IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Abdul Qadeer Afridi
19/02/2022

Outline

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

Executive Summary

- The entire project consists of multiple operations on data and each operation is associated with a particular methodology in order to complete it. Every crucial step in the process required a specialized method and by the means of these the project was completed.
- With great pride it can be claimed that all results obtained were satisfactory. Each of the results obtained provide intriguing insights regarding the data. Most of the insights would be gone unnoticed if the data analysis wasn’t performed.

Introduction

- The project is based upon the commercial space exploration company ‘SpaceX’. SpaceX launches flights to the various orbits of earth at respective distances. The main agenda of the project was to figure if the first stage of the SpaceX flight would land back successfully or not.
- Falcon 9 Aircraft is the main focus of the project. Falcon 9 costs $65 million because SpaceX reuses the aircraft once it lands successfully.
- Various problems were formulated in order to gain insights such as ‘is there a relation between the flight number and the payload mass’? , or ‘is the relationship between launch site and the version of boosters used which determines success or failure of landing’ etc.

Section 1

Methodology

Methodology

Executive Summary

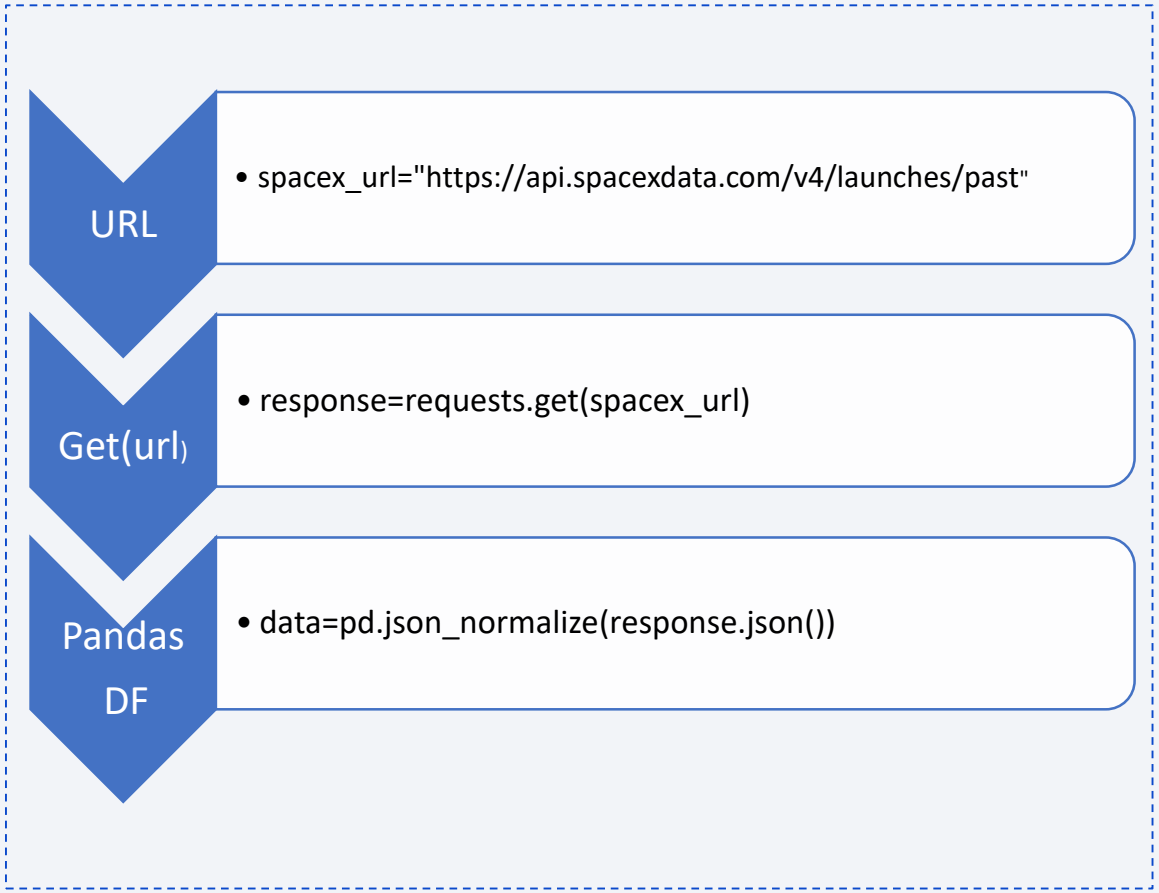
- Data collection methodology:
  - Using a url of SpaceX website and requests library from python data was read into a pandas data frame.
- Perform data wrangling
  - Performed using the Wikipedia records and reading the data to HTML Tables
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - By using various python libraries such as Scikit Learn and using the in-built functions such as KNN and Confusion Matrix.

Data Collection

- Describe how data sets were collected: Data is firstly collected by locating the URL of the required site. Then a connection is used by the means of an Application Program Interface(API). Then the requests library in the python is used to load the data from the website. Finally the data loaded is read into a pandas data frame for further exploratory analysis.
- You need to present your data collection process use key phrases and flowchart

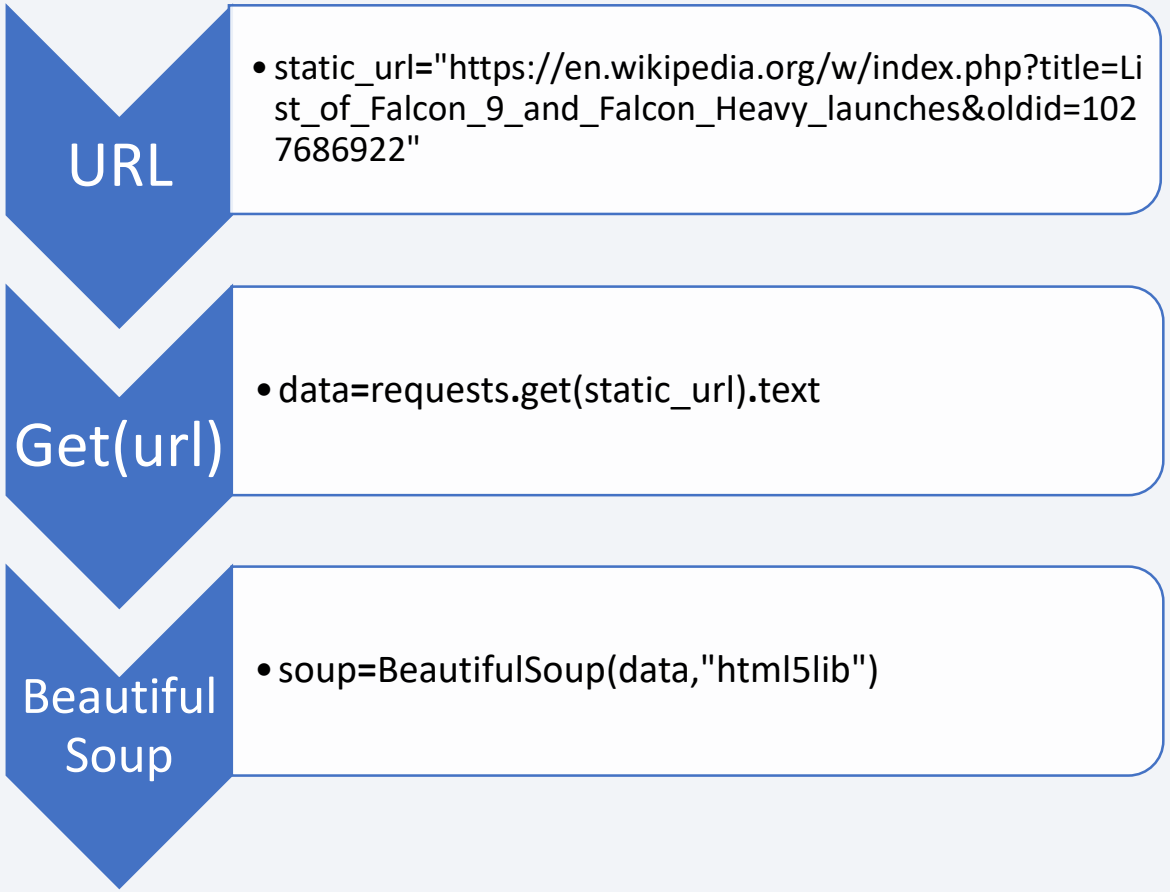
Data Collection - SpaceX API

- To collect the data SpaceX REST a number of calls were made, they are shown in the adjacent flowchart.
- The following is a link to the github ipynb file on data collection.
- https://github.com/aqafridi/Data-Science-Specialization/blob/main/Applied%20Data%20Science%20Capstone/5.%20Present%20Data-Driven%20Insights/Applied%20DS%20EDA%20Data%20Wrang.ipynb



