



IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Adedayo Abiodun Adeboye
14th February 2023



Outline

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

Executive Summary

- **Summary of methodologies**

- API Data Collection

- Web Scraping Data Collection

- Data Wrangling

- Exploratory Data Analysis with SQL

- Exploratory Data Analysis with Data Visualization

- Interactive Visual Analytics with Folium

- Machine Learning Prediction

- **Summary of all results**

- Exploratory/Descriptive Data Analysis

- Interactive visualization analytics

- Creating dashboard for the team

- Predictive Analytics

Introduction

- **Project background and context**

There is an advertisement on Space X's website that Falcon 9 rocket launches cost 62 million dollars, while other providers charge upwards of 165 million dollars per launch. Savings were largely achieved through the reuse of the first stage. It is therefore possible to determine the cost of a launch if we can predict the landing of the first stage. SpaceX may be able to use this information if an alternative company wishes to bid on a rocket launch against it. Our project is designed to construct a machine-learning pipeline that will be able to predict whether the first stage will be successful in a successful landing.

- **Problems you want to find answers**

To determine the price of each launch.

Create dashboard to determine if SpaceX will reuse the first stage.

The factor that determine successful landing

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:

 - Data collection using API and Web Scraping

- Perform data wrangling

 - Exploratory data analysis was performed to categorize outcomes into the training category

- 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

- **Data Collection Methods**

The data was obtained from the SpaceX REST API. A get request was performed using the request library to obtain the launch data

The result was viewed by calling the `.json()` method

The result was in the form of JSON, specifically a list of JSON objects.

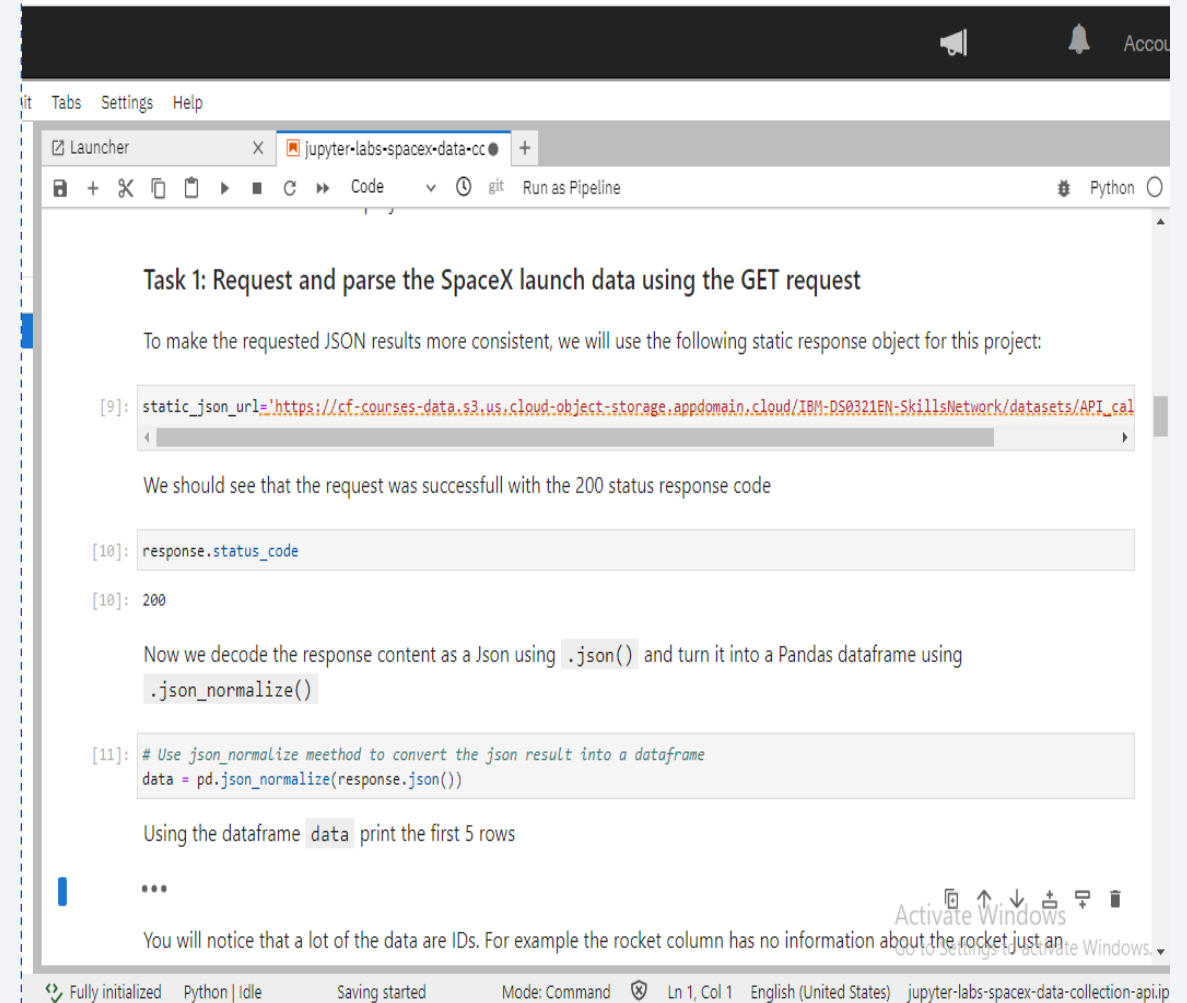
JSON list of Objects then converted to DataFrame by using the `jsom_normalize` function

Web Scraping using Beautiful soap was another method used to scrape some HTML tables that contain valuable Falcon9 Launch records

Data Collection – SpaceX API

- To gather data from SpaceX's API, we used the get request and did some basic data wrangling.
- Here is a link to the first API data collection process:

[https://github.com/Dayorboye/Spaces_X-s_Falcon_9_launch_Project/blob/main/jupyter-labs-spacex-data-collection-api%20\(1\).ipynb](https://github.com/Dayorboye/Spaces_X-s_Falcon_9_launch_Project/blob/main/jupyter-labs-spacex-data-collection-api%20(1).ipynb)



The screenshot shows a Jupyter Notebook window titled "jupyter-labs-spacex-data-cc". The notebook content includes:

- Task 1: Request and parse the SpaceX launch data using the GET request**
- Text: "To make the requested JSON results more consistent, we will use the following static response object for this project:"
- Code cell [9]: `static_json_url = 'https://cf.courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DS0321EN-SkillsNetwork/datasets/API_call'`
- Text: "We should see that the request was successfull with the 200 status response code"
- Code cell [10]: `response.status_code`
- Text: "Now we decode the response content as a Json using `.json()` and turn it into a Pandas dataframe using `.json_normalize()`"
- Code cell [11]:

```
# Use json_normalize meethod to convert the json result into a dataframe
data = pd.json_normalize(response.json())
```
- Text: "Using the dataframe `data` print the first 5 rows"
- Text: "You will notice that a lot of the data are IDs. For example the rocket column has no information about the rocket just an"

The bottom status bar indicates: "Fully initialized Python | Idle Saving started Mode: Command Ln 1, Col 1 English (United States) jupyter-labs-spacex-data-collection-api.ipynb".

