Winning Space Race with Data Science

Isaac Abraham Odeh
March 7, 2022

Outline

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

Executive Summary

- Summary of methodologies
  - Data Collection through API
  - Data Collection with Web Scraping
  - 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
  - Result of Exploratory Data Analysis
  - Screenshots of Interactive Analysis
  - Result of Predictive Analysis

Introduction

- Project background and context:
  - The project fundamentally revolves on the success of the first state landing of Space X's Falcon 9. Space X advertised it's rocket at a price of 62 million dollars, which is significantly lower to its competitors whose rockets average at about 165 million dollars. The game changer presented by Space X is that it is able to reuse its components after the first stage. It is for this reason why it is vital to determine the success of the landing. If we are able to do so we will accurately be able to predict the final cost of the very launch itself. Via the creation of a machine learning model we will be able to predict the landing success.
- Problems you want to find answers:
  - What are the variables that could potentially impact the rocket's landing?
  - How do the variables/components relate to one another with respect to the landing success?
  - What are the necessary steps or requirements to be fulfilled in order to enable a successful landing?

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - The data was collected from 2 sources: Space X API & Webscraping off Wikipeida.
- Perform data wrangling
  - One Hot Encoding was applied for machine learning with the removal of unnecessary data columns.
- Perform exploratory data analysis (EDA) using visualization and SQL
  - Scatter plots and bar graph were used to get a better understanding of the relationship between the variables involved.
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - How to build, tune, evaluate classification models

Data Collection

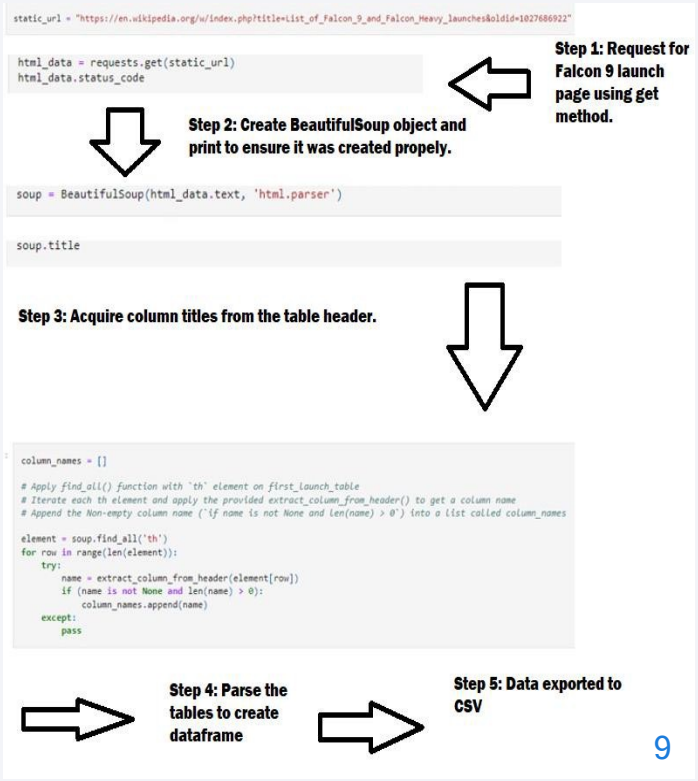
- The data sets were procured via the following methodologies:
  - The data collection was executed through the application of GetRequest to the Space X API.
  - The response content was decoded via the application of json function call and converted into a pandas dataframe.
  - The data was then cleaned and missing values were filled with the prescribed requirements.
  - Through the application of BeautifulSoup we webscraped Wikipedia to acquire Falcon 9 launch records.

Data Collection - SpaceX API

- The flow chart towards the right indicates the steps executed. We used GetRequest to acquire the data. This step was then followed up with the conversion, normalizing and cleaning up of data

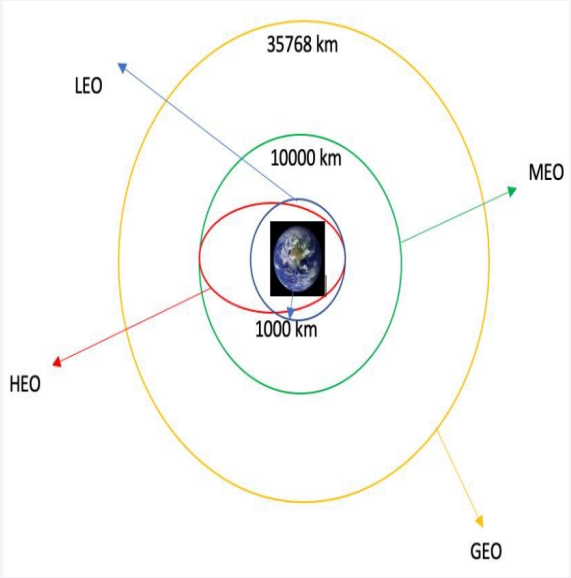
Data Collection - Scraping

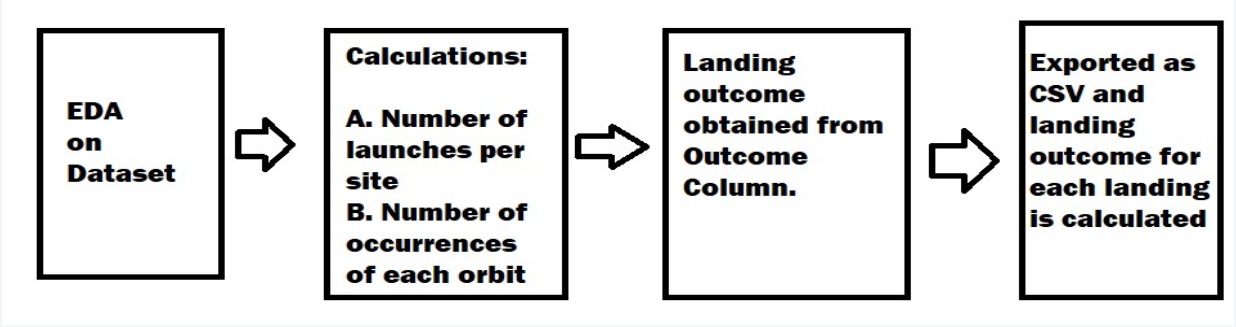
- Web Scraping was initiated by using get request and BeautifulSoup was applied. The beautiful soup object was created and the table was converted into a datafame



