IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Francisco Molina Gambin
16/11/2022

Outline

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

Executive Summary

- Summary of methodologies
  - Data Collection trough API
  - Data Collection with Web Scrapping
  - Data Wrangling
  - Exploratory Data Analysis with SQL
  - Exploratory Data Analysis with Data Visualization
  - Interactive Visual Analytics with Folium
  - Machine learning Prediction
- Summary of all results
  - Exploratory Data Analysis results
  - Interactive analytics in screenshots
  - Predictive Analysis results

Introduction

- Project background and context
  - Space Exploration Technologies Corp., conocida como SpaceX, es una empresa estadounidense de fabricación aeroespacial y de servicios de transporte espacial con sede en Hawthorne (California). Fue fundada en 2002 por Elon Musk con el objetivo de reducir los costes de viajar al espacio para facilitar la colonización de Marte. SpaceX ha desarrollado varios vehículos de lanzamiento, la constelación Starlink la nave de carga Dragon y llevado astronautas a la Estación Espacial Internacional en la Dragon 2
  - Problems you want to find answers
    - Factors to determine a successful land of the rocket
    - How to determine a successful rate of successful landing
    - Conditions needed for a successful landing program

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - SpaceX API and web scraping from WIkipedia
- Perform data wrangling
  - One hot enconding to categorical features
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - How to build, tune, evaluate classification models

Data Collection

- Describe how data sets were collected.

Data collection was made with get request in SpaceX API

Then, Json was used using .json() function call and then turned to pandas datraframe with .json normalize()

Then, cleaned data and checked missing values and fill missing values

Also, web scrapping from Wikipedia with BeautifulSoup

For extracting and HTML table and convert it to pandas dataframe

Data Collection – SpaceX API

- Present your data collection with SpaceX REST calls using key phrases and flowcharts
- Add the GitHub URL of the completed SpaceX API calls notebook https://github.com/FranMG99/Final-project/blob/298bd834b4812422768adab9de56ced7a62a58c5/jupyter-labs-spacex-data-collection-api.ipynb

```
spacex_url="https://api.spacexdata.com/v4/launches/past"
```

```
response = requests.get(spacex_url)
```

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

```
# Calculate the mean value of PayloadMass column
PayloadMass = pd.DataFrame(data_falcon9['PayloadMass'].values.tolist()).mean(1)
print(PayloadMass)

# Replace the np.nan values with its mean value
rows = data_falcon9['PayloadMass'].values.tolist()[0]

df_rows = pd.DataFrame(rows)
df_rows = df_rows.replace(np.nan, PayloadMass)

data_falcon9['PayloadMass'][0] = df_rows.values
data_falcon9
```

Data Collection - Scraping

- Present your web scraping process using key phrases and flowcharts
- Add the GitHub URL of the completed web scraping notebook, https://github.com/FranMG99/Final-project/blob/298bd834b4812422768adab9de56ced7a62a58c5/jupyter-labs-webscraping%20(1).ipynb

```
static_url = "https://en.wikipedia.org/w/index.php?title=List_of_Falcon_9_and_Falcon_Heavy_launches&oldid=1027686922"
```

```
# use requests.get() method with the provided static_url
# assign the response to a object
html_data = requests.get(static_url)
html_data.status_code
```

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

Print the page title to verify if the `BeautifulSoup` object was created properly

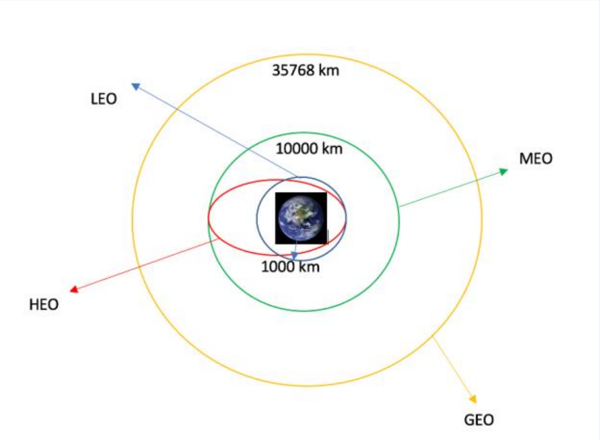
```
# Use soup.title attribute
soup.title
```

```
column_names = []

# Apply find_all() function with `th` element on first_launch_table
# Iterate each th element and apply the provided extract_column_from_header() to get a column name
# Append the Non-empty column name (`if name is not None and len(name) > 0`) into a list called column_names
element = soup.find_all('th')
for row in range(len(element)):
    try:
        name = extract_column_from_header(element[row])
        if (name is not None and len(name) > 0):
            column_names.append(name)
    except:
        pass
```

