IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

<Bohdan Karpinskyy>
<2023-01-23>

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
  - Exploratory Data Analysis result
  - Interactive analytics in screenshots
  - Predictive Analytics result

Introduction

- Project background and context

  Space X 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 Space X 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 space X for a rocket launch.

  Machine learning can be used for prediction of successful landing of the first stage.

- Problems you want to find answers

  What set of parameter have impact on the successful landing?

  How to ensure the successful landing?

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - The data are available online thought the SpaceX REST API. As an alternative, data can be collected from Wikipedia with the webscrapping technique.
- Perform data wrangling
  - Exploratory analysis was proceed and the training labels were assigned to the newly created “Outcome” column to store the outcome in a binary form (1-success, 0-fail).
- 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

1. SpaceX API was used for collecting various types of data which are related to SpaceX launches, such as rocket booster name, launch site, payload, outcome
2. The responses, decoded as JSON, were normalized and transformed into the Pandas dataframe.
3. Data filtering was applied to keep only Falcon9 launches.
4. The missing values in “PayloadMass” were replaced by the average value.

Data Collection – SpaceX API

1. Data reading

```
[6]: spacex_url="https://api.spacexdata.com/v4/launches/past"

[7]: response = requests.get(spacex_url)
```

Check the content of the response

```
[8]: print(response.content)

b'[{"fairings":{"reused":false,"recovery_attempt":false,"r
```

2. Constructing DataFrame

```
[24]: # Create a data from launch_dict
      df = pd.DataFrame(launch_dict)
```

3. Data selection (“Falcon9” only)

```
data_falcon9 = df [df['BoosterVersion']!='Falcon 1']
data_falcon9.describe
```

4. Dealing with missing values

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

# Replace the np.nan values with its mean value
data_falcon9['PayloadMass'] = data_falcon9['PayloadMass'].replace(np.nan, tmp)
print (data_falcon9.isnull().sum())
```

- URL with code : https://drive.google.com/file/d/1Iu6PZ6pnecumNyYcflFBjI21BCTZ75fW/view?usp=share_link

Data Collection - Scraping

1. Web scrapping was applied with BeautifulSoup from the Wikipedia website

```
soup = BeautifulSoup(response, 'html.parser')
```

```
soup = BeautifulSoup(response, 'html.parser')
```

2. The relevant column names from the HTML table header were collected
3. The data frame was create by parsing the launch HTML tables
4. The result was stored in “spacex_web_scraped.csv”

- URL with code : https://drive.google.com/file/d/17nbimRKn5f9GeqTK9Il3XJV6LGjeYt7A/view?usp=share_link

Data Wrangling

1. The preliminary data analysis was processed.
2. The types of landing were labeled as 1 (successful) and 0 (failure).
3. Resulting dataframe was stored to "dataset_part_2.csv" for further data processing

```
df.to_csv("dataset_part_2.csv", index=False)
```

```
df.to_csv("dataset_part_2.csv", index=False)
```

```
[13]: df['Class']=landing_class
      df[['Class']].head(8)
```

[13]:

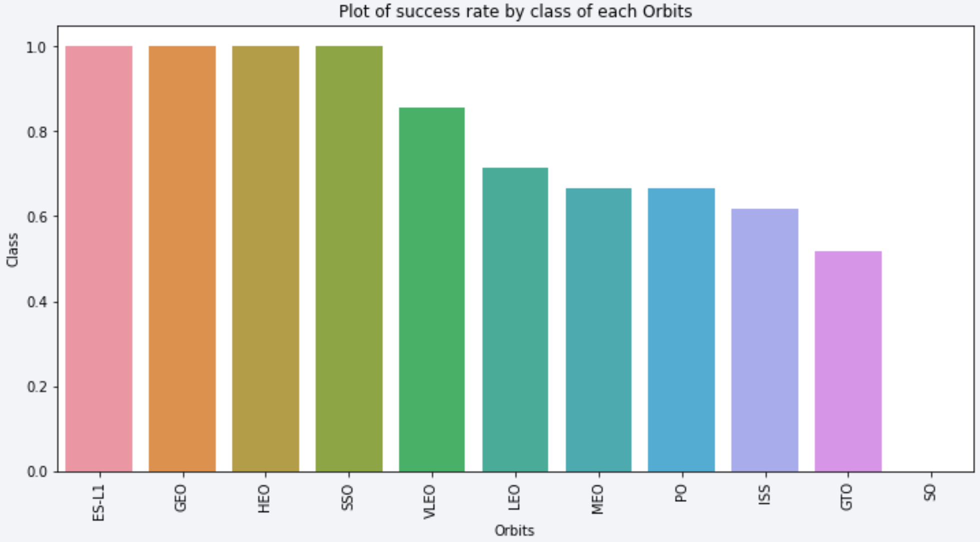
	Class
0	0
1	0
2	0
3	0
4	0
5	0
6	1
7	1

- URL with code : https://drive.google.com/file/d/1tv9jYUtNsukmTa42IGyUR9_Dp4F1PvcT/view?usp=share_link

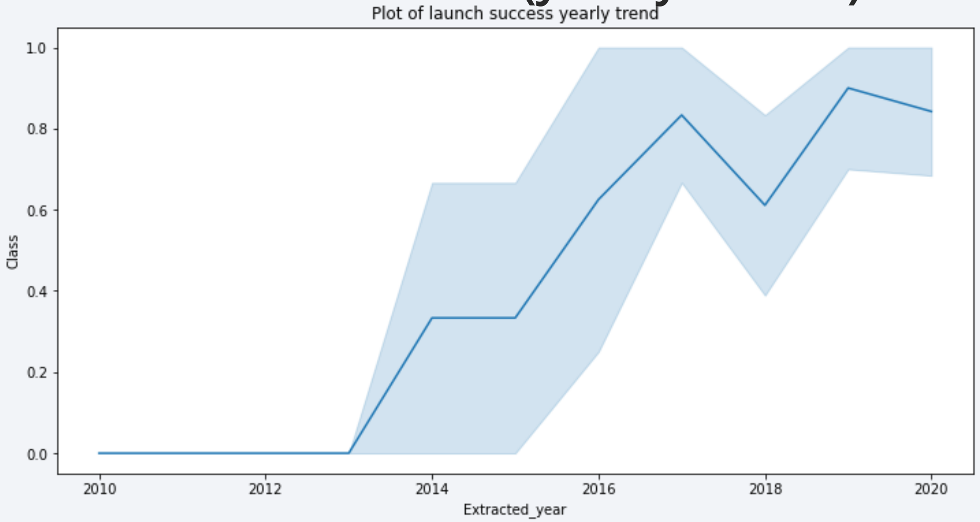
EDA with Data Visualization

Exploratory Data Analysis (Visualization) and Feature Engineering were processed

1. Success rate on each orbit :



2. Launch success (yearly based)



- URL with code : https://drive.google.com/file/d/1y2uTxDMpkn-Cm-Xon5nN9y5q7yKgpGQh/view?usp=share_link

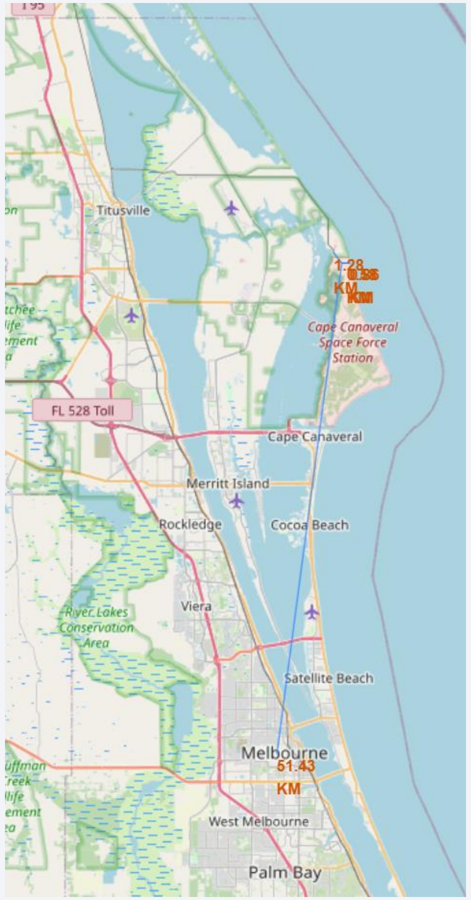
EDA with SQL

- We loaded the SpaceX dataset into a PostgreSQL database without leaving the jupyter notebook.
- 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.
- URL with code : https://drive.google.com/file/d/1McHOxaMxDOog6dEAlmxzE2ZpyY3Qa9qi/view?usp=share_link

Build an Interactive Map with Folium

Analysis of Launch Sites Locations was performed with Folium tool

1. All launch sites were marked on the map
2. The success/failed launches for each site on the map were marked
3. Distances between a launch site to its proximities were calculated



- URL with code : https://drive.google.com/file/d/1hfaiaYsCt5oU3Y1m1I0OMYFnYykA25gF/view?usp=share_link

Build a Dashboard with Plotly Dash

- The interactive dashboard with Plotly dash was build
- The pie charts was plotted to show the total launches by a certain sites
- Scatter graph was plotted that shows relationship with Outcome and Payload Mass (Kg) for the different booster version

- URL with code : https://drive.google.com/file/d/14RZ2hK3fSY6GNbEA-rE3t4drhyjpMFMw/view?usp=share_link

Predictive Analysis (Classification)

Exploratory Data Analysis was processed and the Training Labels were determined

- The class column was created and data were standardized
- Split into training data and test data was performed

The best Hyperparameter for SVM, Classification Trees and Logistic Regression were found.

Decision tree method outperforms the others:

TASK 12

Find the method performs best:

