IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Sree Vignesh V
21-07-2024

Outline

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

Executive Summary

- Obtained data from the SpaceX Wikipedia page and the public SpaceX API. A column called "class" was created to categorize successful landings. Used dashboards, folium maps, SQL, and visualization to explore data. Compiled relevant columns for the features. One hot encoding was used to convert all category data to binary. To determine the ideal parameters for machine learning models, standardized data was used along with GridSearchCV. Visualized accuracy score of all models.
- The following four machine learning models were created: K Nearest Neighbors, Decision Tree Classifier, Support Vector Machine, and Logistic Regression. All achieved comparable outcomes, with an accuracy percentage of 83.33%. Every model overestimated the number of successful landings. To improve the determination and accuracy of the models, more data is needed.

Introduction

- Project Background
    1. SpaceX is successful in the private space exploration sector due to their reusable Falcon 9 rockets, which significantly reduce costs
    2. However, the Falcon 9 rockets do not have a 100% reuse rate due to various factors like crashes, payload mass etc.
- Problem Statement
    1. As a data scientist working for SpaceY, The objective of the project is to predict whether the Falcon 9 rocket will successfully land or not.

ection 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Combined data from SpaceX public API and SpaceX Wikipedia page
- Perform data wrangling
  - Classifying true landings as successful and unsuccessful otherwise
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Tuned models using GridSearchCV

Data Collection

- Data collection process involved a combination of API requests from Space X public API and web scraping data from a table in Space X's Wikipedia entry.
- The next slide will show the flowchart of data collection from API and the one after will show the flowchart of data collection from webscraping.

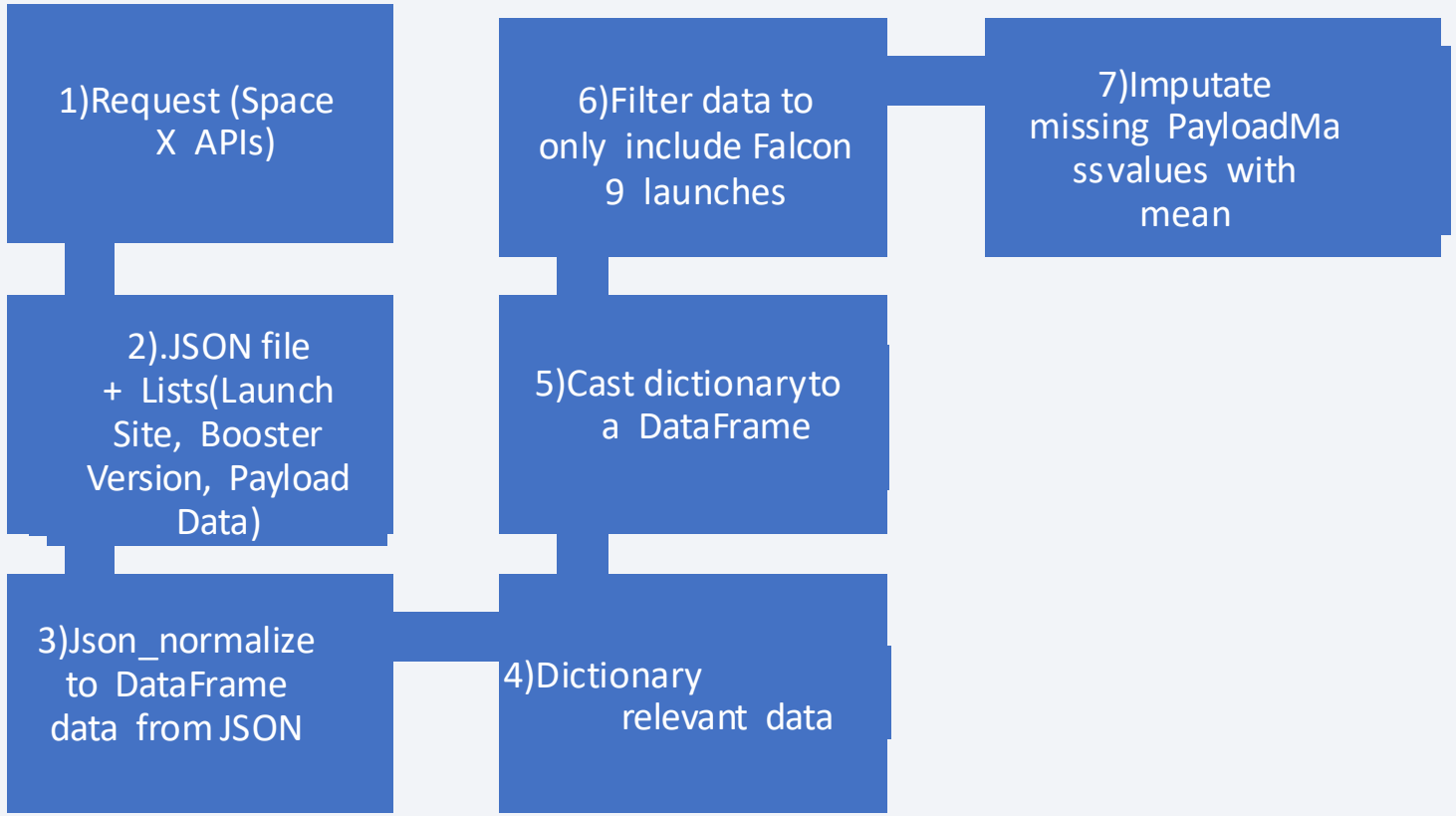
Space X API Data Columns:

- FlightNumber, Date, BoosterVersion, PayloadMass, Orbit, LaunchSite, Outcome, Flights, GridFins,
- Reused, Legs, LandingPad, Block, ReusedCount, Serial, Longitude, Latitude

Wikipedia Webscrape Data Columns:

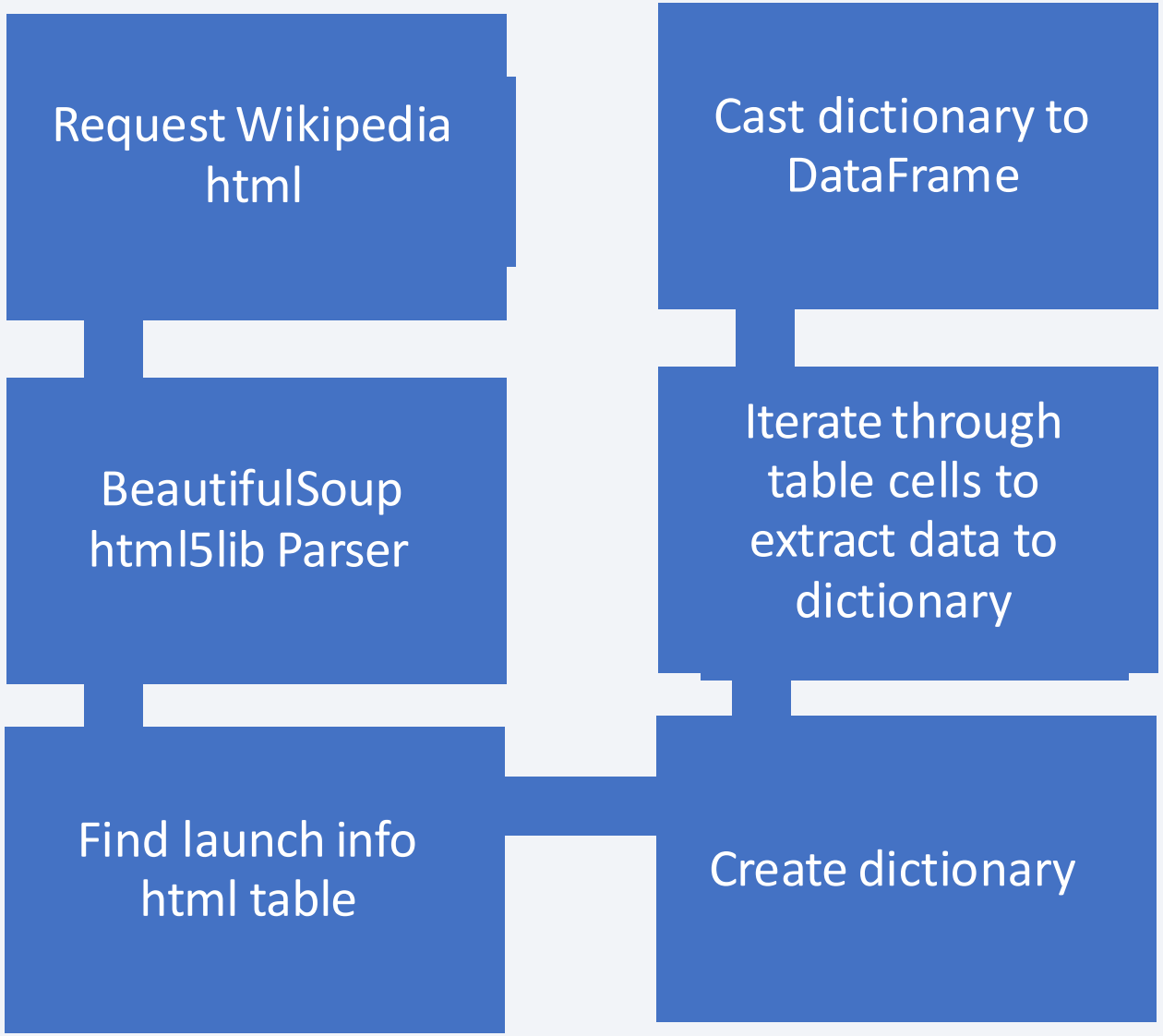
- Flight No., Launch site, Payload, PayloadMass, Orbit, Customer, Launch outcome, Version Booster, Booster landing, Date, Time