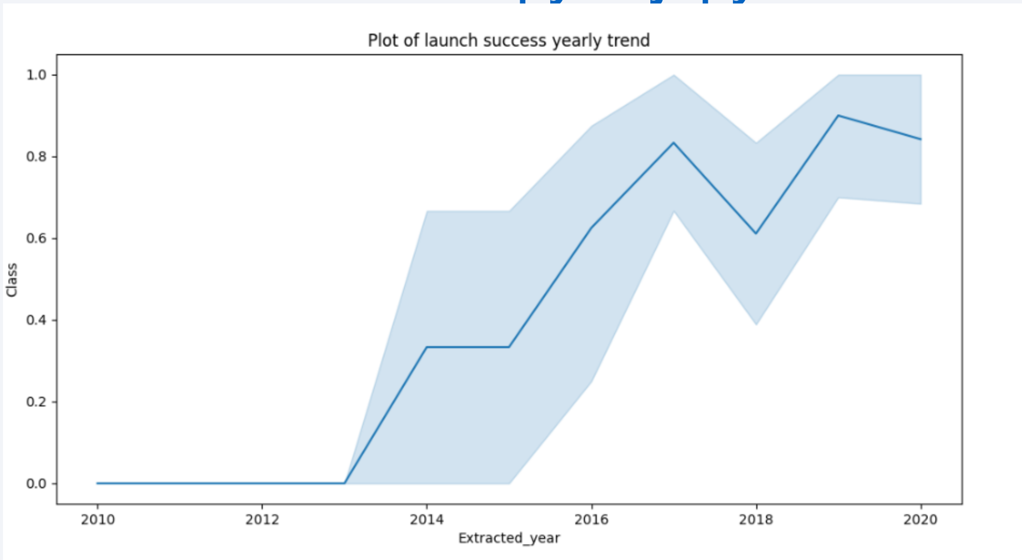
Data Wrangling

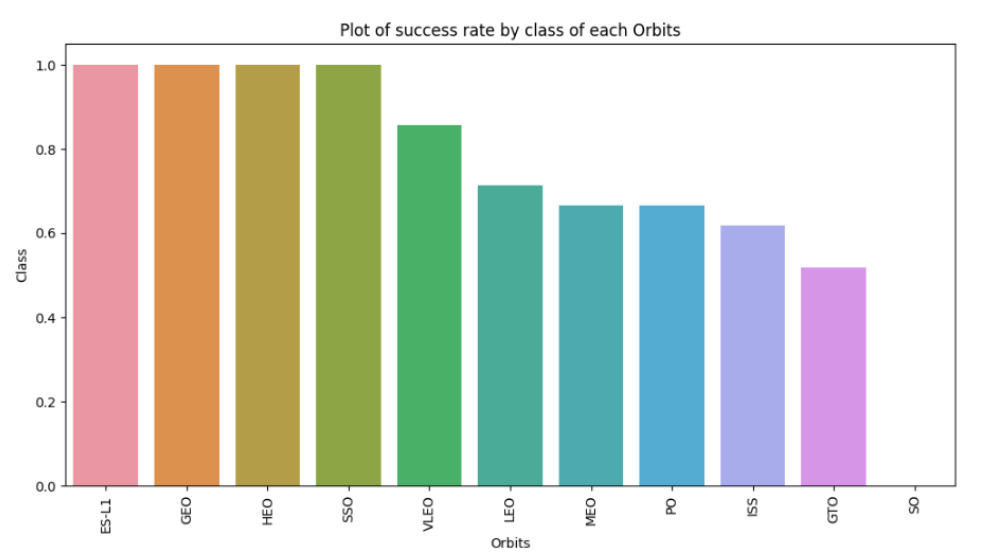
- Data were processed with data analysis and training labels
- Number of launches from the different places and the frequence of the orbits was calculated
- And then we saved to a csv
- Add the GitHub URL of your completed data wrangling related notebooks, https://github.com/FranMG99/Final-project/blob/298bd834b4812422768adab9de56ced7a62a58c5/labs-jupyter-spacex-data_wrangling_jupyterlite.jupyterlite.ipynb



EDA with Data Visualization

- I used a bar plot chart and a line plot chart to observe which orbit have higher success rate and to observe that the success rate sinc e2013 kept increasing till 2020
- Add the GitHub URL of your completed EDA with data visualization notebook, https://github.com/FranMG99/Final-project/blob/f05cef84686f5fa5d333bd6a7b2586107494063b/jupyter-labs-eda-dataviz.ipynb.jupyterlite%20(1).ipynb





