Data Science Capstone Project

Chukwuka Orefo

09/11/2023

Outline

- Executive Summary (3)
- Introduction (4)
- Methodology (6)
- Results (16)
- Conclusion (46)
- Appendix (47)

Executive Summary

I collected data from both the public SpaceX API and the SpaceX Wikipedia page and created a 'class' column to classify successful landings. Using SQL, visualization, Folium maps, and dashboards, I explored the data comprehensively. Relevant columns were selected as features, categorical variables were converted into binary using one-hot encoding, and data standardization was performed. I used GridSearchCV to optimize machine learning model parameters and visualized the accuracy scores of all models, allowing for a thorough assessment of their performance in classifying successful landings.

- Four machine learning models were generated, including Logistic Regression, Support Vector Machine, Decision Tree Classifier, and K Nearest Neighbors. These models demonstrated a consistent performance, achieving an accuracy rate of approximately 83.33%. However, it is noteworthy that all of these models tended to over-predict successful landings, indicating a potential bias in the dataset or the need for a more balanced dataset. To enhance model determination and overall accuracy, it is evident that acquiring additional data or addressing dataset imbalances would be beneficial.

Introduction

SpaceX Falcon 9 Rocket – The Verge

Background

- In the era of commercial space exploration, SpaceX has taken the lead with its remarkably cost-effective launch services, priced at just $62 million USD compared to the industry average of $165 million USD. SpaceX's cost advantage stems from its groundbreaking rocket stage recovery technology. Now, Space Y aims to compete with SpaceX in this highly competitive market, promising innovation and competition that could reshape the commercial space industry.

Problem

- Space Y has tasked us with training a machine learning model to predict the successful recovery of Stage 1 rockets.

Methodology

- Certainly, here are the key steps in bullet points for the data collection methodology:
-
- - Combined data from SpaceX public API and SpaceX Wikipedia page.
- - Conducted data wrangling to clean and prepare the dataset.
- - Classified true landings as successful and others as unsuccessful.
- - Performed exploratory data analysis (EDA) using visualization and SQL.
- - Utilized interactive visual analytics tools like Folium and Plotly Dash.
- - Conducted predictive analysis using classification models.
- - Optimized model performance through parameter tuning using GridSearchCV.
-

Methodology

OVERVIEW OF DATA COLLECTION, WRANGLING,
VISUALIZATION,
DASHBOARD, AND MODEL METHODS

Data Collection Overview

Data Collection Process:

- Data collection involved a two-step process:
  - API Requests: We gathered data from SpaceX's public API by making requests to retrieve information on various rocket launches.
  - Web Scraping: Additionally, we scraped data from a table within SpaceX's Wikipedia entry, extracting relevant details.

SpaceX API Data Columns:

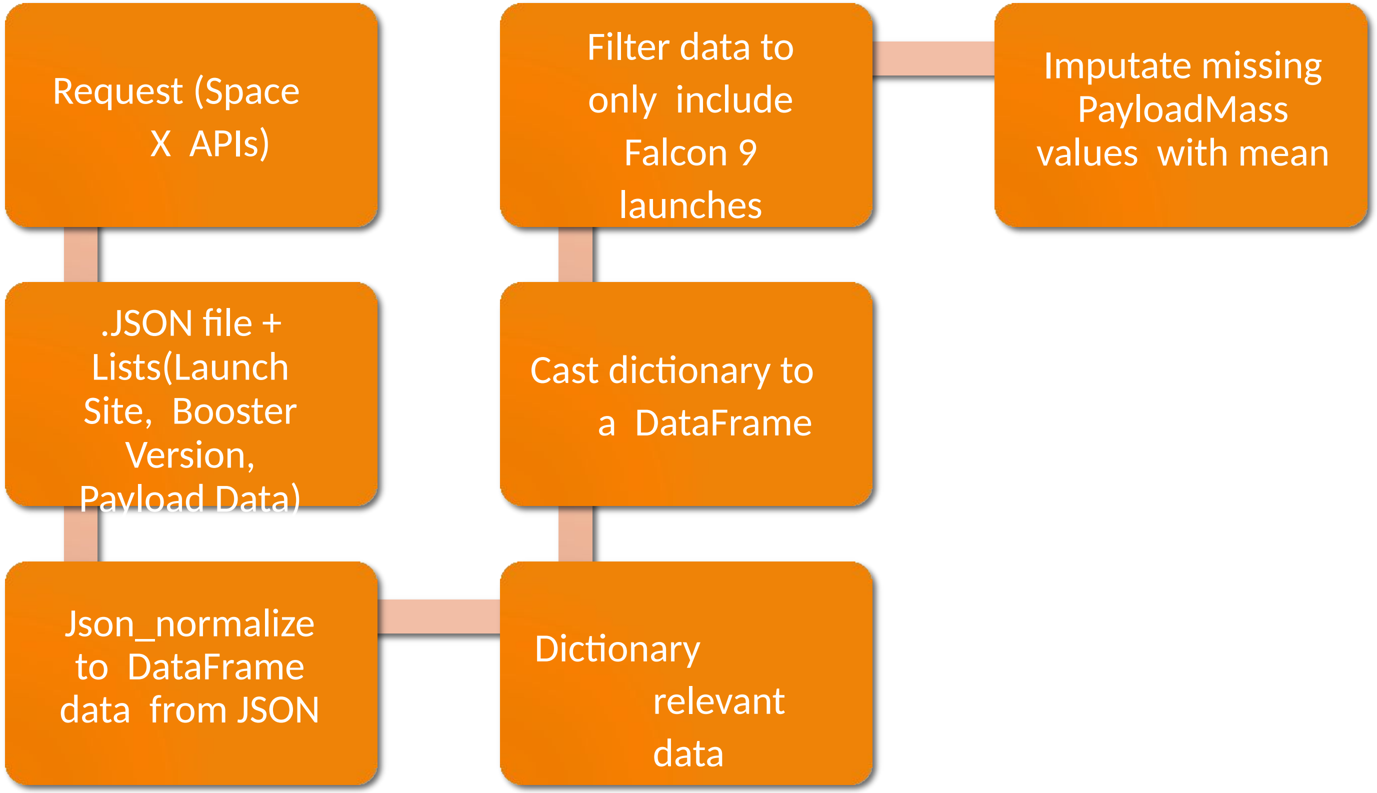
- The dataset obtained from the SpaceX API includes the following columns:
  - FlightNumber
  - Date
  - BoosterVersion
  - PayloadMass
  - Orbit
  - LaunchSite
  - Outcome
  - Flights
  - GridFins
  - Reused
  - Legs
  - LandingPad
  - Block
  - ReusedCount
  - Serial
  - Longitude
  - Latitude

Wikipedia Web Scraped Data Columns:

- The dataset obtained through web scraping from SpaceX's Wikipedia page consists of the following columns:
  - Flight No.
  - Launch site
  - Payload
  - PayloadMass
  - Orbit
  - Customer
  - Launch outcome
  - Version of the Booster
  - Booster landing
  - Date
  - Time

The upcoming slides will display flowcharts illustrating the data collection process from both the SpaceX API and the web scraping procedure.

Data Collection – SpaceX API



Request (Space X APIs)

.JSON file + Lists(Launch Site, Booster Version, Payload Data)

