IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Panuwat Tanapornchinpong
28/7/2022

Outline

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

Executive Summary

- Summary of methodologies
  - Data collection through SpaceX API
  - Data collection with Web Scraping
  - Data Wrangling
  - EDA with Data Visualization
  - EDA with SQL
  - Interactive Map with Folium and Plotly Dash
  - ML Prediction
- Summary of all results
  - Exploratory data analysis results
  - Interactive analytics demo in screenshots
  - Predictive analysis results

Introduction

Project background and context

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

- What factors do tell that the rocket will land success?
- The interaction among features that determine success rate of landing
- What operating conditions needs to ensure successful landing.

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Using SpaceX API and web scraping from Wikipedia website.
- Perform data wrangling
  - One-hot encoding was used to categorize 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
  - Using grid search and use many ML method such as LR, SVM.

Data Collection

- Data was collected with various methods
  - Request data from SpaceX API
  - Decode the response content using .json function and turn to pandas dataframe
  - Check missing values . Then, clean the data if necessary
  - Also, do some web scraping from Wikipedia to get Falcon 9 launch records using BeautifulSoup.
  - The data from Wikipedia was recorded as HTML ,so it converted to pandas dataframe from easier analysis

Data Collection – SpaceX API

- We requested data from SpaceX API to collect data, clean the data and did basic data formatting.

- [SpaceX_IBM/data-collection-api-spacexv2.ipynb at b0375adf5ad05680ea6c4b8e70be32151251c383 · ampppppppp/SpaceX_IBM (github.com)]

1. Get data from URL using SpaceX API and turn into pandas dataframe

To make the requested JSON results more consistent, we will use the following static response object for this project:

```
static_json_url='https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DS0321EN-SkillsNetwork/datasets/API_call_spacex_api.json
```

We should see that the request was successfull with the 200 status response code

```
response.status_code
```

```
200
```

Now we decode the response content as a Json using `.json()` and turn it into a Pandas dataframe using `.json_normalize()`

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

2. Replace some missing values

Calculate below the mean for the `PayloadMass` using the `.mean()`. Then use the mean and the `.replace()` function to replace `np.nan` values in the data with the mean you calculated.

```
# Calculate the mean value of PayloadMass column
mean=data_falcon9['PayloadMass'].mean()


data_falcon9['PayloadMass']=data_falcon9['PayloadMass'].replace(np.nan,mean)


data_falcon9.isnull().sum()
# Replace the np.nan values with its mean value
```

Data Collection - Scraping

- Using Beautifulsoup to scraping from Wikipedia website to get Falcon 9 flight records. We parsed the table and turned into pandas dataframe.

- [SpaceX_IBM/data-collection-webscraping-spacex.ipynb at main · amppppppp/SpaceX_IBM (github.com)]()

1. Apply HTTP get method to request data from website and create BeautifulSoup object

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

```
In [7]: # use requests.get() method with the provided static_url
        response=requests.get(static_url).text
        # assign the response to a object
```

Create a `BeautifulSoup` object from the HTML `response`

```
In [8]: # Use BeautifulSoup() to create a BeautifulSoup object from a response text content
        soup=BeautifulSoup(response,'html.parser')
```

2.Extract all column names from HTML table header

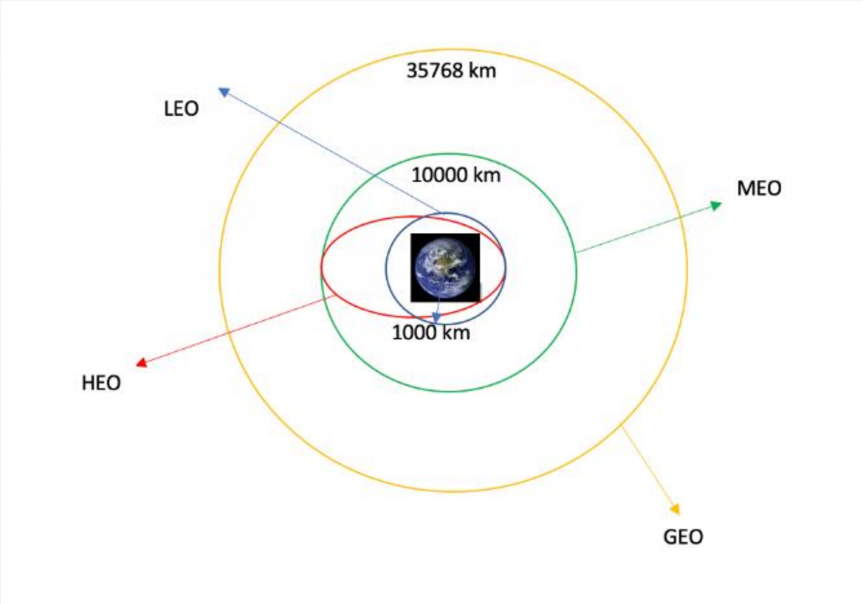
```
In [15]:
column_names = []

# 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_

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
```

