IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Jan Baijens
July 17th 2023

Outline

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

Executive Summary

Data collection using webscraping and SpaceX API

Data cleaning and preprocessing using Python

Data exploration and visualization using Python and SQL

Graphical data visualization using Folium, Dash

Classification model generation and predictions using Sklearn

Successful landing of SpaceX Falcon 9 rocket boosters can be accurately predicted using multiple features

Introduction

- SpaceX can reuse rocket boosters by safely landing them after use
- Reuse lowers costs, improves sustainability
- Therefore, higher landing success rates are highly beneficial
- How to predict successful landing?
- Classification models trained on real data

Section 1

Methodology

Methodology

Executive Summary

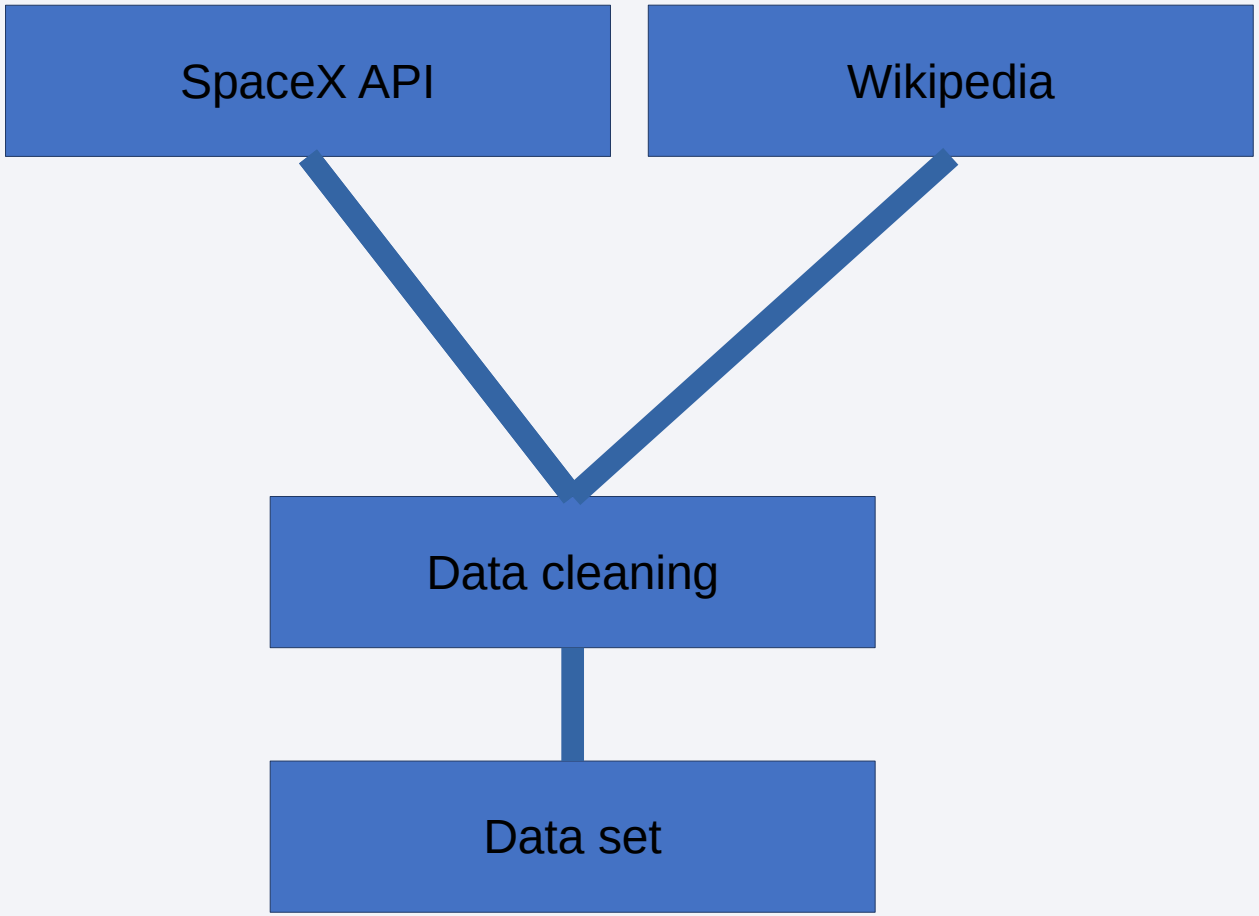
- Data collection methodology:
  - SpaceX API
- Perform data wrangling
  - Cleaned data and classified by landing outcome
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Models predict succesful landing based on available features

