IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Gloria Gonzalez
13Jan24

Outline

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

Executive Summary

The SpaceX launch data analysis employed a robust methodology leveraging advanced Python libraries, including Pandas for data processing, Dash for interactive web applications, and Plotly Express for dynamic visualizations. This approach facilitated a systematic exploration of launch site performance, success distribution, payload dynamics, and potential correlations between payload mass and launch success. Utilizing a user-friendly dropdown menu, stakeholders could navigate site-specific trends, while a dynamic pie chart vividly portrayed the distribution of successful launches. The interactive slider allowed real-time exploration of payload dynamics, complemented by a visually compelling scatter plot suggesting no clear correlation between payload mass and launch success. This methodology, combined with cutting-edge technologies, provided a sophisticated and comprehensive understanding of SpaceX's launch history.

The analysis uncovered insightful trends across four launch sites, emphasizing the effectiveness of dynamic visualizations in presenting success outcomes and payload dynamics. Preliminary findings indicated no clear correlation between payload mass and launch success, prompting stakeholders to further investigate specific launch outcomes, temporal trends, and additional success-influencing factors. The integration of advanced technologies ensured an engaging exploration, offering stakeholders an interactive and visually compelling journey through the complexities of SpaceX's launch data.

Introduction

- **Project background and context**

In the ever-evolving landscape of commercial space exploration, SpaceX has emerged as a trailblazer, achieving significant milestones in affordable and reusable rocket technology. As the commercial space age gains momentum, a new player, Space Y, seeks to enter the arena, aspiring to compete with SpaceX's established dominance. Founded by Billionaire industrialist Allon Musk, Space Y envisions a future where space travel is not only accessible but also economically viable. The focal point of this project is to navigate the complexities of pricing rocket launches, a critical aspect in the highly competitive industry. SpaceX's success lies in the reusability of its Falcon 9 rocket's first stage, a cost-saving innovation that Space Y aims to emulate. To address this challenge, the project endeavors to leverage machine learning and public information to predict the likelihood of first-stage reusability, thereby determining the cost of each launch for Space Y. This introduction sets the stage for an exploration into the dynamics of commercial space travel, the technological advancements pioneered by SpaceX, and the strategic goals of Space Y in this high-stakes endeavor.

- **Problems you want to find answers**

    - First-Stage Reusability Prediction:

The primary challenge revolves around predicting the success of first-stage landings for SpaceX's Falcon 9 rockets. By leveraging machine learning and public data, the project aims to provide Space Y with insights into the likelihood of reusing the first stage, a critical factor in determining launch costs.

    - Competitive Pricing Strategy:

Understanding the cost dynamics associated with reusable rocket technology, the project seeks to formulate a competitive pricing strategy for Space Y. This involves gathering comprehensive information about SpaceX's pricing model and using predictive analytics to establish a pricing structure that aligns with industry standards and ensures Space Y's competitiveness.

    - Strategic Decision-Making for Space Y:

As Space Y embarks on its journey to compete with SpaceX, strategic decision-making becomes paramount. The project aims to empower Space Y's team with informative dashboards, derived from SpaceX data, offering valuable insights into launch costs, success rates, and the intricacies of the commercial space market.

Section 1

Methodology

Methodology

- **Executive Summary: Data Analysis for Space Y's Rocket Launch Pricing Strategy**
- **Data Collection Methodology:**
  - Comprehensive sourcing of data from SpaceX and public repositories.
  - Meticulous data collection to ensure a robust foundation for analysis.
- **Data Wrangling:**
  - Thorough data cleaning and preprocessing for accuracy.
  - Rigorous handling of missing or inconsistent data points.
- **Exploratory Data Analysis (EDA):**
  - Utilization of SQL for in-depth exploration.
  - Visualization techniques applied to uncover key insights into SpaceX's launch dynamics.
- **Interactive Visual Analytics:**
  - Deployment of Folium and Plotly Dash for interactive dashboards.
  - Empowerment of Space Y's team with dynamic visualizations showcasing launch success rates, cost dynamics, and industry benchmarks.
- **Predictive Analysis using Classification Models:**
  - Building, tuning, and evaluating classification algorithms.
  - Crafting a predictive framework to determine the likelihood of first-stage reusability.
  - Integration of machine learning to adapt to evolving trends in the commercial space market.
- **Strategic Insights for Space Y:**
  - Actionable insights derived from the fusion of advanced analytics, machine learning, and interactive visualization.
  - Equipping Space Y with a strategic advantage in their pursuit to compete with industry giants like SpaceX.

