IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Twarit Tarpara
Sep 27, 2022

Outline

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

Executive Summary

- **Summary of methodologies**
  - Data Collection
  - Data Wrangling
  - Exploratory Data Analysis with Data Visualization
  - Exploratory Data Analysis with SQL
  - Machine Learning model building
  - Building Dashboards and Interactive maps
- **Summary of all results**
  - Exploratory Data Analysis
  - Machine learning model outcome – Prediction or Prescriptive Analysis
  - Interactive Visual Analytics - Dashboards

Introduction

- **Project background and context**
  - We predicted 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**
  - To determine whether first stage of Falcon 9 rocket would land successfully?
  - Identify the factors required for successful landing of program

Section 1

Methodology

Methodology

Executive Summary

- **Data collection methodology:**
  - Data was collected using SpaceX API
  - Web scraping information from Wikipedia page
- **Perform data wrangling**
  - One-hot encoding was applied 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 – SpaceX API

- We used the get request to the SpaceX API to collect data, clean the requested data and did some basic data wrangling and formatting.

    1 .Getting Response from API

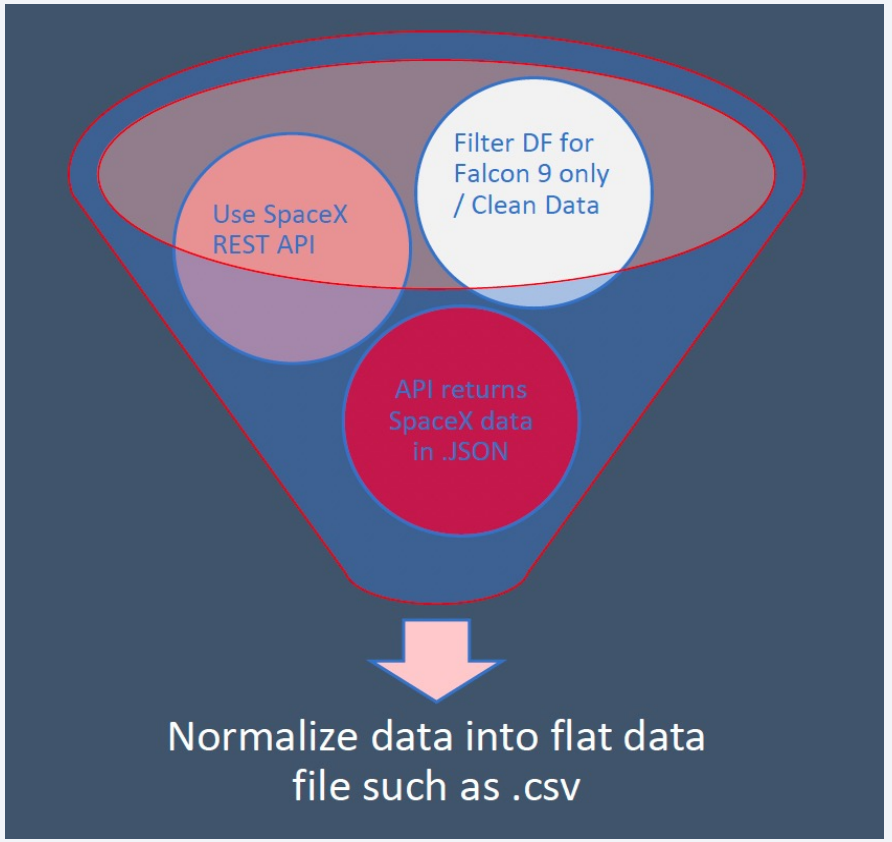
    2. Converting Response to a .json file

    3. Apply custom functions to clean data

    4. Assign list to dictionary then dataframe

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

- Github URL
- https://github.com/twarit/AppliedDataScience/blob/master/SpaceX-Falcon-DataCollection.ipynb



Data Collection – Scraping

We followed below mentioned steps to for data collection using Web Scrapping. To see code how actually it is done, kindly refer to my attached notebook link.

1 .Getting Response from HTML

2. Creating BeautifulSoup Object

3. Finding tables

4. Getting column names

5. Creation of dictionary

6. Appending data to keys

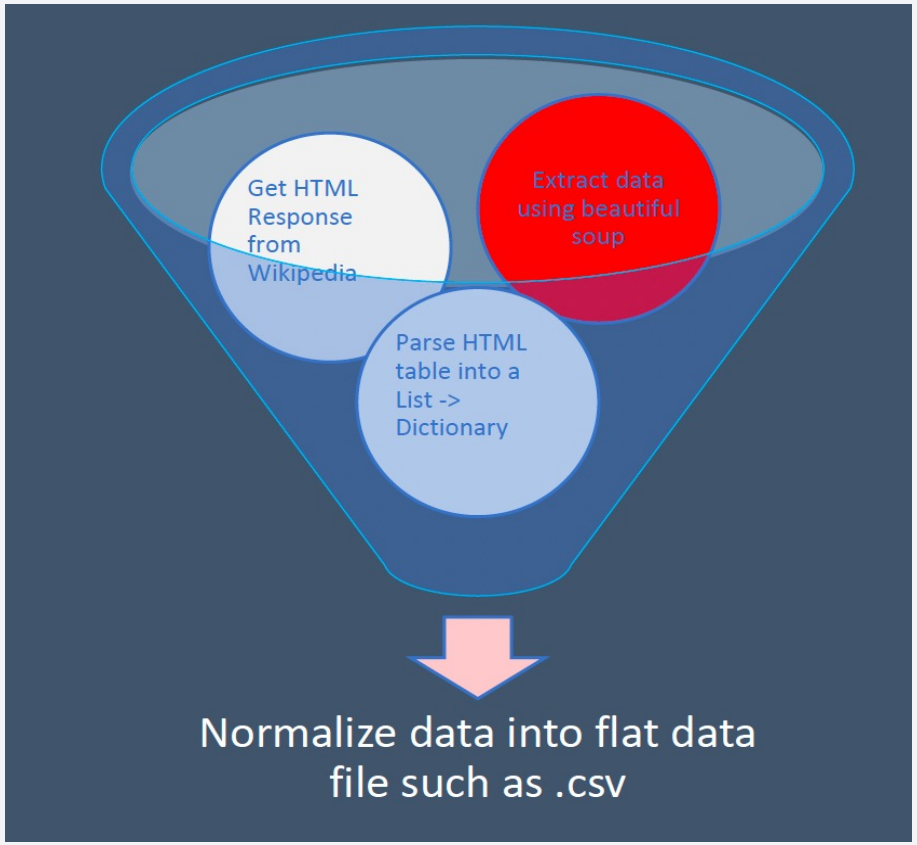
7. Converting dictionary to dataframe

8. Dataframe to .CSV

Github URL:

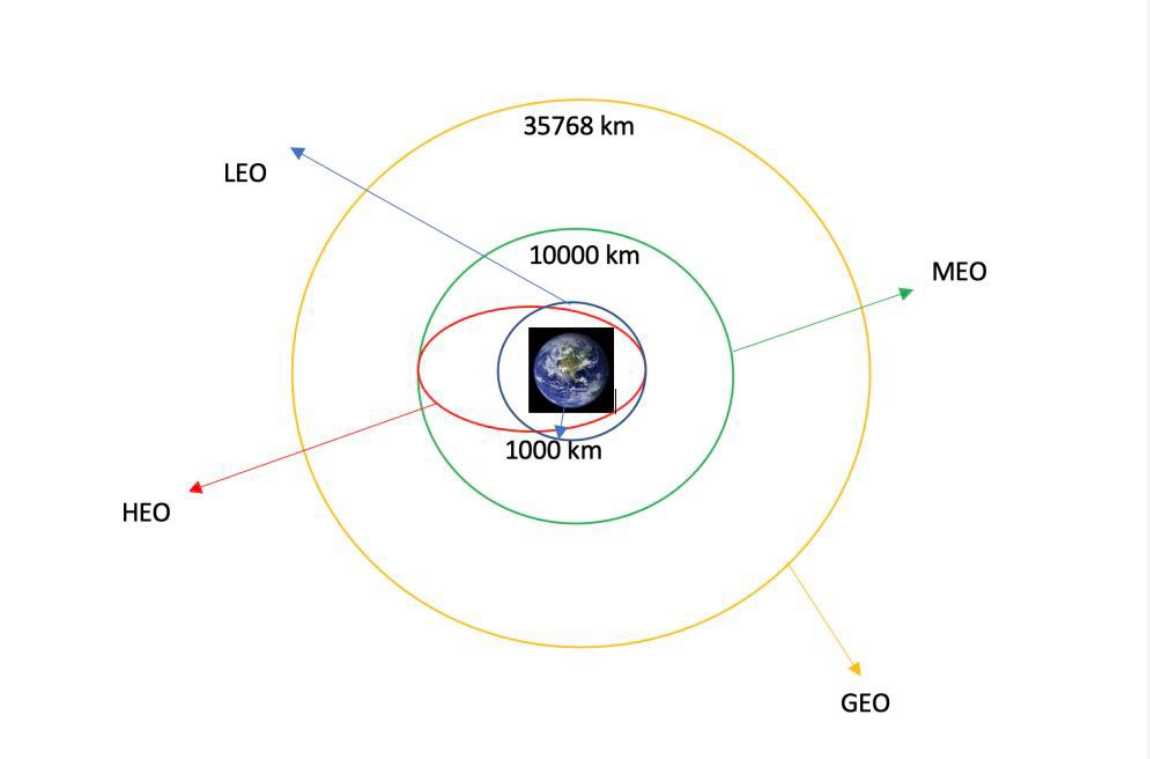
https://github.com/twarit/AppliedDataScience/blob/master

/SpaceX-DataCollection-WebScraping.ipynb



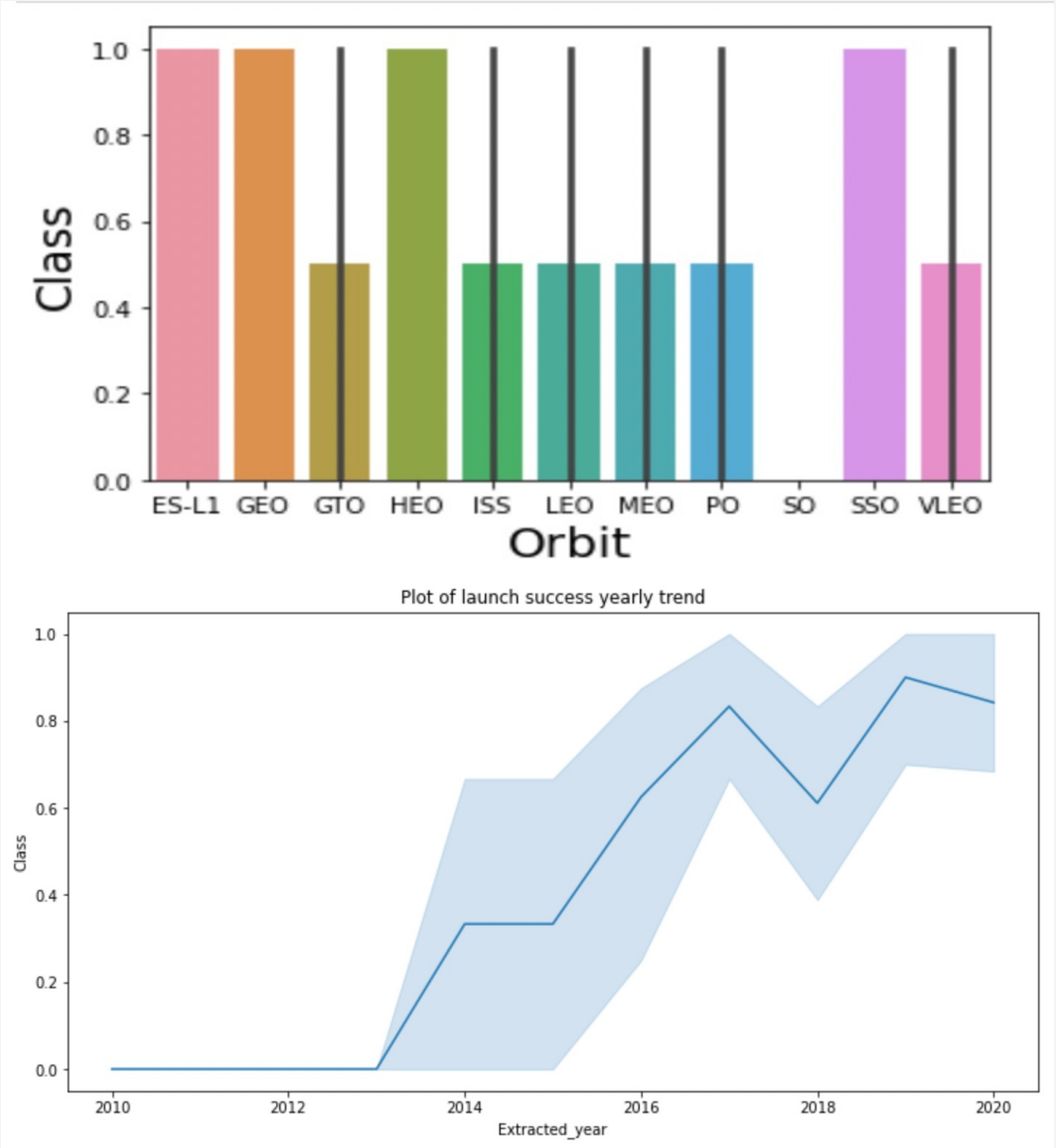
Data Wrangling

- We performed exploratory data analysis and determined the training labels.
- We calculated the number of launches at each site, and the number and occurrence of each orbits
- We created landing outcome label from outcome column and exported the results to csv.
- GitHub URL:
- https://github.com/twarit/AppliedDataScience/blob/master/SpaceX-DataWrangling-EDA.ipynb



EDA with Data Visualization

- **Bar Graph - Relationship between success rate of each orbit type**
  - Analyzed the plotted bar graph to find which orbits have high success rate.
- **Line Graph - The launch success yearly trend**
  - Analyzed the plotted line graph to find yearly trend of launch success rate
- GitHub URL
- https://github.com/twarit/AppliedDataScience/blob/master/EDA_DATAVISUALIZATION.ipynb



EDA with SQL

- **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 mission.
  - The total payload mass carried by boosters launched by NASA (CRS)
  - The average payload mass carried by booster version F9 v1.1
  - The total number of successful and failure mission outcomes
  - The failed landing outcomes in drone ship, their booster version and launch site names

- GitHub URL:
- https://github.com/twarit/AppliedDataScience/blob/master/EDA_DATAVISUALIZATION.ipynb

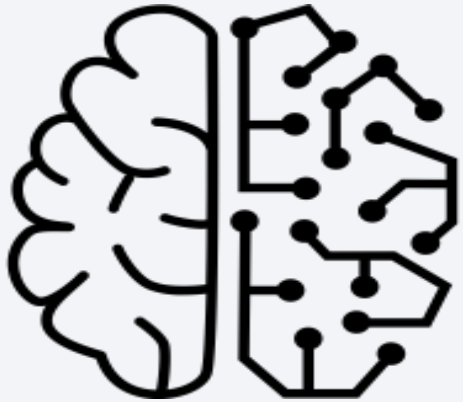
Build an Interactive Map with Folium

- 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()
- We calculated the distances between a launch site to its proximities. We answered some question for instance:
  - Are launch sites near railways, highways and coastlines.
  - Do launch sites keep certain distance away from cities.
- GitHub URL
- https://github.com/twarit/AppliedDataScience/blob/master/Interactive%20Visuals%20%26%20Dashboard%20with%20Folium.ipynb

