IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Mohammad Zayd
20-01-2022

Outline

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

Executive Summary

The goal of the project is to use the data to predict whether the SpaceX's Falcon 9 first stage will land successfully. The data for this project was sourced form SpaceX REST API and Wikipedia using web scraping. Web wrangling was performed to make the data useful for Machine Learning and further analysis. SQL query was used to bring into light various insights from the data. Exploratory Data Analysis, feature scaling is done with visualization using scatter and bar plots to make the data insightful and determine the best predictors. Further, Machine Learning models are created to predict the future outcomes.

It was found that the predicted outcome was dependent on the Flight Number, Launch Site, Payload and Orbit.

Introduction

The evolution of technologies has changed the lives of people a lot, and with the current technologies, we are even building commercial space flights. This could one day make humans multi-planetary species. There are major companies in this space race, namely Blue Origin, Virgin Galactic, and SpaceX. The current leader in this race seems to be SpaceX, and the reason behind this is the reusability of their Stage 1 of rocket. This reduces the cost of launch from an estimated $165 million to around $62 million per launch.

But a problem arises is how to predict the launch price of Falcon 9 rocket?

The predicted valuation using the data can be used against companies that want to compete with SpaceX. Our goal is to use this data to predict whether SpaceX will attempt to land a rocket or not. Predicting that whether Stage 1 of the rocket will land successfully or not can play a crucial role in predicting the launch price. Since this stage of the rocket can be reused again with different payloads, hence it can reduce the cost by more than half of the original.

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

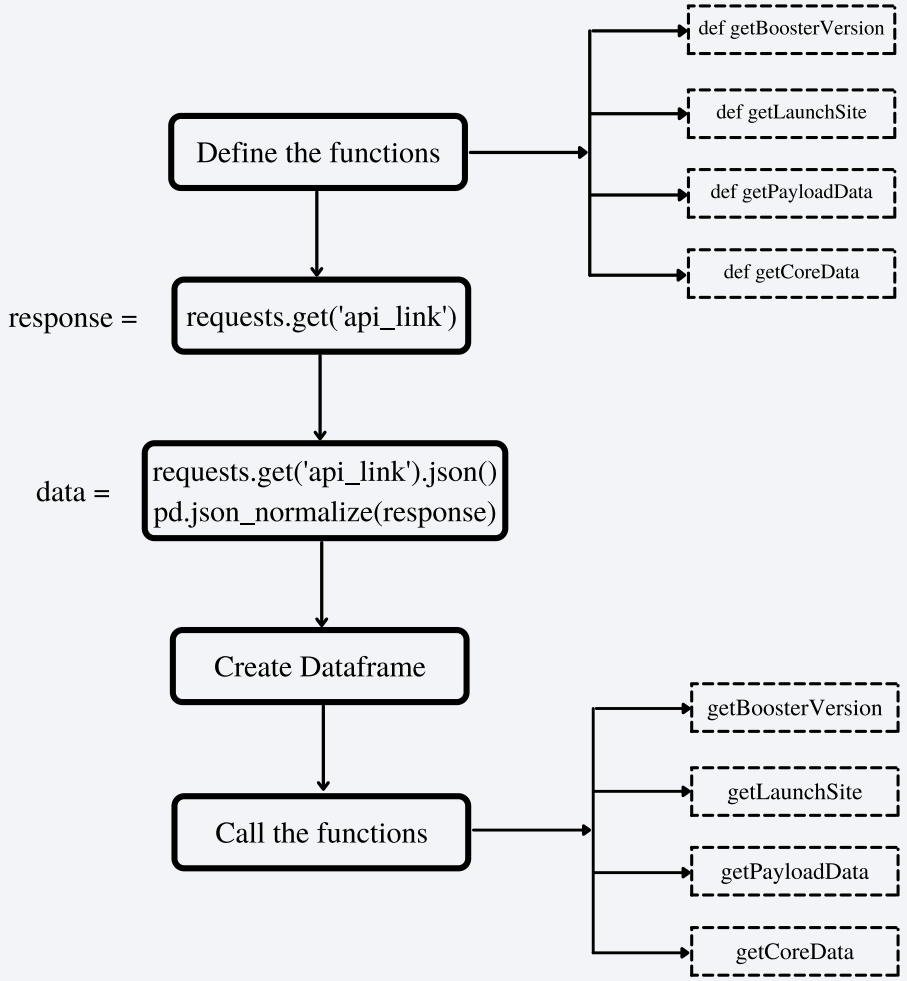
Data Collection

- The data was first collected from the SpaceX REST API. This API gives us data about launches, including information about the rocket used, payload delivered, launch specifications, landing specifications, and landing outcome.
- Used this data to predict if the rocket will land safely or not.
- Also obtained Falcon 9's launch data by Web Scraping the Wikipedia link using BeautifulSoup.

Data Collection – SpaceX API

- SpaceX launch data that is gathered from an API, specifically the SpaceX REST API. This API will give us data about launches, including information about the rocket used, payload delivered, launch specifications, landing specifications, and landing outcome.
- We are working with the endpoint api.spacexdata.com/v4/launches/past. We use this URL to target a specific endpoint of the API to get past launch data.

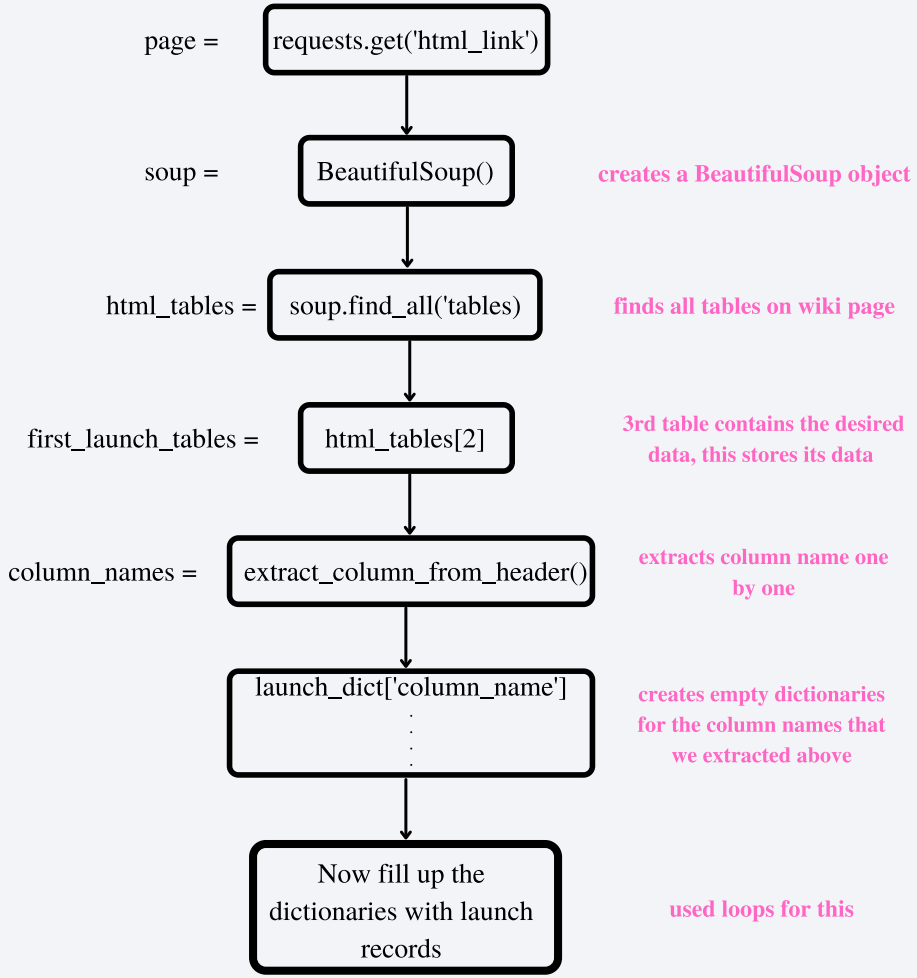
GitHub URL: Link



Data Collection - Scraping

- Some of the essential data was collected from Wikipedia using web scrapping with the help of beautiful soup framework.

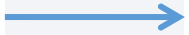
GitHub URL: [Link]



Data Wrangling

- In the data set, there are several different cases where the booster did not land successfully. Sometimes a landing was attempted but failed due to an accident.
- For example, True Ocean means the mission outcome was successfully landed to a specific region of the ocean while False Ocean means the mission outcome was unsuccessfully landed to a specific region of the ocean. True RTLS means the mission outcome was successfully landed to a ground pad False RTLS means the mission outcome was unsuccessfully landed to a ground pad. True ASDS means the mission outcome was successfully landed on a drone ship False ASDS means the mission outcome was unsuccessfully landed on a drone ship.
- Then convert these outcomes into training labels with 1 meaning that the booster successfully landed and 0 meaning that it was unsuccessful.

