Winning Space Race with Data Science

<Eduardo Antonio Lira Tapia>
<29-12-2021>

Outline

Executive Summary

Summary of methodologies

- ❑Data Collection
- ❑Data Wrangling
- ❑EDA with Data Visualization
- ❑Building an interactive map with Folium
- ❑Building a Dashboard with Plotly Dash
- ❑Predictive Analysis (Classification)

Summary of all results

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

Introduction

❑ Project background and context

We predicted if the Falcon 9 first stage will land successfully. 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 is because SpaceX can reuse 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 if an alternate company wants to bid against SpaceX for a rocket launch.

❑ Problems you want to find answers

- What influences if the rocket will land successfully?
- The effect each relationship with certain rocket variables will impact in determining the success rate of a successful landing.
- What conditions does SpaceX have to achieve to get the best results and ensure the best rocket success landing rate.

Section 1

Methodology

Methodology

- Executive Summary
- Data collection methodology
  - SpaceX Rest API
  - Web Scrapping from Wikipedia
- Perform data wrangling (Transforming data for Machine Learning)
  - One Hot Encoding data fields for Machine Learning and dropping irrelevant columns
- Perform exploratory data analysis (EDA) using visualization and SQL
  - Plotting: Scatter Graphs, Bar Graphs to show relationship between variables to show patterns of data
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - How to build, tune, evaluate classification models

Data Collection

The following datasets was collected by

- We worked with SpaceX launch data that is gathered from the SpaceX REST API.
- This API will give us data about launches, including information about the rocket used, payload delivered, launch specifications, landing specifications, and landing outcome.
- Our goal is to use this data to predict whether SpaceX will attempt to land a rocket or not.
- The SpaceX REST API endpoints, or URL, starts with api.spacexdata.com/v4/.
- Another popular data source for obtaining Falcon 9 Launch data is web scraping Wikipedia using BeautifulSoup.

Data Collection

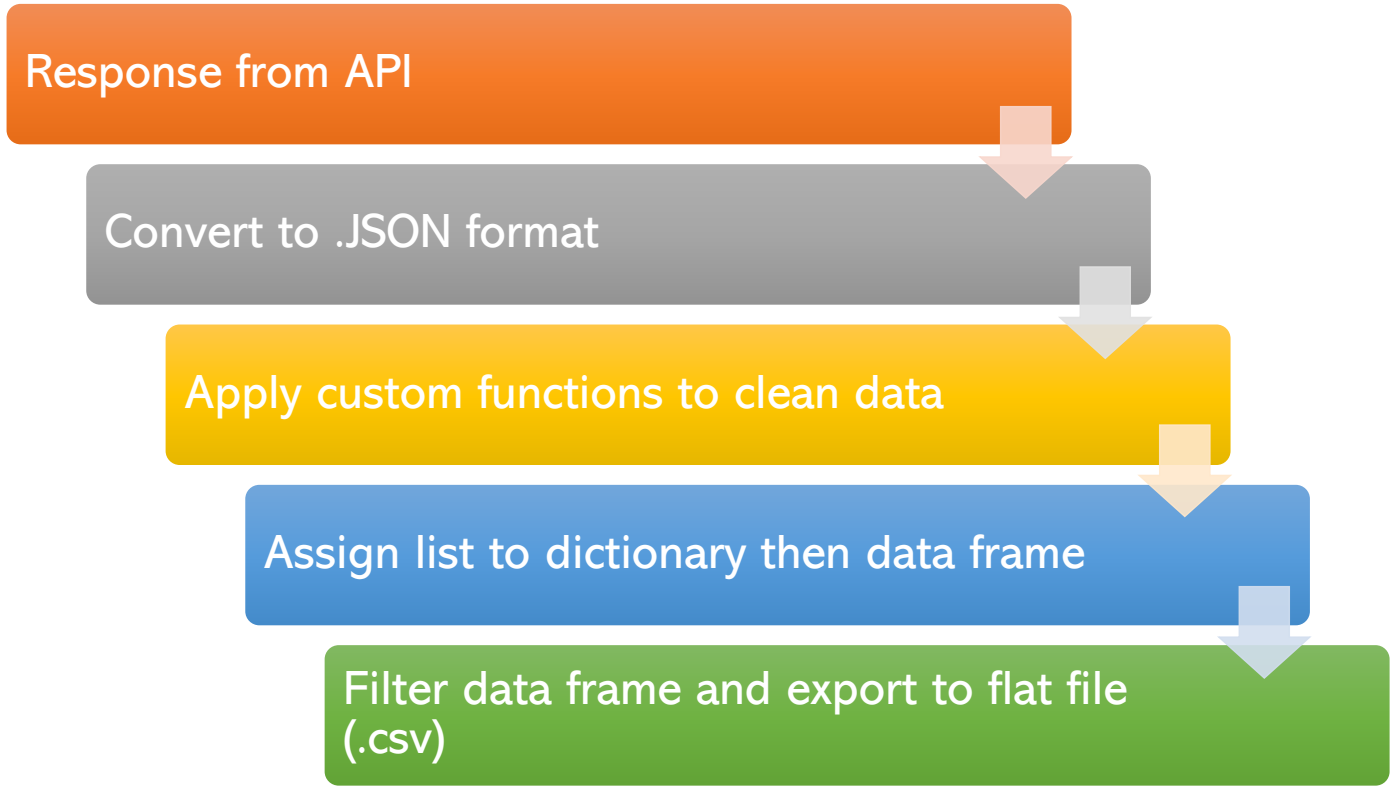
SpaceX API

- Use SpaceX REST API
- API returns SpaceX data in .JSON
- Normalize data into flat data file such as .csv

Web Scrapping

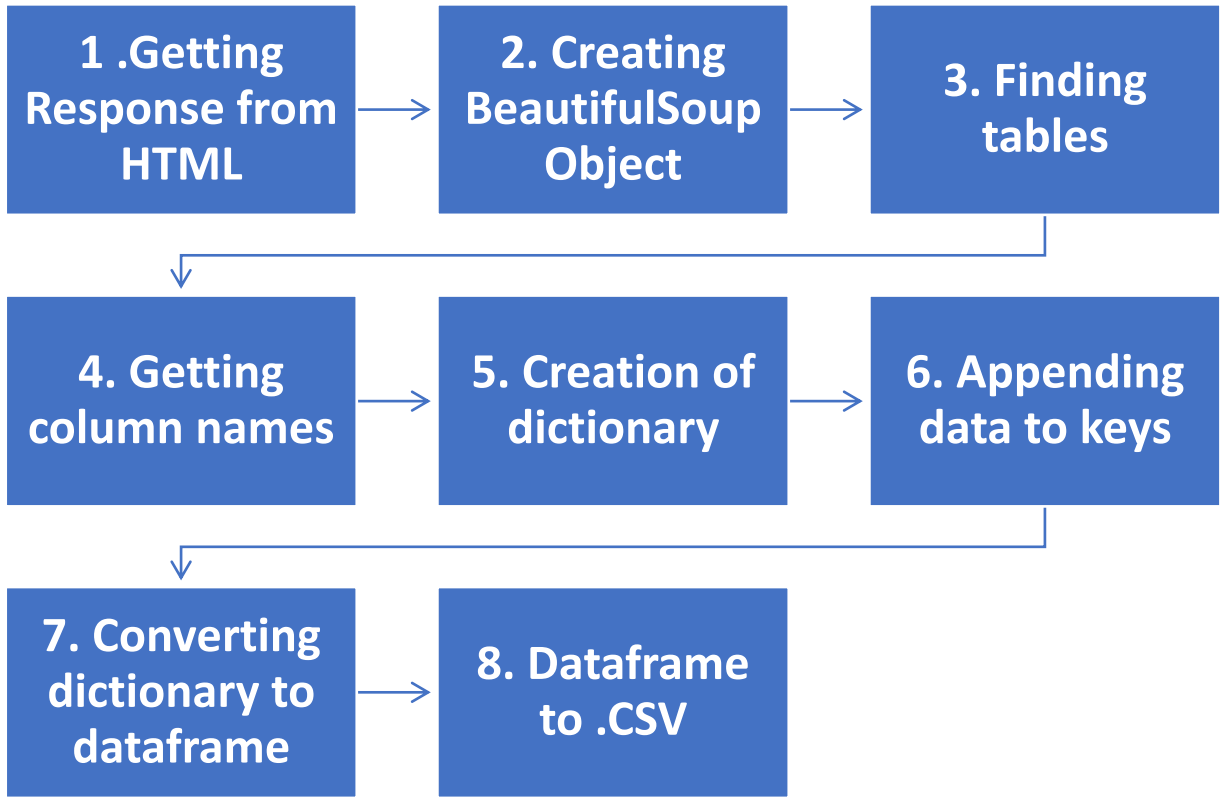
- Get HTML Response from Wikipedia
- Extract data using beautiful soup
- Normalize data into flat data file such as .csv

Data Collection – SpaceX API



GitHub URL NoteBook

Data Collection - Scraping



GitHub URL NoteBook

Data Wrangling

Introduction

In the data set, there are several different cases where the booster did not land successfully. Sometimes a landing was attempted but failed due to an accident; for example, True Ocean means the mission outcome was successfully landed to a specific region of the ocean while False Ocean means the mission outcome was unsuccessfully landed to a specific region of the ocean. True RTLS means the mission outcome was successfully landed to a ground pad False RTLS means the mission outcome was unsuccessfully landed to a ground pad. True ASDS means the mission outcome was successfully landed on a drone ship False ASDS means the mission outcome was unsuccessfully landed on a drone ship. We mainly convert those outcomes into Training Labels with 1 means the booster successfully landed0means it was unsuccessful.

