IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Sutirtha Mukherjee
07/07/2023

Outline

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

Executive Summary

Summary of methodology

Reuse of the first stage of a rocket can greatly reduce the cost of launching as shown by SpaceX. Using Data Science, here we predict if the Falcon 9 first stage will land successfully. First, the historical launch data are collected from SpaceX API and by web scrapping. Then, we perform exploratory data analysis to extract meaningful patterns and determine the important features to be used in success prediction. Finally we use different machine learning classification algorithms to build a predictive model for successful landing.

Summary of all results

From exploratory data analysis results, we get insights about how each important variable, like "payload mass", "launch site", and "orbit" etc., would affect the success rate. Interactive visual analytics show us how success rate varies with location and proximities of a launch site. Finally, from the machine learning computations, we obtain the predictive models with the best accuracy.

Introduction

- **Project background and context :**
  - If the first stage of a rocket is reused, it can greatly reduce the cost of launching. Space X advertises Falcon 9 rocket launches with a cost of 62 million dollars, whereas other providers cost upward of165 million dollars each. Much of the savings of Space X is because they 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 space X for a rocket launch.

- **Problems you want to find answers :**
  - The question that we want to answer is "If the first stage of a Falcon 9 rocket will land successfully?" Using Data Science methodology, we attempt to build a predictive model to answer the question.

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Data collected from the SpaceX REST API.
  - Data collected by web scrapping from Wikipedia.
- Perform data wrangling
  - Map the successful and unsuccessful mission outcomes to integers 1 and 0 respectively.
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Standardizing data, train-test splitting, and tuning hyperparameters for SVM, classification Trees, logistic regression, and KNN model.

Data Collection

- Describe how data sets were collected.
- SpaceX launch data is first gathered 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. The SpaceX REST API endpoints, or URL, starts with api.spacexdata.com/v4/, and we have the different end points.
- The historical launch records of Falcon rockets are collected also by web scraping from the Wikipedia page https://en.wikipedia.org/wiki/List_of_Falcon_9_and_Falcon_Heavy_launches.

Data Collection – SpaceX API

Flow Chart

Request and parse the SpaceX launch data using the GET request
static_json_url="https://cf-courses-data........"

Convert the json result into a dataframe
data=pd.json_normalize(response.json())

Some launch data are just IDs

Use the API to get information about the launch data that are just IDs, specifically using features: rocket, payloads, launchpad, and cores

Create a new pandas data frame with all the collected Launch data

Filter Dataframe to only include Falcon 9 launches

Dealing with Missing Values

GitHub URL
: https://github.com/Aapugithub/IBM_DataScienceCapston/blob/master/Data-collection-api-spacex-jupyterSM.ipynb

Data Collection - Scraping

Flow Chart

HTTP GET method to request the Falcon9 launch HTML page from its URL

↓

Create Beautiful Soup object to parse HTML

↓

Extract relevant column names from HTML table header

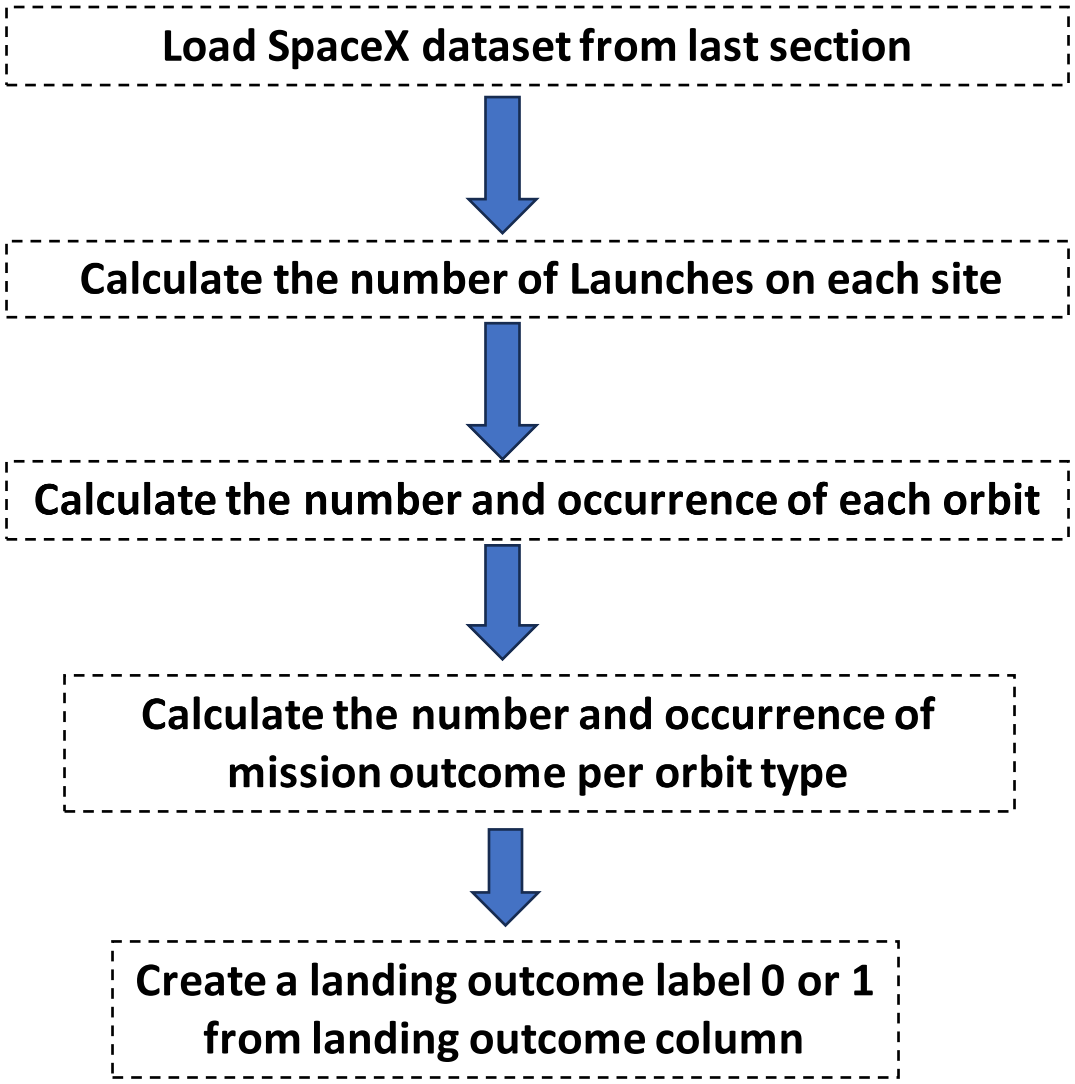
↓

Clean the data and create a data frame by parsing the launch HTML tables

GitHub URL :
https://github.com/Aapugithub/IBM_DataScienceCapston/blob/master/Webscraping-jupyter-labs-SM.ipynb

Data Wrangling

Flow Chart



GitHub URL: https://github.com/Aapugithub/IBM_DataScienceCapston/blob/master/Data_wranglingIBM-DS0321EN-SkillsNetwork_labs_module_1_L3_labs-jupyter-spacex_jupyterlite.jupyterlite%20(1).ipynb

EDA with Data Visualization

- Summarize what charts were plotted and why you used those charts
- Scatter Point Charts
  - Visualizing the relationship between Flight Number and Payload mass
  - Visualizing the relationship between Flight Number and Launch Site
  - Visualizing the relationship between Payload mass and Launch Site
  - Visualizing the relationship between Flight Number and Orbit type
  - Visualizing the relationship between Payload mass and Orbit type
- Bar Charts
  - Visualizing the relationship between success rate of each orbit type
- Line Chart
  - Visualizing the launch success yearly trend

GitHub URL : EDA with DATA_Visualization

EDA with SQL

- Displaying the names of the unique launch sites in the space mission.
- Displaying 5 records where launch sites begin with the string 'CCA'.
- 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 when the first successful landing outcome in ground pad was achieved.
- Listing the names of the boosters which have success in drone ship 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, failure landing_outcomes in drone ship ,booster versions, launch_site for the months in year 2015.
- Ranking 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.

