IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

<KHAN SAAHIL ALAM>
<14/07/2024>

Outline

Executive Summary

Methodology:

- Data collected through web scraping and using SpaceX REST API
- Performed data wrangling to created successful/fail outcome variable
- Carried out exploratory data analysis using SQL and data visualization
- Built a dashboard visualizing launch sites with the most success and successful payload ranges
- Built classification models to predict landing outcomes using logistic regression, support vector machine, K-nearest neighbor and decision tree

Results:

- Launch success has improved throughout the years
- KSC LC-39A has the highest success rate among landing sites
- Payload Mass contributed most to the chances of a launch being successful given the type of orbit type and booster version
- Orbits ES-L1, GEO, HEO, and SSO have a 100% success rate
- Most launch sites are close to the coast
- All models performed similarly on the test set where the decision tree model slightly outperformed

Introduction

- SpaceX, a leader in the space industry, is on a mission to enable affordable space travel for the general public. The novel reuse of its Falcon 9 rocket has resulted in the relatively inexpensive rocket launches, with a cost of 62 million dollars. Other providers that are not able to reuse its first stage, cost upwards of 165 million dollars each. Therefore, **if we can determine whether 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:
  - How payload mass, launch site, number of flights, and orbits affect first-stage landing success
  - Does a launch site's location and proximity affect the success rate?
  - Best predictive model for successful landing

Section 1

Methodology

Data Collection

- The data on the Falcon 9 first-stage landings were collecting using two methods. For the first method, I gathered data using an API, specifically the SpaceX REST API. The data collected included the type of rockets used, payload delivered, launch specifications, landing specifications and landing outcome. This data was used to predict whether SpaceX will attempt to launch a rocket or not. For the second method, I web scraped tables on Wiki pages containing Falcon 9 launch records using the Python BeautifulSoup package.

[102]:

	FlightNumber	Date	BoosterVersion	PayloadMass	Orbit	LaunchSite	Outcome	Flights	GridFins	Reused	Legs	LandingPad	Block	ReusedCount	Serial	Longitude	Latitude
4	1	2010-06-04	Falcon 9	NaN	LEO	CCSFS SLC 40	None None	1	False	False	False	None	1.0	0	B0003	-80.577366	28.561857
5	2	2012-05-22	Falcon 9	525.0	LEO	CCSFS SLC 40	None None	1	False	False	False	None	1.0	0	B0005	-80.577366	28.561857
6	3	2013-03-01	Falcon 9	677.0	ISS	CCSFS SLC 40	None None	1	False	False	False	None	1.0	0	B0007	-80.577366	28.561857
7	4	2013-09-29	Falcon 9	500.0	PO	VAFB SLC 4E	False Ocean	1	False	False	False	None	1.0	0	B1003	-120.610829	34.632093
8	5	2013-12-03	Falcon 9	3170.0	GTO	CCSFS SLC 40	None None	1	False	False	False	None	1.0	0	B1004	-80.577366	28.561857
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
91	86	2020-09-03	Falcon 9	15600.0	VLEO	KSC LC 39A	True ASDS	2	True	True	True	5e9e3032383ecb6bb234e7ca	5.0	12	B1060	-80.603956	28.608058
92	87	2020-10-06	Falcon 9	15600.0	VLEO	KSC LC 39A	True ASDS	3	True	True	True	5e9e3032383ecb6bb234e7ca	5.0	13	B1058	-80.603956	28.608058
93	88	2020-10-18	Falcon 9	15600.0	VLEO	KSC LC 39A	True ASDS	6	True	True	True	5e9e3032383ecb6bb234e7ca	5.0	12	B1051	-80.603956	28.608058
94	89	2020-10-24	Falcon 9	15600.0	VLEO	CCSFS SLC 40	True ASDS	3	True	True	True	5e9e3033383ecbb9e534e7cc	5.0	12	B1060	-80.577366	28.561857
95	90	2020-11-05	Falcon 9	3681.0	MEO	CCSFS SLC 40	True ASDS	1	True	False	True	5e9e3032383ecb6bb234e7ca	5.0	8	B1062	-80.577366	28.561857

90 rows × 17 columns

Data Collection – SpaceX API

- Imported libraries
- Defined a series of helper functions to help use the API to extract information using identification numbers in the launch data.
- Requested and parsed the SpaceX launch data using the GET request
- Used the API again to get information about the launches using the IDs given for each launch.
- The data from these requests were stored in lists and used to create a new dataframe.
- Filtered the dataframe to only include Falcon 9 launches.
- Replaced np.nan values in the PayloadMass column with the .mean() function.
- GitHub URL

Data Collection – Scraping

- Imported required libraries
- Performed an HTTP GET method to request the Falcon9 Launch HTML page, as an HTTP response.
- Created a BeautifulSoup object from the HTML response.
- Extracted all column/variable names from the HTML table header.
- Created an empty dictionary with keys from the extracted column names.
- Filled up the launch_dict with launch records extracted from table rows.
- Created a dataframe from the launch_dict
- [GitHub URL]

Out[15]:

	Flight No.	Launch site	Payload	Payload mass	Orbit	Customer	Launch outcome	Version Booster	Booster landing	Date	Time
0	1	CCAFS	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success\n	F9 v1.0B0003.1	Failure	4 June 2010	18:45
1	2	CCAFS	Dragon	0	LEO	NASA	Success	F9 v1.0B0004.1	Failure	8 December 2010	15:43
2	3	CCAFS	Dragon	525 kg	LEO	NASA	Success	F9 v1.0B0005.1	No attempt\n	22 May 2012	07:44
3	4	CCAFS	SpaceX CRS-1	4,700 kg	LEO	NASA	Success\n	F9 v1.0B0006.1	No attempt	8 October 2012	00:35
4	5	CCAFS	SpaceX CRS-2	4,877 kg	LEO	NASA	Success\n	F9 v1.0B0007.1	No attempt\n	1 March 2013	15:10

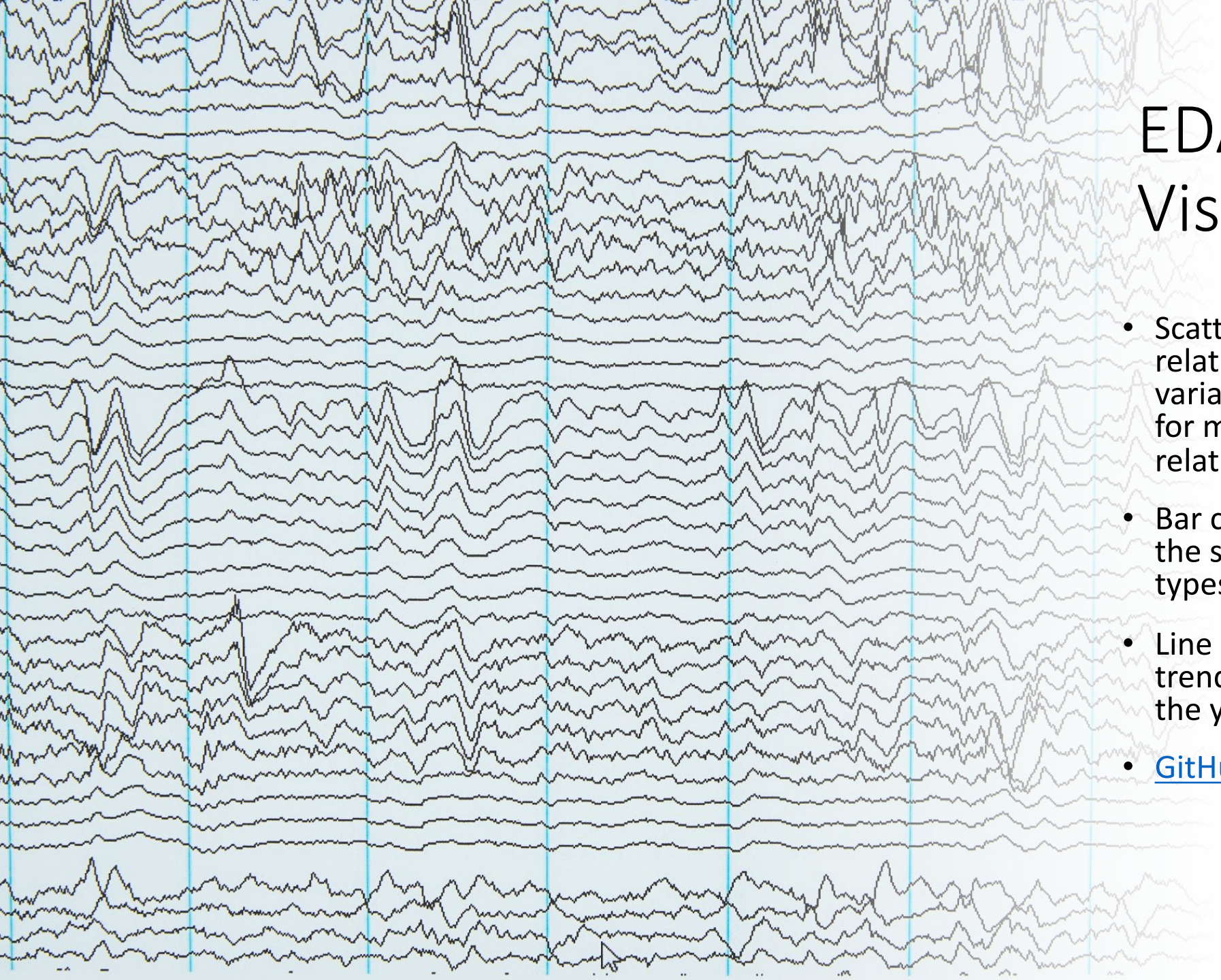
Data Wrangling

- Performed exploratory data analysis using the pandas and numpy libraries
- Determined the training labels
  - Created a binary landing outcome columns (dependent variable)
  - This variable will represent the classification variable that represents the outcome of each launch. If the value is zero, the first stage did not land successfully; one means the first stage landed Successfully
  - GitHub URL

