IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Lingzhi Li
August 2025

Outline

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

Executive Summary

- Summary of methodologies
  - Data collection through API and web scraping
  - Data wrangling
  - Exploratory data analysis with SQL and data visualization
  - Interactive visual analytics with folium and plotly dash
  - Machine learning prediction
- Summary of all results
  - Result from exploratory data analysis
  - Screenshots of interactive analytics

Introduction

- Project background and context
    - SpaceX lists the cost of a Falcon 9 rocket launch at 62 million dollars on its website, while other providers charge upwards of 165 million dollars. Much of SpaceX's cost savings comes from reusing the rocket's first stage. So, if we can predict whether the first stage will land successfully, we can estimate the launch cost. This information could be useful for competing companies looking to bid against SpaceX. The goal of this project is to build a machine learning pipeline that can predict the successful landing of the first stage.
- Problems I want to find answers
    - What factors influence whether a rocket lands successfully?
    - How do different features interact to affect the chances of a successful landing?
    - What operating conditions are necessary to support a reliable landing program?

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Data was collected using SpaceX API and webscraping from wikipedia
- Perform data wrangling
  - The collected data was enhanced by generating a landing outcome label, created after summarizing and analyzing the outcome features.
- 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, evaluate classification models

Data Collection

- To build a machine learning model predicting the successful landing of the Falcon 9’s first stage, we collected and enriched data using a combination of APIs and web scraping. This ensured a comprehensive and reliable dataset that includes mission details, launch sites, booster versions, payload mass, landing outcomes, and more.

Data Collection – SpaceX API

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

- https://github.com/LLZSG/Data-science-capstone/blob/main/jupyter-labs-spacex-data-collection-api.ipynb

1
- Request and parse the SpaceX launch data using the GET request

2
- Decode the response content as a Json and turn it into a Pandas dataframe

3
- Normalize data into flat data file such as .csv

Data Collection - Scraping

- Present your web scraping process using key phrases and flowcharts

- https://github.com/LLZSG/Data-science-capstone/blob/main/jupyter-labs-webscraping.ipynb

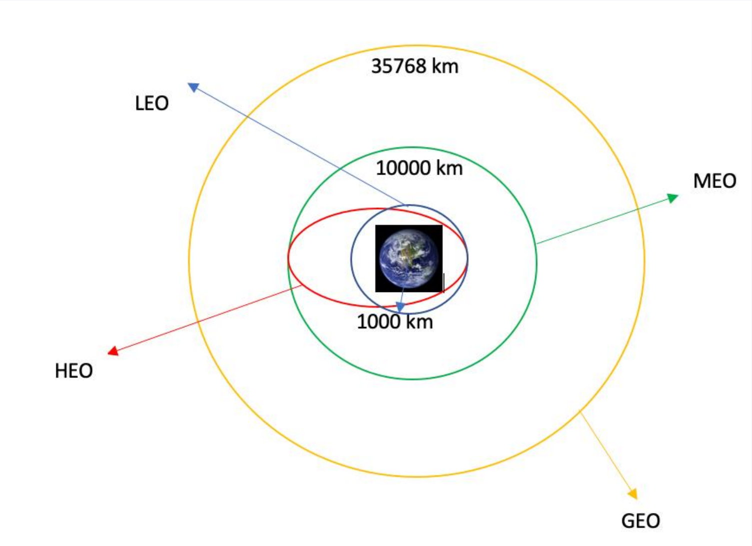
1
- Get HTML response from Wikipedia

2
- Extract data using beautiful soup

3
- Normalize data into flat data file such as .csv

Data Wrangling

- Initial Feature Selection
  - Selected key features: rocket, payloads, launchpad, cores, flight_number, date_utc
- Filtering Data
  - Removed launches with multiple payloads or multiple cores
  - Filtered by launch date
- Flattening Nested Data
  - Extracted single entries from nested lists in cores and payloads fields
- Date Conversion
- Enriching with External Data
- Final Output
- https://github.com/LLZSG/Data-science-capstone/blob/main/labs-jupyter-spacex-Data%20wrangling.ipynb