Data Collection - Scraping

- A web scraper was used to collect Falcon 9 launch records, and using BeautifulSoup, the table was converted into a pandas dataframe.
- Here is a link to the Web Scraping data collection process:

https://github.com/Dayorboye/Spaces_X-s_Falcon_9_launch_Project/blob/main/jupyter-labs-webscraping.ipynb

```

TASK 1: Request the Falcon9 Launch Wiki page from its URL

First, let's perform an HTTP GET method to request the Falcon9 Launch HTML page, as an HTTP response.

[5]: # use requests.get() method with the provided static_url
# assign the response to a object
data = requests.get(static_url).text

Create a BeautifulSoup object from the HTML response

[6]: # Use BeautifulSoup() to create a BeautifulSoup object from a response text content
soup = BeautifulSoup(data, 'html5lib')

Print the page title to verify if the BeautifulSoup object was created properly

[7]: # Use soup.title attribute
print(soup.title)

<title>List of Falcon 9 and Falcon Heavy launches - Wikipedia</title>

TASK 2: Extract all column/variable names from the HTML table header

Next, we want to collect all relevant column names from the HTML table header

Let's try to find all tables on the wiki page first. If you need to refresh your memory about BeautifulSoup, please check the
external reference link towards the end of this lab

```

Data Wrangling

- Exploratory Data Analysis (EDA) to find some patterns in the data and determine what would be the label for training supervised models which leads to the following finding
 - Number of launches on each site
 - Number and occurrence of each orbit
 - Number and occurrence of mission outcomes per orbit type
 - Create a landing outcome label from the Outcome column
- Here is a link to the Web Scraping data collection process:

https://github.com/Dayorboye/Spaces_X-s_Falcon_9_launch_Project/blob/main/labs-jupyter-spacex-data_wrangling_jupyterlite.jupyterlite.ipynb

EDA with Data Visualization

- We perform exploratory Data Analysis and Feature Engineering using Pandas and Matplotlib to visualize the data:

The FlightNumber (indicating the number of launches attempted) and Payload variables are plotted to observe how they affect the launch outcome.

Observed the relationship between the FlightNumber and LaunchSite, plot the FlightNumber against the LaunchSite using the function catplot.

The relationship between the success rate of each orbit type can be visualized

Developed a visual representation of the relationship between FlightNumber and Orbit type

- Here is a link to data visualization process:

https://github.com/Dayorboye/Spaces_X-s_Falcon_9_launch_Project/blob/main/jupyter-labs-eda-dataviz.ipynb.jupyterlite.ipynb

EDA with SQL

We applied EDA with SQL to get insight into the following queries.

- We queried the names of the unique launch sites in the space mission
- We queried 5 records where launch sites begin with the string 'CCA'
- We queried the total payload mass carried by boosters launched by NASA (CRS)
- We queried average payload mass carried by booster version F9
- List the date when the first successful landing outcome in ground pad was achieved.
- 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
- Rank the count of successful landing_outcomes between the date 04-06-2010 and 20-03-2017 in descending order
- Here is a link to data Queries process:

https://github.com/Dayorboye/Spaces_X-s_Falcon_9_launch_Project/blob/main/jupyter-labs-eda-sql-coursera_sqlite.ipynb

Build a Dashboard with Plotly Dash

- Pie charts were plotted to illustrate the number of launches made by certain sites
- Relationship between Outcome and Payload Mass (Kg) for the different booster versions visualized by plotting Scatter graph
- Here is a link to data dashboard server:
https://github.com/Dayorboye/Spaces_X-s_Falcon_9_launch_Project/commit/b124c1237afa00abcc2eb98f79aae7c5c1bd00b9

Predictive Analysis (Classification)

- Prediction Steps

Necessary Libraries: numpy, pandas as well as seaborn imported to get started

Data collection using dataset url and converted to dataframe

Data collected processed into features (X_train) and target label (Y_train)

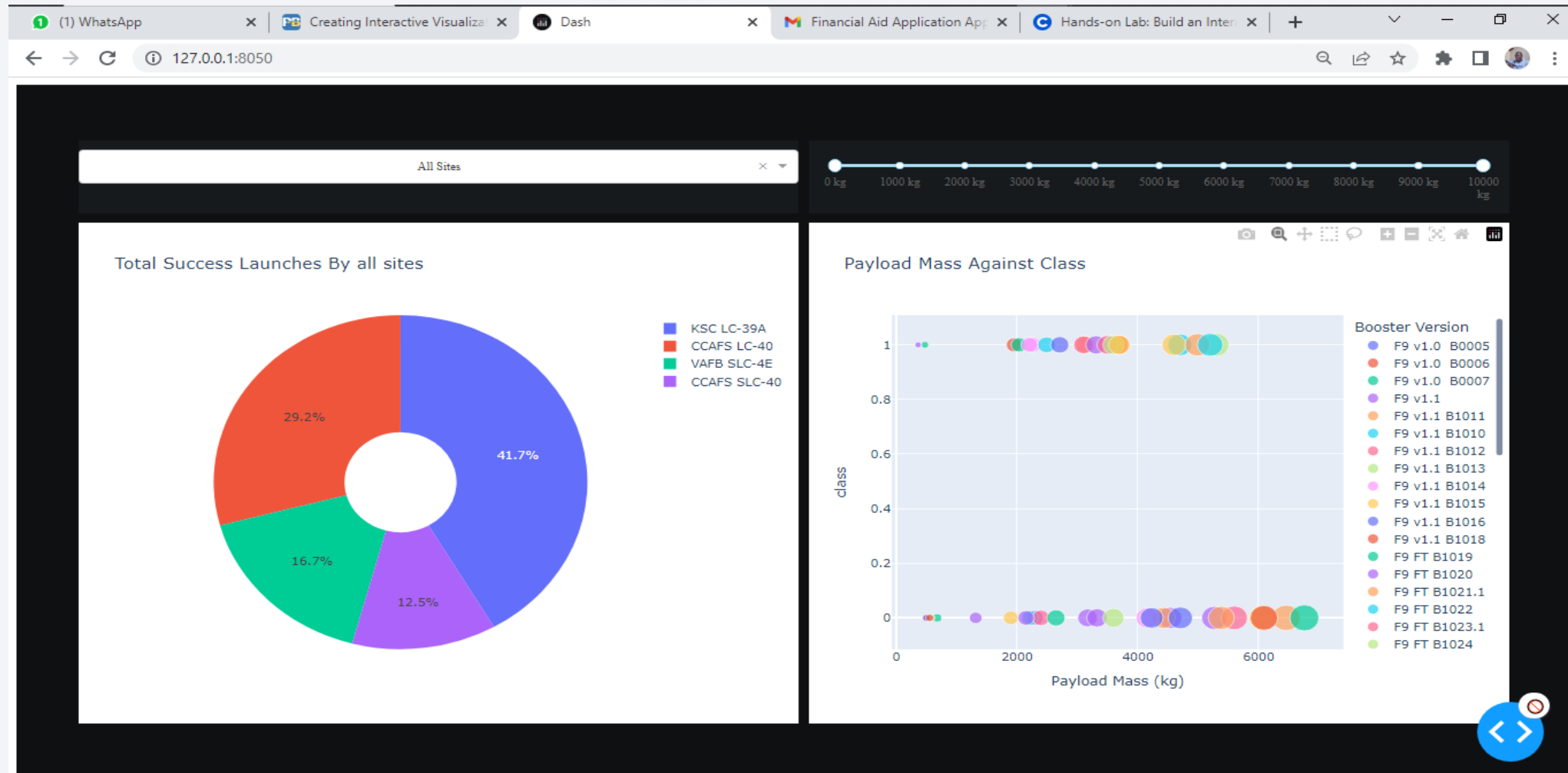
Accuracy model conducted to validate performance of our model

- Here is a link to data dashboard server

https://github.com/Dayorboye/Spaces_X-s_Falcon_9_launch_Project/blob/main/SpaceX_Machine_Learning_Prediction_Part_5.jupyterlite.ipynb

Results

Interactive analytics demo in screenshots



The background of the slide is a complex, abstract composition. It features a dark blue base color. Overlaid on this are numerous diagonal streaks and lines in shades of red and cyan. These lines vary in thickness and opacity, creating a sense of depth and movement. A faint, light blue grid pattern is also visible, particularly in the lower-left quadrant. The overall effect is a high-tech, digital aesthetic.

