IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Asim Shah
7/15/2023

Outline

Executive Summary

- Summary of methodologies
  - Data Collection – Web Scraping & Interacting w/ SpaceX API's
  - Data Wrangling, Pre-processing, & Feature Engineering
  - EDA - Data Visualization w/ Python & Tableau
  - EDA w/ SQL – Gaining Insights from Aggregating Data
  - Building interactive Maps using Folium
  - Build & Deploy an Interactive Dashboard using Plotly & Dash
  - Predictive Analysis – Build, Train, and Evaluate Different ML Classification Models Using the Optimal combinations of Hyperparameters
- Summary of all results
  - EDA Results
  - Interactive analytics Results
  - Predictive analysis Results

Introduction

Project Background and Context:

- Objective: Predict the successful landing of Falcon 9 first stage rockets.
- Importance: Cost optimization in rocket launches, with SpaceX's reusability feature leading to significant savings

Problems to Find Answers:

- Cost Comparison: Understand how SpaceX saves money through reusability, offering launches at $62 million compared to competitors' costs of $165 million.
- Landing Success: Determine factors that contribute to successful landings and identify reasons behind unsuccessful landings.
- Cost Estimation: Utilize machine learning models to predict the success or failure of Falcon 9 first stage landings, and leverage these predictions to estimate the cost of a launch

Section 1

Methodology

Methodology

Executive Summary:

- **Data collection methodology**
  - Interacting with SpaceX API's
  - Web-Scraping a SpaceX Wikipedia Table using Beautiful Soup
- **Perform data wrangling**
  - One hot encoding data fields for Machine Learning, Handling Missing Values, Dropping Irrelevant Columns, Data Formatting & Data Types, Constructing the Landing Success Column (Output)
- **Perform exploratory data analysis (EDA) using visualization and SQL**
- **Perform interactive visual analytics using Folium and Plotly Dash**
- **Perform predictive analysis using classification models**
  - Performing Feature Scaling
  - Build and Train Logistic Regression, SVM, Decision Tree, and K-Nearest Neighbors Classification models using the optimal hyperparameters/Cross Validation
  - Use evaluation metrics to see how well the models perform on training data & generalize on test data.

Data Collection

Interacting with SpaceX API's:

- This process involves retrieving data from the SpaceX API, organizing it into a tabular structure, selecting specific columns, fetching additional launch details using API keys, and transforming the gathered data into a Data Frame for analysis and visualization.
- By doing this, we are able to collect all sorts of SpaceX Launch data, obtaining features such as booster name, payload mass, orbit, etc.

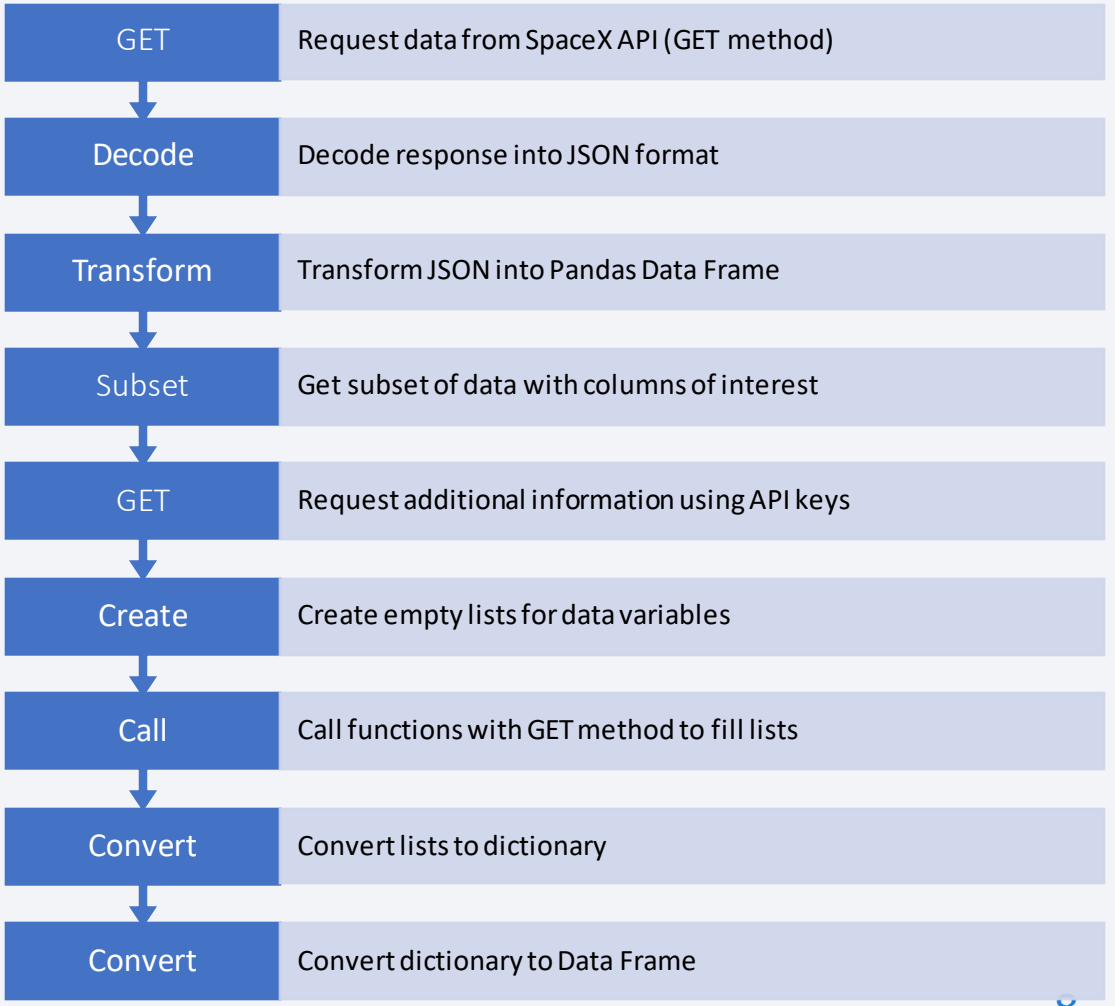
Web-Scraping a SpaceX Wikipedia Table using Beautiful Soup:

- In this process, data was extracted from a Wiki page by creating a Beautiful Soup object. The column names were assigned as keys in a dictionary, and empty lists were created for each key. By parsing the table on the page, the data was extracted and appended to the respective lists. Finally, the dictionary was converted into a Data Frame for further analysis.

Data Collection – SpaceX API

Interacting with SpaceX API's:

- We first requested data from the SpaceX API using the GET method and containing the result in an object
- Then decoded the response content into a Json type format and transformed it into a Pandas Data Frame
- After we have our data defined, we then got a subset of that Data with the columns of interest such as 'Payload', 'Date', etc.
- We then use SpaceX API's again to get additional information about the launches using the keys, for example we can use the payload key to get data mass amount variable or the rocket key to get the booster version type
- We create empty lists for all of the data variables for which we want to gather using our keys, we call our functions which use the GET method and fill the lists which we then convert to a DICT and then to a Data Frame

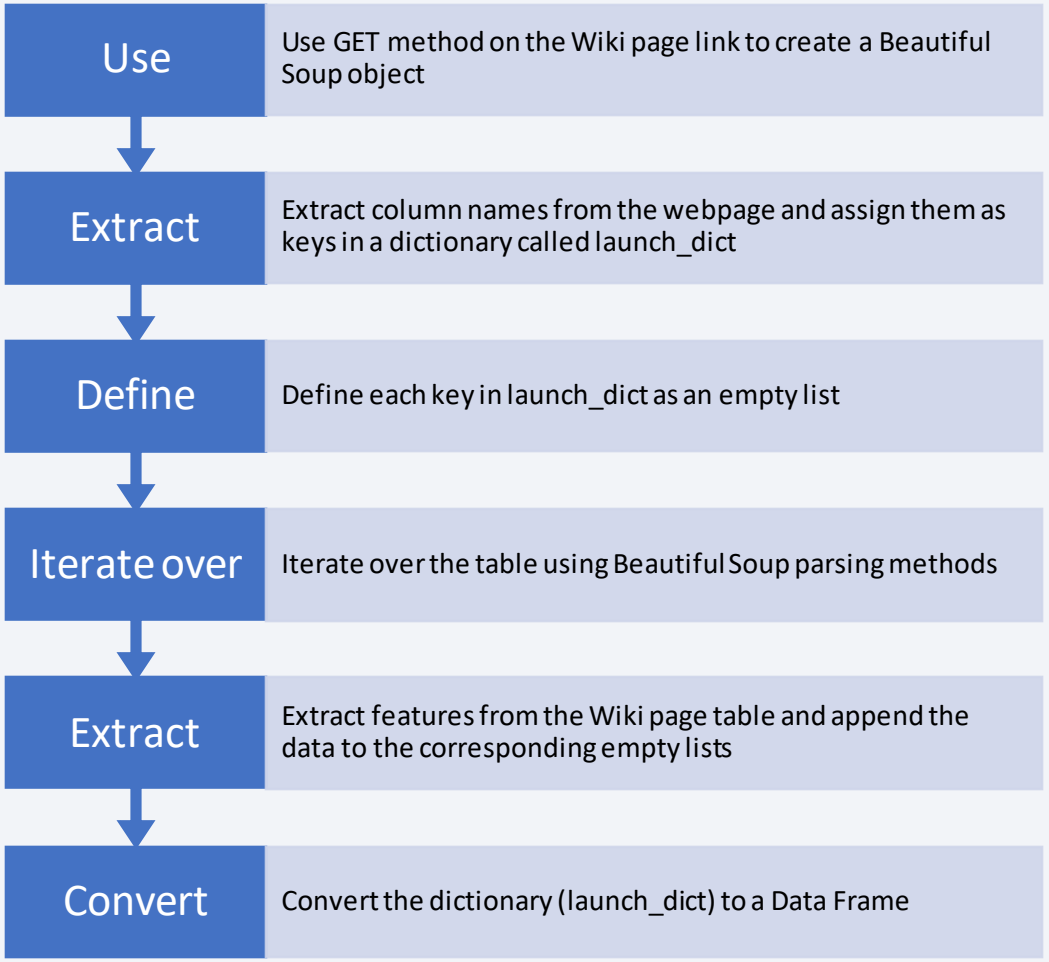


Git-hub Link: https://github.com/AsimAShah/SpaceX-Data-Science-Capstone-Project/blob/Main-Branch/Data%20Collection%20-%20SpaceX%20API's.ipynb

Data Collection - Web Scraping

Web-Scraping a SpaceX Wikipedia Table using Beautiful Soup:

- The first thing we did was use the GET method on the Wiki page link and created a Beautiful Soup Object
- We then extracted the column names from the webpage and assigned them to keys in a DICT called launch_dict
- We then define each of these keys as empty lists
- Lastly, we iterate over the table using Beautiful Soup parsing methods and extract the features from Wiki page table and append the data to our empty lists
- We then converted the DICT to a Data Frame



Git-hub Link: https://github.com/AsimAShah/SpaceX-Data-Science-Capstone-Project/blob/Main-Branch/WebScraping%20wiki%20w%20Beautiful%20Soup.ipynb

Data Wrangling

Data Wrangling Techniques Applied:

Handling Missing Values:

- Payload mass: Replace 5 NULL values with the payload mean.
- Landing pad: Leave missing values to indicate no landing pad was used.

Dropping Irrelevant Columns/Filtering:

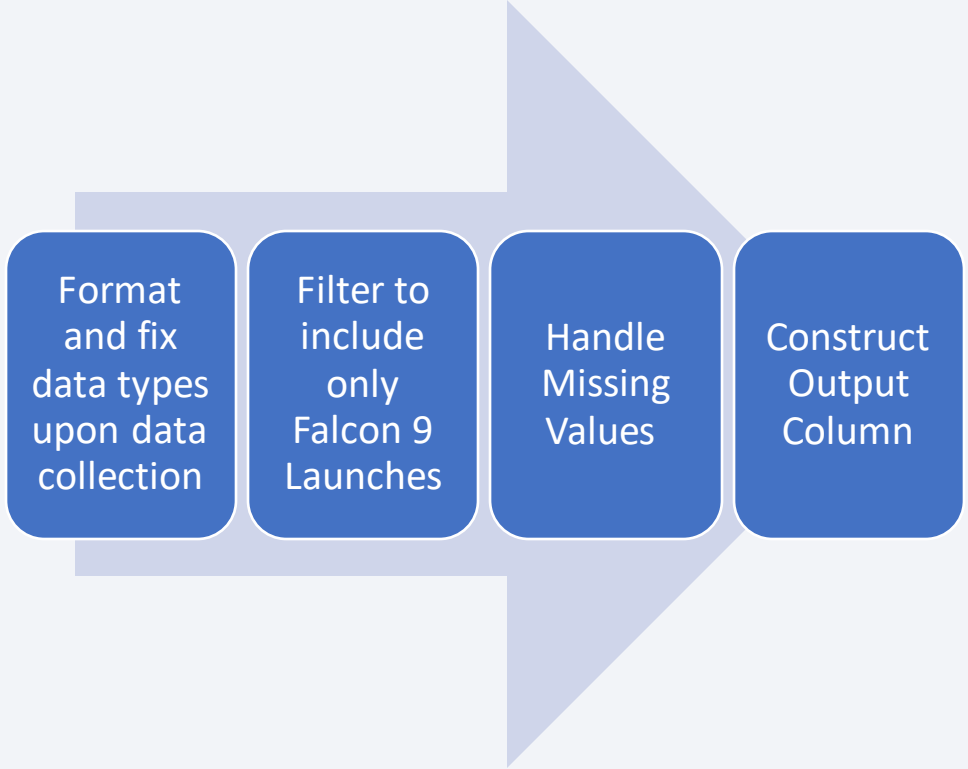
- Falcon 9 launches: Ensure the dataset only includes Falcon 9 launches.
- Multiple cores & payloads: Remove rows with multiple cores and multiple payloads.