3. Create pandas dataframe from HTML tables

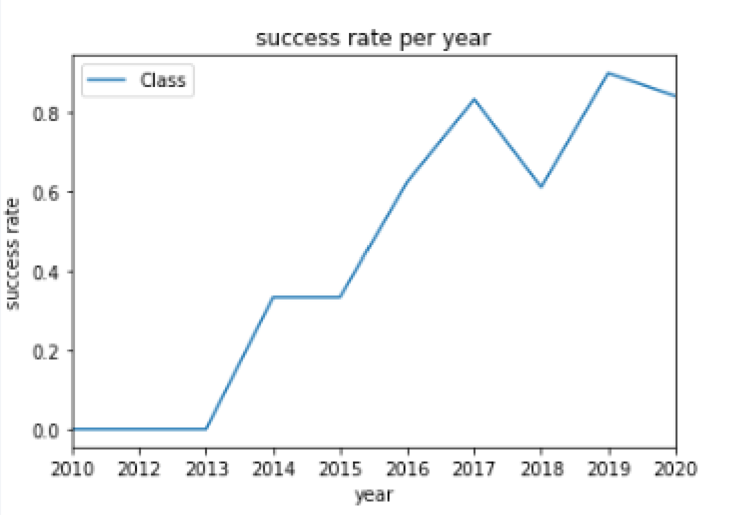
Data Wrangling

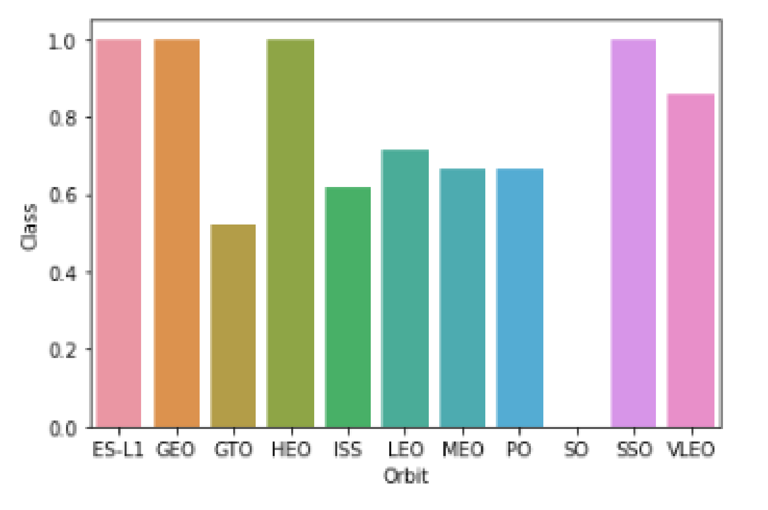


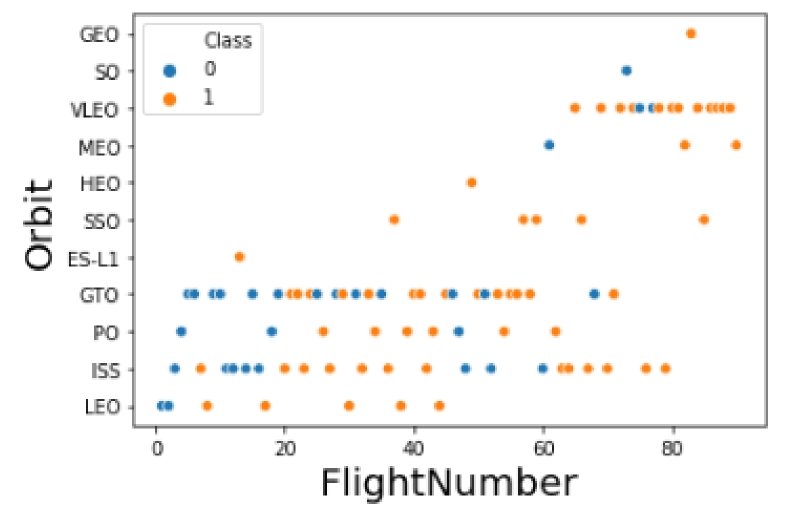
- The exploratory analysis was performed and training labels were determined.
- We calculated the number of Falcon 9's launches on each site. Moreover, the number and occurrence of each orbits.
- We also created a landing outcome label for further analysis.
- [SpaceX_IBM/Data-wrangling-spacex.ipynb at main · ampppppppp/SpaceX_IBM (github.com)]

EDA with Data Visualization

- We used visualization to show relationship between different features from the dataset such as success rate vs orbit type, number of flight vs Orbit type, payload weight vs orbit types ,and overall success rate of landing from 2010-2020.







[SpaceX_IBM/eda-dataviz-spacex.ipynb at main · amppppppp/SpaceX_IBM (github.com)]

EDA with SQL

- We applied EDA with SQL using many SQL command for doing tasks such as
  - Display distinct launching sites name
  - Display payload mass by specific booster version
  - List the date when first successful landing on land
  - List name of boosters which success in drone ship and specific payload mass range
  - Rank the successful landing outcomes within date range.

SpaceX_IBM/eda-sql-coursera_sqllite.ipynb at main · amppppppp/SpaceX_IBM (github.com)

Build an Interactive Map with Folium

- We marked Mark the success/failed launches for each site on the map and calculate the distances between a launch site on Folium map.
- The feature outcomes were assigned to 1 for success and 0 for failure.
- We also calculated between launch site and its proximityies such as railway, highway and city.

SpaceX_IBM/launch_site_location-gro-spacex.ipynb at main · ampppppppp/SpaceX_IBM (github.com)

Build a Dashboard with Plotly Dash

- We plotted pie chars showing the total launches by a certain sites and scatter graph showing relationship between outcome and payload mass for different booster version.

SpaceX_IBM/plotly-spacex at main · ampppppppp/SpaceX_IBM (github.com)

Predictive Analysis (Classification)

- We loaded data using numpy and pandas .Then, we prepare data for training ML model by split the data into testing set and training set.
- We built different ML models and tune different hyperparameters using GridSearch
- The best performing classification is Decision Tree

SpaceX_IBM/Machine Learning Prediction_SpaceX_.ipynb at main · amppppppp/SpaceX_IBM (github.com)

