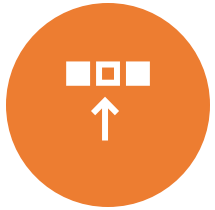
IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Charles Lee Ying
December 05, 2024

Outline



EXECUTIVE SUMMARY

INTRODUCTION

METHODOLOGY

RESULTS

CONCLUSION

APPENDIX

Executive Summary

Methodologies

- Data Collection via API calls and Web Scraping
- Data Wrangling using pandas and numpy
- Exploratory Data Analysis using SQL and visualization
- Visual analytics using Folium and Plotly Dash
- Classification Model creation, tuning and testing

Results

A model can be built, using historical data such as launch site, payload and booster version, to predict the possibility of successfully landing the first stage booster. It could also be noted that as time progressed and SpaceX gained more experience, they are better able to create the right conditions that favors the desired outcome.

Introduction

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 realised by SpaceX is due to reuse of 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 by an alternate company to verify if they can be competitive against SpaceX for a rocket launch.

- The aim of this project is to create a classification model using historical data, to predict landing a Falcon 9 first stage successfully.
- We will also verify what is the best Classification Model to be used for these predictions.

Section 1

Methodology

Methodology

- Executive Summary
- Data collection methodology:
  - Data collected via calls to Spacex APIs (e.g. https://api.spacexdata.com/v4/rockets/)
  - Web scraping of Wikipedia data (https://en.wikipedia.org/wiki/List_of_Falcon_9_and_Falcon_Heavy_launches)
- Perform data wrangling
  - Filtered the collected data.
  - Cleaning of data and dealing with missing values.
  - One Hot Encoding of categorical data to facilitate analysis.

Methodology

- Executive Summary
- Perform exploratory data analysis (EDA) using visualization and SQL
    - Several graphs/charts were plotted to analyze the relationships between the payload mass, booster version and launch site.
- Perform interactive visual analytics using Folium and Plotly Dash
    - A map was created to analyze launch site data, as well as a dashboard to look at the ratios of successful launches per site vs payload mass and booster version.
- Perform predictive analysis using classification models
    - Several classification models were built, tuned, and evaluated to identify the best hyperparameters for use in predicting future launches/landings.

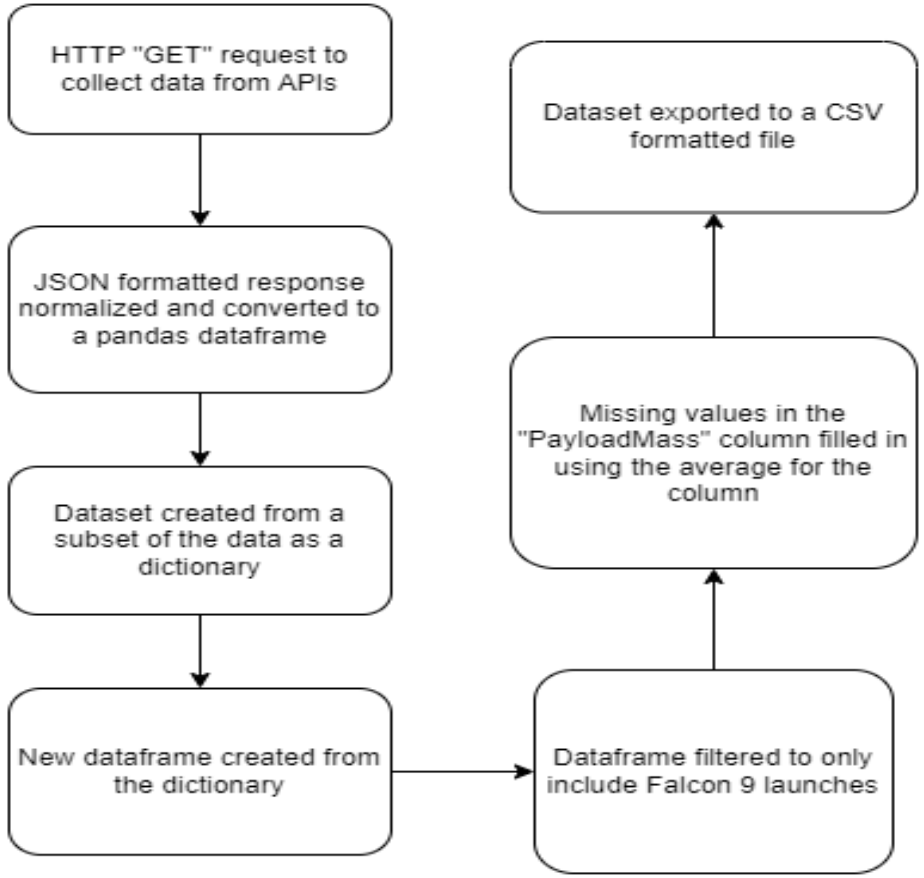
Data Collection

- Using JupyterLabs Notebooks, data was collected from the SpaceX APIs for rockets, launchpads, payloads and cores.
- The collected API data was processed into a suitable format for use in exploratory data analysis. In this instance, custom functions were used to extract a subset of the data into a dictionary. This was converted into a new dataframe eand subsequently xported to a CSV file after further filtering and processing.
- Historical launch data was collected by scraping the Wikipedia page for Falcon 9 and Falcon Heavy launches (https://en.wikipedia.org/wiki/List_of_Falcon_9_and_Falcon_Heavy_launches).
- The HTML tables containing these historical records were parsed using BeautifulSoup into a dictionary. This was subsequently converted to a pandas dataframe to provide the required data for analysis.

Data Collection – SpaceX API

- API Data collected with a GET request, and the JSON formatted response was normalized and converted to a pandas dataframe. Custom functions were used to extract data to a dictionary. This subset of data was then converted into a new dataframe before exporting to a CSV file.

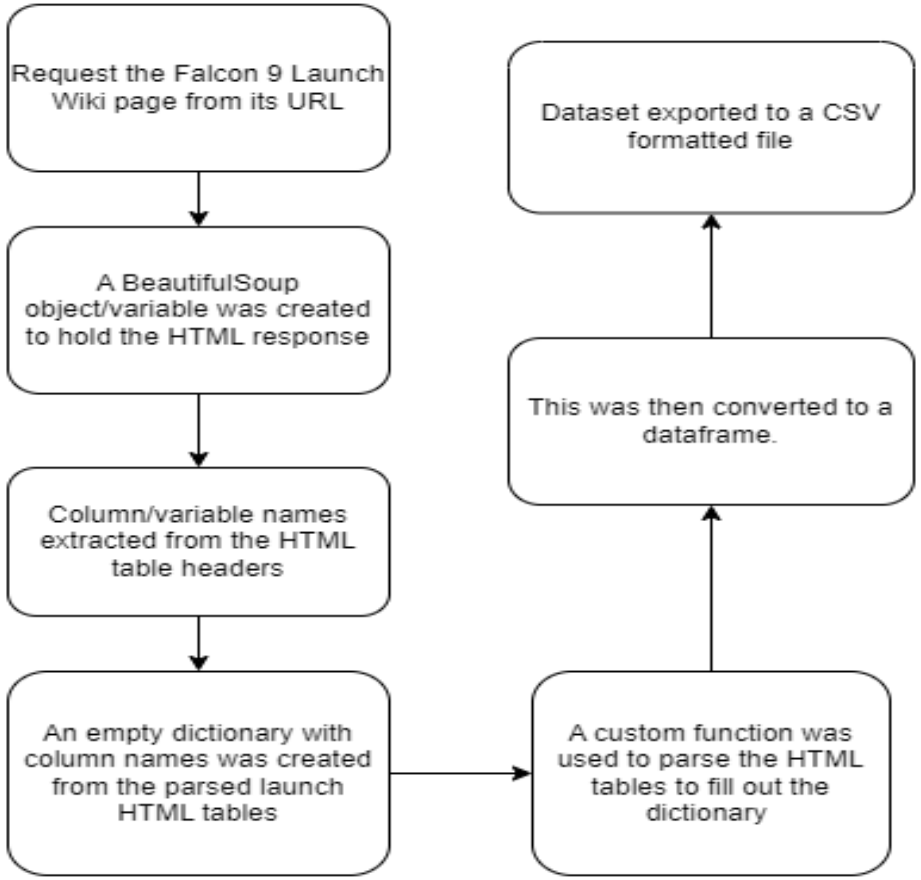
[GitHub: Data Collection - API]



Data Collection - Scraping

- The HTML tables containing these historical records were parsed into a dictionary using BeautifulSoup, and subsequently converted to a pandas dataframe to provide the required data for analysis.

