Applied Data Science

JOSÉ DIEGO ÁLVAREZ SÁNCHEZ

22/11/2023

OUTLINE

- Executive Summary
- Introduction
- Methodology
- Results
  - Visualization – Charts
  - Dashboard
- Discussion
  - Findings & Implications
- Conclusion
- Appendix

EXECUTIVE SUMMARY

Summary of methodologies

- Obtaining information from the use of the RestAPI.
- Web Scraping
- Data wrangling.
- Exploratory data analysis with SQL.
- Data visualization.
- Visual analytics and dashboards.
- Machine Learning.

Summary of results

- Data Analysis
- Interactive Analytics
- Predictive Analysis

INTRODUCTION

The idea of a rocket launch is too complex, it has many parameters but the data science can do that job easily and fast.

In this report, you can see a whole study for the people who want to get into this world, where the prediction of the cost, the launch place or even if it will work, you will get.

This study gather data from the Falcon 9 of SpaceX, where you can get all the data that you need from an API.

METHODOLOGY

- Jupyter Notebook is the main tool where you can unify your work, which is used with the Python language.
- The RestAPI was used to get the data from Space X.
- Some of the libraries that were used are pandas, seaborn, numpy, sckit-learn, etc.
- Hands on labs was useful to deploy dashboards with the information.

Data Collection(API)

Once we get the information from the RestAPI we filter the data with only Falcon 9. Then we replace the None values with the mean of the PayloadMass.

```
[9]  static_json_url='https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DS0321EN-SkillsNetwork/da
```

```
[10] response.status_code

     200
```

```
[11] # Use json_normalize meethod to convert the json result into a dataframe
     data = pd.json_normalize(response.json())
```

```
#Filtered data
data_falcon9 = launch[launch['BoosterVersion']!='Falcon 1']
data_falcon9.shape

(90, 17)
```

```
[27] # Calculate the mean value of PayloadMass column
     PayloadMassMean = data_falcon9['PayloadMass'].mean()

     # Replace the np.nan values with its mean value
     data_falcon9.loc[:,'PayloadMass']=data_falcon9['PayloadMass'].replace(np.nan,PayloadMassMean)
```

Data Collection(Web scraping)

We use web scrapping to get information from the Falcon 9 by html with BeautifulSoup.

At the end we convert it to a DataFrame.

First, let's perform an HTTP GET method to request the Falcon9 Launch HTML page, as an HTTP response.

```
[6] # requests.get() method with the provided static_url
    # assign the response to a object
    response = requests.get(static_url)
```

`BeautifulSoup` object from the HTML `response`

```
# Use BeautifulSoup() to create a BeautifulSoup object from a response text content
soup = BeautifulSoup(response.text,'html.parser')
```

```
[9] html_tables = soup.find_all('table')
```

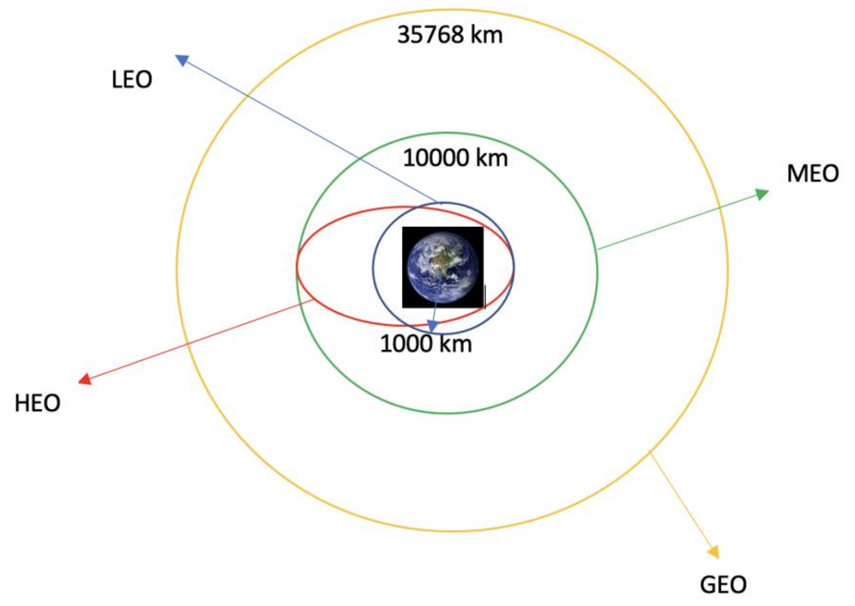
Starting from the third table is our target table contains the actual launch records.

```
first_launch_table = html_tables[2]
print(first_launch_table)
```

