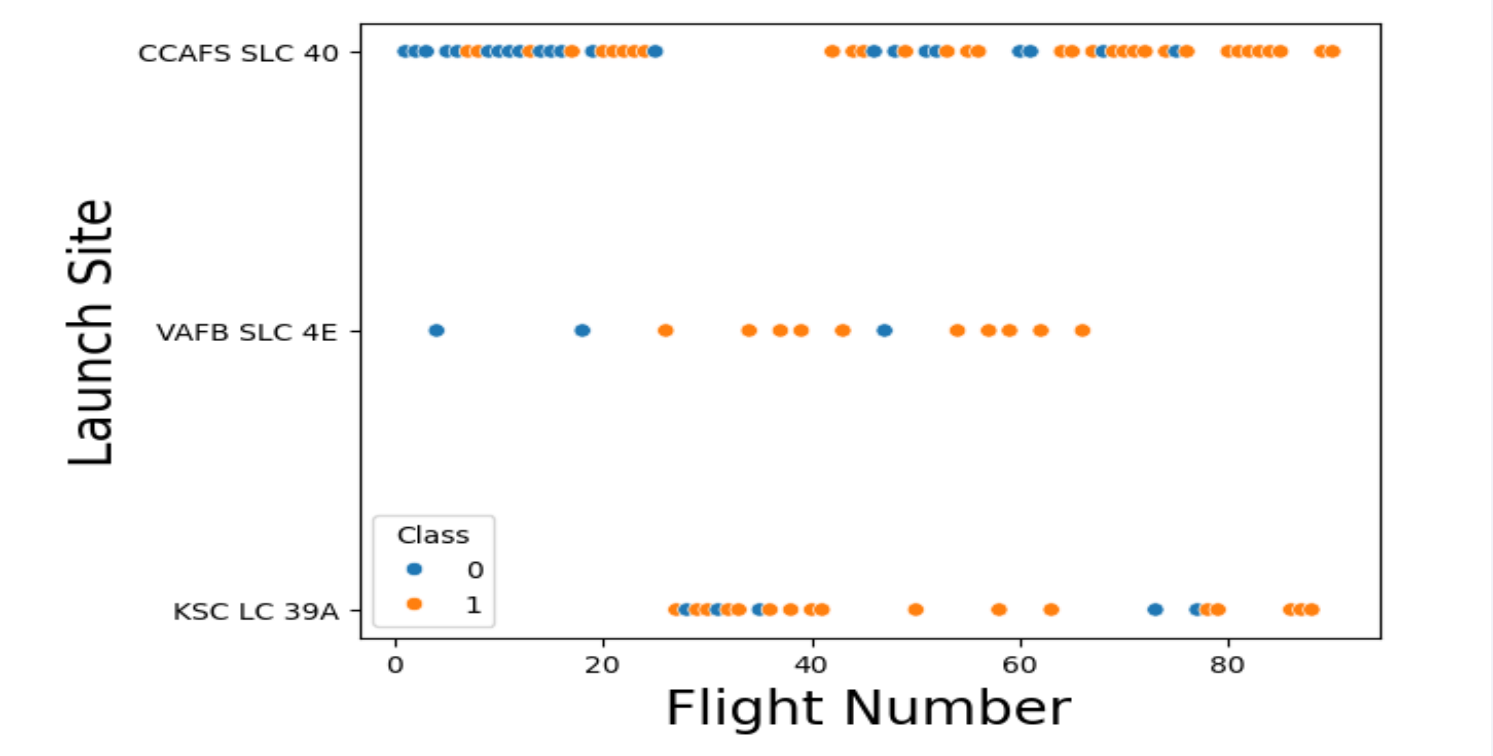
- GitHub URL of the completed predictive analysis lab: https://github.com/Nthabiseng551/Data-Science-Coursera/blob/main/SpaceX_Machine_Learning_Prediction_Part_5.jupyterlite%20(2).ipynb

