Data Science Capstone Project

Name: Collin Bashore

Date: 03/09/2023

https://github.com/collinbashore

Image Source: https://wccftech.com/spacex-believes-amazons-proposed-license-rules-changes-better-for-pre-space-shuttle-era/

Outline

Executive Summary

Methodology Summary

Data was extracted from the SpaceX Wikipedia page and public SpaceX API. The column 'Class' was created to categorize true successful landings. Utilized SQL, data visualization, folium maps, and plotly dashboards to explore the data. Compiled important columns for use as features for predictive analysis. Used one hot encoding to convert all categorical variables to binary values and all numeric columns to float64 data type. GridSearchCV was used to determine the ideal parameters for machine learning models using standardized data. Displayed the accuracy rating for each model and confusion matrix for best performing model(s).

Results Summary

Four machine learning models were used for predictive analysis: Logistic Regression, Support Vector Machines, Decision Tree Classifier, and K Nearest Neighbors (KNN). All models (except for KNN – 77.78%) produced similar results around 83.33% accuracy rate. All models over predicted true successful landings and more data will be required for better model determination (dataset used had only 90 records).

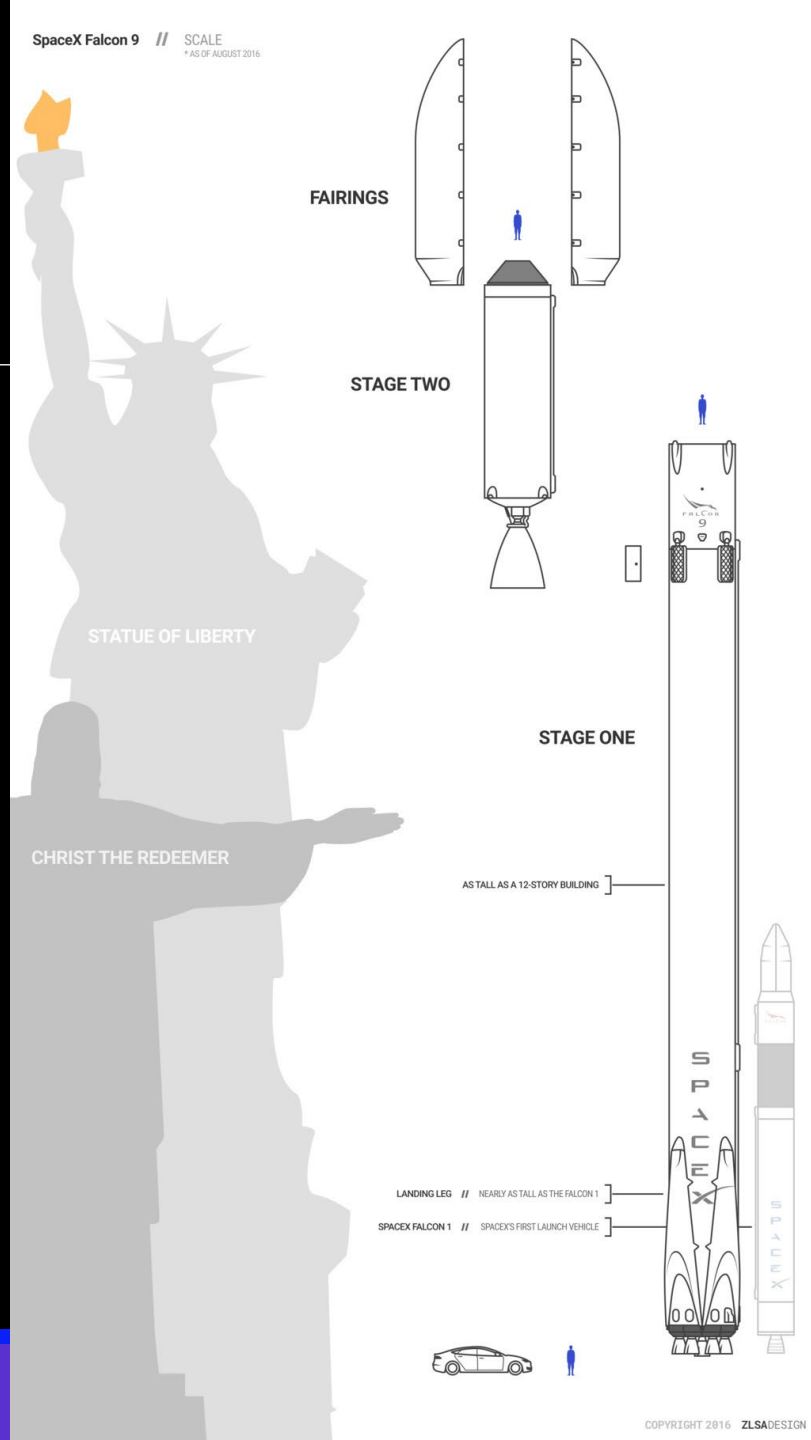
Introduction

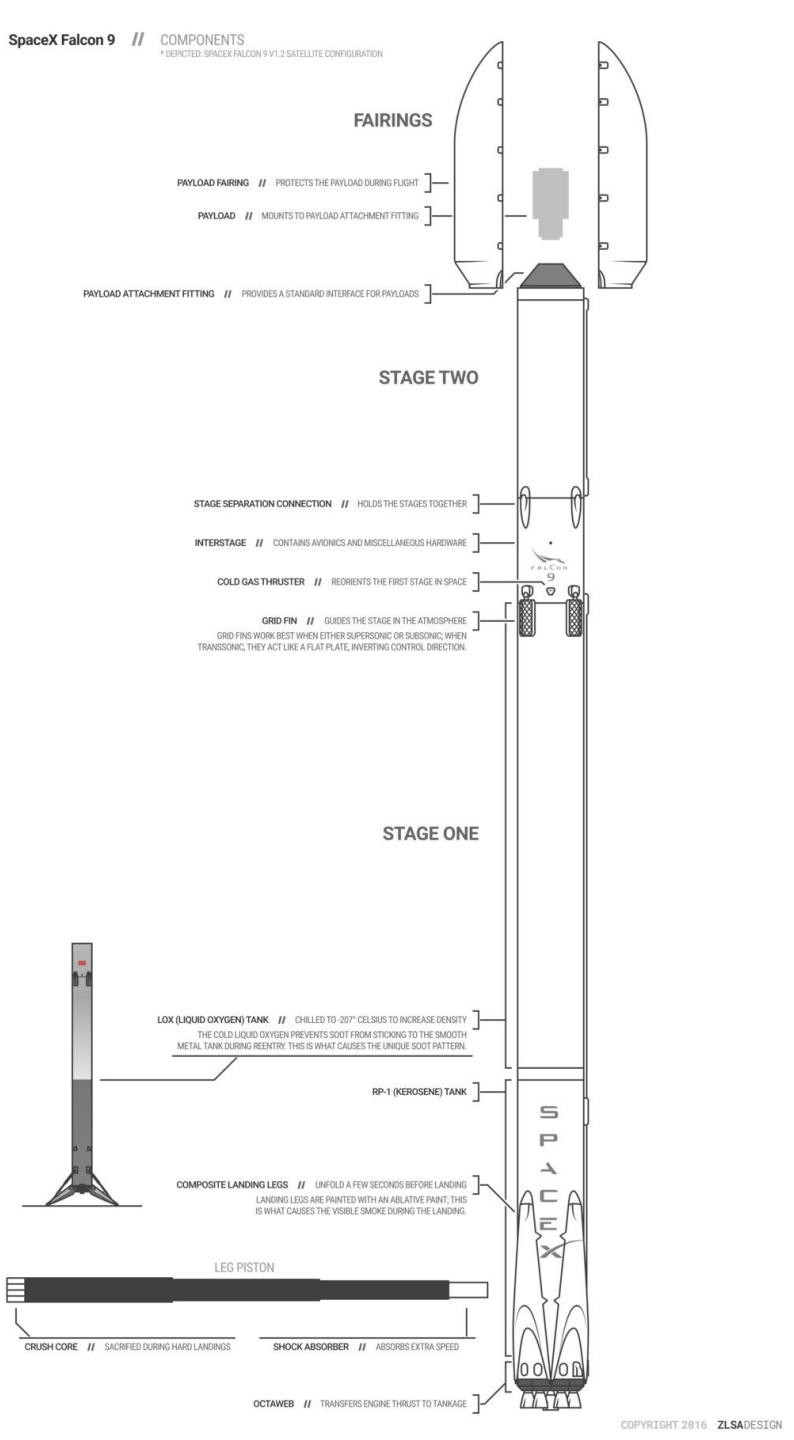
Project Background

- → During the existing commercial space age, companies are making space travel affordable for everyone
- → Space X conducts most inexpensive launches ($62 million vs. $165 million)
- → Due to recovery of rocket parts (Stage One)
- → Space Y wants to compete with Space X

Problem

→ Tasked by Space Y to train a machine learning model to predict successful Stage One recovery





Images sourced from https://zlsadesign.com/

Methodology

- **Data collection methodology:**
  - Collected and combined data utilizing public Space X API and scraping Space X Wikipedia page
- **Perform data wrangling**
  - Classified true landings as successful (class=1) and unsuccessful (class=0)
- **Perform exploratory data analysis (EDA) using visualization and SQL**
- **Perform interactive visual analytics using Folium and Plotly Dash**
- **Perform predictive analysis using classification models**
  - Tuned models using standardized data and finding the best parameters using GridSearchCV

Section 1

Methodology

Data Collection

Data Collection process involved a combination of API requests from Space X public API and webscraping data from a table in SpaceX's Wikipedia entry

The next two slides shows flowcharts of data collection from public Space X API and data collection from webscraping

Space X Data Columns:

FlightNumber, Date, BoosterVersion, PayloadMass, Orbit, LaunchSite, Outcome, Flights, GridFins, Reused, Legs, LandingPad, Block, ReusedCount, Serial, Longitude, Latitude

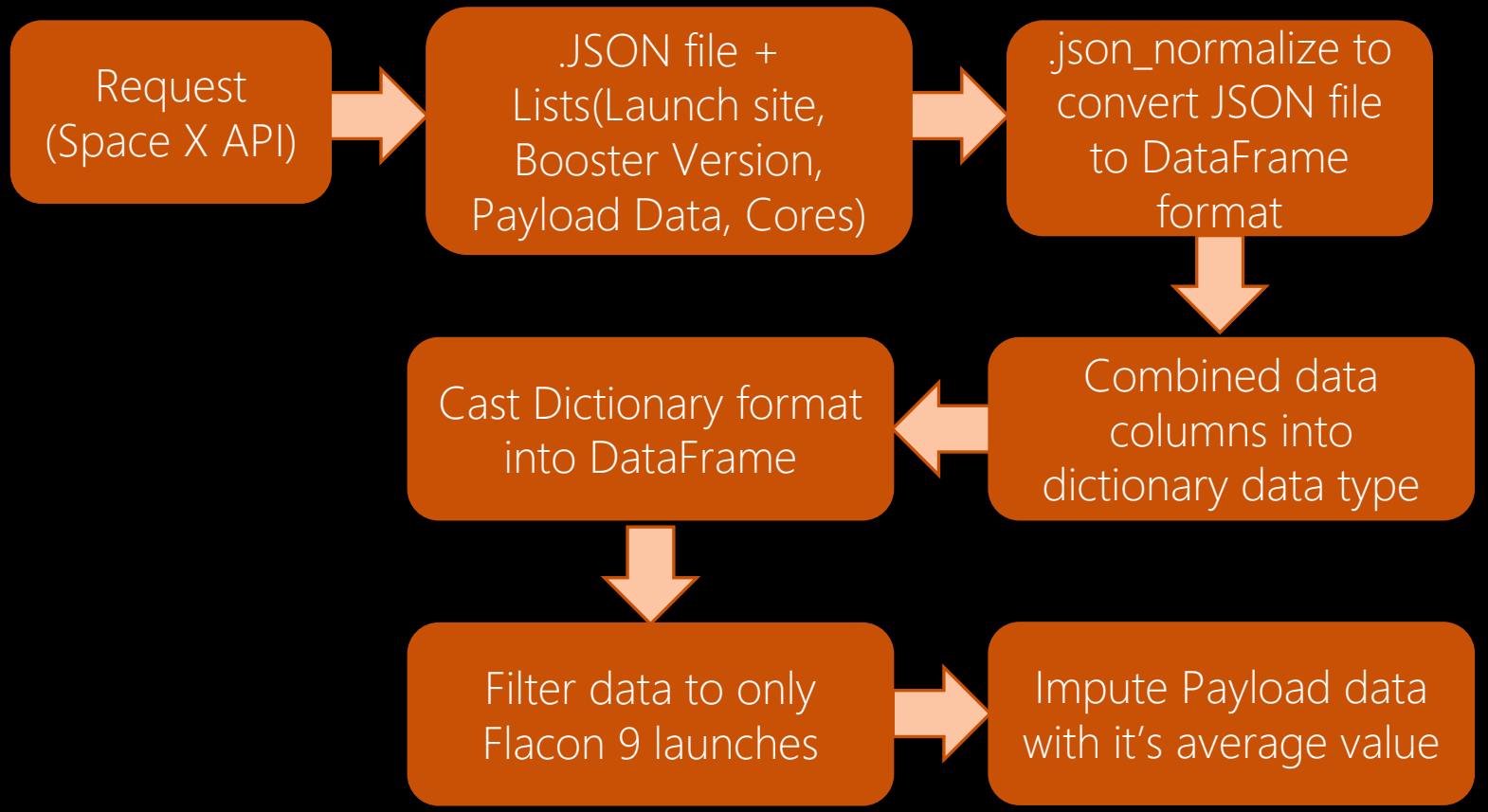
Wikipedia Webscrape Data Columns:

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

Data Collection – SpaceX API

GitHub URL

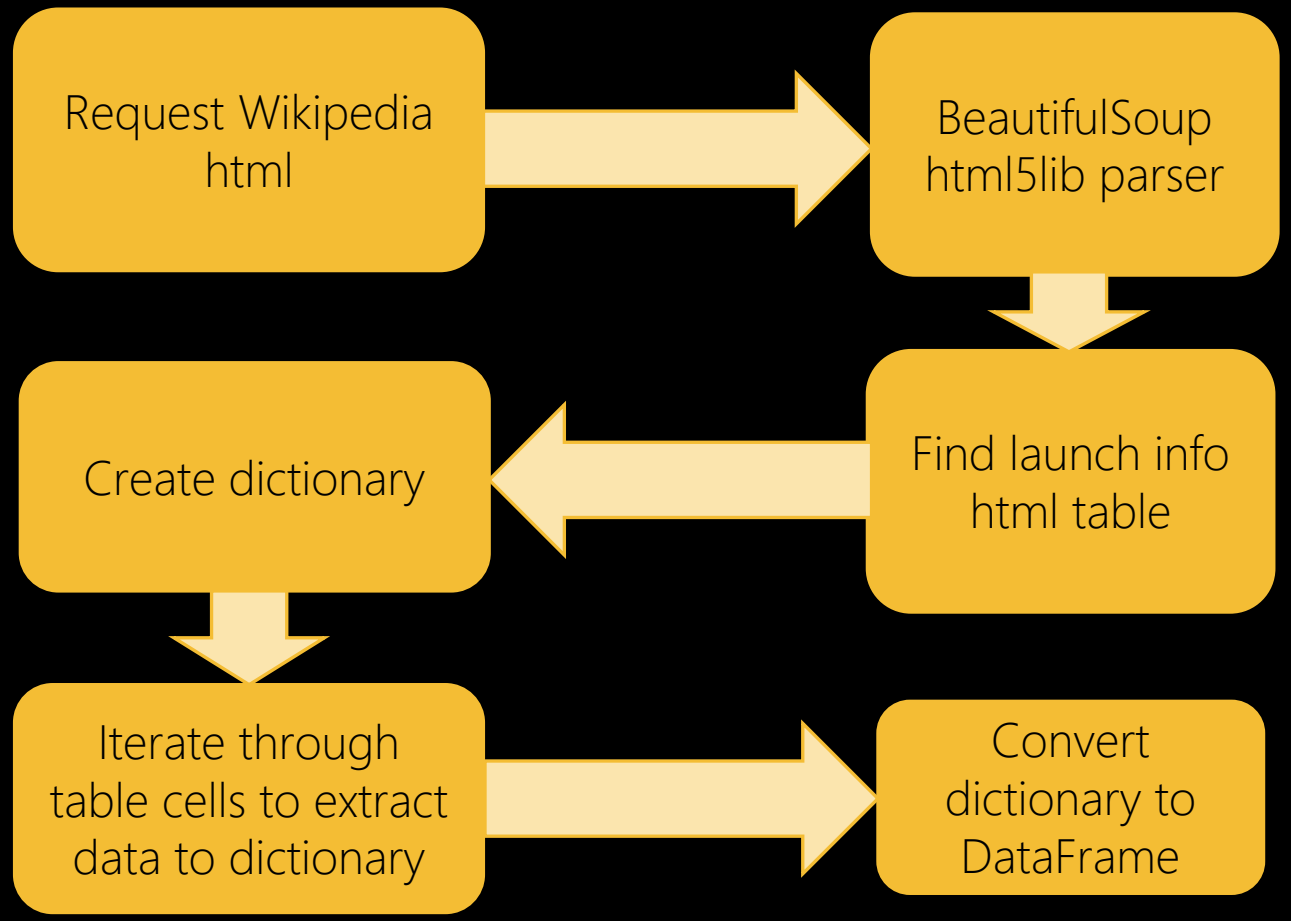
https://github.com/collinbashore/IBM-Data-Science-Professional-Certification/blob/main/10%20-%20Capstone%20Project/Week%201%20Data%20Cleaning%20-%20Webscraping%20-%20Data%20Wrangling/jupyter-labs-spacex-data-collection-api.ipynb



Data Collection – Web Scraping

GitHub URL

https://github.com/collinbashore/IBM-Data-Science-Professional-Certification/blob/main/10%20-%20Capstone%20Project/Week%201%20Data%20Cleaning%20-%20Webscraping%20-%20Data%20Wrangling/jupyter-labs-webscraping.ipynb



Data Wrangling

GitHub URL

https://github.com/collinbashore/IBM-Data-Science-Professional-Certification/blob/main/10%20-%20Capstone%20Project/Week%201%20Data%20Cleaning%20-%20Webscraping%20-%20Data%20Wrangling/labs-jupyter-spacex-Data%20wrangling.ipynb

enumerate all successful and unsuccessful mission outcomes

Create set of bad outcomes where mission outcome is "False" or "None"

Create training label column 'Class' with landing outcomes where landing_class = 0 if bad_outcome and landing_class = 1 if successful outcome

Value Mapping:

Class = 1

True ASDS, True RTLS, and True Ocean

Class = 0

None None, False ASDS, None ASDS, False Ocean, False RTLS

EDA with SQL

- Installed sqlalchemy and python-sql integration

```
[1]: !pip install sqlalchemy
     !pip install python-sql
     !pip install ipython-sql
```

- Connected to sqlite database

```
[2]: %load_ext sql

[3]: import csv, sqlite3

     con = sqlite3.connect("my_data2.db")
     cur = con.cursor()

[4]: %sql sqlite:///my_data2.db

[5]: import pandas as pd
     df = pd.read_csv("SPACEX-edited.csv")
     df.to_sql("SPACEX", con, if_exists='replace', index=False,method="multi")
```

SQL Queries Performed

- → List of Launch Site Names (Task 1)
- → First five Launch sites beginning with 'CCA' (Task 2)
- → Total and Average Payload Mass for Customers and Booster Versions (Tasks 3 and 4)
- → Year first successful landing outcome pad was achieved (Task 5)
- → List of Booster Names with successful drone ship and payload mass of 4000 to 6000 kg (Task 6)
- → Total number of successful and unsuccessful mission outcomes (Task 7)
- → Booster versions carried max payload mass (Task 8)
- → Month, Failure Landing Outcomes, Booster Versions, and Launch Site for Drone Ship (Task 9)
- → Count and rank of successful landing outcomes (Task 10)

GitHub URL

https://github.com/collinbashore/IBM-Data-Science-Professional-Certification/blob/main/10%20-%20Capstone%20Project/Week%202%20Exploratory%20Data%20Analysis/jupyter-labs-eda-sql-coursera_sqllite(1).ipynb

EDA with Data Visualization

Variables used for plots:

→ Flight Number, Payload Mass, Launch Site, Orbit, Class, Year

Plots created:

→ **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**

→ **Success Yearly Trend**

Plot types

→ Scatter plots
→ Line charts
→ Bar plots

Used to compare relationships between variable to decide if the relationships between each variable exist so that the variables are used for training the machine learning models used for predictive analytics

GitHub URL

https://github.com/collinbashore/IBM-Data-Science-Professional-Certification/blob/main/10%20-%20Capstone%20Project/Week%202%20Exploratory%20Data%20Analysis/jupyter-labs-eda-dataviz.ipynb

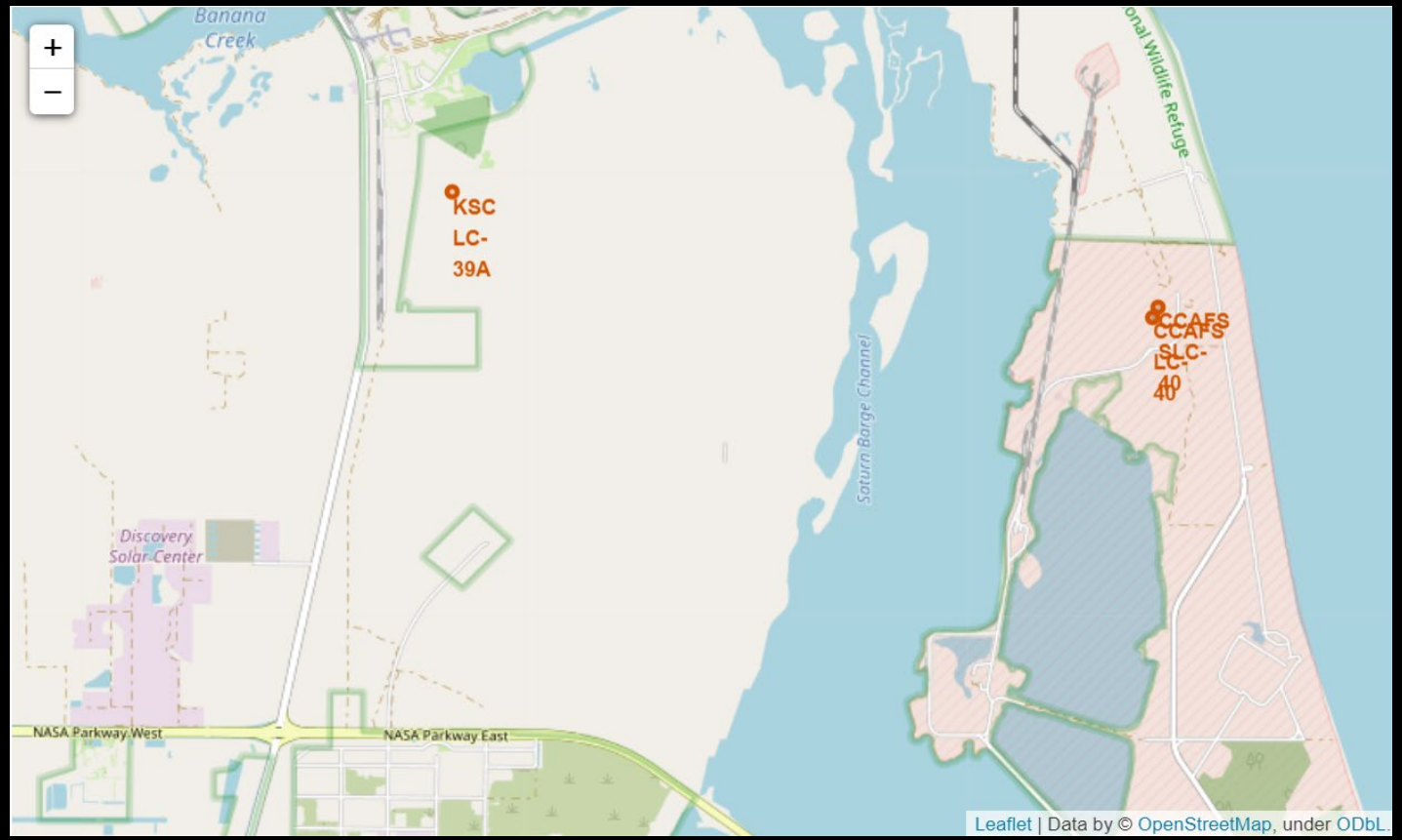
Build an Interactive Map with Folium

- Map objects in Folium such as markers, circles, and lines locates the launch sites, highlights successful and unsuccessful landings, and proximity to the following key locations: Railway, Coastline, City, Highway

- Purpose of having those map objects helps us understand why the launch sites are located in those specific locations and visualizes all successful landings relative to key locations

GitHub URL

https://github.com/collinbashore/IBM-Data-Science-Professional-Certification/blob/main/10%20-%20Capstone%20Project/Week%203%20Visual%20Analytics%20and%20Interactive%20Dashboards/lab_jupyter_launch_site_location.ipynb



Build a Dashboard with Plotly Dash

- Dashboard includes a pie chart, range slider, and a scatter plot

- Pie chart shows the distribution of successful landings across all launch sites and can be selected to visualize individual launch site success rates

