Space X Falcon 9 First Stage Landing Prediction

Riccardo Saruis
16-10-2021

Outline

- Executive Summary
- Introduction
- Methodology
- Results
- Conclusion

Executive Summary

Summary of methodologies

- Data collection
- Data wrangling
- EDA with data visualization
- Building an interactive map with Folium
- Building a Dashboard with Plotly Dash
- Predictive analysis (Classification)

Summary of all results

- Results visualization
- Conclusions

Introduction

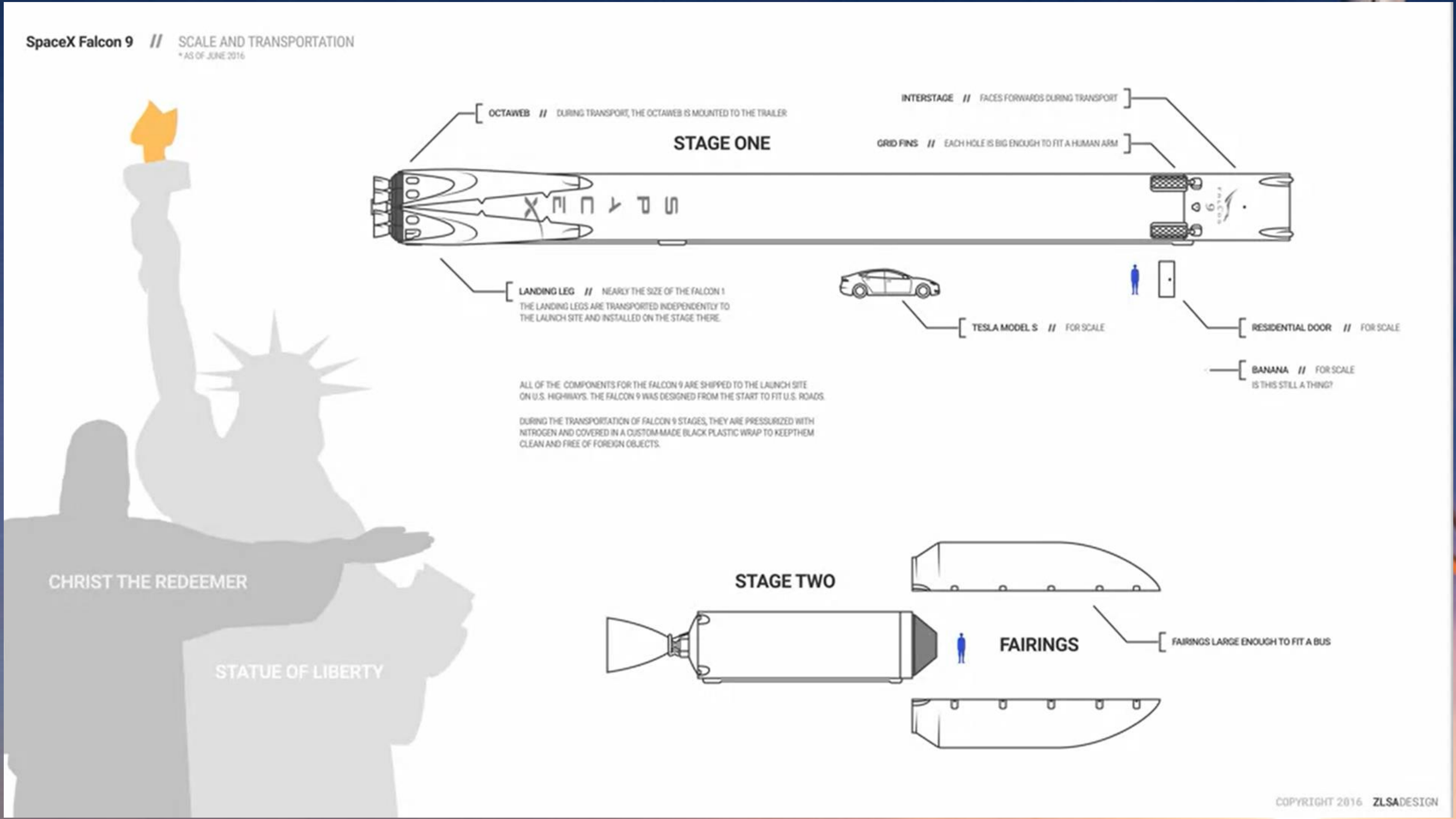
Project background and context

- We will predict if the Falcon 9 first stage will land successfully. SpaceX advertises Falcon 9 rocket launches on its website, with a cost of 62 million dollars; other providers cost upward of 165 million dollars each, much of the savings is because SpaceX can reuse the first stage. Therefore, if we can determine if the first stage will land, we can determine the cost of a launch. This information can be used if an alternate company wants to bid against SpaceX for a rocket launch.

Problems you want to find answers

- What influences if the rocket will land successfully?
- The effect each relationship with certain rocket variables will impact in determining the success rate of a successful landing.
- What conditions does SpaceX have to achieve to get the best results and ensure the best rocket success landing rate.

Rocket components



First stage landing

Section 1

Methodology

Winning Space Race with Data Science

Methodology

Executive Summary

Data collection methodology:

- SpaceX Rest API
- Web Scraping from: Wikipedia

Perform data wrangling (Transforming data for Machine Learning)

- One Hot Encoding data fields for Machine Learning and dropping irrelevant columns

Perform exploratory data analysis (EDA) using visualization and SQL

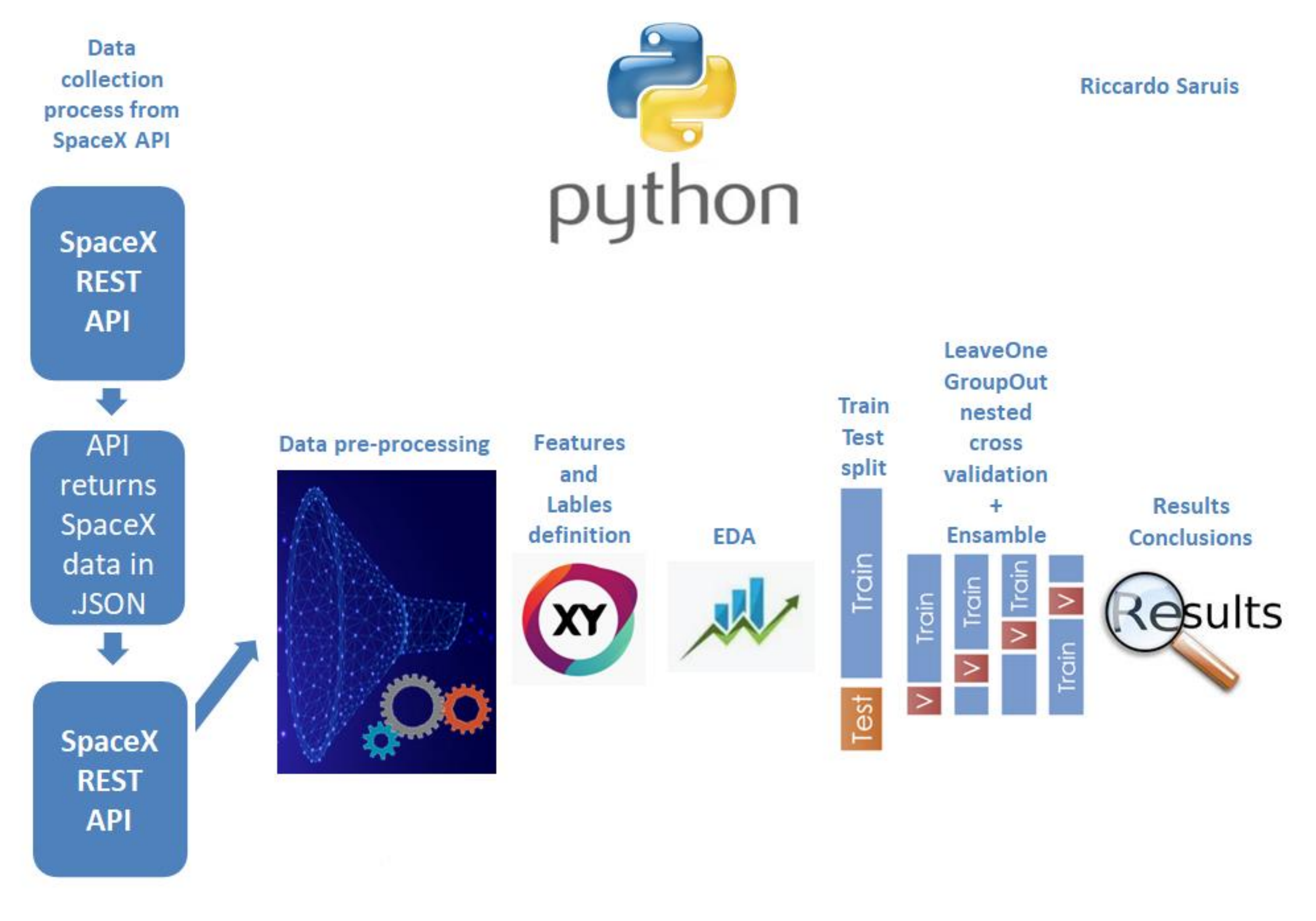
- Plotting : Scatter Graphs, Bar Graphs to show relationships between variables to show patterns of data.

