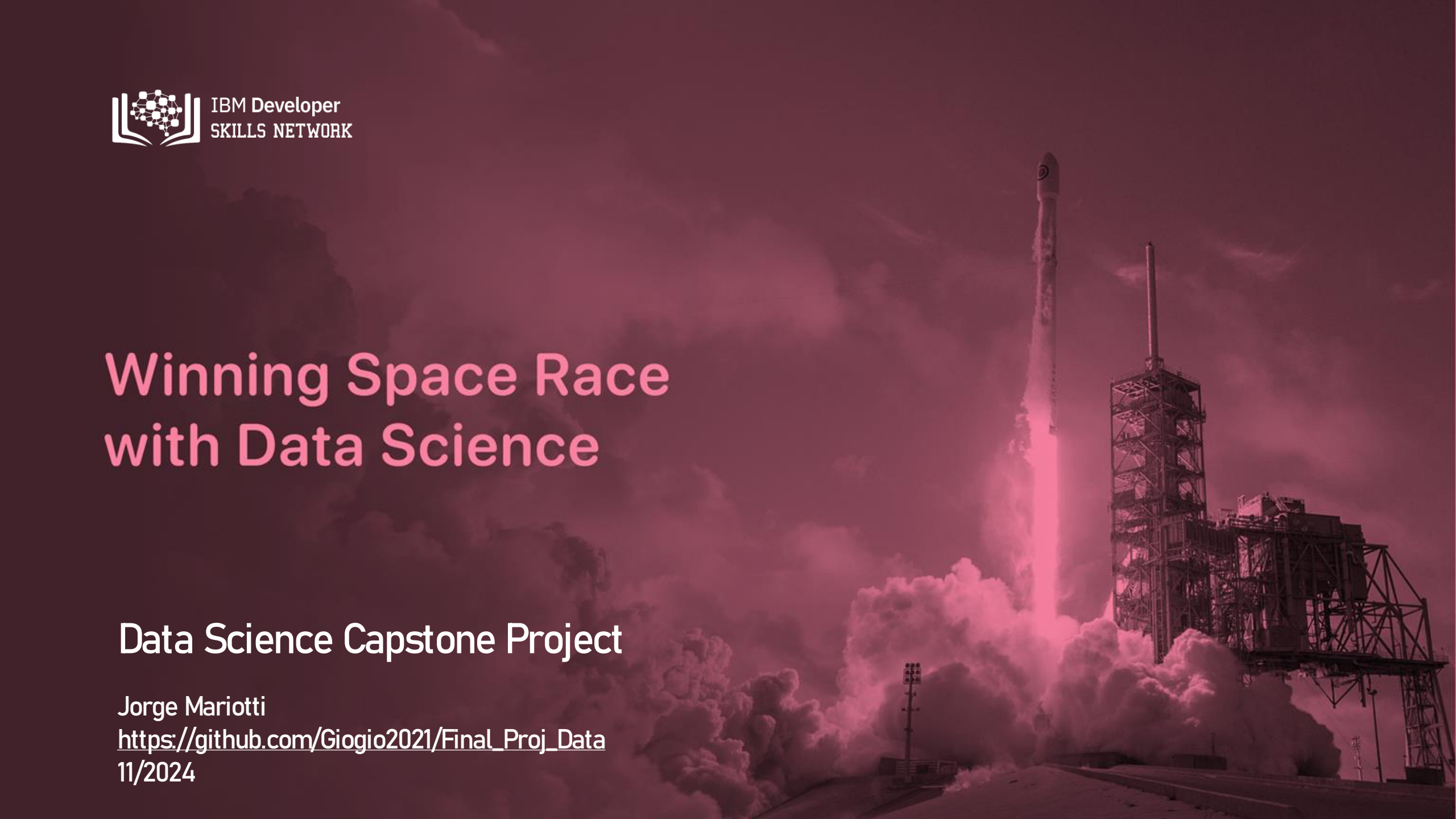


Winning Space Race with Data Science

Data Science Capstone Project

Jorge Mariotti

https://github.com/Giogio2021/Final_Proj_Data

11/2024

Outline

Executive Summary

- Summary of methodologies

Data Dive:
We delved into the rich dataset from SpaceX's public API and Wikipedia. By labeling successful landings and exploring the data through SQL queries, visualizations, and interactive maps, we gained valuable insights.

Feature Engineering and Model Selection:
We carefully selected relevant features and transformed categorical variables for seamless model training. After fine-tuning hyperparameters with GridSearchCV, we trained four machine learning models: Logistic Regression, Support Vector Machine, Decision Tree Classifier, and K-Nearest Neighbors.

- Summary of all results

Model Performance and Future Directions:
While all models achieved a respectable accuracy of around 83.33%, they tended to overpredict successful landings. To enhance model performance and make more accurate predictions, we'll need to gather more data to capture a wider range of scenarios.

Introduction

- Project background and context

Commercial Space Age is Here
Space X has best pricing ($62 million vs. $165 million USD)
Largely due to ability to recover part of rocket (Stage 1)
Space Y wants to compete with Space X

- Problems you want to find answers

Space Y tasks us to train a machine learning model to predict successful Stage 1 recovery

Methodology

- Data collection methodology
  - Combined data from SpaceX public API and SpaceX Wikipedia page
- Perform data wrangling
  - Classifying true landings as successful and unsuccessful otherwise
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - How to build, tune, evaluate classification models

Data Collection Overview

Data collection process involved a combination of API requests from Space X public API and web scraping data from a table in Space X's Wikipedia entry.

The next slide will show the flowchart of data collection from API and the one after will show the flowchart of data collection from webscraping.

Space X API Data Columns:

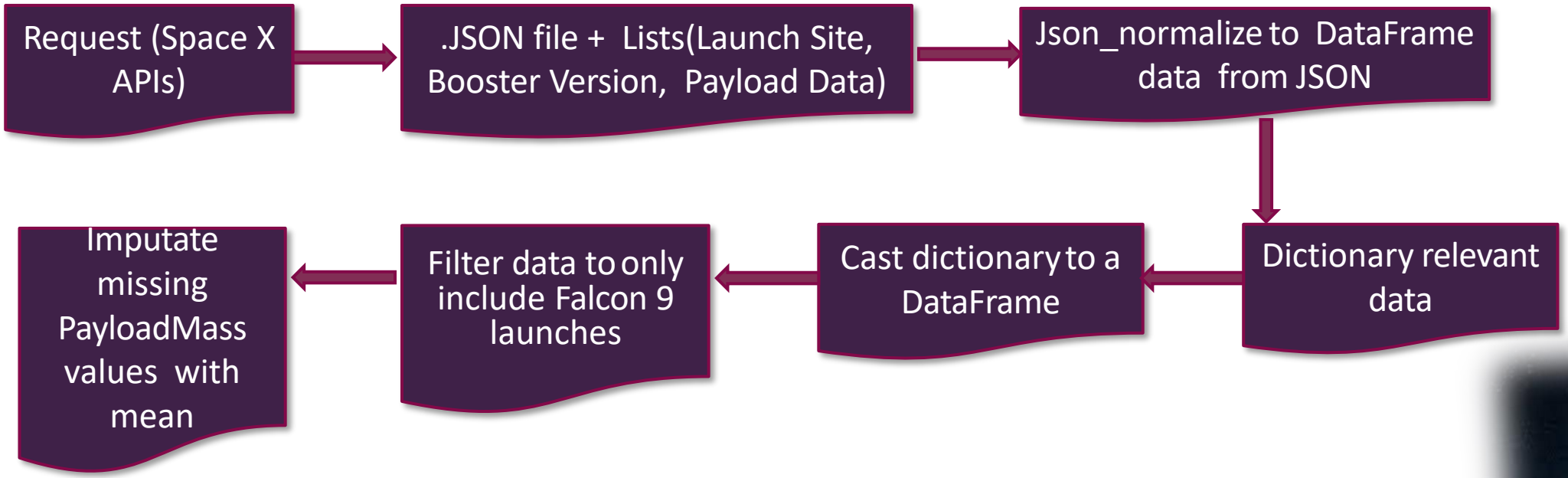
FlightNumber, Date, BoosterVersion, PayloadMass, Orbit, LaunchSite, Outcome, Flights, GridFins,

