IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Pavan Jahagirdar

27-11-2021

Outline

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

Executive Summary

Summary of methodologies

- Data collection with API and Web Scrapping
- Data Wrangling (EDA)
- EDA with SQL
- EDA for Data Visualization using Pandas and Matplotlib
- Interactive Dashboard with Plotly Dash
- Interactive Visual Analytics with Folium
- Predictive Analysis (Classification)

Summary of all results

- Data Analysis with interactive model
- Best Model for Predictive analysis

Introduction

Project background and context

In this capstone, we will predict if the Falcon 9 first stage will land successfully. 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.

Problems you want to find answers

- With what factors, the rocket will land successfully?
- The effect of each relationship of rocket variables on outcome.
- Conditions which will aid SpaceX have to achieve the best results.

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - With SpaceX rest API
  - Web Scraping from Wikipedia
- Perform data wrangling
  - One hot encoding data fields for machine learning and dropping irrelevant columns.
- Perform exploratory data analysis (EDA) using visualization and SQL
  - Scatters and Bar graphs to show pattern between data.
- 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 is the process of gathering and measuring information on targeted variables in an established system, which then enables one to answer relevant questions and evaluate outcomes.

DATA COLLECTION involves:

- Getting Data from API or Web Page
- Make Dataframe from it
- Filter Dataframe as per requirement
- Export to flat file

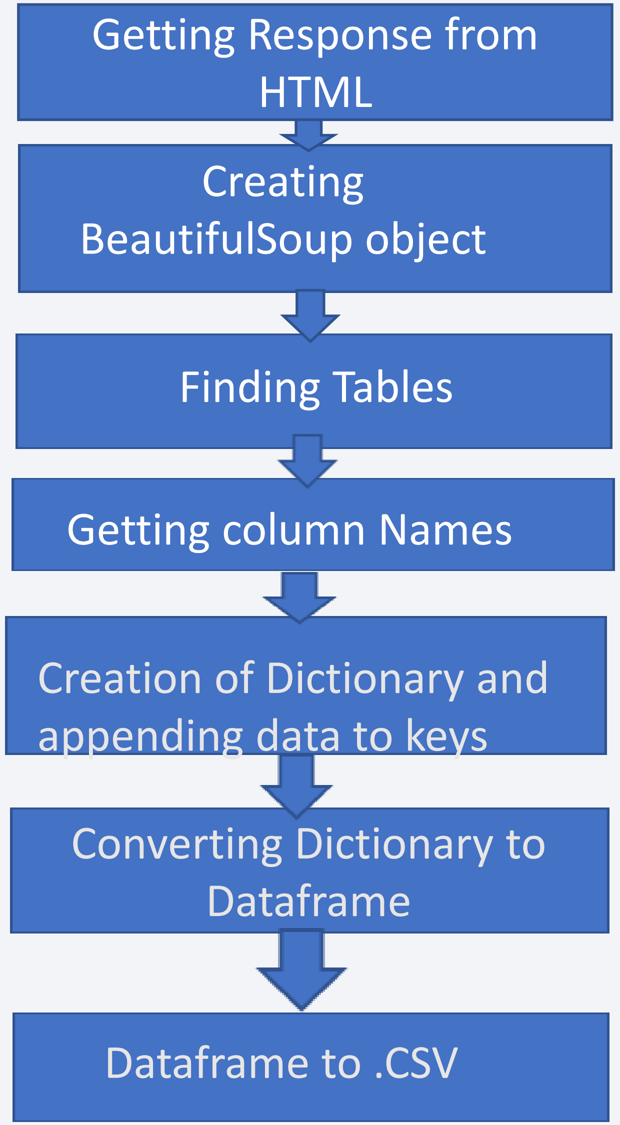
Data Collection – SpaceX API

1. Getting response from the API
2. Converting Response to a .json file
3. Apply custom function to clean data
4. Assign list to dictionary then create dataframe
5. Filter dataframe and export to flat file

	FlightNumber	Date	BoosterVersion	PayloadMass	Orbit	LaunchSite	Outcome	Flights	GridFins	Reused	Legs	LandingPad	Block	ReusedCount	Serial
4	1	2010-06-04	Falcon 9	NaN	LEO	CCSFS SLC 40	None None	1	False	False	False	None	1.0	0	B0003
5	2	2012-05-22	Falcon 9	525.0	LEO	CCSFS SLC 40	None None	1	False	False	False	None	1.0	0	B0005
6	3	2013-03-01	Falcon 9	677.0	ISS	CCSFS SLC 40	None None	1	False	False	False	None	1.0	0	B0007
7	4	2013-09-29	Falcon 9	500.0	PO	VAFB SLC 4E	False Ocean	1	False	False	False	None	1.0	0	B1003
8	5	2013-12-03	Falcon 9	3170.0	GTO	CCSFS SLC 40	None None	1	False	False	False	None	1.0	0	B1004

[GitHub link]

Data Collection - Scraping



	Flight No.	Launch site	Payload	Payload mass	Orbit	Customer	Launch outcome	Version Booster	Booster landing	Date	Time
0	0	CCAFS	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success\n	0	Failure	4 June 2010	18:45
1	0	CCAFS	Dragon	0	LEO	NASA	Success	0	Failure	8 December 2010	15:43
2	0	CCAFS	Dragon	525 kg	LEO	NASA	Success	0	No attempt\n	22 May 2012	07:44
3	0	CCAFS	SpaceX CRS-1	4,700 kg	LEO	NASA	Success\n	0	No attempt	8 October 2012	00:35
4	0	CCAFS	SpaceX CRS-2	4,877 kg	LEO	NASA	Success\n	0	No attempt\n	1 March 2013	15:10

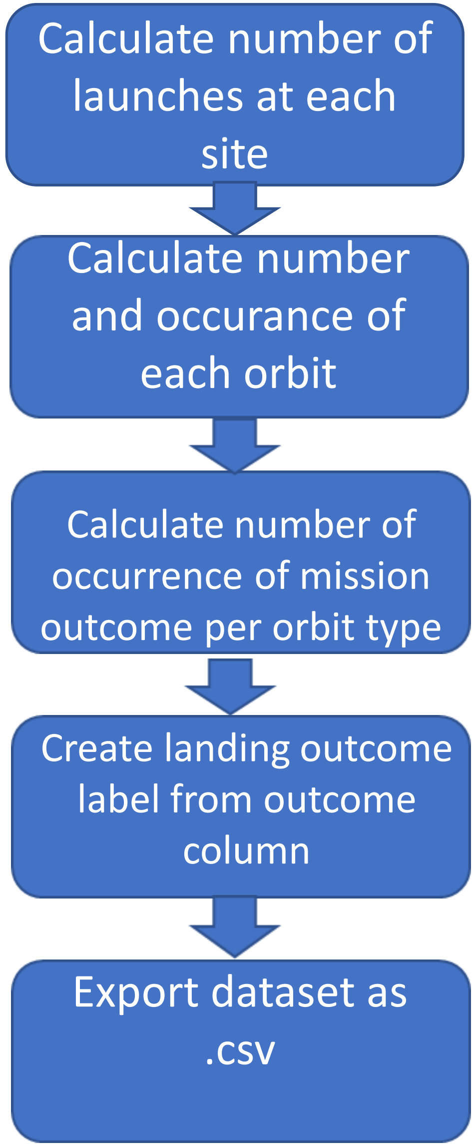
[GitHub Link]

Data Wrangling

Data Wrangling is the process of Cleaning and Unifying messy and complex data sets for easy access and analysis.

