IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Dylan Greene
05 May 2023

Outline

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

Executive Summary

- This project aims to use Machine Learning to predict if a SpaceX Falcon 9 rocket booster stage will successfully and safely land.
- The project covers a wide range of Data Science skills, including:
  - Data collection and wrangling
  - Exploratory data analysis (EDA) with visualization
  - ML model creation, comparison, and evaluation
- An interactive dashboard was also created to explore the data using different types of plots.
- Ultimately, we were able to predict the target value of successful landing with an accuracy of 83% on the test set of data.

Introduction

- SpaceX is a private launch company disrupting the space industry by reusing rocket booster stages to dramatically reduce the cost of orbital space flight.
- In order to know how to price launches, it is important to know if the booster stage will safely land.
- Machine learning can be used to predict if a booster will land successfully, and thus be available for reuse.
- The data to predict this target variable is available:
  - Through API access of datasets.
  - Through web scrapping.
- The objective is to predict, with a reasonable level of accuracy, whether a booster will successfully land based on the collected data features.

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Data was collected two ways: Requests from the SpaceX API, and by web scrapping launch records from Wikipedia
- Perform data wrangling
  - Data was analyzed with Pandas to determine training labels and separate outcomes
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Multiple classification models were trained with a subset of the data using grid search cross validation and evaluated with test data to score the accuracy of each

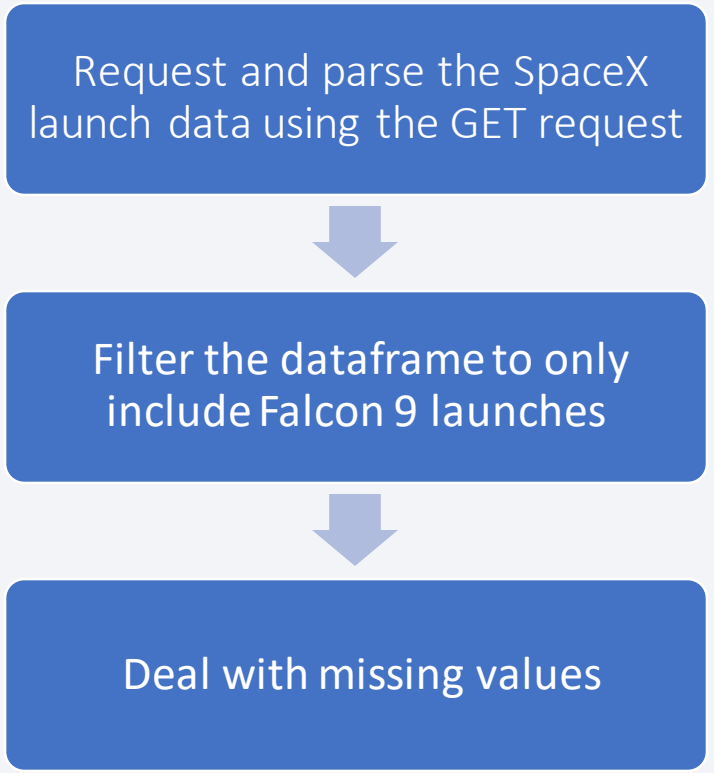
Data Collection

- Data sets were collected from two sources: The SpaceX API and Wikipedia
- The SpaceX API was accessed using REST API calls
- A Wikipedia article on Falcon9 launches was accessed by web scraping techniques and extracting the HTML data using BeautifulSoup
- The specifics of each of these methods are including in the proceeding slides.

Data Collection – SpaceX API

- Data was collected in the following notebook by performing REST calls to the SpaceX API

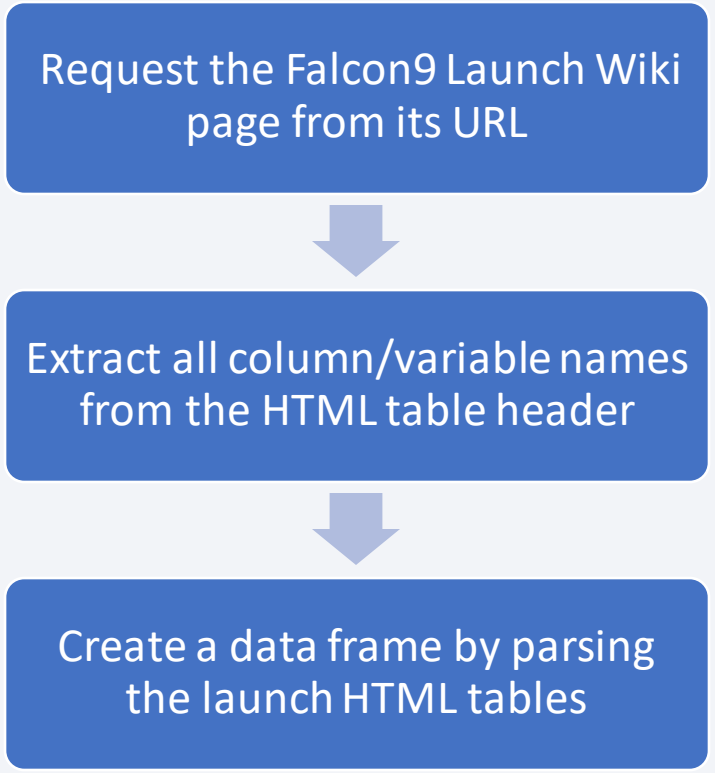
- https://github.com/dylang-test/applied_data_science_capstone/blob/main/data_collection_api.ipynb



Data Collection - Scraping

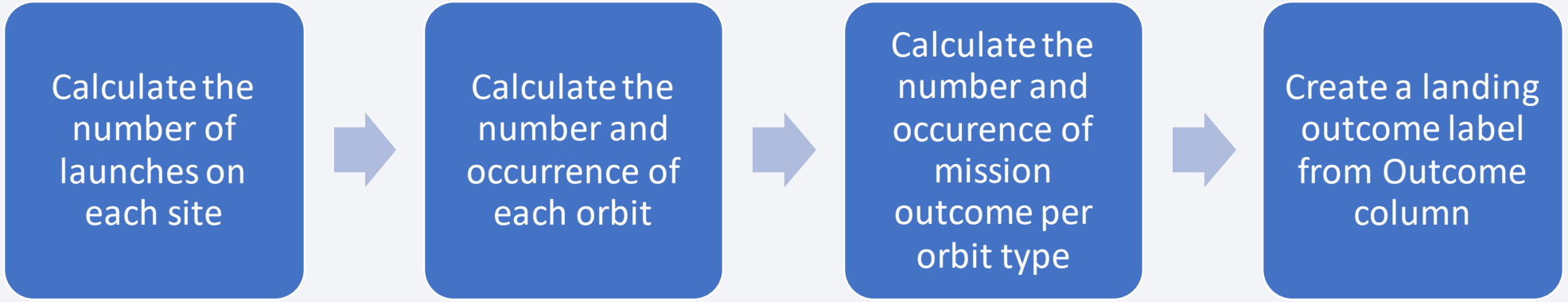
- Falcon9 launch data was collected by scraping a Wikipedia page and using BeautifulSoup in the below notebook

- https://github.com/dylang-test/applied_data_science_capstone/blob/main/webscrapping.ipynb



