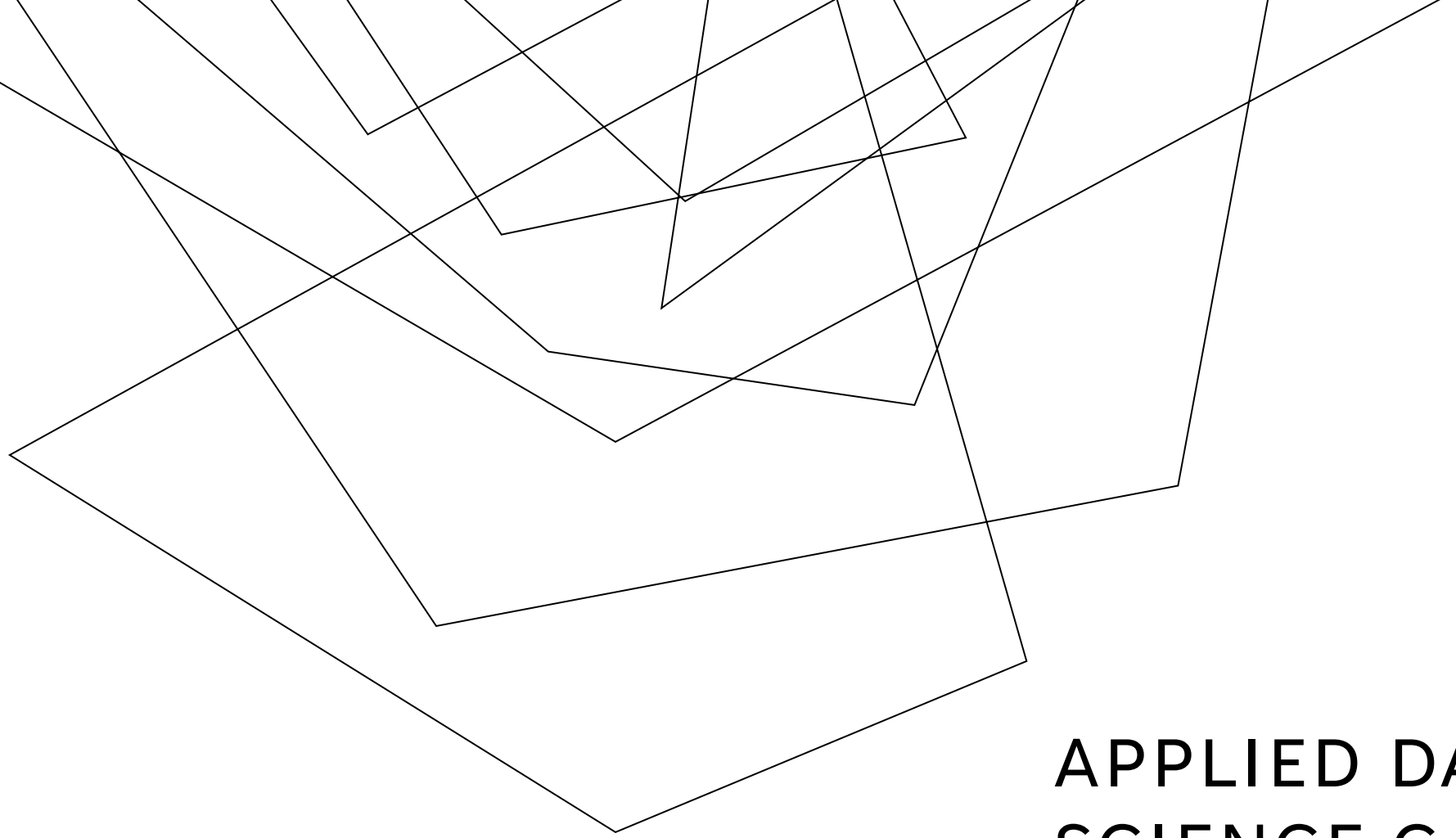


APPLIED DATA SCIENCE CAPSTONE - JOURNEY

AGENDA

- Executive Summary slide (1 pt)
- Introduction slide (1 pt)
- data collection and data wrangling methodology related slides (1 pt)
- EDA and interactive visual analytics methodology related slides (3 pts)
- predictive analysis methodology related slides (1 pt)
- EDA with visualization results slides (6 pts)
- EDA with SQL results slides (10 pts)
- interactive map with Folium results slides (3 pts)
- Plotly Dash dashboard results slides (3 pts)
- predictive analysis (classification) results slides (6 pts)
- Conclusion slide (1 pts)
- Applied your creativity to improve the presentation beyond the template (1 pts)
- Displayed any innovative insights (1 pts)

EXECUTIVE SUMMARY & INTRO

SUMMARY

The **IBM Data Science Professional Certificate** on Coursera provides a comprehensive introduction to key data science skills, including Python programming, data analysis, machine learning, and data visualization. Designed for beginners, the program offers hands-on projects and prepares learners for entry-level data science roles by building real-world, job-ready expertise. Let us see how I accomplished all the requirements in the following slides.

Junyper notebooks can be found under the following GitHub:

https://github.com/CavdadHun/Datascience_cert

INTRODUCTION

- Collected data from public SpaceX API and SpaceX Wikipedia page. Created labels column ‘class’ which classifies successful landings. Explored data using SQL, visualization, folium maps, and dashboards. Gathered relevant columns to be used as features. Changed all categorical variables to binary using one hot encoding. Standardized data and used GridSearchCV to find best parameters for machine learning models. Visualize accuracy score of all models.
- Four machine learning models were produced: Logistic Regression, Support Vector Machine, Decision Tree Classifier, and K Nearest Neighbors. All produced similar results with accuracy rate of about 83.33%. All models over predicted successful landings. More data is needed for better model determination and accuracy.

