Data Science Capstone Project

Lakshay Gupta

03/07/2024

OUTLINE

- Executive Summary
- Introduction
- Methodology
- Results
  - Visualization – Charts
  - Dashboard
- Discussion
  - Findings & Implications
- Conclusion
- Appendix

EXECUTIVE SUMMARY

- Gathered information from the SpaceX Wikipedia page and the public SpaceX API. a labels column called "class" was created to categorize successful landings. used dashboards, folium maps, SQL, and visualization to explore data. compiled pertinent columns for the features. one hot encoding was used to convert all categorical variables to binary. To determine the ideal parameters for machine learning models, standardized data was used along with GridSearchCV. Display each model's accuracy score visually.

- The following four machine learning models were created: K Nearest Neighbors, Decision Tree Classifier, Support Vector Machine, and Logistic Regression. All yielded comparable outcomes, with an accuracy percentage of roughly 83.33%. Every model overestimated the number of successful landings. To improve the determination and accuracy of the model, more data are required.

INTRODUCTION

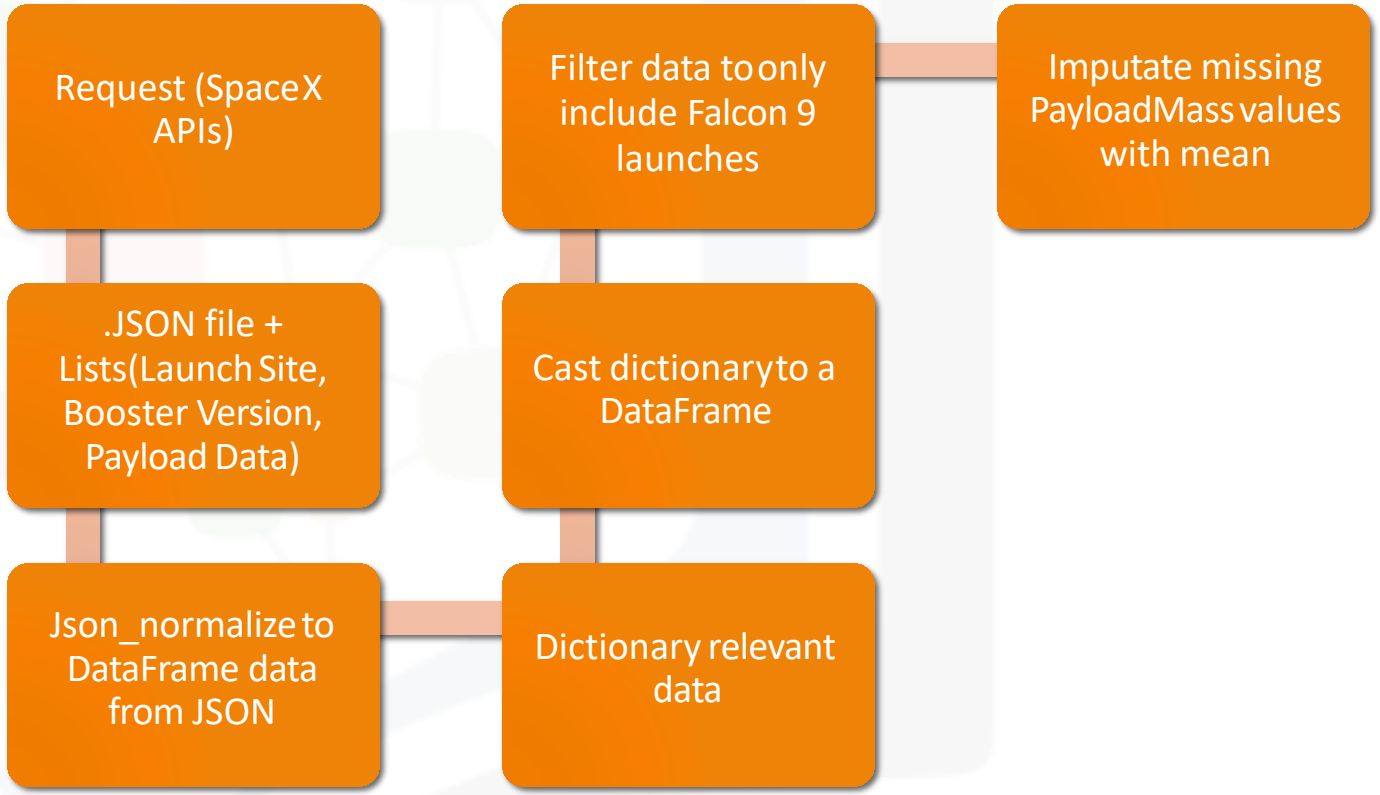
- Problem
  - SpaceY gave us a task to predict successful Stage 1 Recovery with data
- Background
  - SpaceY wants to compete with SpaceX
  - Space is next big thing
  - This project will help us identify key pointers for launch

METHODOLOGY

- Combined data from SpaceX public API and SpaceX Wikipedia page
- Classifying true landings as successful and unsuccessful otherwise
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models

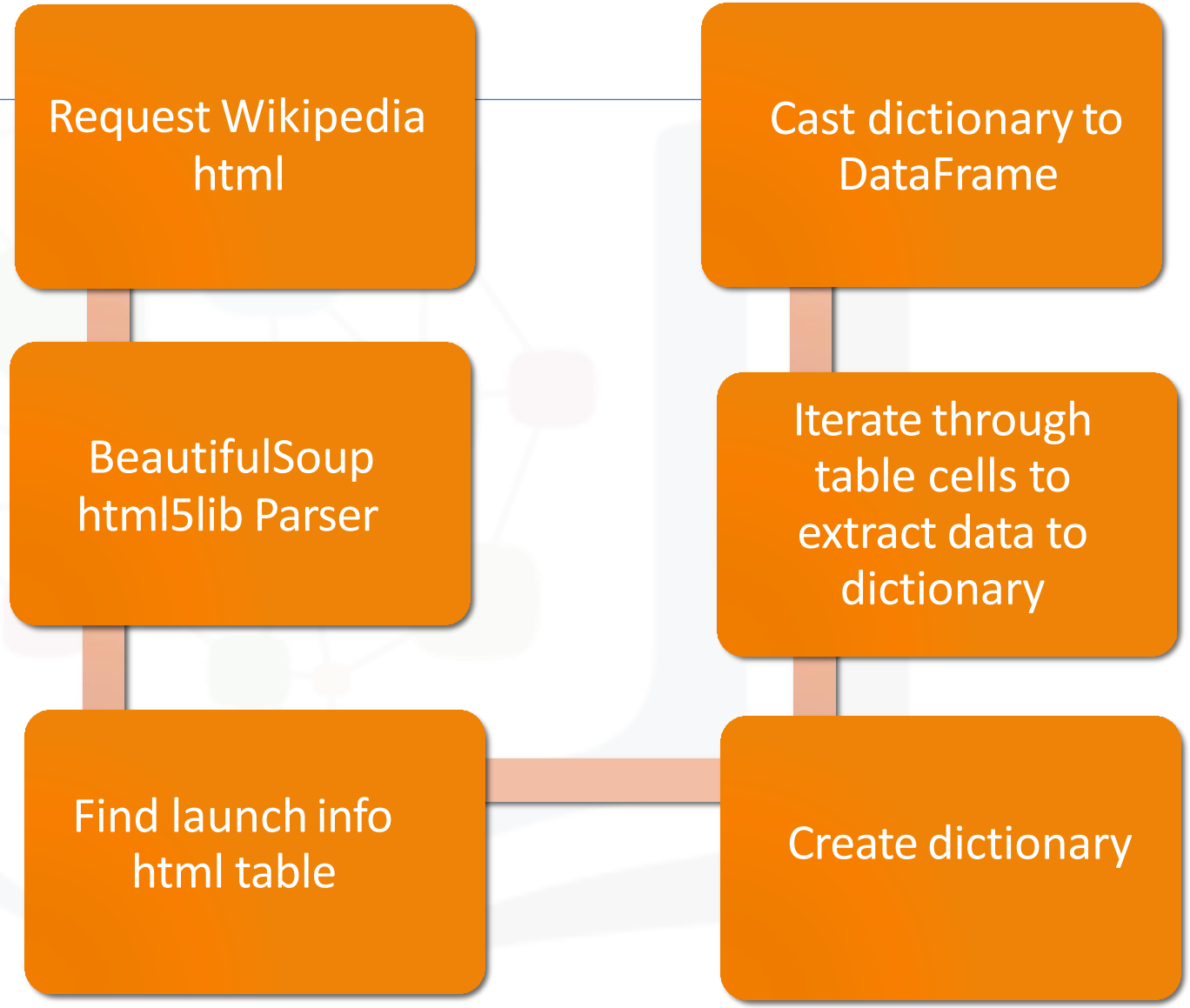
Data Collection

- Github URL –
- URL to DataCollectionNotebook



Web Scraping

- Github URL –
- URL to WeScrapingNotebook



Data Wrangling

- Create a training label with landing outcomes where successful = 1 & failure = 0.
- Outcome column has two components: ‘Mission Outcome’ ‘Landing Location’
- New training label column ‘class’ with a value of 1 if ‘Mission Outcome’ is True and 0 otherwise. Value Mapping:
- True ASDS, True RTLS, & True Ocean – set to -> 1
- None None, False ASDS, None ASDS, False Ocean, False RTLS – set to -> 0
- Github URL –
- URL to DataWranglingNotebook