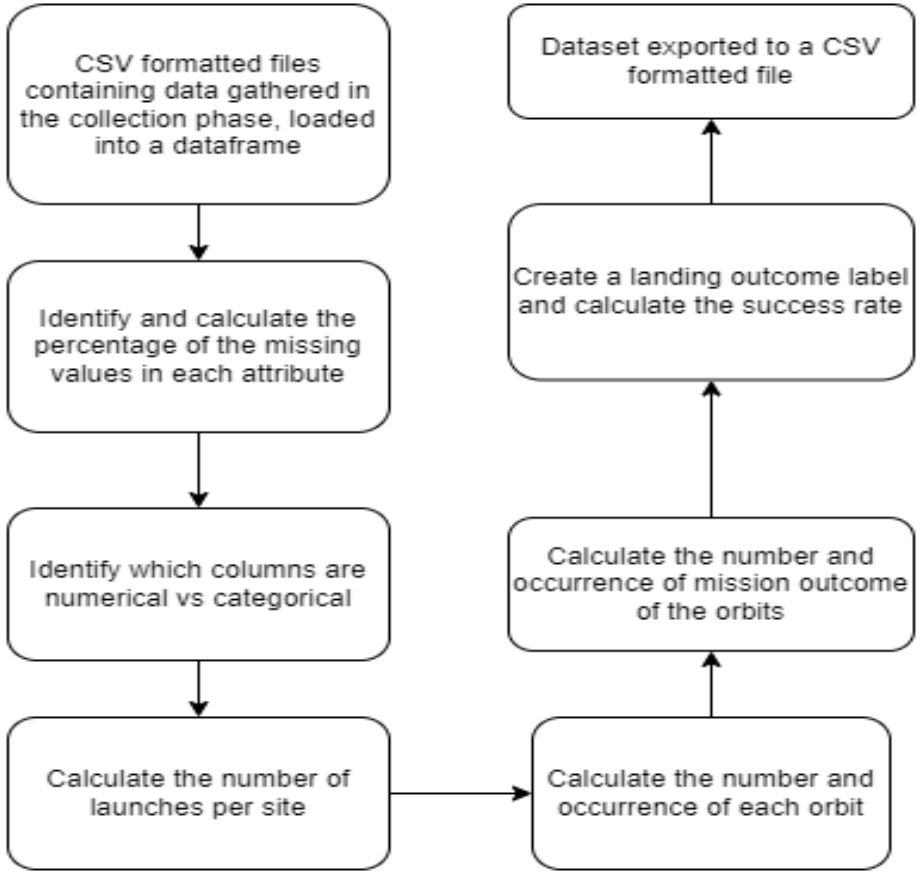
[GitHub: Data Collection - Webscraping]



Data Wrangling

- Using JupyterLabs Notebooks, the CSV formatted files from the data collection phase were loaded into a dataframe, where the data was checked for null values and to identify categorical vs numerical data.
- Several calculations were also done to determine the count of launches per launch site, as well as the count and success ratio of missions based on orbit type. Finally, a label for successful vs failed first launch landings was created.

[GitHub: Data Wrangling]



EDA with Data Visualization

- Several charts, such as count of flights per launch site, payload mass versus orbit type or launch site and yearly success trend, were plotted to identify what factors had the greatest impact on the success of landing the first stage booster.
- These charts were used to visualize the ratio of successful first stage landings and the most common factors associated with this outcome.

GitHub: Exploratory Data Analysis with Data Visualization

EDA with SQL

- Exploratory Data Analysis (EDA) with SQL
  - 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 succesful landing outcome in ground pad was acheived.
  - List the boosters which have success in drone ship and payload mass between 4000 and 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.

GitHub: Exploratory Data Analysis with SQL

Build an Interactive Map with Folium

- An interactive map, built on Folium, was developed to provide an interactive visualization of launch site success ratios. To this end, colored circles were used to highlight launch sites, while markers were employed to show the success ratios. Lines were added to highlight the distance from a launch site to nearby notable features such as a city, highway or railway track.
- These objects were added to the map to mark all the launch sites from the data set. The markers gave an interactive visual highlight of the ratio of failed to successful launches per site, while the distances between a launch site and its proximities were to assist in finding any geographical patterns about launch sites.

GitHub: Interactive Map with Folium

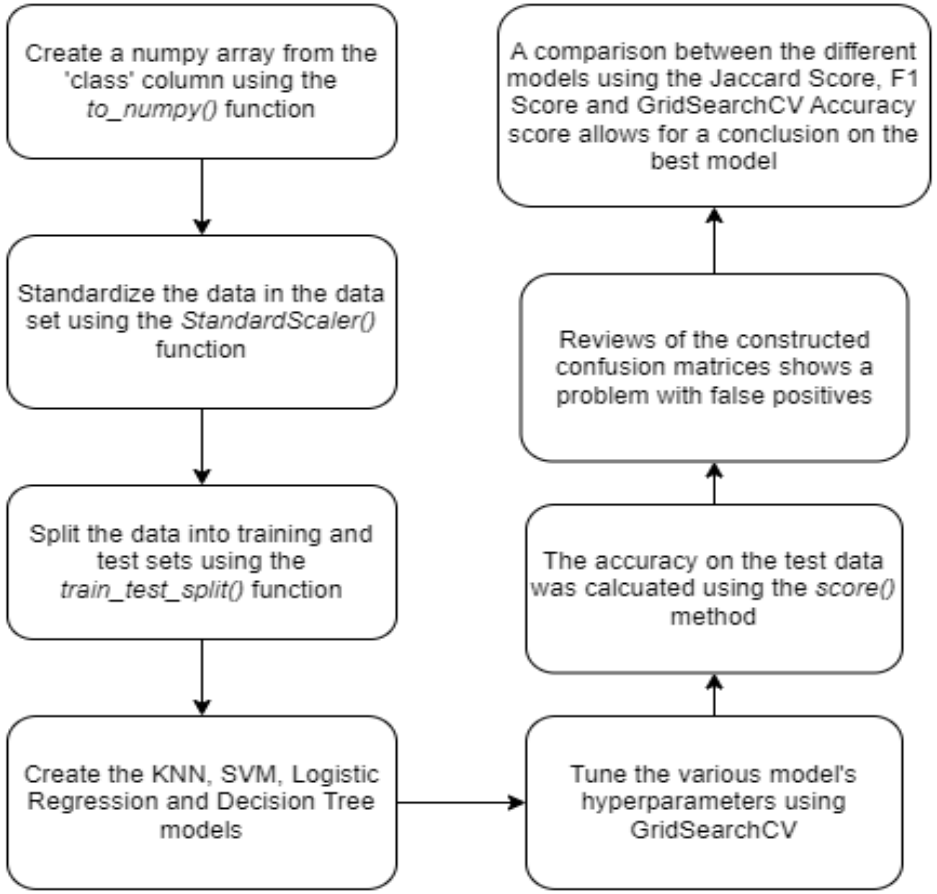
Build a Dashboard with Plotly Dash

- A Plotly Dash dashboard, consisting of a dropdown menu, pie chart and scatter plot was built giving you the option to select either all of the launch sites as a group view or a single site. The pie chart displays the percentage of successful launches versus failed launches, while the scatter plot highlights the number of launches per site based on the booster version and payload mass.
- The dashboard elements allows the user to explore the correlation between payload and launch success, while also being able to determine the relevant launch site.

GitHub: Dashboard with Plotly Dash

Predictive Analysis (Classification)