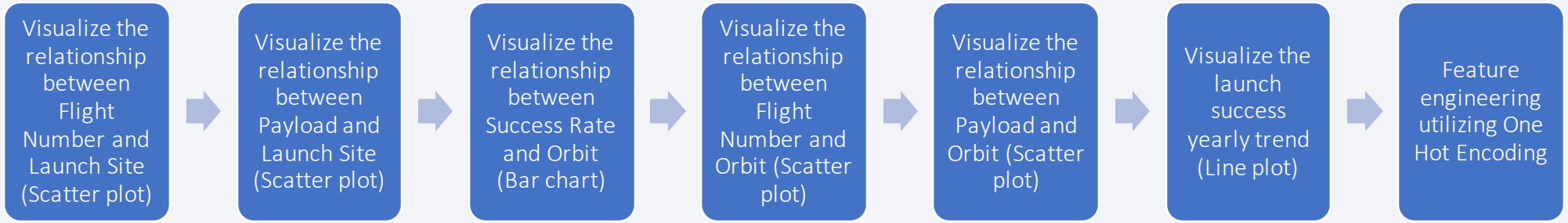
Data Wrangling

- Once the data from the previous sources was collected, it needed to be wrangled for use in training supervised ML models.
- The data explored and, based on that analysis, training labels were determined and calculated in the following notebook, according to flow chart below.
- https://github.com/dylang-test/applied_data_science_capstone/blob/main/data_wrangling.ipynb



EDA with Data Visualization

- Exploratory Data Analysis was done utilizing a variety of seaborn plots in order prepare for feature engineering
- https://github.com/dylang-test/applied_data_science_capstone/blob/main/eda-dataviz.ipynb



EDA with SQL

- SQL was used to understand the SpaceX dataset by loading the dataset into a table in a Db2 database and executing SQL queries to gain insights
  - Displayed the names of the unique launch sites
  - Displayed 5 records where launch sites begin with the string 'CCA'
  - Display the total payload mass carried by boosters launched by NASA (CRS)
  - Display average payload mass carried by booster version F9 v1.1
  - List the date when the first succesful landing outcome in ground pad was acheived.
  - List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000
  - List the total number of successful and failure mission outcomes
  - List the names of the booster_versions which have carried the maximum payload mass. Use a subquery
  - 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.
  - Rank the count of successful landing_outcomes between the date 04-06-2010 and 20-03-2017 in descending order.
- The following notebook contains the SQL queries and results for all of the above.
- https://github.com/dylang-test/applied_data_science_capstone/blob/main/eda-sql.ipynb

Build an Interactive Map with Folium

- An interactive map was created containing the following objects/markers:
  - All launch sites
  - Success/failed launches for each site (in clusters)
  - The distance of launch site to nearby proximities (such as the ocean and nearby city)
- It is possible that the location and proximities of a launch site may have an effect on the success rate of launches and landings. The purpose of creating these maps with these objects and markers was to explore this and hopefully discover some of the factors in building optimal launch sites by analyzing the existing sites.
- https://github.com/dylang-test/applied_data_science_capstone/blob/main/launch_site_location.ipynb

Build a Dashboard with Plotly Dash

- A dashboard was created with a drop down selector for launch site locations and a slider to select payload ranges of launches. These criteria were used to generate a pie chart of launch success and a scatter chart showing correlation between payload and success.
- The purpose of this dashboard and these charts was to visually be able to explore and analyze the data interactively and to be able to identify trends.
- https://github.com/dylang-test/applied_data_science_capstone/blob/main/launch_site_location.ipynb

Predictive Analysis (Classification)

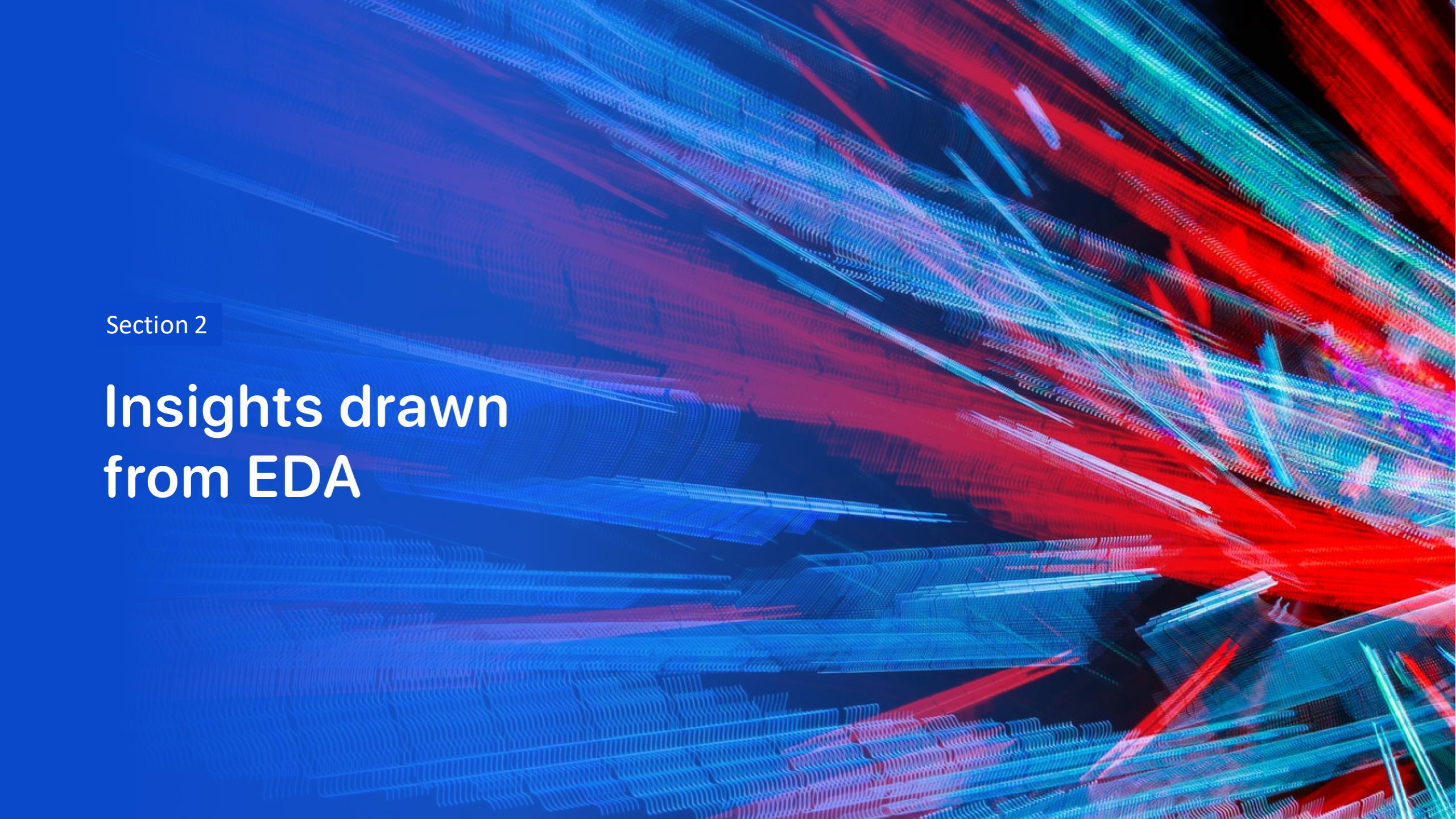
- In order to predict if a booster landing would be successful, several steps were required (these are outlined in the flowchart below).
- Multiple models were trained and evaluated: SVM, Classification Trees, KNN, and Logistic Regression.
- https://github.com/dylang-test/applied_data_science_capstone/blob/main/machine_learning_prediction.ipynb

Results

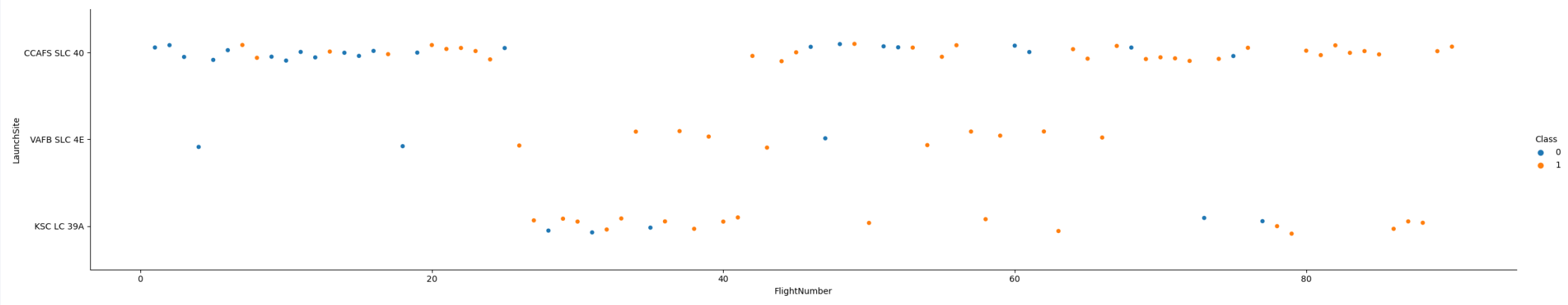
- Exploratory data analysis results
- Interactive analytics demo in screenshots
- The predictive analysis showed that all models performed the same on the test set of data.
  - Specifically, the Logistic Regression, SVM, Decision Tree, and KNN classifiers all had 83.33% accuracy on the test subset with the same confusion matrix.

