IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

NAME: SABAU FLORIN
DATE: 10.01.2023

Outline

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

Executive Summary

In this project, we attempted to predict if the Falcon 9 first stage will land successfully using the SpaceX data set extracted from the SpaceX REST API.

We performed Exploratory Data Analysis using SQL and visualizations from Folium, Seaborn, Plotly Dash and used grid search function to train several models and evaluate their performance. We compared the performance of Logistic Regression, SVM, Decision Tree and KNN models on the test data.

The results show that all models (Logistic Regression, SVM, Decision Tree, and KNN) have a similar accuracy on the test data of approximately 0.89. However, KNN has the best performance with a recall of 0.93 and ROC-AUC of 0.96.

It's worth noting that the dataset we used was relatively small, with only 93 observations and 83 features. This may limit the generalizability of our findings and increase the risk of overfitting the model. To mitigate the effects of this small dataset, we used Cross-Validation technique. However, more data would always be beneficial in increasing robustness and reliability of the results.

Introduction

The project's goal is to predict the success of SpaceX missions using their historical data. The space industry is an important field, not only for scientific research but also for commercial purposes. However, as many space missions are high-cost and high-risk, it is important for companies to have the ability to predict which missions are most likely to be successful. The data used in this project is provided by SpaceX through their public REST API, which contains information on past launches, rockets and more. By using machine learning techniques we hope to create a model that will help the company to predict the success of their future missions and help them to optimize their resources.

Section 1

Methodology

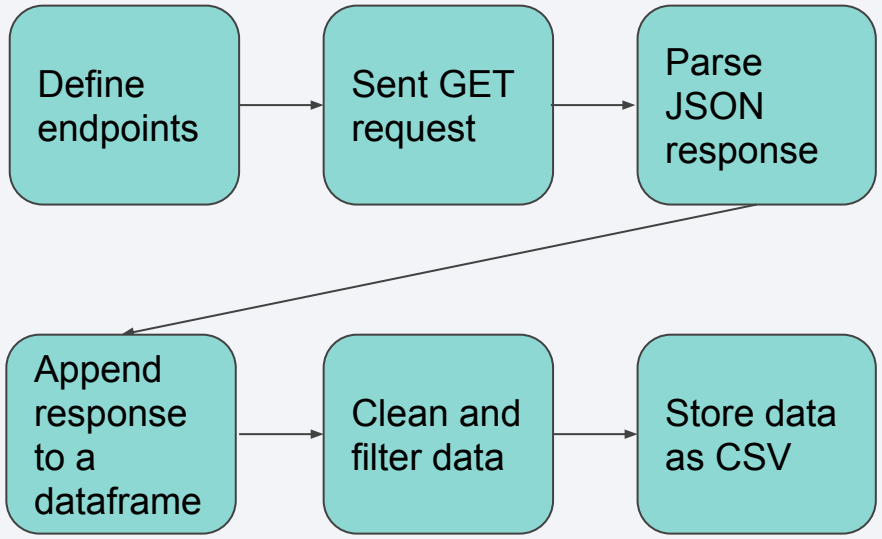
Methodology

Executive Summary

The data for this analysis was collected through both requesting data from the SpaceX public API and web scraping the SpaceX Wikipedia page. The data was then cleaned and filtered to only include Falcon 9 launches, and several exploratory data analysis techniques were applied using visualization and SQL. Interactive visual analytics were also performed using Folium and Plotly Dash. Finally, predictive analysis was conducted using four classification models, which were built and optimized using cross-validation and parameter selection.

Data Collection – SpaceX API

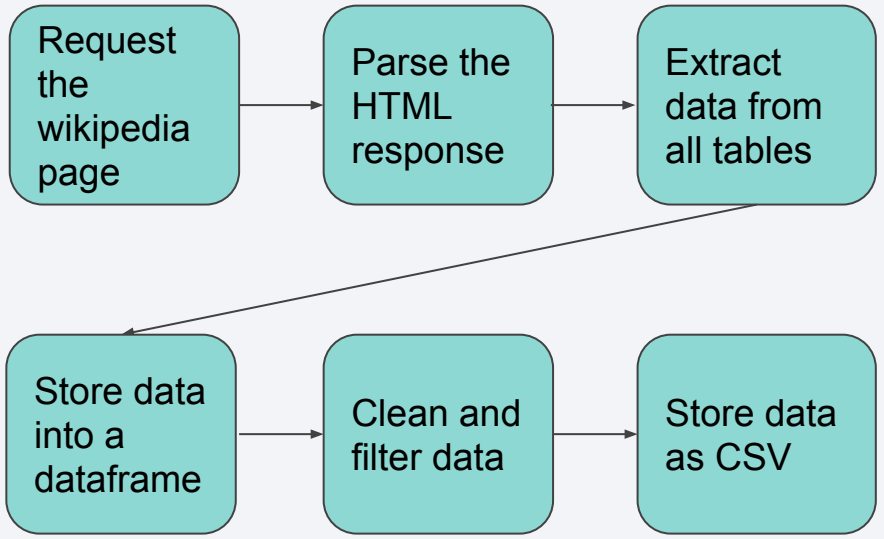
- Data was extracted using five endpoints from the SpaceX REST API: past launches, rockets, launchpads, payloads, cores.
- This is a free API so no authentication was required.
- The JSON response was sent to a pandas dataframe where data was cleaned and filtered and the final table was saved as a CSV file.



- GitHub link: https://github.com/sfs-projects/ml/tree/develop/ibm-data-science-project

Data Collection - Scraping

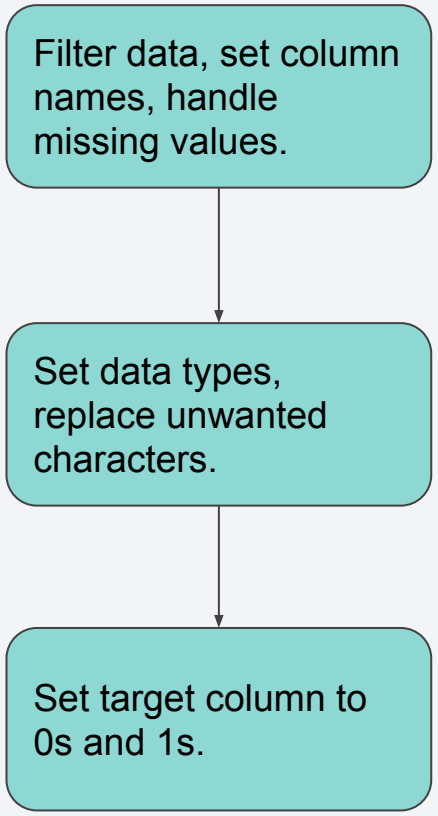
- Web scraping from the Wikipedia page was done with a number functions which access the html tables.
- The response was parsed and data was extracted into a pandas dataframe where cleaning and transforming was done.



- GitHub link: https://github.com/sfs-projects/ml/tree/develop/ibm-data-science-project

Data Wrangling

- Column names were set, data was filtered and missing values were handled.
- Data types were set for all columns and all columns were cleaned so that they keep the same format.
- Target column was transformed to 0s and 1s to be ready for machine learning models.



- GitHub link: https://github.com/sfs-projects/ml/tree/develop/ibm-data-science-project

EDA with Data Visualization

- Scatter plots: Scatter plots were employed to investigate the relationships between different variables such as flight number and launch site, payload mass and launch site, flight number and orbit type, and payload mass and orbit type. These plots can be seen by using seaborn library's *catplot()* function.
- Bar chart: Bar charts were used to compare the success rate for different orbit types, it was achieved by groupby 'Orbit' and calculating the mean success rate for each orbit type and then creating a bar chart by using seaborn library's *barplot()* function.
- Line chart: Line charts were used to show the trend of success rate over time, it was achieved by extracting the year from the Date column and then grouping the data by year and calculating the success rate for each year, it can be seen by using pandas *groupby()* function.

- GitHub link: https://github.com/sfs-projects/ml/tree/develop/ibm-data-science-project

EDA with SQL

A number of SQL queries were performed in this section order to retrieve:

- launch sites and number of launches.
- 5 records where launch site starts with "CCA".
- total payload mass carried by NASA (CRS)
- average payload mass carried by booster version F9 v1.1
- date of first successful landing outcome on ground pad.
- booster versions with successful drone ship landings and payload mass between 4000 and 6000.
- number of successful and failed mission outcomes.
- booster versions with maximum payload mass.
- records of month, failure landing outcomes, booster version, and launch site for year 2015.
- count of successful landings in descending order for a specific date range (between 2010-06-04 and 2017-03-20)

- GitHub link: https://github.com/sfs-projects/ml/tree/develop/ibm-data-science-project

Build an Interactive Map with Folium

- The main task performed in this section was creating and adding visual objects to a Folium map in order to analyze geographical patterns related to launch sites. Specifically, marker objects were used to display all launch sites on the map, including information about successful and failed launches at each site. Line objects were also used to calculate the distances between a launch site and its proximities.
- Through this analysis, it was found that launch sites are typically located in close proximity to railways, highways, and coastlines, and tend to keep a certain distance away from cities.