GitHub URL:
[EDA with SQL]

Build an Interactive Map with Folium

- **Summarize what map objects such as markers, circles, lines, etc. you created and added to a folium map. Explain why you added those objects**
- folium.Circle
  - To locate each launch site on the folium map.
- folium.Marker
  - To locate each launch site on the folium map.
  - To indicate if the launch was successful or failed.
  - To display distance.
- MarkerCluster ( )
  - To simplify the map containing many markers having the same launch site coordinates.
  - The color-labeled markers in marker clusters help to identify the success rate of each site.
- MousePosition
  - To get coordinate for a mouse over a point on the map
- Folium.PolyLine
  - To draw a line between a launch site to its closest city, railway, highway, etc

GitHub URL : [Folium Map]()

Build a Dashboard with Plotly Dash

- **Summarize what plots/graphs and interactions you have added to a dashboard**
- Pie chart
  - To show launch success count for all sites.
  - To find the launch site with highest launch success ratio.
- Scatter plot
  - To show correlation between payload and success rate.
  - Tells us which payload range or booster version have the largest success rate.
- Drop-down Input Component
  - To let us select different launch sites.
- Range Slider
  - To select payload

GitHub URL : https://github.com/Aapugithub/IBM_DataScienceCapston/blob/master/spacex_dash_app.py

Predictive Analysis (Classification)

- **Summarize how you built, evaluated, improved, and found the best performing classification model**

  - Store dependent and independent variables separately.
  - Standardize the dependent variables.
  - Splitt the entire data into a training and a testing set.
  - Create machine learning classifier objects such as Logistic regression, SVM, Decision tree, and KNN and fit them with the training data set.
  - Use GridsearchCV to find the best hyperparameters for those classifiers.
  - Calculate the accuracy of the models on the test data with the help of confusion matrix and other relevant error measurement metrics.
  - Choose the model with highest accuracy.

Github URL : [Predictive Analysis]