Section 2

Insights drawn from EDA

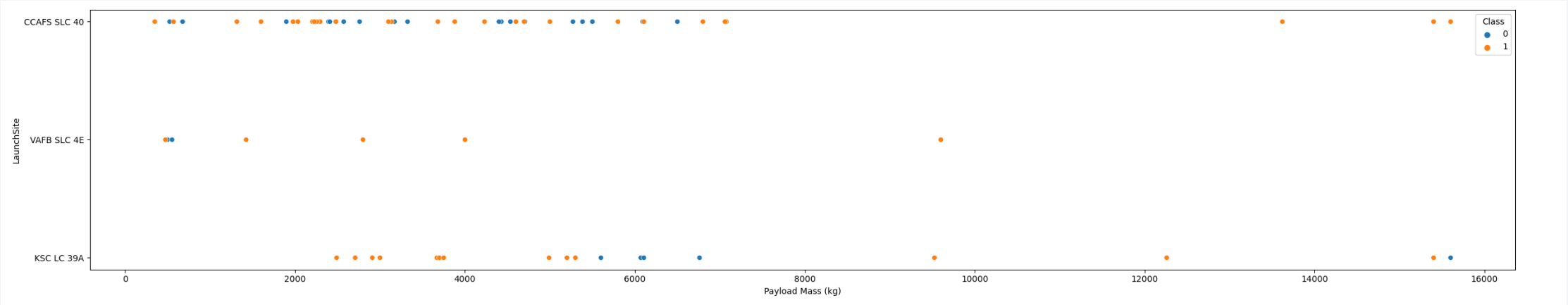


Flight Number vs. Launch Site



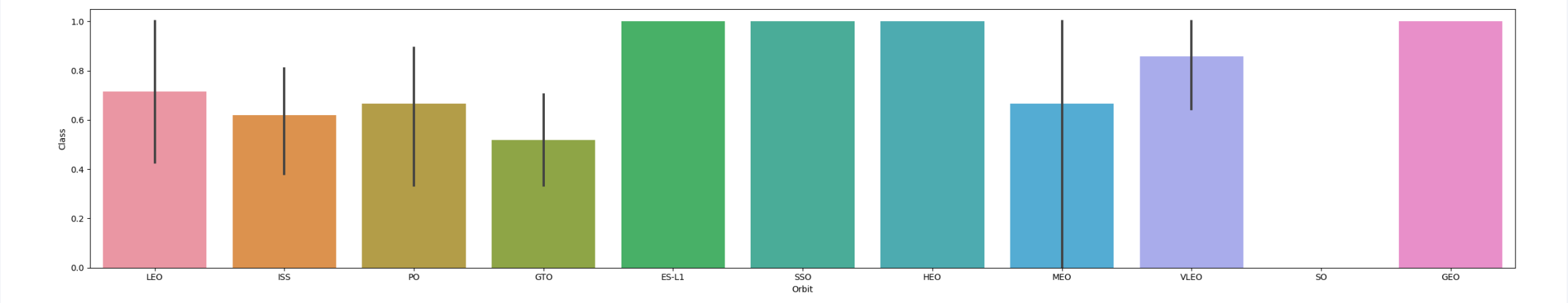
- This scatter plot shows Flight Number vs. Launch Site.
- This shows that for some launch sites (in particular CCAFS SLC 40) that success is more likely with higher launch numbers indicating an improvement at that site.

Payload vs. Launch Site



- This is a scatter plot of Payload vs. Launch Site
- This plot shows that at the VAFB-SLC launch site there are no rockets launched for heavy payload mass (greater than 10000).

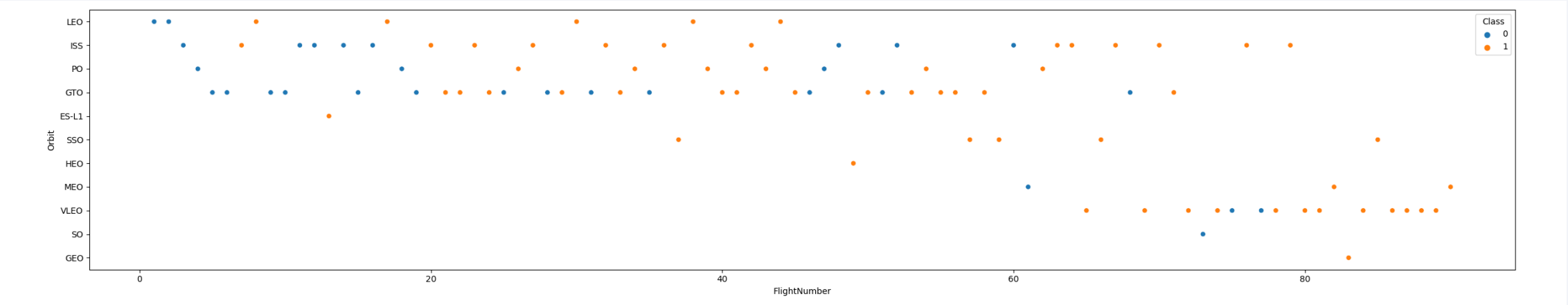
Success Rate vs. Orbit Type



- This plot is a bar chart for the success rate of each orbit type

- This chart demonstrates which orbit types have high success rates, such as VLEO

