Finding the best 'ML model' to predict the success rate of SpaceX Launch vehicle

Manoj Kumar Bahuguna

08/12/2022

OUTLINE

- Executive Summary
- Introduction
- Methodology
- Results
- Findings & Implications
- Conclusion
- Links

EXECUTIVE SUMMARY

- **Introduction (Understanding Business Problem)**
- **Data collection(Method, Sources).**
  - SpaceX Public API
  - Web scrapping of SpaceX Wikipedia page.
- **Data wrangling.**
  - Exploratory data analysis
  - Determining training label
- **EDA done with Visualization**
  - Visual analytics with Folium
  - Interactive Dashboard with Plotly
- **Predictive analysis**
  - ML model building
  - Evaluation
- **Summary of results**
  - Finding and implications
  - Conclusion

INTRODUCTION

Understanding the Business problem:-

- *Here we will trying to determine the cost of each launch through finding the success rate of launch vehicle.*
- *Therefore if we can determine 'if the first stage will land', we can determine the cost of a launch.*

Use of Data science and methodology, it includes collecting data and analyzing it with various methods, visualization and prediction.

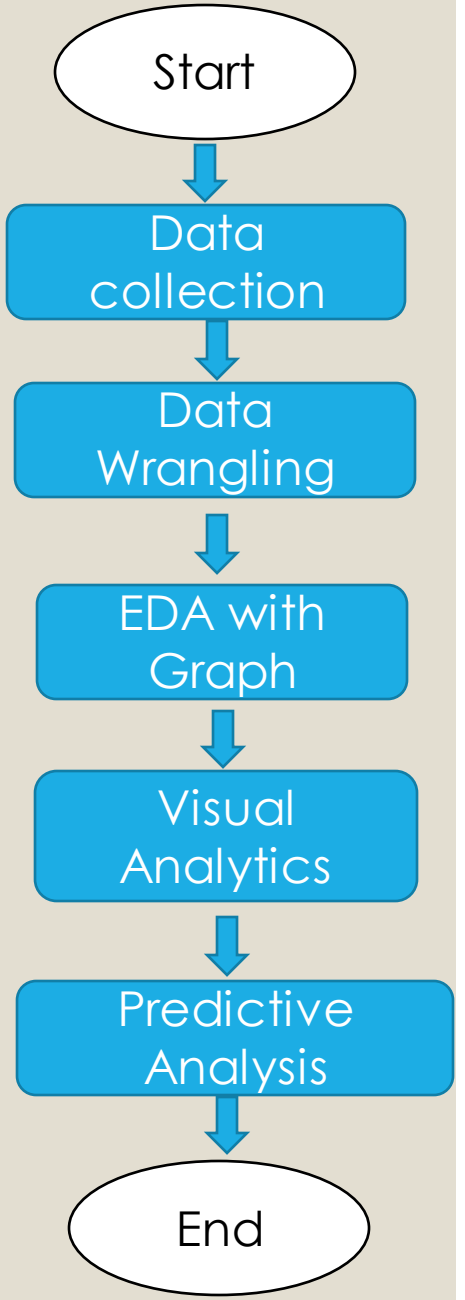
- *We will make an use of classification model for launch outcome for predicting success rate.*
- *Finding best classifying model for the purpose, Applying various ML models and scrutinizing through evaluation to check for best accuracy*

METHODOLOGY

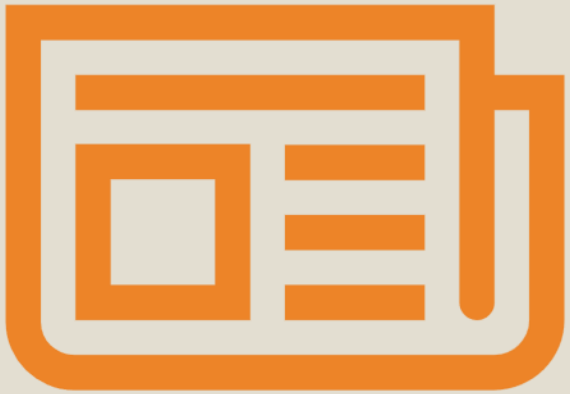
The "Methodology" followed for finding the answer for our Business problem is as follow:-

- **Data collection**
- **Data wrangling.**
- **Exploratory Data analysis (EDA) with Graph**
- **Data visualization by the help of Folium map**
- **Data visualization by the help of Interactive Dashboard**
- **Predictive analysis**

All these pointers are further described in next Slides



METHODOLOGY Data collection & Data wrangling.



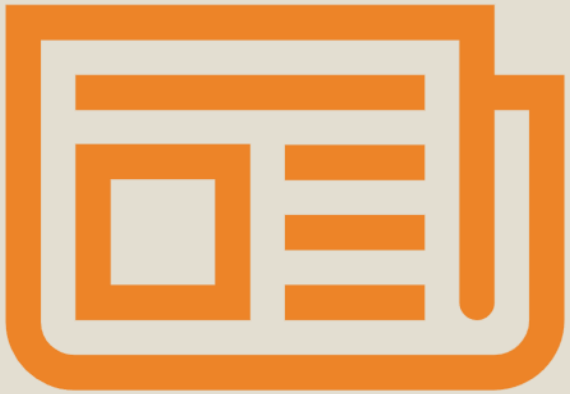
Collection of Data

- Collecting data from the SpaceX API and cleaning
- Extracting a Falcon 9 launch records HTML table from Wikipedia
- Parsing the table and converting it into a Pandas data frame

Data wrangling

- Exploratory Data Analysis (EDA) to find some patterns in the data
- Determining what would be the label for training supervised models.

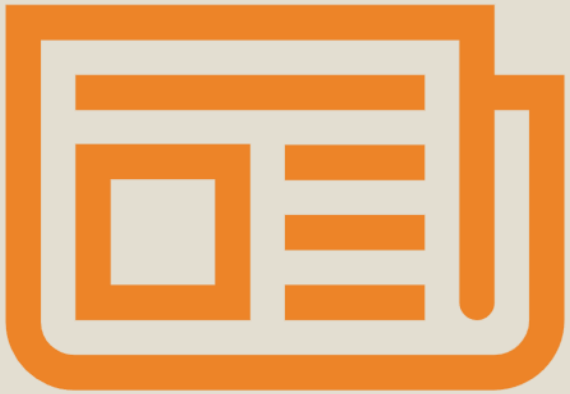
METHODOLOGY Data visualization by the help of Folium map



Launch Sites Locations Analysis with Folium

- Marking all launch sites on a map
- Marking the success/failed launches for each site on the map
- Calculating the distances between a launch site to its proximities

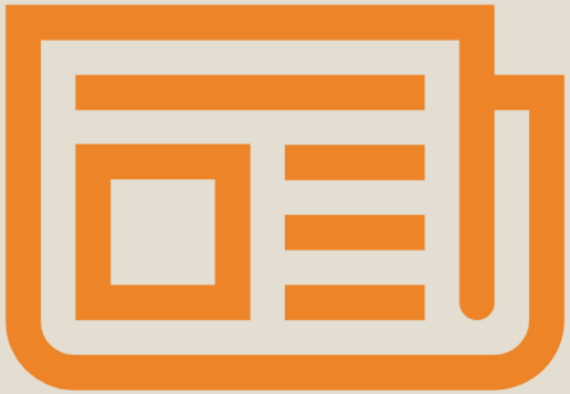
METHODOLOGY Data visualization by the help of Interactive Dashboard



Building Dashboard Application with Plotly Dash to find:

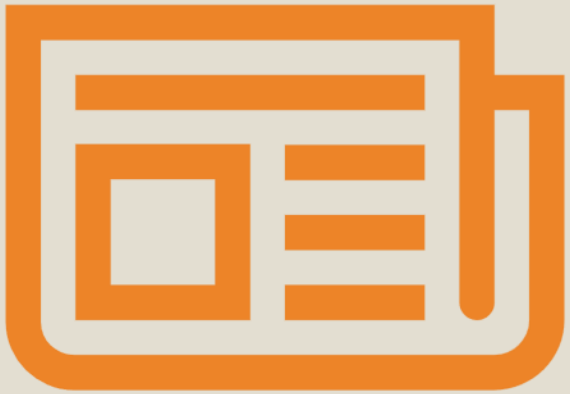
- The Sites having the largest successful launches
- The Sites having the highest launch success rate
- Payload range(s) having the highest launch success rate
- Payload range(s) having the lowest launch success rate
- The F9 Booster version having the highest launch success rate

METHODOLOGY Exploratory Data analysis (EDA)



- Understanding the Collected Spacex Dataset
- Analyzing Success and Failure w.r.t Launch sites, Payload Mass and Booster versions.
- Analyzing the relation between features and launch outcome
- Cleaning the data and handling missing values.
- Feature engineering, one hot coding and normalization of data through standard scalar.
- Use of Classification model for prediction
  - Class "0" for Failure
  - Class "1" for Success

METHODOLOGY Predictive analysis



- Model applied for predicting launch outcome
  - Logistic regression
  - Decision tree classifier
  - Support vector machine(SVM)
  - K nearest Neighbour (KNN)
- Using train_test_split to split the data into training and test data.
- Finding best Hyperparameter for SVM, Classification Trees, KNN and Logistic Regression
- Finding best accuracy on the validation Data and creating confusion Matrix.