	Outcome
0	None None
1	None None
2	None None
3	False Ocean
4	None None
5	None None
6	True Ocean
7	True Ocean
8	None None
9	None None
10	False Ocean
11	False ASDS
12	True Ocean
13	False ASDS
14	None None



	Class
0	0
1	0
2	0
3	0
4	0
5	0
6	1
7	1
8	0
9	0
10	0
11	0
12	1
13	0
14	0

GitHub URL: Link

EDA with Data Visualization

Scatter plots and Bar Graphs for different inputs in the axes were realized. Following visuals were made:

- FlightNumber vs. PayloadMass
- PayloadMass vs. LaunchSite
- Orbit vs. Success Rate
- FlightNumber vs. Orbit
- Orbit vs. PayloadMass
- Year vs. Success Rate

GitHub URL: Link

EDA with SQL

Performed SQL queries to gather information about the dataset. Following questions were answered using SQL queries to get the answers in the dataset.

- Displaying the names of the unique launch sites in the space mission.
- Displaying 5 records where launch sites begin with the string 'KSC'.
- Displaying the total payload mass carried by boosters launched by NASA (CRS) Displaying average payload mass carried by booster version F9 v1.1.
- Listing the date where the successful landing outcome in drone ship was achieved.
- Listing the names of the boosters which have success in ground pad and have payload mass greater than 4000 but less than 6000.
- Listing the total number of successful and failure mission outcomes.
- Listing the names of the booster_versions which have carried the maximum payload mass.
- Listing the records which will display the month names, successful landing_outcomes in ground pad , booster versions, launch_site for the months in year 2017.
- Ranking the count of successful.

GitHub URL: Link

Build an Interactive Map with Folium

- An interactive map is created for visualizing the various factors, markers, and highlighted circles with popups are added for different launch sites to easily spot them on the map.
- If a launch was successful (class=1), then we use a green marker and if a launch was failed, we use a red marker (class=0).
- Also calculated the distance between two points on the map based on their Lat and Long values. For this we mark down a point on the landmark using MousePosition and calculate the distance by putting the coordinates of the landmark in the calculate_distance function. Then draw a PolyLine between a launch site to the selected landmark.

GitHub URL: Link

Build a Dashboard with Plotly Dash

- An interactive web application is created using dash, with a drop-down menu to select the launch site, and range-slider to choose the range of payload.
- An interactive pie chart shows success rate of all launch sites by default, and a scatter plot shows launch outcomes of all the launch sites according to their payloads in the default range (0-10000).
- The dropdown menu allows the user to choose the launch site, this alters the figure of pie chart and scatter plot to show outcomes of the chosen launch site. Through the range-slider the user can select the range of payload on the x-axis of scatter plot.
- These interactions allows the user to visualize the data in depth according to understand the data thoroughly.
- GitHub URL: Link

Predictive Analysis (Classification)

- Summarize how you built, evaluated, improved, and found the best performing classification model
- You need present your model development process using key phrases and flowchart
- Add the GitHub URL of your completed predictive analysis lab, as an external reference and peer-review purpose

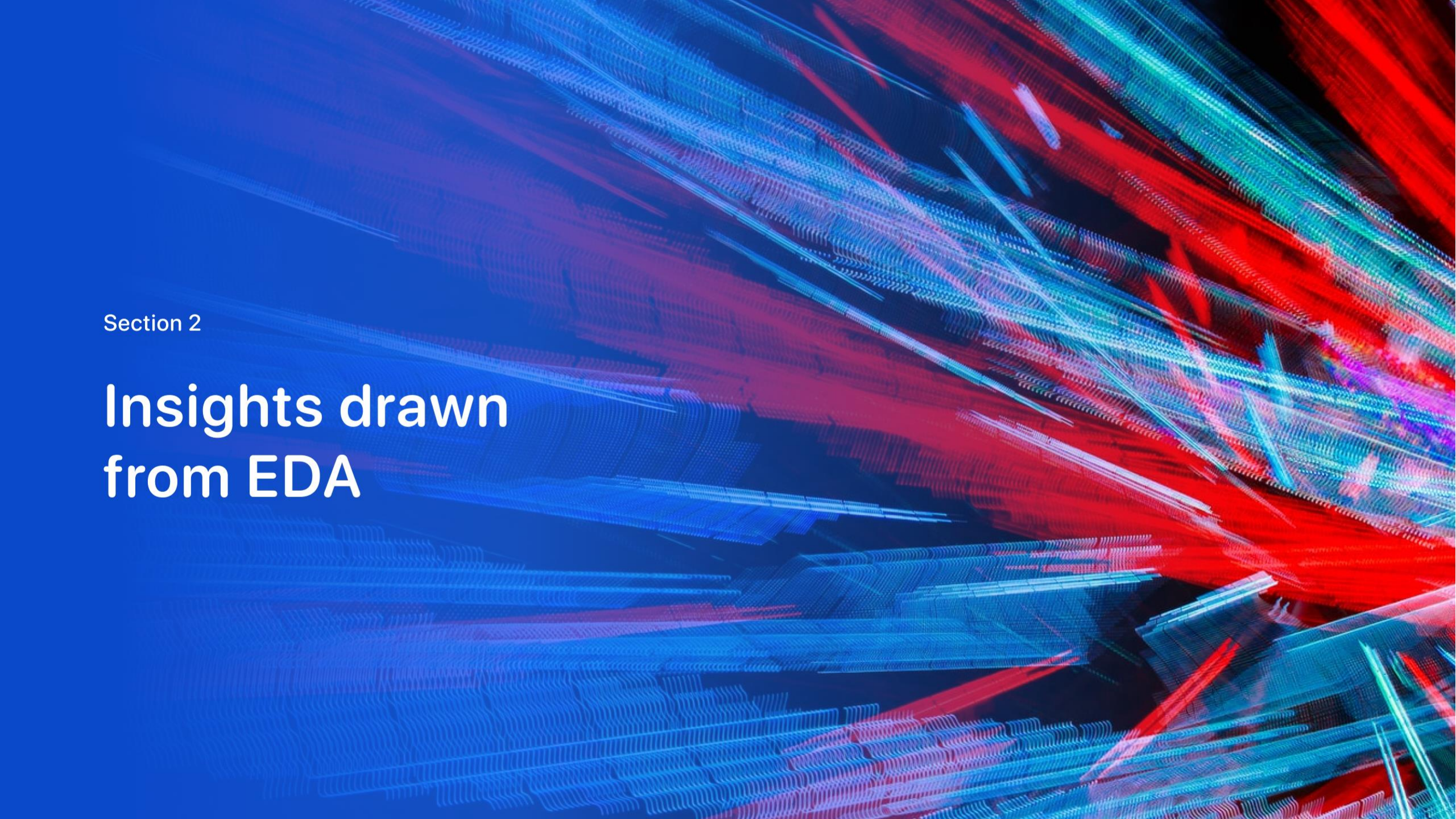
GitHub URL: Link

Results

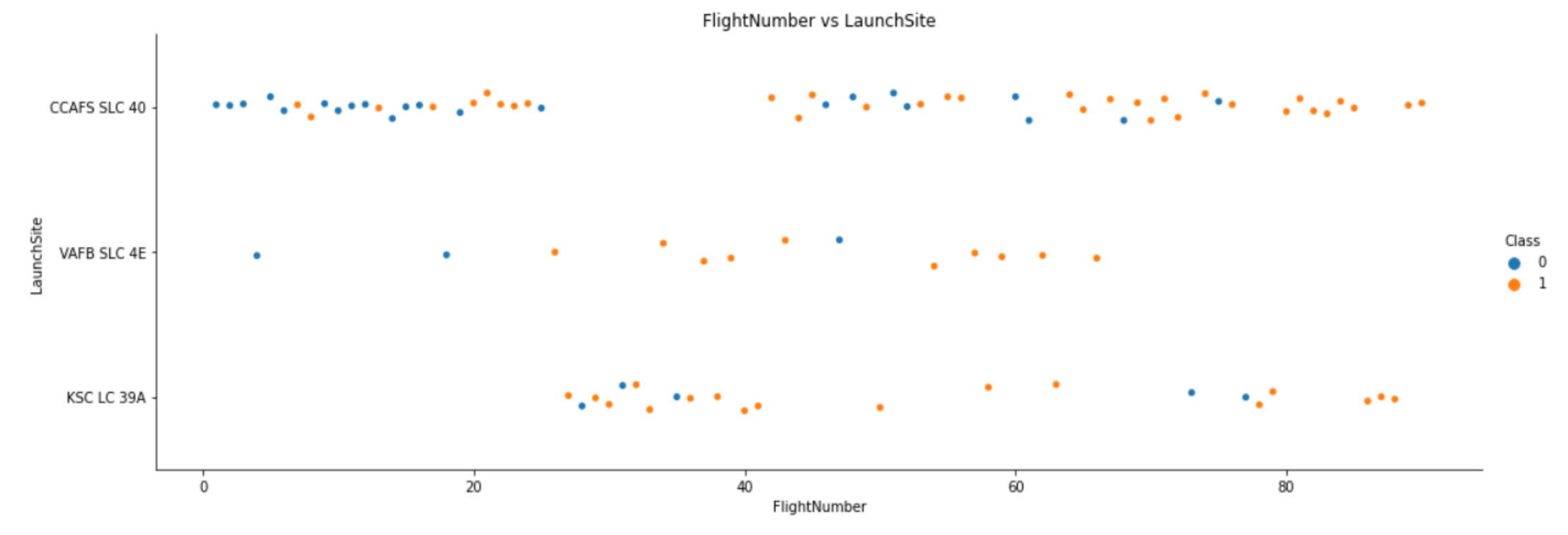
- Exploratory data analysis results
- Interactive analytics demo in screenshots
- Predictive analysis results