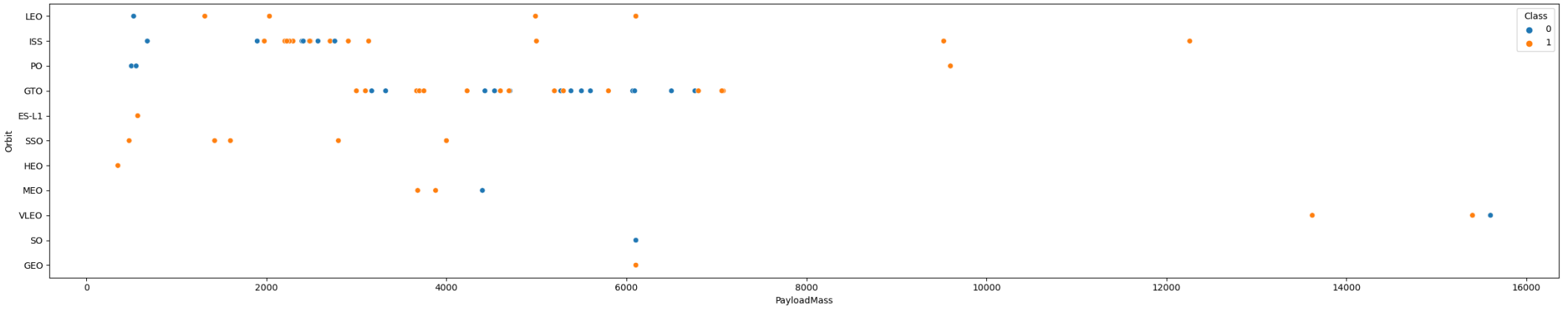
Flight Number vs. Orbit Type



- This is a scatter point of Flight number vs. Orbit type

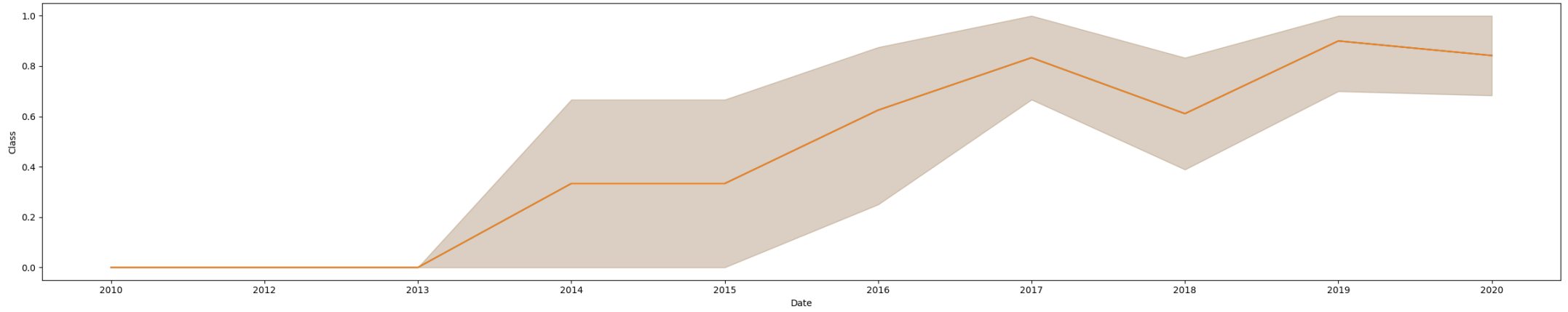
- In the LEO orbit the Success appears related to the number of flights; on the other hand, there seems to be no relationship between flight number when in GTO orbit.

Payload vs. Orbit Type



- This shows a scatter point of payload vs. orbit type
- With heavy payloads the successful landing or positive landing rate are more for Polar, LEO and ISS.
- However for GTO we cannot distinguish this well as both positive landing rate and negative landing (unsuccessful mission) are both there here.

Launch Success Yearly Trend



- This line chart shows yearly average success rate

- The clear trend is that success rates have kept increasing from 2013 to 2020.

All Launch Site Names

- The query shows each unique launch site in the data set

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

In [19]:
```
%sql select distinct Launch_Site from SPACEXTBL
```

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

Out[19]:

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

Launch Site Names Begin with 'CCA'

- This query filters the result set to only include launch sites beginning with 'CCA'

In [20]:
```
%sql select * from SPACEXTBL where Launch_Site like 'CCA%' limit 5
```

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

Out[20]:

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Custo
04-06-2010	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	Sp
08-12-2010	15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight C1, two CubeSats, barrel of Brouere cheese	0	LEO (ISS)	N (C
22-05-2012	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	N (C
08-10-2012	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	N (
01-03-2013	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	N (

Total Payload Mass

- This query calculates the total payload carried by boosters from NASA
- The total is 107010 kg

Task 3

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

```
In [12]: %sql select sum(PAYLOAD_MASS__KG_) from SPACEXTBL where Customer like '%NASA%'
 * sqlite:///my_data1.db
Done.
```

Out[12]:

sum(PAYLOAD_MASS__KG_)
107010

Average Payload Mass by F9 v1.1

- This query calculates the average payload mass carried by booster version F9 v1.1
- The average is 2534.67 kg

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

In [33]:
```
%sql select avg(PAYLOAD_MASS__KG_) from SPACEXTBL where Booster_Version like '%F9 v1.1%'
```

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

Out[33]:

avg(PAYLOAD_MASS__KG_)
2534.6666666666665

First Successful Ground Landing Date

- This query finds the date of the first successful landing outcome on ground pad
- The first time a booster successfully landed on a ground pad was December 22, 2015

In [40]:
```
%%sql
select min(substr(Date,7,4) || substr(Date,4,2) || substr(Date,1,2)) as DATE from SPACEXTBL
where "Landing _Outcome"='Success (ground pad)'
```

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

Out[40]:

DATE
20151222

Successful Drone Ship Landing with Payload between 4000 and 6000

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

In [42]:

```
%%sql
select distinct Booster_Version from SPACEXTBL
where "Landing _Outcome" = "Success (drone ship)"
and PAYLOAD_MASS__KG_ between 4000 and 6000
```

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

Out[42]:

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

Total Number of Successful and Failure Mission Outcomes

- This query calculates the total number of successful and failure mission outcomes

In [32]:
```
%sql select Mission_Outcome, count(*) as N from SPACEXTBL group by Mission_Outcome
```

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

Out[32]:

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

Boosters Carried Maximum Payload

- This query lists the names of the booster which have carried the maximum payload mass

In [34]:
```
%%sql
select Booster_Version from SPACEXTBL
where PAYLOAD_MASS__KG_ = (select max(PAYLOAD_MASS__KG_) from SPACEXTBL)
```

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

Out[34]:

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

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

In [39]:
```
%%sql
select substr(Date, 4, 2) as MONTH, "Landing _Outcome", Booster_Version, Launch_Site from SPACEXTBL
where substr(Date, 7, 4)='2015' and "Landing _Outcome"='Failure (drone ship)'
```

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

Out[39]:

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

- This query ranks 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

In [44]:

```
%%sql
select "Landing _Outcome", count(*) as N from SPACEXTBL
where Date between '04-06-2010' and '20-03-2017'
group by "Landing _Outcome"
order by N desc
```

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

Out[44]:

Landing _Outcome	N
Success	20
No attempt	10
Success (drone ship)	8
Success (ground pad)	6
Failure (drone ship)	4
Failure	3
Controlled (ocean)	3
Failure (parachute)	2
No attempt	1


Section 3
Launch Sites
Proximities Analysis

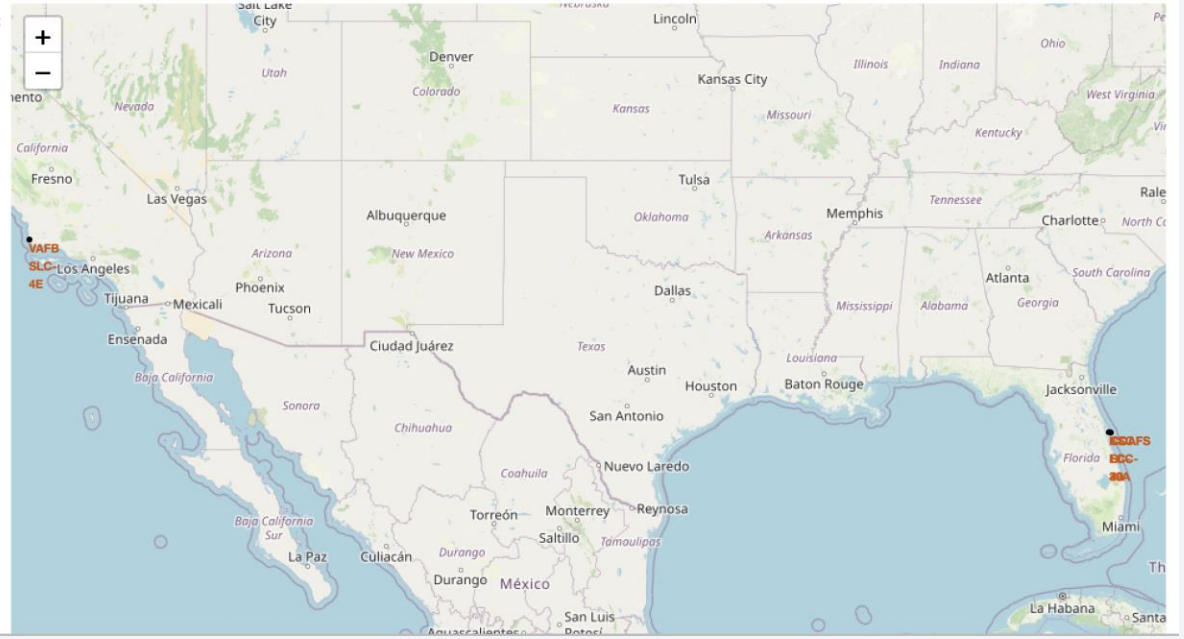
Interactive Map of SpaceX Launch Sites

- The code on the side creates an interactive map that can be used to view the SpaceX launch sites as shown.

