Winning the Space Race with Data Science

IBM Data Science Professional Certificate Capstone Project

Presented by Houndboy
04/03/2023

https://en.wikipedia.org/wiki/List_of_Falcon_9_and_Falcon_Heavy_launches

Outline

- Executive Summary
- Introduction
- Methodology (Section 1)
- Results (Sections 2 – 5)
- Conclusions (Section 6)
- Appendices (Section 7)

Falcon 9 successful first stage landing

https://github.com/houndboy/IBM-Data-Science-Professional-Certificate-Program/blob/main/C10.Capstone.lab2.data-collection.webscraping.ipynb

Executive Summary: Methodologies

- Data Collection: Data requests (SpaceX API), Webscraping (Wikipedia)
- Data wrangling: converting landing outcomes into training labels
- Exploring 'SpaceX DataSet' with SQL (sqlite)
- Data visualization and Feature Engineering (Pandas, Matplotlib)
- Geospatial analysis of select United States launch sites (folium)
- Interactive visual analytics of SpaceX launch data (Plotly, Dash)
- Modify data, standardize data, split it into training data and test data
- Perform predictive analysis (supervised machine learning)

Executive Summary: Results

- Collected necessary roclet and launch data for analysis
- Prepared data for later predictive analysis – feature selection and one hot encoding
- Explored data to determine variables and relationships between variables impacting first stage landing outcome
- Visualized launch site location and launch success and launch failure data per launch site
- Created a Plotly Dash appilication enabling the user to visually explore SpaceX launch data in real time
- Determined the best model and parameters to predict a a successful first stage landing from Logistic Regression, KNN, SVM, and Decision Tree methodologies

Introduction: Goal, Context, and Inquiry (slide 1)

Goal:

The purpose of this project is to determine which supervised machine learning algorithm is best in predicting a successful landing of the first stage of a Falcon 9 launch using data obtained from the SpaceX API and from Wikipedia's SpaceX entry.

Project Context:

SpaceX advertises Falcon 9 rocket launches on its website with a cost of 62 million dollars; other providers cost upward of 165 million dollars each, much of the savings is because SpaceX can reuse the first stage. Therefore, if the first stage lands successfully, the cost of a launch can be determined. This information is beneficial to SpaceX in bid situations with other companies competing against SpaceX for market share in rocket launches.

Introduction: Goal, Context, and Inquiry (slide 2)

Questions to be Answered:

- What relationships exist between critical variables (e.g., flight number versus launch site, flight number versus orbit type, and payload mass versus launch site) and landing outcome?
- What is the relationship between time (in years) and mean landing outcome?
- Which variables should be used for predictive modeling (e.g., payload mass, orbit, or launch site)?
- What is known about previous Falcon 9 launches and landing outcomes (e.g., booster versions, payload masses carried, and mission outcomes)?

Introduction: Goal, Context, and Inquiry (slide 3)

Questions to be Answered (continued):

- **W**here are launch sites located and how many launches were there per launch site?
- Which launch site(s) had the most success in first stage landings?
- What is the relationship between booster version, payload mass, and landing outcome for previous launches?
- What is the best supervised machine learning algorithm (i.e., Logistic Regression, K Nearest Neighbors, Support Vector Machine Object, or Decision Tree) for predicting a successful Falcon 9 first stage landing?

Section 1

Methodology

Methodology (slide 1)

Executive Summary

- **Data collection methodology:** Data was requested from the SpaceX API and scraped from a SpaceX Wikipedia table , cleaned, and converted to a pandas DataFrame.
- **Perform data wrangling:** Mission (landing) outcomes were identified in the data and converted to training labels.
- **Perform exploratory data analysis (EDA) using visualization and SQL:** Variables were visualized as scatterplots relative to landing outcome ('class'), orbits and landing outcome ('class') was visualized as a bar plot, and mean success rate over time (in years) was visualized as a line graph. Specific information was identified (e.g., names of launch sites, payload mass, average payload mass, and date of first successful landing) using SQL.

Methodology (slide 2)

Executive Summary

- **Perform interactive visual analytics using Folium and Plotly Dash:** A geospatial analysis of launch site locations was visualized using folium; landing outcome versus payload mass per booster version was visualized as a schatter plot using Plotly Dash.
- **Perform predictive analysis using classification models:** Data was prepared during the data wrangling proces (i.e., landing outcomes converted to training labels) . The training labels were: '0' for failure and '1' for success. A column named 'class' was created for training labels as a numpy array and added to the data set. The data was then standardized and split into two data sets, one for testing and the other for training. Then, objects were created (e.g., Logistic Regression) and the object (model) was fitted to the data. Classification accuracy and parameter accuracy were calculated for each model; and a confusion matrix was visualized for each model. Finally, the 'max' function was used to find the best model relative to classification and parameter accuracy.

Data Collection Overview

Data was collected using the SpaceX API and doing web scraping of a table from Wikipedia's SpaceX page. The purpose of the data collection process was to obtain information on SpaceX Falcon 9 rockets and launches. Samples of the 'unwrangled' data sources can be found in the Appendix (slide 2).

The data collected from requests made to the SpaceX API included the following: booster version, name of launch site, launch site latitude and longitude, mass of payload, orbit, outcome of landing, type of landing, number of flights, whether gridfins were used, whether legs were used, number used to separate version of cores (block of core), number of times this specific core has been reused, and serial of core.

The data collected from web scraping a list from SpaceX's Wikipedia page ('List of Falcon 9 and Falcon 9 Heavy Launches') included the following: flight number, launch site, payload, payload mass, orbit, customer, launch outcome, booster version, booster landing, date, and time.

Data Collection – SpaceX API

Request data from SpaceX API: 'response = requests.get(spacex_url)'

Use static response object; decode the response content as json :'json_data = response.json()'

Normalize the data ; convert it to a pandas DataFrame: 'data = pd.json_normalize(json_data)'

Request launch data from API –rocket, payload, launch pad, and cores; clean, organize, and store data in lists

Comvert the lists of relevant data into a dictionary: 'launch_dict'

Retreive stored data using the 'get . . .' call; e.g., 'getCoreData(data)'

Convert launch_dict (a dictionary) into a pandas DataFrame: 'launch_df = pd.DataFrame(launch_dict)'

Filter data to include only Falcon 9 launches, reset the flight number column, and replace missing values with mean values; convert DataFrame to CSV format

https://github.com/houndboy/IBM-Data-Science-Professional-Certificate-Program/blob/main/C10.Capstone.lab1.data-collection.api.ipynb

Data Collection – Web Scraping Wikipedia

Request Falcon 9 launch page from Wikipedia using a static url: 'html_data=requests.get (static_url)'

→

Create a Beautiful Soup object from the html_data: "soup = BeautifulSoup(html_data.text, 'html parser')"

→

Extract all column names from the html table header: "html_tables = soup.find_all('table')" and 'first_launch_table = html_tables[2]'

→

Extract column names one by one using the 'find_all' function and put them in a list ('column_names')

→

Create a dictionary ('launch_dict') with keys from the 'column_names' and launch records extracted from table rows

→

After launch records were parsed into 'launch_dict', a pandas DataFrame was created: "df= pd.DataFrame('launch_dict')"

→

The pandas DataFrame ('df') was converted to a CSV file using '.to_csv' method

https://github.com/houndboy/IBM-Data-Science-Professional-Certificate-Program/blob/main/C10.Capstone.lab2.data-collection.webscraping.ipynb

Data Wrangling: SpaceX API Data

1. Import CSV with SpaceX launch data and load data as a pandas DataFrame (‘df’)
2. Identify and calculate the percentage of missing values for each attribute
3. Calculate the number of launches from each launch site – one in California and two in Florida
4. Calculate the number of launch occurrences per each of eleven possible orbits
5. Calculate the number of each landing outcome
6. Create a first stage landing outcome label: 0 = failure, 1 = success
7. Create a new variable (column) in the ‘df’ DataFrame named ‘class’ for the landing outcome label
8. Calculate the mean of the ‘class’ column (mean = 0.666); convert the DataFrame to CSV format using the ‘to_csv’ method

https://github.com/houndboy/IBM-Data-Science-Professional-Certificate-Program/blob/main/C10.Capstone.lab3.data-wrangling.ipynb

EDA with Data Visualization (slide 1)

The following graphs were created using the SpaceX API data set plus the addition of a 'class' (landing outcome) column added during the data wrangling process.

Flight Number vs. Payload Mass: relationship between continuous launch attempts and landing success and landing outcome

Flight Number vs. Launch Site: relationship between continuous launch attempts and launch site and landing outcome

Payload Mass vs. Launch Site: relationship between payload mass and launch site and landing outcome

Success Rate vs. Orbit: relationship between landing outcome ('class') and orbit (eleven unique orbits) used

EDA with Data Visualization (slide 2)

Flight Number vs. Orbit: relationship between continuous launches and different orbits and landing outcome

Payload Mass versus Orbit: relationship between payload mass and orbit and landing outcome

Mean Success Rate versus Year of Launch: relationship between mean landing success over time (in years)

Features Selection: selecting variables (e.g., Flight Number, Payload Mass) that will be used for predictive modeling

Create Features One Hot DataFrame: select feature variables and converting landing outcomes (variable = 'class') to decimal values (either 0.0 for failure or 1.0 for success)

https://github.com/houndboy/IBM-Data-Science-Professional-Certificate-Program/blob/main/C10.Capstone.lab5.eda-dataviz.ipynb

EDA with SQL: sqlite Queries/SpaceX Data Set

Exploratory data analysis of the SpaceX data set (scraped from web) included the following queries:

- Names of launch sites
- Five records beginning with 'CCA'
- Total payload mass carried by boosters launched by NASA (CRS)
- Average payload mass carried by booster version Falcon 9 v1.1
- Date of first successful landing in ground pad
- Boosters having success landing on a drone ship with a payload mass = 4,000 and 6,000 kilograms
- Number of successful and failed missions
- Booster versions carrying the maximum payload
- Failed landings in 2015: month of and booster version
- Count for each landing outcome from 04/06/2010 – 03/20/2017

https://github.com/houndboy/IBM-Data-Science-Professional-Certificate-Program/blob/main/C10.Capstone.lab4.eda-sqlite.ipynb

Folium Interactive Map: Launch Site Location Analysis (slide 1)

For this visual analysis, the folium package was used with an augmented data set containing the latitude and longitude for each launchsite in addition to other launch data (‘spacex_launch_geo.csv’).

- For select launch sites in the United States, using ‘folium.Circle’ (a highlighted circle marker with a text label), appears on the map. This gives a clear indication of where the launch site is located and its’ name. Similarities between launch site locations (i.e., near the equator, near a coastline, and away from populated areas) can be seen.
- For each launch site, successful and failed landing outcomes are marked in a ‘marker_cluster’ with a ‘folium.Marker’. The marker is green if the landing was a success and red if the landing failed. Now, in addition to the launch site name, a yellow circle appears with the total number of launches per launch site.

Folium Interactive Map: Launch Site Location Analysis (slide 2)

For this visual analysis, the folium package was used with an augmented data set containing the latitude and longitude for each launchsite in addition to other launch data ('spacex_launch_geo.csv').

- Next the distance is calculated from the launch site to a city (Lompoc), a railway (Santa Barbara MT1), and a highway (Spring Canyon) in Southern California. A 'folium.Marker' is placed at each geographic site using the site's latitude and longitude. A 'folium.Polyline' is created to connect the launch site to each geographic site using a red line. A text label at the geographic site gives the distance (in kilometers) from the launch site to each georaphic site. This is a clear illustration of the separation of launch sites from heavily populated areas.

https://github.com/houndboy/IBM-Data-Science-Professional-Certificate-Program/blob/main/C10.Capstone.lab6.eda-folium.ipynb

Plotly Dash Dashboard

Interactive plots and graphs:

- Selecting 'All Sites' shows the percentage of successful landings per launch site in piechart form
- Selecting an individual launch site name shows the percentage of successful and failed landings in piechart form

This interactive visualization provides the user a method to examine landing outcomes per launch site and between launch sites.