Section 2

Insights drawn from EDA

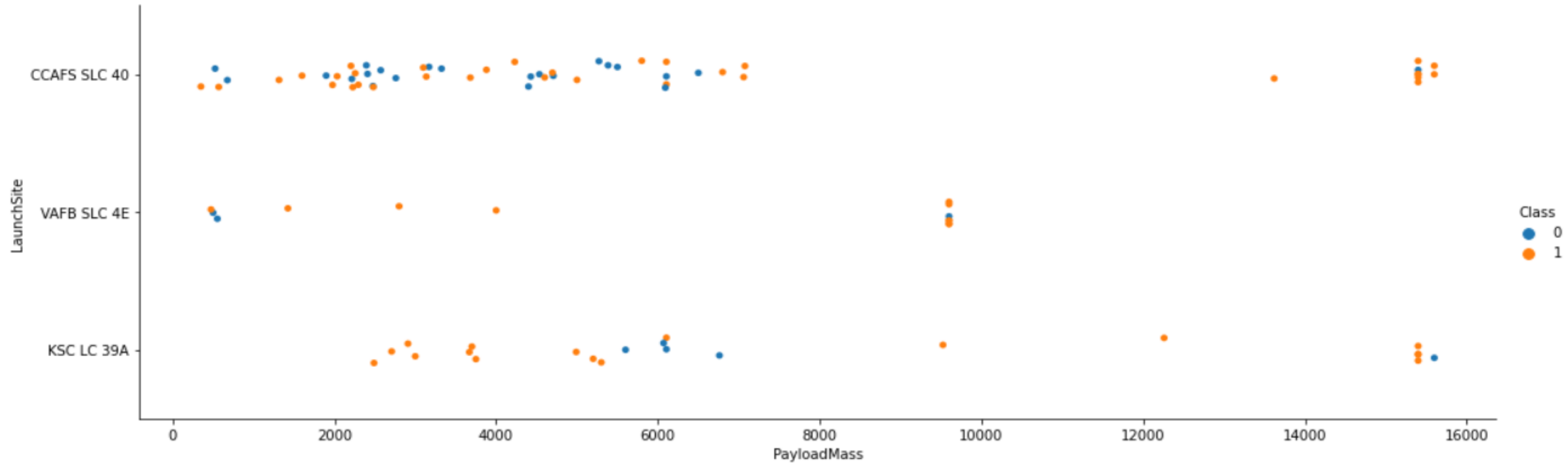


Flight Number vs. Launch Site



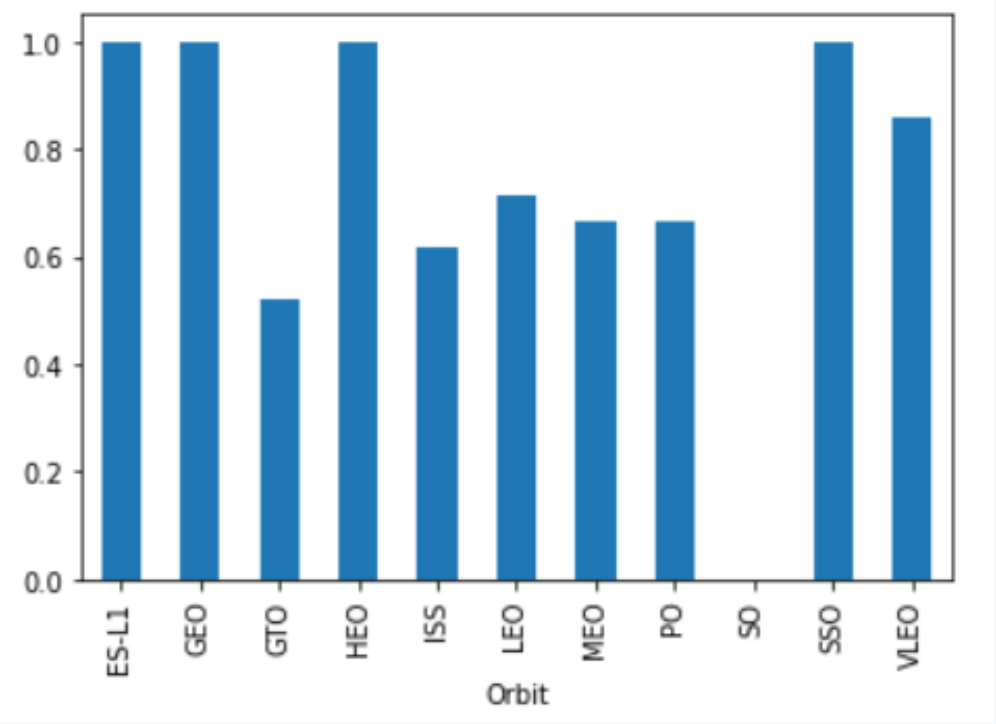
- In a general sense it can be seen that the more the number of flights from a particular site the greater the success rate.

Payload vs. Launch Site



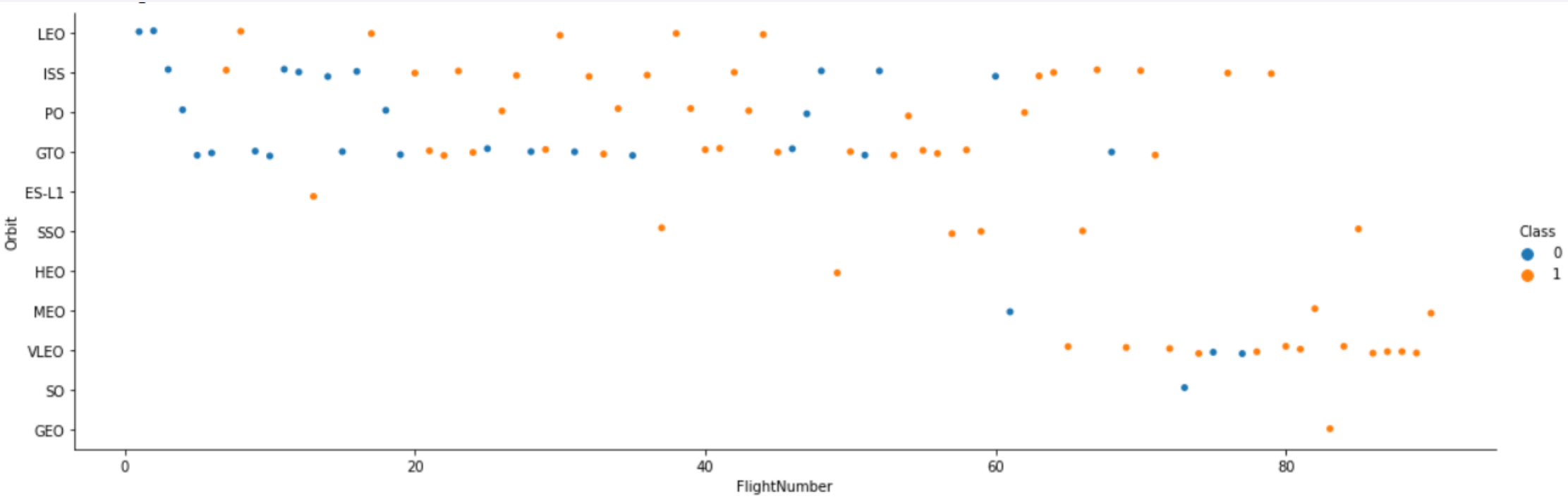
- For the CCAFS SLC 40 launch site there is very high chances of success for payload mass greater than 10000.
- For VAFB-SLC launch site there are no rockets launched for payload mass greater than 10000.
- For KSC LC 39A launch site there is high chances of success for payload mass lesser than 6000.