Data Collection – SpaceX API



Github link - https://github.com/Sree-Vignesh-V/IBM_Data_science_Capstone/blob/main/data_collection_spacex_api.ipynb

Data Collection - Scraping



Github link - https://github.com/Sree-Vignesh-V/IBM_Data_science_Capstone/blob/main/data_collection_spacex_web_scrapping.ipynb

Data Wrangling

- The objective of Data Wrangling is to segregate successful and failed launches; failure = 0 and successful = 1.
- There are two parts to the outcome column: "Mission Outcome" and "Landing Location"
- A new column called "class" will have a value of 1 if the "Mission Outcome" is true and 0 otherwise.
- Mapping Values:
  - Set True Ocean, True RTLS, and True ASDS to 1.
  - False Ocean, False RTLS, None None, False ASDS, and None ASDS - set to -> 0

Github link - https://github.com/Sree-Vignesh-V/IBM_Data_science_Capstone/blob/main/data_wrangling_spacex.ipynb

EDA with Data Visualization

- Exploratory Data Analysis performed on variables Flight Number, Payload Mass, Launch Site, Orbit, Class and Year.
- Plots Used:
    - Flight Number vs. Payload Mass, Flight Number vs. Launch Site, Payload Mass vs. Launch Site, Orbit vs. Success Rate, Flight Number vs. Orbit, Payload vs Orbit, and Success Yearly Trend
    - Scatter plots, line charts, and bar plots were used to compare relationships between variables to
- EDA performed to identify any correlations/relationships between any of the features to the class column

Github link - https://github.com/Sree-Vignesh-V/IBM_Data_science_Capstone/blob/main/spacex_eda_visualization.ipynb

EDA with SQL

- Loaded data set into IBM DB2 Database.
- Queried using SQL Python integration.
- Queries were made to get a better understanding of the dataset.
- Queried information about launch site names, mission outcomes, various pay load sizes of customers and booster versions, and landing outcomes

Github link - https://github.com/Sree-Vignesh-V/IBM_Data_science_Capstone/blob/main/spacex_sql_eda.ipynb

Build an Interactive Map with Folium

- Folium is used to plot maps which mark Launch Sites, successful and unsuccessful landings, and a proximity example to key locations: Railway, Highway, Coast, and City.
- This allows us to understand why launch sites may be located where they are. Also visualizes successful landings relative to location.

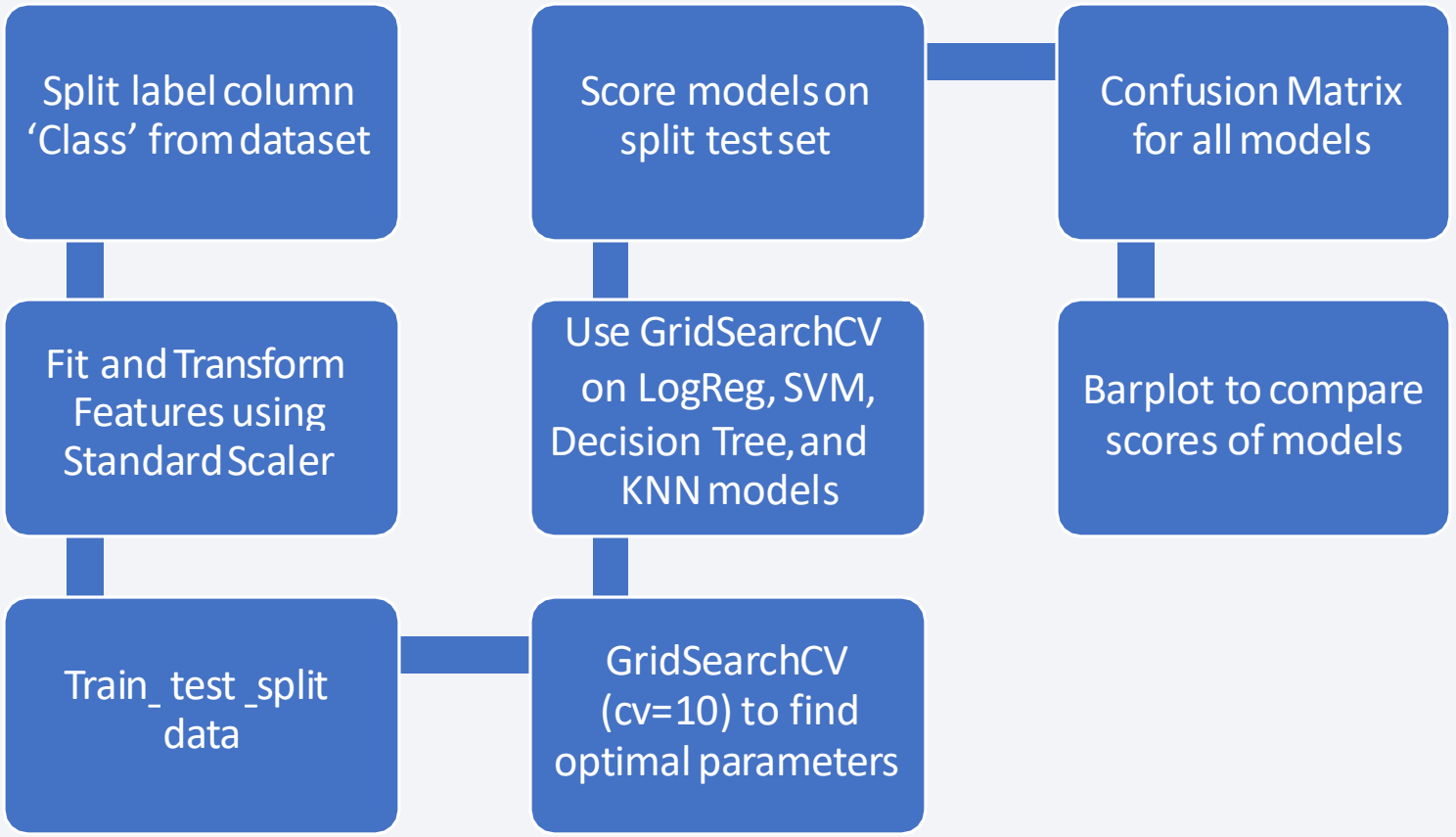
Github link - https://github.com/Sree-Vignesh-V/IBM_Data_science_Capstone/blob/main/spacex_visualization_folium.ipynb

Build a Dashboard with Plotly Dash

- Dashboard includes a pie chart and a scatter plot.
- Pie chart shows the distribution of successful landings across all launch sites and can be selected to show individual launch site success rates.
- Scatter plot takes two inputs: All sites or individual site and payload mass on a slider between 0 and 10000 kg.
- The pie chart is used to visualize launch site success rate.
- The scatter plot can help us see how success varies across launch sites, payload mass, and booster version category.

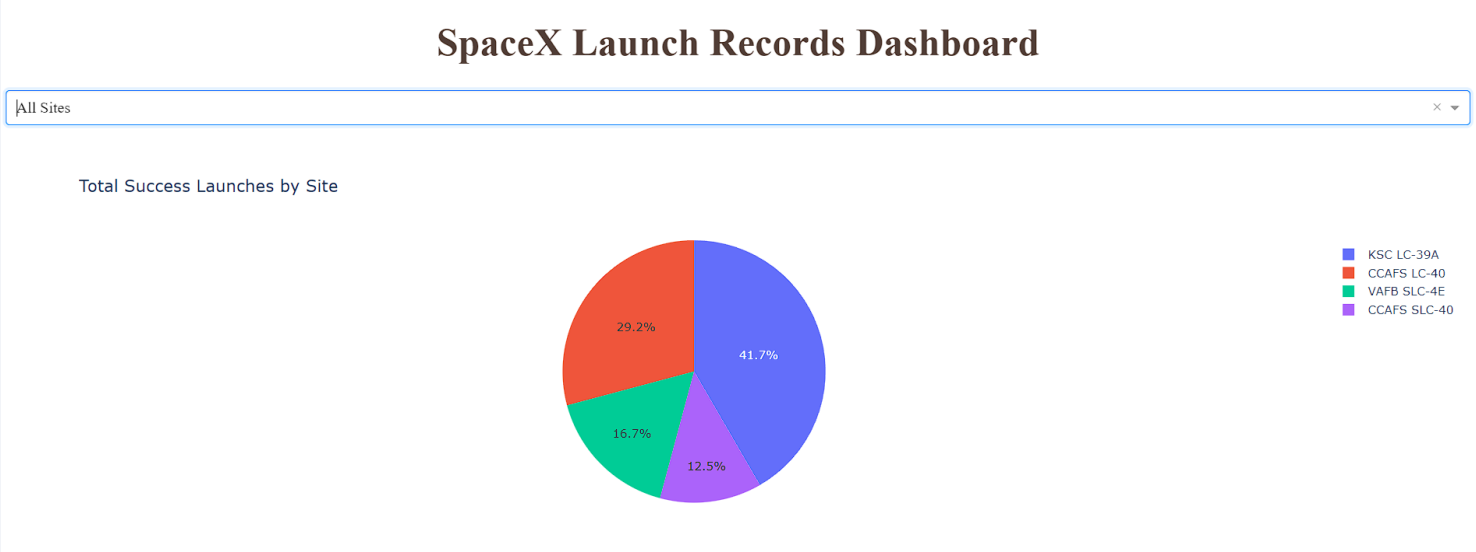
Github link - https://github.com/Sree-Vignesh-V/IBM_Data_science_Capstone/blob/main/spacex_dash_app.py

