IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

<Aditya Kumar Sony>
<14-Dec-2025>

Outline

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

Executive Summary

- This project analyzes SpaceX Falcon 9 launches to understand how first-stage reuse impacts launch costs and it provides data for minimizing launch cost.
- The methodologies used are:
  - Data Collection, Data preparation, Exploratory Data Analysis (EDA), Data Visualization and Machine Learning
- Summary of all results
  - Data collection by web-scraping from open source
  - Feature selection by EDA
  - Predictive machine learning model for useful information

Introduction

- The commercial space age has begun, with private companies making space travel more affordable.
- SpaceX leads this revolution through reusable rocket technology and low-cost launches.
- Predicting first-stage reuse is key to understanding and reducing launch costs.
- This study aims to provide insights from Space X data to Space Y in order to compete Space X using machine learning technique.

The objective is to:

- Analyze SpaceX launch data,
- Build dashboards for insights,
- Predict whether the first stage will be reused using **machine learning**.

Section 1

Methodology

Methodology

Executive Summary

- Perform data collection methodology
- Perform data wrangling
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models

Data Collection

We will be working with SpaceX launch data that is gathered from an API, specifically the SpaceX REST API and from Wikipedia by web scraping.

This API will give us data about launches, including information about the rocket used, payload delivered, launch specifications, landing specifications, and landing outcome.

Our goal is to use this data to predict whether SpaceX will attempt to land a rocket or not. The SpaceX REST API endpoints, or URL, starts with api.spacexdata.com/v4/.

We have the different end points, for example: /capsules and /cores We will be working with the endpoint api.spacexdata.com/v4/launches/past.

Data Collection – SpaceX API

Key phrases:

- Public SpaceX data source
- Lauch records retrieval
- Mission and booster information
- Payload and orbit data
- Landing outcome data
- Dataset consolidation

- GitHub URL: https://github.com/adityakumarsony100-stack/Applied-Data-Science-Capstone-Project-IBM/blob/74f5655da2723c50e857c595c66ac2c8c4128261/DataCollection_by_REST_API.ipynb

Flowchart:

Public SpaceX Sources

↓

Launch & Mission Data

↓

Booster & Payload Details

↓

Orbit & Launch Site Info

↓

Landing Outcome Records

↓

Final Combined Dataset

Data Collection - Scraping

Key phrases:

- Identify target website
- Inspect HTML structure
- Parse webpage content
- Extract required fields
- Store structured data

- GitHub
  URL:https://github.com/adityakumarsony100-stack/Applied-Data-Science-Capstone-Project-IBM/blob/74f5655da2723c50e857c595c66ac2c8c4128261/DataCollection-webscraping.ipynb

Flowchart

Target Website
↓
HTML Inspection
↓
HTTP Request
↓
HTML Parsing
↓
Data Extraction
↓
Structured Dataset

Data Wrangling

The data was cleaned and relevant attributes such as launch site, orbit, and landing outcome were selected. The landing outcome was standardized and converted into a binary variable, where **1** indicates a successful first-stage landing and **0** indicates failure. This processed data was then prepared for classification analysis.

Key phrases:

- Data collection
- Feature selection
- Data cleaning
- Categorical encoding
- Outcome binarization
- Ready for Modeling

GitHub URL:https://github.com/adityakumarsony100-stack/Applied-Data-Science-Capstone-Project-IBM/blob/74f5655da2723c50e857c595c66ac2c8c4128261/labs-jupyter-spacex-Data%20wrangling.ipynb

Flowchart:

Raw Launch Data
↓
Attribute Selection
↓
Data Cleaning & Formatting
↓
Categorical Variables Identified
↓
Outcome Standardization
(True/False → 1/0)
↓
Final Dataset
(Ready for Analysis & ML)

EDA with Data Visualization

- Exploratory data analysis helps to understand the relationship between different variable. It helps in feature selection
- The plots generated are:
  - Catplot
  - Scatterplot
  - Barchart
  - Line plot
  - Heatmap
  - Boxplots
- Github URL:https://github.com/adityakumarsony100-stack/Applied-Data-Science-Capstone-Project-IBM/blob/74f5655da2723c50e857c595c66ac2c8c4128261/EDA.ipynb

EDA with SQL

- Summary of the SQL queries performed are:
  - Display the number of unique launch sites in the space mission
  - Display 5 records where launch sites begin with the string 'CCA'
  - Display the total payload mass carried by boosters launched by NASA(CRS)
  - Display average payload mass carried by booster version F9 v1.1
  - List the date when the first successful landing outcome in ground pad was acheived.
  - List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000
  - List the total number of successful and failure mission outcome
  - List all the booster_versions that have carried the maximum payload mass, using a subquery with a suitable aggregate function.
  - 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.
  - 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.

Githhub URL: https://github.com/adityakumarsony100-stack/Applied-Data-Science-Capstone-Project-IBM/blob/74f5655da2723c50e857c595c66ac2c8c4128261/jupyter-labs-eda-sql-coursera_sqllite%20(1).ipynb

Build an Interactive Map with Folium

- **Markers (folium.Marker)**:
  Used to pinpoint exact locations such as launch sites, closest coastline points, cities, railways, and highways. Markers can have customized icons or colors (e.g., green for success, red for failure) to represent different statuses.
- **Marker Clusters (folium.plugins.MarkerCluster)**:
  Group multiple markers together to improve map readability when many points are close to each other. This helps in managing overlapping markers on the map.
- **Circles (folium.Circle)**:
  Added around launch sites or other points to highlight an area with a radius, often with popup labels to provide additional information.
- **Lines (folium.PolyLine)**:
  Drawn between two points (e.g., launch site and closest feature like coastline or city) to visually represent the proximity or connection between them.
- **DivIcon Markers (folium.features.DivIcon)**:
  Custom markers that display text labels directly on the map, such as showing the calculated distance between two points in kilometers.
- These objects together create an interactive and informative map that helps you analyze spatial relationships related to launch sites.

- Github URL:https://github.com/adityakumarsony100-stack/Applied-Data-Science-Capstone-Project-IBM/blob/74f5655da2723c50e857c595c66ac2c8c4128261/lab_jupyter_launch_site_location%20Folium.ipynb

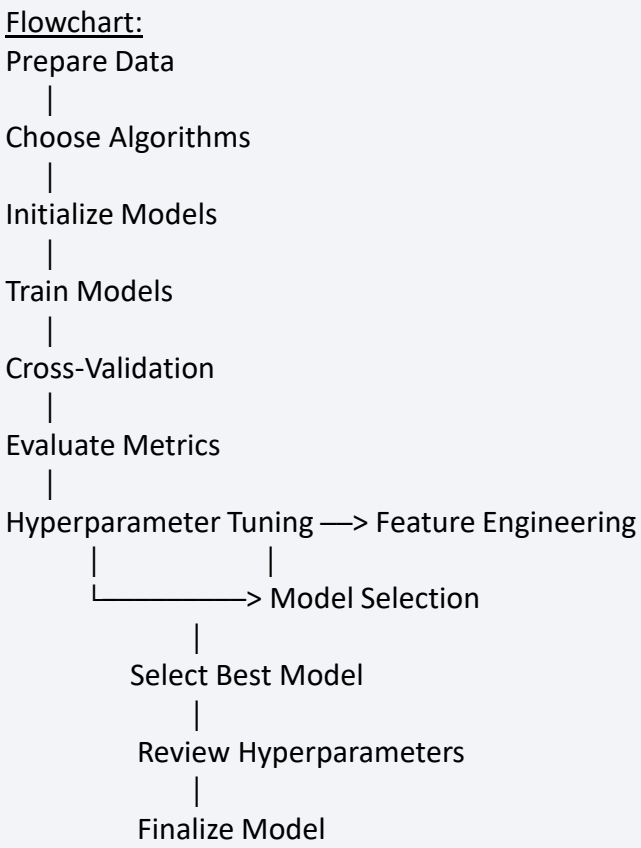
Build a Dashboard with Plotly Dash

- **Plots/Graphs:**
  - **Pie Chart:** Shows launch success counts overall or by selected launch site.
  - **Scatter Plot:** Shows payload mass vs. launch success, color-coded by booster version.
  - **Interactions:**
  - **Dropdown:** Select launch site to update pie chart.
  - **Range Slider:** Filter payload range to update scatter plot.
- **Why:**
- These plots and controls let you explore launch success visually and interactively, helping you analyze SpaceX data easily in your cloud IDE.