Success Rate vs. Orbit Type



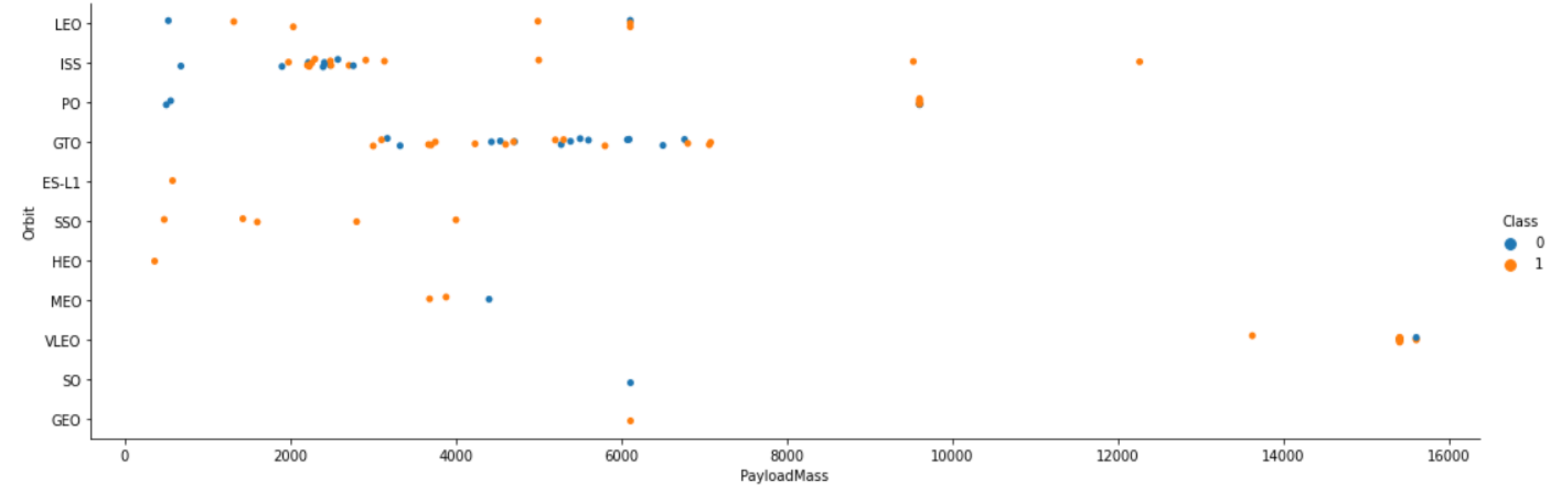
- From the chart we observe that for ES-L1, GEO, HEO and SSO orbits the rocket has the highest success rate.

Flight Number vs. Orbit Type



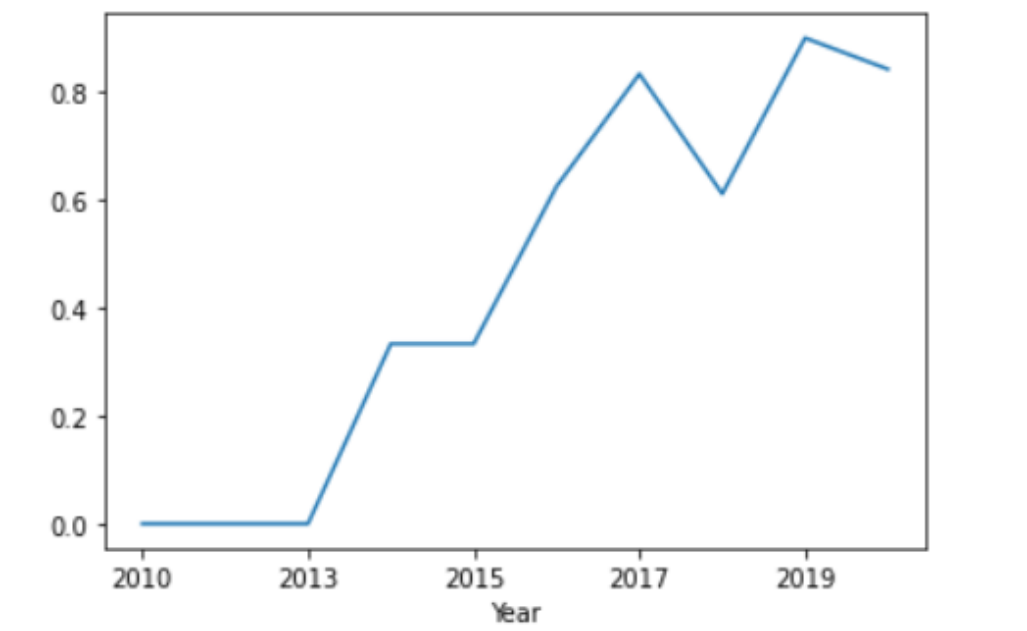
- In the LEO orbit the Success appears related to the number of flights; on the other hand, there seems to be no relationship between flight number when in GTO orbit.

Payload vs. Orbit Type



- 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.
- Orbit ES-L1, GEO, HEO and SSO has the best Success Rate.

Launch Success Yearly Trend



- We can observe that the success rate since 2013 has kept increasing till 2020.

All Launch Site Names

- Using DISTINCT since we only want the values that are unique from the LAUNCH_SITE column.

```
%sql SELECT DISTINCT LAUNCH_SITE FROM SPACE
```

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

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

Launch Site Names Begin with 'CCA'

- Here we are selecting all the columns (*), using WHERE filter the LAUNCH_SITE column to give us the data which only has CCA letters at the beginning doesn't matter what comes after these three letters (%), also limiting the output to give us 5 rows only.

```
%sql SELECT * FROM SPACE WHERE LAUNCH_SITE LIKE 'CCA%' LIMIT 5
```

```
 * sqlite:///my_data.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	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
2012-10-08	00:35:00	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

- Using the SUM function we summate the rows in the PAYLOAD_MASS__KG column, and using WHERE we filter the CUSTOMER column to select only the rows which have the data as NASA (CRS).

```
%sql SELECT SUM(PAYLOAD_MASS__KG_) FROM SPACE WHERE CUSTOMER = 'NASA (CRS)'
```

```
 * sqlite:///my_data.db
Done.
SUM(PAYLOAD_MASS__KG_)
45596
```

Average Payload Mass by F9 v1.1

- Using the SUM function we summate the rows in the PAYLOAD_MASS__KG column, and using WHERE we filter the CUSTOMER column to select only the rows which have the data as NASA (CRS).

```
%sql SELECT AVG(PAYLOAD_MASS__KG_) FROM SPACE WHERE BOOSTER_VERSION = 'F9 v1.1'
```

```
 * sqlite:///my_data.db
Done.
AVG(PAYLOAD_MASS__KG_)
2928.4
```

First Successful Ground Landing Date

- Using the min function we select the row with minimum (here oldest) data (date) in the DATE column, and using WHERE we filter the LANDING__OUTCOME column to select only the rows which have the data as Success (ground pad).

```
%sql SELECT min(DATE) FROM SPACE WHERE LANDING__OUTCOME='Success (ground pad)'
```

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

min(DATE)
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

- We select (want) the BOOSTER_VERSION column's data, using WHERE we filter the LANDING__OUTCOME column to select only the rows which have the data as 'Success (ground pad)' and we apply AND to further filter the data BETWEEN 4000 AND 6000 values from the PAYLOAD_MASS__KG column.

```
%sql SELECT BOOSTER_VERSION FROM SPACE WHERE LANDING__OUTCOME = 'Success (ground pad)' AND PAYLOAD_MASS__KG_ BETWEEN 4000 AND 6000
```

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

BOOSTER_VERSION
F9 FT B1032.1
F9 B4 B1040.1
F9 B4 B1043.1

Total Number of Successful and Failure Mission Outcomes

- We select the number of (count) data in the MISSION_OUTCOME using WHERE, we then filter values using LIKE to include only the word Success in the cells and apply AND to also include the word Failure and filter the data from the MISSION_OUTCOME column.

```
%sql SELECT COUNT(*) FROM SPACE WHERE MISSION_OUTCOME LIKE '%Success%' OR MISSION_OUTCOME LIKE '%Failure%'
```

```
 * sqlite:///my_data.db
Done.
COUNT(*)
101
```

Boosters Carried Maximum Payload

- We select (want) the BOOSTER_VERSION column's data, using WHERE we choose the data from PAYLOAD_MASS__KG column and then we use subquery to SELECT maximum value from PAYLOAD_MASS__KG column using MAX function.