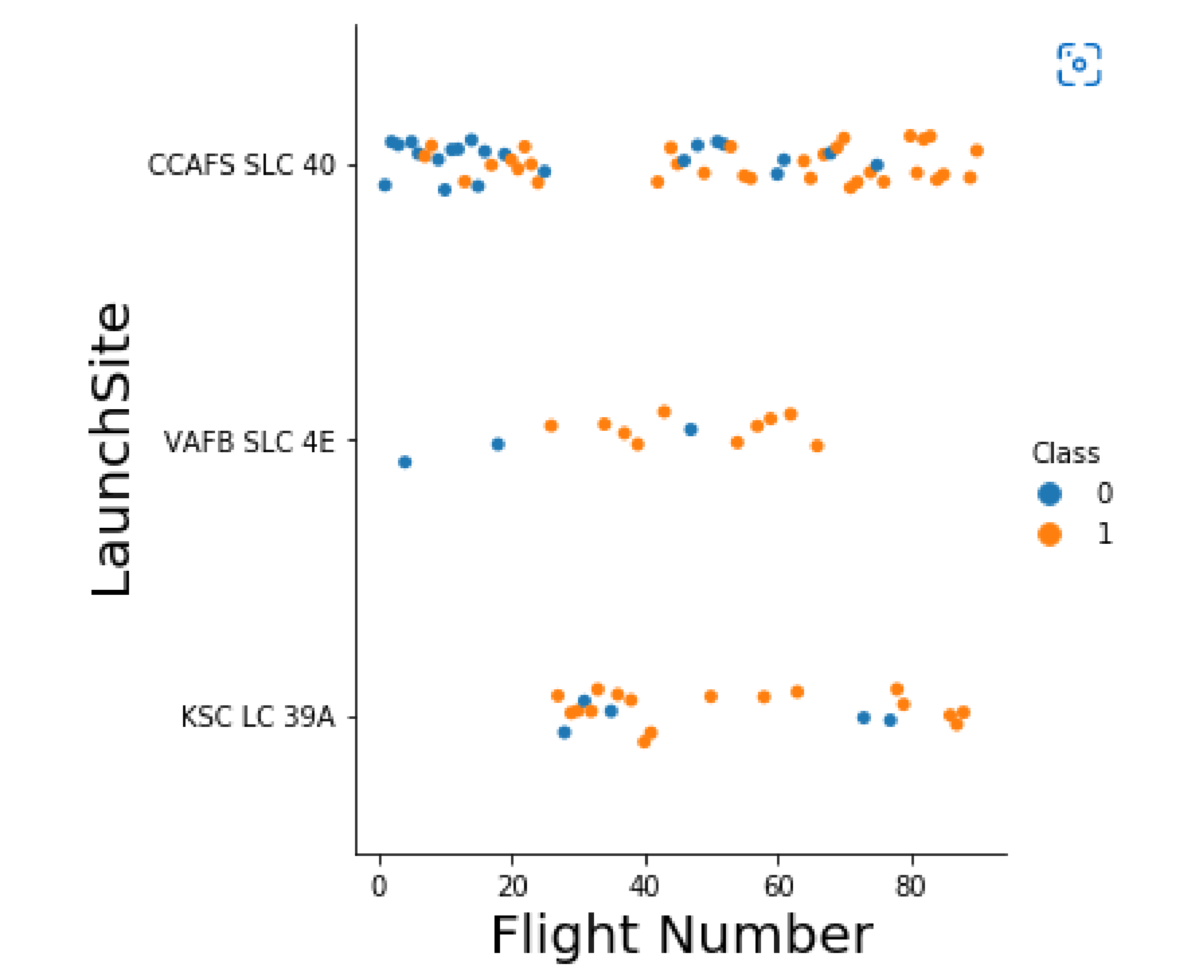
Results

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

Section 2

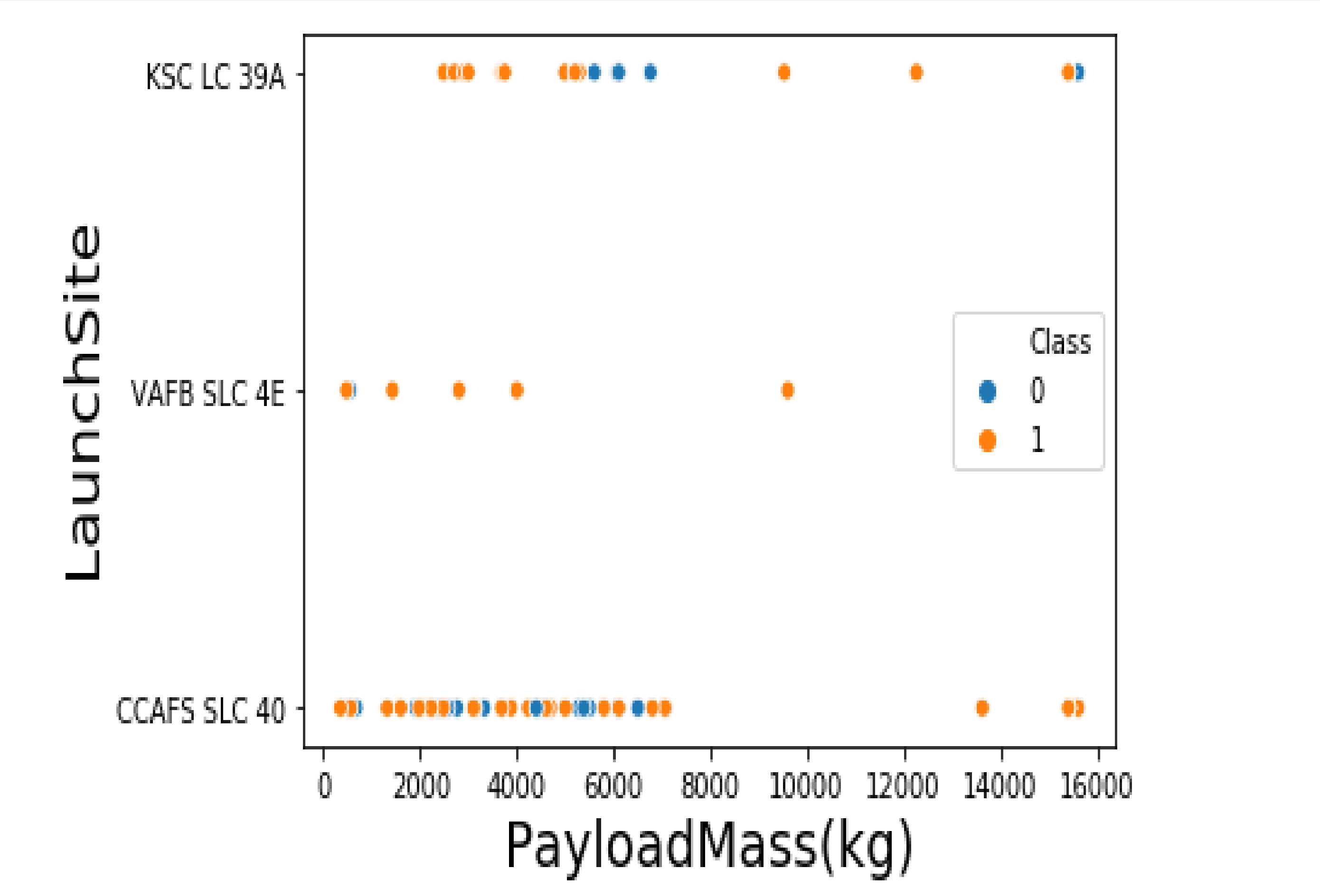
Insights drawn from EDA

Flight Number vs. Launch Site



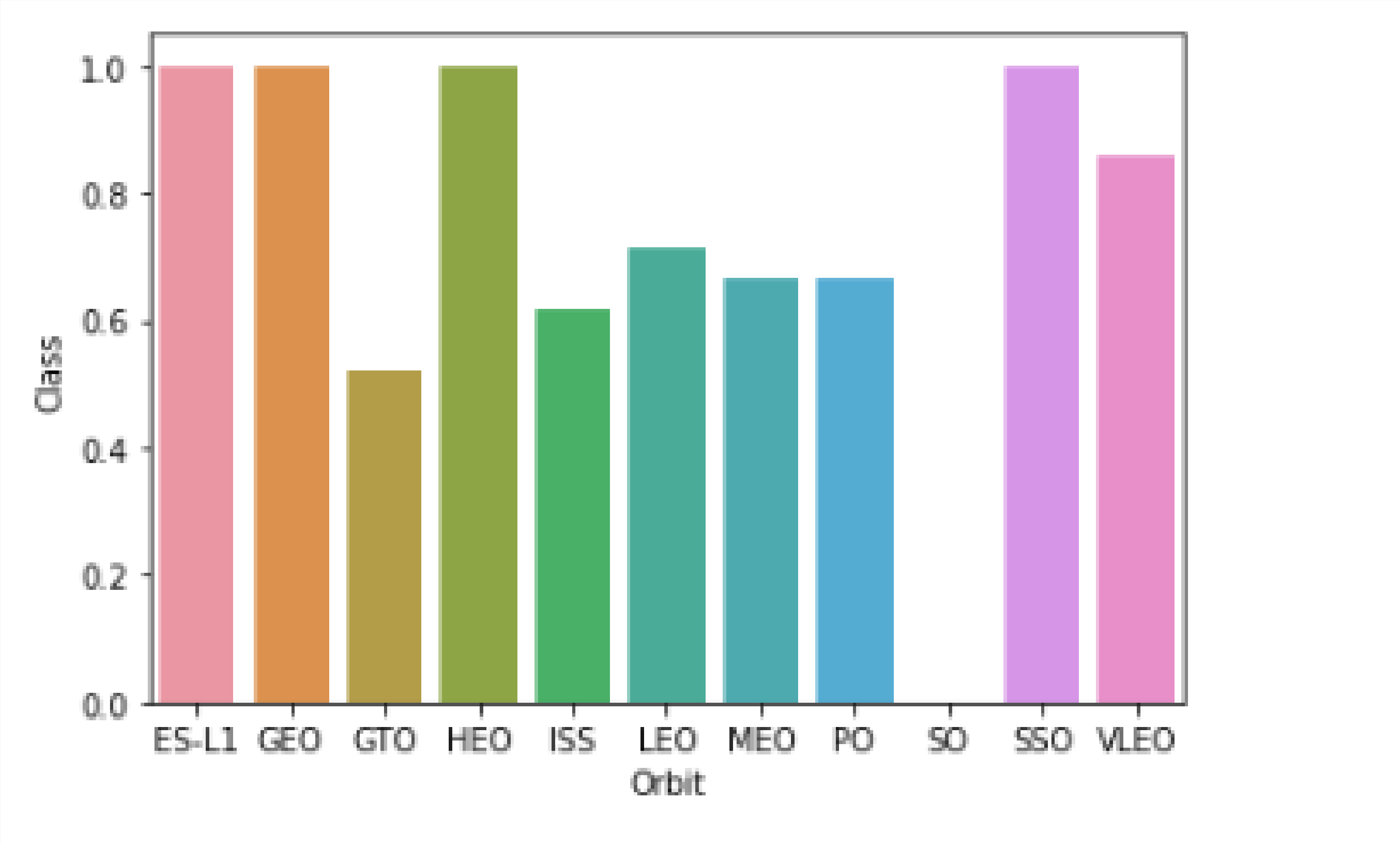
- We found that the larger the flight amount at a launch site, the greater the success rate at a launch site.