Predictive Analysis (Classification)

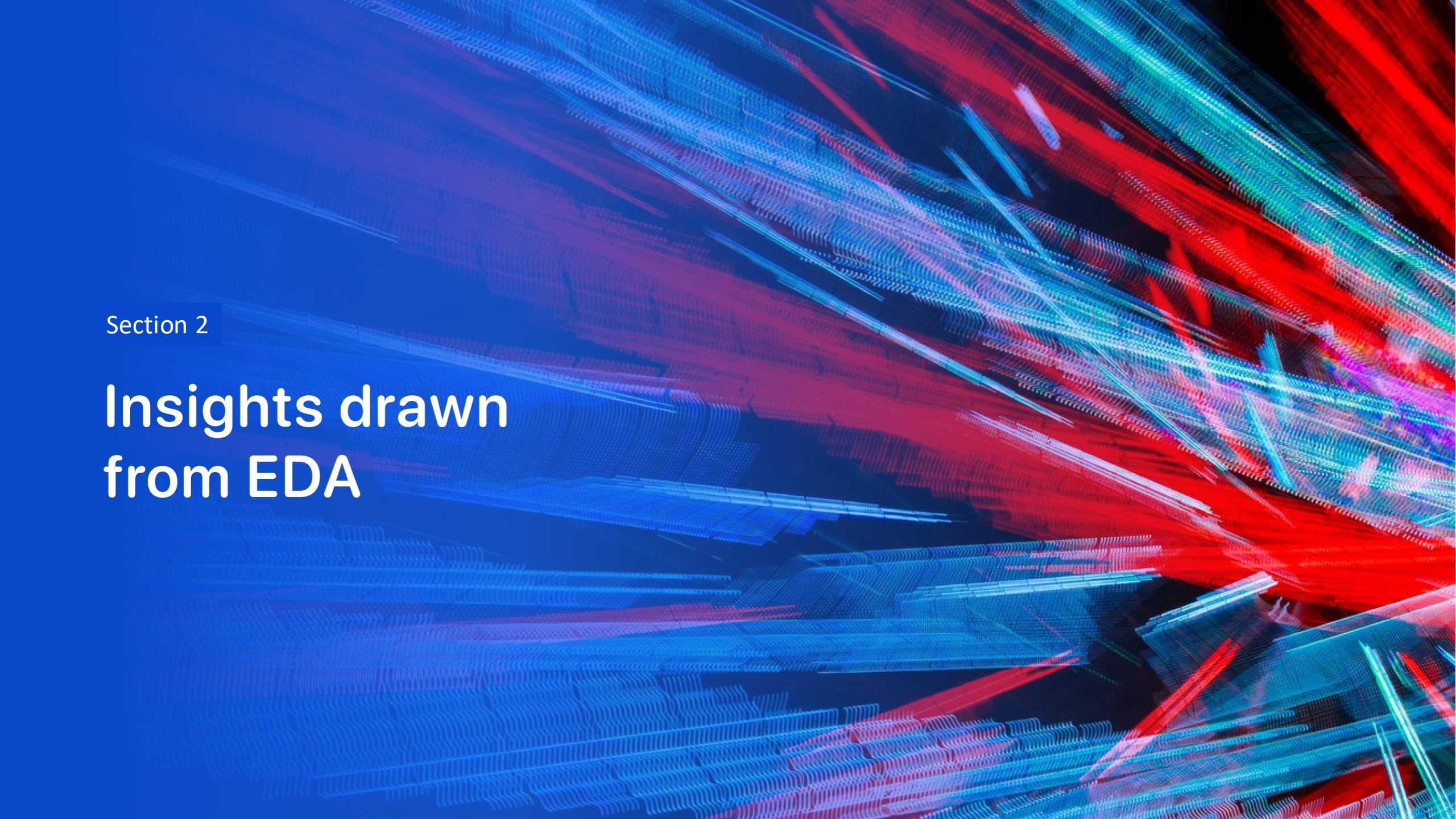


Github link - https://github.com/Sree-Vignesh-V/IBM_Data_science_Capstone/blob/main/SpaceX_Machine%20Learning%20Prediction.ipynb

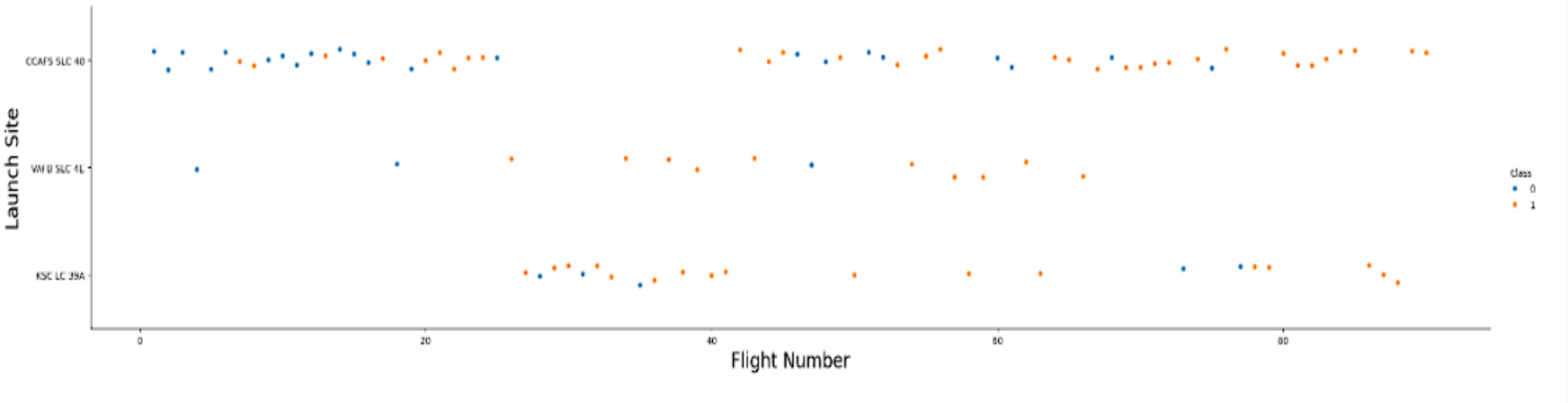
Results



- EDA revealed that there is a correlation between the orbits and the rocket's success rate
- It also revealed that the success rates increased steadily after 2013
- A pie chart showing the success rates of each launch sites reveals that KSC LC 39-A has the highest success rate
- All the trained models have an accuracy of 83%

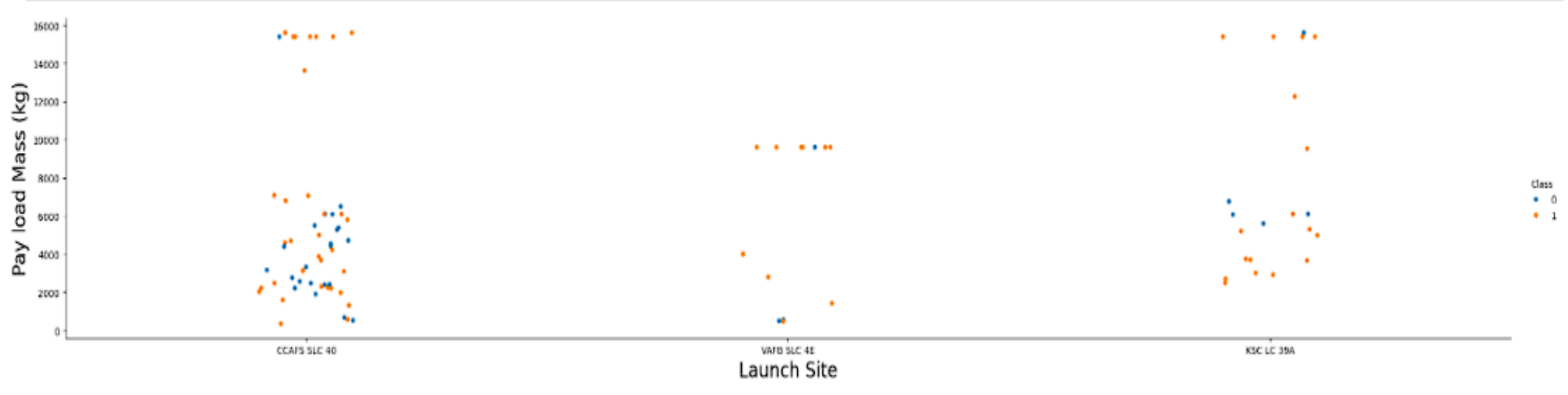

Section 2
Insights drawn from EDA

Flight Number vs. Launch Site



The graph indicates a rising success rate over time (shown by flight number). Probably a major discovery around flight 20, which greatly raised the success rate. Given that it has the highest volume, CCAFS seems to be the primary launch point.

