IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Aazim Ahmed
21-12-2023

Outline

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

Executive Summary

Summary of Methodologies:

- Data Collection using API & Web Scraping.
- Exploratory Data Analysis using SQL, Pandas, Matplotlib & Seaborn.
- Interactive Visualizations and Dashboard using Folium & Dash.
- Predictive Analysis using Machine Learning models like Logistic Regression, SVM, Decision Trees and KNN.

Summary of all Results:

- Exploratory Data Analysis results.
- Predictive Analysis results.

Introduction

Background:

- Leading space company SpaceX works to lower the cost of space travel for all people. Among its achievements are spacecraft missions to the International Space Station and the launch of a constellation of satellites that provide both sending manned space missions and providing internet access. SpaceX is able to accomplish this because, thanks to their creative reuse of the Falcon 9 rocket's first stage, rocket missions are comparatively cheap ($62 million per launch). Some suppliers cost more than $165 million apiece and are unable to repeat the first stage.

Objectives:

- What effects do factors like payload mass, orbit, number of flights, and launch site have on the first stage landing's success?
- Does the percentage of successful landings rise with time?
- Which algorithm would be most effective in this situation?

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Data was collected using the SpaceX API and Web Scraping.
- Perform data wrangling
  - Data was wrangled using the Pandas library of Python.
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Evaluated the different models & their best parameters using cross-validation.

Data Collection – SpaceX API

- Request rocket launch data via the SpaceX API.
- json_normalize is used to transform the response to a dataframe once it has been decoded using json().
- Use custom functions to obtain launch information from the SpaceX API.
- Make a dictionary using the information.
- Using the dictionary, create a dataframe.
- Sort the dataframe such that it only includes Falcon 9 launches.
- Replace any missing Payload Mass values with the computed mean.
- Data exported to a CSV file.

Data Collection – Web Scraping

- Request Wikipedia for the Falcon 9 launch data.
- Create a BeautifulSoup object from the response.
- Take the column names out of the HTML table heading.
- Collecting data by analyzing HTML tables.
- Make a dictionary using the data.
- Using the dictionary, create a dataframe.
- Data was exported to a CSV file.

Data Wrangling

- Performed some exploratory Data Analysis and determined Training Labels.
- Calculated the number of launched per site, occurrence of each orbit and landing outcome of each orbit.
- Created a binary variable about Landing Outcome, which ‘1’ for success and ‘0’ for failure.

https://github.com/Aazim10/IBM-Data-Science-Capstone/blob/main/3-%20Data%20Wrangling.ipynb

EDA with Data Visualization

- We plotted scatter charts to analyze the relationship between Flight Number & Payload, Flight Number & Launch Site, Payload & Launch Site, Flight Number & Orbit, Payload & Orbit.
- We compared the success rate of each Orbit using bar chart.
- We used a line chart to observe the yearly success of launches.

https://github.com/Aazim10/IBM-Data-Science-Capstone/blob/main/5-%20EDA%20Visualization.ipynb

EDA with SQL

Performed the following SQL Queries:

- Displaying the names of the unique launch sites in the space mission.
- Displaying 5 records where launch sites begin with the string 'CCA'.
- Displaying the total payload mass carried by boosters launched by NASA (CRS).
- Displaying average payload mass carried by booster version F9 v1.1.
- Listing the date when the first successful landing outcome in ground pad was achieved.
- Listing the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000.
- Listing the total number of successful and failure mission outcomes.
- Listing the names of the booster versions which have carried the maximum payload mass.
- Listing the failed landing outcomes in drone ship, their booster versions and launch site names for the months in year 2015.
- Ranking 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.

https://github.com/Aazim10/IBM-Data-Science-Capstone/blob/main/4-%20EDA%20SQL.ipynb

Build an Interactive Map with Folium

Markers of all Launch Sites:

- Added Marker with Circle, Popup Label and Text Label of NASA Johnson Space Center and all of the Launch Sites using its latitude and longitude coordinates as a start location.

Colored Markers of the launch outcomes for each Launch Site:

- Added green colored Markers for success and red colored Markers for failed launches using Marker Cluster to identify which launch sites have relatively high success rates.

Distances between a Launch Site to its proximities:

- Added Lines & distance Markers to show distances between the Launch Site VAFB SLC-4E and its proximities like Railway, Highway, Coastline and Closest City.

https://github.com/Aazim10/IBM-Data-Science-Capstone/blob/main/6-%20Interactive%20Visual%20Analytics.ipynb

Build a Dashboard with Plotly Dash

Launch Sites Dropdown List:

- Added a dropdown list to enable Launch Site selection.

Pie Chart showing Success Launches (All Sites/Certain Site):

- Added a pie chart to show the total successful launches count for all sites and the Success vs. Failed counts for the site, if a specific Launch Site was selected.

Slider of Payload Mass Range:

- Added a slider to select Payload range.

Scatter Chart of Payload Mass vs. Success Rate for the different Booster Versions:

- Added a scatter chart to show the correlation between Payload and Launch Success.

https://github.com/Aazim10/IBM-Data-Science-Capstone/blob/main/7-%20Interactive%20Dashboard.ipynb

Predictive Analysis (Classification)

- Created a NumPy array from the Class column.
- Standardized the data with StandardScaler.
- Splitted the data using train_test_split.
- Created a GridSearchCV (To choose the best hyperparameters) object for each of the following models: Logistic Regression, Support Vector Machine, Decision Tree & K-Nearest Neighbor.
- Calculate accuracy on the test data for all models.
- Analyzed the confusion matrix for all models.
- Identified the best model using Accuracy scores on training and testing data.

https://github.com/Aazim10/IBM-Data-Science-Capstone/blob/main/8-%20Predictive%20Analysis.ipynb