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

[12]:

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



EDA with Data Visualization

- Scatter plots were used to deduce relationships between different variables, which could be useful for machine learning if a relationship exists
- Bar chart was used to compare the success rates of different orbit types
- Line chart was used to display the trend of success rate throughout the years
- GitHub URL

EDA with SQL

Using SQL queries, some findings I discovered:

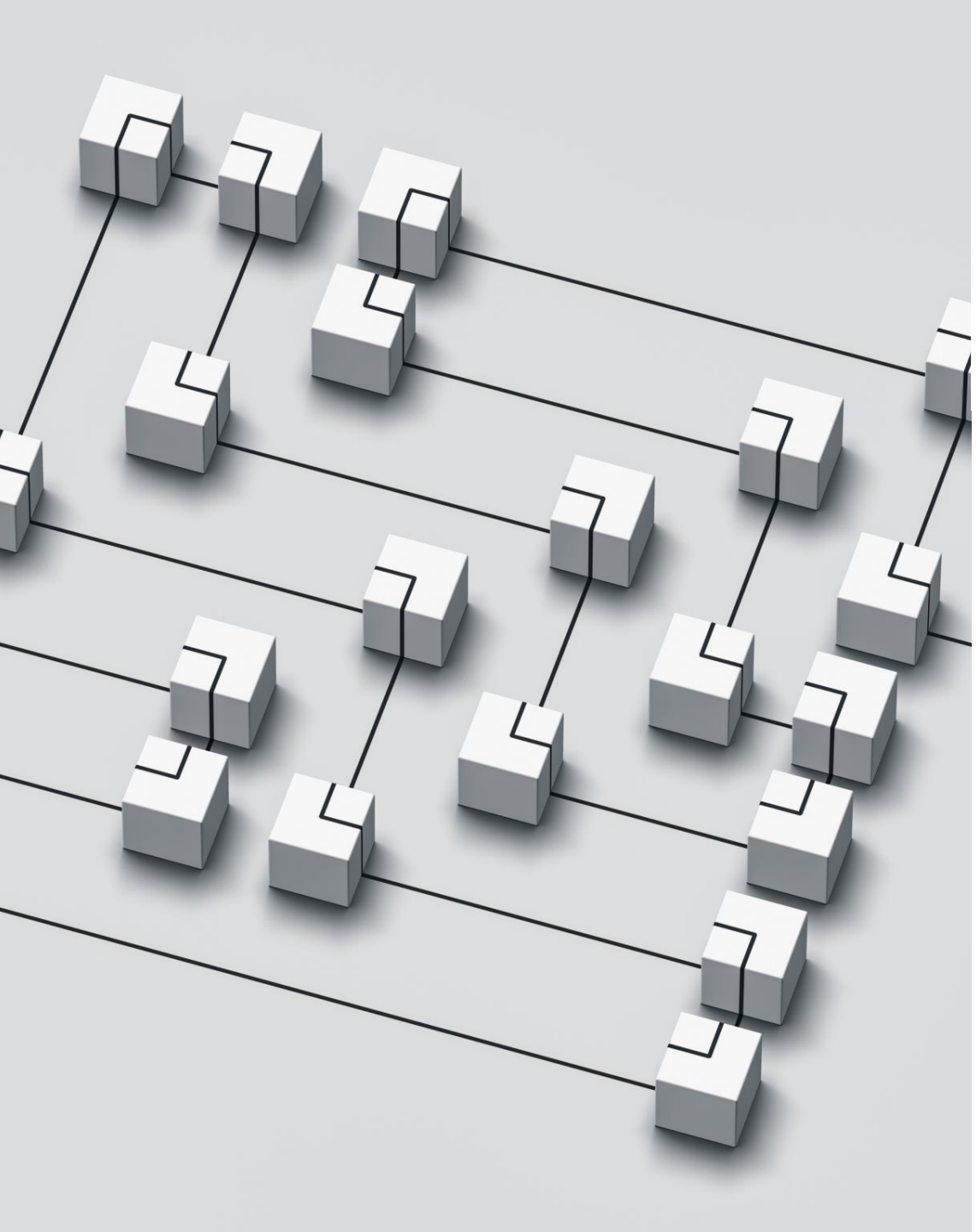
- Names of unique launch sites
- The occurrences of different landing outcomes
- The total and average pay load mass of certain boosters
- Boosters that had a successful landing in a drone ship with certain pay load mass range
- Boosters that carried the maximum pay load mass
- GitHub URL

Build an Interactive Map with Folium

- Created and added a Circle and Marker object for each launch site based on its coordinates
- Created a MarketCluster object since many launch records have the exact same coordinates
- Based on the ‘class’ column in the dataframe, marked the success of a launch green if ‘class=1’ and failure of a launch red if ‘class=0’
- Calculated the distances between a launch site and its proximities to the nearest coastline
- GitHub URL

