IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Pralabh Poudel
LinkedIn: https://www.linkedin.com/in/pralabh-poudel/
Kaggle: https://www.kaggle.com/pralabhpoudel

Outline

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

Executive Summary

- This report consists of all the findings, completed after data analysis of SpaceX data to predict the likelihood of first stage of successful rocket landing and assist in the cost prediction of the launch.

- The data was collected using SpaceX API and web scraped from SpaceX Wikipedia page, then wrangled using Jupyter Notebook as the platform and Python as the programming Language. Exploration of the dataset was done using SQL and Python. Multiple data visualizations were built during EDA process.

- Multiple classification models such as Logistic Regression, Support Vector Machine, Decision Tree and K Nearest Neighbour were tested to figure out the best predictive model.

- Some key findings from the analysis process:
    - More the flight, the greater the success rate at a launch site.
    - In launch site CCAFS SLC 40, there's a higher success rate for rocket, as the payload mass increases.
    - The success rate of the rocket has been increasing since 2013

- Decision Tree is the best algorithm to predict the landing outcome as it has training accuracy of 88% and achieved accuracy of 83% in testing data.

Introduction

Project background and context

SpaceX advertises Falcon 9 rocket launches on its website, with a cost of 62 million dollars; other providers cost upward 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. This information can be used if an alternate company wants to bid against SpaceX for a rocket launch.

SpaceX

Problems you want to find answers

- Find the factors that leads to successful landing outcomes.
- How the features interact with each other when the success rate is higher?
- Will SpaceX will reuse the first stage?

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:

  The collection of data was completed using SpaceX API and web scraping from SpaceX Wikipedia page

- Perform data wrangling

  Landing outcome label was created from outcome column. One-hot encoding was applied to the categorical features.

- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models

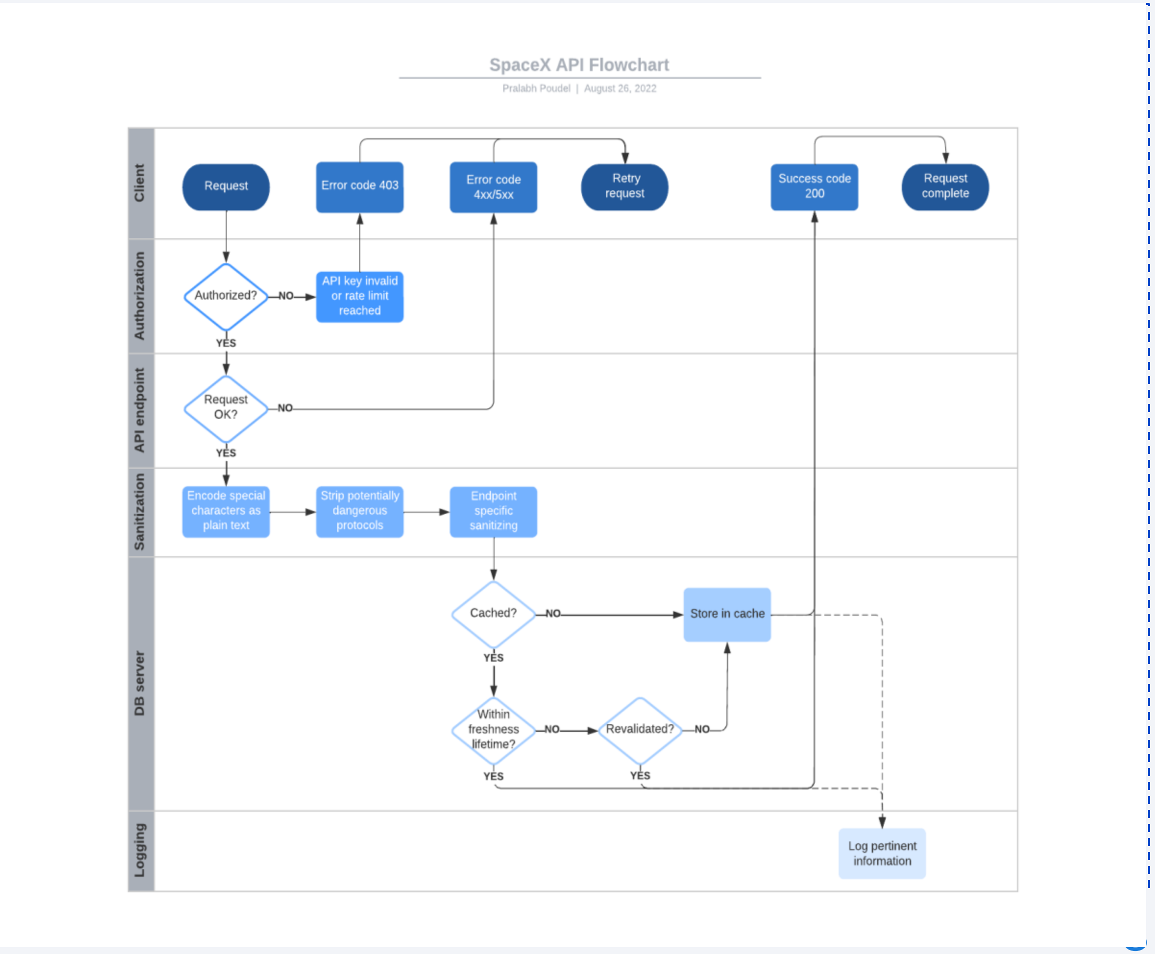
  The models were trained in every way to figure out the best accuracy result, tested with testing data and evaluated using confusion matrix

Data Collection

- Data collection was completed using two methods, I.e. through SpaceX API and Web Scraping SpaceX Wikipedia page using BeautifulSoup python library.
- The data that was collected through SpaceX API was in JSON file format, it was done using .json() function and was converted into Pandas dataframe using .json_normalize(),
- The main goal of web scraping from Wikipedia page of SpaceX was to extract launch records of Falcon9 from HTML Table, parse the table and transform it into Pandas dataframe for further analysis.
- The data was cleaned, checked for missing values and required steps were taken to handle the missing values.

Data Collection – SpaceX API

- A request was send to SpaceX API to collect data, then it was cleaned and wrangled for improving data quality on Jupyter notebook using Python as the programming language.
- Click here to view the notebook.
- The flowchart is also included in the project repository.

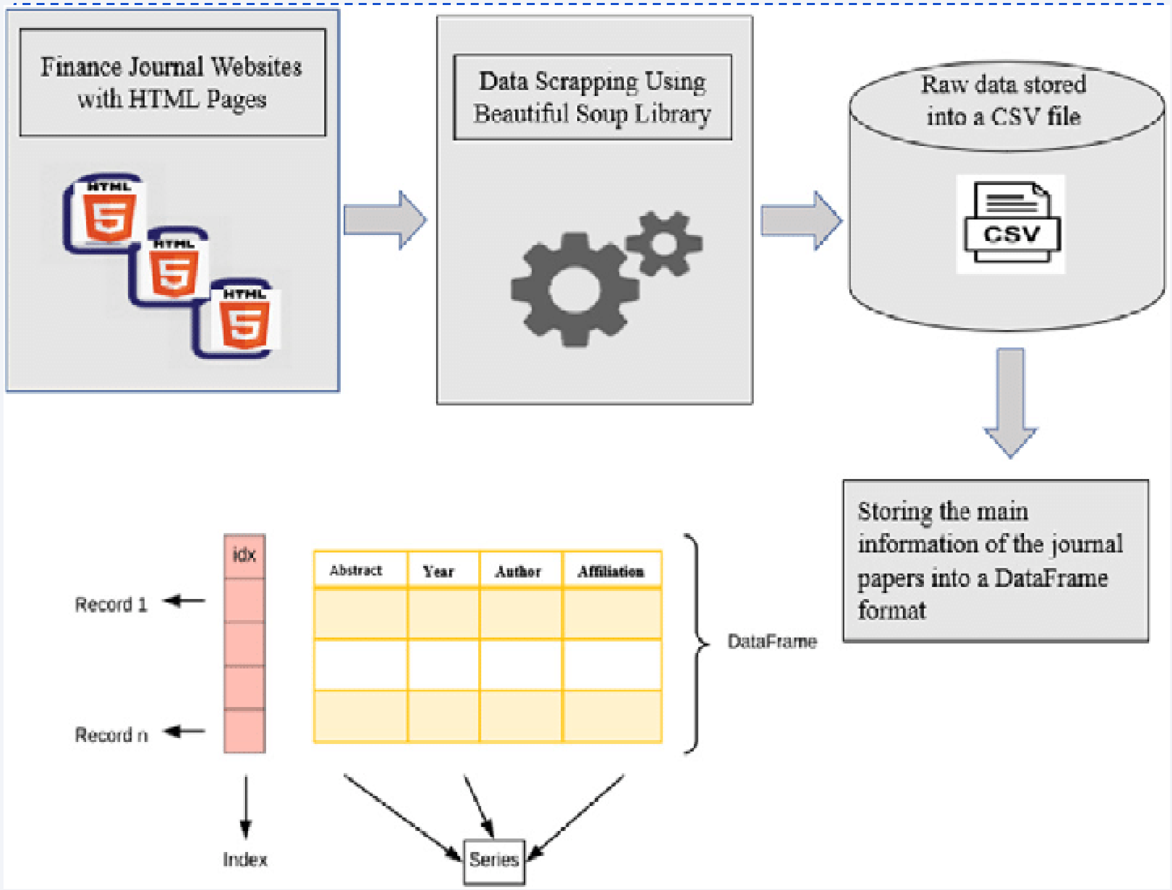


SpaceX API Flowchart

Pralabh Poudel | August 26, 2022

Data Collection - Scraping

- Data from SpaceX Wikipedia page was scarped using BeautifulSoup Library in Jupyter Notebook.
- The scarped data was parsed and was saved as a Pandas dataframe
- Click here to view the notebook


Finance Journal Websites with HTML Pages
HTML
Data Scrapping Using Beautiful Soup Library
Raw data stored into a CSV file
CSV
Storing the main information of the journal papers into a DataFrame format
idx
Abstract
Year
Author
Affiliation
Record 1
Record n
DataFrame
Index
Series

https://www.researchgate.net/figure/Data-collection-and-preparation-flowchart-In-this-step-the-data-is-scraped-from-one-of_fig2_350173610

Data Wrangling

- The dataset was explored to determine the training label
- The data was explored by calculating number of launches from each sites, calculating the number and occurrence of mission outcome per orbit type
- A training label was created by classifying different landing outcomes into Successful and Failed.
- Click here to view the notebook.

