IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

<Nour Stifan>
<12/26/2023>

Outline

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

Executive Summary

- **Summary of methodologies**

- We have more than one methodology in this course:

- Data collection methodology: Using SpaceX REST API to get real-time data.

- Data processing methodology: Using data structuring techniques, where JSON data is normalized and missing values are processed for a coherent data set.

- Exploratory data analysis methodology: identifying patterns and trends by conducting a preliminary examination. It uses descriptive statistics and visual examination techniques.

- Pandas and Matplotlib methodology: Use of Python libraries for exploratory analysis, focusing on statistical analysis and data visualization.

- Interactive dashboard methodology: Using interactive tools to create dynamic visual analyses.

Summary of all results

- A set of SpaceX launch data was collected successfully and with good accuracy.

- An organized data set was created and ready for analysis.

- Data distribution, main trends, correlations were identified.

- Initial insights into SpaceX launch patterns have been gained.

- Data patterns have been visualized for better understanding.

- An effective dashboard has been developed.

Introduction

At this time there is progress in space exploration, understanding the dynamics of space missions is important, especially launch operations. This project focuses on analyzing data from SpaceX, the aerospace company known for its approach to space travel and exploration. The growing number of launches at SpaceX provides a wealth of data for analysis. The project takes advantage of this dataset by using the SpaceX REST API to collect comprehensive data about SpaceX launches. The data includes many complex details, such as missile specifications, payloads, launch and landing details, and results.

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - API: Use SpaceX REST API to collect data. Select target endpoints to collect detailed launch data such as missile specifications, payloads and launch results.
  - Web scraping: - Use web scraping techniques to obtain supplementary data. - Use the Python library BeautifulSoup to extract HTML content.
- Perform data wrangling
  - Inconsistencies and irrelevant information are removed through data cleaning. Data is formatted by structuring it using a usable format.
  - Nested structures are converted into flat tables by normalizing JSON data from the API. Data types are converted and missing values are handled through data transformation.

- **Describe how data was processed**
  - Data from different sources such as API and web aggregation are combined into one dataset. Data is summarized through aggregation to get clear insights.
  - Ratios are calculated or categorical data are extracted from existing variables to enhance the data set.
- **Perform exploratory data analysis (EDA) using visualization and SQL**
  - We use tools such as Matplotlib and Seaborn to visualize data.
  - Graphs, scatter plots, or bar charts are designed to monitor the relationships between data and distributions.
  - Specific data is extracted and relational analysis is performed by executing SQL queries. Aggregation functions used in SQL are used to summarize data information.
- **Perform interactive visual analytics using Folium and Plotly Dash**
  - Interactive maps are created to visualize the geospatial aspects of launch data through Volume.
  - Real-time data interaction and visualization is enabled for users through the use of Plotly Dash.

Data Collection

Start

- **Identify Data Requirements**

Select Data Sources

- API (SpaceX REST API)
- Web Scraping

API Data Collection

- Target Endpoints
- Retrieve JSON Data

Web Scraping

- HTML Parsing
- Data Extraction

Data Aggregation

- Merge API and Web Scraped Data

Data Validation

- Accuracy and Integrity Checks

Data Storage

- Secure and Structured Database

Data Collection – SpaceX API

Start Establish API Requirements

Set Up API Environment

API Endpoint Targeting

-Select Relevant Endpoints

-Perform GET Requests

Data Structuring

-Transform Data

-Store in Database

End

Data Retrieval

-Extract JSON Data

Feedback Loop

-Review and Modify

Continuous Monitoring & Updating

Data Validation & Cleansing

-Check Data Integrity

-Clean and Preprocess

Data Collection - Scraping

Identify Target Websites

Define Data Requirements

Setup Scraping Environment

Access Website Content

Parse HTML Content

Data Cleaning & Preprocessing

Data Integration

Store & Backup Data

Continuous Monitoring & Updates

Data Wrangling

- Data processing for the SpaceX project took place through several key steps to convert the raw data into an analyzable format.

- Integrating data collected from various sources, including the SpaceX REST API and web aggregation.

- Conduct a data cleaning process to correct inconsistencies and deal with missing values.

- Normalize data fields, standardize formats, and classify data.

- Simplify the data set by normalizing numeric fields to maintain scale uniformity.

- Verify the accuracy of processed data and store it effectively to be ready for analysis and modeling.

EDA with Data Visualization

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

-catplot, barplot, scatterplot, and lineplot.

Catplot: To visually represent the relationship and distribution between a categorical variable and other variables, while identifying patterns and trends in categorical data.

Barplot: To compare the size of different groups or categories, as it facilitates analysis of differences in quantitative data.

Scatterplot: To observe possible relationships and correlations between two numerical variables where the influence of one variable on another is analyzed.

Lineplot: To visualize changes over time, as it shows how a quantitative variable develops or fluctuates in a continuous manner.

EDA with SQL

- Using bullet point format, summarize the SQL queries you performed
- Query: SELECT DISTINCT Launch_Site FROM SPACEXTBL;
- Query: SELECT * FROM SPACEXTBL WHERE Launch_Site LIKE 'CCA%' LIMIT 5;
- Query: SELECT SUM(PAYLOAD_MASS__KG_) AS TotalPayload FROM SPACEXTBL WHERE Customer = 'NASA (CRS)';
- Query: SELECT AVG(PAYLOAD_MASS__KG_) AS Average_Payload_Mass FROM SPACEXTBL WHERE Booster_Version LIKE 'F9 v1.1%';
- Query: SELECT MIN(Date) AS FirstSuccessDate FROM SPACEXTBL WHERE Landing_Outcome = 'Success (ground pad)';
- Query: SELECT Booster_Version FROM SPACEXTBL WHERE Landing_Outcome = 'Success (drone ship)' AND PAYLOAD_MASS__KG_ > 4000 AND
- Query: SELECT Mission_Outcome, COUNT(*) as Total FROM SPACEXTBL GROUP BY Mission_Outcome;
- Query: SELECT Booster_Version FROM SPACEXTBL WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTBL);
- Query: SELECT substr(Date, 6, 2) AS Month, Booster_Version, Launch_Site, Landing_Outcome FROM SPACEXTBL WHERE substr(Date, 0, 5) = '2015' AND Landing_Outcome LIKE 'Failure (drone ship)%';
- Query: SELECT Landing_Outcome, COUNT(*) AS OutcomeCount FROM SPACEXTBL WHERE Date BETWEEN '2010-06-04' AND '2017-03-20' GROUP BY Landing_Outcome ORDER BY OutcomeCount DESC;

Build an Interactive Map with Folium

- **Summarize what map objects such as markers, circles, lines, etc. you created and added to a folium map**
- Circle, marker, and lines.
- **Explain why you added those objects**

Circle: To visually represent the geographic radius or coverage area around a specific point, such as a launch site.

