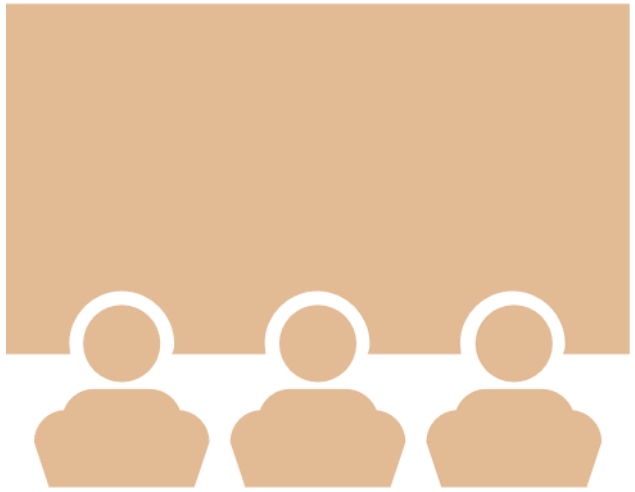
Data Science Capstone Project

Michael Sagbohan

https://github.com/Mikamike123/IBM-DataScience-Certificate

01/05/2023

Outline



Executive Summary

- The process involved in this project included gathering data from both the public SpaceX API and the SpaceX Wikipedia page. A new column titled 'class' was added to classify successful landings. The data was then explored using SQL, visualization techniques such as folium maps and dashboards. Relevant columns were selected to be used as features, and categorical variables were converted to binary using one hot encoding. The data was standardized, and GridSearchCV was utilized to determine the optimal parameters for various machine learning models. Finally, the accuracy scores of all the models were visualized.

- This project generated four machine learning models, namely Logistic Regression, Support Vector Machine, Decision Tree Classifier, and K Nearest Neighbors. These models had comparable performance, achieving an accuracy rate of approximately 83.33%. However, all models exhibited a tendency to over-predict successful landings. To improve the accuracy and precision of the models, additional data is required.

Introduction

SpaceX Falcon 9 Rocket – Source: SpaceX

Background

- Currently in the Commercial Space Age
- SpaceX offer best pricing ($62 million compared to. $165 million USD)
- Due to the capability to recover Stage 1 of rocket
- SpaceY aims to compete with SpaceX

Problem

- SpaceY has tasked us with training a machine learning model that can predict whether Stage 1 recovery will be successful

Methodology

- Data collection methodology:
  - Based on both data from SpaceX public API and SpaceX Wikipedia page
- Data wrangling
  - Classifying true landings as successful or unsuccessful
- Exploratory data analysis (EDA) using visualization and SQL
- Interactive visual analytics using Folium and Plotly Dash
- Predictive analysis using classification models
  - Tuned hyperparameters models using GridSearchCV

Methodology

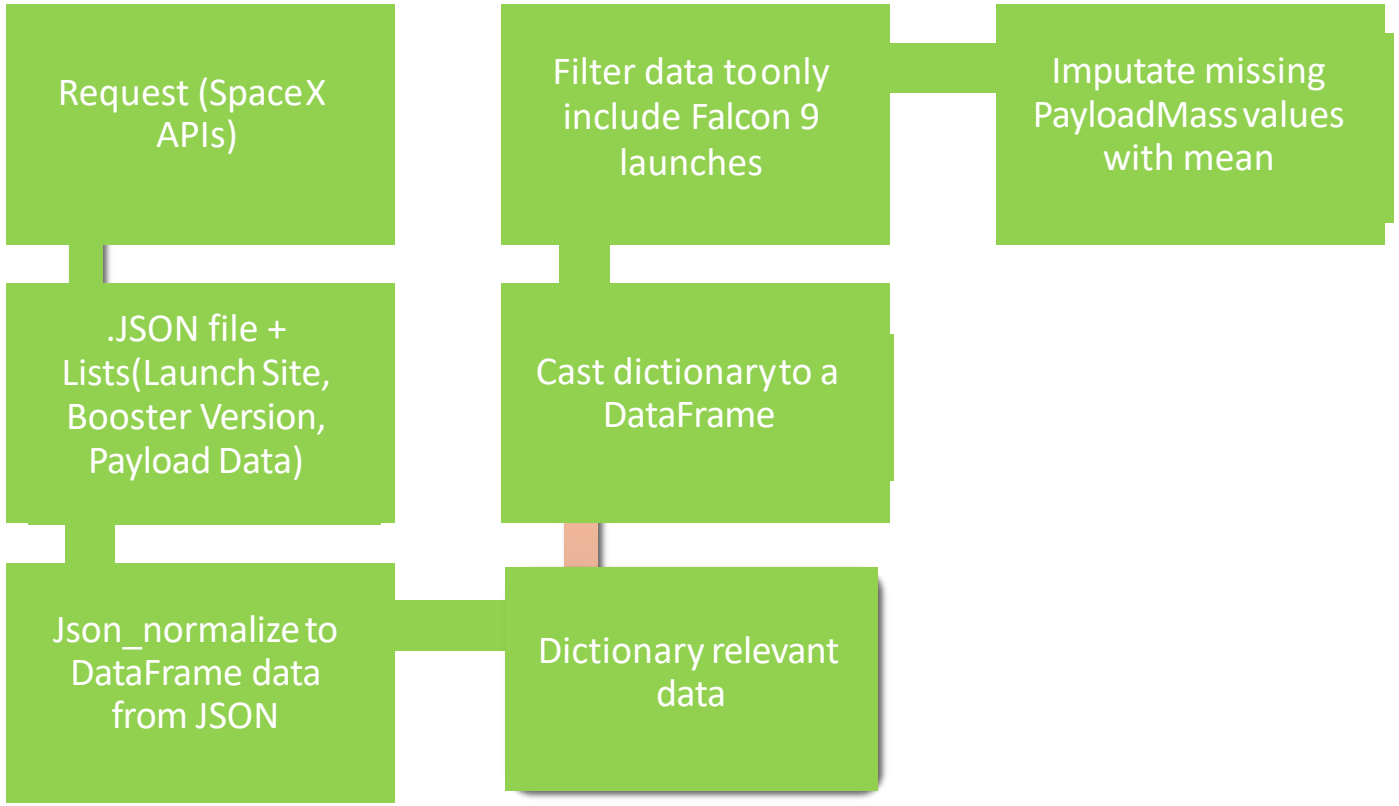
OVERVIEW OF DATACOLLECTION, WRANGLING, VISUALIZATION, DASHBOARD, AND MODEL METHODS

Data Collection Overview

The process of collecting the data involved a combination of requesting data from the SpaceX public API and web scraping a table from SpaceX's Wikipedia page.

The following slide will display the flowchart detailing the data collection from the API, while the one after will show the flowchart for data collection through web scraping.

