IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

<Yilin Yu>
<2024.11.26>

Outline

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

Executive Summary

- Summary of methodologies
  - Data Collection - SpaceX REST API
  - Data Analysis - Visualization, Determine Training Labels, and Launch Sites Locations Analysis with Folium
  - Data modelling - Machine Learning Algorithm and Evaluation
- Summary of all results
  - Logistic Regression, Decision Tree Classifier, and K-Nearest Neighbors all have approximately 83.33% accuracy - indicates predictive results are recommend to believe.

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
- Problems you want to find answers
  - 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.

Section 1

Methodology

Methodology

Executive Summary

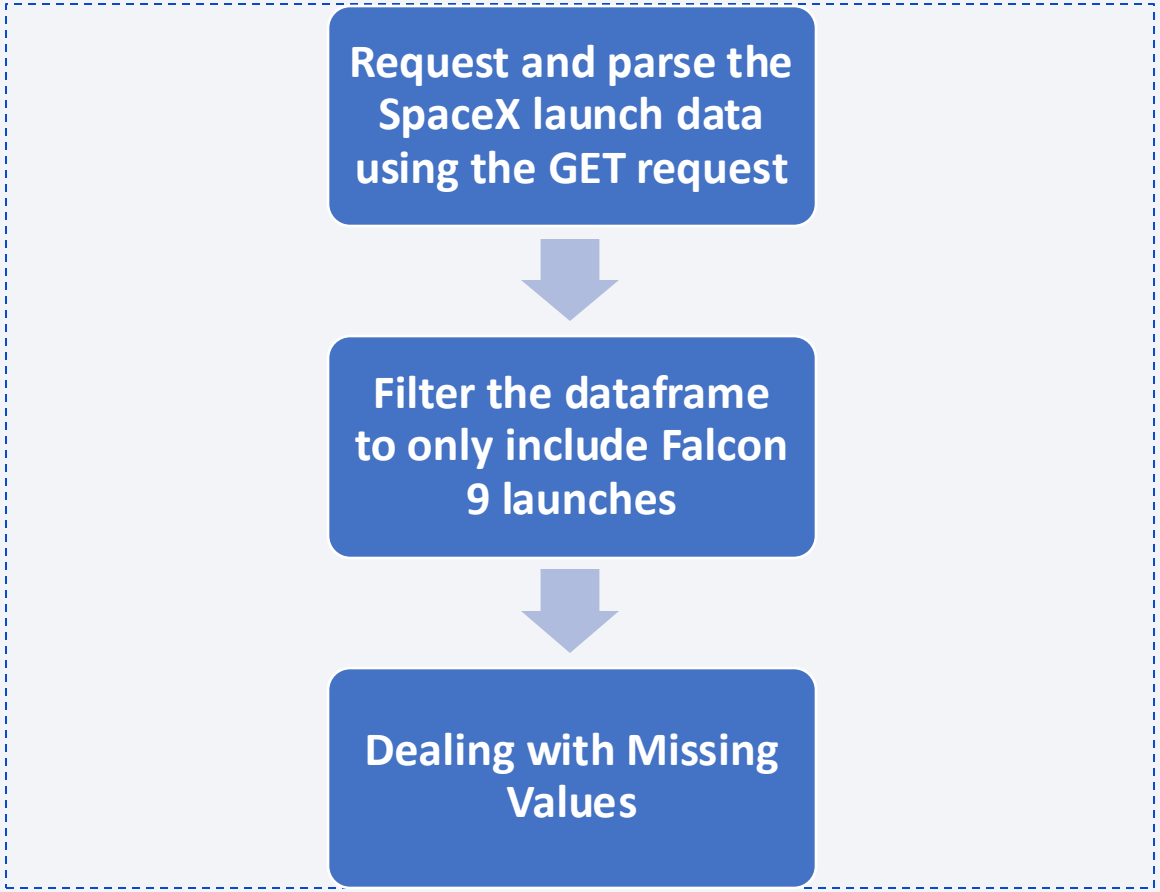
- Data collection methodology:
  - Describe how data was collected
- Perform data wrangling
  - Describe how data was processed
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - How to build, tune, evaluate classification models