Predictive Analysis (Classification)

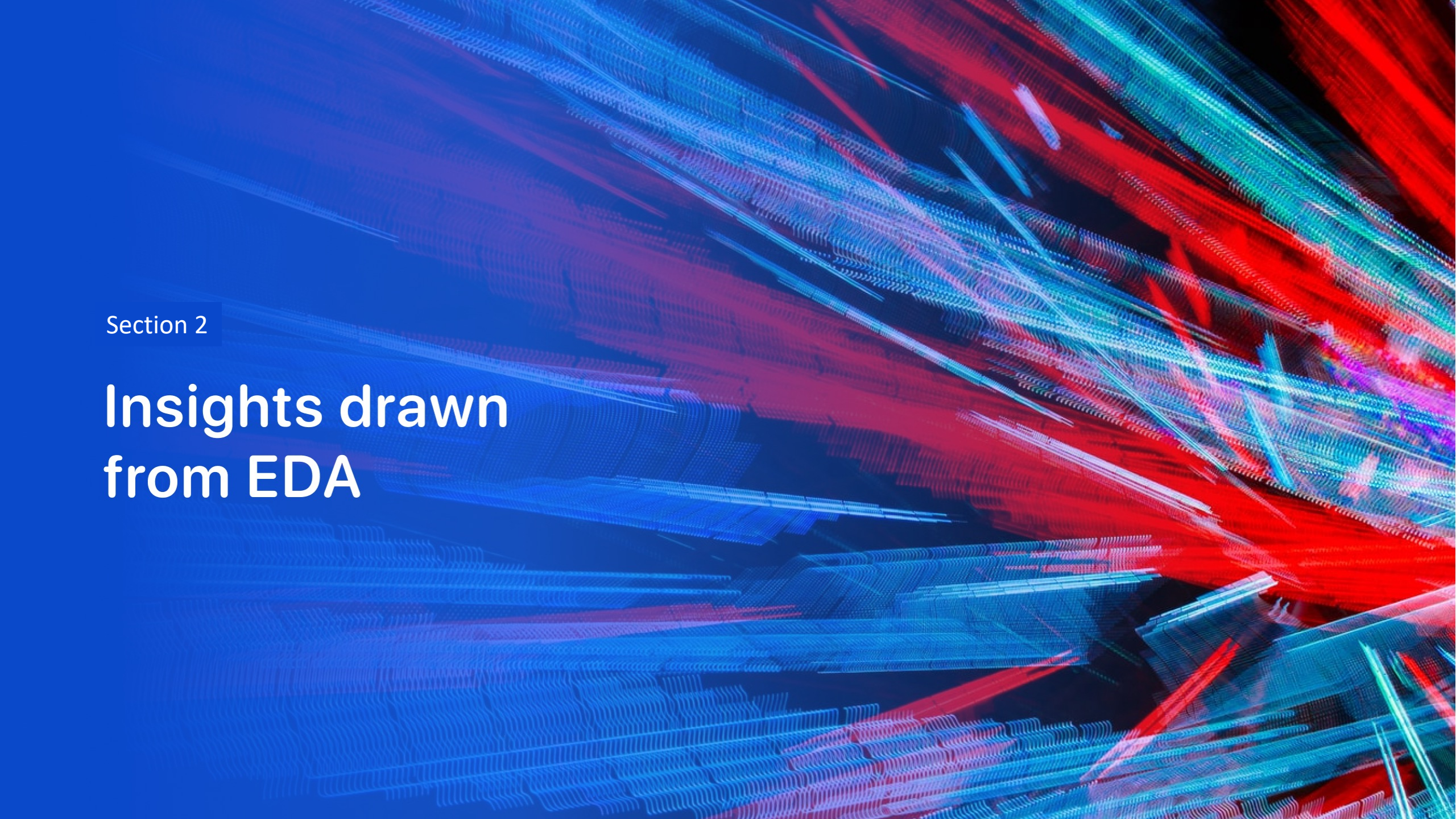
- **BUILDING MODEL**
  - Load our dataset into NumPy and Pandas Data Frame
  - Data Transformation
  - Split our data into training and test data sets
  - Decide which type of machine learning algorithms we want to use
  - •Set our parameters and algorithms to GridSearchCV
  - •Fit our datasets into the GridSearchCVobjects and train our dataset.
- **EVALUATING MODEL**
- **IMPROVING MODEL**
- **FINDING THE BEST PERFORMING CLASSIFICATION MODEL**
- GitHub URL

https://github.com/twarit/AppliedDataScience/blob/master/Machine%20Learning-Predictive%20Analysis.ipynb