Marker: To pinpoint exact locations on a map such as the exact coordinates of SpaceX launch pads, providing a visual reference for specific locations.

Lines: To clarify the paths and communications between different points, such as the SpaceX launch path or the path between the launch site and the landing area, specifying the direction and distance.

Build a Dashboard with Plotly Dash

- **Summarize what plots/graphs and interactions you have added to a dashboard**

Line Chart, Pie Chart, Bar Chart, and Stacked Bar Chart.

- **Explain why you added those plots and interactions**

Line Chart: To view annual car sales during recessions.

Pie Chart: To view the share of spending by vehicle type during recessions.

Bar Chart: Shows average sales by vehicle type during a recession.

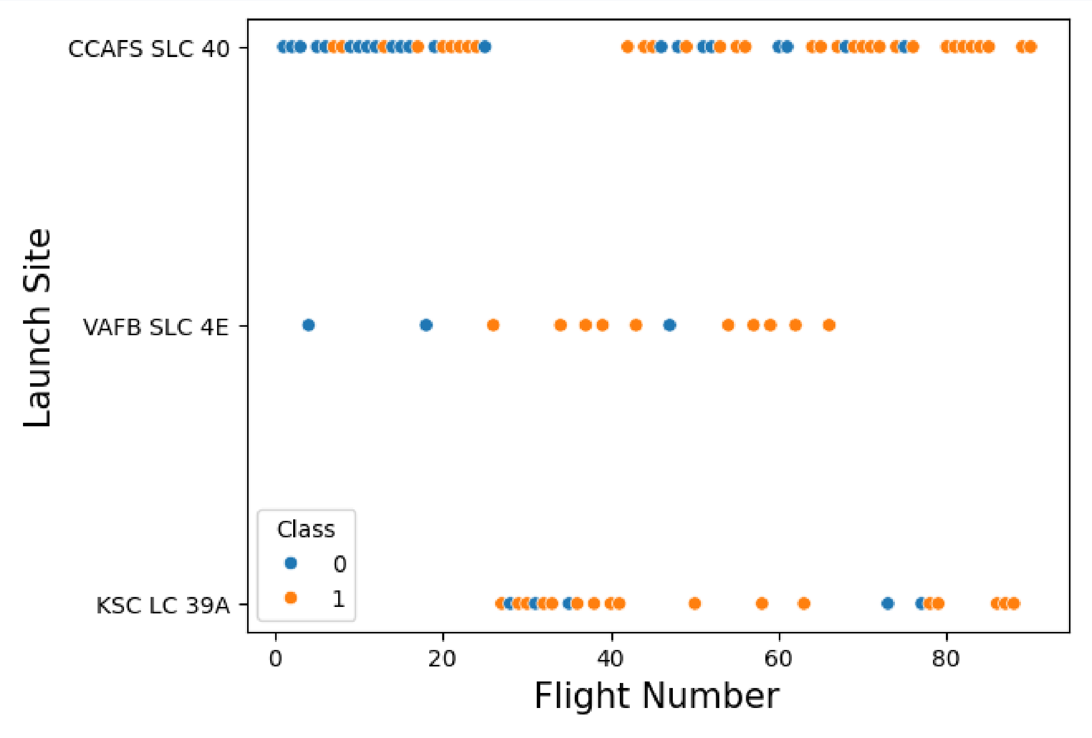
Stacked Bar Chart: To display the effect of the unemployment rate on vehicle sales, the sectors of each column represent different unemployment rates.

Section 2

Insights drawn from EDA

Flight Number vs. Launch Site

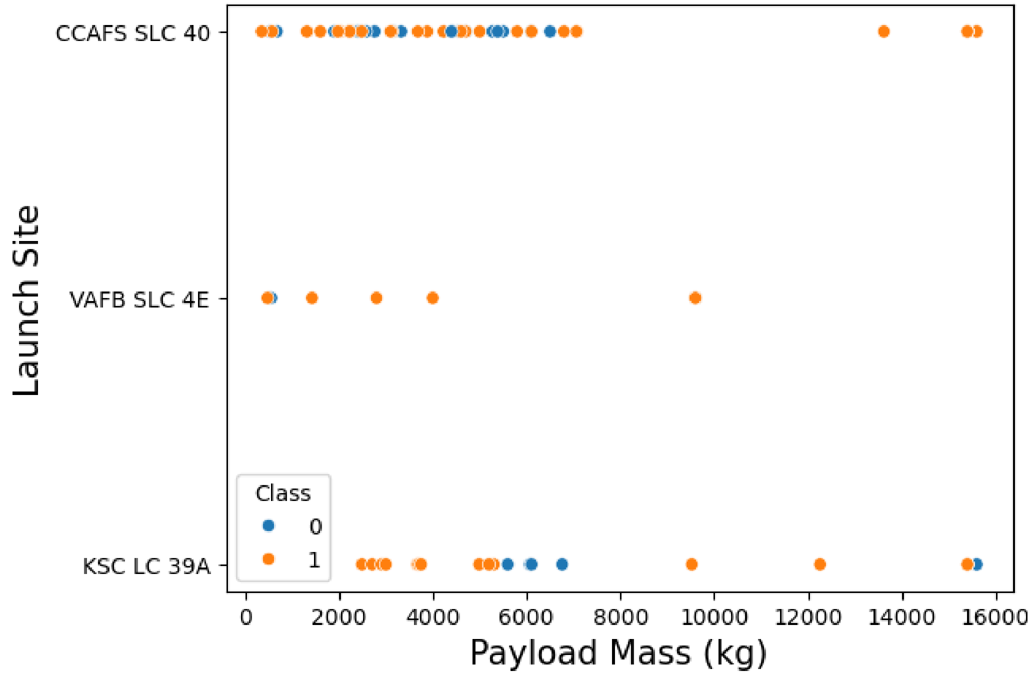
- Show a scatter plot of Flight Number vs. Launch Site



The screenshot depicts a scatter plot of the relationship between SpaceX flight numbers and their launch sites, with data points colored according to a binary category (0 or 1), which can indicate a measure of success, mission type, or other categorical score. The horizontal spread of dots indicates the flight numbers at each level of the launch site and is identified by color.

