Home Service About Us Contact

WINNING SPACE RACE WITH DATA SCIENCE

RICHARD DEAN TANJAYA

29 AUGUST 2024

OUTLINE

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

EXECUTIVE SUMMARY

Summary of Metodologies

- Data collection
- Data wrangling
- Exploratory Data Analysis with Data Visualization
- Exploratory Data Analysis with SQL
- Building an interactive map with Folium
- Building a Dashboard with Plotly Dash
- Predictive analysis (Classification)

Summary of Results

- Exploratory Data Analysis results
- Interactive analytics demo in screenshots
- Predictive analysis results

INTRODUCTION

SpaceX, a leader in the commercial space industry, has revolutionized space travel by making it more affordable. A Falcon 9 rocket launch costs $62 million, significantly less than other providers, who charge upwards of $165 million. The key to these savings is SpaceX's ability to reuse the rocket's first stage. By predicting whether the first stage will successfully land, we can estimate the cost of a launch. Using public data and machine learning models, this project aims to predict SpaceX's likelihood of reusing the first stage.

1. How do factors like payload mass, launch site, number of flights, and orbits influence the success of first stage landings?
2. Has the success rate of landings improved over the years?
3. Which algorithm is most effective for binary classification in this scenario?

METHODOLOGY

METHODOLOGY

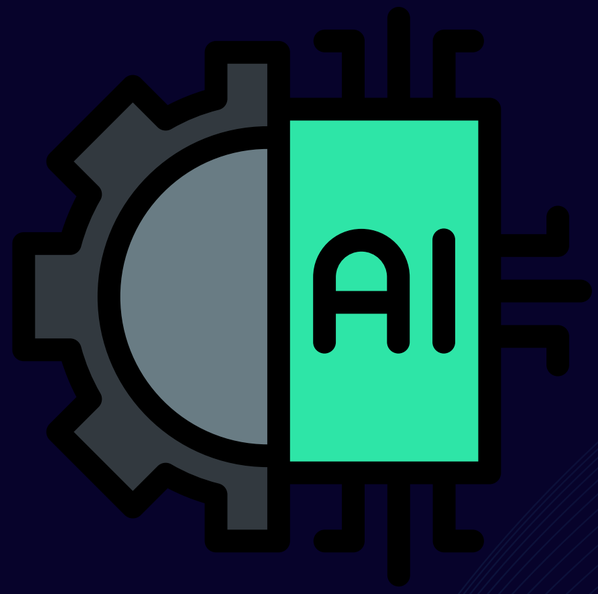
01 Data collection methodology using SpaceX Rest API and Web Scrapping from Wikipedia

02 Data wrangling: filtering the data, dealing with missing values, and One Hot Encoding to prepare the data to a binary classification

03 Exploratory data analysis (EDA) using visualization and SQL

04 Interactive visual analytics using Folium and Plotly Dash

05 Predictive analysis using classification models and ensure the best result



DATA COLLECTION

The data collection process involved retrieving information through a combination of API requests from the SpaceX REST API and web scraping from a table on SpaceX's Wikipedia page. Both methods were necessary to gather comprehensive details about the launches for a thorough analysis.

Data collected from SpaceX REST API:
FlightNumber, Date, BoosterVersion, PayloadMass, Orbit, LaunchSite, Outcome, Flights, GridFins, Reused, Legs, LandingPad, Block, ReusedCount, Serial, Longitude, and Latitude.

Data obtained through web scraping Wikipedia:
Flight No., Launch site, Payload, PayloadMass, Orbit, Customer, Launch outcome, Version Booster, Booster landing, Date, and Time.

DATA COLLECTION – SPACEX API

Request rocket launch data from SpaceX API

→ Decode the response content as a Json using .json() and turn it into a Pandas dataframe using .json_normalize()

→ Request needed information by applying custom functions because most of the data are ID’s

→ Construct our dataset and combine the columns into a dictionary

↓ Turn the dictionary into a dataframe

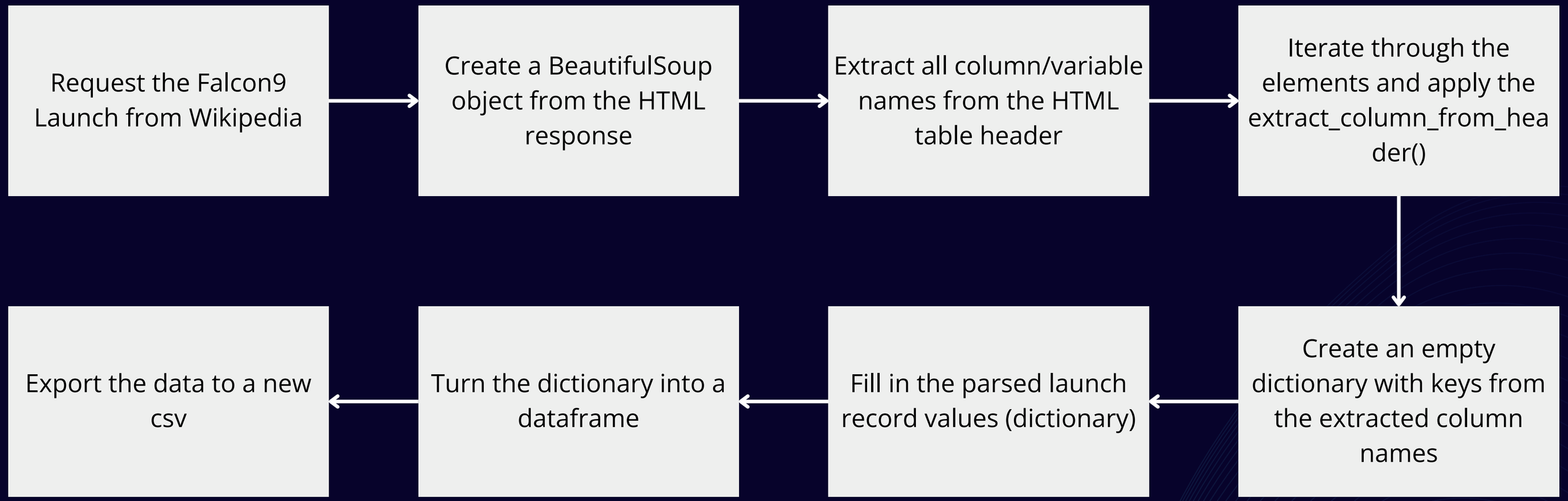
← Filter the data to only include Falcon9

← Fill missing value

← Export the cleaned data to a new csv

Github Link: Data collection SpaceX API

DATA COLLECTION - WEB SCRAP WIKIPEDIA



Github Link: Data collection Wikipedia API

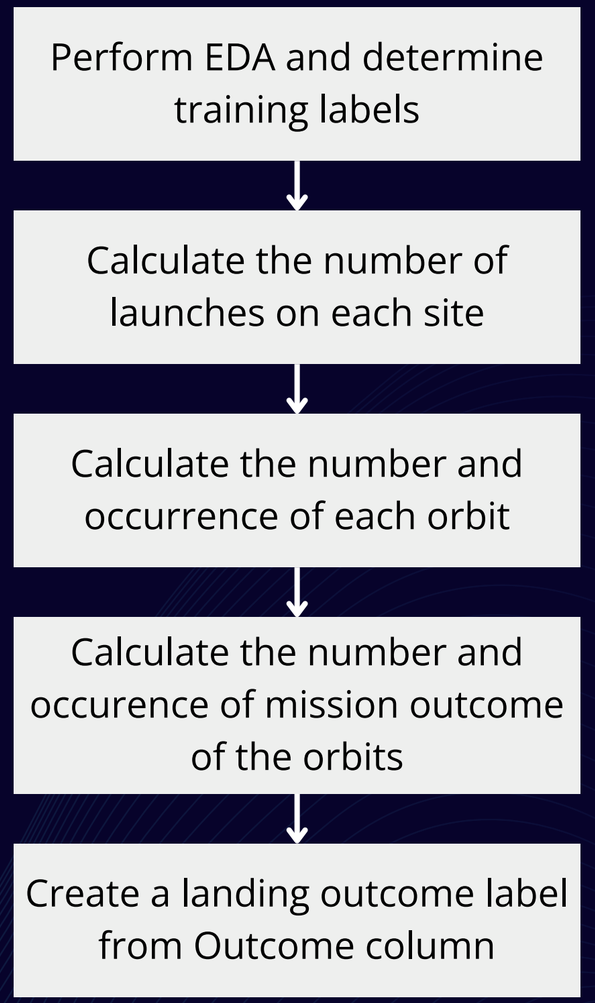
DATA WRANGLING

In the dataset, there are various scenarios where the booster did not land successfully. For instance, if a landing was attempted but failed due to an accident, the outcome differs. "True Ocean" indicates a successful landing in a specific ocean region, while "False Ocean" means the landing attempt in that region was unsuccessful. Similarly, "True RTLS" denotes a successful landing on a ground pad, and "False RTLS" indicates a failed attempt. "True ASDS" signifies a successful landing on a drone ship, whereas "False ASDS" means the attempt on a drone ship was unsuccessful.