Data Collection

- How data sets were collected.
  - SpaceX REST API

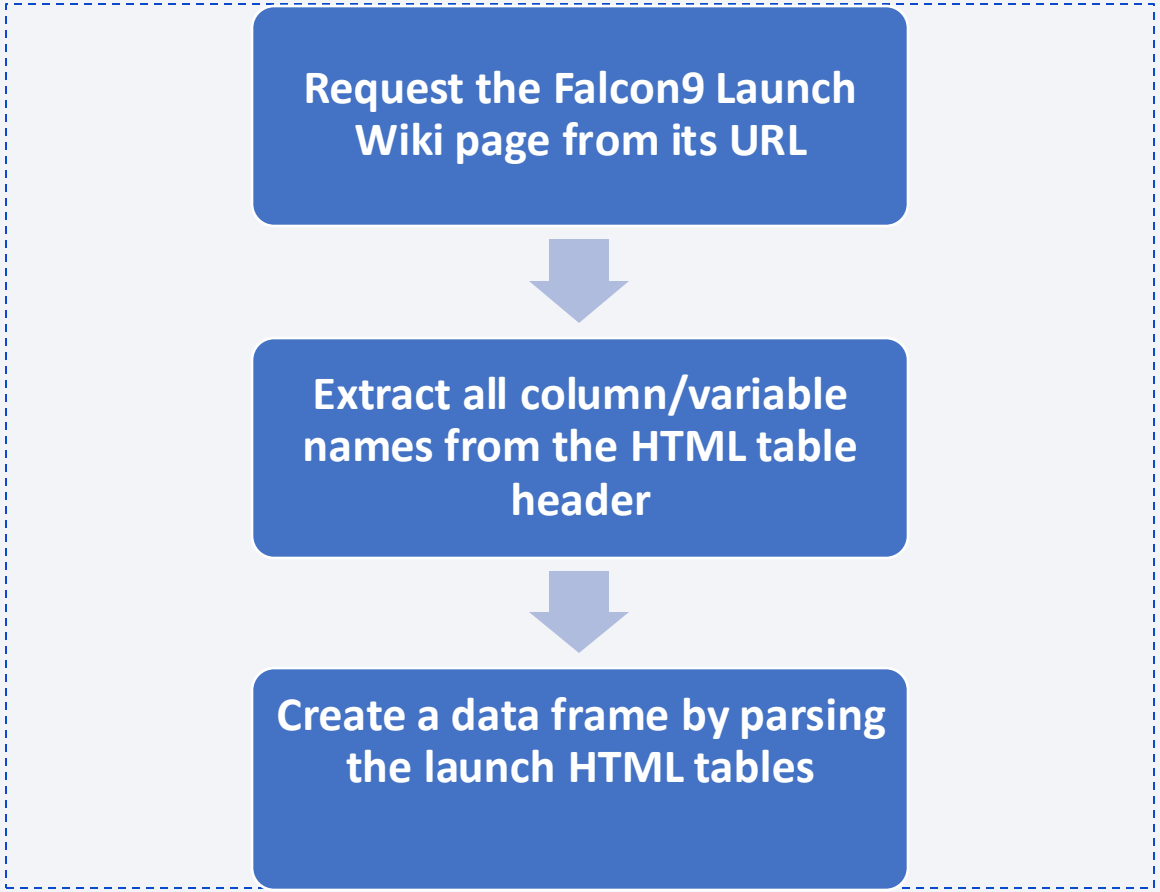
Data Collection – SpaceX API

- https://github.com/Yilin-Yu/Coursera-IBM/blob/master/Applied%20Data%20Science%20Capstone/module%201/jupyter-labs-spacex-data-collection-api-v2.ipynb

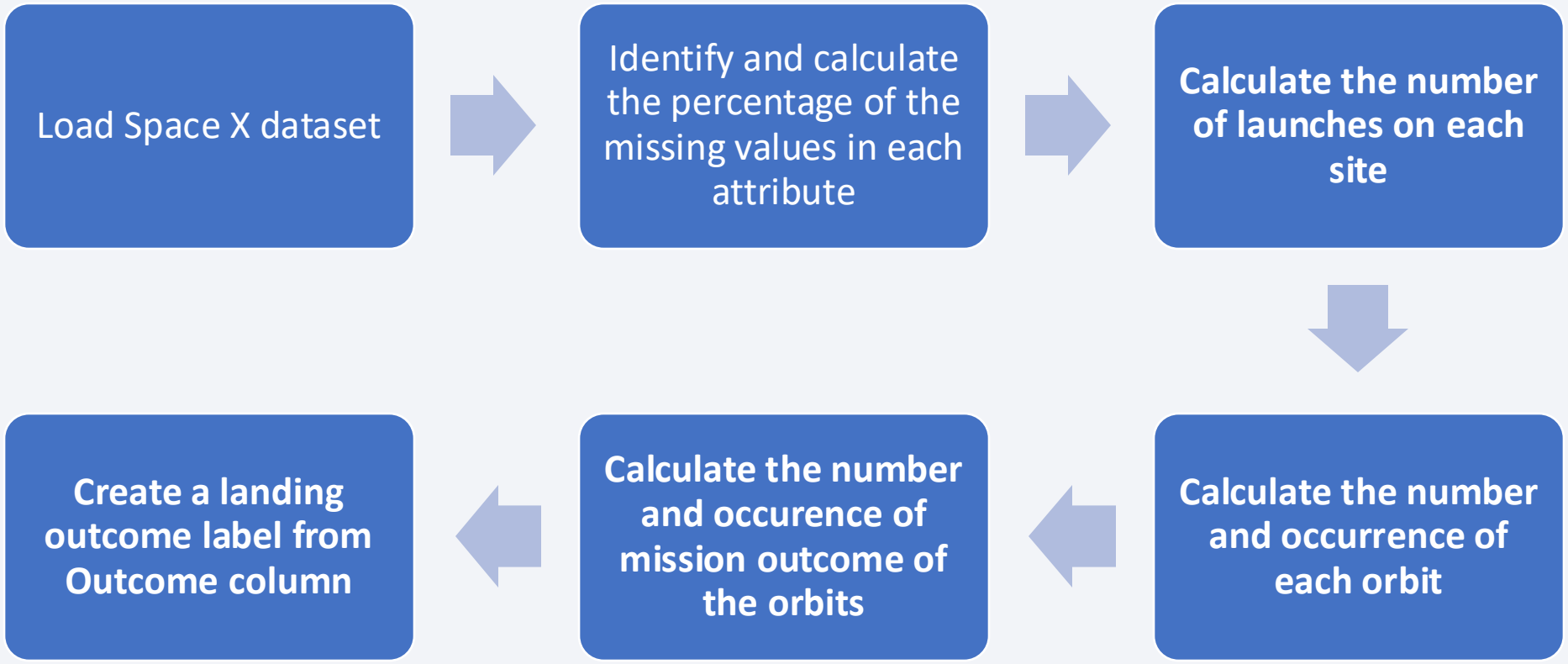


Data Collection - Scraping

- https://github.com/Yilin-Yu/Coursera-IBM/blob/master/Applied%20Data%20Science%20Capstone/module%201/jupyter-labs-webscraping.ipynb



Data Wrangling



- https://github.com/Yilin-Yu/Coursera-IBM/blob/master/Applied%20Data%20Science%20Capstone/module%201/labs-jupyter-spacex-Data%20wrangling-v2.ipynb

EDA with Data Visualization

- Scatter Plot (Two discrete data)
  - Visualize the relationship between Flight Number and Launch Site
  - Visualize the relationship between Payload and Launch Site
  - Visualize the relationship between FlightNumber and Orbit type
  - Visualize the relationship between Payload and Orbit type
- Bar Char (Group data compare)
  - Visualize the relationship between success rate of each orbit type
- Line Chart (To see the trend)
  - Visualize the launch success yearly trend

