Winning Space Race with Data Science

Esayas Tesfagiorgis
April 6 , 2023

Outline

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

Executive Summary

- Summary of methodologies
- ✓ The following methodologies were employed during the process of data manipulation. Data collection with API and Web Scrabbing and Wrangling,
- ✓ Exploratory Data Analysis(EDA) used SQL, Data Visualization and Interactive Analytic(IA) using Folium
- ✓ I also employed different ML methods in the prediction of the outcome of the data analysis
- ✓ I have presented screen shots, links and codes for the methodologies employed during the exercise
- Summary of all results
- ✓ The results of the data analysis and were presented on to the built-in IBM cloud service. We can observed the results of EDA, IA and Predictive analysis
- ✓ I have presented screen shots, links and codes for the results

Introduction

- Project Background
- SpaceX is privately funding the development of orbital launch systems that can be reused many times, in a manner similar to the reusability of aircraft. SpaceX has been developing the technologies over several years to facilitate full and rapid reusability of space launch vehicles. The project's long-term objectives include returning a launch vehicle first stage to the launch site in minutes and to return a second stage to the launch pad following orbital realignment with the launch site and atmospheric reentry in up to 24 hours. SpaceX's long term goal is that both stages of their orbital launch vehicle will be designed to allow reuse a few hours after return.
- In 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. In this module, you will be provided with an overview of the problem and the tools you need to complete the course.
- Problems Need Answers

✓ What are the factors that determine a rockets land successful?

✓ how are the rate of success determined taking into consideration various parts and features of the vehicles?

✓ What conditions should be met to ensure a successful and sustainable landing programs for future?

Link for the project description below

❖ SpaceX reusable launch system development program - Wikipedia

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Data pertaining the project were collected from its origins by employing SpaceX API and Web Scrabbing,
- Perform data wrangling
- Data Wrangling using one-hot encoding methods applied to categorical data
- Exploratory data analysis (EDA) using visualization and SQL was performed
- Interactive visual analytics using Folium and Plotly Dash was performed
- Predictive analysis using various classification models was performed
  - Enabled to build, tune, evaluate classification models like KNN, SVM, and Decision Tree

Data Collection

- Various methods were employed to collect the data necessary to perform the required activities.
- The following chart shows the steps taken during the data collection process and actions taken;

Data collection methods(SpaceX API & Data Scrabbing)	Get Request
Decode Get Request content using Json	Data Frame
Data Wrangling –data cleaning	Null values
Web scrabbing	Null values
Data Wrangling	BeautifulSoup
Extract, parse, convert data frame	HTML

Data Collection – SpaceX API

- Get request to sent SpaceX API then Data Collected with basic Data wrangling performed on the data.
- The collected, cleaned and normalized data was formatted to be ready for analysis
- The link to the note book for this exercise is given at : https://github.com/eTesfagiorgis/ibm_watson_march30.git

The codes used for the calls of the SpaceX API looks like this:

```
# Takes the dataset and uses the cores column to call the API and append the data to the lists
def getCoreData(data):
    for core in data['cores']:
            if core['core'] != None:
                response = requests.get("https://api.spacexdata.com/v4/cores/"+core['core']).json()
                Block.append(response['block'])
                ReusedCount.append(response['reuse_count'])
                Serial.append(response['serial'])
            else:
                Block.append(None)
                ReusedCount.append(None)
                Serial.append(None)
            Outcome.append(str(core['landing_success'])+' '+str(core['landing_type']))
            Flights.append(core['flight'])
            GridFins.append(core['gridfins'])
            Reused.append(core['reused'])
            Legs.append(core['legs'])
            LandingPad.append(core['landpad'])
```

Now let's start requesting rocket launch data from SpaceX API with the following URL:

```
spacex_url="https://api.spacexdata.com/v4/launches/past"
```

```
response = requests.get(spacex_url)
```

Data Collection - Scraping

- Web Scrabbing methods was applied on the website of the Falcon 9 records and pages.
- BeautifulSoup lib was used to parse the tables and converted to pandas data frame
- a link GitHub URL of the exercise can be found here:
- https://github.com/eTesfagiorgis/ibm_watson_march30/Web Scraping.ipynb

THE PROCESS OF DATA SCRABING FROM THE FALCON 9 WICKIPEDIA PAGE CAN BE SUMMARIZED AS FOLLOW

Data collection methods(SpaceX API & Data Scrabbing)

Decode Get Request content using Json

Data Wrangling –data cleaning

Web scrabbing

Data Wrangling

Extract, parse, convert data frame

Get Request

Data Frame

Null values

Null values

BeautifulSoup

HTML

Data Wrangling

- EDA was performed to determine the training labels of the data frame resulted from the Web Scrabbing methods discussed above
- Count of the number of launches at each site vs the number of occurrences at each orbit was calculated
- As a result, landing outcome labels from the outcome columns of the data frame was created and exported as a csv file for future use.
- The ling to the GitHub URL of the author is given here: https://github.com/eTesfagiorgis/ibm_watson_march30.git

/ GitHub_trial / EDA

```
ReusedCount         int64
Serial             object
Longitude         float64
Latitude          float64
dtype: object
```

TASK 1: Calculate the number of launches on each site

The data contains several Space X launch facilities: Cape Canaveral Space Launch Complex 40 **VAFB SLC 4E** , Vandenberg Air Force Base Space Launch Complex 4E **(SLC-4E)**, Kennedy Space Center Launch Complex 39A **KSC LC 39A** .The location of each Launch Is placed in the column `LaunchSite`

Next, let's see the number of launches for each site.

Use the method `value_counts()` on the column `LaunchSite` to determine the number of launches on each site:

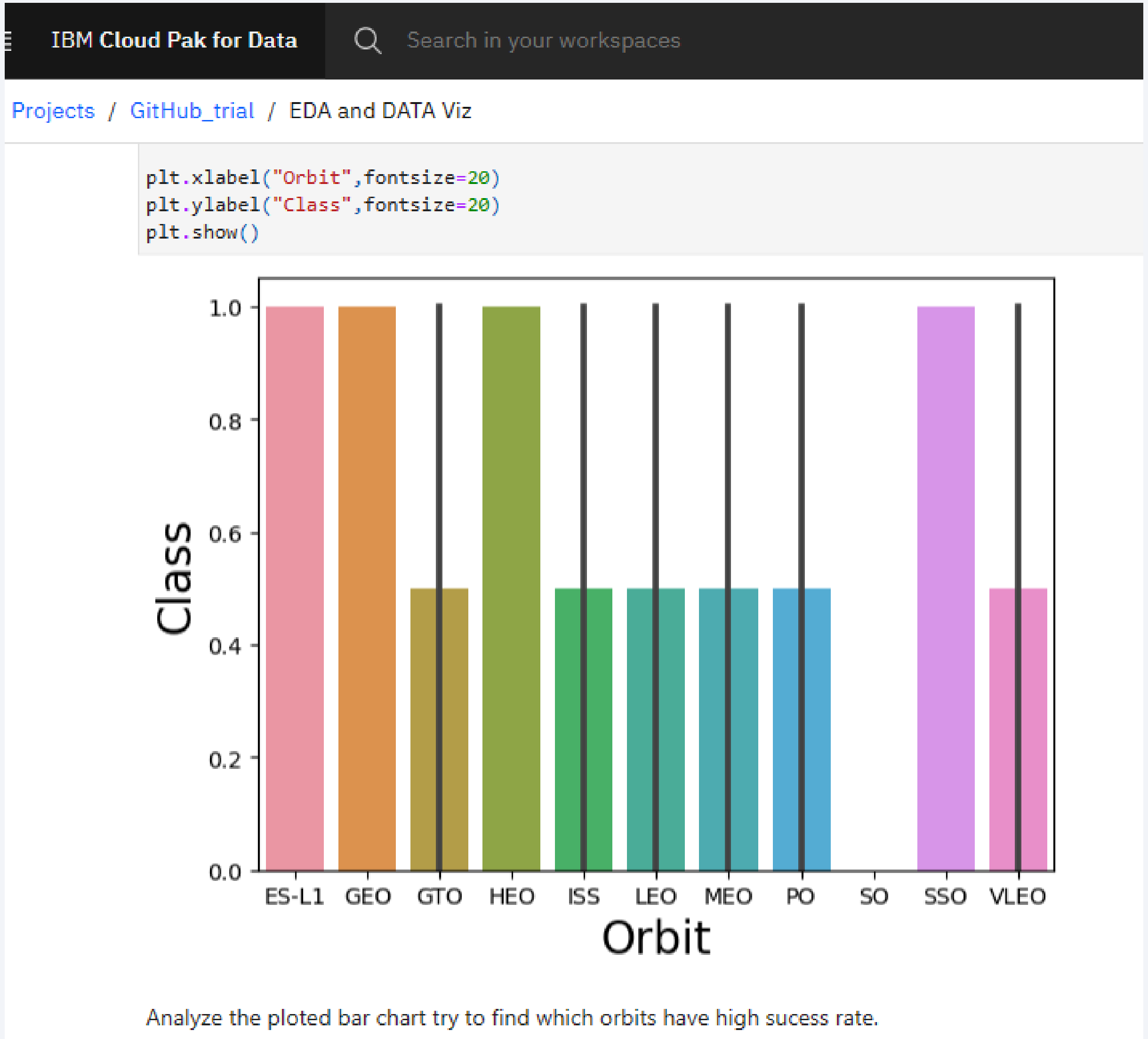
```
5]: # Apply value_counts() on column LaunchSite
    df.LaunchSite.value_counts()
```

```
5]: CCAFS SLC 40    55
    KSC LC 39A      22
    VAFB SLC 4E     13
    Name: LaunchSite, dtype: int64
```

