IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Dushyant Barot
18 Dec. 2025

Outline

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

Executive Summary

Summary of methodologies

1. Data collection: Using SPACEX REST API and webscraping from Wikipedia using BeautifulSoup

2. Data cleaning and processing: Checked for null values. Implemented SQL queries on the data and conducted EDA on the data.

3. Feature Engineering: Performed one-hot encoding on categorical features.

4. Developed interactive visualizations using Folium and Dash

5. Implemented several different classification models and tuned hyperparameters using cross-validation.

Summary of Results

1. Identified factors affecting success of first stage landings.

2. Decision Tree model gave the best 94.44% accuracy with least number of false positives.

3. Launch Site and payload mass have significant impact on landing outcome.

Introduction

Project background and context

- SpaceX reduces launch costs by reusing Falcon 9 first stages
- Predicting landing success helps estimate launch cost and competitiveness

Problems we want to find answers to

- What factors affect Falcon 9 first-stage landing success?
- Can machine learning accurately predict landing outcomes?
- What classification model performs best?

Section 1

Methodology

Methodology

Executive Summary: This project is an end-to-end data science workflow to predict landing outcome of first stage of Falcon 9 including data collection, data wrangling, EDA, interactive analytics and predictive modelling.

Data collection methodology: SpaceX REST API was used to fetch launch details using Python requests. Data was supplemented by web scraping from Wikipedia using BeautifulSoup.

Perform data wrangling: Converted tables into structured pandas DataFrames. Cleaned and standardized the scraped data. Performed one-hot encoding on the categorical variables. Normalized the numerical variables.

Perform exploratory data analysis (EDA) using visualization and SQL

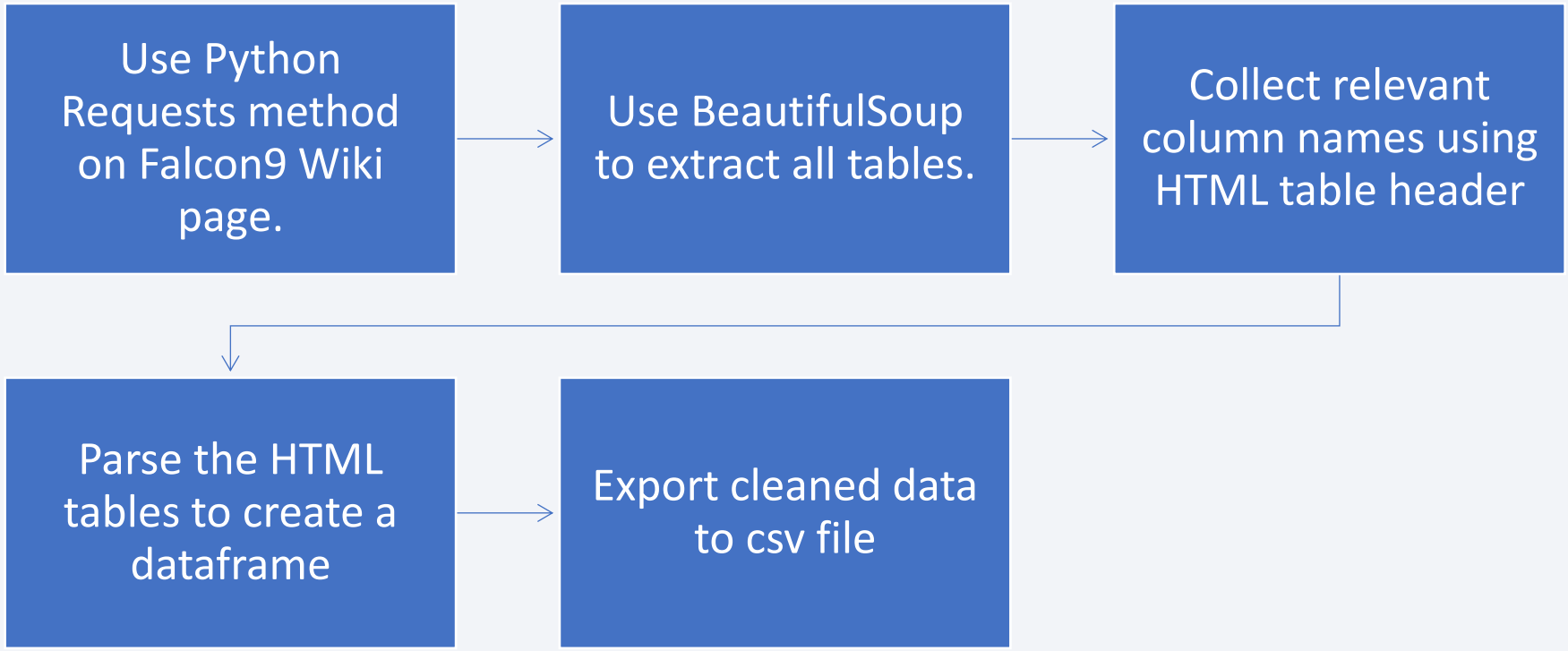
Perform interactive visual analytics using Folium and Plotly Dash

Perform predictive analysis using classification models: Built, tuned and evaluated several classification models.

Data Collection

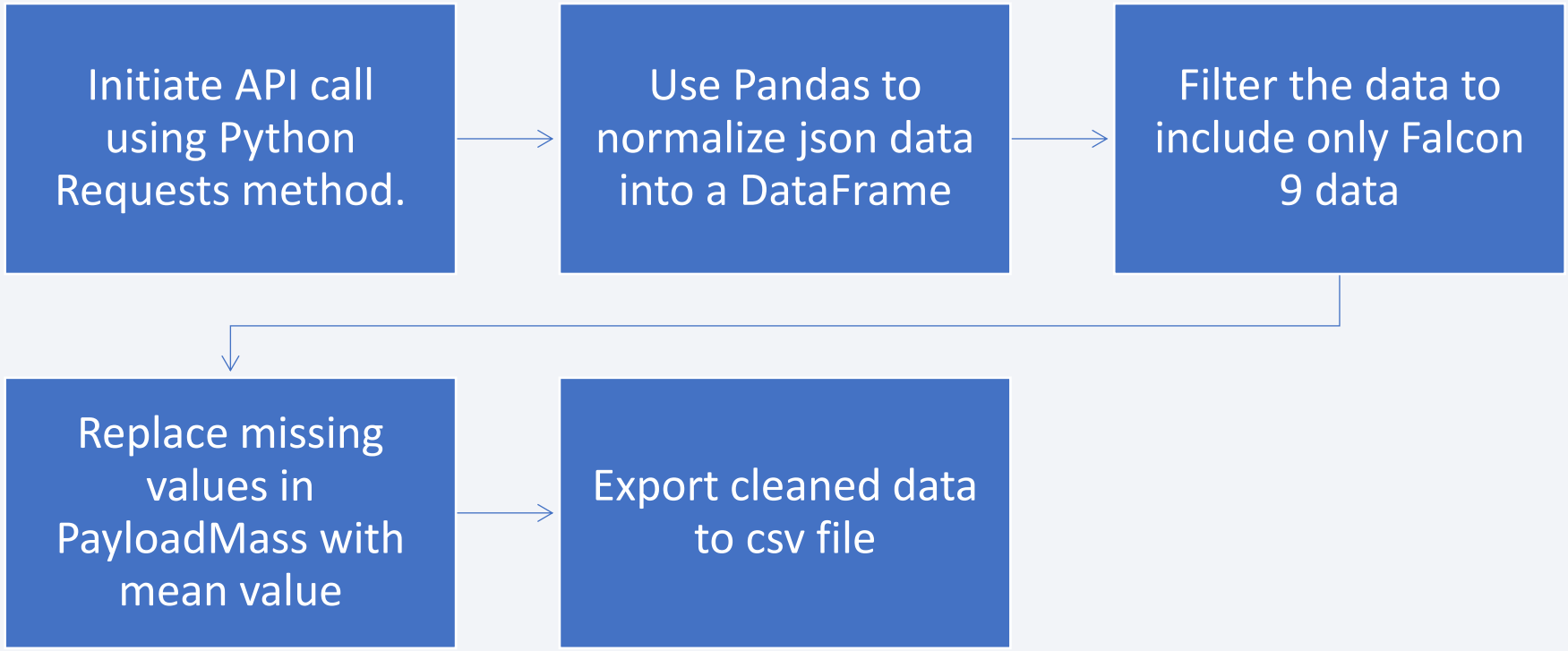
- This step was a combination of using SpaceX REST API and webscraping form SpaceX Falcon Wikipedia page.
- Data collected from both sources were combined to get a complete picture.

Data Collection – SpaceX API



GitHub URL: Data Collection using SpaceX API

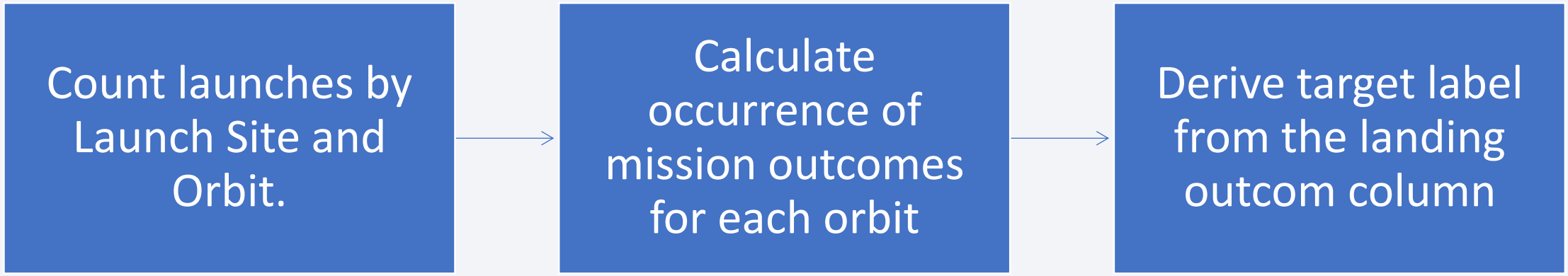
Data Collection – Scraping



GitHub URL: Data Collection using Webscraping

Data Wrangling

Some EDA was performed to find patterns and engineered a target column based on landing outcome.



