IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

D Giridhar

Aug 14, 2022

Outline

Executive Summary

Summary of methodologies

The objective of this thesis is to apply a complete data-driven analytic approach to predict favorable outcome to winning the space race with SpaceX landing data

It involves data collection, data wrangling, EDA with data visualization, EDA with SQL, Building interactive map with Folium, Building a Dashboard with Plotly Dash and Data Classification using Predictive Analysis methodologies

Summary of results

The results are summarized with Exploratory data analysis results, Interactive analytics demo (with screenshots) and Predictive analysis results

Introduction

Project background and Context

In the age of commercial space travelling, SpaceX advertises on its website that Falcon 9 rocket launches 260+% cheaper (with a cost of 62 million dollars while other providers cost upward of 165 million dollars each), much of the savings is because SpaceX can reuse the first stage

Problems to find answers

1) Will SpaceX land successfully in first stage?
2) What are the parameters that influence its optimal success in landing?
3) How to leverage the insight to potentially bid against SpaceX for rocket launch?

Section 1

Methodology

Methodology

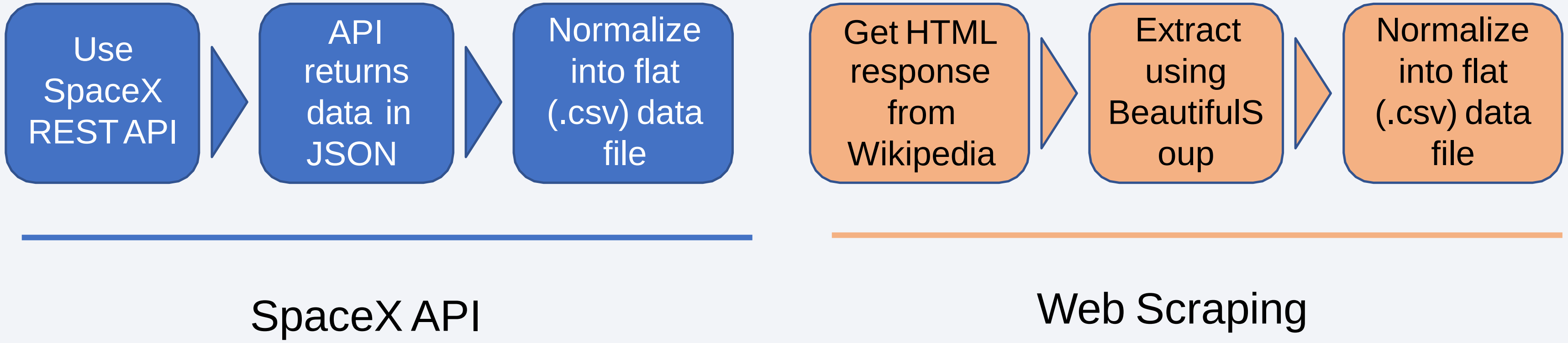
- Data collection methodology:
  - The data was collected by sending get request to SpaceX API and by web scraping Falcon 9 and Falcon Heavy Launches Records from Wikipedia.
- Perform data wrangling
  - Dealing with missing values, creating new columns, dropping irrelevant columns and visualizing through Panda's data frames
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Chose the best performing model from testing four different models of classification: Logistic regression, Tree, SVM and KNN

Data Collection

How was the data collected?

- The data is collected from the SpaceX REST API, and the API will give us data about launches including information about the rocket used, payload delivered, launch specifications, landing specifications, and landing outcome.
- Another data is collected from Wikipedia using BeautifulSoup

Data collection process



Data Collection – SpaceX API

SpaceX REST calls using key phrases and flowcharts

Request and parse the SpaceX launch data using the GET request

Decode the response content as a Json using json ()
and turn it into a Pandas dataframe using.json_normalize

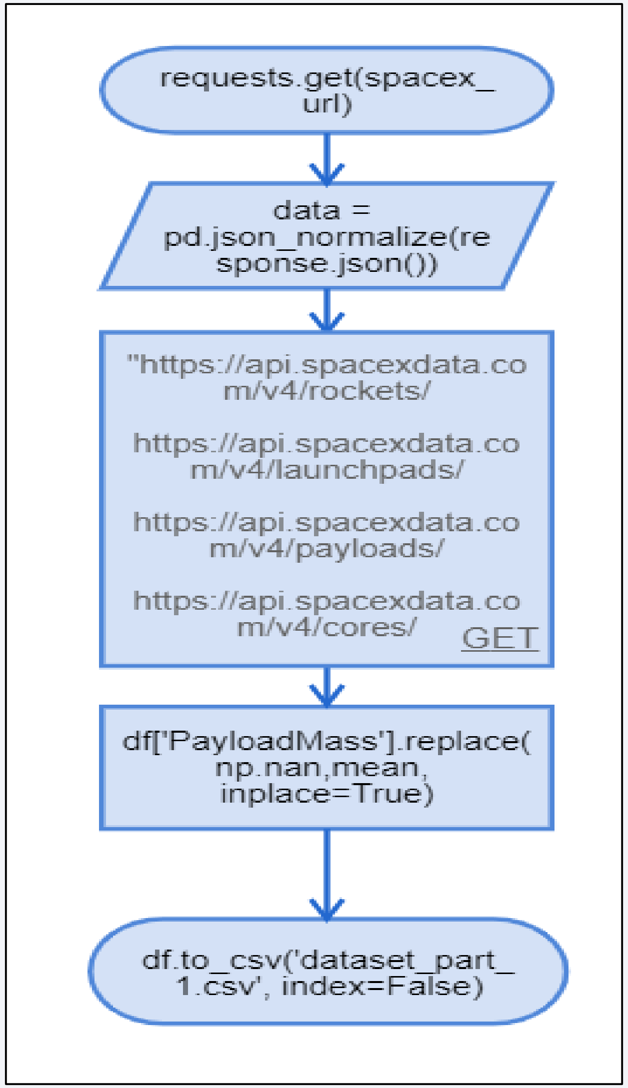
Use the API again to get information about the launches using the IDs given for each launch. Specifically, will be using columns

- Rocket
- Payloads
- Launchpad
- Cores

The mean and the replace () function to replace np.nan values in the data with the mean calculated

GitHub URL

https://github.com/Giridhard/Spacex-Launch/blob/main/Data%20Collection%20API.ipynb



Data Collection - Scraping

Web scraping process

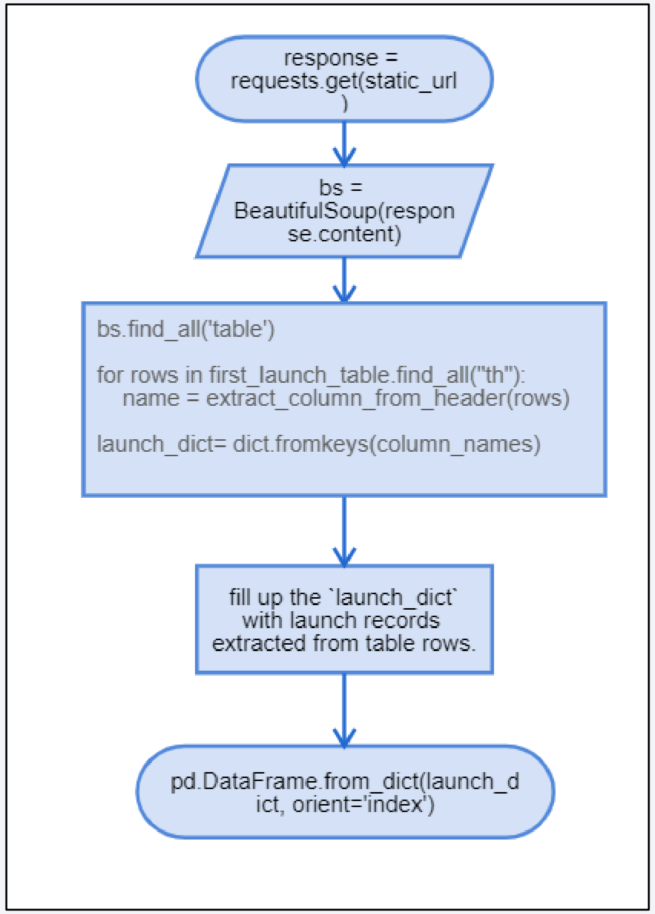
Extract a Falcon 9 launch records HTML table from Wikipedia

Parse the table and convert it into a Pandas data frame

TASK 1: Request the Falcon9 Launch Wiki page from its URL

TASK 2: Extract all column/variable names from the HTML table header

TASK 3: Create a data frame by parsing the launch HTML tables.



