DATA SCIENCE

2025

Outline

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

Executive Summary

- Summary of methodologies

- Python for Data Science
- Wrangling & Analysis
- Data Visualization (Matplotlib, Seaborn, Folium)
- Machine Learning Essentials
- SQL & Databases
- Capstone Project – End-to-end data science problem-solving.

Introduction

- Project background and context The project aims to address data science
- , using data-driven methods to derive insights and solutions.Developed as part of the IBM Data Science Professional Certificate, this project is grounded in real-world data science
- Objectives

Objectives

•Analyze and visualize data to uncover hidden trends.

- Build predictive models to support business decision-making.
- Deliver actionable insights

Section 1

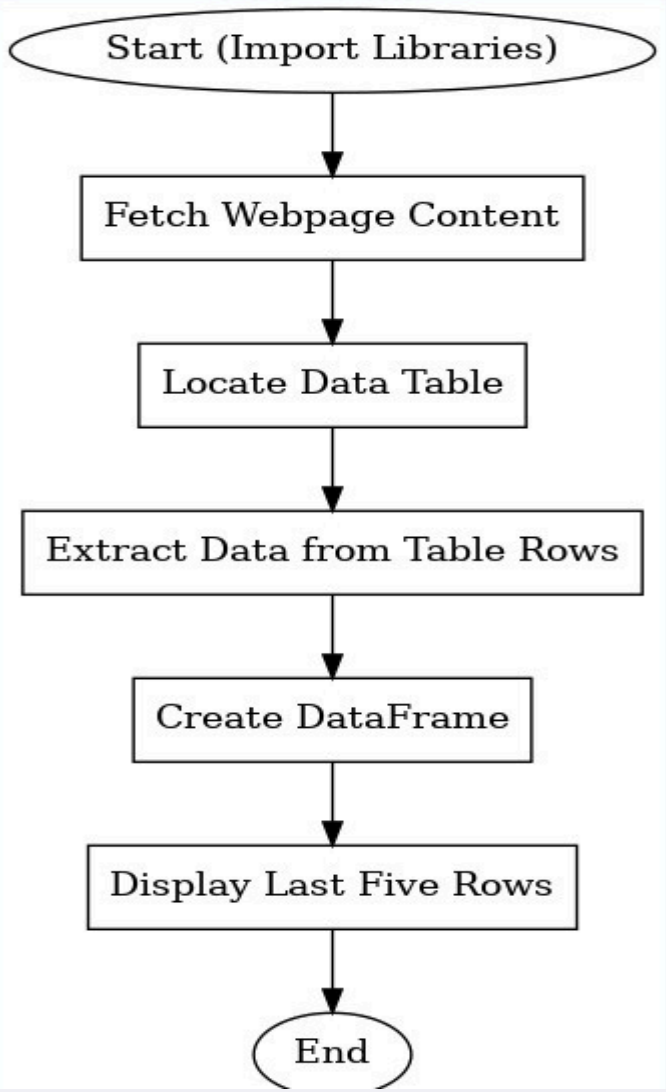
Methodology

Methodology

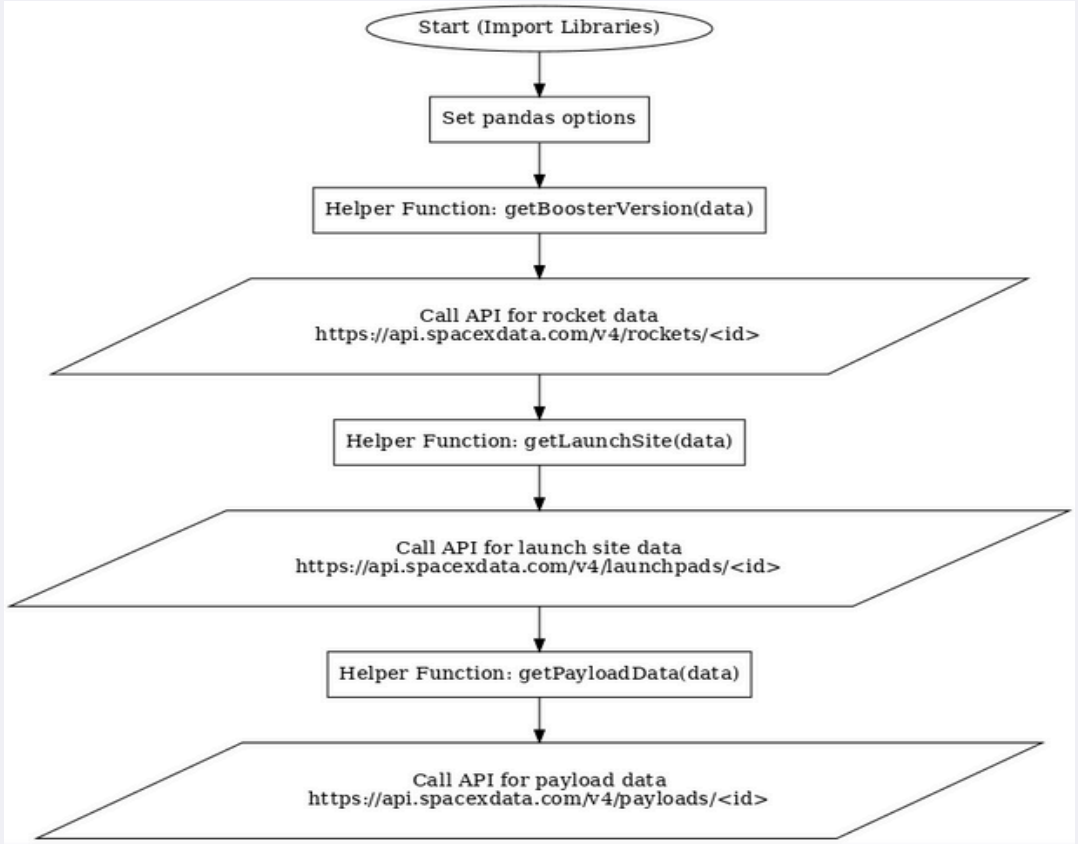
Executive Summary

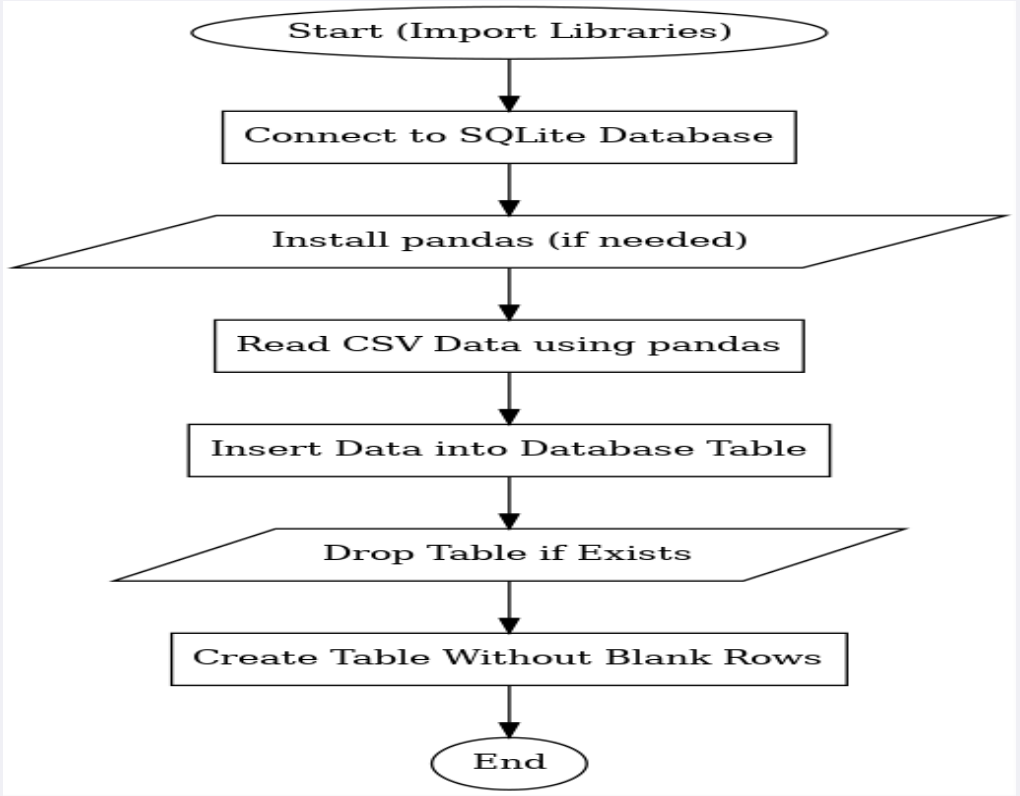
- Data collection methodology:
  - We use Wget function or Yfinance from yahoo for web scraping
- Perform datawrangling
  - We use pandas library to sort, arrange, clean and study data
- Perform exploratory data analysis (EDA) using visualization andSQL
- Perform interactive visual analytics using Folium andPlotlyDash
- Perform predictive analysis using classification models
  - We use Scikitlearn we train test split data and use hyperparameter optimization with gridsearchCV