Reused, Legs, LandingPad, Block, ReusedCount, Serial, Longitude, Latitude

Wikipedia Webscrape Data Columns:

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

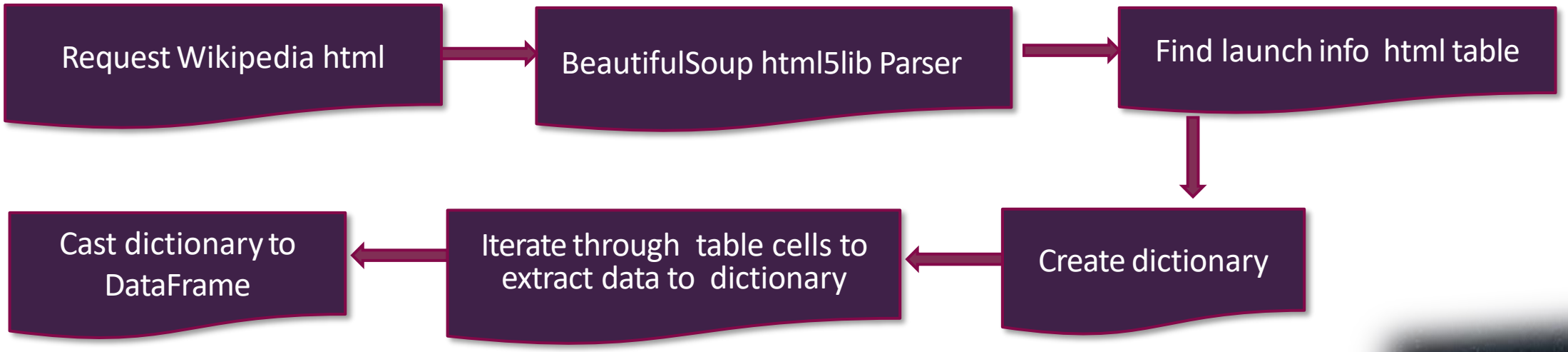
Data Collection – SpaceX API



GitHub URL

https://github.com/Giogio2021/Final_Proj_Data/blob/main/Mod_1/jupyter-labs-spacex-data-collection-api_JM.ipynb

Data Collection – Webscraping



GitHub URL

https://github.com/Giogio2021/Final_Proj_Data/blob/main/Mod_1/jupyter-labs-webscraping_JM.ipynb

Data Wrangling

Create a training label with landing outcomes where successful = 1 & failure = 0.

Outcome column has two components: 'Mission Outcome' 'Landing Location'

New training label column 'class' with a value of 1 if 'Mission Outcome' is True and 0 otherwise.

Value Mapping:

True ASDS, True RTLS, & True Ocean – set to -> 1

None None, False ASDS, None ASDS, False Ocean, False RTLS – set to -> 0

GitHub URL

https://github.com/Giogio2021/Final_Proj_Data/blob/main/Mod_1/labs-jupyter-spacex-Data%20wrangling_JM.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 used to compare relationships between variables to decide if a relationship exists so that they could be used in training the machine learning model

GitHub URL

https://github.com/Giogio2021/Final_Proj_Data/blob/main/Mod_2/edadataviz_JM.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.

Queried information about launch site names, mission outcomes, various pay load sizes of customers and booster versions, and landing outcomes

GitHub URL

https://github.com/Giogio2021/Final_Proj_Data/blob/main/Mod_2/jupyter-labs-eda-sql-coursera_sqllite_JM.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/Giogio2021/Final_Proj_Data/blob/main/Mod_3/lab_jupyter_launch_site_location_JM.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.

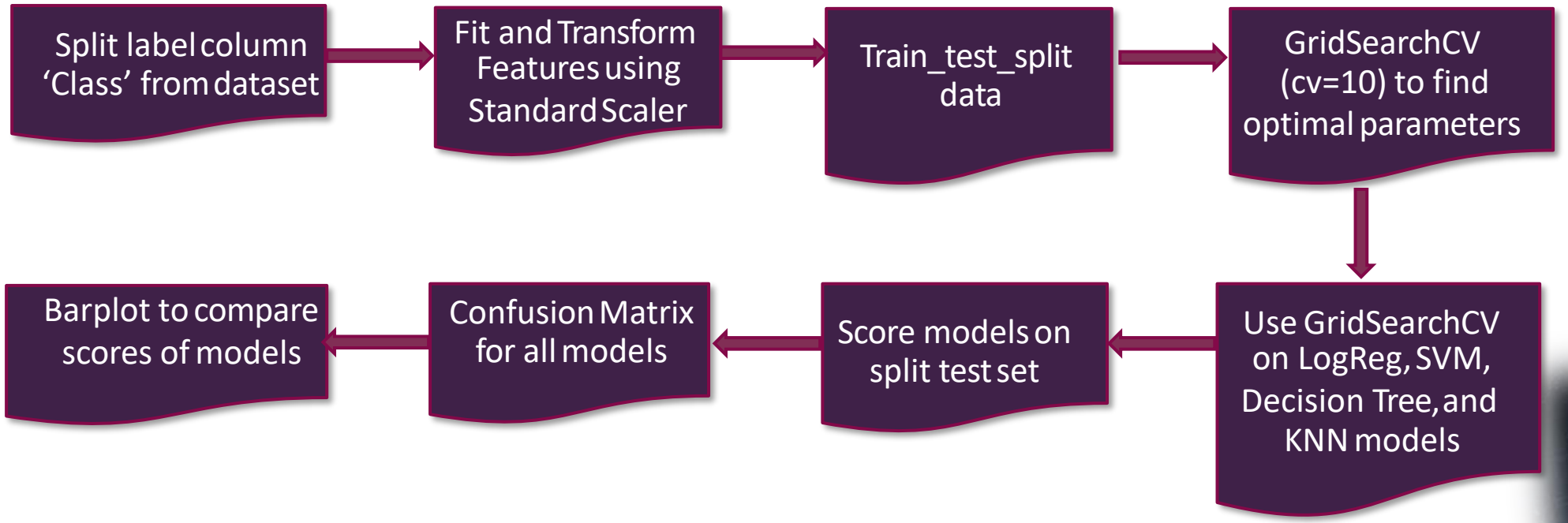
The pie chart is used to visualize launch site success rate.