These outcomes are converted into training labels: "1" for a successful booster landing and "0" for an unsuccessful one.

Github Link: Data Wrangling

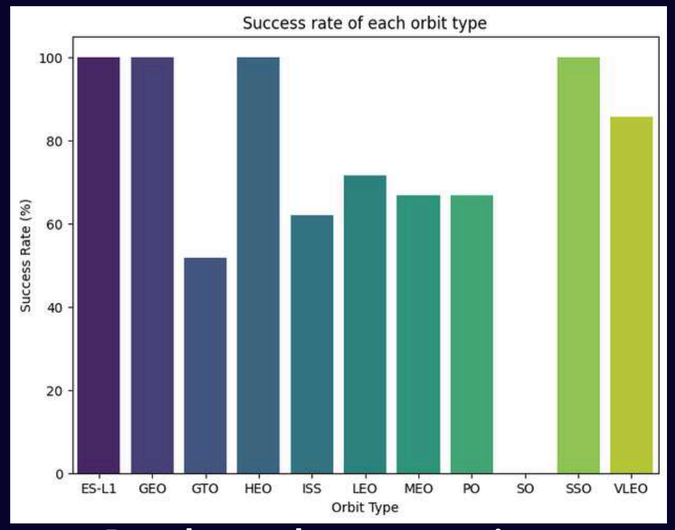
PROCESS



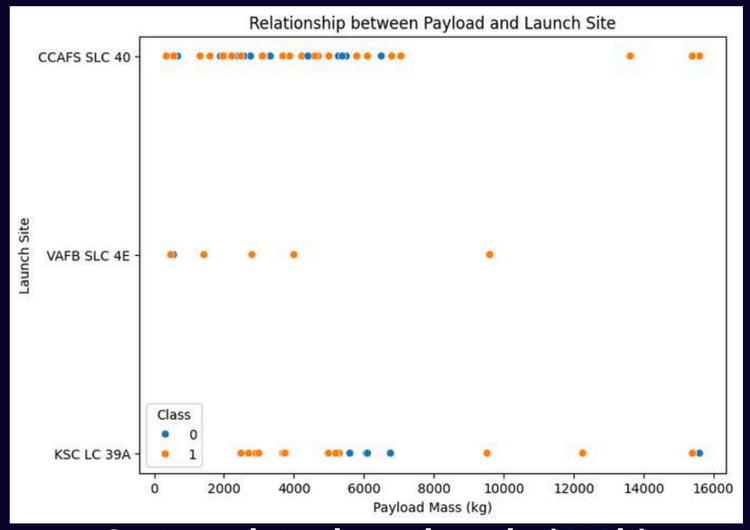
EDA WITH DATA VISUALIZATION

Charts plotted

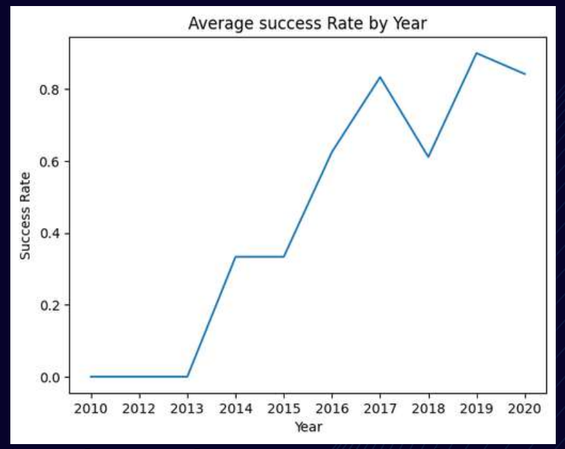
- Flight Number vs Payload Mass
- Flight Number vs Launch Site
- Payload Mass vs Launch Site
- Orbit Type vs Success Rate
- Flight Number vs Orbit Type
- Payload Mass vs Orbit Type
- Success Rate Yearly Trend



Bar charts show comparisons between categories.



Scatter plots show the relationship between variables



Line charts show trends in data over time

Github Link: EDA Data Visualization

EDA WITH SQL

Performed SQL queries

- Displayed the names of the unique launch sites in the space mission
- Displayed 5 records where launch sites begin with the string ‘CCA'
- Displayed the total payload mass carried by boosters launched by NASA (CRS)
- Displayed average payload mass carried by booster version F9 v1.1
- Listed the date when the first successful landing outcome in ground pad was achieved
- Listed the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000
- Listed the total number of successful and failure mission outcomes
- Listed the names of the booster versions which have carried the maximum payload mass
- Listed the failed landing outcomes in drone ship, their booster versions and launch site names for the months in year 2015
- Ranked 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

Github Link: EDA With SQL

BUILD AN INTERACTIVE MAP WITH FOLIUM

Markers of all Launch Sites

- Added Marker with Circle, Popup Label, and Text Label of NASA Johnson Space Center using its latitude and longitude coordinates as a start location
- Added Markers with Circle, Popup Label and Text Label of all Launch Sites using their latitude and longitude coordinates to show their geographical locations and proximity to Equator and coasts

Coloured Markers of the launch outcomes for each Launch Site

- Added coloured Markers of success (Green) and failed (Red) launches using Marker Cluster to identify which launch sites have relatively high success rates

Distances between a Launch Site to its proximities

- Added coloured Lines to show distances between the Launch Site KSC LC-39A and the proximities like Railway, Highway, Coastline and Closest City

Github Link: Interactive Map Folium

BUILD A DASHBOARD WITH PLOTLY DASH

Launch Sites Dropdown List:

- Added dropdown list to enable Launch Site selection

Pie Chart showing Success Launches (All Sites/Certain Site):

- Added a pie chart to show the total successful launches count for all sites and the Success vs Failed counts for the site

Slider of Payload Mass Range:

- Added a slider to select certain Payload range
- Scatter Chart of Payload Mass vs Success Rate for the different Booster Versions:
- Added a scatter chart to show the correlation between Payload and Launch Success

Github Link: Dashboard Plotly Dash

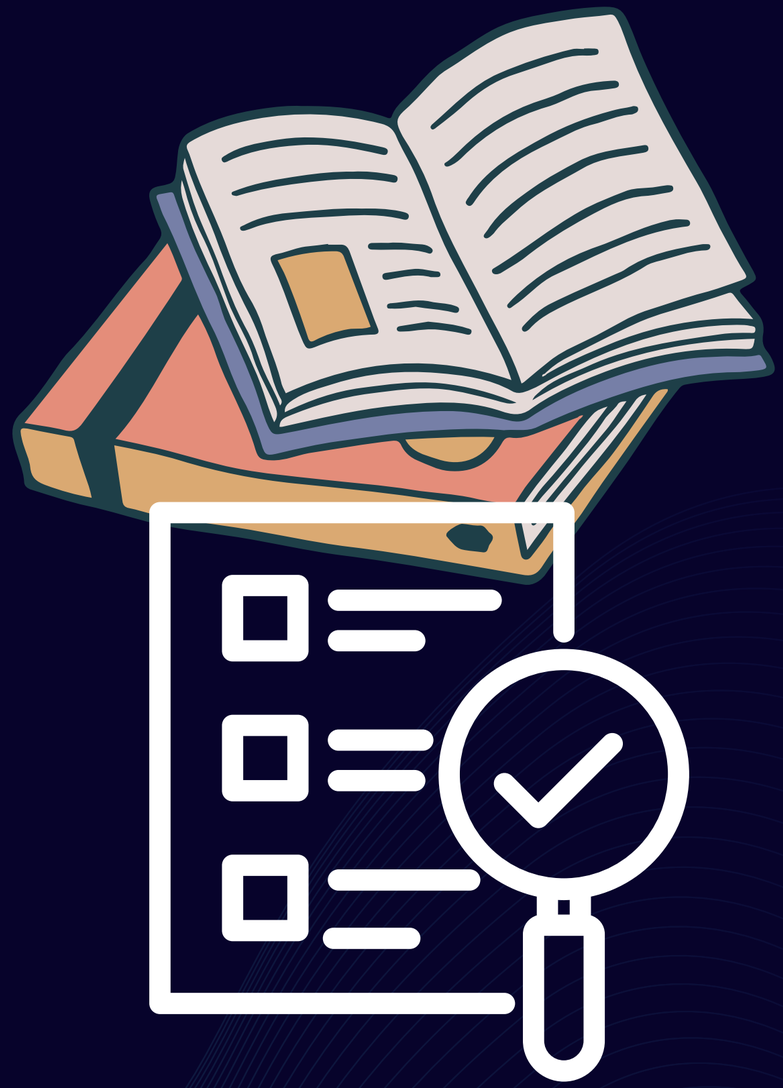
PREDICTIVE ANALYSIS (CLASSIFICATION)

