IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Dr Eduardo Nicolas Schulz
28/02/2022

Outline

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

Executive Summary

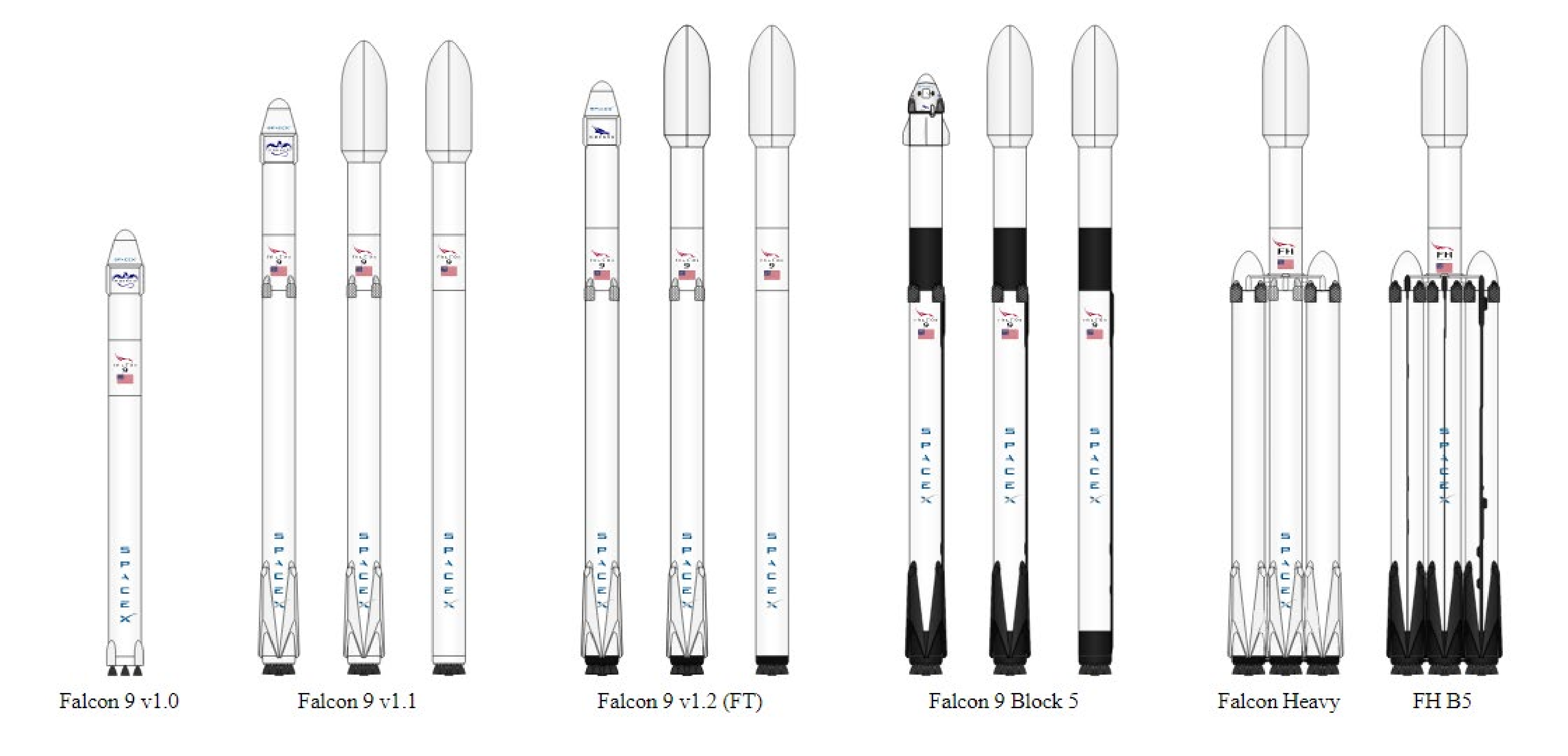
- The following study aims to determine the probability of a successful rocket landing from the company Space-X. In particular the Falcon 9 model. SpaceX advertises Falcon 9 rocket launches on its website with a cost of 62 million dollars; other providers cost upward of 165 million dollars each, much 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.
- Results summary:
- We observed correlations between successful landings and factors such as launch site location, proximity to landmarks, payload and type of orbit achieved.
- Plotting the data showed that the success rate since 2013 kept increasing till 2020
- The KSC LC 39A site, which is the furthest from the coastline, shows the largest success rates.
- The Falcon rockets models FT and B4 seem to have the highest success rates overall.
- The highest success rates seems to be for payloads between 5000 and 7500 Kg

Introduction

- 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.
- We want to find correlations between different factors, such as location, distance to landmarks, to cite some, to be able to predict if Space X will be able to recover the first stages of its launches, based on their historical data. i.e. We will predict if the Falcon 9 first stage will land successfully.

Introduction – Types of rockets

- This image illustrates the different types of Falcon rockets used by Space X



Section 1

Methodology

Methodology

Executive Summary

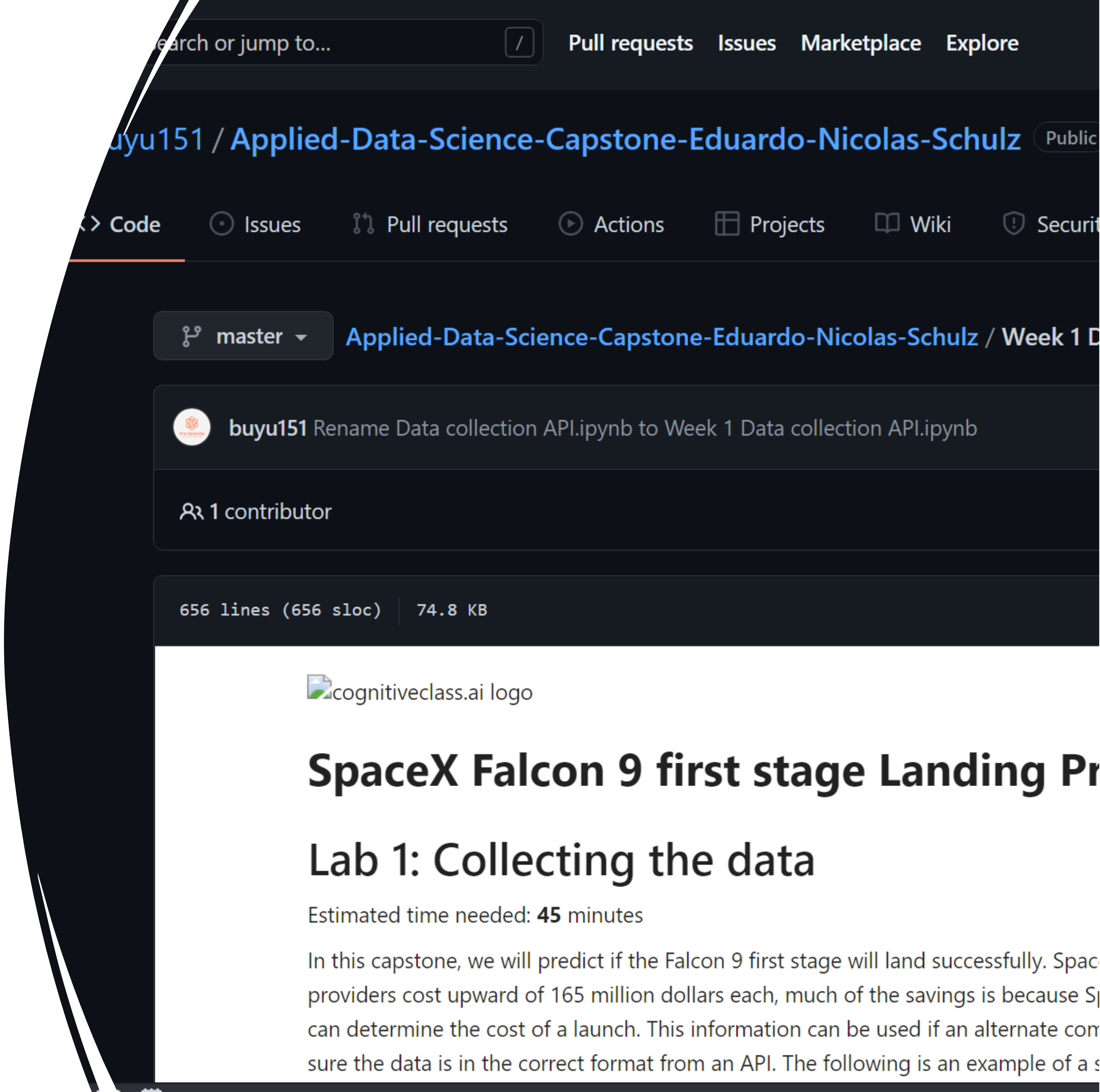
- Data collection methodology:
  - Data was collected in two ways
    - Data from web scraping a Wikipedia page with data tables
    - Form space x API
- Data wrangling
  - The obtained data was then explored using python with Jupyter notebooks.
- A exploratory data analysis (EDA) using visualization and SQL was performed.
- Interactive visual analytics using Folium and Plotly Dash were also explored.
- Finally, predictive analysis using classification models were utilized

Data Collection

- Data was collected mainly by two means:
- Web scraping a static version of a Wikipedia page containing tables of Flacon rocket launches.
- This was done using the BeautifulSoupe library in python.
- The second way was through the Space X API, by using the request library to get Jason packages form the API containing launch data.

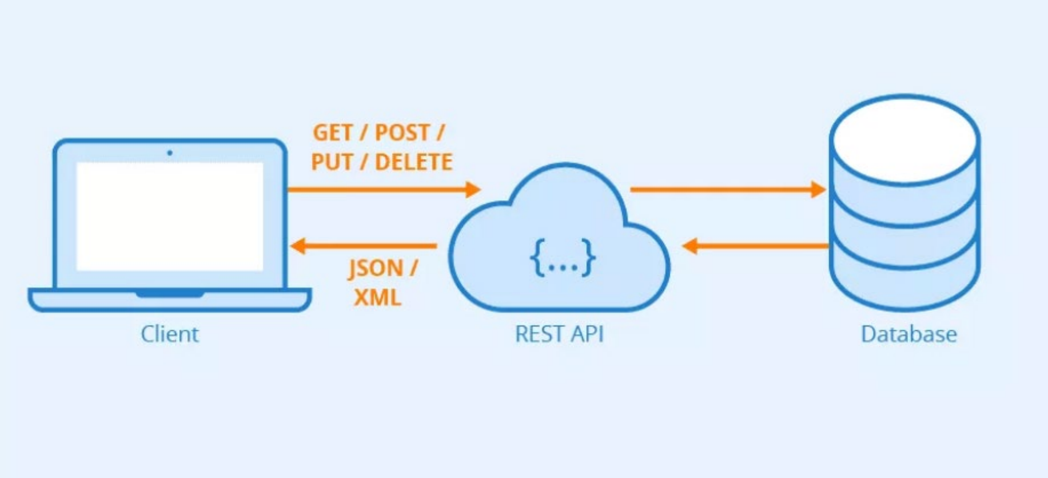
Data Collection – SpaceX API

- GitHub URL to the Jupyter notebook: https://github.com/buyu151/Applied-Data-Science-Capstone-Eduardo-Nicolas-Schulz/blob/master/Week%201%20Data%20collection%20API.ipynb



Data Collection – SpaceX API

- API data collection workflow



Import necessary libraries:

requests, pandas, numpy and datetime

We then get the API URL and assign it to an object:

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

We can then use that object to request information in the form of a json file (similar to a python dictionary):

response = requests.get(spacex_url)

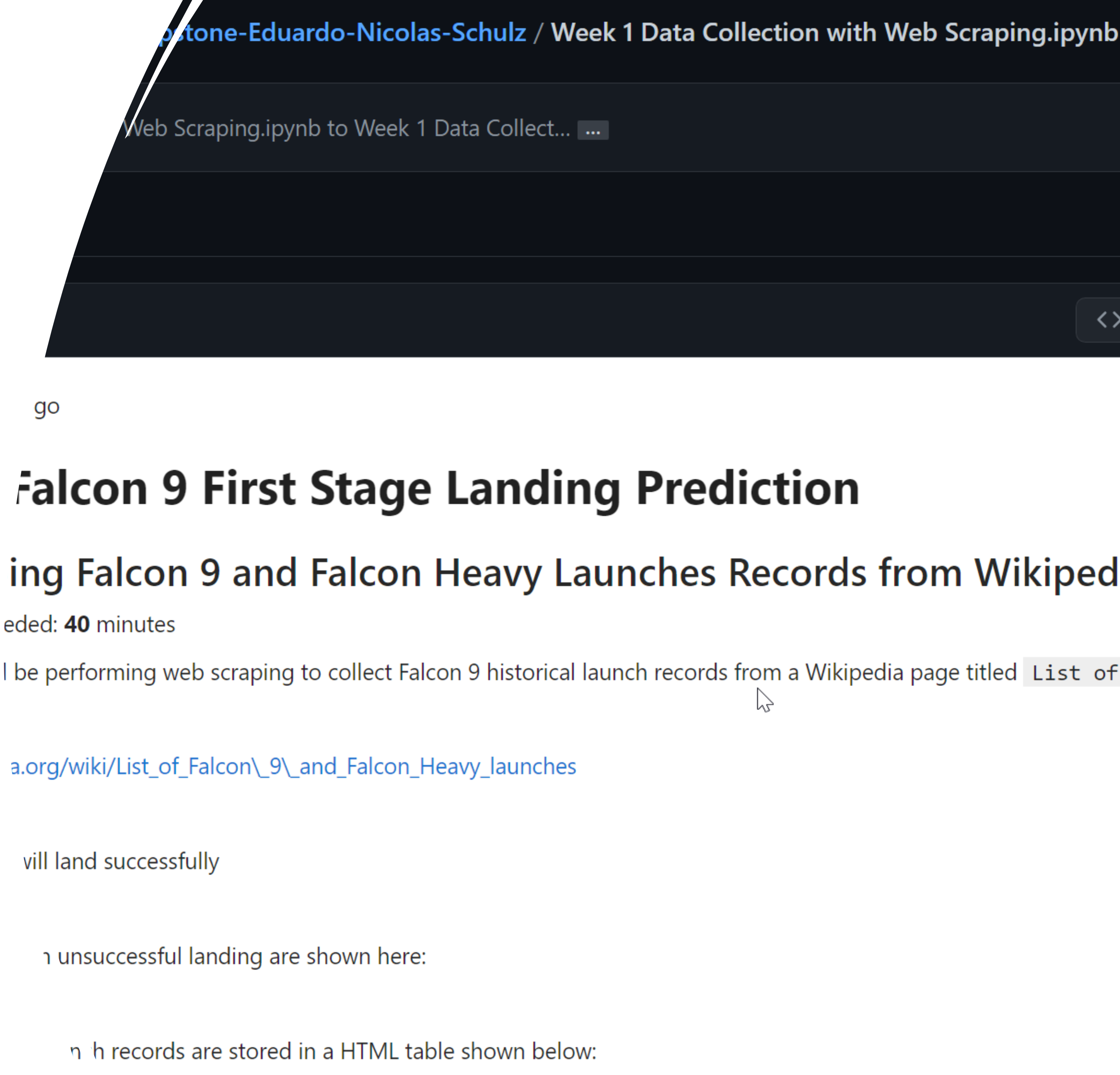
response.json()

Then we convert the json into a pandas dataframe for easier data manipulation and cleaning:

data = pd.json_normalize(response.json())

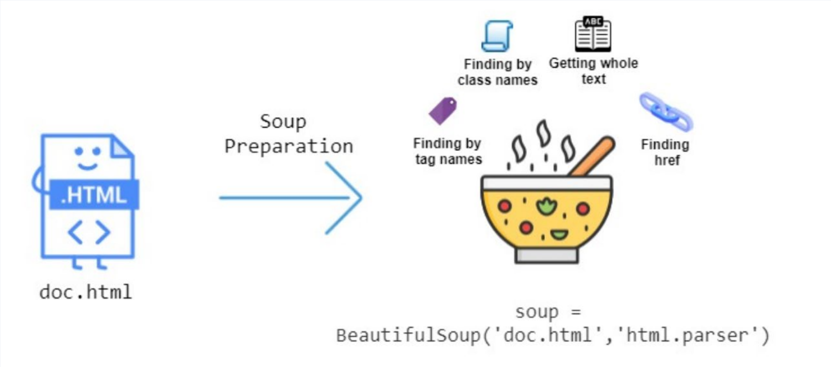
Data Collection - Scraping

- GitHub URL: https://github.com/buyu151/Applied-Data-Science-Capstone-Eduardo-Nicolas-Schulz/blob/master/Week%201%20Data%20Collection%20with%20Web%20Scraping.ipynb



Data Collection - Scraping

- Present your web scraping process using key phrases and flowcharts



Import necessary libraries:

sys, request, BeautifulSoup, re, unicodedata and pandas

We then get the website URL and assign it to an object:

static_url = https://en.wikipedia.org/w/index.php?title=List_of_Falcon_9_and_Falcon_Heavy_launches&oldid=1027686922

We get a response from the website:

response = requests.get(static_url)

We then create a BeautifulSoup object from the HTML response

soup = BeautifulSoup(response.content, 'html.parser')

We then explore the tree that BeautifulSoupe creates out of the HTML, to extract the relevant tables and convert them to a pandas dataframe (See Notebook in the github link):

df = pd.DataFrame.from_dict(launch_dict, orient='index')

df = df.transpose()