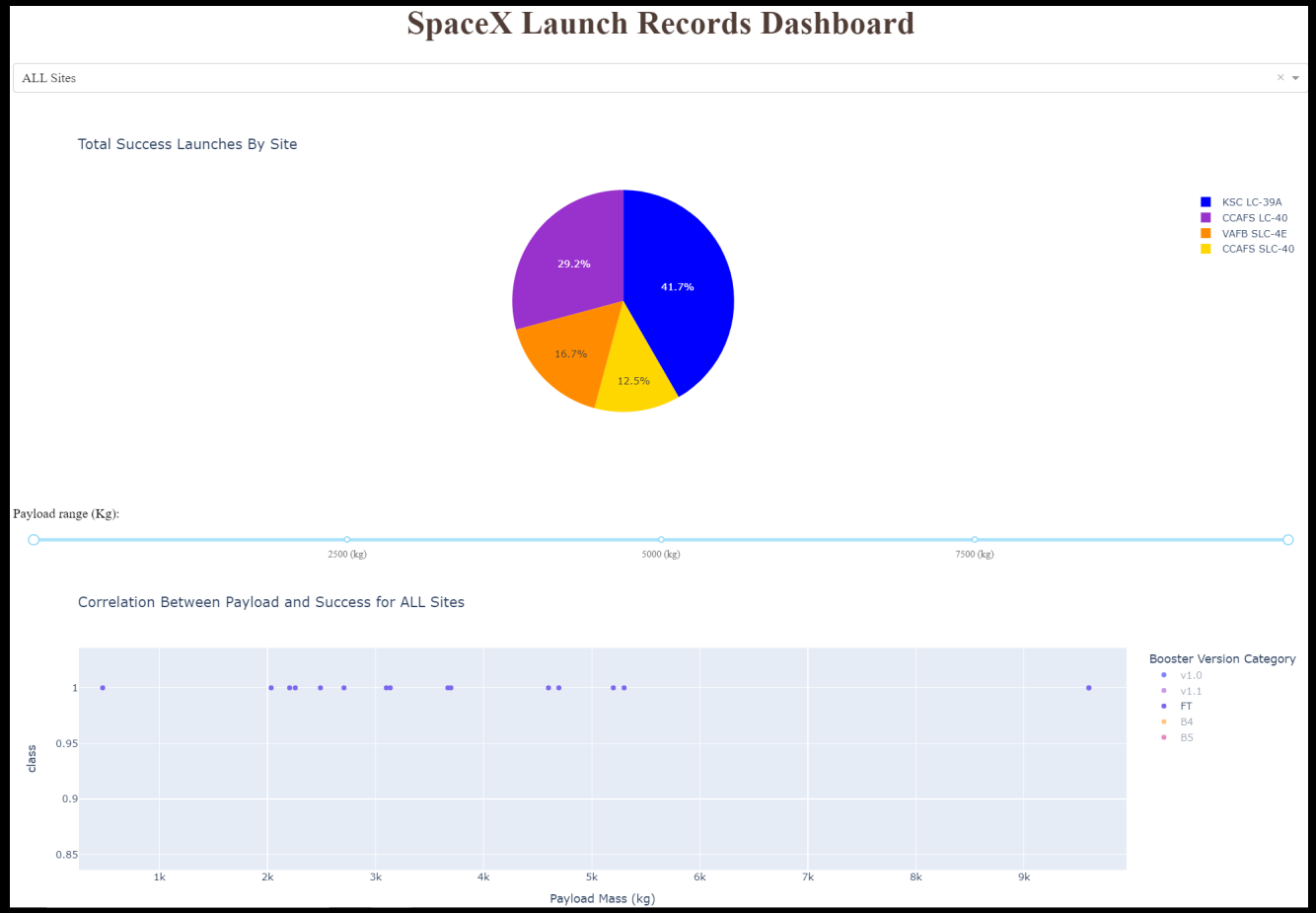
  - Two inputs: All sites or individual launch sites

- Scatter plot shows how success varies across launch sites, payload mass, and booster version category

  - Two inputs: All sites or individual sites and payload mass on interactive slider between 0 and 10,000 kg

GitHub URL

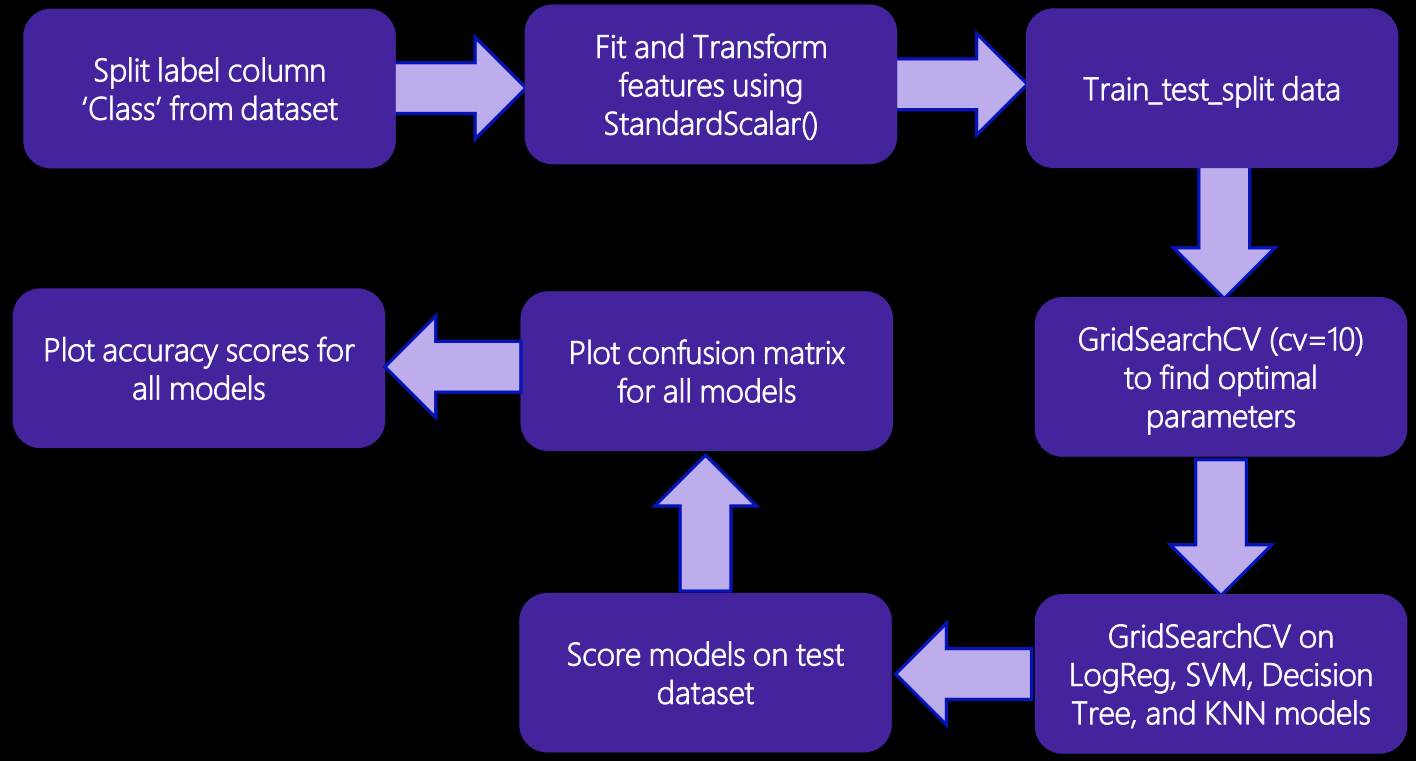
https://github.com/collinbashore/IBM-Data-Science-Professional-Certification/blob/main/10%20-%20Capstone%20Project/Week%203%20Visual%20Analytics%20and%20Interactive%20Dashboards/spacex_dash_app.py



Predictive Analysis (Classification)

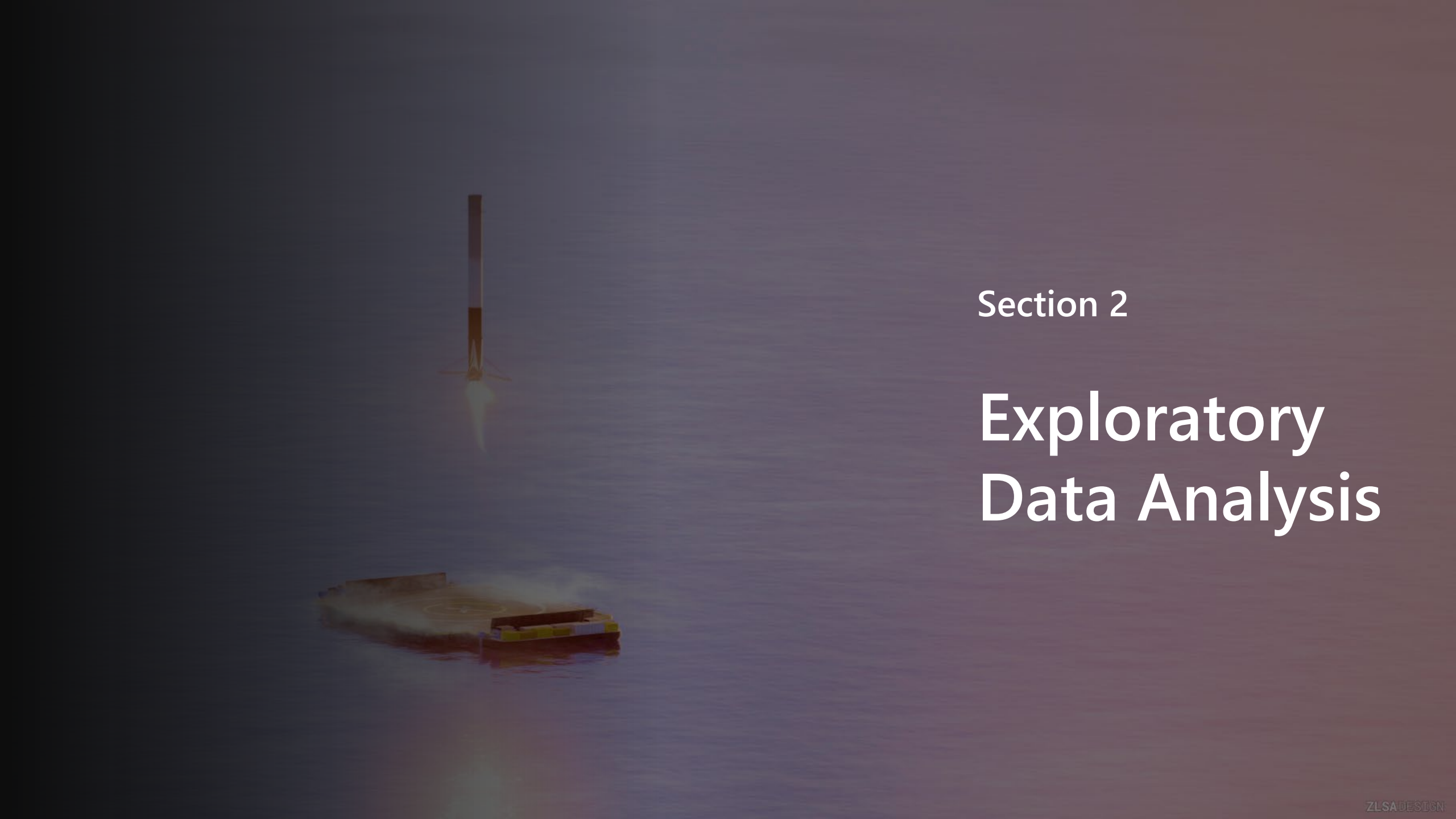
GitHub URL

https://github.com/collinbashore/IBM-Data-Science-Professional-Certification/blob/main/10%20-%20Capstone%20Project/Week%204%20Predictive%20Analytics%20(Classification)/SpaceX_Machine%20Learning%20Prediction_Part_5.ipynb



Results

- **Exploratory data analysis results**
  - Orbits ES-L1, GEO, HEO, and SSO have the highest average success rates
  - There are no rockets launched for heavy payload mass (greater than 10,000 kg) for the VAFB-SLC launch site
- **Interactive analytics demo in screenshots**
  - KSC LC-39A launch site has highest success rate of 41.7%
  - FT Booster has the most successful landings (13 of 15 with payload mass range of 2,000 to 5,500 kg
- **Predictive analysis results**
  - Three models: Logistic Regression, Decision Tree, and Support Vector Machines have test accuracy rate of 83.33%
  - K Nearest Neighbors had test accuracy rate of 77.78%