Perform interactive visual analytics using Folium and Plotly Dash

Perform predictive analysis using classification models

- How to build, tune, evaluate classification models

Methodology



Data Collection

The following datasets was collected by:

- We worked with SpaceX launch data that is gathered from the SpaceX REST API.
- This API will give us data about launches, including information about the rocket used,
- payload delivered, launch specifications, landing specifications, and landing outcome.
- Our goal is to use this data to predict whether SpaceX will attempt to land a rocket or not.
- The SpaceX REST API endpoints, or URL, starts with api.spacexdata.com/v4/.
- Another popular data source for obtaining Falcon 9 Launch data is web scraping Wikipedia using BeautifulSoup.

Data collection process from SpaceX API

SpaceX REST API → API returns SpaceX data in .JSON → **SpaceX REST API**

Web Scraping from Wikipedia page

Get HTML Response from Wikipedia → Extracting data using beautiful soup → Normalize data into flat data file such as .csv

Data Collection SpaceX API

1. Getting response from API

```python
static_json_url='https://cf-courses-data.s3.us.cloud-
```

```python
response = requests.get(static_json_url)
```

2. Converting Response to a .json file

```python
data = pd.json_normalize(response.json())
```

3. Apply custom functions to clean data and assign list to dictionary then dataframe

```python
getBoosterVersion(data)
getLaunchSite(data)
getPayloadData(data)
getCoreData(data)
```

```python
launch_dict = {'FlightNumber': list(data['flight_number']),
'Date': list(data['date']),
'BoosterVersion':BoosterVersion,
'PayloadMass':PayloadMass,
'Orbit':Orbit,
'LaunchSite':LaunchSite,
'Outcome':Outcome,
'Flights':Flights,
'GridFins':GridFins,
'Reused':Reused,
'Legs':Legs,
'LandingPad':LandingPad,
'Block':Block,
'ReusedCount':ReusedCount,
'Serial':Serial,
'Longitude': Longitude,
'Latitude': Latitude}
```

```python
df = pd.DataFrame.from_dict(launch_dict)
```

4. Filter dataframe and export to flat file (.csv)

```python
data_falcon9 = df[df['BoosterVersion']=='Falcon 9']
```

```python
data_falcon9.to_csv('dataset_part_1.csv', index=False)
```

GitHub

Click GitHub Logo

Data Collection Web Scraping

1 .Getting Response from HTML

```
static_url = "https://en.wikipedia.org/w/
```

```
page = requests.get(static_url).text
```

2. Creating BeautifulSoup Object

```
soup = BeautifulSoup(page,"html.parser")
```

3. Finding tables and getting column names

```
html_tables = soup.find_all("table")
```

```
column_names = []
# Apply find_all() function with `th` element on
columns = first_launch_table.find_all("th")

print(f'Raw column names:\n')

# Iterate each th element and apply the provided
for th in columns:
    name = extract_column_from_header(th)
    #print(name)

    # Append the Non-empty column name (`if name
    if str(name)!='None' and len(str(name))>0:
        column_names.append(name)
```

4. Creation of dictionary and appending data to keys

```
launch_dict= dict.fromkeys(column_names)

# Remove an irrelvant column
del launch_dict['Date and time ( )']

# Let's initial the launch_dict with each va
launch_dict['Flight No.'] = []
launch_dict['Launch site'] = []
launch_dict['Payload'] = []
launch_dict['Payload mass'] = []
launch_dict['Orbit'] = []
launch_dict['Customer'] = []
launch_dict['Launch outcome'] = []
# Added some new columns
launch_dict['Version Booster']=[]
launch_dict['Booster landing']=[]
launch_dict['Date']=[]
launch_dict['Time']=[]
```

```
extracted_row = 0
#Extract each table
for table_number,table in enumerate(soup.find_all('table',
   # get table row
    for rows in table.find_all("tr"):
        #check to see if first table heading is as number
        if rows.th:
            if rows.th.string:
                flight_number=rows.th.string.strip()
                flag=flight_number.isdigit()
        else:
```

5. Converting dictionary to dataframe and save it in .csv

```
df = pd.DataFrame.from_dict(launch_dict,orient='index').T
```

```
df.to_csv('spacex_web_scraped.csv', index=False)
```

Click GitHub Logo

Data Wrangling

- In the data set, there are several different cases where the booster did not land successfully. Sometimes a landing was attempted but failed due to an accident; for example,True Oceanmeans the mission outcome was successfully landed to a specific region of the ocean whileFalse Oceanmeans the mission outcome was unsuccessfully landed to a specific region of the ocean.True RTLSmeans the mission outcome was successfully landed to a ground padFalse RTLSmeans the mission outcome was unsuccessfully landed to a ground pad. True ASDSmeans the mission outcome was successfully landed on a drone shipFalse ASDSmeans the mission outcome was unsuccessfully landed on a drone ship.
- We mainly convert those outcomes into Training Labels with1means the booster successfully landed0means it was unsuccessful.

Process

Perform Exploratory Data Analysis EDA on dataset

Calculate the number of launches at each site

Calculate the number and occurrence of each orbit

Calculate the number and occurrence of mission outcome per orbit type

Export dataset as .CSV

Create a landing outcome label from Outcome column

Work out success rate for every landing in dataset

Each launch aims to an dedicated orbit, and here are some common orbit types:

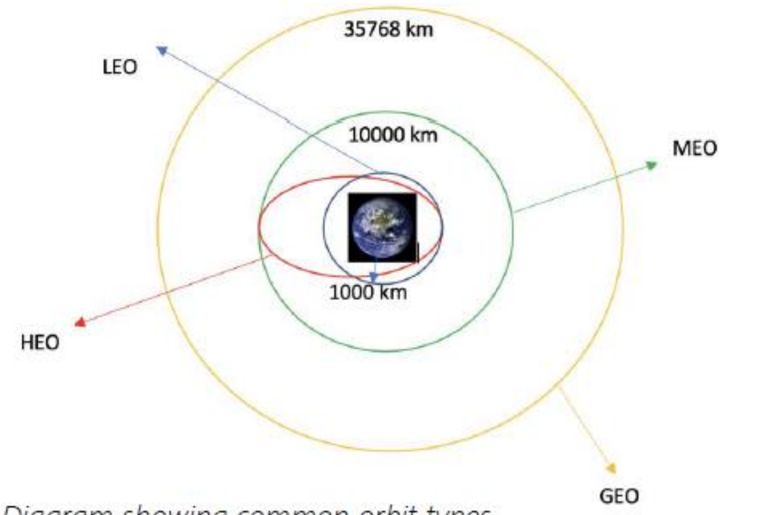


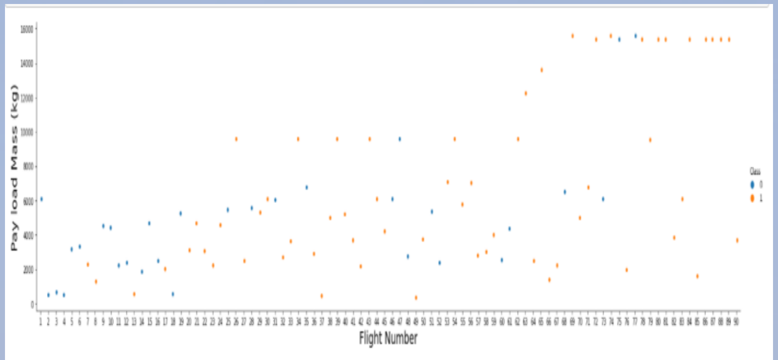
Diagram showing common orbit types SpaceX uses

EDA with Data Visualization

Scatter Graphs being drawn:

- Flight Number VS. Payload Mass
- Flight Number VS. Launch Site
- Payload VS. Launch Site
- Orbit VS. Flight Number
- Payload VS. Orbit Type
- Orbit VS. Payload Mass

Scatter plots showhow much one variable is affected by another. The relationship between two variables is called their correlation .Scatter plotsusually consist of a large body of data.