Data Formatting & Data Types:

- Date column: Convert to date type, extracting the date and excluding the time.
- Lists with one value: Remove lists that contain only one value.

Constructing the Landing Success Column (Output):

- Outcome column analysis: Count unique outcomes and their frequencies.
- Identifying launch failures: Remove counts and assign index numbers to identify launch failures. Create a set for unsuccessful second stage launches.
- Class column creation: Iterate over the Outcome column, appending 0 for failure and 1 for success. Create a new column (Class) and assign the appended data.

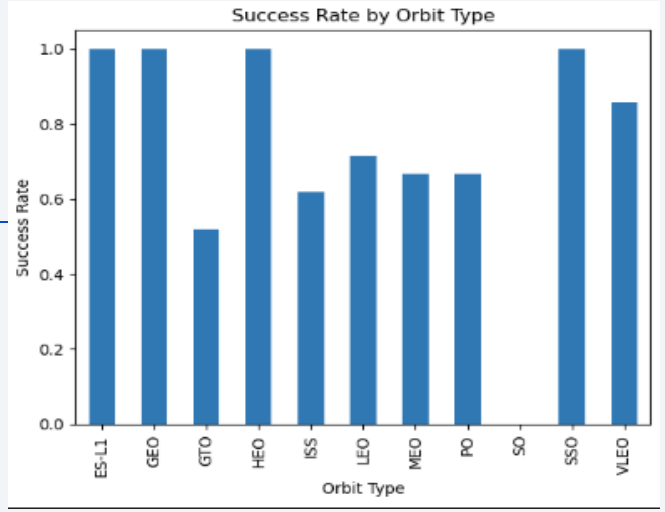


Git-hub Link: https://github.com/AsimAShah/SpaceX-Data-Science-Capstone-Project/blob/Main-Branch/Data%20Wrangling.ipynb

EDA with Data Visualization

Git-hub Link: https://github.com/AsimAShah/SpaceX-Data-Science-Capstone-Project/blob/Main-Branch/EDA%20-%20Data%20Vizualizations.ipynb

Used a combination of Dot Plots, Strip Plots, Bar Charts, and Line Charts along with color classified data points to determine which points are successful/failures in order to explore the data & gain insights

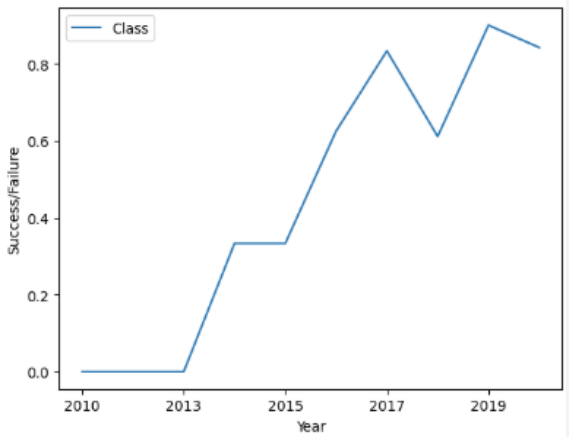


Bar Charts:

- We can see that Orbit types ES-l1, GEO, HEO, and SSO have the highest success rate

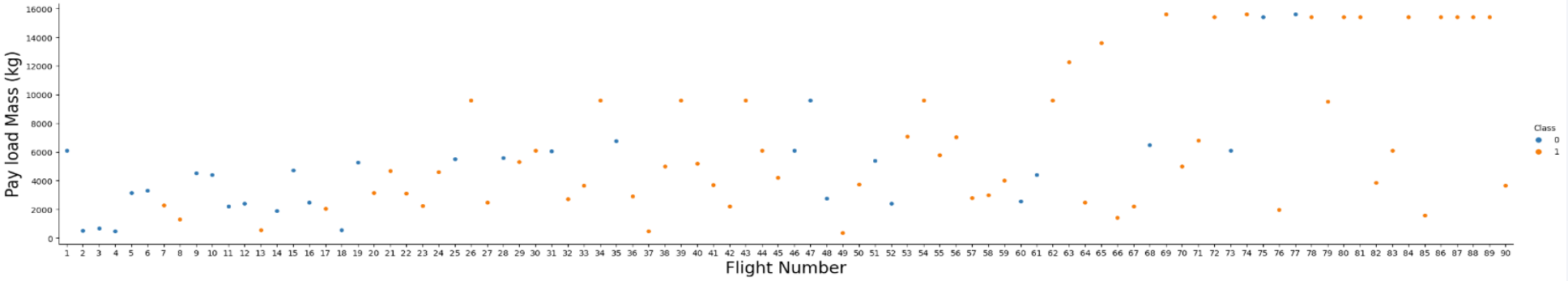
Line Chart:

- We use a Line Chart to represent the success rate and how it fluctuated from years 2010 - 2019

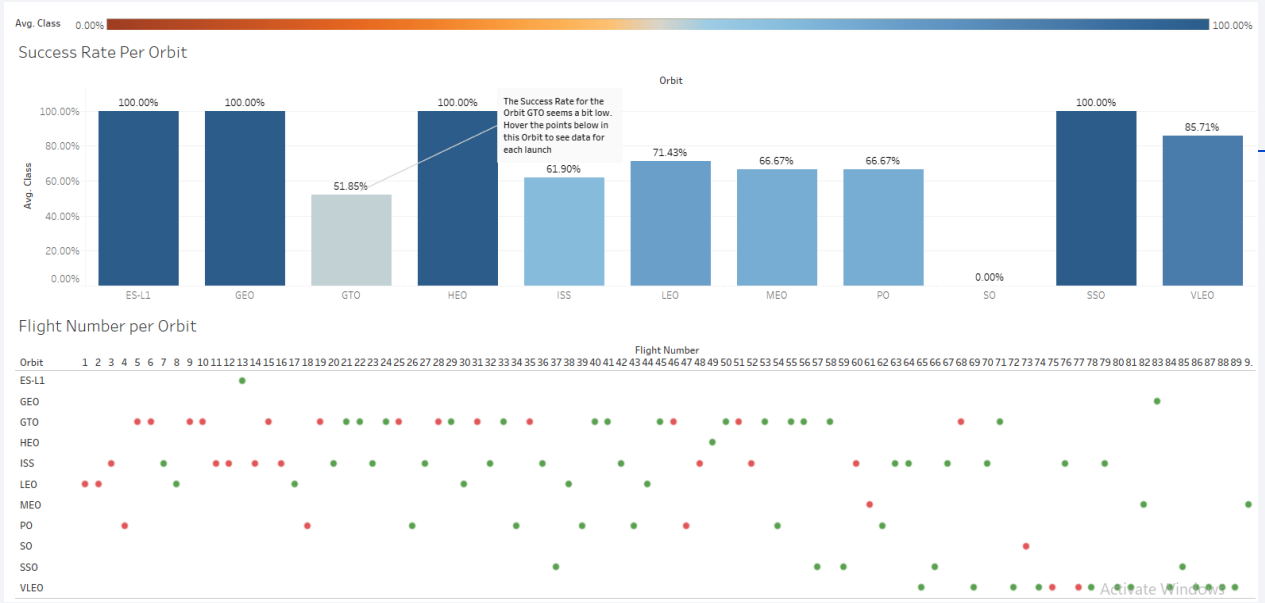


Dot Plots & Strip Plots:

- We can see that most of the successful launches have a payload mass of 10k (kg) or less
- We can see that site CCAFS LC-40: has the most launches
- We can see that for Site: VAFB SLC 4, we have no launches above 10k payload mass
- We noticed that the last 5 launches to Orbit LEO have been successful & PO and ISS orbits show trends that have a higher success rate for heavier payloads



EDA with Data Visualization – Tableau Dashboard



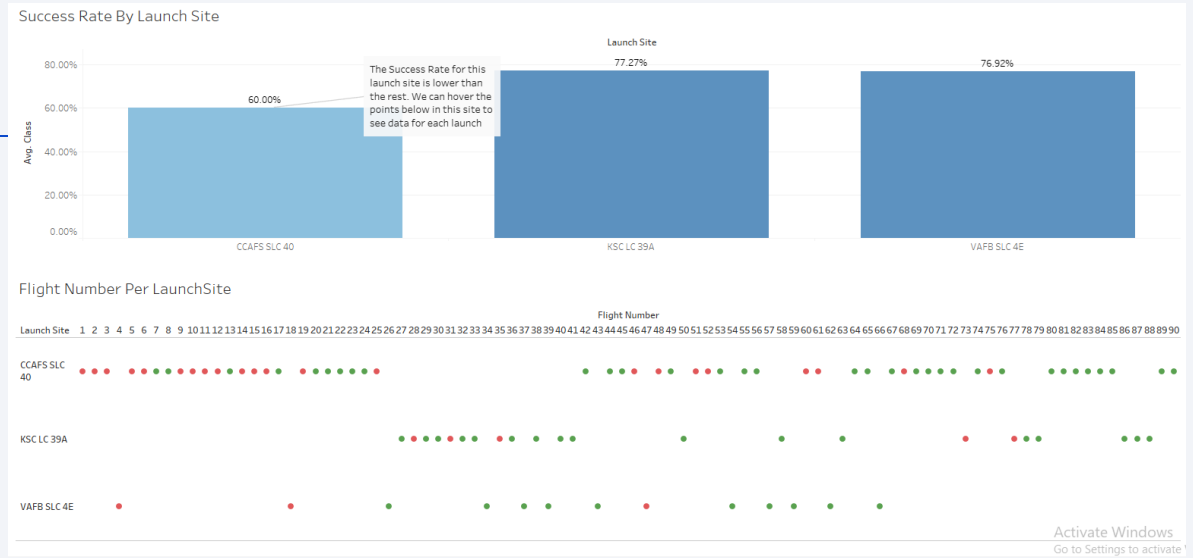
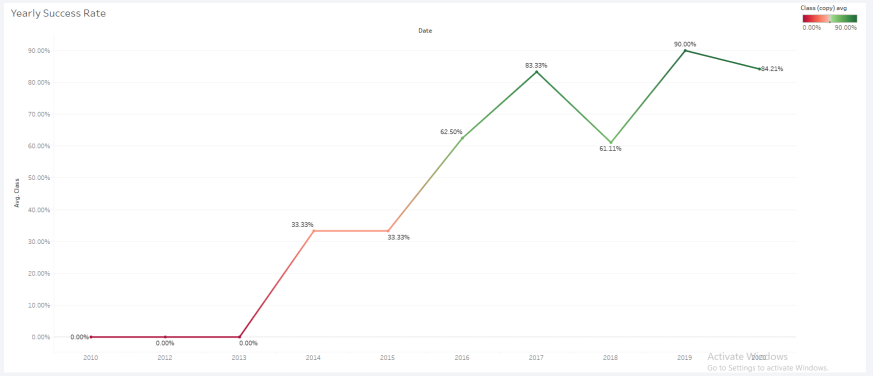
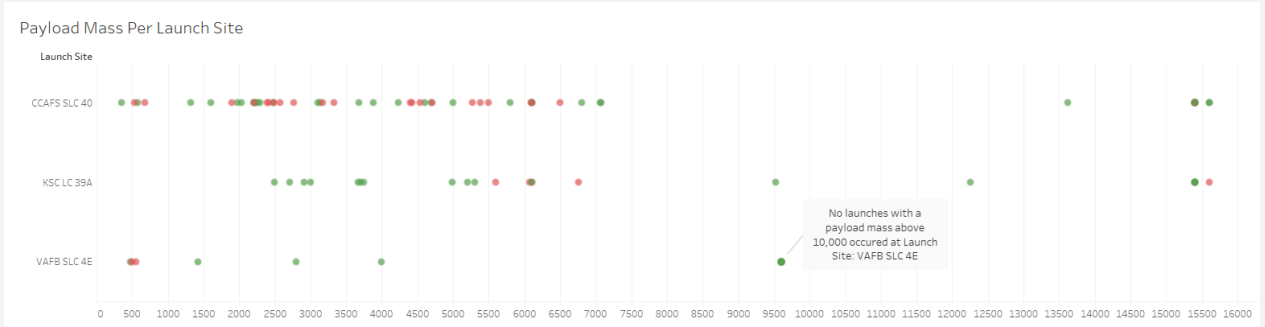


Tableau Dashboard

Link: https://public.tableau.com/app/profile/asim.shah5745/viz/SpaceXStageOneLandingFactorsSuccessRates/Story1?publish=yes





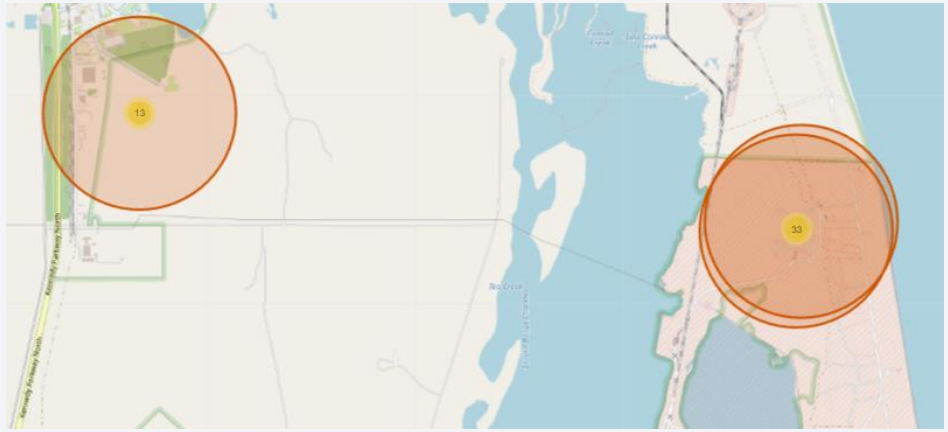
EDA with SQL

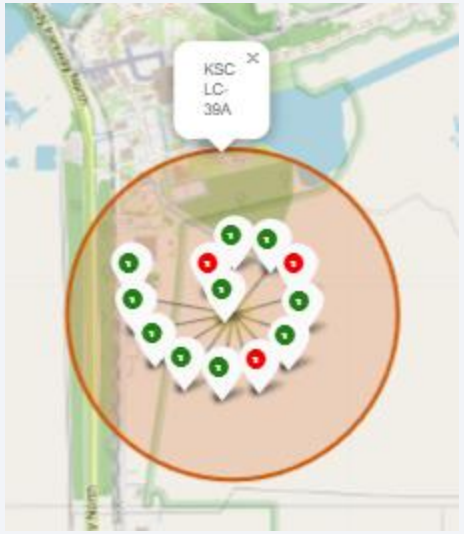
1. Display the names of the unique launch sites in the space mission
2. Display 5 records where launch sites begin with the string 'CCA'
3. Display the total payload mass carried by boosters launched by NASA (CRS)
4. Display average payload mass carried by booster version F9 v1.1
5. List the date when 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 landing outcomes (such as Failure (drone ship) or Success (ground pad)) between the date 2010-06-04 and 2017-03-20, in descending order.