Payload vs. Launch Site

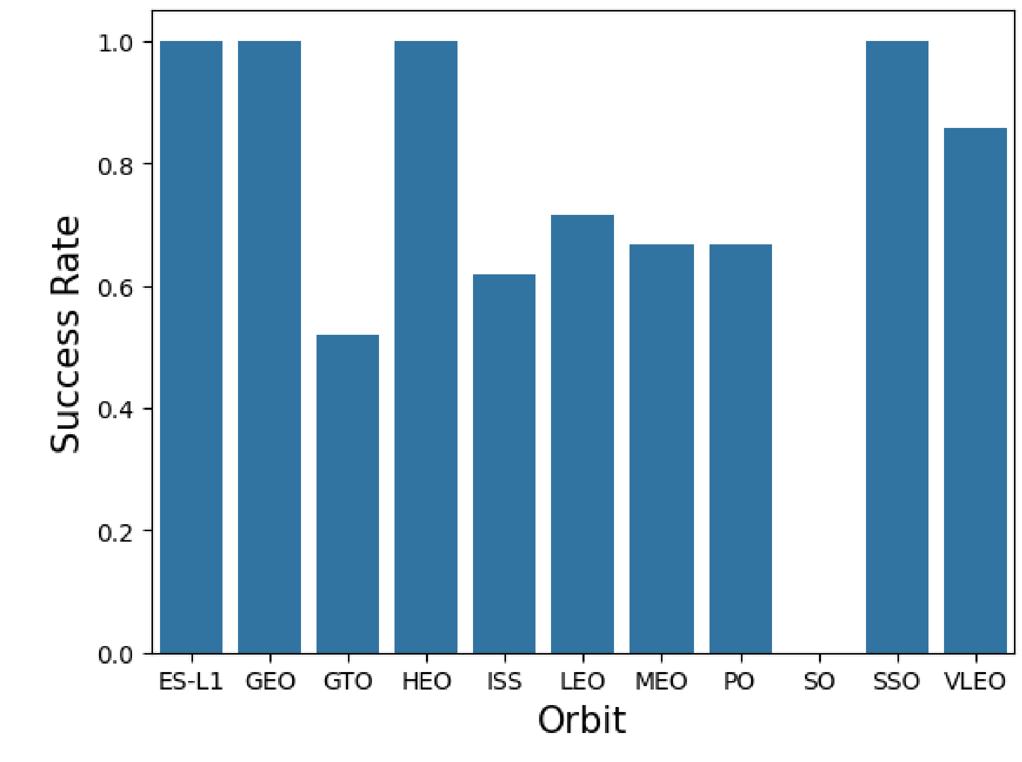
- Show a scatter plot of Payload vs. Launch Site



The screenshot includes a scatter plot graph. The scatter plot shows the relationship between launch sites and payload masses, with color-coded data points representing two categories. Most of the data points for CCAFS SLC 40 fall into Category 1, while KSC LC 39A shows a mixture of both categories.

Success Rate vs. Orbit Type

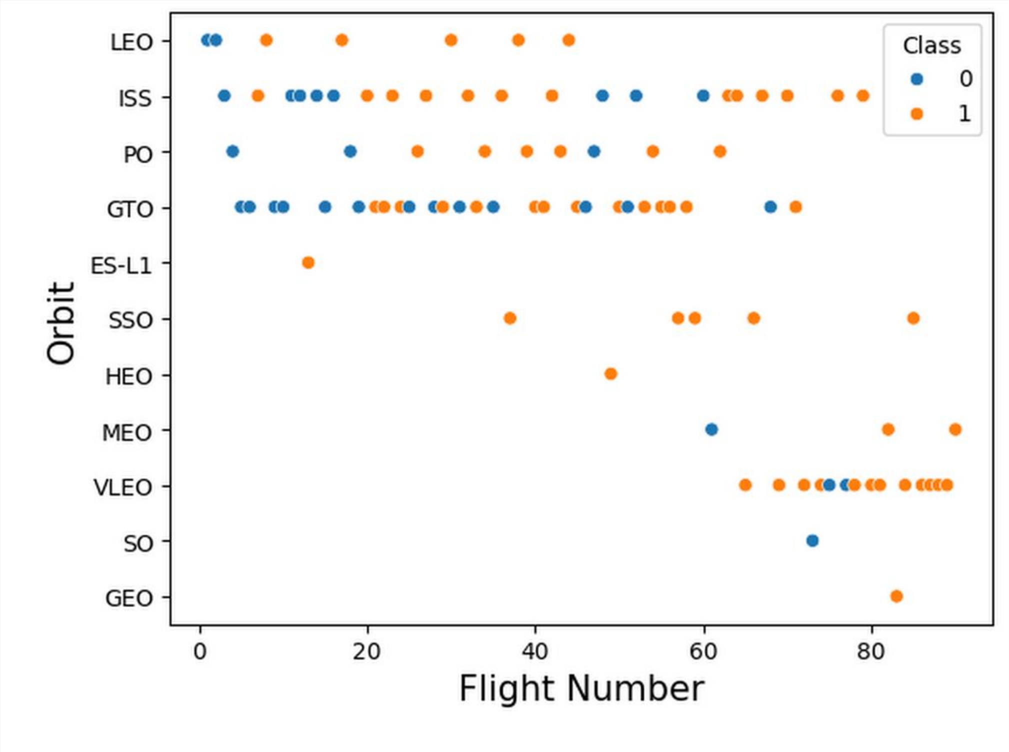
- Show a bar chart for the success rate of each orbit type



The screenshot includes a bar chart indicating the success rate of space missions to different orbital destinations, and the height of each bar indicates the percentage of successful missions.

