IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Nuwaya Terefe
2/17/2023

Outline

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

Executive Summary

- Summary of methodologies
  - Data collection
  - Data wrangling
  - EDA with SQL
  - Prediction Analysis
- Summary of all results

Introduction

- Project background and context

  <u>**Falcon 9:**</u>

    - Reusable, two-stage rocket designed and manufactured by SpaceX
    - Cost: 62 million dollars (~ 100 million Dollar less expensive than others)

- Problems you want to find answers
  - How successful is Falcon 9 first stage landing?

Section 1

Methodology

Methodology

Executive Summary

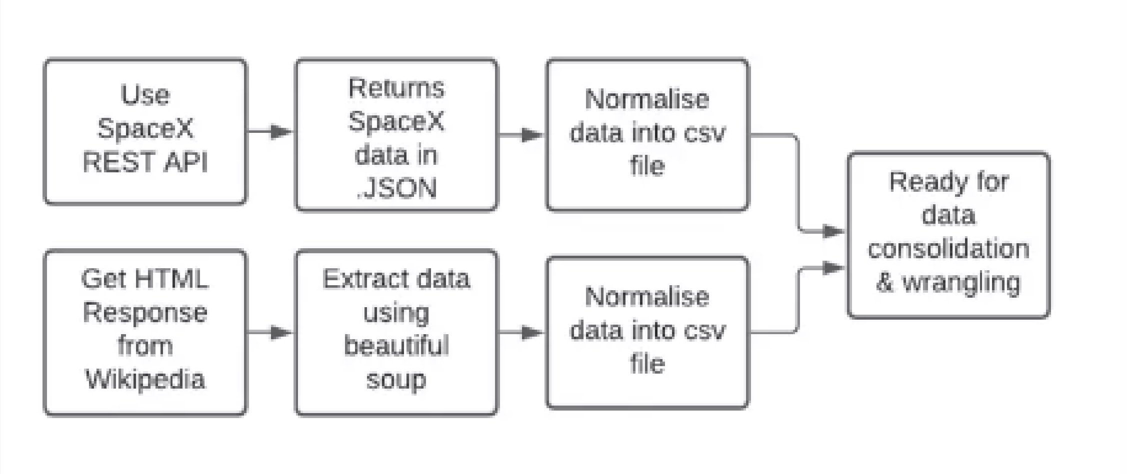
- Data collection methodology:
  1. Rest API
  2. Web Scraping – From Wikipedia
- Perform data wrangling
  - Data inspection
  - Data Cleaning
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - building, tuning, and evaluating classification models requires a combination of data preparation, feature selection, model building, model tuning, and model evaluation.

Data Collection

- REST (Representational State Transfer) API is a common way to access data and perform operations over the web. It is a lightweight and flexible architecture that allows for communication between different systems and devices. REST API can be used in data collection to retrieve data from various sources and integrate it into a centralized database for further processing and analysis.

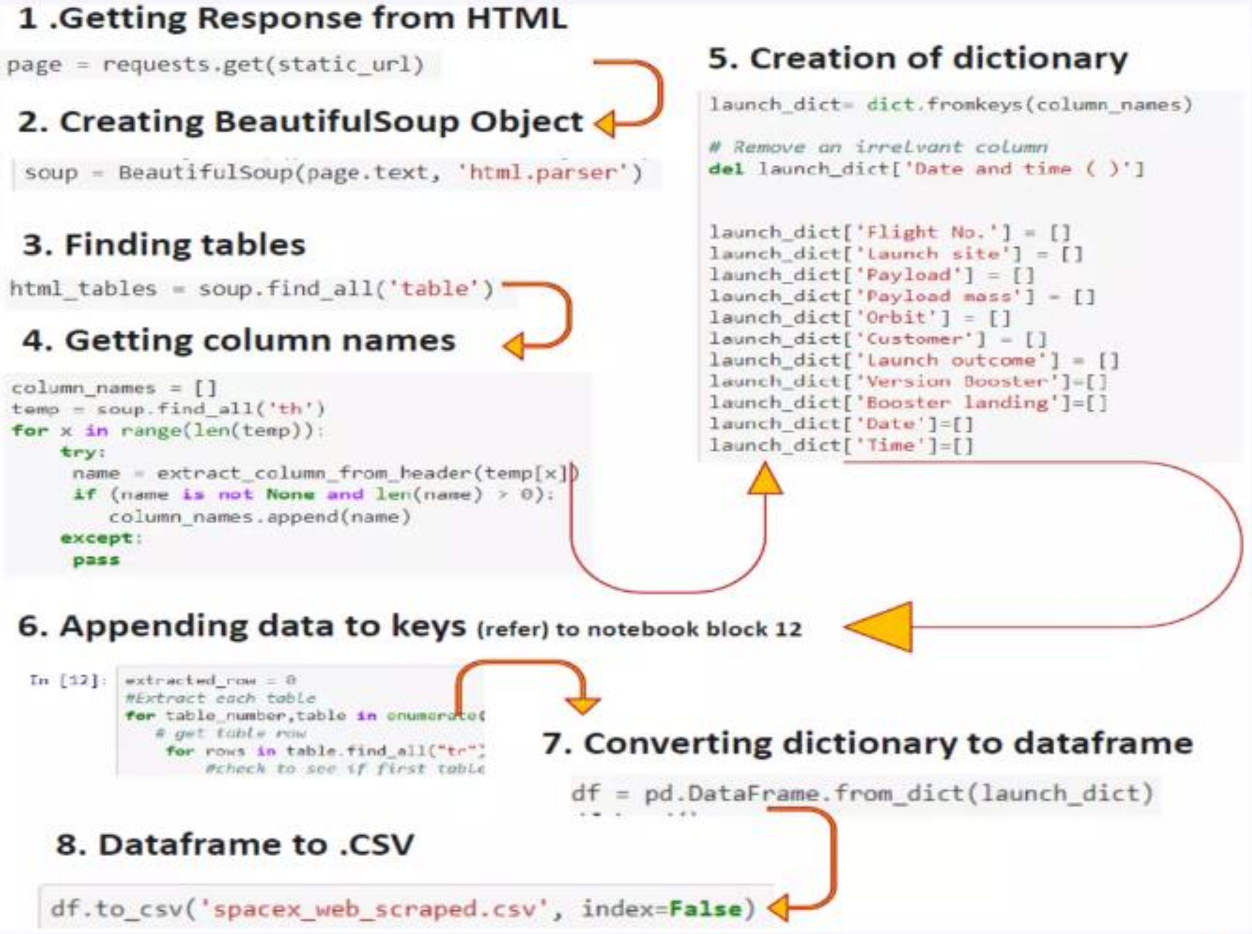
Data Collection – SpaceX API

- This API will give us data about: launches, including information about the rocket used, payload delivered, launch specifications, landing specifications, and landing outcome.
- GitHub SpaceX API calls notebook