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

GitHub URL

https://github.com/Giogio2021/Final_Proj_Data/blob/main/Mod_3/spacex_dash_app_JM.py

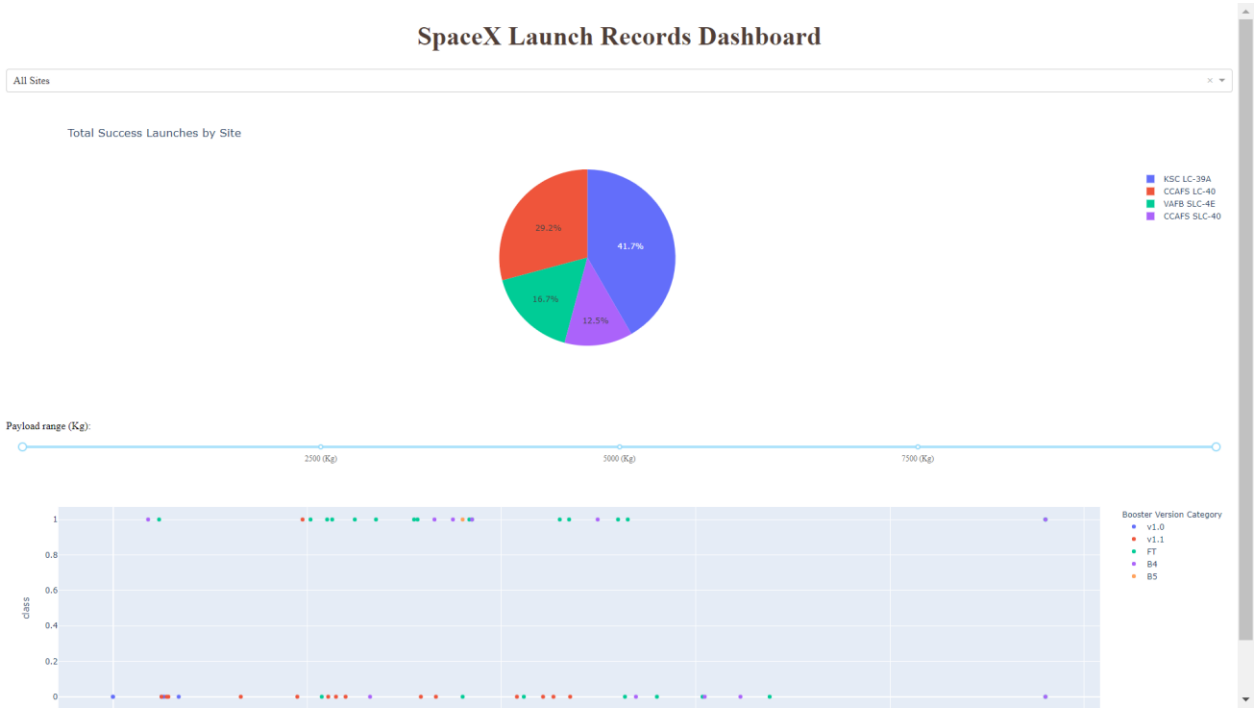
Predictive Analysis (Classification)



GitHub URL

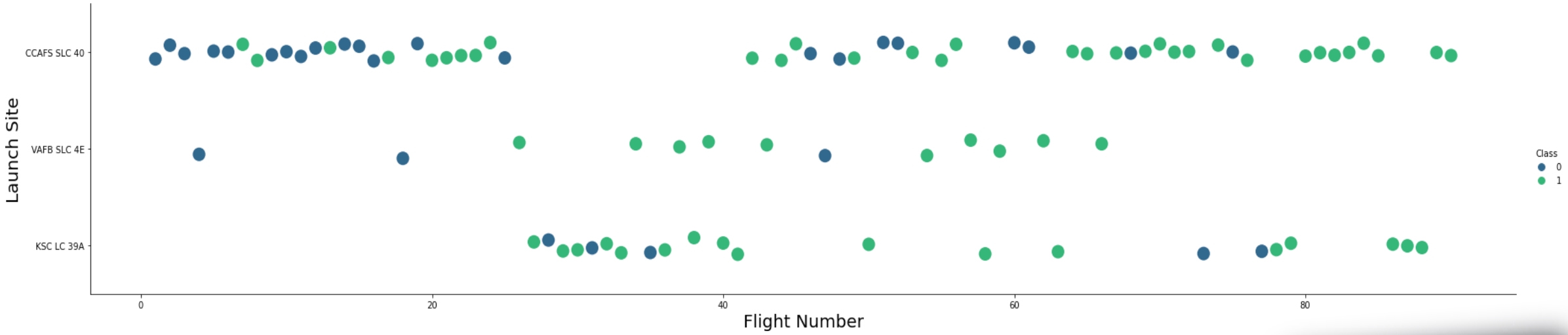
https://github.com/Giogio2021/Final_Proj_Data/blob/main/Mod_4/SpaceX_Machine%20Learning%20Prediction_Part_5_JM.ipynb

Results



This is a preview of the Plotly dashboard. The following sides will show the results of EDA with visualization, EDA with SQL, Interactive Map with Folium, and finally the results of our model with about 83% accuracy.

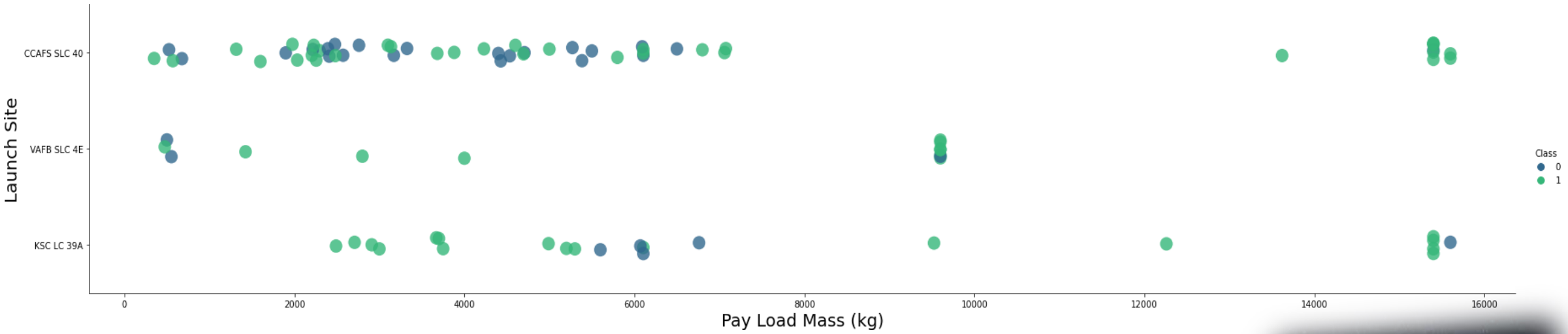
Flight Number vs. Launch Site



Green indicates successful launch; Purple 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



Green indicates successful launch; Purple indicates unsuccessful launch.

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

Different launch sites also seem to use different payload mass.