Each launch aims to a dedicated orbit, and here are some common orbit types:

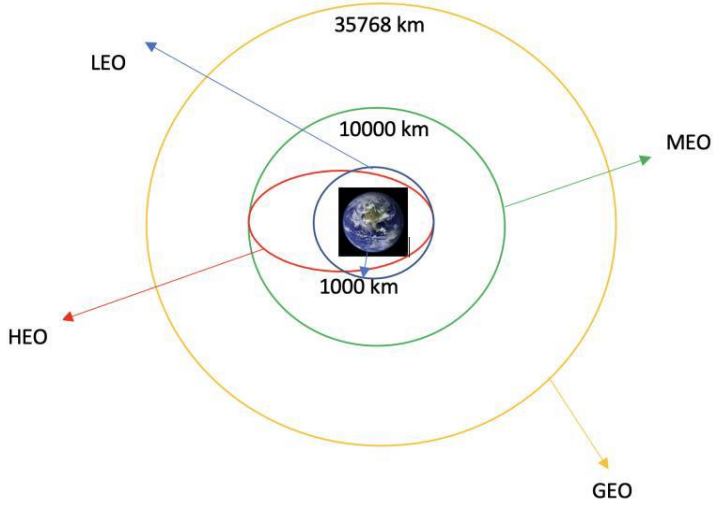
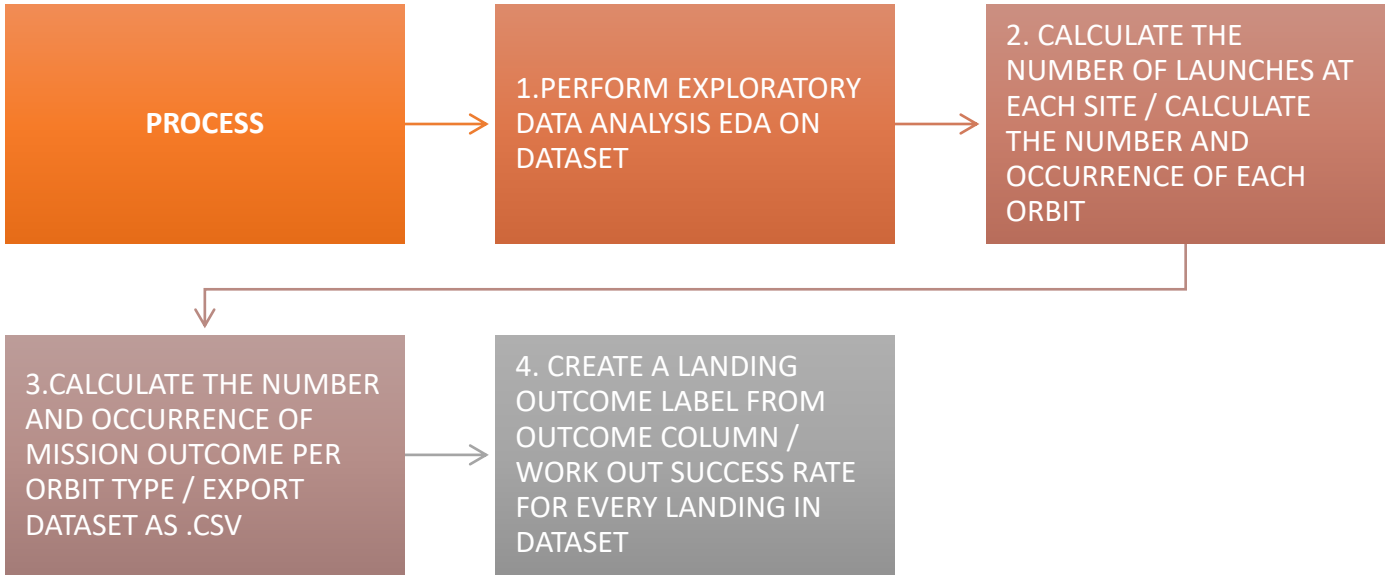


Diagram showing common orbit types SpaceX uses

Data Wrangling



EDA with Data Visualization

Scatter charts

Flight Number vs Payload Mass

Flight Number vs Launch Site

Payload vs Launch Site

Orbit vs Flight Number

Payload vs Orbit Type

Orbit vs Payload Mass

Scatter plots are particularly helpful graphs when we want to see if there is a **linear relationship among data points**. They indicate both the direction of the relationship between the x variables and the y variables, and the strength of the relationship.

Bar charts

Mean vs Orbit

A bar diagram makes it easy to compare sets of data between different groups at a glance. The graph represents categories on one axis and a discrete value in the other. The goal is to show the relationship between the two axes. Bar charts can also show big changes in data over time.

Line charts

Success Rate vs Year

A line graph is a graphical display of information that changes continuously over time. Within a line graph, there are various data points connected together by a straight line that reveals a continuous change in the values represented by the data points

URL GitHub NoteBook

EDA with SQL

- **Performed SQL queries to gather information about the dataset.**

- 1.Display the names of the unique launch sites in the space mission
- 2.Display 5 records where launch sites begin with the string 'CCA'
- 3. Display the total payload mass carried by boosters launched by NASA (CRS)
- 4. Display average payload mass carried by booster version F9 v1.
- 5. List the date when the first successful landing outcome in ground pad was achieved
- 6.List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000
- 7. List the total number of successful and failure mission outcomes
- 8. List the names of the booster_versions which have carried the maximum payload mass. Use a subquery
- 9. 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
- 10. Rank the count of successful landing_outcomes between the date 2010-06-04 and 2017-03-20 in descending order.

- GitHub URL NoteBook

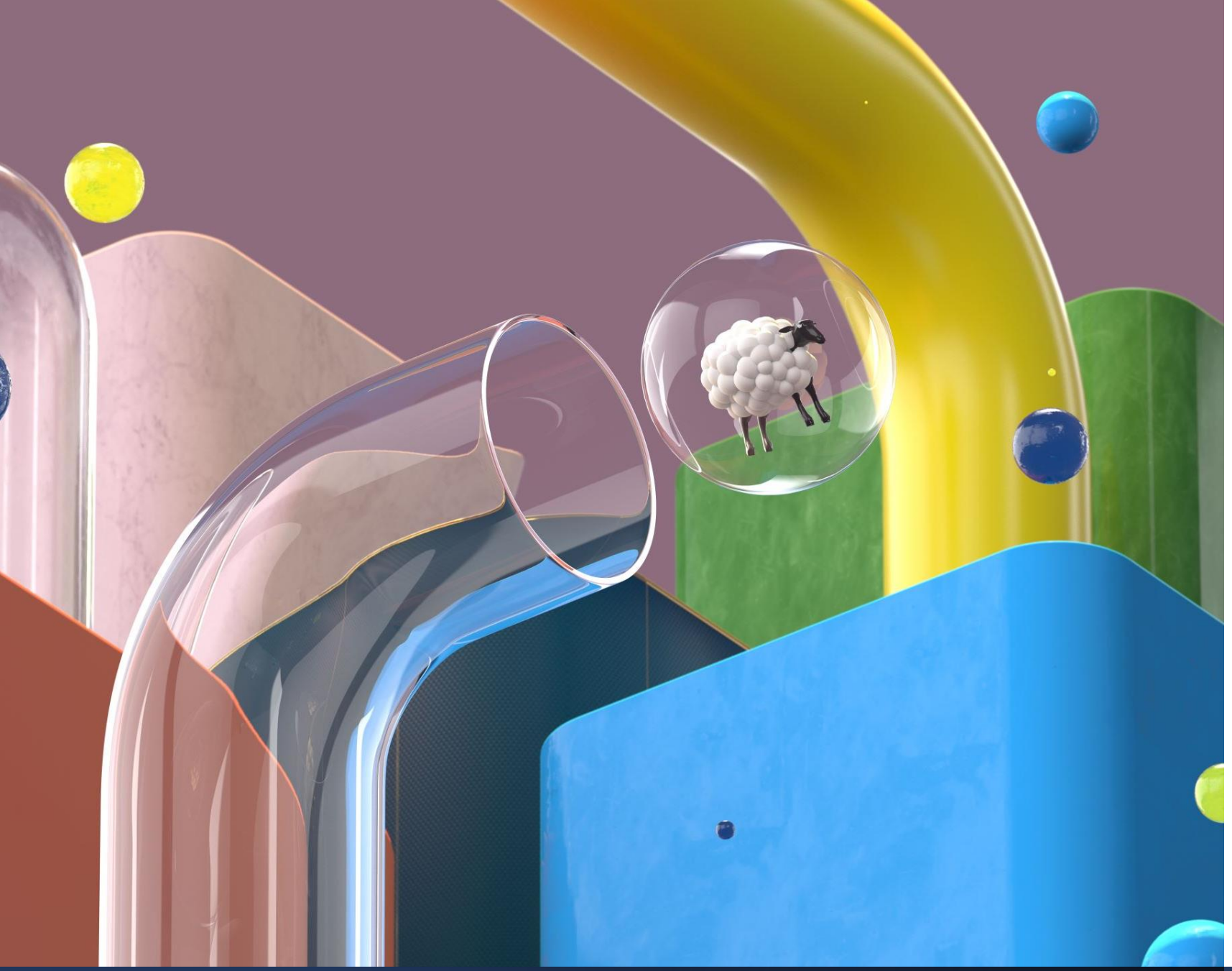
Build an Interactive Map with Folium

- **To visualize the Launch Data into an interactive map.** We took the Latitude and Longitude Coordinates at each launch site and added a *Circle Marker around each launch site with a label of the name of the launch site.*

- **We assigned the dataframe launch_outcomes(failures, successes) to *classes 0 and 1*** with Green and Red markers on the map in a MarkerCluster()

- **Using Haversine's formula we calculated the distance** from the Launch Site to various landmarks to find various trends about what is around the Launch Site to measure patterns. **Lines** are drawn on the map to measure distance to landmarks

- GitHub URL NoteBook



Predictive Analysis (Classification)

BUILDING MODEL

- Load our dataset into NumPy and Pandas
- Transform Data
- Split our data into training and test data sets
- Check how many test samples we have
- Decide which type of machine learning algorithms we want to use
- Set our parameters and algorithms to GridSearchCV
- Fit our datasets into the GridSearchCVobjects and train our dataset.