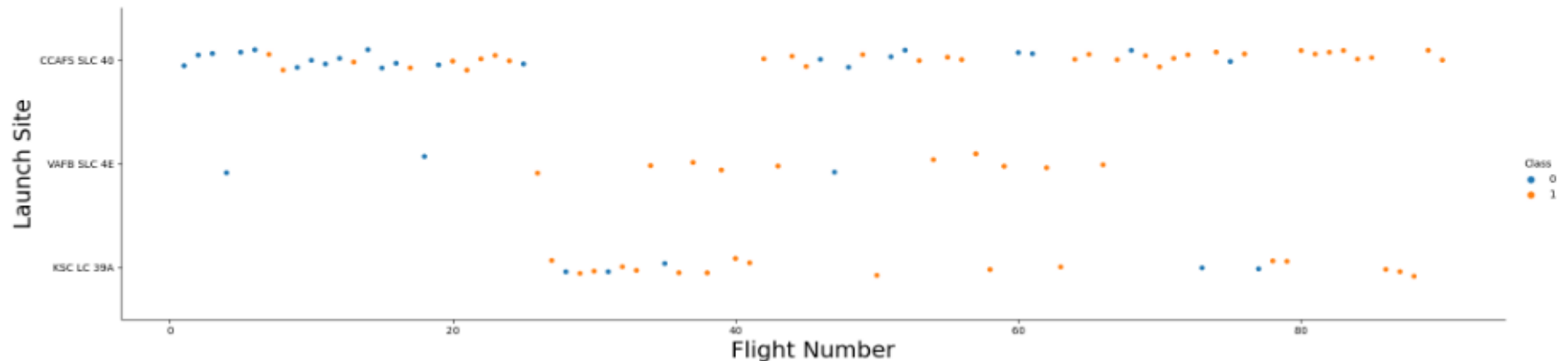
Section 2

Insights drawn from EDA

Flight Number vs. Launch Site

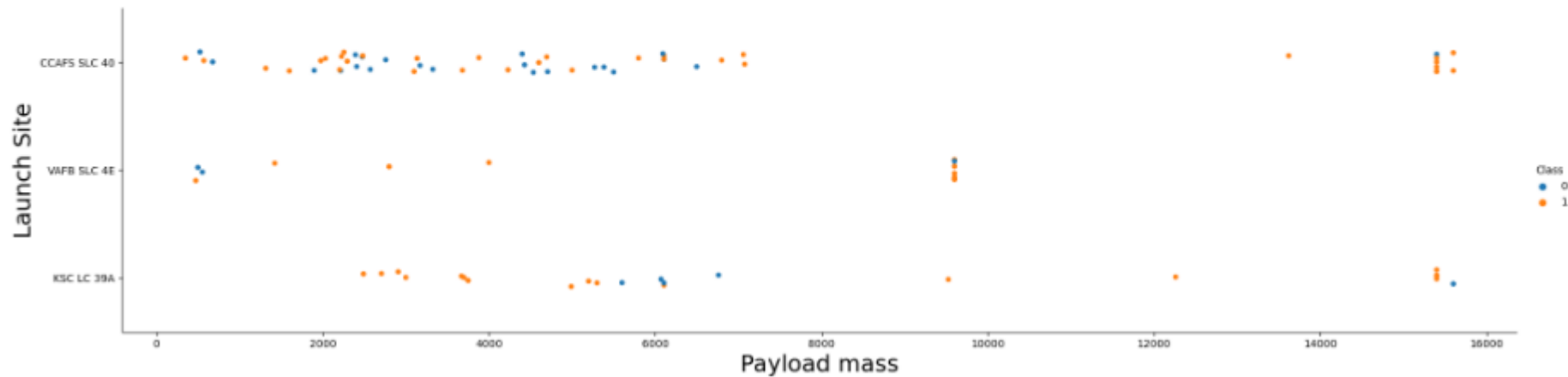
- Launch site have significant impact on the success of the launch site rate

```
[7]: # Plot a scatter point chart with x axis to be Flight Number and y axis to be the launch site, and hue to be the class value
sns.catplot(y="LaunchSite",x="FlightNumber",hue='Class',data=df, aspect=4)
plt.xlabel("Flight Number",fontsize=22)
plt.ylabel("Launch Site",fontsize=22)
plt.show()
```



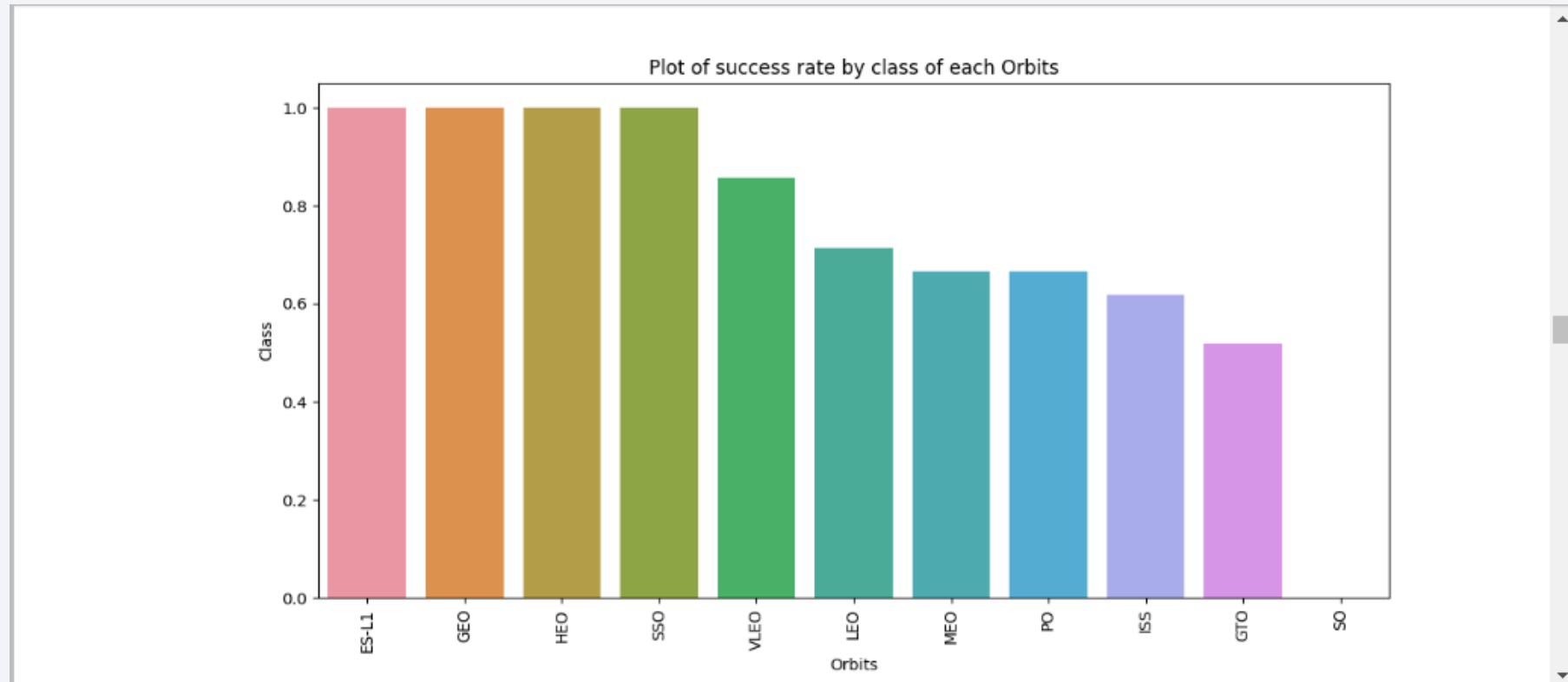
Payload vs. Launch Site

```
[19]: # Plot a scatter point chart with x axis to be PayloadMass and y axis to be the launch site, and hue to be the class value
sns.catplot(y="LaunchSite",x="PayloadMass",hue='Class',data=df, aspect=4)
plt.xlabel("Payload mass",fontsize=22)
plt.ylabel("Launch Site",fontsize=22)
plt.show()
```