- Selecting a payload mass between 0 and 10,000 (in increments of 1000 kilograms) produces a scatterplot showing the success rate on the y axis and payload mass on the x axis with the individual point color coded for the booster version used. A color-coded legend, with booster version names, is found on the right-side in the graph's margin.

This interactive visualization provides the user a method to examine relationships between payload masses in kilograms, landing outcomes , and booster versions.

https://github.com/houndboy/IBM-Data-Science-Professional-Certificate-Program/blob/main/C10.Capstone.lab7.dash-app.ipynb

Predictive Analysis (Classification): Process Summary

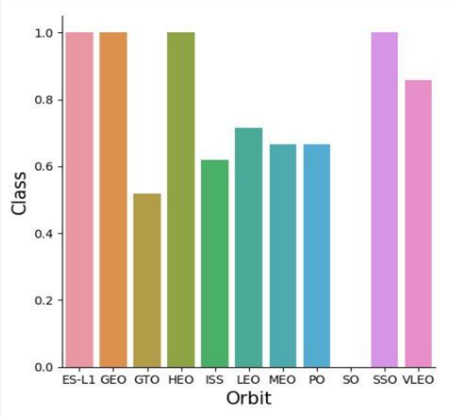
1. Create numpy array from class column, assign to ‘Y’
2. Standardize data in ‘X’, assign to ‘X’
3. Reassign data to ‘X’ using transform method
4. Split ‘X’ and ‘Y’ data into training and testing data
5. Create object (e.g., Logistic Regression)
6. Fit object to find the best parameters
7. Test accuracy of model (fitted object) using the score method
8. Visualize the model using a confusion matrix
9. Find the best model and parameters using the ‘max’ function

The above process was utilized to create, fit, test, and visualize four objects (models): Logistic Regression, K Nearest Neighbors , Support Vector Machine, and Decision Tree.

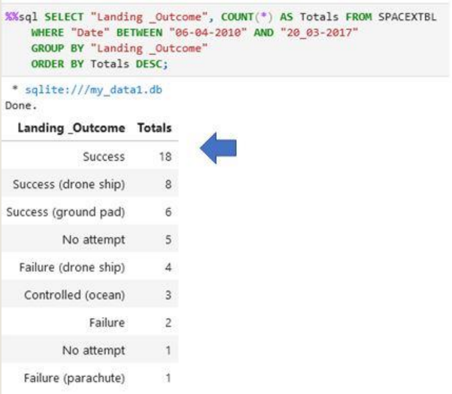
https://github.com/houndboy/IBM-Data-Science-Professional-Certificate-Program/blob/main/C10.Capstone.lab8.machine-learning.ipynb

Sections 2 through 5: Results from . . .

Exploratory Data Analysis:



Graphic Visualizations (Section 2)



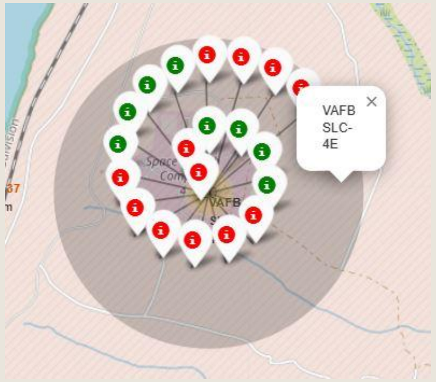
```
%%sql SELECT "Landing _Outcome", COUNT(*) AS Totals FROM SPACEXTBL
    WHERE "Date" BETWEEN "06-04-2010" AND "20_03-2017"
    GROUP BY "Landing _Outcome"
    ORDER BY Totals DESC;
```

* sqlite:///my_data1.db
Done.

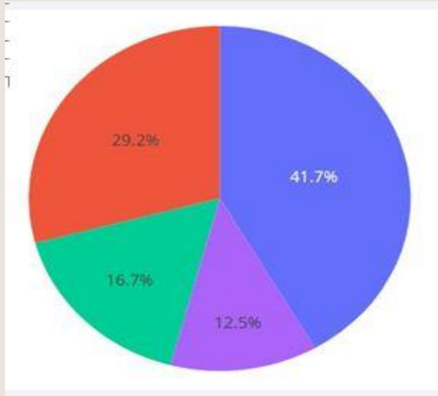
Landing _Outcome	Totals
Success	18
Success (drone ship)	8
Success (ground pad)	6
No attempt	5
Failure (drone ship)	4
Controlled (ocean)	3
Failure	2
No attempt	1
Failure (parachute)	1

SQL Queries (Section 2)

Interactive Visual Analytics:

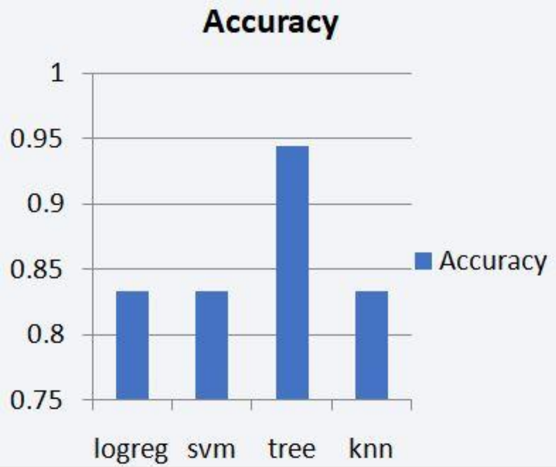


Using folium (Section 3)

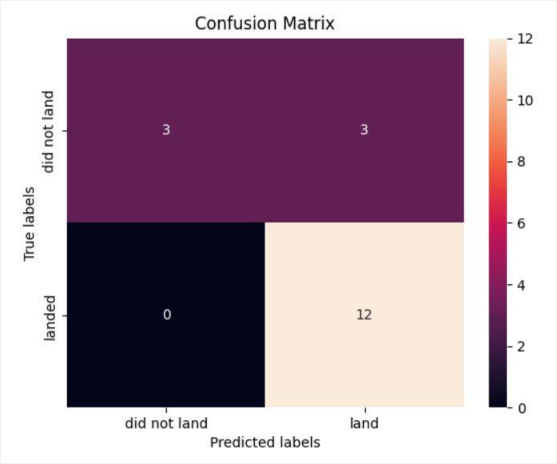


Using Plotly and Dash (Section 4)

Predictive Analysis:



Classification Accuracy (Section 5)

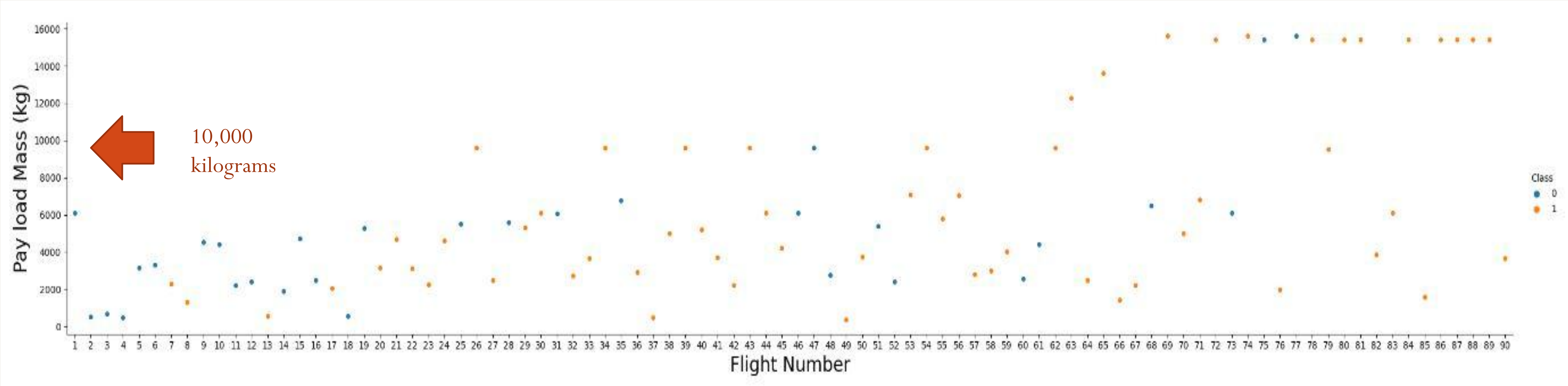


Visualizing Classification Accuracy: Model Confusion Matrix (Section 5)

Section 2: Graphic Visualizations

Insights from Exploratory Data Analysis

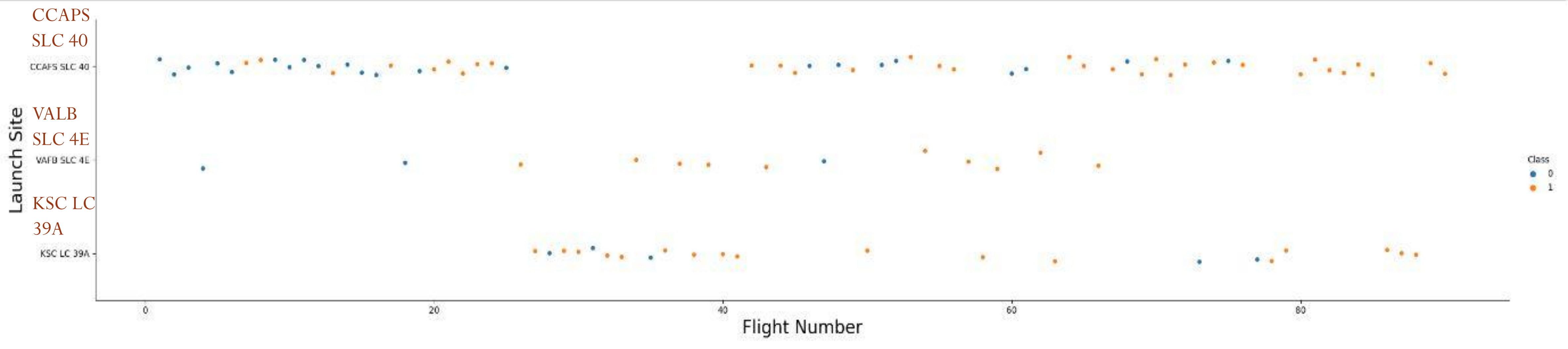
Flight Number vs. Payload Mass




```
"""
1. There is a positive relationship between flight number increase and successful first stage landing.
2. Most launches were with payloads below 6,000 kilograms.
3. From this scatterplot, it is unclear if an increase in payload mass decreases the likelihood
   of a successful landing outcome (i.e., 3 failures and 20 successes with payloads >= 10,000 kg)
"""
```

https://github.com/houndboy/IBM-Data-Science-Professional-Certificate-Program/blob/main/C10.Capstone.lab5.eda-dataviz.ipynb

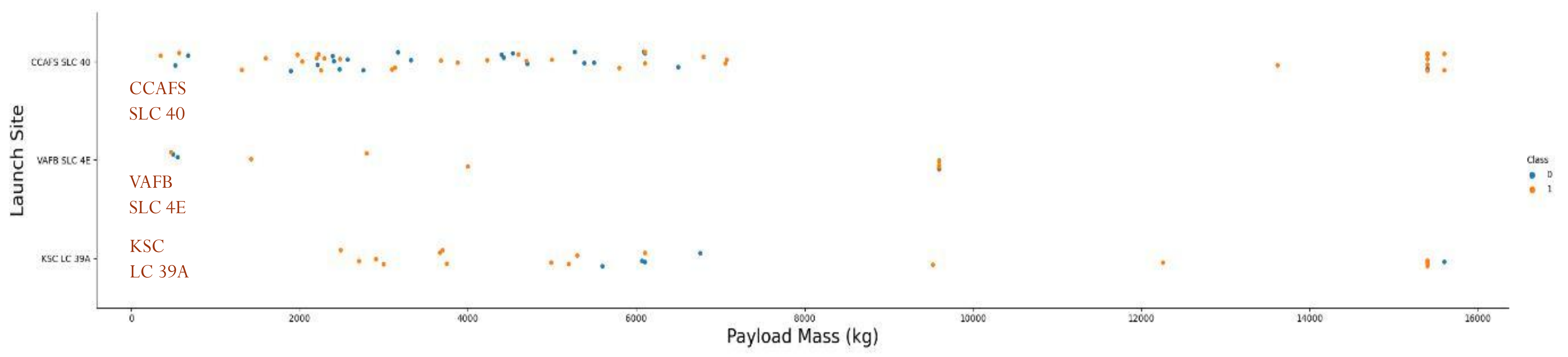
Flight Number vs. Launch Site




```
"""
1. CCAPS SLC 40 had the highest flight number, but the lowest landing success rate.
2. The VAFB SLC 4E had the lowest number of flights, but a high landing success rate.
3. VAB SLC 4E and KSC LC 39A had the highest rate of successful landings.
"""
```

https://github.com/houndboy/IBM-Data-Science-Professional-Certificate-Program/blob/main/C10.Capstone.lab5.eda-dataviz.ipynb