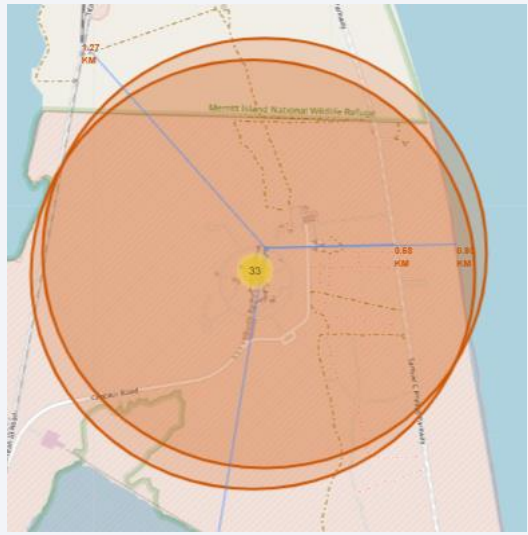
Git-hub Link: https://github.com/AsimAShah/SpaceX-Data-Science-Capstone-Project/blob/Main-Branch/EDA%20-%20SQL.ipynb

Build an Interactive Map with Folium

- We first got a subset of the data we are interested in such as Launch Site , Lat, Long, and Class (success/fail)
- We created a folium map and added our Launch Sites to the map
- We then created a marker color column whose values were green if class =1 and red if class=0
- We then created clusters on our map which represented the number of launches, as you zoom in you can see number of launches per site, and if you zoom on the launch site, you will see red and green tags for fail/success launches
- We then analyzed a Launch site, and plotted the nearest coastline, railway, highway, and city. We plotted lines connecting the launch site to the points and displayed the distance in KM







Git-hub Link: https://github.com/AsimAShah/SpaceX-Data-Science-Capstone-Project/blob/Main-Branch/Folium%20Maps%20-%20Geosp.%20Launch%20DataAnalysis.ipynb

Build a Dashboard with Plotly Dash

Plotly Dash App Features:

- We also added a drop down feature so the user can select ALL SITES or select an individual site to filter both charts
- We added a slider as well so users can adjust the payload range and drill down on how many successes/ failures are in each range

Scatter Chart:

- We created a scatter chart with class (success) percentage along the Y-axis and payload mass along the x axis. By doing this we can clearly see the points either at 0 % (Failures), or at %100 – (successes), along with how much payload mass for each point
- We also added a filter so you can filter the Scatter chart by Booster Version

By doing this, we were able to discover:

Which site has the largest successful launches? CCAFS LC-4

Which site has the highest launch success rate? CCAFS SLC-40

Which payload range(s) has the highest launch success rate?

- 2k to 4k (KG) 13 successes
- 4k-6k (KG) 5 successes

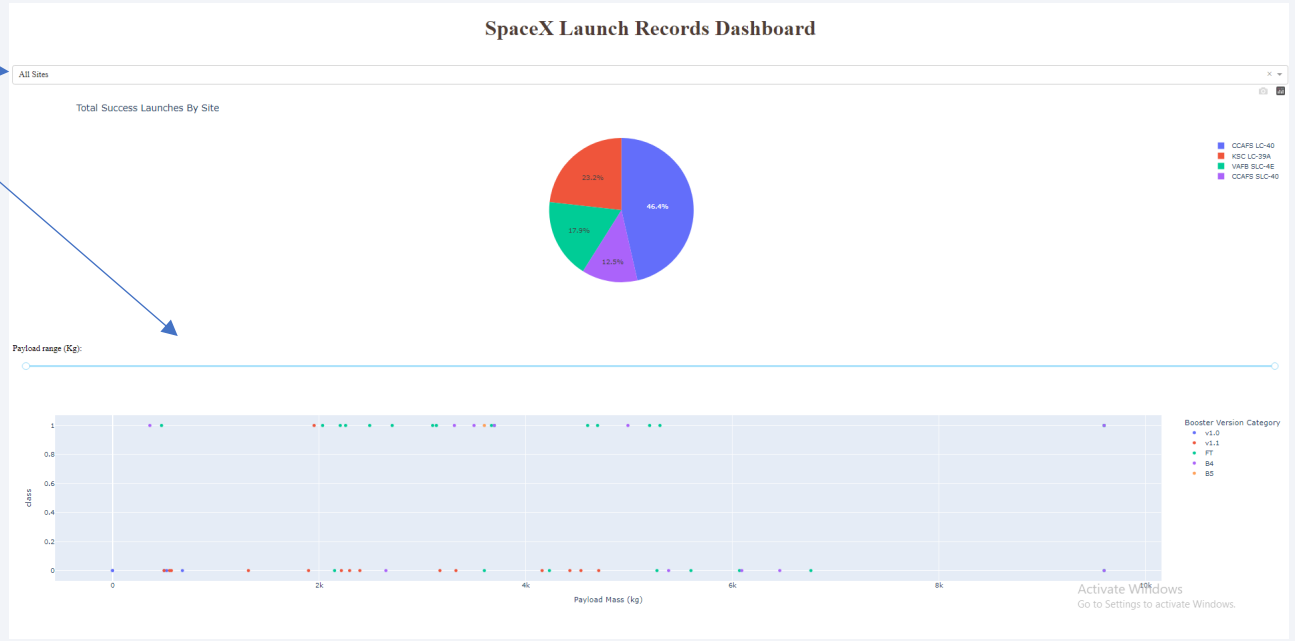
Which payload range(s) has the lowest launch success rate?

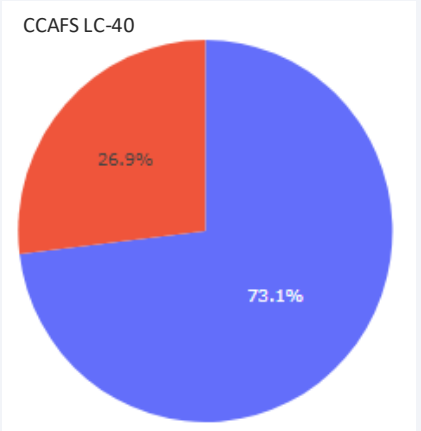
- 0k to 2k (KG) 8 fails
- 2k to 4k (KG) 8 fails
- 4k to 6k (KG) 8 fails

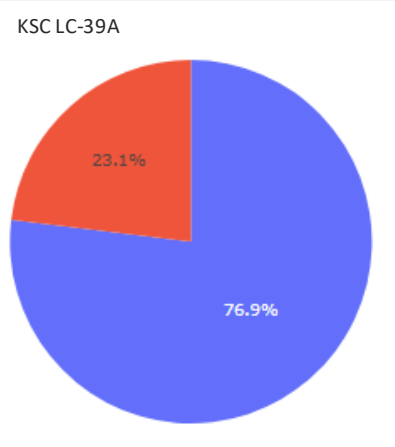
Which F9 Booster version (v1.0, v1.1, FT, B4, B5, etc.) has the highest launch success rate? Booster Version FT

Pie Chart :

- We can see from the pie chart and overall Launch Site success rate : CCAFS LC40 has the highest overall success rate at 46.4%
- When we select CCAFS LC-40 from the dropdown, we can also see the ratio for that site CCAFS LC-40 : 26.9% - Class 0 (Failure), 73.1% - Class 1 (success)
- When we select KSC LC-39A from the dropdown, we can see the ratio of that individual site: 23.1% - Class 0 (Failure), 76.9% - Class 1 (success)







Git-hub Link: https://github.com/AsimAShah/SpaceX-Interactive-Dash-App-w-Plotly

Predictive Analysis (Classification)

Built and Trained Logistic Regression, SVM, Decision Tree, and K-Nearest Neighbors Classification models using the optimal hyperparameters/Cross Validation & using evaluation metrics to see how well the models performed on training data & test data

- We first start by defining our X and Y (Output) variables
- We then use the StandardScalar class from the Sklearn library to perform a transformation on our X features by standardizing them and then applying them back our X object
- We then perform a Train Test Split on the data so we can see how well our models generalize to unseen data, we use a test_size of 0.2 (20%)
- **Logistic Regression** – create dictionary object with the hyperparameter values such as the penalty type & magnitude of the penalty. Create a logistic regression model object, create a GridSearch model object that contains: LR model object, parameters, number of folds for Cross Validation. Train the GridSearch object on the training data, obtain the 1. Best Parameters, 2. Cross Validation Score 3. Accuracy Score on Test Data
- We can then plot a confusion matrix and can see that we have many False Positives for our Logistic Regression Model

(Other common hyperparameters that we used for our models include hyperparameters such as tree depth for our Decision tree and the value of K in KNN)

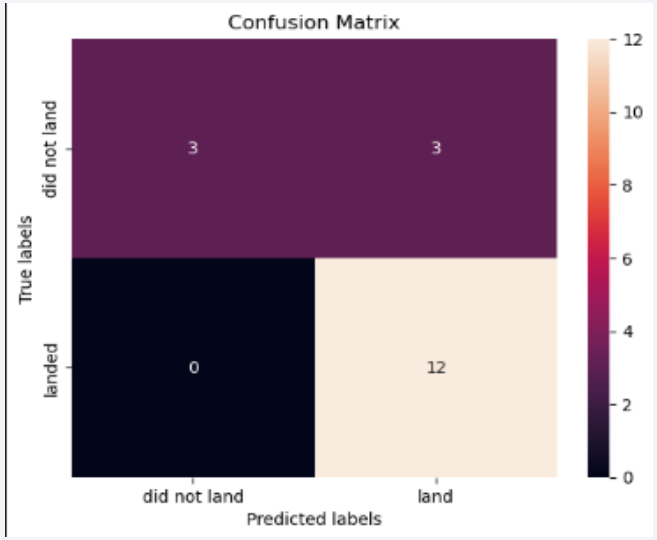
Other Models:

We then built KNN, SVM, and Decision Tree models as well and computed the metrics/CV scores for each. We also train each model will Hyperparameters that are specific to that Machine Learning Model in order to get the best combination of Hyperparameter's for each model.

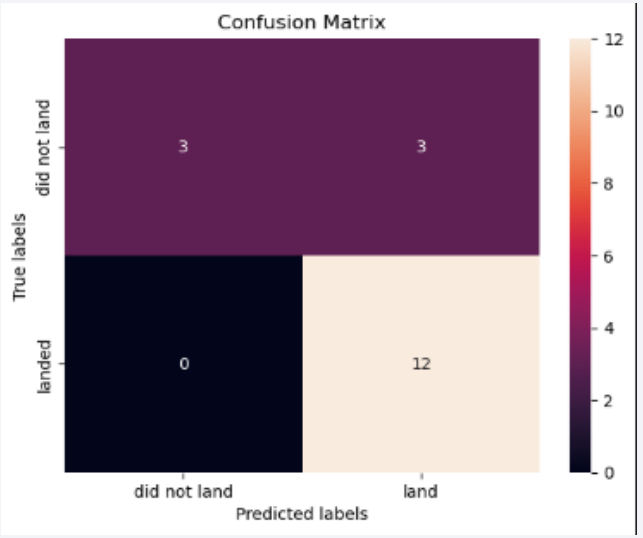
Git-hub Link: https://github.com/AsimAShah/SpaceX-Data-Science-Capstone-Project/blob/Main-Branch/SpaceX%20-%20Applying%20ML%20Classification%20Algos.ipynb

Predictive Analysis Results ...