It involves:

- Loading Data
- Making dataframe from it
- Cleaning data
- Simplifying it to Boolean values
- Export it to flat file



	FlightNumber	Date	BoosterVersion	PayloadMass	Orbit	LaunchSite	Outcome	Flights	GridFins	Reused	Legs	LandingPad	Block	ReusedCount	Serial
0	1	2010-06-04	Falcon 9	6104.959412	LEO	CCAFS SLC 40	None None	1	False	False	False	NaN	1.0	0	B0003
1	2	2012-05-22	Falcon 9	525.000000	LEO	CCAFS SLC 40	None None	1	False	False	False	NaN	1.0	0	B0005
2	3	2013-03-01	Falcon 9	677.000000	ISS	CCAFS SLC 40	None None	1	False	False	False	NaN	1.0	0	B0007
3	4	2013-09-29	Falcon 9	500.000000	PO	VAFB SLC 4E	False Ocean	1	False	False	False	NaN	1.0	0	B1003
4	5	2013-12-03	Falcon 9	3170.000000	GTO	CCAFS SLC 40	None None	1	False	False	False	NaN	1.0	0	B1004
5	6	2014-01-06	Falcon 9	3325.000000	GTO	CCAFS SLC 40	None None	1	False	False	False	NaN	1.0	0	B1005

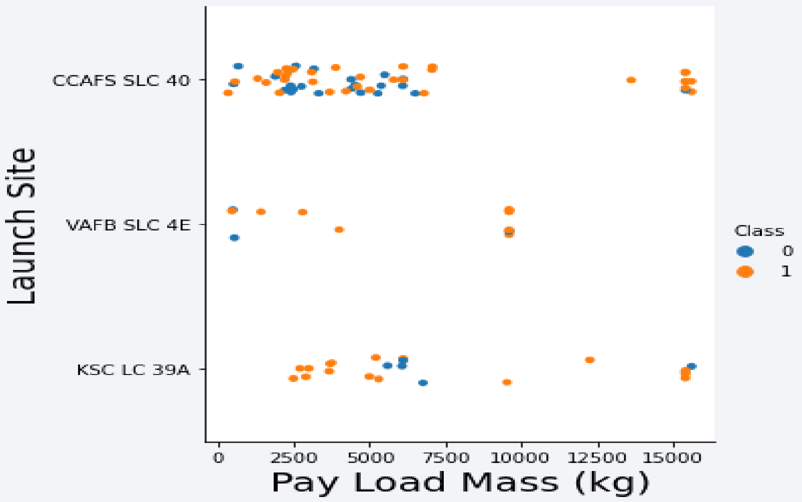
GitHub Link

EDA with Data Visualization

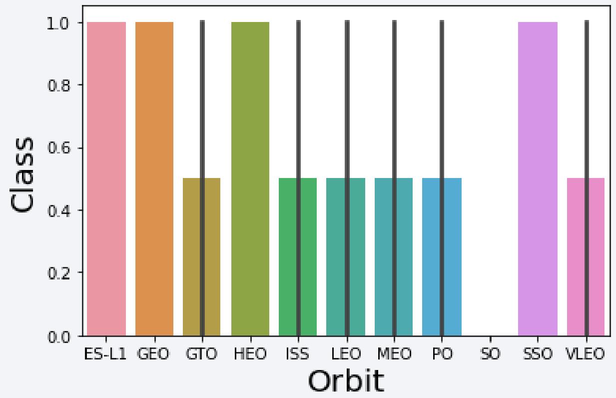
Exploratory Data Analysis is a approach of analyzing data sets to summarize their main characteristics, using statistical graphics and other data visualization method.

Scatter graphs drawn:

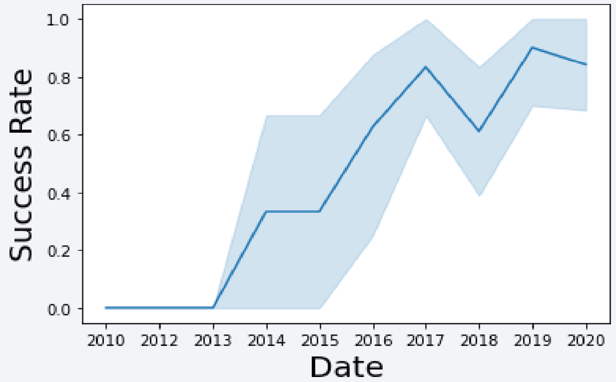
- Payload and Flight number
- Flight number and launchsites
- Payload and Launchsites
- Flight number and Orbit type
- Payload and Orbit type



Bar graph drawn:
Success rate(Class) vs Orbit Type



Line graph:
Success rate vs Date



[GitHub Link]

EDA with SQL

SQL is indispensable tool for Data Scientists and Analysts as most of the real world data is stored in databases. It's not the only standard language for Relational database for operations, but also incredibly powerful tool for analyzing data and drawing useful insights from it. Here IBM's DB2 cloud is used, which is fully managed SQL database provided as a service.

The SQL queries performed to gather information from given dataset:

- Displaying 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 F9v1.1
- Listing the date where the successful landing outcome in drone ship was achieved.
- Listing the names of the booster which have success in ground pad and have payload mass greater than 4000 but less than 6000
- Listing the total number of successful and failure mission outcomes
- Listing the number of booster_versions which have carried the maximum payload mass
- Listing the failed landing_outcomes in drone ship, their booster version and launch sites names for the 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 Link

Build an Interactive Map with Folium

Folium makes it easy to visualize data that's been manipulated in Python on an interactive leaflet map. We use the latitude and longitude button for find line text of transform text from image in google coordinates for each launch site and added a Circle Marker around each launch site with a label of the name of the launch site. It is also easy to visualize the number of success and failure for each launch site with Green and Red markers on the map.

Map objects	Code	Results
Map marker	Folium.marker()	Map object to mark on map.
Icon marker	Folium.icon()	Create an icon map
Circle marker	Folium.circle()	Create a circle where marker is pointed
Polyline	Folium.polyline()	Create a line between points
Marker cluster object	Markercluster()	This is the good way to simplify a map containing many markers having the same coordinate.
Antpath	Folium.plugins.anthpath()	Create an animated line between points.

GitHub Link

Build a Dashboard with Plotly Dash

Pie Chart showing the total success for all sites or by certain launch site

- Percentage of success in relation to launch site

Scatter Graph showing the correlation between Payload and Success for all sites or by certain launch site

- It shows the relationship between Success rate and Booster Version

Maps Object	Code	Results
Dash and its components	import dash import dash_html_components as html import dash_core_components as dcc from dash.dependencies *import* input, output	Plotly stewards Python's leading data viz and UI libraries. With Dash Open Source, Dash apps run on your local laptop or server. The Dash Core Component library contains a set of higher-level components like sliders, graphs, dropdowns, tables, and more. Dash provides all of the HTML tags as user-friendly Python classes.
pandas	*Import* pandas as pd	Fetching values from CSV and creating a dataframe
Plotly	*Import* plotly.express as px	Plot the graphs with interactive plotly liberary
Dropdown	dcc.dropdown()	Create a dropdown for launch sites
Rangeslider	dcc.rangeslider()	Create a rangeslider for payload mass range selection
Pie chart	Px.pie()	Creating the pie graph for success percentage display
Scatter chart	Px.scatter()	Creating the scattering graph for correleation display

Predictive Analysis (Classification)

1. Building Model

- Load our feature engineered data into dataframe
- Transform it into Numpy arrays
- Standardize and transform data
- Split data into training and test data sets
- Check how many test samples has been created
- List down machine learning algorithms we want to use
- Set our parameters and algorithms to GridSearchCV
- Fit our datasets into the GridSearchCV objects and train our model