Flight Number vs. Orbit Type

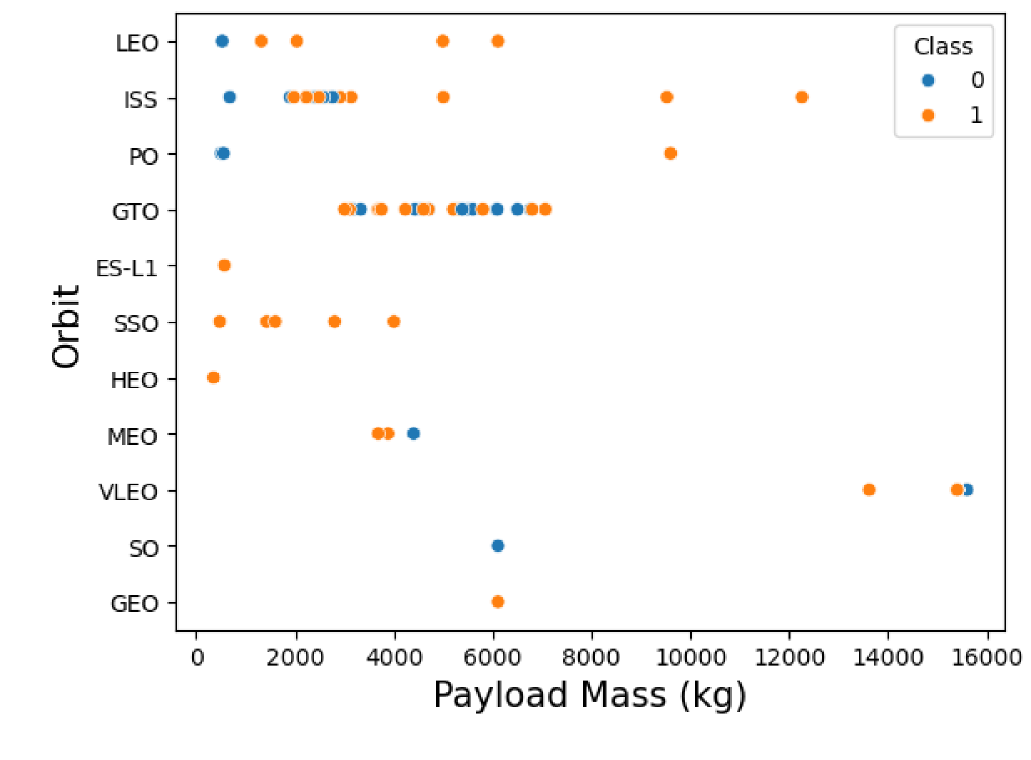
- Show a scatter point of Flight number vs. Orbit type



The screenshot shows a scatter plot representing the various SpaceX flights, categorized by orbit type on the y-axis and flight number on the x-axis, with data points color-coded to represent two categories (0 or 1), which may indicate mission success or failure.

Payload vs. Orbit Type

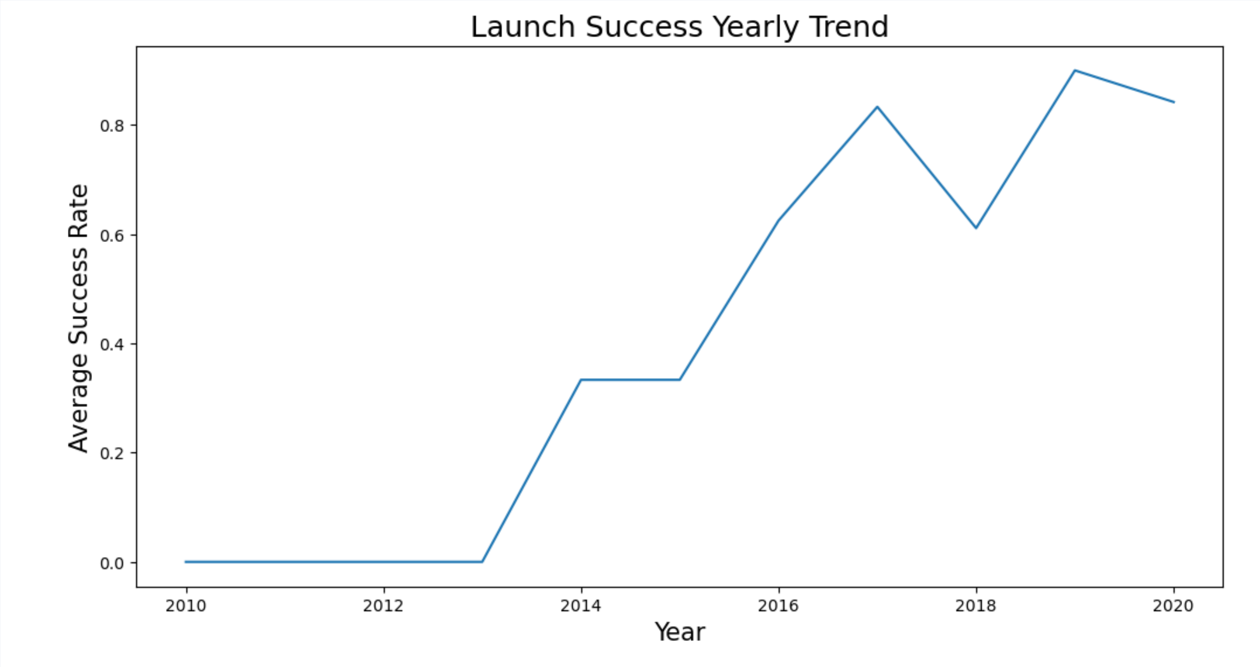
- Show a scatter point of payload vs. orbit type



A scatterplot indicating the relationship between payload mass (in kilograms) and the type of orbit selected for various space missions, with points marked by a binary class indicator (0 or 1). The spread of data points across the payload mass range for each orbit type may indicate correlations between payload mass and the mission success category, or another categorical outcome represented by the category.

Launch Success Yearly Trend

- Show a line chart of yearly average success rate



This line graph shows the trend of average launch success rates over the years, from 2010 to around 2020. The graph indicates an overall increasing trend in launch success rate over time.

All Launch Site Names

- Find the names of the unique launch sites
- %%sql
- SELECT DISTINCT Launch_Site
- FROM SPACEXTBL;
- Present your query result with a short explanation here

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

The result indicates that there are four distinct launch sites from which SpaceX was launched: CCAFS LC-40, VAFB SLC-4E, KSC LC-39A, and CCAFS SLC-40.

Launch Site Names Begin with 'CCA'

- Find 5 records where launch sites begin with `CCA`
- %%sql
- SELECT *
- FROM SPACEXTBL
- WHERE Launch_Site LIKE 'CCA%'
- LIMIT 5;
- Present your query result with a short explanation here

The resulting data includes information about the date and time of each launch, the booster version, the launch site, payload, payload mass, orbit, customer, mission outcome, and landing outcome. The output specifically shows details for five launches, all of which were successful missions, although the first two had failures with the parachute landing approach.

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
2010-12-08	15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight C1, two CubeSats, barrel of Brouere cheese	0	LEO (ISS)	NASA (COTS) NRO	Success	Failure (parachute)
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
2012-10-08	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
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

```
%%sql
SELECT SUM(PAYLOAD_MASS__KG_) AS TotalPayload
FROM SPACEXTBL
WHERE Customer = 'NASA (CRS)';
```

- Present your query result with a short explanation here

TotalPayload
45596

The query result indicates that the combined payload mass for these missions is 45.596 kilograms.