Data Wrangling

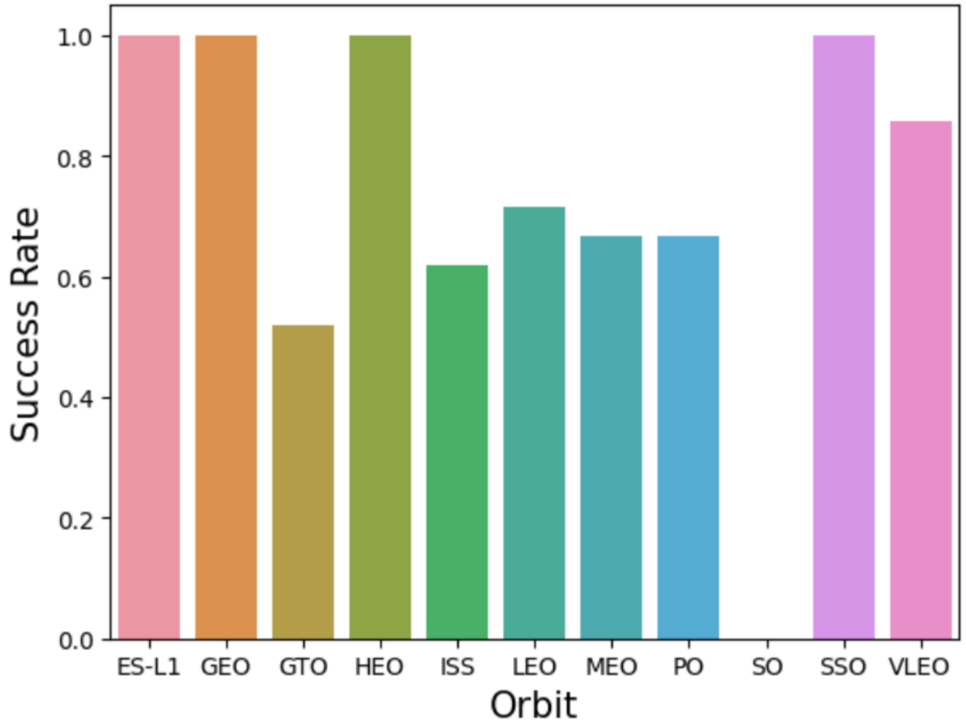
In this step we try to filter and manage the information in base of how we want to find out.

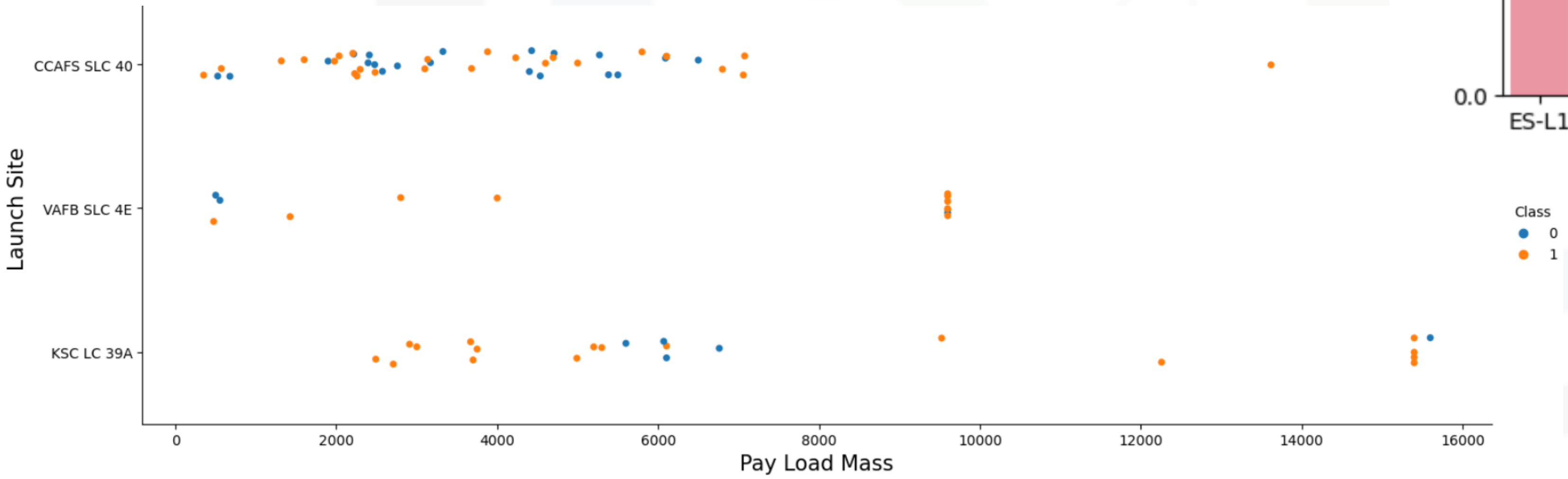
Here we calculated the number of launches at each site and the number of ocurrence of each orbits.