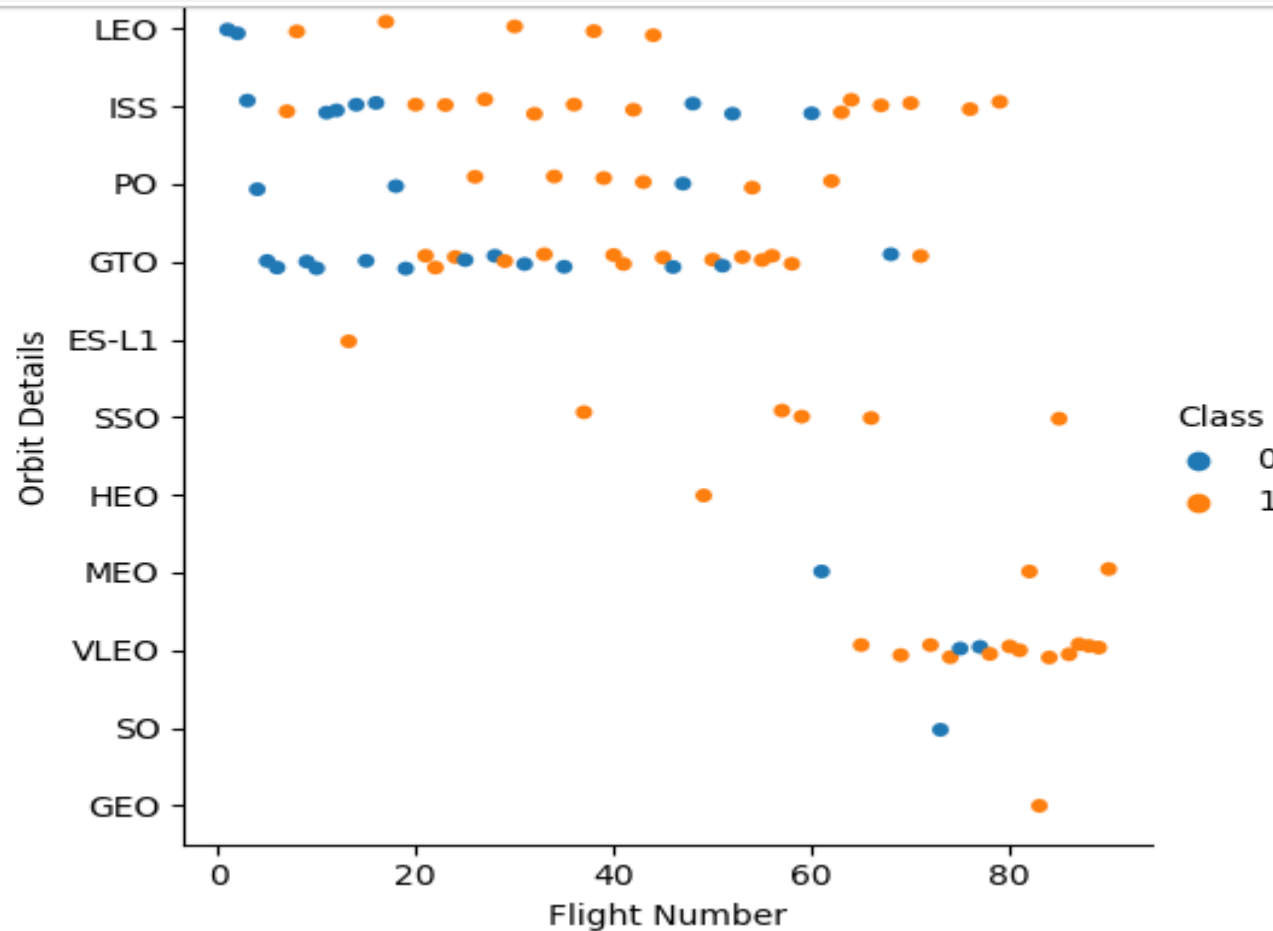


Success Rate vs. Orbit Type

- Respective success rate is observed from below plotted graph

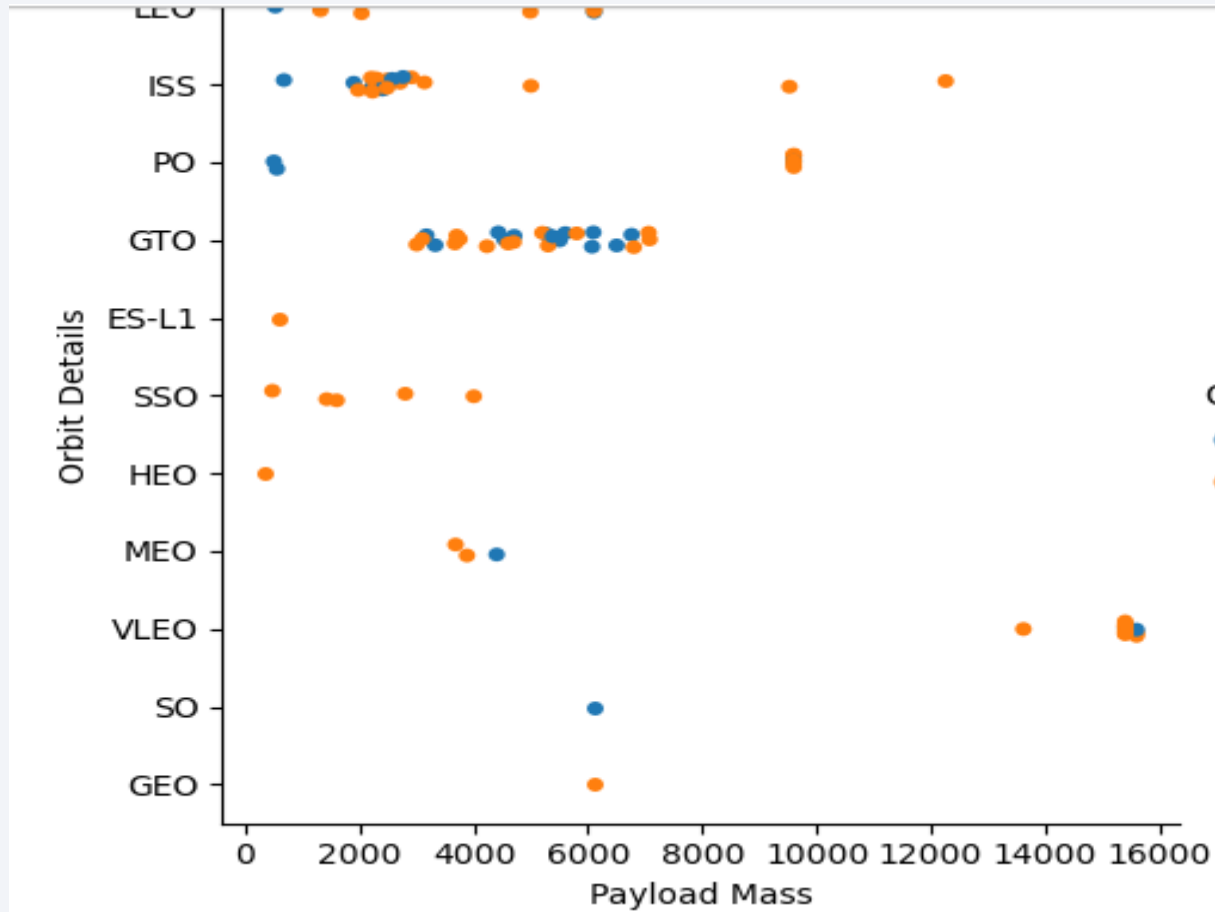


Flight Number vs. Orbit Type



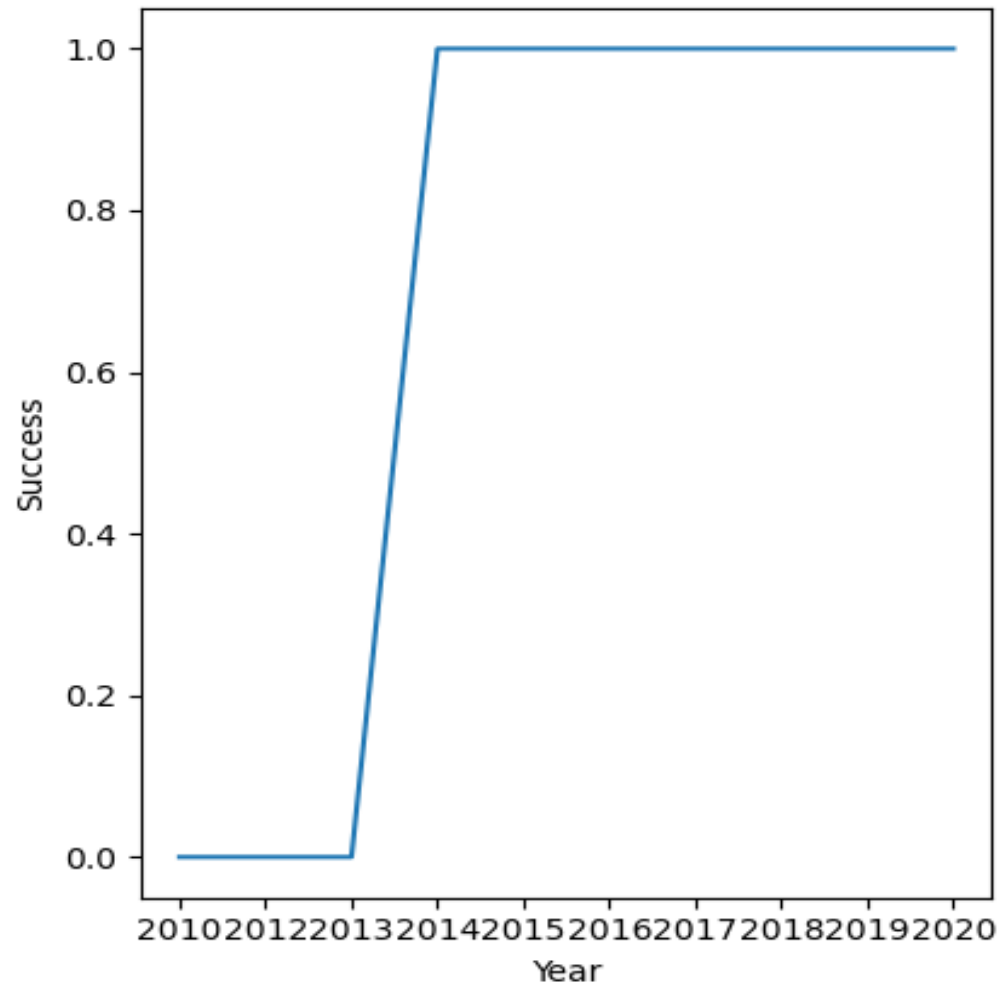
Success have
significant relationship
with number of flights

Payload vs. Orbit Type



We can see here strong relationship between PO, LEO and ISS orbits with payloads mass

Launch Success Yearly Trend



All Launch Site Names

The key word DISTINCT get the request

For the unique sites from the data

(background on this error at: <http://sqlite.org/e/e3q0>)

```
[17]: %sql select DISTINCT LAUNCH_SITE FROM SPACEXTBL
```

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

```
Done.
```

```
[17]: Launch_Site
```

```
CCAFS LC-40
```

```
VAFB SLC-4E
```

```
KSC LC-39A
```

```
CCAFS SLC-40
```


Launch Site Names Begin with 'CCA'

- Use of %sql SELECT LAUNCH_SITE from SPACEXTBL where LAUNCH_SITE LIKE 'CCA%' LIMIT 5 printed 5 record where launch sites begin with 'CCA'

```
Display 5 records where launch sites begin with the string 'CCA'
```

```
[8]: %sql SELECT LAUNCH_SITE from SPACEXTBL where LAUNCH_SITE LIKE 'CCA%' LIMIT 5;
```

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

```
Done.
```

```
[8]: Launch_Site
```