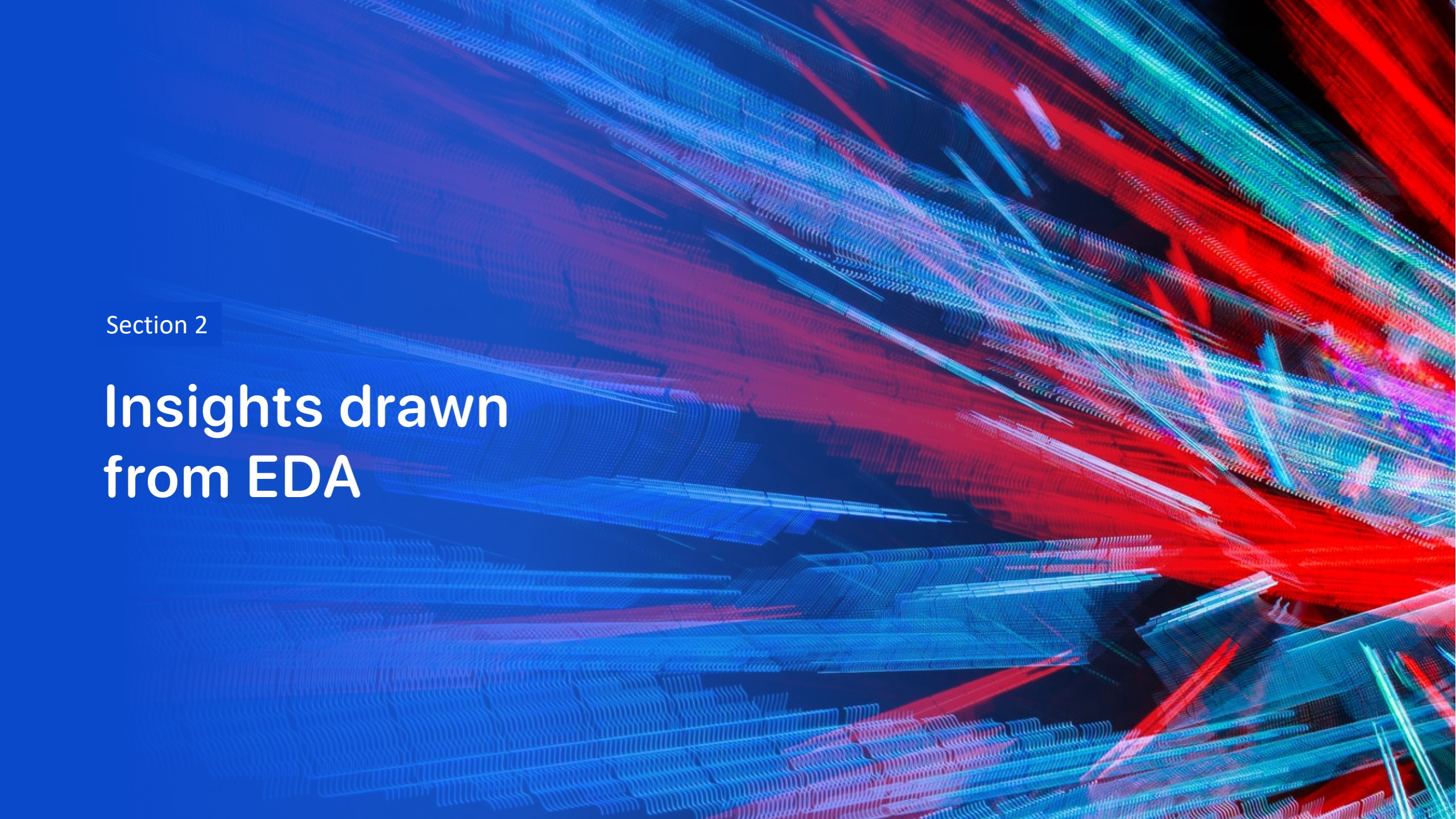
Build a Dashboard with Plotly Dash

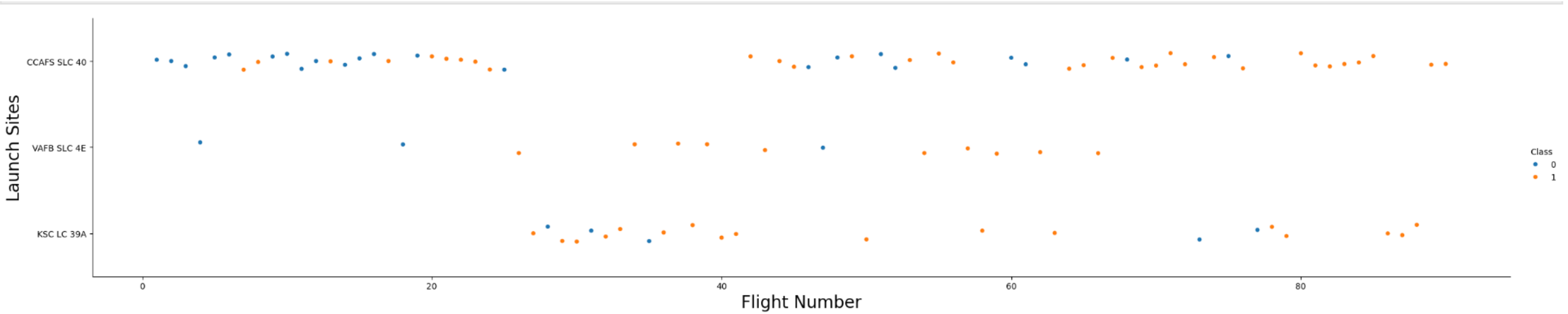
- Added a pie chart and a scatter plot with a dropdown list and range slider respectively
- Added these plots to easily gain insights on launch sites and payload ranges that had highest success rate
- GitHub URL



Predictive Analysis (Classification)

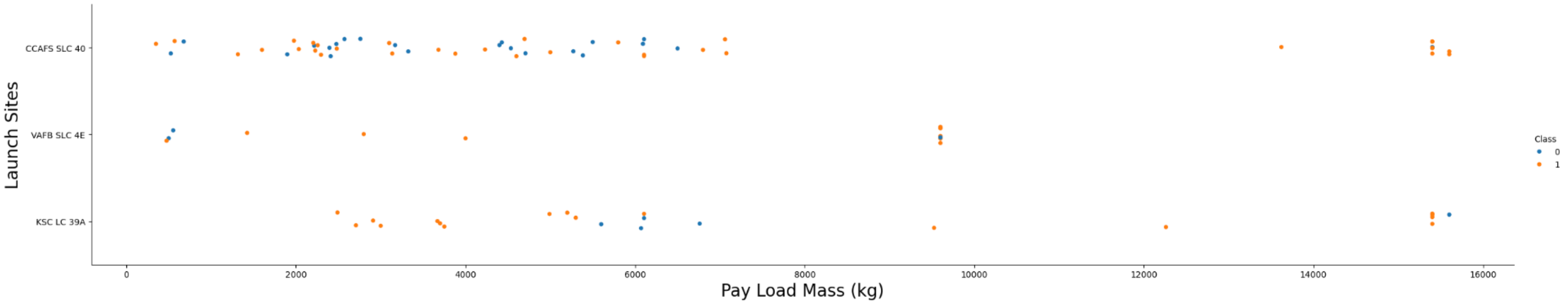
- Created NumPy array from the Class column
- Standardized the data with StandardScaler
- Split the data using train_test_split
- Found the best hyperparameter for each model by creating a GridSearchCV object with cv=10
- Applied GridSearchCV on different algorithms
- Calculated the accuracy on the test data using .score()
- Created a confusion matrix for all models
- Compared the F1_Score, Jaccard_Score snd Accuracy to choose the best performing classification model
- GitHub URL


Section 2
Insights drawn from EDA



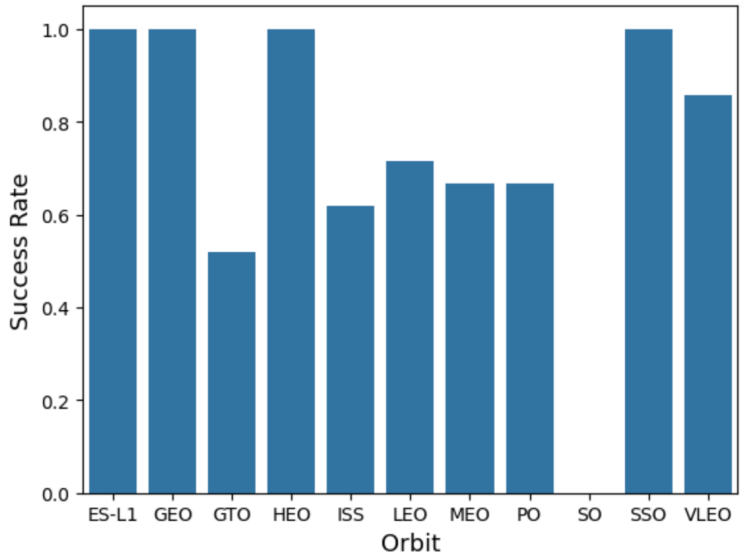
Flight Number vs. Launch Site

- From the scatter plot, we can see that as the flight number increases, the first stage is more likely to land successfully. We can infer that new launches have a higher success rate.



Payload vs. Launch Site

- From the scatter plot, we can see that the higher the payload mass (kg), the higher the success rate. Most launces with a payload greater than 7,000 kg were successful. It can be particularly seen in the case of CCAFS SLC 40 that the first stage is less likely to land successfully for a lower pay load mass.

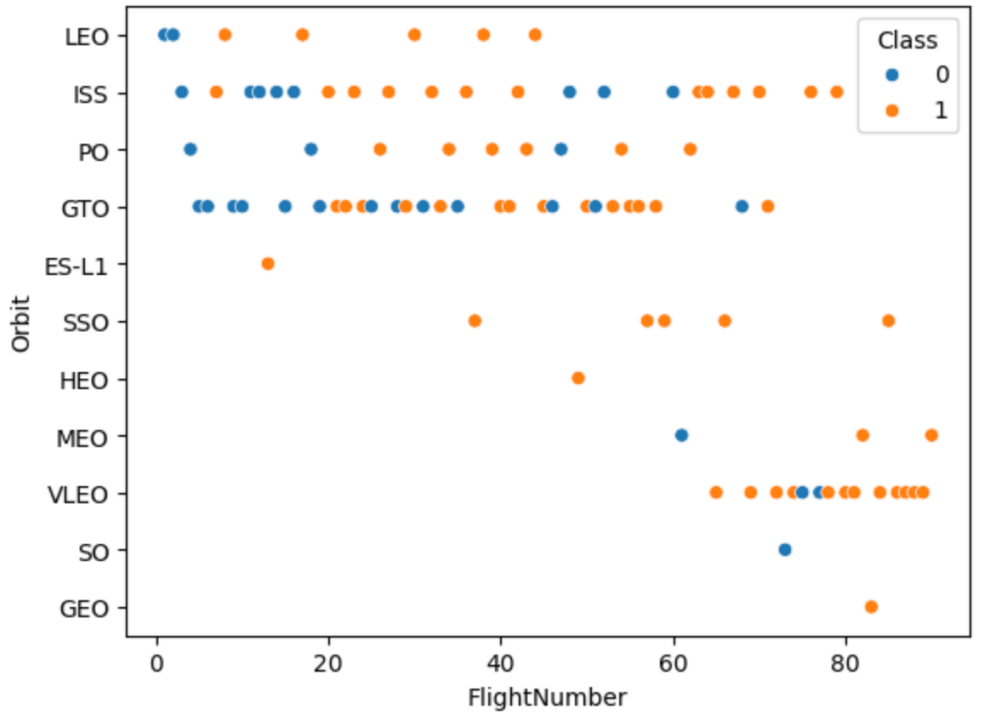


Success Rate vs. Orbit Type

- From the bar chart, we can see orbit types ES-L1, GEO, HEO and SSO have 100% success rate while orbit type SO has 0% success rate.