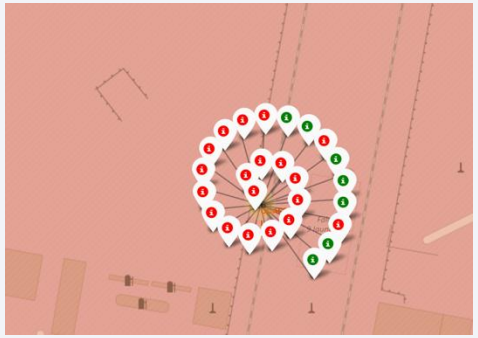
- https://github.com/Yilin-Yu/Coursera-IBM/blob/master/Applied%20Data%20Science%20Capstone/module%202/jupyter-labs-eda-dataviz-v2.ipynb

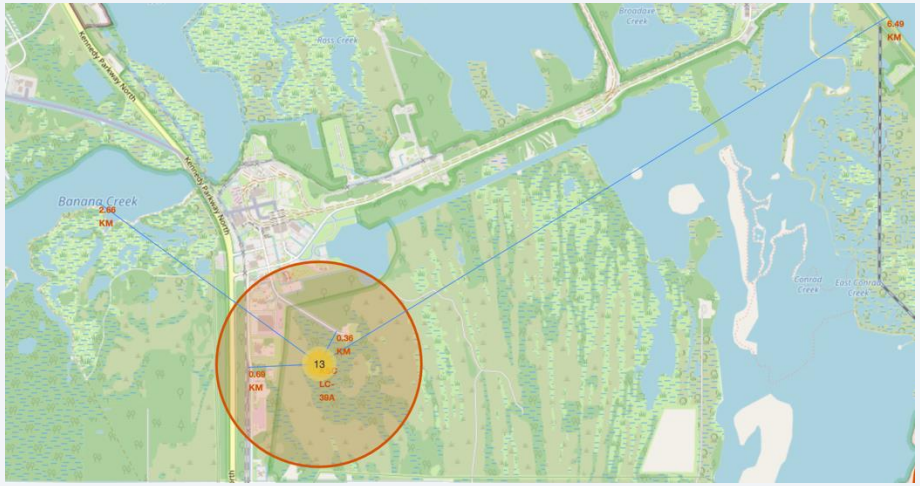
EDA with SQL

- Using SQL to figure out:
  - The names of the unique launch sites in the space mission
  - Total payload mass carried by boosters
  - Average payload mass carried by booster
  - The total number of successful and failure mission outcomes
  - etc.

- https://github.com/Yilin-Yu/Coursera-IBM/blob/master/Applied%20Data%20Science%20Capstone/module%202/jupyter-labs-eda-sql-coursera_sqllite.ipynb

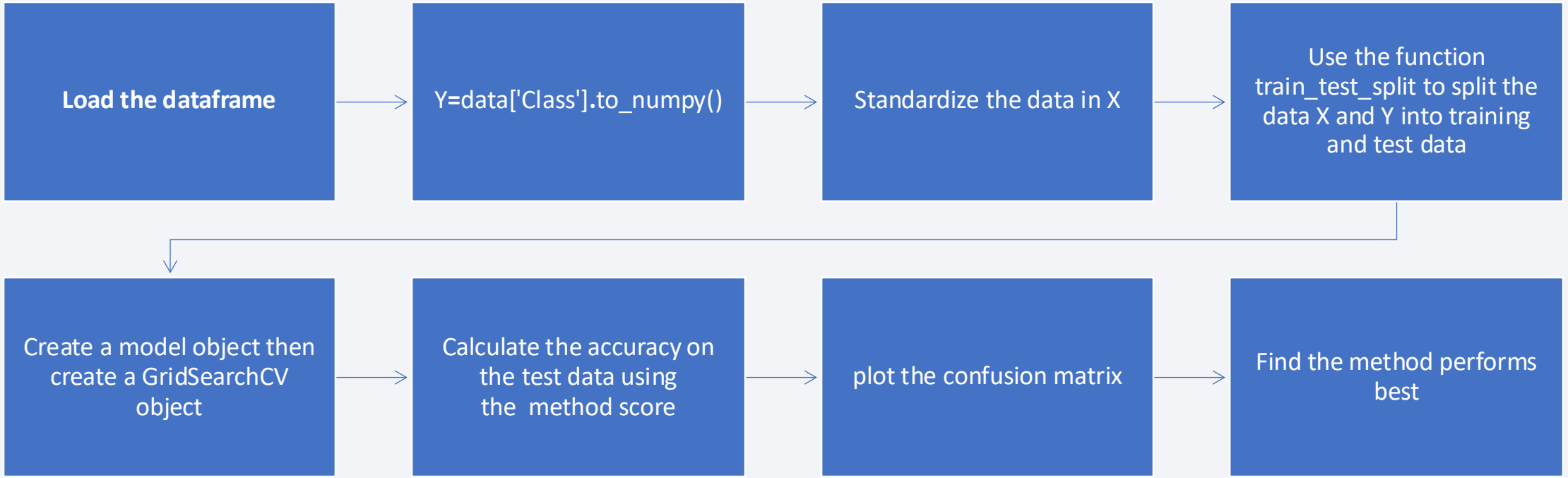
Build an Interactive Map with Folium





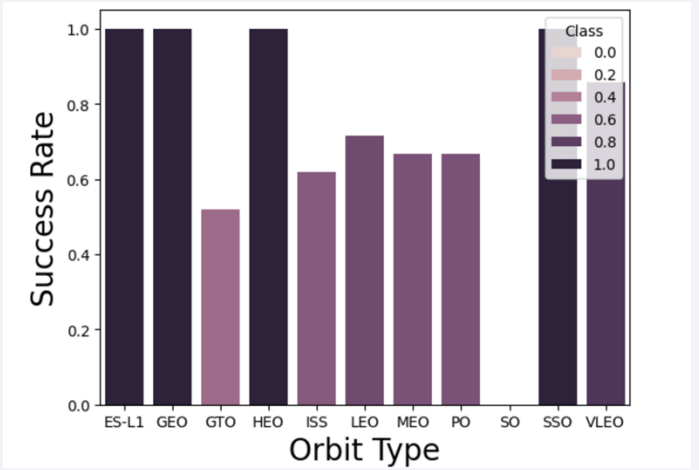
- From the color-labeled markers in marker clusters, it should be able to easily identify which launch sites have relatively high success rates.
- After plot the distance lines, it will be easy to find the proximity
  - https://github.com/Yilin-Yu/Coursera-IBM/blob/master/Applied%20Data%20Science%20Capstone/module%203/lab-jupyter-launch-site-location-v2.ipynb