EVALUATING MODEL

- Check accuracy for each model
- Get tuned hyperparameters for each type of algorithms
- Plot Confusion Matrix

IMPROVING MODEL

- Feature Engineering
- Algorithm Tuning

FINDING THE BEST PERFORMING CLASSIFICATION MODEL

- The model with the best accuracy score wins the best performing model
- In the notebook there is a dictionary of algorithms with scores at the bottom of the notebook.

- GitHub URL Notebook

Results

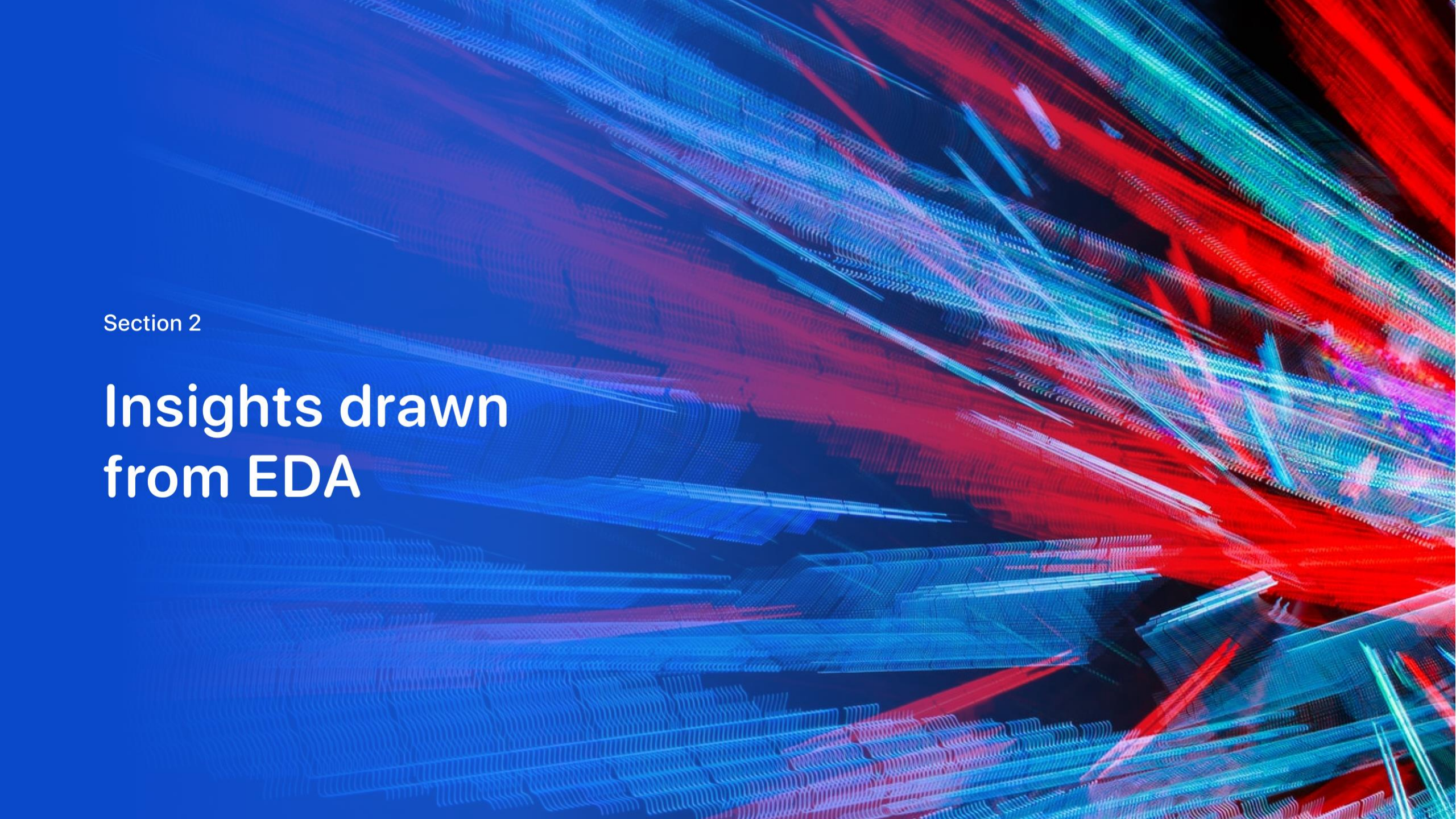
Exploratory data analysis results

Interactive analytics demo in screenshots

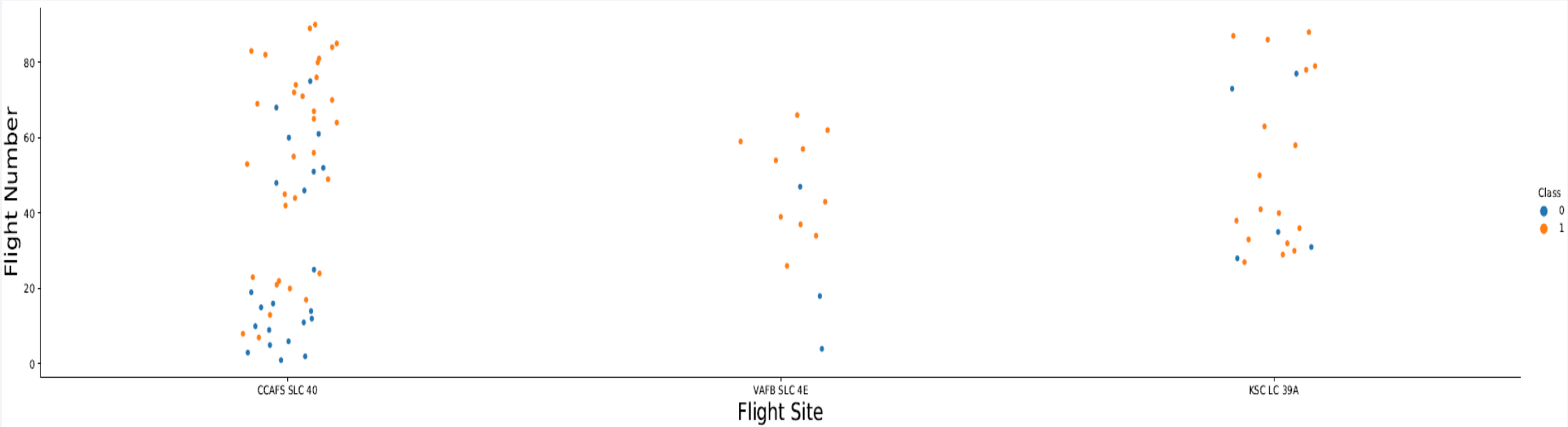
Predictive analysis results

Section 2

Insights drawn from EDA

