IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Saima Nisar
27 Nov, 2023

Outline

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

Executive Summary

Summary of Methodologies

- Data Collection
  - API and Web Scraping
- Data Wrangling
- EDA with Visualization
- EDA with SQL
- An Interactive Map with Folium
- A Dashboard with Plotly Dash
- Predictive Analysis

Summary of all Results

- Exploratory Data Analysis
- Interactive Analytics
- Predictive Analysis Results

Introduction

Project Background and Context

Booming Commercial Space Industry

- Companies revolutionizing space travel.

SpaceX's Competitive Edge

- Reusable Falcon 9 first stage.
- Cuts launch costs by over 60%.

Project Focus

- Predict Falcon 9 first stage landing.

Role: Data Scientists

- Working for "Space Y" startup.

Exploring Project Solutions

Objectives:

- Data-driven pricing decisions.
- Machine Learning-based reusability predictions.

Real-world Applications:

- Gain hands-on experience.

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Describe how data was collected
- Perform data wrangling
  - Describe how data was processed
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
- How to build, tune, and evaluate classification models

Data Collection API

Requesting Data

- Data collection of SpaceX launch records
- Utilize HTTP GET to the SpaceX REST API endpoint; (api.spacexdata.com/v4/launches/past) for obtaining past SpaceX launch data
- Receive JSON responses

JSON Data Processing

- Converting JSON responses into a structured pandas data frame using JSON normalization.
- Extract details of past SpaceX launches, including rocket, payload, and landing information.

Filtering Falcon 9 Launches

- Data filter to exclusively include Falcon 9 launches.

Handling Missing Values

- Deal with NULL values in the data, particularly in the 'PayloadMass' column, by calculating the mean and replacing NULLs.

Out[28]:

	FlightNumber	Date	BoosterVersion	PayloadMass	Orbit	LaunchSite	Outcome	Flights	GridFins	Reused	Legs	La
4	1	2010-06-04	Falcon 9	NaN	LEO	CCSFS SLC 40	None None	1	False	False	False	
5	2	2012-05-22	Falcon 9	525.0	LEO	CCSFS SLC 40	None None	1	False	False	False	
6	3	2013-03-01	Falcon 9	677.0	ISS	CCSFS SLC 40	None None	1	False	False	False	
7	4	2013-09-29	Falcon 9	500.0	PO	VAFB SLC 4E	False Ocean	1	False	False	False	
8	5	2013-12-03	Falcon 9	3170.0	GTO	CCSFS SLC 40	None None	1	False	False	False	
...	...	...	...	...	...	...	...	...	...	...	...	
89	86	2020-09-03	Falcon 9	15600.0	VLEO	KSC LC 39A	True ASDS	2	True	True	True	5e9e3032383ecb6t
90	87	2020-10-06	Falcon 9	15600.0	VLEO	KSC LC 39A	True ASDS	3	True	True	True	5e9e3032383ecb6t
91	88	2020-10-18	Falcon 9	15600.0	VLEO	KSC LC 39A	True ASDS	6	True	True	True	5e9e3032383ecb6t
92	89	2020-10-24	Falcon 9	15600.0	VLEO	CCSFS SLC 40	True ASDS	3	True	True	True	5e9e3033383ecbb9
93	90	2020-11-05	Falcon 9	3681.0	MEO	CCSFS SLC 40	True ASDS	1	True	False	True	5e9e3032383ecb6t

Falcon 9 Launch Data

Missing Values

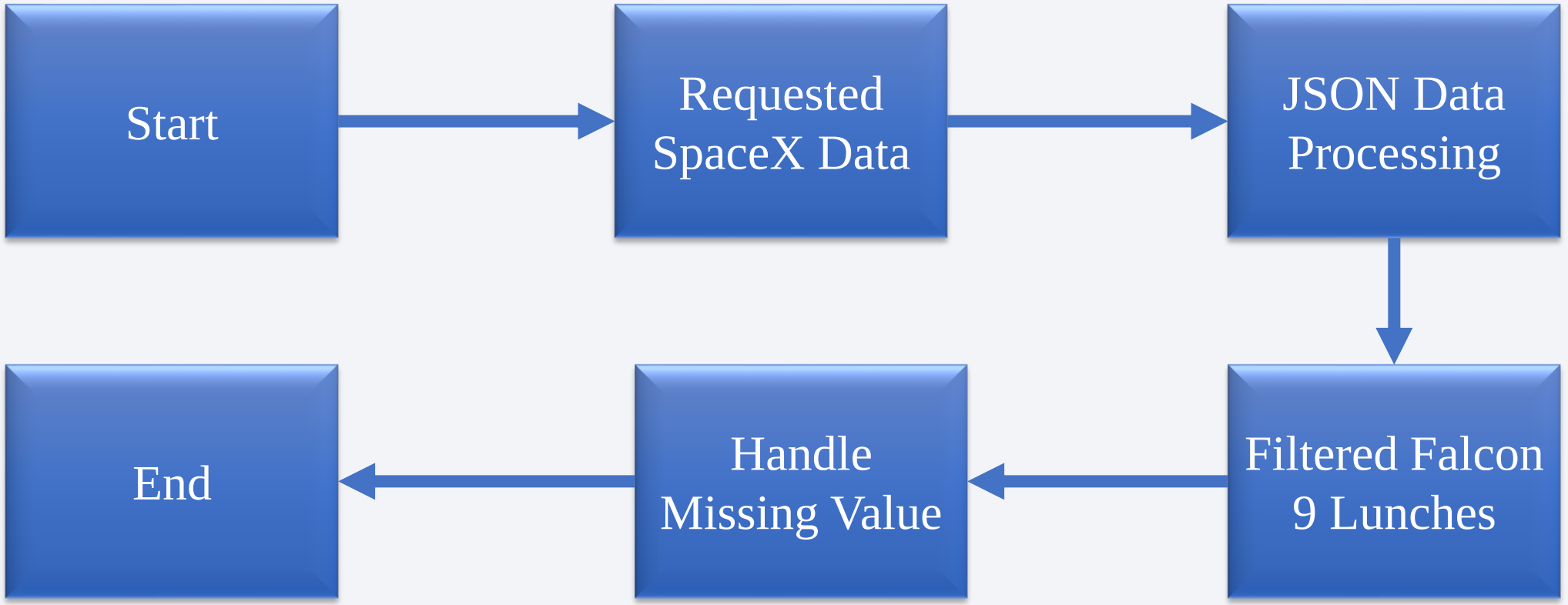
```
In [31]: data_falcon9.isnull().sum()

Out[31]: FlightNumber      0
         Date              0
         BoosterVersion    0
         PayloadMass       0
         Orbit             0
         LaunchSite        0
         Outcome           0
         Flights           0
         GridFins          0
         Reused            0
         Legs              0
         LandingPad       26
         Block             0
         ReusedCount       0
         Serial            0
         Longitude         0
         Latitude          0
         dtype: int64
```

Now we should have no missing values in our dataset except for in `LandingPad`.