Data Visualization

Here we can visualize some relations in infromation like for example the Launch Site and Pay Load Mass or the relation between Success Rate and Orbit.





IBM Developer

SKILLS NETWORK

SQL

We loaded the SpaceX dataset into a PostgreSQL database without leaving the jupyter notebook.

We applied EDA with SQL to get insight from the data. We wrote queries to find out for instance:

- The names of unique launch sites in the space misión.
- The total payload mass carried by boosters launched by NASA (CRS)
- The average payload mass carried by booster versión F9 v1.1
- The total number of successful and failure misión outcomes
- The failed landing outcomes in drone ship, their booster versión and launch site names.

Interactive Map with Folium

We marked all launch sites, and addd map objects such as markers, circles, lines to mark the success or failure of launches for each site in the folium map.

We assigned the feature launch outcomes depending of failure or success with the class, where it can be a 1 or 0.

Later we calculated distaces between some launches and some interest places like cities, railways and highways.

Dashboard

We built an interactive dashboard with Plotly Dash

Here we plotted pie charts showing the total launches by certain sites.

We also plotted scatte graph showing the relationship with Outcomes and Pay Load Mass (kg) for the different booster versión.

Predictive Analysis (Classification)

We loaded the data using numpy and pandas, transformated the data and split our data into training and testing.

We built different machine learning models and got the best hyperparameters using GridSearchCV.

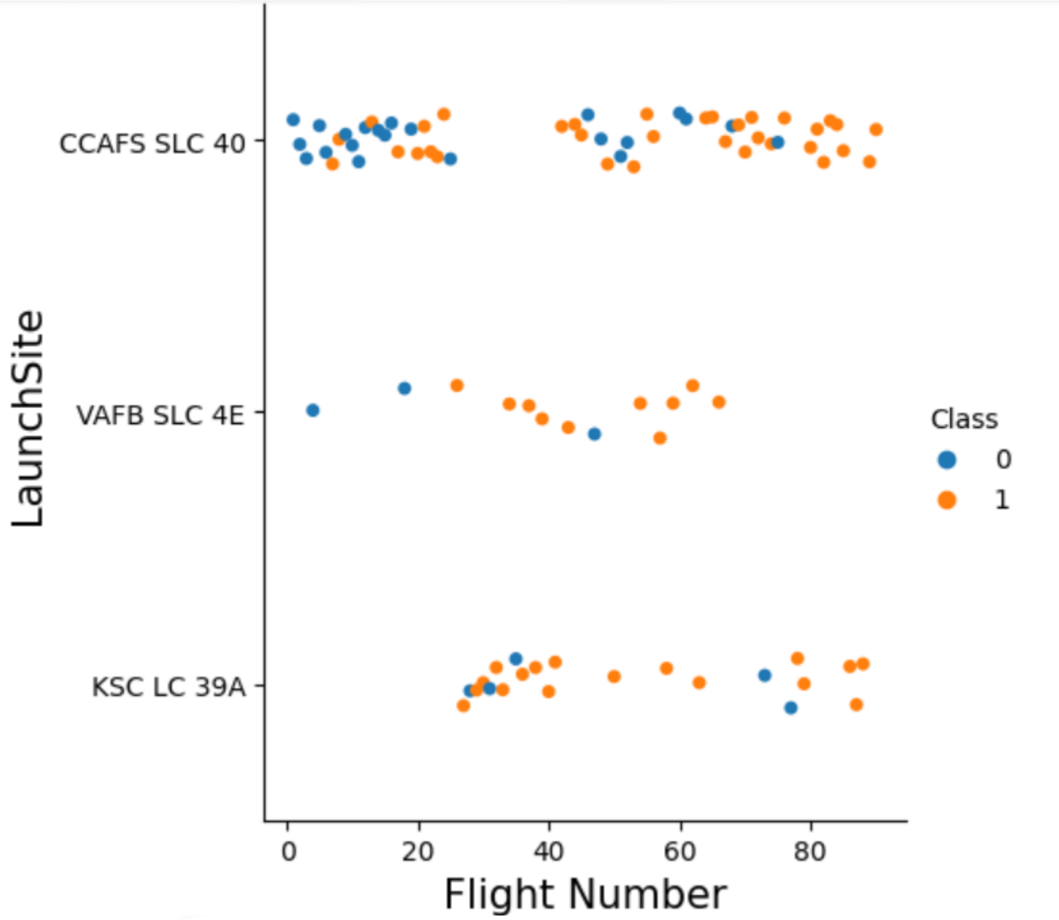
We found the best performing classification using accuracy as the metric for our model.