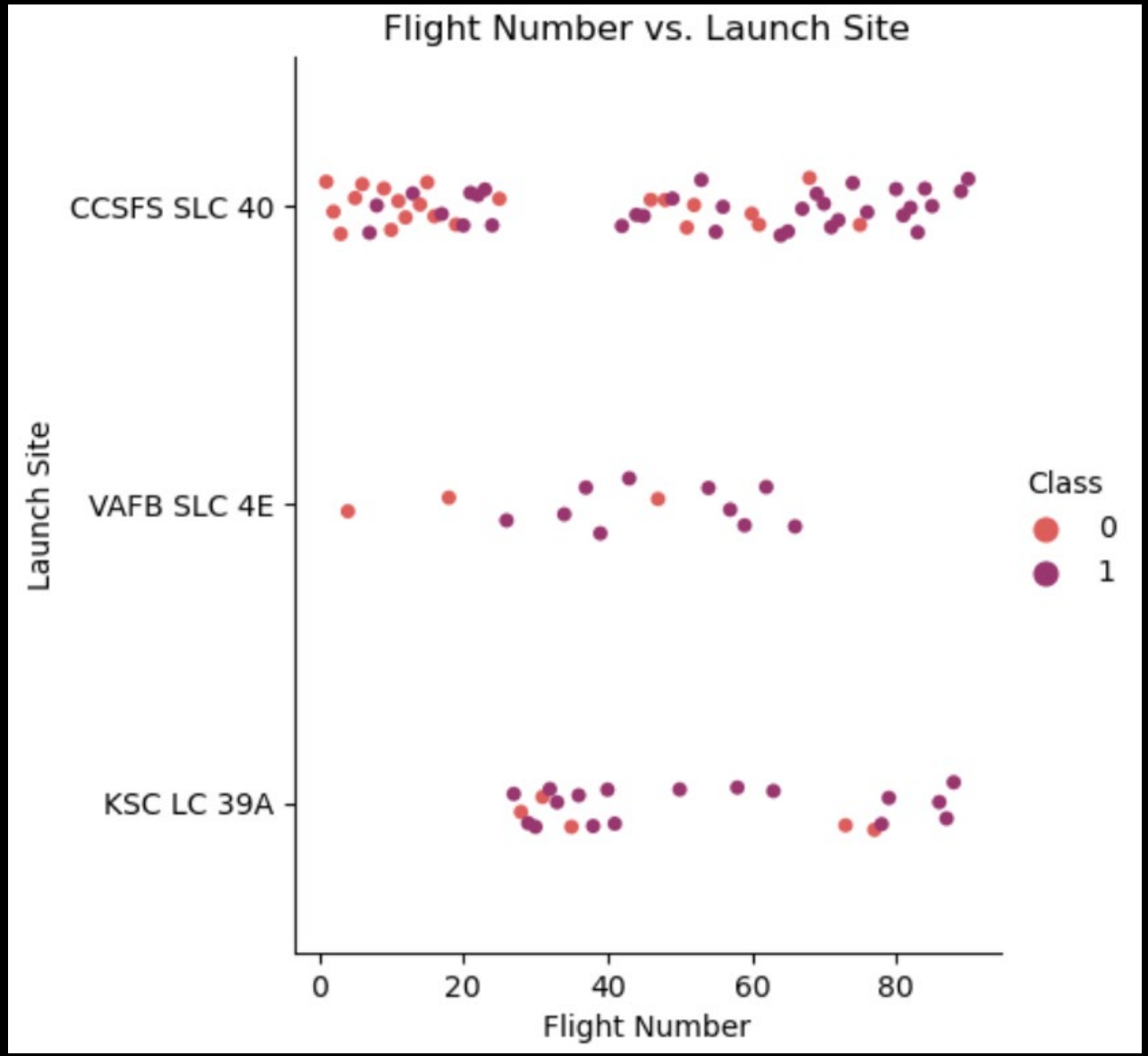


Section 2

Exploratory Data Analysis

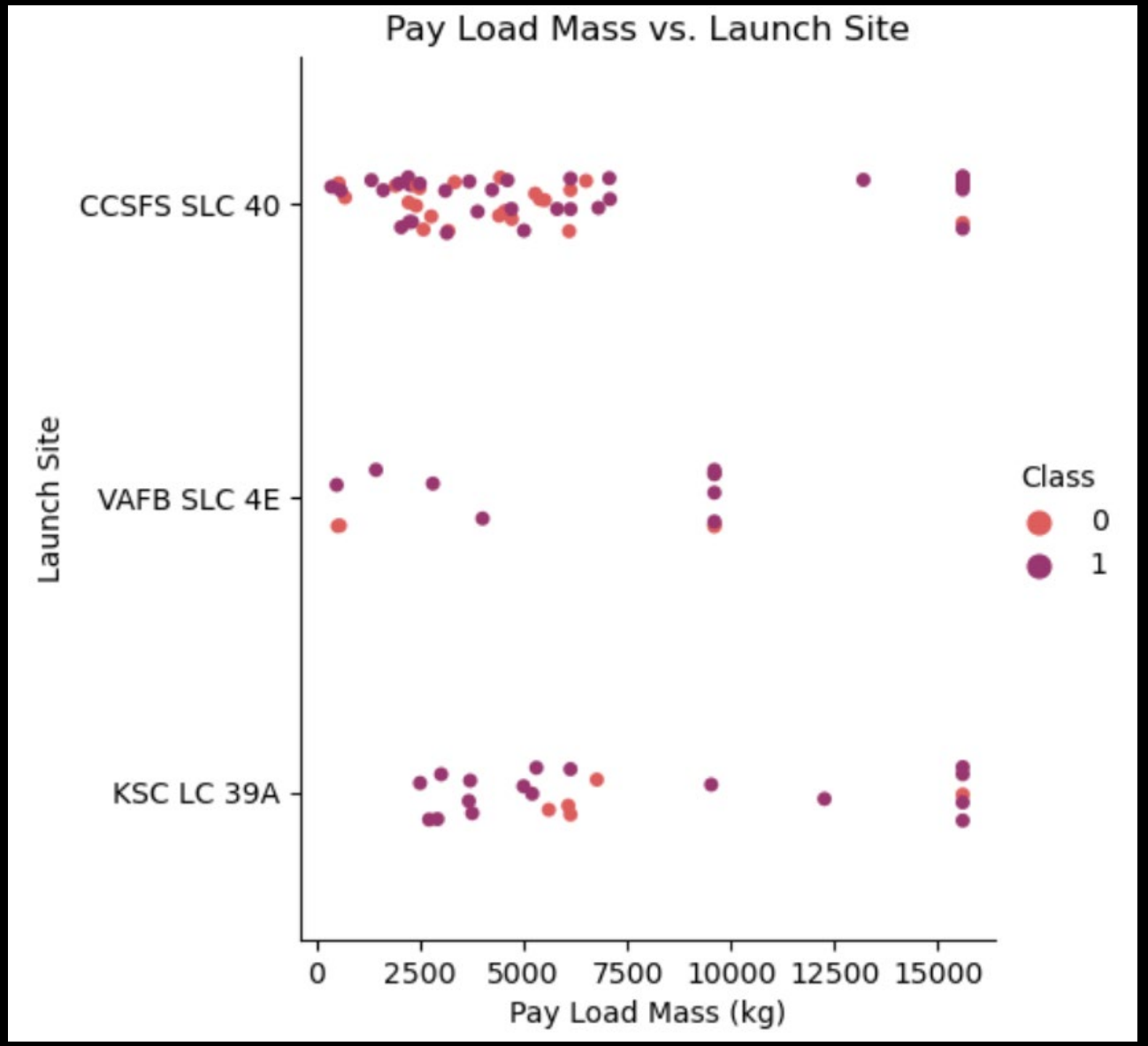
Flight Number vs. Launch Site

- Purple indicates successful launches, pink indicates unsuccessful launces
- Plot suggests an increase in success rate over time
- Possible breakthrough around Flight Number 20 where there's an increase in successful launces
- CCAFS SLC 40 has the most launches and has the most volume of launches



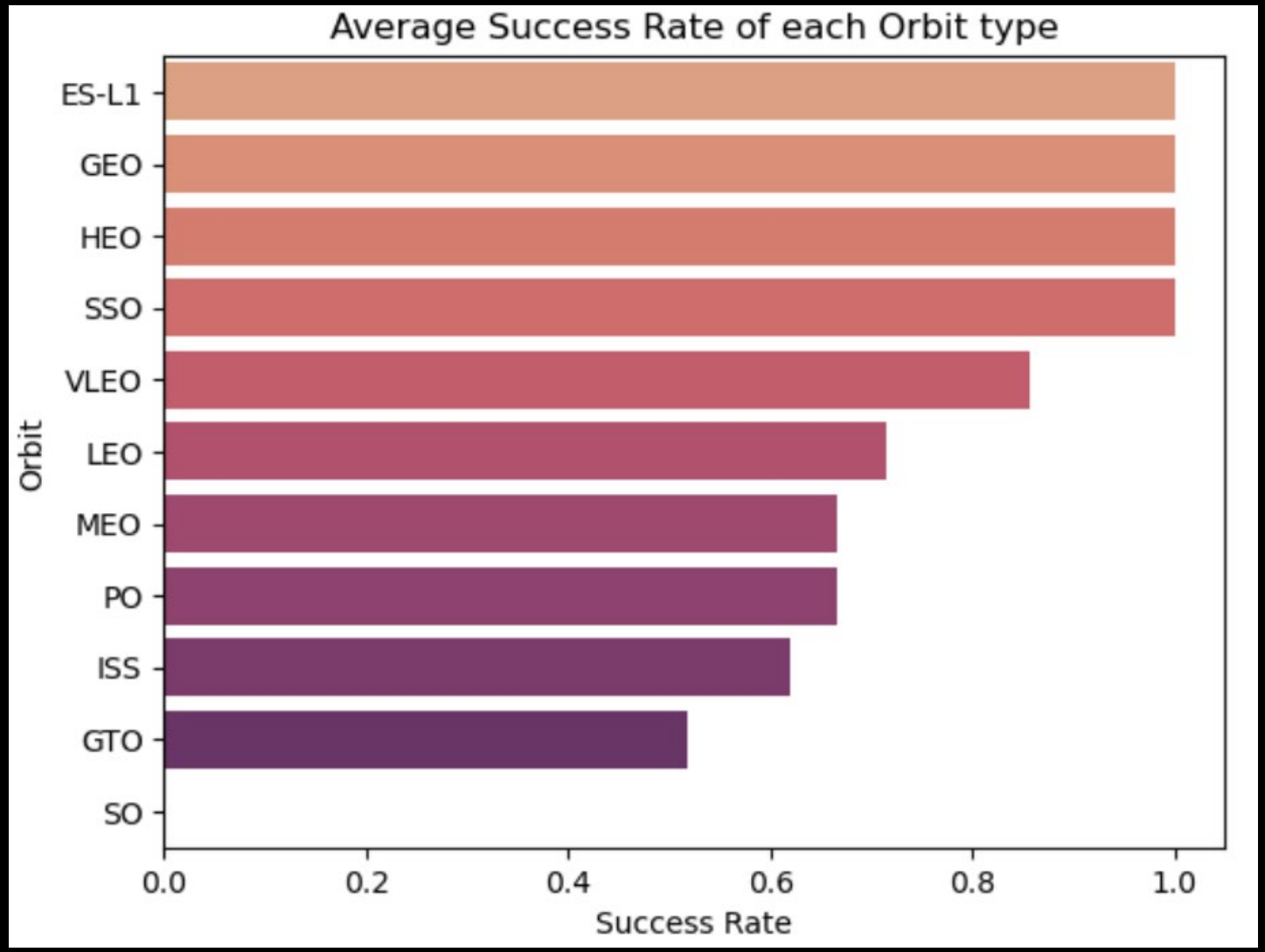
Payload Mass vs. Launch Site

- Purple indicates successful launches, pink indicates unsuccessful launces
- Payload mass mostly appears from 0 to 7,500 kg
- Each of the launch sites have different payload masses
- VAFB SLC 4E doesn't have payload masses greater than 10,000 kg