GitHub

Data Collection API



Data Collection – Web Scraping

Web Scraping with BeautifulSoup

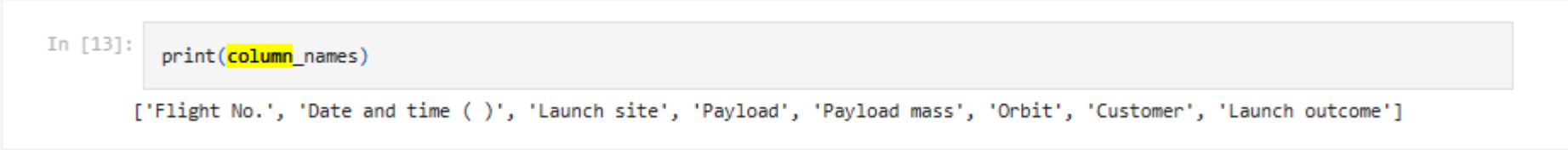
- Extract data from HTML tables on related Wiki pages.
- Gather Falcon 9 launch records.
- Utilize Python's BeautifulSoup package for web scraping

Extracting Column/Variable Names

- Parsing the HTML table header to retrieve all column or variable names present in the launch records

Creating Data Frame

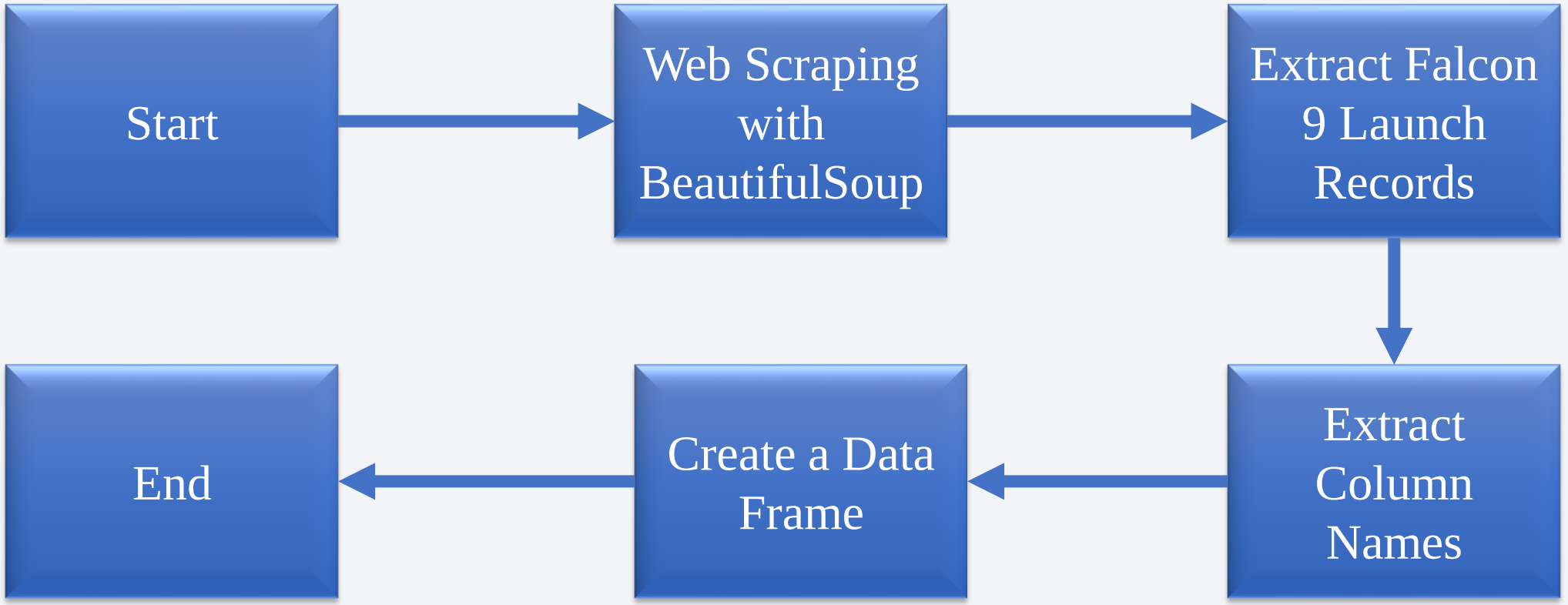
- Generating a Pandas data frame through the parsing of HTML tables.
- Facilitating the organization and analysis of the collected data.



Extracted Column Names

GitHub

Data Collection – Web Scraping



Data Wrangling

Number of Launches per Site

- Calculate the number of launches from each SpaceX launch site
- Example Output:
  - CCAFS SLC 40: 55 launches
  - KSC LC 39A: 22 launches
  - VAFB SLC 4E: 13 launches

Number and Occurrence of Orbits

- Calculate the number and occurrence of different orbit types for SpaceX launches.

Number and Occurrence of Mission Outcomes

- Calculate the number and occurrence of mission outcomes for the different orbits.