Data Collection

Data collected from SpaceX API and web scraping from Wikipedia

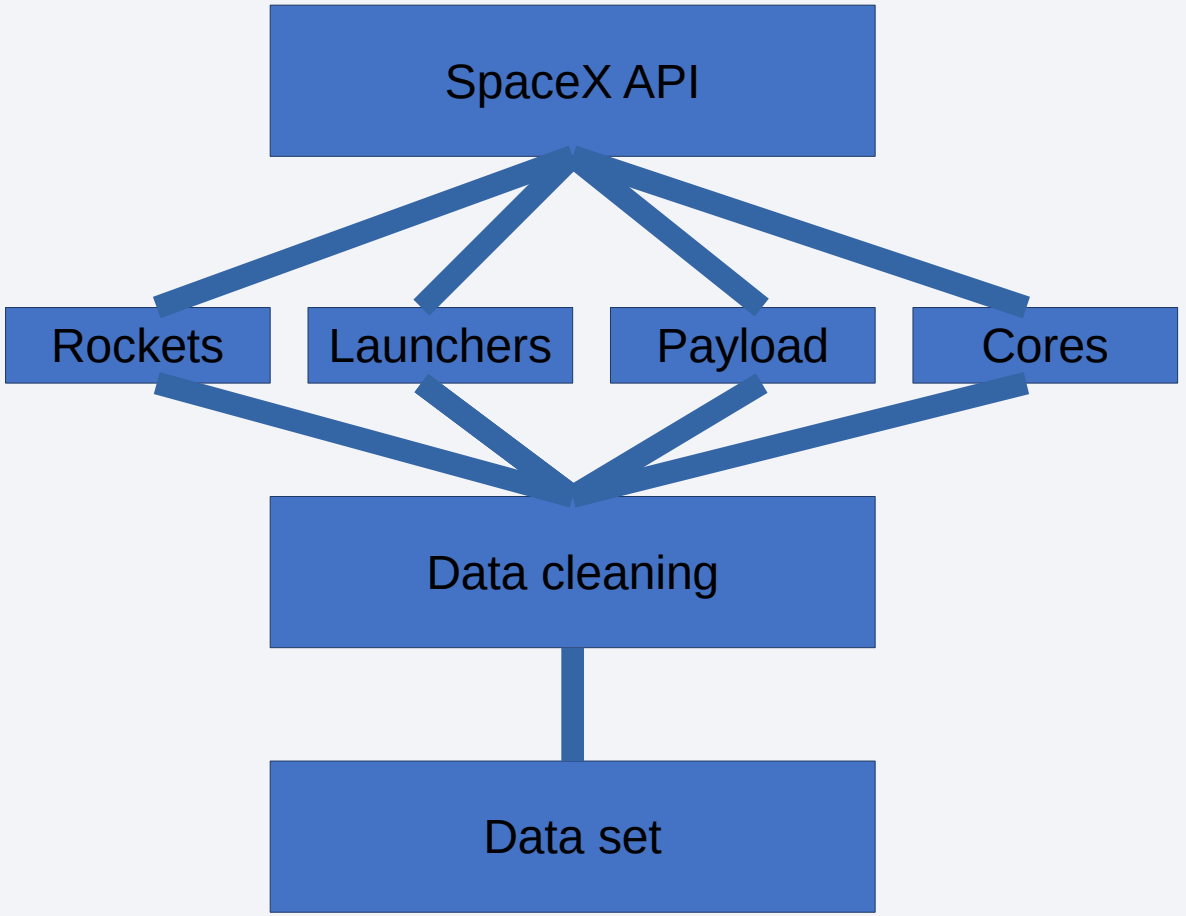
Filtered for Falcon9 launches

Missing values removed



Data Collection – SpaceX API

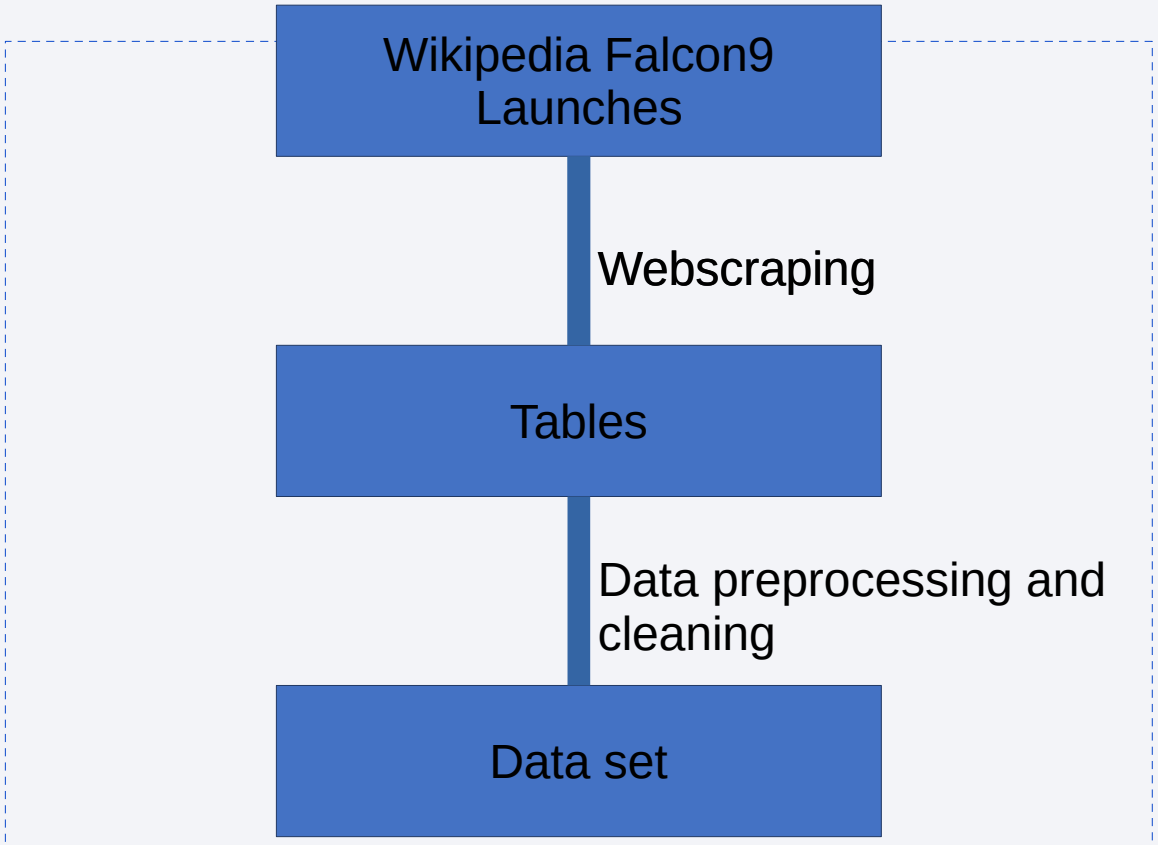
- Special REST API call for four datasets on SpaceX API
- Rockets, Launchers, Payload, Cores
- Stored data using Python Pandas DataFrame
-



https://github.com/JBaijens/IBM_Data_Science/blob/main/jupyter-labs-spacex-data-collection-api.ipynb

Data Collection - Scraping

- Data from tables in wikipedia Falcon9 launches page
- Webscraping using beautifulsoup



https://github.com/JBaijens/IBM_Data_Science/blob/main/jupyter-labs-webscraping.ipynb

Data Wrangling

Number of launches per site calculated

Number and occurrence of orbits calculated

Number of succesful and failed missions per orbit type calculated

Landing outcome label generated per flight

https://github.com/JBaijens/IBM_Data_Science/blob/main/labs-jupyter-spacex-data_wrangling_jupyterlite.jupyterlite.ipynb

EDA with Data Visualization

- Launch Site, Payloadmass per flight number
  - See trend over time
- Relationship between Payload and Launch Site
  - See differences in launch sites
- Relationship between success rate of each orbit type
  - See effect of orbit type on successful landing
- Relationship between Payload and Orbit type
  - See differences in orbit types
- Launch succes yearly trend to see improvement over time

https://github.com/JBaijens/IBM_Data_Science/blob/main/jupyter-labs-eda-dataviz.ipynb

EDA with SQL (1)

- Select unique launch site names
- 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 succesful 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

https://github.com/JBaijens/IBM_Data_Science/blob/main/jupyter-labs-eda-sql-coursera_sqllite.ipynb

EDA with SQL (2)

- List the total number of successful and failure mission outcomes
- List the names of the booster_versions which have carried the maximum payload mass. Use a subquery
- 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.

https://github.com/JBaijens/IBM_Data_Science/blob/main/jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

Added circles and markers to launch sites

These show mission landing succes and number of missions per launch site

Provide a graphical intuition of launches and successful landing

https://github.com/JBaijens/IBM_Data_Science/blob/main/lab_jupyter_launch_site_location.ipynb

Build a Dashboard with Plotly Dash

Added pie chart of number of launches, percentage successful launches per launch site

To see which launch site has the highest succes rate, highest number of successful landings

Added scatterplot of payload vs landing success

See which payload range has the highest success rate

https://github.com/JBaijens/IBM_Data_Science/blob/main/spacex_dash_app.py

Predictive Analysis (Classification)

Performed feature engineering, normalized the data using one-hot-encoding

Standardized the data using Sklearn preprocessing StandardScaler

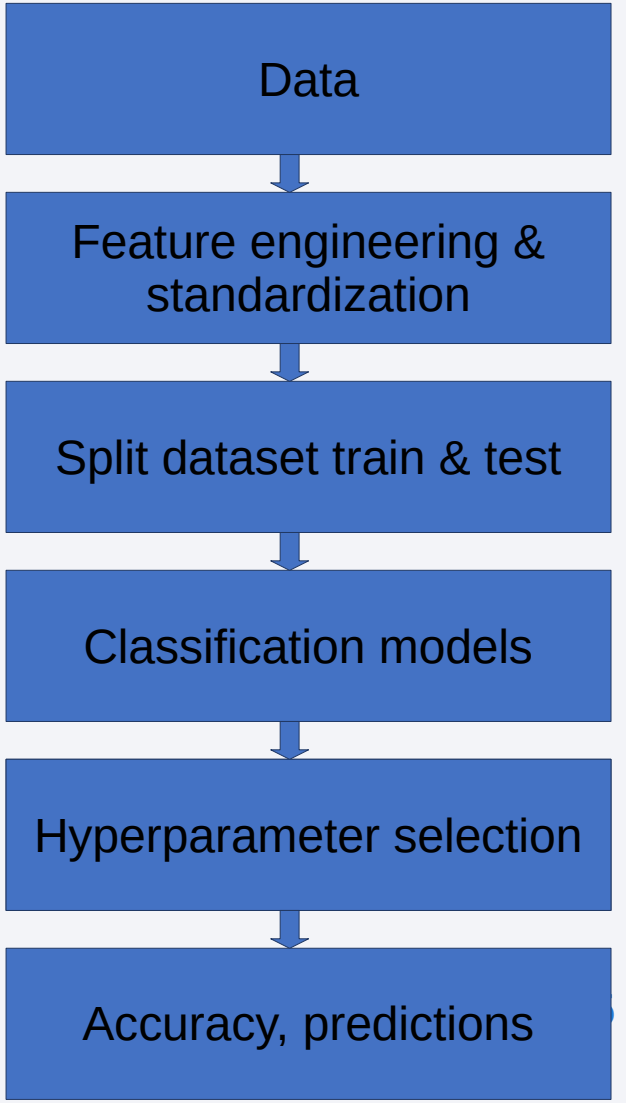
Split the data into training and test sets