Predictive Analysis (Classification)



- https://github.com/Yilin-Yu/Coursera-IBM/blob/master/Applied%20Data%20Science%20Capstone/module%204/SpaceX-Machine-Learning-Prediction-Part-5-v1.ipynb

Results

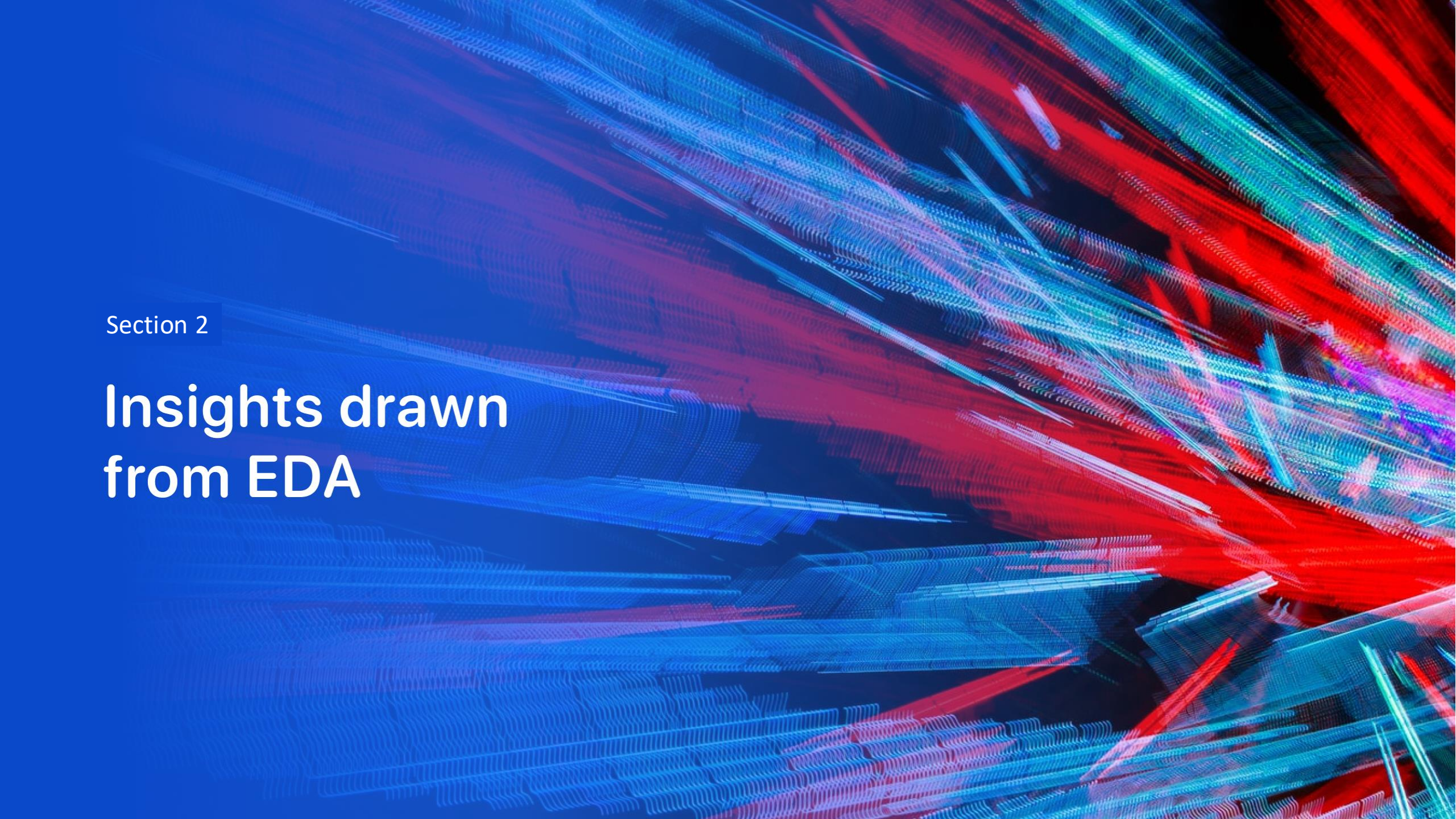


TASK 12

Find the method performs best:

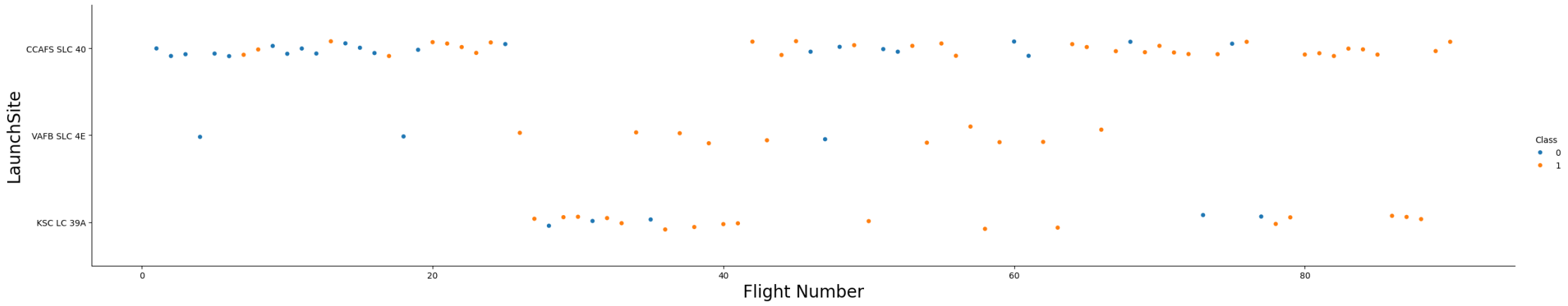
```
In [32]: logreg_cv_score, svm_cv_score, tree_cv_score, knn_cv_score

Out[32]: (0.8333333333333334,
          0.8333333333333334,
          0.7777777777777778,
          0.8333333333333334)
```