Data Collection – SpaceX API



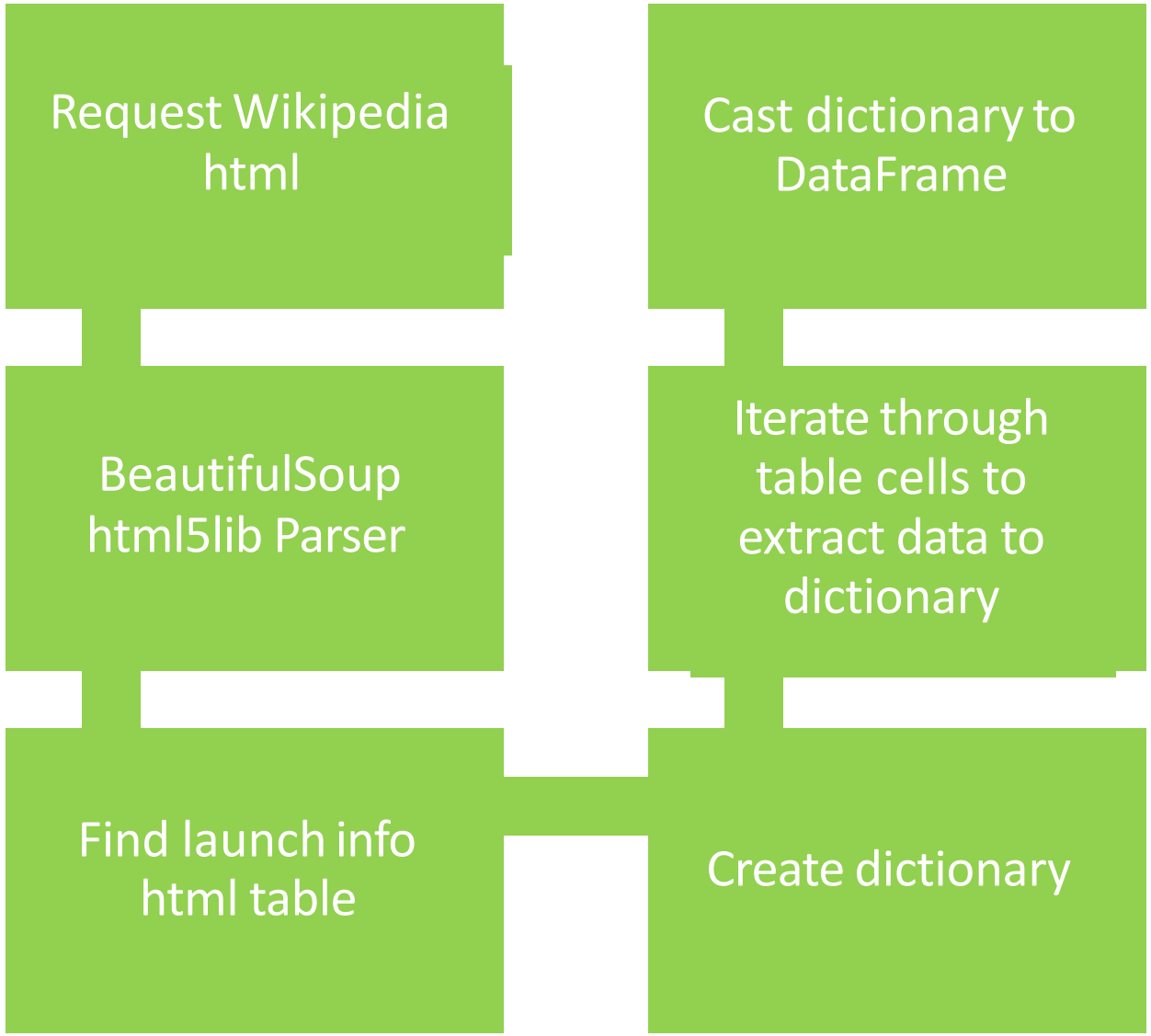
GitHub url:

https://github.com/Mikamike123/IBM-DataScience-Certificate/blob/d6acc6bd18ecfdcfccf97917570b0535aef5fdcb/jupyter-labs-spacex-data-collection-api.ipynb

Data Collection – Web Scraping

GitHub url:

https://github.com/Mikamike123/IBM-DataScience-Certificate/blob/d6acc6bd18ecfdcfccf97917570b0535aef5fdcb/jupyter-labs-webscraping.ipynb



Data Wrangling

- Calculate the number of launches on each site
- Calculate the number and occurrence of each orbit
- Calculate the number and occurence of mission outcome of the orbits
- New training label column ‘class’ that represents the outcome of each launch.

If the value is zero, the first stage did not land successfully; one means the first stage landed Successfully

- GitHub url: https://github.com/Mikamike123/IBM-DataScience-Certificate/blob/d6acc6bd18ecfdcfccf97917570b0535aef5fdcb/labs-jupyter-spacex-Data%20wrangling.ipynb

EDA with Data Visualization

Exploratory Data Analysis performed on variables Flight Number, Payload Mass, Launch Site, Orbit, Class and Year.

Plots Used:

Flight Number vs. Payload Mass, Flight Number vs. Launch Site, Payload Mass vs. Launch Site, Orbit vs. Success Rate, Flight Number vs. Orbit, Payload vs Orbit, and Success Yearly Trend

Scatter plots, line charts, and bar plots were utilized to examine the correlation between different variables to determine their potential use in training the machine learning model. These visualizations were used to investigate the relationships between the variables and to establish if any significant relationships existed.

GitHub url: https://github.com/Mikamike123/IBM-DataScience-Certificate/blob/d6acc6bd18ecfdcfccf97917570b0535aef5fdcb/jupyter-labs-eda-dataviz.ipynb

EDA with SQL

Loaded data set into IBM DB2 Database.

Queried using SQL Python integration.

Queries were made to get a better understanding of the dataset.

GitHub url:

https://github.com/Mikamike123/IBM-DataScience-Certificate/blob/d6acc6bd18ecfdcfccf97917570b0535aef5fdcb/jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an interactive map with Folium

Folium maps mark Launch Sites, successful and unsuccessful landings, and a proximity example to key locations: Railway, Highway, Coast, and City.

This allows us to understand why launch sites may be located where they are. Also visualizes successful landings relative to location.

GitHub url:

https://github.com/Mikamike123/IBM-DataScience-Certificate/blob/d6acc6bd18ecfdcfccf97917570b0535aef5fdcb/lab_jupyter_launch_site_location.ipynb

Build a Dashboard with Plotly Dash

Dashboard includes a pie chart and a scatter plot.

Pie chart can be selected to show distribution of successful landings across all launch sites and can be selected to show individual launch site success rates.

Scatter plot takes two inputs: All sites or individual site and payload mass on a slider between 0 and 10000 kg.

The pie chart is used to visualize launch site success rate.