Results

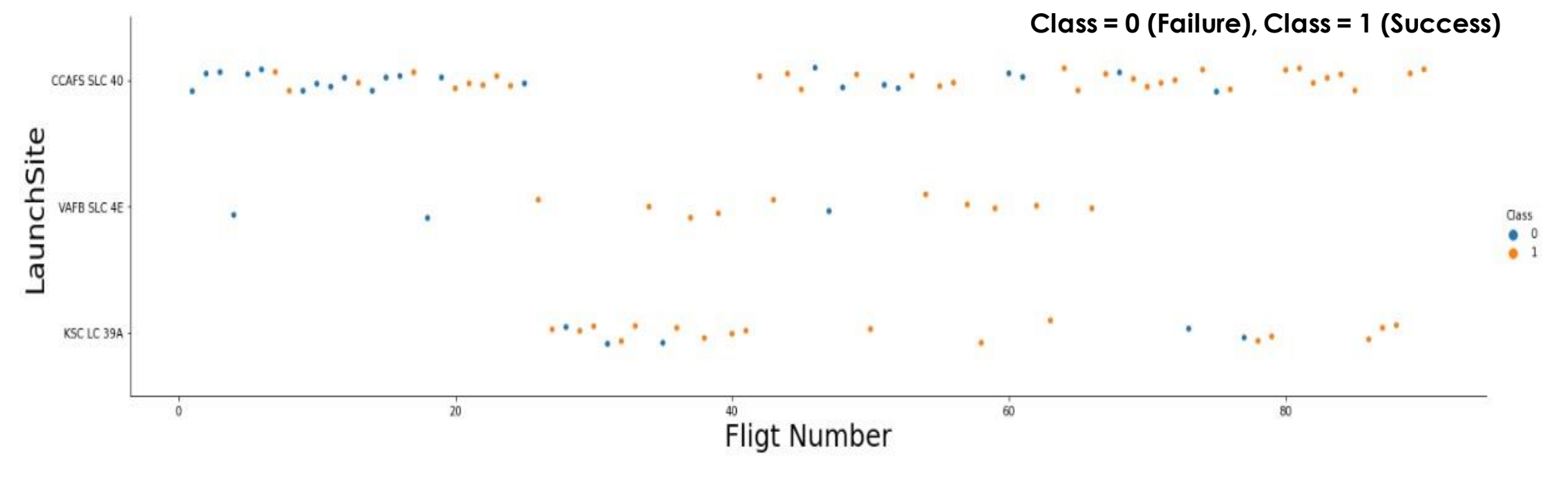
By performing Exploratory data analysis (EDA) we have certain finding on important features, variables, factors etc. in relations to the launch outcome.

Results outcome is described in the slides under following categories:-

- EDA results with Visualization.
- Interactive map with Folium results
- Plotly Dash dashboard results

EDA Results with Visualization - 1

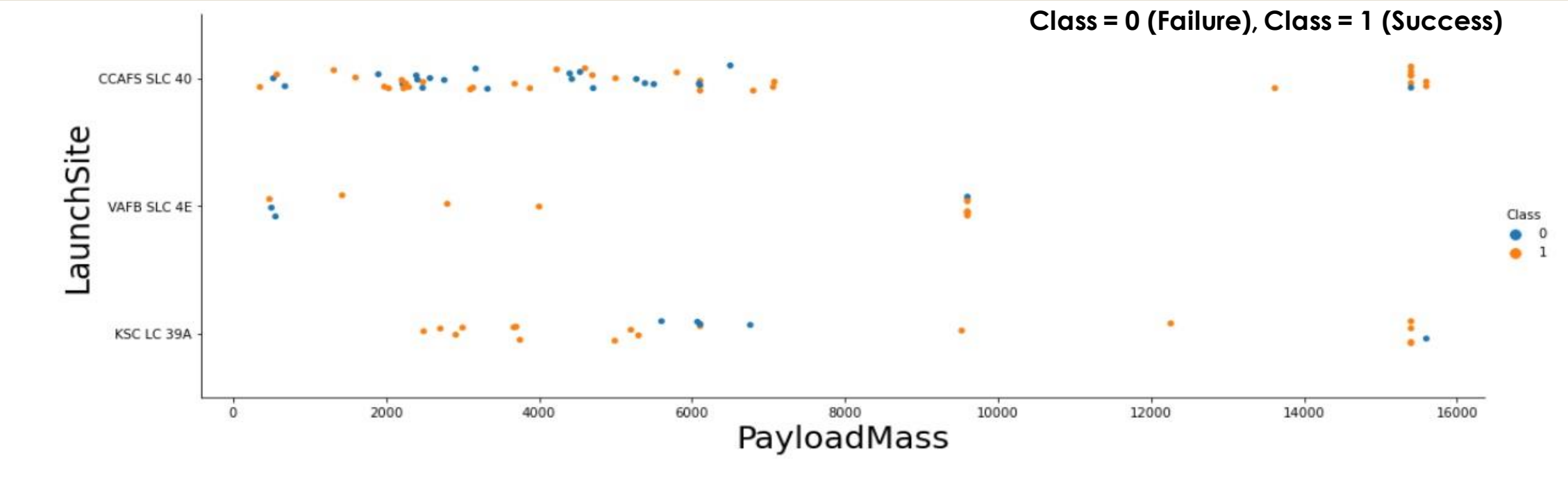
Visualizing the relationship between Flight Number and Launch Site



Through this scatter plot we can see that the success rate of launch as per Launch-site increases as the number of flights increase.

EDA Results with Visualization -2

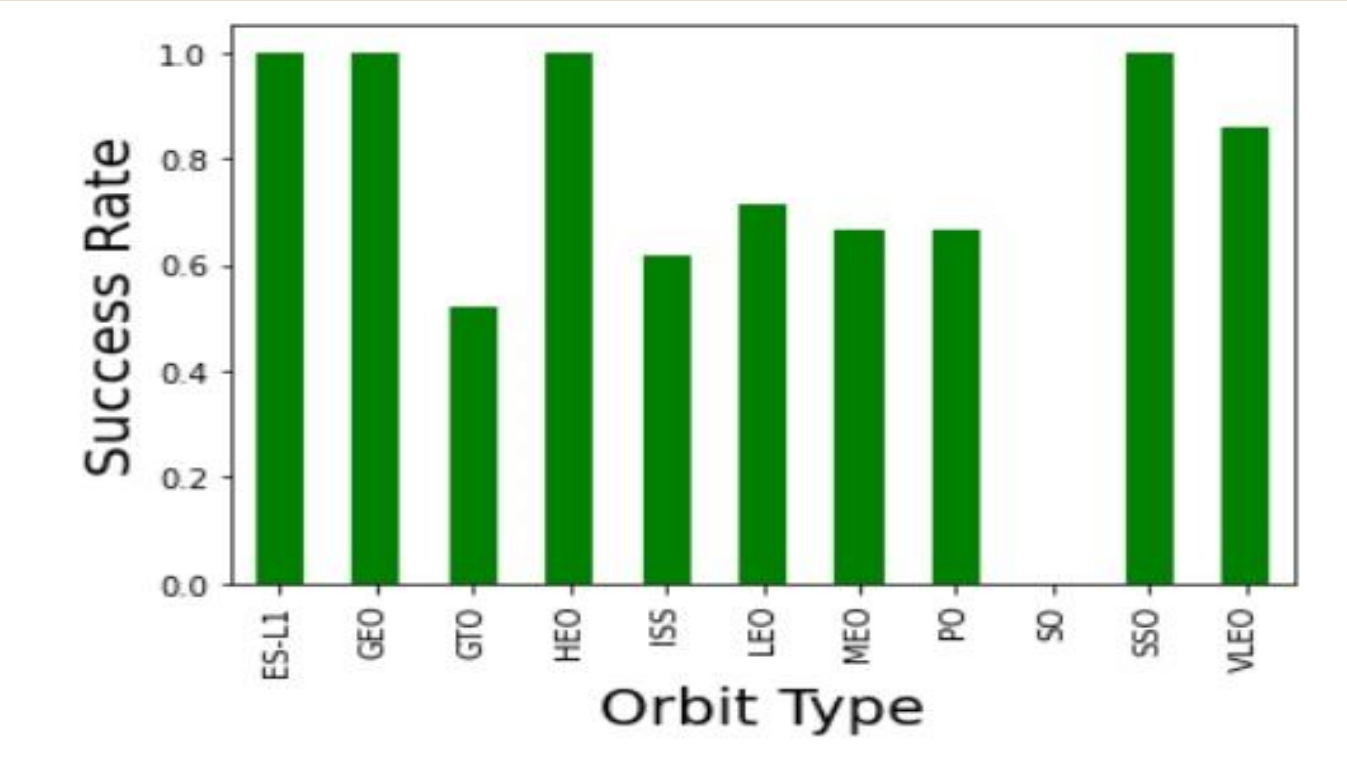
Visualizing the relationship between Payload and Launch Site



Through this scatter plot we can see that the success of launch vehicle becomes better for higher payload Mass.