EDA – Data Visualization

- Exploratory Data Analysis performed on variables Flight Number, Payload Mass, Launch Site, Orbit, Class and Year.
- Plots Used:
  - Flight Number vs. Payload Mass, Flight Number vs. Launch Site, Payload Mass vs. Launch Site, Orbit vs. Success Rate, Flight Number vs. Orbit, Payload vs Orbit, and Success Yearly Trend
  - Scatter plots, line charts, and bar plots were used to compare relationships between variables to decide if a relationship exists so that they could be used in training the machine learning model
- Github URL –
- URL to DataVizualizationNotebook

EDA - SQL

- Loaded data set into IBM DB2 Database.
  - Queried using SQL Python integration.
  - Queries were made to get a better understanding of the dataset.
  - Queried information about launch site names, mission outcomes, various pay load sizes of customers and booster versions, and landing outcomes

- Github URL –
- URL to SQL_Notebook

Folium

- Folium maps mark Launch Sites, successful and unsuccessful landings, and a proximity example to key locations: Railway, Highway, Coast, and City.
- This allows us to understand why launch sites may be located where they are. Also visualizes successful landings relative to location.

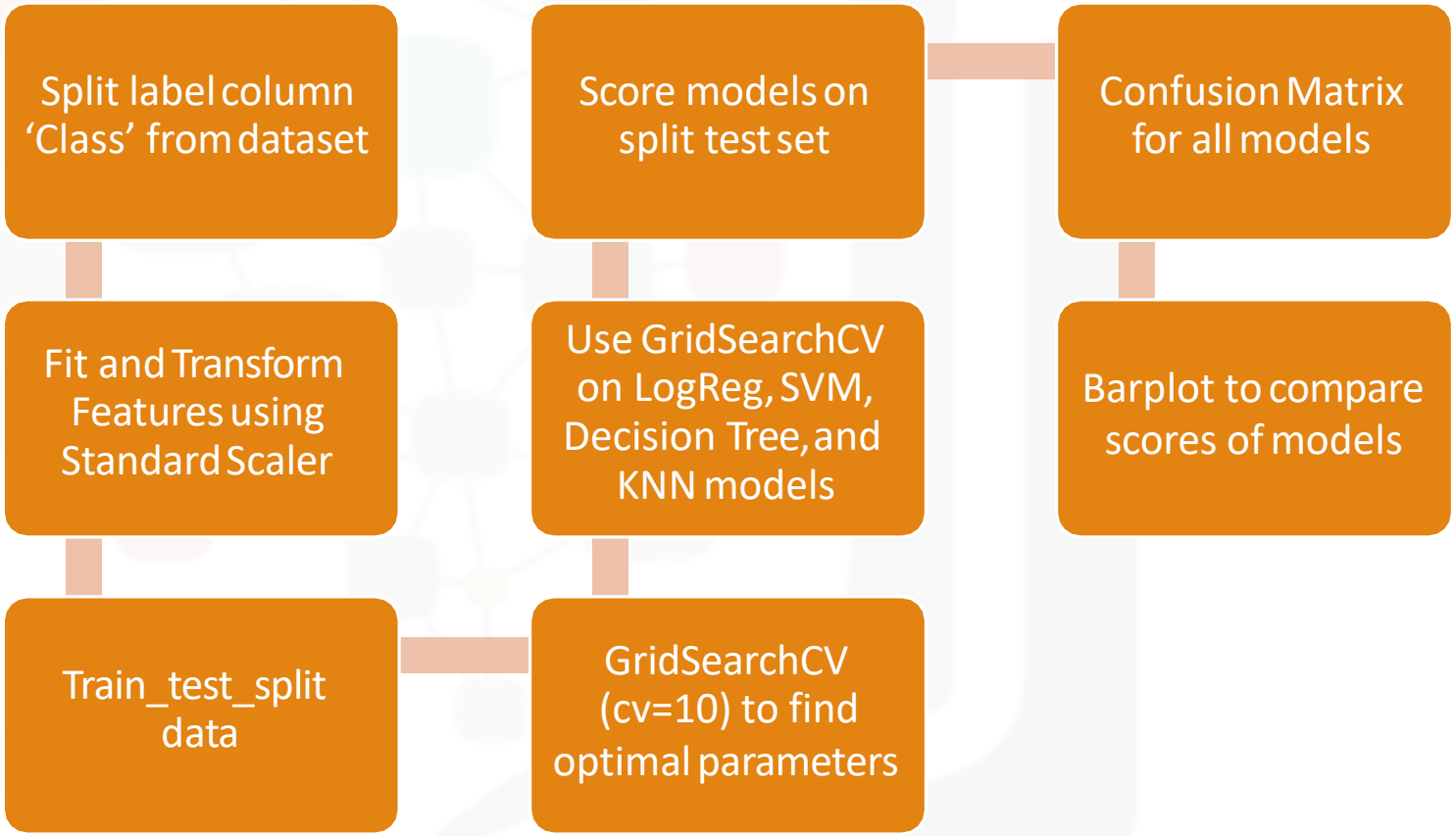
- Github URL –
- URL to FoliumNotebook

Plotly Dash

- Dashboard includes a pie chart and a scatter plot.
- Pie chart can be selected to show distribution of successful landings across all launch sites and can be selected to show individual launch site success rates.
- Scatter plot takes two inputs: All sites or individual site and payload mass on a slider between 0 and 10000 kg.
- The pie chart is used to visualize launch site success rate.
- The scatter plot can help us see how success varies across launch sites, payload mass, and
- booster version category.
- Github URL –
- URL to PlotlyFile

Predictive Analysis(Classification)

- Github URL –
- URL to ML_Notebook

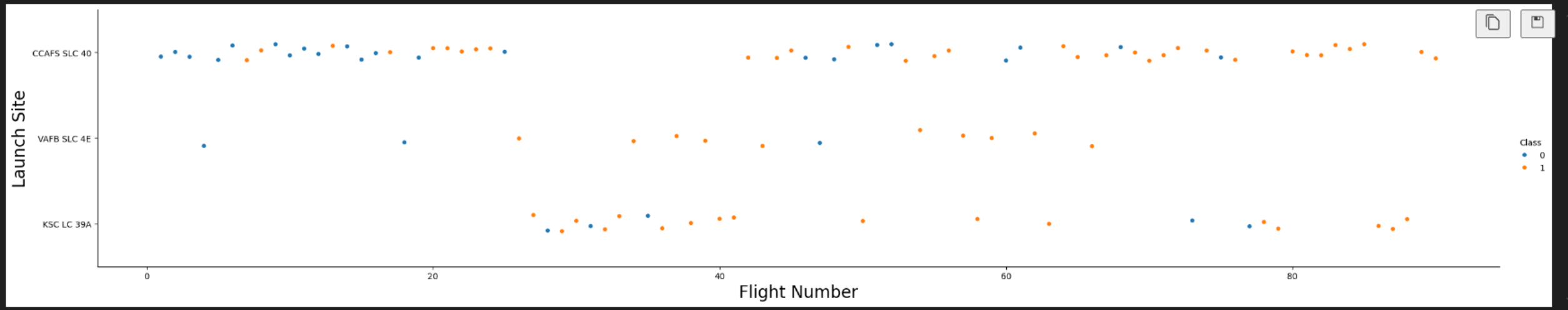


EDA Visualization

- From this FlightNumber vs LaunchSite, we can see that most of the flight goes from CCAFS SLC 40

```
### TASK 1: Visualize the relationship between Flight Number and Launch Site
sns.catplot(x = "FlightNumber", y = "LaunchSite", data = df, hue = "Class",aspect = 5)
plt.xlabel("Flight Number",fontsize=20)
plt.ylabel("Launch Site",fontsize=20)
plt.show()
```

