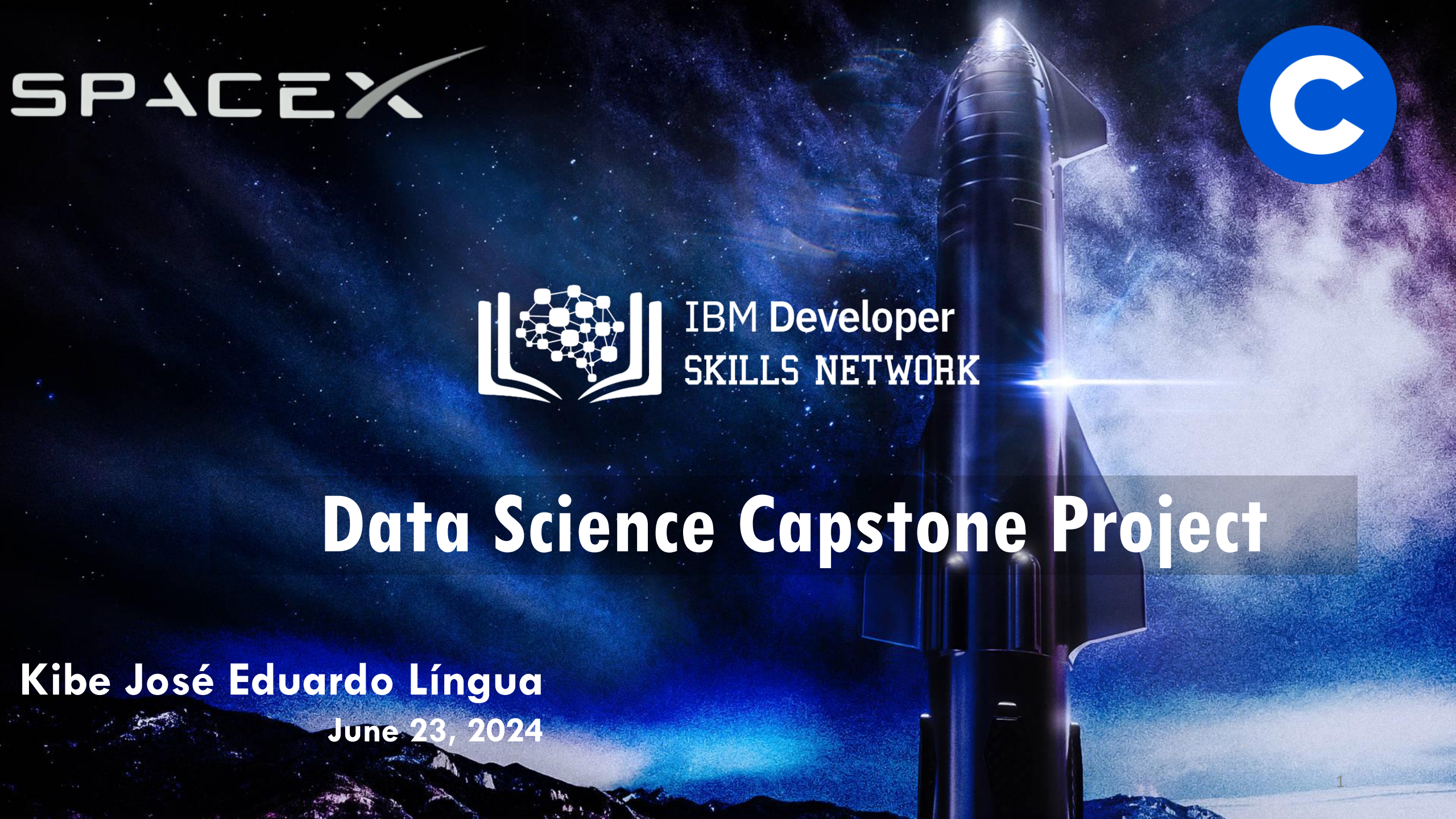


Data Science Capstone Project

Kibe José Eduardo Língua

June 23, 2024

Outline

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

Executive Summary

Summary of methodologies

- Data Collection through API
- Data Collection with Web Scraping
- Data Wrangling
- Exploratory Data Analysis with SQL
- Exploratory Data Analysis with Data Visualization
- Interactive Visual Analytics with Folium
- Machine Learning Prediction

Summary of all results

- Exploratory Data Analysis results
- Interactive analytics in screenshots
- Predictive Analytics results

Introduction

Background

- SpaceX is the most successful company of the commercial space age, making space travel affordable.
- The company 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. Based on public information and machine learning models, we are going to predict if SpaceX will reuse the first stage.

Questions to be answered

- How do variables such as payload mass, launch site, number of flights, and orbits affect the success of the first stage landing?
- Does the rate of successful landings increase over the years?
- What is the best algorithm that can be used for binary classification in this case?

Methodology

Data collection methodology:

- Using SpaceX API (https://api.spacexdata.com/v4/rockets/)
- Using Web Scrapping from Wikipedia (https://en.wikipedia.org/wiki/List_of_Falcon_9_and_Falcon_Heavy_launches)

Performed data wrangling

- Collected data was enriched by creating a landing outcome label based on outcome data after summarizing and analyzing features

Performed exploratory data analysis (EDA) using visualization and SQL

Performed interactive visual analytics using Folium and Plotly Dash

Performed predictive analysis using classification models

- Data that was collected until this step were normalized, divided in training and test data sets and evaluated by four different classification models, being the accuracy of each model evaluated using different combinations of parameters.

Data collection

- The data was collected using various methods:

  - Data collection was done using get request to the SpaceX API.

  - Next, we decoded the response content as a Json using .json() function call and turn it into a pandas dataframe using .json_normalize().

  - We then cleaned the data, checked for missing values and fill in missing values where necessary.

  - In addition, we performed web scraping from Wikipedia for Falcon 9 launch records with BeautifulSoup.

  - The objective was to extract the launch records as HTML table, parse the table and convert it to a pandas dataframe for future analysis.

Data collection – SpaceX API

1. Get request for rocket launch data using API

In [6]:
```
spacex_url="https://api.spacexdata.com/v4/launches/past"
```

In [7]:
```
response = requests.get(spacex_url)
```

2. Use json_normalize method to convert json result to dataframe

In [12]:
```
# Use json_normalize method to convert the json result into a dataframe

# decode response content as json
static_json_df = res.json()
```

In [13]:
```
# apply json_normalize
data = pd.json_normalize(static_json_df)
```

3. We then performed data cleaning and filling in the missing values

In [30]:
```
rows = data_falcon9['PayloadMass'].values.tolist()[0]

df_rows = pd.DataFrame(rows)
df_rows = df_rows.replace(np.nan, PayloadMass)

data_falcon9['PayloadMass'][0] = df_rows.values
data_falcon9
```

- **We used the get request to the SpaceX API to collect data, clean the requested data and did some basic data wrangling and formatting.**

[GitHub URL: Data Collection API]

Data collection – Web scraping

1. Apply HTTP Get method to request the Falcon 9 rocket launch page

In [4]:
```
static_url = "https://en.wikipedia.org/w/index.php?title=List_of_Falcon_9_and_Falcon_Heavy_launches&oldid=1027686922"
```

In [5]:
```
# use requests.get() method with the provided static_url
# assign the response to a object
html_data = requests.get(static_url)
html_data.status_code
```

Out[5]:
```
200
```

2. Create a BeautifulSoup object from the HTML response

In [6]:
```
# Use BeautifulSoup() to create a BeautifulSoup object from a response text content
soup = BeautifulSoup(html_data.text, 'html.parser')
```

Print the page title to verify if the `BeautifulSoup` object was created properly

In [7]:
```
# Use soup.title attribute
soup.title
```

Out[7]:
```
<title>List of Falcon 9 and Falcon Heavy launches - Wikipedia</title>
```

3. Extract all column names from the HTML table header

In [10]:
```
column_names = []

# Apply find_all() function with `th` element on first_launch_table
# Iterate each th element and apply the provided extract_column_from_header() to get a column name
# Append the Non-empty column name (`if name is not None and len(name) > 0`) into a list called column_names

element = soup.find_all('th')
for row in range(len(element)):
    try:
        name = extract_column_from_header(element[row])
        if (name is not None and len(name) > 0):
            column_names.append(name)
    except:
        pass
```

4. Create a dataframe by parsing the launch HTML tables
5. Export data to csv

- We applied web scrapping to webscrap Falcon 9 launch records with BeautifulSoup
- We parsed the table and converted it into a pandas dataframe.

GitHub URL: Data Collection with Web Scraping

Data wrangling

Steps

- **Perform EDA** and determine data labels
- **Calculate:**

  # of launches for each site

  # and occurrence of orbit

  # and occurrence of mission outcome per orbit type

- **Create binary** landing outcome column (dependent variable)
- **Export data** to csv file

Landing Outcome

- Landing was not always successful
- **True Ocean:** mission outcome had a successful landing to a specific region of the ocean

Landing Outcome (Cont.)