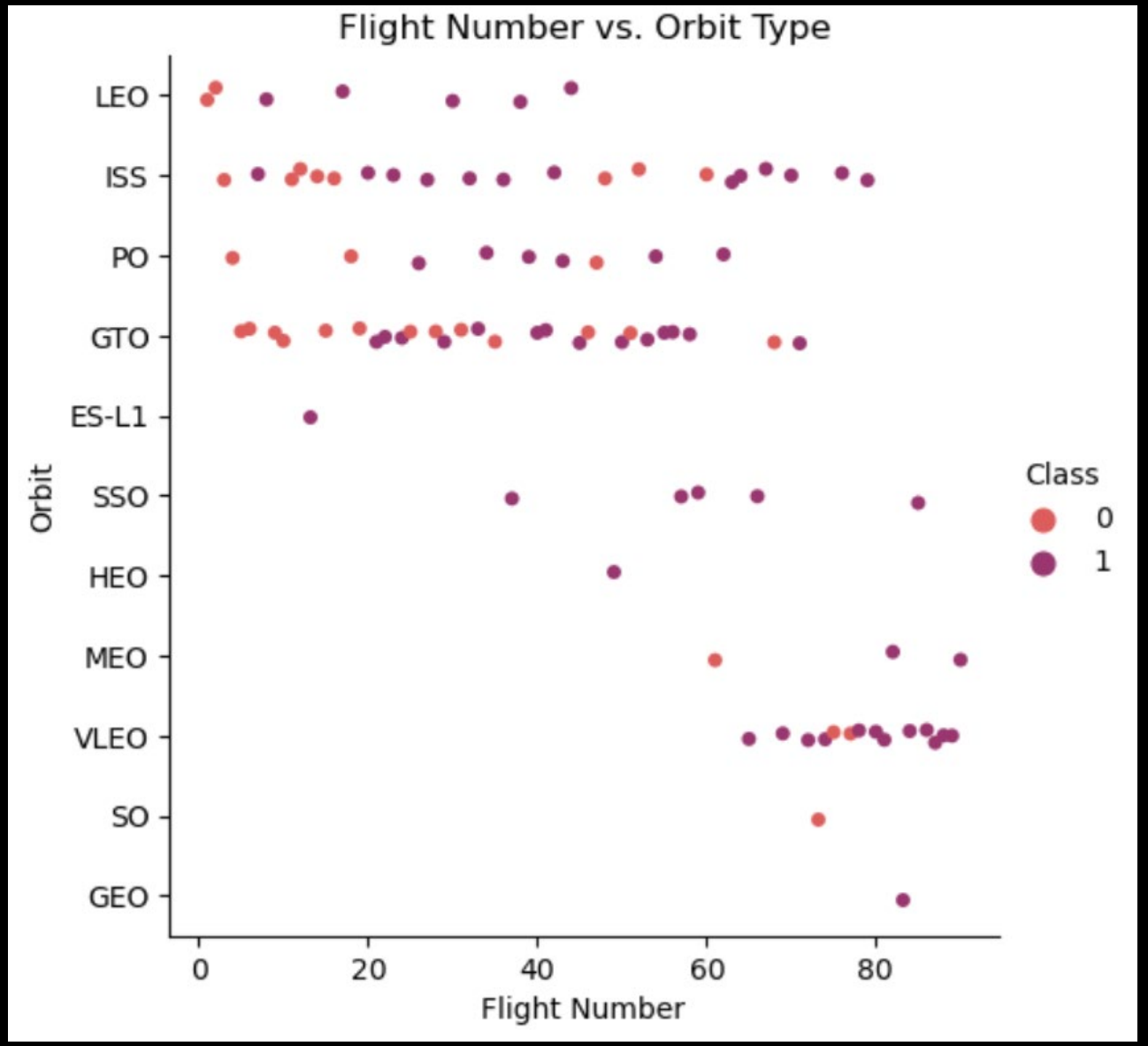
Success Rate vs. Orbit Type

- ES-L1 (1), GEO (1), HEO(1), and SSO (5) all have success rate of 1.0 (100%).
- MEO (3) and PO (9) have success rate of 0.67 (67%)
- GTO (27) has success rate of 0.5 (50%)
- SO (1) had a success rate of 0 (0%)



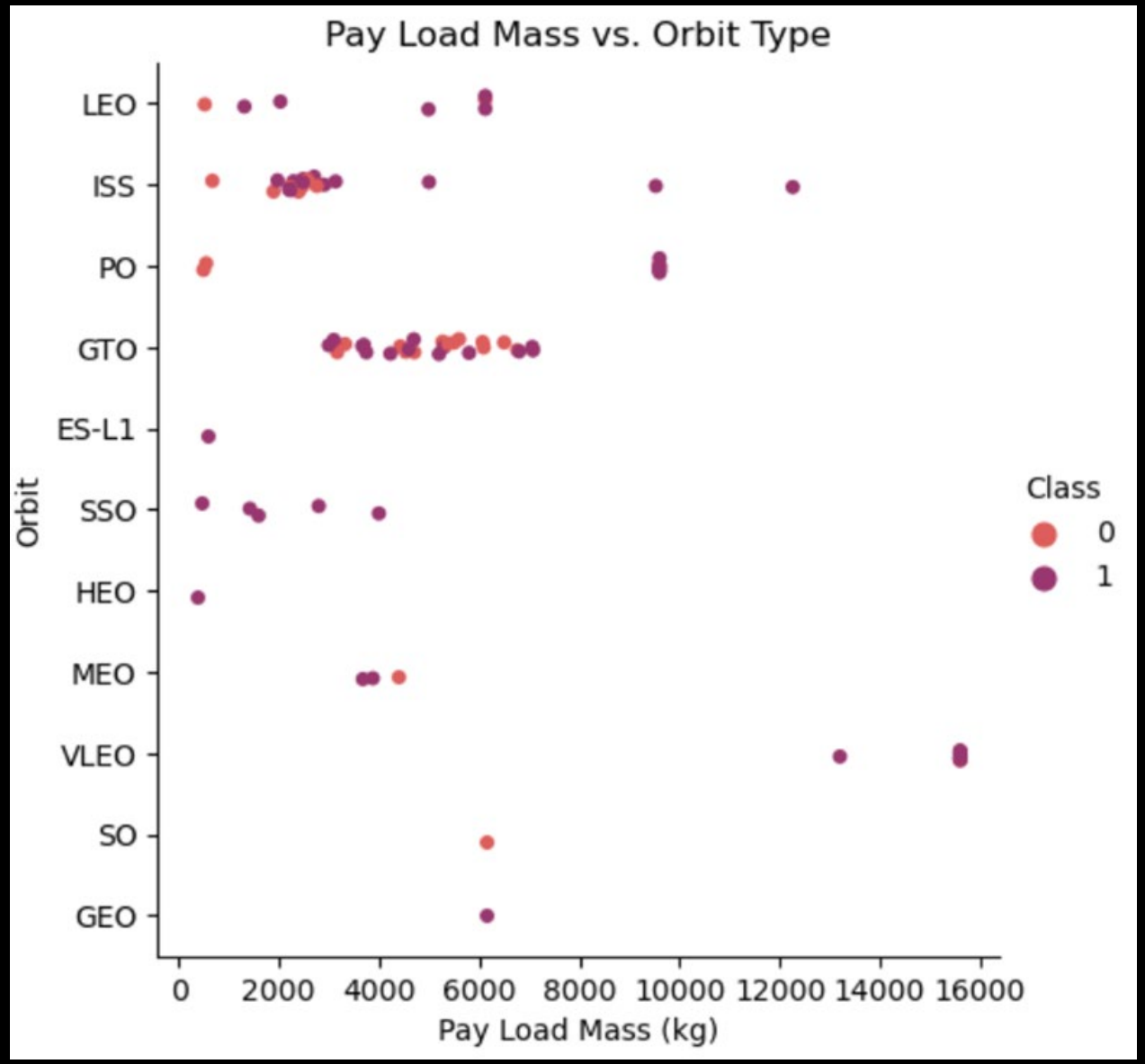
Flight Number vs. Orbit Type

- Purple indicates successful launches, pink indicates unsuccessful launces
- There seems to be more successful launches for Flight numbers greater than 60
- Space X appears to have successful launches in lower orbits or Sun-synchronous orbits



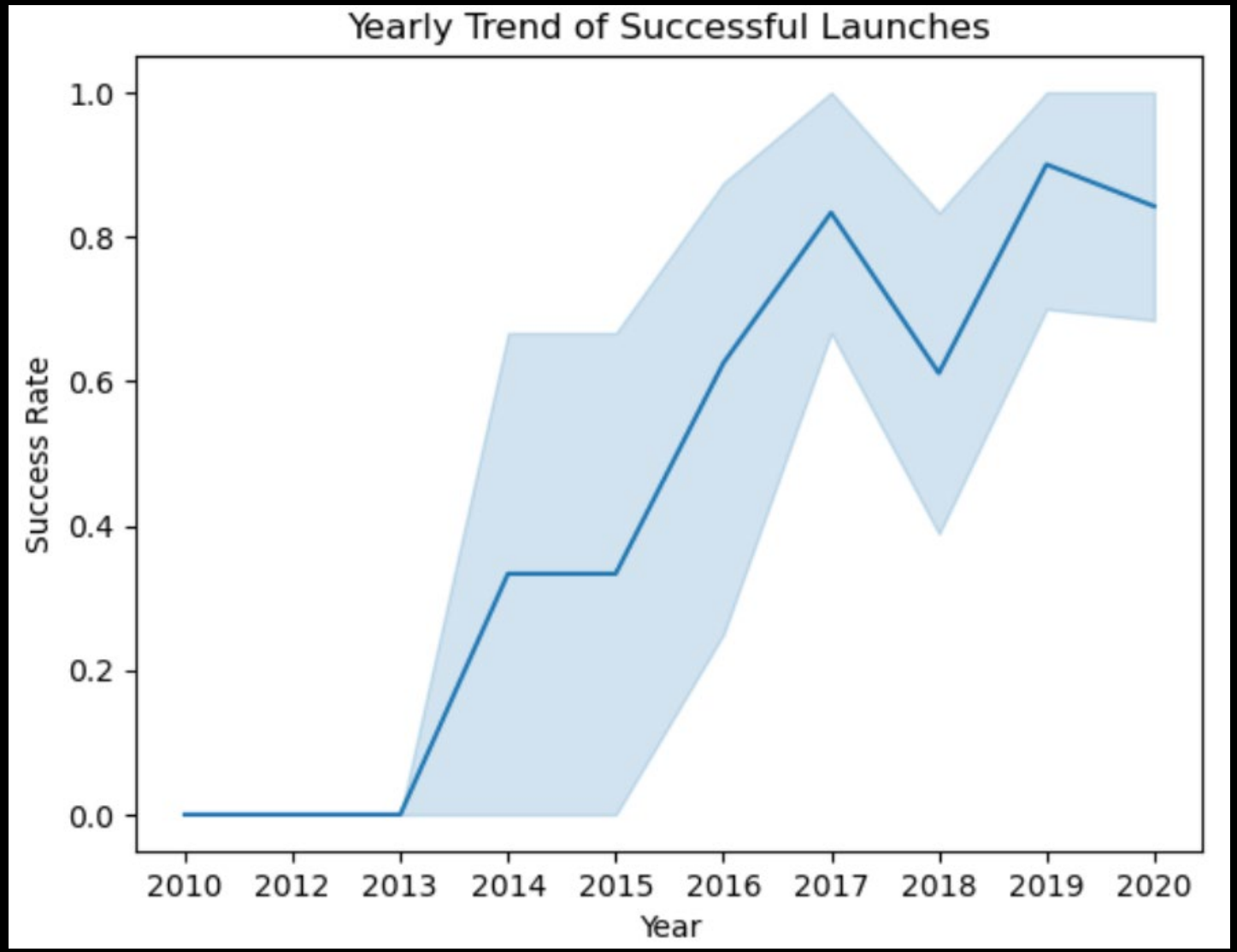
Payload Mass vs. Orbit Type

- Purple indicates successful launches, pink indicates unsuccessful launces
- LEO and SSO appear to have low payload masses
- VLEO only has playload masses at the higher range
- GTO has a higher concentration of launches with payload masses between 2,500 to 7,500 kg



Yearly Trend of Successful Launches

- Light blue shading indicates 95% confidence interval
- Number of successful launches increases over time starting in year 2013, with slight dip in 2018
- Most recent years has success rates around 80%



All Launch Site Names

- Data entry errors are highly likely for launch sites CCAFS LC-40 and CCAFS SLC-40
- Noting the possible data entry error, there's more than likely only 3 unique launch sites
  - CCAFS SLC-40, KSC LC-39A, and VAFB SLC-4E

[11]:
```
%%sql
SELECT "Launch_Site"
FROM SPACEX
GROUP BY "Launch_Site";
```

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

[11]:

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

Launch Site Names Begin with 'CCA'

Note on the two date columns:

- The original date format in the "SpaceX.csv" file is in month-day-year
- The "DATES" column has a date format year-month-day
- Both of those date columns were used for generating SQL queries
- See "Appendix" Section for how the DATES column was created

[12]:
```
%%sql
SELECT *
FROM SPACEX
WHERE "Launch_Site" LIKE 'CCA%'
LIMIT 5;
```

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

[12]:

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

Total Payload Mass

- The query below sums all the payload mass (in kg) values where NASA was the customer
- CRS strands for Commercial Resupply Services indicating that the payloads were sent to the International Space Station (ISS)

```
[13]: %%sql SELECT Customer, SUM("PAYLOAD_MASS__KG_") AS Total_Payload_Mass
      FROM SPACEX
      WHERE Customer = 'NASA (CRS)';
```

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

[13]:

Customer	Total_Payload_Mass
NASA (CRS)	45596

Average Payload Mass by F9 v1.1

- The query to the right calculates the average payload mass carried by booster version F9 v1.1
- Average payload mass of F9 v1.1 is on the low end of the payload mass range

[14]:
```
%%sql
SELECT "Booster_Version", AVG("PAYLOAD_MASS__KG_") AS Average_Payload_Mass_Kg
FROM SPACEX
WHERE "Booster_Version" = 'F9 v1.1';
```

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

[14]:

Booster_Version	Average_Payload_Mass_Kg
F9 v1.1	2928.4

First Successful Ground Landing Date

- The query returns the first successful ground pad landing date
- First grounding pad launch didn't appear until the end of 2015
- Successful launches didn't appear until 2014

```
[15]: %%sql
SELECT MIN(DATES) AS Year_First_Successful_Landing_Ground_Pad
FROM SPACEX
WHERE "Landing _Outcome" = 'Success (ground pad)';
```

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

[15]:

Year_First_Successful_Landing_Ground_Pad
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

This query returns the four booster versions that had successful drone ship landings and a payload mass between 4000 and 6000 kg non-inclusively

[16]:
```
%%sql
SELECT "Booster_Version"
FROM SPACEX
WHERE "Landing _Outcome" = 'Success (drone ship)'
    AND "PAYLOAD_MASS__KG_" BETWEEN 4001 AND 5999;
```

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

[16]:

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

Total Number of Successful and Failure Mission Outcomes

- This query calculates the total number of successful and failure mission outcomes
- The two “Success” rows could come from a data entry error noted on Slide 24. There should be a total of 99 Success Missions
- Only one launch failed in flight and the other launch had an unknown payload status

[17]:
```
%%sql
SELECT "Mission_Outcome", COUNT("Mission_Outcome") AS Total_Number_of_Outcomes
FROM SPACEX
GROUP BY "Mission_Outcome";
```

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

[17]:

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

Boosters Carried Maximum Payload

- The query returns the booster versions that carry a maximum payload mass of 15,600 kg
- All boosters fall in the “F9 B5 B10xx.x” variety
- There appears that the payload mass correlates with the booster version used.