2. Evaluating Model

- Check accuracy for each model
- Get best hyperparameters for each type of algorithms
- Plot Confusion Matrix

3. Finding Best Performing Classification Model

The model with best accuracy score wins the best performing model

GitHub Link

Results

Predictive analysis results

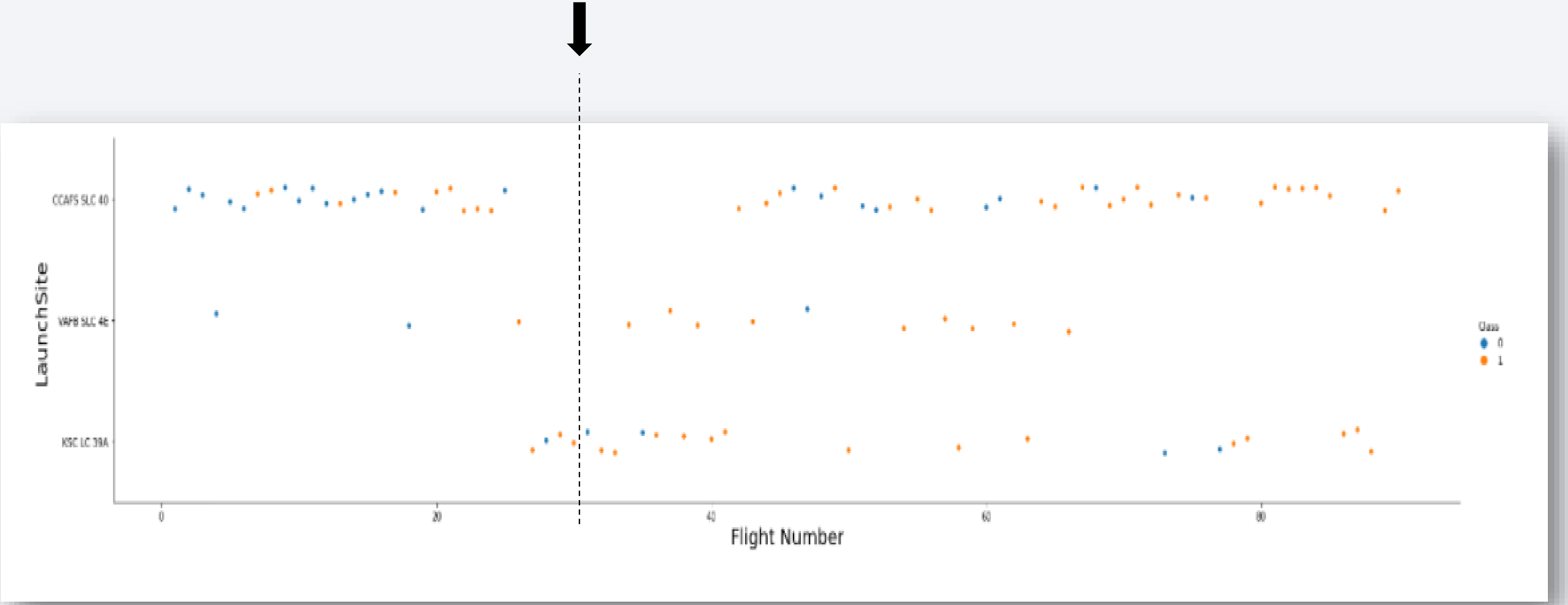
	Algorithm	Accuracy
0	KNN	0.848214
1	Decision Tree	0.885714
2	Logistic Regression	0.846429
3	SVM	0.848214

Section 2

Insights drawn from EDA

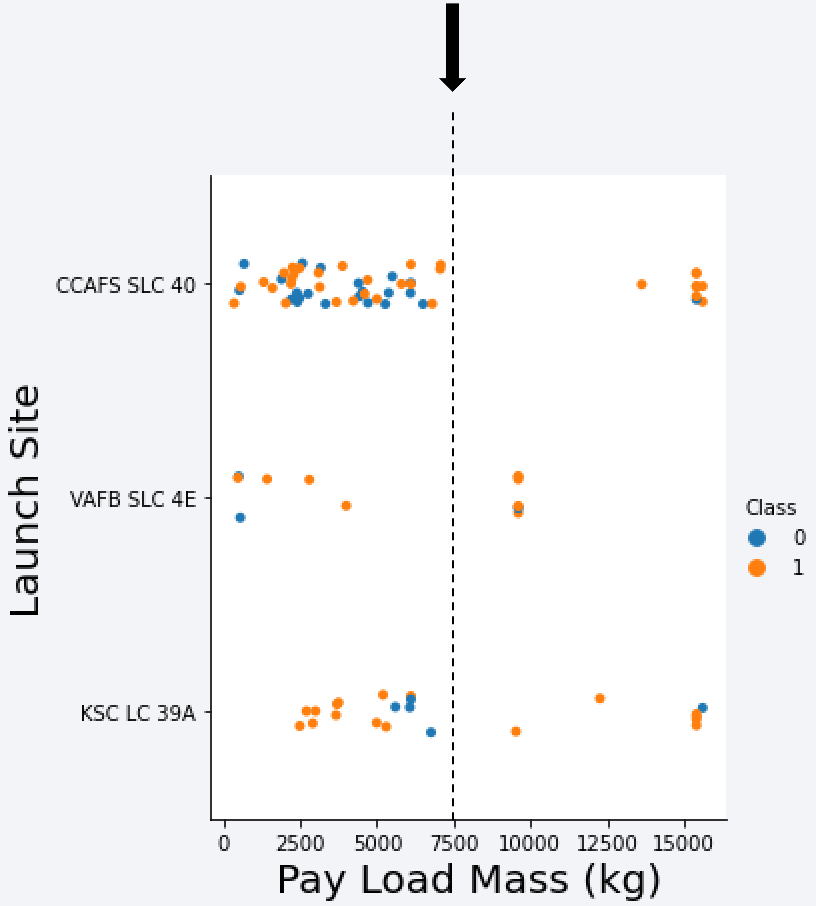
Flight Number vs. Launch Site

With Higher flight numbers(greater than 30) the success rate for the rocket is increasing