Background:

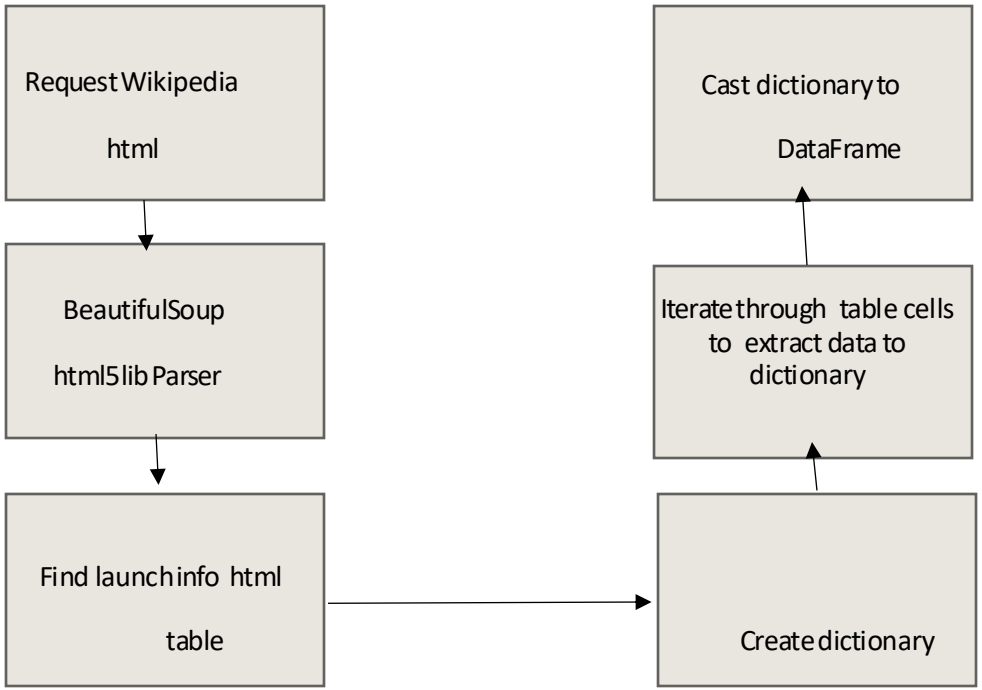
- Commercial Space Age is approaching
- Space X has best pricing primarily due to ability to recover part of rocket (Stage 1)
- Space Y wants to compete with Space X

Problem:

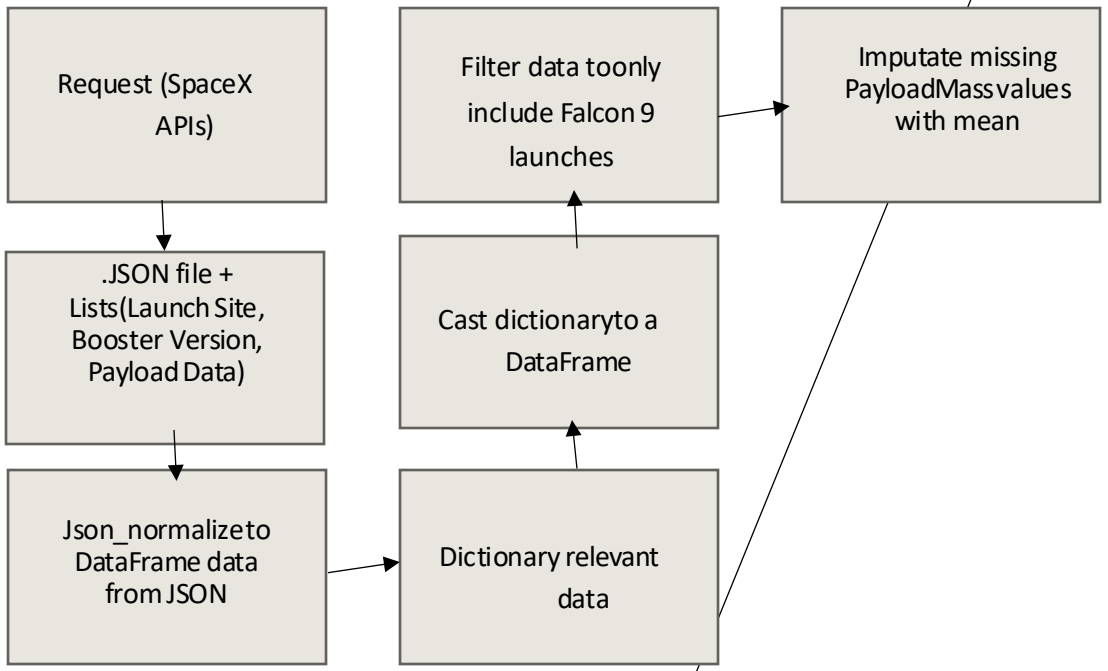
- Space Y tasks us to train a machine learning model to predict successful Stage 1 recovery

