
IBM Developer
SKILLS NETWORK
WINNING SPACE RACE
WITH DATA SCIENCE
SPACEX PROJECT
MESUT SUHAN SISMAN
24.06.2023

Outline

EXECUTIVE SUMMARY

The IBM Applied Data Science Capstone project focuses on analyzing SpaceX data to advance space technology and exploration. The project applies data science techniques to SpaceX's launches, missions, rockets, and other relevant data.

In this project, data is collected using SpaceX REST API and Wikipedia web pages, followed by data preprocessing to determine launch success outcomes. Exploratory data analysis and visualization techniques are employed to examine relationships and patterns in the dataset.

Based on the data analysis, an interactive dashboard and maps are created. The dashboard displays success/failure rates and the relationship between rocket versions, payload mass, and launch sites. The maps illustrate the locations of launch sites and their corresponding success outcomes.

Furthermore, four machine learning algorithms are utilized to make success predictions. These algorithms include Logistic Regression, Support Vector Machine, Decision Tree, and K Nearest Neighbors classification methods. The Decision Tree Classification method yields the highest success rate during the evaluation The utilization of data science techniques emphasizes their significance in the field of space technology

INTRODUCTION

This project aims to predict the probability of a rocket successfully landing. SpaceX, with its ability to reuse the first stage of Falcon 9 rockets, offers lower launch costs compared to other providers. Therefore, determining the successful landing of the first stage is crucial in estimating the cost of a launch. In this project, we utilize APIs and web scraping techniques to obtain insights from the SpaceX data, as the company does not provide the data directly.

The main objective of this project is to predict the successful landing probability of a rocket using the insights derived from the SpaceX data set. The data set includes variables such as payload mass, orbit, booster version, launch sites, and their geographic locations. Analyzing and understanding these relationships will form the foundation for developing machine learning algorithms in the later stages of the project.

We employ popular machine learning algorithms such as Logistic Regression, Support Vector Machine, Decision Tree, and K Nearest Neighbors in this project. The data set is split into training and test sets, utilizing data collected from APIs and web scraping techniques. The models are trained on the training data and evaluated using the test data. Based on the evaluation results, the Decision Tree Classification Model performs the best, achieving an accuracy of 88.89%.

The findings of this study provide valuable insights in the fields of rocket design, launch planning, and cost estimation. Even in the absence of direct access to SpaceX data, we demonstrate that reliable predictions can be made using APIs and web scraping techniques. Further analysis and optimization of the results will be pursued in the subsequent stages of the project.

This project holds significant potential for all stakeholders aiming to advance space exploration and technology.

SECTION 1

METHODOLOGY

Executive Summary

- Data collection methodology:
  - Gathered data from SpaceX public API and by scrapping SpaceX Wikipedia page
- Perform data wrangling
  - Classifying true landings as successful and unsuccessful otherwise
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - We tuned the models using GridSearchCV

Data collection

Data collection process involved a combination of API requests from Space X public API and web scraping data from a table in Space X's Wikipedia entry.

The next slides will show the flowchart of data collection from API and the one after will show the flowchart of data collection from webscraping

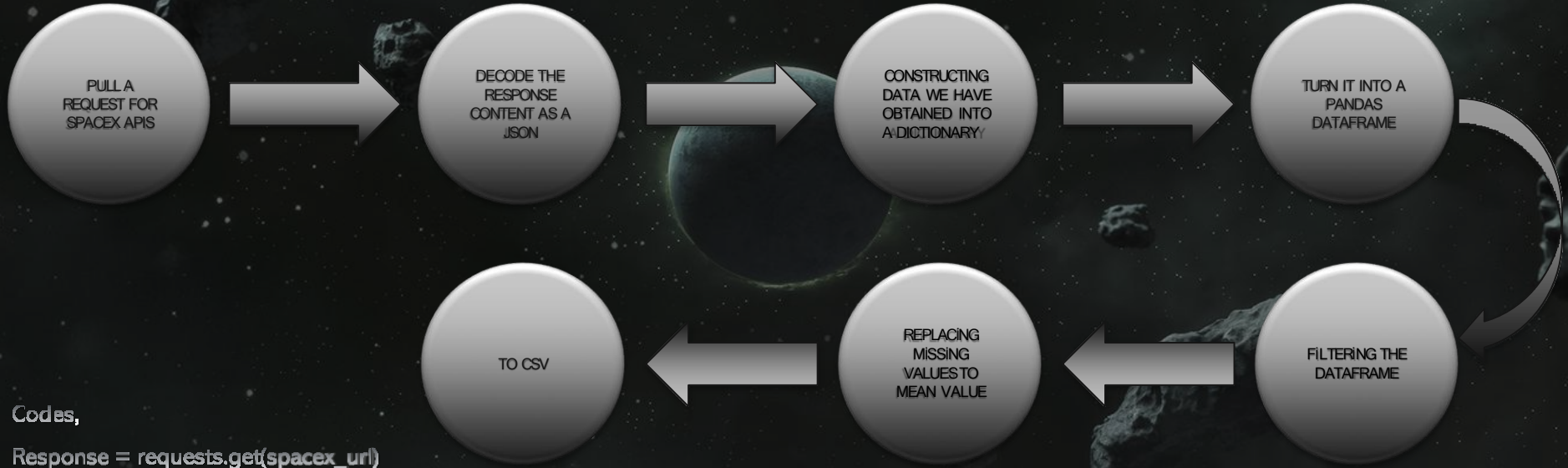
Space X API Data Columns:

FlightNumber, Date, BoosterVersion, PayloadMass, Orbit, LaunchSite, Outcome, Flights, GridFins, Reused, Legs, LandingPad, Block, ReusedCount, Serial, Longitude, Latitude

Wikipedia Web Scrapping Data Columns:

Flight No., Launch site, Payload, PayloadMass, Orbit, Customer, Launch outcome, Version Booster, Booster landing, Date, Time

