IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Hamzah Shabbir
12.12.2021

Outline

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

Executive Summary

- Data for this were collected from SpeceX API and Wikipedia page of SpaceX. Data were sampled for classification whether landing was successful or not. Exploratory analysis were done using SQL. For detailed analysis visualization were done in folium map and dashboards.
- Prediction analysis were done on four machine learning algorithm to select best accurate model by hyper tuning model using grid search.
- Models:
    - Logistic Regression
    - Support Vector Machine
    - Decision Tree Classifier
    - K Nearest Neighbors
- Accuracy of all models were between 80 % to 85 % which can be considered good.

Introduction

Background

- With the commercialization of space, there is going to be huge competition in space market considering investment, pricing and strategy
- Possible due to ability to recover part of rocket from stage I
- Hypothetical space company analysis to present scenario to compete with SpaceX

Problem

- Prediction of successful stage I recovery by training model on machine learning classification algorithm

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Integration of data from SpaceX API and data collected from scrapping wikipedia
- Perform data wrangling
  - Classifying successful landing as successful and other as unsuccessful
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Hyper parameters of model were tuned using grid search

Data Collection

- Data were extracted using two method and sources:
    1. Extracted from SpaceX API and then conversion of result from json to dataframe using pandas
    2. Web scrapping was done to extract data from Wikipedia page of SpaecX using request and beautifulsoup library in python
- SpaceX API Data columns

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

- Wikipedia Data columns

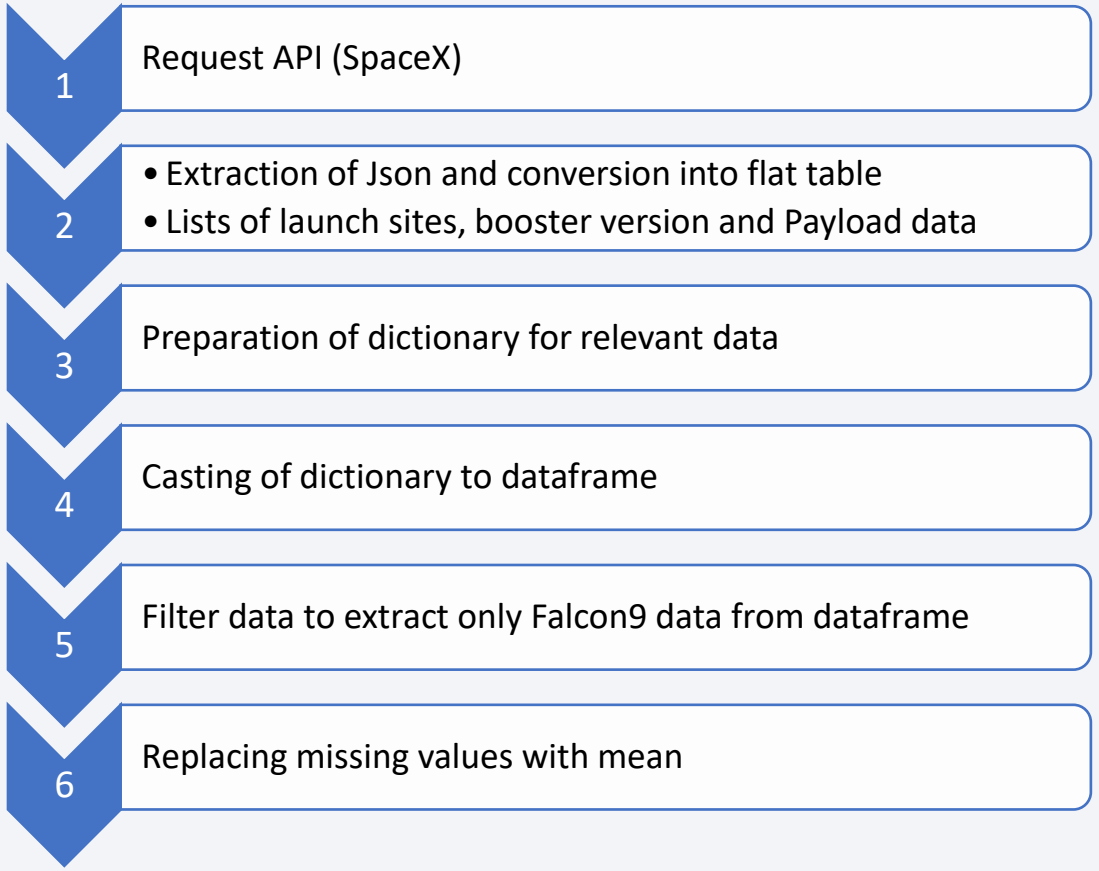
Flight No., Launch site, Payload, PayloadMass, Orbit, Customer, Launch outcome, Version Booster, Booster landing, Date, Time

Data Collection – SpaceX API

- Data collection approach (API)

- Github

https://github.com/hamzahshabbir96/IBM-Data-Science-project/blob/master/Project%20mission%20space.ipynb



Data Collection - Scraping

- Data collection approach (Web scraping)

- Github

https://github.com/hamzahshabbir96/IBM-Data-Science-project/blob/master/Data%20collection%20with%20web%20scraping.ipynb

1. Request Wikipedia html of SpaceX
2. Parsing request in beautifulsoup library of python
3. Find relevant table and loop over them to extract data
4. Prepare list of each column from iteration
5. Creation of dictionary with data column lists
6. Conversion of list to dataframe

Data Wrangling

- Creation of data column with training label with outcomes where successful is mapped with 1 and failure with 0
- Outcome has columns mission outcome and landing location
- Mapping of different features with True in it with 1 and all other with 0
- Github

https://github.com/hamzahshabbir96/IBM-Data-Science-project/blob/master/Data%20wrangling.ipynb