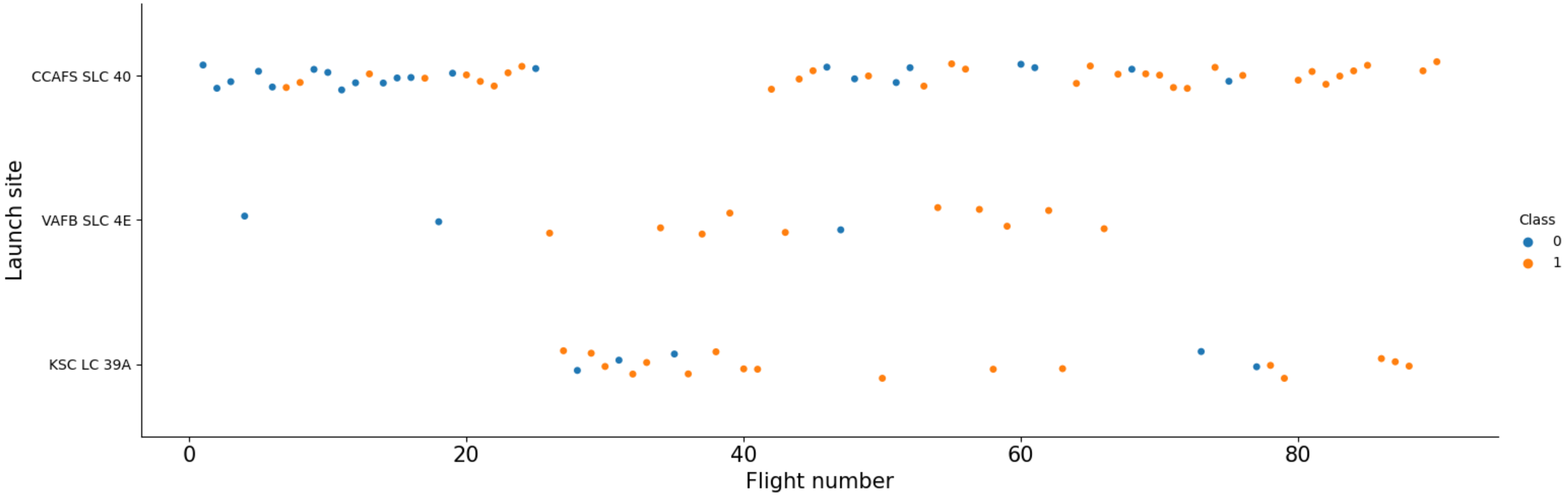
Results

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

Section 2

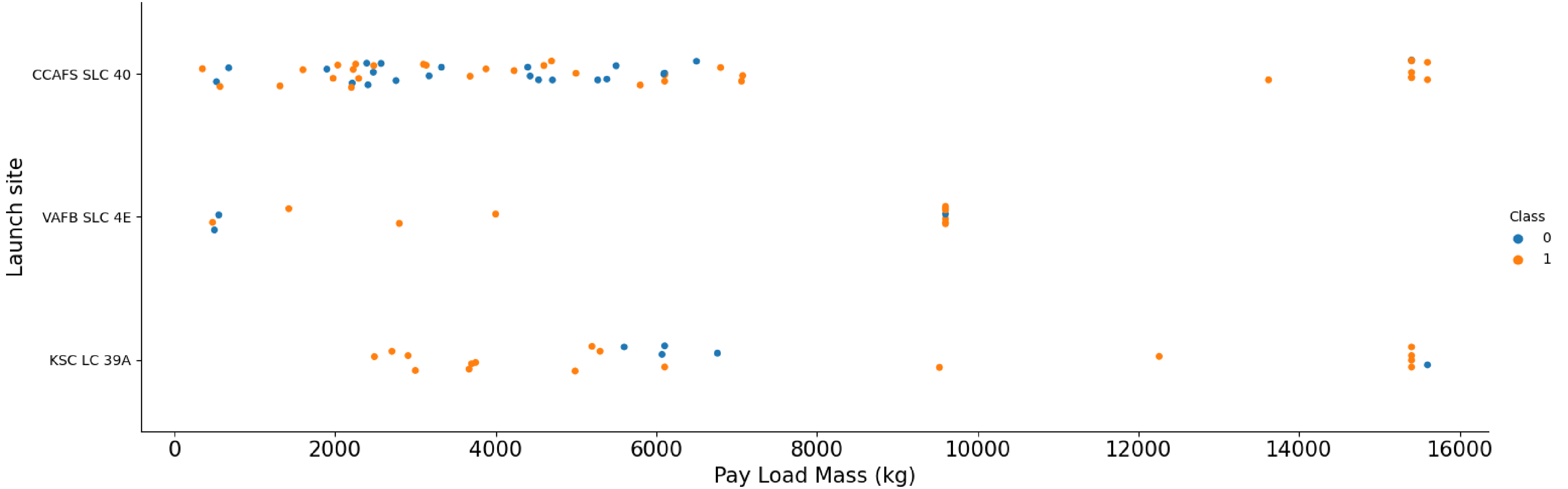
Insights drawn from EDA

Flight Number vs. Launch Site



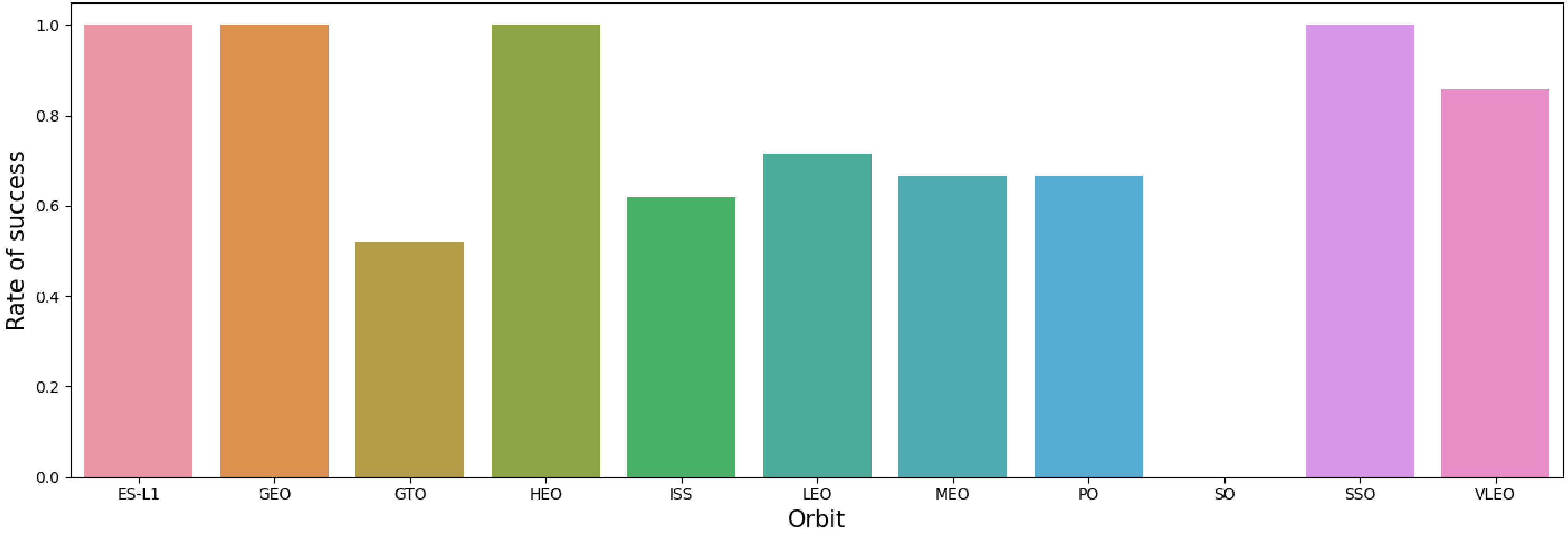
- For all the Launch sites, the success rate increases with increasing flight number.
- KSC LC 39 A was not used for roughly first 30 flights, VAFB SLC 4E was not used for roughly last 30 flights, and in the middle, CCAFS SLC 40 was not used for roughly 20 flights.

Payload vs. Launch Site



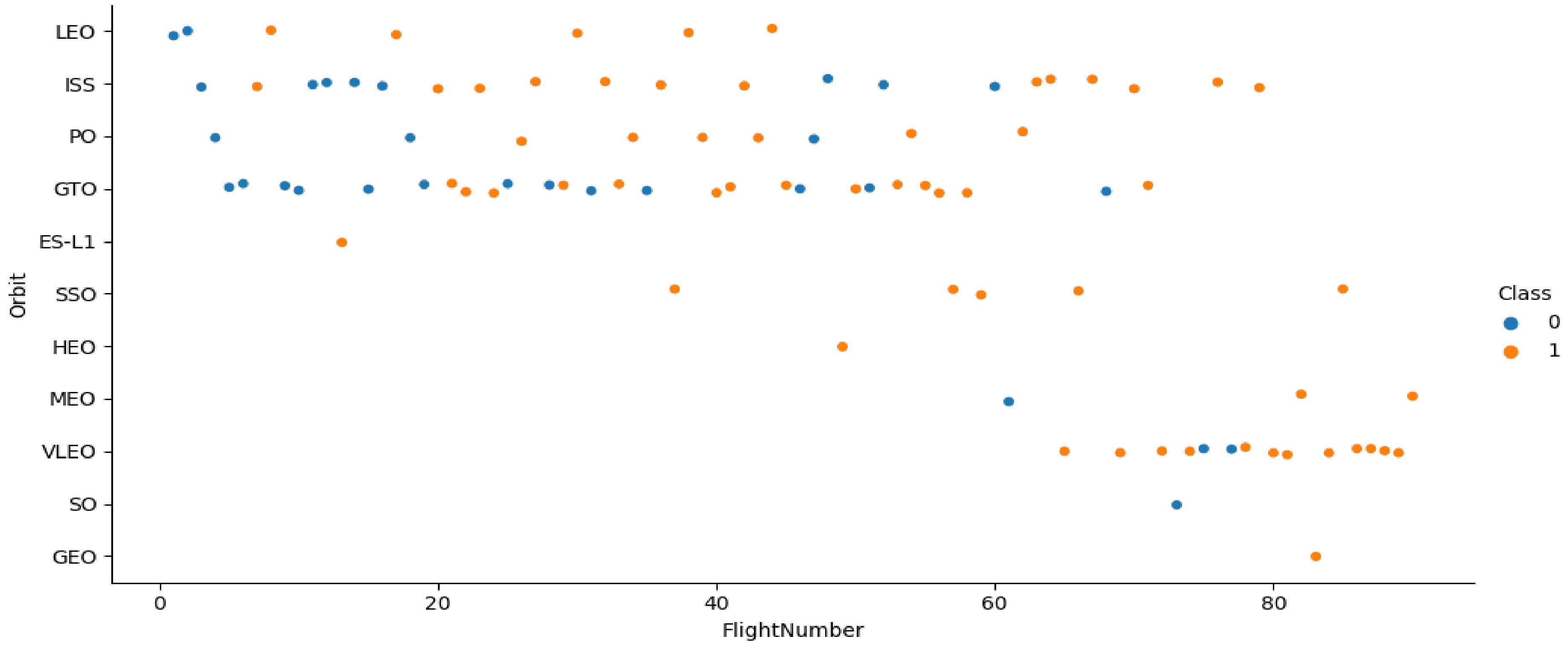
- For the VAFB-SLC launch site, there are no rockets launched for heavy payload mass(greater than 10000).
- Rockets with payload mass greater than 8000 has very high success rate.