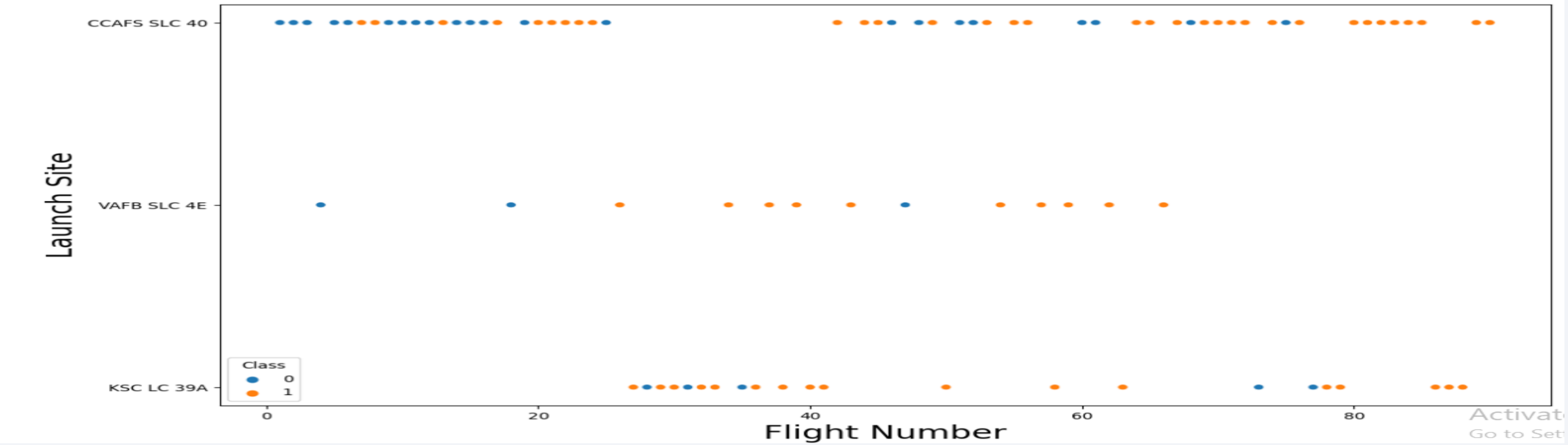
Results

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

Section 2

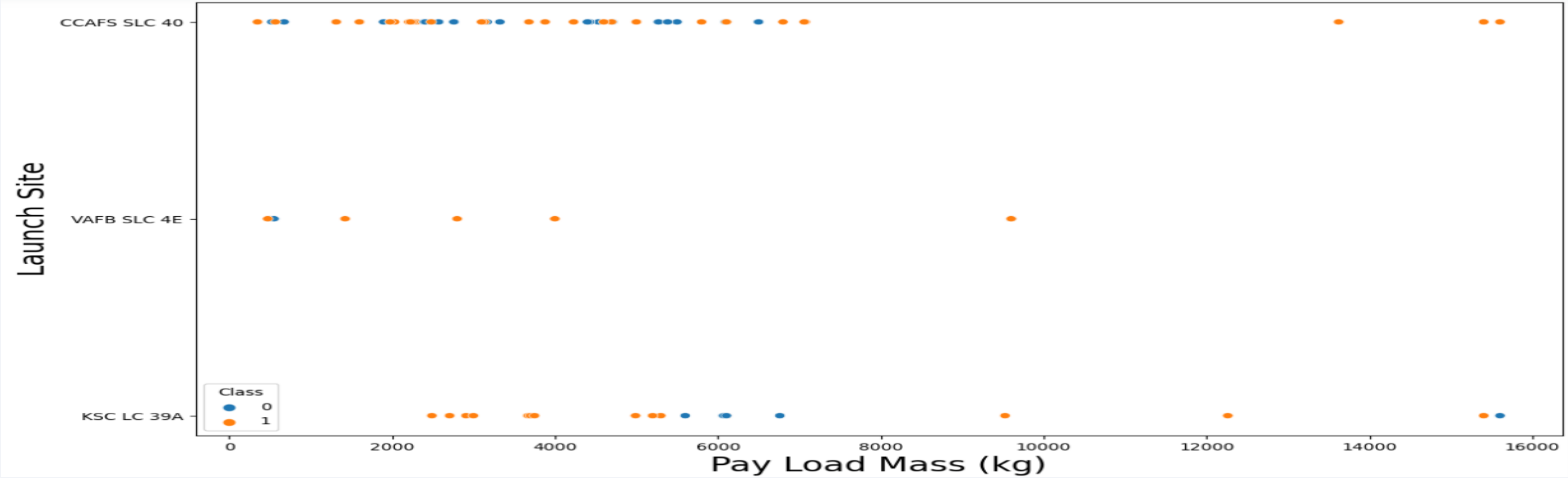
Insights drawn from EDA

Flight Number vs. Launch Site



- Earlier Flights had a lower success rate than recent ones.
- A large proportion of flights are from CCAFS CLC-40 launch site.
- KSC LC-39A has the overall highest success rate.