- Add the GitHub URL: https://github.com/adityakumarsony100-stack/Applied-Data-Science-Capstone-Project-IBM/blob/74f5655da2723c50e857c595c66ac2c8c4128261/spacex-dash-app.py

Predictive Analysis (Classification)

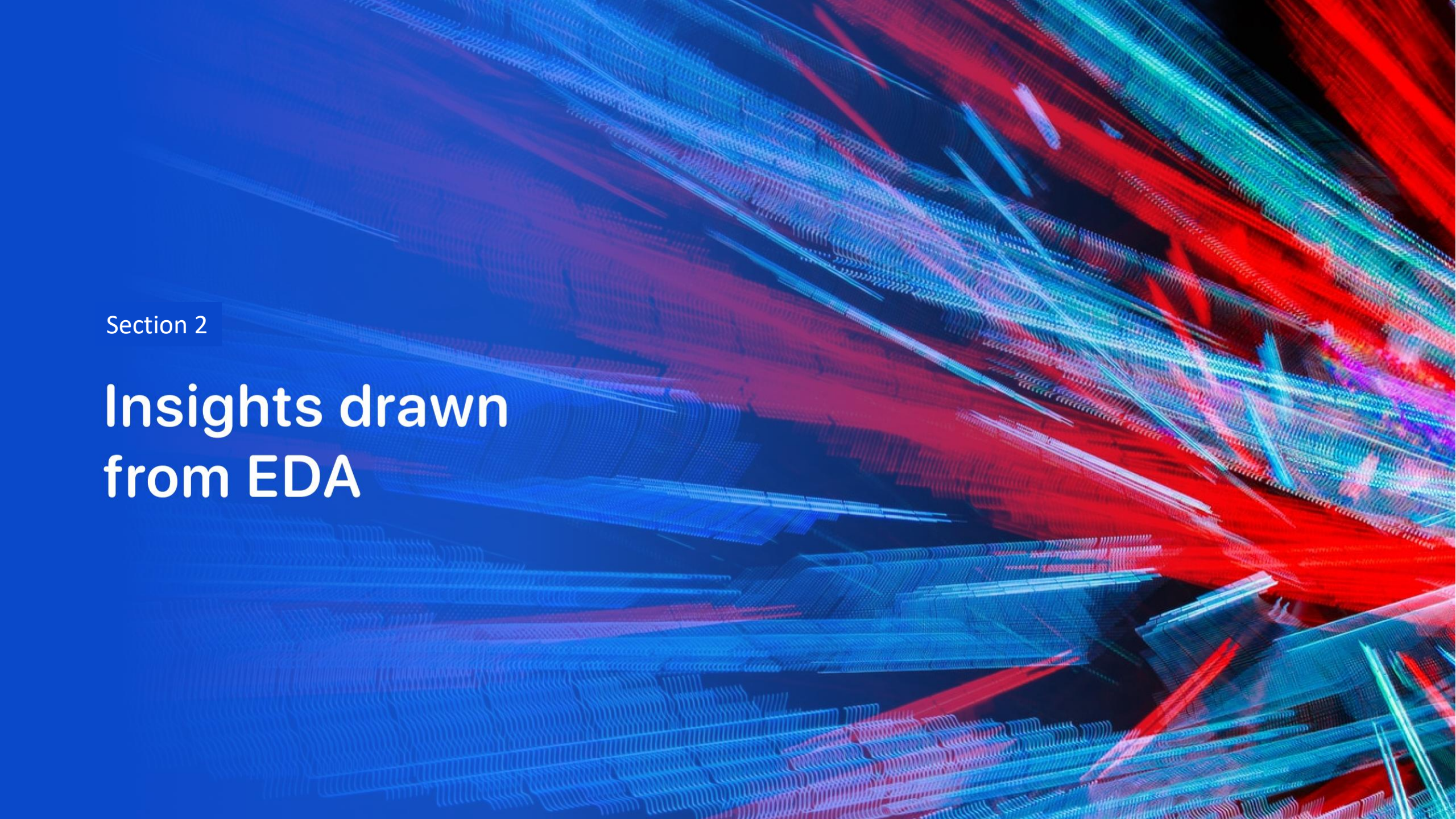
- **1. Build Model**
- Prepare Data → Clean, preprocess, normalize/standardize, split train/test
- Choose Algorithms → SVM, Decision Tree, Logistic Regression, KNN
- Initialize Models → Create model objects
- **2. Evaluate Model**
- Train Models → Fit on training data
- Cross-Validation → GridSearchCV with CV
- Metrics → Test accuracy / performance evaluation
- **3. Improve Model**
- Hyperparameter Tuning → GridSearchCV for best params
- Feature Engineering → Create/select better features
- Model Selection → Compare tuned models
- **4. Select Best Model**
- Identify best model → Highest accuracy / performance
- Review hyperparameters → From grid search
- Finalize model → Ready for deployment



GitHub URL: https://github.com/adityakumarsony100-stack/Applied-Data-Science-Capstone-Project-IBM/blob/74f5655da2723c50e857c595c66ac2c8c4128261/SpaceX_Machine%20Learning%20Prediction_Part_5.ipynb