EDA with SQL

- Names of unique launch sites
- Total payload mass carried by boosters launched by NASA
- Average payload mass carried by booster version F9 v1.1
- Total number of successful and failure mission outcomes
- Failed landing outcomes in drone ship, booster version and launch site names

- Add the GitHub URL of your completed EDA with SQL notebook, https://github.com/FranMG99/Final-project/blob/f05cef84686f5fa5d333bd6a7b2586107494063b/jupyter-labs-eda-sql-coursera_sqllite.ipynb

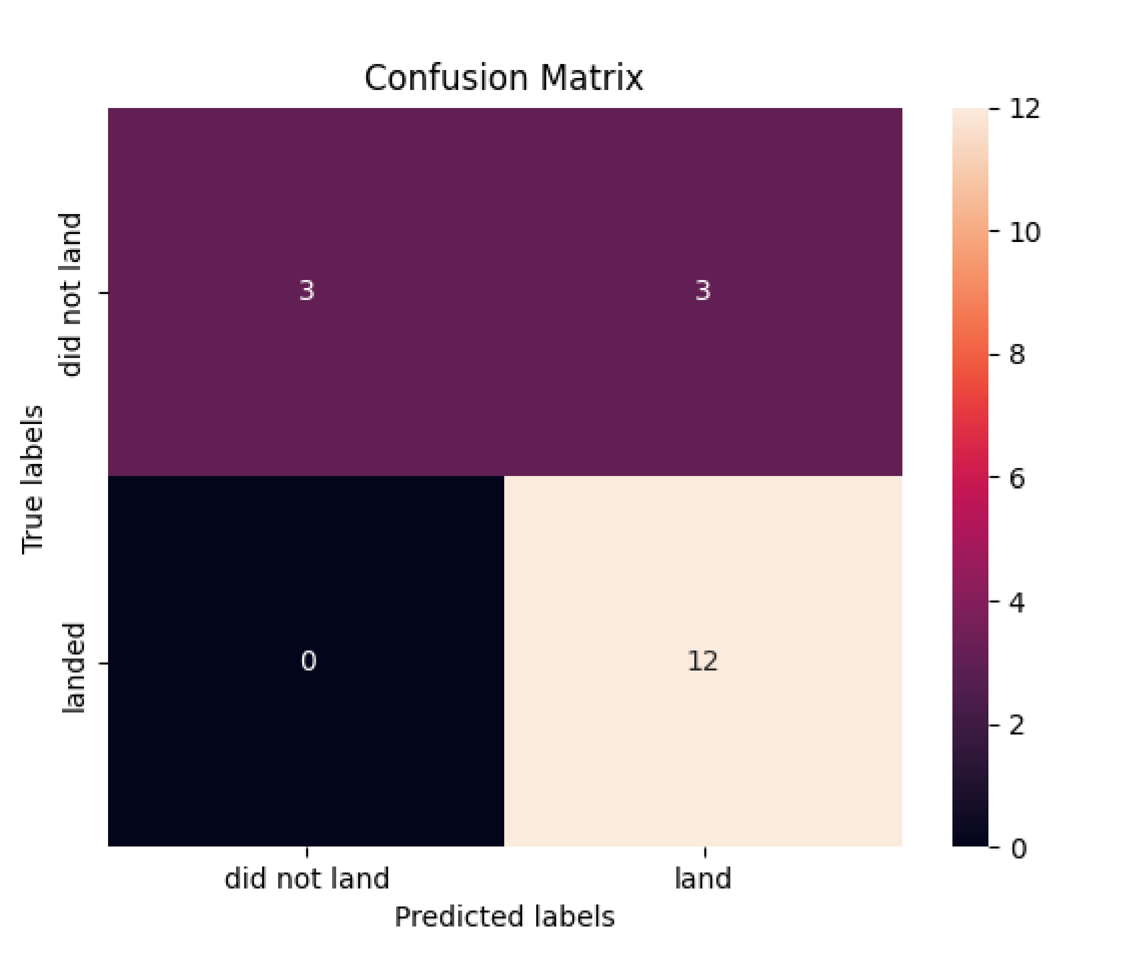
Build an Interactive Map with Folium

- I marked all launch sites, and added map objects such as markers, circles, lines
- I assigned the feature launch outcomes (failure or success) to class 0 and 1
- I identified which launch sites have relatively high success rate, using color-labeled marker clusters
- I calculated the distances between a launch site to its proximities.
- Add the GitHub URL of your completed interactive map with Folium map, https://github.com/FranMG99/Final-project/blob/8930fa67997580 96d5560bc8408ab0760f188679/lab_jupyter_launch_site_location.jupyterlite%20(3).ipynb I don't know why the maps are not shown even that I have the notebook trusted so I put the pics in a pdf in the github too.

