IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Khaled Ahmed
03/06/2022

Outline

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

Executive Summary

- Summary of methodologies
  - Data Collection using SpaceX API and Web Scaping (Wikipedia)
  - Data Wrangling using Pandas
  - Exploratory Data Analysis (EDA) with SQL, and Visualization (Pandas and Matplotlib)
  - Interactive Visual Analytics and Dashboard with Folium and Plotly Dash
  - Machine Learning Prediction by Predictive Analysis (Classification) using NumPy and Scikit Learn
- Summary of all results
  - Data collection is possible using SpaceX API and Wikipedia
  - Missing data was identified using Data Wrangling
  - Best features to predict success launches were identified using EDA
  - Visual Analytics and Dashboard showed proximity and success rate of launches by sites and payload
  - Machine Learning helped select the best Classification model to predict success and failure

Introduction

- The objective is to take the role of a data scientist for a new rocket company, Space Y, that would like to compete with SpaceX

- Desirable answers
    - Determine the price of each launch by gathering information about SpaceX and creating analytics dashboard for the team as the best way to estimate the total cost for launches is by predicting the successful landings of the first stage of rockets
    - Determine if SpaceX will reuse the first stage using machine learning model and public information

Section 1

Methodology

Methodology

Executive Summary

- Data Collection
  - How data was collected is described
- Data Wrangling
  - How data was processed is described
- Exploratory data analysis (EDA) is performed using visualization and SQL
- Interactive visual analytics is performed using Folium and Plotly Dash
- Predictive analysis is performed using classification models
  - How to build, tune, evaluate classification models is described

Data Collection

- The datasets were collected using;
  - SpaceX API (https://api.spacexdata.com/v4)
  - Web Scaping from Wikipedia (https://en.wikipedia.org/wiki/List_of_Falcon_9_and_Falcon_Heavy_launches)

SpaceX API

	FlightNumber	Date	BoosterVersion	PayloadMass	Orbit	LaunchSite	Outcome	Flights	GridFins	Reused	Legs	LandingPad	Block	ReusedCount	Serial	Longitude	Latitude
4	1	2010-06-04	Falcon 9	NaN	LEO	CCSFS SLC 40	None None	1	False	False	False	None	1.0	0	B0003	-80.577366	28.561857
5	2	2012-05-22	Falcon 9	525.0	LEO	CCSFS SLC 40	None None	1	False	False	False	None	1.0	0	B0005	-80.577366	28.561857
6	3	2013-03-01	Falcon 9	677.0	ISS	CCSFS SLC 40	None None	1	False	False	False	None	1.0	0	B0007	-80.577366	28.561857
7	4	2013-09-29	Falcon 9	500.0	PO	VAFB SLC 4E	False Ocean	1	False	False	False	None	1.0	0	B1003	-120.610829	34.632093
8	5	2013-12-03	Falcon 9	3170.0	GTO	CCSFS SLC 40	None None	1	False	False	False	None	1.0	0	B1004	-80.577366	28.561857
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...

Web Scraping

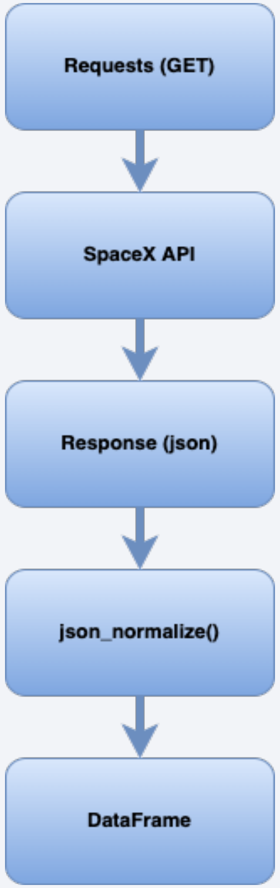
	Flight No.	Launch site	Payload	Payload mass	Orbit	Customer	Launch outcome	Version Booster	Booster landing	Date	Time
0	1	CCAFS	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success\n	F9 v1.0B0003.1	Failure	4 June 2010	18:45
1	2	CCAFS	Dragon	0	LEO	NASA	Success	F9 v1.0B0004.1	Failure	8 December 2010	15:43
2	3	CCAFS	Dragon	525 kg	LEO	NASA	Success	F9 v1.0B0005.1	No attempt\n	22 May 2012	07:44
3	4	CCAFS	SpaceX CRS-1	4,700 kg	LEO	NASA	Success\n	F9 v1.0B0006.1	No attempt	8 October 2012	00:35
4	5	CCAFS	SpaceX CRS-2	4,877 kg	LEO	NASA	Success\n	F9 v1.0B0007.1	No attempt\n	1 March 2013	15:10

Data Collection – SpaceX API

- Data was collected by a GET API call to the SpaceX API, and the response was converted to json format, which (json) was normalized to a DataFrame using json_normalize()
- Self-defined helper functions were used to populate and normalize only the relevant columns and data
- Only data for 'Falcon 9' was taken and the missing data was handled

❑ Source Code:

- https://github.com/khhaledahmaad/IBM-Data-Science-Professional-Certificate/blob/14bd495b945037a64f8381ac41d98005c7696b01/Applied%20Data%20Science%20Capstone/week1/Collecting%20the%20Data/jupyter-labs-spacex-data-collection-api.ipynb

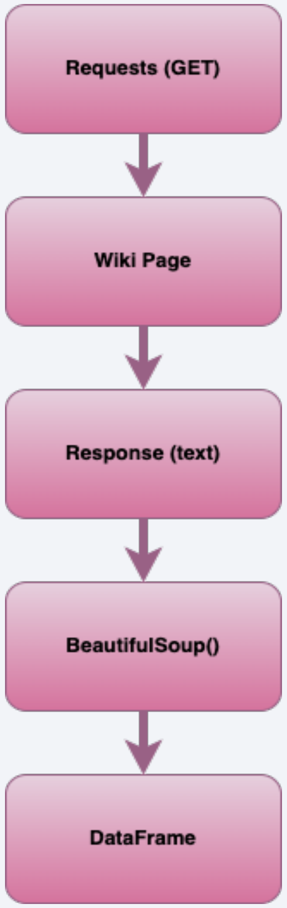


Data Collection – Web Scraping

- Data was collected by a GET API call to the Wikipedia Page, and the response was converted to text (html) format, which (text) was normalized to a DataFrame using BeautifulSoup()
- Self-defined helper functions were used to populate and normalize only the relevant columns and data
- Relevant columns and data were extracted from the table elements ('table', 'th' and 'tr') from the BeautifulSoup() object

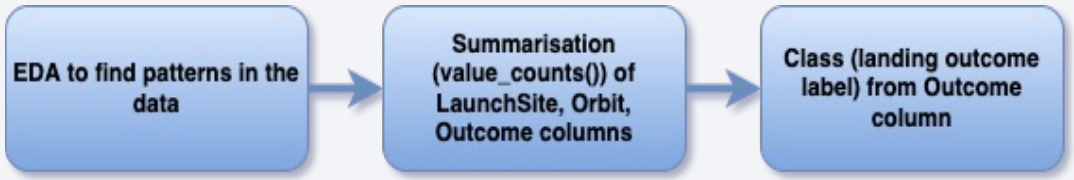
❑ Source Code:

- https://github.com/khhaledahmaad/IBM-Data-Science-Professional-Certificate/blob/14bd495b945037a64f8381ac41d98005c7696b01/Applied Data Science Capstone/week1/Collecting the Data/jupyter-labs-webscraping.ipynb



Data Wrangling

- Some Exploratory Data Analysis (EDA) is performed to determine the label for training supervised models
- The number and occurrences of launches on each site, each orbit and mission outcome per orbit type were calculated
- A landing outcome label, Class, with 1's and 0's from Outcome column was create where;
    - '1' is the good outcome, i.e., the booster did land
    - '0' is the bad outcome, i.e., the booster did not land



❑ Source Code:

- https://github.com/khhaledahmaad/IBM-Data-Science-Professional-Certificate/blob/14bd495b945037a64f8381ac41d98005c7696b01/Applied Data Science Capstone/week1/Data Wrangling/labs-jupyter-spacex-Data wrangling.ipynb

EDA with Data Visualization

- To perform EDA, scatter plots and a bar chart and a line graph were plotted to find the correlation between pair of features
- The following graphs were added:
  - Flight Number vs. Launch Site scatter plot
  - Payload vs. Launch Site scatter plot
  - Success rate (mean of 'Class') vs. Orbit type bar chart
  - Flight Number vs. Orbit type scatter plot
  - Payload vs. Orbit type scatter plot
  - Date (Year) vs. Success (Class) line graph

❑ Source Code:

- https://github.com/khhaledahmaad/IBM-Data-Science-Professional-Certificate/blob/14bd495b945037a64f8381ac41d98005c7696b01/Applied Data Science Capstone/week2/Exploratory Analysis Using Pandas and Matplotlib/jupyter-labs-eda-dataviz.ipynb

EDA with SQL

- The SQL queries performed:

1. Display the names of the unique launch sites in the space mission
2. Display 5 records where launch sites begin with the string 'CCA'
3. Display the total payload mass carried by boosters launched by NASA (CRS)
4. Display average payload mass carried by booster version F9 v1.1
5. List the date when the first successful landing outcome in ground pad was achieved
6. List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000
7. List the total number of successful and failure mission outcomes
8. List the names of the booster_versions which have carried the maximum payload mass, using a subquery
9. List the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015
10. 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

❑ Source Code:

- https://github.com/khhaledahmaad/IBM-Data-Science-Professional-Certificate/blob/14bd495b945037a64f8381ac41d98005c7696b01/Applied Data Science Capstone/week2/Exploratory Analysis Using SQL/jupyter-labs-eda-sql-coursera.ipynb

Build an Interactive Map with Folium

- Markers, circles, lines, and marker clusters were added to the folium map
  - Markers indicate text label around a specific coordinate
  - Circle indicate highlighted areas around a specific coordinate, like NASA Johnson Space
  - Lines are used indicate distance between two coordinates
  - Marker Clusters are used as a good way to simply a map containing many markers

❑ Source Code:

  - https://github.com/khhaledahmaad/IBM-Data-Science-Professional-Certificate/blob/14bd495b945037a64f8381ac41d98005c7696b01/Applied Data Science Capstone/week3/Interactive Visual Analytics and Dashboard/lab_jupyter_launch_site_location.ipynb

Build a Dashboard with Plotly Dash

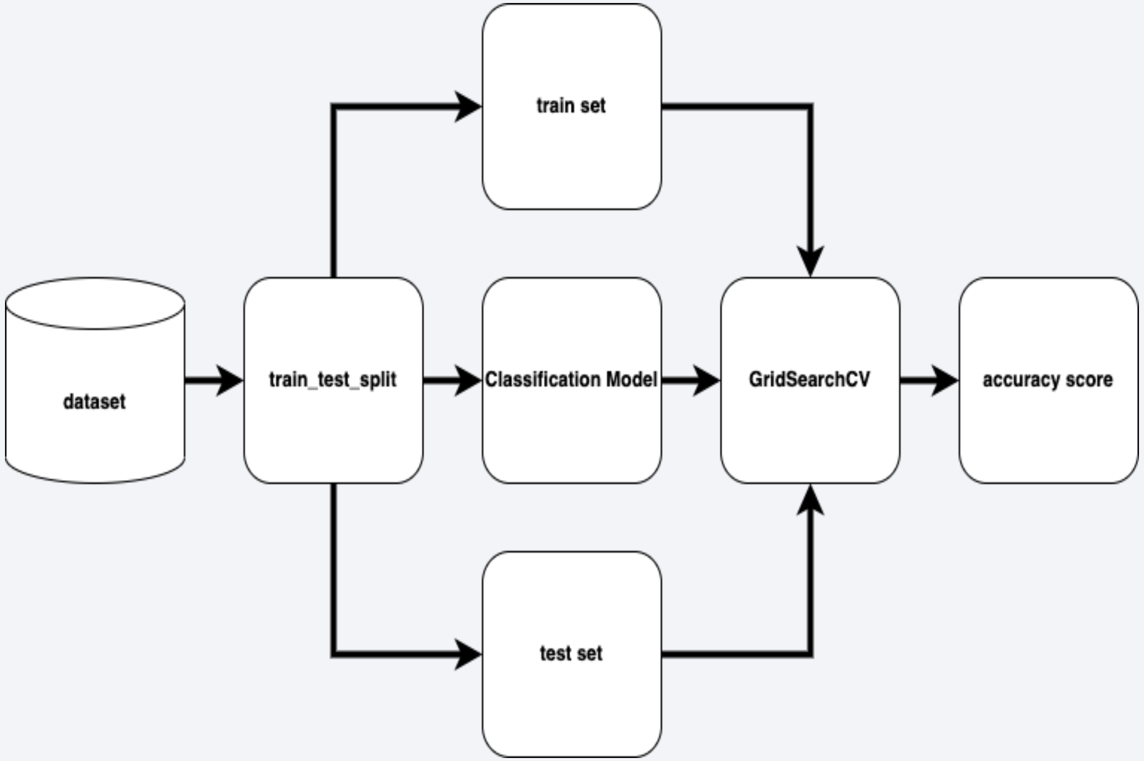
- The following plots/graphs and interactions were added to the dashboard:
  - Launch Site Drop-down Input Component
  - Range Slider to Select Payload
  - Launch site vs. Success rate pie chart based on selected site dropdown
  - Payload vs. Success scatter plot based on selected payload range
- The dropdown selection was useful to select a specific site to visualize the success rate for only that specific site
- The range slider was useful to get scatter plot for payload vs. success for a specific range of payload

❑ Source Code:

- https://github.com/khhaledahmaad/IBM-Data-Science-Professional-Certificate/blob/14bd495b945037a64f8381ac41d98005c7696b01/Applied Data Science Capstone/week3/Interactive Visual Analytics and Dashboard/spacex_dash_app.py

Predictive Analysis (Classification)

- The dataset was split into train and test sets using train_test_split
- Four Classification models were built; ‘Logistic Regression’, ‘Support Vector Machine’, ‘Decision Tree’, and ‘K-nearest Neighbours’
- For each model, to find the best hyperparameters and best accuracy score, ‘GridSearchCV’ was used and fitted with the train set
- ‘GridSearchCV’ object for each model was tested against the test set and accuracy score was calculated



❑ Source Code:

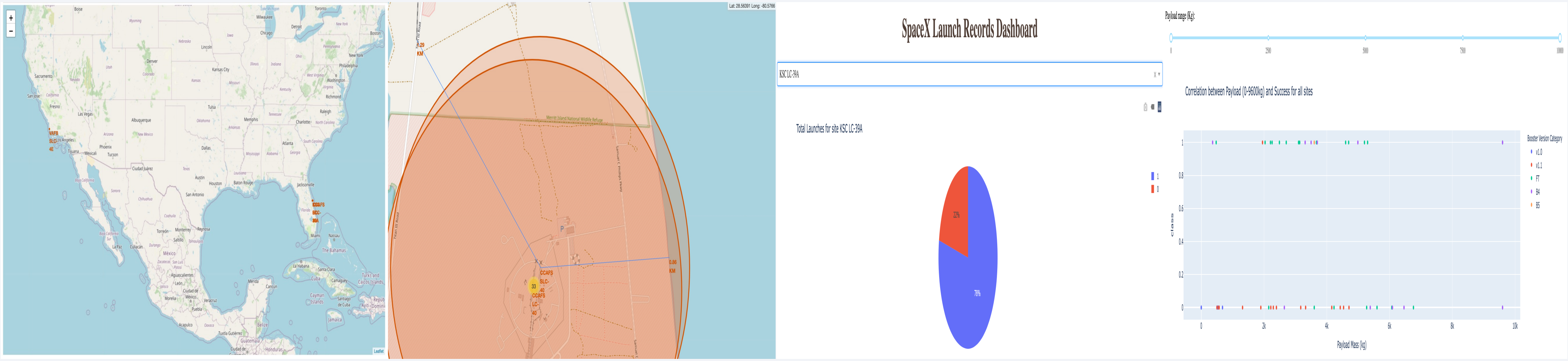
- https://github.com/khhaledahmaad/IBM-Data-Science-Professional-Certificate/blob/14bd495b945037a64f8381ac41d98005c7696b01/Applied Data Science Capstone/week4/Predictive Analysis (Classification)/SpaceX_Machine Learning Prediction_Part_5.ipynb

Results

- Exploratory data analysis results:
  - SpaceX uses 4 launch sites
  - There is no successful landing from 2010 till 2013
  - With increase in the flight numbers the success rate also increased
  - After ‘2013’ till ‘2020’, the success rate showed a significant increasing trend
  - The average payload mass carried by booster version ‘F9 v1.1’ is ‘2928 kg’
  - The first successful landing outcome in ground pad was achieved in ‘2015-12-22’
  - Only 4 boosters have success in drone ship and with payload between ‘4000 kg’ to ‘6000 kg’
  - Almost 100% (99/101) mission outcome was successful
  - ’12’ ‘F9’ booster versions have carried the maximum payload mass
  - ‘2’ ‘F9’ booster versions, have failed landing outcome in drone ship in year 2015
  - Between '2010-06-04' and '2017-03-20’ the most popular landing outcome is ‘No attempt’

Results

- Interactive analytics demo in screenshots:

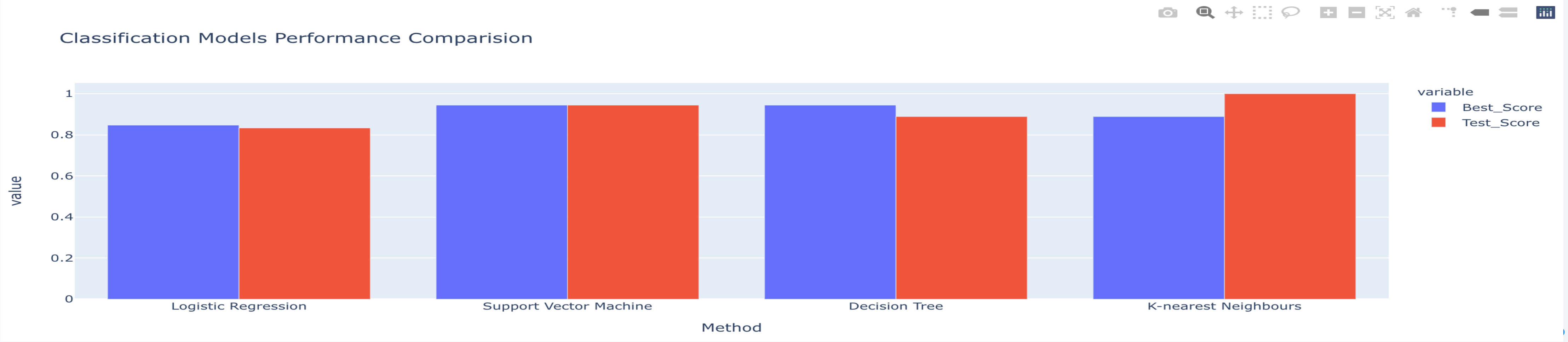


- All the launch sites are near cost lines, perhaps for safety and to prevent accidents also their proximities to roads and railroads have made them easily accessible incase of hazards

- The interactive dashboard can filter success rate by launch site and payload range for quick and effective analysis

Results

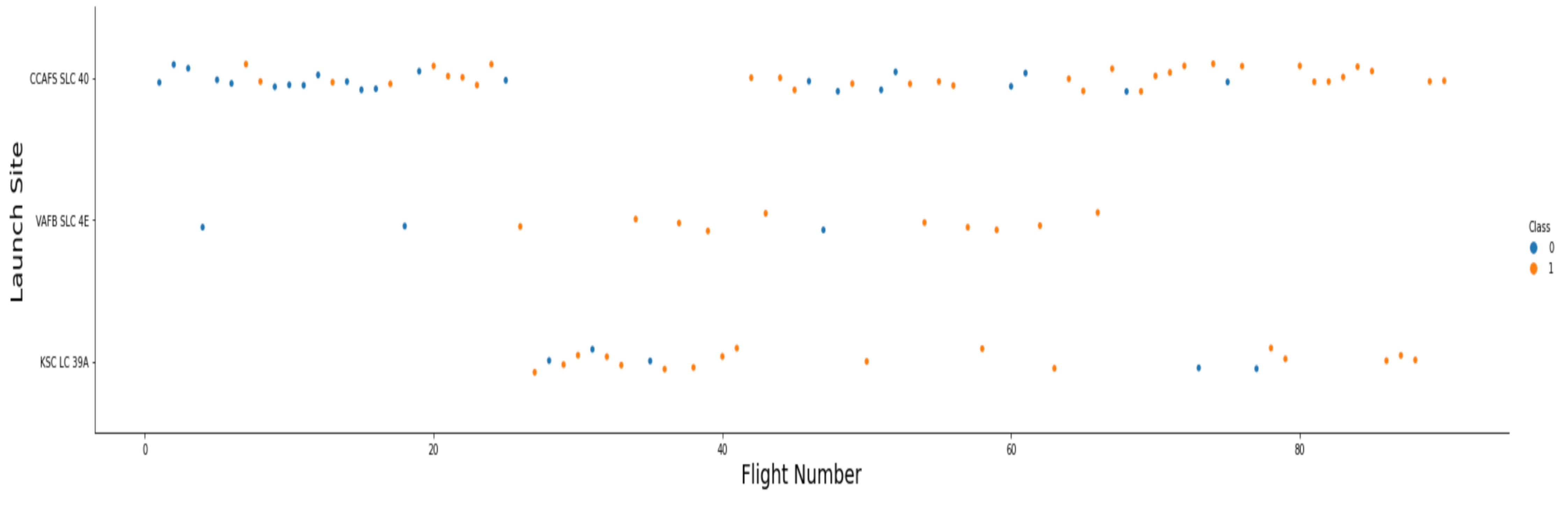
- Predictive analysis results:
  - All the Classification models are can predict the landing outcome more than 80% precisely
  - The best accuracy score, which is around 94%, is achieved for the ‘Support Vector Machine’ and the ‘Decision Tree’ classifier models
  - The highest accuracy score against the test dataset is achieved by ‘K-nearest Neighbours’ classifier model, which is 100%, meaning the model was able to predict the landing outcome for the test dataset 100% precisely, despite having one of the lowest best accuracy scores among the models