RESULTS

- Exploratory data análisis results
- Interactive analytics demo in sreenshots
- Predictive análisis results

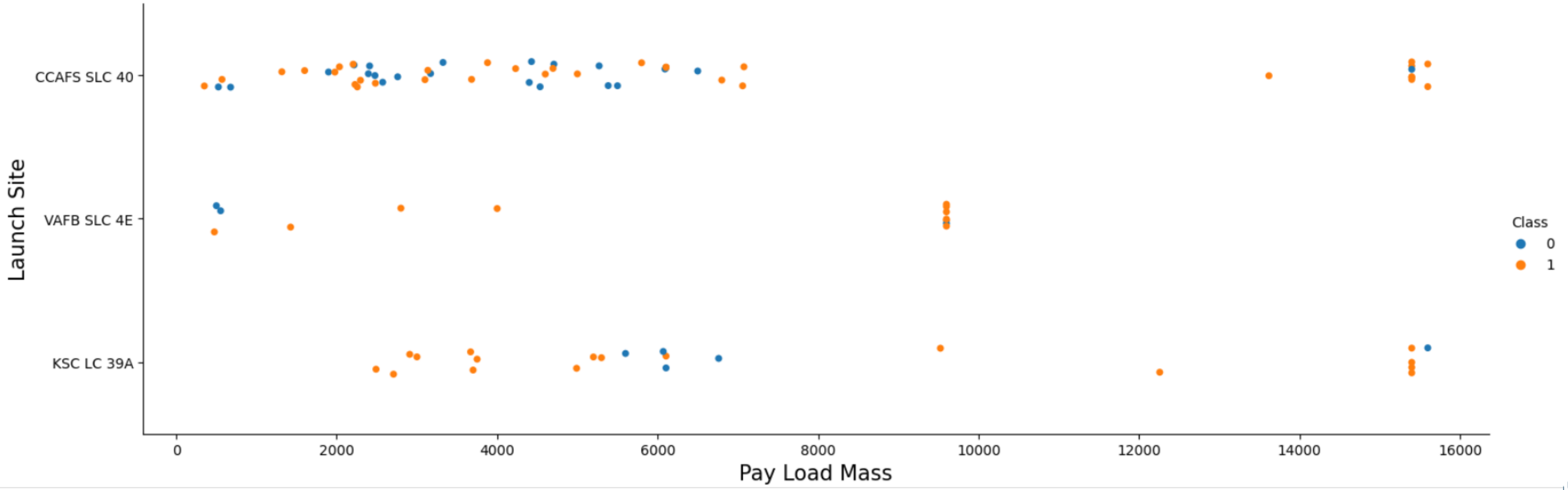
Flight Number vs Launch Site

- Here we can see the relation between flight number and launch when is in VAFB SLC 4E and KSC LC 39ª where the more flight number the more probability to be a 1.