Success Rate vs. Orbit Type



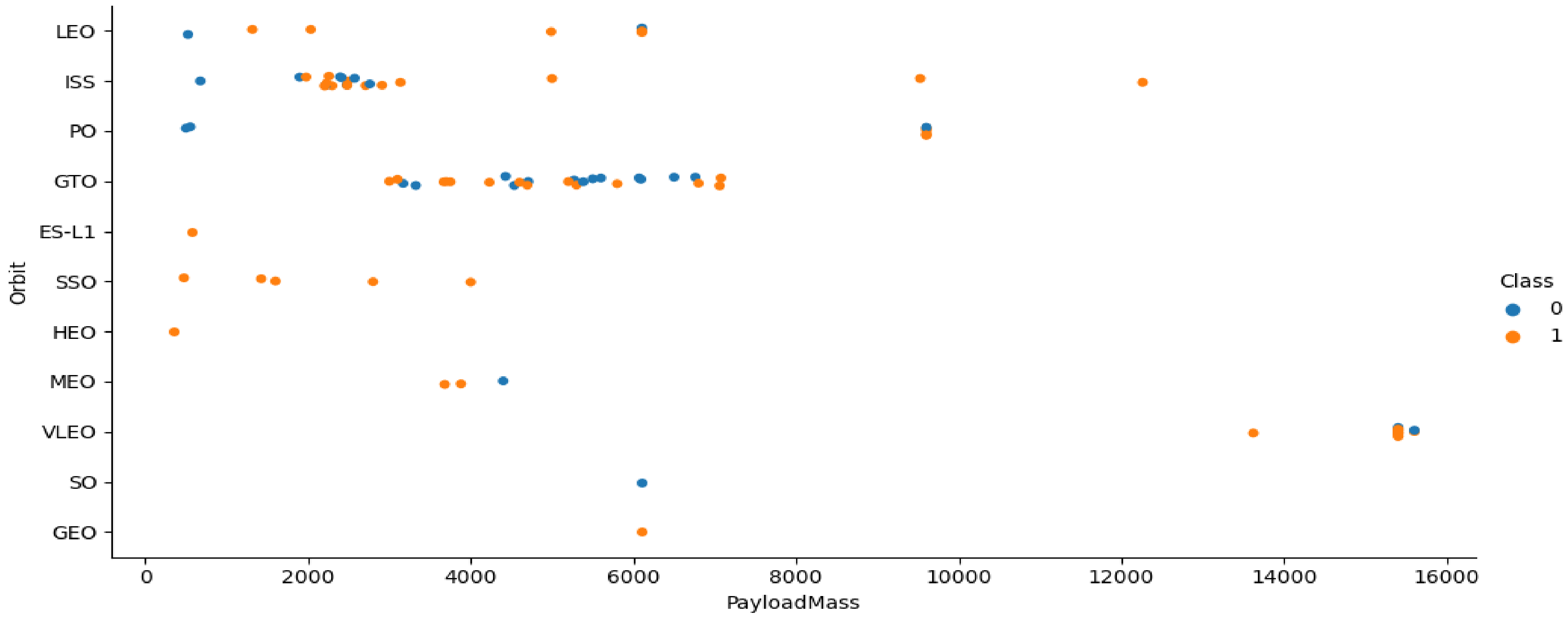
- The orbits ES-L1, GEO, HEO, and SSO has cent per cent success rate.

Flight Number vs. Orbit Type



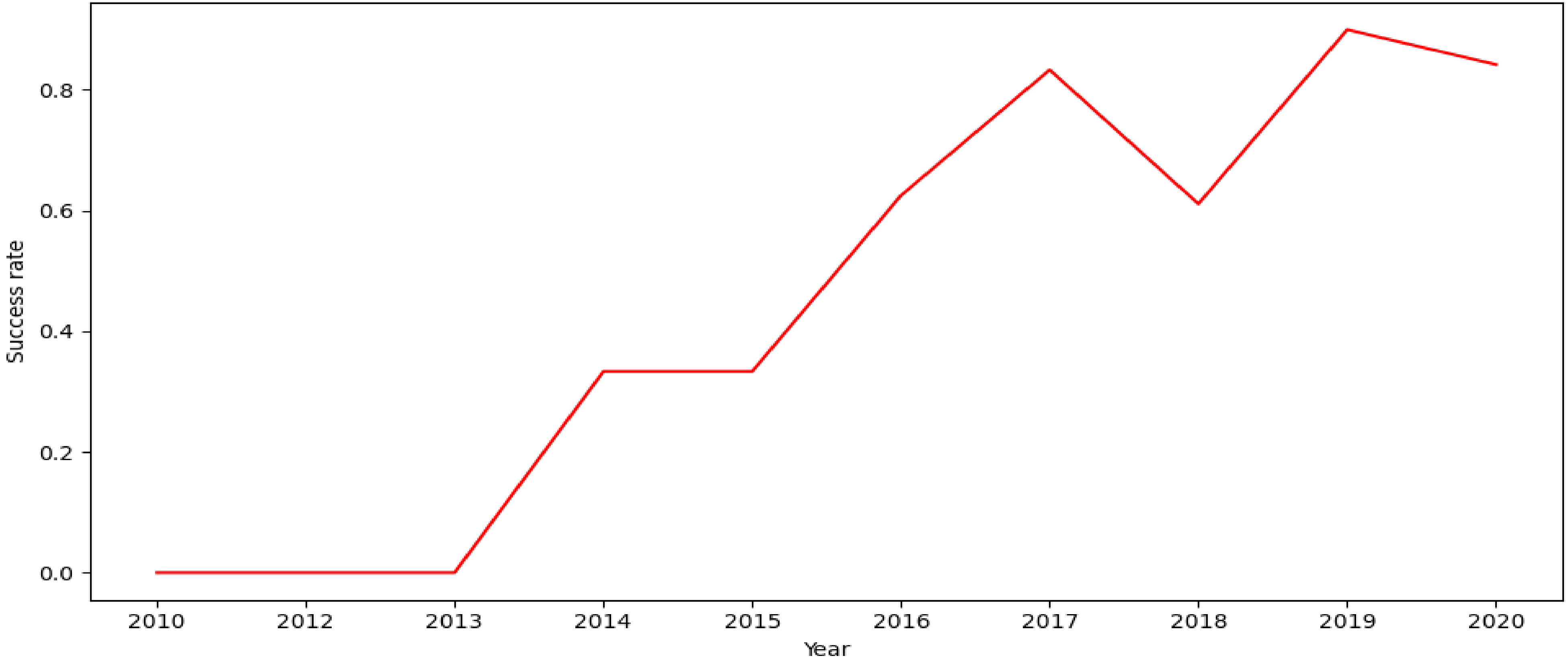
- In the LEO orbit the Success appears related to the number of flights; on the other hand, there seems to be no relationship between flight number when in GTO orbit.

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 here.

Launch Success Yearly Trend



- The success rate since 2013 kept increasing till 2020.

All Launch Site Names

- **Find the names of the unique launch sites**

- There are four unique launch sites as shown in the table.