Data Wrangling

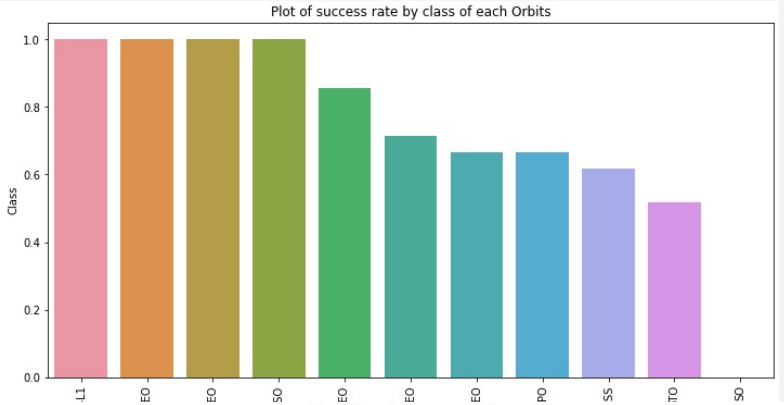
- Exploratory data analysis was conducted and the training labels were identified.
- The number of launches at each site and the number of occurrences of each orbit were calculated.
- From the outcome column the landing outcome label was acquired and the results were then exported as a CSV

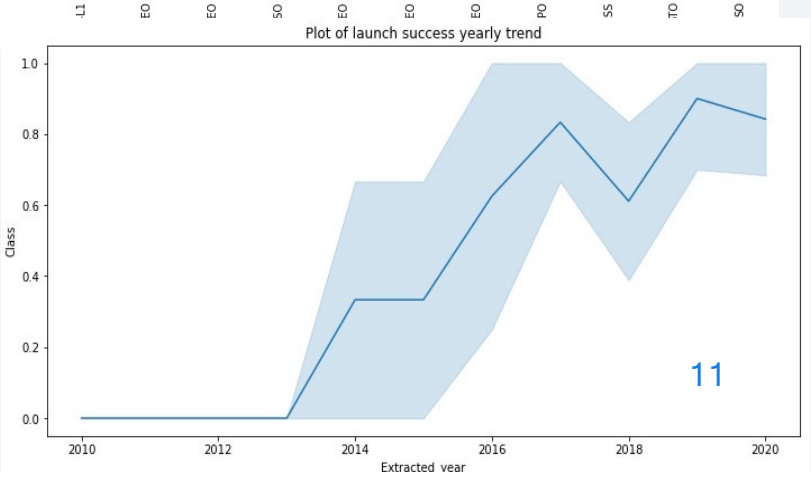




EDA with Data Visualization

- The following relationships were examined:
  - Flight number and Launch Site,
  - Payload and Launch site,
  - Success rate of each orbit type
  - Flight number & Orbit Type,
  - Launch success Annual Trend





EDA with SQL

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

Build an Interactive Map with Folium

- All launch sites were marked and map was added with objects (markers, circles, lines) to indicate either the success or failure of launches for the corresponding sites.
- Each feature launch was assigned with outcomes (failure/success. Class 0 for failure and Class1 for success.
- Launch sites with high success rates were marked with color-labeled marker clusters
- The distances between a launch site to its proximities were calculated. A few more additional aspects were looked into such as the positioning of a launch site with respect to other public areas (railways, highways etc) and the distance of launch sites from cities

Build a Dashboard with Plotly Dash

- An interactive dashboard was built with Plotly Dash.
- Pie Charts were placed to display launches on the basis of sites.
- Scatter graphs were used to display the relationship between Outcome and Payload Mass for varying booster versions

Predictive Analysis (Classification)

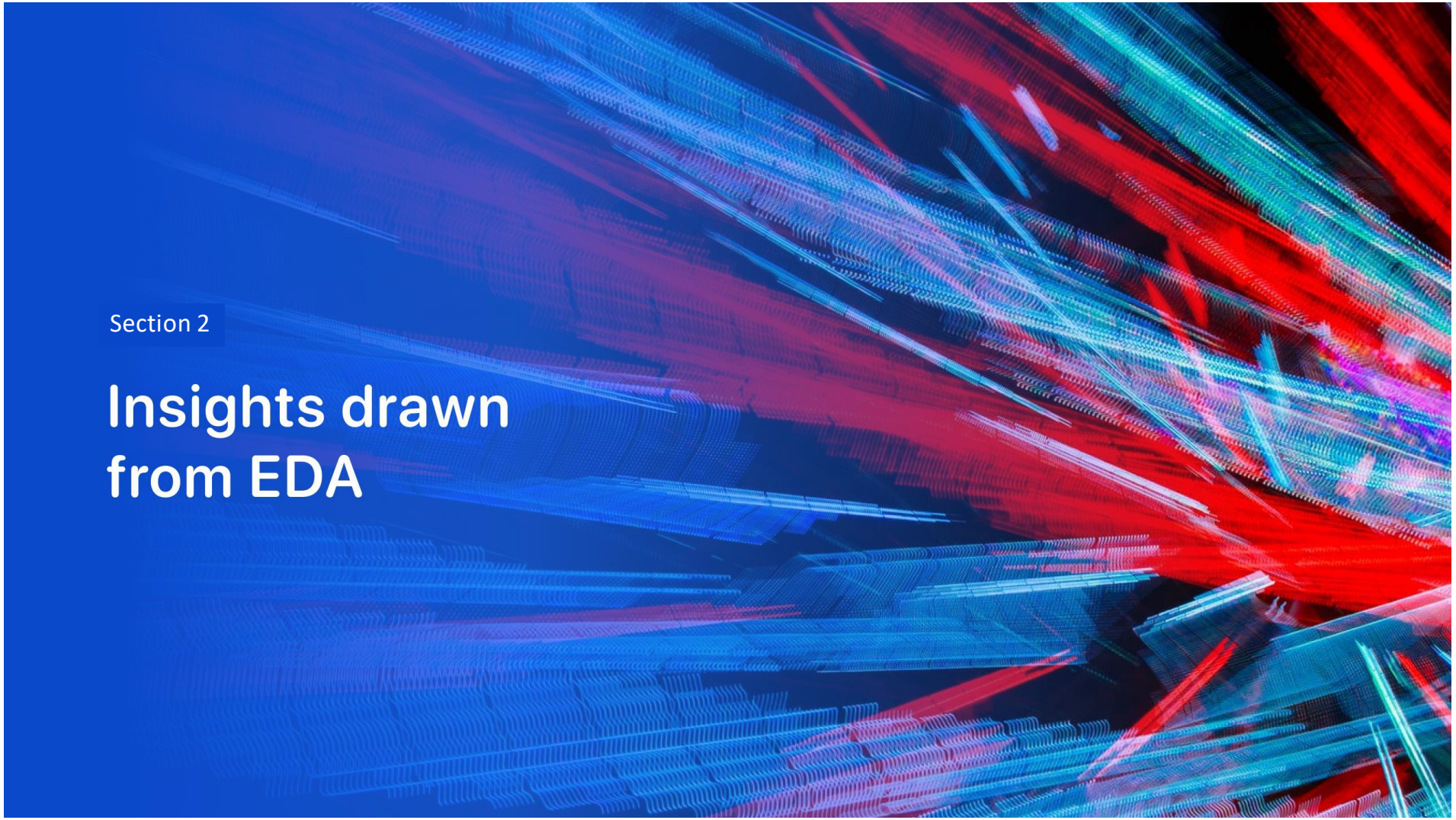
- Data was loaded using Numpy and Pandas.
- Data was split between training and testing.
- Different models were experimented with and using GridSearchCV various hyperparameters were dealt with.
- With feature engineering and algorithm tuning the accuracy was measured and improve upon.
- On the basis upon which the most optimal classification model was chosen