Create a Numpy array from the column “Class” in data → Use StandardScaler for data standarization, then fit and transorm the data → Split the data into training and testing → Create a GridSearchCV object with cv = 10 to find best parameters → Apply the GridSearchCV to LogReg, SVM, Decision Tree, and KNN → Calculate accuracy on the test data using .score() → Plot the confusion matrix for all models → Finding the best perfrmance by examining the Jaccard_score and F1_score metrics

Github Link: Predictive Analysis

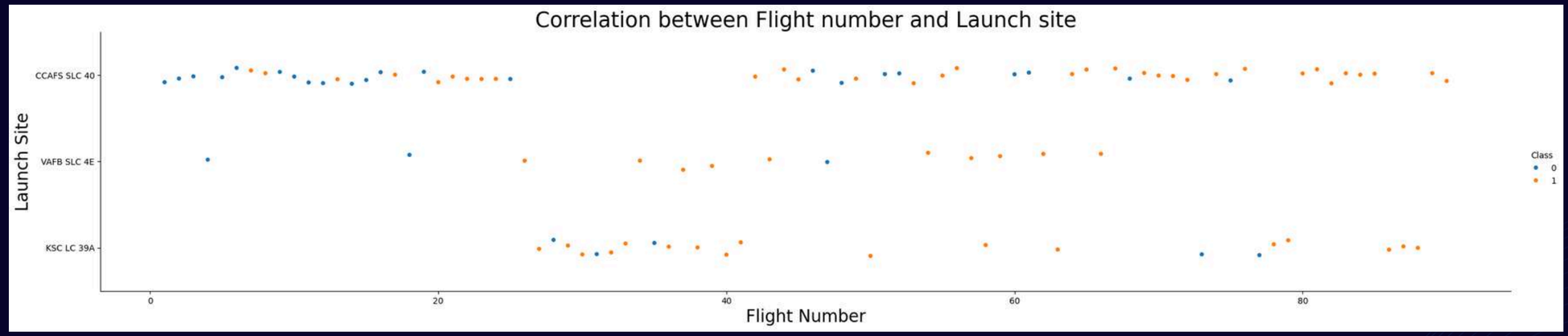
RESULT

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



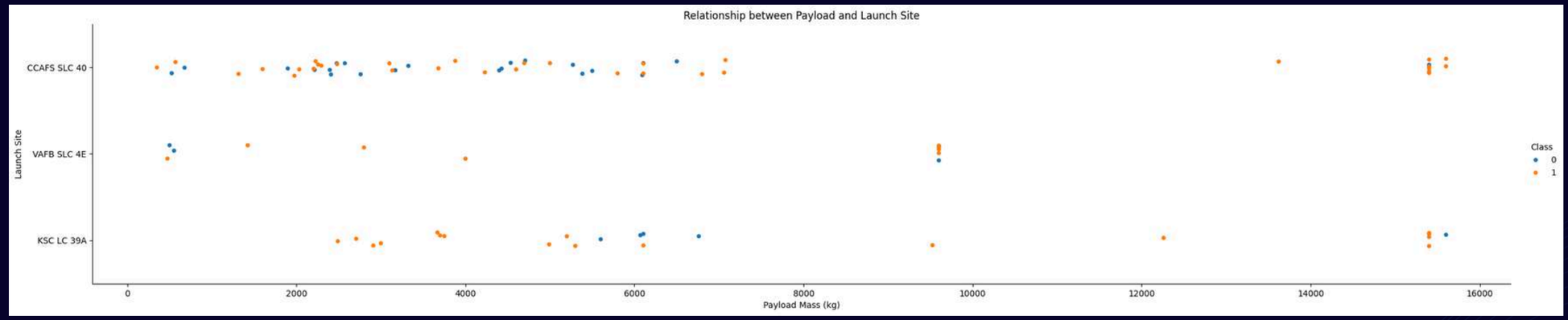
INSIGHT DRAWN FROM EDA

FLIGHT NUMBER VS. LAUNCH SITE



- **We can conclude that CCAFS SLC 40 have the most launch among others**
- **from time to time, the success rate of the launch site increases with having barely any failed attempt**

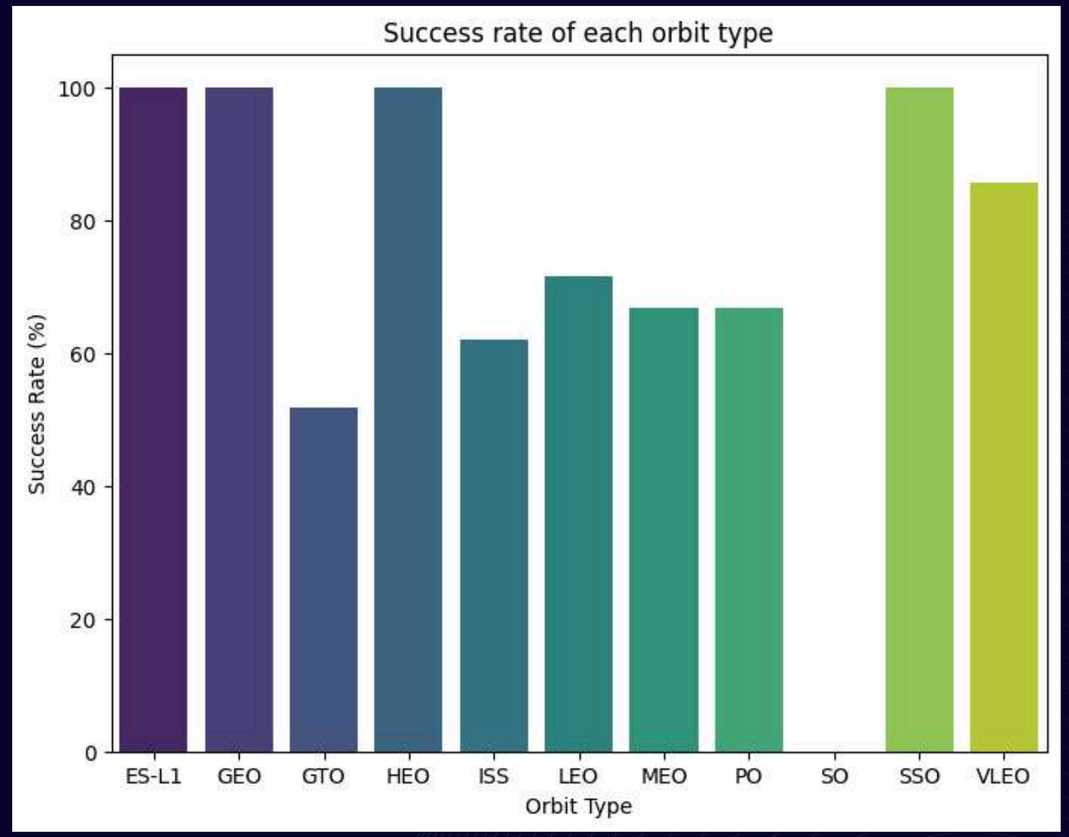
PAYLOAD VS. LAUNCH SITE



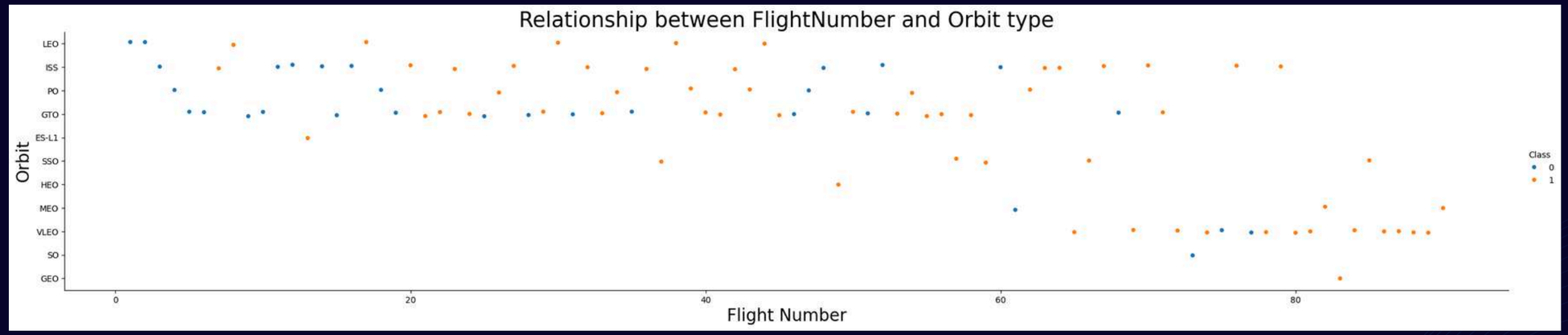
- **The higher the payload mass, the higher the chance of success rate**
- **Most payload mass that exceed 8000kg have a high succes rate**