- GitHub link: https://github.com/sfs-projects/ml/tree/develop/ibm-data-science-project/anexes

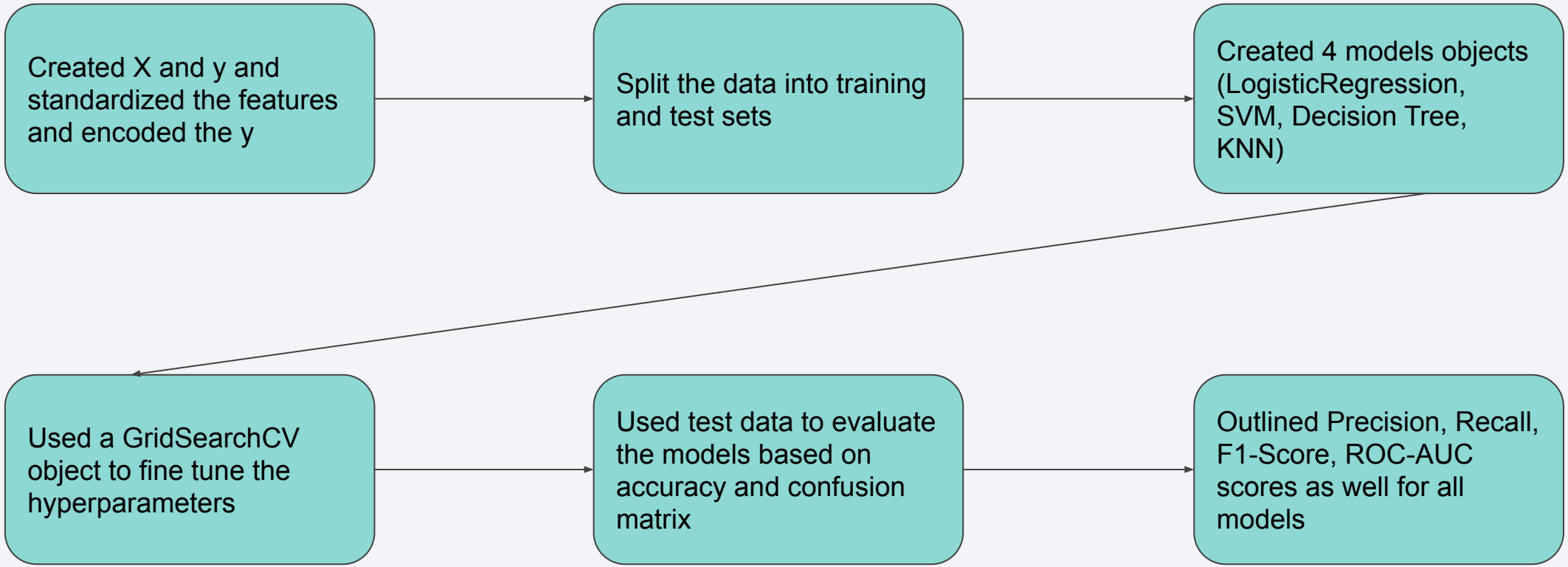
Build a Dashboard with Plotly Dash

In this section, a dashboard was created using the Dash library. The dashboard has the following features:

- A dropdown list that allows users to select a launch site, with the default selection being "All Sites".
- A pie chart that displays the total successful launches count for all sites, or shows the Success vs. Failed counts for a specific launch site if one is selected from the dropdown.
- A slider that allows users to select a range of payload weight in kilograms.
- A scatter chart that displays the correlation between payload mass and launch success for launches within the selected payload range.
- The purpose of the dashboard is to easily explore and understand patterns in SpaceX launch data, by providing visualizations of launch success rate, payload mass, and launch site.

- GitHub link: https://github.com/sfs-projects/ml/tree/develop/ibm-data-science-project/anexes

Predictive Analysis (Classification)