Section 2

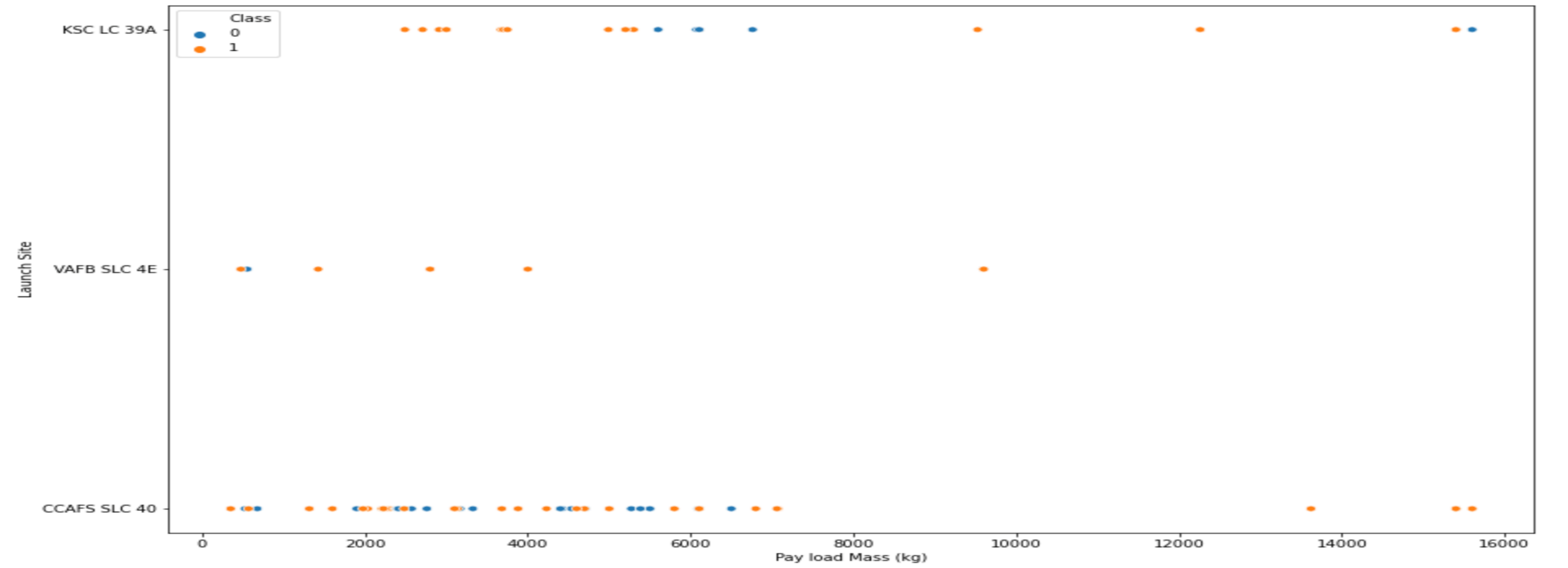
Insights drawn from EDA

Flight Number vs. Launch Site



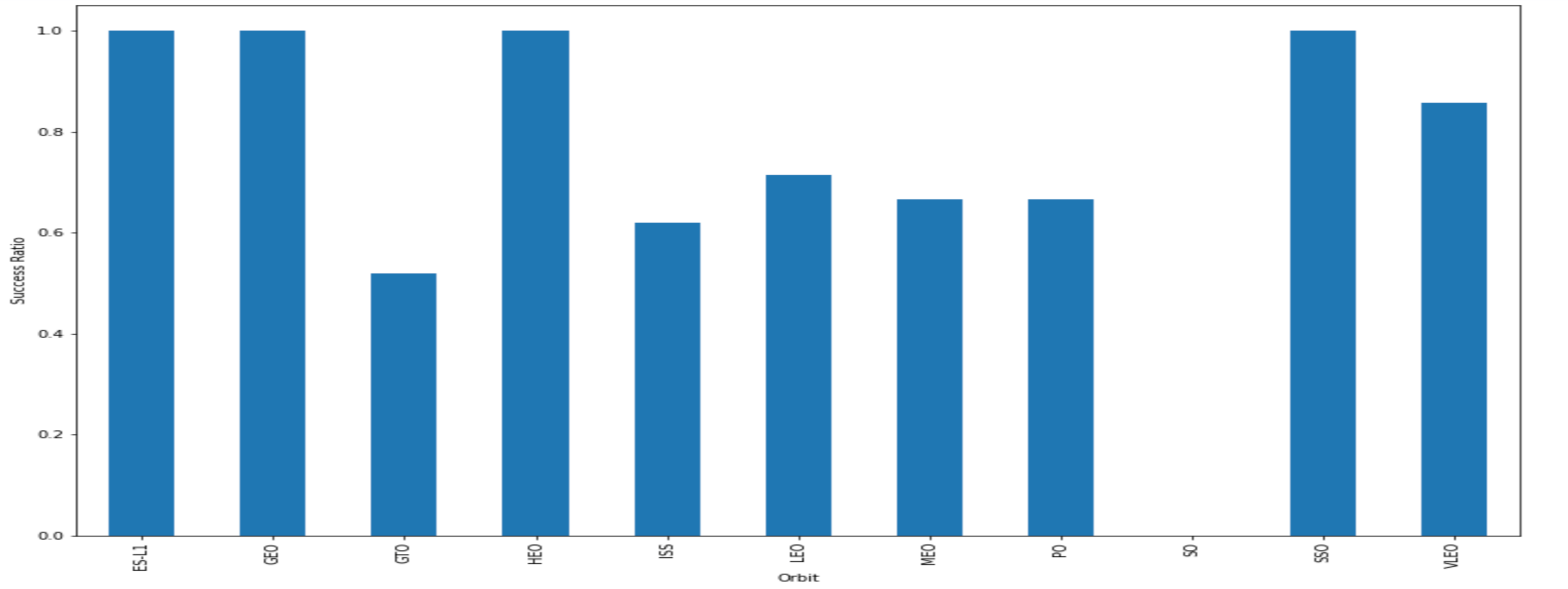
- It is apparent that with increase in the flight numbers the success rate also increased
- There is not any success for 'VAFB SLC 4E' and 'KSC LC 39A' sites for the first 20 flights
- There is not any failure for 'CCAFS SLC 40' and 'KSC LC 39A' sites after 80 flights

Payload vs. Launch Site



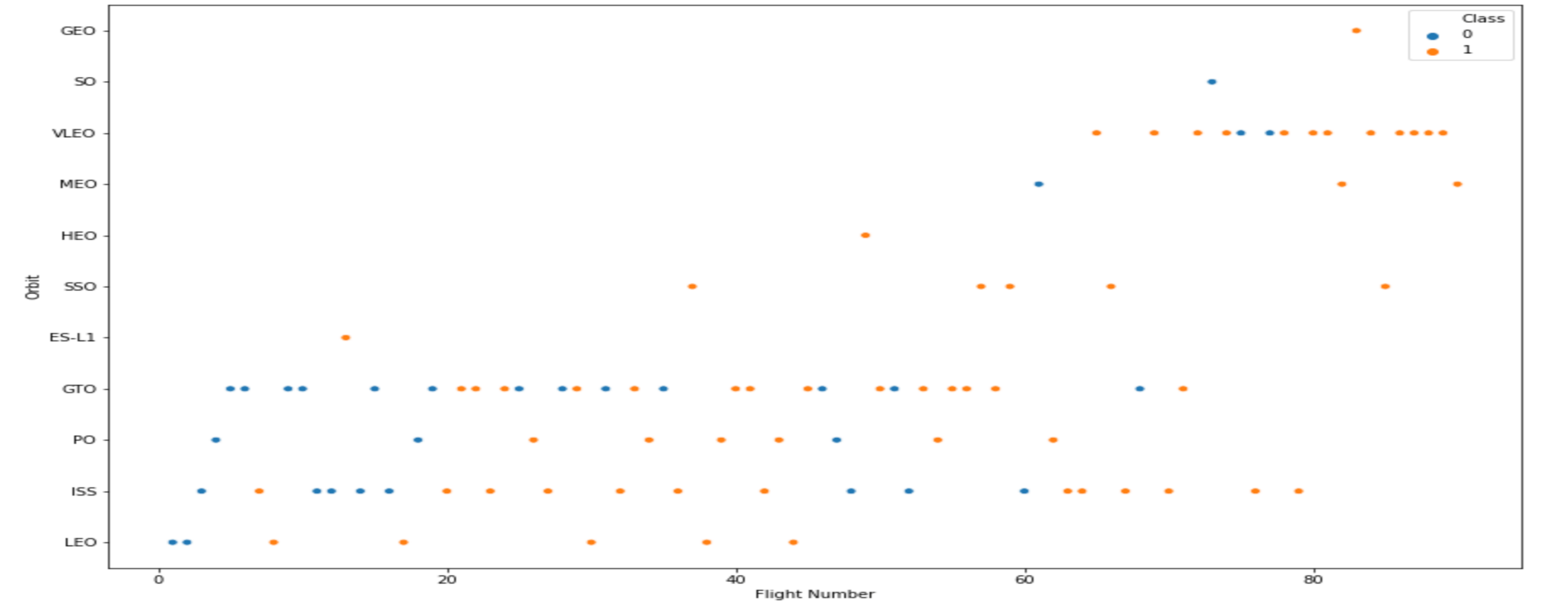
- The ‘CCAFS SLC 40’ site is the most successful for payload over 8,000 kg
- There is not any launches for ‘VAFB SLC 4E’ site for payload over 10,000 kg

Success Rate vs. Orbit Type



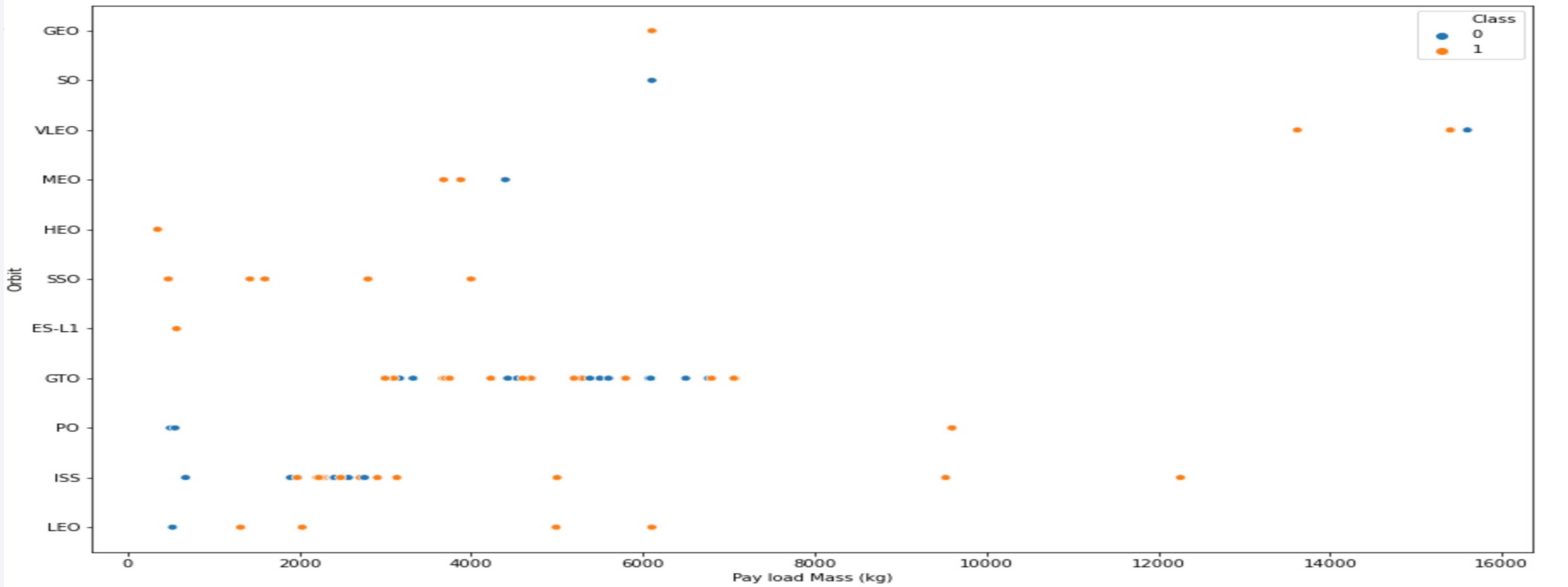
- The highest success rate (100%) is in the 'ES-L1', 'GEO', 'HEO' and 'SSO' orbits
- The lowest success rate (0%) is in the 'SO' orbit