Payload vs. Launch Site



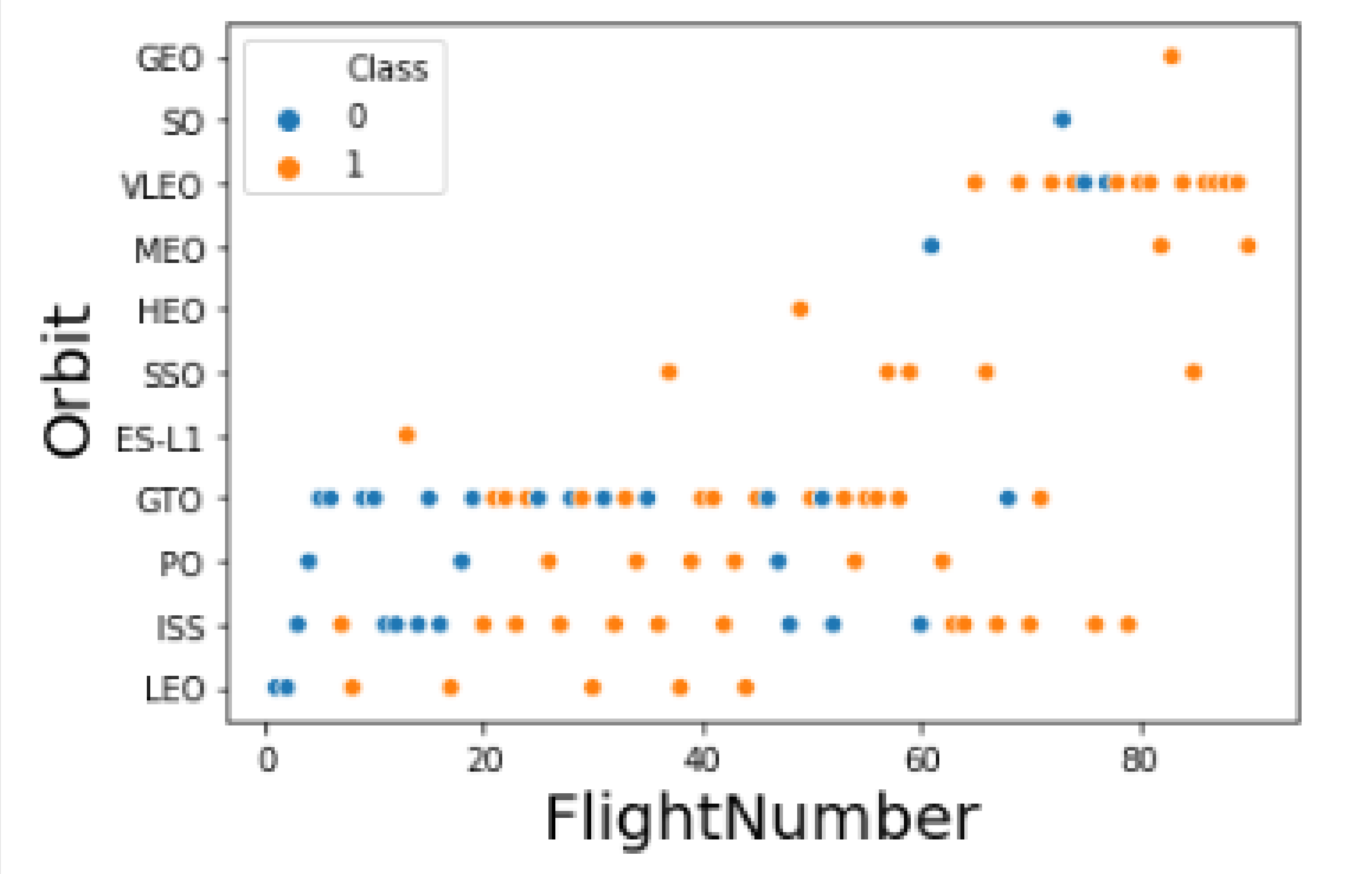
- The greater the payload mass at CCAFS SLC 40 means, the higher success rate for the rocket

Success Rate vs. Orbit Type



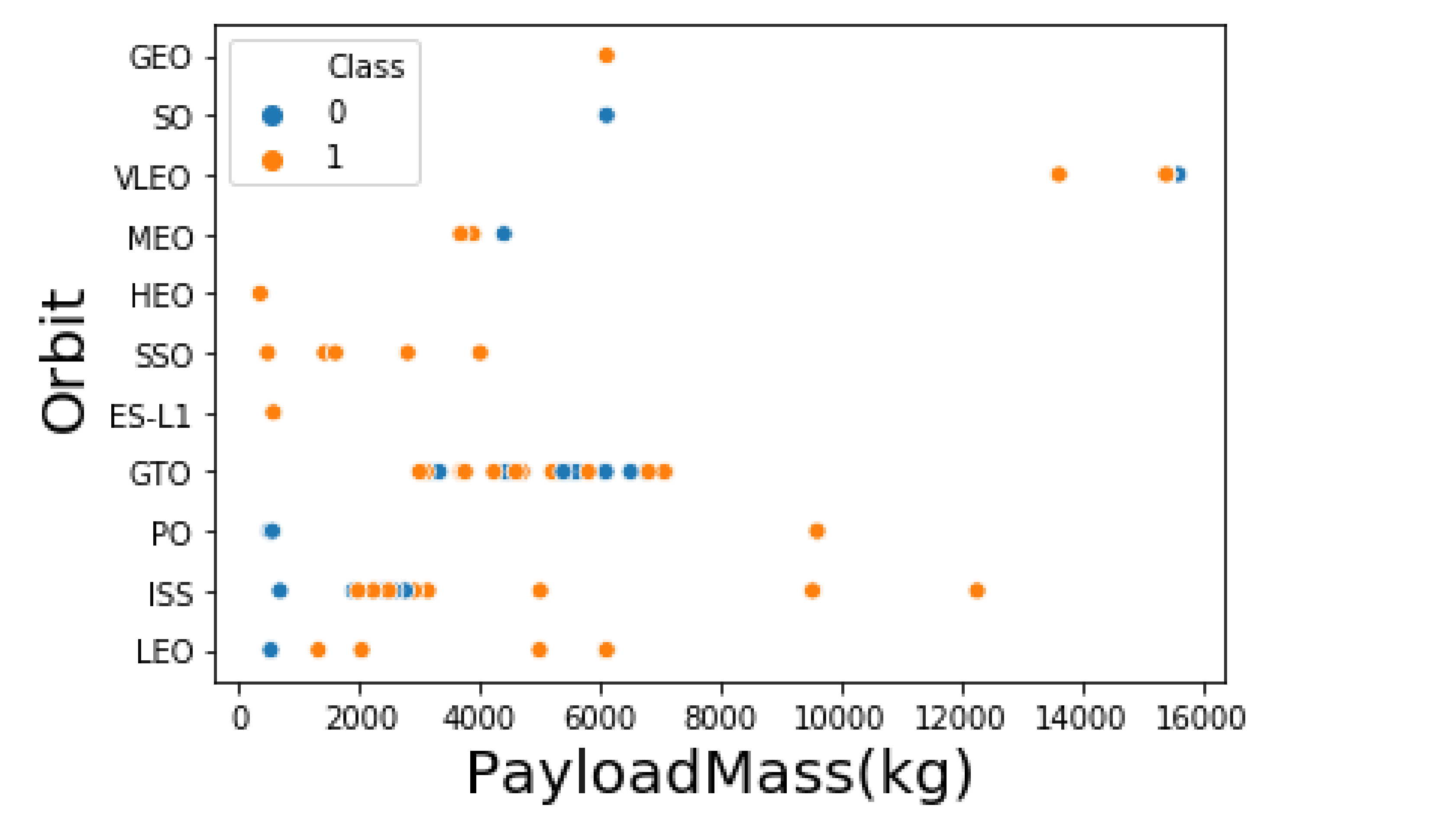
- ES-L1,GEO, SSO, VLEO has the highest rate of success among the other orbit type