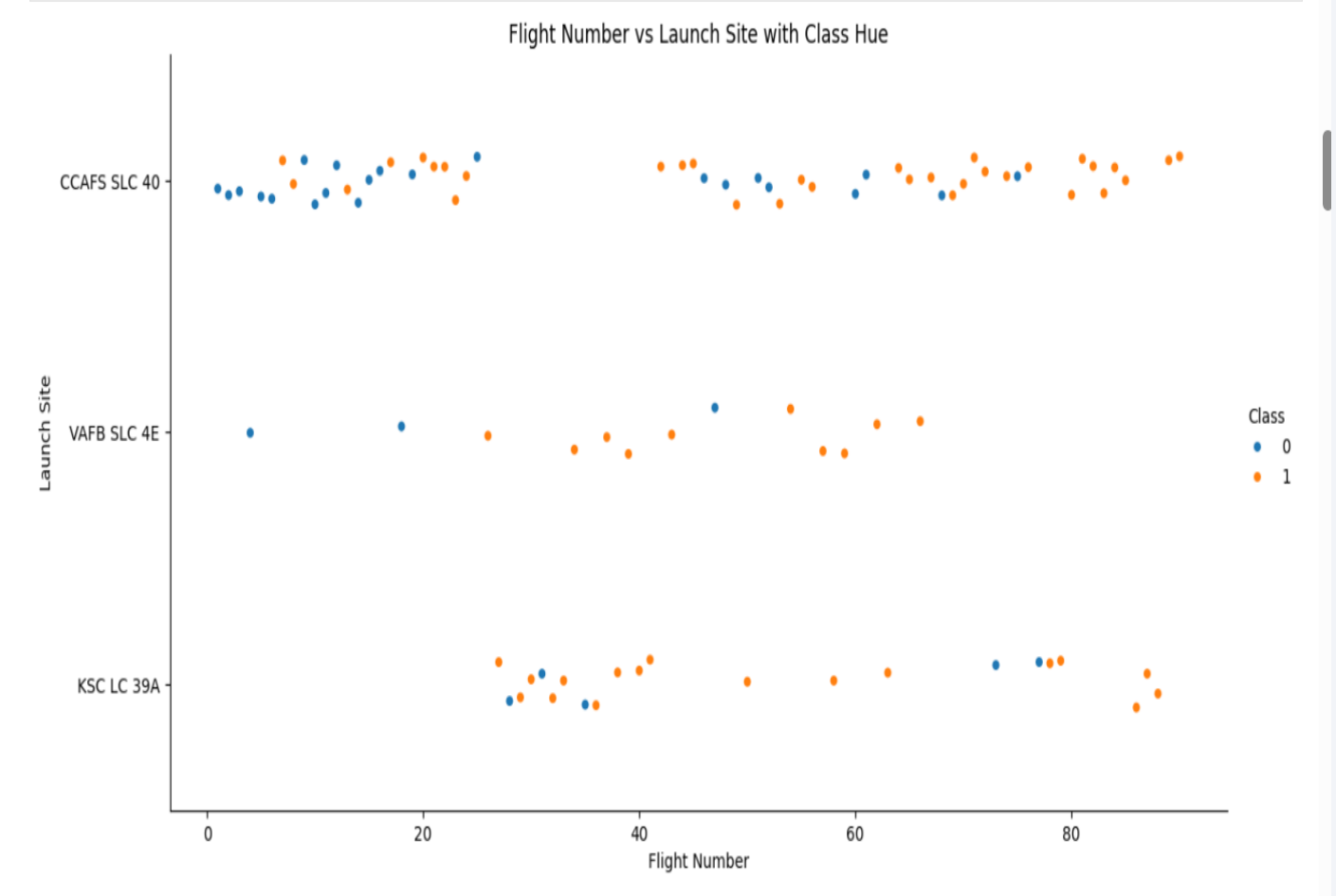
Results

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


Section 2
Insights drawn from EDA

Flight Number vs. Launch Site

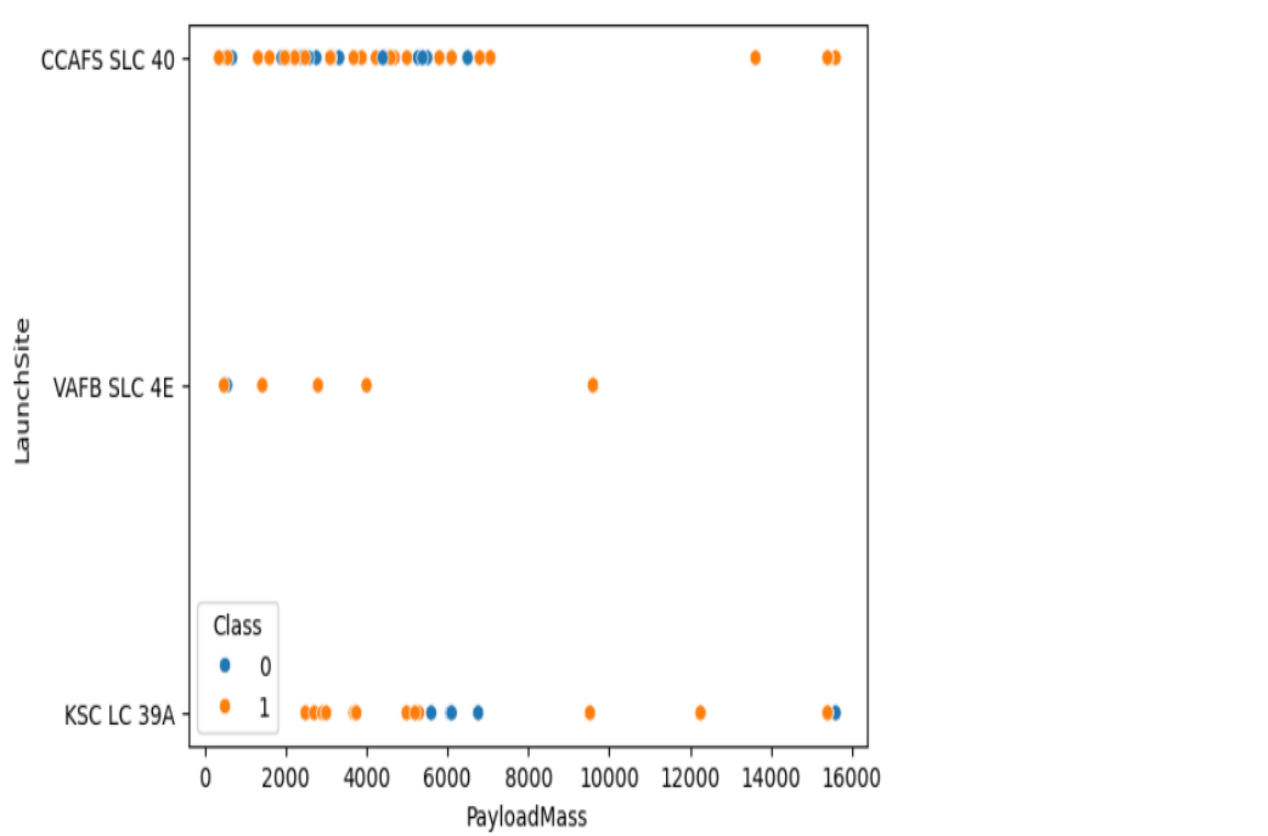
- The pattern between Flight Number and Launch Site shows how launch activity varies over time at different sites.
- Higher flight numbers indicate later launches, and you can see which sites were active during those times.
- Success rates (hue 'Class') may also vary by site and flight number.



Payload vs. Launch Site

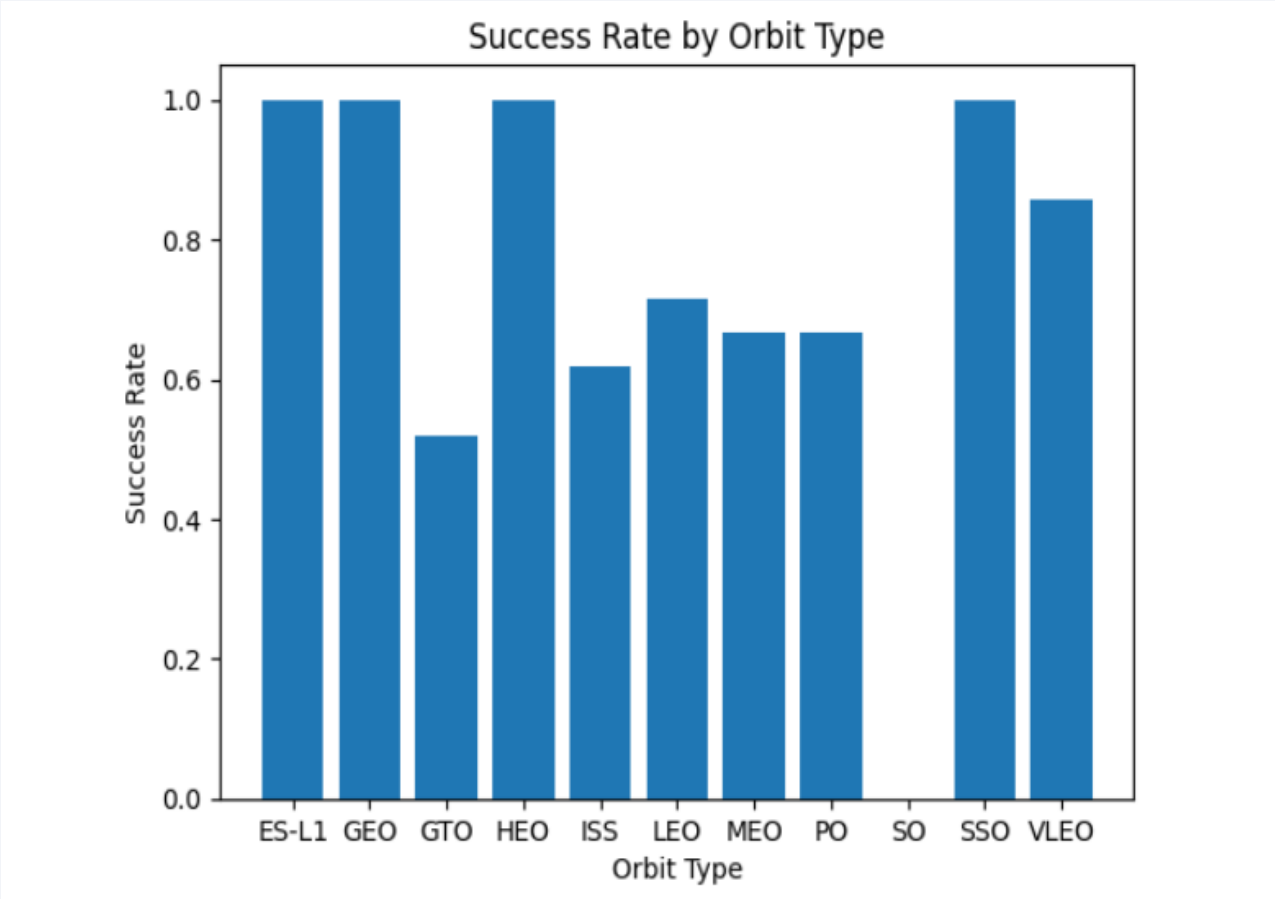
- The pattern shows that some launch sites (like VAFB-SLC) don’t launch heavy payloads (over 10,000 kg), while others handle a wider range of payload masses.
- This indicates site-specific payload capabilities.

```
# 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 class value
sns.scatterplot(x='PayloadMass',y='LaunchSite',data=df,hue='Class')
plt.show()
```