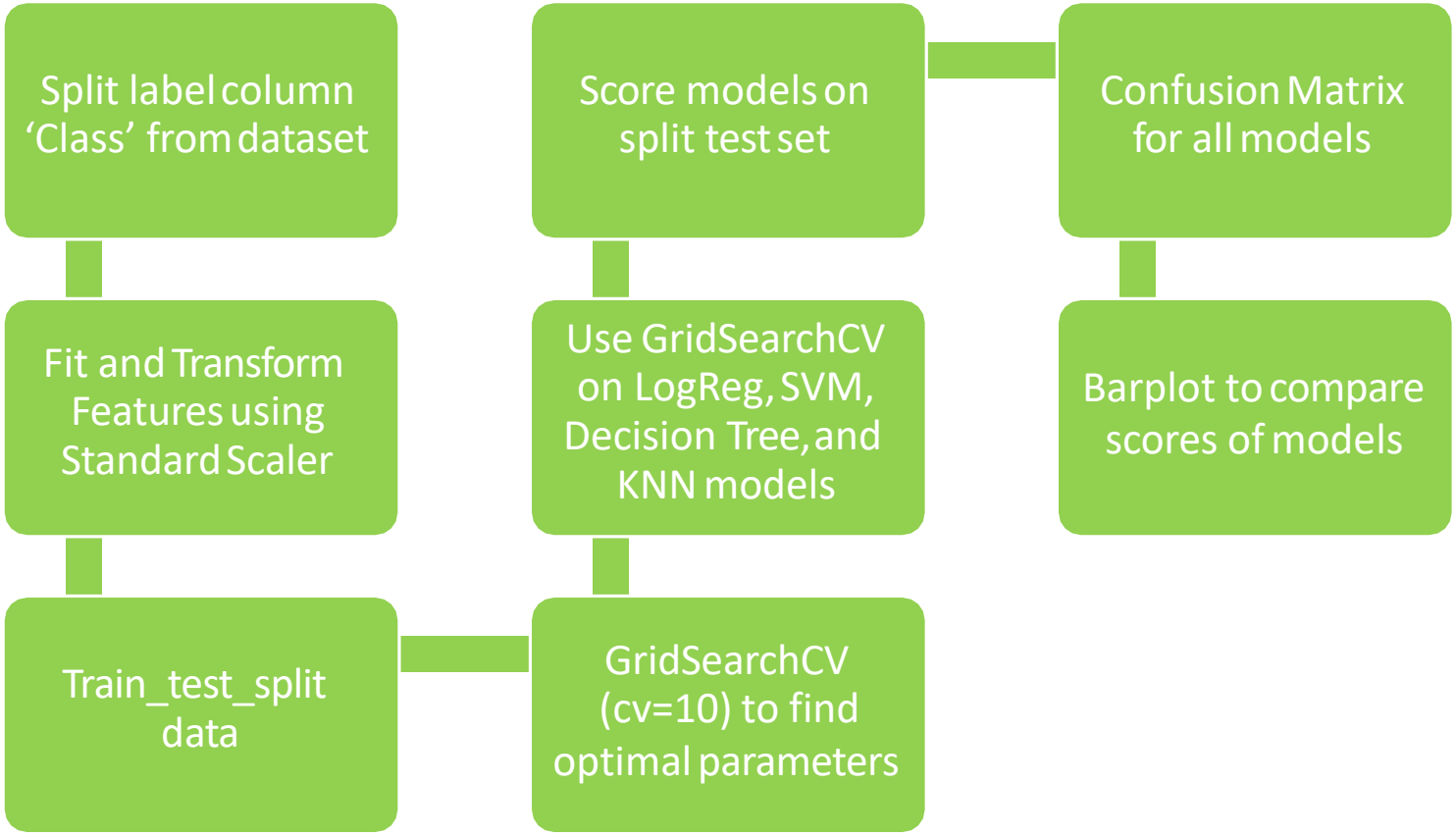
The scatter plot can help us see how success varies across launch sites, payload mass, and booster version category.

GitHub url:

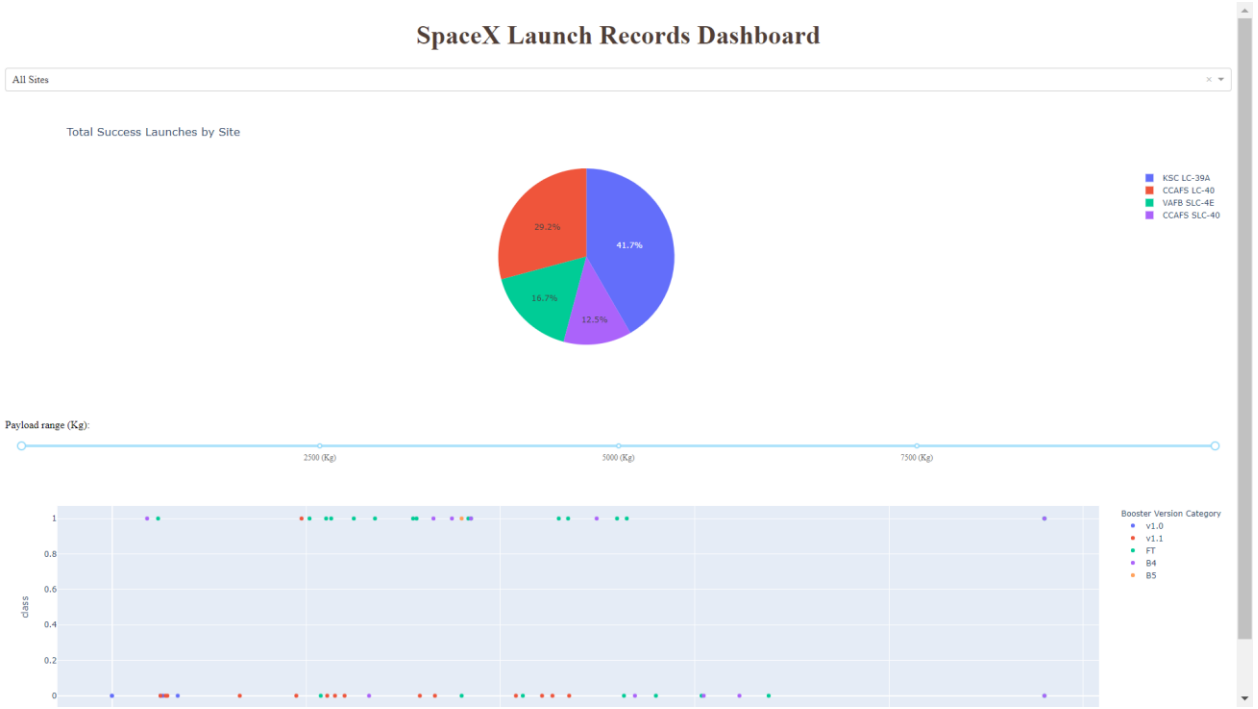
https://github.com/Mikamike123/IBM-DataScience-Certificate/blob/d6acc6bd18ecfdcfccf97917570b0535aef5fdcb/spaceX_dash_app.py

Predictive analysis (Classification)

GitHub url: https://github.com/Mikamike123/IBM-DataScience-Certificate/blob/c4ac2716bddeb97e13f607a55b5c473584d336fa/SpaceX_Machine_Learning_Prediction_Part_5.jupyterlite.ipynb



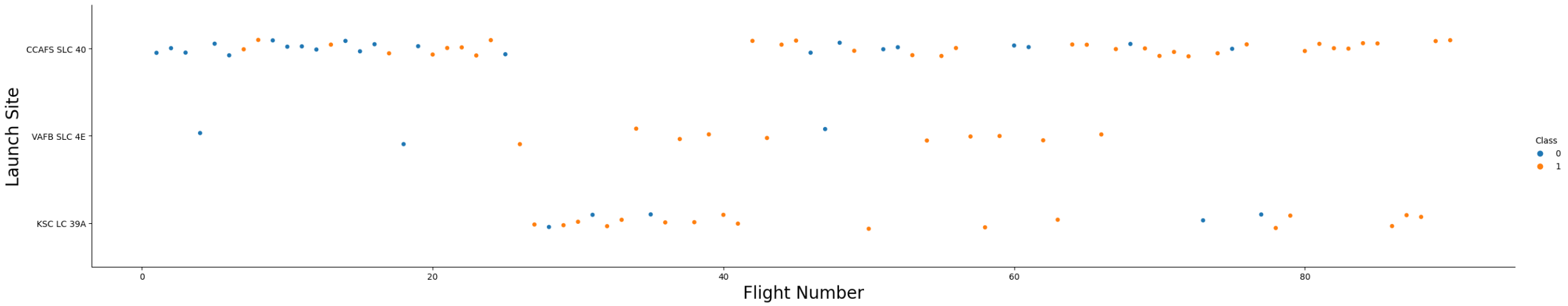
Results



Here above is a preview of the Plotly dashboard.