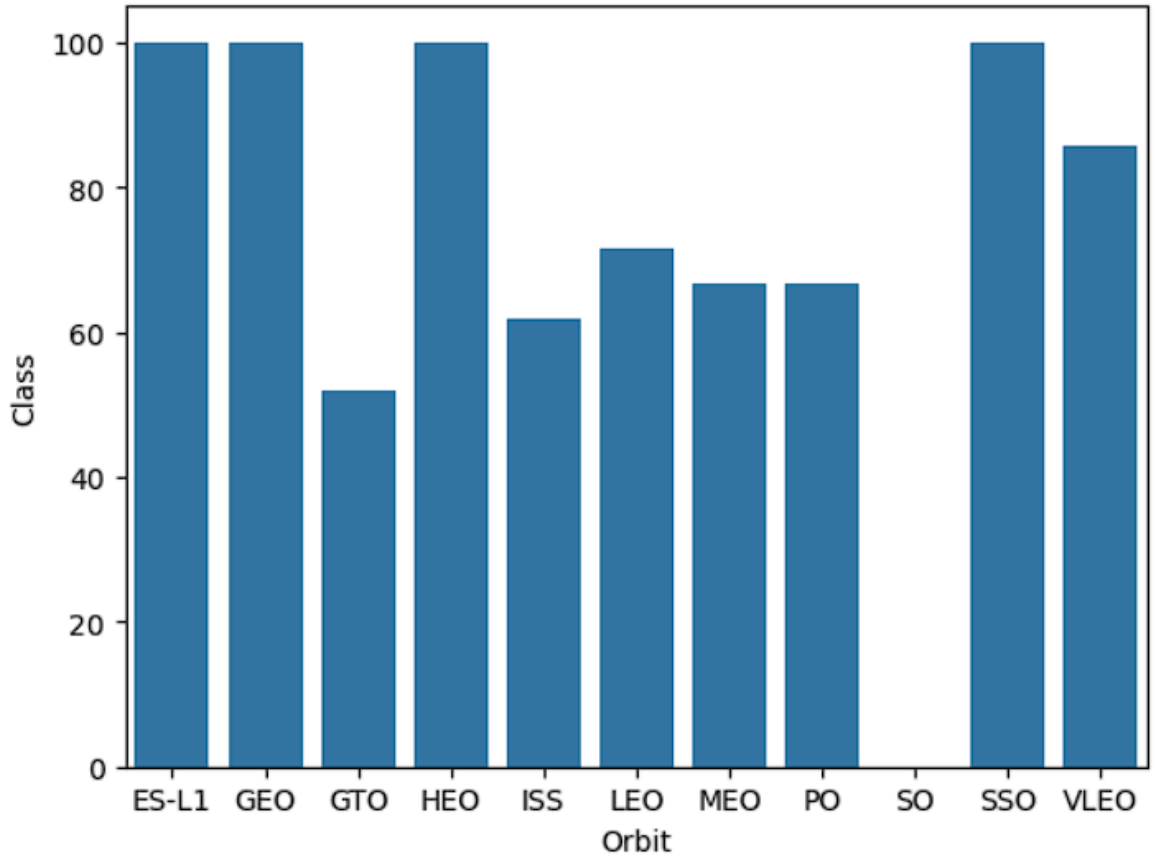
Payload vs. Launch Site



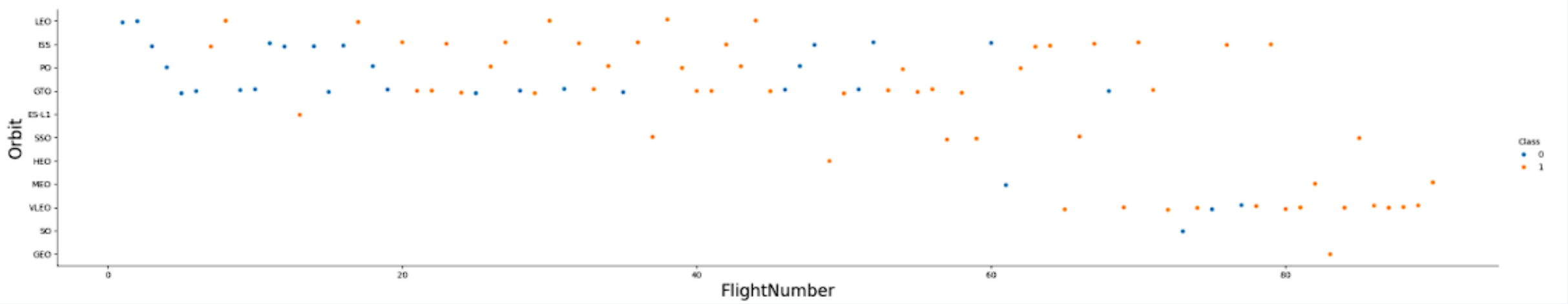
Payload mass appears to fall mostly between 0-6000 kg. Different launch sites also seem to use different payload mass.

Success Rate vs. Orbit Type

- S-L1 , GEO , HEO and SSO has 100% success rate
- SO has 0% success rate
- GTO has the around 50% success rate but largest sample