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

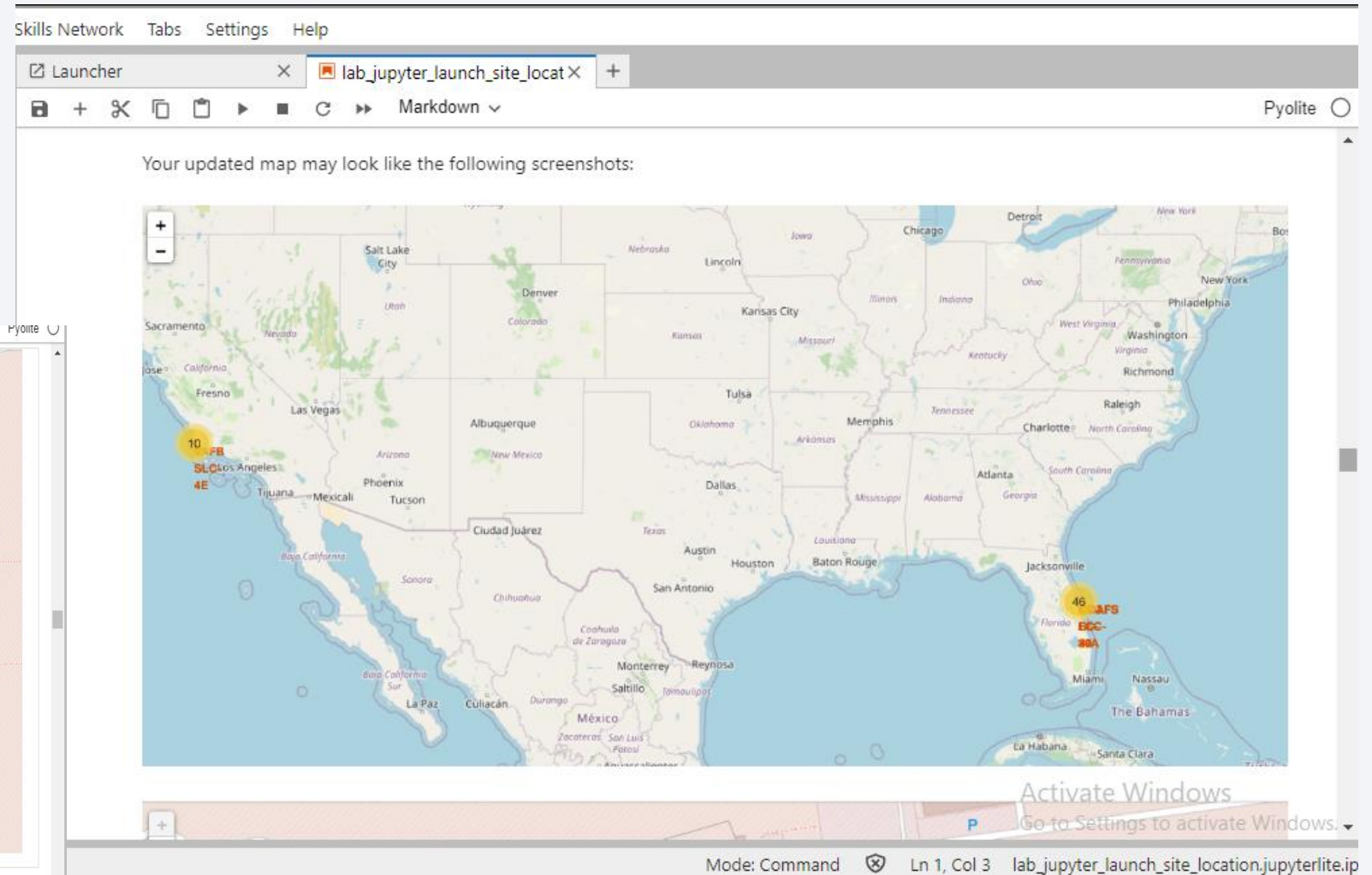
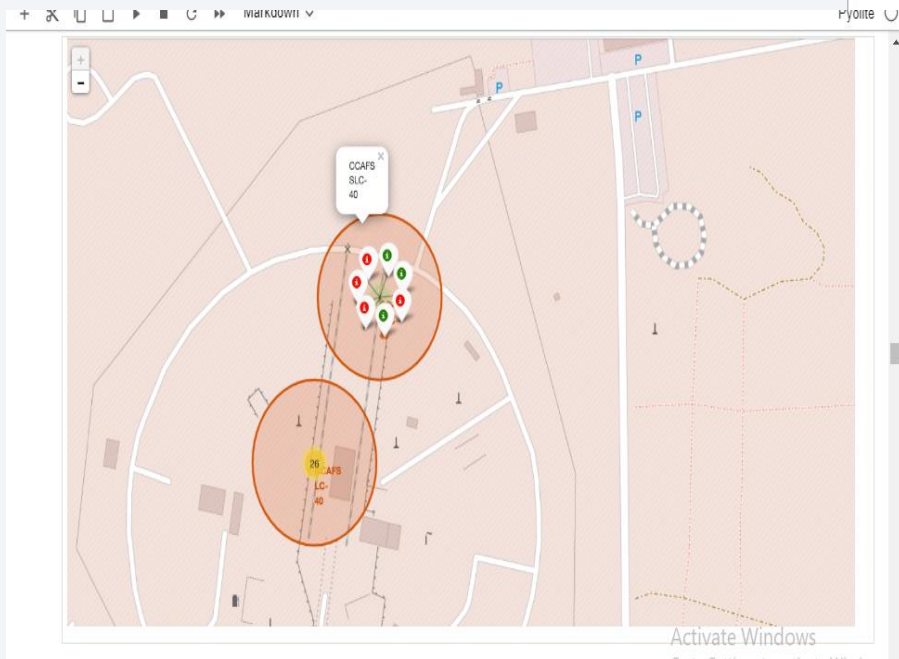
A satellite view of Earth from space, showing the curvature of the planet and city lights at night. The background is a deep blue gradient.

Section 3

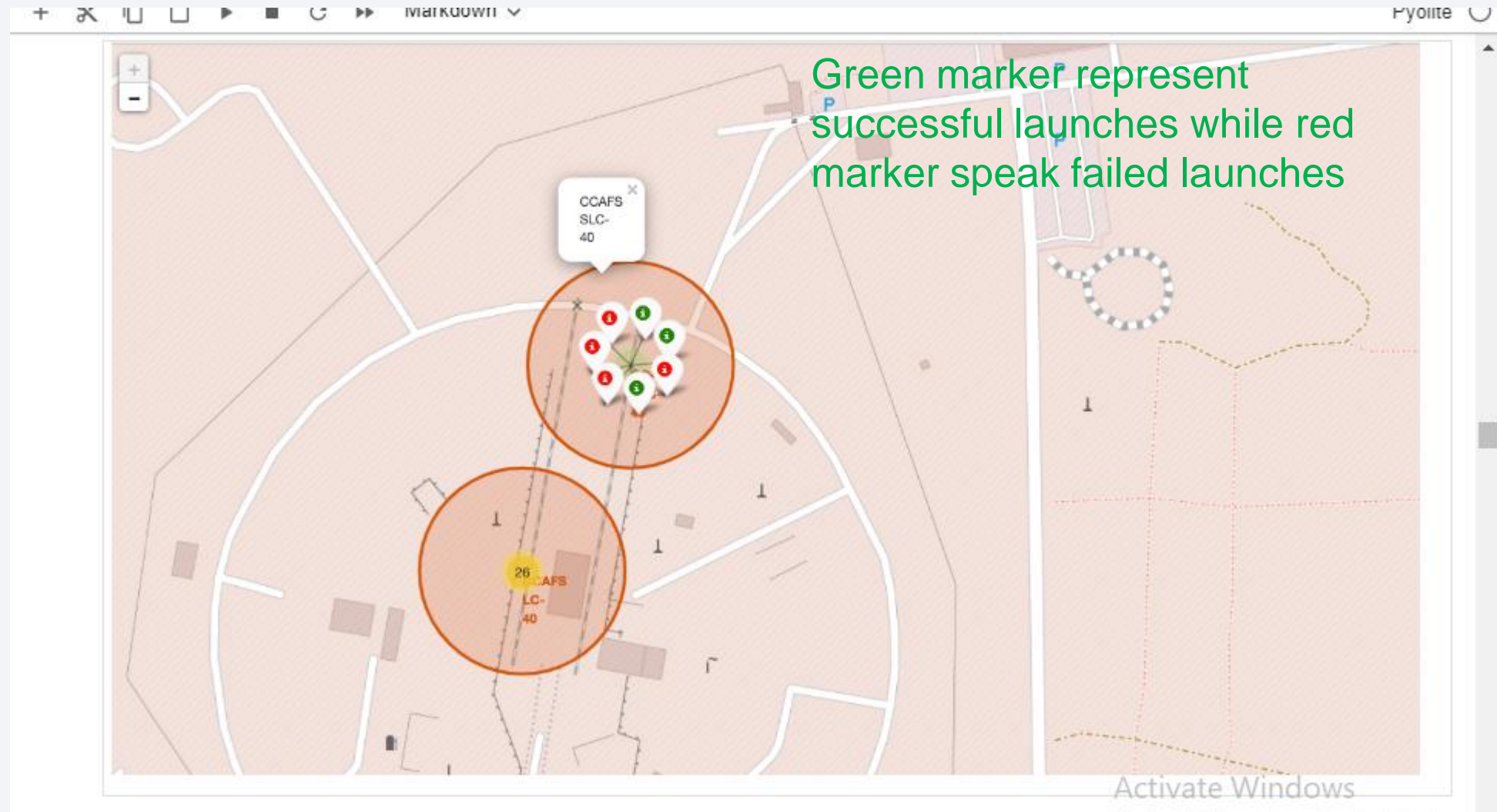
Launch Sites Proximities Analysis

<Folium Map Screenshot 1>

The global map show both successful and failed launch site represented with green marker for success while red marker for failure