EDA with Data Visualization

- Carried out Exploratory data analysis on different relations of data column
- Different types of plot such as scatter plot, line plot and bar plots were used to see relationship between different variables and to see trend
- Github

https://github.com/hamzahshabbir96/IBM-Data-Science-project/blob/master/EDA%20with%20Data%20visualization.ipynb

EDA with SQL

- Data were loaded and stored into IBM DB2 database on cloud
- Ran different queries by integrating SQL API with Python
- Different analysis were done using queries such as mission outcomes, various payloads size of customers etc. get deep understanding of data

- Github

https://github.com/hamzahshabbir96/IBM-Data-Science-project/blob/master/EDA%20with%20SQL.ipynb

Build an Interactive Map with Folium

- Applied folium library to build an interactive map with information such as Launch sites , successful and unsuccessful landing with nearby location such as Railway, highway etc
- Explain why you added those objects

Github

https://github.com/hamzahshabbir96/IBM-Data-Science-project/blob/master/Interactive%20Visual%20Analytics.ipynb

Build a Dashboard with Plotly Dash

- Using plotly dash created an interactive dashboard where user have option to select certain parameter.
- Scatter plot takes input in the form of drowpdown with All site or individual site as options. Other input is a slider to select payload mass between 0 and 10000 kg.
- Pie chart created to visualize success rate

Github

https://github.com/hamzahshabbir96/IBM-Data-Science-project/blob/master/dash.py

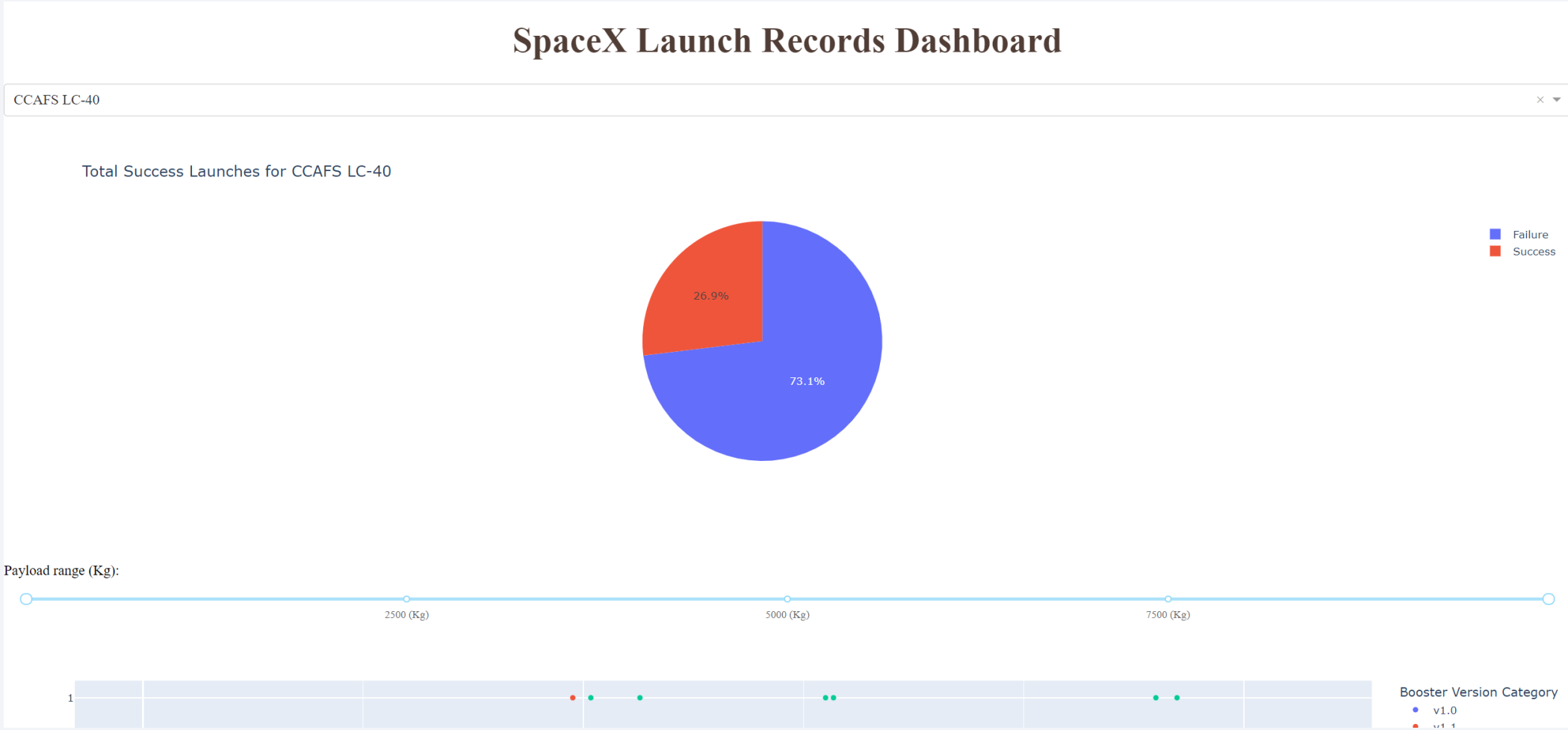
Predictive Analysis (Classification)

- Summary of approach
- Github:

https://github.com/hamzahshabbir96/IBM-Data-Science-project/blob/master/Machine%20learning%20prediction.ipynb

1. Prepare dataset with label column class
2. Feature scaling
3. Splitting of test and train dataset
4. Performance of model on test data
5. Grid search to tune hyper parameter
6. Confusion matrix and bar plot for model comparison