DATA COLLECTION AND DATA WRANGLING METHODOLOGY

DATA COLLECTION –WEB SCRAPING



DATA COLLECTION –SPACEX API



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

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

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

BUILD AN INTERACTIVE MAP WITH FOLIUM

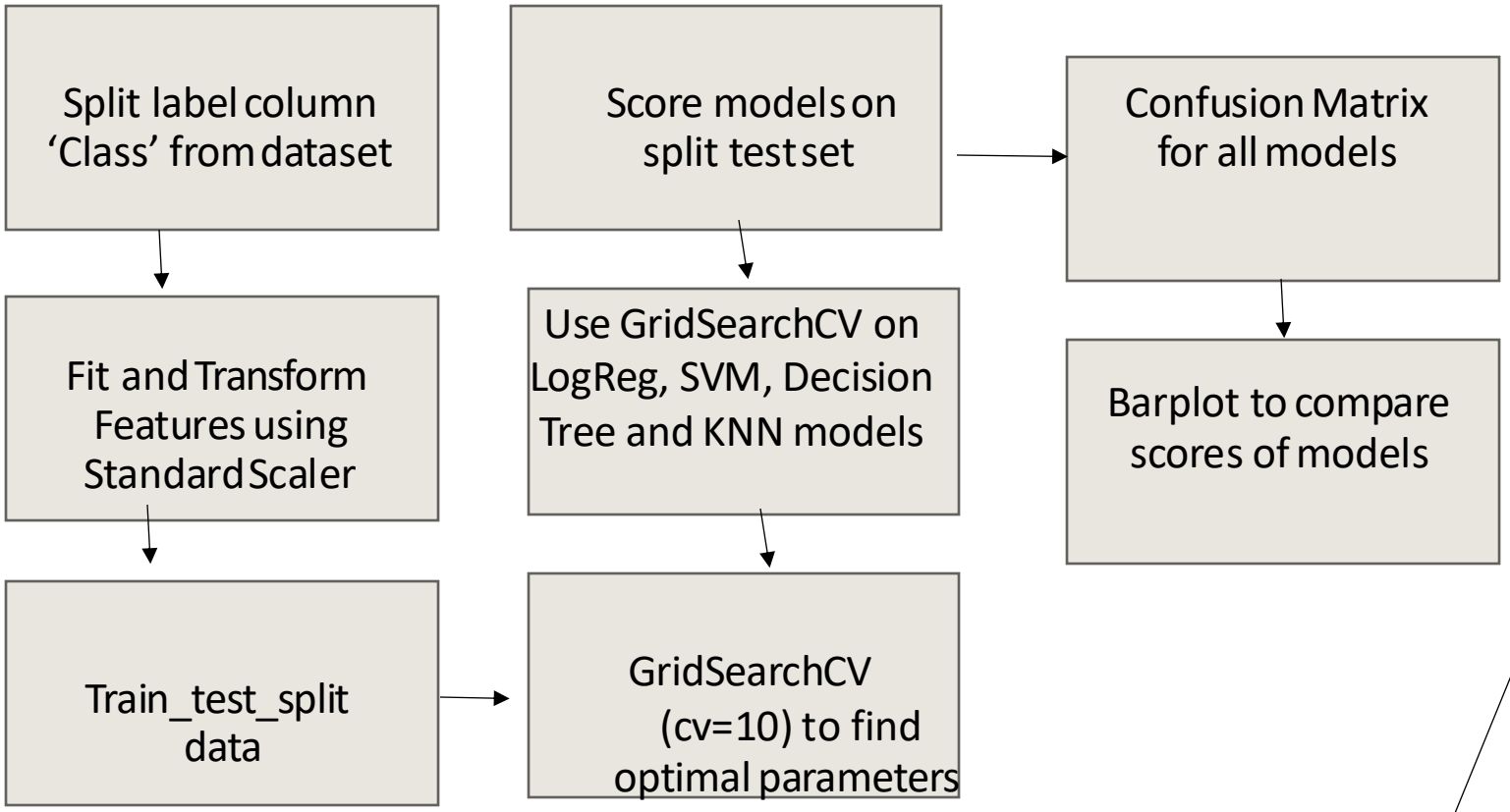
Folium maps mark Launch Sites, successful and unsuccessful landings, and a proximity example to key locations: Railway, Highway, Coast, and City.

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

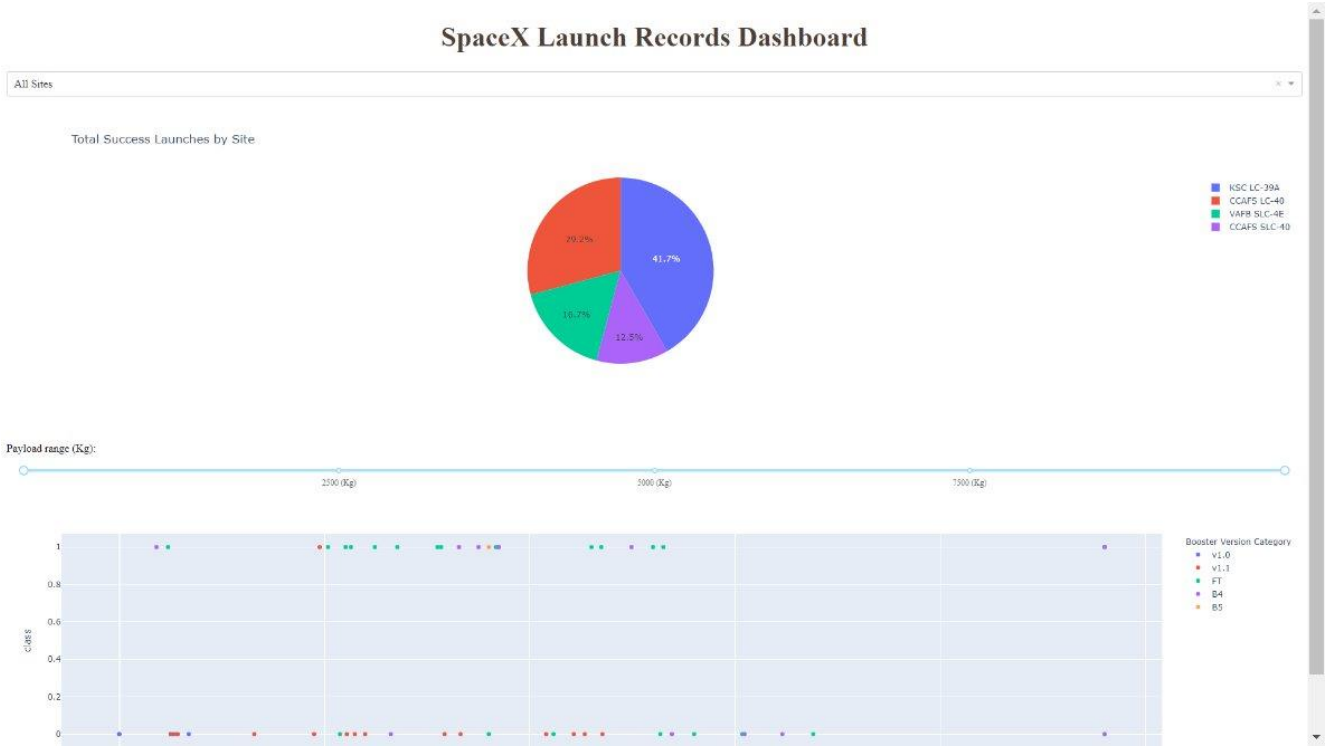
Build a Dashboard with Plotly Dash

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.

