IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Afonso Diela
15/09/2022

Outline

- Executive Summary
- Introduction
- Methodology
- Results
- Conclusion

Executive Summary

- Summary of methodologies
  - Business understanding and Analytic approach
  - Data collection and preparation with SpaceX API and Web Scrapping
  - Data undertanding using EDA, data cleansing, wrangling and interactive visualization
  - Modeling & Evaluation
- Summary of all results
  - Exploratory Data Analysis outcomes
  - Interactive visual analytics and dashboard
  - Predictive Analysis (Classification)
  - Stakeholders presentation

Introduction

- Project background and context
  - As a Data Scientist in SPACE Y (a new spaceship company) my job is to study the successful landing outcome of a Space X Falcon 9 rocket and to estimate the **cost** of each launch
- Problems you want to find answers
  - Understand how Space X Falcon 9 works through data
  - Predict if the Space X Falcon 9 first stage will land successfully
  - Determine the cost of a launch
  - Which the information to help Business function to build a innovation strategy for SPACE Y to compete against SpaceX

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Data was collected using :
    - Space X API : https://api.spacexdata.com/v4/launches/past
    - Web Scraping : https://en.wikipedia.org/w/index.php?title=List_of_Falcon_9_and_Falcon_Heavy_launches&oldid=1027686922
- Perform data wrangling
  - Data was processed by exploratory Data Analysis to determine Training Labels
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash

Methodology

Executive Summary

- Perform predictive analysis using classification models
  - Split the data into training testing data
  - Train different classification models (SVM, Classification Trees, and Logistic Regression)
  - Find the best method using hyperparameter grid search
  - Modeling & Evaluation

Data Collection

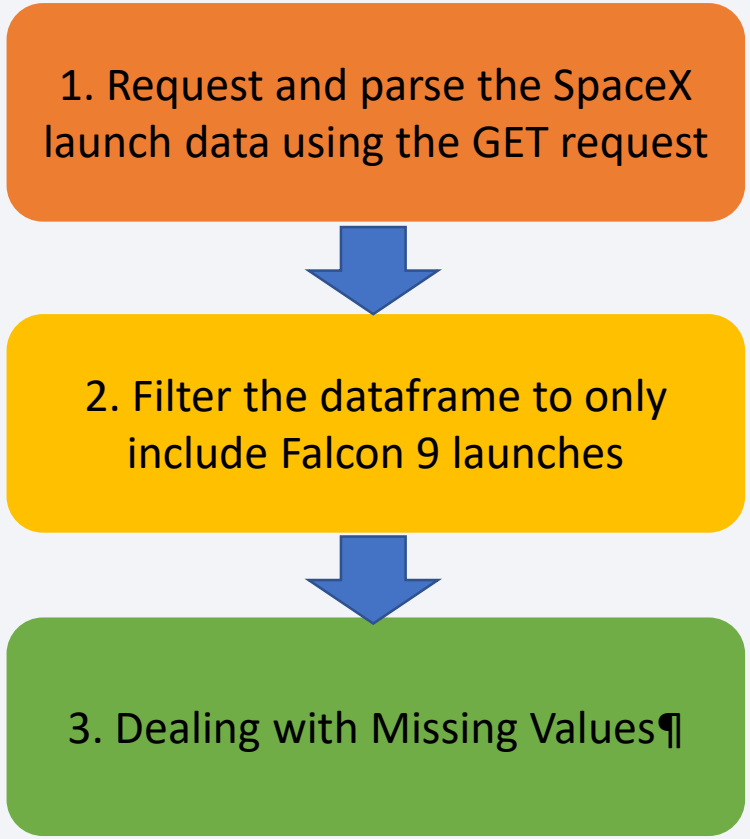
- Describe how data sets were collected.
  - Data Collection using Space X API : https://api.spacexdata.com/v4/rockets/
    - Request and parse the SpaceX launch data using the GET request
    - Filter the dataframe to only include Falcon 9 launches
    - Data wrangling : Dealing with Missing Values¶
  - Data Collection using Web Scraping : https://en.wikipedia.org/w/index.php?title=List_of_Falcon_9_and_Falcon_Heavy_launches&oldid=1027686922
    - Request the Falcon9 Launch Wiki page from its URL
    - Extract all column/variable names from the HTML table header
    - Create a data frame by parsing the launch HTML tables

Data Collection – SpaceX API

- Data is requested from url and parsed as DataFrame
- We filter the data to have only Falcon9 launches
- We perform data cleansing and formatting

- GitHub notebook URL :

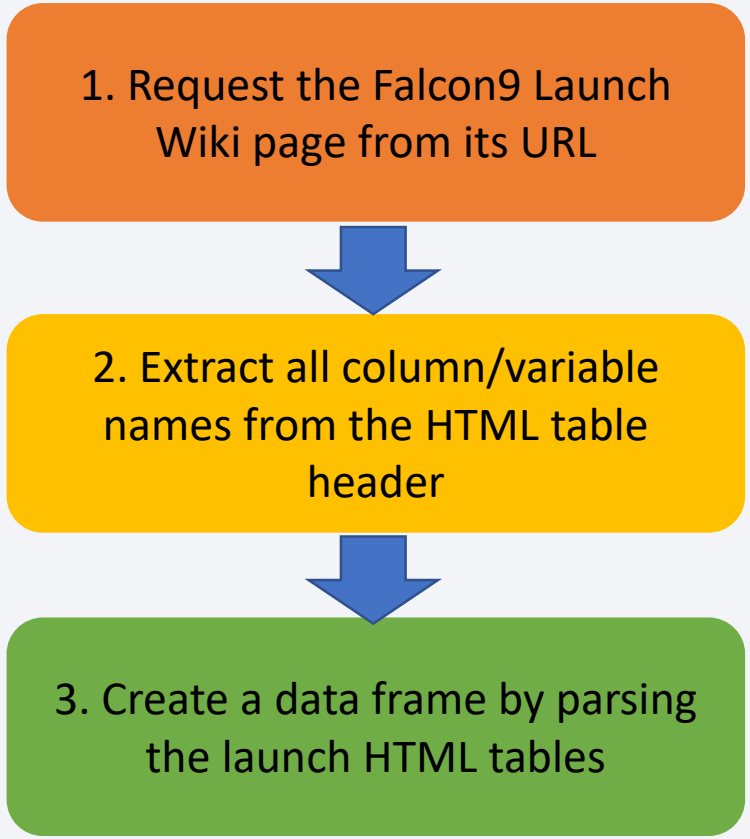
https://github.com/afondiel/ibm-data-science-capstone-project/blob/main/jupyter-labs-spacex-data-collection-api.ipynb



Data Collection - Scraping

- Data is requested from Wikipedia link
- We extract all columnnames from the HTML table header using Beautifulsoup library
- Create a data frame by parsing the launch HTML tables
- GitHub note URL :

https://github.com/afondiel/ibm-data-science-capstone-project/blob/main/jupyter-labs-webscraping-last.ipynb

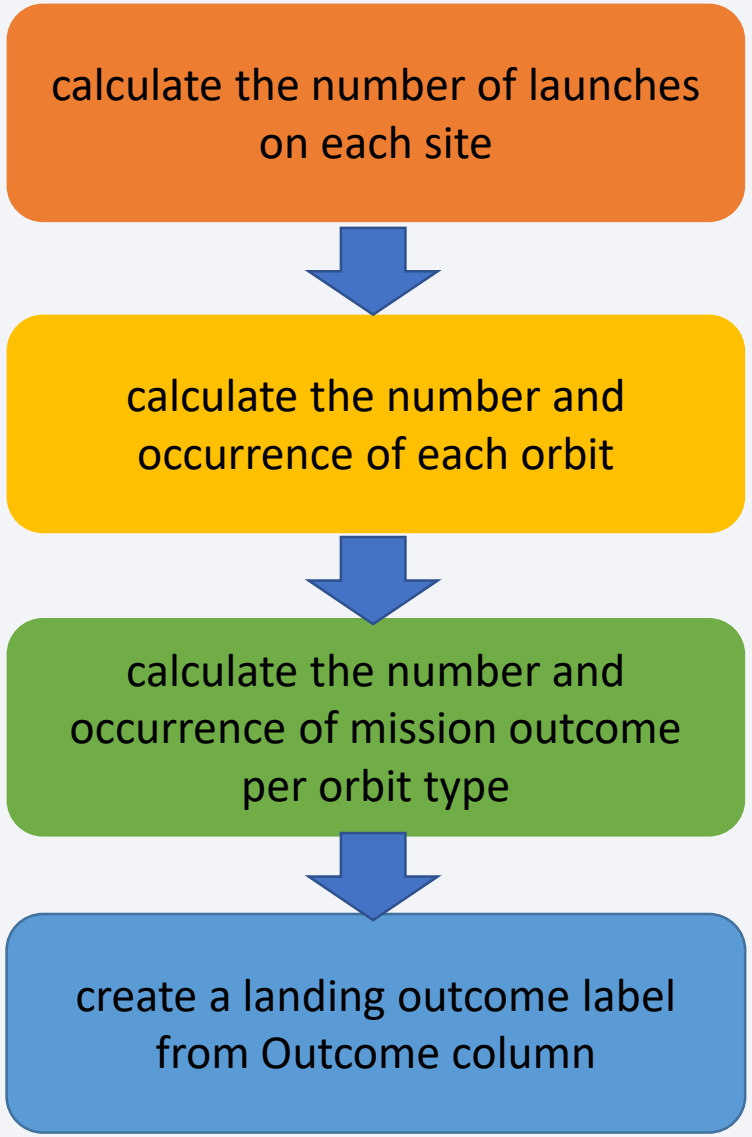


Data Wrangling

- We calculate the number of launches on each site
- We calculate the number and occurrence of each orbit
- We calculate the number and occurrence of mission outcome per orbit type
- We create a landing outcome label from Outcome column