GitHub URL

https://github.com/Giridhard/Spacex-Launch/blob/main/Data%20Collection%20Using%20WebScraping.ipynb

Data Wrangling

Data Wrangling process

Data was processed using pandas data frame features

Data Wrangling process contains 5 steps for understanding data

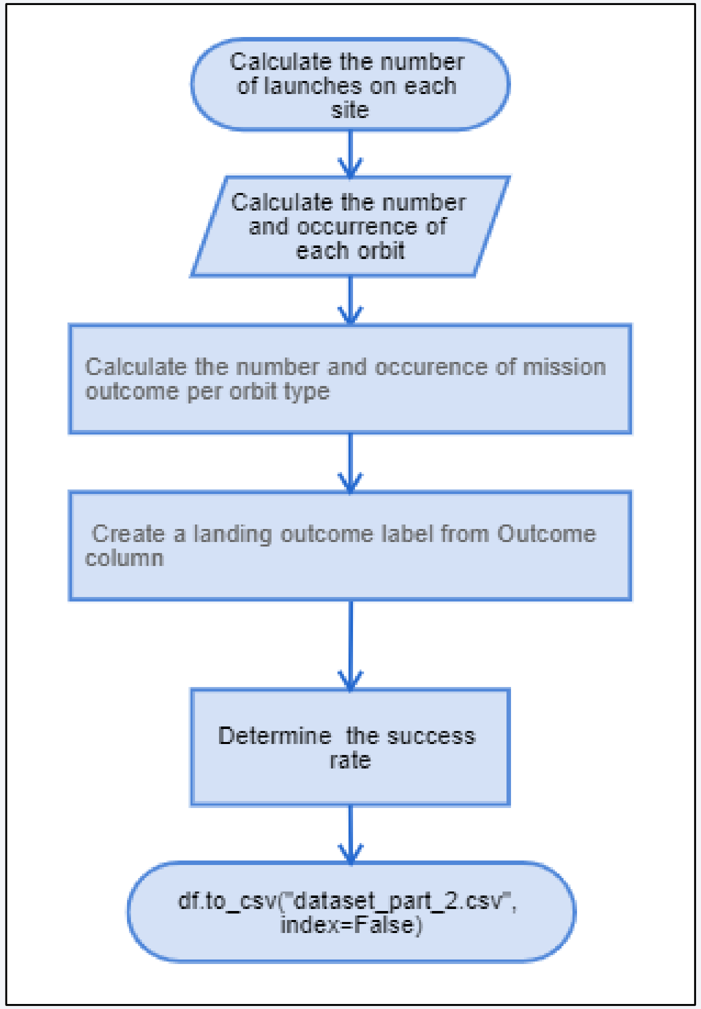
1) Data Analysis

2) Calculate number of launch on each site

3) Calculate the number of occurrence of each orbit

4) Calculate the number and occurrence of mission outcome per orbit type

5) Create a landing outcome label from Outcome column



GitHub URL

https://github.com/Giridhard/Spacex-Launch/blob/main/Data%20Wrangling%20EDA.ipynb

EDA with Data Visualization

Visualization charts and reason for the choice of the chart

- Scatter Graphs helps to visualize the data - show the data pattern & identify correlation between variables. Scatter plots consist of a larger body of data - Fight Number vs. Payload Mass, Fight Number vs. Launch Site, Payload vs. Launch Site, Orbit vs. Fight Number, Payload vs. Orbit Type, Orbit vs. Payload Mass
- Bar Graph makes it easy to compare sets of data between different groups immediately. 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. Mean vs. Orbit
- Line Graphs are useful in that they show data variables and trends very clearly and can help to make predictions about the results of data not yet recorded

GitHub URL

https://github.com/Giridhard/Spacex-Launch/blob/main/EDA%20with%20Data%20Visualization.ipynb

EDA with SQL

SQL Queries performed to explore 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 v1.1
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 failed landing outcomes in drone ship, their booster versions, and launch site names for 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 0 3 20, in descending order

GitHub URL

https://github.com/Giridhard/Spacex-Launch/blob/main/EDA_SQL.ipynb

Build an Interactive Map with Folium

Folium map Summary

- Marked all launch sites visually on a map with latitude, longitude coordinates circled
- Marked the success/failed launches for each site, assigned the data frame launch_outcomes (failures, successes) to classes 0 and 1 with Green and Red markers on the map in a MarkerCluster()
- Calculated the distance from the launch site to various land markers. Lines are drawn on the map to measure distance to landmarks
- This helps to answer below questions easily:
  - *Are launch sites in close proximity to railways?*
  - *Are launch sites in close proximity to highways?*
  - *Are launch sites in close proximity to coastline?*
  - *Do launch sites keep certain distance away from cities?*

GitHub URL

https://github.com/Giridhard/Spacex-Launch/blob/main/Interactive%20Visual%20Analytics%20with%20Folium%20lab.ipynb

Build a Dashboard with Plotly Dash

Dashboard Summary

- Plotly is a python wrapper on the JavaScript library 'leaflet'. It enables us to interact with our data visualizations and host it as a website:

What's included:

Pie Chart: That shows number of launches from each launch site as well as number of successful and failed launches from those sites

Callback function for `site dropdown` as input, `success pie chart` as output

Callback function for `site dropdown` and `payload slider` as inputs, `success payload scatter chart` as output

Scatter Graph: Relationship between the success of a launch (Outcome) and Payload (in kg) for different versions of boosters

GitHub URL

https://github.com/Giridhard/Spacex-Launch/blob/main/Dashboard_With_plotly_dash.py

Predictive Analysis (Classification)

Classification Model Summary

Building Model

- Load dataset into NumPy and Pandas and Transform data.
- Split data into Training and Test datasets
- Standardize/Normalize all the parameters

Evaluating Model

- Check accuracy for each model

Get tuned hyperparameters for each type of algorithms and Confusion matrix for the same

Improvimg Model

- Use feature engineering and model tuning

Finding the best model

- Find the best performing classification model

Find the model with best accuracy score and also the best performing parameters/ variables for that model

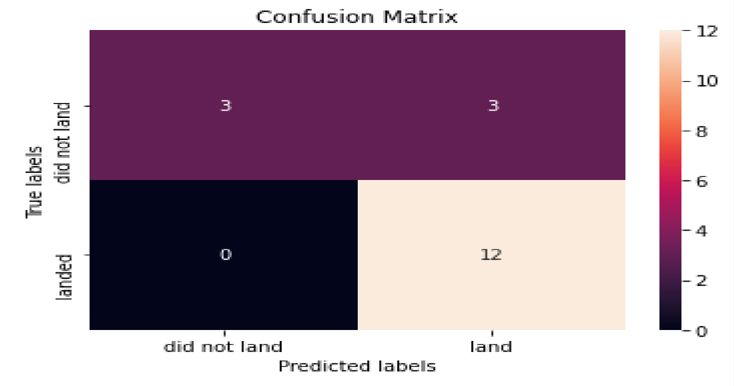
GitHub URL

https://github.com/Giridhard/Spacex-Launch/blob/main/Machine%20Learning%20Prediction.ipynb