Results

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

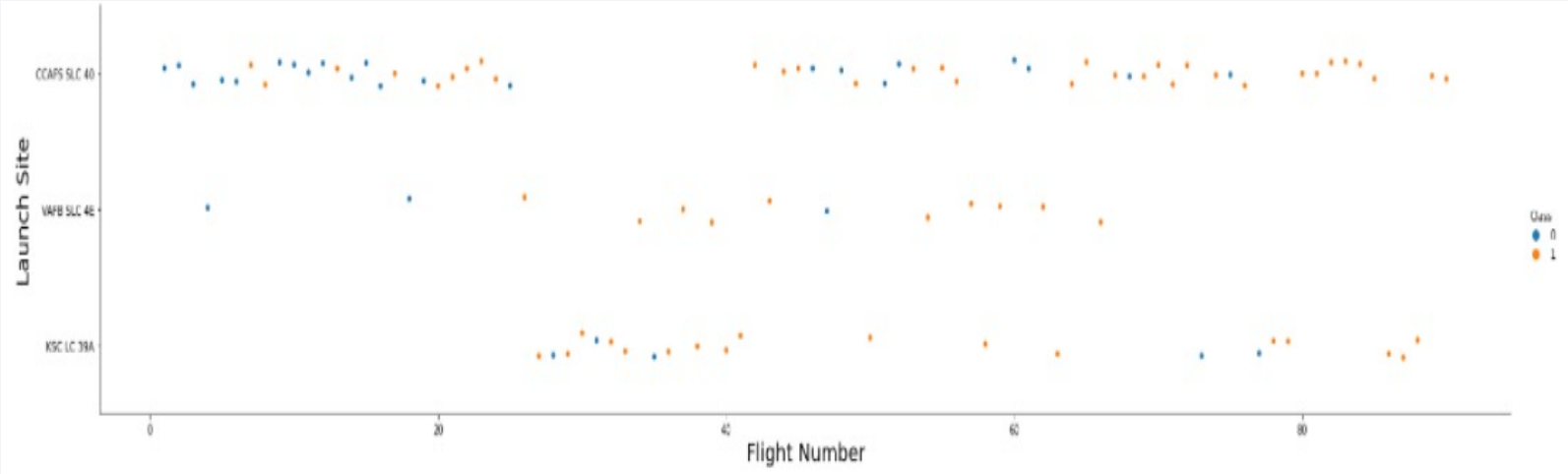
Section 2

Insights drawn from EDA



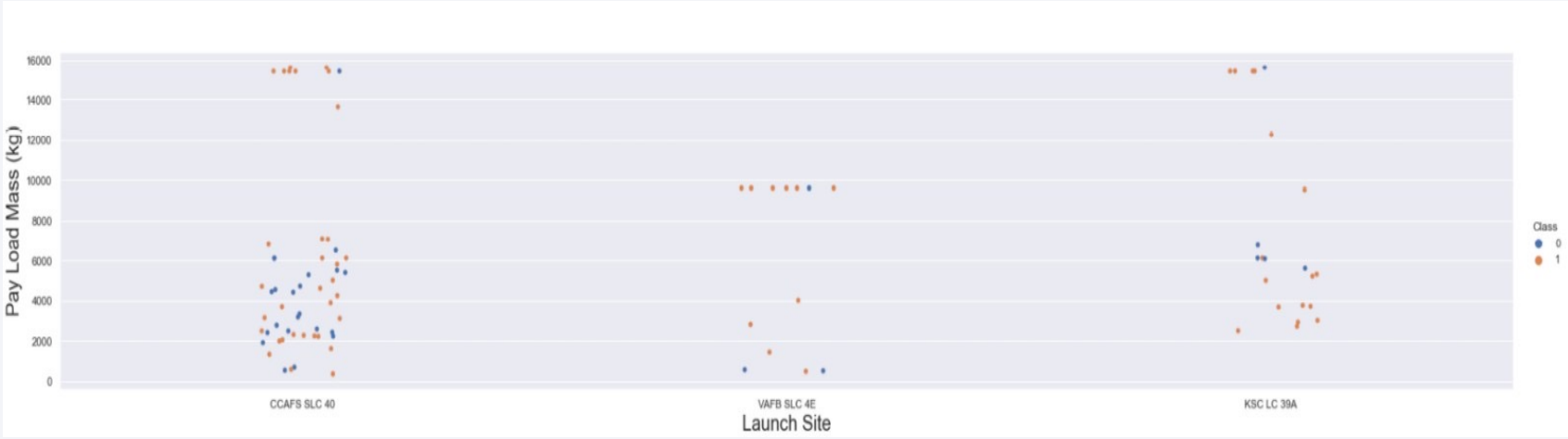
Flight Number vs. Launch Site

- The scatter plot revealed that the greater the number of flights at a particular site the greater was the success rate for the corresponding site.



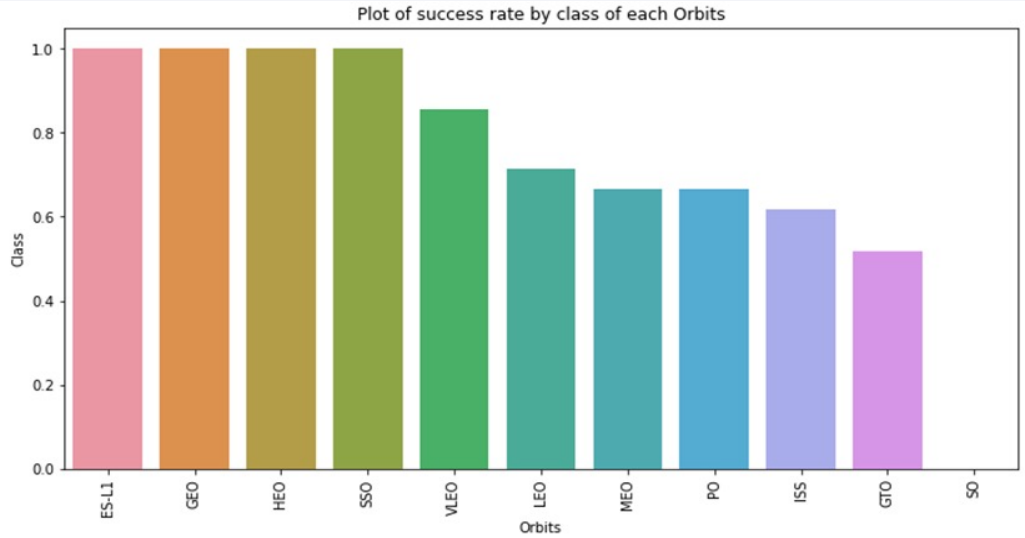
Payload vs. Launch Site

- The scatter plot revealed that the greater the payload mass for Launch Site CCAFS SLC 40 the greater was the success rate for the rocket.