- **False Ocean:** represented an unsuccessful landing to a specific region of ocean
- **True RTLS:** meant the mission had a successful landing on a ground pad
- **False RTLS:** represented an unsuccessful landing on a ground pad
- **True ASDS:** meant the mission outcome had a successful landing on a drone ship
- **False ASDS:** represented an unsuccessful landing on drone ship
- **Outcomes converted** into 1 for a successful landing and 0 for an unsuccessful landing

GitHub URL: Data wrangling

EDA with SQL

- Performed SQL queries:

GitHub URL: EDA with SQL

- Displaying the names of the unique launch sites in the space mission
- Displaying 5 records where launch sites begin with the string ‘CCA'
- Displaying the total payload mass carried by boosters launched by NASA (CRS)
- Displaying average payload mass carried by booster version F9 v1.1
- Listing the date when the first successful landing outcome in ground pad was achieved
- Listing the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000
- Listing the total number of successful and failure mission outcomes
- Listing the names of the booster versions which have carried the maximum payload mass
- Listing the failed landing outcomes in drone ship, their booster versions and launch site names for the months in year 2015
- Ranking the count of landing outcomes (such as Failure (drone ship) or Success (ground pad)) between the date 2010-06-04 and 2017-03-20 in descending order

EDA with data visualization

- **Exploratory Data Analysis performed on variables Flight Number, Payload Mass, Launch Site, Orbit, Class and Year.**

Plots Used:

- **Flight Number vs. Payload Mass, Flight Number vs. Launch Site, Payload Mass vs. Launch Site, Orbit vs. Success Rate, Flight Number vs. Orbit, Payload vs Orbit, and Success Yearly Trend**

- **Scatter plots, line charts, and bar plots were used to compare relationships between variables to decide if a relationship exists so that they could be used in training the machine learning model**

GitHub URL: EDA with Data Visualization

Build an interactive map with Folium

Launch Site Markers:

- Utilized latitude and longitude coordinates to mark the locations of all launch sites.
- Incorporated markers with circles, popup labels, and text labels, starting with NASA Johnson Space Center as the reference point.

Launch Outcome Visualization:

- Distinguished launch outcomes using colored markers (green for success, red for failure) across all launch sites.
- Employed marker clusters to identify launch sites with higher success rates based on launch outcomes.

Distance Visualization:

- Visualized distances between a launch site (e.g., KSC LC-39A) and its surroundings.
- Employed colored lines to represent distances to proximities such as railways, highways, coastlines, and closest cities.

GitHub URL: Interactive Visual Analytics with Folium

Build a Dashboard with Plotly Dash

Launch Site Selection:

- Implemented a dropdown menu to facilitate the selection of launch sites.

Success Launches Visualization:

- Introduced a pie chart to display the total count of successful launches across all sites.
- For specific launch sites, the chart depicts the distribution of success and failure counts.

Payload Mass Range Slider:

- Incorporated a slider for selecting the payload mass range.

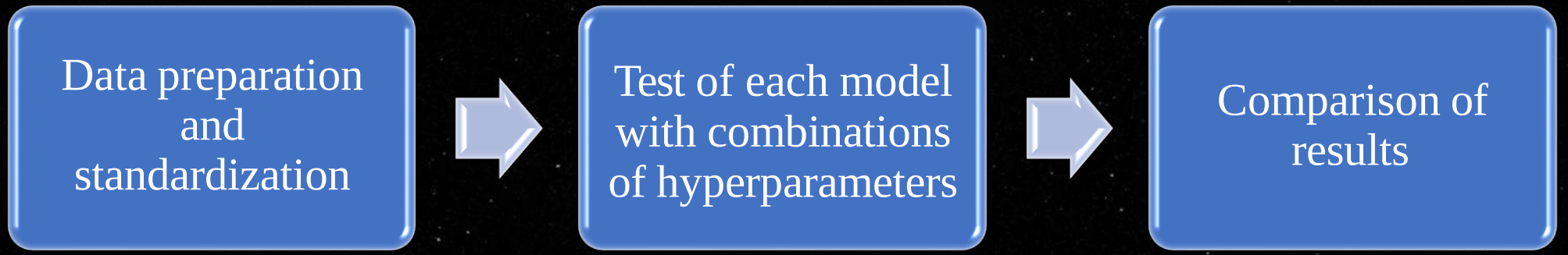
Payload vs. Success Rate Scatter Chart:

- Developed a scatter chart illustrating the relationship between payload mass and launch success rates.
- The chart categorizes data by different booster versions for comparative analysis.

GitHub URL: SpaceX Dash App

Predictive analysis (Classification)

- Four classification models were compared: logistic regression, support vector machine, decision tree and k nearest neighbors.



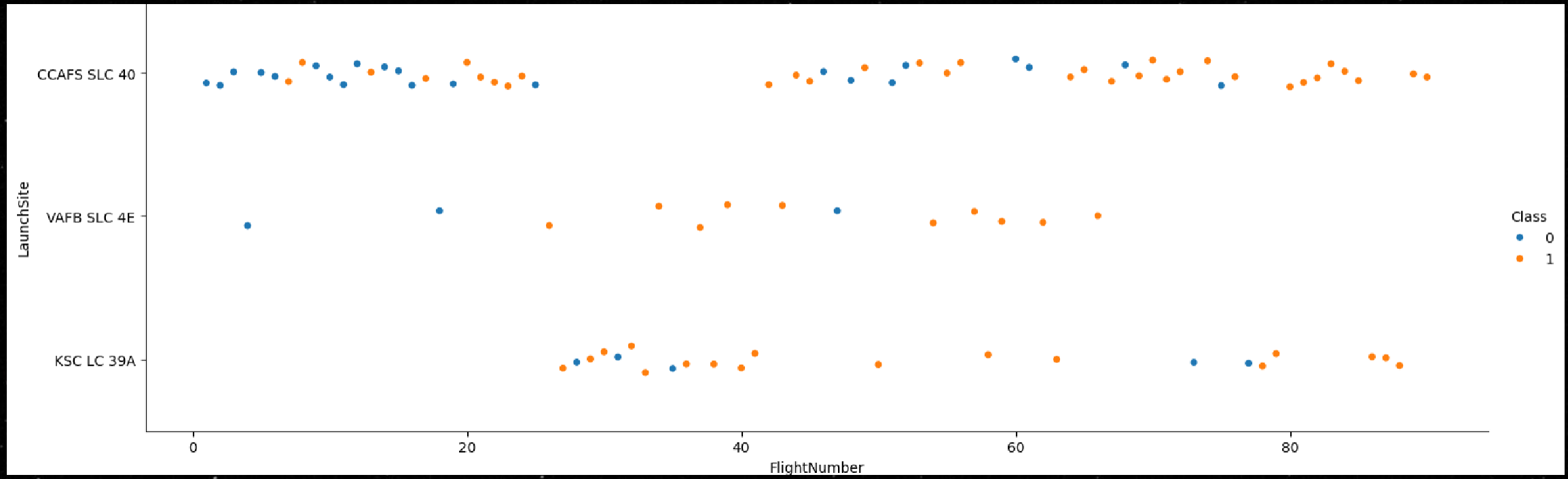
GitHub URL: Machine Learning Prediction

Results

- Exploratory data analysis results
- Interactive analytics demo in screenshots
- Predictive analysis results

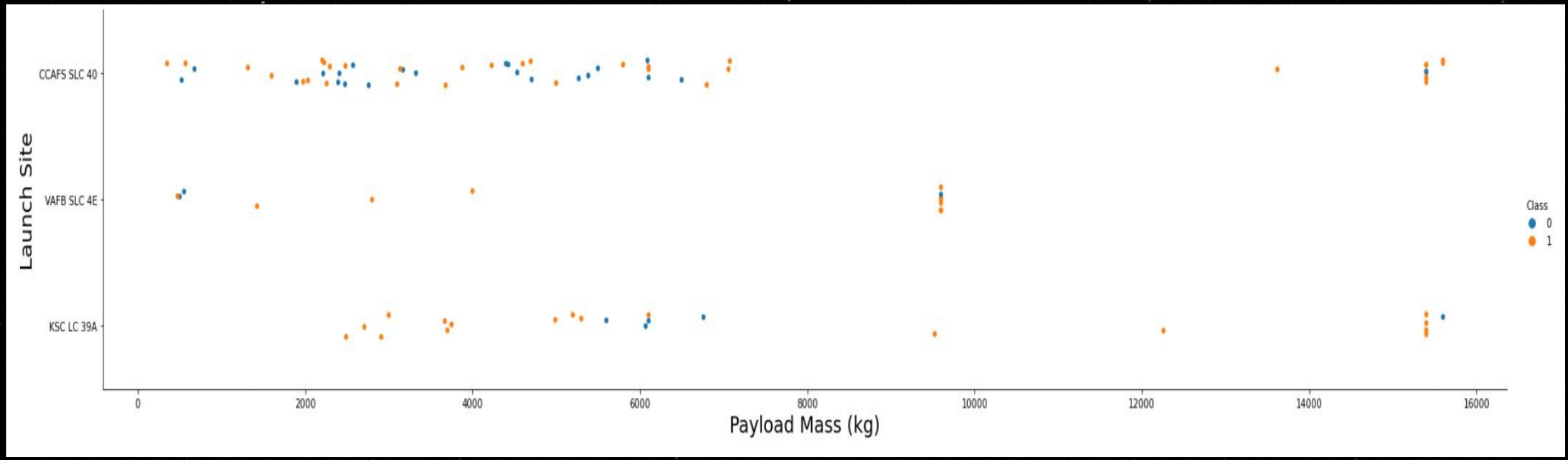
EDA with Visualization

Flight Number vs. Launch Site



- According to the plot above, it's possible to verify that the best launch site nowadays is CCAF5 SLC 40, where most of recent launches were successful;
- In second place VAFB SLC 4E and third place KSC LC 39A;
- It's also possible to see that the general success rate improved over time.

