IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Takayuki Hayakawa
August 16, 2024

Outline

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

Executive Summary

- Summary of methodologies
  - Collecting data on the Falcon 9 first-stage landings using a RESTful API and web scraping. Also converting the data into a dataframe and then perform some data wrangling.
  - Next building a dashboard to analyze launch records interactively with Plotly Dash. Then building an interactive map to analyze the launch site proximity with Folium.
  - Finally using machine learning to determine if the first stage of Falcon 9 will land successfully. Splitting the data into training data and test data to find the best Hyperparameter for SVM, Classification Trees, and Logistic Regression. Then find the method that performs best using test data.
- Summary of all results
  - Utilizing the data analysis tools to load a dataset, cleaning it, and finding out interesting insights from it.
  - Building an interactive map to analyze the launch site proximity.
  - Utilizing the machine learning skills to build a predictive model to help a business function more efficiently.

Introduction

- Project background and context
  - The commercial space age is here, companies are making space travel affordable for everyone. Perhaps the most successful is SpaceX. One reason SpaceX can do this is the rocket launches are relatively inexpensive.
  - SpaceX advertises Falcon 9 rocket launches on its website with a cost of 62 million dollars; other providers cost upwards of 165 million dollars each, much of the savings is because SpaceX can reuse the first stage.
  - Therefore, if we can determine if the first stage will land, we can determine the cost of a launch.

- Problems you want to find answers
  - The problem is to determine the price of each launch. Doing this by gathering information about Space X and creating dashboards and also to determine if SpaceX will reuse the first stage.
  - Instead of using rocket science to determine if the first stage will land successfully, I train a machine learning model and use public information to predict if SpaceX will reuse the first stage.

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Collecting data on the Falcon 9 first-stage landings using a RESTful API and web scraping.
- Perform data wrangling
  - Converting the data into a dataframe and then perform some data wrangling.
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Using machine learning to determine if the first stage of Falcon 9 will land successfully. First splitting the data into training data and test data to find the best Hyperparameter for SVM, Classification Trees, and Logistic Regression. Then finding the method that performs best using test data.

Data Collection

- Describe how data sets were collected.
  - To collecting data sets, making a get request to the SpaceX API, and also doing some basic data wrangling and formating.
  - Request to the SpaceX API
  - Clean the requested data
- You need to present your data collection process use key phrases and flowcharts

1. Import Libraries and Define Auxiliary Functions
2. Request rocket launch data from SpaceX API with the following URL:

   https://api.spacexdata.com/v4/launches/past

3. Request and parse the SpaceX launch data using the GET request

   https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DS0321EN-SkillsNetwork/datasets/API_call_spacex_api.json

4. Filter the dataframe to only include Falcon 9 launches
5. Dealing with Missing Values

Data Collection – SpaceX API

- Present your data collection with SpaceX REST calls using key phrases and flowcharts

- Add the GitHub URL of the completed SpaceX API calls notebook (must include completed code cell and outcome cell), as an external reference and peer-review purpose

  https://github.com/briskwalking53/SpaceXFalcon9firststageLandingPrediction/blob/main/jupyter-labs-spacex-data-collection-api.ipynb

1. Request rocket launch data from SpaceX API with the following URL:
2. response = requests.get(spacex_url)

Data Collection - Scraping

- Present your web scraping process using key phrases and flowcharts

- Add the GitHub URL of the completed web scraping notebook, as an external reference and peer-review purpose

https://github.com/briskwalking53/SpaceXFalcon9firststageLandingPrediction/blob/main/jupyter-labs-spacex-data-collection-api.ipynb

1. To make the requested JSON results more consistent, use the following static response object for this project:
2. Decode the response content as a Json using .json() and turn it into a Pandas dataframe using .json_normalize()
3. To get information about the launches using the IDs given for each launch, and also specifically using columns rocket, payloads, launchpad, and cores.
4. Remove the Falcon 1 launches keeping only the Falcon 9 launches. Filter the data dataframe using the BoosterVersion column to only keep the Falcon 9 launches. Save the filtered data to a new dataframe called data_falcon9.

Data Wrangling

- Describe how data were processed
  - Dealing with Missing Values
  - The LandingPad column will retain None values to represent when landing pads were not used.
- You need to present your data wrangling process using key phrases and flowcharts
- Add the GitHub URL of your completed data wrangling related notebooks, as an external reference and peer-review purpose

  https://github.com/briskwalking53/SpaceXFalcon9firststageLandingPrediction/blob/main/jupyter-labs-spacex-data-collection-api.ipynb

1. Calculate the mean for the PayloadMass using the .mean().
2. Then use the mean and the .replace() function to replace np.nan values in the data with the mean calculated above.

EDA with Data Visualization

- Summarize what charts were plotted and why you used those charts
  - Plot out the FlightNumber vs. PayloadMass to see how their variables would affect the launch outcome.
  - Plot out the FlightNumber vs LaunchSite to see the relationship between Flight Number and Launch Site.
  - Plot out the Payload Mass and Launch Site to see the relationship between Payload Mass and Launch Site.
  - Plot out the Orbit and Class to see the relationship between success rate of each orbit type.
  - Plot out the FlightNumber and Orbit to see the relationship between FlightNumber and Orbit type.
  - Plot out the Payload Mass and Orbit to see the relationship between Payload Mass and Orbit type.
  - Plot out the Year and average success rateto see the launch success yearly trend
- Add the GitHub URL of your completed EDA with data visualization notebook, as an external reference and peer-review purpose

https://github.com/briskwalking53/SpaceXFalcon9firststageLandingPrediction/blob/main/edadataviz.ipynb

EDA with SQL

- Using bullet point format, summarize the SQL queries you performed
  - Display the names of the unique launch sites in the space mission
  - Display 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 landing outcomes between the date 2010-06-04 and 2017-03-20, in descending order.
- Add the GitHub URL of your completed EDA with SQL notebook, as an external reference and peer-review purpose

  https://github.com/briskwalking53/SpaceXFalcon9firststageLandingPrediction/blob/main/jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

- Summarize what map objects such as markers, circles, lines, etc. you created and added to a folium mapa nd explain why you added those objects
  - Create a folium Map object to see an initial center location to be NASA Johnson Space Center at Houston, Texas.
  - Add a circle for each launch site in data frame launch_sites.
  - Create markers for all launch records to see if a launch was successful.
  - Add a MousePosition on the map to get coordinate for a mouse over a point on the map.
  - Mark down a point on the closest coastline using MousePosition and calculate the distance between the coastline point and the launch site.
  - Draw a line between a launch site to its closest city, railway, highway, and etc. to answer launch sites in close proximity to railways, haigway, coastline and city.
- Add the GitHub URL of your completed interactive map with Folium map, as an external reference and peer-review purpose

https://github.com/briskwalking53/SpaceXFalcon9firststageLandingPrediction/blob/main/lab_jupyter_launch_site_location.ipynb

Build a Dashboard with Plotly Dash

- Summarize what plots/graphs and interactions you have added to a dashboard and explain why you added those plots and interactions
  - Pie chart with drop-down list to see the total sucess launches for sites.
  - Scatter point chart with range-slider to see the correltion between Payload and Success for sites
- Add the GitHub URL of your completed Plotly Dash lab, as an external reference and peer-review purpose

  https://github.com/briskwalking53/SpaceXFalcon9firststageLandingPrediction/blob/main/spacex_dash_app.py