Flight Number vs. Orbit Type



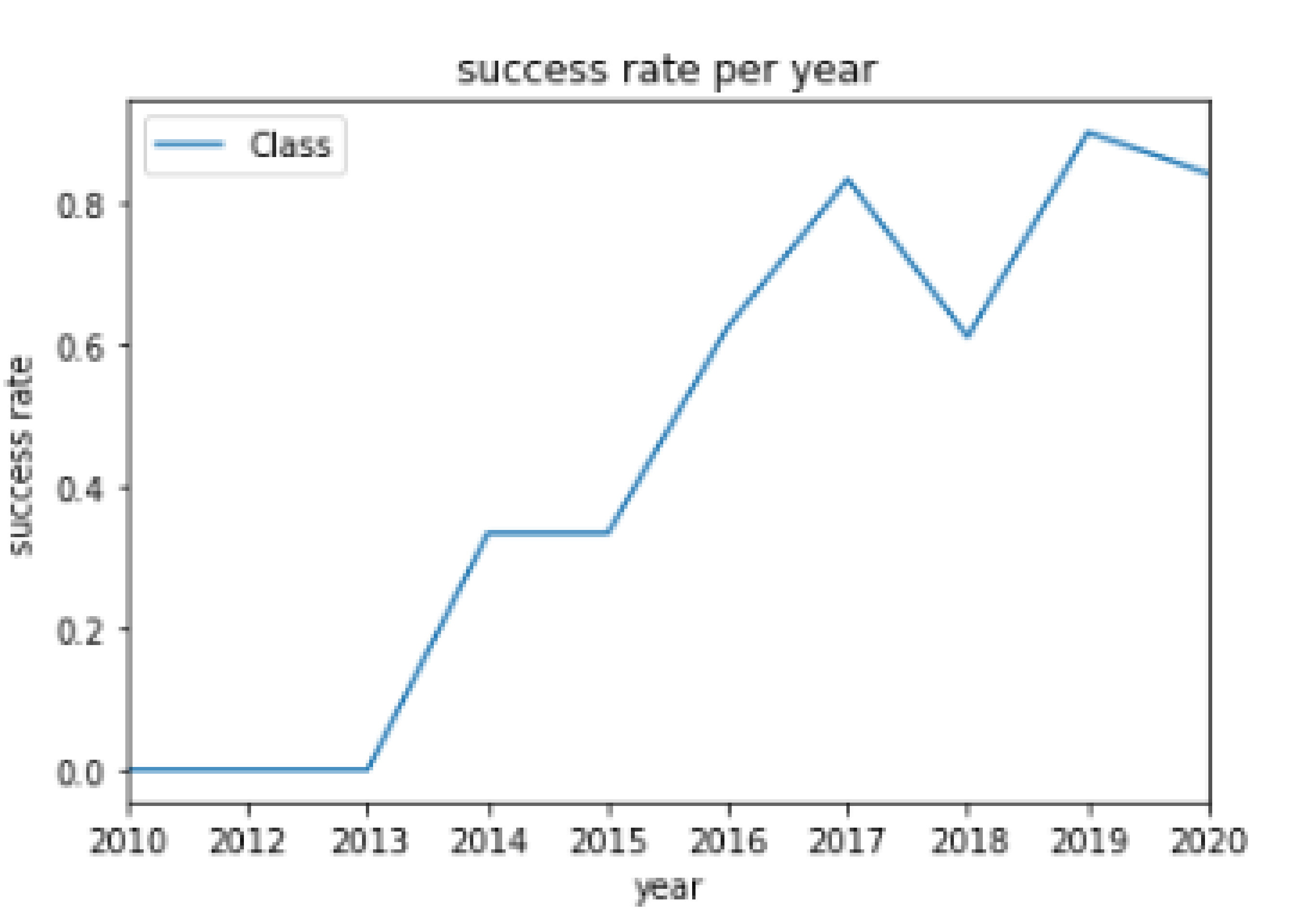
- It can be seen that most of orbit type has no relationship between flight number and the orbit type

Payload vs. Orbit Type



- PO, LEO, ISS had successful landing with heavier load.

Launch Success Yearly Trend



- From the plot, it can be observed that success rate since 2013 kept on increasing till 2020.

All Launch Site Names

- **Using DISTINCT** to show only unique

```
task_1 = '''
        SELECT DISTINCT LaunchSite
        FROM SpaceX
'''
```

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'

- **Using where** to input condition of searching name with ‘CCA’

```
task_2 = '''
        SELECT *
        FROM SpaceX
        WHERE LaunchSite LIKE 'CCA%'
        LIMIT 5
        '''
```

	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

```
task_3 = '''
        SELECT SUM(PayloadMassKG) AS Total_PayloadMass
        FROM SpaceX
        WHERE Customer LIKE 'NASA (CRS)'
        '''
```

	total_payloadmass
0	45596

Average Payload Mass by F9 v1.1

```
task_4 = '''
        SELECT AVG(PayloadMassKG) AS Avg_PayloadMass
        FROM SpaceX
        WHERE BoosterVersion = 'F9 v1.1'
        '''
```

	avg_payloadmass
0	2928.4

First Successful Ground Landing Date

```
task_5 = '''
        SELECT MIN(Date) AS FirstSuccessfull_landing_date
        FROM SpaceX
        WHERE LandingOutcome LIKE 'Success (ground pad)'
        '''
```

	firstsuccessfull_landing_date
0	2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

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

	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

```
task_71 = '''
        SELECT COUNT(MissionOutcome) AS SuccessOutcome
        FROM SpaceX
        WHERE MissionOutcome LIKE 'Success%'
        '''

task_72 = '''
        SELECT COUNT(MissionOutcome) AS FailureOutcome
        FROM SpaceX
        WHERE MissionOutcome LIKE 'Failure%'
        '''
print('The total number of successful mission outcome is:')

print('The total number of failed mission outcome is:')
```

	successoutcome
0	100

	failureoutcome
0	1

Boosters Carried Maximum Payload

```
task_8 = '''
        SELECT BoosterVersion, PayloadMassKG
        FROM SpaceX
        WHERE PayloadMassKG = (
                                SELECT MAX(PayloadMassKG)
                                FROM SpaceX
                                )
        ORDER BY BoosterVersion
        '''
```