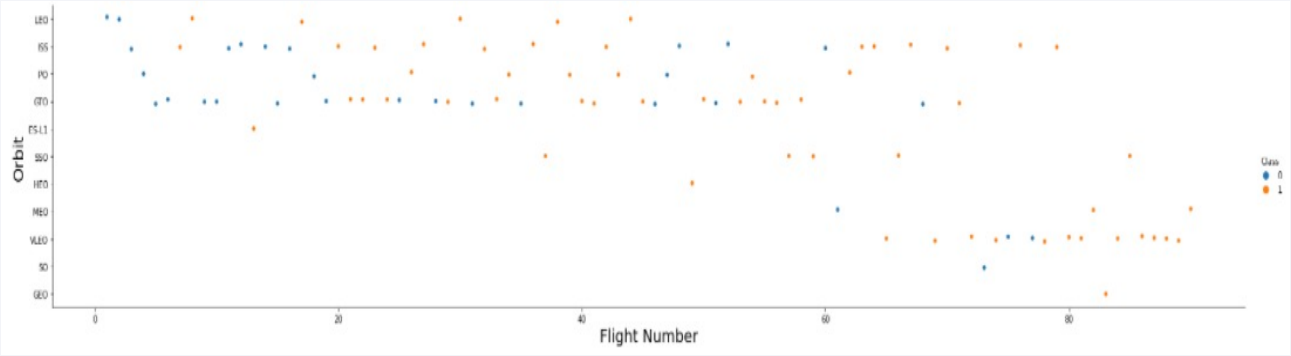
Success Rate vs. Orbit Type

- The bar chart revealed that ES-L1, GEO, HEO, SSO, VLEO had the greatest success rate.



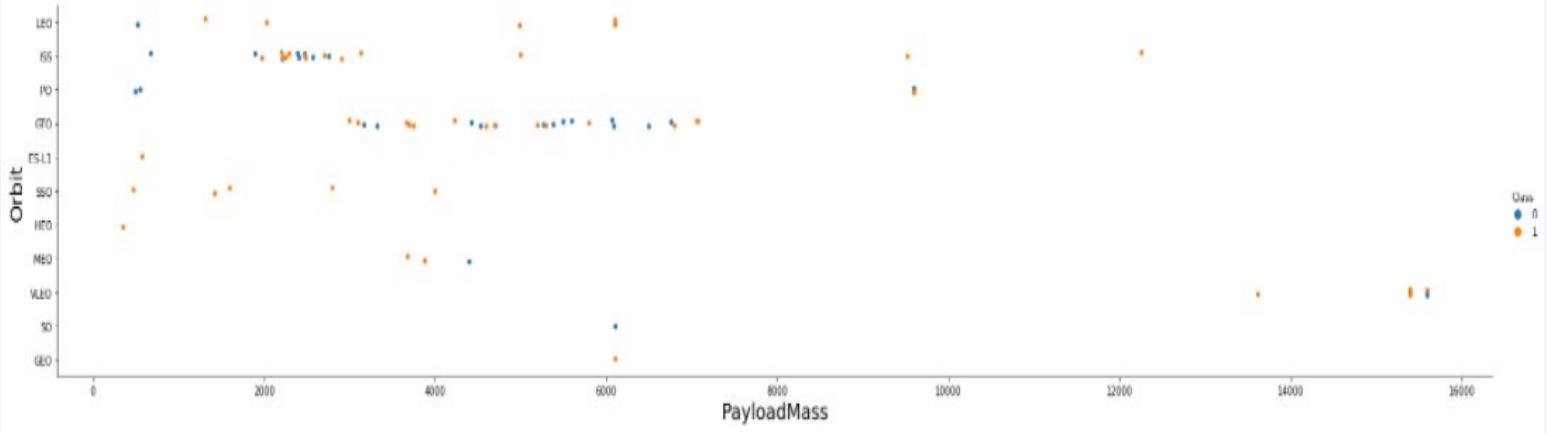
Flight Number vs. Orbit Type

- The scatter plot displayed 2 main factors. The first one being that in the LEO orbit there was a direct relationship with the number of flights. The second observation was that in the GTO orbit there was no observable relationship between the number of flights and the orbit itself.



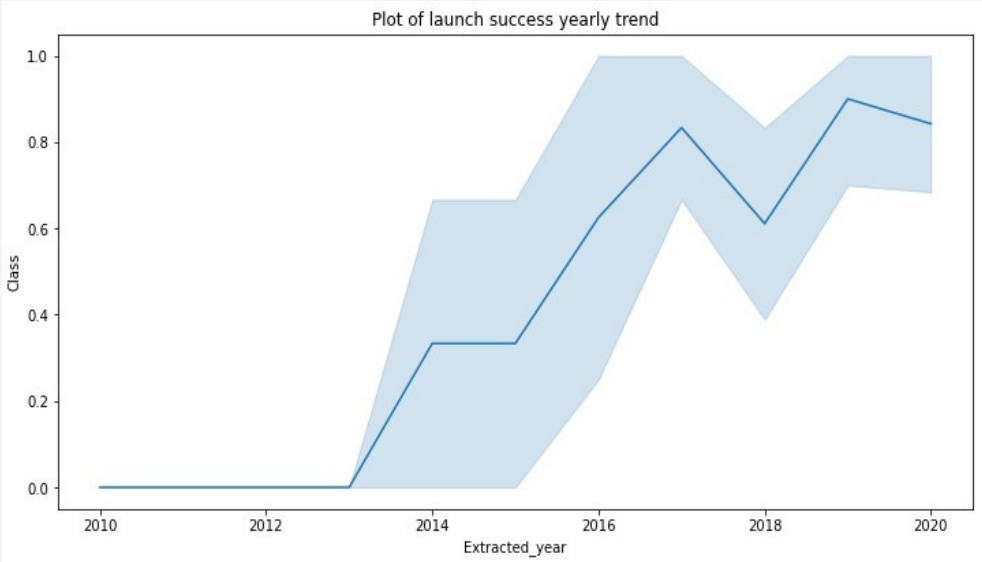
Payload vs. Orbit Type

- The scatter plot revealed that there were more successful landings for the PO, LEO and ISS orbits when heavy payloads are involved.



Launch Success Yearly Trend

- The line chart displays that the launch success improves from 2013 till 2020 with a dip in 2018.



All Launch Site Names

- DISTINCT was applied in order to identify all the unique launch sites.

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'

- The Select query along with Limit was used to find 5 records where launch sites begin with `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

Total Payload Mass

- The Select query along with Where was used to calculate the total payload carried by boosters from NASA.

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

- The average payload mass carried by booster version F9 v1.1 was identified using the following query:

In [13]:
```
task_4 = '''
        SELECT AVG(PayloadMassKG) AS Avg_PayloadMass
        FROM SpaceX
        WHERE BoosterVersion = 'F9 v1.1'
        '''
create_pandas_df(task_4, database=conn)
```

Out[13]:

	avg_payloadmass
0	2928.4

First Successful Ground Landing Date

- Through the application of the following query we were able to identify the first successful landing on the pad to be December 22, 2015.