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

Launch Site Names Begin with 'CCA'

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
06/04/2010	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0.0	LEO	SpaceX	Success	Failure (parachute)
12/08/2010	15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight C1, two CubeSats, barrel of Brouere cheese	0.0	LEO (ISS)	NASA (COTS) NRO	Success	Failure (parachute)
22/05/2012	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525.0	LEO (ISS)	NASA (COTS)	Success	No attempt
10/08/2012	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500.0	LEO (ISS)	NASA (CRS)	Success	No attempt
03/01/2013	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677.0	LEO (ISS)	NASA (CRS)	Success	No attempt

- The launch sites with name begin with 'CCA' are CCAFS LC 40 and CCAFS SLC 40. The table shows first 5 records of such launch sites.

Total Payload Mass

- The total payload carried by boosters from NASA only is 45596 KG.

Total_payload_mass_NASA (KG)
45596.0

Average Payload Mass by F9 v1.1

- The average payload mass carried by booster version F9 v1.1 is 2928.4 KG.

AVG PAYLOAD MASS (KG)
2928.4

First Successful Ground Landing Date

Date
22/12/2015

- The first successful landing outcome on ground pad was achieved on 22/12/2015

Successful Drone Ship Landing with Payload between 4000 and 6000

- There are 4 boosters which have successfully landed on drone ship and had payload mass greater than 4000 but less than 6000, as shown in the table.

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

Total Number of Successful and Failure Mission Outcomes

- There are total 100 successful and just 1 failure mission outcomes.

Total_successful	Total_failure
100	1

Boosters Carried Maximum Payload

- The table shows the list of the names of the booster which have carried the maximum payload mass which is 15600 KG.

Booster_Version	PAYLOAD_MASS (KG)
F9 B5 B1048.4	15600.0
F9 B5 B1049.4	15600.0
F9 B5 B1051.3	15600.0
F9 B5 B1056.4	15600.0
F9 B5 B1048.5	15600.0
F9 B5 B1051.4	15600.0
F9 B5 B1049.5	15600.0
F9 B5 B1060.2	15600.0
F9 B5 B1058.3	15600.0
F9 B5 B1051.6	15600.0
F9 B5 B1060.3	15600.0
F9 B5 B1049.7	15600.0

2015 Launch Records

- The table shows the 2 failed landing outcomes in drone ship, their booster versions, and launch site names for in year 2015

Date	Month	Landing_Outcome	Booster_Version	Launch_Site
01/10/2015	10	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
14/04/2015	04	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

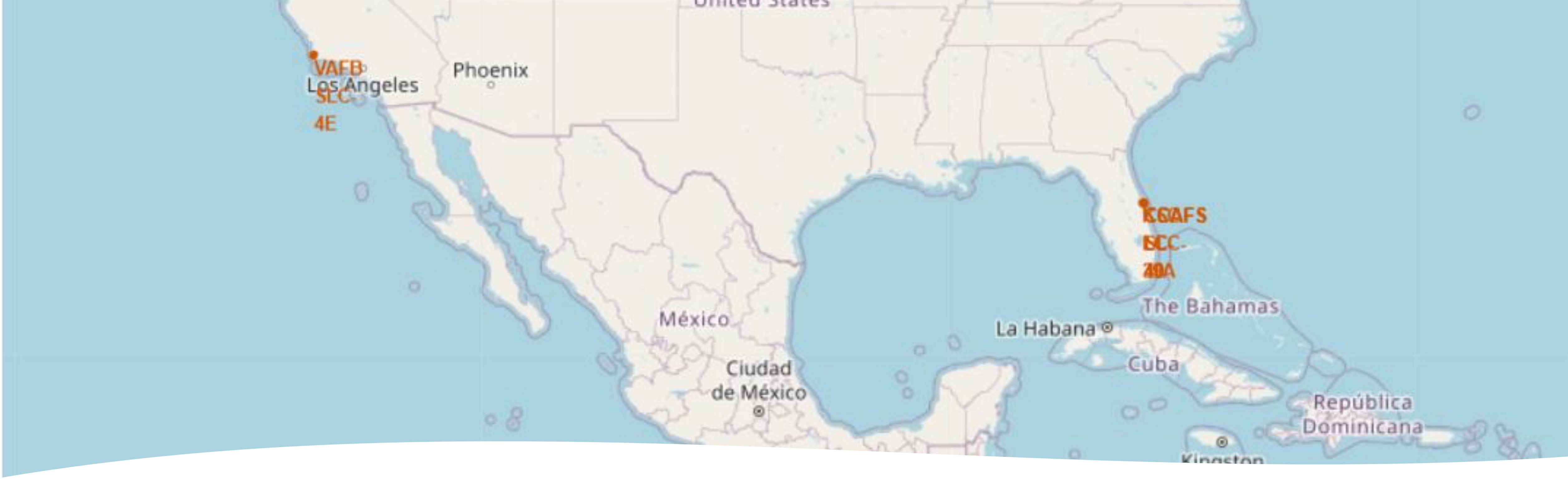
Rank Landing Outcomes Between 2010-06-04 and 2017-03-20

- The table shows 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.

Date_str	Landing_Outcome	Count_outcomes
20120522	No attempt	10
20151222	Success (ground pad)	5
20160804	Success (drone ship)	5
20151001	Failure (drone ship)	5
20140418	Controlled (ocean)	3
20130929	Uncontrolled (ocean)	2
20150628	Precluded (drone ship)	1
20100812	Failure (parachute)	1

Section 3

Launch Sites Proximities Analysis



Launch Sites' Locations on Folium map

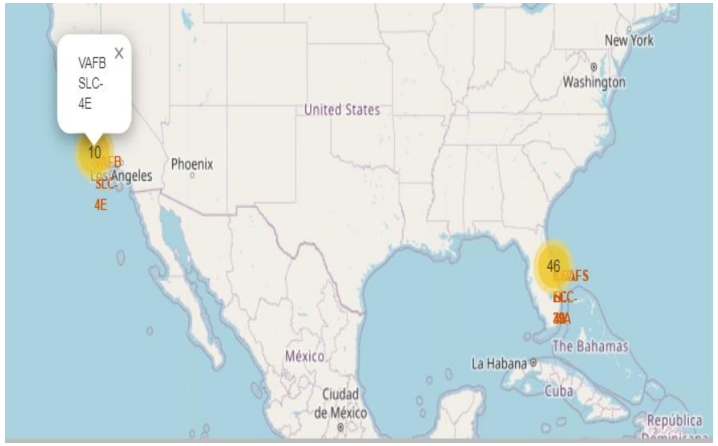
- The figure shows the generated folium map that includes all launch sites' location markers on a global map.
- The launch sites are concentrated mainly in two regions.
- All launch sites are in very close proximity to the coast.

