IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Name : Timothee Nkwar
Date : 30 -April-2025

Outline

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

Executive Summary

- Summary of methodologies
  - Data collection
  - Data wragling
  - EDA with python
  - EDA wit SQL
- Summary of all results
  - EDA result
  - Prediction result

Introduction

- Project background and context
  - SpaceX, a leading private aerospace company, has conducted hundreds of rocket launches aimed at revolutionizing space transportation. Understanding the factors that influence the success of these launches is essential for improving mission reliability and operational planning.
- Problems you want to find answers
  - Analyze and model SpaceX launch data to understand trends and predict mission success.

Section 1

Methodology

Methodology

Executive Summary

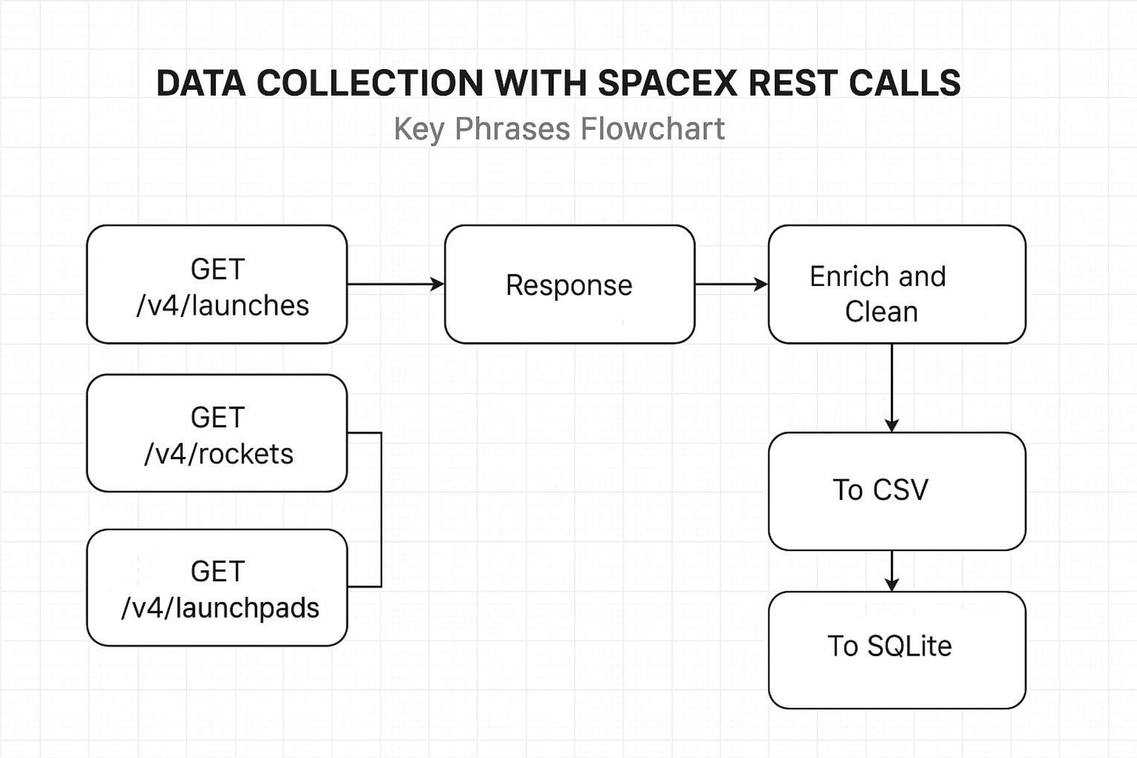
- Data collection methodology:
  - Collected data from SpaceX API (https://api.spacexdata.com/v4/launches).
  - Enriched with rocket and launchpad details using /rockets and /launchpads endpoints.
- Perform data wrangling
  - Cleaned data: Selected columns (name, date_utc, success, rocket, launchpad), handled missing values, added year and month.
  - Stored in CSV and SQLite for analysis.
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Use hyperparameter tuning (e.g., Grid Search).

Data Collection

- Describe how data sets were collected.

The datasets were collected from SpaceX's public APIs, which provided information about launches, rockets, and launchpads. Here's the summary:

- Launch Data: Fetched from the SpaceX Launches API and converted into a pandas DataFrame.
- Rocket Data: Retrieved from the SpaceX Rockets API and stored in a DataFrame.
- Launchpad Data: Collected from the SpaceX Launchpads API and stored in a separate DataFrame.
- Data Integration: The launch data was enriched by mapping rocket and launchpad IDs to their names. Additional columns (year, month) were created, and missing values were filled.
- Storage: The cleaned data was saved to a CSV file and an SQLite database for easy access and analysis.
- This process ensured that the data was organized and ready for analysis.



Data Collection – SpaceX API

- Present your data collection with SpaceX REST calls using key phrases and flowcharts

- https://github.com/TimotheeNkwar/SpaceX_Launch_Analysis/blob/main/SpaceX_Launch_Analysis/notebooks/data_collection.ipynb

```python
import requests
import pandas as pd
import sqlite3
import os
# Fetch launches
url = "https://api.spacexdata.com/v4/launches"
response = requests.get(url)
launches = response.json()
df = pd.DataFrame(launches)

# Clean column names
df.columns = df.columns.str.strip()

# Fetch rockets and launchpads for mapping
rockets = pd.DataFrame(requests.get("https://api.spacexdata.com/v4/rockets").json())
launchpads = pd.DataFrame(requests.get("https://api.spacexdata.com/v4/launchpads").json())
rocket_map = dict(zip(rockets["id"], rockets["name"]))
launchpad_map = dict(zip(launchpads["id"], launchpads["name"]))
launchpad_loc = dict(zip(launchpads["id"], launchpads[["latitude", "longitude"]].values.tolist()))

# Select and clean data
df = df[["name", "date_utc", "success", "rocket", "launchpad"]]
df["rocket_name"] = df["rocket"].map(rocket_map)
df["launchpad_name"] = df["launchpad"].map(launchpad_map)
df["year"] = pd.to_datetime(df["date_utc"]).dt.year
df["month"] = pd.to_datetime(df["date_utc"]).dt.month
# Avoid future warning
df["success"] = df["success"].fillna(False)
df = df.infer_objects(copy=False)
# Save to CSV and SQLite
os.makedirs("data", exist_ok=True)
df.to_csv("data/spacex_cleaned.csv", index=False)
conn = sqlite3.connect("data/spacex.db")
df.to_sql("launches", conn, if_exists="replace", index=False)
conn.close()
```

Data Collection - Scraping

- Present your web scraping process using key phrases and flowcharts

- https://github.com/TimotheeNkwar/SpaceX_Launch_Analysis/blob/main/SpaceX_Launch_Analysis/notebooks/Data_collection_scraping.ipynb

```
# Import libraries
import requests
import pandas as pd
import sqlite3
from datetime import datetime

# Fetch launches
print("Fetching launch data...")
url = "https://api.spacexdata.com/v4/launches"
try:
    response = requests.get(url)
    response.raise_for_status()  # Check for HTTP errors
    launches = response.json()
    df = pd.DataFrame(launches)
    print(f"Retrieved {len(df)} launches")
except requests.RequestException as e:
    print(f"Error fetching launches: {e}")
    df = pd.DataFrame()

# Fetch rockets
print("Fetching rocket data...")
try:
    rockets = pd.DataFrame(requests.get("https://api.spacexdata.com/v4/rockets").json())
    rocket_map = dict(zip(rockets["id"], rockets["name"]))
except requests.RequestException as e:
    print(f"Error fetching rockets: {e}")
    rocket_map = {}

# Fetch launchpads
print("Fetching launchpad data...")
try:
    launchpads = pd.DataFrame(requests.get("https://api.spacexdata.com/v4/launchpads").json())
    launchpad_map = dict(zip(launchpads["id"], launchpads["name"]))
    launchpad_loc = dict(zip(launchpads["id"], launchpads[["latitude", "longitude"]].values.tolist()))
except requests.RequestException as e:
    print(f"Error fetching launchpads: {e}")
```