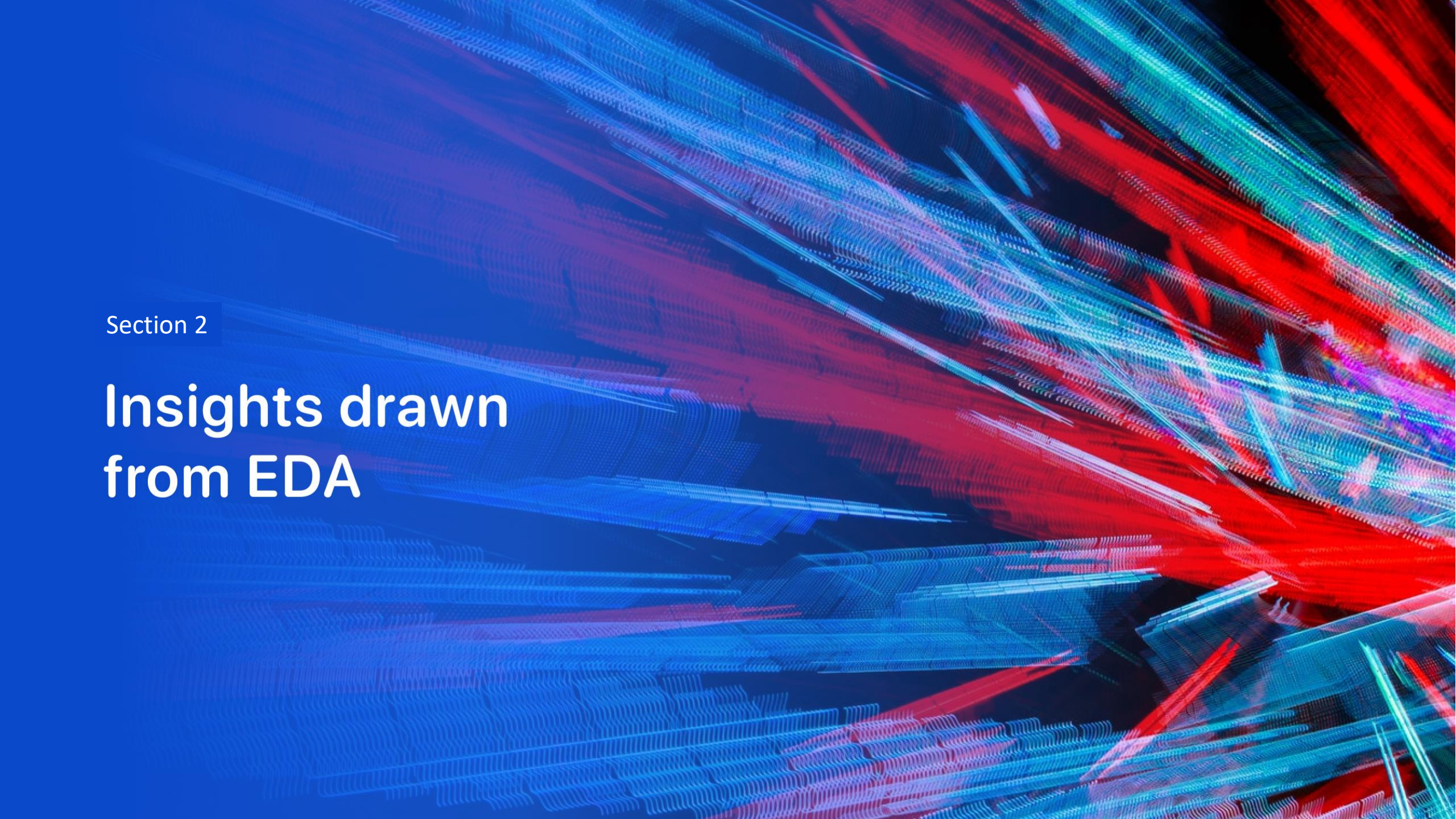
Creating Landing Outcomes Labels

- Create a classification variable, “landing_class”, based on the outcomes of each launch
- Details:
  - If the outcomes are in the set “bad_outcomes”, assign a value of zero (0)
  - Otherwise, assign a value of one (1)

Data Export:

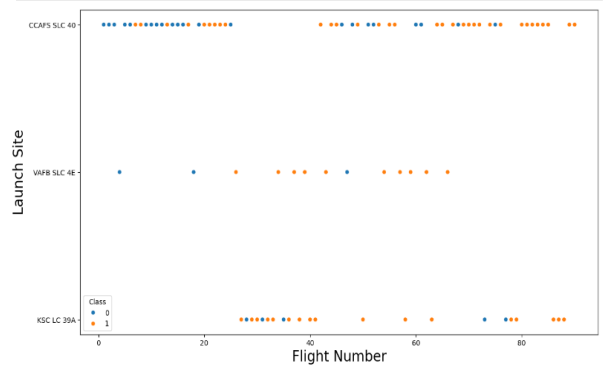
- Export the dataset with the created label to a CSV file for use in the next section.

GitHub

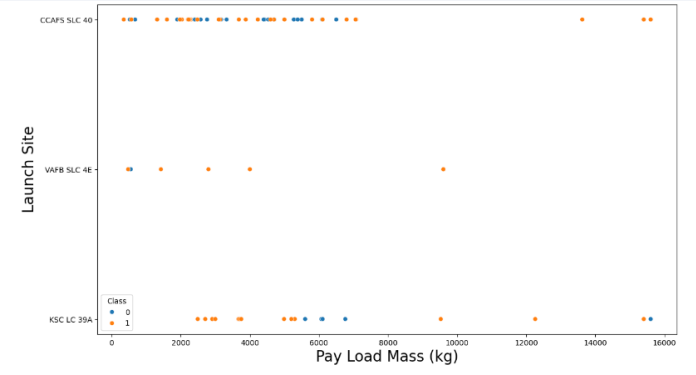

Section 2
Insights drawn from EDA

EDA with Data Visualization

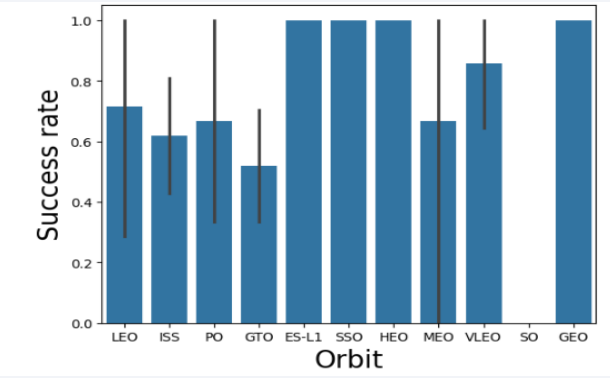
Relationship Between Flight Number and Launch Site



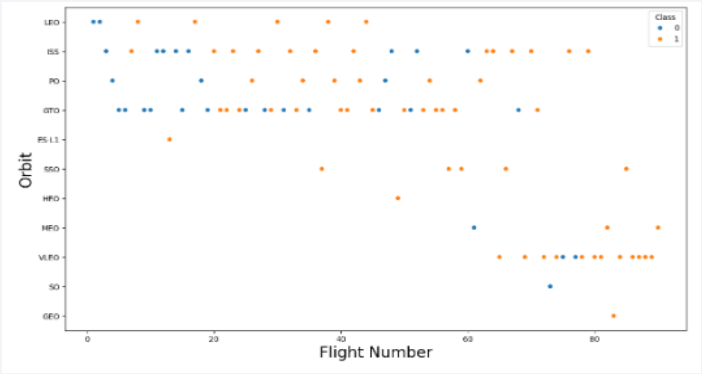
Relationship Between Payload and Launch Site



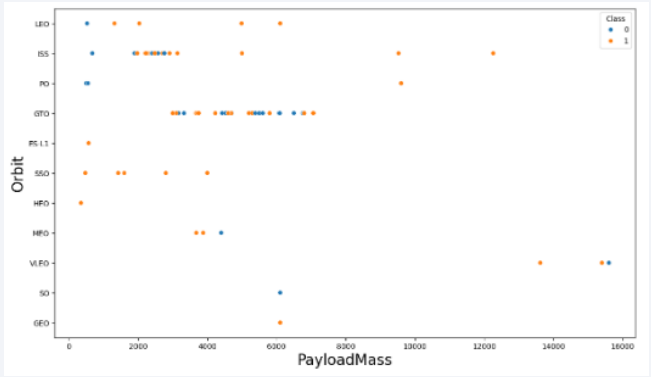
Relationship Between Success Rate and Orbit Type



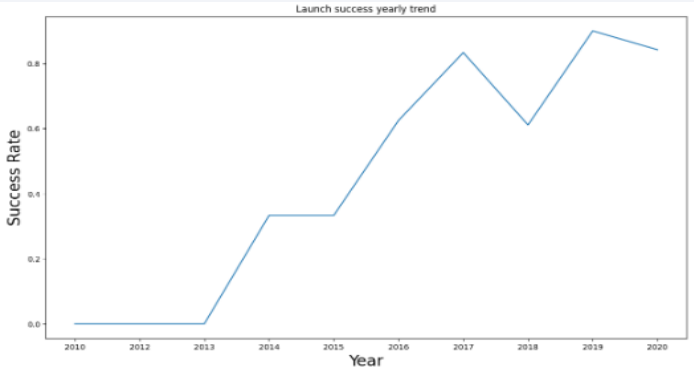
Relationship Between Flight Number and Orbit Type



Relationship Between Payload and Orbit type



Launch Success Yearly Trend



GitHub

EDA with SQL

Unique Launche Sites

- Display the names of the unique launch sites in the space mission
- Launch Sites:
  - ❑ CCAFS LC-40
  - ❑ VAFB SLC-4E
  - ❑ KSC LC-39A

Total Payload Mass by NASA (CRS) Boosters

- The total payload mass carried by boosters launched by NASA (CRS) is 45,596 kilograms.