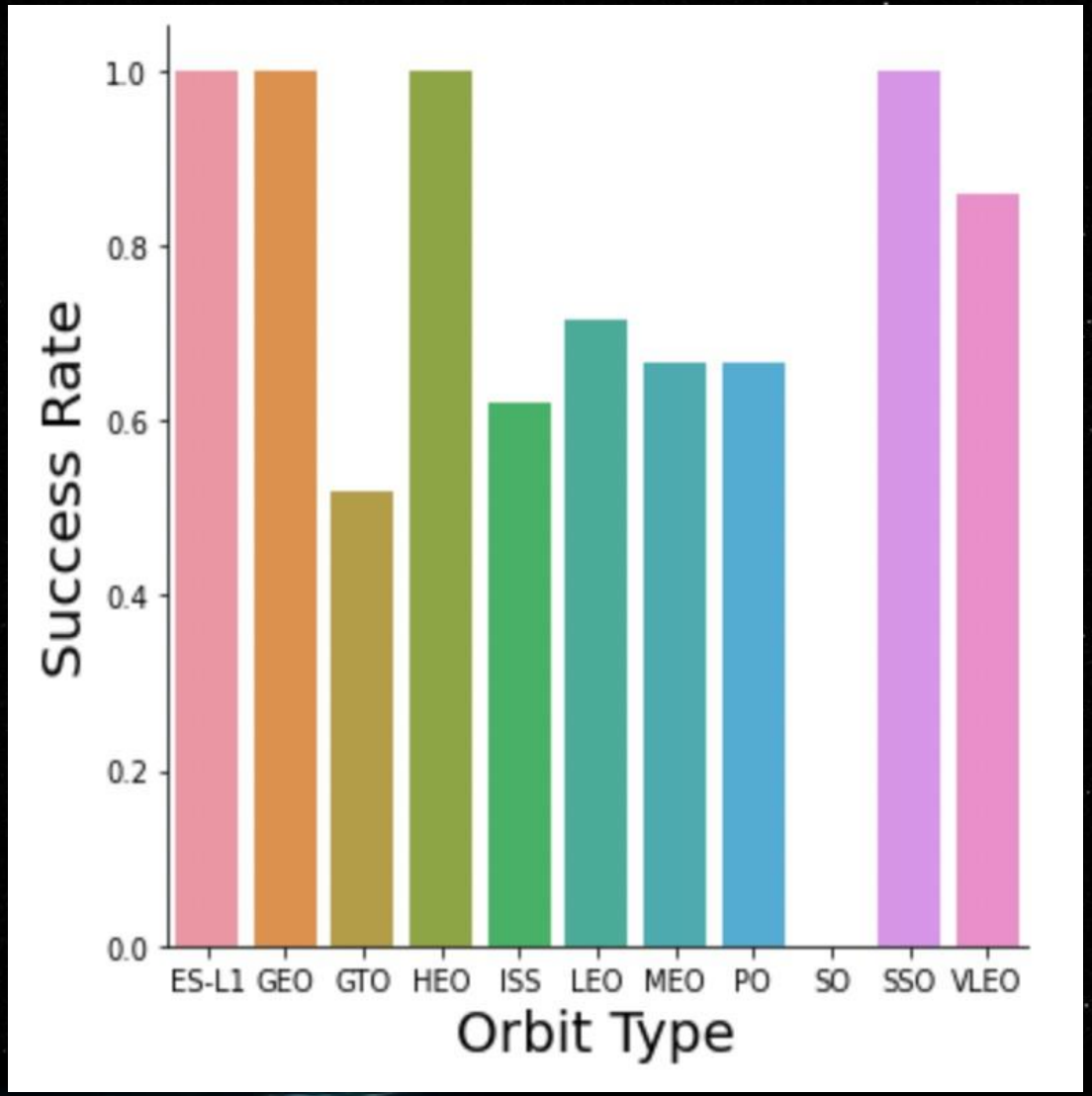
Payload vs. Launch Site



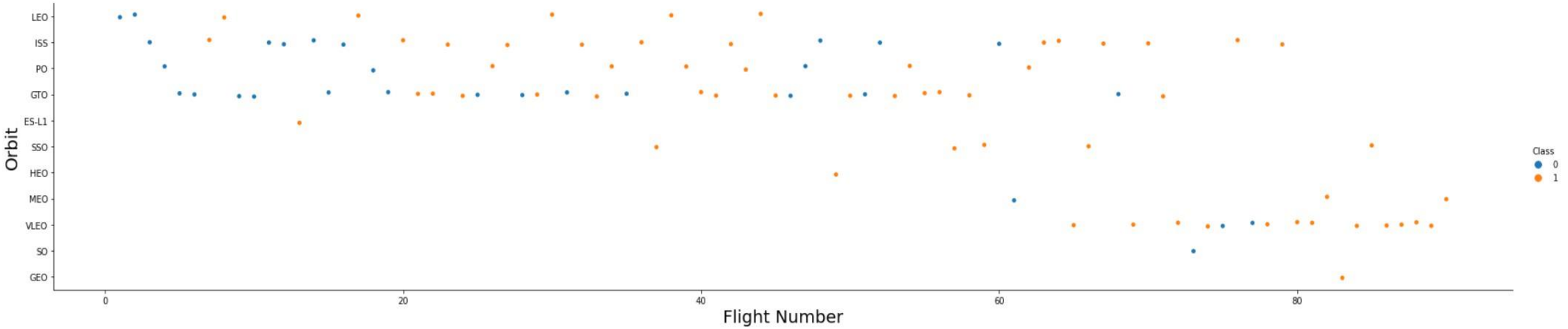
- Payloads over 9,000kg (about the weight of a school bus) have excellent success rate;
- Payloads over 12,000kg seems to be possible only on CCAFS SLC 40 and KSC LC 39A launch sites.

Success rate vs. Orbit type

- Orbits with 100% success rate:
  - ES-L1, GEO, HEO, SSO
- Orbits with 0% success rate:
  - SO
- Orbits with success rate between 50% and 85%:
  - GTO, ISS, LEO, MEO, PO

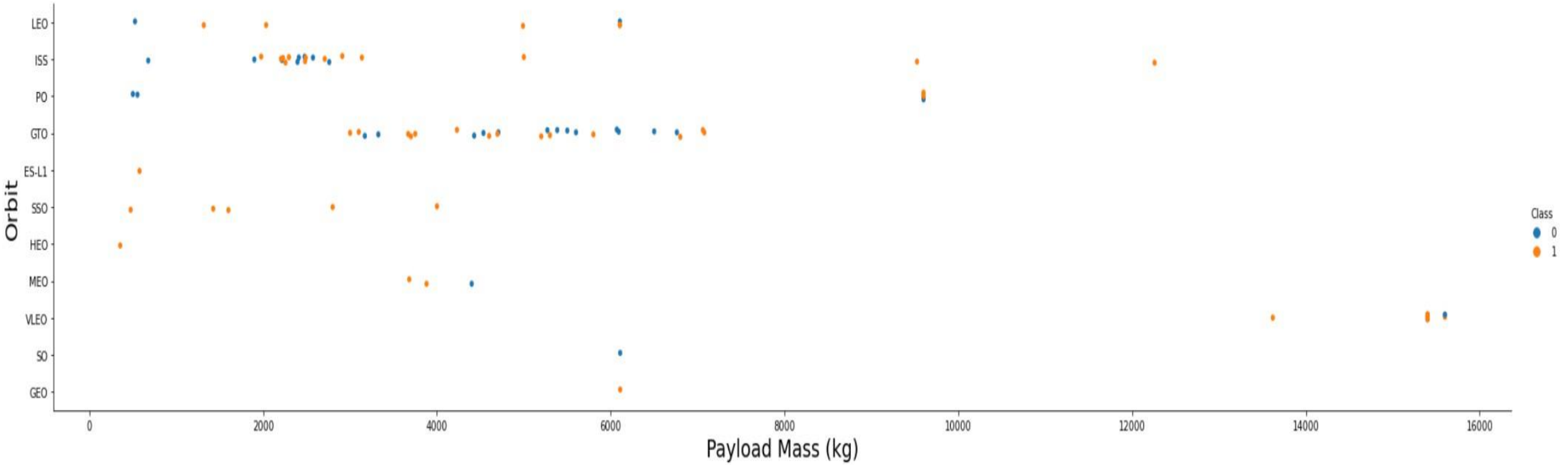


Flight Number vs. Orbit type



- Apparently, success rate improved over time to all orbits;
- VLEO orbit seems a new business opportunity, due to recent increase of its frequency.