```
print('Accuracy for Logistics Regression method:', logreg_cv.score(X_test, Y_test))
print( 'Accuracy for Support Vector Machine method:', svm_cv.score(X_test, Y_test))
print('Accuracy for Decision tree method:', tree_cv.score(X_test, Y_test))
print('Accuracy for K nearsdt neighbors method:', knn_cv.score(X_test, Y_test))
```

```
Accuracy for Logistics Regression method: 0.8333333333333334
Accuracy for Support Vector Machine method: 0.8333333333333334
Accuracy for Decision tree method: 0.8888888888888888
Accuracy for K nearsdt neighbors method: 0.8333333333333334
```

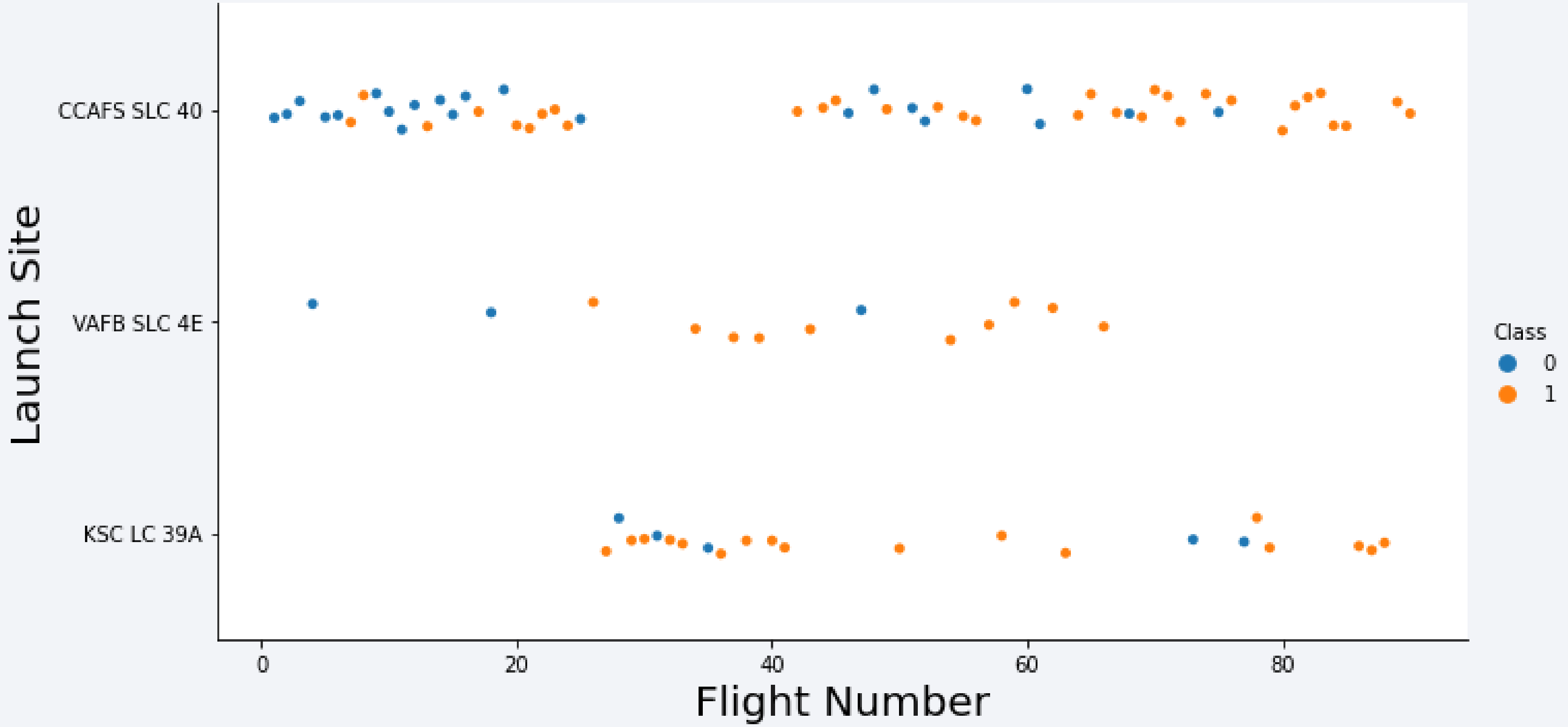
- URL with code : https://drive.google.com/file/d/1A7UeWk2SwrT5W68fCSmR76A3OxyN2NLY/view?usp=sharing

Section 2

Insights drawn from EDA

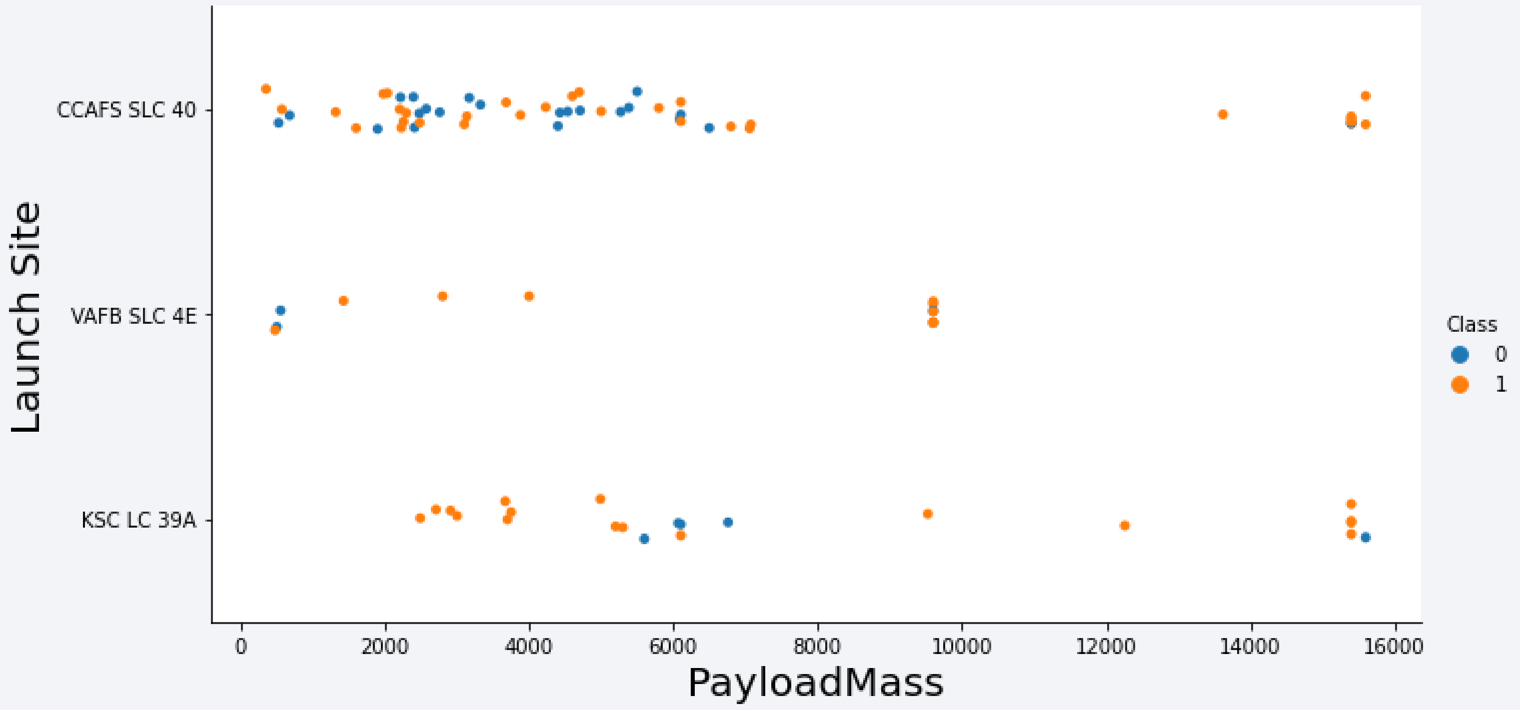
Flight Number vs. Launch Site

A scatter plot of Flight Number vs. Launch Site is below:



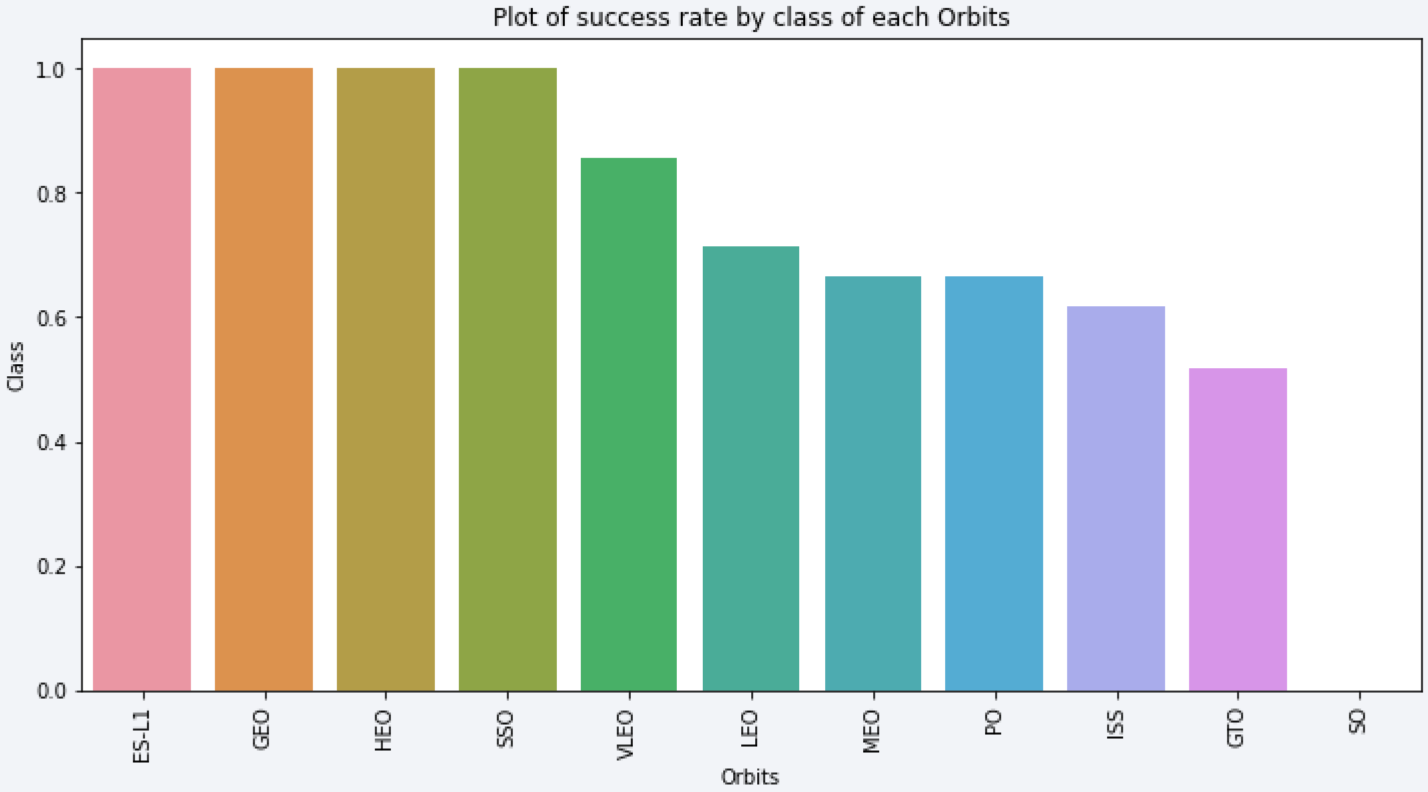
Payload vs. Launch Site

A scatter plot of Payload vs. Launch Site is below:



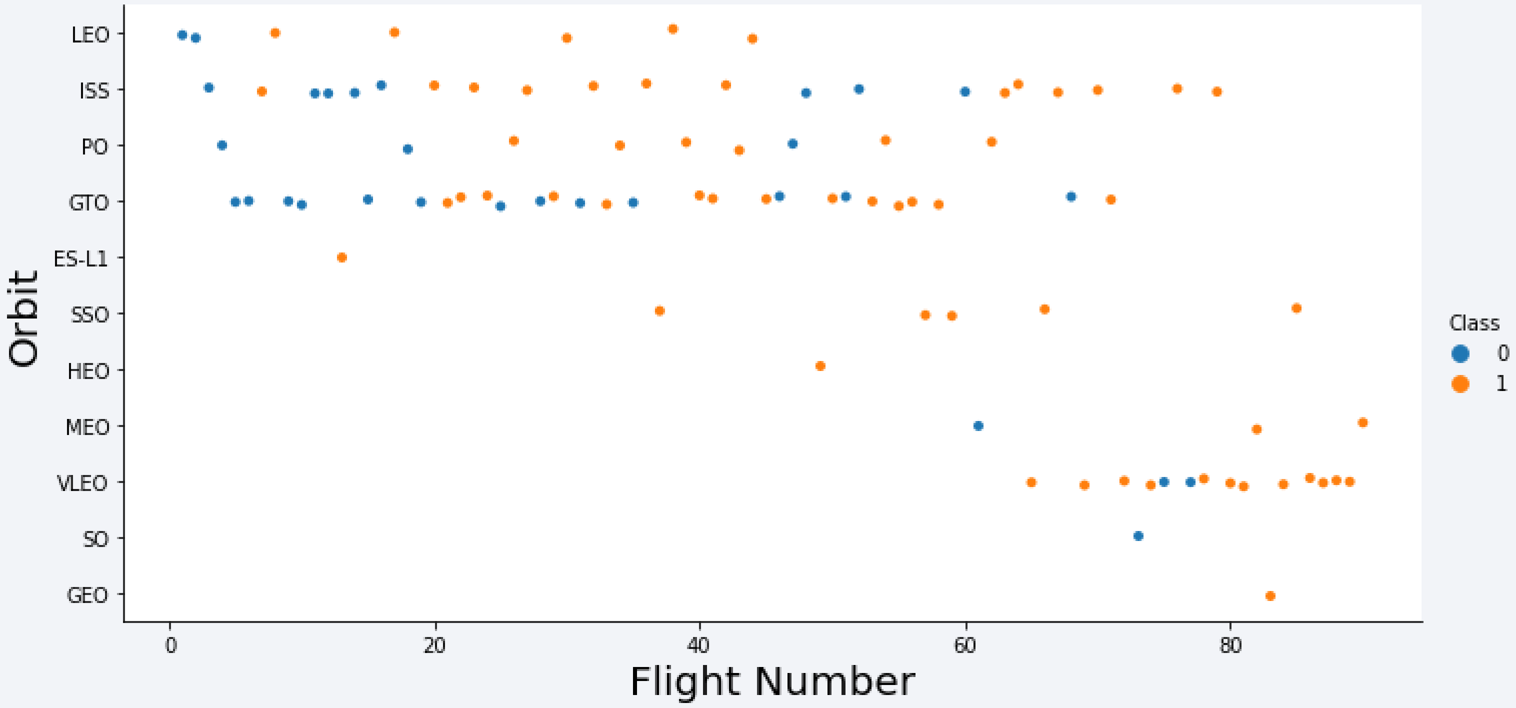