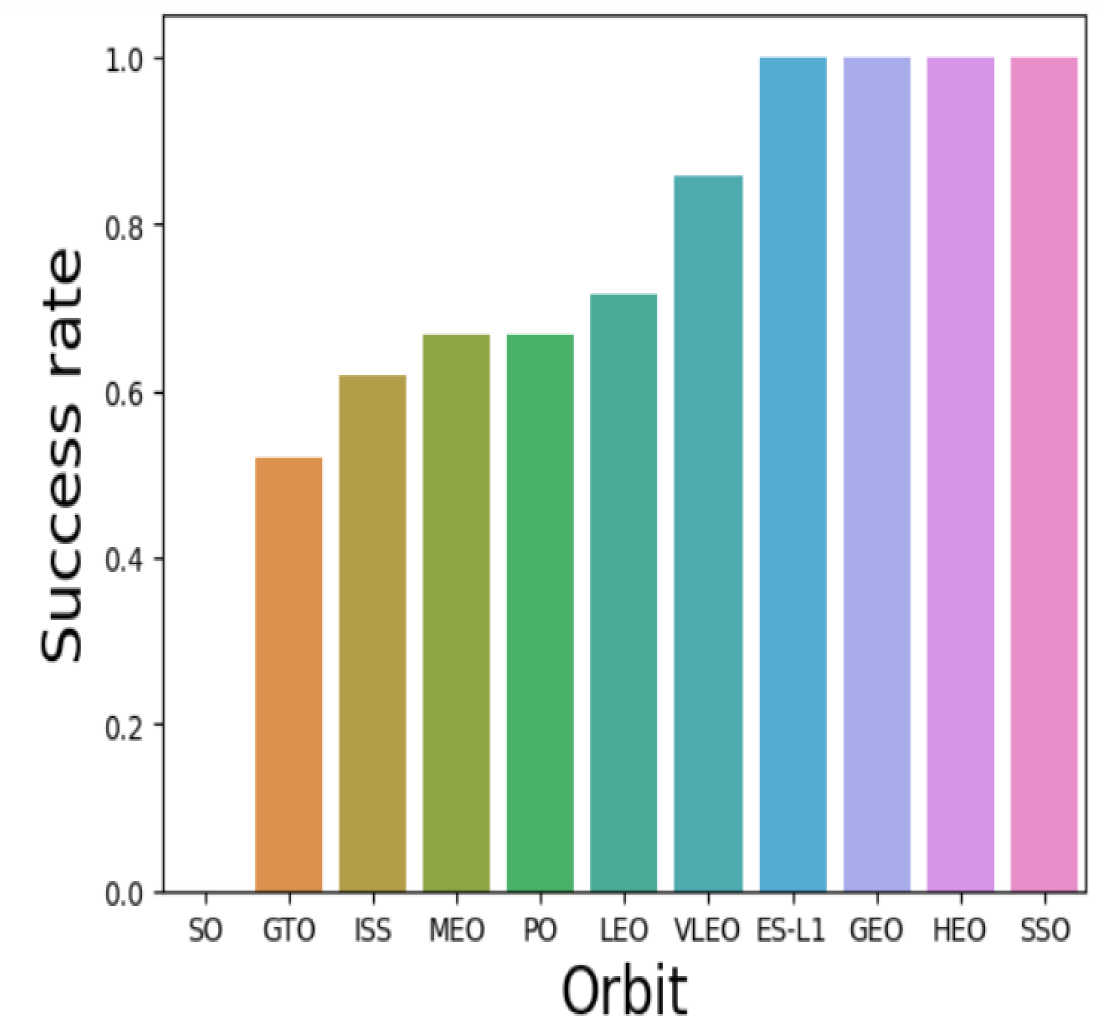
Payload vs. Launch Site



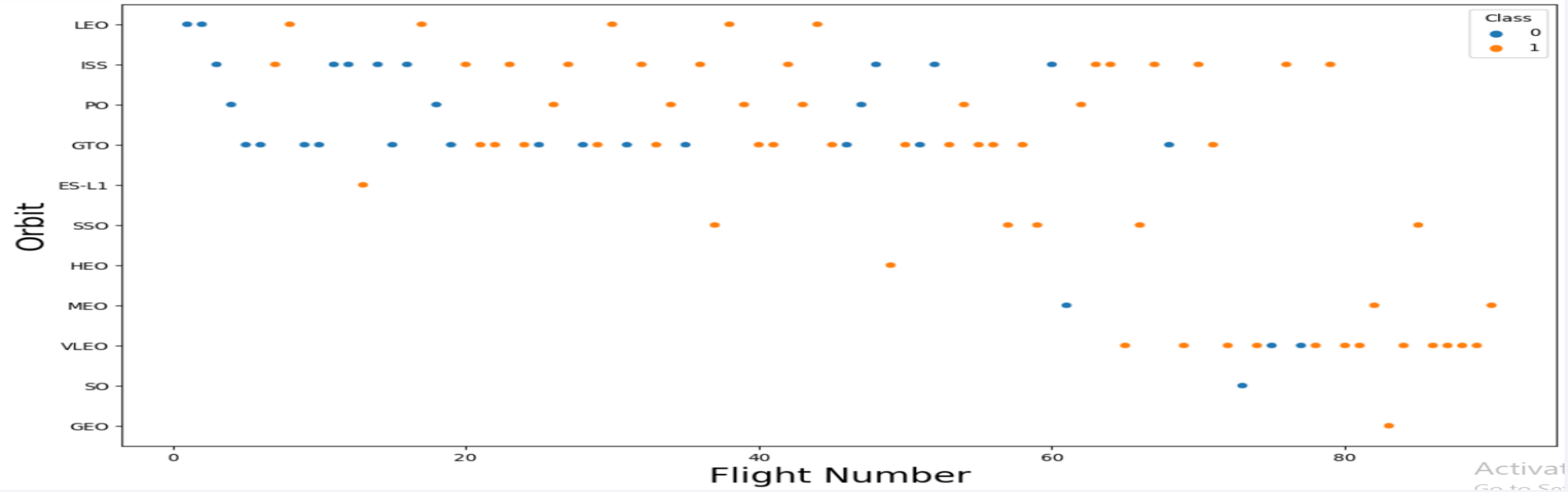
- Higher Payloads have more success rate.
- Flights with a Payload greater 8000 kg are very view.

Success Rate vs. Orbit Type

- ES-L1, GEO, HEO, SSO have a success rate of 100%.
- There are no successes from SO Orbit.
- VLEO & LEO also have high success rates.

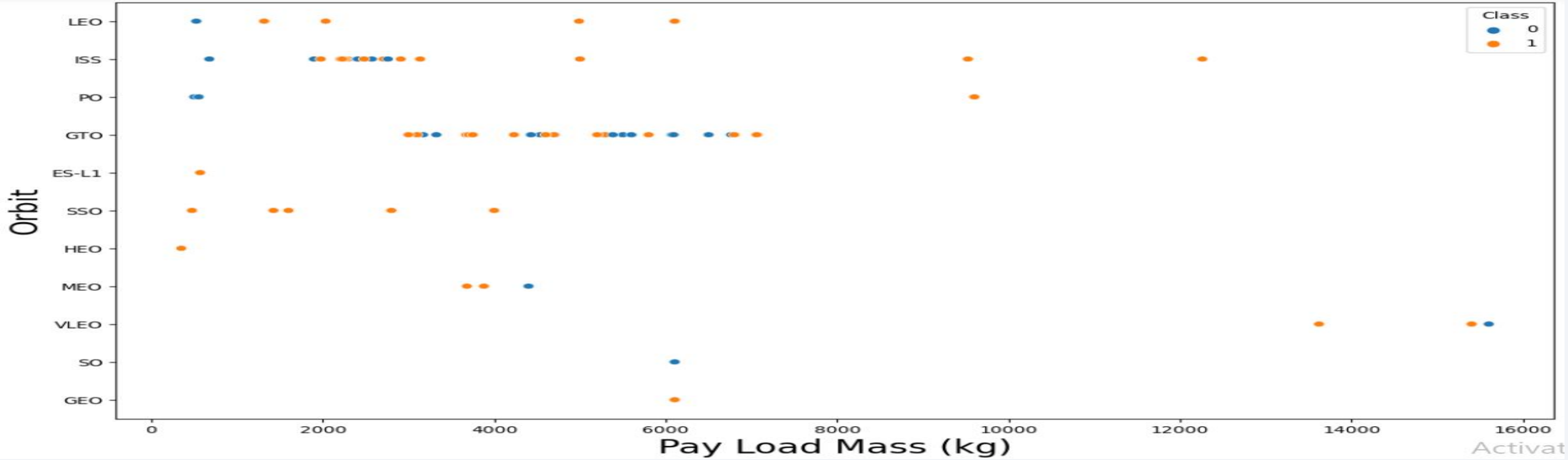


Flight Number vs. Orbit Type



- Success rates have improved over time across all Orbits.
- A lot of the recent launches have been made through the VLEO Orbit.