Data Collection – Scraping – Website table

- The tables at the Wikipedia site were as follows

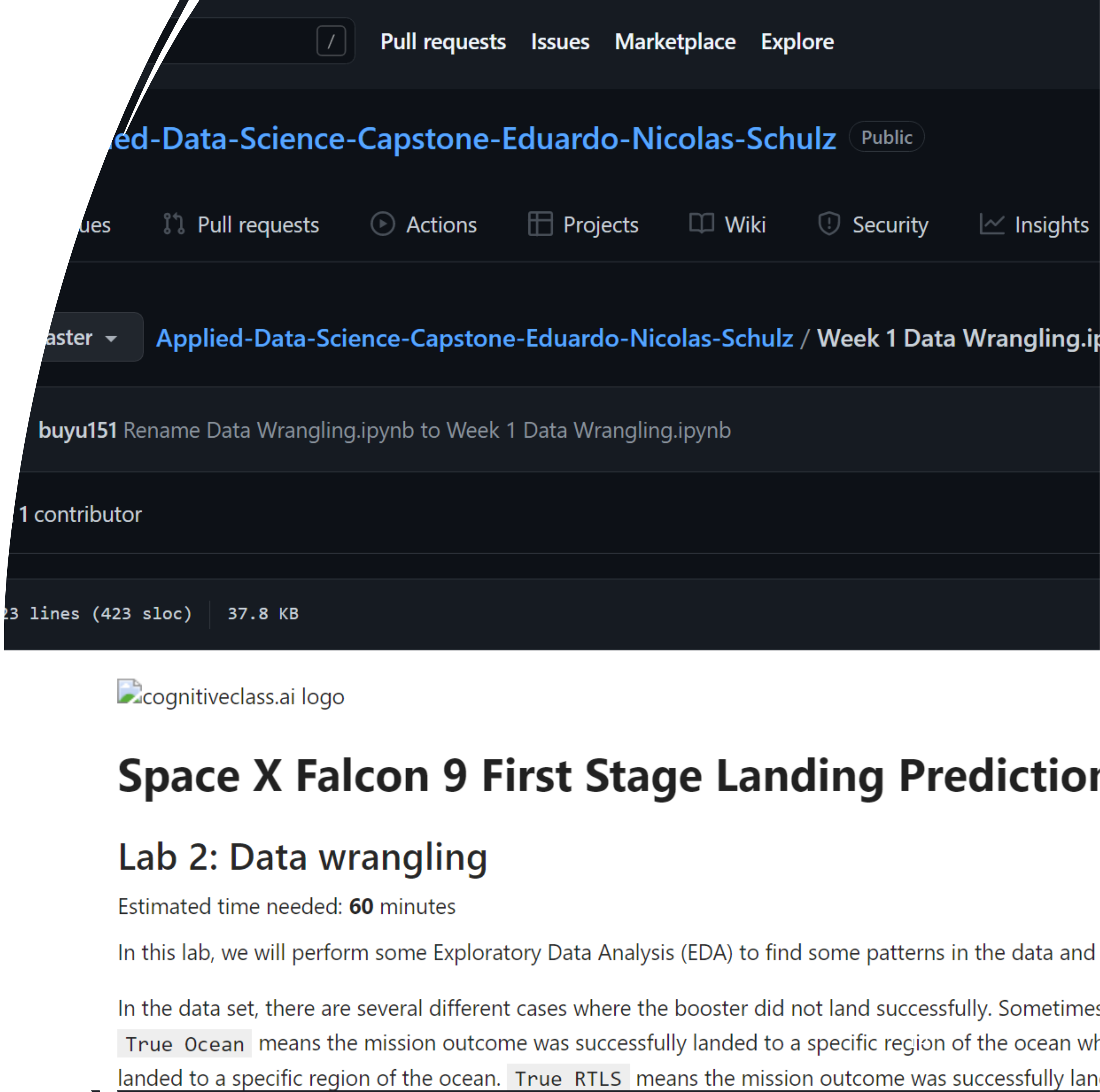
2020 [edit]

In late 2019, Gwynne Shotwell stated that SpaceX hoped for as many as 24 launches for Starlink satellites in 2020,[490] in addition to 14 or 15 non-Starlink launches. At 26 launches, 13 of which for Starlink satellites, Falcon 9 had its most prolific year, and Falcon rockets were second most prolific rocket family of 2020, only behind China's Long March rocket family.[491]

[hide] Flight No.	Date and time (UTC)	Version, Booster[b]	Launch site	Payload[c]	Payload mass	Orbit	Customer	Launch outcome	Booster landing
78	7 January 2020, 02:19:21[492]	F9 B5 ♺ B1049.4	CCAFS, SLC-40	Starlink 2 v1.0 (60 satellites)	15,600 kg (34,400 lb)[5]	LEO	SpaceX	Success	Success (drone ship)
	Third large batch and second operational flight of Starlink constellation. One of the 60 satellites included a test coating to make the satellite less reflective, and thus less likely to interfere with ground-based astronomical observations.[493]								
79	19 January 2020, 15:30[494]	F9 B5 ♺ B1046.4	KSC, LC-39A	Crew Dragon in-flight abort test[495] (Dragon C205.1)	12,050 kg (26,570 lb)	Sub-orbital[496]	NASA (CTS)[497]	Success	No attempt
	An atmospheric test of the Dragon 2 abort system after Max Q. The capsule fired its SuperDraco engines, reached an apogee of 40 km (25 mi), deployed parachutes after reentry, and splashed down in the ocean 31 km (19 mi) downrange from the launch site. The test was previously slated to be accomplished with the Crew Dragon Demo-1 capsule;[498] but that test article exploded during a ground test of SuperDraco engines on 20 April 2019.[419] The abort test used the capsule originally intended for the first crewed flight.[499] As expected, the booster was destroyed by aerodynamic forces after the capsule aborted.[500] First flight of a Falcon 9 with only one functional stage — the second stage had a mass simulator in place of its engine.								
80	29 January 2020, 14:07[501]	F9 B5 ♺ B1051.3	CCAFS, SLC-40	Starlink 3 v1.0 (60 satellites)	15,600 kg (34,400 lb)[5]	LEO	SpaceX	Success	Success (drone ship)
	Third operational and fourth large batch of Starlink satellites, deployed in a circular 290 km (180 mi) orbit. One of the fairing halves was caught, while the other was fished out of the ocean.[502]								
81	17 February 2020, 15:05[503]	F9 B5 ♺ B1056.4	CCAFS, SLC-40	Starlink 4 v1.0 (60 satellites)	15,600 kg (34,400 lb)[5]	LEO	SpaceX	Success	Failure (drone ship)
	Fourth operational and fifth large batch of Starlink satellites. Used a new flight profile which deployed into a 212 km × 386 km (132 mi × 240 mi) elliptical orbit instead of launching into a circular orbit and firing the second stage engine twice. The first stage booster failed to land on the drone ship[504] due to incorrect wind data.[505] This was the first time a flight proven booster failed to land.								
82	7 March 2020, 04:50[506]	F9 B5 ♺ B1059.2	CCAFS, SLC-40	SpaceX CRS-20 (Dragon C112.3 ♺)	1,977 kg (4,359 lb)[507]	LEO (ISS)	NASA (CRS)	Success	Success (ground pad)
	Last launch of phase 1 of the CRS contract. Carries *Bartolomeo*, an ESA platform for hosting external payloads onto ISS.[508] Originally scheduled to launch on 2 March 2020, the launch date was pushed back due to a second stage engine failure. SpaceX decided to swap out the second stage instead of replacing the faulty part.[509] It was SpaceX's 50th successful landing of a first stage booster, the third flight of the Dragon C112 and the last launch of the cargo Dragon spacecraft.								
83	18 March 2020, 12:16[510]	F9 B5 ♺ B1048.5	KSC, LC-39A	Starlink 5 v1.0 (60 satellites)	15,600 kg (34,400 lb)[5]	LEO	SpaceX	Success	Failure (drone ship)
	Fifth operational launch of Starlink satellites. It was the first time a first stage booster flew for a fifth time and the second time the fairings were reused (Starlink flight in May 2019).[511] Towards the end of the first stage burn, the booster suffered premature shut down of an engine, the first of a Merlin 1D variant and first since the CRS-1 mission in October 2012. However, the payload still reached the targeted orbit.[512] This was the second Starlink launch booster landing failure in a row, later revealed to be caused by residual cleaning fluid trapped inside a sensor.[513]								
84	22 April 2020, 19:30[514]	F9 B5 ♺ B1051.4	KSC, LC-39A	Starlink 6 v1.0 (60 satellites)	15,600 kg (34,400 lb)[5]	LEO	SpaceX	Success	Success (drone ship)

Data Wrangling – Github repository

- The Jupyter notebook conatianign the whole data wrangling process can be found at:
- https://github.com/buyu151/Applied-Data-Science-Capstone-Eduardo-Nicolas-Schulz/blob/master/Week%201%20Data%20Wrangling.ipynb



Data Wrangling

- We performed some Exploratory Data Analysis (EDA) to find some patterns in the data and determine what would be the label for training supervised models.

- In the data set, there are several different cases where the booster did not land successfully. Sometimes a landing was attempted but failed due to an accident; for example, “True Ocean” means the mission outcome was successfully landed to a specific region of the ocean while “False Ocean” means the mission outcome was unsuccessfully landed to a specific region of the ocean. “True RTLS” means the mission outcome was successfully landed to a ground pad “False RTLS” means the mission outcome was unsuccessfully landed to a ground pad. “True ASDS” means the mission outcome was successfully landed on a drone ship “False ASDS” means the mission outcome was unsuccessfully landed on a drone ship.

Data Wrangling – Workflow part 1

- The data was processed as follows:

Import packages Pandas and Numpy

Explore the data table with the head() attribute:

In [2]:
```
df=pd.read_csv("https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DS0321EN-SkillsNetwork/datasets/dataset_part_1.csv")
df.head(10)
```

Out[2]:

	FlightNumber	Date	BoosterVersion	PayloadMass	Orbit	LaunchSite	Outcome	Flights	GridFins	Reused	Legs	LandingPad	Block	ReusedCount	Serial	Longitude	Latitude
0	1	2010-06-04	Falcon 9	6104.959412	LEO	CCAFS SLC 40	None None	1	False	False	False	NaN	1.0	0	B0003	-80.577366	28.561857
1	2	2012-05-22	Falcon 9	525.000000	LEO	CCAFS SLC 40	None None	1	False	False	False	NaN	1.0	0	B0005	-80.577366	28.561857
2	3	2013-03-01	Falcon 9	677.000000	ISS	CCAFS SLC 40	None None	1	False	False	False	NaN	1.0	0	B0007	-80.577366	28.561857
3	4	2013-09-29	Falcon 9	500.000000	PO	VAFB SLC 4E	False Ocean	1	False	False	False	NaN	1.0	0	B1003	-120.610829	34.632093
4	5	2013-12-03	Falcon 9	3170.000000	GTO	CCAFS SLC 40	None None	1	False	False	False	NaN	1.0	0	B1004	-80.577366	28.561857
5	6	2014-01-06	Falcon 9	3325.000000	GTO	CCAFS SLC 40	None None	1	False	False	False	NaN	1.0	0	B1005	-80.577366	28.561857
6	7	2014-04-18	Falcon 9	2296.000000	ISS	CCAFS SLC 40	True Ocean	1	False	False	True	NaN	1.0	0	B1006	-80.577366	28.561857
7	8	2014-07-14	Falcon 9	1316.000000	LEO	CCAFS SLC 40	True Ocean	1	False	False	True	NaN	1.0	0	B1007	-80.577366	28.561857
8	9	2014-08-05	Falcon 9	4535.000000	GTO	CCAFS SLC 40	None None	1	False	False	False	NaN	1.0	0	B1008	-80.577366	28.561857
9	10	2014-09-07	Falcon 9	4428.000000	GTO	CCAFS SLC 40	None None	1	False	False	False	NaN	1.0	0	B1011	-80.577366	28.561857

Check for missing values with the attributes count and sum on the data frame object:

```
df.isnull().sum()/df.count()*100
```

```
3]: FlightNumber       0.000
    Date               0.000
    BoosterVersion     0.000
    PayloadMass        0.000
    Orbit              0.000
    LaunchSite         0.000
    Outcome            0.000
    Flights            0.000
    GridFins           0.000
    Reused             0.000
    Legs               0.000
    LandingPad        40.625
    Block              0.000
    ReusedCount        0.000
    Serial             0.000
    Longitude          0.000
    Latitude           0.000
    dtype: float64
```

We see around 40% of the Landing Pad data is missing values

Data Wrangling – Workflow part 2

We explore the types of the data in the data frame:

```
df.dtypes
```

```
4]:  FlightNumber        int64
     Date               object
     BoosterVersion     object
     PayloadMass       float64
     Orbit              object
     LaunchSite         object
     Outcome            object
     Flights             int64
     GridFins             bool
     Reused               bool
     Legs                 bool
     LandingPad         object
     Block             float64
     ReusedCount         int64
     Serial             object
     Longitude         float64
     Latitude          float64
     dtype: object
```

We can calculate the number of launches on each site:

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

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

And the occurrence of each type of orbit:

```
]:  # Apply value_counts on Orbit column
    df['Orbit'].value_counts()

[6]:  GTO      27
      ISS      21
      VLEO     14
      PO        9
      LEO       7
      SSO       5
      MEO       3
      HEO       1
      GEO       1
      SO        1
      ES-L1     1
      Name: Orbit, dtype: int64
```

We create a “Class” column using a binary encoding for success (1) or failure (0). This allows us to then calculate the success rate as follows:

```
In [12]:  df['Class']=landing_class
          df[['Class']].head(8)

Out[12]:
```

	Class
0	0
1	0
2	0
3	0
4	0
5	0
6	1
7	1

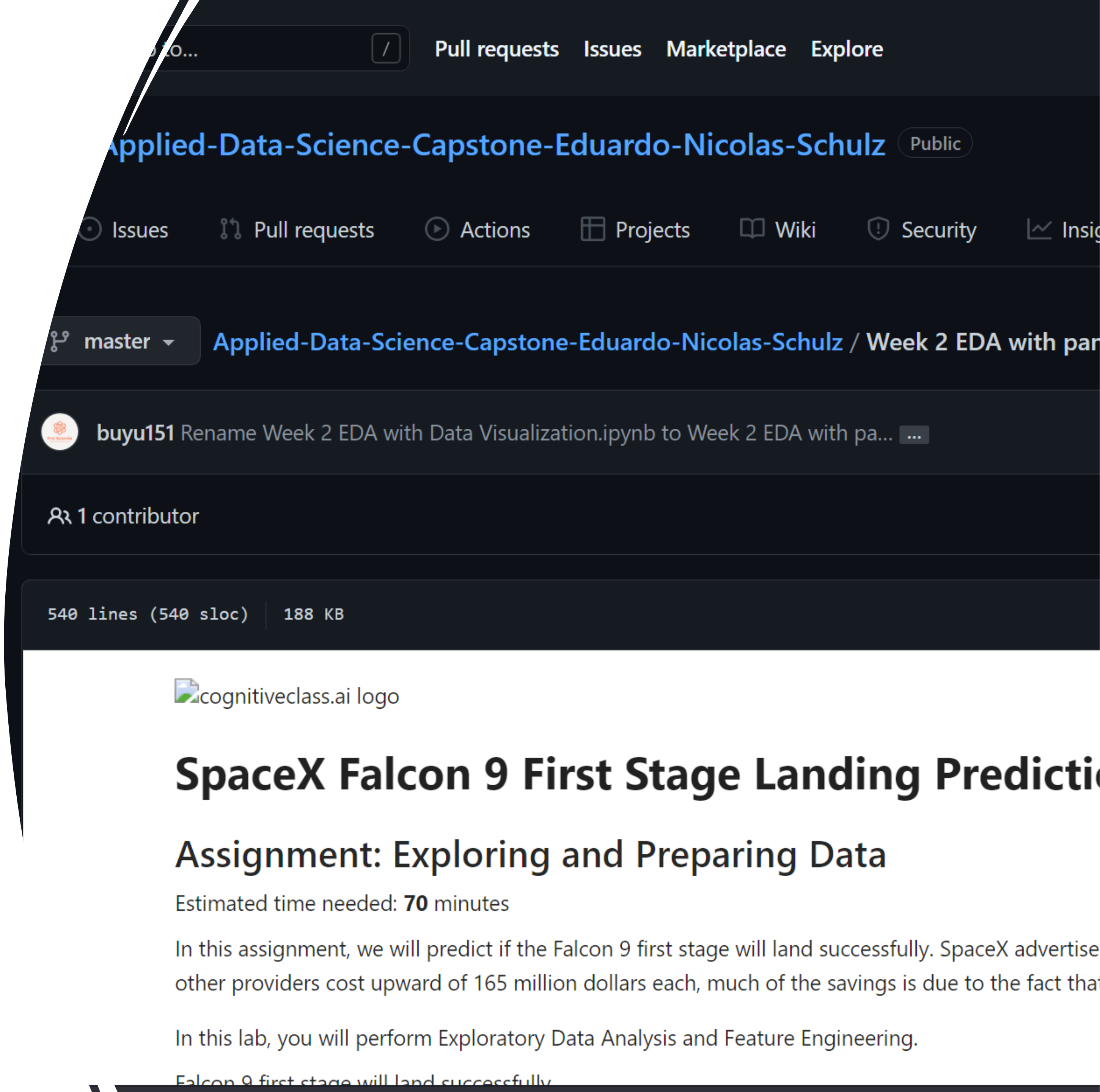
We can use the following line of code to determine the success rate:

```
In [14]:  df["Class"].mean()

Out[14]:  0.6666666666666666
```

66.66% success rate

EDA with Data Visualization

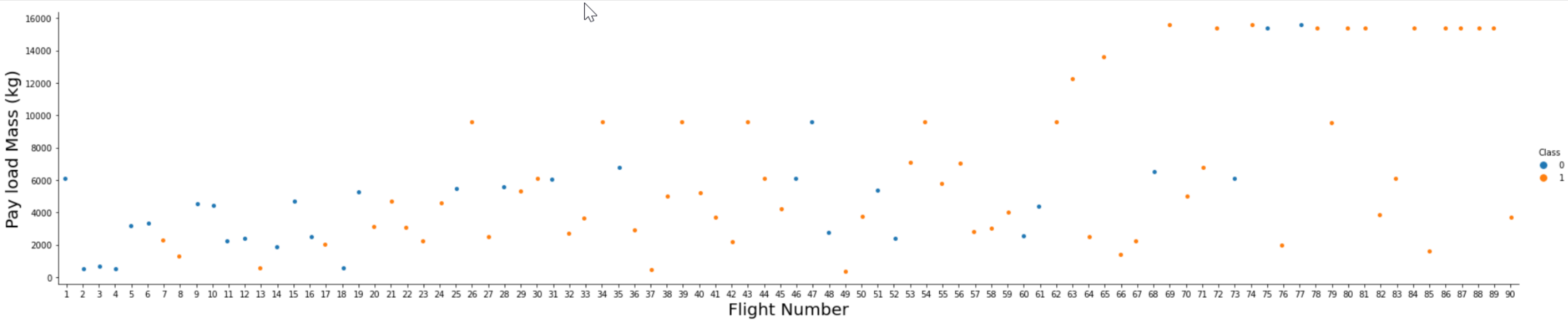
- In the following slides we explore different scatter plots to get an idea of the interaction between different variables and the success rates.
- The GitHub URL is:
- https://github.com/buyu151/Applied-Data-Science-Capstone-Eduardo-Nicolas-Schulz/blob/master/Week%202%20EDA%20with%20pandas%20and%20matplotlib.ipynb



EDA with Data Visualization

- First, we try to see how the `FlightNumber` (indicating the continuous launch attempts.) and `Payload` variables would affect the launch outcome:
- We see that different payloads have different success rates. Payloads of 14 tons being among the more consistently successful.