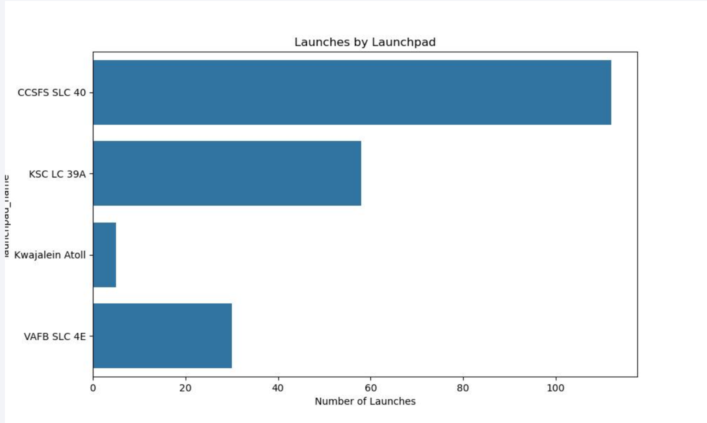
Data Wrangling

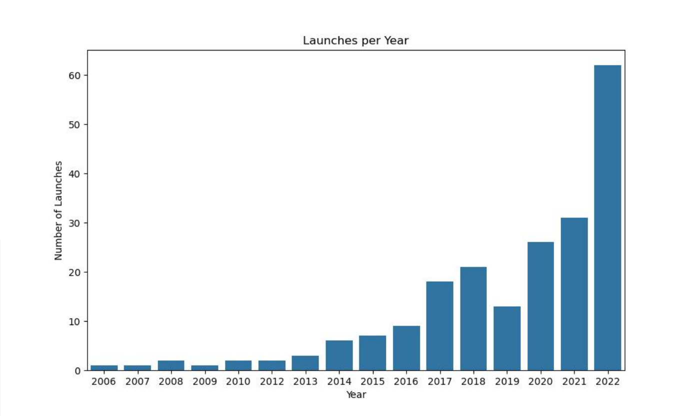
- Describe how data were processed
- You need to present your data wrangling process using key phrases and flowcharts

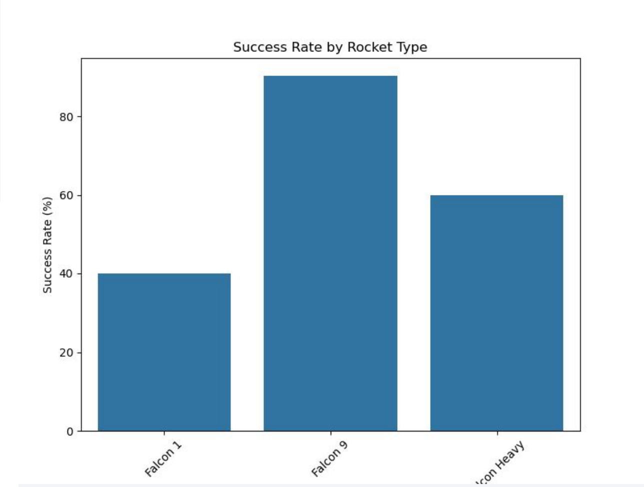
https://github.com/TimotheeNkwar/SpaceX_Launch_Analysis/blob/main/SpaceX_Launch_Analysis/notebooks/eda_visualization.ipynb

EDA with Data Visualization

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



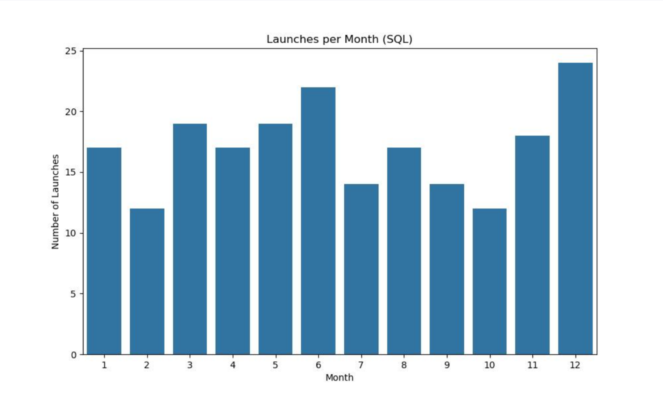


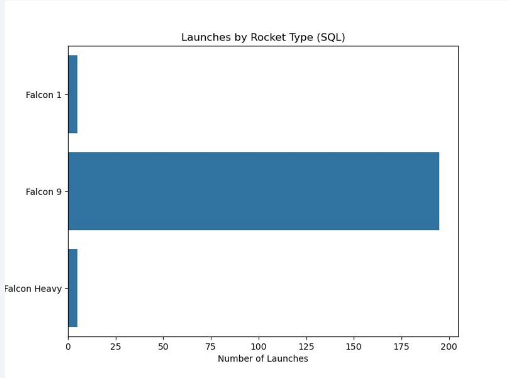


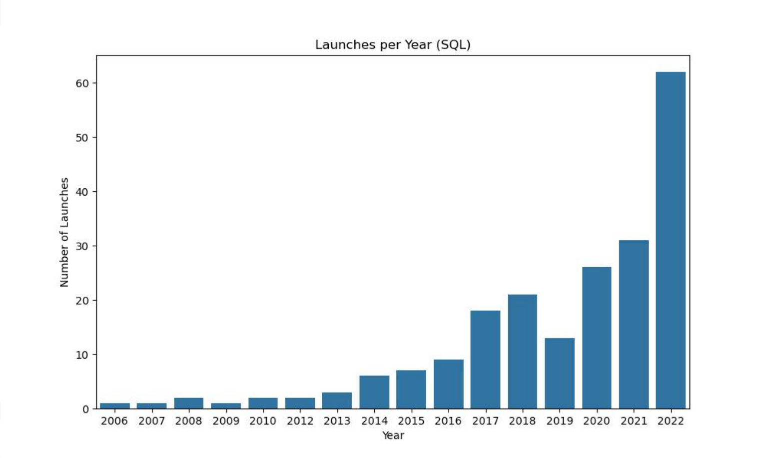
https://github.com/TimotheeNkwar/SpaceX_Launch_Analysis/tree/main/SpaceX_Launch_Analysis/notebooks/presentation