Data Collection – SpaceX API

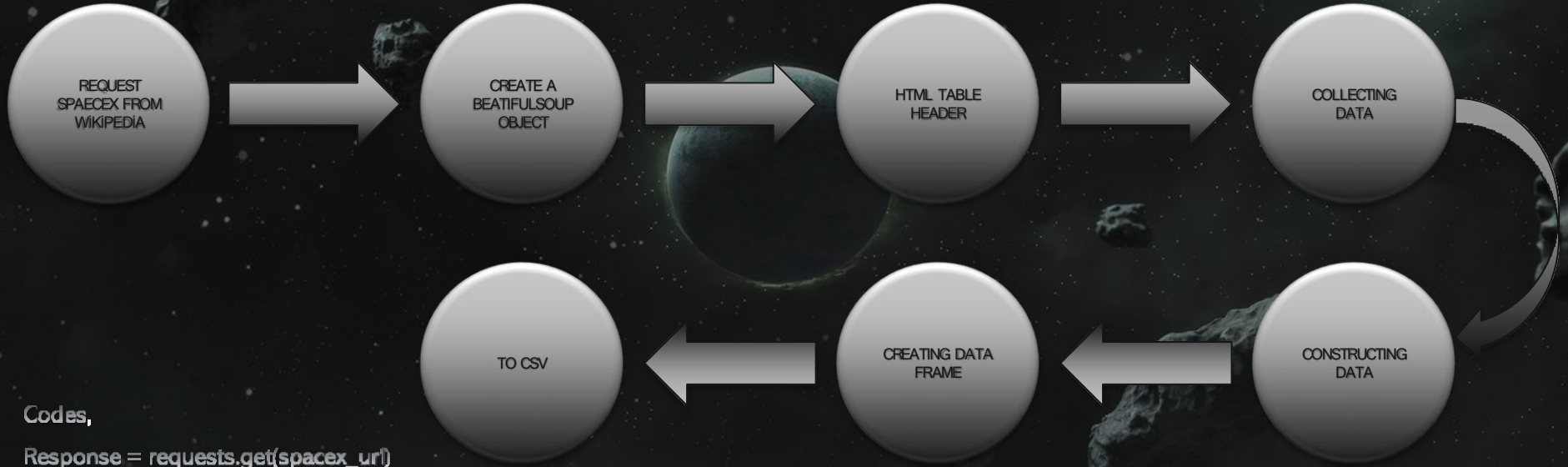


Codes,

```
Response = requests.get(spacex_url)
data = response.json()
data = pd.json_normalize(data)
data.head()
```

GitHub URL:https://github.com/Mesutssmn/IBM_SpaceX_Data_Science_Capstone/blob/main/SpaceX_Data_Collection_API.ipynb

Data Collection – Web Scrapping



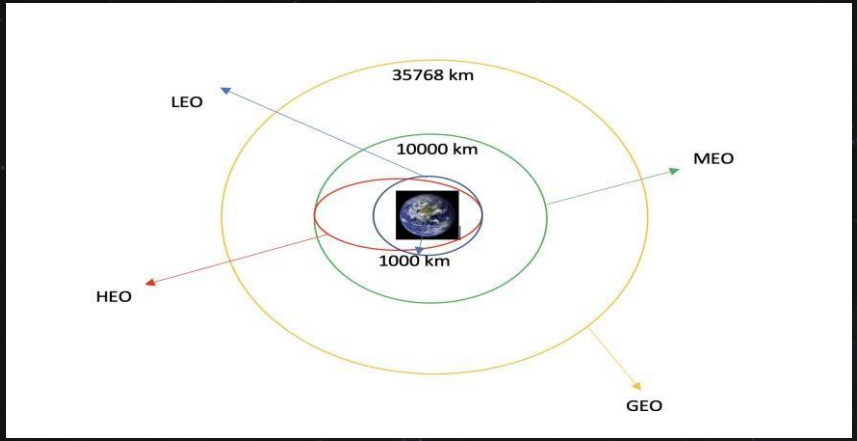
Codes,

```
Response = requests.get(spacex_url)
data = response.json()
data = pd.json_normalize(data)
data.head()
```

GitHub URL:https://github.com/Mesutssmn/IBM SpaceX Data Science Capstone/blob/main/SpaceX%20Data_Collection_Web_Scraping.ipynb

Data Wrangling

In this project, a new training label has been created by combining the 'Mission Outcome' and 'Landing Location' components of the 'Outcome' column. The new label, called 'class', is set to 1 if the 'Mission Outcome' is True, indicating a successful landing, and set to 0 otherwise, indicating a failure. This allows us to classify the landing outcomes as either successful (1) or failure (0) based on the given criteria. This labeling approach is used to predict the probability of a rocket's successful or failed landing based on the outcome information.



Codes,

```
Response = requests.get(spacex_url)
data = response.json()
data = pd.json_normalize(data)
data.head()
```

GitHub URL:https://github.com/Mesutssmn/IBM SpaceX Data Science Capstone/blob/main/SpaceX_Data_Wrangling.ipynb

EDA with Data Visualization

Exploratory Data Analysis performed on variables Flight Number, Payload Mass, Launch Site, Orbit, Class and Year.

Plots Used:

Flight Number vs. Payload Mass, Flight Number vs. Launch Site, Payload Mass vs. Launch Site, Orbit vs. Success Rate, Flight Number vs. Orbit, Payload vs Orbit, and Success Yearly Trend

Scatter plots, line charts, and bar plots were used to compare relationships between variables to decide if a relationship exists so that they could be used in training the machine learning model

GitHub URL:https://github.com/Mesutssrn/IBM SpaceX Data Science Capstone/blob/main/SpaceX%20EDA_Data_Visualization.ipynb

EDA WITH SQL

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

GitHub URL:https://github.com/Mesutssmn/IBM SpaceX Data Science Capstone/blob/main/SpaceX%20EDA_with_SQL.ipynb

Build an Interactive Map with Folium

Folium maps mark Launch Sites, successful and unsuccessful landings, and a proximity example to key locations: Railway, Highway, Coast, and City.

This allows us to understand why launch sites may be located where they are. Also visualizes successful landings relative to location.

GitHub URL: https://github.com/Mesutssmn/IBM_SpaceX_Data_Science_Capstone/blob/main/SpaceX_Generating_Maps_with_Folium.ipynb

Build a Dashboard with Plotly Dash

Launch Sites Dropdown List:

Added a dropdown list to enable Launch Site selection. Pie Chart showing Success Launches (All Sites/Certain Site):