Json_normalize to DataFrame data from JSON

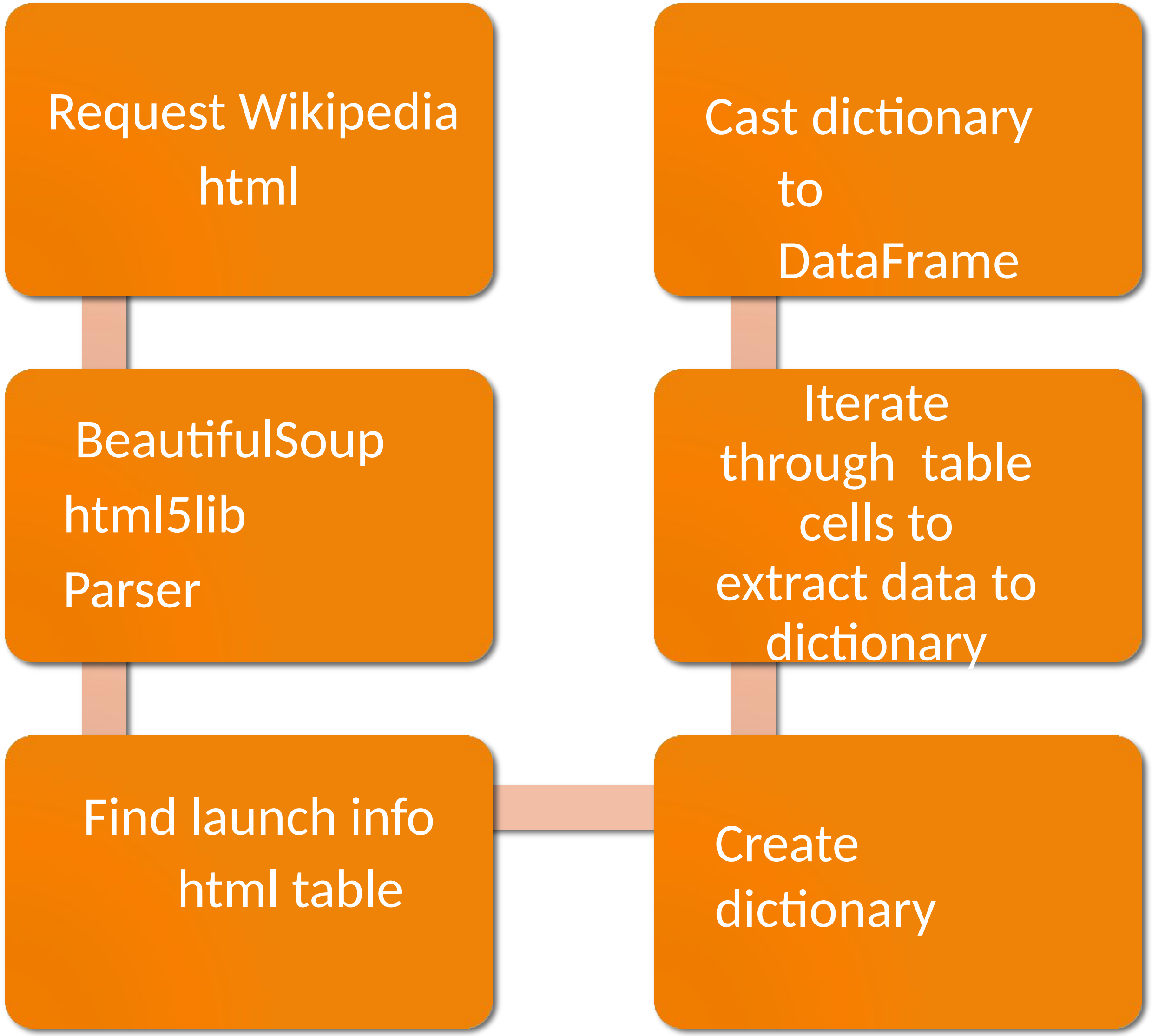
Dictionary relevant data

Cast dictionary to a DataFrame

Filter data to only include Falcon 9 launches

Imputate missing PayloadMass values with mean

Data Collection – Web Scraping



Request Wikipedia html

BeautifulSoup html5lib Parser

Find launch info html table

Create dictionary

Iterate through table cells to extract data to dictionary

Cast dictionary to DataFrame

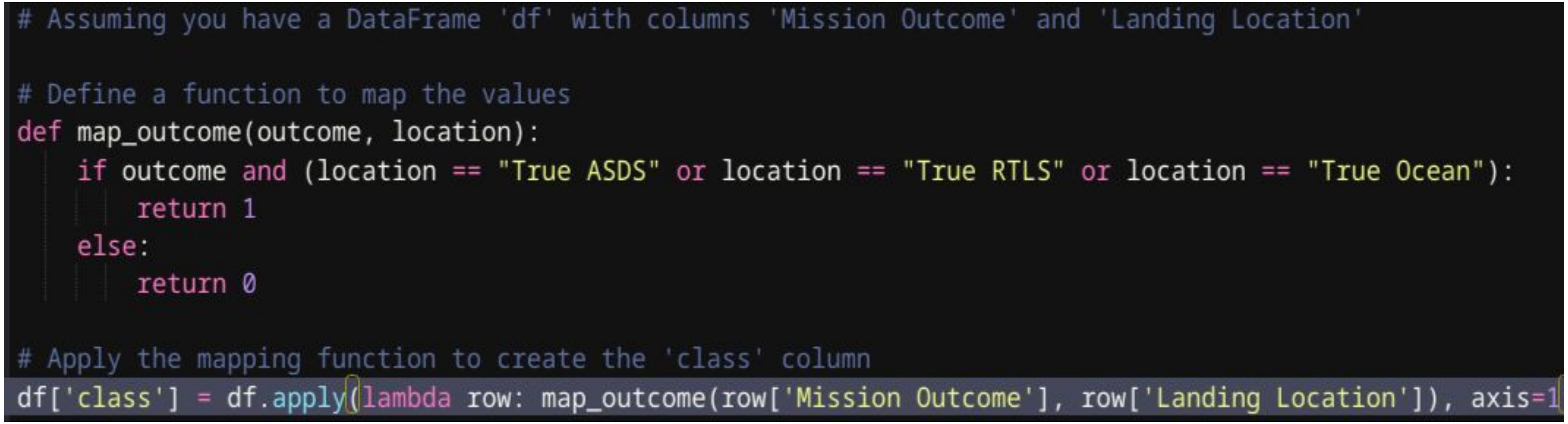
Data Wrangling

o create a new training label column 'class' based on the 'Mission Outcome' and 'Landing Location' columns with the specified value mapping, you can follow these steps:

Evaluate the 'Mission Outcome' and 'Landing Location' columns to determine the outcome of the landing.
Apply the following value mapping rules:

- If 'Mission Outcome' is True and 'Landing Location' is True for ASDS, True for RTLS, or True for Ocean, set 'class' to 1.
- If 'Mission Outcome' is None and 'Landing Location' is None, False for ASDS, None for ASDS, False for Ocean, or False for RTLS, set 'class' to 0.

Here's a Python-like pseudocode snippet to implement this logic:



```python
# Assuming you have a DataFrame 'df' with columns 'Mission Outcome' and 'Landing Location'

# Define a function to map the values
def map_outcome(outcome, location):
    if outcome and (location == "True ASDS" or location == "True RTLS" or location == "True Ocean"):
        return 1
    else:
        return 0

# Apply the mapping function to create the 'class' column
df['class'] = df.apply(lambda row: map_outcome(row['Mission Outcome'], row['Landing Location']), axis=1
```

EDA with Data Visualization

Exploratory Data Analysis (EDA) was conducted on several key variables, including Flight Number, Payload Mass, Launch Site, Orbit, Class, and Year, to gain insights and identify potential relationships. Various types of plots were utilized to visualize these relationships and assess their relevance for training the machine learning model. The following plots were generated:

Flight Number vs. Payload Mass: A scatter plot was used to examine the relationship between flight number and payload mass.
Flight Number vs. Launch Site: Another scatter plot was employed to analyze how flight numbers corresponded to different launch sites.
Payload Mass vs. Launch Site: This plot explored the relationship between payload mass and launch site using a scatter plot.
Orbit vs. Success Rate: A bar plot was used to visualize the success rate of different orbit types.
Flight Number vs. Orbit: A scatter plot illustrated the relationship between flight numbers and the chosen orbit.
Payload vs. Orbit: This scatter plot examined the connection between payload mass and orbit selection.
Success Yearly Trend: A line chart depicted the yearly trend in mission success, providing insights into any patterns or trends over time.

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 take center stage in our analysis, offering visualizations of launch sites, successful and unsuccessful landings, and their proximity to critical locations such as railways, highways, coasts, and cities. These maps yield insights into launch site selection strategies and geographical factors influencing successful landings.

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.

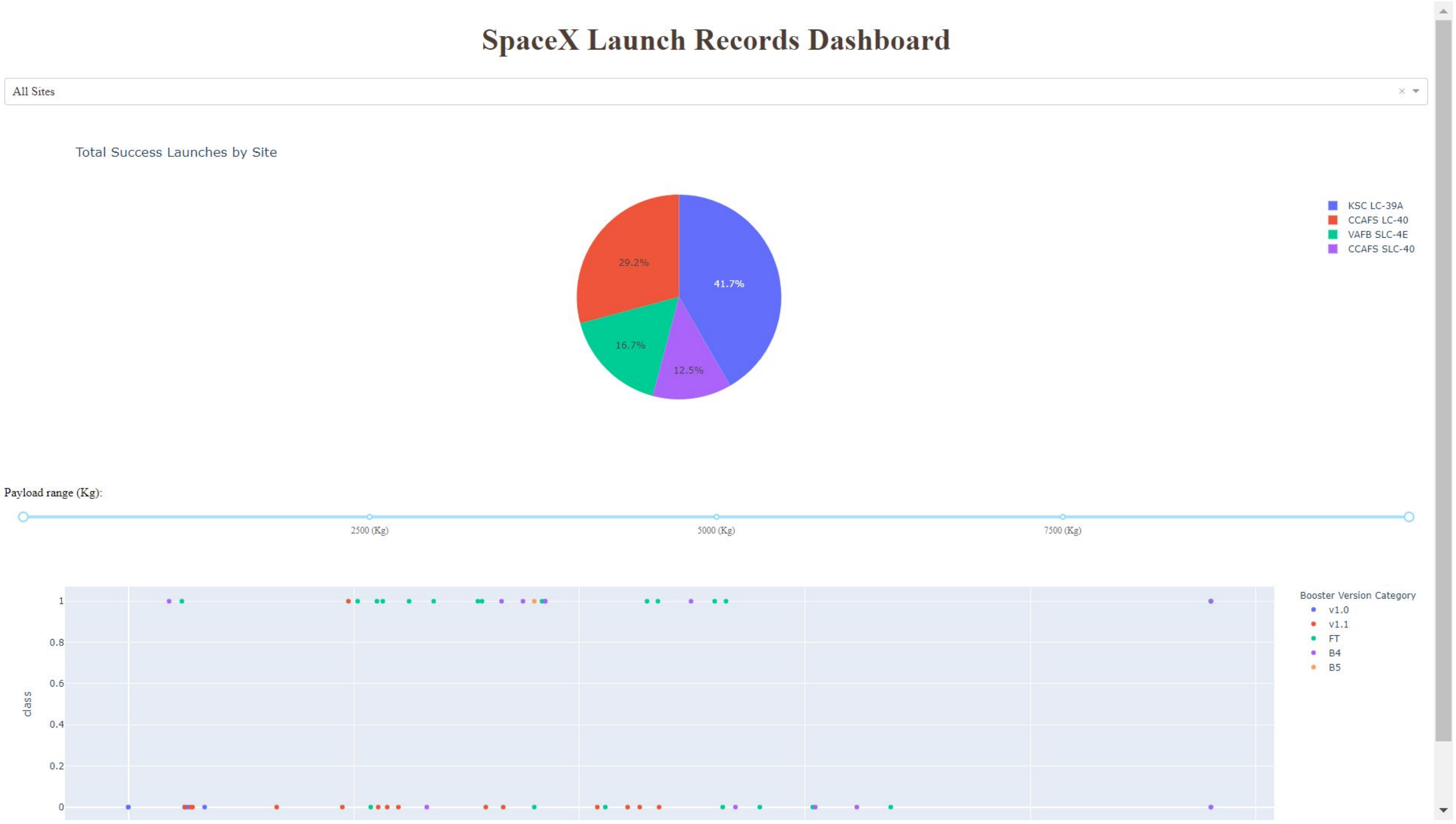
GitHub url:
https://github.com/navassherif98/IBM_Data_Science_Professional_Certification/blob/master/10.Applied_Data_Science_Capstone/Week%203%20Interactive%20Visual%20Analytics%20and%20Dashboard/spacex_dash_app.py

Predictive analysis (Classification)

1. Split label column from the dataset.
2. Split the dataset into a training set and a test set for all models.
3. Use GridSearchCV with cross-validation (cv=10) to find optimal parameters.
4. Fit and transform the data on Logistic Regression, SVM, Decision Tree, and Standard Scaler for KNN models.
5. Calculate confusion matrices for all models using the 'Class' column from the dataset.
6. Score the models based on their performance.
7. Create a barplot to compare the scores of the models.

Results

- Preview of the Plotly dashboard.
- Presentation of the results of Exploratory Data Analysis (EDA) with visualizations.
- Presentation of the results of EDA with SQL.
- Showcase of the Interactive Map with Folium.
- Presentation of the results of our model with an accuracy of about 83%.