[18]:
```
%%sql
SELECT DISTINCT("Booster_Version"), "PAYLOAD_MASS__KG_"
FROM SPACEX
WHERE "PAYLOAD_MASS__KG_" IN (SELECT MAX("PAYLOAD_MASS__KG_")
                              FROM SPACEX)
ORDER BY "Booster Version";
```

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

[18]:

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

2015 Launch Records

- This query returns the Month number, landing outcome, booster version, and launch site for the 2015 launches where Stage One failed to land on a drone ship

- Two of those occurrences were at the same launch site

Note: SQLite does not support monthnames function. So SUBSTR(Date, 4, 2) as month to get the months and SUBSTR(Date,7,4)='2015' for year.

[19]:
```
%%sql
SELECT CASE SUBSTR(Date,4,2)
            WHEN '01' THEN 'January'
            WHEN '02' THEN 'February'
            WHEN '03' THEN 'March'
            WHEN '04' THEN 'April'
            WHEN '05' THEN 'May'
            WHEN '06' THEN 'June'
            WHEN '07' THEN 'July'
            WHEN '08' THEN 'August'
            WHEN '09' THEN 'September'
            WHEN '10' THEN 'October'
            WHEN '11' THEN 'November'
       ELSE 'December'END AS Month,
        "Landing _Outcome",
        "Booster_Version",
        "Launch_Site"
FROM SPACEX
WHERE SUBSTR(Date,7,4) = '2015'
    AND "Landing _Outcome" = 'Failure (drone ship)';
```

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

[19]:

Month	Landing _Outcome	Booster_Version	Launch_Site
January	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
April	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

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

- The query returns a list of successful landings between 2010-06-04 and 2017-03-20
- Two types of successful landing outcomes
  - Ground pad
  - Drone ship
- There are a total of 8 successful landings during that time period

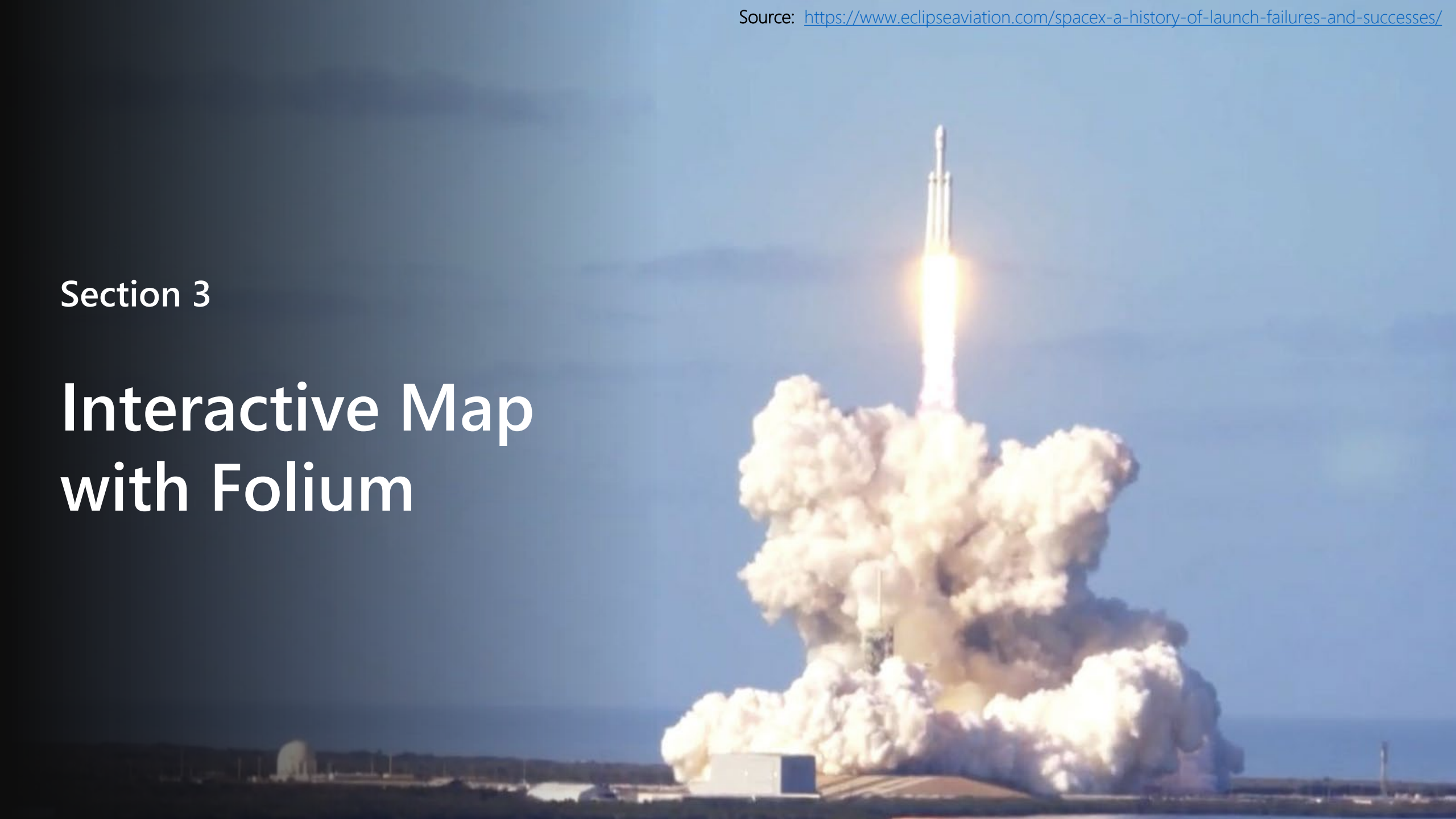
[20]:
```
%%sql
SELECT "Landing _Outcome",
       COUNT(*) AS Total_Success_Landing_Outcomes
FROM SPACEX
WHERE "Landing _Outcome" LIKE 'Success%'
    AND substr(Date,7,4) || substr(Date,4,2) || substr(Date,1,2)
    BETWEEN '20100604'AND '20170320'
GROUP BY "Landing _Outcome"
ORDER BY Total_Success_Landing_Outcomes DESC;
```

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

[20]:

Landing _Outcome	Total_Success_Landing_Outcomes
Success (drone ship)	5
Success (ground pad)	3

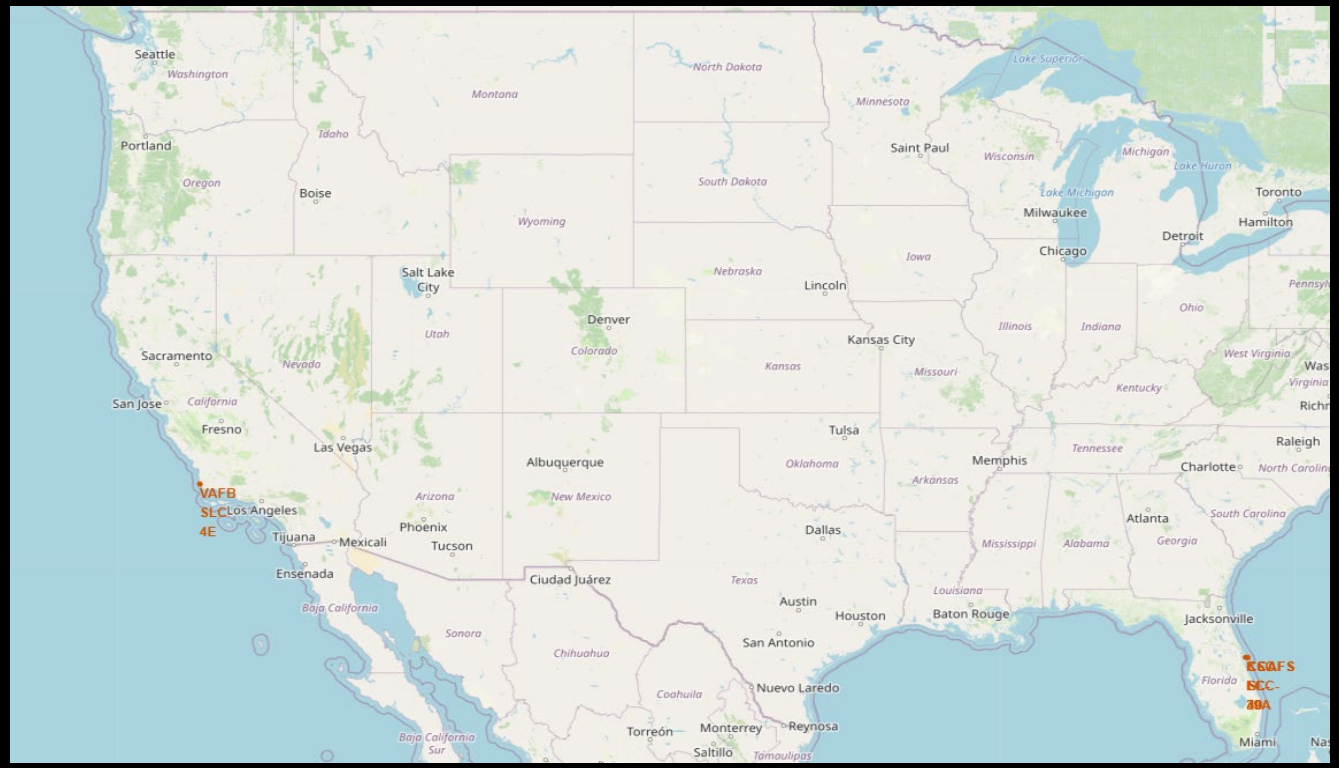
Source: https://www.eclipseaviation.com/spacex-a-history-of-launch-failures-and-successes/