Average Payload Mass for F9 v1.1 Boosters

- The average payload mass for booster version F9 v1.1 is approximately 2534.67 kilograms

Date of First Successful Landing on Ground Pad

- The first successful landing on a ground pad occurred on December 22, 2015

Boosters with Successful Landings on Drone Ship and Payload Mass

- The boosters listed below achieved successful landings on a drone ship and had payload masses greater than 4000 kilograms but less than 6000 kilograms.

Boosters:

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

EDA with SQL

Total Number of Successful and Failed Mission Outcomes

- Export Number of Successful Mission Outcomes: Value: 100
- Number of Failure Mission Outcomes: Value: 1

Boosters with Maximum Payload Mass

- 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

Records with Failure Landing Outcomes on Drone Ship for 2015

- The F9 v1.1 B1012, CCAFS LC-40, Failure (drone ship), 2015-01-10
- F9 v1.1 B1015, CCAFS LC-40, Failure (drone ship), 2015-04-14

Ranking of Landing Outcomes (2010-06-04 to 2017-03-20)

- No attempt: 10
- Success (drone ship): 5
- Failure (drone ship): 5
- Success (ground pad): 3
- Controlled (ocean): 3
- Uncontrolled (ocean): 2
- Failure (parachute): 2
- Precluded (drone ship): 1

Section 3

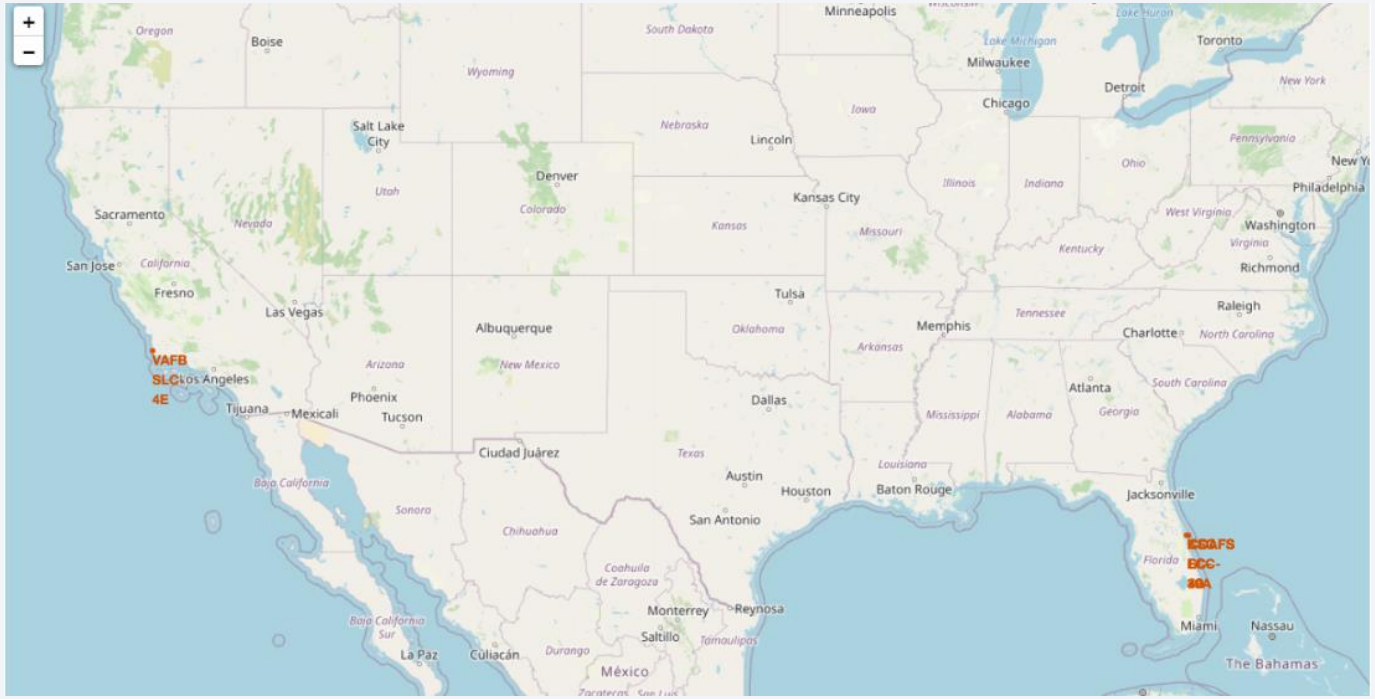
Launch Sites Proximities Analysis

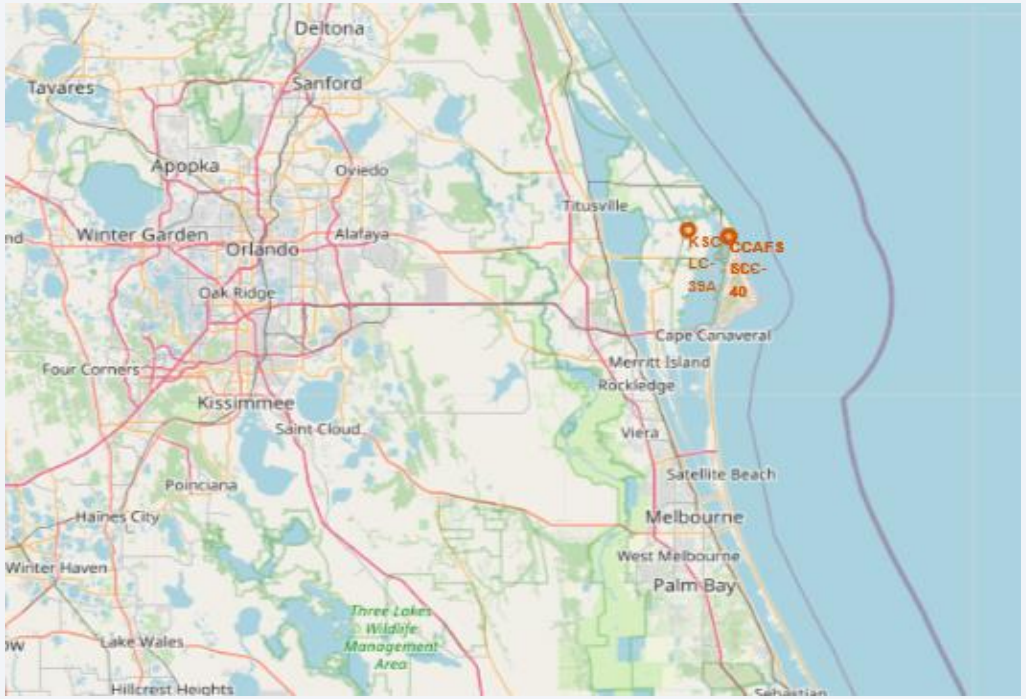
Build an Interactive Map with Folium

Launches Site Distribution

- Map overview:
  - ❑ Left Map: Displays all launch sites relative to the US map.
  - ❑ Right Map: Focuses on two launch sites in Florida, showcasing their proximity.

- Launch Site Proximity
  - ❑ The right map emphasizes the close proximity of two Florida launch sites.
  - ❑ Noteworthy: All launch sites strategically located near the ocean.