Results

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

landing__outcome	COUNT
Success (drone ship)	2
Success (ground pad)	2
Failure (drone ship)	1
No attempt	1



Accuracy for Logistics Regression method: 0.8333333333333334

Accuracy for Support Vector Machine method: 0.8333333333333334

Accuracy for Decision tree method: 0.6111111111111112

Accuracy for K nearsdt neighbors method: 0.8333333333333334

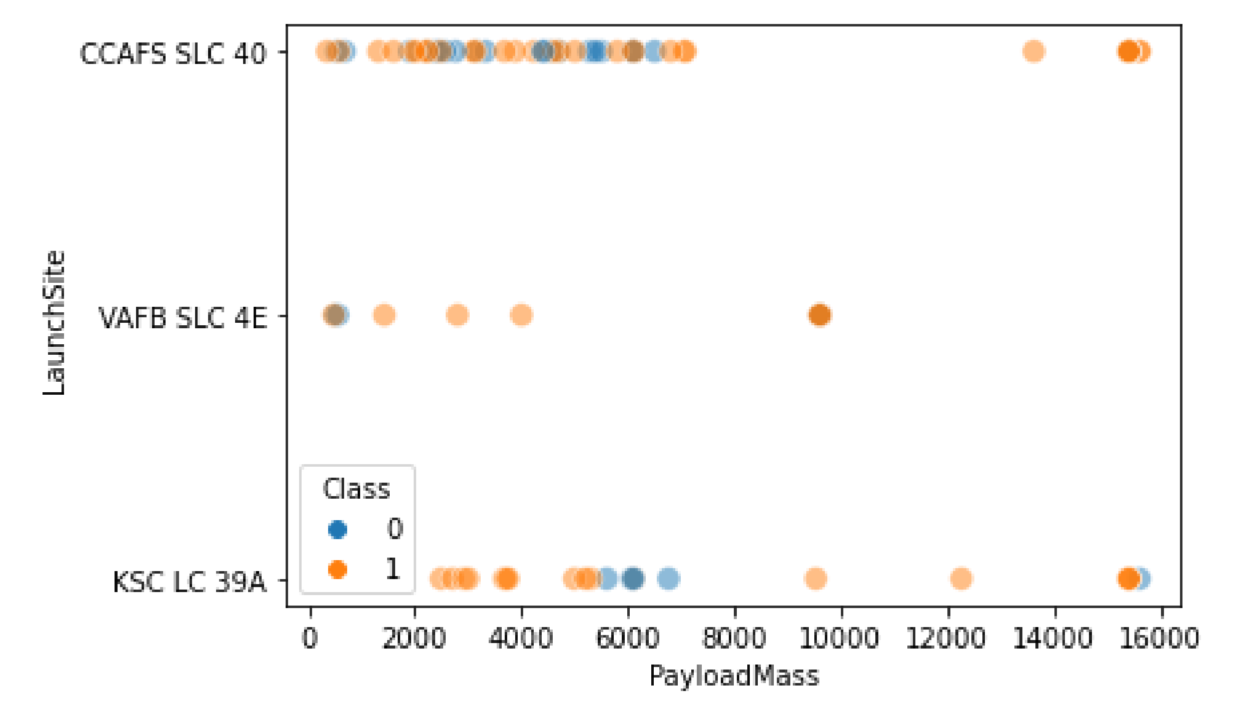
Section 2

Insights drawn from EDA

Flight Number vs. Launch Site

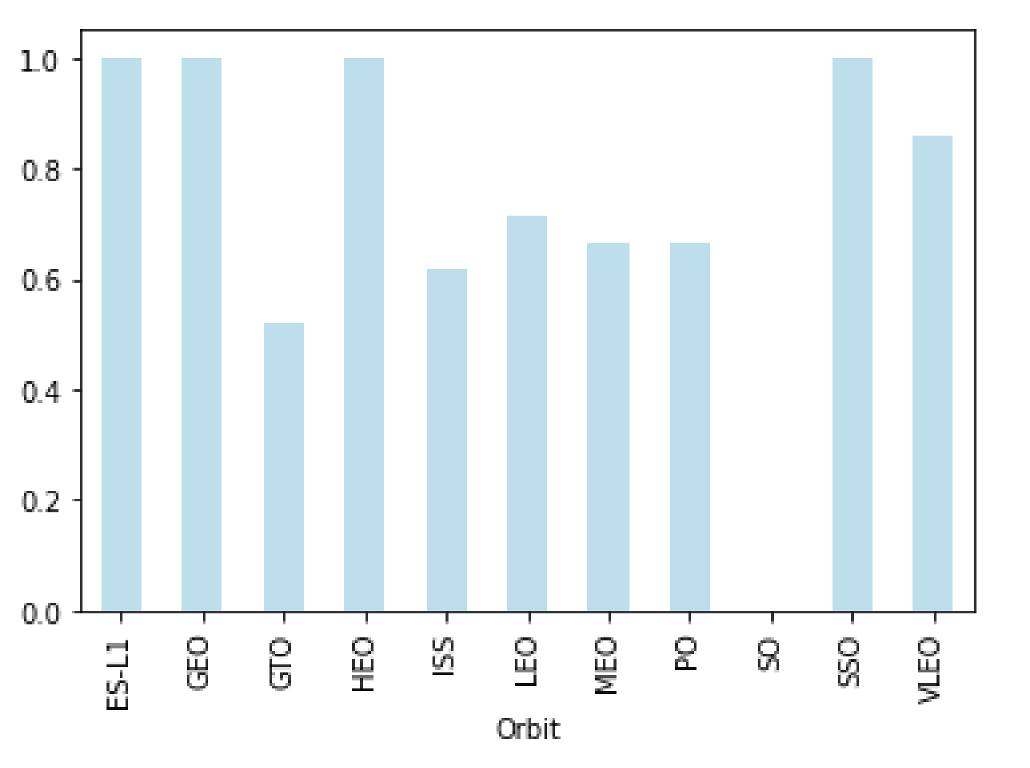
“Depicts the total launches by flight and launch site. The CCAFS SLC 40 has most launches across all flight numbers and have most failed launches in lower flight numbers and reduces as flight number increases”