EDA Results with Visualization -3

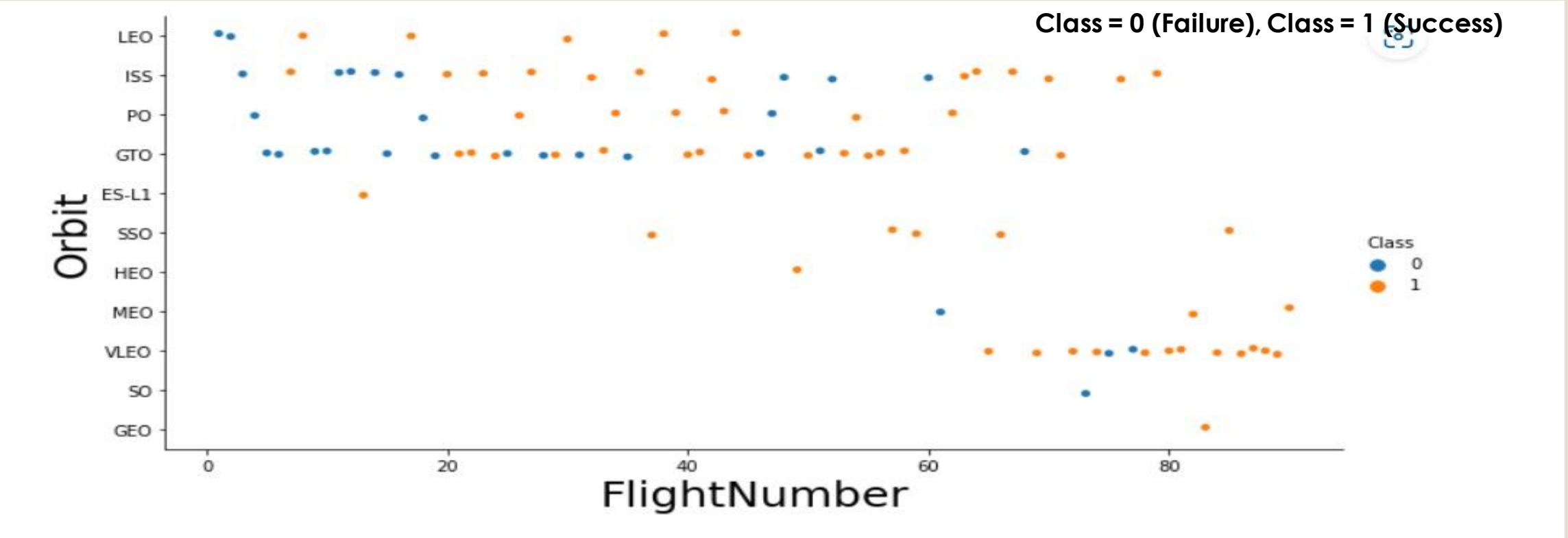
Visualizing the success rate w.r.t the orbit type



With the help of Bar Chart we can observe that the success rate for orbits ES-L1, GEO, HEO and SSO is the best.

EDA Results with Visualization -4

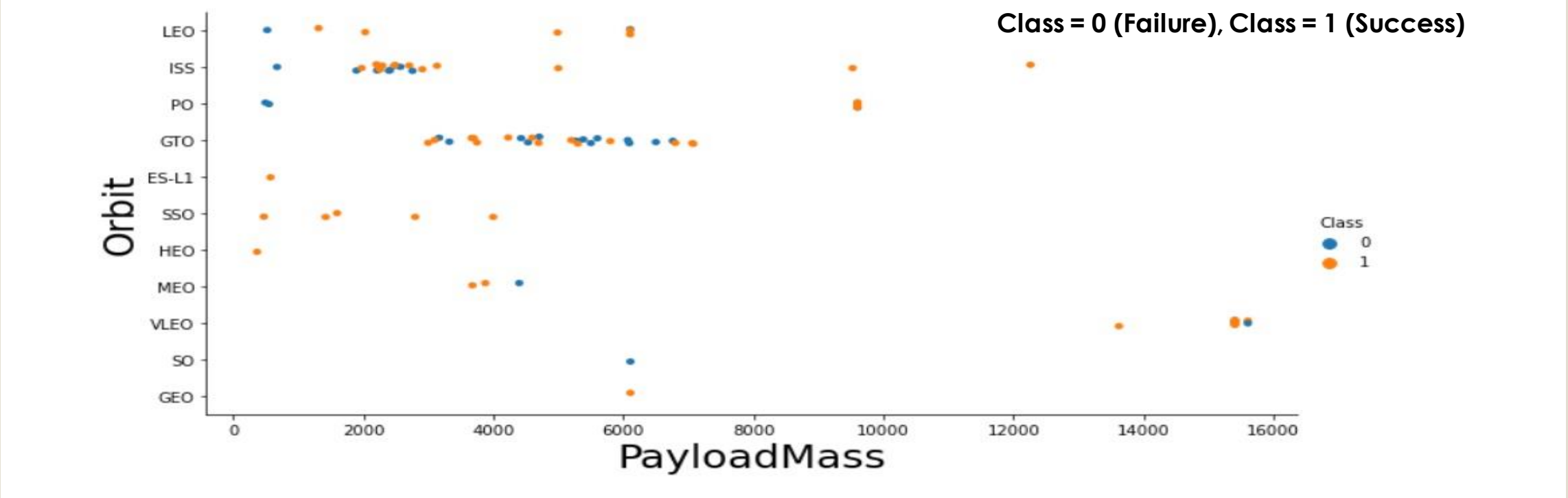
Visualizing the relationship between Flight Number and Orbit type



This scatterplot gives us mix information, for 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.

EDA Results with Visualization -5

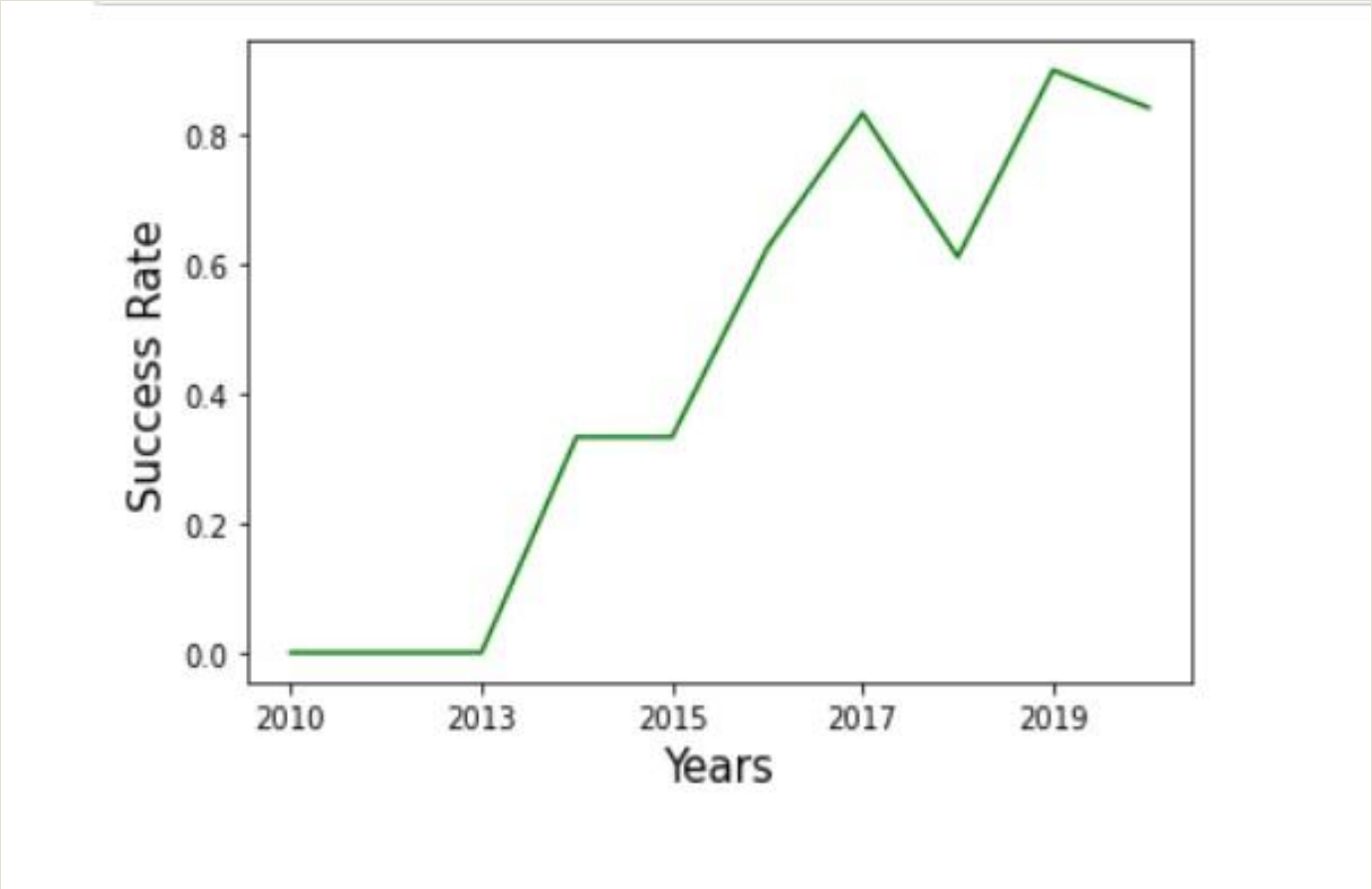
Visualizing the relationship between Payload Mass and Orbit type.