Predictive Analysis (Classification)

- I loaded the data using numpy and pandas, transformed the data, and splitted the data into training and testing.
- I built different machine learning models and used GridSearchCV.
- I used accuracy as the metric for the model and improved the model using feature engineering and algorithm tuning.
- And found the best performing classification model
- Add the GitHub URL of your completed predictive analysis lab, https://github.com/FranMG99/Final-project/blob/43cbb27b7bd1e2e6351d23bbfd79eb82439da7ef/SpaceX_Machine_Learning_Prediction_Part_5.jupyterlite.ipynb

Results

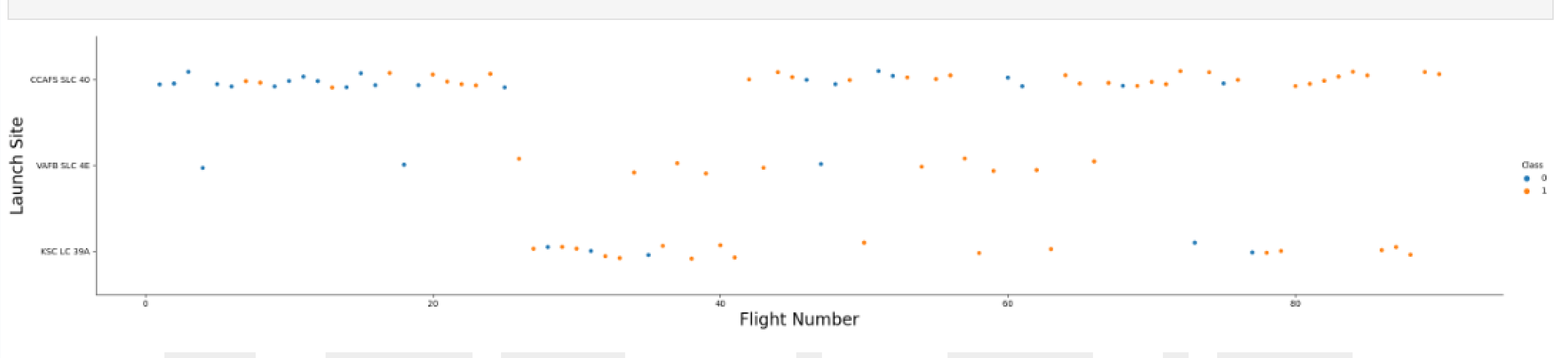


Confusion Matrix
True labels
did not land
landed
3
3
0
12
did not land
land
Predicted labels
12
10
8
6
4
2
0

Section 2

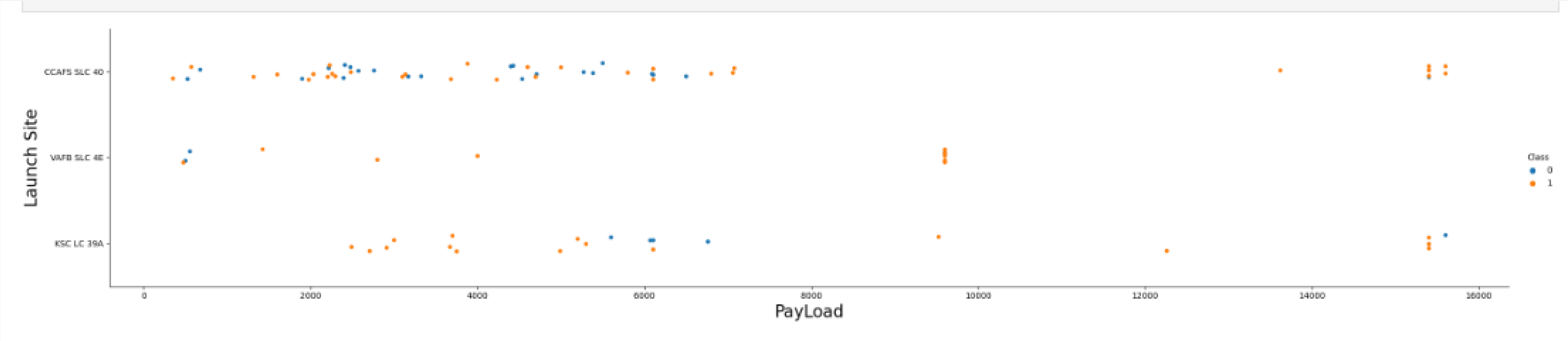
Insights drawn from EDA

Flight Number vs. Launch Site



We see that more amount of the number of the fligst increase the success rate in a launch site