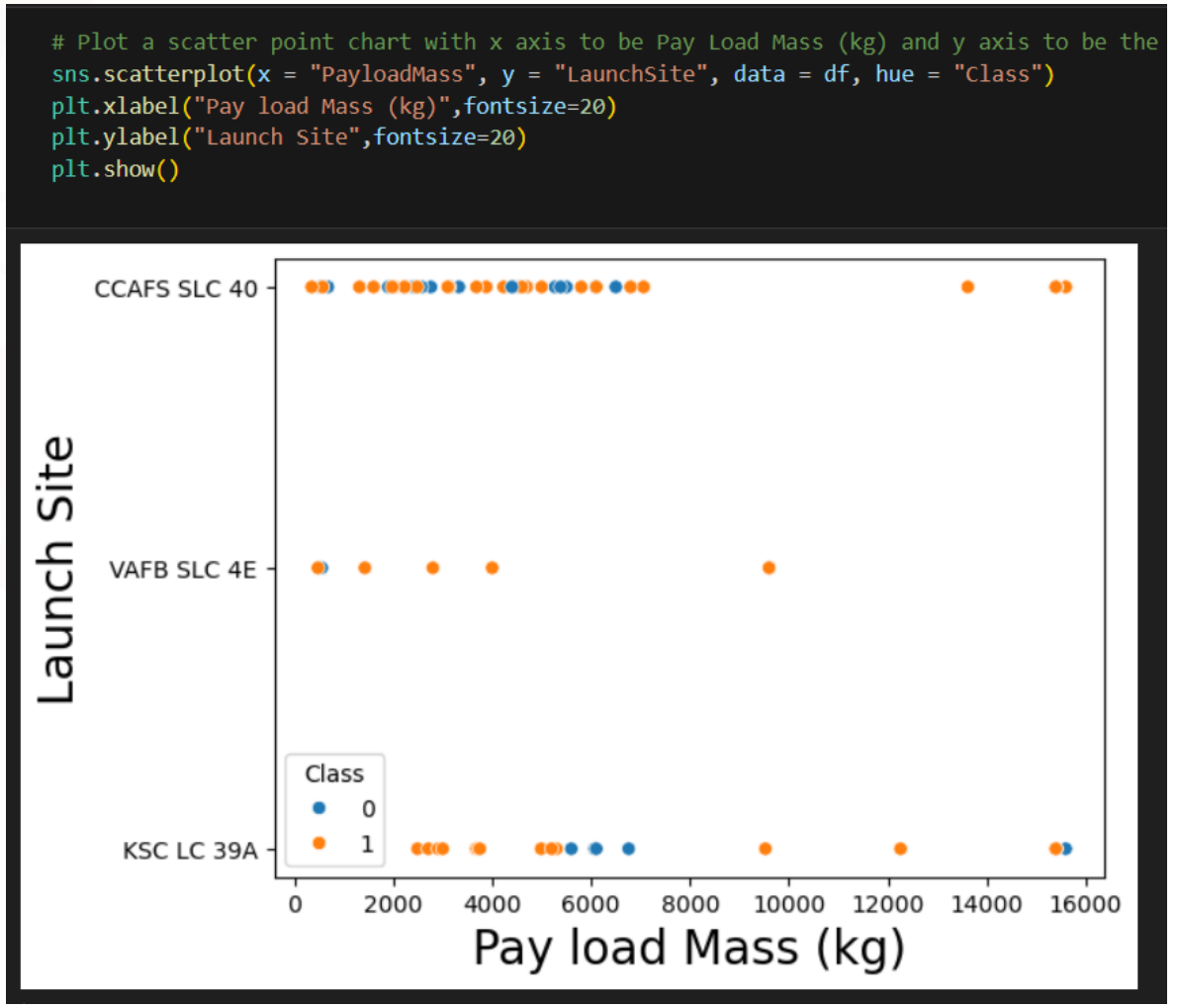
Python



EDA Visualization

Payload mass appears to fall mostly between 0-6000 kg. Different launch sites also seem to use different payload mass

```
# Plot a scatter point chart with x axis to be Pay Load Mass (kg) and y axis to be the
sns.scatterplot(x = "PayloadMass", y = "LaunchSite", data = df, hue = "Class")
plt.xlabel("Pay load Mass (kg)",fontsize=20)
plt.ylabel("Launch Site",fontsize=20)
plt.show()
```

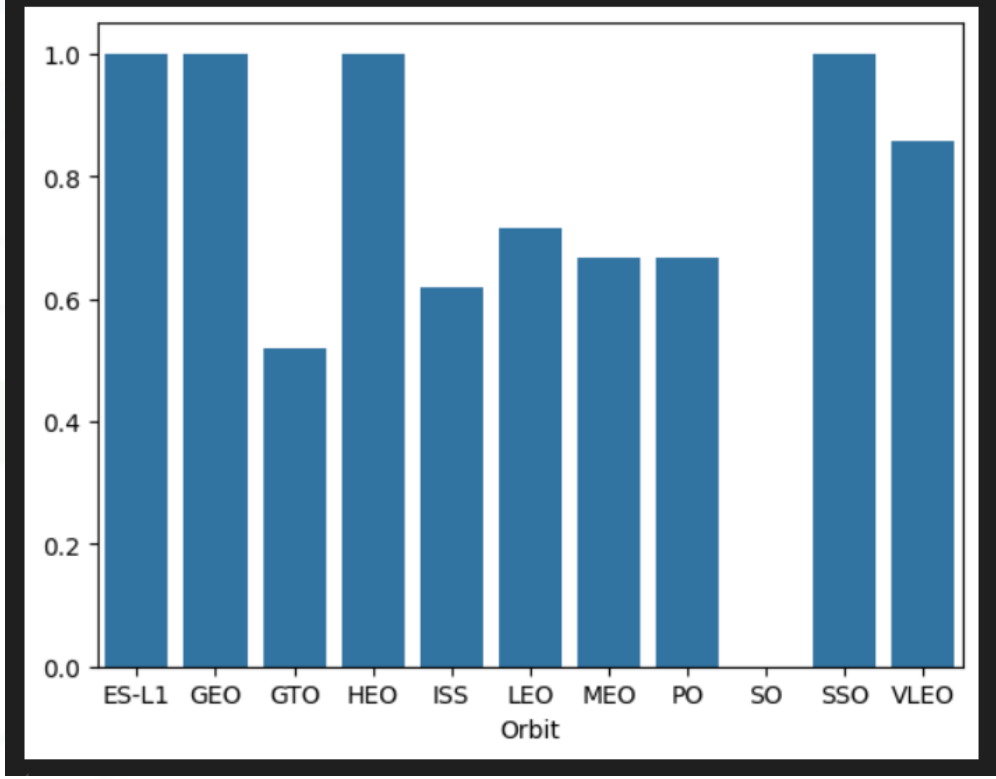


EDA Visualization

- Success Rate vs Orbit- ES-L1 (1), GEO (1), HEO (1) have 100% success rate (sample sizes in parenthesis) SSO (5) has 100% success rate
- VLEO (14) has decent success rate and attempts
- SO (1) has 0% success rate
- GTO (27) has the around 50% success rate but largest sample

```
# HINT use groupby method on Orbit column and get the mean of Class column
bar_df = df.groupby("Orbit")["Class"].mean()
sns.barplot(x = bar_df.index, y = bar_df.values)
```

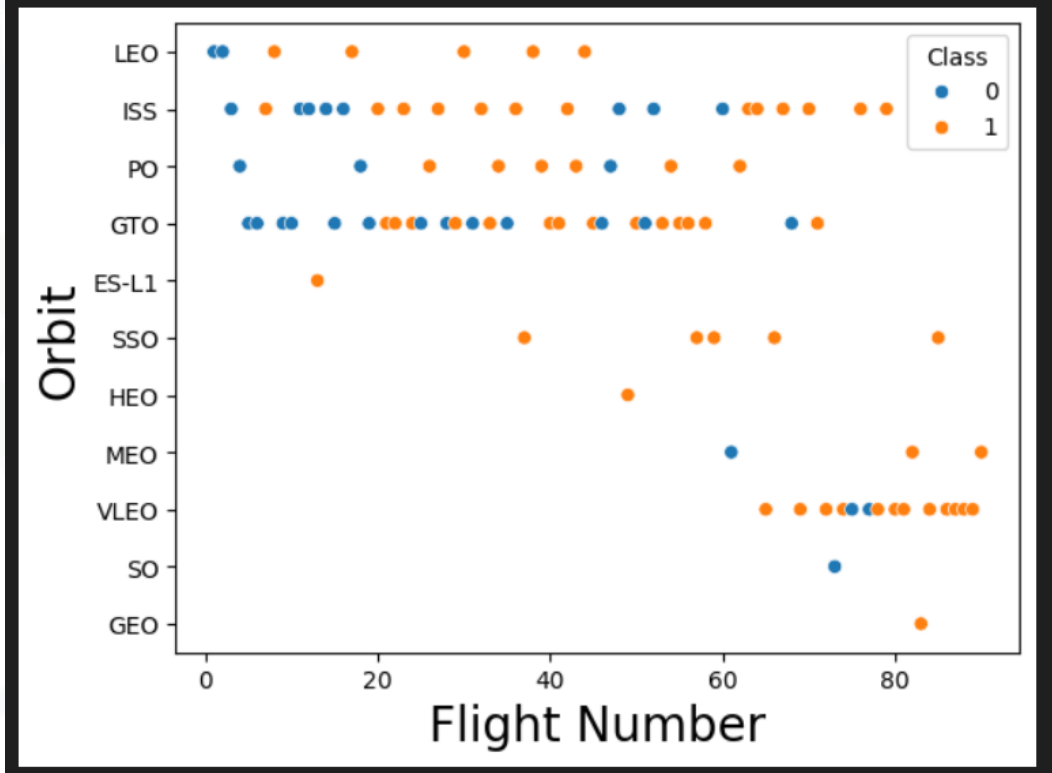
```
<AxesSubplot:xlabel='Orbit'>
```



EDA Visualization

- Launch Orbit preferences changed over Flight Number.
- Launch Outcome seems to correlate with this preference.
- SpaceX started with LEO orbits which saw moderate success LEO and returned to VLEO in recent launches
- SpaceX appears to perform better in lower orbits or Sun-synchronous orbits