This scatter plot shows that for the heavy payloads the successful landing rates are more for Polar, LEO and ISS. But for GTO the results are not much distinguishable.

EDA Results with Visualization -6

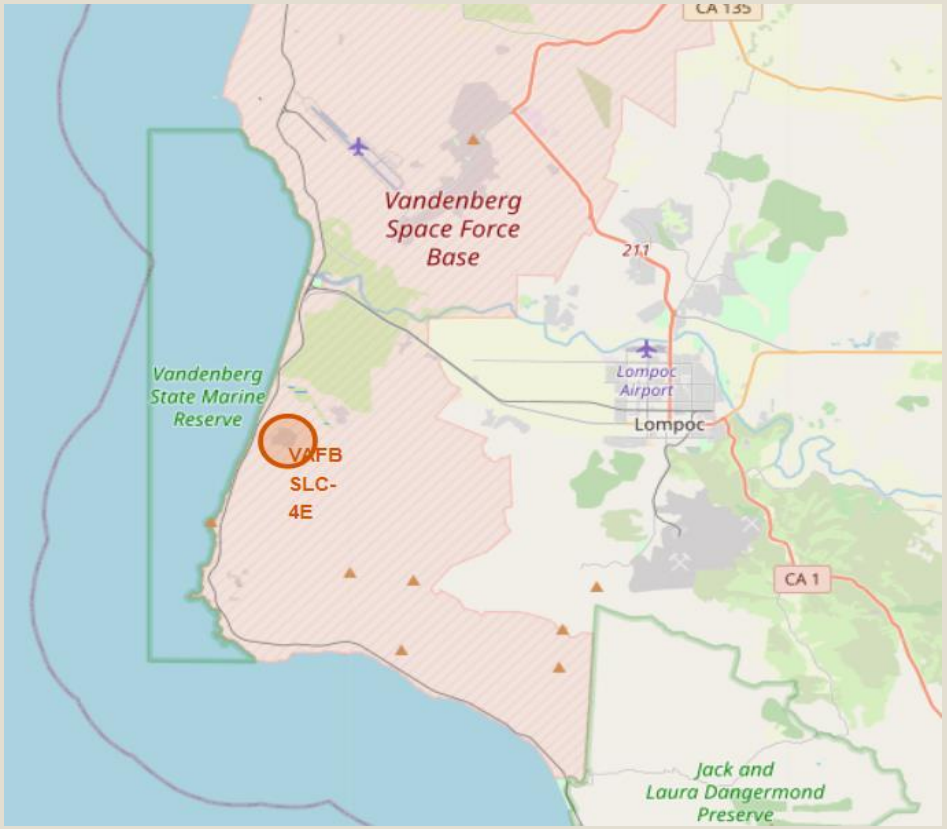
Visualizing the launch success yearly trend

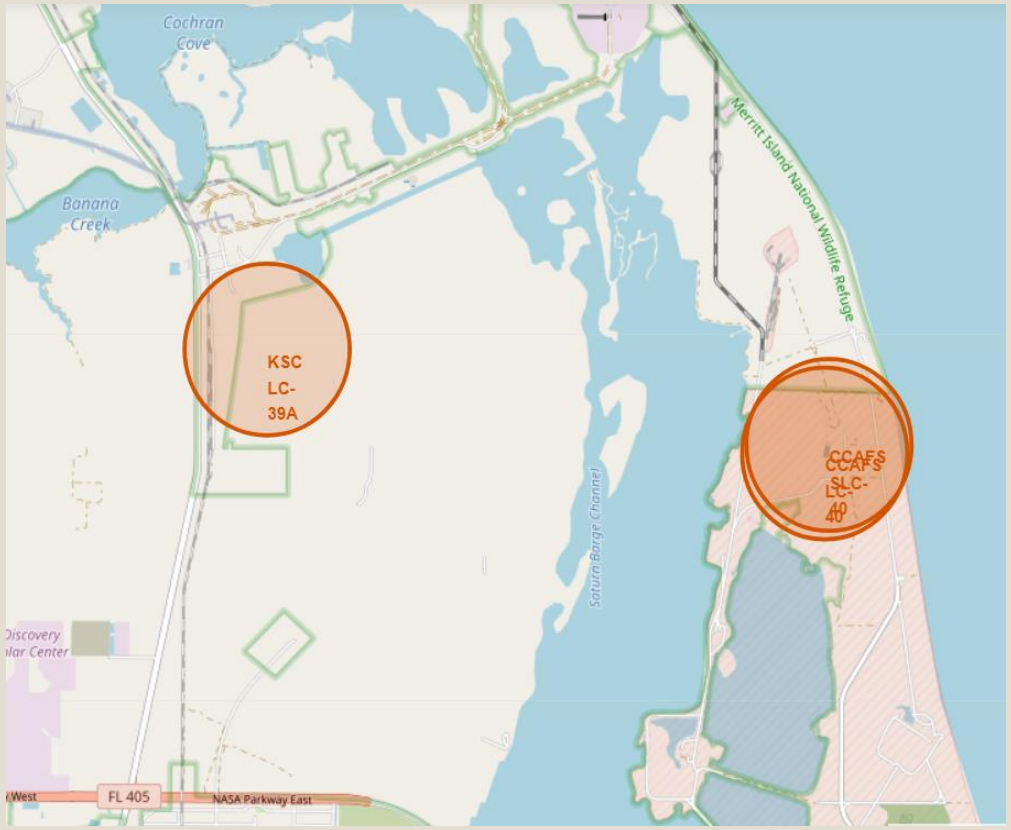


Following the yearly trend the success rate kept on increasing since 2013 to 2020

Results- Interactive map with Folium

Marking all launch sites on a map

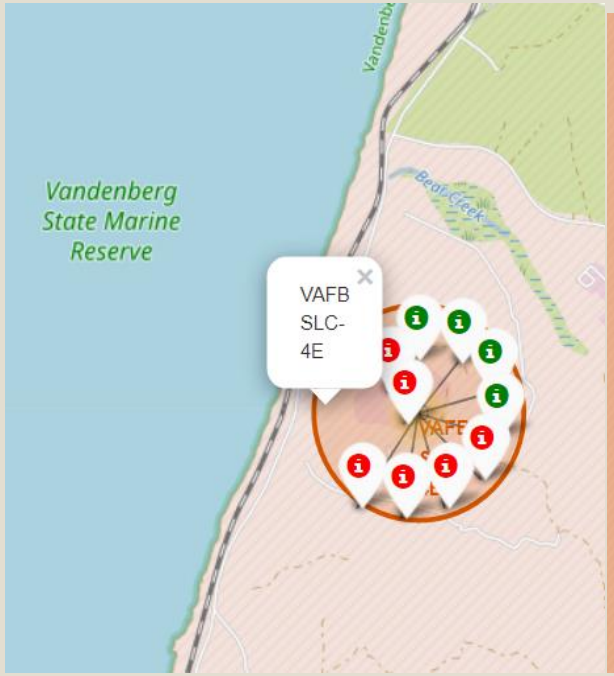


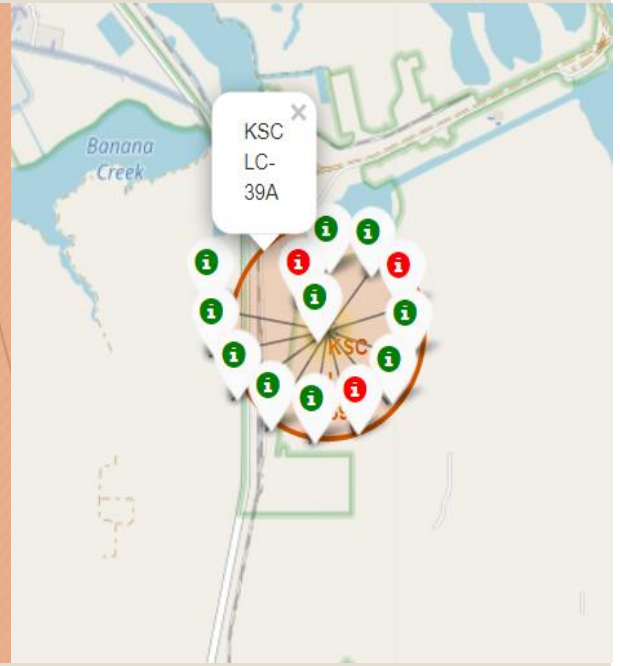


We can observer that all launch sites are in proximity to the Equator line and to the coast.