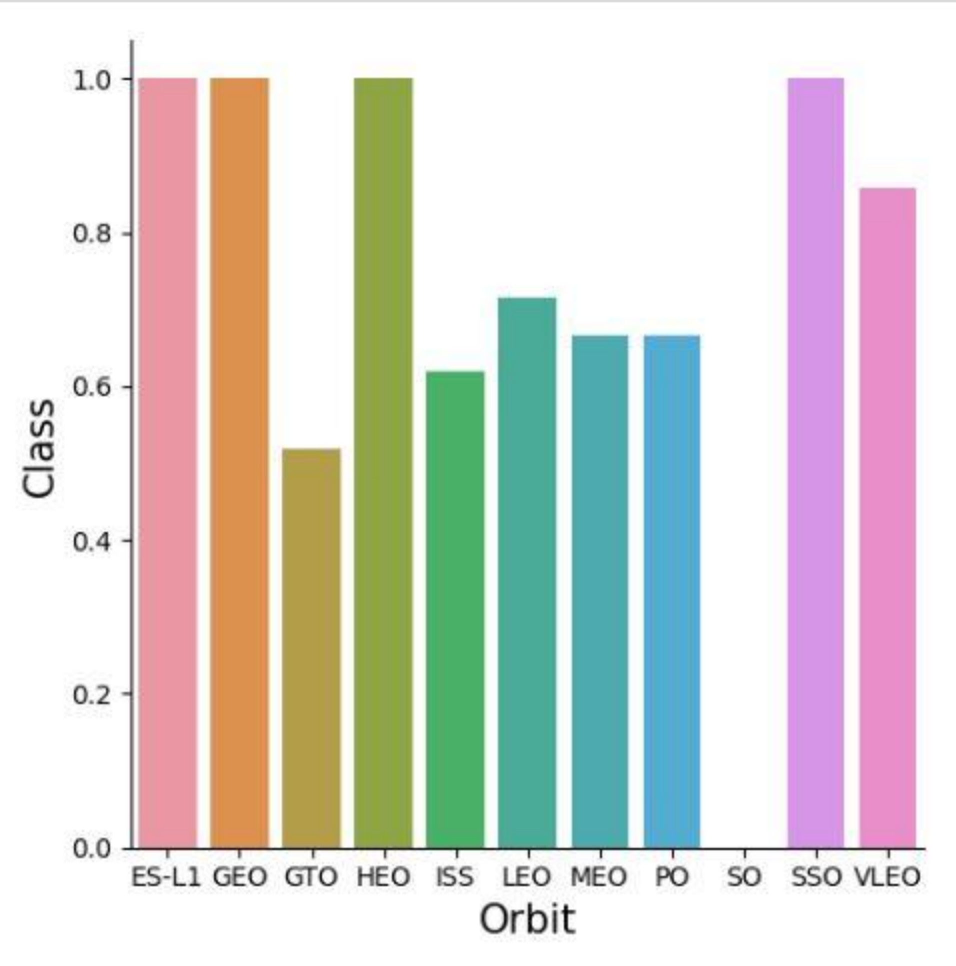
Payload Mass vs. Launch Site




```
"""
1. For VAFB SLC 4E, no rockets were launched for payloads greater than 10,000 kilograms.
2. Most launches carried payloads under 8,000 kilograms.
3. Only CCAFS SLC 40 and KSC LC 39A had launches that carried payloads greater than
   10,000 kilograms.
"""
```

https://github.com/houndboy/IBM-Data-Science-Professional-Certificate-Program/blob/main/C10.Capstone.lab5.eda-dataviz.ipynb

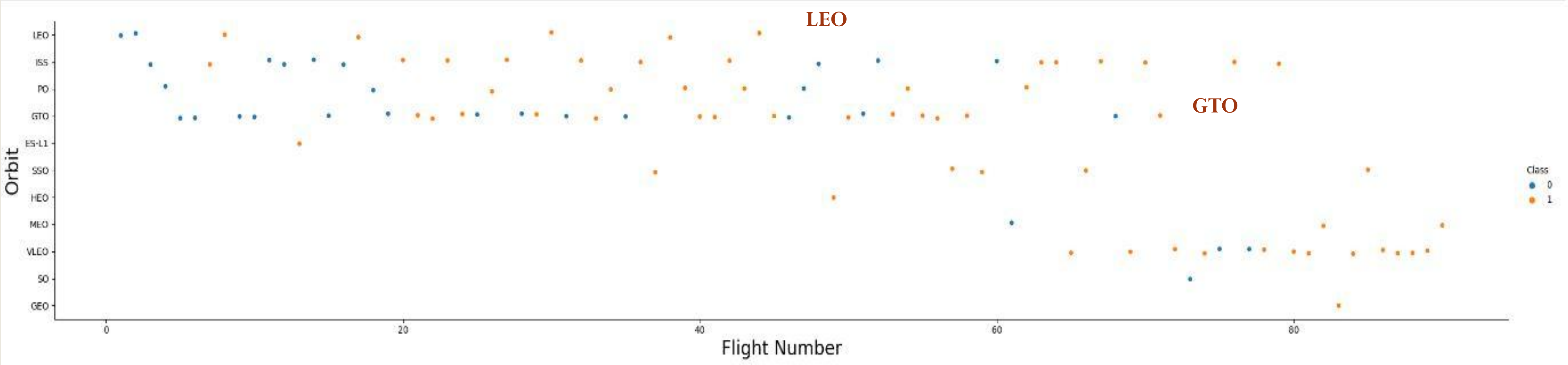
Success Rate vs. Orbit Type




```
"""
1. =100% success rate:            ES-L!, GEO, HEO, and SSO
2. >80% but <100% success rate: VLEO
3. >60% but >80% success rate:   ISS, LEO, MEO, PO
4. <60% success rate:             GTO
5. =0% success rate:              SO
"""
```

https://github.com/houndboy/IBM-Data-Science-Professional-Certificate-Program/blob/main/C10.Capstone.lab5.eda-dataviz.ipynb

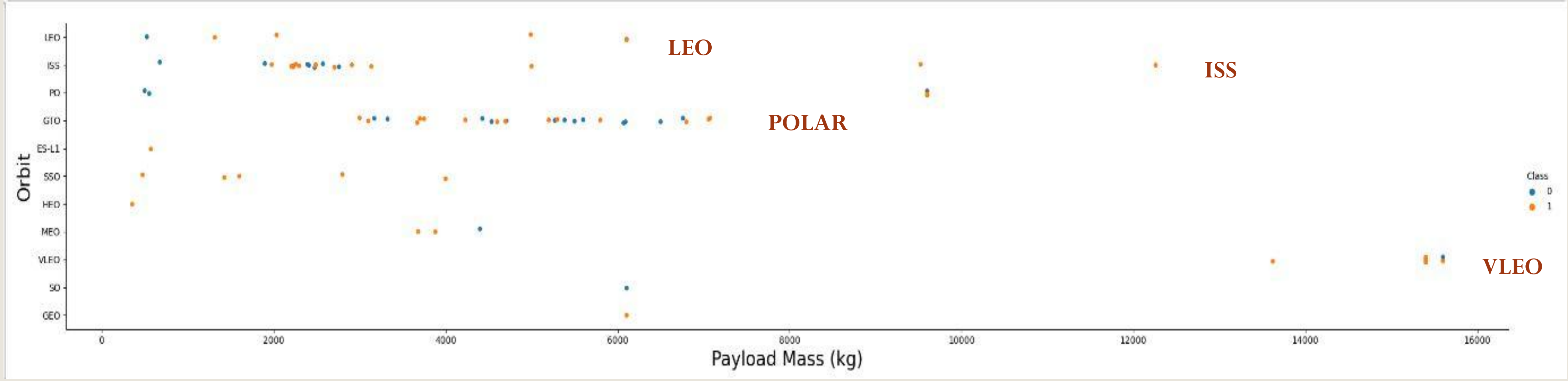
Flight Number vs. Orbit Type




```
"""
1. For the LEO orbit, the flight number was less than 50, but landing successes outnumbered landing failures.
2. For the GTO orbit, the flight number was greater than 60, but landing outcomes were mixed: successes = 14 and failures = 13.
"""
```

https://github.com/houndboy/IBM-Data-Science-Professional-Certificate-Program/blob/main/C10.Capstone.lab5.eda-dataviz.ipynb

Payload vs. Orbit Type

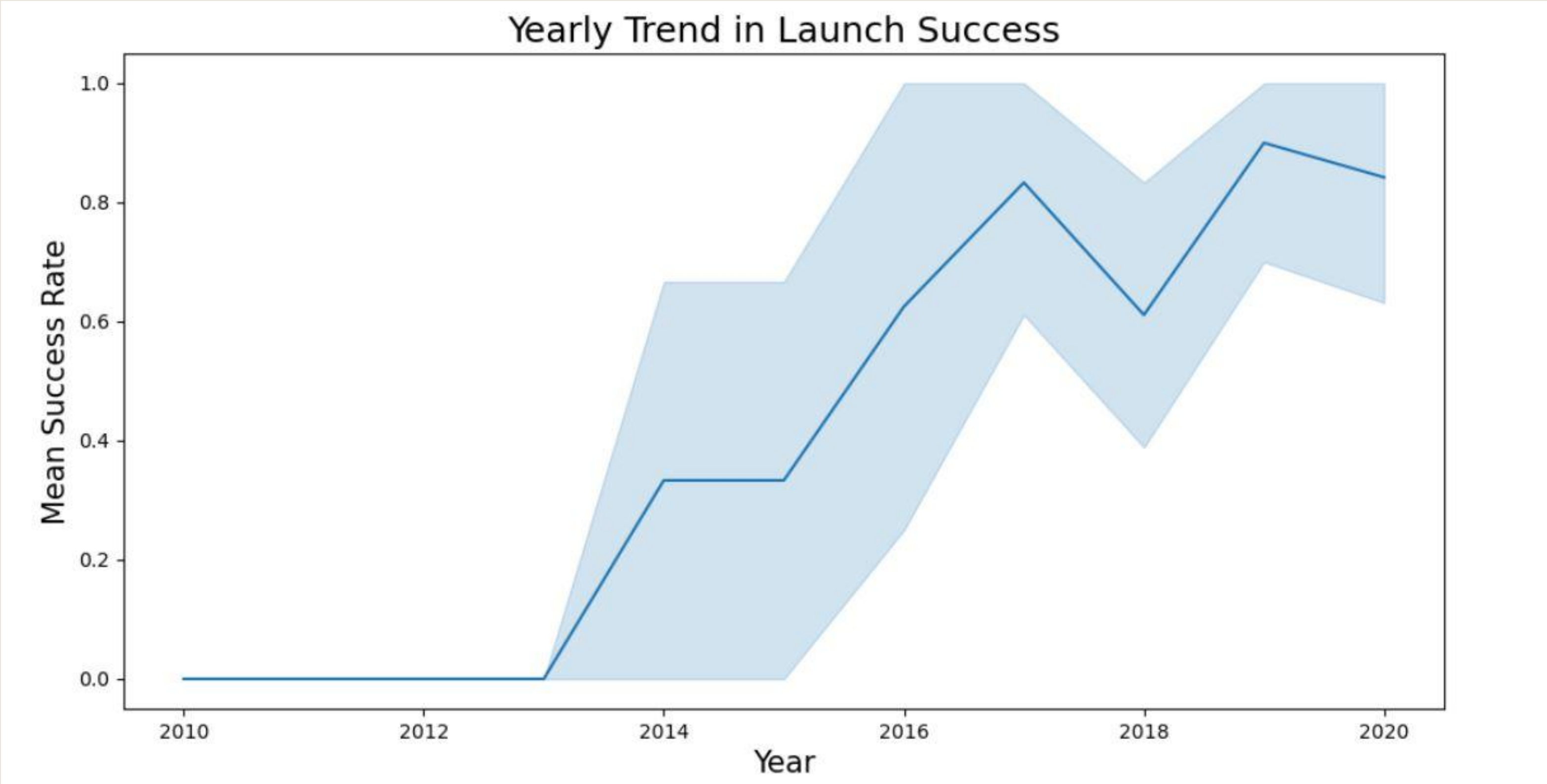



```
"""
1. LEO, ISS, and VLEO orbits are associated with successful landing outcomes and heavy payloads.
2. Launches with the heaviest payloads utilized the ISS and VLEO orbits.
3. There appears to be no relationship between payload mass and the GTO Orbit relative to landing success.
"""
```

https://github.com/houndboy/IBM-Data-Science-Professional-Certificate-Program/blob/main/C10.Capstone.lab5.eda-dataviz.ipynb

Landing Success Yearly Trend

This graph, overall, indicates a positive trend in successful landings over time; however, nuisances include three years of no successful landings (2010 – 2013), a success plateau in 2014, and a setback in 2018.



https://github.com/houndboy/IBM-Data-Science-Professional-Certificate-Program/blob/main/C10.Capstone.lab5.eda-dataviz.ipynb

Section 2: SQL Queries with sqlite

Insights from Exploratory Data Analysis

All Launch Site Names

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)

The SpaceX dataset includes a record for each payload carried during a SpaceX mission into outer space. For exploratory data analysis, using sqlite, the dataset was converted from a comma separated values format (csv) to a table format (‘SPACEXTBL’) - see table head above.