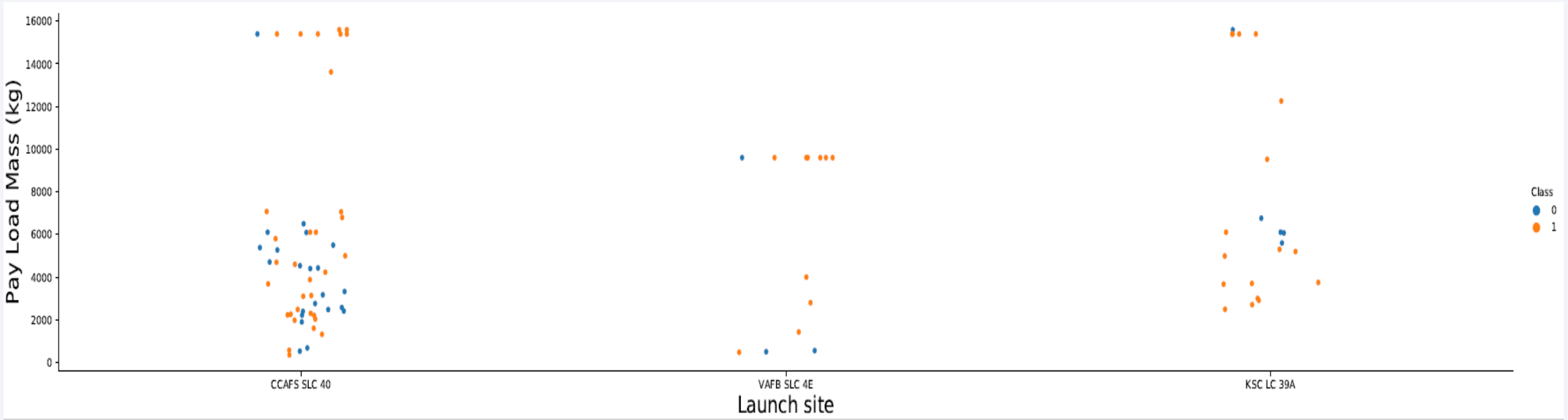


Flight Number vs. Launch Site



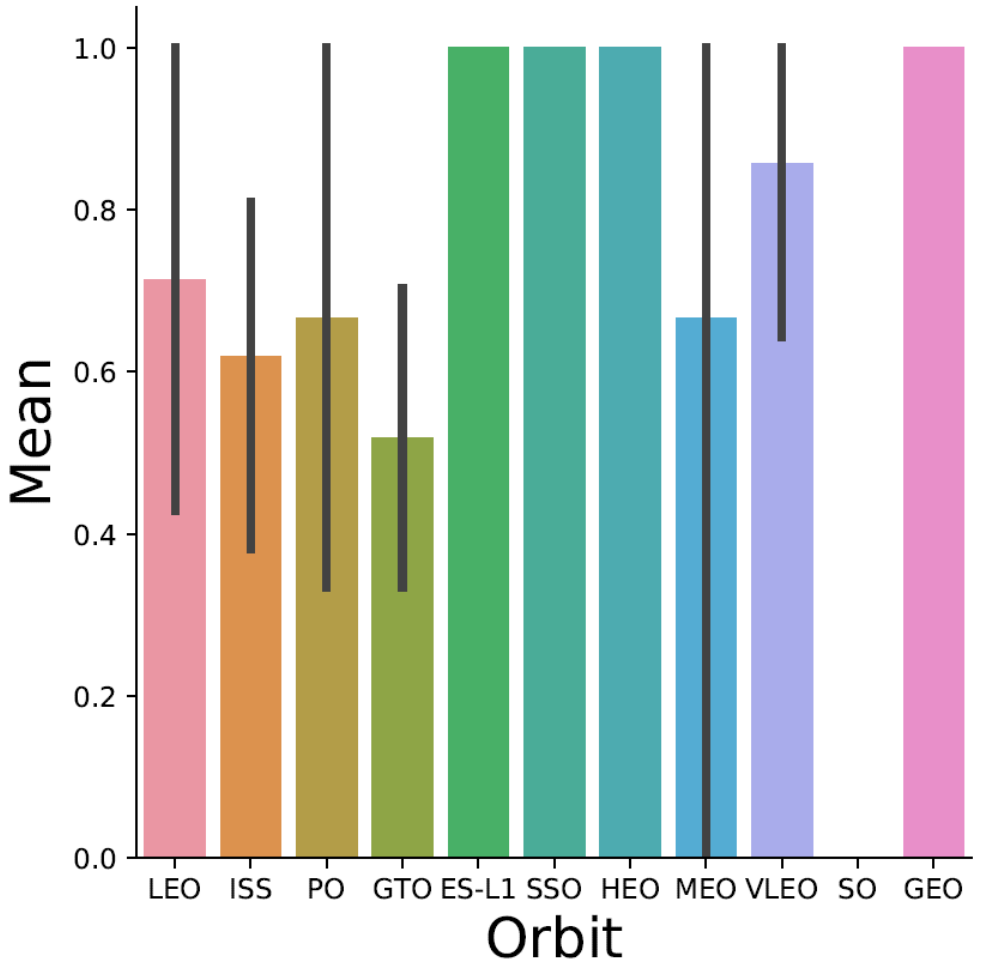
The greater number of flights at a launch site means a greater success rate at a launch site.

Payload vs. Launch Site



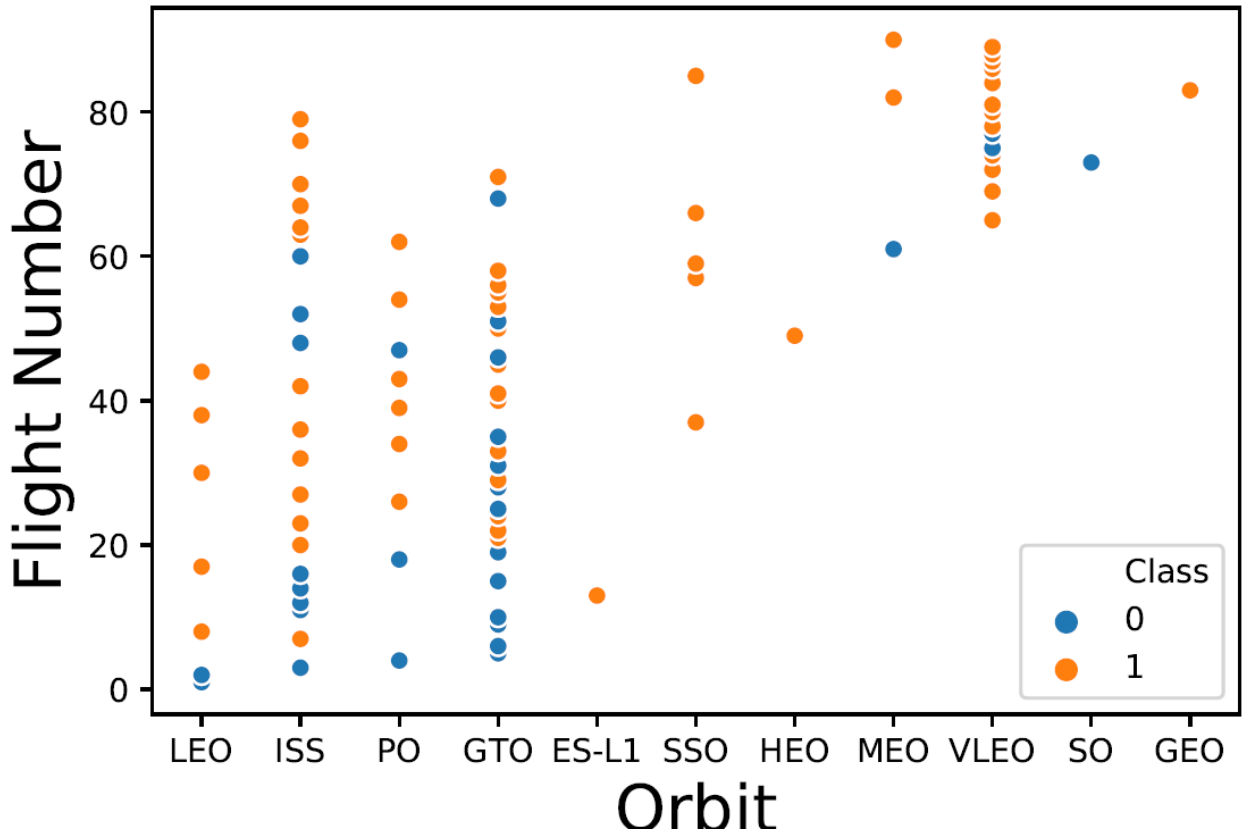
- The greater the payload mass for Launch Site CCAFS SLC 40 the higher the success rate for the Rocket.
- This visualization is not enough to decide if the Launch Site is dependent on Pay Load Mass for a success launch.

Success Rate vs. Orbit Type



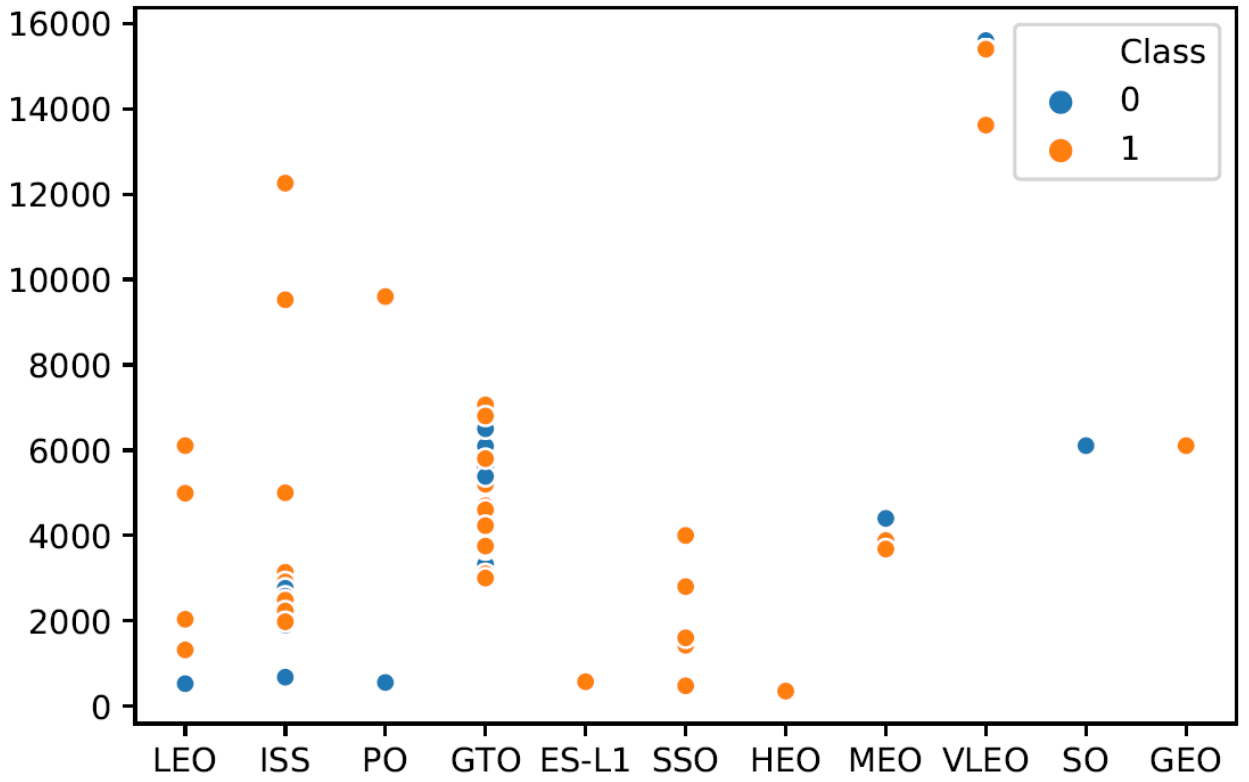
The Bar Chart shows that Orbit GEO,HEO,SSO,ES-L1 has the better Success Rate