PREDICTIVE ANALYSIS (CLASSIFICATION)



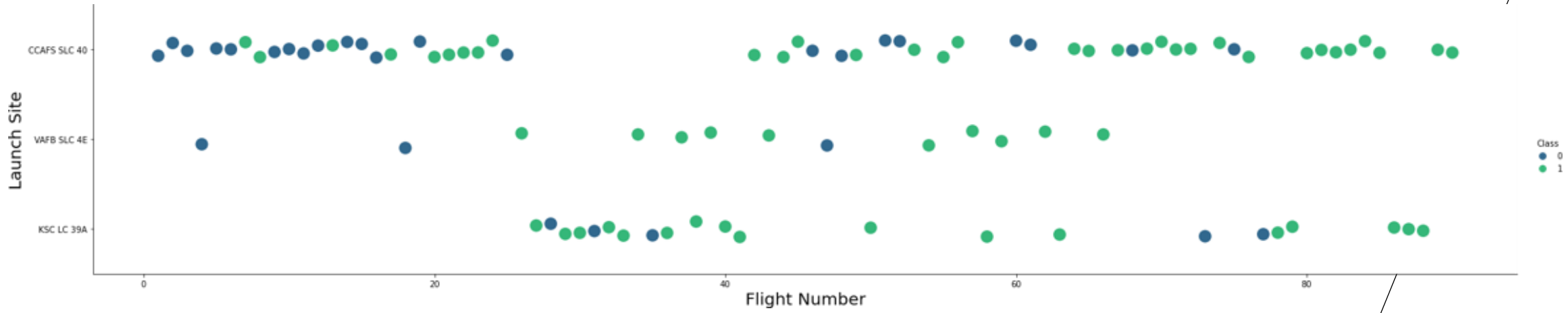
RESULTS OF MODELLING



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.

E D A WITH VISUALIZATION

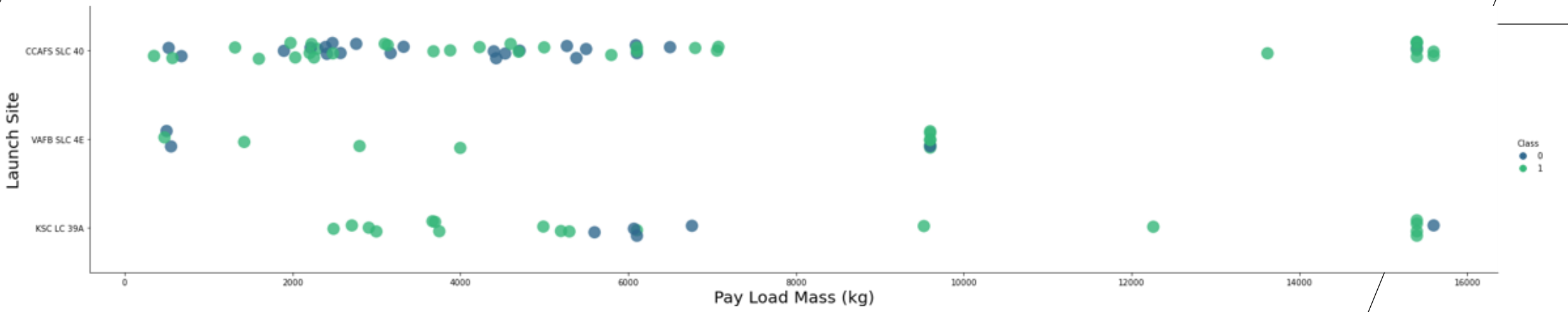
FLIGHT NUMBER VS LAUNCH SITE



The graphic indicates a rising success rate over time, as shown by the increasing flight numbers. A major breakthrough seems to have occurred around flight 20, leading to a significant improvement in success rate. CCAFS stands out as the primary launch site, handling the highest number of launches.