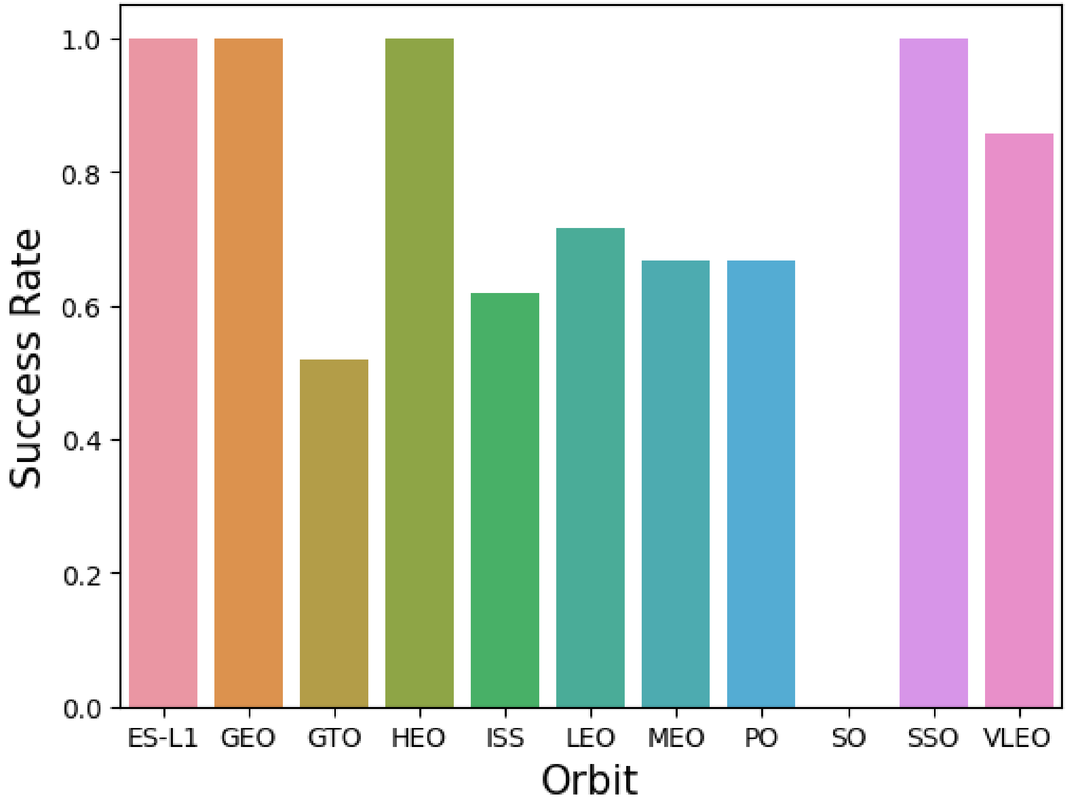


Launch Site vs Paid Load Mass

- Here we can see a relation with VAFB SLC 4E and KSC LC 39 where the greater the Pay Load Mass the higher success rate



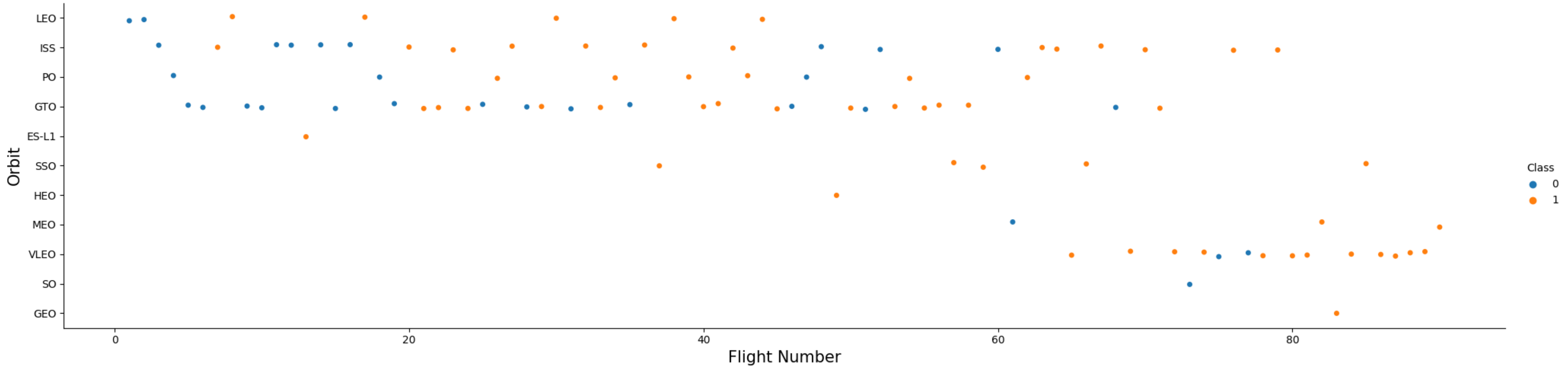
Success Rate vs Orbit



- In this graphic we can see the orbits that have more success rate like ES-L1, GEO, HEO, SSO and VLEO.