Payload vs. Launch Site



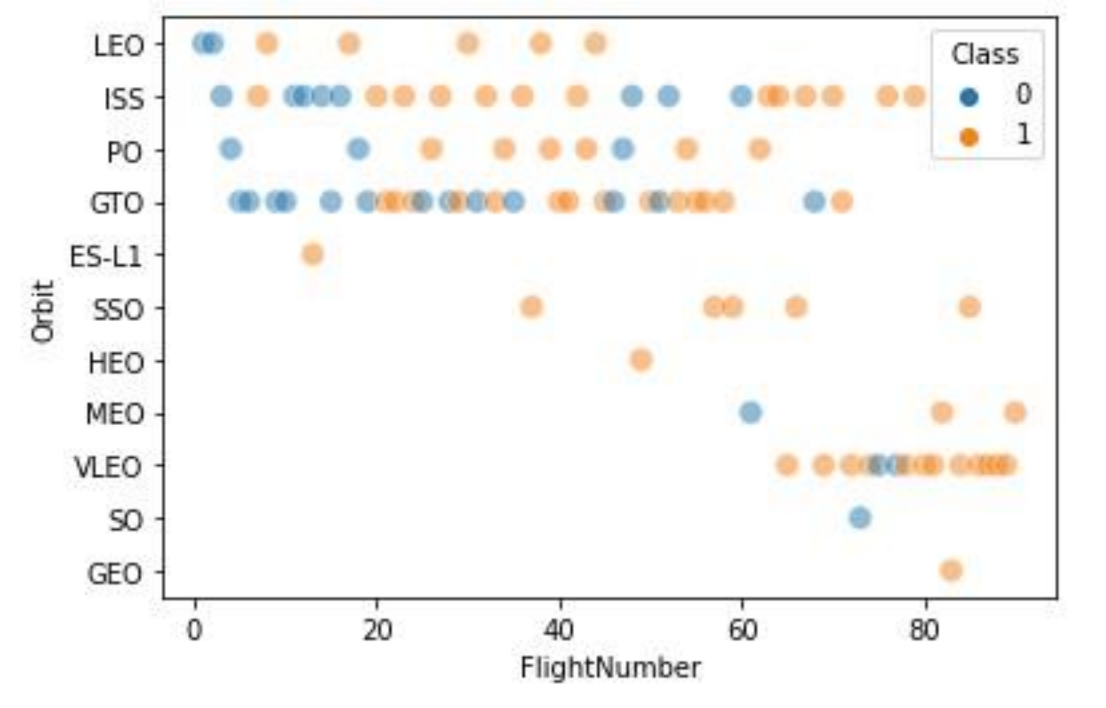
“No launches of VAFB SLC 4E beyond 10000 PayloadMass. No failed launches in the 8000-15000 PayloadMass range and just 1 failed launch of KSC LC 39A in the 8000-16000 range”

Success Rate vs. Orbit Type



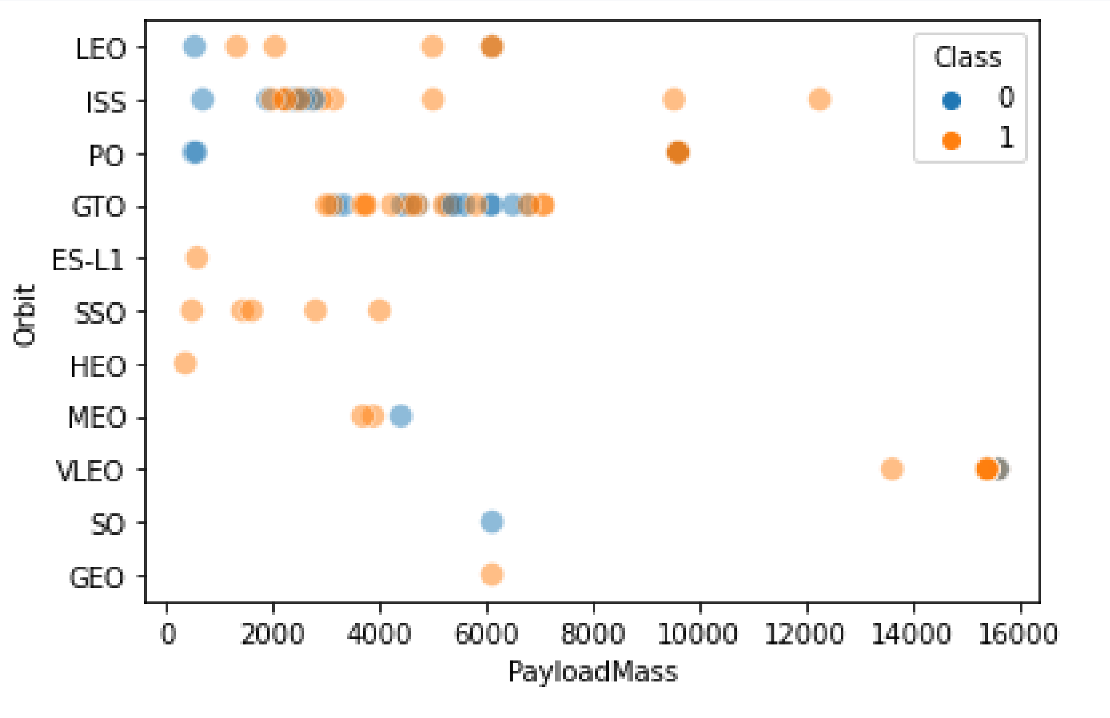
“ES-L1, GEO, HEO and SSO have 100% Success Rate”

Flight Number vs. Orbit Type



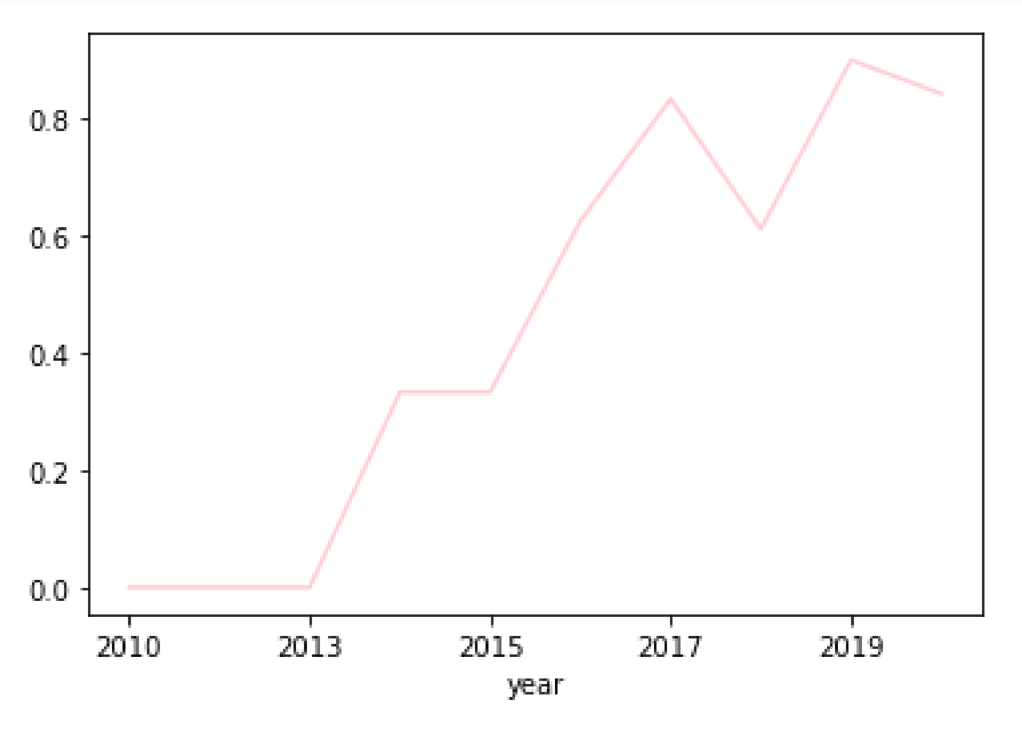
“LEO Orbit, SSO, HEO, GEO seems to have direct correlation to Success and GTO Orbit seems to have Failures at all Flight Numbers”

Payload vs. Orbit Type



"LEO, ISS, PO, SSO seems to have direct correlation to Success by PayloadMass. While SSO has high Success in lower PayloadMass, LEO, ISS, PO have higher Success in higher PayloadMass category"

Launch Success Yearly Trend



“Overall, there’s continuous increase in Success Rate from 2013-2020 years range (despite a flat 2014 and a small dip in 2018)”

All Launch Site Names

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

```
%sql SELECT DISTINCT(LAUNCH_SITE) FROM SPACEXTBL;
```