Predictive Analysis (Classification)

- Perform exploratory Data Analysis and determine Training Labels
  - ✓Create a column for the class
  - ✓Standardize the data
  - ✓Split into training data and test data
- Find best Hyperparameter for SVM, Classification Trees and Logistic Regression
  - ✓Find the method performs best using test data
- Add the GitHub URL of your completed predictive analysis lab, as an external reference and peer-review purpose

https://github.com/briskwalking53/SpaceXFalcon9firststageLandingPrediction/blob/main/SpaceX_Machine Learning Prediction_Part_5.ipynb

1. Load the data.
2. Create a NumPy array from the column Class in data, then assign it to the variable Y.
3. Standardize the data in X then reassign it to the variable X.
4. Use the function train_test_split to split the data X and Y into training and test data.
5. Create a logistic regression object then create a GridSearchCV object logreg_cv with cv = 10. Fit the object to find the best parameters from the dictionary parameters.
6. Calculate the accuracy on the test data using the method score.
7. Create a support vector machine object then create a GridSearchCV object svm_cv with cv = 10. Fit the object to find the best parameters from the dictionary parameters.
8. Calculate the accuracy on the test data using the method score.
9. Create a decision tree classifier object then create a GridSearchCV object tree_cv with cv = 10. Fit the object to find the best parameters from the dictionary parameters.
10. Calculate the accuracy of tree_cv on the test data using the method score.
11. Create a k nearest neighbors object then create a GridSearchCV object knn_cv with cv = 10. Fit the object to find the best parameters from the dictionary parameters.
12. Calculate the accuracy of knn_cv on the test data using the method score.
13. Find the method performs best.

Results

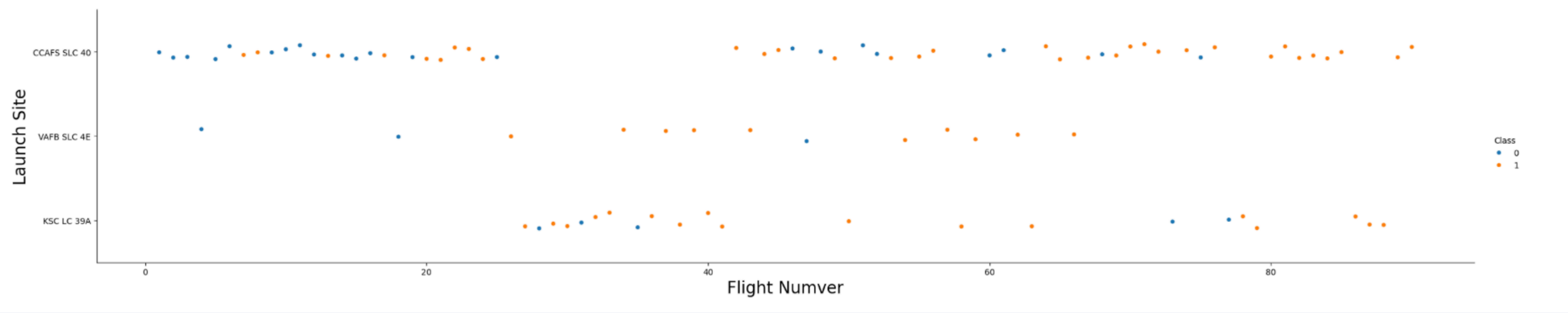
- Exploratory data analysis results
  - ✓The larger the Flight Number, the better the success rate at launch sites
  - ✓The VAFB-SLC launch site there are no rockets launched for heavypayload mass(greater than 10000).
  - ✓ES-L1, GEO, HEO and SSO have the highest success rates.
  - ✓The success rate since 2013 kept increasing till 2020.
  - ✓Find the total number of successful and failure mission outcomes
- Interactive analytics demo in screenshots
  - ✓Interactive analytics is able to identify launch sites easier.
- Predictive analysis results
  - ✓Predictive analysis shows Decision Tree is the best model to predict successful landings, having accuracy about 87%.