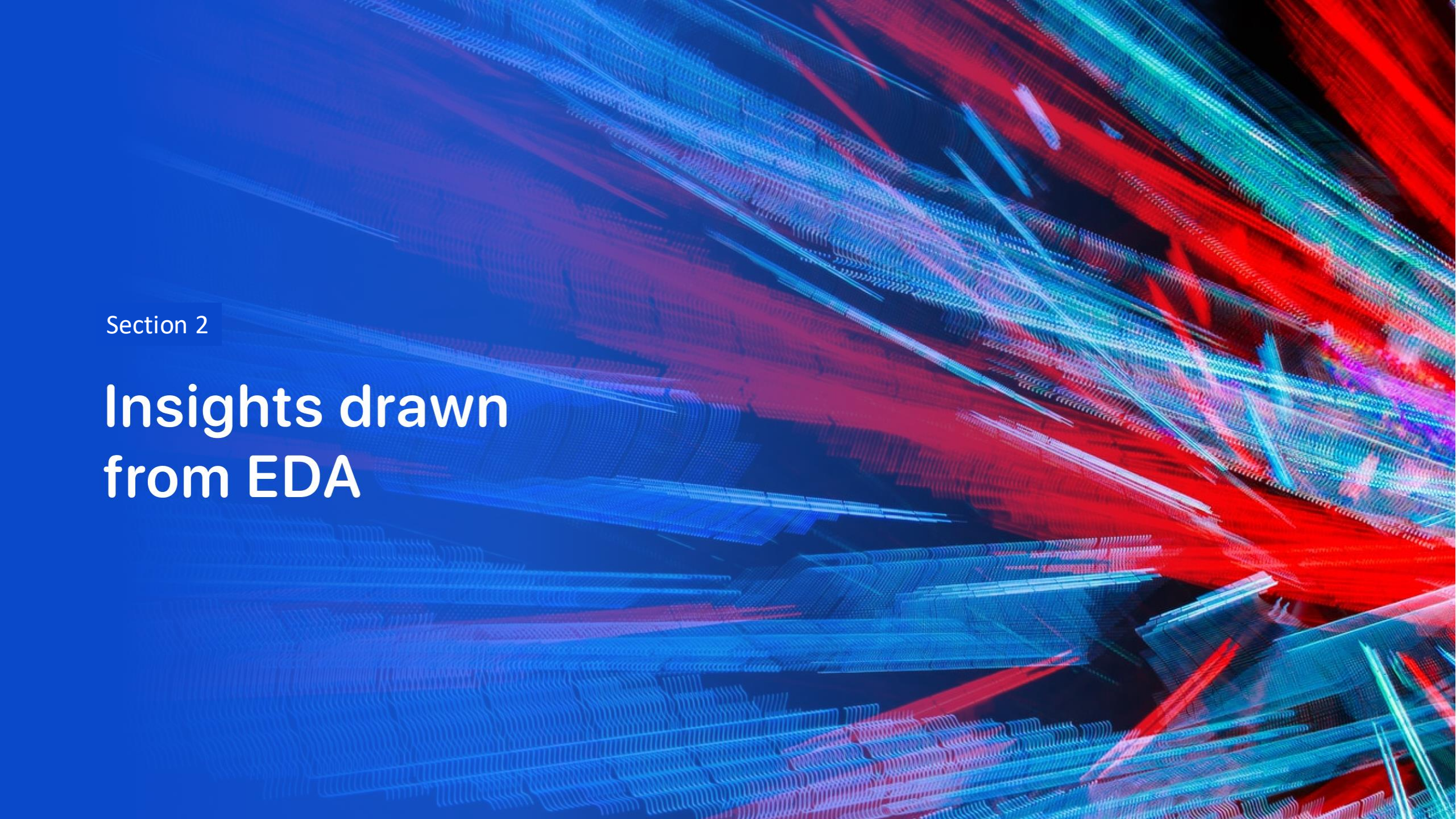
The collected data was standardized and split into training and test data. Four classification models were built and their hyperparameters tuned using GridSearchCV. Based on the Accuracy Score, the Decision Tree Classification Model appears to be the best option for predicting first stage landings.



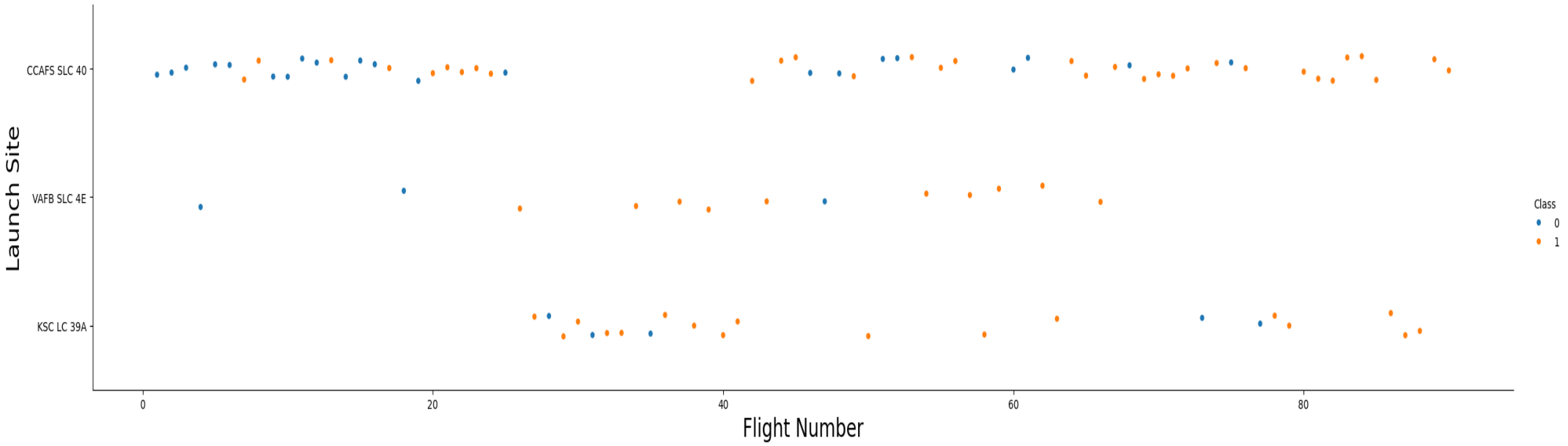
[GitHub: Predictive Analysis (Classification)]

Results

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

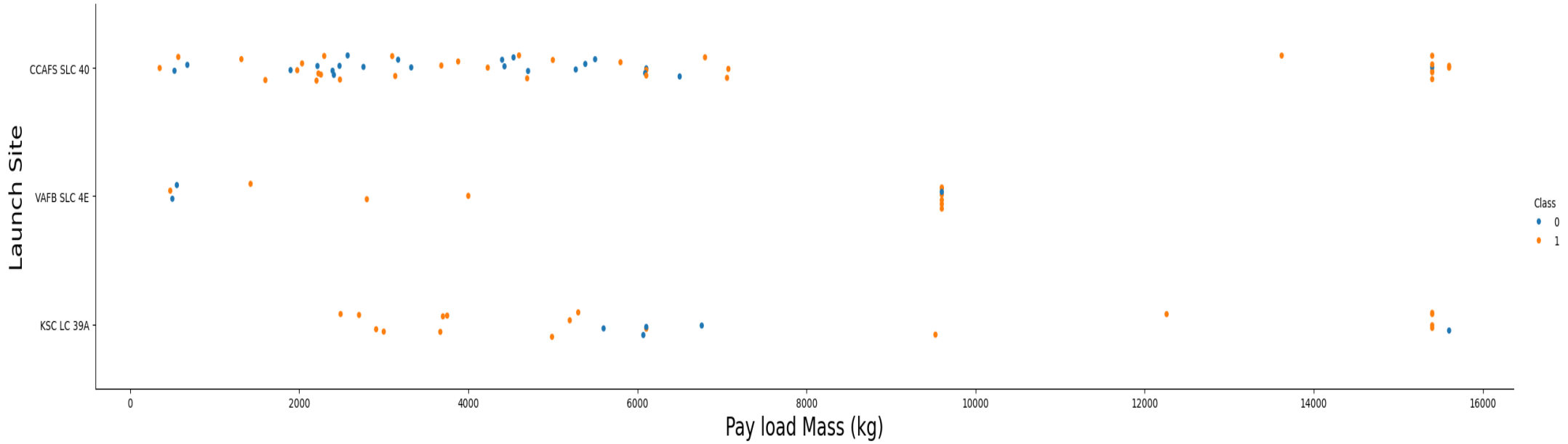

Section 2
Insights drawn from EDA

Flight Number vs. Launch Site



- There is a greater number of failures in the earlier time period and intial launches. As SpaceX continues launching rockets, we see a higher ratio of successful outcomes.