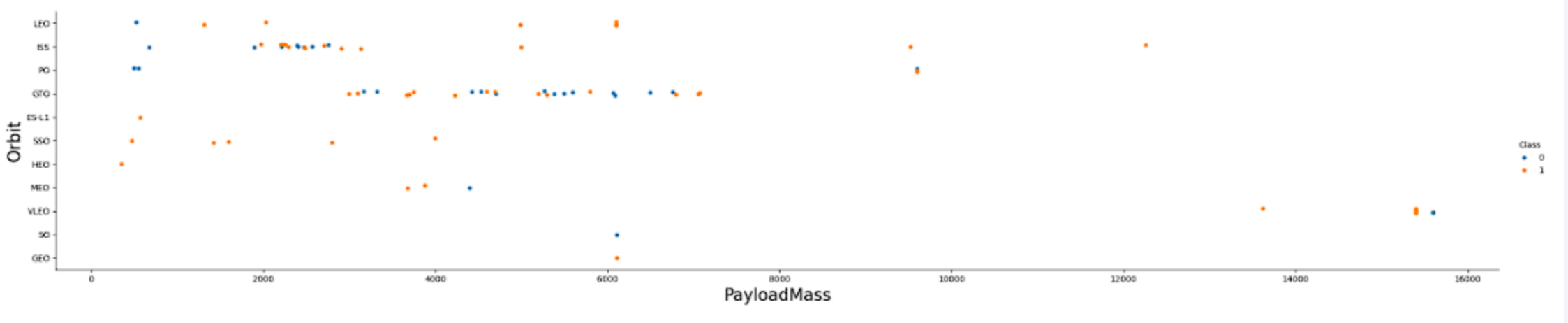


Flight Number vs. Orbit Type



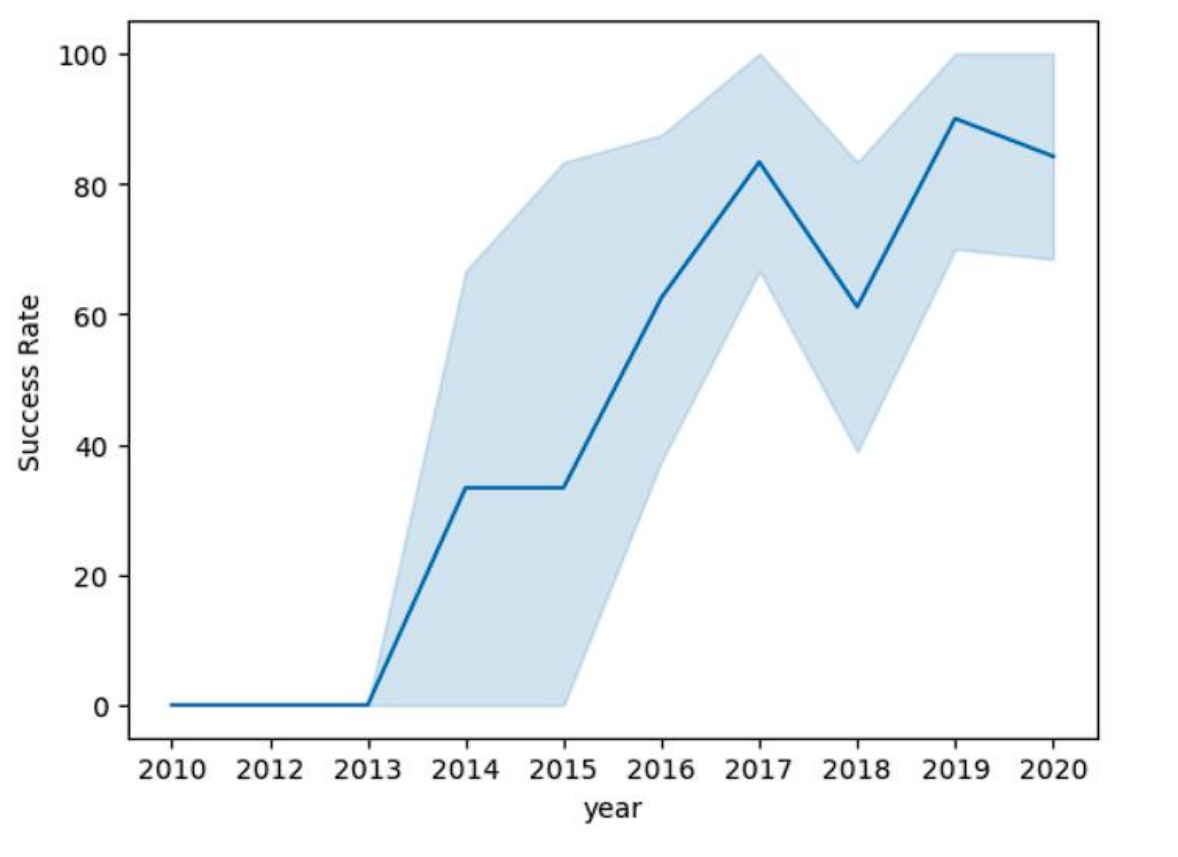
- Preferences for Launch Orbit were adjusted across Flight Number. This preference appears to be related to the Launch Outcome.

Payload vs. Orbit Type



- Payload mass seems to correlate with orbit
- LEO and SSO seem to have relatively low payload mass
- The other most successful orbit VLEO only has payload mass values in the higher end of the ranges

Launch Success Yearly Trend



Success rates has been steadily increasing from 2013 with a slight dip in 2018

All Launch Site Names

In [10]:
```
%sql select DISTINCT LAUNCH_SITE from SPACEXTABLE
```

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

Out[10]:

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

The query returns 4 Unique Launch sites

Launch Site Names Begin with 'CCA'

```
In [11]: %sql select * from SPACEXTABLE where launch_site like 'CCA%' limit 5
 * sqlite:///my_data1.db
Done.
```

Out[11]:

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

This query returns the first 5 entries in the database where the launch site's name begins with CCA

Total Payload Mass

```
In [12]: %sql select sum(payload_mass__kg_) as sum from SPACEXTABLE where customer like 'NASA (CRS)'
 * sqlite:///my_data1.db
Done.
Out[12]:
```

sum
45596

This query returns the total sum of all payloads which has been taken for NASA

Average Payload Mass by F9 v1.1

```
In [13]: %sql select avg(payload_mass__kg_) as Average from SPACEXTABLE where booster_version like 'F9 v1.1%'
 * sqlite:///my_data1.db
Done.
Out[13]:
```

Average
2534.6666666666665

This query returns the average payload mass carried by Booster version F9 v1.1

First Successful Ground Landing Date

```
In [14]: %sql select min(date) as Date from SPACEXTABLE where mission_outcome like 'Success'
 * sqlite:///my_data1.db
Done.
```

Out[14]:

Date
2010-06-04

This query returns the date when the first successful mission was launched

Successful Drone Ship Landing with Payload between 4000 and 6000

In [15]:
```
%sql select booster_version from SPACEXTABLE where (mission_outcome like 'Success') AND (payload_mass__kg_ BETWEEN 4000 AND
```

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

Out[15]:

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

This query returns all the booster versions which have successfully launched payloads ranging from 4000 kg to 6000 kg

Total Number of Successful and Failure Mission Outcomes

In [16]:
```
%sql SELECT mission_outcome, count(*) as Count FROM SPACEXTABLE GROUP by mission_outcome ORDER BY mission_outcome
```

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

Out[16]:

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

This query returns the breakdown of all mission outcomes with their counts

Boosters Carried Maximum Payload

In [17]:
```
maxm = %sql select max(payload_mass__kg_) from SPACEXTABLE
maxv = maxm[0][0]
%sql select booster_version from SPACEXTABLE where payload_mass__kg_=(select max(payload_mass__kg_) from SPACEXTABLE)
```

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

Out[17]:

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

This query returns all the booster names which has carried the maximum payload

2015 Launch Records

In [18]:
```
%sql select SUBSTR(DATE,6,2) as Month, landing_outcome, booster_version, launch_site from SPACEXTABLE where DATE like '2015%
```

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

Out[18]:

Month	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

This query returns all the lauches that happened in 2015

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

```
In [19]: %sql select landing_outcome, count(*) as count from SPACEXTABLE where Date >= '2010-06-04' AND Date <= '2017-03-20' GROUP by
 * sqlite:///my_data1.db
Done.
```

Out[19]:

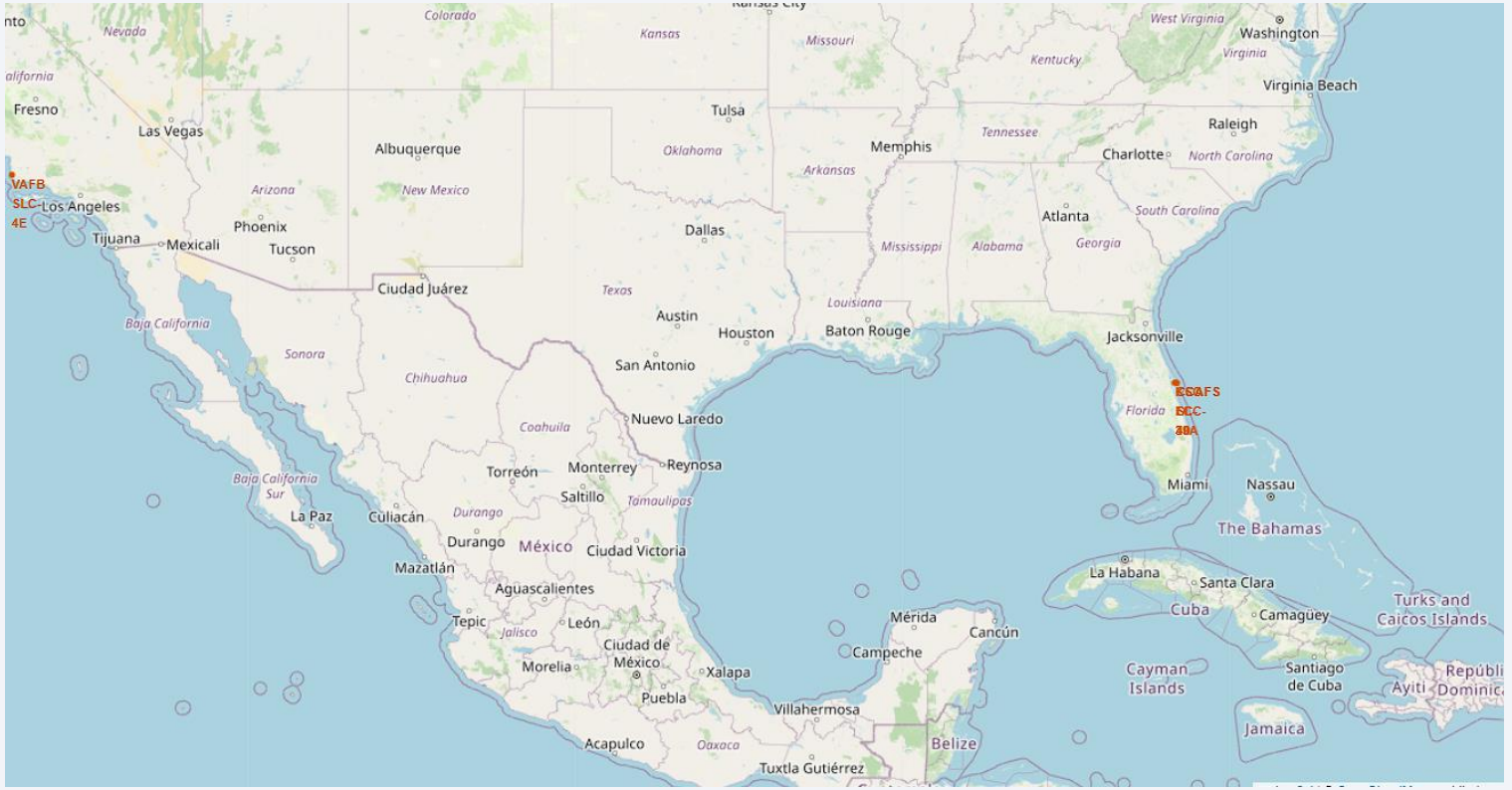
Landing_Outcome	count
No attempt	10
Success (drone ship)	5
Failure (drone ship)	5
Success (ground pad)	3
Controlled (ocean)	3
Uncontrolled (ocean)	2
Failure (parachute)	2
Precluded (drone ship)	1