Flight Number vs. Orbit Type



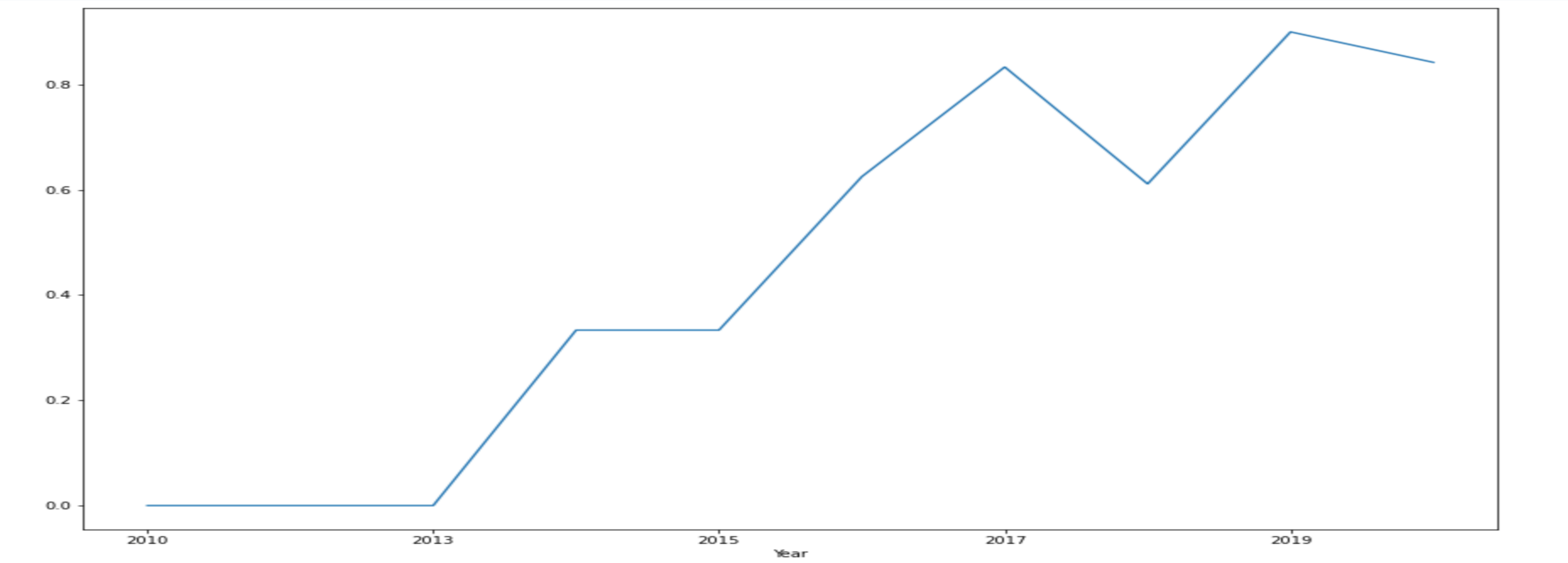
- In the ‘LEO’ orbit the Success appears related to the number of flights
- There seems to be no relationship between flight numbers and orbit when in ‘GTO’ orbit

Payload vs. Orbit Type



- 'LEO' and 'ISS' have the most successful landings for payload over 4,000 kg
- Landing outcome in the 'GTO' orbit does not seem to correlate to payload

Launch Success Yearly Trend



- There is not any successful landings from ‘2010’ to ‘2013’
- After ‘2013’ till ‘2020’, the success rate showed a significant increasing trend

All Launch Site Names

- There seems to be only four unique launch sites for SpaceX

```
%sql select distinct(launch_site) from SPACEXTBL;
```

```
 * ibm_db_sa://mkm98183:***@2d46b6b4-cbf6-40eb-bbce-6251e6ba0300.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:32328/BLUDB
Done.
```

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

- The unique launch sites can be obtained with 'DISTICT' keyword or 'UNIQUE( )' function in the 'SELECT" query followed by the 'launch_site' attribute in the dataset

Launch Site Names Begin with 'CCA'

- 5 records where launch sites begin with `CCA`:

```
%sql select * from SPACEXTBL where launch_site like 'CCA%' limit 5;
```

```
 * ibm_db_sa://mkm98183:***@2d46b6b4-cbf6-40eb-bbce-6251e6ba0300.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:32328/BLUDB
Done.
```

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

- The outcome is achieved using SQL logical operator ‘LIKE’ and keyword ‘LIMIT’

Total Payload Mass

- The total payload carried by boosters from NASA:

```
%sql select sum(payload_mass__kg_) from SPACEXTBL where customer='NASA (CRS)';
```

```
 * ibm_db_sa://mkm98183:***@2d46b6b4-cbf6-40eb-bbce-6251e6ba0300.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:32328/BLUDB
Done.
```

1
45596

- The outcome is achieved using SQL ‘SUM( )’ function for the ‘payload_mass__kg_’ attribute and a ‘WHERE’ clause for ‘customer’ attribute which is equal to (=) ‘NASA (CRS)’

Average Payload Mass by F9 v1.1

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

```
%sql select avg(payload_mass__kg_) from SPACEXTBL where booster_version='F9 v1.1';
```

```
 * ibm_db_sa://mkm98183:***@2d46b6b4-cbf6-40eb-bbce-6251e6ba0300.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:32328/BLUDB
Done.
```

1
2928

- The outcome is achieved using SQL ‘AVG( )’ function for the ‘payload_mass__kg_’ attribute and a ‘WHERE’ clause for ‘booster_version’ attribute which is equal to (=) ‘F9 v1.1’

First Successful Ground Landing Date

- The dates of the first successful landing outcome on ground pad:

```
%sql select min("DATE") from SPACEXTBL where landing__outcome='Success (ground pad)';
```

```
 * ibm_db_sa://mkm98183:***@2d46b6b4-cbf6-40eb-bbce-6251e6ba0300.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:32328/BLUDB
Done.
```

1
2015-12-22

- The outcome is achieved using SQL 'MIN( )' function for the 'DATE' attribute and a 'WHERE' clause for 'landing__outcome' attribute which is equal to (=) 'Success (ground pad)'

Successful Drone Ship Landing with Payload between 4000 and 6000

- 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, landing__outcome, payload_mass__kg_ from SPACEXTBL where landing__outcome='Success (drone ship)' and payload_mass__kg_ between 4000 and 6000;
```

```
 * ibm_db_sa://mkm98183:***@2d46b6b4-cbf6-40eb-bbce-6251e6ba0300.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:32328/BLUDB
Done.
```

booster_version	landing__outcome	payload_mass__kg_
F9 FT B1022	Success (drone ship)	4696
F9 FT B1026	Success (drone ship)	4600
F9 FT B1021.2	Success (drone ship)	5300
F9 FT B1031.2	Success (drone ship)	5200

- The outcome is achieved using a SQL 'WHERE' clause for 'landing__outcome' attribute which is equal to (=) 'Success (drone ship)' and 'payload_mass__kg_' attribute with 'BERWEEN' keyword to range between '4000' to '6000' values only

Total Number of Successful and Failure Mission Outcomes

- The total number of successful and failure mission outcomes:

```
%sql select mission_outcome, count(mission_outcome) as total from SPACEXTBL group by mission_outcome order by total desc;
```

```
 * ibm_db_sa://mkm98183:***@2d46b6b4-cbf6-40eb-bbce-6251e6ba0300.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:32328/BLUDB
Done.
```

mission_outcome	total
Success	99
Failure (in flight)	1
Success (payload status unclear)	1

- The outcome is achieved using SQL 'COUNT( )' function for the 'mission_outcome' attribute and a 'GROUP BY' clause for 'mission_outcome' attribute, where the count as shown as alias 'total' and the result is ordered by 'total' using the 'ORDER BY' keyword

Boosters Carried Maximum Payload

- 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);
```

```
 * ibm_db_sa://mkm98183:***@2d46b6b4-cbf6-40eb-bbce-6251e6ba0300.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:32328/BLUDB
Done.
```

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

- The outcome is achieved using a SQL ‘WHERE’ clause for ‘payload_mass__kg_’ attribute which is equal to (=) the value from the subquery which uses the ‘MAX( )’ function for the ‘payload_mass__kg_’ attribute

2015 Launch Records

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

```
%sql select landing__outcome, booster_version, launch_site, "DATE" from SPACEXTBL where landing__outcome like '%Failure%' and year("DATE")=2015;
```

```
 * ibm_db_sa://mkm98183:***@2d46b6b4-cbf6-40eb-bbce-6251e6ba0300.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:32328/BLUDB
Done.
```

landing__outcome	booster_version	launch_site	DATE
Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40	2015-01-10
Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40	2015-04-14