Success Rate vs. Orbit Type

A bar chart for the success rate of each orbit type is below:



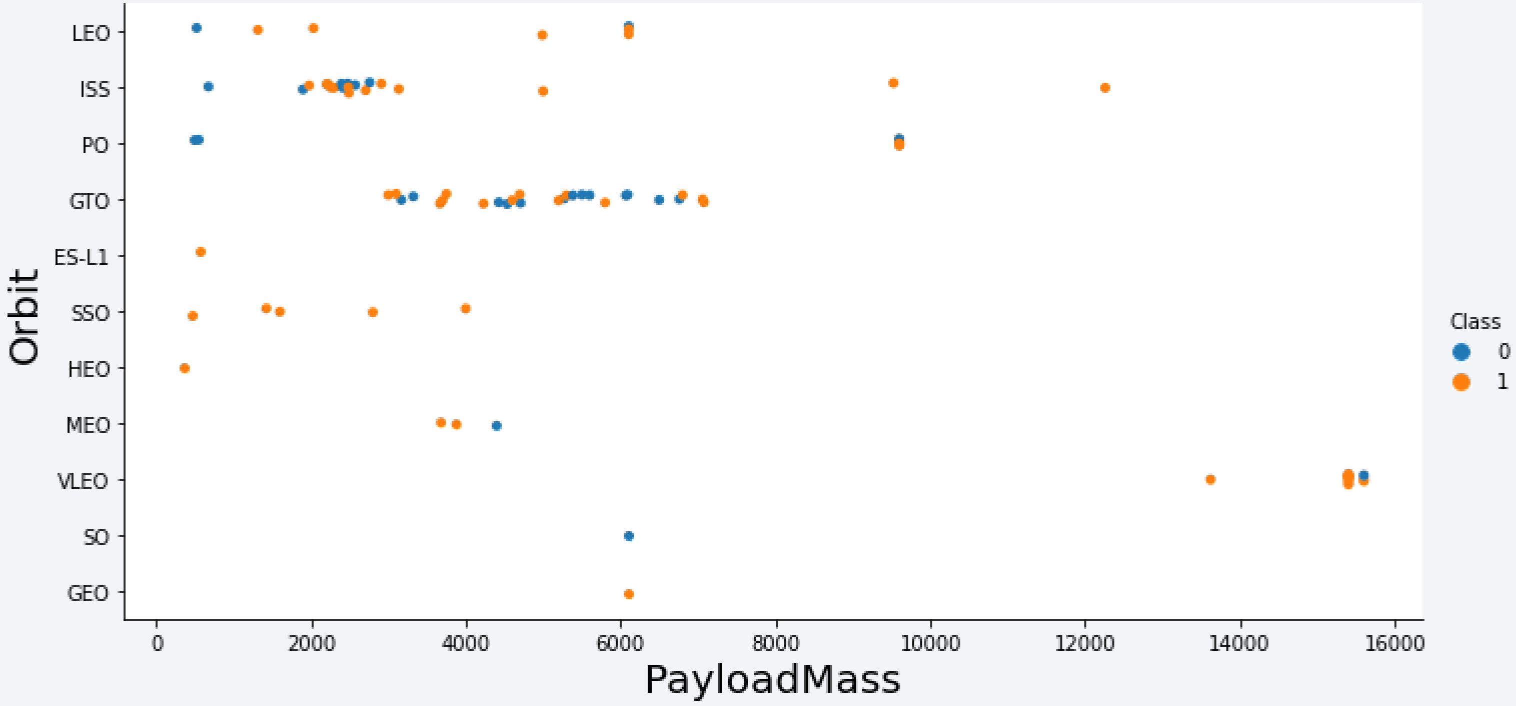
Flight Number vs. Orbit Type

A scatter point of Flight number vs. Orbit type is below:



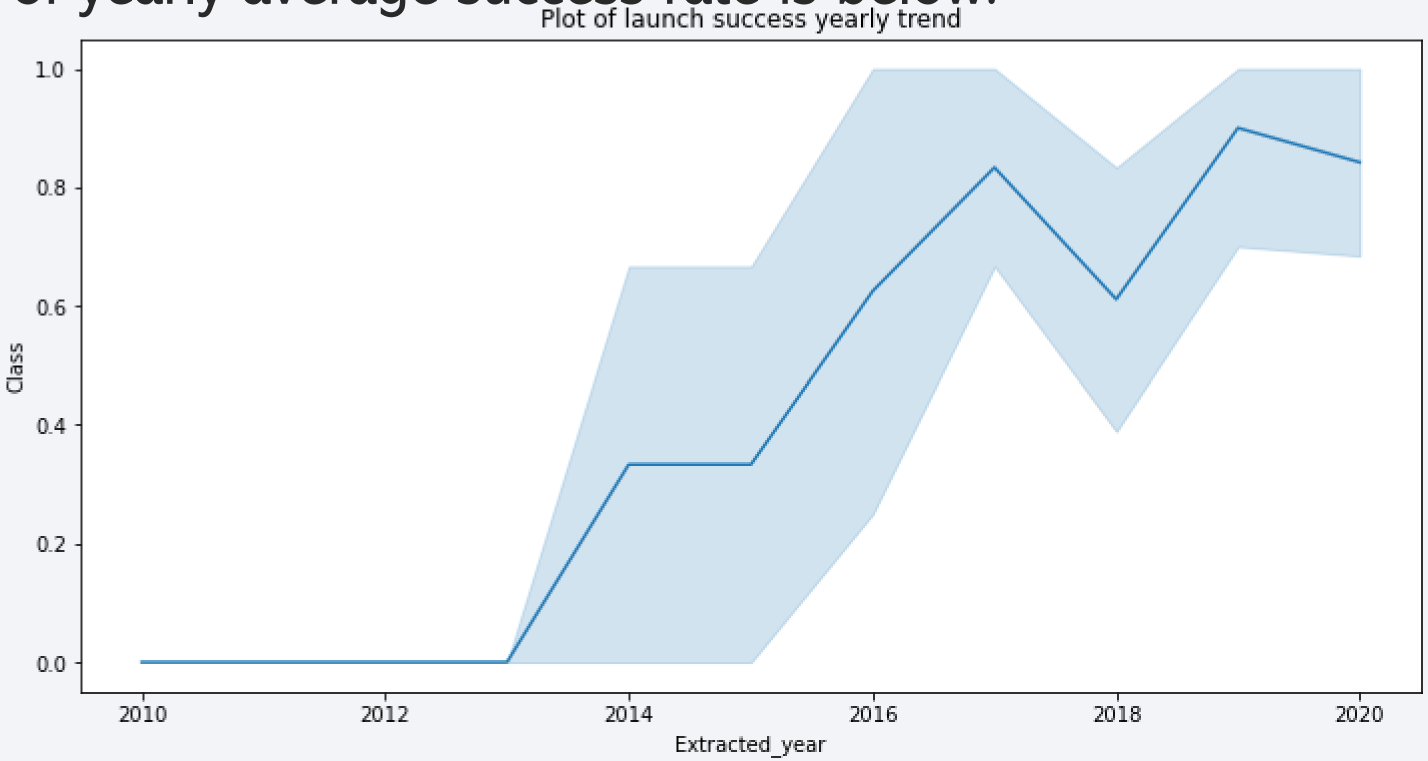
Payload vs. Orbit Type

A scatter point of payload vs. orbit type is below:



Launch Success Yearly Trend

A line chart of yearly average success rate is below:



All Launch Site Names

The unique launch sites were selected with DISTINCT keyword