GitHub URL: Data Wrangling

EDA with Data Visualization

1. **Scatter plots** of Flight number vs Class and **Line plot** Year vs Success Rate revealed how launch success has improved over time supporting narrative of SpaceX's increasing reliability.
2. **Scatter plots** of Payload Mass vs Orbit/ Launch Site with **hue** as launch outcome helped identity non-linear relationships and operational constraints affect landing success.
3. **Bar chart** to visualize success rate of each orbit type.

GitHub URL: EDA with Data Visualization

EDA with SQL

- Identify all distinct launch sites used
- Retrieve sample launch records with launch site names starting with ‘CCA’.
- Compute the total payload mass delivered under NASA (CRS) program.
- Calculate average payload mass carried by the F9 v1.1 booster version.
- Determine the earliest date on which a successful ground pad landing was achieved.
- List booster versions with successful drone ship landing with payload between 4000 and 6000 kg
- Count and summarize the number of successful and failed mission outcomes.

EDA with SQL

- Identify all booster versions that carried max. payload mass using subquery and aggregate function.
- Display 2015 mission records showing month, failed drone ship landings, booster versions and launch sites.
- Rank different landing outcomes by their frequency, in descending order, within the range 2010-06-04 to 2017-03-20

GitHub URL: EDA with SQL

Build an Interactive Map with Folium

- A marker and circle was created for each launch site on the site map.
- Green and red markers were added to each launch site based on success or failure of a launch. This was done using MarkerCluster Object as multiple launches were there for the same site.
- Line was drawn from launch site '*KSC LC-39A*' to nearest city and coast to identify proximity of launch site from city and coast.

GitHub URL: Interactive Map with Folium

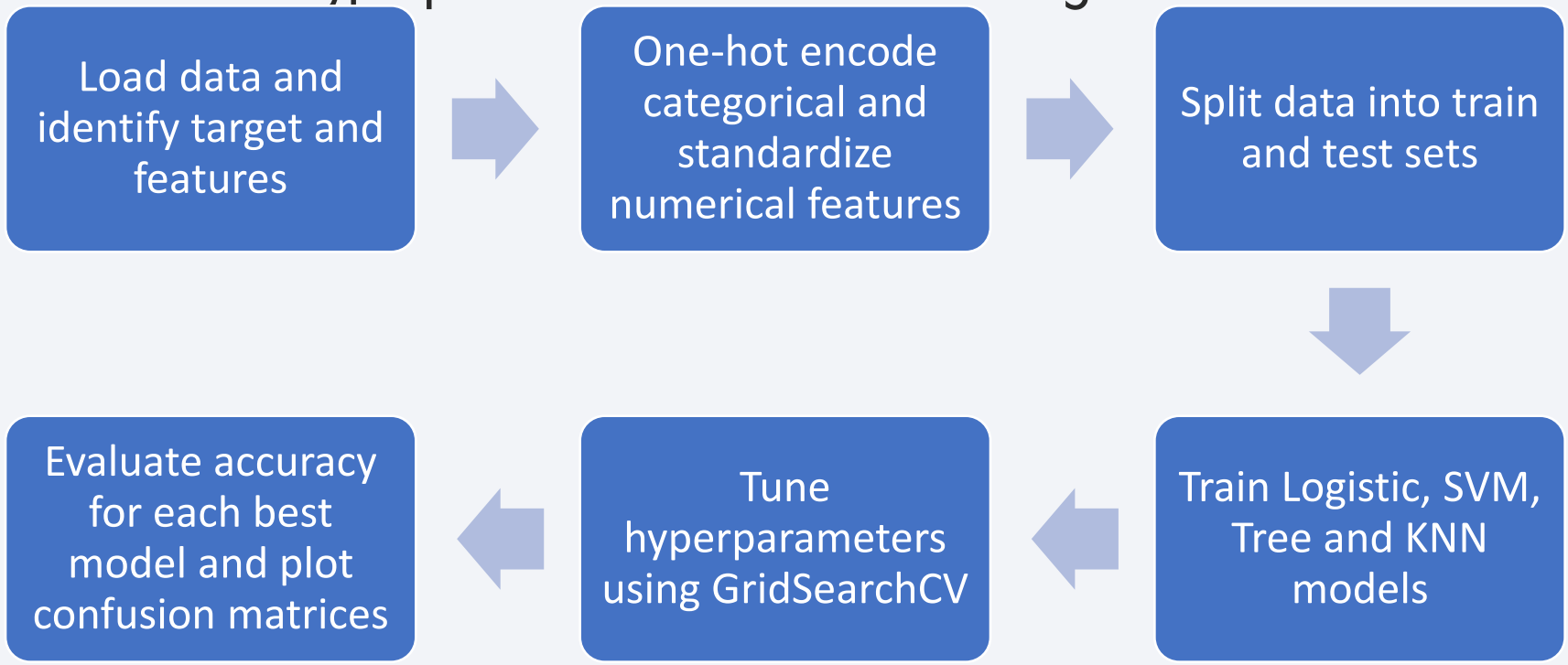
Build a Dashboard with Plotly Dash

- Drop-down was added to select launch site. Based on the selection a pie chart was displayed for percentage of successful landing for each launch site.
- Range slider was added to select range of Payload mass. Within this range of payload a scatter plot was displayed between Class on y-axis and payload on x-axis for each booster version.

GitHub URL: Dashboard with Plotly Dash

Predictive Analysis (Classification)

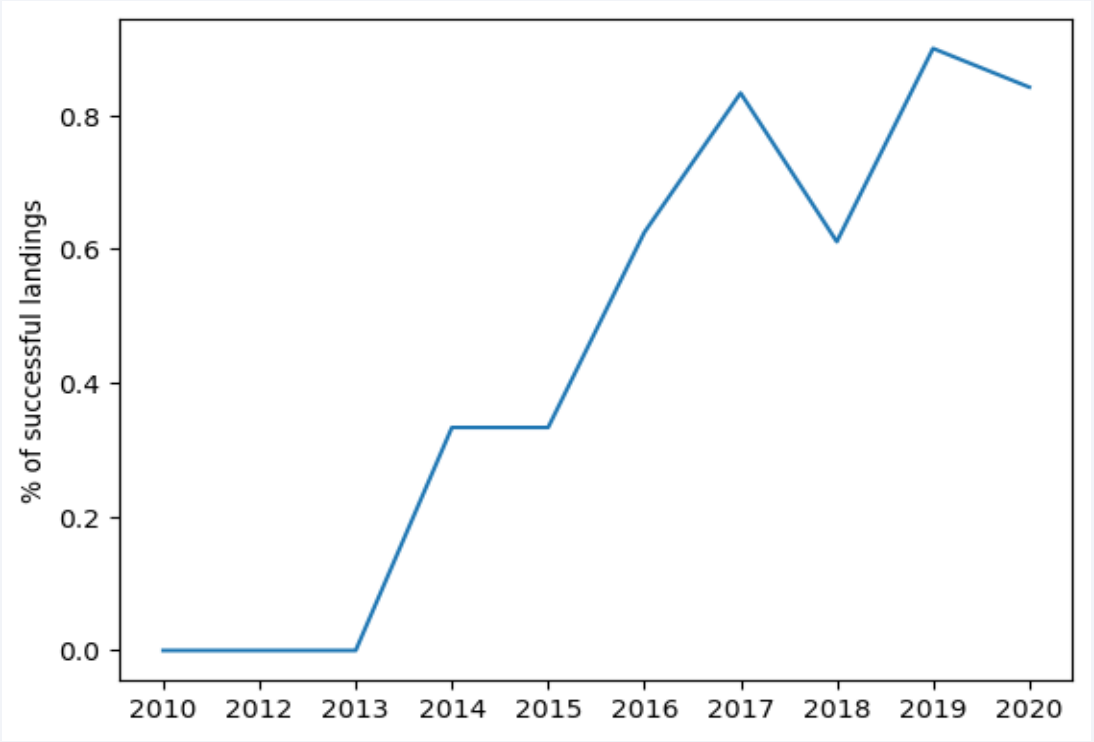
- Four classification methods: SVM, Decision Tree, Logistic Regression and KNN were tested and hyperparameters were tuned using cross-validation.



