IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Pedro J. Galarza
3/20/2025

Outline

- Executive Summary
- Introduction
- Methodology
- Results
- Conclusion
- Appendix

Executive Summary

Summary of Methodologies:

- **Data Collection:** SpaceX API & Web scraping SpaceX's Wikipedia page

- **Data Cleaning & Wrangling:** Calculated and replaced missing values, standardized formats, transformed, exported and stored data

- **Exploratory Data Analysis (EDA):** Used SQL, Folium and other data visualization tools to create plots, graphs, and maps to analyze and find any relationships between different variables and the landing outcome

- **Dashboard:** Used Plotly Dash to build a Dashboard

- **Model Selection:** Used Logistics Regression, SVM, Decision Tree and KNN models

- **Model Training:** Split data (training and testing sets); used GridSearchCV for cross-validation, best_score, best_params to tune hyperparameters

Executive Summary (Part 2)

- **Summary of all results:**
  - Key Factors: Payload mass, orbit type, launch sites, and landing outcomes
  - Most Successful Launch Site: **KSC LC-39A: 41.7% (Kennedy Space Center Launch Complex 39A)**
  - Launch Success Rate for **KSC LC-39A: 76.9%**
  - Ideal Payload Mass: Between **2,000 to 5,000 kilograms (kg)**
  - Ideal Orbit Type: **LEO, ISS and PO**
  - Best Classification Model: **Logistical Regression with 83.33% Accuracy**
  - Confusion Matrix: **Logistical Regression**
    - **True Positive - 12** (True label is landed, Predicted label is also landed)
    - **True Negative – 3** (True label is not landed, Predicted label is not landed)
    - **False Positive - 3** (True label is not landed, Predicted label is landed)

Introduction

- **Project background and context**
  - The project states “In this capstone, we will predict if the Falcon 9 first stage will land successfully. SpaceX advertises Falcon 9 rocket launches on its website, with a cost of 62 million US dollars; other providers cost upward of 165 million US dollars each, much of the savings is because SpaceX can reuse the rocket’s first stage.” Therefore, if we can determine if the first stage will land, we can determine the cost of a launch.
  - The context of this project is to analyze the launch data to see what insights we can find from all the data. The project also mentioned that “this information can be used if an alternate company wants to bid against SpaceX for a rocket launch.”

- **Problems you want to find answers**
  - What is the launch success rate?
  - What is the most successful launch site?
  - What are the factors behind successful and failed launches? Is there any relationship between flight number, orbit type, payloads, launch sites and launch outcomes?

Section 1

Methodology

Methodology

- **Data collection methodology:**
  - Utilized the SpaceX API to gather up-to-date data on SpaceX launches and web scraping to collect historical data from SpaceX's Wikipedia page, specifically Falcon 9 rockets.
- **Perform data cleaning and wrangling:**
  - Cleaned and pre-processed the data. This involved handling missing values, normalizing data formats, and transforming (dataframe) and storing data to optimize its structure for analysis.
- **Perform exploratory data analysis (EDA) using visualization and SQL tools:**
  - Used these tools to conduct an EDA process to uncover key patterns, and trends within the dataset.
- **Perform interactive visual analytics using Folium and Plotly Dash**
- **Perform predictive analysis using classification models**
  - Built and tuned classification models to predict landing success.

Data Collection

To collect data for the SpaceX project, two methods were used: SpaceX REST API and Web scraping Wikipedia SpaceX pages. First, for the SpaceX API, the **GET request** with the API or Static URL was used to fetch and extract the JSON data using the necessary endpoints. JSON data was then viewed using the **.json() method** and converted to a structured format using Pandas DataFrame **.json_normalize**. The data was then filtered, cleaned and exported to a CSV file. Second, for Web scraping, the GET request with the Static URL was used. **Python's BeautifulSoup** was used to parse, locate and extract the HTML data table, columns and rows. Data was then store in a Launch Library; and converted into a Pandas DataFrame using **=pd.DataFrame().** Finally, the data was exported to a CSV file.

SPACEX REST API

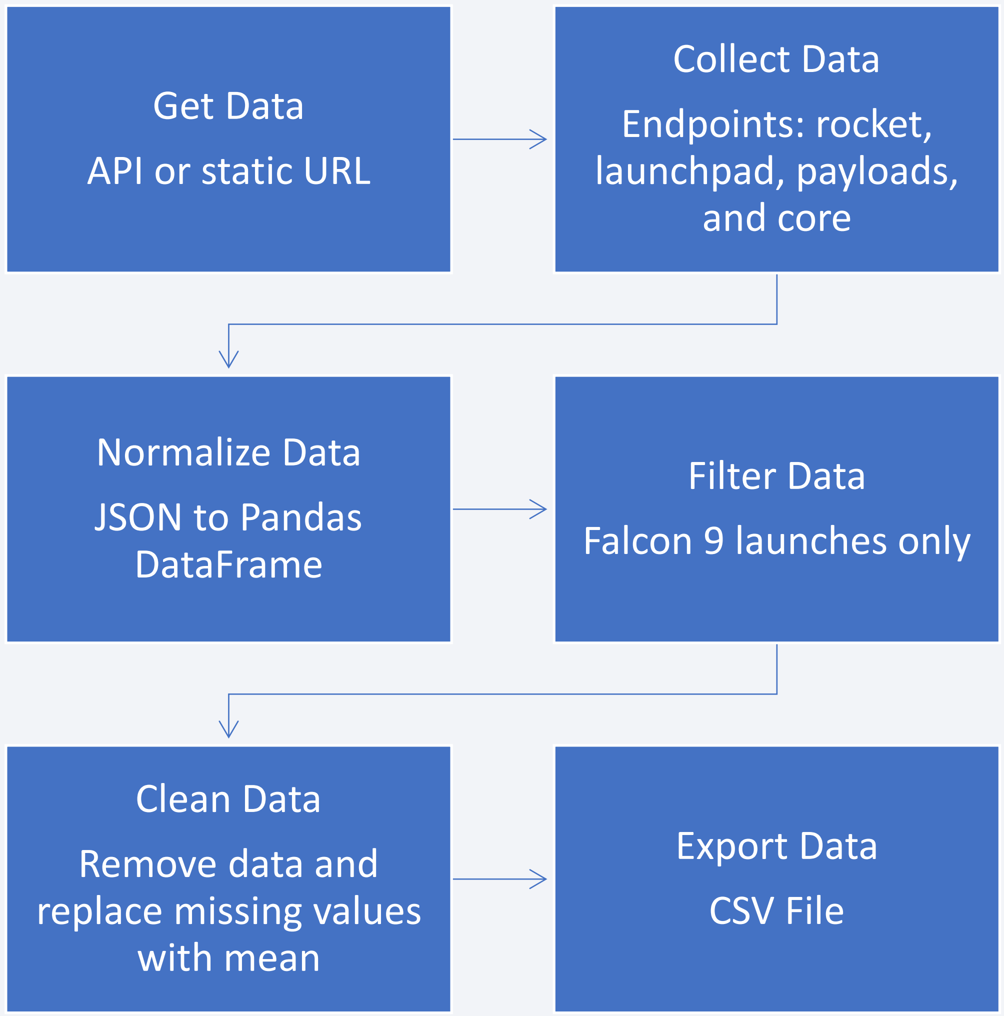
1. Fetch Data
   GET request
2. Collect Data
   Necessary Endpoints
3. Normalize Data
   JSON to Pandas DataFrame
4. Filter Data
   Falcon 9 launches only
5. Clean Data
   Replace data with Mean
6. Export Data
   CSV file

Web scraping Wikipedia

1. Fetch Data
   GET request
2. Collect Data
   BeautifulSoup
3. Extract Data
   HTML Columns, Rows & Variables
4. Store Data
   Launch Library
5. Convert Data
   HTML to Pandas DataFrame
6. Export Data
   CSV file

Data Collection – SpaceX API

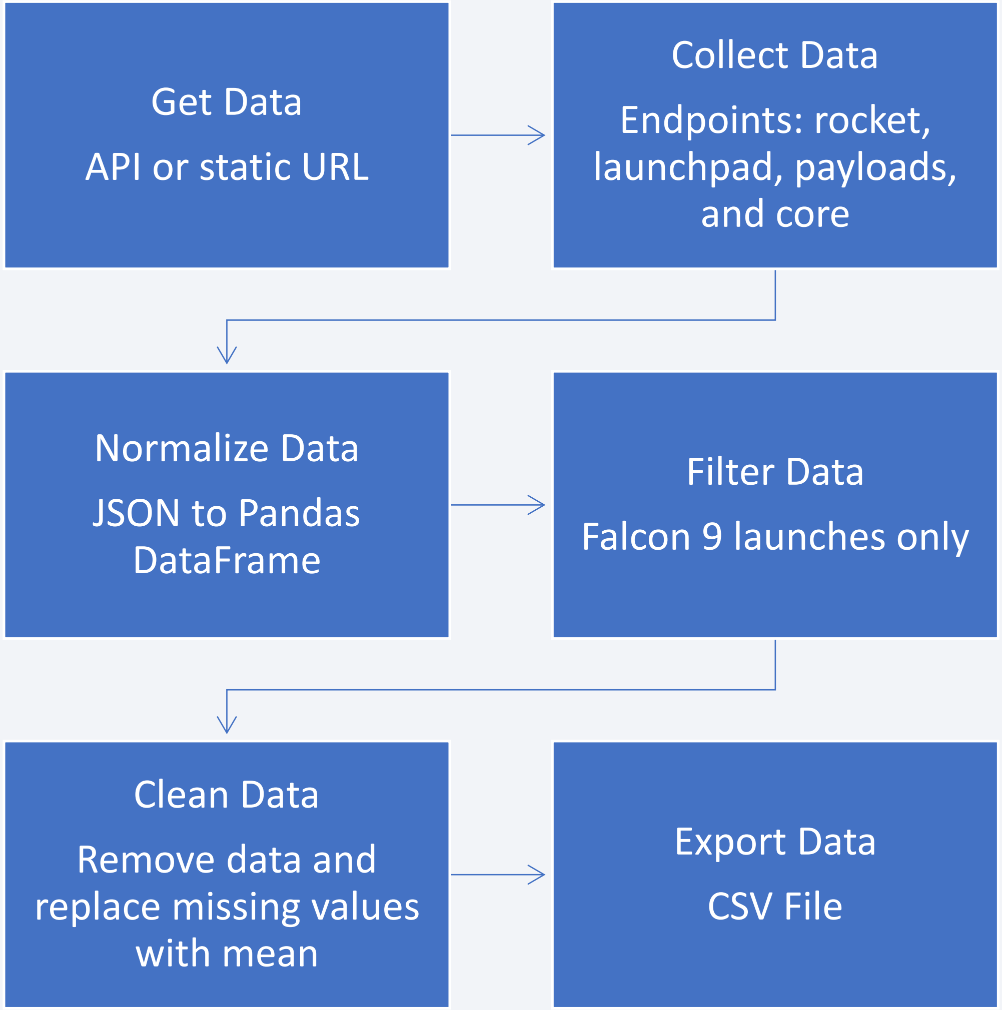
- Import libraries: **Requests, Pandas, NumPy**
- Find API or static JSON URL: **Use GET =requests.get()**
- Collect data: From rocket, launchpad, payloads, and core. **DEF,FOR,IF,ELSE**
- Normalize Data: JSON data to Pandas DataFrame using **.json_normalize()**
- Filter Data: Apply filter for Falcon 9 launches only. **=data_falcon9[]**
- Clean Data: Remove, convert data, and **replace missing values with mean**
- Export Data: Save to CSV file **data_falcon9.to_csv()**



Data Collection – SpaceX API (Part 2)

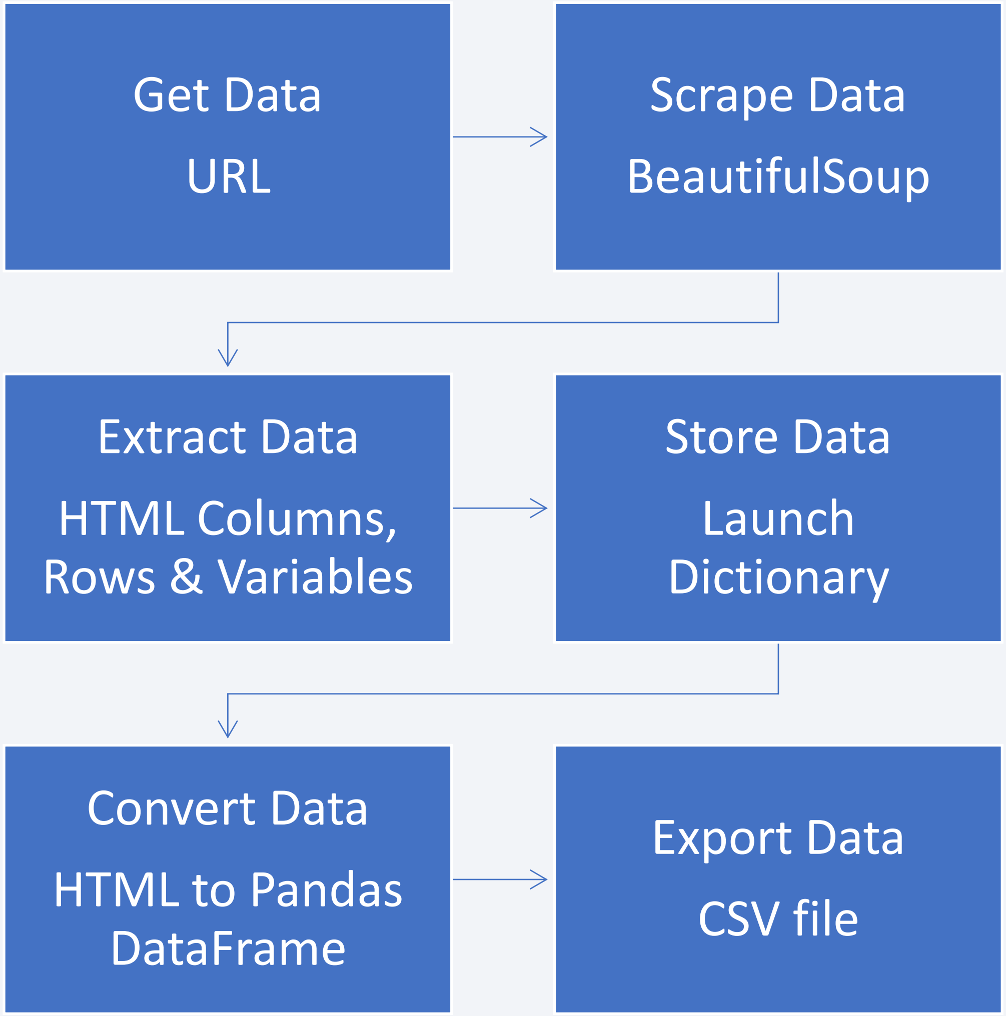
GitHub URL:

- https://github.com/projectstrawberrylemonade/project1/blob/08e2dafc49ba7c8eaae3e14b4dd28cda2bbf838c/1%20-%20Data%20Collection%20-%20jupyter-labs-spacex-data-collection-api.ipynb



Data Collection - Web Scraping

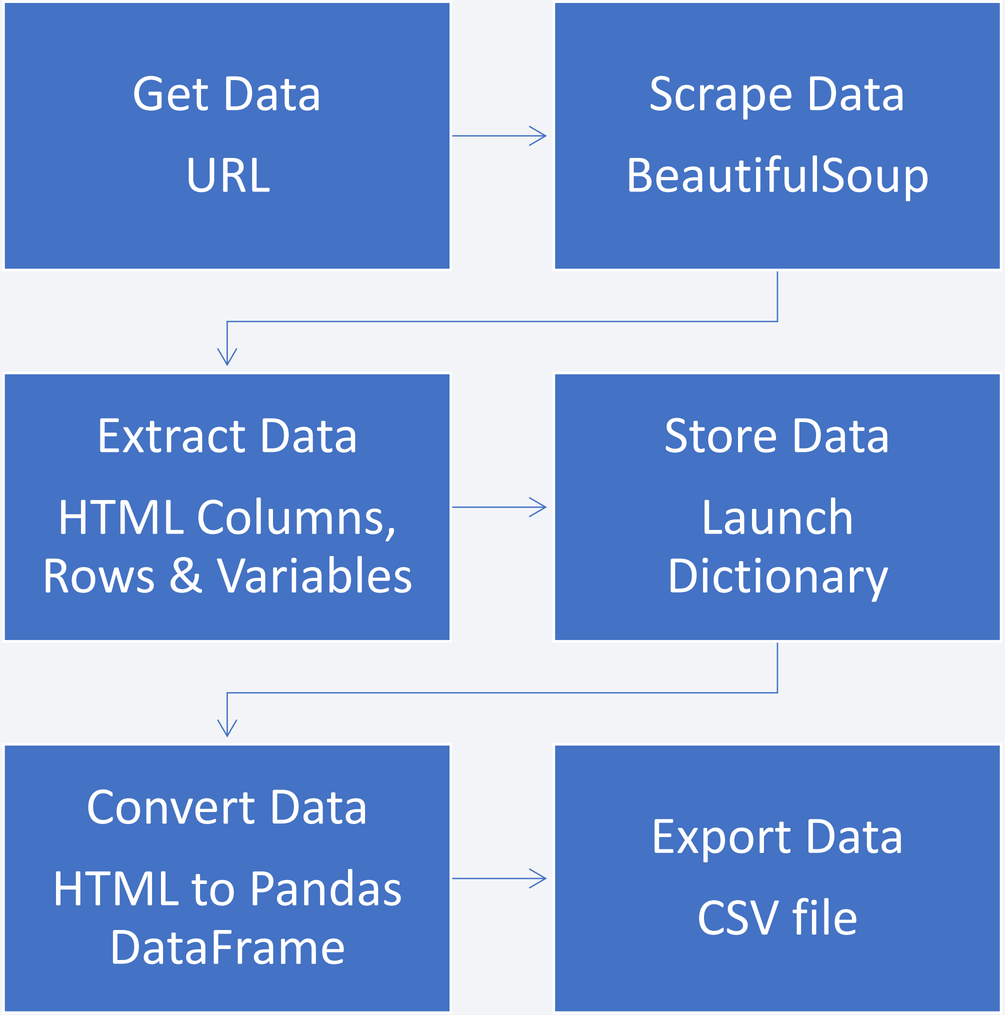
- Import libraries: **Sys, Request, BeautifulSoup, RE, Unicodedata, Pandas**
- Find Wikipedia SpaceX or static URL: Use GET request: **=requests.get()**
- Scrap Data: Use BeautifulSoup for parsing HTML data **=BeautifulSoup()**
- Extract Data: Columns, Rows, Variables from HTML table: **=soup.find_all(th), =extract_column_from_header(th), =table.find_all("tr"), = table.find_all("td")**
- Store Data: **Using launch_dict[]**
- Convert Data: Convert extracted data to Pandas Data Frame. **=pd.DataFrame()**
- Export Data: Save to CSV file. **df.to_csv()**



Data Collection - Web Scraping (Part 2)

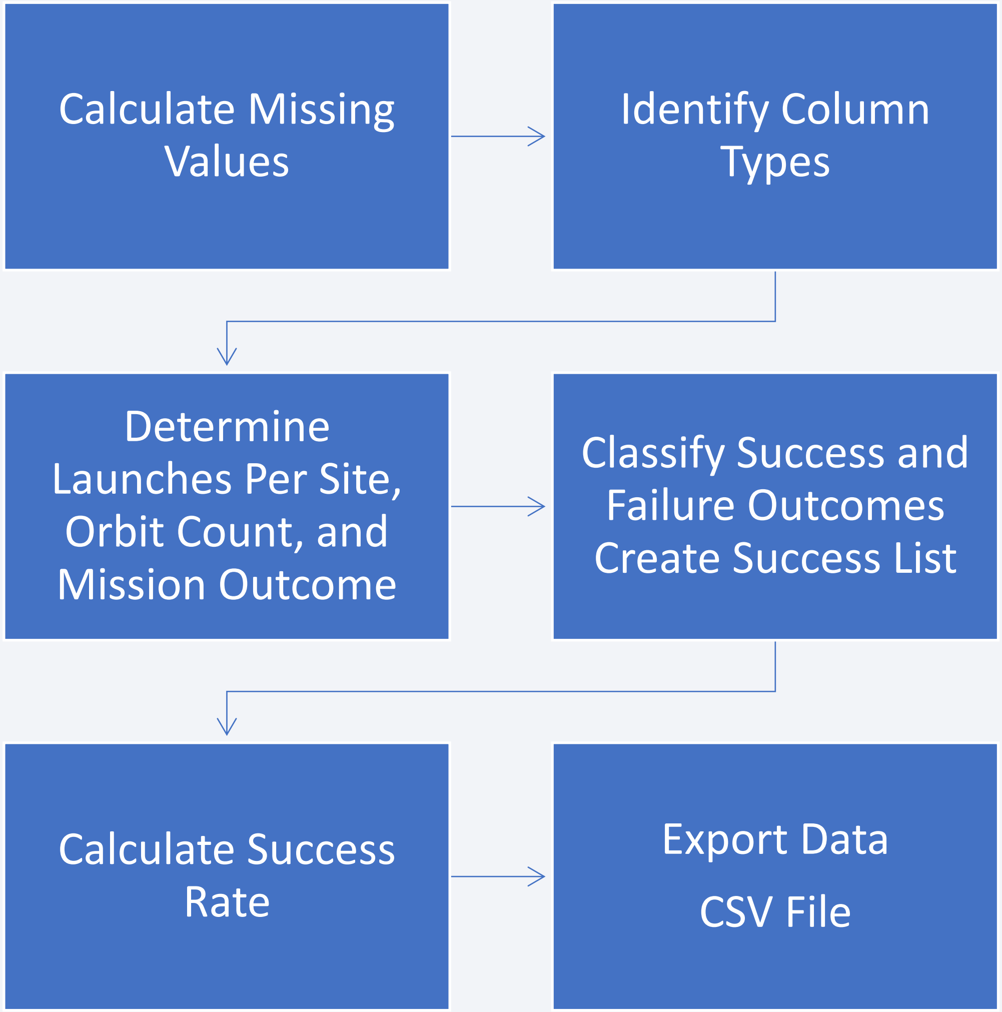
GitHub URL:

- https://github.com/projectstrawberrylemonade/project1/blob/08e2dafc49ba7c8eaae3e14b4dd28cda2bbf838c/1%20-%20Data%20Collection%20-%20dataset_part_1.csv



Data Wrangling

- Calculate the percentage of missing values on each attribute: **isnull().sum()/len(df)*100**
- Identify if columns are numerical or categorical: **dtypes**
- Determine the number of Launches Per Site, Orbits, and Outcome: **value_counts()**
- Classify Success and Failure Outcomes and create a new list using a ‘Class’ feature for landing success **(1 for success, 0 for failure): IF, ELSE**
- Calculate success rate using **‘Class’**
- Store Data: Saves the processed data to a new CSV file **(dataset_part_2.csv)** for use in the next stage of the project.

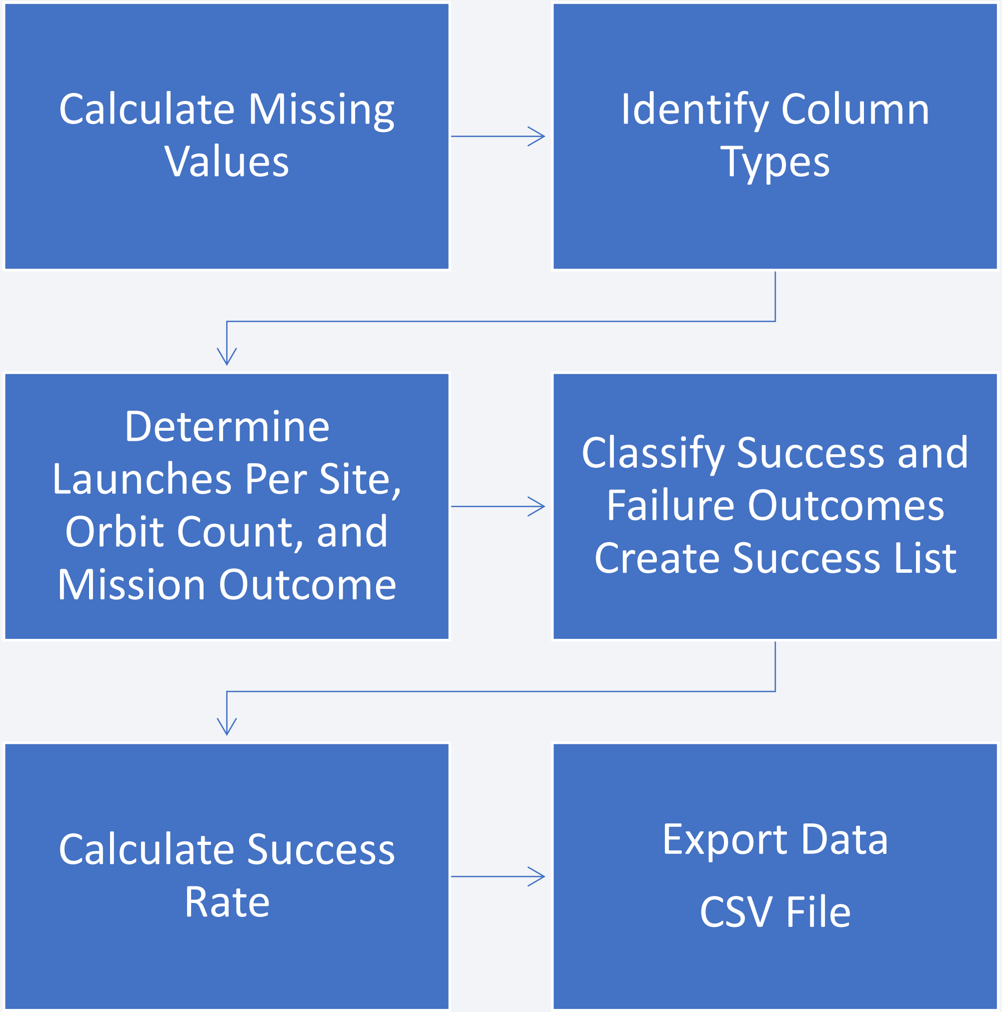


Data Wrangling (Part 2)

GitHub URLs:

- https://github.com/projectstrawberrylemonade/project1/blob/08e2dafc49ba7c8eaae3e14b4dd28cda2bbf838c/3%20-%20Data%20Wrangling%20-%20labs-jupyter-spacex-Data%20wrangling.ipynb

- https://github.com/projectstrawberrylemonade/project1/blob/08e2dafc49ba7c8eaae3e14b4dd28cda2bbf838c/3%20-%20Data%20Wrangling%20-%20dataset_part_2.csv



EDA with Data Visualization

- **Summarize what charts were plotted and why you used those charts.**

1. Scatter Plot:

- Used it to visualize the relationship or correlation between two variables: **(Flight Number vs Payload, Flight Number vs Launch Site, Payload vs Launch Site, Flight Number vs Orbit Type, Payload vs Orbit Type)**

2. Bar Chart:

- Used it to compare the success rate of each orbit type. **(Success Rate vs Orbit Type).**

3. Line Chart:

- Used it to find a trend and analyze success rate over the years. **(Success Rate vs Year)**

- https://github.com/projectstrawberrylemonade/project1/blob/08e2dafc49ba7c8eaae3e14b4dd28cda2bbf838c/5%20-%20EDA%20-%20Data%20Visualization%20-%20edadataviz.ipynb

EDA with SQL (Part 1 of 3)

- **Using bullet point format, summarize the SQL queries you performed.**
- **SELECT DISTINCT Launch_Site FROM SPACEXTABLE:**
  This query retrieved all the unique launch site locations.
- **SELECT * FROM SPACEXTABLE WHERE Launch_Site LIKE 'CCA%' LIMIT 5:**
  This query selected all columns for records where the launch site starts with 'CCA'. It was limited to 5 results.
- **SELECT SUM(PAYLOAD_MASS__KG_) FROM SPACEXTABLE WHERE Customer='NASA (CRS)':**
  This query calculated the total payload mass for missions with NASA (CRS) as the customer.
- **SELECT AVG(PAYLOAD_MASS__KG_) FROM SPACEXTABLE WHERE Booster_Version='F9 v1.1':**
  This query calculated the average payload mass for missions using booster version 'F9 v1.1'.

