IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

<Yassin Mahmoud>
<4/27/2025>
<https://github.com/yassinm05/Space-Y-Project>

Outline

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

Executive Summary

- Summary of methodologies
  - We collected rocket launch data from SpaceX API
  - We performed web scraping to collect Falcon 9 historical launch records from a Wikipedia page titled List of Falcon 9 and Falcon Heavy launches
  - We performed some Exploratory Data Analysis (EDA) to find some patterns in the data using:
    - Pandas
    - Matplotlib
    - SQL
  - Interactive visual We performed analytics using folium and Plotly Dash
- Summary of all results
  - We performed Machine Learning Prediction to predict if the first stage will land

Introduction

- Project background and context
  - I will be working as a Data Scientist for SpaceY company Founded by Allon Mask
  - 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
  - determine the price of each launch using SpaceX data.
  - determine wether or not SpaceX will use the first stage it's rocket.
  - Predict the wether the the first stage will land

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Data was collected using SpaceX API and Pandas.
  - Data was using Web Scraping to collect Falcon 9 historical launch records from a Wikipedia page titled List of Falcon 9 and Falcon Heavy launches.
- Perform data wrangling
  - finding some patterns in the data to determine what would be the label for training supervised models.
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - We built the classification models using scikitlearn and used gridsearch to determine the best model.

Data Collection

- We use SpaceX REST API for for launch, rocket, core, capsule, starlink, launchpad, and landing pad data. https://api.spacexdata.com/v4/

- But a lot of the data are IDs. For example the rocket column has no information about the rocket just an identification number. use the API again to get information about the launches using the IDs given for each launch. Specifically we will be using
columns rocket, payloads, launchpad, and cores. Using specific functions Then add it to the Data set
  - rocket column https://api.spacexdata.com/v4/rockets/
  - launchpad column https://api.spacexdata.com/v4/launchpads/
  - payloads column https://api.spacexdata.com/v4/payloads/
  - cores column https://api.spacexdata.com/v4/cores/

- We web scraping to collect Falcon 9 historical launch records from a Wikipedia page titled List of Falcon 9 and Falcon Heavy launches

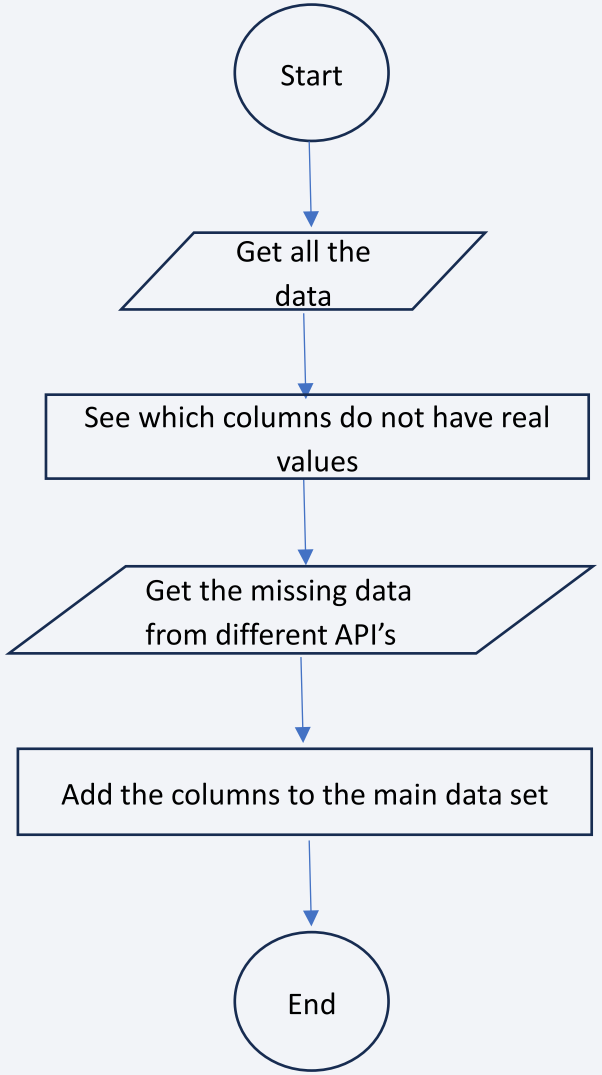
Data Collection – SpaceX API

- Get the whole Data
  https://api.spacexdata.com/v4/
- Get the rocket column
  https://api.spacexdata.com/v4/rockets/
- Get the launchpad column
  https://api.spacexdata.com/v4/launchpads/
- Get the payloads column
  https://api.spacexdata.com/v4/payloads/

- Get the payloads column
  https://api.spacexdata.com/v4/cores/

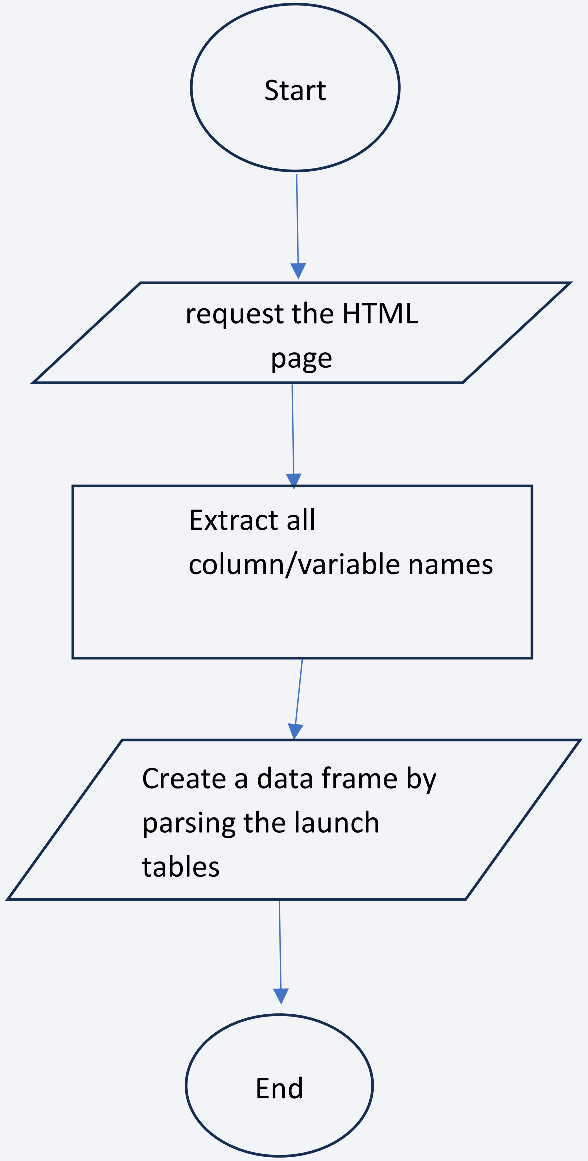
- Replace the column that have ID's only in the main Data set with the ones we got from different API

https://github.com/yassinm05/Sapace-Y-Project/blob/main/Data%20collection%20using%20api.ipynb