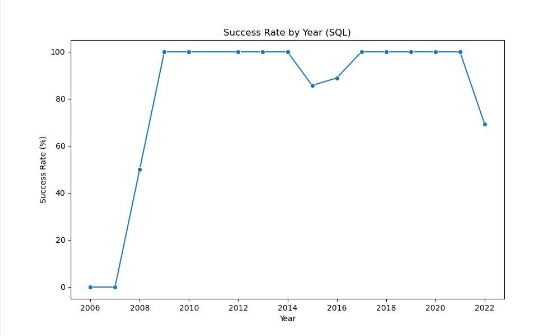
EDA with SQL

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





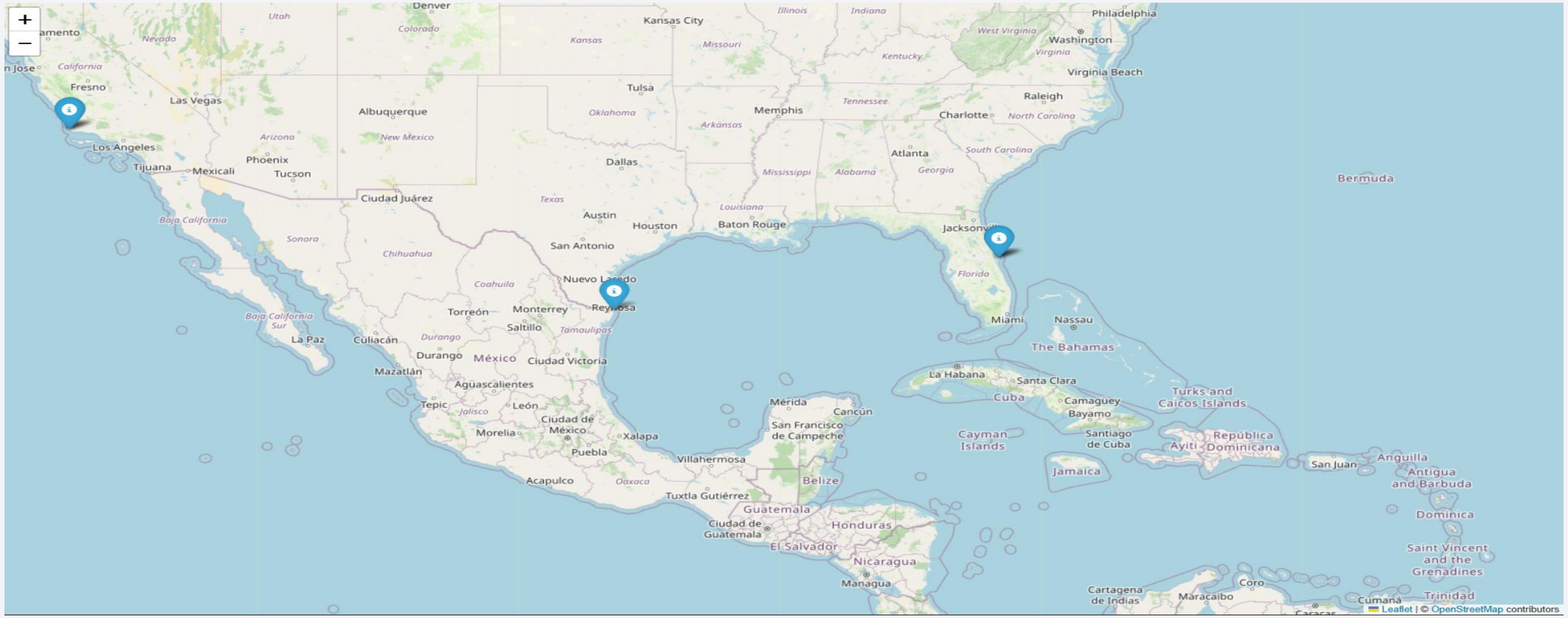




- Lauches by Month
  - Higher launches in Q2/Q3
- Lauches By rocket Type
  - Falcon 9 dominates
- Success rates by years
  - ~95% success rate from 2008 to 2021
- Lauches by Years
  - Steady increase in launches.

Build an Interactive Map with Folium

- Summarize what map objects such as markers, circles, lines, etc. you created and added to a folium map



- https://github.com/TimotheeNkwar/SpaceX_Launch_Analysis/blob/main/SpaceX_Launch_Analysis/notebooks/presentation/launch_map.html

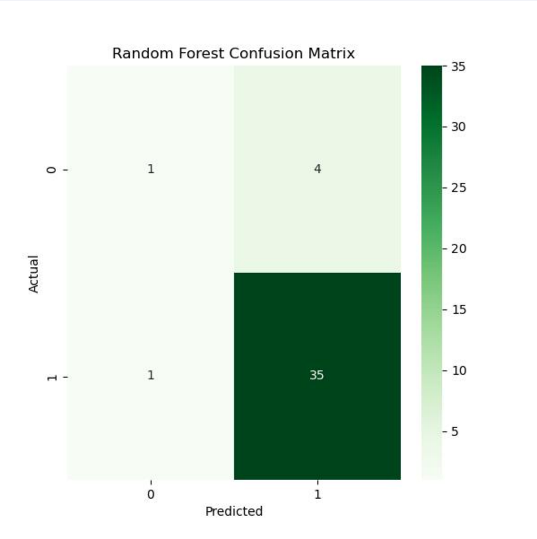
Build a Dashboard with Plotly Dash

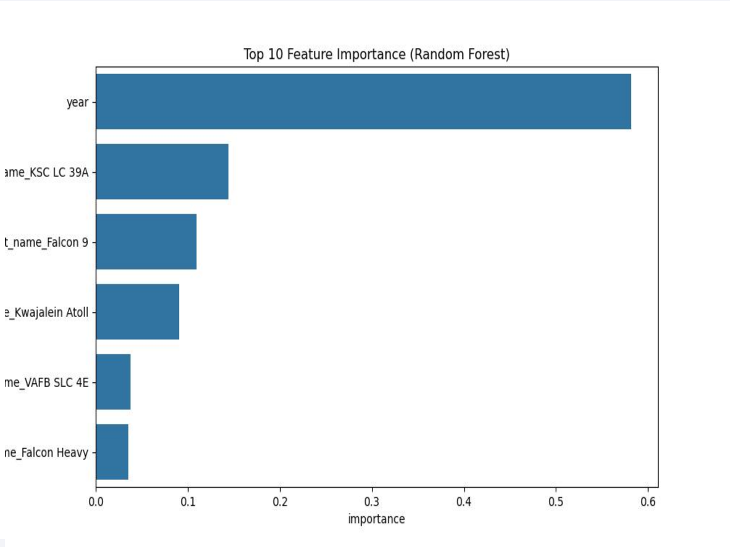
- Summarize what plots/graphs and interactions you have added to a dashboard

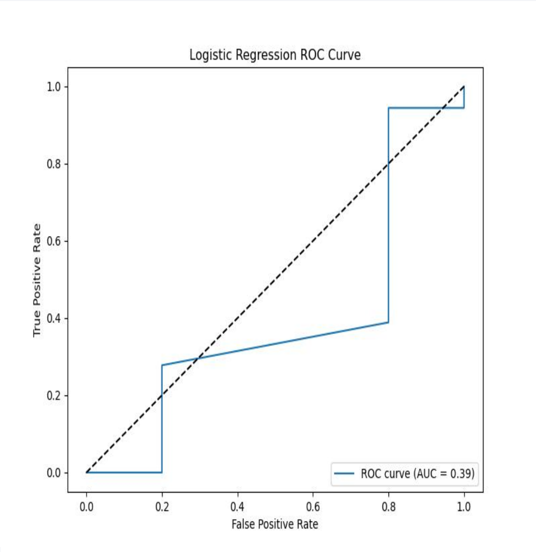
- https://github.com/TimotheeNkwar/SpaceX_Launch_Analysis/blob/main/SpaceX_Launch_Analysis/notebooks/presentation/interactive_scatter.html

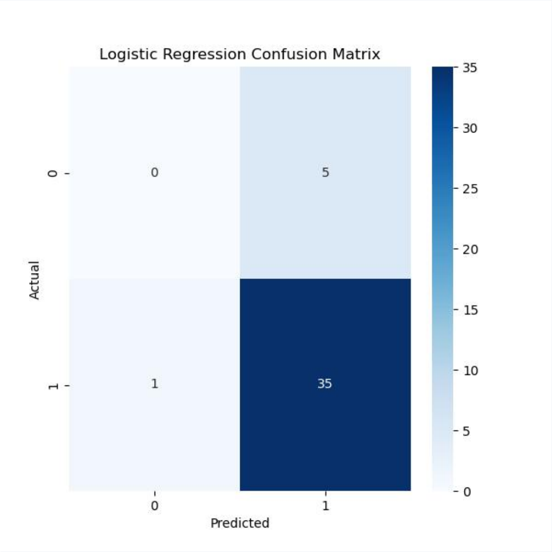
Predictive Analysis (Classification)

- Summarize how you built, evaluated, improved, and found the best performing classification model









- https://github.com/TimotheeNkwar/SpaceX_Launch_Analysis/blob/main/SpaceX_Launch_Analysis/notebooks/predictive_analysis.ipynb

Results

- **Exploratory data analysis results**

- **The logistic regression** model achieved a solid accuracy of 85%, indicating good overall performance. However, the macro average scores (precision: 44%, recall: 49%) suggest that performance is imbalanced across classes—likely due to class imbalance. Despite this, the weighted average (81%) shows the model performs well on the majority class. Improvements could focus on better handling the minority class.

```
Logistic Regression Classification Report:
...
    accuracy                        0.85      41
   macro avg      0.44      0.49    0.46      41
weighted avg      0.77      0.85    0.81      41
```

- **The Random Forest** model achieved a strong accuracy of 88%, with excellent performance on the majority class (True), showing precision of 90% and recall of 97%. However, its performance on the minority class (False) was weak, with only 20% recall, indicating it struggles to correctly identify failed launches. While the macro average f1-score (61%) highlights this imbalance, the weighted average (85%) confirms strong overall performance. Enhancing minority class detection could further improve the model.