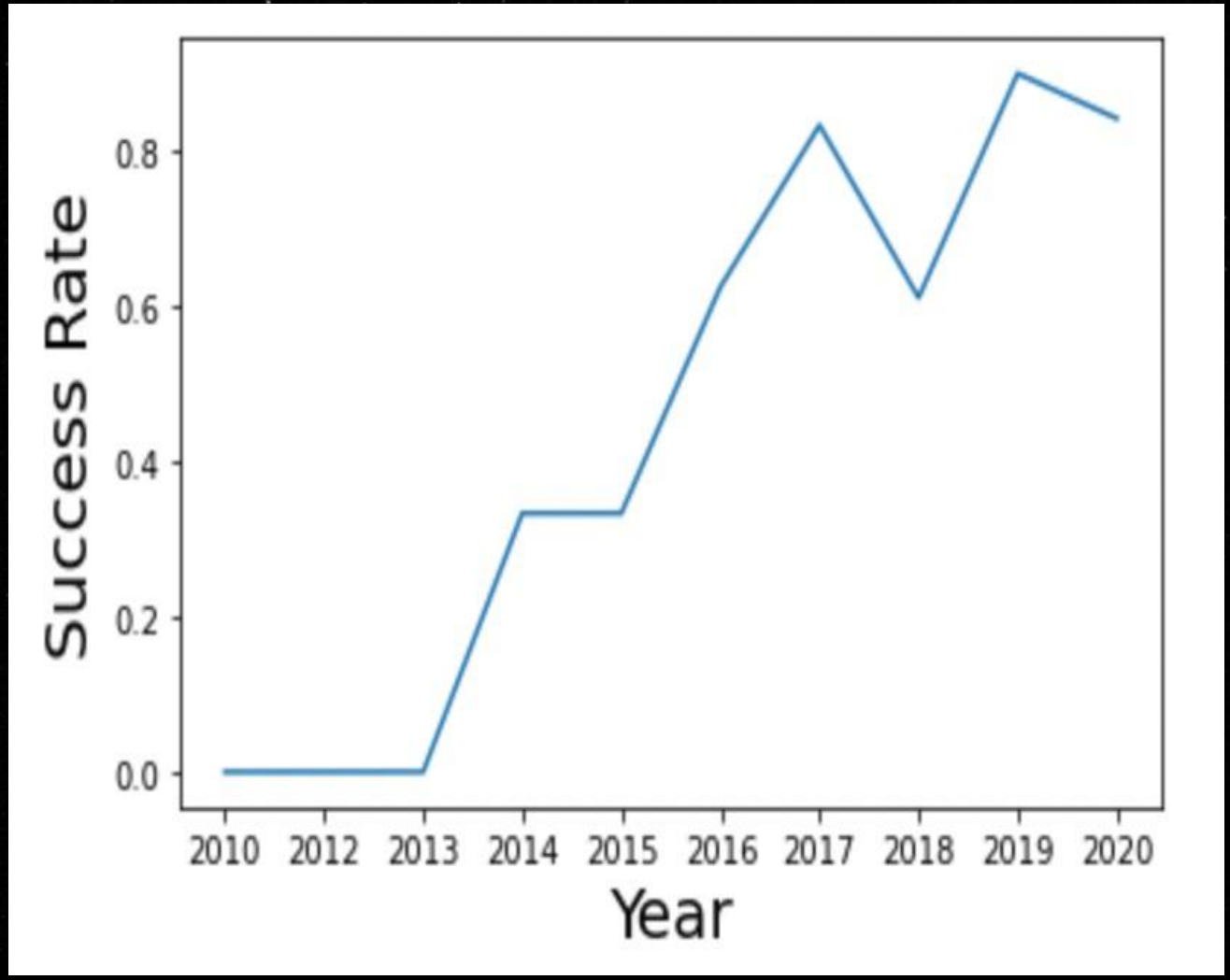
Payload Mass vs. Orbit type



- Apparently, there is no relation between payload and success rate to orbit GTO;
- ISS orbit has the widest range of payload and a good rate of success;
- There are few launches to the orbits SO and GEO.

Launch success yearly trend

- Success rate started increasing in 2013 and kept until 2020;
- It seems that the first three years were a period of adjusts and improvement of technology.



EDA with SQL

All launch site names

- Displaying the names of the unique launch sites in the space mission.

```
%%sql
SELECT DISTINCT(Launch_Site) FROM SPACEXTABLE
```

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

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

Launch site names begin with `CCA`

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

```
 * ibm_db_sa://wzf08322:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90108kqblod8lcg.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

Displaying 5 records where launch sites begin with the string 'CCA'.

Total payload mass

```
%%sql
SELECT SUM(PAYLOAD_MASS__KG_) FROM SPACEXTABLE
WHERE Customer == "NASA (CRS)"
```

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

SUM(PAYLOAD_MASS__KG_)
45596

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

Average payload mass by F9 v1.1

```
%%sql
SELECT AVG(PAYLOAD_MASS__KG_) FROM SPACEXTABLE
WHERE Booster_Version == "F9 v1.1"
```

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

AVG(PAYLOAD_MASS__KG_)
2928.4

- Filtering data by the booster version above and calculating the average payload mass we obtained the value of 2928,4 kg.

First successful ground landing date

```
%%sql
SELECT MIN(Date) FROM SPACEXTABLE
WHERE Landing_Outcome == "Success (ground pad)"
```

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

MIN(Date)
2015-12-22

- By filtering data by successful landing outcome on ground pad and getting the minimum value for date it’s possible to identify the first occurrence, that happened on 22/12/2015.

Successful drone ship landing with payload between 4000 and 6000

```
%%sql
SELECT Payload FROM SPACEXTABLE
WHERE Landing_Outcome == "Success (drone ship)"
AND PAYLOAD_MASS__KG_ BETWEEN 4000 AND 6000
```

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

Payload
JCSAT-14
JCSAT-16
SES-10
SES-11 / EchoStar 105

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

Total number of successful and failure mission outcomes

```
%%sql
SELECT COUNT(Mission_Outcome) FROM SPACEXTABLE
```

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

COUNT(Mission_Outcome)
101

- Grouping mission outcomes and counting records for each group led us to the summary above.

Mission Outcome	Occurrences
Success	99
Success (payload status unclear)	1
Failure (in flight)	1

Boosters carried maximum payload

```
%%sql
SELECT Booster_Version from SPACEXTABLE
WHERE PAYLOAD_MASS__KG_ IN (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTABLE)
```

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

- These are the boosters which have carried the maximum payload mass registered in the dataset.

2015 launch records

```
%%sql select monthname(date) as month, date, booster_version, launch_site, landing__outcome from SPACEXDATASET
      where landing__outcome = 'Failure (drone ship)' and year(date)=2015;
```

```
 * ibm_db_sa://wzf08322:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90108kqblod8lcg.databases.appdomain.cloud:31198/bludb
Done.
```

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

- Failed landing outcomes in drone ship, their booster versions, and launch site names for in year 2015
- The list above has the only two occurrences.

Rank success count between 2010-06-04 and 2017-03-20