```
task_1 = '''
          SELECT DISTINCT LaunchSite
          FROM SpaceX
'''
create_pandas_df(task_1, database=conn)
```

The result is

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

Launch Site Names Begin with 'CCA'

The following query

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

was used to display 5 records were launch sites begin with “CCA”

	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 total payload carried by boosters from NASA was calculated with the following query:

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

The outcome of this query is:

	total_payloadmass
0	45596

Average Payload Mass by F9 v1.1

The average payload mass carried by booster version F9 v1.1 was calculated.

The following query was used

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

and the query result is

	avg_payloadmass
0	2928.4

First Successful Ground Landing Date

The dates of the first successful landing outcome on ground pad were observed. The corresponded query is :

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

create_pandas_df(task_5, database=conn)
```

And its outcome is

	firstsuccessfull_landing_date
0	2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

- The list of names of successfully landed boosters on drone ship and payload mass between 4000 and 6000 was selected. The corresponded query is

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

And the query outcome is

	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

The total number of successful and failure mission outcomes was calculated as following:

```
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)
```

```
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

The names of the booster which have carried the maximum payload mass was listed with the following query:

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

The query outcome is

	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

2015 Launch Records

The failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015 were listed with the following query:

```
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)
```

The outcome of this query is

	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 count of landing outcomes (such as Failure (drone ship) or Success (ground pad)) was ranked for period from 2010-06-04 to 2017-03-20 and shown in descending order