- The outcome is achieved using a SQL ‘WHERE’ clause for ‘landing__outcome’ attribute with ‘LIKE’ logical operator for only ‘Failure’ and ‘YEAR’ function was used for ‘DATE’ attribute which is equal to (=) ‘2015’

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

- Ranking of 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 landing__outcome, count(landing__outcome) as total from SPACEXTBL \
    where "DATE" between '2010-06-04' and '2017-03-20' group by landing__outcome order by total desc;
```

```
 * ibm_db_sa://mkm98183:***@2d46b6b4-cbf6-40eb-bbce-6251e6ba0300.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:32328/BLUDB
Done.
```

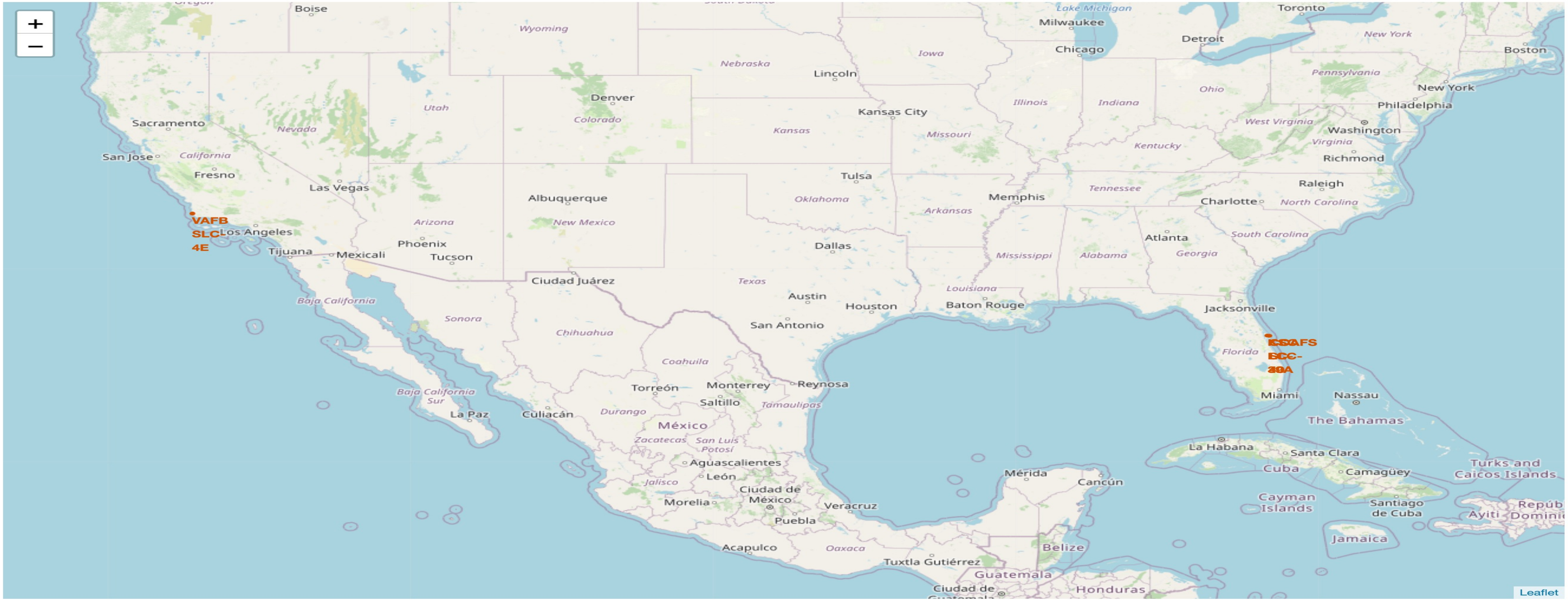
landing__outcome	total
No attempt	10
Failure (drone ship)	5
Success (drone ship)	5
Controlled (ocean)	3
Success (ground pad)	3
Failure (parachute)	2
Uncontrolled (ocean)	2
Precluded (drone ship)	1

- The outcome is achieved using SQL 'COUNT( )' function for the 'landing__outcome' attribute with a 'WHERE' clause with 'BETWEEN' keyword for 'DATE' attribute for date between the expected range and a 'GROUP BY' clause for 'landing__outcome' attribute, where the count as shown as alias 'total' and the result is ordered by 'total' using the 'ORDER BY' keyword in the descending order using 'DESC' keyword

Section 3

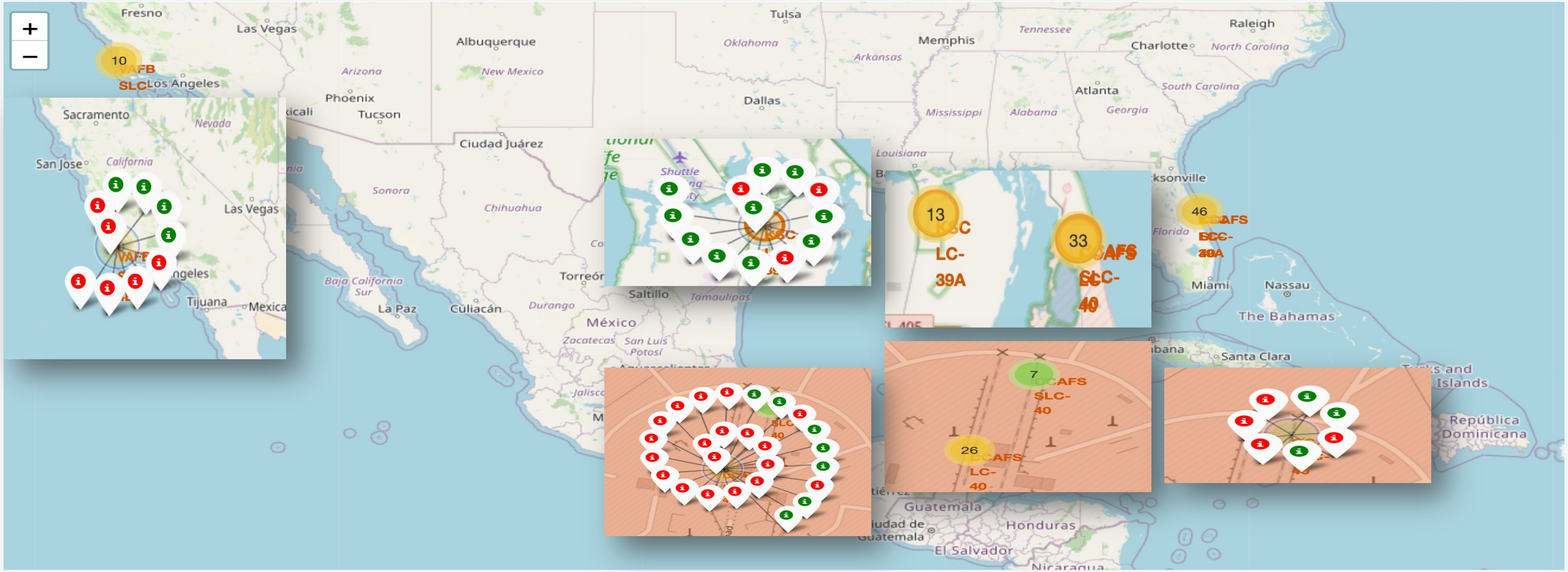
Launch Sites Proximities Analysis

Launch Sites Location



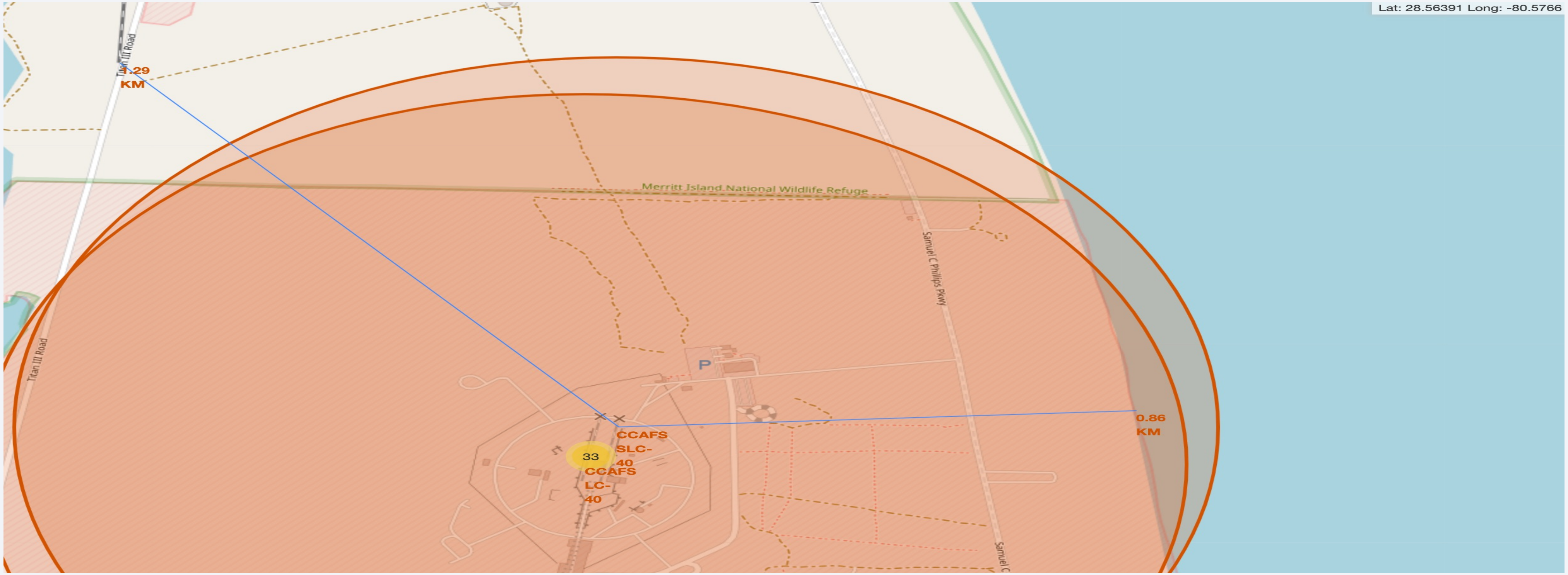
- All the launch sites are near cost lines, perhaps for safety and to prevent accidents
- The launch sites are also not very far from the roads and railroads