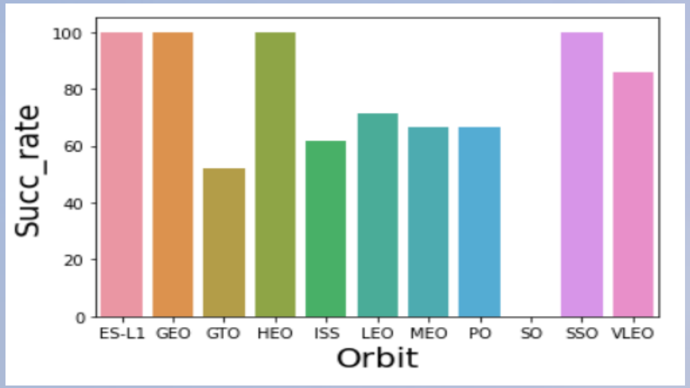


Bar Graph being drawn:

- Mean VS. Orbit

A bar diagram makes it easy to compare sets of data between different groups at a glance.

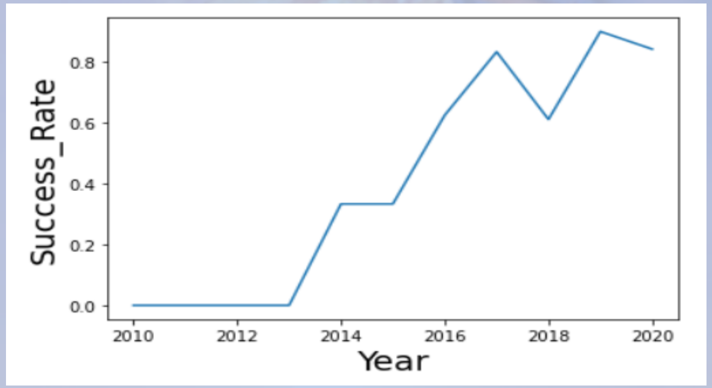
The graph represents categories on one axis and a discrete value in the other. The goal is to show the relationship between the two axes. Bar charts can also show big changes in data over time.



Line Graph being drawn:

- Success Rate VS. Year

Line graphs are useful in that they show data variables and trends very clearly and can help to make predictions about the results of data not yet recorded



Build an Interactive Map with Folium

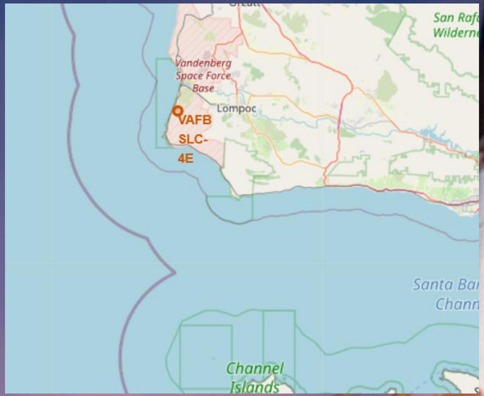
To visualize the Launch Data into an interactive map. We took the Latitude and Longitude Coordinates at each launch site and added a Circle Marker around each launch site with a label of the name of the launch site.

We assigned the dataframe launch_outcomes(failures, successes) to classes 0 and 1with Green and Red markers on the map in a MarkerCluster()

Using Haversine's formula we calculated the distance from the Launch Site to various landmarks to find various trends about what is around the Launch Site to measure patterns. Linesare drawn on the map to measure distance to landmarks

Example of some trends in which the Launch Site is situated in.

- Are launch sites in close proximity to railways? No
- Are launch sites in close proximity to highways? No
- Are launch sites in close proximity to coastline? Yes
- Do launch sites keep certain distance away from cities? Yes



Build a Dashboard with Plotly Dash

Used Python Anywhere to host the website live 24/7 so your can play around with the data and view the data

The dashboard is built with Flask and Dash web framework.

Graphs

- **Pie Chart showing the total launches by a certain site/all sites**
- **display relative proportions of multiple classes of data.**
- **size of the circle can be made proportional to the total quantity it represents.**

Scatter Graph showing the relationship with Outcome and Payload Mass (Kg) for the different Booster Versions

- **It shows the relationship between two variables.**
- **It is the best method to show you a non linear pattern.**
- **The range of data flow, i.e. maximum and minimum value, can be determined.**
- **Observation and reading are straightforward.**

Predictive Analysis (Classification)

BUILDING MODEL

- Load our dataset into NumPy and Pandas
- Transform Data
- Split our data into training and test data sets
- Check how many test samples we have
- Decide which type of machine learning algorithms we want to use
- Set our parameters and algorithms to GridSearchCV
- Fit our datasets into the GridSearchCV object and train our dataset.

EVALUATING MODEL

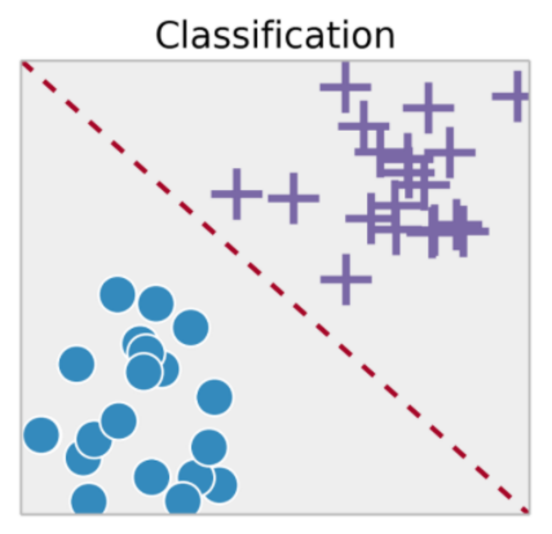
- Check accuracy for each model
- Get tuned hyperparameters for each type of algorithms
- Plot Confusion Matrix

IMPROVING MODEL

- Feature Engineering
- Algorithm Tuning

FINDING THE BEST PERFORMING CLASSIFICATION MODEL

- The model with the best accuracy score wins the best performing model
- In the notebook there is a dictionary of algorithms with scores at the bottom of the notebook.



Results

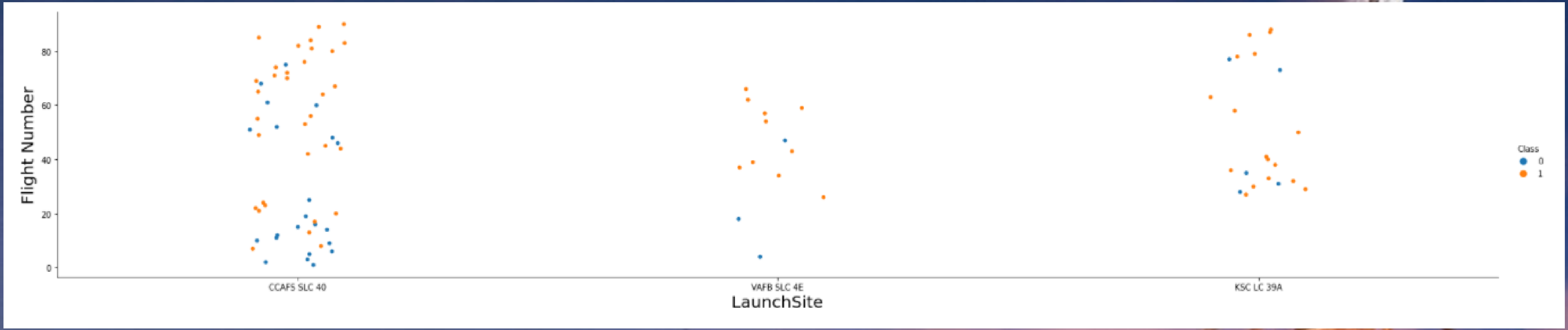
- Exploratory Data Analysis (EDA) results
- Interactive analytics demo in screenshots
- Predictive analysis results

