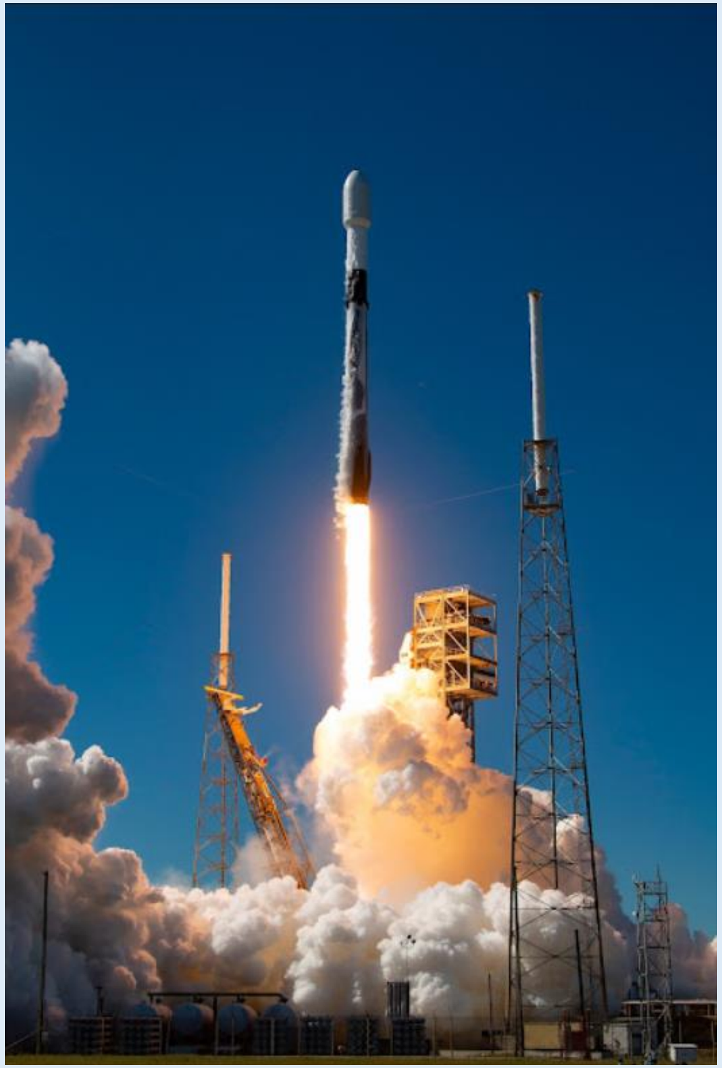
IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Rocío Rodríguez Oliva
23/03/2025

Outline

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



Executive Summary

- Summary of methodologies
  - SpaceX Data Collection SpaceXAPI
  - SpaceX Data Collection WEB Scrapping
  - SpaceX Data Collection Wrangling
  - SpaceX EDA with Visualizations
  - SpaceX EDA with SQL
  - SpaceX Interactive map with Folium
  - Interactive Dashboard with Plotly Dash
  - SpaceX Machine Learning Prediction

- Summary of results
  - EDA results
  - Interactive Visual Analytics and Dashboards
  - Predictive Analysis (Classification)

Introduction

This analysis is set in the commercial aerospace industry, where companies like SpaceX have revolutionized the sector by making space launches more affordable through rocket reusability.

The central problem lies in **determining the cost of a rocket launch**, considering the feasibility of reusing its first stage. Since SpaceX can recover this section of the rocket, its costs are significantly lower compared to other providers (62 million dollars vs 165 million dollars).

To address this problem, the following key questions are posed:

- What factors determine whether the first stage of a Falcon 9 rocket will be reusable?
- How does rocket reusability impact the total launch cost?
- Can machine learning models predict the successful landing of the first stage?

In this project, a data-driven analysis will be conducted using public information about SpaceX, applying data science and machine learning techniques to predict the reusability of the first stage and estimate launch costs.

Section 1

Methodology

Data Collection - Scraping

- BeautifulSoup will be used to extract data from HTML tables on Wikipedia List of Falcon 9 and Falcon Heavy launches - Wikipediaabout Falcon 9 launches. Parsed the table and converted in to a Pandas data frame.
- https://github.com/RRO0904/SpaceY/blob/main/2%20SpaceX%20Data%20Collection%20Web%20Scrapping%20(1).ipynb

TASK 1: Request the Falcon9 Launch Wiki page from its URL

First, let's perform an HTTP GET method to request the Falcon9 Launch HTML page, as an HTTP response.

```
# use requests.get() method with the provided static_url
data = requests.get(static_url)
# assign the response to a object
print(data.status_code)
data.text
```

```
th_elements = soup.find_all('th')

# Lista para almacenar los nombres de las columnas
column_names = []

# Iterar sobre cada elemento <th> y extraer el texto
for th in th_elements:
    # Obtener el texto limpio, sin etiquetas HTML
    text = th.get_text(separator=' ', strip=True)

    # Asegurarse de que el texto no esté vacío
    if text:
        column_names.append(text)

# Imprimir los nombres de las columnas
print(column_names)
```

Data Wrangling

Key Steps in Data Wrangling:

1.Data Acquisition

Extract SpaceX launch data from available sources.

Identify key attributes: *Flight Number, Date, Booster Version, Payload Mass, Orbit, Launch Site, Outcome, Grid Fins, Legs, Landing Pad, Block, Reused Count, Serial, Longitude, and Latitude.*

2.Data Cleaning & Transformation

Standardize column names and formats.

Handle missing values and inconsistencies.

3.Feature Engineering

Launch Site Categorization: Identify launch sites (*Vandenberg AFB, Kennedy Space Center, CCAFS SLC 40*).

Orbit Classification: Define orbit types (*LEO - Low Earth Orbit, GTO - Geosynchronous Transfer Orbit…*).

Landing Outcome Encoding: Convert landing results into binary classification (*0 = Failed, 1 = Successful*).

4.Data Encoding & Preparation

Convert categorical variables into numerical format.

Create the target variable Y for model training.

Methodology

Executive Summary

- Data collection methodology:

    Describe how data were collected.

- Perform data wrangling:

    Describe how data were processed.

- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models.

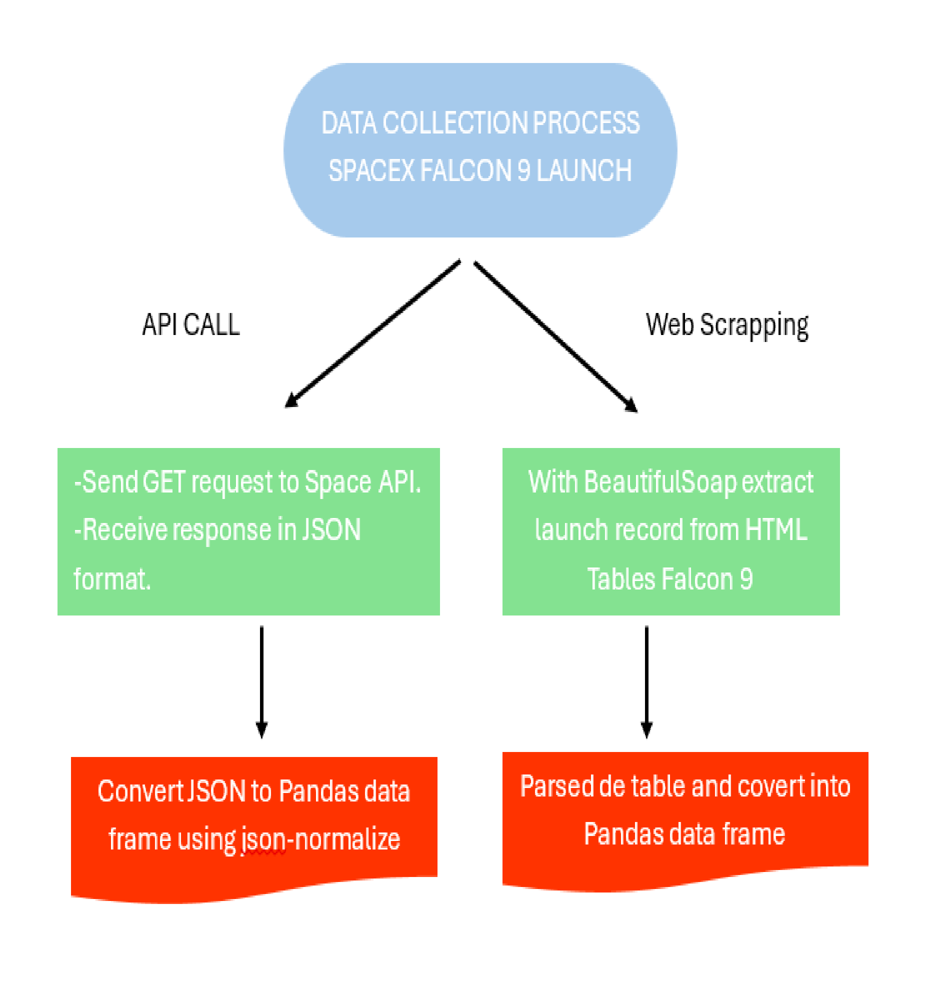
    How to build, tune, evaluate classification.