Results- Interactive map with Folium

Marking success/failed launches for each site on the map

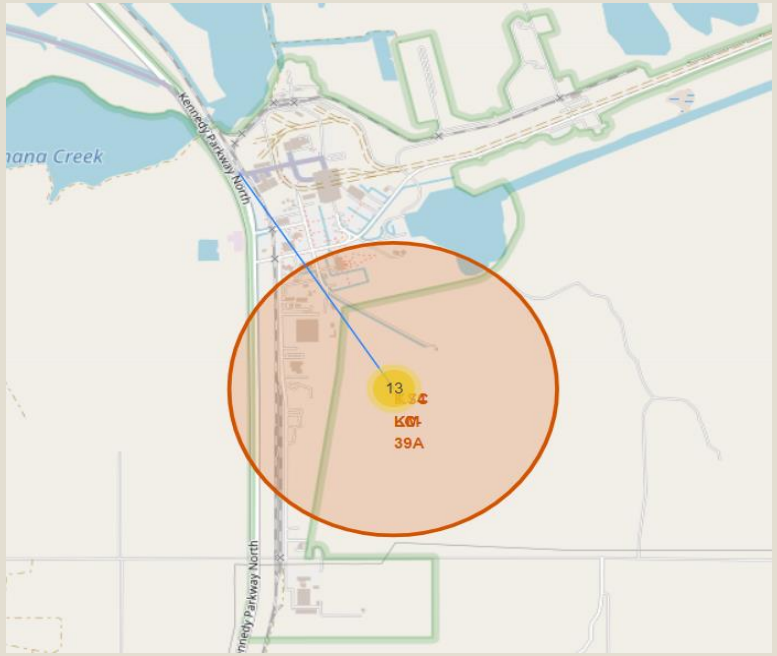




We can observe all four launch sites and their success failure rates displayed by "green" and "red" colour respectively.

Results- Interactive map with Folium

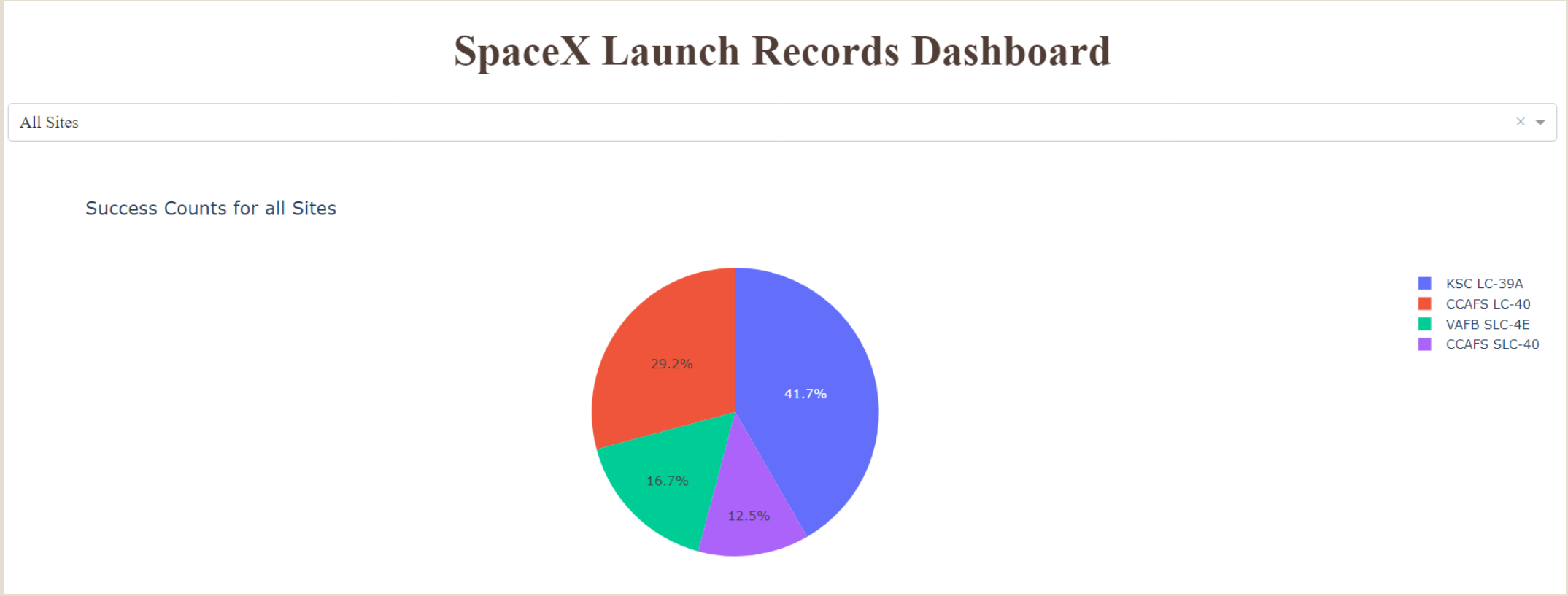
Calculating the distances between a launch site to its proximities



We can observe that for logistic purpose the launch site are in proximities of coasts, highway, railway etc.

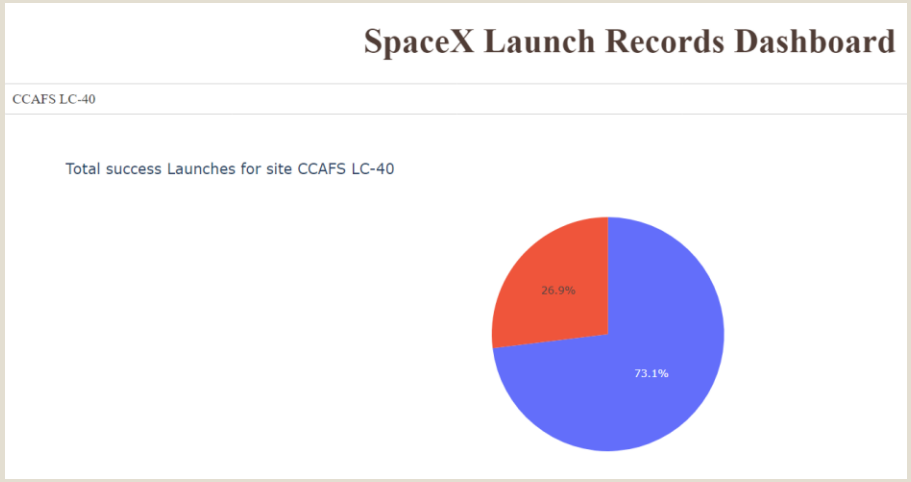
Results- Interactive Dashboard with Plotly

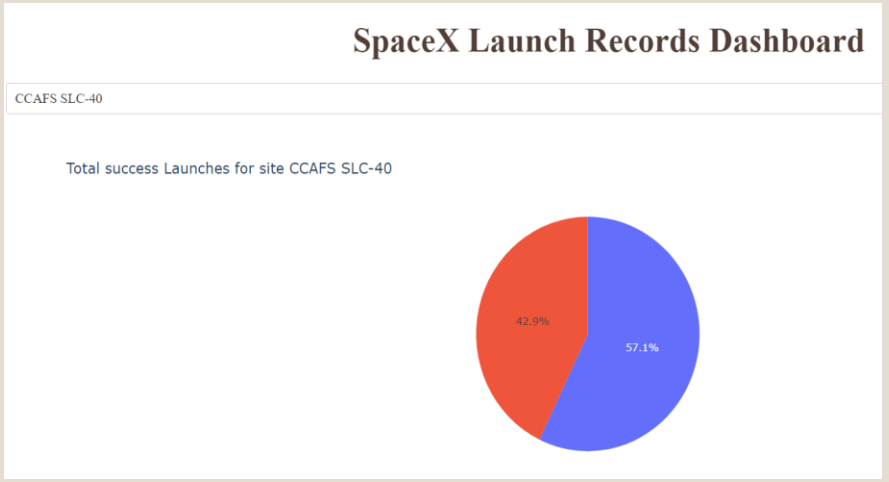
Creating pie chart in interactive dashboard for all Launch Sites and comparing their success proportion. We can clearly see that the KSC-LC has the most successful rate in launch.