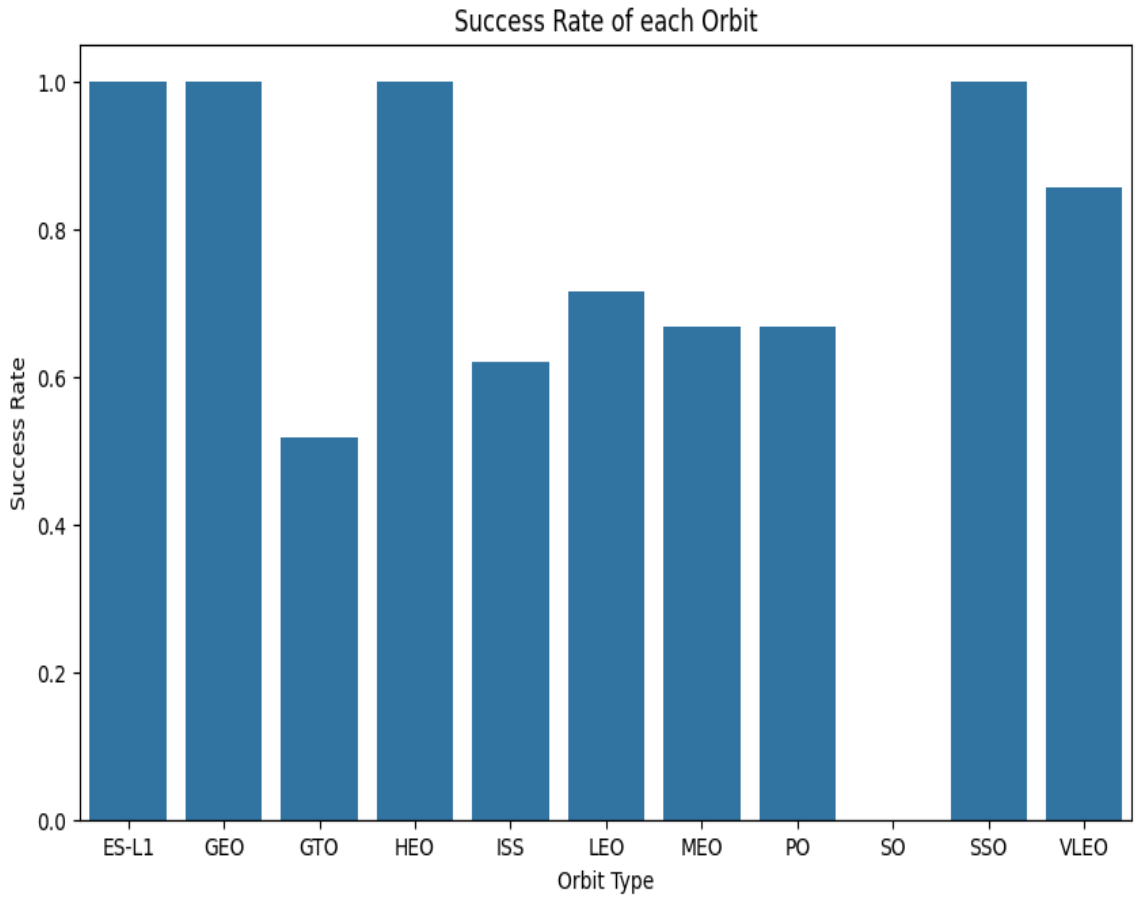
Payload vs. Launch Site



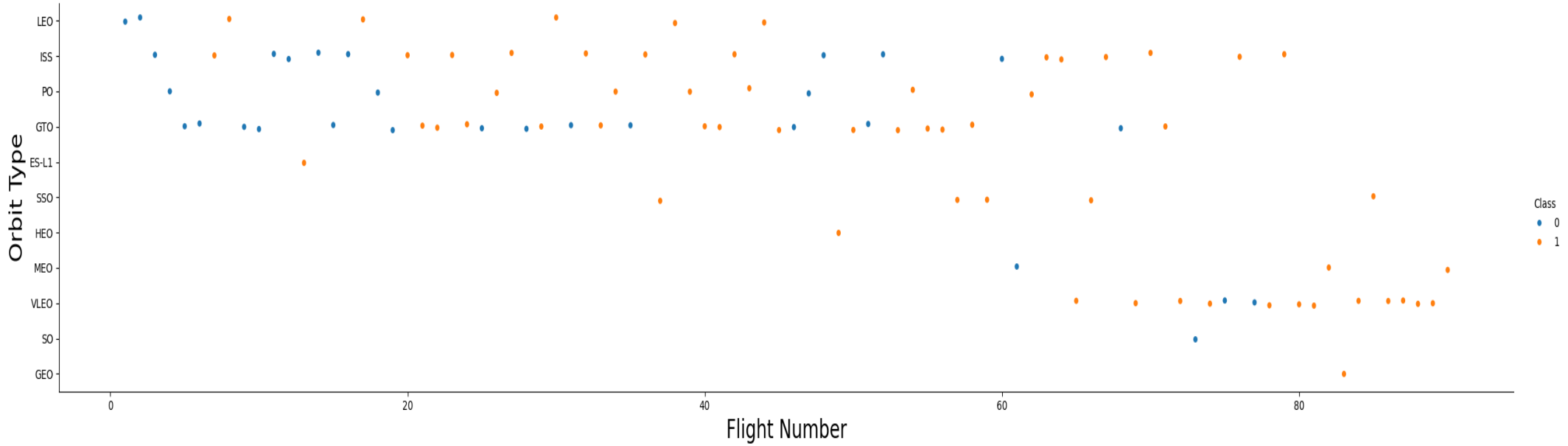
- Now if you observe Payload Mass Vs. Launch Site scatter point chart you will find for the VAFB-SLC launchsite there are no rockets launched for heavy payload mass (greater than 10000).

Success Rate vs. Orbit Type

- The ES-L1, GEO, HEO and SSO orbit types have the highest success rates, followed by LEO. There seems to be a correlation between successful outcomes and the orbit type.

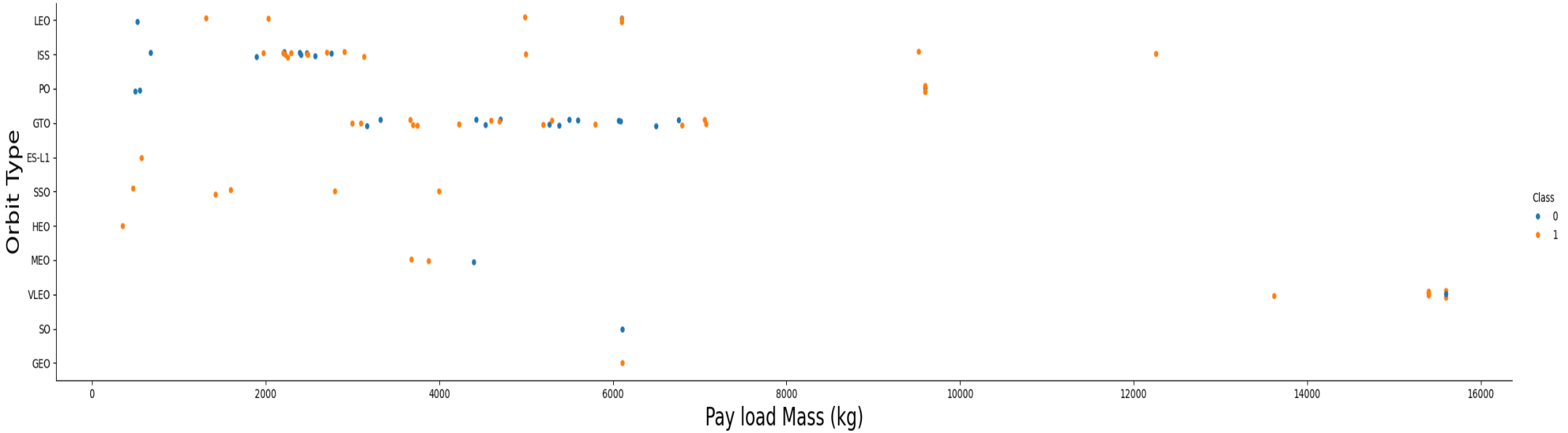


Flight Number vs. Orbit Type



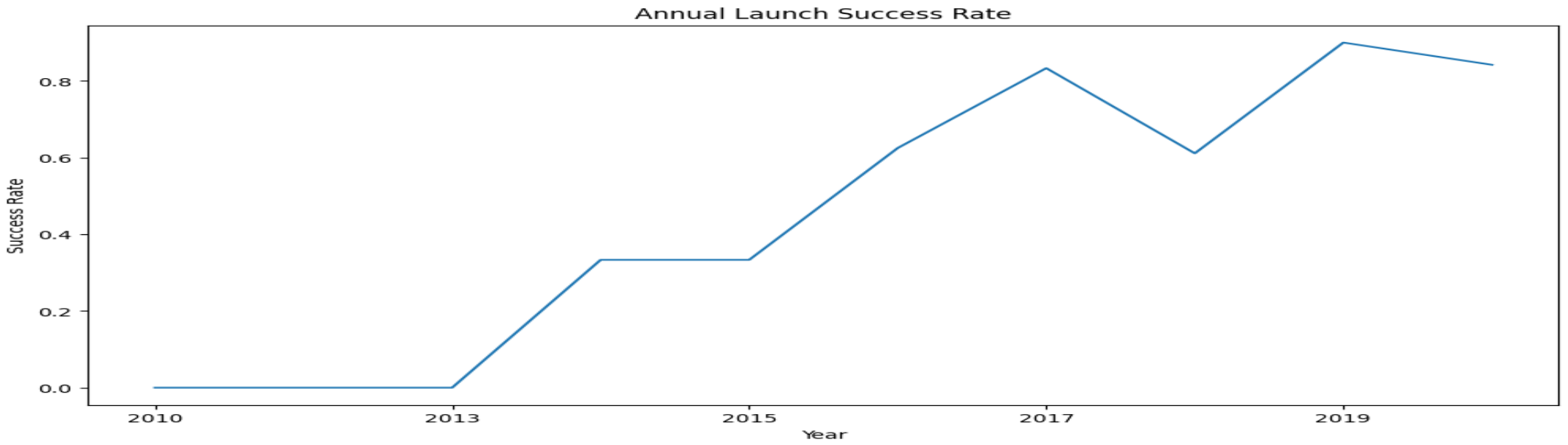
- You can observe that in the LEO orbit, success seems to be related to the number of flights. Conversely, in the GTO orbit, there appears to be no relationship between flight number and success.