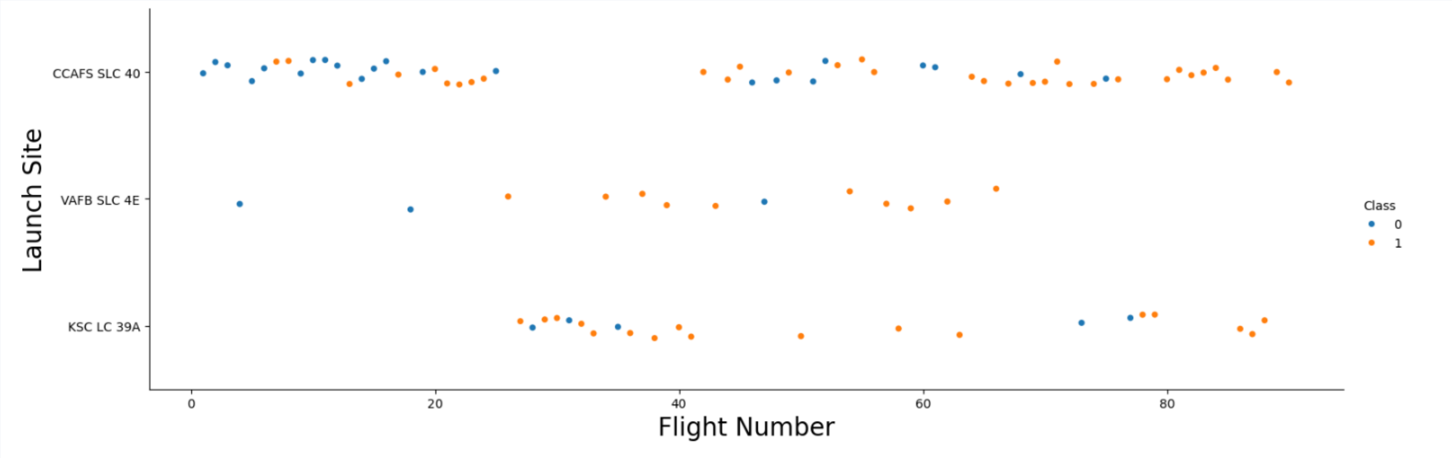


EDA with SQL

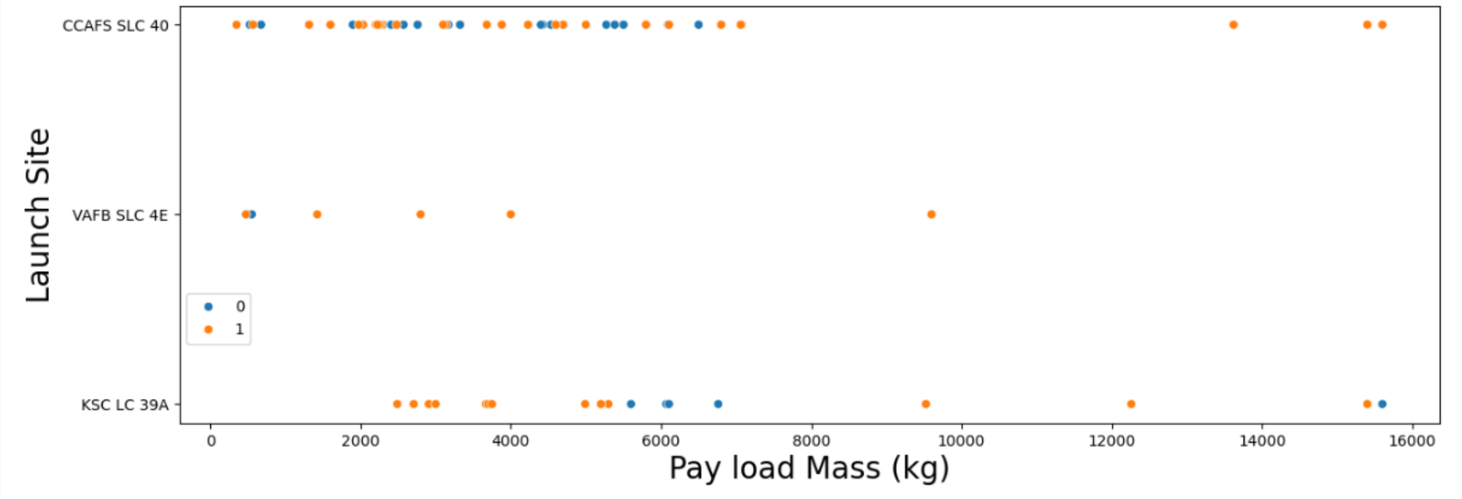
- SQL queries I performed
  - names of the unique launch sites in the space mission
  - records where launch sites begin with the string 'CCA'
  - total payload mass carried by boosters launched by NASA (CRS)
  - average payload mass carried by booster version F9 v1.1
  - date when the first successful landing outcome in ground pad was achieved.
  - names of the boosters which have been successful in drone ship and have payload mass greater than 4000 but less than 6000
  - total number of successful and failure mission outcomes
  - booster_versions that have carried the maximum payload mass, using a subquery with a suitable aggregate function.
  - records which will display the month names, failure landing_outcomes in drone ship ,booster versions, launch_site for the months in year 2015.
  - 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

https://github.com/LLZSG/Data-science-capstone/blob/main/jupyter-labs-eda-sql-coursera_sqllite%20(1).ipynb

EDA with Data Visualization

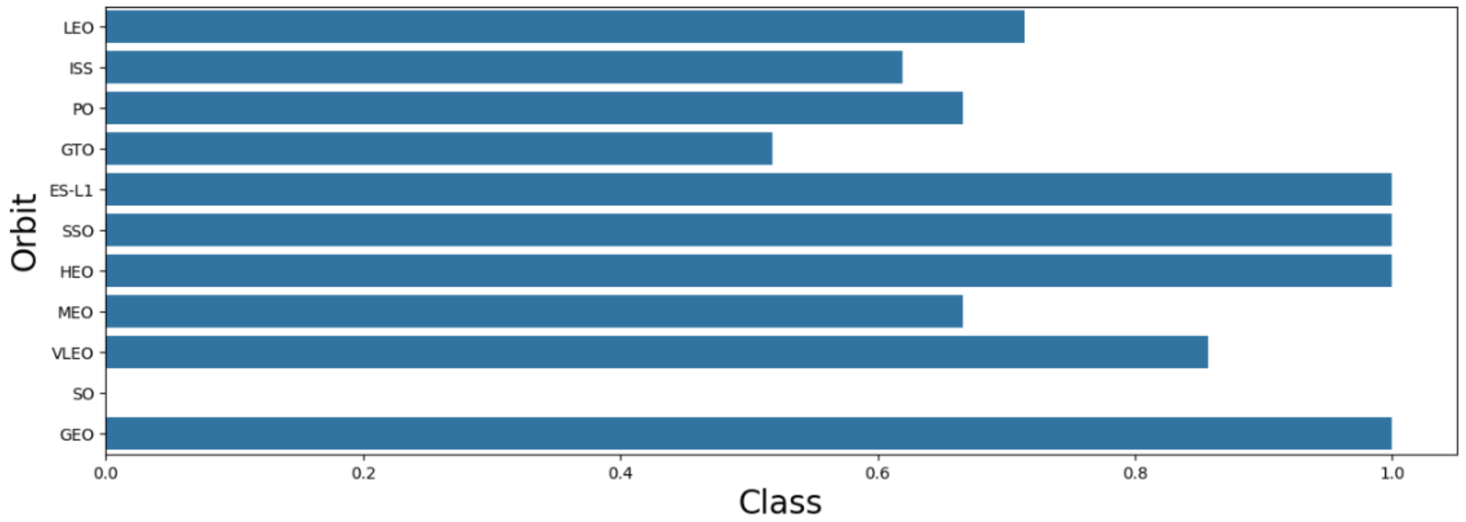


Relationship between Flight Number and Launch Site

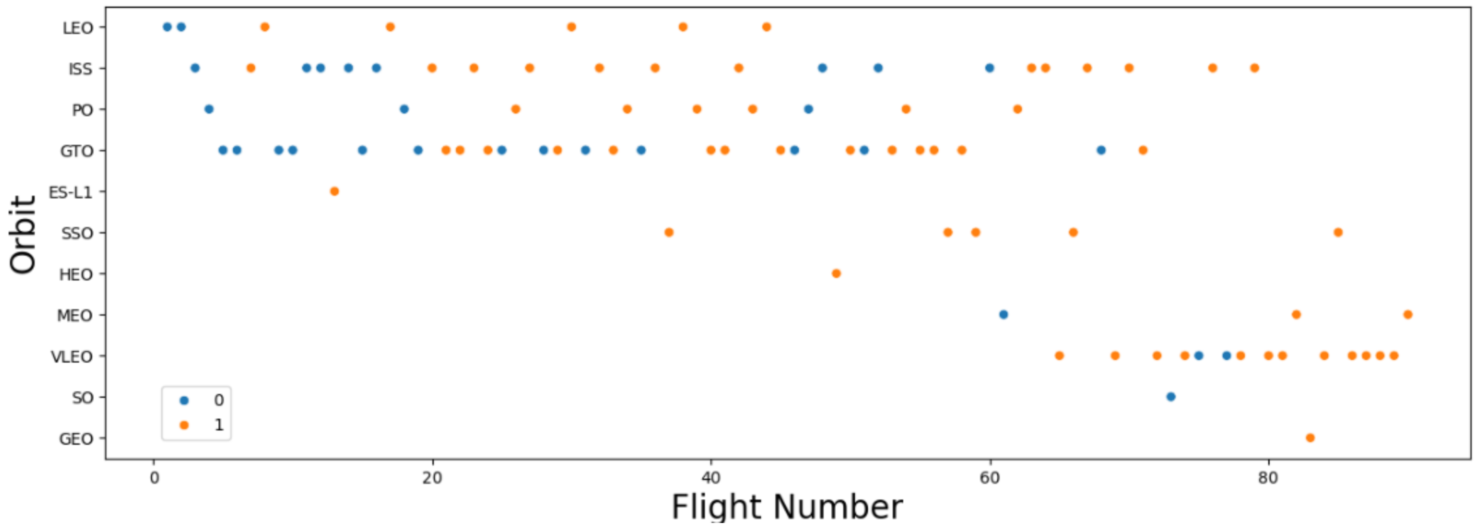


Relationship between Payload Mass and Launch Site
for the VAFB-SLC launch site there are no rockets launched for heavy payload mass(greater than 10000).