Data Collection

1. **Initiating GET Request to SpaceX API:**
    1. Commenced data collection by executing a GET request to the SpaceX API.
    2. Retrieved live and real-time data directly from SpaceX's servers.
2. **API Response Handling and Data Scraping:**
    2. Processed the API response to extract relevant data.
    3. Applied data scraping techniques to retrieve specific information from the response.
3. **Data Cleansing and Wrangling:**
    3. Conducted initial validation checks to identify and address data anomalies.
    4. Utilized data wrangling techniques to handle missing values and outliers.
4. **Formatting and Standardization:**
    4. Formatted the collected data to ensure uniformity in structure and presentation.
    5. Standardized units, timestamps, and relevant variables for cohesive analysis.

- This streamlined data collection process involves initiating a GET request to the SpaceX API, handling the API response with data scraping, and subsequent steps of data cleansing, wrangling, and formatting for further analysis.

Data Collection – SpaceX API

- **GET Request to SpaceX API:**

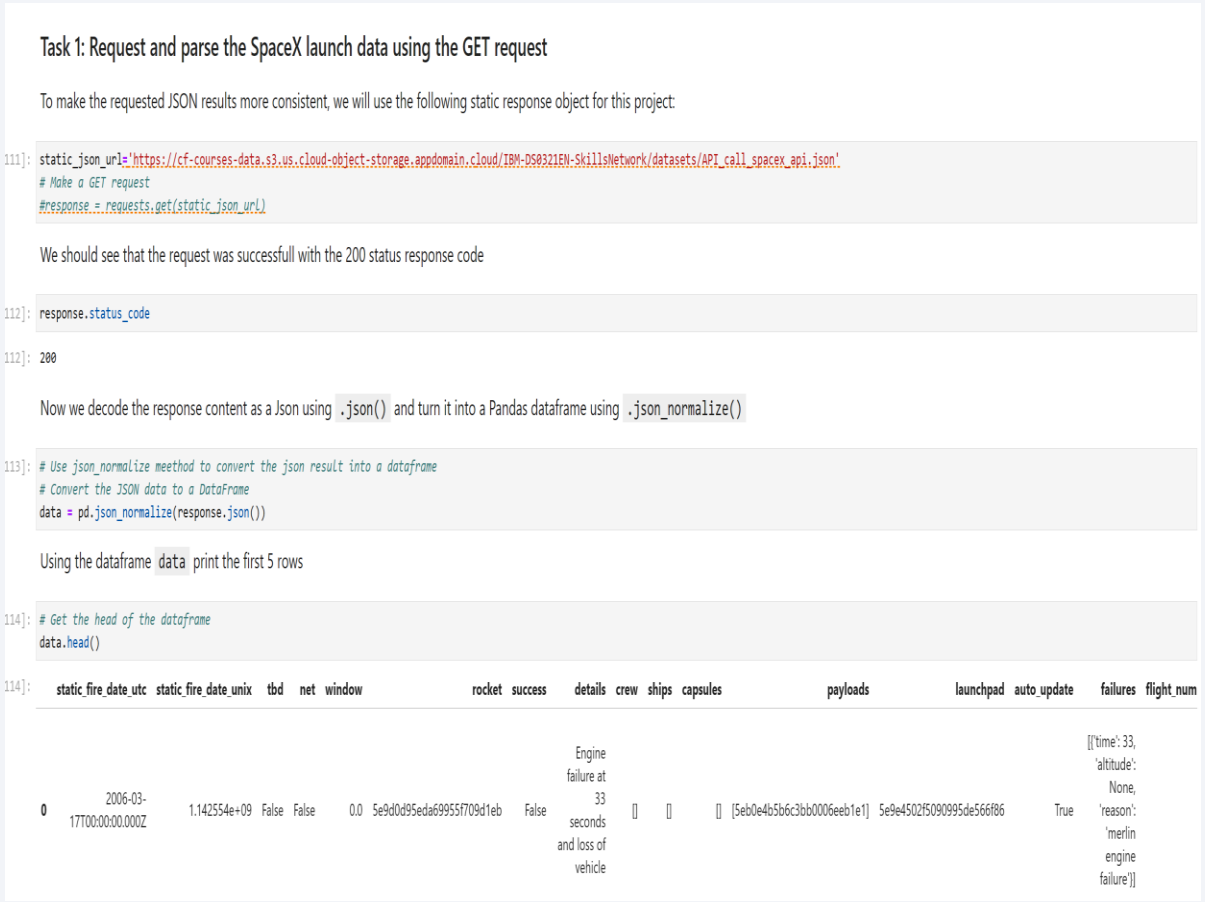
  Use a rectangle to represent the process of initiating a GET request to the SpaceX API.