```
%sql SELECT Landing_Outcome, COUNT(*) AS COUNT_LAUNCHES FROM SPACEXTABLE WHERE DATE BETWEEN '2010-06-04' AND '2017-03-20' GR
```

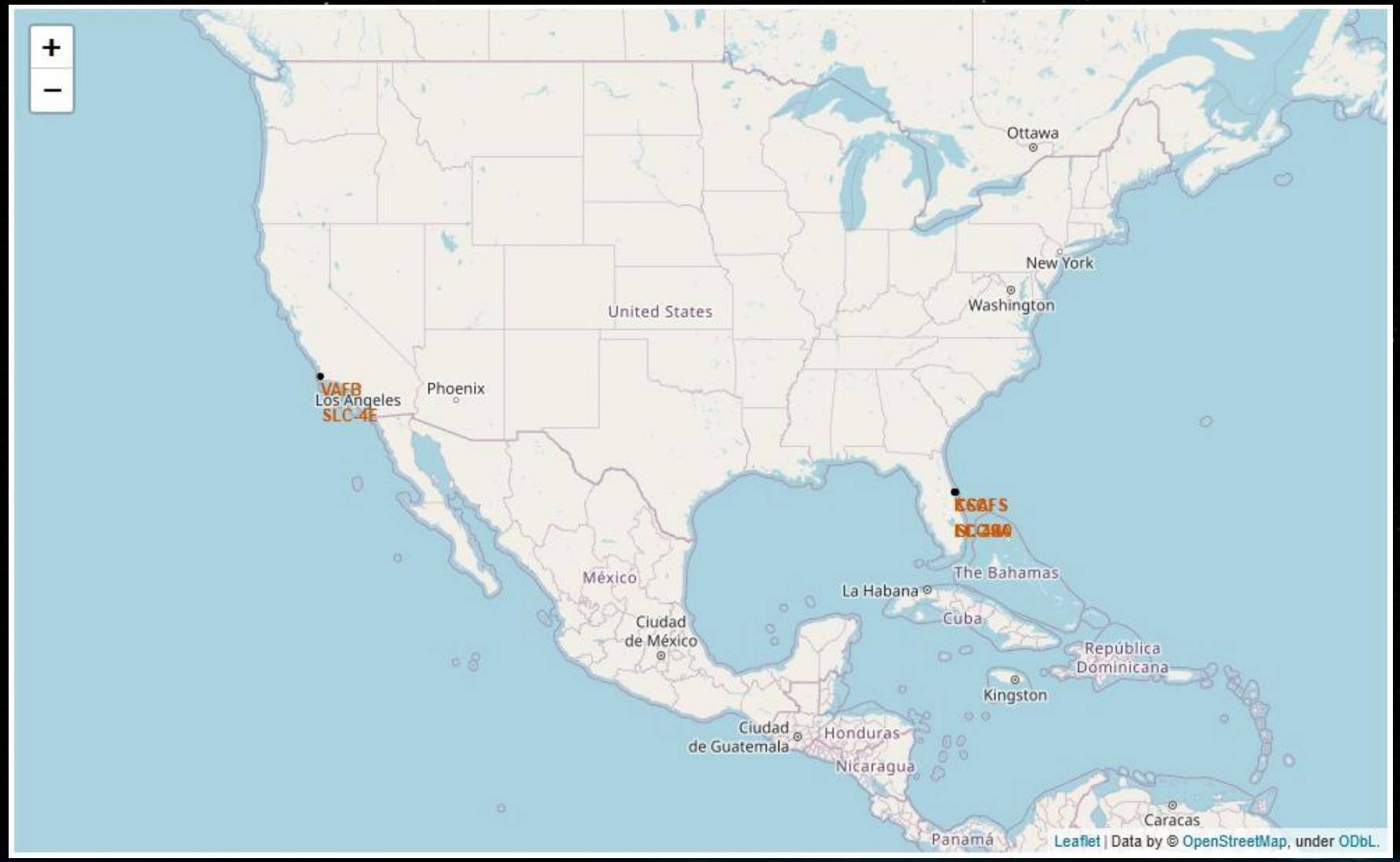
```
 * sqlite:///my_data1.db
Done.
```

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

Ranking the count of landing outcomes (such as Failure (drone ship) or Success (ground pad)) between the date 2010-06-04 and 2017-03-20 in descending order.

Interactive map with Folium

All launch sites' location markers on a global map



- Launch sites are near sea, probably by safety, but not too far from roads and railroads.

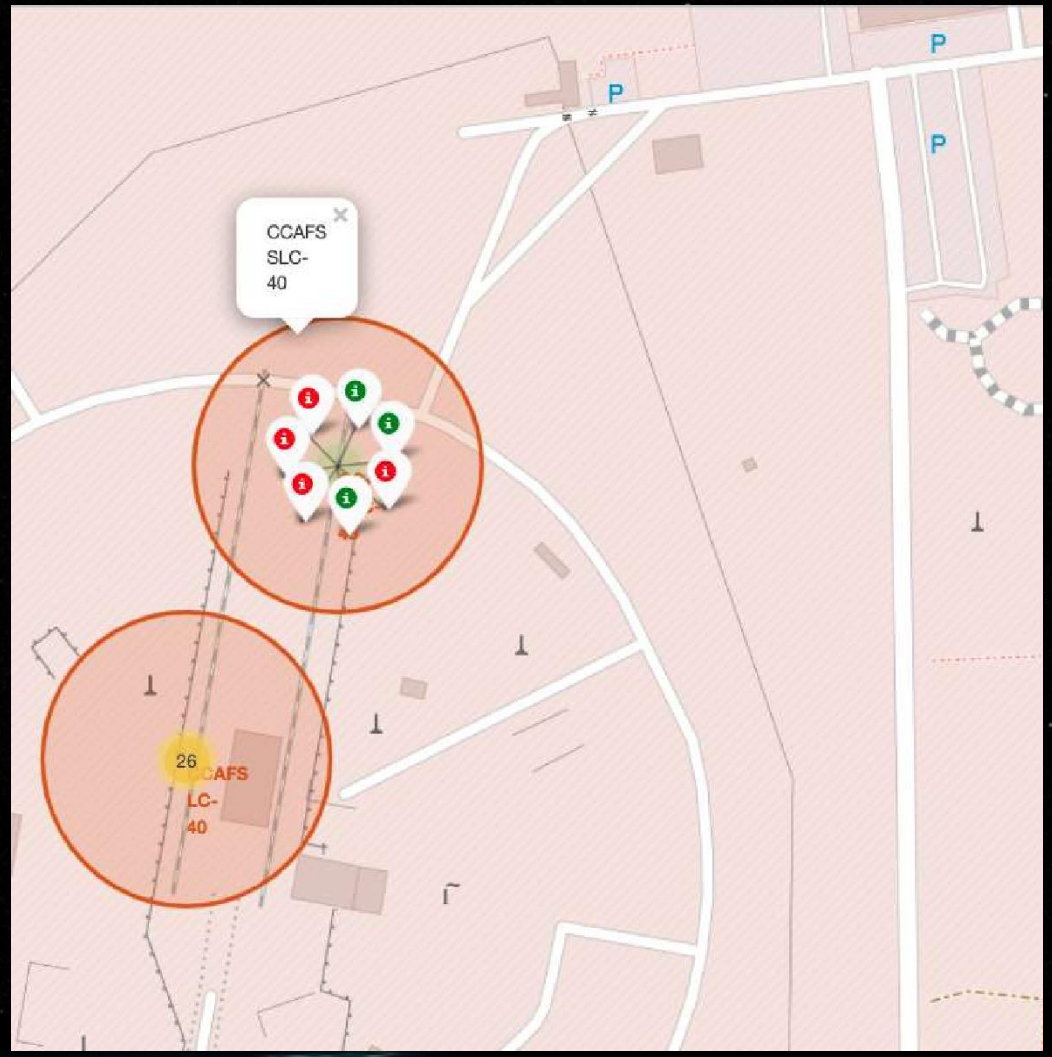
Colour-labeled launch records on the map

•From the colour-labeled markers we should be able to easily identify which launch sites have relatively high success rates.