Data Collection - Scraping

- Use Python BeautifulSoup package to web scrape some HTML tables that contain valuable Falcon 9 launch records.
- Parse the data from those tables and convert them into a Pandas data frame for further visualization and analysis.
- [Github Data Collection – Scraping]()



Data Wrangling

- Transform the raw data into a clean dataset.
  - Wrangling Data using an API, Sampling Data, and Dealing with Nulls.
- Use the API targeting Booster, Launchpad, payload, and core. The data will be stored in lists and will be used to create our dataset.
- Filter/sample the data:
  - to remove Falcon 1 launches.
- Finally, deal with the NULL values inside the PayloadMass

  By calculating the mean of the PayloadMass data and then replace the null values in PayloadMass with the mean.

EDA with SQL

- Helps to see if data can be used to automatically determine if the Falcon 9's second stage will land.
- Steps:
  - Load the data (spacexDataSet - CSV) to DB2
  - Connect to the database
    - Using DB2 magic
    - Format: **%sql ibm_db_sa://my-username:my-password@my-hostname:my-port/my-db-name**

EDA with SQL

- Overview of the DataSet

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

launch_site
CCAFS LC-40
CCAFS LC-40
CCAFS LC-40
CCAFS LC-40
CCAFS LC-40

total payload mass carried by boosters launched by NASA (CRS): **619967**

average payload mass carried by booster version F9 v1.1: **6138**

the date when the first successful landing outcome in ground pad was achieved: **2010-06-04**

names of the boosters which have success in drone ship and have 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

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
No attempt
Success (ground pad)
Success (drone ship)
Success (drone ship)
Success (ground pad)
Failure (drone ship)
Success (drone ship)
Success (drone ship)
Success (drone ship)
Failure (drone ship)
Failure (drone ship)
Success (ground pad)
Precluded (drone ship)
No attempt
Failure (drone ship)
No attempt
Controlled (ocean)
Failure (drone ship)
Uncontrolled (ocean)
No attempt
No attempt
Controlled (ocean)
Controlled (ocean)
No attempt
No attempt
Uncontrolled (ocean)
No attempt
No attempt
No attempt
Failure (parachute)
Failure (parachute)

the total number of successful and failure mission outcomes:

1
99
1

names of the booster versions which have carried the maximum payload mass. Use a subquery:

boosterversion
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 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

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

1	mission_outcome	booster_version	launch_site
1	Success	F9 v1.1 B1012	CCAFS LC-40
2	Success	F9 v1.1 B1013	CCAFS LC-40
3	Success	F9 v1.1 B1014	CCAFS LC-40
4	Success	F9 v1.1 B1015	CCAFS LC-40
4	Success	F9 v1.1 B1016	CCAFS LC-40
6	Failure (in flight)	F9 v1.1 B1018	CCAFS LC-40
12	Success	F9 FT B1019	CCAFS LC-40

- [Github EDA with SQL]

Build an Interactive Map with Folium

- Markers: You added several markers to the map, each representing a specific location. Each marker displayed a tooltip with additional information when clicked.
- Circles: You also added circles to the map, each representing a specific area. The circles were filled with a specific color and had a popup displaying additional information when clicked.
- Lines: You added lines to the map, each representing a specific route or path. Each line was drawn with a specific color and weight and had a popup displaying additional information when clicked.
- GeoJSON layers: You added several GeoJSON layers to the map, each displaying specific features such as country boundaries, city boundaries, or points of interest. Each GeoJSON layer had a specific color and opacity and displayed additional information when clicked. Explain why you added those objects
- GitHub – Analysis with Folium

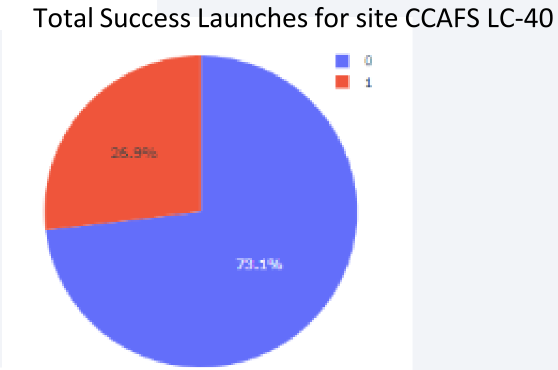
Section 4

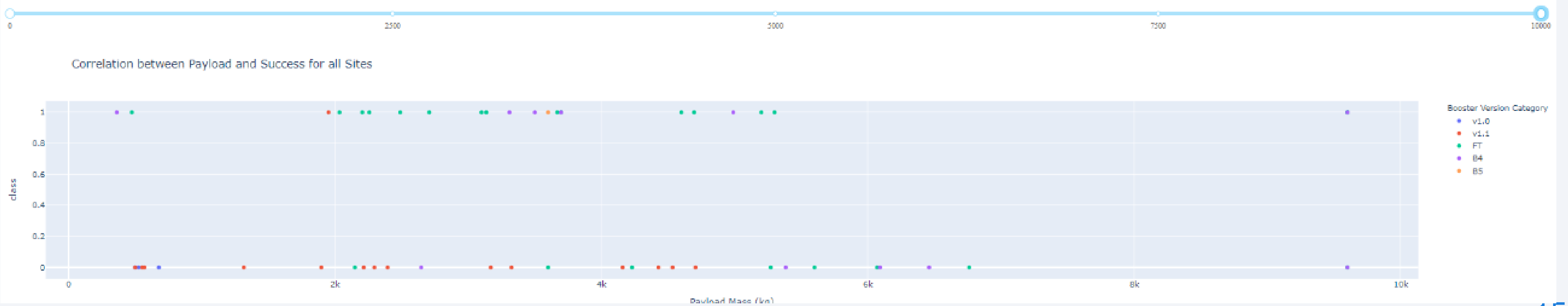
Build a Dashboard with Plotly Dash

Build a Dashboard with Plotly Dash

The visual analysis using the dashboard, answer the following five questions:

1. Which site has the largest successful launches?
2. Which site has the highest launch success rate?
3. Which payload range(s) has the highest launch success rate?
4. Which payload range(s) has the lowest launch success rate?
5. Which F9 Booster version (v1.0, v1.1, FT, B4, B5, etc.) has the highest



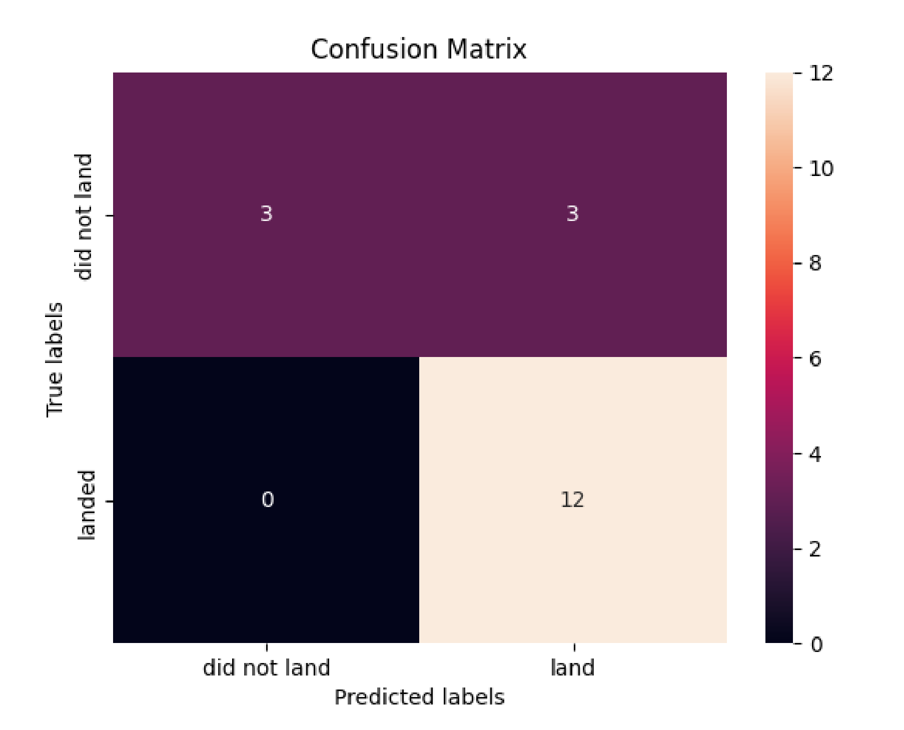


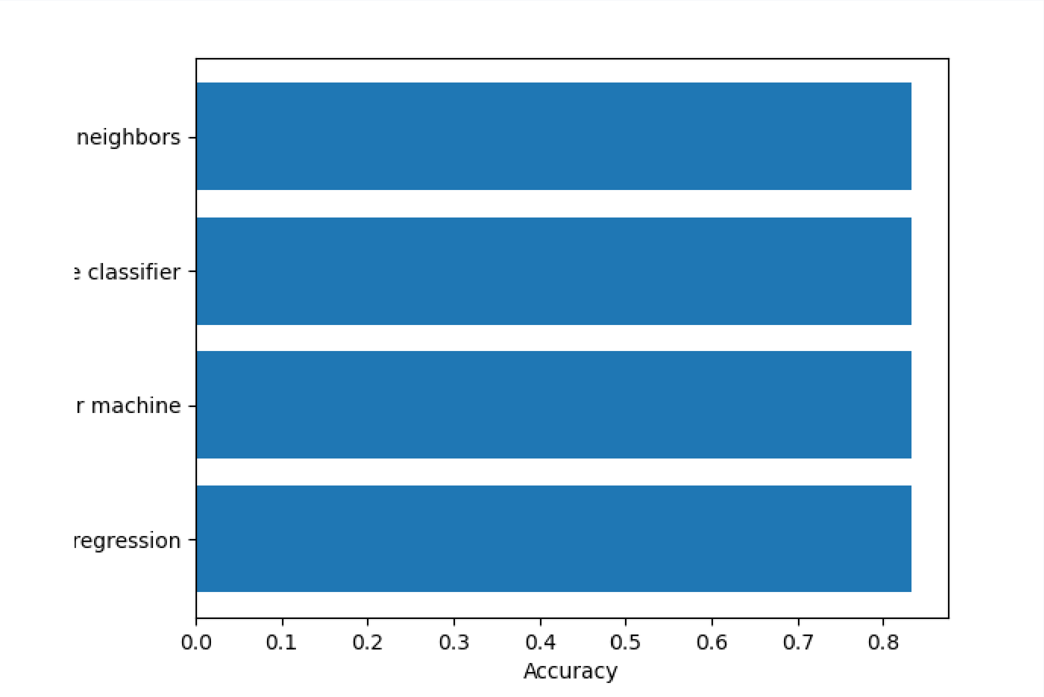
Section 5

Predictive Analysis (Classification)

Predictive Analysis (Classification)

1. create a column for the class
2. Standardize the data
3. Split into training data and test data
   - Find best Hyperparameter for SVM, Classification Trees and Logistic Regression
4. Find the method performs best using test data





Predictive Analysis (Classification)

Steps followed:

1. Create a NumPy array, then assignee to a variable – Y. Standardize the data in X
2. Split the data X and Y into training and test data.
3. Create a logistic regression object, display best parameter and accuracy
4. Calculate the accuracy on the test data
5. Find the method that performs best