GitHub URL: Predictive Analysis

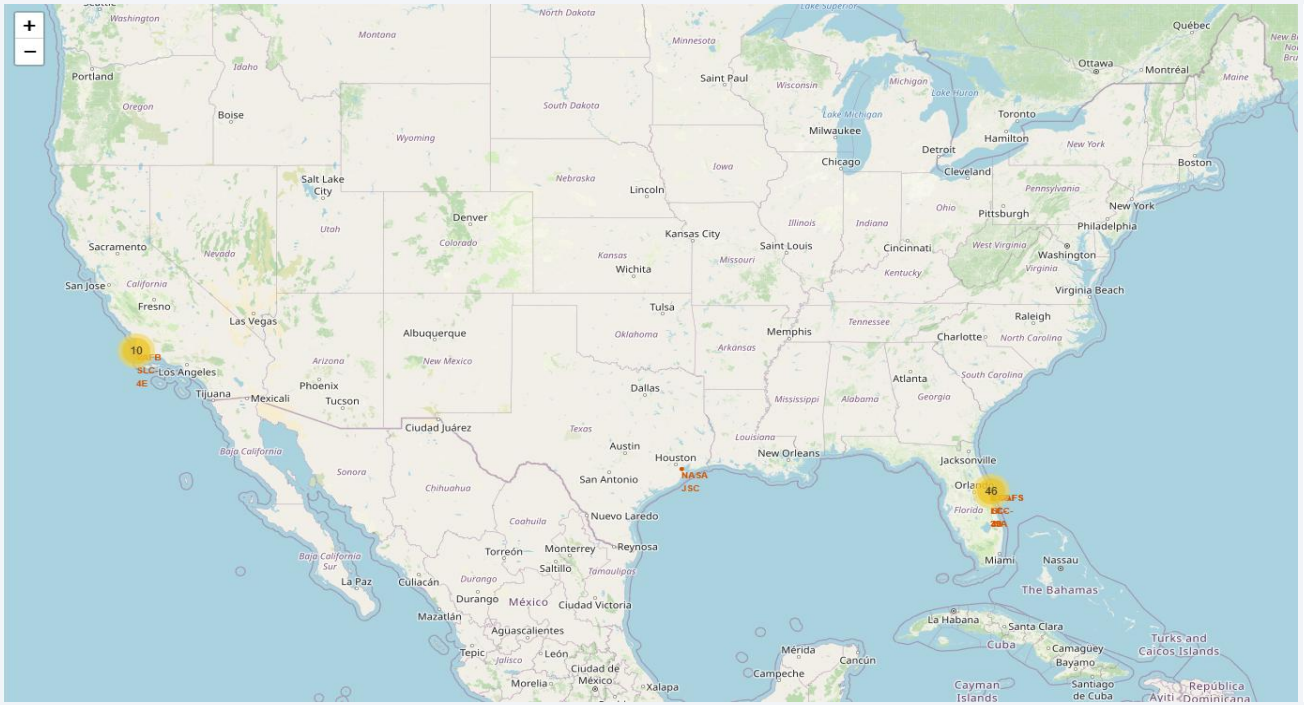
Results

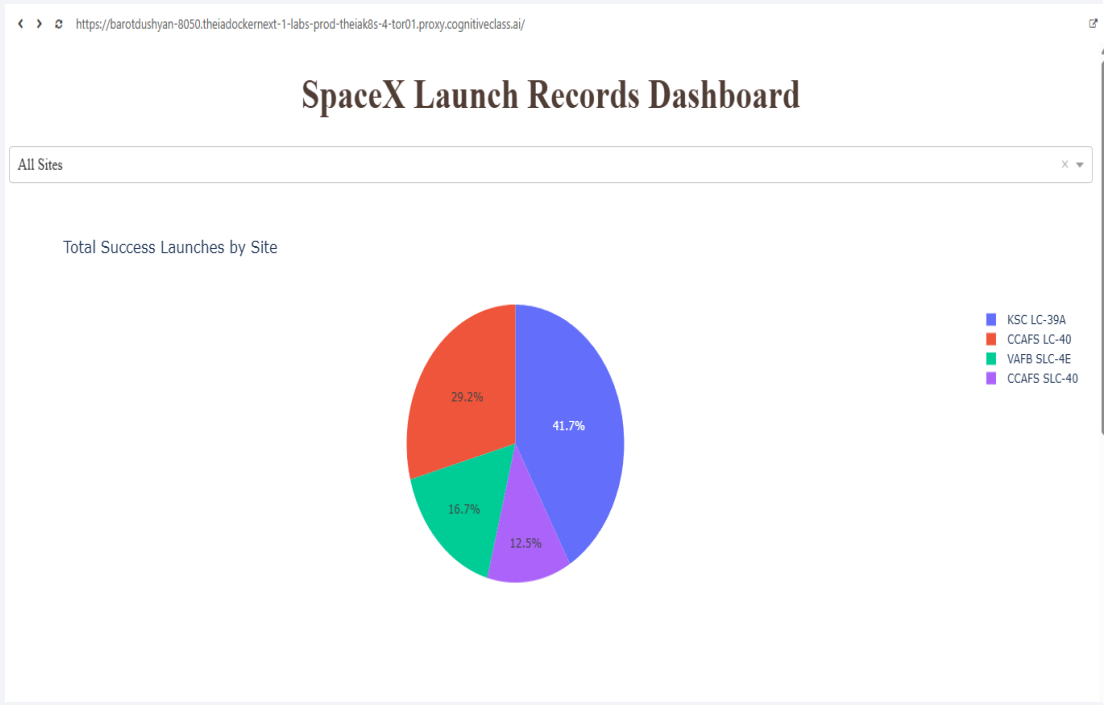
- **Exploratory data analysis results**
  - 4 launch sites were used.
  - First successful landing 5 years after launch
  - Landing outcomes got better with time



Results

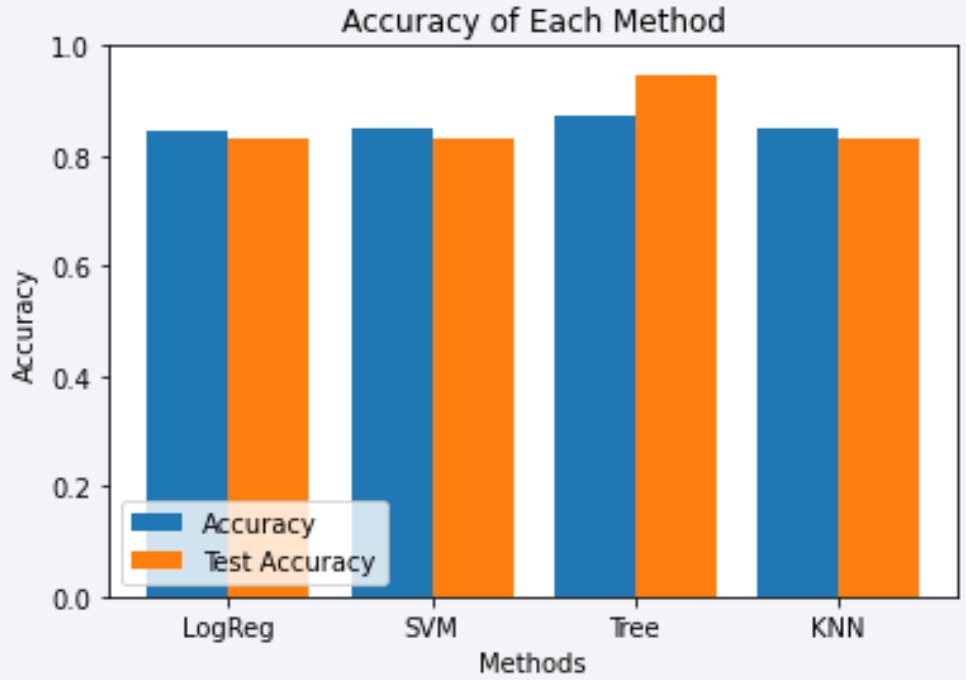
- **Interactive analytics demo in screenshots**
  - Most launch sites were near coasts and away from cities
  - Percentage of successful landing for each sites was plotted on a pie chart.

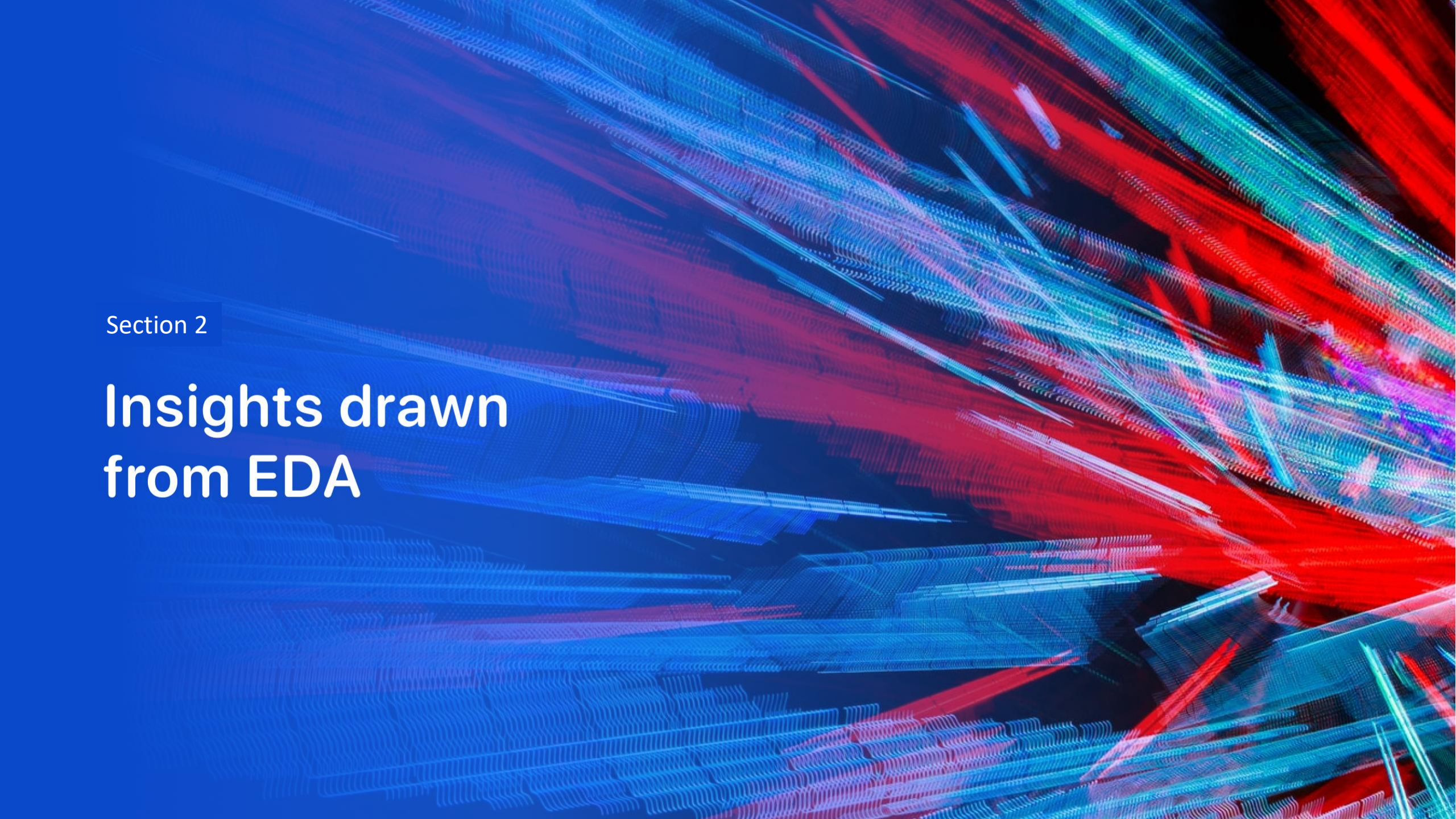




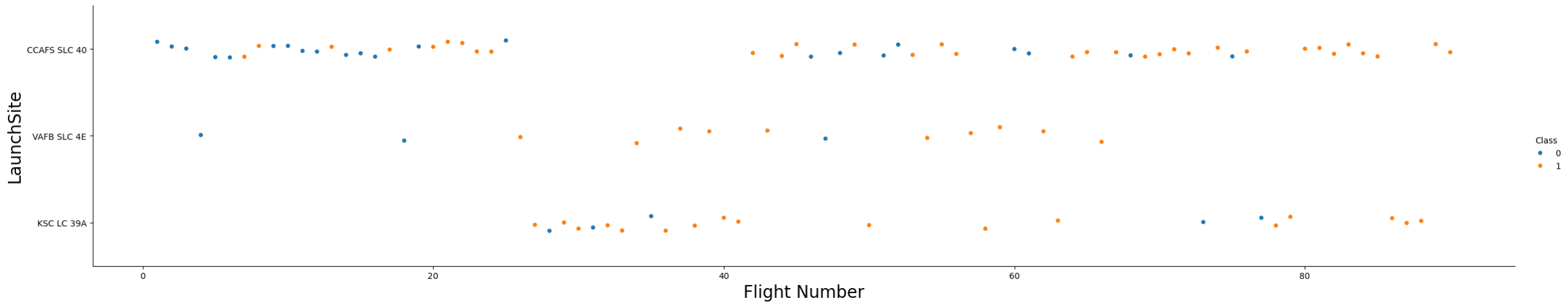
Results

- **Predictive analysis results**
  - 4 models were tuned using cross-validation
  - Decision Tree gave the best test accuracy of 94.44% with least false positives.