Created classification models: logistic regression, SVM, decision tree, KNN

Selected hyperparameters using grid search

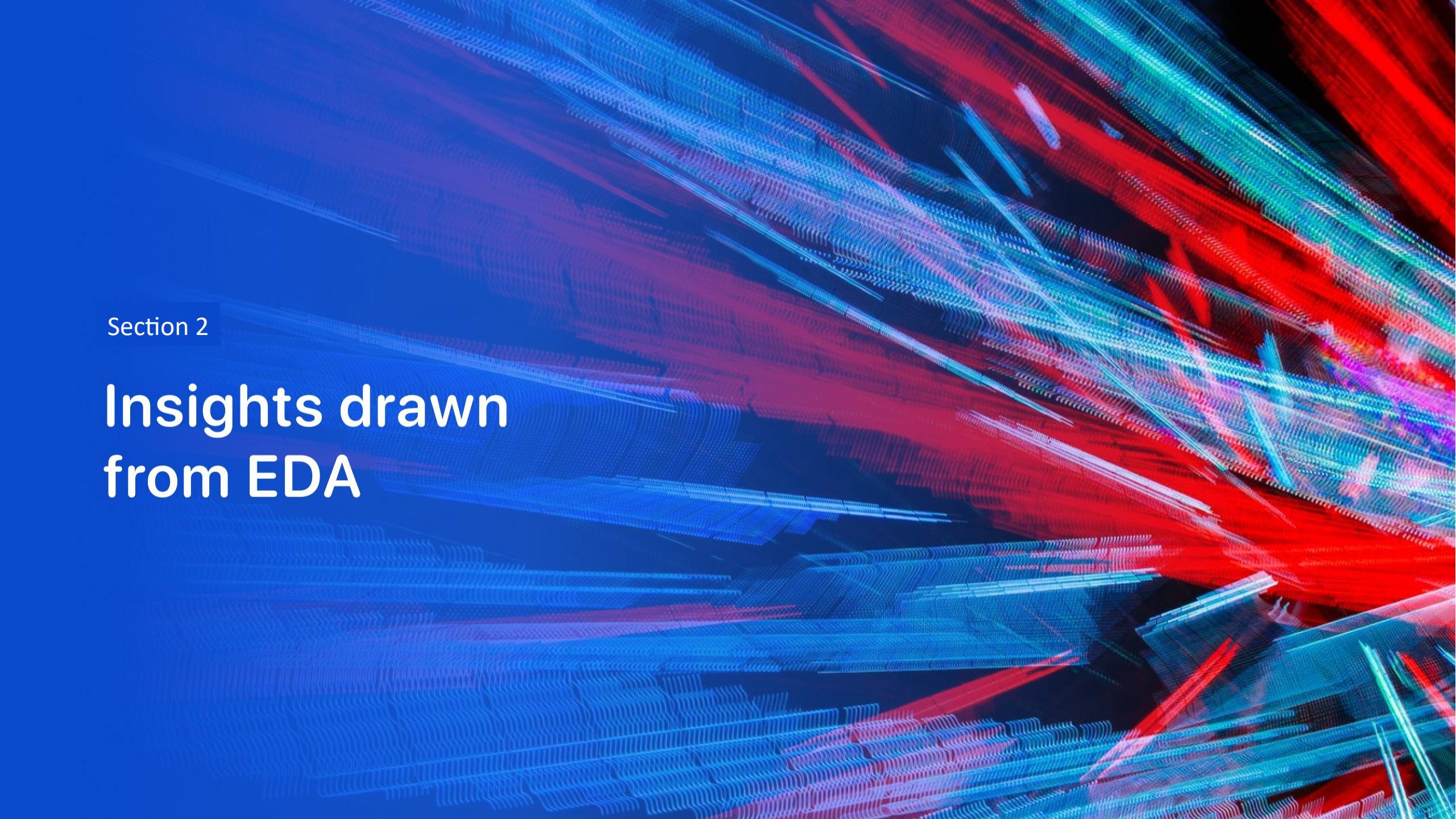
Calculated accuracy, made confusion plots

https://github.com/JBaijens/IBM_Data_Science/blob/main/SpaceX_Machine_Learning_Prediction_Part_5.jupyterlite.ipynb

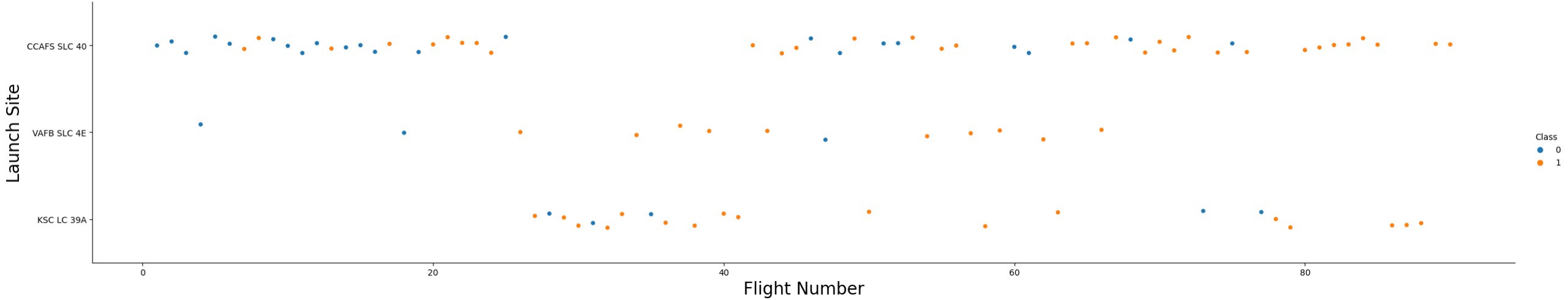


Results

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


Section 2
Insights drawn from EDA

Flight Number vs. Launch Site

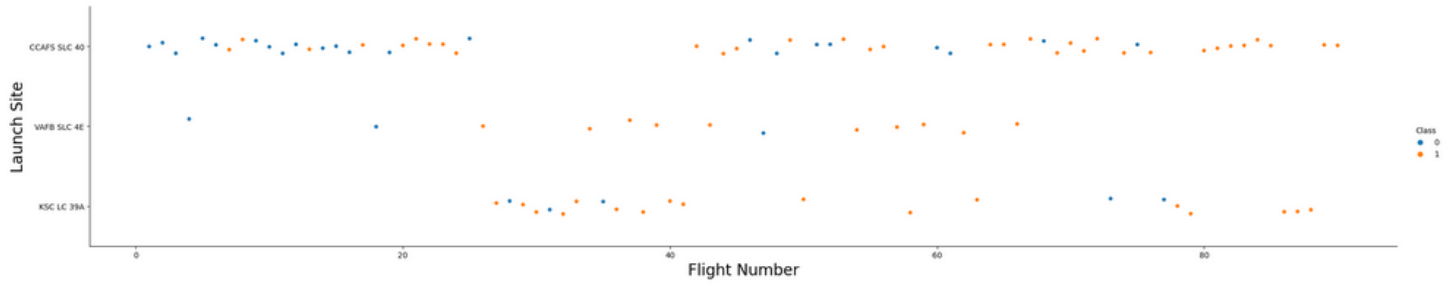


- Early flight numbers had more landing failures than later flight numbers
- KSC location was used latest