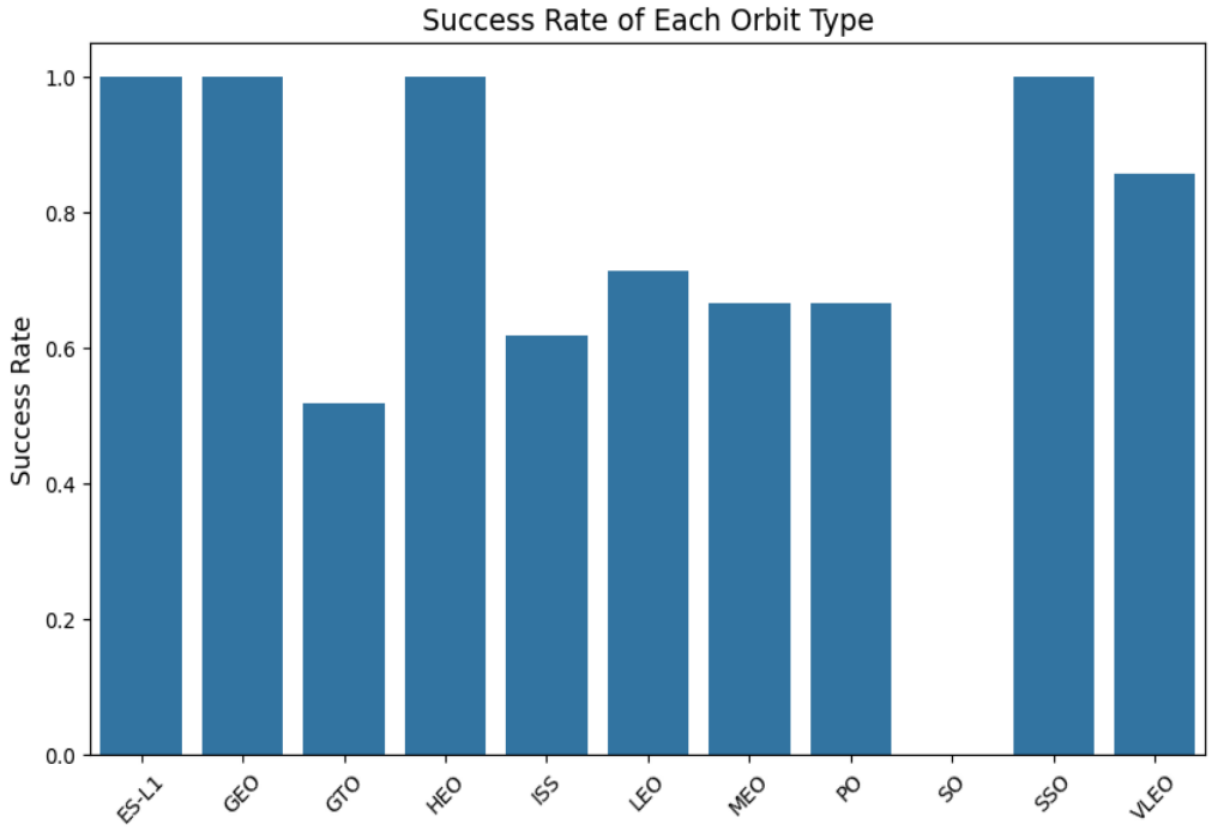
Success Rate vs. Orbit Type

ES-L1 (1), GEO (1), HEO (1) have 100% success rate (sample sizes in parenthesis) SSO (5) has 100% success rate