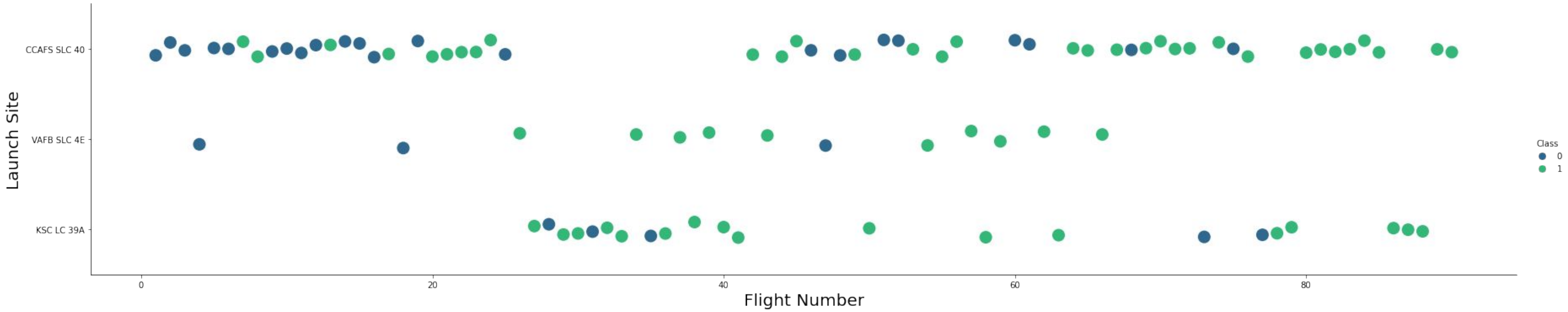


SpaceX Launch Records Dashboard

All Sites

Payload range (Kg):

Flight Number vs. Launch Site

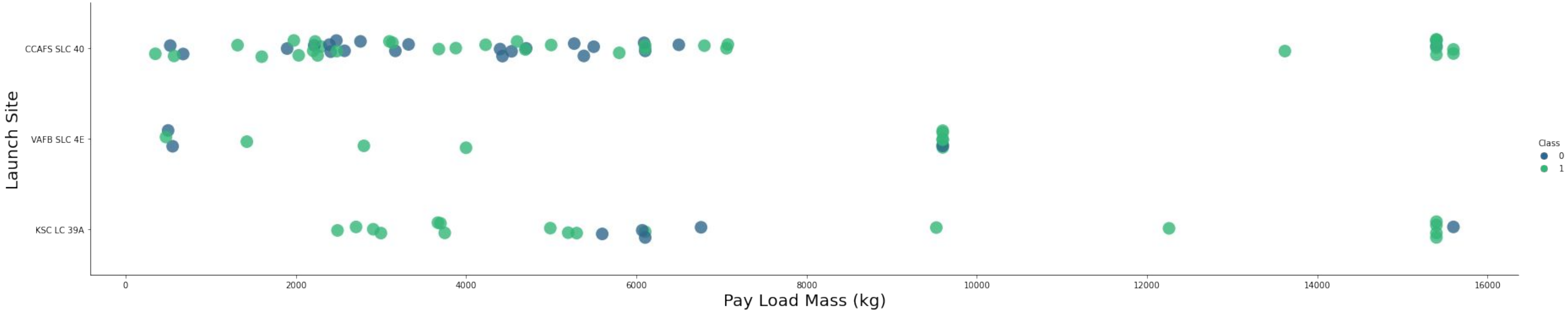


Green indicates successful launch; Purple indicates unsuccessful launch.

It appears from the graphic that there is a noticeable increase in the success rate over time, as indicated by the Flight Number. There seems to be a significant breakthrough in success rate around flight 20, which could signify a pivotal moment in the company's history.

Additionally, the graphic suggests that Cape Canaveral Air Force Station (CCAFS) is the primary launch site, as it has the highest volume of launches. This observation underscores the importance of CCAFS in SpaceX's operations and launch strategy.

Payload vs. Launch Site

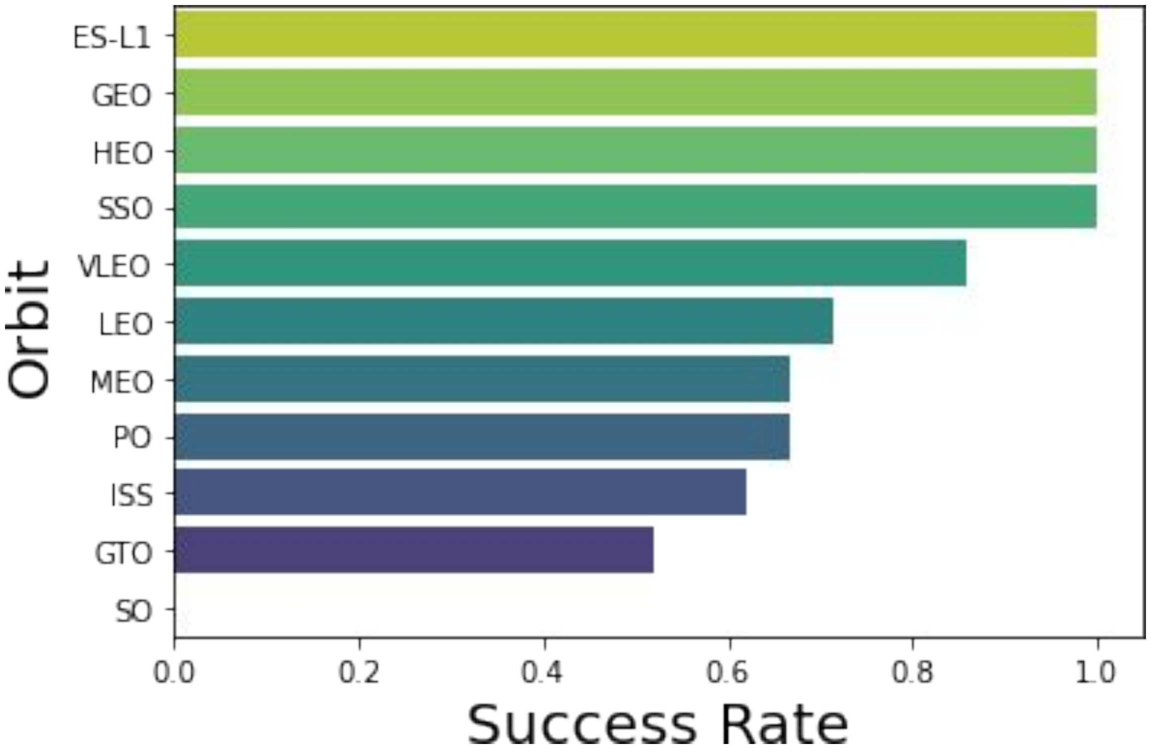


Green indicates successful launch; Purple indicates unsuccessful launch.

It's evident from the data that the payload mass typically falls within the range of 0-6000 kg for SpaceX missions. This range signifies a common payload capacity for their launches.

Furthermore, the observation that different launch sites utilize varying payload masses underscores the adaptability of SpaceX's launch strategy. The choice of launch site may be influenced by factors such as payload size and destination orbit, and this flexibility allows SpaceX to optimize their launch operations for different mission requirements.

Success Rate vs. Orbit Type



Success Rate Scale
with 0 as 0%
0.6 as
60% 1 as
100%

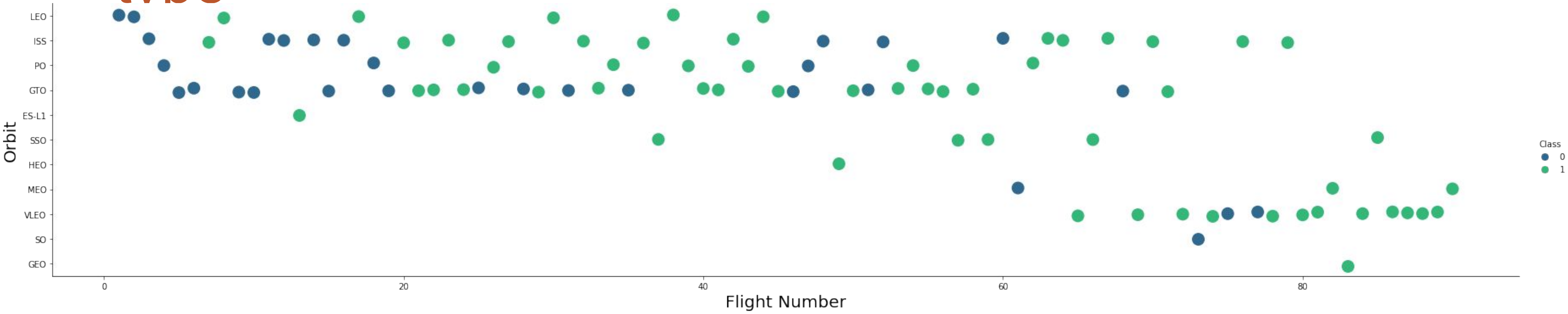
ES-L1 (1), GEO (1), HEO (1) have 100% success rate (sample sizes in parenthesis) SSO (5) has 100% success rate

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

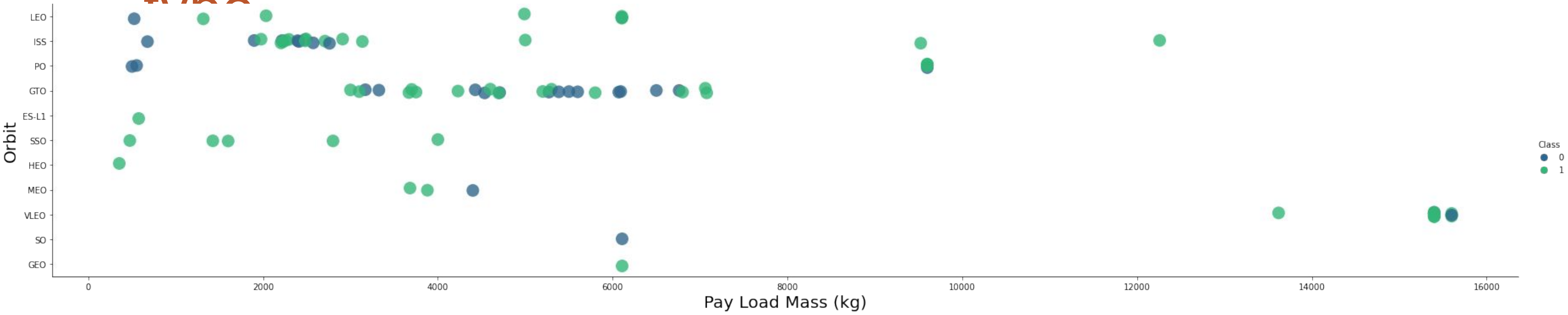
Flight Number vs. Orbit type



Green indicates successful launch; Purple indicates unsuccessful launch.

The data suggests that SpaceX's launch orbit preferences have shifted over time, with an initial focus on Low Earth Orbit (LEO) missions and a recent shift towards Very Low Earth Orbit (VLEO). This transition appears to correlate with changes in launch outcomes. Notably, SpaceX tends to achieve higher success rates in lower orbits, such as LEO and Sun-synchronous orbits, indicating their effectiveness in these specific mission profiles.