Data Collection



Data Collection – SpaceX API





Data Collection - Scraping

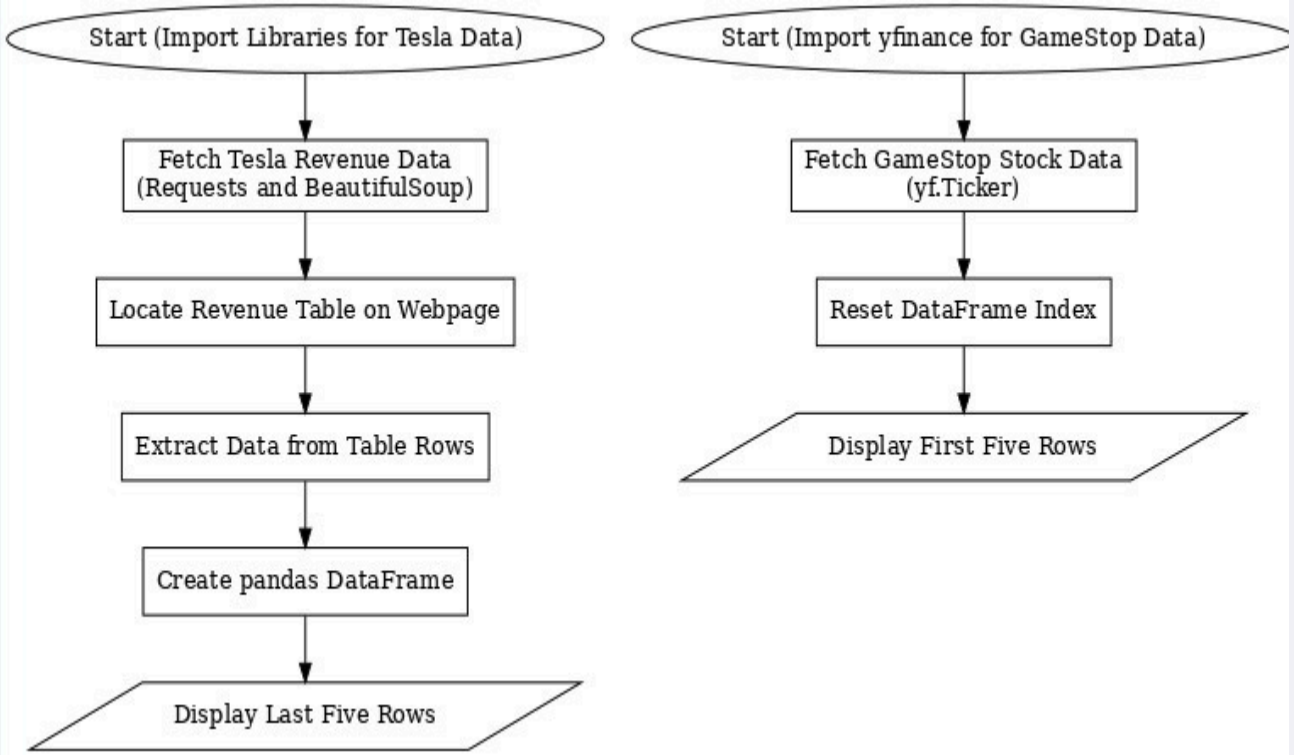
Start (Import Libraries)
requests, BeautifulSoup, pandas.

FetchTesla RevenueData
requests.get(), URL, headers, BeautifulSoup.

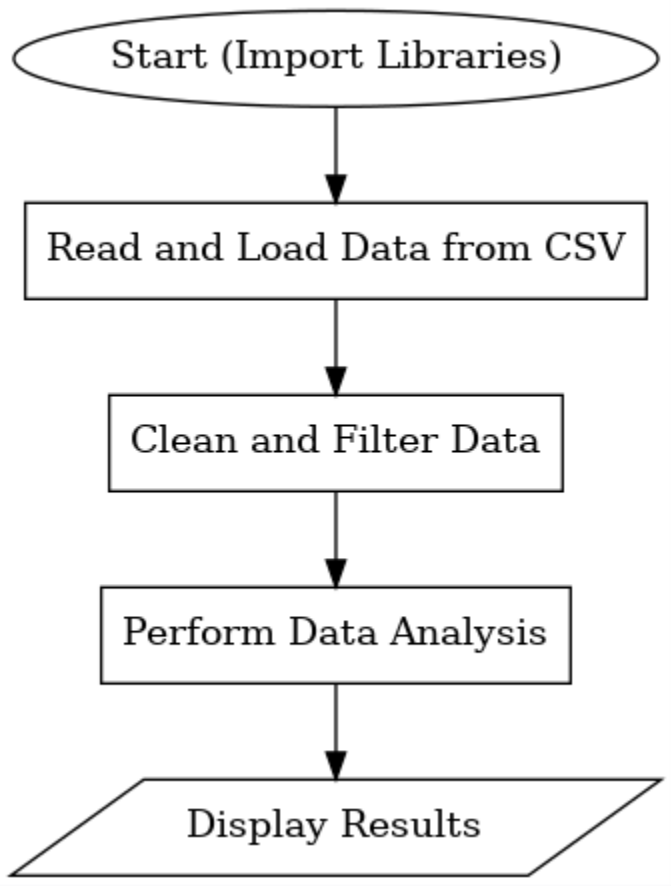
Locate RevenueTable:
soup.find_all(), table, historical_data_table.

ExtractData fromTable Rows: forloop, find_all('td'), text.strip(), append.

Createpandas DataFrame: pd.DataFrame(), data.



Data Wrangling



EDA with Data Visualization

Visualizing the Relationship between Flight Number and Launch Site Plot Type: Seaborn categorical plot (sns.catplot).Reason: This plot allows a clear comparison of how flights are distributed across various launch sites. Using hue="Class" distinguishes between successful and unsuccessful launches, providing additional insight into the distribution of outcomes by site and flight number Visualizing Success Rate by Orbit TypePlot Type: Bar plot (kind='bar').Reason: Aggregating the success rate for each orbit type simplifies the comparison, allowing the viewer to quickly see which orbits have the highest or lowest success rates. A bar plot effectively communicates grouped summary statistics. Visualizing Launch Success Yearly TrendPlot Type: Line plot (kind='line').Reason: Line plots are ideal for time series data. This plot shows the trend of launch success rates over years, making it easy to spot improvements or regressions over time.

EDA withSQL

- Create a Clean Table:

CREATE TABLE SPACEXTABLE AS SELECT * FROM SPACEXTBL WHERE Date IS NOT NULL

- Unique Launch Sites

SELECT DISTINCT Launch_Site FROM SPACEXTABLE

- Filtered Launch site Records

SELECT * FROM SPACEXTABLE WHERE Launch_Site LIKE 'CCA%' LIMIT 5

- Total Payload by NASA CRS Missions

SELECT SUM(PAYLOAD_MASS__KG_) as total_payload FROM SPACEXTABLE WHERE Customer LIKE '%NASA (CRS)%';.

- Average Payload for Booster Version F9 v1.1

SELECT AVG(PAYLOAD_MASS__KG_) as avg_payload FROM SPACEXTABLE WHERE Booster_Version = 'F9 v1.1';

Build an Interactive Map with Folium

- • Circle Marker ：
- A circlewasaddedtomarkthelocationofNASA Johnson SpaceCenter.
- Features:
  - Radius: 1000 meters
  - Popuplabel: "NASA Johnson SpaceCenter"
- •Text LabelMarker ：
- A Markerobjectwasaddedwitha DivIcondisplayingthetext"NASA JSC."
- Purpose: IdentifytheJohnson SpaceCenter witha visible label.
- • Launch Outcome Markers ：
- Markerswereaddedforeachlaunchrecordin thedataset.
- Features:
  - Color-codedbasedonsuccess(green) orfailure(red) ofthelaunch.
  - ManagedusingtheMarkerClusterplugin toreduce mapclutterforoverlappingpoints.
- •Mouse Position Plugin ：
- Addedtothemaptodisplaycoordinatesofpointsunderthemouse cursor.
- Purpose: Tohelpidentifylocations(e.g., railways, highways, coastline) nearthelaunchsites forfurtherproximityanalysis.
- • Distance Markers ：
- Markersworecreatedtoindicatedistancesfromlaunchsites topointsofinterest(e.g., closestcoastline, city, orhighway).
- Features:
  - CustomHTML `DivIcon` used to display the distance.

Build a Dashboard with Plotly Dash

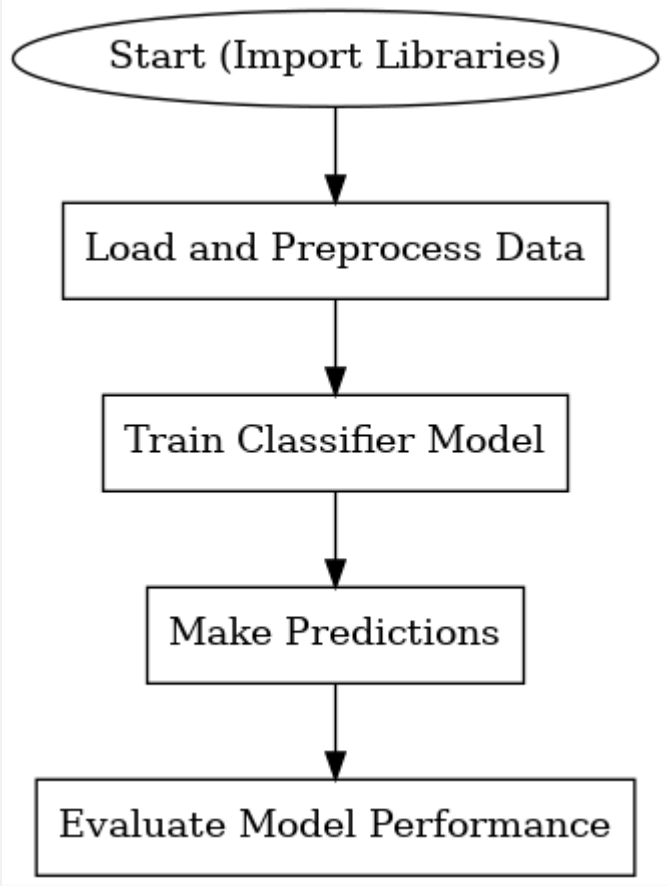
1. Dropdown Menu (Launch Site Selection)
   - Purpose: Allows the user to select a specific launch site or view data for all sites.
   - Interaction: Triggers updates in both the pie chart and scatter plot.