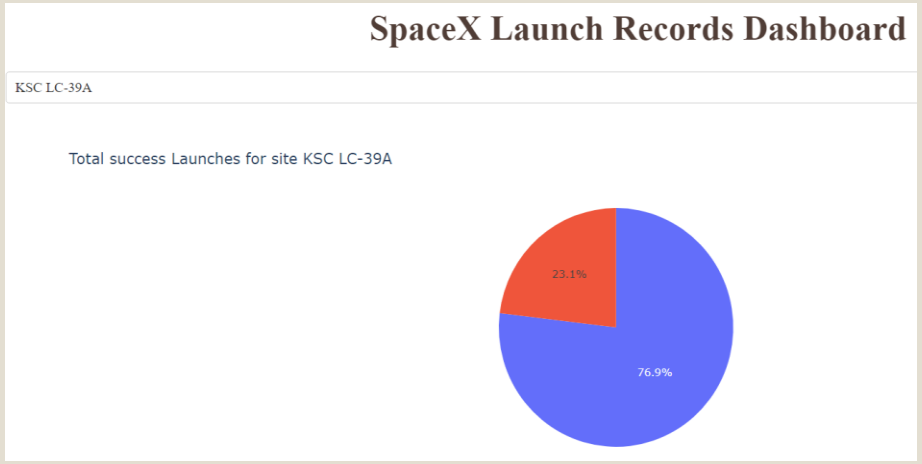


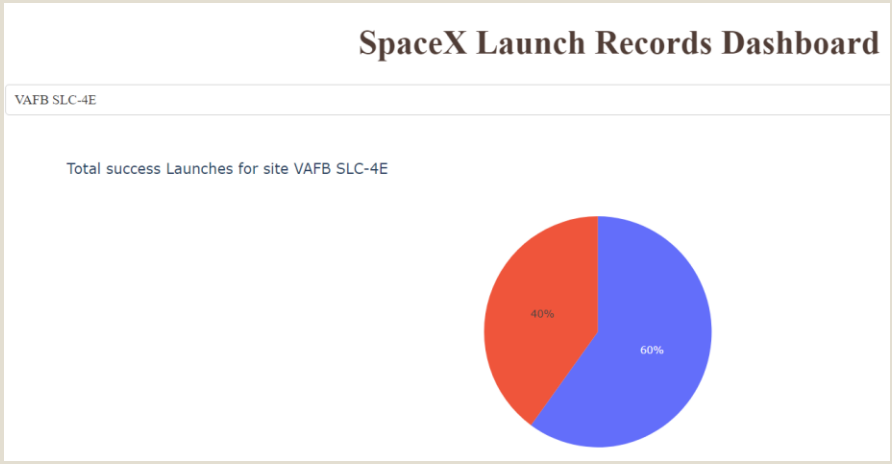
Results- Interactive Dashboard with Plotly

Creating pie chart in interactive dashboard for Individual sites and showing their success failure proportion. Red shows "Success" proportion and Blue "Failure".



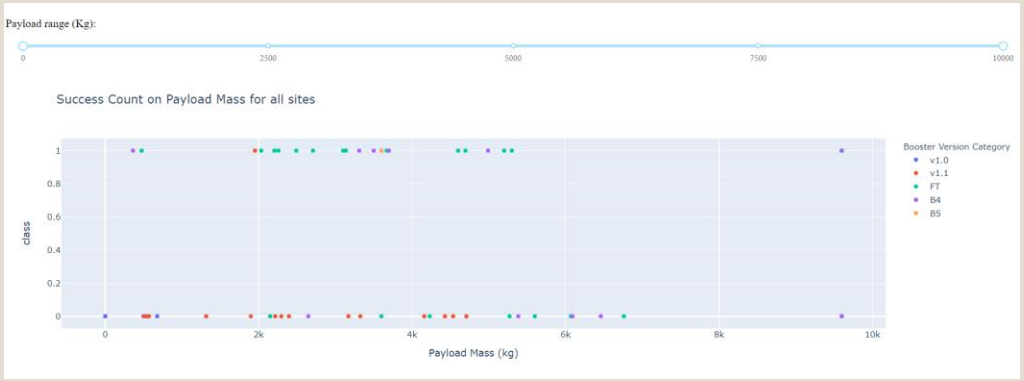


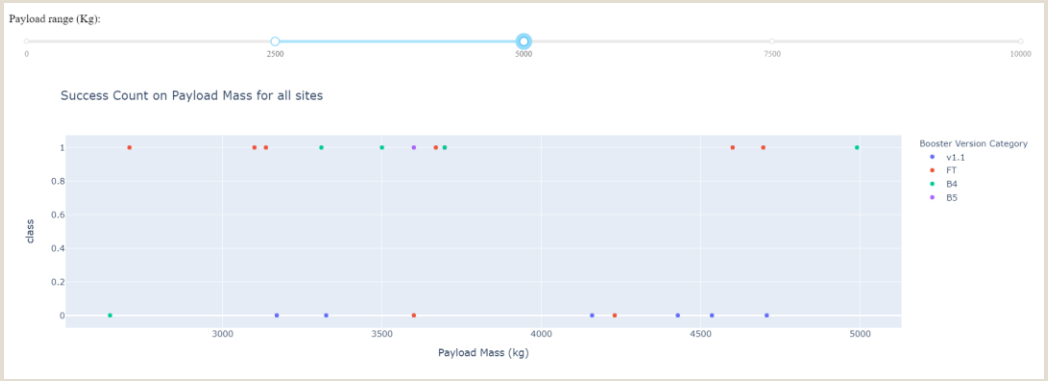


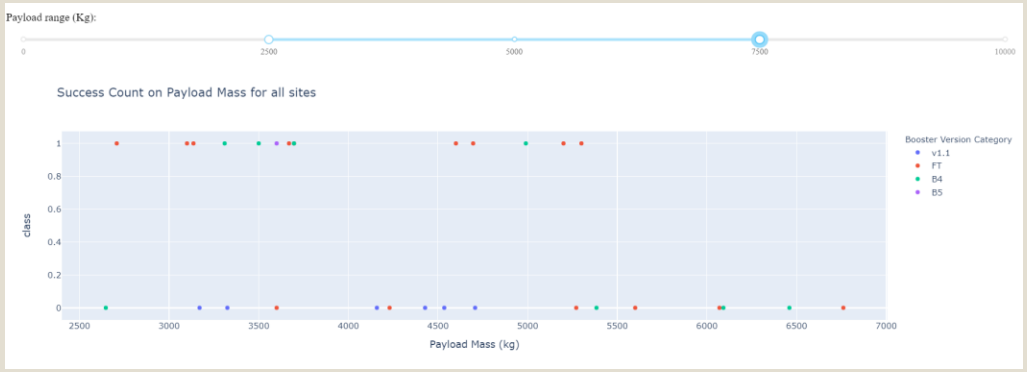


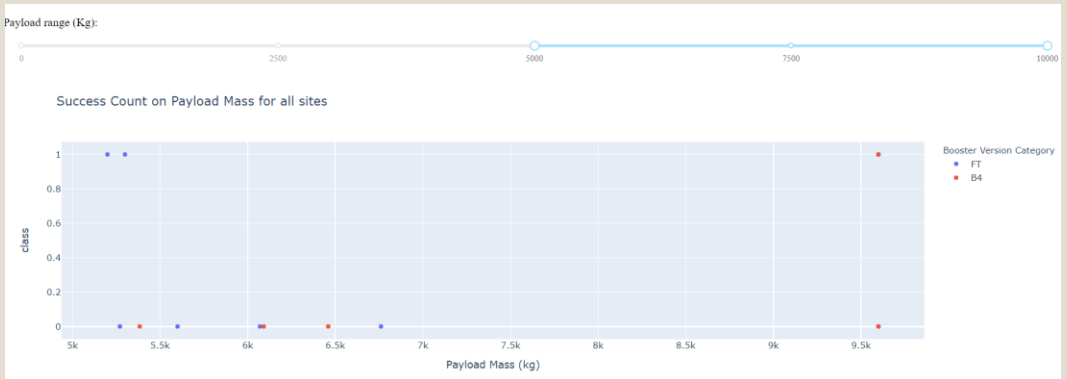
Results- Interactive Dashboard with Plotly

Creating scatter plot in interactive dashboard for all Sites and showing successful launch in variable payload categories.