Data Collection

This analysis focuses on collecting SpaceX launch data through the SpaceX REST API and web scraping Wikipedia page called List of Falcon 9 and Falcon Heavy launches - Wikipedia. The main goal is to predict whether SpaceX will attempt to land a rocket using detailed launch information, such as rocket type, payload, launch specifications, and landing outcome.

To achieve this, the following key steps will be taken:

- Data Collection from the API: The requests library will be used to make GET requests and extract data in JSON format. Then, the data will be normalized using json_normalize to convert it into a Pandas data frame.
- Web Scraping: BeautifulSoup will be used to extract data from HTML tables on Wikipedia about Falcon 9 launches. Then parsed the table and converted into a Pandas data frame.



Data Collection – SpaceX API

- The requests library will be used to make GET requests and extract data in JSON format(api.spacexdata.com/v4/launches/past). Receive response in JSON format. Then, the data will be normalized using json_normalize to convert it into a Pandas DataFrame.

- https://github.com/RRO0904/SpaceY/blob/main/1%20SpaceX%20Data%20Collection%20SpaceX%20API%20(1).ipynb

Task 1: Request and parse the SpaceX launch data using the GET request

To make the requested JSON results more consistent, we will use the following static response object for this project:

```
static_json_url='https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DS0321EN-SkillsNetwork/datasets/API_call_spacex_api.json'
```

We should see that the request was successfull with the 200 status response code

```
response.status_code
```

Now we decode the response content as a Json using `.json()` and turn it into a Pandas dataframe using `.json_normalize()`

```
# Use json_normalize meethod to convert the json result into a dataframe
respjson = response.json()
data = pd.json_normalize(respjson)
```

Using the dataframe `data` print the first 5 rows

```
# Get the head of the dataframe
data.head()
```

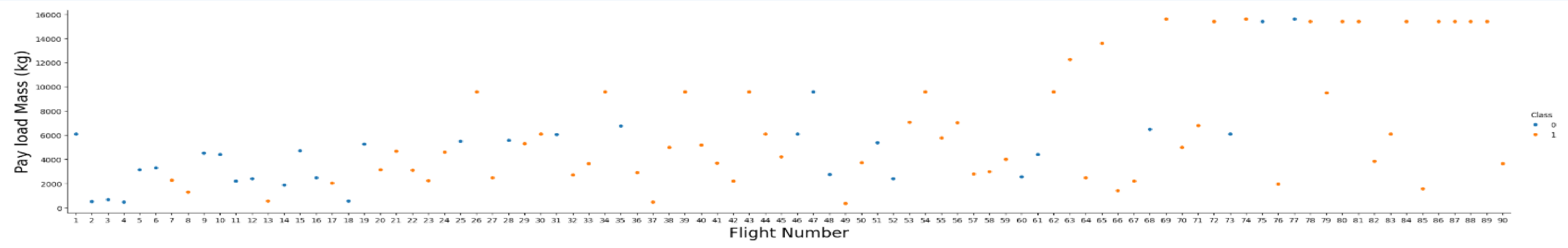
Data Wrangling

- Cleaning and transforming SpaceX launch data, focusing on key variables related to rockets and their performance. Different launch sites are identified, orbit types are defined, booster landing outcomes are categorized, and the Outcome variable is redefined as a binary classification variable. This transformation facilitates analysis and enables training Machine Learning models to predict booster reuse.
- https://github.com/RRO0904/SpaceY/blob/main/3%20SpaceX%20Data%20Wrangling%20(1).ipynb
- **Flowchart Representation of Data Wrangling Process:**

Data Acquisition --> Data Cleaning --> Feature Engineering --> Encoding & Preparation --> Final Dataset

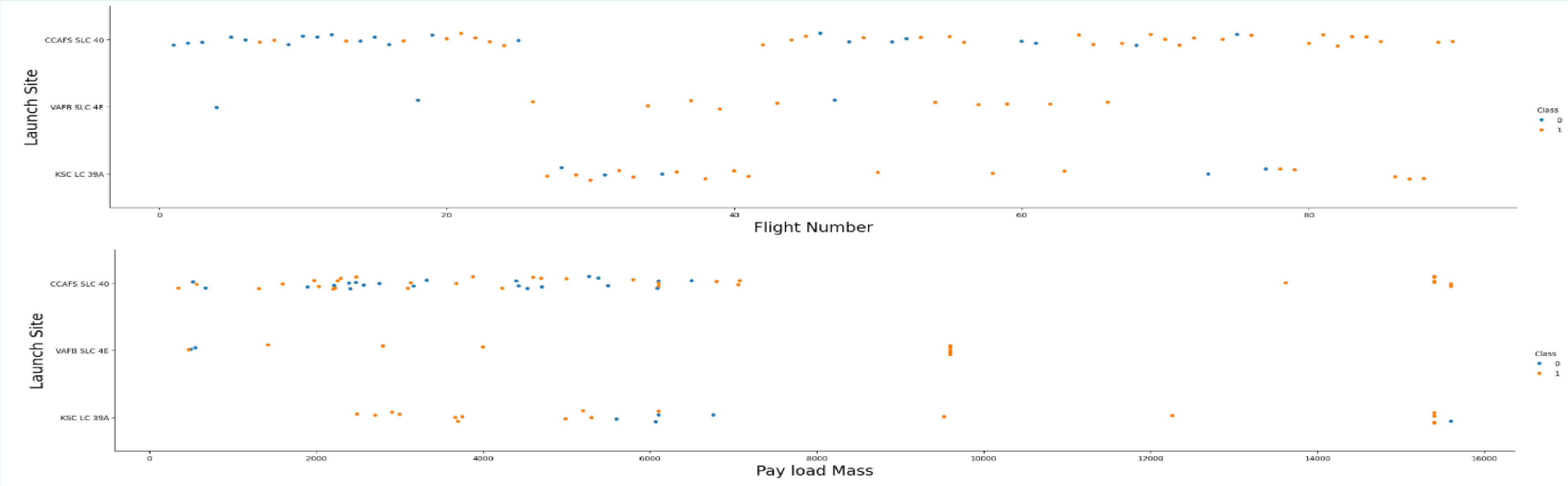
EDA with Data Visualization

An Exploratory Data Analysis (EDA) is performed using scatterplots, bar charts, and line plots to gain preliminary insights into how each important variable affects the success rate. Next, feature engineering will be conducted to select the most relevant features for building a predictive model.