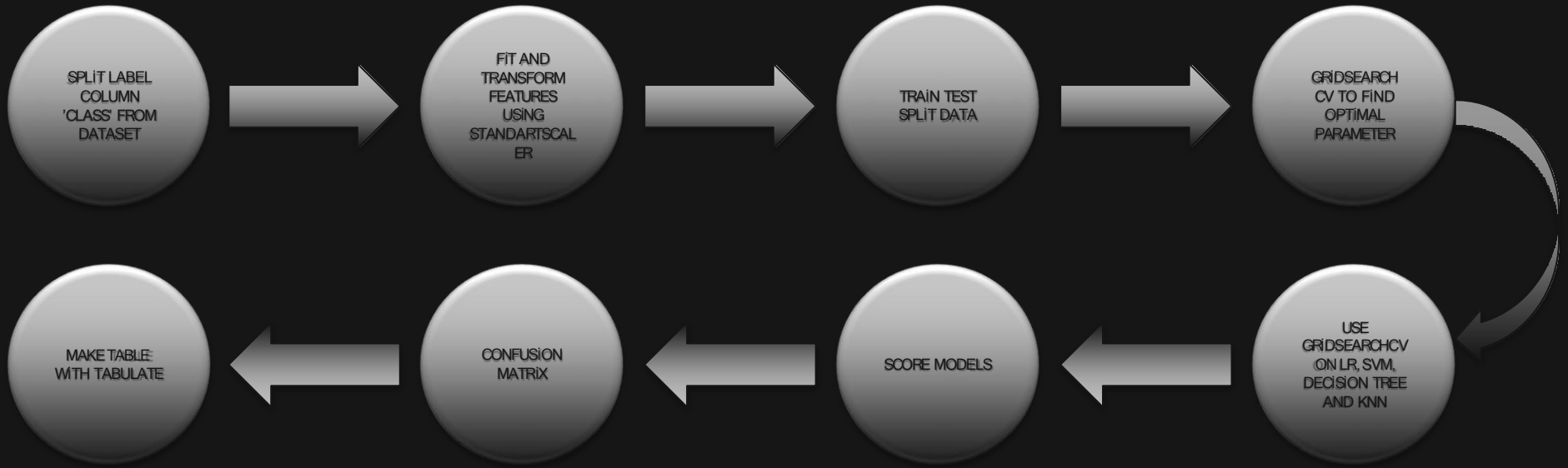
Added a pie chart to show the total successful launches count for all sites and the Success vs. Failed counts for the site, if a specific Launch Site was selected. Slider of Payload Mass Range:

Added a slider to select 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.

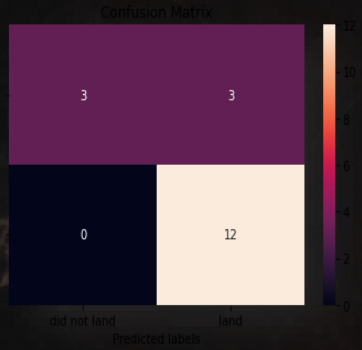
GitHub URL: https://github.com/Mesutssmn/IBM_SpaceX_Data_Science_Capstone/blob/main/SpaceX_Dash_app.ipynb

PREDICTIVE ANALYSIS(CLASSIFICATION)



GitHub URL:https://github.com/Mesutssmn/IBM SpaceX Data Science Capstone/blob/main/SpaceX_Machine_Learning_Prediction.ipynb