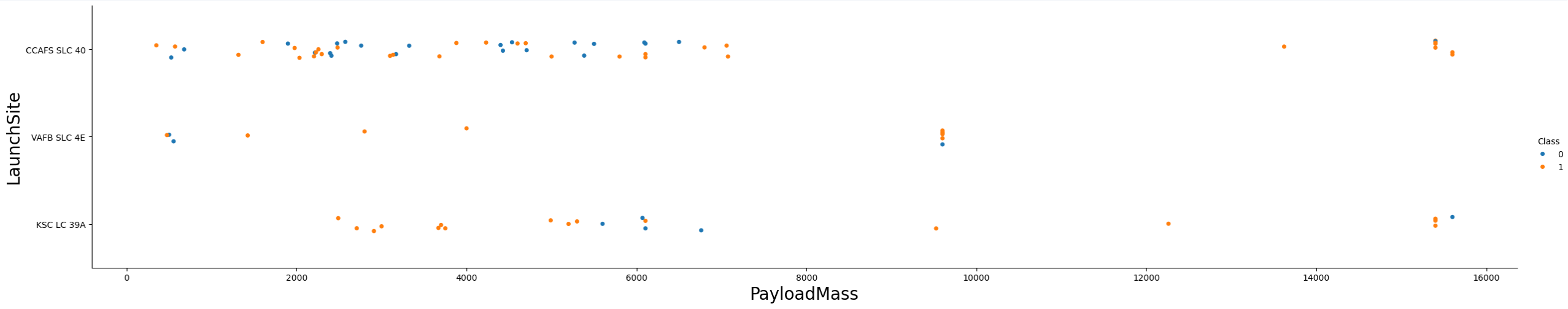

Section 2
Insights drawn from EDA

Flight Number vs. Launch Site



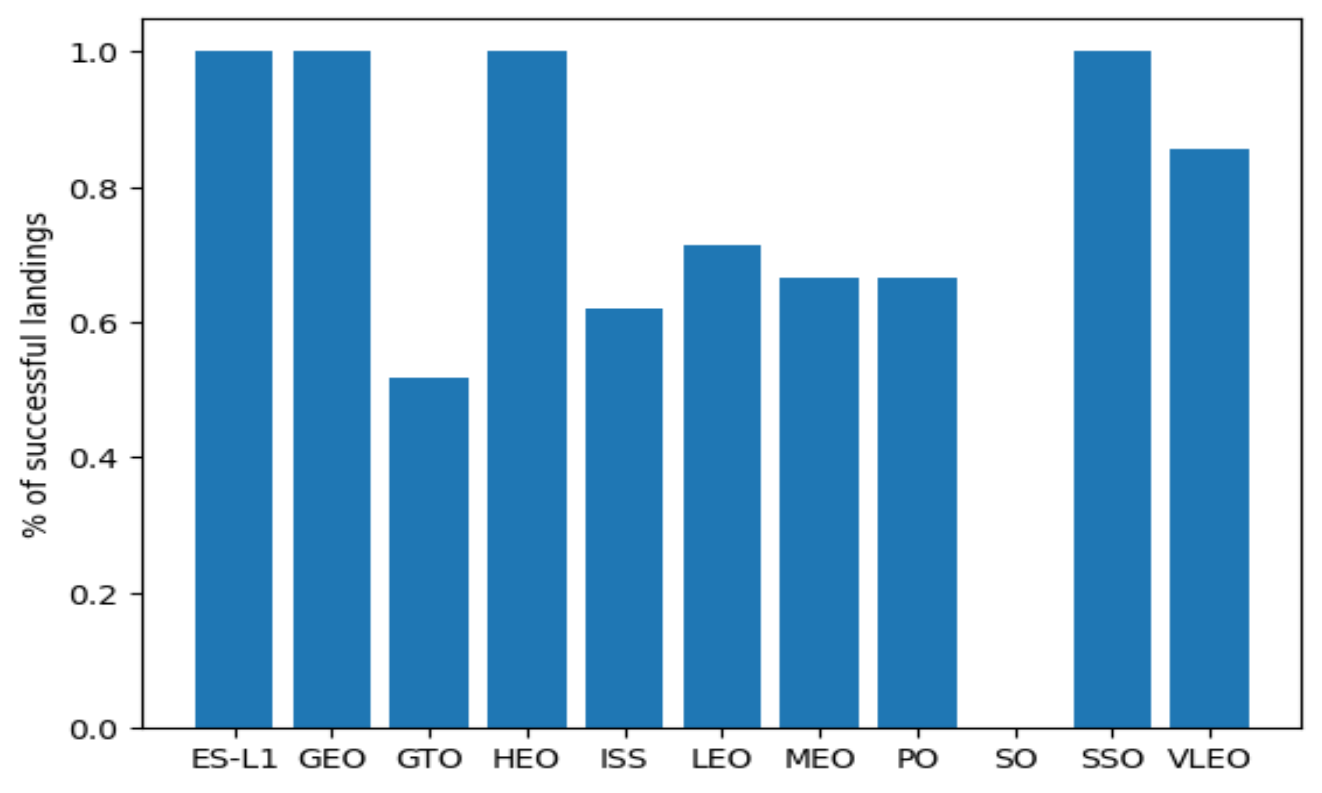
- We see success rate has improved over time for all launch site.
- CCAF5 SLC 40 is the most widely used launch site.

Payload vs. Launch Site



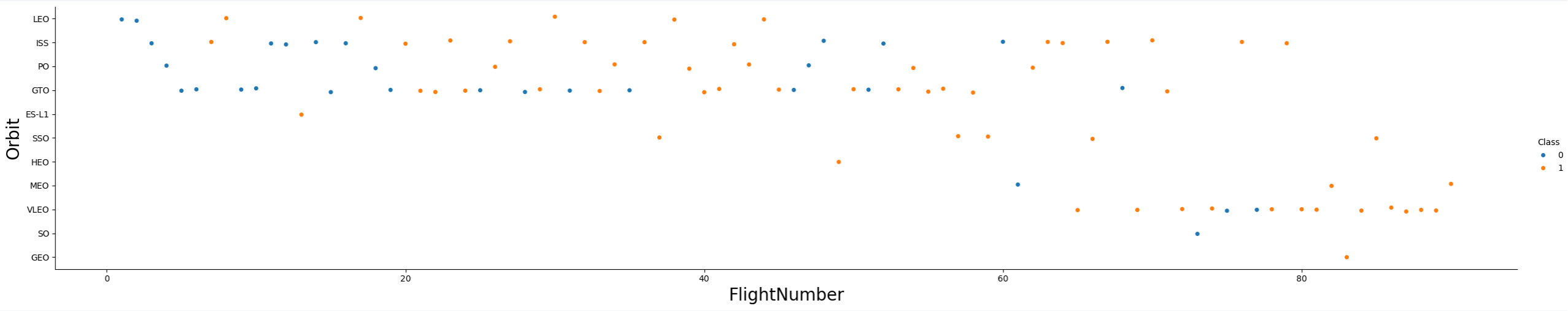
- Payloads over 12000 kg are mostly from CCAFS SLC 40 and KSC LC
- Smaller payloads have relatively higher success rate

Success Rate vs. Orbit Type



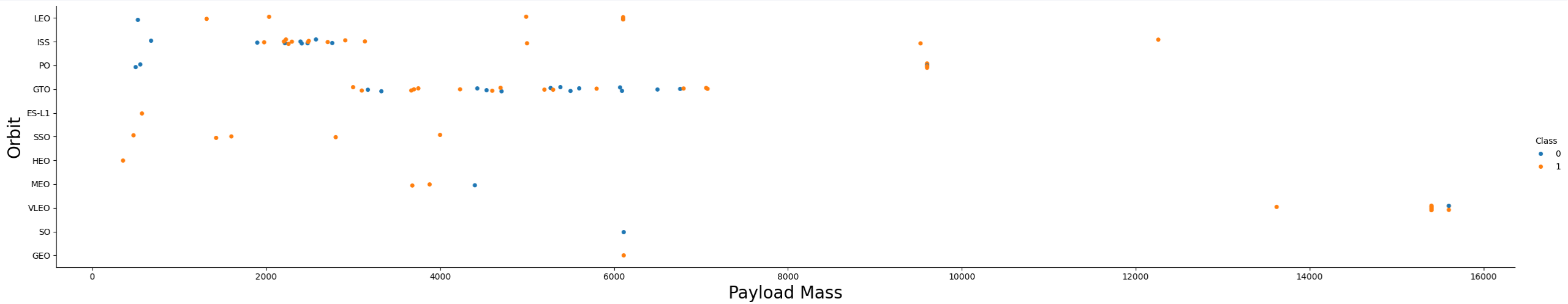
- ES-L1, GEO, HEO, SSO have high success rates.
- GTO has lowest success rate while LEO and VLEO has moderate success.