Section 2

Insights drawn from EDA

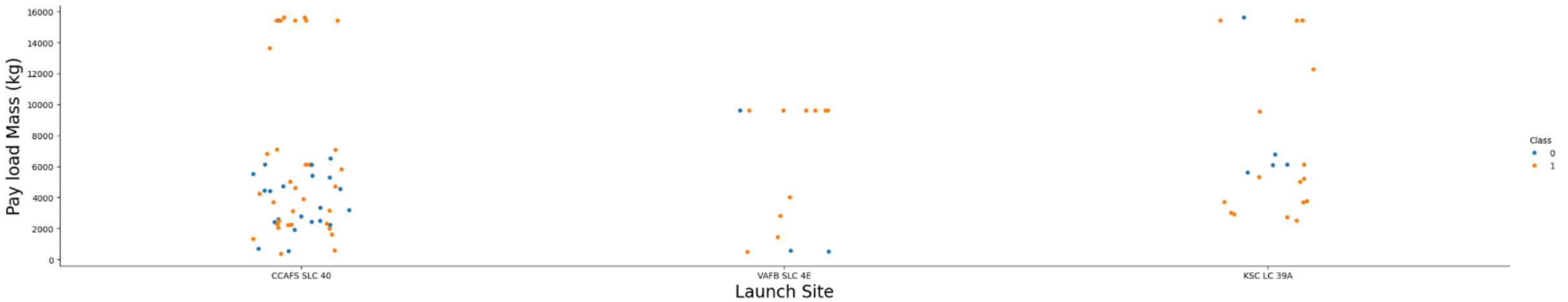
Flight Number vs. Launch Site

- The larger the Flight Number, the better the success rate at launch sites



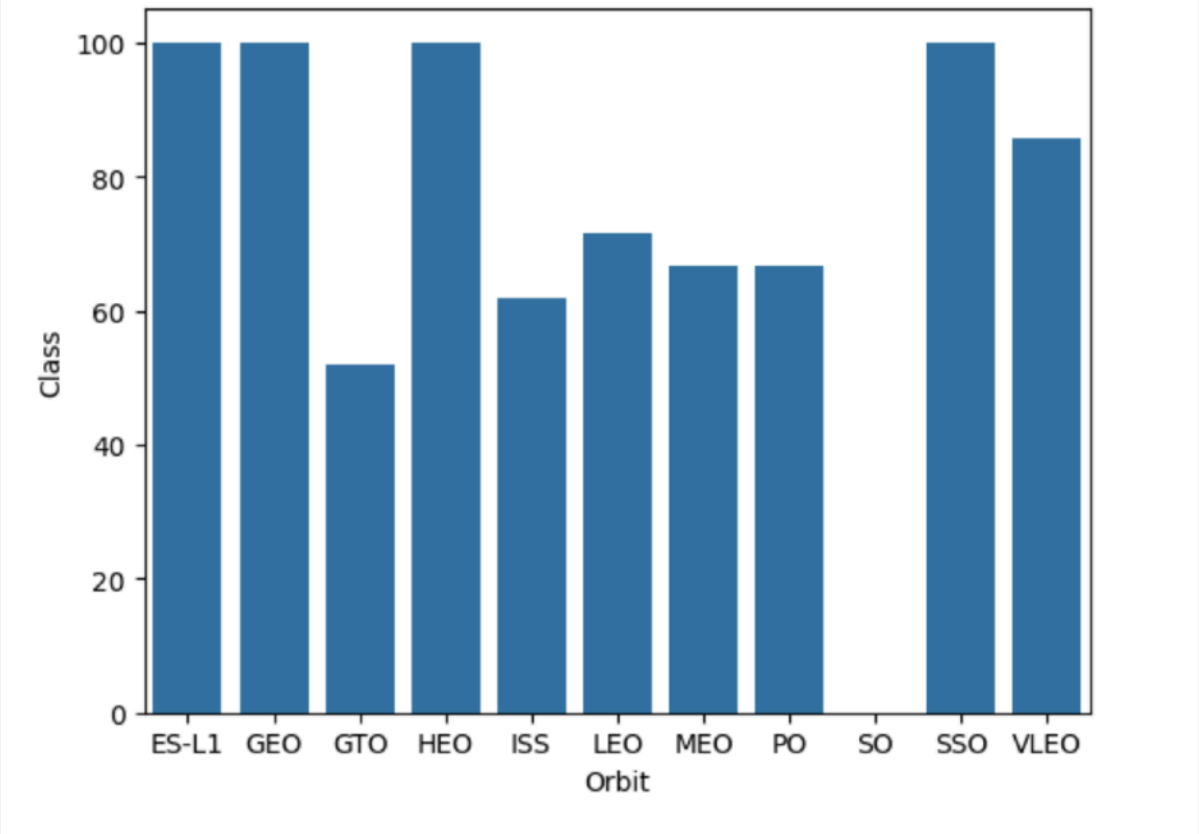
Payload vs. Launch Site

- The VAFB-SLC launch site there are no rockets launched for heavypayload mass(greater than 10000).



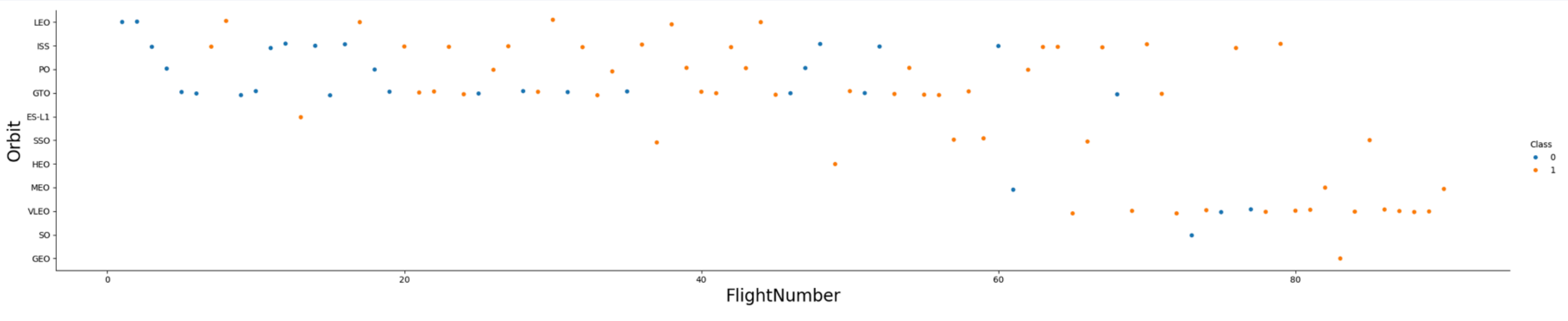
Success Rate vs. Orbit Type

- ES-L1, GEO, HEO and SSO have the highest success rates.



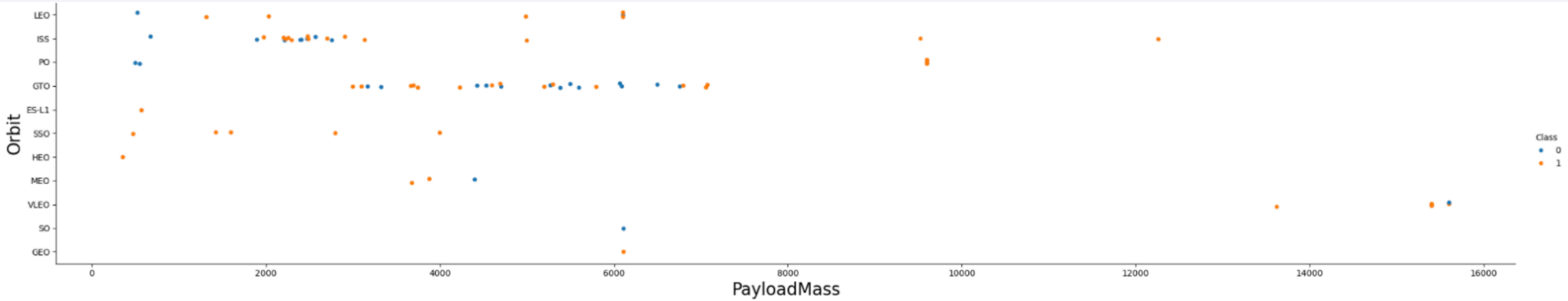
Flight Number vs. Orbit Type

- In the LEO orbit, success is related to the number of flights. Conversely, in the GTO orbit, there appears to be no relationship between flight number and success.