Results

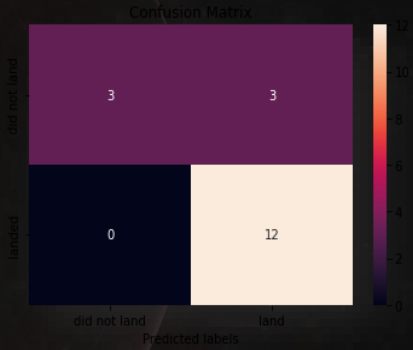


Logaritmic Regression

Accuracy: 83.33%

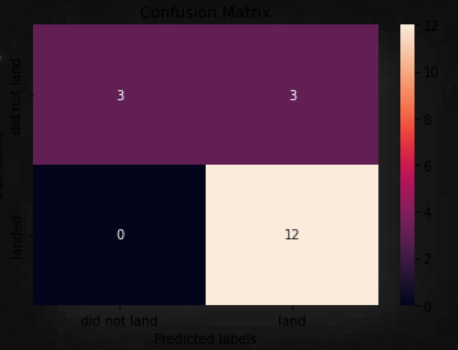
SVM

Accuracy: 83.33%



Decision Tree

Accuracy: 83.33%



KNN

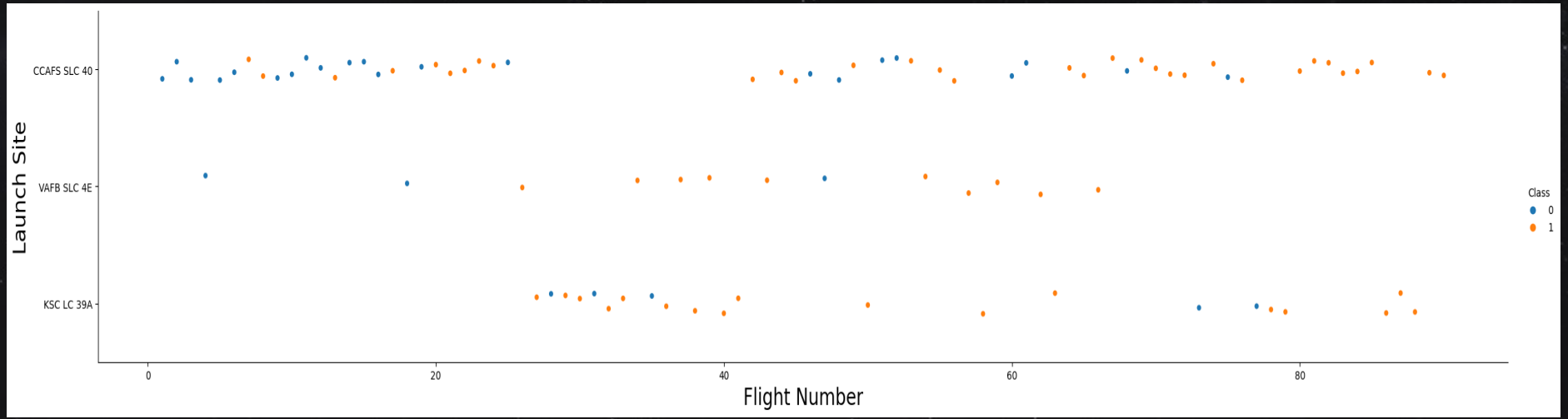
Accuracy: 83.33%

SECTION 2

INSIGHT DRAWN FROM EDA

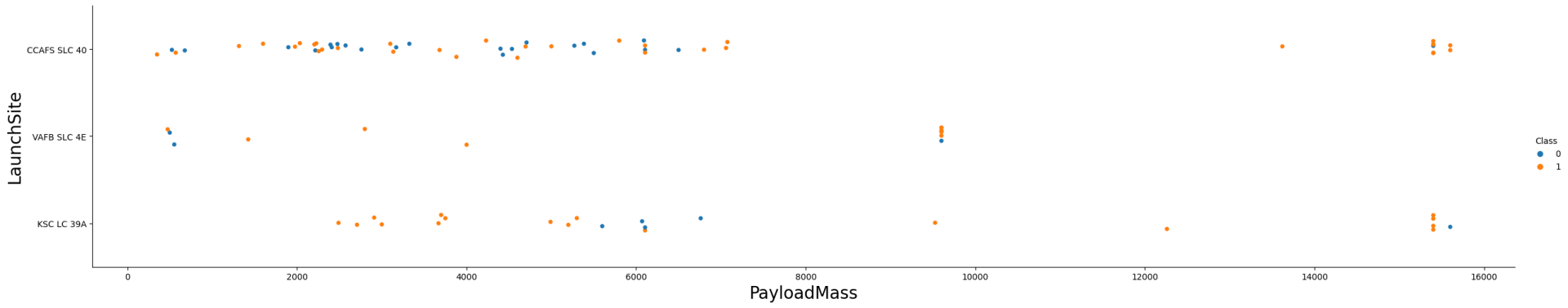
Flight Number vs. Launch Site

Blue indicates successful launch and orange indicates unsuccessful launch. Graphic suggests an increase in success rate over time (indicated in Flight Number). Likely a big breakthrough around flight 20 which significantly increased success rate.
CCAFS appears to be the main launch site as it has the most volume.



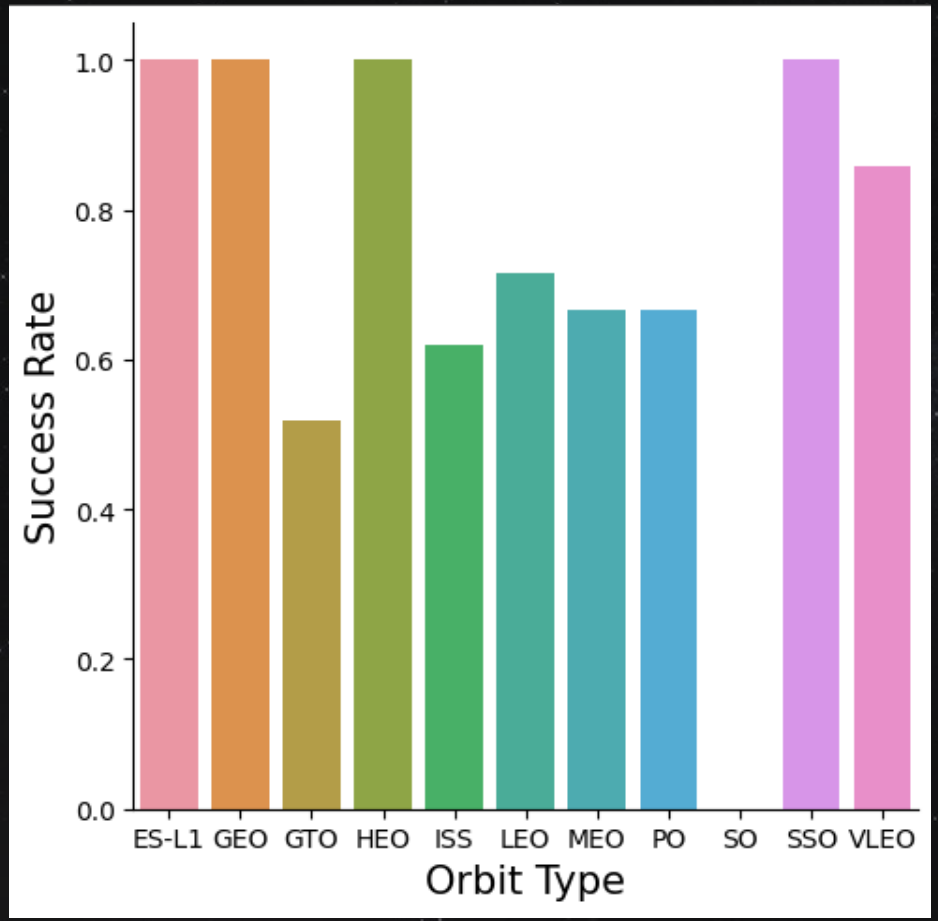
Payload vs. Launch Site

Blue indicates successful launch and orange indicates unsuccessful launch. Payload mass appears to fall mostly between 0 6000 kg. Different launch sites also seem to use different payload mass.



Success Rate vs. Orbit Type

- ES L1, GEO, HEO have 100% success rate, each of them has one sample
- SSO has 100% success rate with 5 samples
- VLEO has 85% success rate
- SO has 0% success rate
- GTO has the around 50% success rate but it has 27 samples



Flight Number vs. Orbit Type

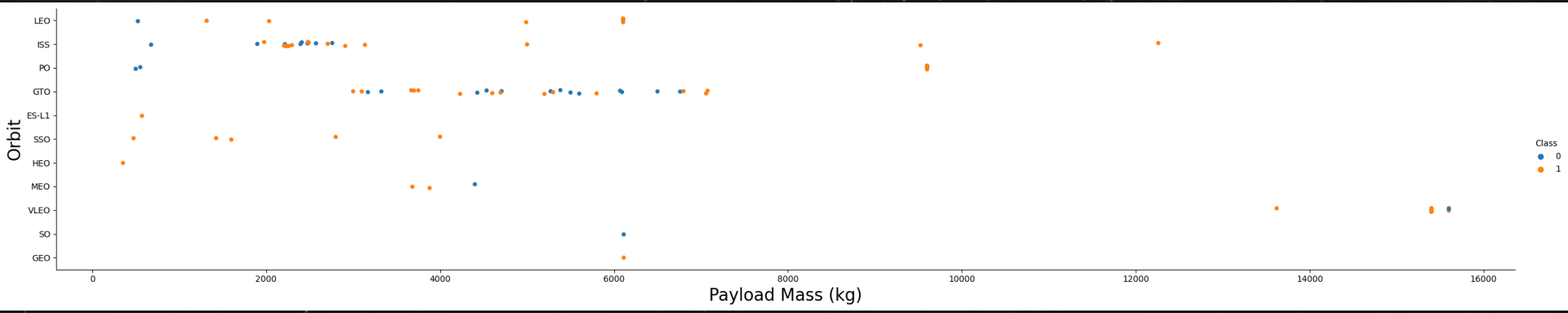
Blue indicates successful launch and orange indicates unsuccessful launch.