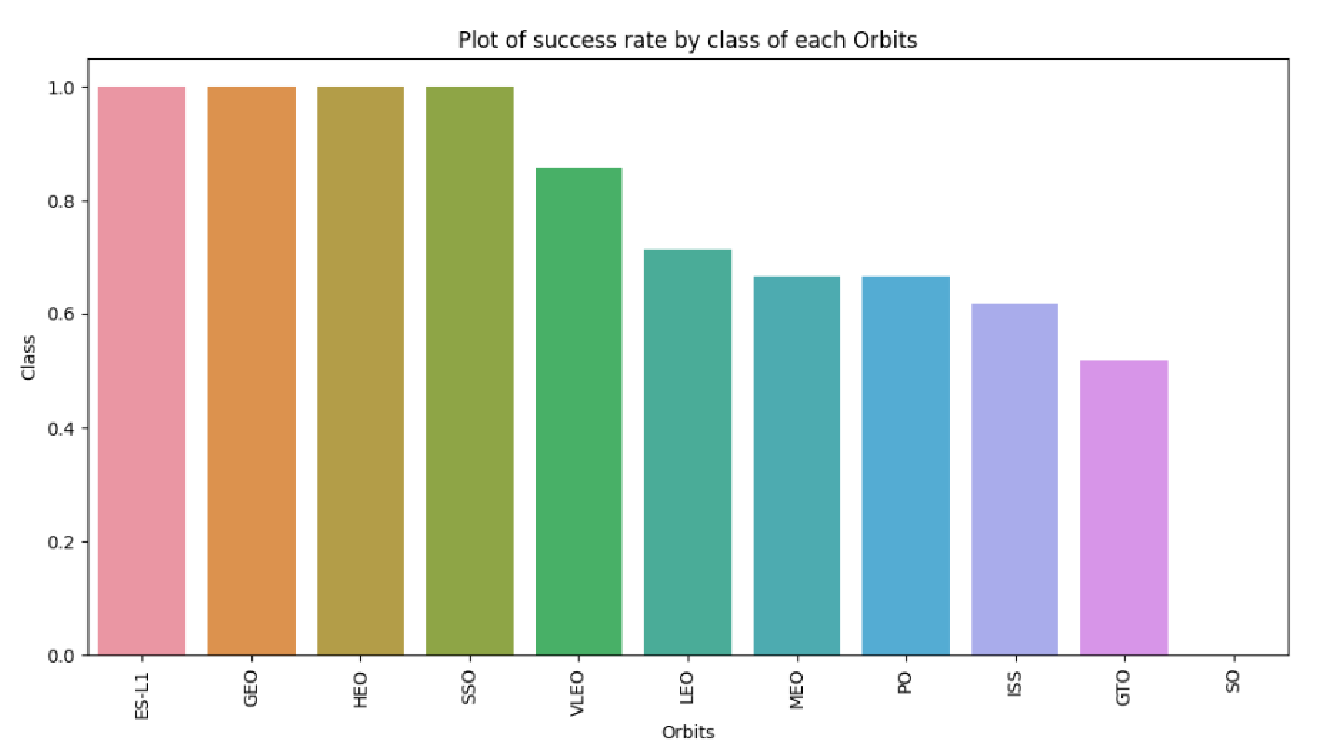
Payload vs. Launch Site



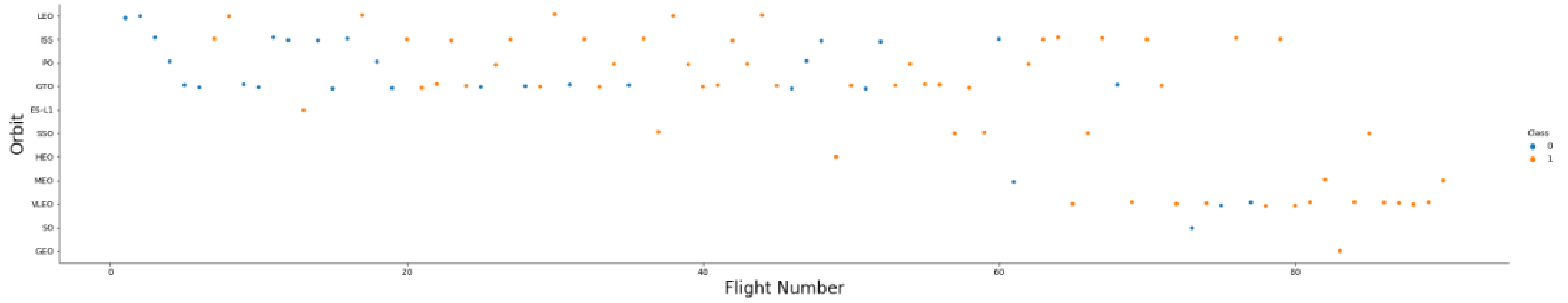
Now if you observe Payload Vs. Launch Site scatter point chart you will find for the VAFB-SLC launch site there are no rockets launched for heavy payload mass