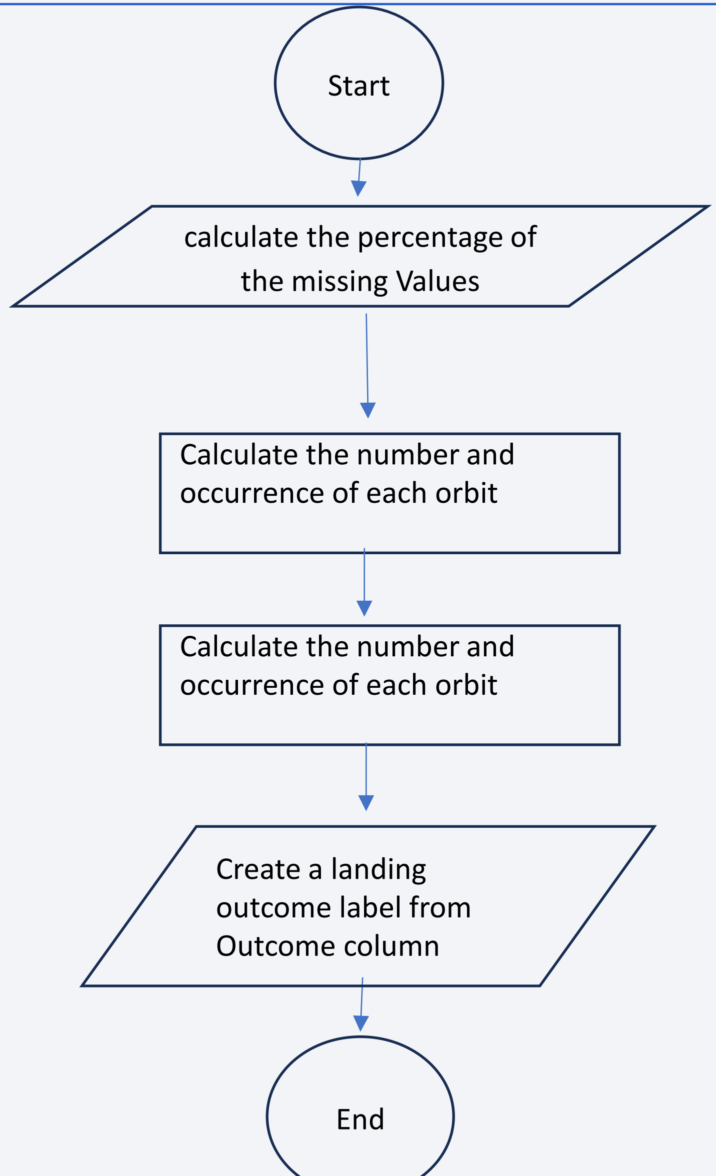
Data Collection - Scraping

- request the HTML page from the URL <https://en.wikipedia.org/w/index.php?title=List_of_Falcon_9_and_Falcon_Heavy_launches&oldid=1027686922>
- Extract all column/variable names
- Create a data frame by parsing the launch tables
- https://github.com/yassinm05/Space-Y-Project/blob/main/Data%20collection%20using%20webscrapping.ipynb



Data Wrangling

- We start Identify and calculate the percentage of the missing values in each attribute
- Calculate the number and occurrence of each orbit
- Calculate the number and occurence of mission outcome of the orbits
- Create a landing outcome label from Outcome column
- We will use this as the target variable we want to predict
- https://github.com/yassinm05/Space-Y-Project/blob/main/Data%20wrangling.ipynb



EDA with Data Visualization

- a scatter point chart to see how the FlightNumber (indicating the continuous launch attempts.) and Payload variables would affect the launch outcome.
- a scatter point chart to Visualize the relationship between Flight Number and Launch Site
- a scatter point chart to Visualize the relationship between Payload Mass and Launch Site
- a bar chart Visualize the relationship between success rate of each orbit type a bar chart
- a scatter point chart to Visualize the relationship between FlightNumber and Orbit type
- a scatter point chart to Visualize the relationship between Payload Mass and Orbit type
- Line chart to Visualize the launch success yearly trend
- apply OneHotEncoder to the Categorical columns
- <https://github.com/yassinm05/Space-Y-Project/blob/main/EDA%20using%20Pandas%20and%20matplotlib.ipynb>

EDA with SQL

- %sql select * from SPACEXTABLE group by "Launch_Site"
- %sql select * from SPACEXTABLE where "Launch_Site" like 'CCA%' limit 5
- %sql select sum(PAYLOAD_MASS__KG_) from SPACEXTABLE where "Customer" = 'NASA (CRS)’
- %sql select avg(PAYLOAD_MASS__KG_) from SPACEXTABLE where "Booster_Version" = 'F9 v1.1’
- %sql select min(Date) from SPACEXTABLE
- %sql select "Booster_Version" from SPACEXTABLE where "Landing_Outcome" = 'Success (drone ship)' and "PAYLOAD_MASS__KG_" BETWEEN 4000 AND 6000
- %sql select "Landing_Outcome" , count(*) from SPACEXTABLE group by "Landing_Outcome“
- %sql select "Booster_Version" from SPACEXTABLE where "PAYLOAD_MASS__KG_" = (select max(PAYLOAD_MASS__KG_) from SPACEXTABLE)
- %sql select substr("Date", 6, 2) AS Month,"Landing_Outcome","Booster_Version","Launch_Site" from SPACEXTABLE where "Landing_Outcome" = "Failure (drone ship)“
- %sql select "Landing_Outcome", count("Landing_Outcome") from SPACEXTABLE where Date BETWEEN '2010-06-04' AND '2017-03-20' GROUP BY "Landing_Outcome" order by count("Landing_Outcome") desc

- < https://github.com/yassinm05/Space-Y-Project/blob/main/EDA%20using%20SQL.ipynb>

Build an Interactive Map with Folium

- Mark all launch sites on a map to add each site's location on a map using site's latitude and longitude coordinates
- Mark the success/failed launches for each site on the map
- Calculate the distances between a launch site to its proximities
- < https://github.com/yassinm05/Space-Y-Project/blob/main/Interactive%20visual%20analytics%20using%20folium.ipynb >

Build a Dashboard with Plotly Dash

- Pie chart to explain launch success count for all sites
- Satter plot of payload vs launch success
- < https://github.com/yassinm05/Space-Y-Project/blob/main/plotly%20Dash%20Interactive%20Dashboard.py >

Predictive Analysis (Classification)

- Use the function train_test_split to split the data X and Y into training and test data.
- We Created a logistic regression object
- we created a GridSearchCV object with cv = 10.
- form the dictionary parameters
- We Fit the object to find the best parameters from the dictionary parameters.
- We did the same thing for SVM, decision tree classifier, k nearest neighbors
- We calculated the accuracy and plotted the confusion matrix for each one of the models
- <https://github.com/yassinm05/Space-Y-Project/blob/main/Machine%20learning%20predictions.ipynb>