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'`

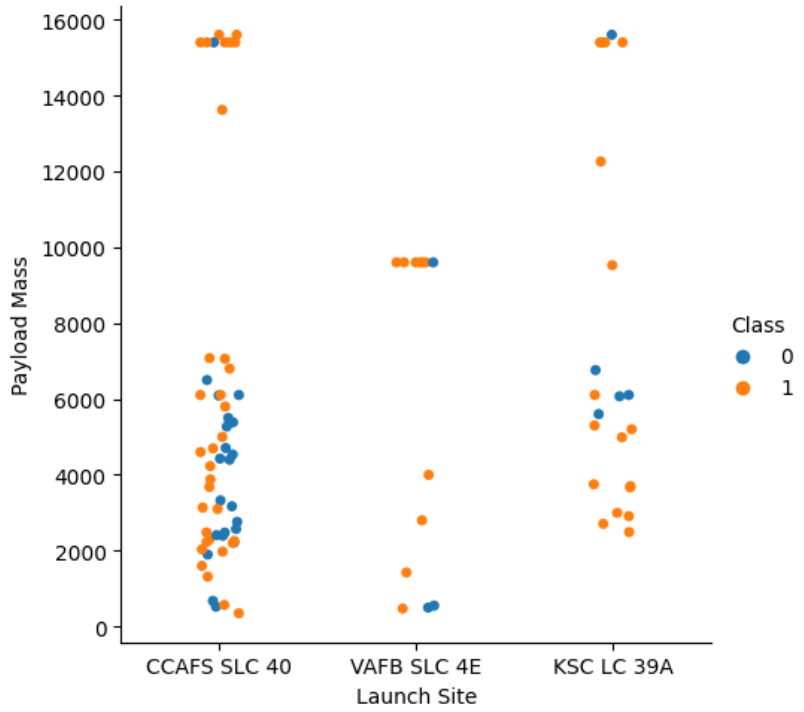
```
In [5]: # Plot a scatter point chart with x axis to be Flight Number and y axis to be the launch site, and hue to be the class value
sns.catplot(y="LaunchSite", x="FlightNumber", hue="Class", data=df, aspect = 5)
plt.xlabel("Flight Number",fontsize=20)
plt.ylabel("Launch Site",fontsize=20)
plt.show()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



Payload vs. Launch Site

- VAFB launch site does not include the highest payload masses
- CCAFS launch site include both many high and low payload masses
- Plot generated with seaborn

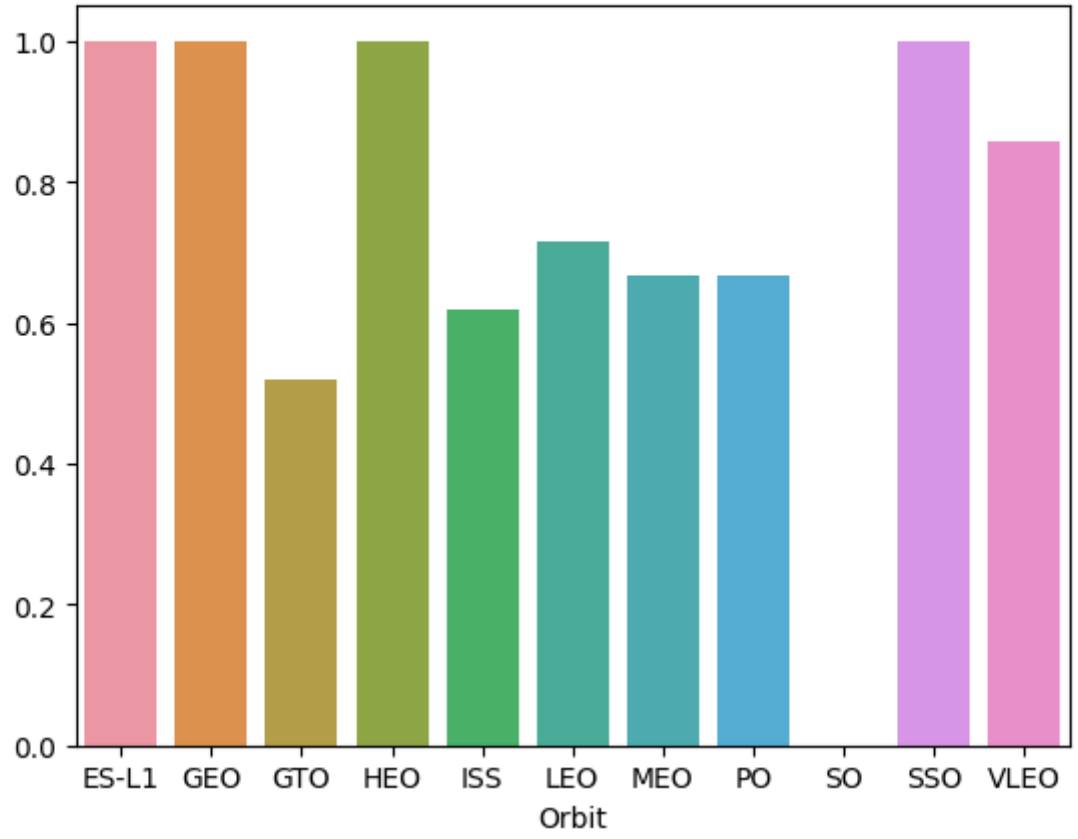
```
In [8]: # Plot a scatter point chart with x axis to be Pay Load Mass (kg) and y axis to be the launch site, and hue to be the class
sns.catplot(y="PayloadMass", x="LaunchSite", hue="Class", data=df)
plt.xlabel("Launch Site")
plt.ylabel("Payload Mass")
plt.show()
```



Success Rate vs. Orbit Type

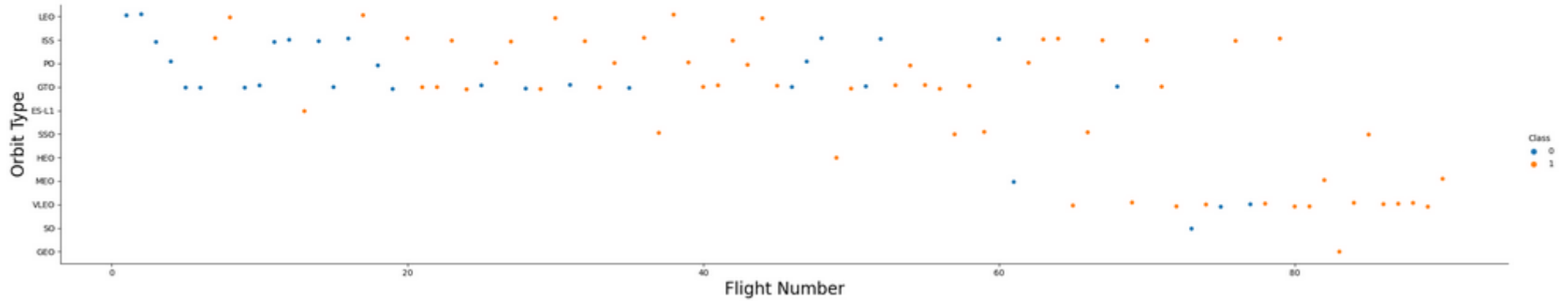
- Highest succes rates for ES-L1, GEO, HEO, ans SSO orbits
- No successes for SO orbit

```
In [12]: # HINT use groupby method on Orbit column and get the mean of Class column
         orbit_success = df.groupby('Orbit').Class.mean()
         sns.barplot(x = orbit_success.index, y = orbit_success.values)
Out[12]: <Axes: xlabel='Orbit'>
```



Flight Number vs. Orbit Type

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



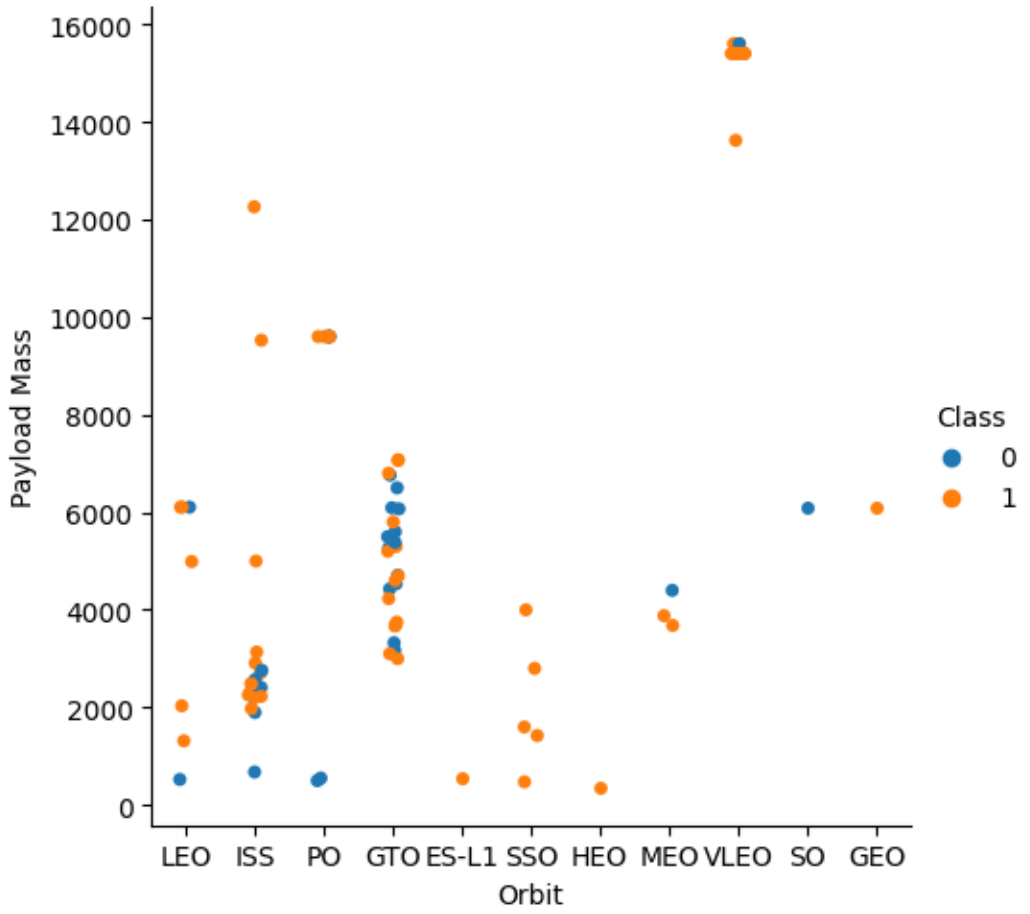
- Points in blue indicate failed landings, orange indicate successful landings

- First flight numbers were mainly lower orbits (LEO, ISS), later flight numbers include higher orbits

Payload vs. Orbit Type

- Distinct differences in payload mass between different orbit types
- Very low earth orbit (VLEO) has the highest payloads

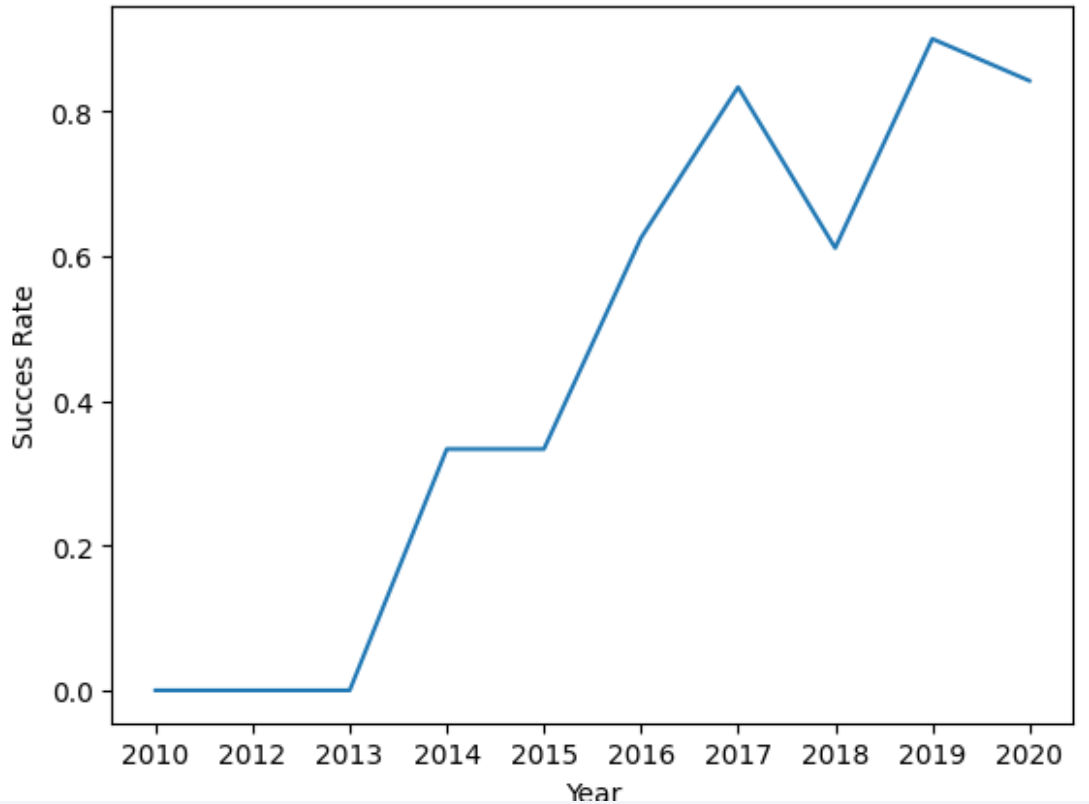
```
# Plot a scatter point chart with x axis to be Payload and y axis to be the Orbit, and hue to be the class value
sns.catplot(y="PayloadMass", x="Orbit", hue="Class", data=df)
plt.xlabel("Orbit")
plt.ylabel("Payload Mass")
plt.show()
```



Launch Success Yearly Trend

- No successful landings in the first three years
- High success rates in later years