In [14]:
```
task_5 = '''
        SELECT MIN(Date) AS FirstSuccessfull_landing_date
        FROM SpaceX
        WHERE LandingOutcome LIKE 'Success (ground pad)'
        '''
create_pandas_df(task_5, database=conn)
```

Out[14]:

	firstsuccessfull_landing_date
0	2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

- In order to filter for boosters that have successfully landed on the drone ship with payload mass between 4000 and 6000 the WHERE and AND clauses were applied in the following manner:

In [15]:

```
task_6 = '''
        SELECT BoosterVersion
        FROM SpaceX
        WHERE LandingOutcome = 'Success (drone ship)'
            AND PayloadMassKG > 4000
            AND PayloadMassKG < 6000
        '''
create_pandas_df(task_6, database=conn)
```

Out[15]:

	boosterversion
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

- We used the wildcard (%) along with subqueries in order to identify if a Mission Outcome was a success or a failure.

In [16]:

```
task_7a = '''
        SELECT COUNT(MissionOutcome) AS SuccessOutcome
        FROM SpaceX
        WHERE MissionOutcome LIKE 'Success%'
        '''

task_7b = '''
        SELECT COUNT(MissionOutcome) AS FailureOutcome
        FROM SpaceX
        WHERE MissionOutcome LIKE 'Failure%'
        '''
print('The total number of successful mission outcome is:')
display(create_pandas_df(task_7a, database=conn))
print()
print('The total number of failed mission outcome is:')
create_pandas_df(task_7b, database=conn)
```

Out[16]:

```
The total number of successful mission outcome is:
```

	successoutcome
0	100

```
The total number of failed mission outcome is:
```

	failureoutcome
0	1

Boosters Carried Maximum Payload

- Using subquery in the WHERE function along with MAX() we identified the boosters that carried maximum payload.

In [17]:
```
task_8 = '''
        SELECT BoosterVersion, PayloadMassKG
        FROM SpaceX
        WHERE PayloadMassKG = (
                                SELECT MAX(PayloadMassKG)
                                FROM SpaceX
                                )
        ORDER BY BoosterVersion
        '''
create_pandas_df(task_8, database=conn)
```

Out[17]:

	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

- Through the usage of WHERE, LIKE, AND & BETWEEN clauses along with conditions the landing outcomes in drone ship, their booster versions, and launch site names for year 2015 were filtered.

In [18]:
```
task_9 = '''
        SELECT BoosterVersion, LaunchSite, LandingOutcome
        FROM SpaceX
        WHERE LandingOutcome LIKE 'Failure (drone ship)'
            AND Date BETWEEN '2015-01-01' AND '2015-12-31'
        '''
create_pandas_df(task_9, database=conn)
```

Out[18]:

	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

- The Landing outcomes and their corresponding COUNT were selected.
- The WHERE and BETWEEN clauses were to filter for landings between 2010-06-04 to 2010-03-20.
- The landing outcomes were grouped using GROUP BY and were order in descending fashion using ORDER BY.

In [19]:

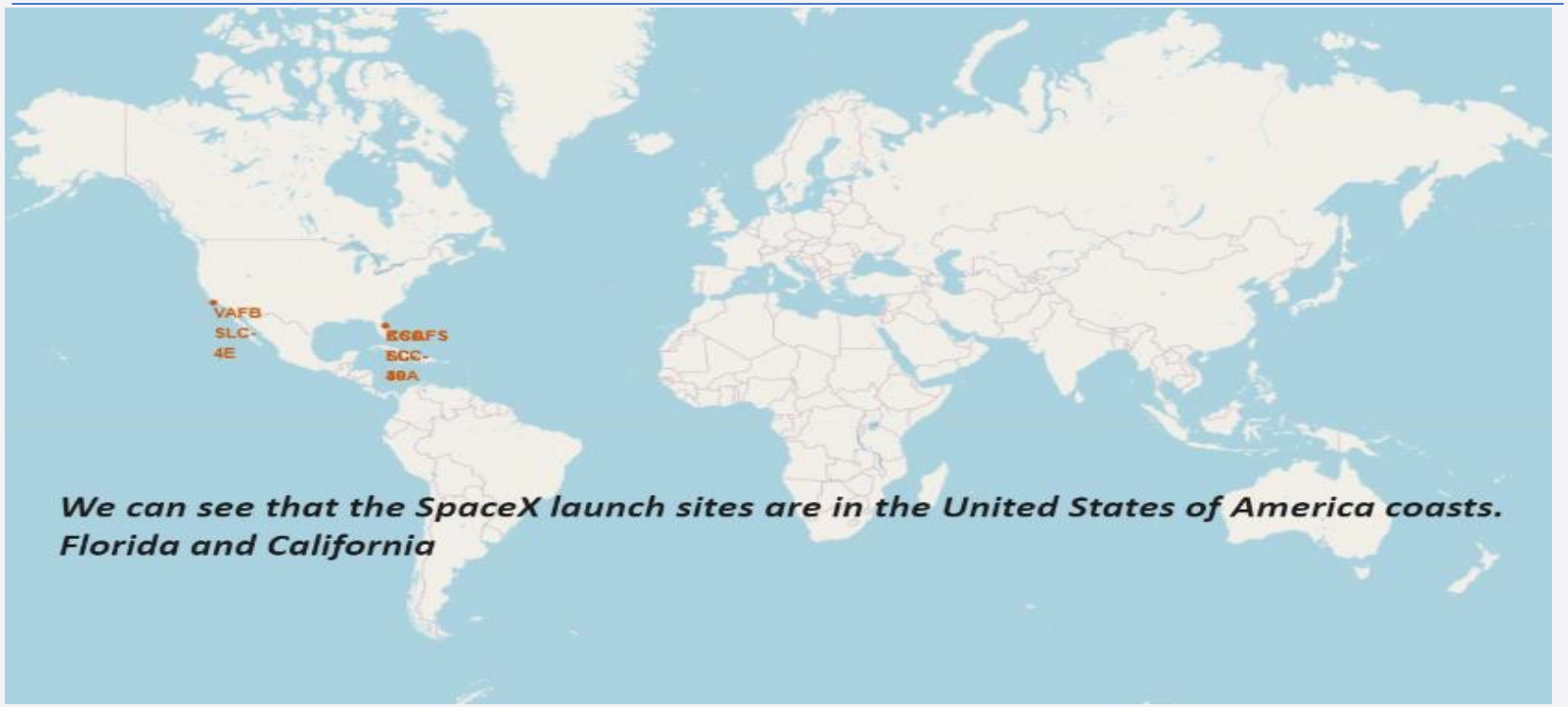
```
task_10 = '''
        SELECT LandingOutcome, COUNT(LandingOutcome)
        FROM SpaceX
        WHERE DATE BETWEEN '2010-06-04' AND '2017-03-20'
        GROUP BY LandingOutcome
        ORDER BY COUNT(LandingOutcome) DESC
        '''
create_pandas_df(task_10, database=conn)
```