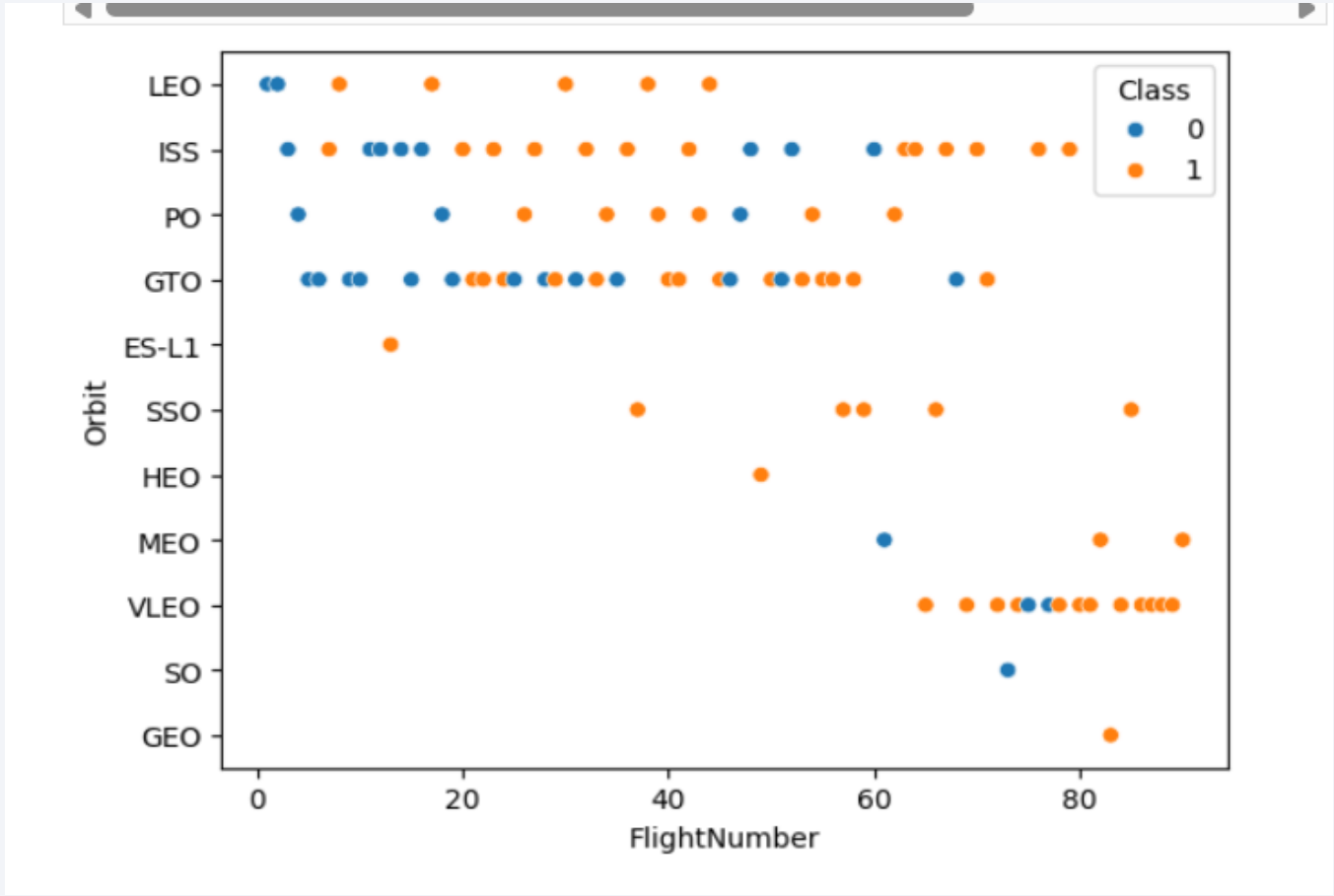
Success Rate vs. Orbit Type

- The pattern shows that orbits like LEO and ISS have the highest success rates, while others like GTO tend to have lower success.
- This indicates launches to certain orbits are more reliable.



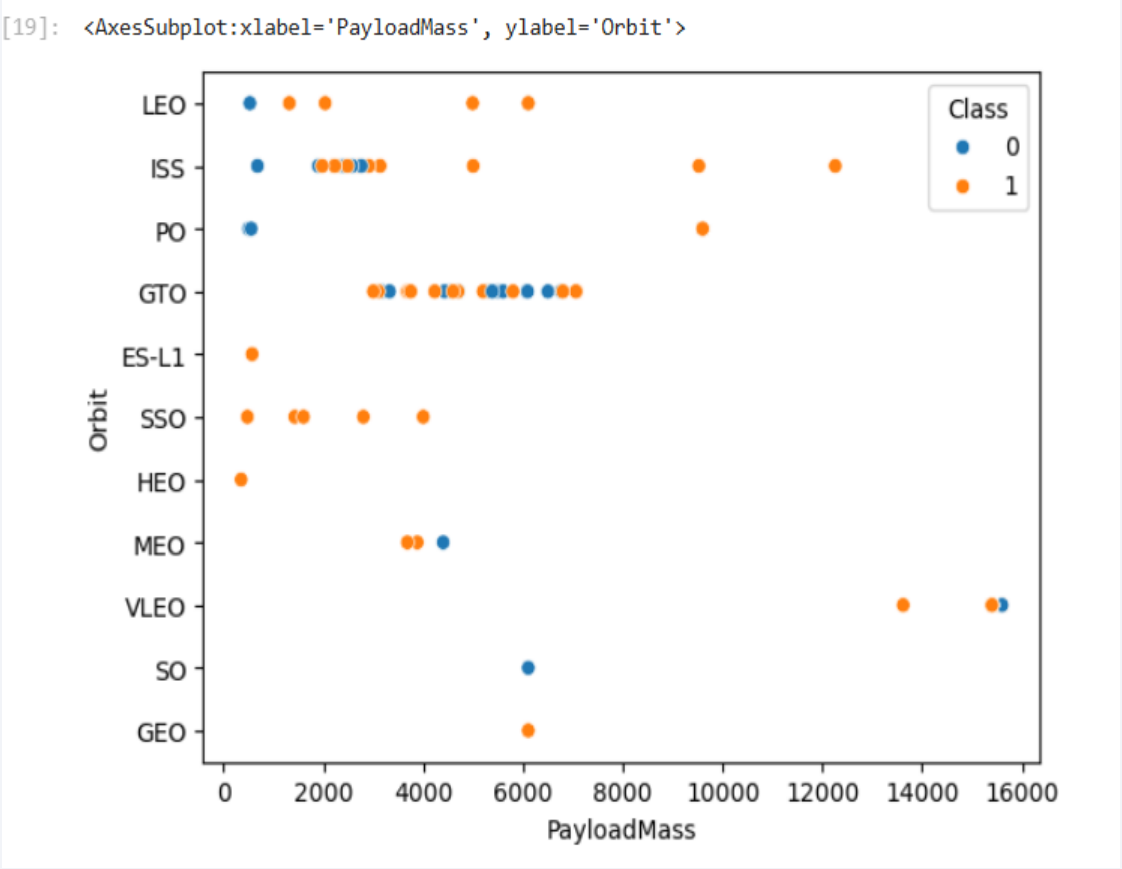
Flight Number vs. Orbit Type

- The pattern between Orbit Type and Flight Number shows how different orbits have been targeted over time.
- Early flight numbers may focus on certain orbits, while later flights expand to others.
- Some orbits have more frequent launches, reflected by more flight numbers.
- This trend helps understand the evolution of mission goals and orbit preferences.



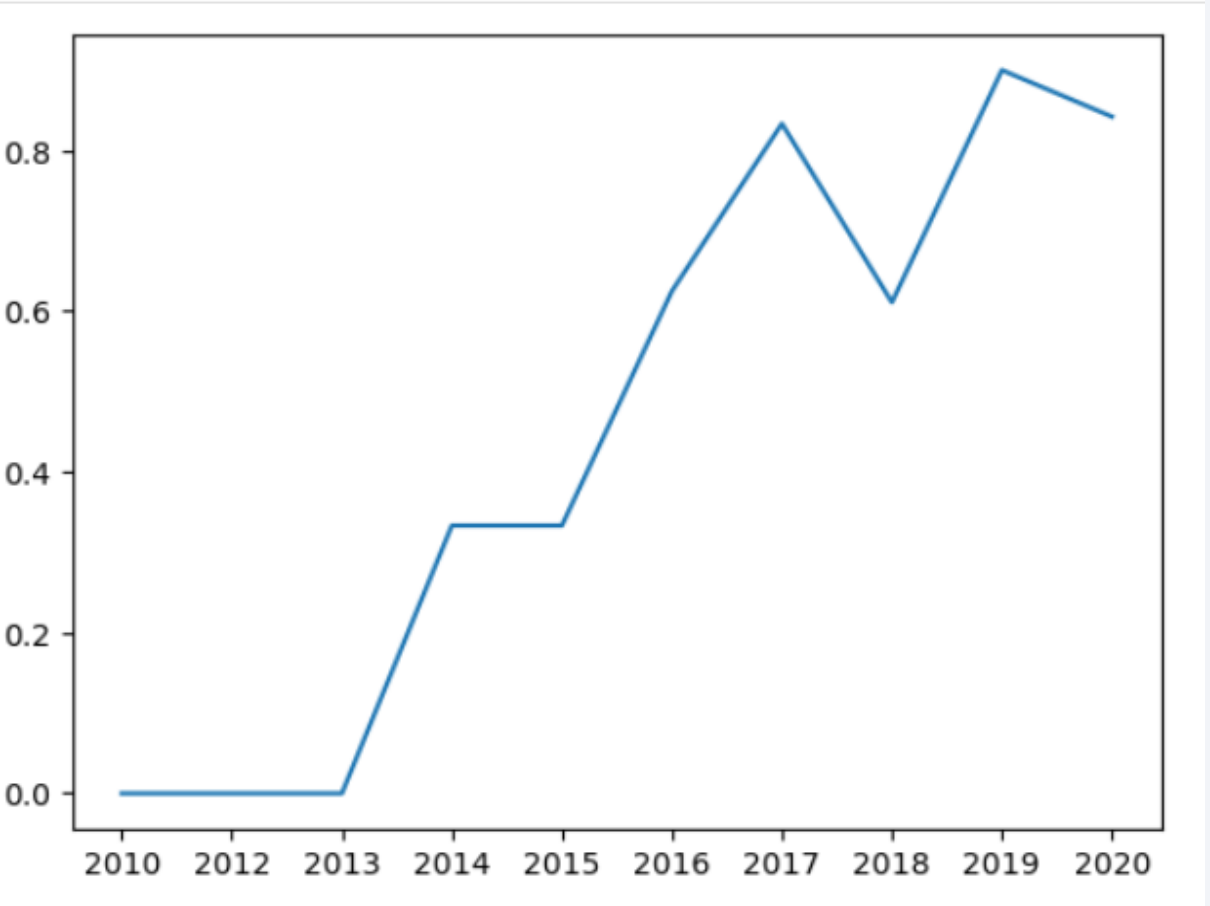
Payload vs. Orbit Type

- Heavier payloads are usually launched to orbits like GTO, while lighter payloads go to LEO.
- This shows how payload mass varies with orbit type.