“Selecting unique launches sites from LAUNCH_SITE column from SpaceX launches dataset using SQL Query”

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	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attemp
2012-10-08	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attemp
2013-03-01	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attemp

```
%%sql
SELECT *
FROM SPACEXTBL
WHERE LAUNCH_SITE LIKE 'CCA%'
LIMIT 5;
```

“Selecting 5 records from SpaceX dataset with Launch Site names starting with CCA”

Total Payload Mass

45596

```
%%sql
SELECT SUM(PAYLOAD_MASS__KG_)
FROM SPACEXTBL
WHERE CUSTOMER = 'NASA (CRS)';
```

“Calculating total of all booster’s ‘Payload Mass’ from SpaceX dataset launches by ‘NASA (CRS)’ customer”

Average Payload Mass by F9 v1.1

```
2534
```

```
%%sql
SELECT AVG(PAYLOAD_MASS__KG_)
FROM SPACEXTBL
WHERE BOOSTER_VERSION LIKE 'F9 v1.1%';
```

"Calculating average payload mass of Booster version F9 v1.1 only"

First Successful Ground Landing Date

2015-12-22

```
%%sql
SELECT MIN(DATE)
FROM SPACEXTBL
WHERE LANDING__OUTCOME = 'Success (ground pad)';
```

“Calculating first successful ground landing date using MIN function on landing date column”

Successful Drone Ship Landing with Payload between 4000 and 6000

booster_version	landing__outcome	payload_mass__kg_
F9 FT B1021.2	Success (drone ship)	5300
F9 FT B1031.2	Success (drone ship)	5200
F9 FT B1022	Success (drone ship)	4696
F9 FT B1026	Success (drone ship)	4600

```
%%sql
SELECT DISTINCT(BOOSTER_VERSION), LANDING__OUTCOME, PAYLOAD_MASS__KG_
FROM SPACEXTBL
WHERE LANDING__OUTCOME = 'Success (drone ship)' AND PAYLOAD_MASS__KG_ BETWEEN 4000 AND 6000;
```

"Getting list of booster versions which have successful drone ship landing and their payload is in the range of 4000 to 6000 KG"

Total Number of Successful and Failure Mission Outcomes

successful_missions
61

failure_missions
10

```
%%sql
SELECT COUNT(LANDING__OUTCOME) AS SUCCESSFUL_MISSIONS
FROM SPACEXTBL
WHERE LANDING__OUTCOME LIKE 'Success%';
```

```
%%sql
SELECT COUNT(LANDING__OUTCOME) AS FAILURE_MISSIONS
FROM SPACEXTBL
WHERE LANDING__OUTCOME LIKE 'Failure%';
```

“Getting count of total successful and failure missions from SpaceX dataset based on landing outcome name starting with ‘Success’ for successful missions and ‘Failure’” for failure missions”

Boosters Carried Maximum Payload

booster_version	payload_mass__kg_
F9 B5 B1048.4	15600
F9 B5 B1048.5	15600
F9 B5 B1049.4	15600
F9 B5 B1049.5	15600
F9 B5 B1049.7	15600
F9 B5 B1051.3	15600
F9 B5 B1051.4	15600
F9 B5 B1051.6	15600
F9 B5 B1056.4	15600
F9 B5 B1058.3	15600
F9 B5 B1060.2	15600
F9 B5 B1060.3	15600

```
%%sql
SELECT DISTINCT(BOOSTER_VERSION), PAYLOAD_MASS__KG_
FROM SPACEXTBL
WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTBL)
```

"Obtained boosters carrying maximum payload by first selecting maximum of payload from SpaceX dataset and then selecting the booster version, payload using a subquery (query inside a query)"

2015 Launch Records

landing__outcome	booster_version	launch_site	date_year
Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40	2015
Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40	2015

```
%%sql
SELECT LANDING__OUTCOME, BOOSTER_VERSION, LAUNCH_SITE, YEAR(DATE) AS DATE_YEAR
FROM SPACEXTBL
WHERE LANDING__OUTCOME = 'Failure (drone ship)' AND YEAR(DATE) = '2015'
```

"Get list of failed landing_outcome along with the booster version and launch site for the year 2015 from SpaceX table using WHERE clause"

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

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

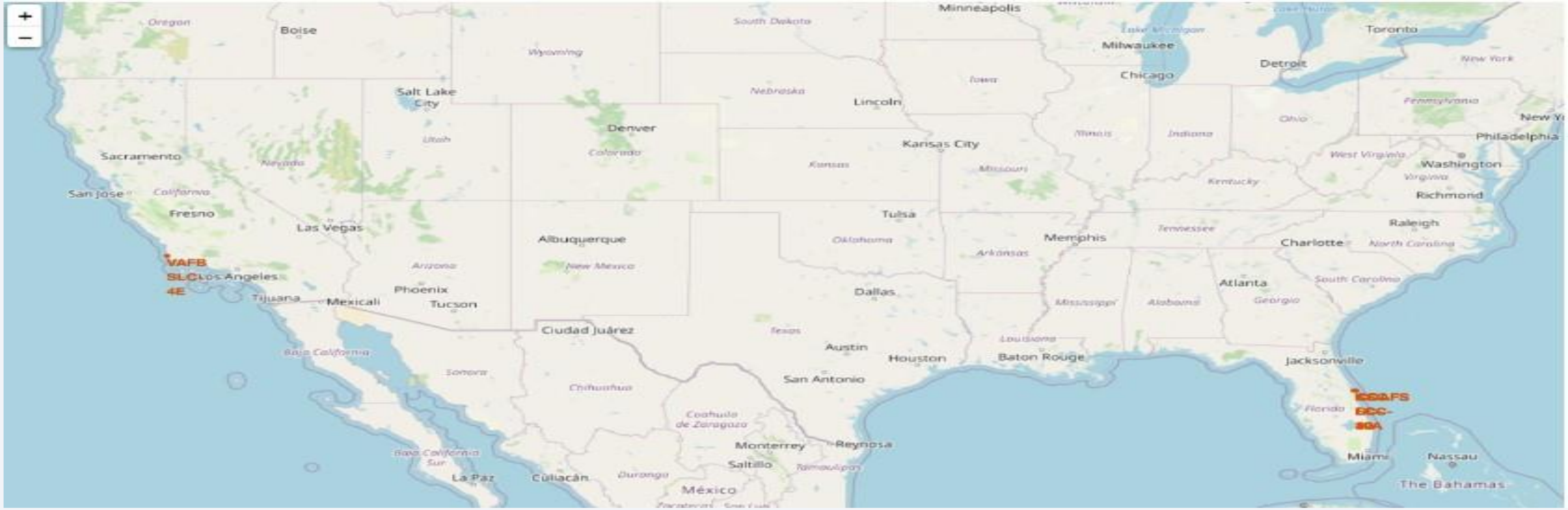
```
%%sql
SELECT LANDING__OUTCOME, COUNT(LANDING__OUTCOME) AS COUNT
FROM SPACEXTBL
WHERE DATE BETWEEN '2010-06-04'  AND '2017-03-20'
GROUP BY LANDING__OUTCOME
ORDER BY COUNT DESC
```

"Obtained count of landing_outcome in descending order occurred between above 2 dates by GROUP BY and ORDER BY clauses "