Section 2
Insights drawn from EDA

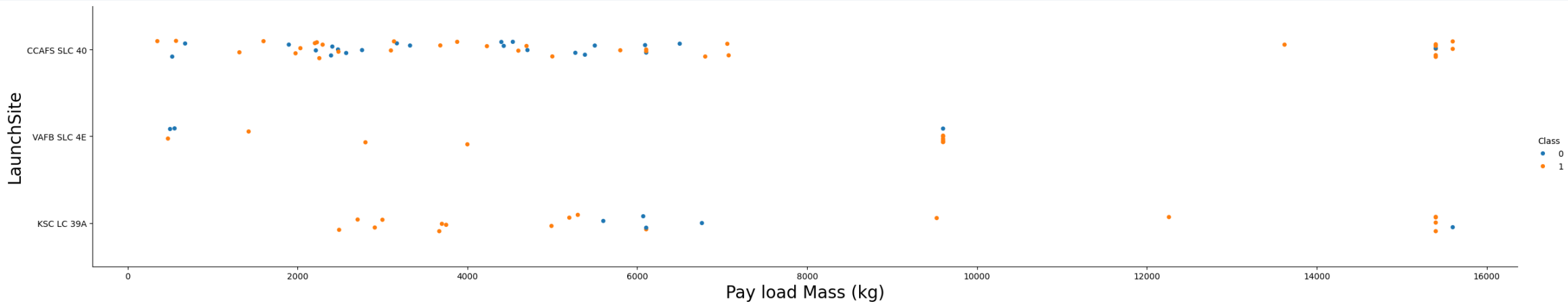
Flight Number vs. Launch Site

- Show a scatter plot of Flight Number vs. Launch Site



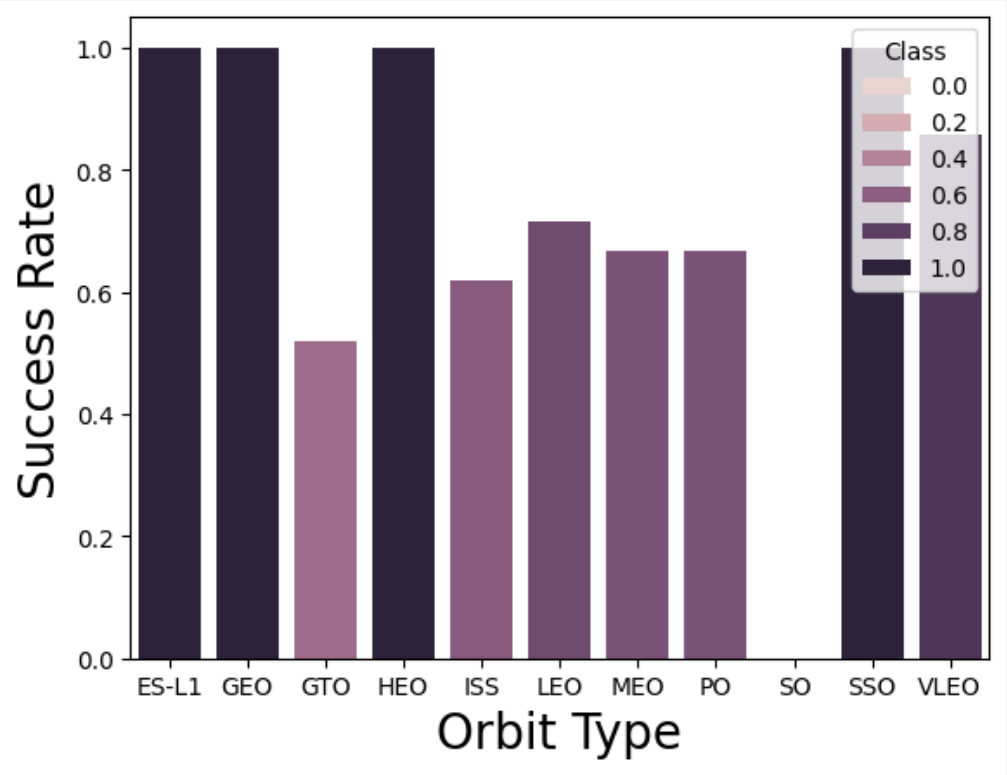
Payload vs. Launch Site

- Show a scatter plot of Payload vs. Launch Site



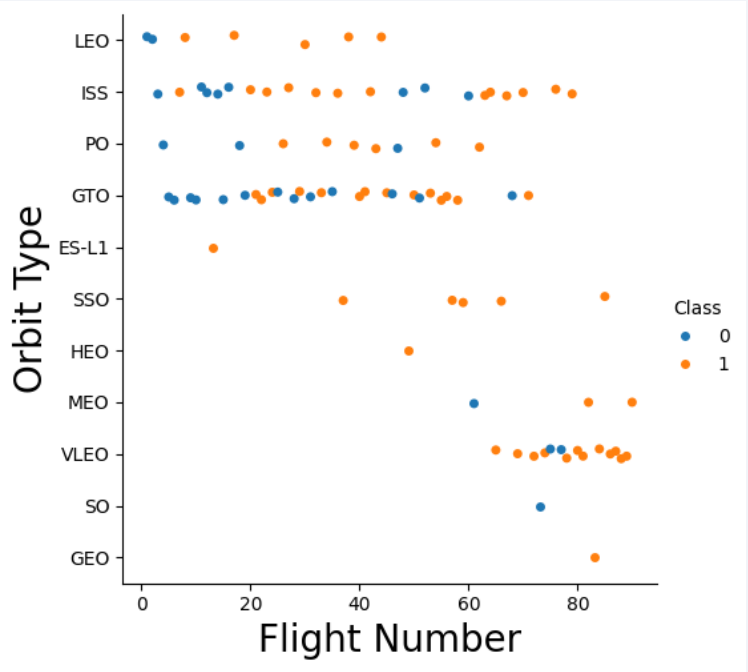
Success Rate vs. Orbit Type

- Show a bar chart for the success rate of each orbit type



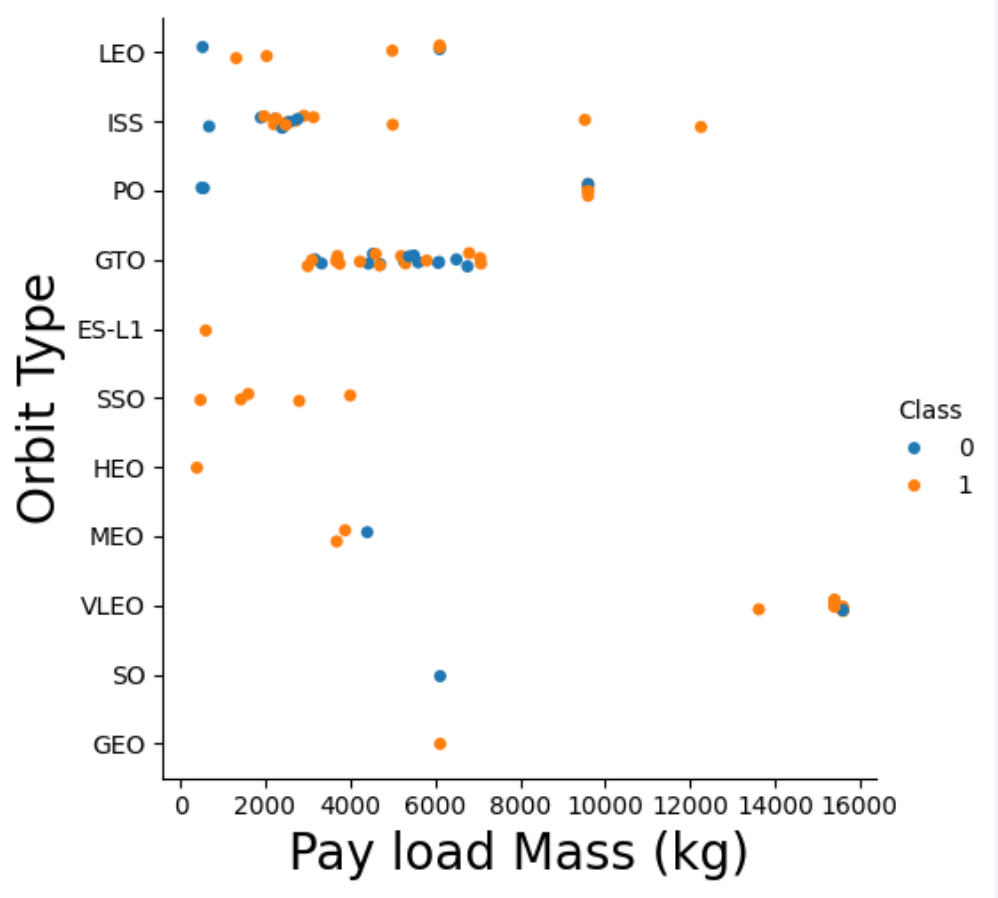
Flight Number vs. Orbit Type

- Show a scatter point of Flight number vs. Orbit type