2. Pie Chart (Launch Success Overview)
   - Purpose: Displays total successful launches for all sites or success vs. failure for a selected site.
   - Data:
     - When "All Sites" is selected, it shows success counts per site.
     - When a specific site is selected, it shows a breakdown of success vs. failure launches.
   - Dynamic Interaction: Updates based on the selected launch site from the dropdown.

3. Payload Range Slider
   - Purpose: Enables users to filter data based on the payload mass (in kilograms).
   - Interaction: The slider adjusts the payload range for the scatter plot.

4. Scatter Plot (Payload vs. Launch Success)
   - Purpose: Visualizes the correlation between payload mass and launch success rate.
   - Data:
     - X-axis: Payload Mass (kg)
     - Y-axis: Launch outcome (0 = Failure, 1 = Success)
     - Color: Booster version category
   - Dynamic Interaction: Updates based on both the selected launch site and payload range

Predictive Analysis (Classification)



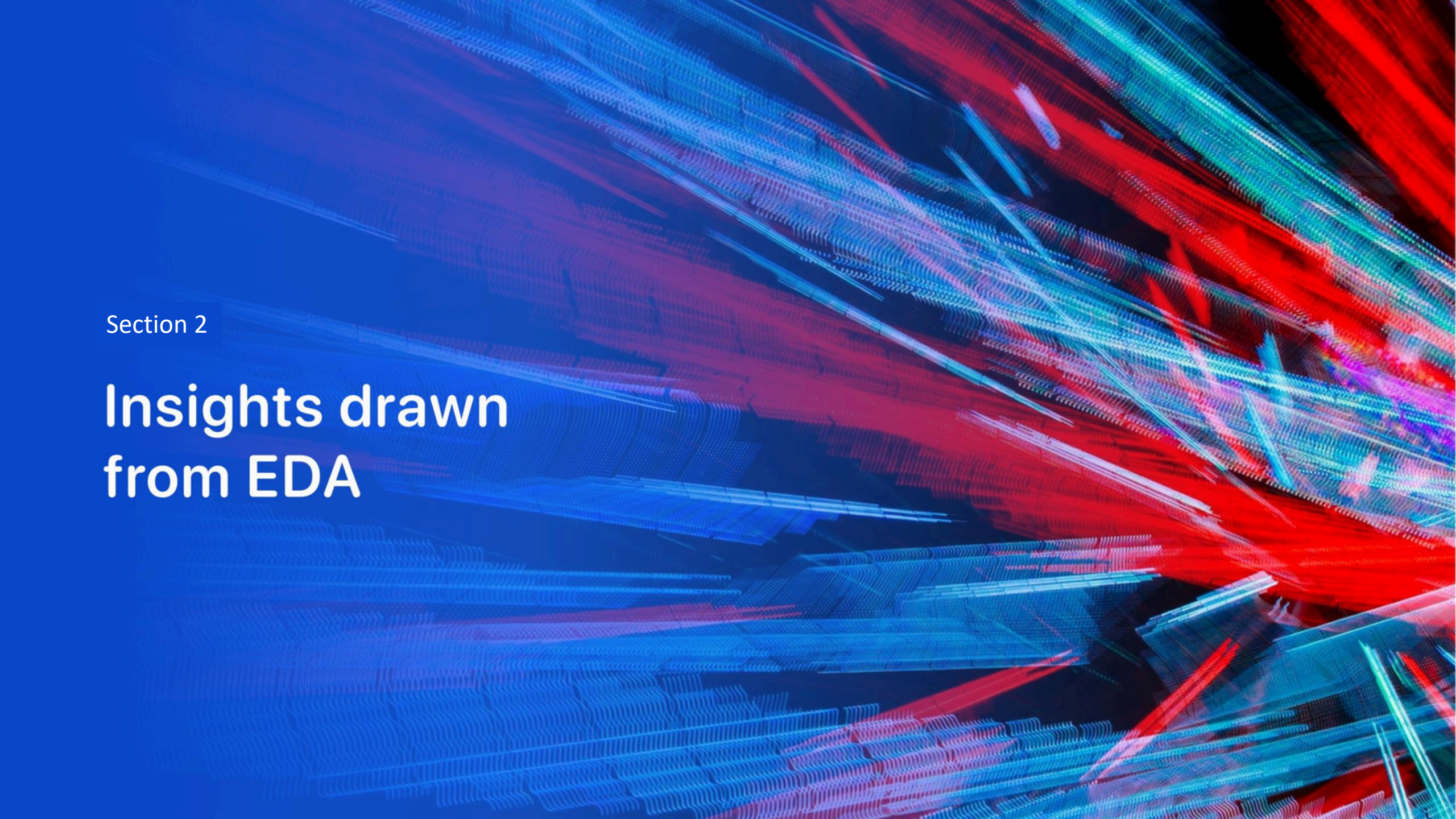
Results

- Data Loading and Preparation: Data was read and cleaned to remove missing values or inconsistencies.Model Training:

- Classifiers were trained on the processed dataset to predict weather patterns.

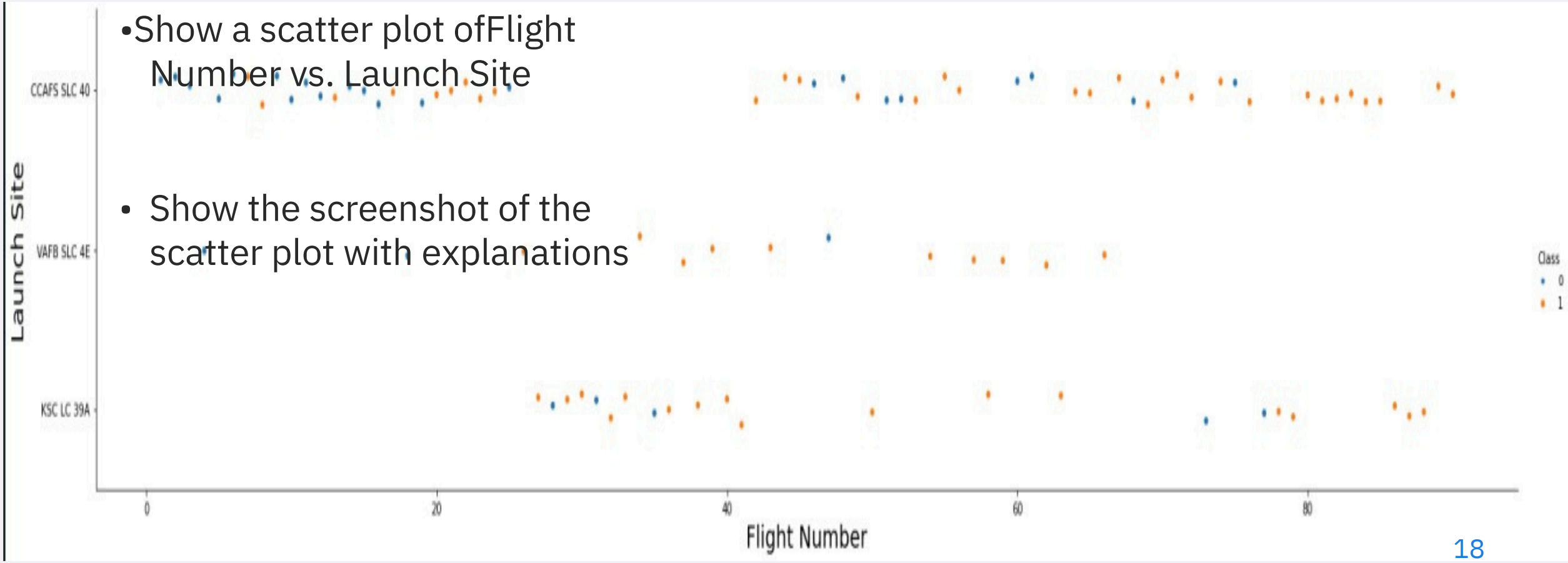
- Evaluation Metrics: The model's accuracy, precision, and recall were calculated to assess performance.

- Insights and Visualization: Charts or tables likely summarized key weather-related insights, such as temperature trends or humidity effects on outcomes.