SUCCESS RATE VS. ORBIT TYPE

- There are 4 orbit’s with a 100% success rate (ES-L1, GEO, HEO, SSO)
- There is only 1 orbit that have a 0% success rate (SO)

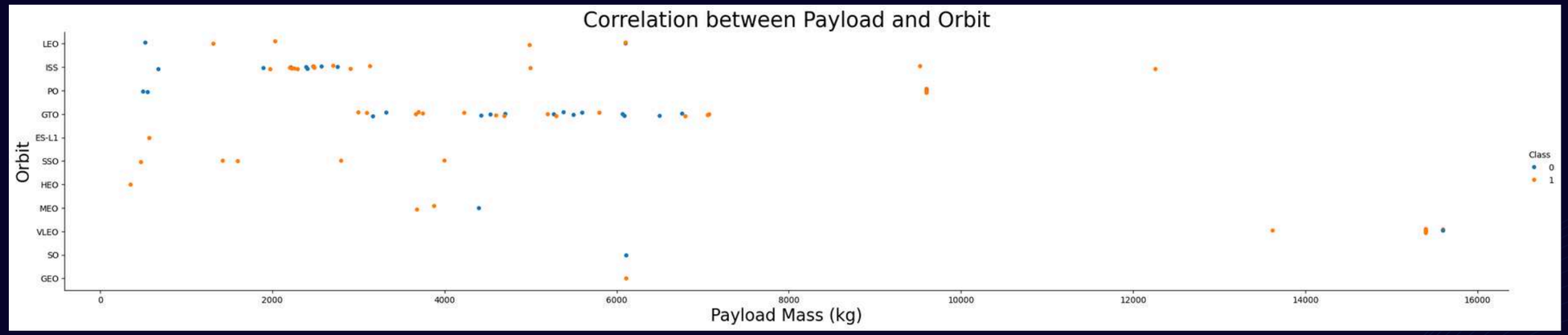


FLIGHT NUMBER VS. ORBIT TYPE



- **Most orbit can be seen having a correlation between flight number and success rate, meanwhile GTO have no correlation at all**

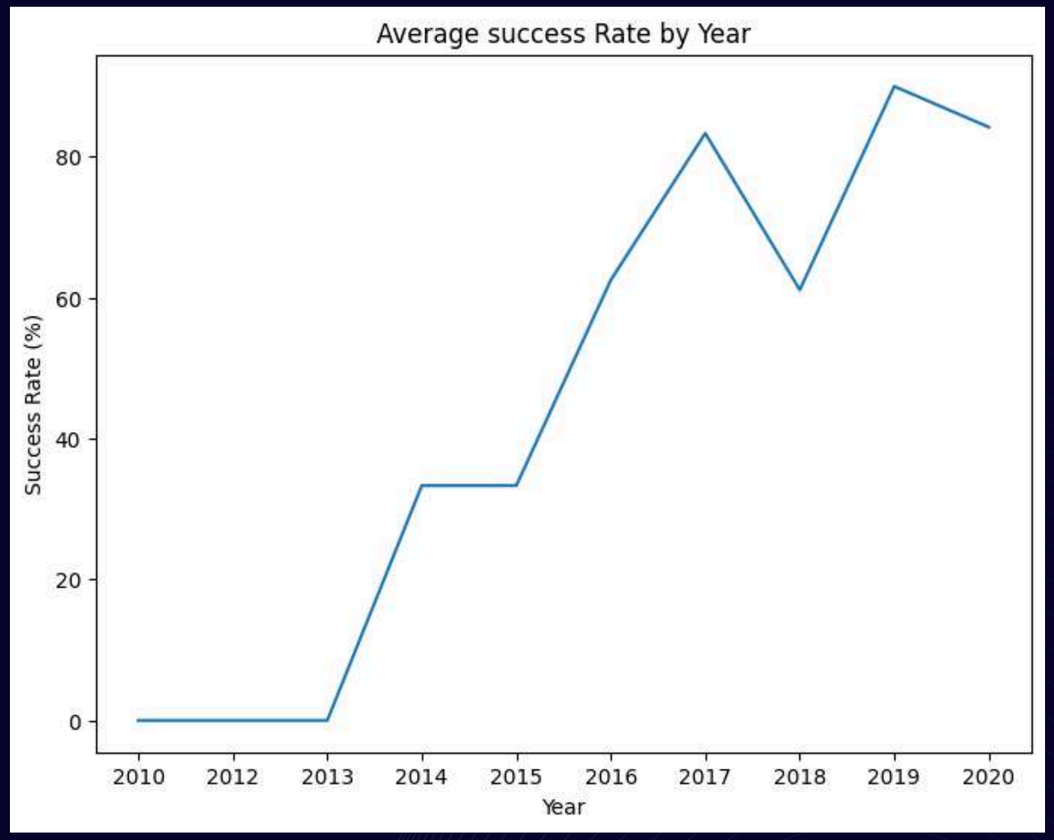
PAYLOAD VS. ORBIT TYPE



- **LEO, ISS, PO, and VLEO can be seen having a higher success rate when having heavy payload mass**
- **GTO doesn’t have any correlation between payload mass**

LAUNCH SUCCESS YEARLY TREND

- From time to time, the launch success rate have increased significantly
- 2019 being the peak success rate



EDA WITH SQL

ALL LAUNCH SITE NAMES

```
[15] result = %sql select distinct launch_site from SPACEXTABLE
     result.DataFrame()
```

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

	Launch_Site
0	CCAFS LC-40
1	VAFB SLC-4E
2	KSC LC-39A
3	CCAFS SLC-40

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

LAUNCH SITE NAMES BEGIN WITH 'CCA'

```
[16] result = %sql select * from SPACEXTABLE where launch_site like 'CCA%' limit 5;
     result.DataFrame()
```

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

	Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
0	2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
1	2010-12-08	15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight C1, two CubeSats, barrel of...	0	LEO (ISS)	NASA (COTS) NRO	Success	Failure (parachute)
2	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
3	2012-10-08	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
4	2013-03-01	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

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

TOTAL PAYLOAD MASS

```
[17] result = %sql select sum(payload_mass__kg_) as total_payload_mass from SPACEXTABLE where customer = 'NASA (CRS)';
     result.DataFrame()
```

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

	total_payload_mass
0	45596

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

AVERAGE PAYLOAD MASS BY F9 V1.1

```
[18] result = %sql select avg(payload_mass__kg_) as average_payload_mass from SPACEXTABLE where booster_version like '%F9 v1.1%';
     result.DataFrame()
```

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

	average_payload_mass
0	2534.666667

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

FIRST SUCCESSFUL GROUND LANDING DATE

```
[19] result = %sql select min(date) as first_successful_landing from SPACEXTABLE where Landing_Outcome = 'Success (ground pad)';
     result.DataFrame()
```

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

first_successful_landing
2015-12-22

List the date when the first successful landing outcome in ground pad was achieved

SUCCESSFUL DRONE SHIP LANDING WITH PAYLOAD BETWEEN 4000 AND 6000

```
[20] result = %sql select booster_version from SPACEXTABLE where Landing_Outcome = 'Success (drone ship)' and payload_mass__kg_ between 4000 and 6000;
     result.DataFrame()
```

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

	Booster_Version
0	F9 FT B1022
1	F9 FT B1026
2	F9 FT B1021.2
3	F9 FT B1031.2

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

TOTAL NUMBER OF SUCCESSFUL AND FAILURE MISSION OUTCOMES

```
[21] result = %sql select mission_outcome, count(*) as total_number from SPACEXTABLE group by mission_outcome;
     result.DataFrame()
```

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

	Mission_Outcome	total_number
0	Failure (in flight)	1
1	Success	98
2	Success	1
3	Success (payload status unclear)	1

List the total number of successful and failure mission outcomes.