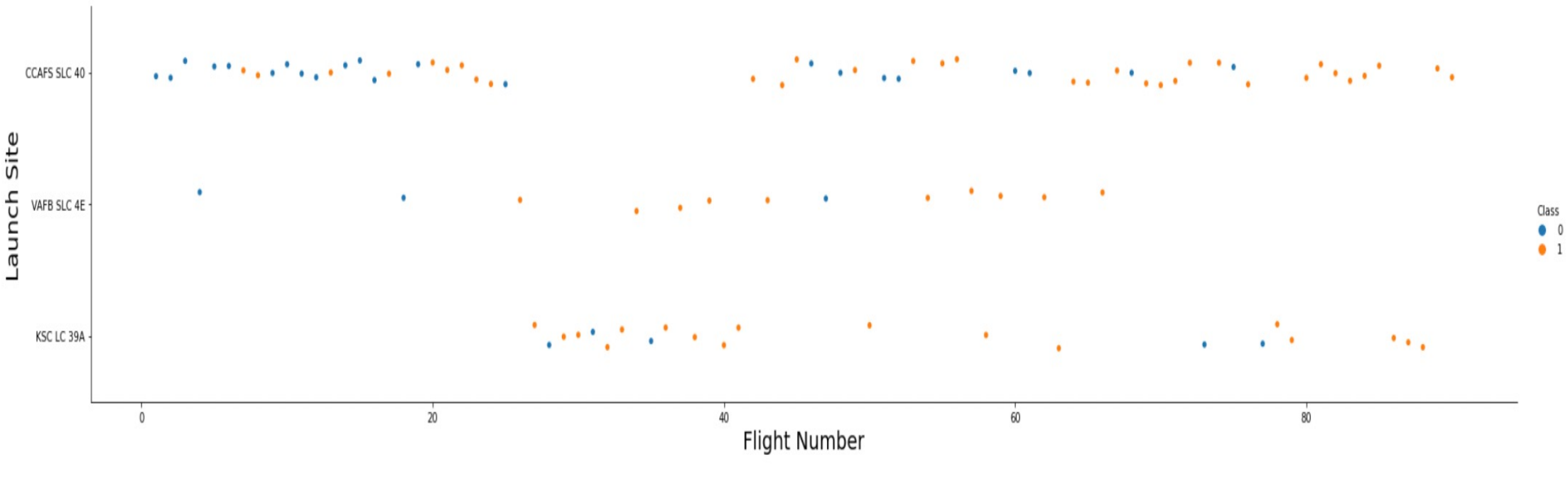


Results

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

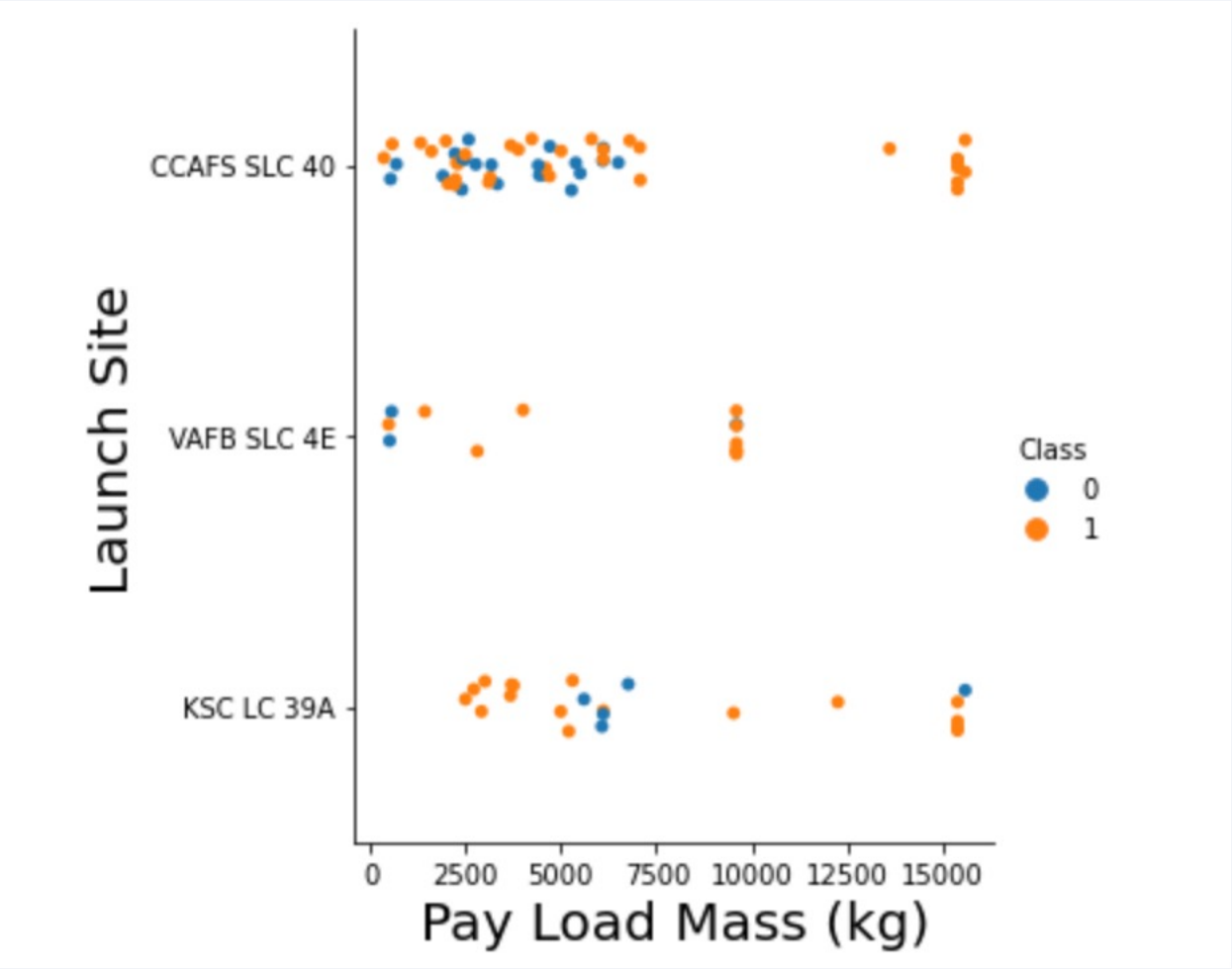

Section 2
Insights drawn from EDA

Flight Number vs. Launch Site



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

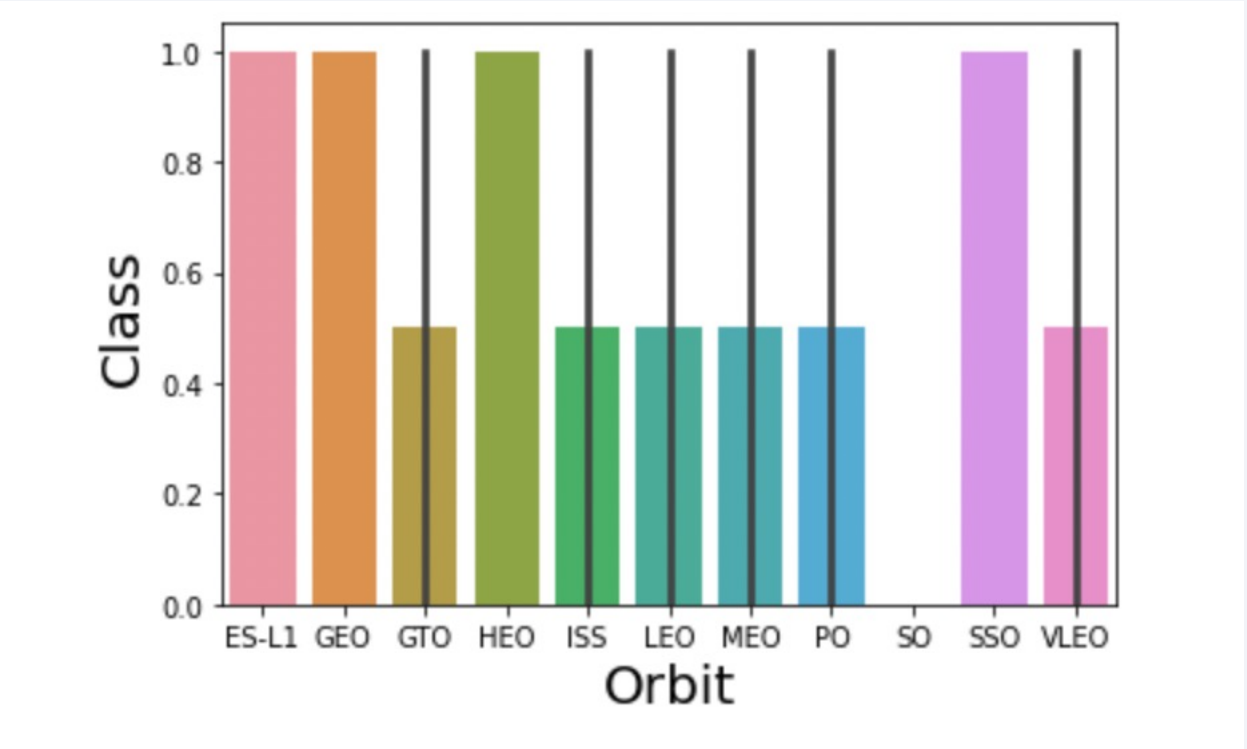
Payload vs. Launch Site



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.