EDA with Data Visualization

- Summarize what charts were plotted and why you used those charts
- Visualizations were made using Matplotlib and Seaborn library.
- The following charts were made to gain different insights
  - Scatterplot:
    - Interaction between flight number and launch site
    - Interaction between payload and launch site
    - Interaction between flight number and orbit type
    - Interaction between payload and orbit type
  - Bar Chart
    - Success rate on each Orbit Type
  - Line plot:
    - Trend of Launch Succeses
- Click here to view the notebook

EDA with SQL

- The clean data was imported into IBM Db2 database and later was accessed through Jupyter notebook to query the data.
- Different SQL queries were executed to gain multiple information like:
  - Names of unique launch sites in the space mission.
  - Total payload mass carried by boosters launched by NASA(CRS).
  - Average payload mass carried by booster version F9 v1.1
  - Total number of successful and failure mission outcomes
  - The failed landing outcomes in drone ship, their booster version and launch site names
- Click <u>here</u> to view the notebook.

Build an Interactive Map with Folium

- The map consist of different map objects such as markers, circles, lines to display important information on the map like the location of the launch sites, their distance near to the coastline or the highway, etc.
- Landing outcome was assigned as 0 for failure and 1 for success.
- Using color-labeled marker cluster, the launch site with high success rate could be determined.
- Click here to view the notebook.

Build a Dashboard with Plotly Dash

- An interactive dashboard was built with Plotly dash showcasing information with various data visualizations such as scatter plot, pie chart, etc.
- The viewer can filter the data to gain information according to their needs.
- A pie chart was made to showcase the total launches by a certain sites because as there were no more than 4 launch site, the visualization would not be noisy and could convey the required information to the viewer.
- A scatter plot was made to showcase the relationship between payload and success rate for all sites.
- Click here to view the code.

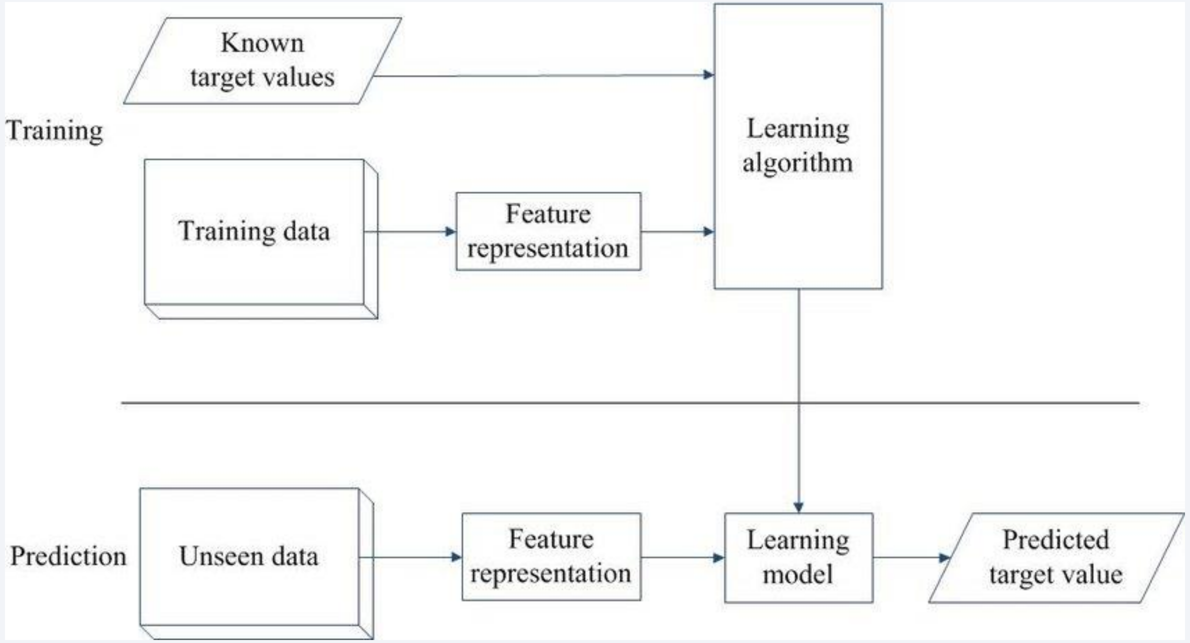
Predictive Analysis (Classification)

- The data was split into training and testing data using sklearn package.

- Multiple classification models such as Logistic Regression, Support Vector Machine, Decision Tree and K Nearest Neighbour were tested to figure out the best predictive model.

- The models were tuned using GridSearchCV to figure out the best hyperparameters for training the model

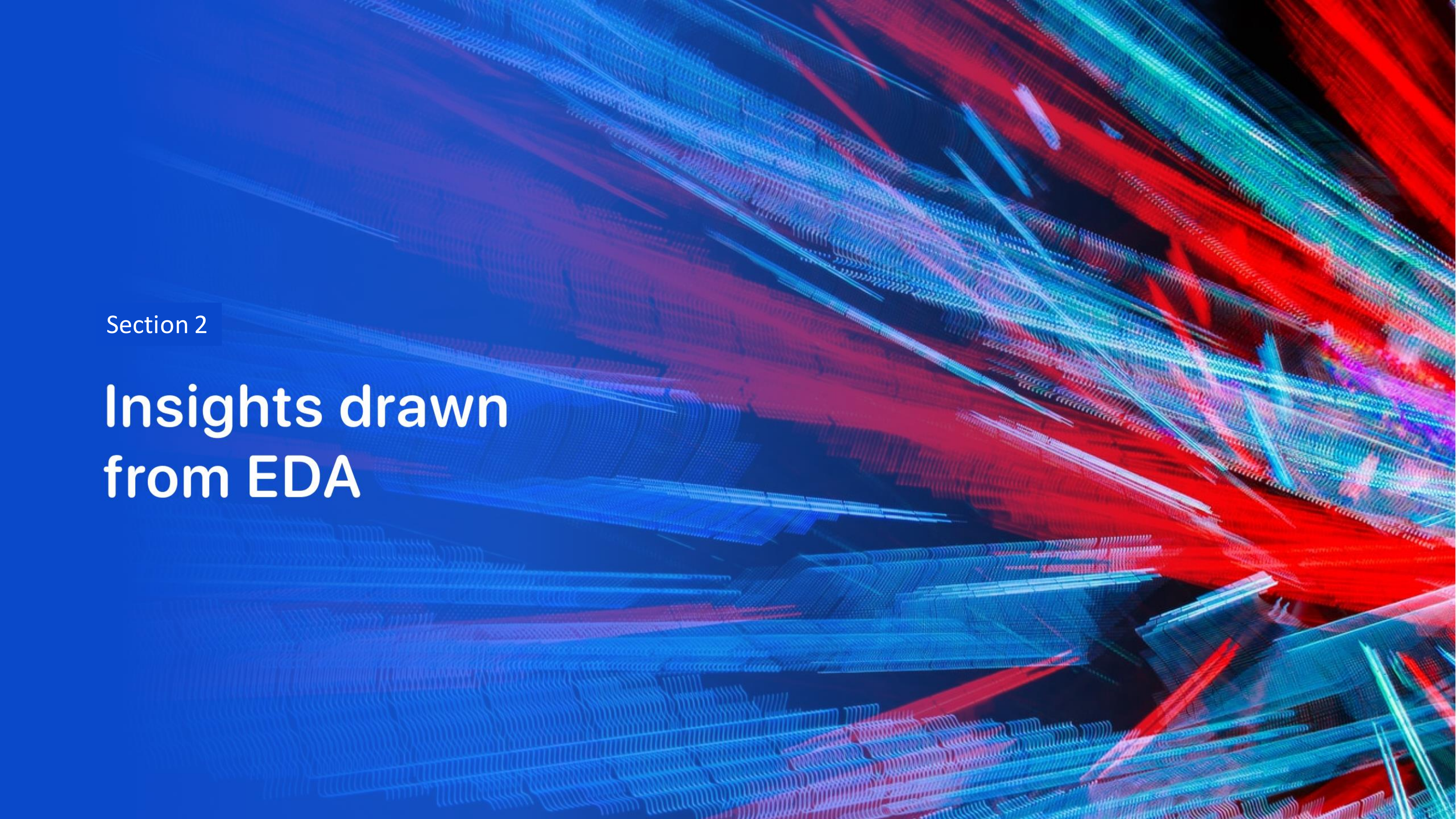
- Click here to view the notebook.



https://www.researchgate.net/profile/Dymitr-Ruta/publication/304549988/figure/fig1/AS:685506144788483@1540448809759/Flowchart-of-learning-performance-prediction-system.jpg

Results

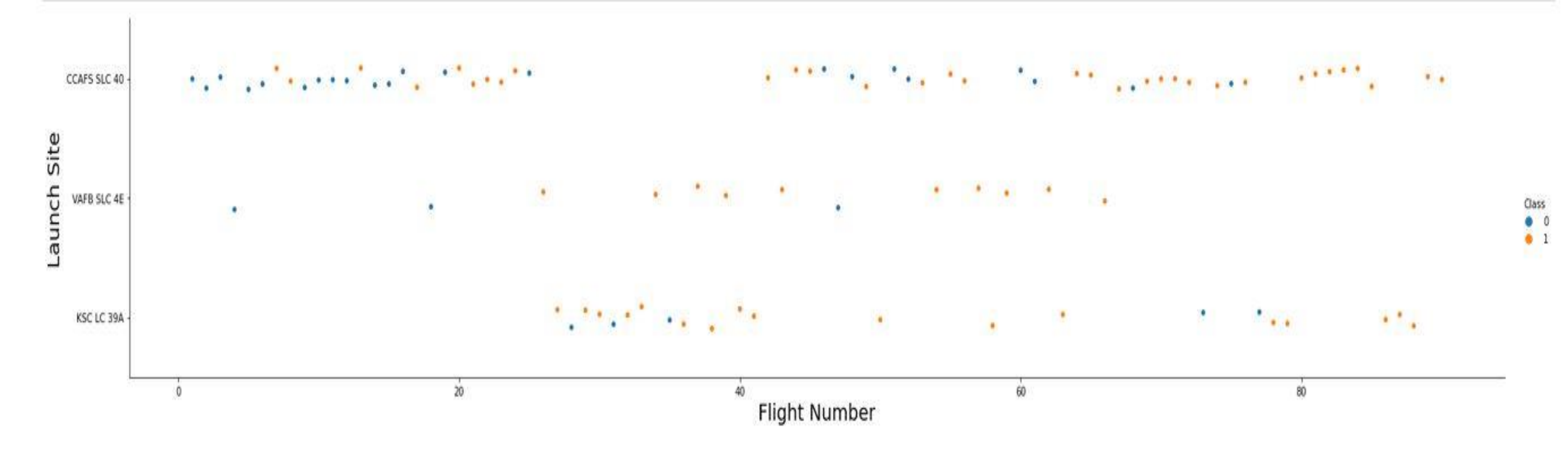
- Exploratory data analysis results
- Interactive analytics demo in screenshots
- Predictive analysis results