EDA with SQL (Part 2 of 3)

- **Using bullet point format, summarize the SQL queries you performed.**
- **SELECT MIN(Date) FROM SPACEXTABLE WHERE Landing_Outcome='Success (ground pad)':**
  This query searched for the earliest date of a successful ground pad landing.
- **SELECT DISTINCT Booster_Version FROM SPACEXTABLE WHERE Landing_Outcome='Success (drone ship)' AND PAYLOAD_MASS__KG_ BETWEEN 4000 AND 6000:**
  This query retrieved unique booster versions for successful drone ship landings with payload mass between 4000 and 6000 kg.
- **SELECT SUBSTR(Mission_Outcome,1,7) AS Mission_Outcome, COUNT(*) FROM SPACEXTABLE GROUP BY 1:**
  This query groups and counts mission outcomes (Success or Failure).

EDA with SQL (Part 3 of 3)

- **Using bullet point format, summarize the SQL queries you performed.**

- **SELECT DISTINCT Booster_Version FROM SPACEXTABLE WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTABLE):**
  This query identifies booster versions carrying the maximum payload mass using a subquery.

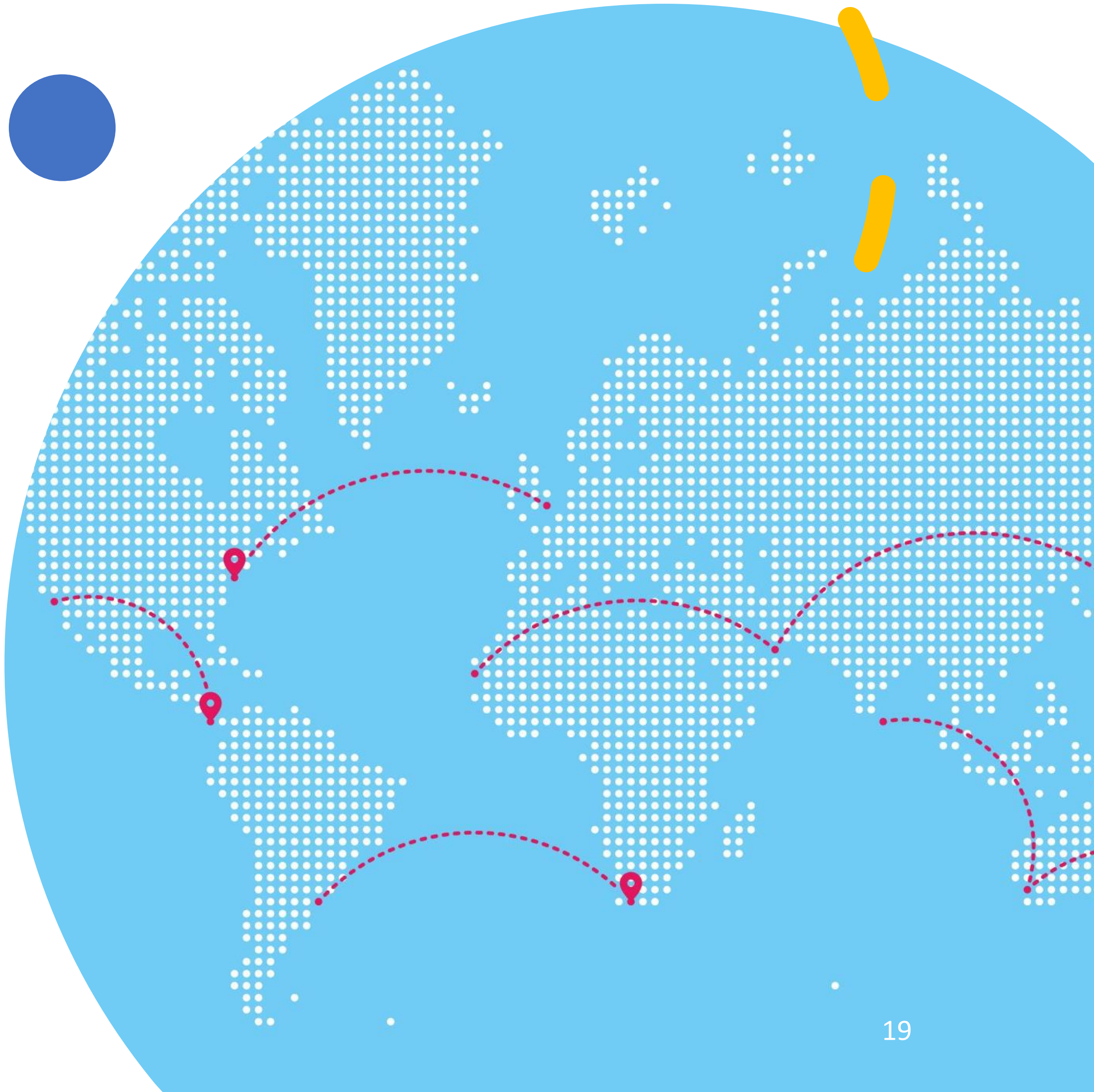
- **SELECT DISTINCT Landing_Outcome, Booster_Version, Launch_Site FROM SPACEXTABLE WHERE Landing_Outcome='Failure (drone ship)':**
  This query retrieves records for missions with failure landing outcomes on drone ships.

- **SELECT Landing_Outcome, COUNT(*) FROM SPACEXTABLE WHERE Date BETWEEN '2011-06-04' AND '2017-03-20' GROUP BY Landing_Outcome ORDER BY 2 DESC:**
  This query ranks landing outcomes within a specific date range based on their frequency.

- https://github.com/projectstrawberrylemonade/project1/blob/08e2dafc49ba7c8eaae3e14b4dd28cda2bbf838c/4%20-%20EDA_SQL_jupyter_labs_eda_sql_coursera_sqllite.ipynb

Build an Interactive Map with Folium

- **Summarize what map objects such as markers, circles, lines, etc. you created and added to a folium map. Explain why you added those objects.**
- **Map (folium.map):** Created the map to visualize the launch sites and outcomes.
- **Markers (folium.marker):** Inserted markers to show the locations of Falcon 9 launch sites.
- **Circles (folium.circle):** Used orange circles to highlight the location of launch sites.
- **Lines (folium.polyline):** Draw lines to show the distance between the launch site and the nearest coast, highway, and city.
- https://github.com/projectstrawberrylemonade/project1/blob/08e2dafc49ba7c8eaae3e14b4dd28cda2bbf838c/6%20-%20Folium%20-%20Site%20Location%20-%20lab_jupyter_launch_site_location.ipynb



Build a Dashboard with Plotly Dash

- **Summarize what plots/graphs and interactions you have added to a dashboard. Explain why you added those plots and interactions. Add the GitHub URL of your completed Plotly Dash lab, as an external reference and peer-review purposes.**

- **Pie Chart:** Used it to show the total successful vs. failed launches for launch sites.

- **Scatter Plot:** Used it to display the correlation between payload mass and launch success.

- **Dropdown Menu:** Used it to allow users to select a specific launch site to filter the data.

- **Range Slider:** Used it to enable users to select a range of payload masses to visualize in the scatter plot.

Build a Dashboard with Plotly Dash

Github Links:

Code:

- https://github.com/projectstrawberrylemonade/project1/blob/08e2dafc49ba7c8eaae3e14b4dd28cda2bbf838c/7%20-%20Spacex_dash_app.py

Pie Chart #1: Total Success Launches by Site

- https://github.com/projectstrawberrylemonade/project1/blob/08e2dafc49ba7c8eaae3e14b4dd28cda2bbf838c/7%20-%20Pie%20Chart%201.png

Pie Chart #2: Total Launches for Site KSC LC-39A

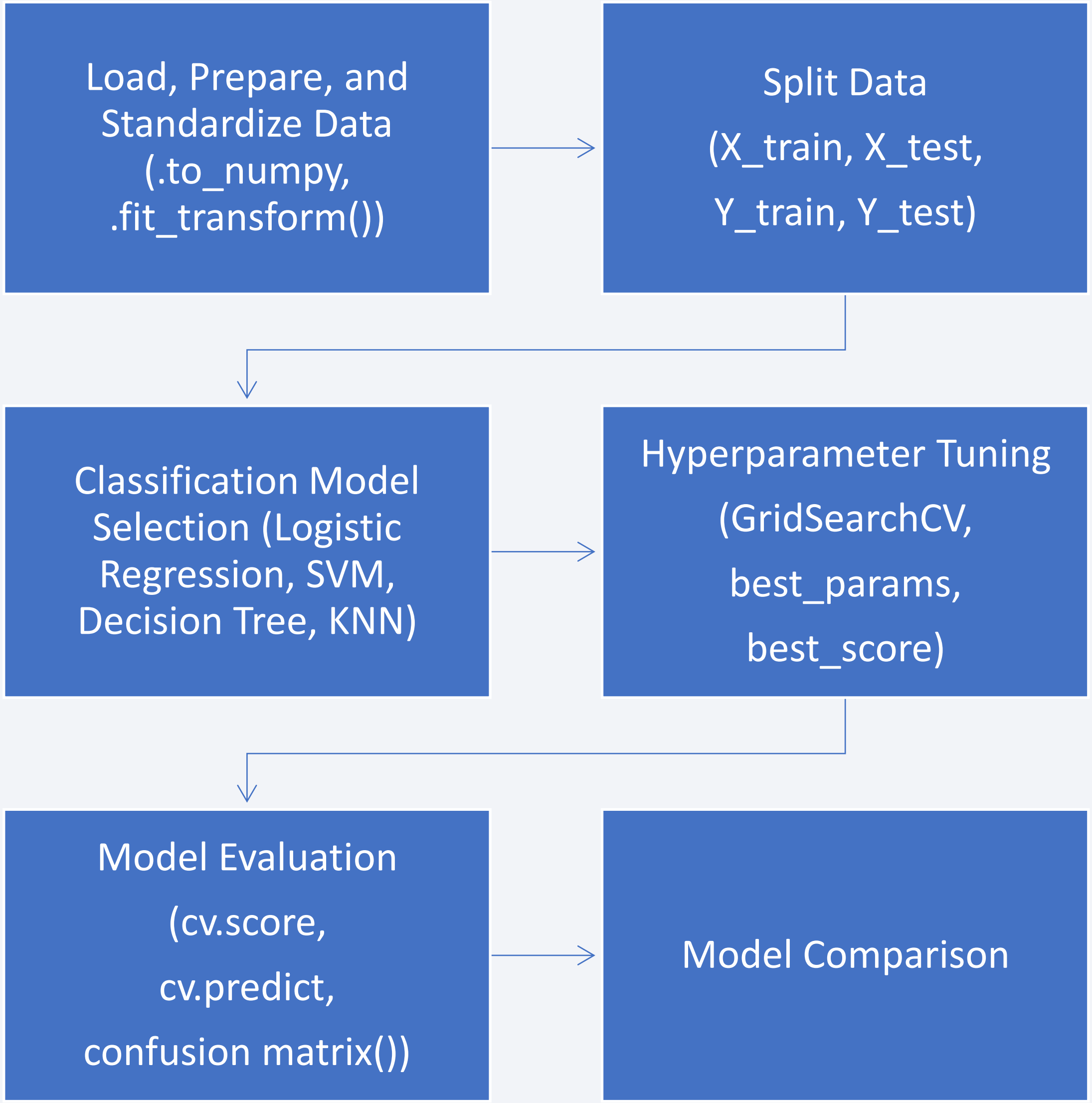
https://github.com/projectstrawberrylemonade/project1/blob/08e2dafc49ba7c8eaae3e14b4dd28cda2bbf838c/7%20-%20Pie%20Chart%202.png

Plot Graph: All sites Payload Mass 0 to 9,600 kg

https://github.com/projectstrawberrylemonade/project1/blob/08e2dafc49ba7c8eaae3e14b4dd28cda2bbf838c/7%20-%20Payload%20Mass%20Graph.png

Predictive Analysis (Classification)

- Import libraries and load data: **Pandas, NumPy and Scikit-learn. Prepare data: Y, X. Y = data['Class'].to_numpy()**
- Standardize data: **Use feature scaling. transform = preprocessing.StandardScaler() and .fit_transform(X)**
- Split Data: Training and testing sets.
  X_train, X_test, Y_train, Y_test
- Classification Model Selection:
  Logistic Regression, SVM, Decision Tree, KNN
- Hyperparameter Tuning: **Use GridSearchCV, best_params_, best_score_ to find optimal hyperparameters for each model.**
- Model Evaluation: Assess model performance using **model_cv.score(X_test, Y_test)) for model test accuracy, yhat=model_cv.predict(X_test), plot_confusion_matrix(Y_test,yhat)** for confusion matrix.
- Model Comparison: **.best_score_**
- https://github.com/projectstrawberrylemonade/project1/blob/08e2dafc49ba7c8eaae3e14b4dd28cda2bbf838c/8%20-%20SpaceX_Machine%20Learning%20Prediction_Part_5.ipynb



Results

- **Exploratory data analysis results.**
Interactive analytics demo in screenshots.
Predictive analysis results.

- EDA results, graphs and screenshots appeared in the next section titled: “Insights drawn from EDA”.

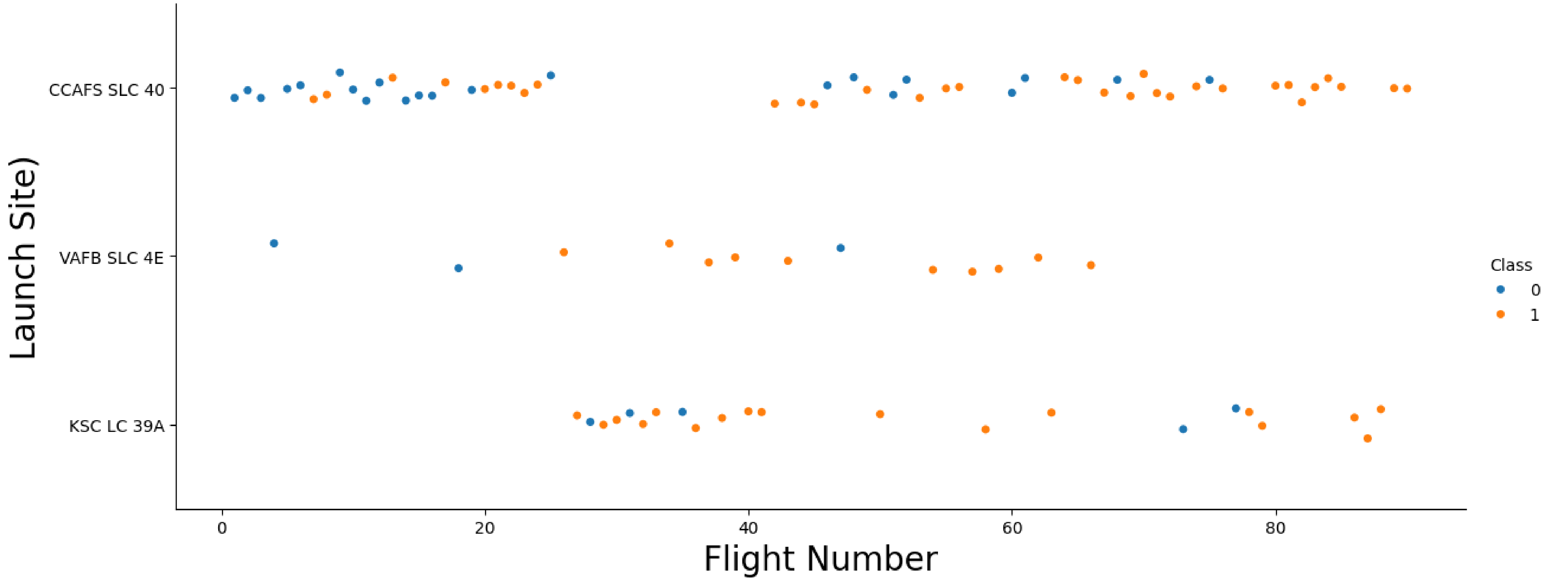
- Predictive analysis results appeared at the “Predictive Analysis (Classification)” section.