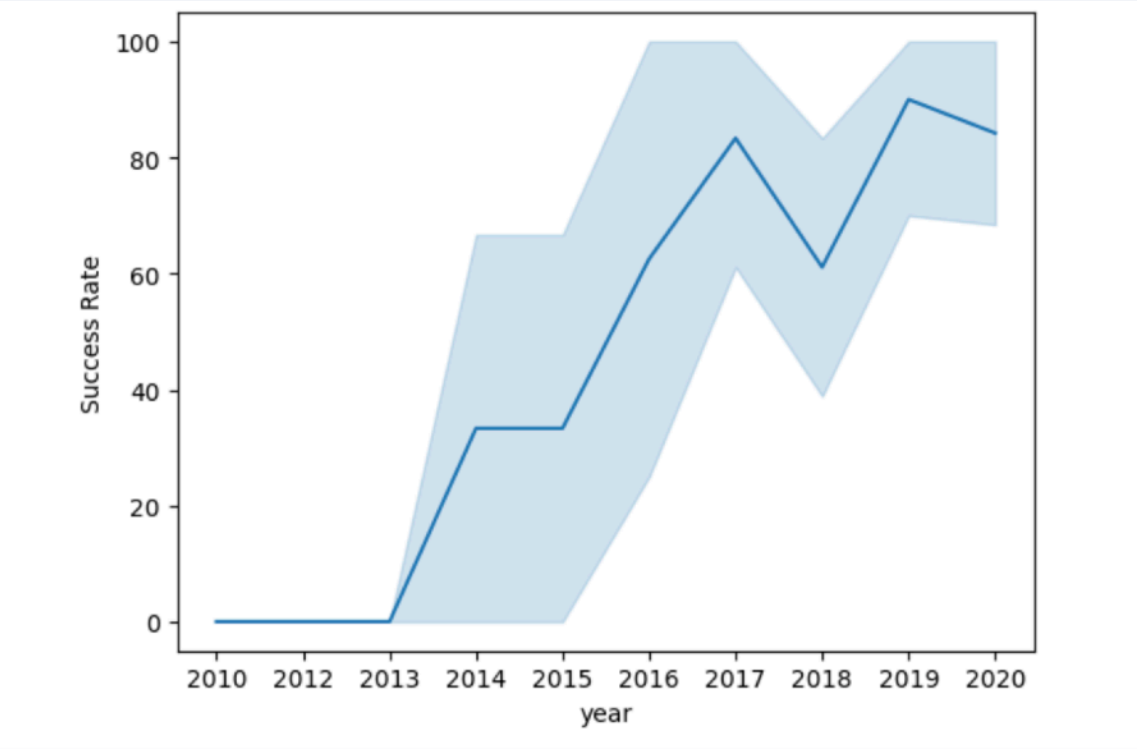
Payload vs. Orbit Type

- With heavy payloads the successful landing or positive landing rate are more for Polar,VLEO and ISS.
- However, for GTO, it's difficult to distinguish between successful and unsuccessful landings as both outcomes are present.



Launch Success Yearly Trend

- The success rate since 2013 kept increasing till 2020.



All Launch Site Names

- Display the names of the unique launch sites in the space mission using DISTINCT.

In [10]:
```
%sql select distinct launch_site from SPACEXTBL
```

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

Out[10]:

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

Launch Site Names Begin with 'CCA'

- Display 5 records where launch sites begin with the string ‘CCA' using LIKE,% and LIMIT.

In [11]:
```
%sql select * from SPACEXTBL where launch_site like 'CCA%' limit 5
```

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

Out[11]:

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_
2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (p
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 (p
2012-05-22	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	N
2012-10-08	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	N
2013-03-01	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	N

Total Payload Mass

- Display the total payload mass carried by boosters launched by NASA (CRS) using SUM.

```
%sql select sum(payload_mass__kg_) as sum from SPACEXTBL where customer like 'NASA (CRS)'
```

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

sum
45596

Average Payload Mass by F9 v1.1

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

```
%sql select avg(payload_mass__kg_) as average from SPACEXTBL where booster_version like 'F9 v1.1%'
```

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

average
2534.6666666666665

First Successful Ground Landing Date

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

```
%sql select min(date) as date from SPACEXTBL where mission_outcome like 'Success'
```

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

date
2010-06-04

Successful Drone Ship Landing with Payload between 4000 and 6000

- 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 SPACEXTBL where (mission_outcome like 'Success')   and (landing_outcome like 'Su
```

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

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

Total Number of Successful and Failure Mission Outcomes

- List the total number of successful and failure mission outcomes using COUNT, GROUP BY and ORDER BY.

```
%sql select mission_outcome, count(*) as count from SPACEXTBL group by mission_outcome order by mission_outcome
```

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

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

Boosters Carried Maximum Payload

- List the names of the booster_versions which have carried the maximum payload mass using a subquery

```
%sql select booster_version from SPACEXTBL where payload_mass__kg_ = (select max(payload_mass__kg_) from SPACEXTB
```

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

Booster_Version
F9 B5 B1048.4
F9 B5 B1049.4
F9 B5 B1051.3
F9 B5 B1056.4
F9 B5 B1048.5
F9 B5 B1051.4
F9 B5 B1049.5
F9 B5 B1060.2
F9 B5 B1058.3
F9 B5 B1051.6
F9 B5 B1060.3
F9 B5 B1049.7

2015 Launch Records

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

```
%sql select substr(Date,6,2) as month, landing_outcome, booster_version, launch_site from SPACEXTBL where substr(
```

```
 * 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

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

- Rank the count of landing outcomes (such as Failure (drone ship) or Success (ground pad)) between the date 2010-06-04 and 2017-03-20, in descending order.

```
%sql select landing_outcome, count(*) as count from SPACEXTBL where date >= '2010-06-04' and date <= '2017-03-20'
```

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

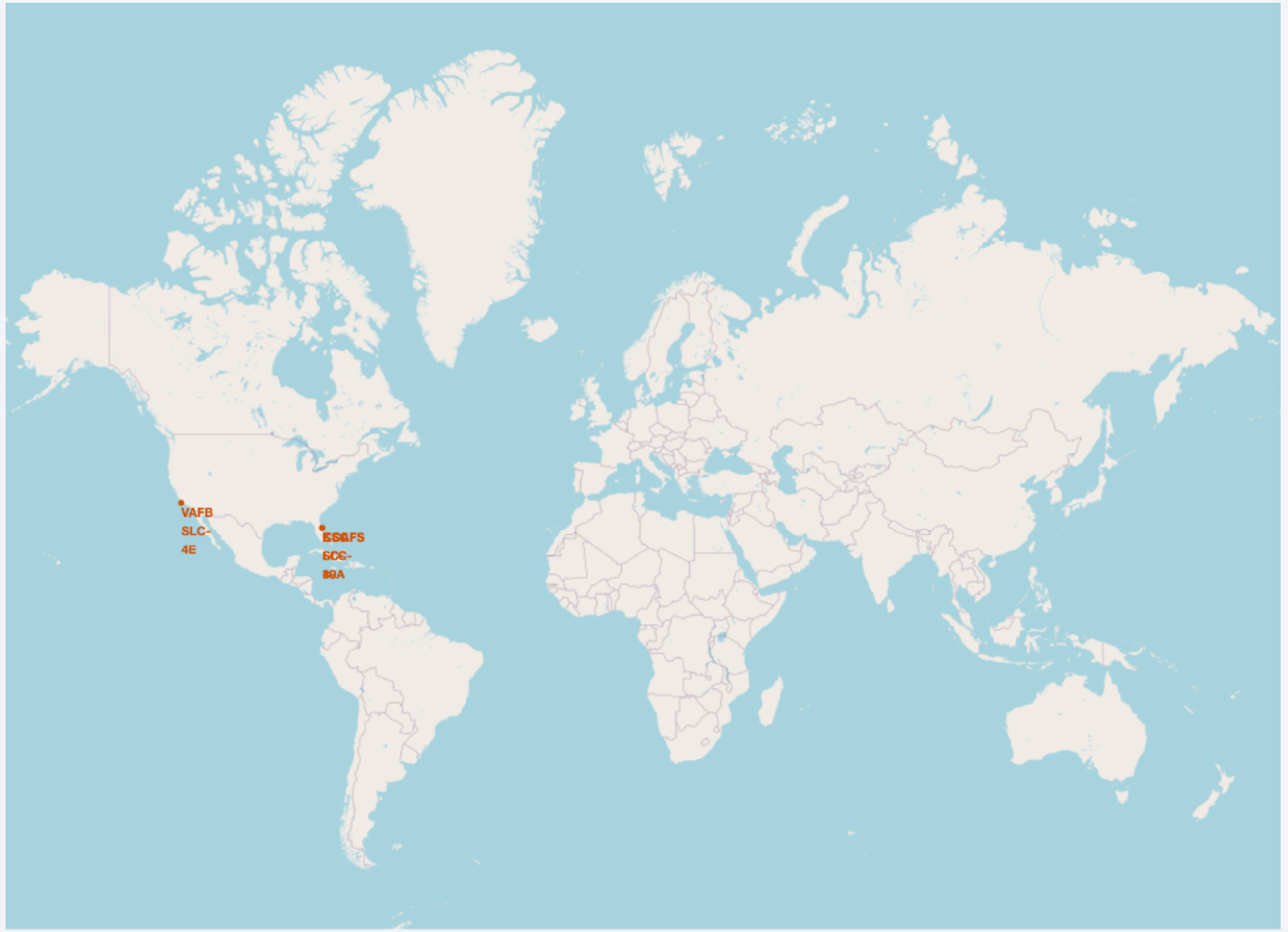
Landing_Outcome	count
No attempt	10
Success (drone ship)	5
Failure (drone ship)	5
Success (ground pad)	3
Controlled (ocean)	3
Uncontrolled (ocean)	2
Failure (parachute)	2
Precluded (drone ship)	1