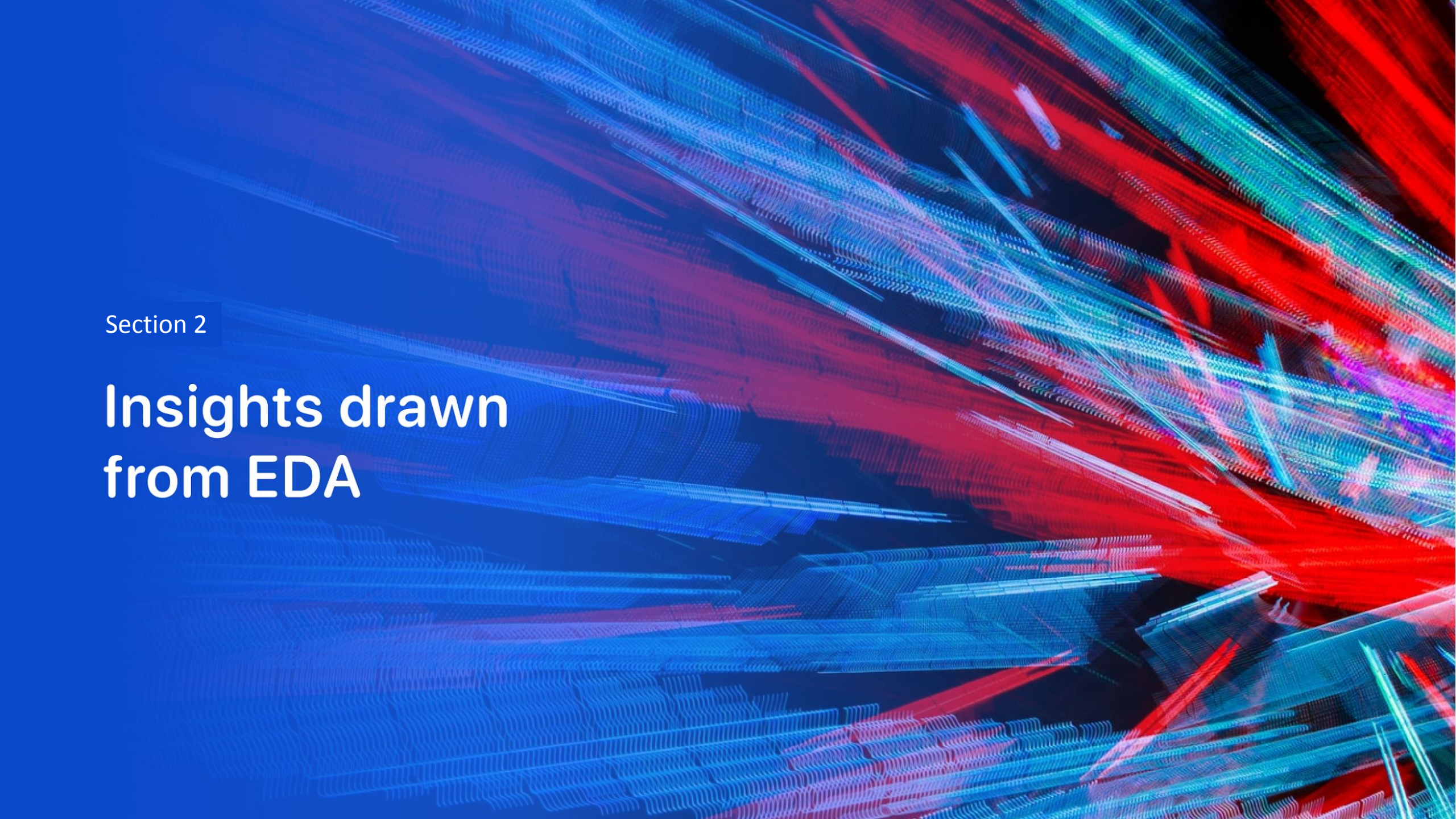


Results

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

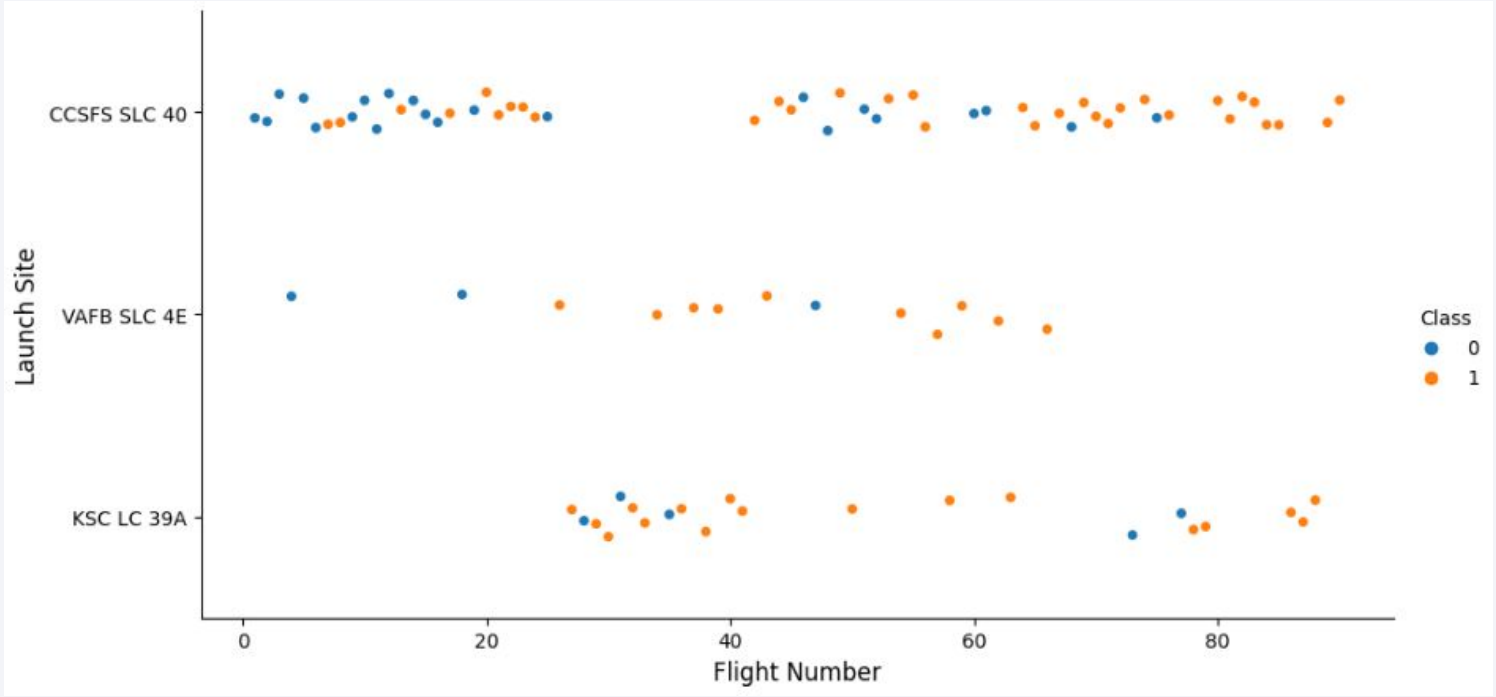
Section 2

Insights drawn from EDA



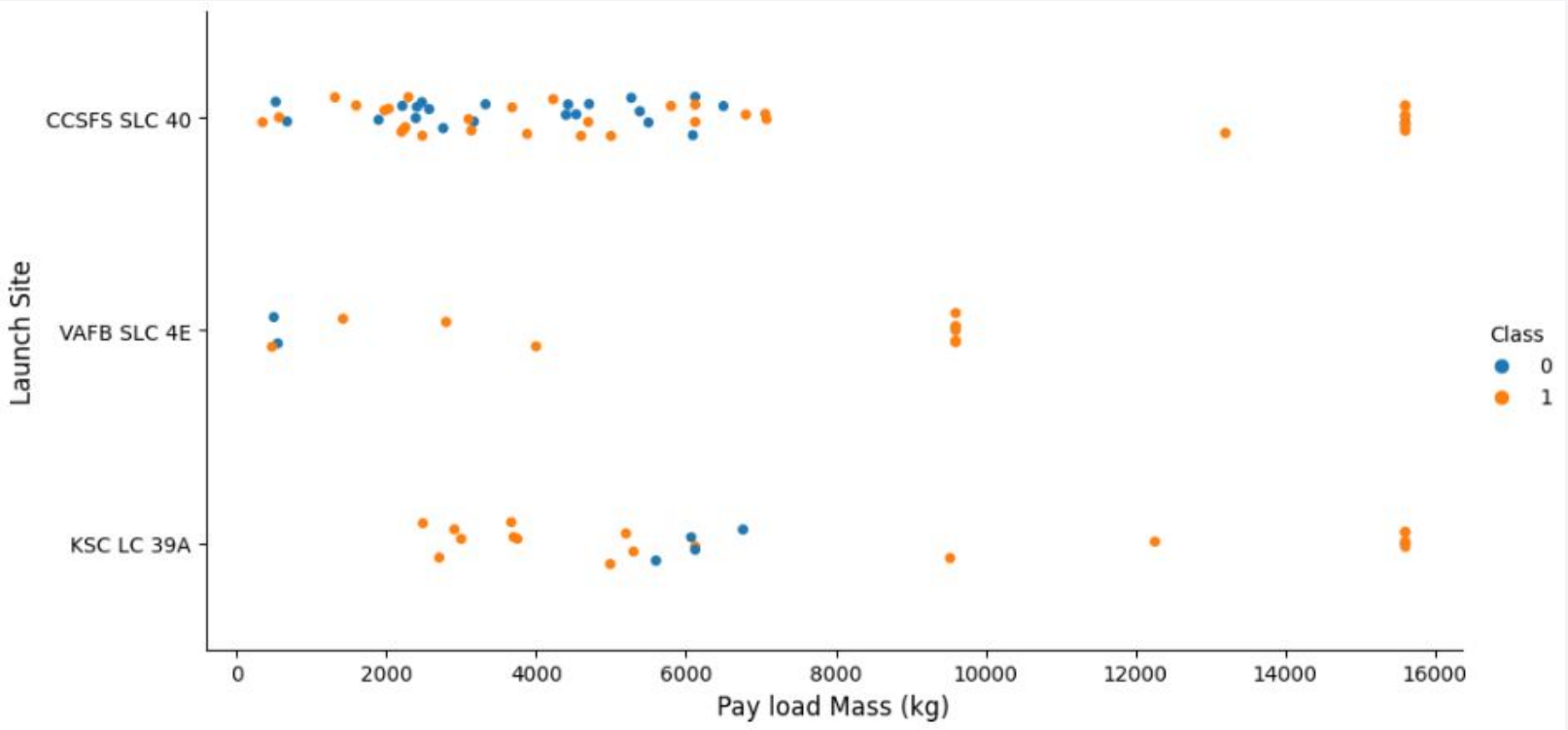
Flight Number vs. Launch Site

- A scatter plot of flight number vs launch site shows each flight represented as a point on a graph, with the x-axis representing the flight number and the y-axis representing the launch site.
- The points indicate whether the flight was successful (Class 1) or not (Class 0). As the flight number increases, we see an overall trend of more successful launches.