This query returns 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

Section 3

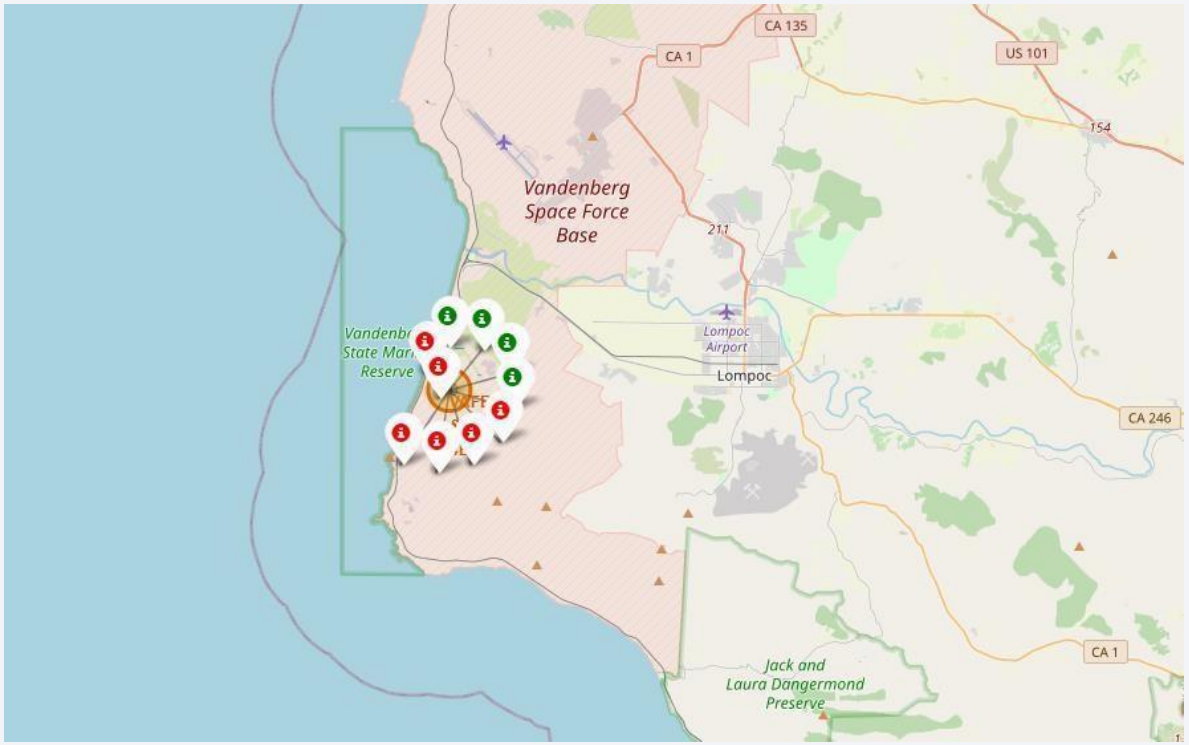
Launch Sites Proximities Analysis

Launch site Locations



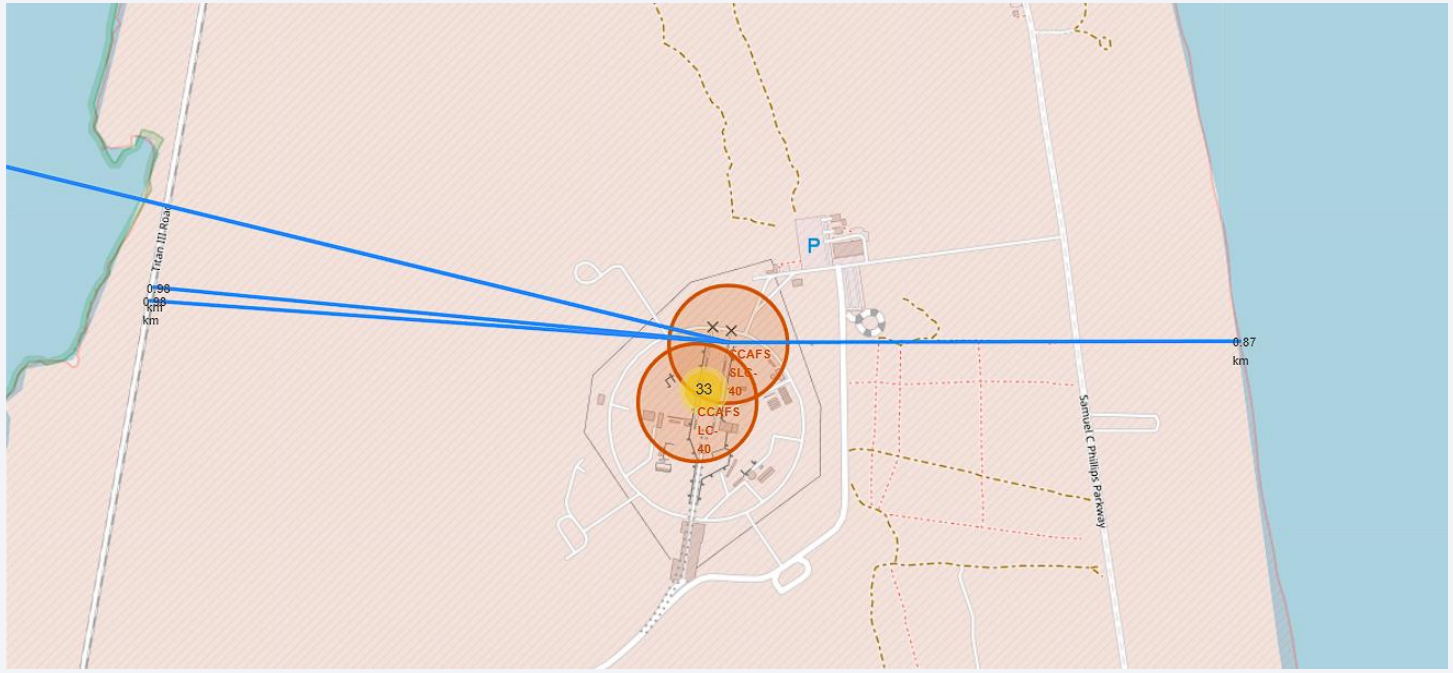
This map marks the locations of all launch sites.