GitHub - Machine Learning Prediction

Results

18 test samples
tuned hpyerparameters :(best parameters) {'C': 0.01, 'penalty': 'l2', 'solver': 'lbfgs'}
accuracy : 0.8464285714285713

tuned hpyerparameters :(best parameters) {'criterion': 'gini', 'max_depth': 18, 'max_features': 'sqrt', 'min_samples_leaf': 4, 'min_samples_split': 2, 'splitter': 'random'}
accuracy : 0.8875

accuracy of tree_cv on the test data
test set accuracy : 0.8333333333333334

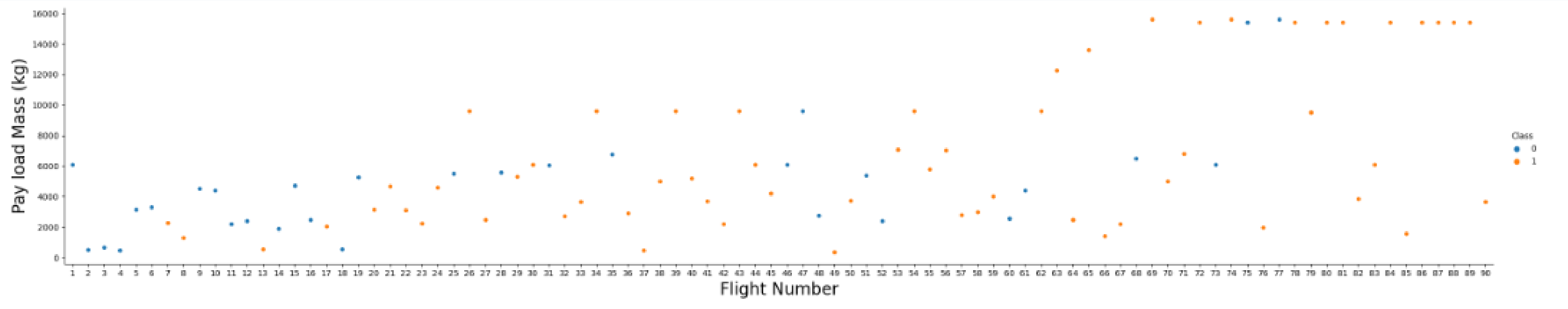
k nearest neighbors
tuned hpyerparameters :(best parameters) {'algorithm': 'auto', 'n_neighbors': 10, 'p': 1}
accuracy : 0.8482142857142858

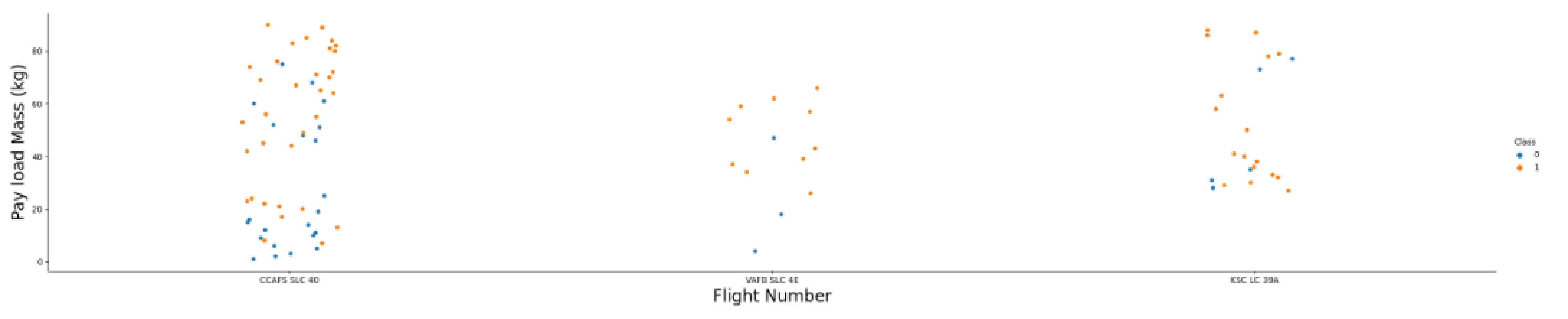
accuracy of knn_cv on the test data
test set accuracy : 0.8333333333333334