Launch Sites with Coloured Markers for Success and Failure



- The green markers indicate success and red markers indicates failure for a particular site
- Most number of lunches (26), success and failure combined, were launched from ‘CCAFS SLC-40’ site

CCAFS SLC-40 Proximities to Railroad and Coastline

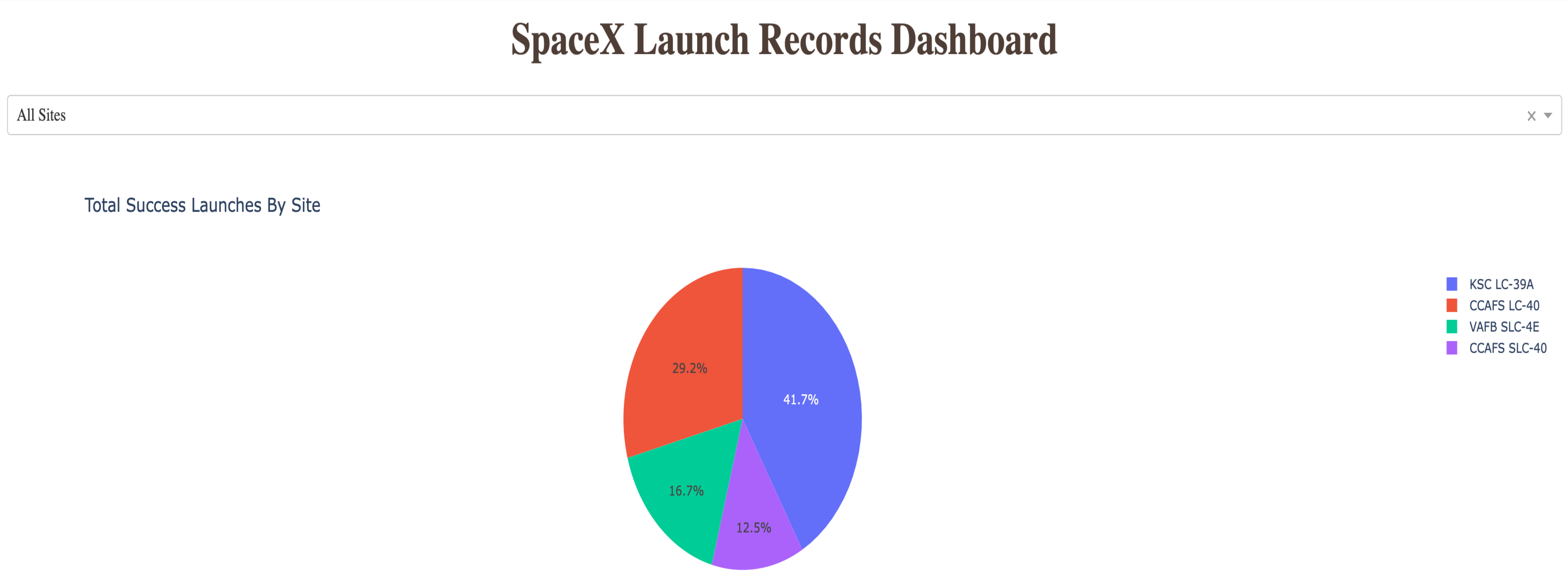


- Proximities of launch sites to infrastructure and transport is important in case of crash landing
- ‘CCAFS SLC-40’ site is only 0.86 km far away from cost line and 1.29 km from the nearest railroad, but far away from inhabited areas, therefore, not hazardous

Section 4

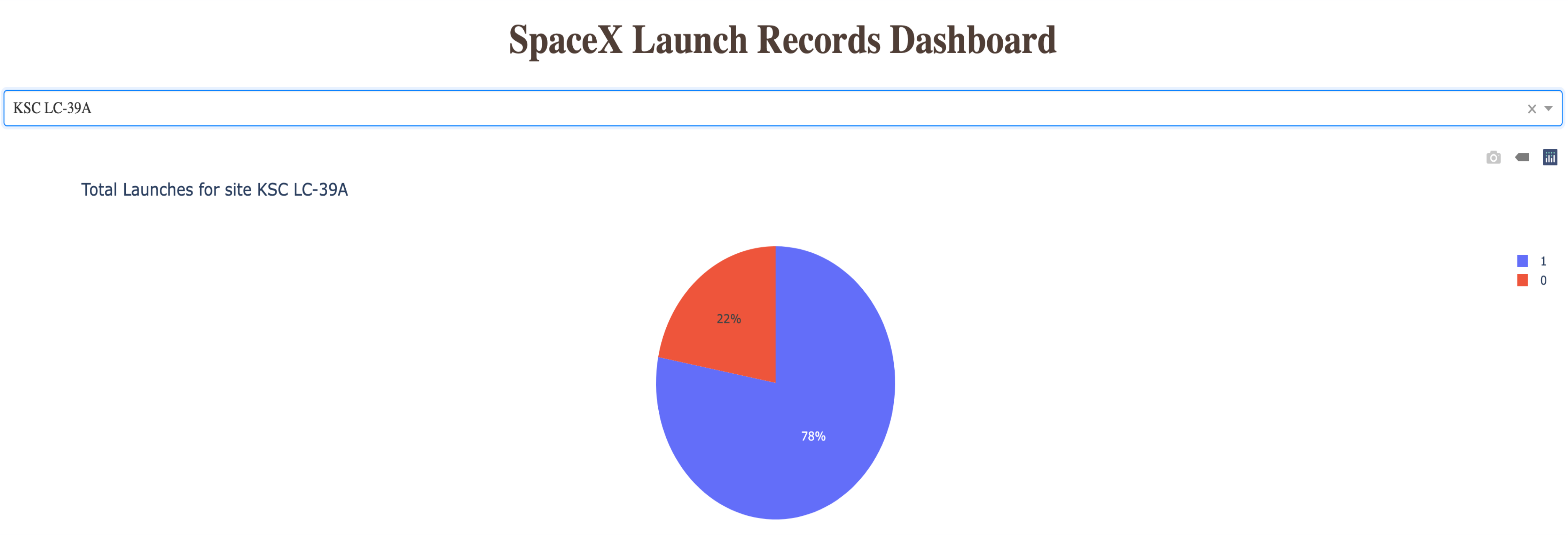
Build a Dashboard with Plotly Dash

SpaceX Successful Launches by Sites



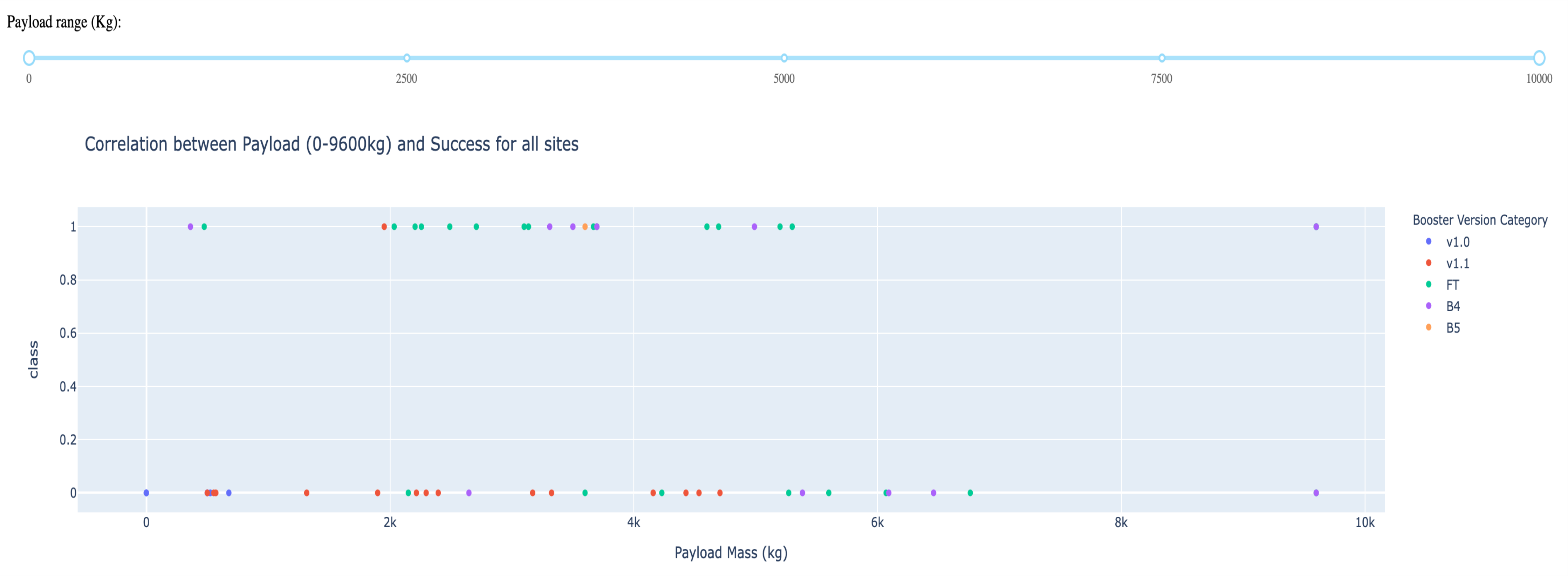
- Success rate of the sites is a very important factor to choose the right site for a successful launch, for example, it is 41.7% more likely to launch successfully from the 'KSC LC-39A' site

Success Launch Rate for KSC LC-39A Site



- 78% launches were successful from this site
- It is likely to choose this site as the most reliable for successful launches

Payload vs. Launch Outcome by Booster Version for all Sites



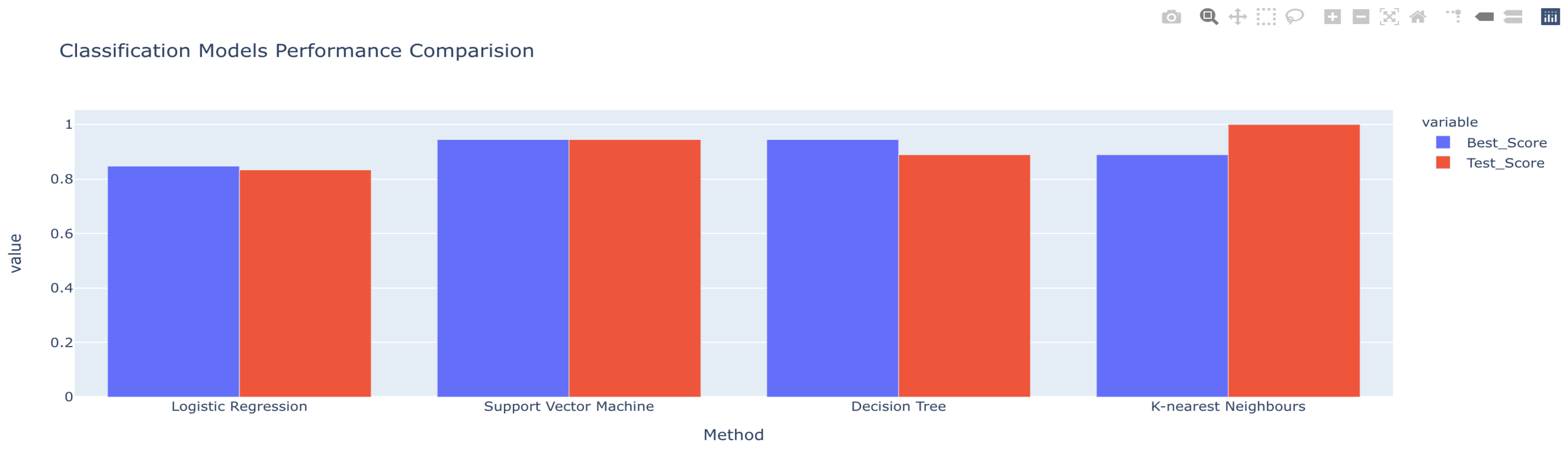
- Booster version ‘B4’ is the most successful for payload over 8,000 kg
- Booster version ‘FT’ has the most successful rate for payload ranging from 2,000 to 4,000 kg

Section 5

Predictive Analysis (Classification)

Classification Accuracy

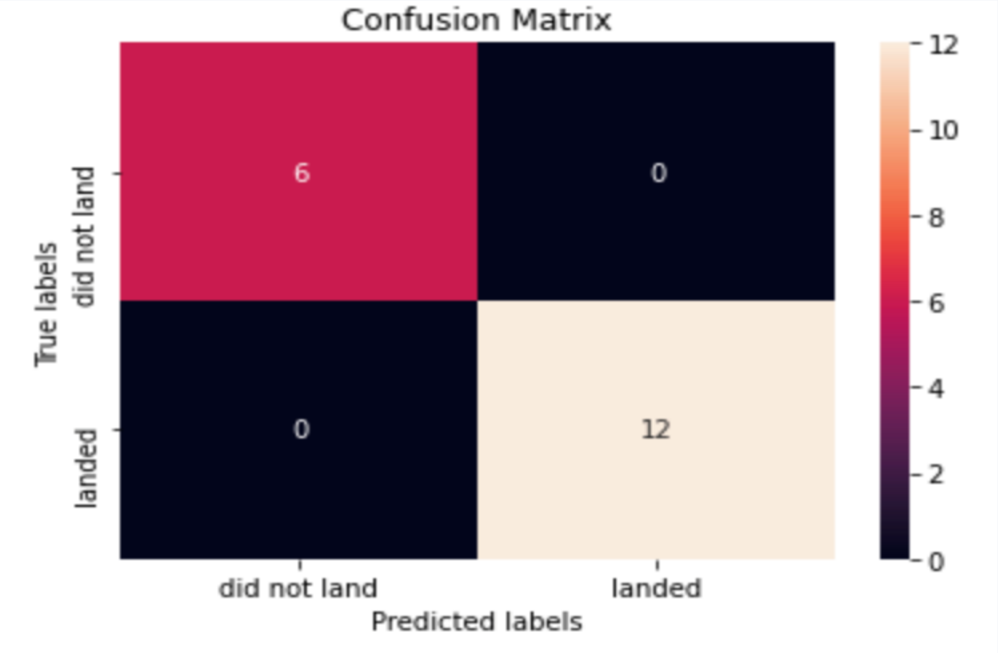