In [3]:
```
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()
```

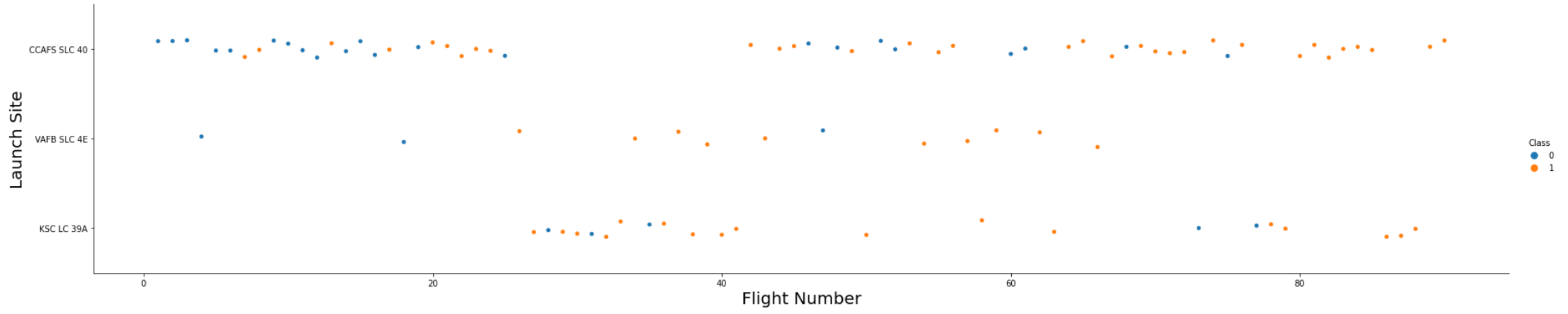


EDA with Data Visualization

- Similarly, we Visualize the relationship between Flight Number and Launch Site.
- We see that different launch sites have different success rates. “CCAFS LC-40”, has a success rate of 60 %, while “KSC LC-39A” and “VAFB SLC 4E” has a success rate of 77%.

In [4]:

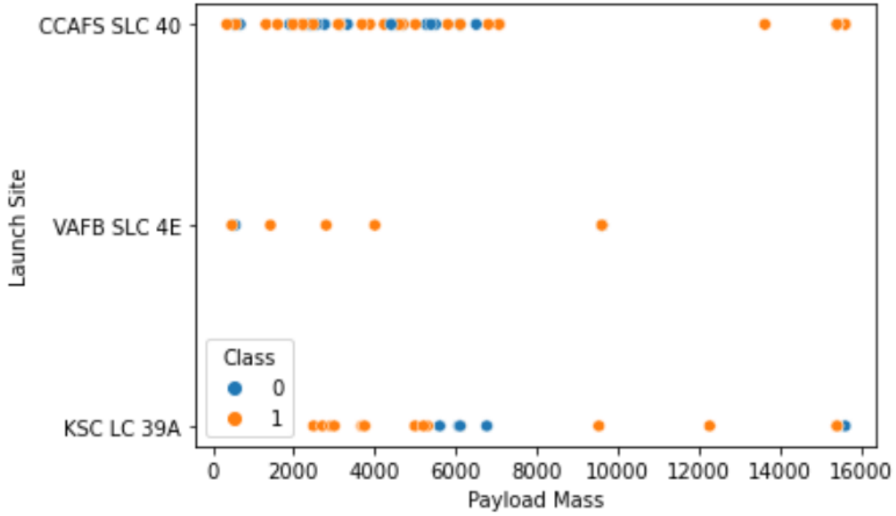
```
# Plot a scatter point chart with x axis to be Flight Number and y axis to be the launch site, and hue to be the class value
sns.catplot(y="LaunchSite", x="FlightNumber", hue="Class", data=df, aspect = 5)
plt.xlabel("Flight Number",fontsize=20)
plt.ylabel("Launch Site",fontsize=20)
plt.show()
```



EDA with Data Visualization

- Relationship between Payload and Launch Site
- We see that at the VAFB-SLC launch site there are no rockets launched for heavy payload mass greater than 10000 Kg.

```
sns.scatterplot(data=df, x="PayloadMass", y="LaunchSite", hue="Class")
plt.xlabel("Payload Mass",fontsize=10)
plt.ylabel("Launch Site",fontsize=10)
plt.show()
```

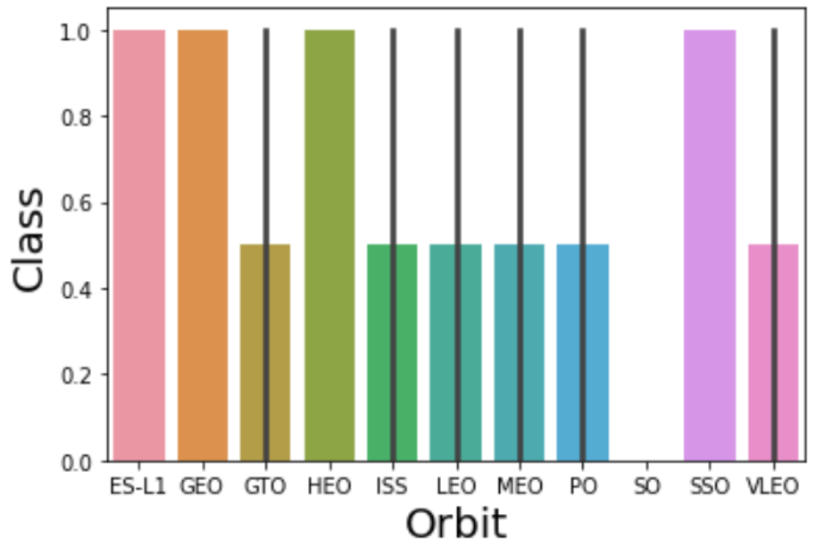


EDA with Data Visualization

- Relationship between success rate of each orbit type
- We see that at the SSO, GEO, ES-L1 and HEO orbits have the highest success rate.

In [7]:

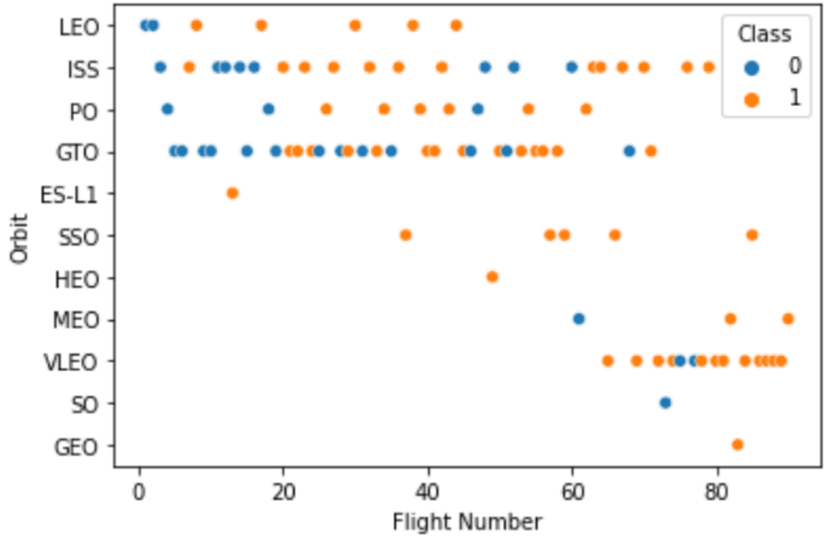
```
sns.barplot(y="Class", x="Orbit", data=t)

plt.xlabel("Orbit",fontsize=20)
plt.ylabel("Class",fontsize=20)
plt.show()
```



EDA with Data Visualization

- Relationship between FlightNumber and Orbit type
- We see that at the SSO, GEO, ES-L1, VLEO and HEO orbits have the highest success rate.

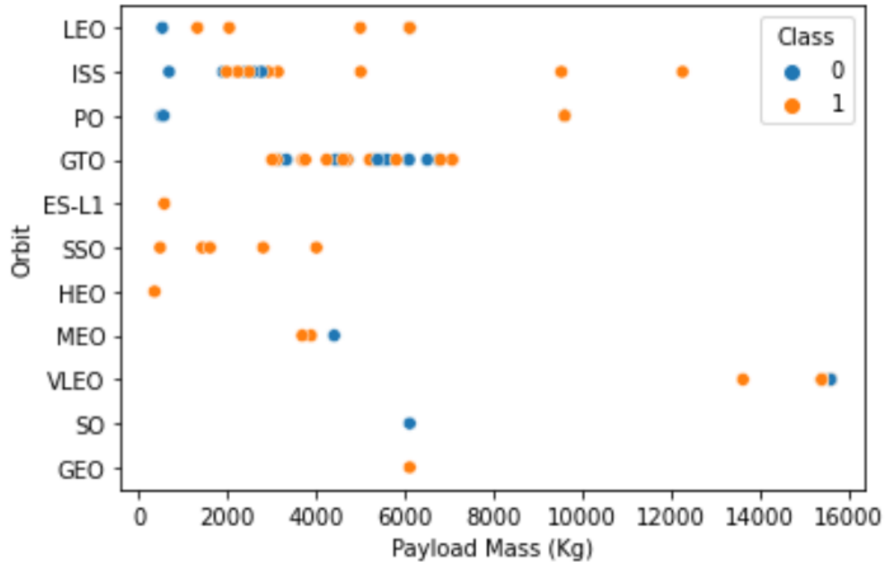
```
sns.scatterplot(data=df, x="FlightNumber", y="Orbit", hue="Class")
plt.xlabel("Flight Number",fontsize=10)
plt.ylabel("Orbit",fontsize=10)
plt.show()
```



EDA with Data Visualization

- Relationship between Payload and Orbit type
- With heavy payloads the successful landing or positive landing rate are more for Polar, LEO and ISS.
- However, for GTO we cannot distinguish this well as both positive landing rate and negative landing(unsuccessful mission) are both there here.

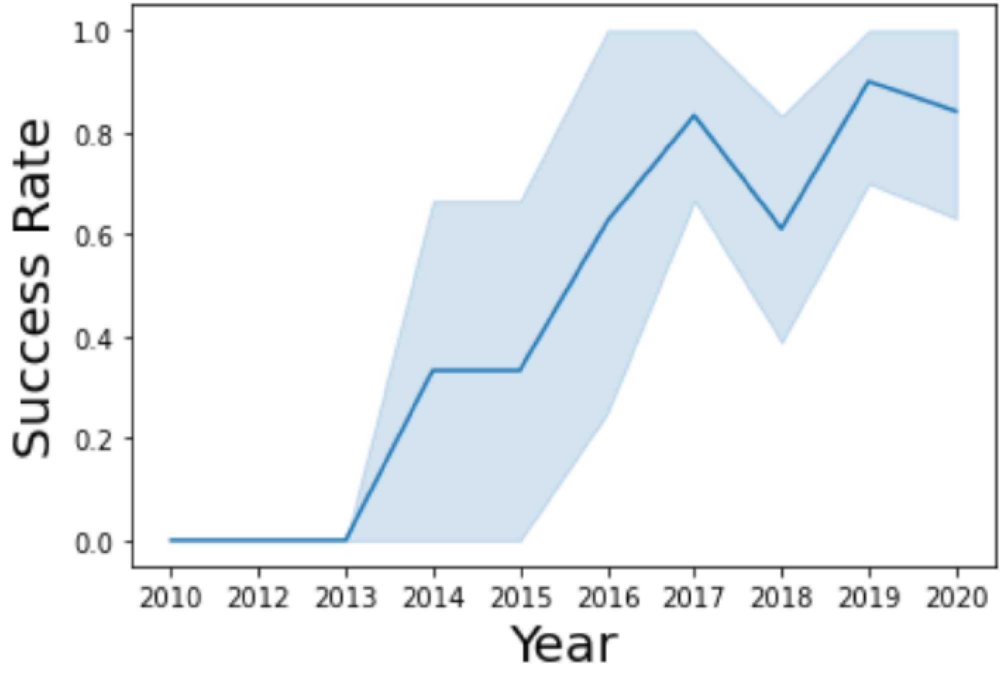
```
sns.scatterplot(data=df, x="PayloadMass", y="Orbit", hue="Class")
plt.xlabel("Payload Mass (Kg)",fontsize=10)
plt.ylabel("Orbit",fontsize=10)
plt.show()
```



EDA with Data Visualization

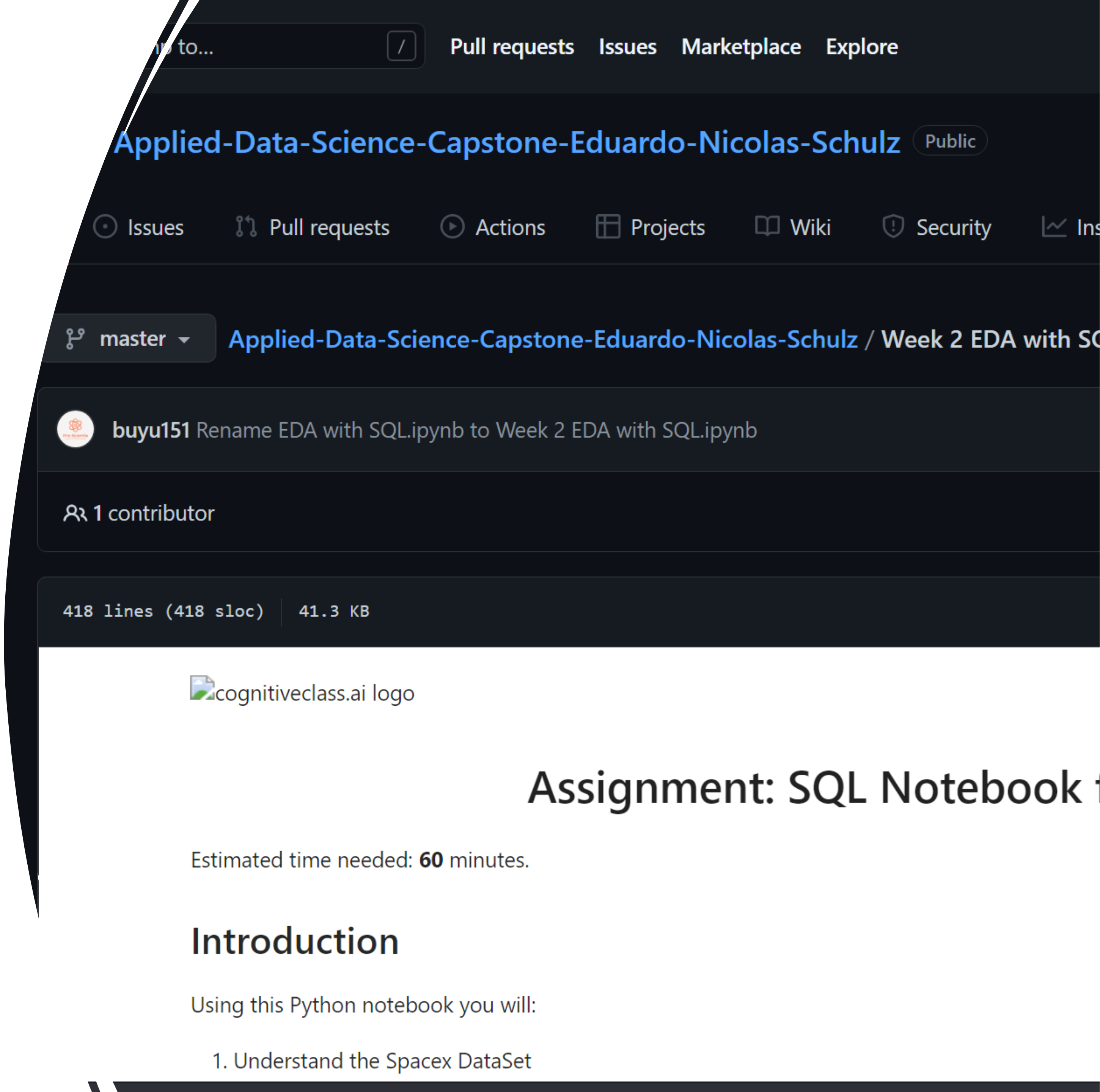
- Visualize the launch success yearly trend
- we can observe that the success rate since 2013 kept increasing till 2020

```
sns.lineplot(data=df1, x="Year", y="Class")
plt.xlabel("Year",fontsize=20)
plt.ylabel("Success Rate",fontsize=20)
plt.show()
```