EDA with Visualization

Flight Number vs. LaunchSite



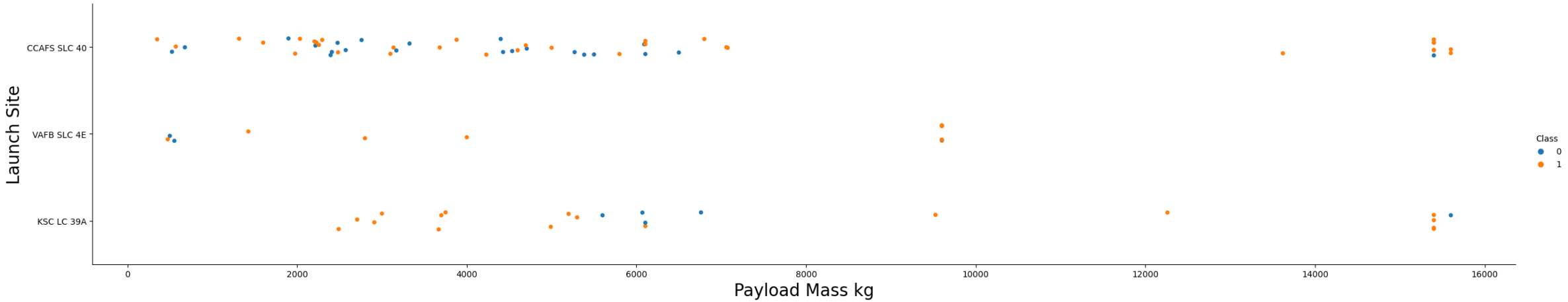
Orange indicates successful launch; Blue indicates unsuccessful launch.

Graphic suggests an increase in success rate over time (indicated in Flight Number).

Likely a big breakthrough around flight 20 which significantly increased success rate.

CCAFS appears to be the main launch site as it has the most volume.

Payload vs. Launch Site

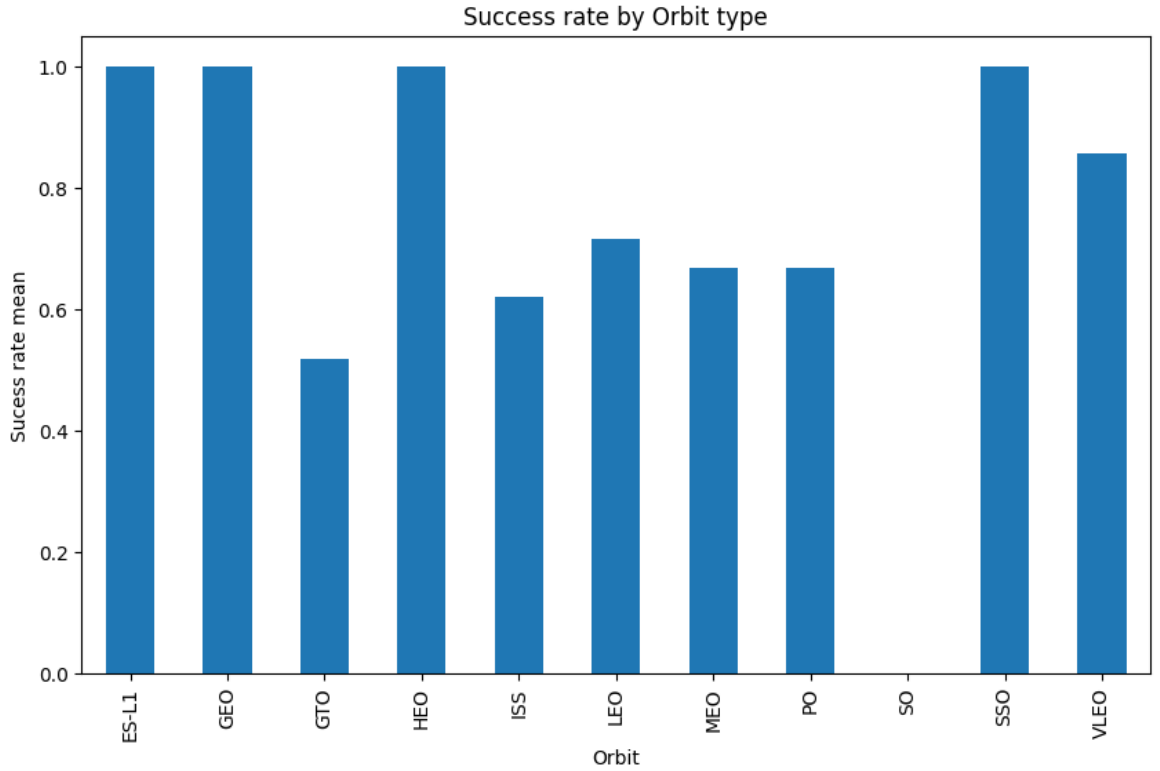


Orange indicates successful launch; Blue indicates unsuccessful launch.

Payload mass appears to fall mostly between 0-6000 kg.

Different launch sites also seem to use different payload mass.

Success rate vs. Orbit type



Success Rate Scale with
0 as 0%
1 as 100%

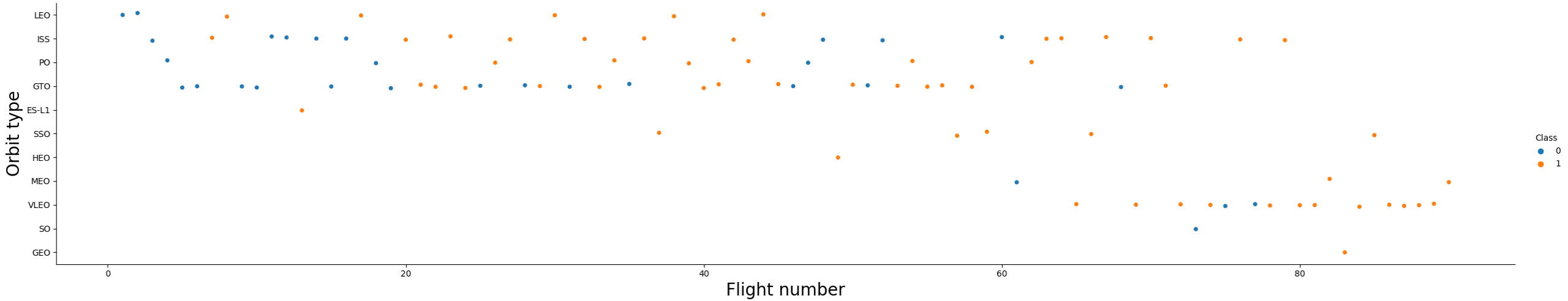
ES-L1, GEO, HEO, SSO have 100% success rate

VLEO has decent success rate

SO has 0% success rate

GTO has around 50% success rate

Flight Number vs. Orbittype



Orange indicates successful launch; Blue indicates unsuccessful launch.