Section 2

Insights drawn from EDA

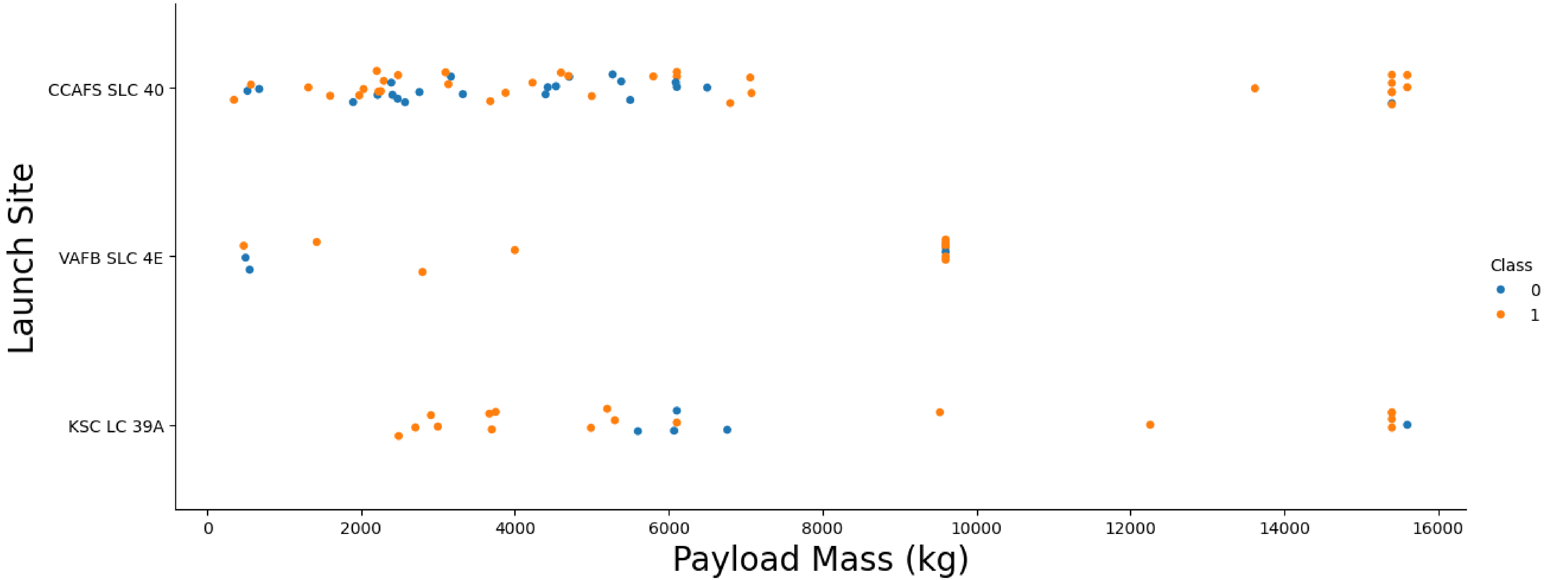
Flight Number vs. Launch Site

- This chart shows that most of the launches of SpaceX's Falcon 9 occurred in the CCAFS SLC 40 launch site. **The most successful seems to be KSC LC-39A**. Overall, including all launches from all sites, the first 20 were mostly unsuccessful but **after number 60 were mostly successful landings showing an upward trend**.



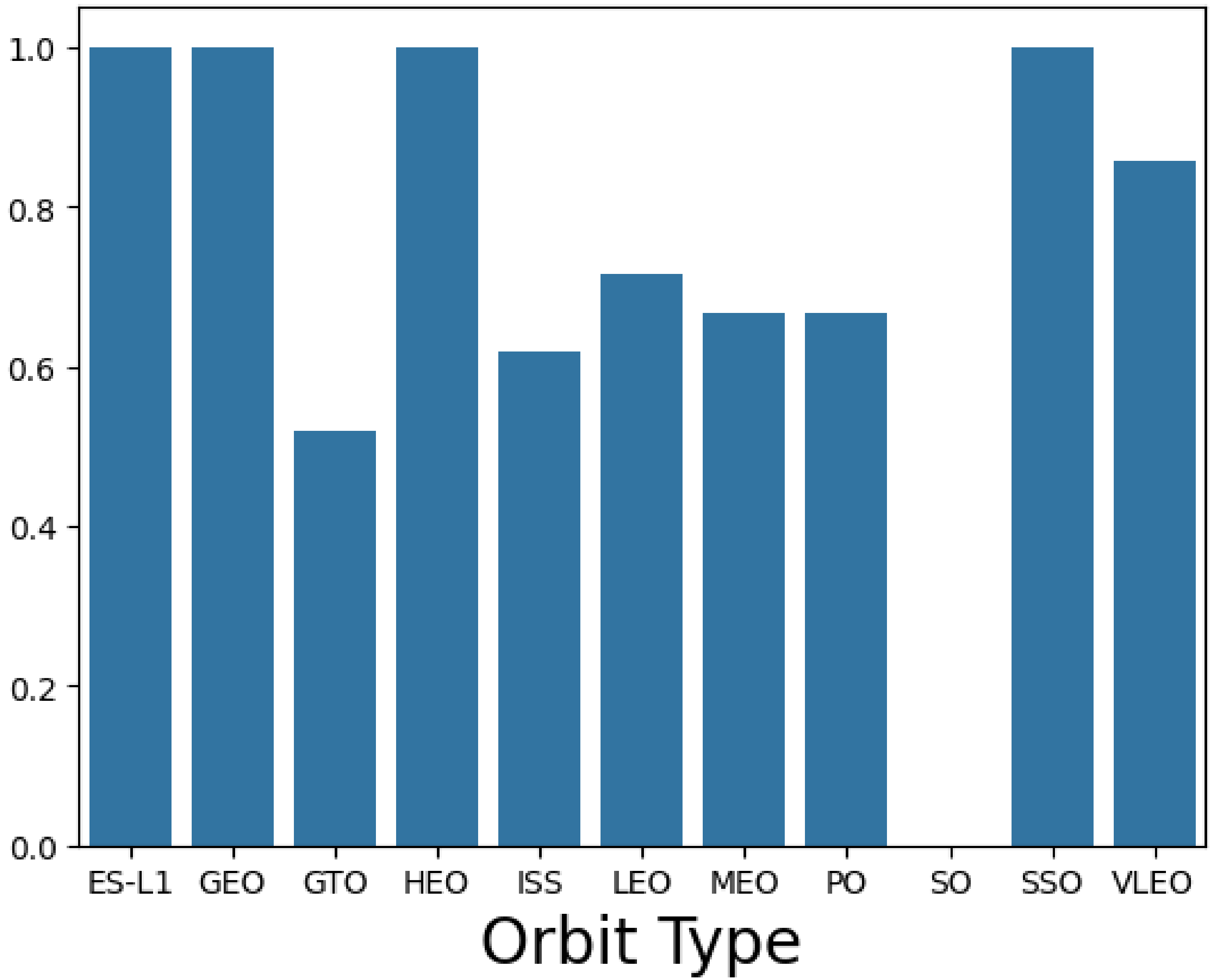
Payload vs. Launch Site

- This chart shows **most launches have been under 7,000kg for all sites**. Launches of 9,000kg or more have been mostly successful. VAFB SLC 4E site does not have any launch of 10,000kg or more.



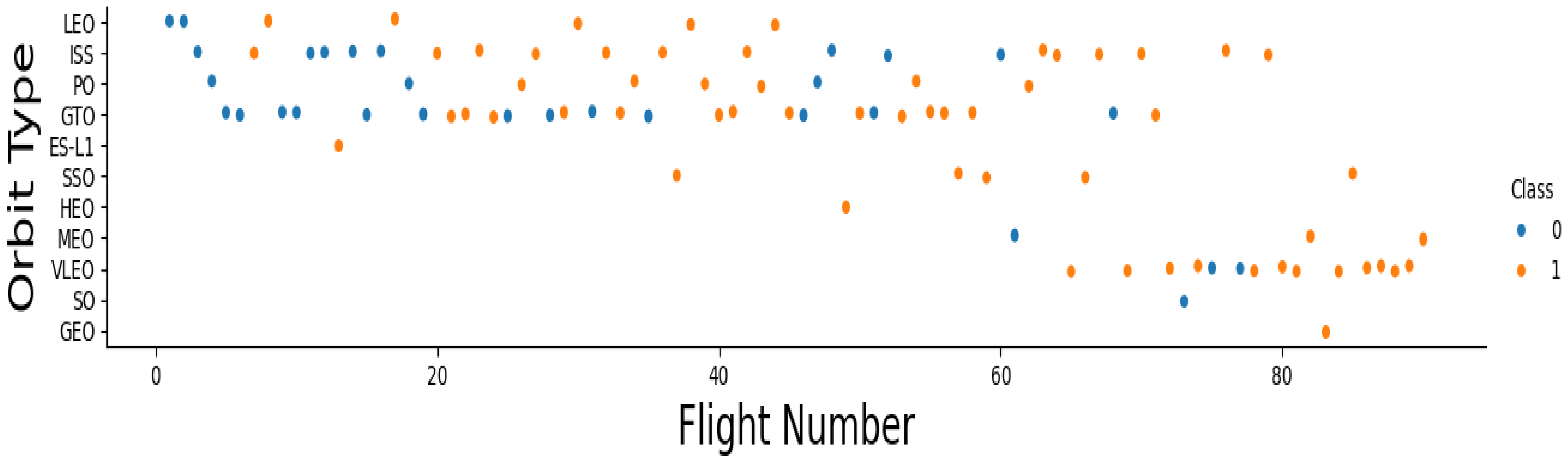
Success Rate vs. Orbit Type

- This chart shows that orbits: **ES-L1, GEO, HEO, and SSO**, have achieved higher success rates for Falcon 9 rockets with **100%**. However, some of those Orbits only have one launch.



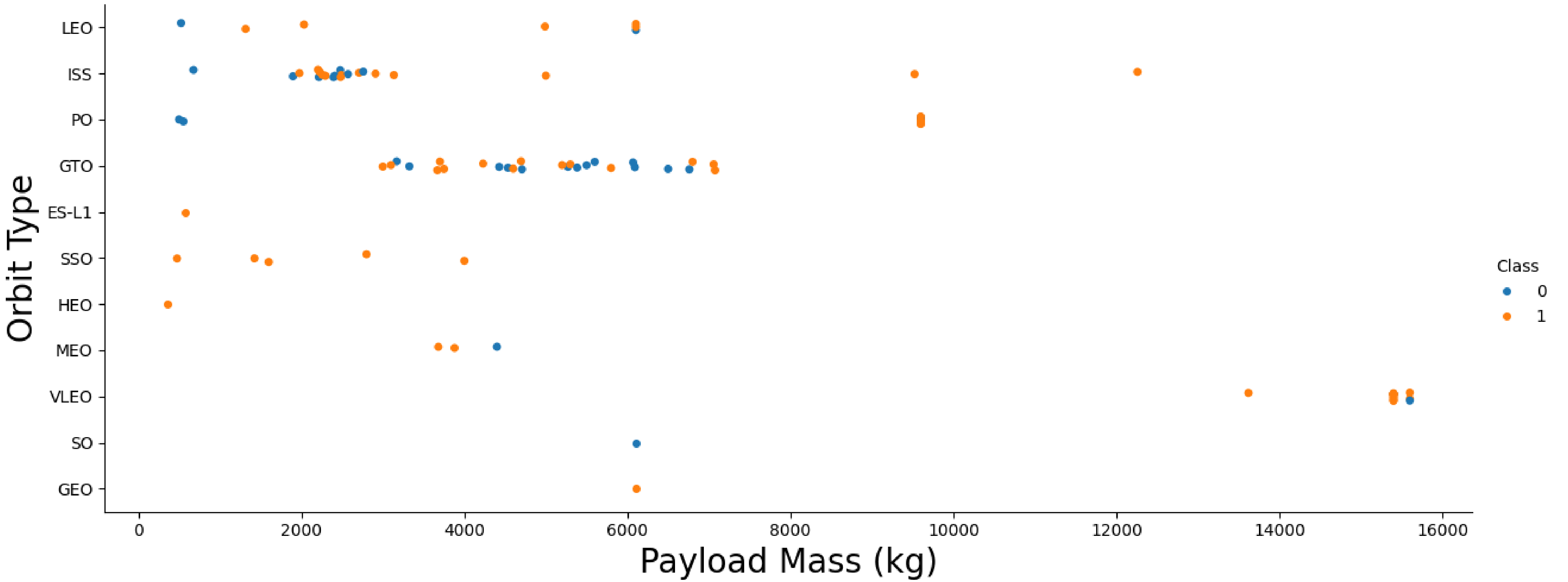
Flight Number vs. Orbit Type

- **SpaceX's Falcon 9 first stage landing success rate for missions to LEO, ISS and PO appears to increase with more flight numbers**, while other orbits show no clear relationship with flight number because there are not enough launches. In conclusion, most launches for **most orbits after flight number 60 are showing an upward trend.**