Out[19]:

	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

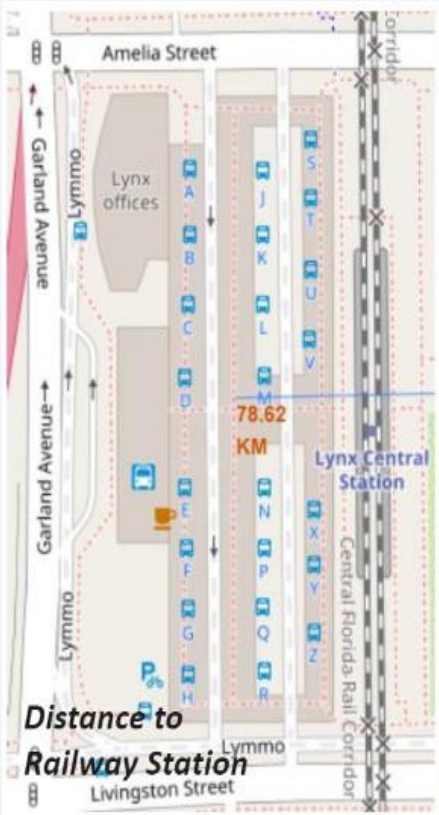

Section 3
Launch Sites
Proximities Analysis

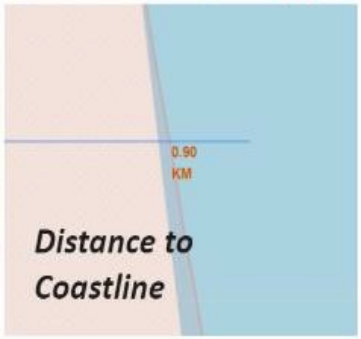
Launch Sites Global Map Markers



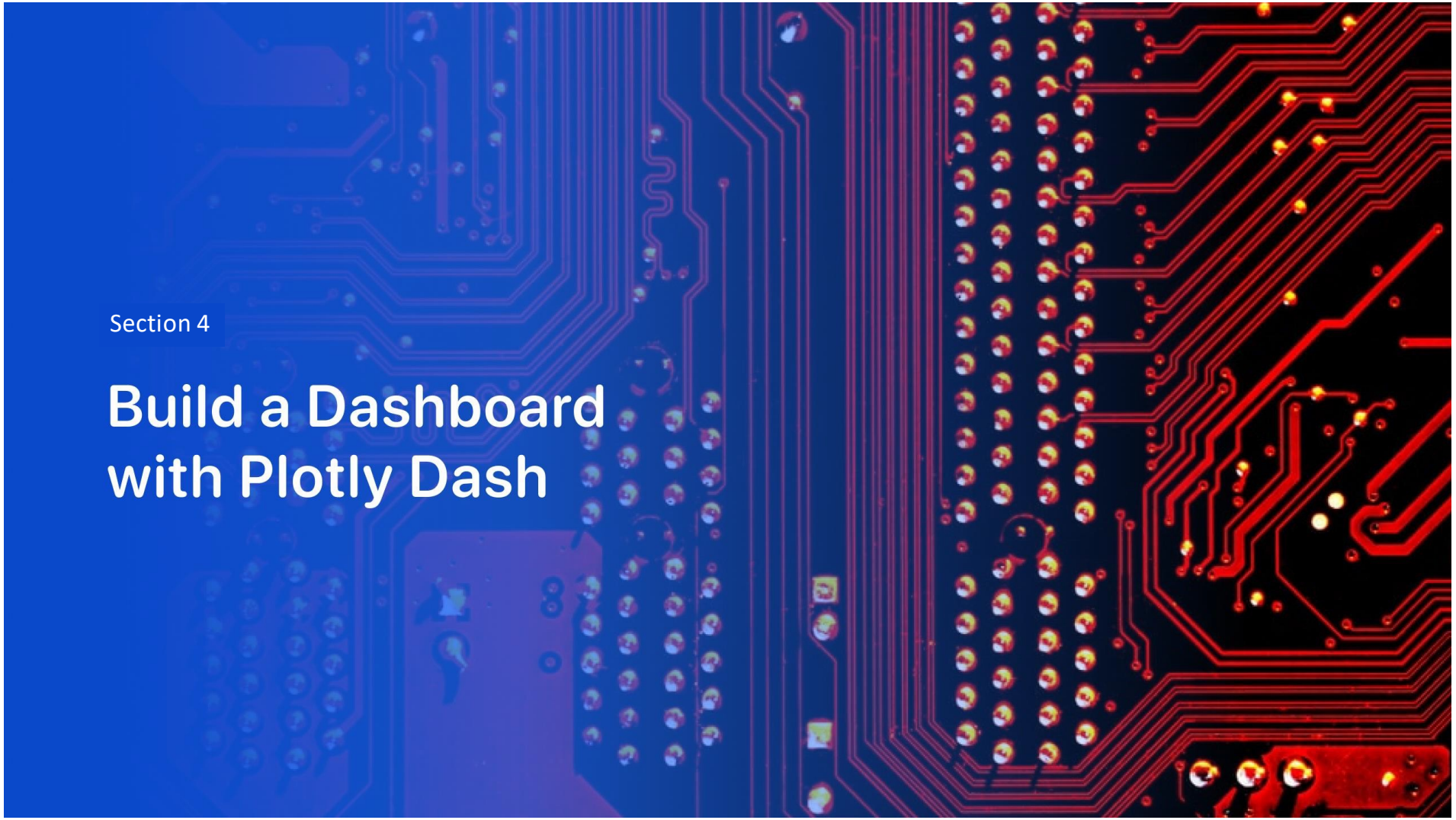
Color Labelled Markers

<Folium Map Screenshot 3>



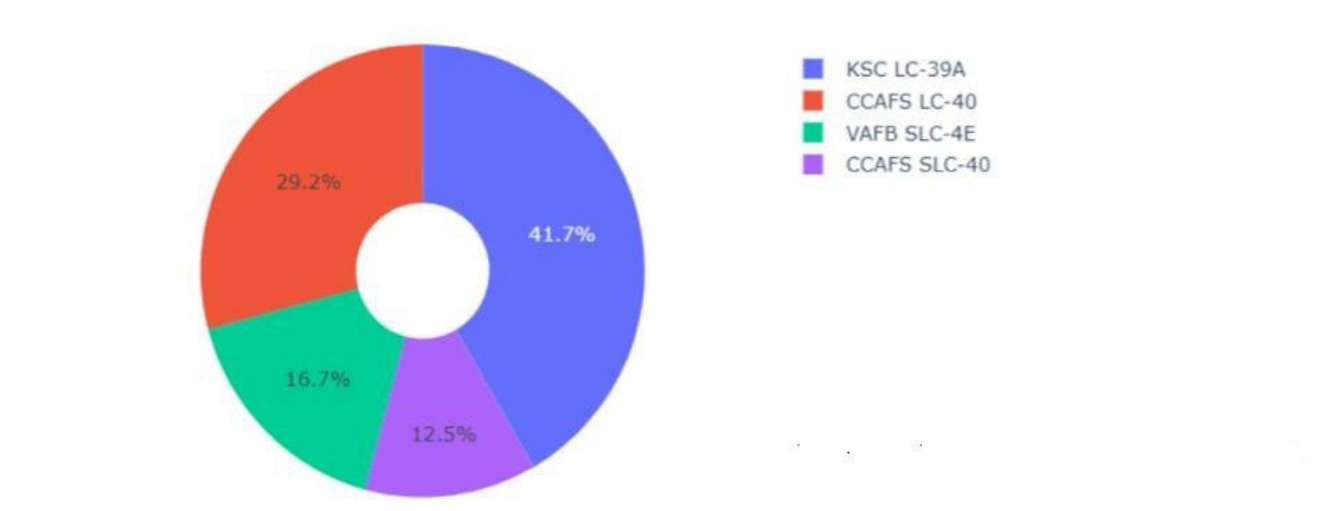


- Are launch sites in close proximity to railways? No
- Are launch sites in close proximity to highways? No
- Are launch sites in close proximity to coastline? Yes
- Do launch sites keep certain distance away from cities? Yes