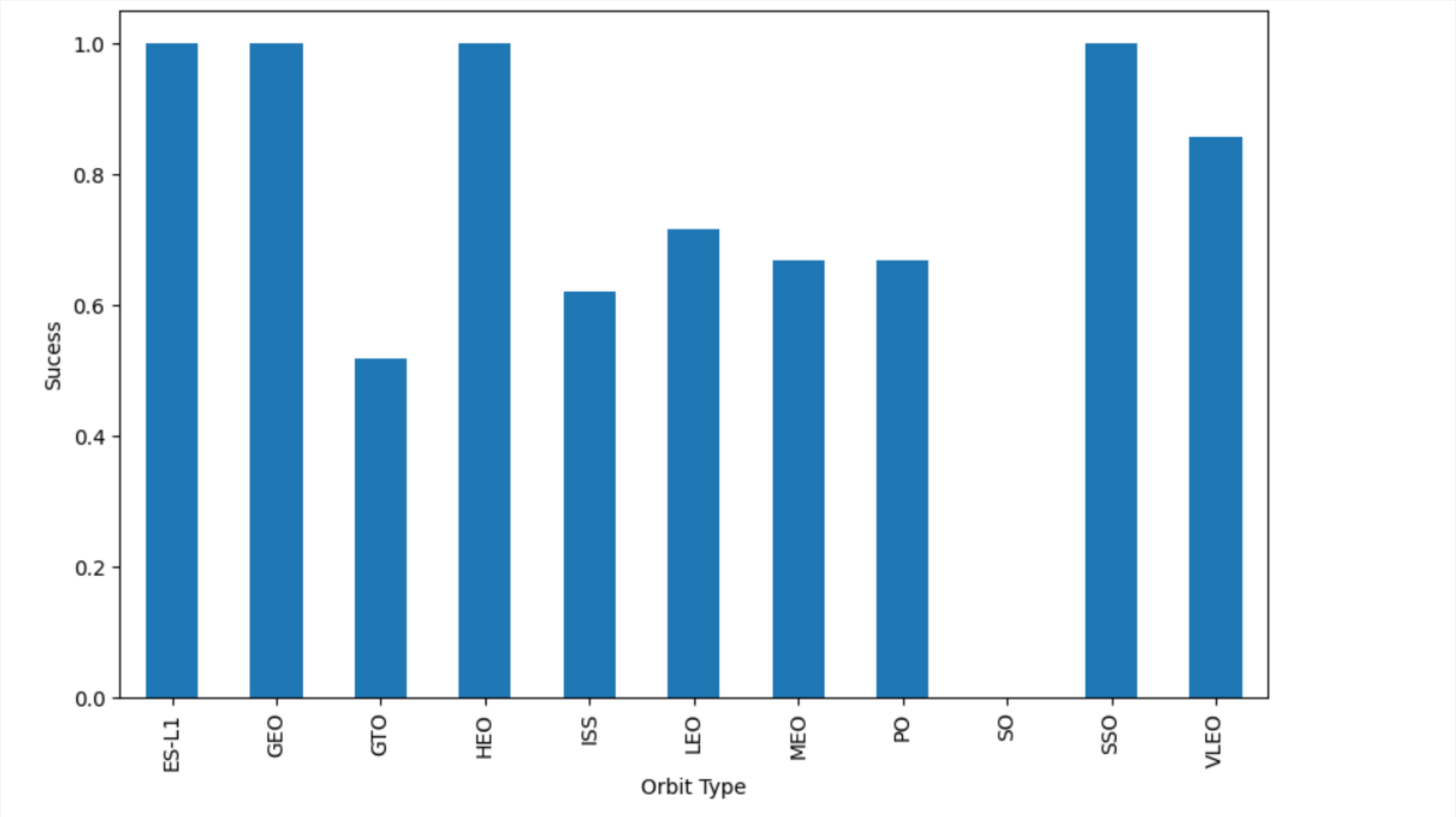
As Flight number increases, the first stage is more likely to land successfully. On the other hand, the more massive the payload, the less likely the first stage will return.

EDA with Data Visualization

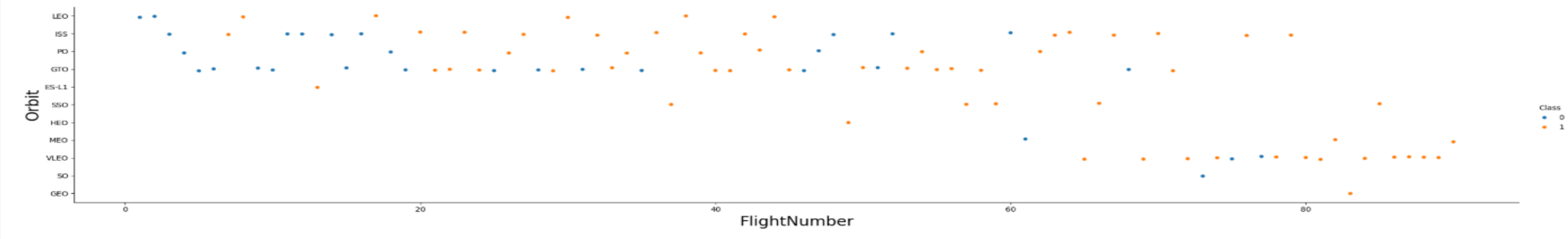


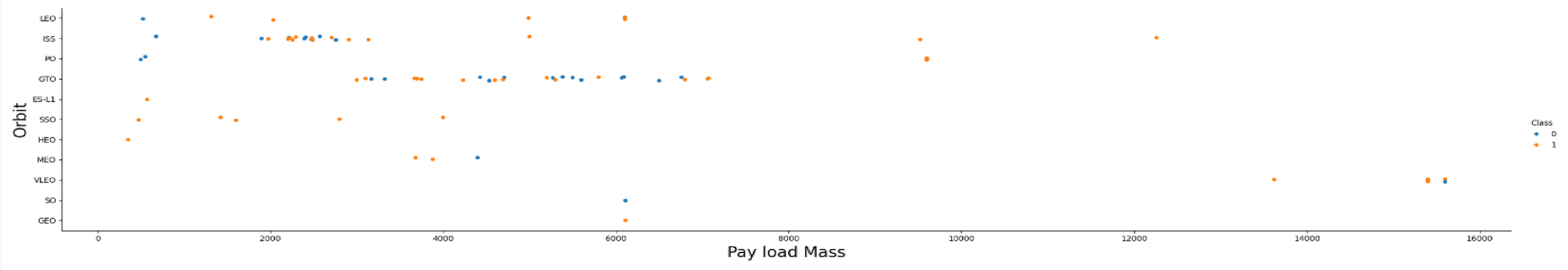
Different launch sites exhibit varying success rates, with CCAFS LC-40 at 60% and KSC LC-39A / VAFB SLC 4E around 77%. At VAFB-SLC launchsite there are no rockets launched for heavypayload mass(greater than 10000).