```
%sql SELECT BOOSTER_VERSION FROM SPACE WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACE)
```

```
 * sqlite:///my_data.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

- We select (want) the data from LANDING_OUTCOME, BOOSTER_VERSION and LAUNCH_SITE column's data, we then use WHERE to filter the data from LANDING__OUTCOME column to give only the rows which have Failure in the cell using LIKE.

```
%sql SELECT LANDING__OUTCOME, BOOSTER_VERSION, LAUNCH_SITE, DATE FROM SPACE WHERE LANDING__OUTCOME LIKE '%Failure%' AND DATE LIKE '%2015%'
```

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

LANDING__OUTCOME	BOOSTER_VERSION	LAUNCH_SITE	DATE
Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40	2015-01-10
Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40	2015-04-14

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

- We select (want) the data from LANDING_OUTCOME and DATE column's data, we then use WHERE to filter the data from DATE column which should have values BETWEEN 2010-06-04 AND 2017-03-20 also the LANDING__OUTCOME column cells should have a value Failure (drone ship) using LIKE.

```
%sql SELECT LANDING__OUTCOME, DATE FROM SPACE WHERE DATE BETWEEN '2010-06-04' AND '2017-03-
20' AND LANDING__OUTCOME LIKE 'Failure (drone ship)' ORDER BY DATE DESC;
```

```
%sql SELECT LANDING__OUTCOME, DATE FROM SPACE WHERE DATE BETWEEN '2010-06-04' AND '2017-03-20' AND LANDI
```

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

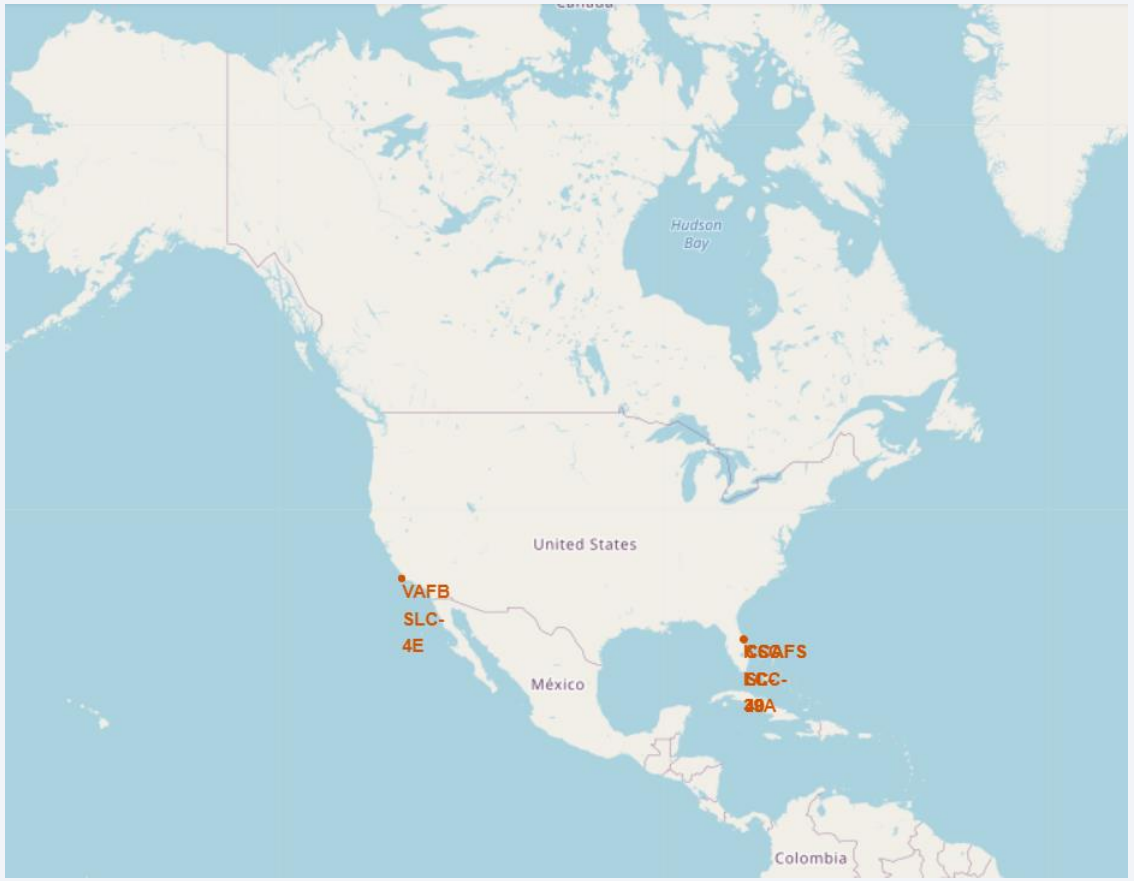
LANDING__OUTCOME	DATE
Failure (drone ship)	2016-06-15
Failure (drone ship)	2016-03-04
Failure (drone ship)	2016-01-17
Failure (drone ship)	2015-04-14
Failure (drone ship)	2015-01-10

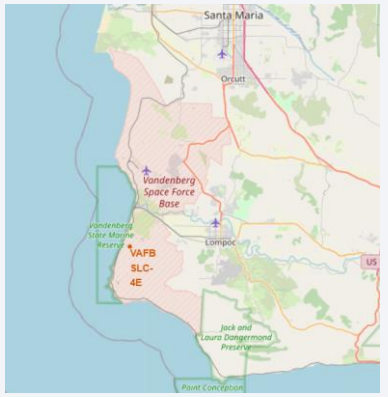
Section 4

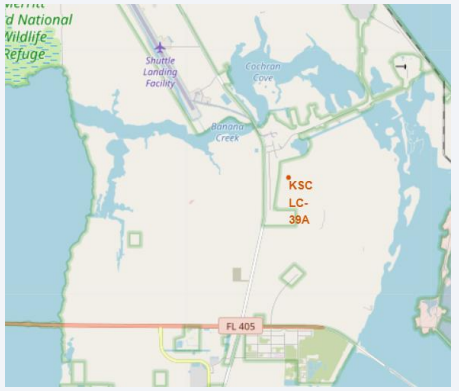
Launch Sites Proximities Analysis

Plotting all launch sites on map

Plotted all the launch sites on the map using the marker and circle objects based on the coordinates of the sites on folium map, and added a popup label.