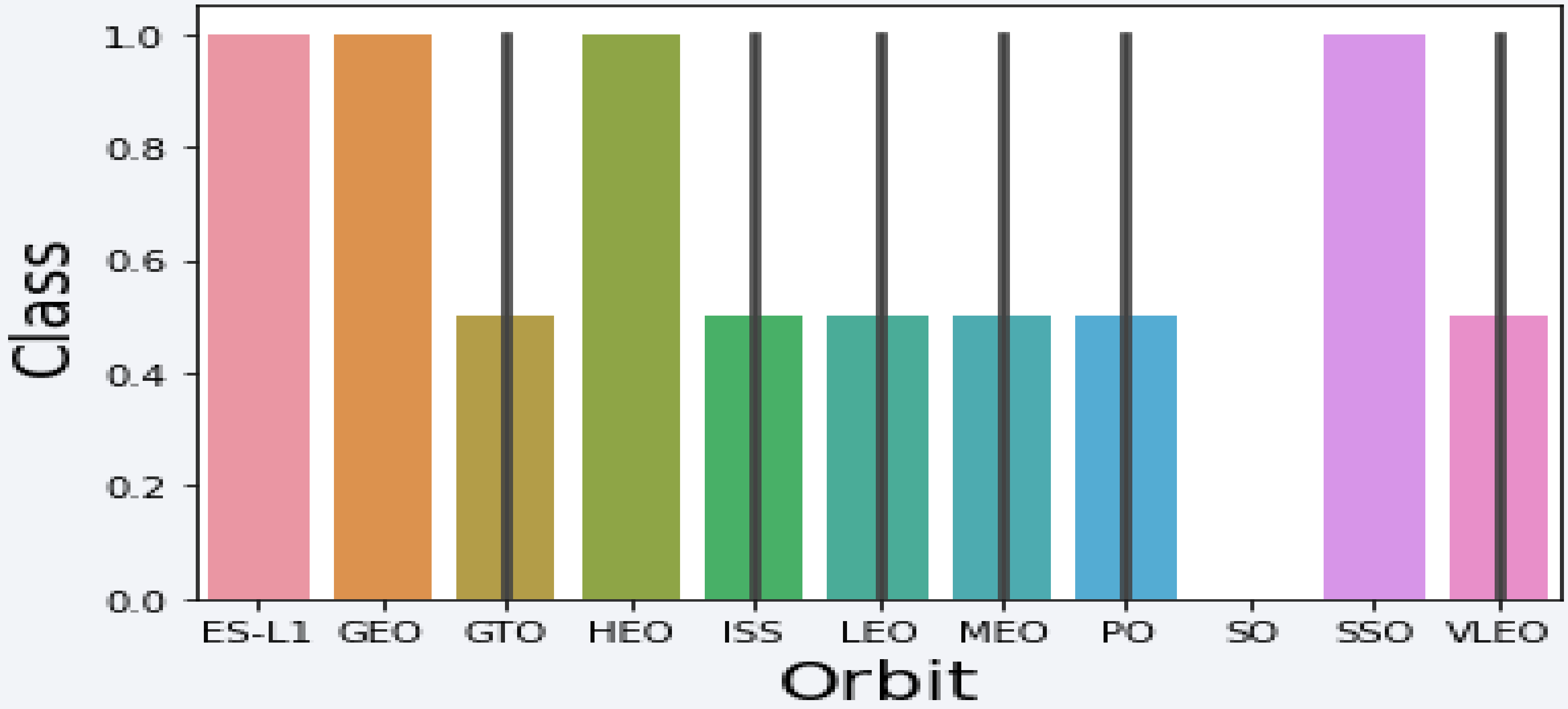
Payload vs. Launch Site

The greater the payload mass(greater than 7500kg) higher the success rate for rocket. But there’s no clear pattern to take a decision, if the launch site is dependent on payload mass for success launch.



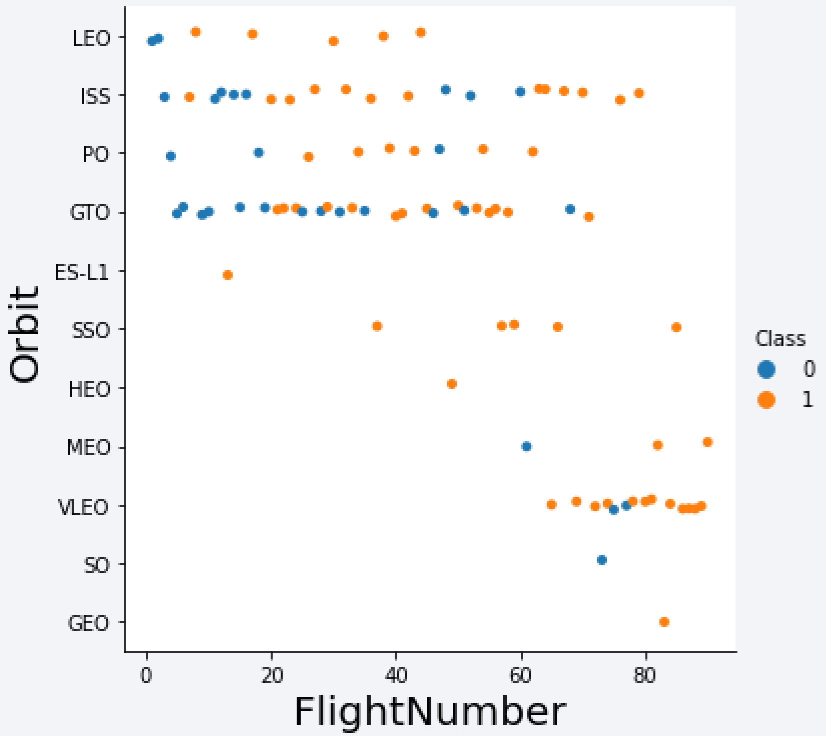
Success Rate vs. Orbit Type

The success rate for Orbits ES-L1, GEO,HEO, SSO are higher.



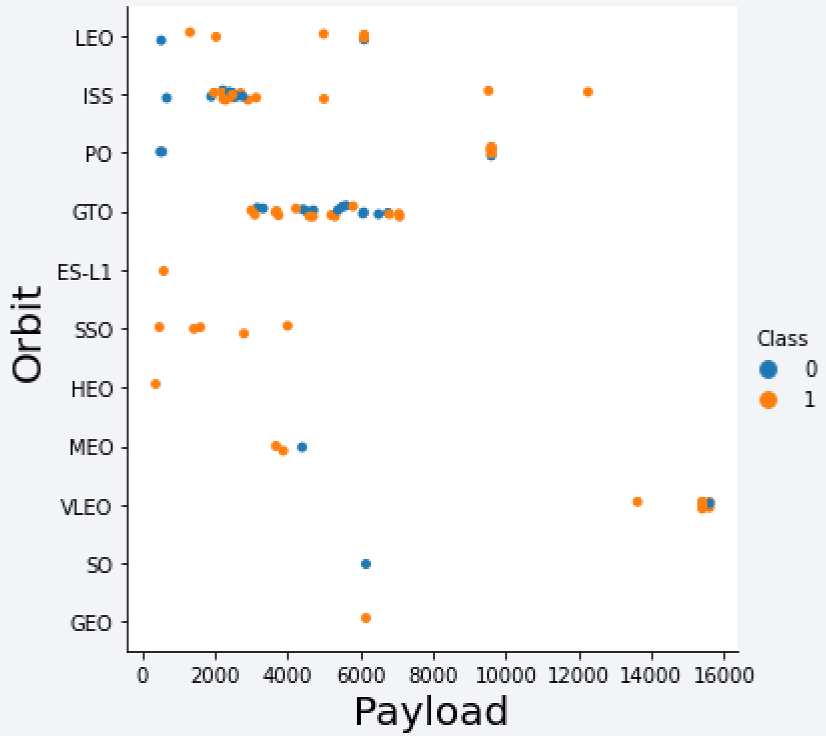
Flight Number vs. Orbit Type

- We can see that for LEO, success rate increases with Flight number.
- For GTO, there seems to be no relationship between orbit and Flight Number.
- And for VLEO, success rate is higher for Flight Number more than 60.