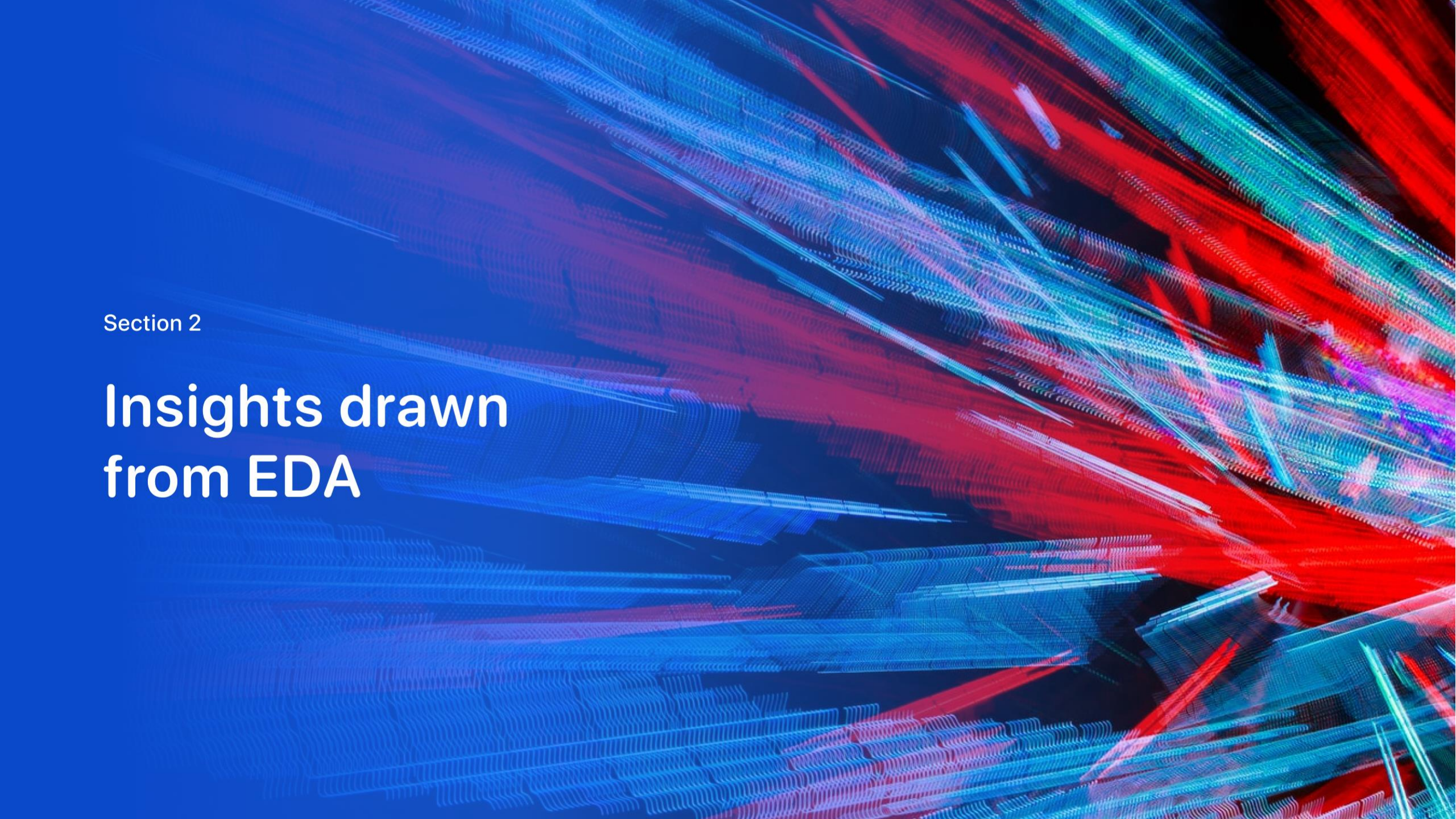
Results



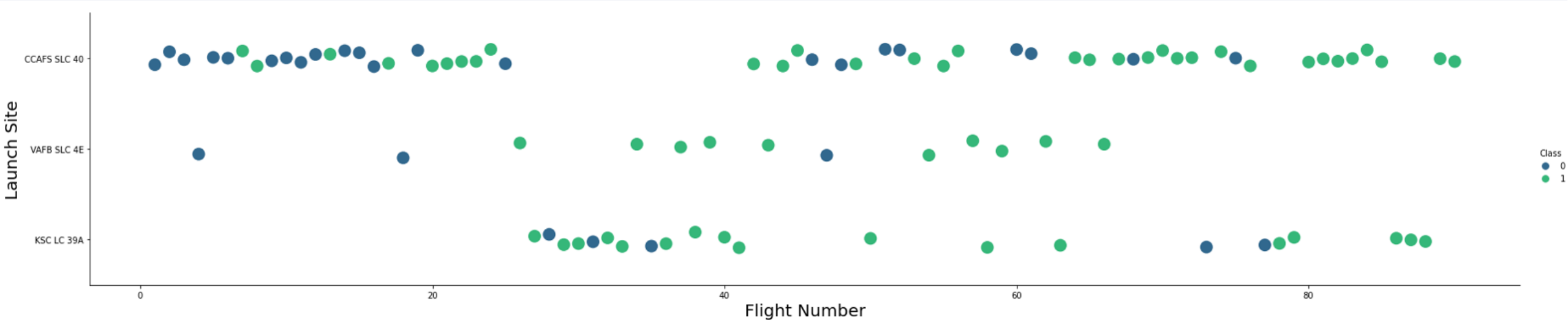
- Screenshot of interactive dashboard created using plotly dash