Section 3

Launch Sites Proximities Analysis

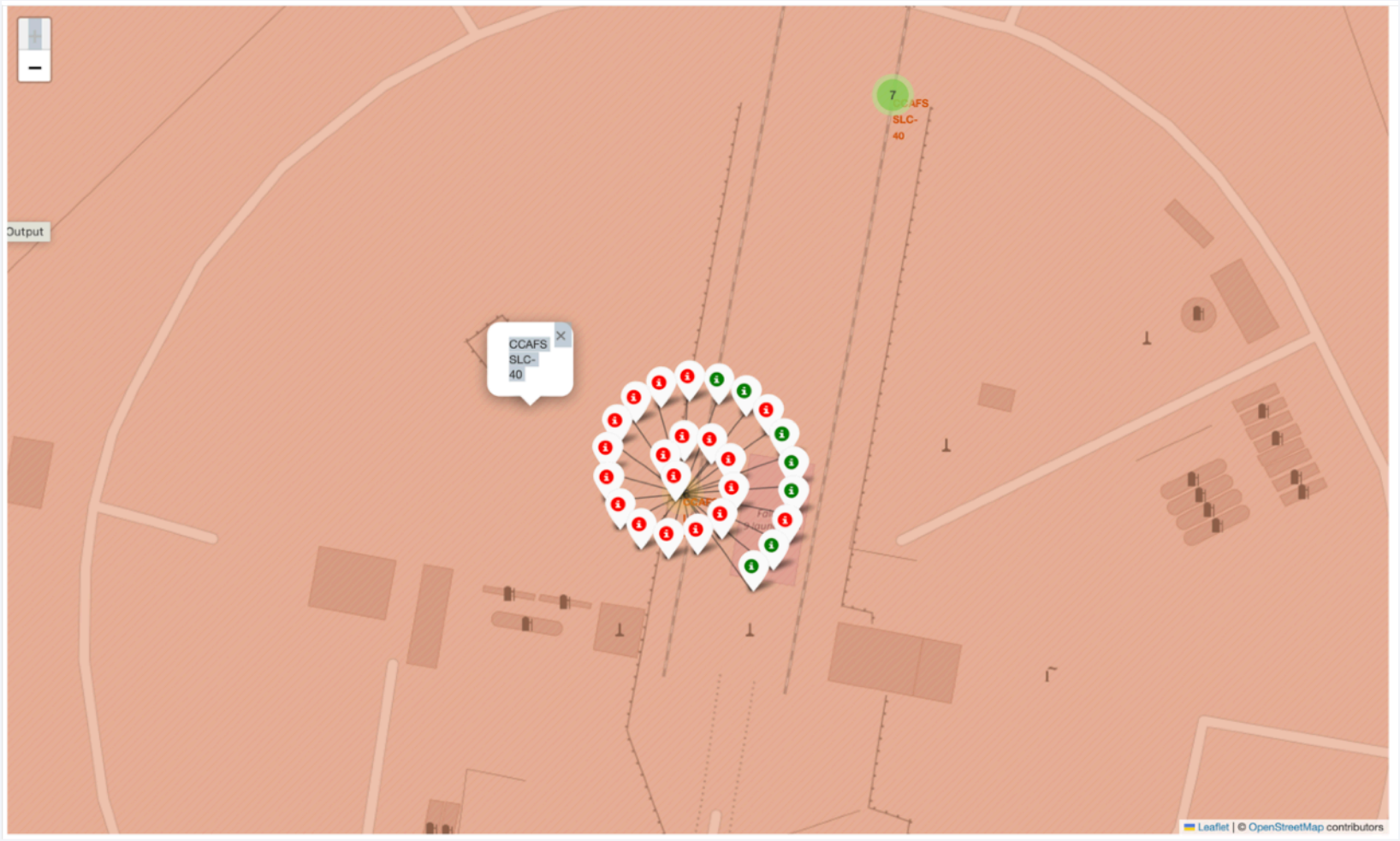
All launch sites' location markers on a global map

- See the two launch sites in the United States of America.



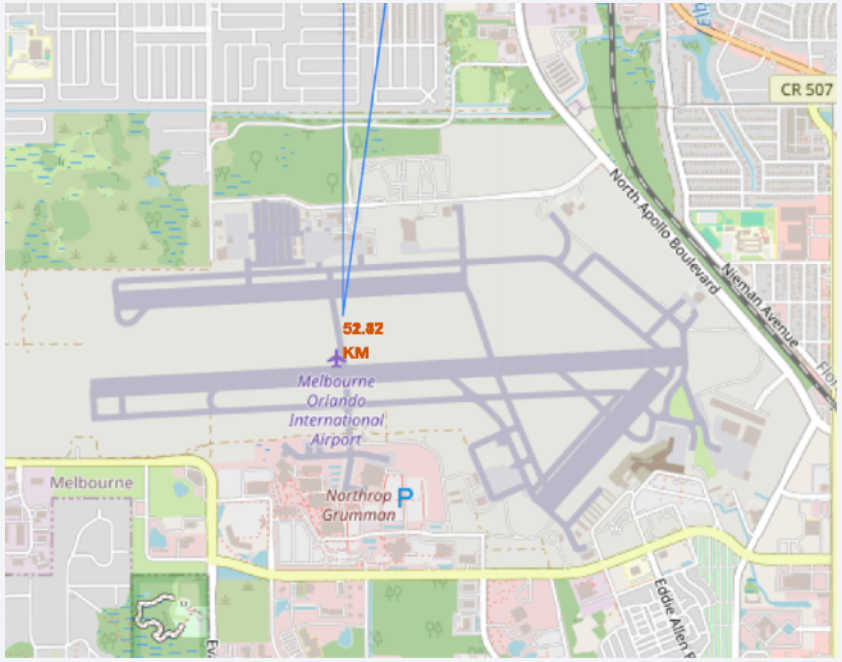
Color-labeled launch outcomes on the map

- Green markers show successful and Reds show failures in CCAFS SCL-40.