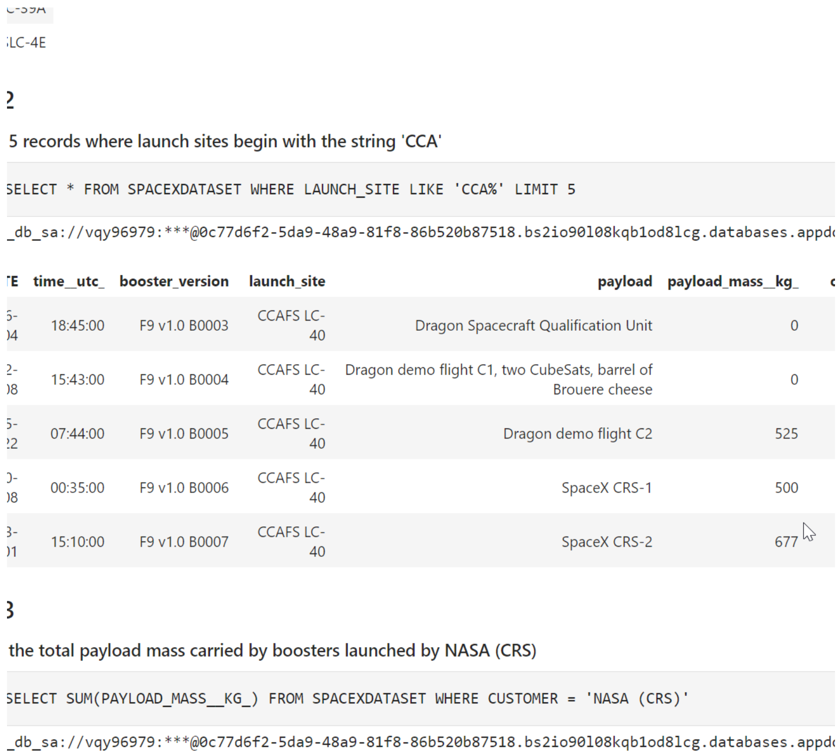
EDA with SQL

- The GitHub URL is: https://github.com/buyu151/Applied-Data-Science-Capstone-Eduardo-Nicolas-Schulz/blob/master/Week%202%20EDA%20with%20SQL.ipynb



EDA with SQL

- SQL queries performed:
  - Display the names of the unique launch sites in the space mission
    - %sql SELECT DISTINCT(LAUNCH_SITE) FROM SPACEXDATASET
  - Display 5 records where launch sites begin with the string 'CCA'
    - %sql SELECT * FROM SPACEXDATASET WHERE LAUNCH_SITE LIKE 'CCA%' LIMIT 5
  - Display the total payload mass carried by boosters launched by NASA (CRS)
    - %sql SELECT SUM(PAYLOAD_MASS__KG_) FROM SPACEXDATASET WHERE CUSTOMER = 'NASA (CRS)'
  - Display average payload mass carried by booster version F9 v1.1
    - %sql SELECT AVG(PAYLOAD_MASS__KG_) FROM SPACEXDATASET WHERE BOOSTER_VERSION = 'F9 v1.1'
  - List the date when the first successful landing outcome in ground pad was achieved.
    - %sql SELECT MIN(DATE) FROM SPACEXDATASET WHERE Landing__Outcome = 'Success (ground pad)'



time__utc_	booster_version	launch_site	payload	payload_mass__kg_
18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0
15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight C1, two CubeSats, barrel of Brouere cheese	0
07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525
00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500
15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677

EDA with SQL

- SQL queries performed:
  - List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000
    - %sql SELECT BOOSTER_VERSION FROM SPACEXDATASET WHERE Landing__Outcome = 'Success (drone ship)' AND PAYLOAD_MASS__KG_ > 4000 AND PAYLOAD_MASS__KG_ < 6000
  - List the total number of successful and failure mission outcomes
    - %sql SELECT COUNT(MISSION_OUTCOME) FROM SPACEXDATASET WHERE MISSION_OUTCOME LIKE 'Success%' OR MISSION_OUTCOME LIKE 'Failure%'
  - ***List the names of the booster_versions which have carried the maximum payload mass. Use a subquery***
    - %sql SELECT BOOSTER_VERSION FROM SPACEXDATASET WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXDATASET)
  - ***List the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015***
    - %sql SELECT Date, Landing__Outcome, Booster_Version, Launch_Site FROM SPACEXDATASET WHERE Landing__Outcome = 'Failure (drone ship)' AND Date BETWEEN '2015-01-01' AND '2015-12-30'
  - ***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***
    - %sql SELECT * FROM SPACEXDATASET WHERE Landing__Outcome LIKE 'Success%' AND (DATE BETWEEN '2010-06-04' AND '2017-03-20') ORDER BY DATE DESC

[67]:

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

Task 10

Rank the count of landing outcomes (such as Failure (drone ship) or Success (ground pad)) between

[68]:
```
%sql SELECT * FROM SPACEXDATASET WHERE Landing__Outcome LIKE 'Success%' AND (DATE BETWEE
```

```
 * ibm_db_sa://vqy96979:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.dat
Done.
```

[68]:

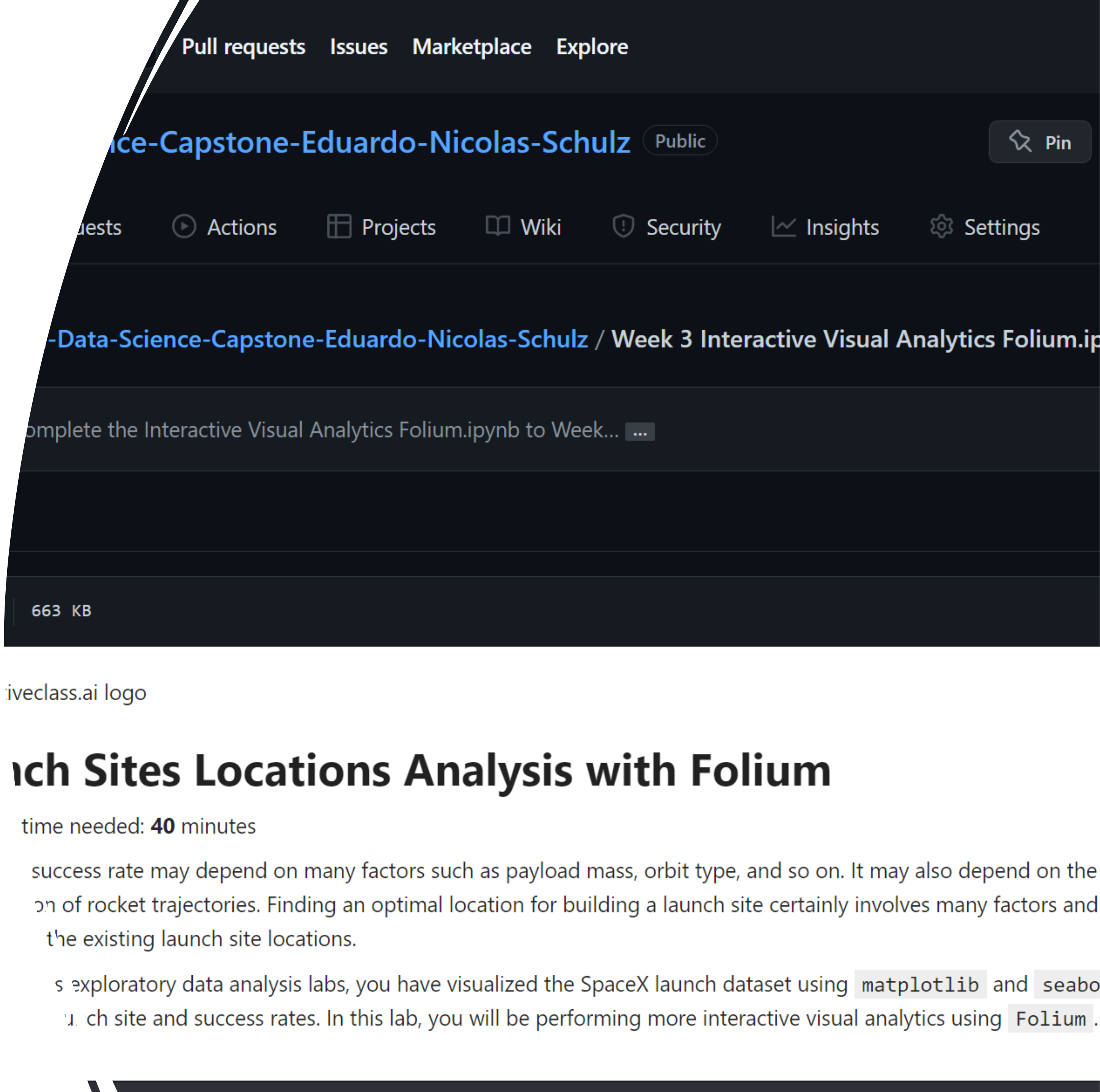
DATE	time__utc_	booster_version	launch_site	payload	payload_mass__kg
2017-02-19	14:39:00	F9 FT B1031.1	KSC LC-39A	SpaceX CRS-10	249
2017-01-14	17:54:00	F9 FT B1029.1	VAFB SLC-4E	Iridium NEXT 1	960
2016-08-14	05:26:00	F9 FT B1026	CCAFS LC-40	JCSAT-16	460
2016-07-18	04:45:00	F9 FT B1025.1	CCAFS LC-40	SpaceX CRS-9	225
2016-05-27	21:39:00	F9 FT B1023.1	CCAFS LC-40	Thaicom 8	310
2016-05-06	05:21:00	F9 FT B1022	CCAFS LC-40	JCSAT-14	469
2016-04-08	20:43:00	F9 FT B1021.1	CCAFS LC-40	SpaceX CRS-8	313
2015-12-22	01:29:00	F9 FT B1019	CCAFS LC-40	OG2 Mission 2 11 Orbcomm-OG2 satellites	203

[69]:
```
%sql SELECT * FROM SPACEXDATASET WHERE Landing__Outcome LIKE 'Failure%' AND (DATE BETWEE
```

```
 * ibm_db_sa://vqy96979:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.dat
```

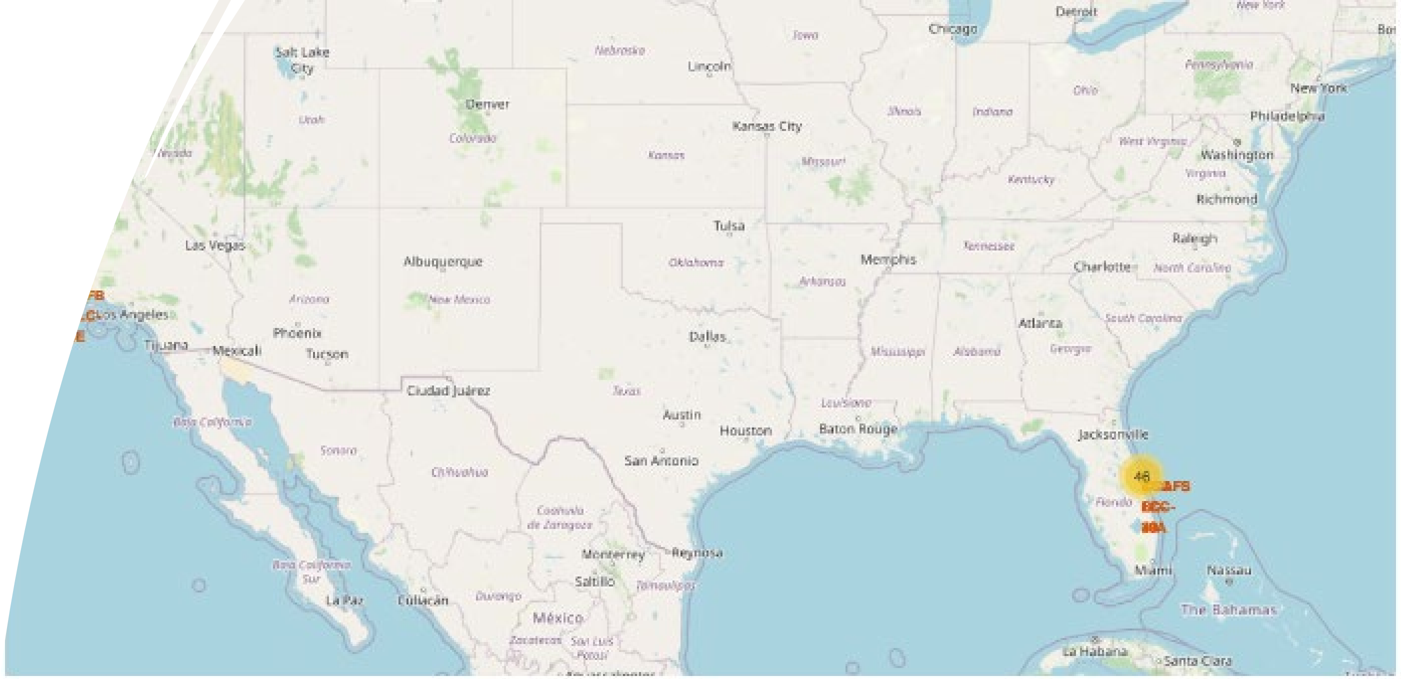
Build an Interactive Map with Folium

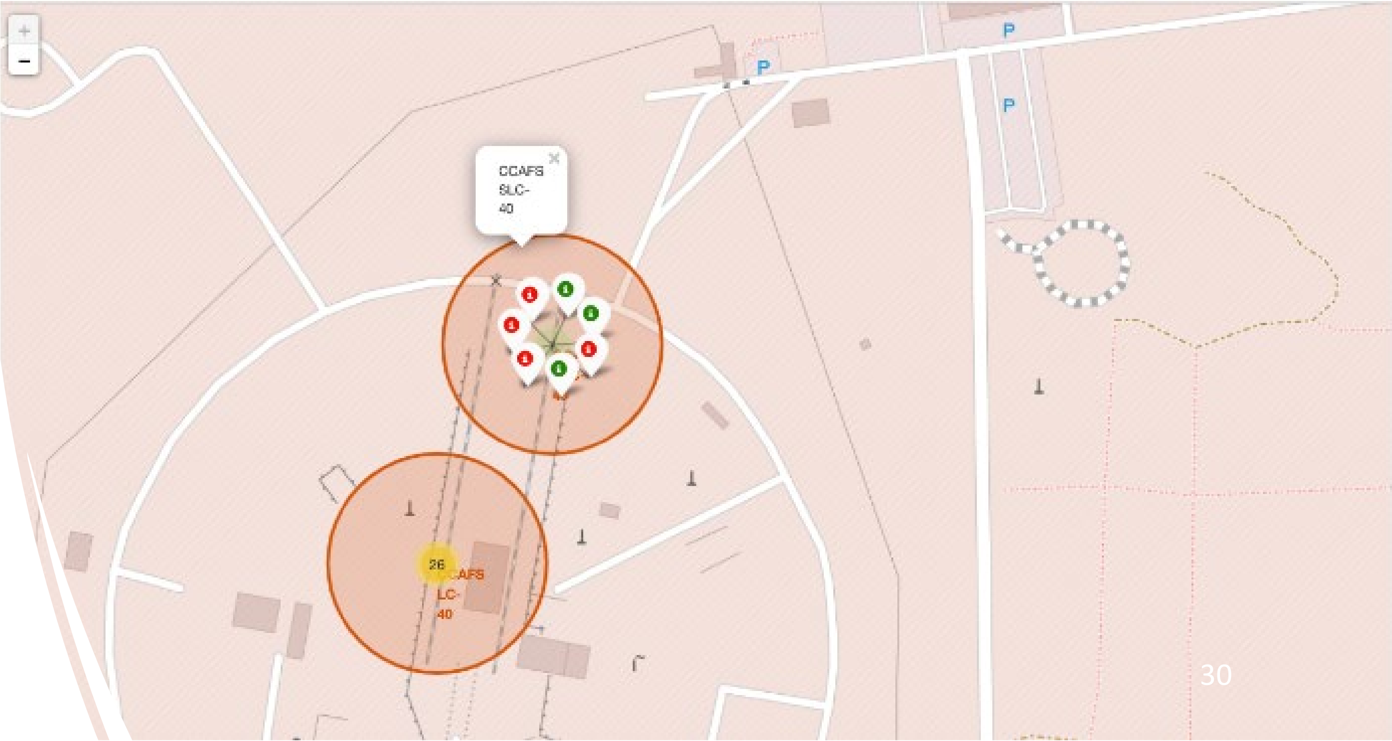
- The GitHub URL is: https://github.com/buyu151/Applied-Data-Science-Capstone-Eduardo-Nicolas-Schulz/blob/master/Week%203%20Interactive%20Visual%20Analytics%20Folium.ipynb



Build an Interactive Map with Folium

- Mark all launch sites on a map
  - We added each site's location on a map using site's latitude and longitude coordinates
  - This easily allows visualization of launch locations in the USA and their position relative to geographical features, such as coastlines.
- Mark the success/failed launches for each site on the map
  - We then enhance the map by adding the launch outcomes for each site, and see which sites have high success rates.
  - This lets us visually infer if there is a particular site with a higher rate of success and if that site has some geographical feature that might contribute to it.
- Calculate the distances between a launch site to its proximities
  - We explore and analyse the proximities of launch sites.
  - This quantifies the distances to features that might affect the success rates of a site.