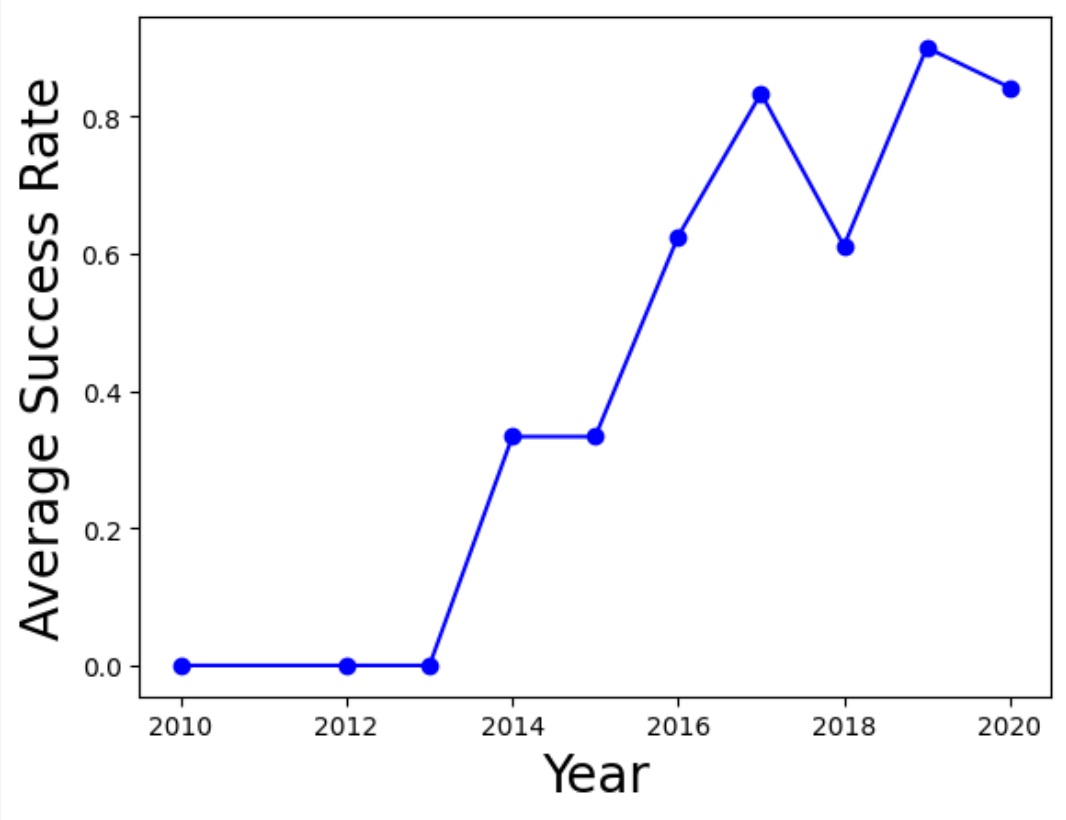
Payload vs. Orbit Type

- Show a scatter point of payload vs. orbit type



Launch Success Yearly Trend

- Show a line chart of yearly average success rate



All Launch Site Names

- Find the names of the unique launch sites
- Present your query result with a short explanation here

Display the names of the unique launch sites in the space mission

```
1  %sql select Distinct(Launch_Site) from SPACEXTABLE;
```

```
 * sqlite:///my_data1.db
Done.
```