EDA with Data Visualization



EDA with Data Visualization

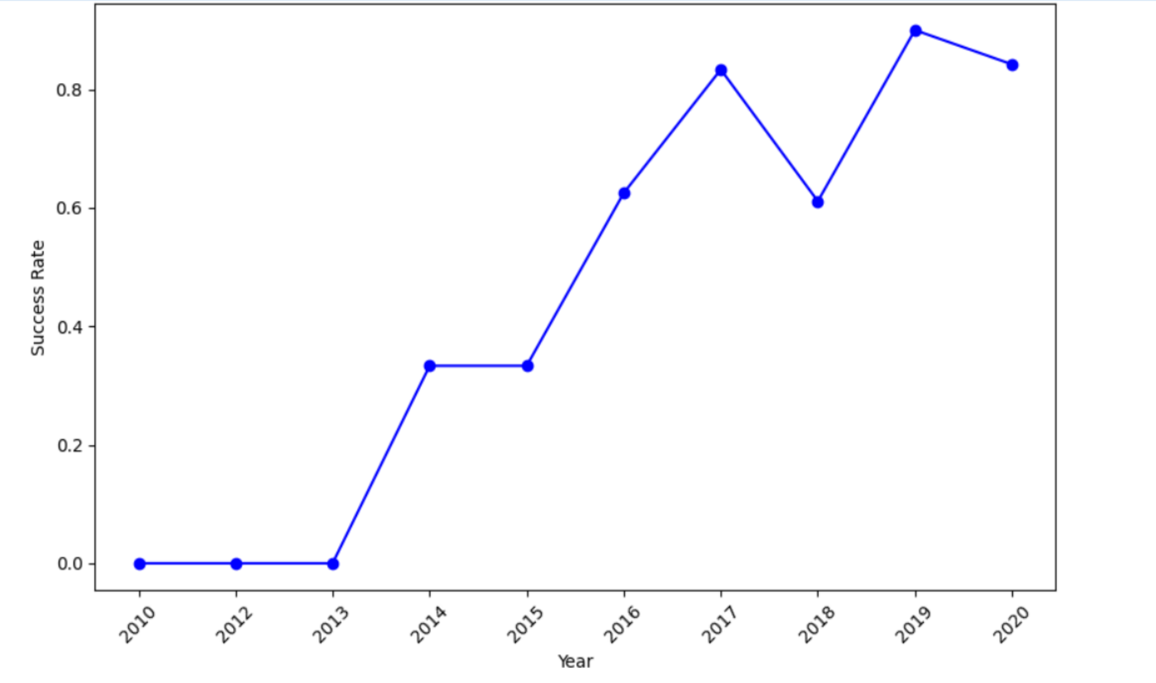




In the LEO orbit the success appears related to the number of flights; on the other hand, there seems to be no relationship between flight number when in GTO orbit. With heavy payloads the successful landing or positive landing rate are more for Polar, LEO and ISS. However for GTO we cannot distinguish this well as both positive landing rate and negative landing (unsuccessful mission) are both there here.

EDA with Data Visualization

The success rate of landings has improved since 2013, highlighting the advancements in technology and operational efficiency.



https://github.com/RRO0904/SpaceY/blob/main/4%20SpaceX%20EDA%20with%20Visualization%20(1).ipynb

EDA with SQL

Task 1

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

```
%sql SELECT DISTINCT "Launch_Site" FROM SPACEXTBL;
```

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

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

Task 2

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

```
%sql SELECT * FROM SPACEXTBL WHERE "Launch_Site" LIKE 'CCA%' LIMIT 5;
```

```
 * sqlite:///my_data1.db
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	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
2012-10-08	0:35:00	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

EDA with SQL

Task 3

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

```
%sql SELECT SUM("PAYLOAD_MASS__KG_") FROM SPACEXTBL WHERE "Customer" LIKE 'NASA (CRS)';
```

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

SUM(PAYLOAD_MASS__KG_)

Task 4

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

```
%sql SELECT AVG("PAYLOAD_MASS__KG_") FROM SPACEXTBL WHERE "Booster_Version" LIKE 'F9 v1.1';
```

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

AVG(PAYLOAD_MASS__KG_)
2928.4

EDA with SQL

Task 5

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

Hint:Use min function

```
%sql select min(Date) from SPACEXTABLE where Landing_Outcome = 'Success (ground pad)'
```

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

min(Date)
2015-12-22

Task 6

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 SPACEXTABLE\
where Landing_Outcome = "Success (drone ship)" and PAYLOAD_MASS__KG_ between 4000 and 6000;
```

EDA with SQL

Task 7

List the total number of successful and failure mission outcomes

```
%sql SELECT Mission_Outcome, COUNT(*) as count from SPACEXTABLE\
GROUP BY Mission_Outcome ORDER BY count DESC;
```

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

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

Task 8

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

```
%sql select Booster_Version from SPACEXTABLE\
where PAYLOAD_MASS__KG_ = (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

EDA with SQL

Task 9

List the records which will display the month names, failure landing_outcomes in drone ship ,booster versions, launch_site for the months in year 2015.

Note: SQLLite does not support monthnames. So you need to use substr(Date, 6,2) as month to get the months and substr(Date,0,5)='2015' for year.

```
%sql select substr(Date, 6, 2) as month, Landing_Outcome, Booster_Version, Launch_Site from SPACEXTABLE\
where substr(Date, 1, 4) = '2015' and Landing_Outcome = 'Failure (drone ship)'
```

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

month	Landing_Outcome	Booster_Version	Launch_Site
01	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
04	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

EDA with SQL

Task 10

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