Legend:
Green: Successful
Purple: Unsuccessful

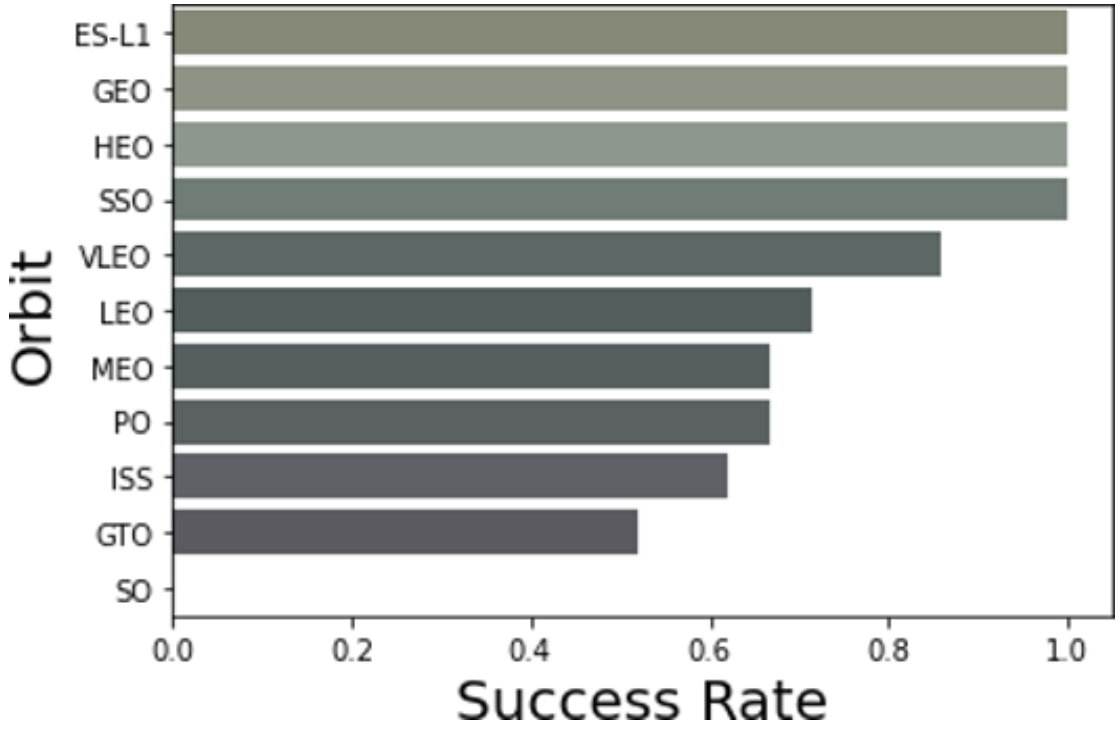
PAYLOAD VS. LAUNCH SITE



Legend:
Green: Successful
Purple: Unsuccessful

Payload mass is predominantly clustered between 0 and 6,000 kg. There also appears to be variation in payload mass depending on the launch site.

SUCCESS RATE VS. ORBIT TYPE



- ES-L1 (1), GEO (1), and HEO (1) all show a 100% success rate, though each has only a single launch.
- SSO (5) also achieved a 100% success rate across five attempts.
- VLEO (14) demonstrates a relatively strong success rate with a fair number of launches.
- SO (1) has a 0% success rate based on a single attempt.
- GTO (27), despite having the largest number of launches, shows an approximate 50% success rate.

EDA WITH SQL

ALL LAUNCH SITE NAMES

In [4]:
```
%%sql
SELECT UNIQUE LAUNCH_SITE
FROM SPACEXDATASET;
```

```
 * ibm_db_sa://ftb12020:***@0c77d6f2
Done.
```

Out[4]:

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

LAUNCH SITE NAMES BEGINNING WITH `CCA`

In [5]:
```
%%sql
SELECT *
FROM SPACEXDATASET
WHERE LAUNCH_SITE LIKE 'CCA%'
LIMIT 5;
```

```
 * ibm_db_sa://ftb12020:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/bludb
Done.
```

Out[5]:

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 attempt
2012-10-08	00:35:00	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

EDA WITH SQL

TOTAL PAYLOAD MASS FROM NASA

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

```
 * ibm_db_sa://ftb12020:***@0c77d6f2-5da9-48a9-81f8-86
Done.
```

sum_payload_mass_kg
45596

AVERAGE PAYLOAD MASS BY F9 V1.1

```
%%sql
SELECT AVG(PAYLOAD_MASS__KG_) AS AVG_PAYLOAD_MASS_KG
FROM SPACEXDATASET
WHERE booster_version = 'F9 v1.1'
```

```
 * ibm_db_sa://ftb12020:***@0c77d6f2-5da9-48a9-81f8-86
Done.
```

avg_payload_mass_kg
2928

This query calculates the total payload mass (in kg) for launches where NASA was the customer.
CRS refers to Commercial Resupply Services, meaning these payloads were delivered to the International Space Station (ISS).

EDA WITH SQL

FIRST SUCCESSFUL GROUND PAD LANDING DATE

```
%%sql
SELECT MIN(DATE) AS FIRST_SUCCESS
FROM SPACEXDATASET
WHERE landing__outcome = 'Success (ground pad)';
```

```
 * ibm_db_sa://ftb12020:***@0c77d6f2-5da9-48a9-81
Done.
```

first_success
2015-12-22

SUCCESSFUL DRONE SHIP LANDING WITH PAYLOAD BETWEEN 4000 AND 6000

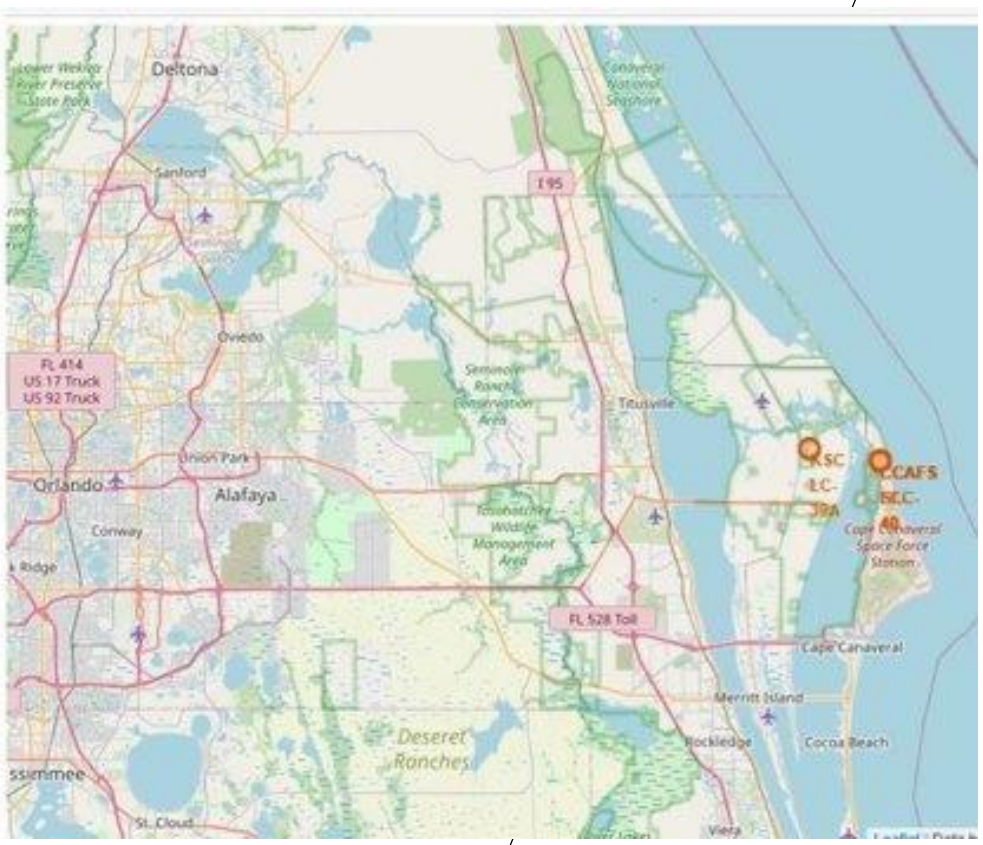
```
%%sql
SELECT booster_version
FROM SPACEXDATASET
WHERE landing__outcome = 'Success (drone ship)' AND payload_mass__kg_ BETWEEN 4001 AND 5999;
```

```
 * ibm_db_sa://ftb12020:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.database
Done.
```