Flight Number vs. Orbit Type



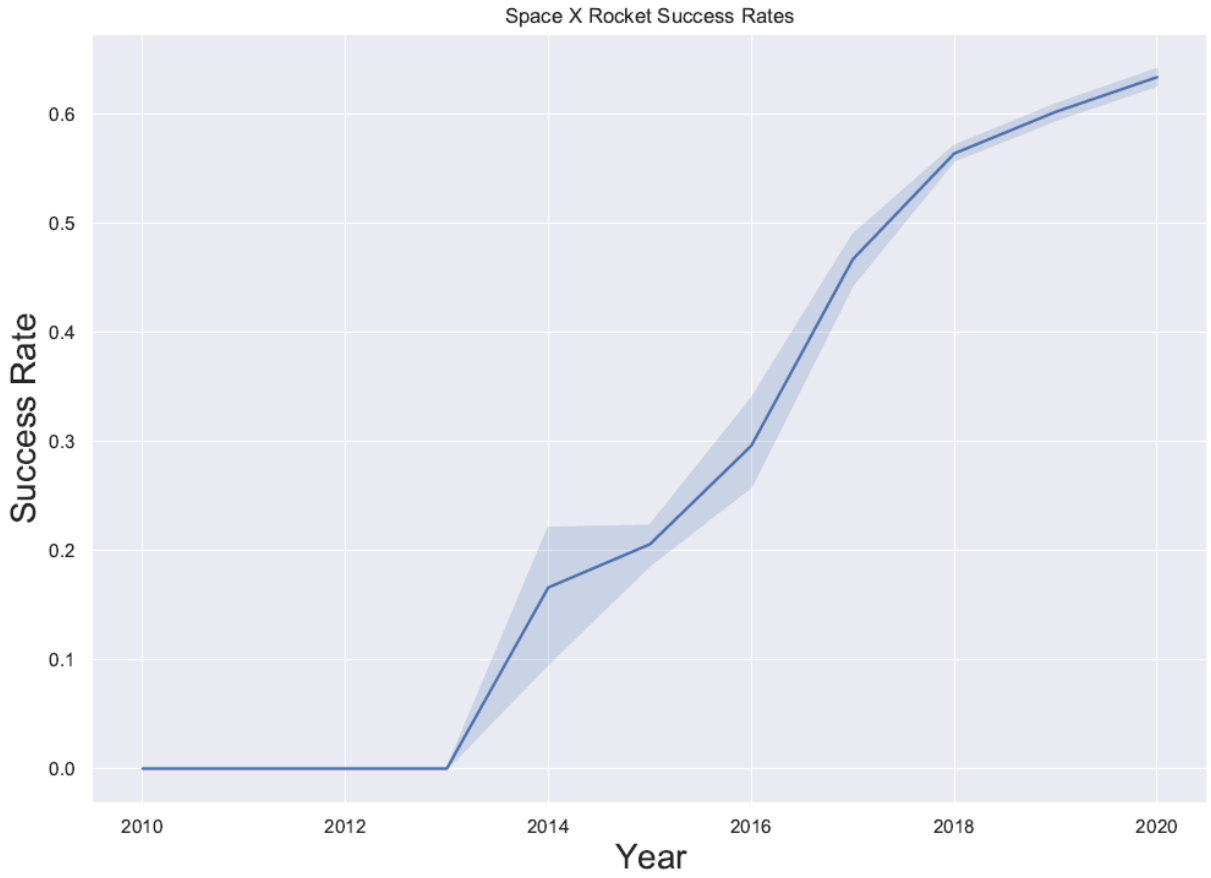
- There is a strong relationship between the Success of LEO orbit and VLEO orbit related to the number of flights
- There seems to be no relationship between flight number when in GTO orbit and ISS orbit.

Payload vs. Orbit Type



Heavy payloads have a negative influence on GTO orbits but have a positive on PO, LEO and ISS orbits.

Launch Success Yearly Trend



Since year 2013, the success rate has been increased every year till 2020!

All Launch Site Names

SQL QUERY

```
%sql select distinct(LAUNCH_SITE) from SPACEXTBL
```

Launch Site
CCAFS LC-40
CCAFS SLC-40
KSC LC-39A
VAFB SLC-4E

QUERY EXPLAINATION

Using the word *DISTINCT* in the query means that it will only show Unique values in the *Launch_Site* column from *SPACEXTBL*

Launch Site Names Begin with 'CCA'

SQL QUERY

```
%sql select * from SPACEXTBL where LAUNCH_SITE like 'CCA%' limit 5
```

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	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
2012-10-08	00:35:00	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

QUERY EXPLAINATION

Using the word ***LIMIT 5*** in the query means that it will only show 5 records from ***SPACEXTBL*** and ***LIKE*** keyword to show the Launch Sites with the words ***'CCA%'*** and the percentage in the end suggests that the Launch Site name must start with CCA.

Total Payload Mass

SQL QUERY

```
%sql select sum(PAYLOAD_MASS__KG_) from SPACEXTBL where CUSTOMER = 'NASA (CRS)'
```

Total Payload Mass
45596

QUERY EXPLAINATION

- Using the function *SUM* summates the total in the column *PAYLOAD_MASS_KG_*
- The *WHERE* clause filters the dataset to only perform calculations on *Customer NASA (CRS)*

Average Payload Mass by F9 v1.1

SQL QUERY

```
%sql select avg(PAYLOAD_MASS__KG_) from SPACEXTBL where BOOSTER_VERSION = 'F9 v1.1'
```

Average Payload Mass
2928.400000

QUERY EXPLAINATION

- Using the function ***AVG*** works out the average in the column ***PAYLOAD_MASS_KG_***
- The ***WHERE*** clause filters the dataset to only perform calculations on ***Booster_version F9 v1.1***

First Successful Ground Landing Date

SQL QUERY

```
%sql select min(DATE) from SPACEXTBL where Landing__Outcome = 'Success (ground pad)'
```

First Successful Ground Landing Date
2015-12-22

QUERY EXPLAINATION

- Using the function ***MIN*** works out the minimum date in the column ***Date***
- The ***WHERE*** clause filters the dataset to only perform calculations on ***Landing_Outcome Success (ground pad)***

Successful Drone Ship Landing with Payload between 4000 and 6000

SQL QUERY

```
%sql select BOOSTER_VERSION from SPACEXTBL where Landing__Outcome = 'Success (drone ship)'
and PAYLOAD_MASS__KG_ > 4000 and PAYLOAD_MASS__KG_ < 6000
```

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

QUERY EXPLAINATION

- Selecting only *Booster_Version*
- The *WHERE* clause filters the dataset to *Landing_Outcome = Success (drone ship)*
- The *AND* clause specifies additional filter conditions *Payload_MASS_KG_*>4000 AND *Payload_MASS_KG_ <6000*

Total Number of Successful and Failure Mission Outcomes

SQL QUERY

```
%sql select count(MISSION_OUTCOME) from SPACEXTBL where MISSION_OUTCOME = 'Success' or MISSION_OUTCOME = 'Failure (in flight)'
```

Successful Mission Outcomes	Failure Mission Outcomes
100	1

QUERY EXPLAINATION

- The *Count* clause is to count every successful or Faiulure Misión Outcomes
- The *WHERE* clause filters the dataset to *Mission_Outcome = Success or Failure*

Boosters Carried Maximum Payload

SQL QUERY

```
%sql select BOOSTER_VERSION from SPACEXTBL
where PAYLOAD_MASS__KG_ = (select max(PAYLOAD_MASS__KG_) from SPACEXTBL)
```

QUERY EXPLAINATION

The subquery is used to select the max value of *PAYLOAD_MASS_KG* from *SPACXTBL* and identify to which *BOOSTER_VERSION* this max value belongs, then each booster version that contains the max value of payload mass is listed.