- GitHub URL source :

https://github.com/afondiel/ibm-data-science-capstone-project/blob/main/labs-jupyter-spacex-Data%20wrangling.ipynb



EDA with SQL

- The SQL queries you performed

```
%sql SELECT DISTINCT(Launch_Site) FROM SPACEXTBL
```

```
%sql SELECT Launch_Site FROM SPACEXTBL WHERE Launch_Site LIKE "CCA%" LIMIT 5
```

```
%sql SELECT Customer, SUM(PAYLOAD_MASS__KG_) AS TOTAL_PAYLOAD_MASS_KG FROM SPACEXTBL WHERE Customer = 'NASA (CRS)'
```

```
%sql SELECT Booster_Version, AVG(PAYLOAD_MASS__KG_) AS AVG_PAYLOAD_MASS_KG FROM SPACEXTBL WHERE Booster_Version = 'F9 v1.1'
```

```
%sql SELECT "Date","Landing _Outcome", MIN("Date") FROM SPACEXTBL WHERE "Landing _Outcome" LIKE "%ground pad%"
```

```
%sql SELECT Booster_Version, PAYLOAD_MASS__KG_, "Landing _Outcome" FROM SPACEXTBL WHERE "Landing _Outcome" LIKE "%success%drone ship%" AND (PAYLOAD_MASS__KG_
BETWEEN 4000 AND 6000)
```

```
%sql SELECT Mission_Outcome, COUNT(Mission_Outcome) AS TOTAL_SUCCESS_FAILURE FROM SPACEXTBL GROUP BY Mission_Outcome
```

```
%sql SELECT Booster_Version, PAYLOAD_MASS__KG_ FROM SPACEXTBL WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTBL)
```

```
%sql SELECT "Date", substr("Date", 4, 2), "Landing _Outcome", Booster_Version, Launch_Site FROM SPACEXTBL WHERE "Landing _Outcome" LIKE "%failure%" AND substr(Date,7,4)='2015'
```

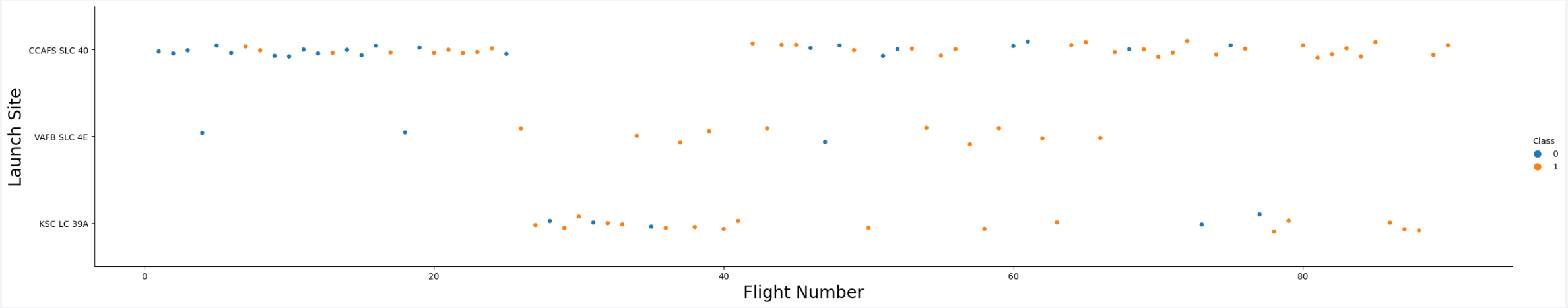
```
%sql SELECT "Date", "Landing _Outcome", COUNT("Landing _Outcome") AS SLO FROM SPACEXTBL  WHERE "Landing _Outcome" LIKE "%Success%" GROUP BY ("Date" BETWEEN '04-06-2010'
AND '20-03-2017') ORDER BY SLO DESC
```

- GitHub URL of completed EDA with SQL notebook :

https://github.com/afondiel/ibm-data-science-capstone-project/blob/main/jupyter-labs-eda-sql-coursera_sqllite.ipynb

EDA with Data Visualization

- We perform EDA to evaluate the relationship between the different features which behavior would affect the launch outcome
- We use scatter plot & barplot to display the results



EDA with Data Visualization

- Bar Chart : we can observe

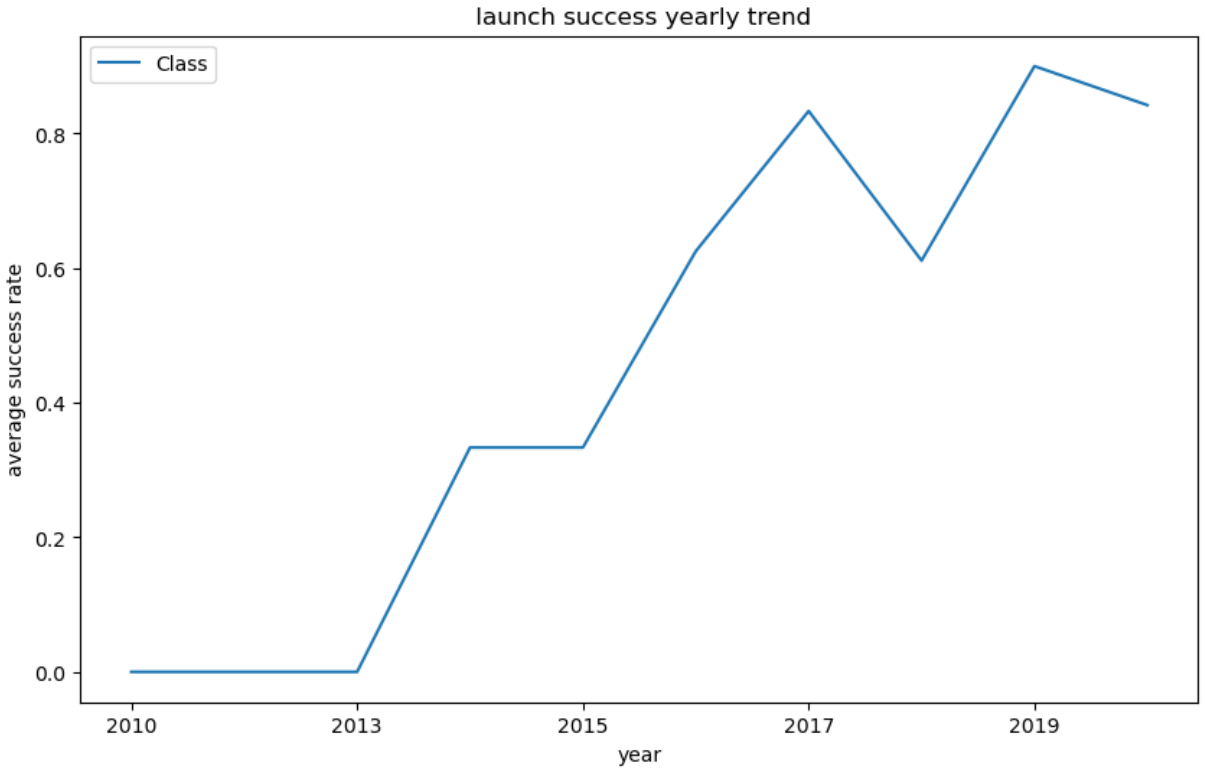
that the success rate since 2013

kept Increasing till 2020

- Finally, we perform one-hot encoding

To format the features that will be used

in success prediction



- GitHub URL of EDA with data visualization notebook :

https://github.com/afondiel/ibm-data-science-capstone-project/blob/main/jupyter-labs-eda-dataviz.ipynb

Build an Interactive Map with Folium

1. Mark all launch sites on a map
   - Use **Folium.Circle()** to create a circle for each launch site on the the map based on its latitude and longitude
   - Use **Folium.map.Marker()** to create a marker for each launch site on the the map on its latitude and longitude
2. Mark the success/failed launches for each site on the map
   - Here we use **MarkerCluster()** to simply map containing many markers having the same coordinate the we label them as green/red if success/failed
3. Calculate the distances between a launch site to its proximities
   - In this section we use **MousePosition** to get the coordinate on the map
   - **folium.PolyLine()** to draw a line between the the marker to the launch site, city, railway, highway

- GitHub URL of your completed interactive map with Folium map :
- https://github.com/afondiel/ibm-data-science-capstone-project/blob/main/lab_jupyter_launch_site_location.ipynb

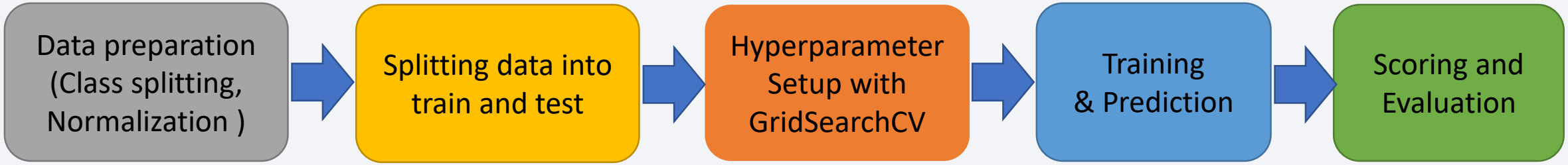
Build a Dashboard with Plotly Dash

- Plots/graphs and interactions added to a dashboard :
  - Pie chart Callback function to show the total successful launches count for all sites
    - **Input** : All/sites or a specific launch site
    - **Output** : success/failed launch outcome
  - Scatter chart Callback function to show the correlation between payload and launch success
    - **inputs** : All/sites or a specific launch Site and Payload
    - **Output** : success/failed launch outcome
  - Slider object to select a playload
- GitHub URL of the completed Plotly Dash lab:

https://github.com/afondiel/ibm-data-science-capstone-project/blob/main/dash/spacex_dash_app.py