In this SQL query, ‘SELECT DISTINCT’ was used to search the ‘Launch_Site’ data column of the table to find unique names of launch sites.

```
%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

https://github.com/houndboy/IBM-Data-Science-Professional-Certificate-Program/blob/main/C10.Capstone.lab4.eda-sqlite.ipynb

Launch Site Names Begin with 'CCA'

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

```
 * 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

In this SQL query, the first five records were selected from the 'Launch_Site' column in the SPACEXTBL, where the name was like 'CCA.'

https://github.com/houndboy/IBM-Data-Science-Professional-Certificate-Program/blob/main/C10.Capstone.lab4.eda-sqlite.ipynb

Total Payload Mass

In this SQL query, payload masses were matched to the name 'NASA (CRS)' from the 'Customer' column. Then, the masses were summed (using the 'sum' function) to determine the total for all payload masses carried by 'NASA (CRS).'

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

```
%sql SELECT sum(PAYLOAD_MASS__KG_) AS total_payload_mass FROM SPACEXTBL WHERE Customer = "NASA (CRS)";
```

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

total_payload_mass
45596

https://github.com/houndboy/IBM-Data-Science-Professional-Certificate-Program/blob/main/C10.Capstone.lab4.eda-sqlite.ipynb

Average Payload Mass Carried by F9 v1.1

In this SQL query, payload masses were matched to ‘F9 v1.1’ from the ‘Booster_ Version’ column. Then, using the ‘avg’ (average) function, the mean payload mass (in kilograms) was calculated.

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

```
%sql SELECT avg(PAYLOAD_MASS__KG_) AS average_payload FROM SPACEXTBL WHERE Booster_Version = "F9 v1.1";
```

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

average_payload
2928.4

https://github.com/houndboy/IBM-Data-Science-Professional-Certificate-Program/blob/main/C10.Capstone.lab4.eda-sqlite.ipynb

First Successful Ground Landing Date

In this SQL query, the 'min' (minimun) function was used with 'Date' and matched to 'Success (ground pad)' in the 'Landing_ Outcome' column to obtain the date of the first successful landing.

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

https://github.com/houndboy/IBM-Data-Science-Professional-Certificate-Program/blob/main/C10.Capstone.lab4.eda-sqlite.ipynb

```
%sql SELECT min("Date") AS First_Successful_Landing FROM SPACEXTBL WHERE "Landing _Outcome" LIKE "Success (ground pad)";
 * sqlite:///my_data1.db
Done.
```

First_Successful_Landing
01-05-2017

Successful Drone Ship Landing (Payload = 4,000 and 6,000 Kg)

In this SQL query, booster versions, carrying a payload mass between 4,000 and 6,000 kilograms, were selected based on the landing outcome - 'Success (drone ship).'

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

```
%sql select "Booster_Version" from SPACEXTBL where "Landing _Outcome"="Success (drone ship)" and PAYLOAD_MASS__KG_ between '4000' and '6000';
 * sqlite:///my_data1.db
Done.
```

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

https://github.com/houndboy/IBM-Data-Science-Professional-Certificate-Program/blob/main/C10.Capstone.lab4.eda-sqlite.ipynb

Total Number of Successful and Failed Mission Outcomes

In this SQL query, the outcomes in the 'Mission_ Outcome' column have been grouped (using the 'group by' query) and summed (using the 'count' function) to produce a table including 'Mission_Outcome' and 'Mission_Count.'

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

https://github.com/houndboy/IBM-Data-Science-Professional-Certificate-Program/blob/main/C10.Capstone.lab4.eda-sqlite.ipynb

```
%sql select Mission_Outcome, count(*) as Mission_Count from SPACEXTBL group by Mission_Outcome;
 * sqlite:///my_data1.db
Done.
```

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

Boosters Carrying the Maximum Payload (slide 1)

This SQL query uses the 'Booster_ Version' data column and the 'PAYLOAD_ MASS_KG' data column to determine what booster versions carried the largest payload mass.

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

https://github.com/houndboy/IBM-Data-Science-Professional-Certificate-Program/blob/main/C10.Capstone.lab4.eda-sqlite.ipynb

Boosters Carrying the Maximum Payload (slide 2)

In this SQL query, booster versions from the 'Booster_Version' column are matched to the maximum payload mass (using the 'max' function) from the 'Payload_Mass' column. This produced a list of booster versions that have carried the maximum payload mass.

```
%sql select Booster_Version from SPACEXTBL where payload_mass__kg_ = (select max(payload_mass__kg_) from SPACEXTBL);
```

```
 * 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

https://github.com/houndboy/IBM-Data-Science-Professional-Certificate-Program/blob/main/C10.Capstone.lab4.eda-sqlite.ipynb

2015 Launch Records: Failed Landings (slide 1)

The purpose of this SQL query was to determine which booster versions were associated with a failure to land on a drone ship in year 2015. The query uses the 'Booster_Version' and 'Landing_Outcome' data columns.

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

https://github.com/houndboy/IBM-Data-Science-Professional-Certificate-Program/blob/main/C10.Capstone.lab4.eda-sqlite.ipynb

2015 Launch Records: Failed Landings (slide 2)

```
%sql SELECT substr(Date, 4, 2) AS month, "Booster_Version", "Landing _Outcome" FROM SPACEXTBL WHERE "Landing _Outcome"='Failure (drone ship)'
AND substr(Date, 7, 4)='2015';
```

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

month	Booster_Version	Landing _Outcome
01	F9 v1.1 B1012	Failure (drone ship)
04	F9 v1.1 B1015	Failure (drone ship)

In this SQL query, the months of the year and the year 2015 were calculated. Booster version and landing outcome columns were selected. For the landing outcome, 'Failure (drone ship)' was specified. The query matched the specified landing outcome to the booster version to produce the above table giving the month, booster version, and specified landing outcome.

https://github.com/houndboy/IBM-Data-Science-Professional-Certificate-Program/blob/main/C10.Capstone.lab4.eda-sqlite.ipynb

Landing Outcomes: 2010-06-04 and 2017-03-20 (slide 1)

The purpose of this SQL query is to find the total number of times each landing outcome occurred between 4/6/2010 and 3/20/2017. The 'Date' and 'Landing_ Outcome' data columns were used.

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

https://github.com/houndboy/IBM-Data-Science-Professional-Certificate-Program/blob/main/C10.Capstone.lab4.eda-sqlite.ipynb

Landing Outcomes: 2010-06-04 and 2017-03-20 (slide 2)

In this SQL query, each landing outcome is grouped (using the ‘group by’ query) across time (between 4/6/2010 and 3/20/2017) and counted (using the ‘count’ function).

This produced a table of the different landing outcomes, each with a total number of occurrences in descending order.