EDA with Data Visualization

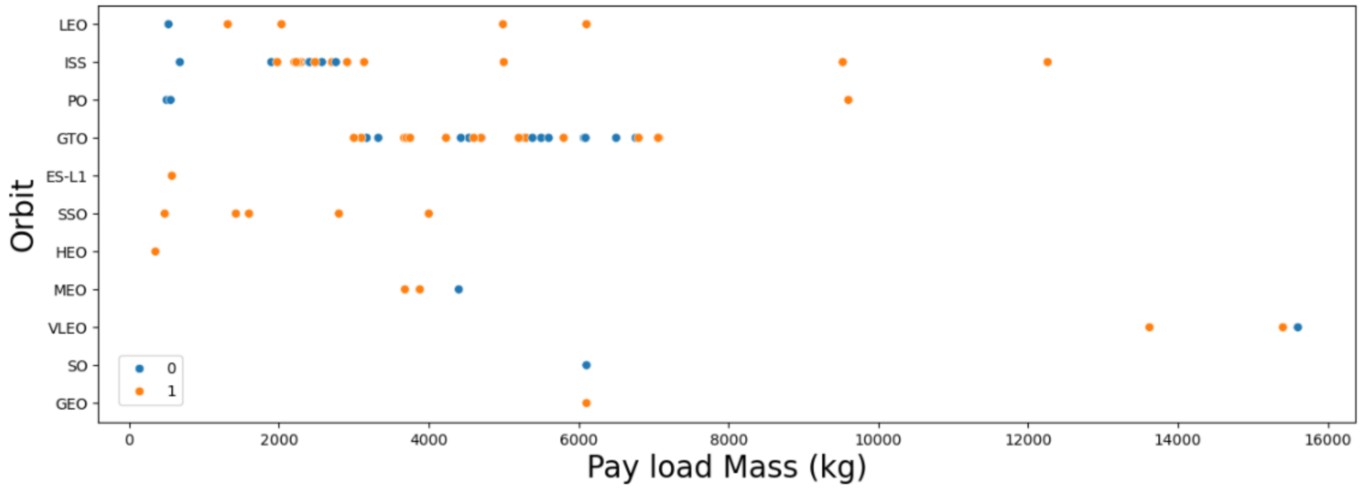


Relationship between success rate of each orbit type

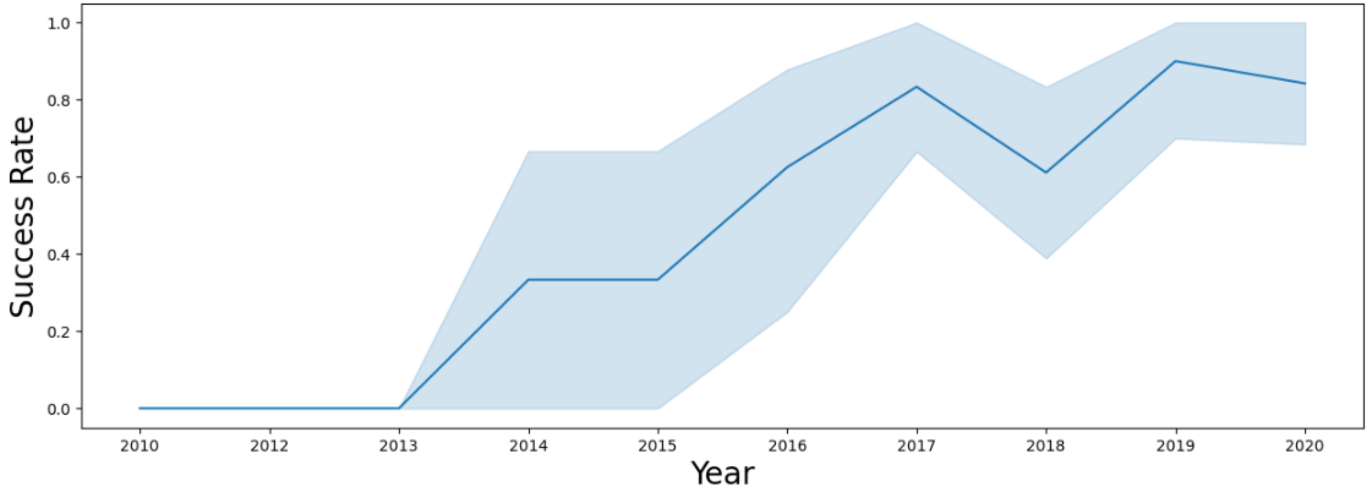


Relationship between Flight Number and Orbit type

EDA with Data Visualization



Relationship between between Payload Mass and Orbit type

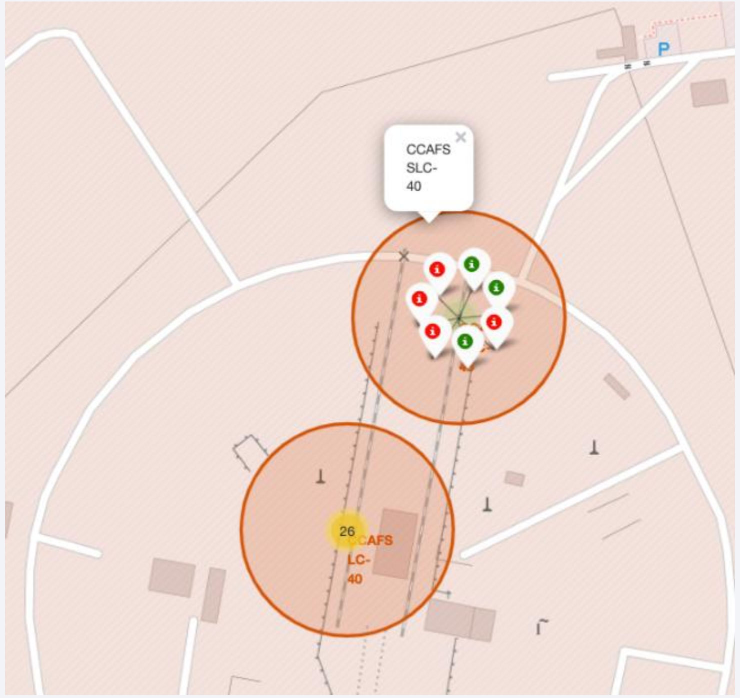


launch success yearly trend

No GitHub URL as I cannot download the file from the lab to upload to GitHub

Build an Interactive Map with Folium

- I marked all launch sites on a Folium map and added map elements such as markers, circles, and lines to show whether each launch was a success or failure.
- I labeled the launch outcomes as binary classes: 0 for failure and 1 for success.
- Using color-coded marker clusters, I was able to identify which launch sites had relatively high success rates.
- I also calculated distances from each launch site to nearby features such as railways, highways, coastlines, and cities.
- No GitHub URL as I cannot download the file from the lab to upload to GitHub