Data Collection - Scraping

- WEB Scraping is another popular method of Data Collection using html5lib.
- Following is a GitHub URL of Web Scraping Notebook.
- https://github.com/aqafridi/Data-Science-Specialization/blob/main/Applied%20Data%20Science%20Capstone/5.%20Present%20Data-Driven%20Insights/Applied%20DS%20Web%20Scrape.ipynb



Data Wrangling

- Once the data was loaded to API, missing values were found and replaced. Then various relationships were established between Launch Site, Orbit and Outcomes using .value_counts(). Finally labels for Outcomes were created.
- https://github.com/aqafridi/Data-Science-Specialization/blob/main/Applied%20Data%20Science%20Capstone/5.%20Present%20Data-Driven%20Insights/Applied%20DS%20EDA%20Data%20Wrang.ipynb

Reading

- df=pd.read_csv()
- df.isnull().sum()/df.count()*100

Value Counts

- launchsitecount=df.value_counts('LaunchSite')
- orbitcount=df.value_counts('Orbit')
- landing_outcomes=df.value_counts('Outcome')

Label Making

- **for** i,outcome**in**enumerate(landing_outcomes.keys()):print(i,outcome)
- bad_outcomes=set(landing_outcomes.keys()[[1,3,5,6,7]])
- landing_class=[]**for**outcome**in**df['Outcome']:**if**outcome **in**bad_outcomes:landing_class.append(0)**else**:landing_class.append(1)

EDA with Data Visualization

- The charts plotted were:
- Flight No v/s Launch Site
- Payload v/s Launch Site
- Bar graph of success rate of each Orbit type
- Flight No v/s Orbit type
- Payload v/s Orbit type
- Yearly Launch Success Trend line graph

- The reason to plot the scatter plots was to understand the relationship between corresponding factors. The scatter plots enable us to understand better the relationship between the factors and help us gain unknown insights.
- https://github.com/aqafridi/Data-Science-Specialization/blob/main/Applied%20Data%20Science%20Capstone/5.%20Present%20Data-Driven%20Insights/Applied%20DS%20EDA%20with%20DV.ipynb

EDA with SQL

➢ The various SQL Queries performed were:

- SELECT Distinct LAUNCH_SITE FROM SPACEXTBL
- SELECT * FROM SPACEXTBL WHERE LAUNCH_SITE LIKE 'CCA%' LIMIT 5
- SELECT SUM(PAYLOAD_MASS__KG_) FROM SPACEXTBL WHERE CUSTOMER='NASA (CRS)'
- SELECT min(DATE) FROM SPACEXTBL WHERE LANDING__OUTCOME='Success (ground pad)'
- SELECT BOOSTER_VERSION FROM SPACEXTBL WHERE PAYLOAD_MASS__KG_ between 4000 and 6000 AND LANDING__OUTCOME='Success (drone ship)'
- SELECT COUNT(*) FROM SPACEXTBL WHERE MISSION_OUTCOME LIKE '%Success%' OR MISSION_OUTCOME LIKE '%Failure%'
- SELECT BOOSTER_VERSION FROM SPACEXTBL WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTBL)

EDA with SQL

- SELECT TO_CHAR(TO_DATE(MONTH("DATE"), 'MM'), 'MONTH') AS MONTH_NAME, LANDING__OUTCOME AS LANDING__OUTCOME, BOOSTER_VERSION AS BOOSTER_VERSION, LAUNCH_SITE AS LAUNCH_SITE FROM SPACEXTBL WHERE LANDING__OUTCOME = 'Failure (drone ship)' AND "DATE" LIKE '%2015%‘
- SELECT "DATE", COUNT(LANDING__OUTCOME) as COUNT FROM SPACEXTBL WHERE "DATE" BETWEEN '2010-06-04' and '2017-03-20' AND LANDING__OUTCOME LIKE '%Success%' GROUP BY "DATE" ORDER BY COUNT(LANDING__OUTCOME) DESC

- GitHub Link:
- https://github.com/aqafridi/Data-Science-Specialization/blob/main/Applied%20Data%20Science%20Capstone/5.%20Present%20Data-Driven%20Insights/Applied%20DS%20EDA%20with%20SQL.ipynb

Build an Interactive Map with Folium

- Different markers have been used while generating maps using Folium. We have mainly used Red Circles, Yellow Circles, Green and Red Markers.
- The Red Circles indicate the Launch Site, when zoomed in, yellow circles can be seen inside red ones. The yellow circles indicate the launch pads along with the number of launches conducted there.
- The green markers indicate a successful landing and red markers indicate a failed landing in the corresponding launch pad.
- Finally we have marked the distance of launch pads from highways and nearest coastlines.
- https://github.com/aqafridi/Data-Science-Specialization/blob/main/Applied%20Data%20Science%20Capstone/5.%20Present%20Data-Driven%20Insights/Applied%20DS%20Int%20VA.ipynb

Build a Dashboard with Plotly Dash

- We've used a pie chart that tells us about the percentage of launches from all launch pads respectively. Scatter plot is used to know the details of the launch from that launch pad.
- The pie chart informs us about the number of launches conducted from each launch pad and their landing success and failure percentage.
- The scatter plots gives us three dimensional information that is payload mass, booster version and their landing success or failure.
- https://github.com/aqafridi/Data-Science-Specialization/blob/main/Applied%20Data%20Science%20Capstone/5.%20Present%20Data-Driven%20Insights/spacex_dash_app.py

Predictive Analysis (Classification)

- First step to build the classification model was to standardize the data and assign it back to variable X. Then Test-Train split method was used to split data for training and testing.
- Then data was used in 4 classification models. Logistic Regression, SVM, Decision Tree and K-Nearest Neighbor.
- Using the confusion matrix the models were evaluated and false positives or negatives were identified and rectified.
- Finally parameters were updated and each model was checked for its accuracy.
- https://github.com/aqafridi/Data-Science-Specialization/blob/main/Applied%20Data%20Science%20Capstone/5.%20Present%20Data-Driven%20Insights/Applied%20DS%20ML%20Pred.ipynb

Exploratory Data Analysis Results

Launch Site	No. of Launches
CCAFS SLC 40	55
KSC LC 39A	22
VAFB SLC 4E	13

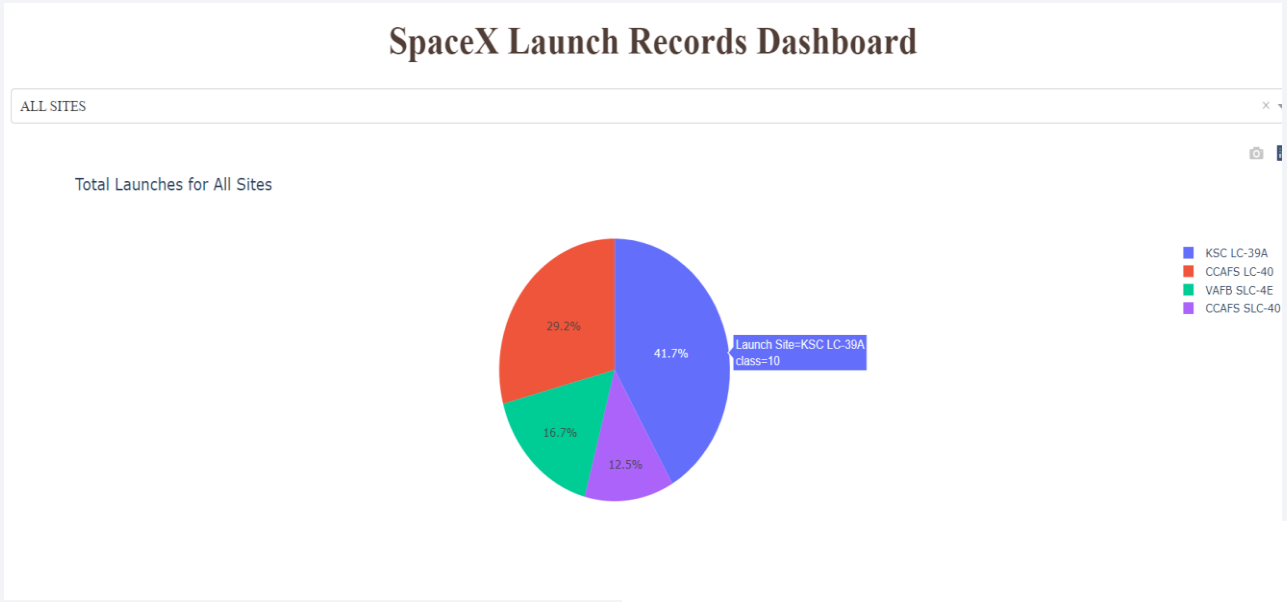
Orbit	No. of Flights
GTO	27
ISS	21
VLEO	14
PO	9
LEO	7
SSO	5
MEO	3
ES-L1	1
GEO	1
HEO	1
SO	1

Exploratory Data Analysis Results

Landing Outcome	No. of Flights
True ASDS	41
None None	19
True RTLS	14
False ASDS	6
True Ocean	5
False Ocean	2
None ASDS	2
False RTLS	1