Section 4

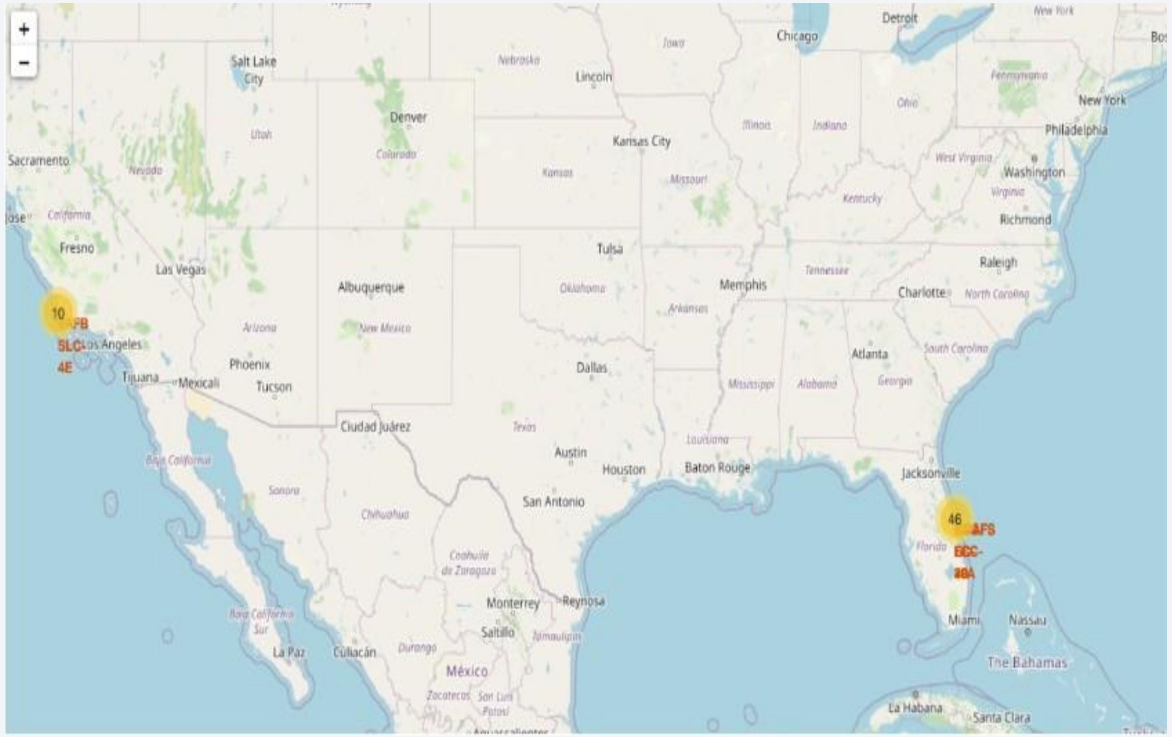
Launch Sites Proximities Analysis

Launch Sites On A Map



"SpaceX launch sites are in– Florida on the USA east coast and California on the USA west coast"

Sites And Launch Outcomes On A Map



“Green colored are the successful launches and red colored are the failed launches”

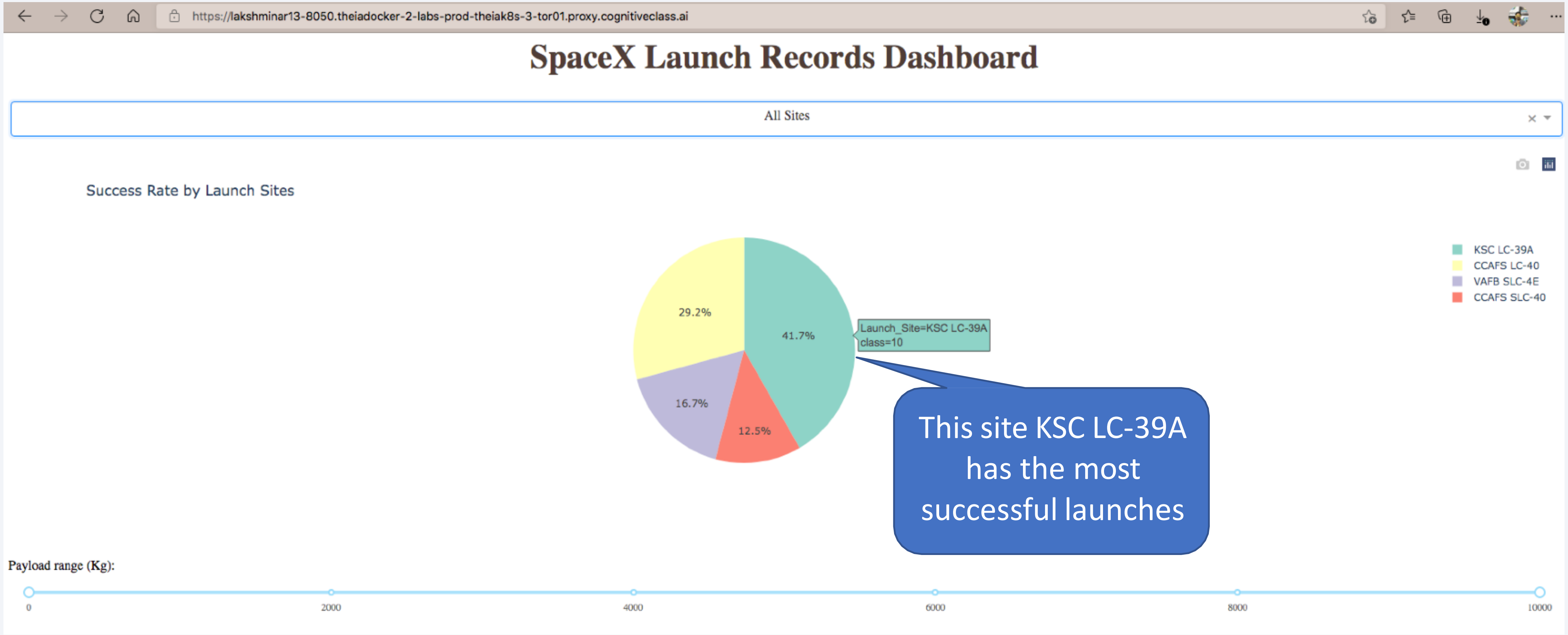
Launch Site Proximities On A Map

“Here, we can observe distance of launch sites from east coast, highways, key road, railway line visualized”