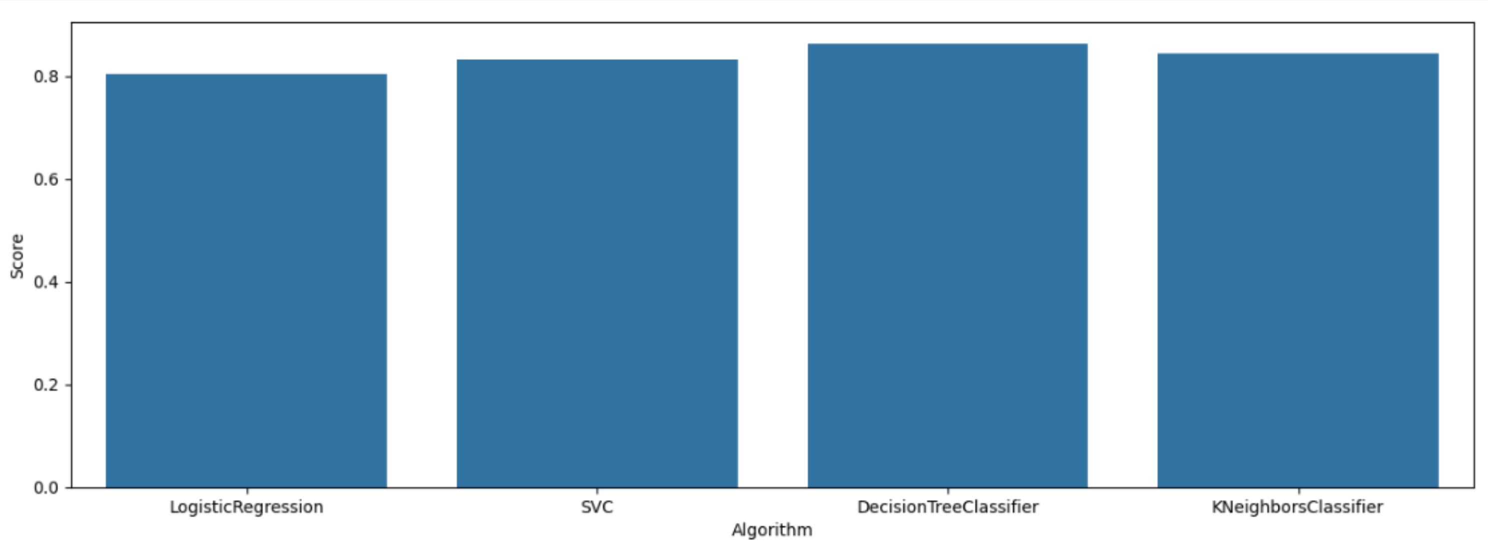


Build a Dashboard with Plotly Dash

- I created pie charts to show the total number of launches at each site.
- I also made scatter plots to show how the outcome of a launch relates to the payload mass (in kg) for different booster versions.

Predictive Analysis (Classification)

- I split the data, one set for training and another set for testing
- I use different parameters and calculate the accuracy on the test data using the method score, followed by plot the confusion matrix
- No GitHub URL as I cannot download the file from the lab to upload to GitHub



the method performs best

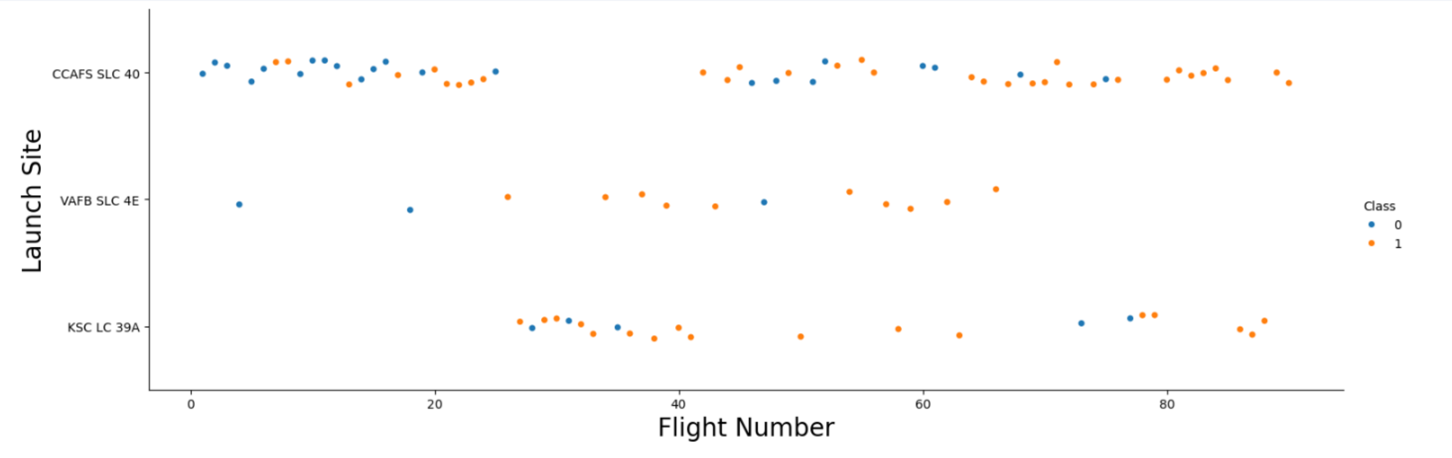
Results

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

Section 2

Insights drawn from EDA

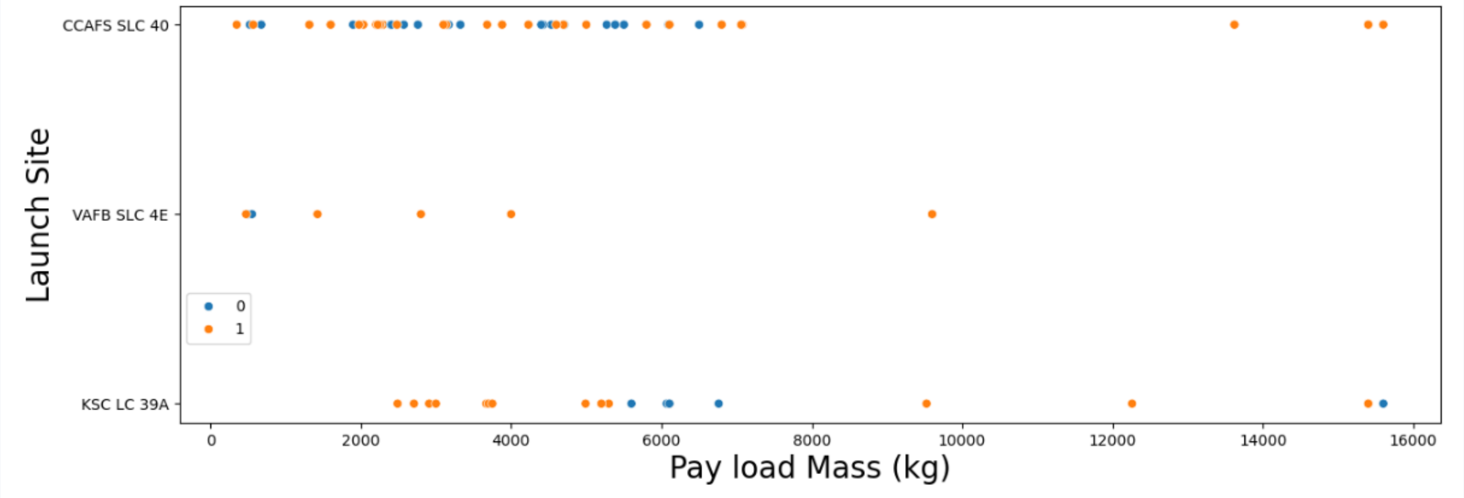
Flight Number vs. Launch Site



Relationship between Flight Number and Launch Site

The higher the flight number for each launch site, the greater is the success rate.