-Green Marker = Successful Launch

-Red Marker = Failed Launch

•Launch Site KSC LC-39A has a very high Success Rate.



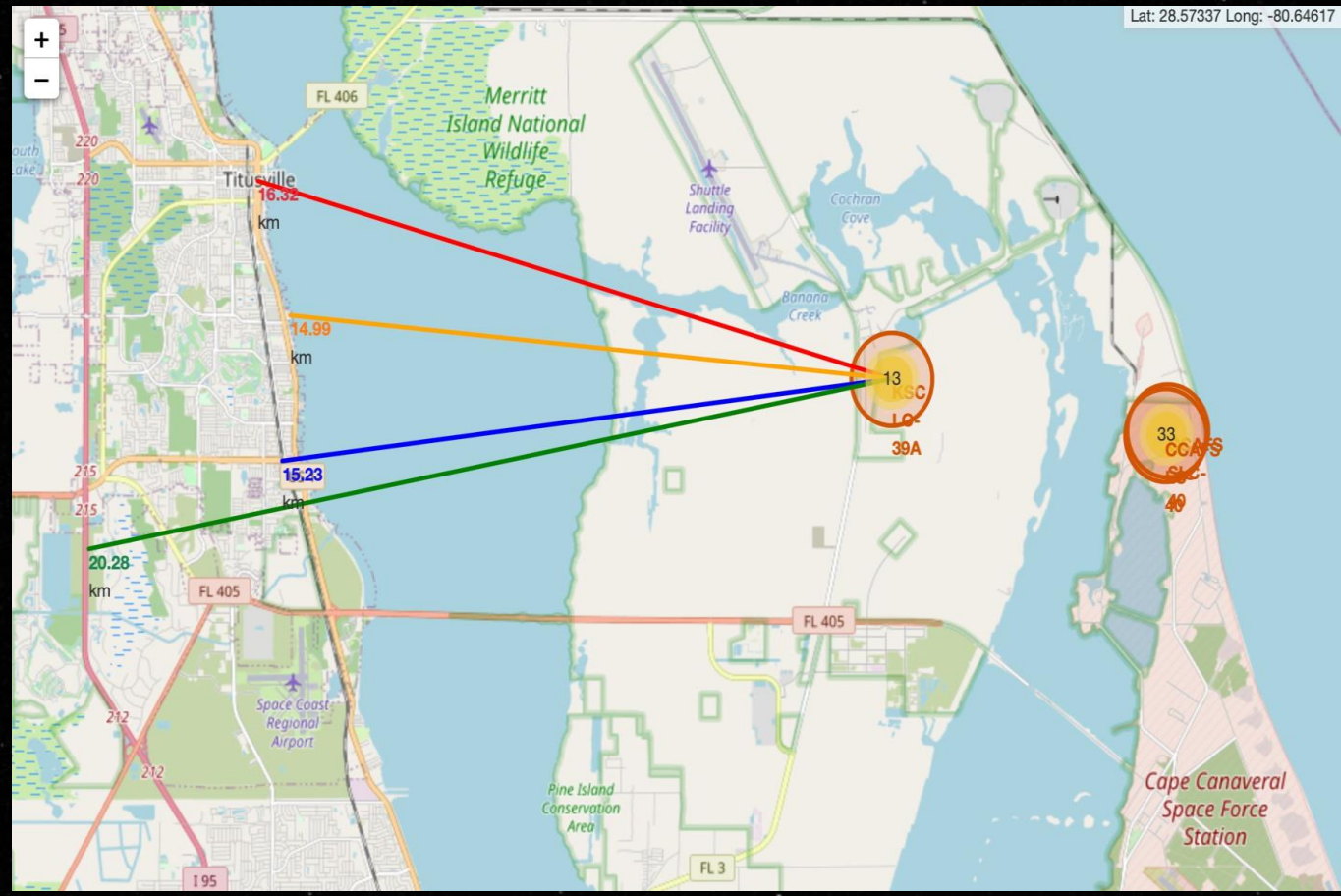
Distance from the launch site KSC LC-39A to its proximities

•From the visual analysis of the launch site KSC LC-39A we can clearly see that it is:

- relative close to railway (15.23 km)
- relative close to highway (20.28 km)
- relative close to coastline (14.99 km)

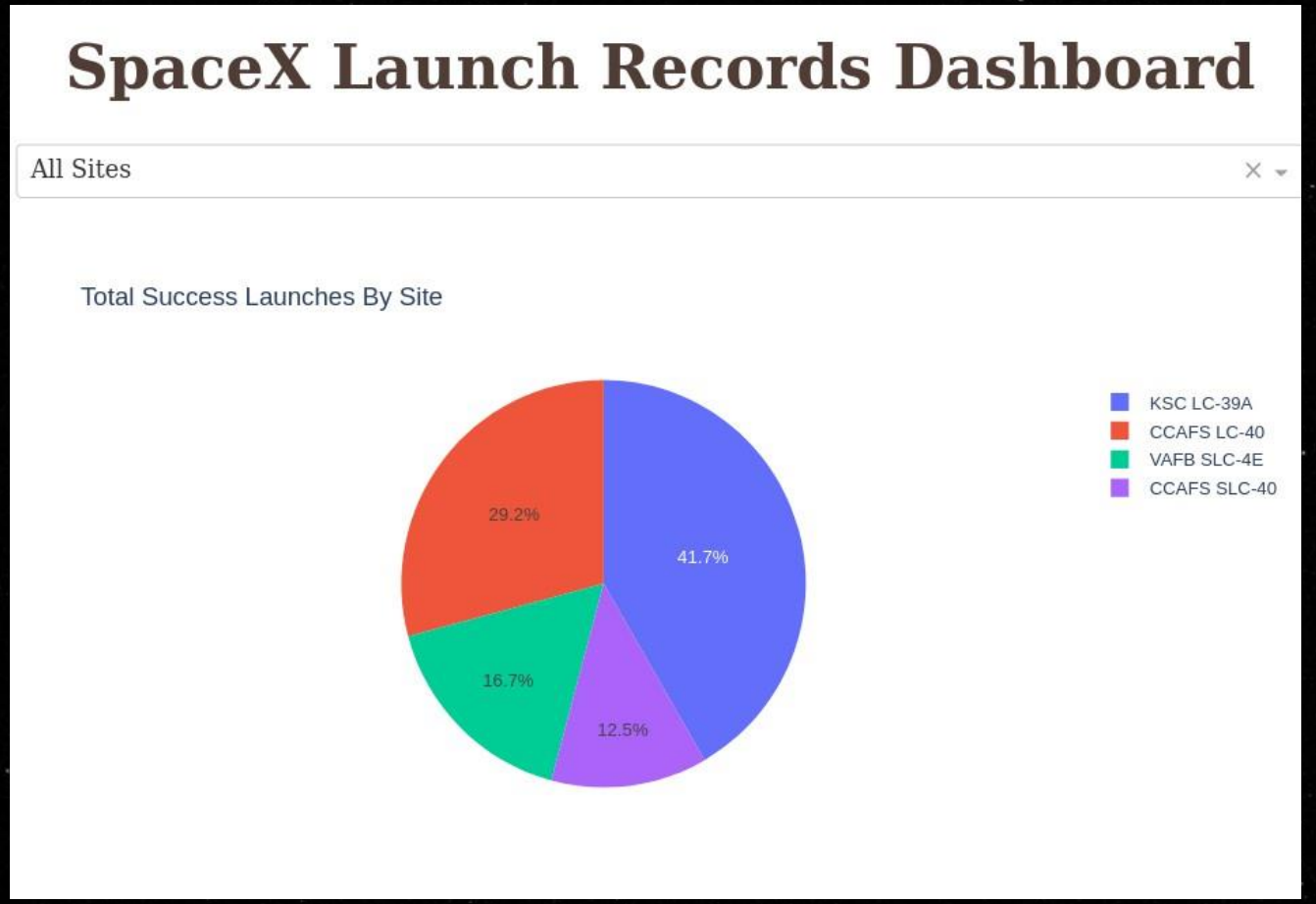
• Also the launch site KSC LC-39A is relative close to its closest city Titusville (16.32 km).

•Failed rocket with its high speed can cover distances like 15-20 km in few seconds. It could be potentially dangerous to populated areas.



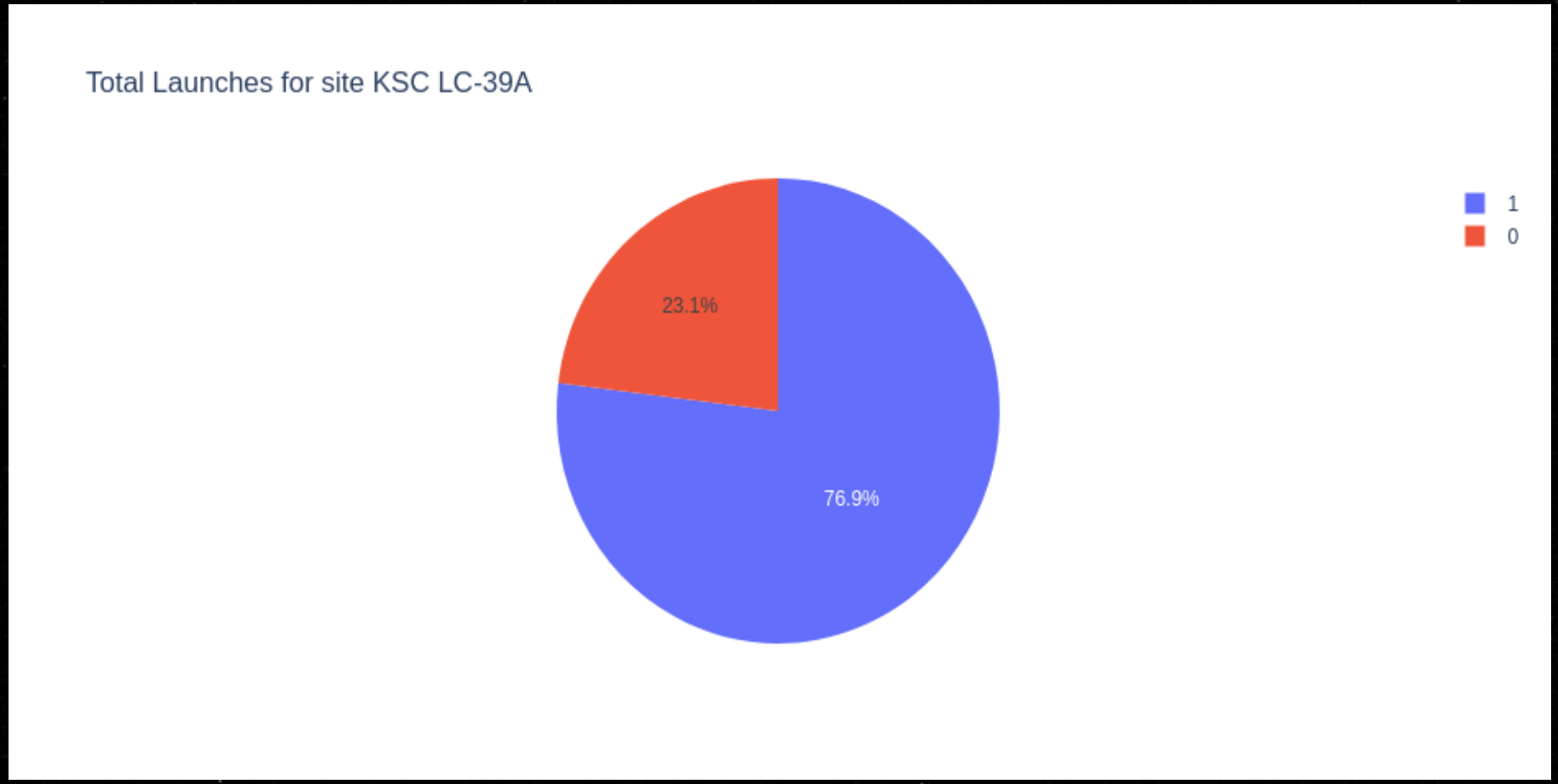
Build a Dashboard with Plotly Dash