Section 2

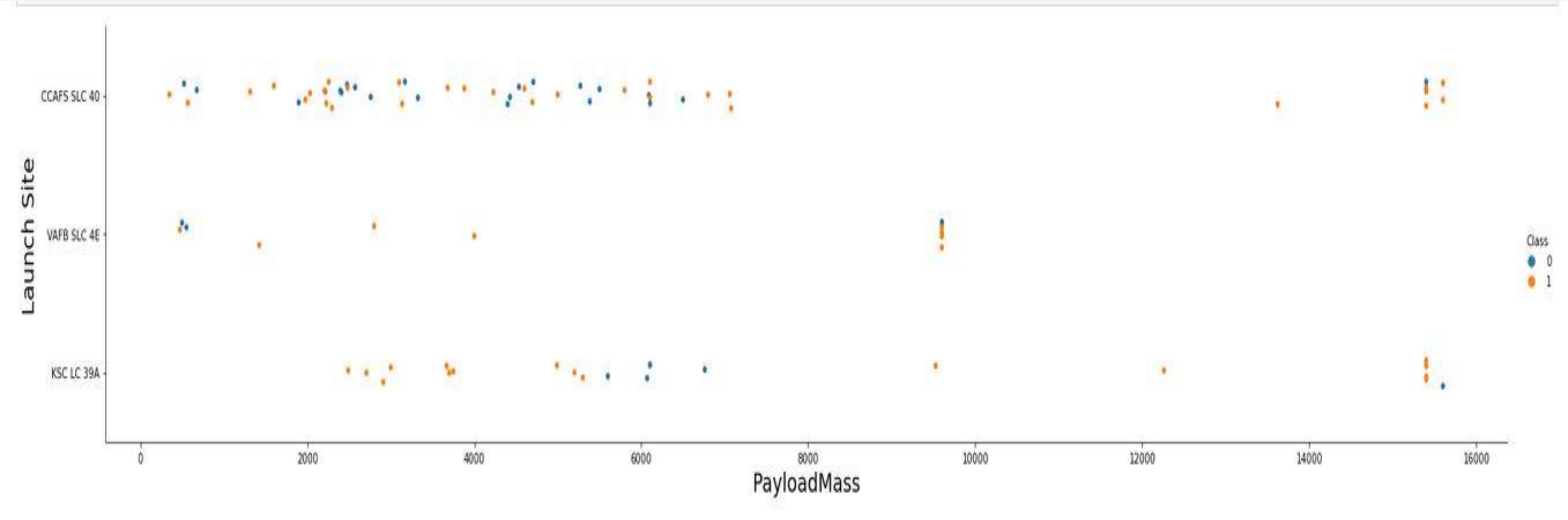
Insights drawn from EDA

Flight Number vs. Launch Site



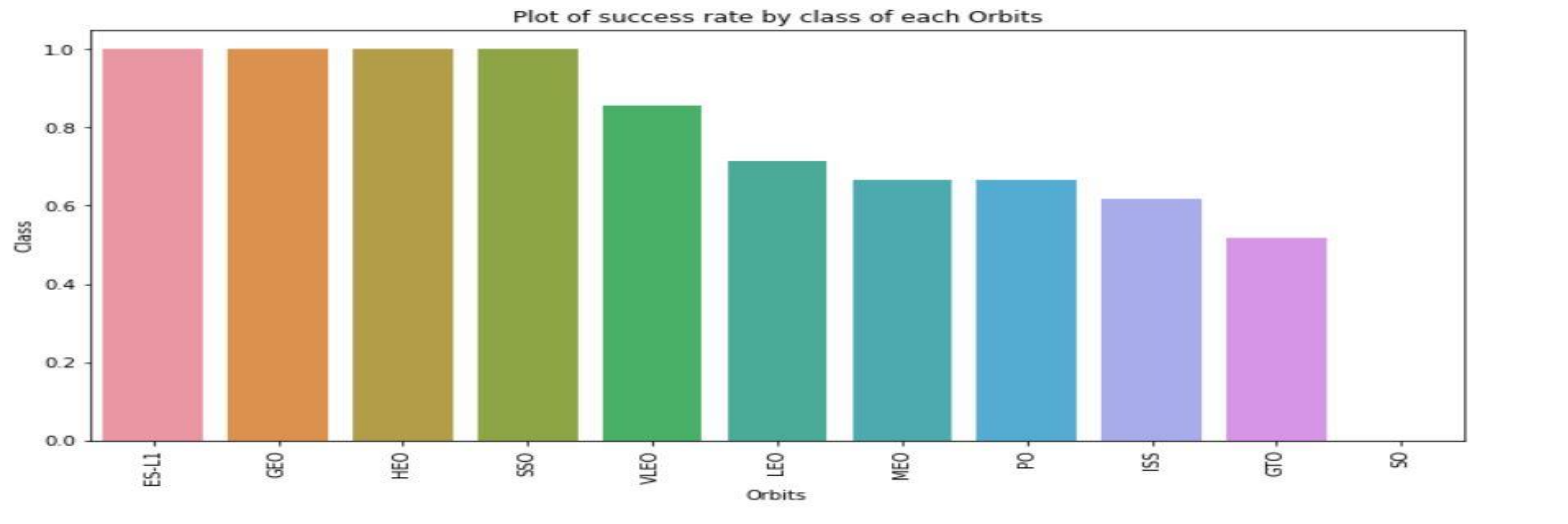
- As the number of flights increased in launch sites, there was an increase of successful landing outcomes as well.

Payload vs. Launch Site



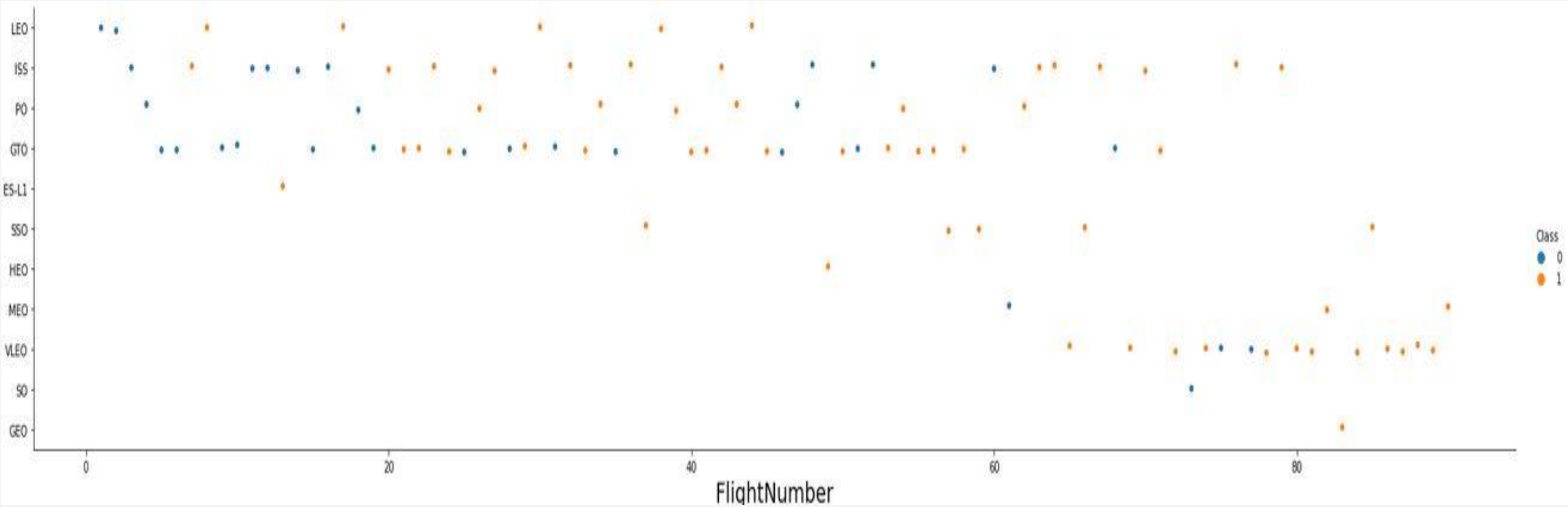
- It was found that greater the payload mass in launch site CCAFS SLC 40, higher the success rate of rocket landing.

Success Rate vs. Orbit Type



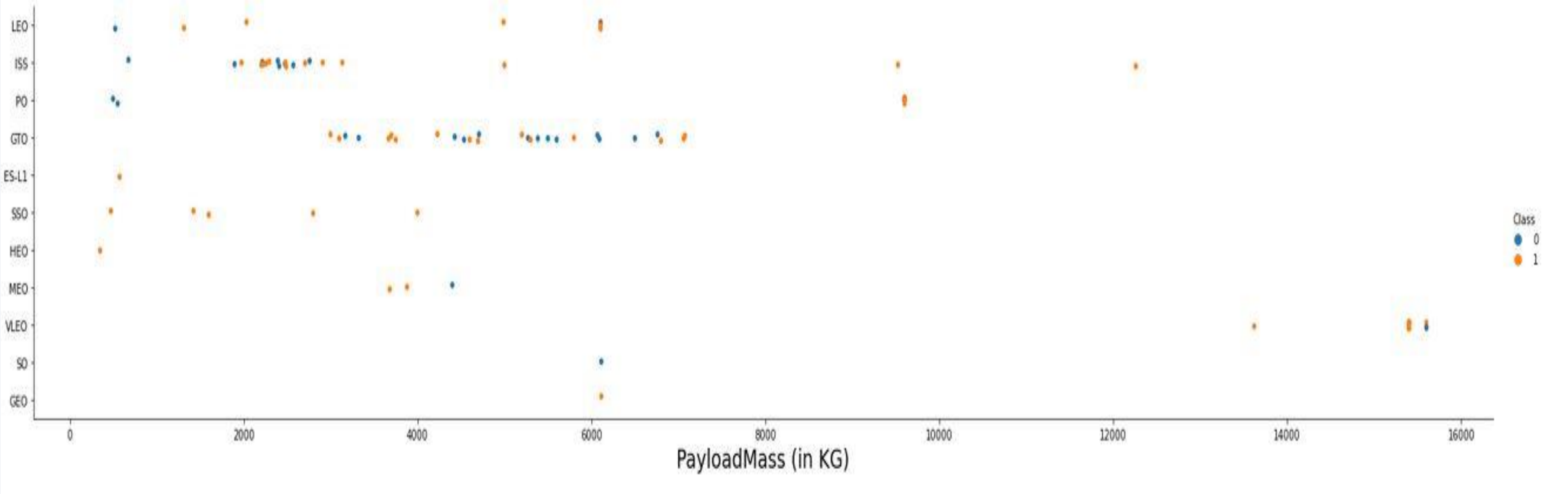
- Orbit type ES-L 1, GEO, HEO, SSO, LVEO had the most success in successful rocket landings