- Launch Orbit preferences changed over Flight Number.
- Launch Outcome seems to correlate with this preference.
- SpaceX started with LEO orbits which saw moderate success LEO and returned to VLEO in recent launches
- SpaceX appears to perform better in lower orbits or Sun synchronous orbits

Payload vs. Orbit Type

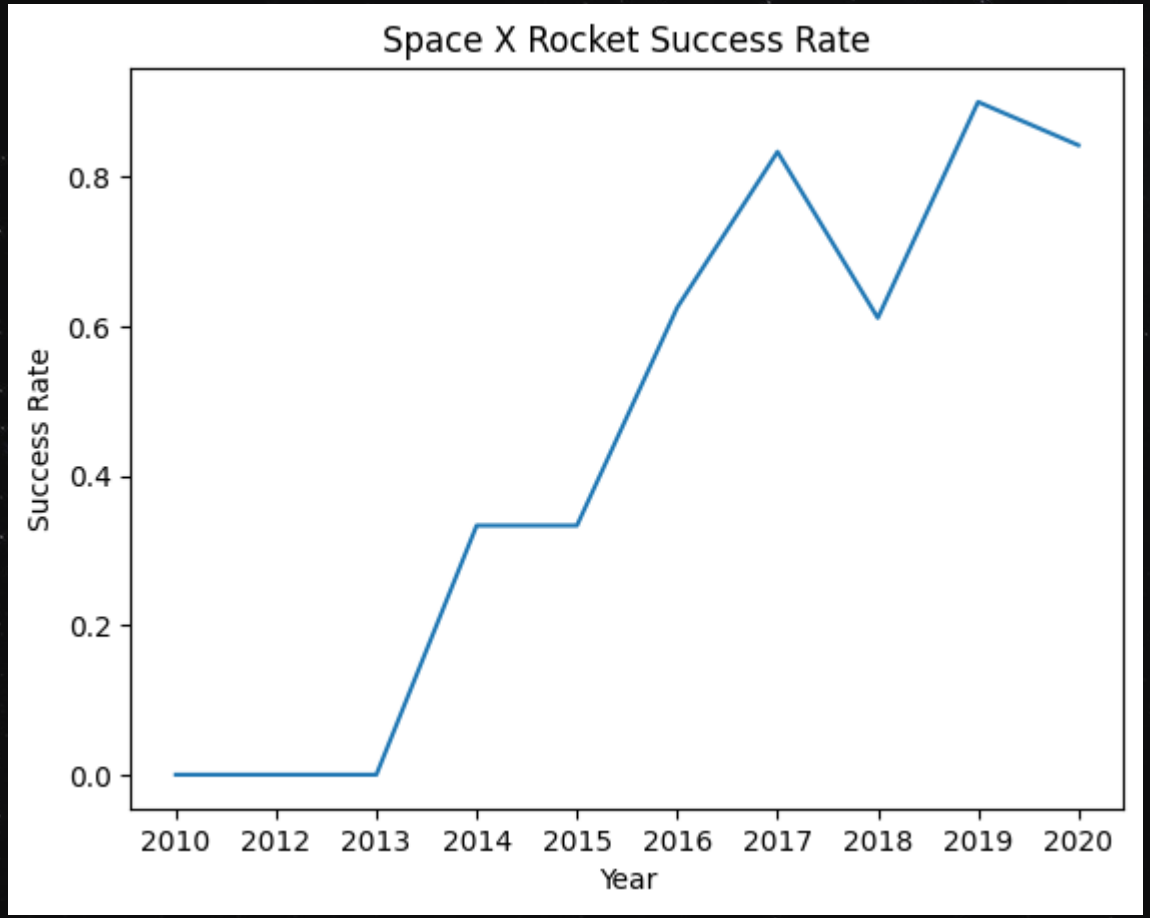
Blue indicates successful launch and orange indicates unsuccessful launch.

- Payload mass seems to correlate with orbit
- LEO and SSO seem to have relatively low payload mass
- The other most successful orbit VLEO only has payload mass values in the higher end of the range



Launch Success Yearly Trend

- Success generally increases over time since 2013 with a slight dip in 2018
- Success in recent years at around 85%



All Launch Site Names

```
%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
None

We can see, there are 4 different launch sites in the dataset and also there are none launch site values

Launch Site Names Begin with 'CCA'

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

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
06/04/2010	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0.0	LEO	SpaceX	Success	Failure (parachute)
12/08/2010	15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight C1, two CubeSats, barrel of Brouere cheese	0.0	LEO (ISS)	NASA (COTS) NRO	Success	Failure (parachute)
22/05/2012	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525.0	LEO (ISS)	NASA (COTS)	Success	No attempt
10/08/2012	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500.0	LEO (ISS)	NASA (CRS)	Success	No attempt
03/01/2013	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677.0	LEO (ISS)	NASA (CRS)	Success	No attempt

In that DF, we can see the first five Launch Sites which starts with 'CCA'

Total Payload Mass

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

```
 * sqlite:///my_data1.db
Done.
 SUM("PAYLOAD_MASS__KG_")
45596.0
```

We found the total Payload Mass with using SUM command in sql It's shown us the total Payload Mass is 45596.0

Average Payload Mass by F9 v1.1

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

```
 * sqlite:///my_data1.db
Done.
AVG("PAYLOAD_MASS__KG_")
2928.4
```

We found the average Payload Mass with using AVG command in sql It's shown us average Payload Mass is 2928.4

First Successful Ground Landing Date

```
%sql SELECT DATE AS FIRST_SUCCESS_GROUND_PAD FROM SPACEXTBL WHERE LANDING_OUTCOME = 'Success (ground pad)'
```

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

FIRST_SUCCESS_GROUND_PAD
22/12/2015
18/07/2016
19/02/2017
05/01/2017
06/03/2017
14/08/2017
09/07/2017
15/12/2017
01/08/2018

As we can see in the list, the first date of succesful ground landing date is '22/12/2015'

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_" > 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

We found the correct booster version which its payload between 4000 and 6000 that land on a drone ship succesfully

Total Number of Successful and Failure Mission Outcomes

```
%sql SELECT "Mission_Outcome", COUNT("Mission_Outcome") FROM SPACEXTBL GROUP BY "Mission_Outcome"
```

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

Mission_Outcome	COUNT("Mission_Outcome")
None	0
Failure (in flight)	1
Success	98
Success	1
Success (payload status unclear)	1

As a result, there is 100 succesful and 1 failure mission outcomes

Boosters Carried Maximum Payload

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

```
 * 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

There are 12 different booster version which have carried maximum payload mass

2015 Launch Records

```
%sql select "Date", substr("January, February, March, April, May, June, July, August, September, October, November, December ", \
substr("Date", 4, 2)*9+1, 9) as Month_Names, "Booster_Version", "Landing_Outcome", "Launch_Site" from \
SPACEXTBL where "Landing_Outcome" = (select "Landing_Outcome" from SPACEXTBL where "Landing_Outcome" like 'Failure (drone ship)') \
and "Launch_Site" = (select "Launch_Site" from SPACEXTBL where substr("Date",7,4) = '2015')
```