- As seen in the map, this reveals that all of the launch sites are close to the equator and near the oceans.

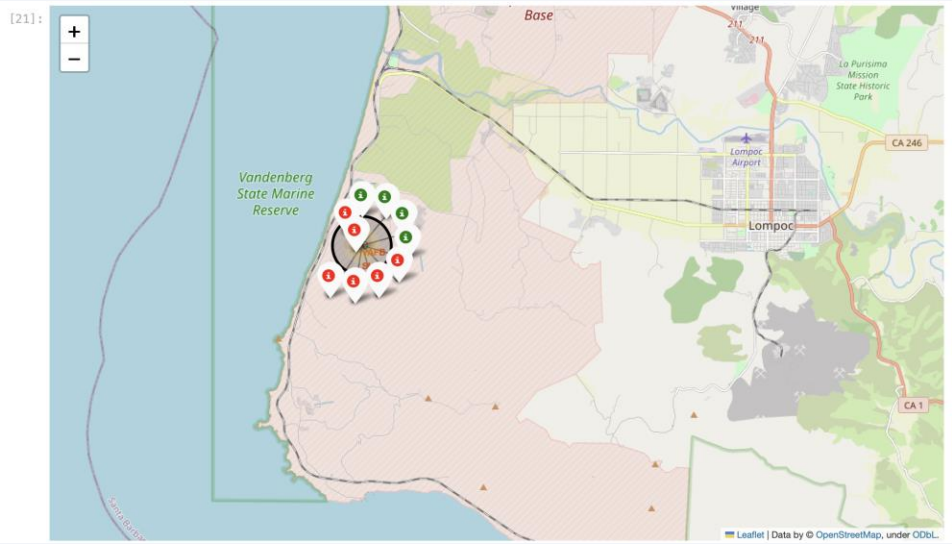
```
[14]: # Initial the map
site_map = folium.Map(location=nasa_coordinate, zoom_start=5)
# For each launch site, add a Circle object based on its coordinate (Lat, Long) values. In addition, add Launch site name as
for _, launch_site in launch_sites_df.iterrows():
    circle = folium.Circle([launch_site['Lat'], launch_site['Long']], radius=1000, color='#000000', fill=True).add_child(foli
    marker = folium.map.Marker([launch_site['Lat'], launch_site['Long']], icon=DivIcon(icon_size=(20,20), icon_anchor=(0,0),
    site_map.add_child(circle)
    site_map.add_child(marker)
site_map
```

[14]:



Adding Clustered Launch Data to the Map

- This updated map now is able to display he clustered launches including number of launches per cluster, and uses red and green markers to display landing failure or success respecitvely