VLEO (14) has decent success rate and attempts

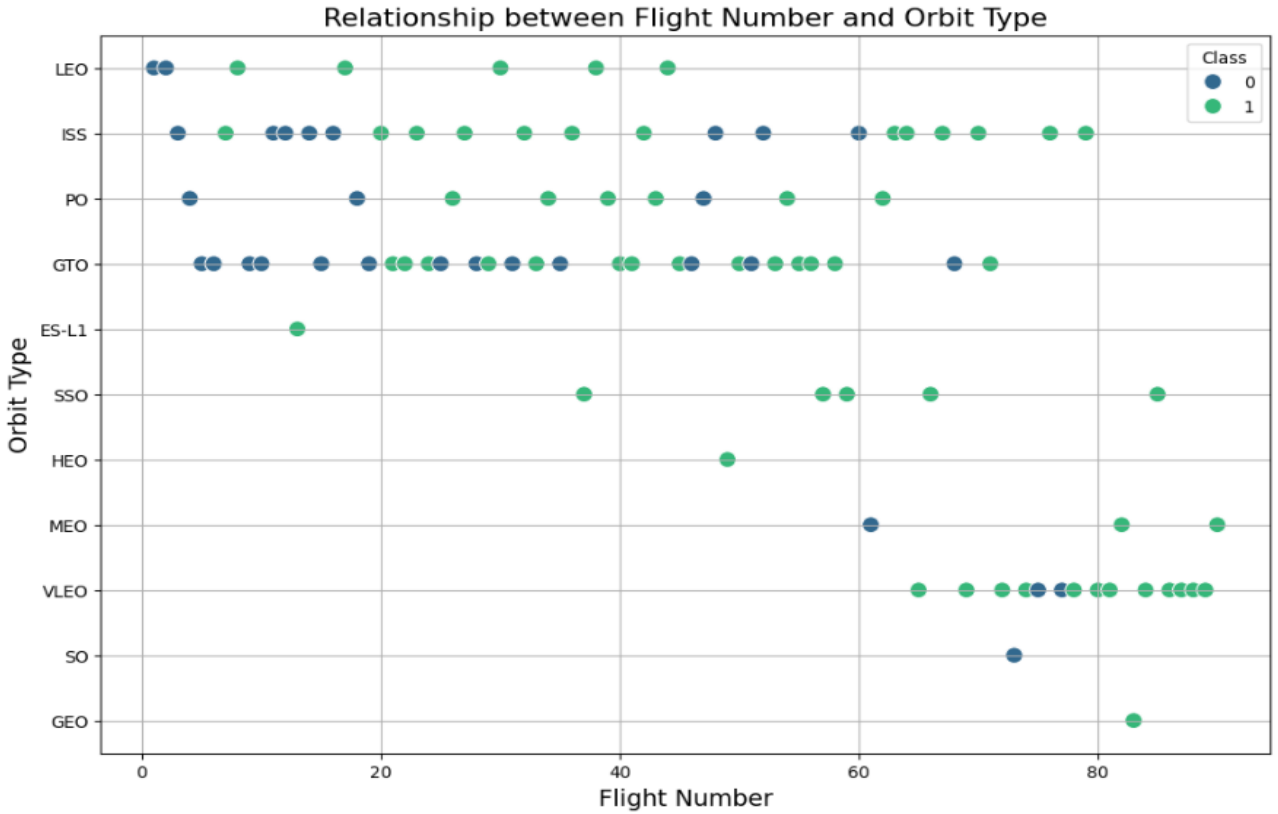
SO (1) has 0% success rate

GTO (27) has the around 50% success rate but largest sample



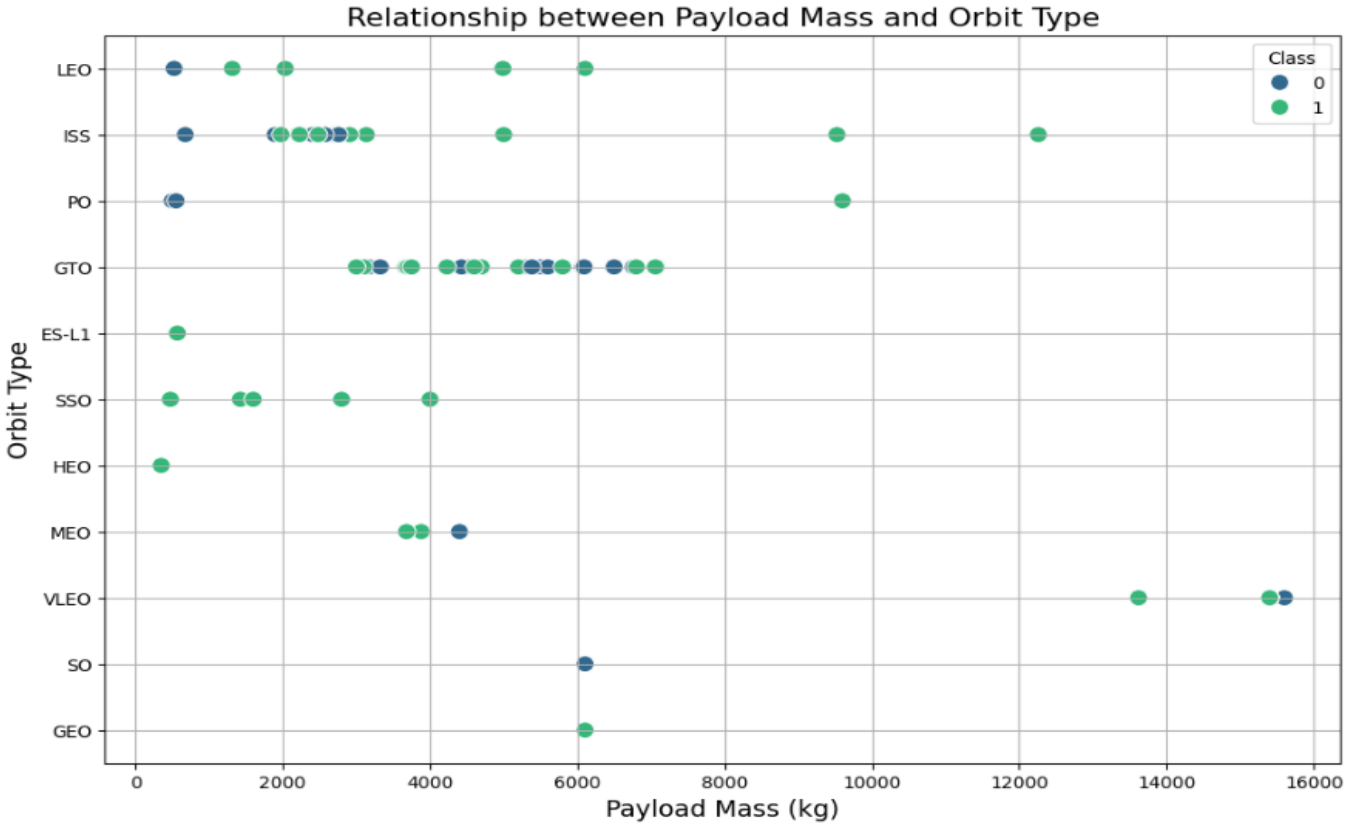
Success Rate Scale with
0 as 0%
0.6 as60%
1 as 100%

Flight Number vs. Orbit Type



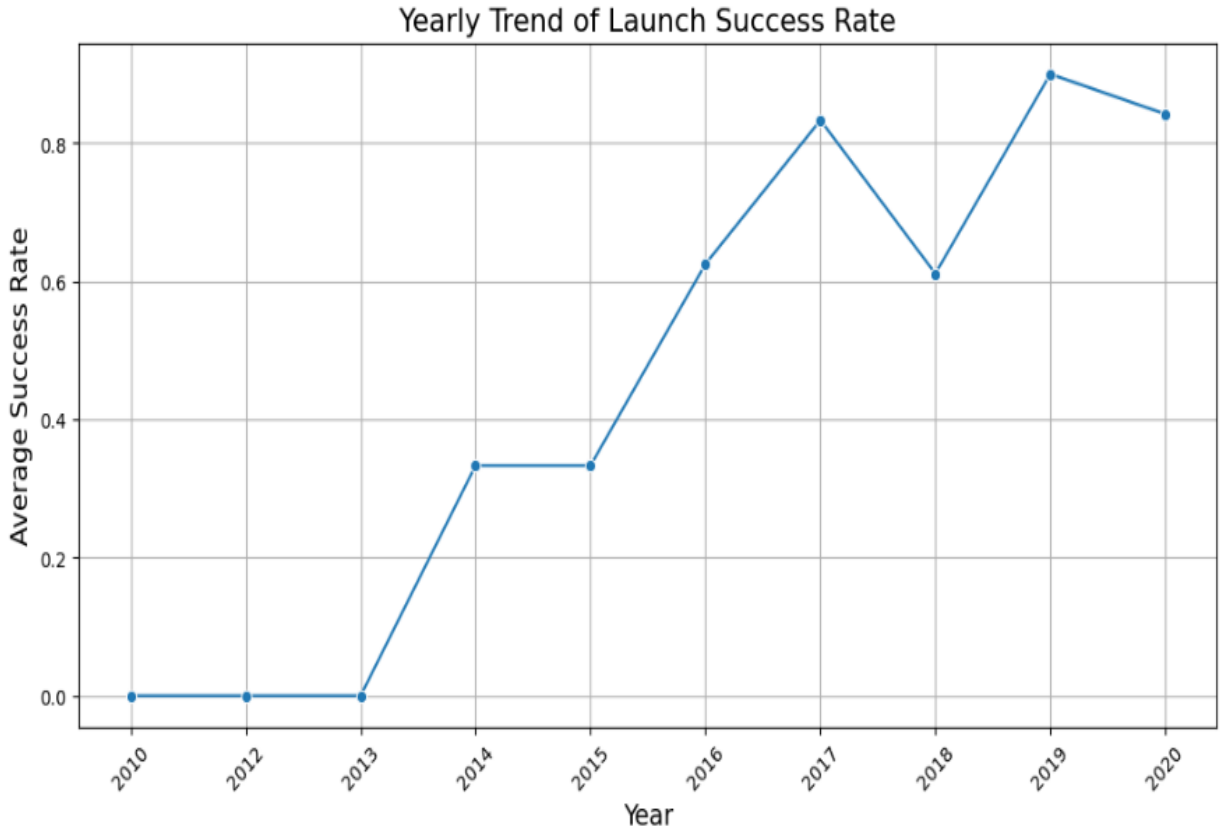
- Launch Orbit preferences changed over Flight Number.
- Launch Outcome seems to correlate with this preference.
- SpaceX started with LEO orbits which saw moderate success LEO and returned to VLEO in recent launches.
- SpaceX appears to perform better in lower orbits or Sun-synchronous orbits

Payload vs. Orbit Type



- 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



Success generally increases over time since 2013 with a slight dip in 2018

Success in recent years at around 80%

All Launch Site Names


```
%sql SELECT DISTINCT "Launch_Site" FROM SPACEXTABLE;
```

```
 * 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 CCAFSSLC-40 likely all represent the same launch site with data entry errors.

CCAFS LC-40 was the previous

name. Likely only 3 unique

launch_site values: CCAFS SLC-40,

KSC LC-39A,VAFB SLC-4E

Launch Site Names Begin with 'CCA'


```
%sql SELECT * FROM SPACEXTABLE WHERE "Launch_Site" LIKE 'CCA%' LIMIT 5;
```

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

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

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

Total Payload Mass

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

CRS stands for Commercial Resupply Services which indicates that these payloads were sent to the International Space Station (ISS).