```
%%sql SELECT "Landing _Outcome", COUNT(*) AS Totals FROM SPACEXTBL
    WHERE "Date" BETWEEN "06-04-2010" AND "20_03-2017"
    GROUP BY "Landing _Outcome"
    ORDER BY Totals DESC;
```

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

Landing _Outcome	Totals
Success	18
Success (drone ship)	8
Success (ground pad)	6
No attempt	5
Failure (drone ship)	4
Controlled (ocean)	3
Failure	2
No attempt	1
Failure (parachute)	1

https://github.com/houndboy/IBM-Data-Science-Professional-Certificate-Program/blob/main/C10.Capstone.lab4.eda-sqlite.ipynb

Section 3

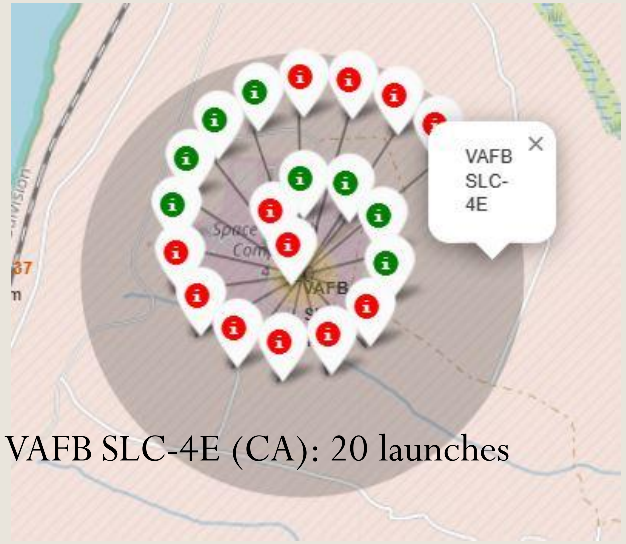
Launch Sites Proximities Analysis Using folium

Launch Site Locations

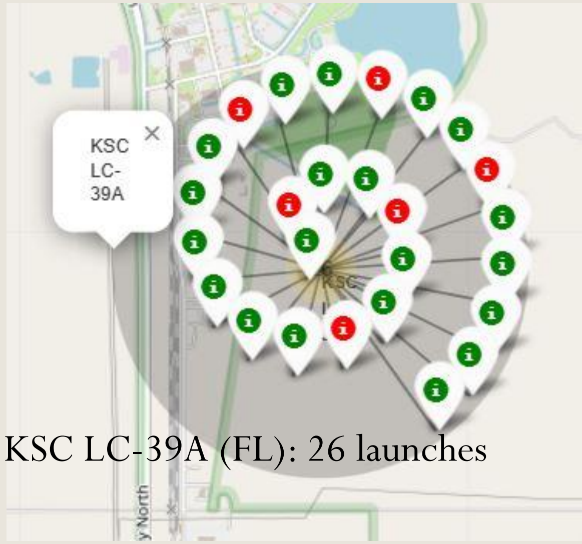
The three launch sites in the United States are all located on a coastline for safety reasons and near the equator to augment launch and orbital success. In California, there is launch site **VAFB SLC – 4E** which had 20 launches. In Florida, there are two launch sites, **KSC LC-39A** with 26 launches and **CCAFS SLC-40** with 66 launches. The total launches for both Florida sites = 92.

https://github.com/houndboy/IBM-Data-Science-Professional-Certificate-Program/blob/main/C10.Capstone.lab6.eda-folium.ipynb

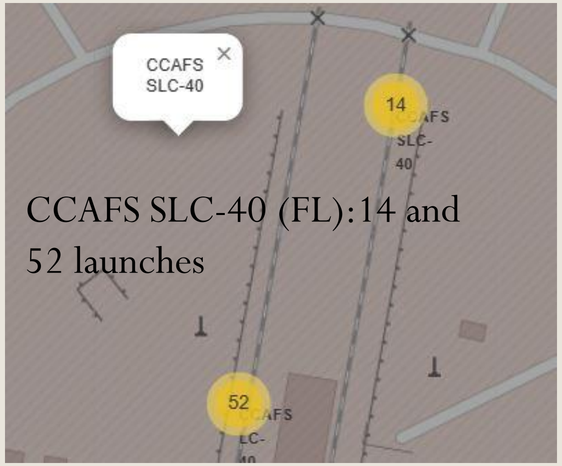
Landing Successes and Failures by Site



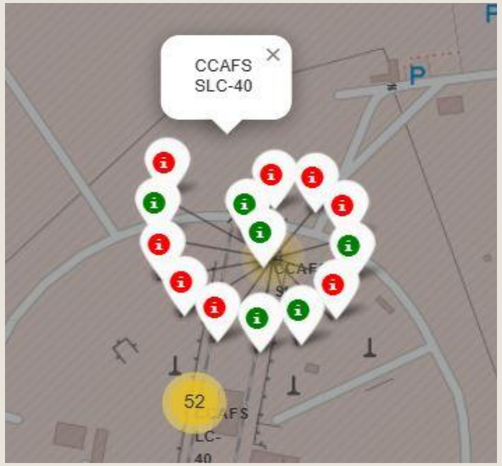
VAFB SLC-4E (CA): 20 launches

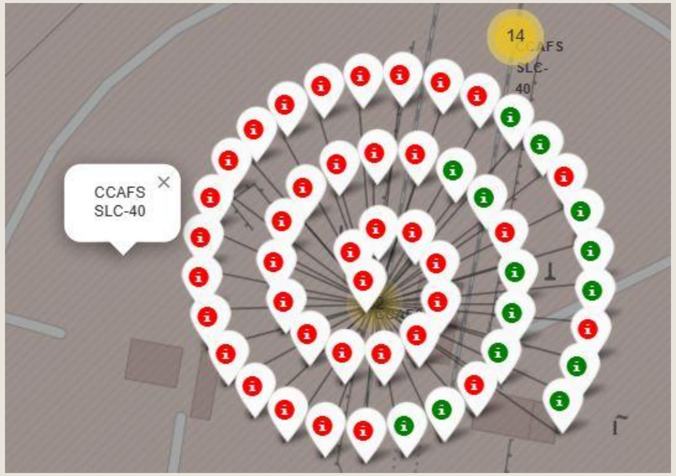


KSC LC-39A (FL): 26 launches



CCAFS SLC-40 (FL):14 and 52 launches





Successful landings are indicated by a green marker and failed landings are indicated by a red marker. Note: CCAFS SLC-40 launch site in Florida consists of two launch sites. One site had14 launches and the other site had 52 launches for a total of 66 launches for CCAFS SLC-40.

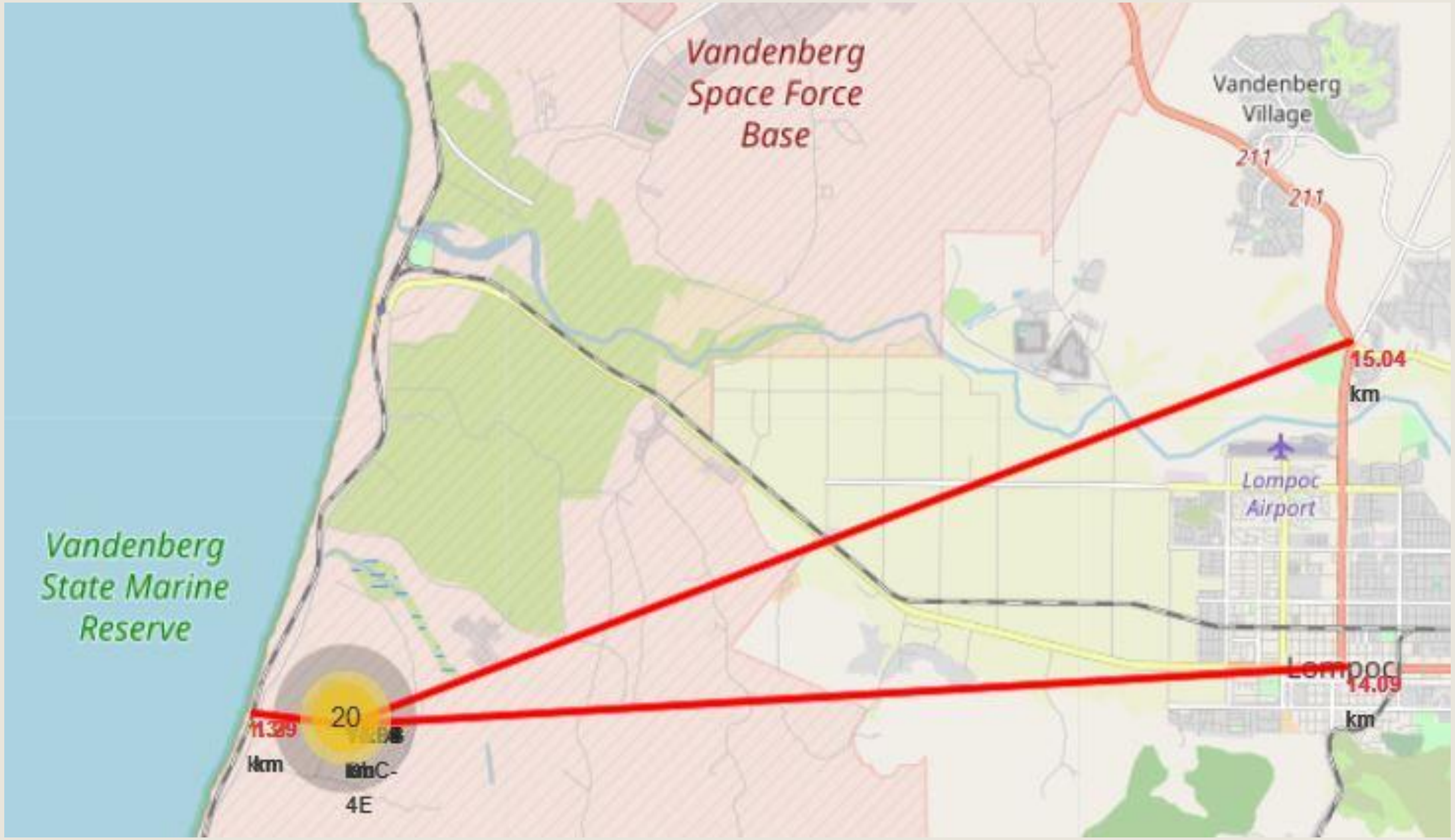
https://github.com/houndboy/IBM-Data-Science-Professional-Certificate-Program/blob/main/C10.Capstone.lab6.eda-folium.ipynb

Distances from Launch Site VAFB SLC-4E (CA)

The shortest line extends to the Santa Barbara MT1 railway, which is 1.29 kilometers away from launch site. Its' close proximity to the launch site is unavoidable, since the MT1 runs along the coastline.

The top long line extends to Spring Canyon Highway, which is 15.04 kilometers from the launch site. The bottom line extends to Lompoc City, which is 14.09 kilometers from the launch Site.

For safety reasons, launch sites are located away from heavily populated areas as much as possible.



https://github.com/houndboy/IBM-Data-Science-Professional-Certificate-Program/blob/main/C10.Capstone.lab6.eda-folium.ipynb

Section 4

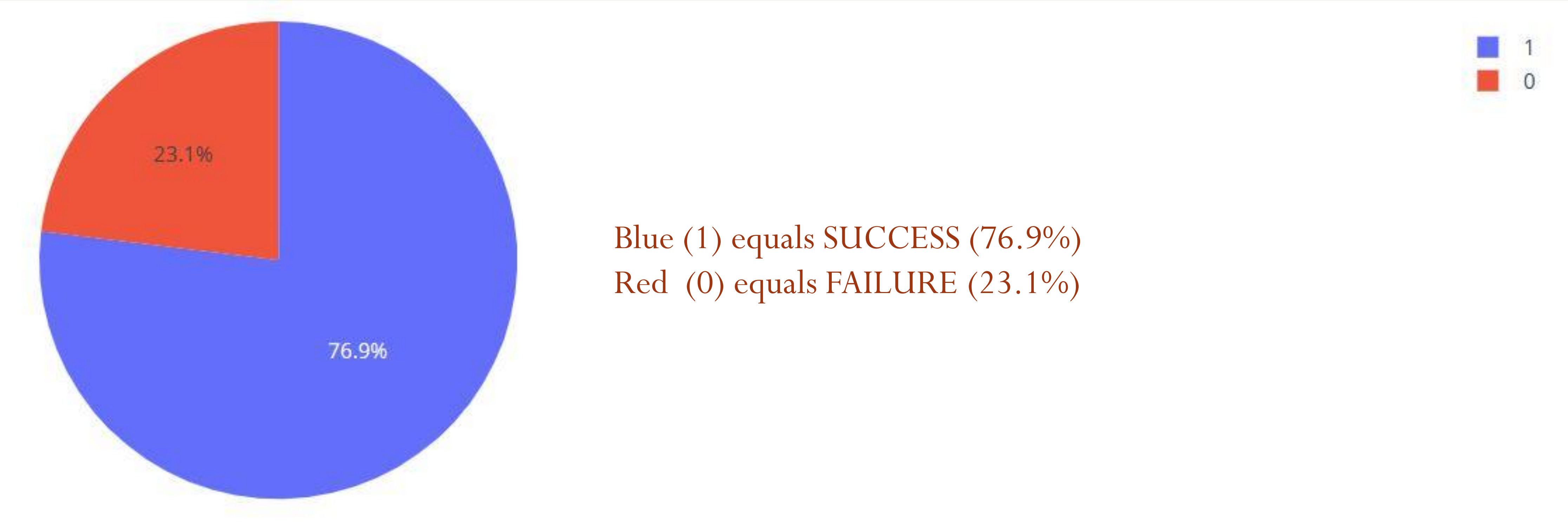
Interactive Dashboard Using Plotly and Dash

Total Successful Launches by Launch Site

KSC LC-39A is the most successful launch site with a 41.7% success rate; followed by CCAFS LC-40 with a 29.2% success rate. VAFB SLC-4E had a 16.7% success rate and CCAFS SLC-40 had the lowest success rate of 12.5%.

https://github.com/houndboy/IBM-Data-Science-Professional-Certificate-Program/blob/main/C10.Capstone.lab7.dash-app.ipynb

Most Successful Launch Site: KSC LC-39A (FL)

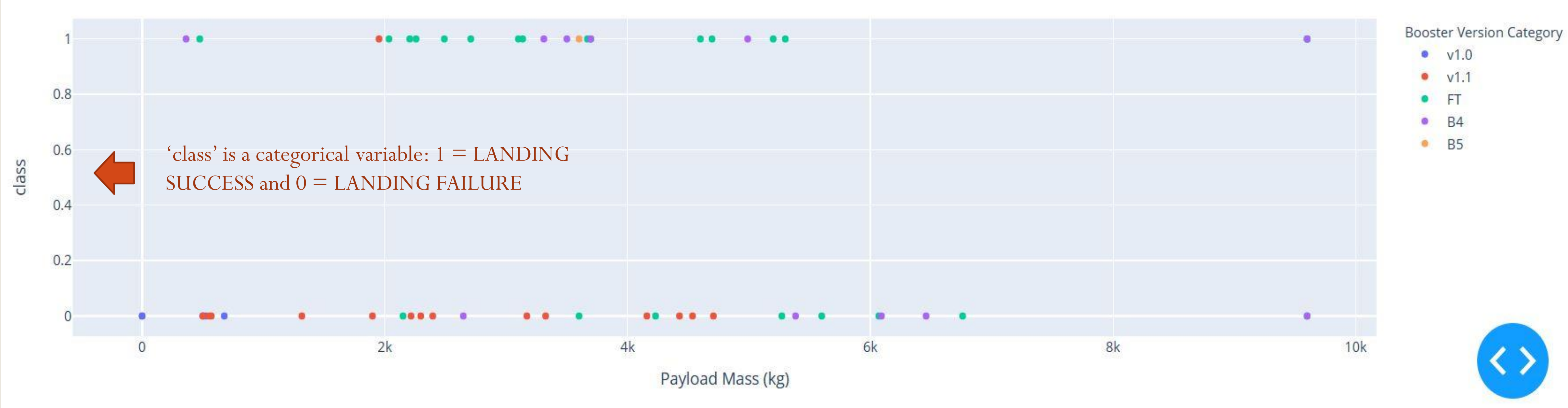


Blue (1) equals SUCCESS (76.9%)
Red (0) equals FAILURE (23.1%)

https://github.com/houndboy/IBM-Data-Science-Professional-Certificate-Program/blob/main/C10.Capstone.lab7.dash-app.ipynb

All Sites - Payload Mass vs. Landing Outcome: 0 - 10,000 Kg

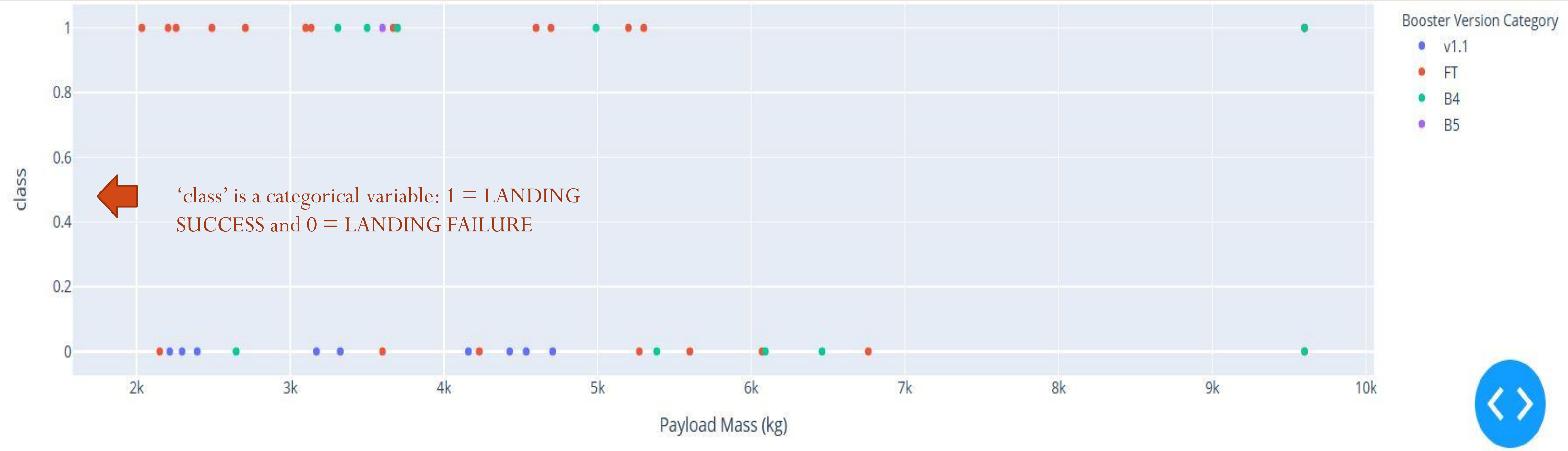