	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

```
task_9 = '''
        SELECT BoosterVersion, LaunchSite, LandingOutcome
        FROM SpaceX
        WHERE LandingOutcome LIKE 'Failure (drone ship)'
            AND Date BETWEEN '2015-01-01' AND '2015-12-31'
        '''
```

	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

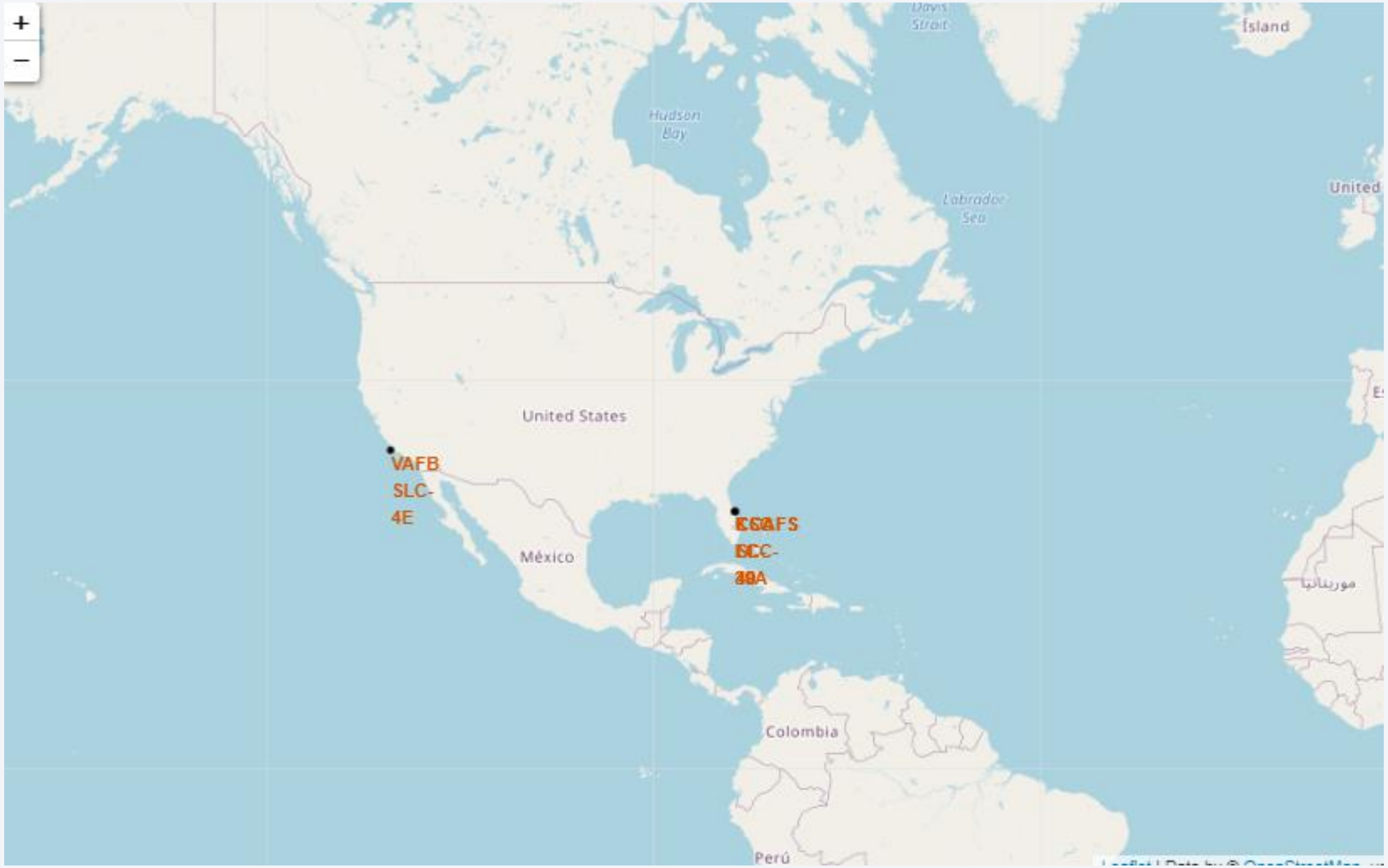
```
task_10 = '''
        SELECT LandingOutcome, COUNT(LandingOutcome)
        FROM SpaceX
        WHERE DATE BETWEEN '2010-06-04' AND '2017-03-20'
        GROUP BY LandingOutcome
        ORDER BY COUNT(LandingOutcome) DESC
        '''
```