Flight Number vs. Orbit Type



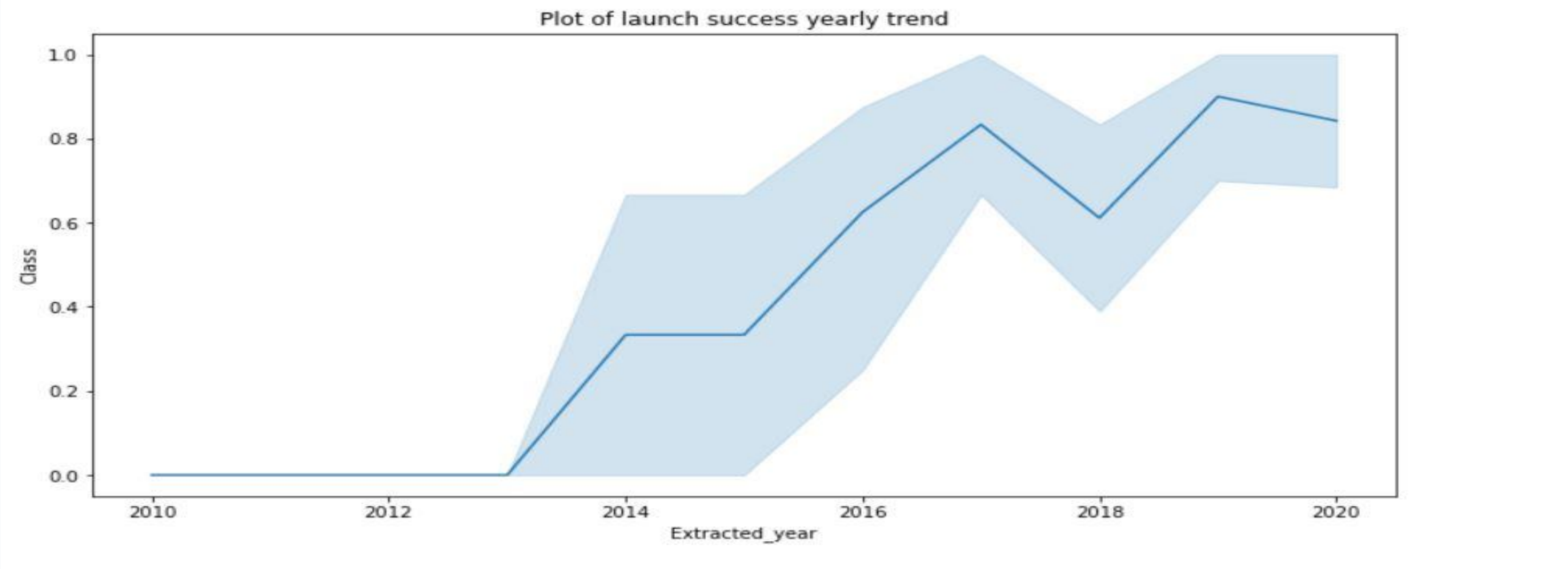
- It can be observed that orbit type LEO has more success as the number of flights increased whereas orbit type GTO, there is no any positive or negative relation between number of flights and success rate

Payload vs. Orbit Type



- Orbits with heavy payload and more successful landing are PO, LEO and ISS orbits

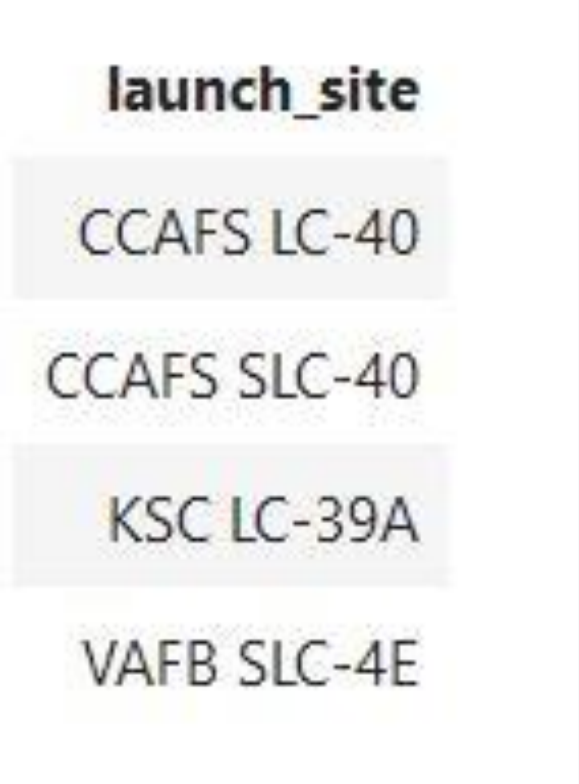
Launch Success Yearly Trend



- The overall success rate has been increasing since 2013 till 2020.

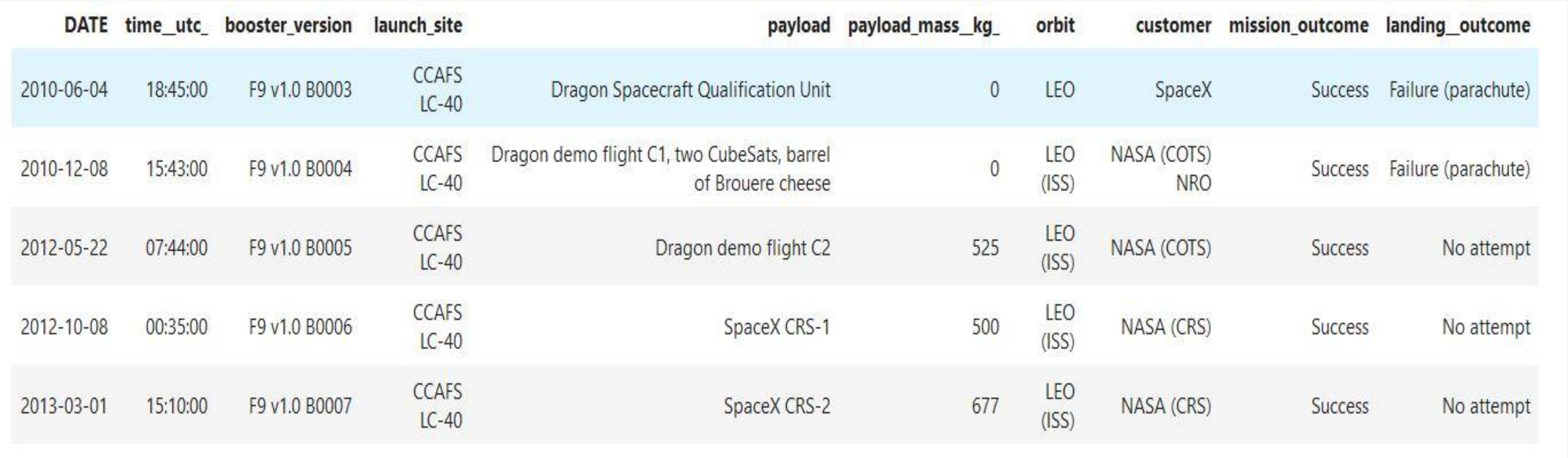
All Launch Site Names

- The key word **DISTINCT** in the SQL query was used to point out unique launch sites from SpaceX data.



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

Launch Site Names Begin with 'CCA'

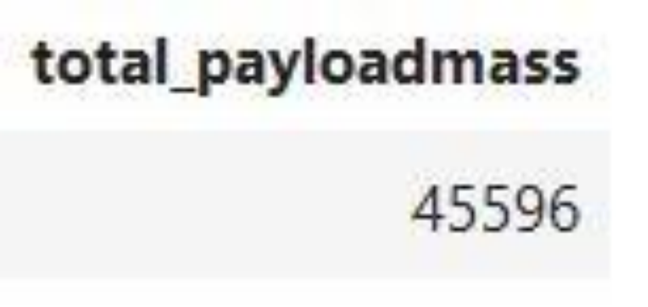


DATE	time_utc_	booster_version	launch_site	payload	payload_mass_kg_	orbit	customer	mission_outcome	landing_outcome
2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
2010-12-08	15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight 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

- Using **LIKE** operator in the SQL query, makes it easier to filter out data that is required.

Total Payload Mass

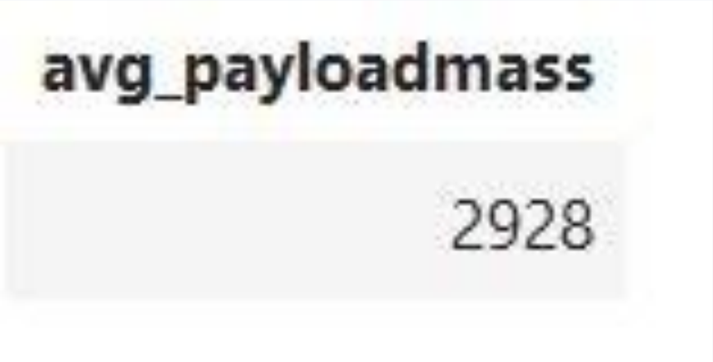
- Using SQL functions **SUM( )**, the total payload carried by boosters launched by NASA(CRS) was calculated with more efficiency.



total_payloadmass
45596

Average Payload Mass by F9 v1.1

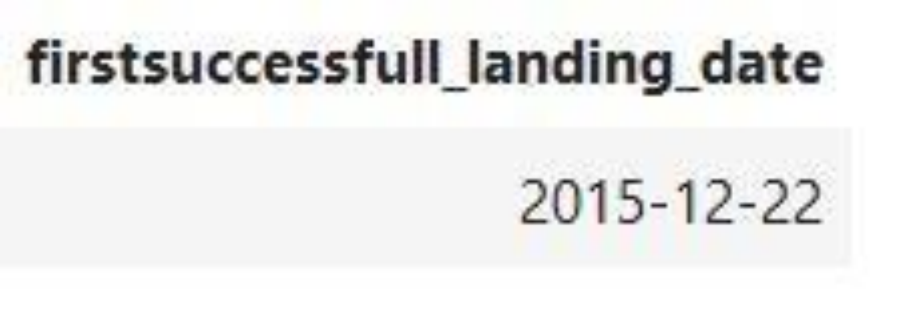
- Using SQL functions **AVG( ),** the average of mean payload carried by boosters F9v1.1 was calculated with more efficiency.



avg_payloadmass
2928

First Successful Ground Landing Date

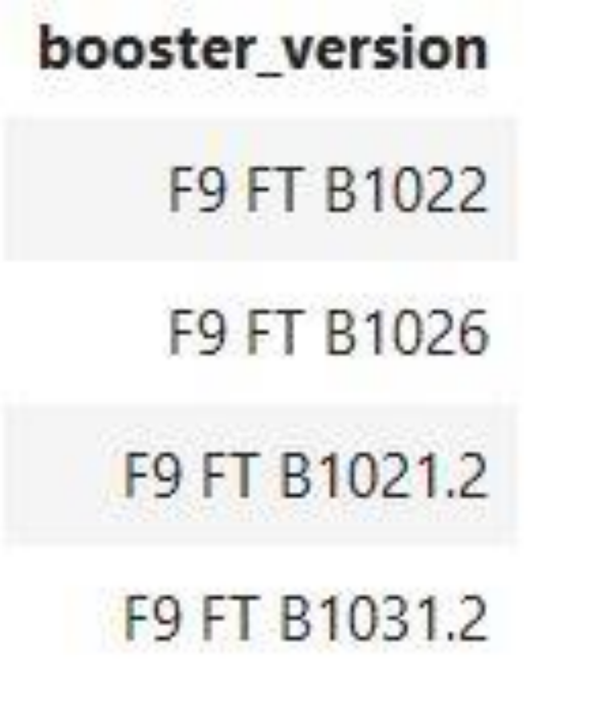
- The first successful ground landing date was calculated by using **MIN()** function on date , where the data was filtered for only successful landing on ground pad.



firstsuccessfull_landing_date
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

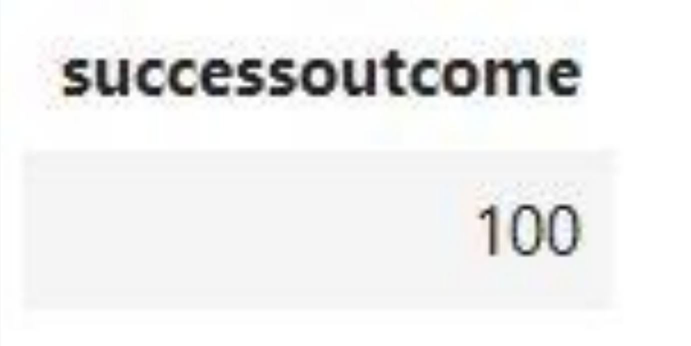
- **WHERE** clause was used to filter data for those boosters which met the condition of successful drone landing and **AND** operator was used to add condition of payload having between 4000 and 6000.