Results

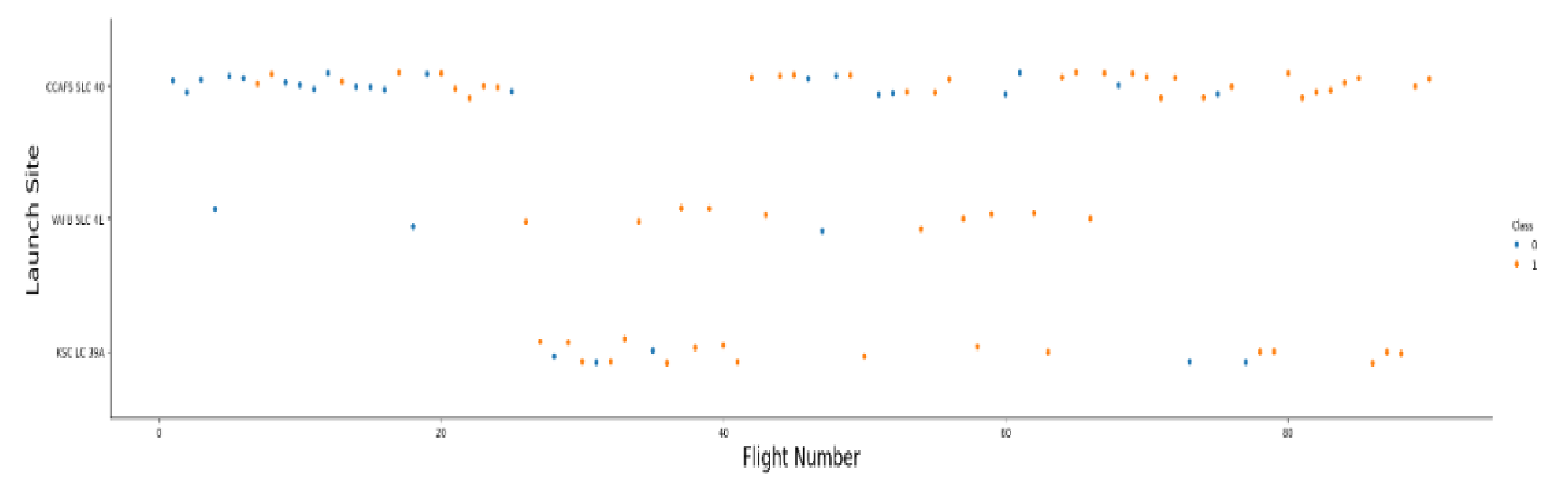
- The best model is KNN with 94% accuracy
- After it the logistic regression with the same accuracy but the knn is a lil better in other aspects
- Then the svm with accuracy of 89%
- Then the Decision tree with 72%

Section 2

Insights drawn from EDA

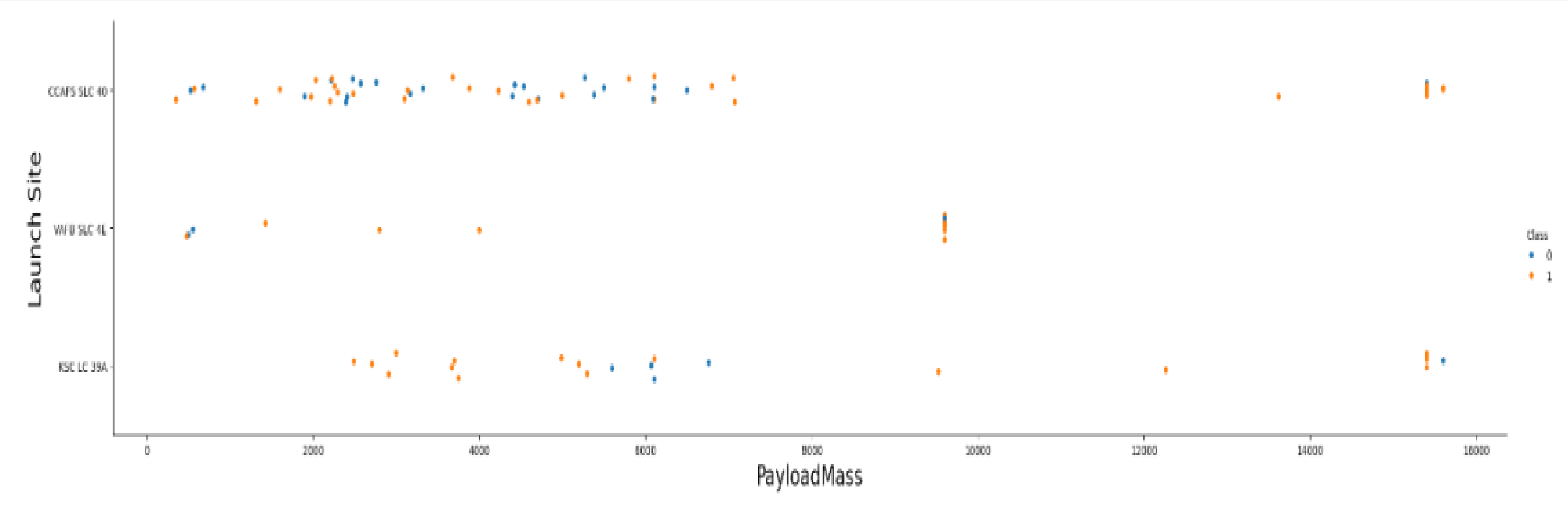
Flight Number vs. Launch Site

- We see that as the flight number increases, the first stage is more likely to land successfully. Also every in every launch site as it increases the first stage is more likely to land successfully
- Every launch site has different percentage of successfull landing we see that the least one of them is CCAFC SLC 40



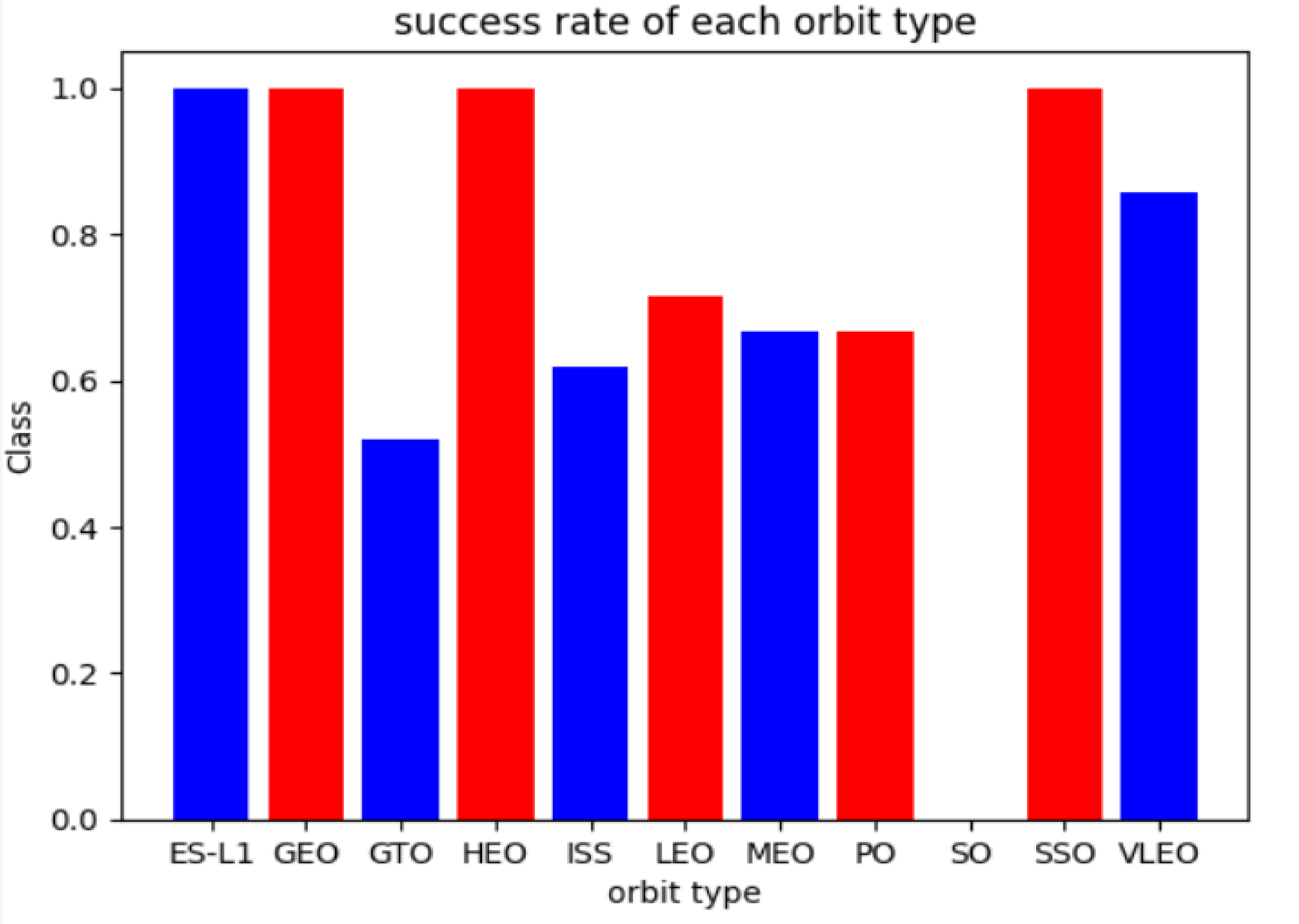
Payload vs. Launch Site

- the VAFB-SLC launchsite there are no rockets launched for heavy payload mass(greater than 10000).