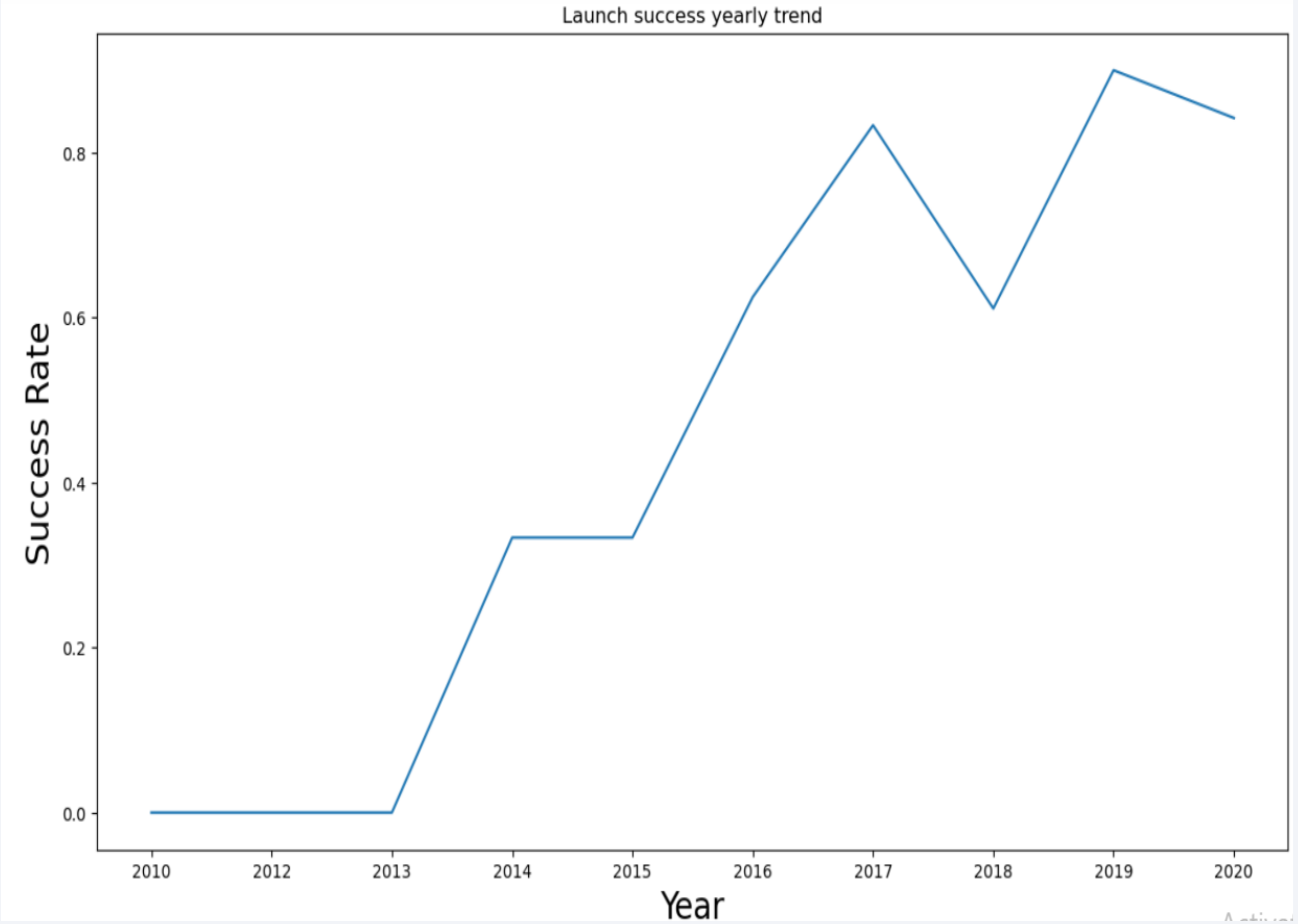
Payload vs. Orbit Type



- GTO shows a neutral relation with Payload.
- ISS and LEO show positive relation with Payload.

Launch Success Yearly Trend

- The success rate of initial years is zero which is reasonable as they were still learning.
- The success rate shows an increase from 2013-17 then a decline in 2018 but still greater than 60%.



All Launch Site Names

```
mycursor.execute('SELECT DISTINCT LAUNCH_SITE as "Launch_Sites" FROM SPACEXTABLES')
display(mycursor)
```

```
('CCAFS LC-40',)
('VAFB SLC-4E',)
('KSC LC-39A',)
('CCAFS SLC-40',)
```

Displaying the names of the unique launch sites in the space mission.

Launch Site Names Begin with 'CCA'

```
mycursor.execute('SELECT * FROM SPACEXTABLES WHERE LAUNCH_SITE LIKE "CCA%" LIMIT 5')
display(mycursor)
```

```
(datetime.datetime(2010, 6, 4, 0, 0), '18:45:00', 'F9 v1.0  B0003', 'CCAFS LC-40', 'Dragon Spacecraft Qualification Unit', 0,
'LEO', 'SpaceX', 'Success', 'Failure (parachute)')
(datetime.datetime(2010, 12, 8, 0, 0), '15:43:00', 'F9 v1.0  B0004', 'CCAFS LC-40', 'Dragon demo flight C1, two CubeSats, barre
l of Brouere cheese', 0, 'LEO (ISS)', 'NASA (COTS) NRO', 'Success', 'Failure (parachute)')
(datetime.datetime(2012, 5, 22, 0, 0), '7:44:00', 'F9 v1.0  B0005', 'CCAFS LC-40', 'Dragon demo flight C2', 525, 'LEO (ISS)',
'NASA (COTS)', 'Success', 'No attempt')
(datetime.datetime(2012, 10, 8, 0, 0), '0:35:00', 'F9 v1.0  B0006', 'CCAFS LC-40', 'SpaceX CRS-1', 500, 'LEO (ISS)', 'NASA (CR
S)', 'Success', 'No attempt')
(datetime.datetime(2013, 3, 1, 0, 0), '15:10:00', 'F9 v1.0  B0007', 'CCAFS LC-40', 'SpaceX CRS-2', 677, 'LEO (ISS)', 'NASA (CR
S)', 'Success', 'No attempt')
```

Displaying 5 records where launch sites begin with the string 'CCA’.

Total Payload Mass

```
mycursor.execute('SELECT SUM(PAYLOAD_MASS__KG_) FROM SPACEXTABLES WHERE CUSTOMER = "NASA (CRS)"')
display(mycursor)
```

```
(Decimal('45596'),)
```

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

Average Payload Mass by F9 v1.1

```
mycursor.execute('SELECT AVG(PAYLOAD_MASS__KG_) FROM SPACEXTABLES WHERE BOOSTER_VERSION="F9 v1.1"')
display(mycursor)
```

```
(Decimal('2928.4000'),)
```

Displaying average payload mass carried by booster version F9 v1.1.

First Successful Ground Landing Date

```
mycursor.execute('SELECT MIN(DATE) FROM SPACEXTABLES WHERE LANDING_OUTCOME LIKE "Success%" ')
display(mycursor)
```

```
(datetime.datetime(2015, 12, 22, 0, 0),)
```

Listing the date when the first successful landing outcome in ground pad was achieved.

Successful Drone Ship Landing with Payload between 4000 and 6000

```
mycursor.execute('SELECT BOOSTER_VERSION FROM SPACEXTABLES WHERE LANDING_OUTCOME = "Success (drone ship)" AND PAYLOAD_MASS__KG_ E
display(mycursor)
```

```
('F9 FT B1022',)
('F9 FT B1026',)
('F9 FT  B1021.2',)
('F9 FT  B1031.2',)
```

Listing the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000.

Total Number of Successful and Failure Mission Outcomes

```
mycursor.execute('SELECT COUNT(MISSION_OUTCOME) AS number_of_success_outcomes FROM SPACEXTABLES WHERE MISSION_OUTCOME LIKE "SUCCE
display(mycursor)
mycursor.execute('SELECT COUNT(MISSION_OUTCOME) AS number_of_failure_outcomes FROM SPACEXTABLES WHERE MISSION_OUTCOME NOT LIKE "S
display(mycursor)
```

```
(100,)
(1,)
```

Listing the total number of successful and failure mission outcomes.

Boosters Carried Maximum Payload

```
mycursor.execute('SELECT DISTINCT BOOSTER_VERSION FROM SPACEXTABLES WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM
display(mycursor)
```

```
('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 ',)
```

Listing the names of the booster versions which have carried the maximum payload mass.

2015 Launch Records

```
mycursor.execute('SELECT DATE, BOOSTER_VERSION, LAUNCH_SITE FROM SPACEXTABLES WHERE DATE >= "2015-01-01" AND DATE < "2016-01-01"
display(mycursor)
```

```
(datetime.datetime(2015, 1, 10, 0, 0), 'F9 v1.1 B1012', 'CCAFS LC-40')
(datetime.datetime(2015, 4, 14, 0, 0), 'F9 v1.1 B1015', 'CCAFS LC-40')
```

Listing the failed landing outcomes in drone ship, their booster versions and launch site names for the months in year 2015.