The FT (green dot), the B4 (purple dot), and the V1.1 (red dot) booster versions were successful with payloads of 0 – 2,000 kg. Launches using the V1.0 (blue dot) booster version had no successful first stage landings.



https://github.com/houndboy/IBM-Data-Science-Professional-Certificate-Program/blob/main/C10.Capstone.lab7.dash-app.ipynb

All Sites - Payload Mass vs. Landing Outcome: 2,000 - 10,000 Kg

The FT (red dot) was the most successful booster version and the B4 (green dot) was the next most successful booster version for 2,000 – 2,600 kg payloads. The B4 had one success with a payload greater than 2,900 kg.

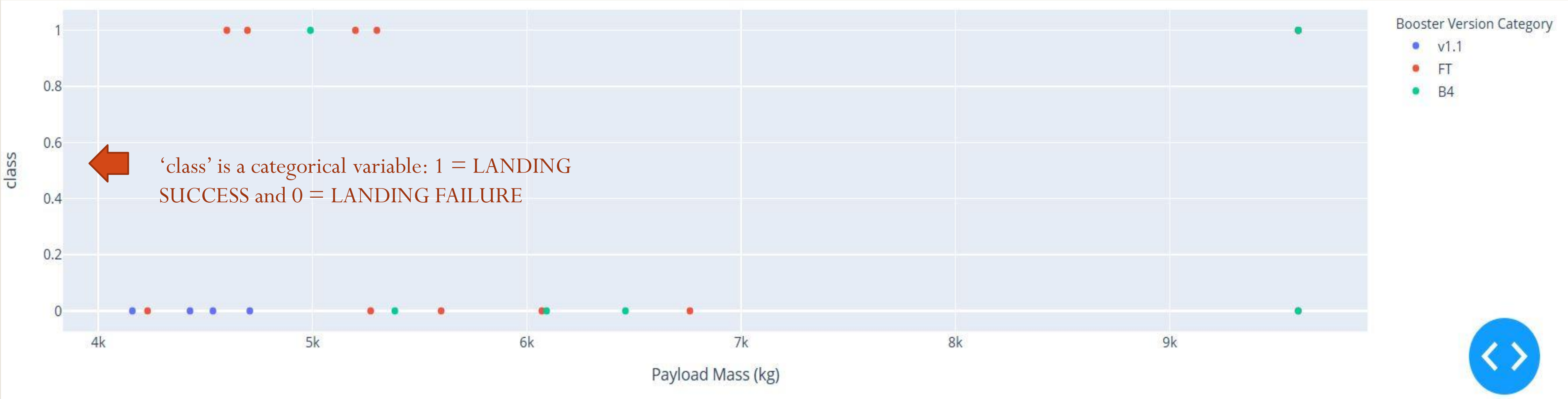


'class' is a categorical variable: 1 = LANDING SUCCESS and 0 = LANDING FAILURE

https://github.com/houndboy/IBM-Data-Science-Professional-Certificate-Program/blob/main/C10.Capstone.lab7.dash-app.ipynb

All Sites – Payload Mass vs. Landing Outcome: 4,000 - 10,000 Kg

The FT booster version (red dot) had four successful launches with 4,000 – 4,600 kg payloads. The B4 booster version (green dot) had one successful launch with a payload of approximately 4,500 kg and with a payload greater than 4,900 kg. The FT booster version handles payloads below 4,600 better than other booster versions.

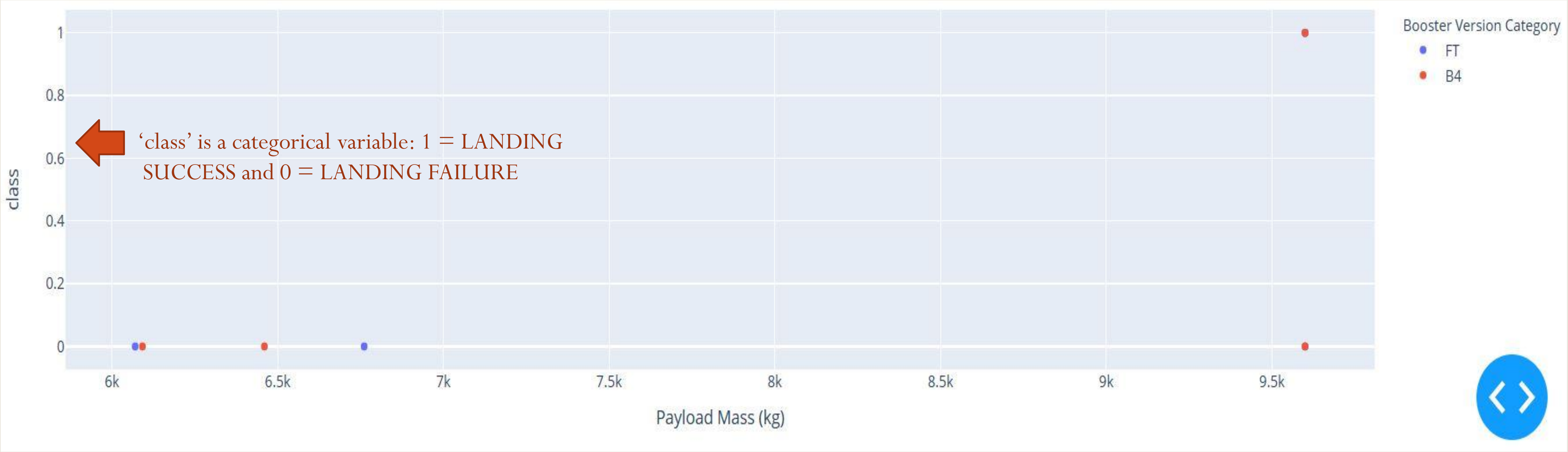


'class' is a categorical variable: 1 = LANDING SUCCESS and 0 = LANDING FAILURE

https://github.com/houndboy/IBM-Data-Science-Professional-Certificate-Program/blob/main/C10.Capstone.lab7.dash-app.ipynb

All Sites – Payload Mass vs. Landing Outcome: 6000 – 10,000 Kg

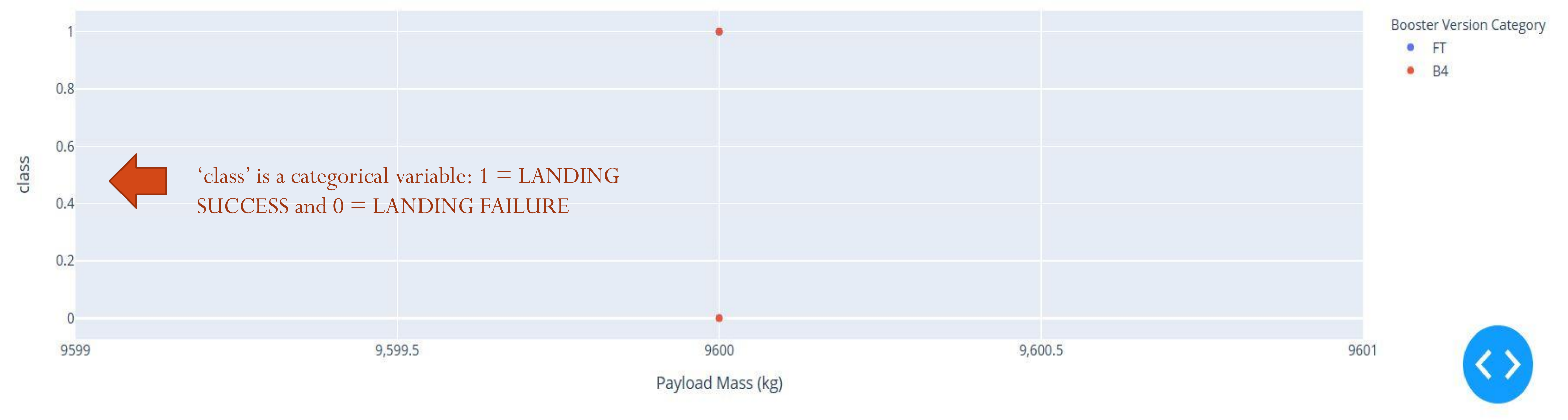
Consistent with previous scatterplots, as payload mass increases, there is a decrease in launches. In this kilogram range, only the FT (blue dot) and the B4 (red dot) booster versions were used in launches. Launches with the FT booster version had no successful outcomes.



https://github.com/houndboy/IBM-Data-Science-Professional-Certificate-Program/blob/main/C10.Capstone.lab7.dash-app.ipynb

All Sites – Payload Mass vs. Landing Outcome: 9,599 – 9,601 Kg

The B4 booster version (red dot) had one successful landing out of two launches with a payload of approximately 9,600 kg. Consistent with previous scatter plots, as payload mass increases, launch attempts decrease. The B4 booster version is more capable of successfully handling payloads greater that 4,600 kg as compared to the performance of other booster versions.



https://github.com/houndboy/IBM-Data-Science-Professional-Certificate-Program/blob/main/C10.Capstone.lab7.dash-app.ipynb

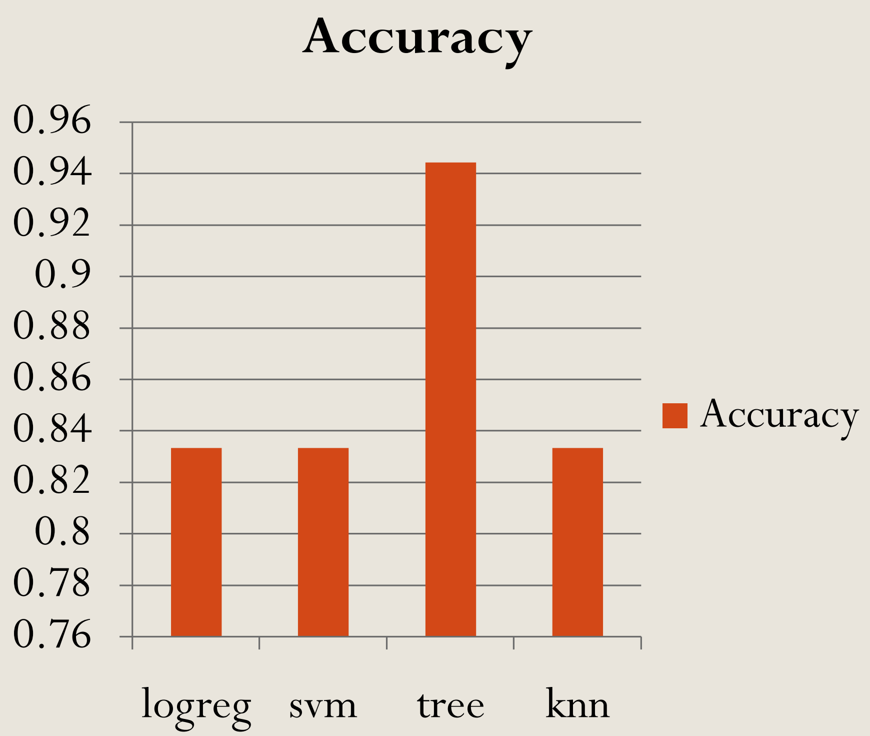
Section 5

Predictive Analysis (Classification)

Classification Accuracy: Correctly Labeling New Data

Using the 'score' method, the accuracy of four statistical classification algorithms are compared:

1. K Nearest Neighbor ('knn'): accuracy = 0.83, and
2. Support Vector Machine Object ('svm'): accuracy = 0.83,
3. Logistic Regression ('logreg'): accuracy = 0.83,
4. Decision Tree ('tree'): accuracy = 0.94.



https://github.com/houndboy/IBM-Data-Science-Professional-Certificate-Program/blob/main/C10.Capstone.lab8.machine-learning.ipynb

Confusion Matrix: Logistic Regression (slide 1)

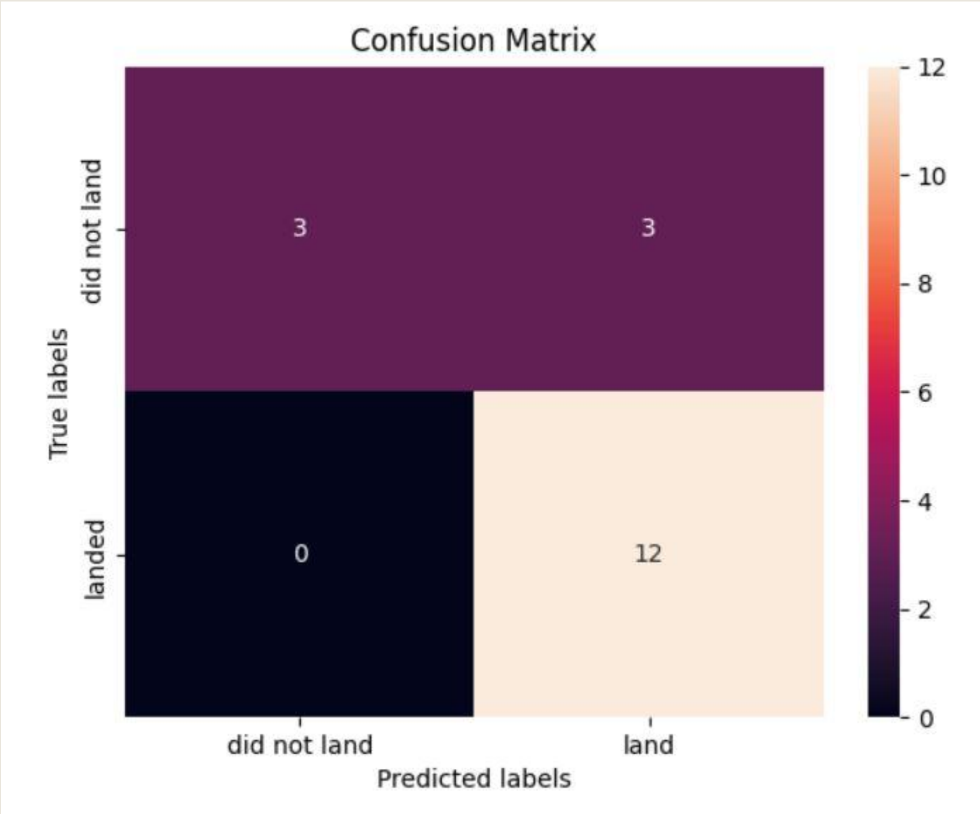
The confusion matrix is a visualization of the outcome of a statistical classification algorithm; i.e., a supervised machine learning model. Each column represents a ***predicted instance*** of an event. Whereas, each row represents an ***actual instance***. Classifaction accuracy is determined using the 'score' method.

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