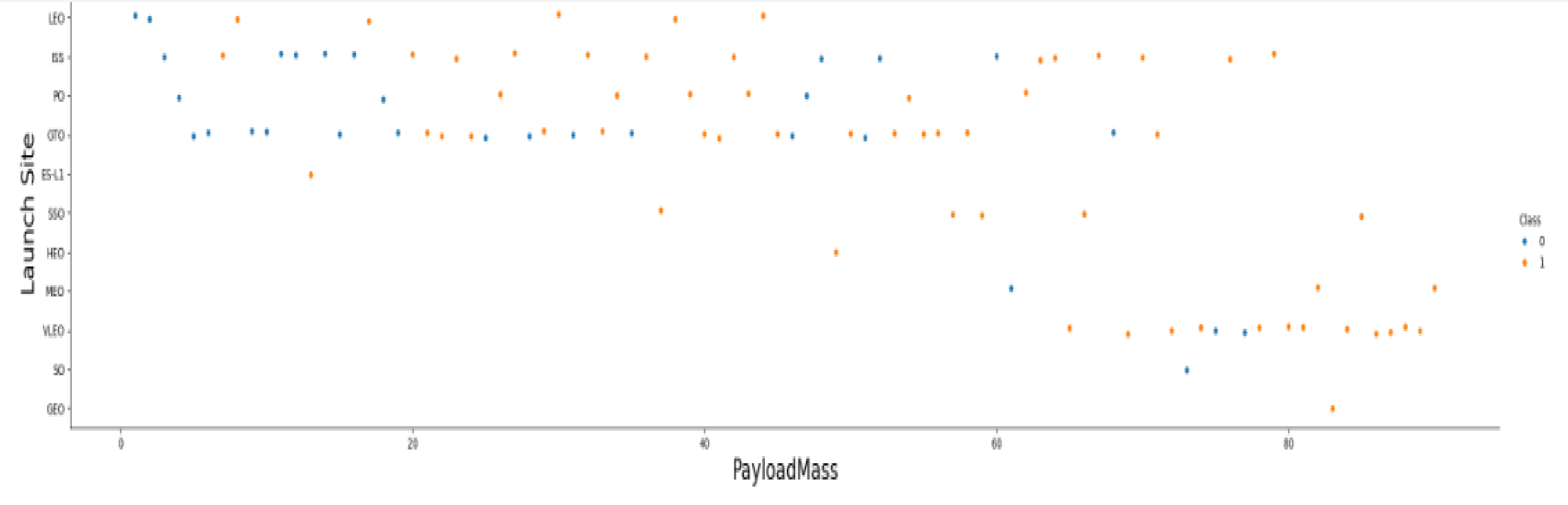
Success Rate vs. Orbit Type

- The orbits with the highest success rate are ES_L1, GEO, HEO, SSO.
- The lowest success rate is SO



Flight Number vs. Orbit Type

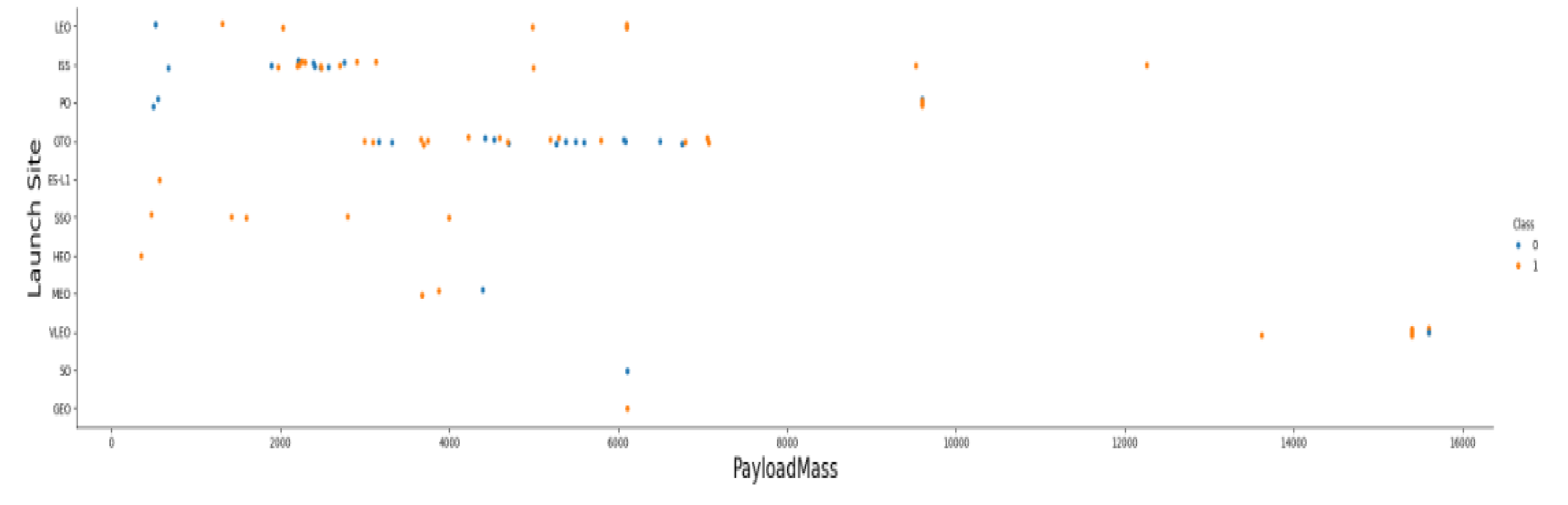
- You can observe that in the LEO orbit, success seems to be related to the number of flights. Conversely, in the GTO orbit, there appears to be no relationship between flight number and success.