- https://github.com/gg-777/capstone/blob/main/jupyter-labs-spacex-data-collection-api%20(1).ipynb

Data Collection - Scraping

- **Handling API Response and Data Scraping:**
  - Draw an arrow from the GET request rectangle to another rectangle representing the process of handling the API response and applying data scraping techniques.

- https://github.com/gg-777/capstone/blob/main/jupyter-labs-spacex-data-collection-api%20(1).ipynb



Task 1: Request and parse the SpaceX launch data using the GET request

To make the requested JSON results more consistent, we will use the following static response object for this project:

```
111]: static_json_url='https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DS0321EN-SkillsNetwork/datasets/API_call_spacex_api.json'
# Make a GET request
#response = requests.get(static_json_url)
```

We should see that the request was successfull with the 200 status response code

```
112]: response.status_code
```

```
112]: 200
```

Now we decode the response content as a Json using `.json()` and turn it into a Pandas dataframe using `.json_normalize()`

```
113]: # Use json_normalize meethod to convert the json result into a dataframe
# Convert the JSON data to a DataFrame
data = pd.json_normalize(response.json())
```

Using the dataframe `data` print the first 5 rows

```
114]: # Get the head of the dataframe
data.head()
```

114]:

	static_fire_date_utc	static_fire_date_unix	tbd	net	window	rocket	success	details	crew	ships	capsules	payloads	launchpad	auto_update	failures	flight_num
0	2006-03-17T00:00:00.000Z	1.142554e+09	False	False	0.0	5e9d0d95eda69955f709d1eb	False	Engine failure at 33 seconds and loss of vehicle	[]	[]	[]	[5eb0e4b5b6c3bb0006eeb1e1]	5e9e4502f5090995de566f86	True	[{'time': 33, 'altitude': None, 'reason': 'merlin engine failure'}]	

Data Wrangling

Data wrangling, also known as data munging or data preprocessing, is a crucial phase in the data analysis pipeline. It involves cleaning and transforming raw data into a structured and usable format for analysis. The data wrangling process typically includes the following key steps:

1. **Data Collection:**
    1. Acquiring raw data from diverse sources, such as APIs, databases, or external files.
2. **Initial Exploration:**
    1. Conducting a preliminary exploration of the data to identify potential issues like missing values, outliers, or inconsistent formats.
3. **Handling Missing Values:**
    1. Identifying and addressing missing data points through strategies like imputation or removal, depending on the extent and nature of missing values.
4. **Dealing with Outliers:**
    1. Identifying and addressing outliers that might skew analysis or lead to inaccurate insights.
5. **Data Cleaning:**
    1. Correcting errors, inconsistencies, or inaccuracies in the data to ensure its accuracy and reliability.
6. **Standardization and Normalization:**
    1. Standardizing units, formats, and scales across variables to facilitate meaningful comparisons.
7. **Transformations:**
    1. Applying transformations to variables, such as logarithmic or power transformations, to meet assumptions of statistical methods.
8. **Handling Duplicates:**
    1. Identifying and addressing duplicate records to avoid redundancy and ensure data integrity.
9. **Data Integration:**
    1. Combining data from multiple sources or datasets, creating a unified dataset for comprehensive analysis.
10. **Feature Engineering:**
    1. Creating new features or modifying existing ones to extract more meaningful insights from the data.
11. **Data Formatting:**
    1. Formatting the dataset for compatibility with analysis tools and models, ensuring a consistent and well-organized structure.
12. **Documentation:**
    1. Documenting the entire data wrangling process, including the steps taken and decisions made, to ensure transparency and reproducibility.

https://github.com/gg-777/capstone/blob/main/jupyter-labs-spacex-data-collection-api%20(1).ipynb