Each launch aims to an dedicated orbit, and here are some common orbit types:

EDA with Data Visualization

- As part of the EDA, Visualization is performed to see the relationships of flight number vs launches site, Payload vs launch site,
- Success rate at each orbit type, flight number, and the launch success yearly trend is shown
- A link of the GitHub URL of the EDA with data visualization notebook is given here: https://github.com/eTesfagiorgis/ibm_watson_march30.git



EDA with SQL

- Using bullet point format, summarize the SQL queries you performed
- Add the GitHub URL of your completed EDA with SQL notebook, as an external reference and peer-review purpose

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
- Add the GitHub URL of your completed interactive map with Folium map, as an external reference and peer-review purpose

Build a Dashboard with Plotly Dash

- Summarize what plots/graphs and interactions you have added to a dashboard
- Explain why you added those plots and interactions
- Add the GitHub URL of your completed Plotly Dash lab, as an external reference and peer-review purpose

Predictive Analysis (Classification)

- Summarize how you built, evaluated, improved, and found the best performing classification model
- You need present your model development process using key phrases and flowchart
- Add the GitHub URL of your completed predictive analysis lab, as an external reference and peer-review purpose

Results

- The following slides shows results that are drawn from the analysis performed above:
    - Exploratory data analysis results
    - Interactive analytics demo in screenshots
    - Predictive analysis results

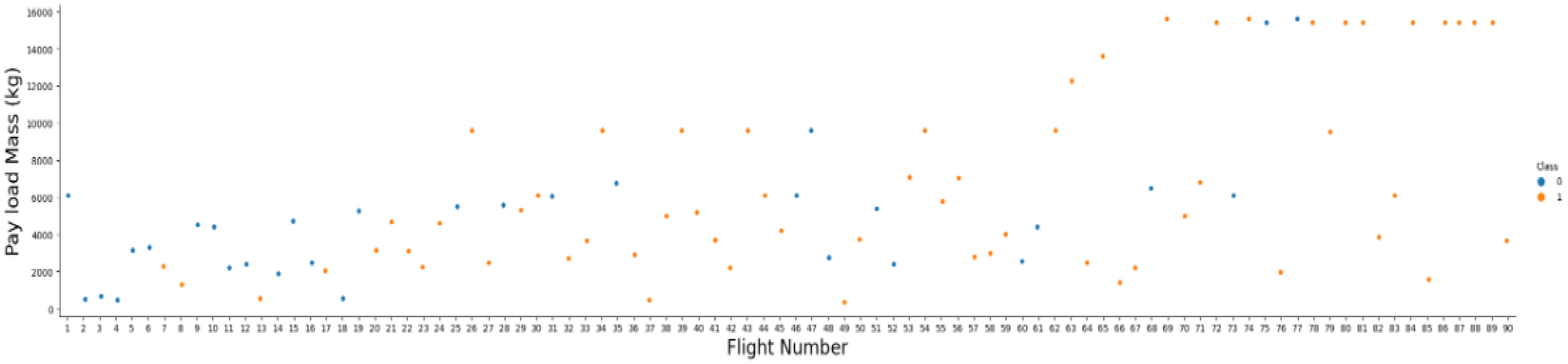
Section 2

Insights drawn from EDA

Flight Number vs. Launch Site

- As it can be deduced from the plots below(screenshot), the larger the flight amount at a launch site, the greater the success rate at a launch site.

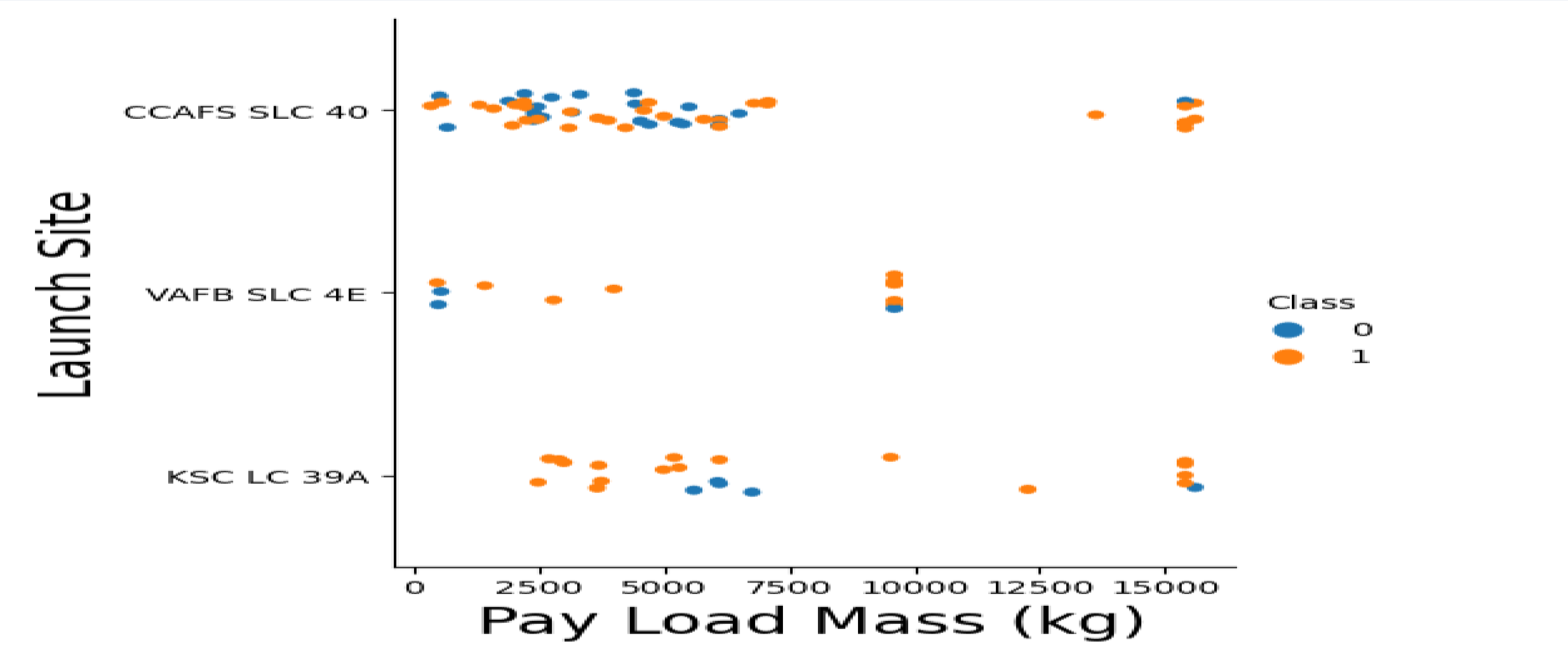
```
sns.catplot(y="PayloadMass", x="FlightNumber", hue="Class", data=df, aspect = 5)
plt.xlabel("Flight Number",fontsize=20)
plt.ylabel("Pay load Mass (kg)",fontsize=20)
plt.show()
```



Payload vs. Launch Site

- The picture below taken from a screen shot of a payload against launch site, shows the greater the payload mass for the launch site CCAFS SLC 40, the higher the success rate for the rocket