Predictive Analysis (Classification)

- The best way to perform classification model :
- Use Hyperparameter to find the best fitting parameter
- Try different classification algorithms : Logistic Regression, SVM, KNN and Decision Tree
- Model development process :

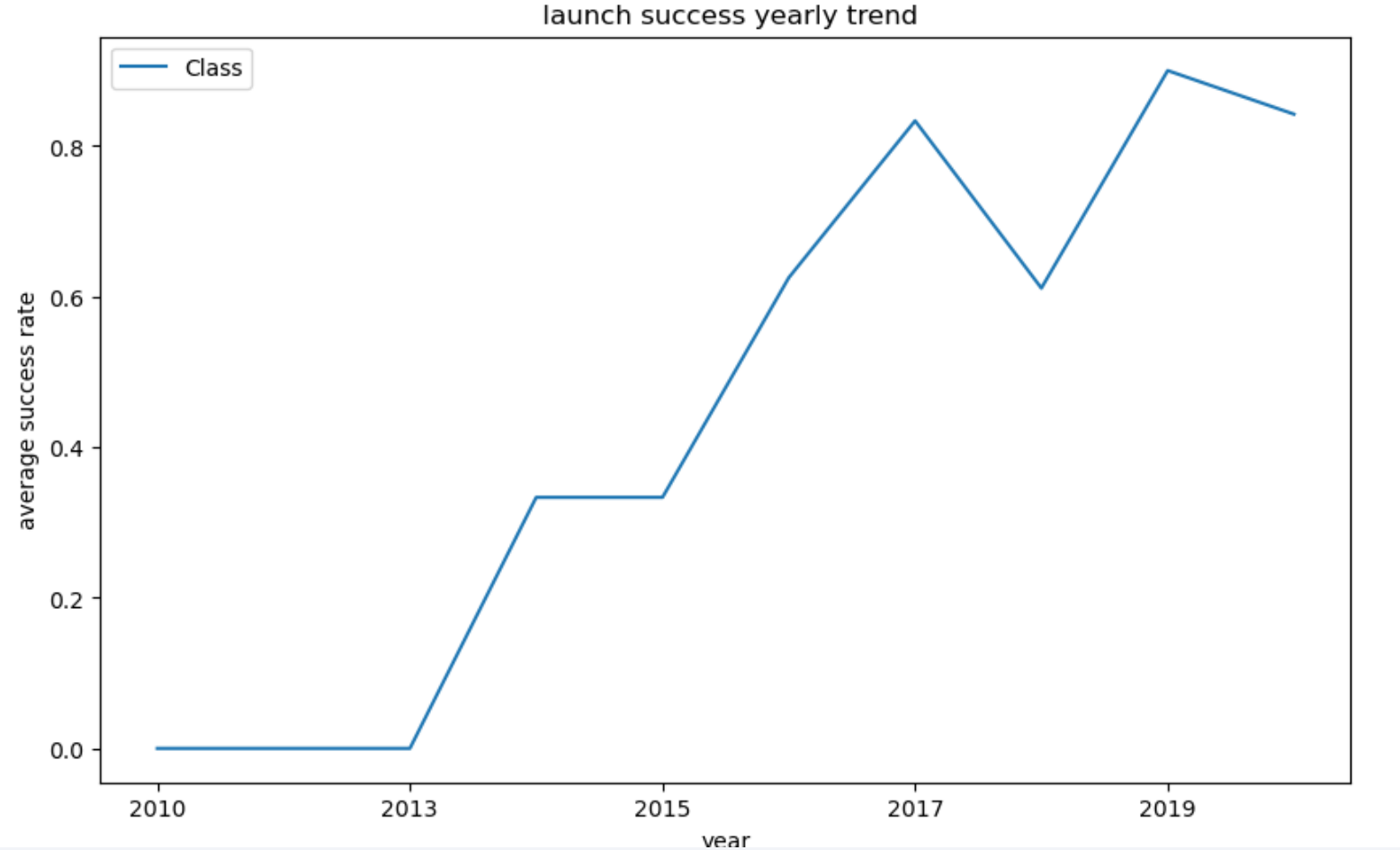


- GitHub URL of predictive analysis lab:

https://github.com/afondiel/ibm-data-science-capstone-project/blob/main/SpaceX_Machine%20Learning%20Prediction_Part_5.ipynb

Results

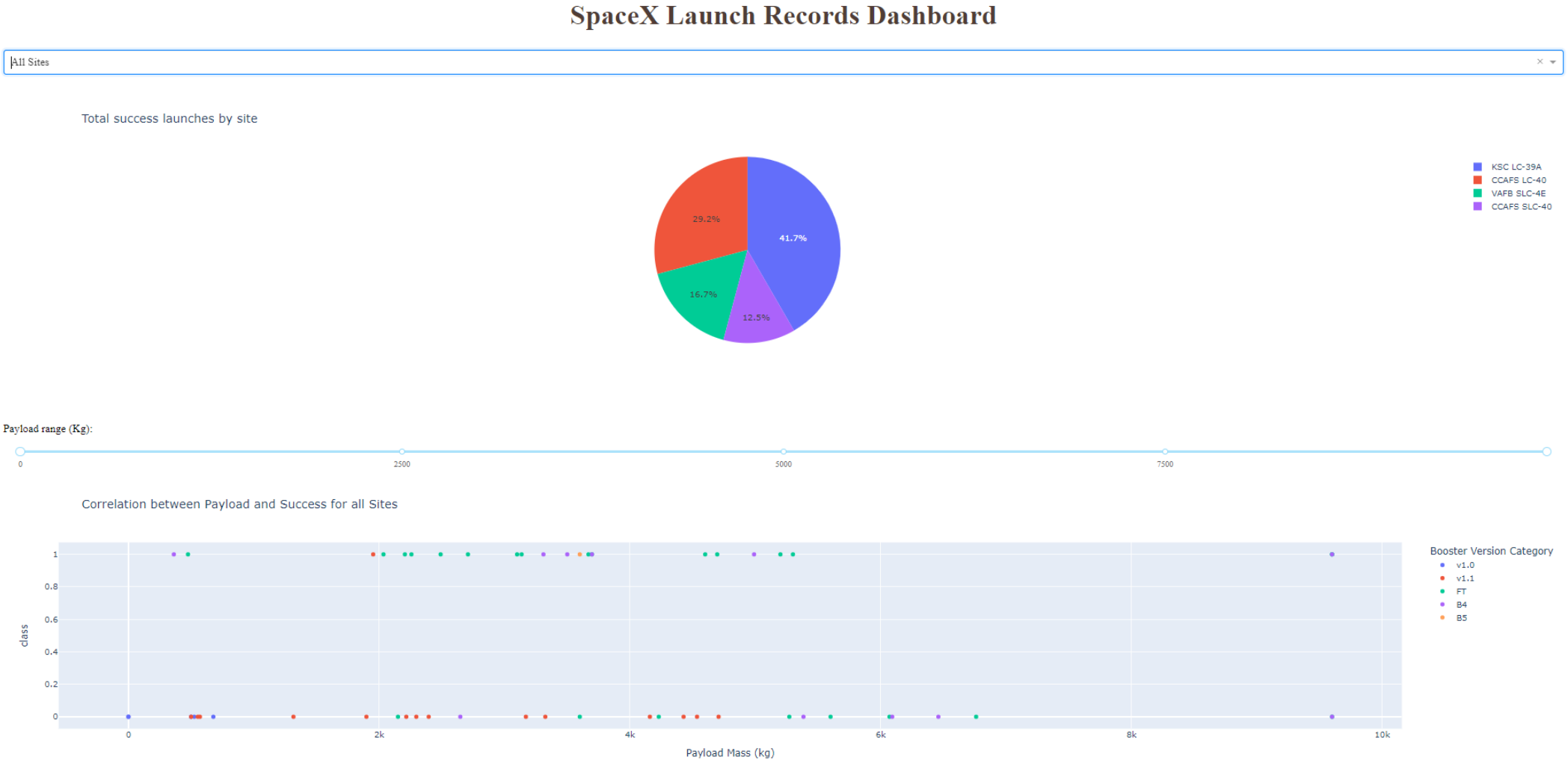
- Exploratory data analysis results :



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

Results

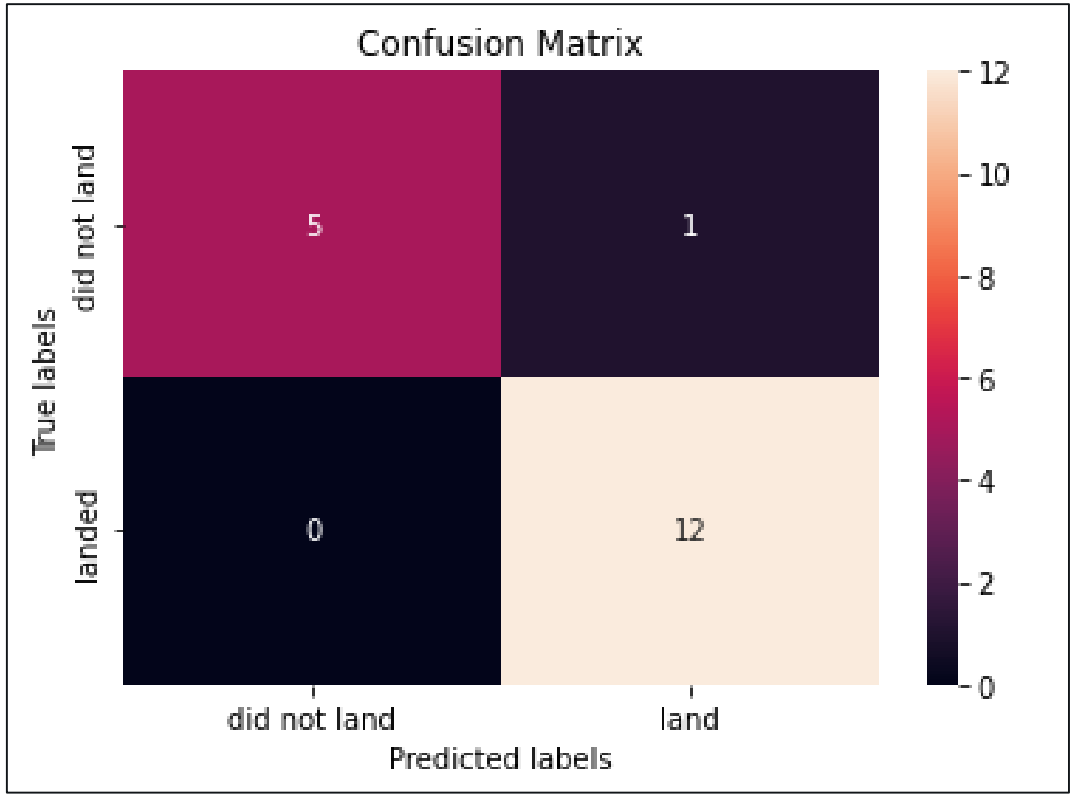
- Interactive analytics demo in screenshots :