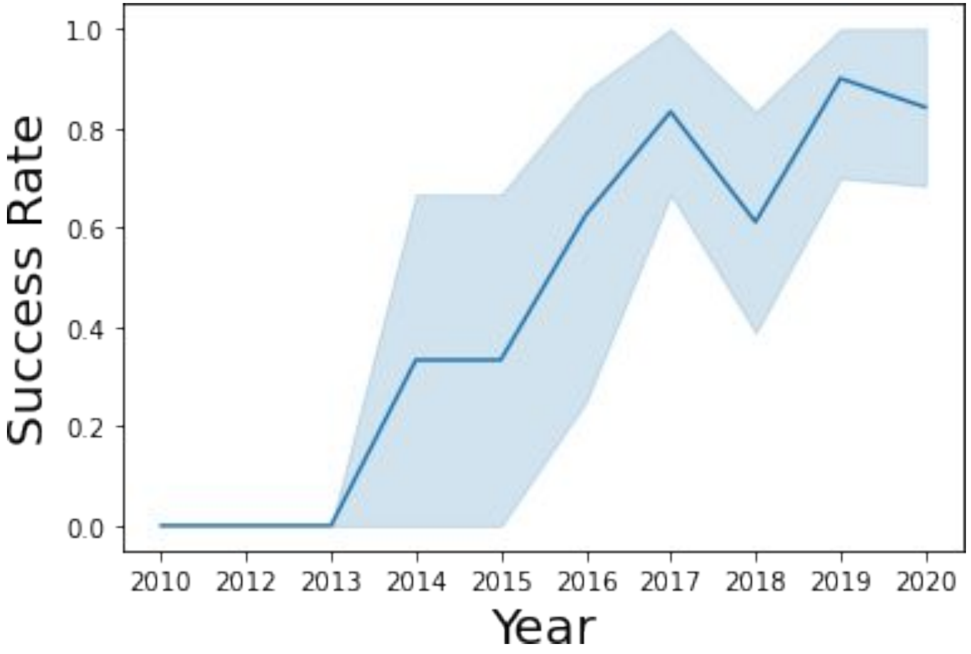
Payload vs. Orbit type



Green indicates successful launch; Purple indicates unsuccessful launch.

The data analysis suggests a correlation between payload mass and launch orbit. Specifically, Low Earth Orbit (LEO) and Sun-synchronous orbit (SSO) missions tend to have relatively lower payload masses. In contrast, the Very Low Earth Orbit (VLEO), which is one of the most successful orbits, consistently features payload mass values at the higher end of the range. This correlation highlights how SpaceX tailors payload mass to suit the specific requirements and challenges associated with each orbit, optimizing the chances of mission success.

Launch Success Yearly Trend



95% confidence interval (light blue shading)

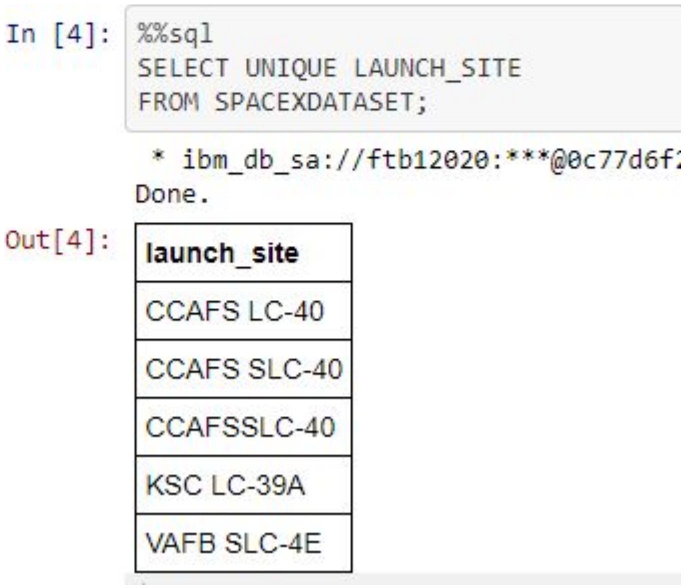
The analysis of the data indicates a general trend of increasing success rates over time since 2013, with a minor dip observed in 2018. In recent years, the success rate has stabilized at around 80%. This trend reflects SpaceX's continuous improvement in mission success, with the occasional fluctuation that is common in the aerospace industry. The consistent 80% success rate in recent years highlights SpaceX's reliability and effectiveness in space missions.

EDA with SQL

EXPLORATORY DATA ANALYSIS WITH SQL DB2

INTEGRATED IN PYTHON WITH SQLALCHEMY

All Launch Site Names



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

         * ibm_db_sa://ftb12020:***@0c77d6f
        Done.
Out[4]:
```

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

Query unique launch site names from database.

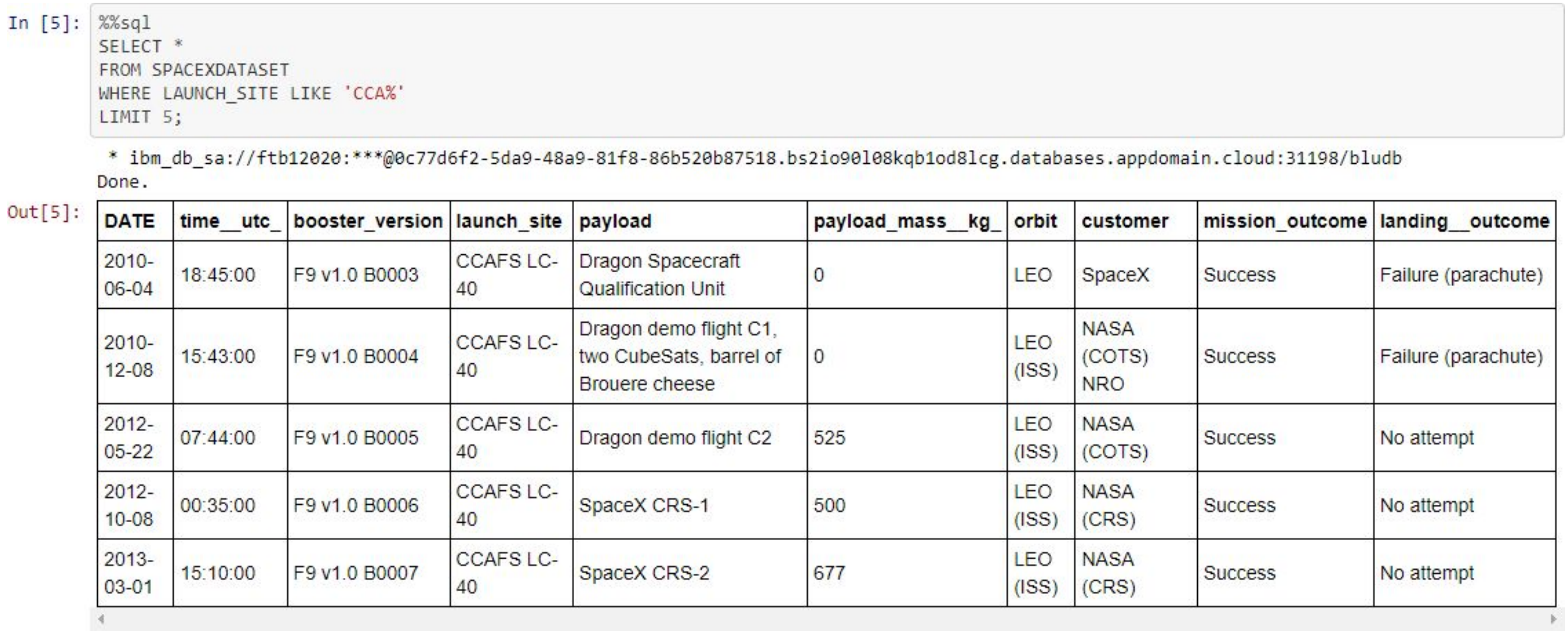
CCAFS SLC-40 and CCAFSSLC-40 likely all represent the same
launch site with data entry errors.
CCAFS LC-40 was the previous name.

Likely only 3 unique launch_site

values: CCAFS SLC-40, KSC LC-39A,

VAFB SLC-4E

Launch Site Names Beginning with `CCA`



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

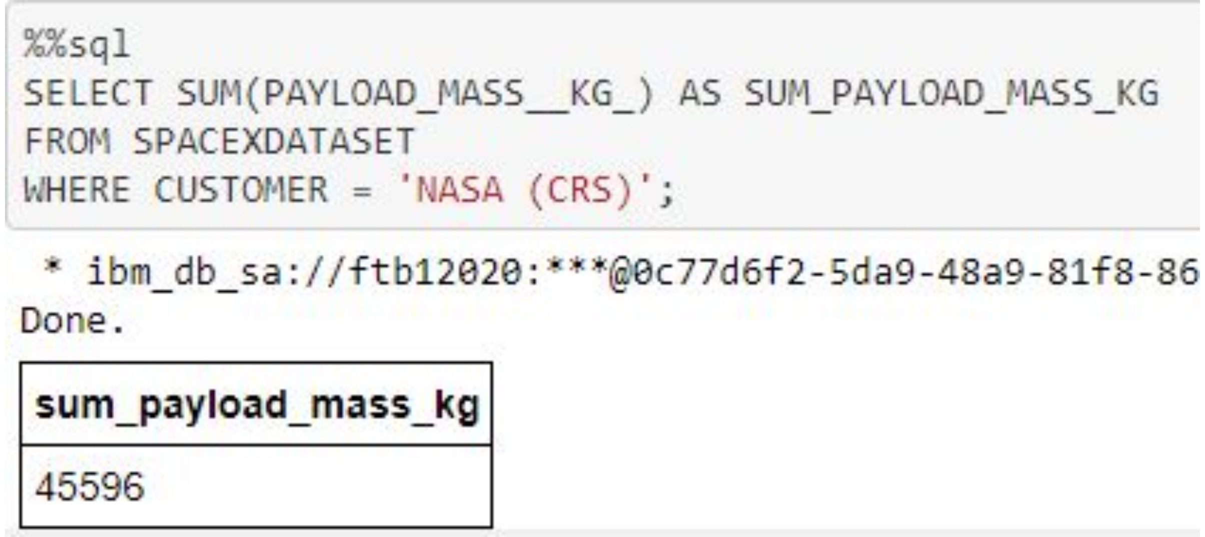
```
 * ibm_db_sa://ftb12020:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/bludb
Done.
```

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

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

Total Payload Mass from NASA



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

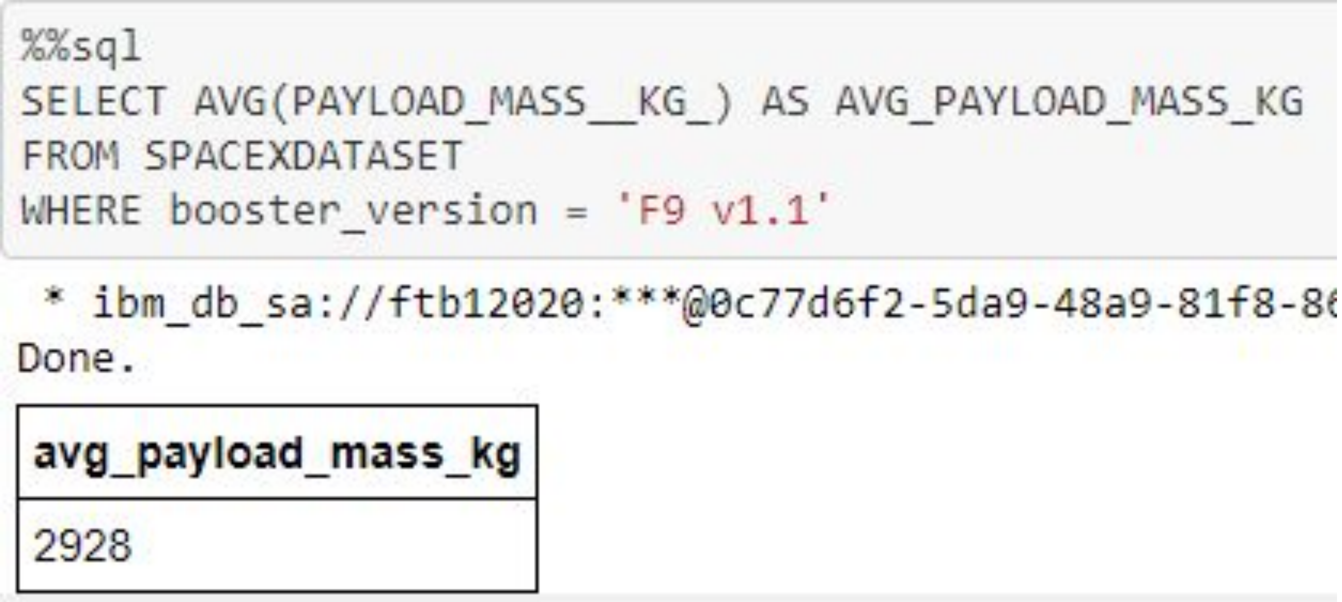
```
 * ibm_db_sa://ftb12020:***@0c77d6f2-5da9-48a9-81f8-86
Done.
```

sum_payload_mass_kg
45596

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).

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'
```