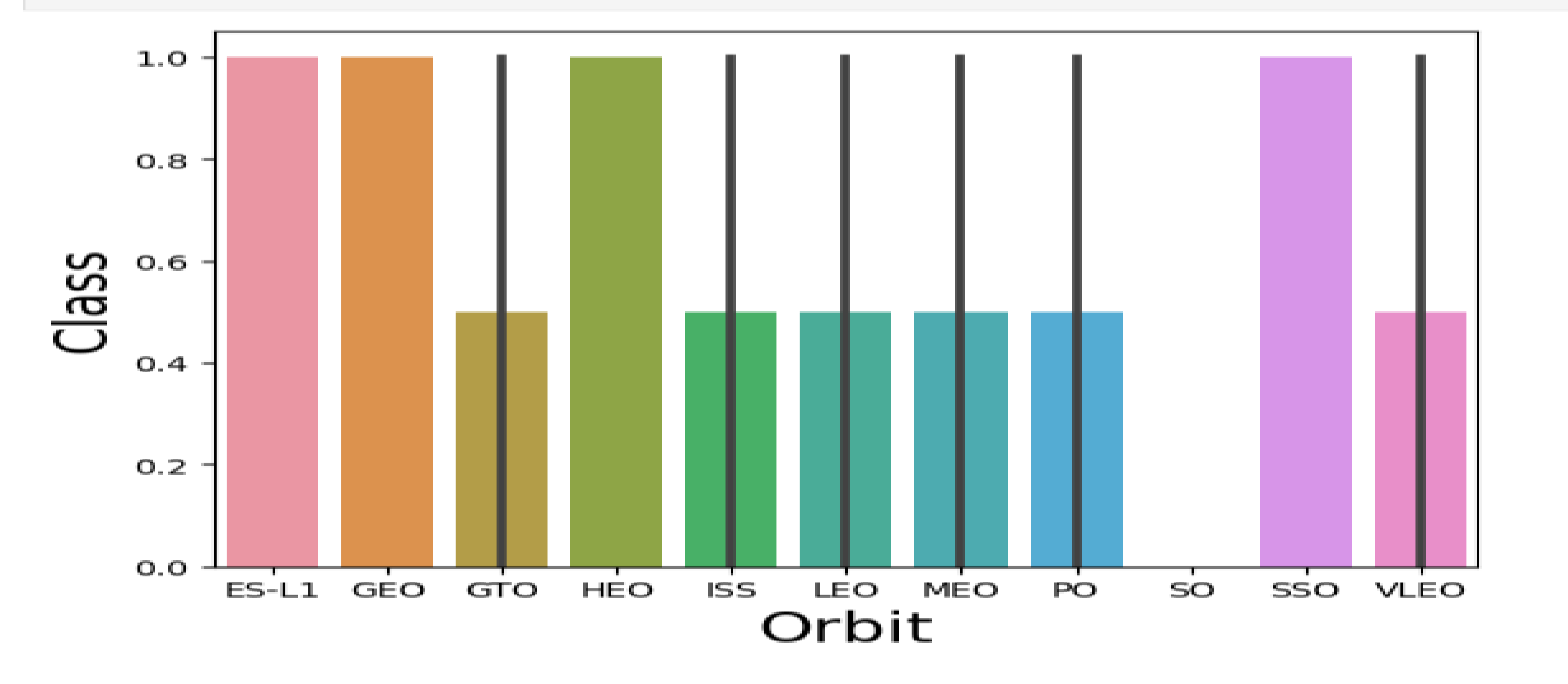
Payload vs. Launch Site



Success Rate vs. Orbit Type

- The histogram(bar chart) given below shows that ESL-L1, GEO, HEO, SSO, had much better success rate compared to the rest of the orbit types:

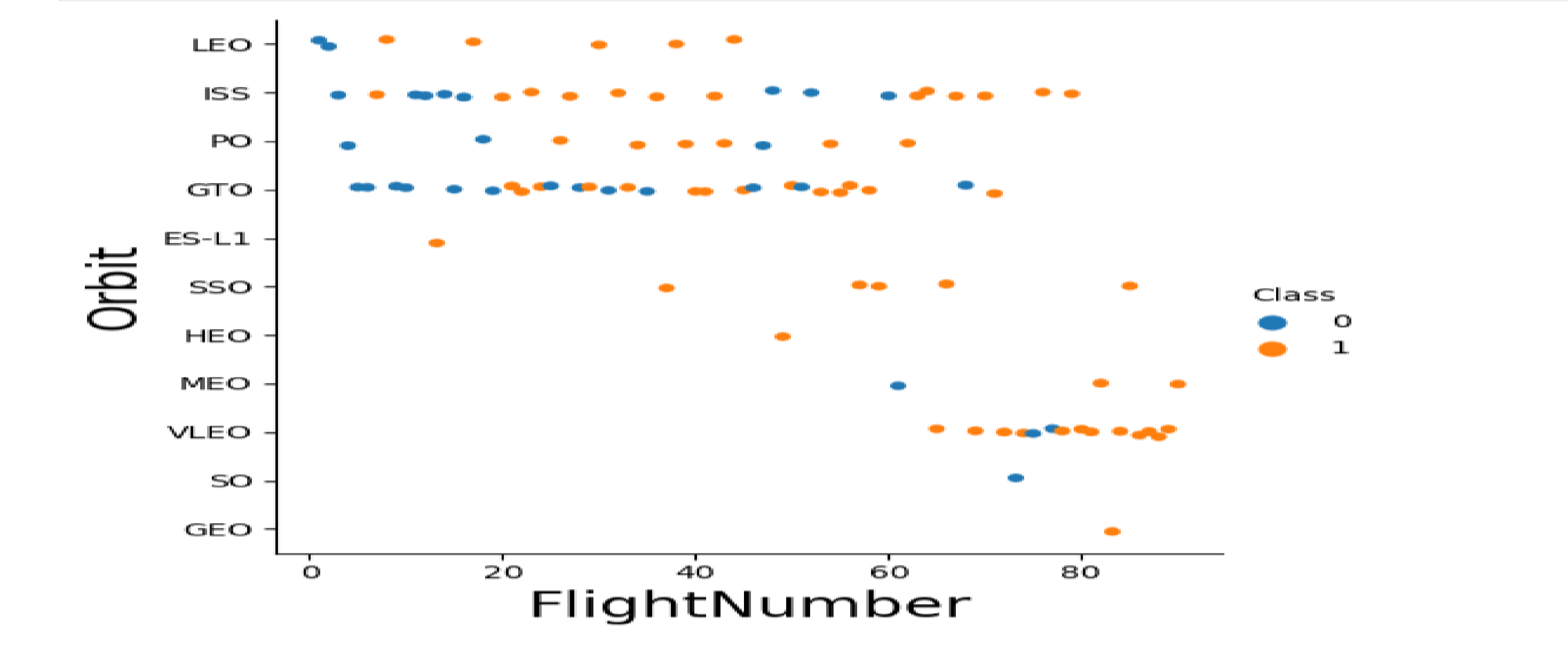
success rate of each orbit type



Flight Number vs. Orbit Type

- From the scatter plot drawn Flight Number versus Orbit type, we can grasp, some flights are successes related to the orbit type for eg. LEO has a relatively good relationship with some of the flights.
- Flights success rate in the GTO orbit show a very little or no relationship to the success of flights and orbits