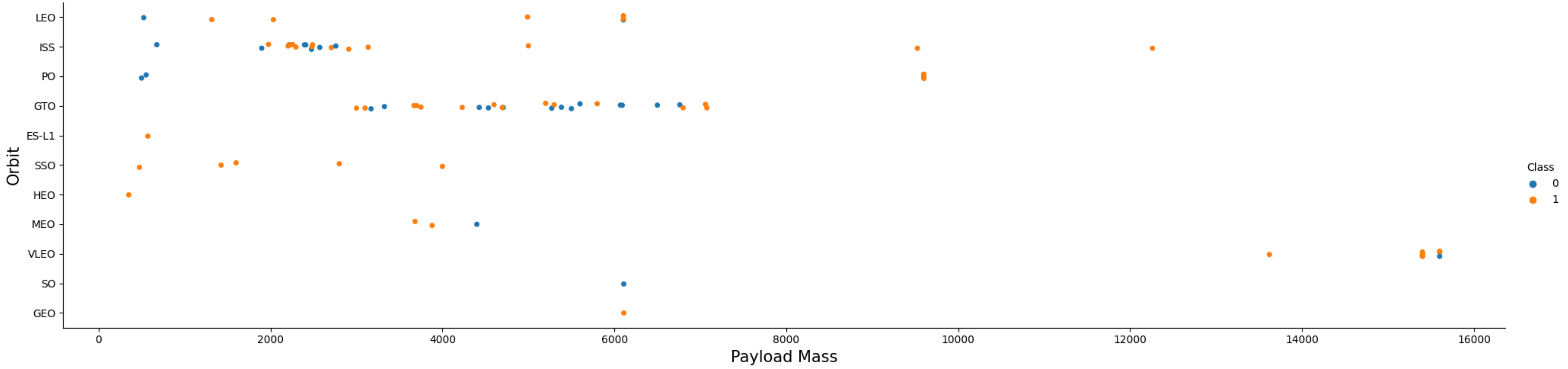
Orbit vs Flight Number

- There is a relation with the orbit and flight number were we can see a high success rate in orbits like VLEO, SSO and LEO.

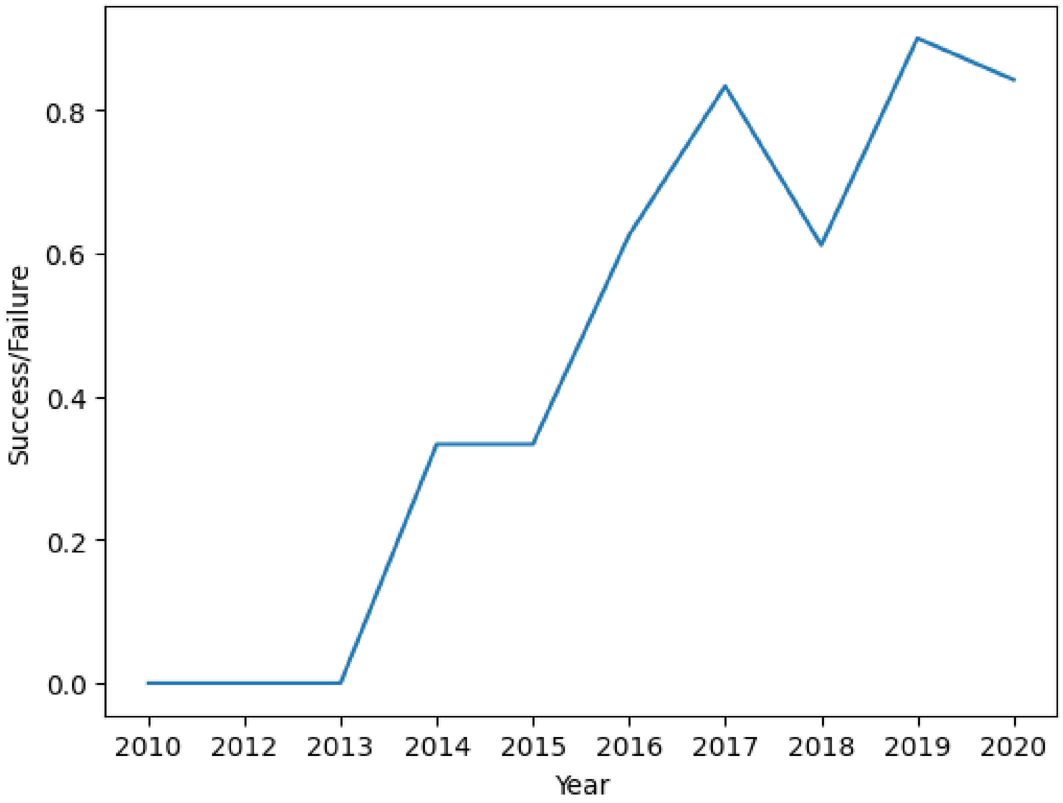


Orbit vs Pay Load Mass

- Then we have the relation Orbit/Pay Load Mass where we can see a high success rate in SSO, LEO and VLEO.