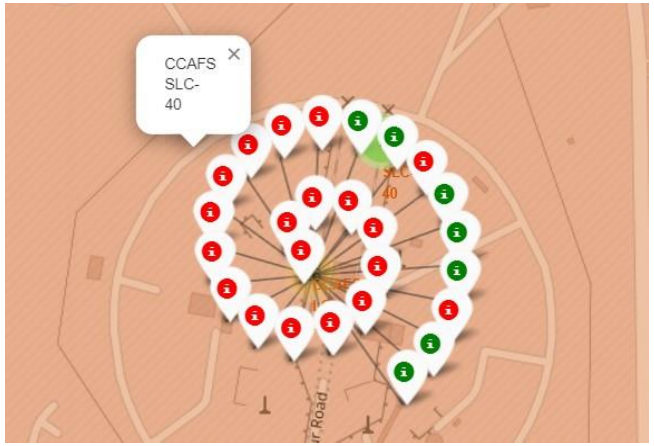
Launch outcomes on map

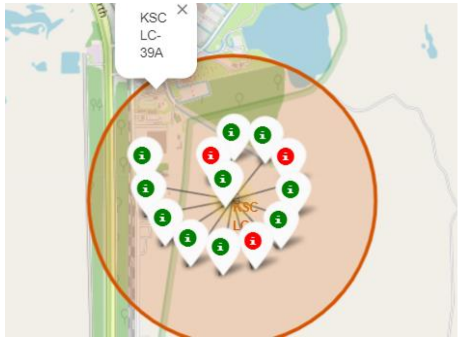
- Figure shows the color-labeled launch outcomes on the folium map.

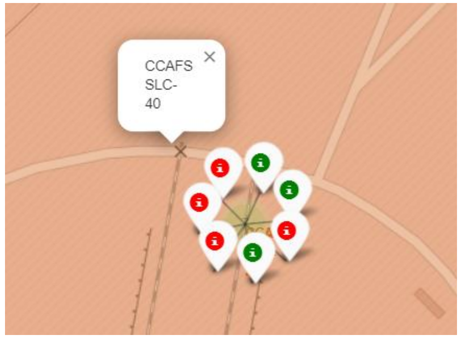
- The numbers represent the total launches happened on the sites. The green and red are for success and failure respectively.

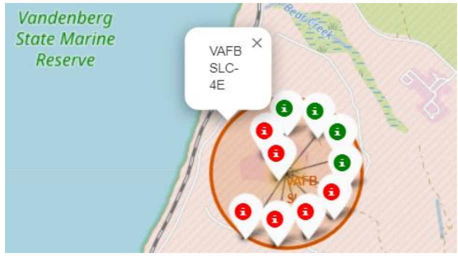
- KSC LC-39A site has the highest success rate.