Section 2

Insights drawn from EDA

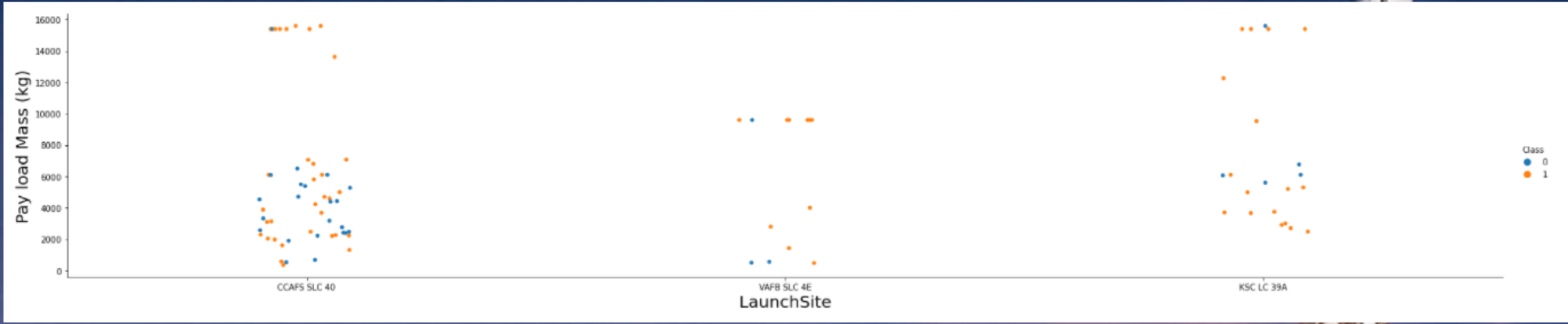
Winning Space Race
with Data Science

Flight Number vs. Launch Site



The more amount of flights at a launch site the greater the success rate at a launch site.

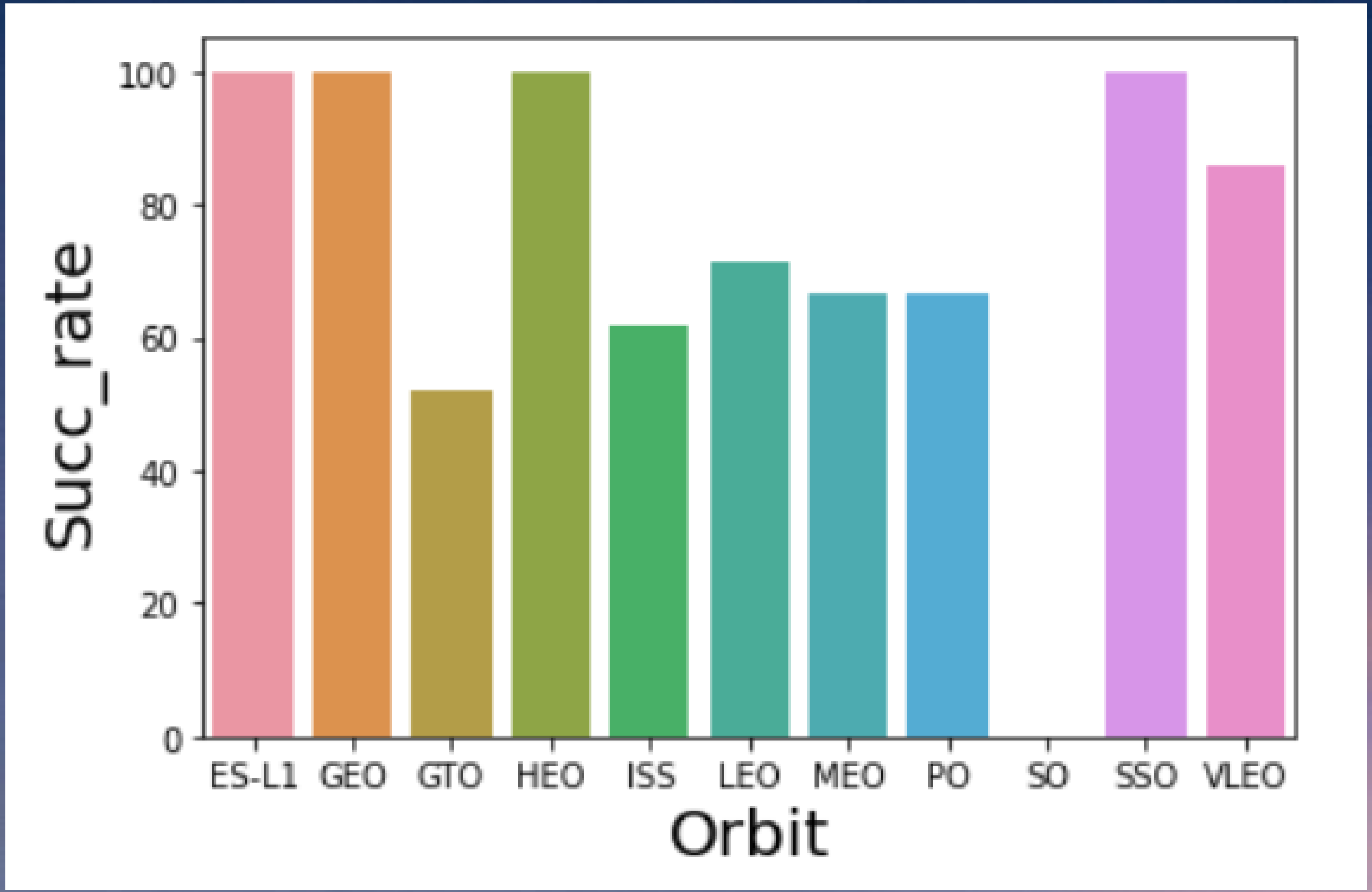
Payload vs. Launch Site



The greater the payload mass for Launch Site CCAFS SLC 40 the higher the success rate for the Rocket.

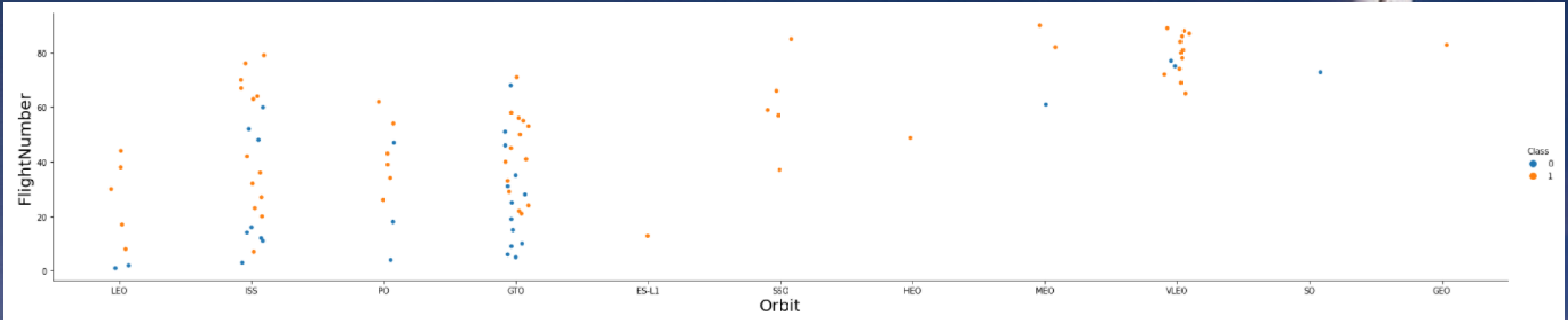
There is not quite a clear pattern to be found using this visualization to make a decision if the Launch Site is dependant on Pay Load Mass for a success launch.