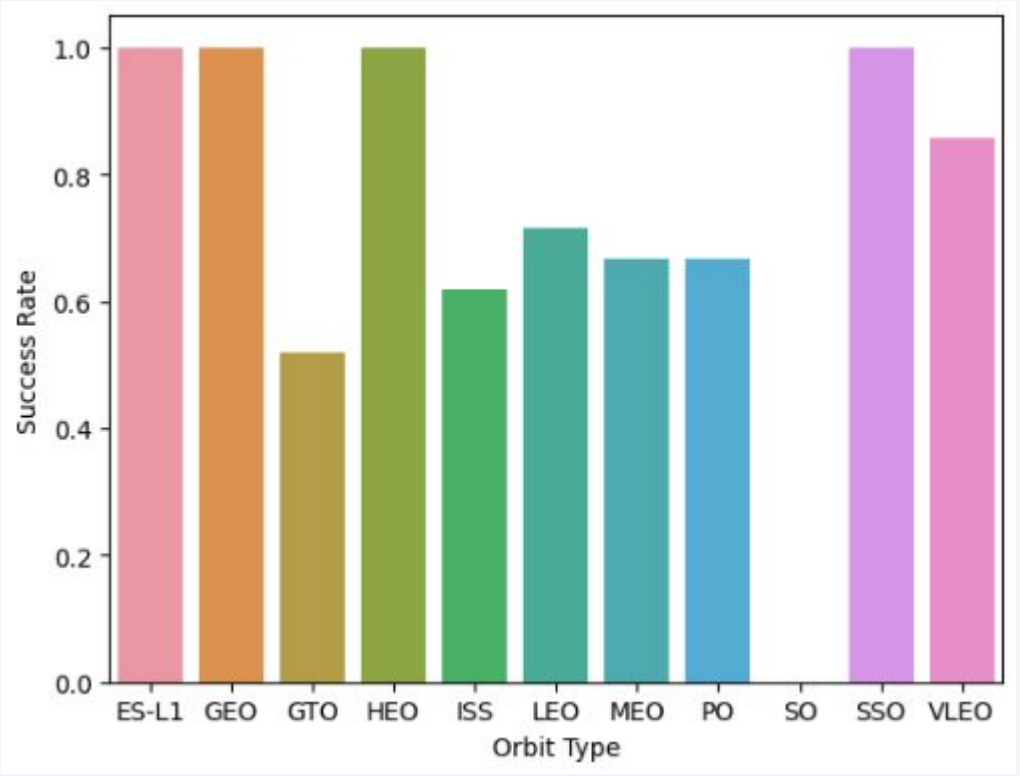
Payload vs. Launch Site

- Similarly here, we can see a trend of more successful launches happening in all launch sites that are better equipped to handle larger payloads.



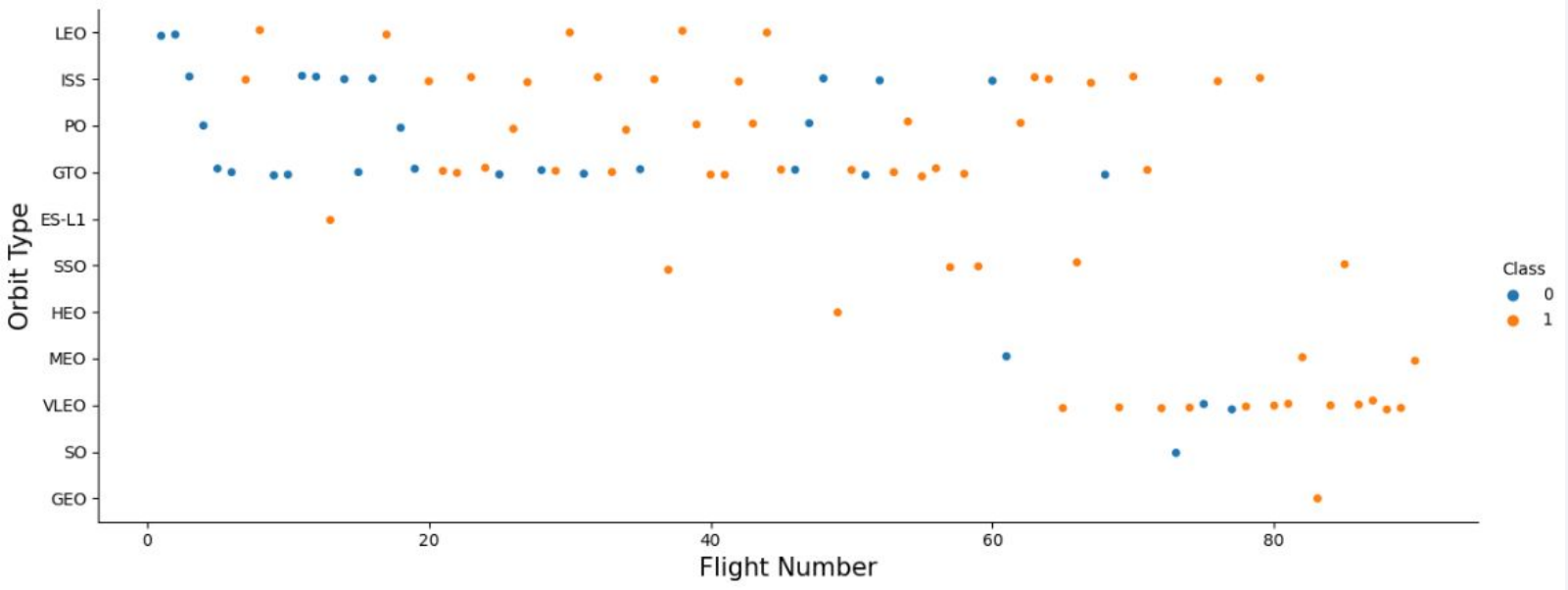
Success Rate vs. Orbit Type

- This bar chart illustrates the success rate of various orbit types, with the y-axis representing the success rate and the x-axis listing the different orbit types. It appears that launches sent from orbit types ES-L1, GEO, HEO, and SSO have a 100% success rate, while launches sent from the SO orbit type have a 0% success rate.



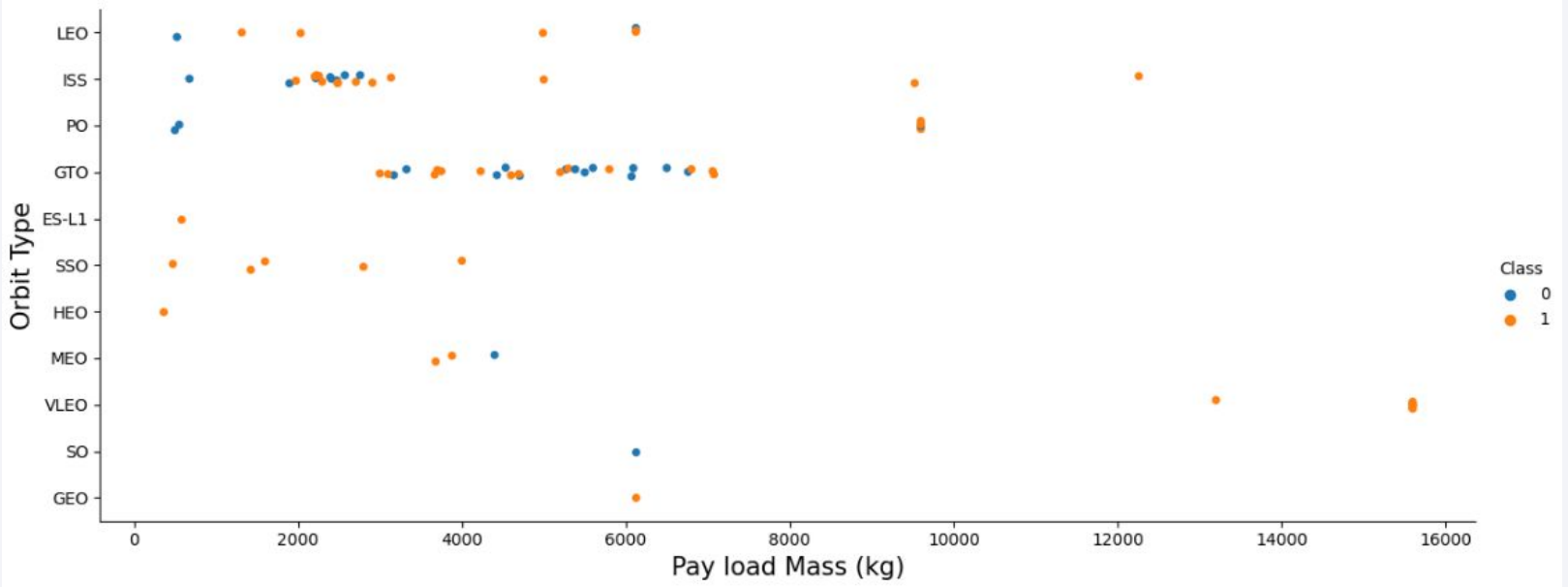
Flight Number vs. Orbit Type

- The scatter point does not indicate a correlation between the flight number and the orbit type.