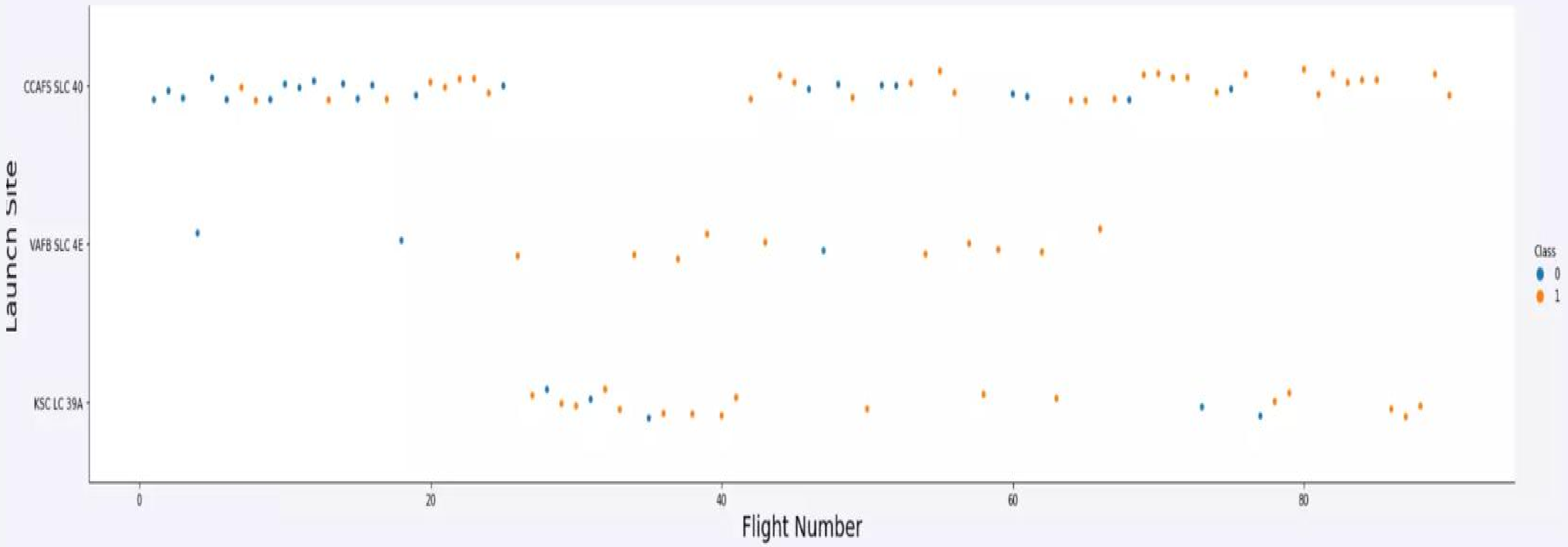
```
Random Forest Classification Report:
              precision    recall  f1-score   support

       False       0.50      0.20      0.29         5
        True       0.90      0.97      0.93        36

    accuracy                           0.88        41
   macro avg       0.70      0.59      0.61        41
weighted avg       0.85      0.88      0.85        41
```

Section 2

Insights drawn from EDA

Flight Number vs. Launch Site

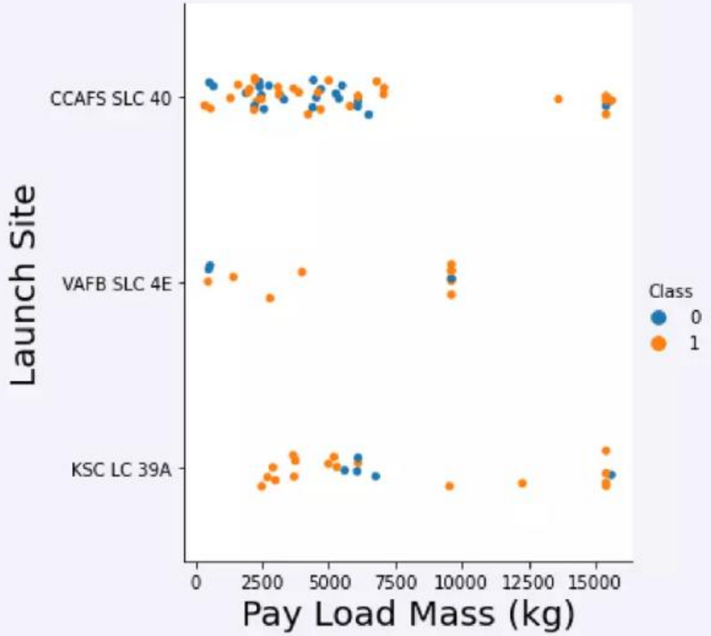
- Show a scatter plot of Flight Number vs. Launch Site



- Show the screenshot of the scatter plot with explanations

Payload vs. Launch Site

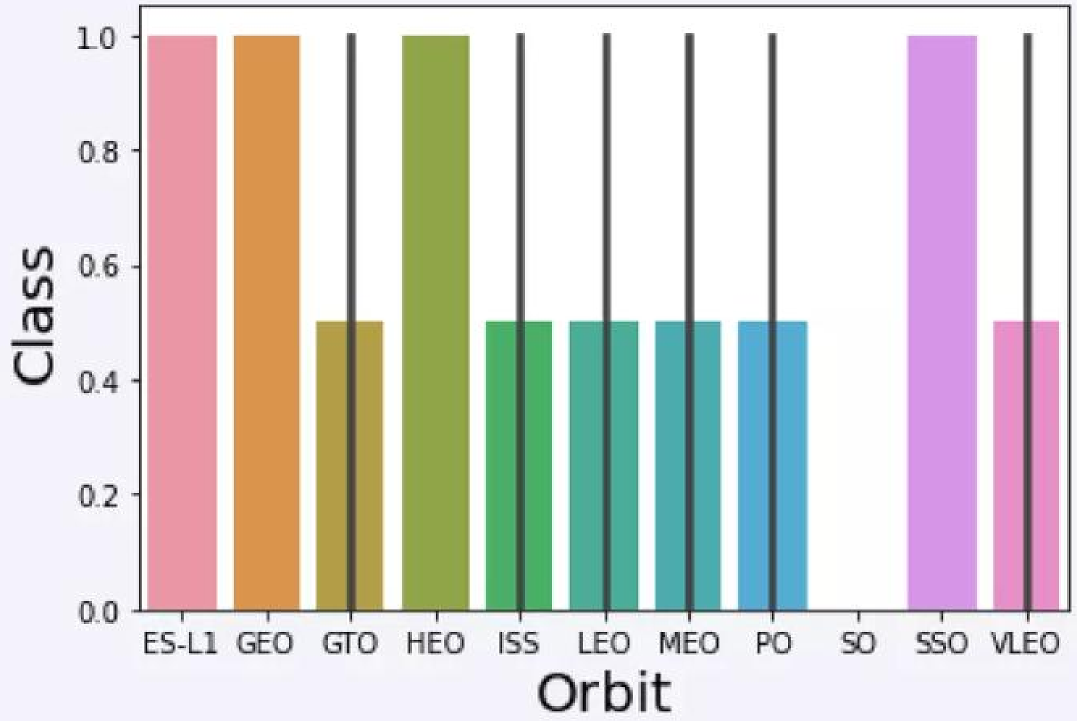
- Show a scatter plot of Payload vs. Launch Site