- Average Successful Landing Outcome= 67%
- Total Payload Mass by NASA CRS= 45596Kg
- First Successful Ground Landing= 22/12/2015
- Total Number of missions= 101

Interactive Analytics Results



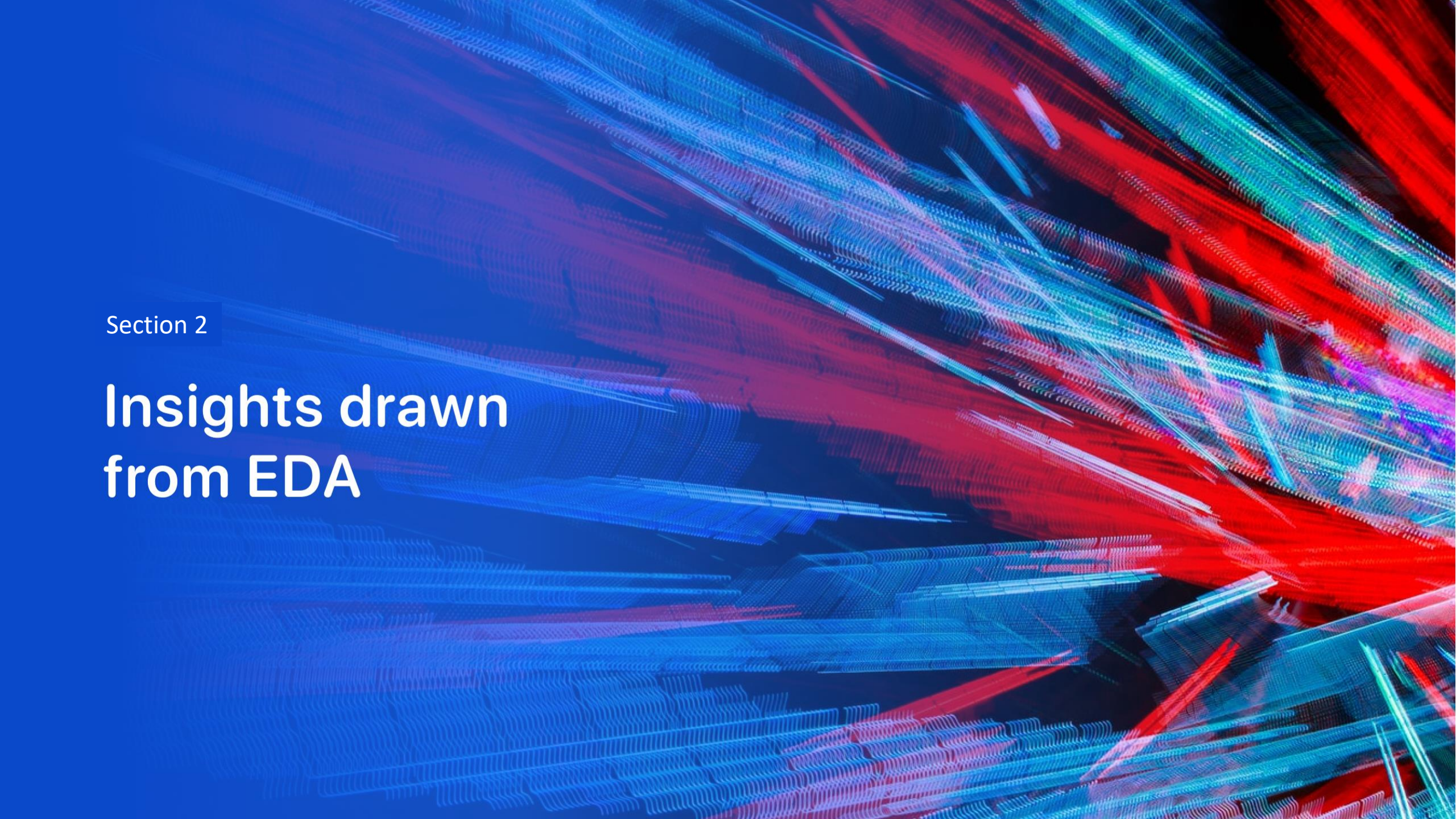
Predictive Analysis Results

Training Data Accuracy

- Logistic Regression: 0.846428571428571 3
- Support Vector Machine: 0.8482142857142856
- Decision Tree: 0.8892857142857145
- K-Nearest Neighbor: 0.8482142857142858

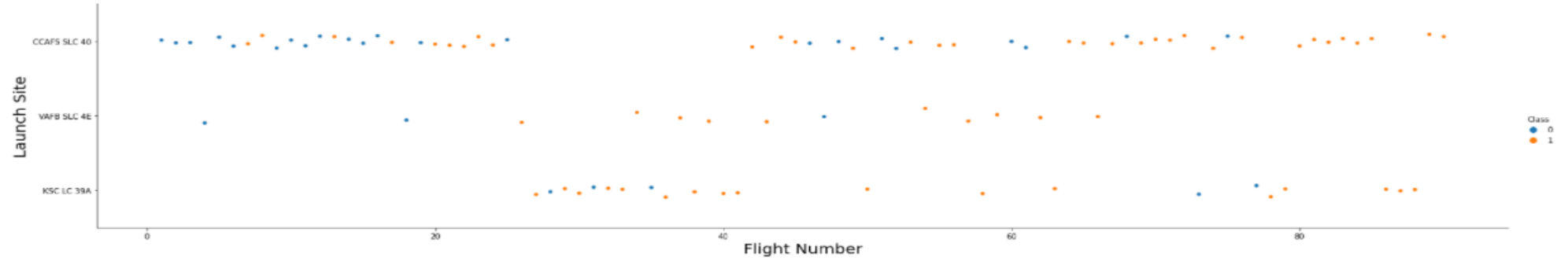
Testing Data Accuracy

- Logistic Regression: 0.8333333333333333
- Support Vector Machine: 0.8333333333333334
- Decision Tree: 0.9444444444444444
- K-Nearest Neighbor: 0.8333333333333334


Section 2
Insights drawn from EDA

Flight Number vs. Launch Site

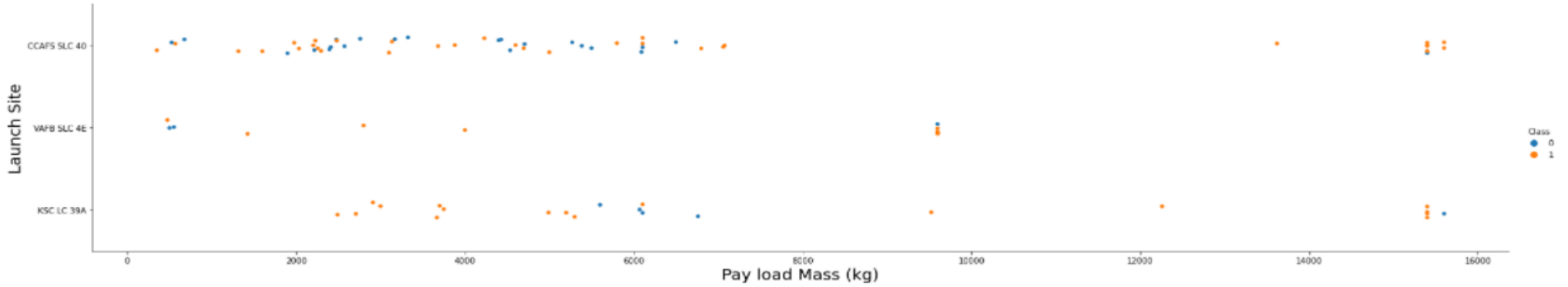
```
In [4]: # Plot a scatter point chart with x axis to be Flight Number and y axis to be the launch site, and hue to be the class value
sns.catplot(y="LaunchSite", x="FlightNumber", hue="Class", data=df, aspect = 5)
plt.xlabel("Flight Number",fontsize=20)
plt.ylabel("Launch Site",fontsize=20)
plt.show()
```



- The scatter plot gives us the relationship between flight number and launch site.
- It can be seen that after 80 flights there have been no failed missions on any launch site.
- Majority of the flights were launched from CCAFS SLC 40 launch site.