```
# Plot a line chart with x axis to be the extracted year and y axis to be
year=[]
year = Extract_year(df)
df['Year'] = year
year_succes = df.groupby('Year').Class.mean()
sns.lineplot(x = year_succes.index, y = year_succes.values)
plt.ylabel('Succes Rate')
plt.show()
```



All Launch Site Names

Find the names of the unique launch sites

This is done using a simple SQL query

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

In [11]:
```
%%sql
SELECT DISTINCT Launch_Site FROM SPACEXTBL
```

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

Out[11]:

Launch_Site
CCAFS LC-40
VAFB SLC-4E
KSC LC-39A
CCAFS SLC-40
None

Launch Site Names Begin with 'CCA'

Find 5 records where launch sites begin with `CCA`

Names starting with CCA are selected using string operation command LIKE

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

```
%%sql
SELECT Launch_Site FROM SPACEXTBL WHERE Launch_Site LIKE 'CCA%' LIMIT 5
```

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

Launch_Site
CCAFS LC-40
CCAFS LC-40
CCAFS LC-40
CCAFS LC-40
CCAFS LC-40

Total Payload Mass

Calculate the total payload carried by boosters from NASA

The sum of payload mass from NASA (CRS) is 45596 KG

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

```
%%sql
SELECT Customer, SUM("PAYLOAD_MASS__KG_") FROM SPACEXTBL WHERE Customer = 'NASA (CRS)'
```

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

Customer	SUM("PAYLOAD_MASS__KG_")
NASA (CRS)	45596.0

Average Payload Mass by F9 v1.1

The average payload mass carried by booster version F9 is 2928.4 Kg

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

```
%%sql
SELECT Booster_Version, AVG("PAYLOAD_MASS__KG_") FROM SPACEXTBL WHERE Booster_Version = 'F9 v1.1'
```

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

Booster_Version	AVG("PAYLOAD_MASS__KG_")
F9 v1.1	2928.4

First Successful Ground Landing Date

The dates of the first successful landing outcome on ground pad is 22/12/2015

This is retrieved using a select statement, the results are ordered descending using DESC, and only the first result is returned using LIMIT 1

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

Hint:Use min function

```
%%sql
SELECT Date, Landing_Outcome FROM SPACEXTBL WHERE Landing_Outcome = 'Success (ground pad)' ORDER BY Date DESC LIMIT 1
```

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

Date	Landing_Outcome
22/12/2015	Success (ground pad)

Successful Drone Ship Landing with Payload between 4000 and 6000

F9 FT Booster versions B1021.2, B1022, B1026 and B1031.2 have successfully landed on drone ship and had payload mass greater than 4000 but less than 6000

The results are selected using a Where statement with 3 requirements

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 distinct Booster_Version from SPACEXTBL
where Landing_Outcome = 'Success (drone ship)' and "PAYLOAD_MASS__KG_" > 4000 and "PAYLOAD_MASS__KG_" < 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

Calculate the total number of successful and failure mission outcomes

The results are grouped by mission outcome, and then counted using COUNT

```
%%sql
SELECT Mission_Outcome, COUNT(*) AS count
FROM SPACEXTBL
GROUP BY Mission_Outcome;
```

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

Mission_Outcome	count
None	898
Failure (in flight)	1
Success	98
Success	1
Success (payload status unclear)	1

Boosters Carried Maximum Payload

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

The subquery selects the maximum payload mass

All distinct booster versions which have carried the maximum payload mass are returned

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

```
%%sql
SELECT DISTINCT Booster_Version FROM SPACEXTBL
WHERE "PAYLOAD_MASS__KG_" = (SELECT MAX("PAYLOAD_MASS__KG_") FROM SPACEXTBL)
```

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

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

Substring method is used to handle the date specifications

WHERE is used to select the data with the right date and landing outcome

The results are ordered by month

```
%%sql
SELECT
    substr(Date, 4, 2) AS Month,
    Landing_Outcome,
    Booster_Version,
    Launch_Site
FROM
    SPACEXTBL
WHERE
    substr(Date, 7, 4) = '2015' AND
    Landing_Outcome = 'Failure (drone ship)'
ORDER BY
    Month
```

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

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

The right results are specified using the WHERE statement, grouped by landing outcome, and ordered by the count, which is generated using the COUNT() function