- 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

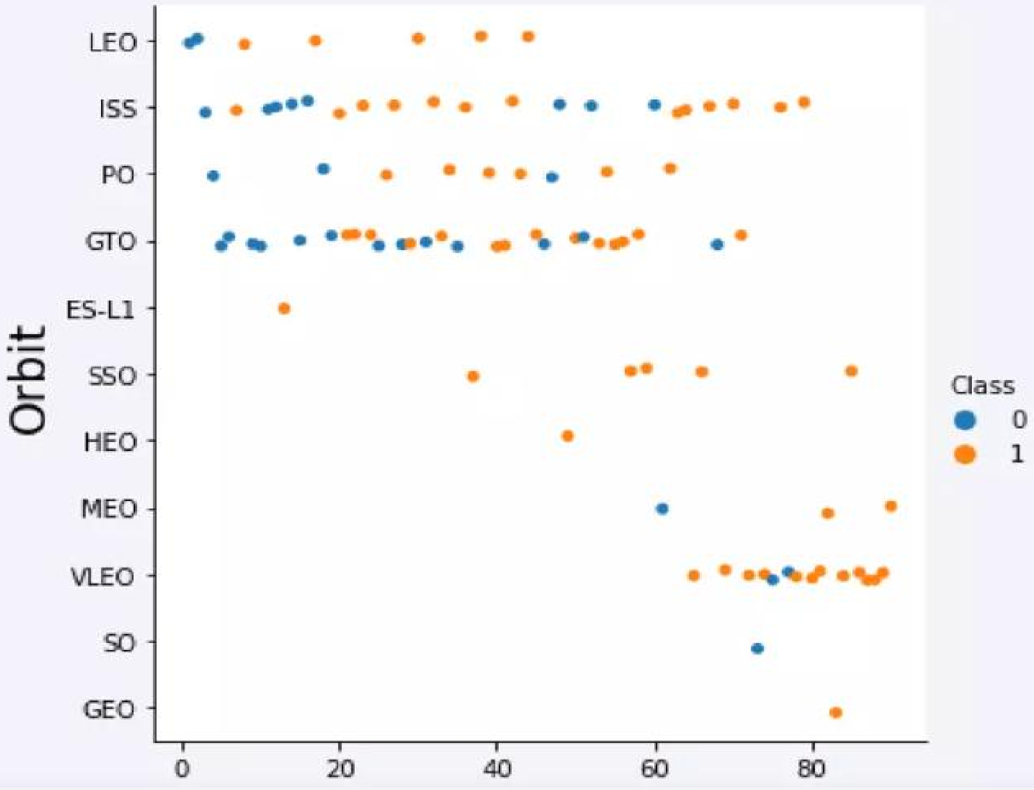


- Show the screenshot of the scatter plot with explanations

Flight Number vs. Orbit Type

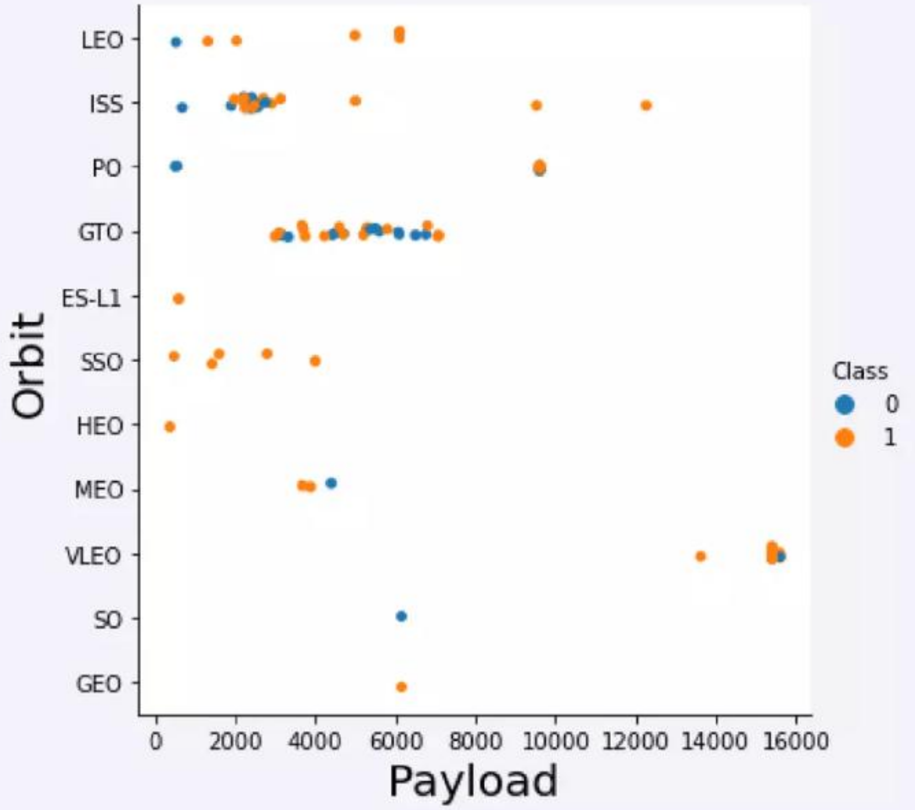
- Show a scatter point of Flight number vs. Orbit type

- Show the screenshot of the scatter plot with explanations



Payload vs. Orbit Type

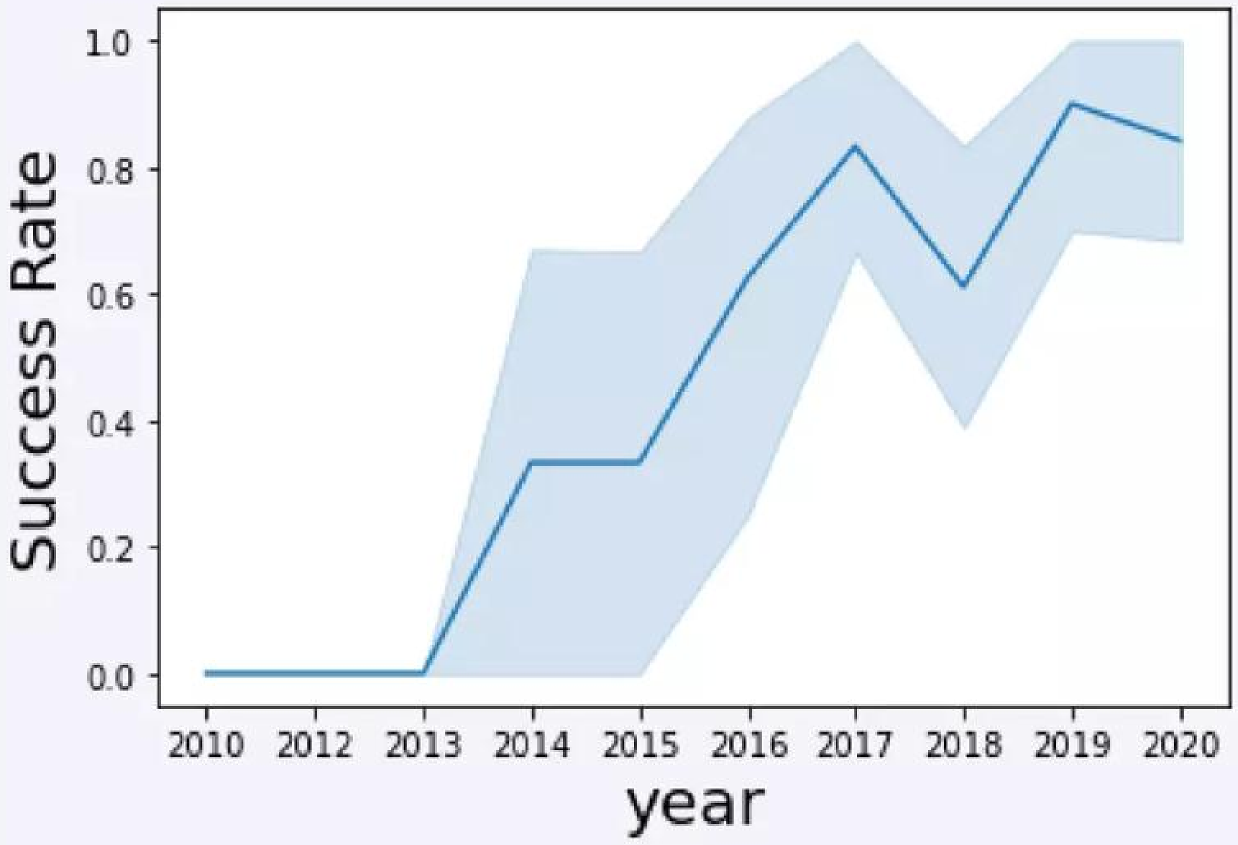
- Show a scatter point of payload vs. orbit type