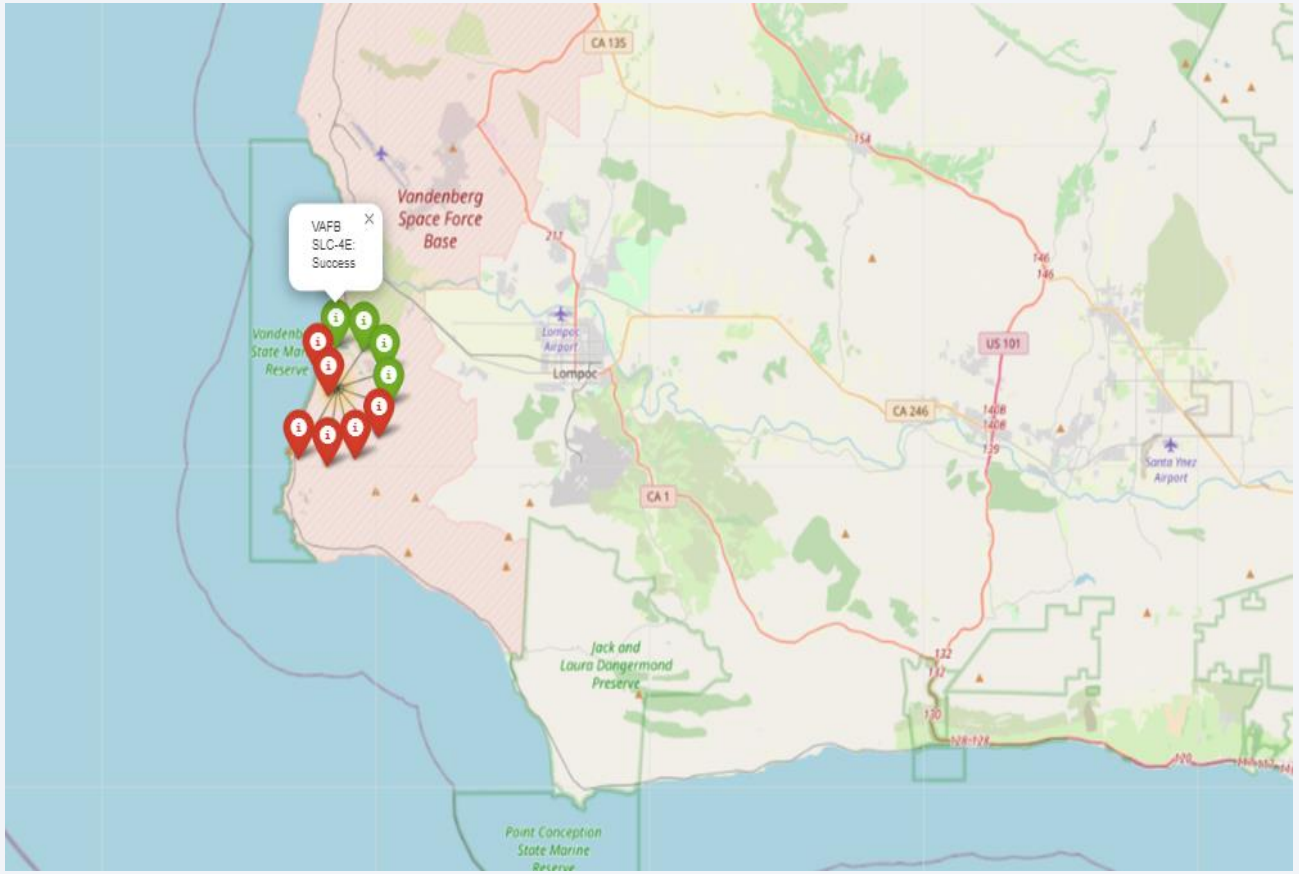


GitHub

Build an Interactive Map with Folium

Color-Coded Launch Markers

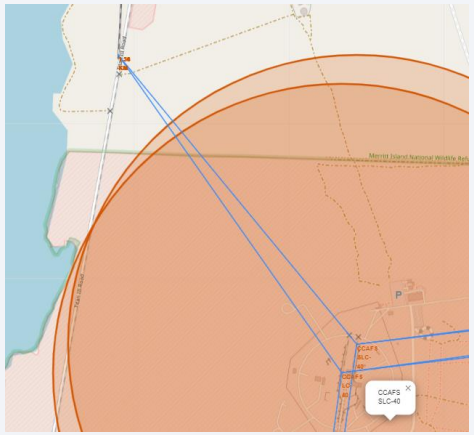
- Map overview:
  - ❑ Interactive Clusters: Clickable clusters on the Folium map.
  - ❑ Color-Coded Markers: Distinguish between successful (green) and failed (red) landings.
- Example Observation
  - ❑ VAFB SLC-4E: Demonstrates 4 successful landings and 6 failed landings.



Build an Interactive Map with Folium

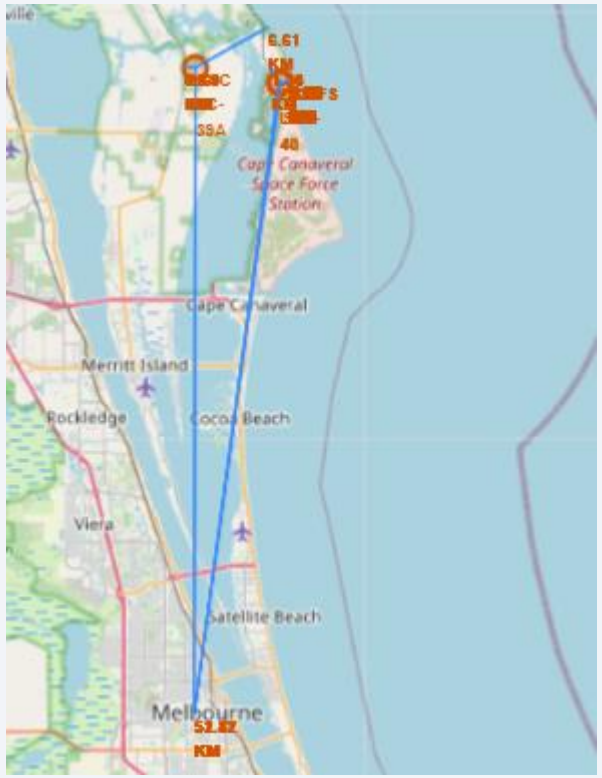
Key Location Proximities

- Example Site: CCAFS SLC-40
- Railway Proximity: Launch sites situated in close proximity to railways for efficient large-scale transportation.
- Highway Accessibility: Strategic location near highways for both human and supply transport.
- Coastal Placement: Launch sites strategically positioned near coasts for launch failures to land in the sea, avoiding densely populated areas.
- City Distance: Relatively far from cities to minimize risk in case of launch failures.