EDA with Data Visualization

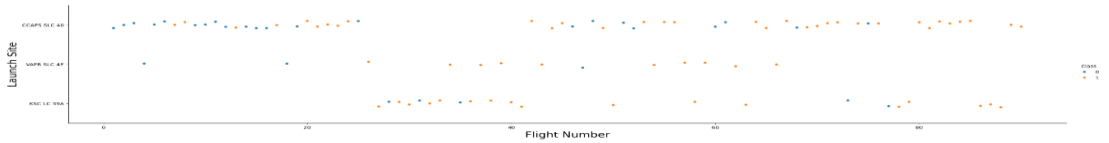
- **Flight Number vs. Launch Site:**
- *Chart Type:* Scatter plot.
- *Purpose:* The scatter plot depicting the relationship between Flight Number and Launch Site serves as a crucial visualization tool to gain insights into the distribution of spaceflights across different launch sites. By plotting Flight Numbers against Launch Sites, patterns and trends emerge, allowing for the identification of any sequential or chronological aspects related to the launches from specific sites. This visualization aids in understanding the historical progression of spaceflights, potentially uncovering site-specific launch preferences or mission sequencing strategies employed by the space agency.
- **Payload vs. Launch Site:**
- *Chart Type:* Box plot or bar chart.
- *Purpose:* The visual representation of the relationship between Payload and Launch Site, through a box plot or bar chart, is instrumental in assessing the distribution of payload masses associated with different launch sites. This visualization provides valuable insights into the payload capacities and variations among various launch sites. Analyzing payload distributions assists in evaluating the payload management capabilities of each site and identifying potential correlations between launch sites and the types of payloads they accommodate. This understanding is crucial for optimizing payload planning and allocation strategies across different launch facilities, contributing to enhanced operational efficiency in the aerospace industry.

- [capstone/jupyter-labs-eda-dataviz.ipynb.jupyterlite (1).ipynb at main · gg-777/capstone (github.com)]

TASK 1: Visualize the relationship between Flight Number and Launch Site

Use the function `catplot` to plot `FlightNumber` vs `LaunchSite`, set the parameter `x` parameter to `FlightNumber`,set the `y` to `Launch Site` and set the parameter `hue` to `'class'`

```
# Plot a scatter point chart with x axis to be Flight Number and y axis to be the Launch site, and hue to be the class va
sns.catplot(y="LaunchSite", x="FlightNumber", hue="Class", data=df, aspect = 5)
plt.xlabel("Flight Number",fontsize=20)
plt.ylabel("Launch Site",fontsize=20)
plt.show()
```

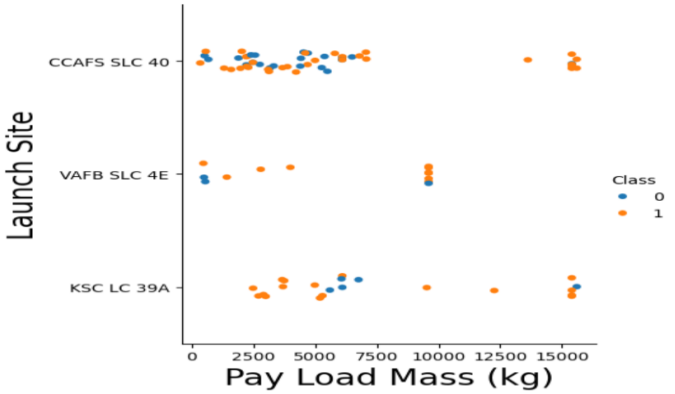


Now try to explain the patterns you found in the Flight Number vs. Launch Site scatter point plots.

TASK 2: Visualize the relationship between Payload and Launch Site

We also want to observe if there is any relationship between launch sites and their payload mass.

```
# Plot a scatter point chart with x axis to be Pay Load Mass (kg) and y axis to be the Launch site, and hue to be the cla
sns.catplot(y="LaunchSite", x="PayloadMass", hue="Class", data=df)
plt.xlabel("Pay Load Mass (kg)",fontsize=20)
plt.ylabel("Launch Site",fontsize=20)
plt.show()
```