Flight Number vs. Orbit Type



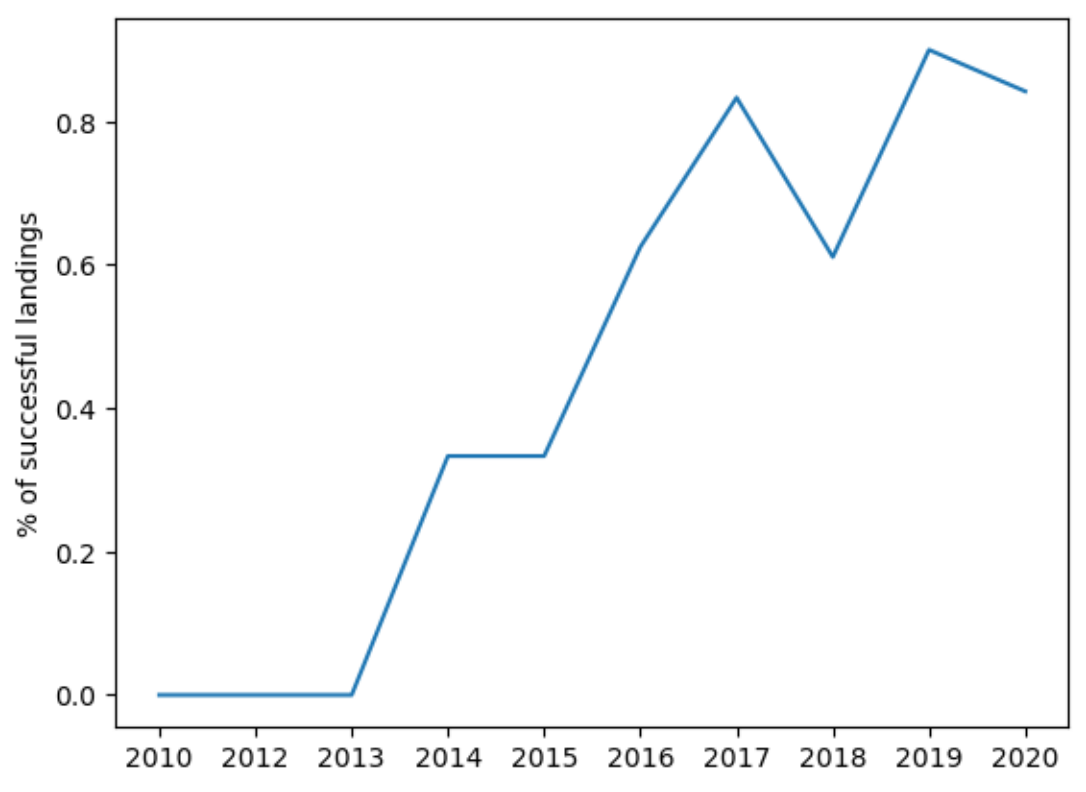
- Initial flights were lower earth orbits only and mostly failures
- Last 20 or so flights had mostly bigger orbits especially VLEO.
- Success rate improved across all orbits with time

Payload vs. Orbit Type



- For GTO orbit, the success rate has not varied with payload range.
- Large earth orbits also have much larger payloads.
- Very few launches for the SO and GEO orbits.

Launch Success Yearly Trend



- Till 2013 all launches were failures
- Success rate has improved steadily since then

All Launch Site Names

```
In [12]: %sql SELECT DISTINCT Launch_Site from SPACEXTABLE
 * sqlite:///my_data1.db
Done.
```

Out[12]:

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

We selected only the Launch_Site column and applied the DISTINCT clause to avoid repetition

Launch Site Names Begin with 'CCA'

We selected all columns and in the WHERE clause used LIKE and then ‘CCA%’ which means anything that starts with CCA and then LIMIT 5 to limit to first 5 results.

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

In [13]:
```
%sql SELECT * FROM SPACEXTABLE WHERE Launch_Site LIKE 'CCA%' LIMIT 5
```

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

Out[13]:

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)

```
In [14]: %sql SELECT SUM(PAYLOAD_MASS__KG_) as total_payload_mass FROM SPACEXTABLE WHERE Customer = 'NASA (CRS)'
 * sqlite:///my_data1.db
Done.
```

Out[14]:

total_payload_mass
45596

We summed up all the payload using SUM function while considering only when Customer was ‘NASA (CRS)’. This was done by adding a WHERE clause.

Average Payload Mass by F9 v1.1

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

```
In [15]: %sql SELECT AVG(PAYLOAD_MASS__KG_) as average_payload_mass FROM SPACEXTABLE WHERE Booster_Version LIKE 'F9 v1.1%'
 * sqlite:///my_data1.db
Done.
```

Out[15]:

average_payload_mass
2534.6666666666665

We first took average of payload mass using AVG function. Then in WHERE clause we used LIKE ‘F9 v1.1%’ which means Booster_Version name starting with F9 v1.1

First Successful Ground Landing Date

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

Hint:Use min function

In [16]:
```
%sql SELECT MIN(DATE) first_success_landing FROM SPACEXTABLE WHERE Landing_Outcome = 'Success (ground pad)'
```

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

Out[16]:

first_success_landing
2015-12-22

We used MIN function on first_success_landing column and in WHERE clause specified the landing outcome.

Successful Drone Ship Landing with Payload between 4000 and 6000

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

[17]:
```
%sql SELECT DISTINCT Booster_Version FROM SPACEXTABLE WHERE Landing_Outcome = 'Success (drone ship)' AND PAYLOAD_MASS__KG_ > 4000 AND PAYLOAD_MASS__KG_ < 6000
```

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

[17]:

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

In WHERE clause we specified landing outcome and specified range of payload mass. Then finally used DISTINCT clause to avoid repetition.

Total Number of Successful and Failure Mission Outcomes

List the total number of successful and failure mission outcomes

```
[23]: %sql SELECT TRIM(mission_outcome), COUNT(*) as num_outcomes FROM SPACEXTABLE GROUP BY TRIM(Mission_Outcome)
```

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

[23]:

TRIM(mission_outcome)	num_outcomes
Failure (in flight)	1
Success	99
Success (payload status unclear)	1

We grouped by TRIM(Mission_Outcome) because some Success outcomes had trailing space and some didn't but SQL would see them as different.

Boosters Carried Maximum Payload

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

```
[19]: %sql SELECT DISTINCT Booster_Version FROM SPACEXTABLE WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTABLE)
 * sqlite:///my_data1.db
Done.
```

[19]:

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

First there is a subquery to calculate maximum payload using aggregate function MAX, then in WHERE clause we set payload mass to be equal to this max payload and then DISTINCT clause to avoid repetition.

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.

[20]:
```
%sql SELECT substr(Date, 6, 2) AS month,landing_outcome,booster_version,launch_site FROM SPACEXTABLE WHERE substr(Date, 0, 5) = '2015' AND landing_outcome ='Failure (drone ship)'
```

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

[20]:

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

Date and year was extracted as suggested above. Then in WHERE clause set year extracted to 2015 and landing outcome to ‘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.

```
[21]: %sql SELECT Landing_Outcome, COUNT(*) as num_outcomes 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.
```

[21]:

Landing_Outcome	num_outcomes
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

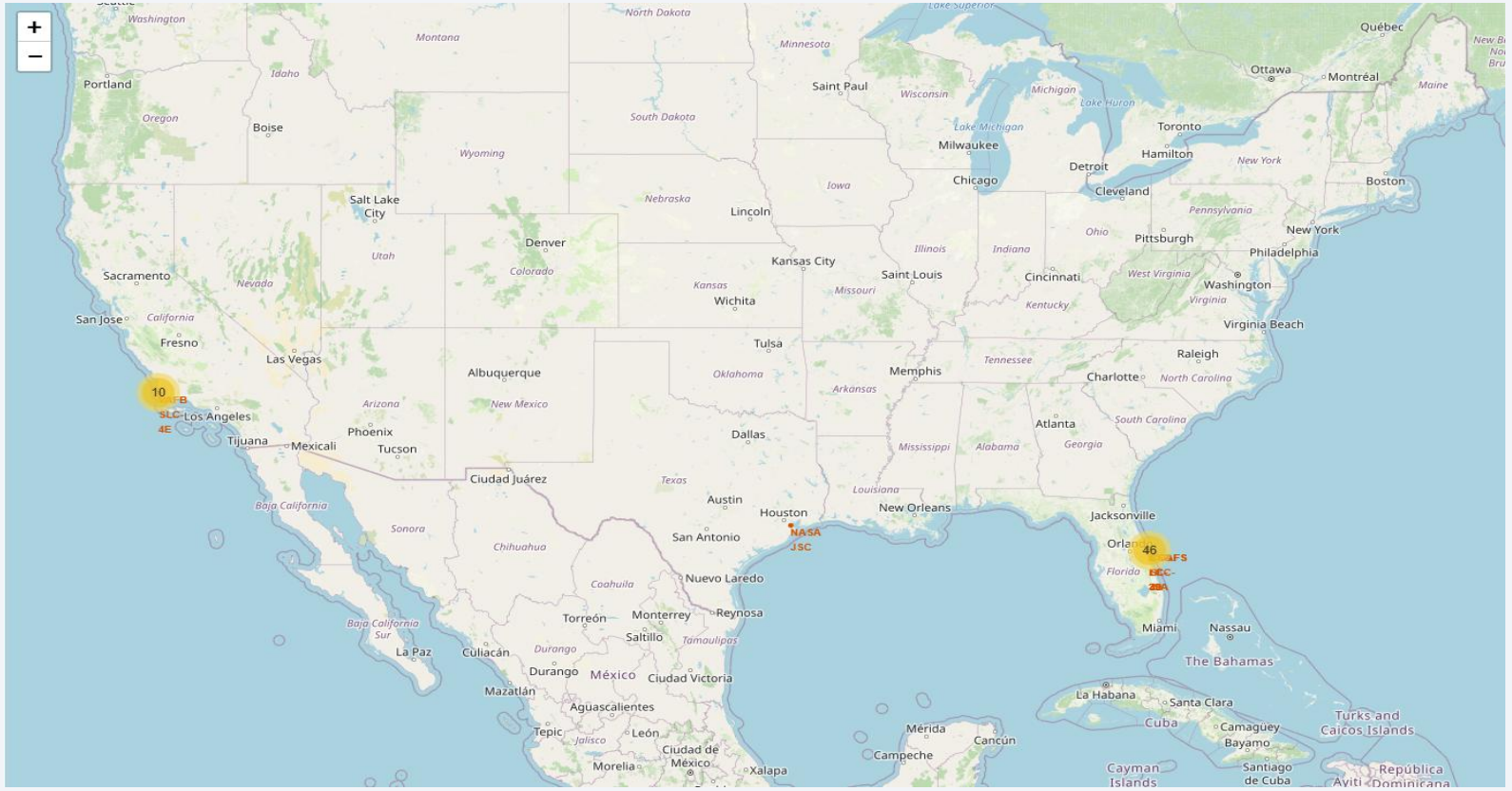
We set the date range using BETWEEN clause and then GROUP BY to group by the landing outcome and then ORDER BY to show in descending order of number of outcome. We see highest number of outcomes had 'No attempt'.