Results

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

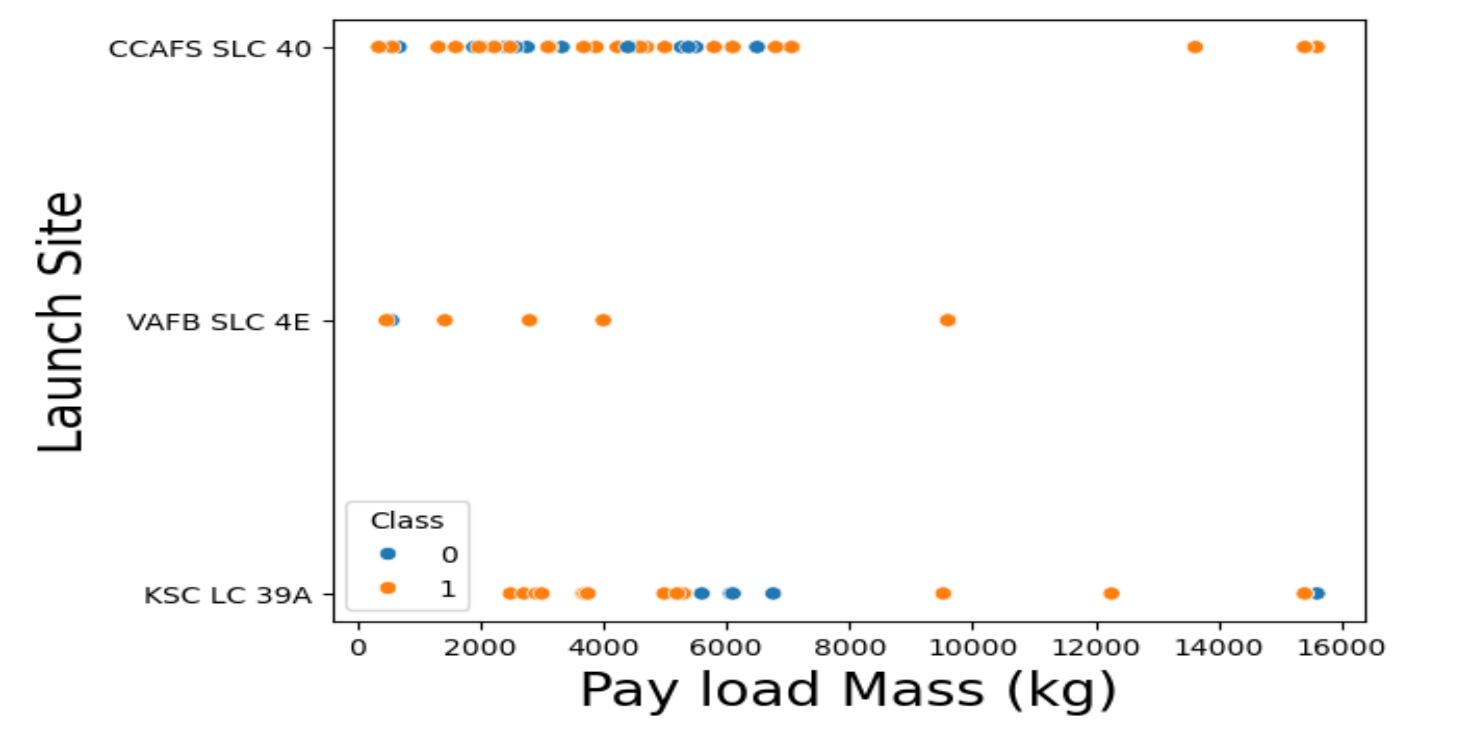

Section 2
Insights drawn from EDA

Flight Number vs. Launch Site



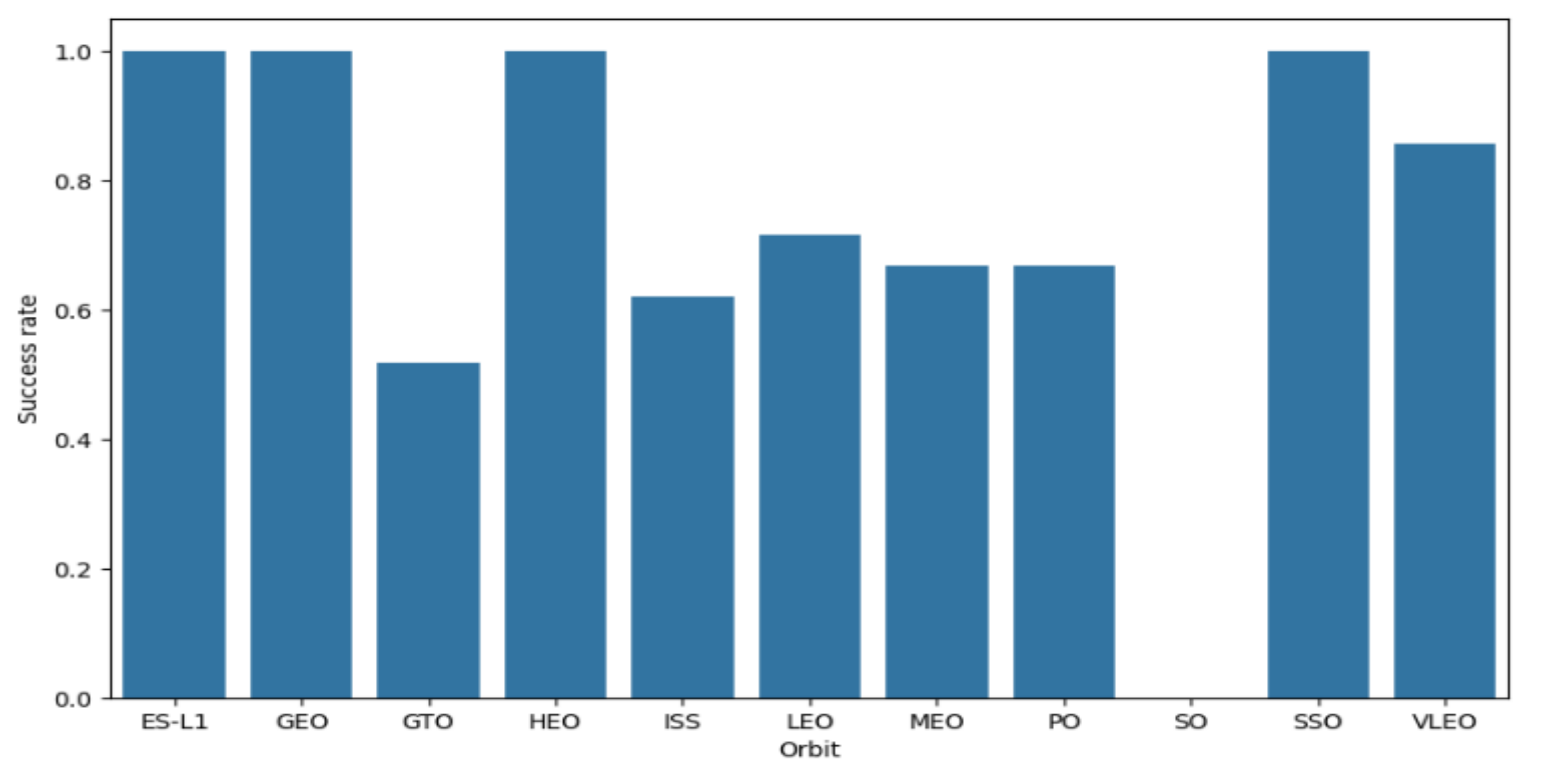
We see that as the flight number increases, the first stage is more likely to land successfully.

Payload vs. Launch Site



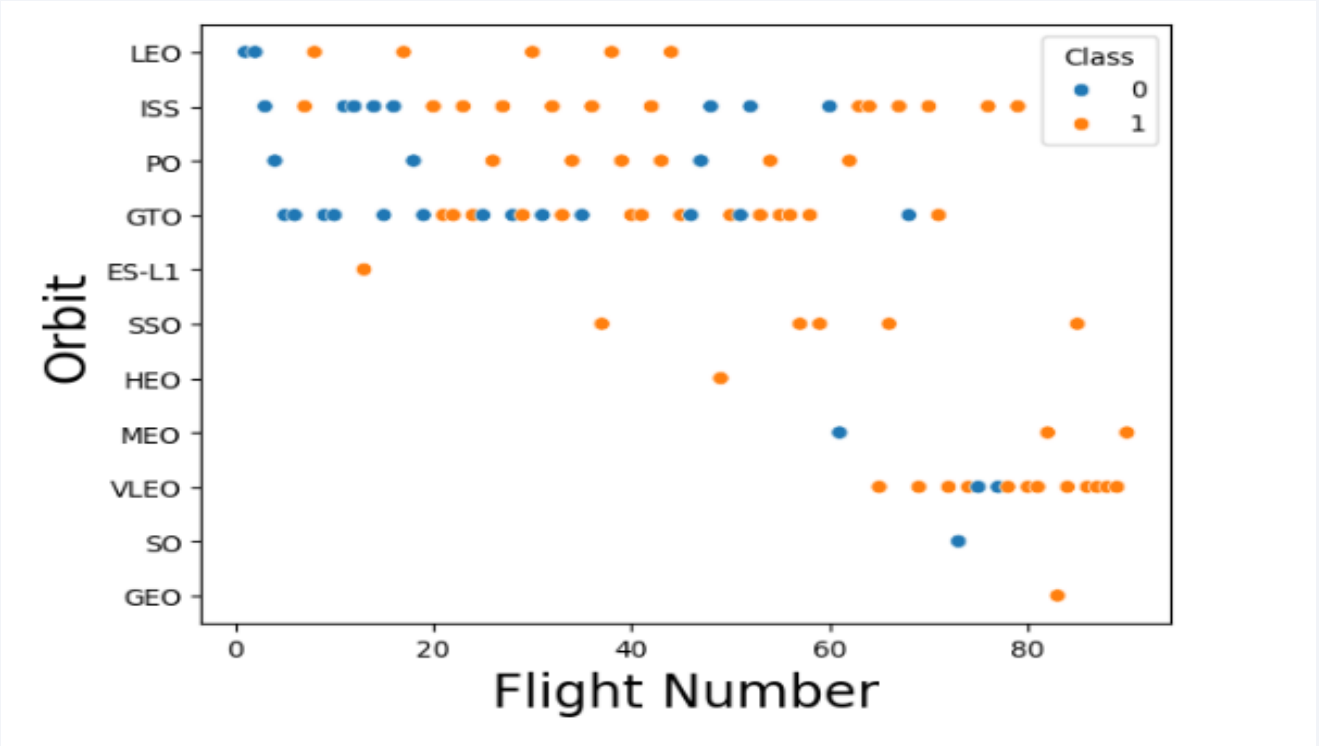
For the VAFB-SLC launch site, there are no rockets launched for heavy payload mass(greater than 10000 kg).