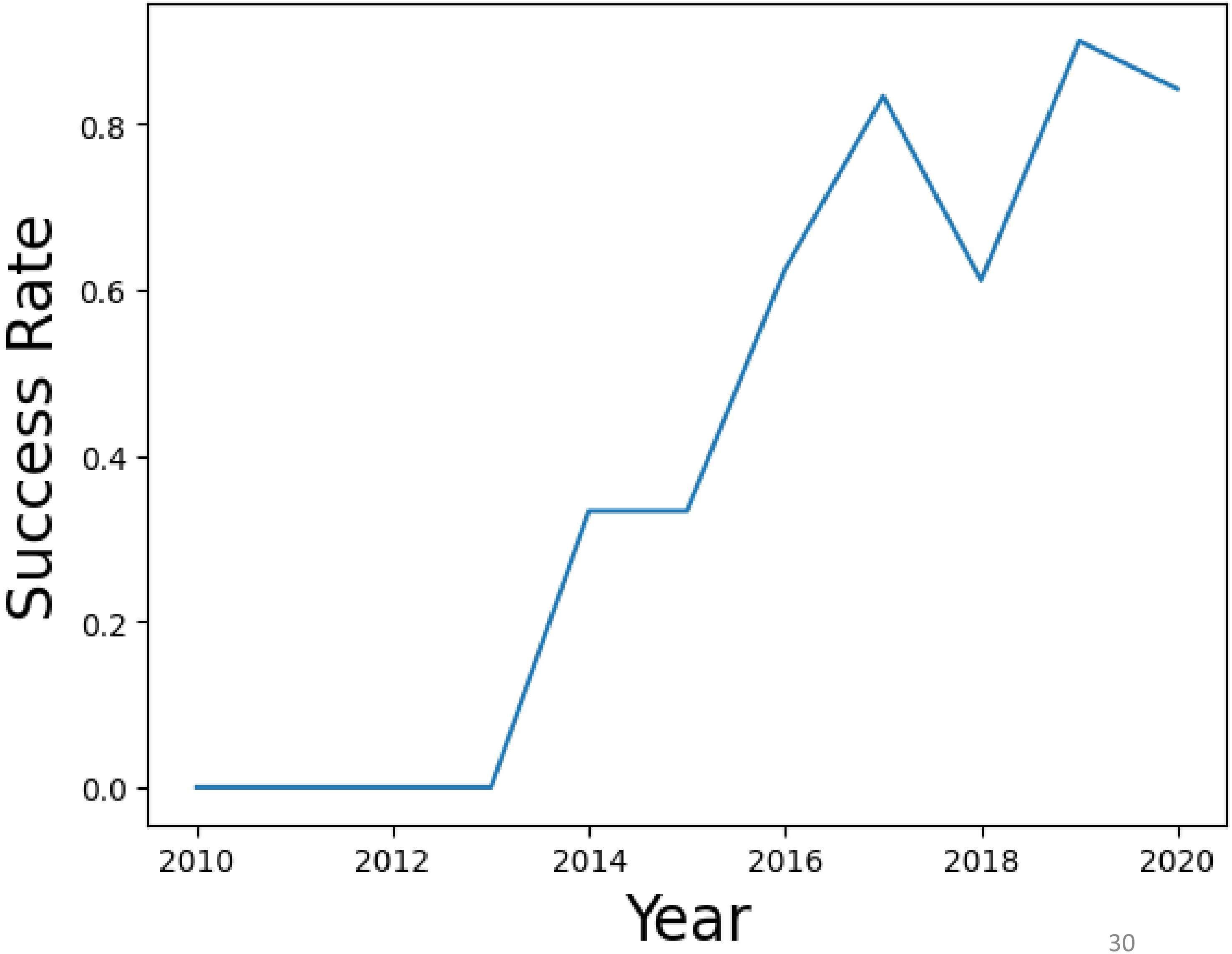
Payload vs. Orbit Type

- The graph indicates that **heavier payloads are more likely to have a successful landing with LEO, ISS, and PO orbits.** VLEO looks promising too with a small number of successful landings, but there are not enough launches. However, GTO orbits is less clear due to mix results.



Launch Success Yearly Trend

- This chart shows an overall increasing trend in SpaceX's Falcon 9 first stage landing success rate from **2013 to 2020**. However, in 2014-15 there was a period of no change, **but then the success rate increased again to 80%.** In 2017-18 the success rate went down again to 60%, **but after 2019 it stayed in the 80% or above range. In conclusion, we see an upward trend overall.**



All Launch Site Names

- **names_of_launch_sites = pd.read_sql('select distinct Launch_Site from SPACEXTABLE', con)**
- **names_of_launch_sites**
- This query selects the unique launch sites from the SPACEXTABLE table in the database and stores the results in a Pandas DataFrame named names_of_launch_sites.
- 0 - Cape Canaveral Air Force Station Launch Complex 40 (CCAFS LC-40)
- 1 - Cape Canaveral Space Force Station Space Launch Complex 40 (CCAFS SLC-40)
- 2 - Kennedy Space Center Launch Complex 39A (KSC LC-39A)
- 3 - Vandenberg Air Force Base Space Launch Complex 4E (VAFB SLC-4E)

	Launch Site
0	CCAFS LC-40
1	CCAFS SLC-40
2	KSC LC-39A
3	VAFB SLC-4E

Launch Site Names Begin with 'CCA'

- Find 5 records where launch sites begin with `CCA`. Present your query result with a short explanation here.
- **CCA_records = pd.read_sql("select * from SPACEXTABLE where Launch_Site like 'CCA%' limit 5", con)**
- **CCA_records**
- This query selects all columns from the SPACEXTABLE where Launch_Site starts with "CCA", limiting the results to 5 rows.

	Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
0	2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
1	2010-12-08	15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight C1, two CubeSats, barrel of...	0	LEO (ISS)	NASA (COTS) NRO	Success	Failure (parachute)
2	2012-05-22	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
3	2012-10-08	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
4	2013-03-01	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

Total Payload Mass

- Calculate the total payload carried by boosters from NASA (CRS):
  - **45,596 kilograms**
- Present your query result with a short explanation here:
- This query selects payload missions information, filters the data for NASA (CRS) only, and adds the total payload mass for these missions.
  - **total_payload_mass = pd.read_sql("SELECT SUM(PAYLOAD_MASS__KG_) FROM SPACEXTABLE WHERE Customer = 'NASA (CRS)'", con)**
  - **total_payload_mass**

Average Payload Mass by F9 v1.1

- Calculate the average payload mass carried by booster version F9 v1.1:
  - **2,928.40 Kilograms**
- Present your query result with a short explanation here:
  - This query selects all the payload missions, calculates the average payload mass, and presents the calculated value.
  - **average_payload_mass = pd.read_sql("SELECT avg(PAYLOAD_MASS__KG_) FROM SPACEXTABLE WHERE Booster_Version = 'F9 v1.1'", con)**
  - **average_payload_mass**

First Successful Ground Landing Date

- Find the dates of the first successful landing outcome on ground pad:
  - **2015-12-22**
- Present your query result with a short explanation here:
  - This query finds the earliest date a Space X mission successfully landed on a ground pad. First, it selects the database. Second, it filters for successful ground pad landings. Finally, it selects the earliest (minimum) date from the filtered results.
  - **first_date_of_successful_landing = pd.read_sql("SELECT MIN(Date) FROM SPACEXTABLE WHERE Landing_Outcome = 'Success (ground pad)'", con) first_date_of_successful_landing**

Successful Drone Ship Landing with Payload between 4000 and 6000

- List the names of boosters that have successfully landed on a drone ship and had payload mass greater than 4000 but less than 6000:
  - **F9 FT B1022,**
    F9 FT B1026,
    F9 FT B1021.2,
    F9 FT B1031.2
- Present your query result with a short explanation here:
  - This query selects the distinct names of booster versions that successfully landed on drone ships and carried payloads between 4000 and 6000 kilograms.
  - **Boosters_names = pd.read_sql("SELECT DISTINCT Booster_Version FROM SPACEXTABLE WHERE Landing_Outcome = 'Success (drone ship)' AND PAYLOAD_MASS__KG_ BETWEEN 4000 AND 6000", con)**
  - **Booster_names**

Total Number of Successful and Failure Mission Outcomes

- Calculate the total number of successful and failure mission outcomes:
  - **Failure – 1**
  - **Success – 100**
- Present your query result with a short explanation here:
  - This query selects mission outcomes from the SPACEXTABLE, categorizes them based on the first 7 characters (likely 'Success' or 'Failure'), counts the occurrences of each category, and then presents the results.
  - **total_success_and_failure_outcomes = pd.read_sql("SELECT SUBSTR(Mission_Outcome,1,7) as Mission_Outcome, COUNT(*) from SPACEXTABLE GROUP By 1", con)**
  - **total_success_and_failure_outcomes**

Boosters Carried Maximum Payload

- List the names of the booster which have carried the maximum payload mass:
  - **F9 B5 B1048.4, 1049.4, 1051.3, 1056.4, 1048.5, 1051.4**
  - **F9 B5 B1049.5, 1060.2, 1058.3, 1051.6, 1060.3, 1049.7**
- Present your query result with a short explanation here:
  - This query selects the booster versions that have carried the maximum payload mass.
  - **booster_versions_names = pd.read_sql("SELECT DISTINCT Booster_Version FROM SPACEXTABLE WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTABLE)", con)**
  - **booster_versions_names**

2015 Failed Landing Outcomes in Drone Ship

- List the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015:
  - **Month Landing_Outcome Booster_Version Launch_Site**
  - **0 01 Failure (drone ship) F9 v1.1 B1012 CCAFS LC-40**
  - **1 04 Failure (drone ship) F9 v1.1 B1015 CCAFS LC-40**
  - Present your query result with a short explanation here:
  - This query selects and presents details of SpaceX missions that experienced drone ship landing failures in 2015.
  - **month_names = pd.read_sql("SELECT substr(Date, 6, 2) AS month, Landing_Outcome, Booster_Version, Launch_Site FROM SPACEXTABLE WHERE substr(Date, 1, 4) = '2015' AND Landing_Outcome = 'Failure (drone ship)'", con)**
  - **month_names**

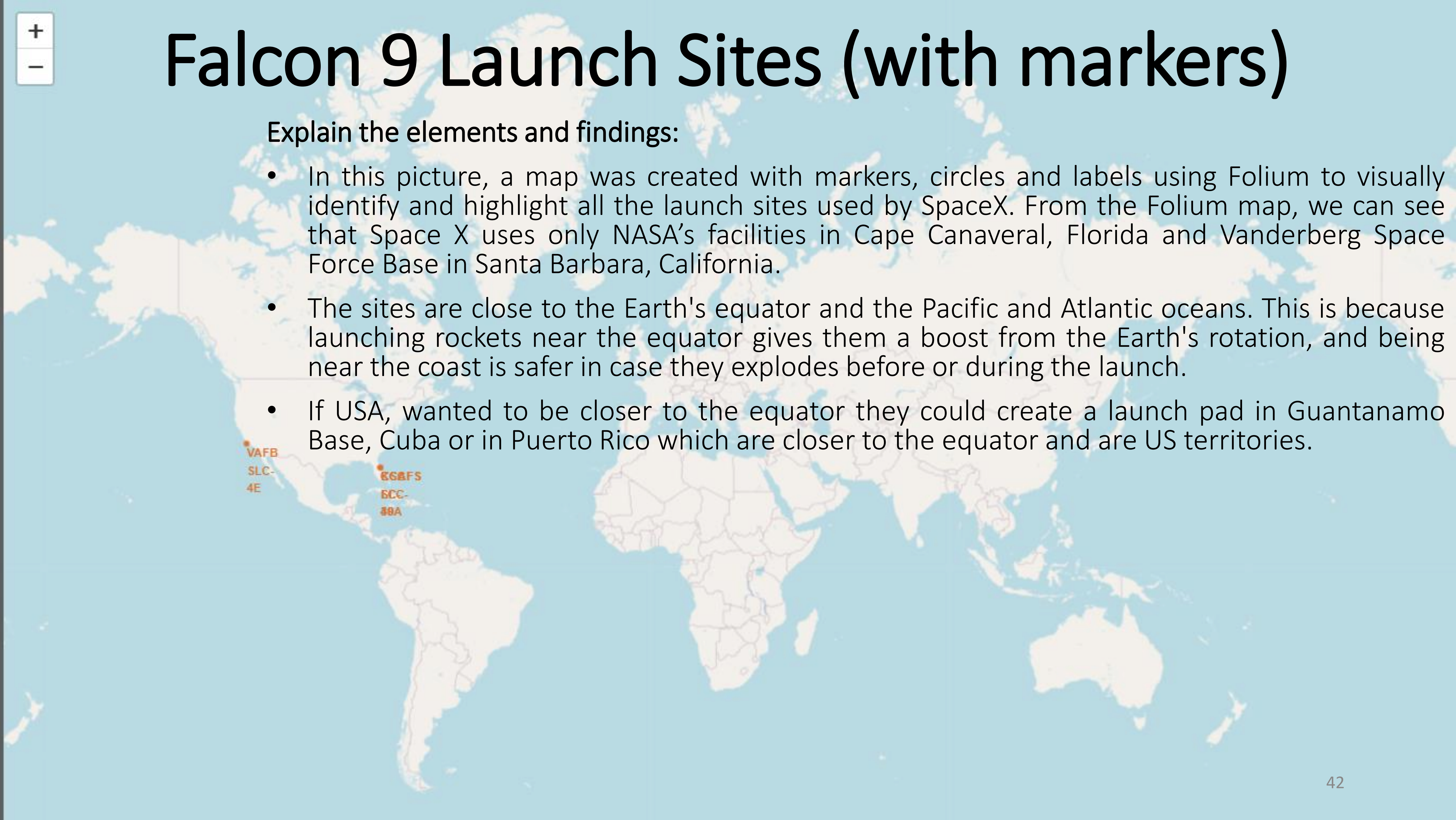
Ranking Landing Outcomes Between 2010-06-04 and 2017-03-20

- Rank the count of landing outcomes (such as Failure (drone ship) or Success (ground pad)) between the date 2010-06-04 and 2017-03-20, in descending order:
  - **No attempt: 10, Success (drone ship): 5, Failure (drone ship): 5, Success(ground pad): 3, Controlled (ocean): 3, Uncontrolled (ocean): 2, Precluded(drone ship): 1**
- Present your query result with a short explanation here:
  - This query selects and ranks the most common landing outcomes for SpaceX launches within a specified timeframe.
  - **ranking_landing_outcomes = pd.read_sql("SELECT Landing_Outcome, COUNT(*) FROM SPACEXTABLE WHERE Date BETWEEN '2011-06-04' AND '2017-03-20' GROUP BY Landing_Outcome ORDER BY 2 DESC", con)**
  - **ranking_landing_outcomes**