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

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

SQL QUERY

```
%sql select * from SPACEXTBL where Landing__Outcome like 'Success%'
and (DATE between '2010-06-04' and '2017-03-20') order by date desc
```

DATE	time__utc_	booster_version	launch_site	payload	payload_mass__kg_	orbit	customer	mission_outcome	landing__outcome
2017-02-19	14:39:00	F9 FT B1031.1	KSC LC-39A	SpaceX CRS-10	2490	LEO (ISS)	NASA (CRS)	Success	Success (ground pad)
2017-01-14	17:54:00	F9 FT B1029.1	VAFB SLC-4E	Iridium NEXT 1	9600	Polar LEO	Iridium Communications	Success	Success (drone ship)
2016-08-14	05:26:00	F9 FT B1026	CCAFS LC-40	JCSAT-16	4600	GTO	SKY Perfect JSAT Group	Success	Success (drone ship)
2016-07-18	04:45:00	F9 FT B1025.1	CCAFS LC-40	SpaceX CRS-9	2257	LEO (ISS)	NASA (CRS)	Success	Success (ground pad)
2016-05-27	21:39:00	F9 FT B1023.1	CCAFS LC-40	Thaicom 8	3100	GTO	Thaicom	Success	Success (drone ship)
2016-05-06	05:21:00	F9 FT B1022	CCAFS LC-40	JCSAT-14	4696	GTO	SKY Perfect JSAT Group	Success	Success (drone ship)
2016-04-08	20:43:00	F9 FT B1021.1	CCAFS LC-40	SpaceX CRS-8	3136	LEO (ISS)	NASA (CRS)	Success	Success (drone ship)
2015-12-22	01:29:00	F9 FT B1019	CCAFS LC-40	OG2 Mission 2 11 Orbcomm-OG2 satellites	2034	LEO	Orbcomm	Success	Success (ground pad)

QUERY EXPLAINATION

- Select all data set using *
- ***Where*** to clause filters ***Landing_Outcome*** Successful between year 2010-2017.
- ***DESC*** means it arranging the date into descending order from 2017 to 2010

Section 4

Launch Sites Proximities Analysis

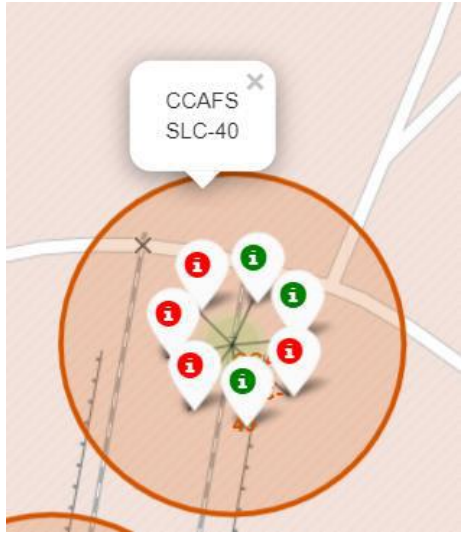
All Launches Sites Location

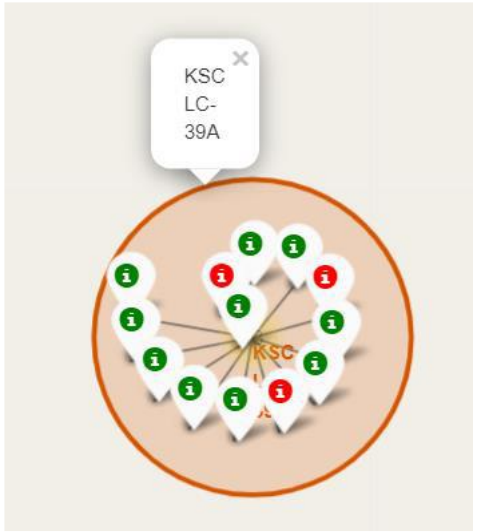
SpaceX launch sites are in the United States of America coasts. Specifically located in Florida and California