	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

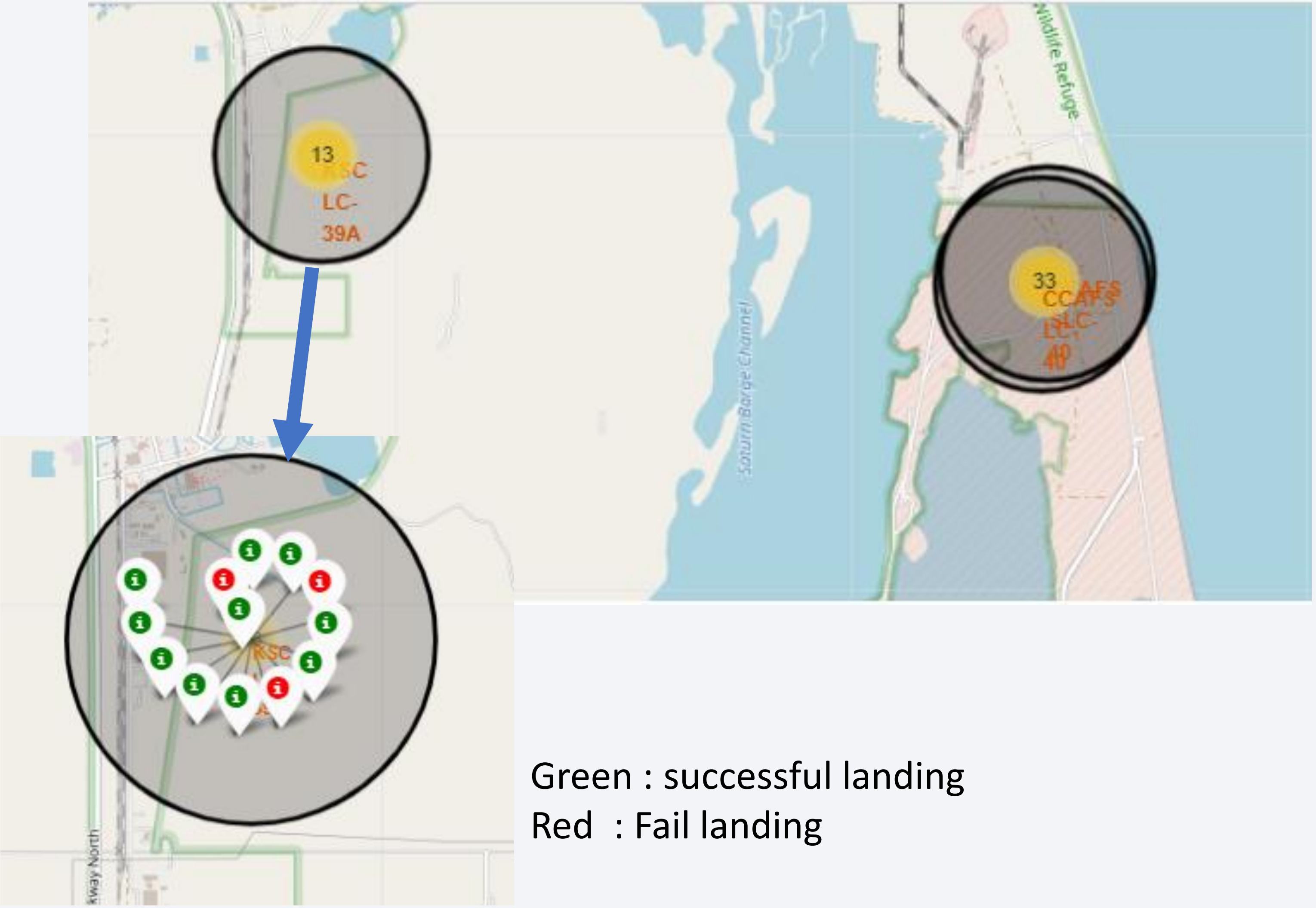
Section 3

Launch Sites Proximities Analysis

All Launch site on Map



success/failed launches for each site on the map



Green : successful landing
Red : Fail landing

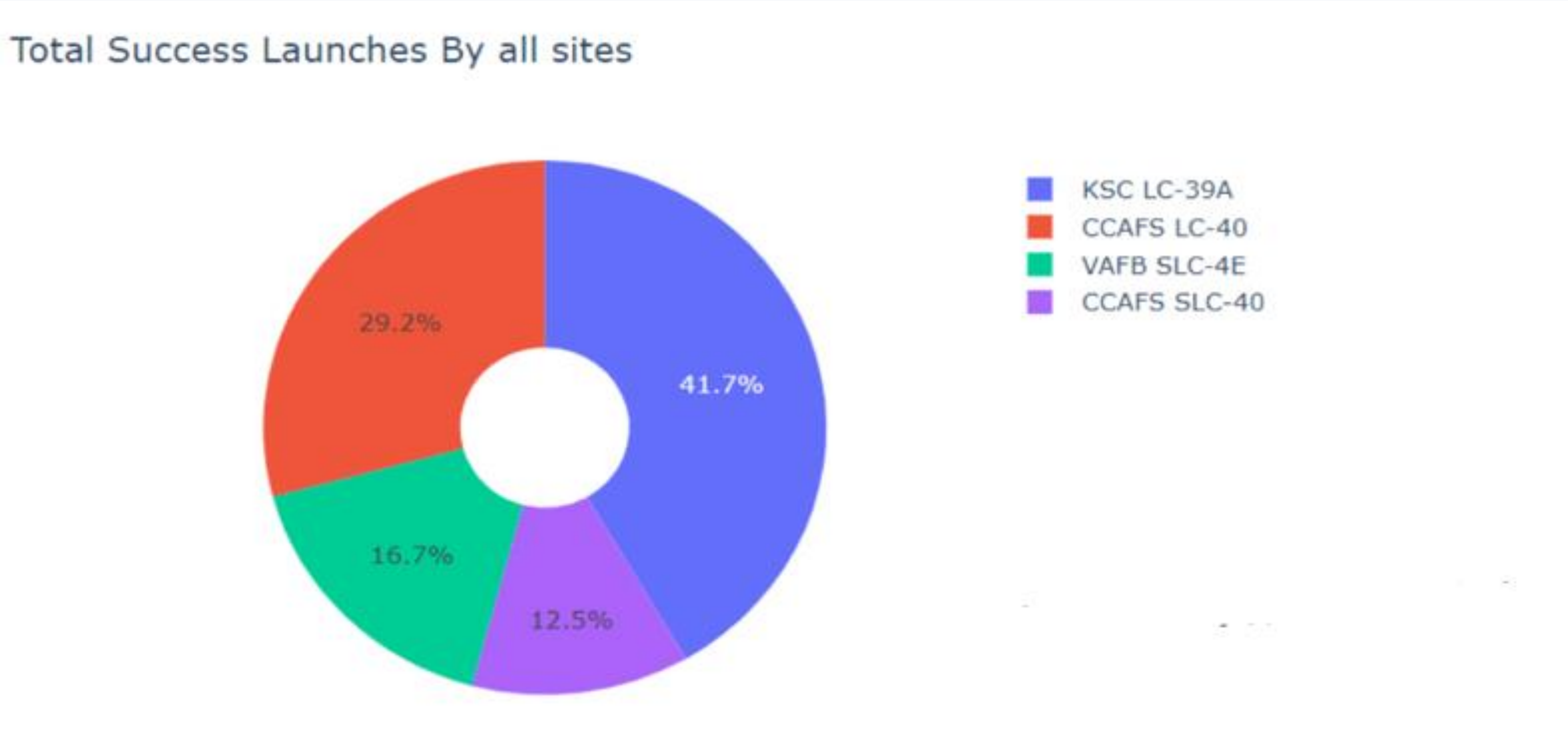
Launch Site distance to landmarks

The blue line show that the launch site is near the coastline around 900 meter from the launch site