Success Rate vs. Orbit Type

The first four are the ones with more success rate



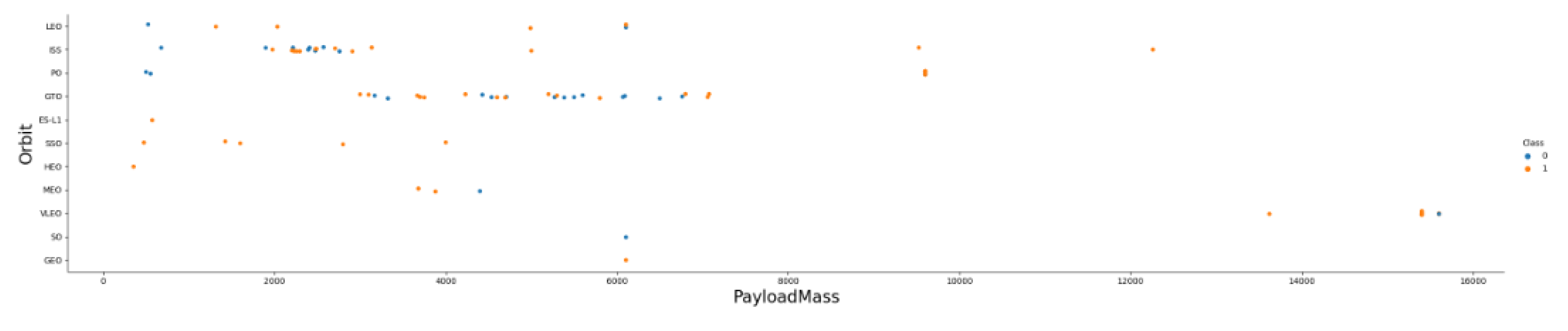
Flight Number vs. Orbit Type



For each orbit, we want to see if there is any relationship between FlightNumber and 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.

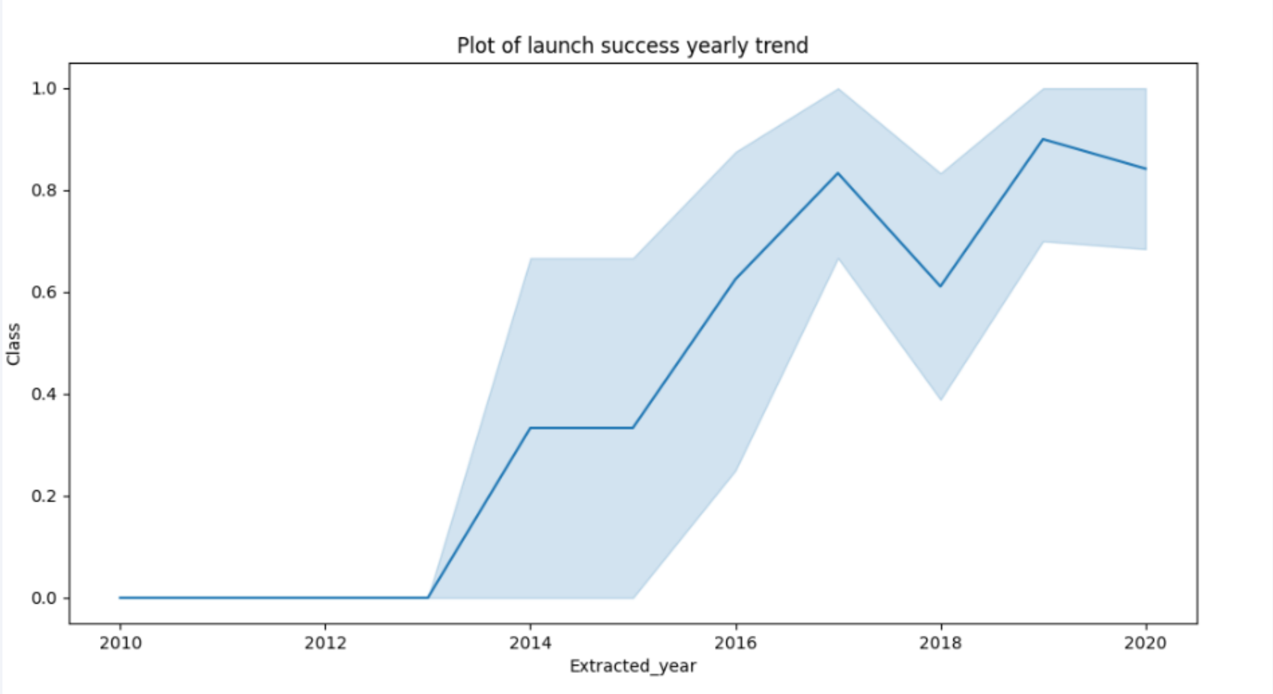
Payload vs. Orbit Type



- With heavy payloads the successful landing or positive landing rate are more for Polar,LEO and ISS.
- However, for GTO we cannot distinguish this well as both positive landing rate and negative landing(unsuccessful mission) are both there here.