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

Date	Month_Names	Booster_Version	Landing_Outcome	Launch_Site
01/10/2015	January,	F9 v1.1 B1012	Failure (drone ship)	CCAFS LC-40
14/04/2015	January,	F9 v1.1 B1015	Failure (drone ship)	CCAFS LC-40
03/04/2016	January,	F9 FT B1020	Failure (drone ship)	CCAFS LC-40
15/06/2016	January,	F9 FT B1024	Failure (drone ship)	CCAFS LC-40

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 Landing Outcomes Between 2010 06 04 and 2017 03 20

```
%sql select "Landing_Outcome", count("Landing_Outcome") as rank from (select "Landing_Outcome" from SPACEXTBL where "Date">'04/06/2010' and "Date"<'20/03/2017') \
group by "Landing_Outcome" \
order by count("Landing_Outcome") desc
```

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

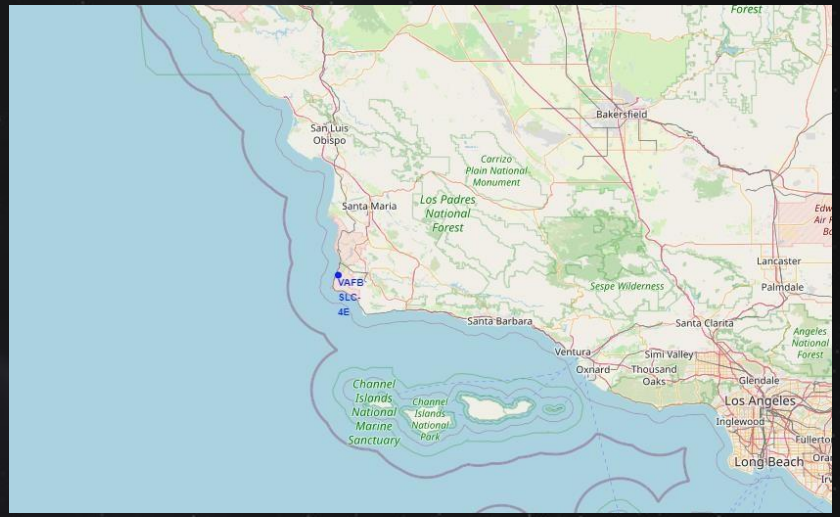
Landing_Outcome	rank
Success	20
No attempt	9
Success (drone ship)	8
Success (ground pad)	7
Failure (drone ship)	3
Failure	3
Failure (parachute)	2
Controlled (ocean)	2
No attempt	1

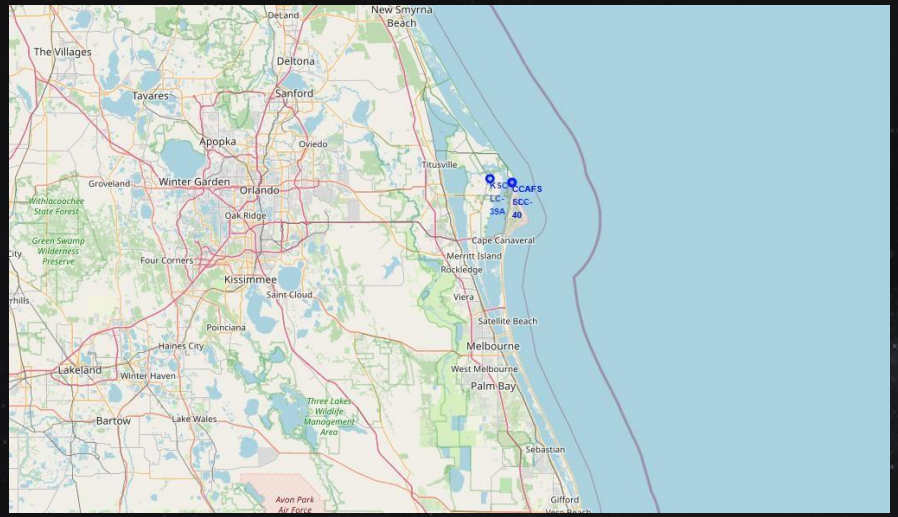
Rank the count of landing outcomes (such as Failure (drone ship) or Success (ground pad)) between the date 2010 06 04 and 2017 03 20, in descending order.

Section 3

Launch sites proximities Analysis

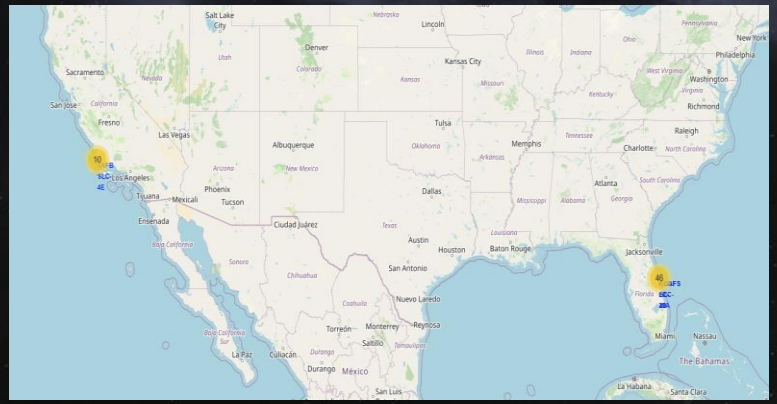
Launch Site Locations





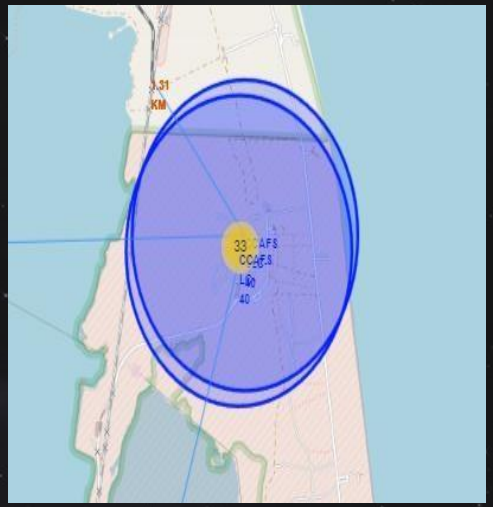
In the map, we can see all launch sites are close to ocean and they are near to ecuador line

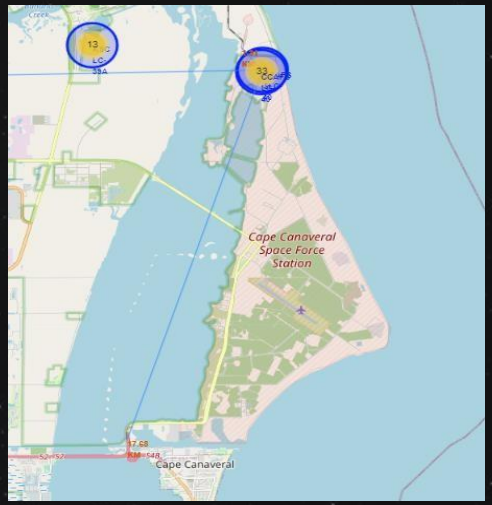
Color Coded Launch Markers



Clusters on Folium map can be clicked on to display each successful landing (green icon) and failed landing (red icon). In this example CCAFS SLC 40 shows 7 successful landings and 19 failed landings.