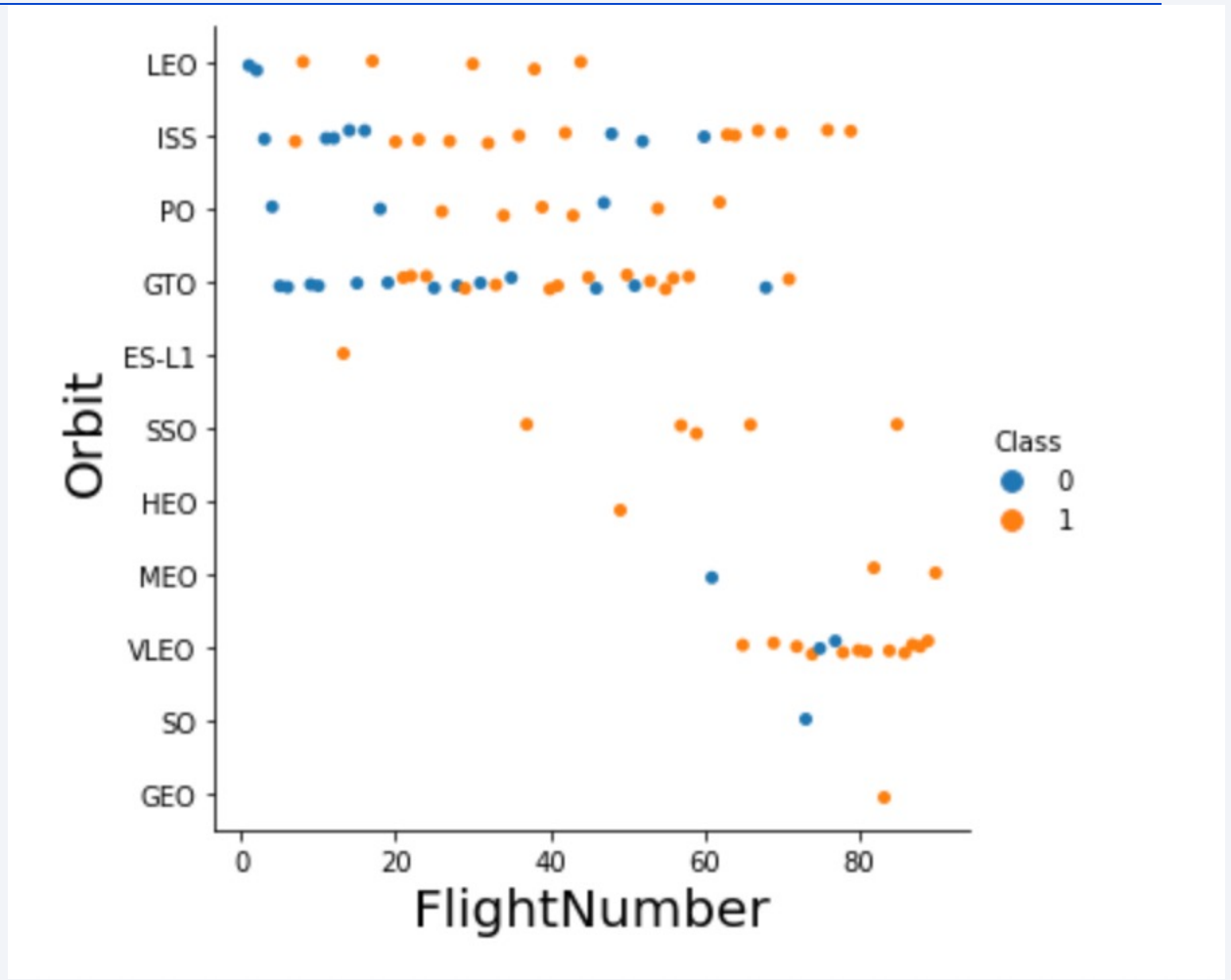
Success Rate vs. Orbit Type

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



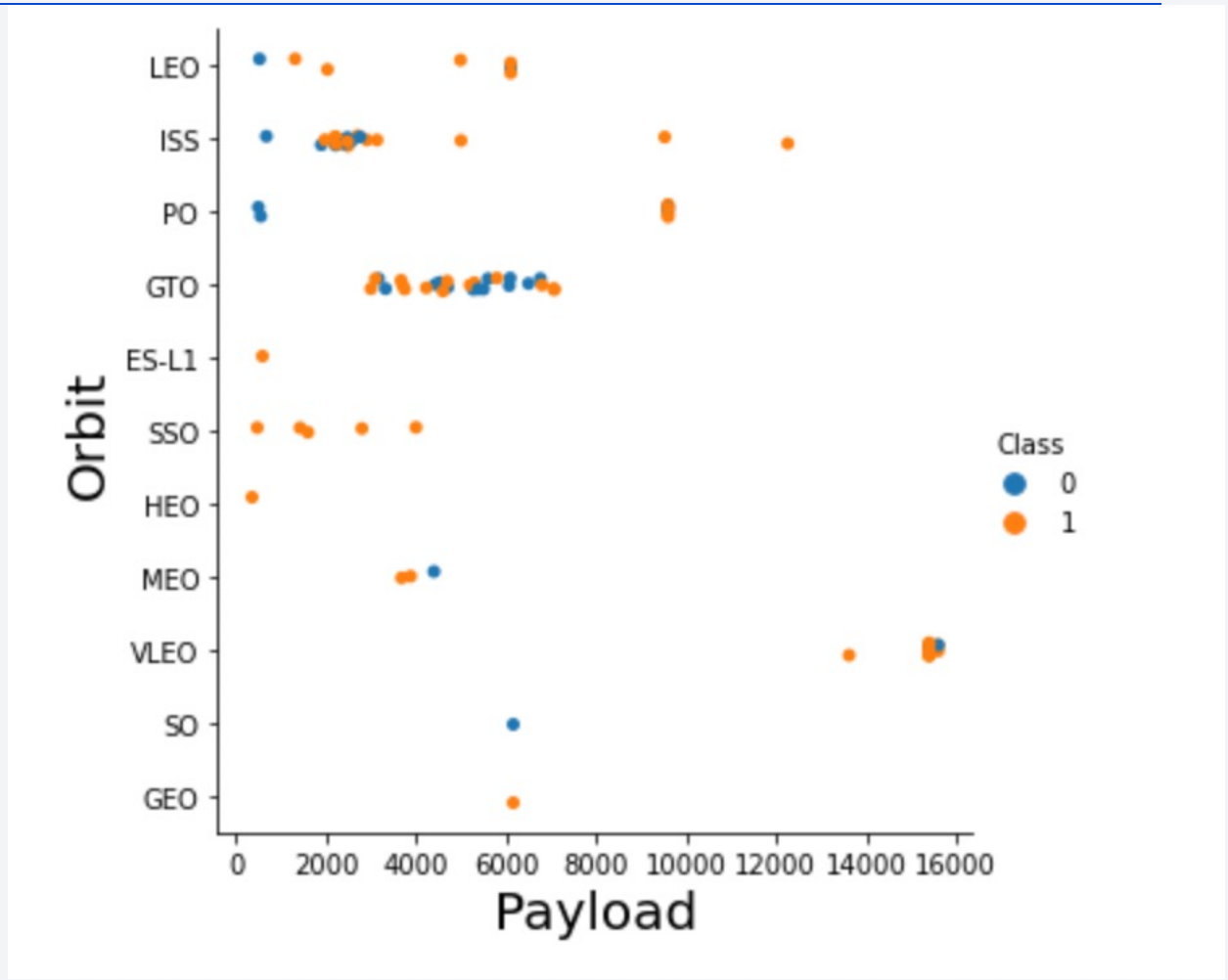
Flight Number vs. Orbit Type

- There is no clear relationship between Flight number and Orbit.