```
%sql select Landing_Outcome, count(*) as count_outcomes \
from SPACEXTABLE\
where Date between '2010-06-04' and '2017-03-20'\
group by Landing_Outcome\
order by count_outcomes Desc;
```

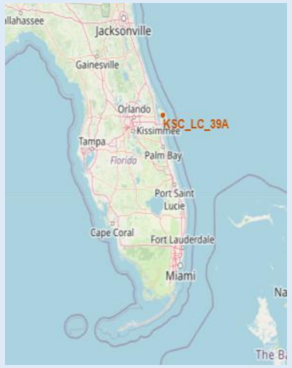
```
 * sqlite:///my_data1.db
Done.
```

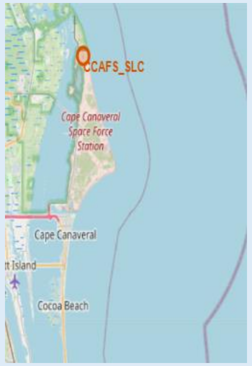
Landing_Outcome	count_outcomes
No attempt	10
Success (drone ship)	5
Failure (drone ship)	5
Success (ground pad)	3
Controlled (ocean)	3
Uncontrolled (ocean)	2
Failure (parachute)	2
Precluded (drone ship)	1

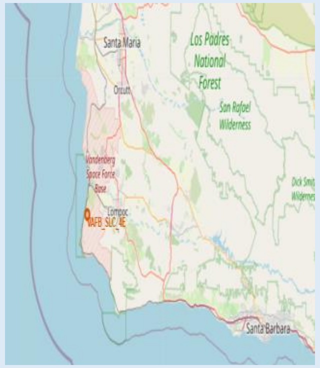
https://github.com/RRO0904/SpaceY/blob/main/5%20SpaceX%20EDA%20with%20SQL%20(1).ipynb

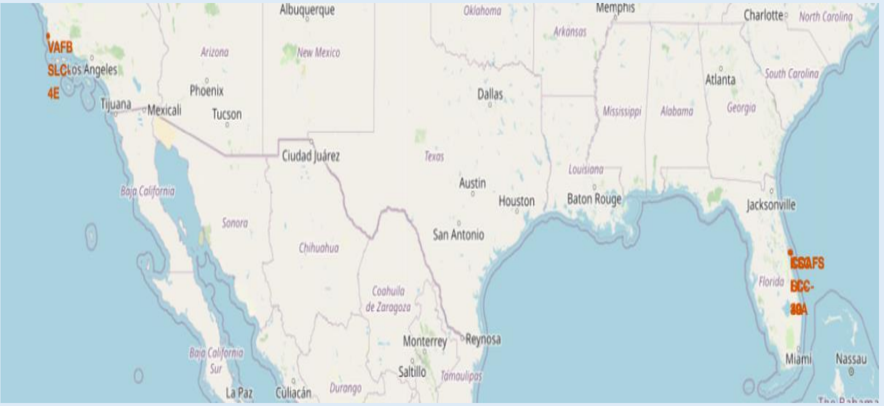
Build an Interactive Map with Folium

It was created a Folium Map object and created and add folium.Circle and folium.Marker for each launch site on the site map.
VAFB SLC is farther from Ecuador line while all the launches sites are proximity to line coast.





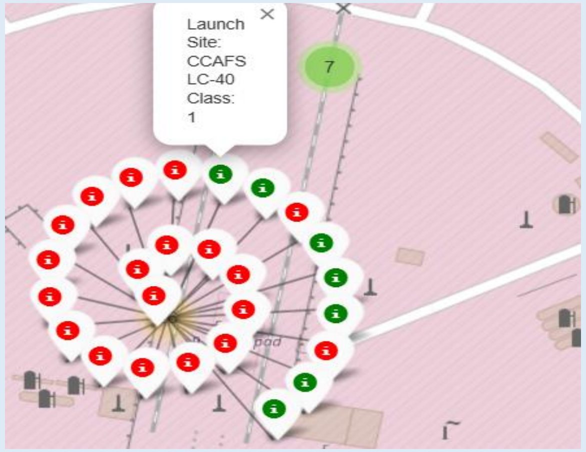


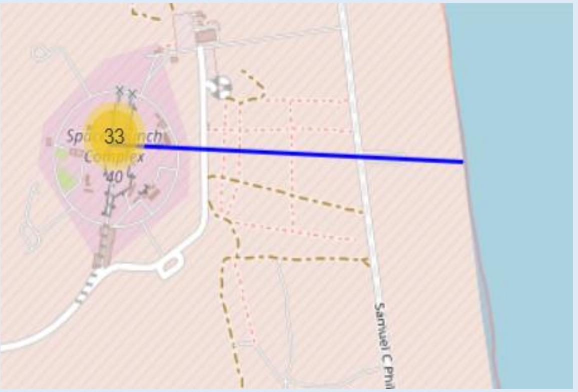


Build an Interactive Map with Folium

Next, the map was enhanced by adding the launch outcomes for each site and to see which site have high success rates.
Finally, it was calculated the distances between a launch site to its proximity by adding lines to know if there are somo patterns o characteristics.
It can be concluded that the launchsites are in close proximity to coast and close to Major Highway and Railway for logistic purposes. But launch sites are far from dense human habitats like cities.

https://github.com/RRO0904/SpaceY/blob/main/6%20SpaceX%20Interactive%20map%20with%20Folium%20%20(1).ipynb





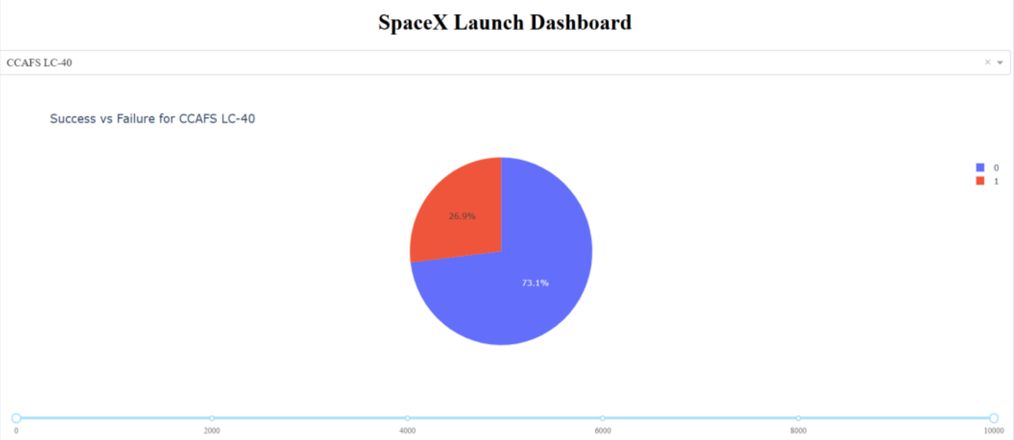
Build a Dashboard with Plotly Dash

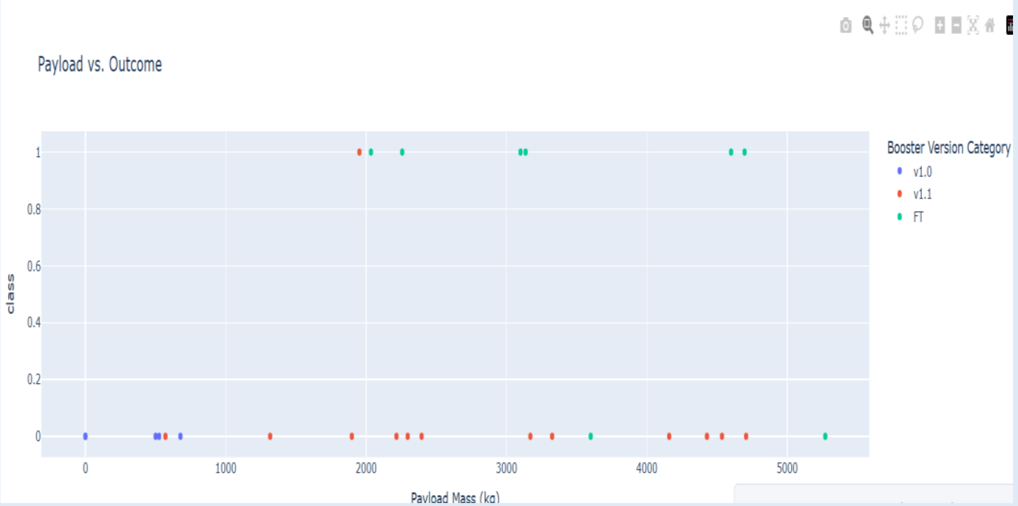
This dashboard application contains input components such as a dropdown list and a range slider to interact with a pie chart and a scatter point chart.

It could be concluded that:

- KSC LC has the largest successfull launches.
- CCAFS SLC has the highest success rate, 43%.
- 2000k – 4000k is the range with the highest launch success rate.
- 0k – 2000k is the range with lowest launch success rate.
- F9 Booster version FT has the highest launch success rate.

https://github.com/RRO0904/SpaceY/blob/main/7%20Interactive%20Dashboard%20with%20Plotly%20Dash.ipynb





Predictive Analysis (Classification)

It has been built a machine learning pipeline to predict if the first stage of Falcon 9 lands successfully.

Key Steps in the Machine Learning Pipeline

- **Preprocessing Data**
    - Convert 'Class' column to a Numpy array (to_numpy()) → Assign to y.
    - Standardize X using fit_transform() Assign to X.
- **Splitting Data**
    - Use train_test_split(X, y, test_size=0.20, random_state=2).
- **Training & Hyperparameter Tuning**
    - Train models and perform **Grid Search** to optimize hyperparameters.
- **Model Evaluation**
    - Select the best model based on accuracy.
    - Test models: **Logistic Regression, SVM, Decision Tree, KNN**.
    - Output **confusion matrix** and evaluate performance

Predictive Analysis (Classification)

Flowchart Representation of Machine Learning Pipeline