Average Payload Mass by F9 v1.1

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

```
%%sql
SELECT AVG(PAYLOAD_MASS__KG_) AS Average_Payload_Mass
FROM SPACEXTBL
WHERE Booster_Version LIKE 'F9 v1.1%';
```

- Present your query result with a short explanation here

Average_Payload_Mass
2534.6666666666665

The query result indicates that the average payload mass carried by this booster version is approximately 2534.67 kilograms.

First Successful Ground Landing Date

- Find the dates of the first successful landing outcome on ground pad

```
%%sql
SELECT MIN(Date) AS FirstSuccessDate
FROM SPACEXTBL
WHERE Landing_Outcome = 'Success (ground pad)';
```

- Present your query result with a short explanation here

FirstSuccessDate
2015-12-22

The result indicates that the first successful ground pad landing occurred on December 22, 2015.

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

```
%%sql
SELECT Booster_Version
FROM SPACEXTBL
WHERE Landing_Outcome = 'Success (drone ship)'
  AND PAYLOAD_MASS__KG_ > 4000
 AND PAYLOAD_MASS__KG_< 6000;
```

- Present your query result with a short explanation here

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

The result lists four Falcon 9 Full Thrust booster versions that meet these criteria: F9 FT B1022, F9 FT B1026, F9 FT B1021.2, and F9 FT B1031.2.

Total Number of Successful and Failure Mission Outcomes

- Calculate the total number of successful and failure mission outcomes

```
%%sql
SELECT
    Mission_Outcome,
    COUNT(*) as Total
FROM
    SPACEXTBL
GROUP BY
    Mission_Outcome;
```

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

The results indicate that there was 1 mission that resulted in a failure during flight, 98 missions that were successful, 1 mission that was a success.

Boosters Carried Maximum Payload

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

```
%%sql

SELECT Booster_Version

FROM SPACEXTBL

WHERE PAYLOAD_MASS__KG_  = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTBL);
```

The result lists multiple instances of the Falcon 9 Block 5 boosters, indicating that these specific versions have been used to launch the heaviest payloads.

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

- %%sql
- SELECT substr(Date, 6, 2) AS Month, Booster_Version, Launch_Site, Landing_Outcome
- FROM SPACEXTBL
- WHERE substr(Date, 0, 5) = '2015'
- AND Landing_Outcome LIKE 'Failure (drone ship)%';

In 2015 results show a failure of version 1.1 of the Falcon 9 rocket, one in January (01) and the other in April (04).

- Present your query result with a short explanation here

Month	Booster_Version	Launch_Site	Landing_Outcome
01	F9 v1.1 B1012	CCAFS LC-40	Failure (drone ship)
04	F9 v1.1 B1015	CCAFS LC-40	Failure (drone ship)

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
- %%sql
- SELECT Landing_Outcome, COUNT(*) AS OutcomeCount
- FROM SPACEXTBL
- WHERE Date BETWEEN '2010-06-04' AND '2017-03-20'
- GROUP BY Landing_Outcome
- ORDER BY OutcomeCount DESC;
- Present your query result with a short explanation here

Landing_Outcome	OutcomeCount
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

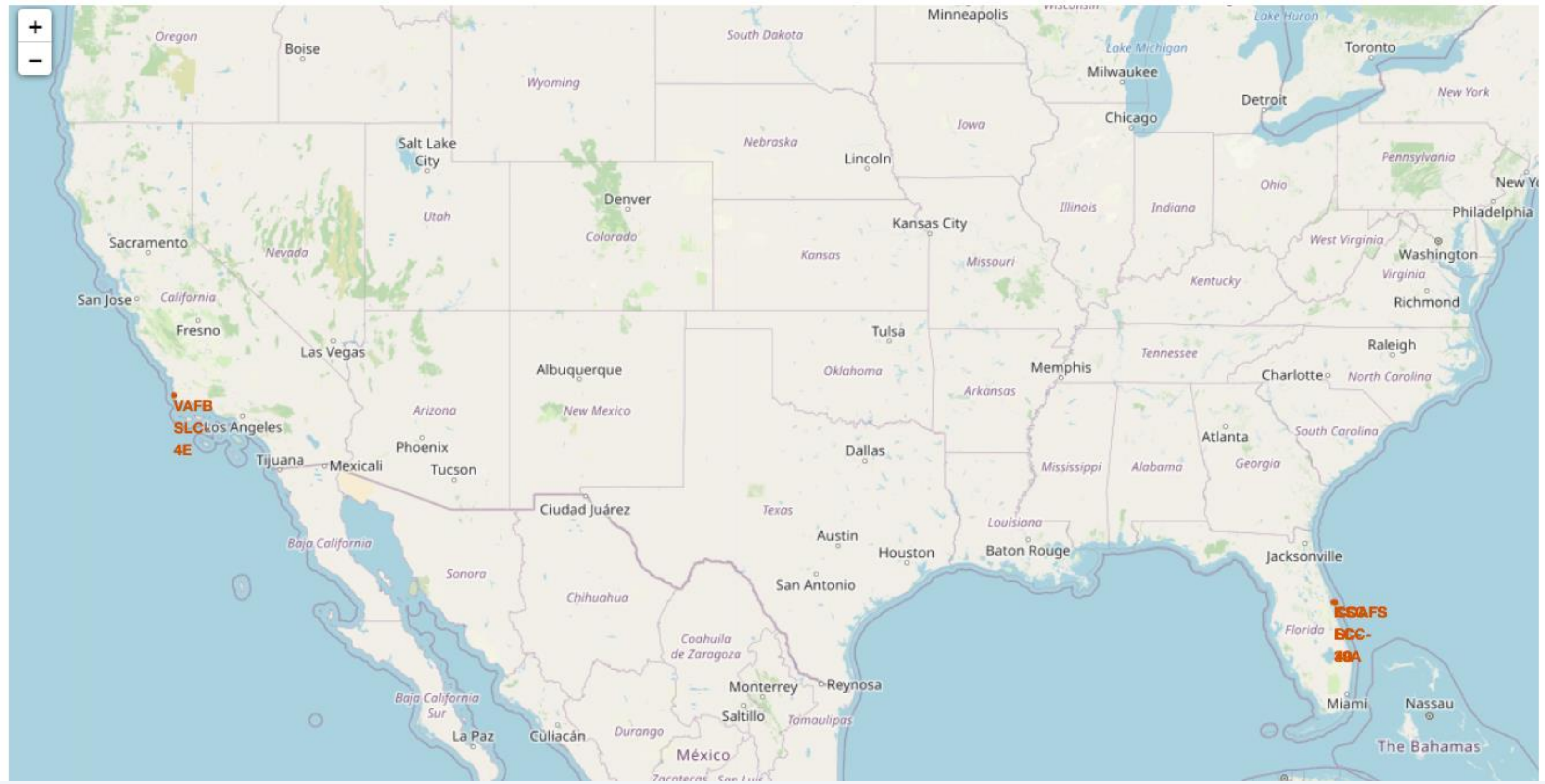
The results indicate the diversity of landing outcomes during this period, with 'No attempt' being the most frequent (10 occurrences), followed by equal counts of 'Success (drone ship)' and 'Failure (drone ship)' (5 each), 'Success (ground pad)' and 'Controlled (ocean)' (3 each), 'Uncontrolled (ocean)' and 'Failure (parachute)' (2 each), and one instance of 'Precluded (drone ship)'.