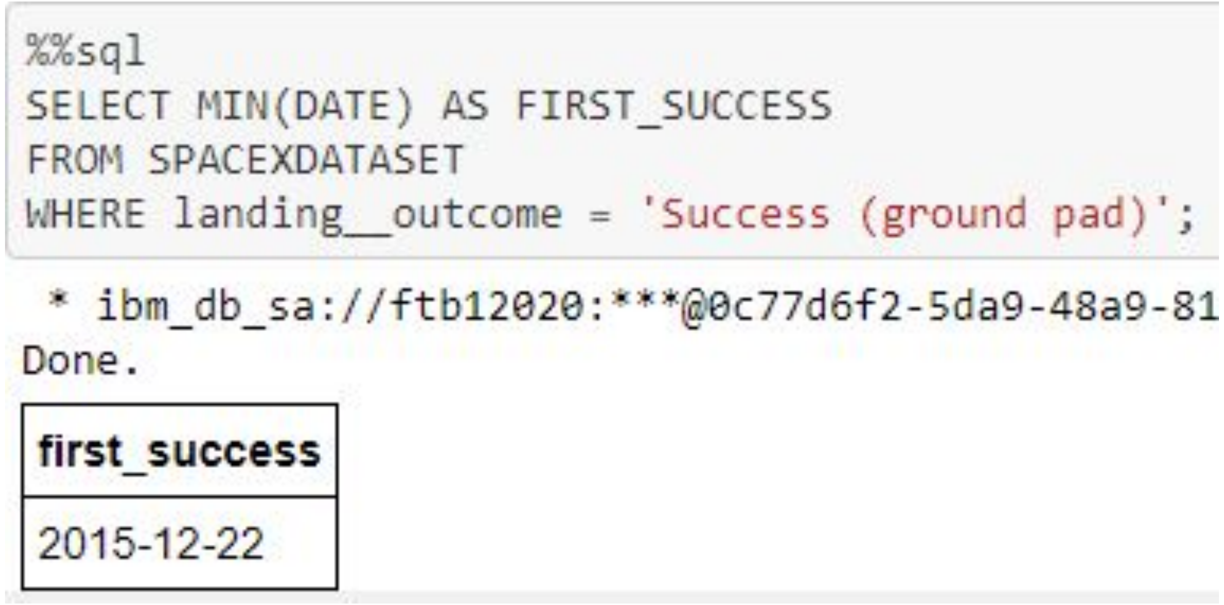
```
 * ibm_db_sa://ftb12020:***@0c77d6f2-5da9-48a9-81f8-86
Done.
```

avg_payload_mass_kg
2928

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

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

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

This query returns the first successful ground pad landing date.

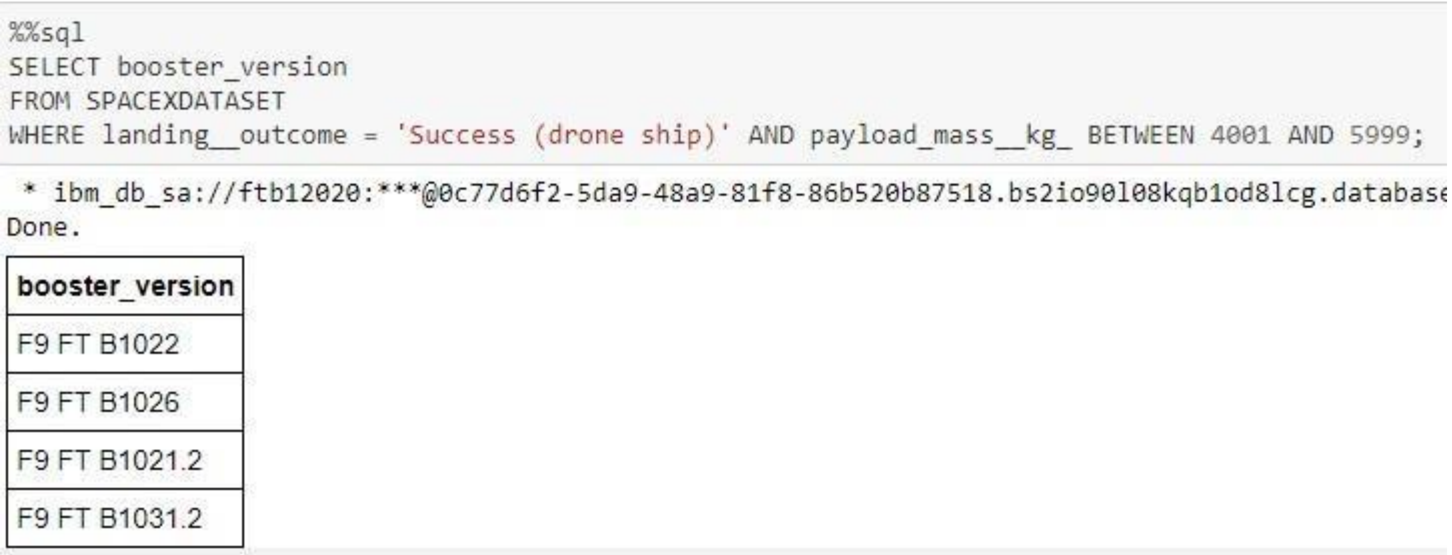
First ground pad landing wasn’t

until the end of 2015.

Successful landings in general

appear starting 2014.

Successful Drone Ship Landing with Payload Between 4000 and 6000



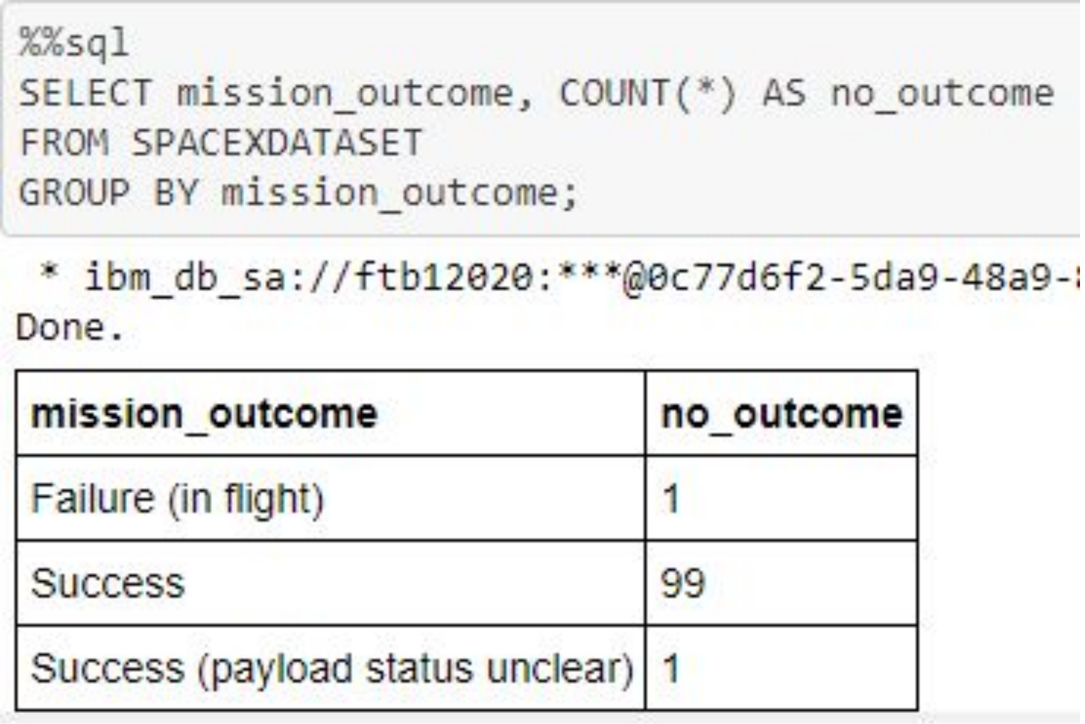
```
%%sql
SELECT booster_version
FROM SPACEXDATASET
WHERE landing__outcome = 'Success (drone ship)' AND payload_mass__kg_ BETWEEN 4001 AND 5999;
```

```
 * 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

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

Total Number of Each Mission Outcome