Section 2

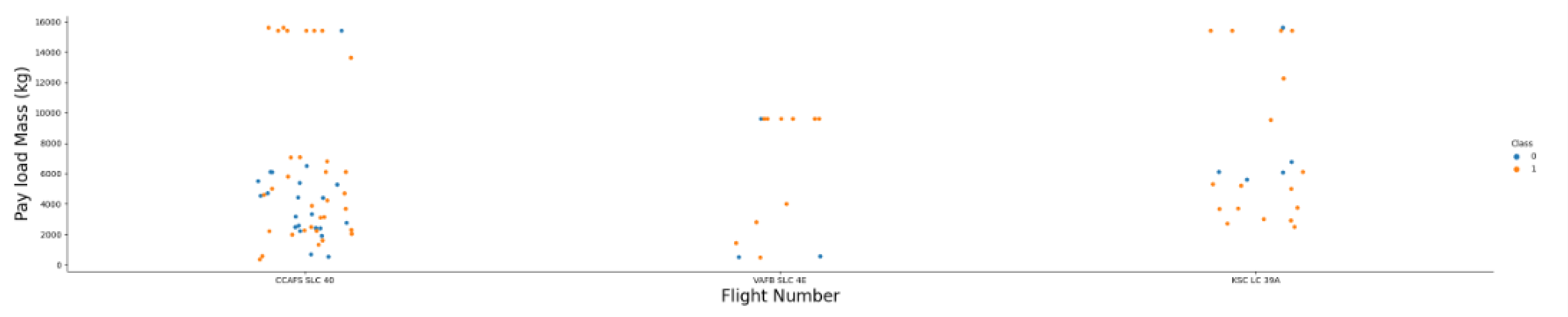
Insights drawn from EDA

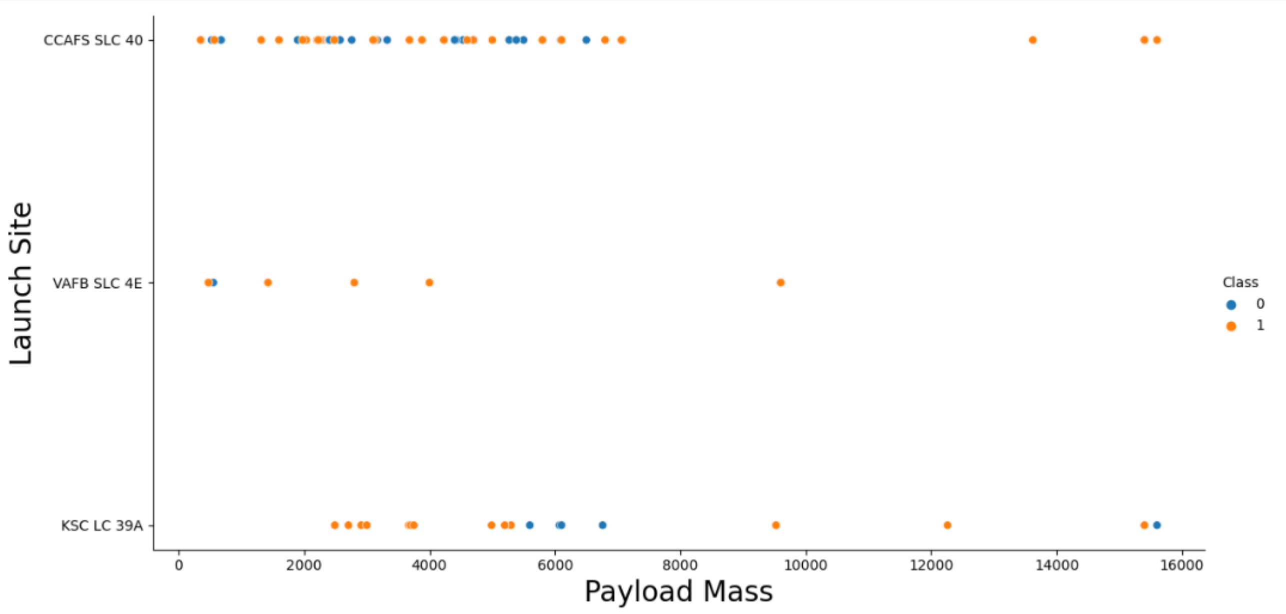
EDA with Data Visualization



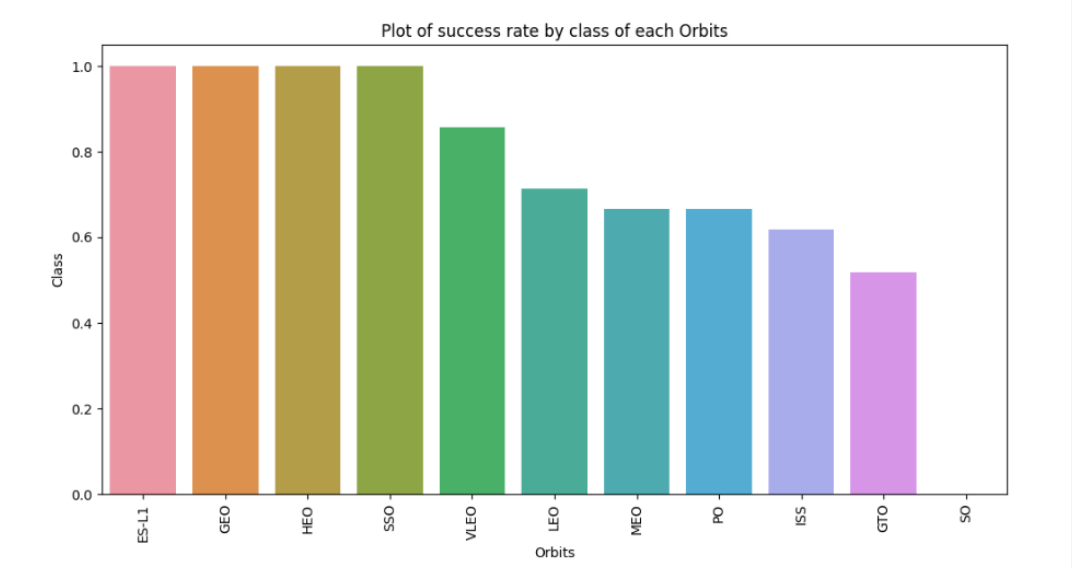


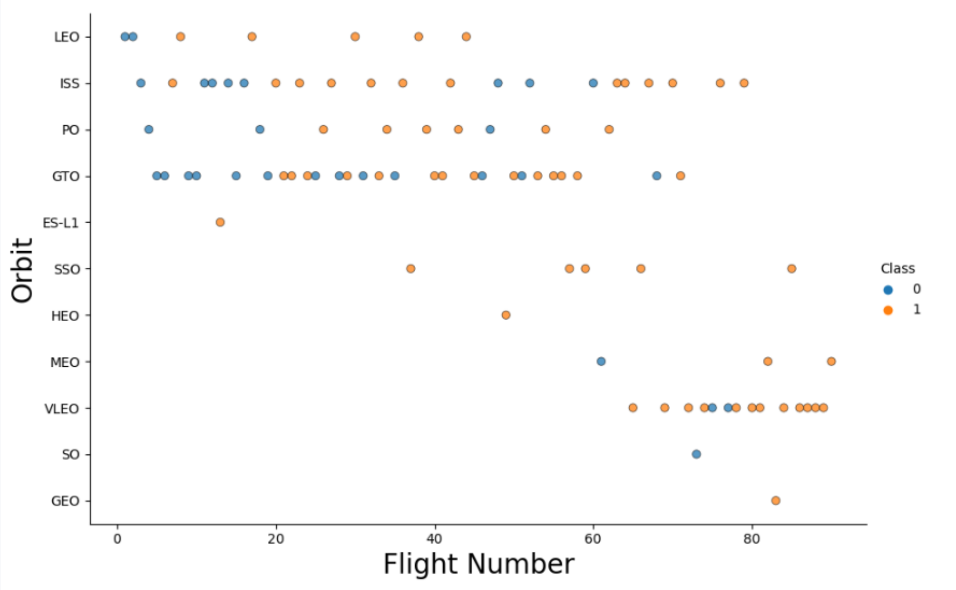
EDA with Data Visualization

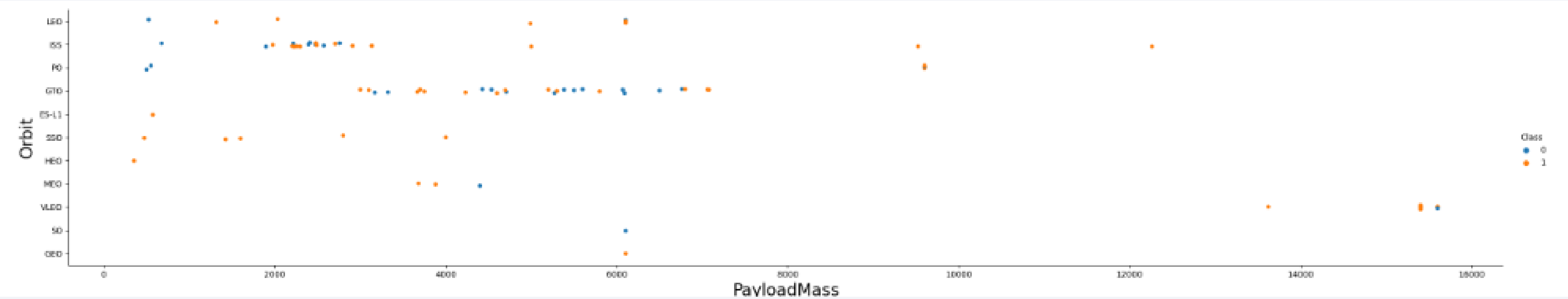




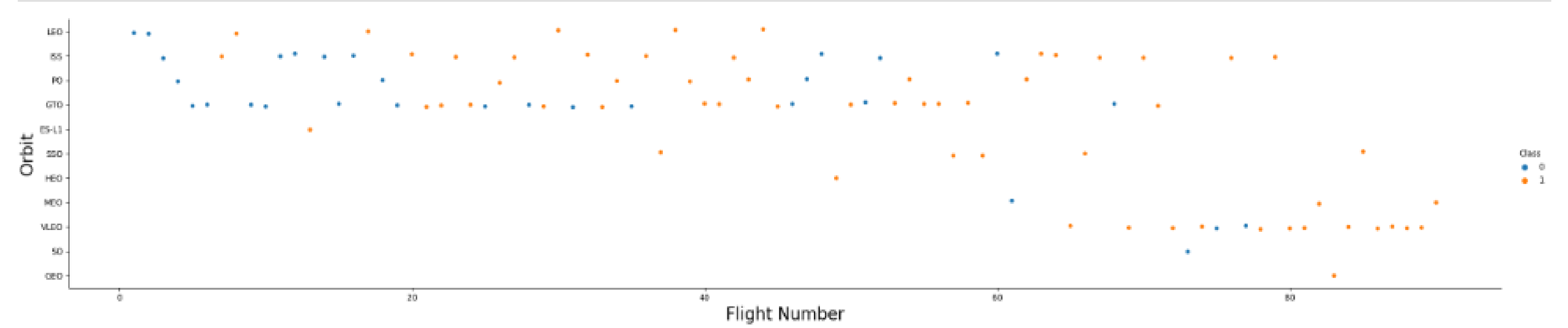