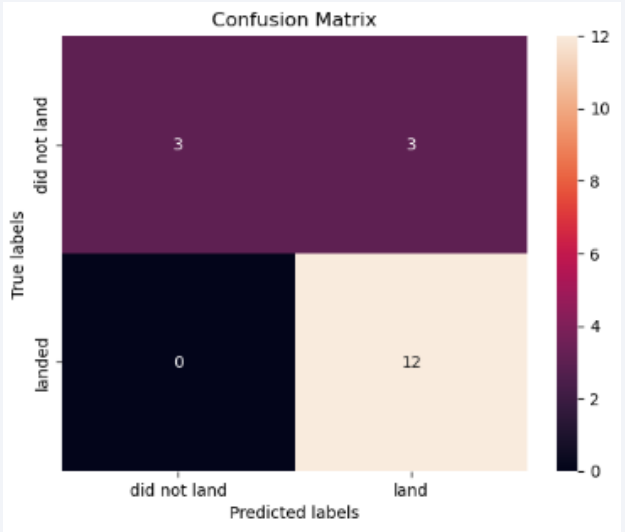
Logistic Regression Model:



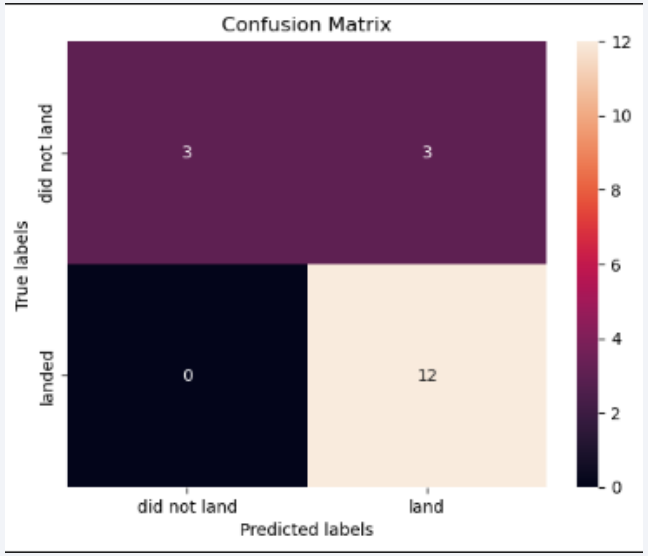
SVM Model:



Decision Tree:

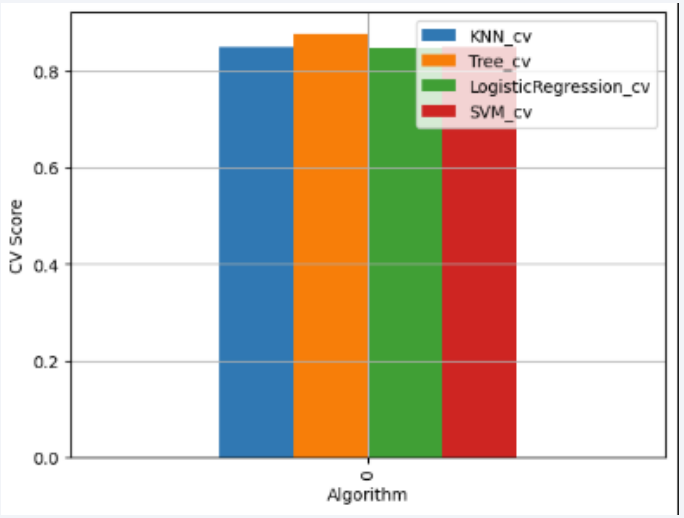


(KNN) K-Nearest Neighbor:

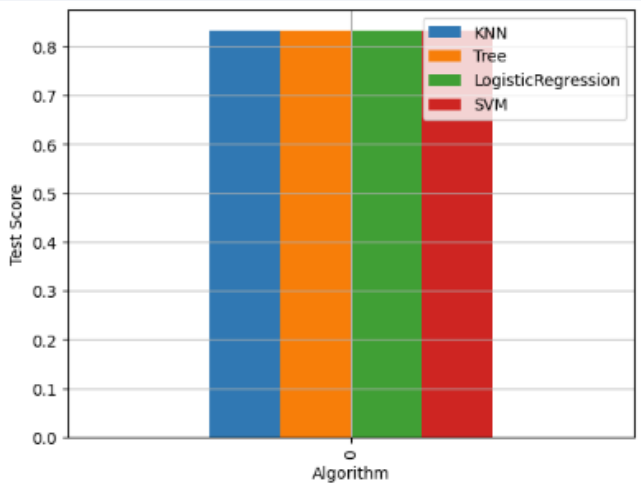


Predictive Analysis Results ...

- The model that resulted in the best **Cross Validation Score** was the Decision Tree Algorithm at 87.4% Accuracy



- However, when testing the **Accuracy** on the test data, all the models generalized similarly to the unseen data - **Accuracy** score of 83.3% for each model



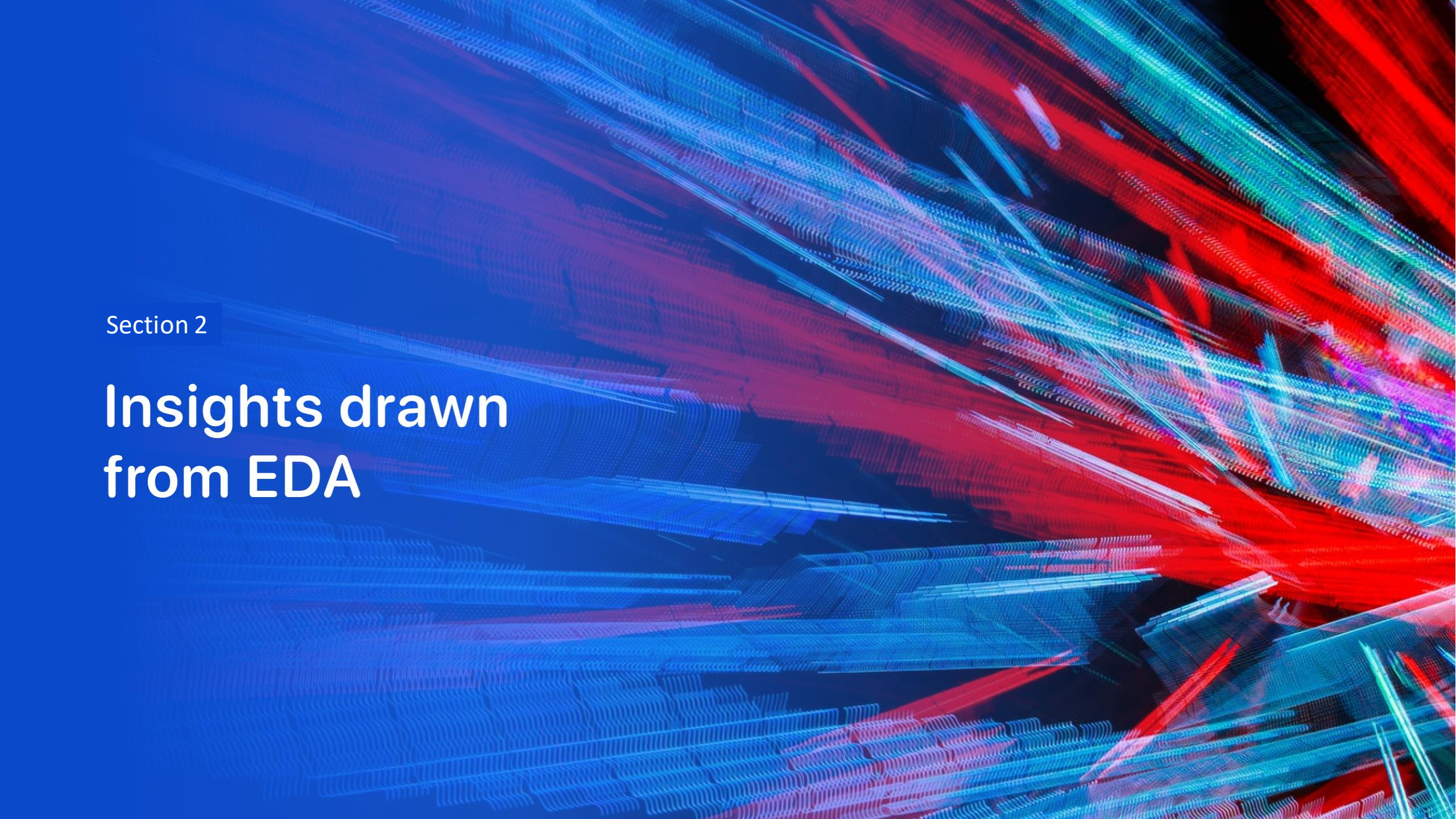
1. We can look at other metrics as well such as Precision, Recall, F1score, and AUC/ROC depending on what our ultimate goal is
2. Apart from performance metrics, other factors can influence the choice of a model, such as interpretability & computational complexity

Exploratory Data Analysis Results

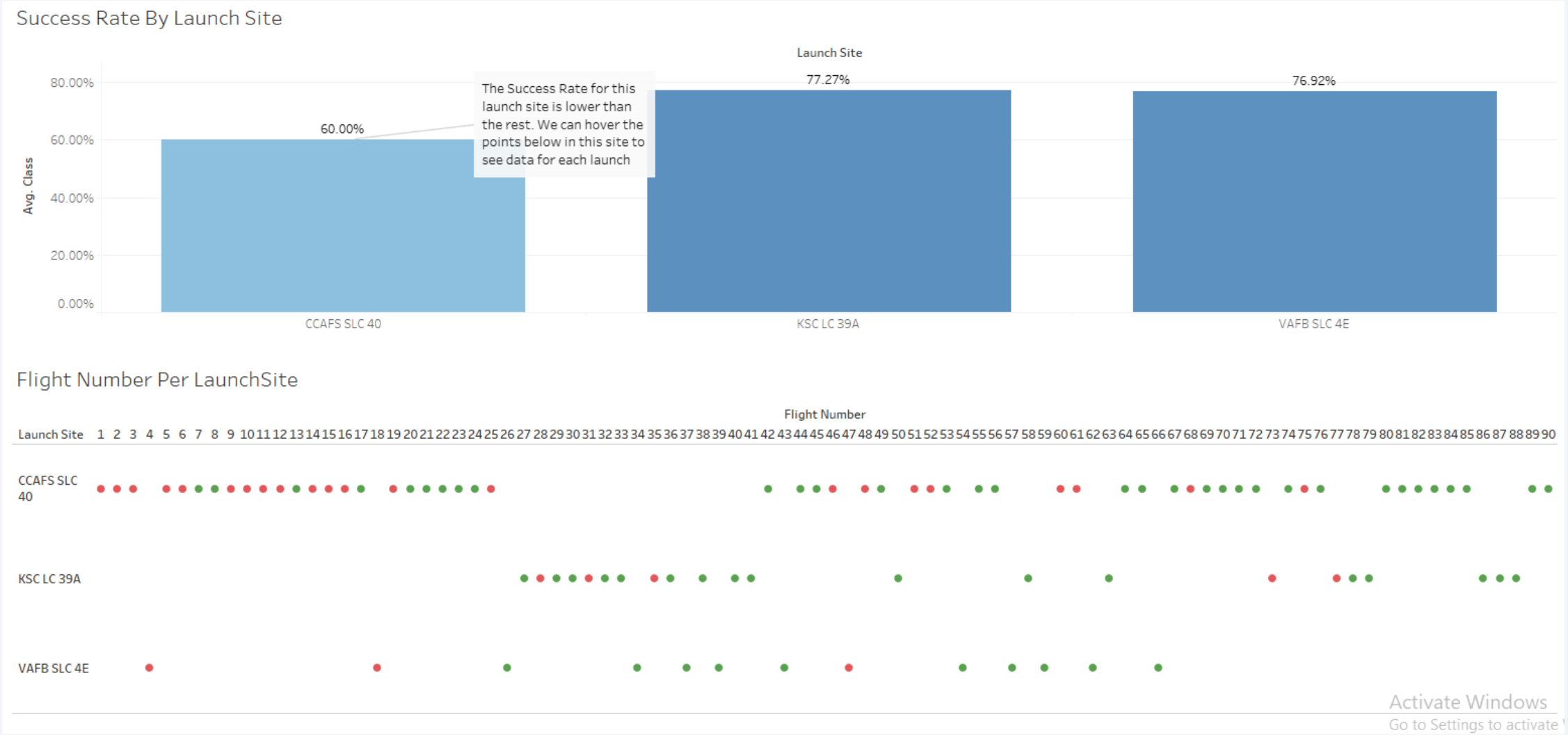
- Orbit types ES-l1, GEO, HEO, and SSO have the highest success rate
- Most of the successful launches have a payload mass of 10k (kg) or less
- Site CCAFS LC-40: has the most launches
- Site: VAFB SLC 4, we have no launches above 10k payload mass
- We noticed that the last 5 launches to Orbit LEO have been successful & PO and ISS orbits show trends that have a higher success rate for heavier payloads
- The success rate from 2010 – 2019 has been steadily increasing with a slight decrease in the year 2017 but then again increasing in 2019
- The success rate of the output column, or the avg of the Class column (success rate) is 66%
- Total payload mass carried by boosters launched by NASA (CRS) - 45596 KG
- Average payload mass carried by booster version F9 v1.1 - 2534.66 KG
- List the date when the first successful landing outcome in ground pad was achieved - 01/08/2018
- List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000 – F9 FT B1022, F9 FT B1026, F9 FT B1021.2, F9 FT B1031.2
- List the total number of successful and failure mission outcomes – 100 successes, 1 mission outcome failure