Launch Success Yearly Trend

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



All Launch Site Names

The different launch site names are:

KSC LC-39A

CCAFS LC-40

CCAFS SLC-40

VAFB SLC-4E

Launch Site Names Begin with 'CCA'

- Find 5 records where launch sites begin with `CCA`

	date	time	boosterversion	launchsite	
0	2010-04-06	18:45:00	F9 v1.0 B0003	CCAFS LC-40	
1	2010-08-12	15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dr
2	2012-05-22	07:44:00	F9 v1.0 B0005	CCAFS LC-40	
3	2012-08-10	00:35:00	F9 v1.0 B0006	CCAFS LC-40	
4	2013-01-03	15:10:00	F9 v1.0 B0007	CCAFS LC-40	

Total Payload Mass

- The total Payload Mass is 45596

Average Payload Mass by F9 v1.1

- The average payload mass carried by booster version F9 v1.1 is 2928.4

First Successful Ground Landing Date

- The dates of the first successful landing outcome on ground pad was 2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

- List the names of boosters which have successfully landed on drone ship and had payload mass greater than 4000 but less than 6000

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

Total Number of Successful and Failure Mission Outcomes

- Calculate the total number of successful and failure mission outcomes
  - Successful: 100
  - Failure: 1

Boosters Carried Maximum Payload

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

	boosterversion	payloadmasskg
0	F9 B5 B1048.4	15600
1	F9 B5 B1048.5	15600
2	F9 B5 B1049.4	15600
3	F9 B5 B1049.5	15600
4	F9 B5 B1049.7	15600
5	F9 B5 B1051.3	15600
6	F9 B5 B1051.4	15600
7	F9 B5 B1051.6	15600
8	F9 B5 B1056.4	15600
9	F9 B5 B1058.3	15600
10	F9 B5 B1060.2	15600
11	F9 B5 B1060.3	15600

2015 Launch Records

- List the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015

	boosterversion	launchsite	landingoutcome
0	F9 v1.1 B1012	CCAFS LC-40	Failure (drone ship)
1	F9 v1.1 B1015	CCAFS LC-40	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

	landingoutcome	count
0	No attempt	10
1	Success (drone ship)	6
2	Failure (drone ship)	5
3	Success (ground pad)	5
4	Controlled (ocean)	3
5	Uncontrolled (ocean)	2
6	Precluded (drone ship)	1
7	Failure (parachute)	1

Section 3

Launch Sites Proximities Analysis

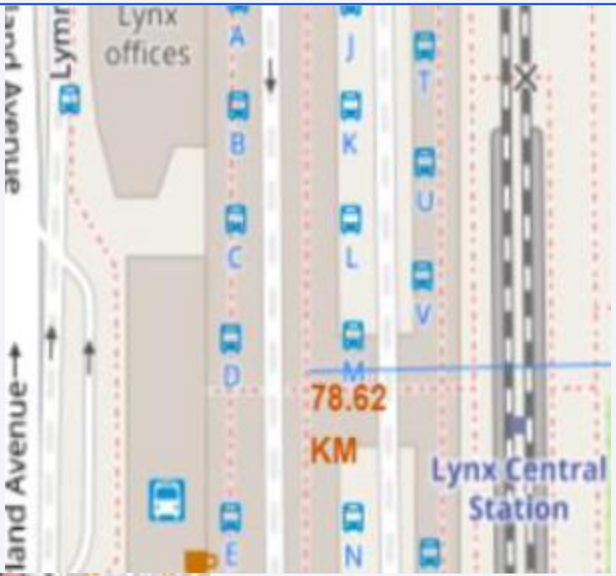
<Folium Map Screenshot 1>

<Folium Map Screenshot 2>

All the launches in california. Green means successful and red failure

<Folium Map Screenshot 3>

Distance to coastline



Distance to train station

Distance to highway

Distance to Orlando

Section 5

Predictive Analysis (Classification)