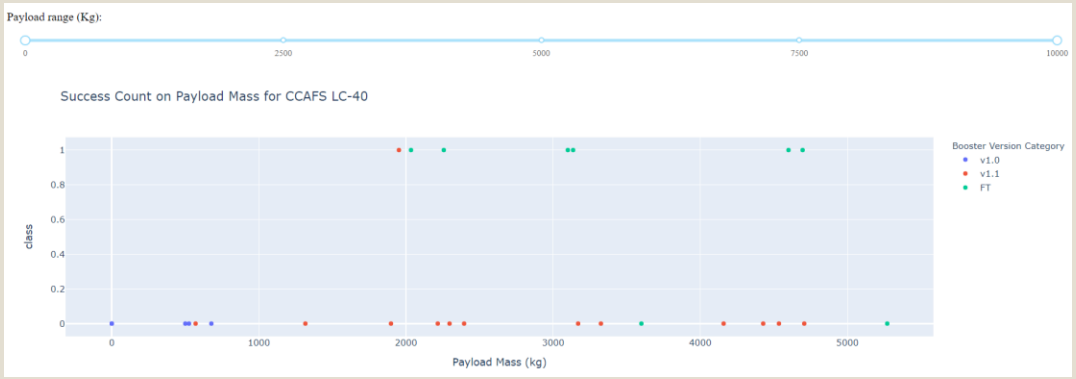




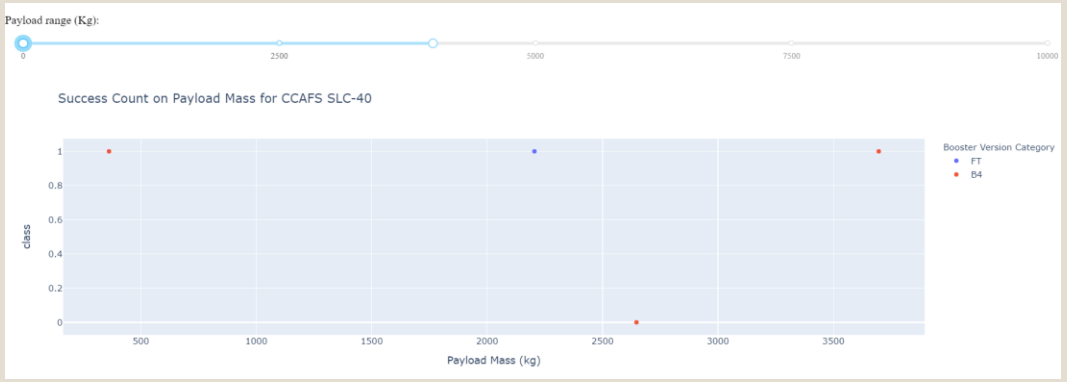


Results- Interactive Dashboard with Plotly

Creating scatter plot in interactive dashboard for individual Sites in relation to their successful payload categories.

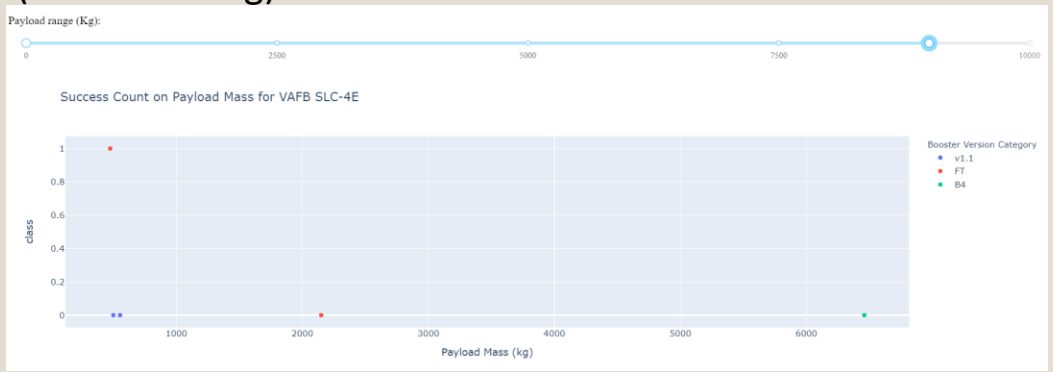


KSC - LC –39A Successful payload category (2000– 5000Kg)



CCAFS SLC –40 Successful payload category (300 – 4000Kg)

KSC - LC –39A Successful payload category (2500– 5500Kg)



VAFB - SLC –E Successful payload category (Below 1000Kg and above 8000)

Results- Predictive analysis (Classification)

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.

For prediction of success outcome we will test these our data set under four classification models:-

- Logistic Regression
- K nearest Neighbor
- Support vector machine (SVM)
- Decision tree classifier

We also require to find the best Hyperparameter for SVM, Classification Trees and Logistic Regression in order to decide best fit ML model for our business problem.

Predictive analysis "Logistic regression"

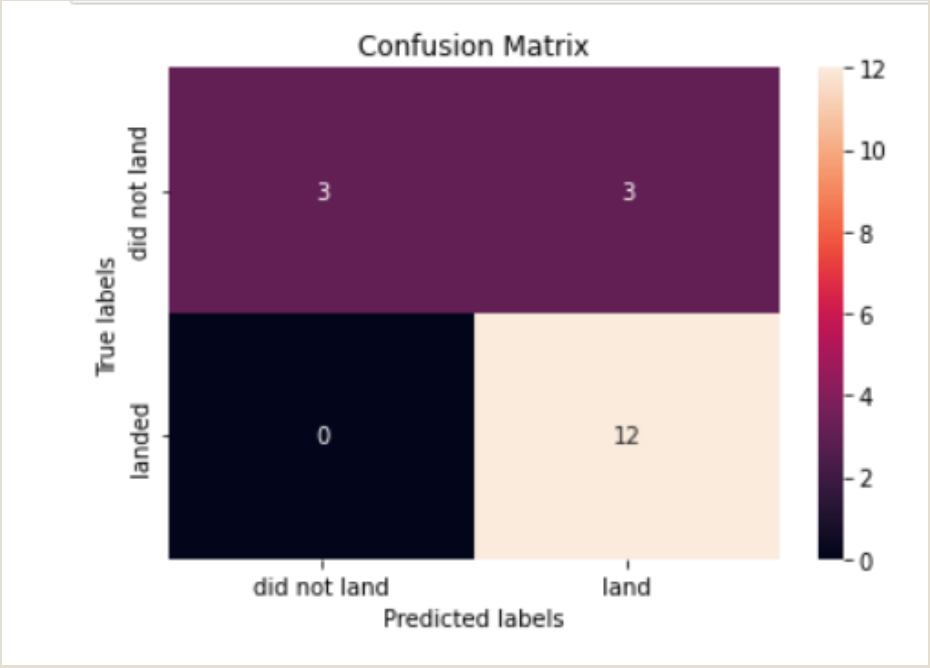
Applying GridSearchCV on a logistic regression object with cv = 10

```
print("accuracy score:- ", logreg_cv.score(X_test, Y_test))
```

```
accuracy score:-  0.8333333333333334
```

```
from sklearn.metrics import classification_report
print (classification_report(Y_test,yhat1))
```

	precision	recall	f1-score	support
0	1.00	0.50	0.67	6
1	0.80	1.00	0.89	12
accuracy			0.83	18
macro avg	0.90	0.75	0.78	18
weighted avg	0.87	0.83	0.81	18




```
print("tuned hpyerparameters :(best parameters) ",logreg_cv.best_params_)
print("accuracy :",logreg_cv.best_score_)
```

```
tuned hpyerparameters :(best parameters)  {'C': 0.01, 'penalty': 'l2', 'solver': 'lbfgs'}
accuracy : 0.8464285714285713
```

Predictive analysis "SVM"

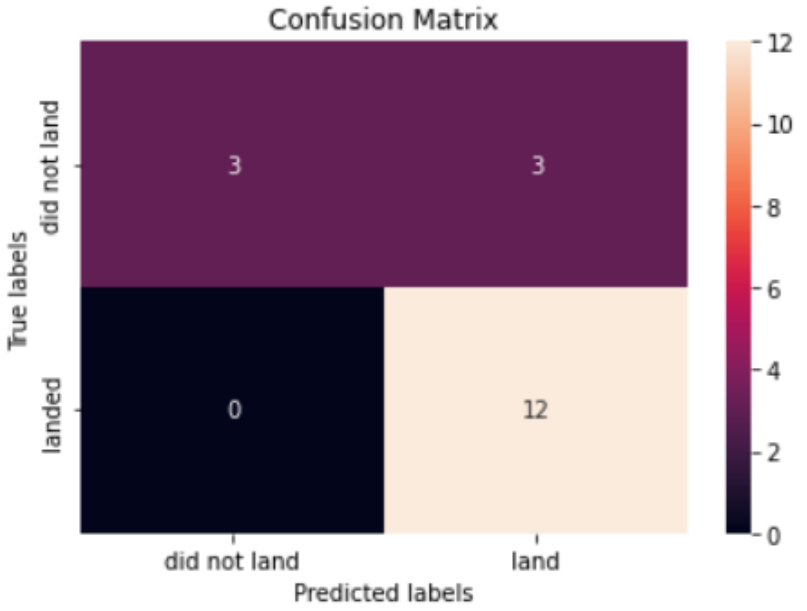
Applying GridSearchCV on a Support vector machine (SVM) object with cv = 10

```
print('Accuracy score: - ', svm_cv.score(X_test, Y_test))
```

```
Accuracy score: -  0.8333333333333334
```

```
print (classification_report(Y_test,yhat2))
```

	precision	recall	f1-score	support
0	1.00	0.50	0.67	6
1	0.80	1.00	0.89	12
accuracy			0.83	18
macro avg	0.90	0.75	0.78	18
weighted avg	0.87	0.83	0.81	18




```
print("tuned hpyerparameters :(best parameters) ",svm_cv.best_params_)
print("accuracy :",svm_cv.best_score_)
```

```
tuned hpyerparameters :(best parameters)  {'C': 1.0, 'gamma': 0.03162277660168379, 'kernel': 'sigmoid'}
accuracy : 0.8482142857142856
```