Section 2

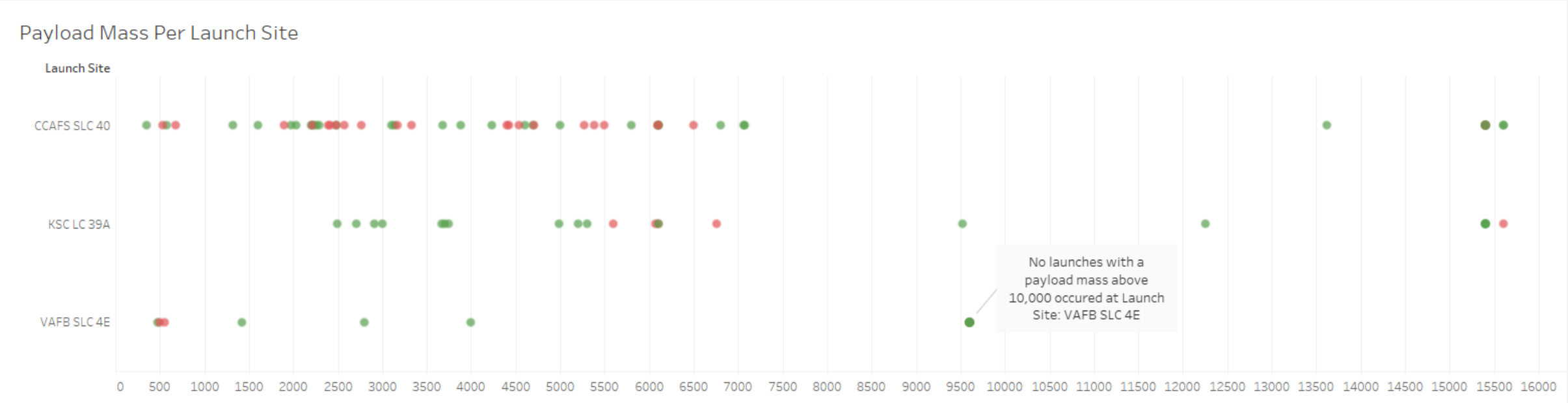
Insights drawn from EDA



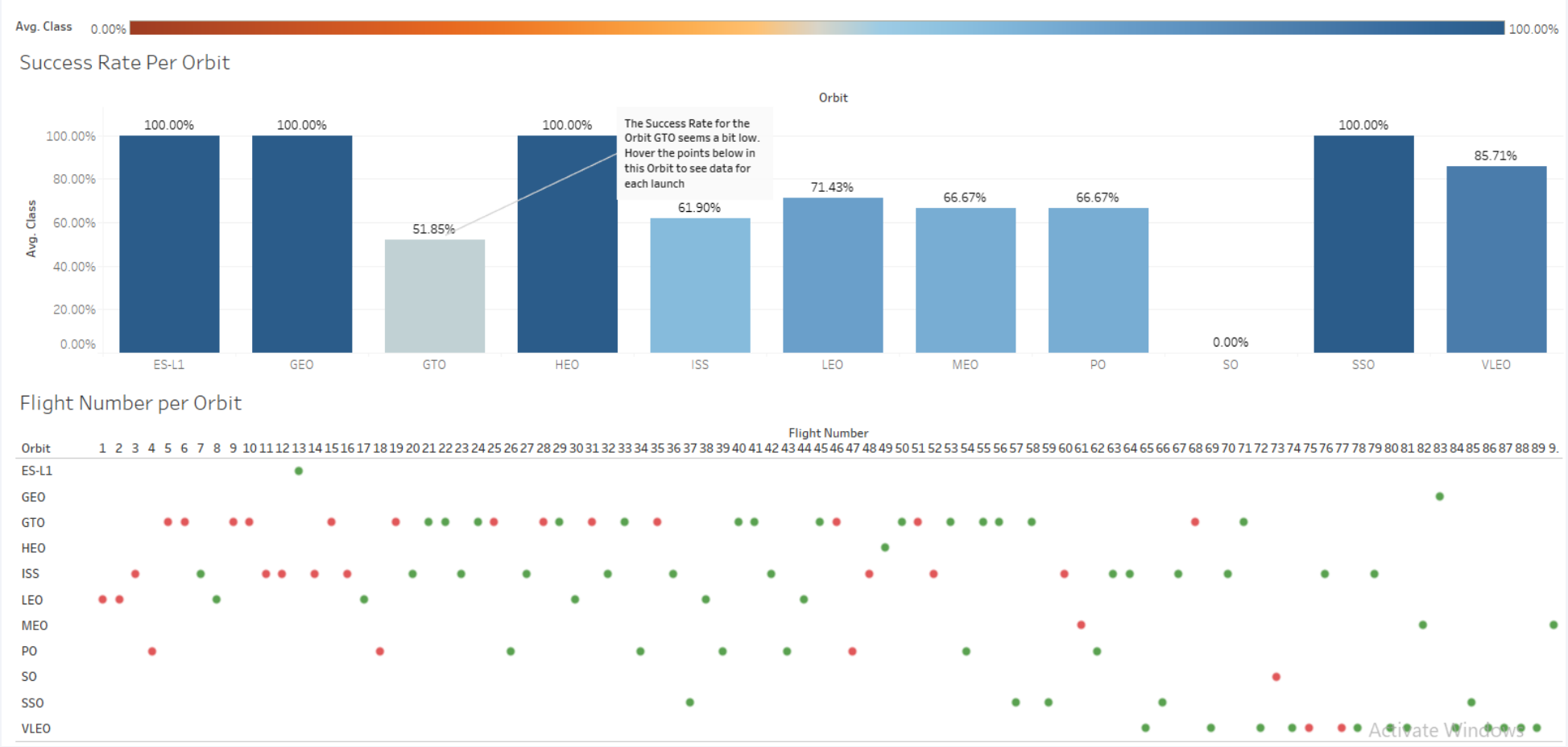
Flight Number vs. Launch Site



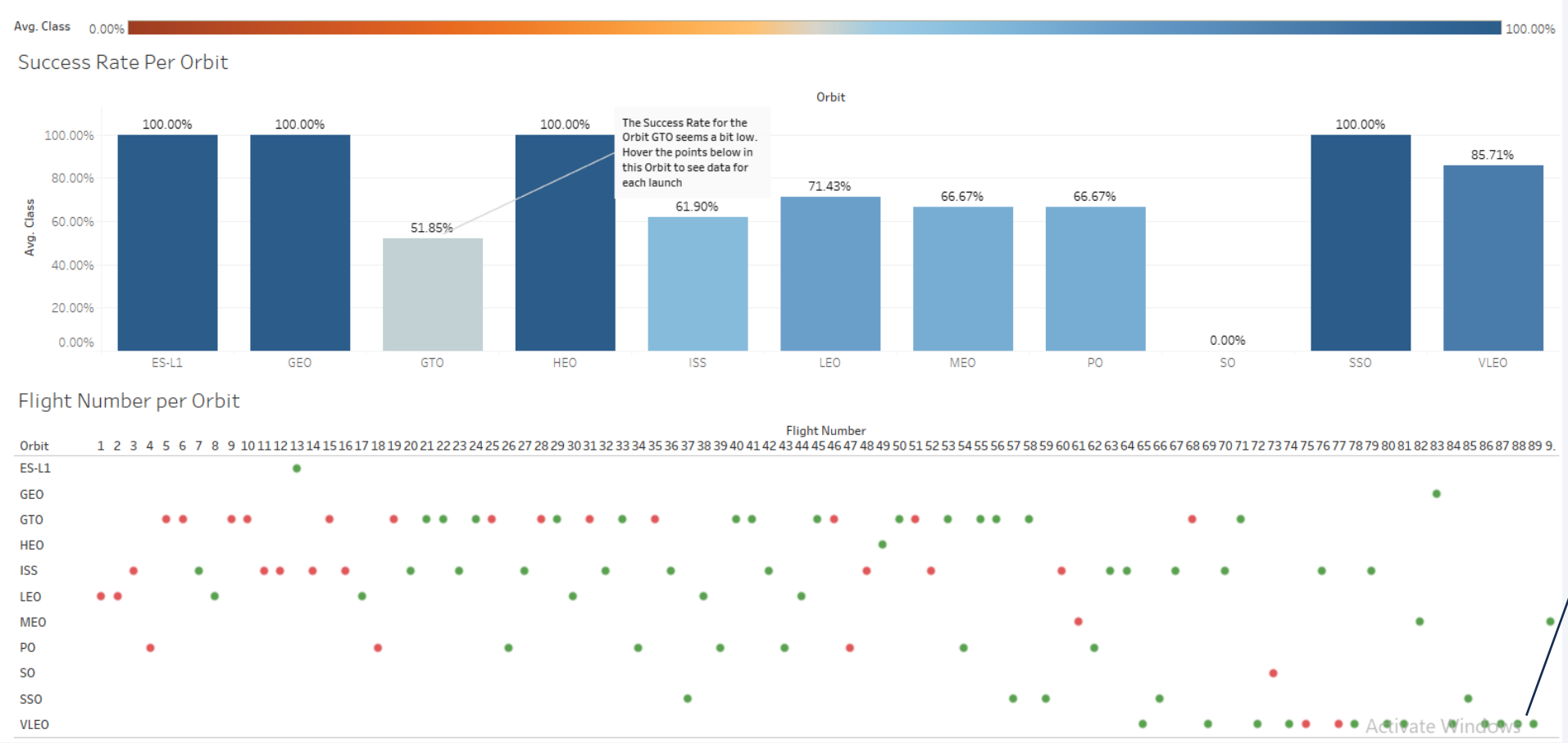
Payload vs. Launch Site



Success Rate vs. Orbit Type

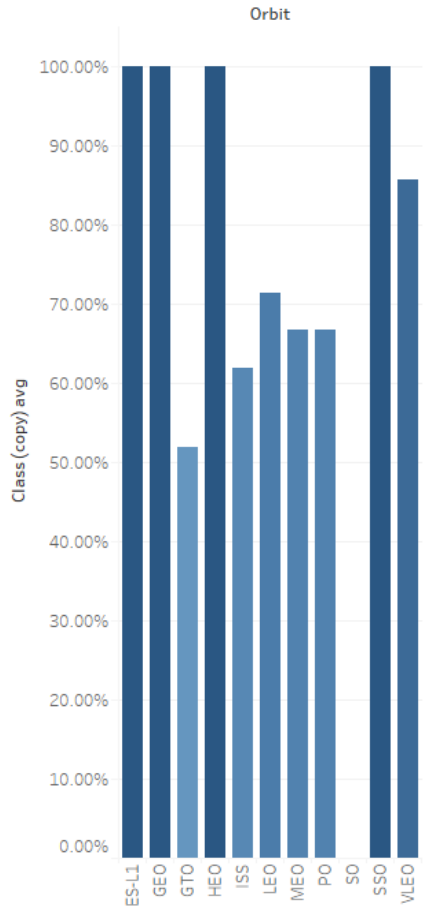


Flight Number vs. Orbit Type

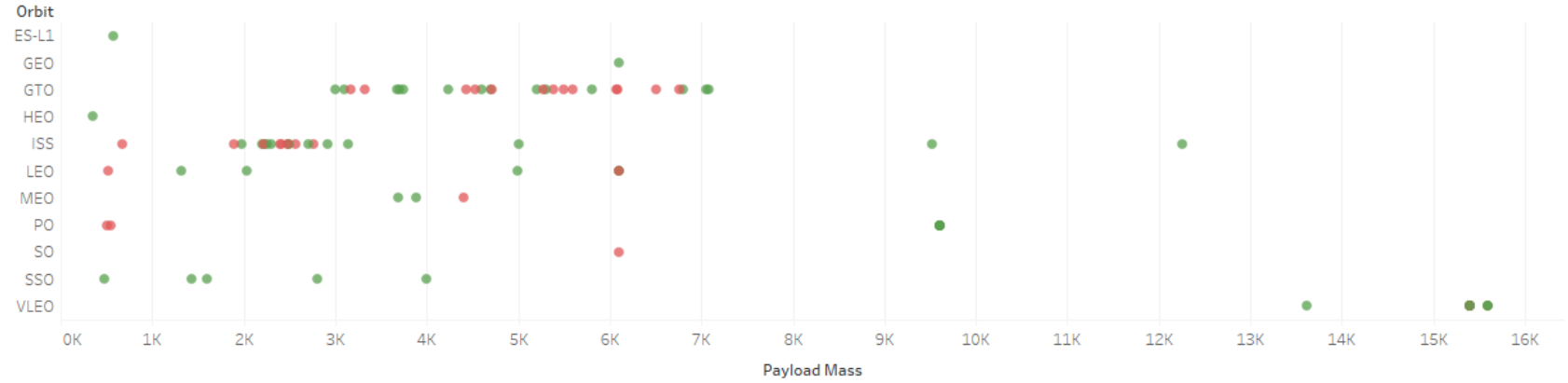


Payload vs. Orbit Type

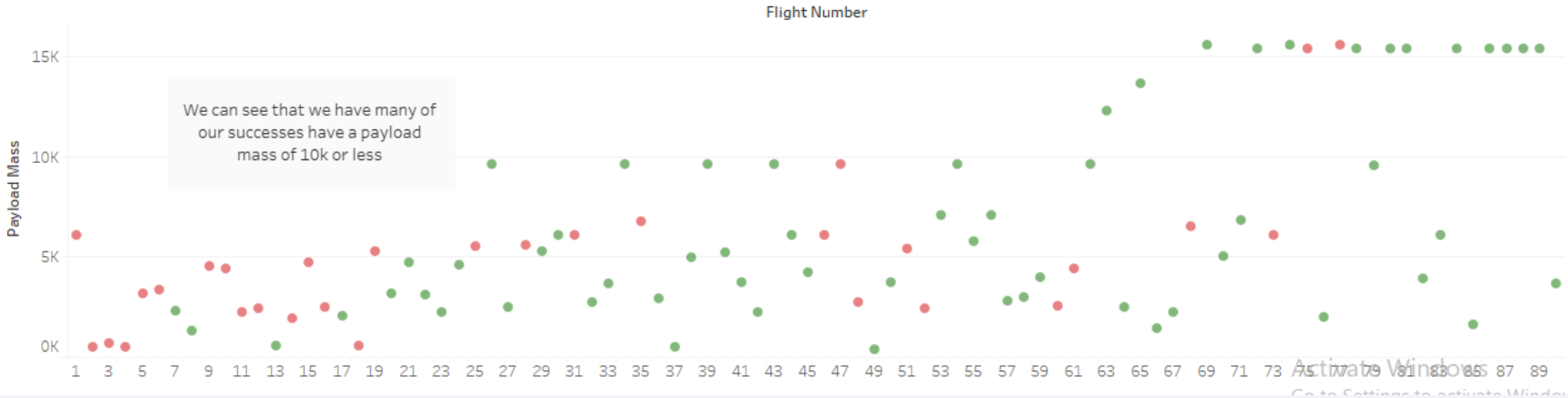
Orbit Success Rate



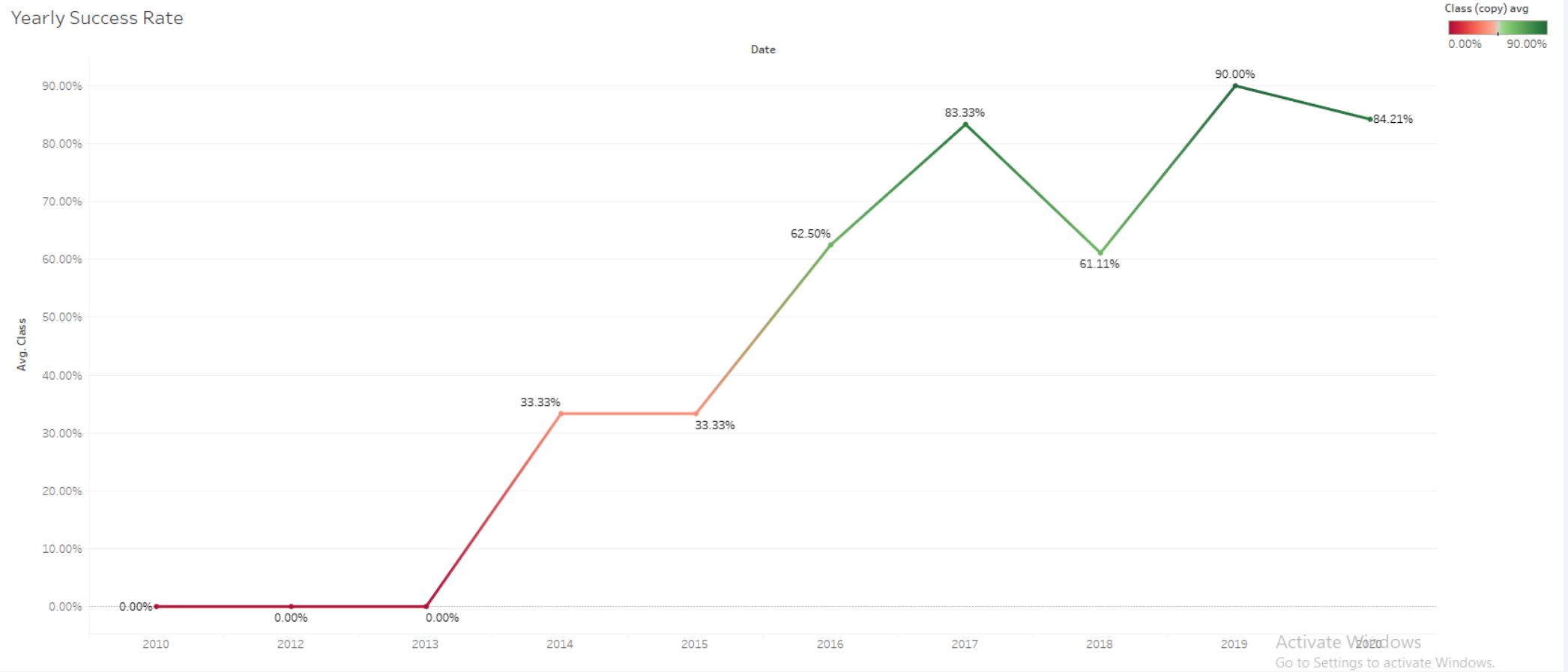
Payload Mass Per Orbit



Payload Mass Per Flight



Launch Success Yearly Trend



All Launch Site Names

- Find the names of the unique launch sites

Launch_Site
CCAFS LC-40
VAFB SLC-4E
KSC LC-39A
CCAFS SLC-40
None

Launch Site Names Begin with 'CCA'

- Find 5 records where launch sites begin with `CCA`

	Launch_Site
	CCAFS LC-40
	CCAFS LC-40
	CCAFS LC-40
	CCAFS LC-40
	CCAFS LC-40

Total Payload Mass

- Calculate the total payload carried by boosters from NASA

SUM(PAYLOAD_MASS__KG_)
45596.0

Average Payload Mass by F9 v1.1

- Calculate the average payload mass carried by booster version F9 v1.1

AVG(PAYLOAD_MASS__KG_)
2534.6666666666665

First Successful Ground Landing Date

- Find the dates of the first successful landing outcome on ground pad

MIN(Date)
01/08/2018

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