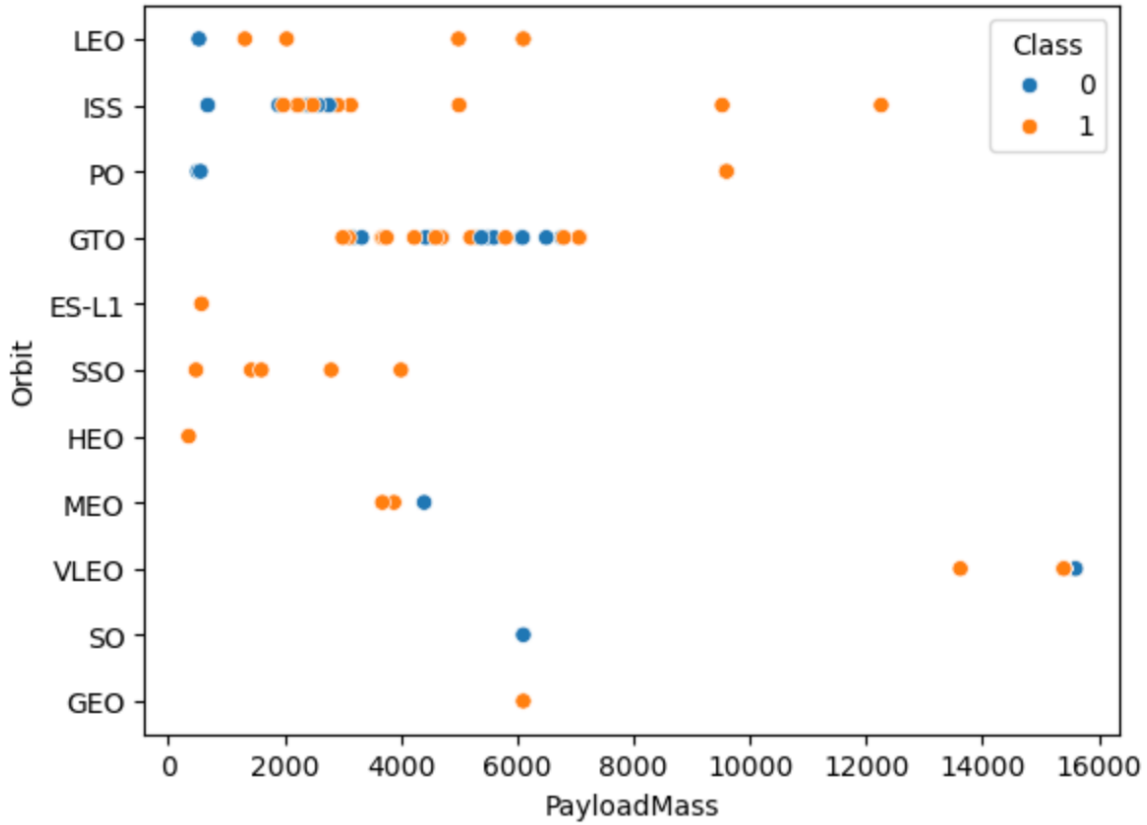
Flight Number vs. Orbit Type

- From the scatter plot, we can't directly deduce a relationship between flight numbers and successful landings. For orbit type LEO, the successful landings are more likely when flight number increases. On the other hand, there seems to be no relationship between flight number and orbit type GTO.

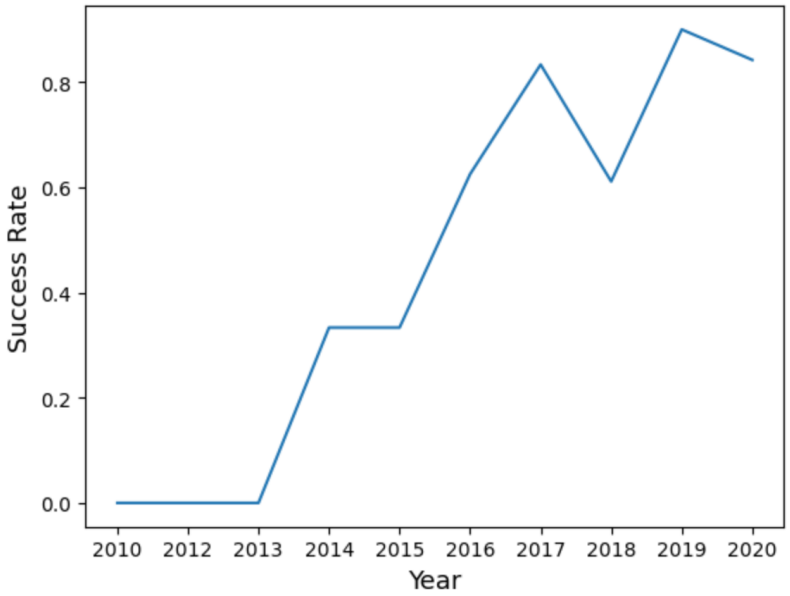


Payload vs. Orbit Type

- From the scatter plot, we can see that the likelihood of successful landing increases with greater pay load mass for orbit types LEO, ISS and PO.



Launch Success Yearly Trend



We can that the success rate has been on an upward trend since 2013.

All Launch Site Names

The names of the launch sites are CCAFS LC-40, VAFB SLC-4E, KSC LC-39A, CCAFS SLC-40.

```
[11]: %sql SELECT DISTINCT Launch_Site FROM SPACEXTABLE;
```

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

[11]:

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

```
[12]: %sql SELECT Launch_Site FROM SPACEXTABLE WHERE Launch_Site LIKE 'CCA%' LIMIT 5
 * sqlite:///my_data1.db
Done.
```

[12]:

Launch_Site
CCAFS LC-40
CCAFS LC-40
CCAFS LC-40
CCAFS LC-40
CCAFS LC-40

Total Payload Mass

The total payload mass carried by boosters launched by NASA (CRS) is 45596 kg.

```
[13]: %sql SELECT SUM(PAYLOAD_MASS__KG_) AS 'Total Payload Mass' FROM SPACEXTABLE WHERE Customer = 'NASA (CRS)'
 * sqlite:///my_data1.db
Done.
```

[13]:

Total Payload Mass
45596

Average Payload Mass by F9 v1.1

The average payload mass carried by booster version F9 v1.1 is 2928.4 kg

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

```
[14]: %sql SELECT AVG(PAYLOAD_MASS__KG_) AS 'Average Payload Mass' FROM SPACEXTABLE WHERE Booster_Version = 'F9 v1.1'
```

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

[14]:

Average Payload Mass
2928.4

First Successful Ground Landing Date

The first successful landing outcome on ground pad was on 22/12/2015

```
[15]: %sql SELECT MIN(DATE) AS 'First Successful Landing in Ground Pad' FROM SPACEXTABLE WHERE Landing_Outcome = 'Success (ground pad)';
```

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

[15]:

First Successful Landing in Ground Pad
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

Names of boosters which have successfully landed on drone ship and had payload mass greater than 4000 but less than 6000

[16]:
```
%sql SELECT Booster_Version FROM SPACEXTABLE WHERE Landing_Outcome = 'Success (drone ship)' AND 4000<Payload_Mass__kg_<6000;
```

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

[16]:

Booster_Version
F9 FT B1021.1
F9 FT B1022
F9 FT B1023.1
F9 FT B1026
F9 FT B1029.1
F9 FT B1021.2
F9 FT B1029.2
F9 FT B1036.1
F9 FT B1038.1
F9 B4 B1041.1
F9 FT B1031.2
F9 B4 B1042.1
F9 B4 B1045.1
F9 B5 B1046.1

Total Number of Successful and Failure Mission Outcomes

These are the total number of different mission outcomes

[17]:
```
%sql SELECT Mission_Outcome, COUNT(*) AS 'Total Number' FROM SPACEXTABLE GROUP BY Mission_Outcome;
```

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

[17]:

Mission_Outcome	Total Number
Failure (in flight)	1
Success	98
Success	1
Success (payload status unclear)	1

Boosters Carried Maximum Payload

Names of boosters which have carried the maximum payload mass

```
[21]: %sql SELECT Booster_Version FROM SPACEXTABLE WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTABLE);
 * sqlite:///my_data1.db
Done.
```

[21]:

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

These are the failed landing outcomes in a drone ship along with their booster versions and launch site names in 2015

```
[29]: %sql SELECT substr(Date, 6,2) AS Month, Landing_Outcome, Booster_Version, Launch_Site FROM SPACEXTABLE WHERE Landing_Outcome LIKE 'Failure (drone ship)' AND substr(Date,0,5)='2015';
 * sqlite:///my_data1.db
Done.
```

[29]:

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

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

- These are the number of occurrences of different landing outcomes between 2010-06-04 and 2017-03-20