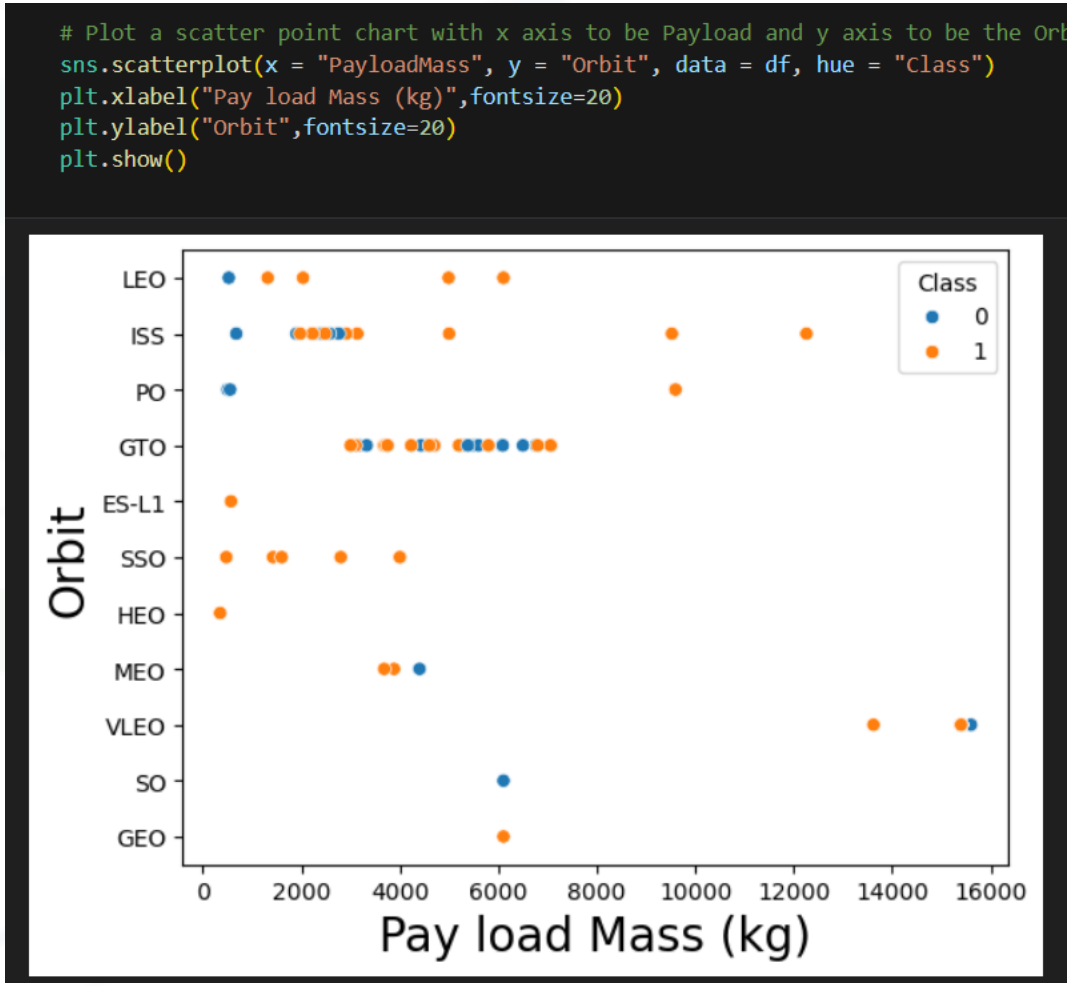
```
# Plot a scatter point chart with x axis to be FlightNumber and y axis to be the
sns.scatterplot(x = "FlightNumber", y = "Orbit", data = df, hue = "Class")
plt.xlabel("Flight Number",fontsize=20)
plt.ylabel("Orbit",fontsize=20)
plt.show()
```



EDA Visualization

- Payload mass seems to correlate with orbit
- LEO and SSO seem to have relatively low payload mass
- The other most successful orbit VLEO only has payload mass values in the higher end of the range

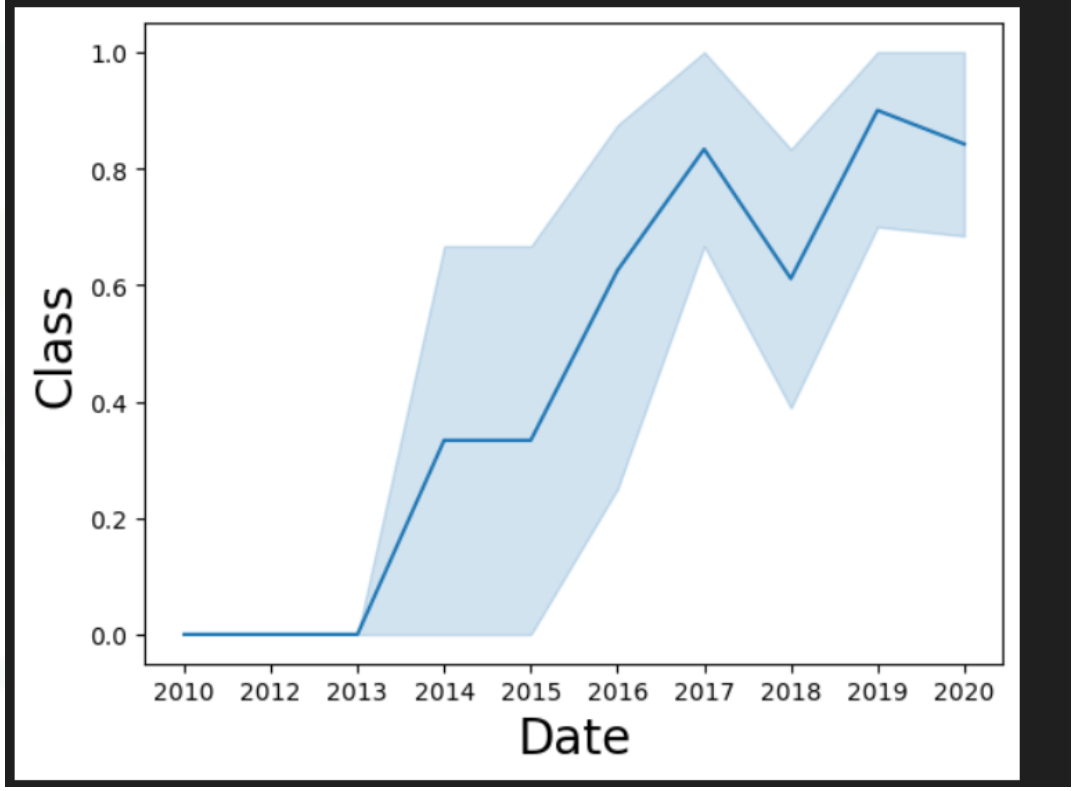
```
# Plot a scatter point chart with x axis to be Payload and y axis to be the Orb
sns.scatterplot(x = "PayloadMass", y = "Orbit", data = df, hue = "Class")
plt.xlabel("Pay load Mass (kg)",fontsize=20)
plt.ylabel("Orbit",fontsize=20)
plt.show()
```



EDA Visualization

- Success generally increases over time since 2013 with a slight dip in 2018
- Success in recent years at around 80%

```
# Plot a line chart with x axis to be the extracted year and y axis to be the
sns.lineplot(x = "Date", y = "Class", data = df)
plt.xlabel("Date",fontsize=20)
plt.ylabel("Class",fontsize=20)
plt.show()
```



EDA SQL

- Query unique launch site names from database.
- CCAFS SLC-40 and CCAFSSLC-40 likely all represent the same launch site with data entry errors.
- CCAFS LC-40 was the previous name.

```
%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

EDA SQL

- First five entries in database with Launch Site name beginning with CCA.

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

Python

```
 * 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

EDA SQL

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

```
%sql select SUM("PAYLOAD_MASS__KG_") as "Sum_Payload_Mass" from SPACEXTABLE where Customer like "NASA (CRS)"
```

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

Sum_Payload_Mass
45596

- This query sums the total payload mass in kg where NASA was the customer.
- CRS stands for Commercial Resupply Services which indicates that these payloads were sent to the International Space Station (ISS).