Section 3

Interactive Map with Folium

Launch Site Locations



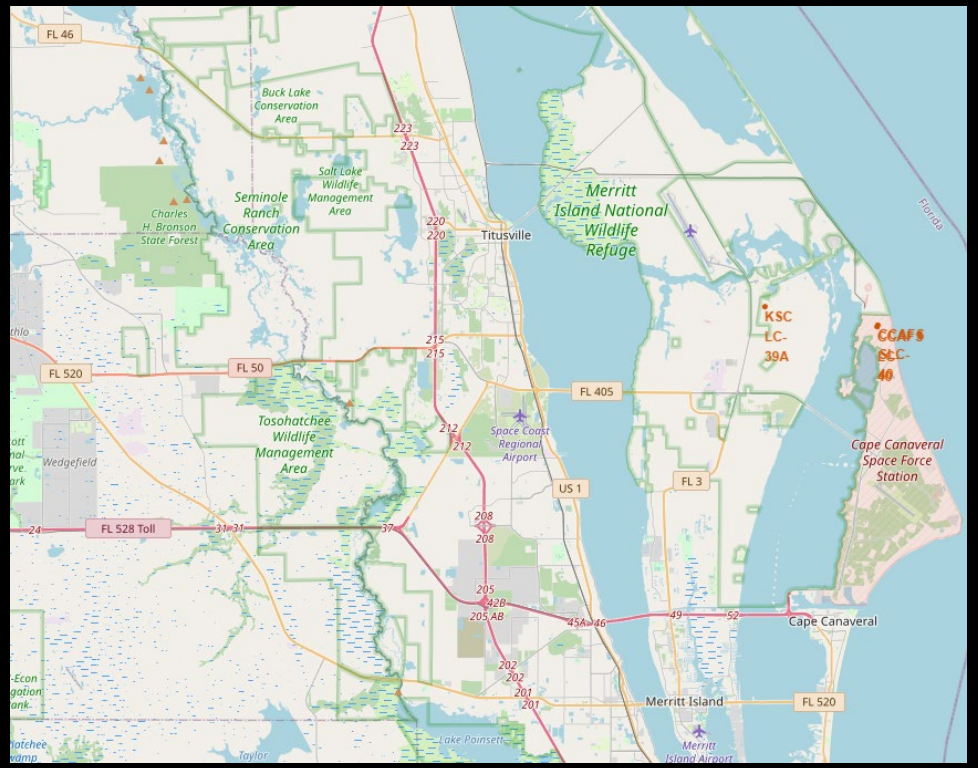
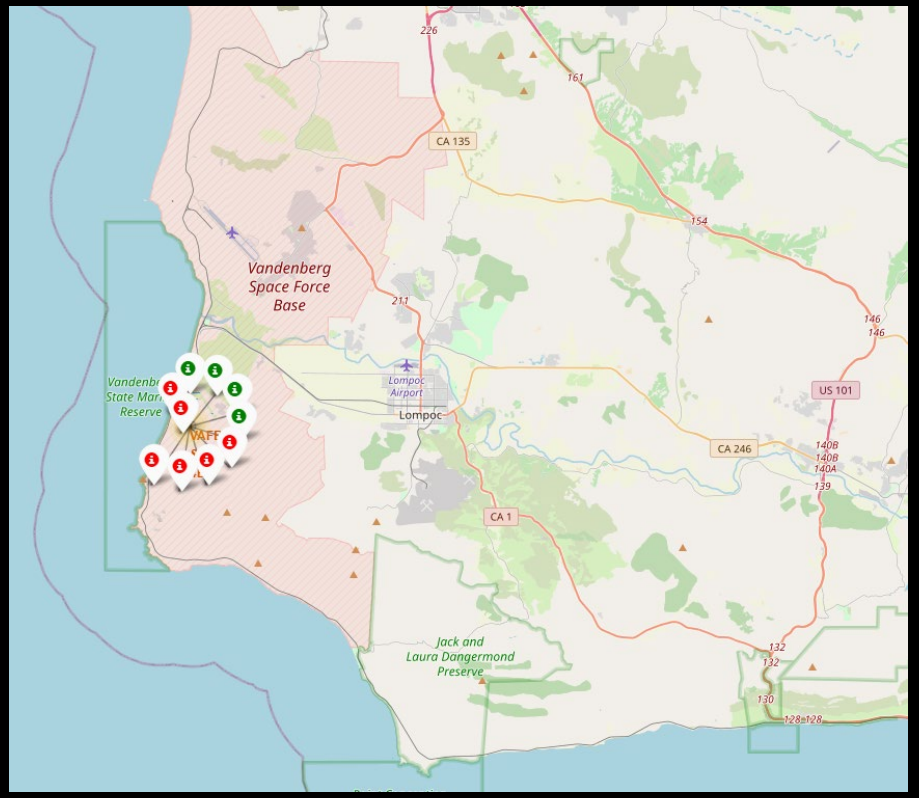
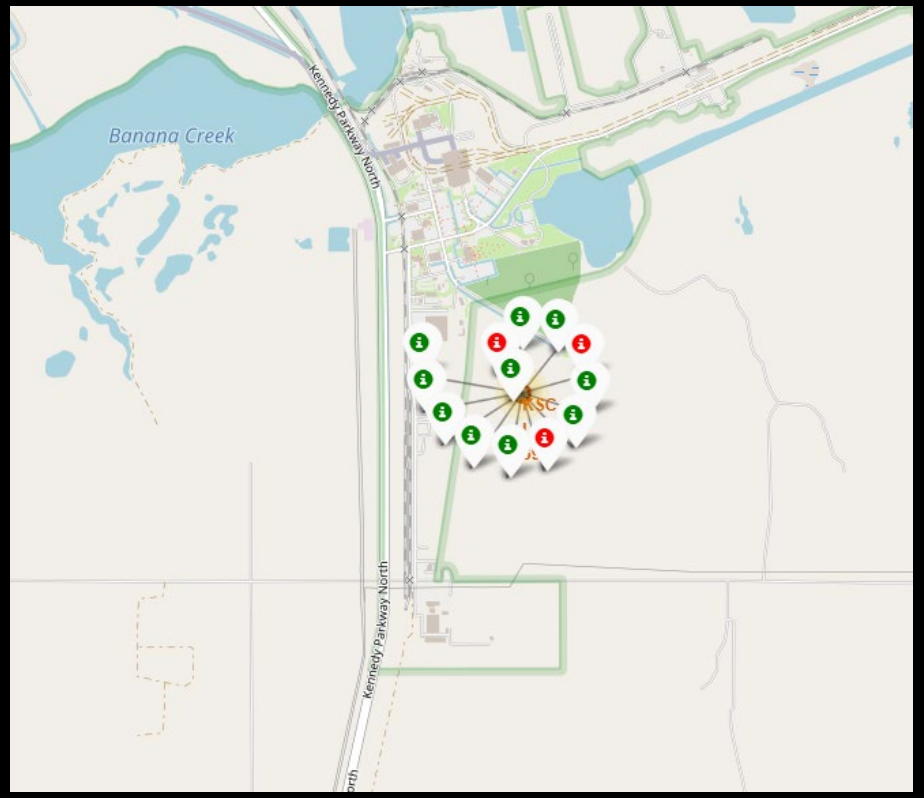


Image on the left shows all the launch site locations on the U.S. map. The image on the right shows the two launch site locations in Florida. All of the launch sites are near the ocean and far from cities.

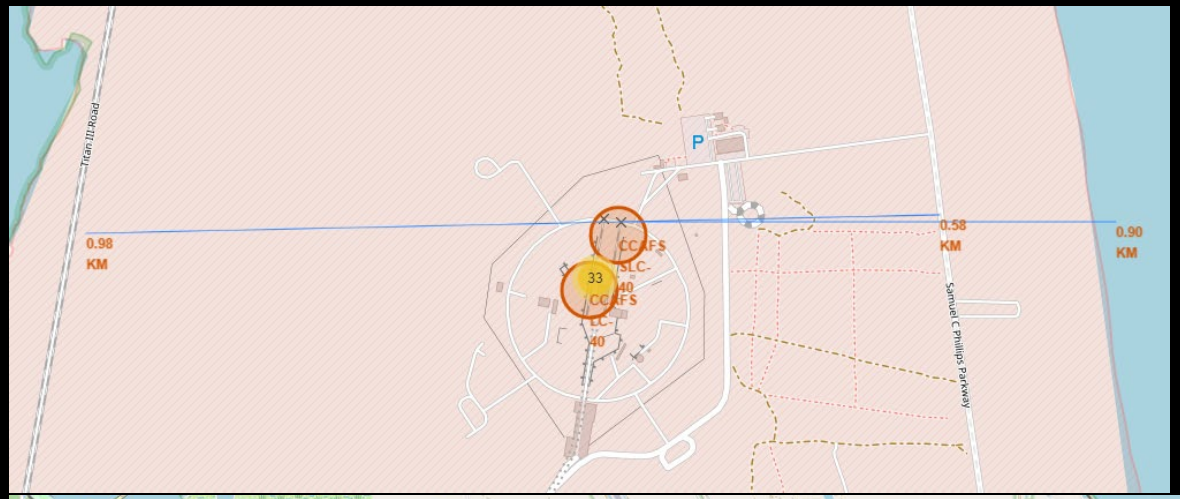
Color-Coded Launch Markers





Marker clusters on Folium map are clickable and are displayed as successful landing (green icon) and failed landing (red icon). Image on the left (VAFB-SLC-4E) shows 4 successful and 6 failed landings while image on right (KSC-LC-39A) shows 10 successful and 3 unsuccessful landings.

Key Location Proximities



- Image on the top shows the distances from CCFAS-SLC-40 launch site to the following key locations: Highway, Railway, and Coastline
  - Distance to nearest coastline: 0.90 km
  - Distance to nearest railway: 0.98 km
  - Distance to nearest highway: 0.58 km

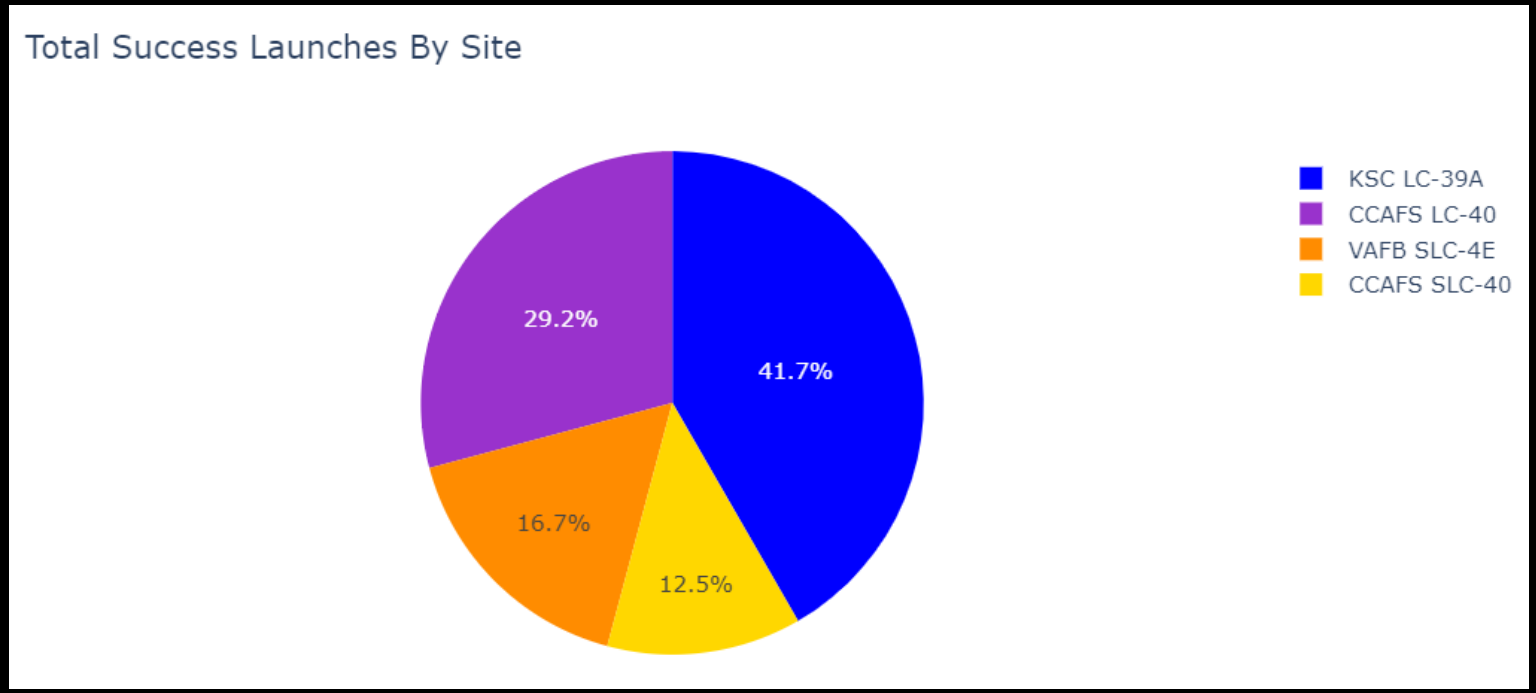
- Bottom image shows distance from CCFAS-SLC-40 launch site to nearest city of Titusville, FL (23.20 km).
- Launch sites are far from cities so that failed launches can land into ocean avoiding rocket parts falling on densely populated areas.

Section 4

Build a Dashboard with Plotly Dash

Source: https://www.eclipseaviation.com/spacex-a-history-of-launch-failures-and-successes/

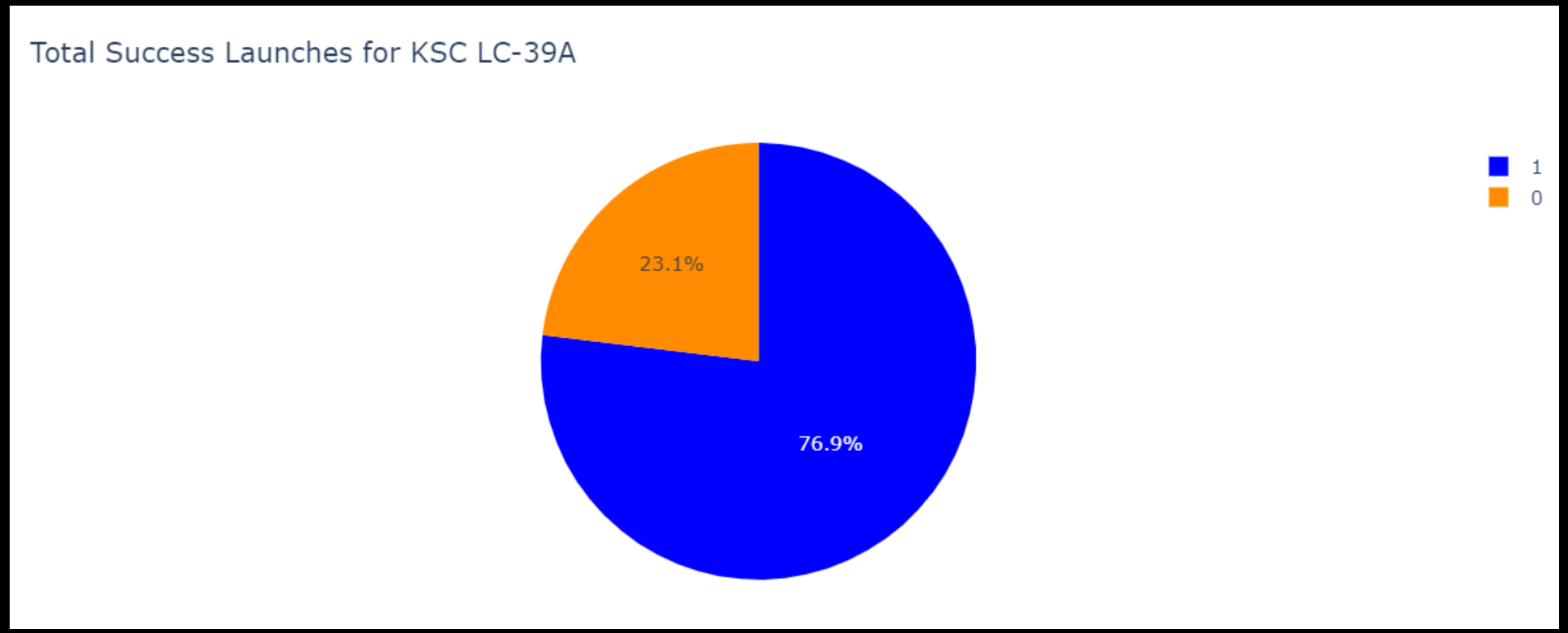
Successful Landings Across Launch Sites



Above is the distribution of successful landings across launch sites. Since CCAFS LC-40 is the older name for CCAFS SLC-40, the number of successful landings for CCAFS SLC-40 (total of 41.7%) and KSC LC-39A (41.7%) have roughly the same amount of successful landings. VAFB SLC-4E has the smallest area of successful landings of 16.7%. This small amount of successful landings in the west coast may be due to smaller sample size and difficulty of launching rockets.