Section 3

Launch Sites Proximities Analysis

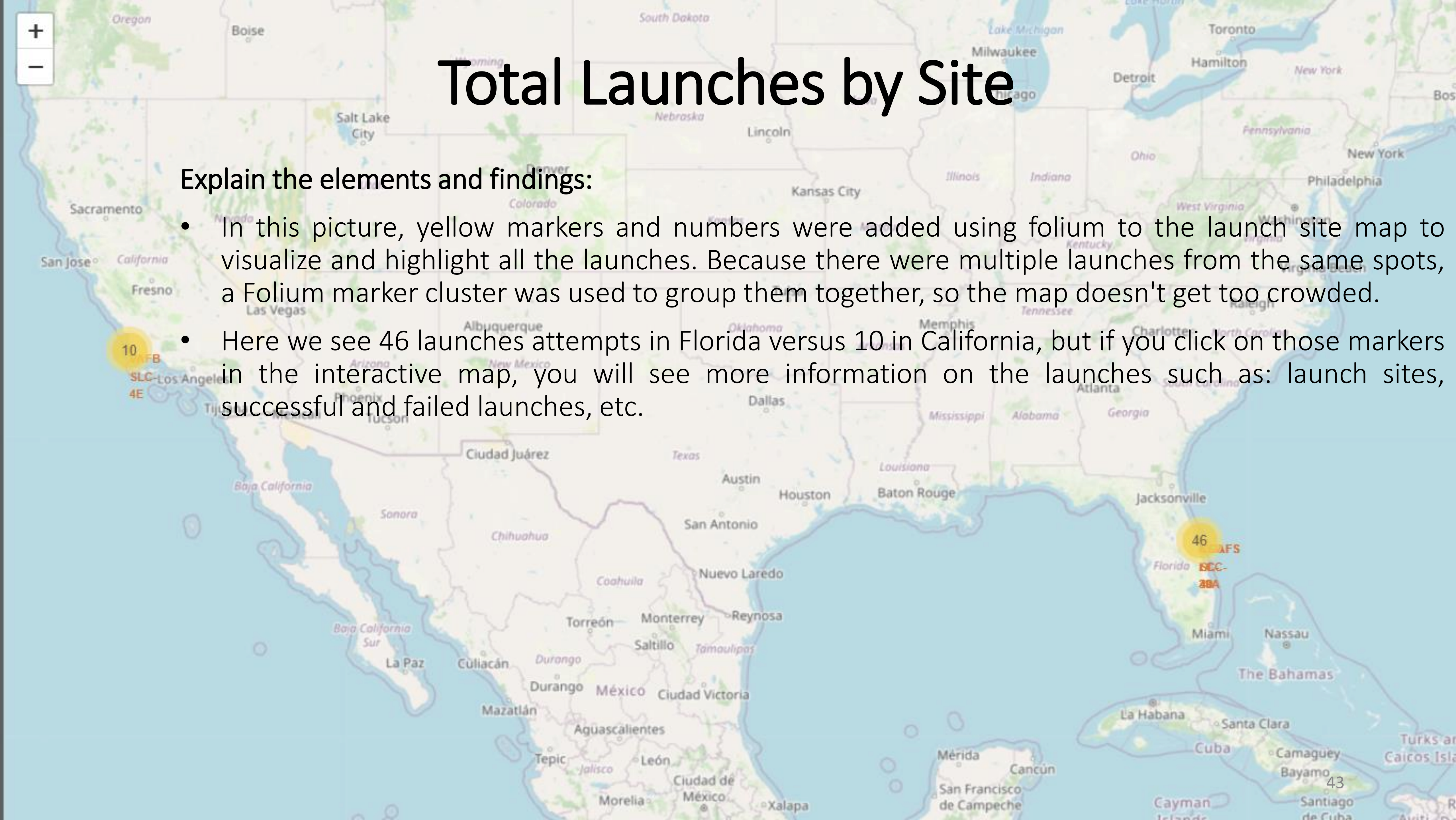
Falcon 9 Launch Sites (with markers)



Explain the elements and findings:

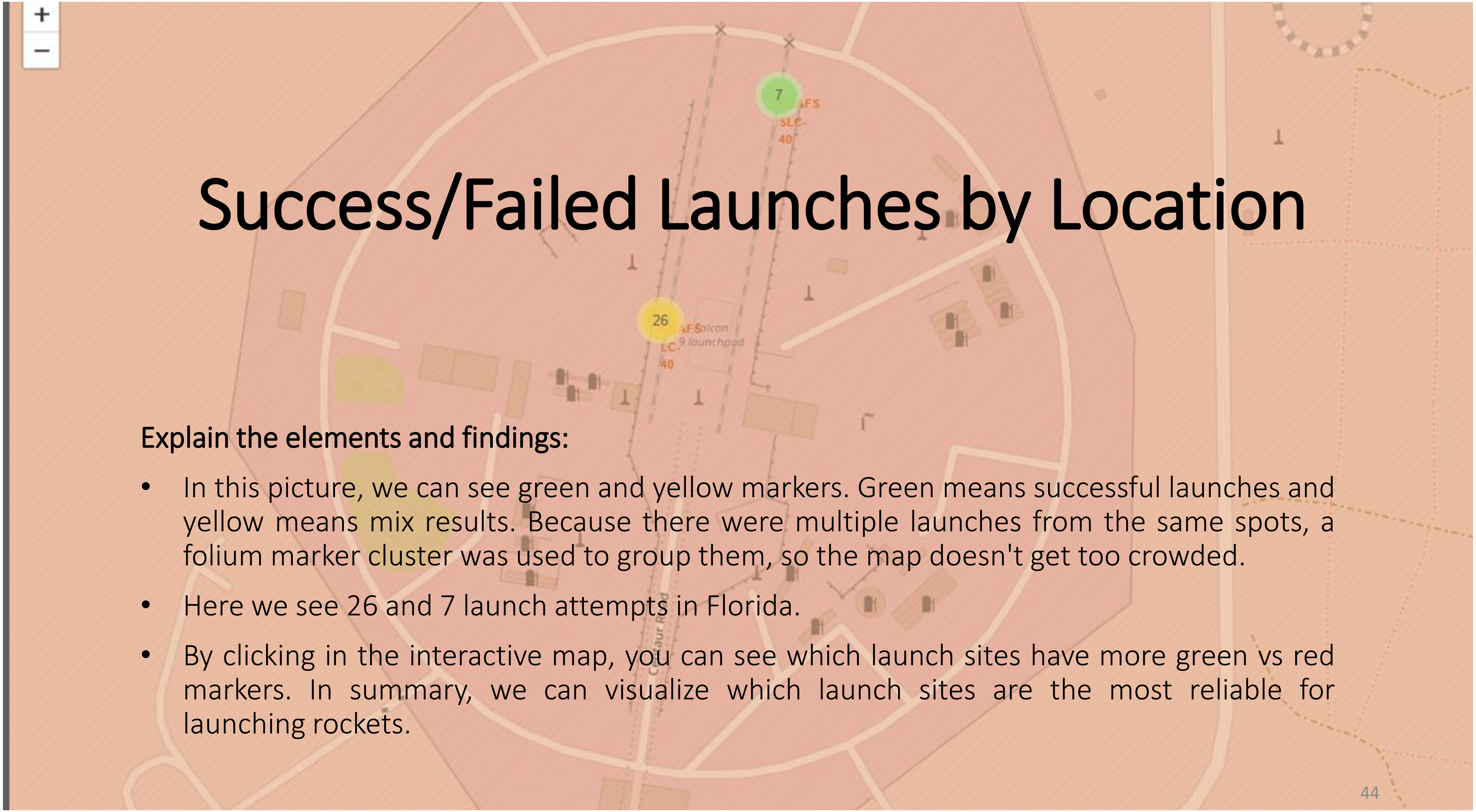
- In this picture, a map was created with markers, circles and labels using Folium to visually identify and highlight all the launch sites used by SpaceX. From the Folium map, we can see that Space X uses only NASA’s facilities in Cape Canaveral, Florida and Vanderberg Space Force Base in Santa Barbara, California.
- The sites are close to the Earth's equator and the Pacific and Atlantic oceans. This is because launching rockets near the equator gives them a boost from the Earth's rotation, and being near the coast is safer in case they explodes before or during the launch.
- If USA, wanted to be closer to the equator they could create a launch pad in Guantanamo Base, Cuba or in Puerto Rico which are closer to the equator and are US territories.

Total Launches by Site



Explain the elements and findings:

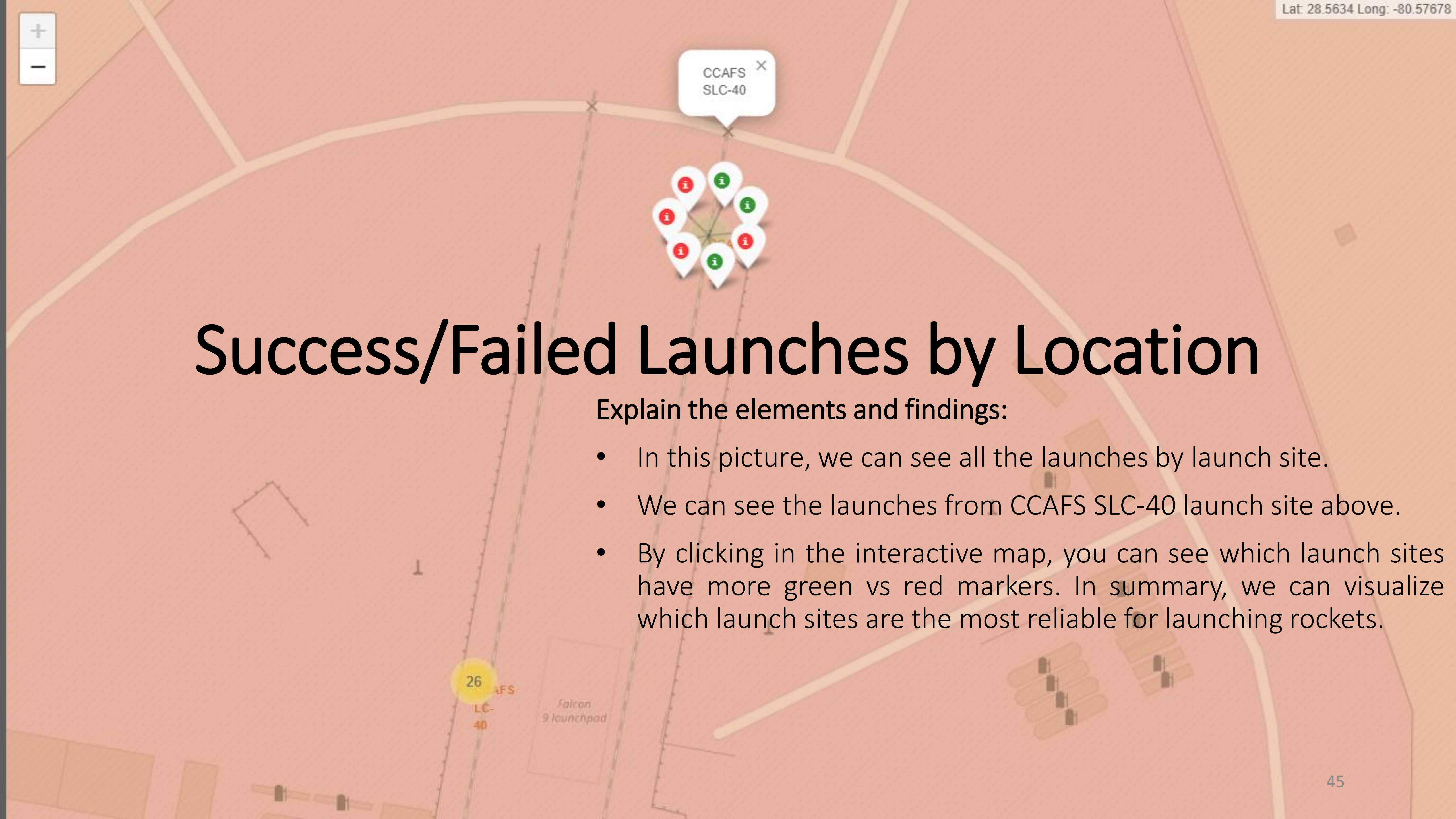
- In this picture, yellow markers and numbers were added using folium to the launch site map to visualize and highlight all the launches. Because there were multiple launches from the same spots, a Folium marker cluster was used to group them together, so the map doesn't get too crowded.
- Here we see 46 launches attempts in Florida versus 10 in California, but if you click on those markers in the interactive map, you will see more information on the launches such as: launch sites, successful and failed launches, etc.



Success/Failed Launches by Location

Explain the elements and findings:

- In this picture, we can see green and yellow markers. Green means successful launches and yellow means mix results. Because there were multiple launches from the same spots, a folium marker cluster was used to group them, so the map doesn't get too crowded.
- Here we see 26 and 7 launch attempts in Florida.
- By clicking in the interactive map, you can see which launch sites have more green vs red markers. In summary, we can visualize which launch sites are the most reliable for launching rockets.