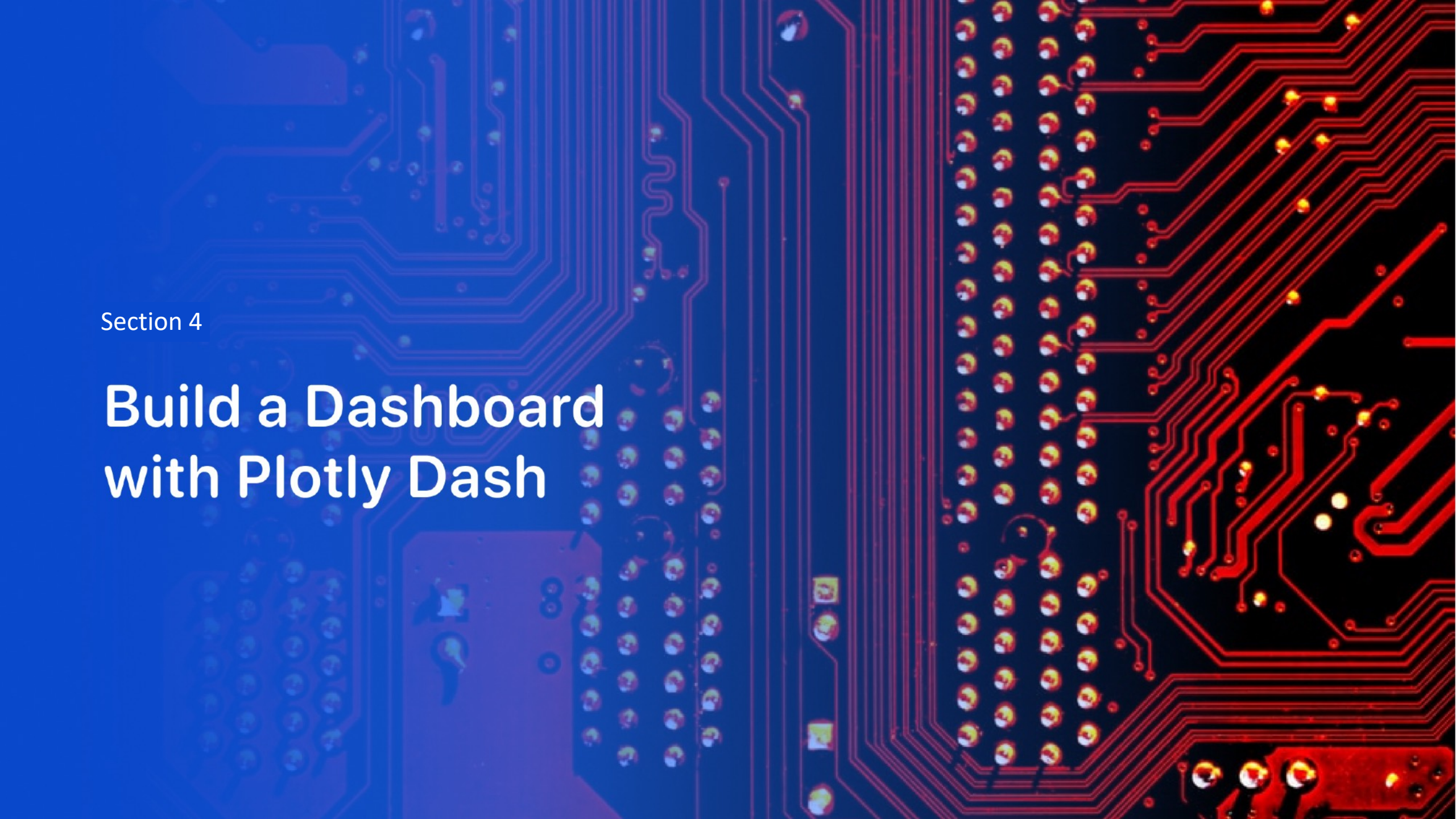
Launch site to its proximities such as railway, highway, coastline with distance

- Railway is closest to KSC LC-39.