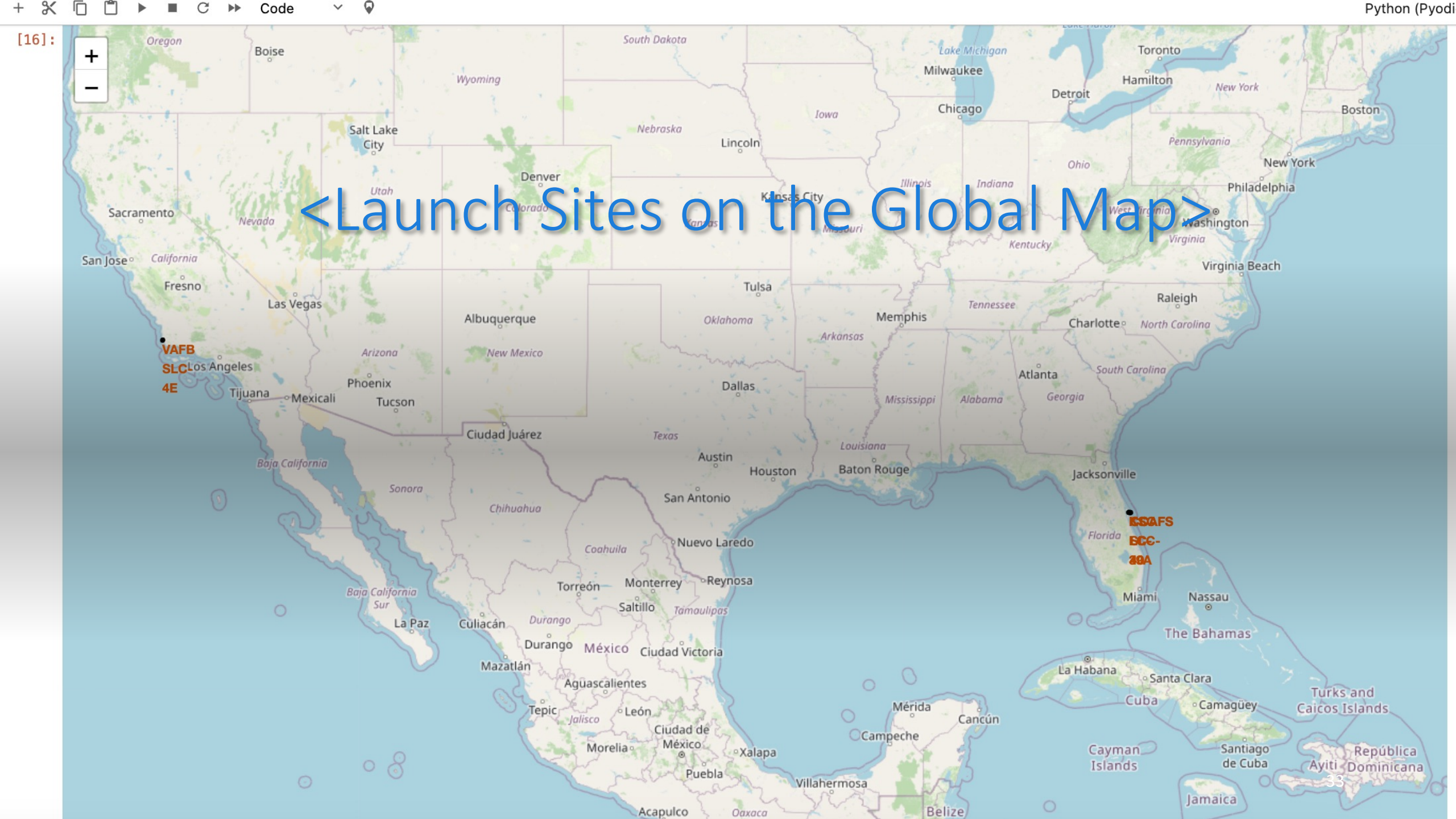
[12]:
```
%sql SELECT Landing_Outcome,COUNT(Landing_Outcome) AS Occurences FROM SPACEXTABLE WHERE 2010-06-04 < Date < 2017-03-20 GROUP BY Landing_Outcome ORDER BY Occurences DESC;
```

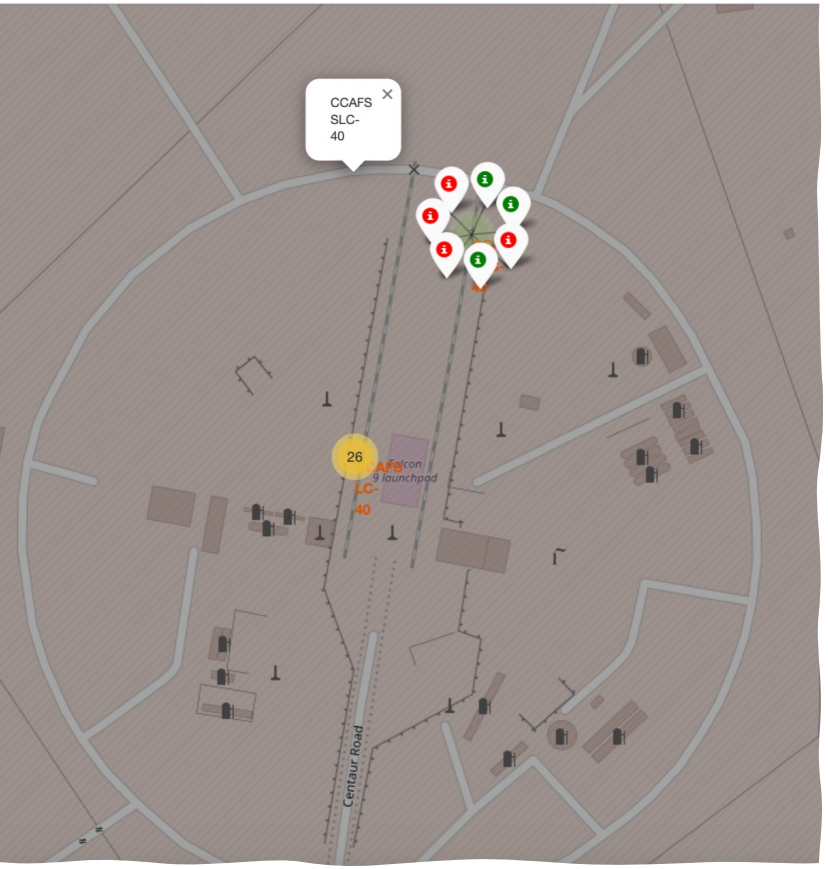
```
 * sqlite:///my_data1.db
Done.
```

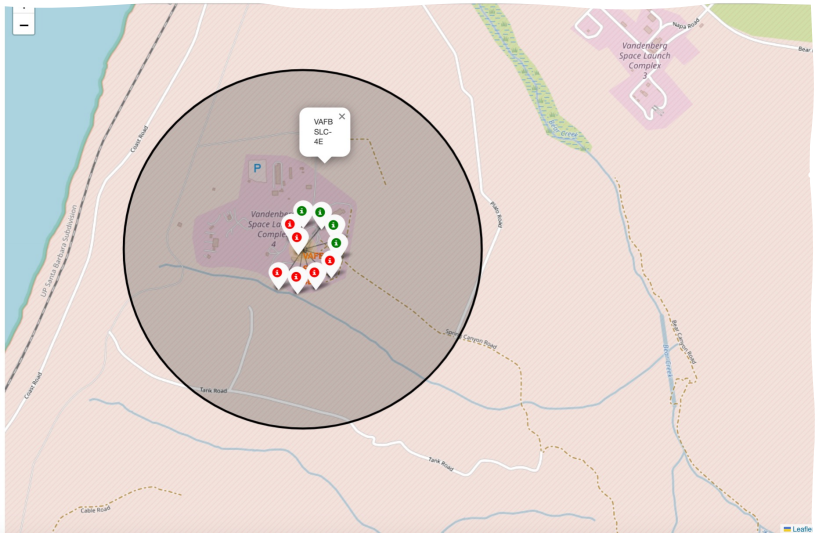
[12]:

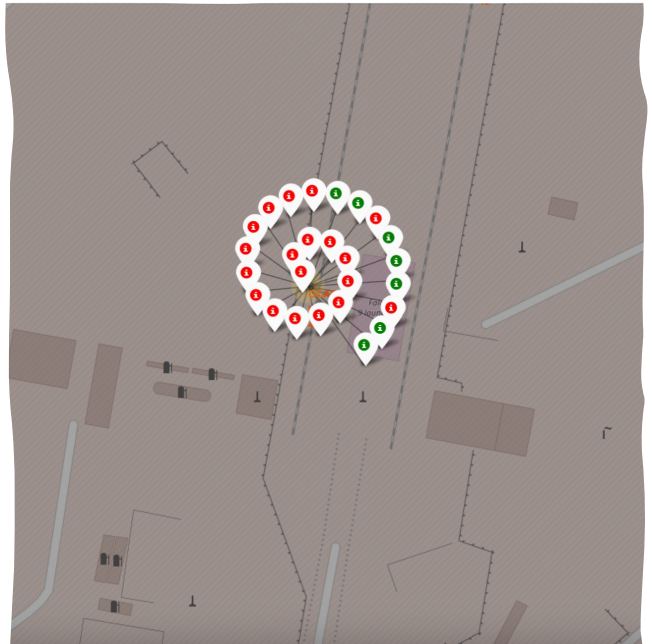
Landing_Outcome	Occurences
Success	38
No attempt	21
Success (drone ship)	14
Success (ground pad)	9
Failure (drone ship)	5
Controlled (ocean)	5
Failure	3
Uncontrolled (ocean)	2
Failure (parachute)	2
Precluded (drone ship)	1
No attempt	1