Launch Success Yearly Trend

- This chart shows Lauch Success is increasing yearly with some fluctuations.



All Launch Site Names

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

```
%sql SELECT DISTINCT Launch_Site FROM SPACEXTBL
```

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

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

Launch Site Names Begin with 'CCA'

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

```
%sql SELECT * FROM SPACEXTBL WHERE "Launch_Site" like 'CCA%'  limit 5 ;
```

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

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

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

```
[15]: %sql SELECT SUM(PAYLOAD_MASS__KG_) FROM SPACEXTBL WHERE "CUSTOMER"='NASA (CRS)'
```

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

[15]:

SUM(PAYLOAD_MASS__KG_)
45596

Average Payload Mass by F9 v1.1

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

```
9]: %sql SELECT AVG(PAYLOAD_MASS__KG_) FROM SPACEXTBL WHERE "BOOSTER_VERSION"='F9 v1.1'
```

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

9]:

AVG(PAYLOAD_MASS__KG_)
2928.4

First Successful Ground Landing Date

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

Hint:Use min function

```
8]: %sql SELECT MIN("Date") FROM SPACEXTBL WHERE "Landing_Outcome" ='Success (ground pad)'
```

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

8]:

MIN("Date")
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

```
]: %sql SELECT Booster_Version FROM SPACEXTBL WHERE "Landing_Outcome"='Success (drone ship)' AND "PAYLOAD_MASS__KG_" BETWEEN 4000 AND 6000;
```

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

]:

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

Total Number of Successful and Failure Mission Outcomes

List the total number of successful and failure mission outcomes

```
%sql SELECT CASE WHEN "Mission_Outcome" LIKE 'Success%' THEN 'Success' ELSE 'Failure' END AS outcome_type, COUNT(*) AS total FROM SPACEXTBL GROUP BY outcome_type;
```

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

outcome_type	total
Failure	1
Success	100

Boosters Carried Maximum Payload

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

```
%sql SELECT Booster_Version FROM SPACEXTBL WHERE "PAYLOAD_MASS__KG"=( SELECT MAX("PAYLOAD_MASS__KG") FROM SPACEXTBL);
```

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

Booster_Version
F9 v1.0 B0003
F9 v1.0 B0004
F9 v1.0 B0005
F9 v1.0 B0006
F9 v1.0 B0007
F9 v1.1 B1003
F9 v1.1
F9 v1.1
F9 v1.1
F9 v1.1
F9 v1.1
F9 v1.1 B1011
F9 v1.1 B1010
F9 v1.1 B1012
F9 v1.1 B1013

2015 Launch Records

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.

```
%sql SELECT  substr("Date", 6, 2) AS Month, "Landing_Outcome", Booster_Version, Launch_Site FROM SPACEXTBL \
    WHERE "Landing_Outcome" LIKE '%Failure%' AND "Landing_Outcome" LIKE '%drone ship%' AND substr("Date", 1, 4) = '2015';
```

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

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

```
%sql SELECT  "Landing_Outcome", COUNT(*) AS total_count FROM SPACEXTBL WHERE "Date" BETWEEN '2010-06-04' AND '2017-03-20' \
    GROUP BY "Landing_Outcome" ORDER BY total_count DESC;
```

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

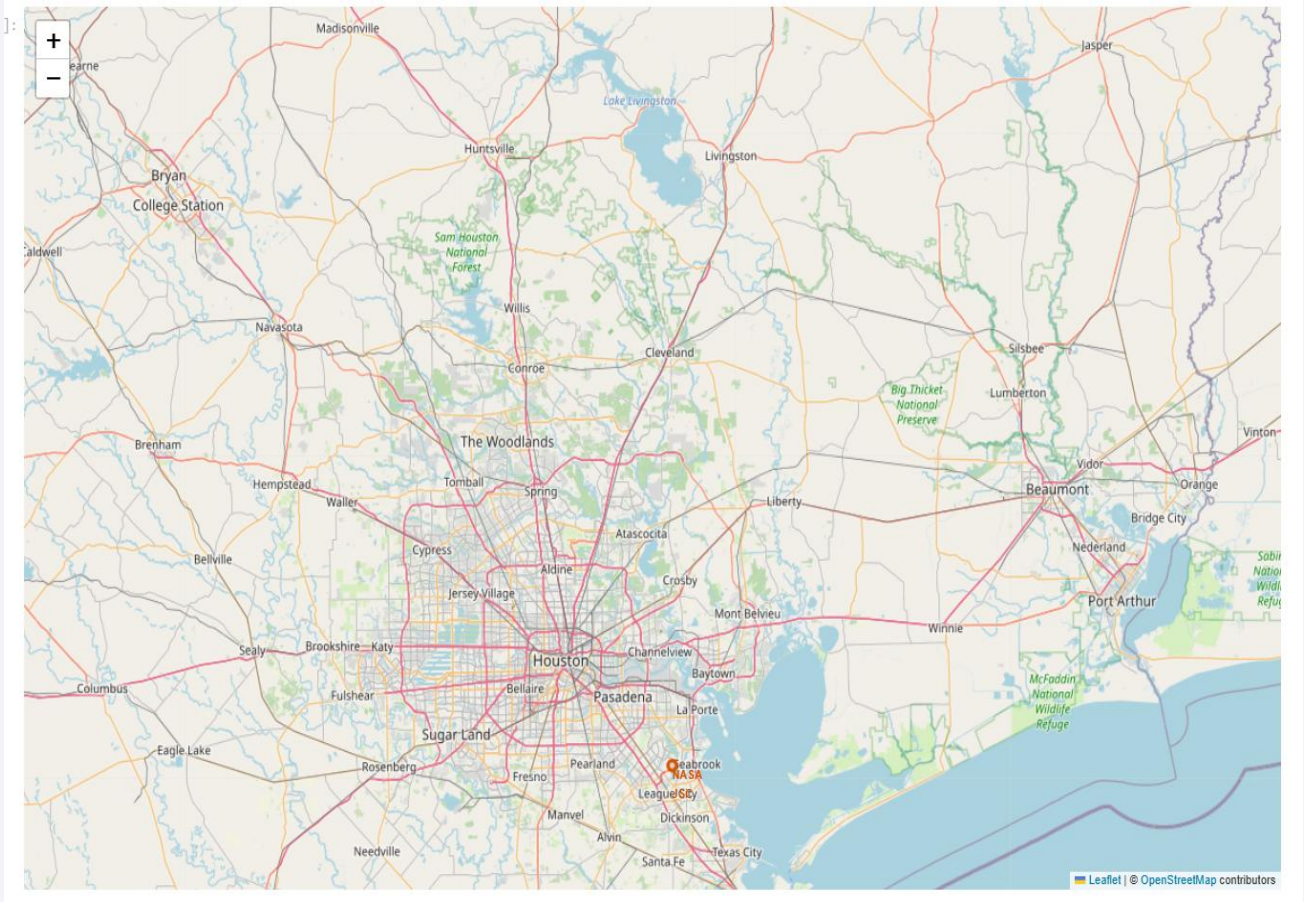
Landing_Outcome	total_count
No attempt	10
Success (drone ship)	5
Failure (drone ship)	5
Success (ground pad)	3
Controlled (ocean)	3
Uncontrolled (ocean)	2
Failure (parachute)	2
Precluded (drone ship)	1