Launch_Site
CCAFS LC-40
VAFB SLC-4E
KSC LC-39A
CCAFS SLC-40

Launch Site Names Begin with 'CCA'

- Find 5 records where launch sites begin with `CCA`
- Present your query result with a short explanation here

Display 5 records where launch sites begin with the string 'CCA'

```
1  %sql select * from SPACEXTABLE where Launch_Site like "%CCA%" limit 5;
```

```
 * sqlite:///my_data1.db
Done.
```

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
2010-12-08	15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight C1, two CubeSats, barrel of Brouere cheese	0	LEO (ISS)	NASA (COTS) NRO	Success	Failure (parachute)
2012-05-22	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
2012-10-08	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
2013-03-01	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

Total Payload Mass

- Calculate the total payload carried by boosters from NASA
- Present your query result with a short explanation here

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

```
1  %sql select sum(PAYLOAD_MASS__KG_) from SPACEXTABLE where Customer like "%NASA (CRS)%"
```

```
 * sqlite:///my_data1.db
Done.
```

sum(PAYLOAD_MASS__KG_)
48213

Average Payload Mass by F9 v1.1

- Calculate the average payload mass carried by booster version F9 v1.1
- Present your query result with a short explanation here

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

```
1  %sql select avg(PAYLOAD_MASS__KG_)  from SPACEXTABLE where Booster_version like "%F9 v1.1%"
```

```
 * sqlite:///my_data1.db
Done.
```

avg(PAYLOAD_MASS__KG_)
2534.6666666666665

First Successful Ground Landing Date

- Find the dates of the first successful landing outcome on ground pad
- Present your query result with a short explanation here

List the date when the first succesful landing outcome in ground pad was acheived.

Hint:Use min function

```
1  %sql select min(Date) from SPACEXTABLE where Landing_Outcome = "Success (ground pad)"
```

```
 * sqlite:///my_data1.db
Done.
```

min(Date)
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

- List the names of boosters which have successfully landed on drone ship and had payload mass greater than 4000 but less than 6000

- Present your query result with a short explanation here

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

```
1  %sql select distinct(Customer) from SPACEXTABLE where Landing_Outcome = "Success (drone ship)" and PAYLOAD_MASS__KG_ > 4000 and PAYLOAD_MASS__KG_ < 6000
```

```
 * sqlite:///my_data1.db
Done.
```

Customer
SKY Perfect JSAT Group
SES
SES EchoStar

Total Number of Successful and Failure Mission Outcomes

- Calculate the total number of successful and failure mission outcomes
- Present your query result with a short explanation here

List the total number of successful and failure mission outcomes

```
1  %sql select Mission_Outcome, count(*) from SPACEXTABLE group by Mission_Outcome
```

```
 * sqlite:///my_data1.db
Done.
```

Mission_Outcome	count(*)
Failure (in flight)	1
Success	98
Success	1
Success (payload status unclear)	1

Boosters Carried Maximum Payload

- List the names of the booster which have carried the maximum payload mass
- Present your query result with a short explanation here

List the names of the booster_versions which have carried the maximum payload mass. Use a subquery

```
1  %sql select distinct(Booster_Version) from SPACEXTABLE where PAYLOAD_MASS__KG_ = (select max(PAYLOAD_MASS__KG_) from SPACEXTABLE)
```

```
 * sqlite:///my_data1.db
Done.
```

Booster_Version
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

2015 Launch Records

- List the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015

- Present your query result with a short explanation here

List the records which will display the month names, failure landing_outcomes in drone ship ,booster versions, launch_site for the months in year 2015.

Note: SQLLite does not support monthnames. So you need to use substr(Date, 6,2) as month to get the months and substr(Date,0,5)='2015' for year.

```
1  %sql select substr(Date,0,5) as Year, substr(Date,6,2) as month, \
2      Landing_Outcome, Booster_Version, Launch_Site from SPACEXTABLE where Landing_Outcome = "Failure (drone ship)" and substr(Date,0,5)="2015"
```

```
 * sqlite:///my_data1.db
Done.
```

Year	month	Landing_Outcome	Booster_Version	Launch_Site
2015	01	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
2015	04	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

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

- Rank 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

- Present your query result with a short explanation here

Rank 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.

```
%sql select Landing_Outcome, count(Landing_Outcome) from (select * from SPACEXTABLE where \
    Landing_Outcome = "Failure (drone ship)" or Landing_Outcome = "Success (ground pad)" and cast(substr(Date,0,5) as real) <= 2017 \
        and cast(substr(Date,6,2) as real) <= 3) group by Landing_Outcome
```

```
 * sqlite:///my_data1.db
Done.
```

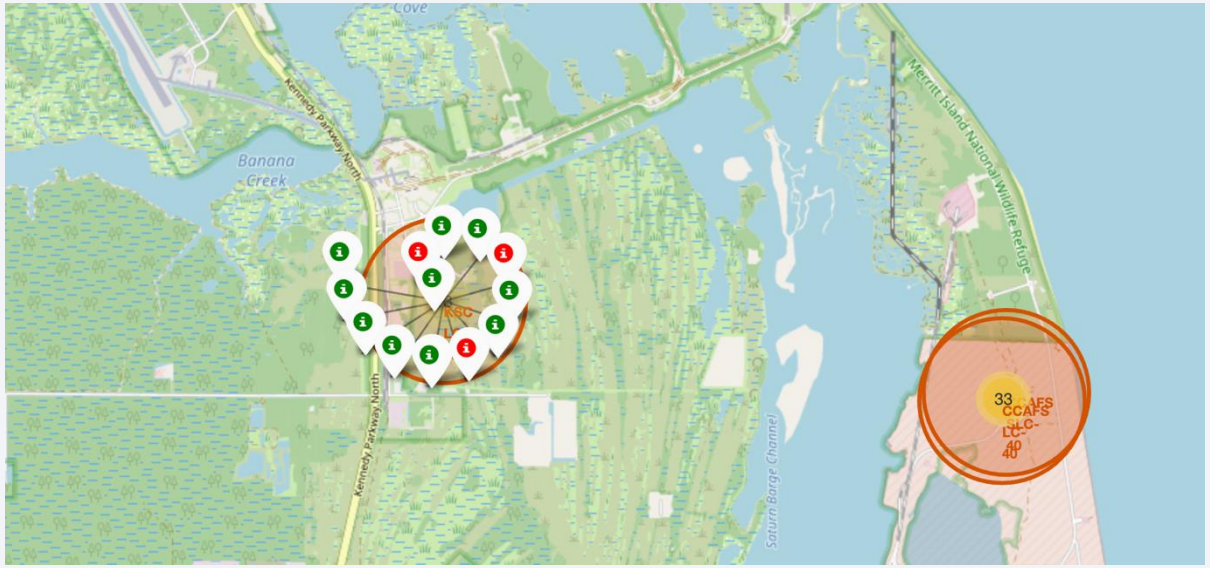
Landing_Outcome	count(Landing_Outcome)
Failure (drone ship)	5
Success (ground pad)	1