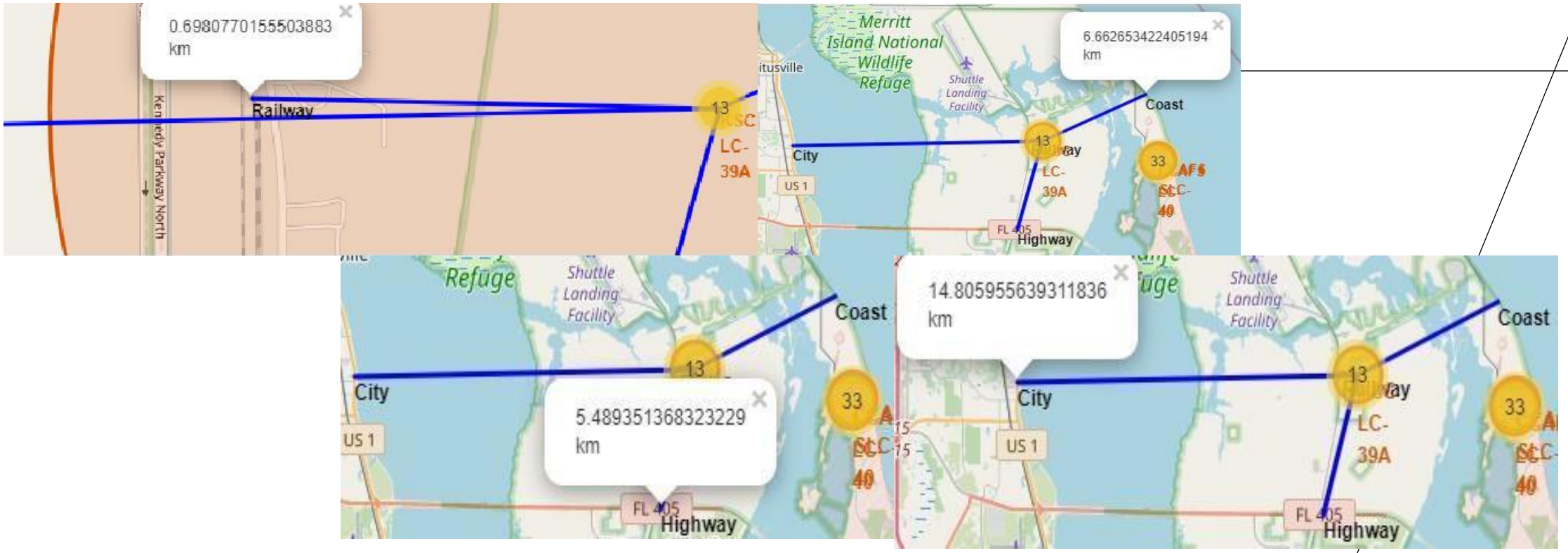
booster_version
F9 FT B1022
F9 FT B1026
F9 FT B1021.2
F9 FT B1031.2