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

```
mycursor.execute('SELECT LANDING_OUTCOME, COUNT(LANDING_OUTCOME) AS COUNT_OF_OUTCOME FROM SPACEXTABLES WHERE DATE BETWEEN "2010-0
display(mycursor)
```

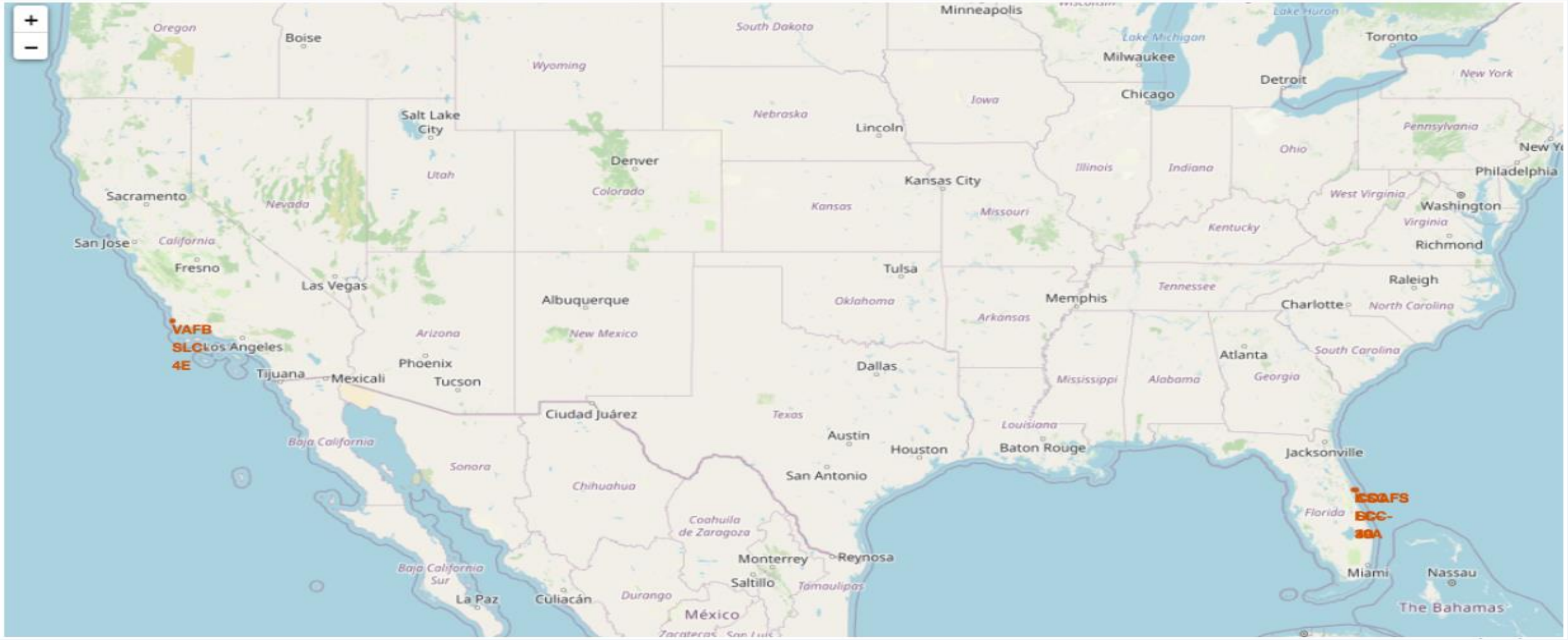
```
('No attempt', 10)
('Failure (drone ship)', 5)
('Success (drone ship)', 5)
('Controlled (ocean)', 3)
('Success (ground pad)', 3)
('Failure (parachute)', 2)
('Uncontrolled (ocean)', 2)
('Precluded (drone ship)', 1)
```

Ranking 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.

Section 3

Launch Sites Proximities Analysis

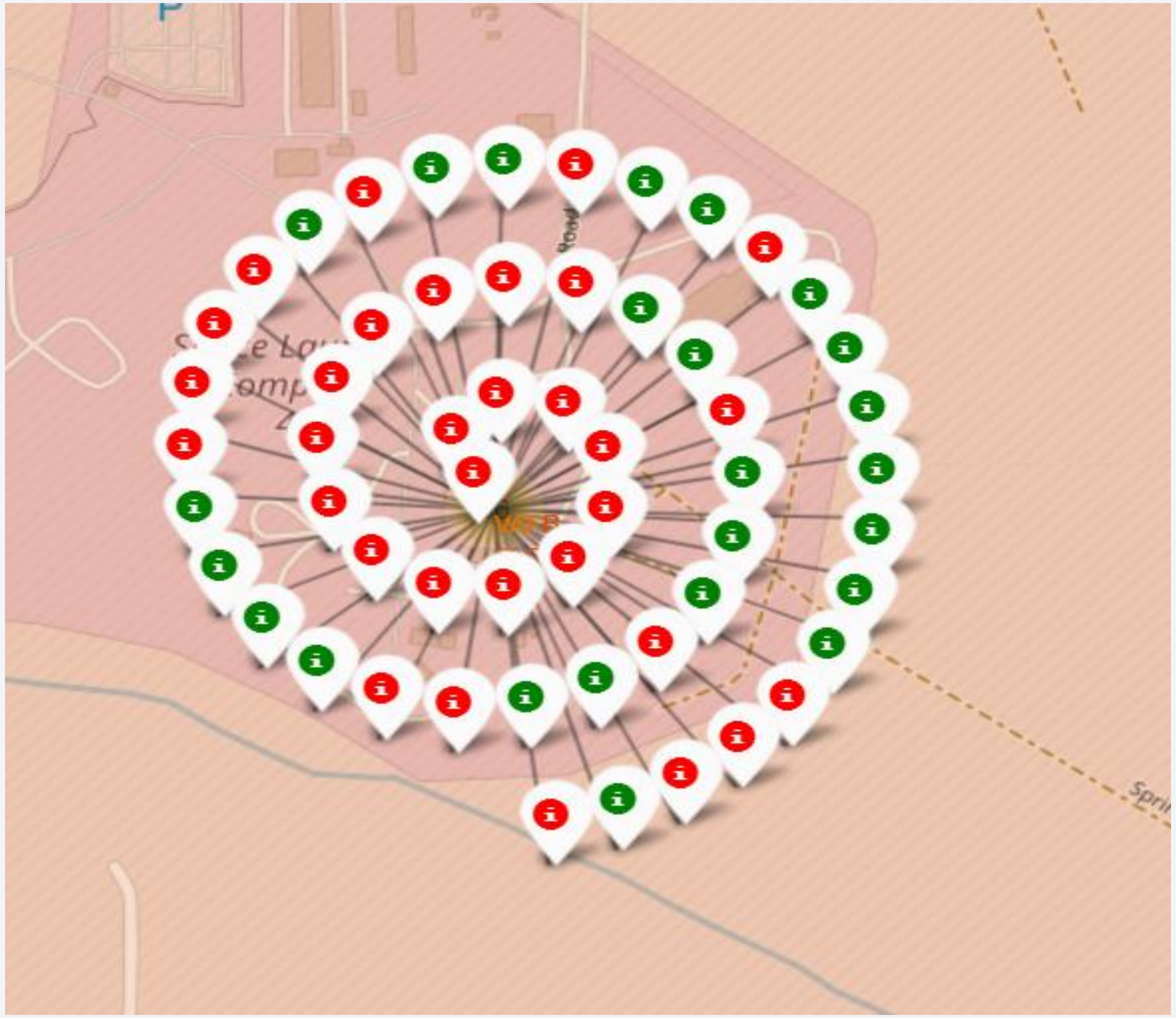
All Launch Sites



- All launch sites are near the sea.