```
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)
```

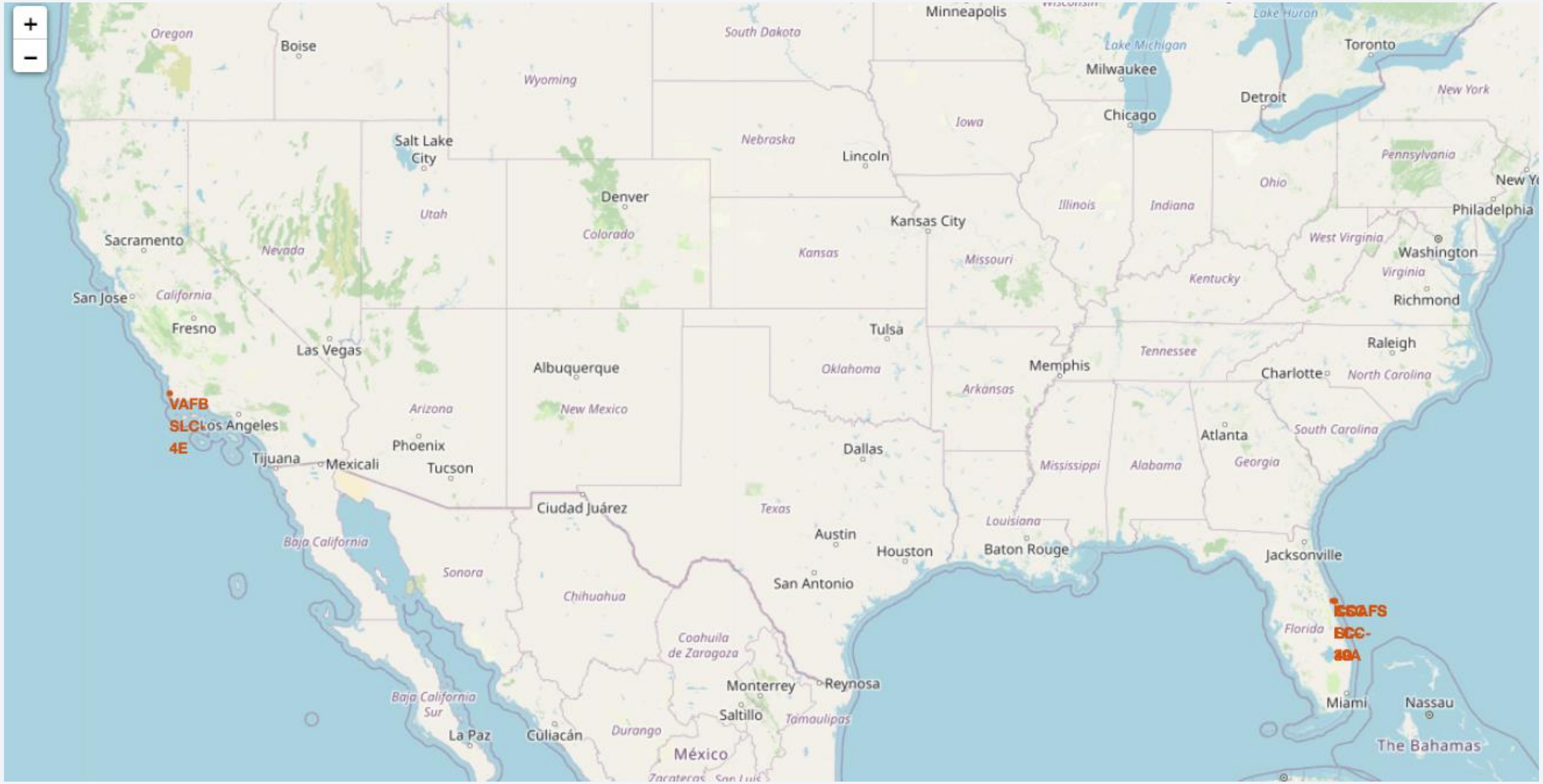
The outcome of this query is:

	landingoutcome	count
0	No attempt	10
1	Success (drone ship)	6
2	Failure (drone ship)	5
3	Success (ground pad)	5

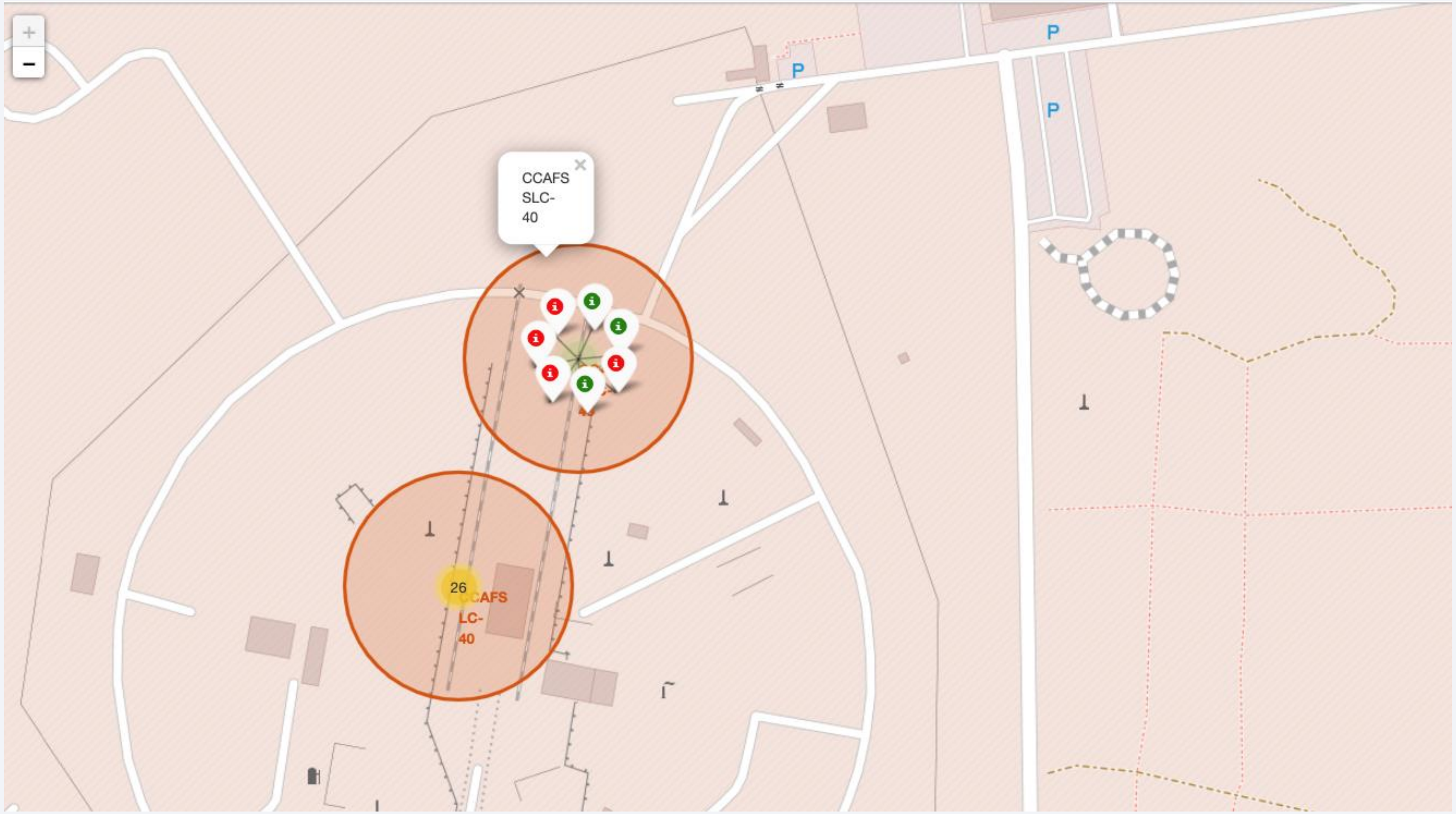
Section 3

Launch Sites Proximities Analysis

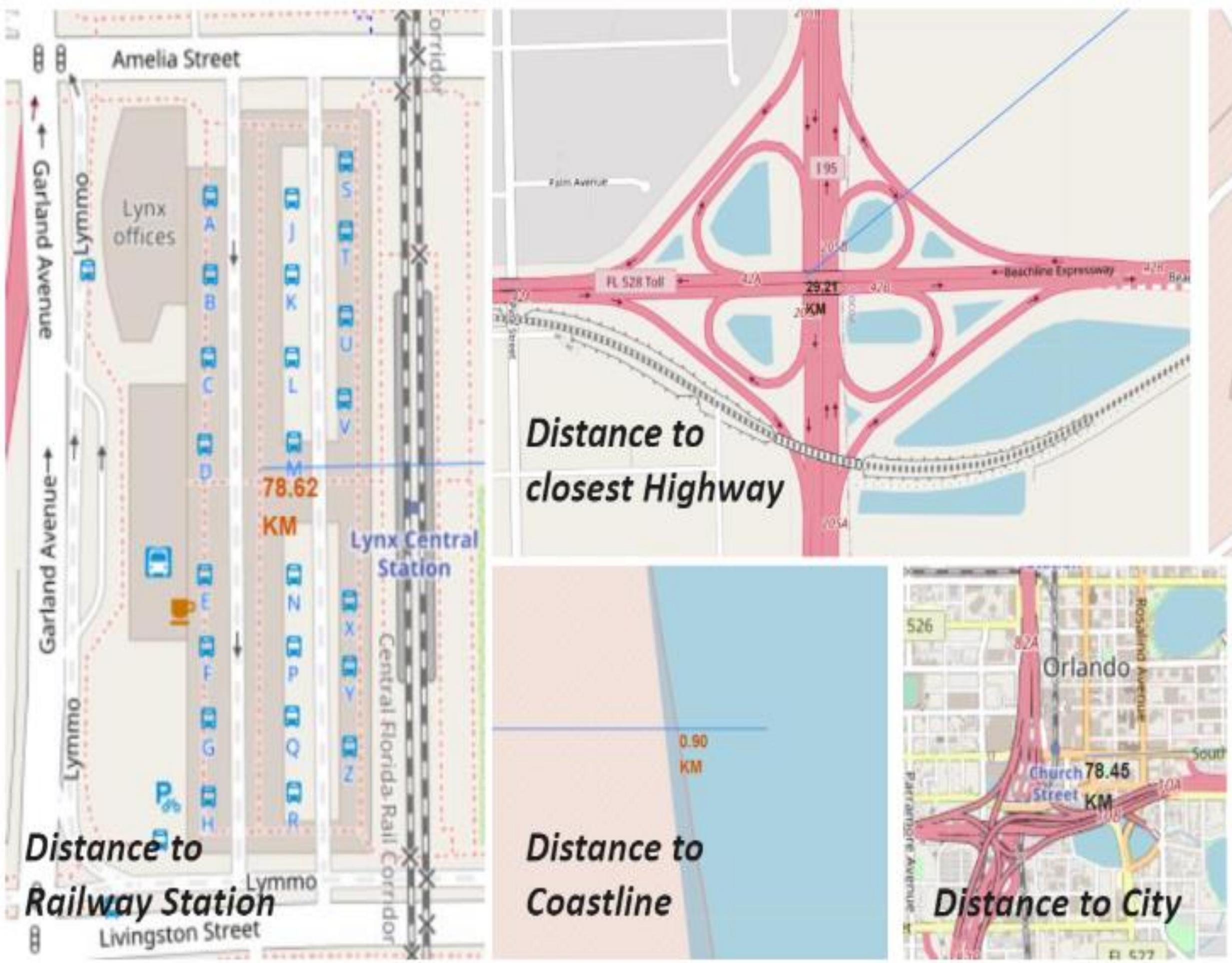
The map of launch sites



The success/failed launches for each site on the map



The distances between a launch site to its proximities

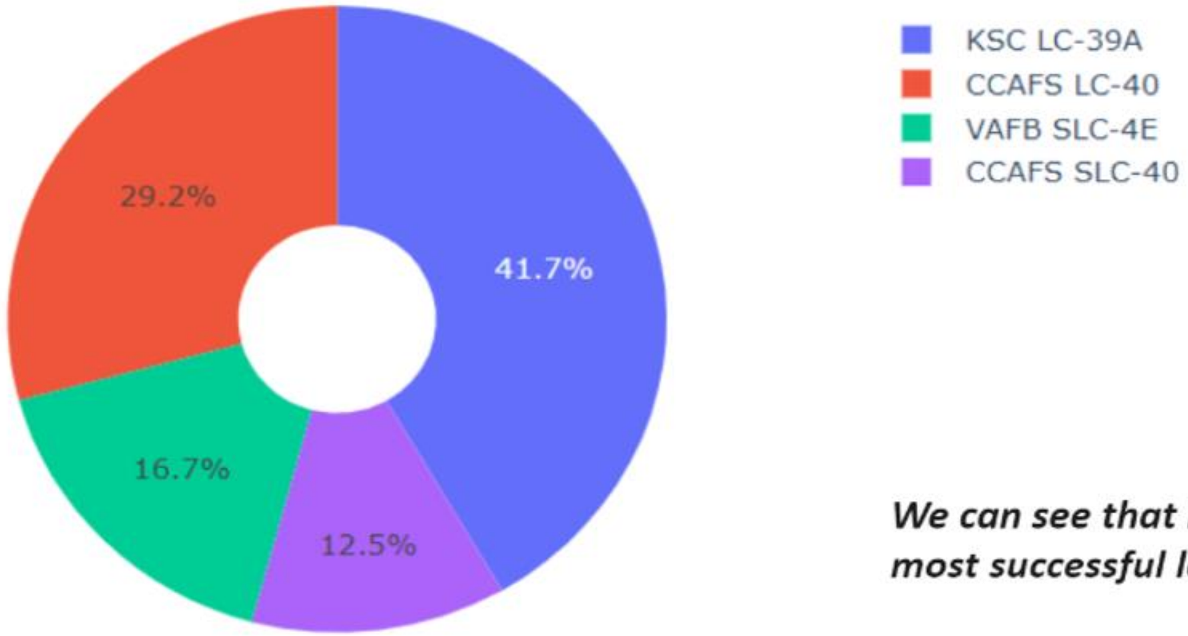


- Are launch sites in close proximity to railways? No
- Are launch sites in close proximity to highways? No
- Are launch sites in close proximity to coastline? Yes
- Do launch sites keep certain distance away from cities? Yes

Section 4

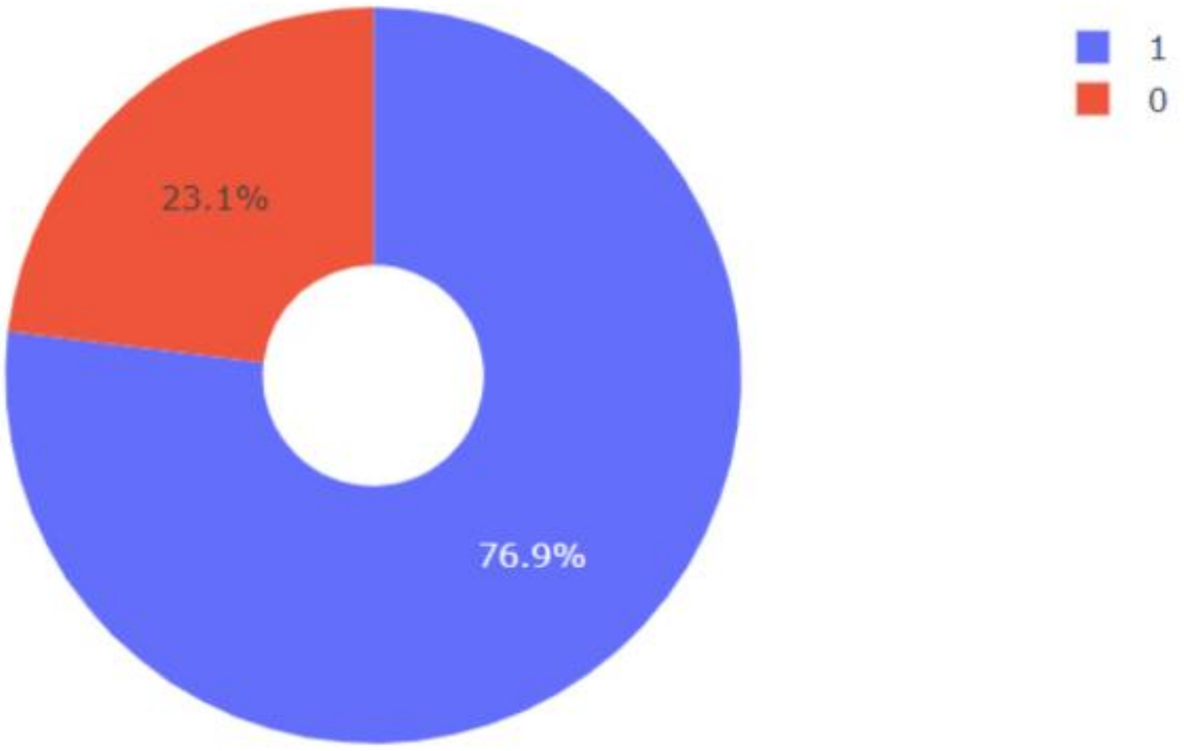
Build a Dashboard with Plotly Dash