Predictive analysis "Decision Tree Classifier"

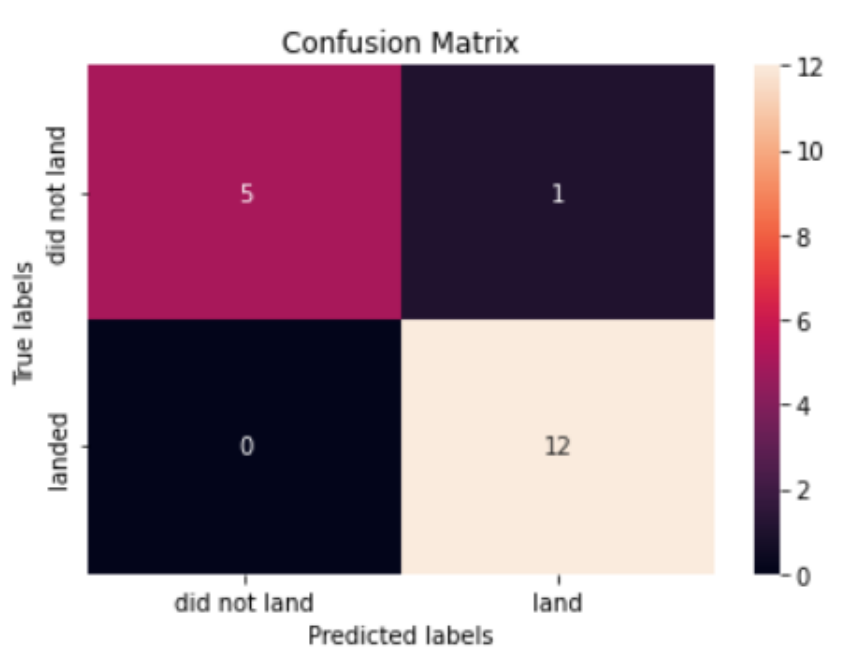
Applying GridSearchCV on a Decision tree classifier object with cv = 10

```
print('Accuracy score: - ', tree_cv.score(X_test, Y_test))
```

```
Accuracy score: -  0.9444444444444444
```

```
print (classification_report(Y_test,yhat3))
```

	precision	recall	f1-score	support
0	1.00	0.83	0.91	6
1	0.92	1.00	0.96	12
accuracy			0.94	18
macro avg	0.96	0.92	0.93	18
weighted avg	0.95	0.94	0.94	18




```
print("tuned hpyerparameters :(best parameters) ",tree_cv.best_params_)
print("accuracy :",tree_cv.best_score_)
```

```
tuned hpyerparameters :(best parameters)  {'criterion': 'gini', 'max_depth': 14, 'max_features': 'sqrt', 'min_samples_leaf': 1,
'min_samples_split': 10, 'splitter': 'best'}
accuracy : 0.8732142857142856
```

Predictive analysis "KNN"

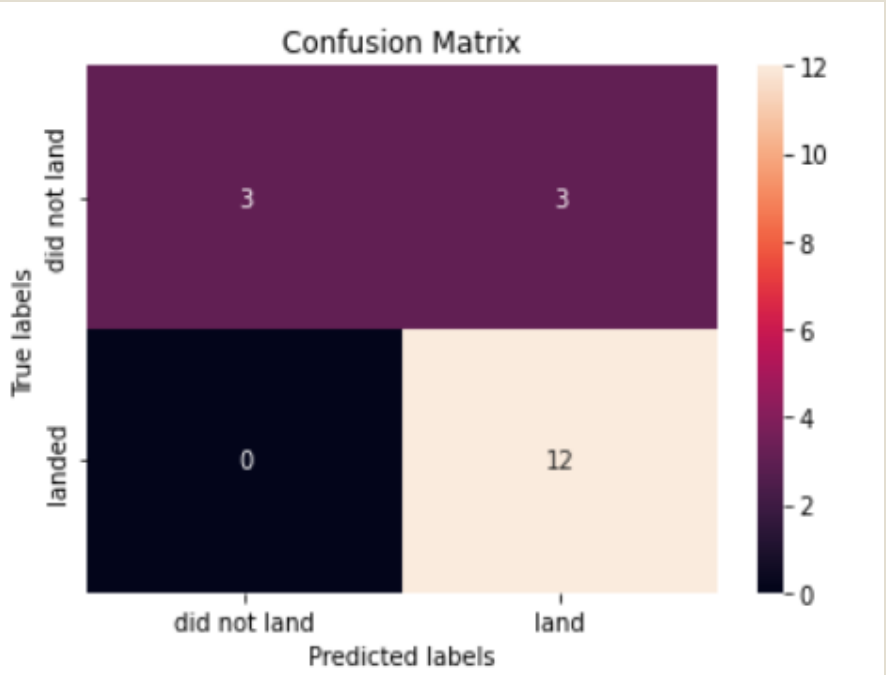
Applying GridSearchCV on a K-Nearest Neighbor object with cv = 10

```
print('Accuracy score: - ', knn_cv.score(X_test, Y_test))
```

```
Accuracy score: -  0.8333333333333334
```

```
print (classification_report(Y_test,yhat4))
```

	precision	recall	f1-score	support
0	1.00	0.50	0.67	6
1	0.80	1.00	0.89	12
accuracy			0.83	18
macro avg	0.90	0.75	0.78	18
weighted avg	0.87	0.83	0.81	18




```
print("tuned hpyerparameters :(best parameters) ",knn_cv.best_params_)
print("accuracy :",knn_cv.best_score_)
```

```
tuned hpyerparameters :(best parameters)  {'algorithm': 'auto', 'n_neighbors': 10, 'p': 1}
accuracy : 0.8482142857142858
```

Best Predictive Algorithm

Finding the Algorithm for predicting highest accuracy along with best suitable parameters.

```
algorithms = {'KNN':knn_cv.best_score_,'Tree':tree_cv.best_score_,'LogisticRegression':logreg_cv.best_score_, 'SVM':svm_cv.best_s
bestalgorithm = max(algorithms, key=algorithms.get)
print('The Best Algorithm from the performed algorithms 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_)
if bestalgorithm == 'SVM':
    print('Best Params is :',svm_cv.best_params_)
```

```
The Best Algorithm from the performed algorithms is Tree with a score of 0.8732142857142856
Best Params is : {'criterion': 'gini', 'max_depth': 14, 'max_features': 'sqrt', 'min_samples_leaf': 1, 'min_samples_split': 10,
'splitter': 'best'}
```

So we can see here that the 'Decision tree classifier' provided the best accuracy amongst all other Algorithms..

- The best score predicted is **0.87**
- The precision is recorded **1.0**
- The recall is **0.83**
- The F1 score is **0.91**

OVERALL FINDINGS & IMPLICATIONS

Findings	Implications
The success rate of Launch vehicle from 2013 to 2020 is good	The rate of success become better as the number of launches increase
Launches with Payload Mass above 7000Kg are mostly went successful.	The Launches with higher payload mass are tend to be more successful
ES-L1, GEO, HEO and SSO Launch outcomes are best.	The higher earth orbits launches like Geosynchronous or Lagrange points are more successful
F9 B5 series booster rockets carried maximum payload mass.	F9 B5 booster series load carrying capacity is best in class.
Successful landing outcome achieved on 22nd Dec 2015	The era of re-useable rockets starts off after this.
All four launching sites are near to highway and Railway line.	The logistic purpose is better handled for minimizing cost.
All four launching sites are in proximities to the equator line.	The equator proximities provides extra momentum to the rocket due to higher centripetal force compared to the polar proximities.
All four launching sites are near to coastline	This is necessary to prevent the collateral damage in case of failure, explosion or mid-air mission collapse.

CONCLUSION

In order to determine the cost of a launch we need to predict if the Falcon 9 first stage will land successfully. After going through Data collection, Cleaning, Visualization, Exploratory Analysis and applying various Classification Model we can summarize our findings as follow:-

- Data Collected for the SpaceX api and Wikipage of SpaceX provided much useful information.
- SpaceX Launches improve over the time.
- The higher Orbit Launches (ES-L1, GEO, HEO and SSO) are mostly successful
- Most successful launch site is KSC LC-39A
- The best Algorithm fits for our purpose is "Decision Tree Classifier"
- The Model performs the prediction with the best accuracy score **.873**

Therefore we can conclude that for our given Business problem the "Decision Tree Classifier" is the best suited Classification ML model.

Links

The link to follow all notebooks of our Data Science project in GitHub.

<https://github.com/Manuviju/Predicting_success_rate_of_SpaceX_Launch_Vehicle.git>