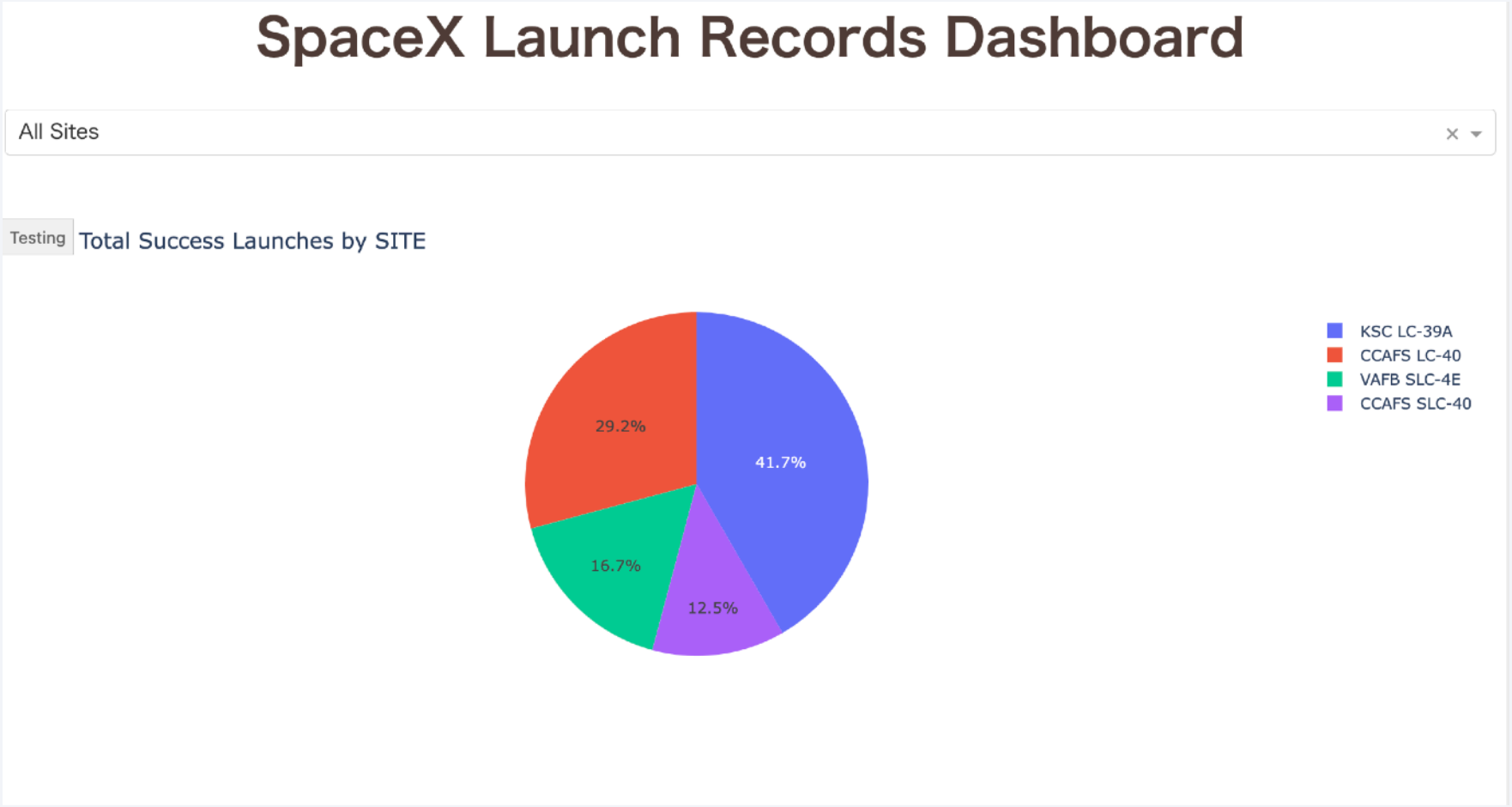
Section 4

Build a Dashboard with Plotly Dash



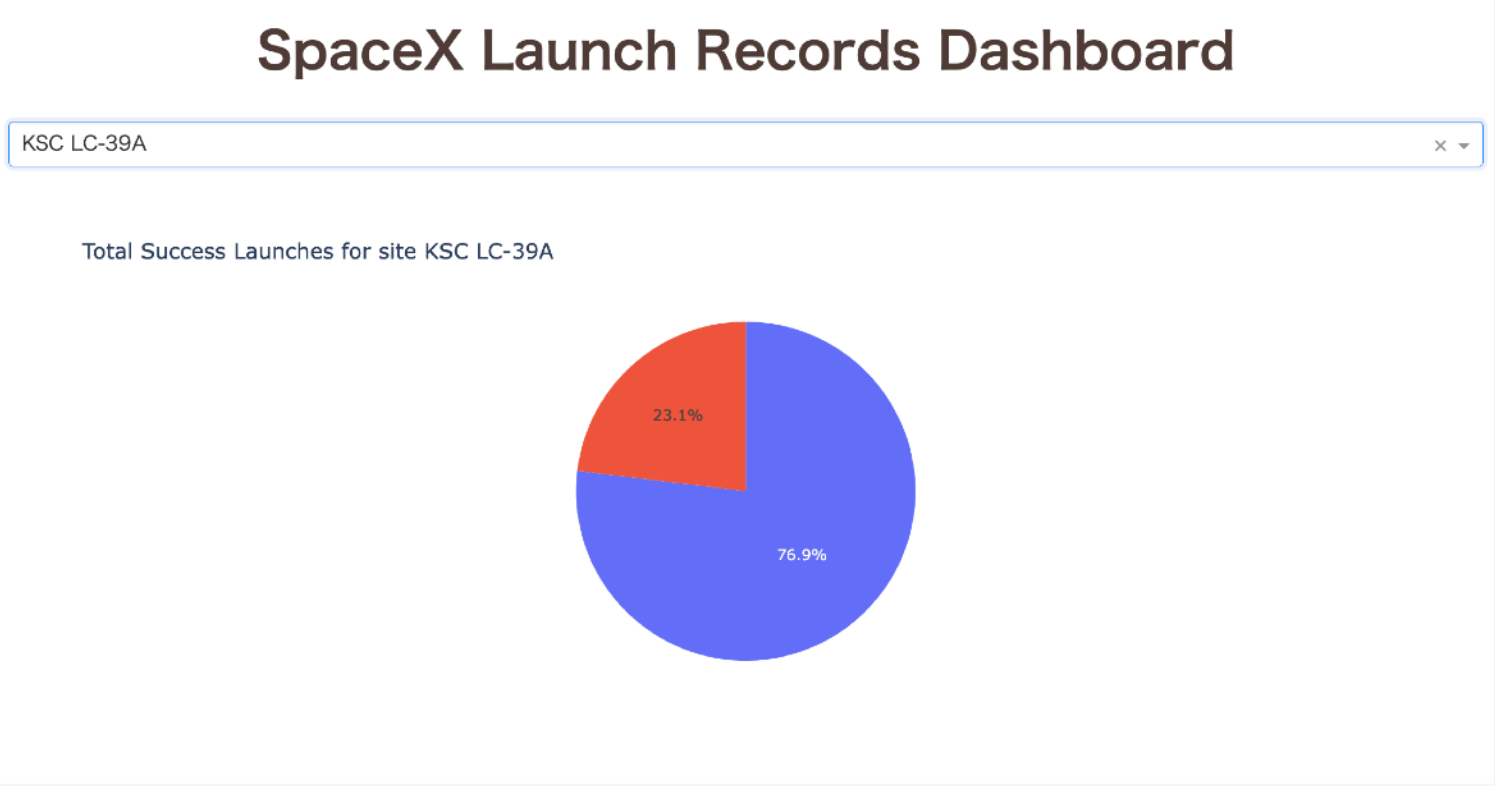
Launch success count for all sites

- KSC LC-39A had most successful launches in all sites.



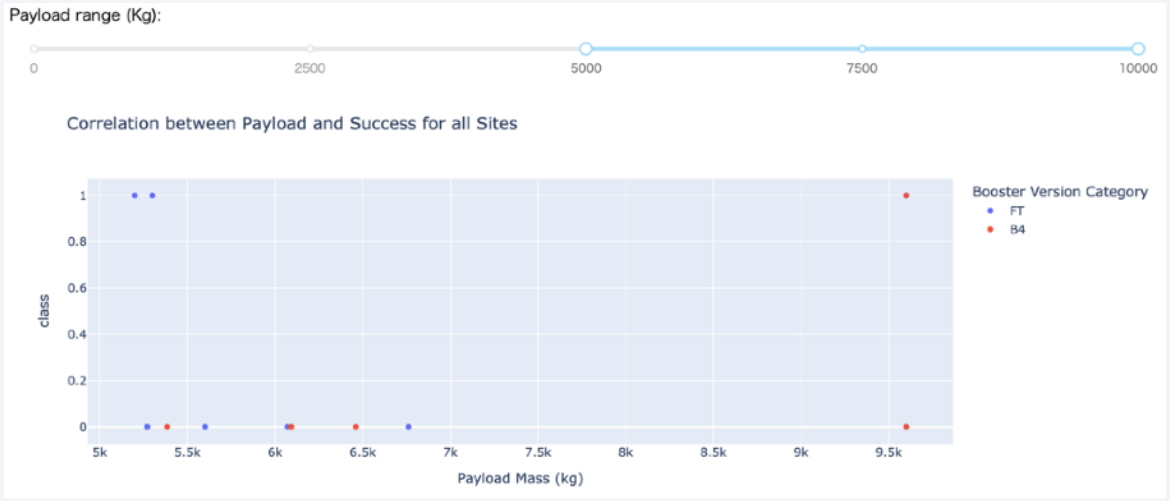
Launch site with highest launch success ratio

- KSC LC-39A had 76.9% success rate.



Payload vs. Launch Outcome for all sites, with different payload selected in the range slider

- Success rate of light Payload Mass is higher than heavy.