Success/Failure vs Year



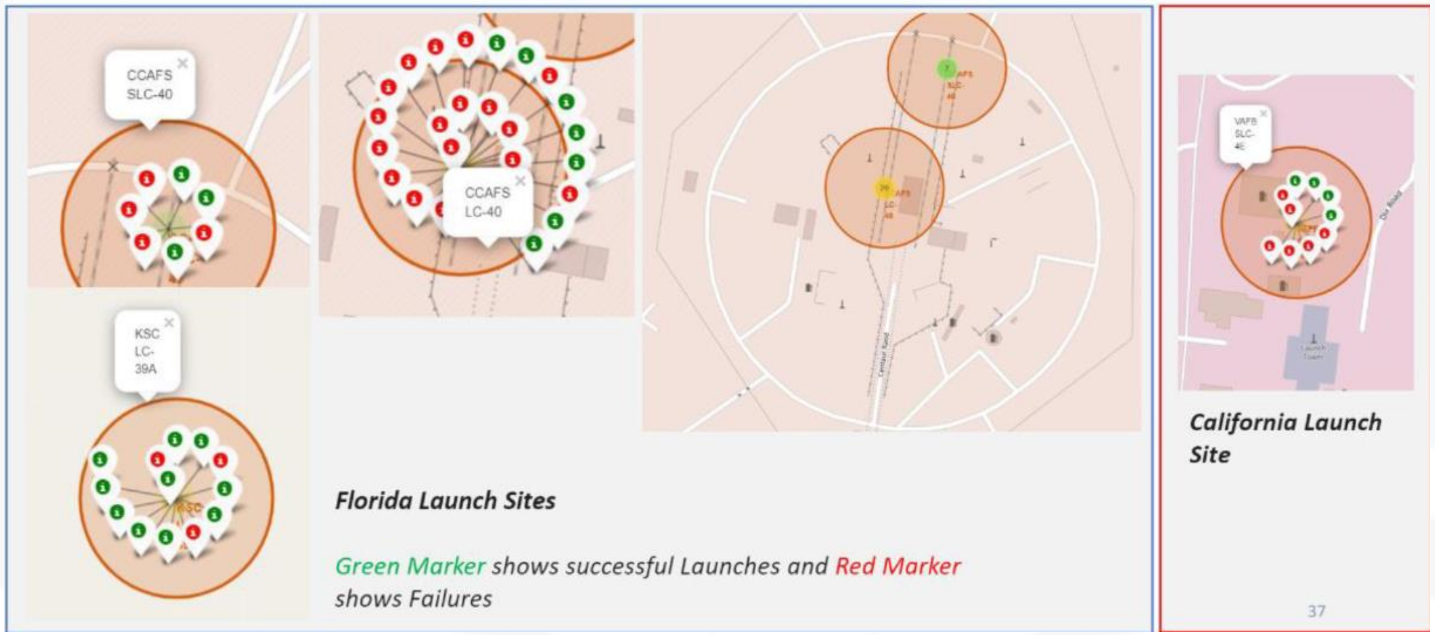
- Then we can see how the success rate increased in almost all the years.

SQL Analysis

- The steps in SQL are, get the unique launch sites
- Get the total payload mass carried by boosters launched by Nasa
- Display average payload mass carried by F9 v1.1
- See boosters wich have success and a pay load mass between 4000 and 6000
- Number of success and failures
- Names of booster versions with máximum payload mass
- At the end we rank the count of landing outcomes between 2010 and 2017