Payload vs. Launch Site

```
In [6]: # Plot a scatter point chart with x axis to be Pay Load Mass (kg) and y axis to be the launch site, and hue to be the class valu
        e
        sns.catplot(y="LaunchSite", x="PayloadMass", hue="Class", data=df, aspect = 5)
        plt.ylabel("Launch Site",fontsize=20)
        plt.xlabel("Pay load Mass (kg)",fontsize=20)
        plt.show()
```

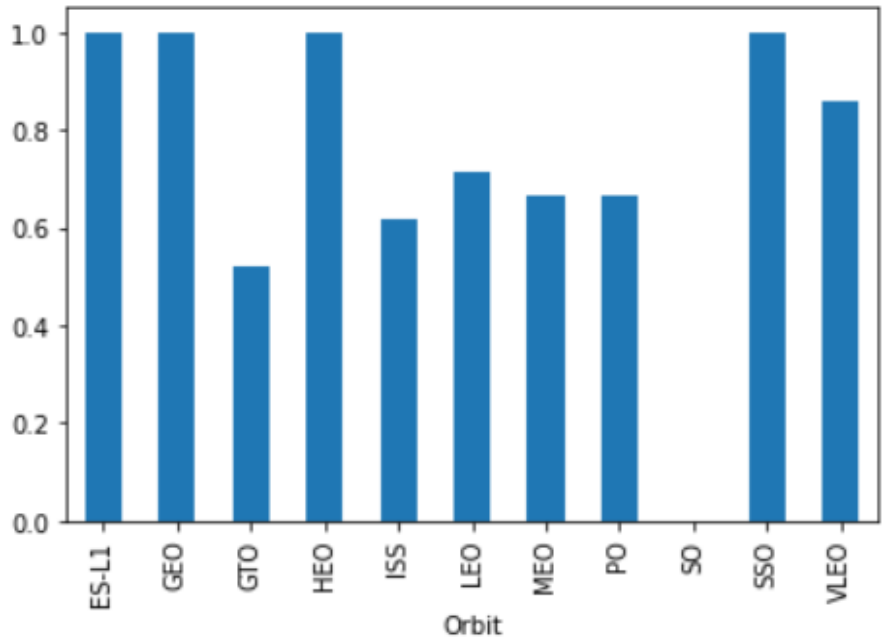


- The orange markers indicate successful landings and blue markers indicate failed landings.
- Majority of the missions carried payload mass less than 8000kg and were launched from CCAFS SLC 40 launch site.

Success Rate vs. Orbit Type

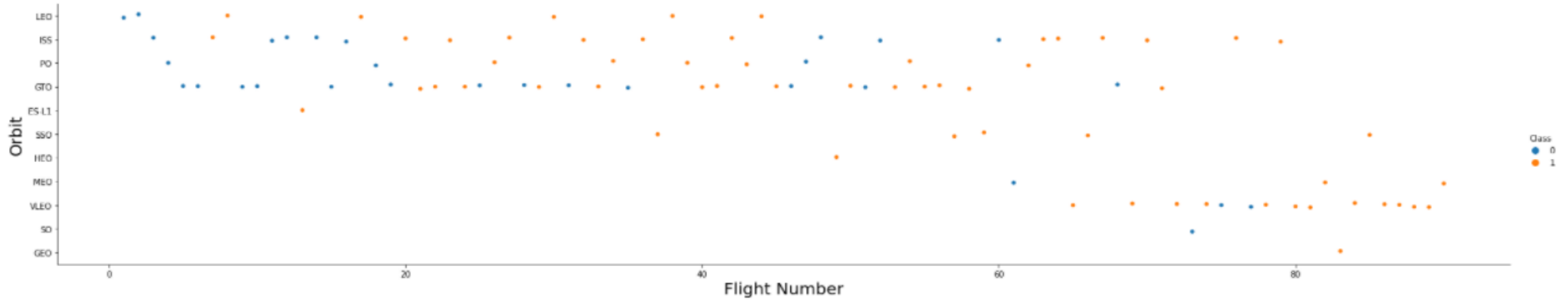
- The most successful missions have been to orbits ES-L1, GEO, HEO, SSO.
- Orbit GTO has the least successful missions with success rate of less than 60%.
- ISS, LEO, MEO,PO have a success rate of greater than 60%

```
In [7]: # HINT use groupby method on Orbit column and get the
        df.groupby("Orbit").mean()['Class'].plot(kind='bar')
Out[7]: <AxesSubplot:xlabel='Orbit'>
```



Flight Number vs. Orbit Type

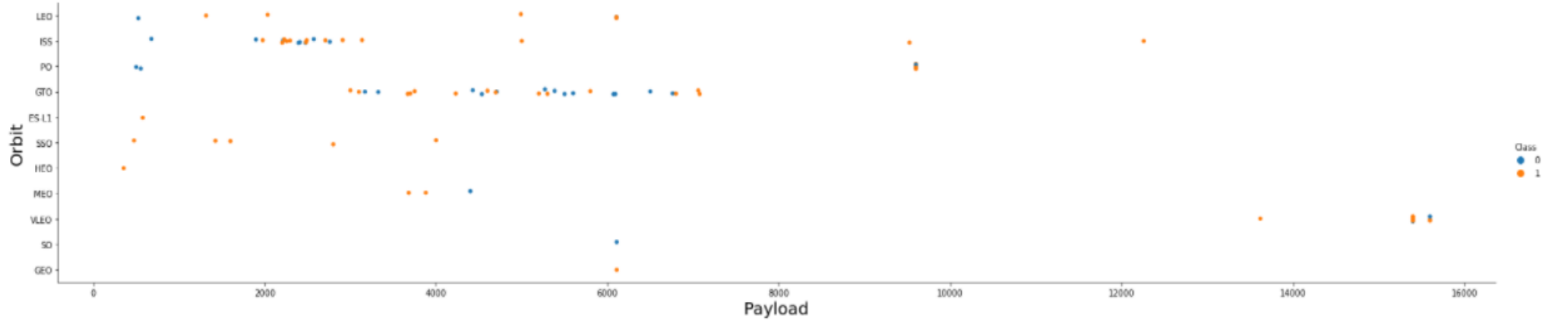
```
In [8]: # Plot a scatter point chart with x axis to be FlightNumber and y axis to be the Orbit, and hue to be the class value
        sns.catplot(y="Orbit", x="FlightNumber", hue="Class", data=df, aspect = 5)
        plt.xlabel("Flight Number",fontsize=20)
        plt.ylabel("Orbit",fontsize=20)
        plt.show()
```



- The plot gives us a relation between flight number and orbit type.
- It is evident that the success rate of missions has increased after 80 flights.
- Prior too that failure rate of the missions was in a considerable amount.

Payload vs. Orbit Type

In [10]:
```
# Plot a scatter point chart with x axis to be Payload and y axis to be the Orbit, and hue to be the class value
sns.catplot(x="PayloadMass", y="Orbit", hue="Class", data=df, aspect = 5)
plt.xlabel("Payload",fontsize=20)
plt.ylabel("Orbit",fontsize=20)
plt.show()
```

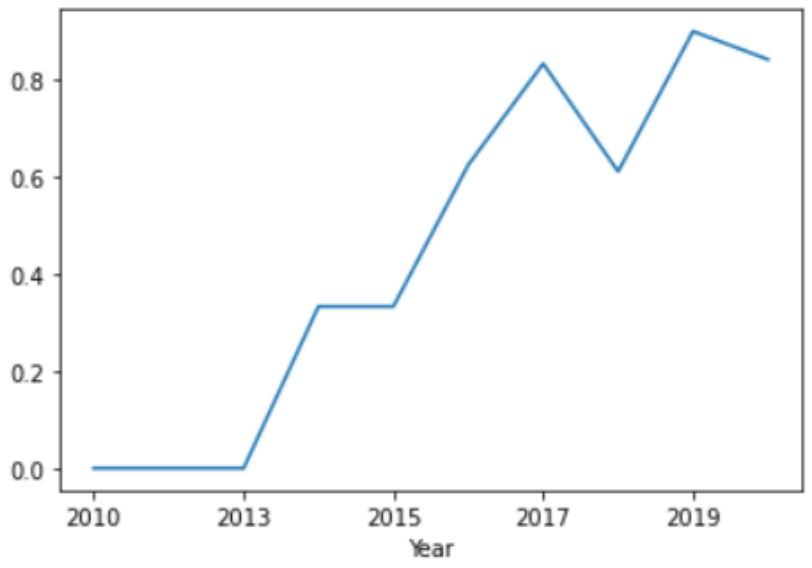


- This plot tell us us in detail about what kind of payload mass was sent to which orbit. This is helpful in further predictions regarding mass, orbit type and success of mission.
- It can be seen that masses less than 4000kg are successful landings in a few orbits.