BOOSTERS CARRIED MAXIMUM PAYLOAD

```
[22] result = %sql select booster_version from SPACEXTABLE where payload_mass__kg_ = (select max(payload_mass__kg_) from SPACEXTABLE);
     result.DataFrame()
```

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

	Booster_Version
0	F9 B5 B1048.4
1	F9 B5 B1049.4
2	F9 B5 B1051.3
3	F9 B5 B1056.4
4	F9 B5 B1048.5
5	F9 B5 B1051.4
6	F9 B5 B1049.5
7	F9 B5 B1060.2
8	F9 B5 B1058.3
9	F9 B5 B1051.6
10	F9 B5 B1060.3
11	F9 B5 B1049.7

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

2015 LAUNCH RECORDS

```
[ ] %%sql select substr(date, 6, 2) as month, date, booster_version, launch_site, Landing_Outcome from SPACEXTABLE
      where Landing_Outcome = 'Failure (drone ship)' and substr(date, 1, 4) = '2015';
```

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

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

List the failed landing outcomes in drone ship, their booster versions and launch site names for the months in year 2015

RANK LANDING OUTCOMES BETWEEN 2010-06-04 AND 2017-03-20

```
%%sql select Landing_Outcome, count(*) as count_outcomes from SPACEXTABLE
      where date between '2010-06-04' and '2017-03-20'
      group by Landing_Outcome
      order by count_outcomes desc;
```

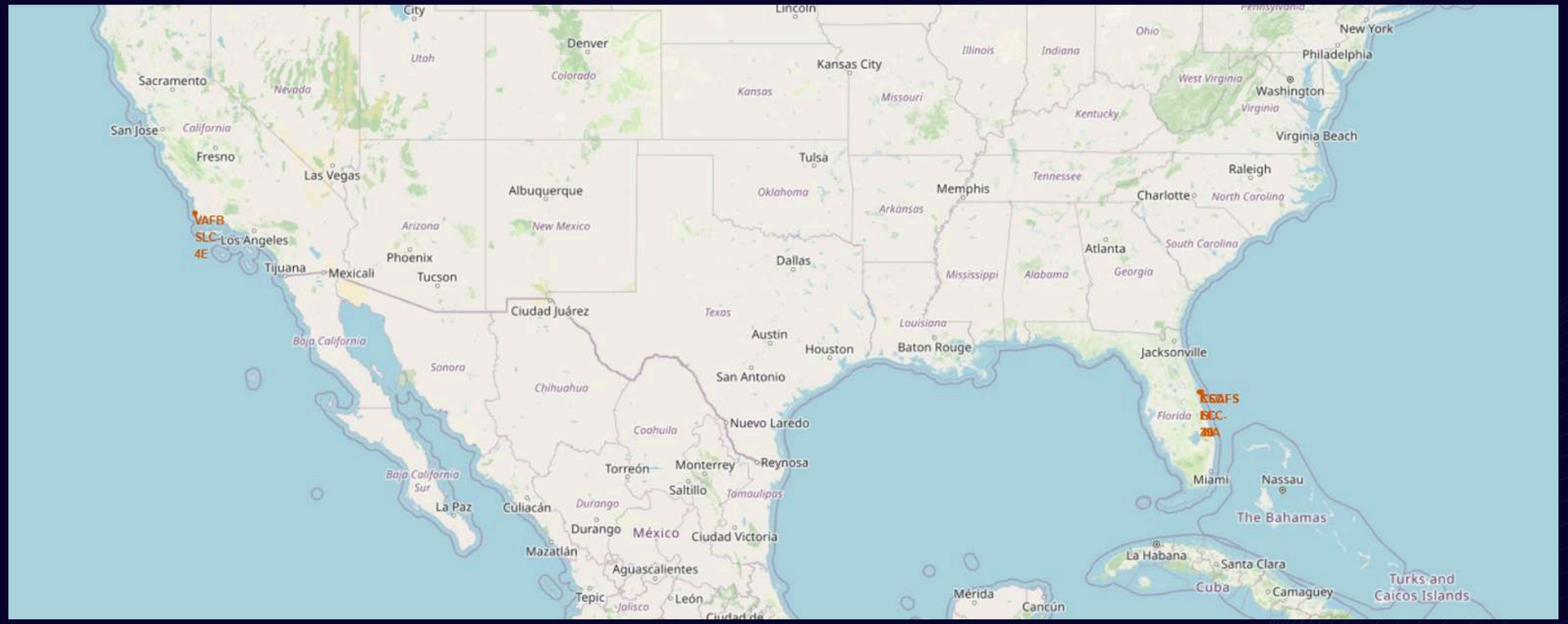
```
 * sqlite:///my_data1.db
Done.
```

Landing_Outcome	count_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

Ranking 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

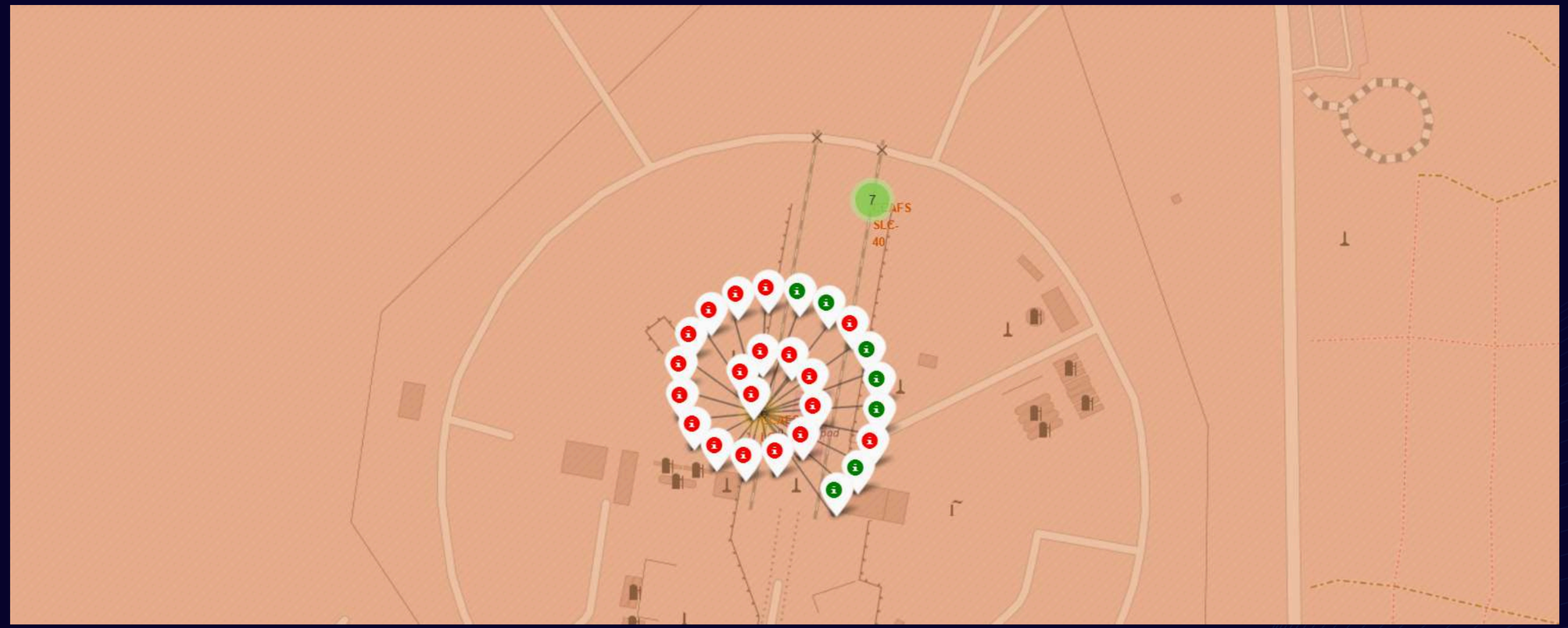
LAUNCH SITE PROXIMITIES ANALYSIS

DISPLAY MARKED MAP



Most launch site are at the edge of a map, because it help minimize danger to the people living nearby