INTERACTIVE MAP WITH FOLIUM

LAUNCHSITE LOCATIONS



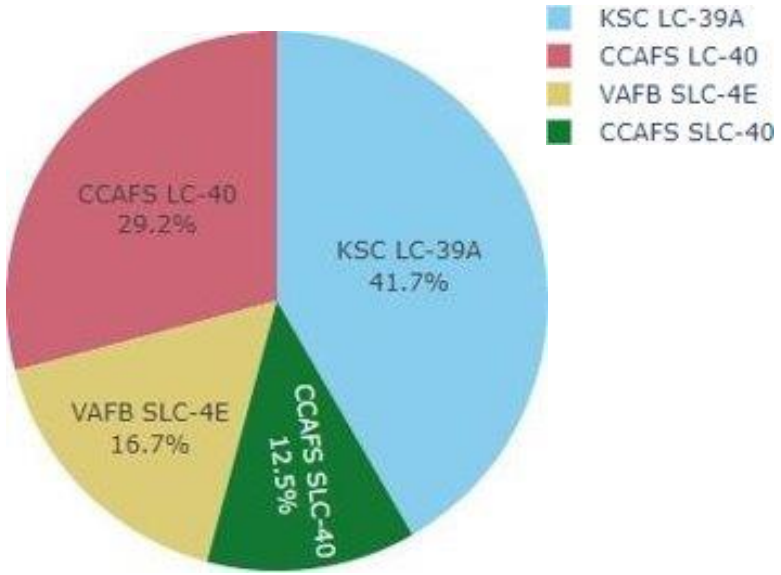
KEY LOCATIONS



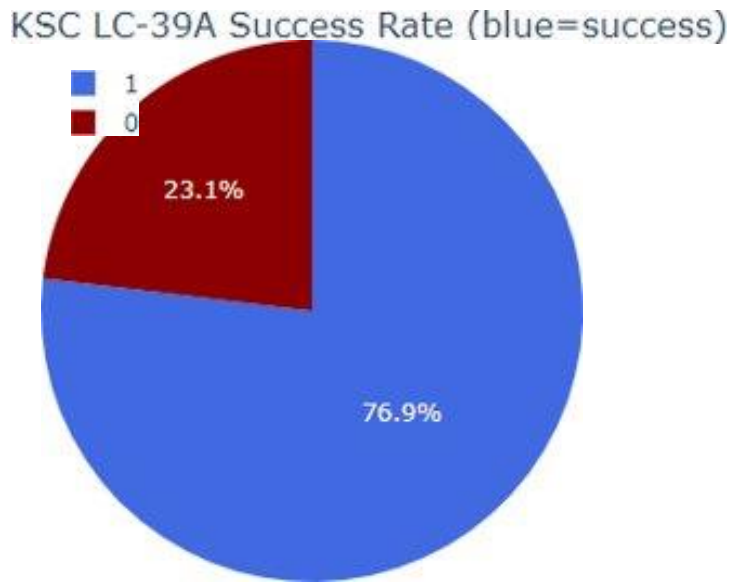
Using KSC LC-39A as an example, launch sites are typically situated near railways to support the transport of large components and supplies. They are also located close to highways for efficient movement of personnel and materials. Additionally, launch sites are positioned near coastlines and at a distance from major cities to ensure that, in the event of a launch failure, debris can fall safely into the ocean rather than in populated areas.

BUILD A DASHBOARD WITH PLOTLY DASH

Successful Launches Across Launch Sites

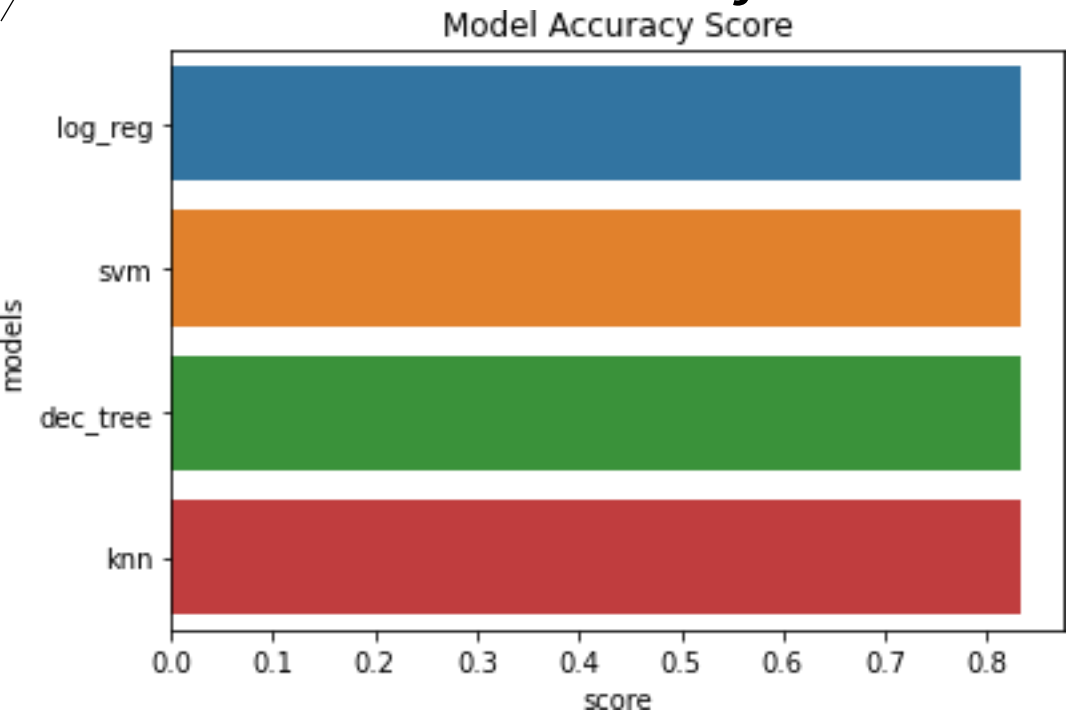


Highest Success Rate Launch Site



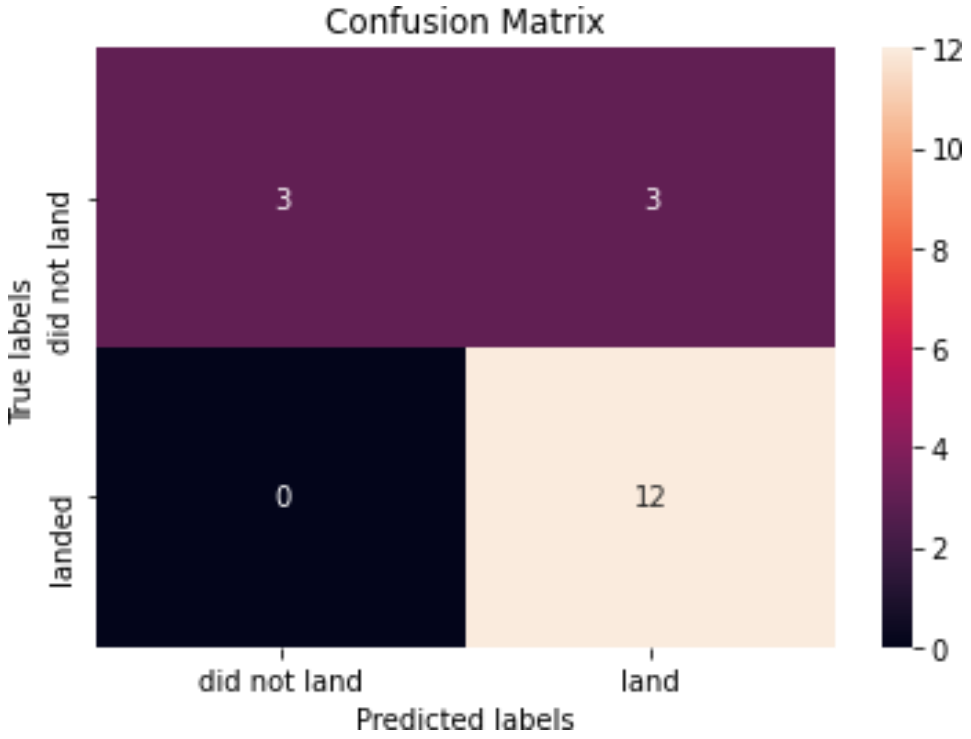
PREDICTIVE ANALYSIS (CLASSIFICATION)