EDA SQL

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

```
%sql select AVG("PAYLOAD_MASS__KG_") as "Average_Payload_Mass" from SPACEXTABLE where Booster_Version like "F9 v1.1%"
```

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

Average_Payload_Mass
2534.6666666666665

- This query calculates the average payload mass or launches which used booster version F9 v1.1
- Average payload mass of F9 1.1 is on the low end of our payload mass range

EDA SQL

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

Hint:Use min function

```
%sql select min("Date") as "First_Successful_Landing" from SPACEXTABLE where Landing_Outcome like "Success%"
```

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

First_Successful_Landing
2015-12-22

- This query returns the first successful ground pad landing date.
- First ground pad landing wasn’t until the end of 2015.
- Successful landings in general appear starting 2014.

EDA SQL

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

```
%sql select "Booster_Version" from SPACEXTABLE where Landing_Outcome like "Success (drone ship)" and PAYLOAD_MASS__KG_ > 4000 and PAYLOAD_MASS__KG_ <6000
```

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

Booster_Version
F9 FT B1022
F9 FT B1026
F9 FT B1021.2
F9 FT B1031.2

- This query returns the four booster versions that had successful drone ship landings and a payload mass between 4000 and 6000 noninclusively.

EDA SQL

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

Hint:Use min function

```
%sql select min("Date") as "First_Successful_Landing" from SPACEXTABLE where Landing_Outcome like "Success%"
```

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

First_Successful_Landing
2015-12-22

- This query returns the first successful ground pad landing date.
- First ground pad landing wasn’t until the end of 2015.
- Successful landings in general appear starting 2014.

EDA SQL

- This query returns the booster versions that carried the highest payload mass of 15600 kg.
- These booster versions are very similar, and all are of the F9 B5 B10xx.x variety.
- This likely indicates payload mass correlates with the booster version that is used.

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

```
%sql select "Booster_Version" from SPACEXTABLE where PAYLOAD_MASS__KG_ like (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

EDA SQL

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.

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

Python

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

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

- This query returns the Month, Landing Outcome, Booster Version, Payload Mass (kg), and Launch site of 2015 launches where stage 1 failed to land on a drone ship.
- There were two such occurrences

EDA SQL

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(*) from SPACEXTABLE where Date > '20100604' and Date < '20170320' group by Landing_Outcome order by count(*) desc
```

Pytho

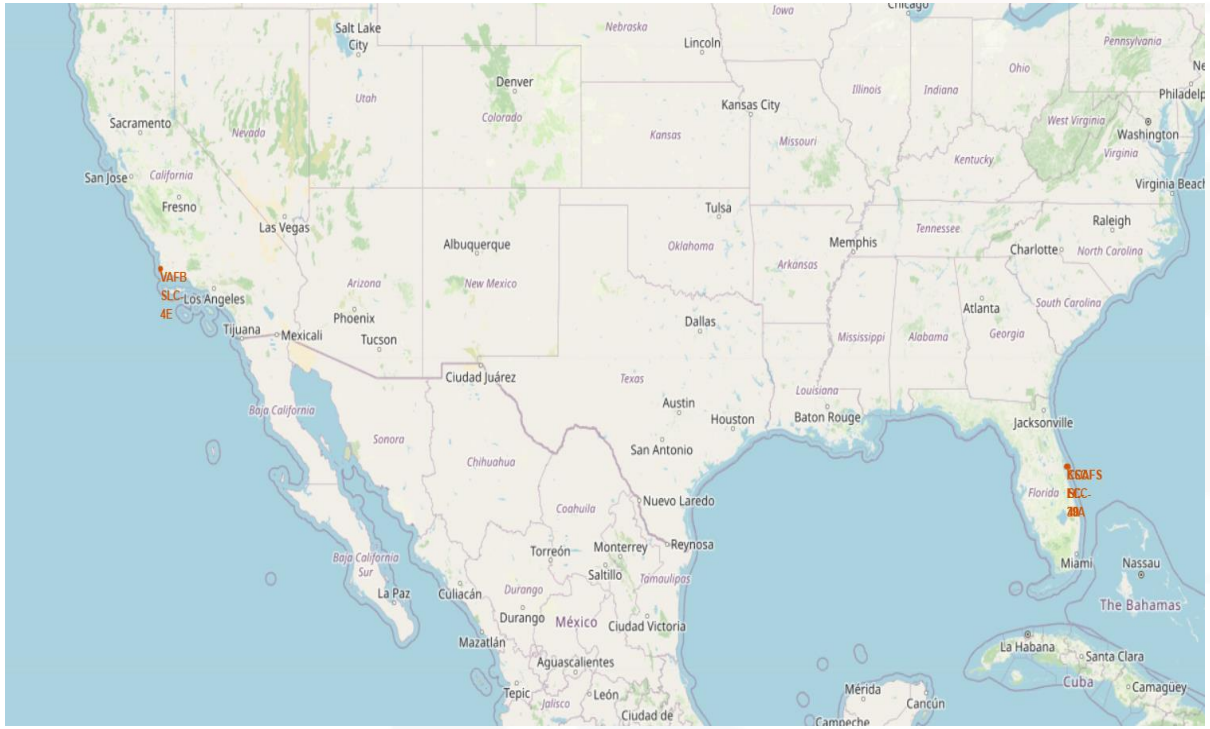
```
 * sqlite:///my_data1.db
Done.
```

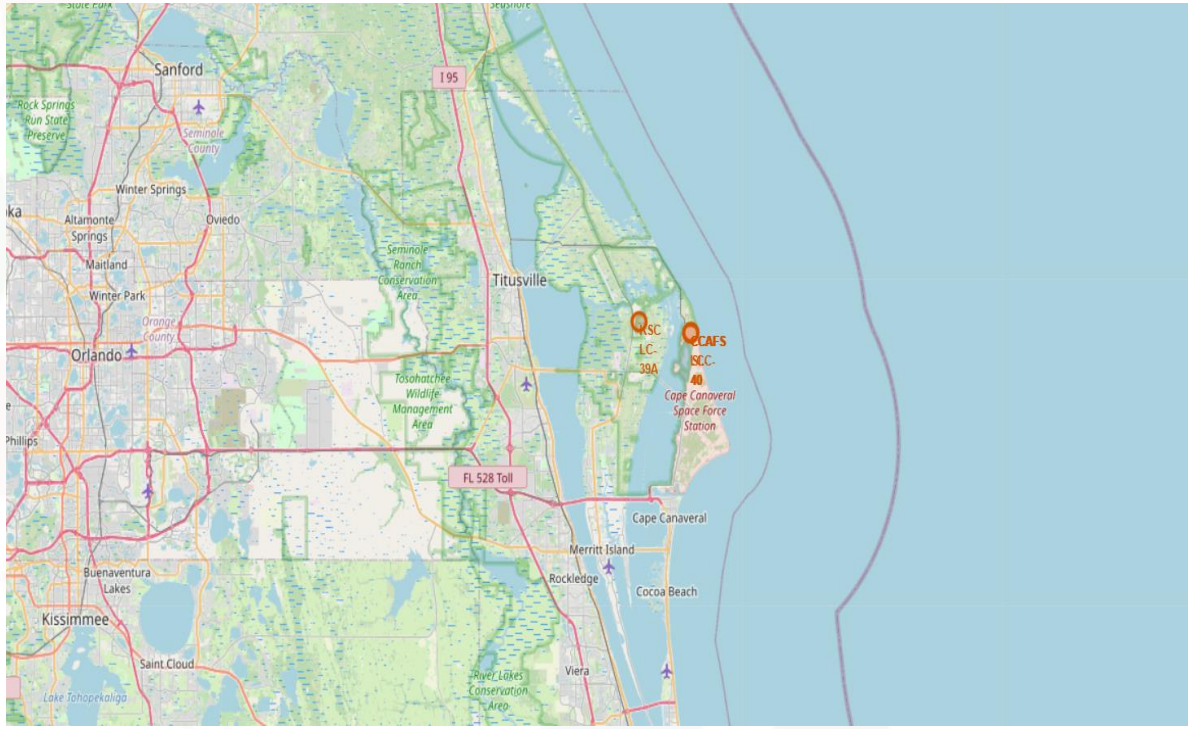
Landing_Outcome	count(*)
Success (drone ship)	12
No attempt	12
Success (ground pad)	8
Failure (drone ship)	5
Controlled (ocean)	4
Uncontrolled (ocean)	2
Precluded (drone ship)	1

+ Code + Markdown

- This query returns the first successful ground pad landing date.
- This query returns a list of successful landings and between 2010-06-04 and 2017-03-20 inclusively.
- There are two types of successful landing outcomes: drone ship and ground pad landings.
- There were 8 successful landings in total during this time period