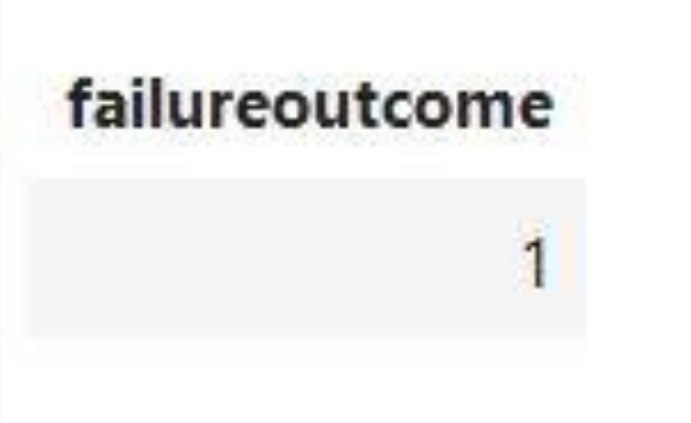
booster_version
F9 FT B1022
F9 FT B1026
F9 FT B1021.2
F9 FT B1031.2

Total Number of Successful and Failure Mission Outcomes

- Wildcard like '%'was used in the SQL query to filter Mission Outcome with the help of **WHERE** clause.



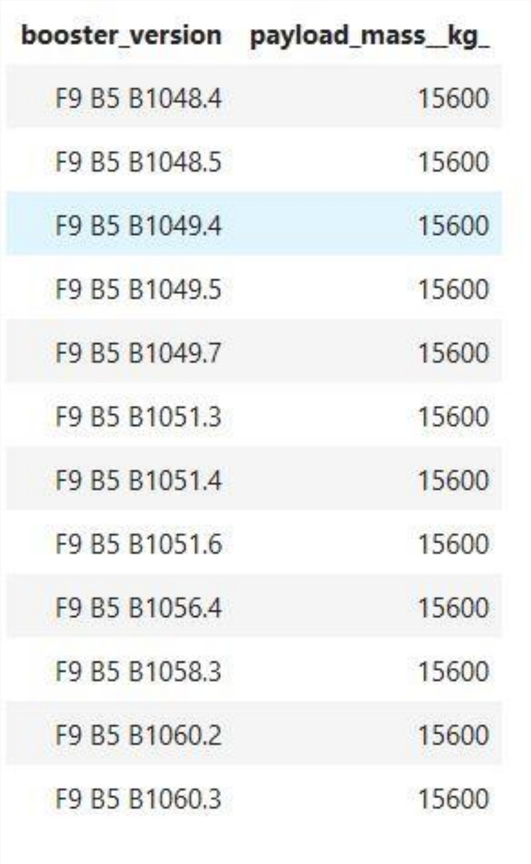
successoutcome
100



failureoutcome
1

Boosters Carried Maximum Payload

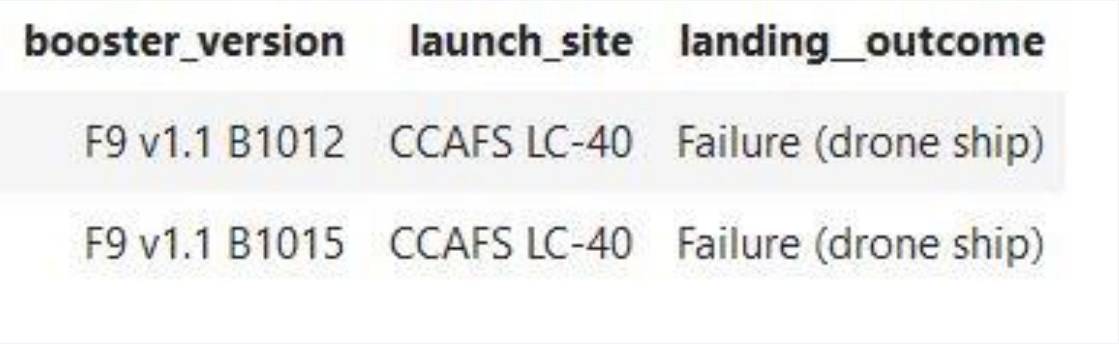
- Booster that carried maximum payload was determined by using subquery which had **WHERE** clause and **MAX()** function.



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

2015 Launch Records

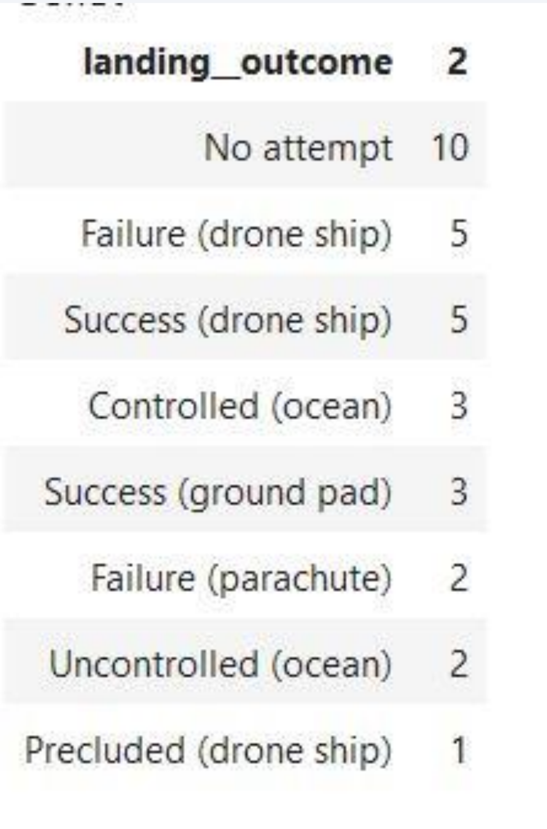
- A combination of clauses, operators such as **WHERE** clause, **LIKE**, **AND**, **BETWEEN** operators were used to filter out information about Booster Version, Launch Site and Landing Outcome from the year 2015.



booster_version	launch_site	landing__outcome
F9 v1.1 B1012	CCAFS LC-40	Failure (drone ship)
F9 v1.1 B1015	CCAFS LC-40	Failure (drone ship)

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

- Pivoting clause **GROUP BY** and **ORDER BY** as well as **COUNT()** function, **WHERE** clause and **BETWEEN** operator was used as a combination to filter and group data of Landing Outcomes from 2010/06/04 to 2017/03/20.



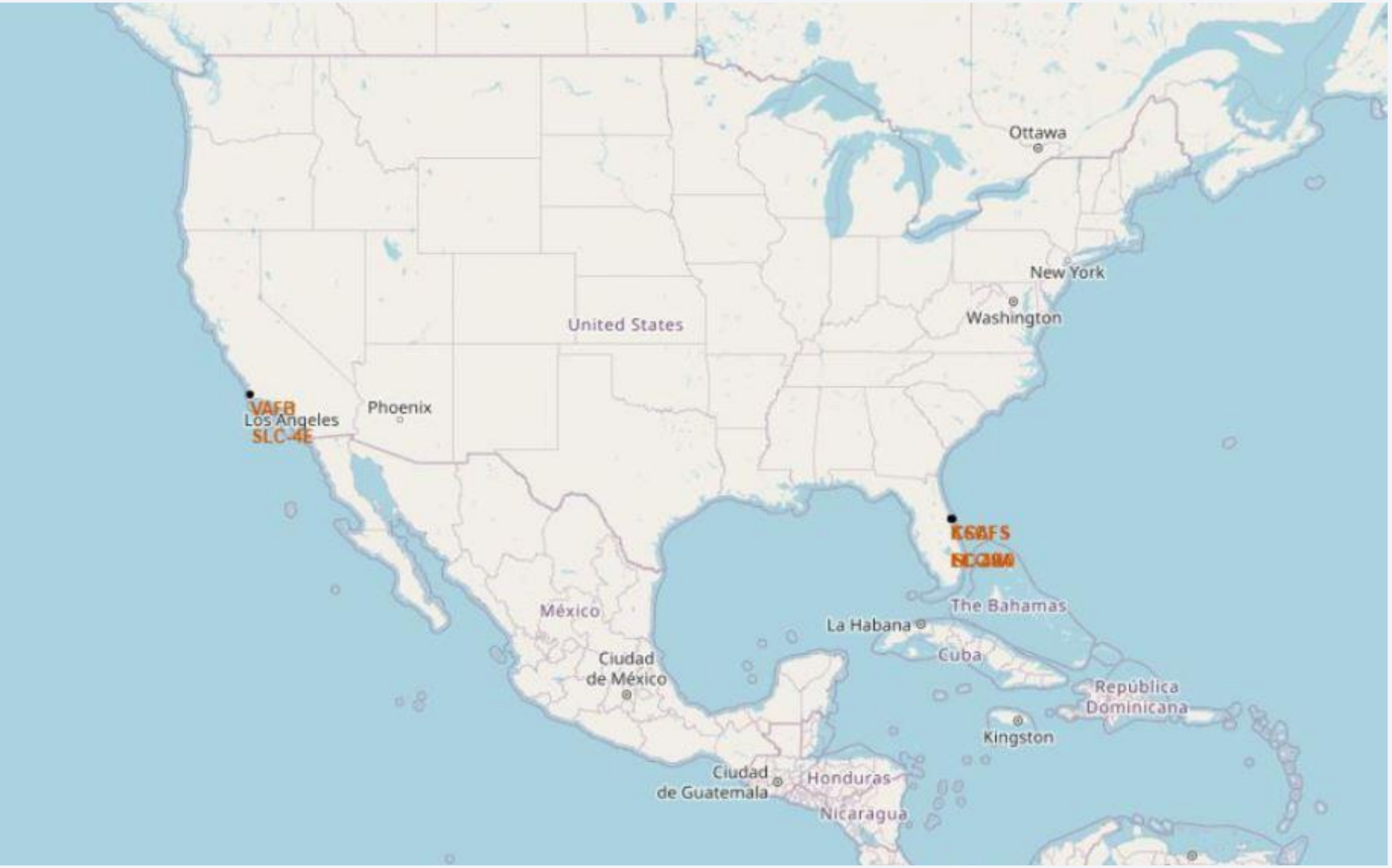
landing_outcome	2
No attempt	10
Failure (drone ship)	5
Success (drone ship)	5
Controlled (ocean)	3
Success (ground pad)	3
Failure (parachute)	2
Uncontrolled (ocean)	2
Precluded (drone ship)	1