EDA with SQL

- Display the names of the unique launch sites in the space mission
    - %sql SELECT DISTINCT "Launch Site" FROM SPACEXTABLE;
- Display 5 records where launch sites begin with the string 'CCA'
- %sql select * from SPACEXTBL where LAUNCH_SITE like 'CCA%' limit 5
- Display the total payload mass carried by boosters launched by NASA (CRS)
- %sql select sum(PAYLOAD_MASS__KG_) from SPACEXTBL where CUSTOMER = 'NASA (CRS)'
- Display average payload mass carried by booster version F9 v1.1
- %sql select avg(PAYLOAD_MASS__KG_) from SPACEXTBL where BOOSTER_VERSION = 'F9 v1.1'
- List the date when the first succesful landing outcome in ground pad was acheived.
- %sql select min(DATE) from SPACEXTBL where Landing__Outcome = 'Success (ground pad)'
- List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000
- %sql select BOOSTER_VERSION from SPACEXTBL where "Landing Outcome" = 'Success (drone ship)' and "Payload Mass (kg)" > 4000 and "Payload Mass (kg)" < 6000;

Remove blank rows from table

%sql create table SPACEXTABLE as select * from SPACEXTBL where Date is not null

List the total number of successful and failure mission outcomes

%sql SELECT "Mission Outcome", COUNT(*) AS "Total Outcomes" FROM SPACEXTABLE GROUP BY "Mission Outcome";

List the names of the booster_versions which have carried the maximum payload mass. Use a subquery

%sql select BOOSTER_VERSION from SPACEXTBL where PAYLOAD_MASS__KG_ = (select max(PAYLOAD_MASS__KG_) from SPACEXTBL)

List the records which will display the month names, failure landing_outcomes in drone ship ,booster versions, launch_site for the months in year 2015.

%sql SELECT strftime('%m', "Date") AS "Month", "Landing Outcome", "Booster Version", "Launch Site" FROM SPACEXTABLE WHERE substr("Date", 0, 5) = '2015' AND "Landing Outcome" LIKE 'Failure (drone ship)';

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.

%sql select * from SPACEXTBL where Landing__Outcome like 'Success%' and (DATE between '2010-06-04' and '2017-03-20') order by "Count" desc;

https://github.com/gg-777/capstone/tree/main/

Build an Interactive Map with Folium

1. **Markers:**
    1. Employed markers to pinpoint specific locations on the map, providing a visual reference for launch sites or other critical points of interest. Markers serve as intuitive indicators for users to quickly identify and associate information with specific geographical coordinates.
2. **Circles:**
    1. Utilized circles to represent areas of influence, significance, or coverage around specific locations on the map. This can be particularly valuable for visualizing launch site radius, potential impact zones, or coverage areas for satellite communication.
3. **Lines:**
    1. Incorporated lines to depict connections or trajectories between different points on the map. Lines were employed to visualize flight paths, routes, or connections between launch sites and destinations, offering a clear representation of spatial relationships.
4. **Polygons:**
    1. Integrated polygons to outline specific regions or boundaries on the map. This is useful for highlighting areas of interest, such as operational zones, regulatory boundaries, or designated regions relevant to the analysis.
5. **Popups:**
    1. Implemented popups associated with markers to display additional information when clicked. Popups provide a user-friendly way to present detailed data, allowing users to access supplementary information without cluttering the map interface.

```
headings = []
for key,values in dict(launch_dict).items():
    if key not in headings:
        headings.append(key)
    if values is None:
        del launch_dict[key]

def pad_dict_list(dict_list, padel):
    lmax = 0
    for lname in dict_list.keys():
        lmax = max(lmax, len(dict_list[lname]))
    for lname in dict_list.keys():
        ll = len(dict_list[lname])
        if  ll < lmax:
            dict_list[lname] += [padel] * (lmax - ll)
    return dict_list

pad_dict_list(launch_dict,0)

df = pd.DataFrame.from_dict(launch_dict)
df.head()
```

	Flight No.	Launch site	Payload	Payload mass	Orbit	Customer	Launch outcome	Version Booster	Booster landing	Date	Time
0	1	CCAFS	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success\n	F9 v1.0B0003.1	Failure	4 June 2010	18:45
1	2	CCAFS	Dragon	0	LEO	NASA	Success	F9 v1.0B0004.1	Failure	8 December 2010	15:43
2	3	CCAFS	Dragon	525 kg	LEO	NASA	Success	F9 v1.0B0005.1	No attempt\n	22 May 2012	07:44
3	4	CCAFS	SpaceX CRS-1	4,700 kg	LEO	NASA	Success\n	F9 v1.0B0006.1	No attempt	8 October 2012	00:35
4	5	CCAFS	SpaceX CRS-2	4,877 kg	LEO	NASA	Success\n	F9 v1.0B0007.1	No attempt\n	1 March 2013	15:10