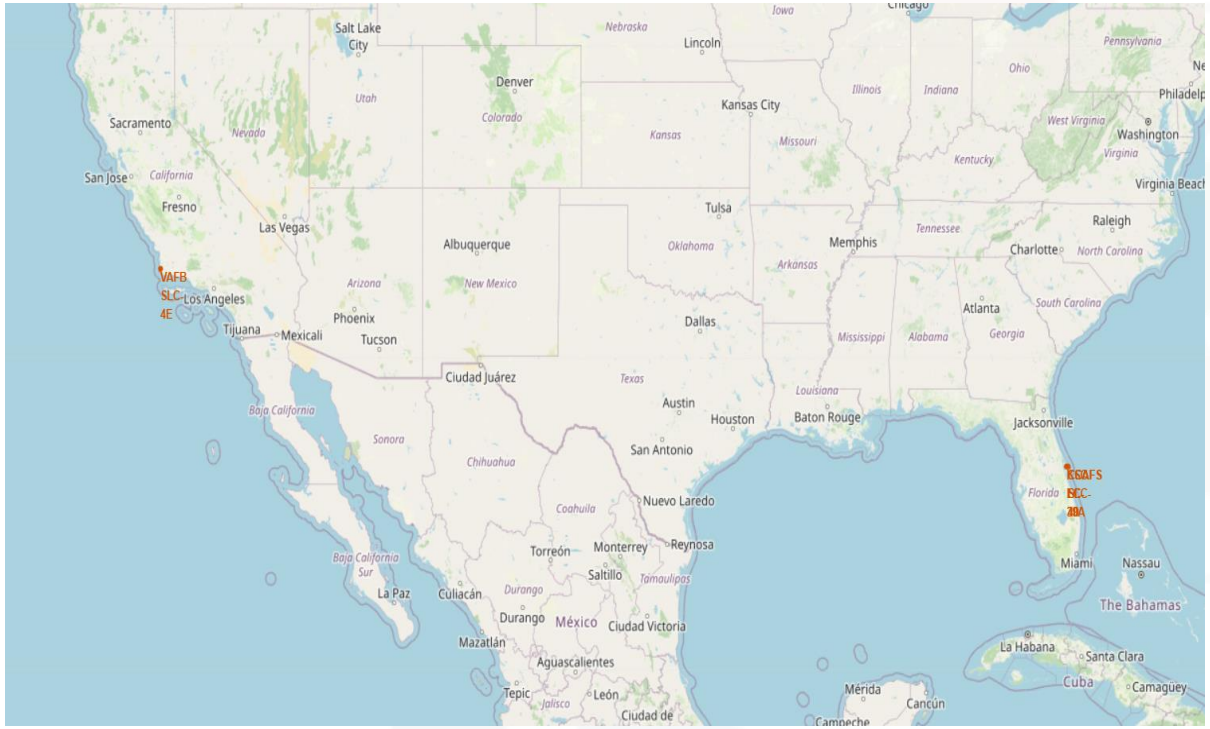
Folium – Site Locations

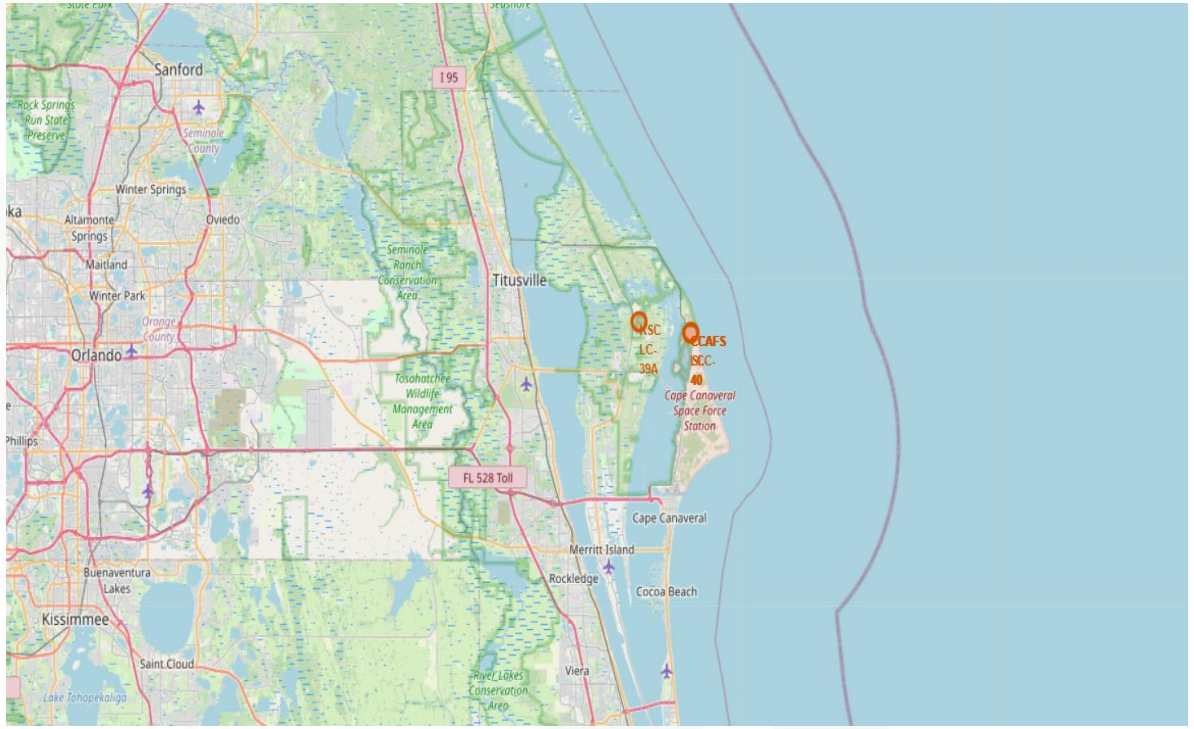




- The left map shows all launch sites relative US map. The right map shows the two Florida launch sites since they are very close to each other.
- All launch sites are near the ocean.