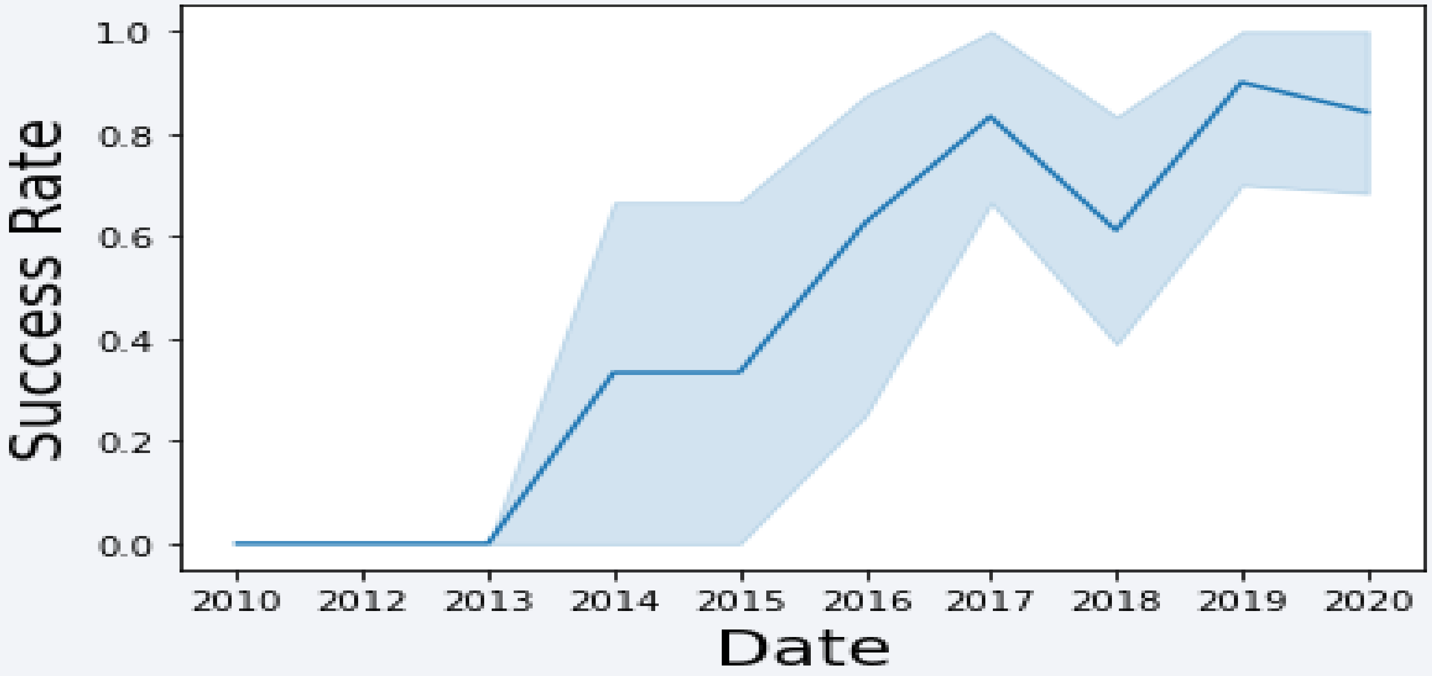
Payload vs. Orbit Type

- We observe that heavy payloads have a negative influence on MEO, GTO , VLEO orbits
- Positive on LEO, ISS orbits



Launch Success Yearly Trend

We can see that the success rate since 2013 kept increasing relatively though there is a slight dip after 2019.



All Launch Site Names

SQL Query

```
%sql select distinct(LAUNCH_SITE) from SPACEX
```

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

Description

Using the word DISTINCT in the query to pull unique value for launch_site column from table SPACEX

Launch Site Names Begin with 'CCA'

SQL Query

```
%sql select * from SPACEX where LAUNCH_SITE like 'CCA%' limit 5
```

Description

Using keyword ‘limit 5’ in the query, 5 records are fetched from the table SPACEX and with condition ‘like’ keyword with wild card – ‘CCA’. The percentage in the end suggests that the ‘LAUNCH_SITE’ name start with CCA

DATE	Time (UTC)	booster_version	launch_site	payload	payload_mass__kg_	orbit	customer	mission_outcome	Landing _Outcome
2010-04-06	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
2010-08-12	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)
2012-08-10	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
2013-01-03	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt
2013-03-12	22:41:00	F9 v1.1	CCAFS LC-40	SES-8	3170	GTO	SES	Success	No attempt

Total Payload Mass

SQL Query

```
%sql select sum(PAYLOAD_MASS__KG_) from SPACEX where CUSTOMER = 'NASA (CRS)'
```

Description

Using the function SUM calculates the total in the column PAYLOAD_MASS_KG_ and where clause filters the data to fetch customers by name ‘NASA (CRS)’

Output

1
22007

Average Payload Mass by F9 v1.1

SQL Query

```
%sql select avg(PAYLOAD_MASS__KG_) from SPACEX where BOOSTER_VERSION = 'F9 v1.1'
```

Description

Using the function 'avg' works out the average in the column PAYLOAD_MASS_KG_
The 'where' clause filters the dataset to only perform calculations on BOOSTER_VERSION F9 v1.1

Output

1
3676

First Successful Ground Landing Date

SQL Query

```
%sql select min(DATE) from SPACEX where 'Landing_Outcome' = 'Success (ground pad)'
```

Description

Using the function MIN works out the minimum date in the column Date and Where clause filters the data to perform calculations on ‘Landing_Outcome’ with values ‘Success (ground pad)’

First Succesful Landing Outcome in Ground Pad
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

SQL Query

```
%sql select BOOSTER_VERSION from SPACEX where 'Landing_Outcome' = 'Success (drone ship)' and PAYLOAD_MASS__KG_ > 4000 and PAYLOA
D_MASS__KG_ < 6000
```

Description

Selecting only Booster_Version,
Where clause filters the dataset to Landing_Outcome = Success(drone ship)

And clause specifies additional filter conditions
PAYLOAD_MASS_KG_ >4000 and PAYLOAD_MASS_KG_ < 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

SQL Query

```
%sql SELECT sum(case when MISSION_OUTCOME LIKE '%Success%' then 1 else 0 end) AS "Successful Mission",\
sum(case when MISSION_OUTCOME LIKE '%Failure%' then 1 else 0 end) AS "Failure Mission" \ from SPACEX;
```

Description
Selecting multiple count is a complex query. Here case clause is used within sub-query for getting both success and failure counts in same query.
Case when MISSION_OUTCOME like '%Success%' then 1 else 0 end returns a Boolean value which we sum to get the result needed.

Successful Mission	Failure Mission
100	1

Boosters Carried Maximum Payload

SQL Query

```
%sql select BOOSTER_VERSION from SPACEX where PAYLOAD_MASS__KG_ = (select max(PAYLOAD_MASS__KG_) from SPACEX)
```

Description

Using the function max works out the maximum PAYLOAD MASS KG in the column and Where clause filters Booster version which had the maximum payload.

booster_version
F9 B5 B1048.4
F9 B5 B1049.4
F9 B5 B1049.5
F9 B5 B1060.2
F9 B5 B1058.3

2015 Launch Records

SQL Query

```
%sql SELECT BOOSTER_VERSION, LAUNCH_SITE FROM SPACEX WHERE year(DATE) = '2015' AND \
'Landing_Outcome' = 'Failure (drone ship)';
```

Description

Here is the list of records which displays the month names, failure landing_outcomes in drone ship, Booster versions, launch_site for the months in the year 2015

Month	booster_version	launch_site
January	F9 v1.1 B1012	CCAFS LC-40
April	F9 v1.1 B1015	CCAFS LC-40

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

SQL Query

```
%sql SELECT LANDING__OUTCOME as "Landing Outcome", COUNT(LANDING__OUTCOME) AS "Total Count" FROM SPACEX \
WHERE DATE BETWEEN '2010-06-04' AND '2017-03-20' \
GROUP BY  LANDING__OUTCOME \
ORDER BY COUNT(LANDING__OUTCOME) DESC ;
```

Description

Selecting only Landing_Outcome ,
Where Clause filters the data with date between 2010-06-04 and 2017-03-20