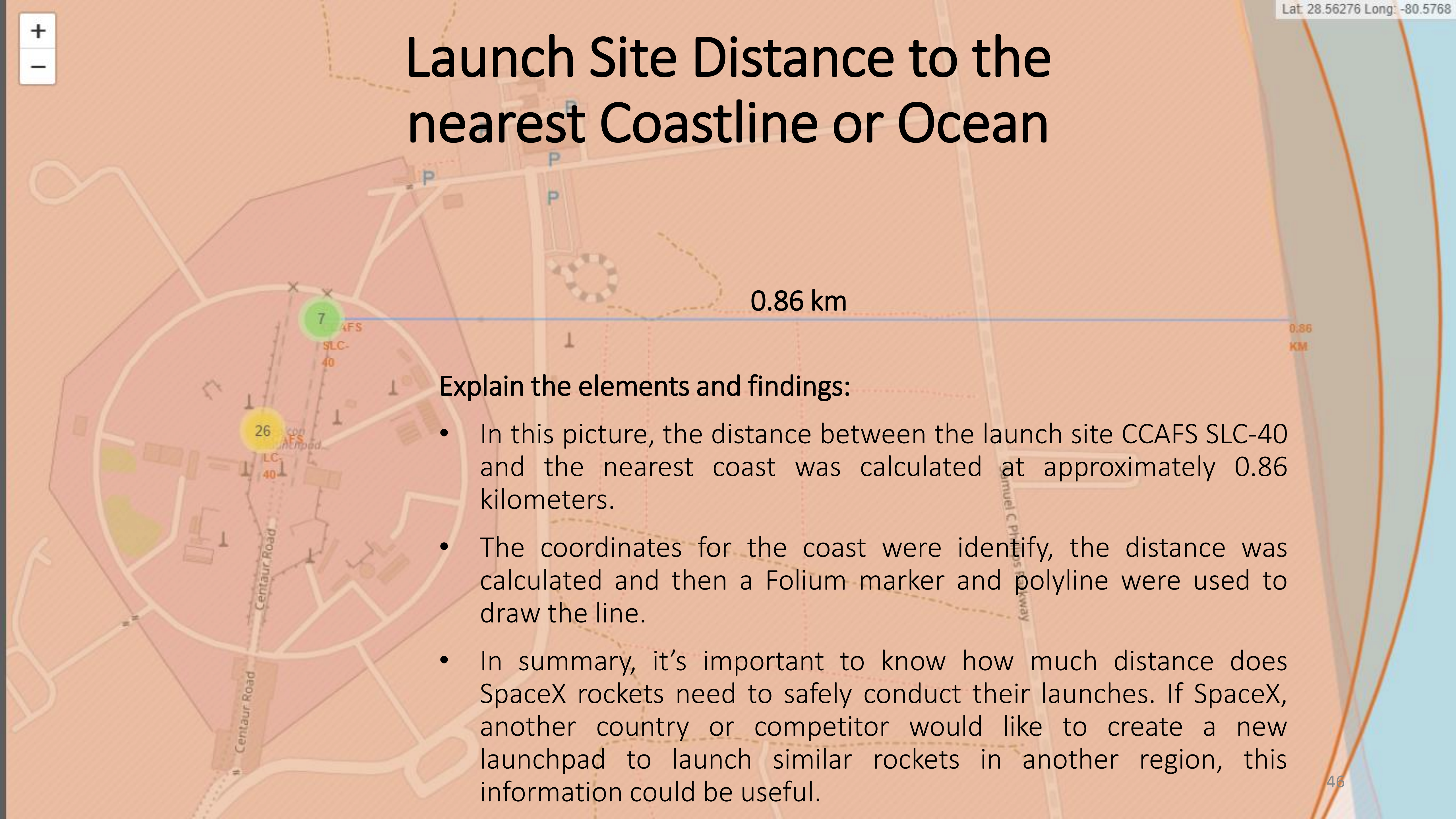


Success/Failed Launches by Location

Explain the elements and findings:

- In this picture, we can see all the launches by launch site.
- We can see the launches from CCAFS SLC-40 launch site above.
- By clicking in the interactive map, you can see which launch sites have more green vs red markers. In summary, we can visualize which launch sites are the most reliable for launching rockets.

Launch Site Distance to the nearest Coastline or Ocean

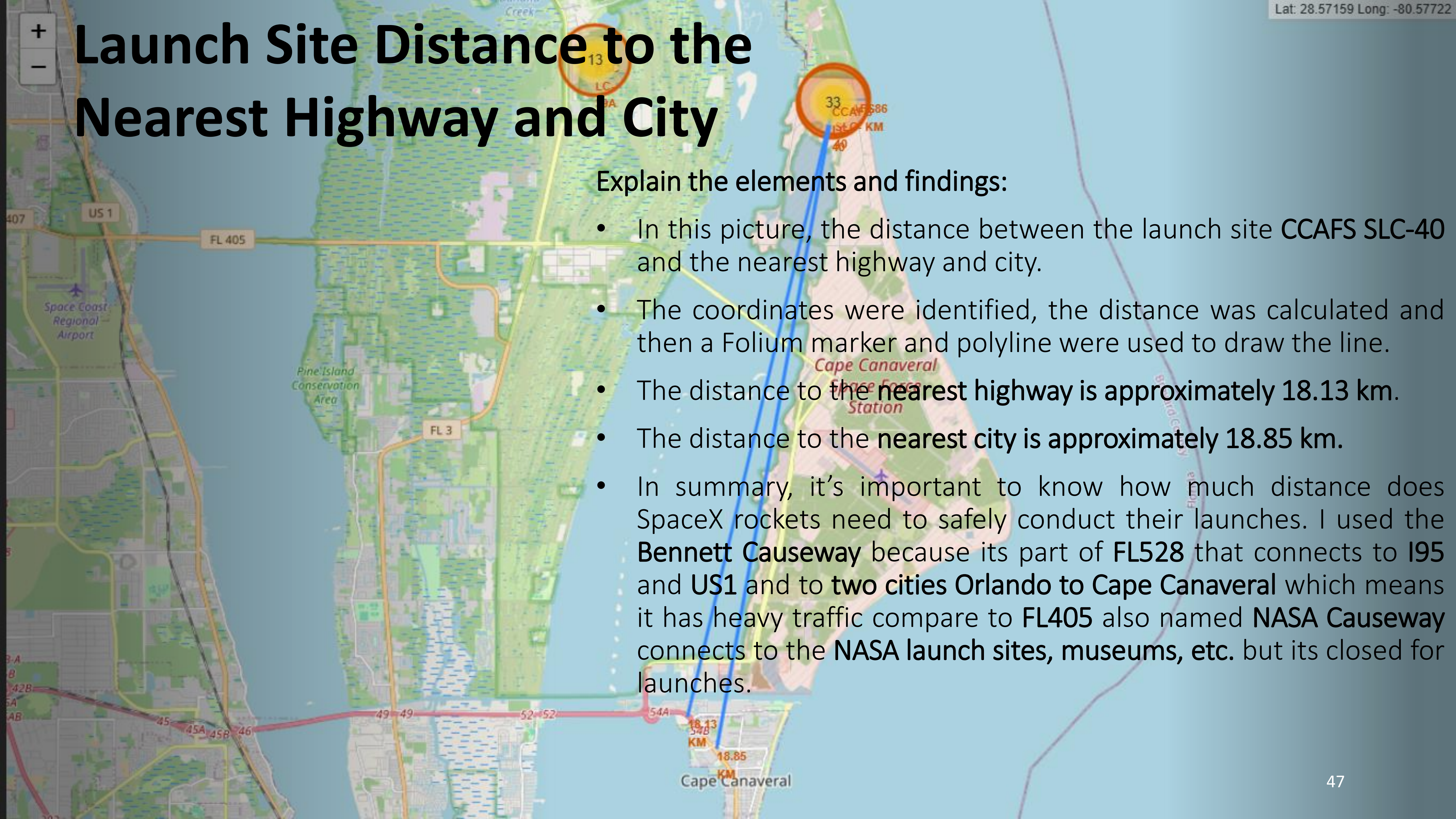


0.86 km

Explain the elements and findings:

- In this picture, the distance between the launch site CCAFS SLC-40 and the nearest coast was calculated at approximately 0.86 kilometers.
- The coordinates for the coast were identify, the distance was calculated and then a Folium marker and polyline were used to draw the line.
- In summary, it's important to know how much distance does SpaceX rockets need to safely conduct their launches. If SpaceX, another country or competitor would like to create a new launchpad to launch similar rockets in another region, this information could be useful.

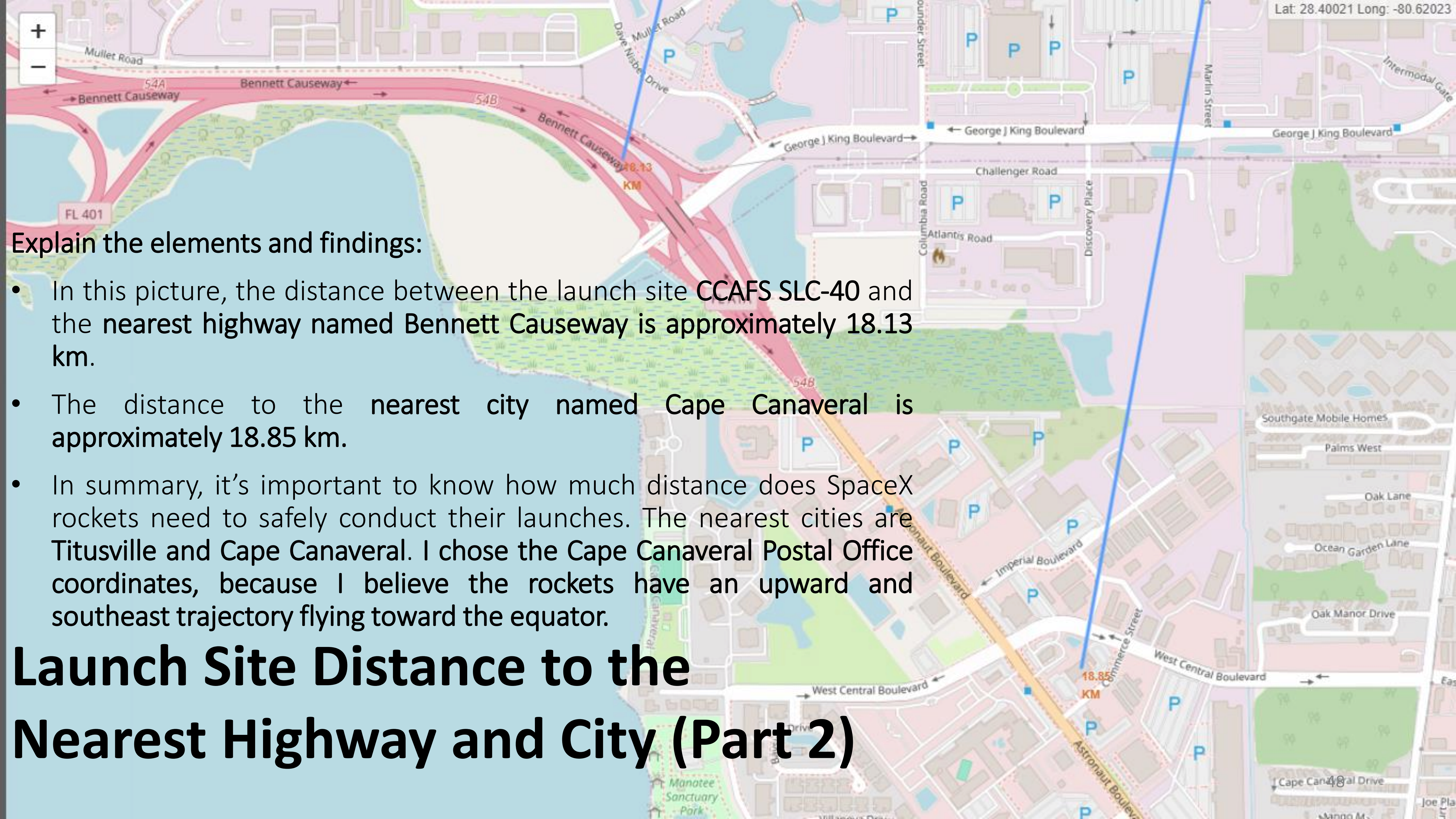
Launch Site Distance to the Nearest Highway and City



Explain the elements and findings:

- In this picture, the distance between the launch site **CCAFS SLC-40** and the nearest highway and city.
- The coordinates were identified, the distance was calculated and then a Folium marker and polyline were used to draw the line.
- The distance to the **nearest highway is approximately 18.13 km**.
- The distance to the **nearest city is approximately 18.85 km.**
- In summary, it's important to know how much distance does SpaceX rockets need to safely conduct their launches. I used the **Bennett Causeway** because its part of **FL528** that connects to **I95** and **US1** and to **two cities Orlando to Cape Canaveral** which means it has heavy traffic compare to **FL405** also named **NASA Causeway** connects to the **NASA launch sites, museums, etc.** but its closed for launches.

Launch Site Distance to the Nearest Highway and City (Part 2)



Explain the elements and findings:

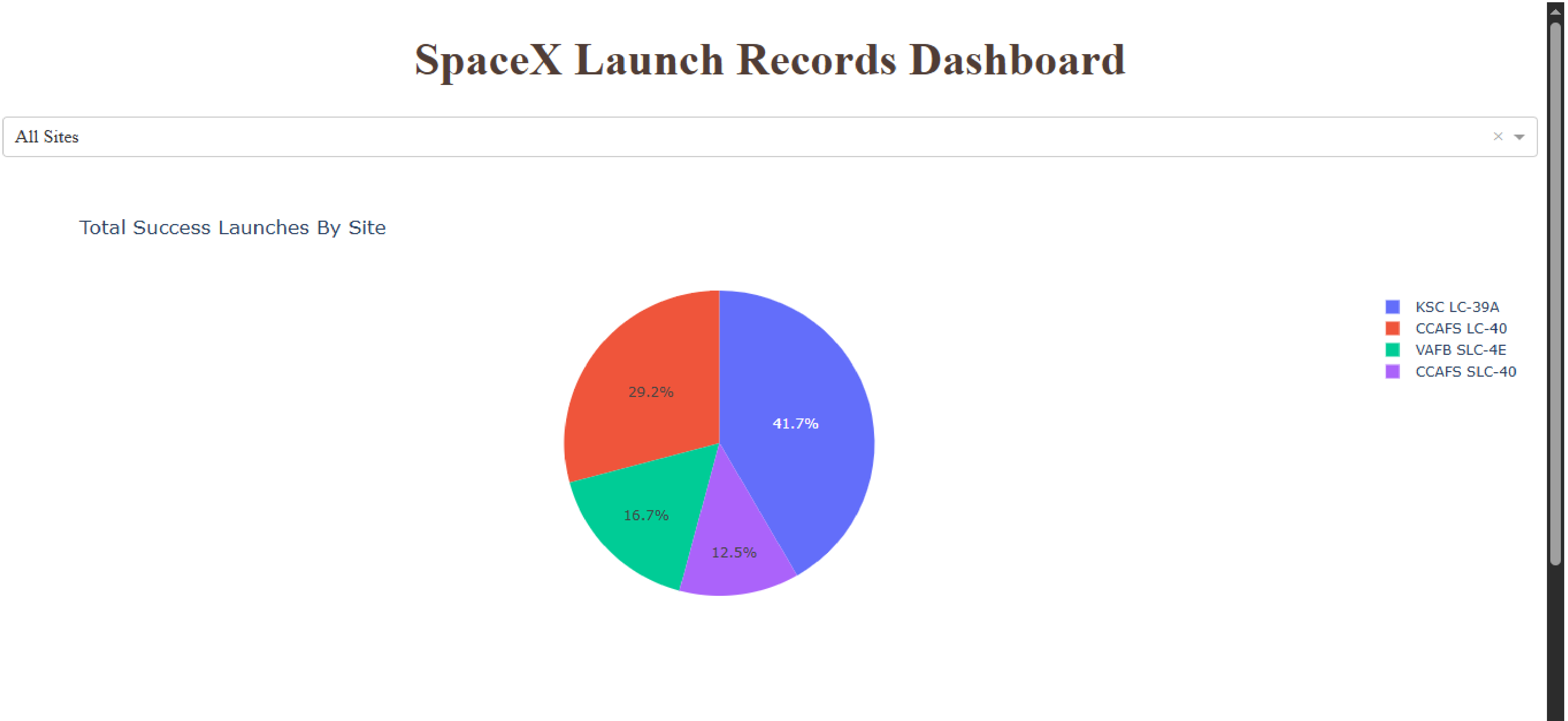
- In this picture, the distance between the launch site CCAFS SLC-40 and the nearest highway named Bennett Causeway is approximately 18.13 km.
- The distance to the nearest city named Cape Canaveral is approximately 18.85 km.
- In summary, it's important to know how much distance does SpaceX rockets need to safely conduct their launches. The nearest cities are Titusville and Cape Canaveral. I chose the Cape Canaveral Postal Office coordinates, because I believe the rockets have an upward and southeast trajectory flying toward the equator.