Section 2

Insights drawn from EDA

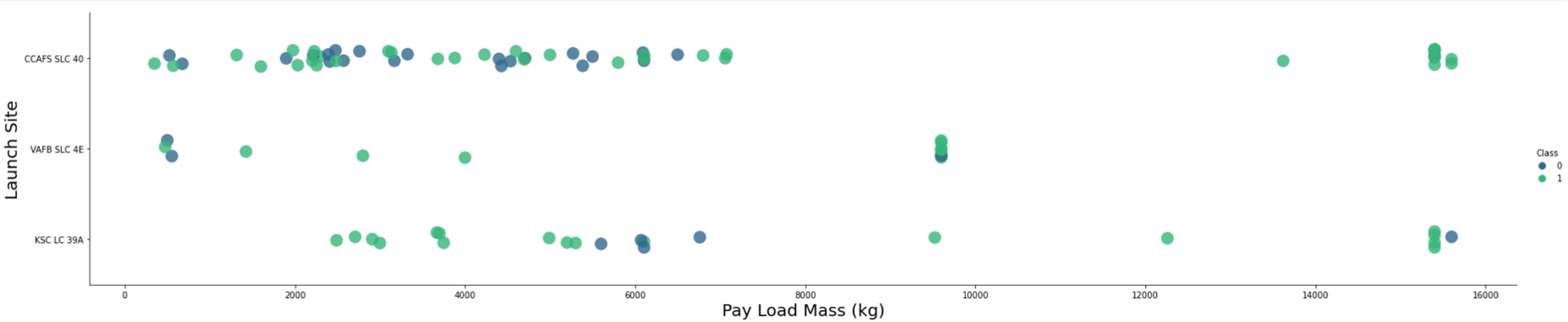


Flight Number vs. Launch Site



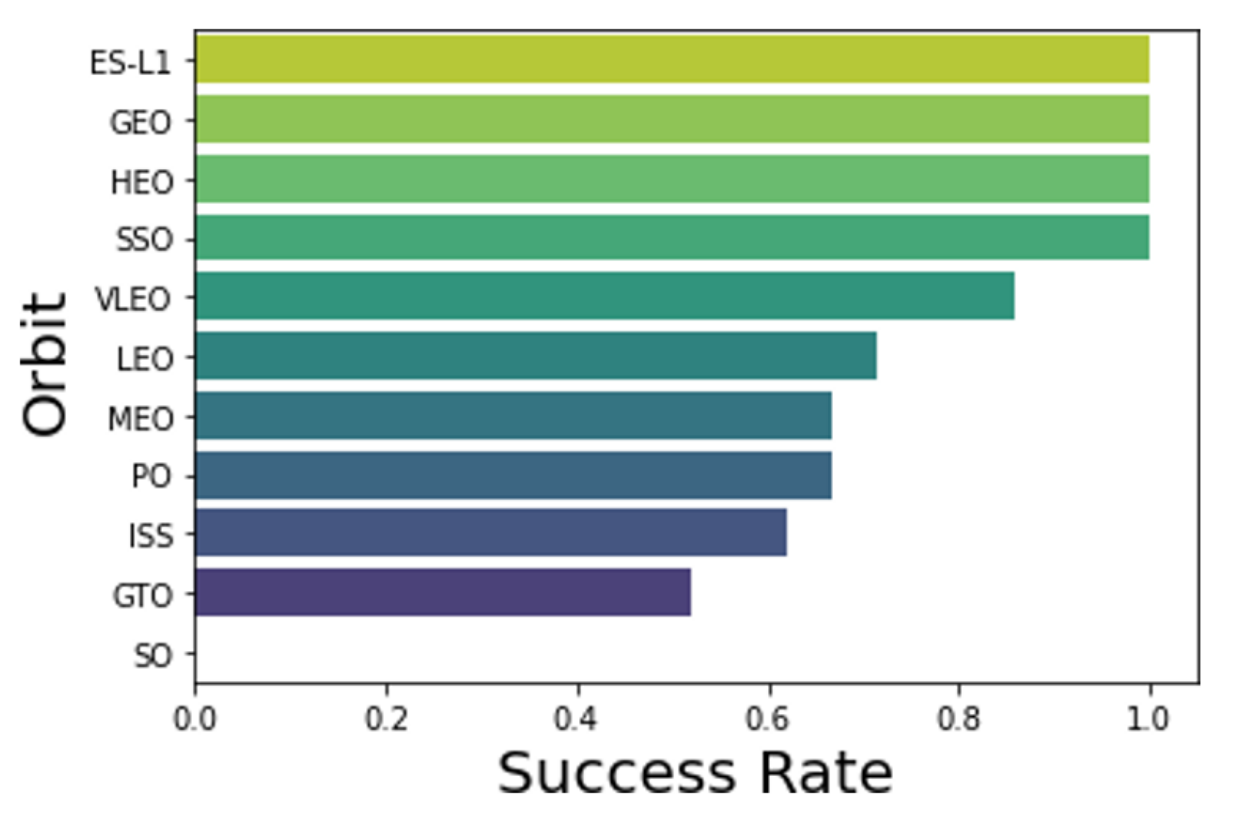
Above plots show relation of launch site with flight number. It can be concluded that CCAFS SLC 40 has maximum number of launch

Payload vs. Launch Site



Above plots show relation between launch site and payload where green dot shows successful launch and blue dot shows unsuccessful launch