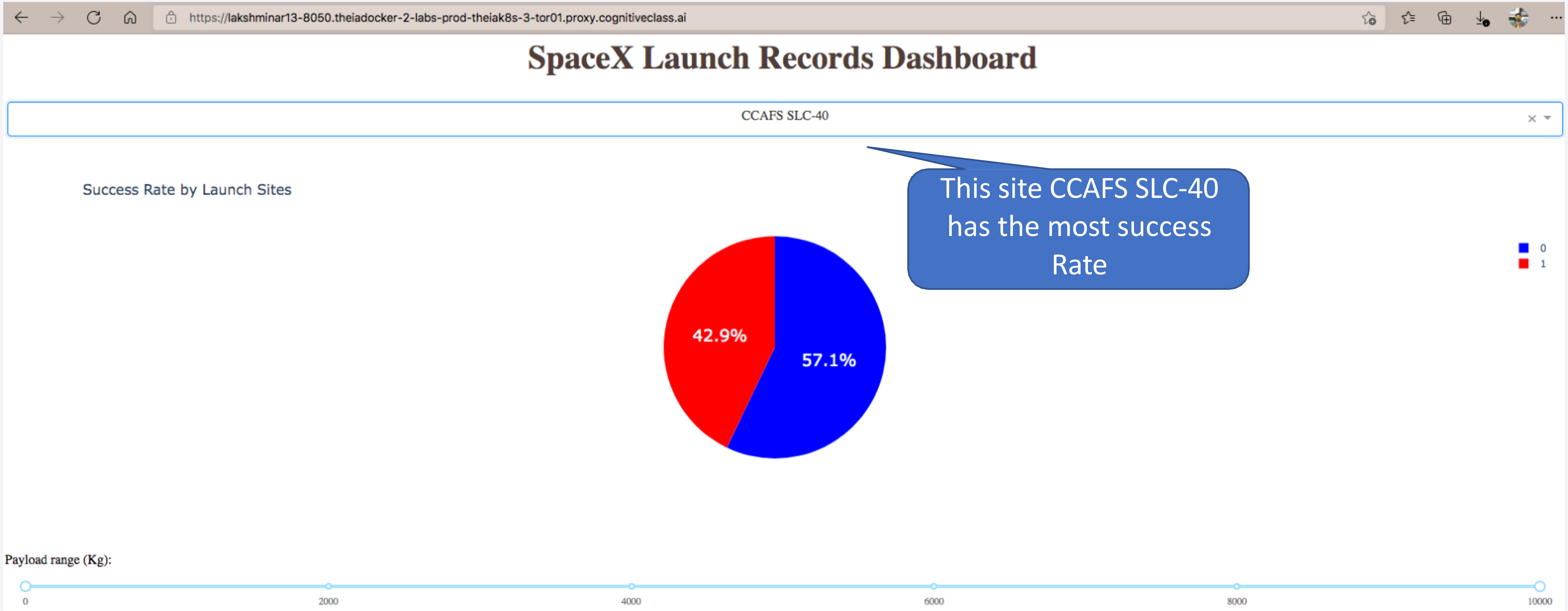
Section 5

Build a Dashboard with Plotly Dash

Success Rate by Launch Sites

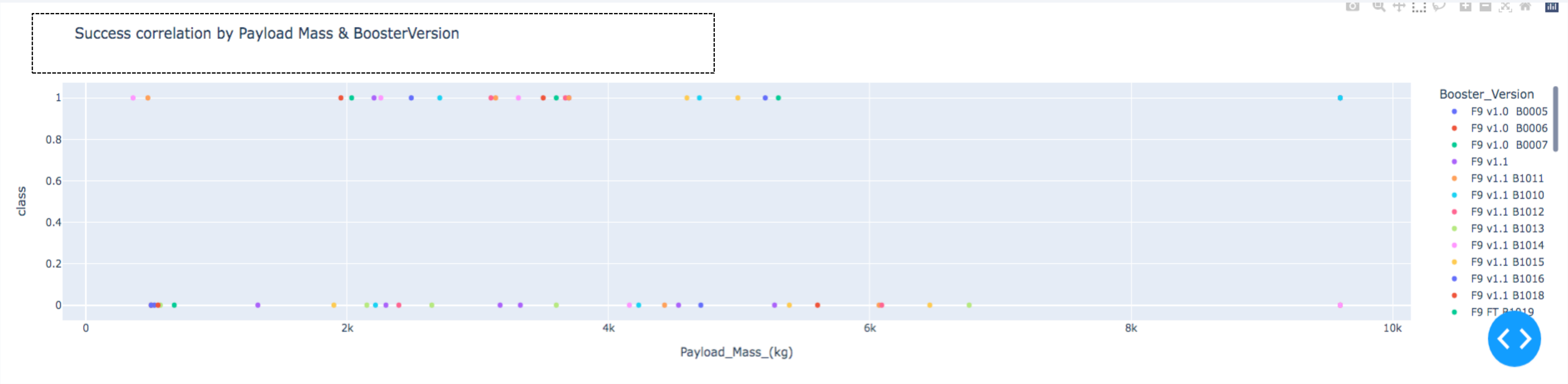


Most Successful Launch Site



Success by Payload Mass & Booster Version

Lower payload launches are more successful

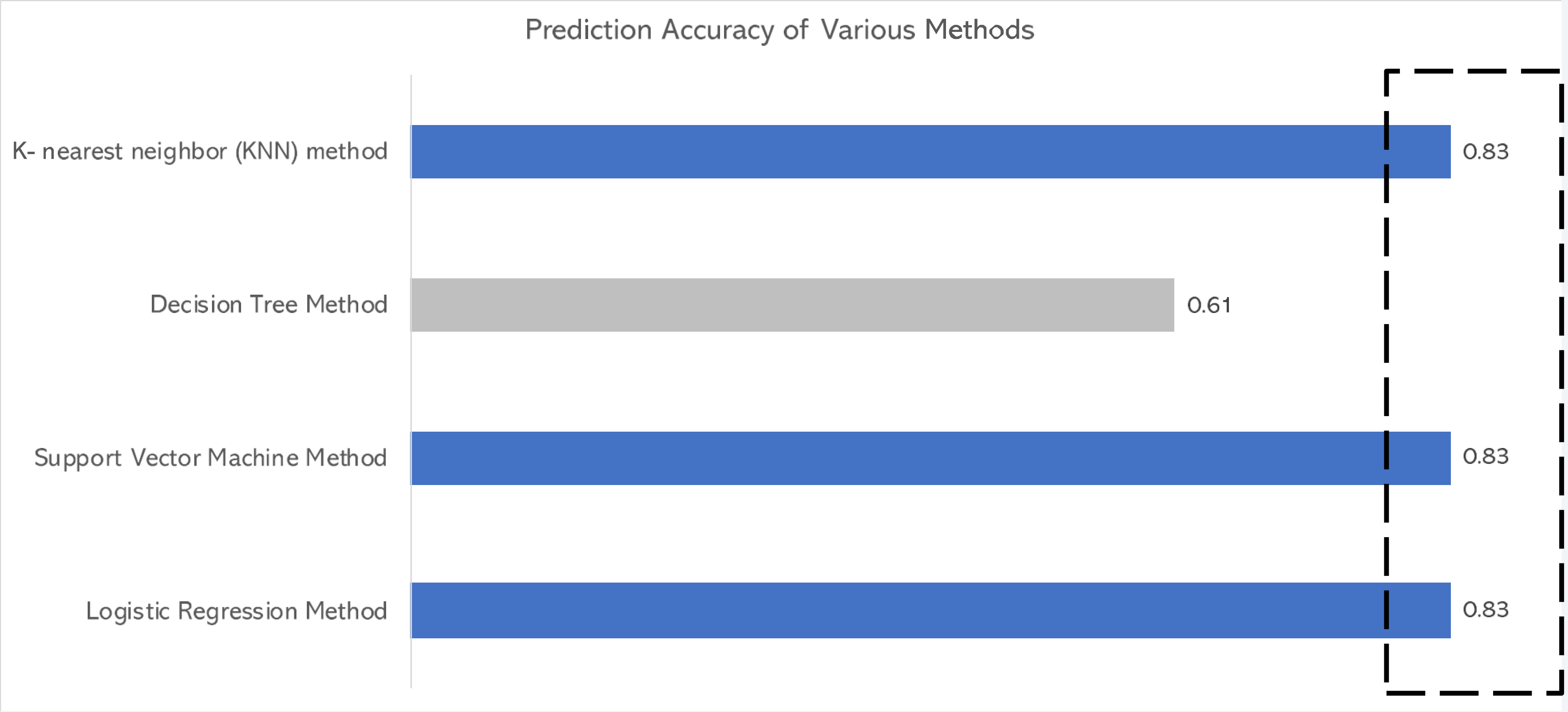


"Lower Payload launches (up to 6,000 kg) are more successful"