Section 3

Launch Sites Proximities Analysis

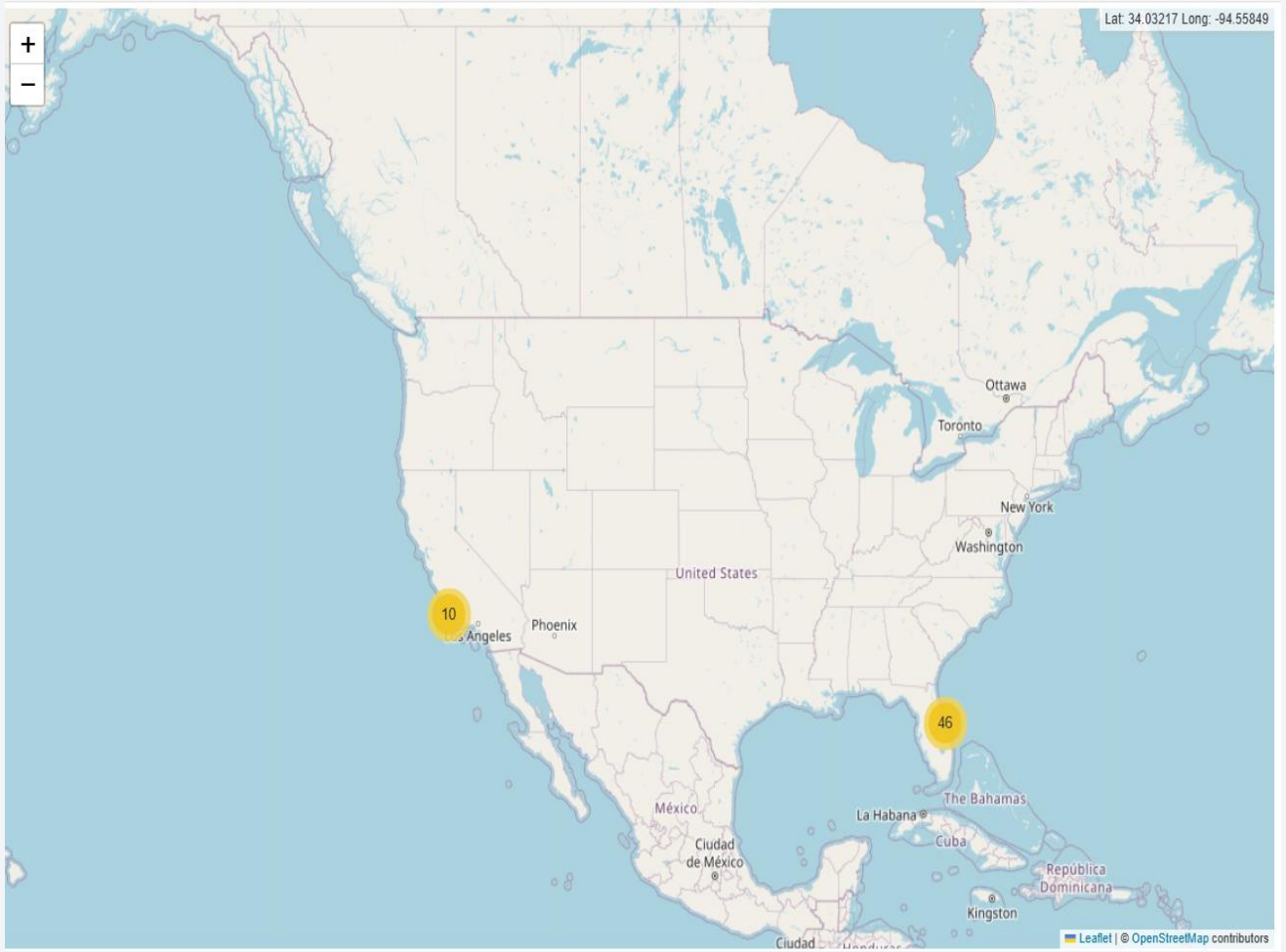
<Folium Map of Launch Site>

- This map shows launch site location of NASA JSC near Houston area.



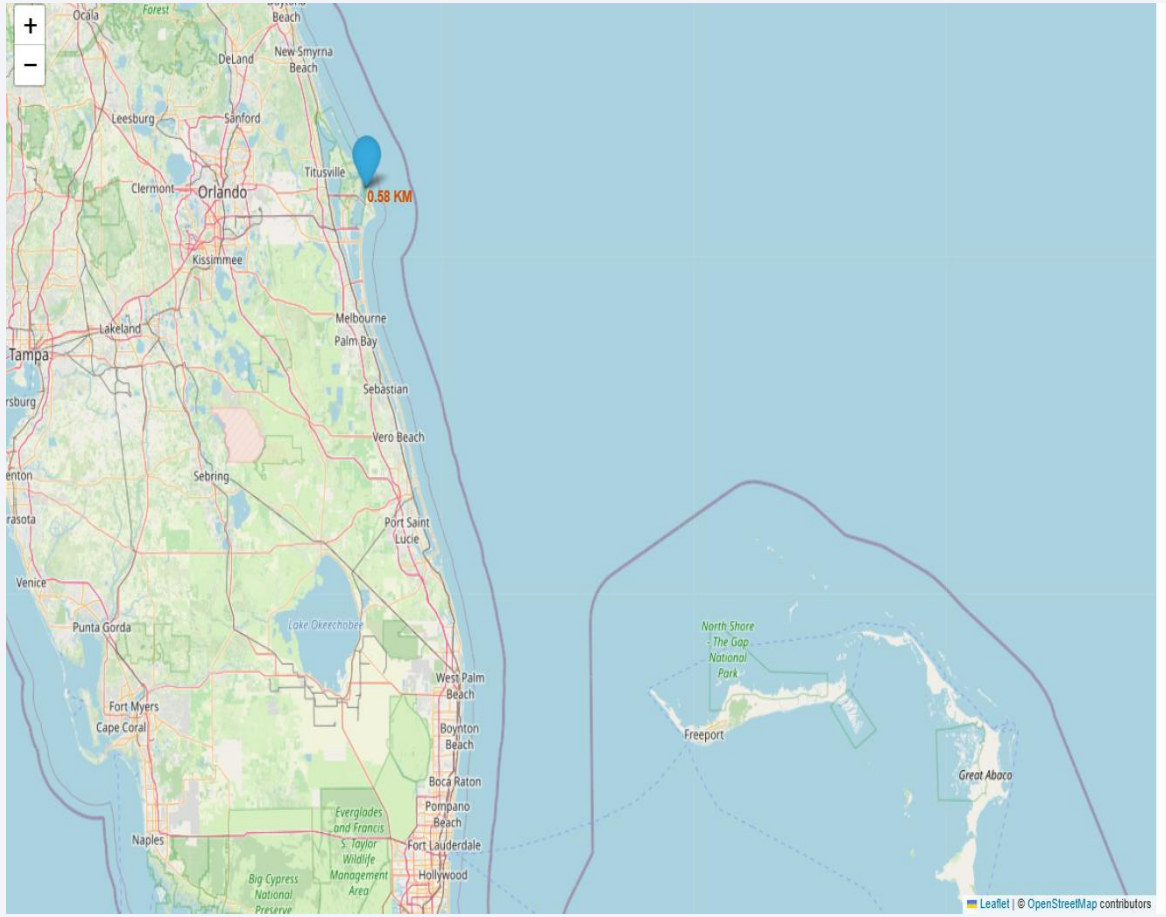
<Folium Map with colored label>

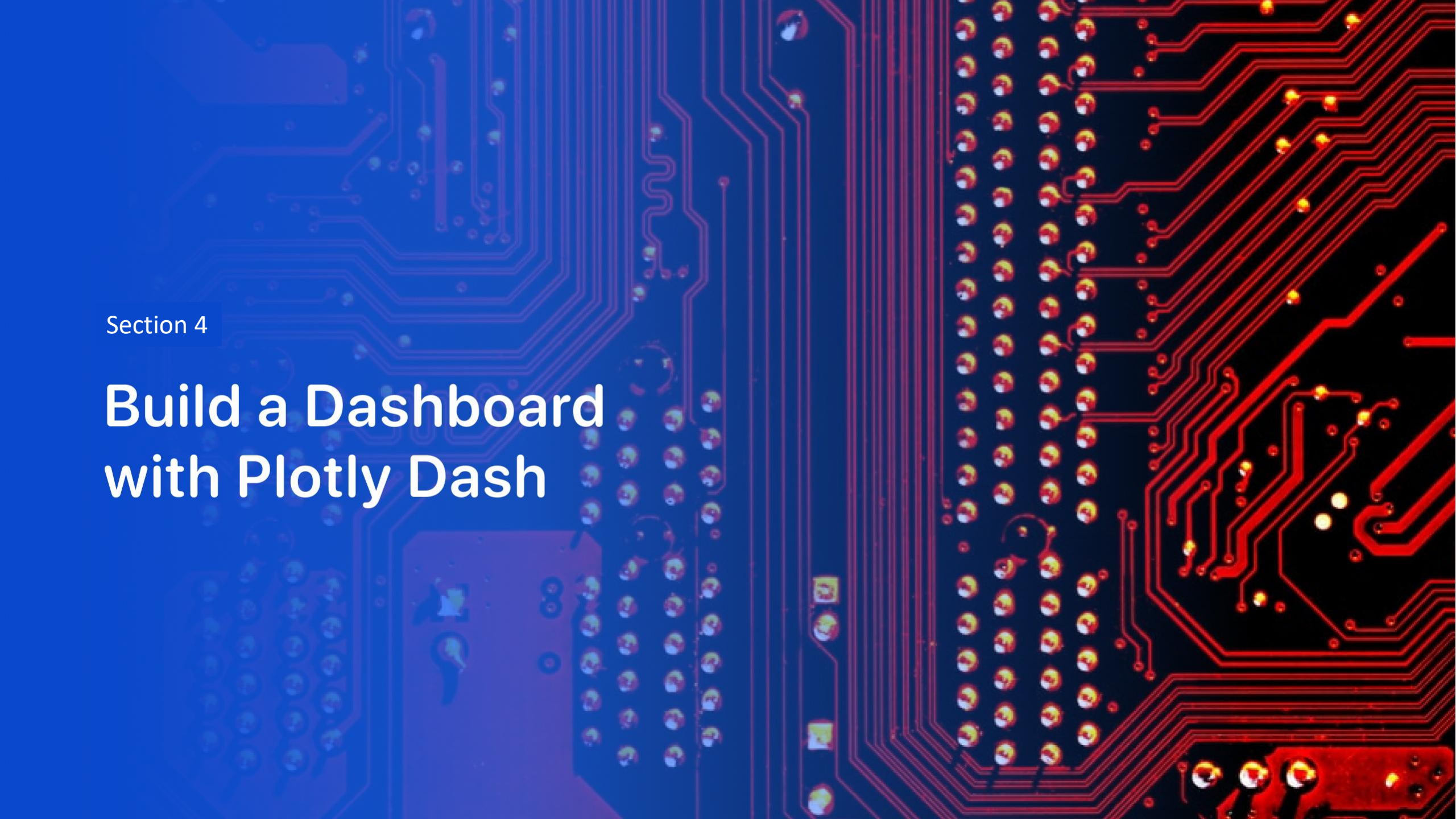
- This map shows colored label of launch sites
- This shows launch sites are near coastal area.