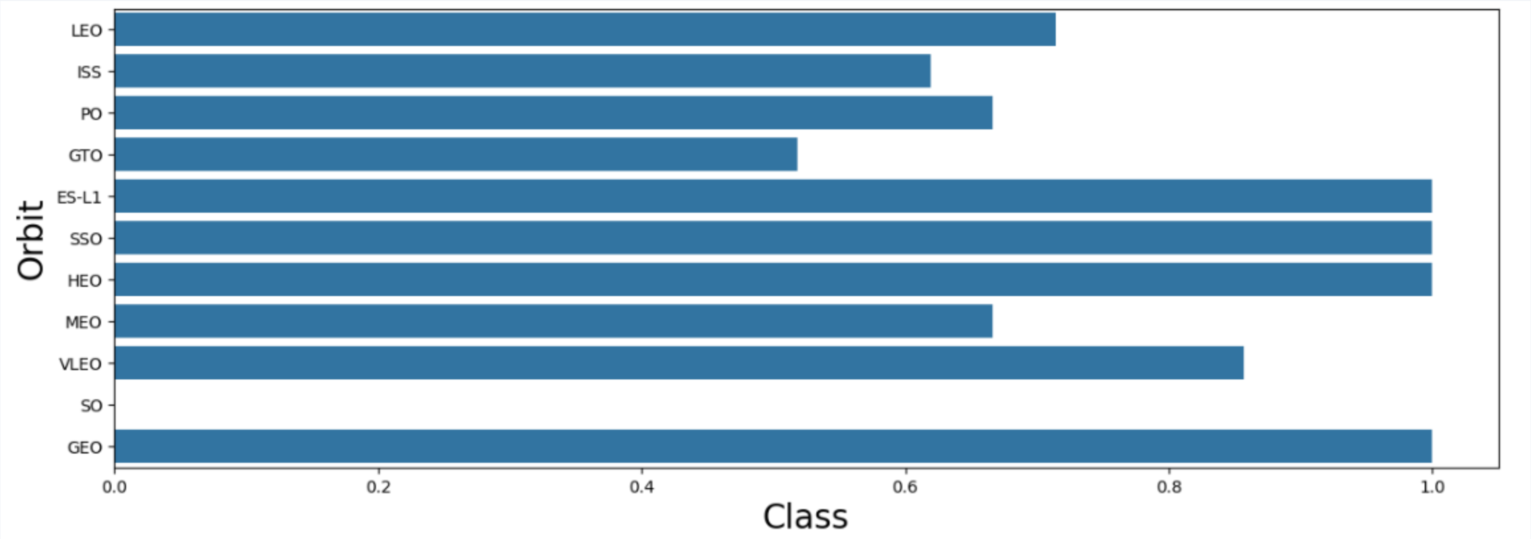
Payload vs. Launch Site



Relationship between Payload Mass and Launch Site

For **the VAFB-SLC launch site there are no rockets launched for heavy payload mass(greater than 10000).**

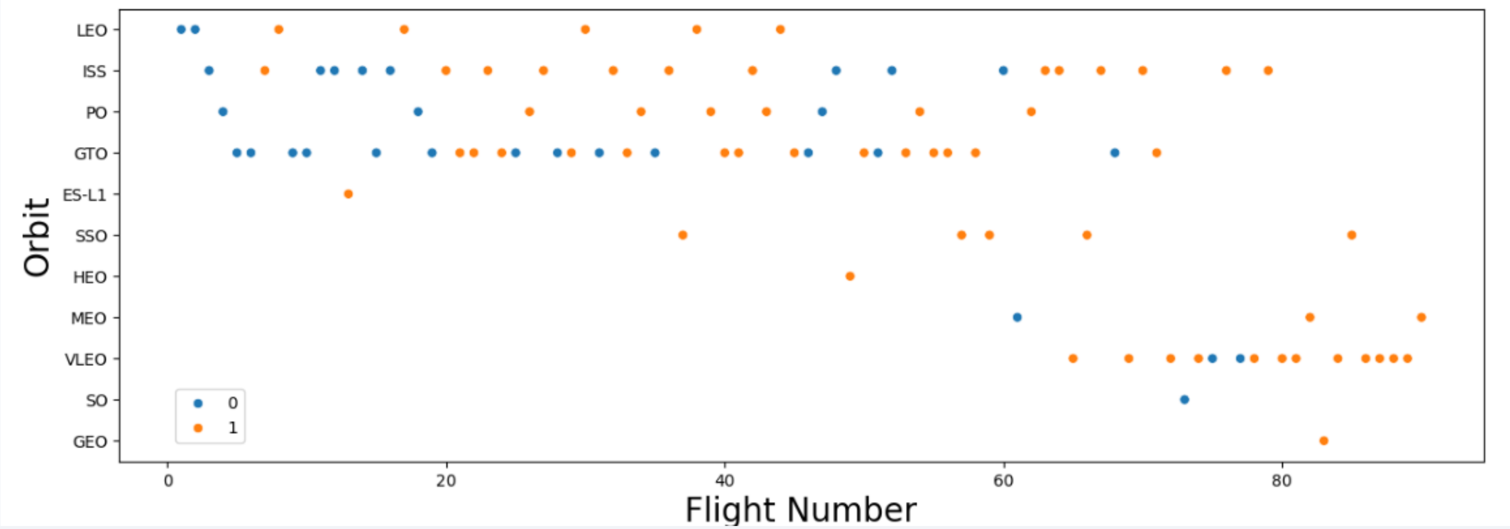
Success Rate vs. Orbit Type



Relationship between success rate of each orbit type

GEO, HEO, SSO and ES-L1 have the most success rate

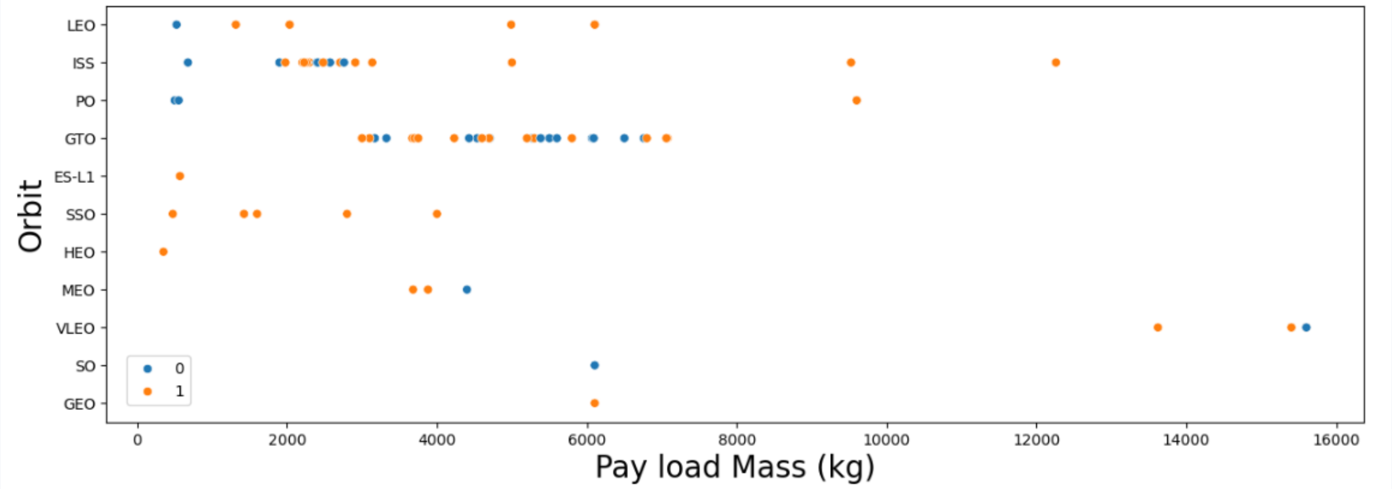
Flight Number vs. Orbit Type



Relationship between Flight Number and Orbit type

For LEO, when the flight number increase, the success rate increase.