Section 4
Build a Dashboard with Plotly Dash

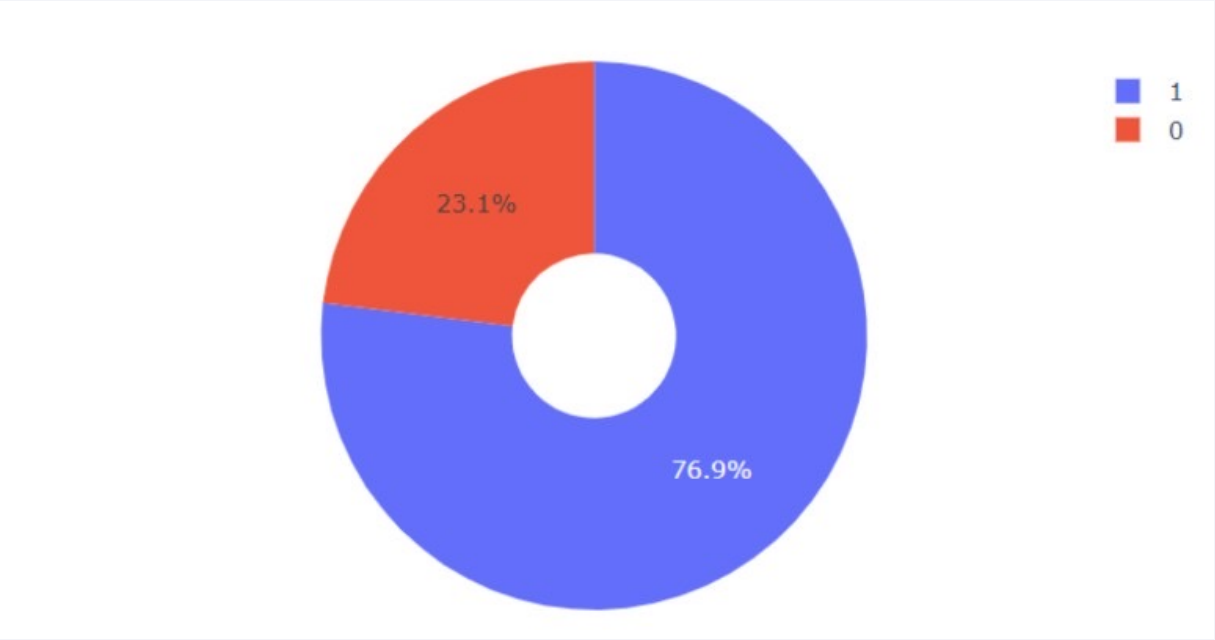
Pie Chart Display of Success Rate of each Launch Site

- The pie chart revealed that KSC LC-39A had the greatest success rate in comparison to other launch sites.



Pie Chart Display of Launch Site with highest launch success rate

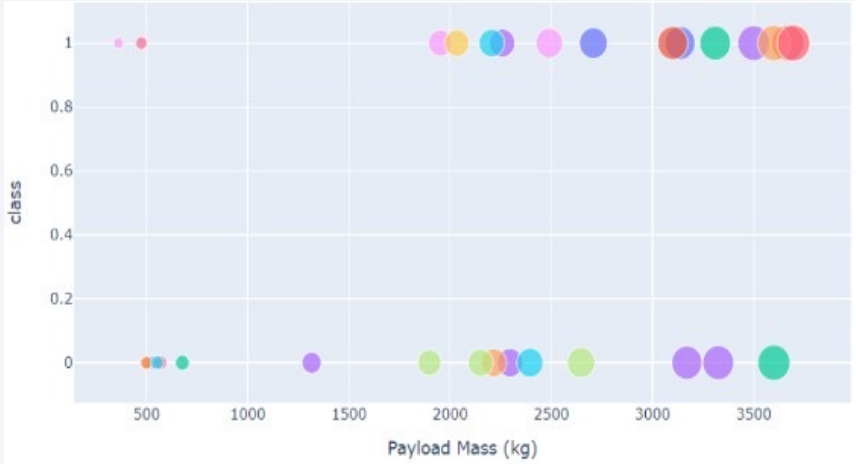
- The pie chart revealed that KSC LC-39A secured a success rate of 76.9% and a failure rate of 23.1%



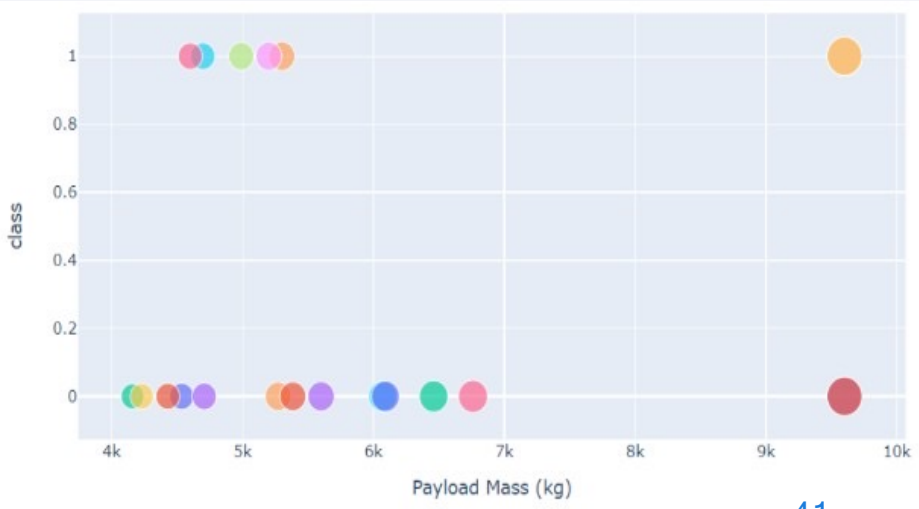
Scatter plot of Payload vs Launch Outcome for all sites along with different payload ranges.