- 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

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

- Present your query result with a short explanation here

Launch Site Names Begin with 'CCA'

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

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	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
2012-10-08	00:35:00	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

- 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

2928.400000

- 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

2015-12-22

- 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

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

- 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

100

- 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

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

- 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

time_utc_	booster_version	launch_site	payload	payload_mass_kg_	orbit	customer	mission_outcome	landing_outcome
14:39:00	F9 FT B1031.1	KSC LC-39A	SpaceX CRS-10	2490	LEO (ISS)	NASA (CRS)	Success	Success (ground pad)
17:54:00	F9 FT B1029.1	VAFB SLC-4E	Iridium NEXT 1	9600	Polar LEO	Iridium Communications	Success	Success (drone ship)
05:26:00	F9 FT B1026	CCAFS LC-40	JCSAT-16	4600	GTO	SKY Perfect JSAT Group	Success	Success (drone ship)
04:45:00	F9 FT B1025.1	CCAFS LC-40	SpaceX CRS-9	2257	LEO (ISS)	NASA (CRS)	Success	Success (ground pad)
21:39:00	F9 FT B1023.1	CCAFS LC-40	Thaicom 8	3100	GTO	Thaicom	Success	Success (drone ship)
[illegible]	[illegible]	CCAFS LC-	[illegible]	[illegible]	[illegible]	SKY Perfect JSAT	[illegible]	[illegible]

- 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

2016-05-27	21:39:00	F9 FT B1023.1	CCAFS LC-40	Thaicom 8	3100	GTO	Thaicom	Success	Success (drone ship)
2016-05-06	05:21:00	F9 FT B1022	CCAFS LC-40	JCSAT-14	4696	GTO	SKY Perfect JSAT Group	Success	Success (drone ship)
2016-04-08	20:43:00	F9 FT B1021.1	CCAFS LC-40	SpaceX CRS-8	3136	LEO (ISS)	NASA (CRS)	Success	Success (drone ship)
2015-12-22	01:29:00	F9 FT B1019	CCAFS LC-40	OG2 Mission 2 11 Orbcomm-OG2 satellites	2034	LEO	Orbcomm	Success	Success (ground pad)

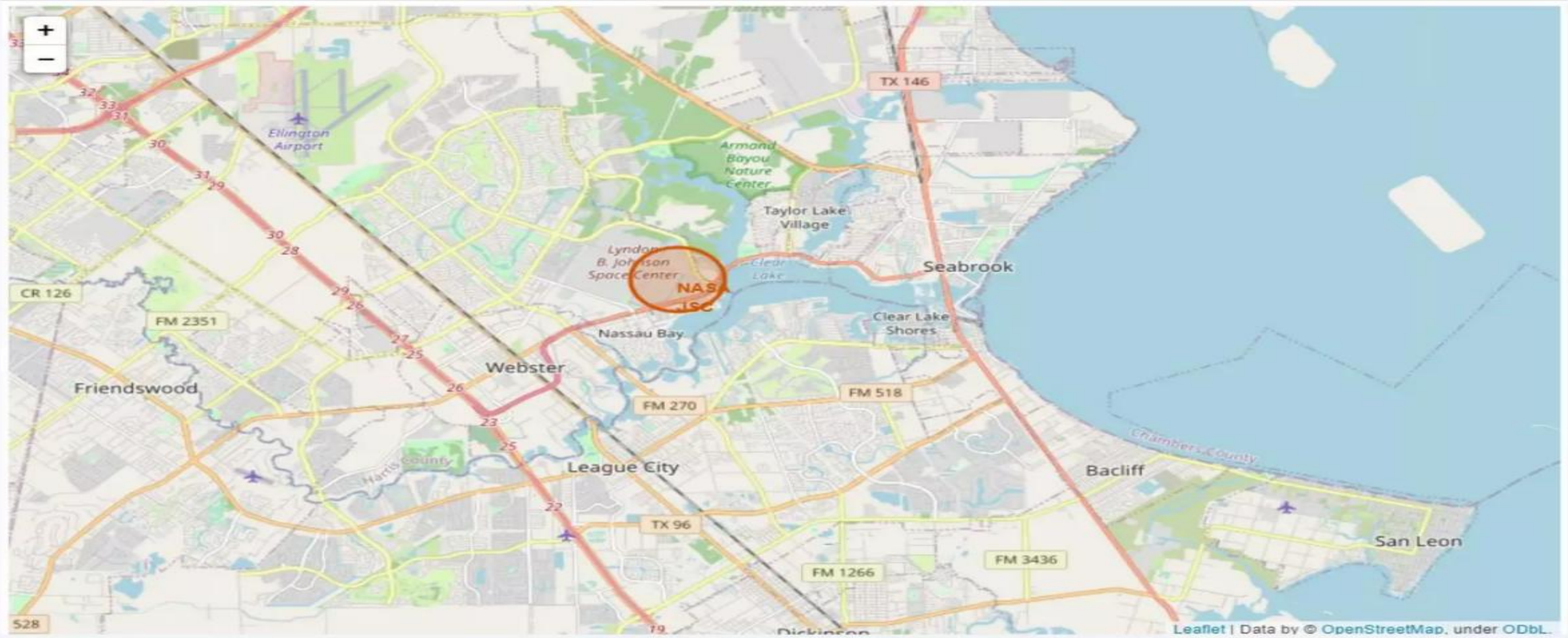
- Present your query result with a short explanation here