```
logreg_accuracy=logreg_cv.score(X_test, Y_test)
logreg_accuracy
```

```
0.8333333333333334
```

Classification Accuracy = 0.833



https://github.com/houndboy/IBM-Data-Science-Professional-Certificate-Program/blob/main/C10.Capstone.lab8.machine-learning.ipynb

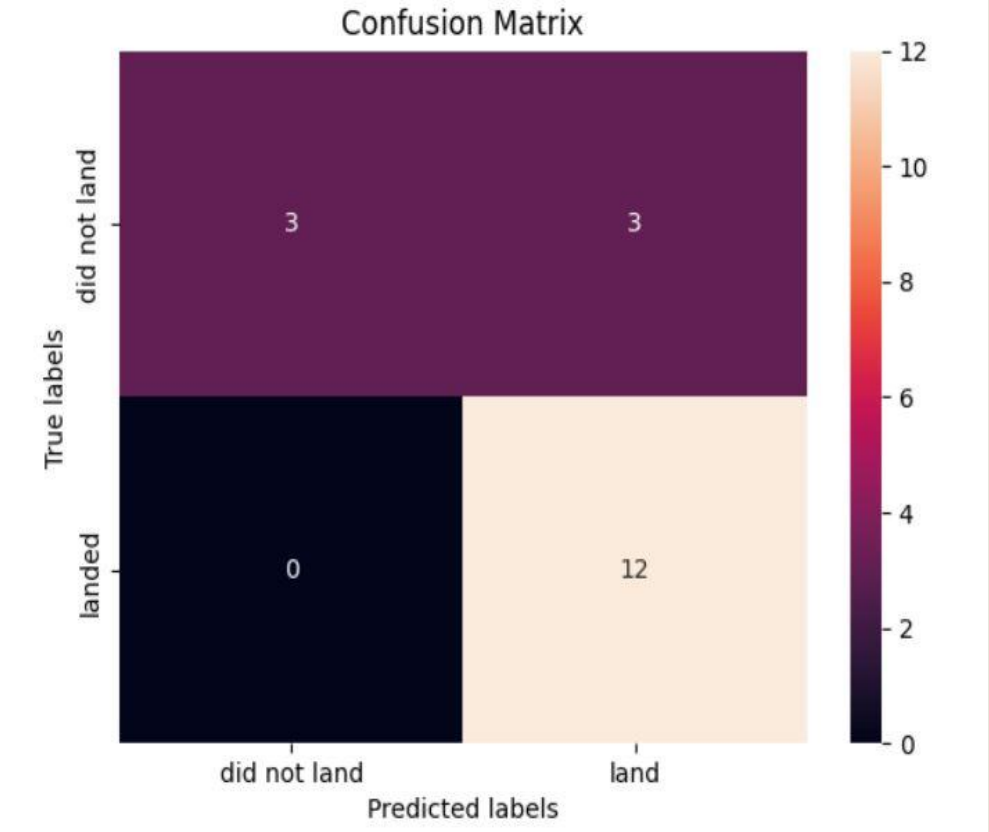
Confusion Matrix: Support Vector Machine (slide 2)

Classification Accuracy = 0.8333

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

```
svm_accuracy=svm_cv.score(X_test, Y_test)
svm_accuracy
```

```
0.8333333333333334
```



https://github.com/houndboy/IBM-Data-Science-Professional-Certificate-Program/blob/main/C10.Capstone.lab8.machine-learning.ipynb

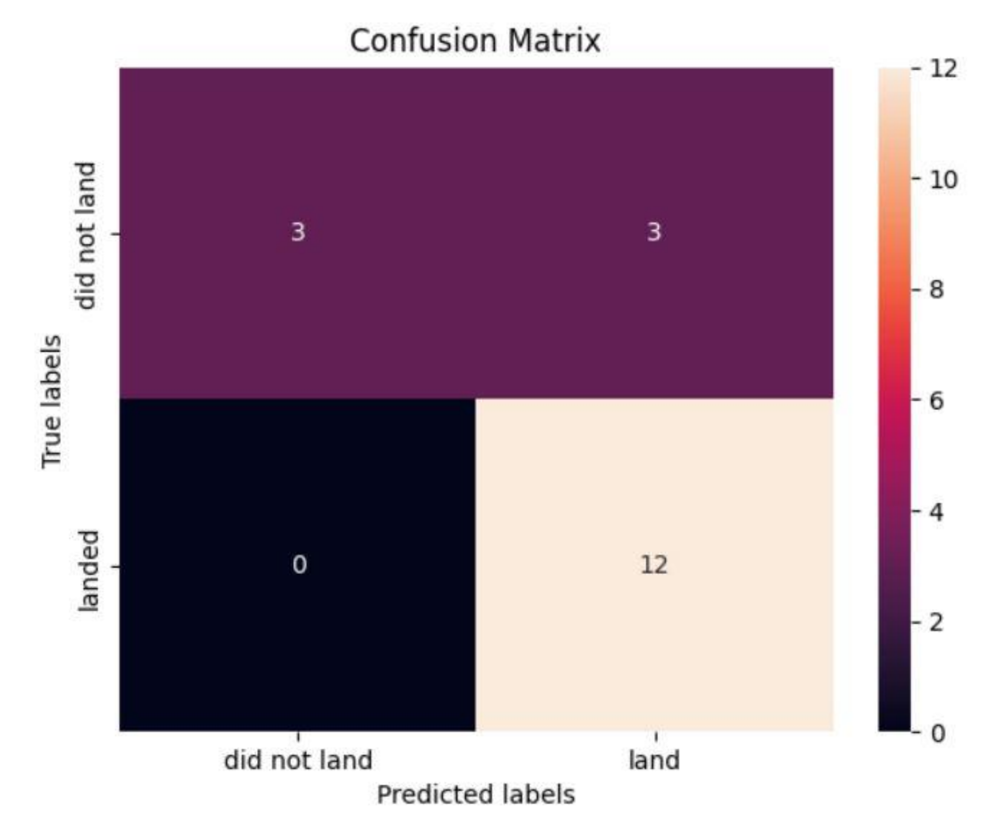
Confusion Matrix: K Nearest Neighbors (slide 3)

Classification Accuracy = 0.8333

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