- https://github.com/gg-777/capstone

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
- https://github.com/gg-777/capstone/blob/main/SpaceX%20Plotly.py

Predictive Analysis (Classification)

- Data Preparation:
- Collected and preprocessed data from SpaceX API.
- Conducted exploratory data analysis (EDA) to understand features and target variable distribution.
- Feature Engineering:
- Engineered relevant features like Booster Version Category.
- Transformed categorical variables and standardized numerical features.
- Train-Test Split:
- Divided the dataset into training and testing sets for model evaluation.
- Model Selection:
- Chose classification models (e.g., Logistic Regression, Random Forest, SVM) based on dataset characteristics.
- Considered ensemble methods for improved performance.
- Baseline Model:
- Created a baseline model for comparison.
- Evaluated model performance using metrics like accuracy, precision, recall, and F1-score.
- Model Evaluation:

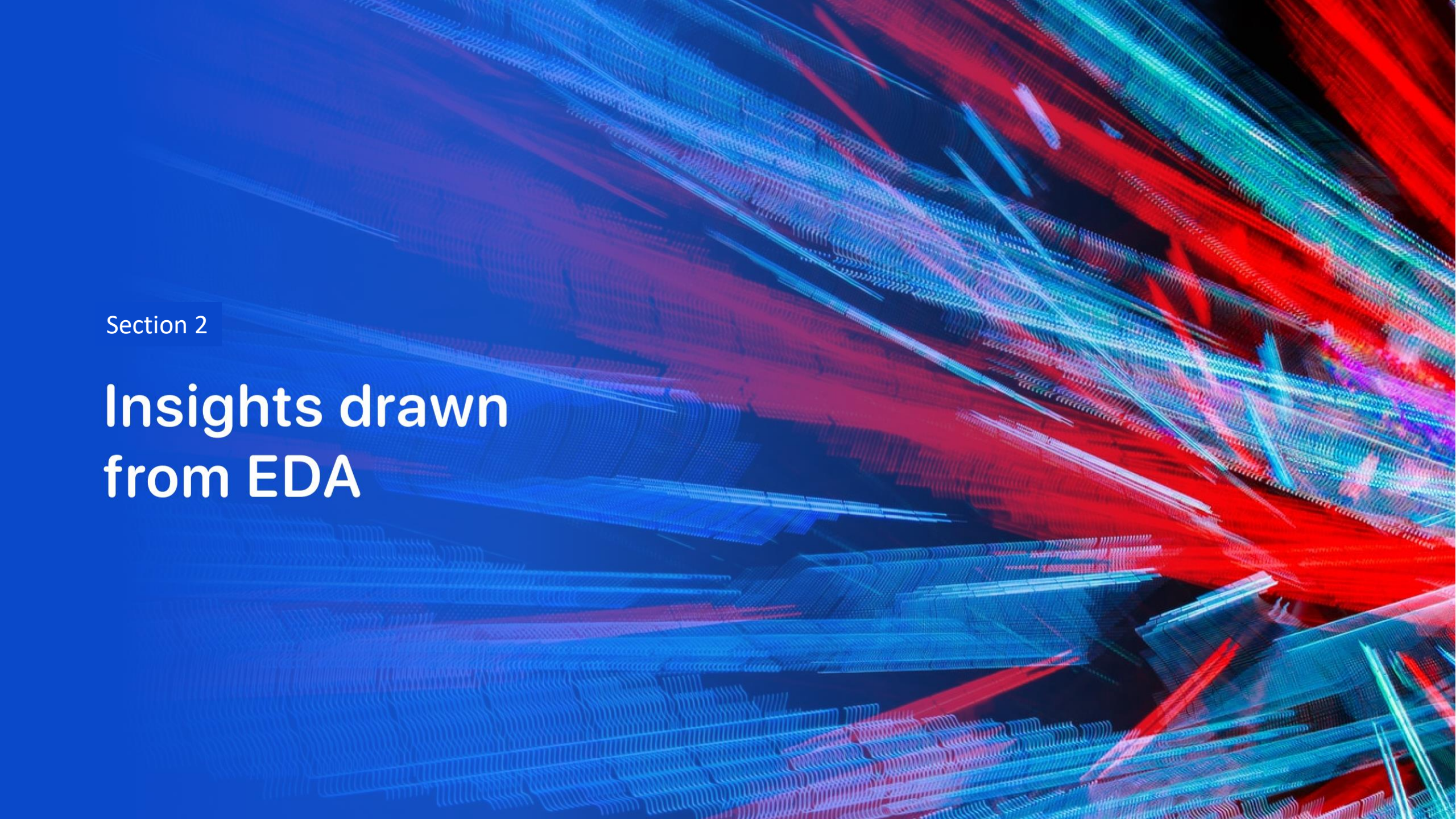
Predictive Analysis (Classification)