<Folium Map of Lauch Site to its proximities>

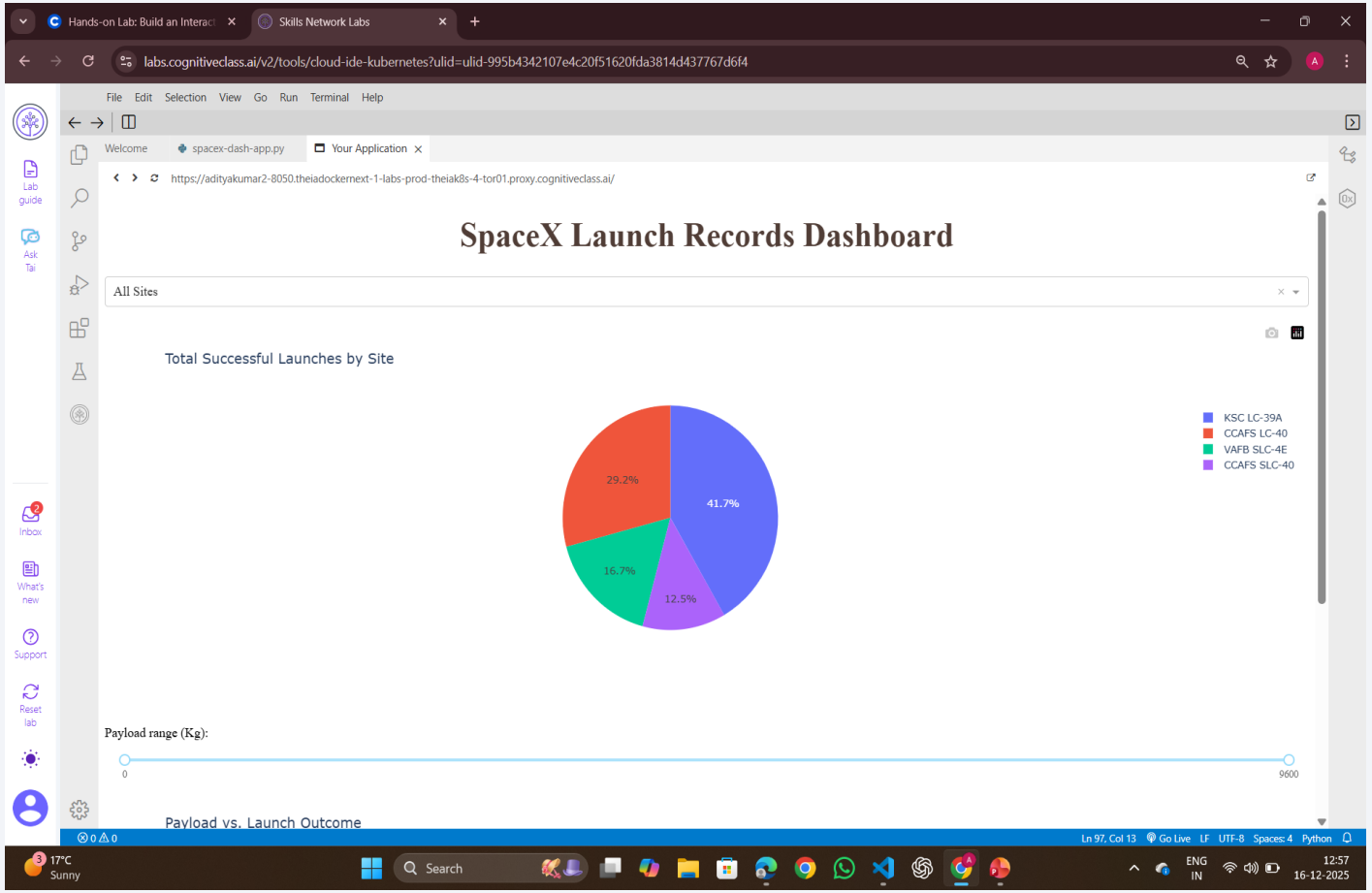
- This map shows a marker with distance to a closest city, railway, highway,




Section 4
Build a Dashboard with Plotly Dash

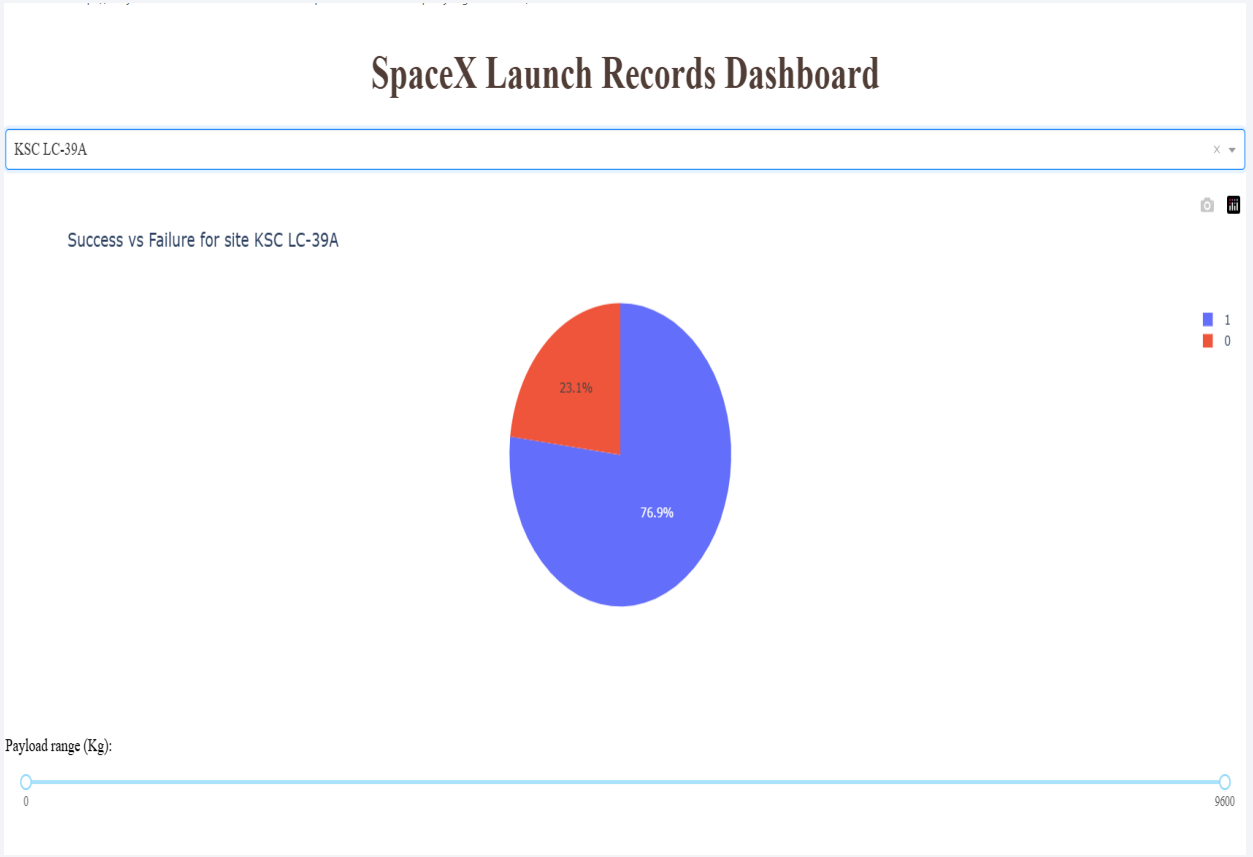
<Launch Success Count of all Sites>

- This pie-chart shows total successful launches by all sites.
- The KSC LC-39A has most successful launches than other sites.



<Launch Site with highest success ratio>

- KSC LC-39A has highest success ratio.
- The success percentage is 76.9% whereas failure percentage is 23.1%.