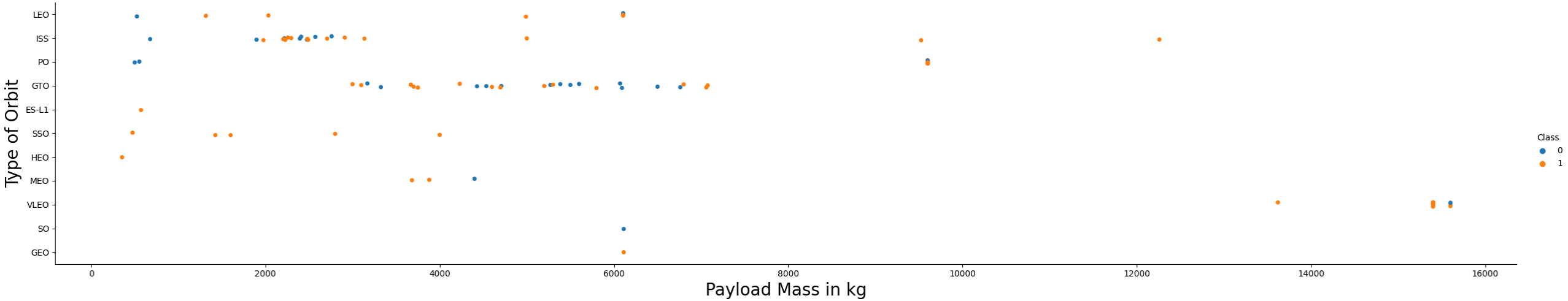
Launch Orbit type changed over Flight Number.

Launch Outcome seems to correlate with this preference.

SpaceX started with LEO orbits with moderate success and returned to VLEO in latest launches

SpaceX appears to perform better in lower orbits or Sun-synchronous orbits

Payload vs. Orbit type



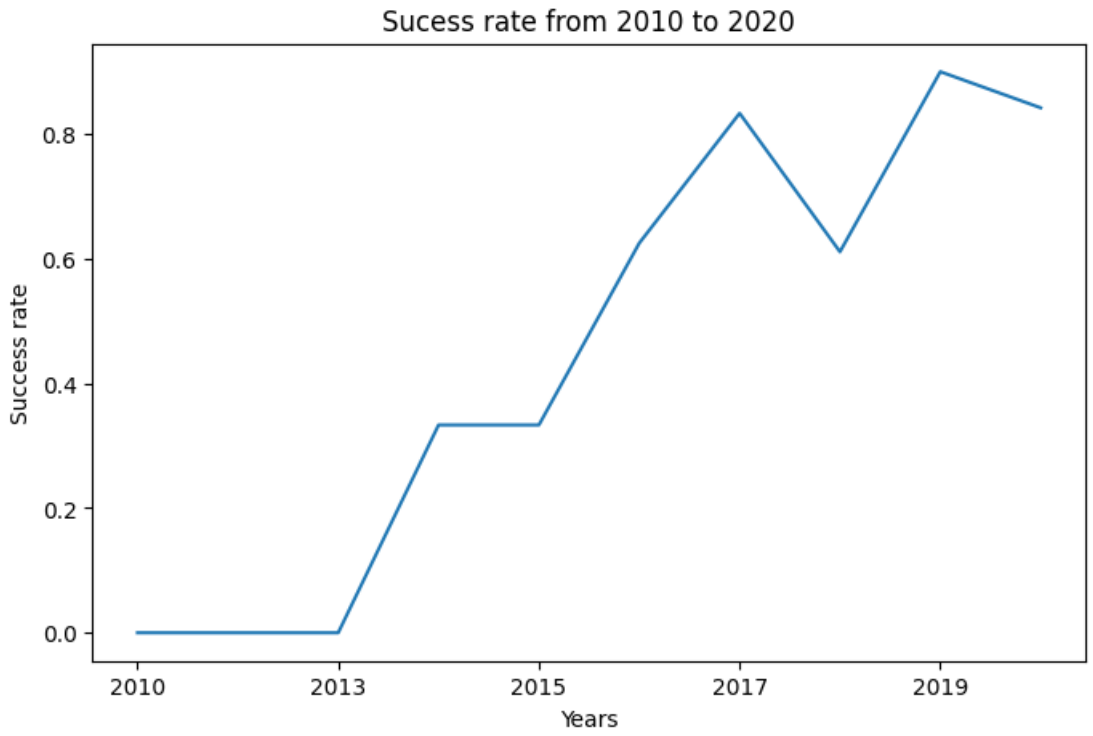
Orange indicates successful launch; Blue indicates unsuccessful launch.

Payload mass seems to correlate with orbit

LEO and SSO seem to have relatively low payload mass

The other most successful orbit VLEO only has payload mass values in the higher end of the range

Launch Success Yearly Trend



We can observe that the sucess rate since 2013 kept increasing till 2020

EDA with SQL

All Launch Site Names

```
1  %sql select distinct(Launch_Site) from SPACEXTBL;
```

✓

```
 * sqlite:///my_data1.db
Done.
```

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

Query unique launch site names from database.

CCAFS SLC-40 and CCAFS LC-40 likely represent the same

Likely only 3 unique launch_site values:
CCAFS SLC-40, KSC LC-39A, VAFB SLC-4E

Launch Site Names Beginning with `CCA`

```
[15]  1  %sql select * from SPACEXTBL where Launch_Site like 'CCA%' limit 5 ;
      ✓ 0.5s
...   * sqlite:///my_data1.db
Done.
```

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
04-06-2010	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
08-12-2010	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)
22-05-2012	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
08-10-2012	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
01-03-2013	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

First five entries in database with Launch Site name beginning with 'CCA'.

Total Payload Mass from NASA

Display the total payload mass carried by boosters launched by NASA (CRS)

```
1  %sql select sum(PAYLOAD_MASS__KG_) from SPACEXTBL where CUSTOMER = 'NASA (CRS)';
```

[27] ✓ 0.1s

```
 * sqlite:///my_data1.db
Done.
```

sum(PAYLOAD_MASS__KG_)
45596

This query sums the total payload mass in kg where NASA was the customer.

These payloads were sent to the International Space Station (ISS).

Average Payload Mass by F9v1.1

Display average payload mass carried by booster version F9 v1.1

```
1  %sql select AVG(PAYLOAD_MASS__KG_) from SPACEXTBL where Booster_Version = 'F9 v1.1';
```

[17] ✓ 0.3s

```
 * sqlite:///my_data1.db
Done.
```

AVG(PAYLOAD_MASS__KG_)
2928.4

This query calculates the average payload mass or launches which used booster version F9 v1.1

First Successful Ground Pad Landing Date

```
1  %sql select min(Date) from SPACEXTBL where Landing_Outcome = 'Success (ground pad)';
```

[28] ✓ 0.1s

```
* sqlite:///my_data1.db
Done.
```

min(Date)
01-05-2017

This query returns the first successful ground pad landing date.

First ground pad landing wasn’t before 2015.

Boosters that Carried Maximum Payload Mass

```
1  %sql select distinct(Booster_Version) from SPACEXTBL where PAYLOAD_MASS__KG_ in (select max(PAYLOAD_MASS__KG_) from SPACEXTBL);
```

[22] ✓ 0.3s

```
 * sqlite:///my_data1.db
Done.
```

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

This query returns the booster versions that carried the highest payload mass of 15600 kg.