Section 3
Launch Sites
Proximities Analysis


[16]:
Code
Python (Pyodi
<Launch Sites on the Global Map>
VAFB
SLC
4E
CCAFS
SLC-
40
Oregon
Boise
South Dakota
Wyoming
Lake Michigan
Milwaukee
Toronto
Hamilton
Detroit
Chicago
New York
Boston
Iowa
Nebraska
Salt Lake City
Lincoln
Pennsylvania
Ohio
New York
Denver
Utah
Illinois
Indiana
Philadelphia
Kansas City
Colorado
Sacramento
Nevada
Missouri
Washington
Virginia
Kentucky
San Jose
California
Virginia Beach
Fresno
Tulsa
Raleigh
Las Vegas
Tennessee
Albuquerque
Oklahoma
Memphis
Charlotte
North Carolina
Arkansas
Arizona
New Mexico
Los Angeles
Atlanta
South Carolina
Phoenix
Dallas
Tijuana
Mexicali
Tucson
Mississippi
Alabama
Georgia
Ciudad Juárez
Texas
Louisiana
Austin
Houston
Baton Rouge
Jacksonville
Baja California
Sonora
San Antonio
Chihuahua
Florida
Nuevo Laredo
Coahuila
Torreón
Monterrey
Reynosa
Baja California Sur
Saltillo
Tamaulipas
Miami
Nassau
La Paz
Culiacán
Durango
The Bahamas
Durango
México
Ciudad Victoria
Mazatlán
La Habana
Santa Clara
Aguascalientes
Cuba
Turks and Caicos Islands
Mérida
Cancún
Tepic
León
Jalisco
Camagüey
Ciudad de México
Campeche
Cayman Islands
Morelia
Xalapa
Santiago de Cuba
República Dominicana
Puebla
Ayiti
Villahermosa
Jamaica
Acapulco
Oaxaca
Belize
33



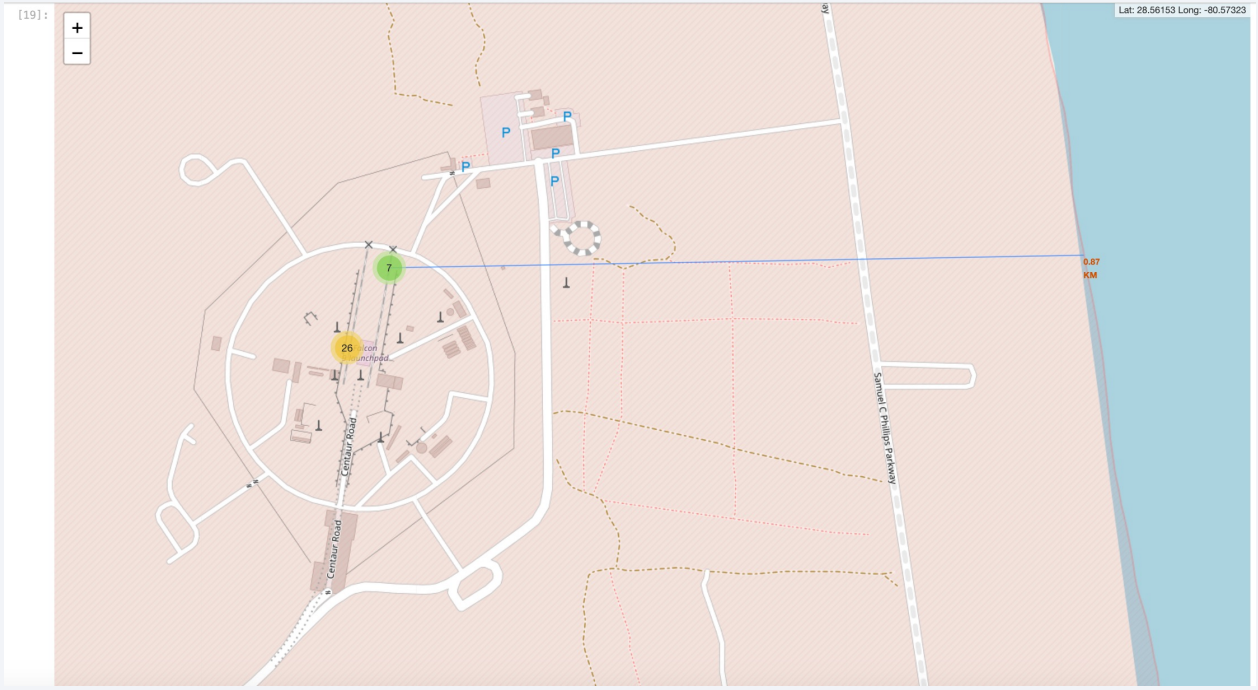




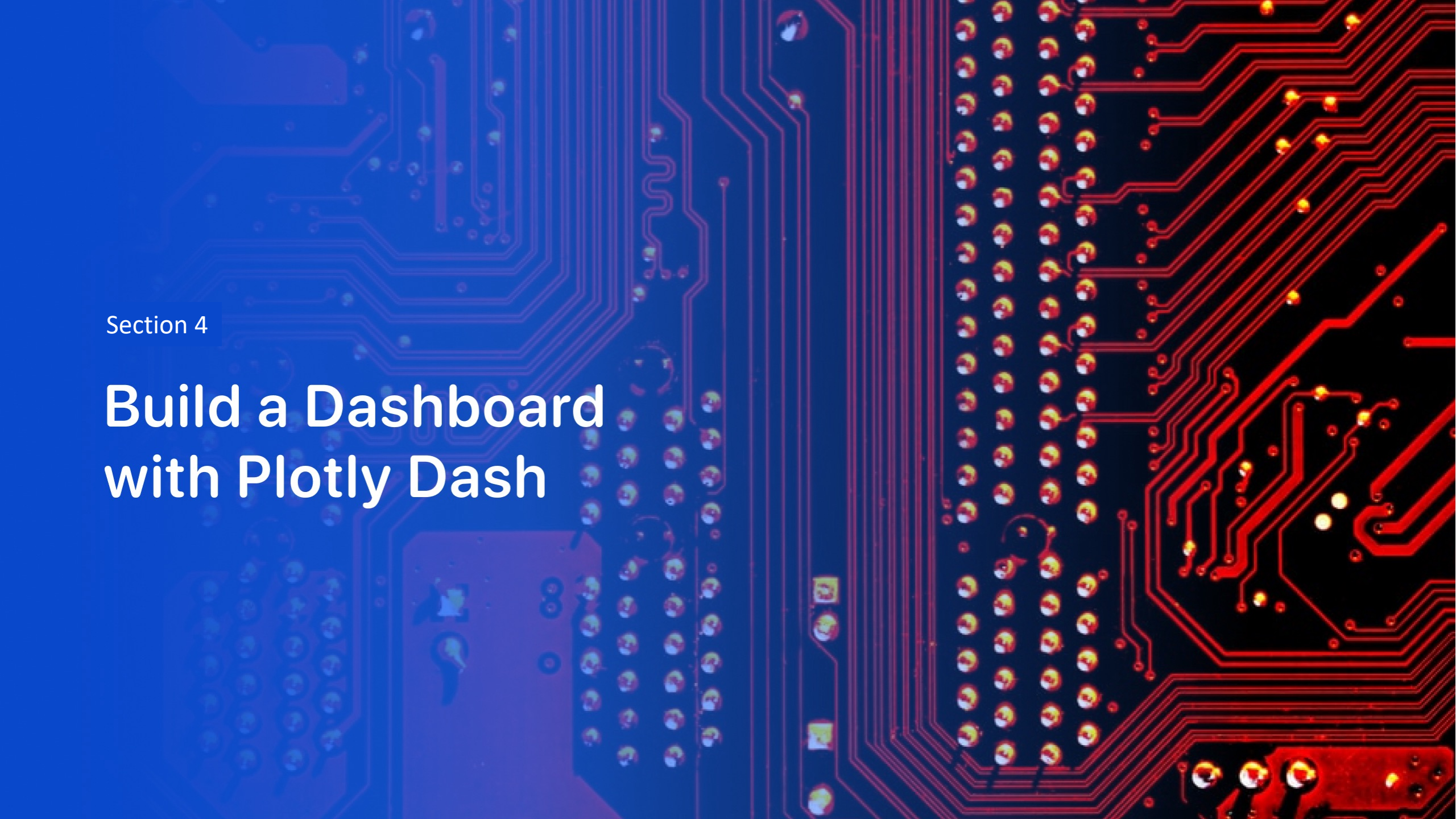
<Launch Sites Success Rate>

Green markers are for successful launches while red are for unsuccessful launches. Launch site KSC LC-39A has the highest success rate while CCAFS LC-40 has the lowest success rate.