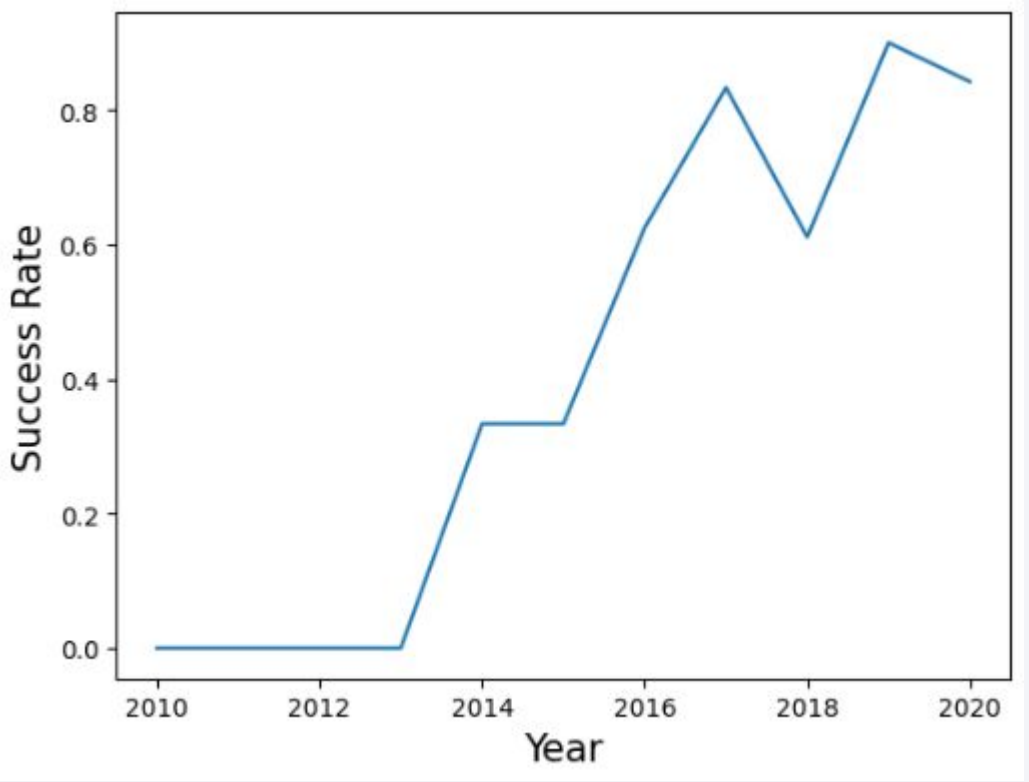
Payload vs. Orbit Type

- This scatter plot the distribution of the payloads for each launch over different orbit types.
- For the LEO, ISS or PO orbit types can notice a positive correlation as more launches were successful as the payload increased.



Launch Success Yearly Trend

- This line chart demonstrates the trend of success rate over the years, with the data points showing the average success rate of all launches in a particular year.
- It shows an increase in successful launches in recent years, indicating the improvement in technology and experience over time.



All Launch Site Names

```
%sql select LAUNCH_SITE, COUNT(LAUNCH_SITE) Missions from SPACEXDATASET GROUP BY LAUNCH_SITE ORDER BY COUNT(LAUNCH_SITE) DESC
```

- The above query is written in SQL and it selects the unique launch sites from the scraped data table under the Wikipedia SpaceX page and groups them by the launch site.

- The result shows the launch sites, CCAFS, KSC, Cape Canaveral, VAFB, and CCSFS, indicating that CCAFS launch site has the highest number of launches followed by KSC.

launch_site	missions
CCAFS	40
KSC	33
Cape Canaveral	20
VAFB	16
CCSFS	12

Launch Site Names Begin with 'CCA'

- The below SQL query shows 5 launch sites that begin with `CCA`, along with other data columns scraped from the Wikipedia page. It uses a where condition and a limit function to show only a small part of the data.

```
%sql select * from SPACEXDATASET WHERE LAUNCH_SITE LIKE 'CCA%' LIMIT 5
```

flight_no_	DATE	TIME	version_booster	launch_site	payload	payload_mass	orbit	customer	launch_outcome	booster_landing
1	2010-06-04	18:45:00	F9 v1.0	CCAFS	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
2	2010-12-08	15:43:00	F9 v1.0	CCAFS	Dragon	0	LEO	NASA (COTS) NRO	Success	Failure (parachute)
3	2012-05-22	07:44:00	F9 v1.0	CCAFS	Dragon	525	LEO	NASA (COTS)	Success	No attempt
4	2012-10-08	00:35:00	F9 v1.0	CCAFS	SpaceX CRS-1	4700	LEO	NASA (CRS)	Success	No attempt
5	2013-03-01	15:10:00	F9 v1.0	CCAFS	SpaceX CRS-2	4877	LEO	NASA (CRS)	Success	No attempt

Total Payload Mass

```
%sql select customer, sum(payload_mass) Total_Mass from SPACEXDATASET WHERE customer = 'NASA (CRS)' group by customer
```

- This SQL query calculates the total payload carried by boosters from NASA. The query uses a group by function and a where condition to show that sum.

customer	total_mass
NASA (CRS)	59941

Average Payload Mass by F9 v1.1

- The result shows the average payload mass of the booster "F9 v1.1". This query will give the average payload mass of all the launches done by this version of the booster.

```
%sql select version_booster, avg(payload_mass) Avg_Mass from SPACEXDATASET WHERE version_booster = 'F9 v1.1' group by version_booster
```

version_booster	total_mass
F9 v1.1	2534

First Successful Ground Landing Date

- The result shows the earliest date of a successful launch from the dataset. It gives an idea of the time frame in which the first successful launch took place.
- It helps track how many successful launches were made during a certain period of time.

```
%sql select min(date) Date from SPACEXDATASET WHERE launch_outcome = 'Success'
```

DATE
2010-06-04

Successful Drone Ship Landing with Payload between 4000 and 6000

- The query shows the different versions of the booster that have successfully landed on a drone ship while carrying a payload mass between 4000 and 6000.

```
%sql select distinct(version_booster) from SPACEXDATASET WHERE payload_mass between 4000 and 6000 and booster_landing = 'Success (drone ship)' order by version_booster
```

version_booster
F9 B5
F9 FT

Total Number of Successful and Failure Mission Outcomes

```
%sql select booster_landing, count(booster_landing) nr_succesful from SPACEXDATASET group by booster_landing ORDER BY count(booster_landing) DESC
```

booster_landing	nr_succesful
Success (drone ship)	64
No attempt	22
Success (ground pad)	16
Failure (drone ship)	8
Controlled (ocean)	5
Failure (parachute)	2
Uncontrolled (ocean)	2
Failure (ground pad)	1
Precluded (drone ship)	1

- The SQL query shows the different types of booster landing and how many successful launches were made with each type of landing. It indicates that most of the successful launches were made by landing the booster on a drone ship.

Boosters Carried Maximum Payload

- The result shows the different versions of the booster that have been used to carry the highest payload mass, and the number of times each version has been used to do so.

```
%sql select distinct(version_booster), count(version_booster) boosters_with_max from SPACEXDATASET \
where payload_mass in (select max(payload_mass) from SPACEXDATASET) group by version_booster
```

version_booster	boosters_with_max
F9 B5	24

2015 Launch Records

```
%sql select DATE, MONTHNAME(DATE) as Month, version_booster, launch_site, booster_landing from SPACEXDATASET where YEAR(DATE) = '2015' and booster_landing = 'Failure (drone ship)'
```

DATE	MONTH	version_booster	launch_site	booster_landing
2015-01-10	January	F9 v1.1	Cape Canaveral	Failure (drone ship)
2015-04-14	April	F9 v1.1	Cape Canaveral	Failure (drone ship)

- The query shows the specific date, version of booster, launch site and landing outcome for all the launches that took place in the year 2015 where the booster failed to land on a drone ship. It is useful to identify patterns that might have led to the failure of the booster landing on a drone ship.

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

- The query gives an idea of the progress of the company in terms of the number of successful launches per year, and it allows to track the company's performance over the years.
- It can be observed that the year 2016 had the most successful launches (8) followed by 2014 and 2015 (6).