```
%%sql
SELECT
    Landing_Outcome, COUNT(*) AS Count
FROM
    SPACEXTBL
WHERE
    substr(Date, 7, 4) || '-' || substr(Date, 1, 2) || '-' || substr(Date, 4, 2)
    BETWEEN '2010-06-04' AND '2017-03-20'
GROUP BY Landing_Outcome
ORDER BY Count DESC;
```

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

Landing_Outcome	Count
No attempt	9
Failure (drone ship)	5
Success (drone ship)	4
Controlled (ocean)	3
Uncontrolled (ocean)	2
Success (ground pad)	2
Failure (parachute)	2
Precluded (drone ship)	1

Section 3

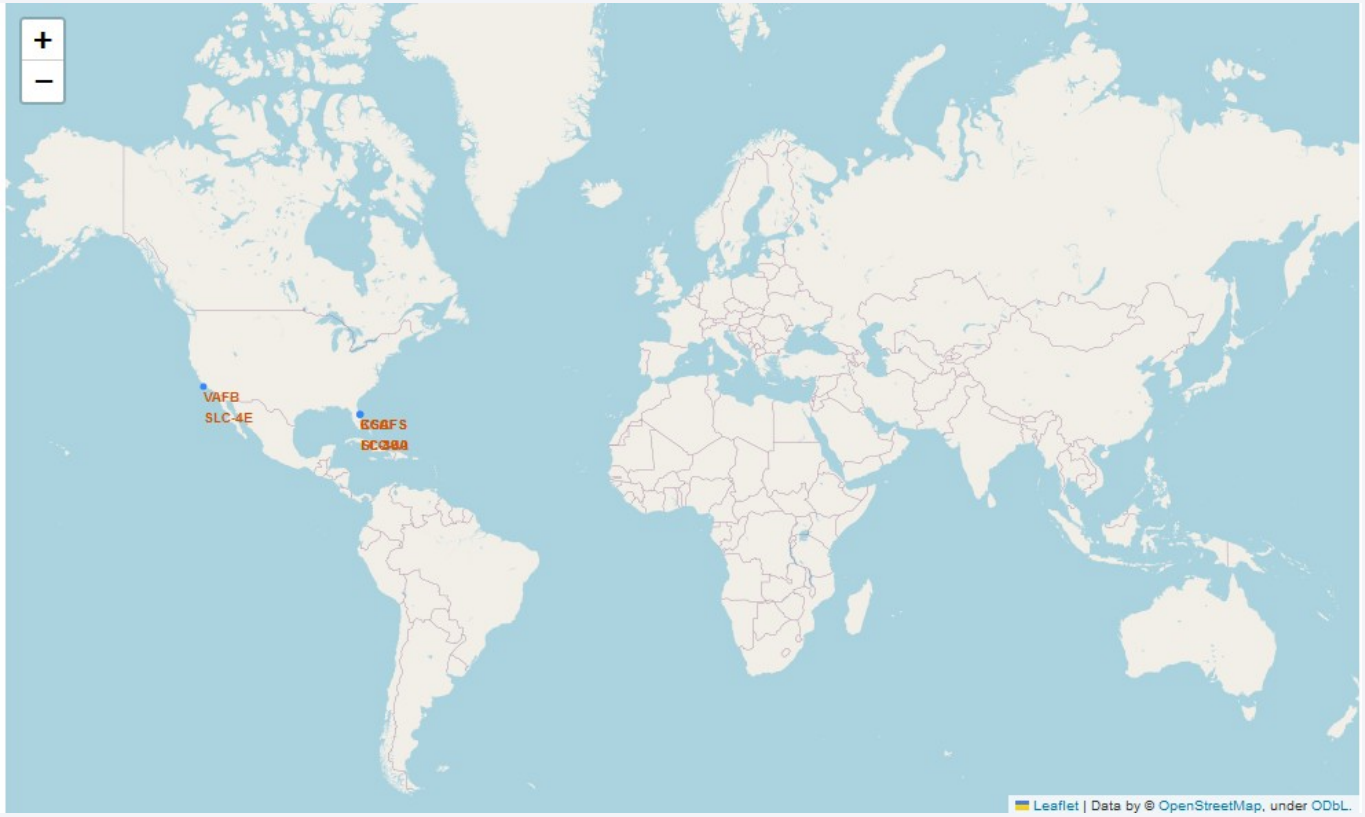
Launch Sites Proximities Analysis

Launch site locations

Launch sites are located in California and Florida, USA

VAFB SLC-4E is located close to the coast in California

KSC LC-39A and CCAFS LC-40 are located next to Cape Canaveral Space force, Florida

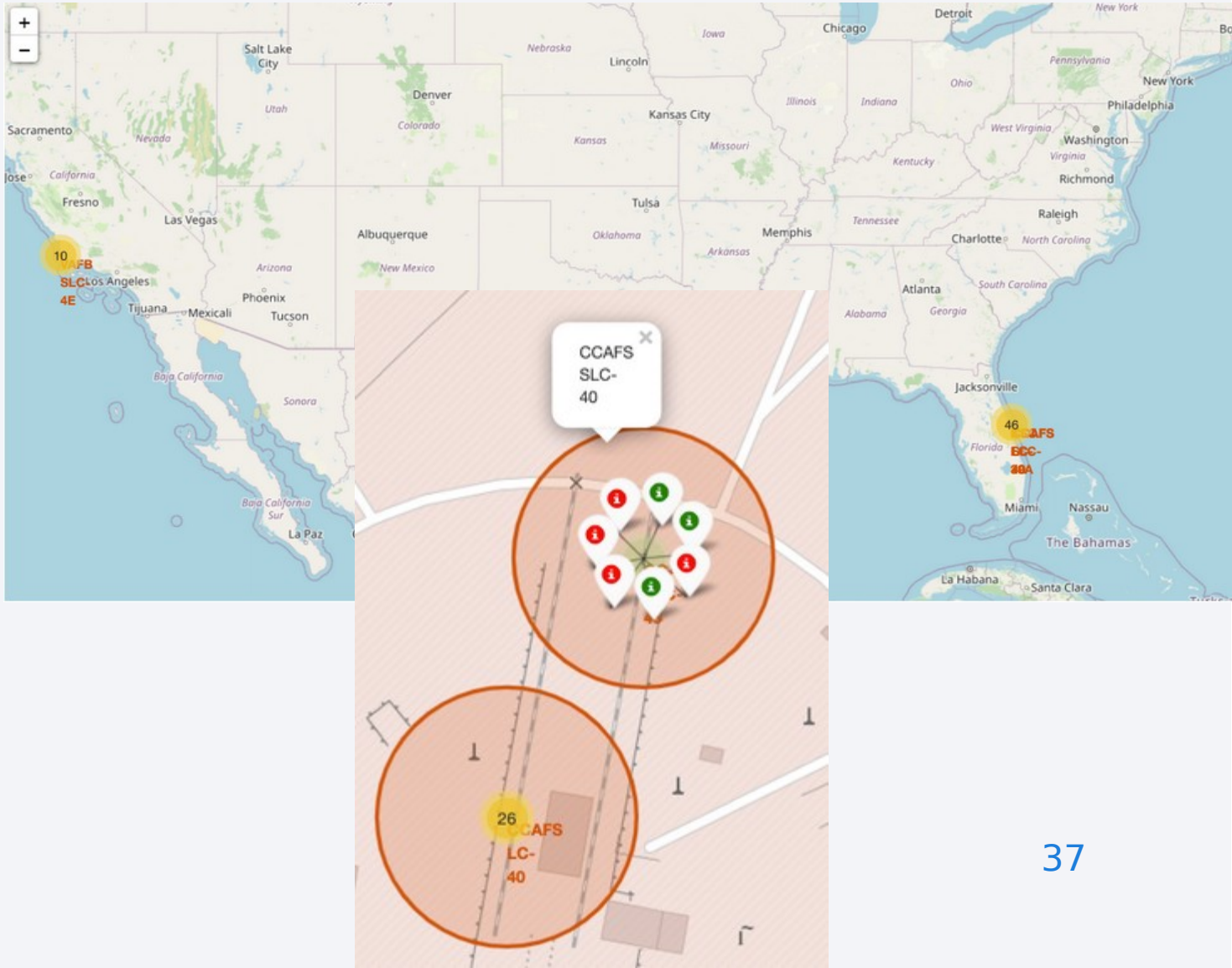


Geographical locations of launch successes and failures

Successful landings are marked green