Section 3

Launch Sites Proximities Analysis

<Folium Map Screenshot 1>

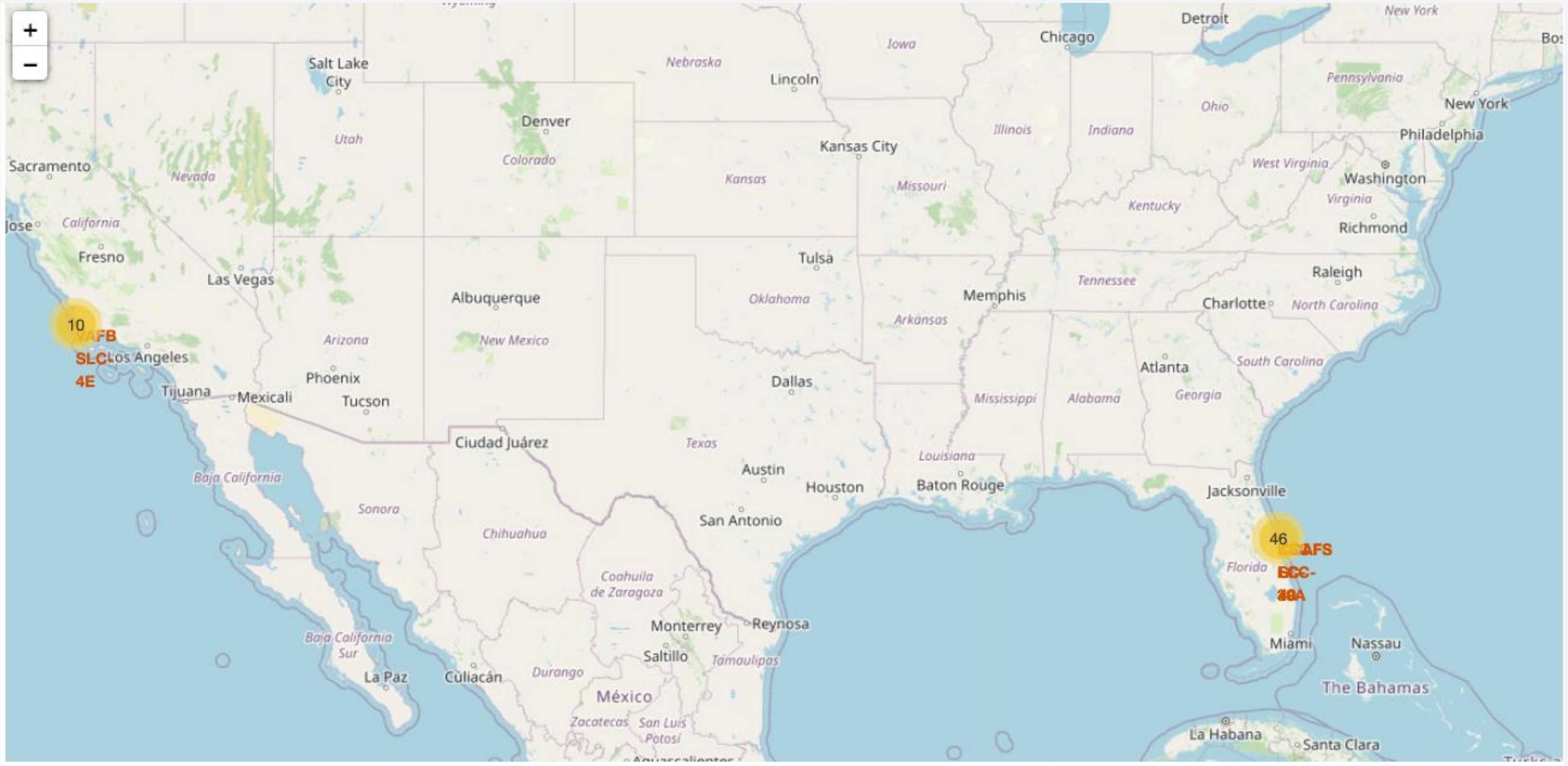
Explore the generated folium map and make a proper screenshot to include all launch sites' location markers on a global map



The map shows markers indicating launch sites on the west coast near Los Angeles (Vandenberg Air Force Base, SLC-4E) and on the east coast in Florida (Cape Canaveral Air Force Station, CCAFS), which are important for their role. As primary launch sites for space missions.

<Folium Map Screenshot 2>

Explore the folium map and make a proper screenshot to show the color-labeled launch outcomes on the map



An important feature of this map is the numerical annotations next to the launch site markers, which represent the number of launches conducted at each site.