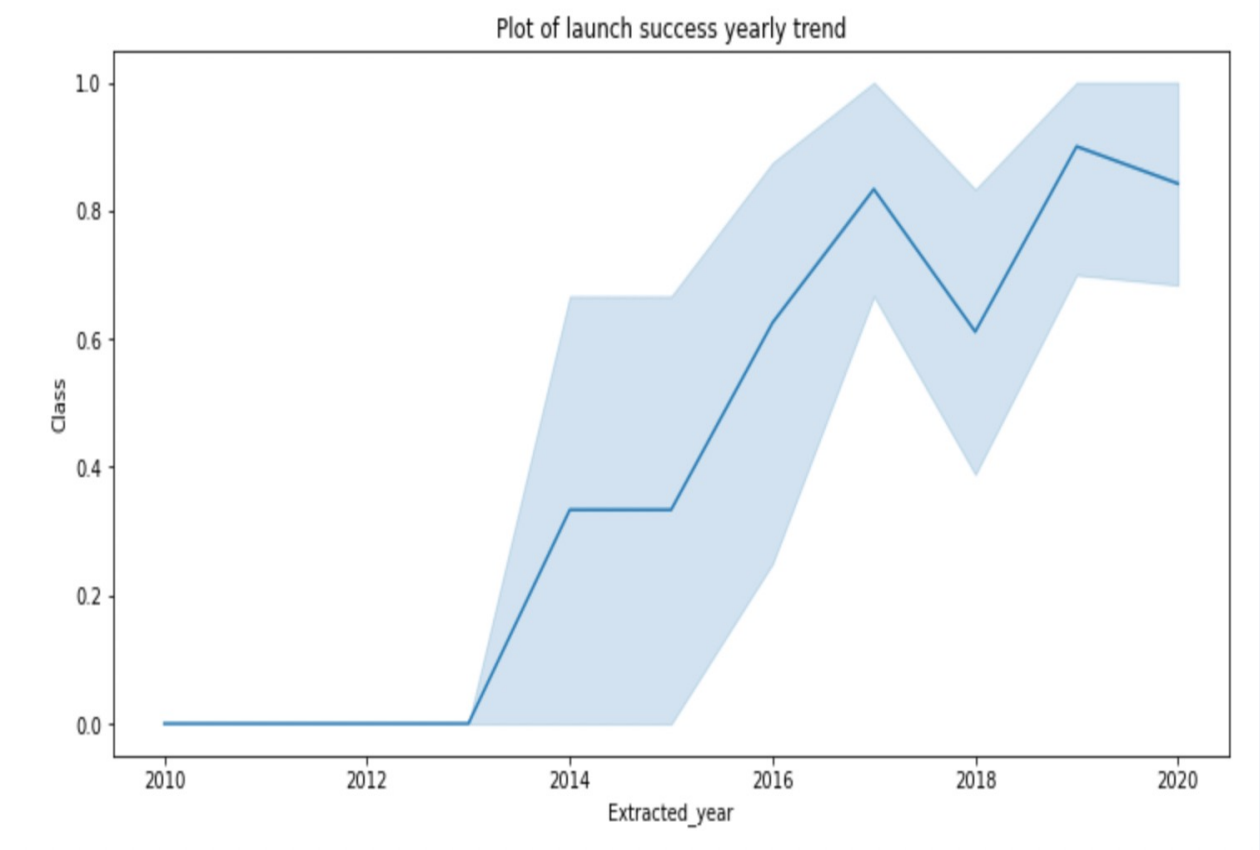
Payload vs. Orbit Type

- Heavy payloads have a negative influence on GTO orbits and positive on GTO and Polar LEO (ISS) orbits.



Launch Success Yearly Trend

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



All Launch Site Names

- Query
- select DISTINCT Launch_Site from tblSpaceX

- **QUERY EXPLAINATION**
- **Using the word DISTINCT in the query means that it will only show Unique values in the Launch_Site column from tblSpaceX**

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

In [10]:
```
task_1 = '''
        SELECT DISTINCT LaunchSite
        FROM SpaceX
'''
create_pandas_df(task_1, database=conn)
```

Out[10]:

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

Launch Site Names Begin with 'CCA'

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

In [11]:
```
task_2 = '''
        SELECT *
        FROM SpaceX
        WHERE LaunchSite LIKE 'CCA%'
        LIMIT 5
        '''
create_pandas_df(task_2, database=conn)
```

Out[11]:

	date	time	boosterversion	launchsite	payload	payloadmasskg	orbit	customer	missionoutcome	landingoutcome
0	2010-04-06	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
1	2010-08-12	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	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
3	2012-08-10	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
4	2013-01-03	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

- We used the query above to display 5 records where launch sites begin with `CCA`

Total Payload Mass

- We calculated the total payload carried by boosters from NASA as 45596 using the query below

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

In [12]:
```
task_3 = '''
        SELECT SUM(PayloadMassKG) AS Total_PayloadMass
        FROM SpaceX
        WHERE Customer LIKE 'NASA (CRS)'
        '''
create_pandas_df(task_3, database=conn)
```

Out[12]:

	total_payloadmass
0	45596

Average Payload Mass by F9 v1.1

We used the avg function to calculate the average payload mass carried by booster version F9 v1.

```
%sql select avg(payload_mass__kg_) from SPACEXDATASET where booster_version = 'F9 v1.1'
```

```
 * ibm_db_sa://ftm88823:***@98538591-7217-4024-b027-8baa776ffad1.c3n41cmd0nqnrk39u98g.databases.appdomain.cloud:30875/BLUDB
Done.
```

1
2928

First Successful Ground Landing Date

We used min function to extract the first successful landing outcome on ground pad.

```
%sql select min(DATE) from SPACEXDATASET where landing__outcome = 'Success (ground pad)'
```

```
 * ibm_db_sa://ftm88823:***@98538591-7217-4024-b027-8baa776ffad1.c3n41cmd0nqnrk39u98g.databases.appdomain.cloud:30875/BLUDB
Done.
```

1
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

We extracted the names of boosters which have successfully landed on drone ship and had payload mass greater than 4000 but less than 6000 using the between function to define the range.

```
%sql select booster_version from SPACEXDATASET where landing__outcome = 'Success (drone ship)' and payload_mass__kg_ between 4000 and 600(
```

```
 * ibm_db_sa://ftm88823:***@98538591-7217-4024-b027-8baa776ffad1.c3n41cmd0nqnrk39u98g.databases.appdomain.cloud:30875/BLUDB
Done.
```

booster_version
F9 FT B1022
F9 FT B1026
F9 FT B1021.2
F9 FT B1031.2

Total Number of Successful and Failure Mission Outcomes

- We used the count and group by function to calculate the total number of of success and failure mission outcomes

```
%sql select mission_outcome, count(*) from SPACEXDATASET where mission_outcome in ('Success','Failure (in flight)') group by mission_outcc
```

```
 * ibm_db_sa://ftm88823:***@98538591-7217-4024-b027-8baa776ffad1.c3n41cmd0nqnrk39u98g.databases.appdomain.cloud:30875/BLUDB
Done.
```

mission_outcome	2
Failure (in flight)	1
Success	99

Boosters Carried Maximum Payload

```
%sql select booster_version from SPACEXDATASET where payload_mass__kg_ = (select max(payload_mass__kg_) from SPACEXDATASET)
```

```
 * ibm_db_sa://ftm88823:***@98538591-7217-4024-b027-8baa776ffad1.c3n41cmd0nqnrk39u98g.databases.appdomain.cloud:30875/BLUDB
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

2015 Launch Records

We used the extract function to extract the year from the date field and list the 2015 launch records.

```
%sql select landing__outcome, booster_version , launch_site, extract(year from DATE) as year from SPACEXDATASET where landing__outcome =
```

```
 * ibm_db_sa://ftm88823:***@98538591-7217-4024-b027-8baa776ffad1.c3n41cmd0nqnrk39u98g.databases.appdomain.cloud:30875/BLUDB
Done.
```

landing__outcome	booster_version	launch_site	YEAR
Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40	2015
Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40	2015

Rank Landing Outcomes Between 2010-06-04 and 2017-03-20

We used the order by and descending function to list the landing outcomes in decreasing ranking.