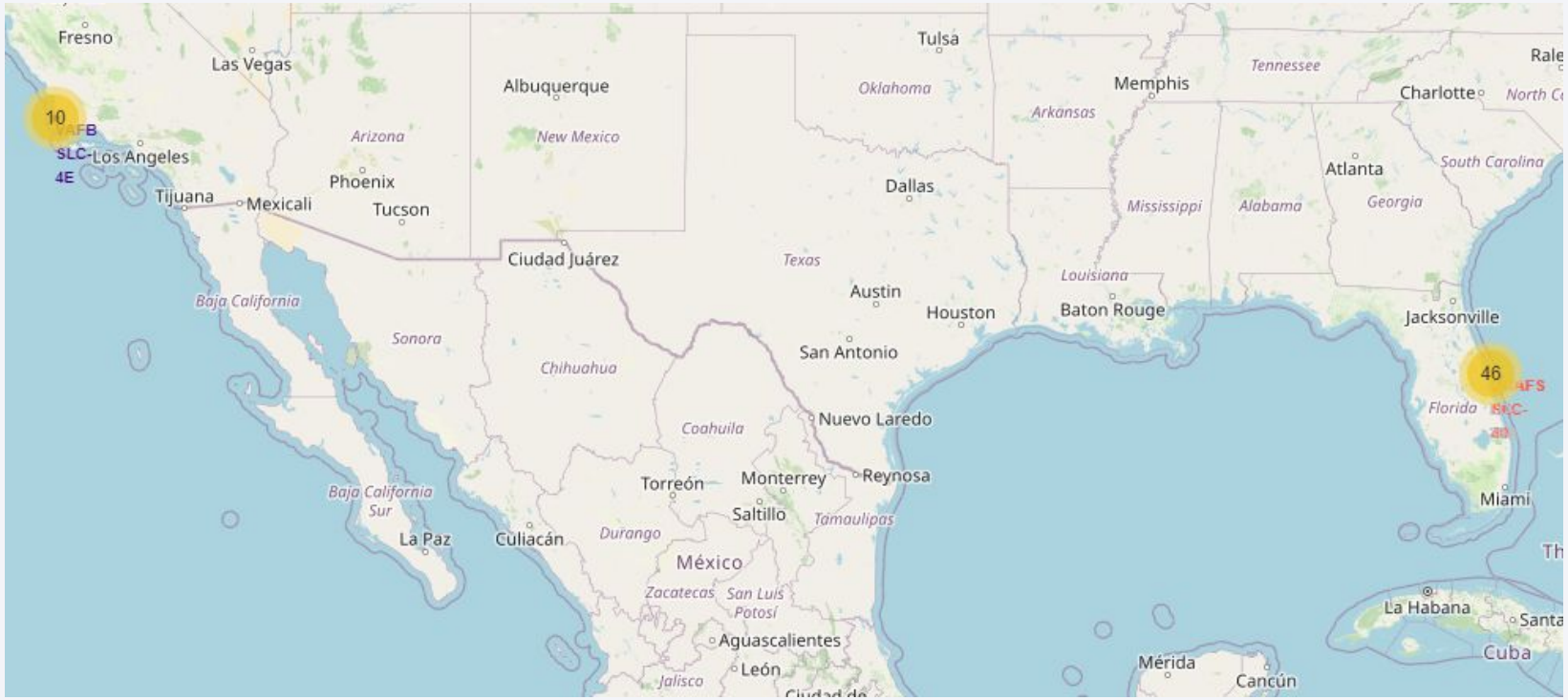
YEAR	succesful_landings
2016	8
2014	6
2015	6
2013	3
2017	3
2010	2
2012	2

```
%sql select YEAR(DATE) as YEAR, count(launch_outcome) succesful_landings from SPACEXDATASET \
where (DATE between '2010-06-04' and '2017-03-20') and launch_outcome = 'Success' group by YEAR(DATE) order by count(launch_outcome) desc
```


Section 3
Launch Sites
Proximities Analysis

Launch sites Folium Map

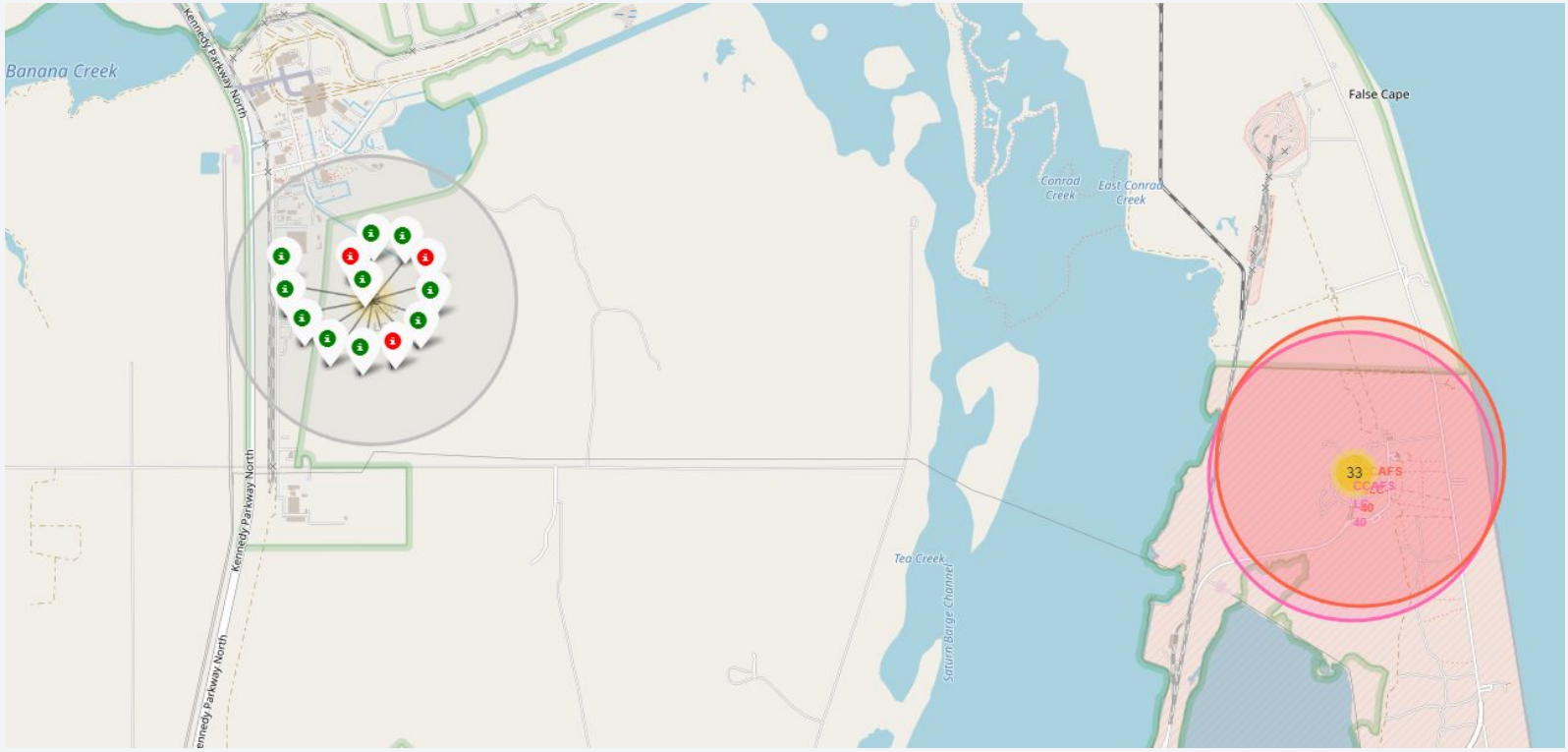
- We can notice that all 4 launch sites' locations are close to the coastline.
- This could provide easier access to the ocean for recovering boosters, and also a clear line of sight for launches.



KSC LC-39A launch site

- Created a Marker cluster to be able to add markers (green for successful and red for failed) to all launches in a site.