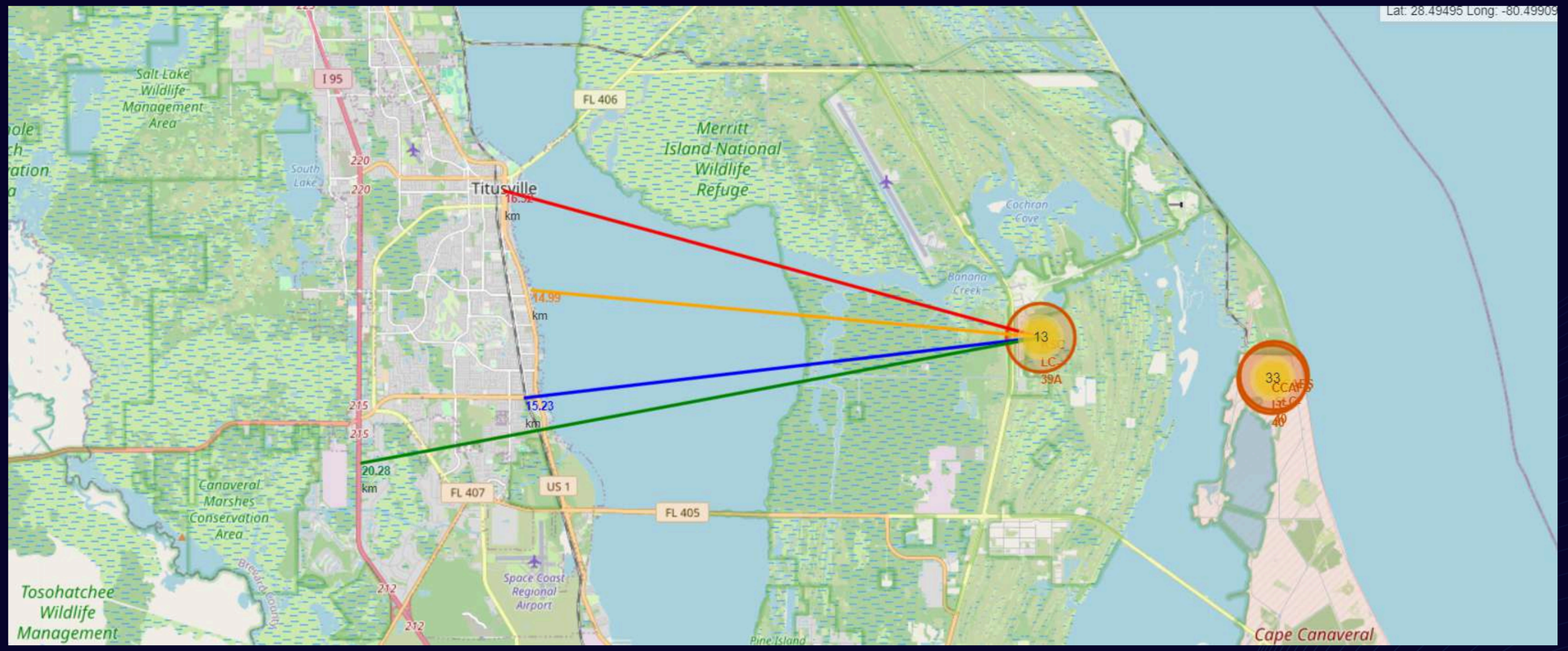
COLOR LABELED LAUNCH RECORD



Color label:

- **Green -> success**
- **Red -> fail**

LAUNCH SITE DISTANCE TO ITS PROXIMITIES

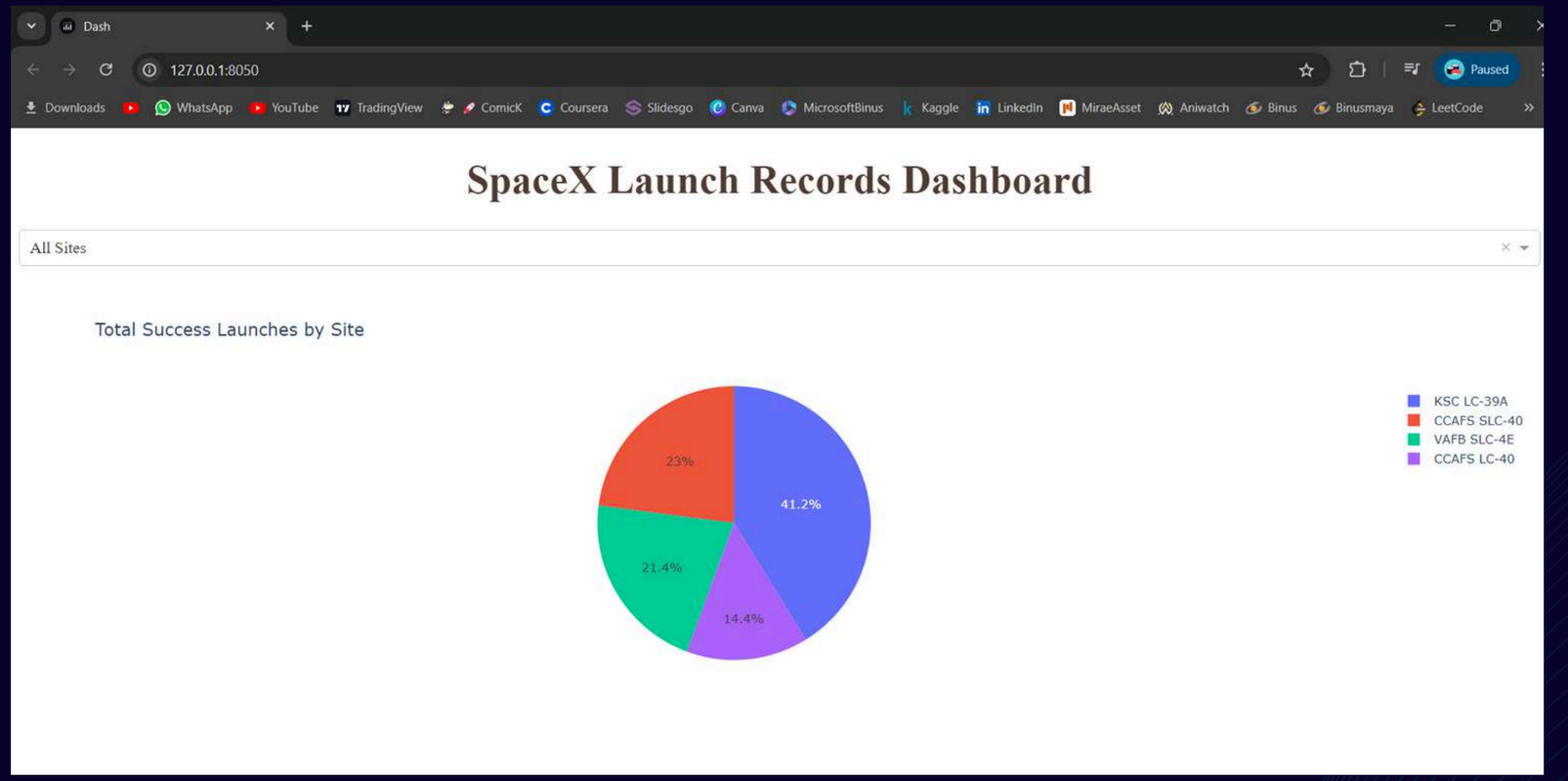


Launch site KSC LC-39A:

- **relative close to railway (15.23 km)**
- **relative close to highway (20.28 km)**
- **relative close to coastline (14.99 km)**
- **relative close to Titusville (16.32 km)**