Classification Accuracy

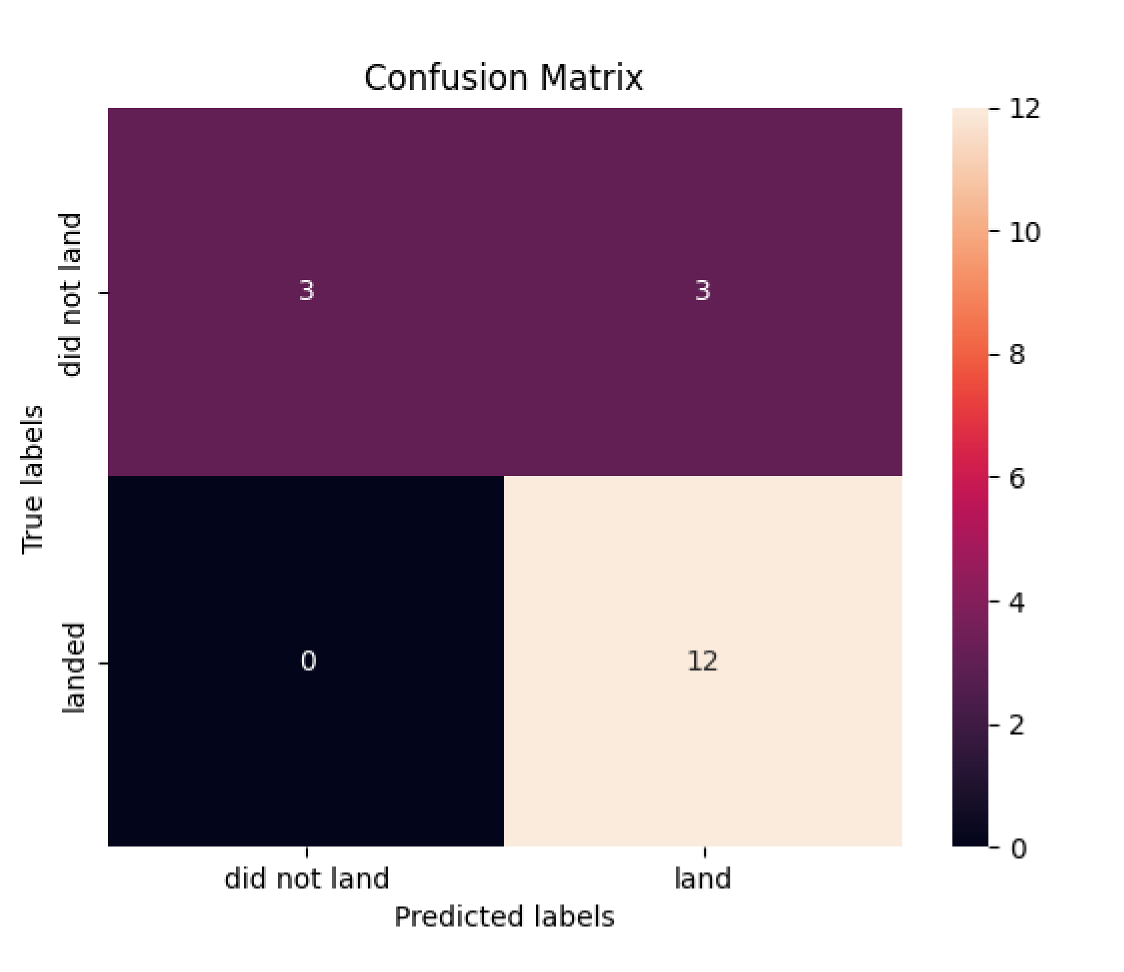
```
models = {'KNeighbors':knn_cv.best_score_,
              'DecisionTree':tree_cv.best_score_,
              'LogisticRegression':logreg_cv.best_score_,
              'SupportVector': svm_cv.best_score_}

bestalgorithm = max(models, key=models.get)
print('Best model is', bestalgorithm,'with a score of', models[bestalgorithm])
if bestalgorithm == 'DecisionTree':
    print('Best params is :', tree_cv.best_params_)
if bestalgorithm == 'KNeighbors':
    print('Best params is :', knn_cv.best_params_)
if bestalgorithm == 'LogisticRegression':
    print('Best params is :', logreg_cv.best_params_)
if bestalgorithm == 'SupportVector':
    print('Best params is :', svm_cv.best_params_)
```

```
Best model is DecisionTree with a score of 0.8732142857142856
Best params is : {'criterion': 'gini', 'max_depth': 6, 'max_features': 'auto', 'min_samples_leaf': 2, 'min_samples_split': 5, 'splitter': 'random'}
```

It says that the decision tree is the most accurate

Confusion Matrix



It shows that the classifier can distinguish between the different classes

Conclusions

- More number of flight at a launch site means that greater is 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 rate.
- KSC LC-39A had the most successful launches of any sites.
- The Decision tree classifier is the best algorithm

Thank you!