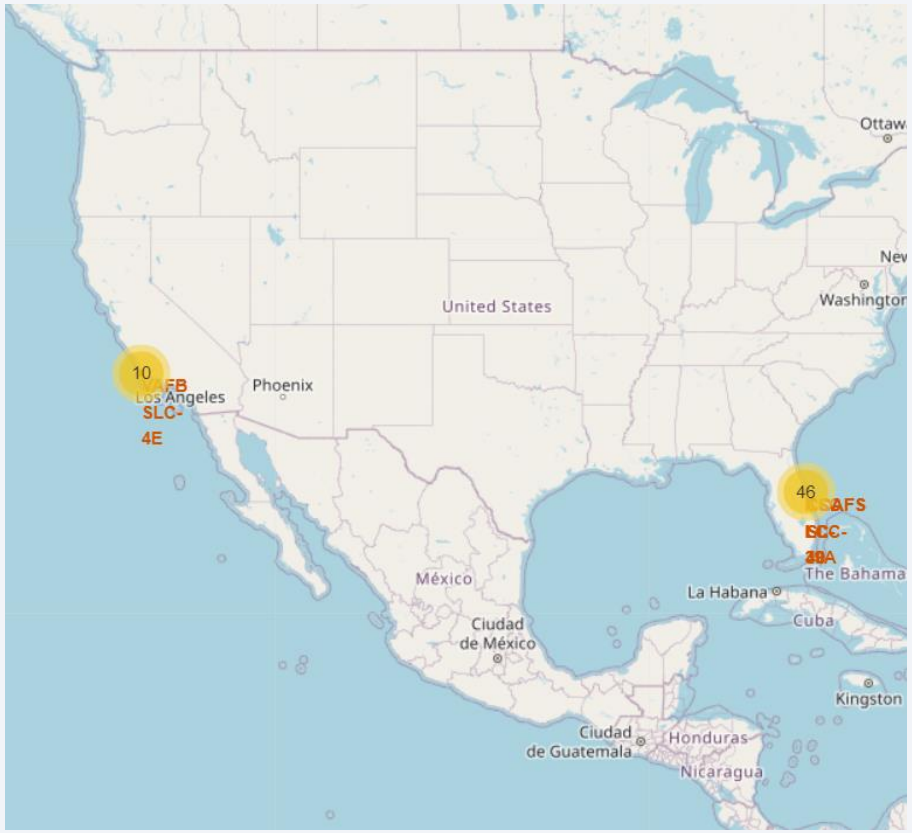




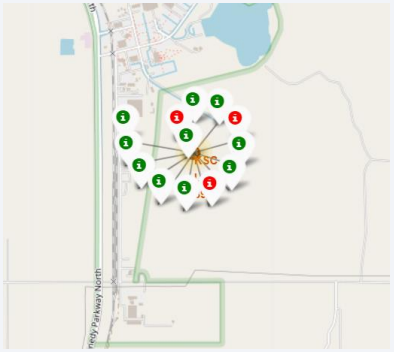


Marking Color Labelled Markers

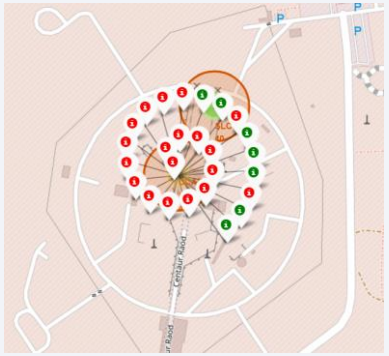
Used a Marker Cluster to represent the total number of launches and their outcomes (launch_outcome). Red represents a failed attempt and Green represents a successful attempt.

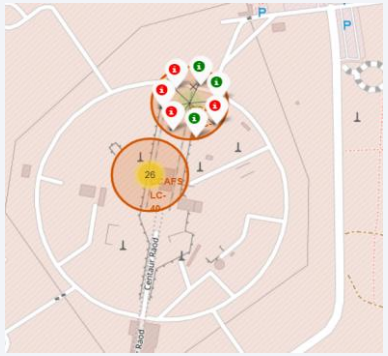


California Launch Site



Florida Launch Site

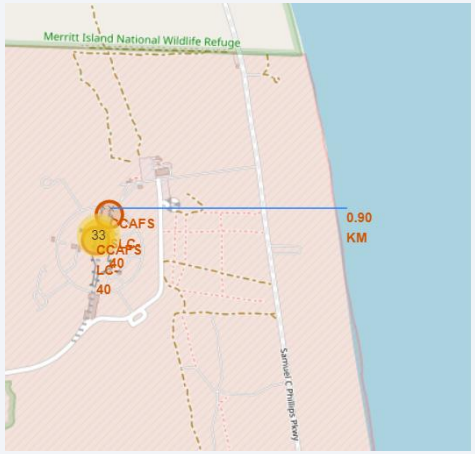




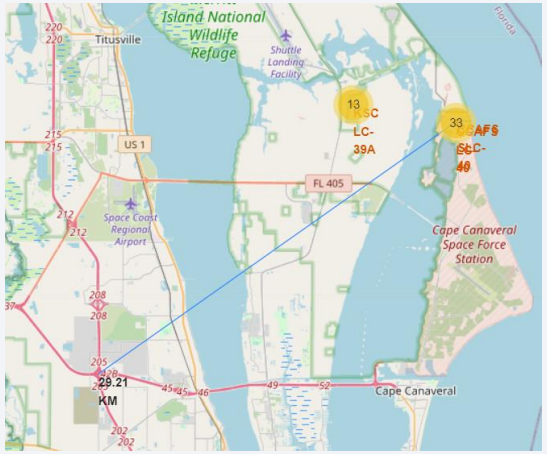
Florida (Cape Canaveral) Launch Site

Distance from closest proximities

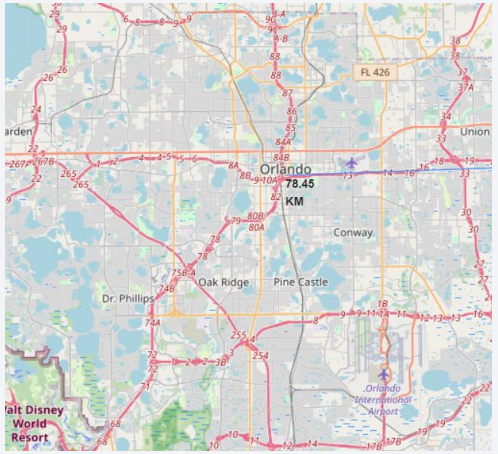
Added a polyline to the folium map representing the distance of launch sites from its closest locations such as coastline, highway, city and railway station.