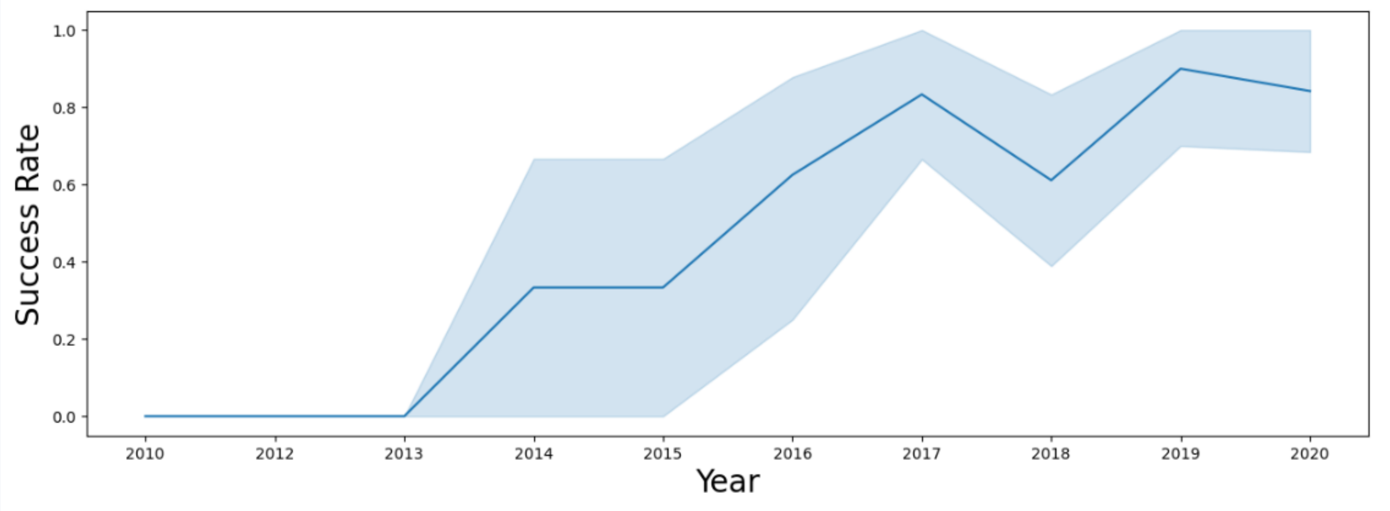
Payload vs. Orbit Type



Relationship between between Payload Mass and Orbit type

SSO has successful landing regardless the pay load mass.

Launch Success Yearly Trend



launch success yearly trend

Increasing success rate from 2013, though there is a dip in 2018

All Launch Site Names

Task 1

Display the names of the unique launch sites in the space mission

In [10]:

```
%%sql

SELECT DISTINCT Launch_Site from SPACEXTABLE;
```

```
 * sqlite:///my_data1.db
Done.
```

Out[10]:

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

Launch Site Names Begin with 'CCA'

Task 2

Display 5 records where launch sites begin with the string 'CCA'

In [11]:
```
%%sql

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

```
 * sqlite:///my_data1.db
Done.
```

Out[11]:

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

Task 3

Display the total payload mass carried by boosters launched by NASA (CRS)

In [12]:
```
%%sql

SELECT SUM(PAYLOAD_MASS__KG_) AS TOTAL_PAYLOAD FROM SPACEXTABLE WHERE Customer = 'NASA (CRS)';
```

```
 * sqlite:///my_data1.db
Done.
```

Out[12]:

TOTAL_PAYLOAD
45596

Average Payload Mass by F9 v1.1

Task 4

Display average payload mass carried by booster version F9 v1.1

In [13]:
```
%%sql

SELECT AVG(PAYLOAD_MASS__KG_) AS AVG_PAYLOAD_MASS FROM SPACEXTABLE WHERE Booster_Version LIKE 'F9 v1.1%';
```

```
 * sqlite:///my_data1.db
Done.
```

Out[13]:

AVG_PAYLOAD_MASS
2534.6666666666665

First Successful Ground Landing Date

Task 5

List the date when the first succesful landing outcome in ground pad was acheived.

Hint:Use min function

In [14]:

```
%%sql

SELECT MIN(Date) as LaunchDate FROM SPACEXTABLE WHERE Landing_Outcome = 'Success (ground pad)';
```

```
 * sqlite:///my_data1.db
Done.
```

Out[14]:

LaunchDate
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

Task 6

List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000

In [15]:

```
%%sql

SELECT Booster_Version, PAYLOAD_MASS__KG_ FROM SPACEXTABLE WHERE Landing_Outcome = 'Success (drone ship)' AND PAYLOAD_MASS_
```

```
 * sqlite:///my_data1.db
Done.
```

Out[15]:

Booster_Version	PAYLOAD_MASS__KG_
F9 FT B1022	4696
F9 FT B1026	4600
F9 FT B1021.2	5300
F9 FT B1031.2	5200

Total Number of Successful and Failure Mission Outcomes

Task 7

List the total number of successful and failure mission outcomes

In [17]:

```
%%sql

SELECT CASE
    WHEN Mission_Outcome LIKE 'Success%' THEN 'Success'
    WHEN Mission_Outcome LIKE 'Failure%' THEN 'Failure'
END as Mission_Status,
COUNT(*)
FROM SPACEXTABLE
GROUP BY Mission_Status;
```

```
 * sqlite:///my_data1.db
Done.
```

Out[17]:

Mission_Status	COUNT(*)
Failure	1
Success	100

Boosters Carried Maximum Payload

Task 8

List all the booster_versions that have carried the maximum payload mass, using a subquery with a suitable aggregate function.

In [18]:

```
%%sql

SELECT DISTINCT Booster_Version, PAYLOAD_MASS__KG_
    FROM SPACEXTABLE
    WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTABLE)
    ORDER BY Booster_Version;
```

```
 * sqlite:///my_data1.db
Done.
```

Out[18]:

Booster_Version	PAYLOAD_MASS__KG_
F9 B5 B1048.4	15600
F9 B5 B1048.5	15600
F9 B5 B1049.4	15600
F9 B5 B1049.5	15600
F9 B5 B1049.7	15600
F9 B5 B1051.3	15600
F9 B5 B1051.4	15600
F9 B5 B1051.6	15600
F9 B5 B1056.4	15600
F9 B5 B1058.3	15600
F9 B5 B1060.2	15600
F9 B5 B1060.3	15600

2015 Launch Records

Task 9

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.

Note: SQLLite does not support monthnames. So you need to use substr(Date, 6,2) as month to get the months and substr(Date,0,5)='2015' for year.

In [19]:

```
%%sql

SELECT
    CASE strftime('%m', Date)
        WHEN '01' THEN 'January'
        WHEN '02' THEN 'February'
        WHEN '03' THEN 'March'
        WHEN '04' THEN 'April'
        WHEN '05' THEN 'May'
        WHEN '06' THEN 'June'
        WHEN '07' THEN 'July'
        WHEN '08' THEN 'August'
        WHEN '09' THEN 'September'
        WHEN '10' THEN 'October'
        WHEN '11' THEN 'November'
        WHEN '12' THEN 'December'
    END as Month,
    Landing_Outcome, Booster_Version, Launch_Site, Date
FROM SPACEXTABLE
WHERE strftime('%Y', Date) = '2015' AND Landing_Outcome = 'Failure (drone ship)';
```

```
 * sqlite:///my_data1.db
Done.
```

Out[19]:

Month	Landing_Outcome	Booster_Version	Launch_Site	Date
January	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40	2015-01-10
April	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40	2015-04-14

Rank Landing Outcomes Between 2010-06-04 and 2017-03-20

Task 10

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.

In [20]:

```
%%sql

SELECT Landing_Outcome, COUNT(*) as Count
    FROM SPACEXTABLE
        WHERE Date BETWEEN '2010-06-04' AND '2017-03-20'
        GROUP BY Landing_Outcome
        ORDER BY Count DESC;
```