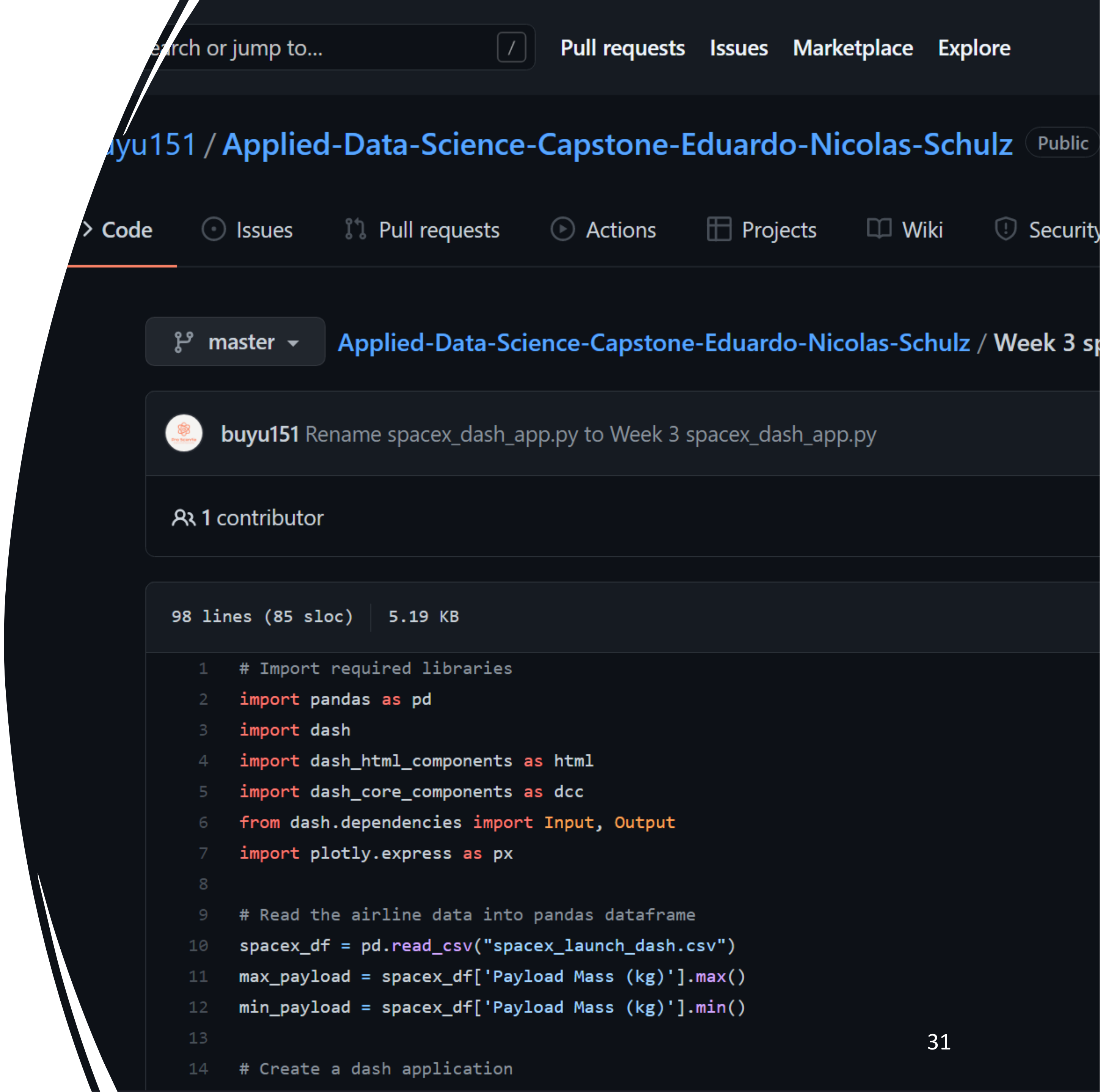




Build a dashboard with Plotly Dash

- The GitHub URL is:

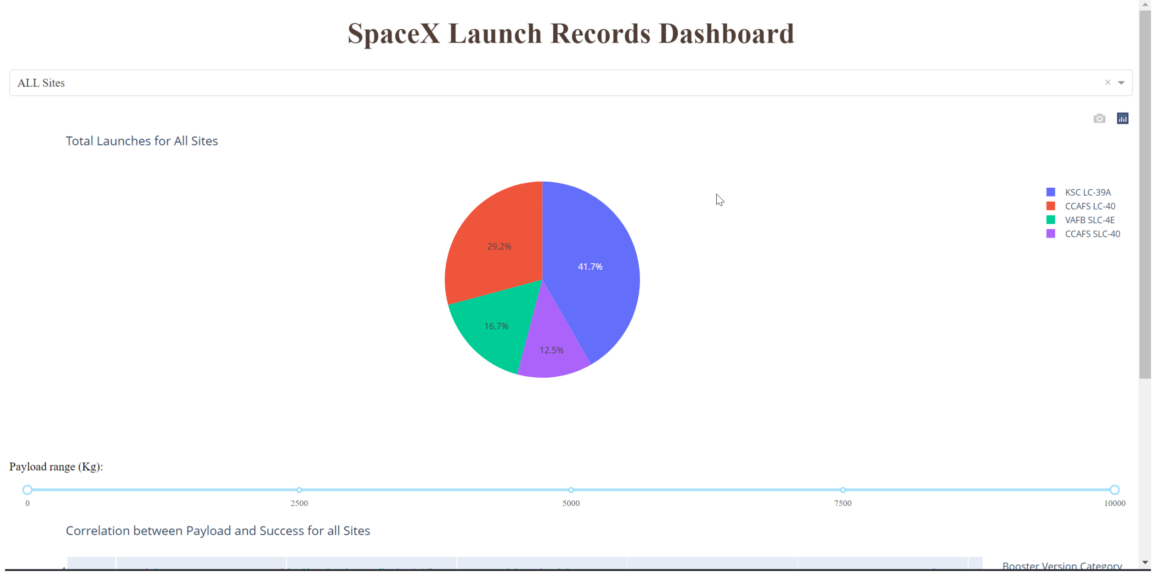
https://github.com/buyu151/Applied-Data-Science-Capstone-Eduardo-Nicolas-Schulz/blob/master/Week%203%20spacex_dash_app.py



Build a Dashboard with Plotly Dash

The first interactive plot is a pie chart of all launch sites success rates, along with a drop-down menu to select each individual launch site's success rates.

The second plot is an interactive scatter plot of success rates vs. Payload Mass , where the payload range can be adjusted.



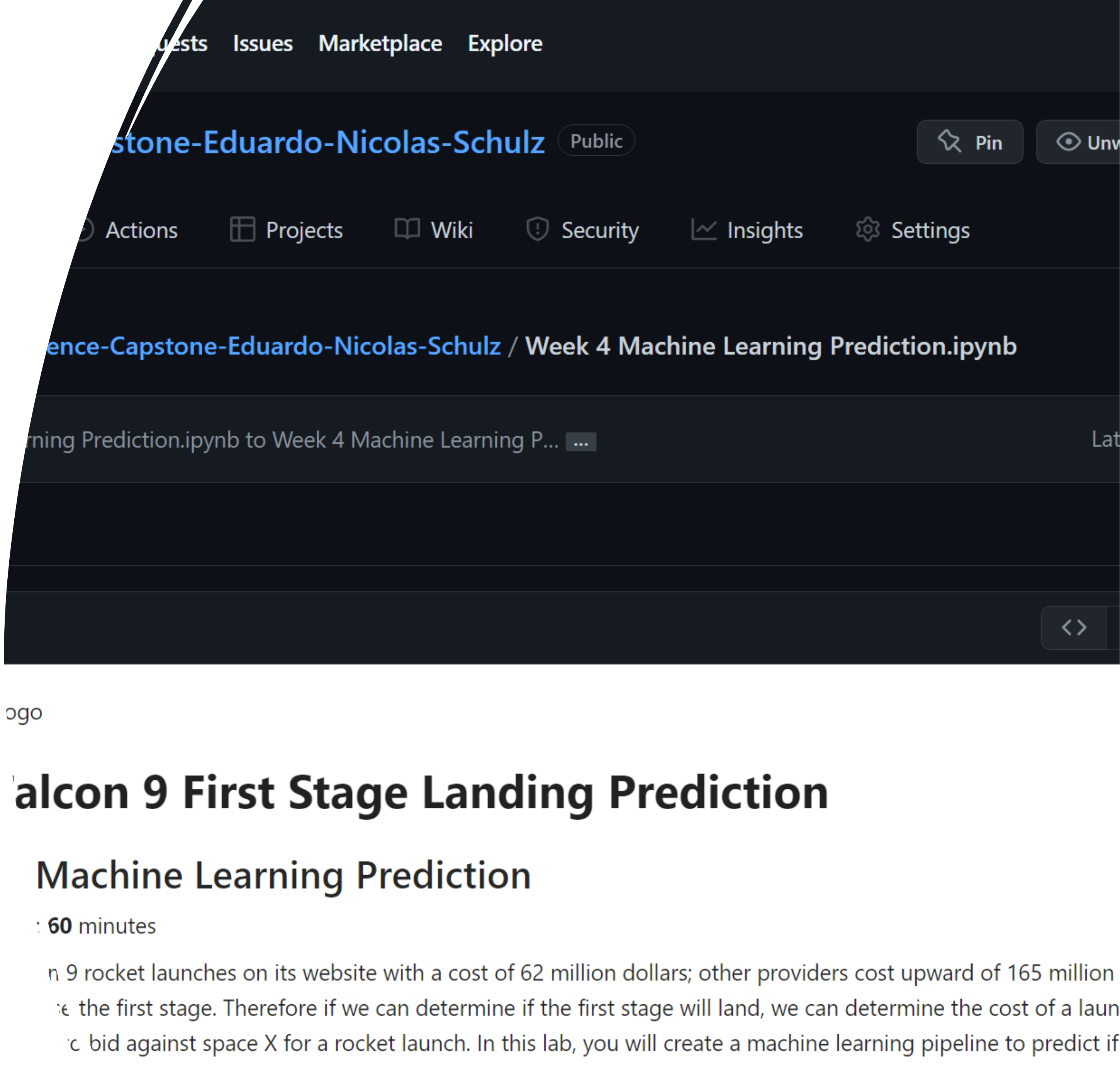
Build a Dashboard with Plotly Dash

- The plots were selected to enhance the ability to visualize the success rate data using different values for Launch Site or Payload, in order to gain a better understanding of the interplay between these variables.

Predictive Analysis (Classification)

- The GitHub URL is:

https://github.com/buyu151/Applied-Data-Science-Capstone-Eduardo-Nicolas-Schulz/blob/master/Week%204%20Machine%20Learning%20Prediction.ipynb



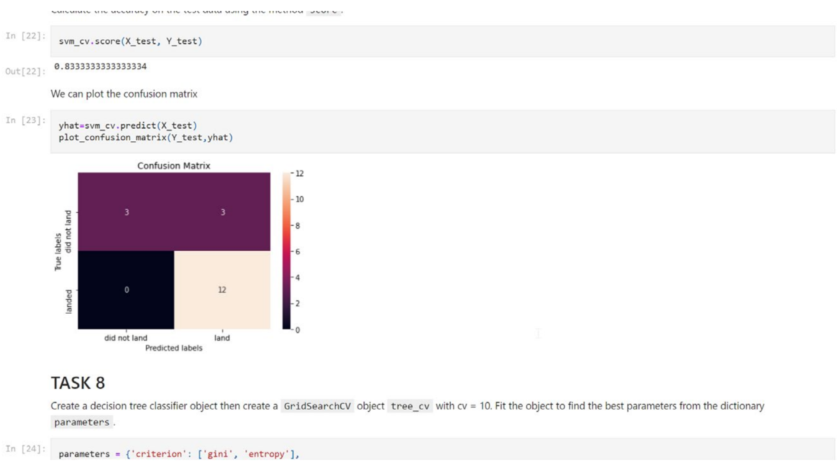
Predictive Analysis (Classification)

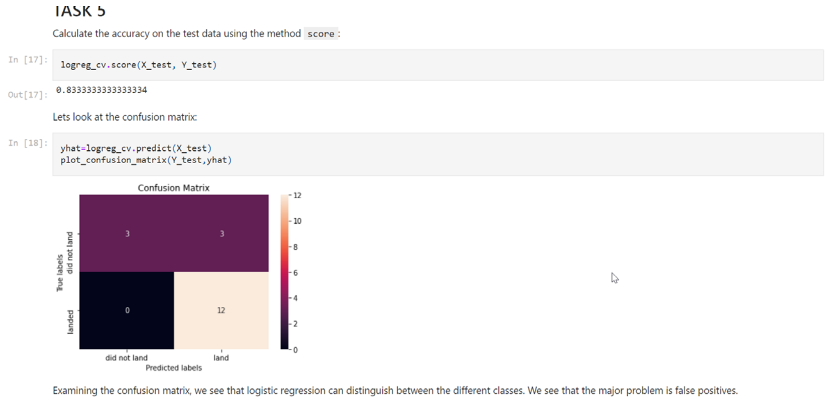
- The objective was to find the best hyperparameter for the following machine learning models: SVM, Classification Tree and Logistic Regression.
- First the data was split into a train and test set suing the train_test_split() attribute:

```
X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size=0.2, random_state=2)
```

Predictive Analysis (Classification)

- Then the models were trained.
- Following this, the test data set was fitted with the trained model and the accuracy of the fitted model was tested against the y data test set.
- Finally, confusion matrices were plotted to asse the effectiveness of each model.

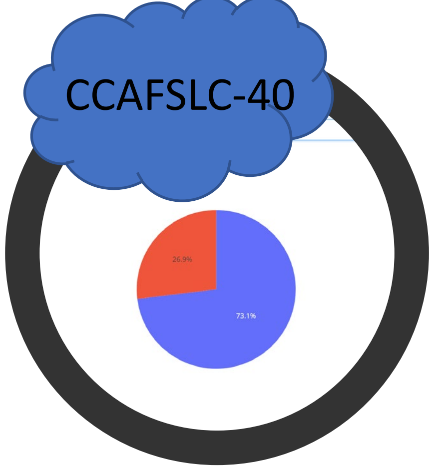


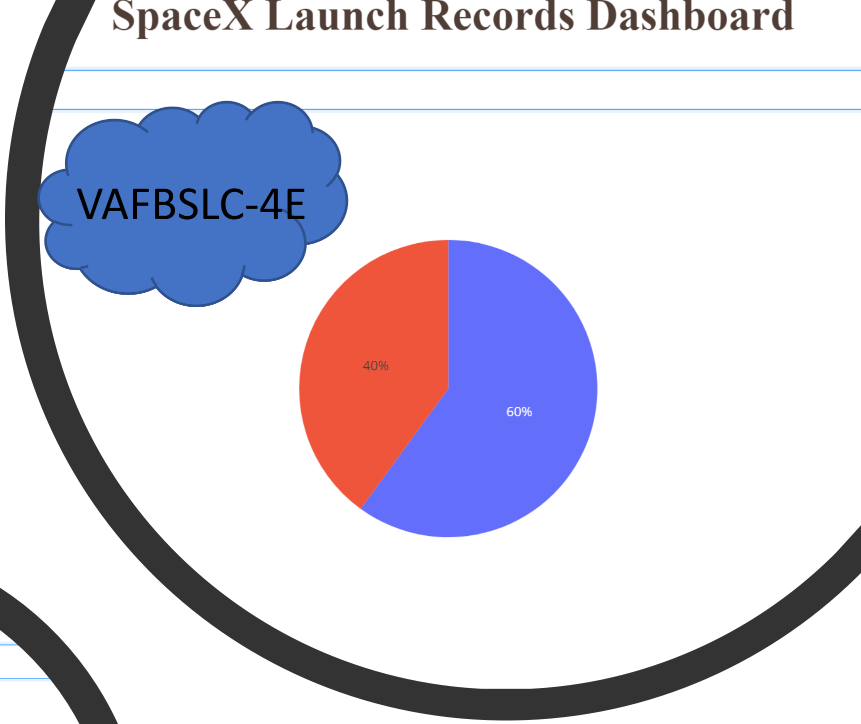


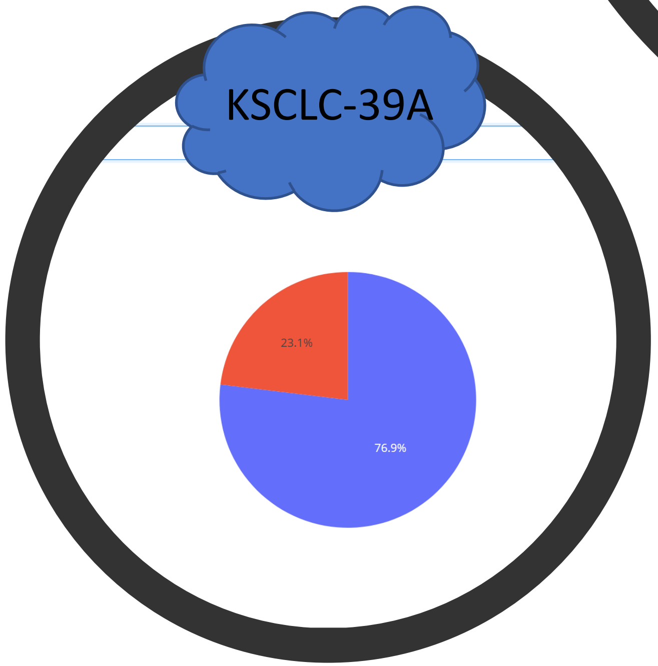
Results

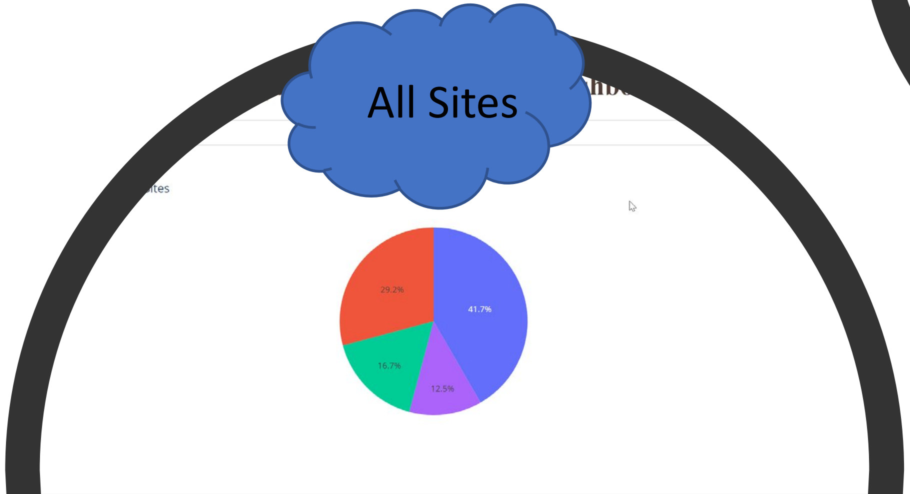
- Exploratory data analysis results
- The EDA showed interplay between success rates and some variables, such as Payload, Orbit type and Launch site location. This will be discussed further in the following slides.
- Predictive analysis results
- The ML learning models showed different levels of accuracy, with the Decision Tree Classifier being the most accurate, with an accuracy of 94% with the test data after being trained.