```
%%sql
SELECT mission_outcome, COUNT(*) AS no_outcome
FROM SPACEXDATASET
GROUP BY mission_outcome;
```

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

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

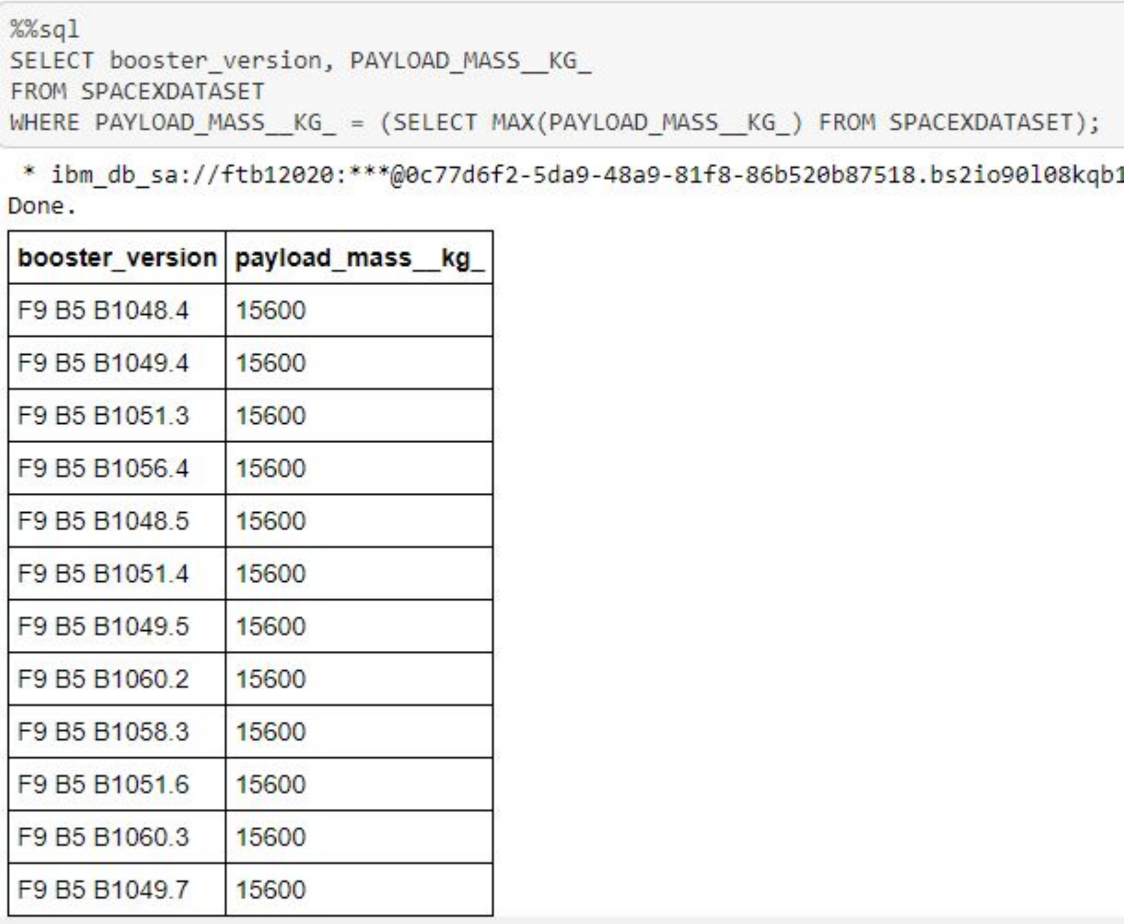
This query returns a count of each mission outcome.

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 that Carried Maximum Payload



```
%%sql
SELECT booster_version, PAYLOAD_MASS__KG_
FROM SPACEXDATASET
WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXDATASET);
```

```
 * ibm_db_sa://ftb12020:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1
Done.
```

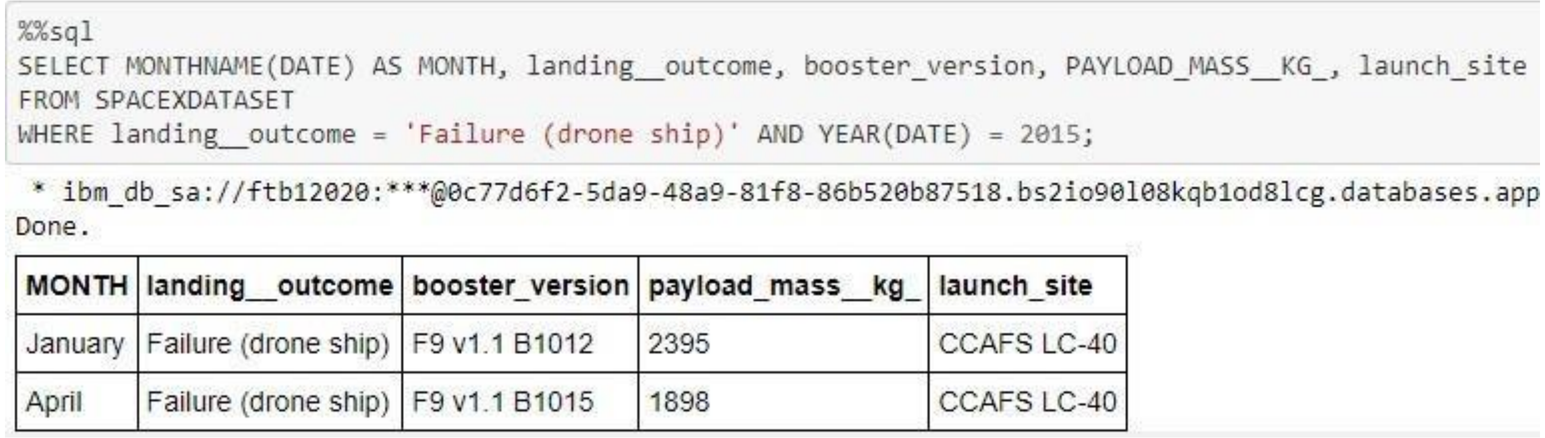
booster_version	payload_mass__kg_
F9 B5 B1048.4	15600
F9 B5 B1049.4	15600
F9 B5 B1051.3	15600
F9 B5 B1056.4	15600
F9 B5 B1048.5	15600
F9 B5 B1051.4	15600
F9 B5 B1049.5	15600
F9 B5 B1060.2	15600
F9 B5 B1058.3	15600
F9 B5 B1051.6	15600
F9 B5 B1060.3	15600
F9 B5 B1049.7	15600

This query returns 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.

This likely indicates payload mass correlates with the booster version that is used.

2015 Failed Drone Ship Landing Records



```
%%sql
SELECT MONTHNAME(DATE) AS MONTH, landing__outcome, booster_version, PAYLOAD_MASS__KG_, launch_site
FROM SPACEXDATASET
WHERE landing__outcome = 'Failure (drone ship)' AND YEAR(DATE) = 2015;
```

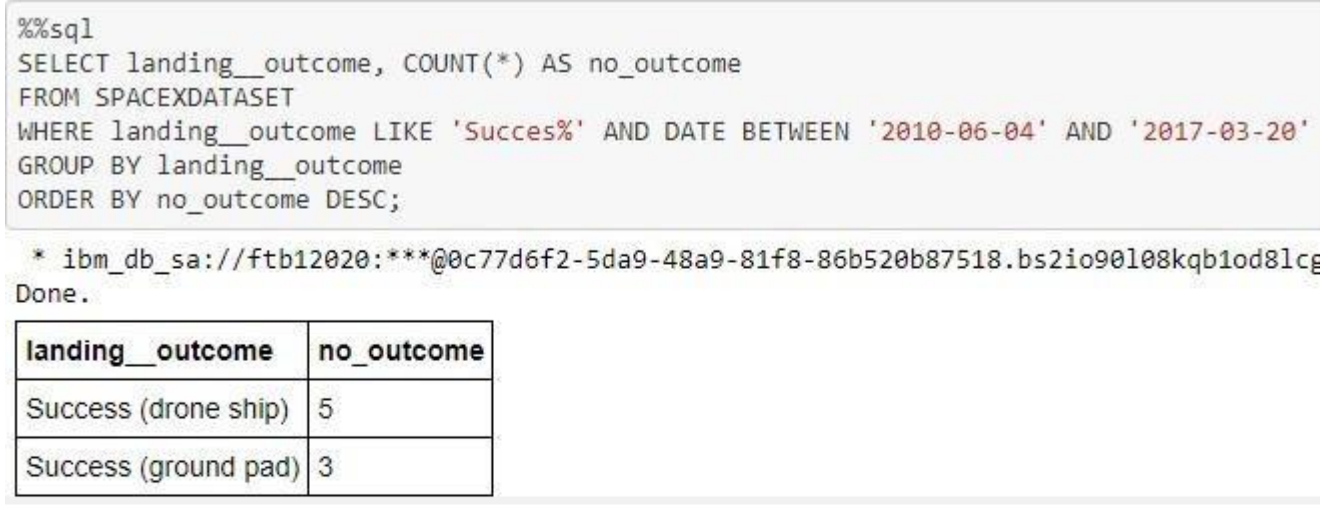
```
 * ibm_db_sa://ftb12020:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.app
Done.
```

MONTH	landing__outcome	booster_version	payload_mass__kg_	launch_site
January	Failure (drone ship)	F9 v1.1 B1012	2395	CCAFS LC-40
April	Failure (drone ship)	F9 v1.1 B1015	1898	CCAFS LC-40

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.

Ranking Counts of Successful Landings Between 2010-06-04 & 2017-03-20