Payload vs. Orbit Type



- With heavy payloads the successful landing or positive landing rate are more for Polar,LEO and ISS. However, for GTO, it's difficult to distinguish between successful and unsuccessful landings as both outcomes are present.

Launch Success Yearly Trend



- You can observe that the sucess rate since 2013 kept increasing till 2020.

All Launch Site Names

- Find the names of the unique launch sites:
  - **%sql** select distinct Launch_Site from SPACEXTABLE
- List of Launch sites in use by SpaceX according to historical records.

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

```
0]: %sql select distinct Launch_Site from SPACEXTABLE;
```

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

0]:

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

Launch Site Names Begin with 'CCA'

- Find 5 records where launch sites begin with `CCA`:
  - `%sql select * from SPACEXTABLE where Launch_Site like "CCA%" limit 5;`
- List of launches based on a filter criteria of sites with a name beginning with 'CCA'.

Display 5 records where launch sites begin with the string 'CCA'

```
%sql select * from SPACEXTABLE where Launch_Site like "CCA%" limit 5;
```

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

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

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

```
%sql select sum(PAYLOAD_MASS__KG_) as "Total Payload Mass" from SPACEXTABLE where Customer like "NASA%";
```

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

Total Payload Mass
99980

- Calculate the total payload carried by boosters from NASA:
  - `%sql select sum(PAYLOAD_MASS__KG_) as "Total Payload Mass" from SPACEXTABLE where Customer like "NASA%";`
- Query to calculate the total tonnage – 99,980 Kg, hauled by SpaceX for NASA.

Average Payload Mass by F9 v1.1

Display average payload mass carried by booster version F9 v1.1

```
[13]: %sql select avg(PAYLOAD_MASS__KG_) as "Average F9 v1.1 Payload Mass" from SPACEXTABLE \
          where Booster_Version like "F9 v1.1%";
```

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

[13]:

Average F9 v1.1 Payload Mass
2534.6666666666665

- Calculate the average payload mass carried by booster version F9 v1.1:
  - `%sql select avg(PAYLOAD_MASS__KG_) as "Average F9 v1.1 Payload Mass" from SPACEXTABLE \`
  - `where Booster_Version like "F9 v1.1%";`

- Average of tonnage – 2534.67 Kg, hauled by SpaceX using their F9 v1.1 booster.

First Successful Ground Landing Date

List the date when the first succesful landing outcome in ground pad was acheived.

Hint:Use min function

[14]:
```
%sql select min(Date) as "First Successful Ground Pad Landing Date" from SPACEXTABLE \
    where Landing_Outcome = "Success (ground pad)";
```

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

[14]:

First Successful Ground Pad Landing Date
2015-12-22

- Find the dates of the first successful landing outcome on ground pad:
  - `%sql select min(Date) as "First Successful Ground Pad Landing Date" from SPACEXTABLE \`
  - `    where Landing_Outcome = "Success (ground pad)";`

- First successful landing on a ground pad on 2015-12-22.

Successful Drone Ship Landing with Payload between 4000 and 6000

- List the names of boosters which have successfully landed on drone ship and had payload mass greater than 4000 but less than 6000:
  - `%sql select distinct Booster_Version from SPACEXTABLE where Landing_Outcome = "Success (drone ship)" \`
  - `and PAYLOAD_MASS__KG_ between 4000 and 6000;`
- List of boosters successfully landed to a drone ship, with payload tonnage between 4000 Kg and 6000 Kg.

List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000

```
%sql select distinct Booster_Version from SPACEXTABLE where Landing_Outcome = "Success (drone ship)" \
    and PAYLOAD_MASS__KG_ between 4000 and 6000;
```

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

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

```
%sql select Mission_Outcome, count(Landing_Outcome) as "Outcome Count" from SPACEXTABLE group by Mission_Outcome;
```

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

Mission_Outcome	Outcome Count
Failure (in flight)	1
Success	98
Success	1
Success (payload status unclear)	1

- Calculate the total number of successful and failure mission outcomes:
  - `%sql select Mission_Outcome, count(Landing_Outcome) as "Outcome Count" from SPACEXTABLE group by Mission_Outcome;`

- Count of successful (100) versus failed (1) mission outcomes.

Boosters Carried Maximum Payload

- List the names of the booster which have carried the maximum payload mass:
  - `%sql select distinct Booster_Version, PAYLOAD_MASS__KG_ as "Maximum Payload Mass" from SPACEXTABLE \`
  - `where PAYLOAD_MASS__KG_ = (select max(PAYLOAD_MASS__KG_) from SPACEXTABLE);`
- List of booster versions that have carried the maximum payload mass.

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

```
%sql select distinct Booster_Version, PAYLOAD_MASS__KG_ as "Maximum Payload Mass" from SPACEXTABLE \
    where PAYLOAD_MASS__KG_ = (select max(PAYLOAD_MASS__KG_) from SPACEXTABLE);
```

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

Booster_Version	Maximum Payload Mass
F9 B5 B1048.4	15600
F9 B5 B1049.4	15600
F9 B5 B1051.3	15600
F9 B5 B1056.4	15600
F9 B5 B1048.5	15600
F9 B5 B1051.4	15600
F9 B5 B1049.5	15600
F9 B5 B1060.2	15600
F9 B5 B1058.3	15600
F9 B5 B1051.6	15600
F9 B5 B1060.3	15600
F9 B5 B1049.7	15600

2015 Launch Records

- List the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015:
  - `%sql select substr(Date,1,4) as "Year", substr(Date,6,2) as "Month", Landing_Outcome, Booster_Version, Launch_Site from SPACEXTABLE \`
  - `where substr(Date,1,4)='2015' and Landing_Outcome = "Failure (drone ship)";`
- List of failed landings to drone ship in 2015.

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.

Note: SQLLite does not support monthnames. So you need to use substr(Date, 6,2) as month to get the months and substr(Date,0,5)='2015' for year.

```
%sql select substr(Date,1,4) as "Year", substr(Date,6,2) as "Month", Landing_Outcome, Booster_Version, Launch_Site f
    where substr(Date,1,4)='2015' and Landing_Outcome = "Failure (drone ship)";
```

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

Year	Month	Landing_Outcome	Booster_Version	Launch_Site
2015	01	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
2015	04	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

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

- 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:
  - `%sql select Landing_Outcome, count(*) from SPACEXTABLE where Date between '2010-06-04' and '2017-03-20' group by Landing_Outcome order by count(*) desc;`
- Ranked list of landing outcomes for the period 2010-06-04 to 2017-03-20

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.

```
%sql select Landing_Outcome, count(*) from SPACEXTABLE where Date between '2010-06-04' and '2017-03-20' group by Landing_Ou
```