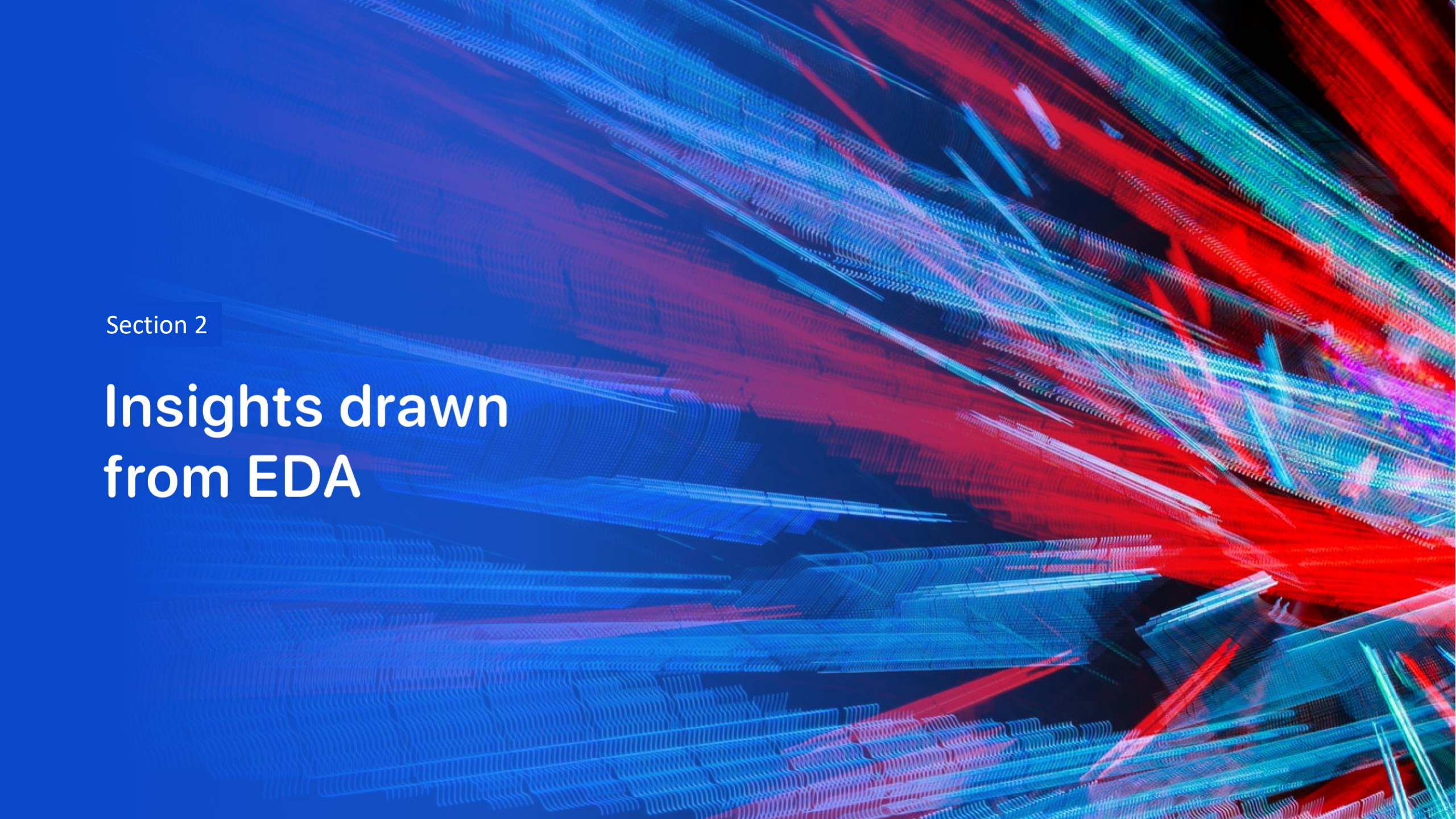


Results

- Predictive analysis results :
  - **Decision Tree** looks to be the best predictive model with Landing accuracy : **88%** and Test Accuracy : **94%**

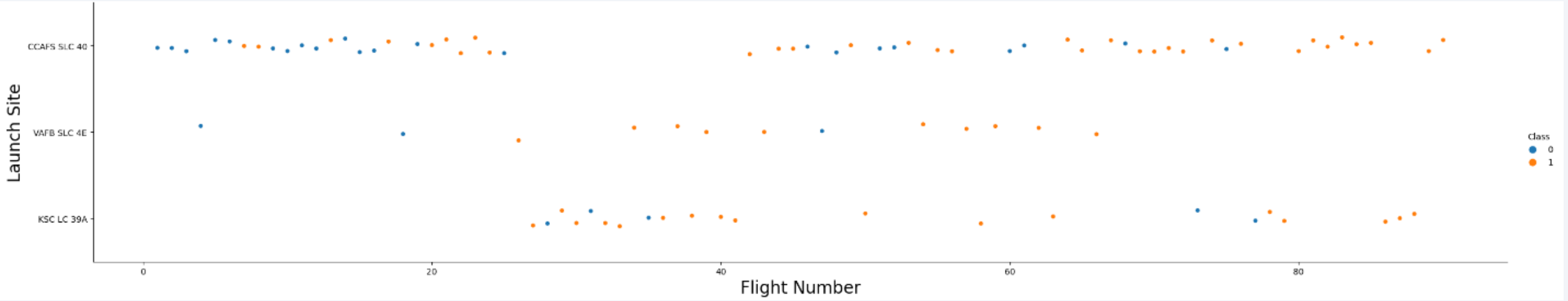
Algorithm	Score (%)
KNN	88.33
Decision Tree	94.44
SVM	83.33
LogisticRegression	83.33




Section 2
Insights drawn from EDA

Flight Number vs. Launch Site

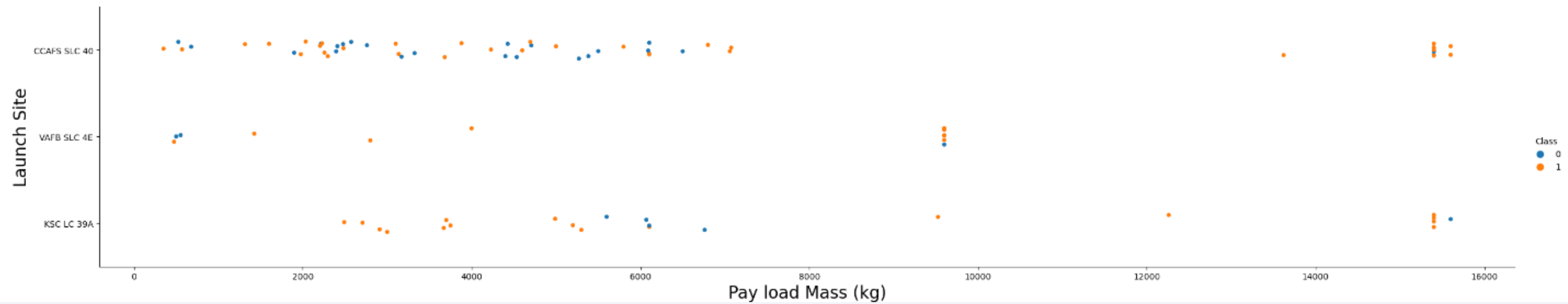
- Show a scatter plot of Flight Number vs. Launch Site



- We see that as the flight number increases, the first stage is more likely to land successfully in each Launch Site