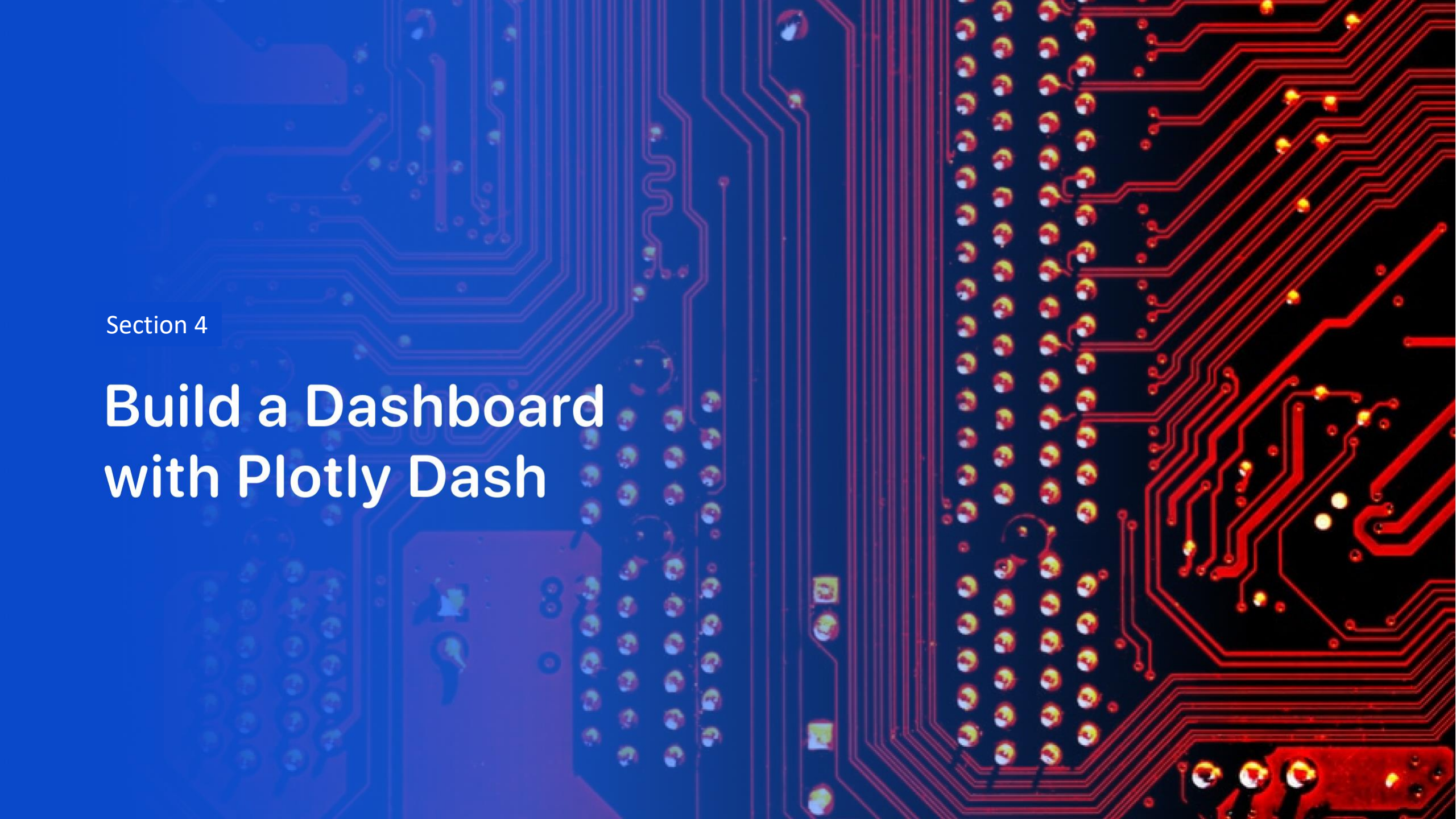


Distance from Railway

Distance from Cost and Highway



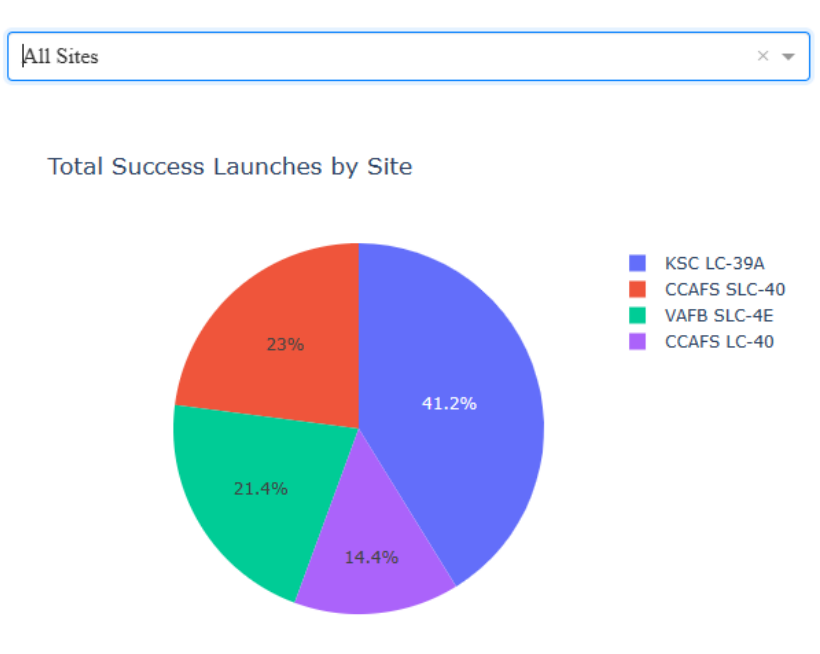
Distance from City


Section 4
Build a Dashboard with Plotly Dash

Build a Dashboard with Plotly Dash

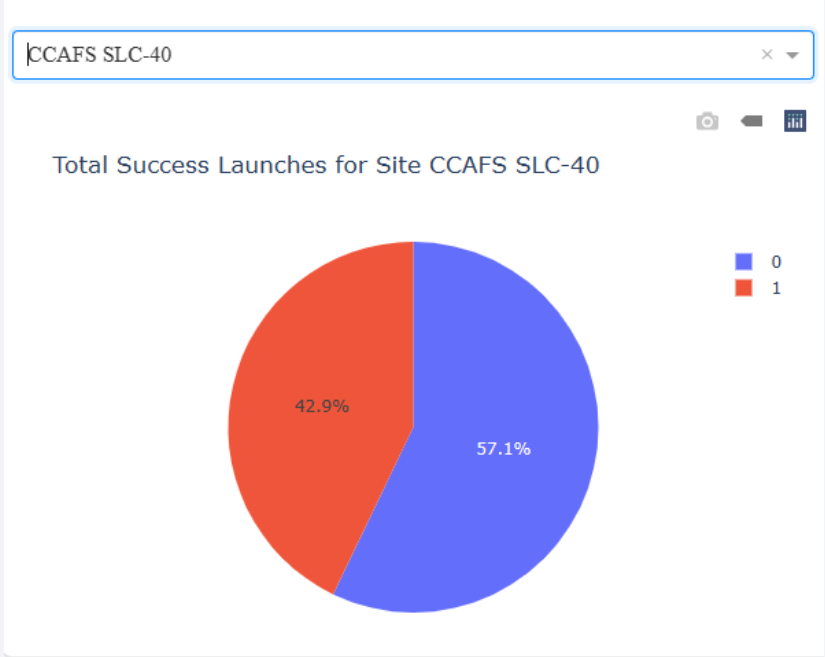
Distribution of Successful Launch

- The distribution of successful landings across all launch sites



Launch Success Rate at CCAFS SL-40

- 0 represents failed launches, and 1 represents successful launches in the provided data.
- Notably, 57.1% of launches at CCAFS SL-40 are categorized as failed launches (0)