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

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

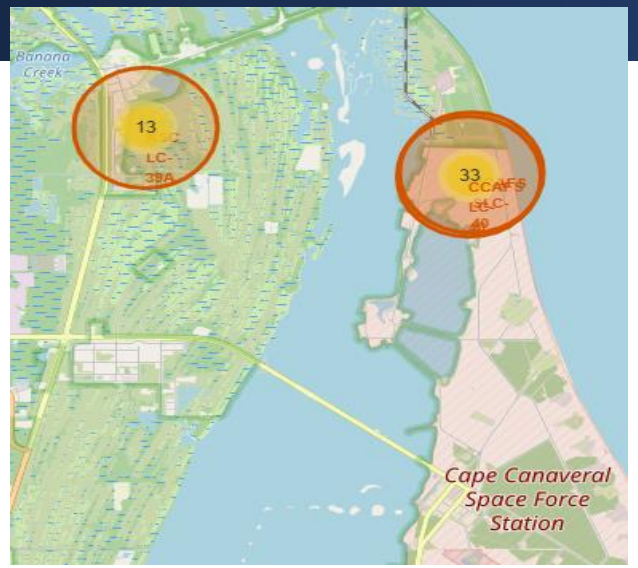
Section 3

Launch Sites Proximities Analysis

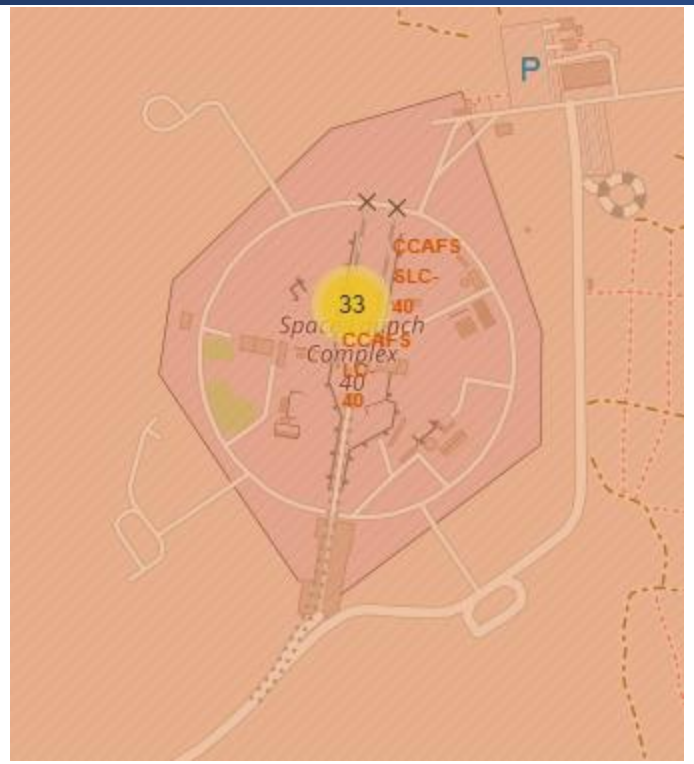
SpaceX Launch Sites Map

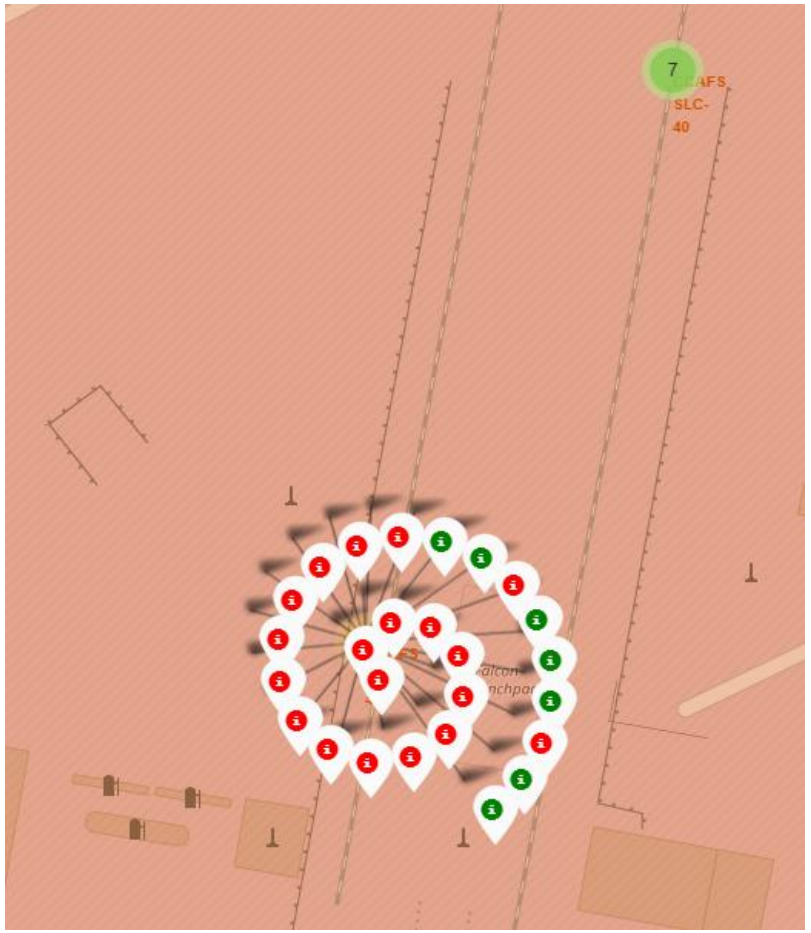
SpaceX has four (4) launch sites split between three (3) on the east coast and one (1) on the west coast. Of further interest, the east coast sites are so close to each other that the names appear blurred as they overlap on the map above. This may indicate that the geographical features/advantages of this location are not easily found elsewhere.

SpaceX Launch Outcomes by Site



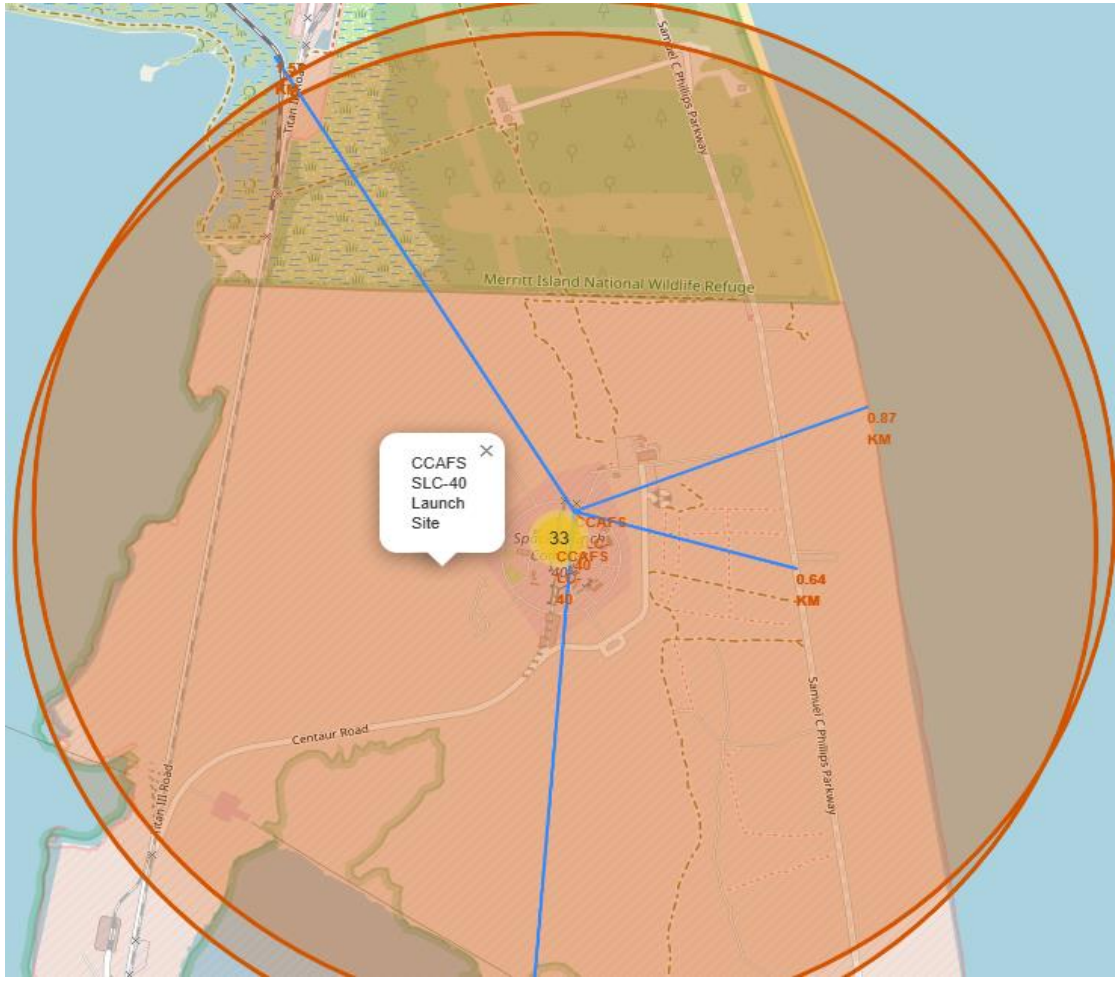
From the screenshots, the proximities of the east coast launch sites can be seen. The launch outcomes for the CCAFS LC-40 site are identified with red markers signifying failed outcomes vs green for successful.