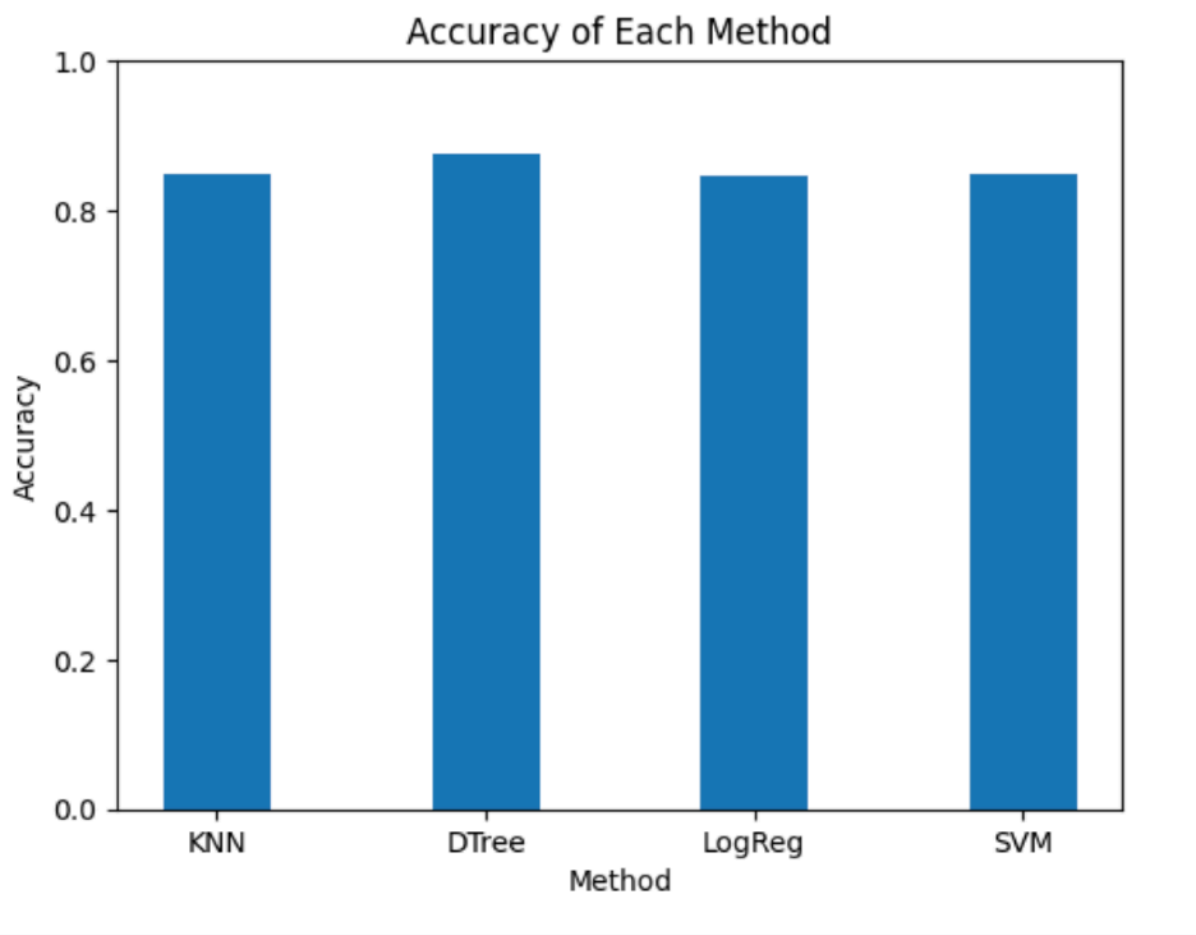


Section 5

Predictive Analysis (Classification)

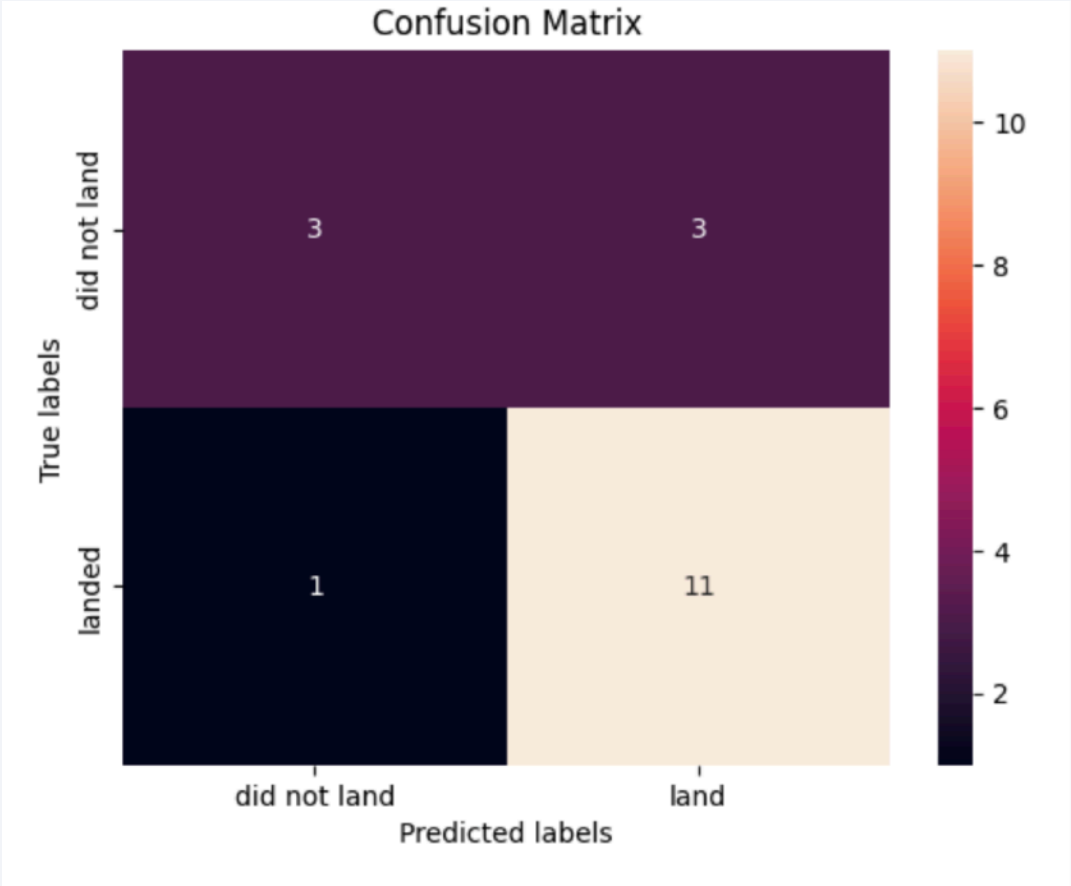
Classification Accuracy

- Decision Tree model has the highest classification accuracy.



Confusion Matrix

- Decision Tree can distinguish between the different classes. And the major problem is false positives.



Conclusions

- The larger the Flight Number, the better the success rate at launch sites.
- Orbit type : ES-L1, GEO, HEO and SSO have the highest success rates.
- The success rate since 2013 kept increasing till 2020.
- KSC LC-39A had most successful launches in all sites. IT had 76.9% success rate.
- Success rate of light Payload Mass is higher than heavy.
- Decision Tree model has the highest classification accuracy.

Thank you!