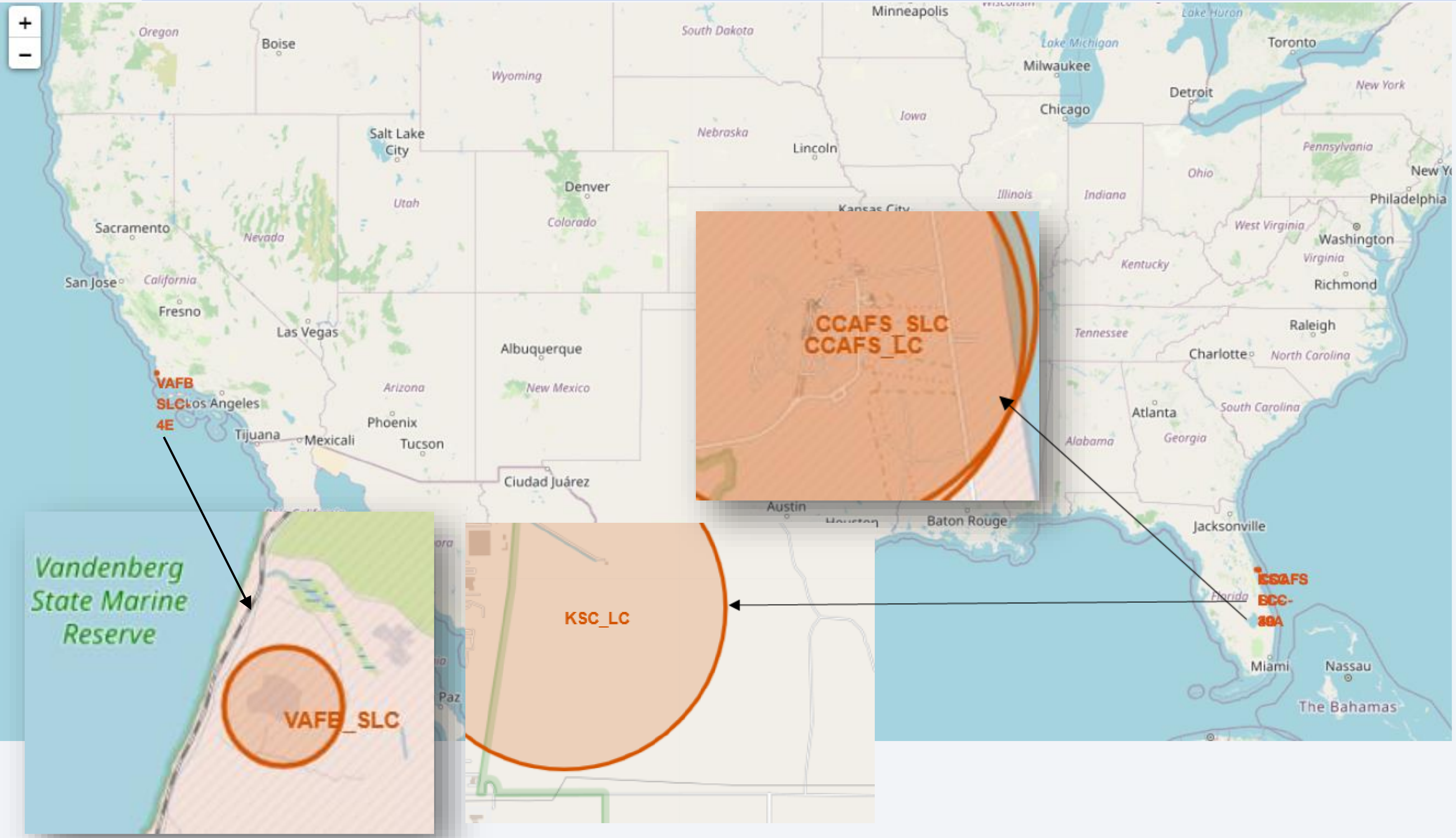
Grouping by Landing_Outcome
Order by Count(Landing_outcome) in descending order

Landing Outcome	Total Count
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