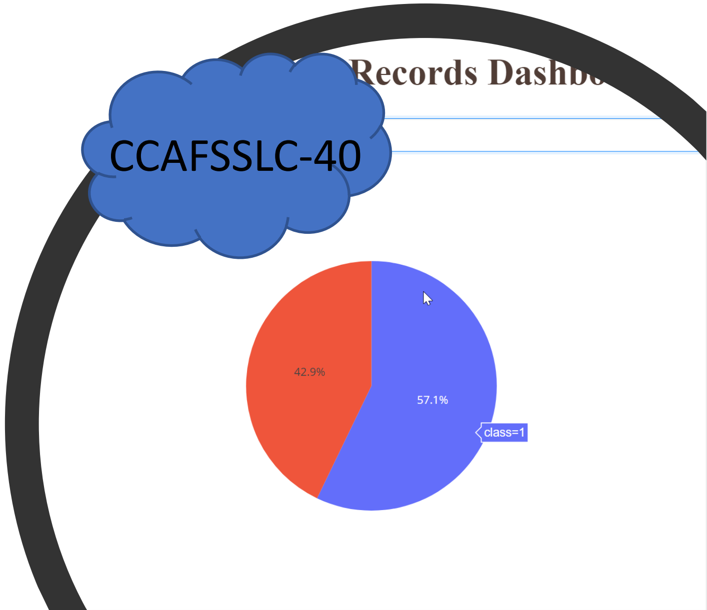
Different pie charts for each site using the Dashboard











Different scatter plots for various payload ranges (All launch sites) - Dashboard

Section 2

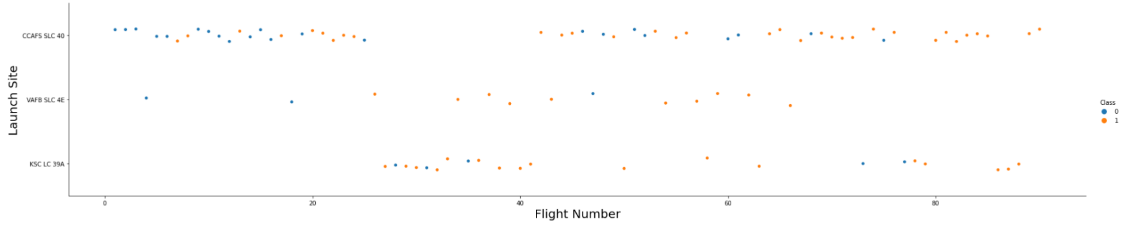
Insights drawn from EDA

Flight Number vs. Launch Site

- Scatter plot of Flight Number vs. Launch Site

- We observe that CCAFS SLC 40 Launch site has the highest success over fail ratio over the range of flight numbers.

```
# 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()
```

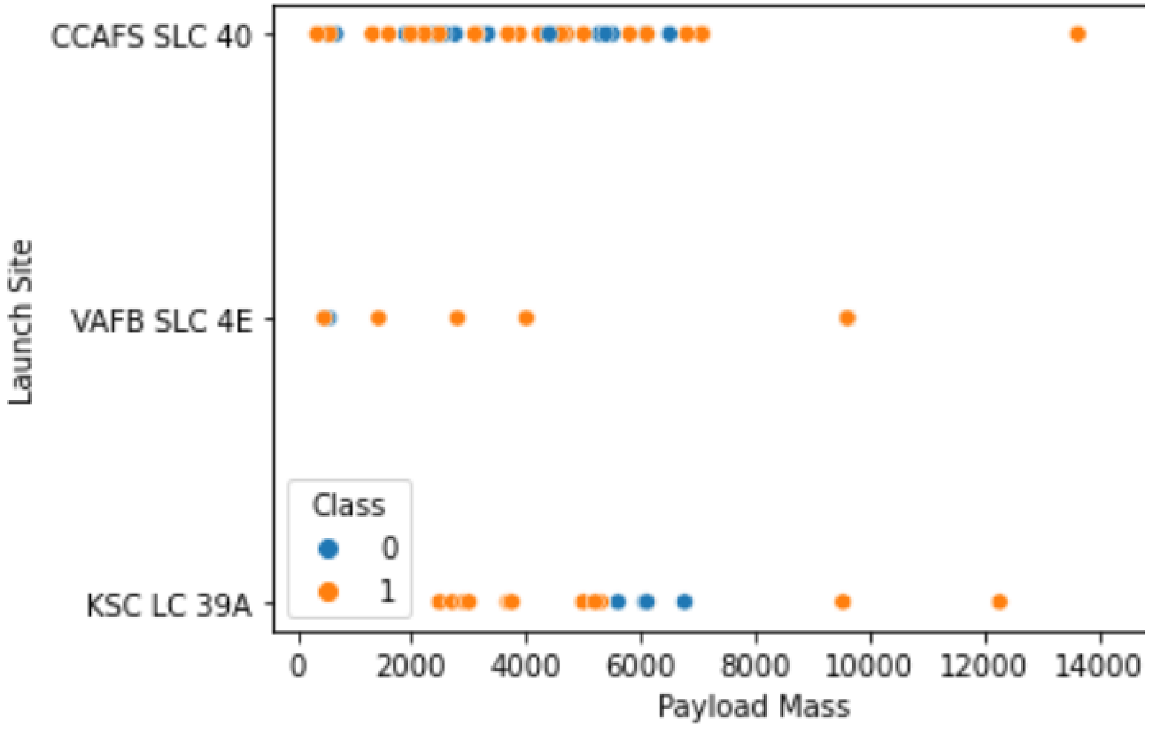


Payload vs. Launch Site

- Scatter plot of Payload vs. Launch Site

- For the VAFB-SLC launch site there are no rockets launched for heavy payload mass(greater than 10000 Kg).
- This site has the highest success vs fail ratio, suggesting the payload has a strong influence on the success of the landing.

```
# Plot a scatter point chart with x axis to be Pay L
e
sns.scatterplot(data=df, x="PayloadMass", y="LaunchS
plt.xlabel("Payload Mass",fontsize=10)
plt.ylabel("Launch Site",fontsize=10)
plt.show()
```

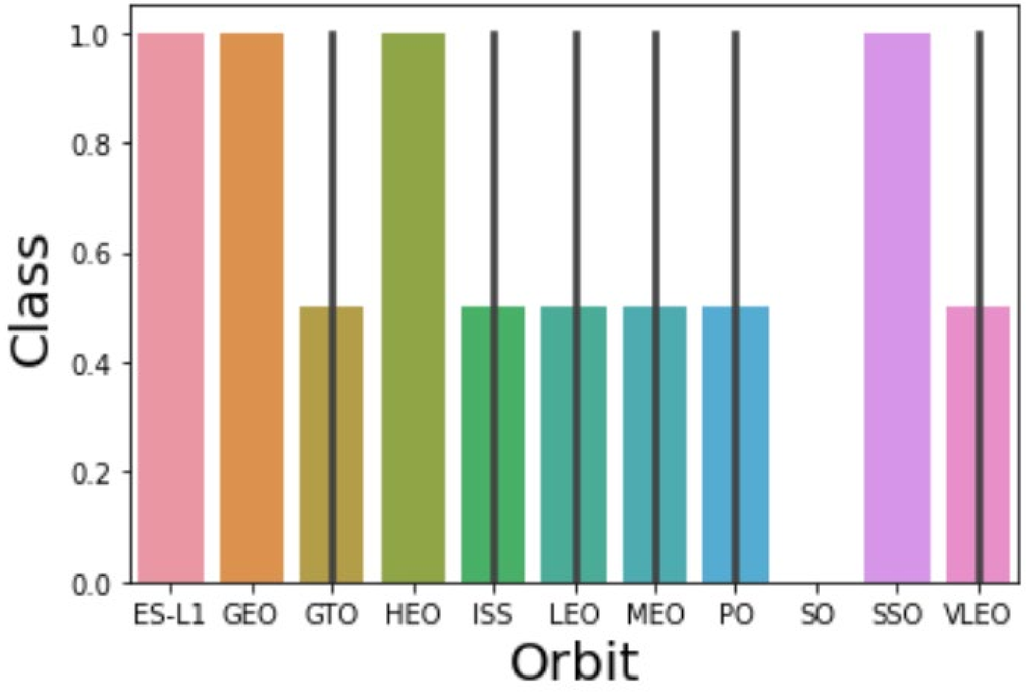


Success Rate vs. Orbit Type

- Bar chart for the success rate of each orbit type

- We can see that ES-L1, GEO, HEO and SSO orbits have the highest success ratios.

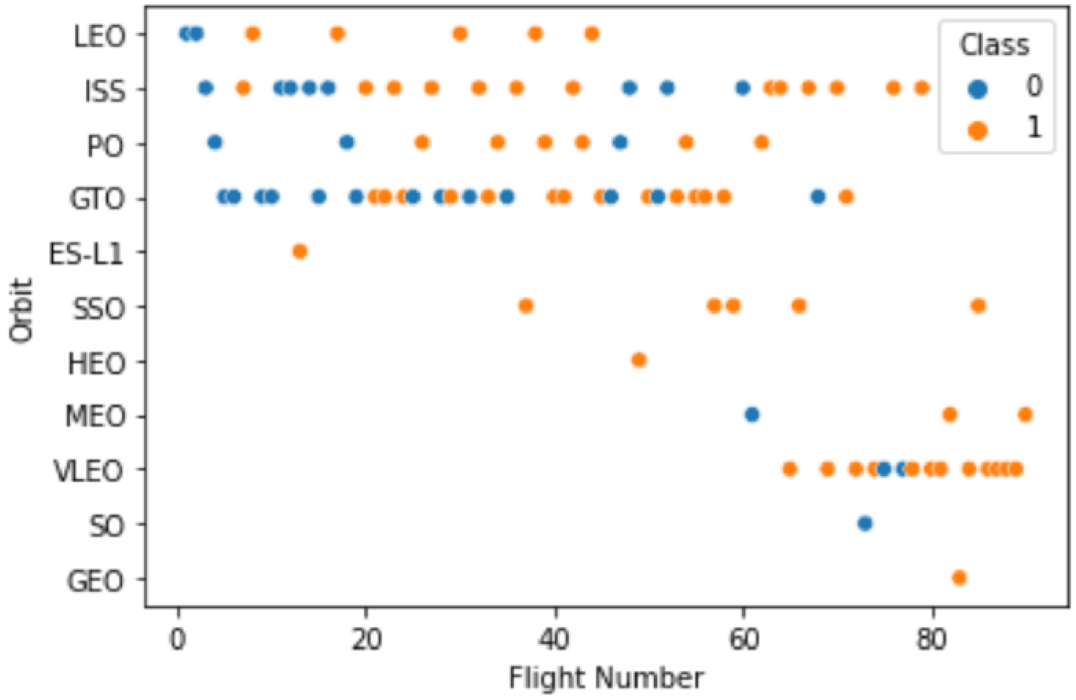
```
sns.barplot(y="Class", x="Orbit", data=t)

plt.xlabel("Orbit",fontsize=20)
plt.ylabel("Class",fontsize=20)
plt.show()
```



Flight Number vs. Orbit Type

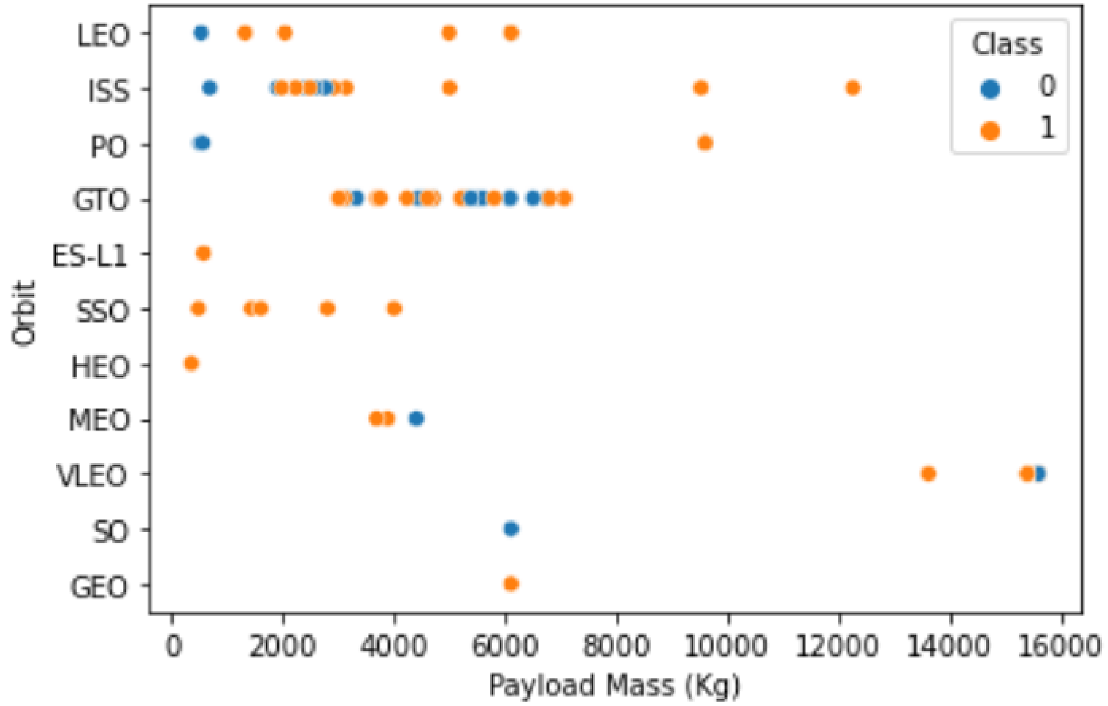
- Show a scatter point of Flight number vs. Orbit type
- Show the screenshot of the scatter plot with explanations

```
sns.scatterplot(data=df, x="FlightNumber", y="Orbit", hue="Class")
plt.xlabel("Flight Number",fontsize=10)
plt.ylabel("Orbit",fontsize=10)
plt.show()
```



Payload vs. Orbit Type

```
sns.scatterplot(data=df, x="PayloadMass", y="Orbit", hue="Class")
plt.xlabel("Payload Mass (Kg)",fontsize=10)
plt.ylabel("Orbit",fontsize=10)
plt.show()
```



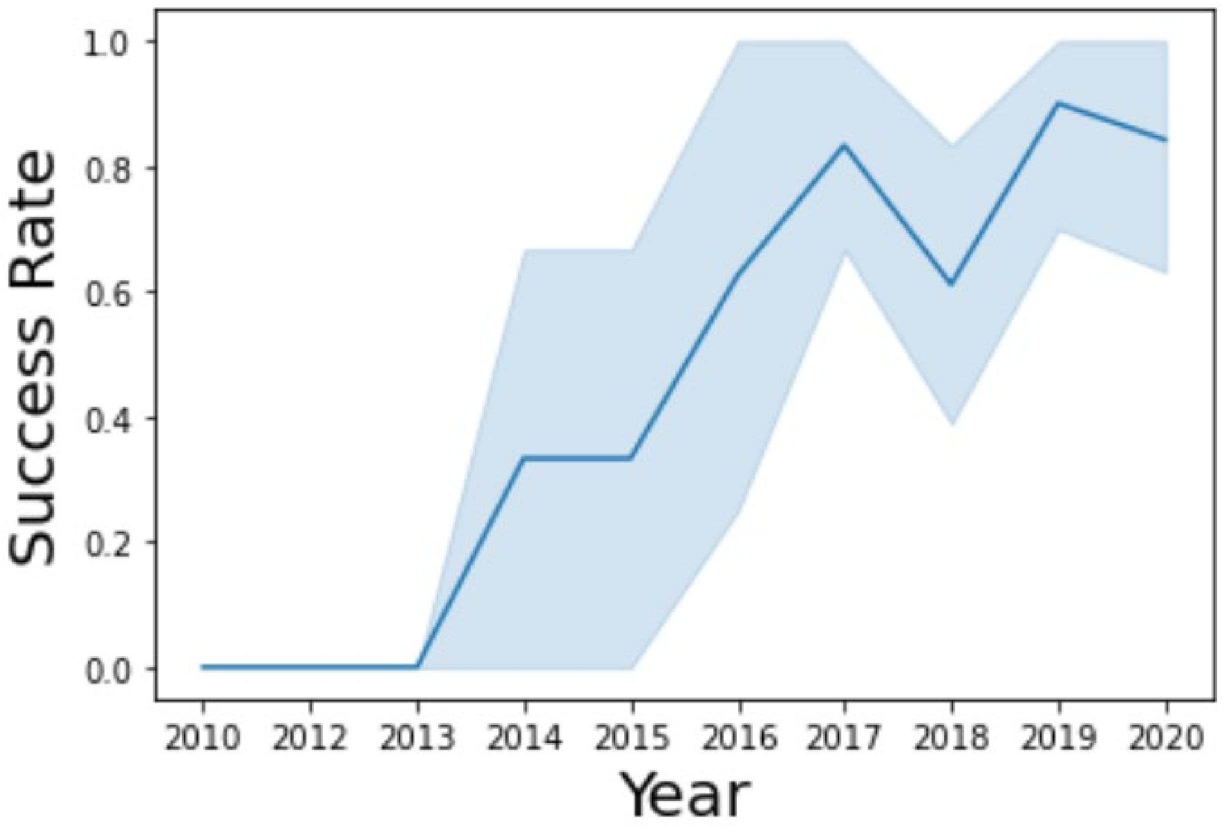
- Scatter point of payload vs. orbit type
- With heavy payloads the successful landing or positive landing rate are more for Polar, LEO and ISS.
- However, for GTO we cannot distinguish this well as both positive landing rate and negative landing(unsuccessful mission) are both there here.