Coloured Launch Site Markers

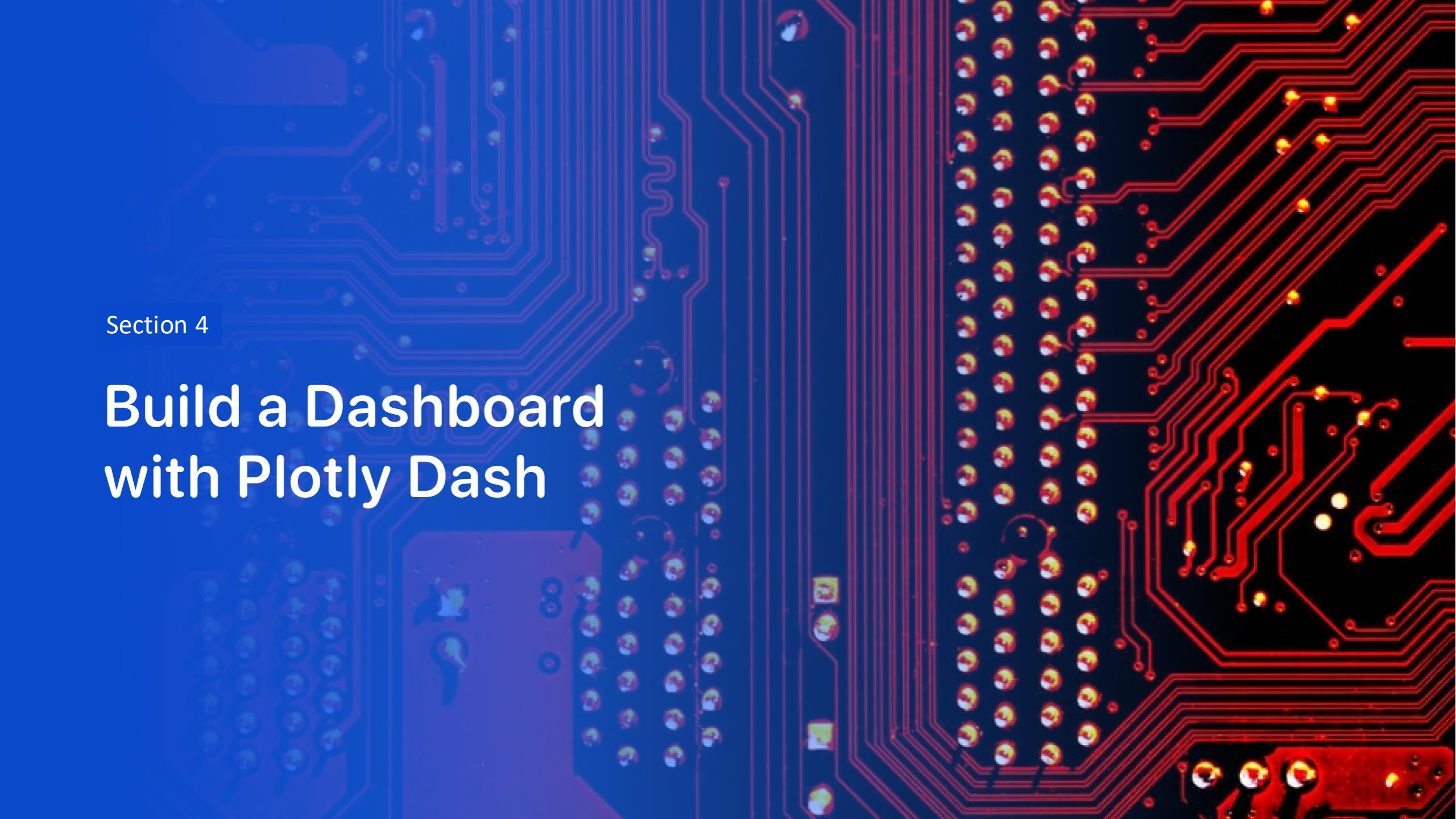


- The red coloured marking represents failed launches while the green markers represent successful launches

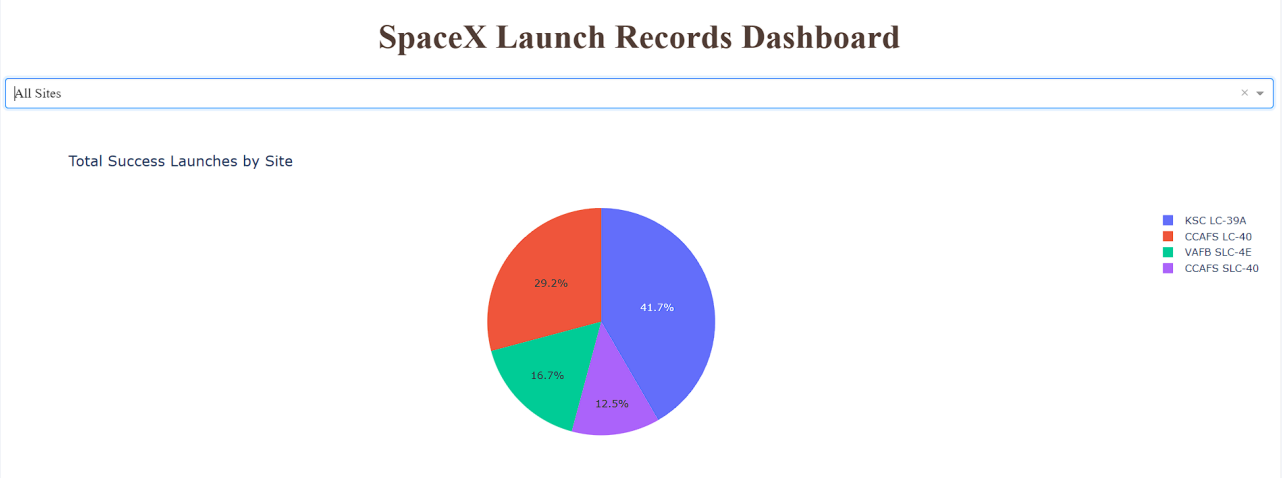
Distance from Launch site to landmarks



This map represents the distance from the launch site to various landmarks like coastline, railway line, city and highway

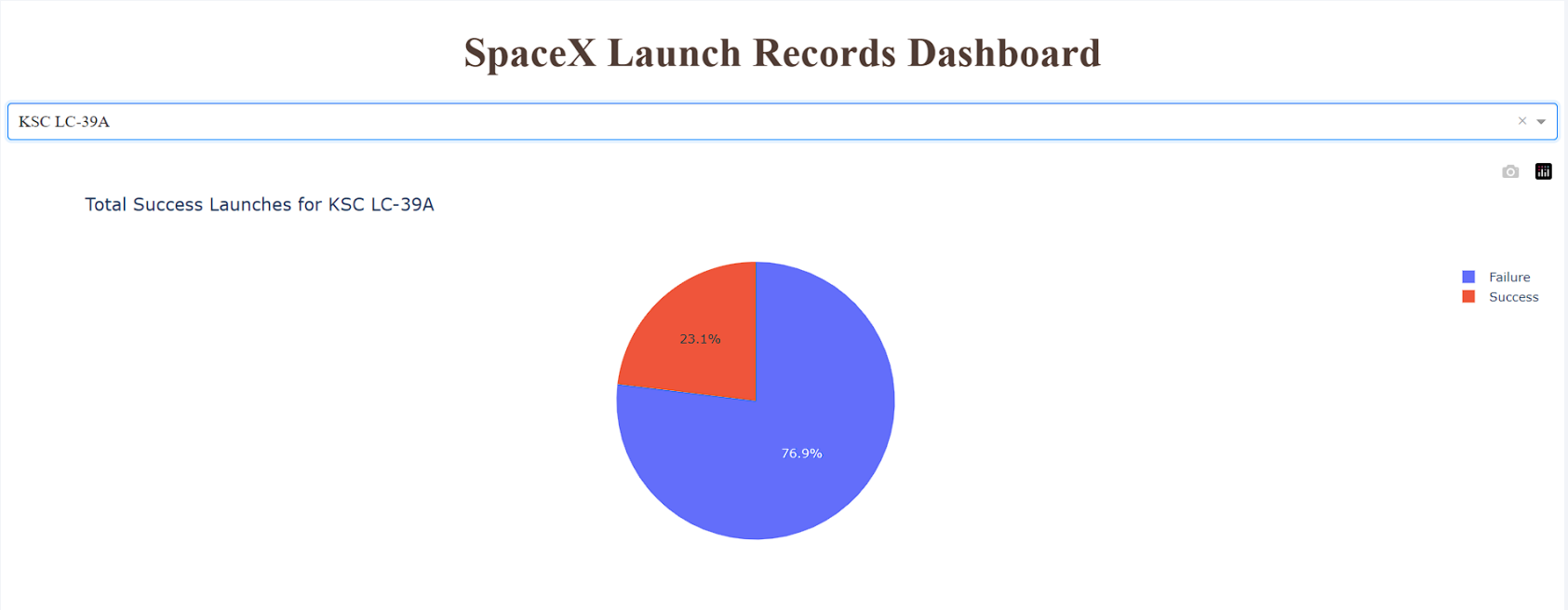

Section 4
Build a Dashboard with Plotly Dash

Launch Site success rates for all launch sites



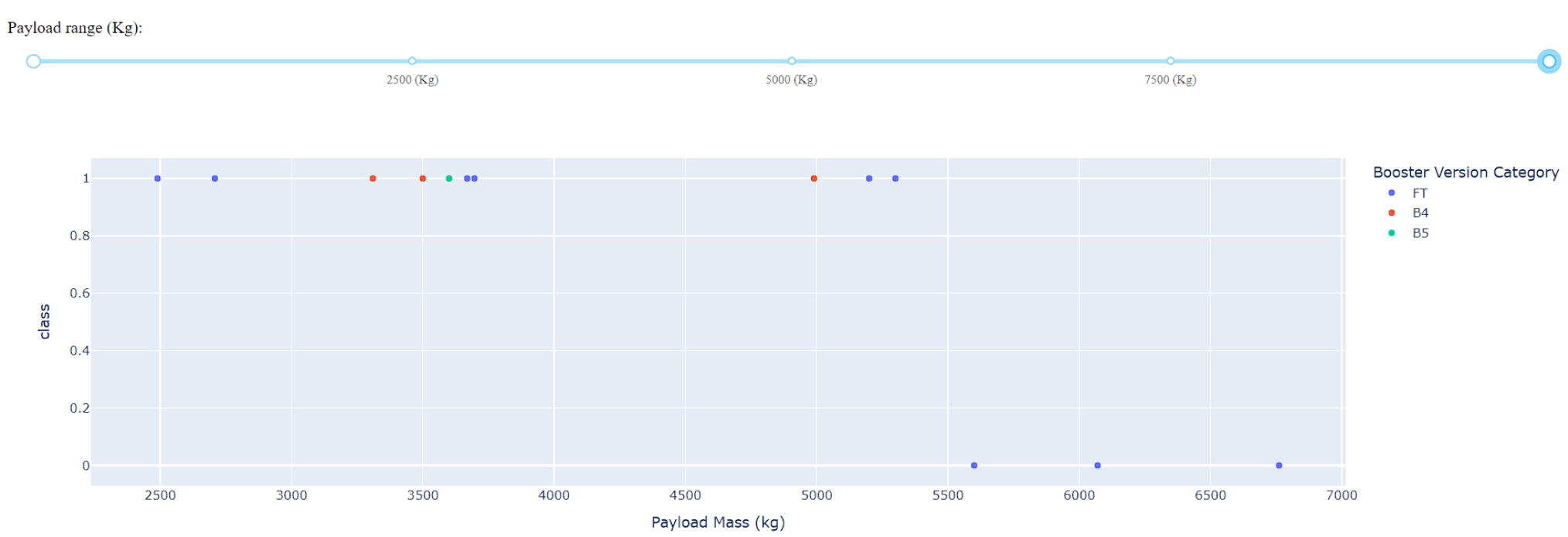
- This pie chart shows the percentage of all successful landings across all launch sites
- Launch site KSC LC 39-A has the highest success rate