```
Start
   |
   v
   Preprocessing --> Data Splitting --> Model Training --> Hyperparameter Tuning --> Model Evaluation --> Best Model Selection
```

https://github.com/RRO0904/SpaceY/blob/main/8%20SpaceX-Machine-Learning-Prediction-Part-5-v1%20(1).ipynb

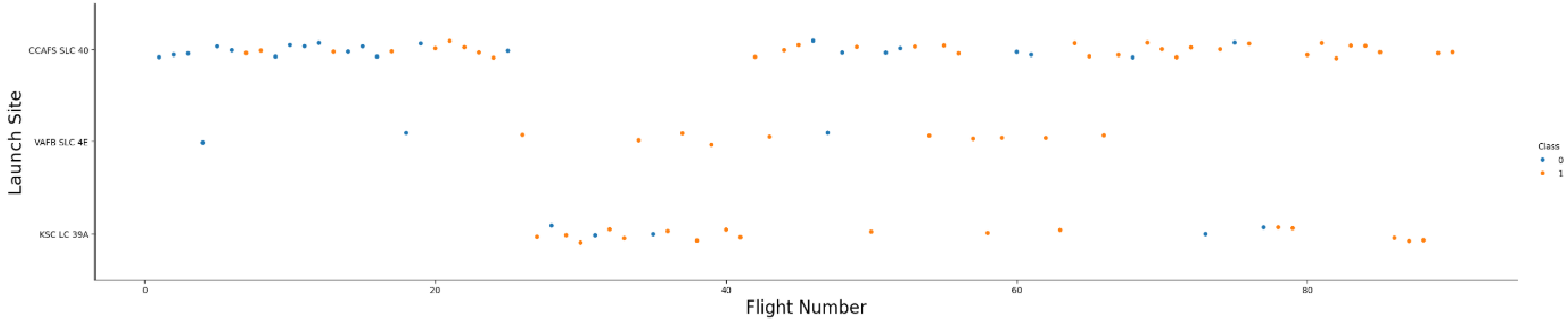
Results

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

Section 2

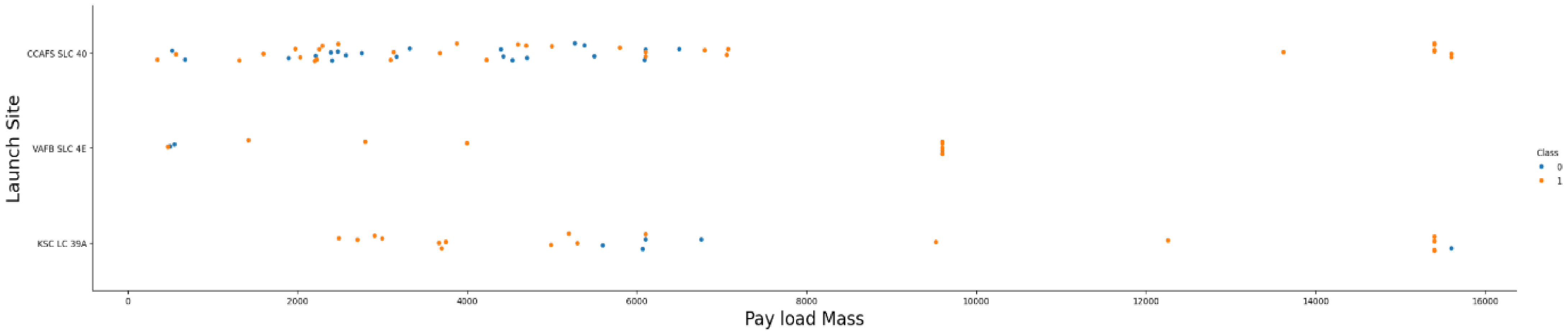
Insights drawn from EDA

Flight Number vs. Launch Site



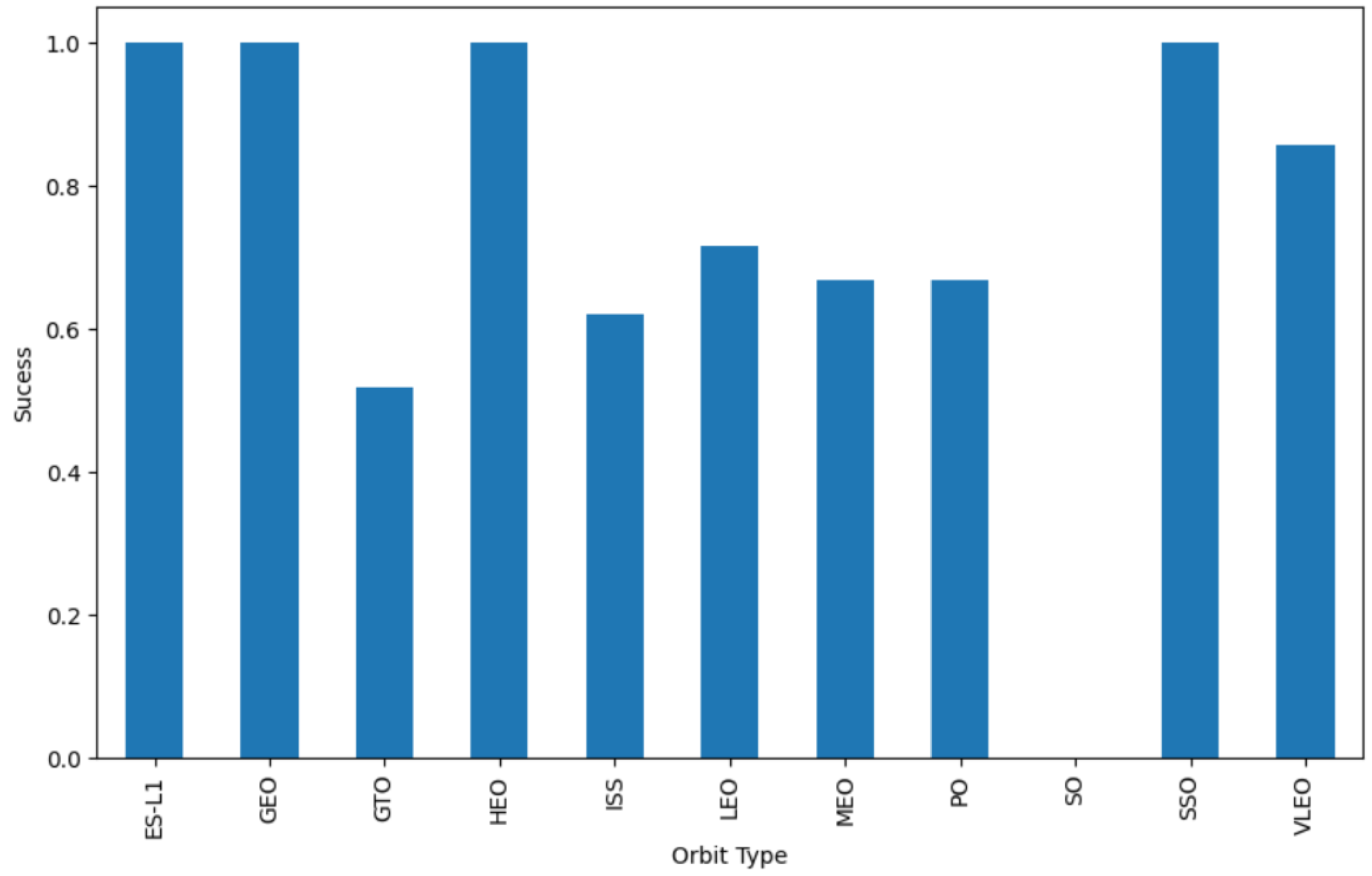
A higher number of flights increases the probability that the first stage will return, although this is more evident in some sites than in others. VAFR SLC 4F and KSC LC 39A had a ratio success class of 77% and CCAFS SLC 40 was 60% althogh this last launch site had de higher number of 'Flight Number'.

Payload vs. Launch Site



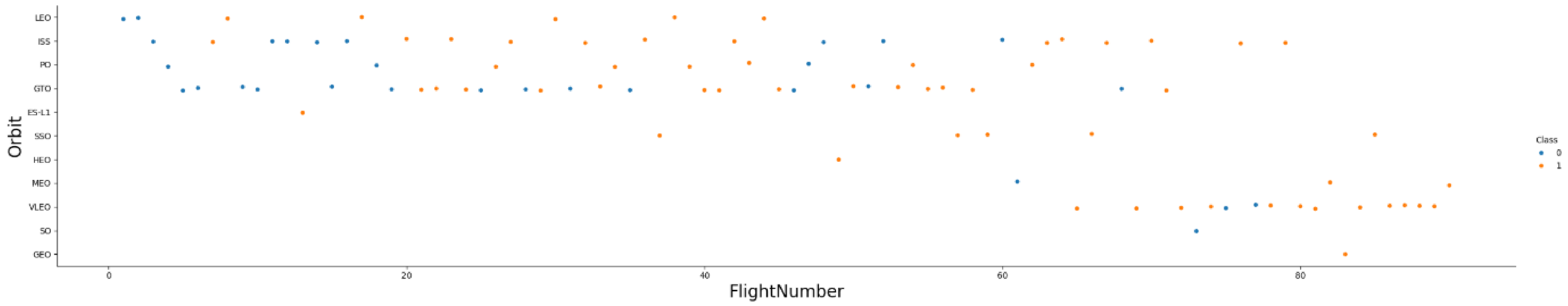
Now if you observe Payload Vs. Launch Site scatter point chart you will find for the VAFB-SLC launchsite there are no rockets launched for heavypayload mass(greater than 10000).

Success Rate vs. Orbit Type



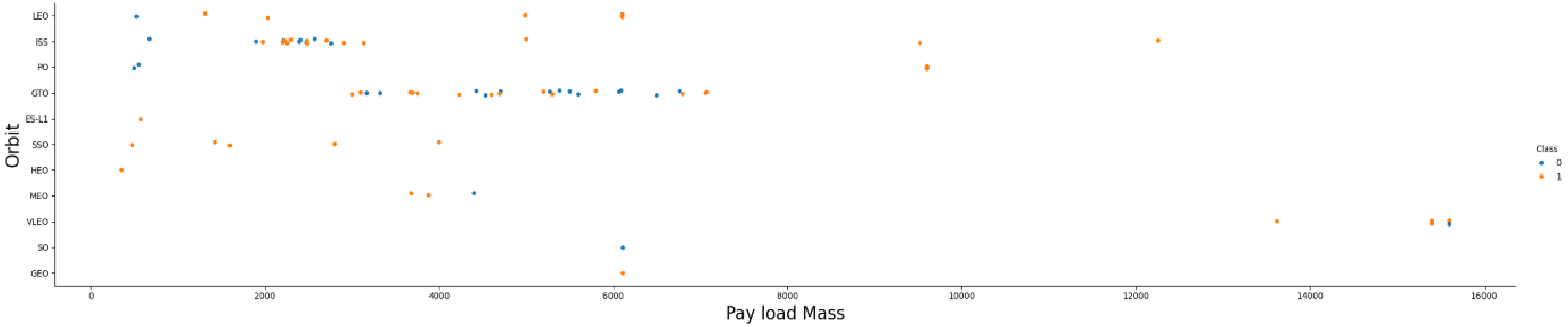
ES-L1, GEO, HEO and SSO had high sucess rate of 100%. On the other hand, SO hasn´t any kind of flight.

Flight Number vs. Orbit Type



You should see that in the LEO orbit the Success appears related to the number of flights; on the other hand, there seems to be no relationship between flight number when in GTO orbit.

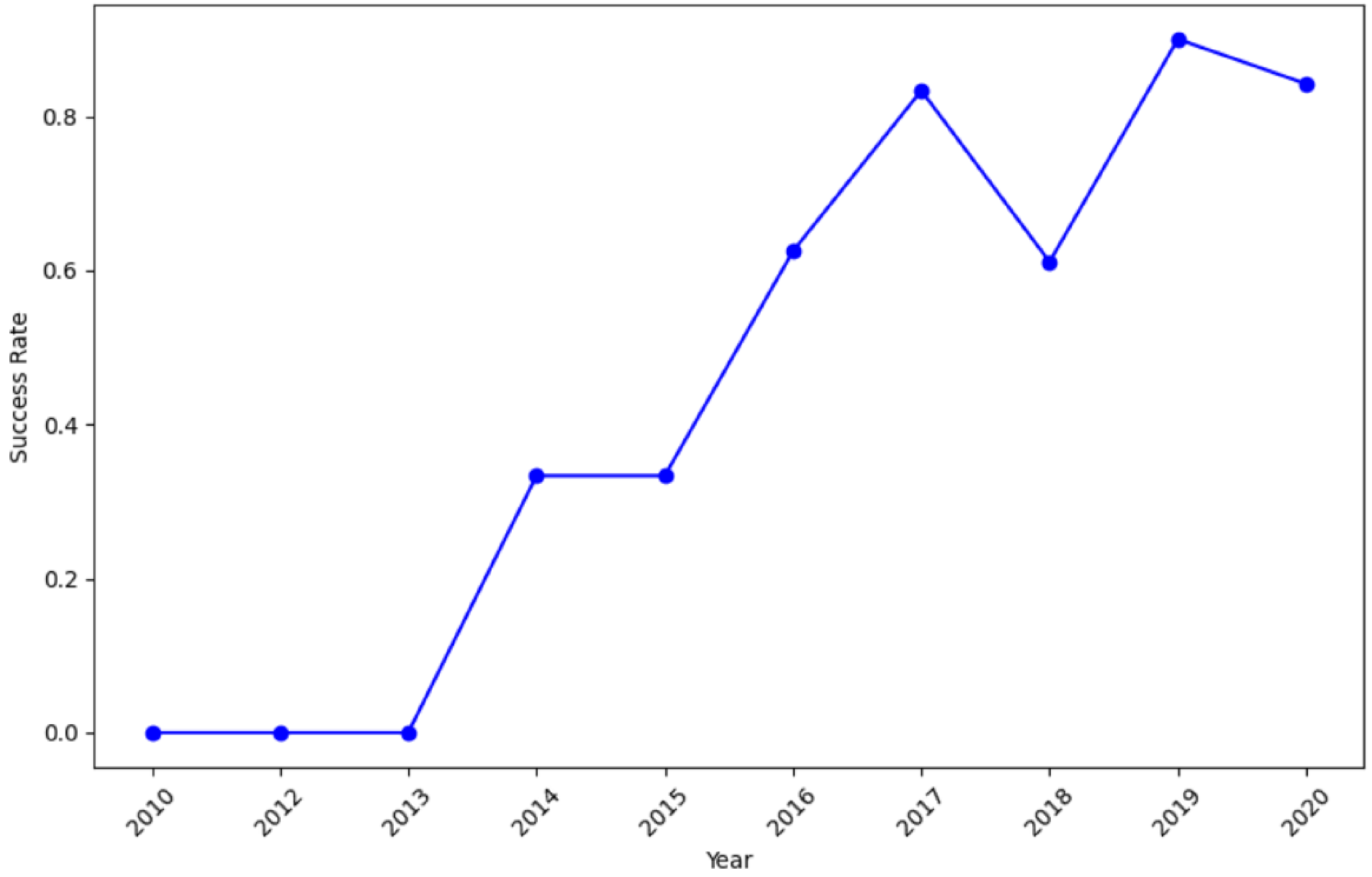
Payload vs. Orbit Type



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



You can observe that the success rate since 2013 kept increasing till 2017 (stable in 2014) and after 2015 it started increasing.

All Launch Site Names