Folium Map Identified Success and Failed Sites



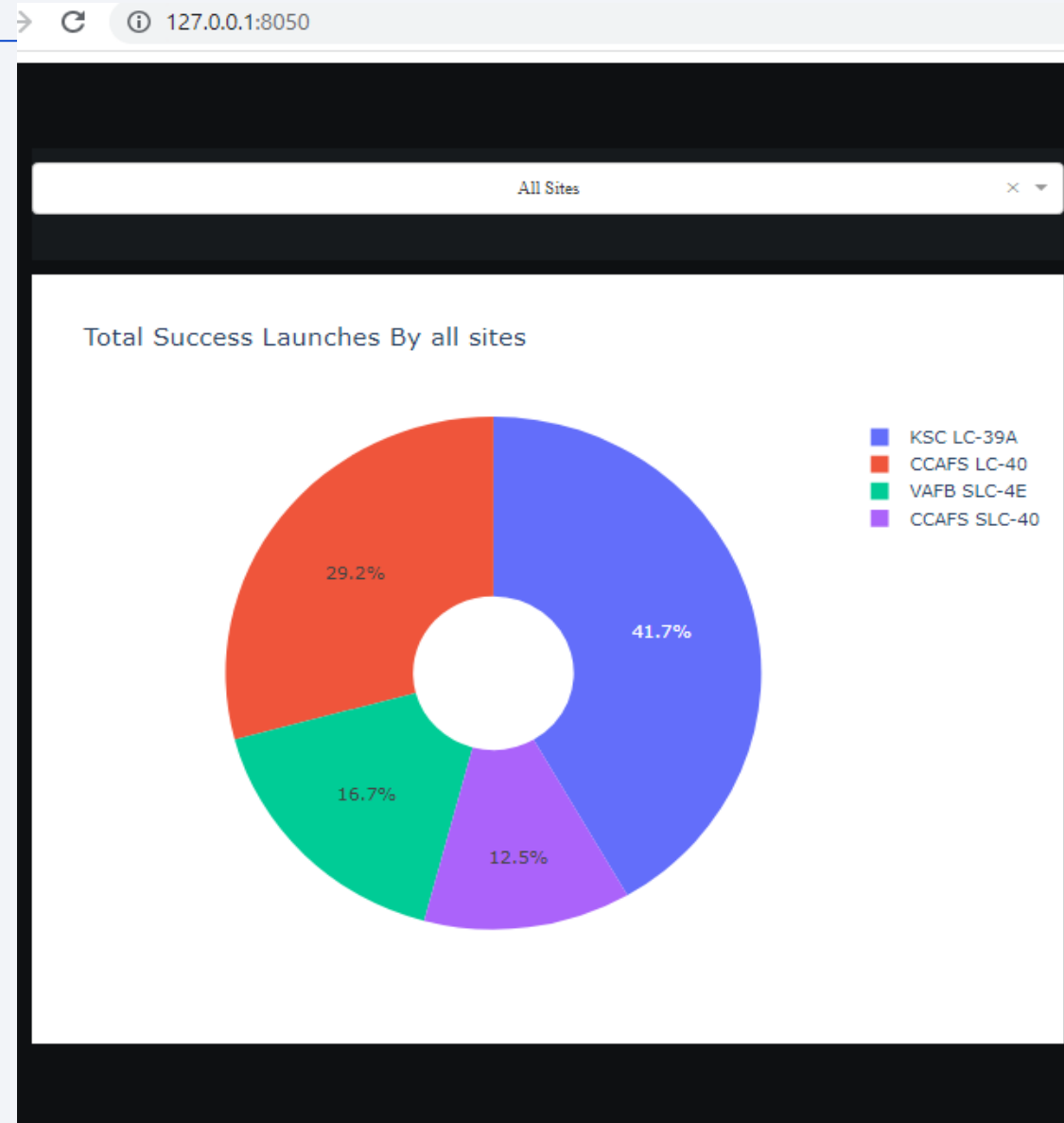


Section 4

Build a Dashboard with Plotly Dash

Pie Chart Visualization Showing % Success

KSC LC-39A record the most successful launches of all the sites



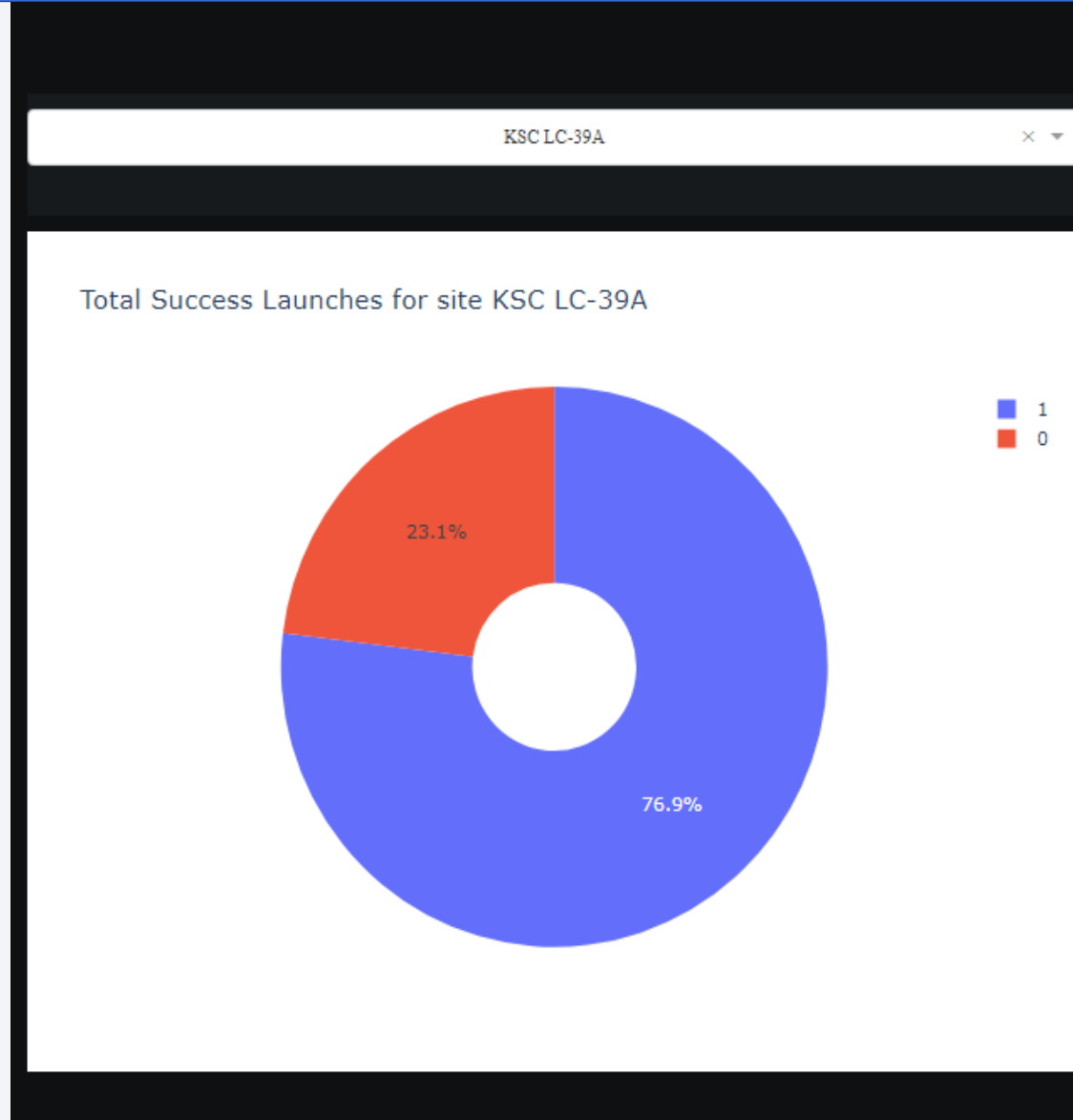
Launch site with highest success rate

KSC LC -39A

achieved a 76.9%

success with 23.1%

failed record



Scatter Plot Shows Relationship Between Payload Mass and Success Rate

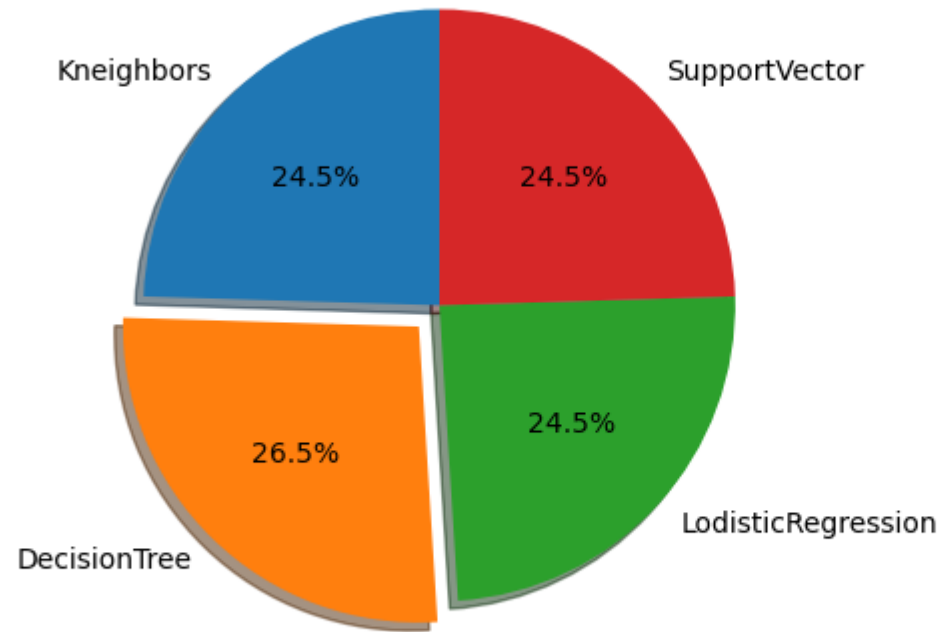
More success recorded when payload mass is lighter



Section 5

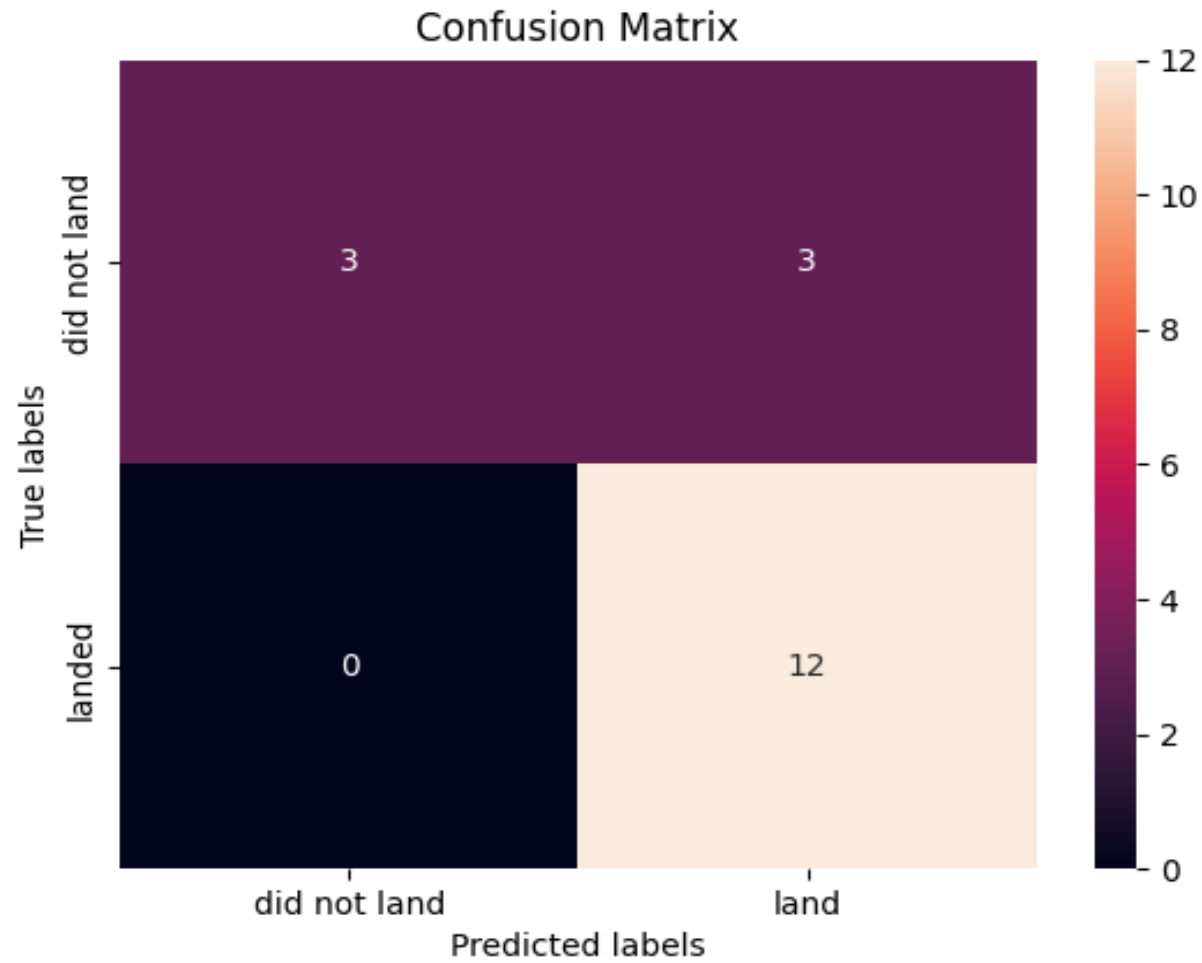
Predictive Analysis (Classification)

Classification Accuracy



The pie chart proved
DecisionTree most
effective model with
accuracy of 26.5%

Confusion Matrix



The

Conclusions

IN CONCLUSION, WE CAN SAY:

The more launches at a launch site, the higher the success rate will be

When the payload mass is lighter, more success is recorded

KSC LC-39A recorded the most successful launches of all sites

Launch success rate started to increase in 2013 till 2020.

Orbits ES-L1, GEO, HEO, SSO, VLEO had the highest success rate.

In this scenario, the decision tree classifier is the most effective machine-learning algorithm.

Thank you!