- Utilized cross-validation to assess model performance robustly.
- Tuned hyperparameters to enhance model effectiveness.
- Feature Importance Analysis:
- Analyzed feature importance to understand variables contributing significantly to the model.
- Model Iteration:
- Iteratively refined models based on evaluation results.
- Applied feature selection techniques to enhance model simplicity and performance.
- Validation on Test Set:
- Validated the final model on the test set to assess generalization capabilities.
- Model Comparison:
- Compared multiple models based on key metrics.
- Selected the best-performing classification model for deployment.

- https://github.com/gg-777/capstone/blob/main/Machine%20Learning%20Prediction%20lab.ipynb

Results

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


Section 2
Insights drawn from EDA

Flight Number vs. Launch Site

- Show a scatter plot of Flight Number vs. Launch Site

- Show the screenshot of the scatter plot with explanations

Payload vs. Launch Site

- Show a scatter plot of Payload vs. Launch Site

- Show the screenshot of the scatter plot with explanations

Success Rate vs. Orbit Type

- Show a bar chart for the success rate of each orbit type

- 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. Orbit Type

- 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
- Present your query result with a short explanation here

Launch Site Names Begin with 'CCA'

- Find 5 records where launch sites begin with `CCA`
- Present your query result with a short explanation here

Total Payload Mass

- Calculate the total payload carried by boosters from NASA
- Present your query result with a short explanation here

Average Payload Mass by F9 v1.1

- Calculate the average payload mass carried by booster version F9 v1.1
- Present your query result with a short explanation here

First Successful Ground Landing Date

- Find the dates of the first successful landing outcome on ground pad
- Present your query result with a short explanation here

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

- Present your query result with a short explanation here

Total Number of Successful and Failure Mission Outcomes

- Calculate the total number of successful and failure mission outcomes
- Present your query result with a short explanation here

Boosters Carried Maximum Payload

- List the names of the booster which have carried the maximum payload mass
- Present your query result with a short explanation here