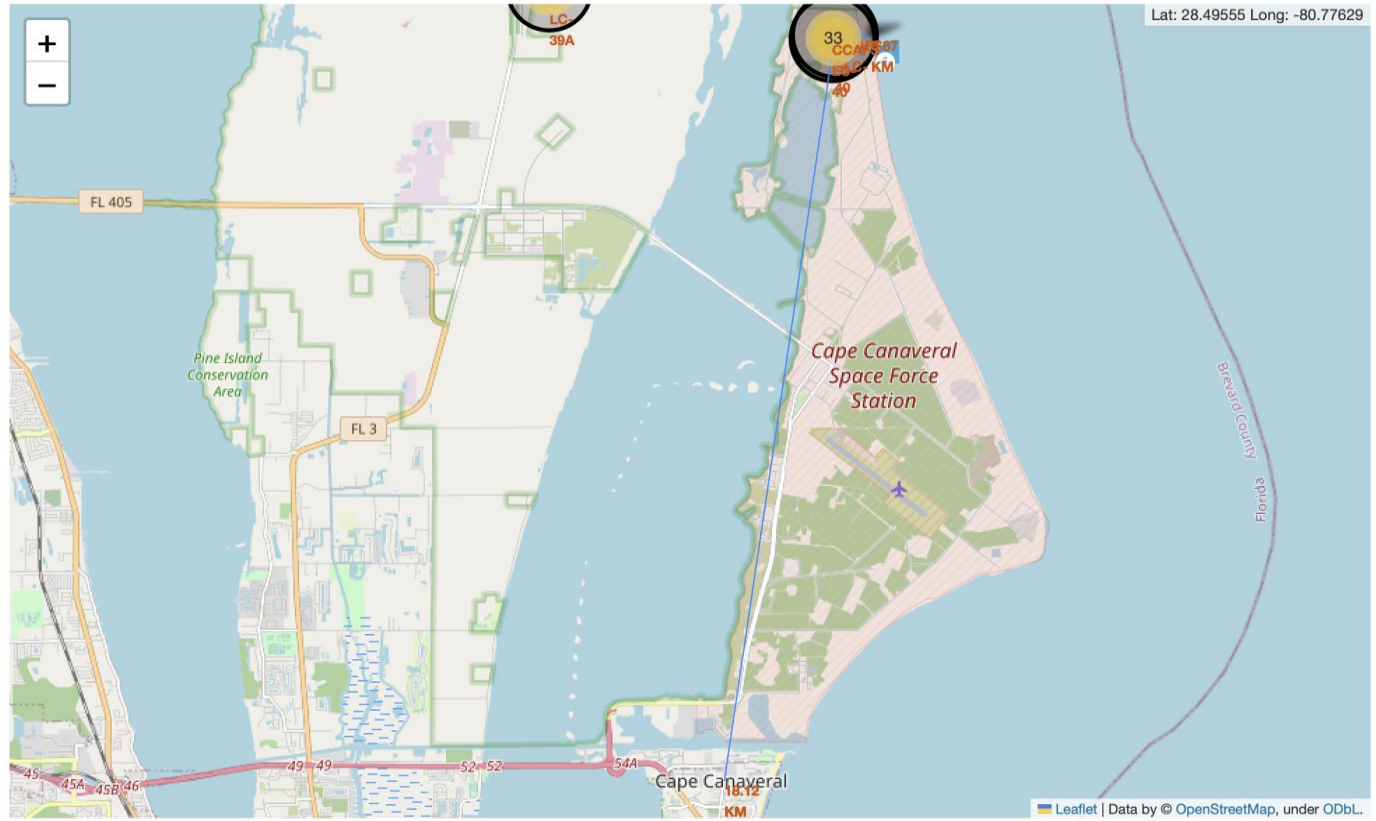


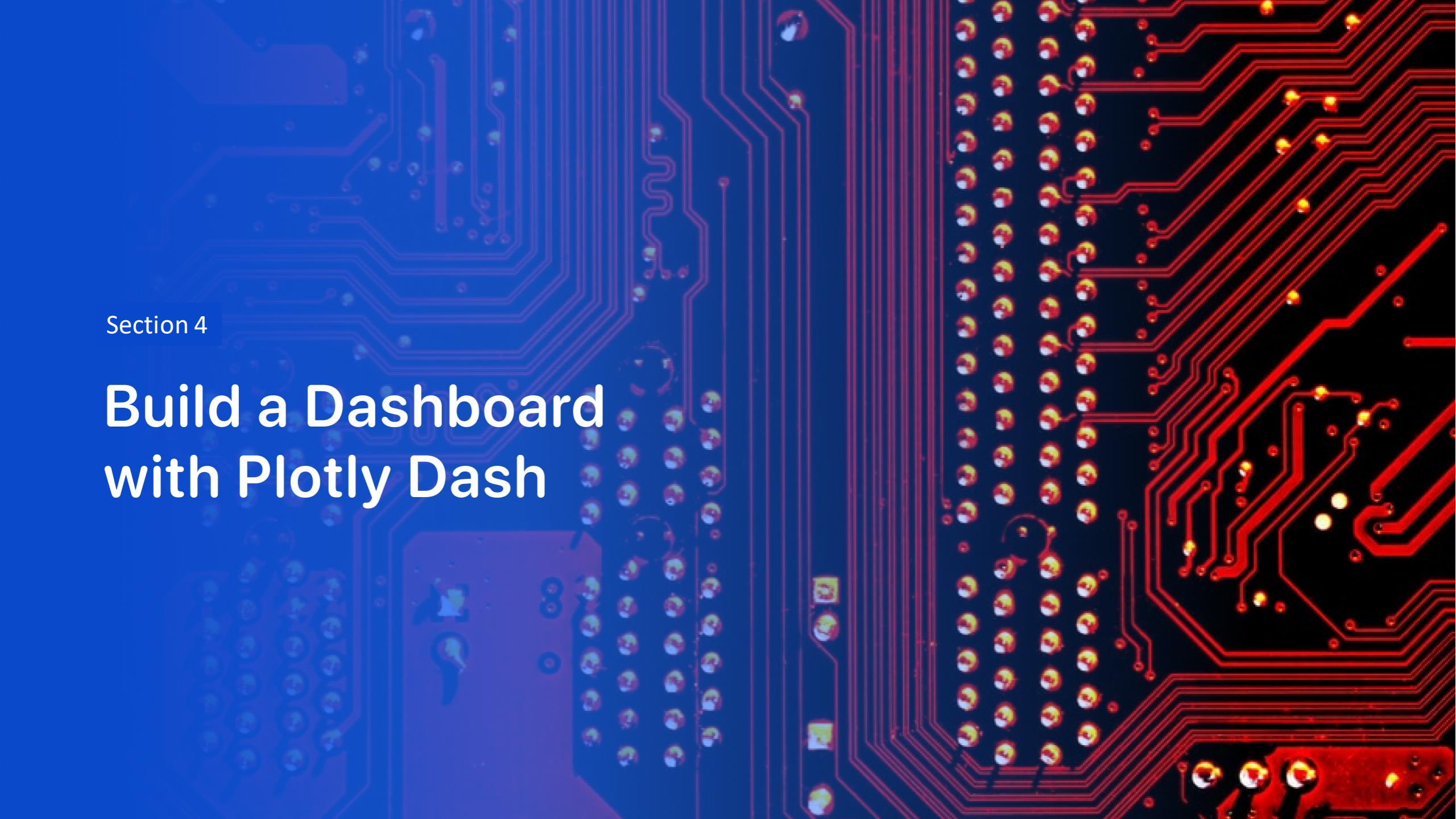

```
[21]: # Add marker_cluster to current site_map
site_map.add_child(marker_cluster)

# for each row in spacex_df data frame
# create a Marker object with its coordinate
# and customize the Marker's icon property to indicate if this launch was successed or failed,
# e.g., icon=folium.Icon(color='white', icon_color=row['marker_color']
for index, record in spacex_df.iterrows():
    # TODO: Create and add a Marker cluster to the site map
    # marker = folium.Marker(...)
    marker = folium.Marker([record['Lat'], record['Long']], icon=folium.Icon(color='white', icon_color=record['marker_color']
    marker_cluster.add_child(marker)

site_map
```

Adding Proximity to Nearby Features to the Map

- By calculating distance to both the ocean and a nearby city, we can add that insight to the interactive map.
- We can see that the launch site is very close to the ocean, at only 0.87 km away.
- However, it is far from the nearby city of Cape Canaveral (18.12 km).

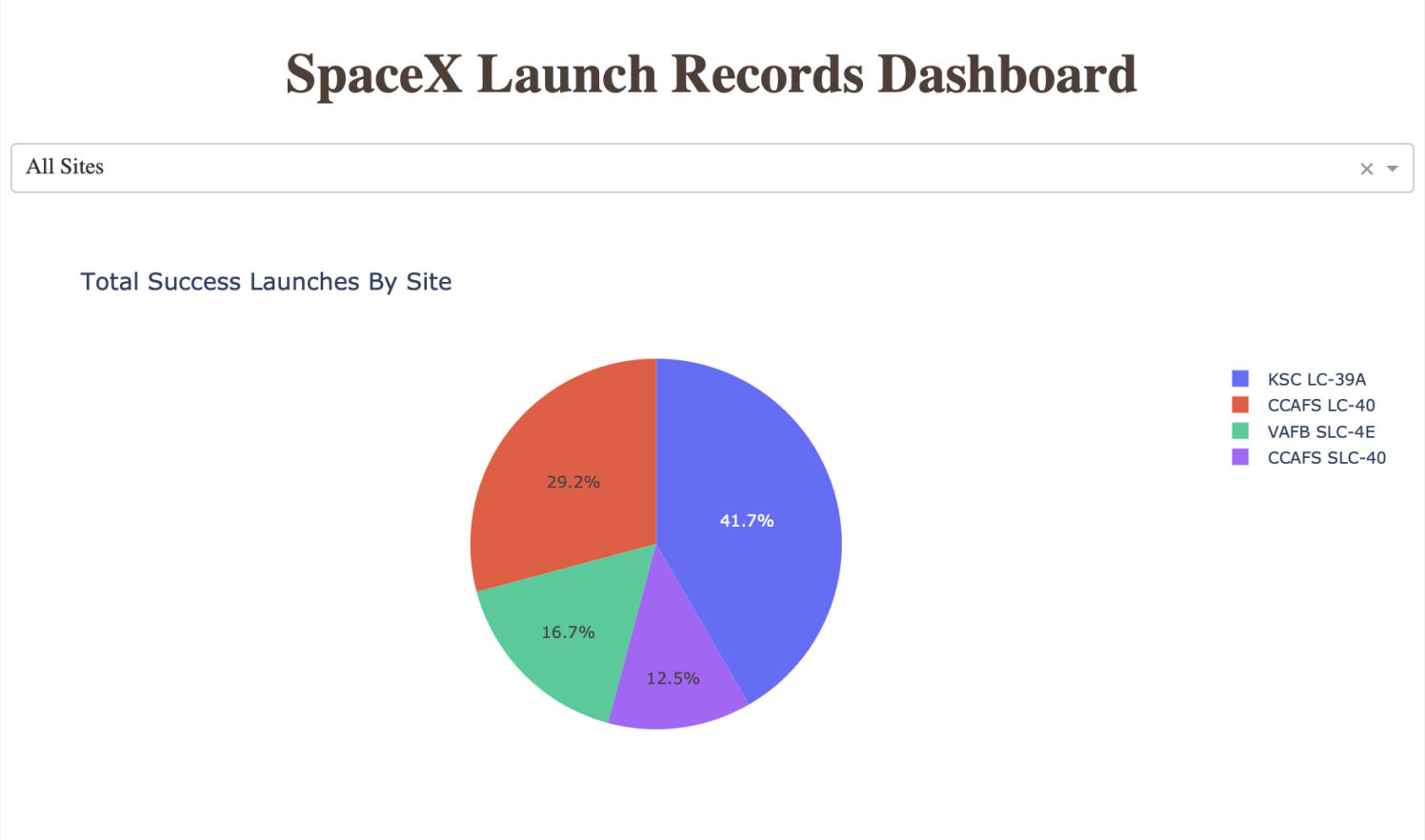
```
[37]: distance_cape_canaveral = calculate_distance(28.56319,-80.57681,28.40205,-80.60411)
distance_marker = folium.Marker([28.40205,-80.60411], icon=DivIcon(icon_size=(20,20), icon_anchor=(0,0), html='<div style="fo
site_map.add_child(distance_marker)
lines = folium.PolyLine(locations=[[28.56319,-80.57681],[28.40205,-80.60411]], weight=1)
site_map.add_child(lines)
```