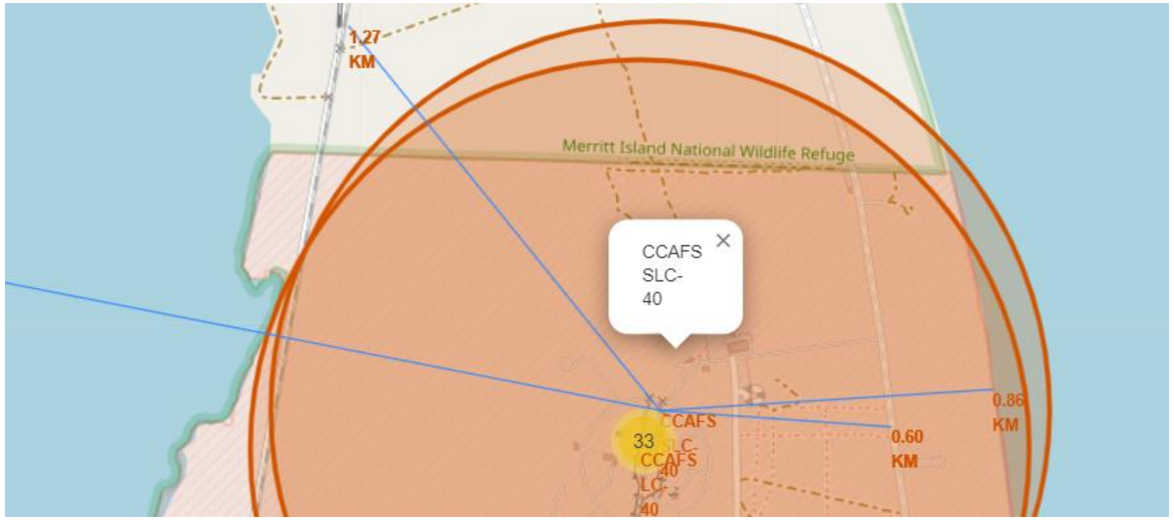




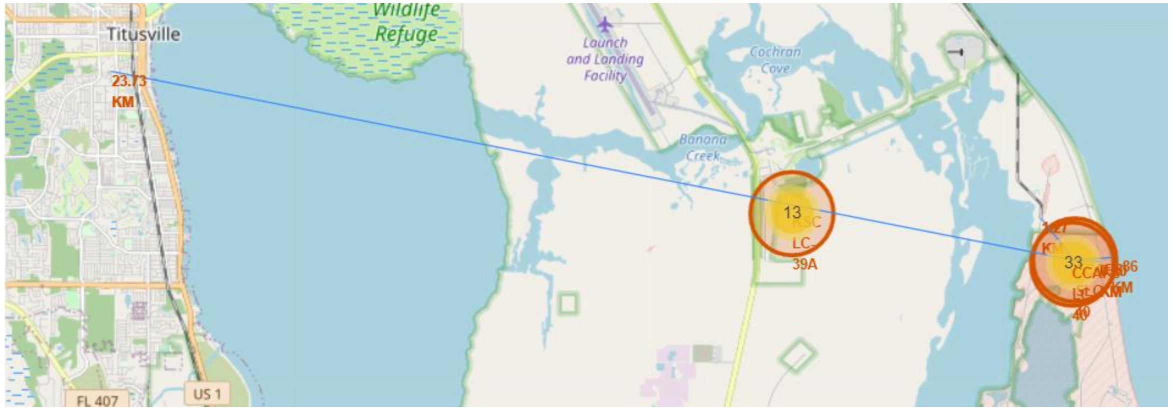




Launch site CCAFS SLC-40 to its proximities

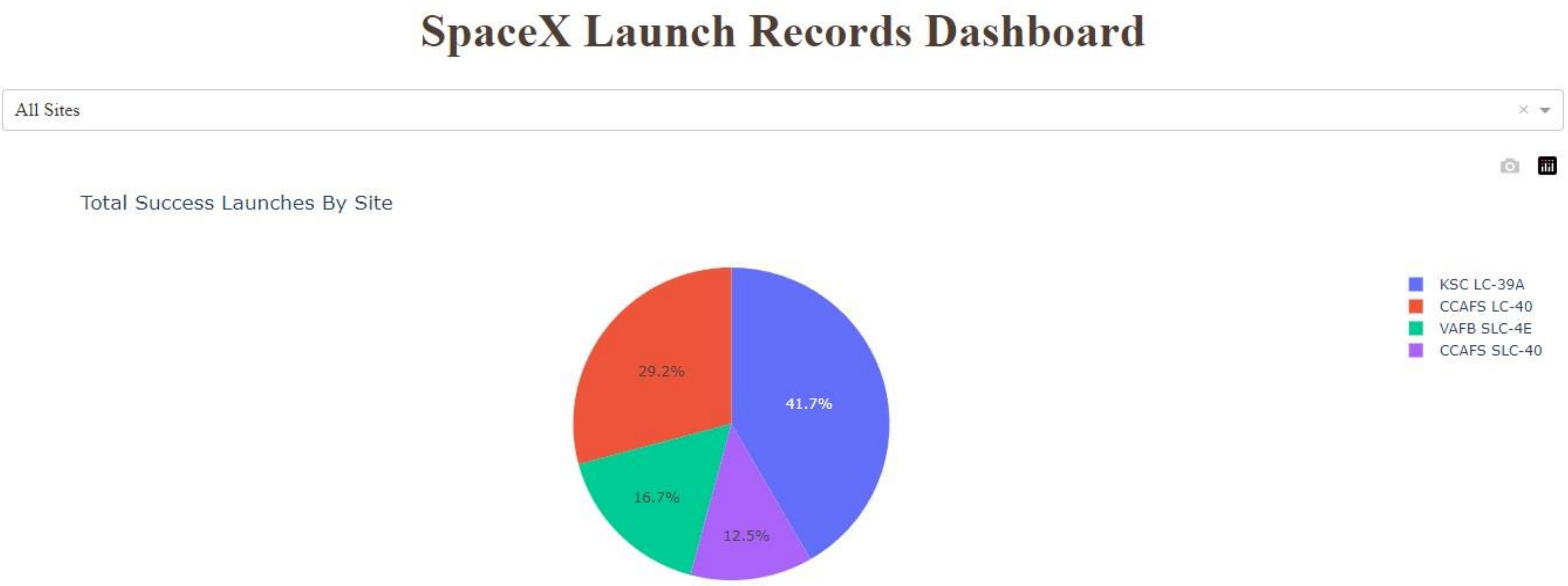


- Figure shows the launch site CCAFS SLC-40 to its proximities such as city, railway, highway, coastline, with distance calculated and displayed.
- The distance of the nearest railway is roughly 1.3 Km, highway is 0.6 Km, coastline is 0.9 Km, and of the nearest city is 24 Km.
- The railway, highway and coastline are in close proximity, whereas the city is far away.



Section 4

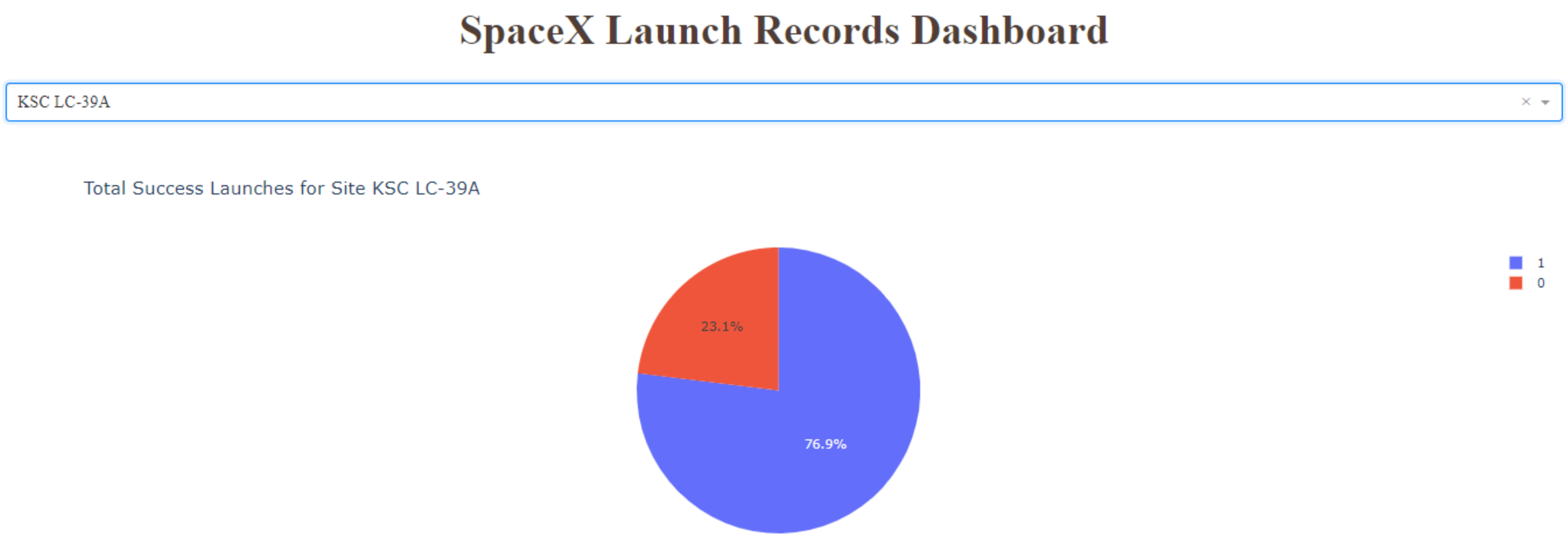
Build a Dashboard with Plotly Dash



Pie chart for launch success count of all sites

- Figure shows the pie chart for launch success count of all sites.
- KSC LC-39A site has the highest success count of 41.7 %.

Pie chart for launch site with highest success ratio



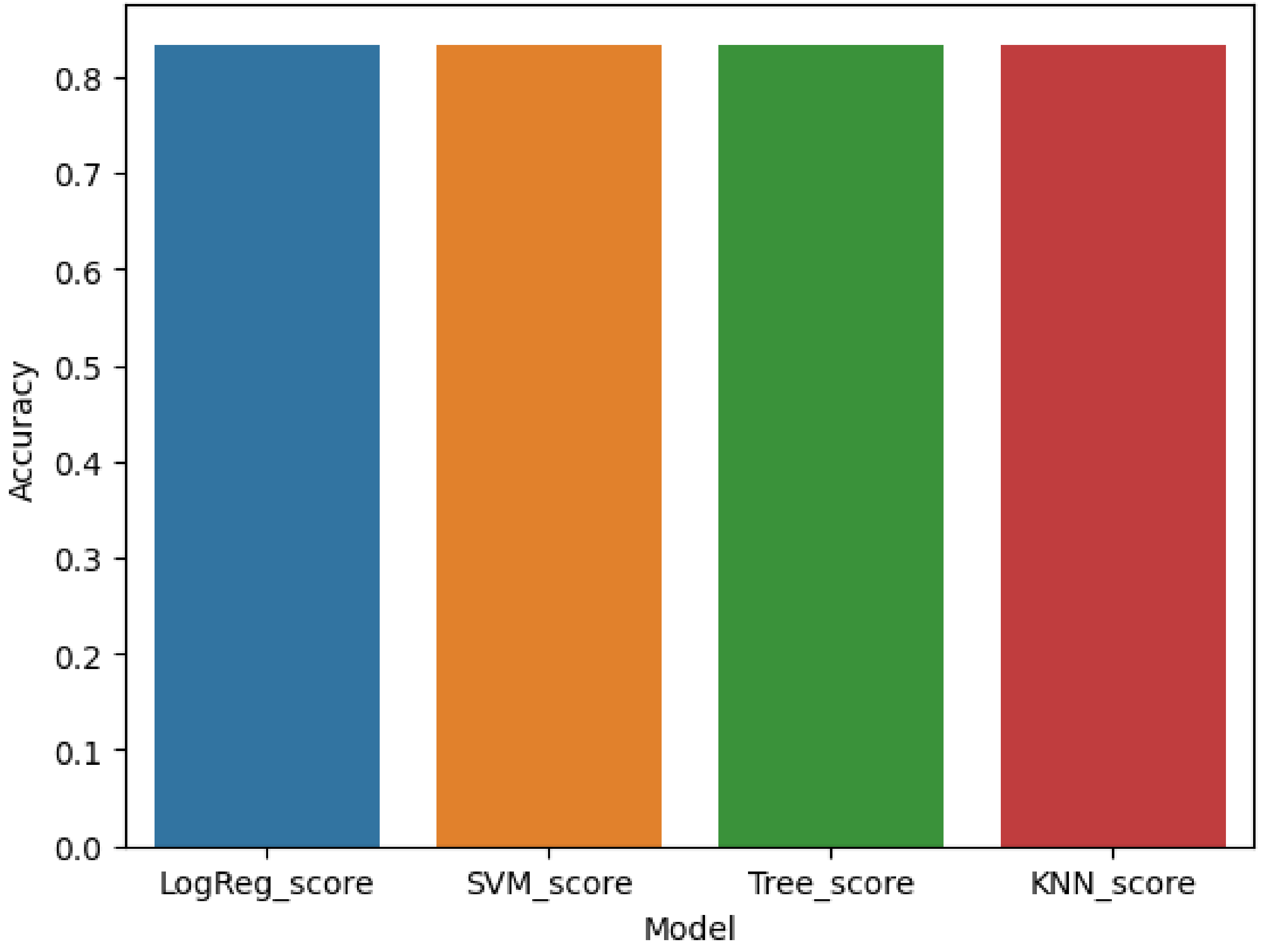
- Figure shows the pie chart for the launch site KSC LC-39A, which has the highest launch success ratio.
- This site has achieved almost 77 % successful launchings.