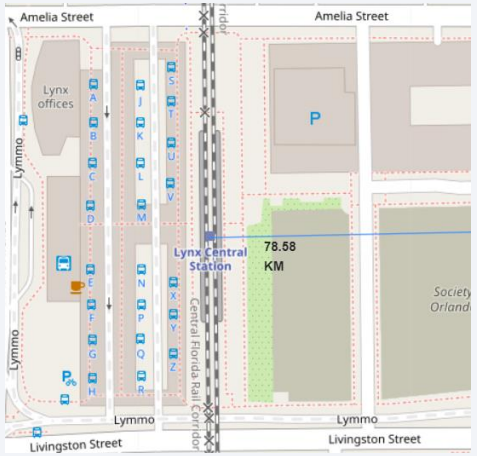
Distance to closest coastline



Distance to closest highway



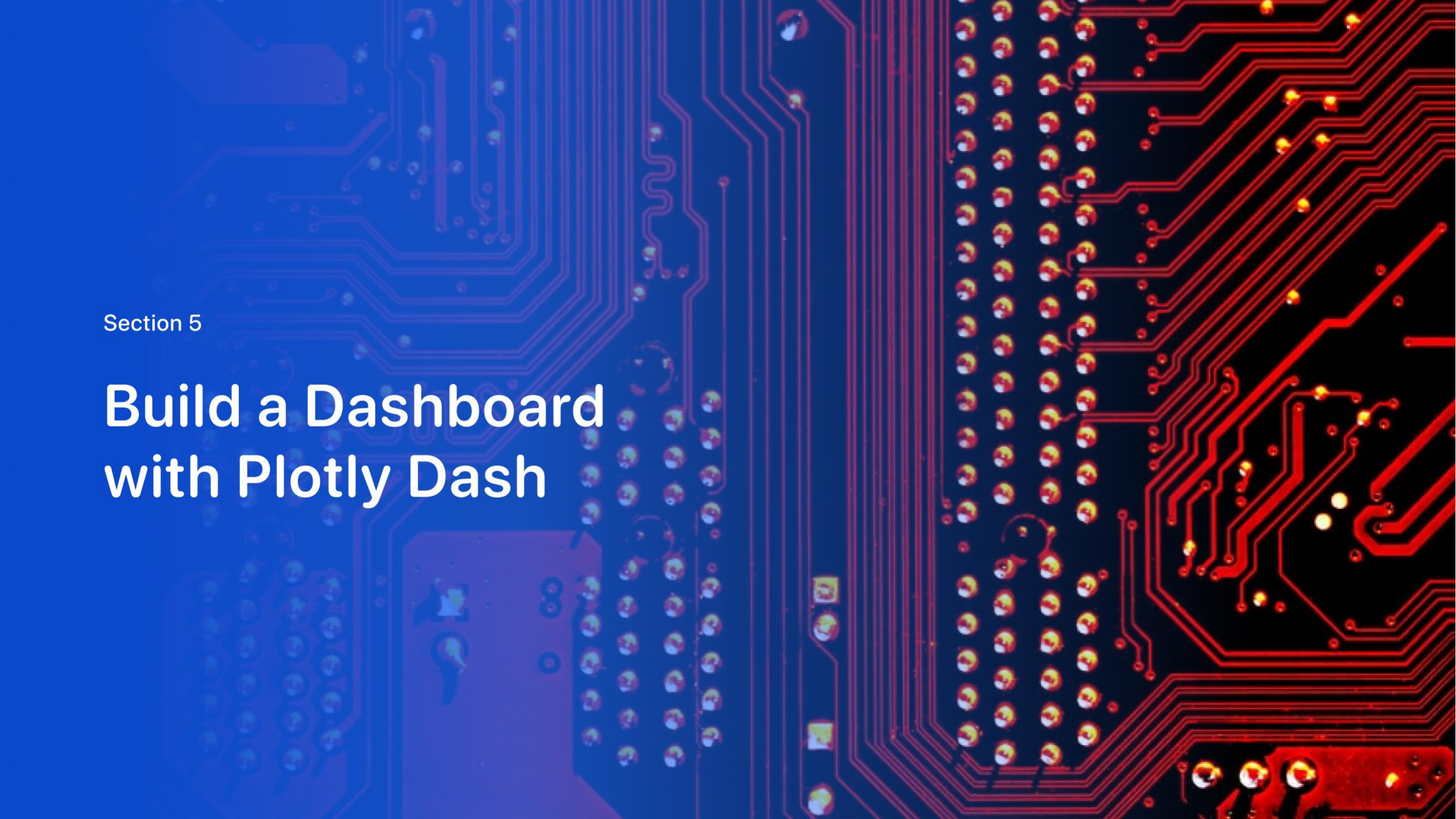
Distance to closest city



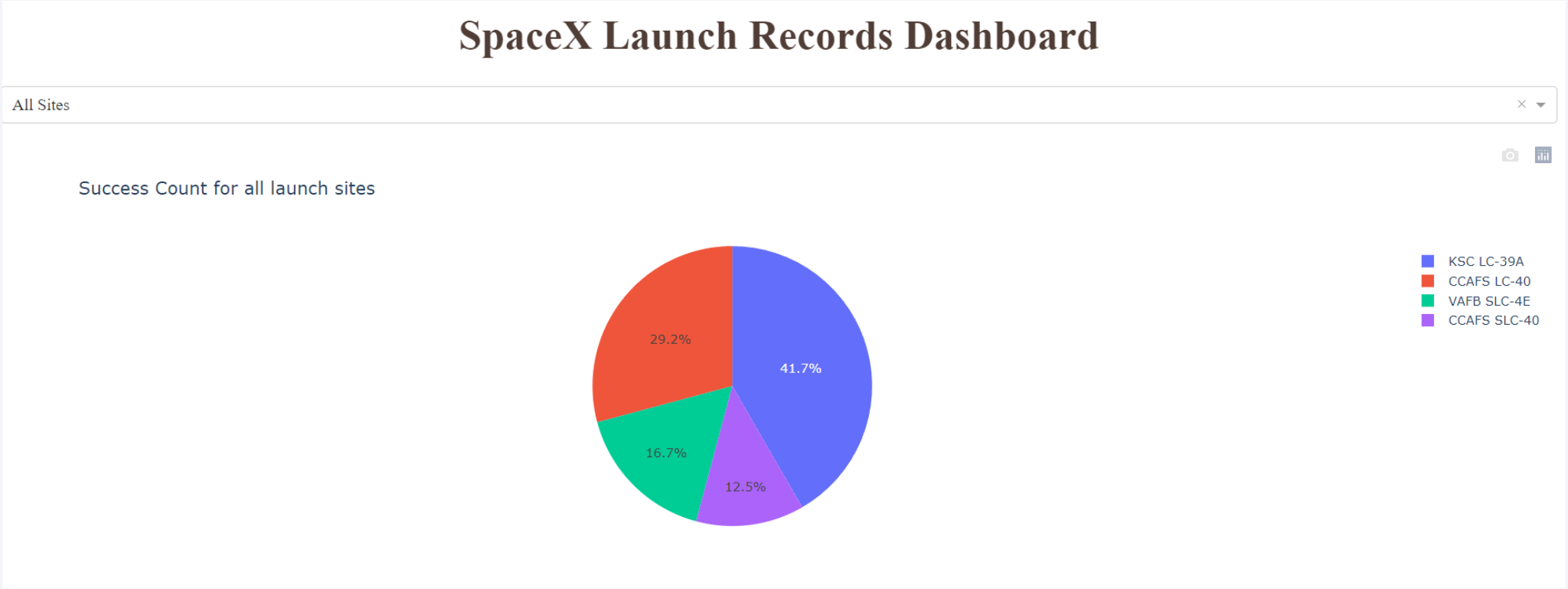
Distance to closest railway station

Section 5

Build a Dashboard with Plotly Dash

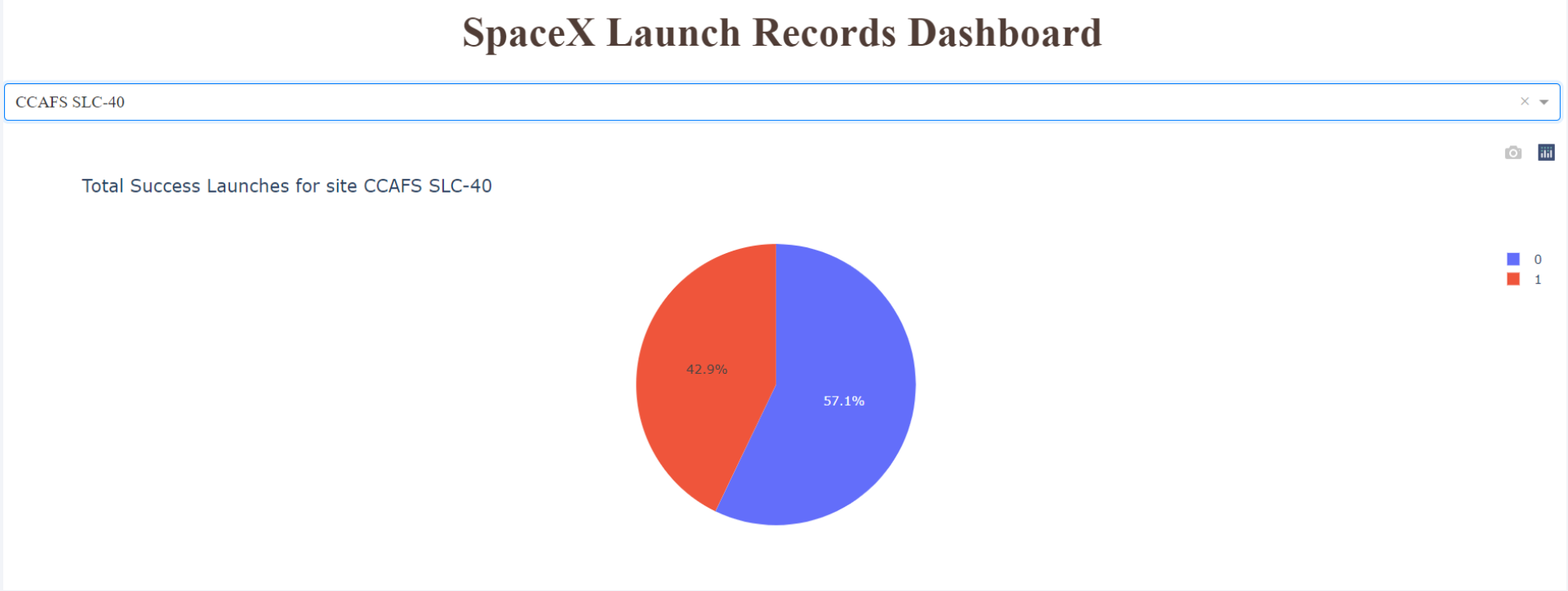


Launch Success Count for all sites



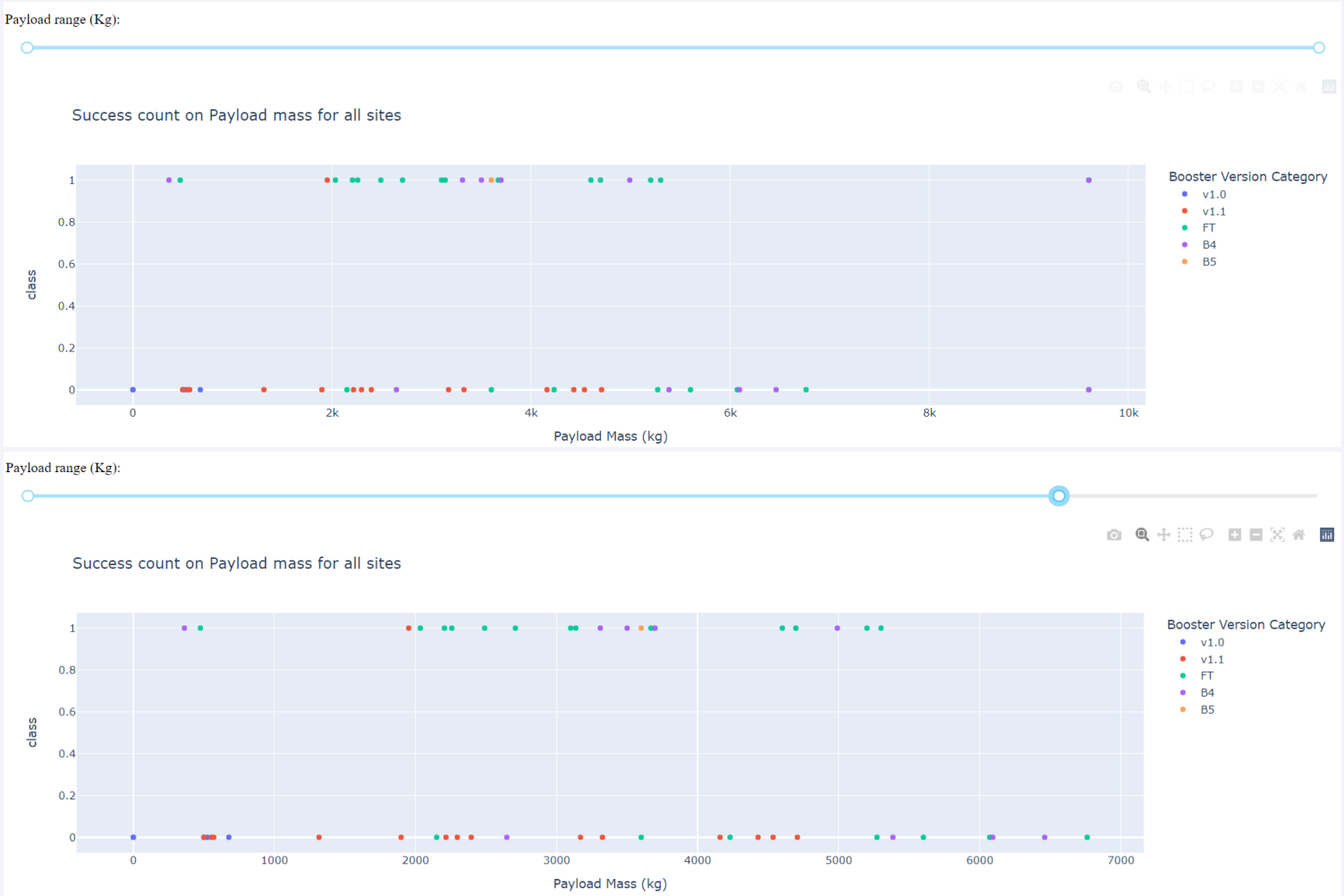
- From the above chart it can be said that the site KSC LC-39A has the highest success percentage while the site CCAFS SLC-40 has the lowest success percentage when compared with others.