Launch Success Yearly Trend

- It can be noted that the success rate of missions increased only after 2013.
- Success rate was constant in between years from 2013 and 2015.
- It grew significantly until 2017 after which it took a hit in 2018 and stabilized back in 2019.

```
In [13]: # Plot a line chart with x axis to be the extracted year
         Extract_year(df["Date"])
         zipped = zip(df['Date'], df['Orbit'], df['Outcome'],df['C
         df1=pd.DataFrame(zipped, columns=['Date', 'Orbit', 'Outco
         df1
         df1.groupby("Year").mean()['Class'].plot(kind='line')
Out[13]: <AxesSubplot:xlabel='Year'>
```



All Launch Site Names

- The image shows the query and result to find the names of all unique launch sites from given data.
- The query implies the function DISTINCT which is used to find out unique names in LAUNCH_SITE column in SPACEXTBL table.

```
In [6]: %sql SELECT Distinct LAUNCH_SITE FROM SPACEXTBL
 * ibm_db_sa://ynd63707:***@0c77d6f2-5da9-48a9-81f8-86b
Done.
```

Out[6]:

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

Launch Site Names Begin with 'CCA'

```
In [7]: %sql SELECT * FROM SPACEXTBL WHERE LAUNCH_SITE LIKE 'CCA%' LIMIT 5
 * ibm_db_sa://ynd63707:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/BLUDB
Done.
```

Out[7]:

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 query looks for all the Launch Site names which has CCA in their names.
- We have used the LIKE ‘CCA%’ query and LIMIT is set to 5. It displays only 5 outputs.

Total Payload Mass

```
In [9]: %sql SELECT SUM(PAYLOAD_MASS__KG_) FROM SPACEXTBL WHERE CUSTOMER='NASA (CRS)'
 * ibm_db_sa://ynd63707:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1o
Done.
```

Out[9]:

1
45596

- The boosters from NASA has totally carried 45596kg in terms of payload mass.
- Our query here adds data in payload mass column and presents it us.
- Since we have used query CUSTOMER = ‘NASA (CRS)’ it only adds payload mass of NASA boosters

Average Payload Mass by F9 v1.1

Task 4

Display average payload mass carried by booster version F9 v1.1

```
In [11]: %sql SELECT AVG(PAYLOAD_MASS__KG_) FROM SPACEXTBL WHERE BOOSTER_VERSION = 'F9 v1.1'
 * ibm_db_sa://ynd63707:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.dat
Done.
```

Out[11]:

1
2928

- The average mass carried by F9 v1.1 boosters is 2928kg.
- The query averages the payload column where the booster version is F9 v1.1

First Successful Ground Landing Date

```
In [13]: %sql SELECT min(DATE) FROM SPACEXTBL WHERE LANDING__OUTCOME='Success (ground pad)'
 * ibm_db_sa://ynd63707:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lc
Done.
```

Out[13]:

1
2015-12-22

- The first successful landing on a ground pad was on 22-12-2015.
- The query selects the minimum of the date in landing outcome column where outcome was (‘Success(ground pad)’).

Successful Drone Ship Landing with Payload between 4000 and 6000

```
In [14]: %sql SELECT BOOSTER_VERSION FROM SPACEXTBL WHERE PAYLOAD_MASS__KG_ between 4000 and 6000 AND LANDING__OUTCOME='Success (drone sh
         ip)'
          * ibm_db_sa://ynd63707:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/BLUDB
         Done.
```

Out[14]:

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

- There are 4 booster versions which have carried a mass between 4000kg and 6000kg and have a successful drone ship landing.
- The query is executed by passing a limit between 4000 and 6000 in payload mass column.

Total Number of Successful and Failure Mission Outcomes

Task 7

List the total number of successful and failure mission outcomes

```
In [15]: %sql SELECT COUNT(*) FROM SPACEXTBL WHERE MISSION_OUTCOME LIKE '%Success%' OR MISSION_OUTCOME LIKE '%Failure%'
 * ibm_db_sa://ynd63707:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/BLUDB
Done.
```

Out[15]:

1
101

- The total number of successful or failed missions are 101.
- The Query uses OR here and set 2 parameters in LIKE which are ‘LIKE %Success% OR LIKE %Failure%’ reason why we can obtain outputs of 2 different categories.

Boosters Carried Maximum Payload

```
In [16]: %sql SELECT BOOSTER_VERSION FROM SPACEXTBL WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTBL)
 * ibm_db_sa://ynd63707:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/BLUDB
Done.
```

Out[16]:

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

- The output of this query is a list of all the booster versions which have carried the maximum capacity.
- This Query consists of a sub-query which calls one output as a result of another.

2015 Launch Records

In [19]:
```
%sql SELECT TO_CHAR(TO_DATE(MONTH("DATE"), 'MM'), 'MONTH') AS MONTH_NAME, \
    LANDING__OUTCOME AS LANDING__OUTCOME, \
    BOOSTER_VERSION AS BOOSTER_VERSION, \
    LAUNCH_SITE AS LAUNCH_SITE \
    FROM SPACEXTBL WHERE LANDING__OUTCOME = 'Failure (drone ship)' AND "DATE" LIKE '%2015%'
```

```
 * ibm_db_sa://ynd63707:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/BLUDB
Done.
```

Out[19]:

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

- There have been 2 failed missions on a drone ship in 2015.
- The Query here calls out multiple columns from data all specifying the same condition where year is 2015.

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

```
In [20]: %sql SELECT "DATE", COUNT(LANDING__OUTCOME) as COUNT FROM SPACEXTBL \
             WHERE "DATE" BETWEEN '2010-06-04' and '2017-03-20' AND LANDING__OUTCOME LIKE '%Success%' \
             GROUP BY "DATE" \
             ORDER BY COUNT(LANDING__OUTCOME) DESC
 * ibm_db_sa://ynd63707:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.a
Done.
```

Out[20]:

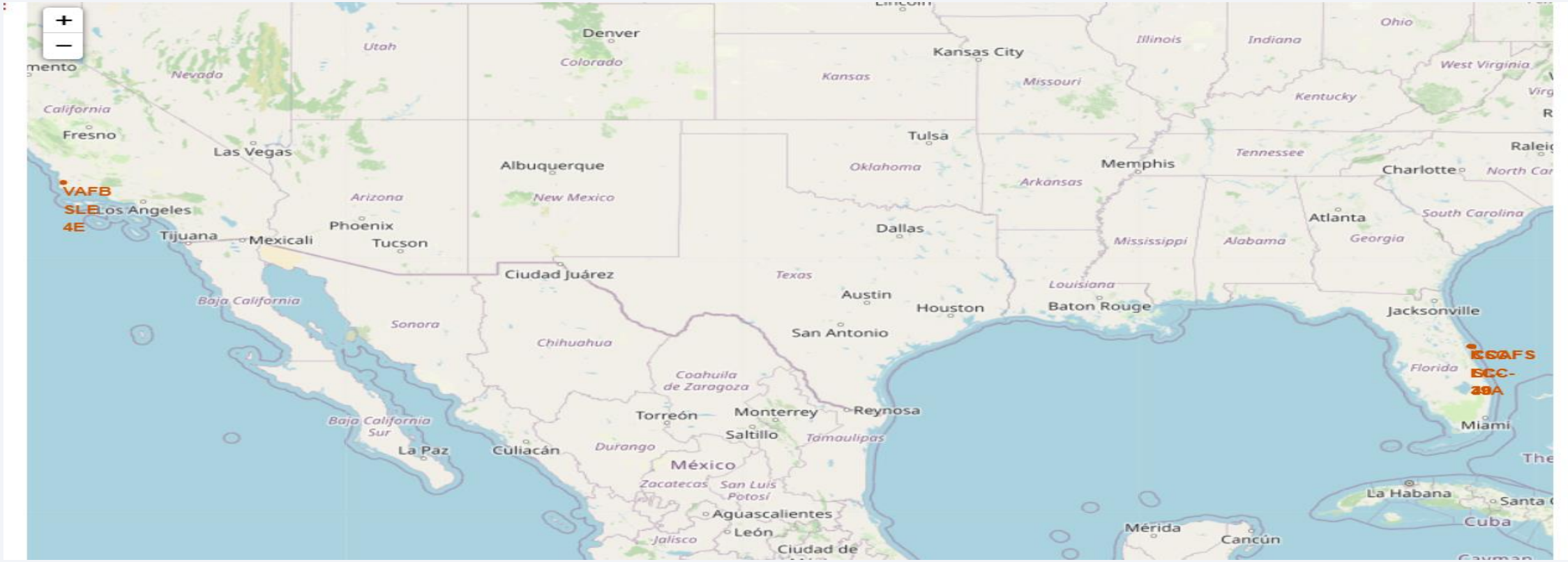
DATE	COUNT
2015-12-22	1
2016-04-08	1
2016-05-06	1
2016-05-27	1
2016-07-18	1
2016-08-14	1
2017-01-14	1
2017-02-19	1

- The list of failed and successful missions between the given date is displayed.
- The main Queries used here are ORDER BY and GROUP BY.
- The list is in descending order as we used DESC query at the end .