Success Rate vs. Orbit Type



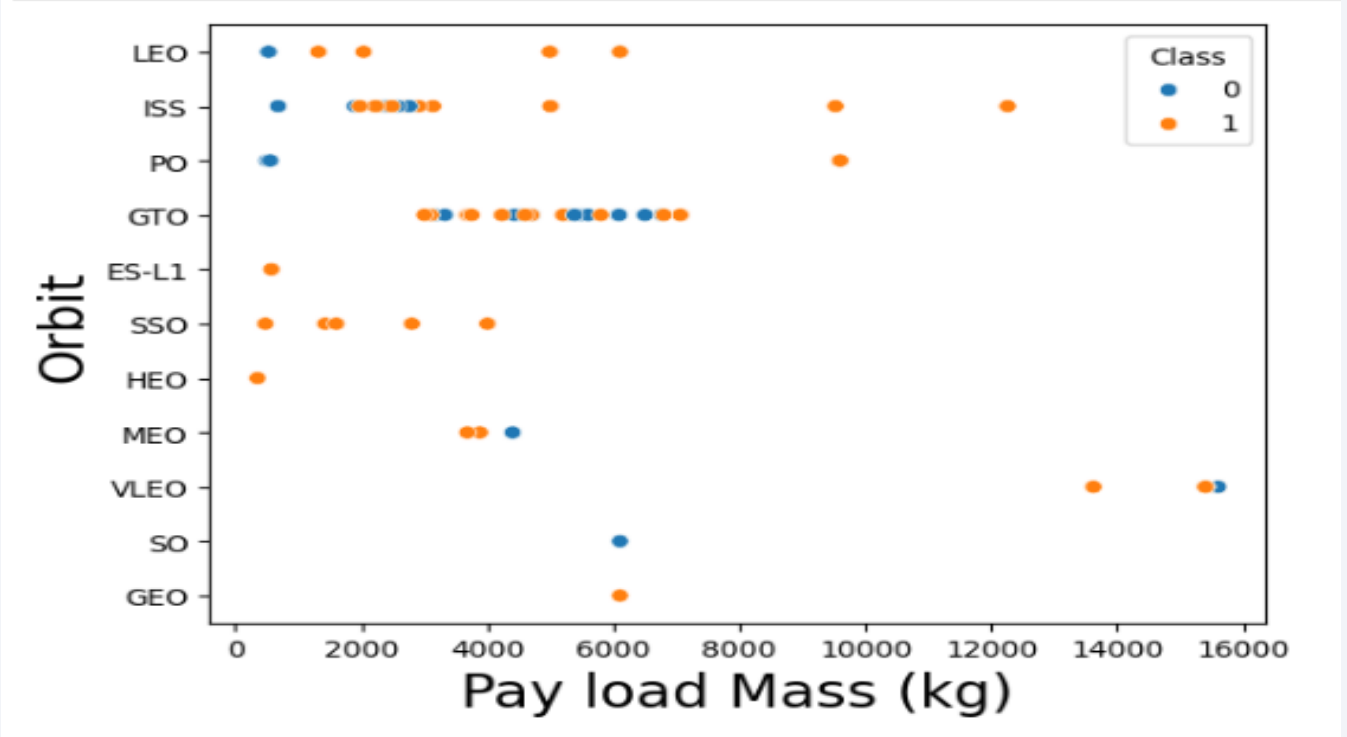
The orbits ES-L1, GEO, HEO and SSO have the highest rate of successful launches.