The success percentage achieved by each launch site



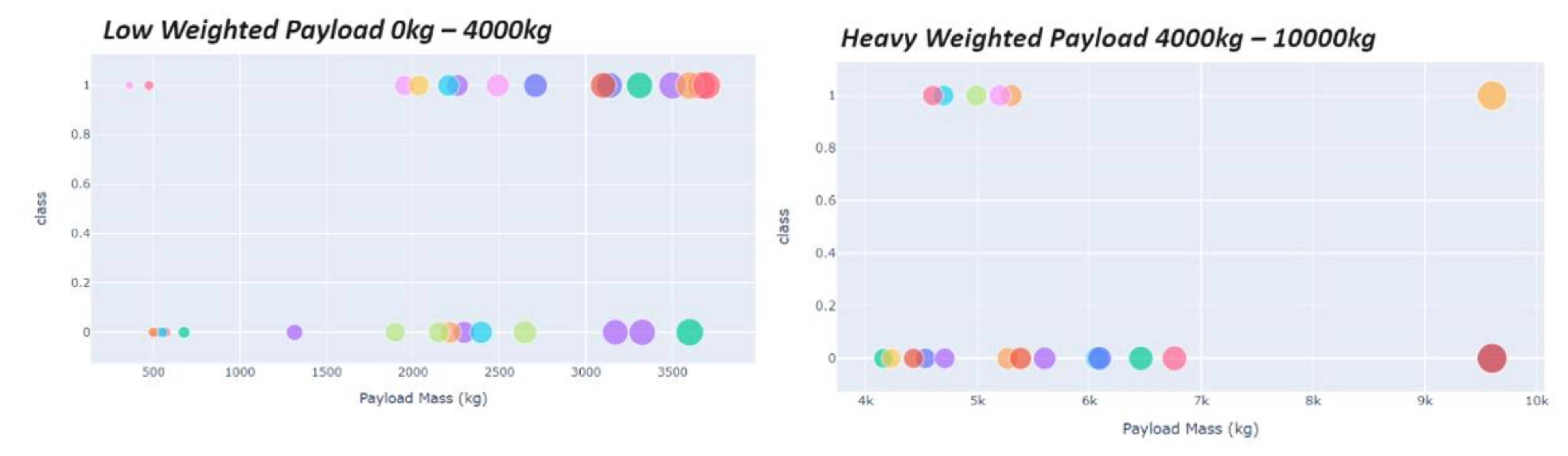
We can see that KSC LC-39A had the most successful launches from all the sites

The Launch site with the highest launch success ratio



KSC LC-39A achieved a 76.9% success rate while getting a 23.1% failure rate

Payload vs Launch Outcome



The success rate for low weight payloads is higher

Section 5

Predictive Analysis (Classification)

Classification Accuracy

Decision tree shows the best prediction outcome

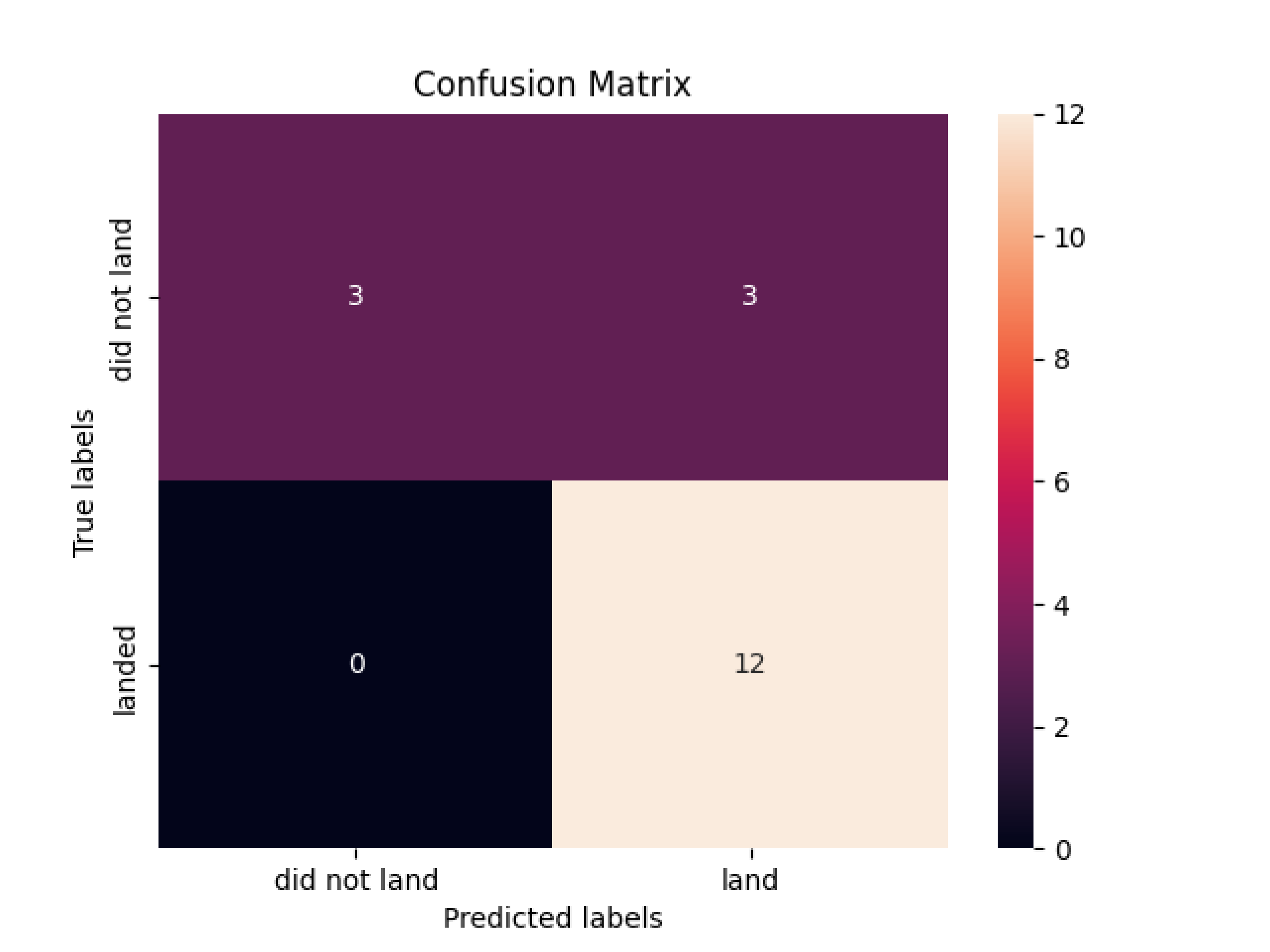
TASK 12

Find the method performs best:

```
print('Accuracy for Logistics Regression method:', logreg_cv.score(X_test, Y_test))
print( 'Accuracy for Support Vector Machine method:', svm_cv.score(X_test, Y_test))
print('Accuracy for Decision tree method:', tree_cv.score(X_test, Y_test))
print('Accuracy for K nearsdt neighbors method:', knn_cv.score(X_test, Y_test))
```

```
Accuracy for Logistics Regression method: 0.8333333333333334
Accuracy for Support Vector Machine method: 0.8333333333333334
Accuracy for Decision tree method: 0.8888888888888888
Accuracy for K nearsdt neighbors method: 0.8333333333333334
```

Confusion Matrix



Conclusions

- There is a positive correlation between the success rate at a launch site and the amount of flights at a launch site
- The success of launches is started from 2013.
- The following orbits have the highest success rate: ES-L1, GEO, HEO, SSO, VLEO.
- KSC LC-39A site is the most successful among the launching sites.
- The outcome of the Decision tree classifier is the best for this dataset.

Appendix

- IBM_DS_assigments are also available in Guthub: https://github.com/kDaniu/IBM_DS_assigments

Thank you!