Classification Accuracy



All models achieved similar performance on the test set, each with an accuracy of 83.33%. However, the test set is quite small, containing only 18 samples, which can lead to significant variability in accuracy—especially noticeable with models like the Decision Tree Classifier across repeated runs.
To reliably identify the best-performing model, a larger dataset is likely needed.

CONFUSION MATRIX



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

Since all models performed identically on the test set, their confusion matrices are also identical. The models correctly predicted 12 successful landings when the true label was a successful landing and correctly predicted 3 unsuccessful landings when the true label was unsuccessful.
However, they also incorrectly predicted 3 successful landings when the true label was actually an unsuccessful landing (false positives), indicating that the models tend to overpredict successful landings.

CONCLUSION

Our objective was to develop a machine learning model for SpaceY, which is aiming to compete with SpaceX.

The model’s goal is to predict the likelihood of a successful Stage 1 landing, helping to potentially save approximately $100 million USD per launch.

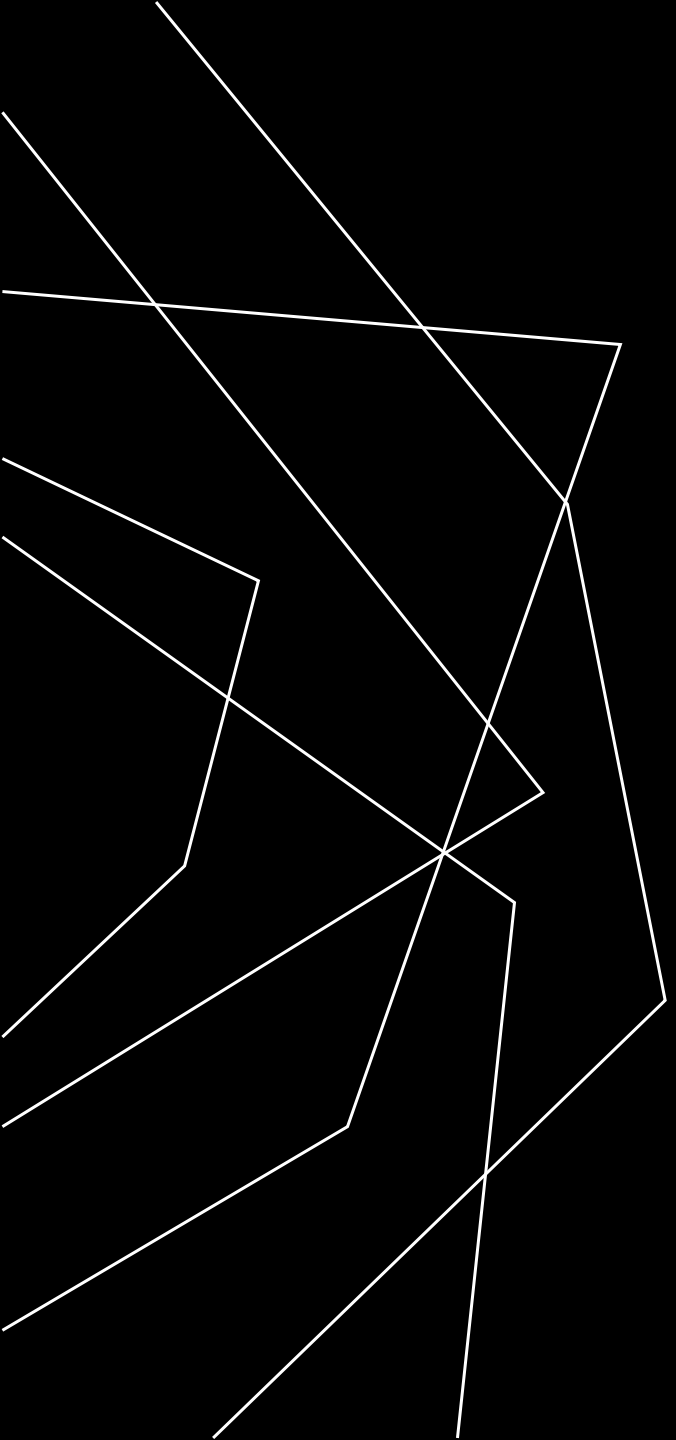
We sourced data from a public SpaceX API and by web scraping SpaceX's Wikipedia page. Data labels were created and stored in a DB2 SQL database.

A dashboard was also built to visualize the data and model results.

We developed a machine learning model achieving 83% accuracy.

Allon Mask, representing SpaceY, can use this model to predict Stage 1 landing success before launch, assisting in deciding whether to proceed with a given launch.

Collecting additional data is recommended to further improve model performance and better identify the most effective machine learning approach.



THANK YOU