Section 3

Launch Sites Proximities Analysis

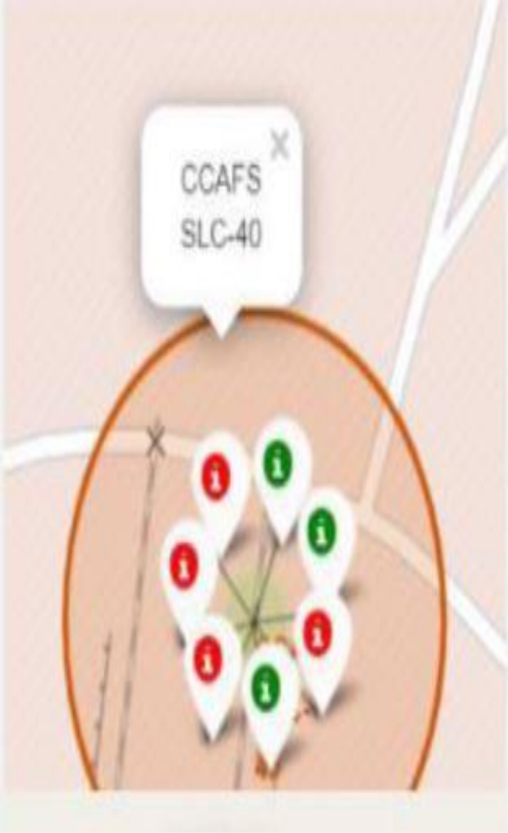
Launch Sites Location

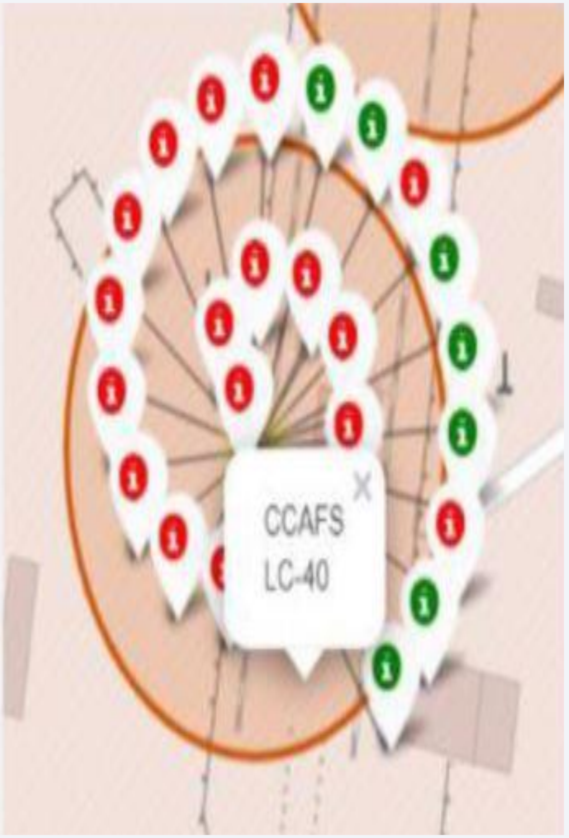
- There are two main locations for launch sites , one is east coast, Florida and another is west coast, California



Landing Outcomes in Launch Sites

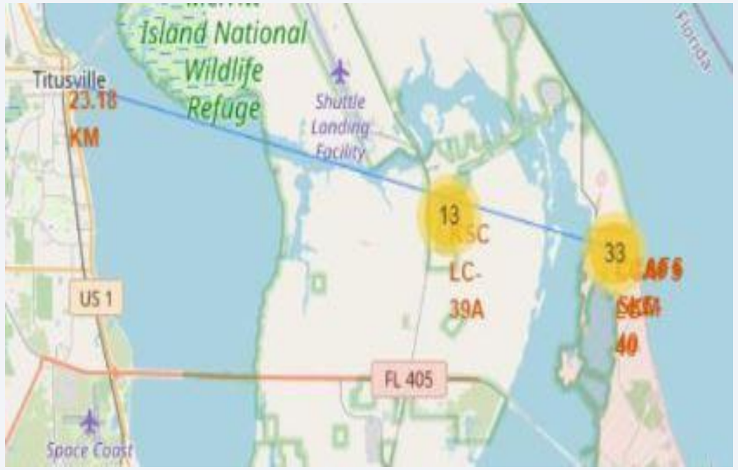
- These are the landing outcomes in CCAFS SLC-40 and CCAFS LC-40
- The red market indicates failed landing outcome, whereas green marker indicates successful landing outcomes.

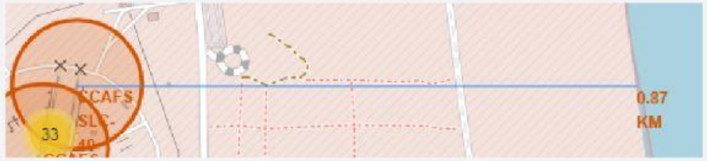


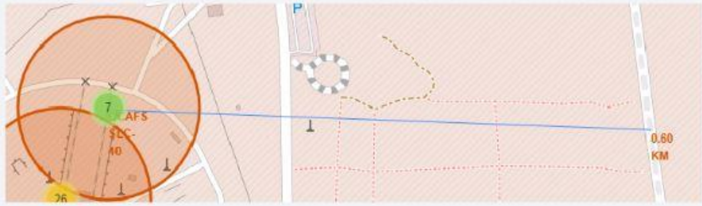


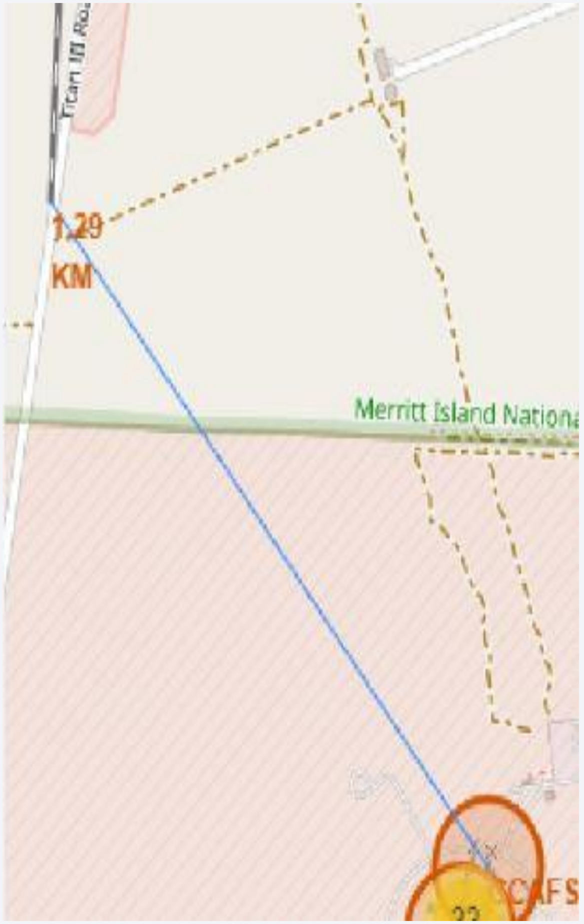
Distance between Launch Site and Civilization

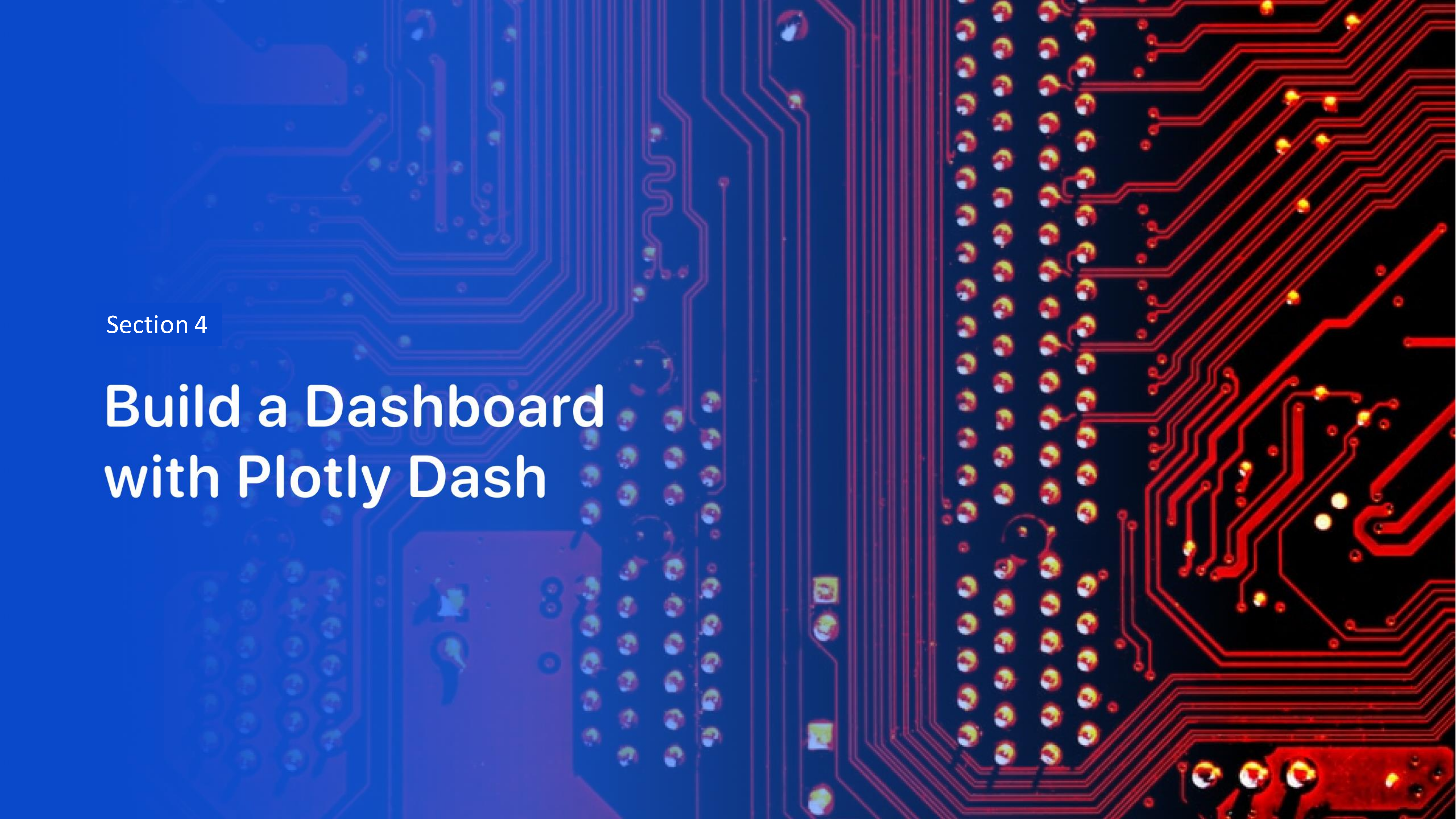
- This is Launch Site CCAFS SLC-40.
- Distance between nearest city from the launch site is 23.18 km.
- Distance between launch site and coastline is 0.87 km.
- Distance between launch site and highway is 0.60 km.
- Distance between launch site and railway is 1.29 km.