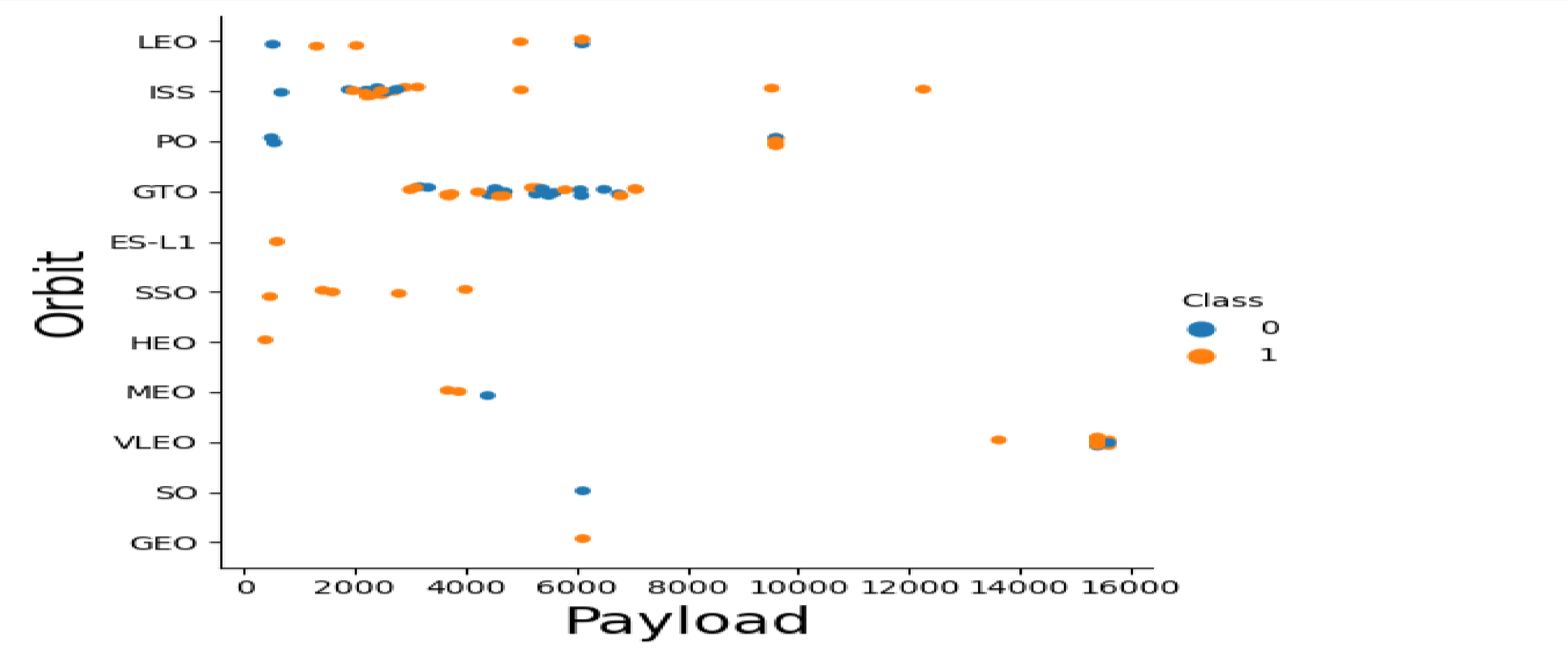
Flight number vs. Orbit type



Payload vs. Orbit Type

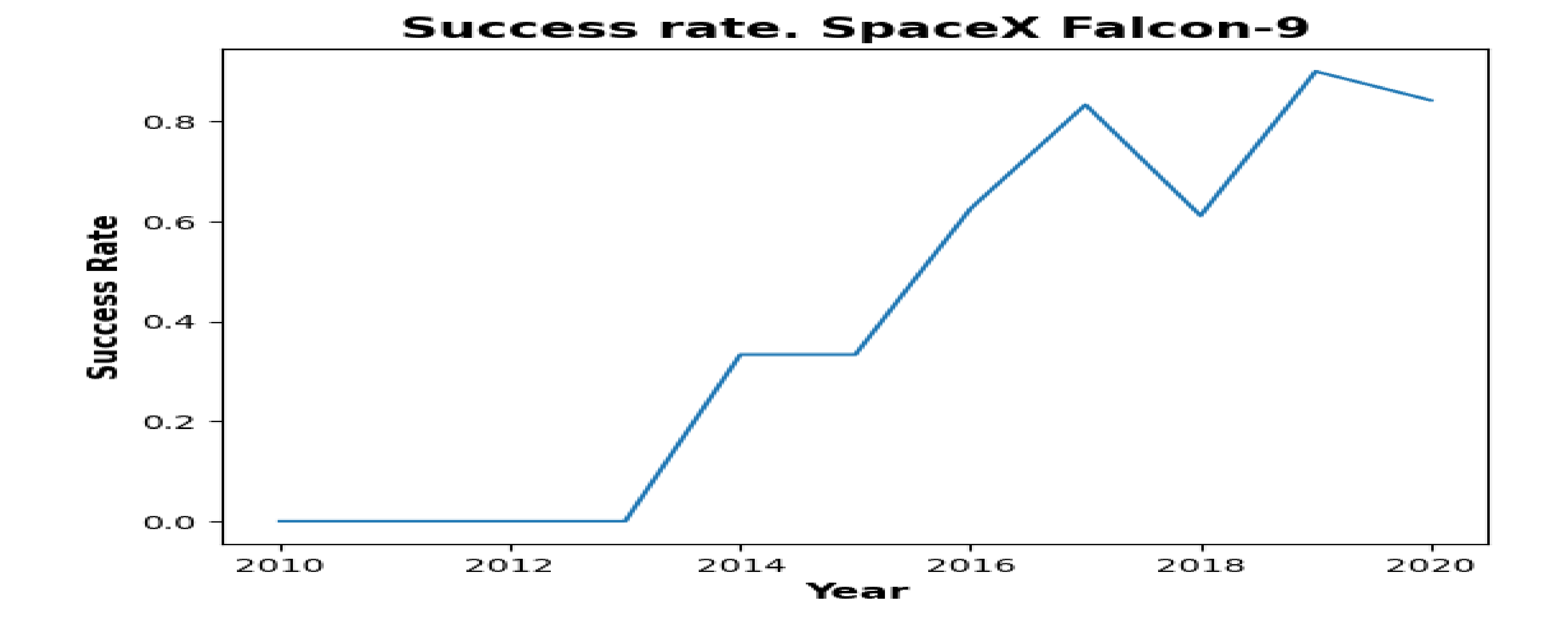
- From the scatter plot drawn PayloadMass versus Orbit type, we can grasp, some flights are successes related to the orbit type for eg. LEO has a relatively good relationship with some of the flights.
- Flights success rate in the GTO orbit show a very little or no relationship to the success of flights and orbits

payload vs. orbit type



Launch Success Yearly Trend

- A line chart of yearly average success rate since 2010 shows a flat trend up to 2013, however, we can see an average increment from 2013 up to 2017, after a small decline in the year 2018, it shows an increase the next year. In 2020, the launce success show a slight decline.
- On average there is an increasing trend of launch success that may be attributed to the increase in the safety and regulation at a launch site by the company



All Launch Site Names

- The code that used to display the unique launch sites in the SpaceX mission is given by:

	DD-MM-YYYY	TIME__UTC_	BOOSTER_VERSION	LAUNCH_SITE	PAYLOAD	PAYLOAD_MASS__KG_	ORBIT	CUSTOMER	MISSION_OUTCOME	LANDING__OUTCOME
0	2010-04-06	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit ...	0	LEO	SpaceX	Success	Failure (parachute)
1	2010-08-12	15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight C1, two CubeSats, barrel of...	0	LEO (ISS)	NASA (COTS) NRO	Success	Failure (parachute)
2	2012-08-10	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1 ...	500	LEO (ISS)	NASA (CRS)	Success	No attempt
3	2013-01-03	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2 ...	677	LEO (ISS)	NASA (CRS)	Success	No attempt
4	2013-03-12	22:41:00	F9 v1.1	CCAFS LC-40	SES-8 ...	3170	GTO	SES	Success	No attempt

Task 2

Display 5 records where launch sites begin with the string 'CCA'

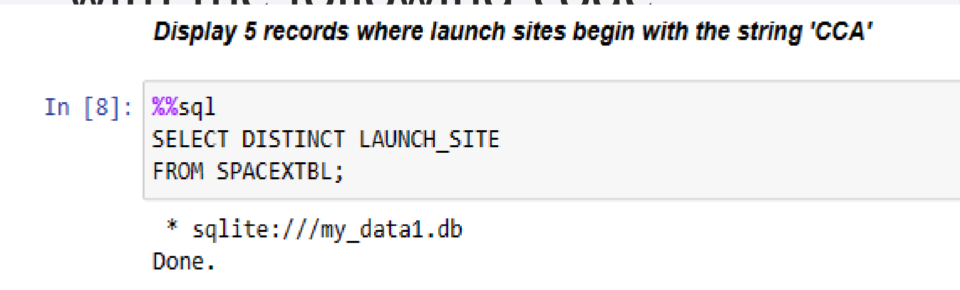
```
SpaceX['LAUNCH_SITE'].unique()
```

```
array(['CCAFS LC-40', 'KSC LC-39A', 'VAFB SLC-4E', 'CCAFS SLC-40'],
      dtype=object)
```

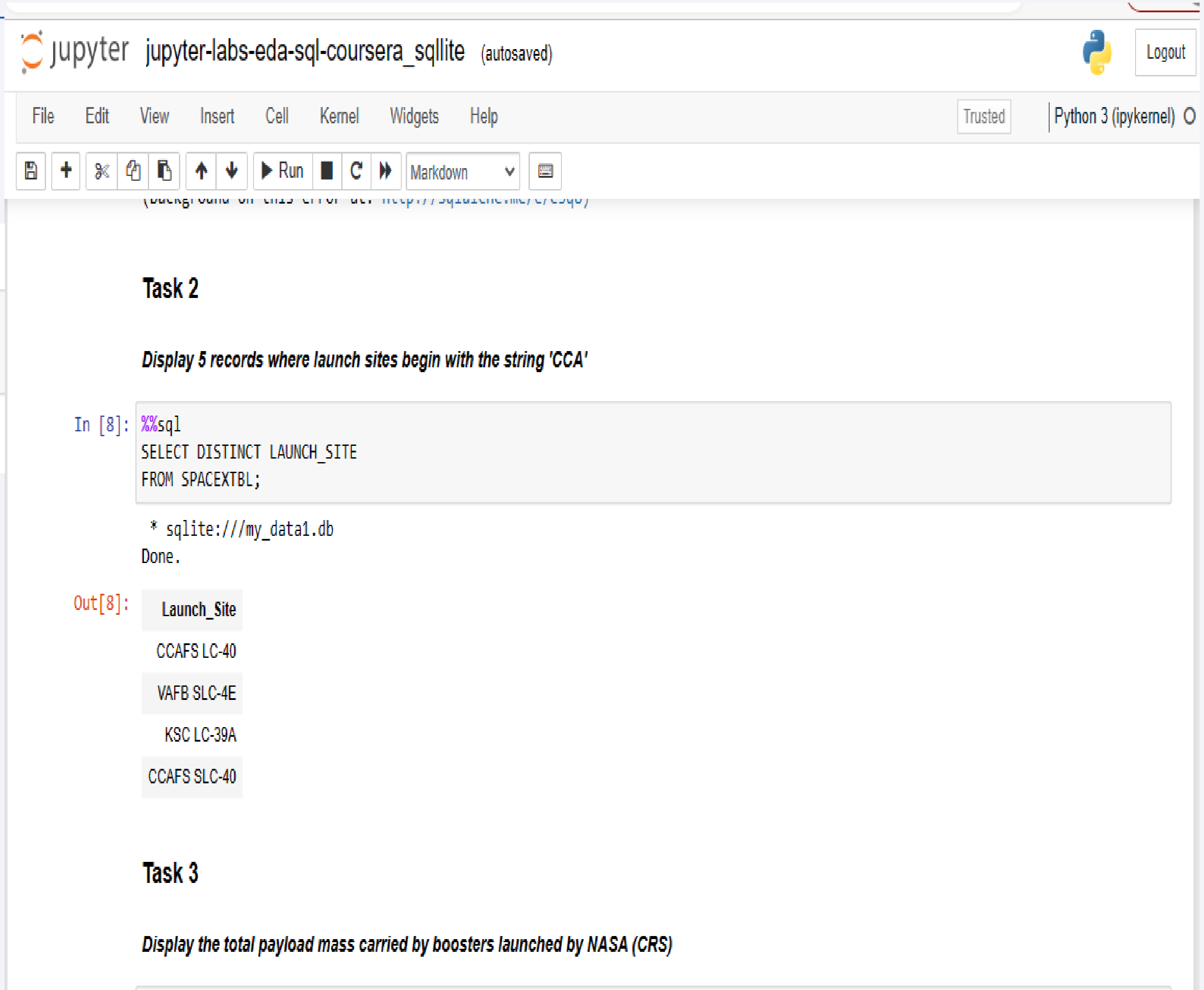
- As we can see in the screen shot of the results of the above code, there are 4 unique launch sites used by the SpaceX. They are namely, KSC LC-39A, CCAFS LC-40, CCAFS SLC-40, and VAFB SLC-4E

Launch Site Names Begin with 'CCA'

- The 5 records where launch sites begin with `CCA` can be obtained with the following code:



- After the code is used to display the 5 launch sites that begin with 'CCA', we used to query the following table with the first 5 indices[0] to [4]



Total Payload Mass

- The total payload mass that is carried by boosters launched by NASA, can be calculated with the following line of code:

Task 3

Display the total payload mass carried by boosters launched by NASA (CRS)

```
In [9]: %%sql
SELECT SUM(PAYLOAD_MASS__KG_)
FROM SPACEXTBL
WHERE Customer = 'NASA (CRS)';

 * sqlite:///my_data1.db
Done.

Out[9]:
```

SUM(PAYLOAD_MASS__KG_)
45596

- As it can be shown in the result above, the total payloadmass carried by the boosters launched by NASA is 45596kgs

Average Payload Mass by F9 v1.1

- The average payload mass that is carried by boosters launched Falcon 9 Verson 1.1, can be calculated with the following line of code:

```
In [17]: %%sql
SELECT AVG(PAYLOAD_MASS__KG_)
FROM SPACEXTBL
WHERE Booster_Version LIKE 'F9 v1.1%';

 * sqlite:///my_data1.db
Done.

Out[17]:
AVG(PAYLOAD_MASS__KG_)
2534.6666666666665
```

The average payload mass that is carried by boosters Falcon 9 version 1.1, is calculated to be 2534.7kgs/

Total Number of Successful and Failure Mission Outcomes

- The total number of successful and failure mission outcomes can be calculated using the following line of code:

- The outcome of the operation shows that there were 100 successful total number of outcomes, whereas, there was only 1 unsuccessful(faillure) mission outcomes

List the total number of successful and failure mission outcomes

In [13]:

```
%%sql
SELECT MISSION_OUTCOME, COUNT(MISSION_OUTCOME) AS TOTAL_NUMBER
FROM SPACEXTBL
GROUP BY MISSION_OUTCOME;
```

```
 * sqlite:///my_data1.db
Done.
```

Out[13]:

Mission_Outcome	TOTAL_NUMBER
Failure (in flight)	1
Success	98
Success	1
Success (payload status unclear)	1

Boosters Carried Maximum Payload

- The following line of code can be used to determine the Boosters that carried the maximum payload in the mission.

- A total of 11 bosster versions bear a maximum load of 15600kbs/lbs/tons. They are namely:

2015 Launch Records

- The following line of code can be used to list the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015

- As we can see from the results(outcomes) of the above code, there are two instances in 2015 where the landing outcomes, the booster versions and launch sites were all determined to be failure,

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

- The following line of code can be used to 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:

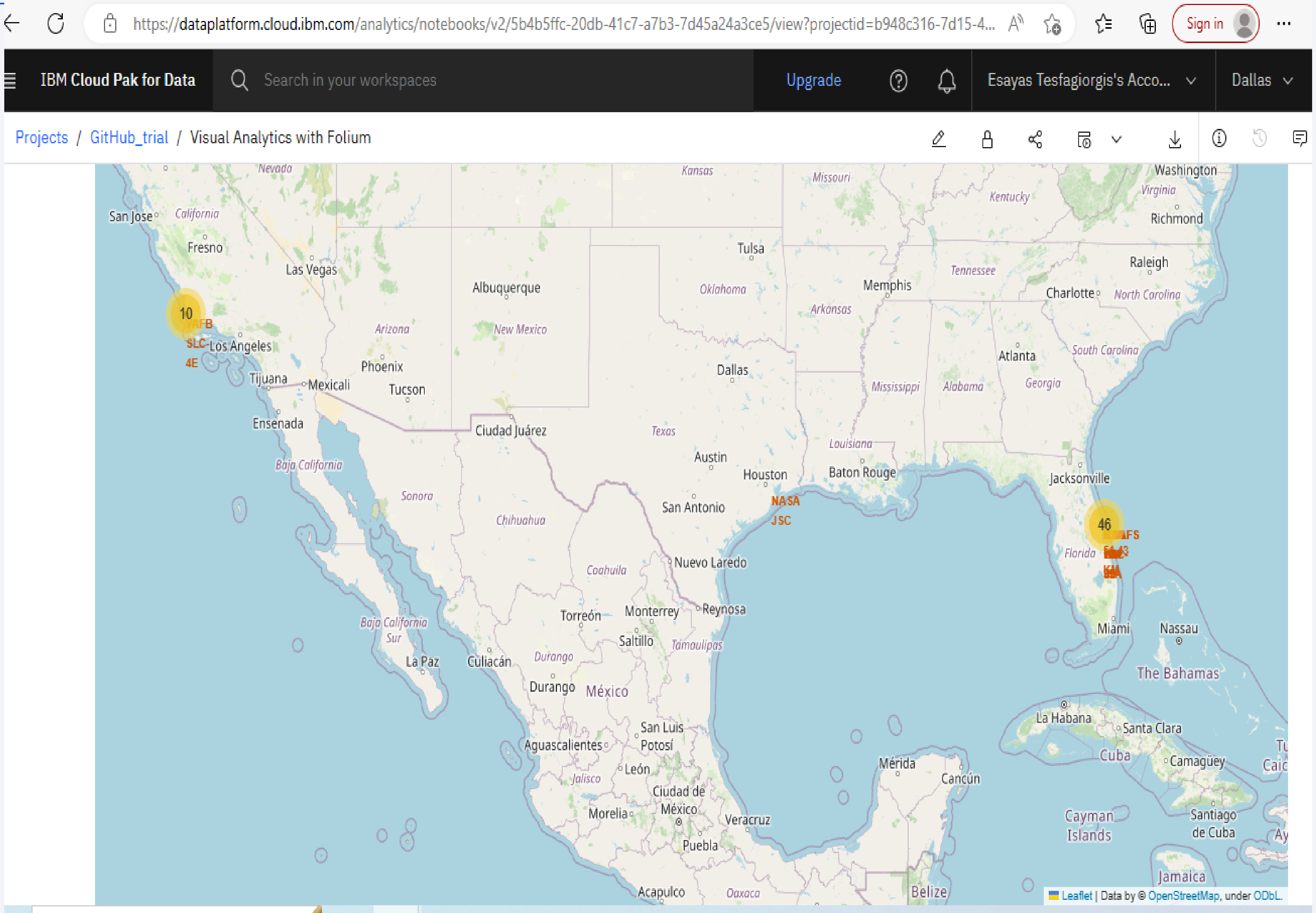
- As a result, between the given dates, there were total 32 times landing outcomes.