```
knn_accuracy=knn_cv.score(X_test, Y_test)
knn_accuracy
```

```
0.8333333333333334
```



https://github.com/houndboy/IBM-Data-Science-Professional-Certificate-Program/blob/main/C10.Capstone.lab8.machine-learning.ipynb

Confusion Matrix: Decision Tree (slide 4)

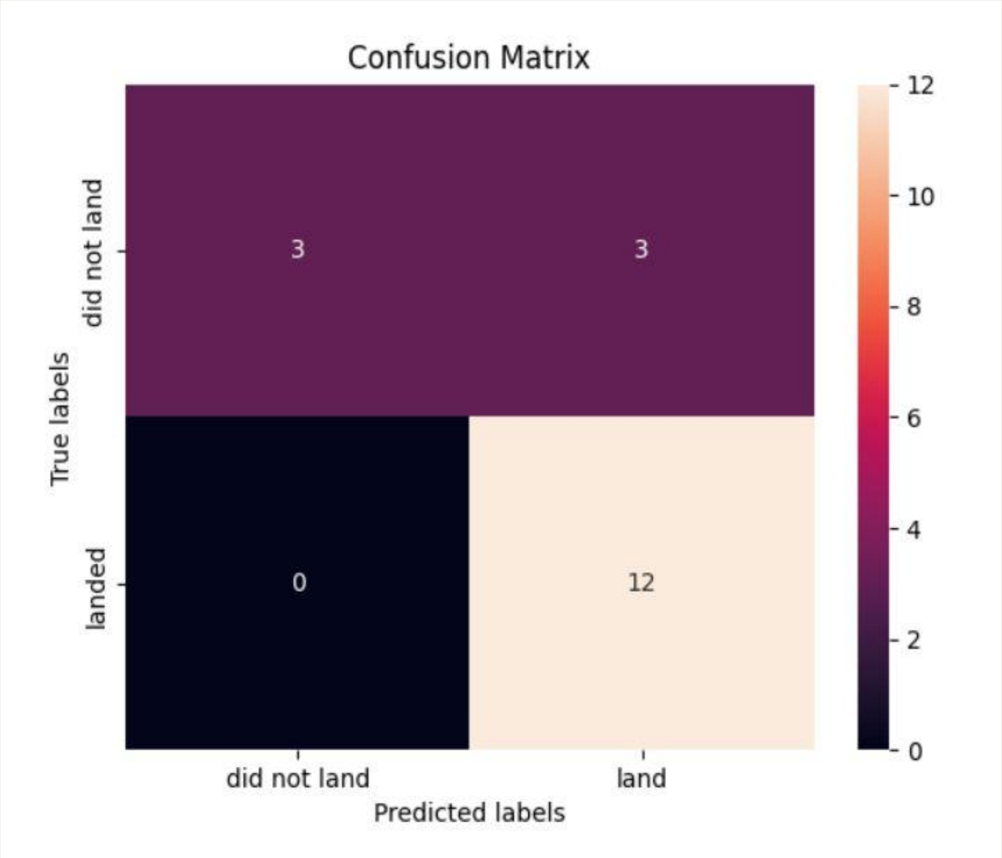
The Decision Tree model has the greatest classification accuracy out of all models.

Classification Accuracy = 0.9444

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

```
tree_accuracy=tree_cv.score(X_test, Y_test)
tree_accuracy
```

```
0.9444444444444444
```



https://github.com/houndboy/IBM-Data-Science-Professional-Certificate-Program/blob/main/C10.Capstone.lab8.machine-learning.ipynb

Tuned Hyperparameters (‘best parameters’ and ‘accuracy’)

For each model, in addition to classification accuracy, the ‘best parameters’ are listed and the ‘accuracy’ of the validation data is calculated. The Decision Tree (‘tree’) model had the highest classification accuracy (0.94), the most (‘best’) parameters, and the highest accuracy for its’ validation data (0.8875).

Logistic Regression (‘logreg’)

```
tuned hpyerparameters :(best parameters)  {'C': 0.01, 'penalty': 'l2', 'solver': 'lbfgs'}
accuracy : 0.8464285714285713
```

Support Vector Machine Object (‘svm’)

```
tuned hpyerparameters :(best parameters)  {'C': 1.0, 'gamma': 0.03162277660168379, 'kernel': 'sigmoid'
accuracy : 0.8482142857142856
```

K Nearest Neighbors (‘knn’)

```
tuned hpyerparameters :(best parameters)  {'algorithm': 'auto', 'n_neighbors': 10, 'p': 1}
accuracy : 0.8482142857142858
```

Decision Tree (‘tree’)

```
tuned hpyerparameters :(best parameters)  {'criterion': 'entropy', 'max_depth': 4, 'max_features': 'sqrt', 'min_samples_leaf':
2, 'min_samples_split': 2, 'splitter': 'random'}
accuracy : 0.8875
```

Decision Tree: Best Model (Accuracy = 0.8875)

```
models={'LogisticRegression':logreg_cv.best_score_, 'SupportVector':svm_cv.best_score_,
        'DecisionTree':tree_cv.best_score_, 'KNeighbors':knn_cv.best_score_}

best_algorithm=max(models, key=models.get)
print('The best model is', best_algorithm, 'with a score of', models[best_algorithm])

if best_algorithm=='LogisticRegression':
    print('The best parameters are:', logreg_cv.best_params_)
if best_algorithm=='SupportVector':
    print('The best parameters are:', svm_cv.best_params_)
if best_algorithm=='DecisionTree':
    print('The best parameters are:', tree_cv.best_params_)
if best_algorithm=='KNeighbors':
    print('The best parameters are:', knn_cv.best_params_)
```

```
The best model is DecisionTree with a score of 0.8875
The best parameters are: {'criterion': 'entropy', 'max_depth': 4, 'max_features': 'sqrt', 'min_samples_leaf': 2, 'min_samples_split': 2, 'splitter': 'r
andom'}
```

https://github.com/houndboy/IBM-Data-Science-Professional-Certificate-Program/blob/main/C10.Capstone.lab8.machine-learning.ipynb

Section 6

Conclusions

Conclusions (slide 1):

- Continuous lauch attempts (flight number) increases the probability of successful first stage landings.
- Large payload mass is inversely related to successful first stage landings and launch number.
- Launch sites vary in landing outcomes; e.g., KSC LC 39A = 77% success; whereas, CCAFS LC-40 = 60% success.
- A relationship exists between the orbit used and successful first stage landing; e.g., GEO = 100% success; whereas, GT0 < 60%.
- In year 2013, positive landing outcomes began to increase each year and an upward trend continued through 2020.

Conclusions (slide 2):

- Launch site locations are near the equator, on a coastline, and away from heavily populated areas for launch and safety reasons.
- There is a relationship between booster version, payload mass, and landing outcome; e.g., B4 booster version had landing success with payload masses > 6,000 kg; whereas, FT booster version had landing success with payload masses < 6,000 kg.
- The best predictive model is Decision Tree with a classification accuracy of 0.94.
- The Decision Tree model had more parameters with greater parameter accuracy (0.8875) than other supervised learning models.

Section 7

Appendices

Appendix (slide 1): Questions to be Answered

For Future Exploration and Analysis:

1. Identifying variables per launch site influencing first stage landing success – **Why is rate of success so variable amongst launch sites and what variables influence its' change over time?**

2. Identifying variables per orbit influencing first stage landing success – **Why are landing outcomes so variable amongst the orbits used during flight?**

3. Identifying variables per booster version influencing first stage landing success – **What characteristics of a booster version are positively correlated with a successful first stage landing?**

4. Identifying the relationship between multiple variables influencing first stage landing success – **How are orbit, payload mass , booster version, and launch site related to successful first stage landing?**

5. As variables are identified and relationships are hypothesized, predictive analysis of more complex data will be required – **What machine learning models are appropriate for multivariable data (e.g., multivariable regression analysis , multivariable adaptive regression spines (MARS), etc.)?**

Appendix (slide 2): Examples of Data Sources

'Unwrangled' Launch Data From a SpaceX API Request

b'[{"fairings":{"reused":false,"recovery_attempt":false,"recovered":false,"ships":[]},"links":{"patch":{"small":"https://images2.imgbox.com/94/f2/NN6Ph
45r_o.png","large":"https://images2.imgbox.com/5b/02/QcxHUb5V_o.png"},"reddit":{"campaign":null,"launch":null,"media":null,"recovery":null},"flickr":{"
small":[],"original":[]},"presskit":null,"webcast":"https://www.youtube.com/watch?v=0a_00nJ_Y88","youtube_id":"0a_00nJ_Y88","article":"https://www.spac
e.com/2196-spacex-inaugural-falcon-1-rocket-lost-launch.html","wikipedia":"https://en.wikipedia.org/wiki/DemoSat"},"static_fire_date_utc":"2006-03-17T0
0:00:00.000Z","static_fire_date_unix":1142553600,"net":false,"window":0,"rocket":"5e9d0d95eda69955f709d1eb","success":false,"failures":[{"time":33,"alt
itude":null,"reason":"merlin engine failure"}],"details":"Engine failure at 33 seconds and loss of vehicle","crew":[],"ships":[],"capsules":[],"payload
s":["5eb0e4b5b6c3bb0006eeb1e1"],"launchpad":"5e9e4502f5090995de566f86","flight_number":1,"name":"FalconSat","date_utc":"2006-03-24T22:30:00.000Z","date
_unix":1143239400,"date_local":"2006-03-25T10:30:00+12:00","date_precision":"hour","upcoming":false,"cores":[{"core":"5e9e289df35918033d3b2623","fligh
t":1,"gridfins":false,"legs":false,"reused":false,"landing_attempt":false,"landing_success":null,"landing_type":null,"landpad":null}],"auto_update":tru

Web Scraped Table Excert From Wikipedia Page – 'Falcon 9 Launches and Heavy Launches'

80	29 January 2020, 14:07[501]	F9 B5 ♺ B1051.3	CCAFS, SLC-40	Starlink 3 v1.0 (60 satellites)	15,600 kg (34,400 lb)[5]	LEO	SpaceX	Success	Success (drone ship)
	Third operational and fourth large batch of Starlink satellites, deployed in a circular 290 km (180 mi) orbit. One of the fairing halves was caught, while the other was fished out of the ocean.[502]								
81	17 February 2020, 15:05[503]	F9 B5 ♺ B1056.4	CCAFS, SLC-40	Starlink 4 v1.0 (60 satellites)	15,600 kg (34,400 lb)[5]	LEO	SpaceX	Success	Failure (drone ship)
	Fourth operational and fifth large batch of Starlink satellites. Used a new flight profile which deployed into a 212 km × 386 km (132 mi × 240 mi) elliptical orbit instead of launching into a circular orbit and firing the second stage engine twice. The first stage booster failed to land on the drone ship[504] due to incorrect wind data.[505] This was the first time a flight proven booster failed to land.								
82	7 March 2020, 04:50[506]	F9 B5 ♺ B1059.2	CCAFS, SLC-40	SpaceX CRS-20 (Dragon C112.3 ♺)	1,977 kg (4,359 lb)[507]	LEO (ISS)	NASA (CRS)	Success	Success (ground pad)
	Last launch of phase 1 of the CRS contract. Carries *Bartolomeo*, an ESA platform for hosting external payloads onto ISS.[508] Originally scheduled to launch on 2 March 2020, the launch date was pushed back due to a second stage engine failure. SpaceX decided to swap out the second stage instead of replacing the faulty part.[509] It was SpaceX's 50th successful landing of a first stage booster, the third flight of the Dragon C112 and the last launch of the cargo Dragon spacecraft.								
	18 March 2020, 12:16[510]	F9 B5 ♺ B1048.5	KSC, LC-39A	Starlink 5 v1.0 (60 satellites)	15,600 kg (34,400 lb)[5]	LEO	SpaceX	Success	Failure (drone ship)

Appendix (slide 3):Thanks to . . .

- The IBM instructional team for the development of the Data Science Professional Certificate Program;
- The IBM Forum mentors for their instructional guidance and quick responses to student concerns and questions;
- Fellow students in this program for their willingness to post problematic code, attempted solutions, and programmatic software glitches;
- The Coursera instructional platform for hosting this program;
- spaceX.com and wikipedia.org for providing the data sources used in this capstone project;
- and, of course, the online data science and Python communities.

*SpaceX CRS-7's disintegrating two minutes after liftoff in June, 2015

Thank you for your time and attention!

Questions?

*Liftoff of Falcon Heavy on its maiden vogage on Febuary 6, 2018

*https://en.wikipedia.org/wiki/List_of_Falcon_9_and_Falcon_Heavy_launches