<Launch Site Proximity to Coastline>

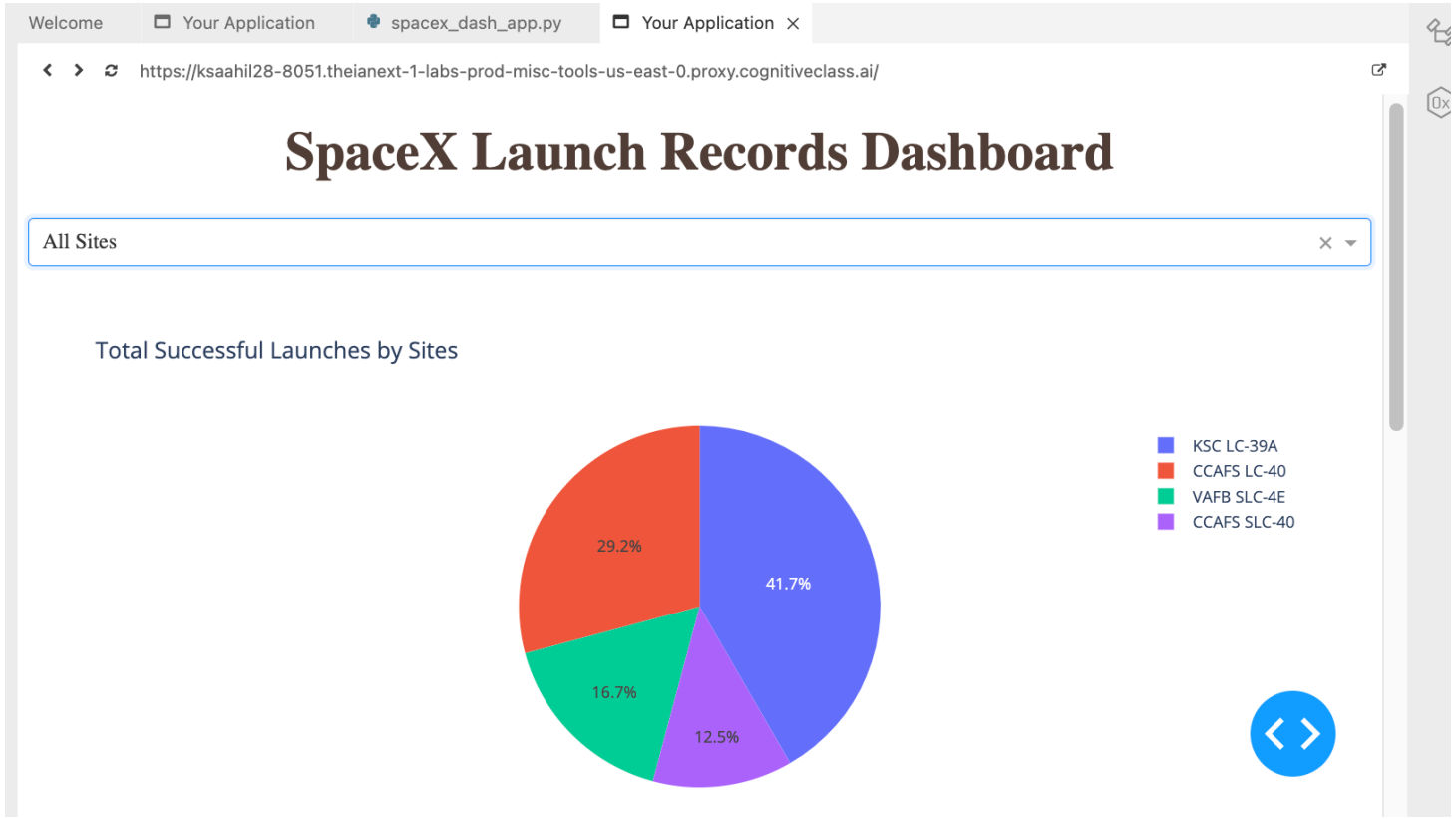


From the screenshot above, we can see that the launch site has a very close proximity to the nearest coastline, only 0.87 km away. This helps ensure that spent stages dropped along the launch path or failed launches don't fall on people or property.


Section 4
Build a Dashboard with Plotly Dash

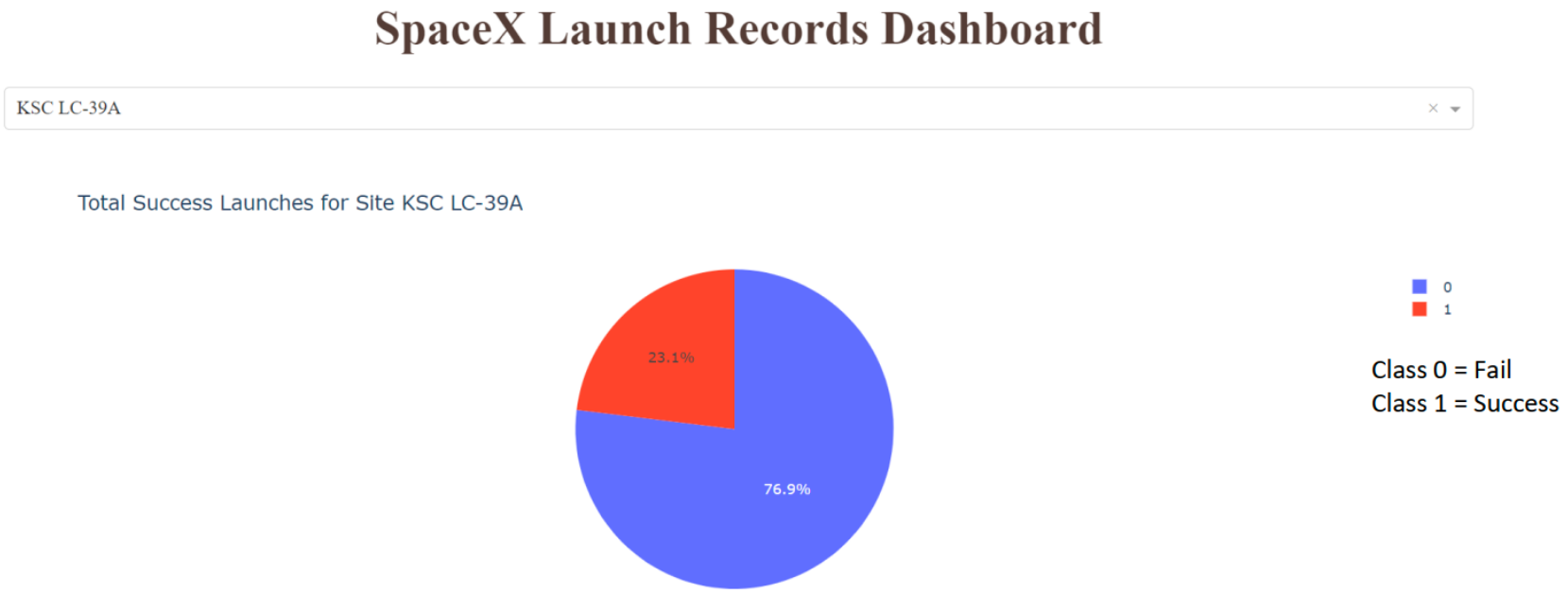
Total Successful Launches for All Sites in Dash

From the pie chart, we can see that the launch site ‘KSC LC-39A’ has the most successful launches while launch site ‘CCAFS SLC-40’ has the least successful launches