```
%sql SELECT SUM("Payload_Mass__kg_") AS Total_Payload_Mass FROM SPACEXTABLE WHERE "Booster_Version" LIKE '%NASA (CRS)%';
```

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

Average Payload Mass by F9 v1.1


```
%sql SELECT AVG("Payload_Mass__kg_") AS Average_Payload_Mass FROM SPACEXTABLE WHERE "Booster_Version" = 'F9 v1.1';
```

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

Average_Payload_Mass
2928.4

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

Average payload mass of F9 1.1 is on the low endof our payload mass range

First Successful Ground Landing Date

This query returns the first successful ground pad landing date.

First ground pad landing wasn't until the end of 2015.

Successful landings in general appear starting 2014.

```
%sql SELECT MIN("Date") AS First_Successful_Landing FROM SPACEXTABLE WHERE "Landing_Outcome" = 'Success (ground pad)';
```

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

First_Successful_Landing
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

This query returns the four booster versions that had successful drone ship landings and a payload mass between 4000 and 6000 noninclusively.

```
%sql SELECT DISTINCT "Booster_Version" FROM SPACEXTABLE WHERE "Landing_Outcome" = 'Success (drone ship)' AND "Payload_Mass_
```

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

Booster_Version
F9 FT B1022
F9 FT B1026
F9 FT B1021.2
F9 FT B1031.2

Total Number of Successful and Failure Mission Outcomes


```
%sql SELECT "Mission_Outcome", COUNT(*) AS Total_Count FROM SPACEXTABLE GROUP BY "Mission_Outcome";
```

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

Mission_Outcome	Total_Count
Failure (in flight)	1
Success	98
Success	1
Success (payload status unclear)	1

This query returns a count of each mission outcome.

SpaceX appears to achieve its mission outcome nearly 99% of the time.

This means that most of the landing failures are intended.

Interestingly, one launch has an unclear payload status and unfortunately one failed in flight.

Boosters Carried Maximum Payload


```
%sql SELECT "Booster_Version" FROM SPACEXTABLE WHERE "Payload_Mass__kg_" = (SELECT MAX("Payload_Mass__kg_") FROM SPACEXTABL
```

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

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

This query returns a count of each mission outcome.

SpaceX appears to achieve its mission outcome nearly 99% of the time.

This means that most of the landing failures are intended.

Interestingly, one launch has an unclear payload status and unfortunately one failed in flight.

2015 Launch Records

This query returns the Month, Landing Outcome, Booster Version, Payload Mass (kg), and Launch site of 2015 launches where stage 1 failed to land on a drone ship.

There were two such occurrences.

```
%sql SELECT SUBSTR("Date", 6, 2) AS Month, "Landing_Outcome", "Booster_Version", "Launch_Site" FROM SPACEXTABLE WHERE "Landi
```

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

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

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