Launch Success Yearly Trend

- Line chart of yearly average success rate

- You can observe that the success rate since 2013 kept increasing till 2020

```
sns.lineplot(data=df1, x="Year", y="Class")
plt.xlabel("Year",fontsize=20)
plt.ylabel("Success Rate",fontsize=20)
plt.show()
```



All Launch Site Names

- SQL query to find all the launch site names.

Display the names of the unique launch sites in the space mission

```
%sql SELECT DISTINCT(LAUNCH_SITE) FROM SPACEXDATASET
```

```
 * ibm_db_sa://vqy96979:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/bludb
Done.
```

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

Launch Site Names Begin with 'CCA'

- SQL query to find 5 records where launch sites begin with `CCA`

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

```
%sql SELECT * FROM SPACEXDATASET WHERE LAUNCH_SITE LIKE 'CCA%' LIMIT 5
```

```
 * ibm_db_sa://vqy96979:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/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

Total Payload Mass

- SQL query to calculate the total payload carried by boosters from NASA.

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

```
%sql SELECT SUM(PAYLOAD_MASS__KG_) FROM SPACEXDATASET WHERE CUSTOMER = 'NASA (CRS)'
```

```
 * ibm_db_sa://vqy96979:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/bludb
Done.
```

1
45596

Average Payload Mass by F9 v1.1

- SQL query tocalculate the average payload mass carried by booster version F9 v1.1.

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

```
%sql SELECT AVG(PAYLOAD_MASS__KG_) FROM SPACEXDATASET WHERE BOOSTER_VERSION = 'F9 v1.1'
```

```
 * ibm_db_sa://vqy96979:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/bludb
Done.
```

1
2928

First Successful Ground Landing Date

- SQL query to find the dates of the first successful landing outcome on ground pad.

List the date when the first successful landing outcome in ground pad was acheived.

Hint:Use min function

```
%sql SELECT MIN(DATE) FROM SPACEXDATASET WHERE Landing__Outcome = 'Success (ground pad)'
```

```
 * ibm_db_sa://vqy96979:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/bludb
Done.
```

1
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

- SQL query to list the names of boosters which have successfully landed on drone ship and had payload mass greater than 4000 but less than 6000.

List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000

```
%sql SELECT BOOSTER_VERSION FROM SPACEXDATASET WHERE Landing__Outcome = 'Success (drone ship)' AND PAYLOAD_MASS__KG_ > 4000 AND
PAYLOAD_MASS__KG_ < 6000
```

```
 * ibm_db_sa://vqy96979:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/bludb
Done.
```

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 to calculate the total number of successful and failure mission outcomes.

List the total number of successful and failure mission outcomes

```
%sql SELECT COUNT(MISSION_OUTCOME) FROM SPACEXDATASET WHERE MISSION_OUTCOME LIKE 'Success%' OR MISSION_OUTCOME LIKE 'Failure%'
```

```
 * ibm_db_sa://vqy96979:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/bludb
Done.
```

1
101

Boosters Carried Maximum Payload

- SQL query to list the names of the booster which have carried the maximum payload mass.

List the names of the booster_versions which have carried the maximum payload mass. Use a subquery

```
%sql SELECT BOOSTER_VERSION FROM SPACEXDATASET WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXDATASET)
```

```
 * ibm_db_sa://vqy96979:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/bludb
Done.
```

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

2015 Launch Records

- SQL query to list the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015

List the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015

```
%sql SELECT Date, Landing__Outcome, Booster_Version, Launch_Site FROM SPACEXDATASET WHERE Landing__Outcome = 'Failure (drone shi
p)' AND Date BETWEEN '2015-01-01' AND '2015-12-30'
```

```
 * ibm_db_sa://vqy96979:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/bludb
Done.
```

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

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

- SQL query 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

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

```
%sql SELECT * FROM SPACEXDATASET WHERE Landing__Outcome LIKE 'Success%' AND (DATE BETWEEN '2010-06-04' AND '2017-03-20') ORDER B
Y DATE DESC
```

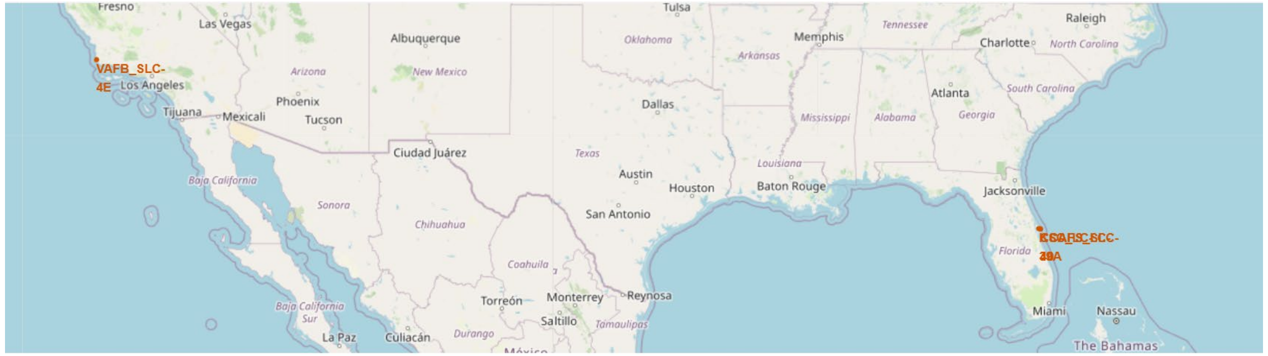
```
 * ibm_db_sa://vqy96979:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/bludb
Done.
```

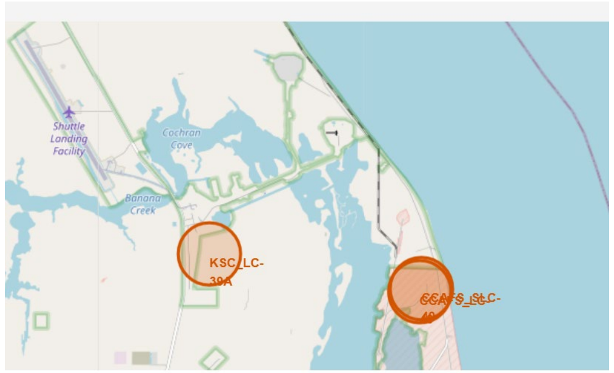
DATE	time__utc_	booster_version	launch_site	payload	payload_mass__kg_	orbit	customer	mission_outcome	landing__outcome
2017-02-19	14:39:00	F9 FT B1031.1	KSC LC-39A	SpaceX CRS-10	2490	LEO (ISS)	NASA (CRS)	Success	Success (ground pad)
2017-01-14	17:54:00	F9 FT B1029.1	VAFB SLC-4E	Iridium NEXT 1	9600	Polar LEO	Iridium Communications	Success	Success (drone ship)
2016-08-14	05:26:00	F9 FT B1026	CCAFS LC-40	JCSAT-16	4600	GTO	SKY Perfect JSAT Group	Success	Success (drone ship)
2016-07-18	04:45:00	F9 FT B1025.1	CCAFS LC-40	SpaceX CRS-9	2257	LEO (ISS)	NASA (CRS)	Success	Success (ground pad)
2016-05-27	21:39:00	F9 FT B1023.1	CCAFS LC-40	Thaicom 8	3100	GTO	Thaicom	Success	Success (drone ship)
2016-05-06	05:21:00	F9 FT B1022	CCAFS LC-40	JCSAT-14	4696	GTO	SKY Perfect JSAT Group	Success	Success (drone ship)
2016-04-08	20:43:00	F9 FT B1021.1	CCAFS LC-40	SpaceX CRS-8	3136	LEO (ISS)	NASA (CRS)	Success	Success (drone ship)
2015-12-22	01:29:00	F9 FT B1019	CCAFS LC-40	OG2 Mission 2 11 Orbcomm-OG2 satellites	2034	LEO	Orbcomm	Success	Success (ground pad)

Section 3

Launch Sites Proximities Analysis

Folium Map Screenshot of all launch sites

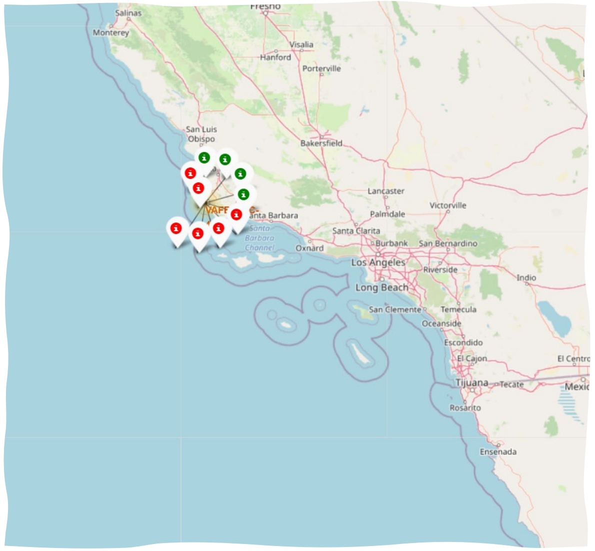


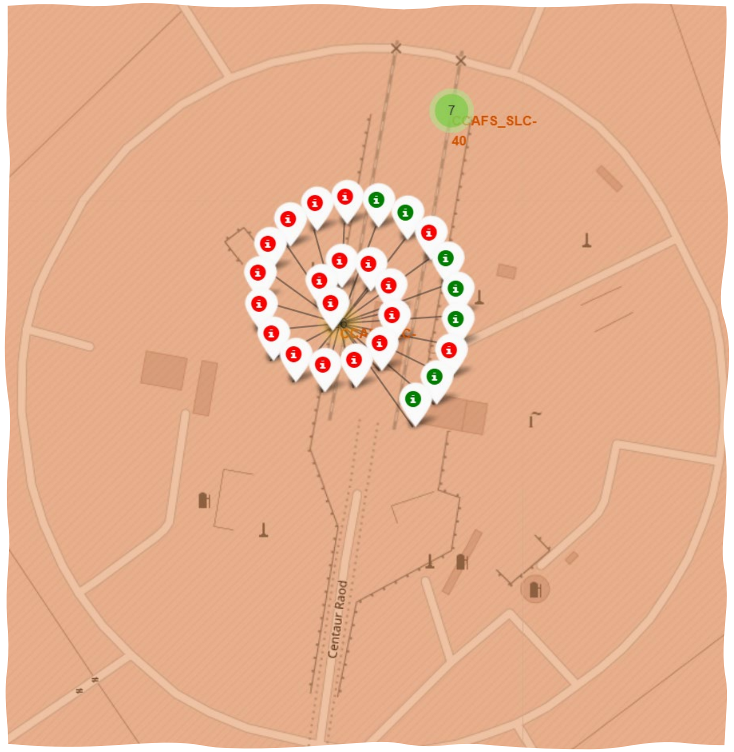


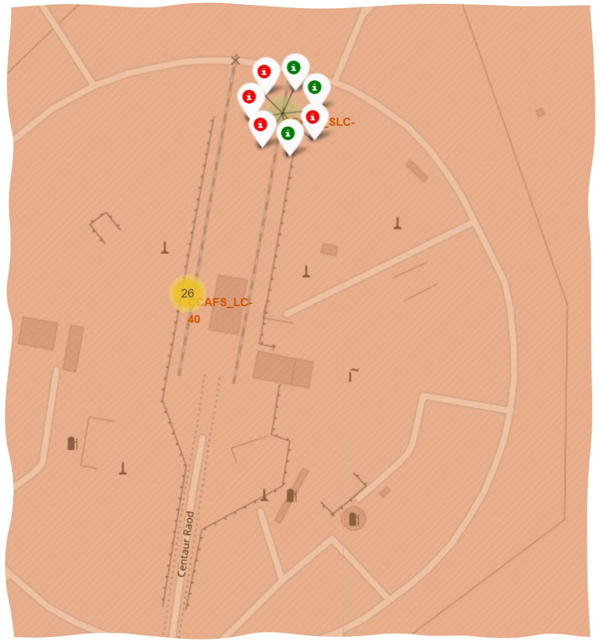
- All Space-X launch site locations in USA calculated from the latitude and longitude data.

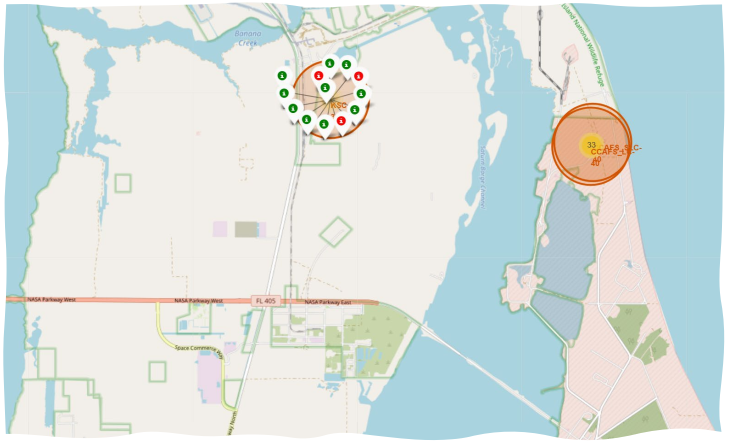
Folium Map Screenshot color-labeled launch outcomes

- Each launch site with color coded labels for each successful (green) or failed (red) landing.



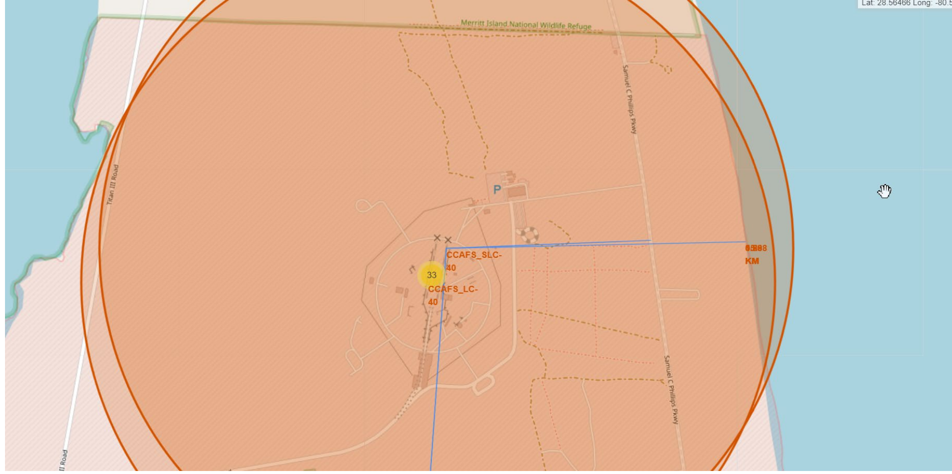


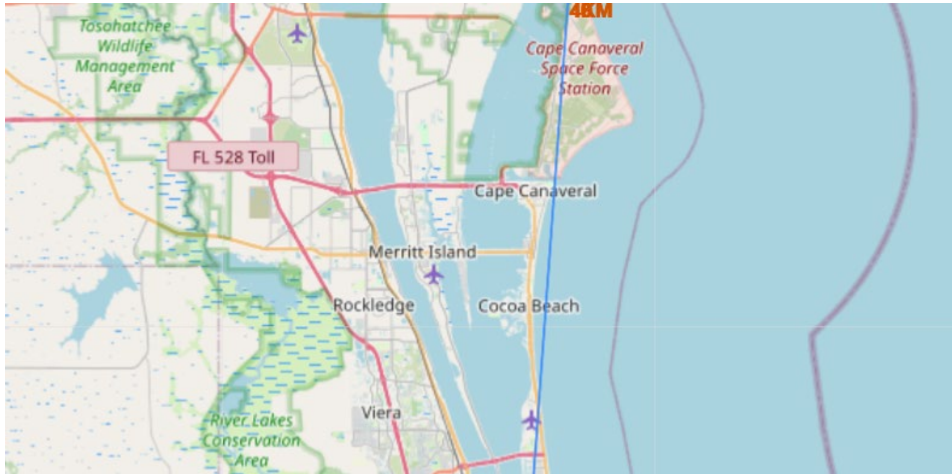




Folium Map Screenshot screenshot of a selected launch site and its proximities

- Distance to coastline (0.58 Km), motorway (0.56 Km) and nearest city (Melbourne 40 Km) for site CCAFS SLC-40



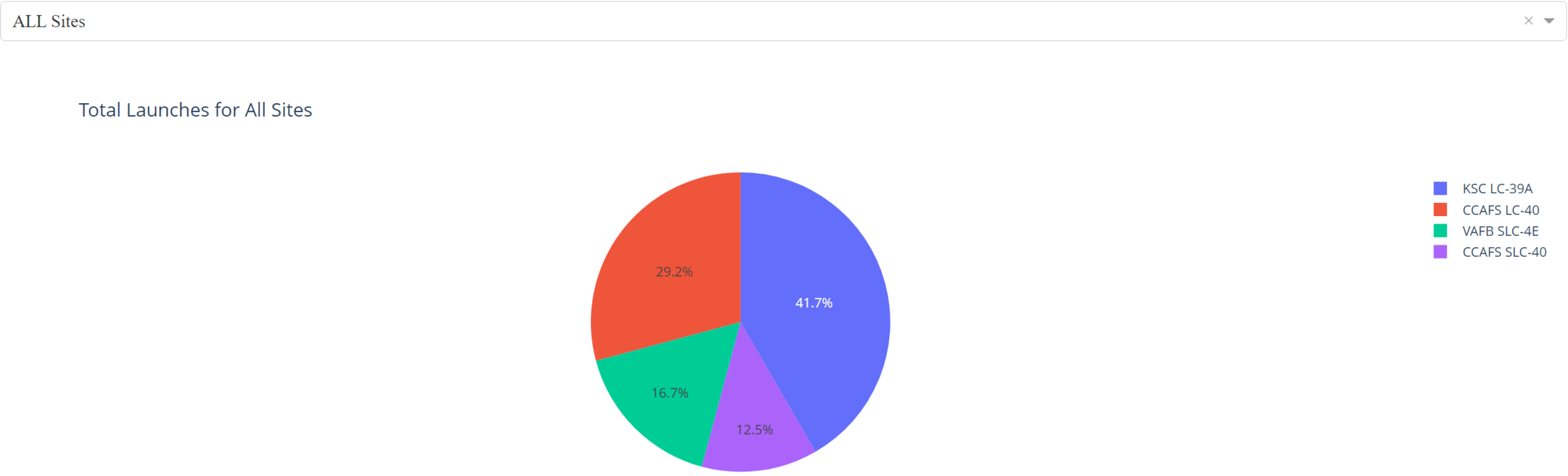


Section 4

Build a Dashboard with Plotly Dash

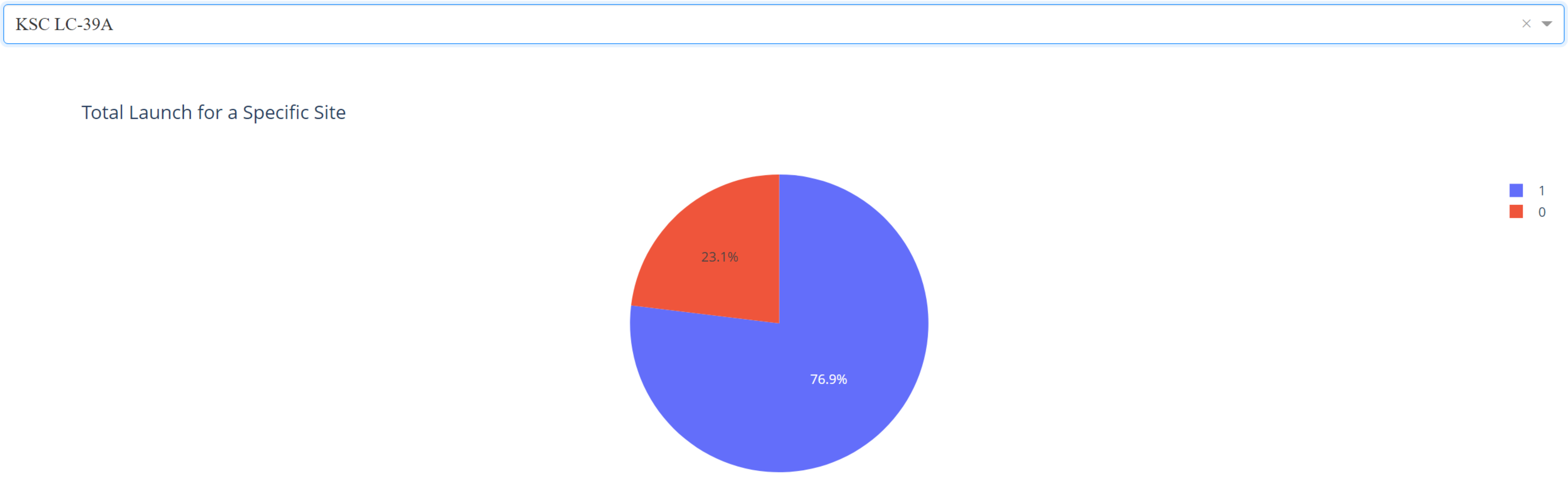
Dashboard success count for all sites, in a pie chart

- The following pie-chart shows the success rate for landing in all of the launch sites.



Dashboard piechart for the launch site with highest launch success ratio

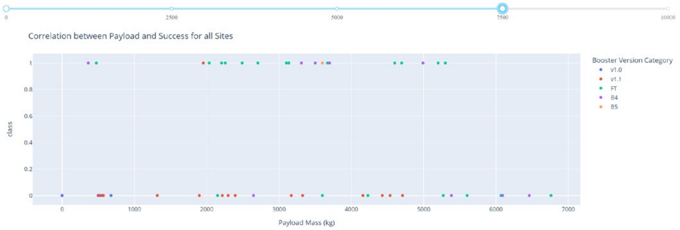
- The site KSC LC-39A shows the highest success ratio.

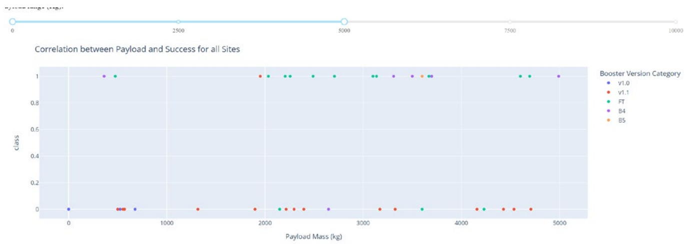


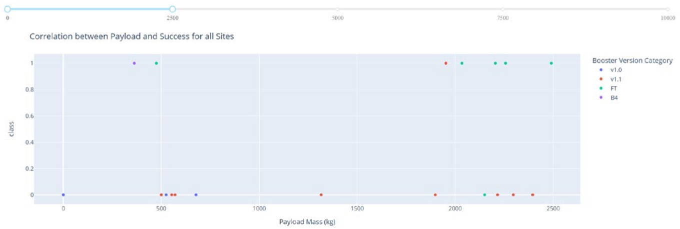
Dashboard Payload vs. Launch Outcome scatter plot for all sites

- The highest success rates seems to be for payloads between 5000 and 7500 Kg

- The Falcon rockets models FT and B4 seem to have the highest success rates overall.







Section 5

Predictive Analysis (Classification)

Classification Accuracy

- We can visualize the built model accuracy for all built classification models, in a bar chart

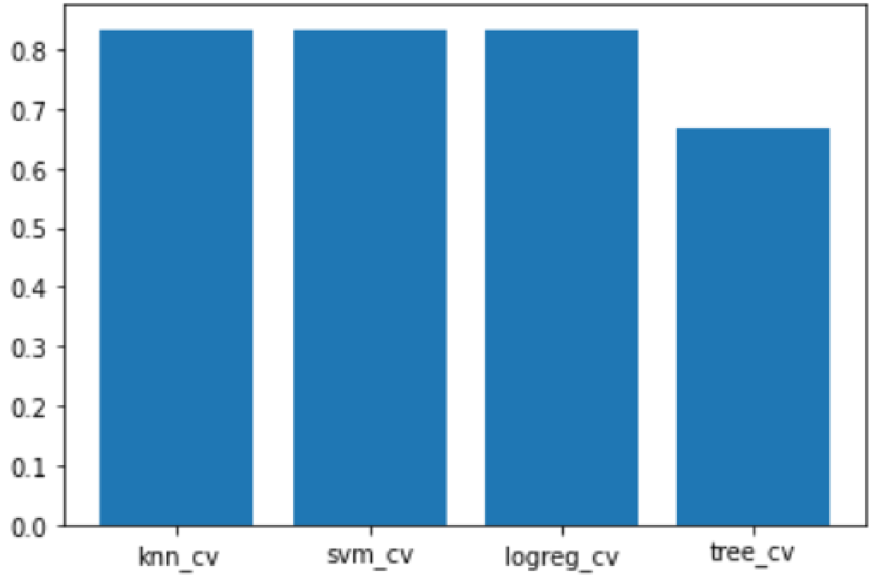
- We find that all models have a similar accuracy in the prediction, except for the Tree classifier.

- The score values can be seen below.