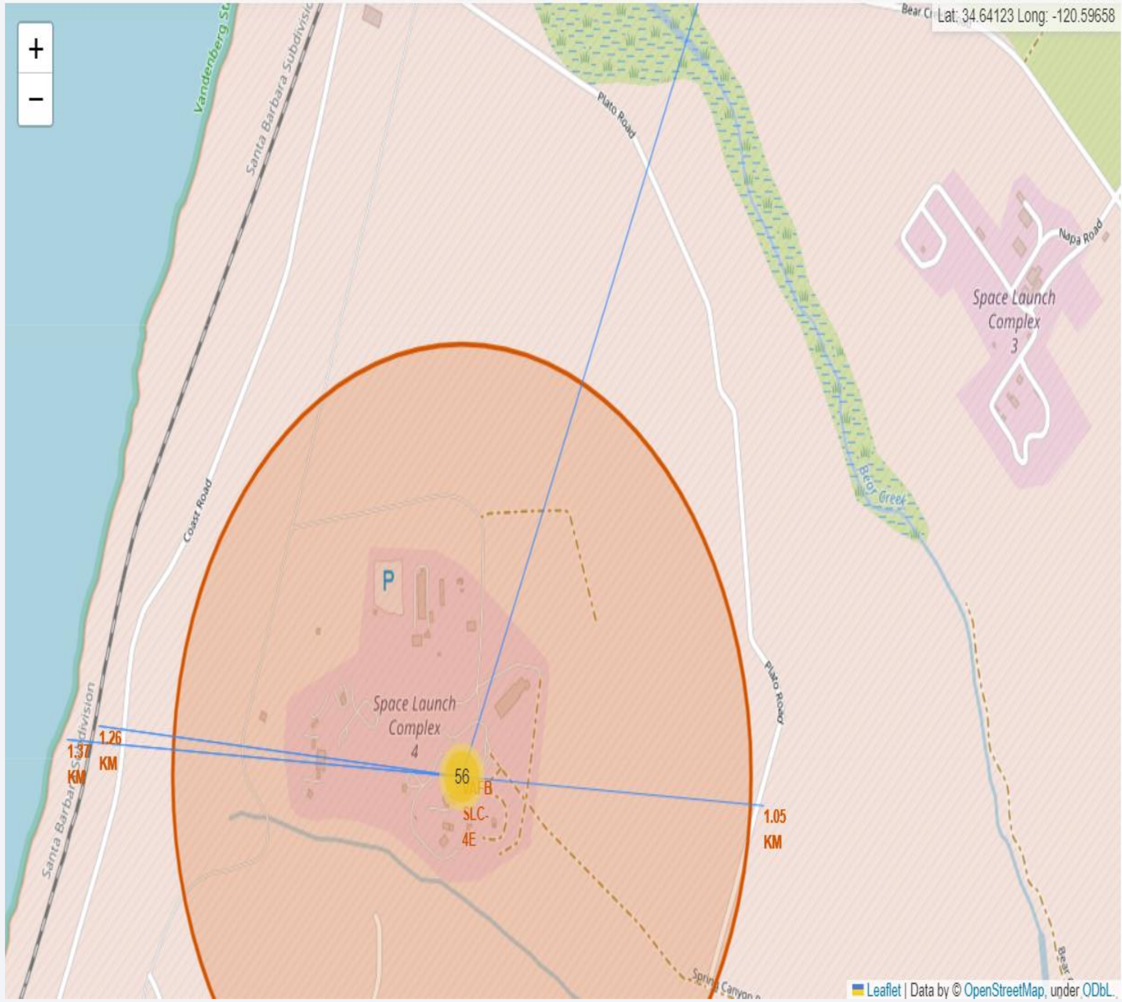
Launch Outcomes by Site

- Example of VAFB SLC-4E Launch Site launch outcomes.
- Green markers indicate successful and red ones indicate failure.



Distance to Proximities

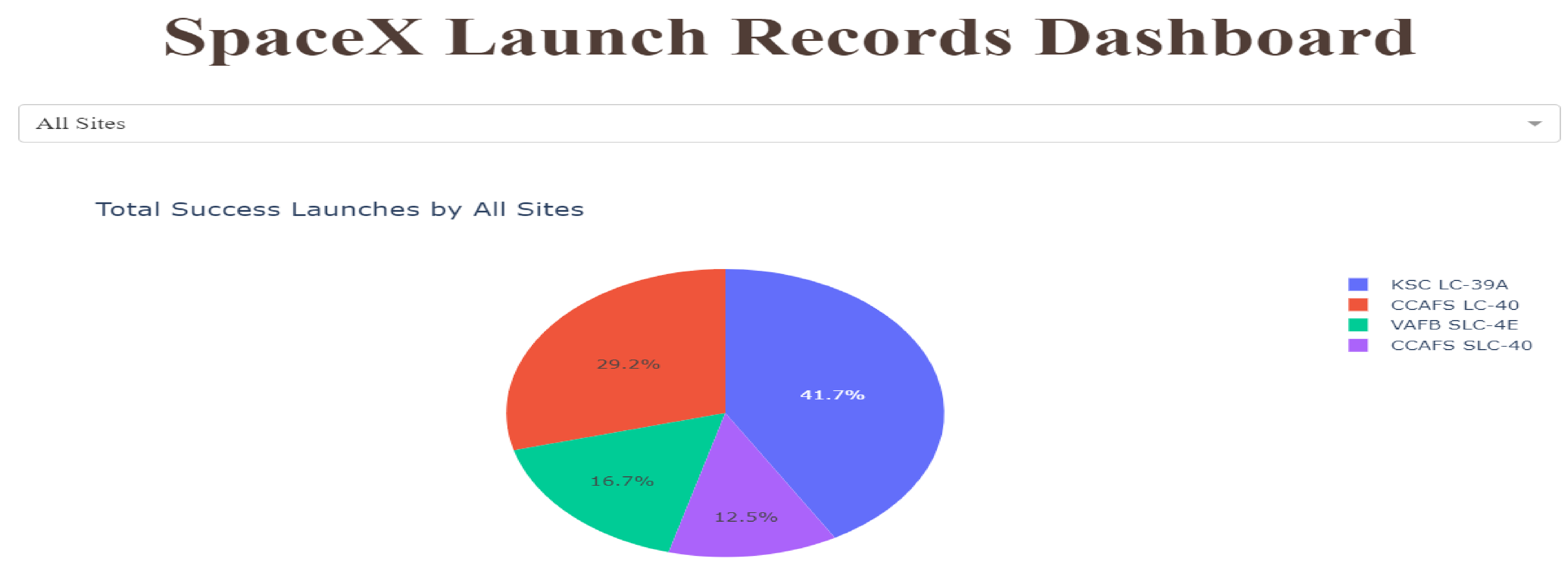
- Distance of the site VAFB SLC-4E to nearest :
  - Railway : 1.26 km
  - Coastline : 1.37 km
  - Highway : 1.05 km
  - City (Santa Maria) : 38.81 km