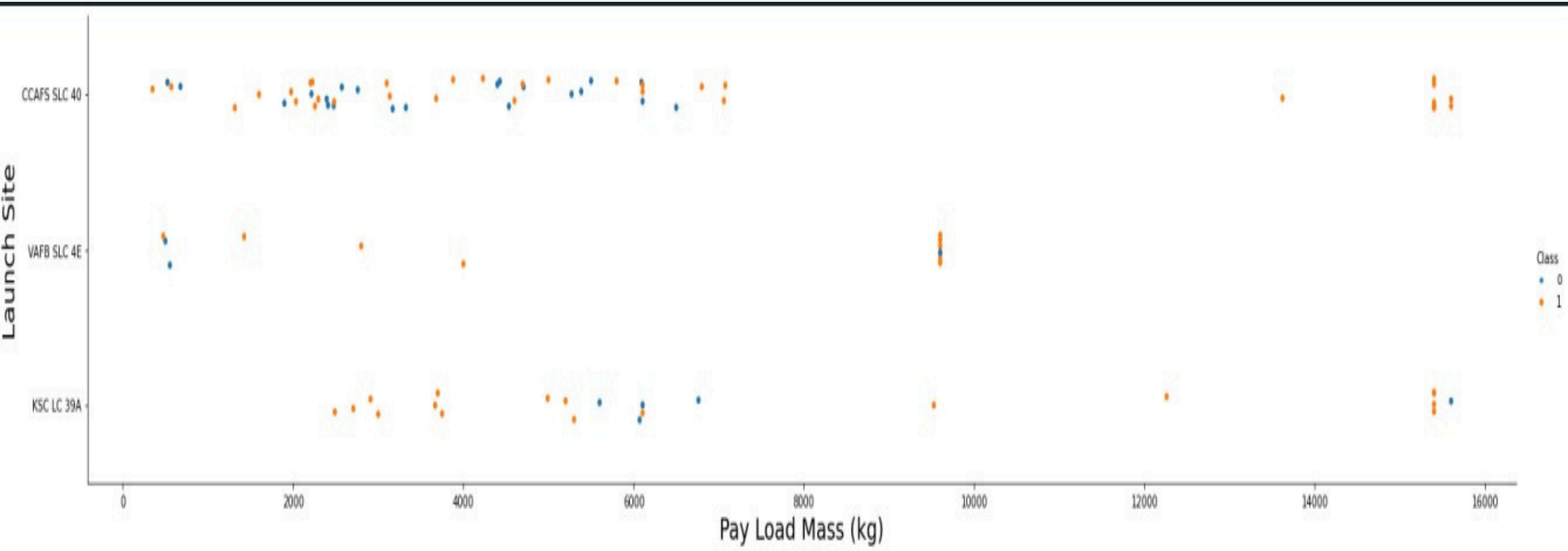

Section 2
Insights drawn
from EDA

Flight Number vs. LaunchSite

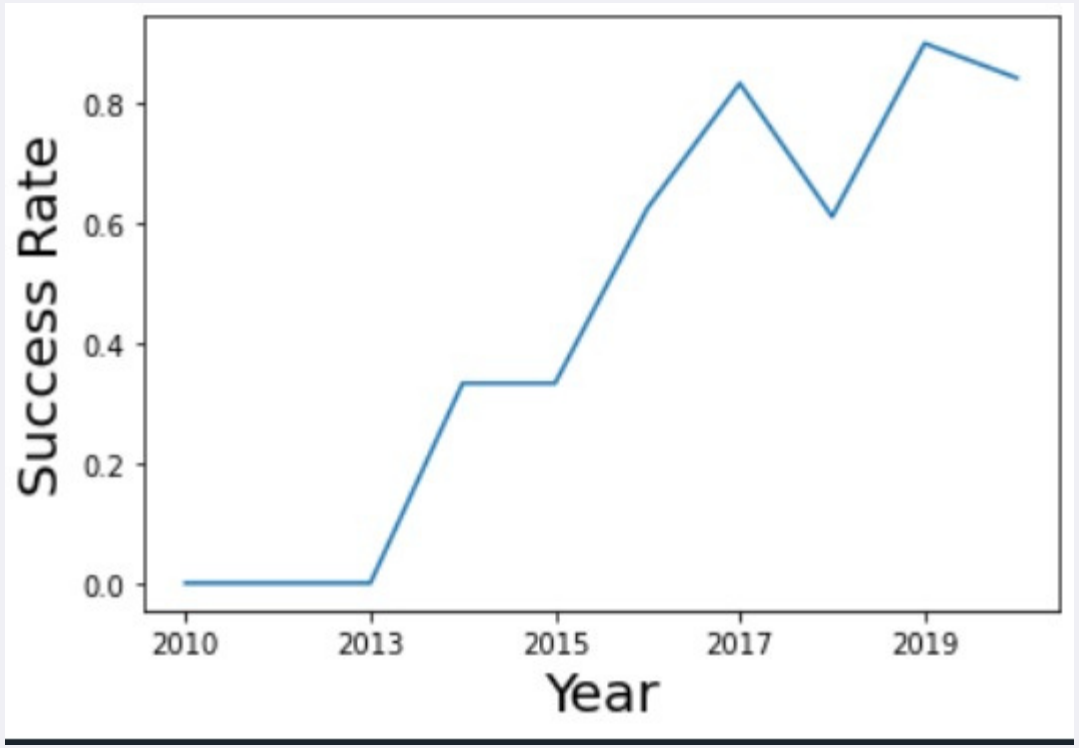
- Show a scatter plot ofFlight Number vs. Launch Site
- Show the screenshot of the scatter plot with explanations

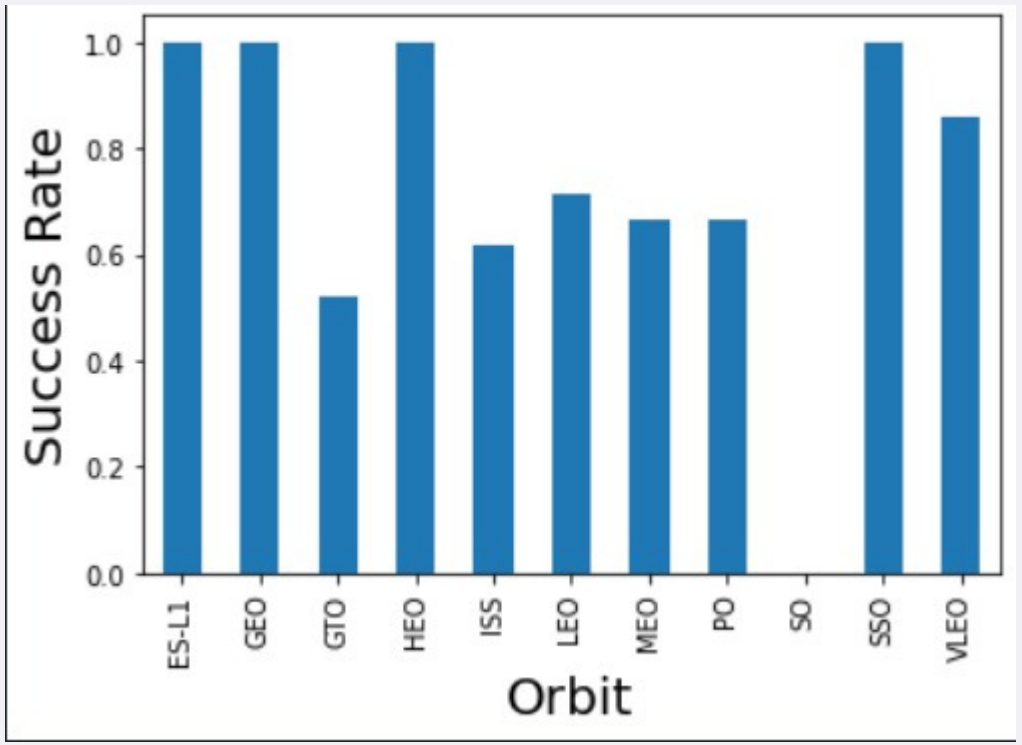


Payload vs. Launch Site



Success Rate vs. Orbit Type

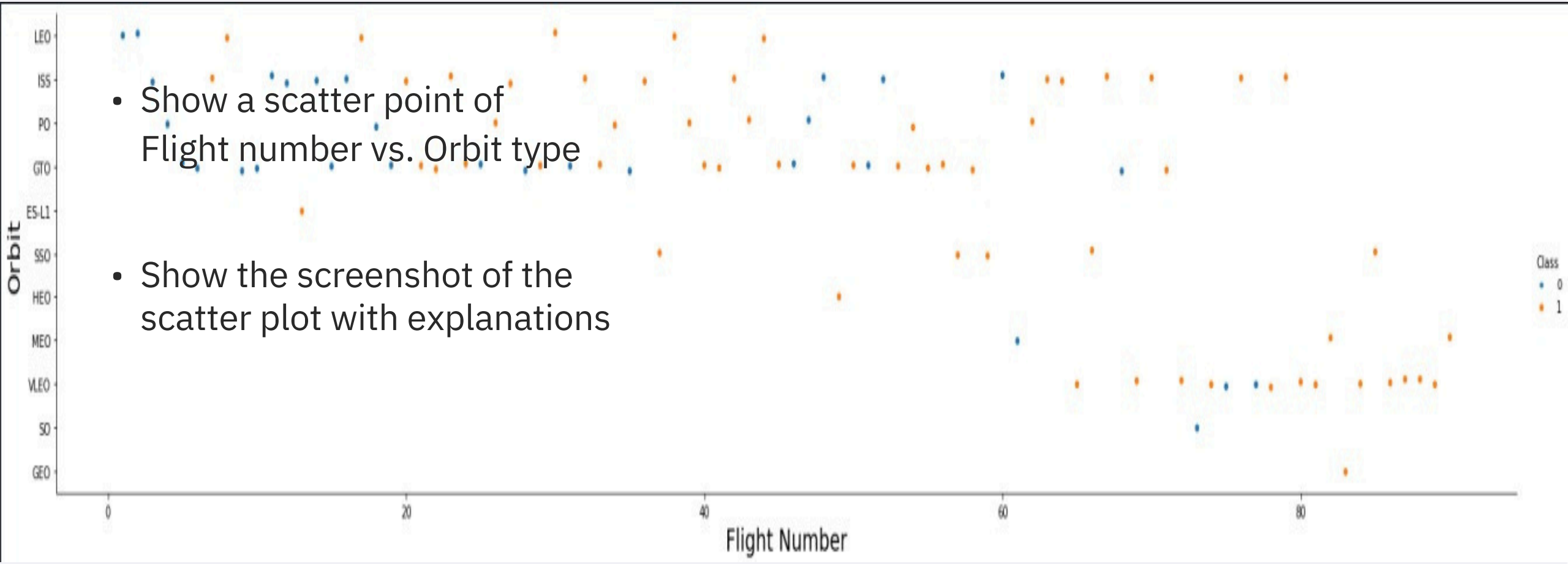




- There was learning about orbits
- Show the screenshot of the scatter plot with explanations