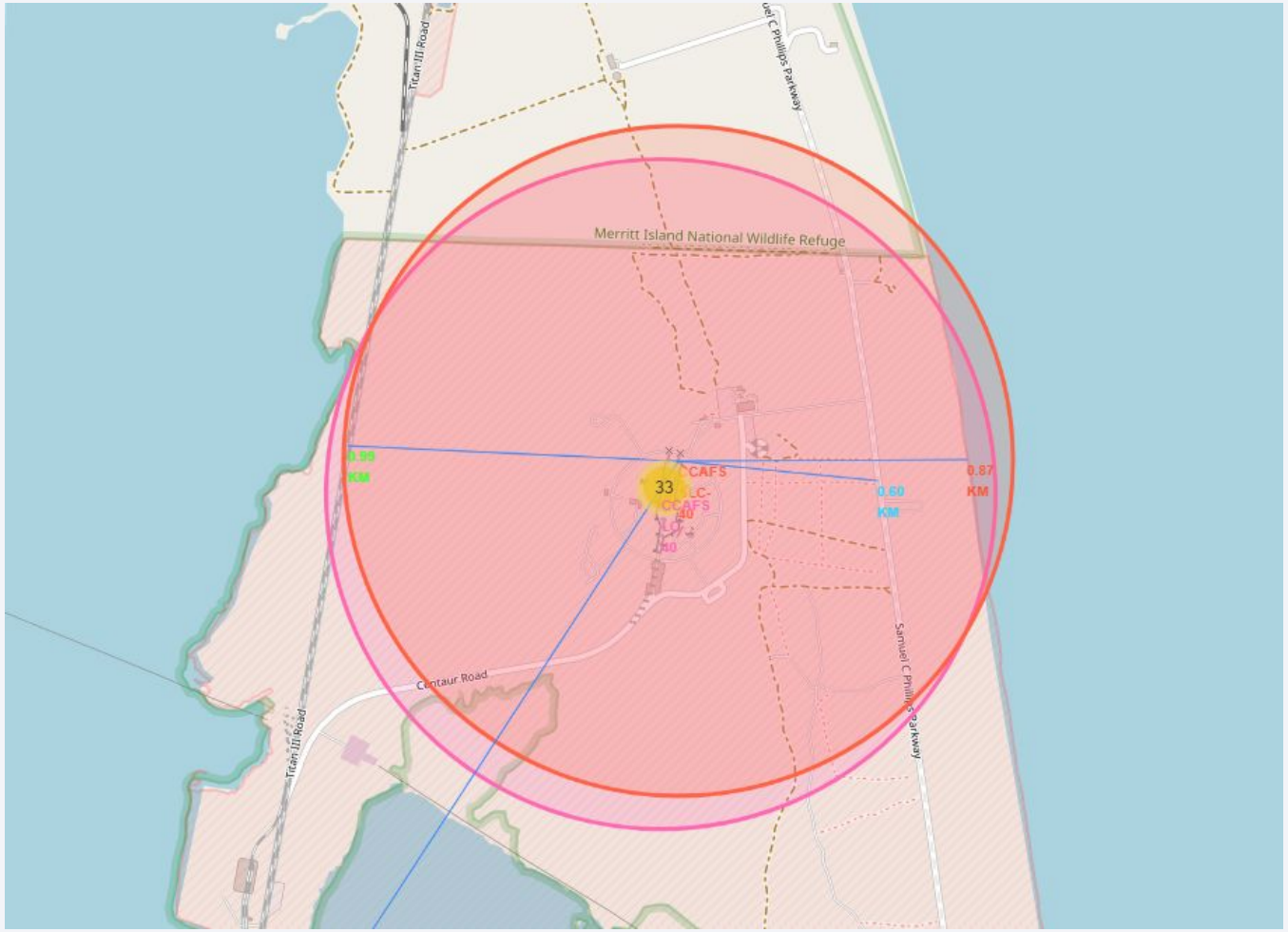
- We can identify that most launches were successful in KSC LC-39A.

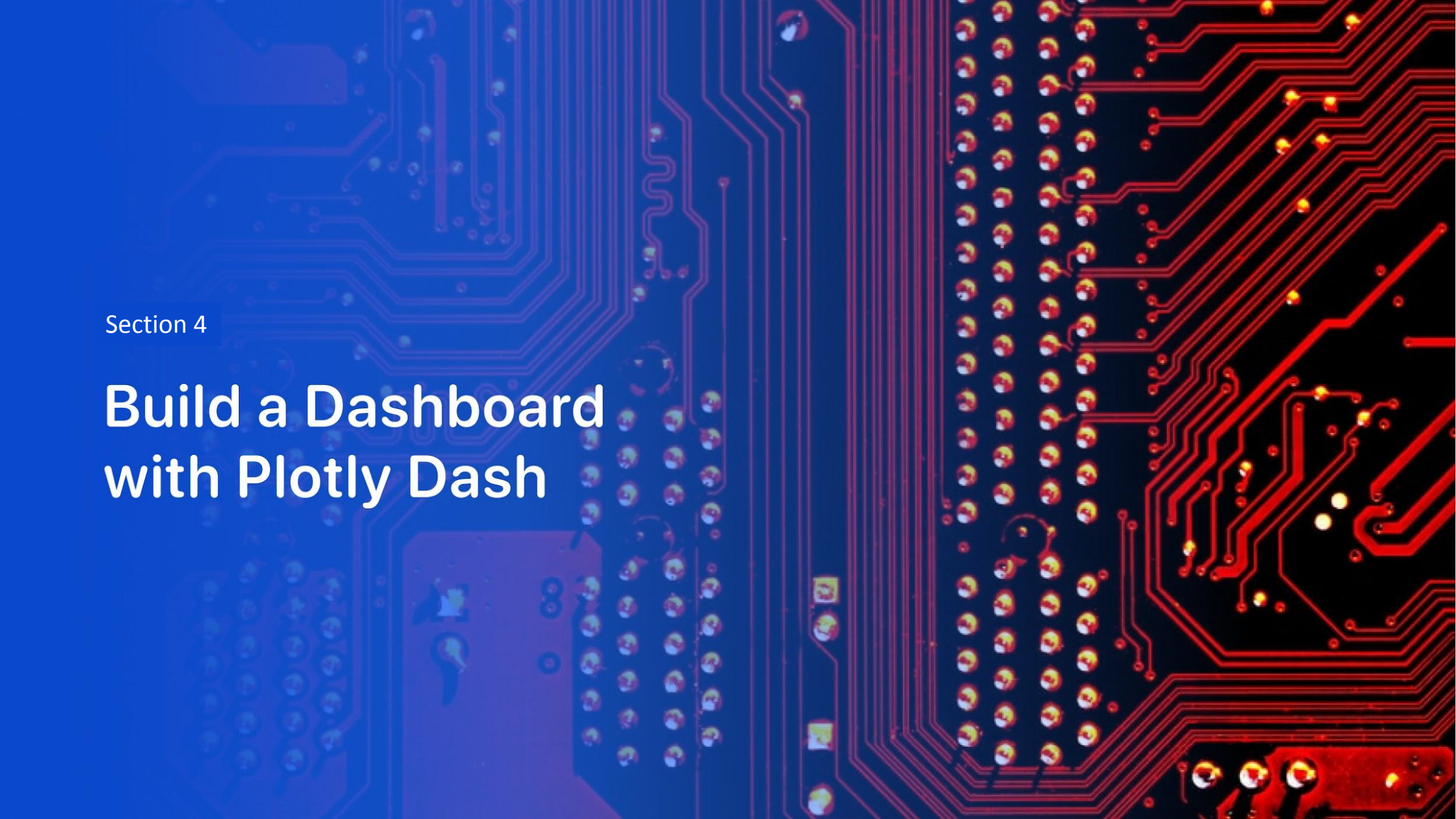


Polylines Folium map

- By using the Folium map and measuring the distances to closest points like coastlines, railways, highways and cities, we can see the proximity of these launch sites to these points of interest.

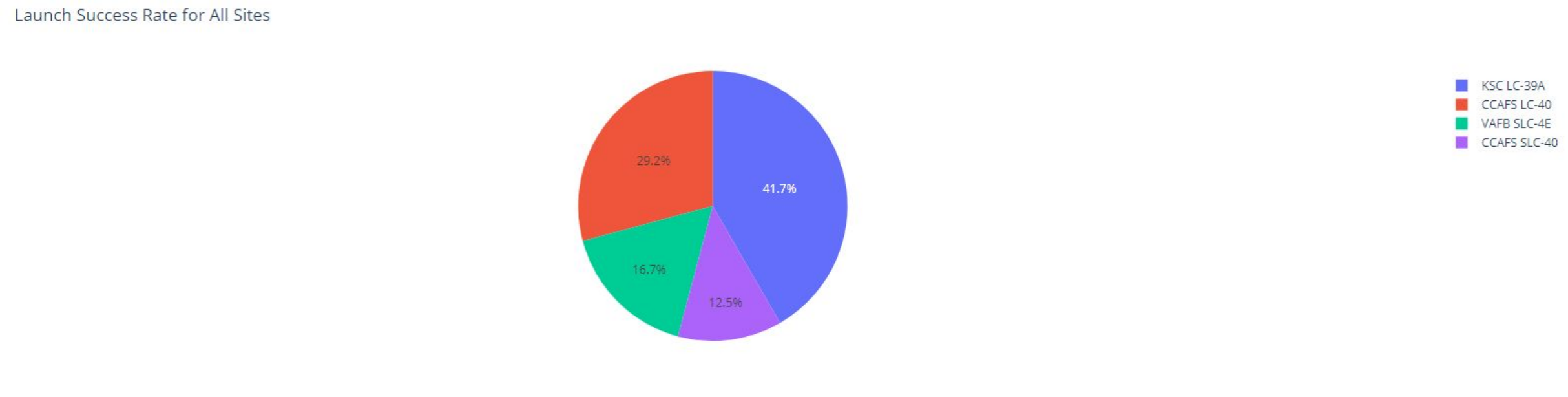
- This can be useful for visualizing the proximity of a launch site to various infrastructure and understanding how it may impact launch operations.