Flight Number vs. Orbit Type



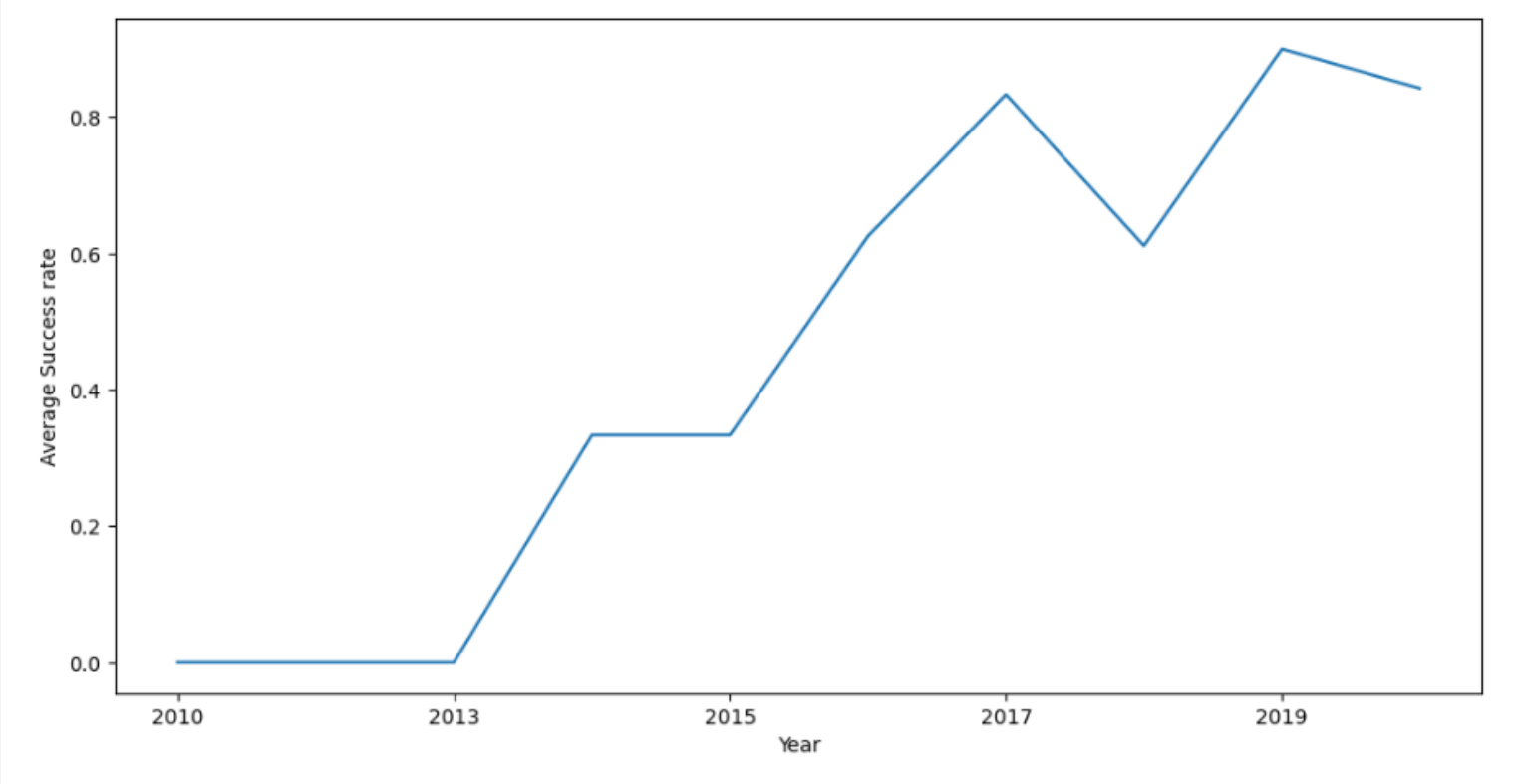
In the LEO orbit the Success appears related to the number of flights; on the other hand, there seems to be no relationship between flight number when in GTO orbit.

Payload vs. Orbit Type



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

Launch Success Yearly Trend



The success rate has since been increasing from 2013 to 2020.

All Launch Site Names

```
[10]: %sql select distinct "Launch_Site" from SPACEXTABLE
```

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

[10]:

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

Launch Site Names Begin with 'CCA'

```
[12]: %sql select * from SPACEXTABLE where Launch_Site LIKE 'CCA%' limit 5
```

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

[12]:

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
2010-12-08	15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight C1, two CubeSats, barrel of Brouere cheese	0	LEO (ISS)	NASA (COTS) NRO	Success	Failure (parachute)
2012-05-22	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
2012-10-08	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
2013-03-01	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

Total Payload Mass carried by boosters from NASA

```
[15]: %sql select sum(PAYLOAD_MASS__KG_) from SPACEXTABLE where "Customer"='NASA (CRS)'
```

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

[15]:

sum(PAYLOAD_MASS__KG_)
45596

Average Payload Mass by F9 v1.1

```
[17]: %sql select avg(PAYLOAD_MASS__KG_) from SPACEXTABLE where "Booster_Version" LIKE 'F9 v1.1%'
```

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

[17]:

avg(PAYLOAD_MASS_KG_)
2534.6666666666665

First Successful Ground Landing Date

```
[20]: %sql select min("Date") from SPACEXTABLE where "Landing_Outcome"='Success (ground pad)'

 * sqlite:///my_data1.db
Done.

[20]: min("Date")
      2015-12-22
```

Boosters with Successful Drone Ship Landing with Payload between 4000 and 6000

```
[21]: %lect Booster_Version from SPACEXTABLE where "Landing_Outcome"='Success (drone ship)' and PAYLOAD_MASS__KG_>4000 and PAYLOAD_MASS__KG_<6
```

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

[21]:

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

Total Number of Successful and Failure Mission Outcomes

```
[23]: %sql select "Mission_Outcome", count(*) from SPACEXTABLE group by "Mission_Outcome"
```

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

[23]:

Mission_Outcome	count(*)
Failure (in flight)	1
Success	98
Success	1
Success (payload status unclear)	1

Boosters which Carried Maximum Payload

```
[25]: %sql select Booster_Version from SPACEXTABLE where PAYLOAD_MASS__KG_=(select max(PAYLOAD_MASS__KG_) from SPACEXTABLE)
 * sqlite:///my_data1.db
Done.
```

[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

2015 Launch Records with Failed landing outcome in drone ship

```
[26]: g_Outcome", "Booster_Version", "Launch_Site" from SPACEXTABLE where "Landing_Outcome"='Failure (drone ship)' and substr(Date,0,5)='2015'
```

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

[26]:

substr(Date, 6,2)	Landing_Outcome	Booster_Version	Launch_Site
01	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
04	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

The first column is the month of the year

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

```
[29]: %sql select "Date", "Landing_Outcome", count(*) as count from SPACEXTABLE group by "Landing_Outcome" order by count desc
```

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

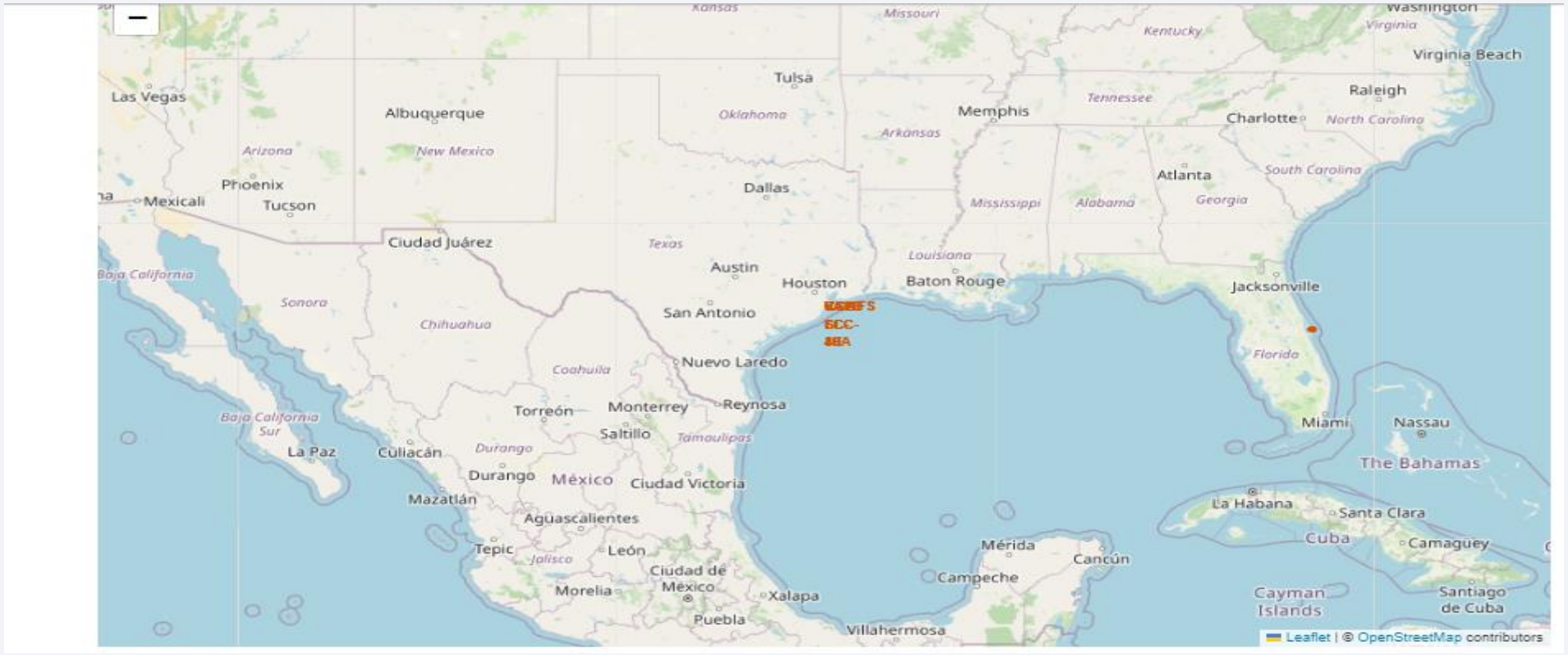
[29]:

Date	Landing_Outcome	count
2018-07-22	Success	38
2012-05-22	No attempt	21
2016-04-08	Success (drone ship)	14
2015-12-22	Success (ground pad)	9
2015-01-10	Failure (drone ship)	5
2014-04-18	Controlled (ocean)	5
2018-12-05	Failure	3
2013-09-29	Uncontrolled (ocean)	2
2010-06-04	Failure (parachute)	2
2015-06-28	Precluded (drone ship)	1
2019-08-06	No attempt	1

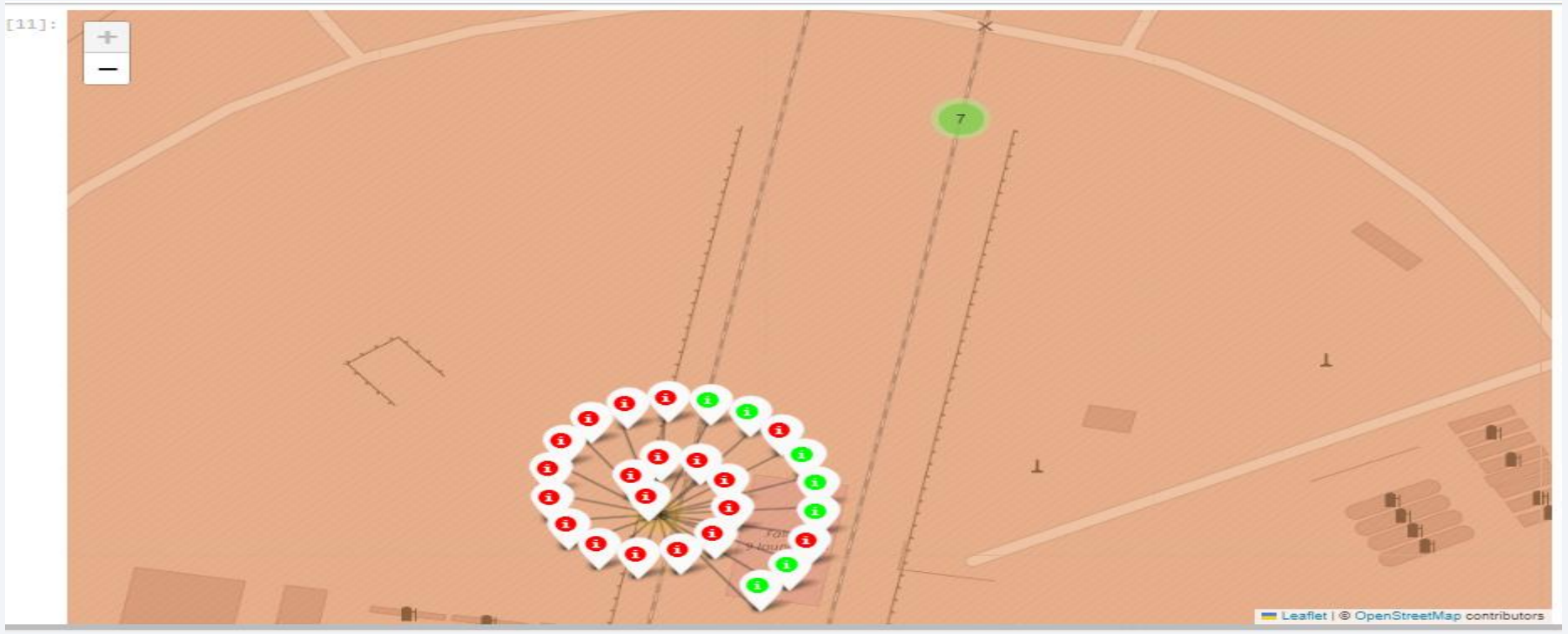