```
%sql SELECT "Landing_Outcome", COUNT(*) AS Outcome_Count FROM SPACEXTABLE WHERE "Date" BETWEEN '2010-06-04' AND '2017-03-20'
```

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

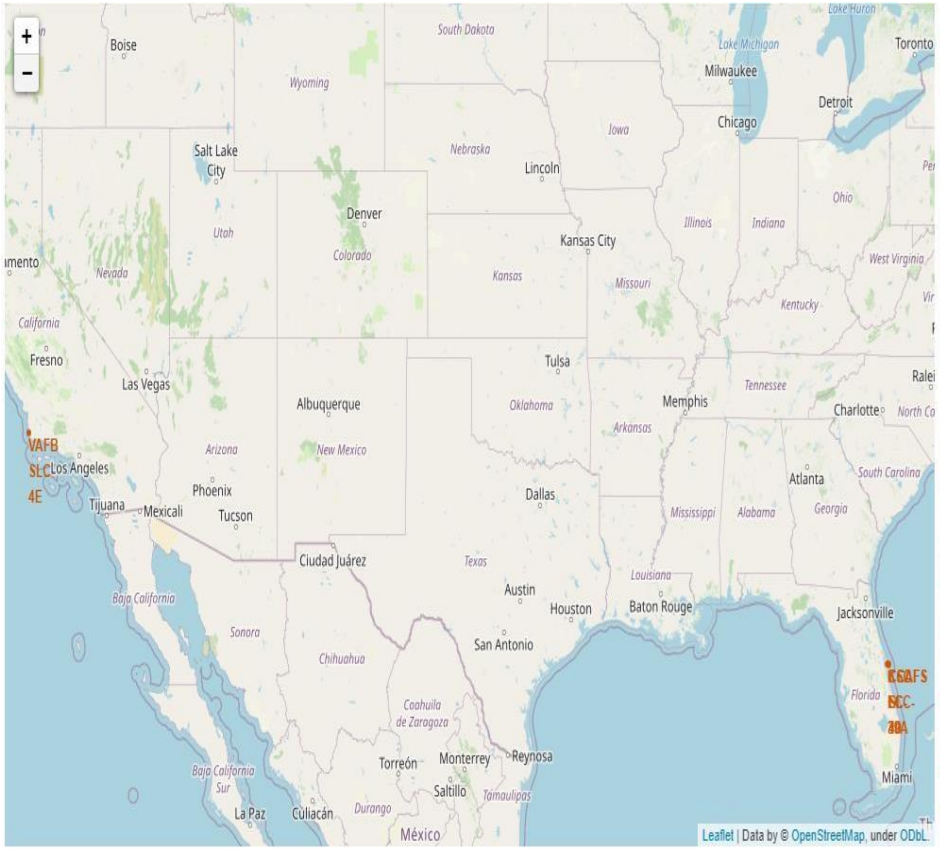
Landing_Outcome	Outcome_Count
No attempt	10
Success (drone ship)	5
Failure (drone ship)	5
Success (ground pad)	3
Controlled (ocean)	3
Uncontrolled (ocean)	2
Failure (parachute)	2
Precluded (drone ship)	1

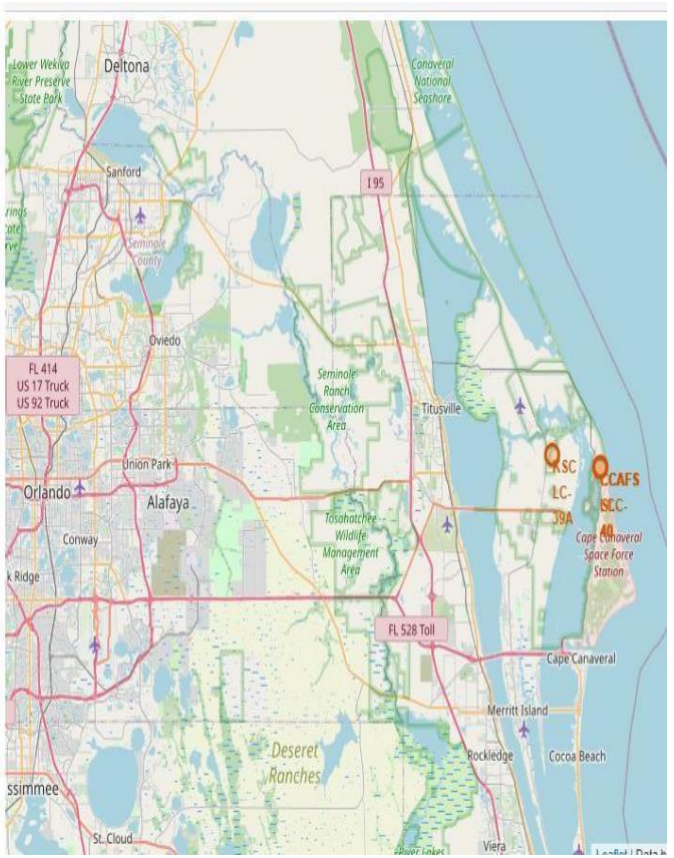
This query returns a list of successful landings and between 2010-06-04 and 2017-03-20 inclusively.

There are two types of successful landing outcomes: drone ship and ground pad landings.

There were 5 + 3 successful landings in total during this time period

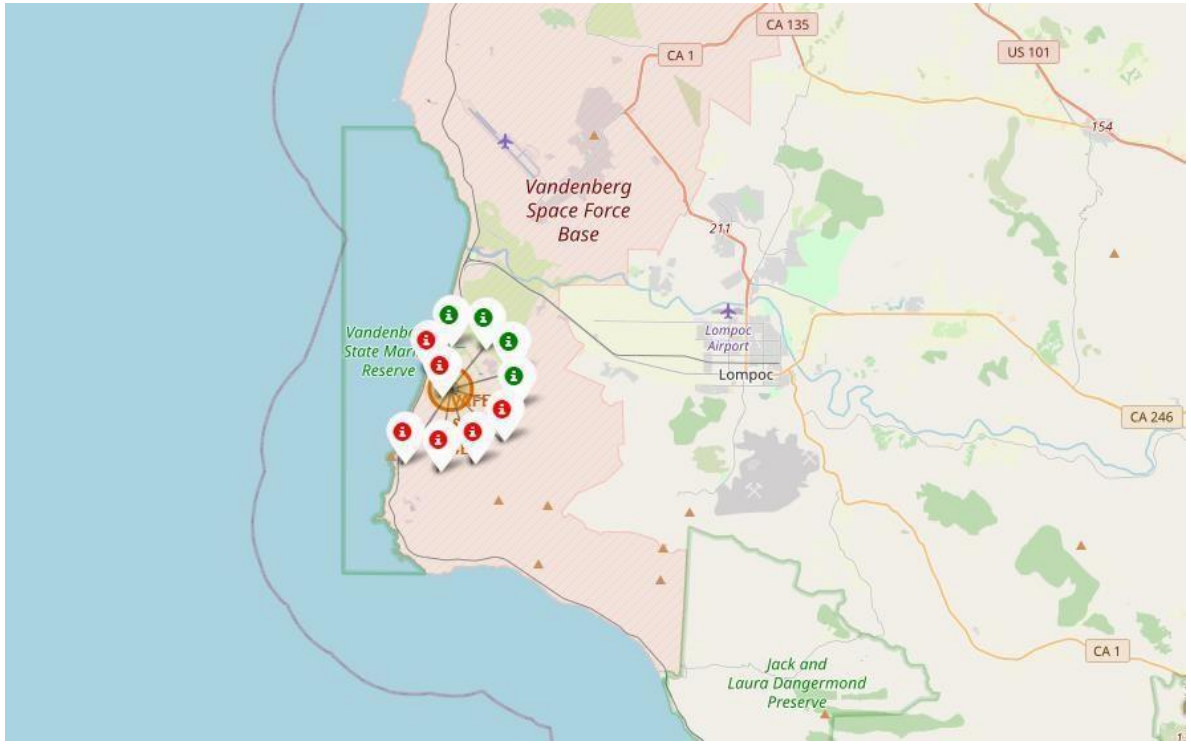
Folium Map Screenshot 1





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.

Folium Map Screenshot 2

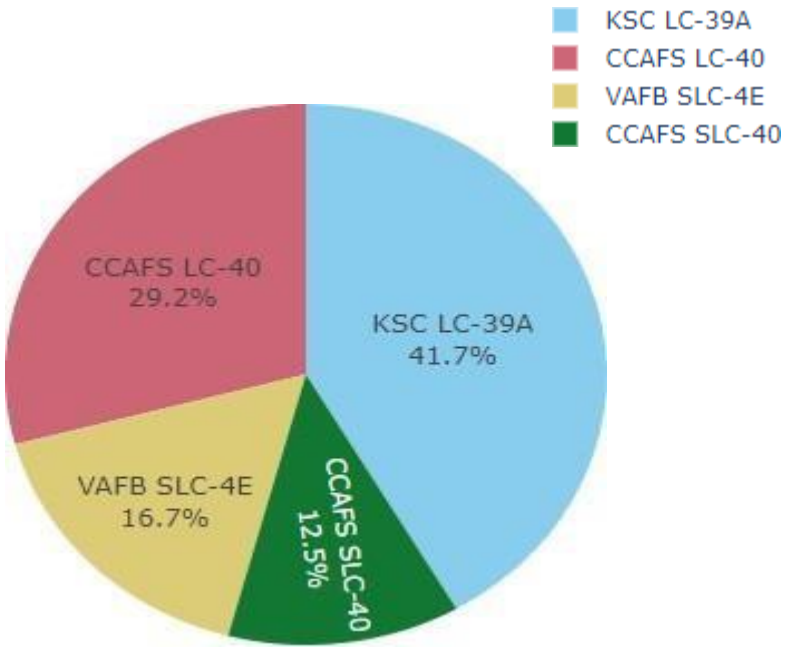


Clusters on Folium map can be clicked on to display each successful landing (green icon) and failed landing (red icon). In this example VAFB SLC-4E shows 4 successful landings and 6 failed landings.

Folium Map Screenshot 3

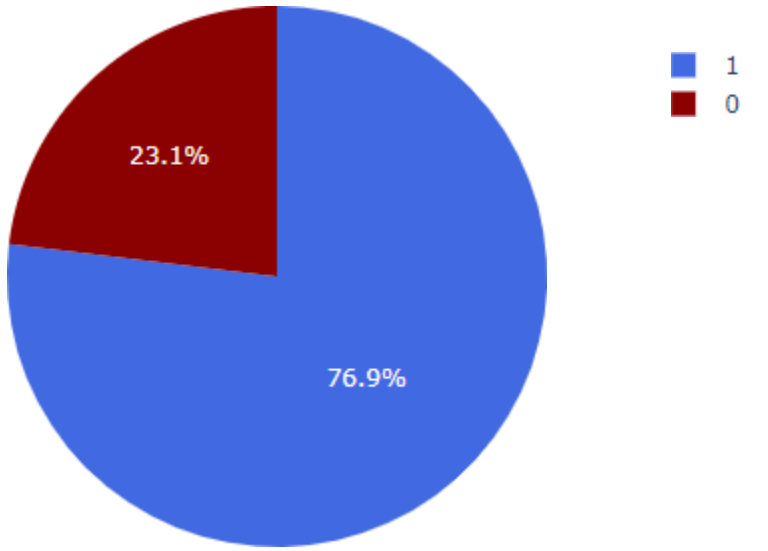
Using KSC LC-39A as an example, launch sites are very close to railways for large part and supply transportation. Launch sites are close to highways for human and supply transport. Launch sites are also close to coasts and relatively far from cities so that launch failures can land in the sea to avoid rockets falling on densely populated areas.

Dashboard Screenshot 1



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.

Dashboard Screenshot 2



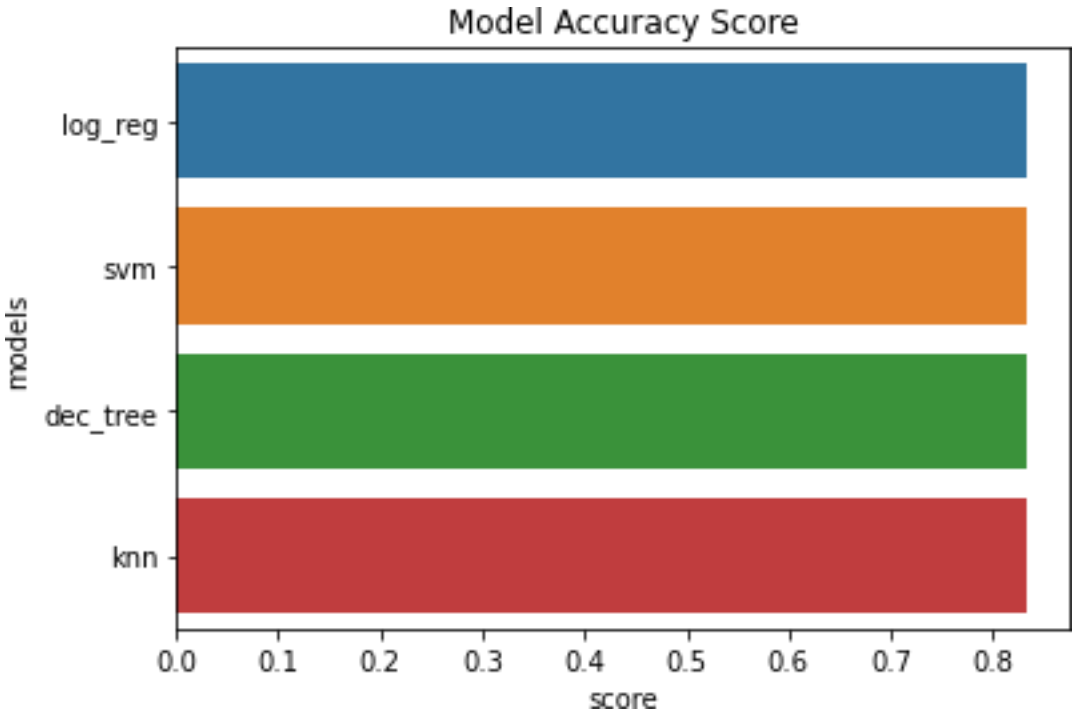
KSC LC-39A Success Rate (blue=success)

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

Dashboard Screenshot 3

Plotly dashboard has a Payload range selector. However, this is set from 0-10000 instead of the max Payload of 15600. Class indicates 1 for successful landing and 0 for failure. Scatter plot also accounts for booster version category in color and number of launches in point size. In this particular range of 0-6000, interestingly there are two failed landings with payloads of zero kg.

Classification Accuracy



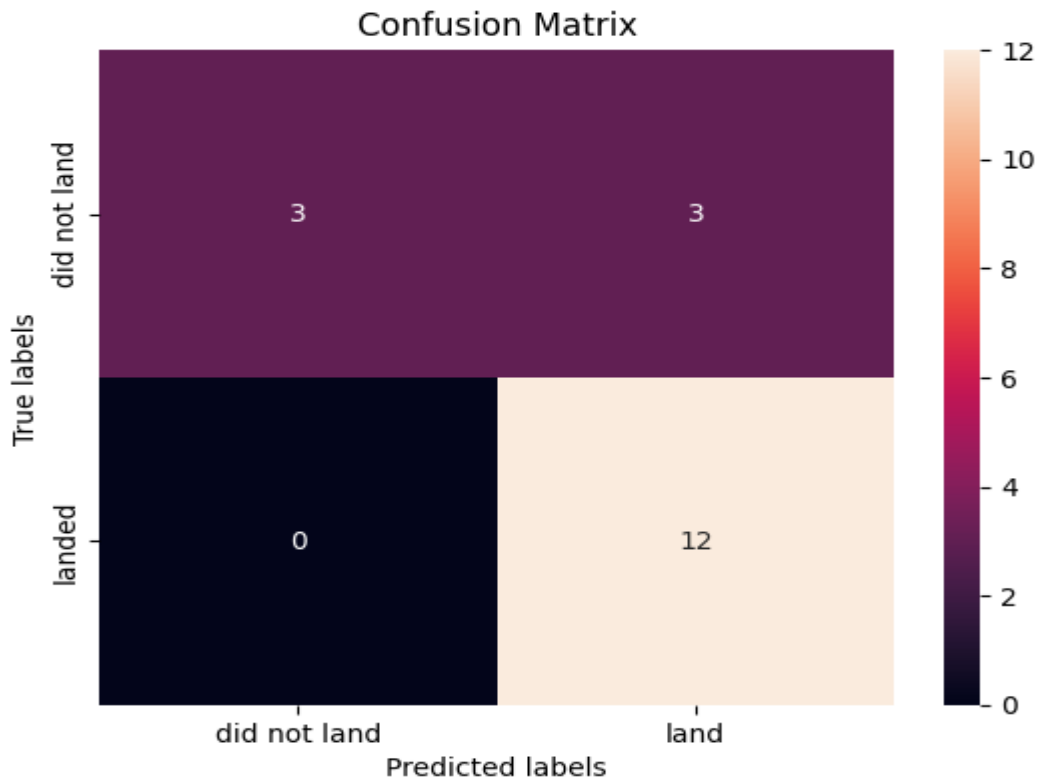
All models had virtually the same accuracy on the test set at 83.33% accuracy. It should be noted that test size is small at only sample size of 18.

This can cause large variance in accuracy results, such as those in Decision Tree Classifier model in repeated runs.

We likely need more data to determine the best model.

Confusion Matrix