Section 3

Launch Sites Proximities Analysis

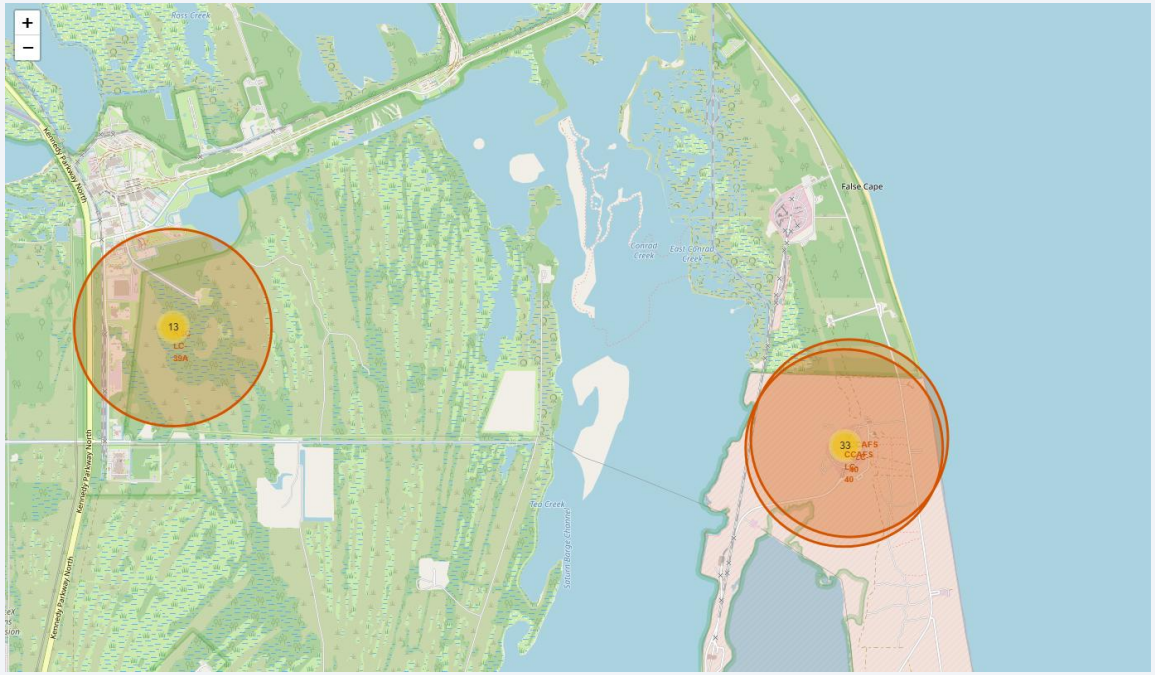
Folium: All launch sites

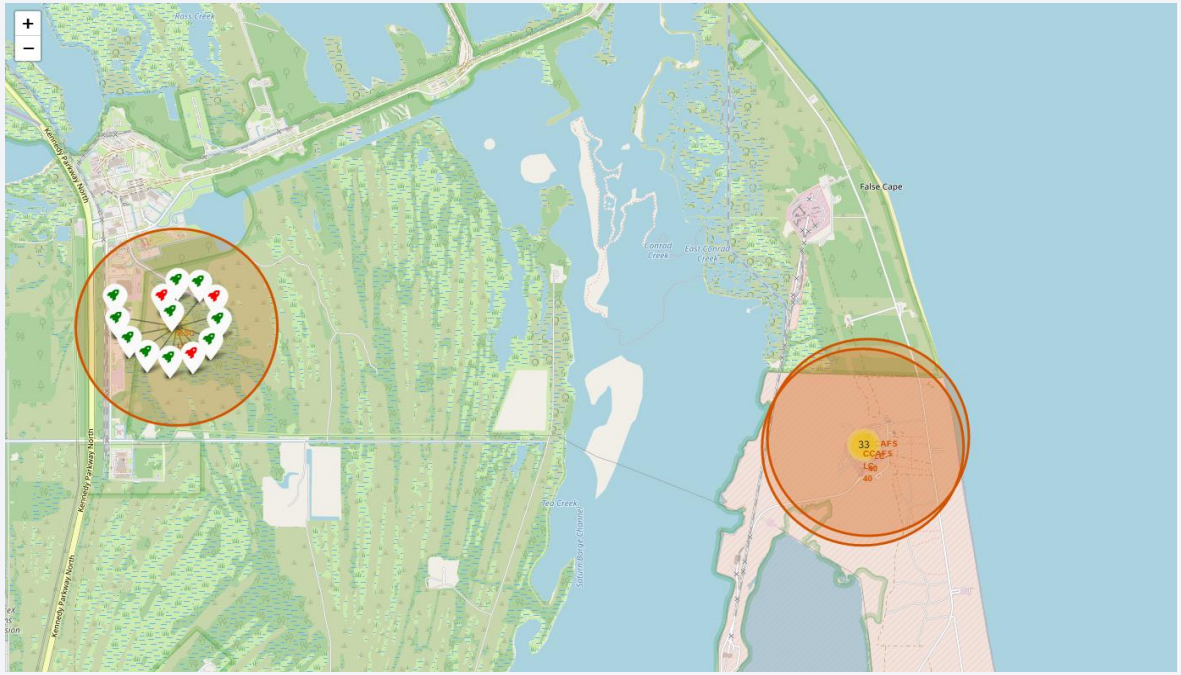
All launch sites are near the coast: east coast, west coast and one NASA JSC in the middle in Houston.



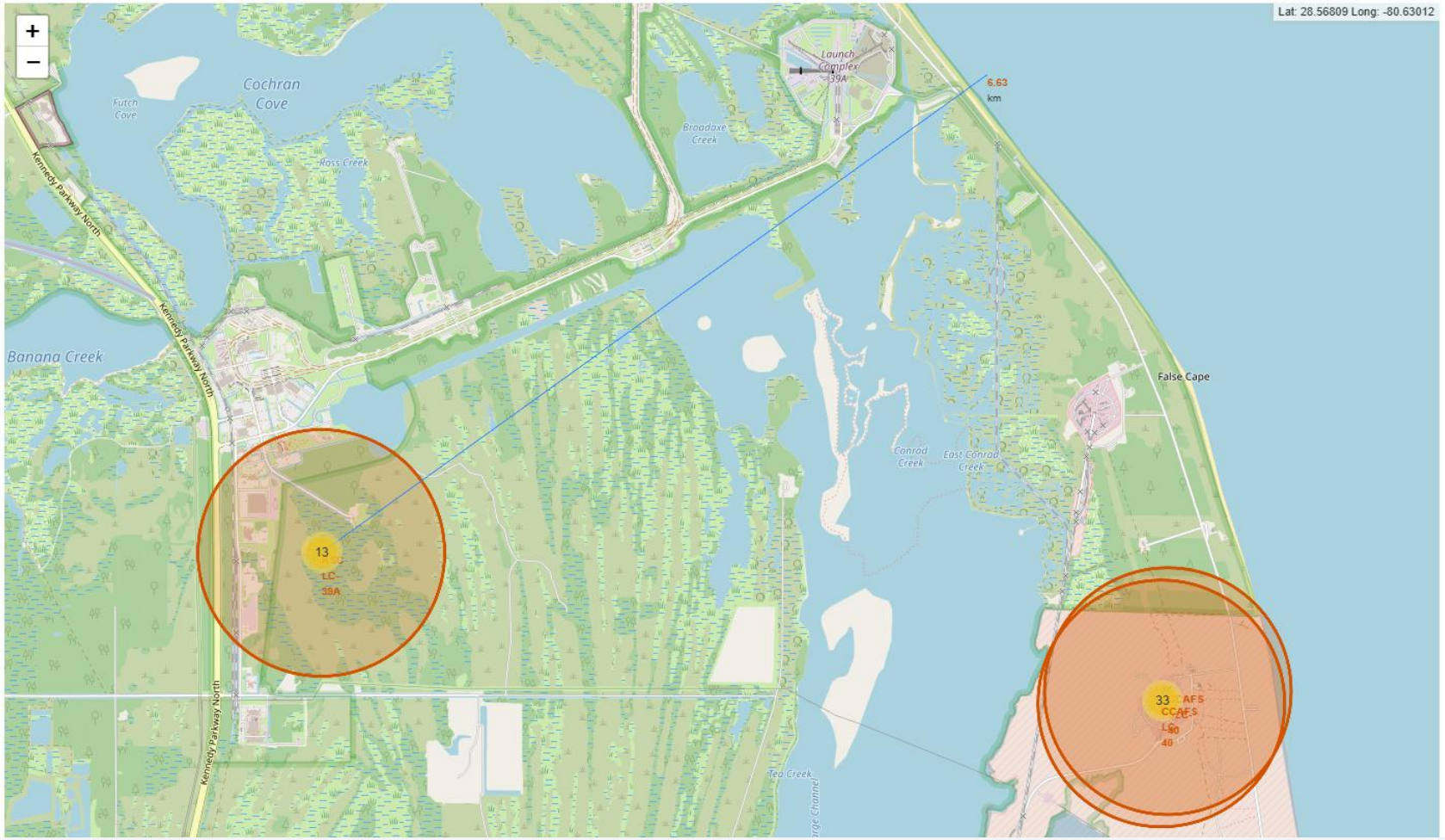
Folium: Launch outcomes

Map shows the successful and failed landing outcomes for KSC LC-39A. A total of 13 outcomes. It is displayed as a spiral but they all have the same coordinates.