```
In [29]: %sql select landing__outcome, count(*) as total from SPACEXDATASET where DATE between '2010-06-04' and '2017-03-20' group by landing__outcome order by total desc
```

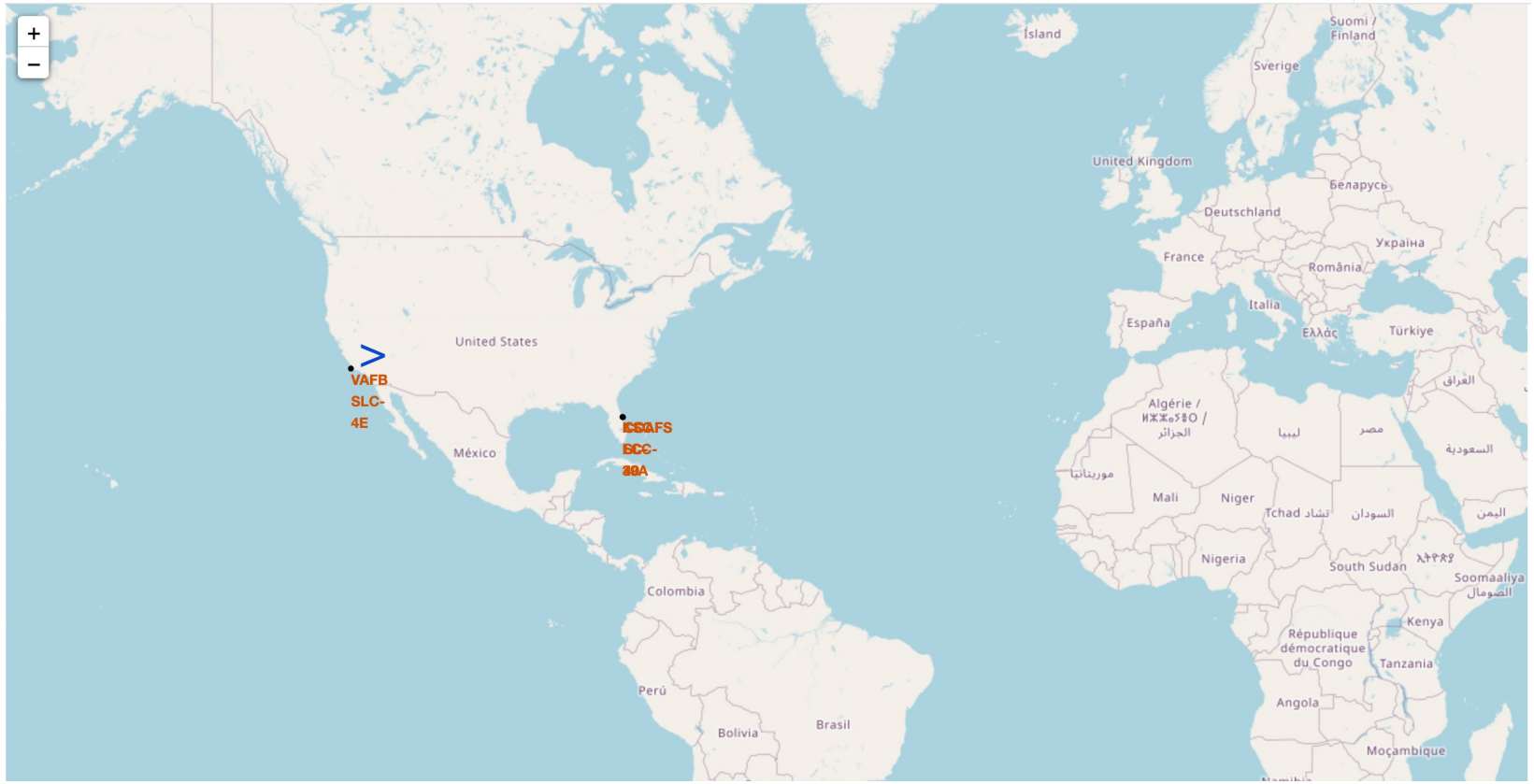
```
 * ibm_db_sa://ftm88823:***@98538591-7217-4024-b027-8baa776ffad1.c3n41cmd0nqnrk39u98g.databases.appdomain.cloud:30875/BLUDB
Done.
```

Out[29]:

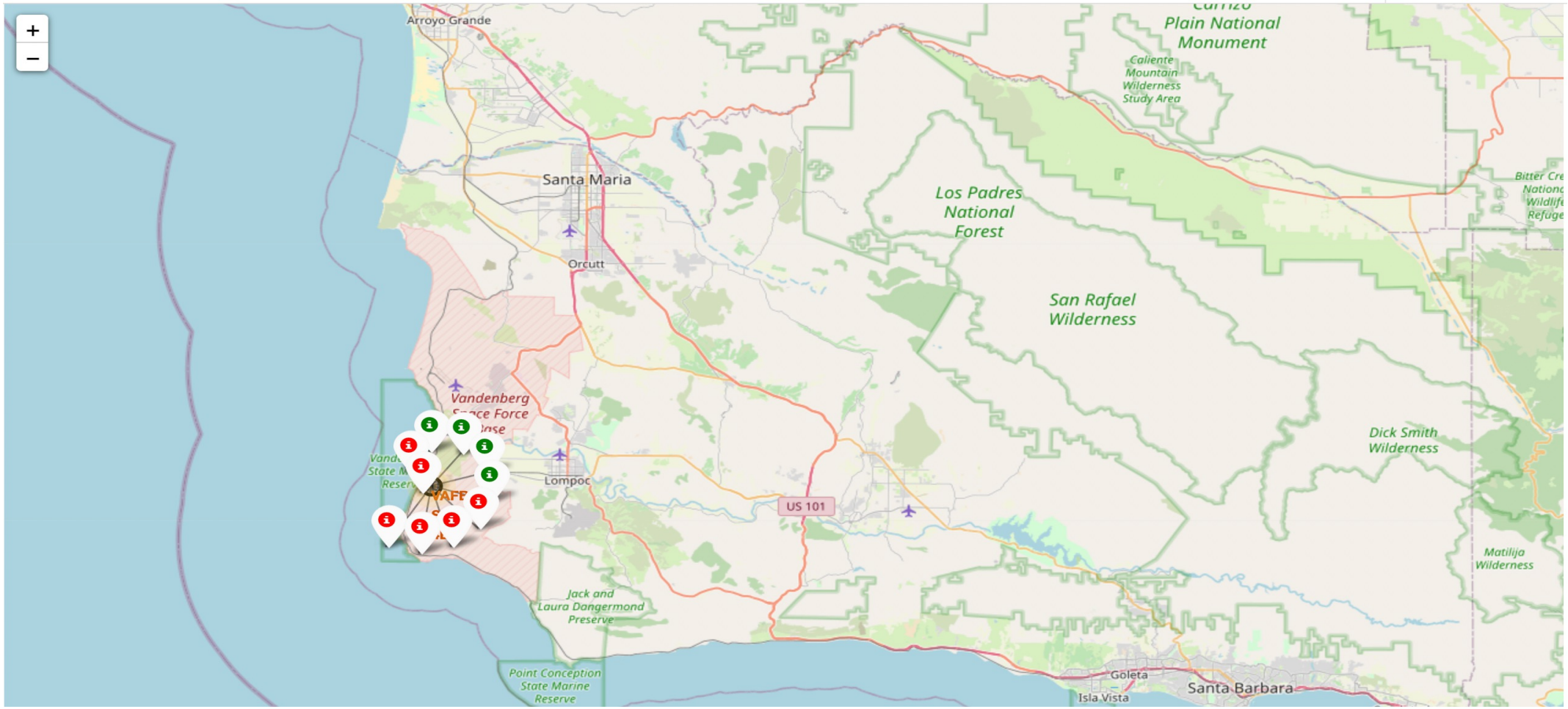
landing__outcome	total
No attempt	10
Failure (drone ship)	5
Success (drone ship)	5
Controlled (ocean)	3
Success (ground pad)	3
Failure (parachute)	2
Uncontrolled (ocean)	2
Precluded (drone ship)	1


Section 3
Launch Sites
Proximities Analysis

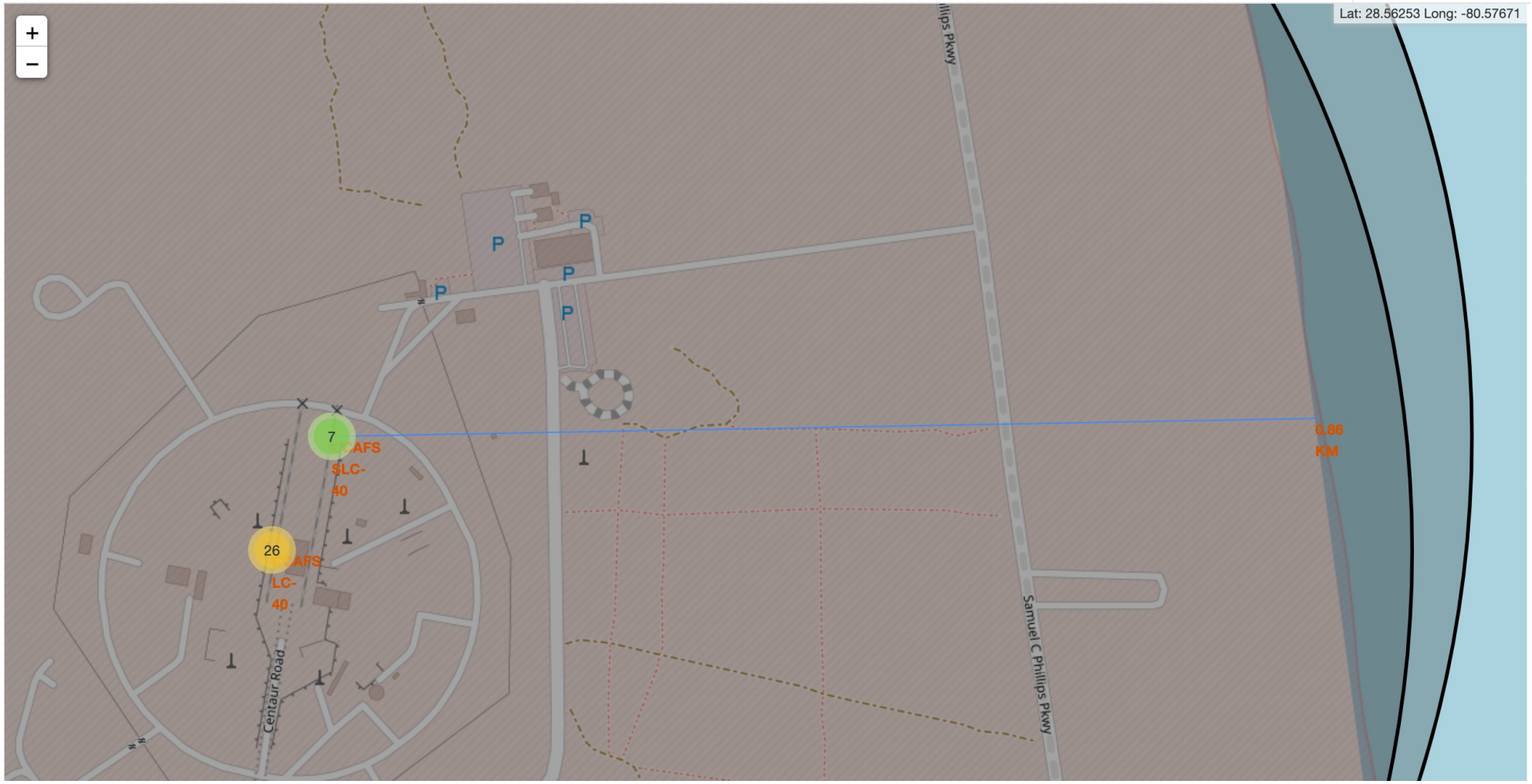
All Launch Sites with Global Map Markers



The Success/Failed launches for each site on the map



Polyline between launch site and selected coastline



Section 5

Predictive Analysis (Classification)

Classification Accuracy

Find the method performs best:

In [43]:

```
algorithms = {'KNN':knn_cv.best_score_,'Tree':tree_cv.best_score_,'LogisticRegression':logreg_cv.best_score_}
bestalgorithm = max(algorithms, key=algorithms.get)
print('Best Algorithm is',bestalgorithm,'with a score of',algorithms[bestalgorithm])
if bestalgorithm == 'Tree':
    print('Best Params is :',tree_cv.best_params_)
if bestalgorithm == 'KNN':
    print('Best Params is :',knn_cv.best_params_)
if bestalgorithm == 'LogisticRegression':
    print('Best Params is :',logreg_cv.best_params_)
```

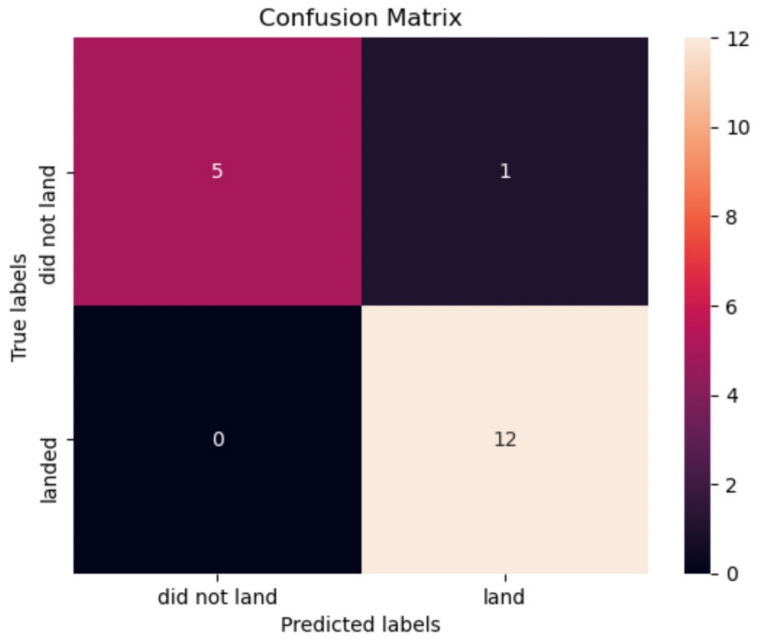
```
Best Algorithm is Tree with a score of 0.9027777777777778
Best Params is : {'criterion': 'gini', 'max_depth': 4, 'max_features': 'sqrt', 'min_samples_leaf': 2, 'min_samples_split': 10, 'splitter':
'best'}
```

The best performing model is the Decision Tree with highest classification accuracy

Confusion Matrix

We can conclude from the confusion matrix the the TRUE positive and TRUE negative rates are really high and only 1 record is misclassified as landed which was actually not landed. The accuracy for the Decision Tree model is 94.4%

```
In [37]: yhat = tree_cv.predict(X_test)
         plot_confusion_matrix(Y_test,yhat)
```



Conclusions

We can conclude that:

- The Decision tree classifier is the best machine learning algorithm for this task with an accuracy of 90%
- We observed that greater the flight number at a launch site, the greater the success rate at that launch site.
- Orbits ES-L1, GEO, HEO, SSO, VLEO have the most success rate.
- KSC LC-39A had the most successful launches as compared to other sites.

Appendix

Link to Github Repository including all notebooks:

https://github.com/twarit/AppliedDataScience/tree/master

Thank you!