These booster versions are very similar and all are of the F9 B5 B10xx.x variety.

.

Ranking Counts between 2010-06-04 and 2017-03-20

```
%%sql SELECT Landing_Outcome, COUNT(*) AS count_launches FROM SPACEXTBL
WHERE Date BETWEEN '04-06-2010' AND '20-03-2017'
GROUP BY Landing_Outcome ORDER BY count_launches DESC;

```

[24] ✓ 0.2s

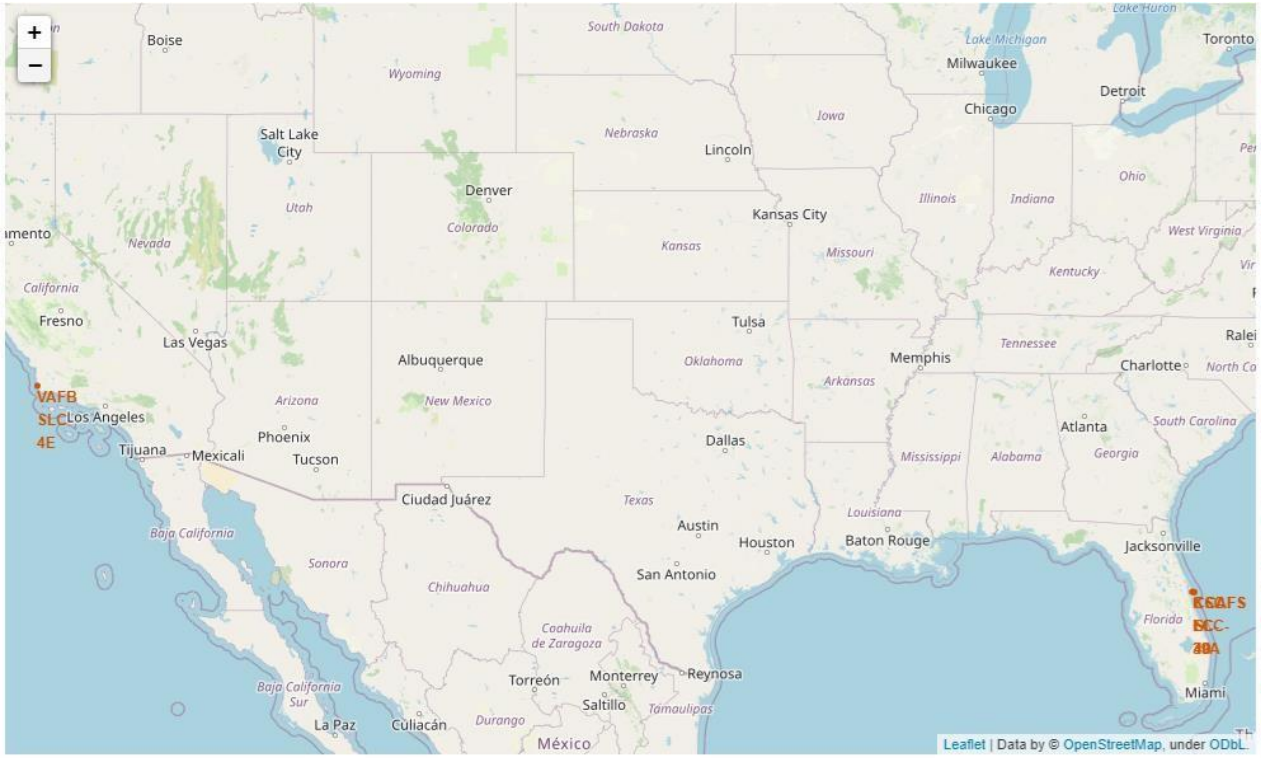
```
 * sqlite:///my_data1.db
Done.
```

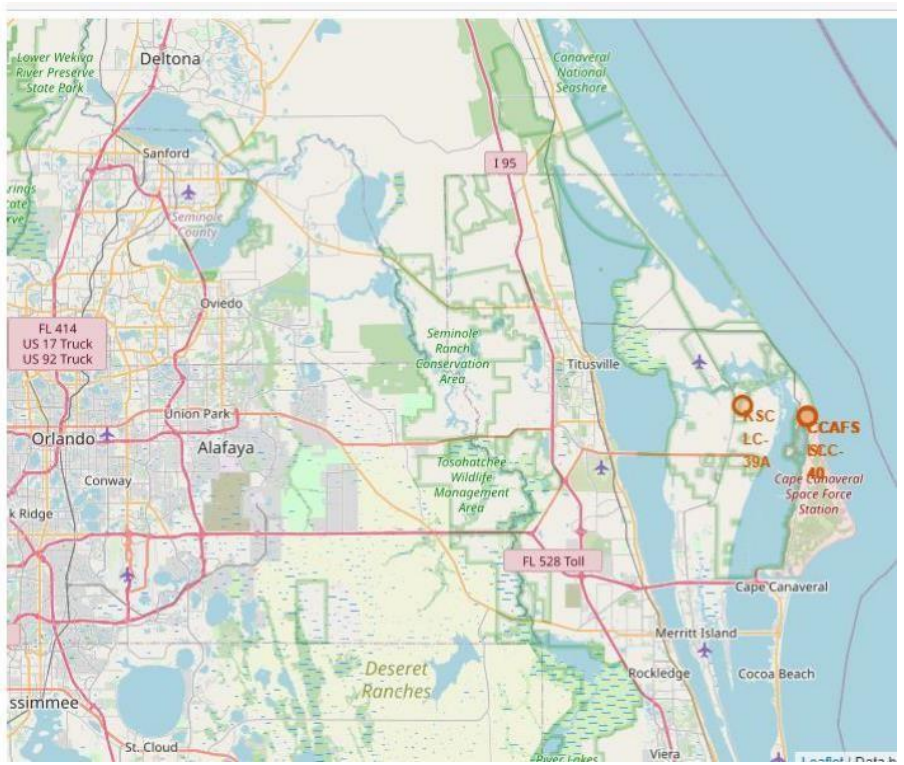
Landing_Outcome	count_launches
Success	20
No attempt	10
Success (drone ship)	8
Success (ground pad)	6
Failure (drone ship)	4
Failure	3
Controlled (ocean)	3
Failure (parachute)	2
No attempt	1

This query returns a list of landings between 2010-06-04 and 2017-03-20.