Section 3

Launch Sites Proximities Analysis

Mark all launch sites on a map

Mark the success/failed launches for each site on the map



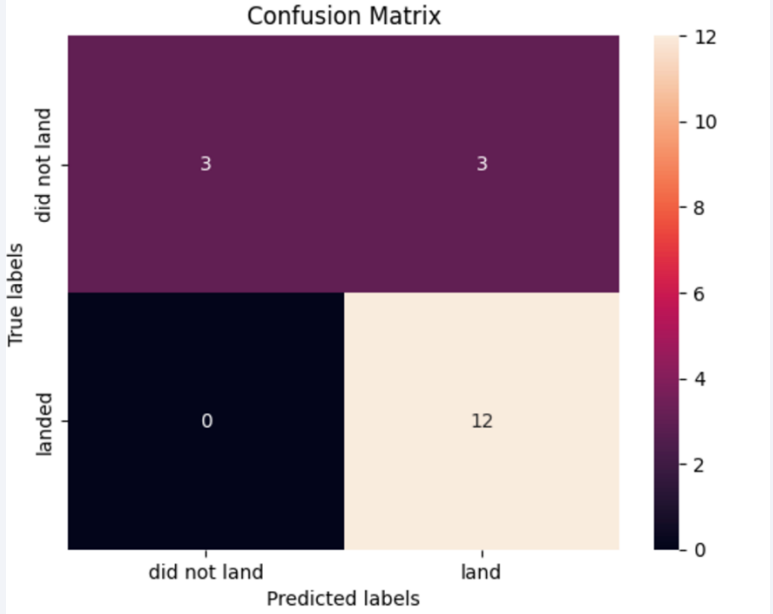
Distances between a launch site to its proximities

Section 5

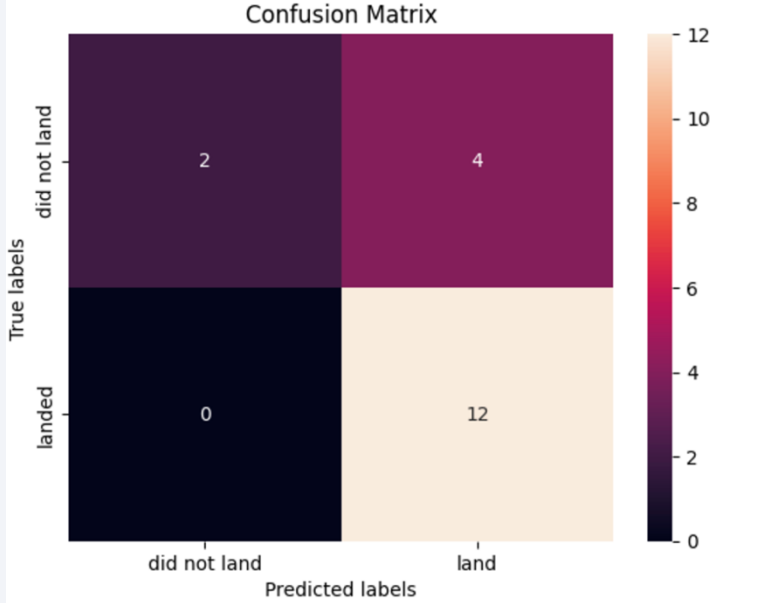
Predictive Analysis (Classification)

Confusion Matrix

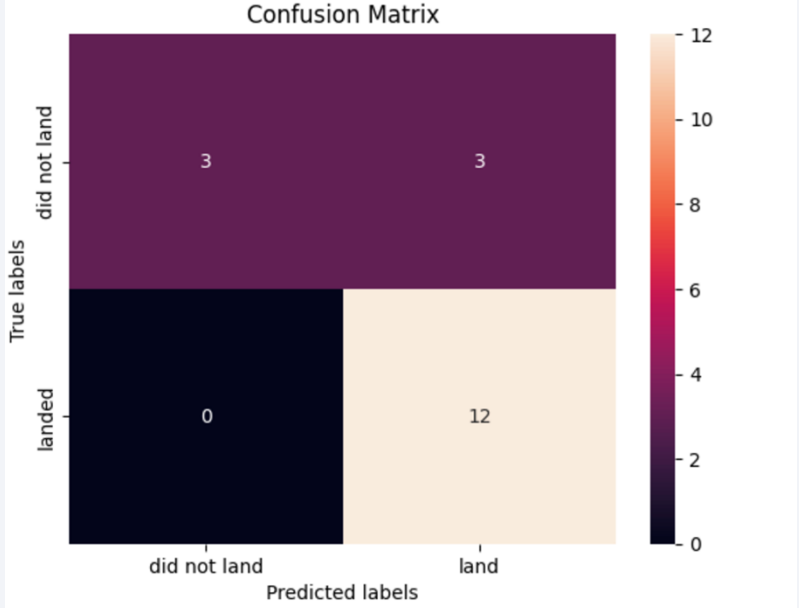
```
yhat=logreg_cv.predict(X_test)
plot_confusion_matrix(Y_test,yhat)
```




```
yhat = tree_cv.predict(X_test)
plot_confusion_matrix(Y_test,yhat)
```




```
yhat = knn_cv.predict(X_test)
plot_confusion_matrix(Y_test,yhat)
```



Thank you!