Section 3

Launch Sites Proximities Analysis

Generate map with marked launch sites

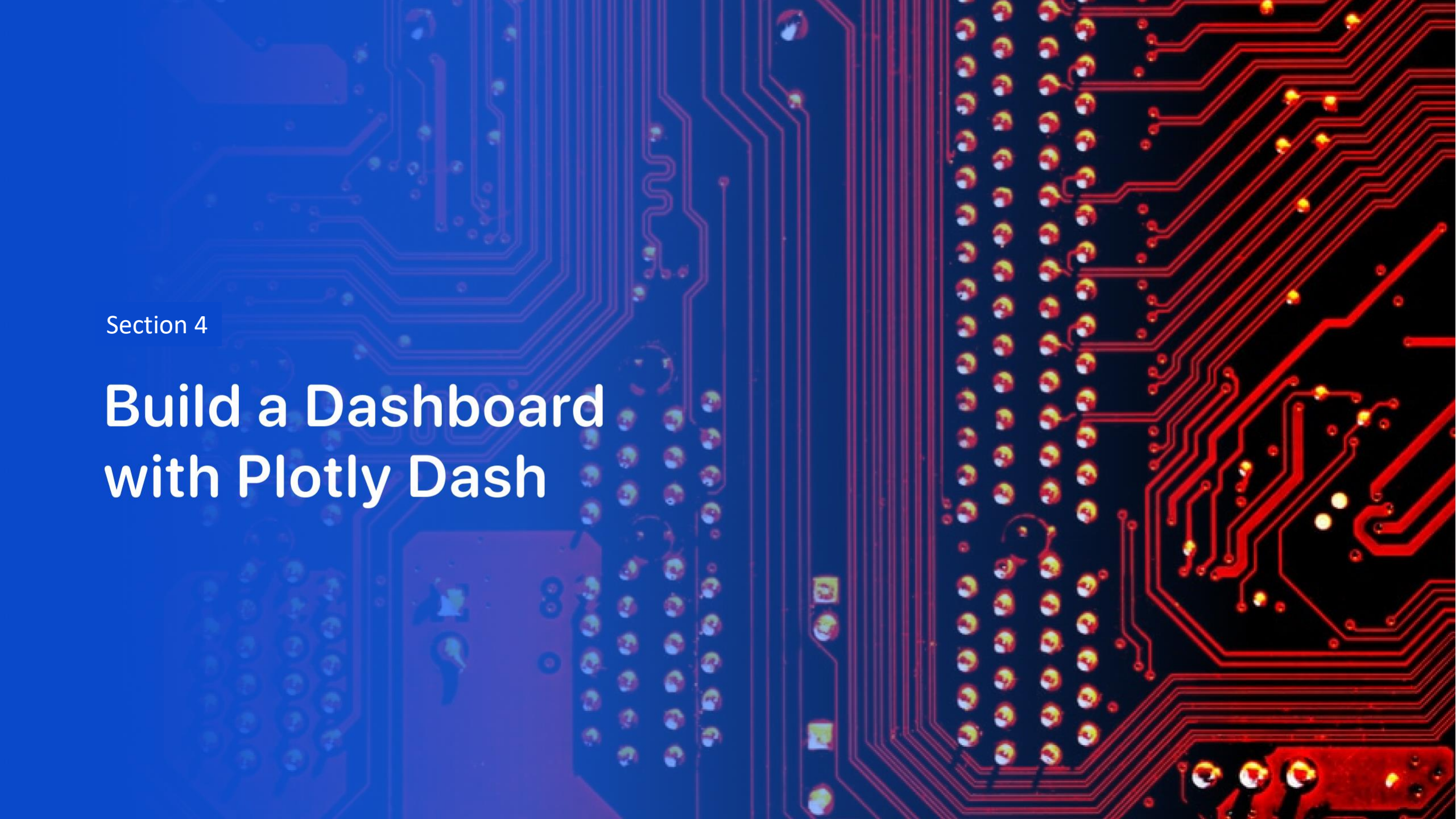


Map with success/failed launches markers for each site

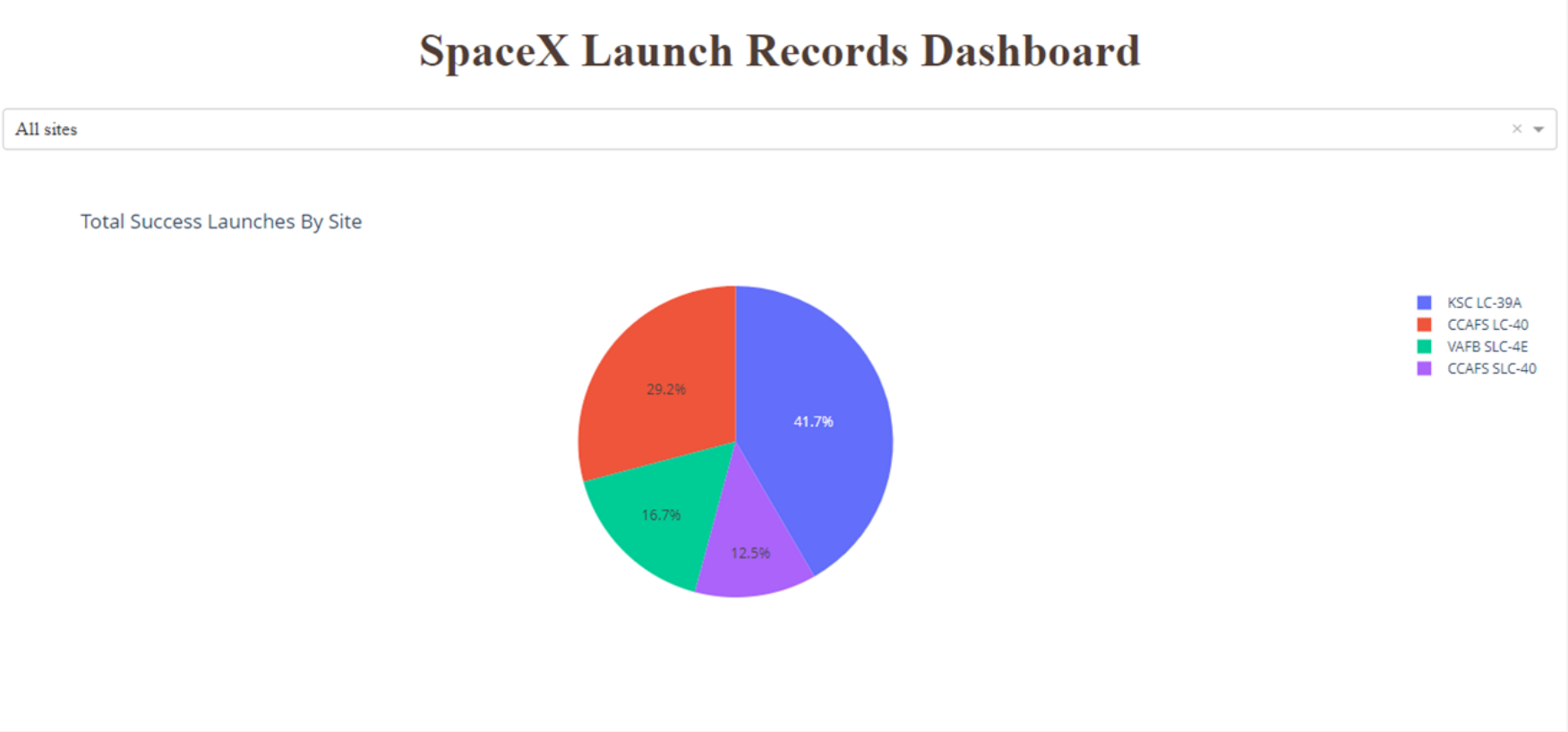


Red markers are for failed launch outcome and green is for successful launches. For this site, there are more failed outcomes than successful ones.