Payload vs. Orbit Type

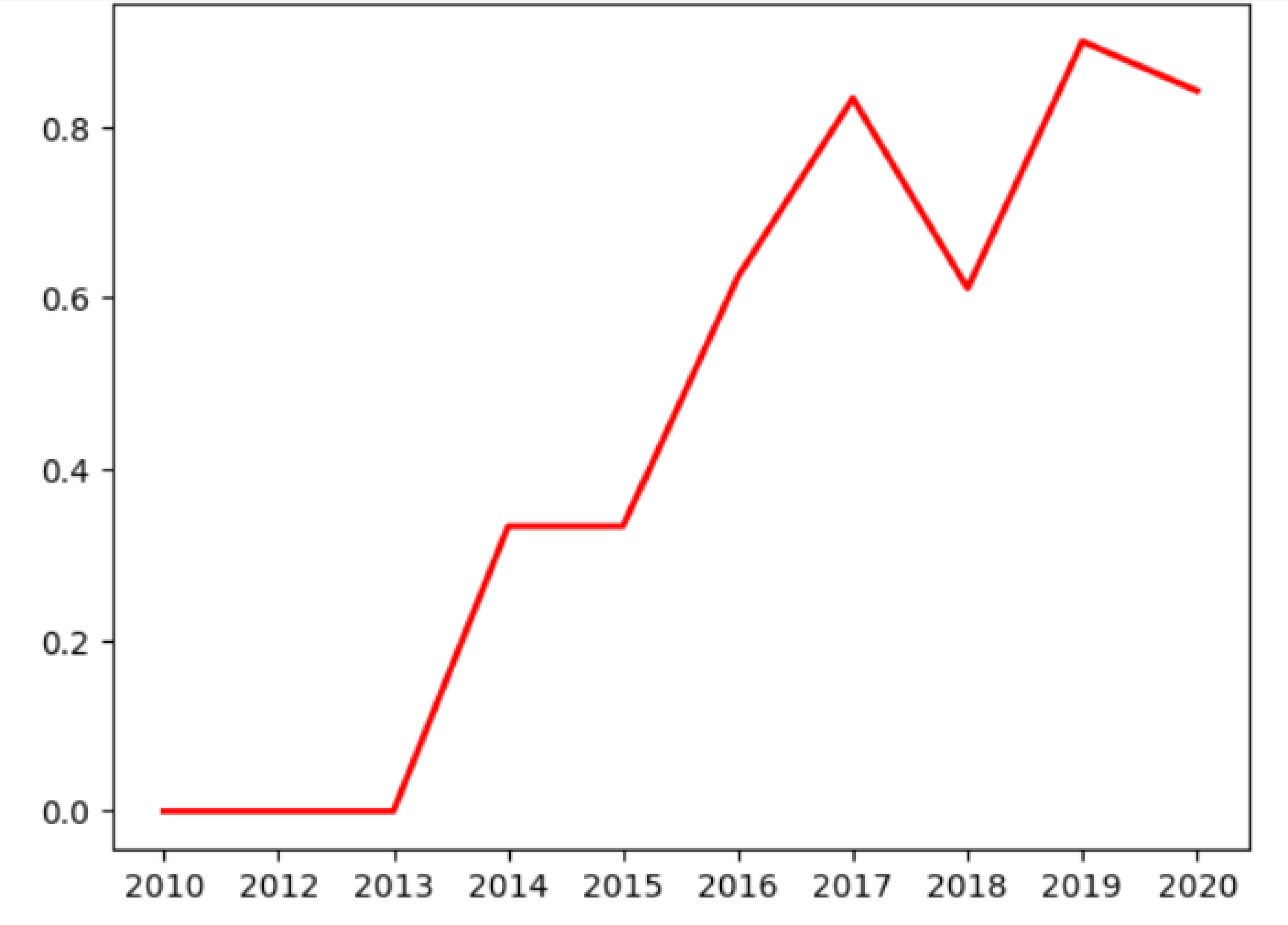
- With heavy payloads the successful landing or positive landing rate are more for Polar,LEO and ISS.

- 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
- The most successful year is 2019



All Launch Site Names

- These are the different launch sites that we have

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

Launch Site Names Begin with 'CCA'

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

This is the result of the sum of all the mass in the payload column in KG

sum(PAYLOAD_MASS__KG_)
45596

Average Payload Mass by F9 v1.1

This is the result of the average of all the mass in the payload column in KG

avg(PAYLOAD_MASS__KG_)
2928.4

First Successful Ground Landing Date

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

min(Date)
2010-06-04

Successful Drone Ship Landing with Payload between 4000 and 6000

- 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

- The total number of successful and failure mission outcome

Landing_Outcome	count(*)
Controlled (ocean)	5
Failure	3
Failure (drone ship)	5
Failure (parachute)	2
No attempt	21
No attempt	1
Precluded (drone ship)	1
Success	38
Success (drone ship)	14
Success (ground pad)	9
Uncontrolled (ocean)	2

Boosters Carried Maximum Payload

- 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

- 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
01	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
04	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40
01	Failure (drone ship)	F9 v1.1 B1017	VAFB SLC-4E
03	Failure (drone ship)	F9 FT B1020	CCAFS LC-40
06	Failure (drone ship)	F9 FT B1024	CCAFS LC-40

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

- Rank of the count of landing outcomes between the date 2010-06-04 and 2017-03-20.

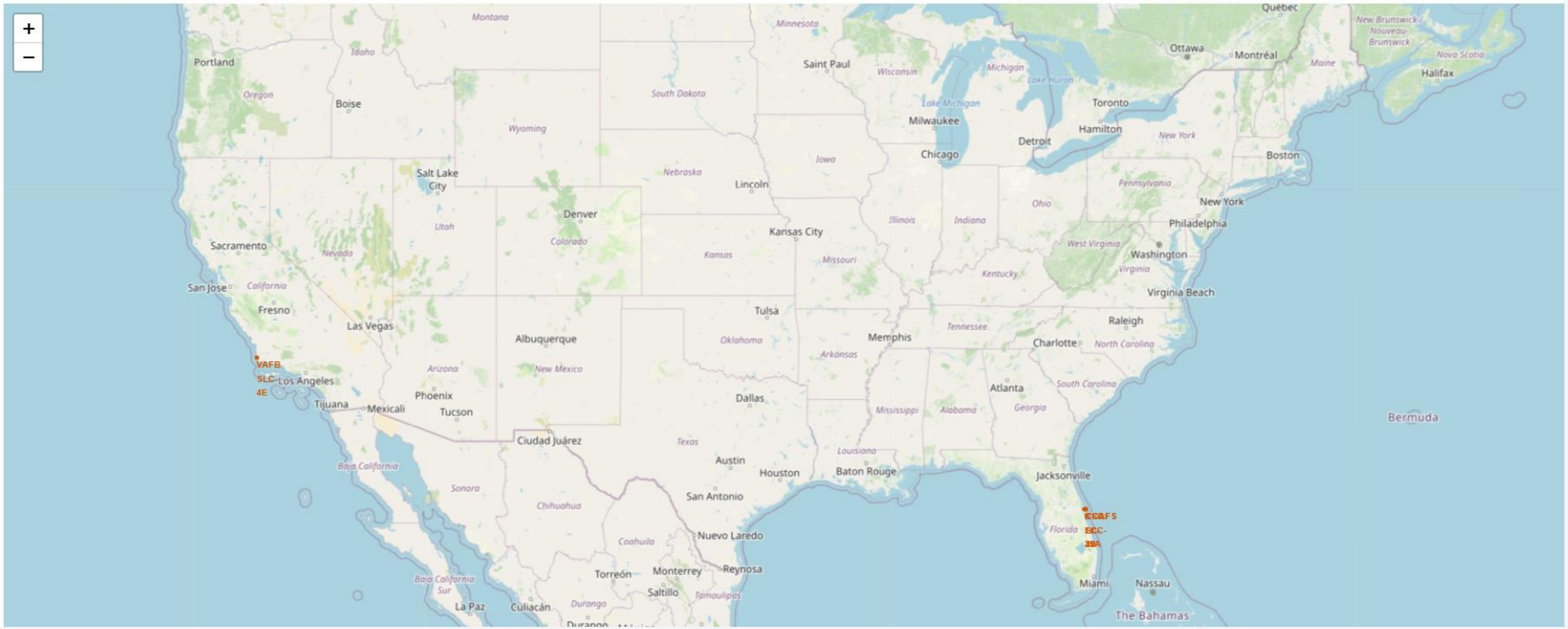
Landing_Outcome	count("Landing_Outcome")
No attempt	10
Success (drone ship)	5
Failure (drone ship)	5
Success (ground pad)	3
Controlled (ocean)	3
Uncontrolled (ocean)	2
Failure (parachute)	2
Precluded (drone ship)	1