Payload vs. Launch Site

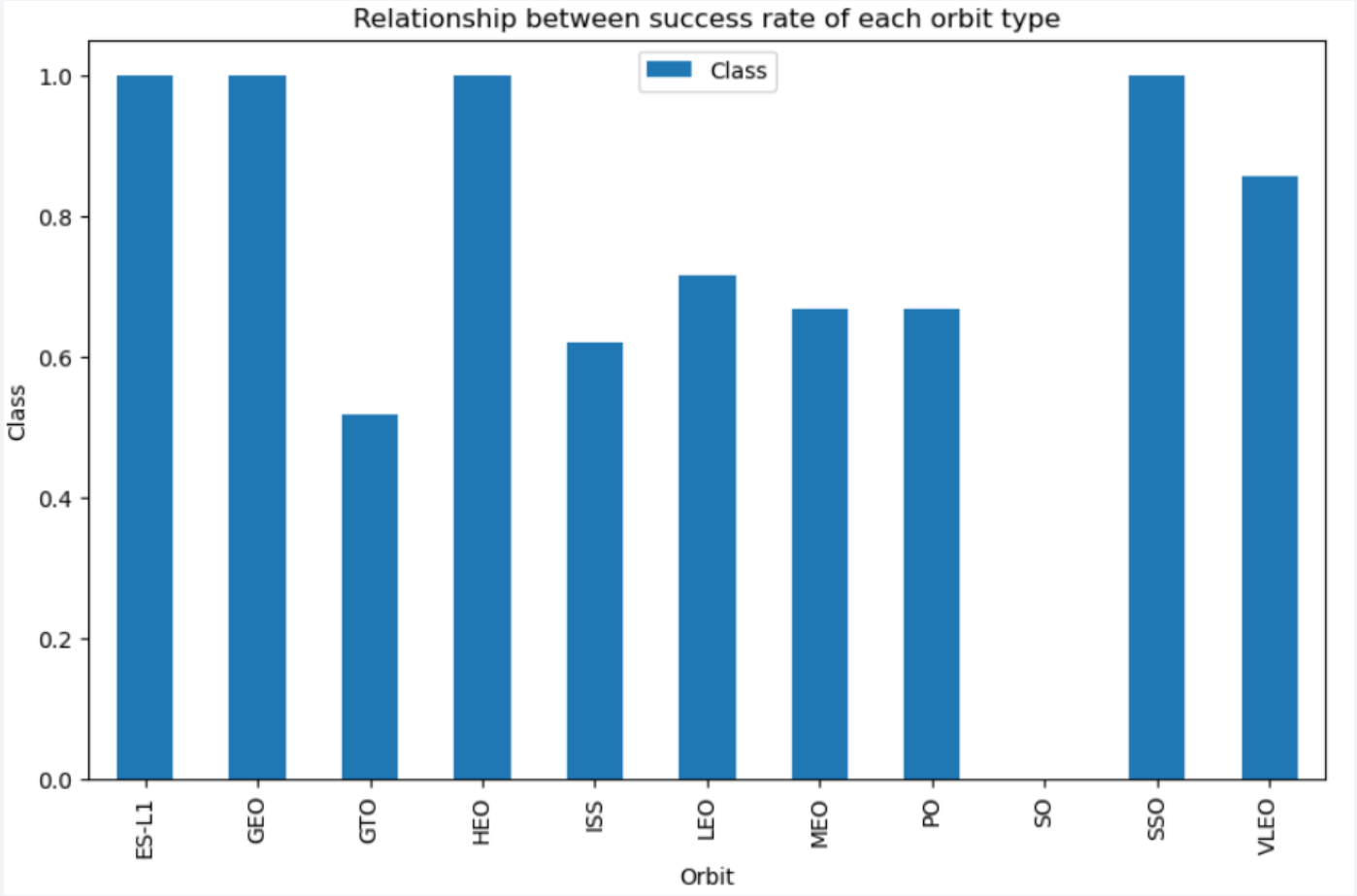
- Scatter plot of Payload vs. Launch Site



- We can observe Payload Vs. Launch Site scatter point chart you will find for the VAFB-SLC launch site there are no rockets launched for heavy payload mass(greater than 10000)

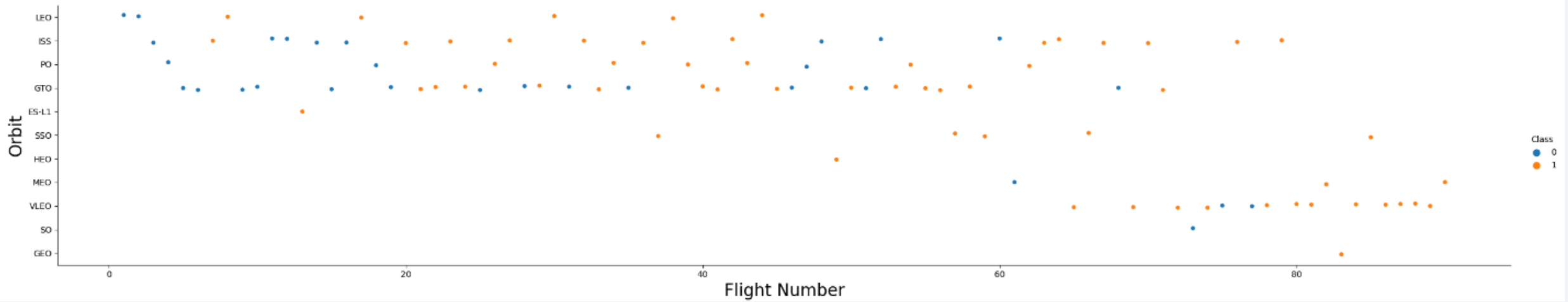
Success Rate vs. Orbit Type

- A bar chart for the success rate of each orbit type
- Orbit ES-L1, GEO, HEO, SSO have the high success rate



Flight Number vs. Orbit Type

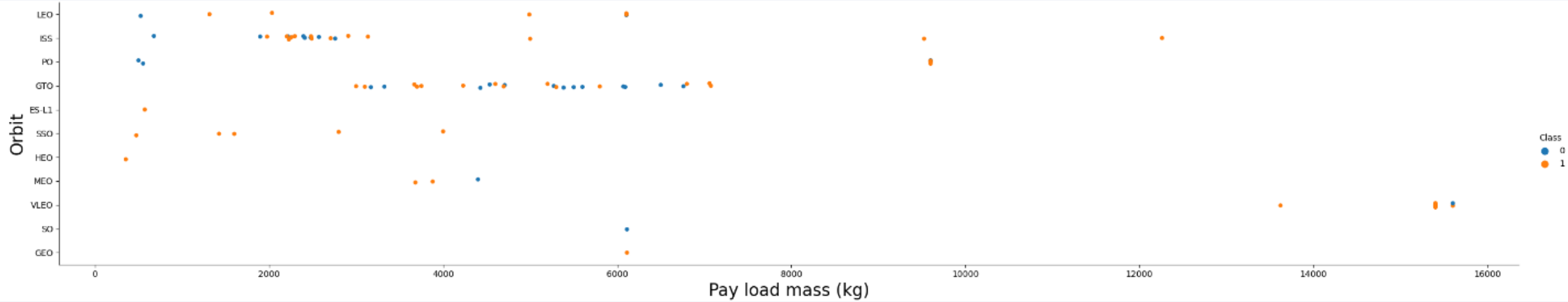
- Show a scatter point of Flight number vs. Orbit type



- We can see that 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

- Scatter point of payload vs. orbit type

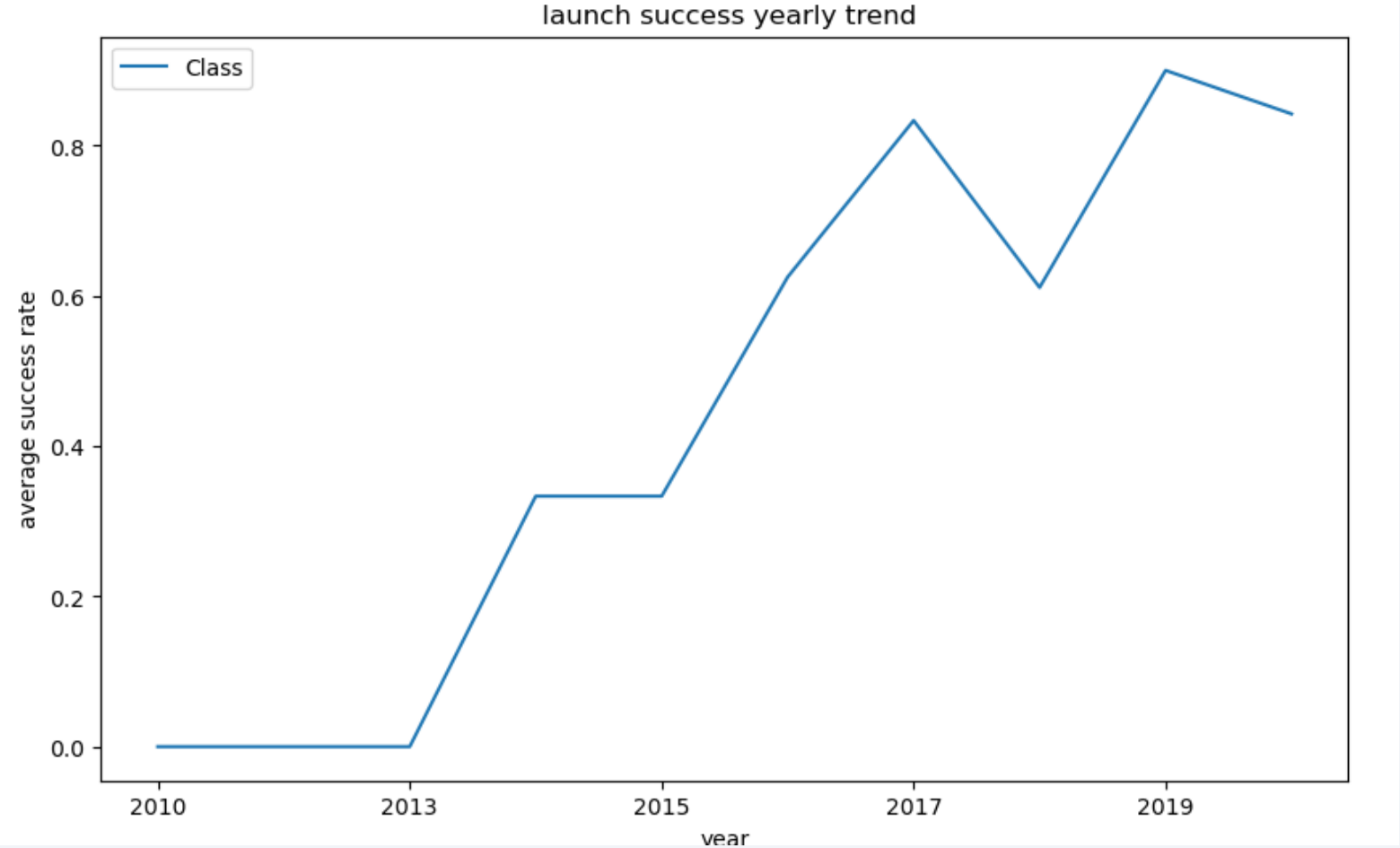


- With heavy payloads the successful landing or positive landing rate are more for Polar, LEO and ISS.
- However for GTO we cannot distinguish this well as both positive landing rate and negative landing(unsuccessful mission) are both there here.