Map with distance line between launch site and proximity

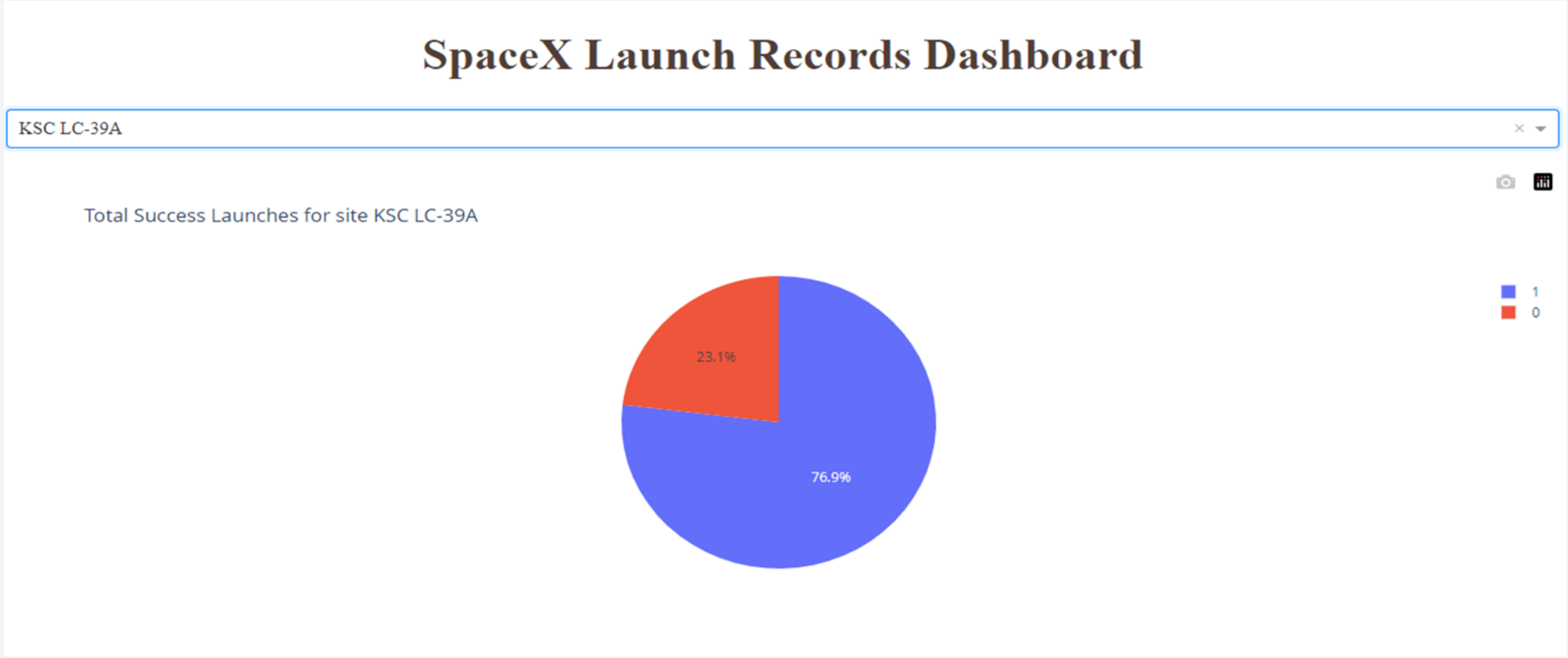

Section 4
Build a Dashboard with Plotly Dash

Launch success count for all sites



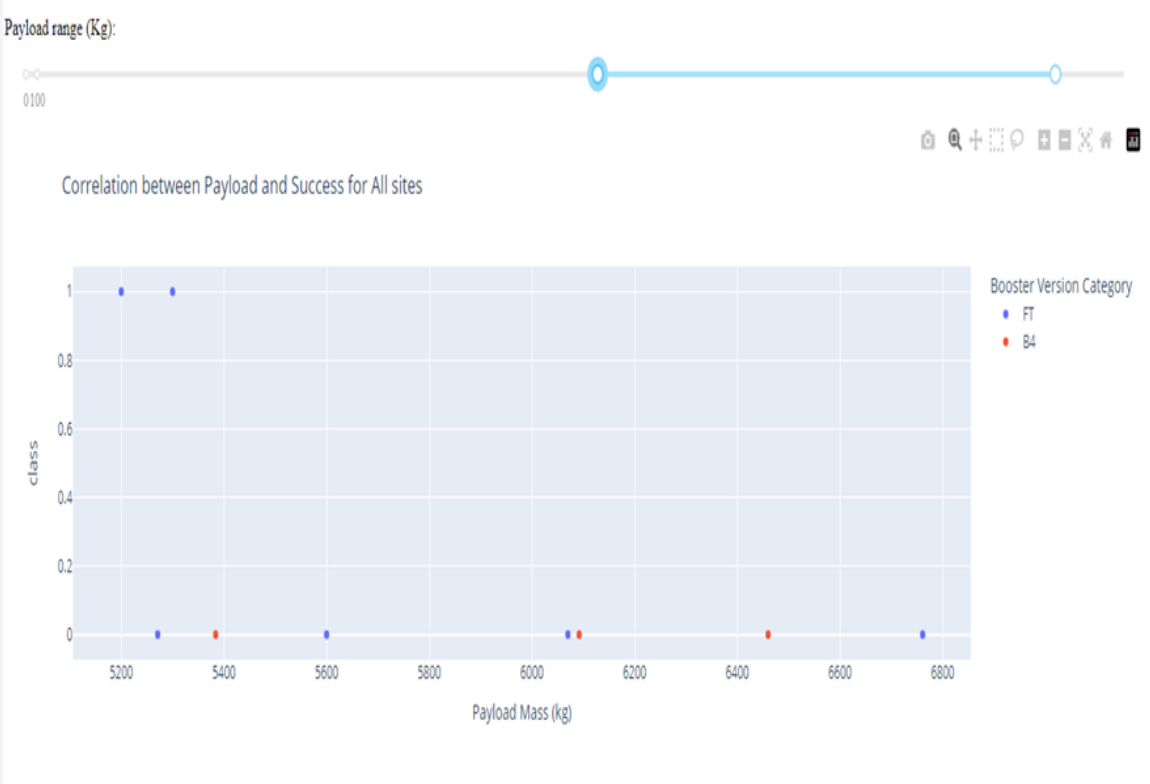
KSC LC-39A site has the highest success launches.

Launch success count for site with highest success ratio



76.9% of launches were successful for the KSC LC-39A site.