Section 3

Launch Sites Proximities Analysis

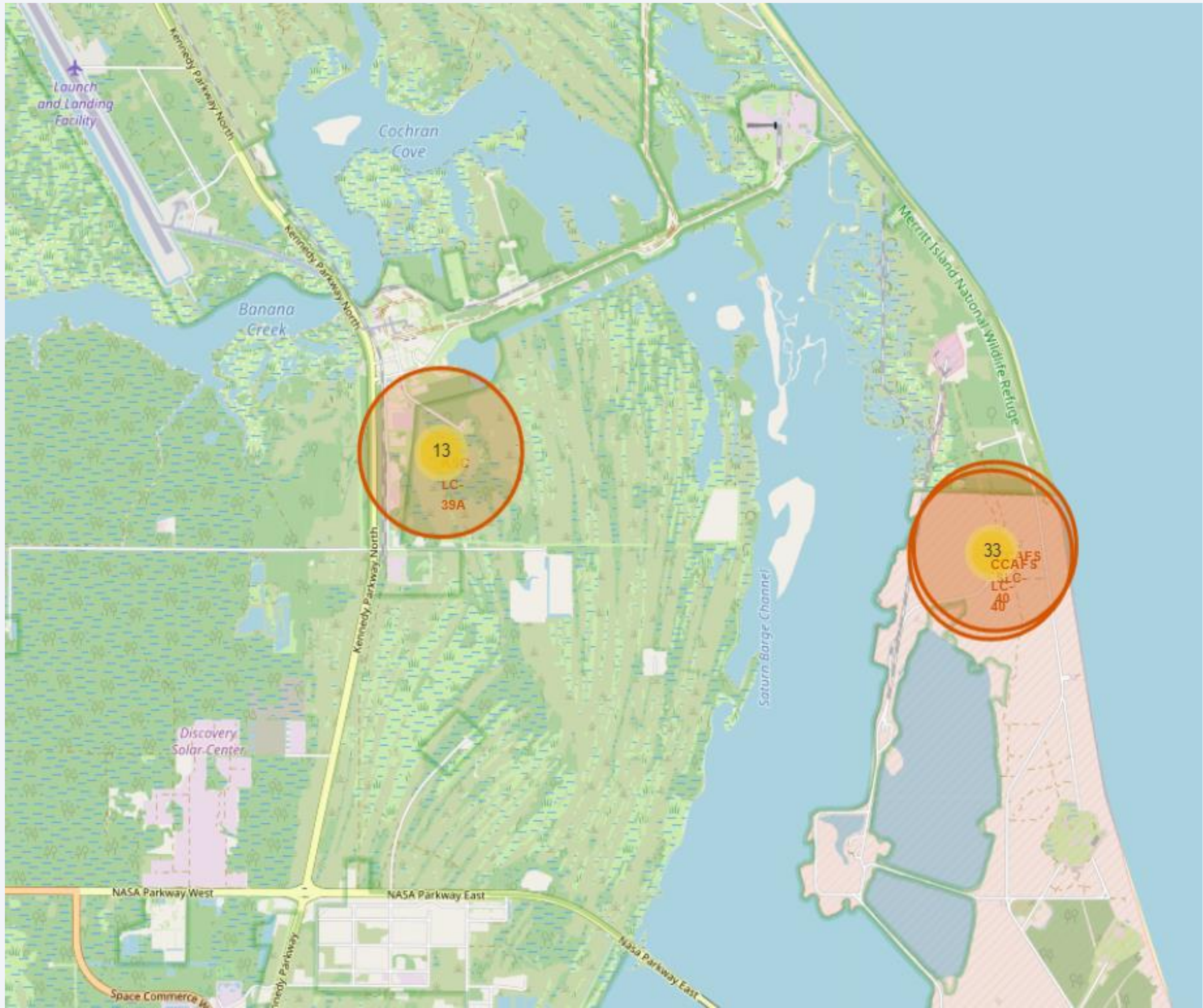
A map with all launch sites marked on

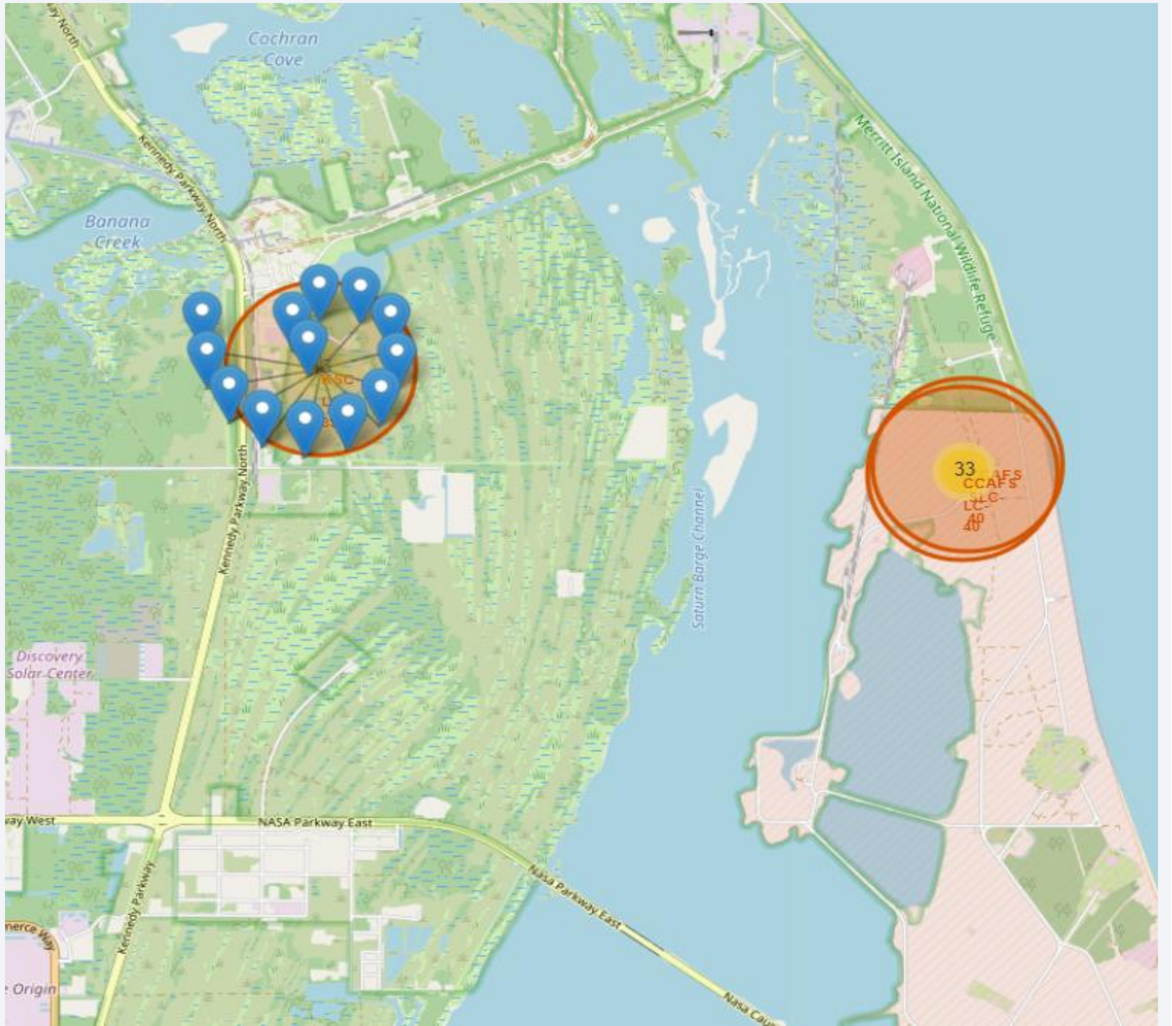
As you see each marker represent a launch site



success/failed launches for each

If class=1, marker_color value will be green, If class=0, marker_color value will be red