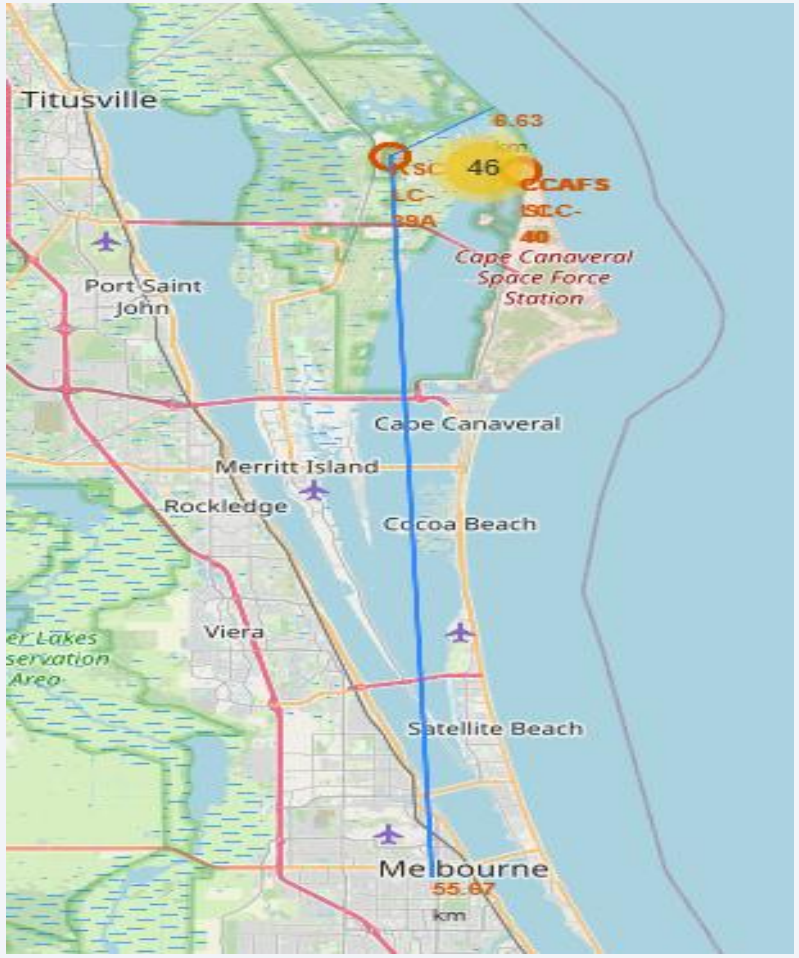


Folium: Distance from coast

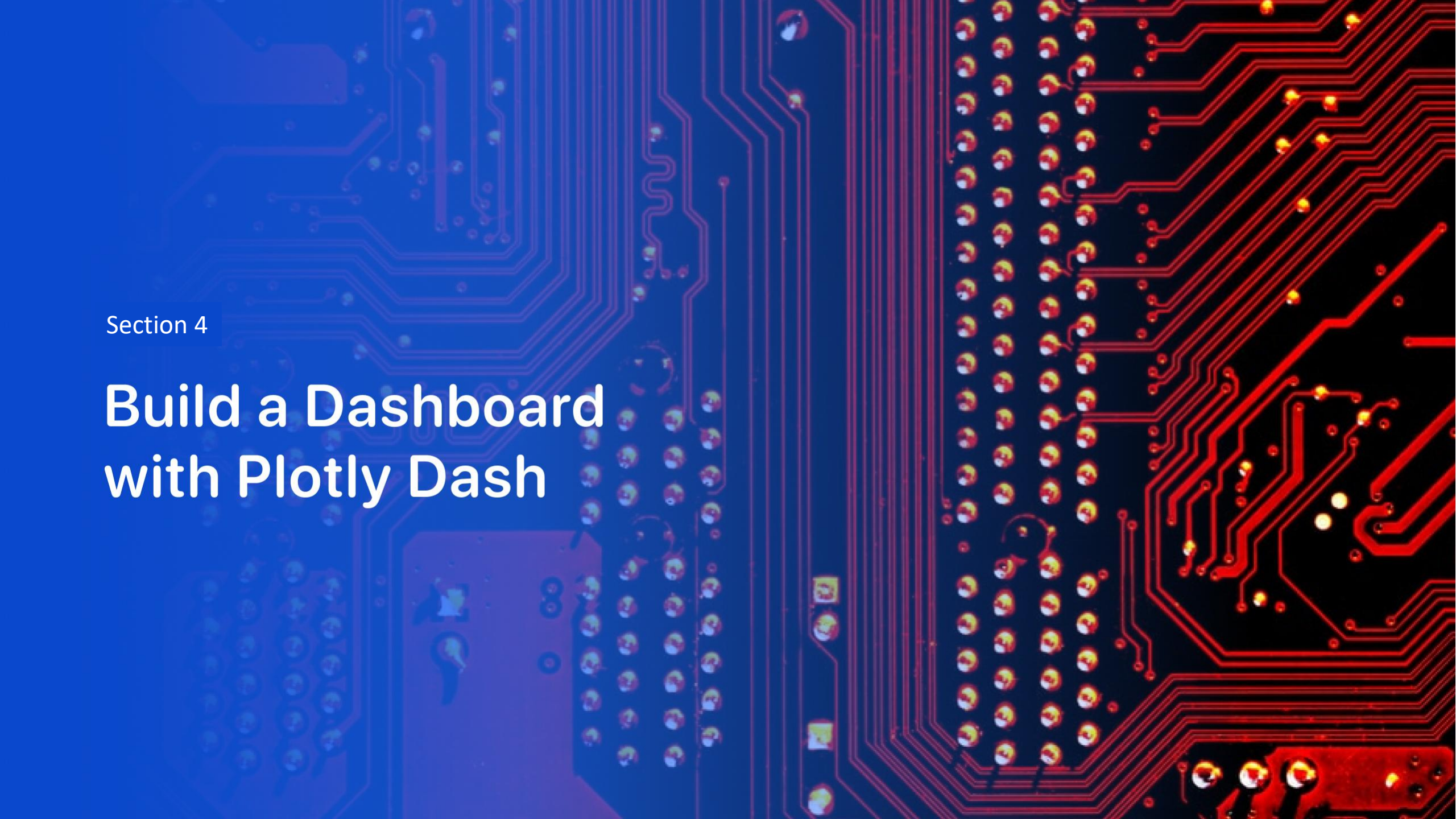


Launch site is only 6.63 km from coast which is expected to ensure transport access is easy.

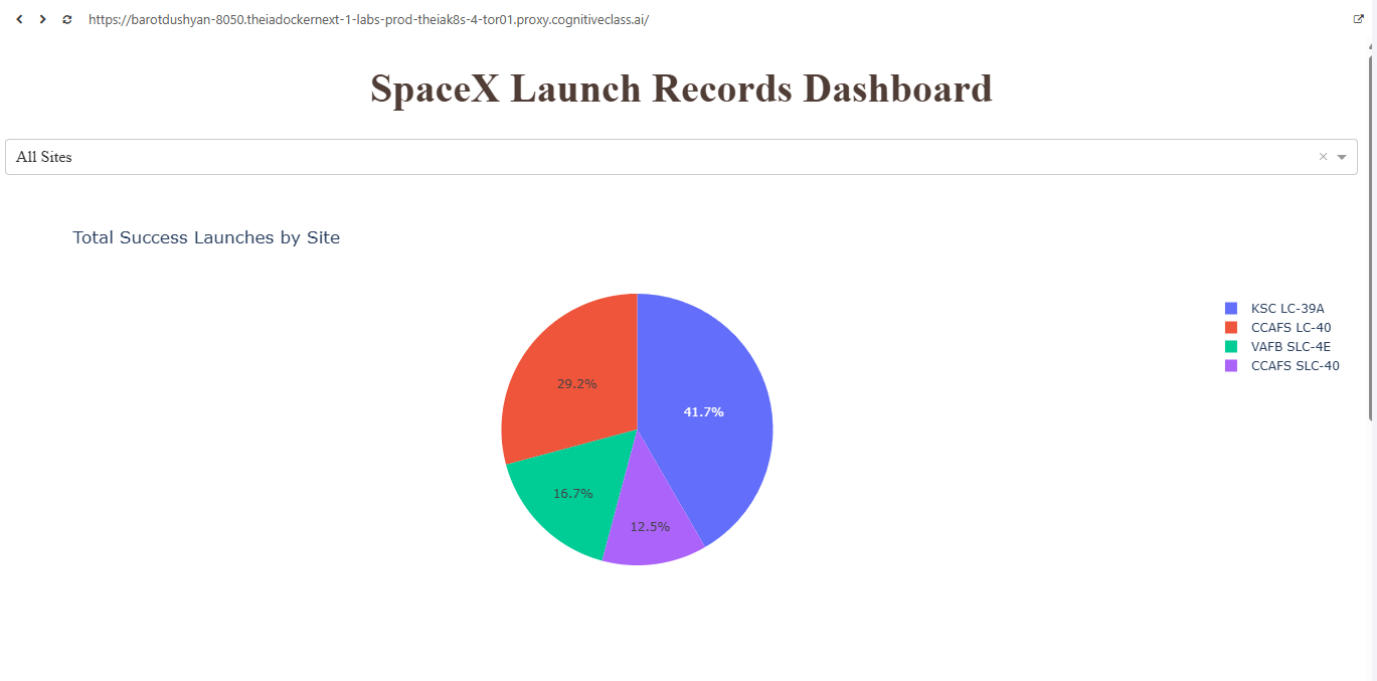
Folium: Distance from city



Launch site is 55 km from nearest city which is again expected because you want to keep launch sites away from densely inhabited areas in case of explosion during flight.