```
predictors = ['knn_cv', 'svm_cv', 'logreg_cv', 'tree_cv']
zipbObj = zip(predictors, results)
dictOfResults = dict(zipbObj)
keys = dictOfResults.keys()
values = dictOfResults.values()
print(keys)

plt.bar(keys, values)
```

```
dict_keys(['knn_cv', 'svm_cv', 'logreg_cv', 'tree_cv'])
```

```
2]: <BarContainer object of 4 artists>
```



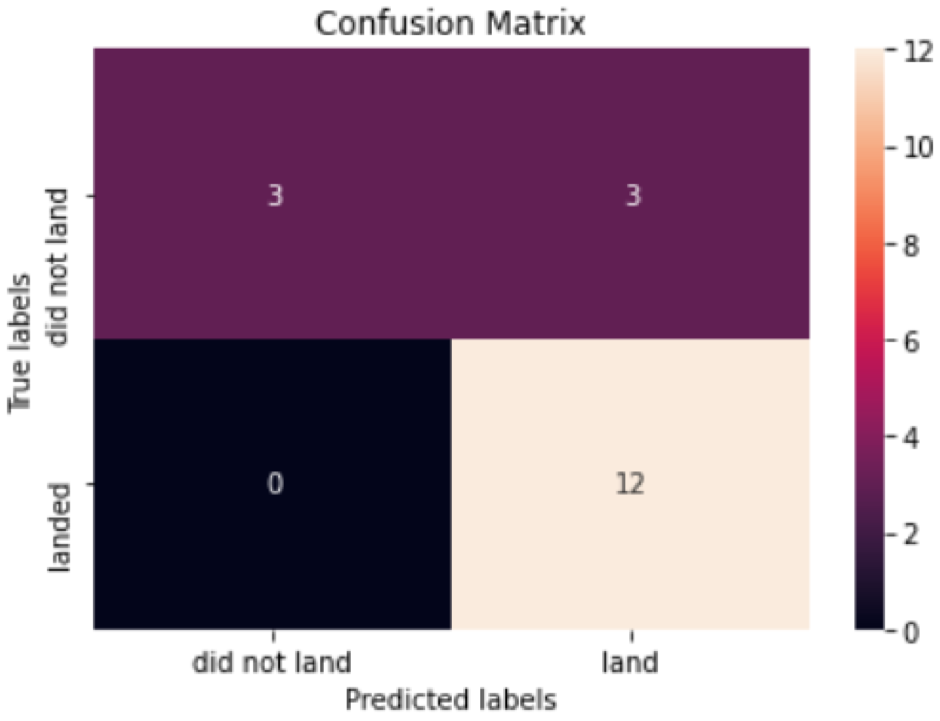

```
plt.bar(keys, values)
```

```
dict_keys(['knn_cv', 'svm_cv', 'logreg_cv', 'tree_cv']) dict_values([0.8333333333333334, 0.8333333333333334, 0.8333333333333334, 0.6666666666666666])
```

Confusion Matrix

- We see that the confusion matrix for SVM shows that the model overestimates the number of successful landings.

```
yhat = svm_cv.predict(X_test)
plot_confusion_matrix(Y_test,yhat)
```



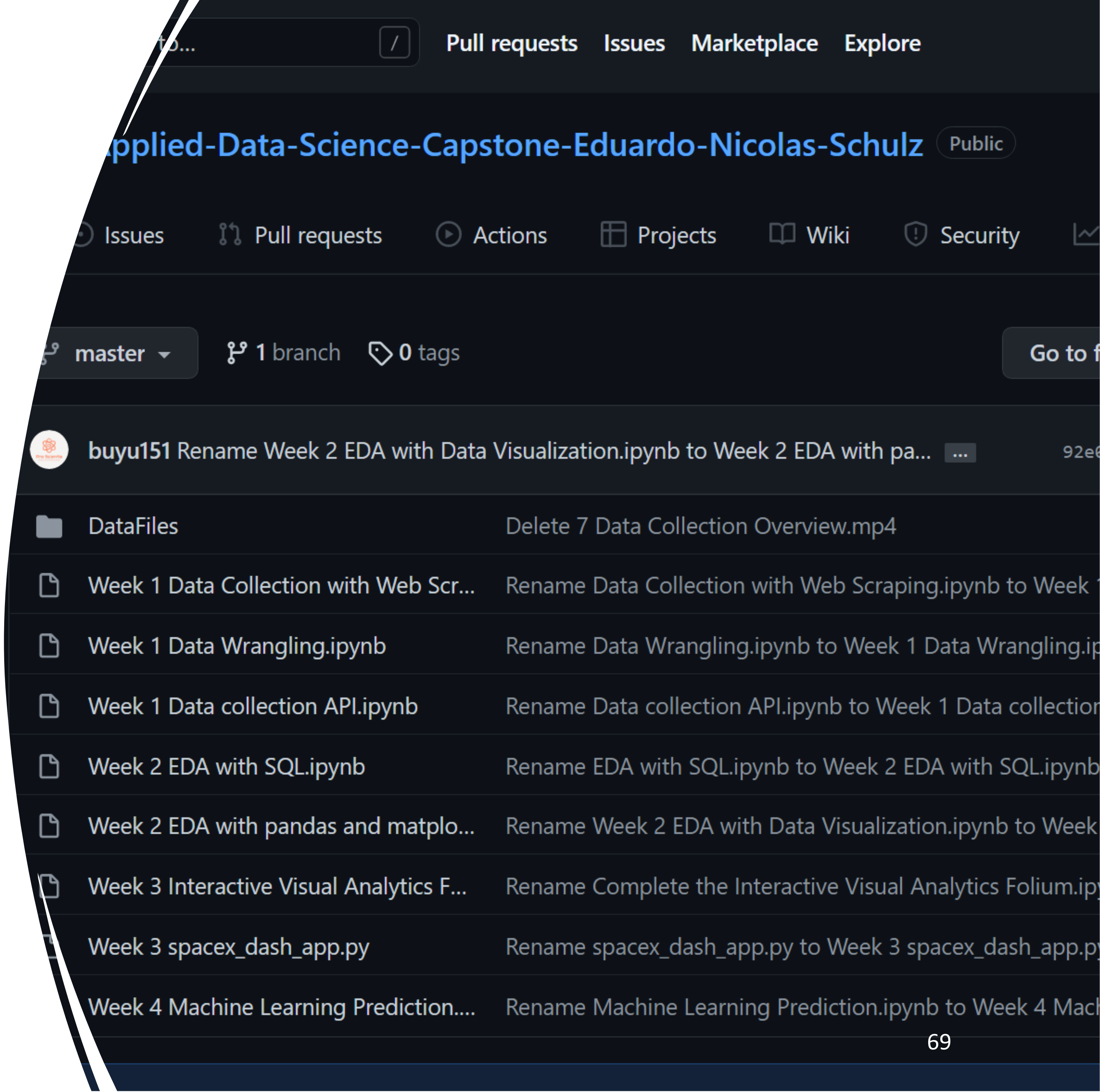
Conclusions

- EDA showed correlations between successful landings and factors such as launch site location, proximity to landmarks, payload and type of orbit achieved.
- Plotting the data showed that the success rate since 2013 kept increasing till 2020
- The KSC LC 39A site, which is the furthest from the coastline, shows the largest success rates.
- The Falcon rockets models FT and B4 seem to have the highest success rates overall.
- The highest success rates seems to be for payloads between 5000 and 7500 Kg

Appendix

- All additional tables and materials can be found in the following Github repository:

https://github.com/buyu151/Applied-Data-Science-Capstone-Eduardo-Nicolas-Schulz



Thank you!