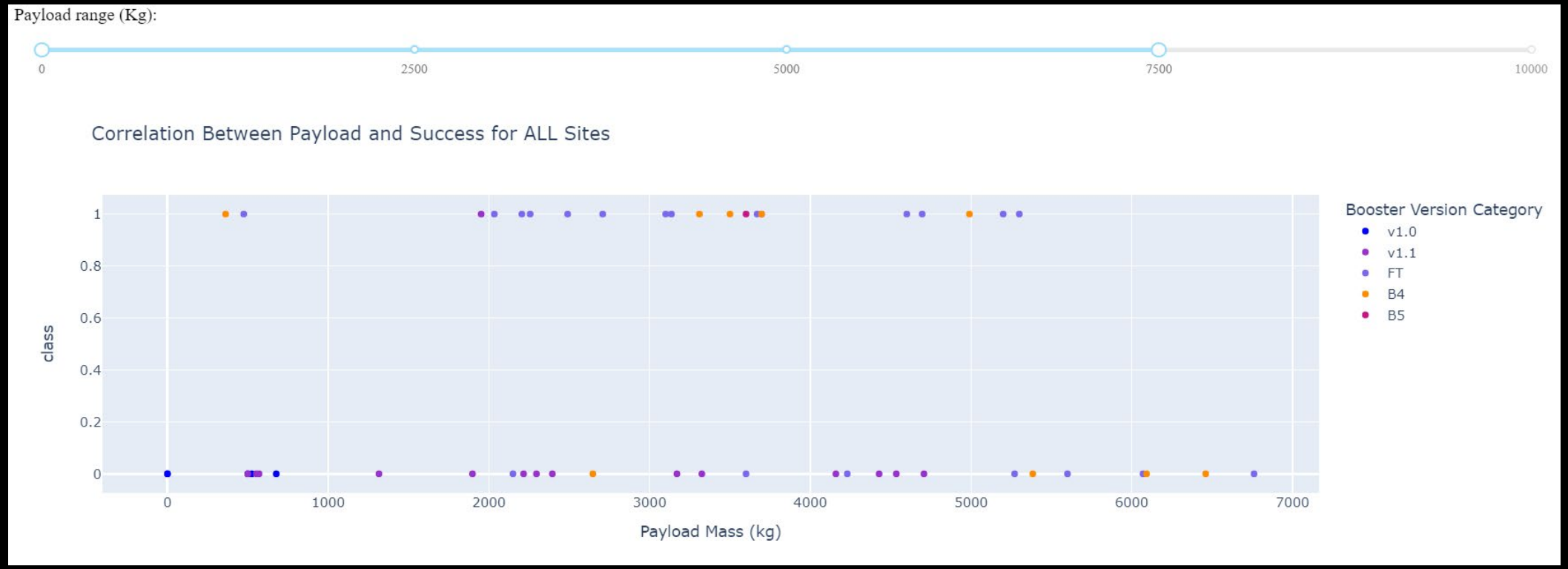
Launch Site with Highest Success Rate

KSC SLC-39A has the highest success rate with 76.9% successful landings and 23.1& failed landings.



Payload vs. Launch Outcome

The Payload range selector is only set from 0 to 10,000 kg, not including payloads greater than 10,000 kg. class 1 indicates successful landing and 0 for failure. The FT booster version accounts for majority of successful landings within the range 0-7,500 kg. The v1.1 booster version accounts for majority of failed landings within the range highlighted below.



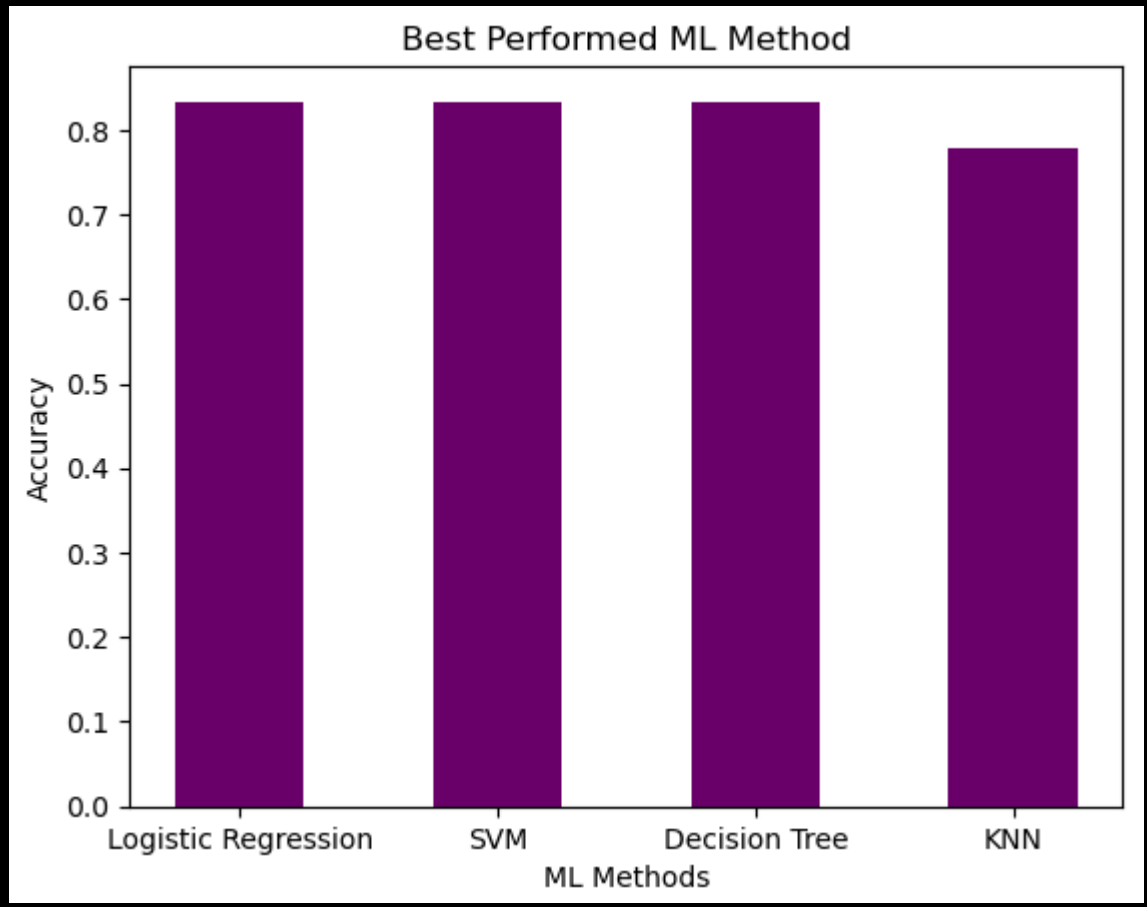
Source: space.com

Section 5

Predictive Analytics (Classification)

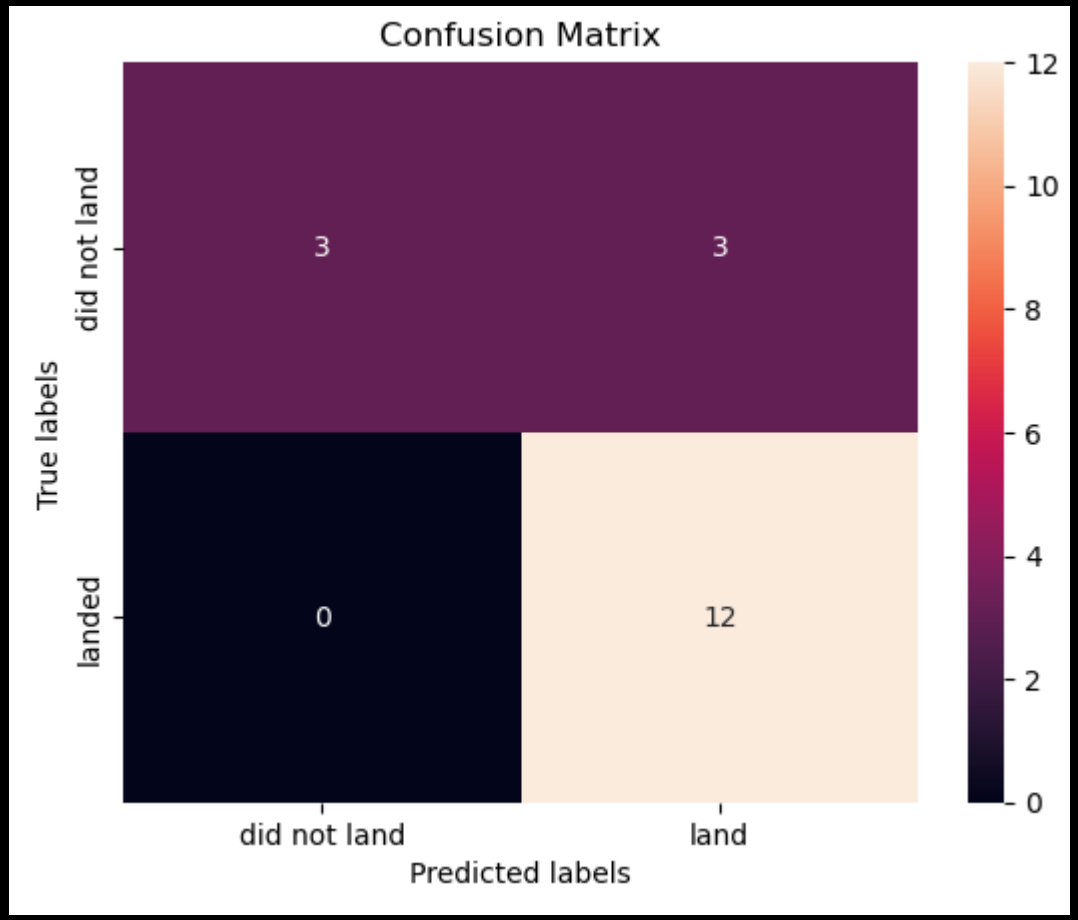
Classification Accuracy

- All (except for KNN) have relatively the same accuracy on the test set of 83.33%
- KNN's test accuracy: 77.78%
- Test size only has 18 samples, which can cause large variance in accuracy result
- More data is more likely needed to determine the most accurate model



Confusion Matrix

- This confusion matrix is the same across for Logistic Regression and Support Vector Machines since the same test set is used to evaluate those models
- All correct predictions are on the diagonal of the matrix starting from top left to bottom right.
- The models predicted 12 successful landings when the true label = landed
- The models predicted 3 unsuccessful landings when the true label = did not land
- The models predicted 3 successful landings when the true label = did not land
- Successful landings were over predicted



Conclusions

- Task: Develop a machine learning model for Space Y who wants to bid against Space X
- A machine model was created with an accuracy of 83.33%
- The goal of the model is to predict when Stage One will successfully land to save ~$100 million USD
- Space Y can implement this model to predict whether a launch will have successful Stage One landing before determining whether a launch should be made or not
- Suggestion: Collect more data , if possible, to better determine the best machine learning model and improve model accuracy

Appendix

- Python code for creating a "DATES" column for Section 2

```
[8]: import datetime as dt

     # Changed date column to a datetime data type
     df["DATES"] = pd.to_datetime(df["Date"])

     # Changed date format to year-month-day and data type to string data type
     df["DATES"] = df["DATES"].dt.strftime('%Y-%m-%d')

[9]: # Saved dataset
     df.to_csv("SPACEX-edited.csv", index=False)

[10]: # Connect to edited csv file
      df2 = pd.read_csv('SPACEX-edited.csv')
      df2.to_sql("SPACEX", con, if_exists='replace', index=False, method="multi")
```

GitHub repository:

https://github.com/collinbashore/IBM-Data-Science-Professional-Certification