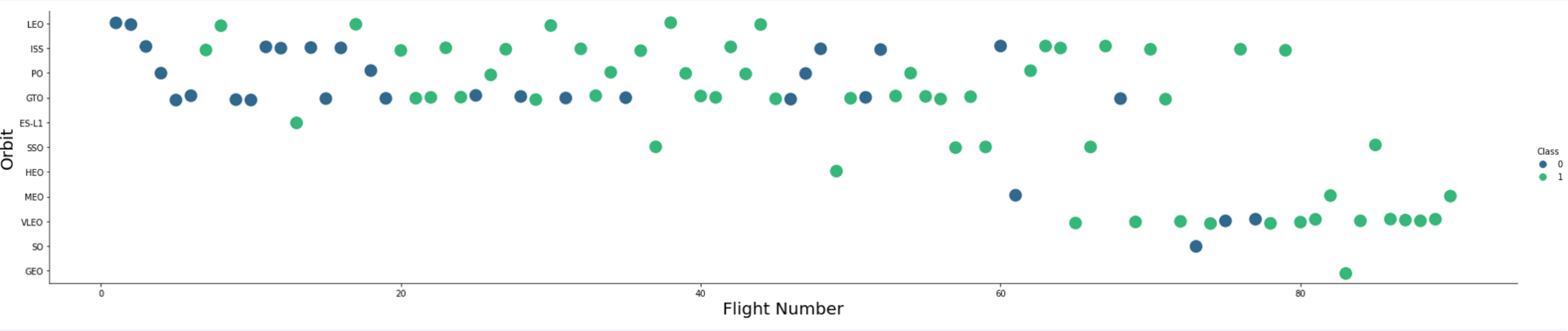
Success Rate vs. Orbit Type



Above plots show bar plot of different orbit with success rate:

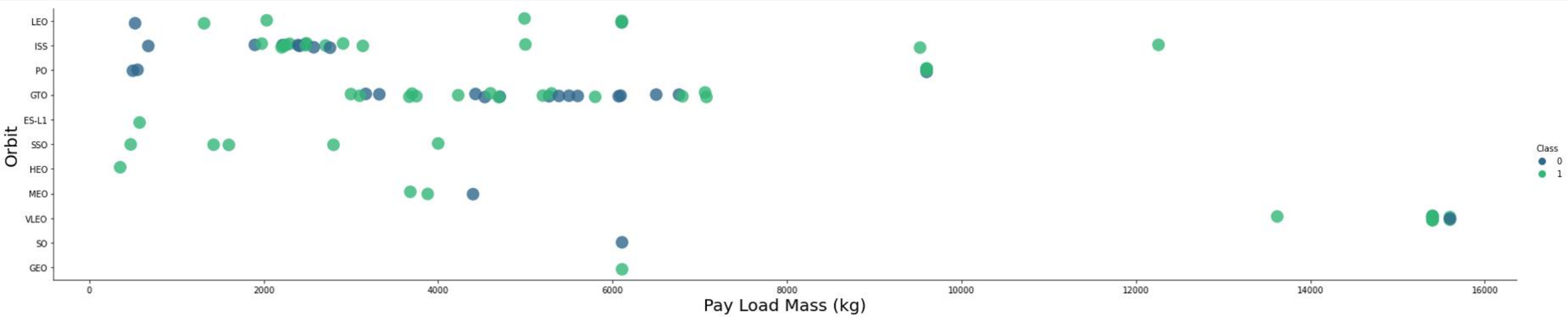
ES-L1, GEO, HEO and SSO has record of 100 % success rate while GTO has lowest success rate with around 55%