Section 4

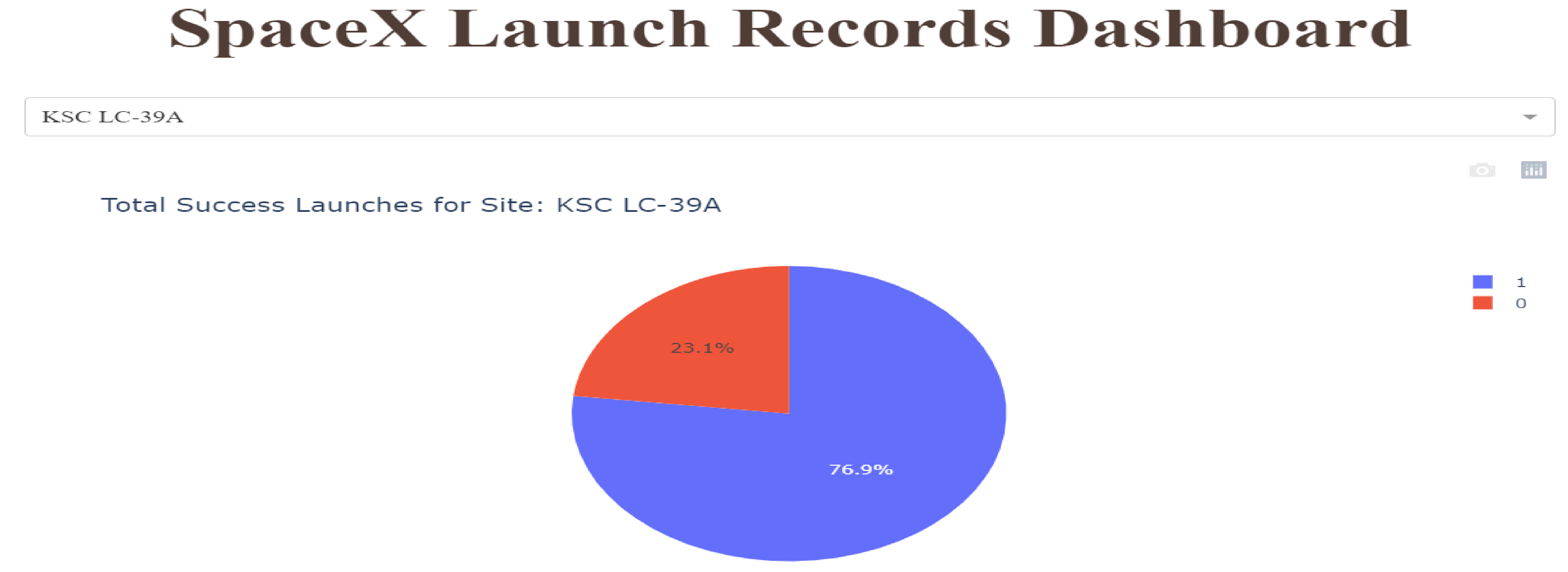
Build a Dashboard with Plotly Dash

Launch Success by Site



- The Launch Site KSC LC-39A has the highest success rate while VAFB SLC-4E has the lowest.

Launch Success of the Site KSC LC-39A



- The Launch Site KSC LC-39A has success ratio of 76.9%.

Payload & Launch Outcome Relationship

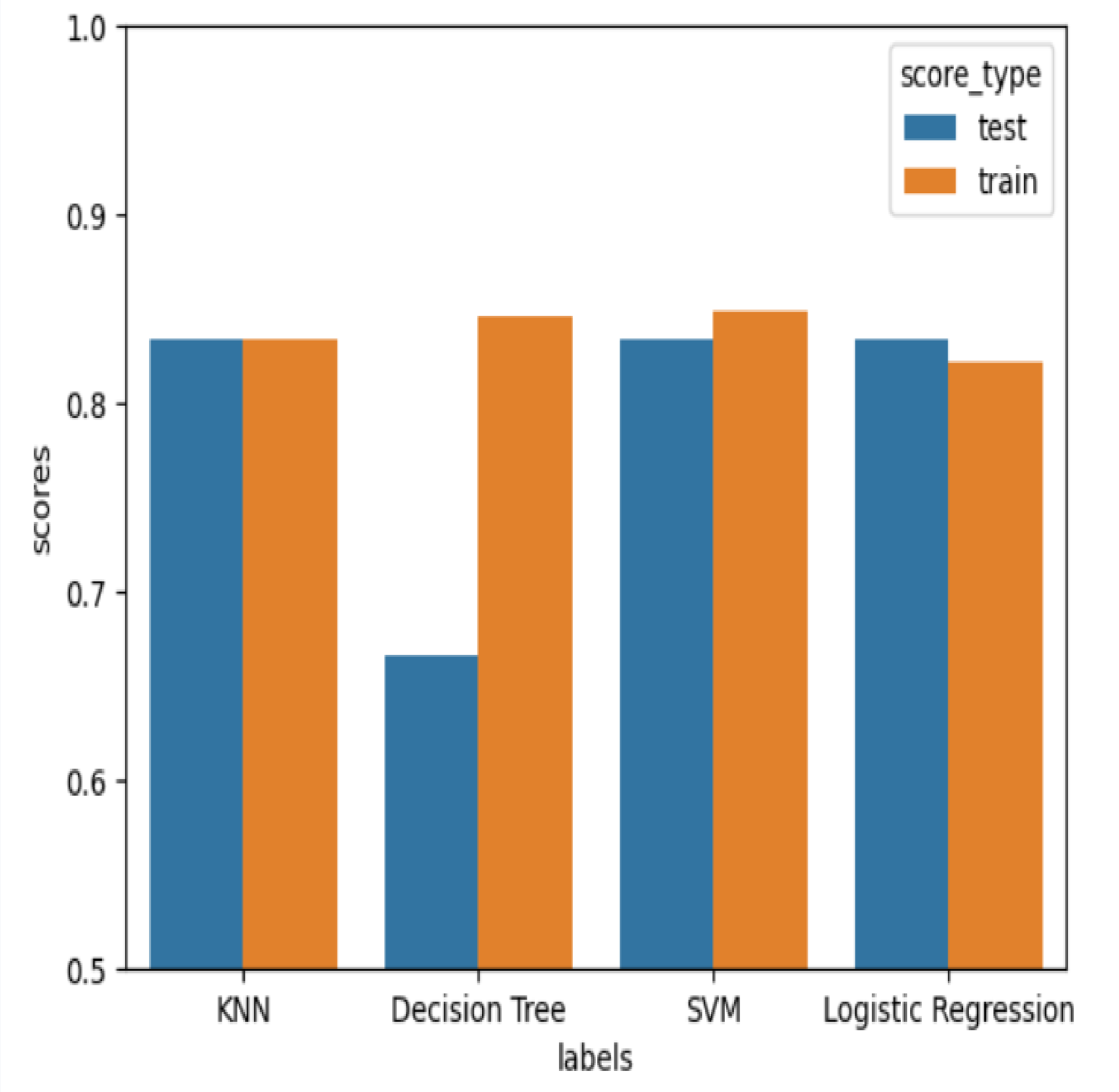
- The charts show that payloads between 2000 and 5500 kg have the highest success rate.

Section 5

Predictive Analysis (Classification)