Section 3

Launch Sites Proximities Analysis

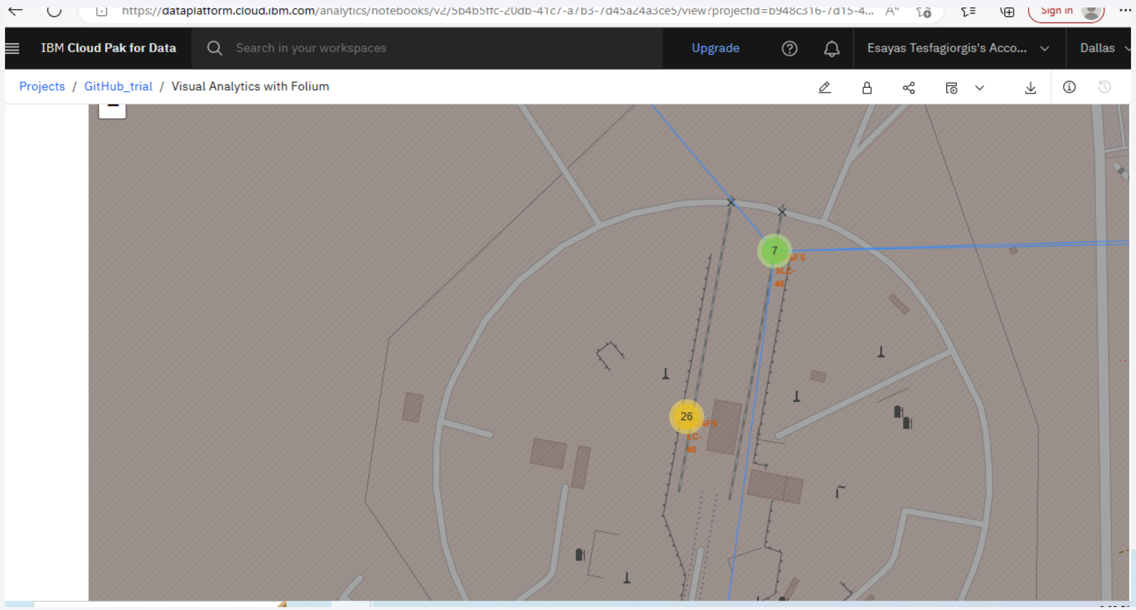
<Folium Map Screenshot 1>

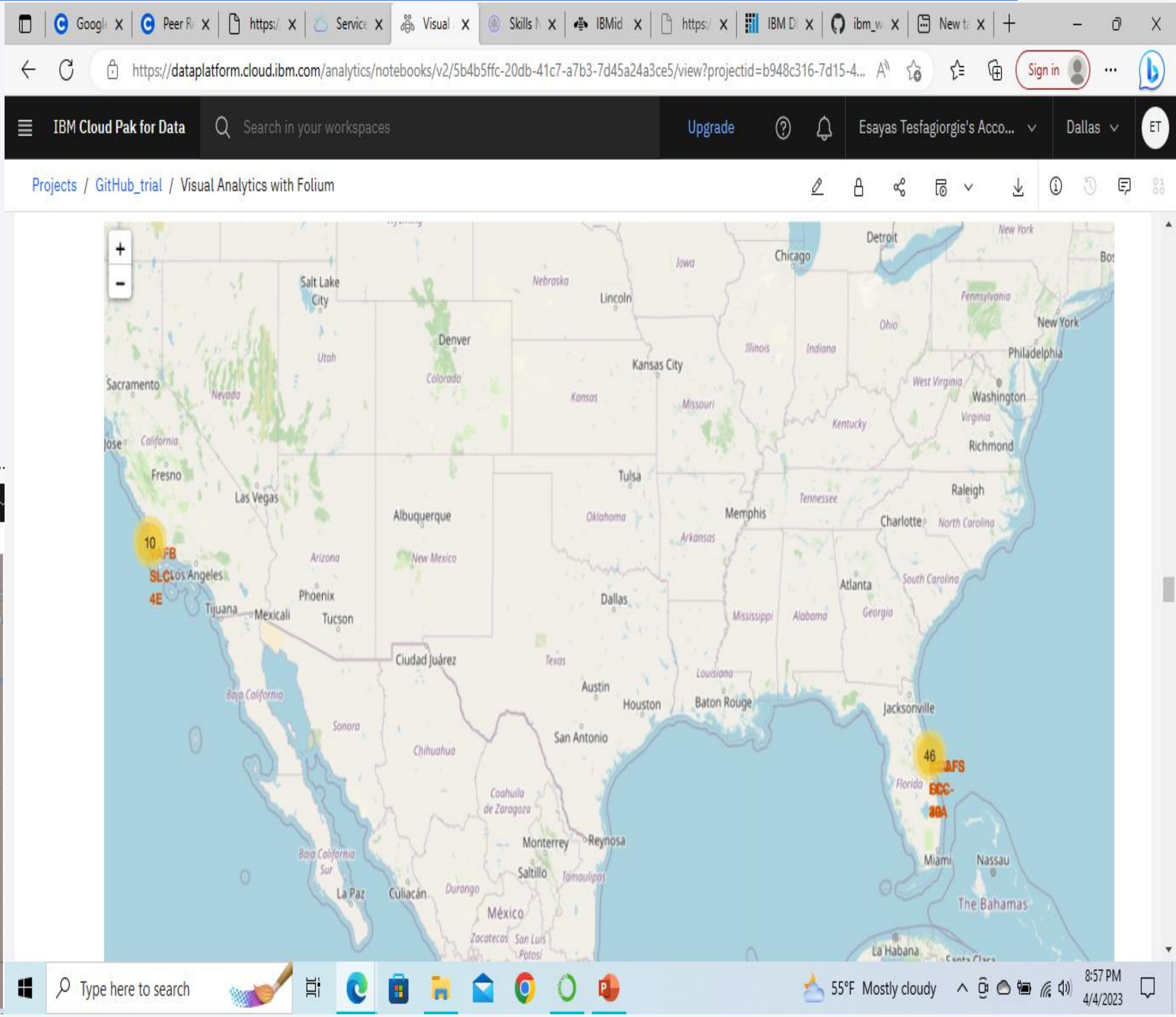
- Replace <Folium map screenshot 1> title with an appropriate title
- Explore the generated folium map and make a proper screenshot to include all launch sites' location markers on a global map
- Explain the important elements and findings on the screenshot



<Folium Map Screenshot 2>

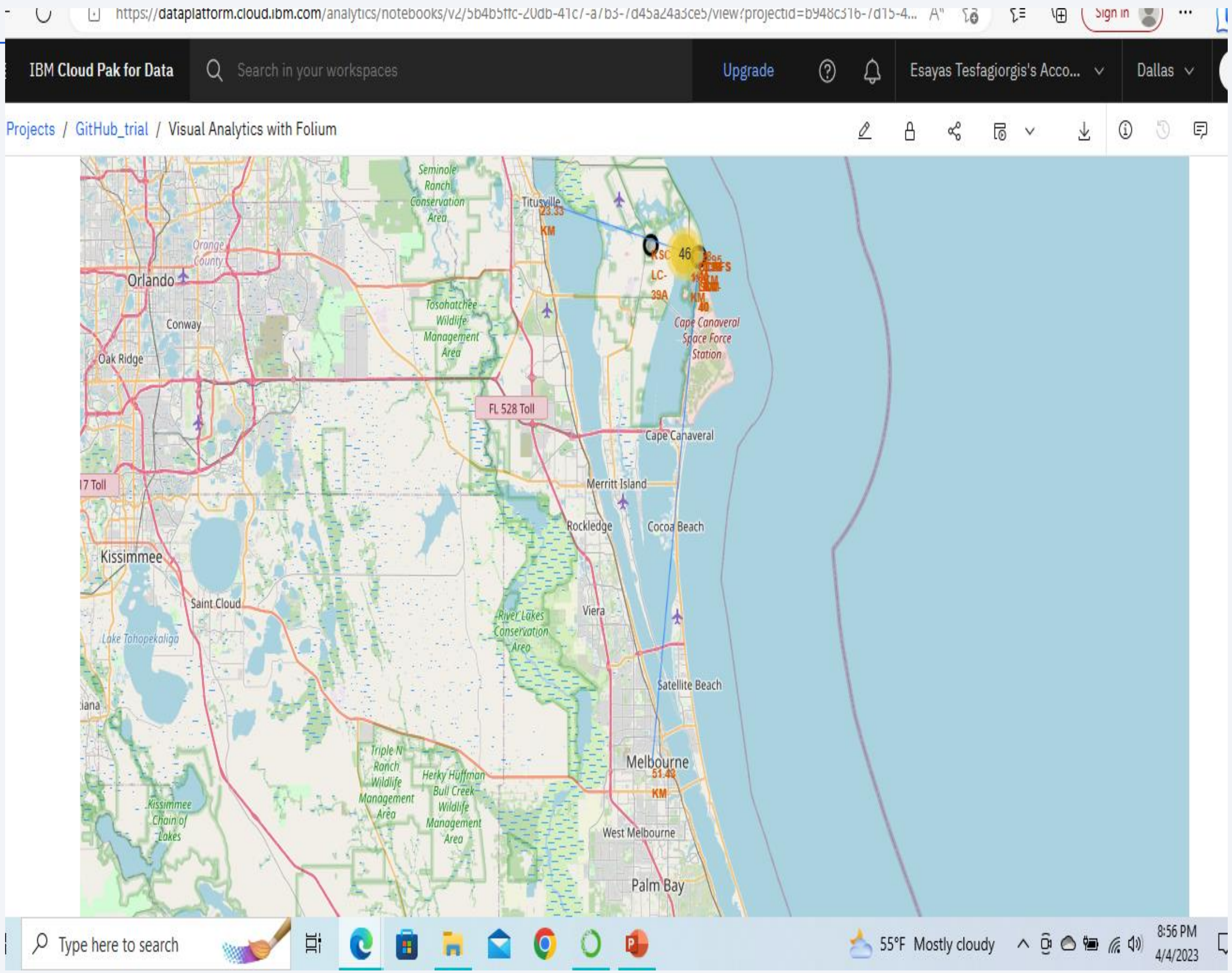
- Replace <Folium map screenshot 2> title with an appropriate title
- Explore the folium map and make a proper screenshot to show the color-labeled launch outcomes on the map
- Explain the important elements and findings on the screenshot





Launch site Distance from Proximites

- Replace <Folium map screenshot 3> title with an appropriate title
- Explore the generated folium map and show the screenshot of a selected launch site to its proximities such as railway, highway, coastline, with distance calculated and displayed
- Explain the important elements and findings on the screenshot