Success Rate vs. Orbit Type



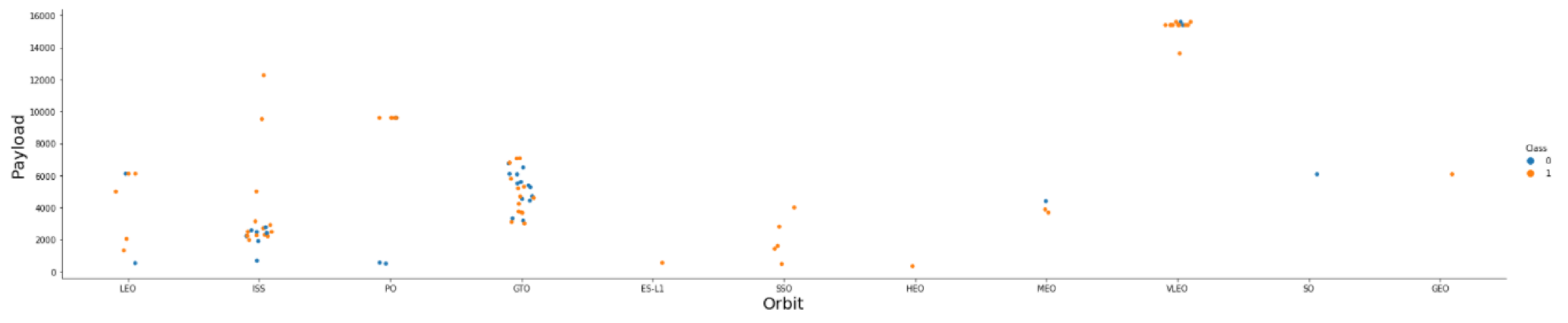
Orbit GEO, HEO, SSO, ES-L1 has the best Success Rate

Flight Number vs. Orbit Type



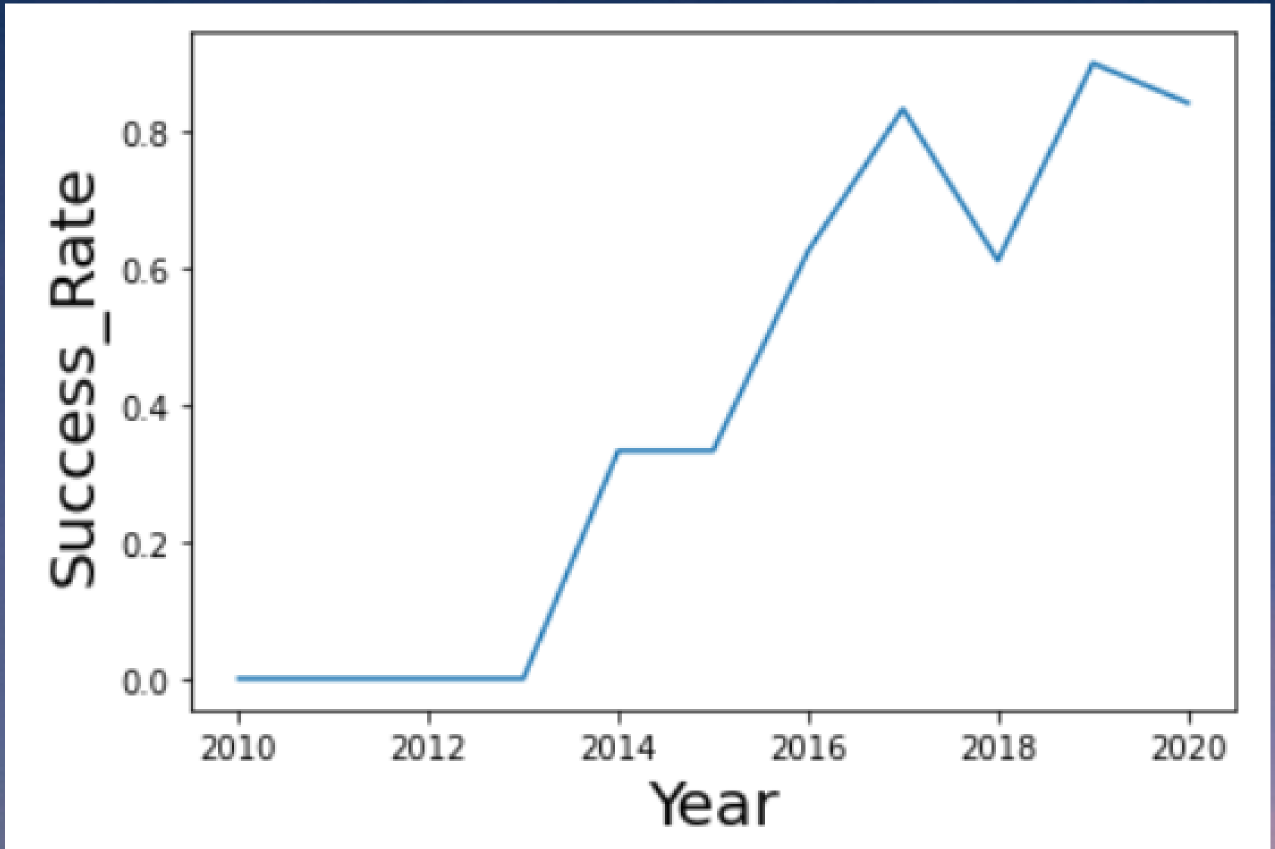
You should see that in the LEO orbit the Success appears related to the number of flights; on the other hand, there seems to be no relationship between flight number when in GTO orbit.

Payload vs. Orbit Type



You should observe that Heavy payloads have a negative influence on GTO orbits and positive on GTO and Polar LEO (ISS) orbits.

Launch Success Yearly Trend



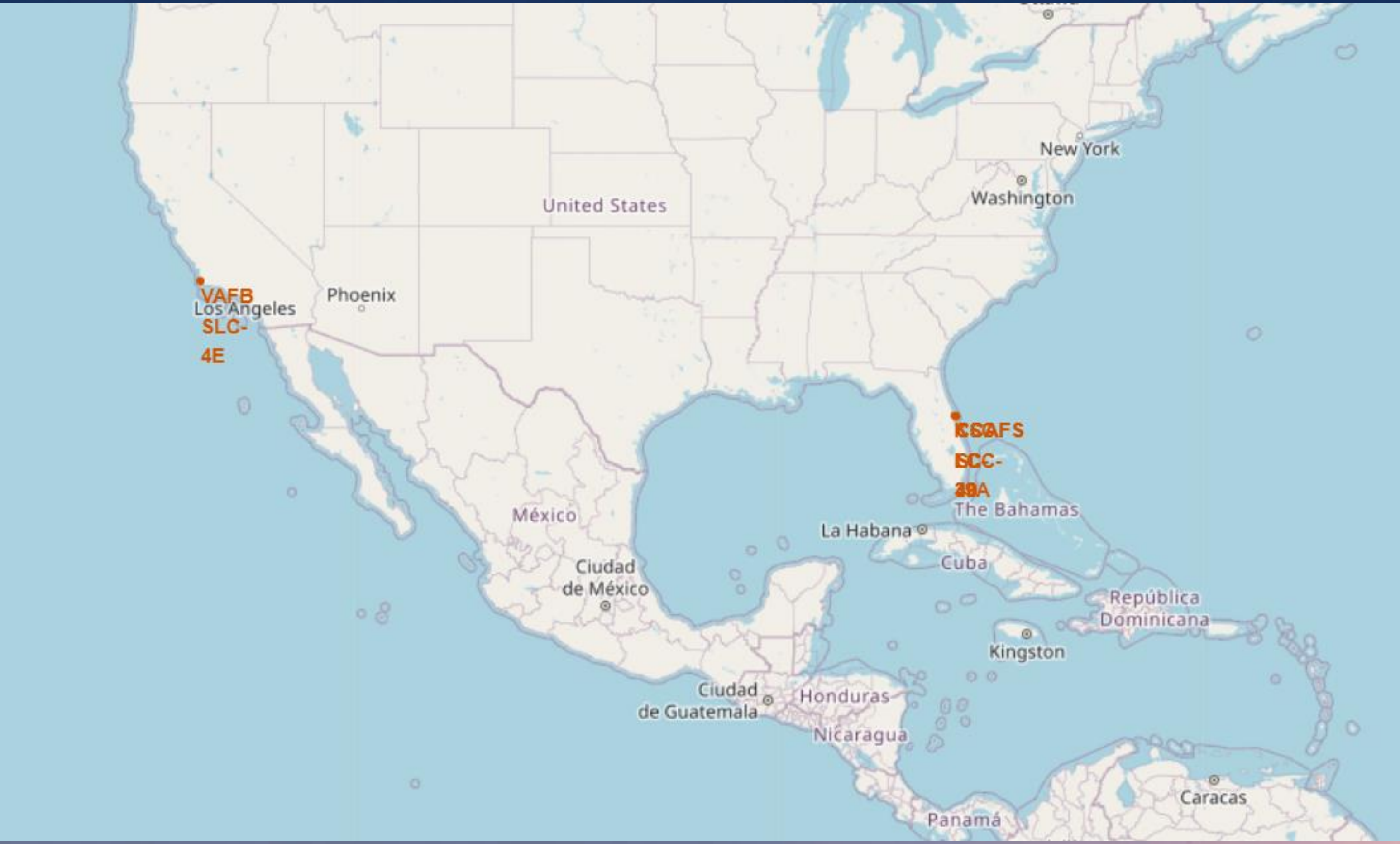
you can observe that the success rate since 2013 kept increasing till 2020

Section 4

Launch sites proximities analysis

Winning Space Race with Data Science

Launch sites map



We can see that the SpaceX launch sites are in the Florida and California coasts.