Interactive Map with Folium

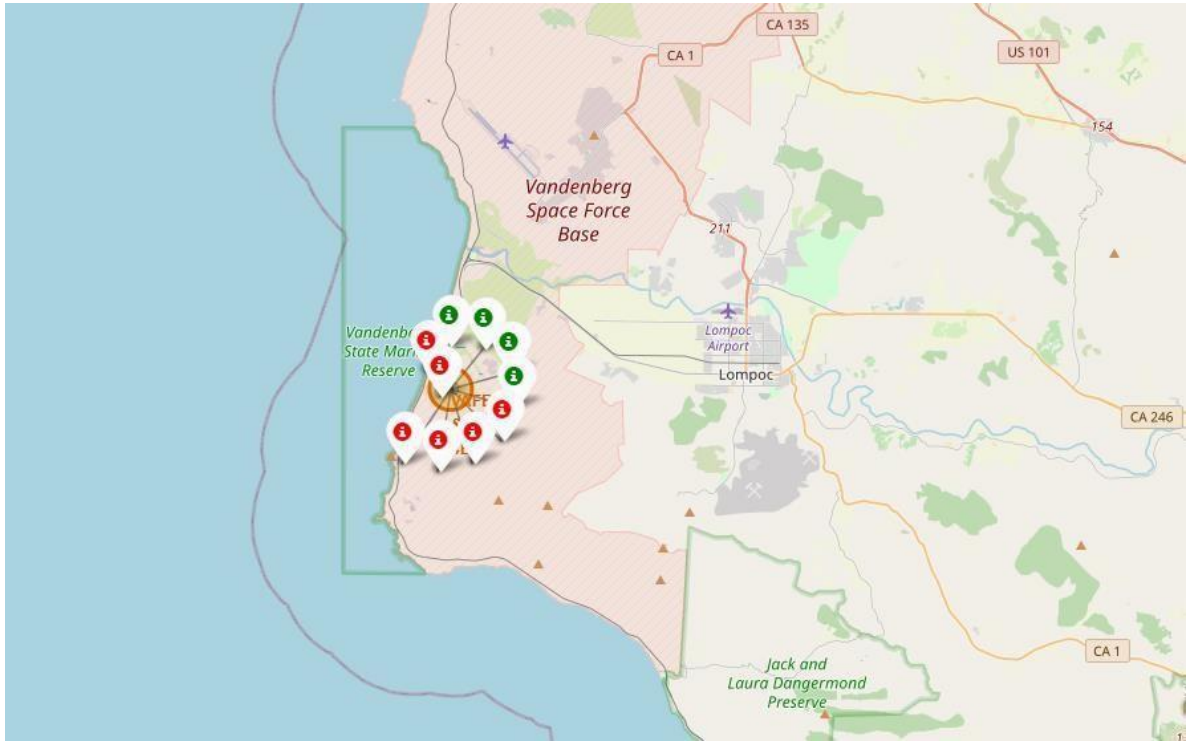
Launch Site Locations





The left map shows all launch sites relative US map. The right map shows the two Florida launch sites since they are very close to each other. All launch sites are near the ocean.

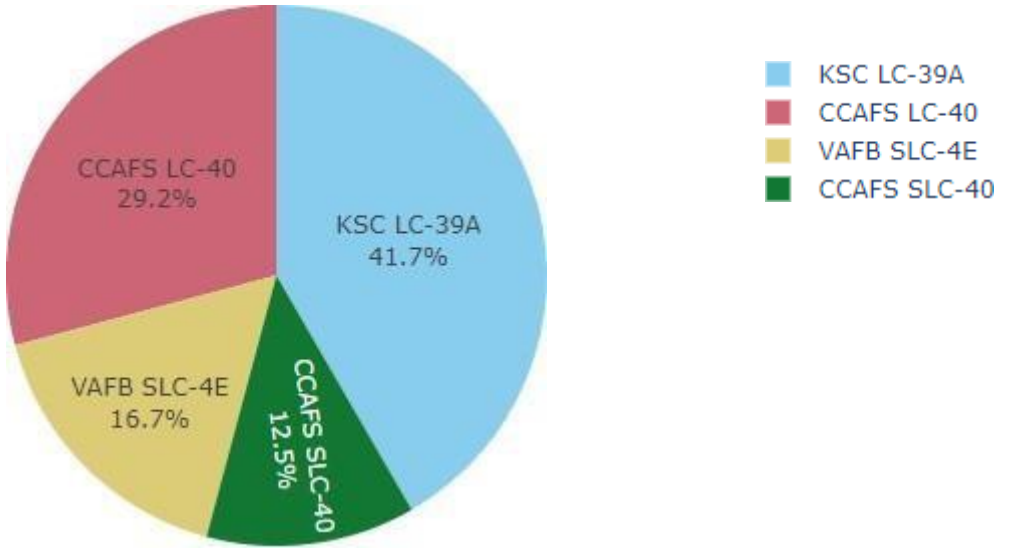
Color-Coded Launch Markers



Clusters on Folium map can be clicked on to display each successful landing (green icon) and failed landing (red icon). In this example VAFB SLC-4E shows 4 successful landings and 6 failed landings.

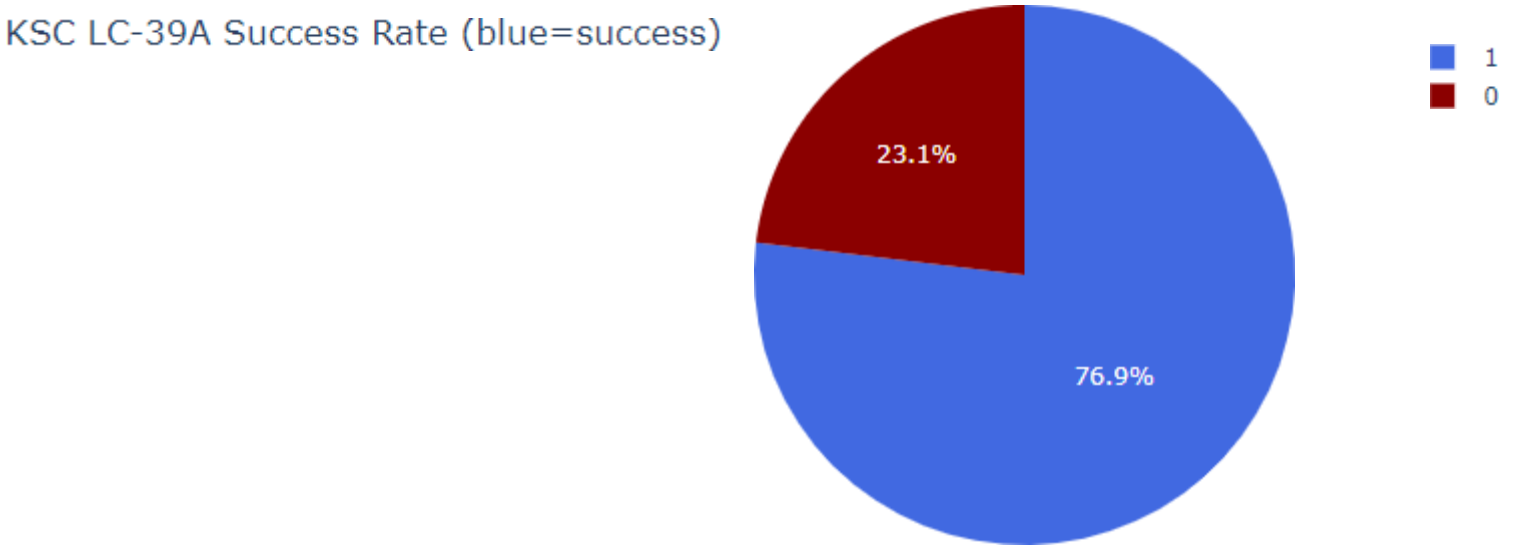
Dashboard with Plotly Dash

Successful Launches Across Launch Sites



This is the distribution of successful landings across all launch sites. CCAFS LC-40 is the old name of CCAFS SLC-40 so CCAFS and KSC have the same amount of successful landings, but a majority of the successful landings where performed before the name change. VAFB has the smallest share of successful landings. This may be due to smaller sample and increase in difficulty of launching in the west coast.