2015 Launch Records

- List the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015

- Present your query result with a short explanation here

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

- Present your query result with a short explanation here


Section 3
Launch Sites
Proximities Analysis

<Folium Map Screenshot 1>

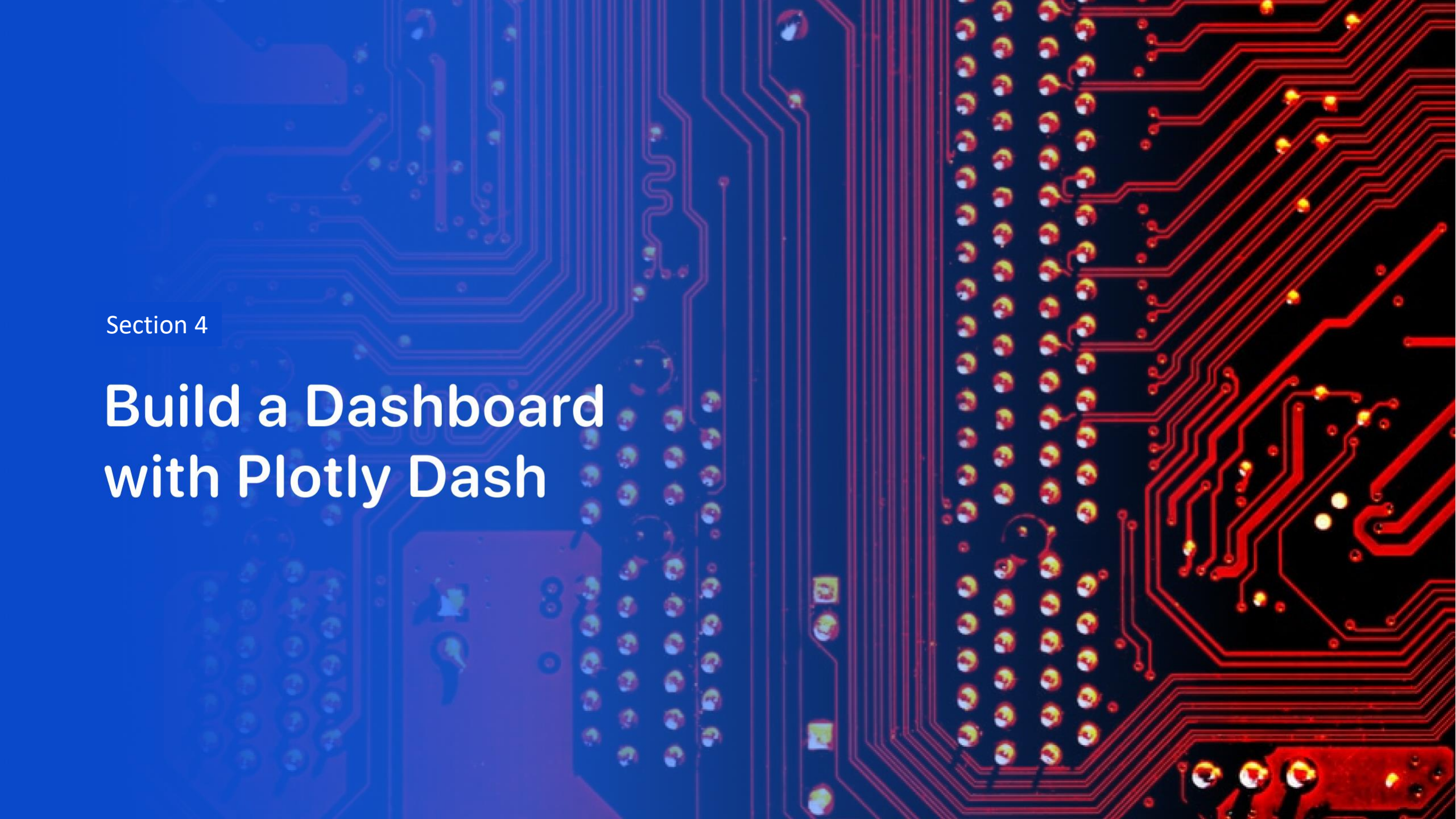
- Replace <Folium map screenshot 1> title with an appropriate title
- Explore the generated folium map and make a proper screenshot to include all launch sites' location markers on a global map
- Explain the important elements and findings on the screenshot

<Folium Map Screenshot 2>

- Replace <Folium map screenshot 2> title with an appropriate title
- Explore the folium map and make a proper screenshot to show the color-labeled launch outcomes on the map
- Explain the important elements and findings on the screenshot

<Folium Map Screenshot 3>

- Replace <Folium map screenshot 3> title with an appropriate title
- 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


Section 4
Build a Dashboard with Plotly Dash

<Dashboard Screenshot 1>

- Replace <Dashboard screenshot 1> title with an appropriate title
- Show the screenshot of launch success count for all sites, in a piechart
- Explain the important elements and findings on the screenshot

<Dashboard Screenshot 2>

- Replace <Dashboard screenshot 2> title with an appropriate title
- Show the screenshot of the piechart for the launch site with highest launch success ratio
- Explain the important elements and findings on the screenshot

<Dashboard Screenshot 3>

- Replace <Dashboard screenshot 3> title with an appropriate title
- Show screenshots of Payload vs. Launch Outcome scatter plot for all sites, with different payload selected in the range slider
- Explain the important elements and findings on the screenshot, such as which payload range or booster version have the largest success rate, etc.

Section 5

Predictive Analysis (Classification)

Classification Accuracy

- Visualize the built model accuracy for all built classification models, in a bar chart

- Find which model has the highest classification accuracy

Confusion Matrix

- Show the confusion matrix of the best performing model with an explanation

Conclusions

- Point 1
- Point 2
- Point 3
- Point 4
- …

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!