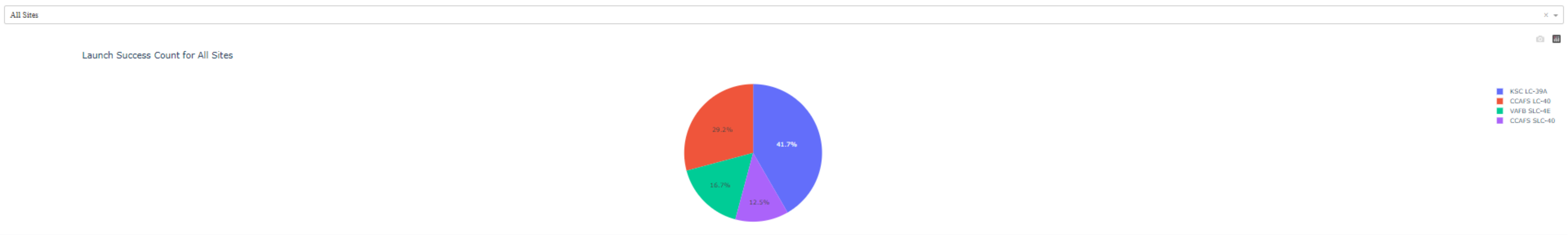




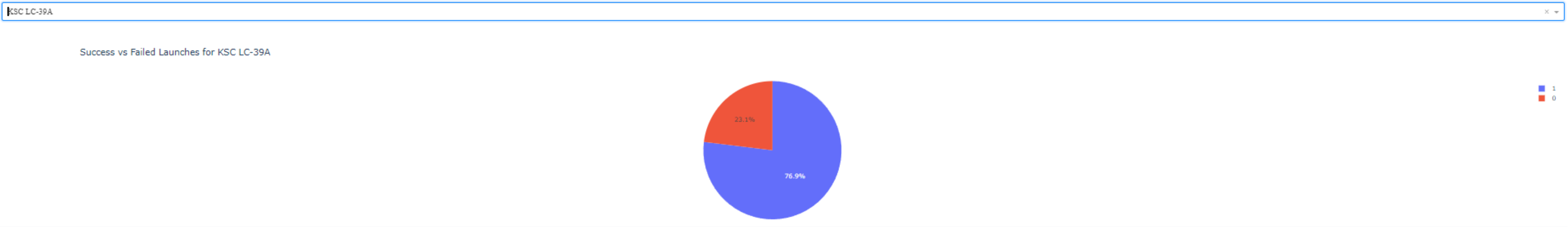
Section 4

Build a Dashboard with Plotly Dash

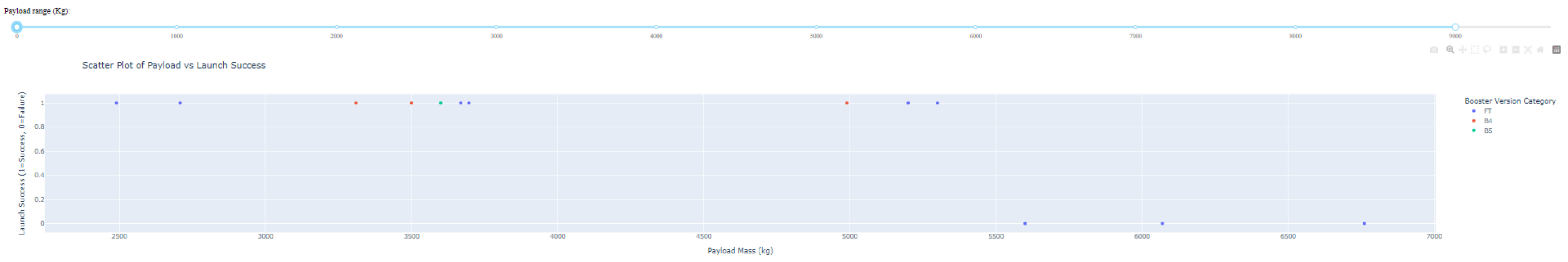
launch success count for all sites



Pie chart for the launch site with highest launch success ratio



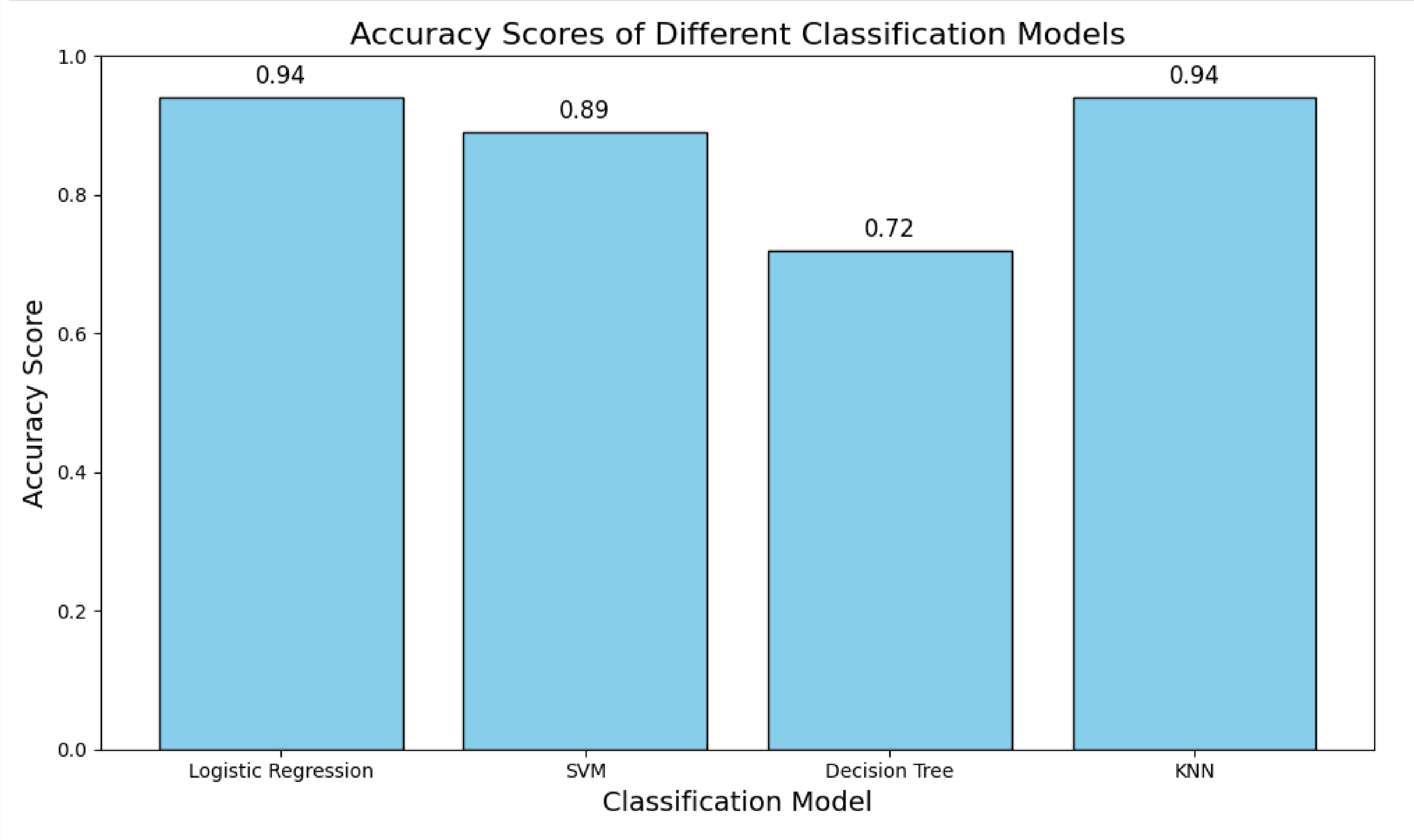
Payload vs. Launch Outcome scatter plot for all sites