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

Total Number of Successful and Failure Mission Outcomes

- Calculate the total number of successful and failure mission outcomes

Total_Outcomes	Total_Successes	Total_Failures
101	100	1

Boosters Carried Maximum Payload

- List the names of the booster which have carried the maximum payload mass

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

2015 Launch Records

- List the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015

month	Landing_Outcome	Booster_Version	Launch_Site
10	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
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

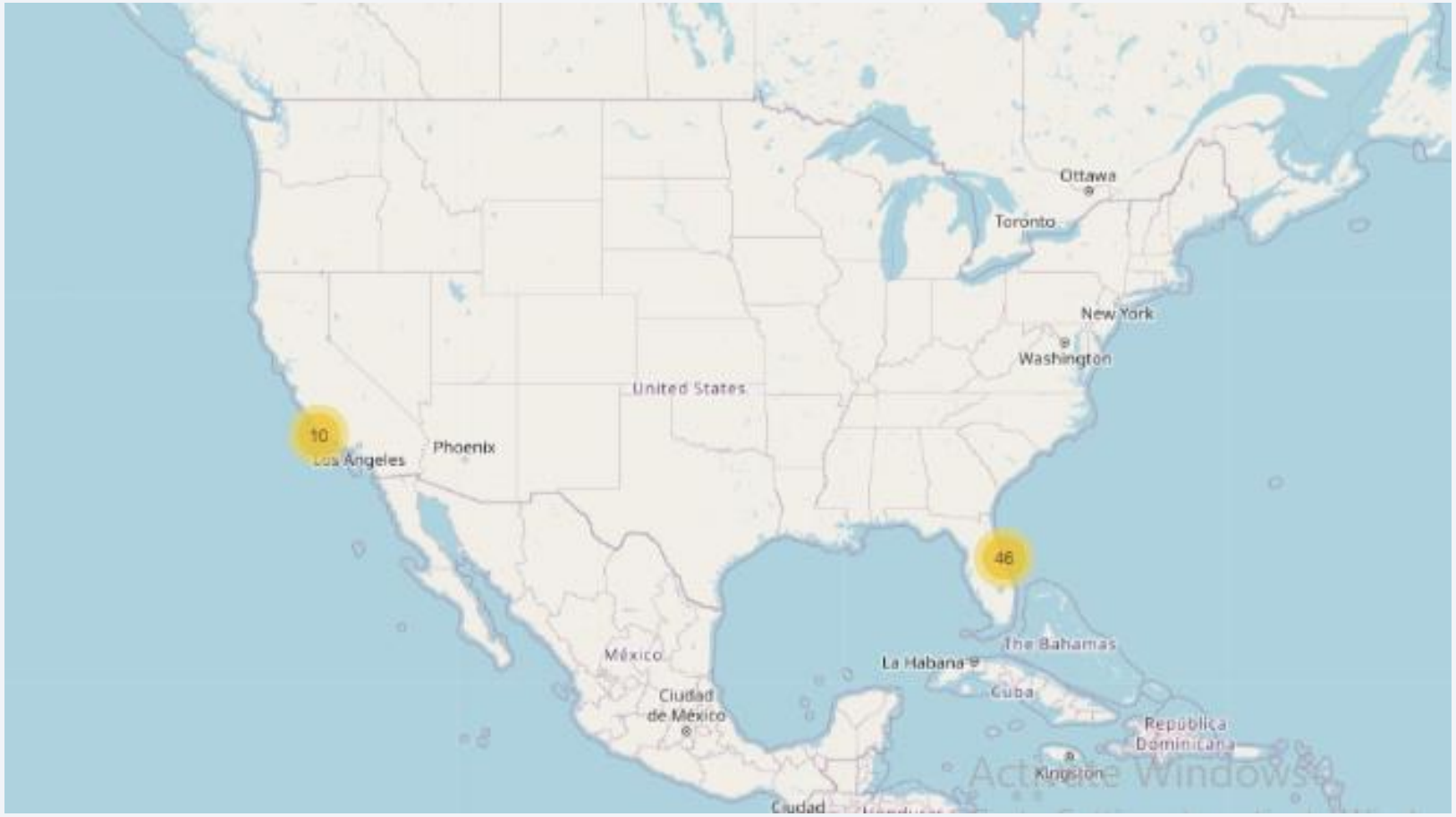
Landing_Outcome	Outcome_Count
No attempt	9
Failure (drone ship)	5
Success (drone ship)	4
Controlled (ocean)	3
Uncontrolled (ocean)	2
Success (ground pad)	2
Failure (parachute)	2
Precluded (drone ship)	1


Section 3
Launch Sites
Proximities Analysis

Maps with Folium: Clustering Data Points on a Map

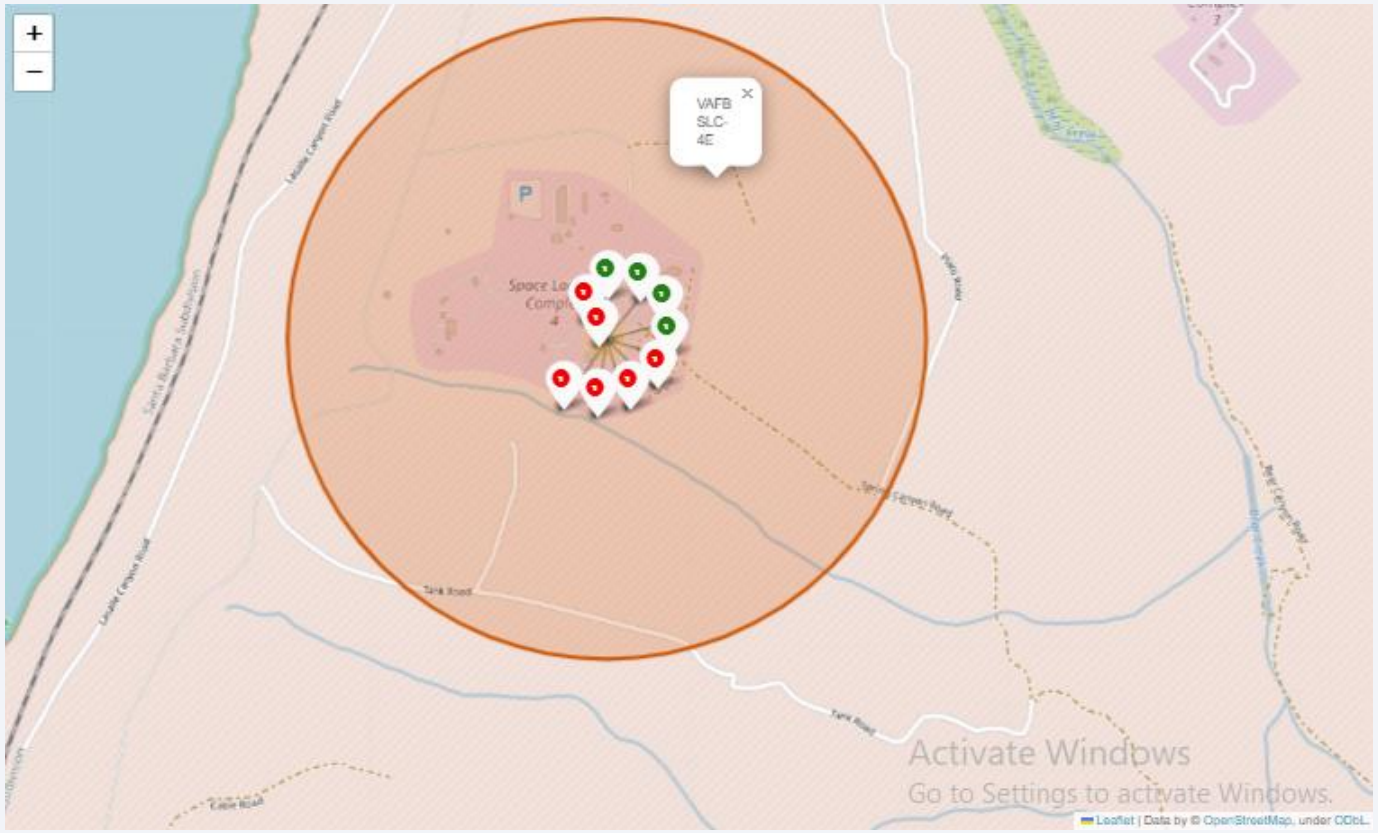
- We create clusters on our map, these clusters represent the number of launches , as you zoom in you can see number of launches per site, and if you zoom in on an individual launch site, you can see red and green tags for fail/success launches.