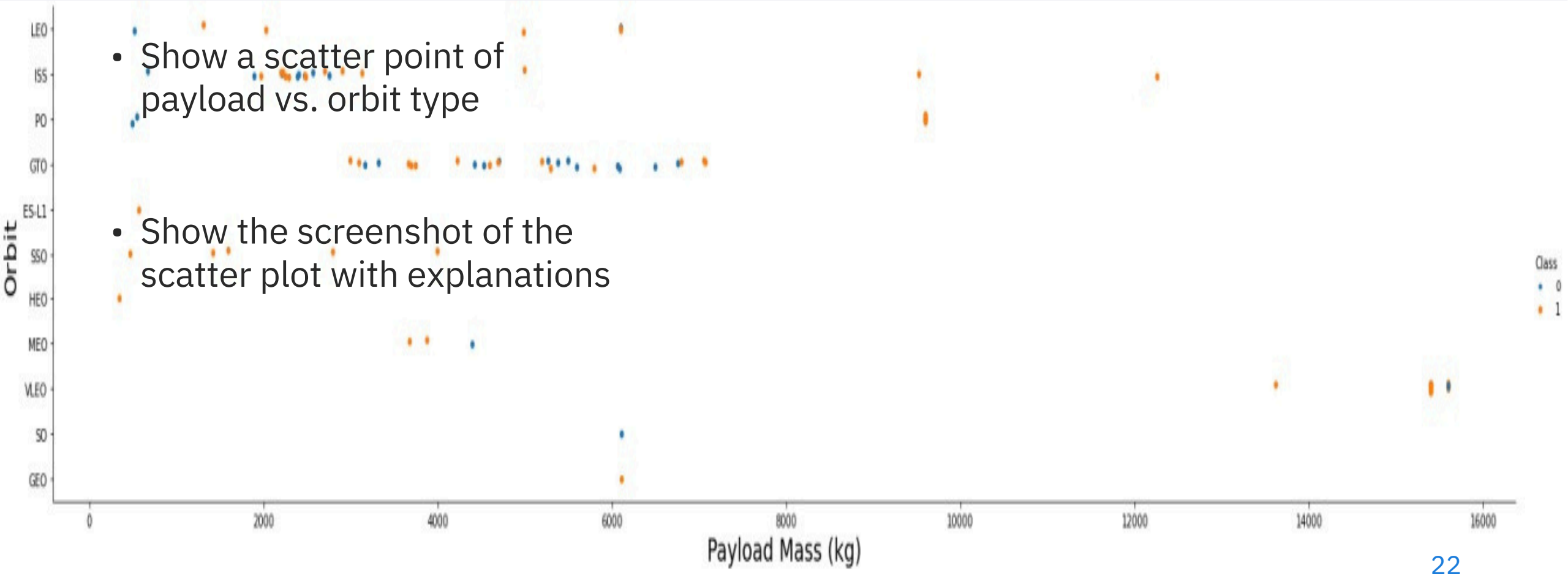
Flight Number vs. Orbit Type

- Show a scatter point of Flight number vs. Orbit type
- Show the screenshot of the scatter plot with explanations

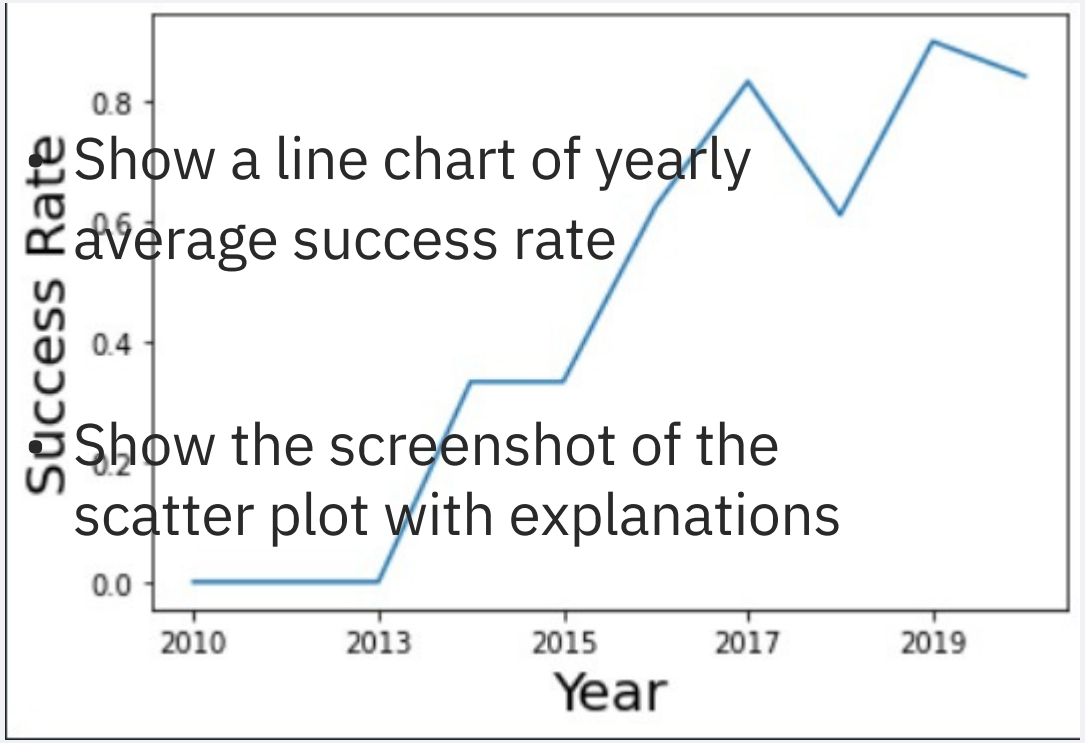


Payload vs. OrbitType

- Show a scatter point of payload vs. orbit type
- Show the screenshot of the scatter plot with explanations



Launch Success Yearly Trend



- Show a line chart of yearly average success rate
- Show the screenshot of the scatter plot with explanations

All Launch Site Names

- Find the names of the unique launch sites
- SELECT DISTINCT Launch_Site FROM SPACEXTABLE;
- Launch Site
- CCAFS LC-401
- VAFB SLC-4E2
- KSC LC-39A3
- CCAFS SLC-40

Launch Site Names Begin with 'CCA'

- Find 5 records where launch sites begin with `CCA`

•query_cca_records = "SELECT * FROM SPACEXTABLE WHERE Launch_Site LIKE 'CCA%' LIMIT 5"cursor.execute(query_cca_records)cca_records = cursor.fetchall()

	Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
1	2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
2	2010-12-08	15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight C1, two CubeSa...	0	LEO (ISS)	NASA (COTS) NRO	Success	Failure (parachute)
3	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
4	2012-10-08	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
5	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 mass carried by all boosters
- total_payload_mass = df['PAYLOAD_MASS__KG_'].sum()

- 619967 Kg

Average Payload Mass by F9 v1.1

- Calculate the average payload mass carried by booster version F9 v1.1

- average_payload_f9_v1_1 = df_new[df_new['Booster_Version'] == 'F9 v1.1']['PAYLOAD_MASS__KG_'].mean()

- average_payload_f9_v1_1 = 2928.4 kg

First Successful Ground Landing Date

- Find the dates of the first successful landing outcome on ground pad
- first_success_ground_pad_date = df_new[df_new['Landing_Outcome'] == 'Success (ground pad)']['Date'].min()

- first_success_ground_pad_date = '2015-12-22'

Successful Drone Ship Landing with Payload between 4000 and 6000

- List the names of boosters which have successfully landed on drone ship and had payload mass greater than 4000 but less than 6000

- boosters_success_drone_ship = df_new[(df_new['Landing_Outcome'] == 'Success (drone ship)') & (df_new['PAYLOAD_MASS__KG_'] > 4000) &(df_new['PAYLOAD_MASS__KG_'] < 6000)]['Booster_Version'].unique()

- The boosters that have successfully landed on a drone ship and carried a payload mass between 4000 and 6000 kg are:

1.F9 FT B1022

2.F9 FT B1026

3.F9 FT B1021.2

4.F9 FT B1031.2

Total Number of Successful and FailureMission Outcomes

•Calculate the total number of successful and failure mission outcomes
•mission_outcomes_counts = df_new['Mission_Outcome'].value_counts()

- mission_outcomes_counts
- Success 98
- Failure (in flight) 1
- Success (payload status unclear) 1
- Success 1

Boosters Carried MaximumPayload

- List the names of the booster which have carried the maximum payload mass

```
max_payload_mass = df_new['PAYLOAD_MASS__KG_'].max()

# Filter the data to find boosters that carried the maximum payload mass
```

- boosters_with_max_payload = df_new[df_new['PAYLOAD_MASS__KG_'] == max_payload_mass]['Booster_Version'].unique()

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

2015 Launch Records

- List the failedlanding_outcomes in drone ship, their booster versions, andlaunch site names for in year 2015

- # Filter the data for failed drone ship landings in the year 2015
- failed_drone_ship_2015 = df_new[
- (df_new['Landing_Outcome'].str.contains('Failure', na=False)) &
- (df_new['Landing_Outcome'].str.contains('ship', na=False)) &
- (df_new['Date'].str.startswith('2015'))
-][['Landing_Outcome', 'Booster_Version', 'Launch_Site']]

- 13 Failure (drone ship)
- F9 v1.1 B1012 CCAFS LC-40
- F9 v1.1 B1015 CCAFS LC-40

Rank 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
- ranked_landing_outcomes = df_new[
- (df_new['Date'] >= '2010-06-04') & (df_new['Date'] <= '2017-03-20')
-]['Landing_Outcome'].value_counts().reset_index()
- # Rename columns for clarity
- ranked_landing_outcomes.columns = ['Landing_Outcome', 'Count']
- # Display the results sorted in descending order
- ranked_landing_outcomes