Section 5

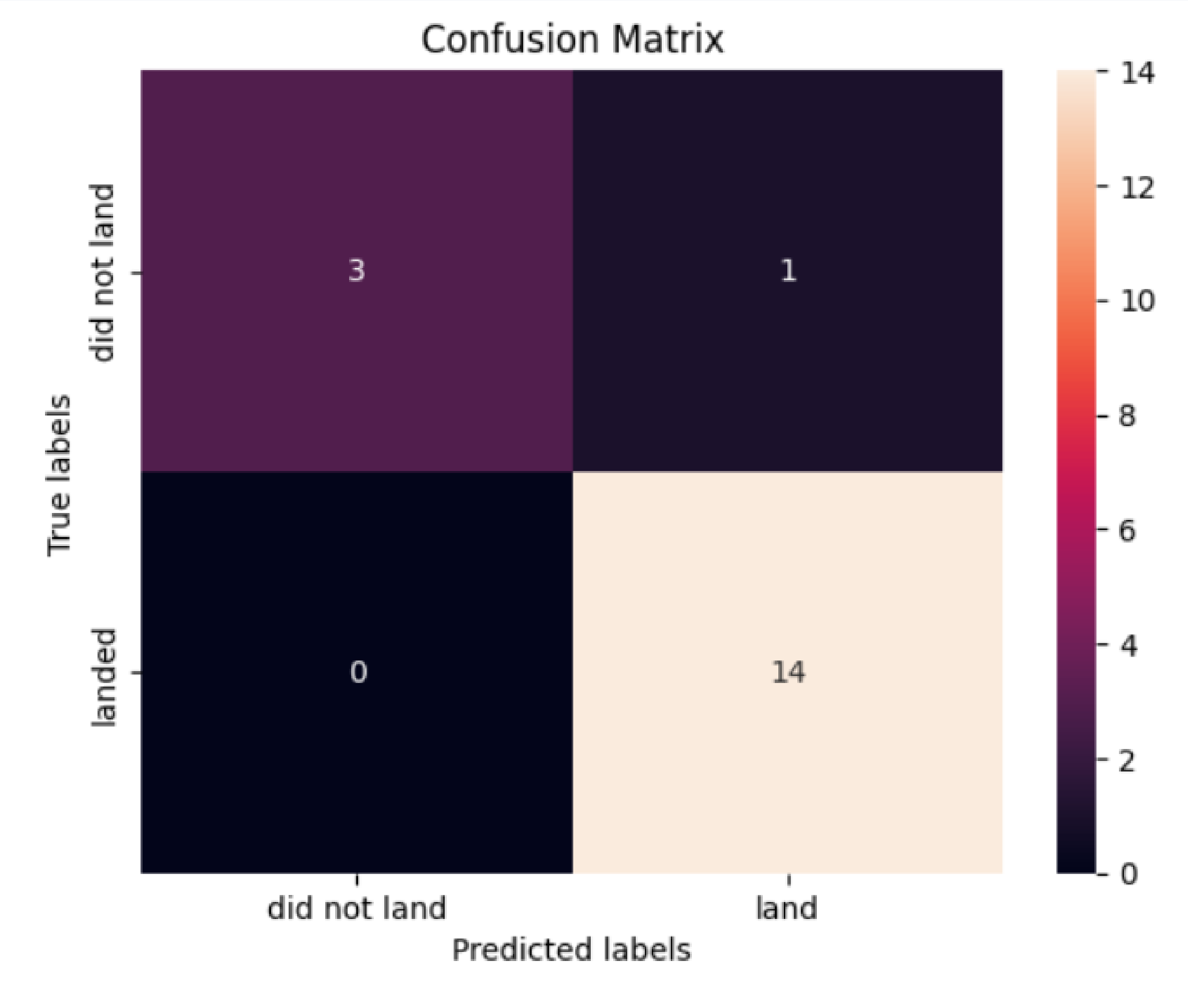
Predictive Analysis (Classification)

Classification Accuracy



Confusion Matrix

- Rhe confusion matrix of the best performing model which is the KNN
- Only false positive and 0 false negative

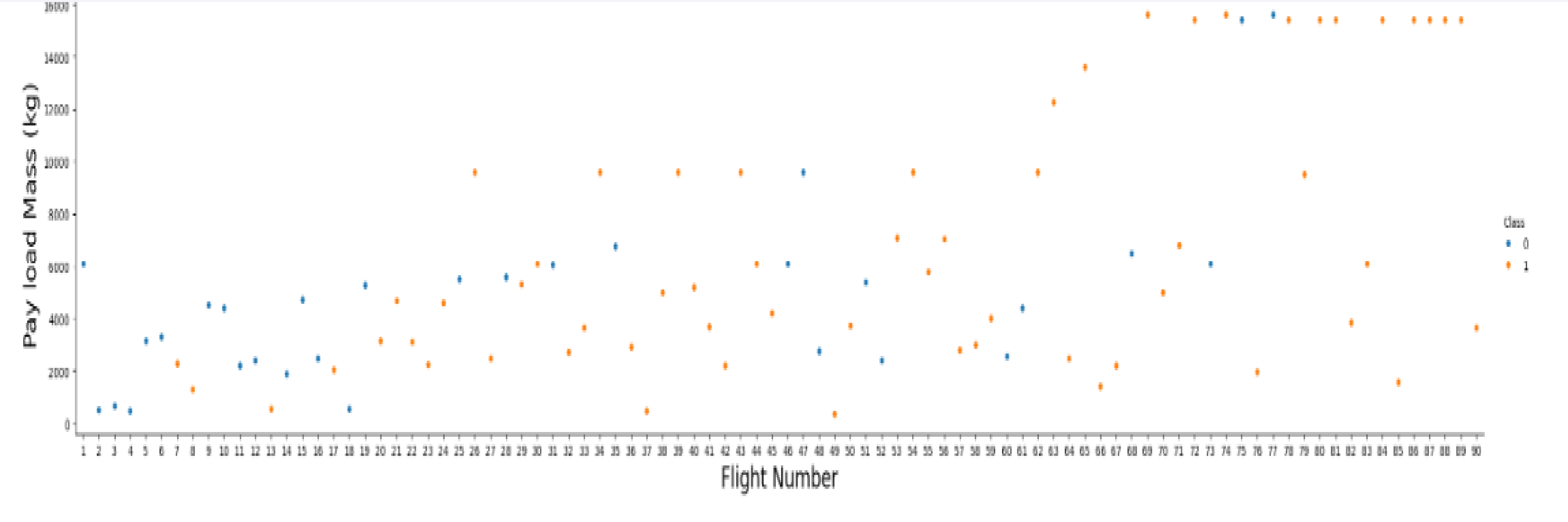


Conclusions

- We collected the data using different sources and API
- We performed Data wrangling and created new features
- We performed EDA and Data visualization using different methods
- We predicted wether or not the space ship will land it’s 1’st stage through varius classification models
- We can do much more with this data

Appendix

- Flight number vs Payload point scatter



Thank you!