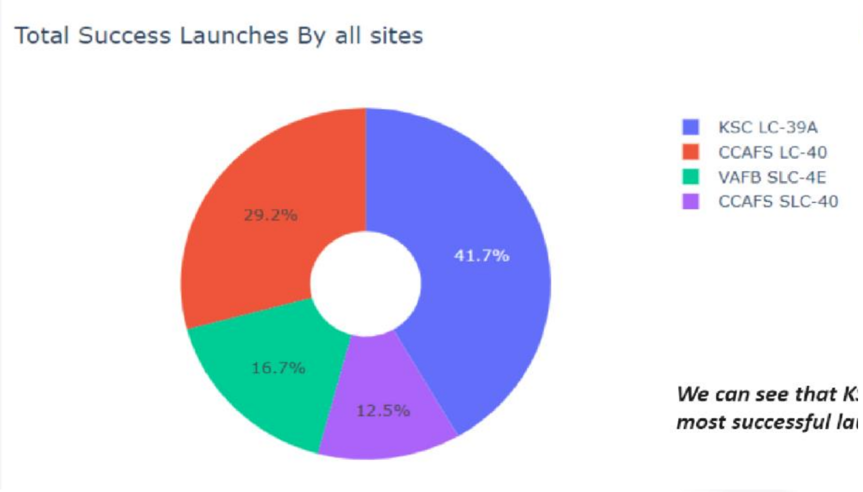
Folium

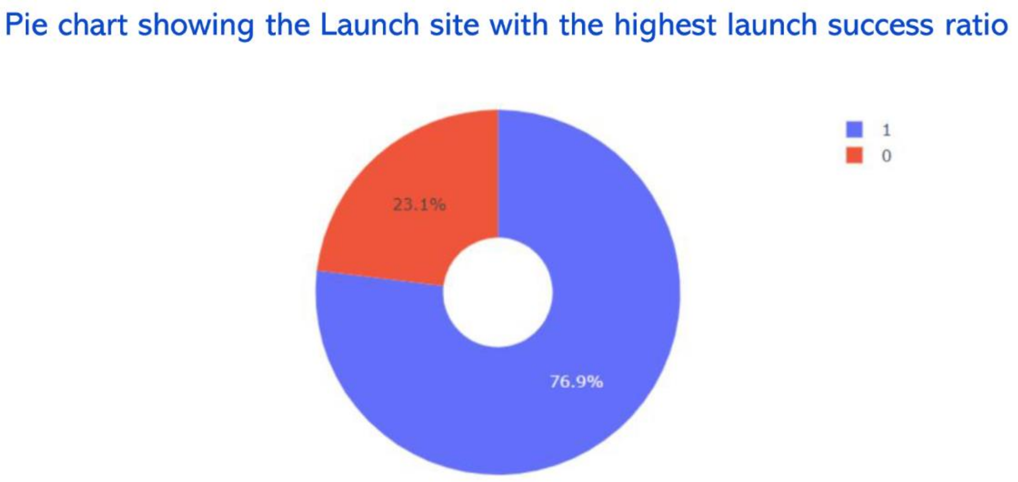
- We use markers to see te diference in the locations between success and failure.



Dashboard

- We use pie chart to get a better understanding in the information.



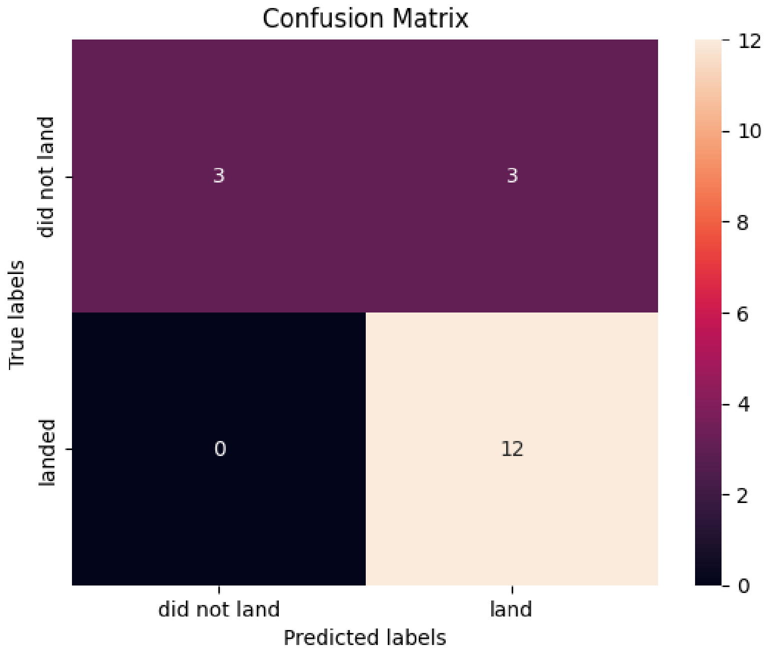


KSC LC-39A achieved a 76.9% success rate while getting a 23.1% failure rate

Machine Learning

- We could see with the code that the best method is Tree with an accuracy of 0.860 and with a confusión matrix like the next one:

```
accuracy = pd.DataFrame(accuracy)
accuracy = accuracy.rename(columns={0: 'Train Accuracy', 1: 'Test Accuracy'})
best = accuracy['Train Accuracy'].idxmax()
best_accuracy = accuracy['Train Accuracy'].max()
Name = pd.DataFrame({'Name Model':['Logistic Regression','SVM','Tree','KNN']})
accuracy = pd.concat([accuracy, Name], axis=1)
best_model_name = accuracy.loc[best, 'Name Model']
print(f"The model {best_model_name} has the best accuracy with {best_accuracy}")
accuracy
```



CONCLUSION

- The Larger the flight amount at a launch site, the greater the success rate at a launch site
- Launch success rate started to increase in 2013 till 2020
- Orbits ES-L1, GEO, HEO, SSO, VLEO had the most success
- KSC LC-39A had the most successful launch of any sites
- The decision tree classifier is the best method