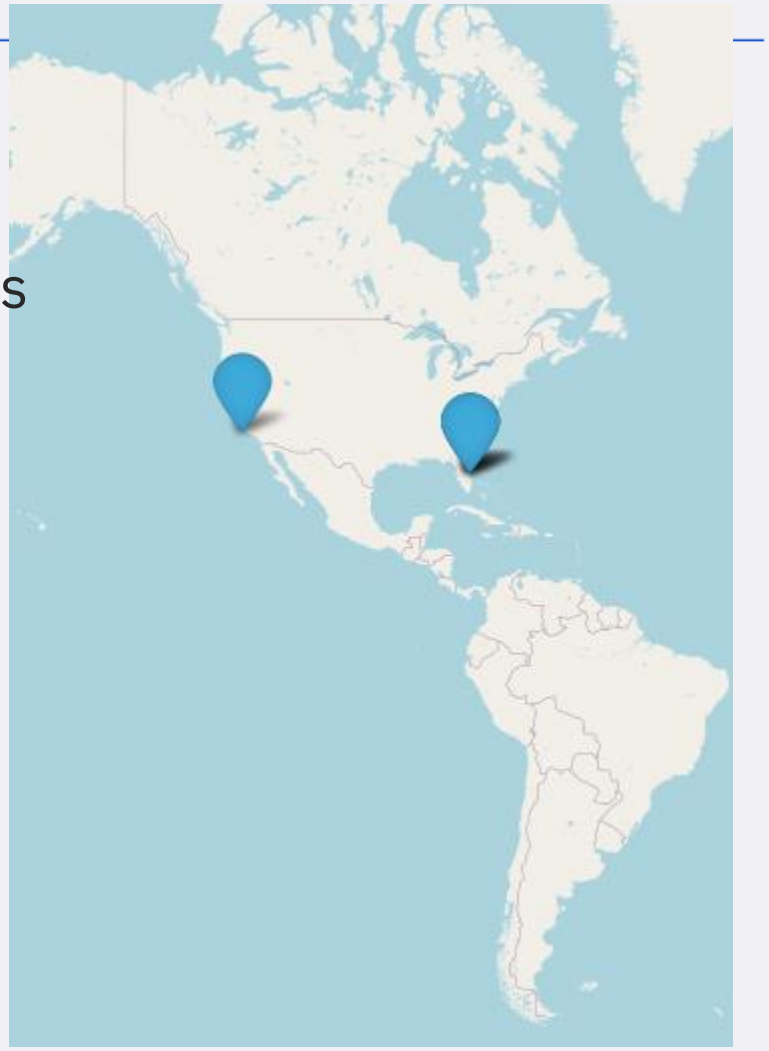
Section 3

Launch Sites Proximities Analysis

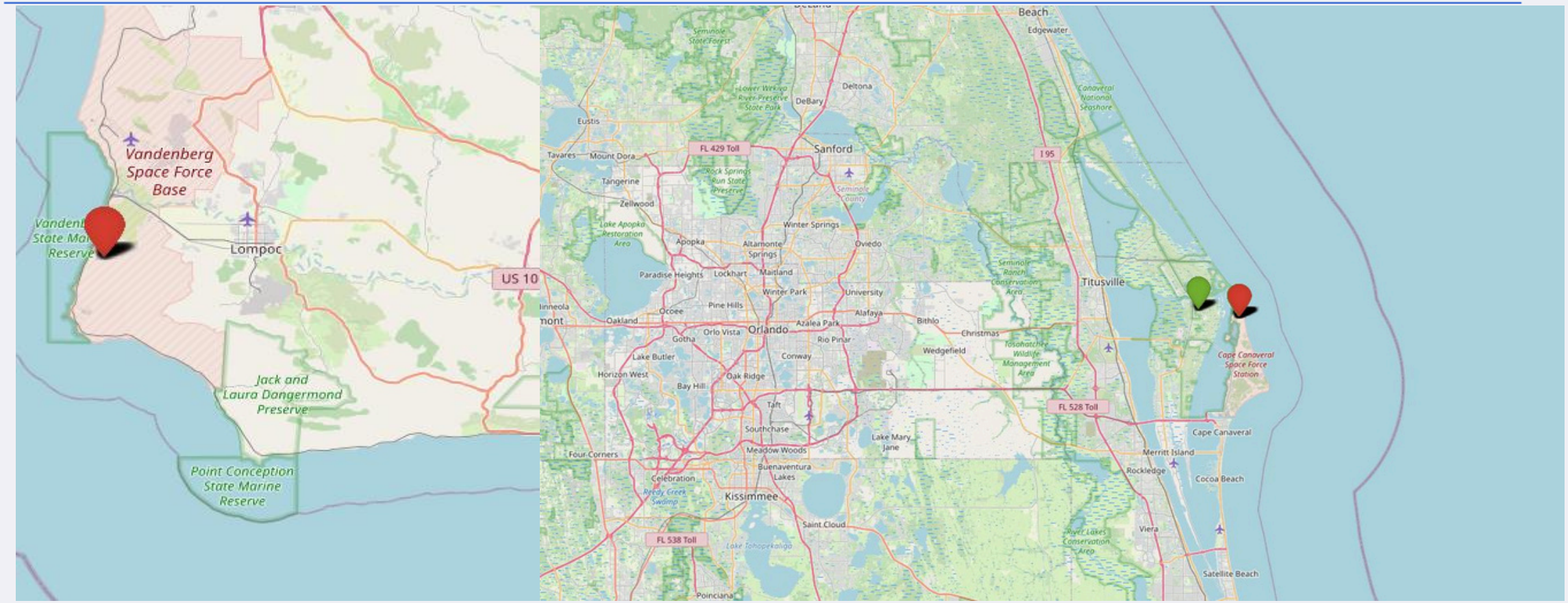
Launch sites map

- Explore the generated folium map and make a proper screenshot to include all launch sites’ location markers on a global map

- Each marker on the map represents a SpaceX launch site.The icon used is a blue rocket, symbolizing a launch location.

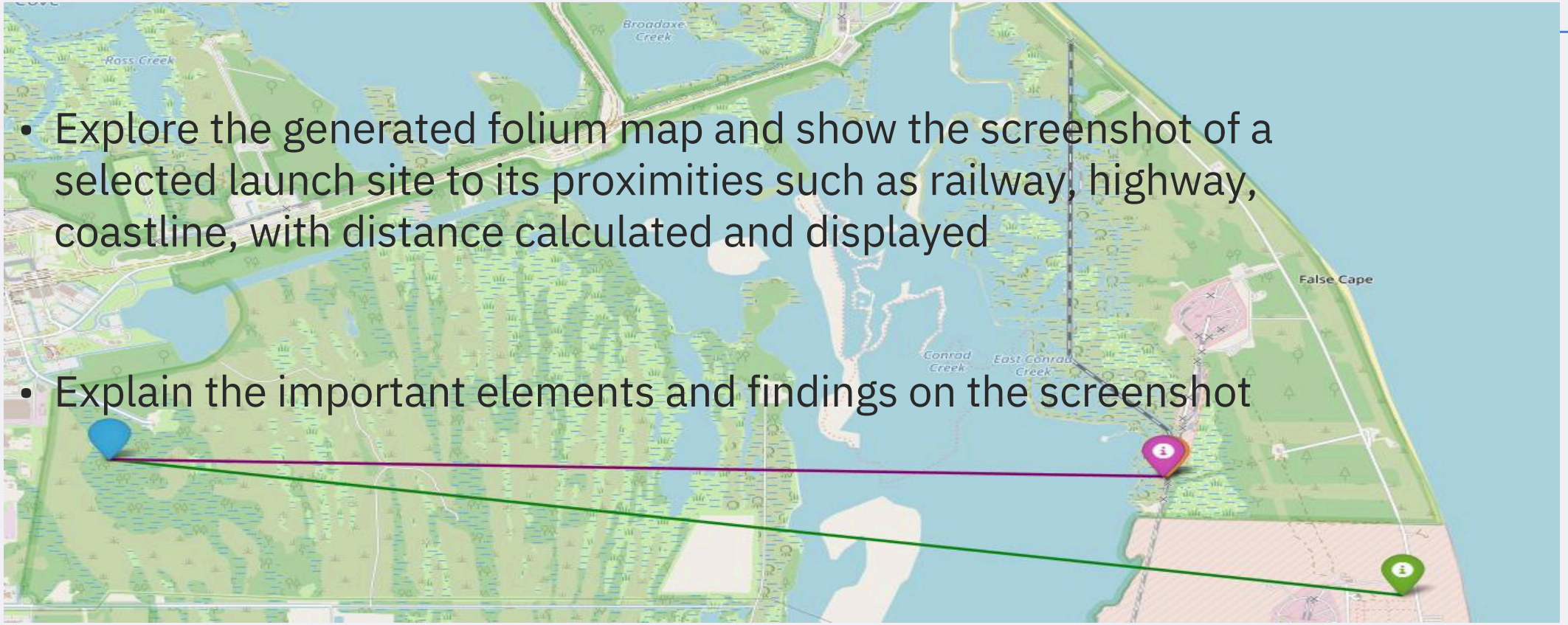


Launch Sites 2



Each launch site marker is color-coded to indicate the outcome of the launch:Green Marker: Represents a successful launch.
Red Marker: Represents a failed launch.