```
 * sqlite:///my_data1.db
Done.
```

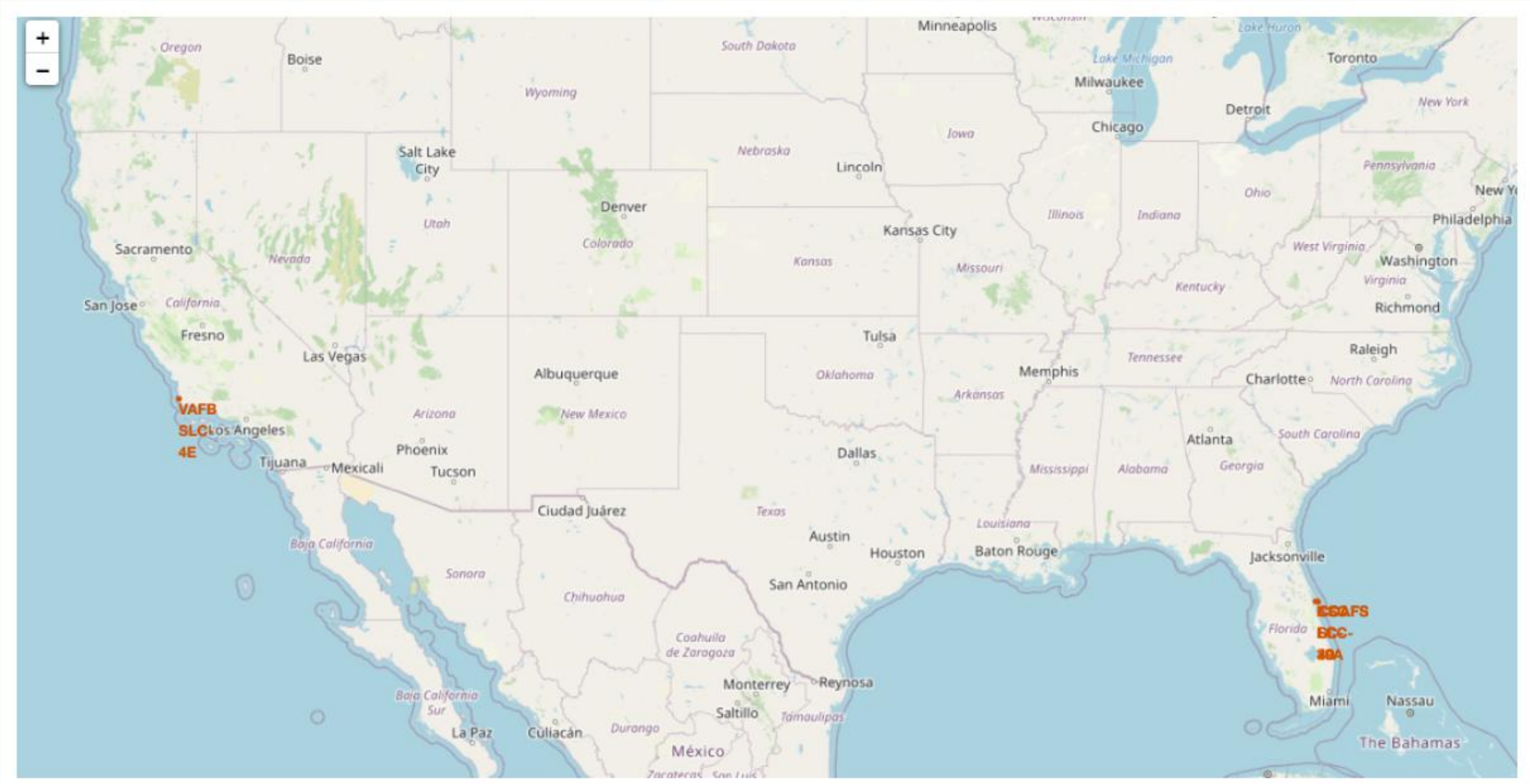
Out[20]:

Landing_Outcome	Count
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

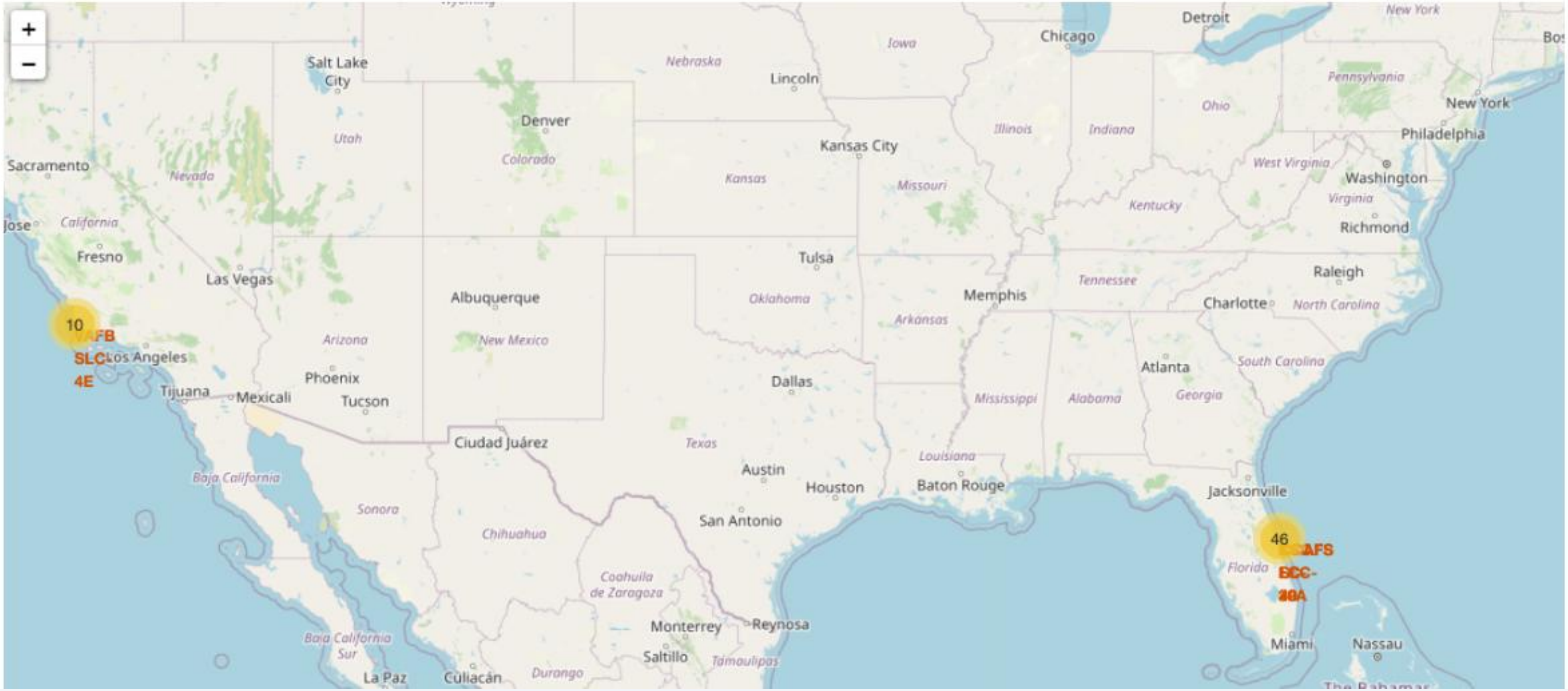
Section 3

Launch Sites Proximities Analysis

Mark all launch sites on a map



Mark the success/failed launches for each site on the map

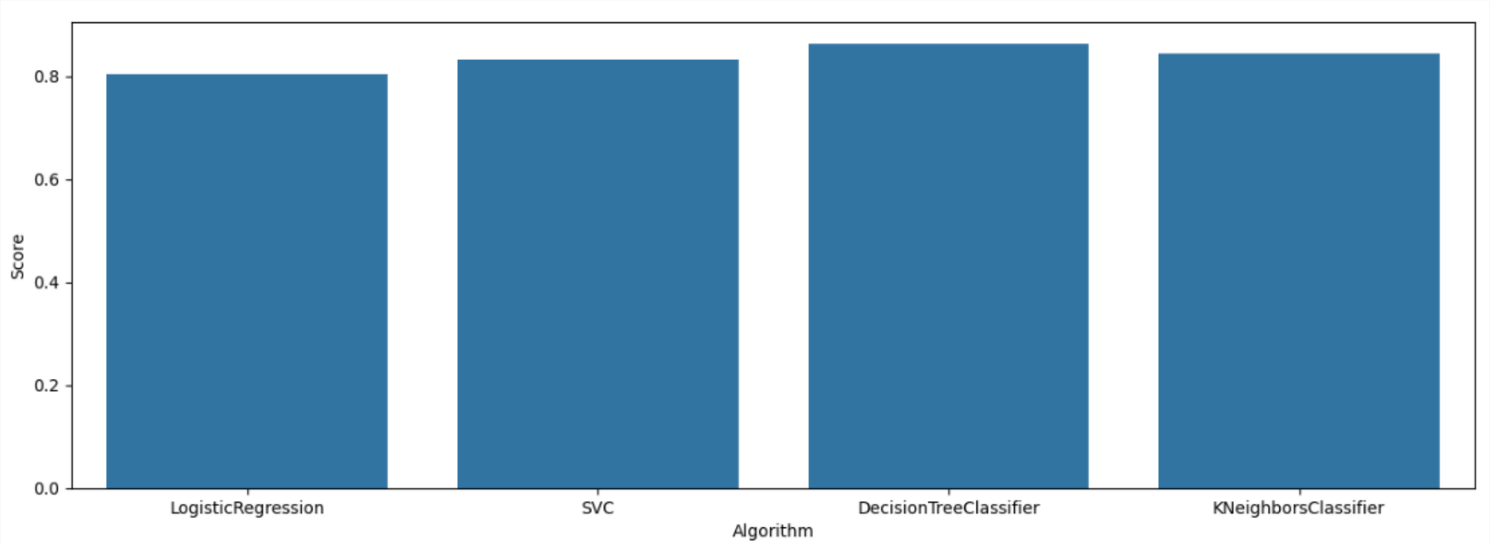


Calculate the distances between a launch site to its proximities

Section 5

Predictive Analysis (Classification)

Classification Accuracy

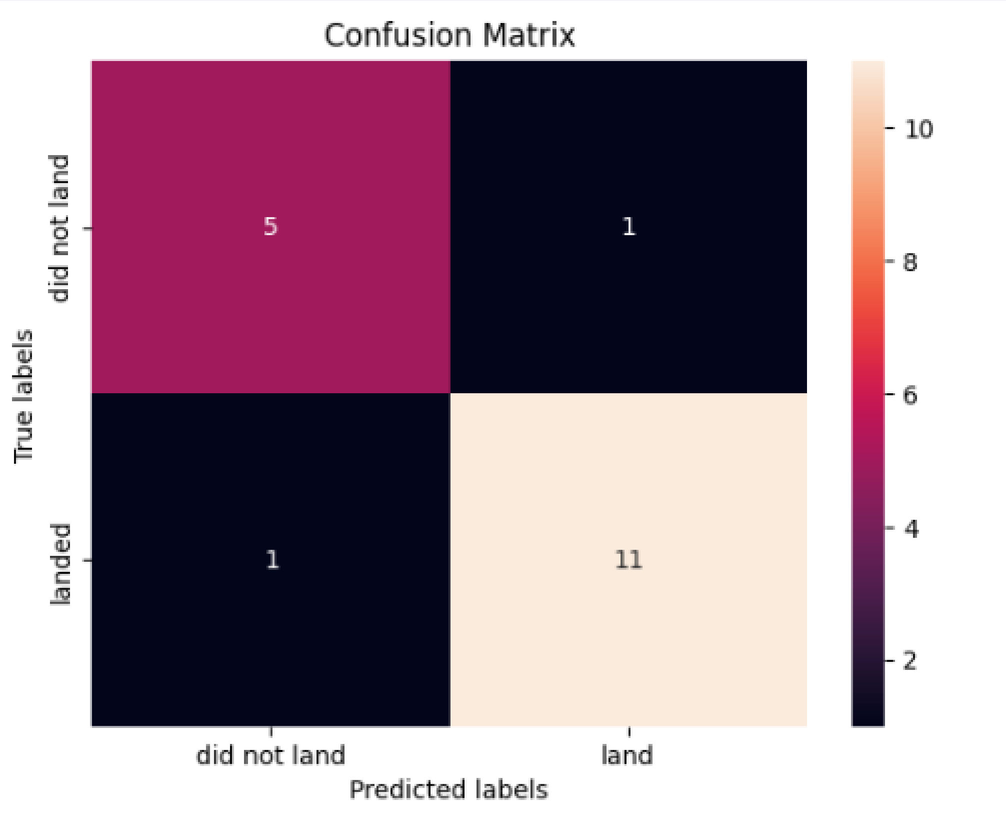


the method performs best

DecisionTreeClassifer

Confusion Matrix

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



Conclusion and Key Takeaways

- Successfully predicted Falcon 9 first stage landing outcomes using machine learning.
- Identified critical factors influencing rocket landing success through data analysis.
- Leveraged combined API and web scraping data collection for robust datasets.
- Applied comprehensive data wrangling and feature engineering techniques.
- Utilized interactive visual analytics for deeper insights and decision-making.
- Provided a predictive tool valuable for competitive launch cost estimation.

Thank you!