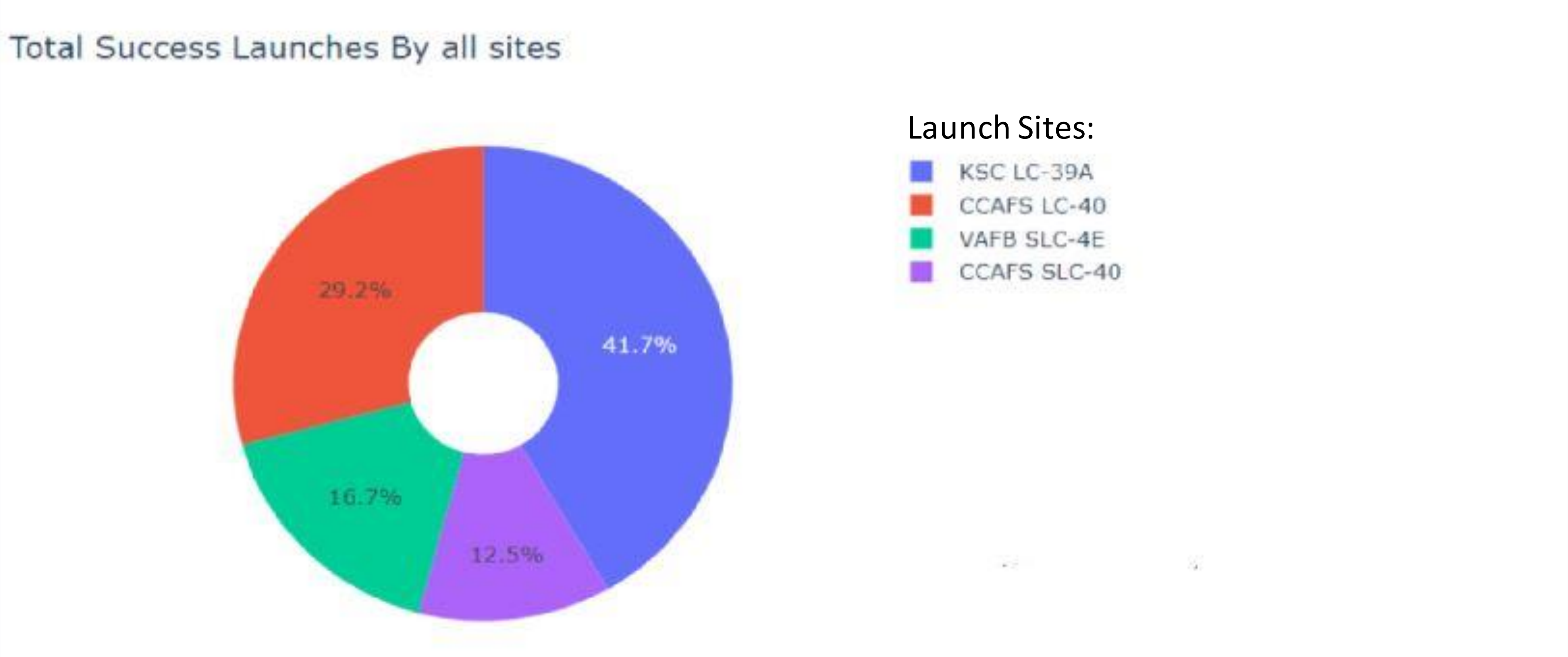




Section 4

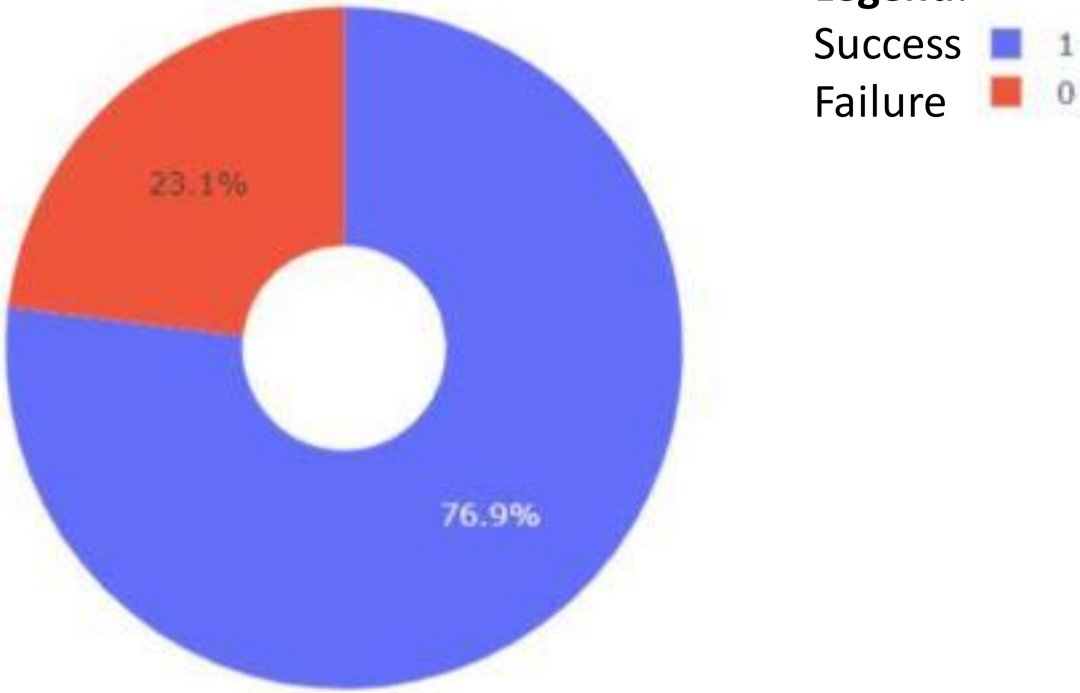
Build a Dashboard with Plotly Dash

KSC LC-39A has the highest success rate



Success Ratio in KSC LC-39A

The success rate in KSC LC-39A is 76.9%



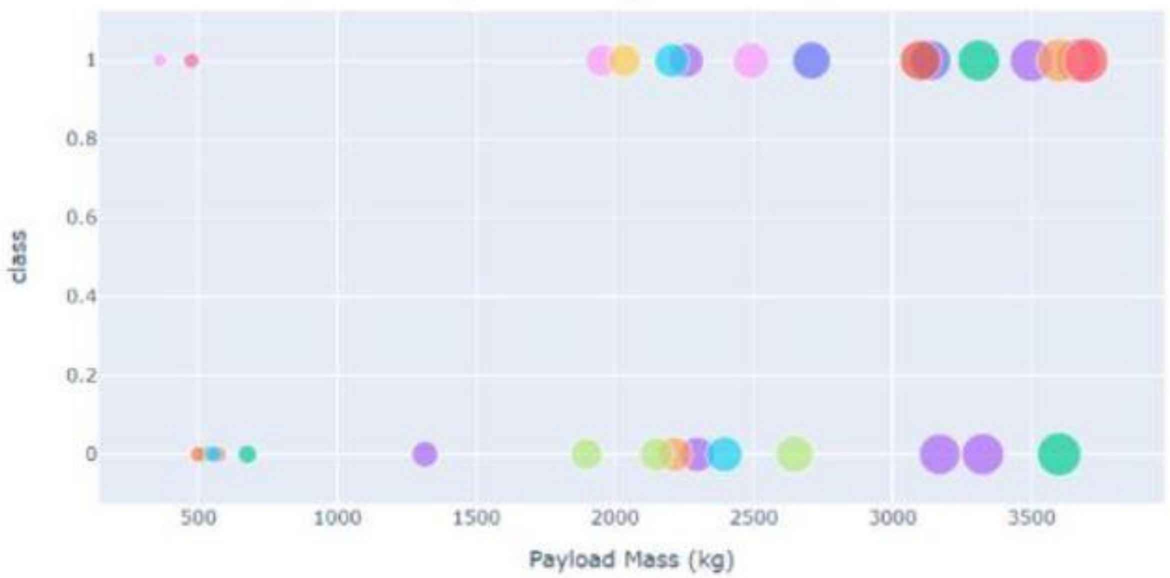
Legend:

Success 🟦 1

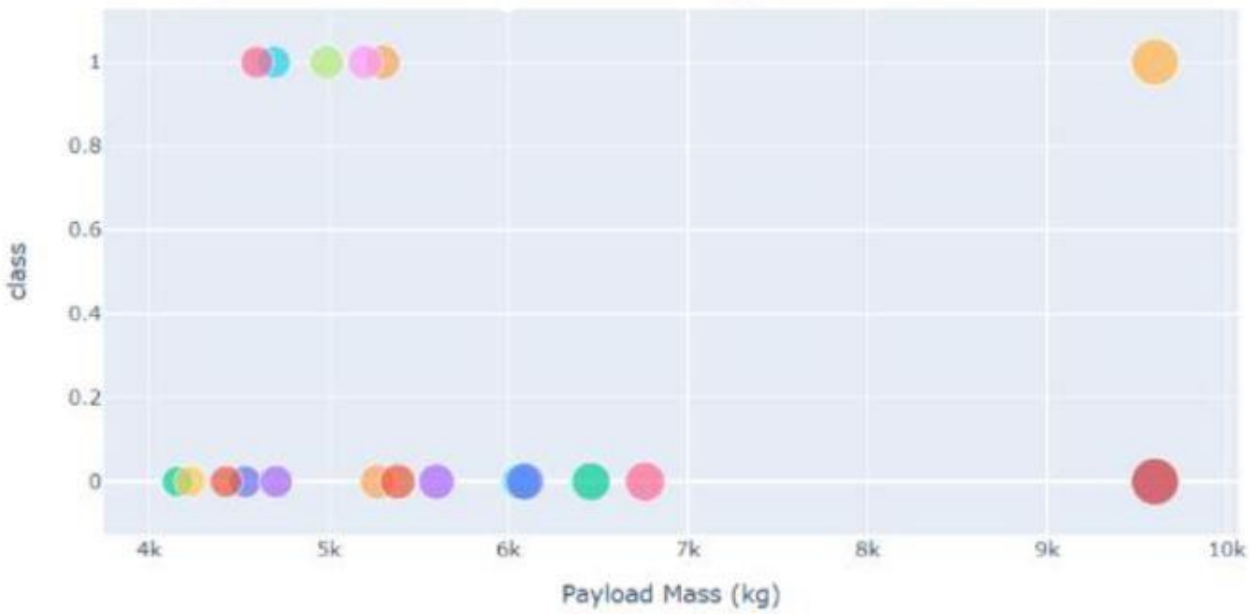
Failure 🟥 0

Scatterplot with Filter Slider

Payload from 0 to 4000 kg



Payload from 4000 to 10000 kg



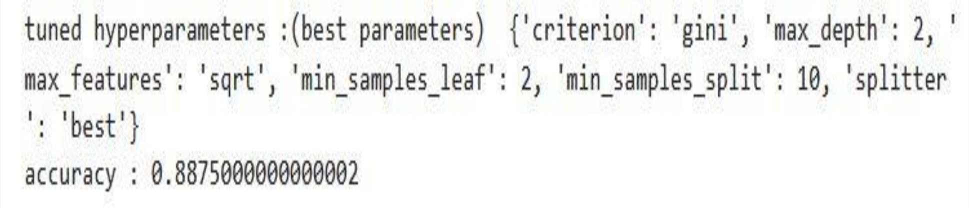
- The Plotly Dashboard consist of slider where the payload range could be filtered as per request