- As you can see we have 3 launch sites on the East coast: CCAFS LC-40,CCAFS SLC-40, KSC LC-39A & 1 one the west coast VAFB SLC-4E



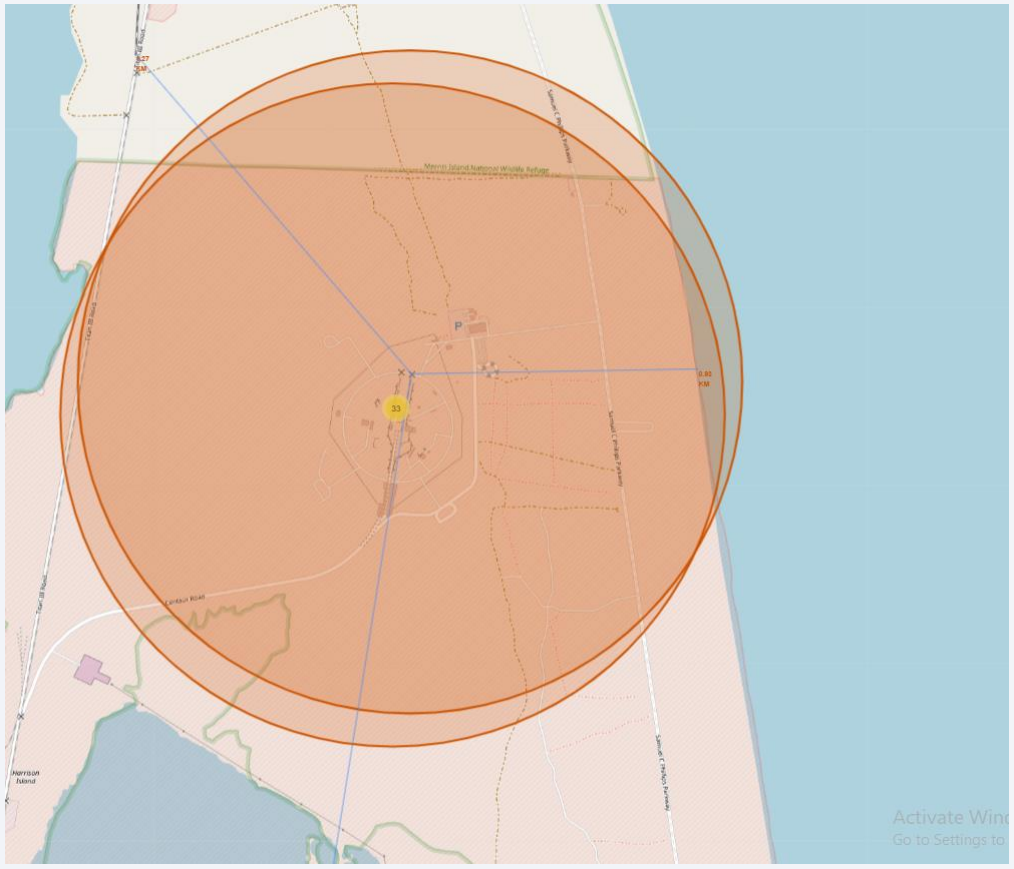
Maps with Folium: Plotting Successes/Failures

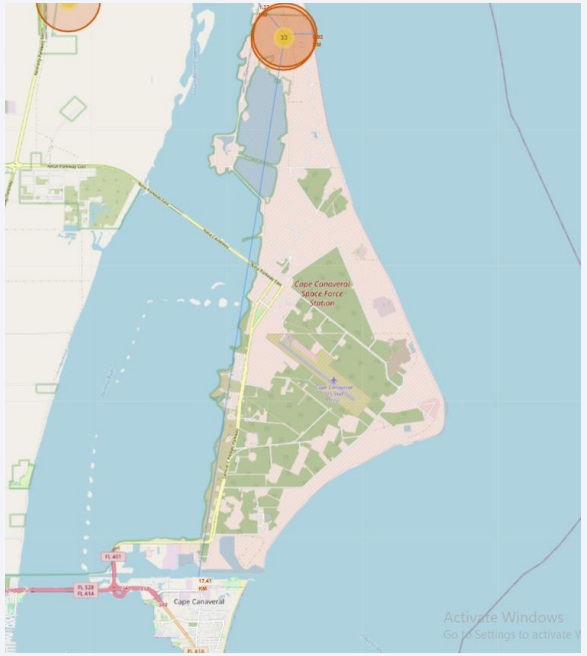
- Zooming in on an individual launch site allows you to see red and green tags for fail/success launches.

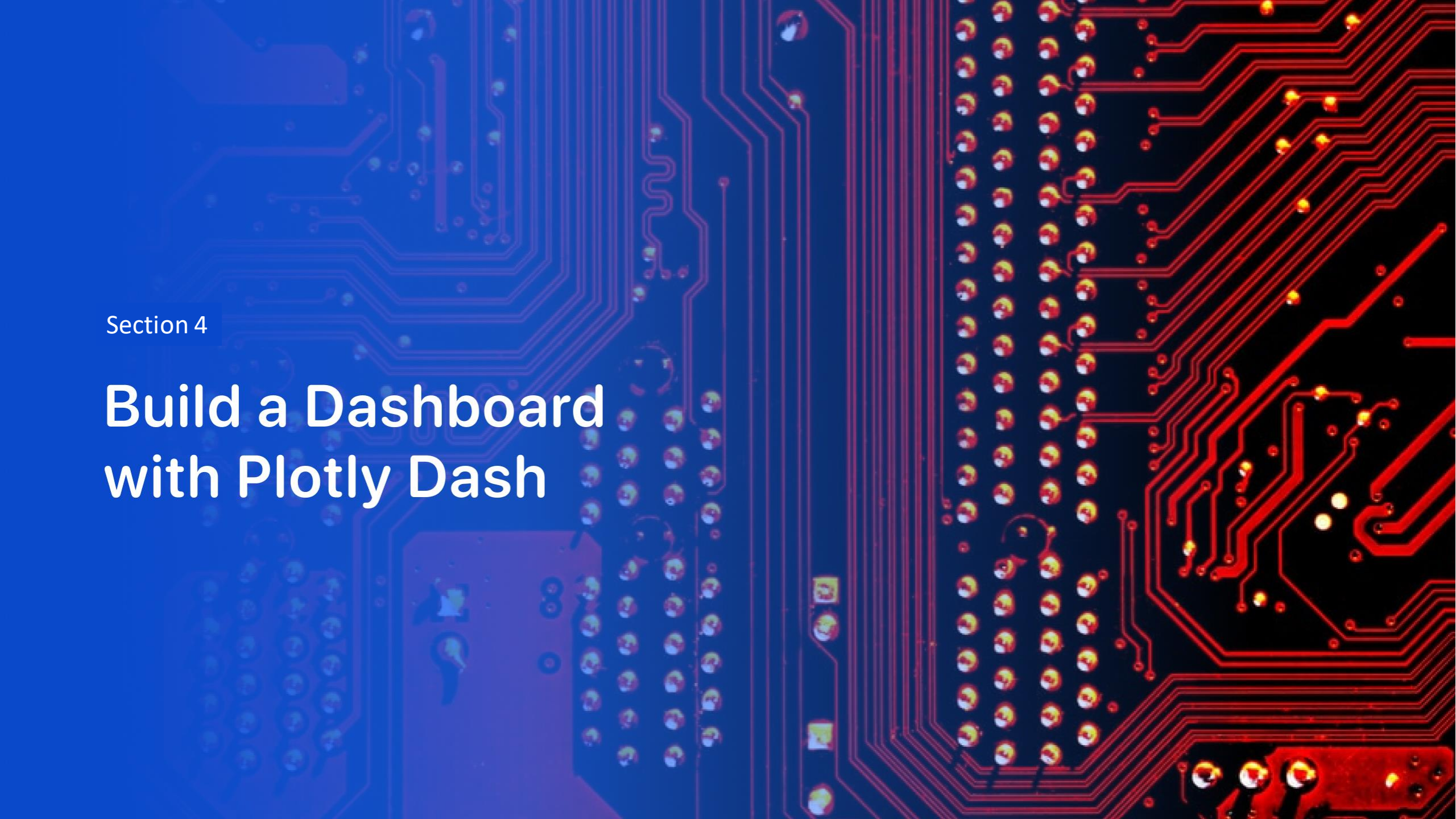


Maps with Folium: Plotting Successes/Failures

- We then analyzed a Launch site, and plotted the nearest coastline, railway, highway, and city. We plotted lines connecting the launch site to the points and displayed the distance in KM

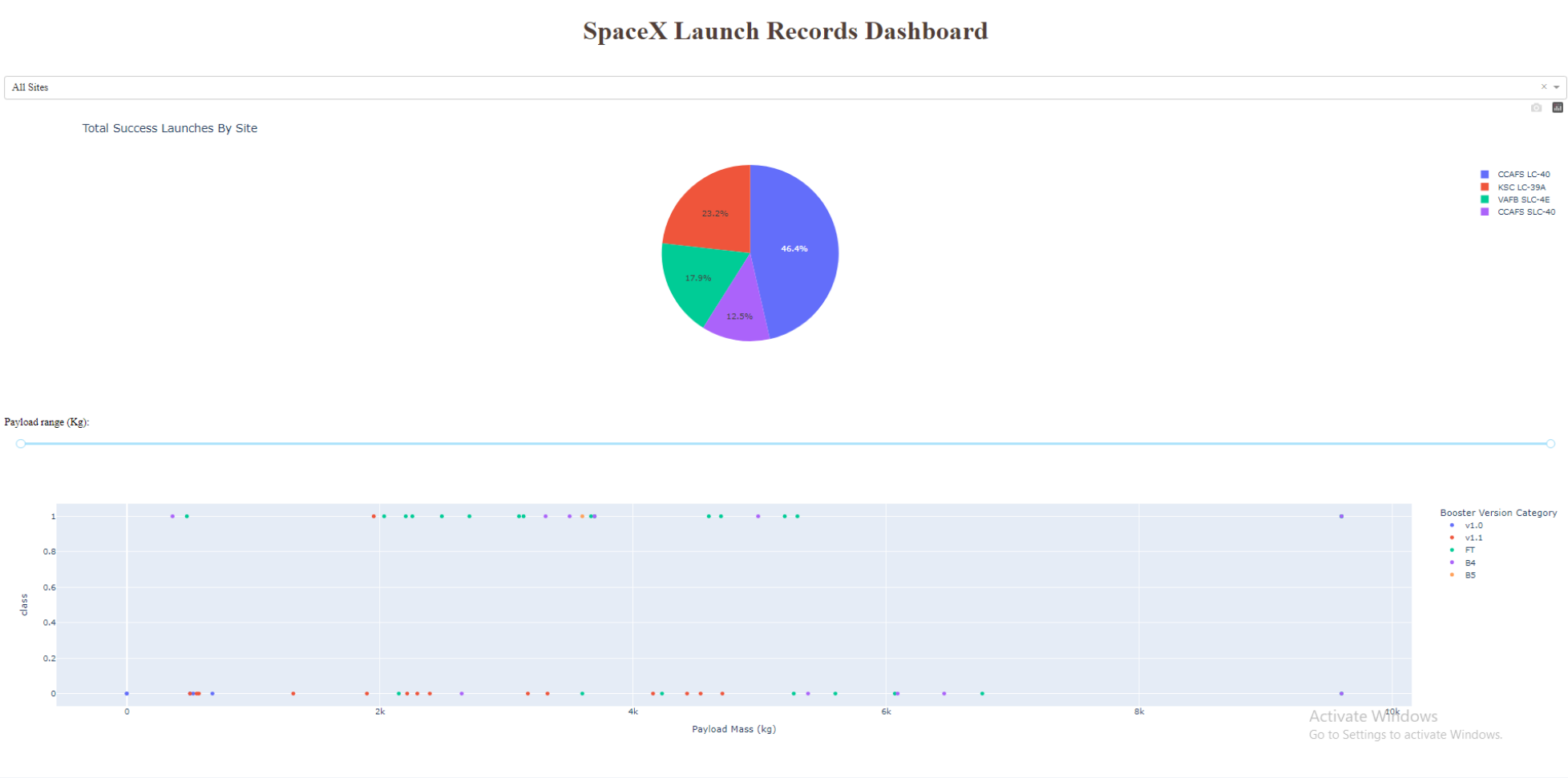





Section 4
Build a Dashboard with Plotly Dash

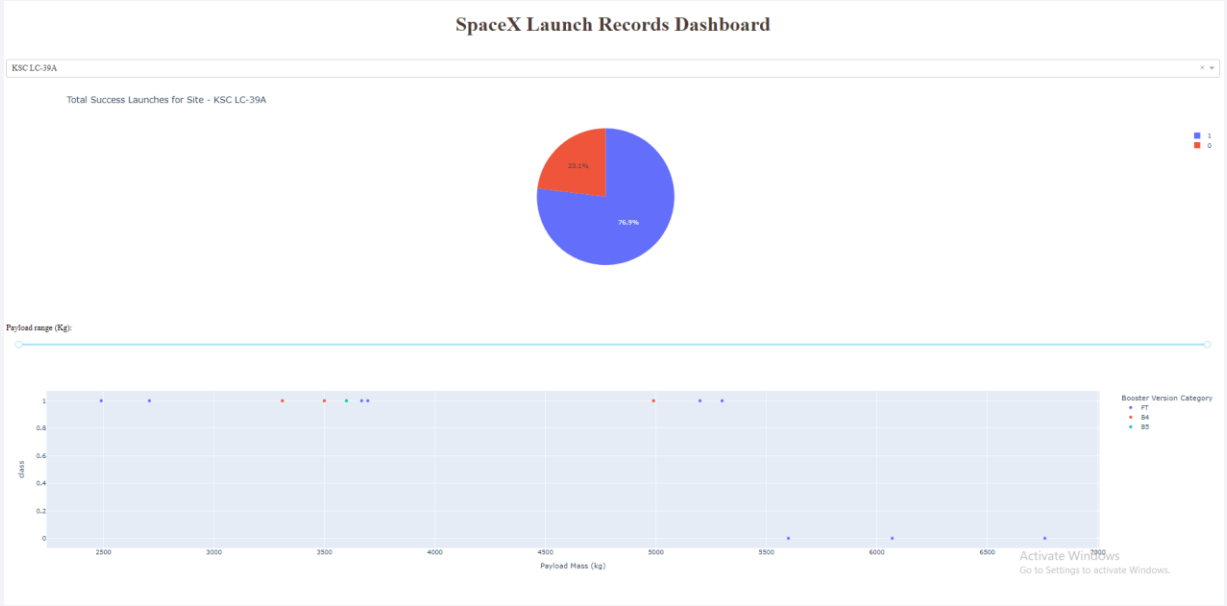
Interactive Web Dashboard using Plotly Dash