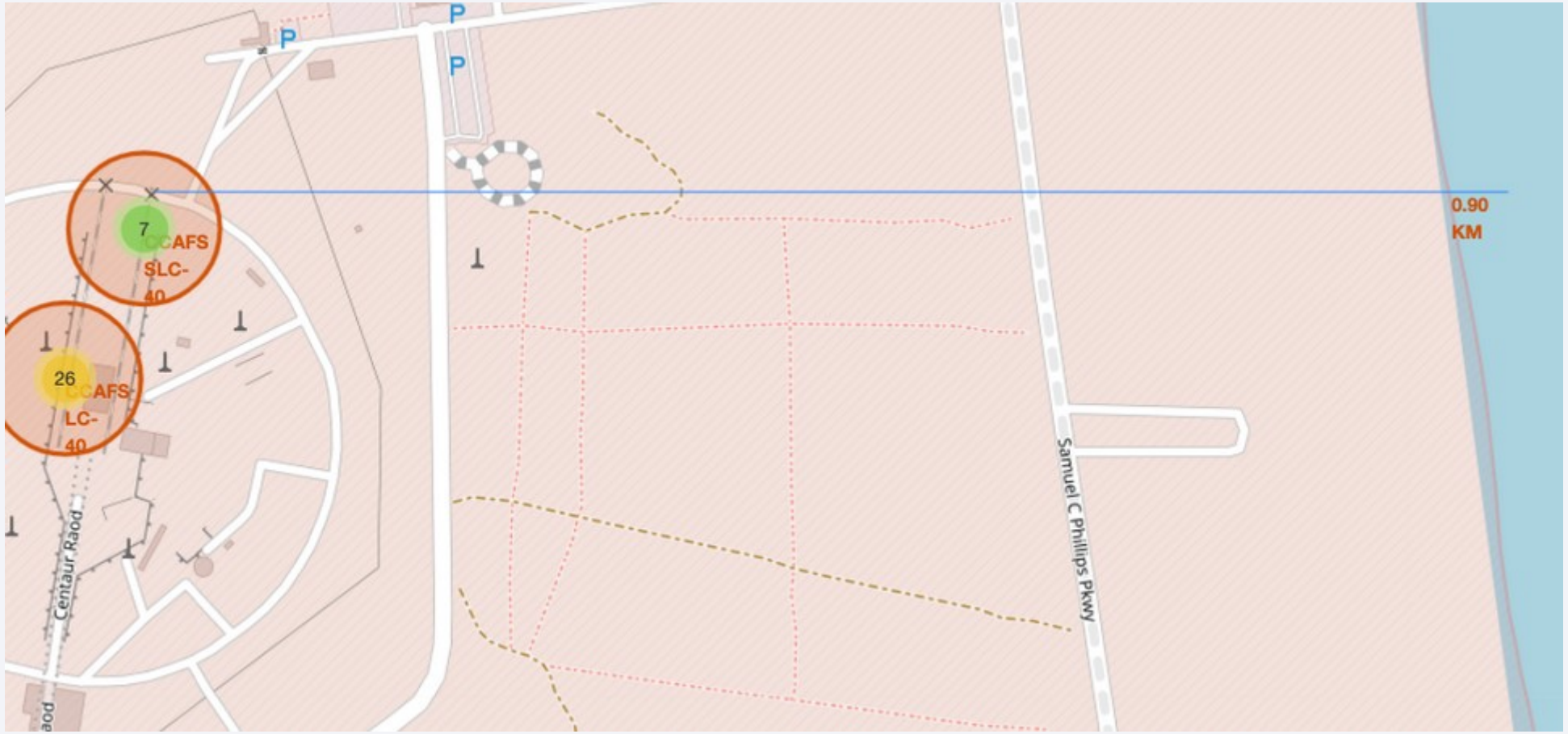
Unsuccessful landings are marked red

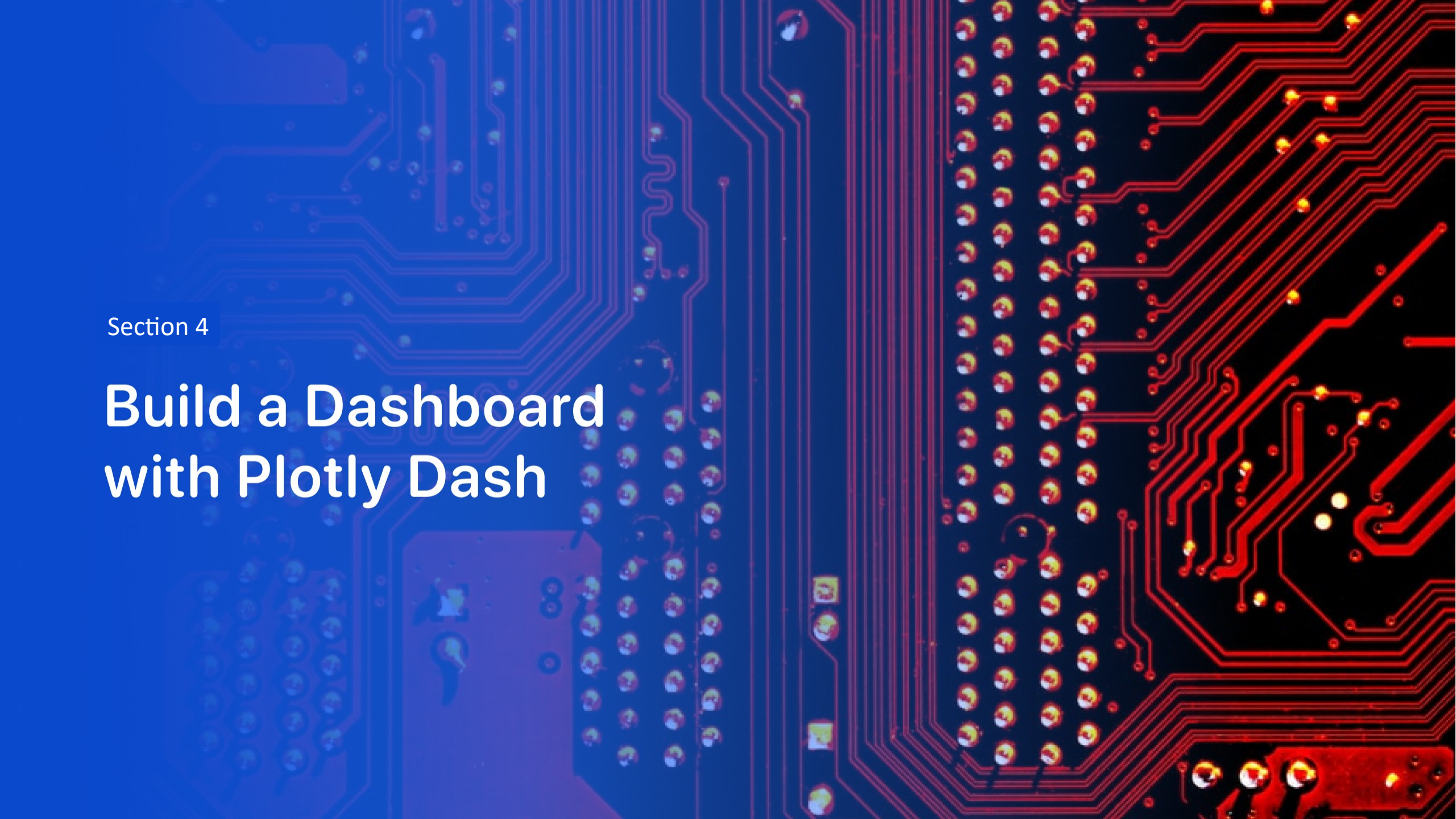
Successful landings are found on all launch site locations



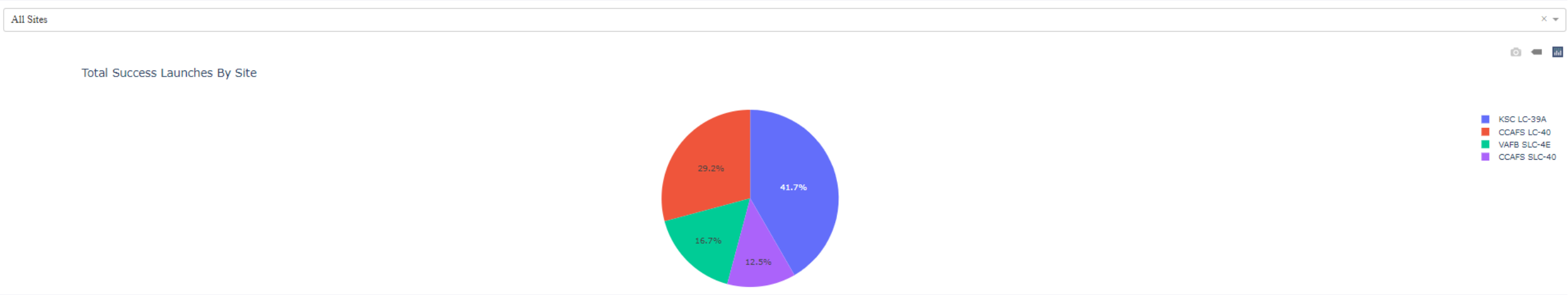
Launch site distances

Distance to coast is 0.9 Kilometers




Section 4
Build a Dashboard with Plotly Dash

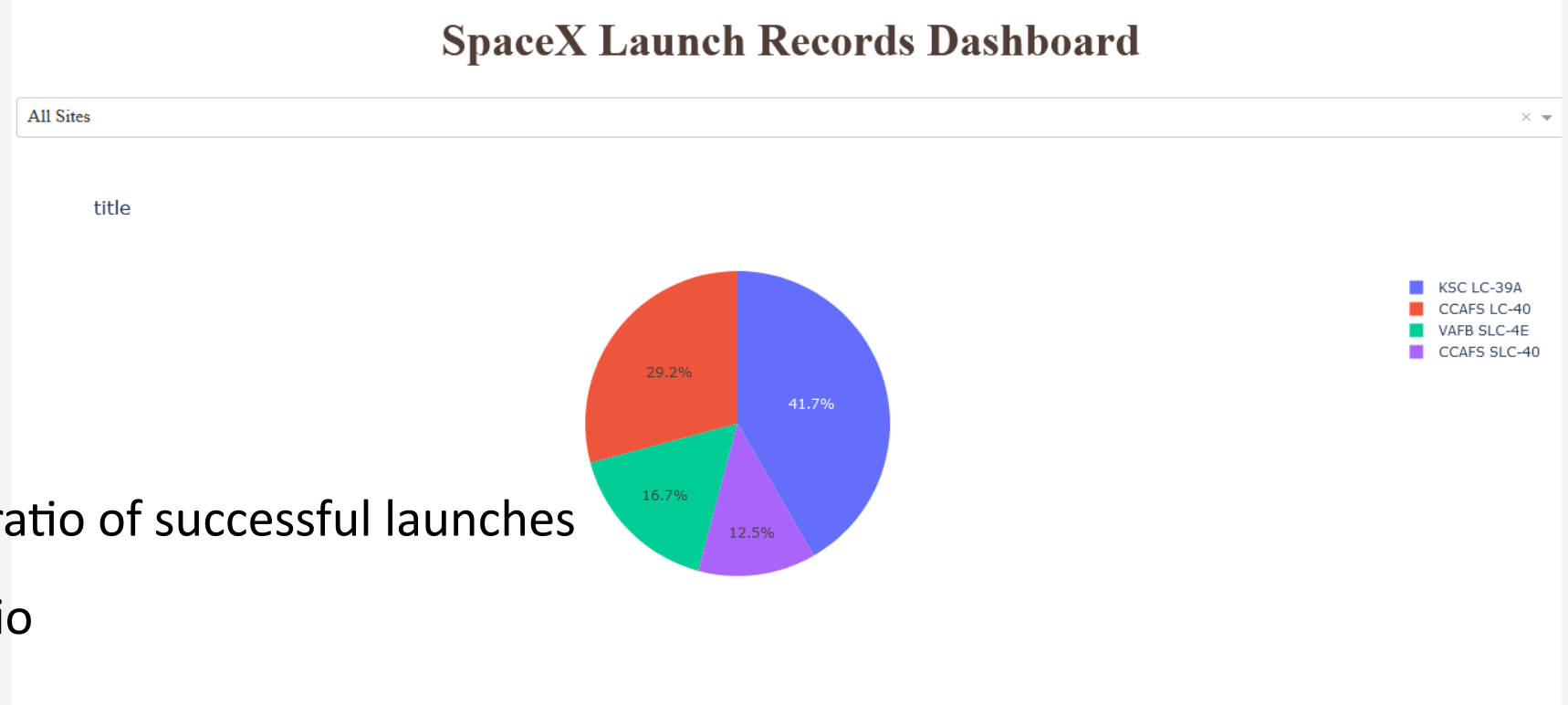
Launch success count per site



KSC LC-39A has the most successful launches

CCAFS SLC-40 the fewest

Launch success rates per launch site



KSC LC-39A has the highest ratio of successful launches

CCAFS SLC-40 the lowest ratio

Relation of payload and success

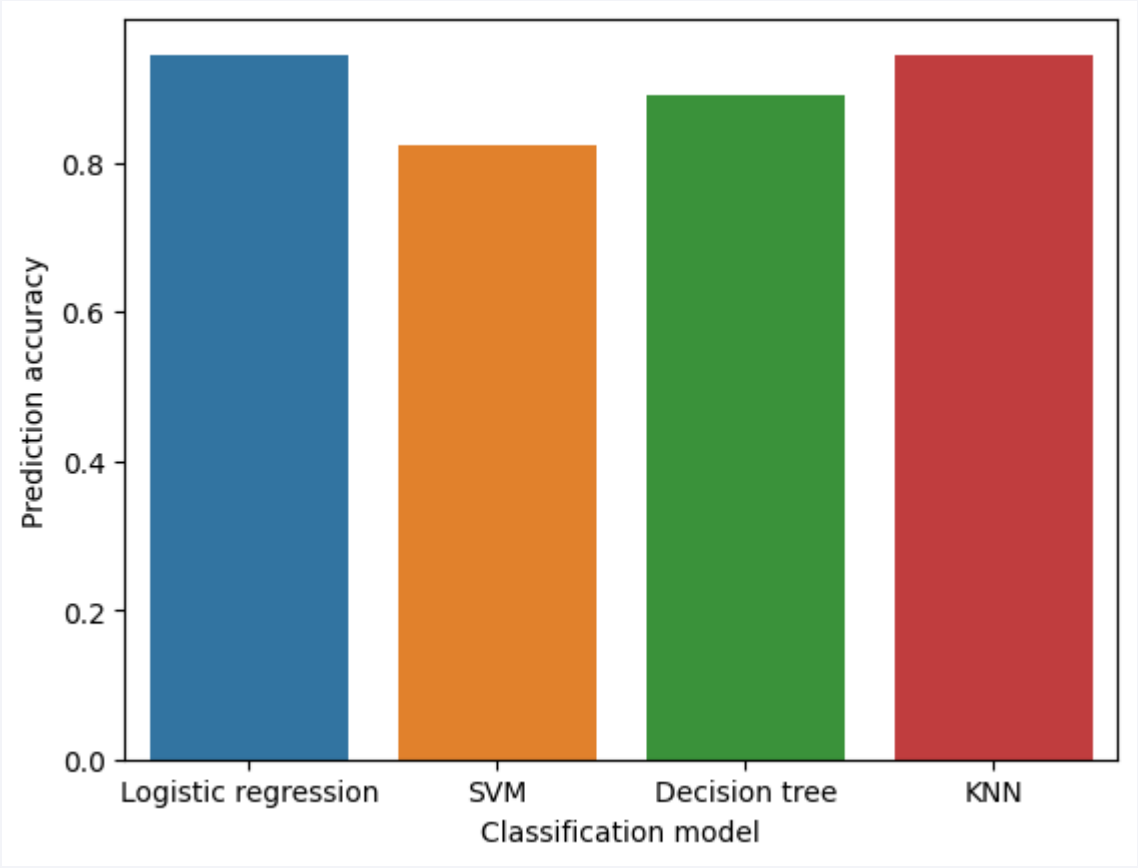
- No clear relation between payload and success

Section 5

Predictive Analysis (Classification)

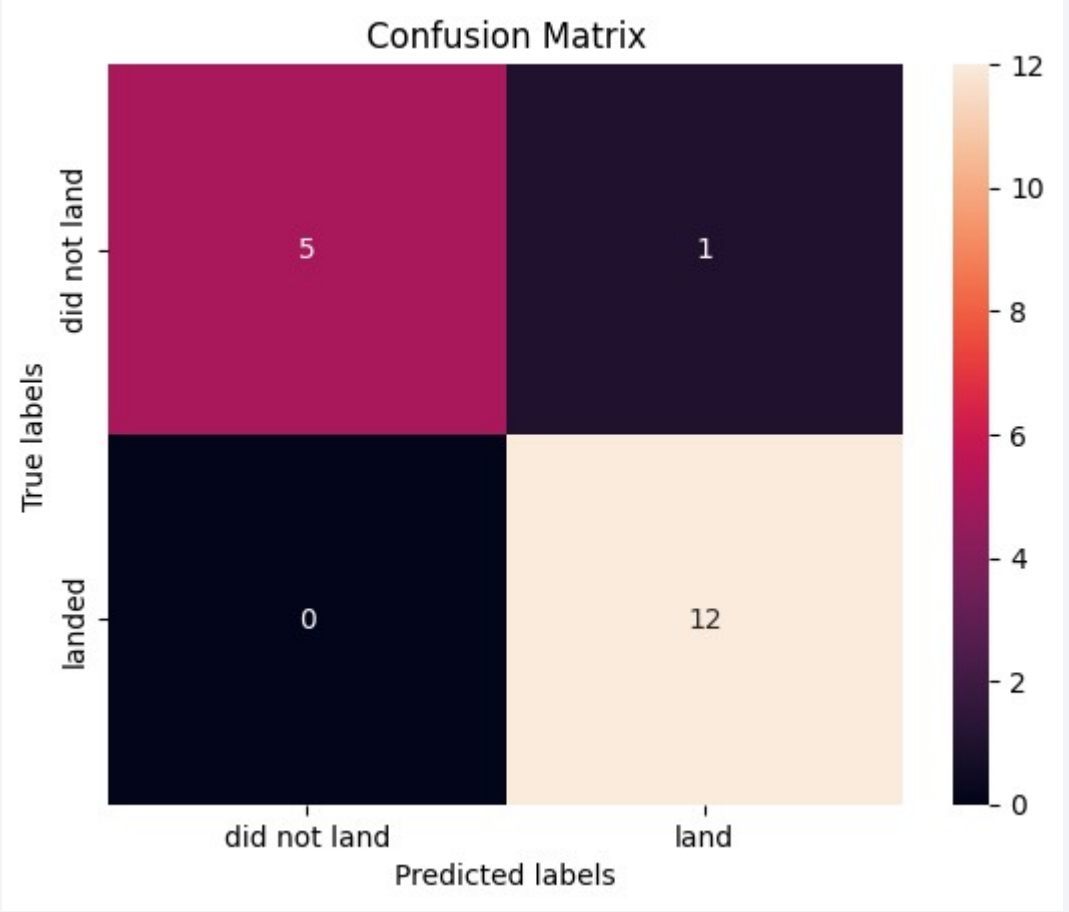
Classification Accuracy

- Logistic regression and KNN classification models have the highest accuracy on the test data
- Both achieve an accuracy of 94.4%, which means they made a single classification mistake in the test data



Confusion Matrix

- All 12 cases where the landing was successful were correctly classified by the model
- The model never predicted an unsuccessful landing when the real landing was successful
- When the landing was not successful, 5 times the model also predicted the landing would be unsuccessful
- The only misclassification is a single case where the model predicted a successful land, but the actual label was did not land



Conclusions

SpaceX launched more than 80 Falcon 9 rockets from multiple launching sites

These rockets carried various payloads into different orbits

Successful landing of rocket boosters became better over time, with a high success rate in recent years

Multiple classification models can predict landing success with high accuracy

Appendix

Link to code used to generate results:

https://github.com/JBaijens/IBM_Data_Science

Thank you!