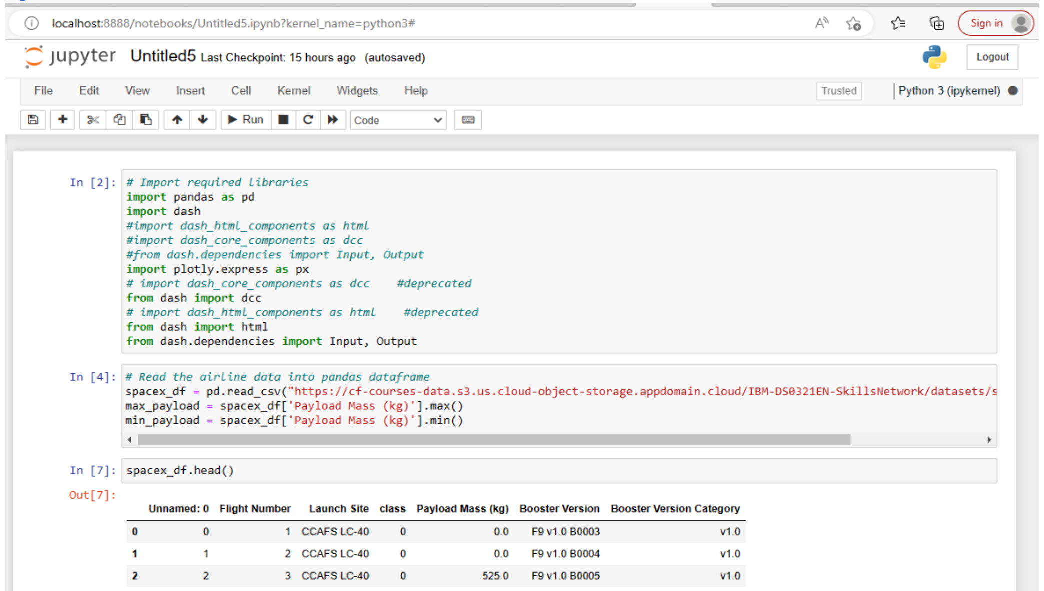


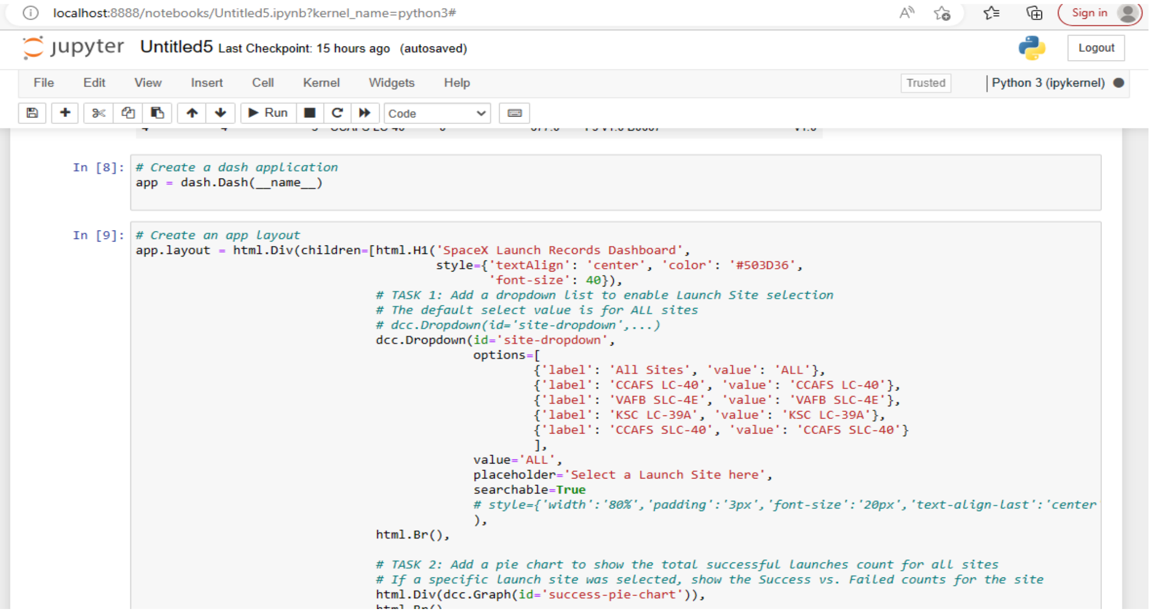
Section 4

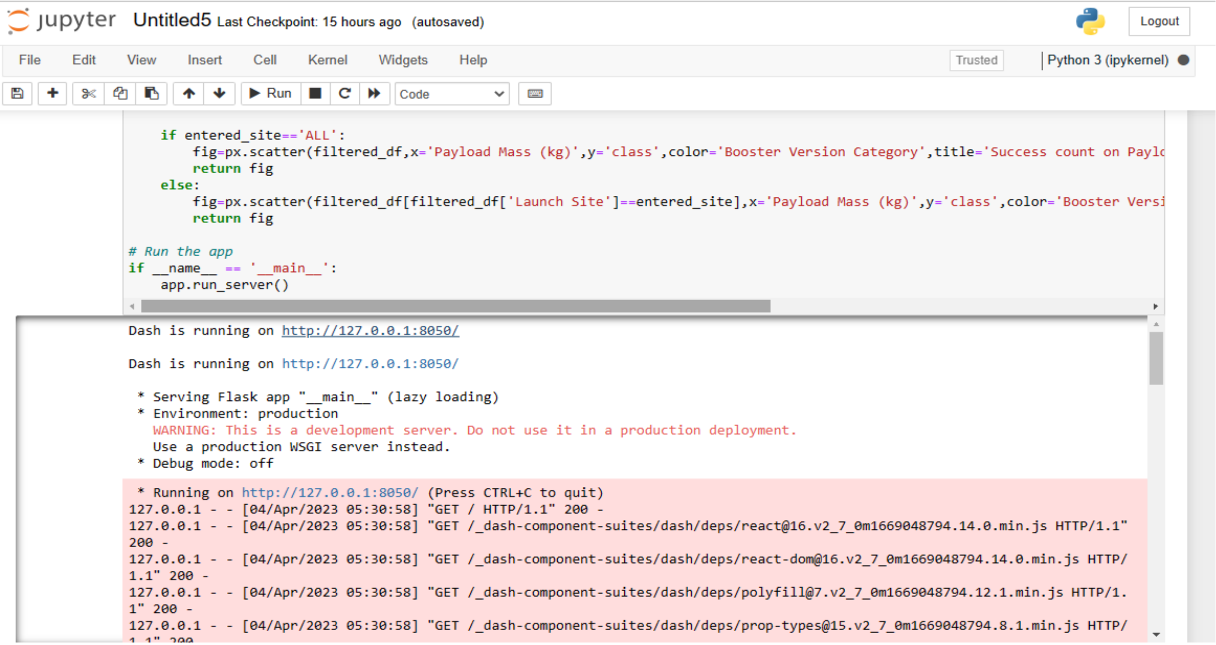
Build a Dashboard with Plotly Dash

How plotly Dashboard App Was Wade

- Due to network problems, on the IBM cloud, this exercise was completed in a local host. There was a little bit of difference with the actual procedure, but the resultant plotly app is similar to the outcome of the app made from the IBM cloud account.
- The simple steps followed were, all necessary libraries were called to jupyter notebook through anaconda distribution, and the dash app is plotted on a separate windows. Part of the code used is given here below.
- The completed jupyter notebook can be accessed on GitHub link given on the following pages.

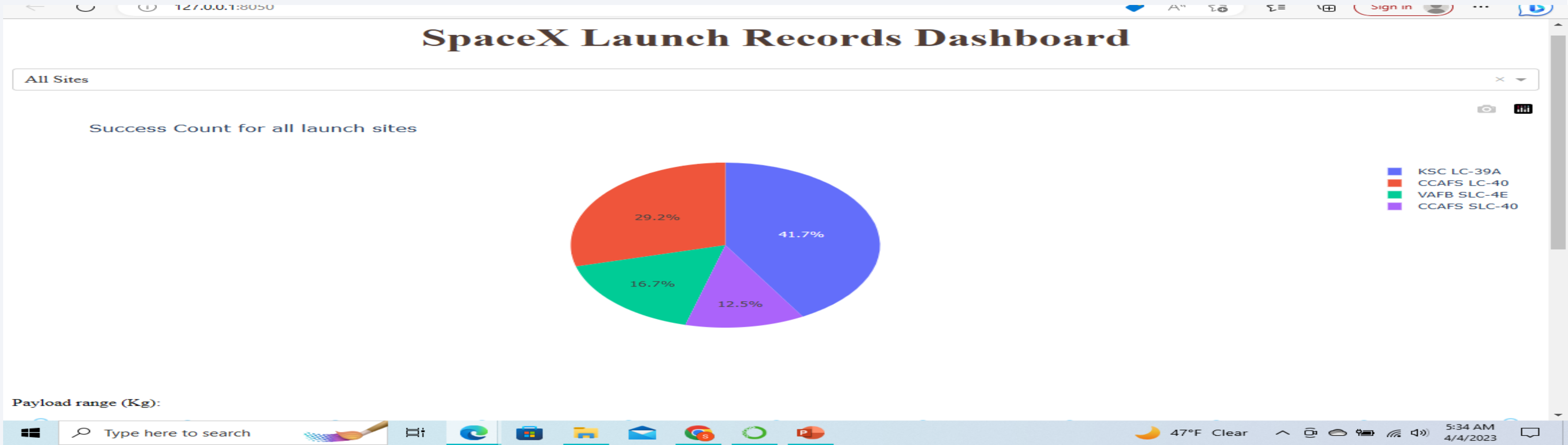






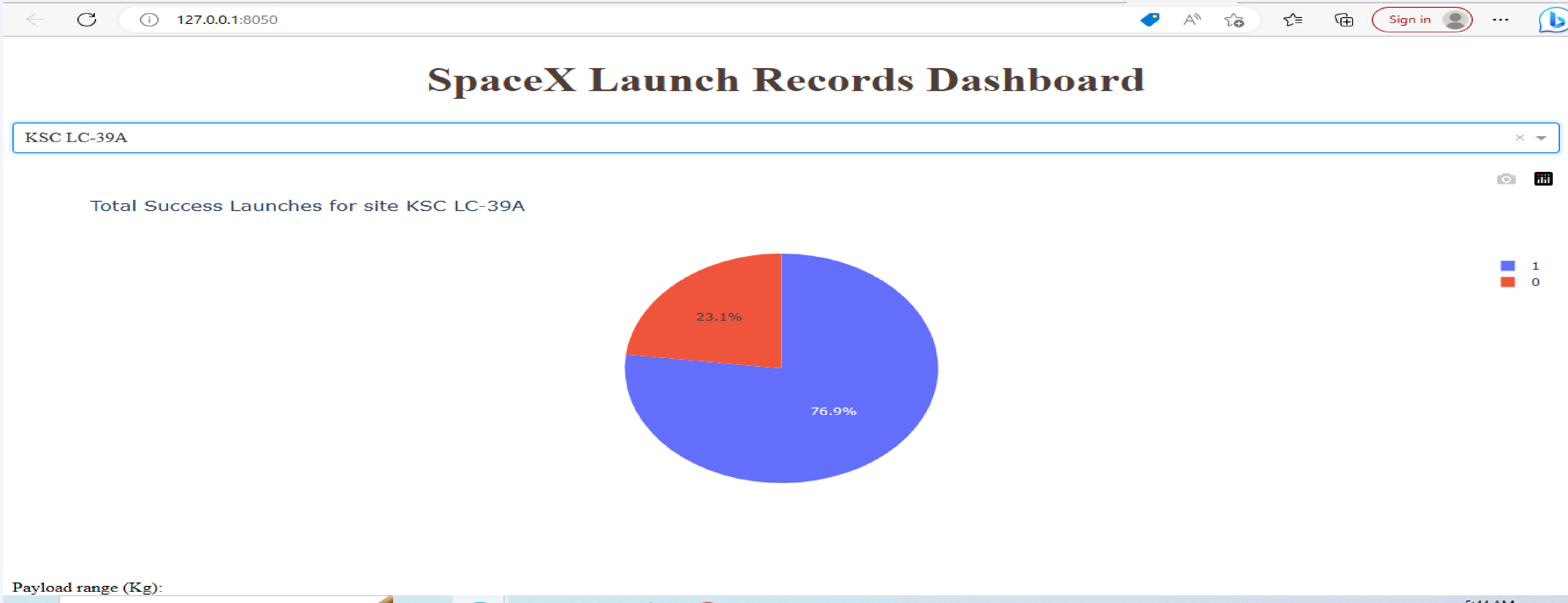
SpaceX Launch Records Dashboard

- The screenshot shows a pie chart launch success count for all sites,
- The pie chart shows the percentages of success of all the launch sites
- Accordingly launch site KSC LC 39A has the highest success count at launch(blue color) with 41.7%, while CCAFS SLC-40 has the lowest(purple color) with 12.5%.