Color Labelled Markers

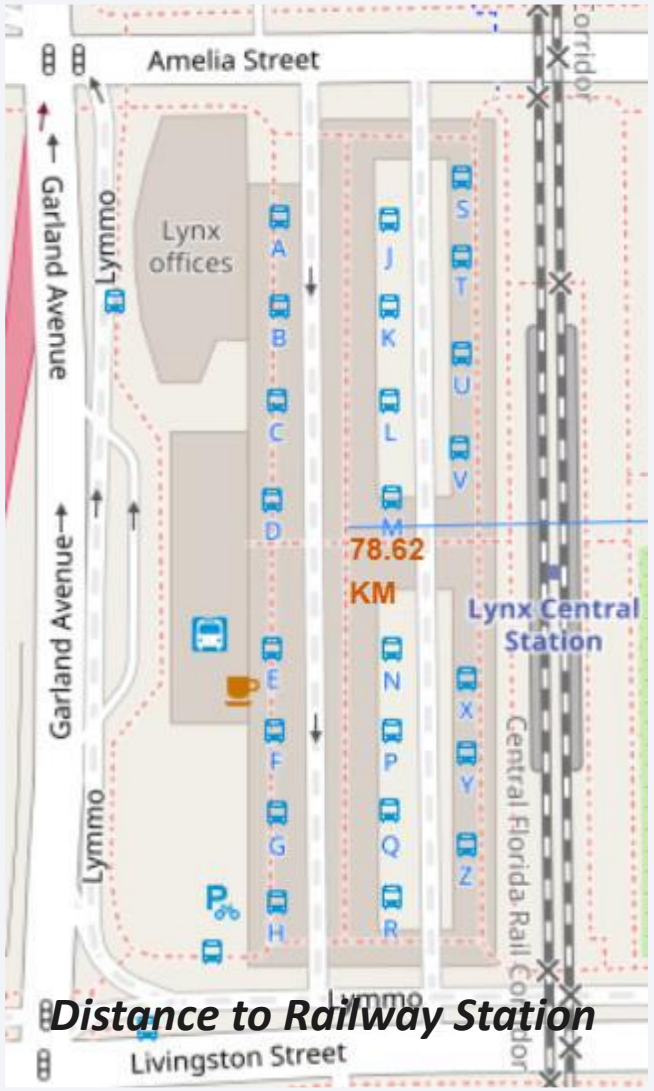
Florida Launch Sites

Green Marker shows successful Launches and Red Marker shows Failures





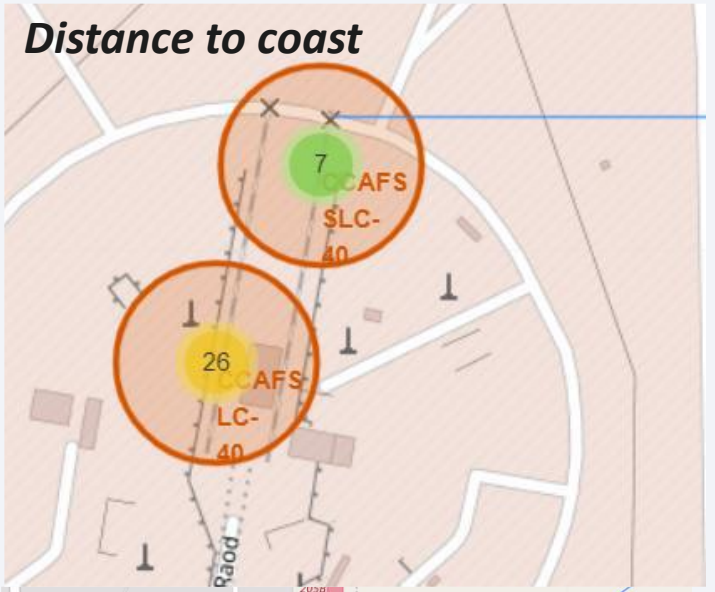
Launch Sites Distances



Distance to Railway Station

Distance to City

Distance to Coastline



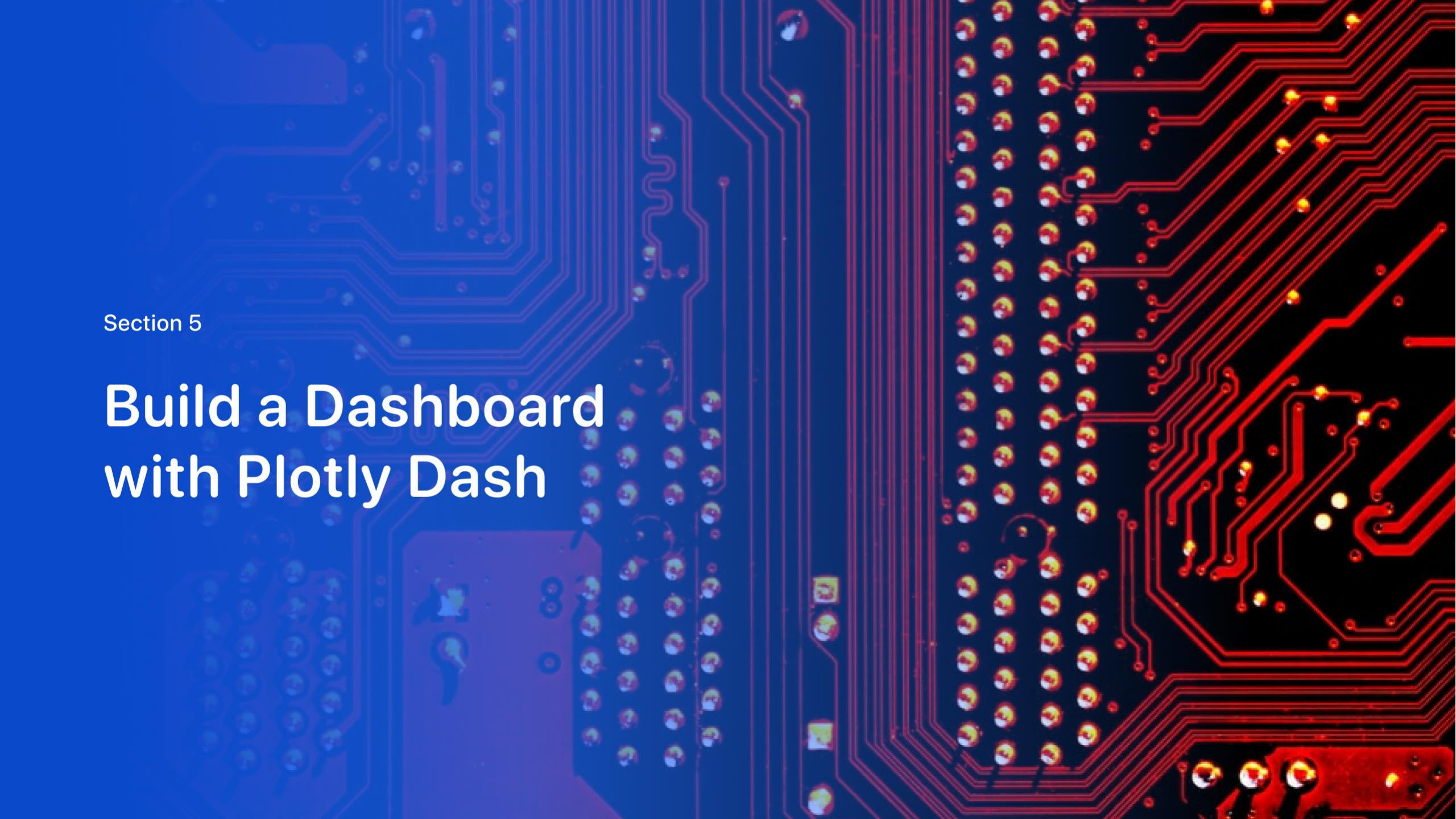
Distance to coast

Distance to closest Highway

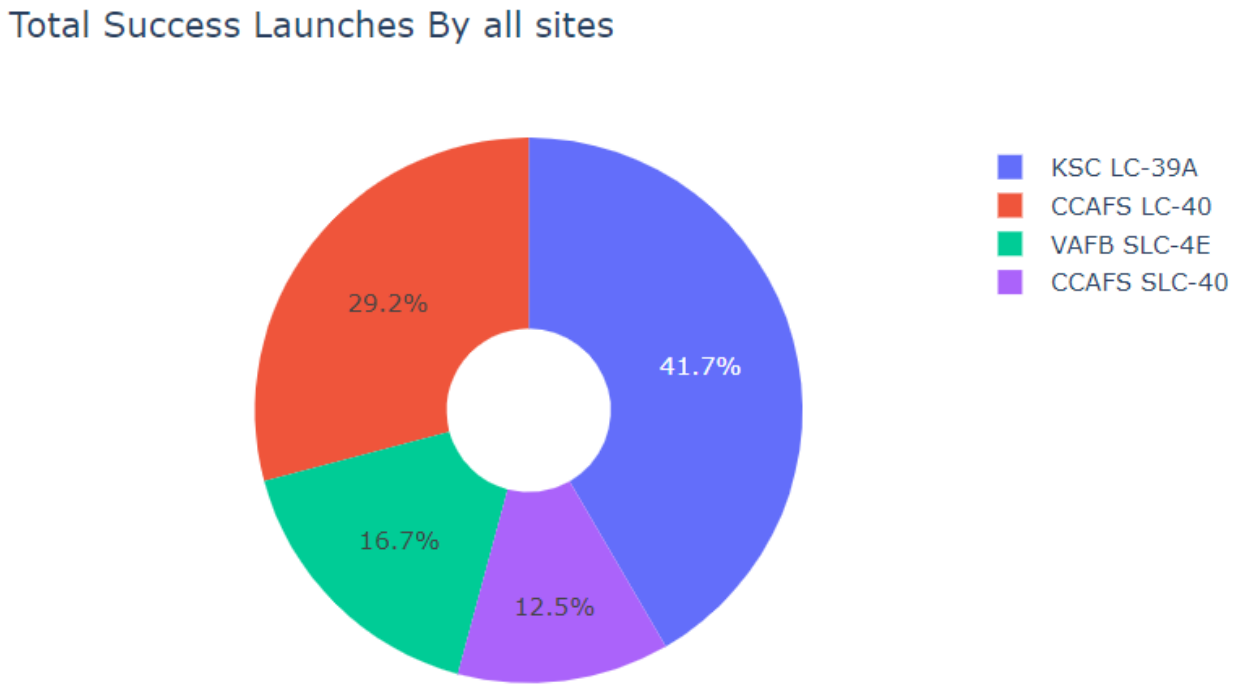
- Launch sites are not close to railways
- Launch sites are not close to highways
- Launch sites are close to coastline
- Launch sites keep certain distance away from cities.

Section 5

Build a Dashboard with Plotly Dash

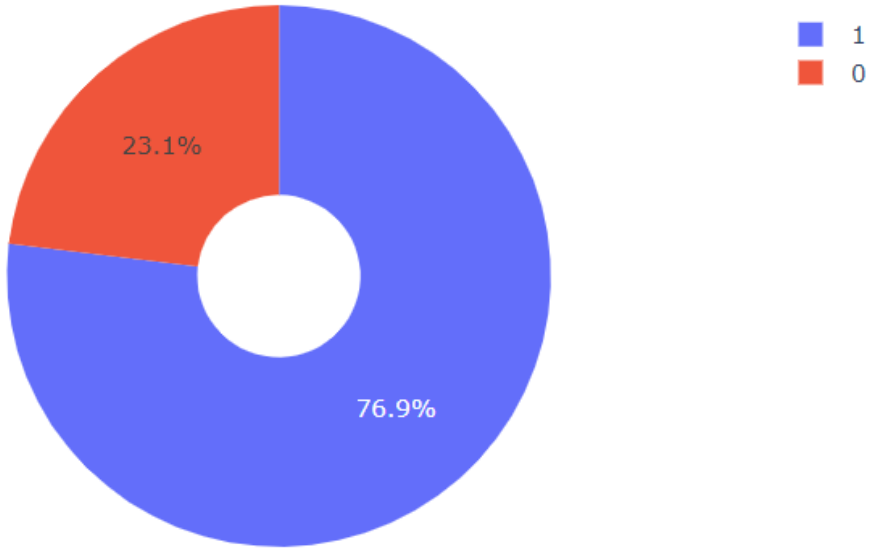


Success percentage achieved by each Launch Site



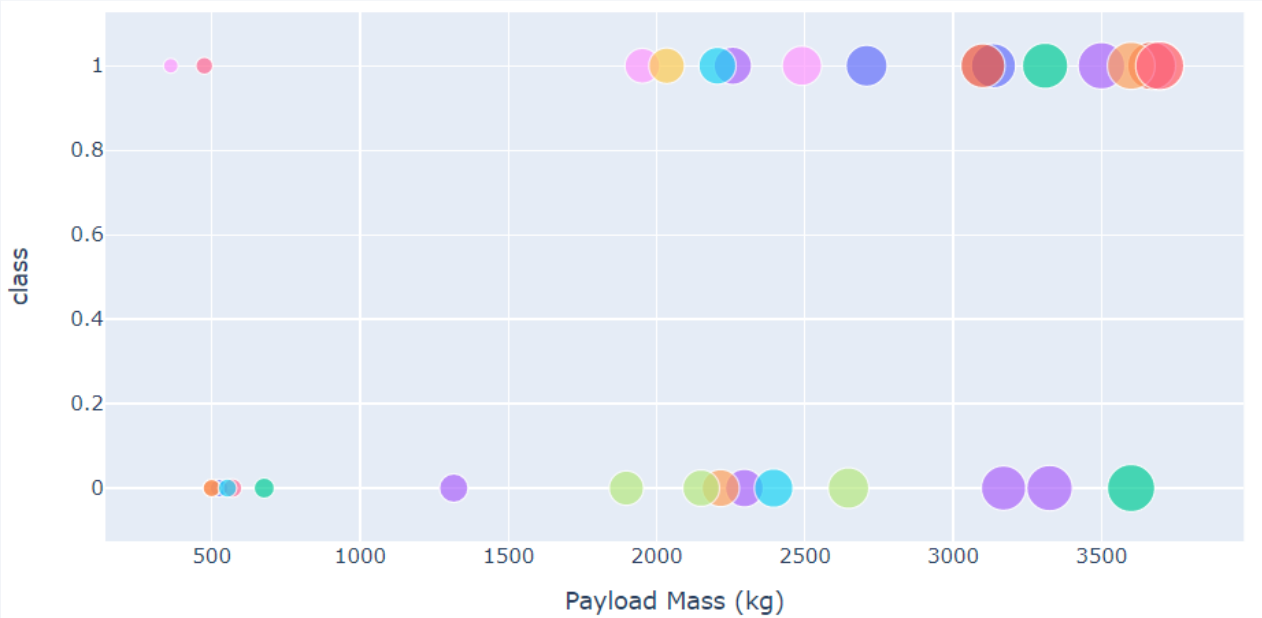
Pie Chart shows that KSC LC-39A had the most successful launches from all the sites