- 4 Classification models were built and their best accuracy and the accuracy against the test dataset were plotted in a bar chart



- The ‘Support Vector Machine’ and the ‘Decision Tree’ models have the highest accuracy of 94%, where the highest accuracy (100%) against the test dataset is achieved by the ‘K-nearest Neighbours’

Confusion Matrix

- The confusion matrix of the best performing model against the test dataset



- The confusion matrix for ‘K-nearest Neighbours’ shows that ‘0’ False Positive and ‘0’ False Negative were predicted by that model against the test dataset
- Thus, this model was 100% successful to predict success and failure against the test dataset

Conclusions

- Different data collection sources were utilized to fulfil data requirements
- Data Wrangling helped prepare the data for supervised learning at the later stage
- EDA helped understand the data and the stories and patterns in them through queries and visualization
- The interactive analytics helped find proximities of sites and analyze success rate for launch sites and payload range through quick dropdown selection and payload ranging

➢ The best launch site is ‘KSC LC-39A’

➢ Although most of the mission was successful, the landing outcome trends to improve within time and increasing number of flights, and perhaps, improvement of technology

Appendix

- GitHub Link to Project:
  - https://github.com/khhaledahmaad/IBM-Data-Science-Professional-Certificate.git
- Tools used for flowcharts:
  - Diagram.net: https://www.diagrams.net/
- Thanks, and accreditation:
  - ❖ IBM: https://www.ibm.com/
  - ❖ Coursera: https://www.coursera.org/

Thank you!