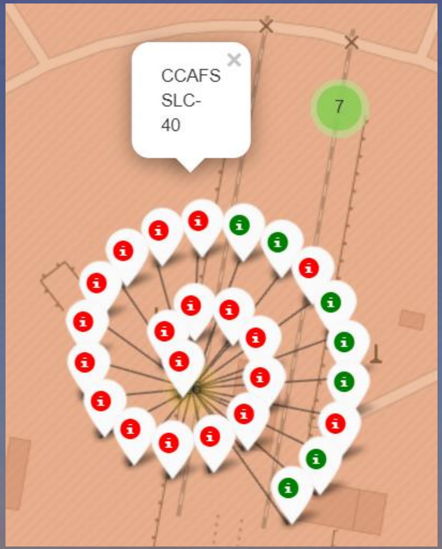
Outcome for each launch site

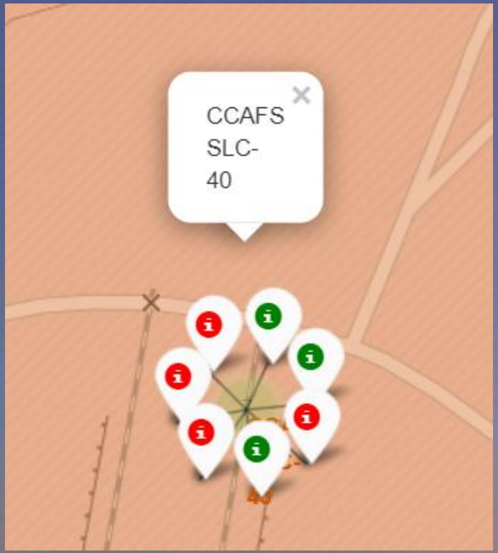
For each launch site are reported in green successful launches and in red unsuccessful launches

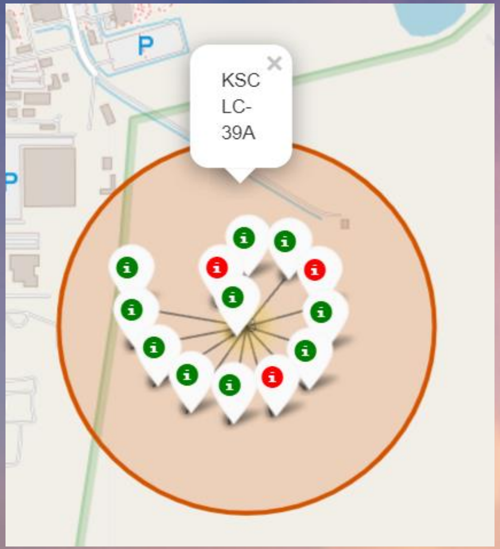
CCAFS SLC 40 - Cape Canaveral Space Force Station - Florida

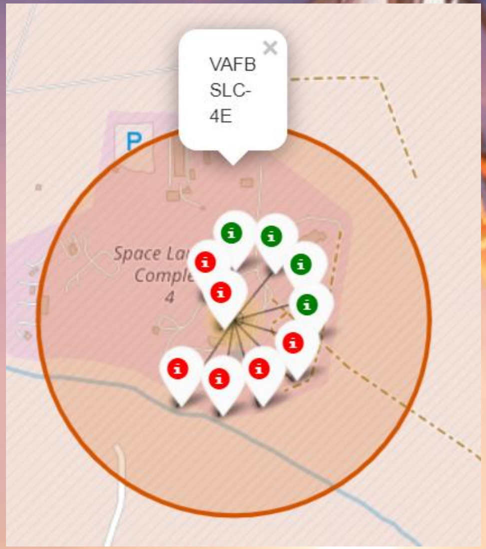
KSC LC 39A - Kennedy Space Center - Florida

VAFB SLC 4E - Vandenberg Air Force Base - California

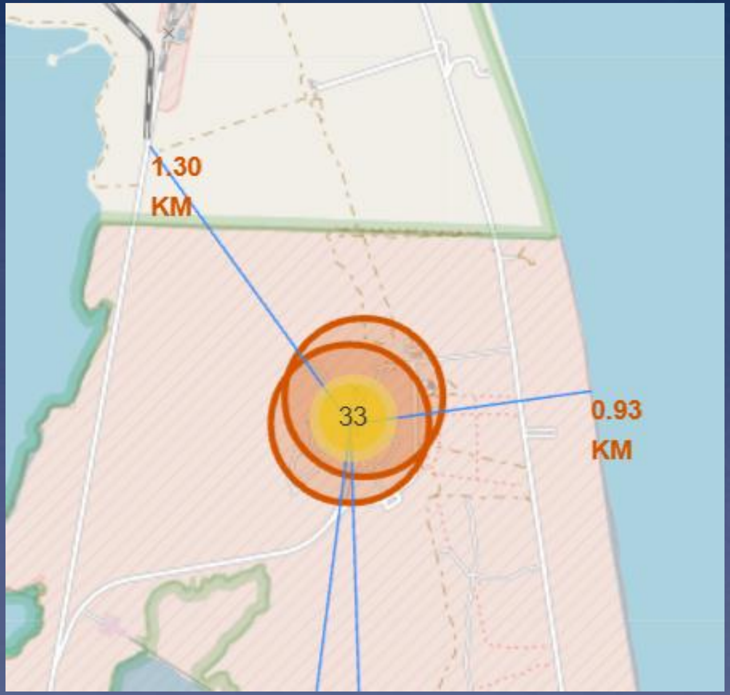


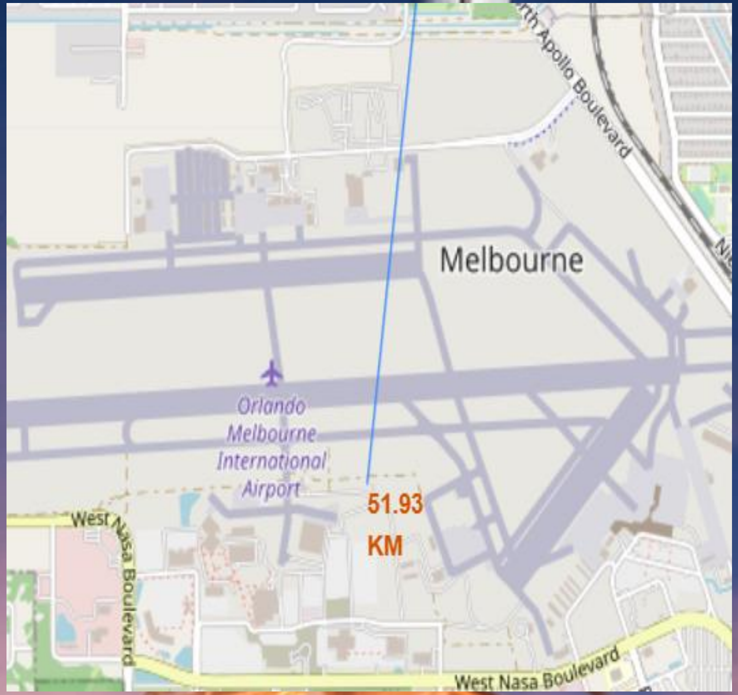






Distance between launche sites and: coastline, railways, highways, cities






```
The distance from "CCAFS LC-40 - Cape Canaveral Space Force Station" to the coast:
0.928 Km

The distance from "CCAFS LC-40 - Cape Canaveral Space Force Station" to the railway:
1.300 Km

The distance from "CCAFS LC-40 - Cape Canaveral Space Force Station" to the highway:
6.967 Km

The distance from "CCAFS LC-40 - Cape Canaveral Space Force Station" to Melbourne:
51.928 Km
```