EDA with Data Visualization

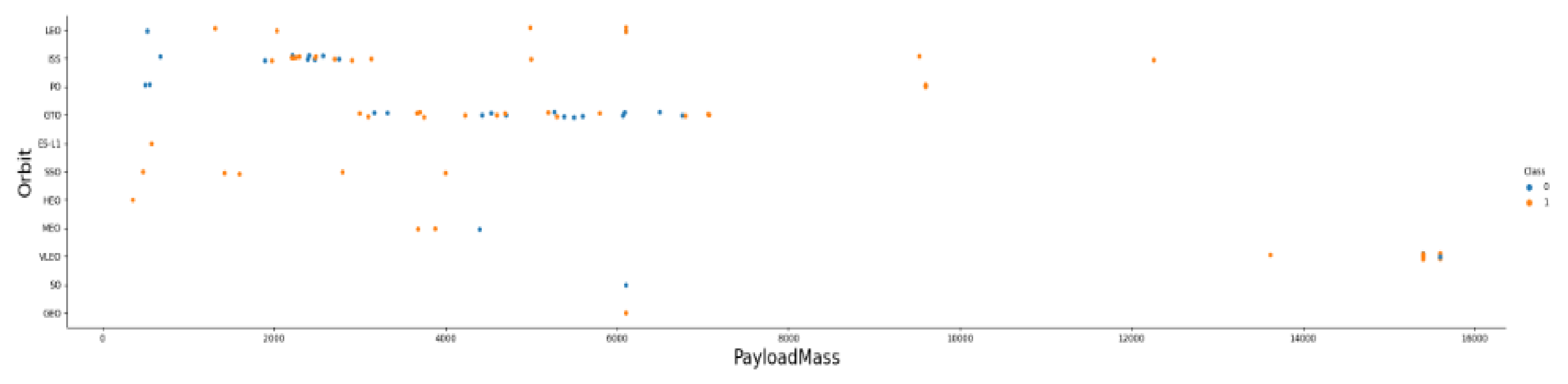




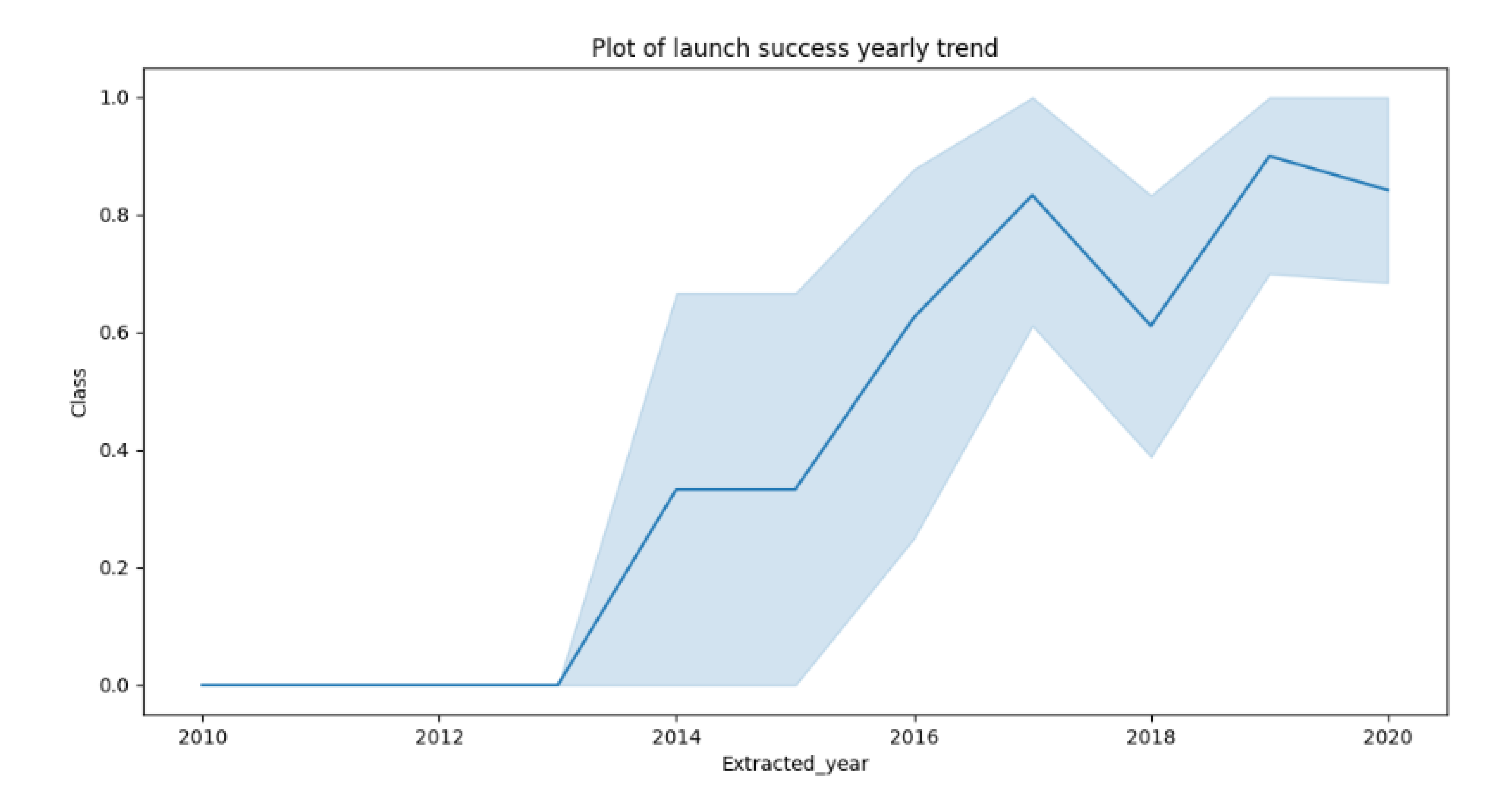


EDA with Data Visualization

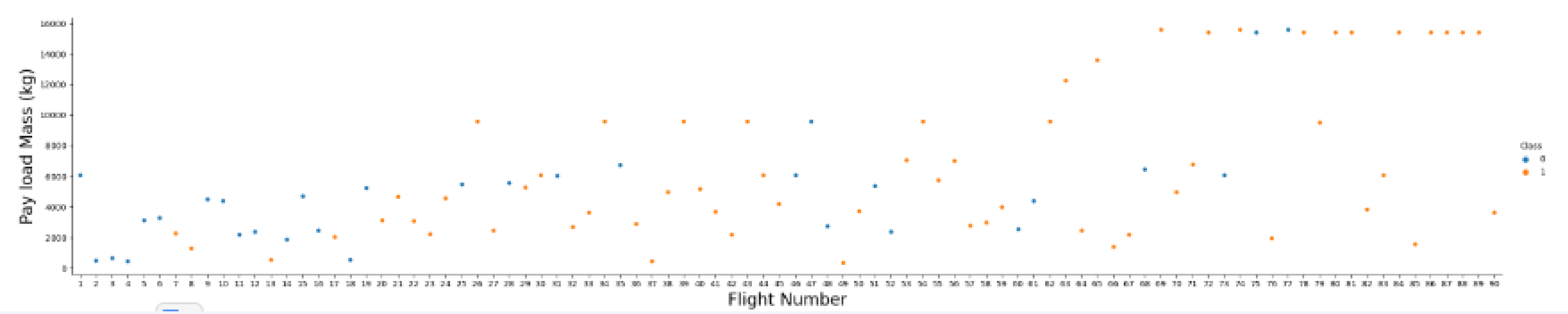




EDA with Data Visualization



EDA with Data Visualization

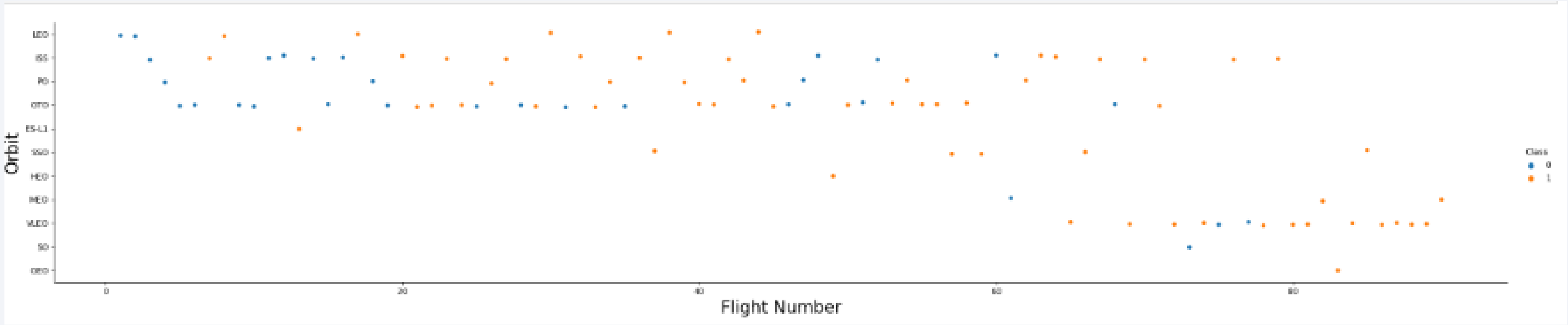


FlightNumber vs. PayloadMassand overlay the outcome of the launch. We see that as the flight number increases, the first stage is more likely to land successfully.

EDA with Data Visualization

- different launch sites have different success rates.
    - CCAFS LC-40, has a success rate of 60 %,
    - KSC LC-39A and VAFB SLC 4E has a success rate of 77%.