Site with Highest Launch Success Ratio



- After viewing the charts for all the sites it was found that the site CCAFS SLC-40 has the highest launch success ratio.

Payload vs. Launch Outcome scatter plot



- From the Payload vs. Launch Outcome scatter plot for all sites it can be said that the 'FT' Booster and the payload range between 2000-4000 kg has the largest success rate.

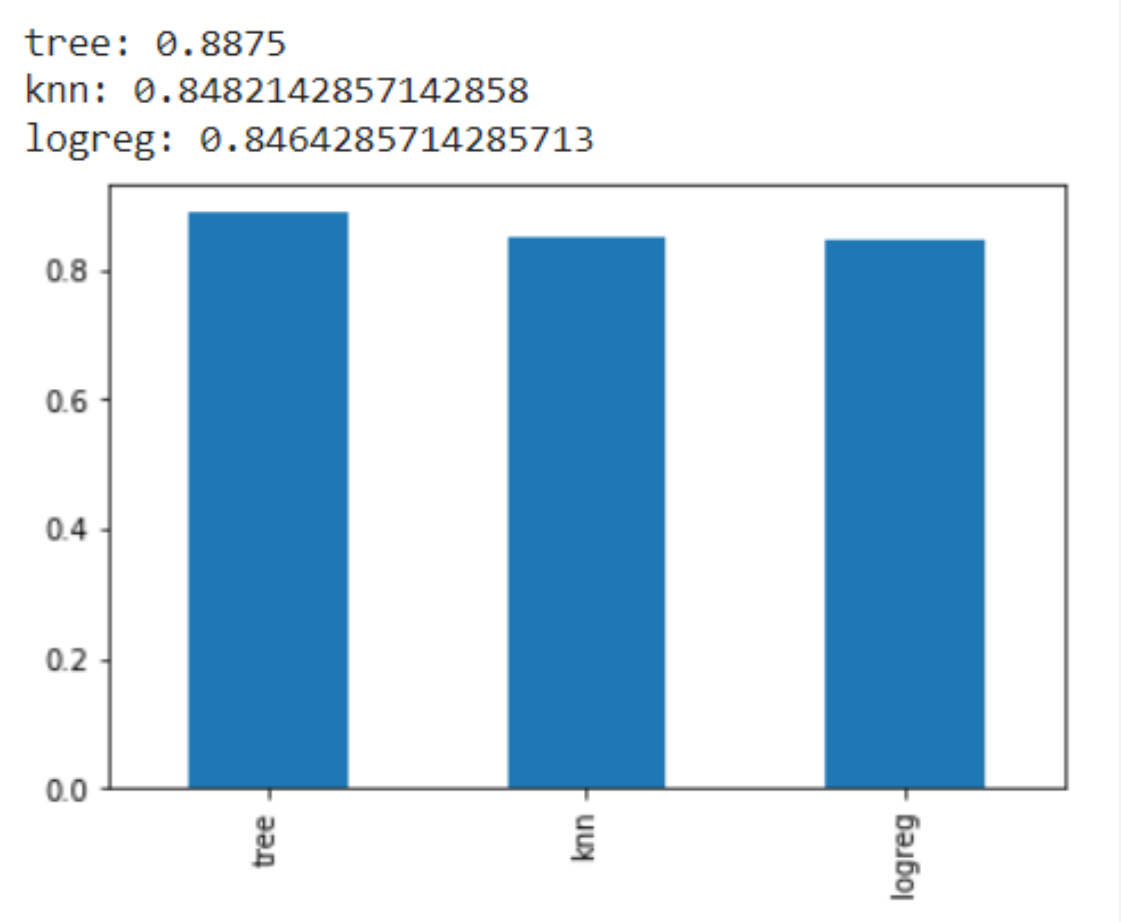
Section 6

Predictive Analysis (Classification)

Classification Accuracy

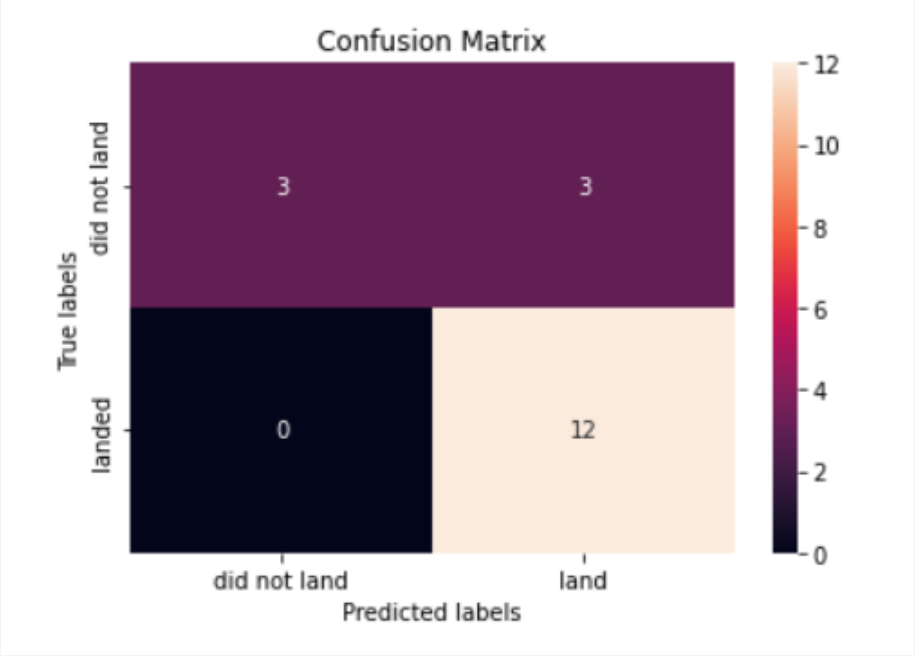
- As can be seen from the plot, the Decision Tree Classifier has the highest classification accuracy.

```
tree: 0.8875
knn: 0.8482142857142858
logreg: 0.8464285714285713
```



Confusion Matrix

- The best algorithm is Decision Tree Classifier with an accuracy of 88.75%.
- Most of our predicted values stands correct but we have some false positives results for our prediction.



Conclusions

- The Decision Tree Classifier Algorithm is the best for Machine Learning for this dataset.
- High Payloads (greater than 7000 kg) perform better than the lower payloads.
- The success rate for launches is directly proportional to time in years, i.e., every year the launch success increase.
- We can see that KSC LC 39A site had the most successful launch rates among the other sites.
- Orbit ES-L1, GEO, HEO and SSO has the best Success Rate.

Appendix

- Link to my GitHub Repository: Link
- Used Google Colab for Jupyter Notebook: https://research.google.com/colaboratory/
- References:

  https://towardsdatascience.com/

  https://www.geeksforgeeks.org/

Thank you!