Are launch sites in close proximity to coastline? Yes

Are launch sites in close proximity to railways? No

Are launch sites in close proximity to highways? No

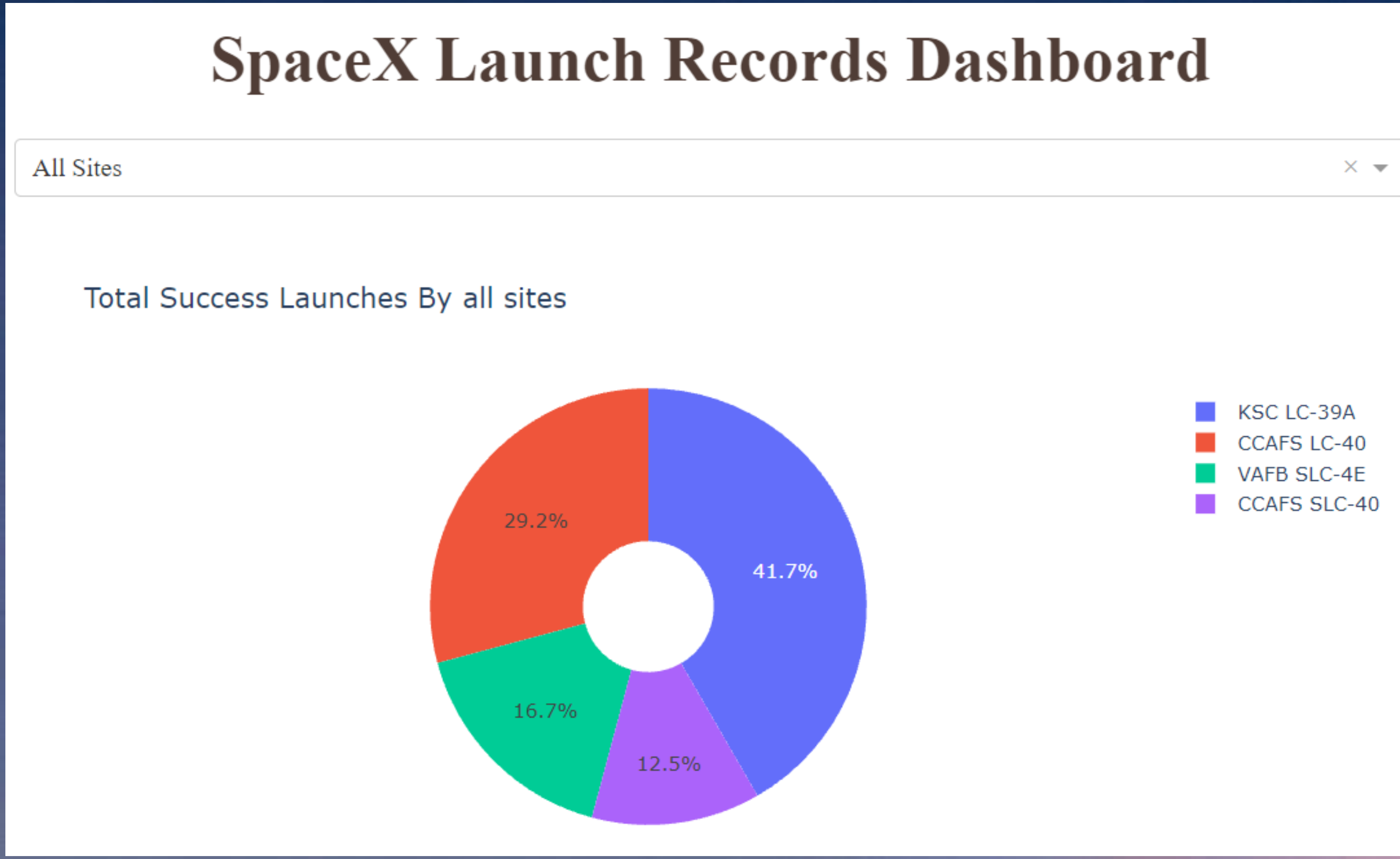
Are launch sites in close proximity to cities? No

Section 5

Plotly Dash dashboard

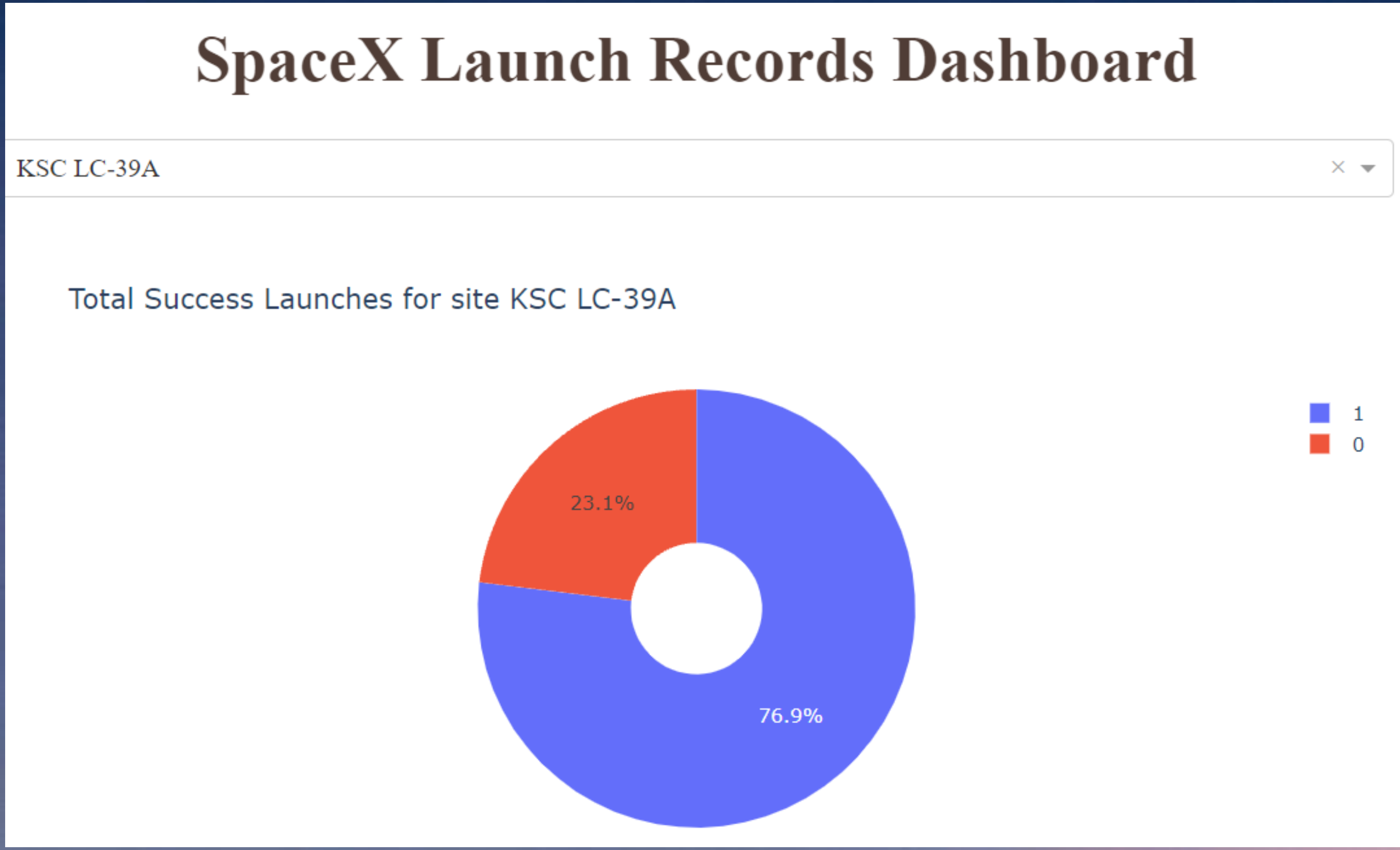
Winning Space Race
with Data Science

Success rate for each launch site



We can see that KSC LC39A had the most successful launches from all the sites

Highest launch success ratio site



KSC LC39A achieved a 76.9% success rate while getting a 23.1% failure rate

Payload vs. Launch Outcome scatter plot

Low Weighted Payload 0kg - 4000kg

Low Weighted Payload 4000kg - 10000kg

We can see the success rates for low weighted payloads is higher than the heavy weighted payloads