Launch Success Yearly Trend

- Line chart of yearly average success rate

- The success rate since 2013 kept increasing till 2020



All Launch Site Names

- Find the names of the unique launch sites

```
%sql SELECT DISTINCT(Launch_Site) FROM SPACEXTBL
```

- query result of name of the 4 unique sites

[158]:

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`

```
%sql SELECT * FROM SPACEXTBL WHERE Launch_Site LIKE "CCA%" LIMIT 5
```

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing _Outcome
04-06-2010	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
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)	NASA (COTS) NRO	Success	Failure (parachute)
22-05-2012	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
08-10-2012	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
01-03-2013	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

Total Payload Mass

- Calculate the total payload carried by boosters from NASA

```
%sql SELECT Customer, SUM(PAYLOAD_MASS__KG_) AS TOTAL_PAYLOAD_MASS_KG FROM SPACEXTBL WHERE Customer = 'NASA (CRS)'
```

- query result

Customer	TOTAL_PAYLOAD_MASS_KG
NASA (CRS)	45596

Average Payload Mass by F9 v1.1

- The average payload mass carried by booster version F9 v1.1

```
%sql SELECT Booster_Version, AVG(PAYLOAD_MASS__KG_) AS AVG_PAYLOAD_MASS_KG FROM SPACEXTBL WHERE Booster_Version = 'F9 v1.1'
```

- Query result

Booster_Version	AVG_PAYLOAD_MASS_KG
F9 v1.1	2928.4

First Successful Ground Landing Date

- Find the dates of the first successful landing outcome on ground pad

```
%sql SELECT "Date","Landing _Outcome", MIN("Date") FROM SPACEXTBL WHERE "Landing _Outcome" LIKE "%ground pad%"
```

- Query result

Date	Landing _Outcome	MIN("Date")
01-05-2017	Success (ground pad)	01-05-2017

Successful Drone Ship Landing with Payload between 4000 and 6000

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

```
%sql SELECT Booster_Version, PAYLOAD_MASS__KG_, "Landing _Outcome" FROM SPACEXTBL WHERE "Landing _Outcome" LIKE "%success%drone ship%" AND (PAYLOAD_MASS__KG_ BETWEEN 4000 AND 6000)
```

- Query result

Booster_Version	PAYLOAD_MASS__KG_	Landing _Outcome
F9 FT B1022	4696	Success (drone ship)
F9 FT B1026	4600	Success (drone ship)
F9 FT B1021.2	5300	Success (drone ship)
F9 FT B1031.2	5200	Success (drone ship)

Total Number of Successful and Failure Mission Outcomes

- Calculate the total number of successful and failure mission outcomes

```
%sql SELECT Mission_Outcome, COUNT(Mission_Outcome) AS TOTAL_SUCCESS_FAILURE FROM SPACEXTBL GROUP BY Mission_Outcome
```

- Query result

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

Boosters Carried Maximum Payload

- List the names of the booster which have carried the maximum payload mass

```
%sql SELECT Booster_Version, PAYLOAD_MASS__KG_ FROM SPACEXTBL \
WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTBL)
```

- Query result on the right

Booster_Version	PAYLOAD_MASS__KG_
F9 B5 B1048.4	15600
F9 B5 B1049.4	15600
F9 B5 B1051.3	15600
F9 B5 B1056.4	15600
F9 B5 B1048.5	15600
F9 B5 B1051.4	15600
F9 B5 B1049.5	15600
F9 B5 B1060.2	15600
F9 B5 B1058.3	15600
F9 B5 B1051.6	15600
F9 B5 B1060.3	15600
F9 B5 B1049.7	15600

2015 Launch Records

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

```
%sql SELECT "Date", substr("Date", 4, 2), "Landing _Outcome", Booster_Version, Launch_Site FROM SPACEXTBL \
WHERE "Landing _Outcome" LIKE "%failure%" AND substr(Date,7,4)='2015'
```