```
%sql SELECT DISTINCT "Launch_Site" FROM SPACEXTBL;
```

```
 * 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 SPACEXTBL WHERE "Launch_Site" LIKE 'CCA%' LIMIT 5;
```

```
 * sqlite:///my_data1.db
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	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
2012-10-08	0:35:00	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 SELECT SUM("PAYLOAD_MASS__KG_") FROM SPACEXTBL WHERE "Customer" LIKE 'NASA (CRS)';
```

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

SUM(PAYLOAD_MASS__KG_)
45596

Average Payload Mass by F9 v1.1

```
%sql SELECT AVG("PAYLOAD_MASS__KG_") FROM SPACEXTBL WHERE "Booster_Version" LIKE 'F9 v1.1';
```

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

AVG(PAYLOAD_MASS__KG_)
2928.4

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

Successful Drone Ship Landing with Payload between 4000 and 6000

```
%sql select Booster_Version from SPACEXTABLE\
where Landing_Outcome = "Success (drone ship)" and PAYLOAD_MASS__KG_ between 4000 and 6000;
```

Total Number of Successful and Failure Mission Outcomes

```
%sql SELECT Mission_Outcome, COUNT(*) as count from SPACEXTABLE\
GROUP BY Mission_Outcome ORDER BY count DESC;
```

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

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

Boosters Carried Maximum Payload