Section 6

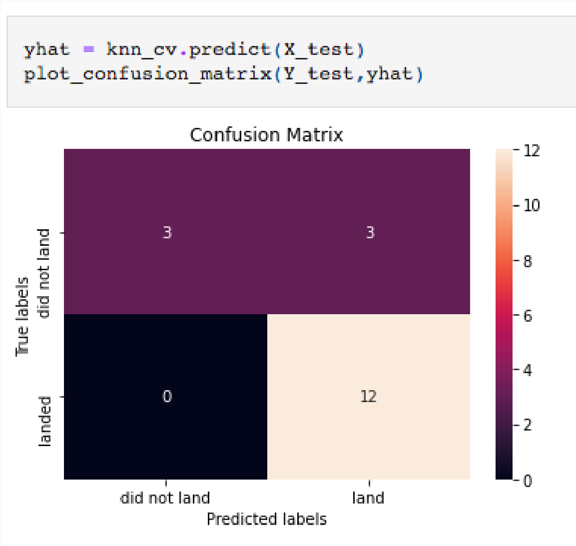
Predictive Analysis (Classification)

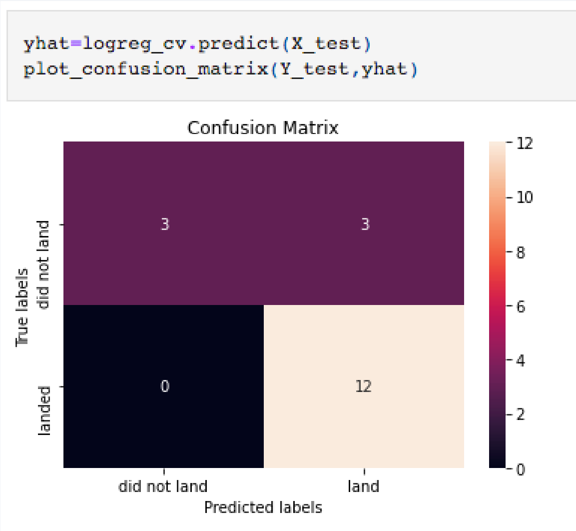
Classification Accuracy

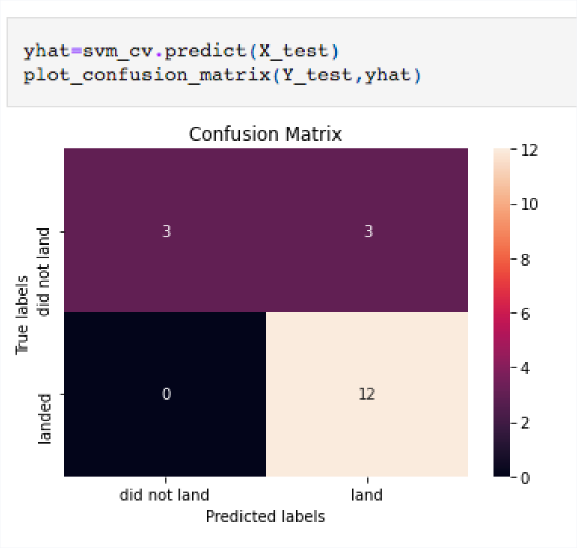


"KNN, Support Vector & Logistic Regression Methods have high accuracy"

Confusion Matrix







"The above confusion matrix shows that all 3 models – KNN, Logistic Regression & SVM have highest true positives and least false negatives"

Conclusions

- KNN, Logistic Regression and SVM are the best classifier models for this dataset
- The lower payload launches have higher success rate than heavier payloads
- Site KSC LC-39A has the most successful launches from all sites
- F9 Booster versions v1.0, v1.1, FT, B4, B5 have the highest launch success rates
- The SpaceX launches have been continuously getting better from year 2013 to 2020 based on data so they have the best chances for perfecting their launches in the future years

Appendix

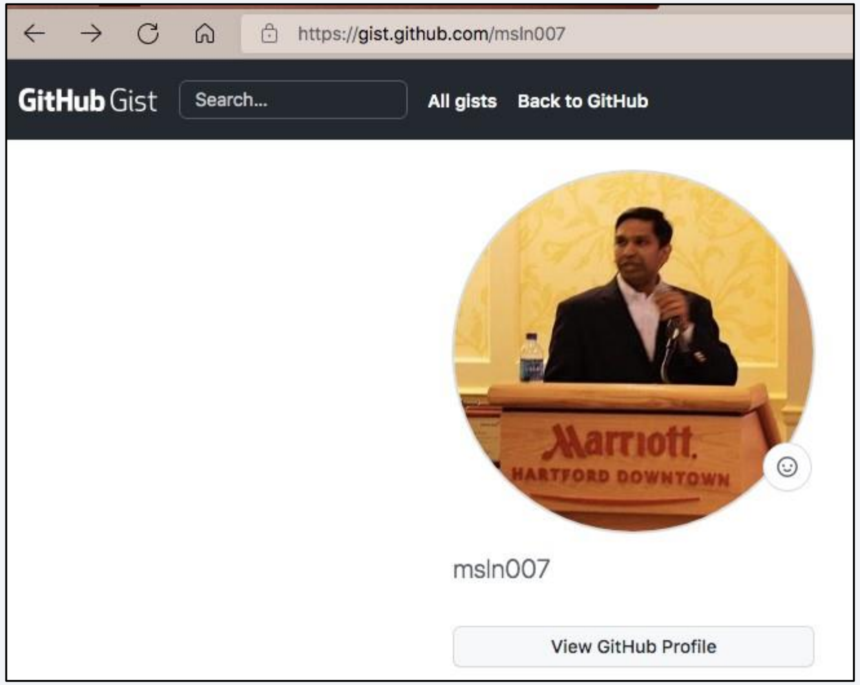
Notebooks

New notebook +

Name	Shared	Scheduled	Status	Language	Last editor	Last modified
Prediction with ML				Python 3.8	Lakshminarasimhan Madrassudarsan	Dec 26, 2021
The best classifier notebook				Python 3.8	Lakshminarasimhan Madrassudarsan	Dec 22, 2021
Project Loan Notebook				Python 3.8	Lakshminarasimhan Madrassudarsan	Dec 22, 2021
SpaceX-ML-Prediction				Python 3.8	Lakshminarasimhan Madrassudarsan	Dec 05, 2021
Interactive Visual Analytics with Folium lab				Python 3.8	Lakshminarasimhan Madrassudarsan	Nov 28, 2021
jupyter-labs-eda-dataviz (1)				Python 3.8	Lakshminarasimhan Madrassudarsan	Nov 28, 2021
EDA with SQL				Python 3.8	Lakshminarasimhan Madrassudarsan	Nov 07, 2021
SpacexDataWrangling				Python 3.8	Lakshminarasimhan Madrassudarsan	Nov 07, 2021
Web Scraping				Python 3.8	Lakshminarasimhan Madrassudarsan	Oct 24, 2021
Data Collection API				Python 3.8	Lakshminarasimhan Madrassudarsan	Oct 23, 2021
The Best classifier				Python 3.8	Lakshminarasimhan Madrassudarsan	Oct 18, 2021

IBM Cloud Pak Assignments

My GitHub, Gists



Credits

Primary Instructors: Joseph Santarcangelo, Yan Luo

Other Contributors & Staff
Project Lead: Rav Ahuja
Instructional Designer: Lakshmi Holla
Lab Authors: Joseph Santarcangelo, Yan Luo, Azim Hirjani, Lakshmi Holla
Technical Advisor: Yan Luo

Production Team
Publishing: Grace Barker, Rachael Jones
Project Coordinators: Kathleen Bergner
Narration: Bella West
Video Production: Simer Preet, Lauren Hall, Hunter Bay, Tanya Singh, Om Singh

Teaching Assistants and Forum Moderators
Malika Singla
Duvvana Mrutyunjaya Naidu

Copyrights and Trademarks

Thank you!