- Query result

Date	substr("Date", 4, 2)	Landing _Outcome	Booster_Version	Launch_Site
10-01-2015	01	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
14-04-2015	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 "Date", "Landing _Outcome", COUNT("Landing _Outcome") AS SLO FROM SPACEXTBL  \
WHERE "Landing _Outcome" LIKE "%Success%" GROUP BY ("Date" BETWEEN '04-06-2010' AND '20-03-2017') ORDER BY SLO DESC
```

- Query result :

Date	Landing _Outcome	SLO
08-04-2016	Success (drone ship)	34
22-12-2015	Success (ground pad)	27

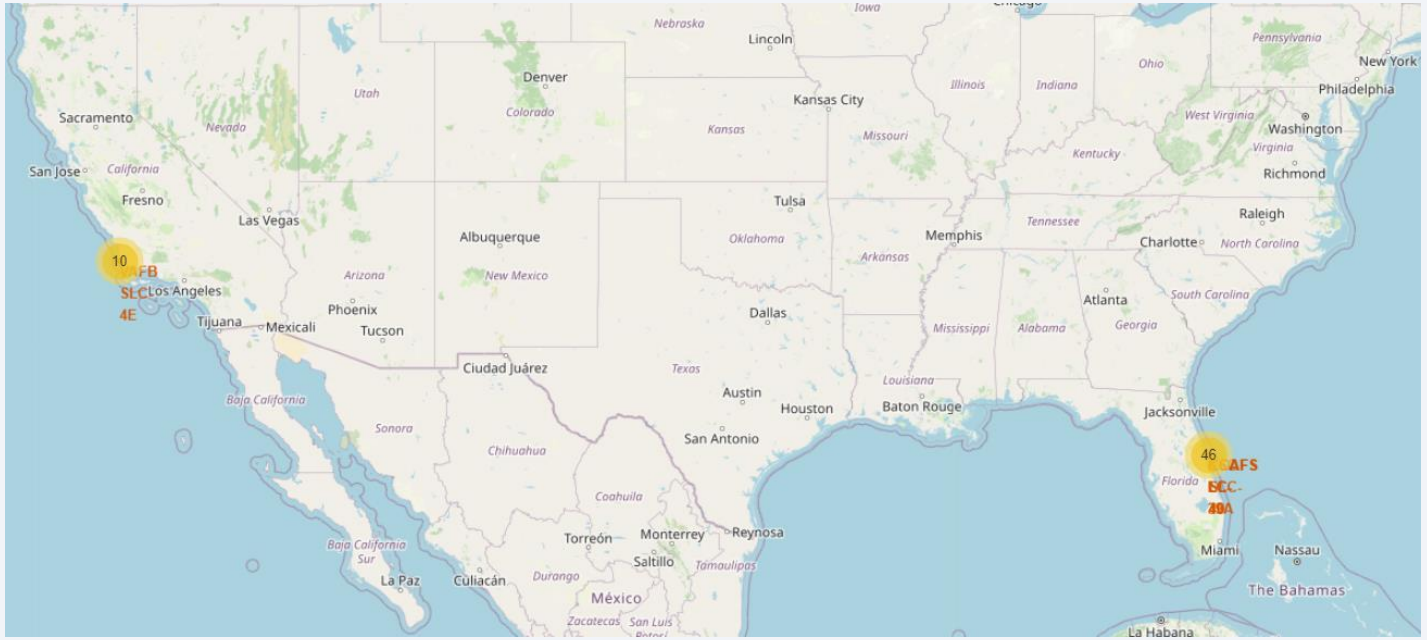
Section 3

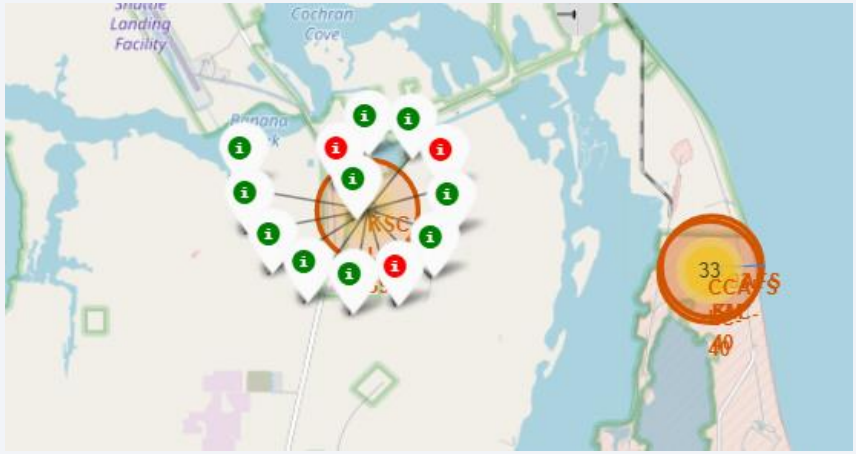
Launch Sites Proximities Analysis

All launch sites on a map

- VAFB SLC-4E is located in California
- CCAFS LC-40, CCAFS SLC-40, KSC LC-39A are located in Florida

Mark of success/failed launches for each site on the map



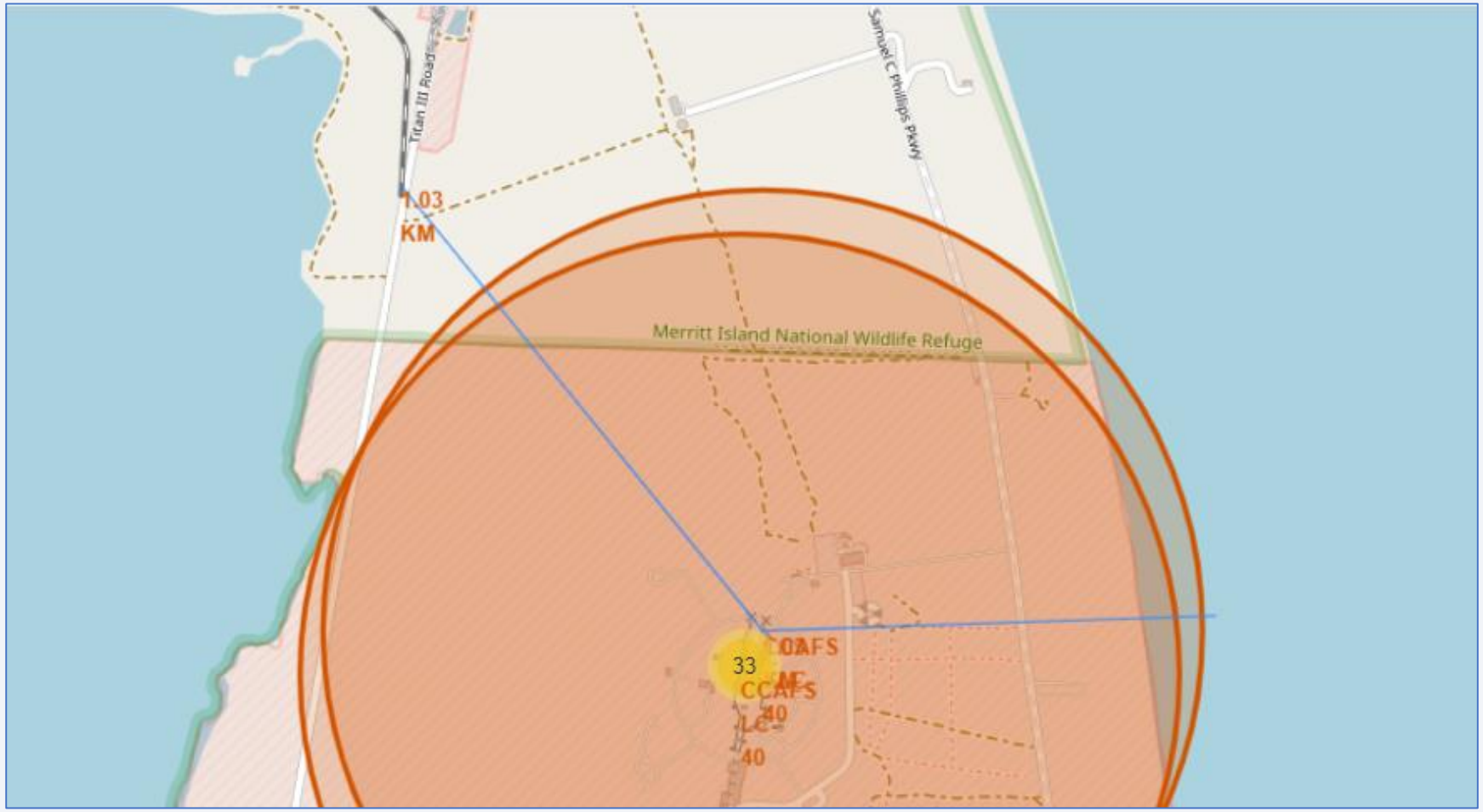


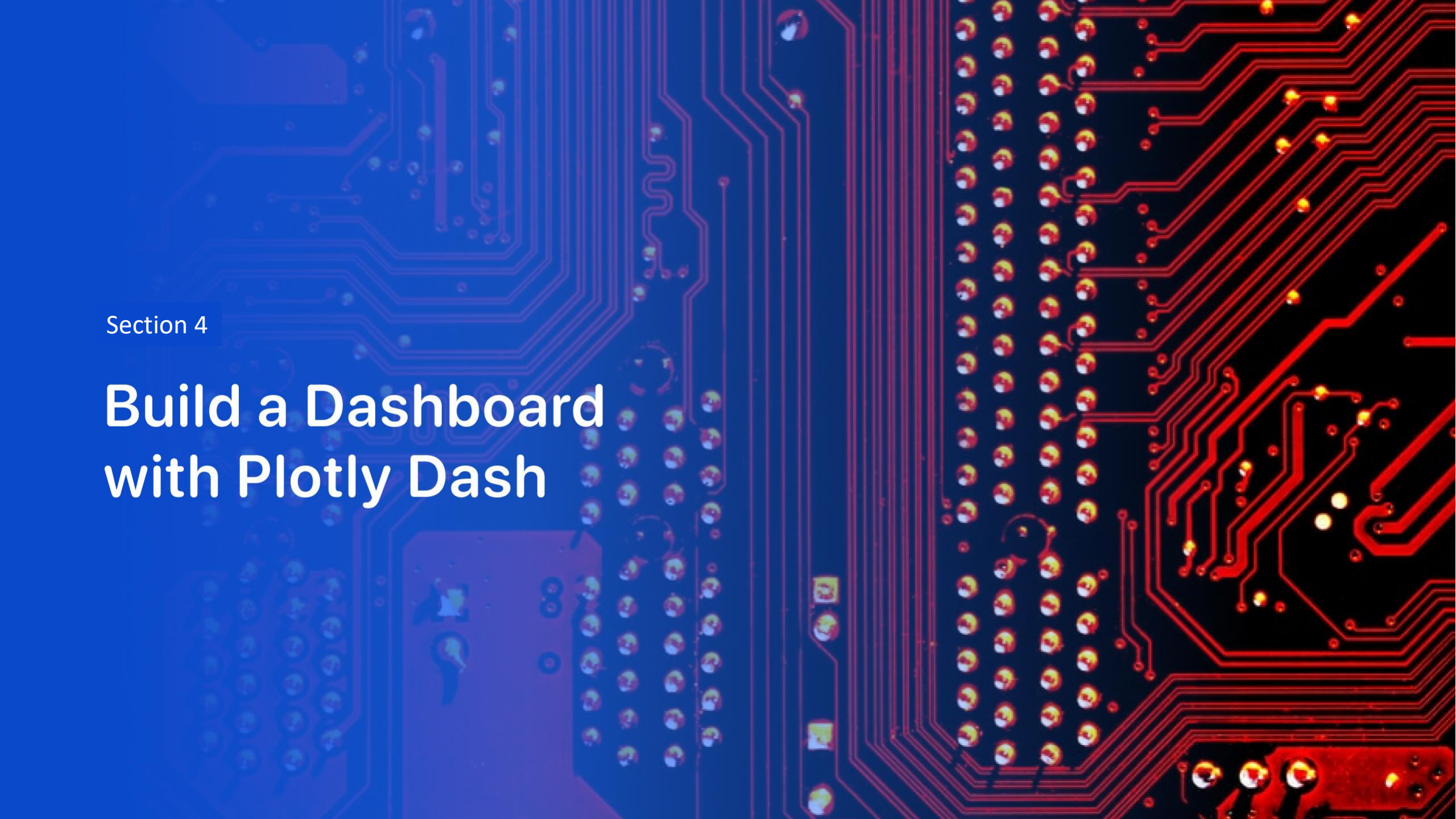
- If we zoom in we can identify which launch sites have relatively high success rates.

- Total of success/failed launches for each site are marked as green/red

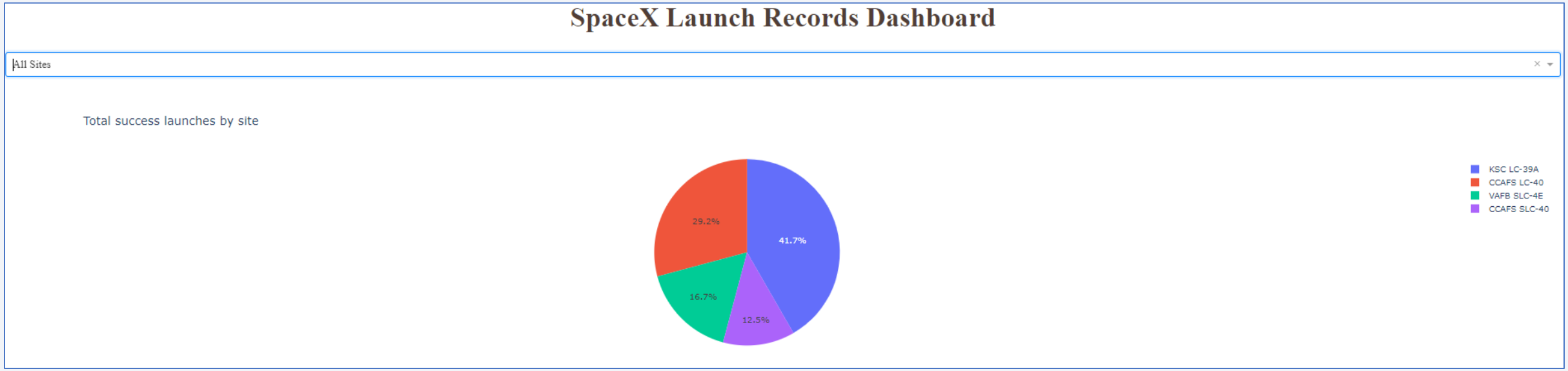
The distances between a launch site to its proximities

Line between a launch site and **coastline**, **railway** to illustrate the finding.
Similar approach for highway and city map



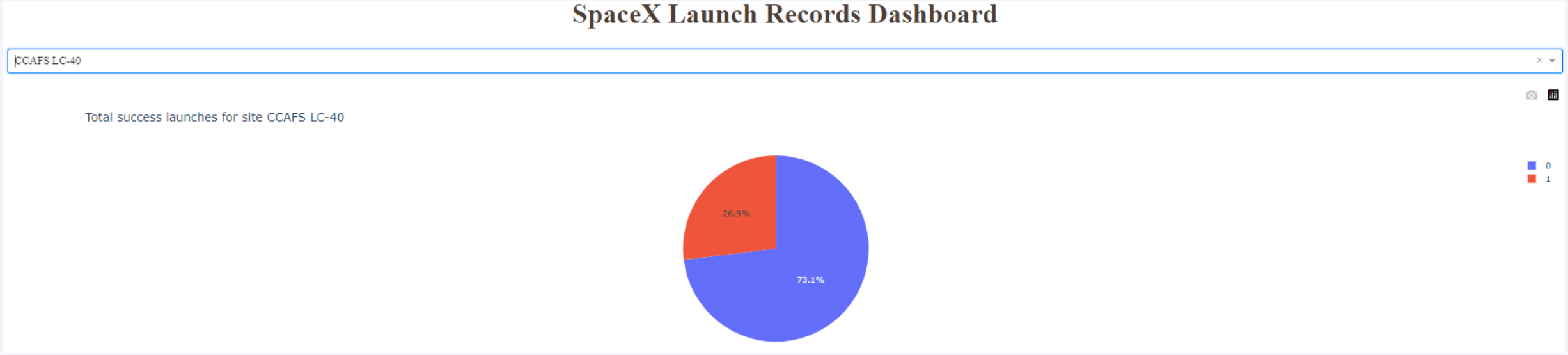

Section 4
Build a Dashboard with Plotly Dash

The screenshot of launch success count for all sites, in a piechart