KEY LOCATION PROXIMITIES

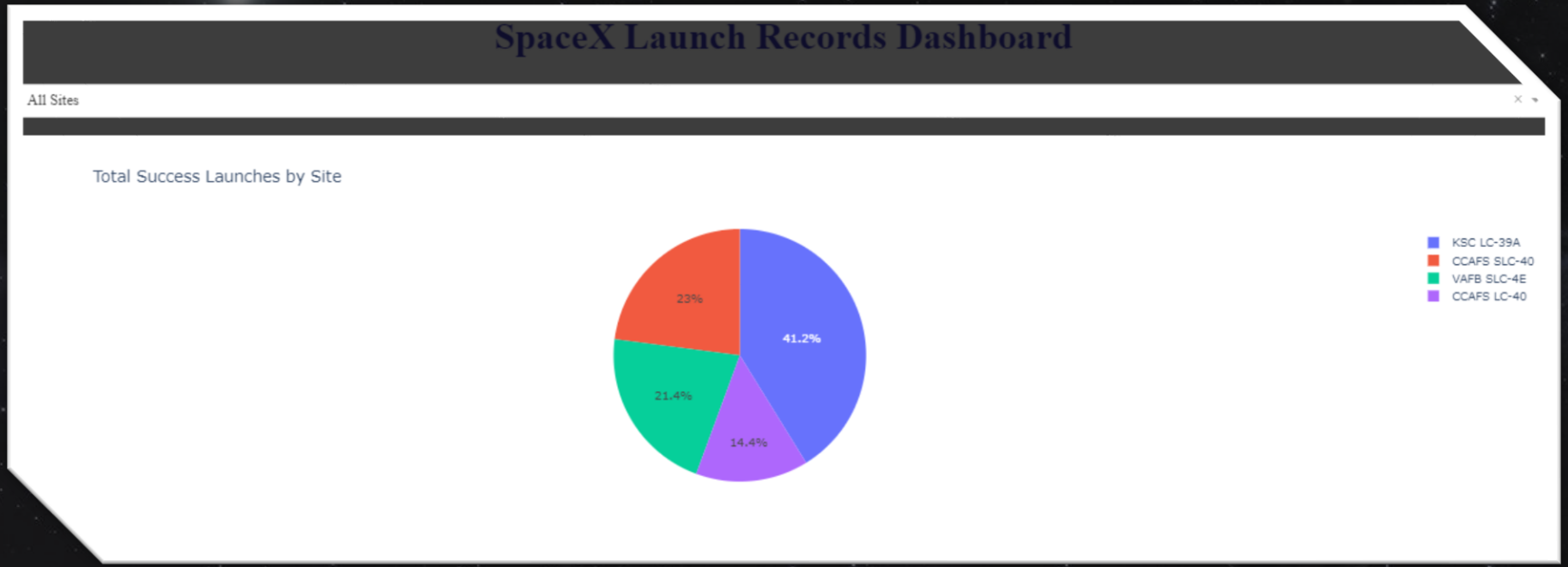




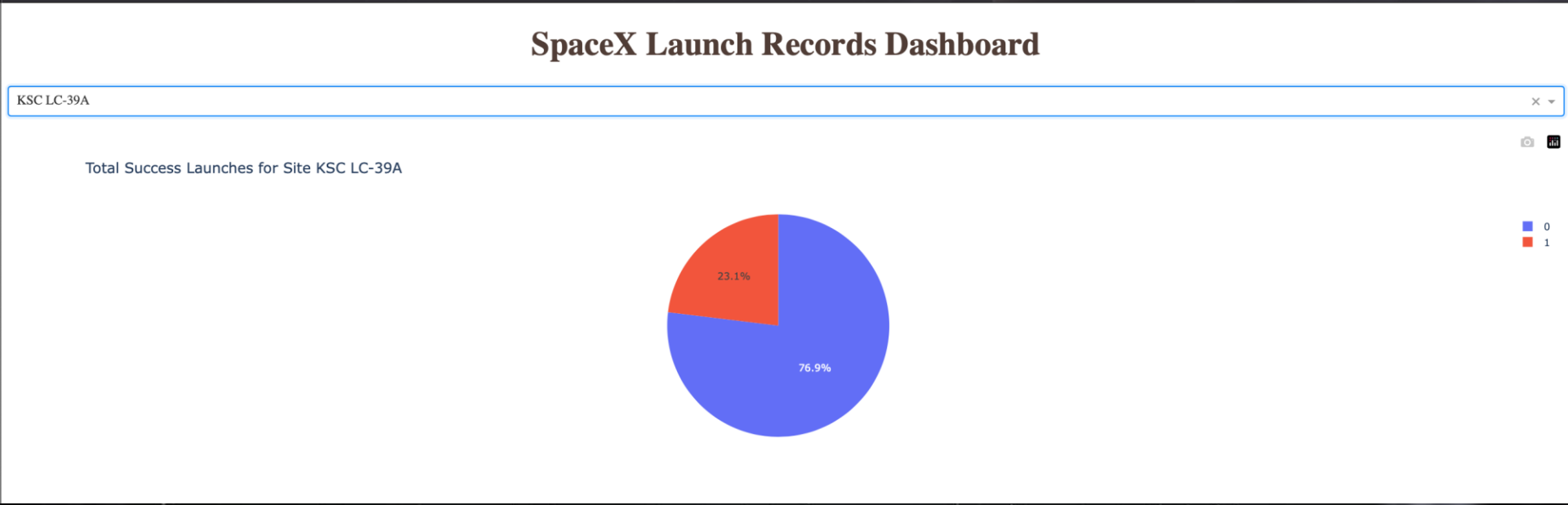
We can see distances of launch site to key locations such as railway, highway and city