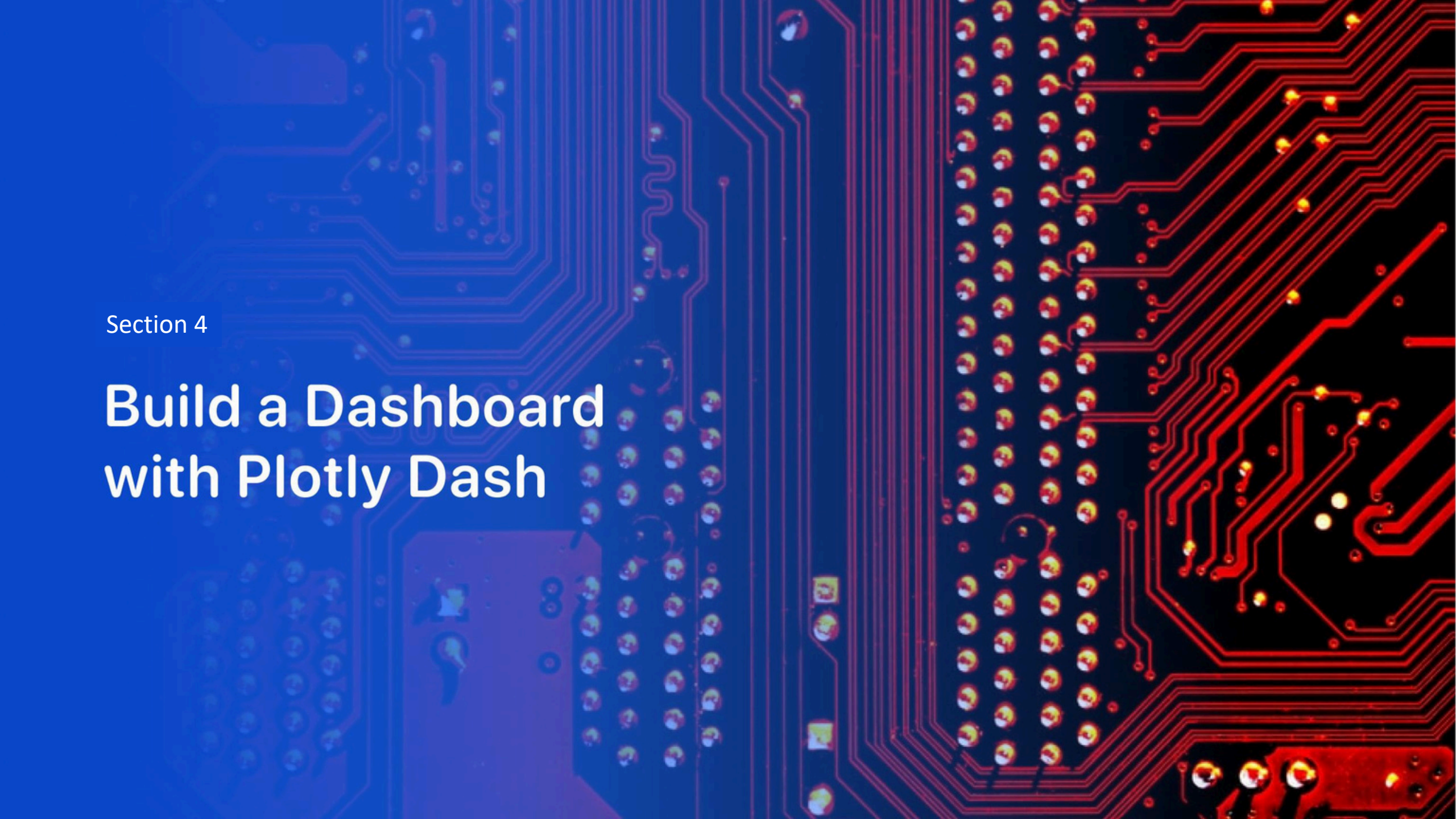
Launch sites Risks



- Explore the generated folium map and show the screenshot of a selected launch site to its proximities such as railway, highway, coastline, with distance calculated and displayed

- Explain the important elements and findings on the screenshot

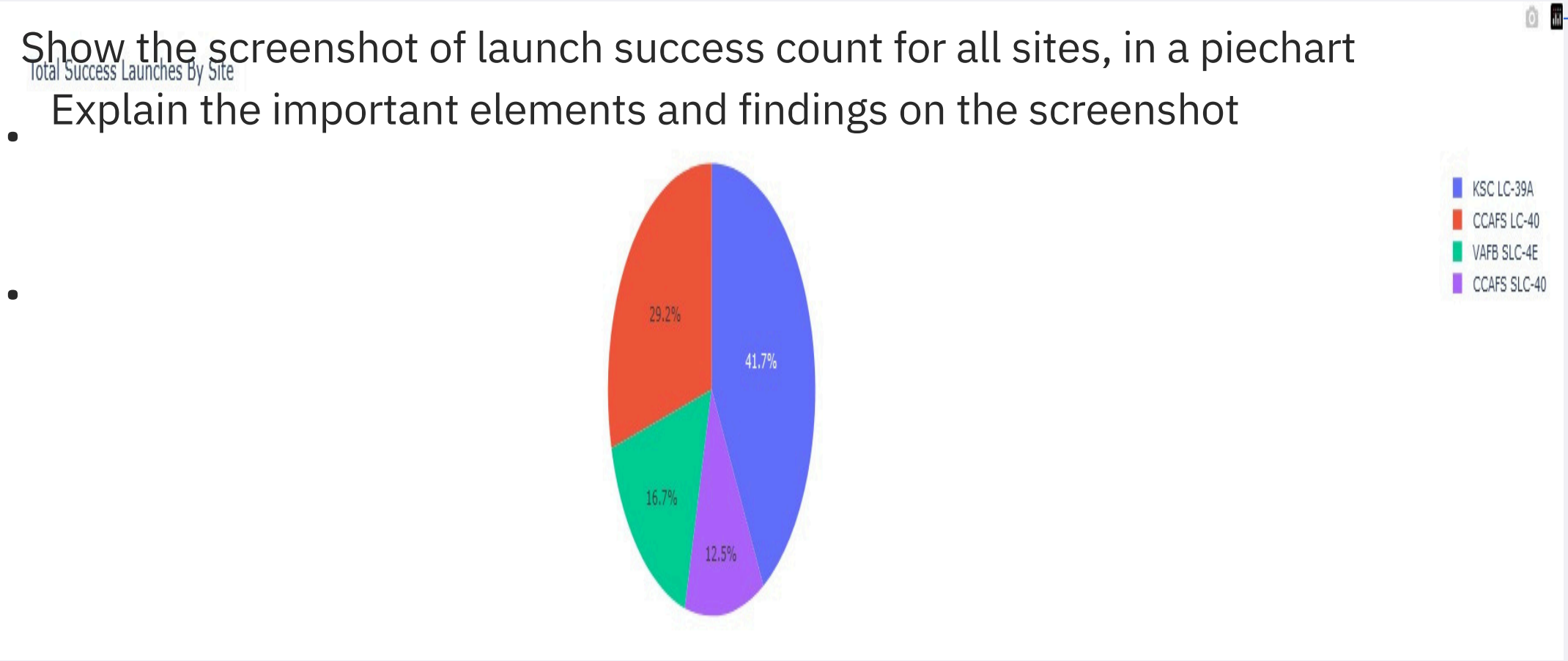
The blue marker with the rocket icon indicates the exact location of the Kennedy Space Center (KSC LC-39A) launch site.Clicking on the marker shows a popup with the name of the launch site.Proximity Markers:Green Marker (Coastline):Represents the closest point on the coastline.Distance from the launch site is displayed in kilometers.Orange Marker (Highway):Represents the nearest highway to the launch site.Distance from the launch site is shown on the marker


Section 4
Build a Dashboard with Plotly Dash

Space X dashboard

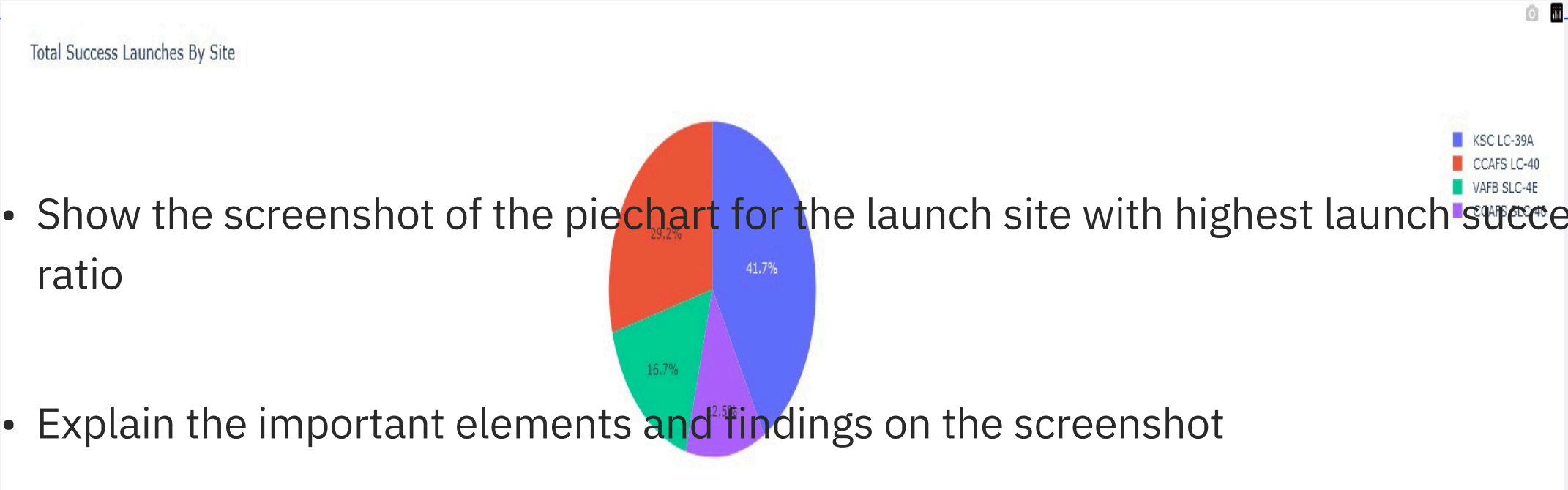
Show the screenshot of launch success count for all sites, in a piechart