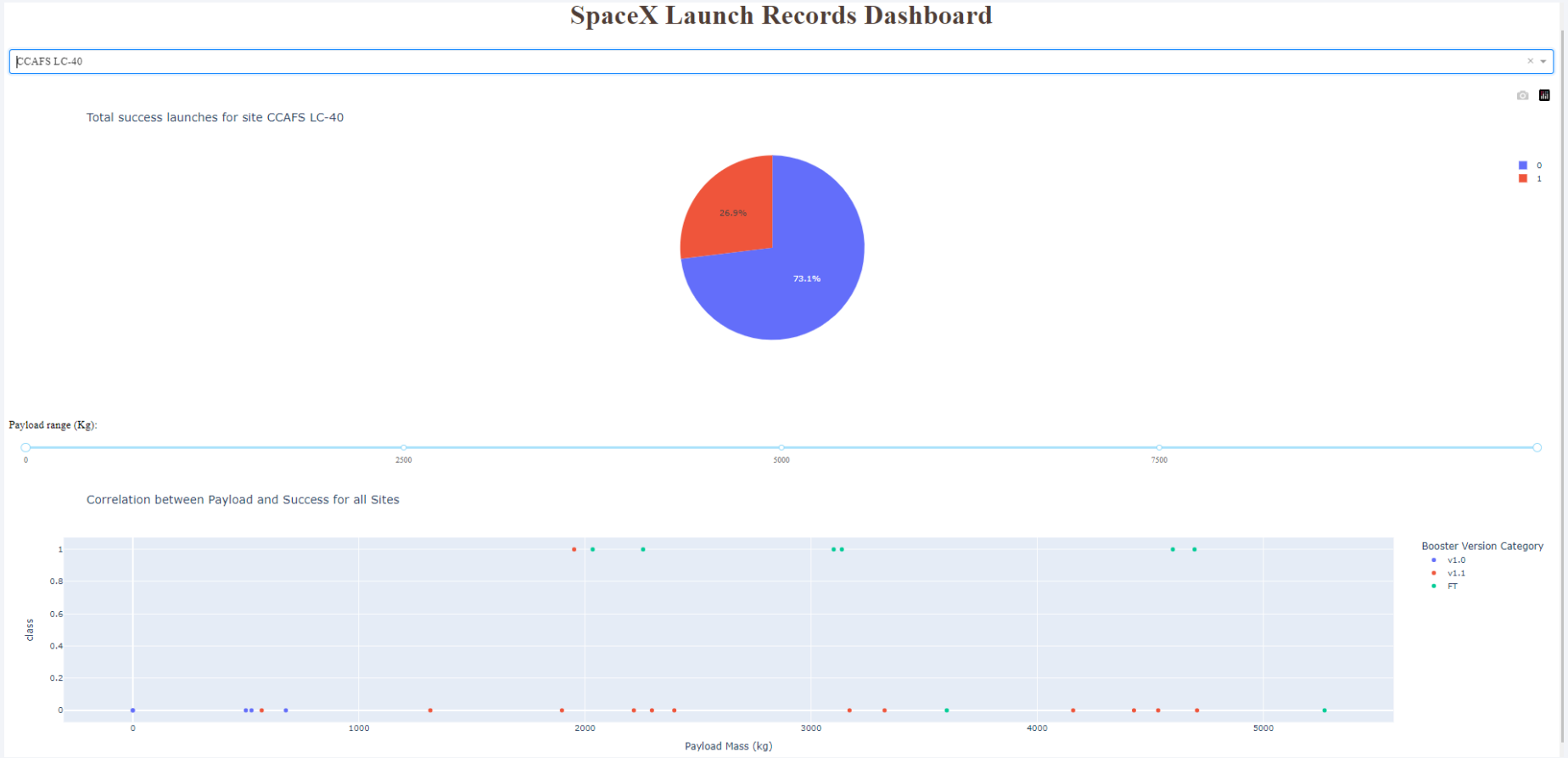
- Total success launch by site, we can see that **KSC LC-39A** has the best rate of success of 41.7%

The screenshot of the piechart for the launch site with highest launch success ratio



- Total success launches for site CCAFS LC-40

Payload vs. Launch Outcome scatter plot for all sites



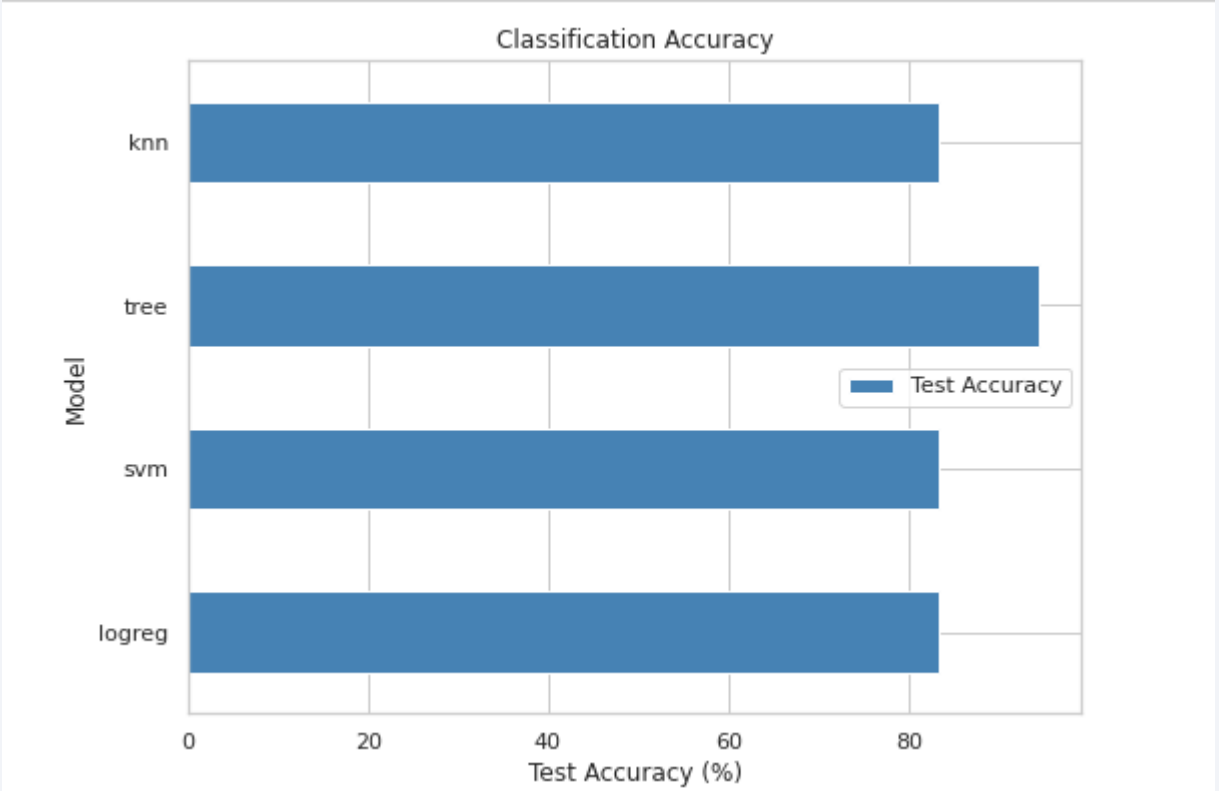
- The range slider allows to select different payloads which can impact the success outcome according to the payload value

Section 5

Predictive Analysis (Classification)

Classification Accuracy

- Built model accuracy for all built classification models, in a bar chart
- The decision Tree has highest accuracy of 94.44%



Confusion Matrix

- The confusion matrix of the best performing model – Decision Tree
  - We can observe that **TP = 12** High number

    and we have **TN = 5** the low one
  - But as shown in the heatmap reflets the FP, FN values

    are very closed which leads to stable model
  - We can calculate the score :
    - Precision = 12/(12 + 1) = 0,92 and Recall = 12(12 + 0) = 1
    - Score = 2*(prc*rec)/(prc + rec) = 2*(0.92)/1.92 = 0.96
    - The score is 96% is closed to 94% predicted by the model

Conclusion

- Launch success rate started to increase in 2013 till 2020
- The best launch site is **KSC LC 39A** of 41.7%
- Orbit ES-L1, GEO, HEO, SSO have the high success rate
- The decision Tree has highest accuracy of 94.44%

Thank you!