Section 3

Launch Sites Proximities Analysis

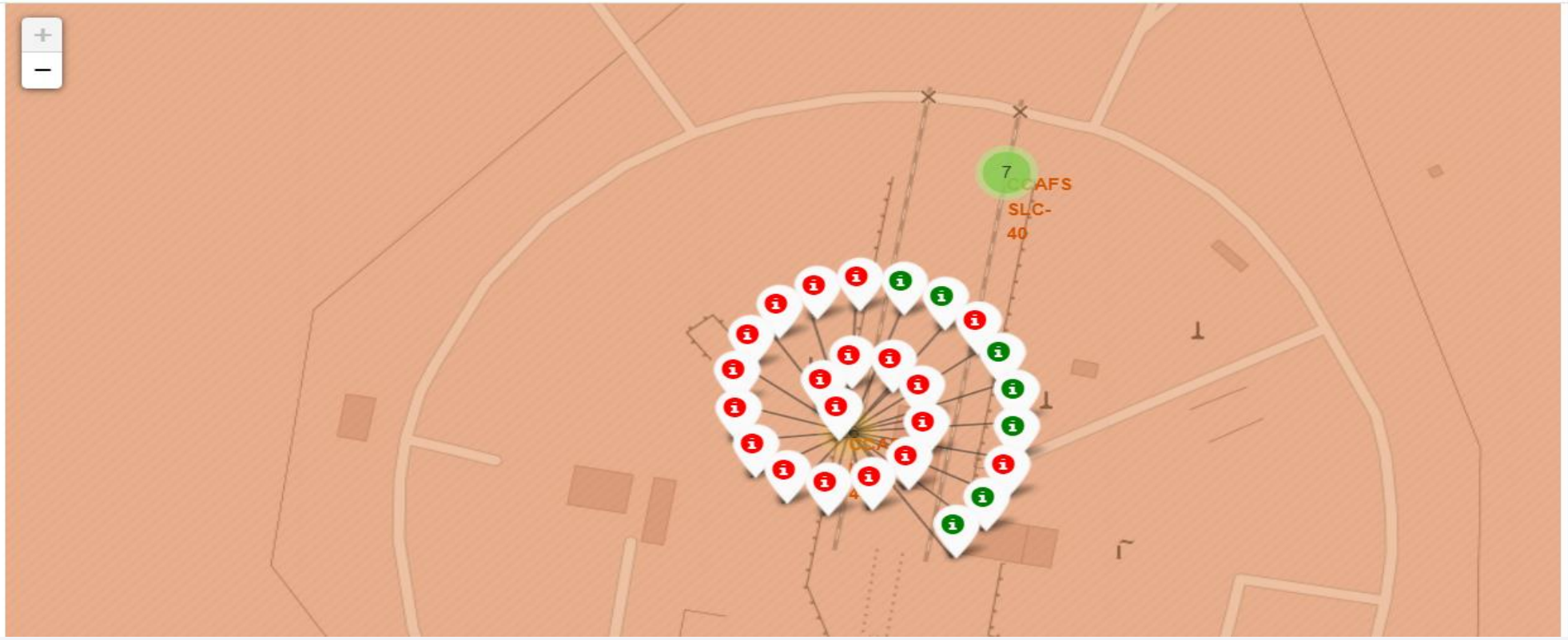
SPACEX Launch Sites Locations Using Folium



- The map shows the different locations of the SPACEX launch sites in USA
- We can see that there are 2 sites in Florida and one near LA. The names are written in red.

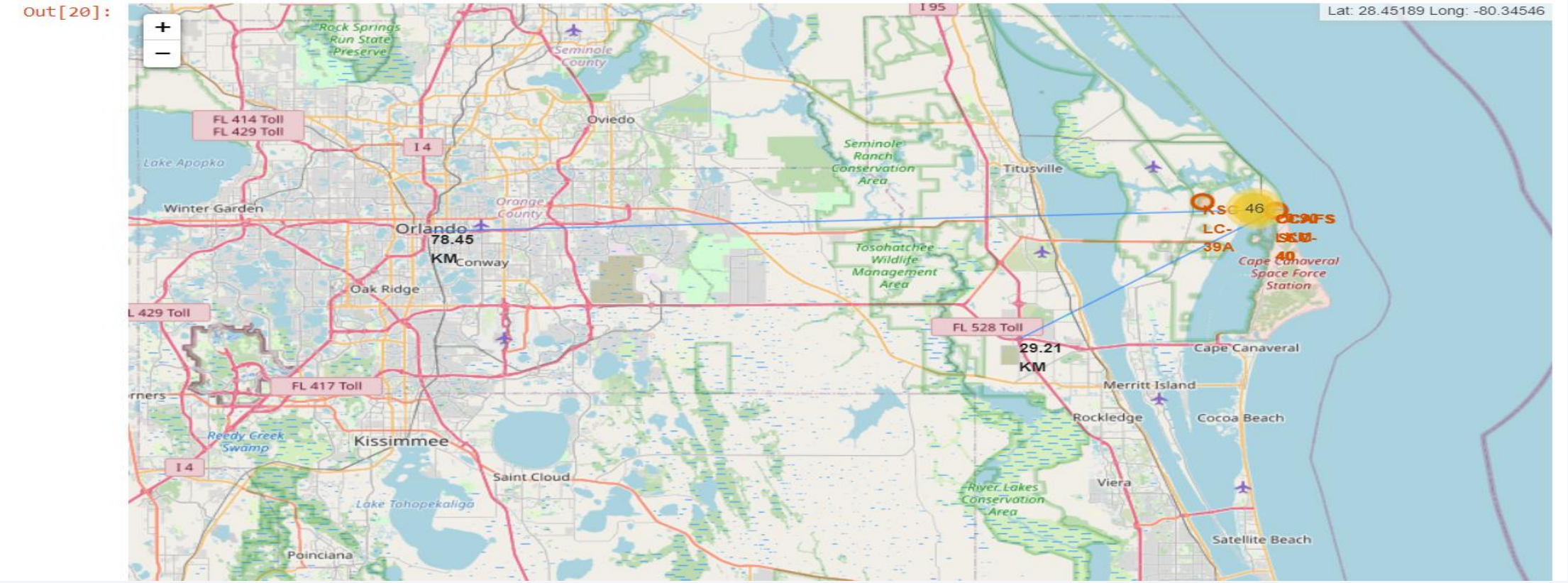
Landing Outcomes On Map Using Folium

Out[15]:

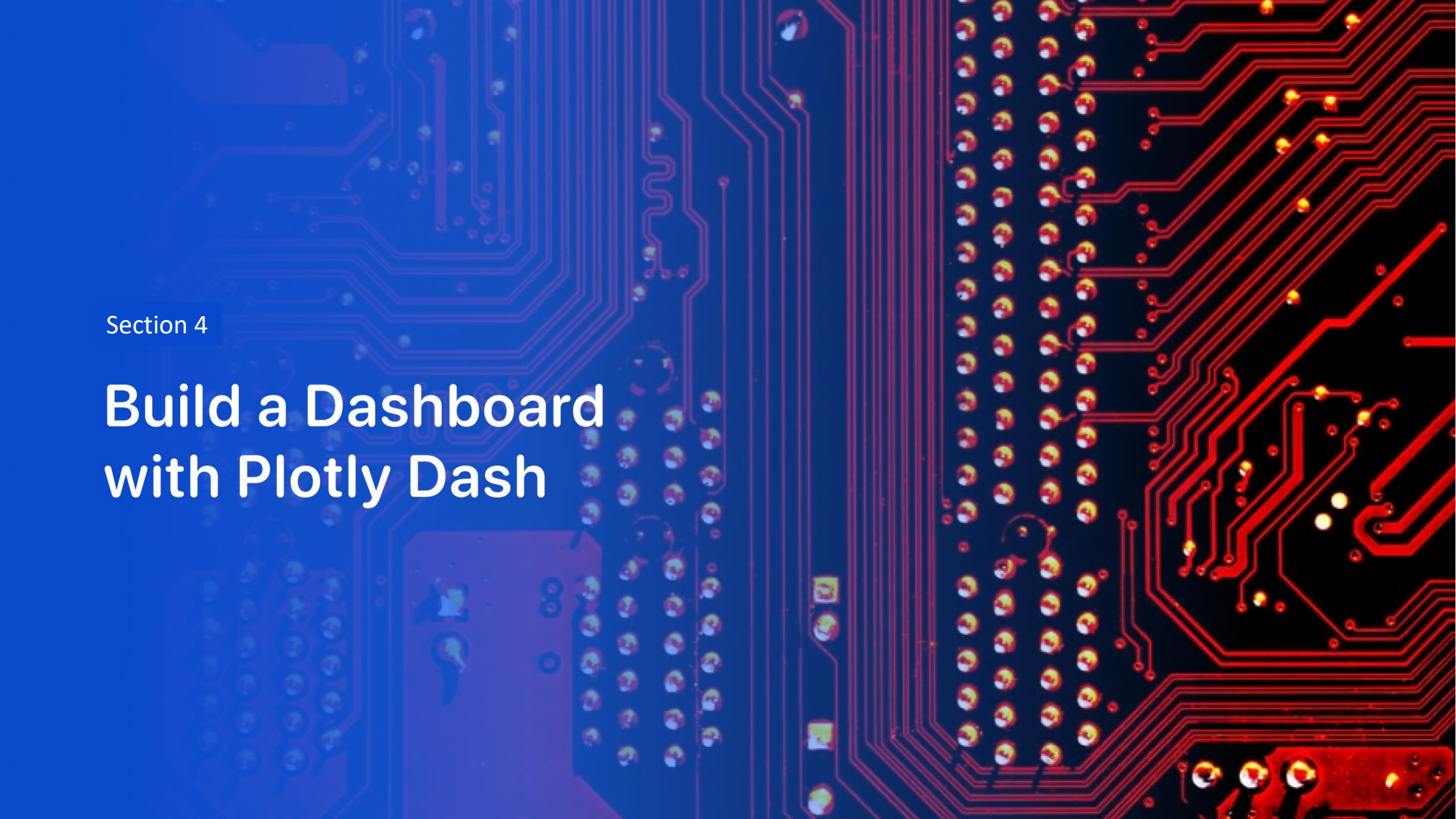


- The map shows the Landing Outcomes from the launch pad CCAF-SLC 40.
- The red markers show the number of unsuccessful landings and green markers show the successful landings.