Section 3

Launch Sites Proximities Analysis

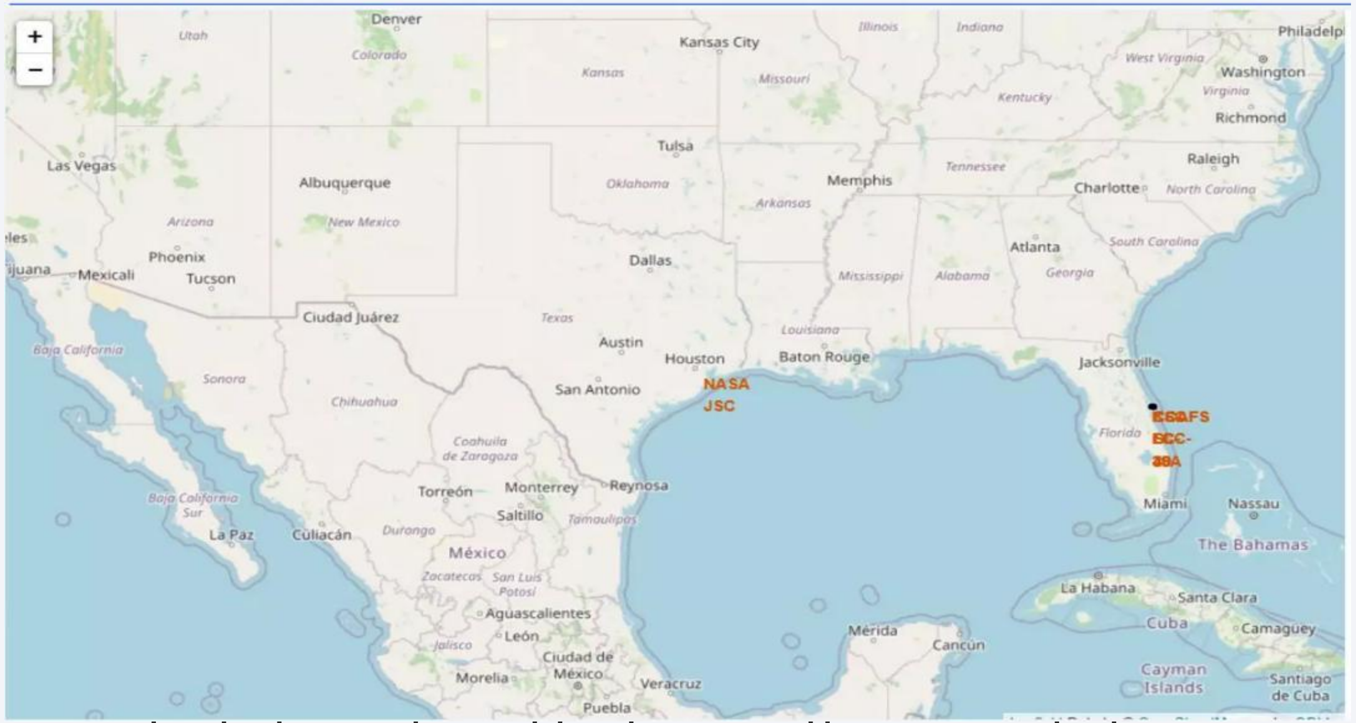
All lauchs sites

- Replace <Folium map screenshot 1> title with an appropriate title



Fail Sites

- Replace <Folium map screenshot 2> title with an appropriate title



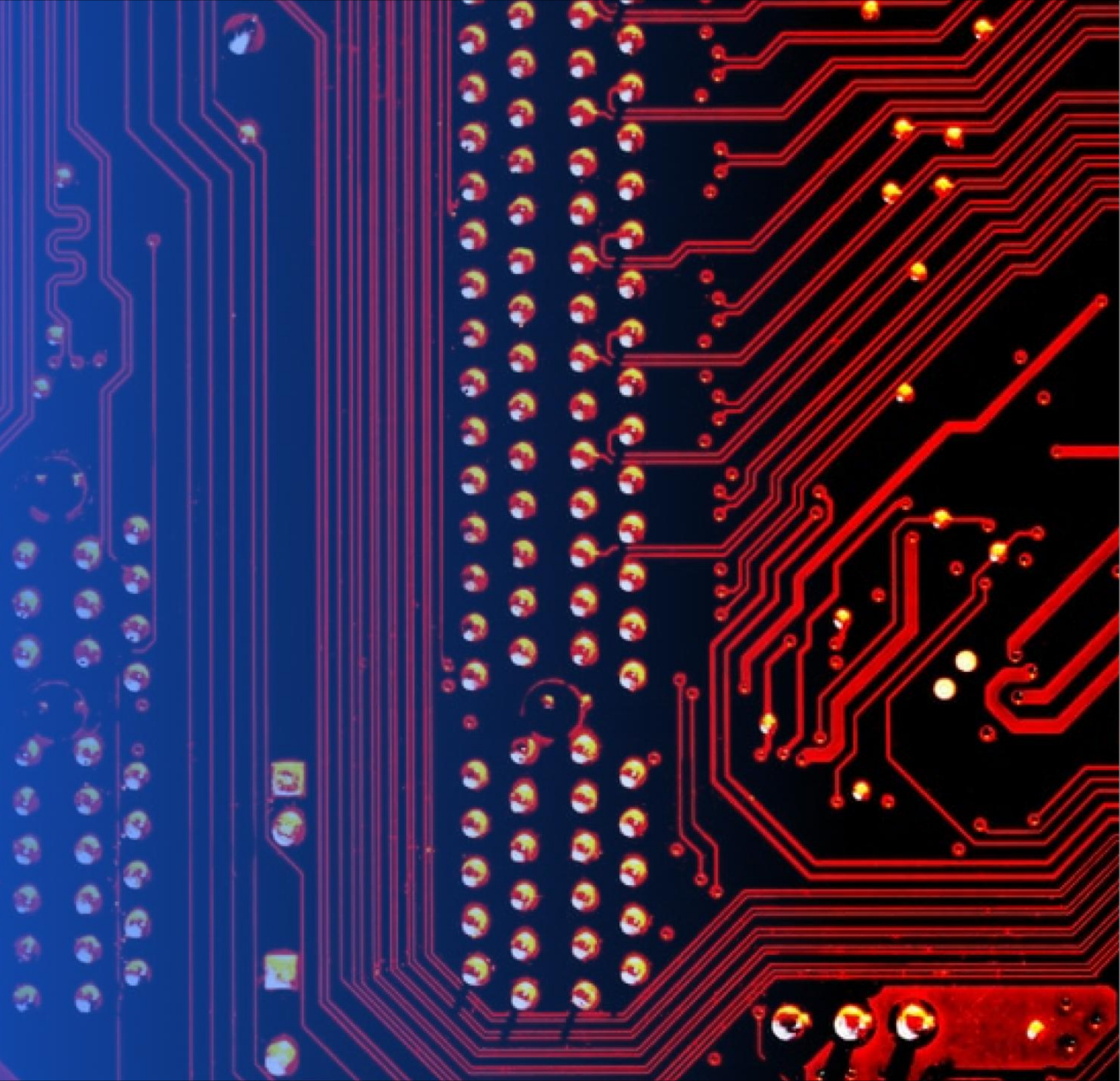
- Explain the important elements and findings on the screenshot

<Folium Map Screenshot 3>

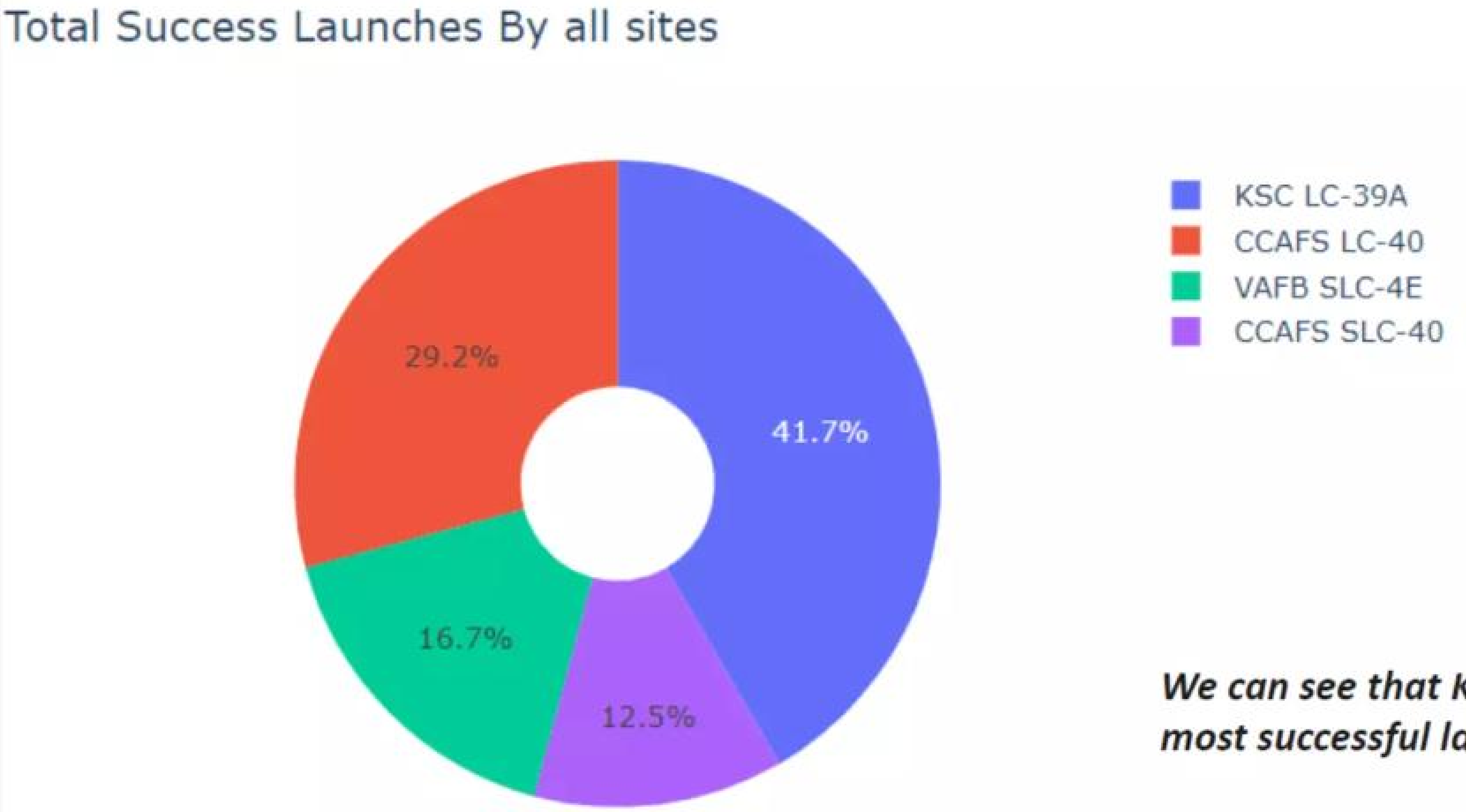
- Replace <Folium map screenshot 3> title with an appropriate title
- Explore the generated folium map and show the screenshot of a selected launch site to its proximities such as railway, highway, coastline, with distance calculated and displayed
- Explain the important elements and findings on the screenshot