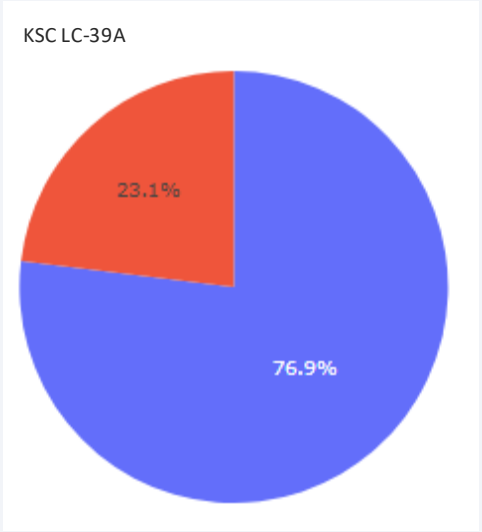
- From the overall Launch Site success rate : CCAFS LC-40 has the highest overall success rate at 46.4%

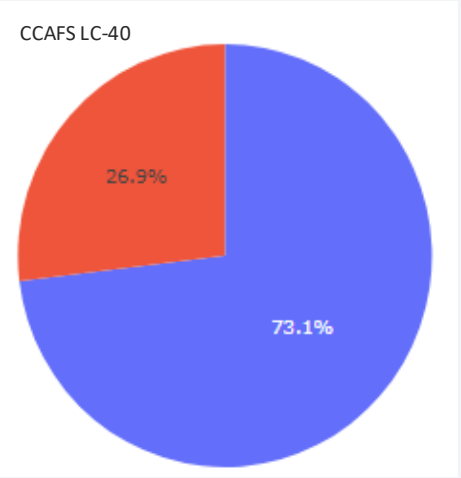


Interactive Web Dashboard using Plotly Dash

- Although CCAFS LC-40 has the highest overall success rate at 46.4%,
- KSC LC-39A has the highest individual success ratio, when we select KSC LC-39A from the dropdown, we can see the ratio in that individual site: 23.1% - Class 0 (Failure), 76.9% - Class 1 (success)







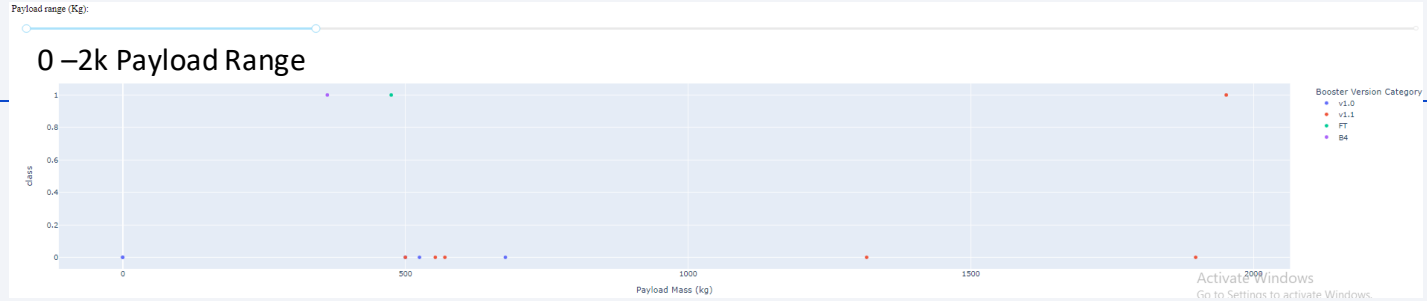
Interactive Web Dashboard using Plotly Dash

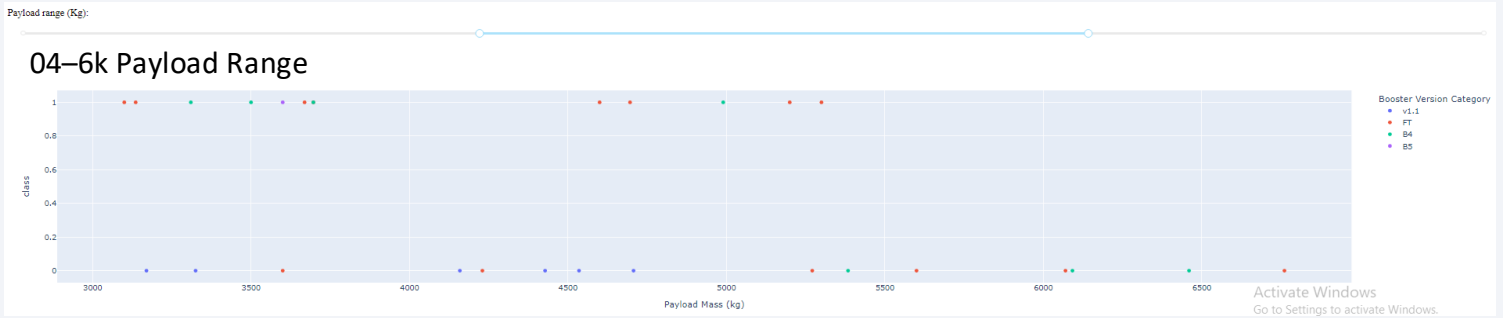
Which payload range(s) have the highest launch success rate?

- 2k to 4k (KG) 13 successes
- 4k-6k (KG) 5 successes

Which payload range(s) have the lowest launch success rate?

- 0k to 2k (KG) 8 fails
- 2k to 4k (KG) 8 fails
- 4k to 6k (KG) 8 fails



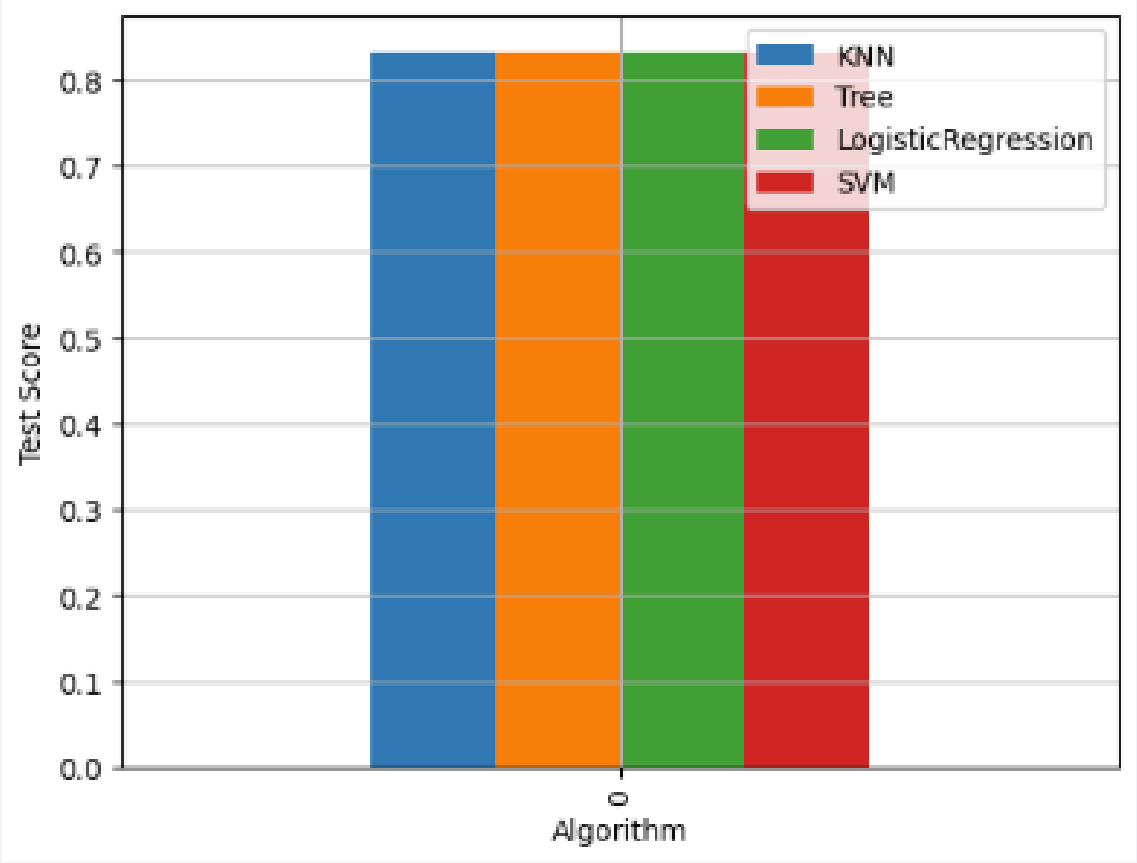


Section 5

Predictive Analysis (Classification)

Classification Accuracy

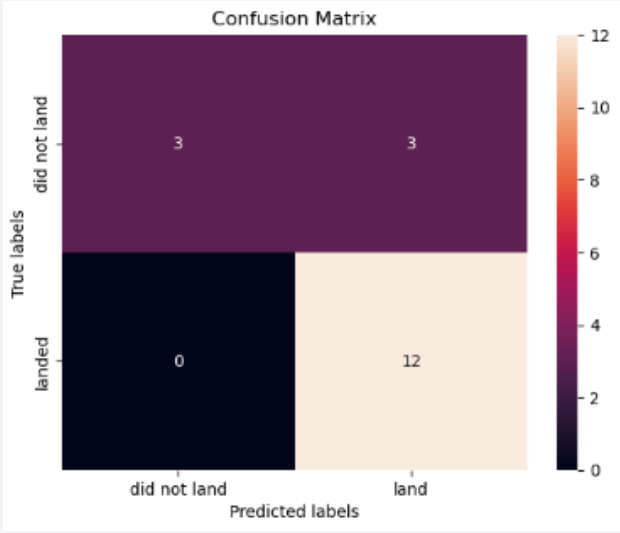
- We can see that in terms of the classification **Accuracy**, all of these models performed similarly on the Test set at about 83.3%

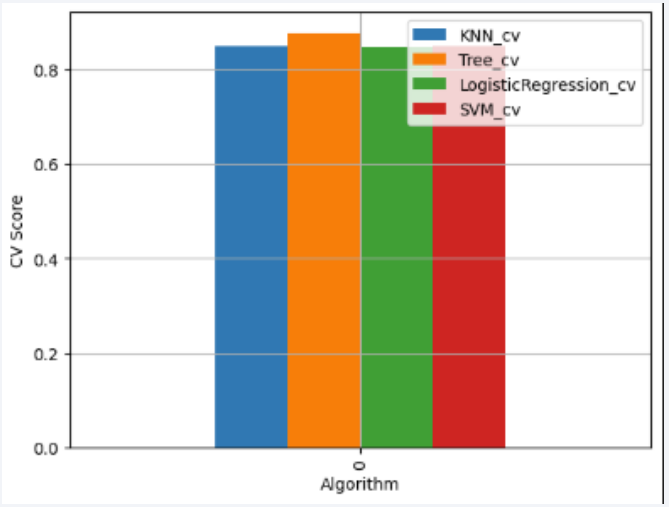


Confusion Matrix

- Upon observing the accuracy on the test sets of these classification models, we can see that they all generalized similarly
- The model which had the best Cross Validation score however, was the Decision Tree
- We can also look at other metrics depending on our goal such as F1-score, precision, recall, AUC/ROC
- Other factors when considering a model can include interpretability, computation speed, scalability (can the model handle large data sets?), Cost, implementation/deployment, etc.

Decision Tree Confusion Matrix:





Appendix

- Data Collection - Git-hub Link: https://github.com/AsimAShah/SpaceX-Data-Science-Capstone-Project/blob/Main-Branch/Data%20Collection%20-%20SpaceX%20API's.ipynb
- Web Scraping - Git-hub Link: https://github.com/AsimAShah/SpaceX-Data-Science-Capstone-Project/blob/Main-Branch/WebScraping%20wiki%20w%20Beautiful%20Soup.ipynb
- Data Wrangling - Git-hub Link: https://github.com/AsimAShah/SpaceX-Data-Science-Capstone-Project/blob/Main-Branch/Data%20Wrangling.ipynb
- EDA using Visualization - Git-hub Link: https://github.com/AsimAShah/SpaceX-Data-Science-Capstone-Project/blob/Main-Branch/EDA%20-%20Data%20Vizualizations.ipynb
- EDA using SQL Git-hub Link: https://github.com/AsimAShah/SpaceX-Data-Science-Capstone-Project/blob/Main-Branch/EDA%20-%20SQL.ipynb
- Interactive Folium Map - Git-hub Link: https://github.com/AsimAShah/SpaceX-Data-Science-Capstone-Project/blob/Main-Branch/Folium%20Maps%20-%20Geosp.%20Launch%20DataAnalysis.ipynb
- Applying ML Algorithms - Git-hub Link: https://github.com/AsimAShah/SpaceX-Data-Science-Capstone-Project/blob/Main-Branch/SpaceX%20-%20Applying%20ML%20Classification%20Algos.ipynb
- Interactive Plotly Dash App - Git-hub Link: https://github.com/AsimAShah/SpaceX-Interactive-Dash-App-w-Plotly
- Tableau
  Dashboard Link: https://public.tableau.com/app/profile/asim.shah5745/viz/SpaceXStageOneLandingFactorsSuccessRates/Story1?publish=yes

Thank you!