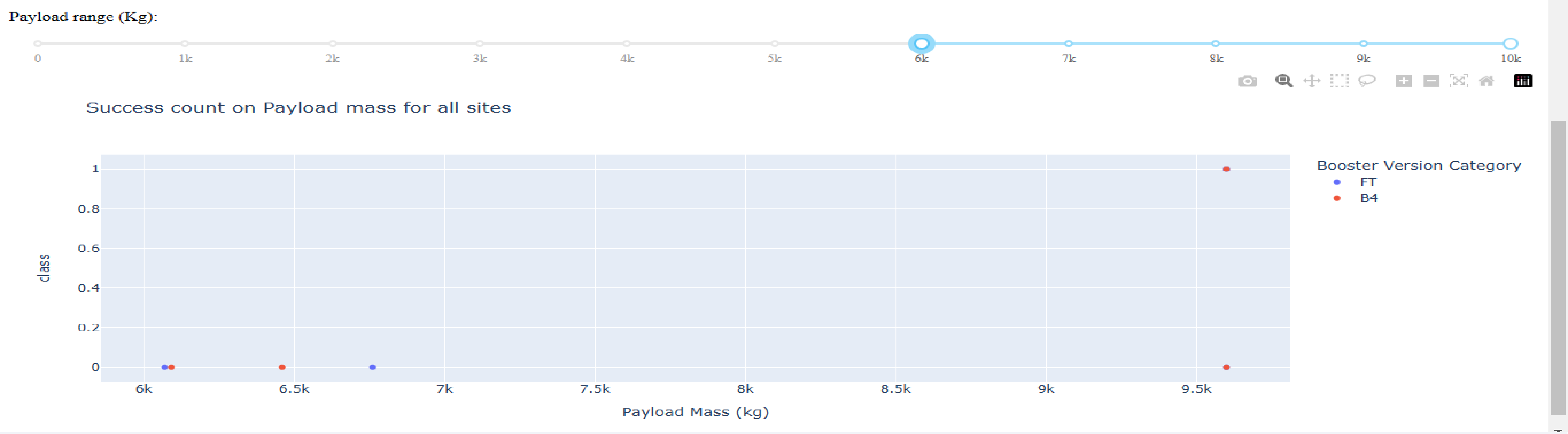
Launch site with highest success ratio

- The highest launch site ration of all the launching sites goes to KSC LC 39A with nearly 77% success ration (76.9%)



Payload Vs Launch Outcome

- Show screenshots of Payload vs. Launch Outcome scatter plot for all sites, with different payload selected in the range slider
- As it can be seen on the screenshot, if the range slider is made b/n 6k and 10k only FT and B4 booster version have the largest success rate.



Section 5

Predictive Analysis (Classification)

Classification Accuracy

- The built classification models in the exercise are given in a tabular form below
- The highest classification accuracy model for accuracy is KNN with 94% accuracy

2]:
```
accuracy_dict = {   'Model': ['Logistic Regression', 'SVM','Decision Tree','KNN'], 'Accuracy': [acc_LR, acc_svm, acc_tree, acc_knn]   }
df_accuracy = pd.DataFrame(accuracy_dict )
blankIndex=[''] * len(df_accuracy)
df_accuracy.index=blankIndex
df_accuracy
```

2]:

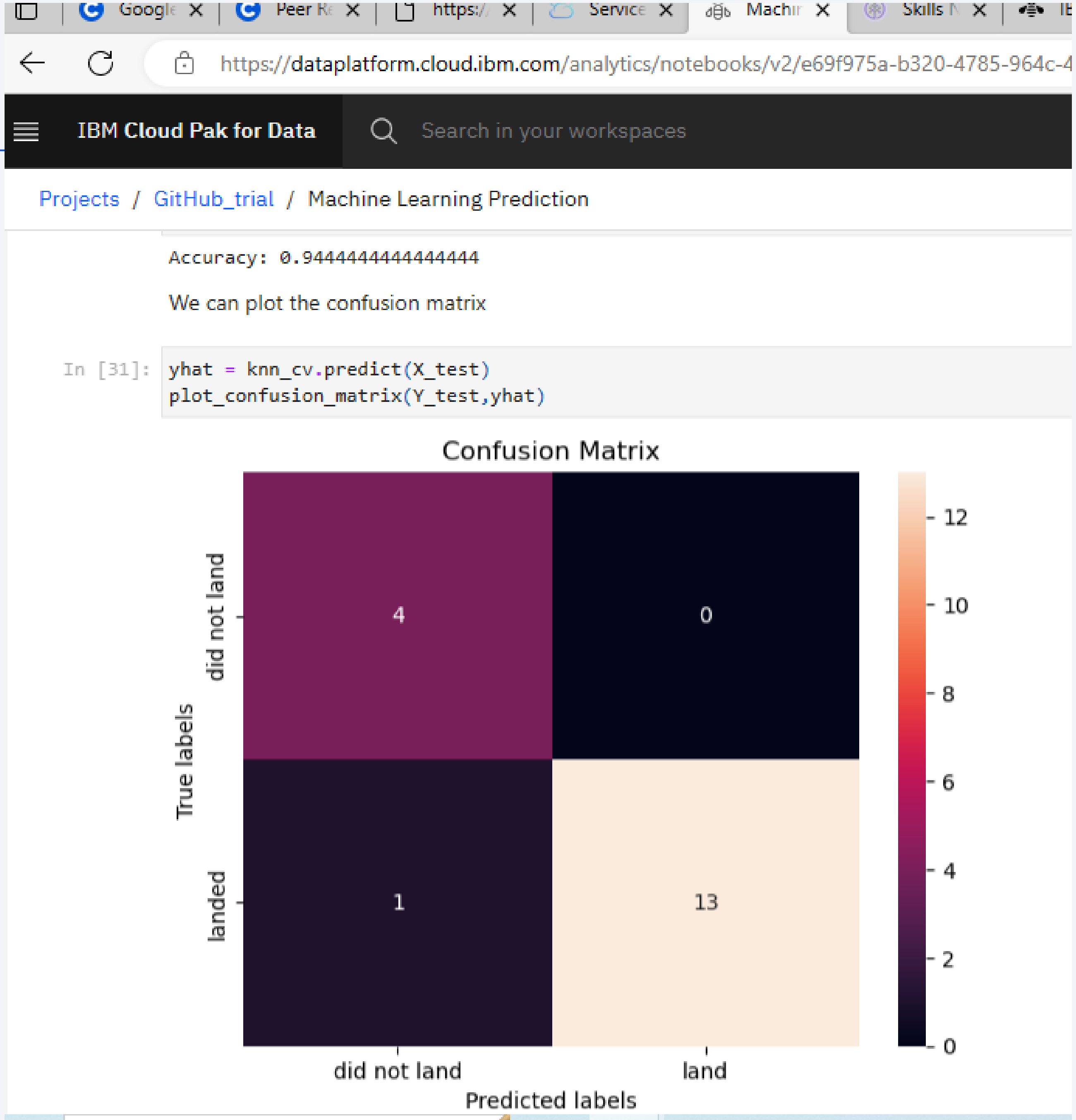
Model	Accuracy
Logistic Regression	0.888889
SVM	0.888889
Decision Tree	0.666667
KNN	0.944444

Conclusion

- KNN Method performs best on the given data set, as it's accuracy is better than SVM and LR, Decision Tree performs the worst

Confusion Matrix

- Show the confusion matrix of the best performing model with an explanation



Conclusions

- From the above, different analysis it can be concluded that:
  - The larger number of flights, at a given launch sites, the greater number of successful rates
  - Launch site success rate increased generally from 2013 to 2020
  - Orbits, ES-L1, GEO, HEO, SSO, and VLEO had the most success rates compared to the rest of the orbits
  - KSC LC-39A had the most successful launches than any of launch sites used
  - Accuracy of the decision Tree Classifier is the best classifier in the predictive analysis part of the task
  - We have more successful launches to failures in the Falcon 9 mission in all the verssions

Appendix

- Include any relevant assets like Python code snippets, SQL queries, charts, Notebook outputs, or data sets that you may have created during this project

Thank you!