Section 4
Build a Dashboard with Plotly Dash

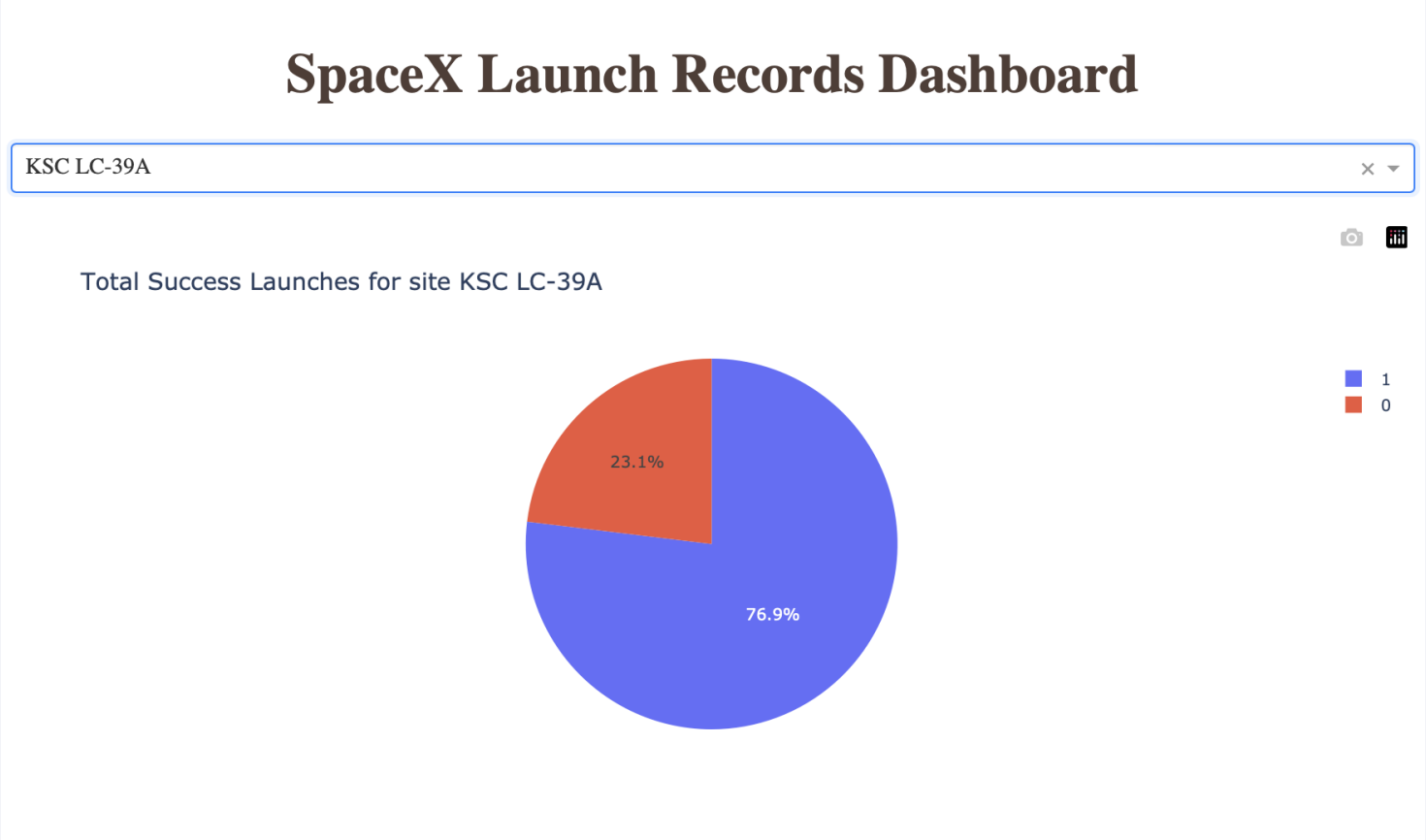
Dashboard: Launch Success For All Sites

- The screenshot shows launch success count for all sites, in a pie chart, on an interactive dashboard
- The pie chart clearly shows that KSC LC-39A has the most launch success

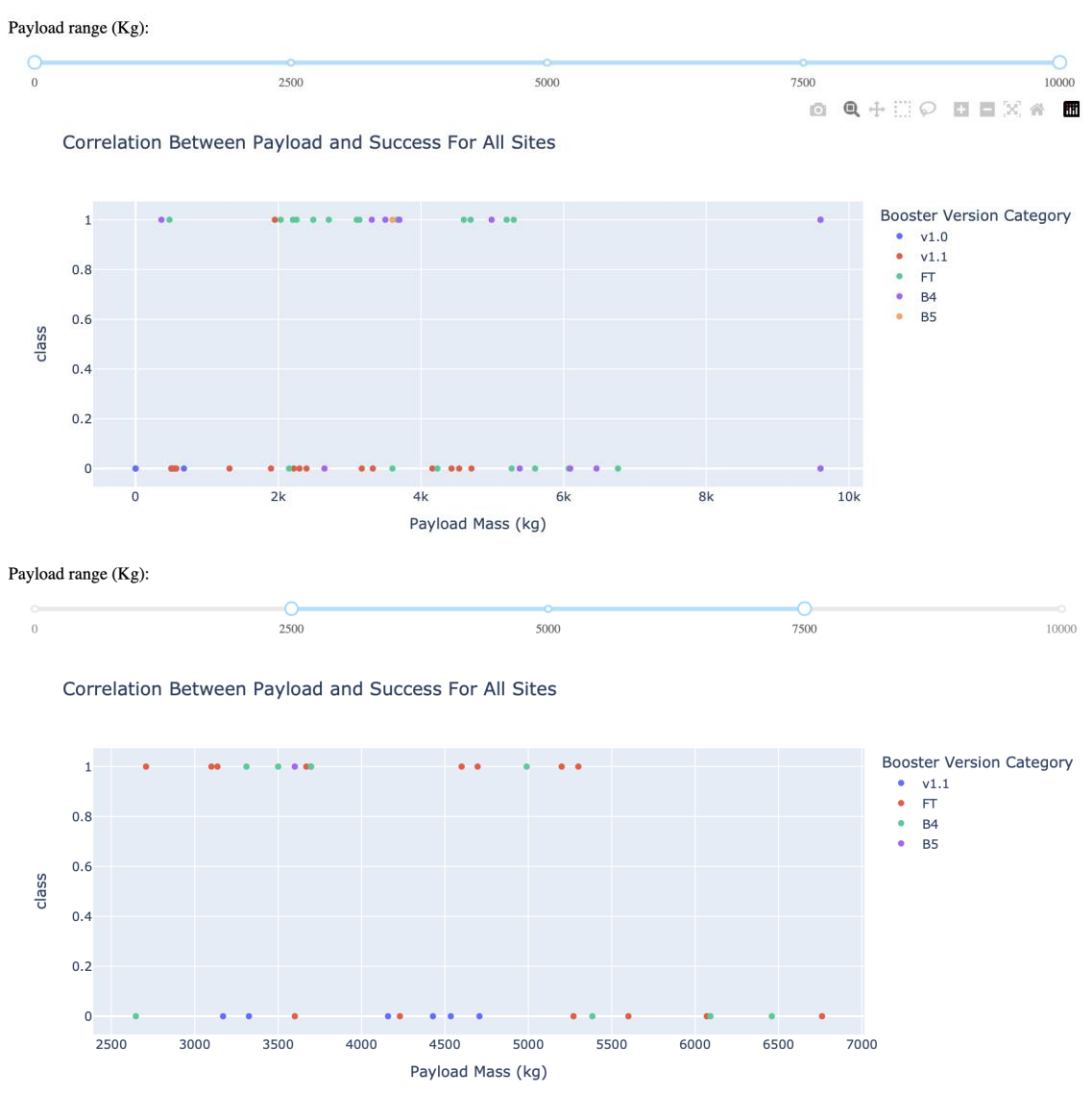


Dashboard: Launch Success Ratio

- The screenshot shows the pie chart for the launch site with highest launch success ratio: KSC LC-39A.
- It was easy to simply select each launch site in the dashboard and quickly, visually determine which site had the highest ratio by looking at the pie chart.