<Folium Map Screenshot 3>

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

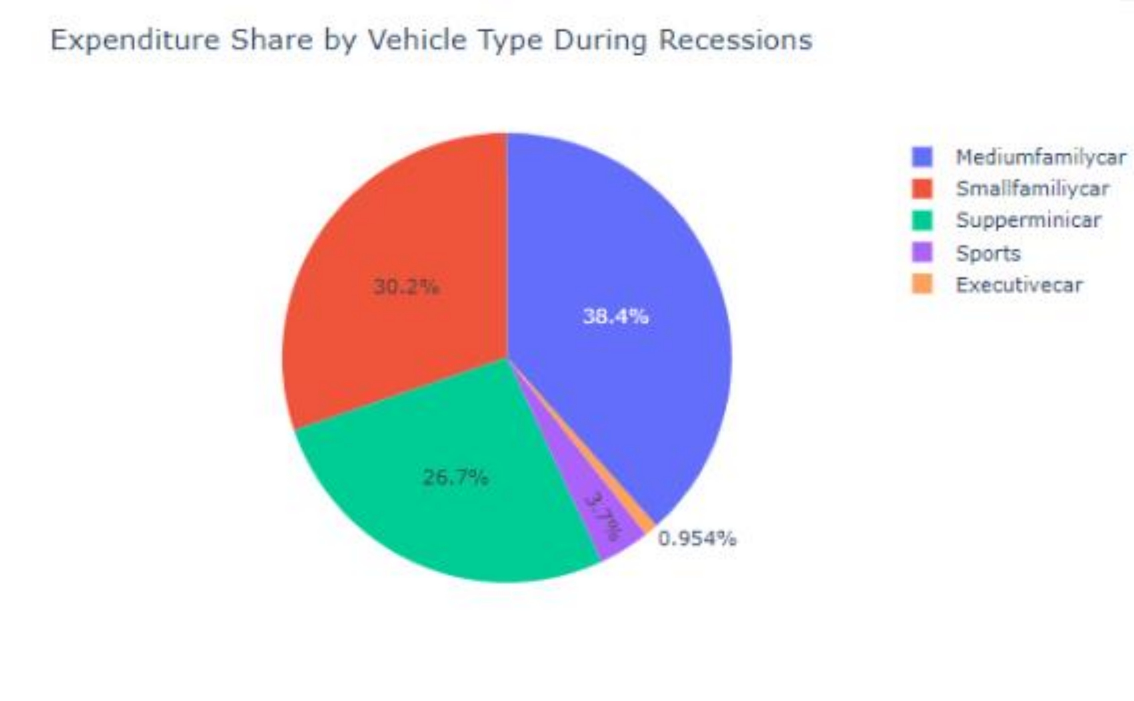
On this map we see circles with numbers indicating specific launch pads at Cape Canaveral Air Force Station (CCAFS).

Section 4

Build a Dashboard with Plotly Dash

<Vehicle Type During Recessions>

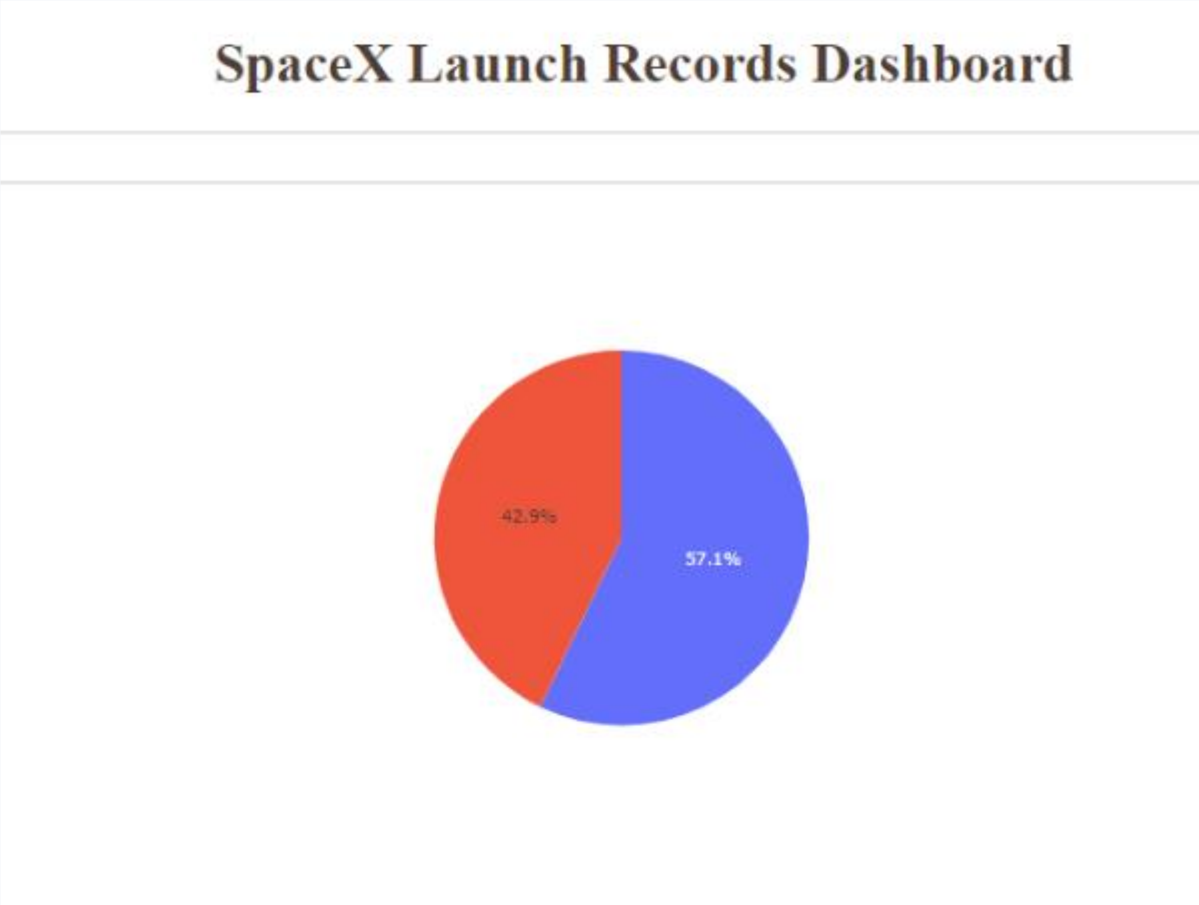
Show the screenshot of launch success count for all sites, in a piechart



This pie chart shows the distribution of auto spending shares by vehicle type during economic downturns. The largest segment belongs to “Mediumfamilycar” at 38.4%, indicating that during recessions, consumers spend more on medium family cars than on other vehicle types.

<SpaceX Lunch Records Dashboard>

- Replace <Dashboard screenshot 2> title with an appropriate title



The circle chart provides an overview of SpaceX's launch performance, with the blue section indicating that 57.1% of operations were successful, and the red section indicating that 42.9% were not successful.