Section 4
Build a Dashboard with Plotly Dash

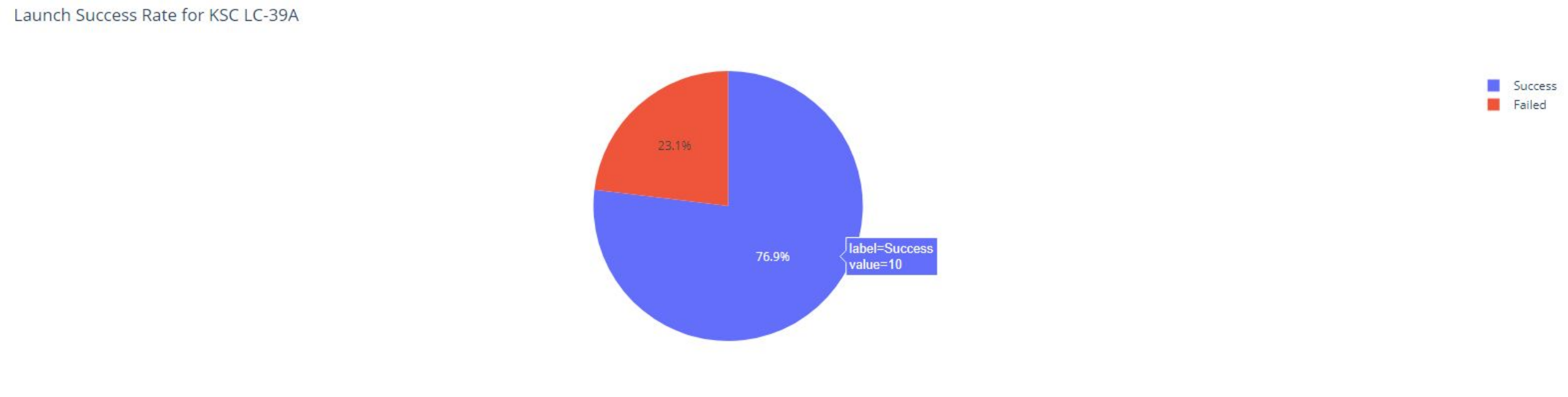
Plotly Dash Launch Success Rate

- We used Plotly Dash to create a pie chart that shows the distribution of successful launches. The most successes were on KSC LC-39A from the total launches.
- This can be used to identify which sites may require improvements or maintenance in order to increase their success rate.



Success Rate for KSC LC-39A

- The pie chart shows that out of a total of 13 launches from KSC LC-39A, 10 launches were successful (77%) and 3 launches failed (23%).



Success Rate based on payload mass

- The scatter plot shows the correlation between payload weight and launch outcome for all launch sites.
- We can see on the first screenshot that most launches with payload of 0-2000 kg have failed.
- At first glance on the second screenshot there wouldn't be much correlation between the success rate and the payload but if we look closer, most of the FT boosters were successful with a higher payload (2000-6000kg).

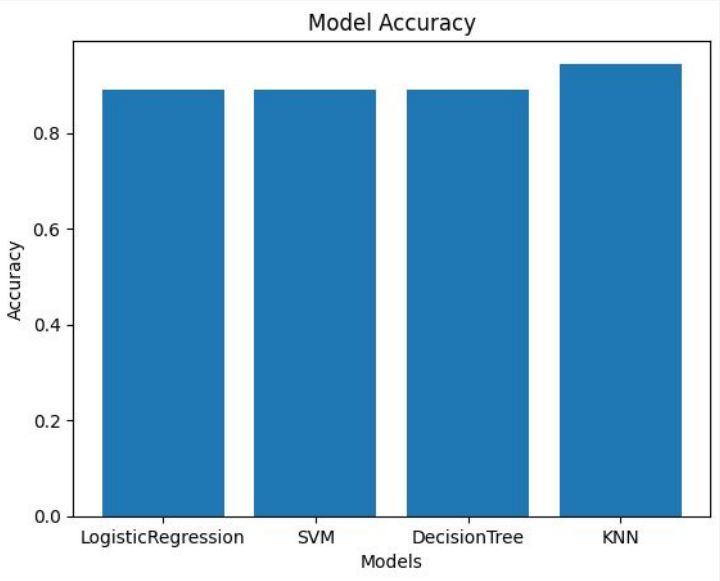
Section 5

Predictive Analysis (Classification)

Classification Accuracy

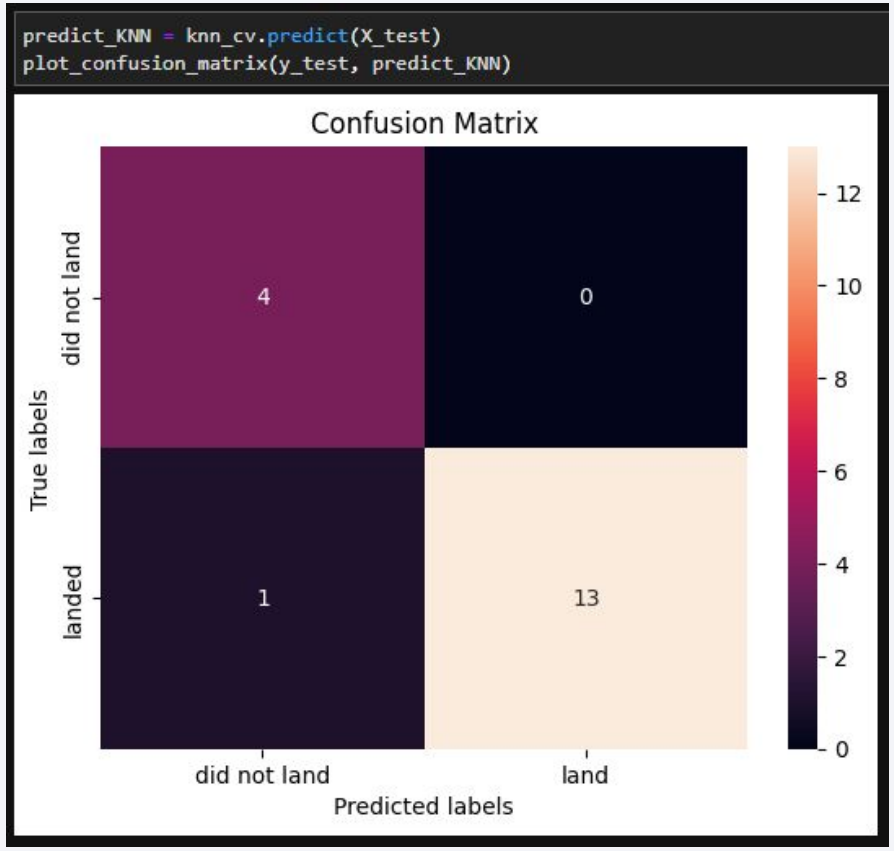
- The accuracy of the built models on the test data is as follows:
  - ❏ LogisticRegression: 88.89%
  - ❏ SVM: 88.89%
  - ❏ Decision Tree: 88.89%
  - ❏ KNN: 94.44%
- The KNN model has the highest accuracy at 94.44%, while the other models (LogisticRegression, SVM, Decision Tree) had an accuracy of 88.89%.

	Model	Accuracy Test Data
0	LogisticRegression	0.888889
1	SVM	0.888889
2	DecisionTree	0.888889
3	KNN	0.944444



Confusion Matrix

- The confusion matrix is used to determine the number of true positives, true negatives, false positives, and false negatives.
- In this case, the best performing model is KNN with an accuracy of 94.44%. The confusion matrix for this model shows 4 true positives, 0 false positives, 1 false negative, and 13 true negatives.
- Overall, the model is performing well with a low number of false positives and false negatives, indicating that it is accurately classifying the majority of observations.



Conclusions

- Success rate increased over the year as technology has improved, we can see that same correlation with the number of flights.
- More successful launches happened where rockets were better equipped to handle larger payloads.
- Some orbits like ES-L1, GEO, HEO, and SSO have a 100% success rate on launches.
- Being closer to highways and railroads is important to facilitate the transportation of components. Equally important is for launch sites to be located outside of densely populated urban areas to minimize the risk of damage or injury to people and property. Being close to coastlines also allows for rockets to be launched over the ocean.
- The best performing model out of the list of classification models (Logistic Regression, SVM, Decision Tree, and KNN) was KNN with an accuracy of 94.44% on the test data.

Appendix

- https://github.com/sfs-projects/ml/tree/develop/ibm-data-science-project/

Thank you!