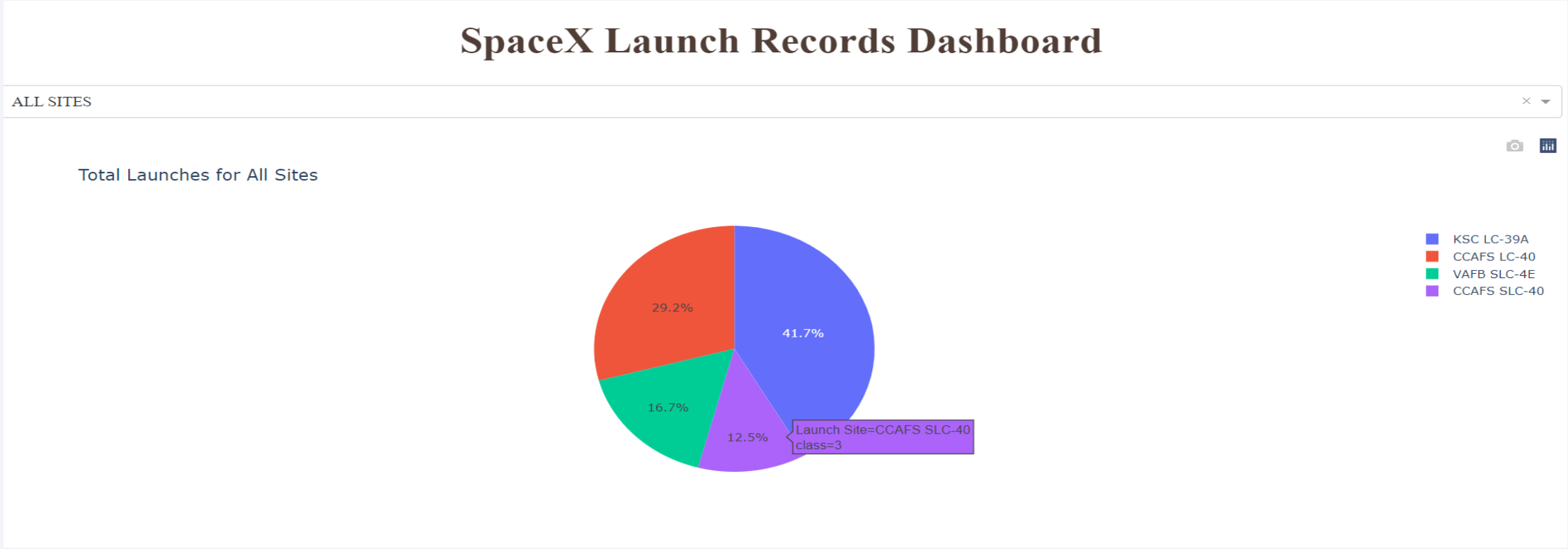
Proximity Distances Using Folium



- The distance from launch site to airport is 78.46 km and distance from launch site to express way is 29.21km. The blue lines on map indicate the distance to each proximity.

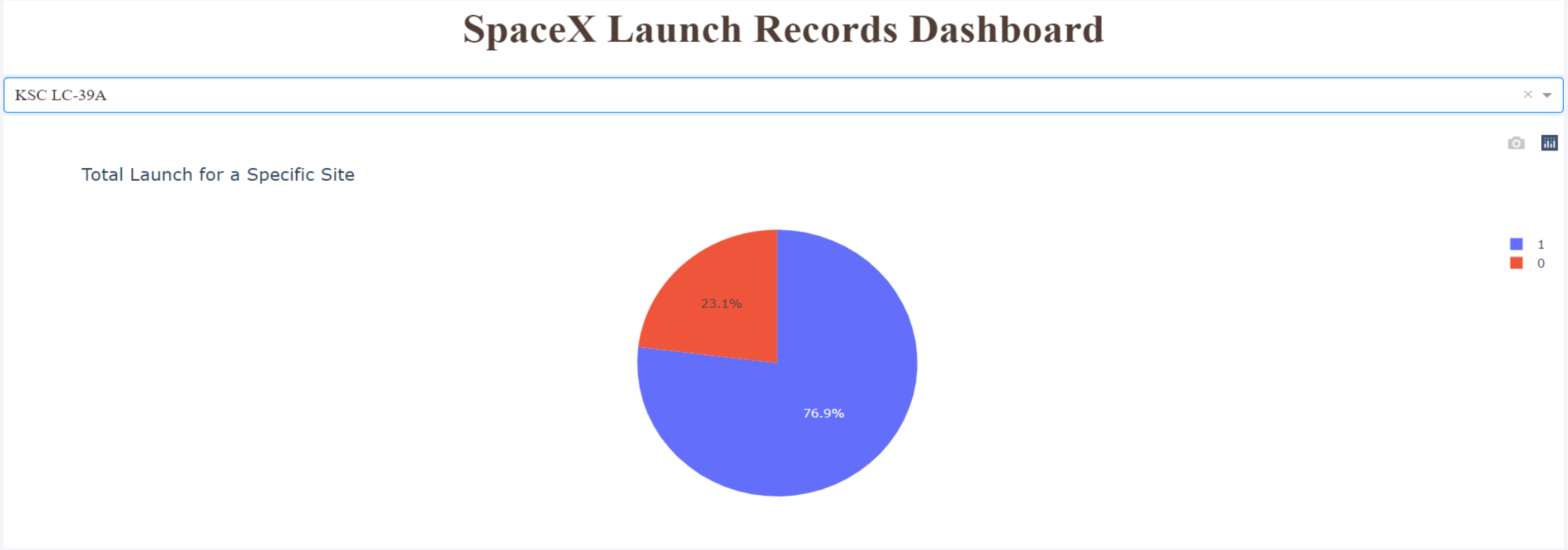

Section 4
Build a Dashboard with Plotly Dash

SPACEX Launch Records – Pie Chart



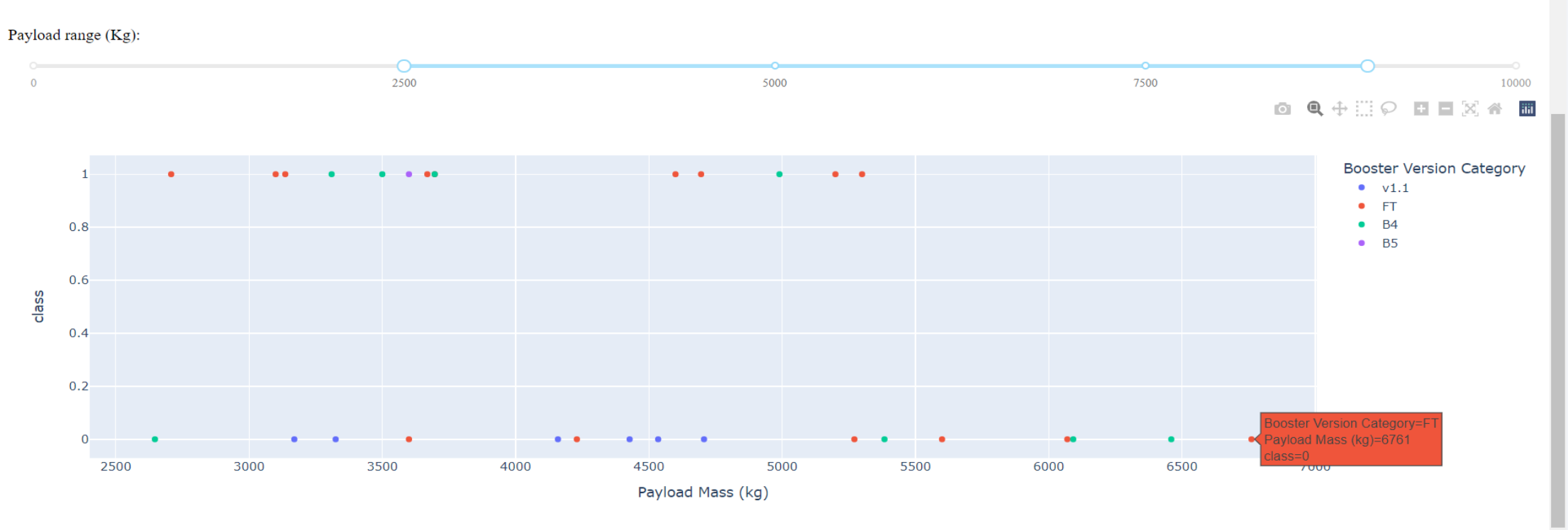
- The pie chart gives us the information of the percentage of launches from each launch site.
- It can be seen that the minimum launches are from CCAF-SLC 40 and maximum from KSC LC 39A launch site.