Launch Site success vs fail rate for KSC LC 39-A



This pie chart shows the total success vs failure rate for the most successful launch site, KSC LC 39-A

Payload mass vs Success vs Booster

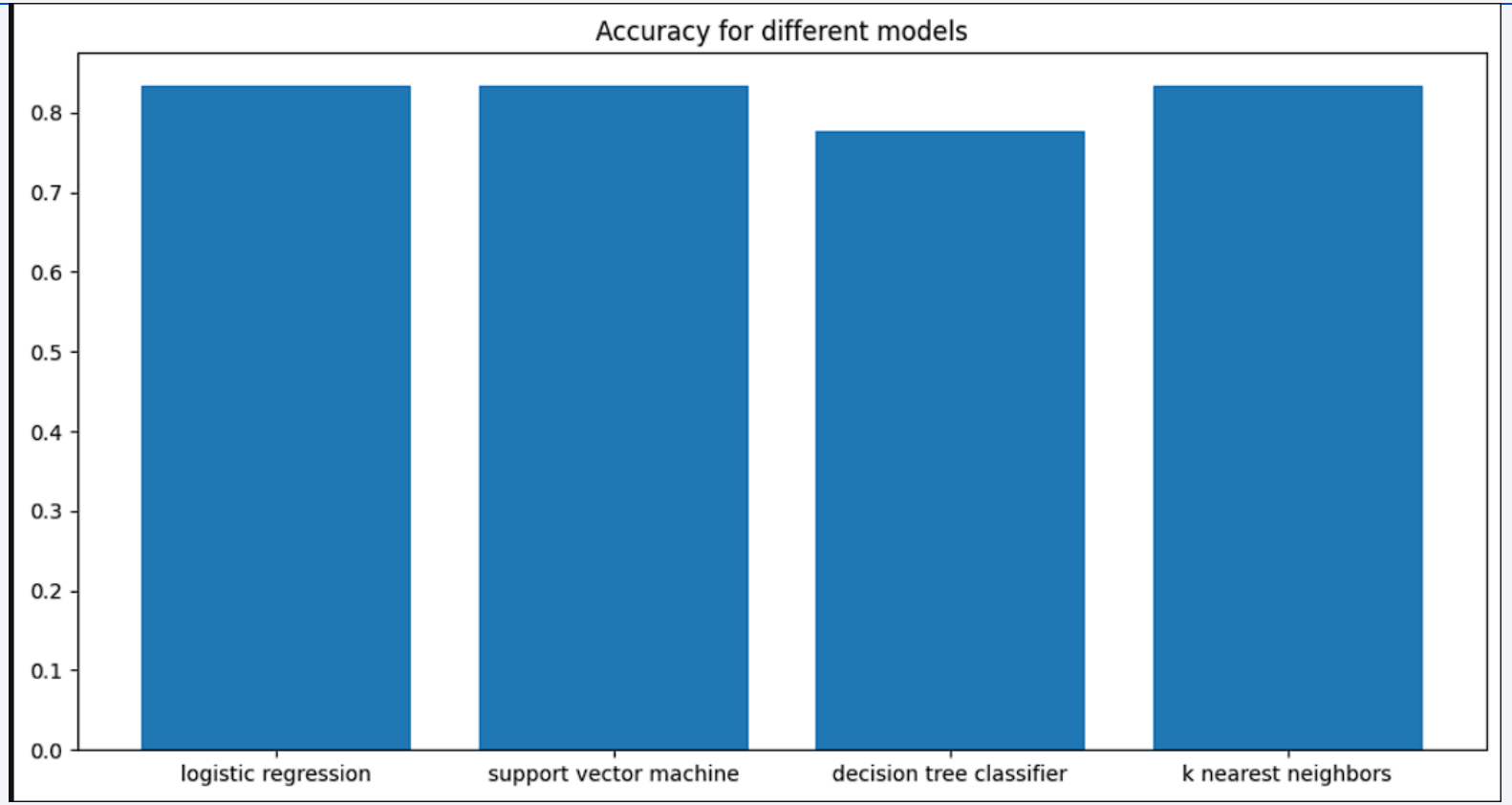


This scatter plot shows the success vs failure for various payload masses, categorized for each booster version

Section 5

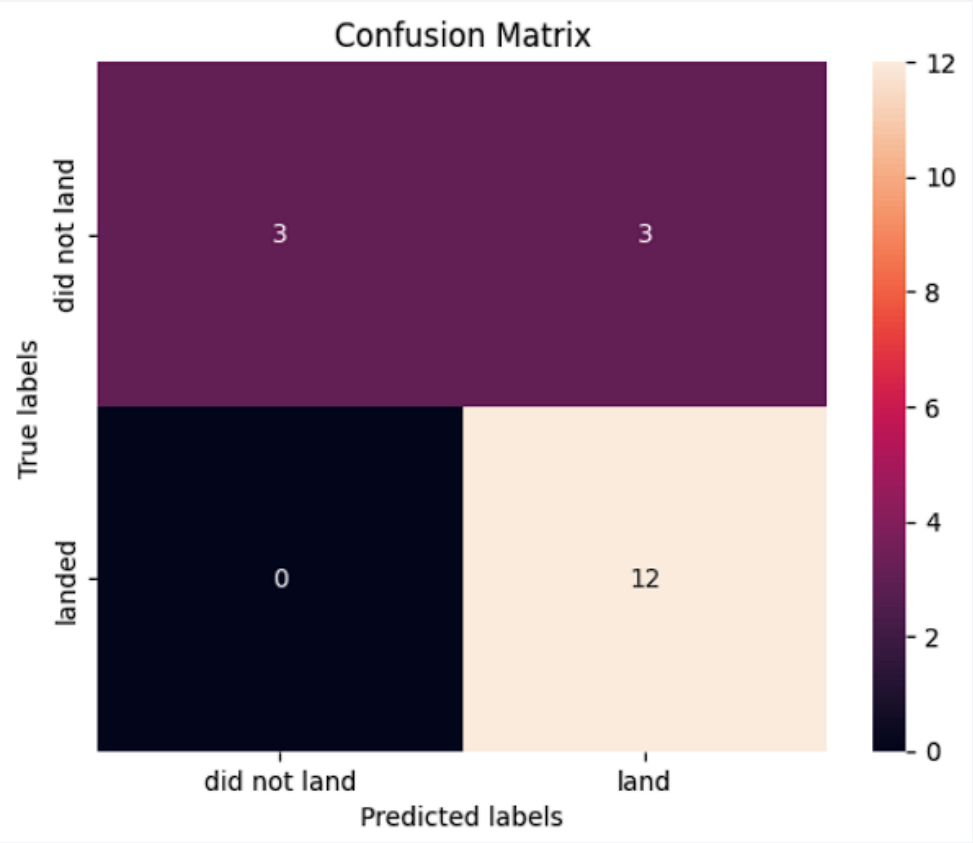
Predictive Analysis (Classification)

Classification Accuracy



- Except Decision tree, all other models have an accuracy of 83.3%
- Decision Tree has an accuracy of 77.7%

Confusion Matrix of SVM



- All 3 models (Logistic regression, SVM and K nearest neighbours) perform similarly and had a similar confusion matrix
- The confusion matrix shows 3 false positives and no false negatives

Conclusions

- Our task: to develop a machine learning model for Space Y who wants to bid against SpaceX
- The goal of model is to predict when Stage 1 will successfully land to save ~$100 million USD
- Used data from a public SpaceX API and web scraping SpaceX Wikipedia page
- Created data labels and stored data into a DB2 SQL database
- Created a dashboard for visualization
- We created a machine learning model with an accuracy of 83%
- Allon Mask of SpaceY can use this model to predict with relatively high accuracy whether a launch will have a successful Stage 1 landing before launch to determine whether the launch should be made or not
- If possible more data should be collected to better determine the best machine learning model and improve accuracy

Appendix

Github repository link - https://github.com/Sree-Vignesh-V/IBM_Data_science_Capstone/tree/main

Thank you!