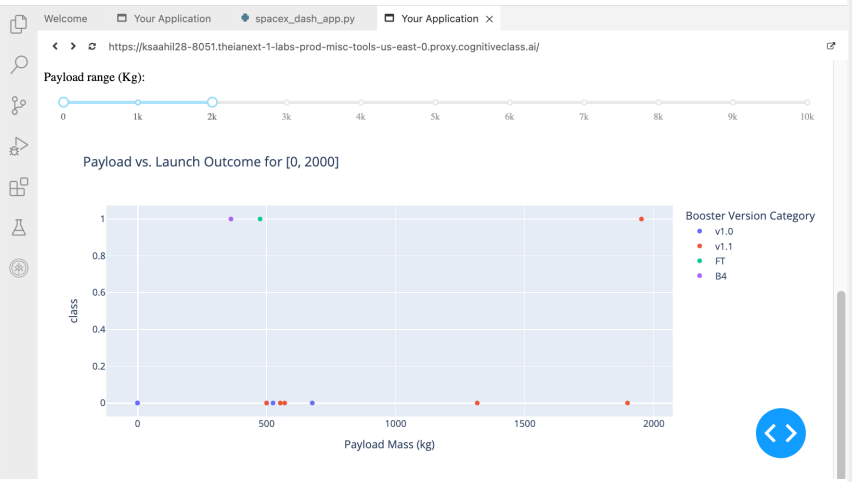
Launch Success of KSC LC-29A

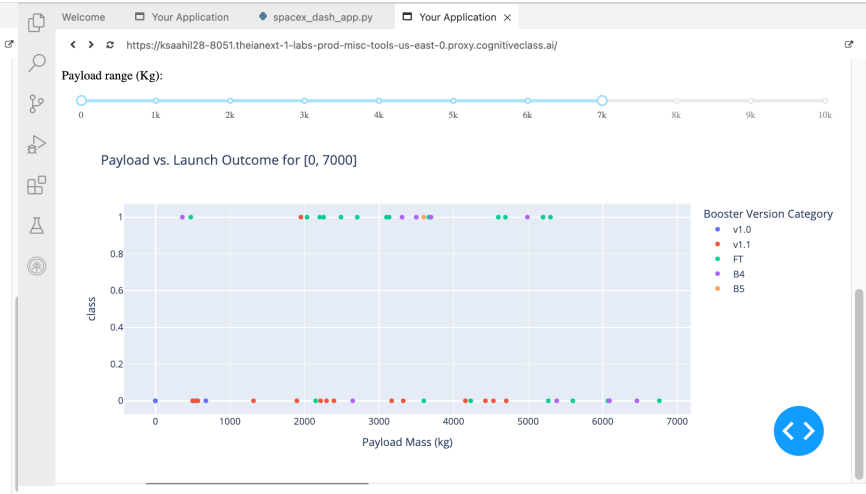
KSC LC-39A has the highest success rate amongst launch sites (76.9%) consisting of 10 successful launches and 3 failed launches



Payload Mass vs Launch Outcome Scatterplot in Dash

From the results, we can see that the payload mass range from 2000 to 5500 kg has the highest success rate where majority of them have the booster version ‘FT’ and ‘B4’





Section 5

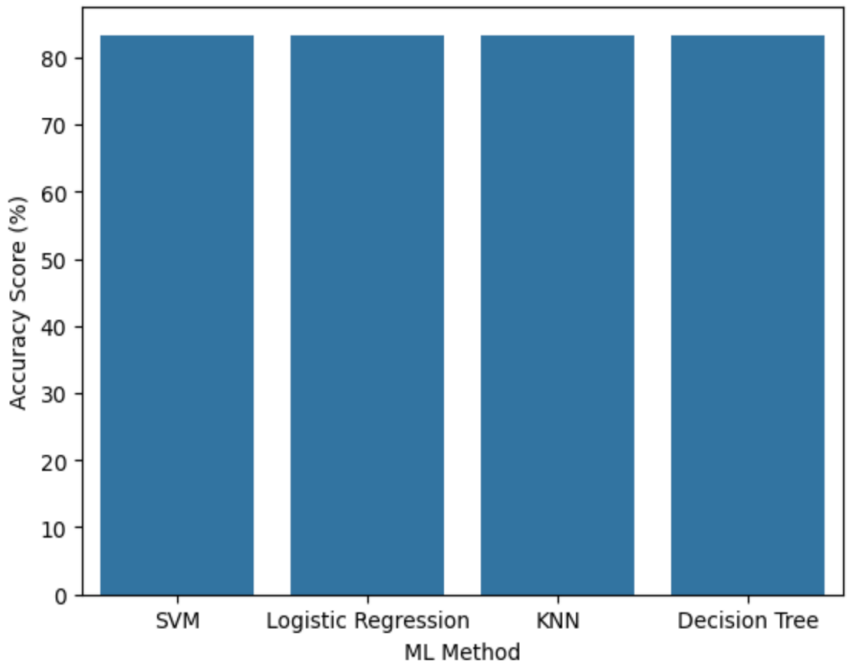
Predictive Analysis (Classification)

Classification Accuracy

From the bar chart, we can see all the models performed at about the same level and have the same accuracy of **83.3%** which is likely due to the small dataset. The Decision Tree model slightly outperformed the rest when looking at **.best_score_**, **.best_score_** is the average of all cv folds for a single combination of the parameters

```
In [44]: sns.barplot(data=ML_df, x='ML Method', y='Accuracy Score (%)')

Out[44]: <AxesSubplot:xlabel='ML Method', ylabel='Accuracy Score (%)'>
```




```
In [23]: print("tuned hpyerparameters :(best parameters) ",tree_cv.best_params_)
         print("accuracy :",tree_cv.best_score_)

tuned hpyerparameters :(best parameters)  {'criterion': 'gini', 'max_depth': 12, 'max_features': 'sqrt', 'min_samp
les_leaf': 1, 'min_samples_split': 2, 'splitter': 'best'}
accuracy : 0.8732142857142857
```

Confusion Matrix

The confusion matrices of all models were identical.

From the confusion matrix, we can see the only problem of the best model (Decision Tree) are **the false positive cases (3)**.

- **Precision** = TP / (TP + FP)

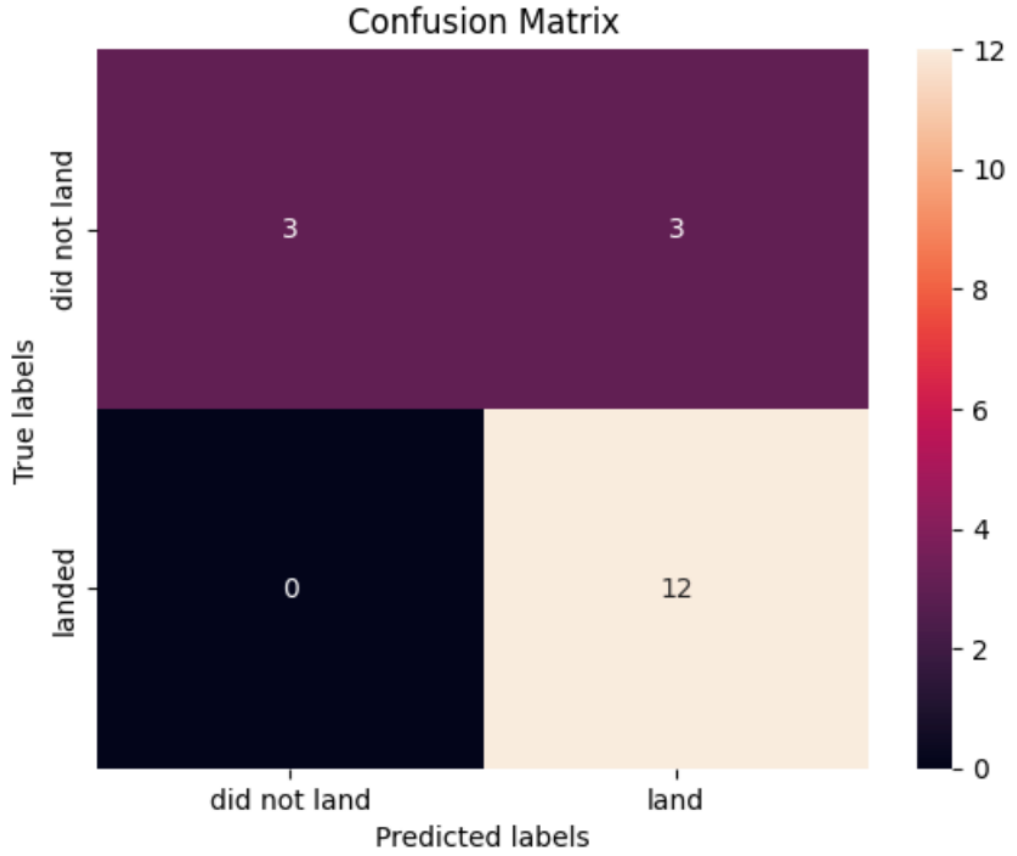
  12 / 15 = .80

- **Recall** = TP / (TP + FN)

  12 / 12 = 1

The model achieved an **f1 score of 0.89**

```
[109]: yhat4 = knn_cv.predict(X_test)
       plot_confusion_matrix(Y_test,yhat4)
```




```
[110]: f1_score(Y_test, yhat4)
```

```
[110]: 0.888888888888889
```

Conclusions

- The models performed similarly on the test set with **decision tree model slightly outperforming**
- All launch sites are located near the coast
- **The launch success increase over time**
- The greater the payload mass, the greater the chances of the launch being successful for all launch sites
- Orbit type **ES-L1, GEO, HEO,** and **SSO** have a 100% success rate
- Booster version **‘FT’** and **‘B4’** have the highest success rate for payload mass range between **2000 and 5500 kg**
- Launch site KSC LC-39A has the most successful launches, while CCAFS LC-40 has the lowest success rate

Thank you!