Flight Number vs. Orbit Type



Above plots show relation of orbit with flight number

Payload vs. Orbit Type

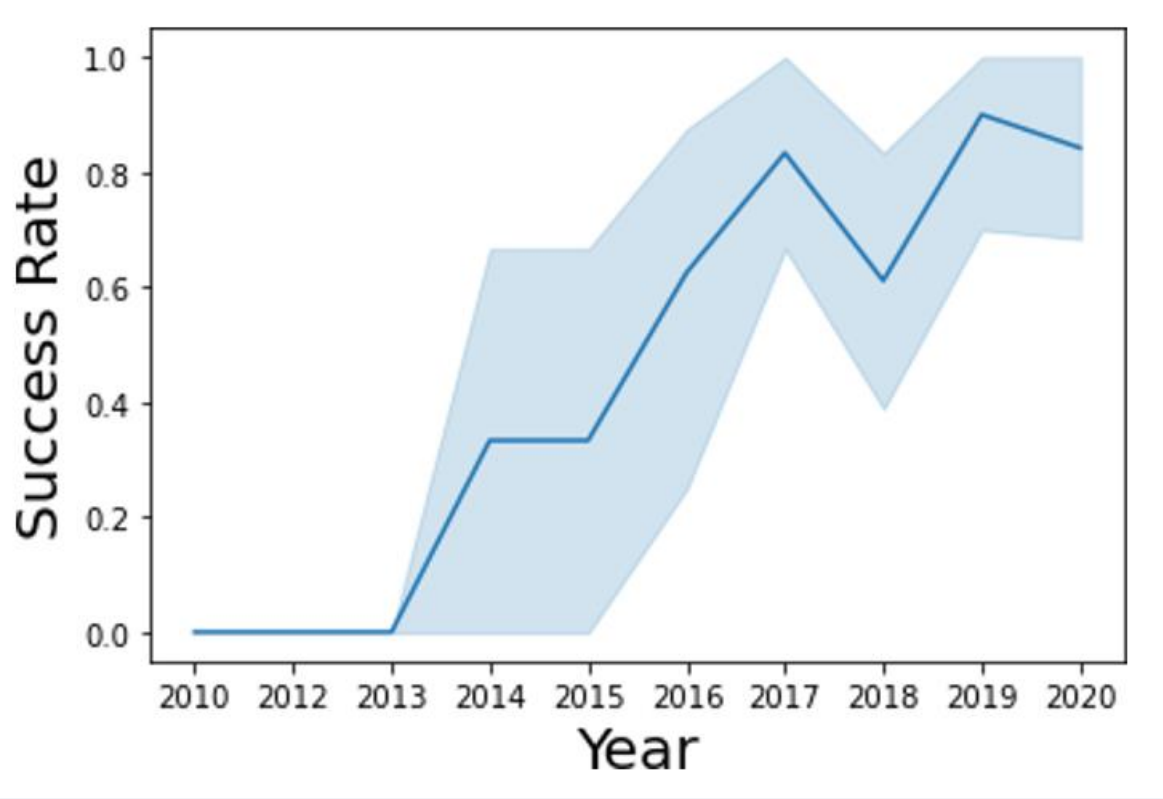


Above plots show relation of orbit with payload

ISS having high payload resulted into successful launch

Size of pyload doesnot mean unsuccessful launch

Launch Success Yearly Trend



Above plots show trend in success rate, it shows success rate has improved significantly in recent years

All Launch Site Names

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

- Used distinct command of sql to get unique launch site name

Launch Site Names Begin with 'CCA'

```
In [8]: %sql select * from SPACEXDATA where launch_site like '%CCA%' limit 5
```

```
 * ibm_db_sa://mqh88227:***@55fbc997-9266-4331-afd3-888b05e734c0.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31929/BLUDB
Done.
```

Out[8]:

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	None	0	LEO	SpaceX	Success	Failure (parachute)
2010-12-08	15:43:00	F9 v1.0 B0004	CCAFS LC-40	None	0	LEO (ISS)	NASA (COTS) NRO	Success	Failure (parachute)
2012-05-22	07:44:00	F9 v1.0 B0005	CCAFS LC-40	None	525	LEO (ISS)	NASA (COTS)	Success	No attempt
2012-10-08	00:35:00	F9 v1.0 B0006	CCAFS LC-40	None	500	LEO (ISS)	NASA (CRS)	Success	No attempt
2013-03-01	15:10:00	F9 v1.0 B0007	CCAFS LC-40	None	677	LEO (ISS)	NASA (CRS)	Success	No attempt

- Query dataset where launch site name contained ‘CCA’ in it and limited result to 5

Total Payload Mass

```
%sql select sum(payload_mass__kg_) as sum from SPACEXDATA where customer like 'NASA (CRS)'
```

```
 * ibm_db_sa://mqh88227:***@55fbc997-9266-4331-afd3-888b05e734c0.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31929/BLUDB
Done.
```

Out[10]:

SUM
45596

- Calculated sum of all payload mass to get sum of total payload
- Filtered customer for NASA only by string matching

Average Payload Mass by F9 v1.1

```
%sql select avg(payload_mass__kg_) as Average from SPACEXDATA where booster_version like 'F9 v1.1%'
```

```
 * ibm_db_sa://mqh88227:***@55fbc997-9266-4331-afd3-888b05e734c0.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31929/BLUDB
Done.
```

ut[12]:

average
2534

- For booster version F9 v1.1, average payload mass was 2534

First Successful Ground Landing Date

```
%sql select date from SPACEXDATA where mission_outcome like 'Success' ORDER BY DATE LIMIT  1
```

```
 * ibm_db_sa://mqh88227:***@55fbc997-9266-4331-afd3-888b05e734c0.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31929/BLUDB
Done.
```

Out[17]:

DATE
2010-06-04

- First successful landing was in 4th June 2010

Total Number of Successful and Failure Mission Outcomes

List the total number of successful and failure mission outcomes

```
3]: %sql SELECT mission_outcome, count(*) as Count FROM SPACEXDATA GROUP by mission_outcome ORDER BY mission_outcome
```

```
 * ibm_db_sa://mqh88227:***@55fbc997-9266-4331-afd3-888b05e734c0.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31929/BLUDB
Done.
```

Out[23]:

mission_outcome	COUNT
Failure (in flight)	1
Success	99
Success (payload status unclear)	1

- Success rate is huge with almost 99%

Boosters Carried Maximum Payload

```
%sql select booster_version from SPACEXDATA where
payload_mass_kg_=(select max(payload_mass_kg_) from SPACEXDATA)
```