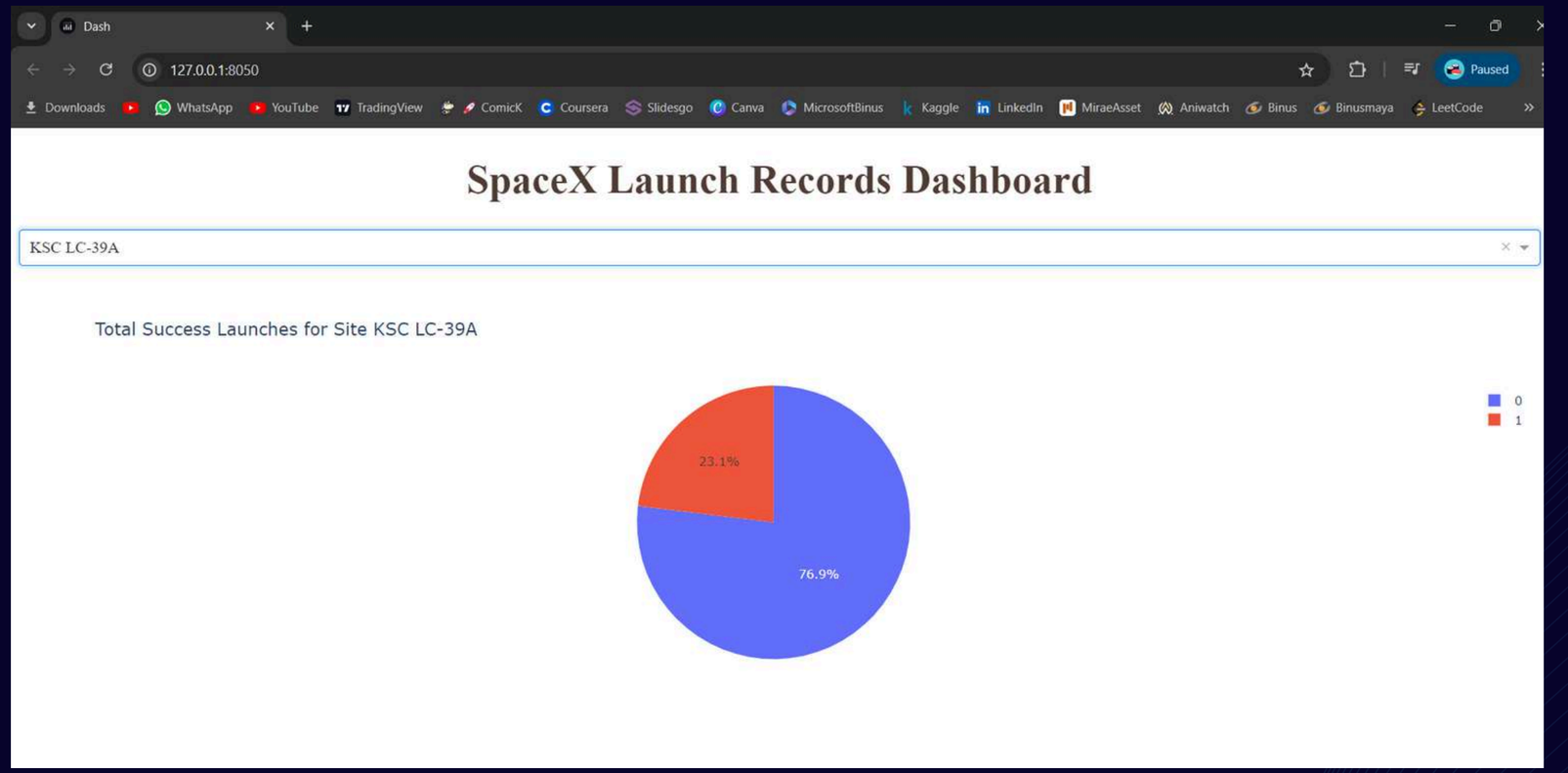
BUILD A DASH BOARD WITH PLOTLY DASH

LAUNCH SUCCESS COUNT FOR ALL SITES



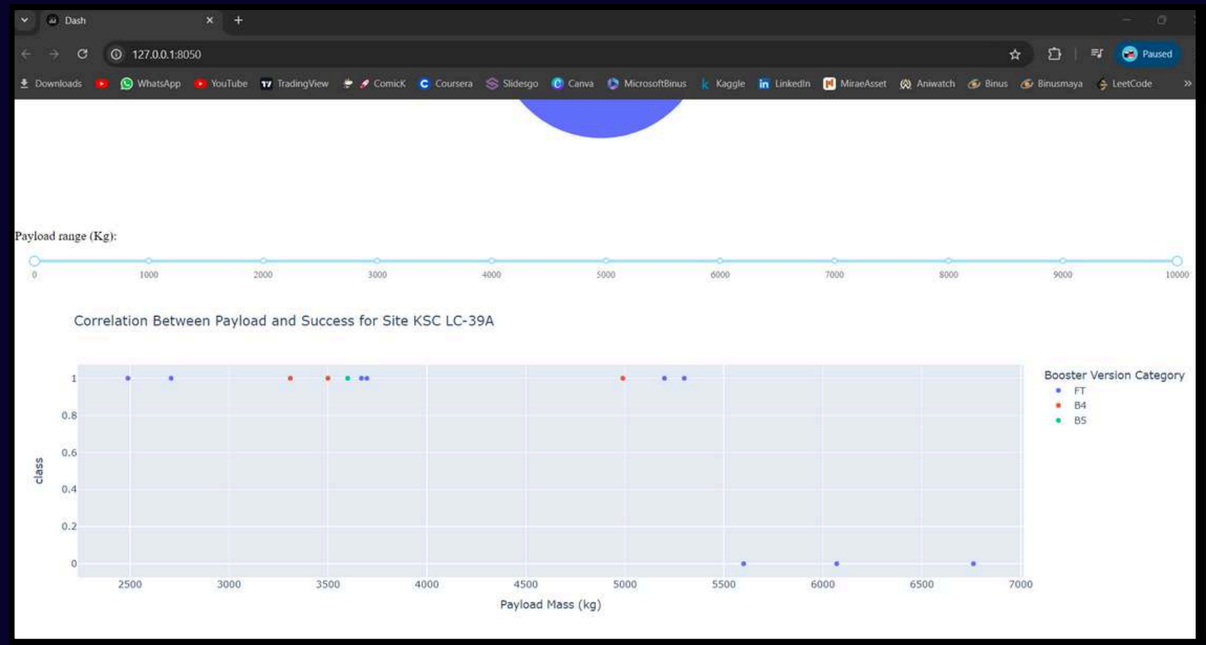
From the pie chart, we can conclude that KSC LC-39A have the highest success rate

LAUNCH SITE WITH HIGHEST SUCCESS RATIO



KSC LC-39A has the highest launch success rate which is 76.9%, with a record of 10 successful and 3 failed landings

LAUNCH SITE WITH HIGHEST SUCCESS RATIO



all slider (0-10000)

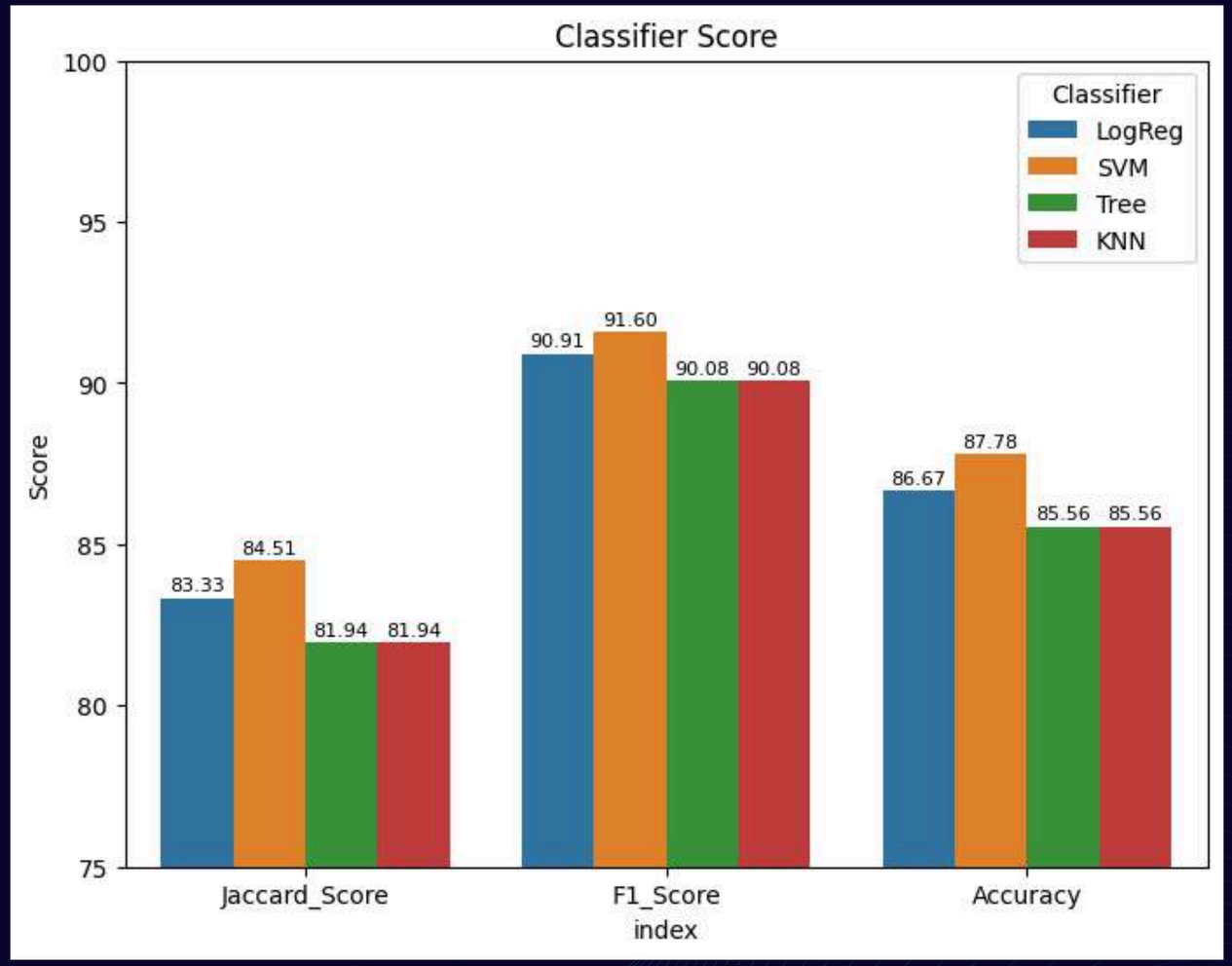
certain slider (0-4000)

This chart shows that payloads within 0 and 5000kg have the highest success rate

PREDICTIVE ANALYSIS (CLASSIFICATION)

COLOR LABELED LAUNCH RECORD

The score of the whole Dataset confirms that the best model is Support Vector Machine (SVM). This model has the highest score on every metrics category.