- The scatter plots revealed that there is a greater success rate when the payload is lighter.

Payload (0-4000KG)

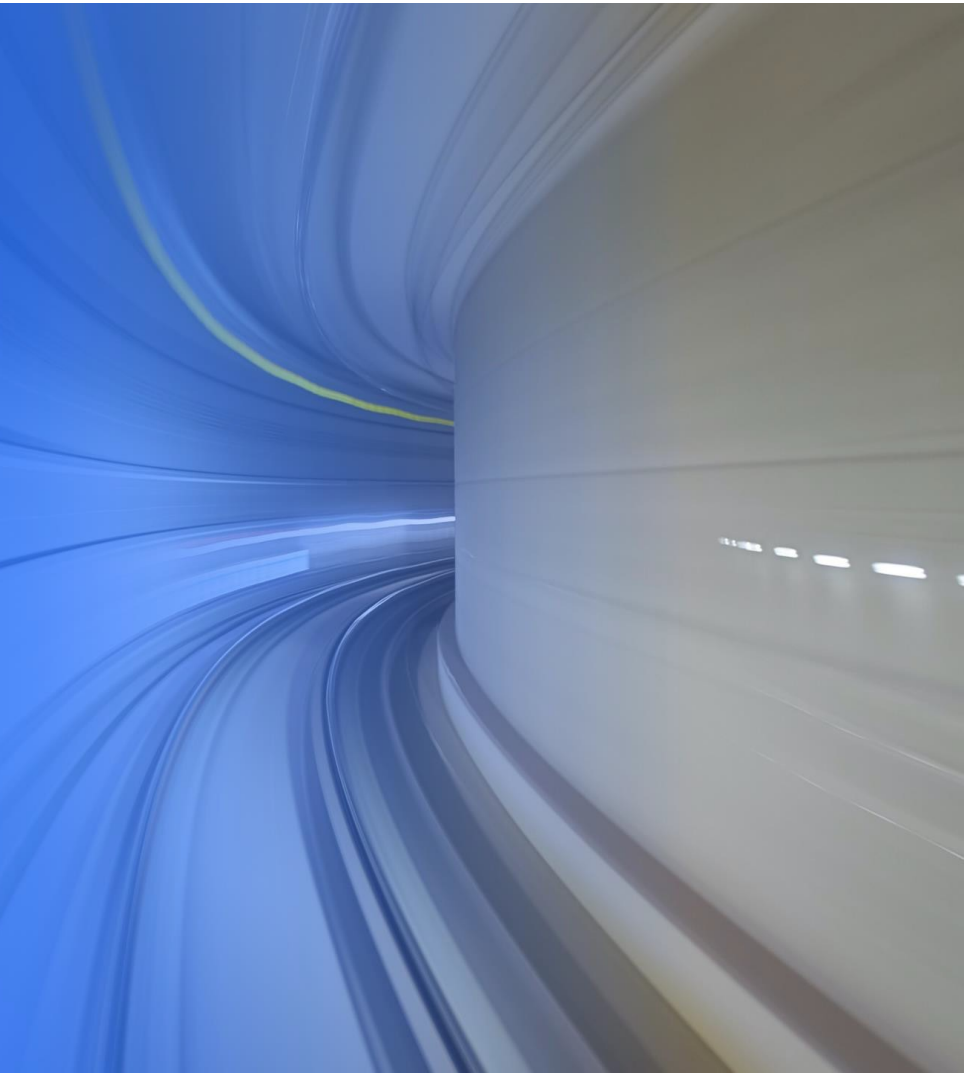


Payload (4000-10000KG)



Section 5

Predictive Analysis (Classification)



Classification Accuracy

- The Decision Tree classifier can be concluded as the model with the highest classification accuracy (as indicated below).

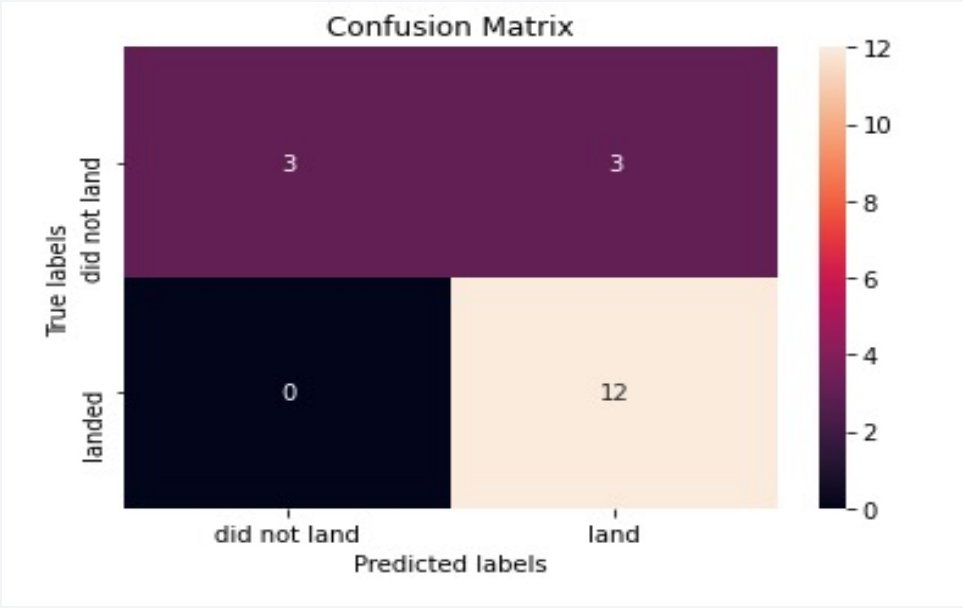
```
models = {'KNeighbors':knn_cv.best_score_,
              'DecisionTree':tree_cv.best_score_,
              'LogisticRegression':logreg_cv.best_score_,
              'SupportVector': svm_cv.best_score_}

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'}
```

Confusion Matrix

- The confusion matrix displays that the decision tree classifier is able to differentiate amongst the various classes. The prime issue lies with the presence of false positives indicated by unsuccessful landings being displayed as successful.



Conclusions

Through the analysis performed for this project the following can be concluded:

- Low weight payloads are much more optimal than heavy weight payloads.
- There is a positive relationship between time and success rate of launches thus indicating further success as time proceeds.
- KSC LC-39 is the most successful launch site in comparison to the other launch sites.
- The orbits GEO, HEO, SSO and ES-L1 had the highest success rates.
- The Decision Tree Classifier is the most optimal machine learning model to opt for with respect to this project.

Thank you!