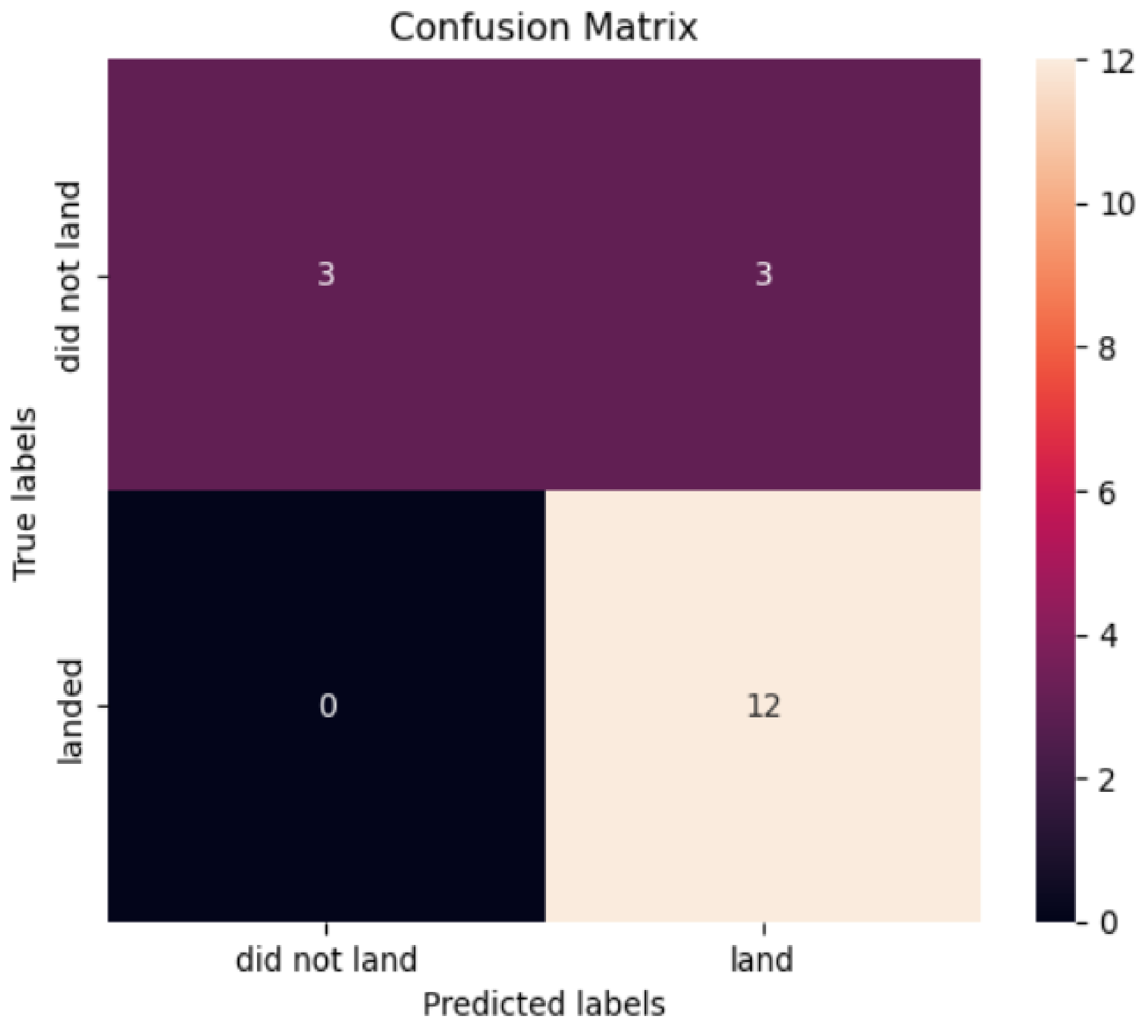
Classification Accuracy

- Based on the accuracy score for both training and testing data, we can conclude that Support Vector Machines is the best model for this classification.



Confusion Matrix

- The confusion matrix shows that the SVM model predicted 15 out 18 test data correctly.



Conclusions

- The best launch site is KSC LC-39A having success ratio of 76.9%.
- All launch sites are near the sea.
- All launch sites are near the Equator.
- Launch success rate increases over time.
- Orbits ES L1, GEO, HEO, and SSO have a 100% success rate.
- Launches with higher Payloads are more successful.
- Support Vector Machines is the best Classifier for this task

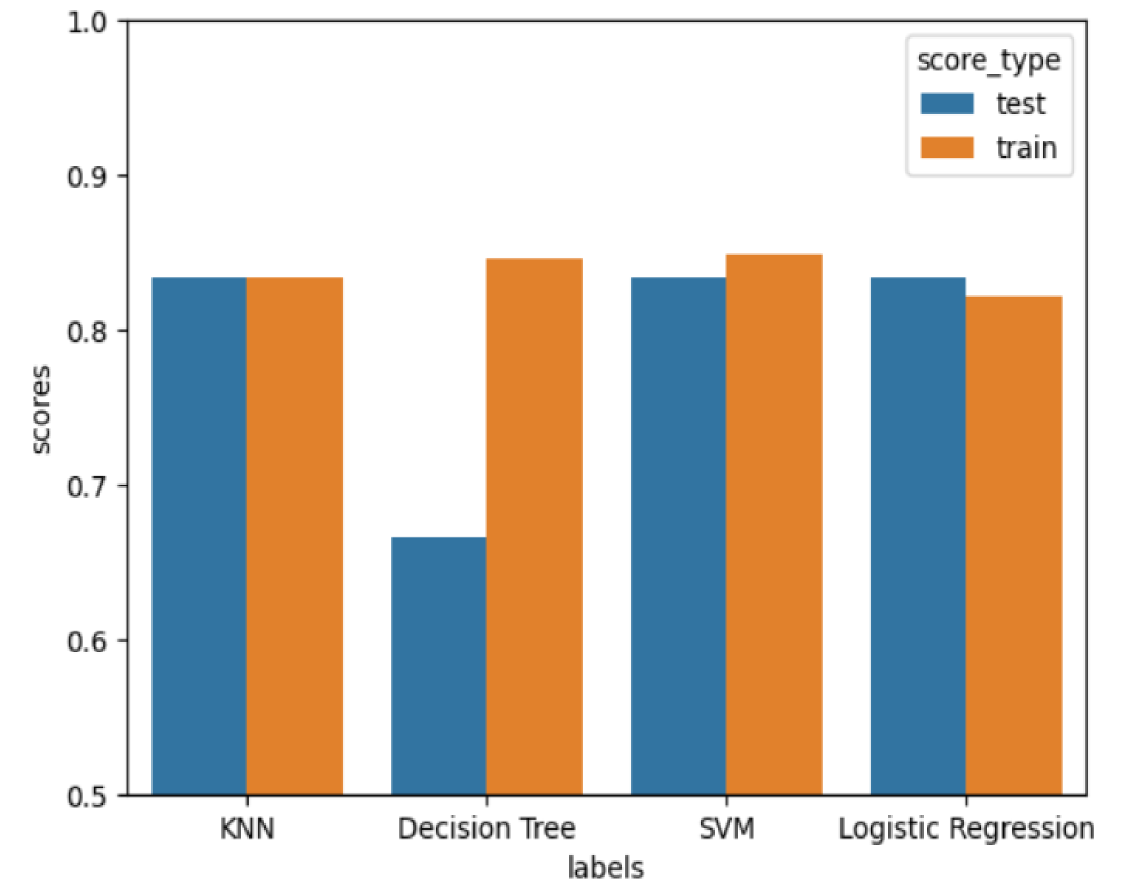
Appendix

Visualizing the Accuracy of different models :

```
labels = ['KNN',  'Decision Tree', 'SVM', 'Logistic Regression','KNN',  'Decision Tree', 'SVM', 'Logistic Regression']
score_type = ['test', 'test', 'test', 'test', 'train', 'train', 'train', 'train']
scores = [knn_cv.score(X_test, Y_test), tree_cv.score(X_test, Y_test), svm_cv.score(X_test, Y_test), logreg_cv.score(X
df_acc = pd.DataFrame()
df_acc['labels'] = pd.Series(labels)
df_acc['scores'] = pd.Series(scores)
df_acc['score_type'] =  pd.Series(score_type)
df_acc.head()
```

	labels	scores	score_type
0	KNN	0.833333	test
1	Decision Tree	0.666667	test
2	SVM	0.833333	test
3	Logistic Regression	0.833333	test
4	KNN	0.833929	train

```
sns.barplot(x='labels', y='scores', hue='score_type', data=df_acc)
plt.ylim(0.5,1)
```



Thank you!