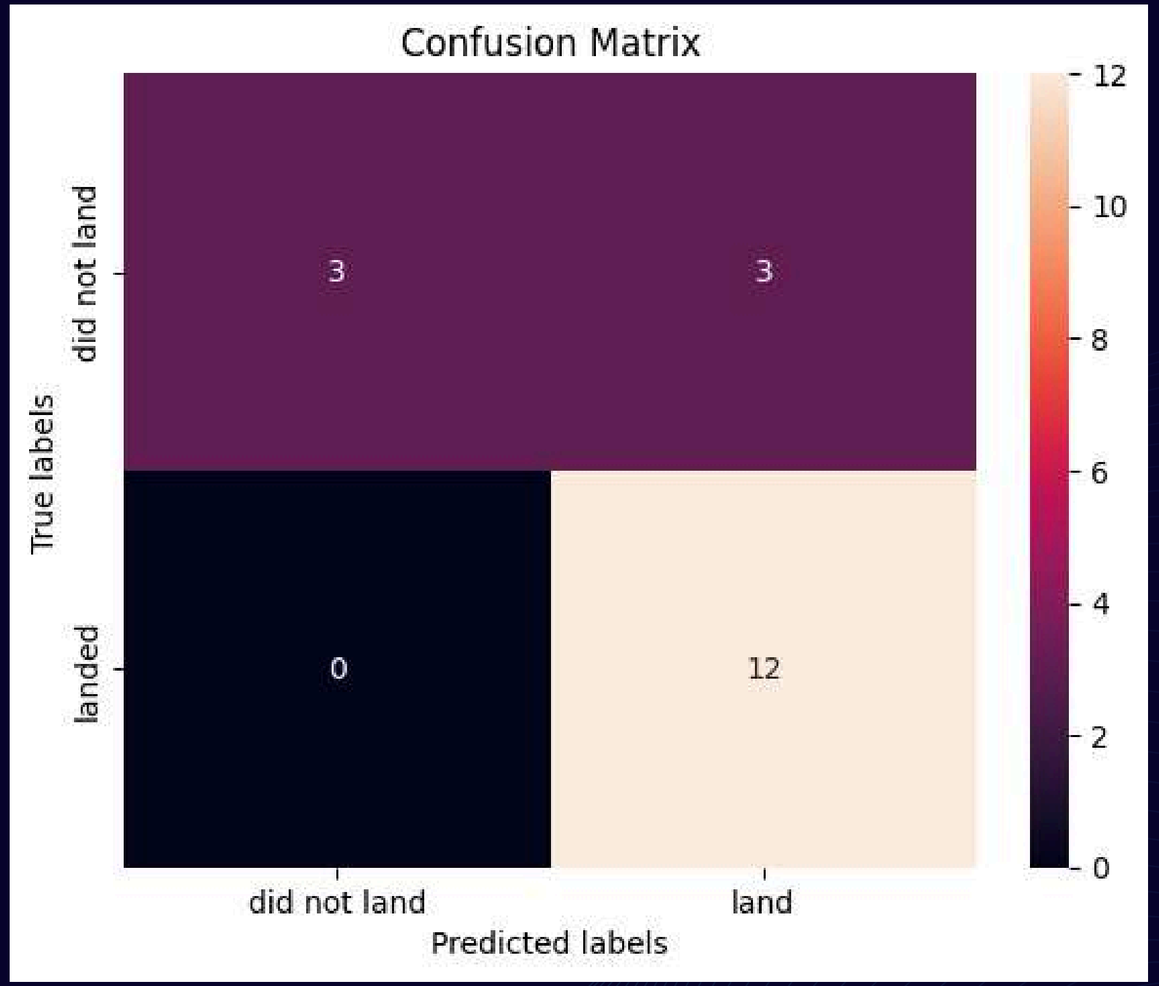


Classifier score on full dataset

COLOR LABELED LAUNCH RECORD

We can see that SVM can mostly distinguish between the different classes. Everything is correct except the False Positive (FP)

		Actual	
		Positive	Negative
Predicted	Positive	True Positive	False Positive
	Negative	False Negative	True Negative



SVM test_dataset Confussion Matrix

CONCLUSION

01 SVM is the best machine learning model for this dataset

02 Launch with higher payload mass and flight number has a significantly higher success rate

03 Average success rate for every launch have been increasing over the past years

03 Most launch site are postitioned on the edge of a country, to ensure safety

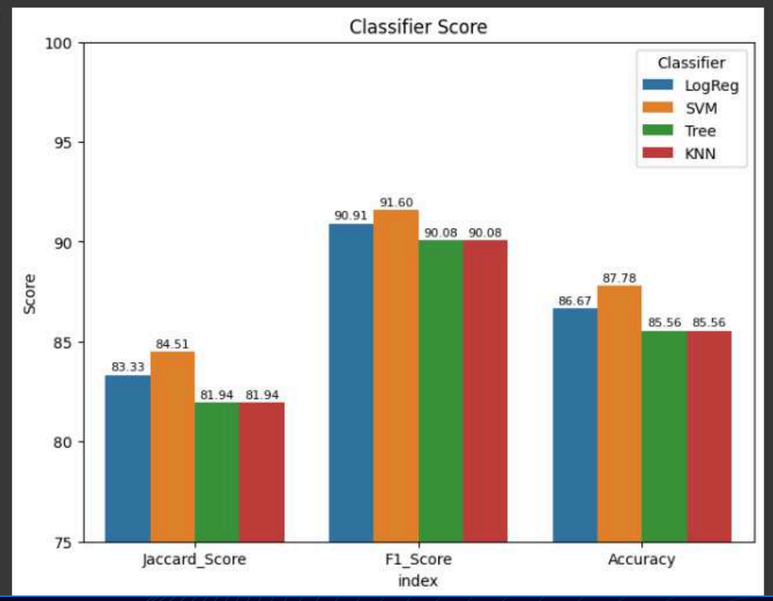
APPENDIX

```
data = scores_full.reset_index()

data_score_full = data.melt(id_vars="index", var_name="Classifier", value_name="Score")
data_score_full['Score'] = data_score_full['Score']*100
data_score_full
```

	index	Classifier	Score
0	Jaccard_Score	LogReg	83.333333
1	F1_Score	LogReg	90.909091
2	Accuracy	LogReg	86.666667
3	Jaccard_Score	SVM	84.507042
4	F1_Score	SVM	91.603053
5	Accuracy	SVM	87.777778
6	Jaccard_Score	Tree	81.944444
7	F1_Score	Tree	90.076336
8	Accuracy	Tree	85.555556
9	Jaccard_Score	KNN	81.944444
10	F1_Score	KNN	90.076336
11	Accuracy	KNN	85.555556

```
plt.figure(figsize = (8,6))
ax = sns.barplot(data=data_score_full, x='index', y='Score', hue='Classifier')
plt.title('Classifier Score')
plt.ylim(75, 100)
for p in ax.patches:
    ax.annotate(f'{p.get_height():.2f}', # retrieves the height of the bar (the y-value)
                (p.get_x() + p.get_width() / 2., p.get_height()), # calculates the x-position at the center of the bar
                ha = 'center', va = 'center', # center the text
                xytext = (0, 5), # place the text 5 points above the bar.
                textcoords = 'offset points',
                fontsize = 8) # change font size
plt.show()
```



Extra code not included in the notebook

SPECIAL THANKS TO

- IBM Institute
- Coursera