```
yhat=logreg_cv.predict(X_test)
plot_confusion_matrix(Y_test,yhat)
```



Since all models performed the same for the test set, the confusion matrix is the same across all models. The models predicted 12 successful landings when the true label was successful landing.

The models predicted 3 unsuccessful landings when the true label was unsuccessful landing.

The models predicted 3 successful landings when the true label was unsuccessful landings (false positives). Our models over predict successful landings.

Correct predictions are on a diagonal from top left to bottom right.

Conclusions

- Task: Develop a machine learning model to assist SpaceY in bidding against SpaceX.
- Goal: Predict successful Stage 1 landings to save approximately $100 million USD per landing.
- Data:
  - Public SpaceX API
  - Web scraping of SpaceX Wikipedia page
- Data Processing:
  - Created data labels
  - Stored data in a DB2 SQL database
  - Developed a visualization dashboard
- Model Development:
  - Built a machine learning model achieving an accuracy of 83%.
- Implications for SpaceY:
  - Elon Musk can leverage this model to predict the likelihood of a successful Stage 1 landing before a launch.
  - This information can inform strategic decisions about whether to proceed with a launch.
- Future Directions:
  - Collecting additional data can further refine the model and potentially improve its accuracy.

Appendix

Credits and Acknowledgments

Primary Instructors:

- Joseph Santarcangelo
- Yan Luo

Special Thanks to All Instructors

Data Science Capstone Project

Jorge Mariotti
https://github.com/Giogio2021/Final_Proj_Data
11/2024