GitHub

Build a Dashboard with Plotly Dash

Class Representation

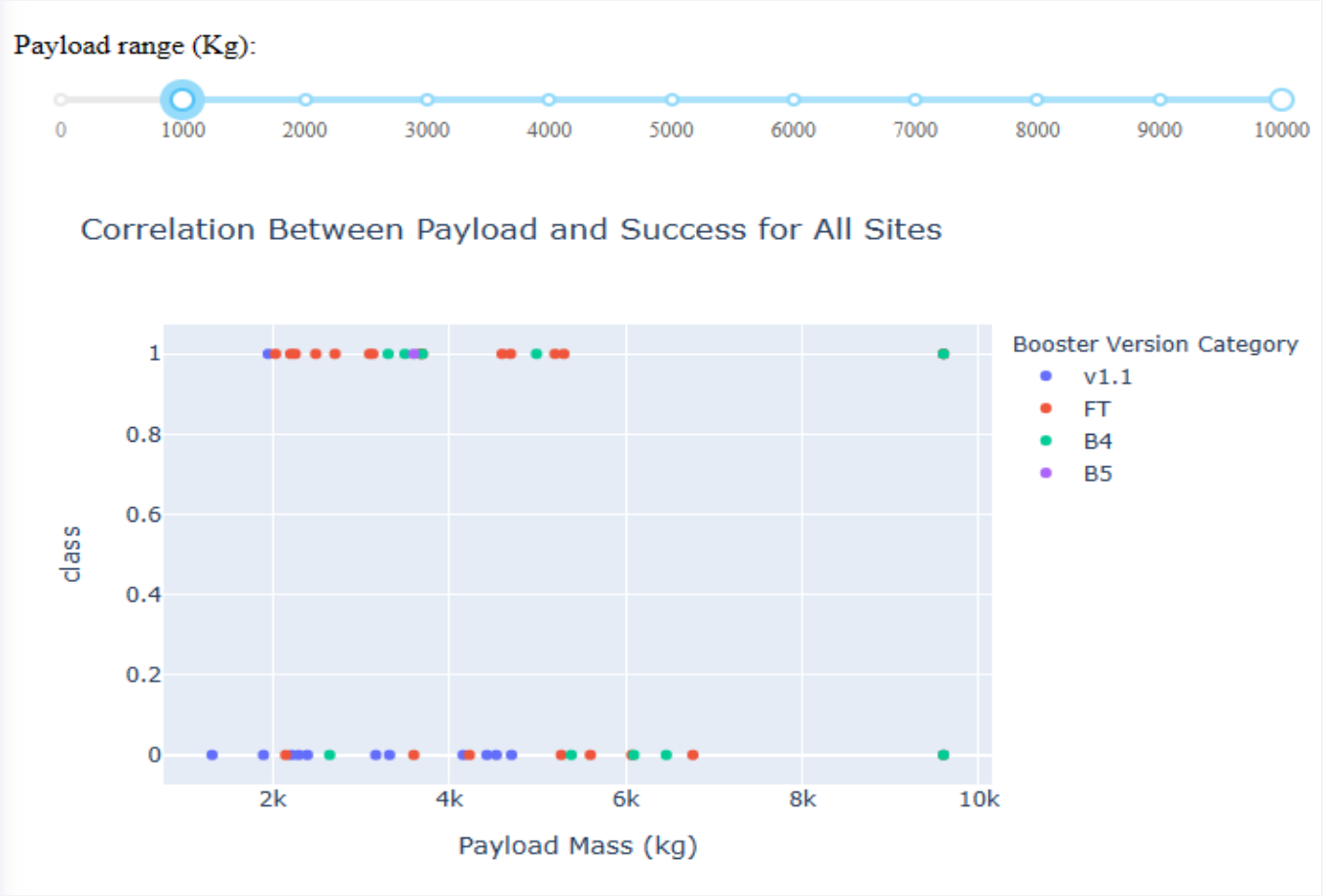
- In the Scatter plot, the "Class" attribute is employed, with 1 denoting a successful landing and 0 indicating a failure.

Booster Version Categorization

- The Scatter plot color-codes data points based on the booster version category.

Launch Count Representation

- The size of each point in the Scatter corresponds to the number of launches associated with the particular data point.



Section 5

Predictive Analysis (Classification)

Predictive Analysis (Classification)

Model Performances

- The models exhibited similar accuracy on the test set, with an overall accuracy of 83.33%. The Decision Tree Classifier had a slightly lower accuracy of 77.77%

Small Sample Size

- It's important to note that the sample size is small, consisting of only 18 data points. This limited sample size can result in significant variance in accuracy results, as observed in the Decision Tree Classifier across repeated runs

Need for More Data

- Due to the small sample size, it challenging to draw definitive conclusions about the best-performing model. More data is likely needed to make a more robust determination.

Individual Model Accuracies

- Logistic Regression Accuracy: 83.33%
- Support Vector Machine Accuracy: 83.33%
- Decision Tree Accuracy: 77.78%
- K-Nearest Neighbors Accuracy: 83.33%

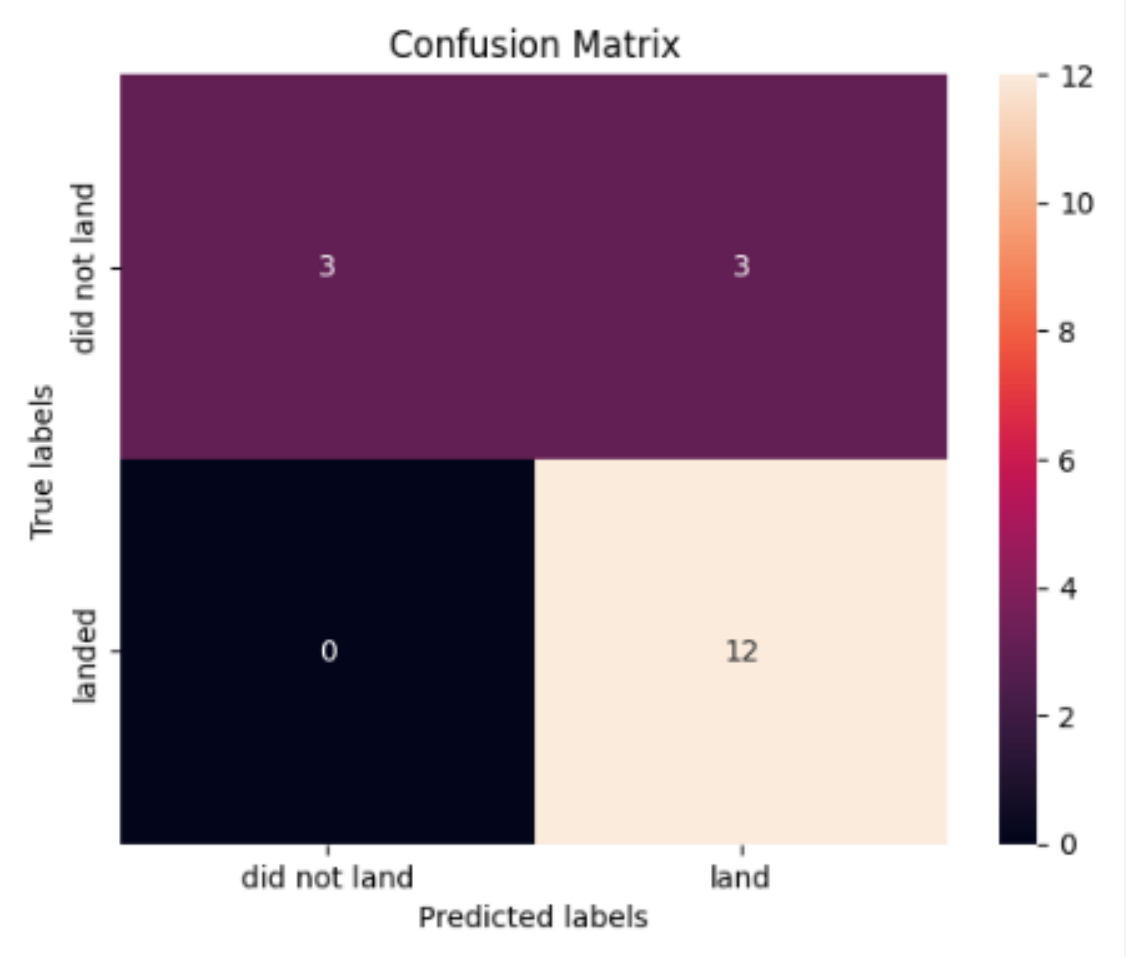
Model	Accuracy
Logistic Regression	83.33%
Support Vector Machine	83.33%
Decision Tree	77.78%
K-Nearest Neighbors	83.33%