Section 4

Build a Dashboard with Plotly Dash



launch success count for all sites, in a piechart



We can see that KSC LC-39A had the most successful launches from all the sites

Section 5

Predictive Analysis (Classification)

Classification Accuracy

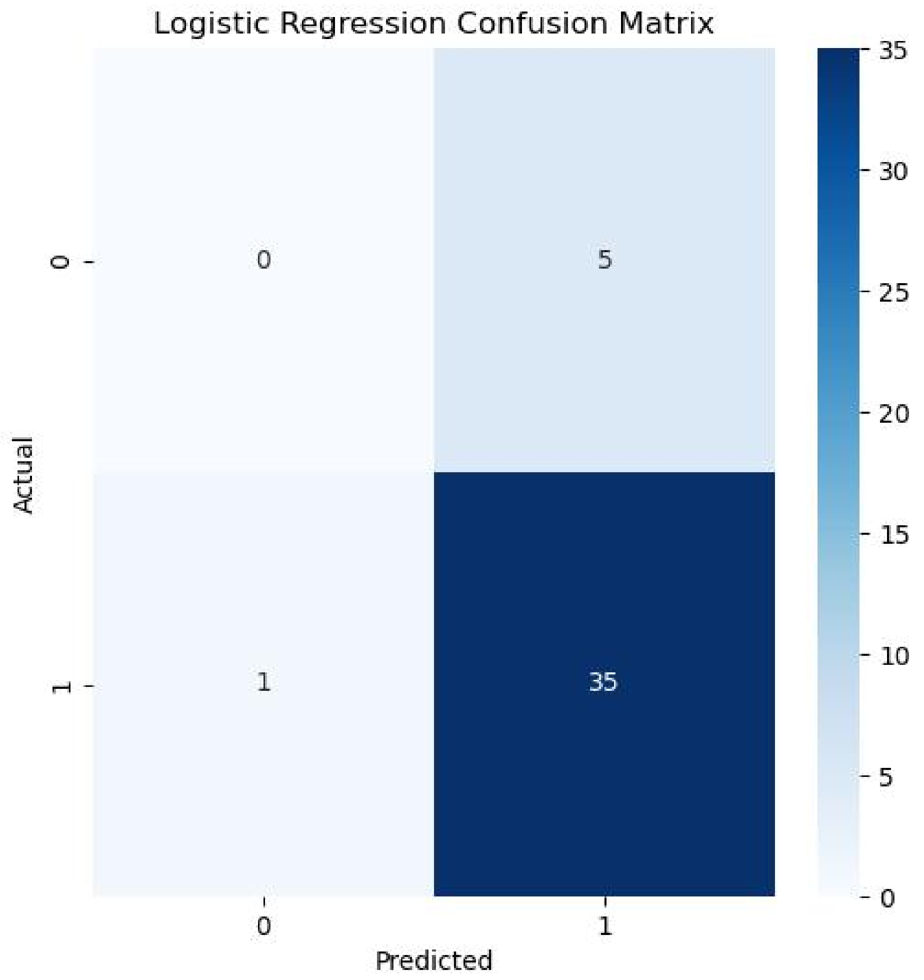
Random Forest Classification Report:

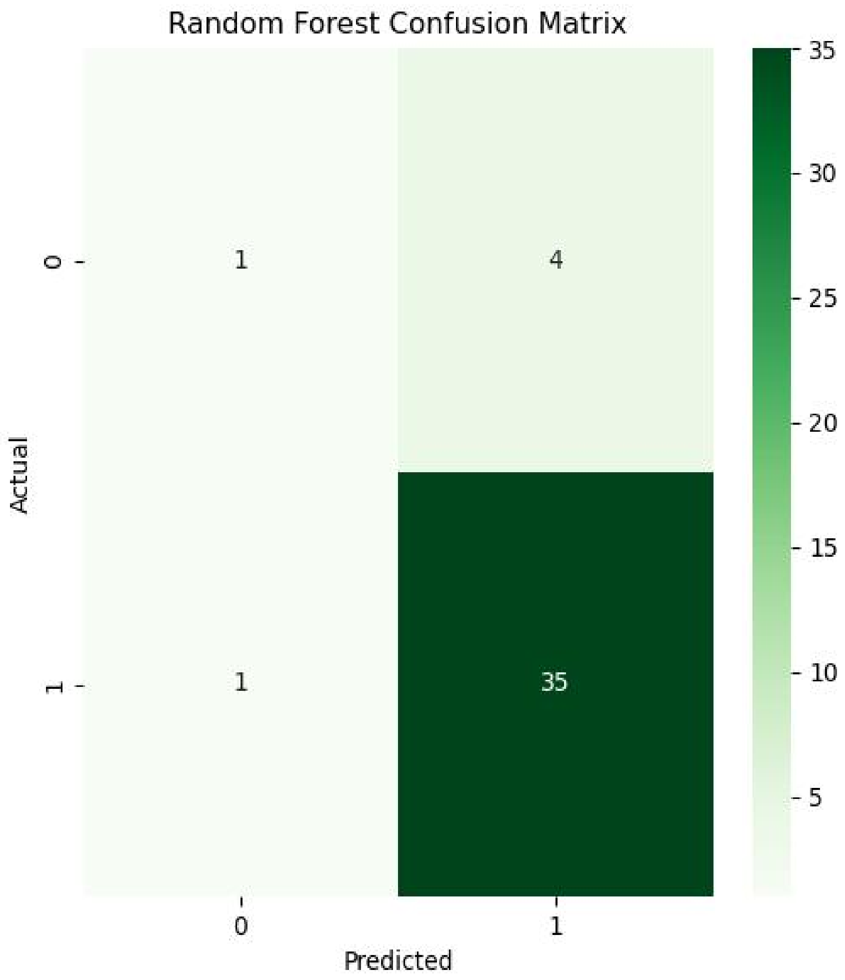
	precision	recall	f1-score	support
False	0.50	0.20	0.29	5
True	0.90	0.97	0.93	36
accuracy			0.88	41
macro avg	0.70	0.59	0.61	41
weighted avg	0.85	0.88	0.85	41

Logistic Regression (Balanced) Classification Report:

	precision	recall	f1-score	support
False	0.11	0.80	0.20	5
True	0.83	0.14	0.24	36
accuracy			0.22	41
macro avg	0.47	0.47	0.22	41
weighted avg	0.75	0.22	0.23	41

Confusion Matrix





Thank you!