SECTION 4
BUILD A DASBOARD WITH PLOTLY DASH

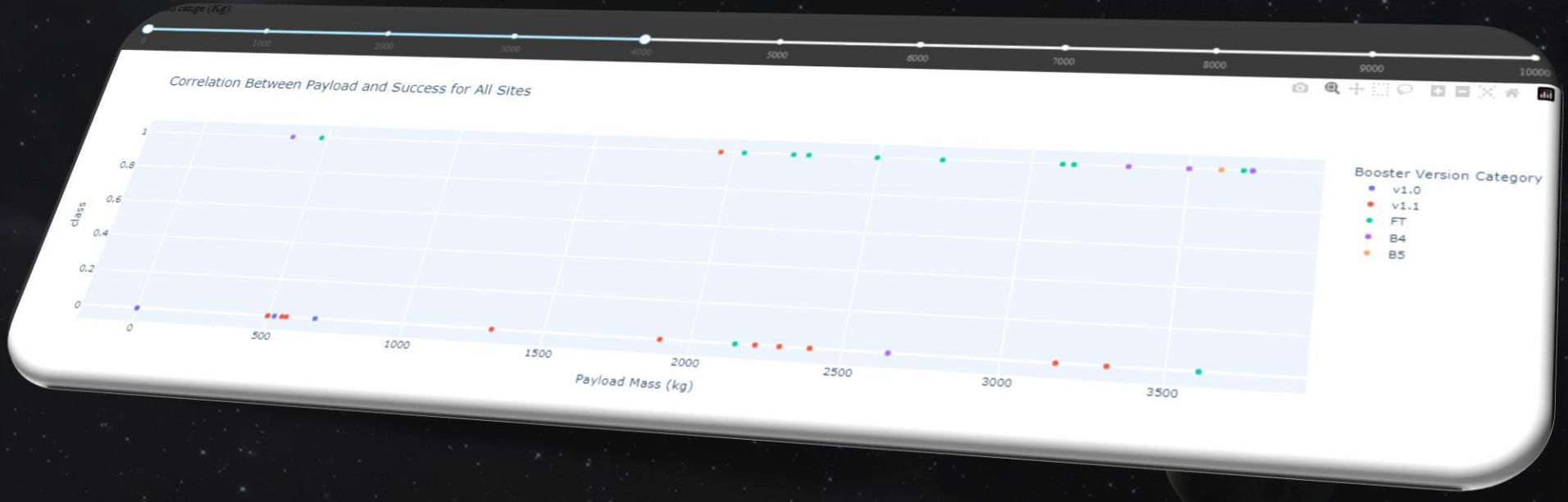
SUCCESSFUL LAUNCH BY ALL SITES



LAUNCH SITE WITH THE BEST SUCCESS RATE



PAYLOAD AND SUCCESS RATE



AS WE CAN SEE UNDER 4000 KG PAYLOAD AND FT BOOSTER IS THE MOST SUCCESSFUL COMBINATION

SECTION 5

PREDICTION ANALYSIS(CLASSIFICATION)

CLASSIFICATION ACCURACY

All models had virtually the same accuracy on the test set at 83.33% accuracy. It should be noted that test size is small at only sample size of 18.

This can cause large variance in accuracy results, such as those in Decision Tree Classifier model in repeated runs.
We likely need more data to determine the best model.

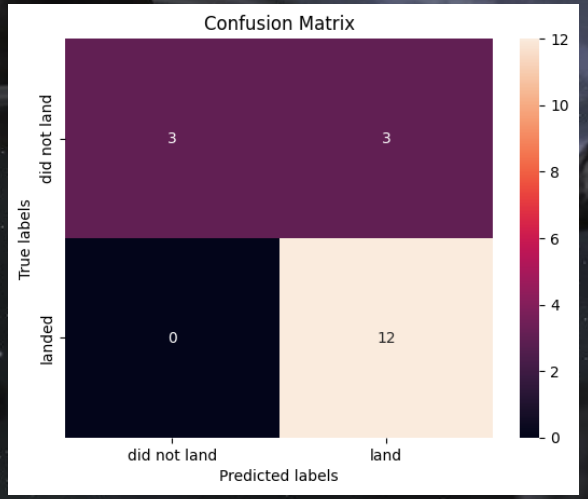
	Accuracy	Jaccard Index	F1-Score	Log Loss
KNN	0.833333	0.8	0.888889	-
Tree	0.833333	0.8	0.888889	-
LR	0.833333	0.8	0.888889	6.007275564852859
SVM	0.833333	0.8	0.888889	-

CONFUSION MATRIX

Since all models performed the same for the test set, the confusion matrix is the same across all models. The models predicted 12 successful landings when the true label was successful landing.

The models predicted 3 unsuccessful landings when the true label was unsuccessful landing.

The models predicted 3 successful landings when the true label was unsuccessful landings (false positives).



CONCLUSIONS

- IF WE LOOK AT THE RESULTS OF ML MODELS WE CAN SEE THEY HAVE SAME SCORES. WE CAN IMPROVE OUR RESULTS AND CHOOSE ONE OF THEM IF WE ADD MORE VALUES TO DATASET OR CHANGE OUR MODELS.
- UNDER PAYLOADS SUCH AS UNDER 6000 KG IS MORE SUCCESSFUL OTHER THAN HIGHER PAYLOADS
- KSC LC 39A HAS BEST SUCCESS RATE
- THE RATE OF SUCCESSFUL LANDINGS ARE IMPROVING OVER THE YEARS
- SSO ORBIT HAS THE BEST SUCCESS RATE

APPENDIX

Improving the accuracy of a machine learning model is a crucial task to ensure reliable predictions. There are several effective strategies that can be employed in this pursuit.

One of the first steps is to acquire more training data. Increasing the size of the dataset allows the model to learn from a broader range of examples and improve its generalization capabilities. Whether it's through collecting more labeled data or using data augmentation techniques, a larger and more diverse training set can greatly enhance the model's accuracy.

Feature engineering plays a significant role in model performance. By carefully selecting or creating relevant features, we can provide the model with better representations of the underlying patterns in the data. Leveraging domain knowledge and employing techniques such as feature scaling, transformation, or dimensionality reduction can greatly enhance the quality of the features used in training.

Optimizing the model's hyperparameters is another effective approach. Adjusting hyperparameters that control the learning process can have a substantial impact on the model's performance.

GitHub Repository URL:

https://github.com/Mesutssmn/IBM SpaceX Data Science Capstone


THANK YOU
Special thanks to IBM for this Journey