GitHub

Confusion Matrix

Logistic Regression, Support Vector Machine, K-Nearest Neighbors Confusion Matrix

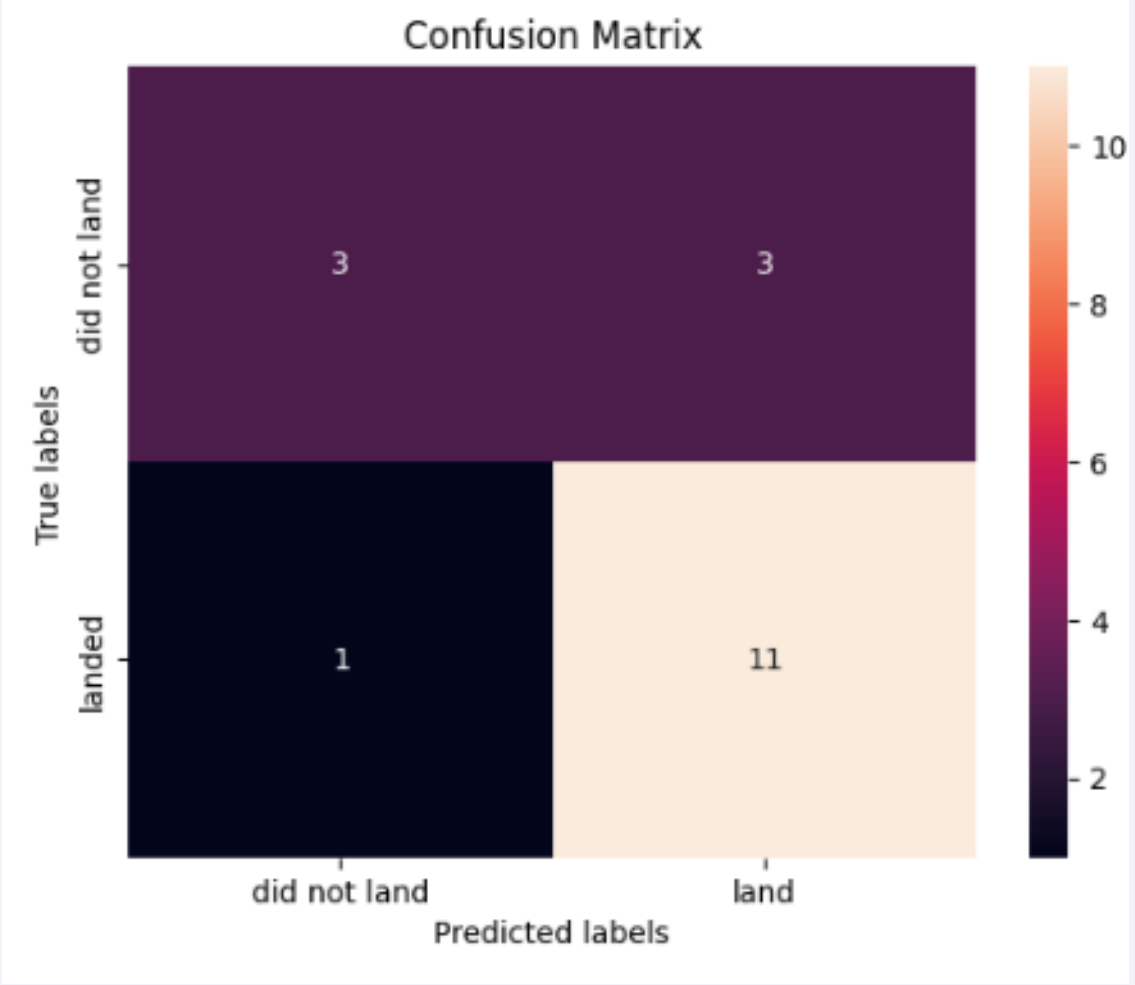
- All three models exhibit identical performance, correctly identifying 3 positive instances and 12 negative instances, with no false positives and 3 false negatives
- Consistency in confusion matrices suggests a robust and reliable predictive capability across Logistic Regression, SVM, and KNN models



Confusion Matrix

Decision Tree Model Evaluation - Confusion Matrix

- The Decision Tree model correctly identified 3 positive instances and 11 negative instances, with 1 false positive and 3 false negatives
- One instance that other models correctly identified as negative was misclassified as positive by the Decision Tree (False Positive)
- Consistency in correctly identifying positive instances (True Positives) with other models



Results

Data Collection:

- Utilized a combination of the SpaceX REST API and web scraping from SpaceX Wikipedia pages.
- Gathered information about launch details, rocket specifications, payload, and landing outcomes.

Data Preparation:

- Created data labels for successful and unsuccessful Stage 1 landings.
- Stored collected data in a DB2 SQL database for efficient retrieval.

Visualization:

- Developed dashboards for effective visualization of key information.
- Utilized visual elements to showcase launch details and outcomes.

Machine Learning Model:

- Built a machine learning model to predict Stage 1 landing success.
- Achieved an accuracy of 83% in model performance.

Use Case:

- The model empowers Space Y to predict, with high accuracy, whether a launch will have a successful Stage 1 landing before launch.
- Allon Musk and Space Y can make informed decisions regarding launch bids against competitors like SpaceX.

Recommendations:

- Collect more data to further refine and improve machine learning models.
- Explore additional features or optimizations to enhance model accuracy.

Conclusion:

The Space Y machine learning project successfully addressed the goal of predicting Stage 1 landing outcomes. The developed model provides a valuable tool for decision-making in the competitive space launch industry, enabling cost savings and strategic bidding against industry leaders like SpaceX. Ongoing data collection and model refinement will contribute to continuous improvement and increased accuracy in predicting launch outcomes

Thank you!