Dashboard: Correlation Between Payload and Success



- These screenshots show Payload vs. Launch Outcome scatter plot for all sites, with different payload selected in the range slider
- By using the slider, it can be determined which ranges of Payload have the highest amount of landing success.
- Specifically, we can see that between 2000 and 4000 kg there are a high amount of successful landings.

Section 5

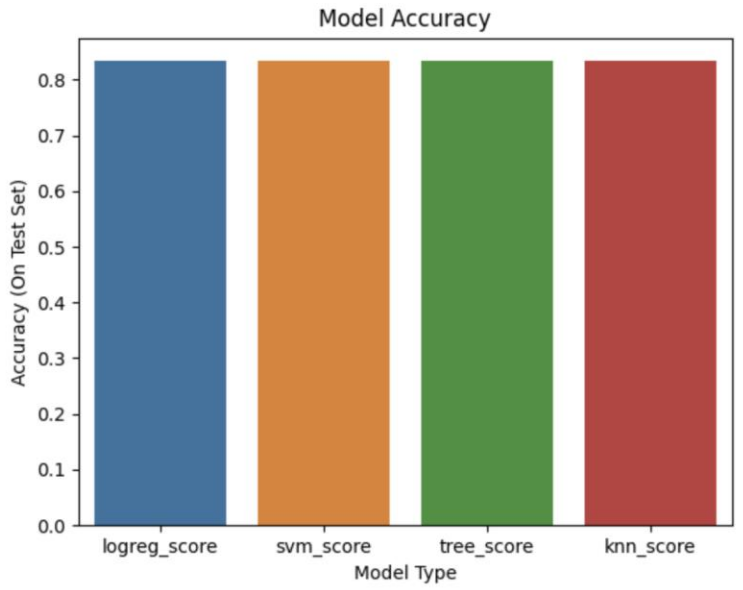
Predictive Analysis (Classification)

Classification Accuracy

- Shown right is a visualization of the built model accuracy for all built classification models, in a bar chart

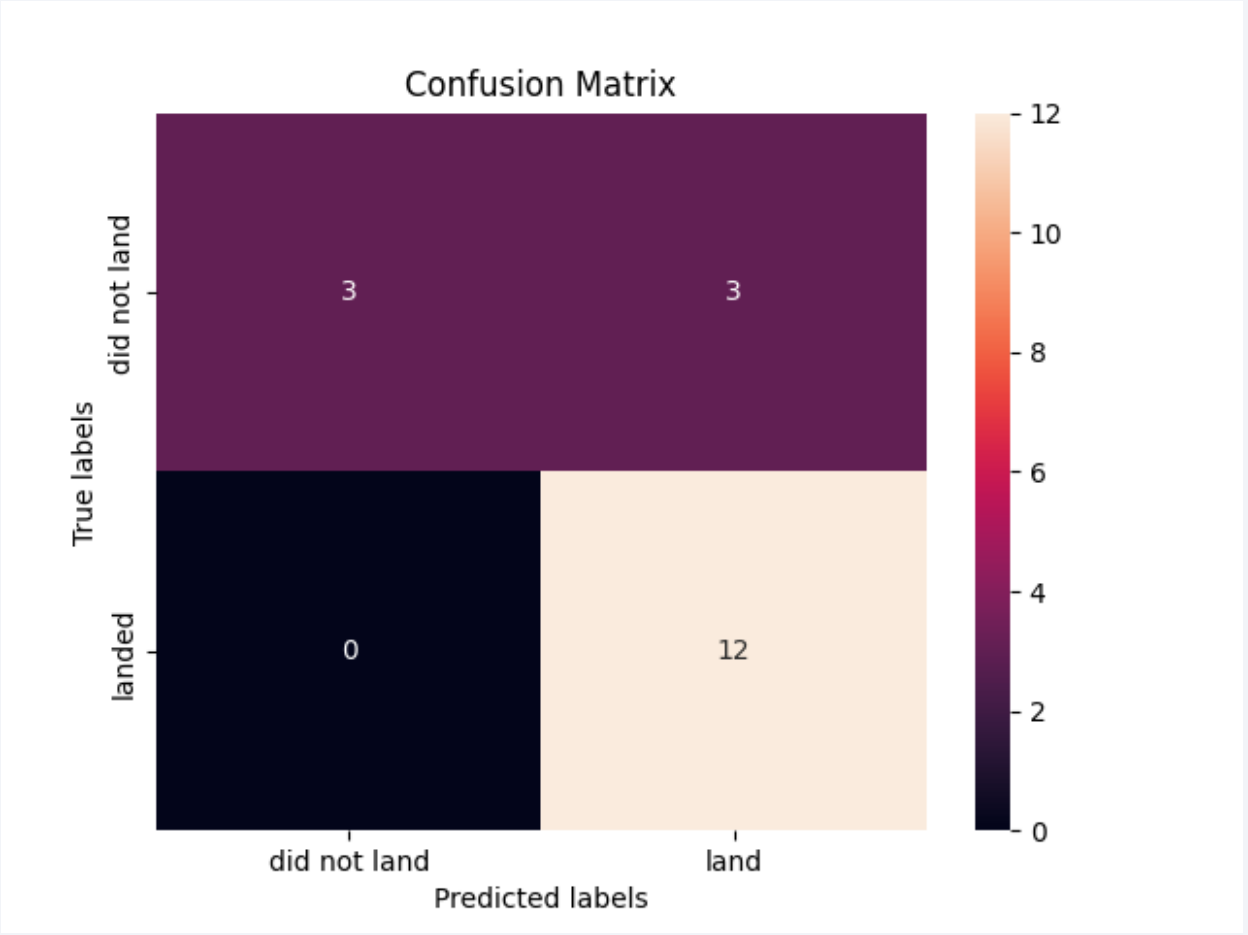
- As seen in the bar chart, all of the model types had the same accuracy (83%) when evaluating the test data set

[37]:
```
scores = {
    'logreg_score': logreg_cv.score(X_test, Y_test),
    'svm_score': svm_cv.score(X_test, Y_test),
    'tree_score': tree_cv.score(X_test, Y_test),
    'knn_score': knn_cv.score(X_test, Y_test)
}
keys = list(scores.keys())
vals = [scores[k] for k in keys]
import seaborn as sns

sns.barplot(x=keys, y=vals).set(xlabel = 'Model Type', ylabel='Accuracy (On Test Set)', title='Model Accuracy')
plt.show()
```



Confusion Matrix

- Shown right is the confusion matrix for the machine learning models trained for the project on the test data set.
- From this we can see the True Positive (3), False Positives (3), False Negatives (0), and True Negatives (12)



Conclusions

- It was possible to collect data from the internet via both REST API and Web Scraping to build a dataset for this Machine Learning task of predicting booster landing success.
- Using SQL and plots to visualize the data provided insights into which features would be useful in predicting the landing success target.
- Interactive maps and dashboards are very useful in easily exploring a dataset to find insights that would otherwise be difficult to see.
- There are multiple Machine Learning models/algorithms that can effectively predict the target.
- We were successful in achieving a high accuracy (83%) in predicting whether a SpaceX booster will land safely.
- The successful insights delivered from this project and accurate predictions can now be used by a business such as a SpaceX competitor.

Thank you!