Folium – Site Locations

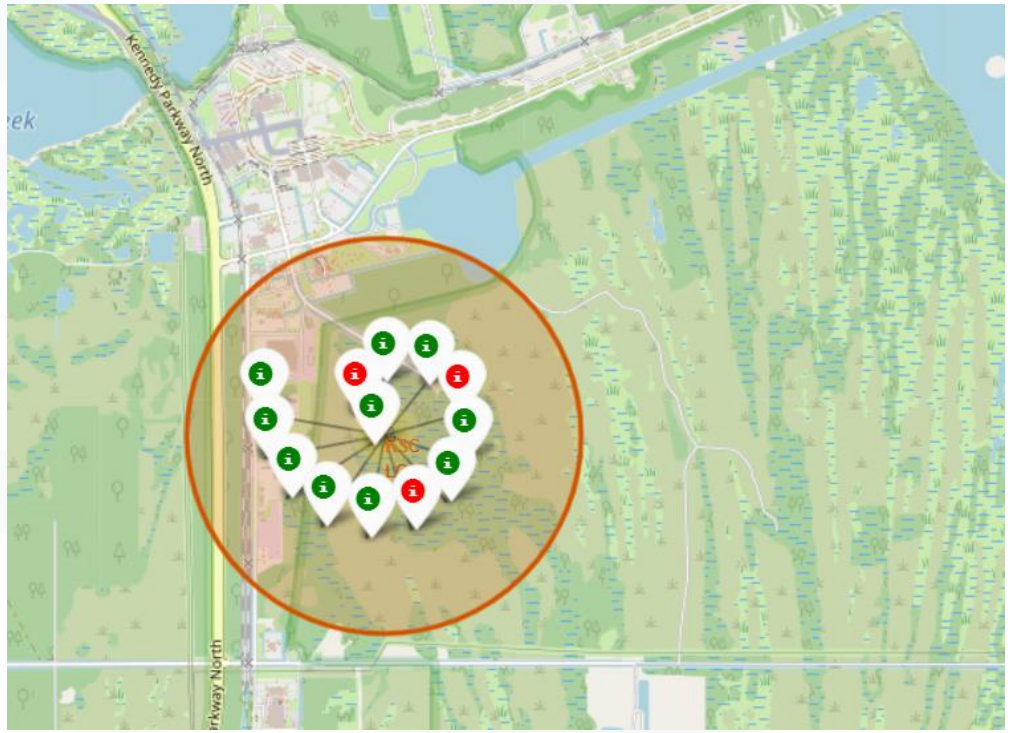




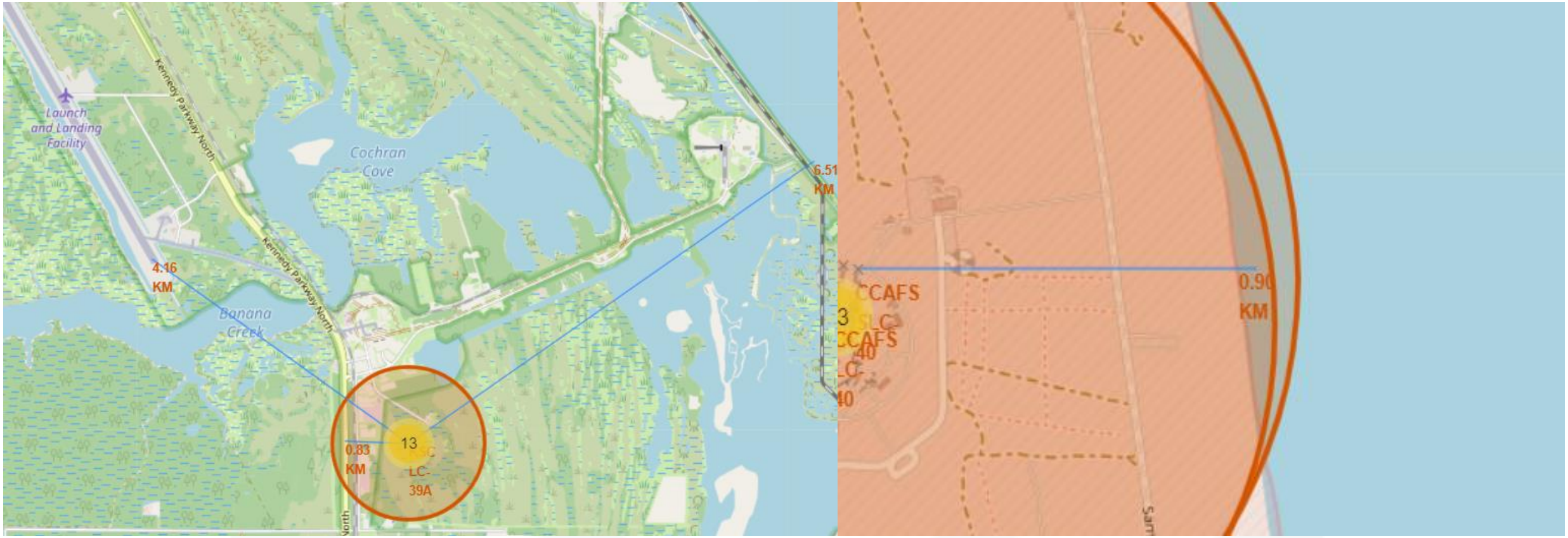
- The left map shows all launch sites relative US map. The right map shows the two Florida launch sites since they are very close to each other.
- All launch sites are near the ocean.

Folium – Color Coded Markers

- Clusters on Folium map can be clicked on to display each successful landing (green icon) and failed
- landing (red icon). In this example KSC LC-39A shows 10 successful landings and 3 failed landings

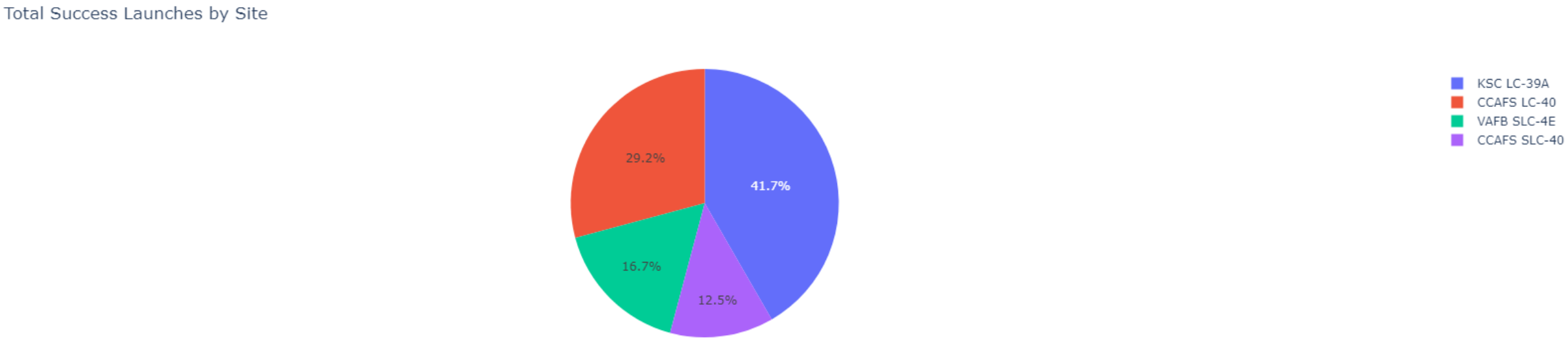


Folium – Site Locations



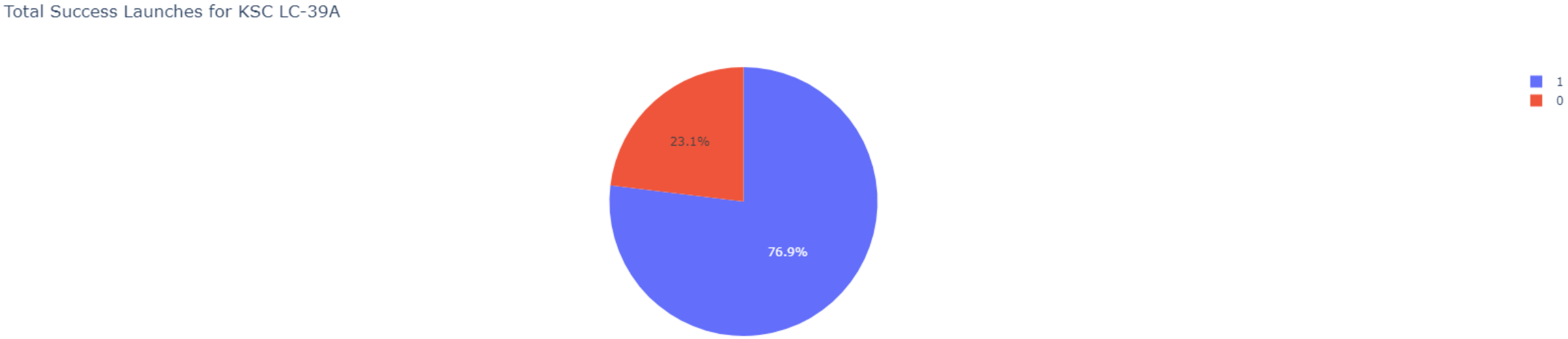
- Using KSC LC-39A as an example, launch sites are very close to railways for large part and supply transportation. Launch sites are close to highways for human and supply transport. Launch sites are also close to coasts and relatively far from cities so that launch failures can land in the sea to avoid rockets falling on densely populated areas.

Plotly – Pie Chart



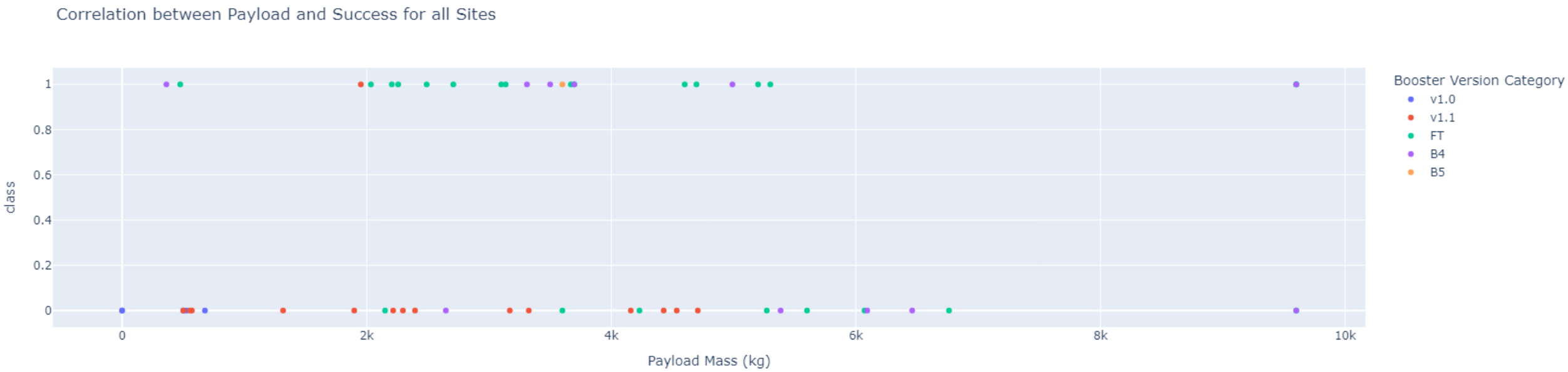
- This is the distribution of successful landings across all launch sites. CCAFS LC-40 is the old name of CCAFS SLC-40 so CCAFS and KSC have the same number of successful landings, but most of the successful landings were performed before the name change. VAFB has the smallest share of successful landings. This may be due to smaller sample and increase in difficulty of launching in the west coast.