The maximum payload was 15600 kg

2015 Launch Records

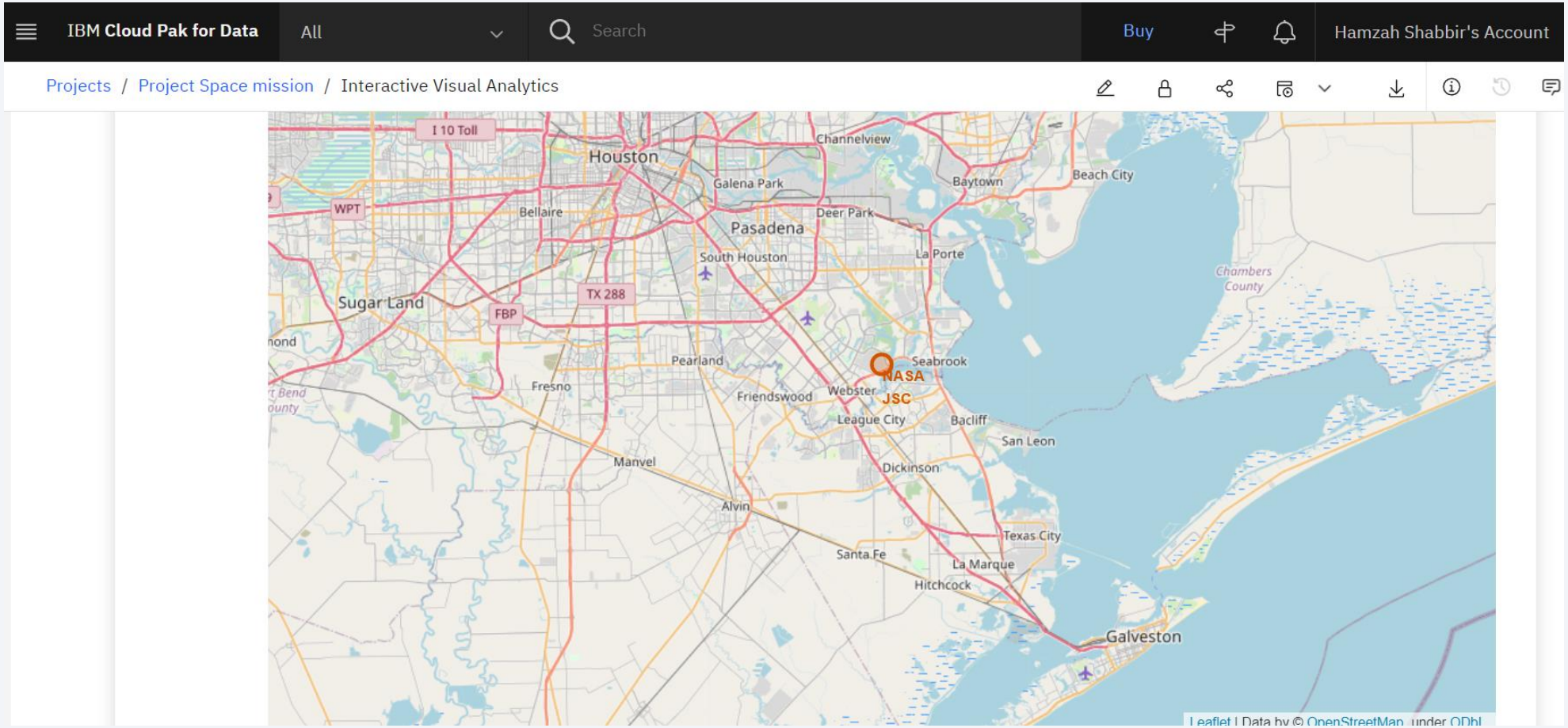
- List the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015

- Present your query result with a short explanation here

Section 4

Launch Sites Proximities Analysis

Launch site location



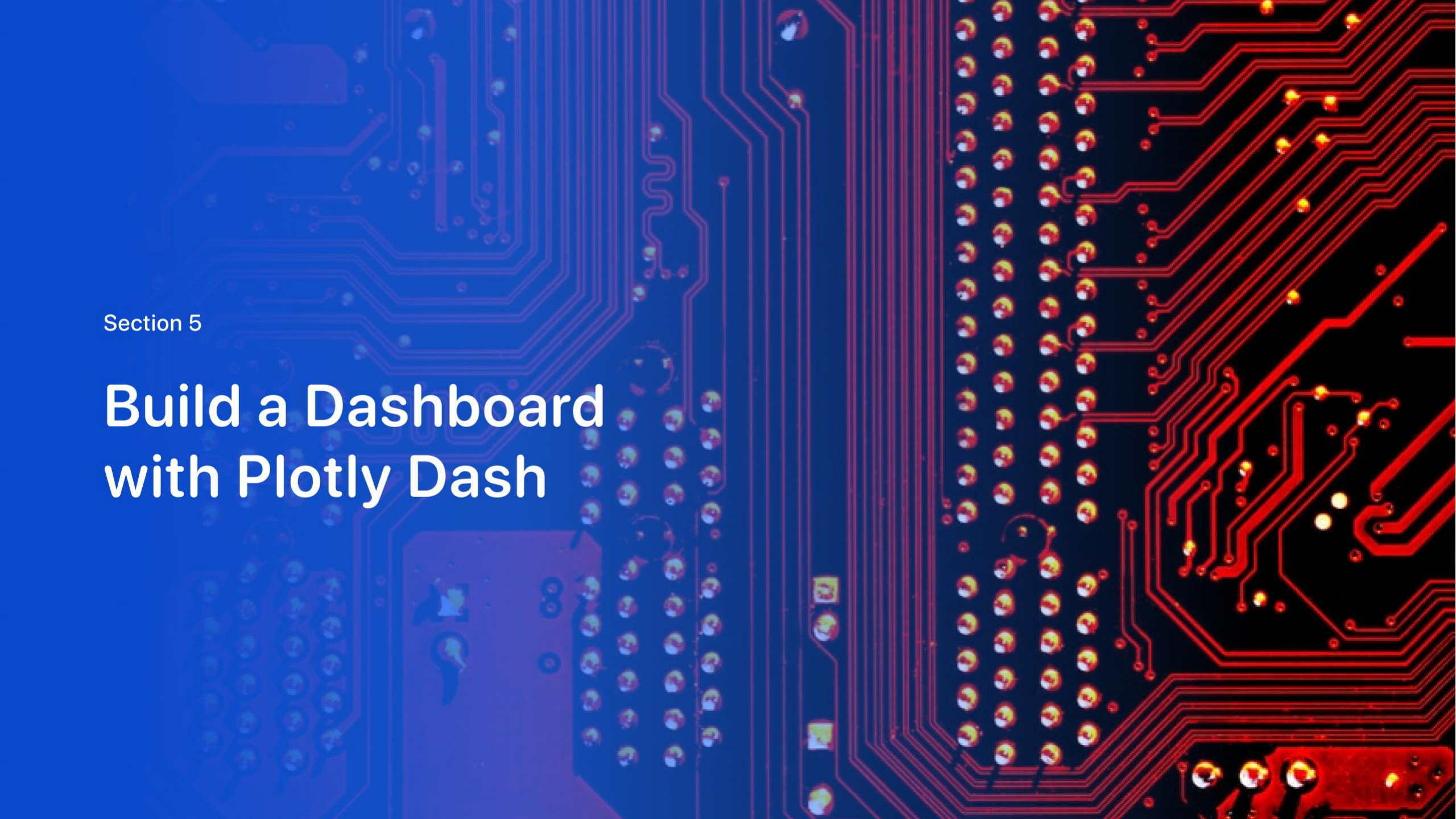
Above map show launch site location in the form of circle