Section 4

Build a Dashboard with Plotly Dash

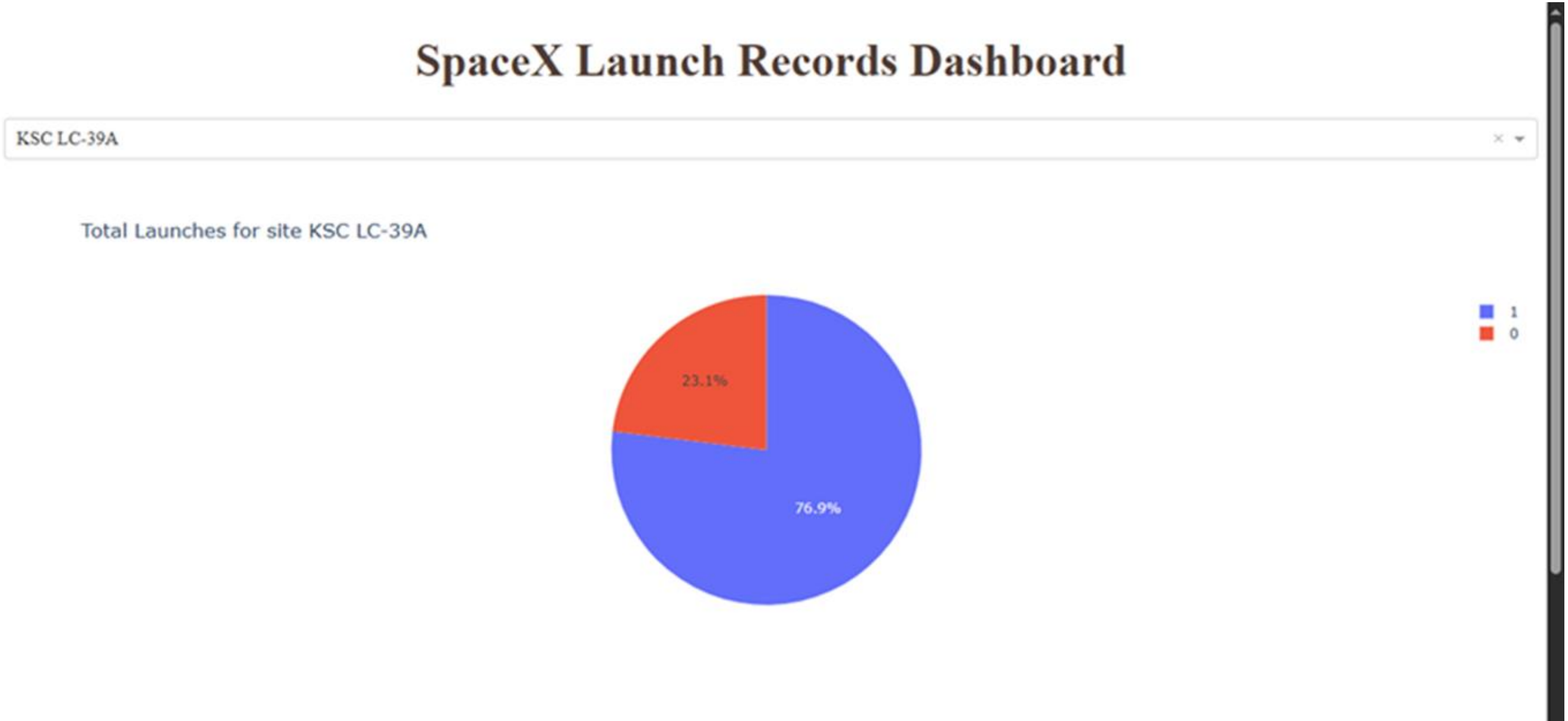
Launch Success Rate for all Sites

- The data shows the most successful launch site was **KSC LC-39A (Kennedy Space Center Launch Complex 39A) with 41.7% success rate compare to other sites.**



Launch Site with Highest Success Rate

- The data shows **76.9%** of all launches in **KSC LC-39A (Kennedy Space Center Launch Complex 39A) had a successful landing outcome.**



Payload Mass vs. Launch Outcome for all Sites

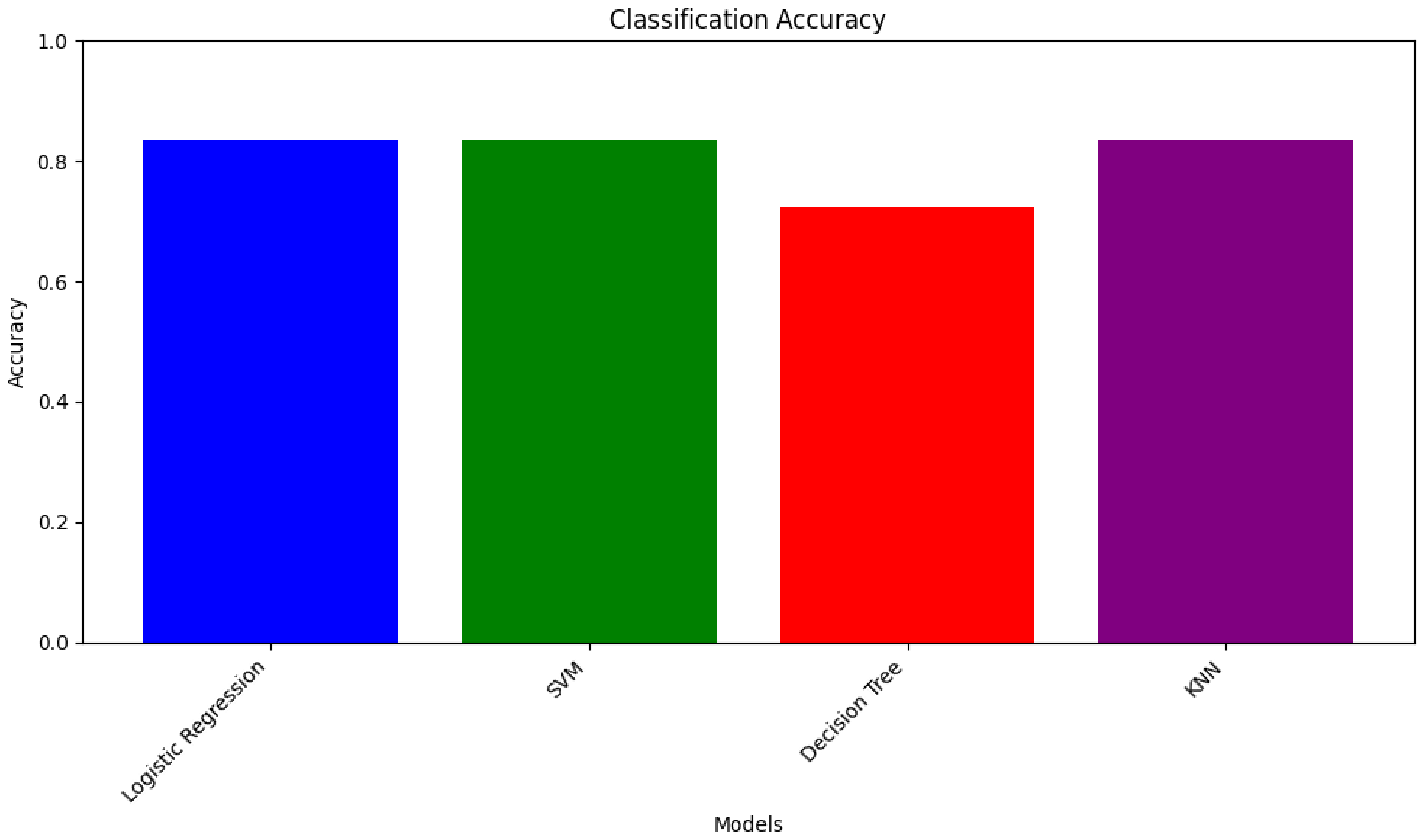
- The data shows that **Payload Mass** between **2,000 and 5,000 kilograms(kg)** have a higher success rate in launch outcomes for all sites.

Section 5

Predictive Analysis (Classification)

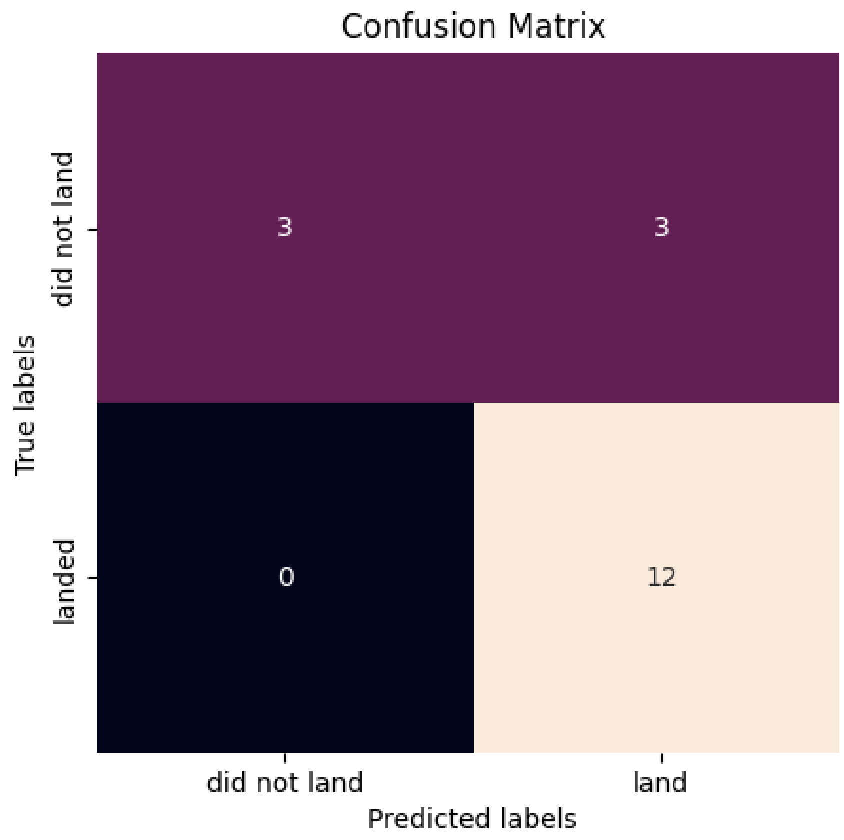
Classification Accuracy

- Logistic Regression: 0.8333333333333334 or **83.33%**
- SVM: 0.8333333333333334 or **83.33%**
- KNN: 0.8333333333333334 or **83.33%**
- Decision Tree: 0.6111111111111112 or **61.11%**



Confusion Matrix

- Examining the confusion matrix, we see that logistic regression can distinguish between two different classes. We see that the problem is false positives.
- **True Positive – 12 (True label is landed, Predicted label is also landed)**
- **True Negative – 3 (True label is not landed, Predicted label is not landed)**
- **False Positive – 3 (True label is not landed, Predicted label is landed)**
- 3 False Positives or 16.67% error in the model could be challenging for predicting future launches.



Conclusion: Results

- Key Factors: **Payload mass, orbit type, launch sites, and landing outcomes**
- Most Successful Launch Site: **KSC LC-39A: 41.7% (Kennedy Space Center Launch Complex 39A)**
- Launch Success Rate for **KSC LC-39A: 76.9%**
- Ideal Payload Mass: **Between 2,000 to 5,000 kg**
- Ideal Orbit Type: **LEO, ISS and PO**
- Best Classification Model: **Logistical Regression with 83.33% Accuracy**
- **Confusion Matrix: Logistical Regression**
- **True Positive - 12** (True label is landed, Predicted label is also landed)
- **True Negative – 3** (True label is not landed, Predicted label is not landed)
- **False Positive - 3** (True label is not landed, Predicted label is landed)

Conclusion: Key Observations

- **Key factors such as: payload mass, orbit type, and launch site** can influence Falcon 9 landing outcomes.
- **Flight numbers 60 and above** resulted mostly in successful landing outcomes. Showing an upward trend.
- **Landing Outcomes** between the **last two years of data (2019-2020)** were above **80%**. Showing an upward trend.
- Rockets with a **Payload Mass between 2,000 to 5,000 kg** have a higher successful rate.
- Rockets that used **Orbit Type LEO, ISS, and PO** resulted in mostly successful landing outcomes.
- Most Successful Launch Site: **KSC LC-39A: 41.7% (Kennedy Space Center Launch Complex 39A)**
- Most classification models such as: **Logistic Regression, SVM, and KNN** predicted results with an accuracy of **83.33%**. The exception was the **Decision Tree** model with a **61.11%.**

Appendix

- The following Github link gives you access to the Project notebooks and documentation.
- https://github.com/projectstrawberrylemonade/project1.git


This Photo by Unknown Author is licensed under CC BY-NC-ND

Thank You!