Section 4

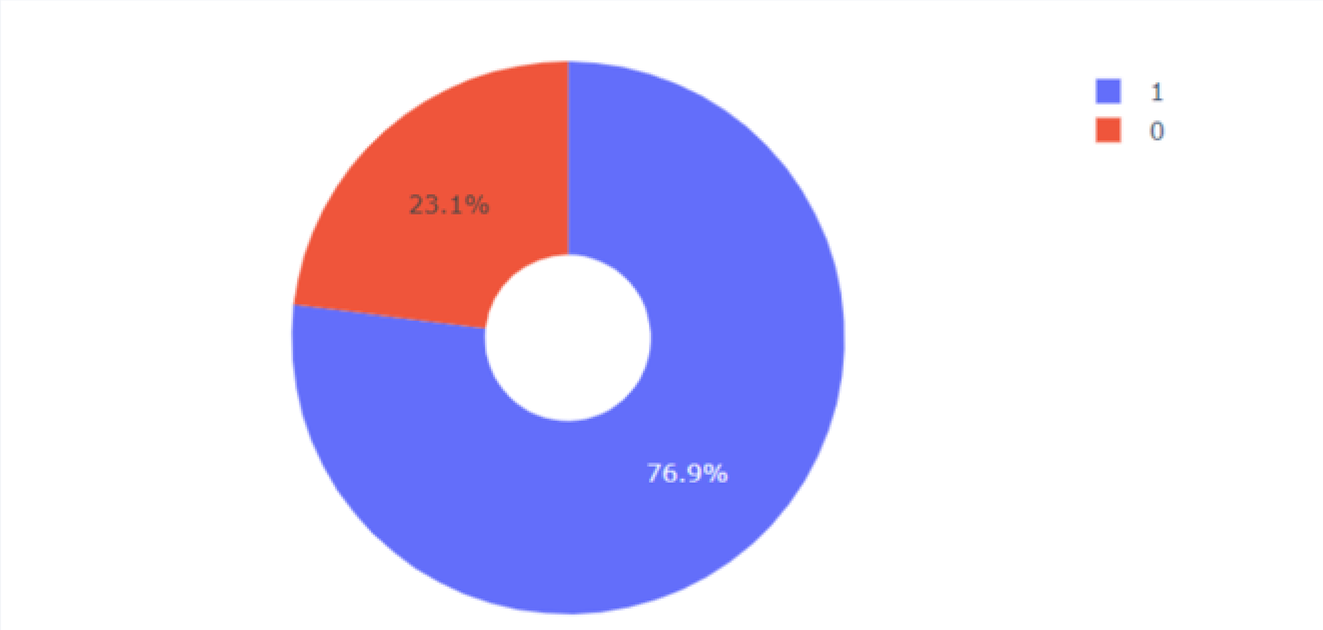
Build a Dashboard with Plotly Dash

Total Success Launches by all sites



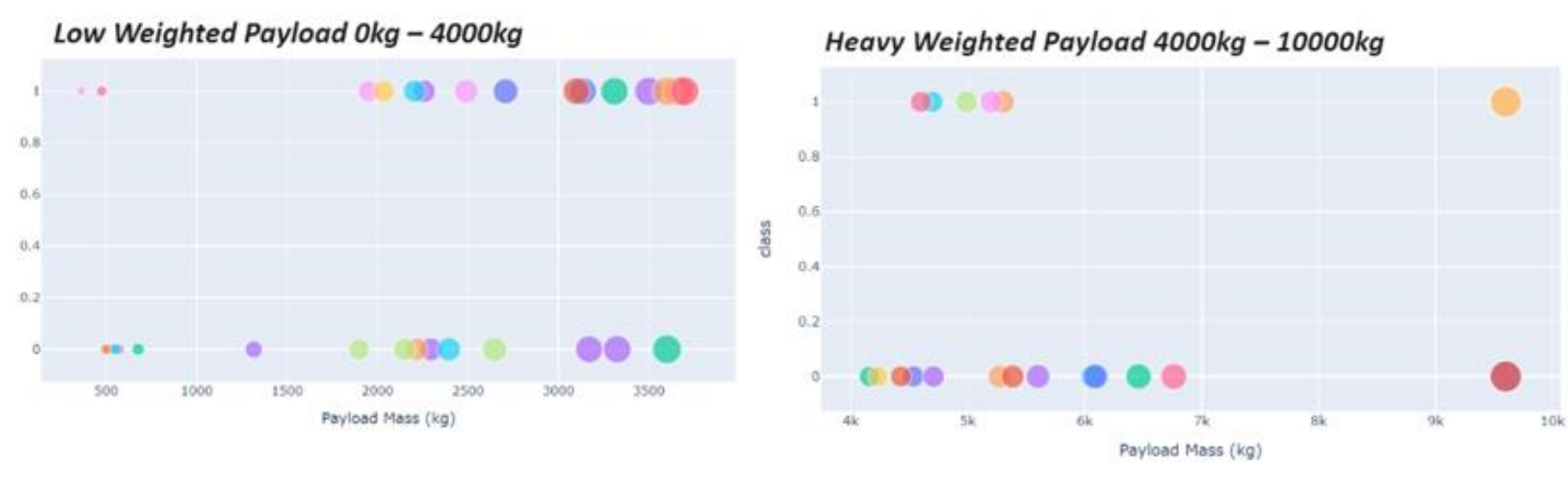
It can see that KSC LC-39A had the highest successful launches among all sites.

Launch site with the highest launch success ratio



- The pie chart show ratio of success and fail launch of KSC LC-39A site.

Scatter plot of payload vs launch outcome comparing between low and heavy payload



It can be seen that low weighted payload has higher success rates comparing to heavy payload.

Section 5

Predictive Analysis (Classification)

Classification Accuracy

```
models={'KNN':knn_cv.best_score_,
         'Logre':logreg_cv.best_score_,
        'SVM':svm_cv.best_score_,
         'Tree':tree_cv.best_score_,
         'KNN':knn_cv.best_score_
        }
bestalgo=max(models, key=models.get)
print ('Best model is ',bestalgo,'score',models[bestalgo])
```

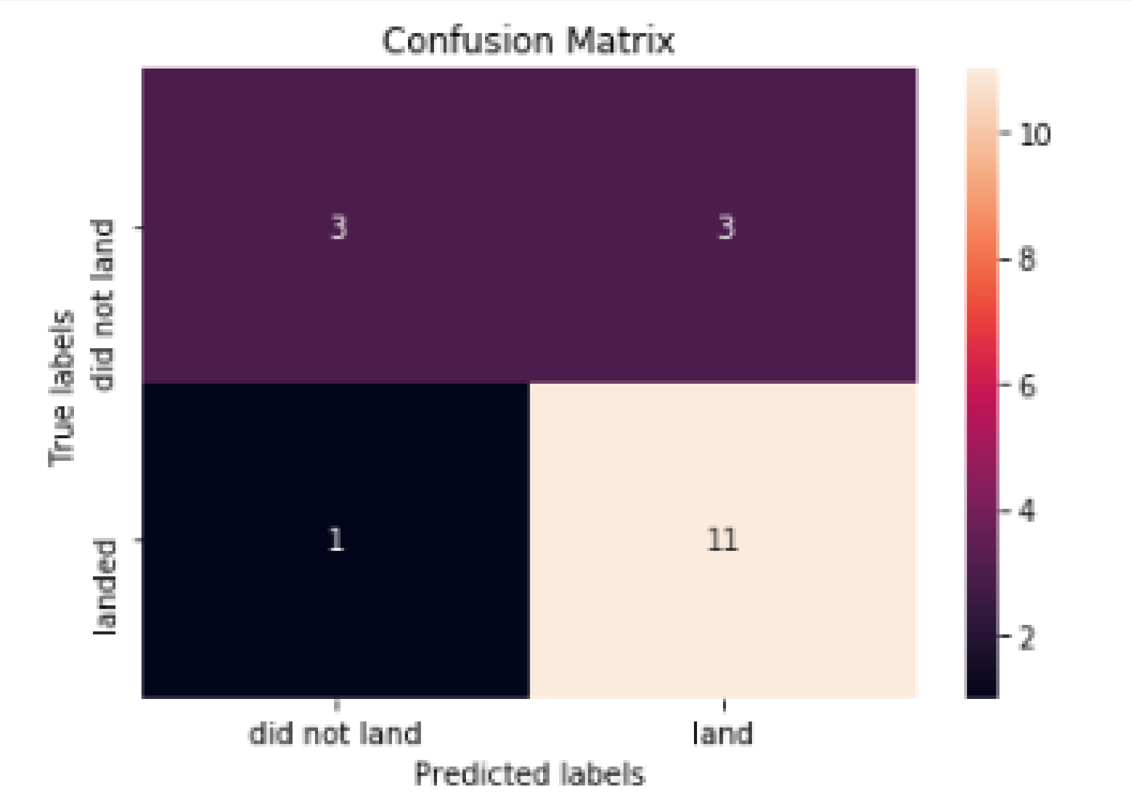
```
Best model is  Tree score 0.875
```

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

```
tuned hpyerparameters :(best parameters)  {'criterion': 'gini', 'max_depth': 4, 'max_features': 'sqrt', 'min_samples_leaf': 1,
'min_samples_split': 2, 'splitter': 'random'}
accuracy : 0.875
```

Confusion Matrix

- Show the confusion matrix of decision tree classifier show the result of prediction compare to true result. It can distinguish between the different classes .



Conclusions

To summarize,

1. The larger number of flights at launch site, the greater success rate at launch site
2. The success rate tend to increase year by year. Reference from 2013-2020
3. Launch at the orbit type ES-L1,GEO, SSO, VLEO has the highest rate of success among the other orbit type
4. KSC LC-39A had the most successful launches of any sites
5. The best classifier for this task is decision tree

Appendix

- Include any relevant assets like Python code snippets, SQL queries, charts, Notebook outputs, or data sets that you may have created during this project

Thank you!