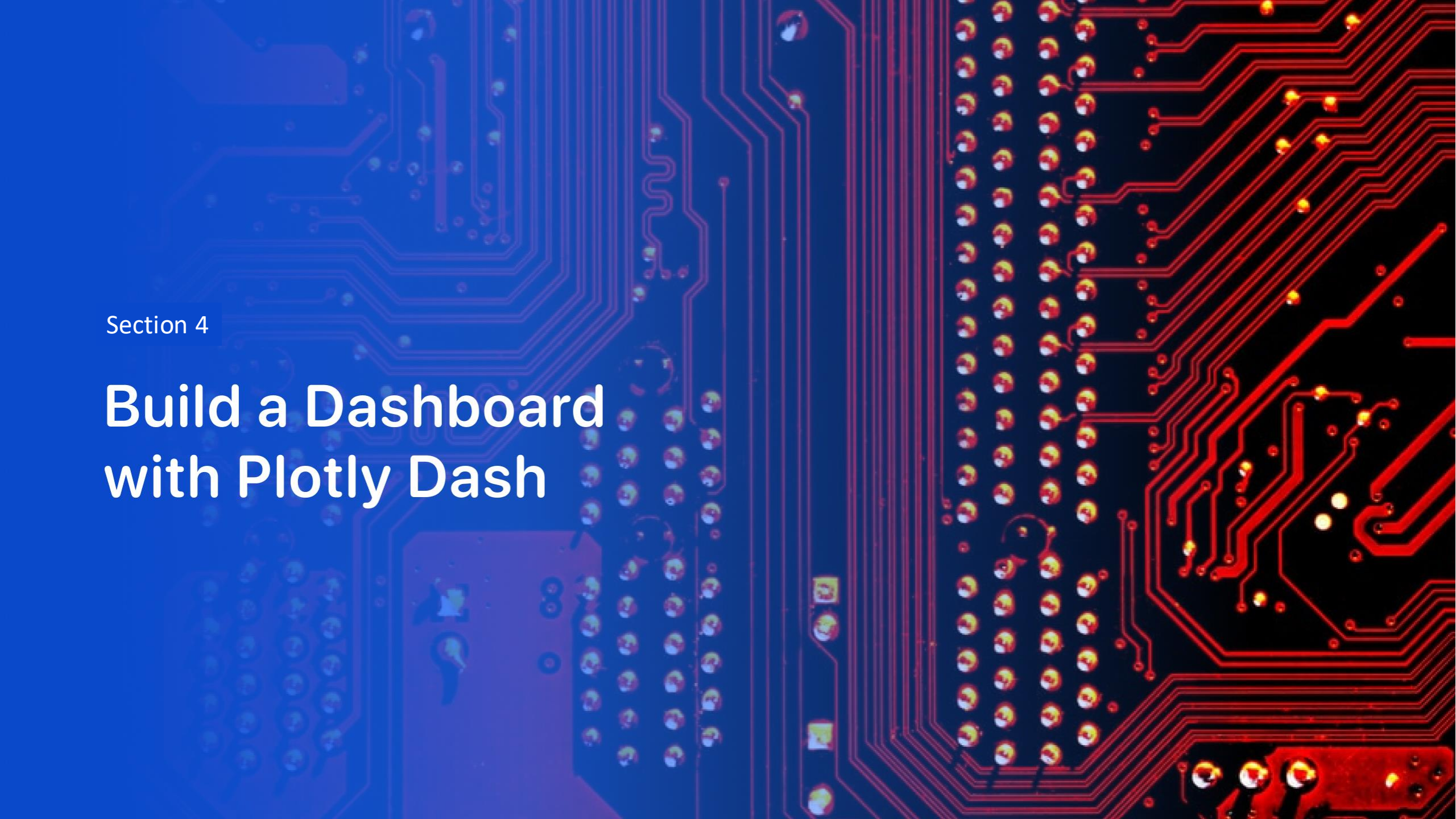




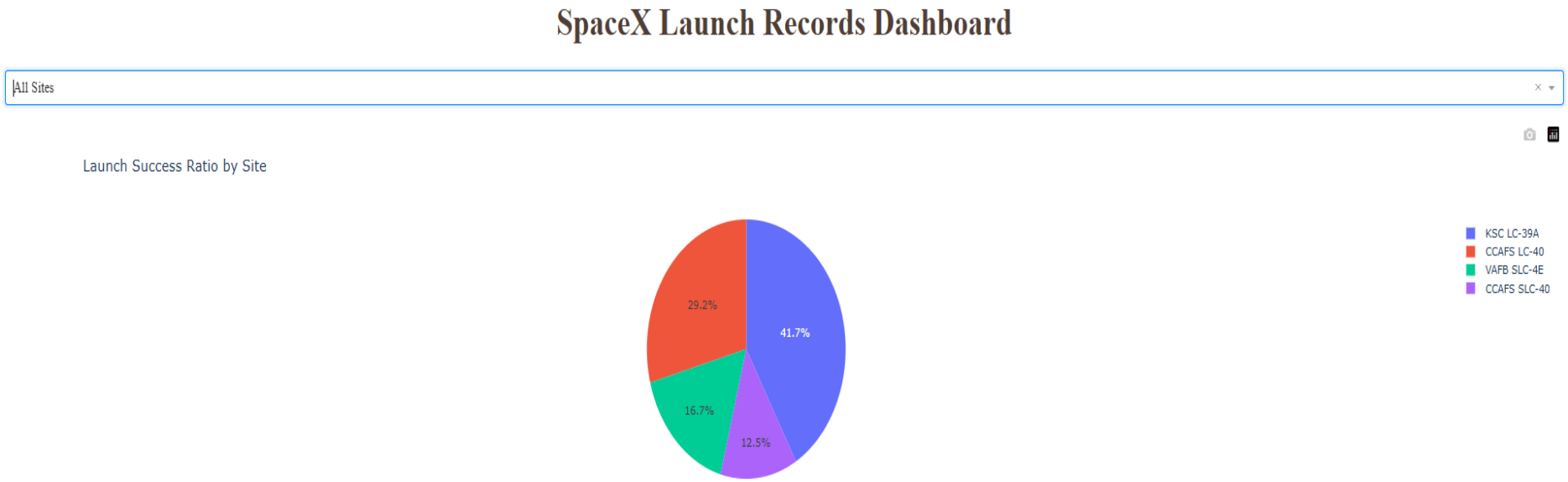
SpaceX Launch Site Environs

Comparable to the west coast site (VAFB) and the other east coast sites, CCAFS LC-40 is relatively close to transportation such as highways and/or railway lines as well as the coast. However, while within driving distance from a city, they are comparatively far from these highly inhabited areas.