Launch markers

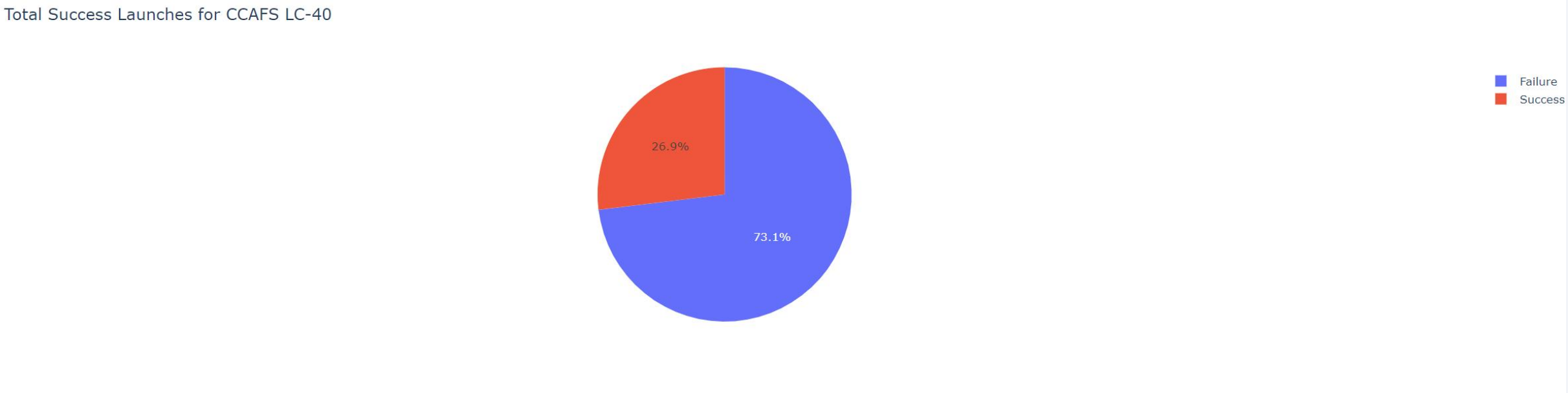
- Green color shows successful landing and red color shows unsuccessful landing

Section 5

Build a Dashboard with Plotly Dash



Pie chart for successful launches



- Above pie chart shows successful and unsuccessful launches

Payload mass vs success vs Booster version

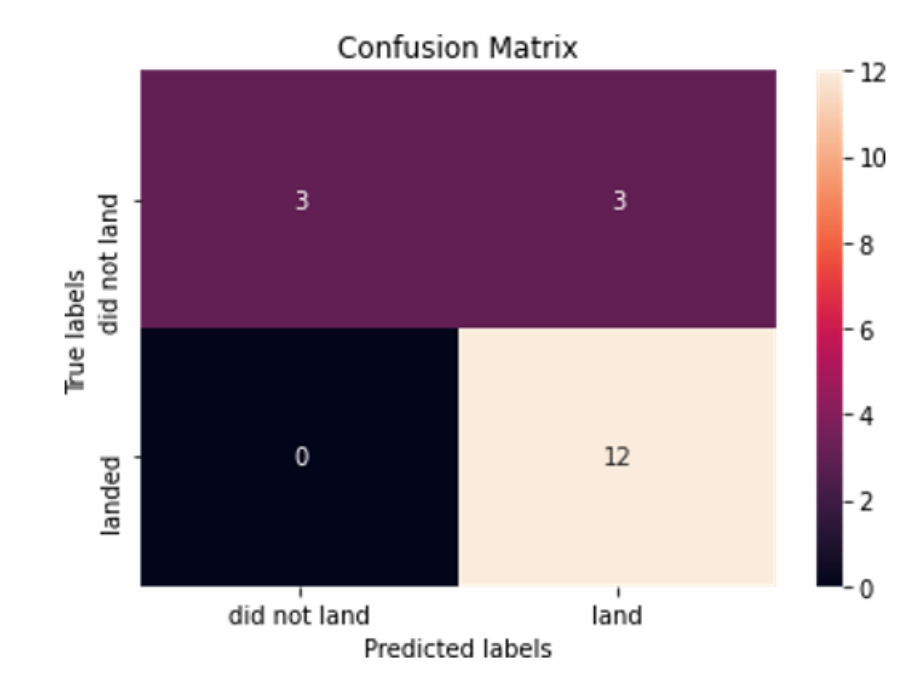
Section 6

Predictive Analysis (Classification)

Classification Accuracy

- Since model were tuned for best hyper parameter, all model were able to give best result on given dataset
- All models has accuracy between 80-85%
- Accuracy also depends on test size, in our case test size was very small

Confusion Matrix



Model predicted 12 successful landing when there was actual landing.

There was also false negative of 3 set

Predicted accurately number of unsuccessful land

Conclusions

- Produced interactive dashboard and visual analysis to conclude some of the important trends
- Model successfully predicted if landing will be successful with great accuracy
- Machine learning model with accuracy of 82% was developed
- Through this model we can find out whether launch should be made or not
- Having more data will make model more accurate

Appendix

- Github repository:

https://github.com/hamzahshabbir96/IBM-Data-Science-project

Thank you!