Payload vs Launch outcome for all sites and boosters

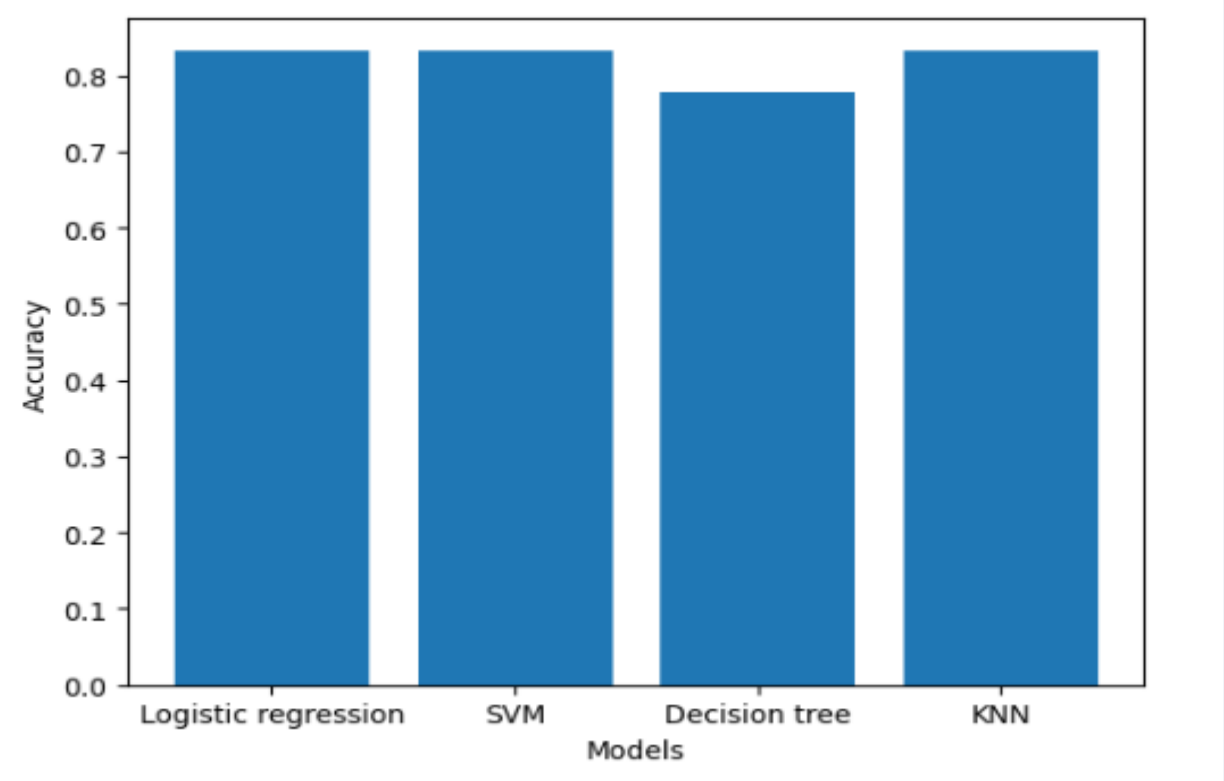


From the first plot, the booster version FT has a high success rate. For the payload mass range, 5200 – 6800 kg ,second plot, there success rate is low.

Section 5

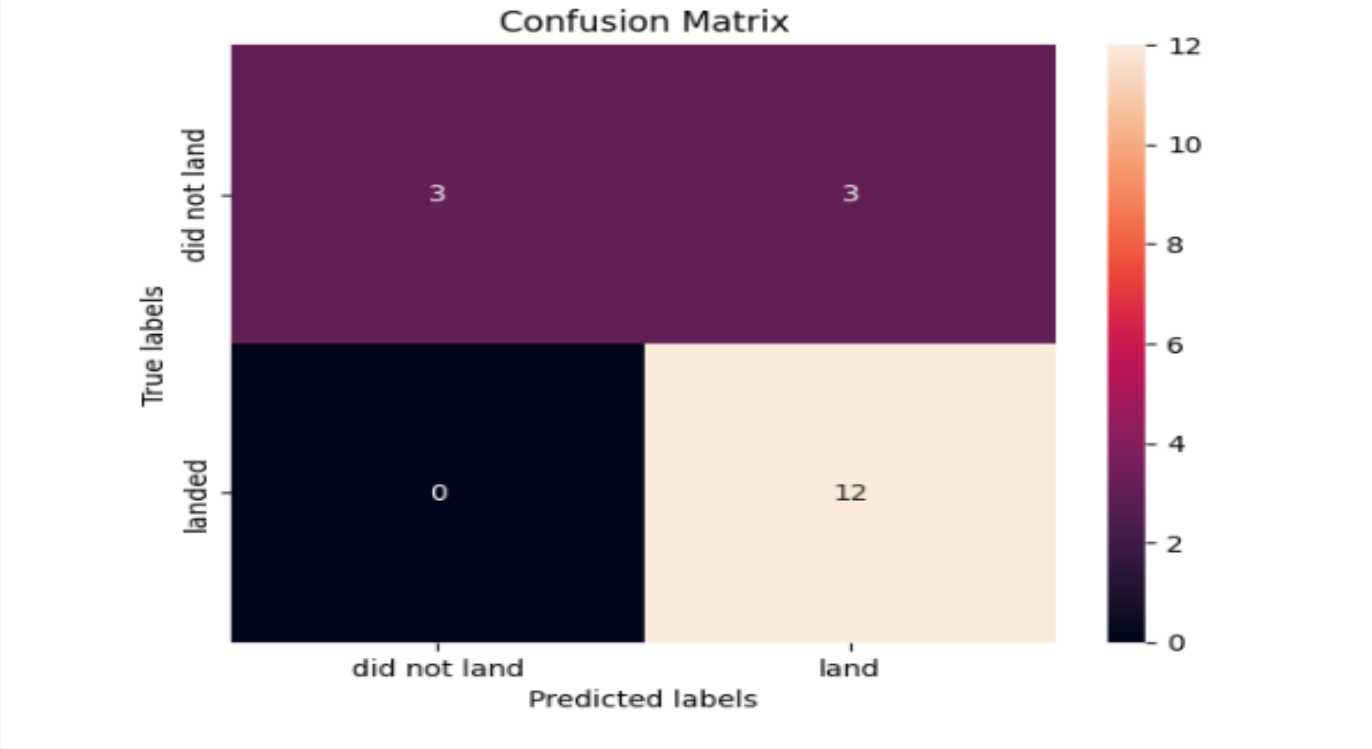
Predictive Analysis (Classification)

Classification Accuracy



The logistic regression, SVM and KNN models all have the highest accuracy of 0.833

Confusion Matrix



The confusion matrix for the best performing models logistic regression, SVM and KNN, which is similar for all three as is their accuracies. Examining the confusion matrix, we see that the models can distinguish between the different classes. We see that the major problem is false positives.

Conclusions

- From the EDA and feature engineering, the following features were selected and used in success prediction; Flight Number, Payload Mass, Orbit, Launch Site, Flights, Grid Fins, Reused, Legs, Landing Pad, Block, Reused Count, Serial.
- The logistic regression, SVM and KNN models all have the highest accuracy of 0.833 leaving the Decision tree as the worst performing model. The confusion matrices of the three best performing models are also similar.

Thank you!