<Correlation between Payload and Succes >

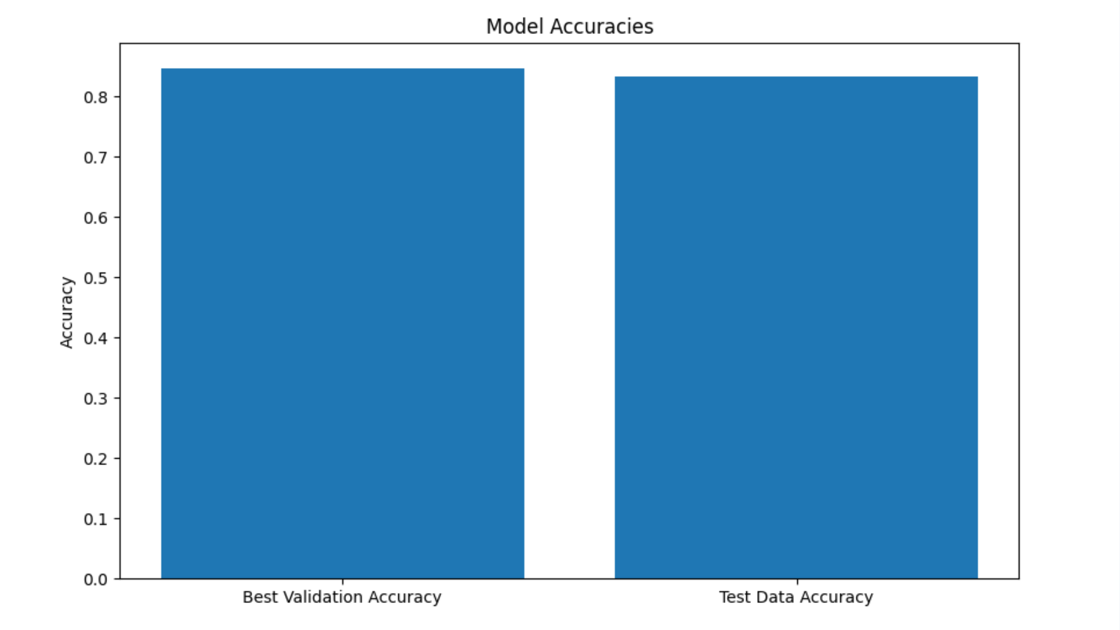
The graph shows Correlation between Payload and Successes for all sites.

Section 5

Predictive Analysis (Classification)

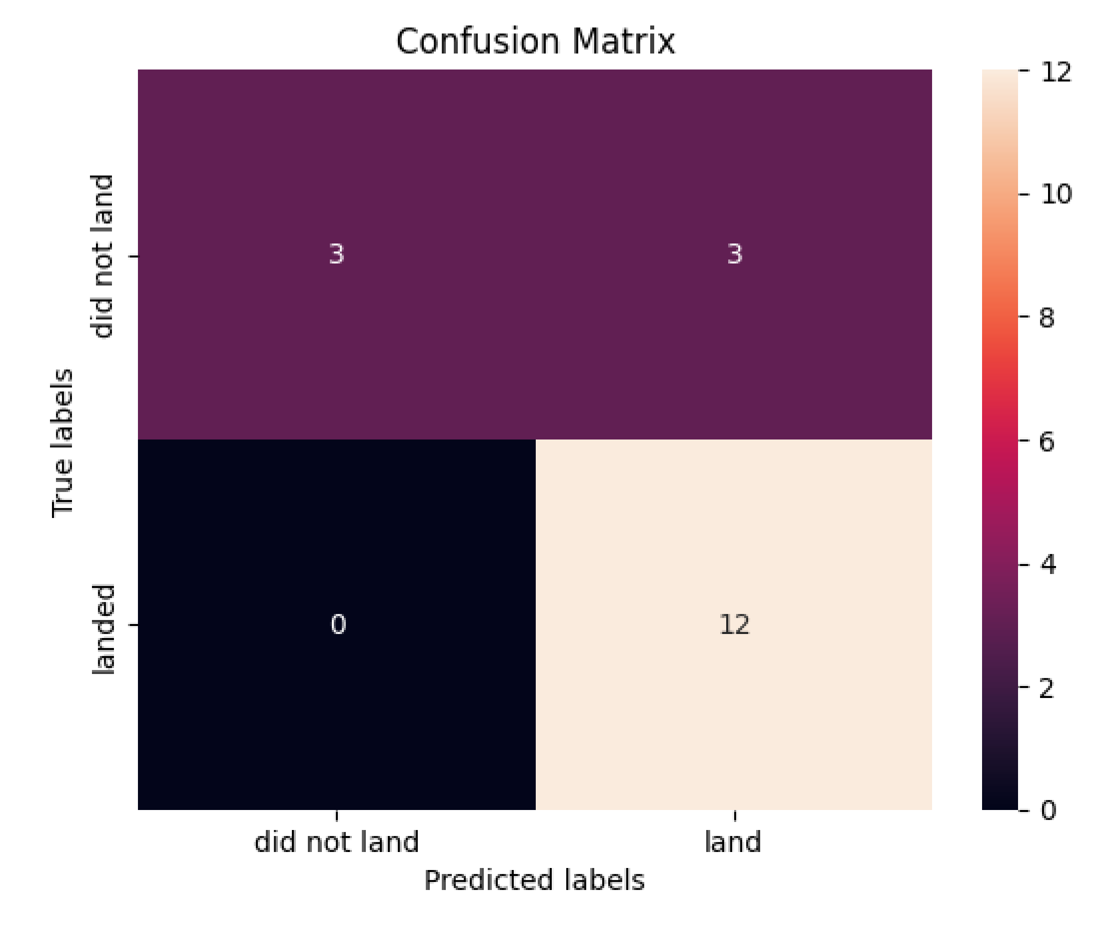
Classification Accuracy

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



The bar chart depicts the accuracy of the logistic regression model optimized using GridSearchCV. “Best cross-validation accuracy” is the highest accuracy obtained from the cross-validation process during parameter tuning, which is about 0.846.

Confusion Matrix



The confusion matrix represents the model's predictions for “landed” and “did not land” cases.

Conclusions

Data collection methodology :

- Real-time data retrieval through the use of SpaceX REST API.

Data processing methodology :

- Normalizing data using data structuring techniques such as JSON.
- Create consistent and coherent data sets by addressing missing values.

Exploratory Data Analysis (EDA) Methodology :

- Conduct an initial careful examination to identify trends and patterns.
- Providing features of descriptive statistics and visual inspection techniques

Pandas and matplotlib libraries:

- Focus on data visualization and statistical analysis through the use of Python libraries for exploratory analysis.

Interactive Dashboard Methodology:

- Providing interactive tools to design dynamic visual analyses.

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
- %%sql
- SELECT DISTINCT Launch_Site
- FROM SPACEXTBL;

- # Create a data from launch_dict
- data = pd.DataFrame(launch_dict)

- # Show the head of the dataframe
- data.info()
- print(data.head())

Thank you!