```
%%sql
SELECT landing__outcome, COUNT(*) AS no_outcome
FROM SPACEXDATASET
WHERE landing__outcome LIKE 'Succes%' AND DATE BETWEEN '2010-06-04' AND '2017-03-20'
GROUP BY landing__outcome
ORDER BY no_outcome DESC;
```

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

landing__outcome	no_outcome
Success (drone ship)	5
Success (ground pad)	3

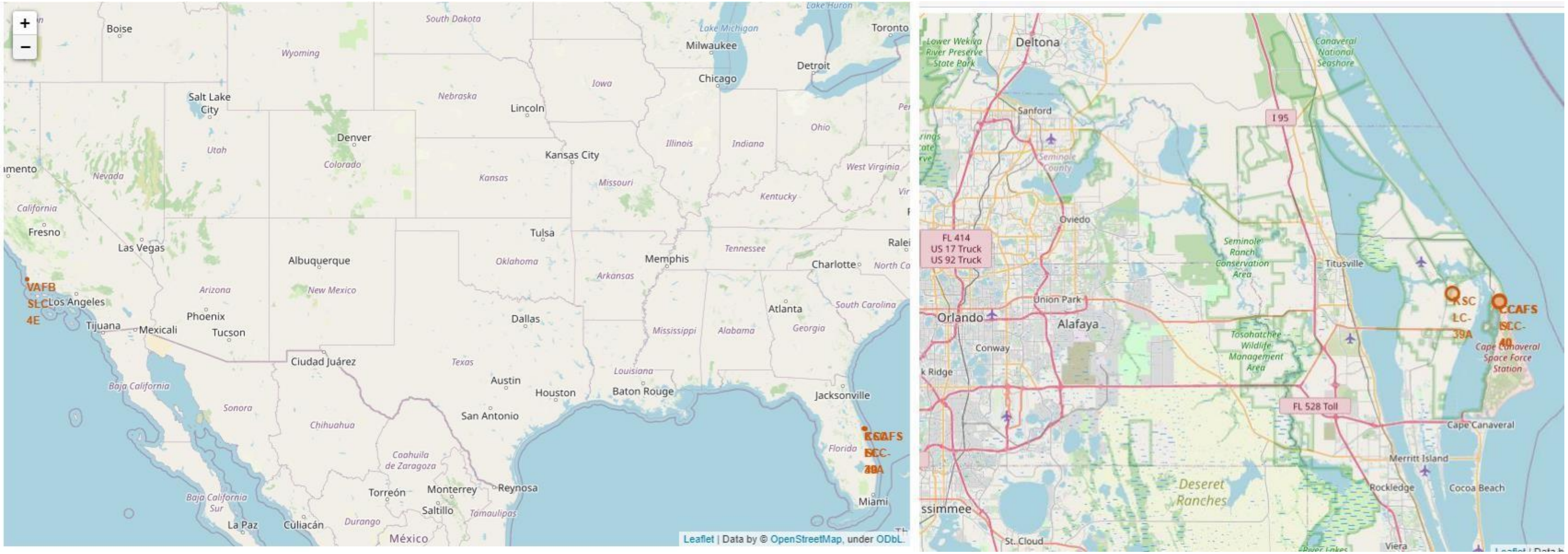
This query returns a list of successful landings and between 2010-06-04 and 2017-03-20 inclusively.

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

There were 8 successful landings in total during this time period

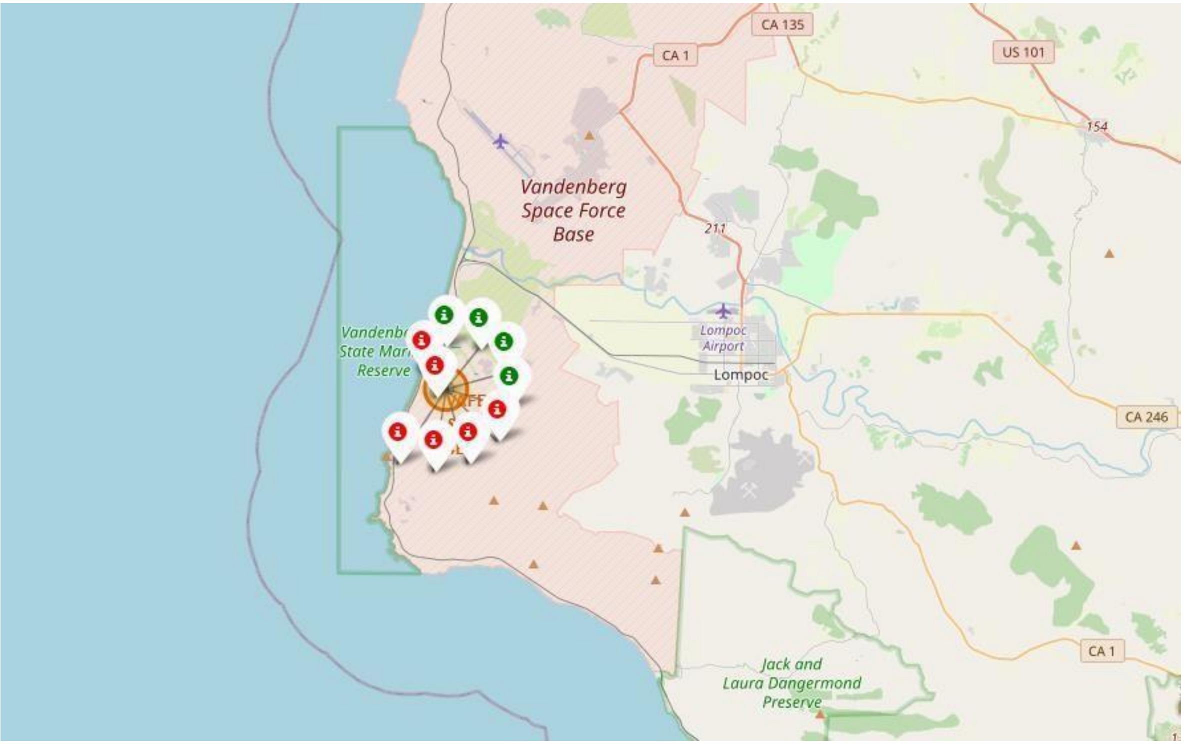
Interactive Map with Folium

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.

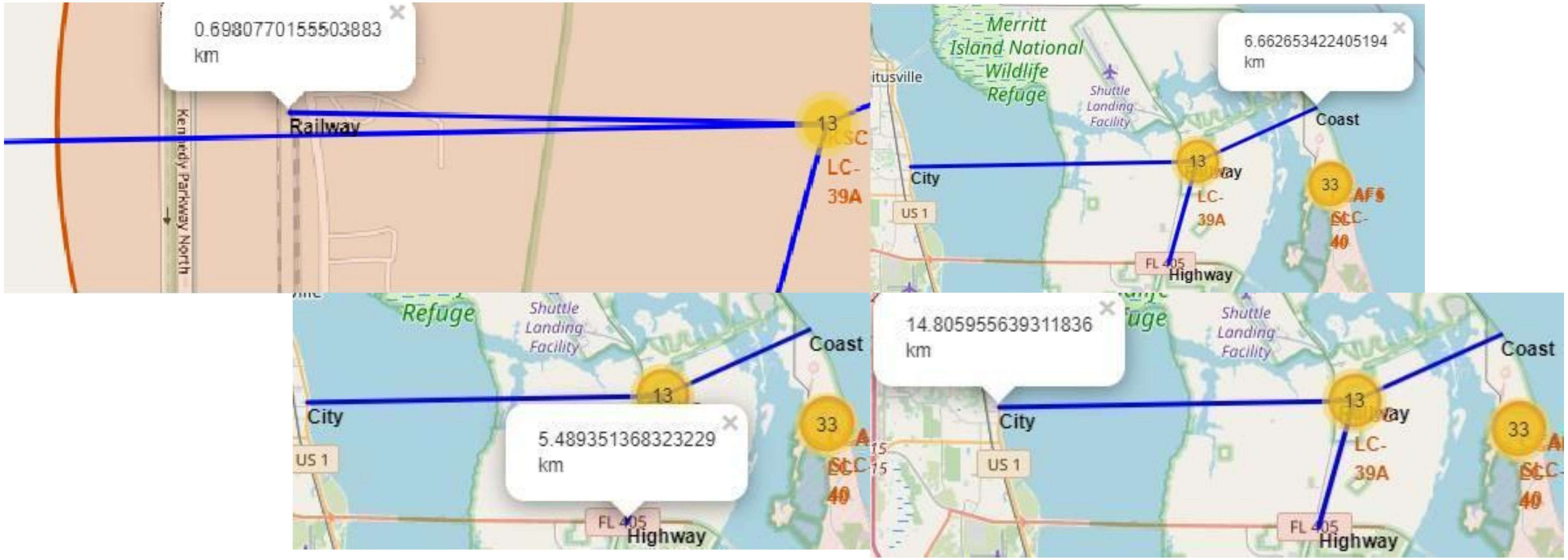
Color-Coded Launch Markers



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.

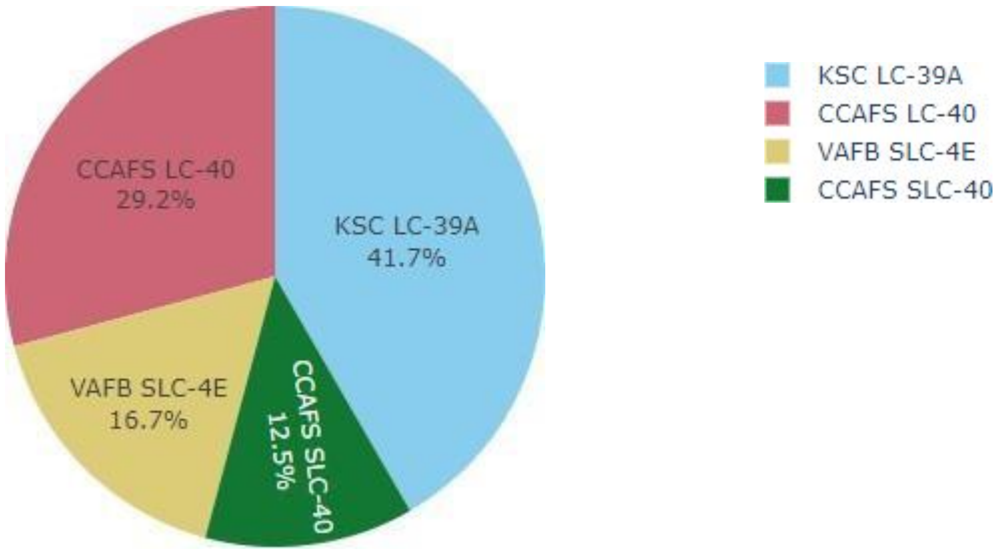
Key Location Proximities



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.

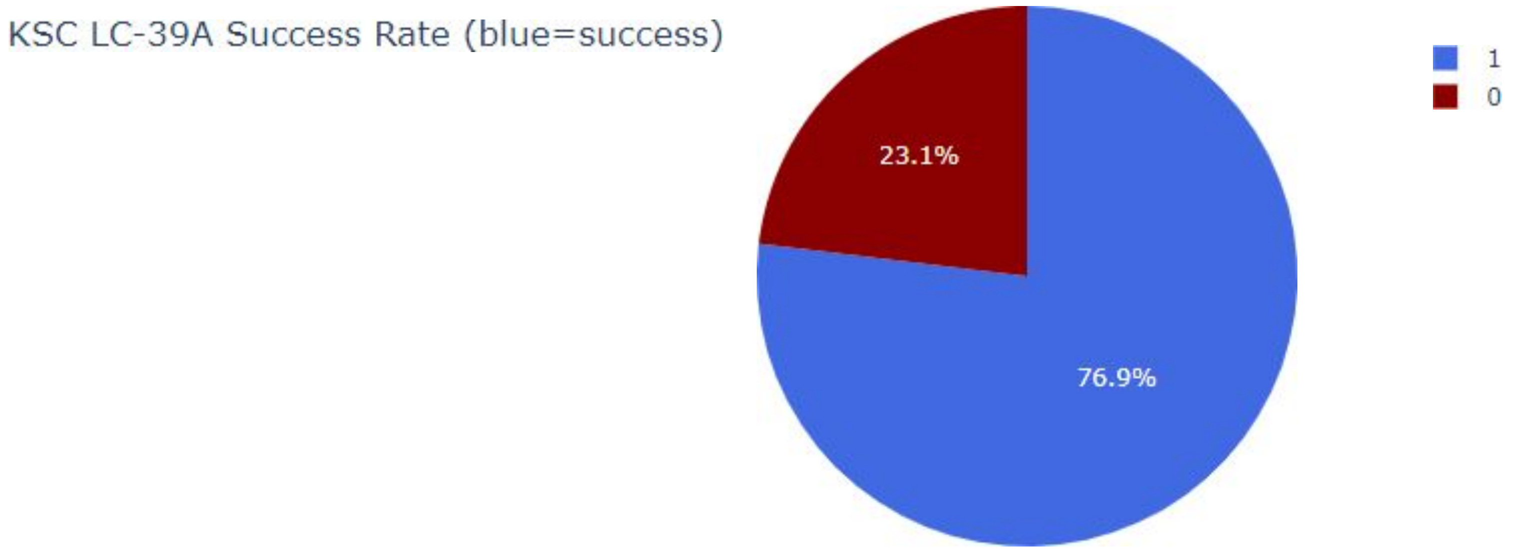
Build a Dashboard with Plotly Dash

Successful Launches Across Launch Sites



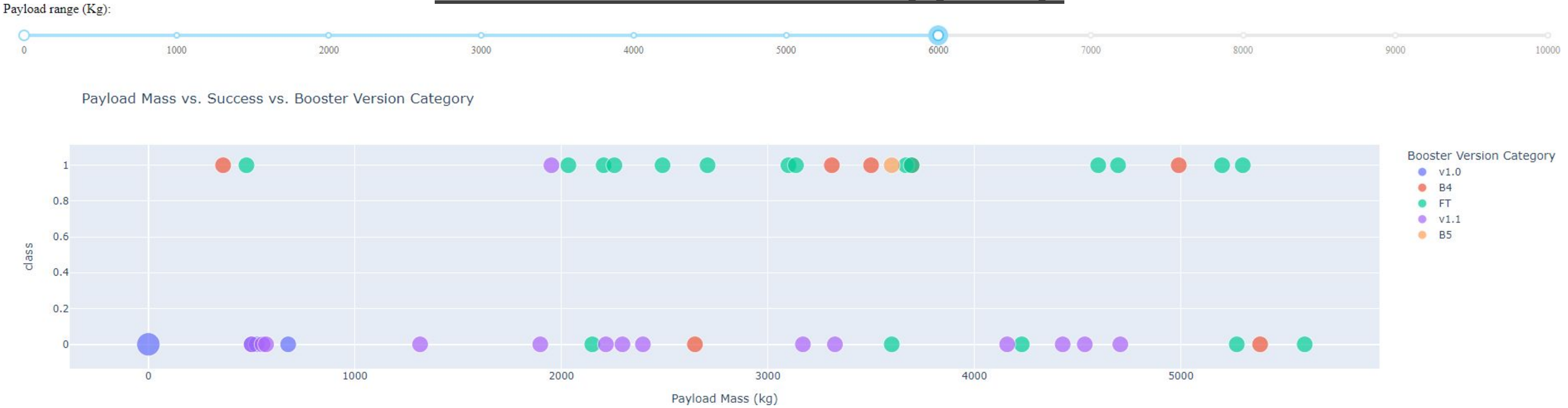
This is the distribution of successful landings across all launch sites. CCAFS LC-40 is the old name of CCAFS SLC-40 so CCAFS and KSC have the same amount of successful landings, but a majority of the successful landings where performed before the name change. VAFB has the smallest share of successful 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.

Payload Mass vs. Success vs. Booster Version Category

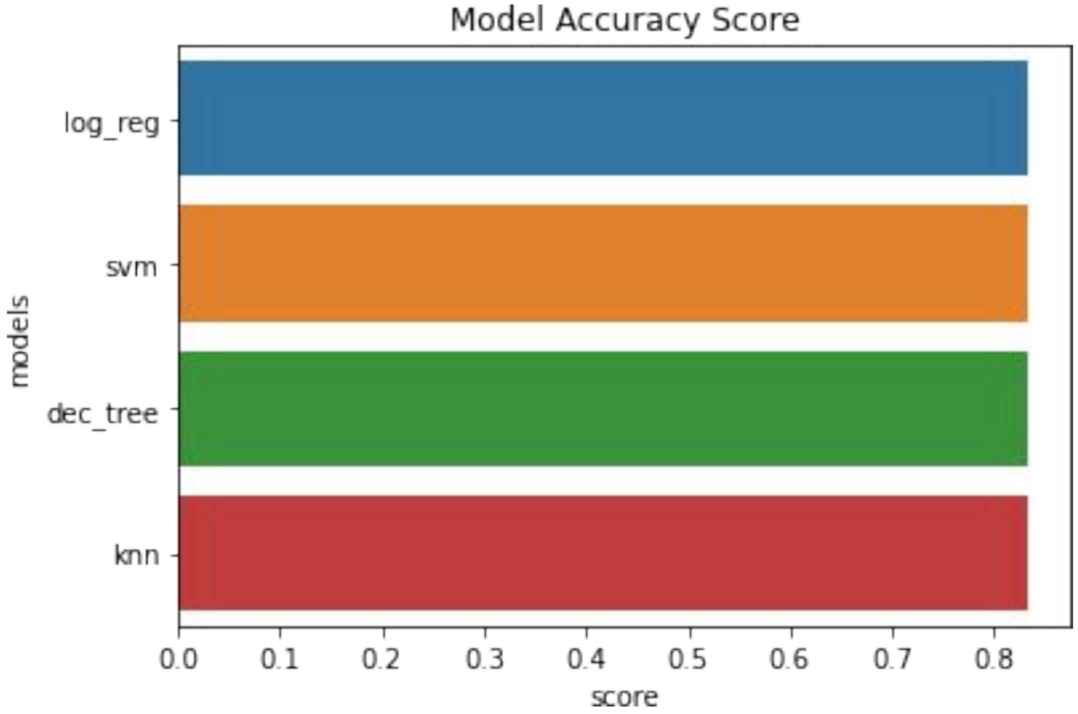