Launch success count for all sites



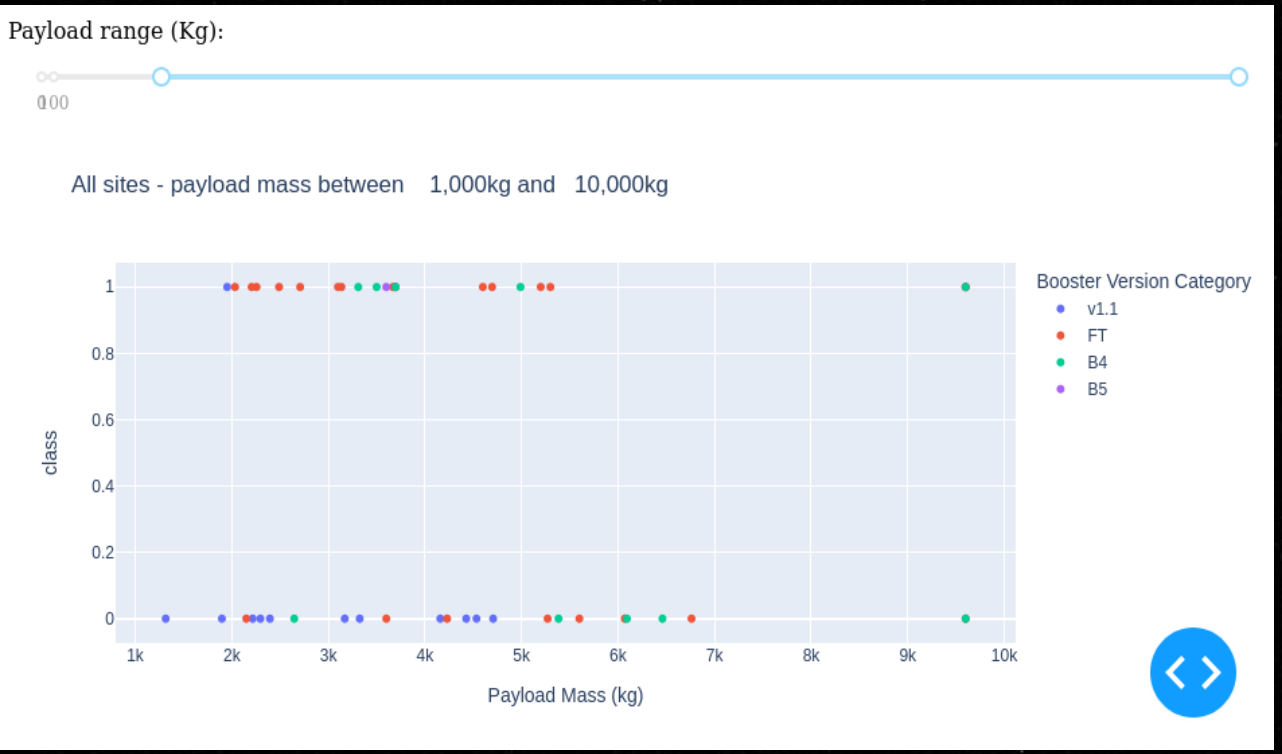
- The place from where launches are done seems to be a very important factor of success of missions.

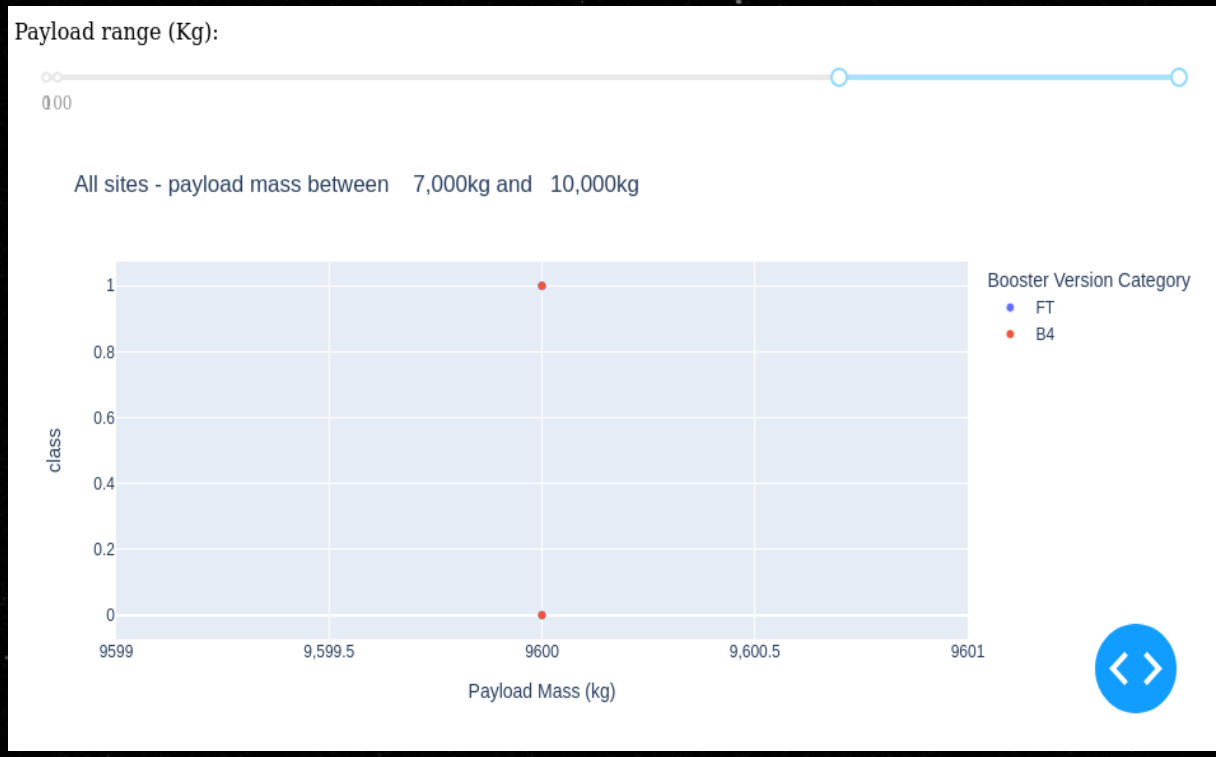
Launch site with highest launch success ratio



- 76.9% of launches are successful in this site.

Payload Mass vs. Launch Outcome for all sites





- Payloads under 6,000kg and FT boosters are the most successful combination.

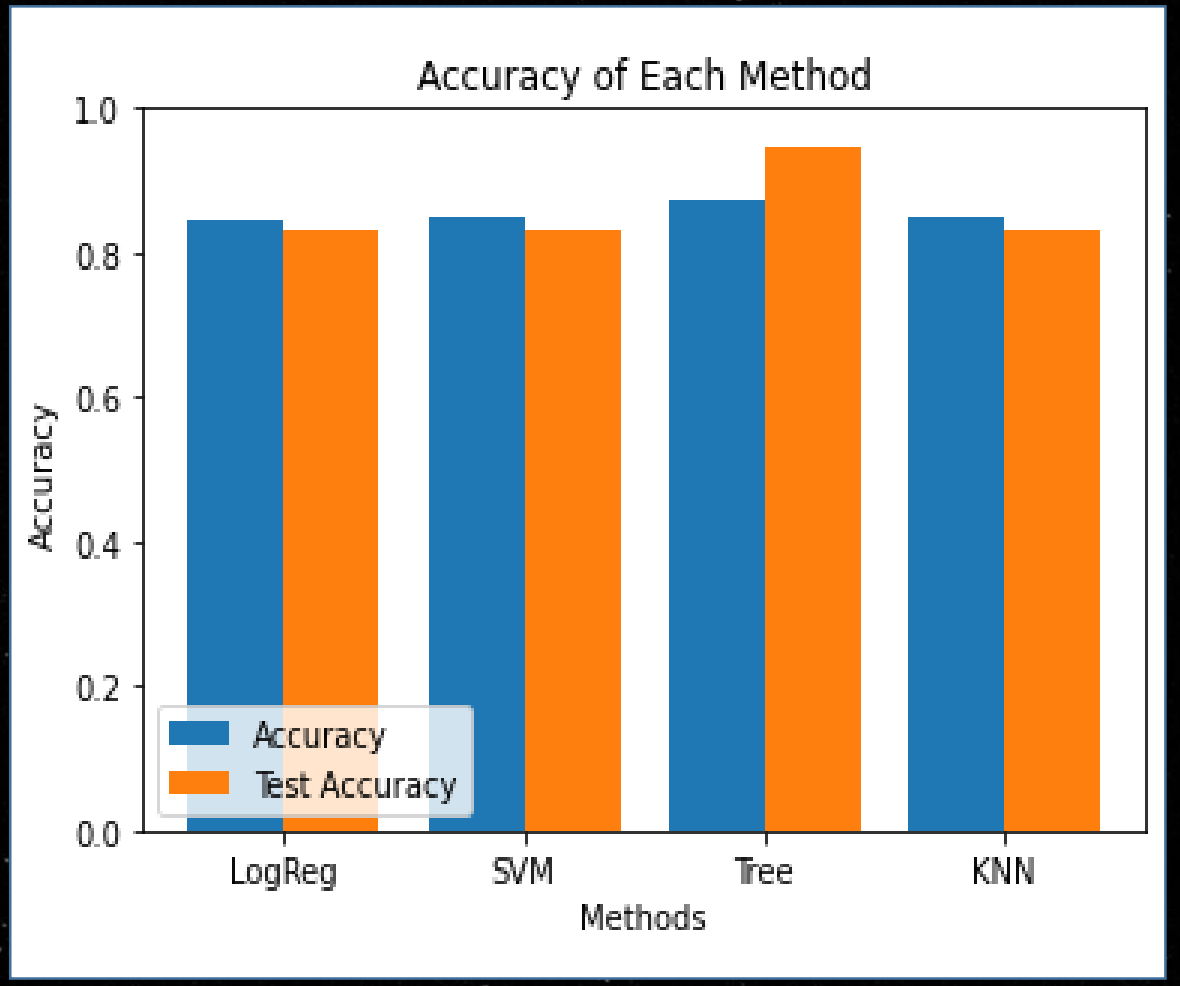
- There's not enough data to estimate risk of launches over 7,000kg

Predictive analysis (Classification)

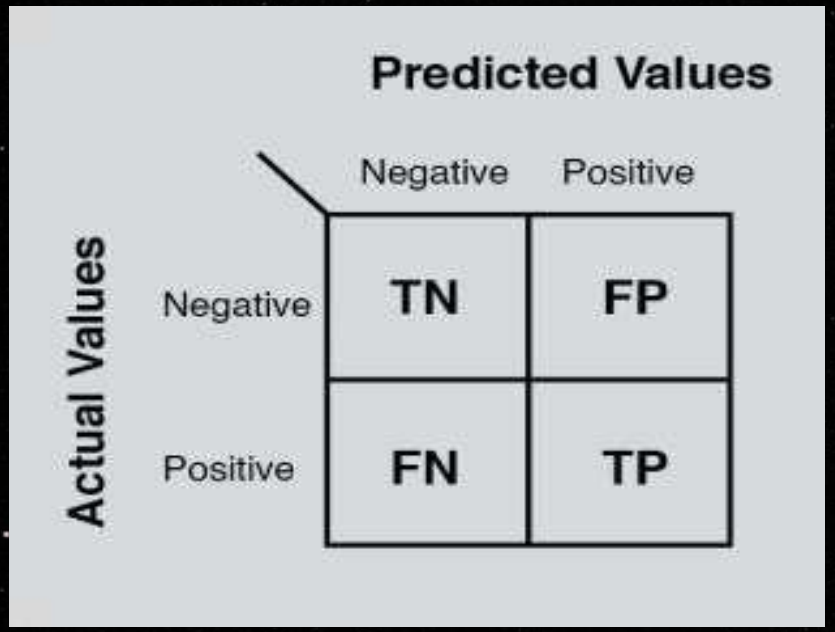
Classification Accuracy

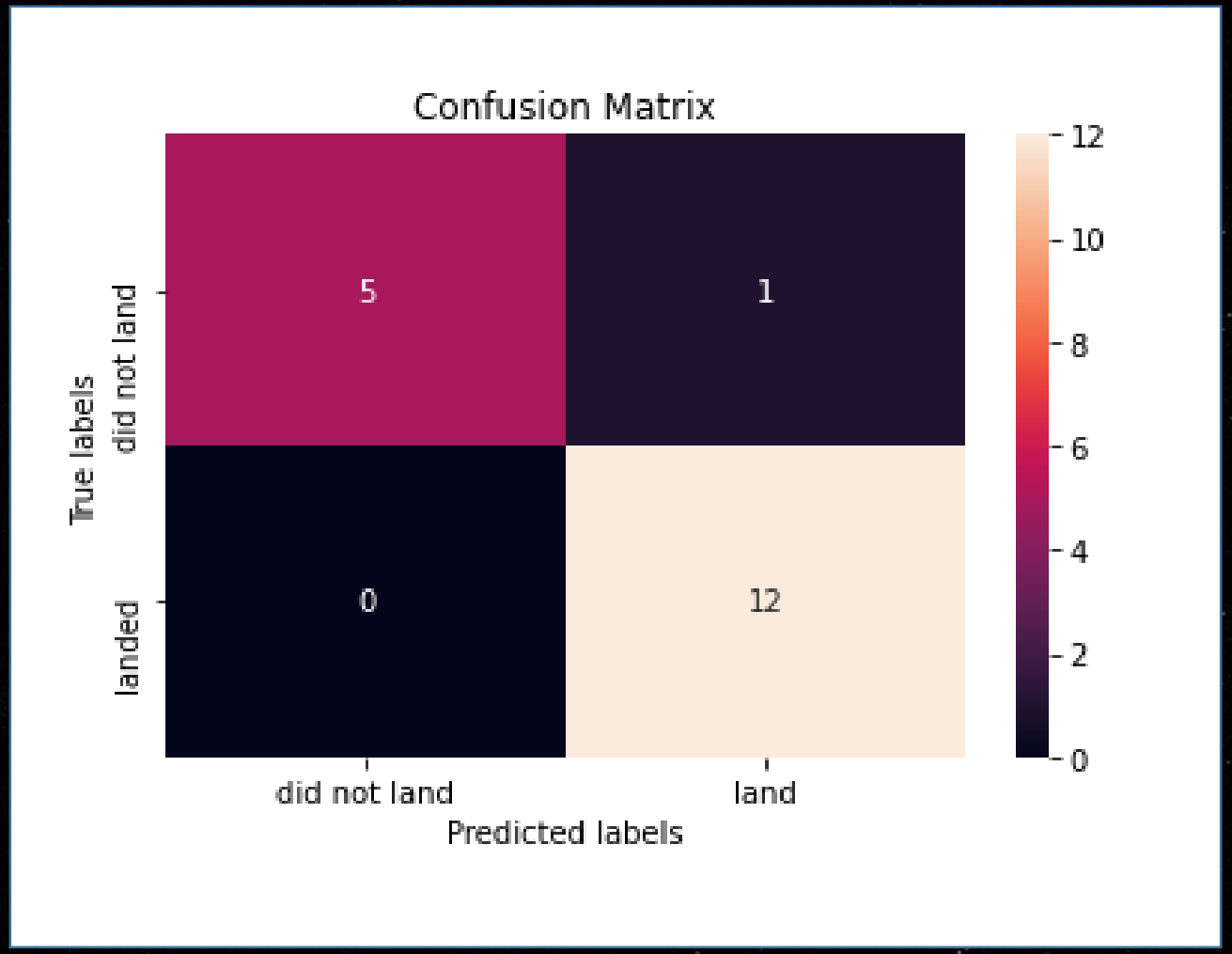
- Four classification models were tested, and their accuracies are plotted beside;

- The model with the highest classification accuracy is Decision Tree Classifier, which has accuracies over than 87%.



Confusion Matrix





- Confusion matrix of Decision Tree Classifier proves its accuracy by showing the big numbers of true positive and true negative compared to the false ones.

Conclusion

- Different data sources were analyzed, refining conclusions along the process;
- The best launch site is KSC LC-39A;
- Launches above 7,000kg are less risky;
- Although most of mission outcomes are successful, successful landing outcomes seem to improve over time, according the evolution of processes and rockets;
- Decision Tree Classifier can be used to predict successful landings and increase profits.

Appendix

SPECIAL THANKS TO:

INSTRUCTORS

COURSERA

IBM