< Payload vs. Launch Outcome scatter plot for all sites >

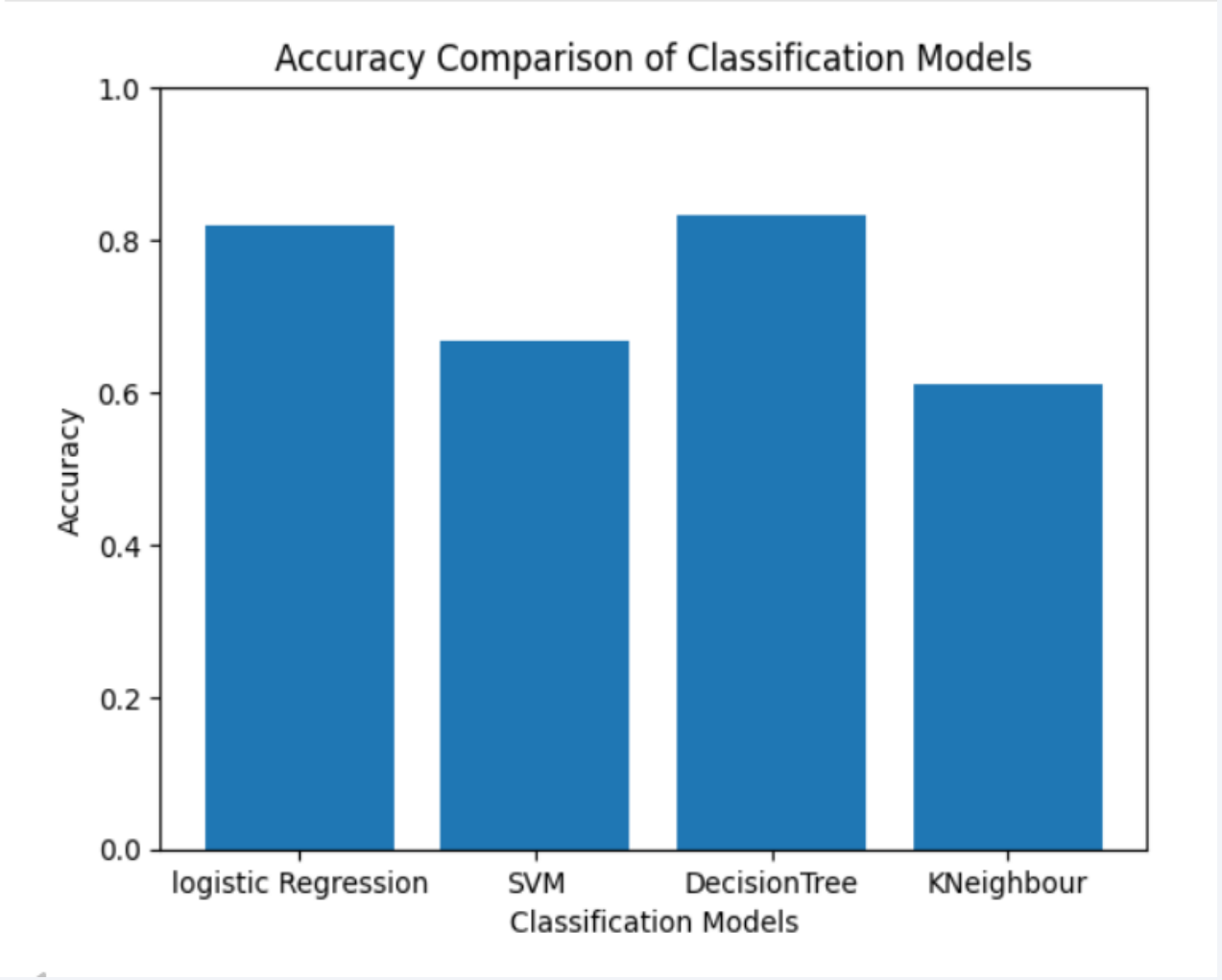
- The payload range (0-9600) has highest success Launch outcome.
- Booster Version FT has highest success Launch outcome.

Section 5

Predictive Analysis (Classification)

Classification Accuracy

- Decision Tree Classifier Model has highest accuracy that is 0.83334



Confusion Matrix

Best performing model

TASK 12

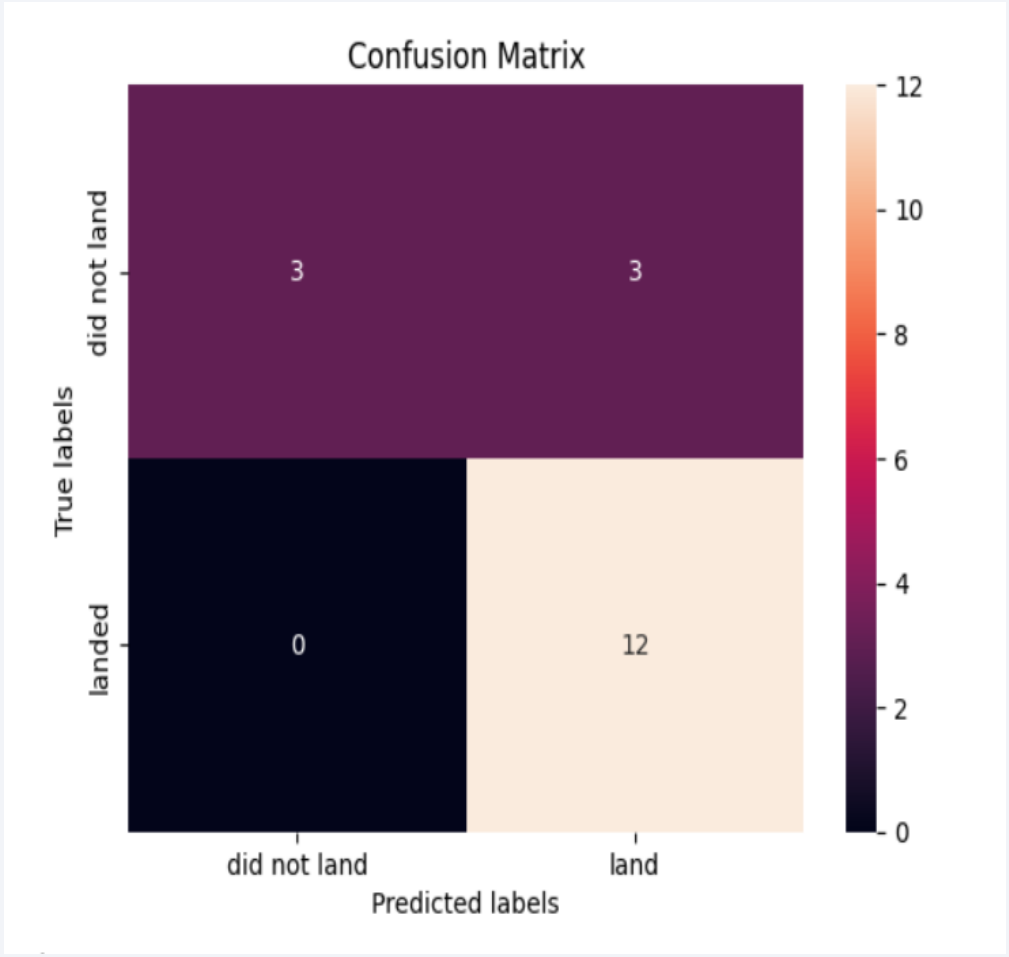
Find the method performs best:

```
models = {
    'Logistic Regression': logreg_cv.best_score_,
    'SVM': svm_cv.best_score_,
    'DecisionTree': tree_cv.best_score_,
    'KNeighbour':knn_cv.best_score_
}

best_method = max(models, key=models.get)
print("Best Method:", best_method)

```

```
Best Method: DecisionTree
```



Conclusions

- Higher the flight number at launch site, higher is success rate
- Launch success rate is found to be improving after each year.
- Orbits ES-L1,GEO, HEO, SSO, VLEO had high success rate
- KSC LC-39A had most success rate than any launch sites
- Decision Tree Classifier is found to be most accurate predictive model.

Appendix

- Every work is done with python language and in EDA we have used SQL too.
- Example of extracted dataset includes is :

spacex_launch_dash.csv > data

```
,Flight Number,Launch Site,class,Payload Mass (kg),Booster Version,Booster Version Category
0,1,CCAFS LC-40,0,0,F9 v1.0  B0003,v1.0
1,2,CCAFS LC-40,0,0,F9 v1.0  B0004,v1.0
2,3,CCAFS LC-40,0,525,F9 v1.0  B0005,v1.0
3,4,CCAFS LC-40,0,500,F9 v1.0  B0006,v1.0
4,5,CCAFS LC-40,0,677,F9 v1.0  B0007,v1.0
5,7,CCAFS LC-40,0,3170,F9 v1.1,v1.1
6,8,CCAFS LC-40,0,3325,F9 v1.1,v1.1
7,9,CCAFS LC-40,0,2296,F9 v1.1,v1.1
8,10,CCAFS LC-40,0,1316,F9 v1.1,v1.1
9,11,CCAFS LC-40,0,4535,F9 v1.1,v1.1
10,12,CCAFS LC-40,0,4428,F9 v1.1 B1011,v1.1
11,13,CCAFS LC-40,0,2216,F9 v1.1 B1010,v1.1
12,14,CCAFS LC-40,0,2395,F9 v1.1 B1012,v1.1
13,15,CCAFS LC-40,0,570,F9 v1.1 B1013,v1.1
14,16,CCAFS LC-40,0,4159,F9 v1.1 B1014,v1.1
15,17,CCAFS LC-40,0,1898,F9 v1.1 B1015,v1.1
16,18,CCAFS LC-40,0,4707,F9 v1.1 B1016,v1.1
17,19,CCAFS LC-40,1,1952,F9 v1.1 B1018,v1.1
18,20,CCAFS LC-40,1,2034,F9 FT B1019,FT
```

Thank you!