Payload vs Launch Outcome

- Figure shows the screenshots of Payload vs. Launch Outcome scatter plot for all sites, with different payload selected in the range slider.
- Most of the successful launching was achieved with payload in the betwteen 2000 Kg to 6000 Kg.
- FT and B4 boosters performed very well.

Section 5

Predictive Analysis (Classification)



Classification Accuracy

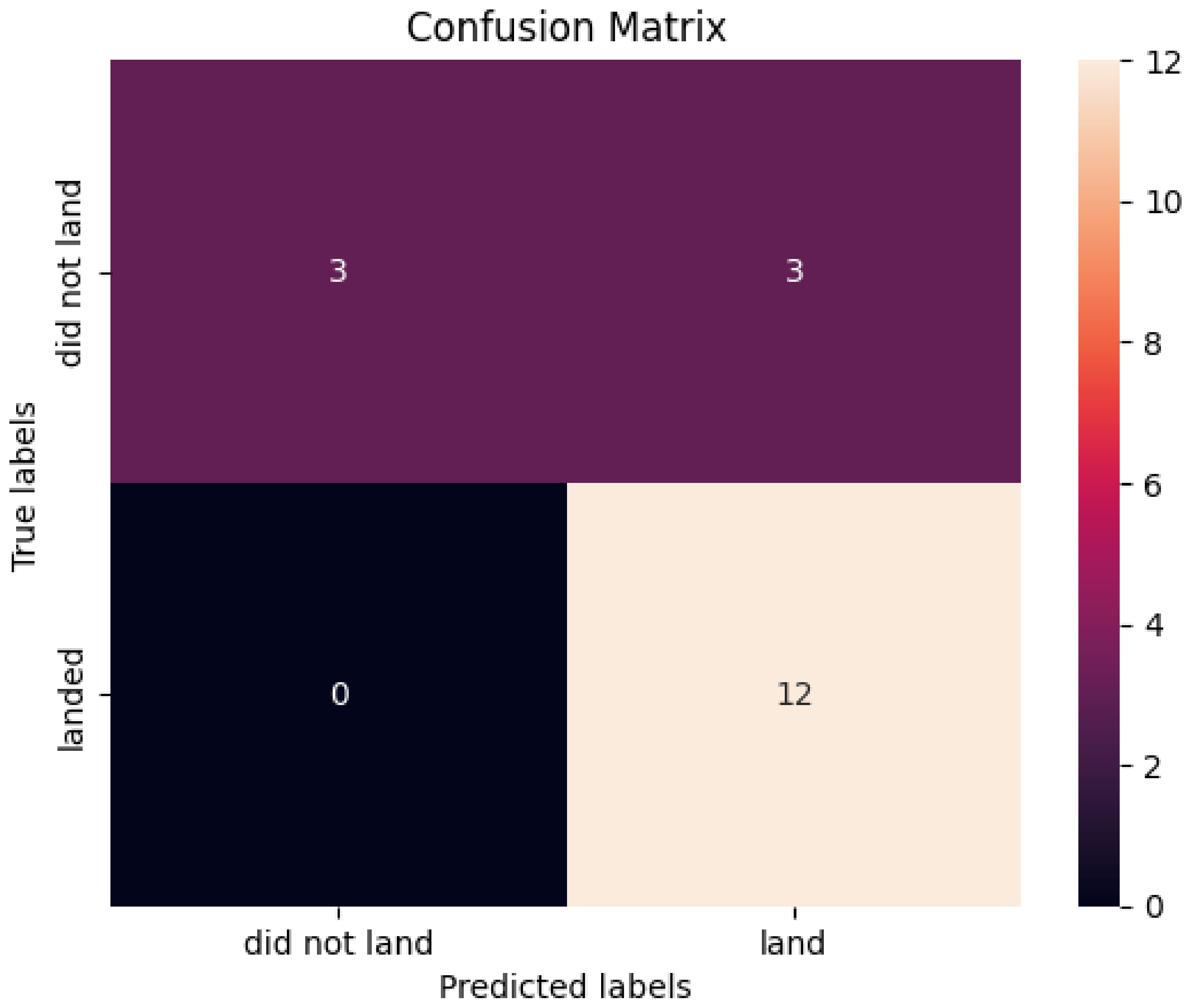
- Figure shows the built model accuracy, in a bar chart, for the classification models : Logistic regression, SVM, Decision Tree, and KNN classifier.

- All the models have the same classification accura cy 0.83.

Confusion Matrix

- Figure shows the confusion matrix, which is the same for all the built models.

- The models perfectly predict the successful landings, but has a problem in predicting failed landings because of the high false positive value.



Conclusions

- Launch site, orbit, payload mass, booster version, landing pad etc., are the important features in determining the success of a launch, and thus useful in constructing predictive models.
- From maps, we find that the launch sites are in close proximity with railway, highway, and coastline, but far away from cities.
- Some rough insights can be drawn from EDA, such as :
  - The launch success rate is increasing with time.
  - Heavier rockets launched for Polar, LEO and ISS orbits have higher success rate.
  - KSC LC39A site and FT and B4 boosters perform better, etc.

However, the actual correlation between the variable is much more intricate, and thus we need to rely on machine learning models.

- Since this is a classification problem, Logistic regression, SVM, Tree Classifier, and KNN algorithm is used to build machine learning models. After tuning, all the built model attained the same accuracy of 83.33 %.
- Further improvement can be done by further feature engineering on the data, and cross validations on the models.

Appendix

- All the Notebooks, Python codes, and presentations used in this project can be downloaded from the link : GitHub link.

Thank you!