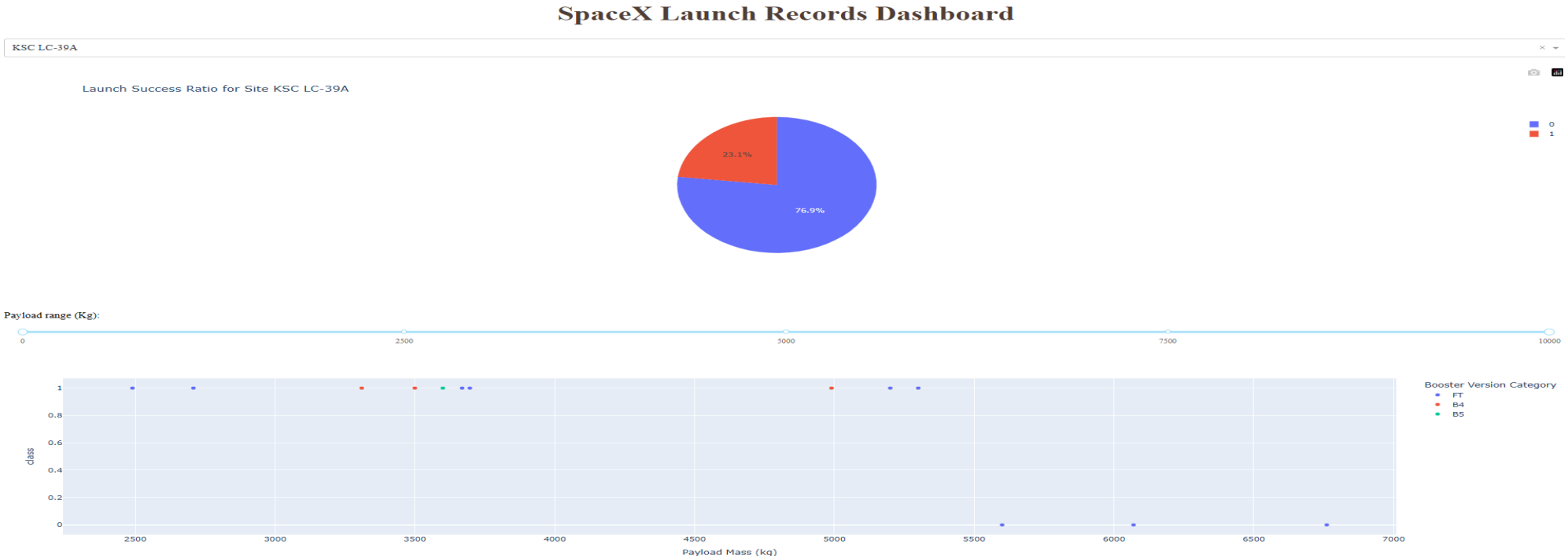

Section 4
Build a Dashboard with Plotly Dash

SpaceX Dashboard – All Sites



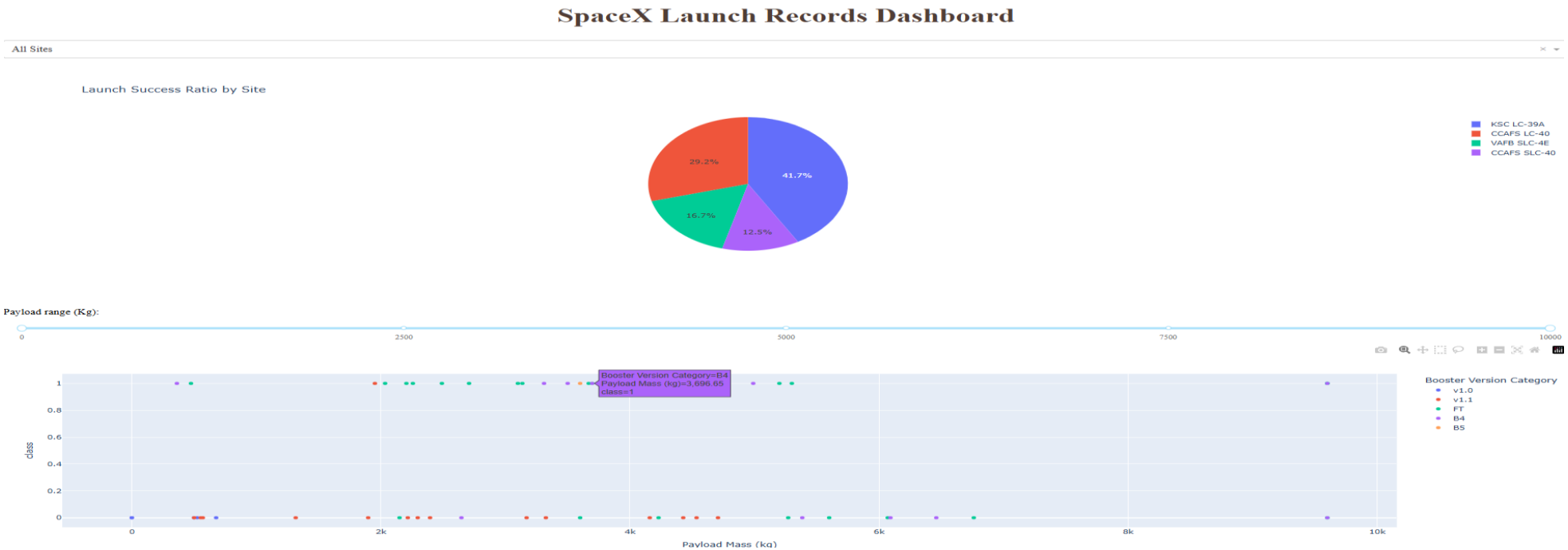
- The ratio of successful launches by site, highlighting the Kennedy Space Centre has the highest count followed by CCAFS LC-40.

KSC LC-39A



- KSC LC-39A charts showing the ratio of successful launches considering both the booster version as well as the payload mass.

Payload vs Launch Outcomes

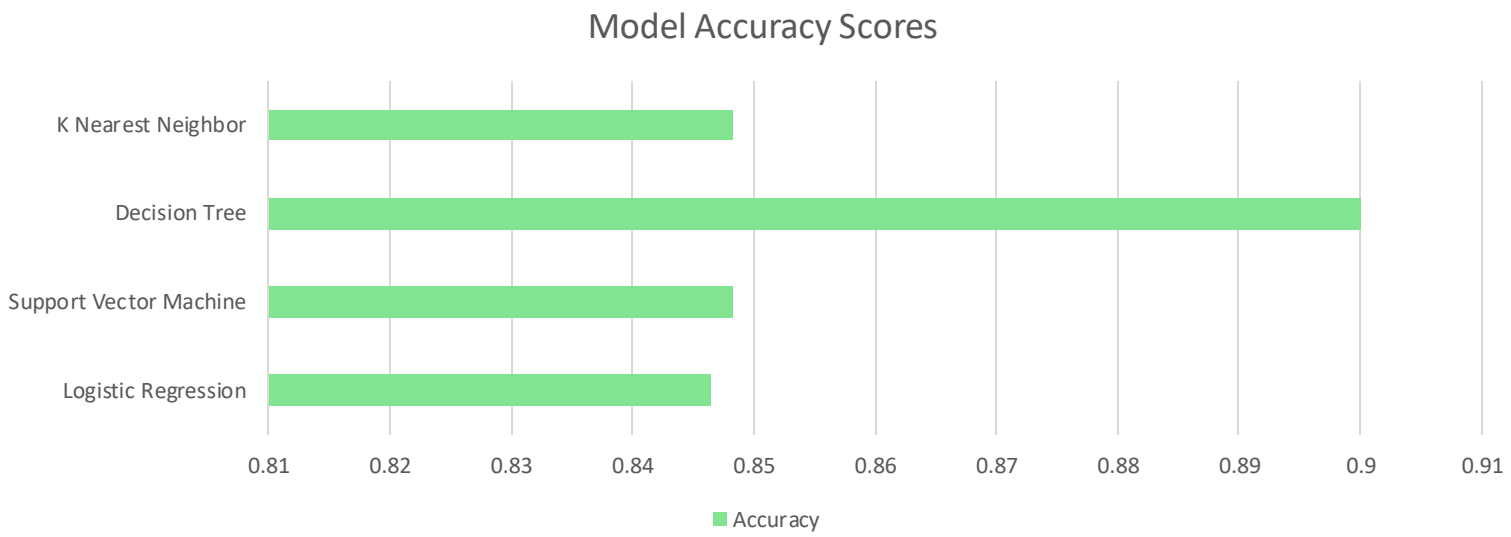


- Based on the scatter plot, the FT Booster appears to have the highest success ratio, regardless of payload. However it should be noted that most payloads do not exceed 7000 Kg.

Section 5

Predictive Analysis (Classification)

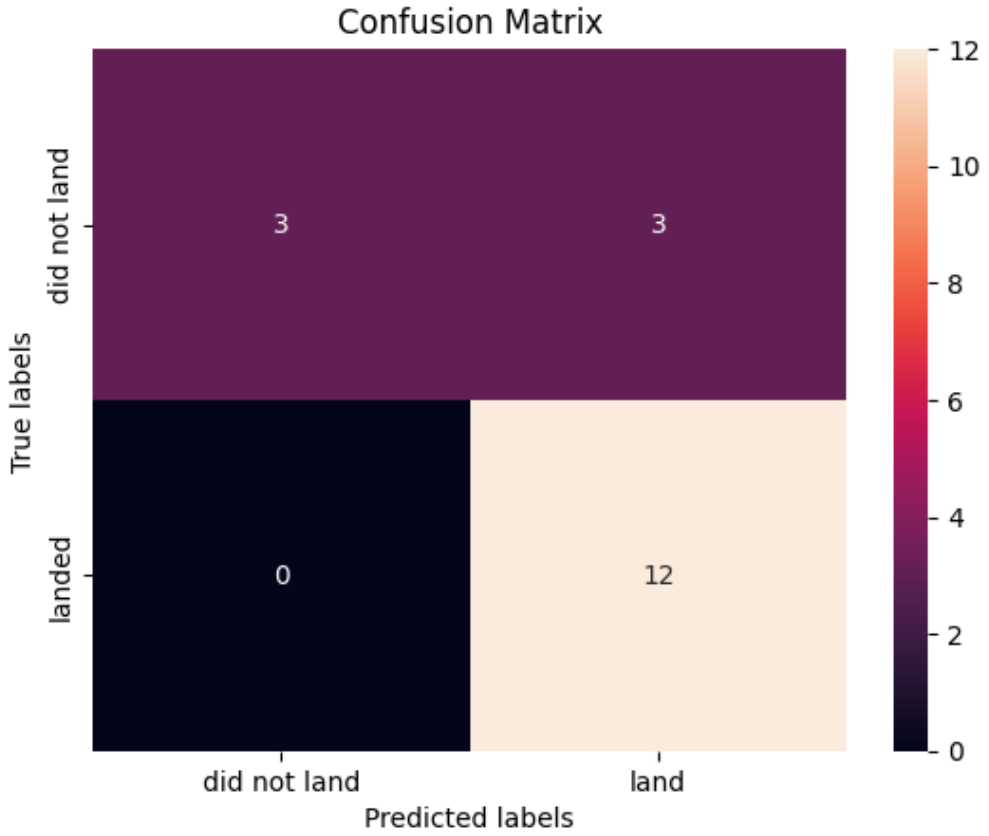
Classification Accuracy



- The Decision Tree Classification Model had the highest classification accuracy, based on the data used. Results will differ based on the sample size as well as changes in the feature engineering.

Confusion Matrix

- Examining the Decision Tree confusion matrix, we see that it can distinguish between the different classes. We see that the problem is false positives.
- Overview:
  - True Postive - 12 (True label is landed, Predicted label is also landed)
  - False Postive - 3 (True label is not landed, Predicted label is landed)



Conclusions

- Based on the accuracy scoring, the Decision Tree comes out as the best option, while there is no clear difference amongst K Nearest Neighbor, Support Vector or Logistic Regression
- The similarity in scores for the Decision Tree (which edges out the other models by a small margin), KNN, SVM and LR models may be due to the limited size of the data set.
- Despite the use of a very limited sample (18 records) and features, we are able to come up with a Classification Model to predict the success of first stage landings.
- While factors such as Payload Mass seem not to have any real impact, launch sites and orbit types appear to have an impact on the success of first stage landings.

Thank you!