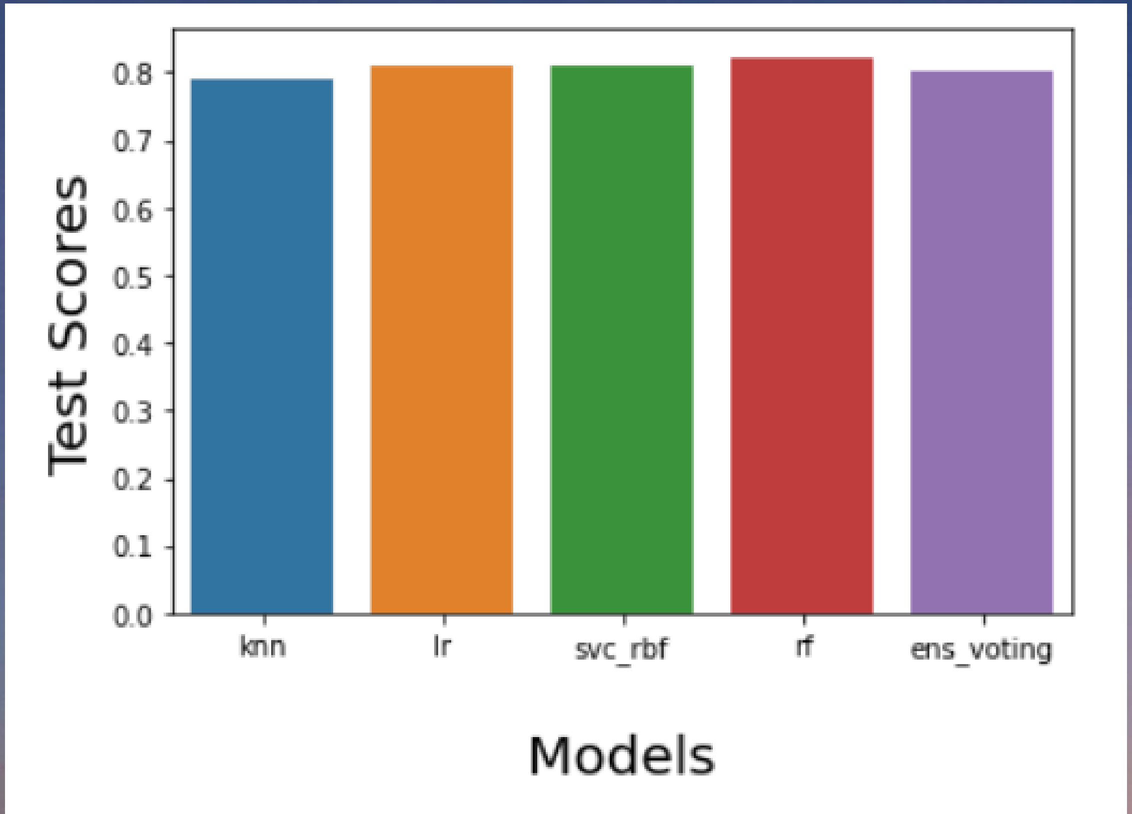
Section 6

Predictive Analysis (Classification)

Winning Space Race with Data Science

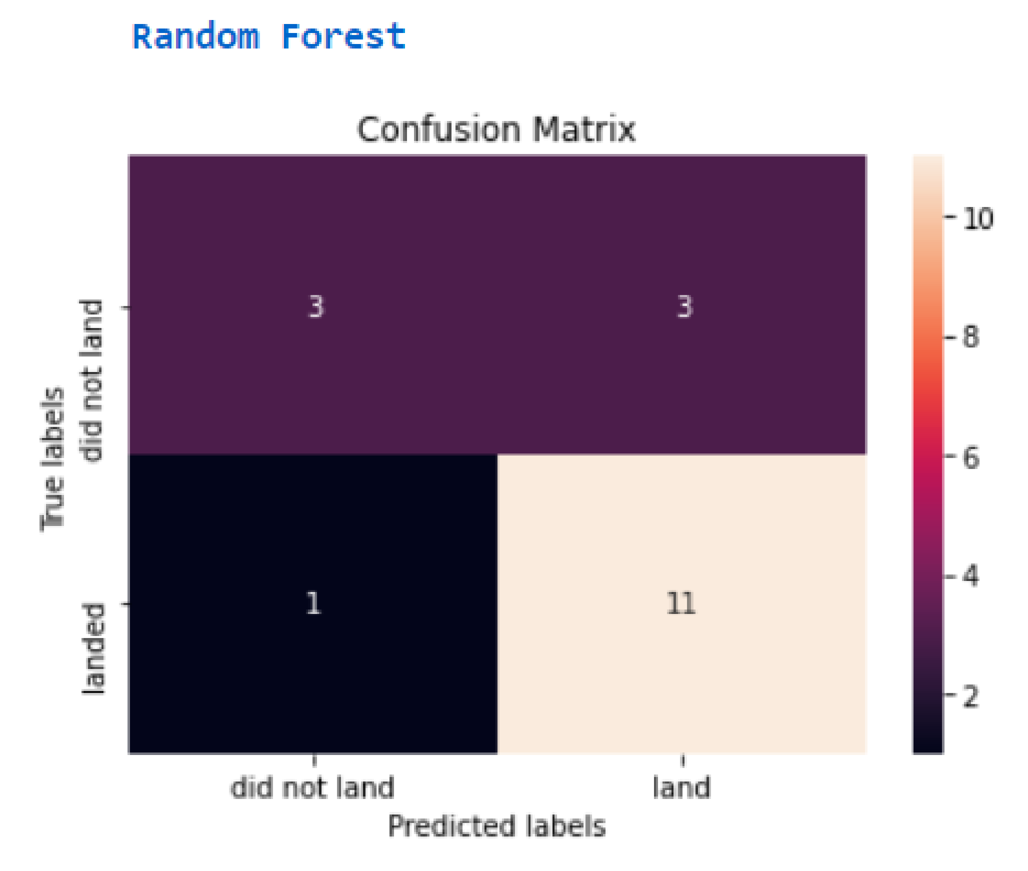
Classification Accuracy

```
Model: knn  Accuracy: 0.7888888888888889%
Model: lr  Accuracy: 0.8111111111111111%
Model: svc_rbf  Accuracy: 0.8111111111111112%
Model: rf  Accuracy: 0.8222222222222222%
Model: ens_voting  Accuracy: 0.7999999999999999%
```



After selecting the best hyper-parameters for the 4 ML algorithms Random Forest results the best performer (82.22%)

Confusion Matrix



Confusion Matrix

	Actually Positive (1)	Actually Negative (0)
Predicted Positive (1)	True Positives (TPs)	False Positives (FPs)
Predicted Negative (0)	False Negatives (FNs)	True Negatives (TNs)

We have 3 False Positive and 1 False Negative.

Conclusions

- All launch sites are very close to the sea, and are fairly close to the equator. In fact the launches of space carriers exploit the sling effect due to the rotational motion of the earth which is precisely greater near the equator
- Are launch sites in close proximity to coastline? Yes
- Are launch sites in close proximity to railways? No
- Are launch sites in close proximity to highways? No
- Are launch sites in close proximity to cities? No
    - The distance from "CCAFS LC-40 - Cape Canaveral Space Force Station" to the coast: 0.928 Km
    - The distance from "CCAFS LC-40 - Cape Canaveral Space Force Station" to the railway: 1.300 Km
    - The distance from "CCAFS LC-40 - Cape Canaveral Space Force Station" to the highway: 6.967 Km
    - The distance from "CCAFS LC-40 - Cape Canaveral Space Force Station" to Melbourne: 51.928 Km
- The orbits on which more launches are made are: "GTO", "ISS", "VLEO"
- The first stage manages to land in 66% of the launches
- With increasing launches and experience they can successfully carry heavier and heavier loads
- The more amount of flights at a launch site the greater the success rate at a launch site.
- The greater the payload mass for Launch Site CCAFS SLC 40 the higher the success rate for the Rocket.
- There is not quite a clear pattern to be found using this visualization to make a decision if the Launch Site is dependant on Pay Load Mass for a success launch.
- Orbit GEO, HEO, SSO, ES-L1 has the best Success Rate

Conclusions

- Heavy payloads have a negative influence on GTO orbits and positive on GTO and Polar LEO (ISS) orbits
- The sucess rate since 2013 kept increasing till 2020
- We can see that KSC LC39A had the most successful launches from all the sites
- KSC LC39A launch site achieved a 76.9% success rate while getting a 23.1% failure rate
- We can see the success rates for low weighted payloads is higher than the heavy weighted payloads
- After selecting the best hyper-parameters for the 4 ML algorithms we achieved the same accuracy (83.33%) on the test data for all models
- For all models we have 3 False Positive¶

Thanks for your attention!

Riccardo Saruis
16-10-2021