- Explain the important elements and findings on the screenshot



KSC LC-39 A mostsuccessful

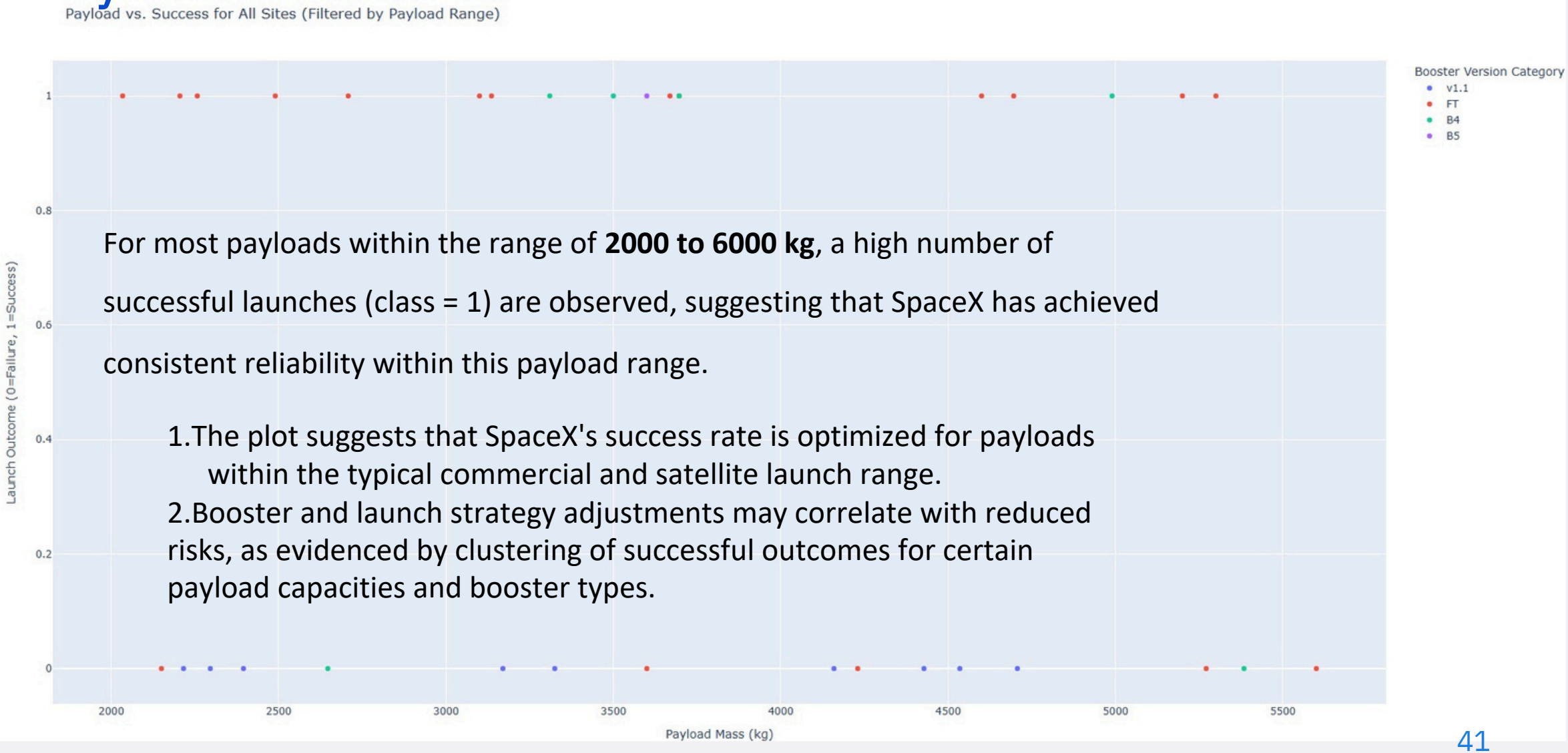
Space X Dashboard 2



- Show the screenshot of the piechart for the launch site with highest launch success ratio
- Explain the important elements and findings on the screenshot

KSC LC-39 A mostsuccessful

Payload vs Success all sites



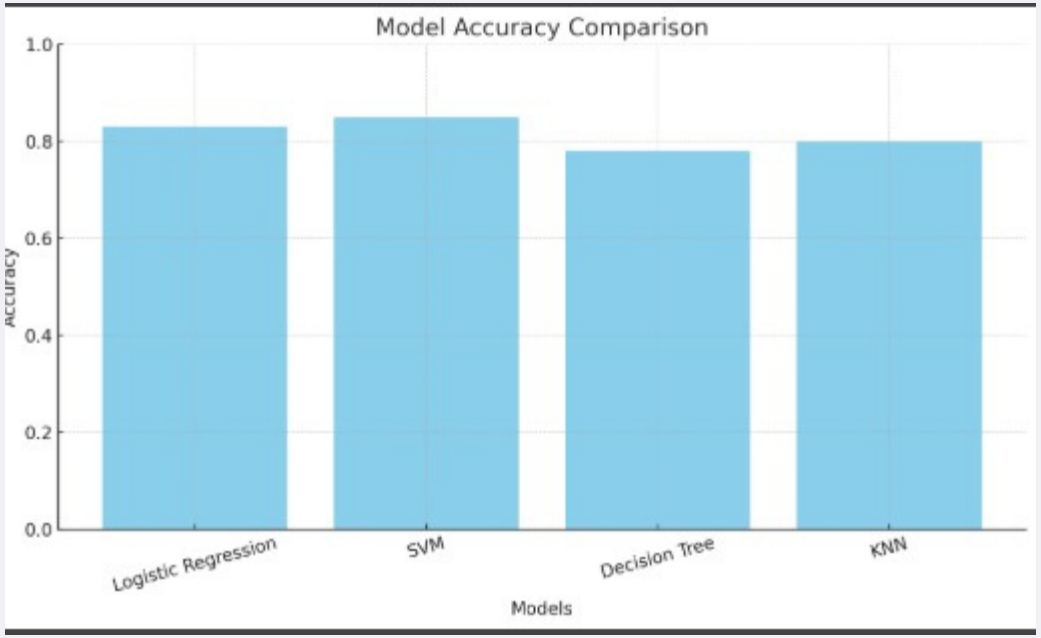
For most payloads within the range of **2000 to 6000 kg**, a high number of successful launches (class = 1) are observed, suggesting that SpaceX has achieved consistent reliability within this payload range.

1. The plot suggests that SpaceX's success rate is optimized for payloads within the typical commercial and satellite launch range.
2. Booster and launch strategy adjustments may correlate with reduced risks, as evidenced by clustering of successful outcomes for certain payload capacities and booster types.

Section 5

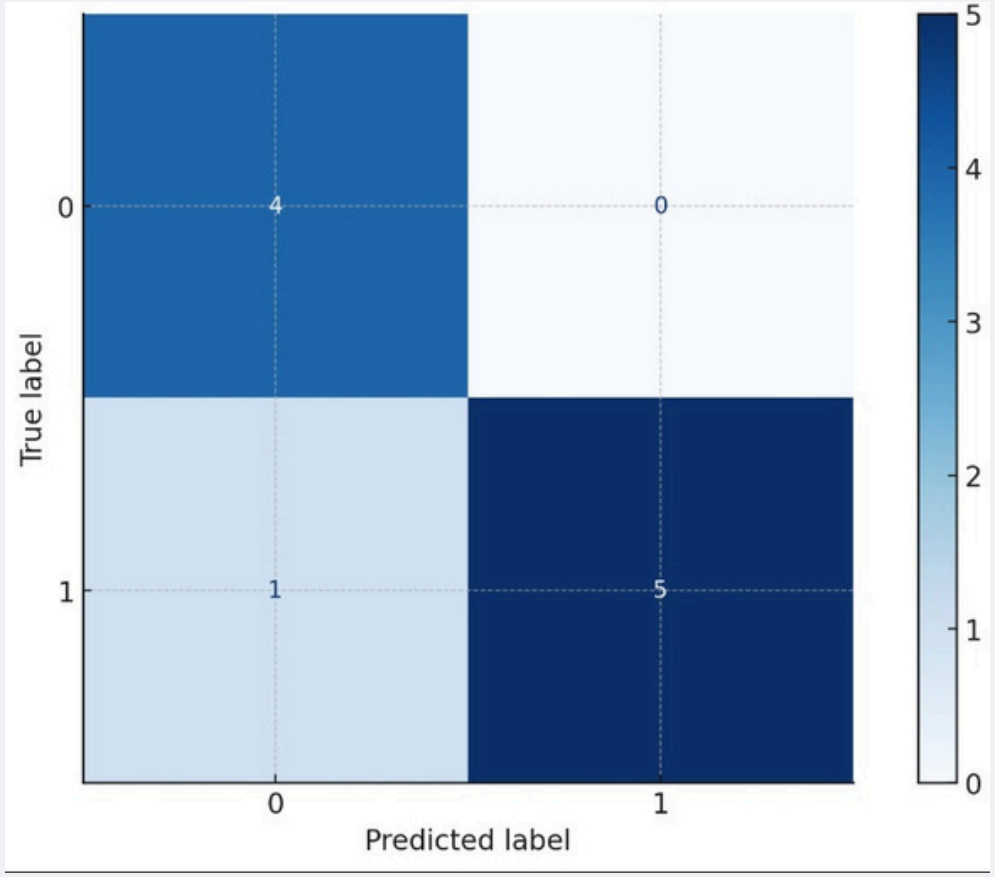
Predictive Analysis (Classification)

Classification Accuracy



The model with the highest accuracy is **Support Vector Machine (SVM)** with an accuracy of **0.85**.

Confusion Matrix



True Negative (TN): 4Correct predictions where the actual outcome was 0 (failure) and the model predicted 0.
False Positive (FP): 0Incorrect predictions where the actual outcome was 0 but the model predicted 1 (success).
False Negative (FN): 1Incorrect predictions where the actual outcome was 1 (success) but the model predicted 0.
True Positive (TP): 5Correct predictions where the actual outcome was 1 (success) and the model predicted 1.

Appendix

- Include any relevant assets like Python code snippets, SQL queries, charts, Notebook outputs, or data sets that you may have created during this project

Thank you!