Plotly – Pie Chart



- KSC LC-39A has the highest success rate with 10 successful landings and 3 failed landings.

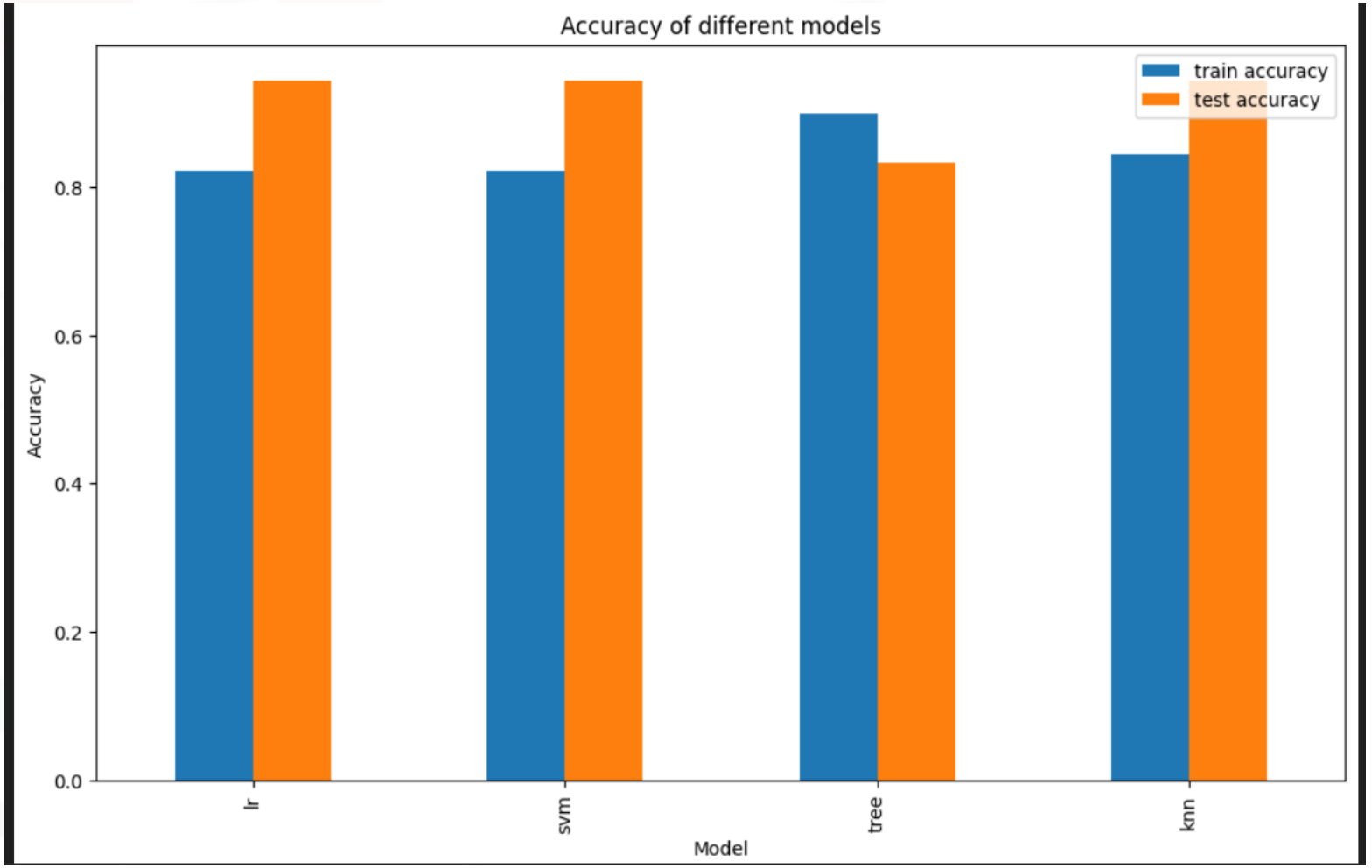
Plotly- Scatter Plot



- Plotly dashboard has a Payload range selector. However, this is set from 0-10000 instead of the max Payload of 15600. Class indicates 1 for successful landing and 0 for failure. Scatter plot also accounts for booster version category in color and number of launches in point size.
- In this particular range of 0-6000, interestingly there are two failed landings with payloads of zero kg.

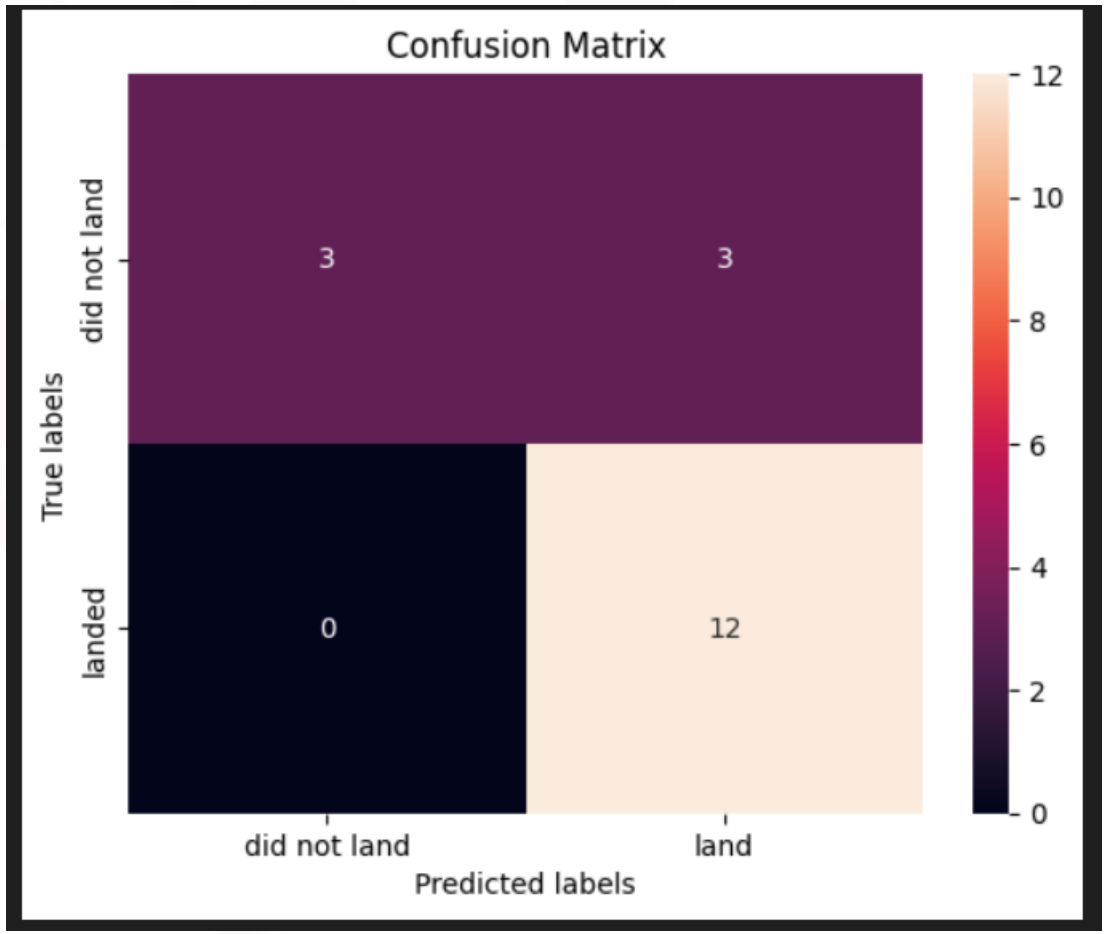
Predictive Analysis

- Mostly the models have same accuracy because dataset size is small
- We will probably need a bigger data size to get some variation is scores



Confusion Matrix

- Since all models performed the same for the test set, the confusion matrix is the same across all models.
- The models predicted 12 successful landings when the true label was successful landing.
- The models predicted 3 unsuccessful landings when the true label was unsuccessful landing.
- The models predicted 3 successful landings when the true label was unsuccessful landings (false positives).
- Our models over predict successful landings.



Conclusion

- Creating a machine learning model for Space Y to bid against SpaceX is our task.
- Creating a machine learning model for Space Y to bid against SpaceX is our task. The model aims to forecast the successful landing date of Stage 1 to save approximately $100 million USD.
- Used information retrieved from the SpaceX Wikipedia page and a public SpaceX API.
- DB2 SQL database was used to store the data labels that were created.
- Made a visualization dashboard

Conclusion

- We developed an 83% accurate machine learning model.
- Using this model, Allon Mask of SpaceY can predict, with a fair degree of accuracy, if a launch will result in a successful Stage 1 landing prior to launch, thereby enabling the determination of whether the launch is warranted.
- To increase accuracy and choose the optimal machine learning model, more data should ideally be gathered.

APPENDIX

- Github Repo Link –
- Capstone_Project_URL