Maximum Successful Landings Launch Site



- The highest success ratio is of KSC LC-39A launch site. The success ratio of landing to failure here is 76.9% : 23.1%
- The blue color represents class 1 which indicates success and red color indicates class 0 which indicates failure.

Payload Mass vs Outcome Class – Scatter Plot



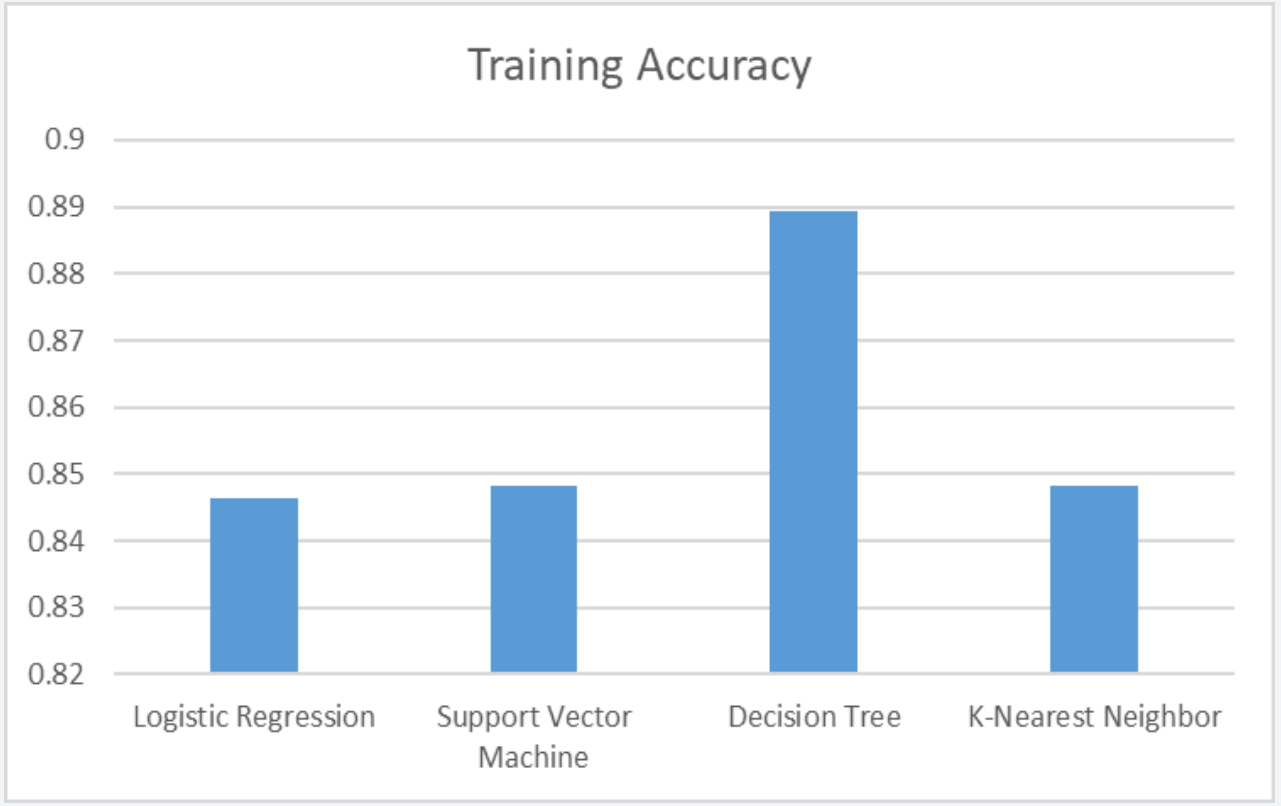
- The payload range selected for this scatter plot is 2500kgs to 9000kgs.
- It is clearly evident that the most successful booster in this range is the FT booster version. It has 8 successful landings and six failed landings.

Section 5

Predictive Analysis (Classification)

Classification Accuracy

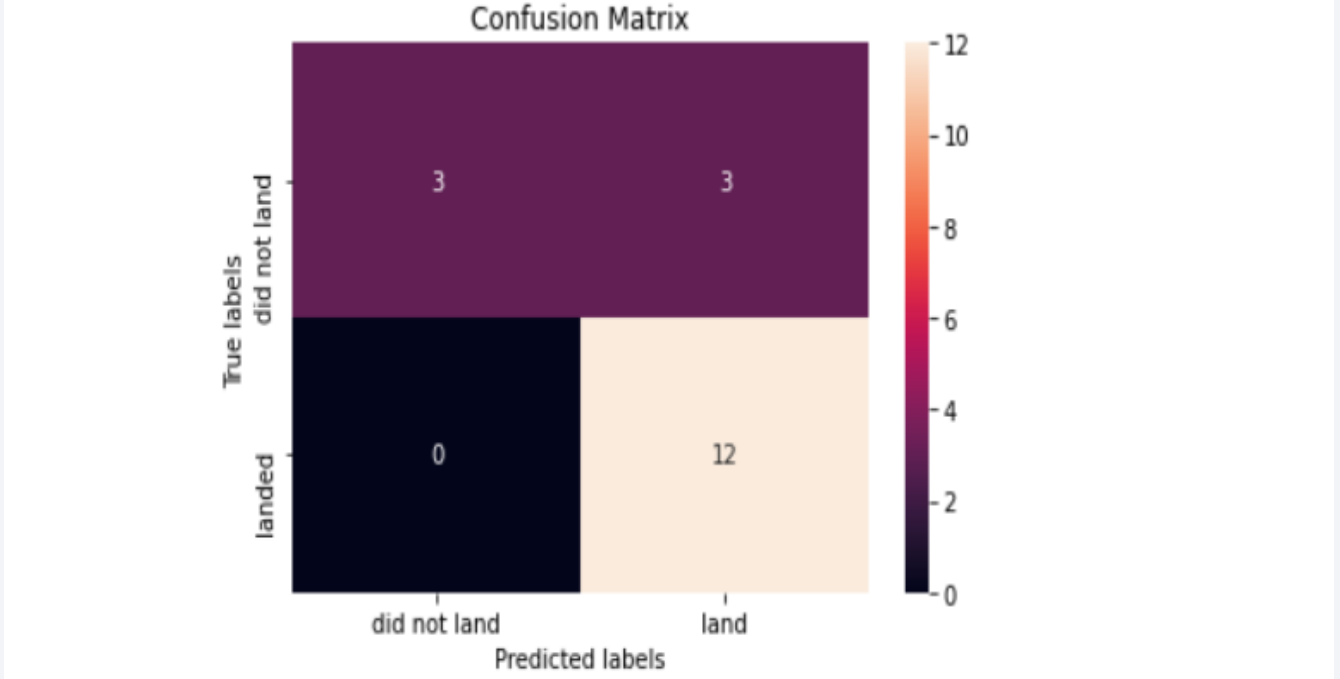
- From the bar chart provided, it is clearly evident that the decision tree classification model has the highest accuracy.
- Though the accuracy differs on decimal level, it can still be considered significant.



Confusion Matrix

- The best performing model based on accuracy was found to be was the decision tree.
- It can be seen that there are 0 false negatives.
- But the model is not 100% accurate, hence it predicts 3 false positives.

```
In [34]: yhat = tree_cv.predict(X_test)
         plot_confusion_matrix(Y_test,yhat)
```



Conclusions

- The data collected can be regarded as reliable as with data wrangling bogus and unnecessary data was cleared out.
- Exploratory Data Analysis was conducted successfully by using Data Visualization, SQL Queries, and creating an interactive plotly dashboard.
- The plotly dashboard created helped us analyze data in many combinations and helped us understand the data better. We could understand various payload mass ranges and most successful booster versions in them.
- Finally classification analysis based on predictions was conducted.
- The most successful model turned out to be Decision Tree Classifier which can be further used to predict the outcome of landing of a new flight based on its factors.

Appendix

- GitHub Links:
- ALL Notebooks are aviable at:
- https://github.com/aqafridi/Data-Science-Specialization/tree/main/Applied%20Data%20Science%20Capstone/5.%20Present%20Data-Driven%20Insights

Thank you!