In the Plotly dashboard, there's an issue with the payload range selector, set at 0-10000 kg instead of considering the actual maximum payload of 15600 kg. The scatter plot uses color coding for booster versions and point size for the number of launches. Interestingly, within the 0-6000 kg payload range, two failed landings are observed with payloads of zero kilograms. This highlights the need for accurate data representation and range selection in the dashboard.

Predictive Analysis (Classification)

GRIDSEARCHCV(CV=10) ON LOGISTIC REGRESSION, SVM, DECISION TREE,AND KNN

Classification Accuracy



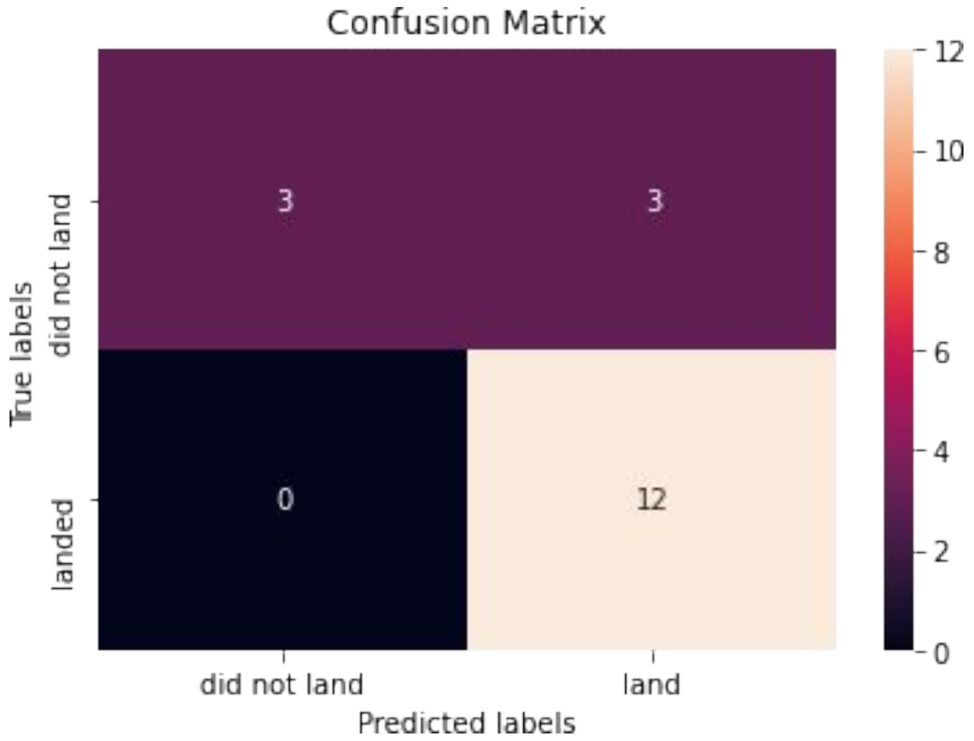
All models had virtually the same accuracy on the test set at 83.33%

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



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

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.

CONCLUSION

Our task was to develop a machine learning model for Space Y, which aims to compete with SpaceX by predicting successful Stage 1 landings and potentially saving around $100 million USD per launch. To achieve this, we utilized data from a public SpaceX API and conducted web scraping on SpaceX's Wikipedia page. We created data labels and stored the collected data in a DB2 SQL database. A user-friendly dashboard was developed for data visualization.

Our machine learning model was successfully created with an accuracy rate of 83%. This model can be invaluable to Space Y, specifically Allon Mask, as it enables relatively high-accuracy predictions of whether a launch will result in a successful Stage 1 landing before the launch decision is made. However, it's important to note that further data collection is recommended to refine the model and enhance its accuracy, ensuring it can make even more precise predictions and support Space Y's competitive efforts in the commercial space industry.