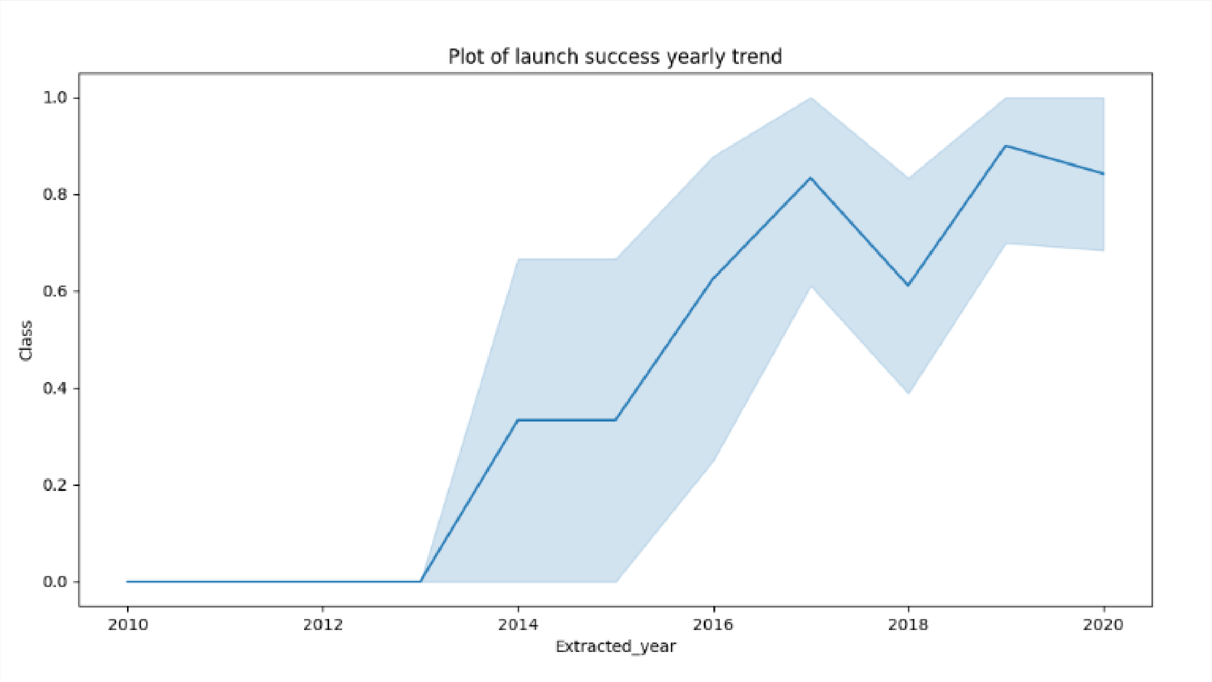
EDA with Data Visualization



- LEO orbit the Success appears related to the number of flights.
- no relationship between flight number when in GTO orbit.

EDA with Data Visualization

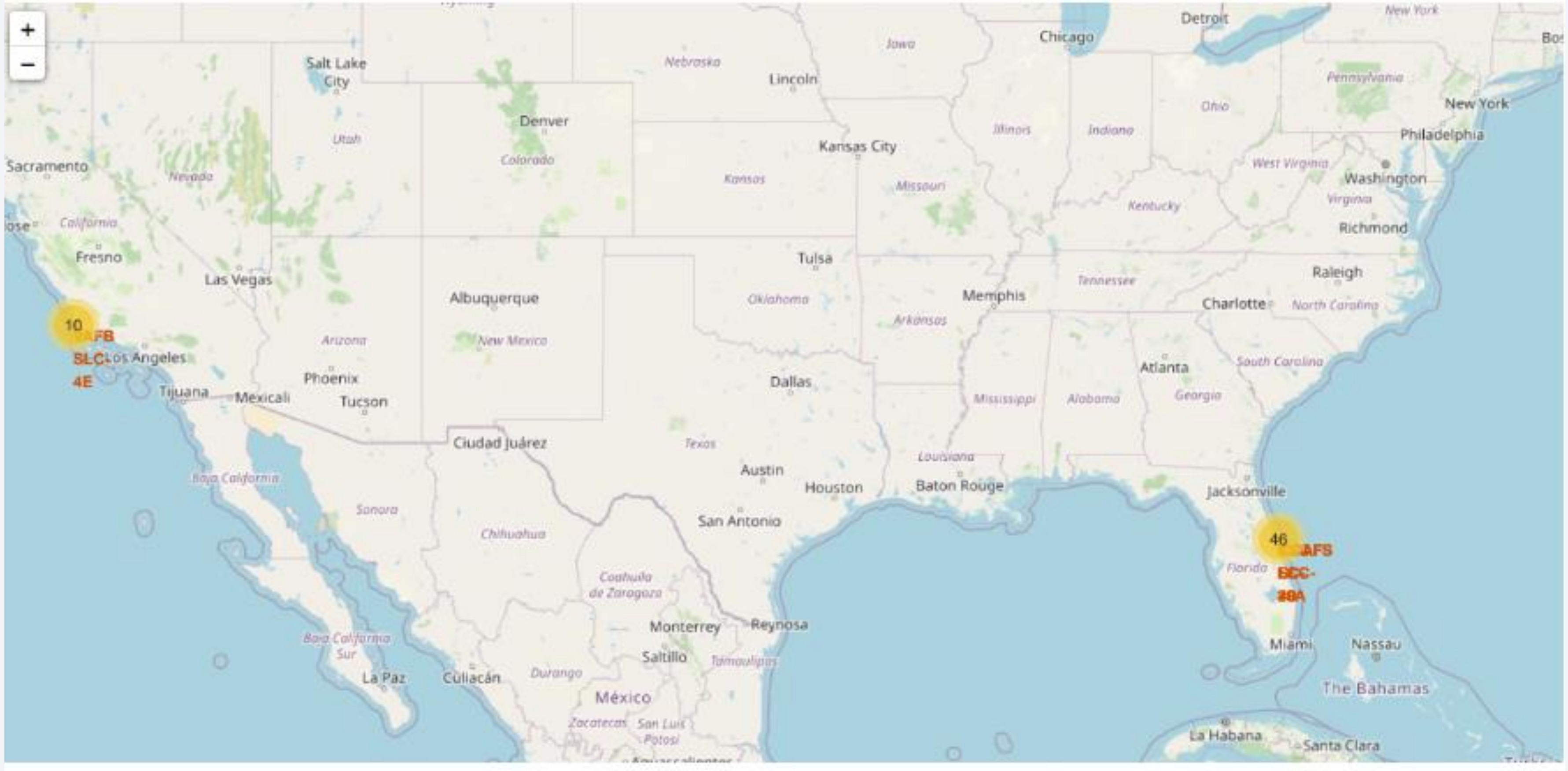
- Yearly Success trend



Section 3

Launch Sites Proximities Analysis

<Folium Map Screenshot 1>



By zooming in and out, it can help to see how the launch sites’ proximity to the equator and coast

<Folium Map Screenshot 2>

Helps to identify which launch sites have relatively high success rates.

<Folium Map Screenshot 3>

Proximity to:

- Coastline,
- Railway
- Highway

- [GitHub – Locations Analysis with Folium]()

Thank you!