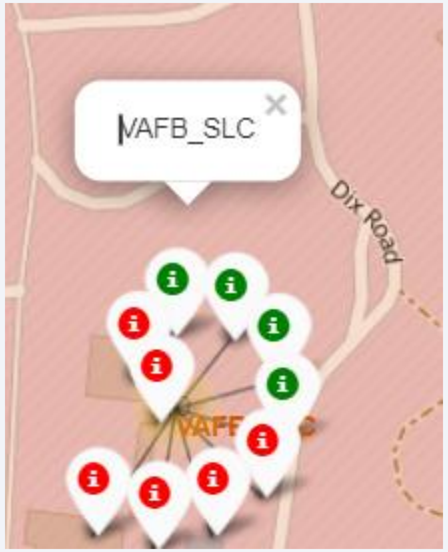
Section 4

Launch Sites Proximities Analysis

Launch Sites of Folium Map

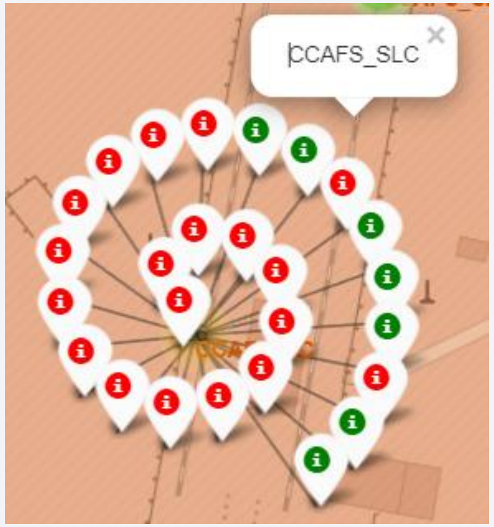


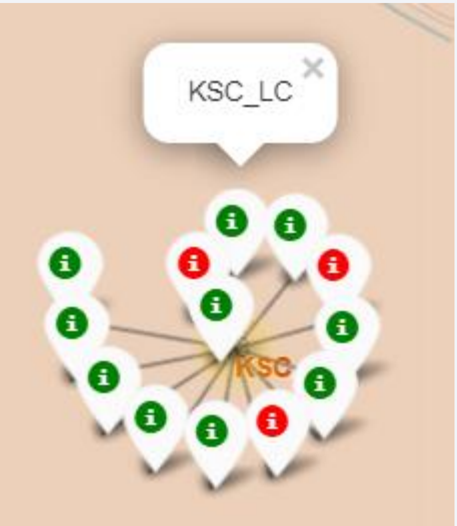
Colour loaded Launch Record

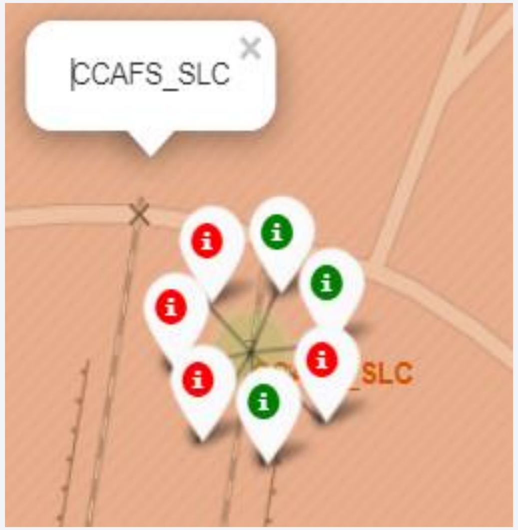


Green Symbol Shows successful launches

Red Symbol Shows Failure launches





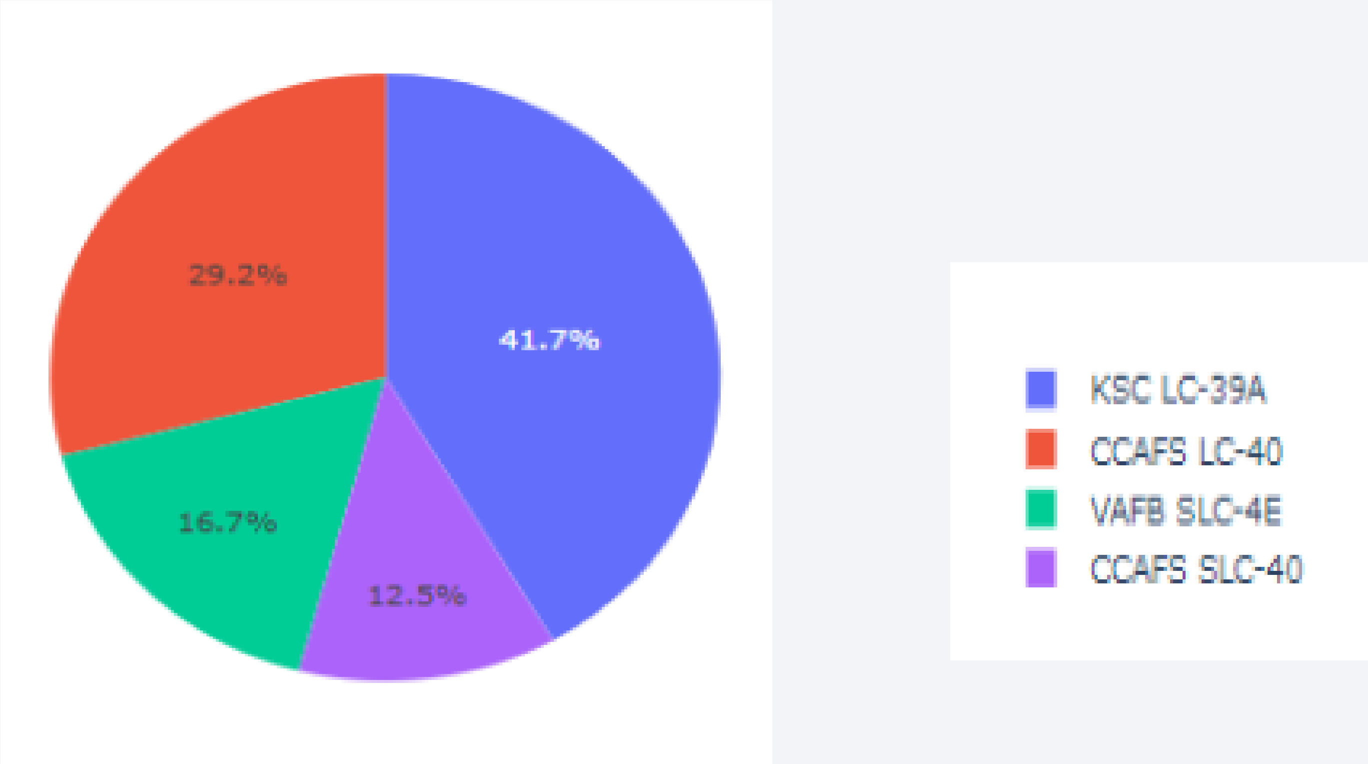


Section 5

Build a Dashboard with Plotly Dash

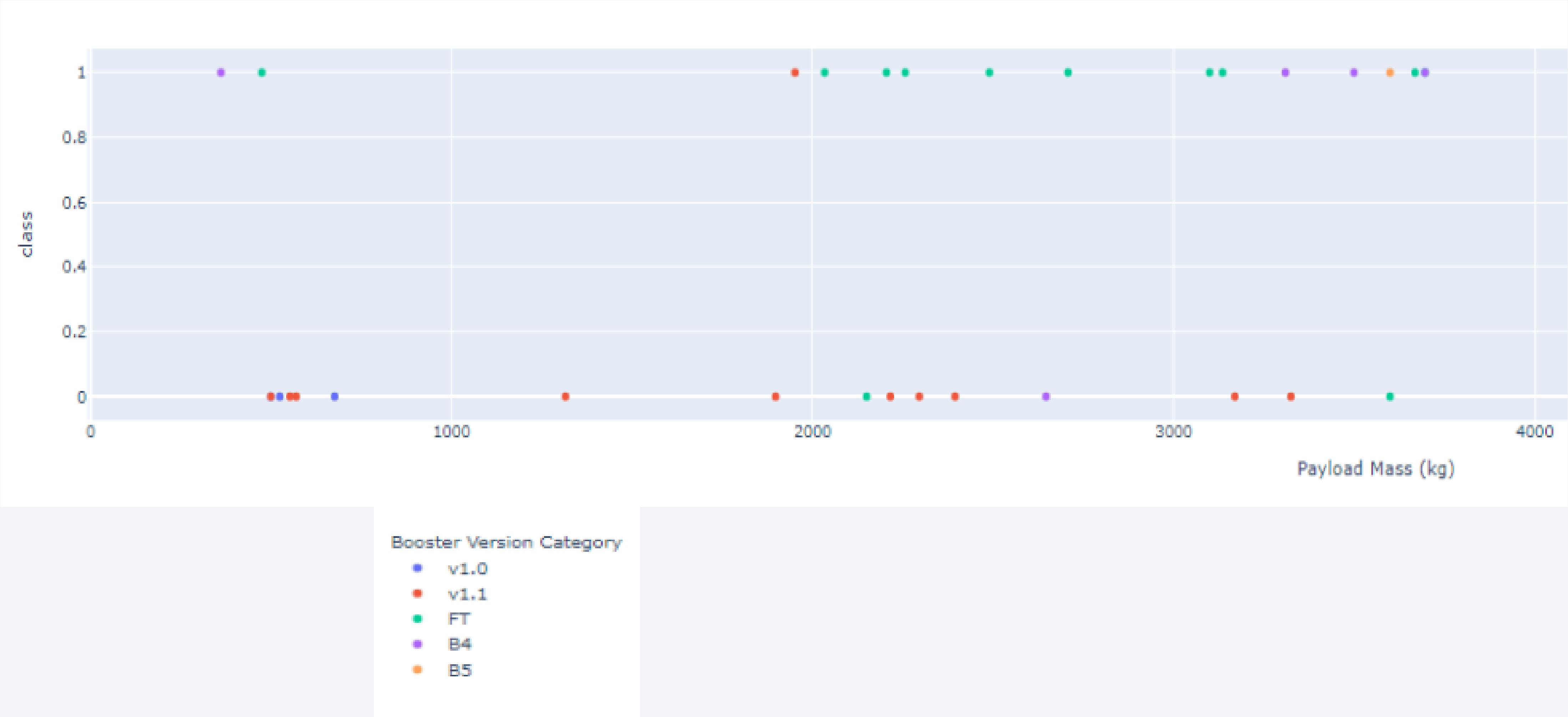
SpaceX Launch Reports Dashboard

We can see that KSC LC has Highest success rates



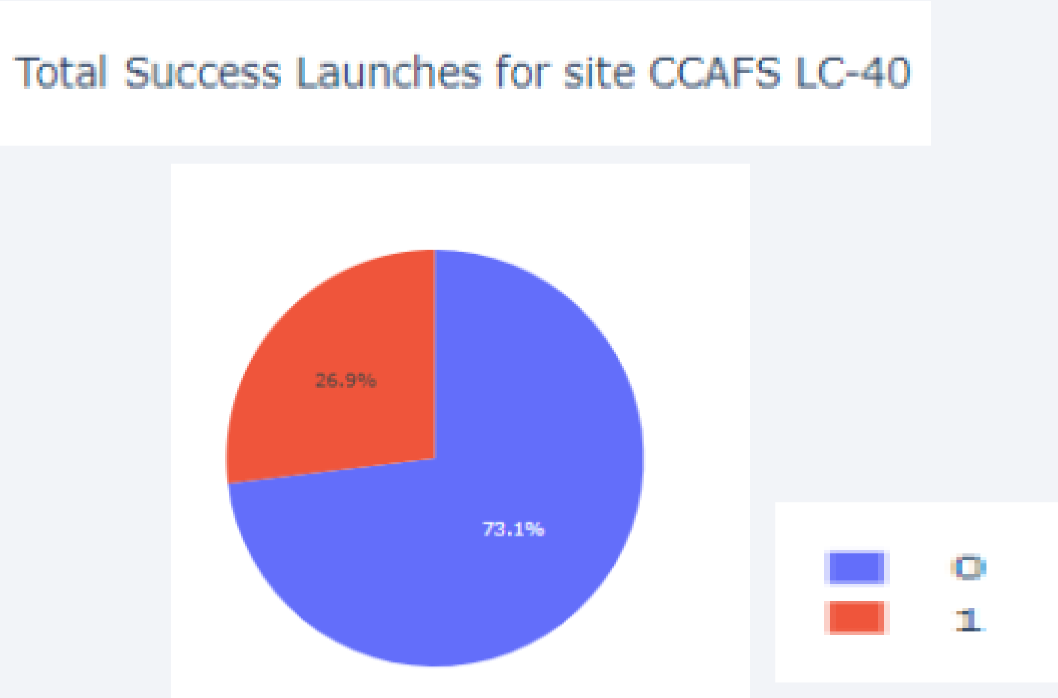
Correlation between Payload and Success for all sites

It can be seen that the success rates for low weighted is higher than the heavy weighted payloads



Launch Site with Heighest Success Launch Ratio

KSC LC-39A achieved a 73.1% success rate while getting a 26.9% failure rate