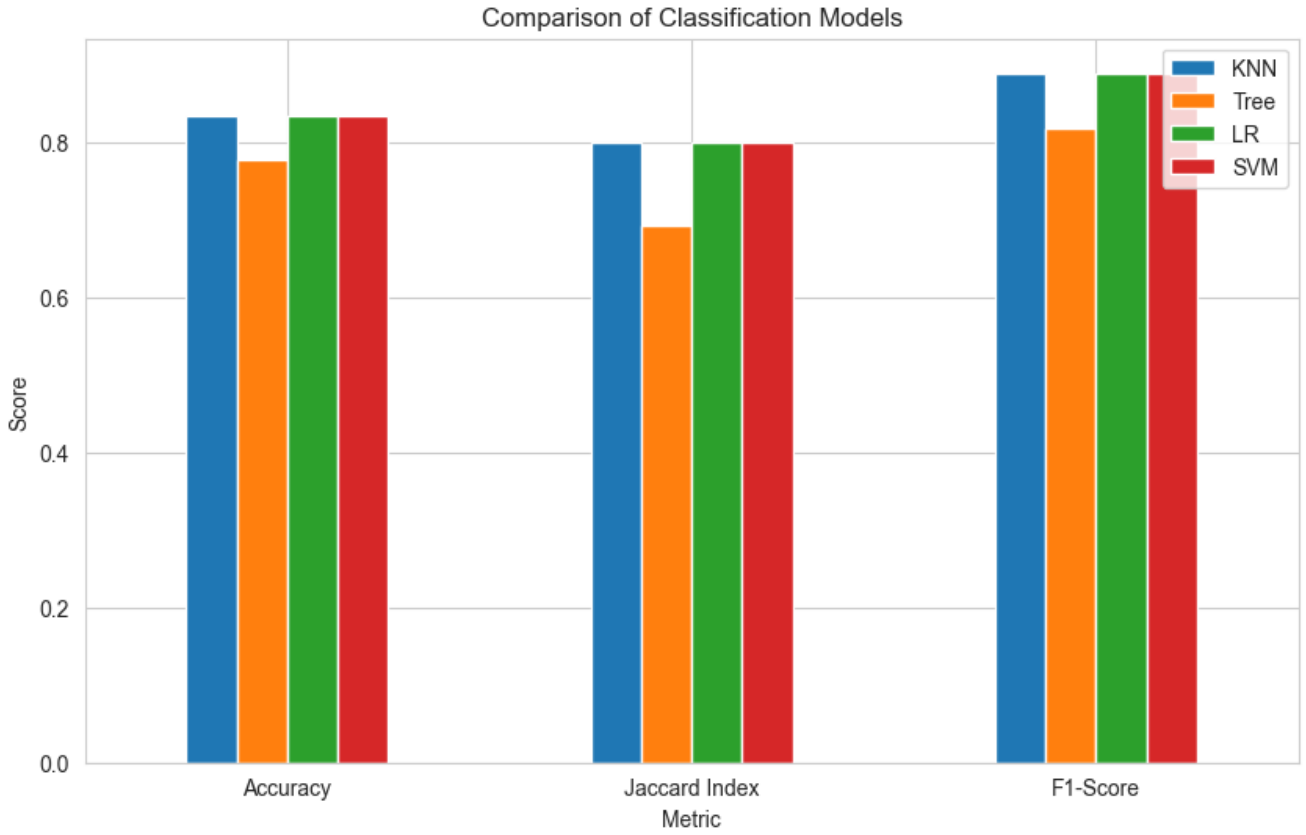
Highest Success Rate Launch Site



KSC LC-39A has the highest success rate with 10 successful landings and 3 failed landings.

Predictive classification

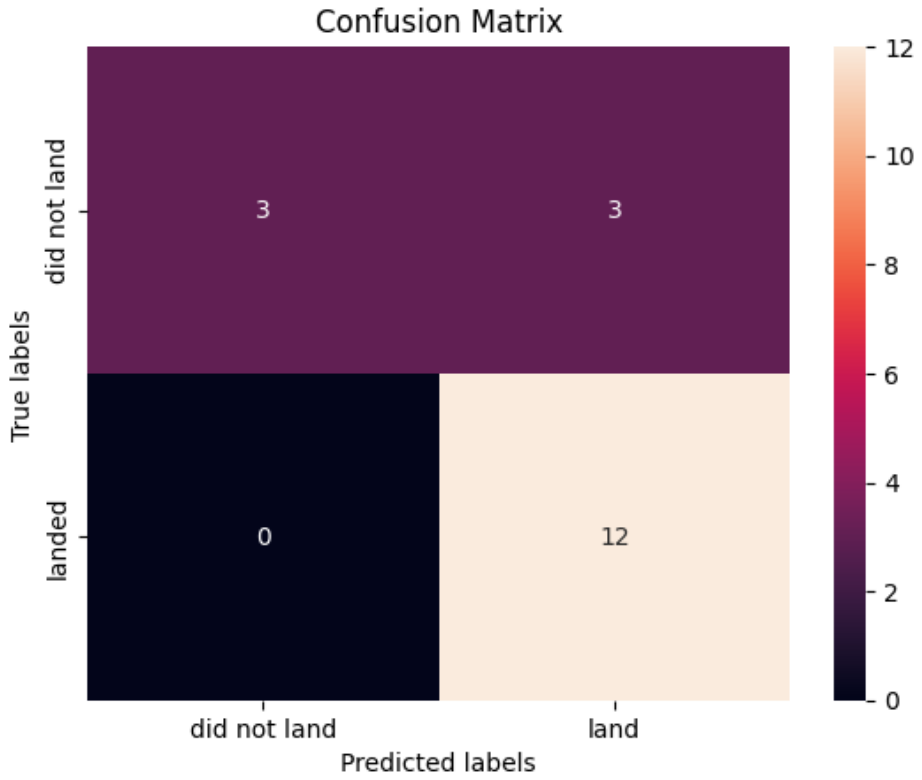
Models Accuracy



When we compare the accuracy, Jaccard index metric, F1-score of the different algorithms: Logistic Regression, Support Vector Machine methods, they all preformed the same on the test set except the Decision Tree model which performed worse on the data set.

The sample size for the test is only 18, which is relatively small. Therefore, it is essential to gather more data to improve the different accuracy metrics of the model and determine the most suitable one.

Confusion Matrix



The confusion matrix remained identical for all models as they showed similar performance on the test set. According to the results, the models correctly predicted 12 successful landings (True Positives), and 3 unsuccessful landings (True Negatives), whereas they falsely predicted 3 successful landings when the actual outcome was unsuccessful (False Positives).

These results indicate that the models have a tendency to over-predict successful landings.

CONCLUSION

- Our task was to develop a machine learning model for Space Y to compete against SpaceX.
- The objective of the model was to predict when Stage 1 would successfully land
- We collected data from a public SpaceX API and web-scraped the SpaceX Wikipedia page.
- Data labels were created, and the data was stored in a DB2 SQL database.
- A dashboard was developed for visualizing the data.
- We created machine learning models with an accuracy rate of 83%.
- Allon Mask of SpaceY can use this model to predict, with relatively high accuracy, whether a launch will have a successful Stage 1 landing before the launch, which can help determine whether the launch should proceed or not.
- Collecting more data could help in determining the best machine learning model and improving accuracy.

APPENDIX

GitHub repository url:

https://github.com/Mikamike123/IBM-DataScience-Certificate

Special Thanks to All Instructors:

https://www.coursera.org/professional-certificates/ibm-data-science?#instructors