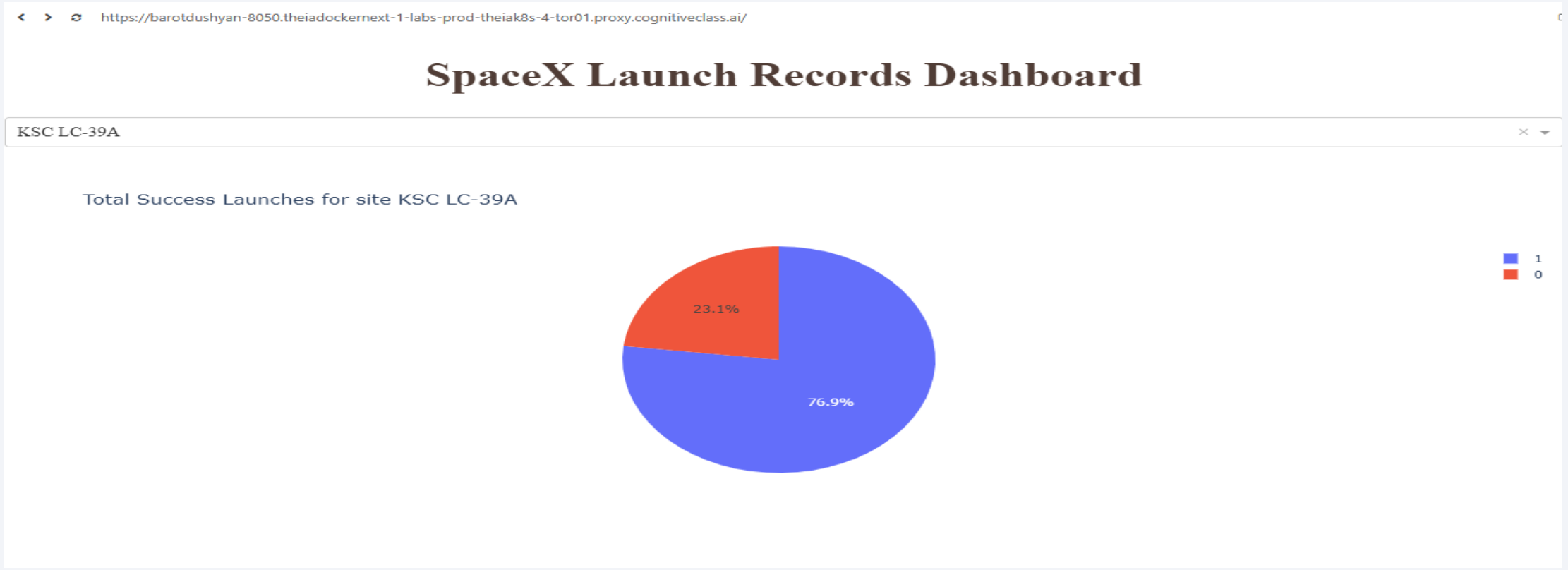

Section 4
Build a Dashboard with Plotly Dash

Dash: Success count for all sites



KSC LC-39A has highest percentage of successful launches with least for CCAFS SLC-40.

Dash: Launch success for KSC LC-39A



This site has 76.9% successful landings.

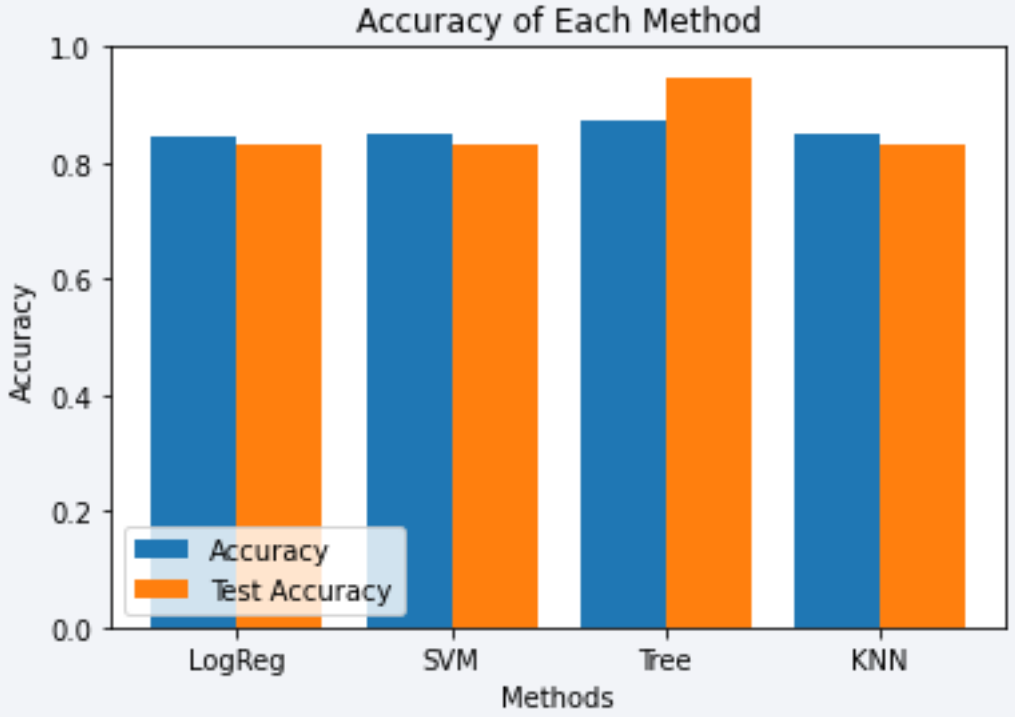
Dash: Payload vs Launch Outcome

- We have selected range of 0 to 8000 kg.
- v1.1 has a large range of payloads and largely failures
- FT has a high success rate and medium to high payload mass.

Section 5

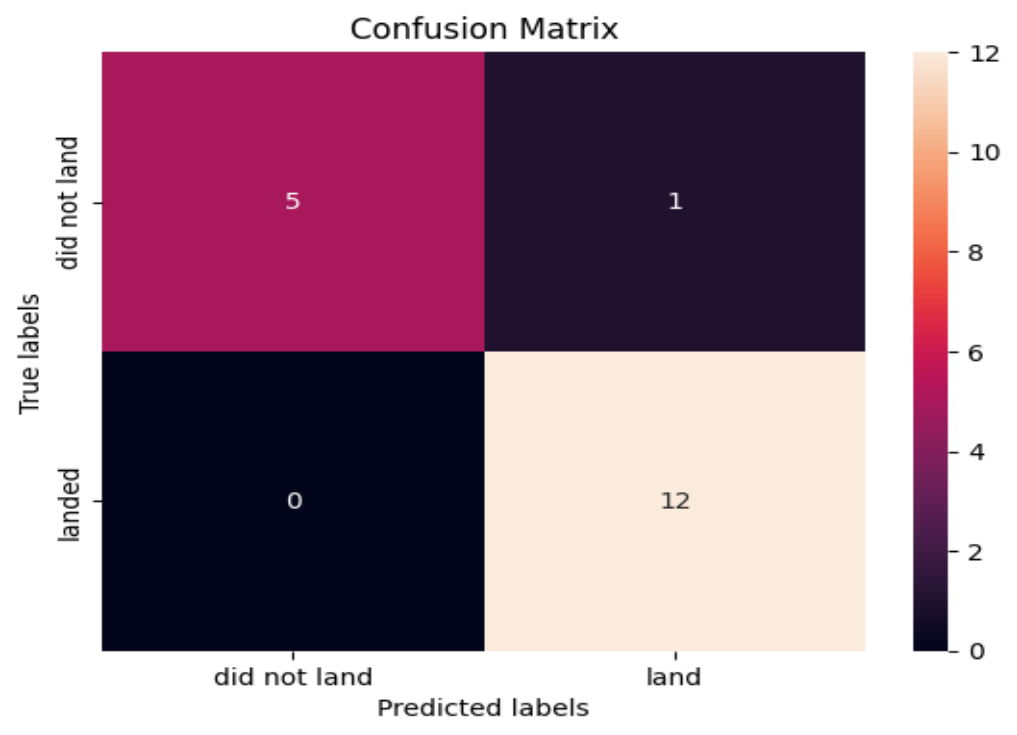
Predictive Analysis (Classification)

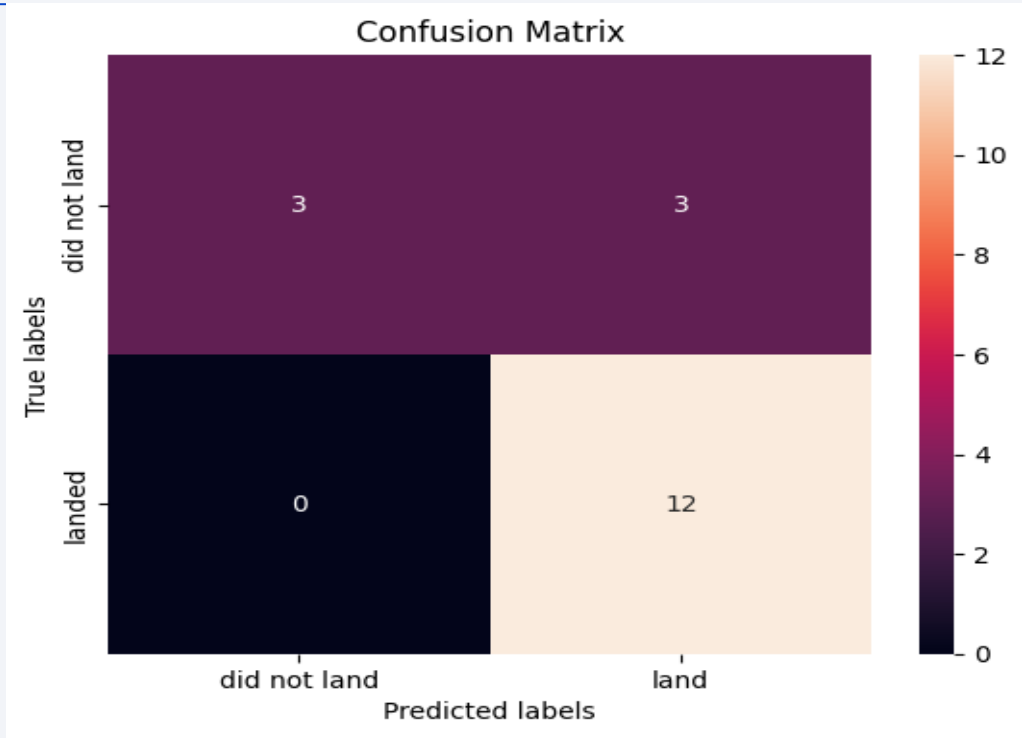
Classification Accuracy



Decision Tree model has highest test accuracy of 94.44%

Confusion Matrix





On the left is confusion matrix for tree model. It has only one false positive which is least of all models. As an example on the right is shown confusion matrix of KNN model which has 3 false positives.

Conclusions

- Percentage of successful landing increased with time as SpaceX finetuned their launches.
- Higher payloads had higher success rates but then higher payloads were seen in later years. In earlier years we saw only small payloads.
- Decision Tree model gave best test accuracy and least false positives.
- In terms of percentage of successful landings KSC LC-39A was the best launch site.

Appendix

- Need to run the jupyter notebook locally to view folium maps, they are not displayed on the notebook shared on github page.

Thank you!