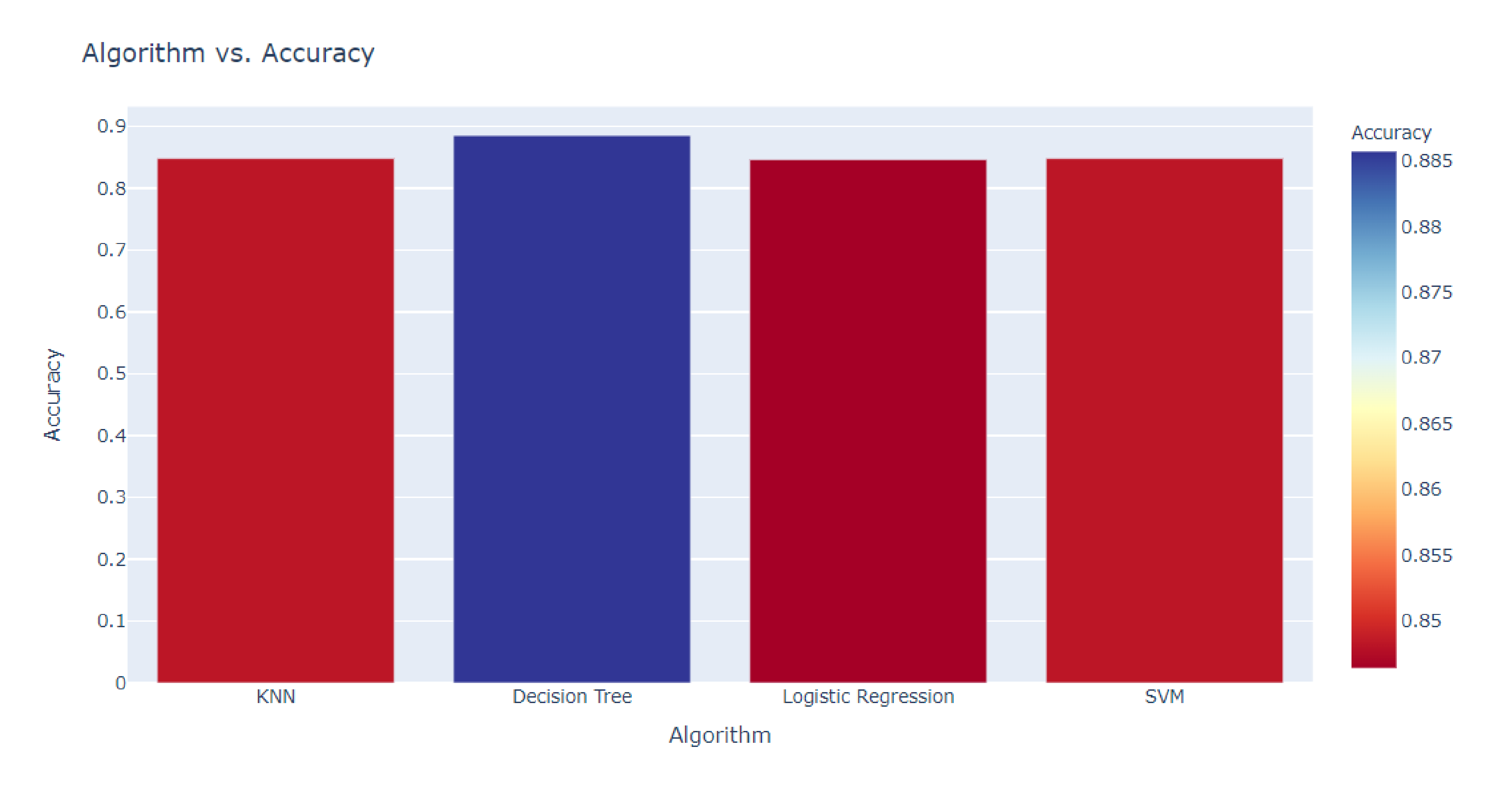
Section 6

Predictive Analysis (Classification)

Classification Accuracy

From Bar graph, it can be seen that Decision tree has the Highest accurate algorithm with accuracy 0.885714

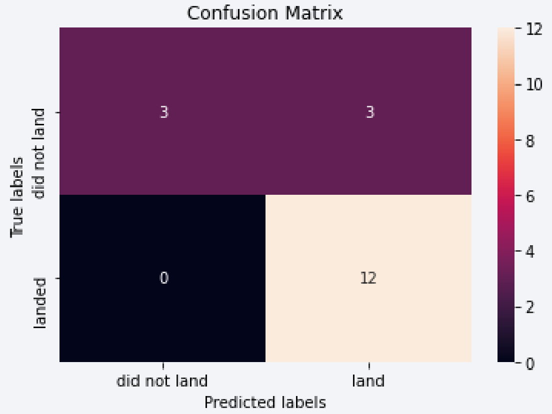
	Algorithm	Accuracy
0	KNN	0.848214
1	Decision Tree	0.885714
2	Logistic Regression	0.846429
3	SVM	0.848214



Confusion Matrix

All Models have same Confusion Matrix

Logistic Regression



SVM

Decision tree

K Nearest Neighbour

Conclusions

- Orbits ES-L1, GEO, HEO, SSO has highest Success rates
- Success rates for SpaceX launches has been increasing relatively with time
- KSC_LC_39A had the most successful launches but increasing payload mass seems to have negative impact on success
- Decision tree classifier algorithm is the best for machine learning model for provided dataset

Thank you!