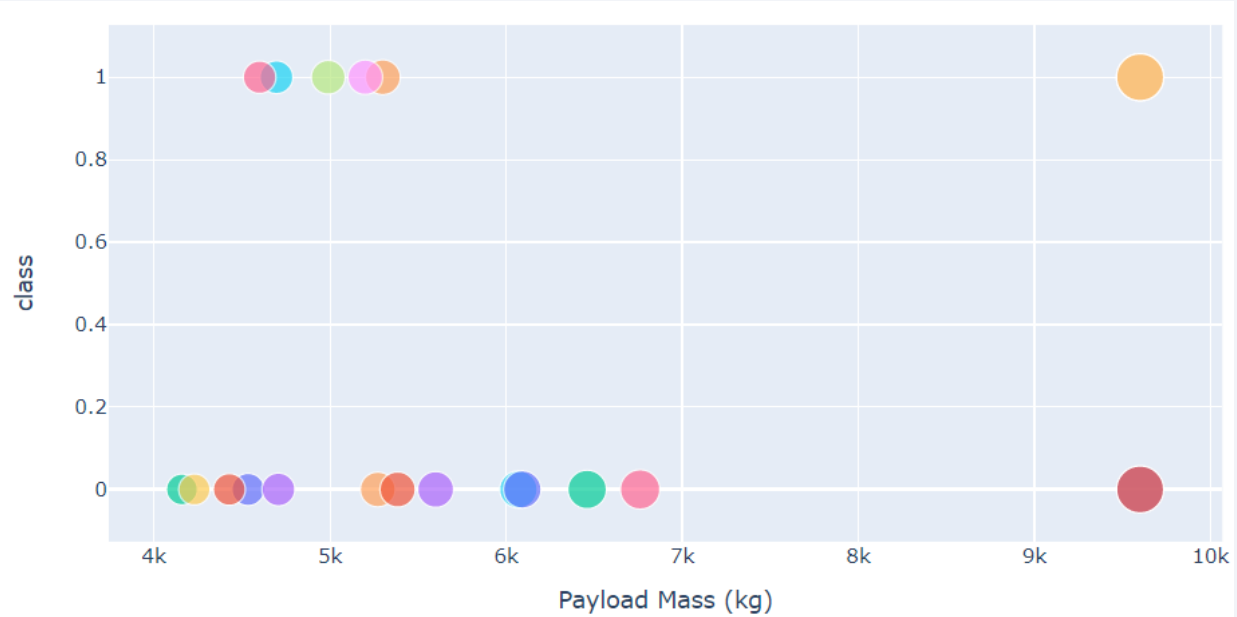
Launch site with highest launch success ratio



KSC LC-39A achieved a 76.9% success rate while getting a 23.1% failure rate

Payload vs. Launch Outcome





The success rates for low weighted payloads is higher than the heavy weighted payloads

Section 6

Predictive Analysis (Classification)

Classification Accuracy

The Algorithm who got the higher accuray is Tree. The next function was used to compare the 3 algorithm used and decide the best classification algorithm

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

```
Best Algorithm is Tree with a score of 0.8892857142857145
Best Params is : {'criterion': 'gini', 'max_depth': 4, 'max_features': 'sqrt', 'min_samples_leaf': 1, 'min_samples_split': 5,
'splitter': 'random'}
```

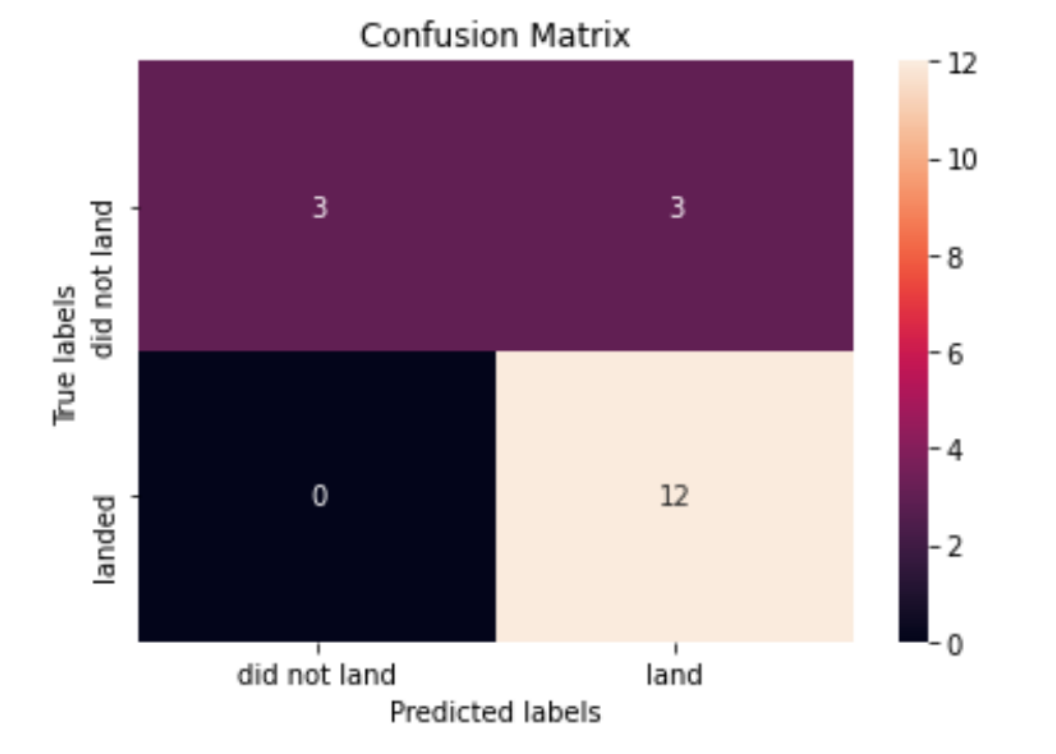
Accuracy	Algorithm
0.8333333333333334	SVM
0.8892857142857145	Tree
0.8482142857142858	KNN

After selecting the best hyperparameters for the decision tree classifier using the validation data, we achieved 83.33% accuracy on the test data.

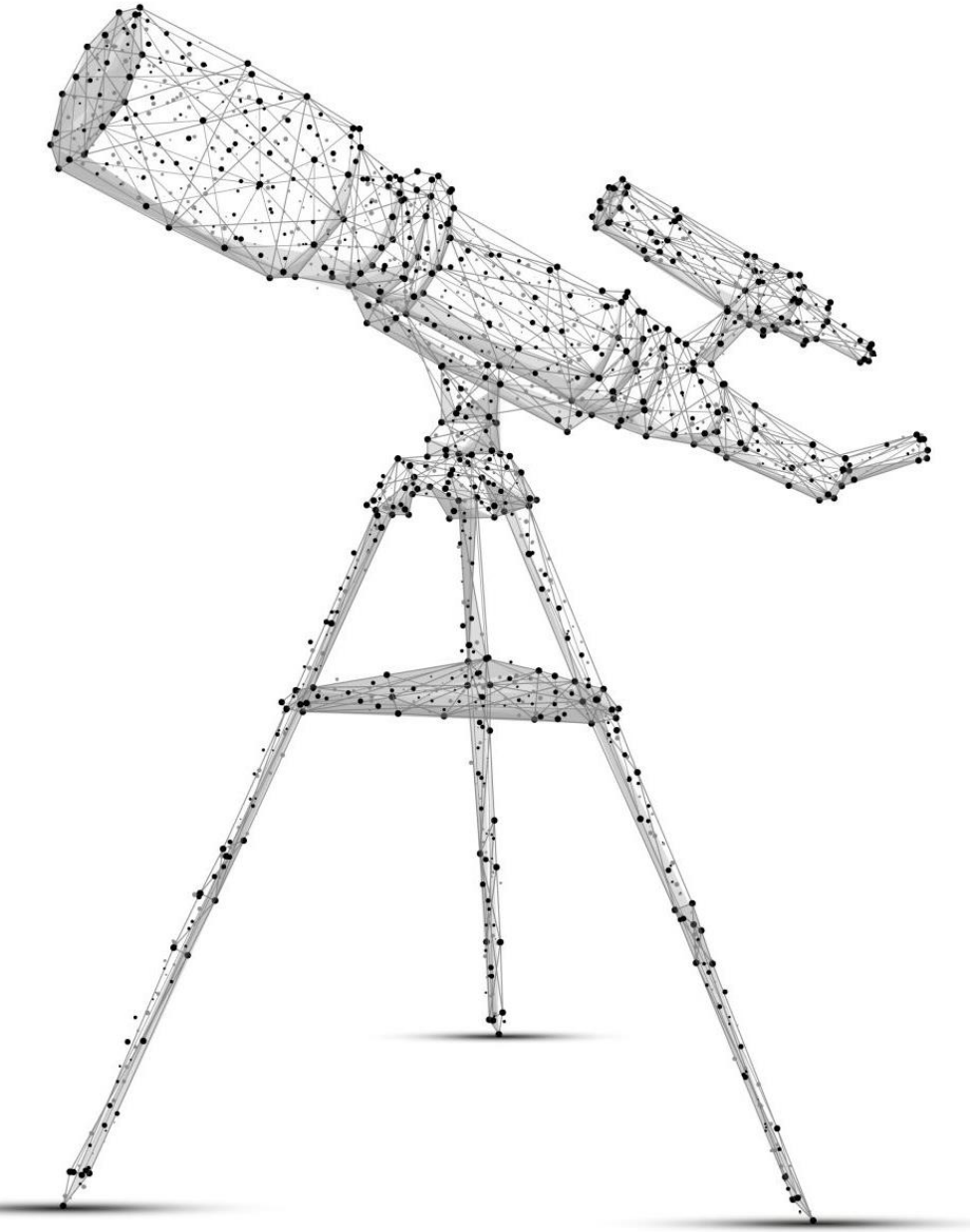
```
1 print("accuracy: ",tree_cv.score(X_test,Y_test))
```

```
accuracy:  0.8333333333333334
```

Confusion Matrix Decision Tree



Examining the confusion matrix, we see that Tree can distinguish between the different classes. We see that the major problem is false positives.



Conclusions

- The Tree Classifier Algorithm is the best for Machine Learning for this dataset
- Low weighted payloads perform better than the heavier payloads
- The success rates for SpaceX launches is directly proportional time in years they will eventually perfect the launches
- KSC LC-39A had the most successful launches from all the sites
- Orbit GEO,HEO,SSO,ES-L1 has the best Success Rate

Thank you!