Section 5

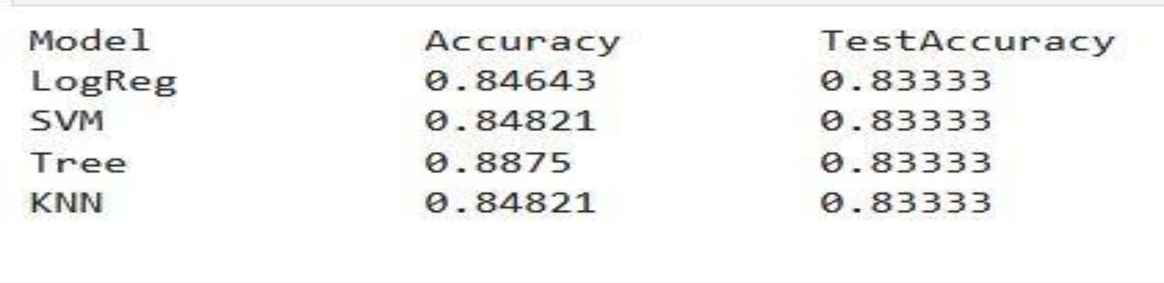
Predictive Analysis (Classification)

Classification Accuracy

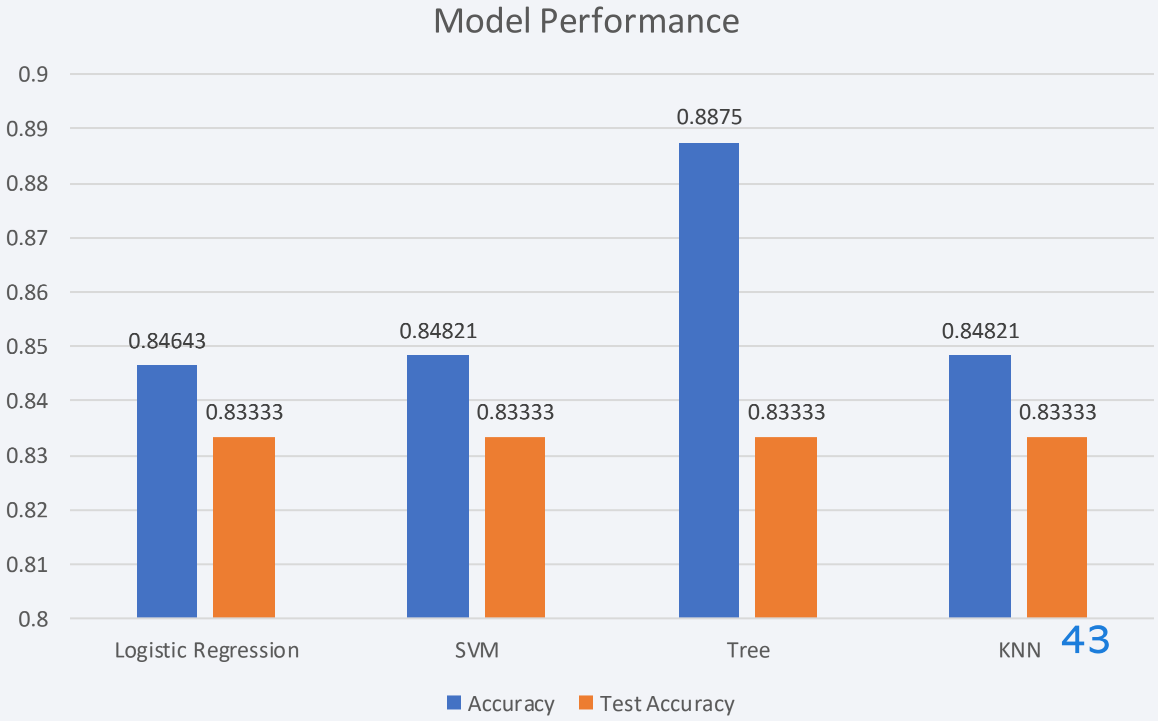
- Decision Tree classifier is the model with highest training accuracy, whereas the testing accuracy of every model is the equal.
- Decision Tree best parameters:



```
tuned hyperparameters :(best parameters)  {'criterion': 'gini', 'max_depth': 2, '
max_features': 'sqrt', 'min_samples_leaf': 2, 'min_samples_split': 10, 'splitter
': 'best'}
accuracy : 0.8875000000000002
```

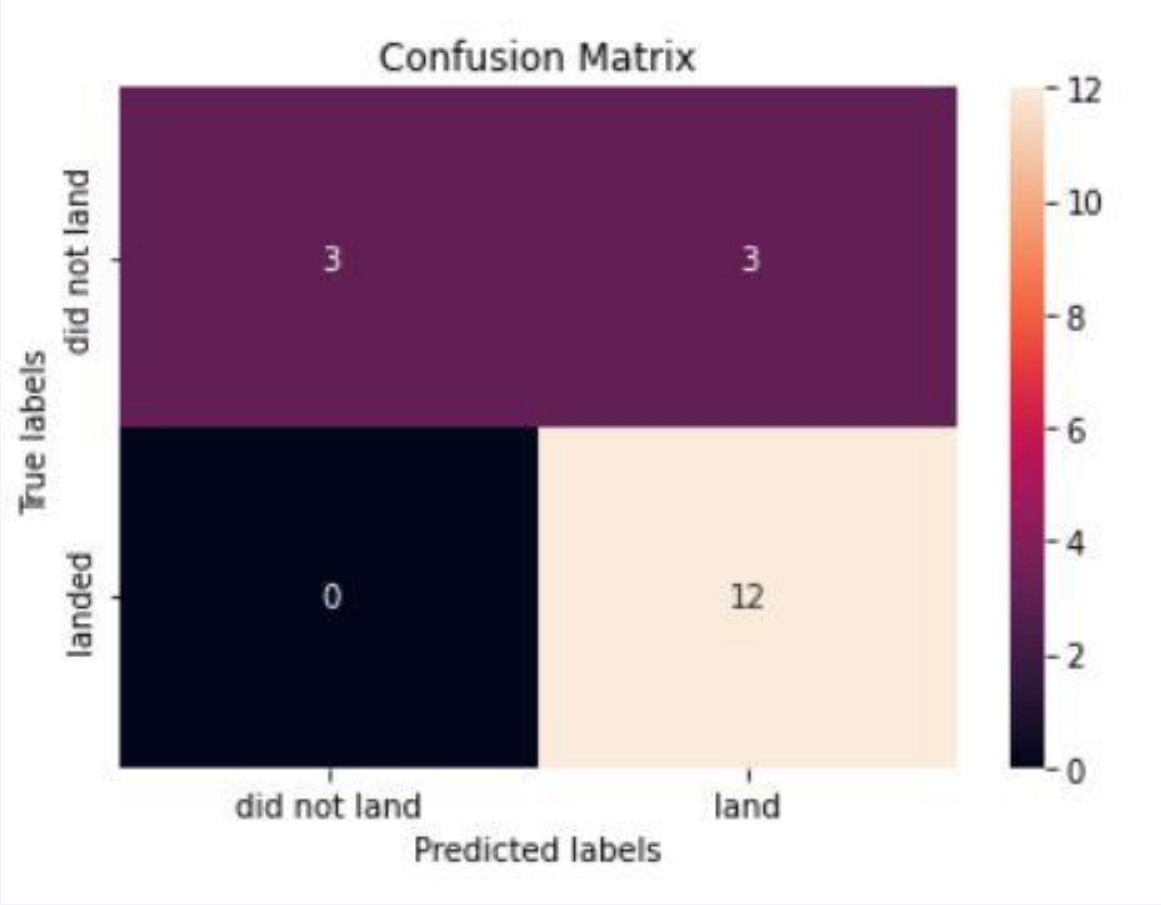


```
Model        Accuracy     TestAccuracy
LogReg       0.84643      0.83333
SVM          0.84821      0.83333
Tree         0.8875       0.83333
KNN          0.84821      0.83333
```



Confusion Matrix

- As all the model performed the same in the test data, the confusion matrix generated by all the models are same.
- The models have falsely predicted 3 result as land, but the real result should have been did not land.



Conclusions

- There are various factors that is responsible for successful landing of the rockets, one of them is number of flights. It can assumed that every flights taken has provided important feedback to improve rocket landing as the number of flight increases, so does successful landing.
- KSC LC-39A is the launching site with more successful landing than other sites., more research should be conducted in this site with relevant data to find the insight on making It the optimal launch site.
- Success rate is high in ES-L1, GEO, HEO, SSO and VLEO orbits.
- Decision tree is the best classification model for this dataset, as it has high training accuracy although test accuracy was the same between all models

Appendix

- The bar chart in page 43 was made using Microsoft Excel.
- Classification Model: A classification model tries to draw some conclusion from the input values given for training. It will predict the class labels/categories for the new data.
- Plotly: Ploty is an interactive, open-source, and browser-based graphing library for Python. It is MIT Licensed. Plotly graphs can be viewed in Jupyter notebooks, standalone HTML files, or integrated into Dash applications.
- Folium: Folium is a powerful Python library that helps you create several types of Leaflet maps. By default, Folium creates a map in a separate HTML file. Since Folium results are interactive, this library is very useful for dashboard building.
- API: An API is a set of programming code that enables data transmission between one software product and another. It also contains the terms of this data exchange.
- Beautiful Soup: Beautiful Soup is a Python package for parsing HTML and XML documents. It creates a parse tree for parsed pages that can be used to extract data from HTML, which is useful for web scraping

Thank you!