```
%sql select Booster_Version from SPACEXTABLE\
where PAYLOAD_MASS__KG_ = (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

2015 Launch Records

```
%sql select substr(Date, 6, 2) as month, Landing_Outcome, Booster_Version, Launch_Site from SPACEXTABLE\
where substr(Date, 1, 4) = '2015' and Landing_Outcome = 'Failure (drone ship)'
```

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

month	Landing_Outcome	Booster_Version	Launch_Site
01	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
04	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

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

```
%sql select Landing_Outcome, count(*) as count_outcomes \
from SPACEXTABLE\
where Date between '2010-06-04' and '2017-03-20'\
group by Landing_Outcome\
order by count_outcomes Desc;
```

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

Landing_Outcome	count_outcomes
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

Section 3

Launch Sites Proximities Analysis

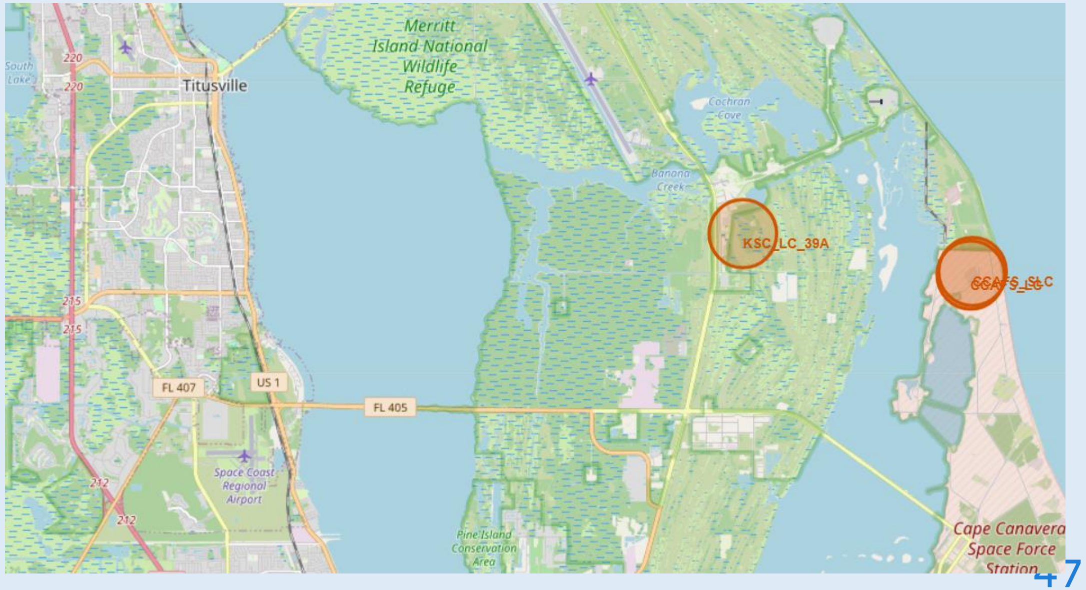
Interactive map Launches sites with Folium

There are four launches sites called:

- CCAFS LC – 40 (Florida)
- CCAFS SLC – 40 (Florida)
- KSC LC – 39A (Florida)
- VAFB SLC – 4E (California)

They are close proximity to coast and close to Major Highway and Railway for logistic purposes. But they are far from dense human habitats loke cities.

Each launchsite has launch outcomes to see the success rate.



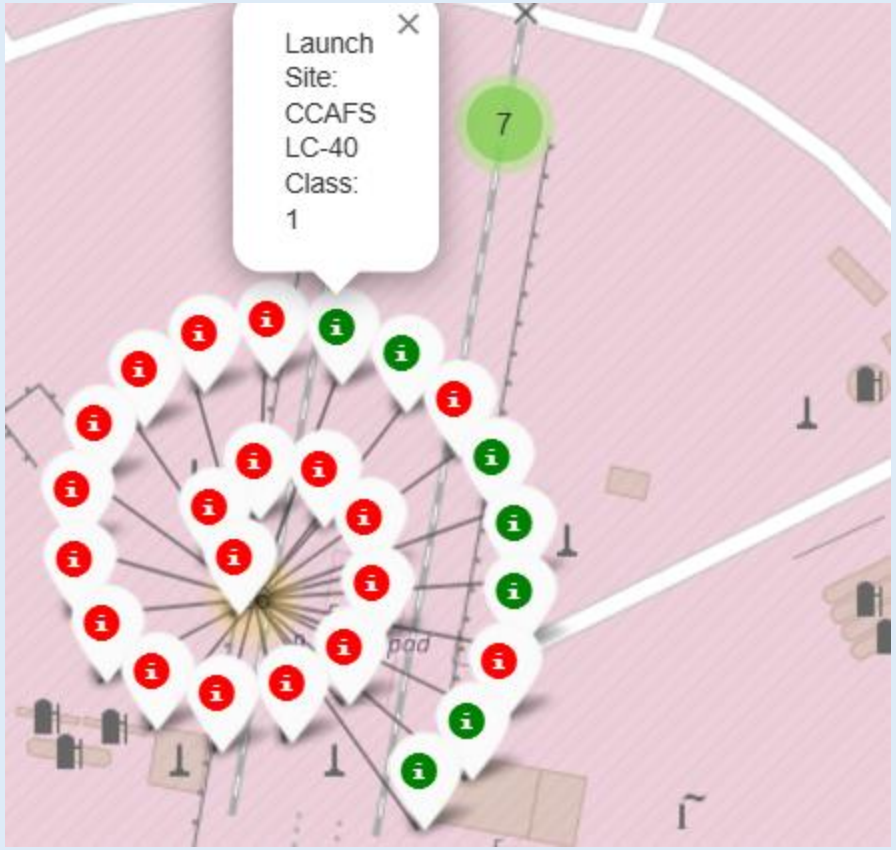
Interactive map color labeled launch outcomes with Folium

There are the launches outcomes of CCAFS LC – 40 which are labeled by color:

- Green: Class1, success.
- Red: Class 0, not success.

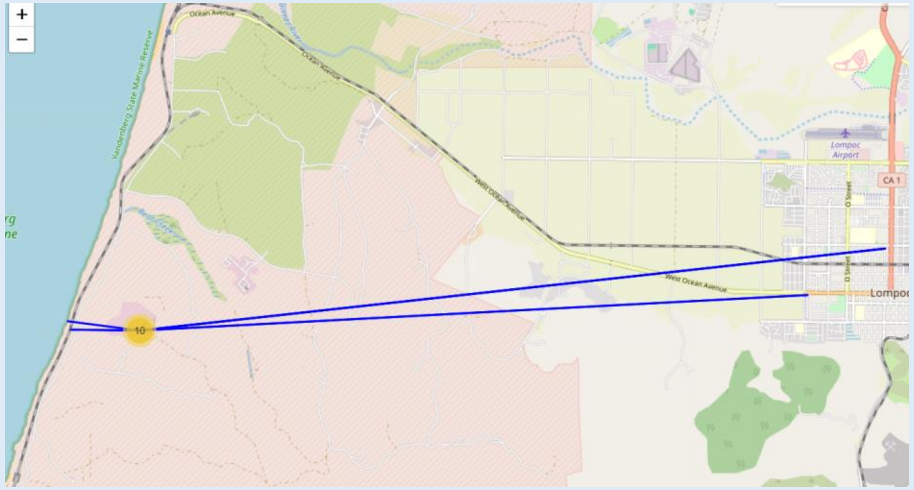
It success rate was 40%.

On the other hand, KSC LC 39 A and VAFR SLC 4 has a rate of 77%.



Interactive map distances between launch sites and its proximities

This is an example about the distances calculated between the launch sites VAFB_SLC_4E and its proximities like coastline, railway, highway and city.
It can be seen that this launch site is close to coastline and railway for logistic purposes and far from dense human habitats like cities.

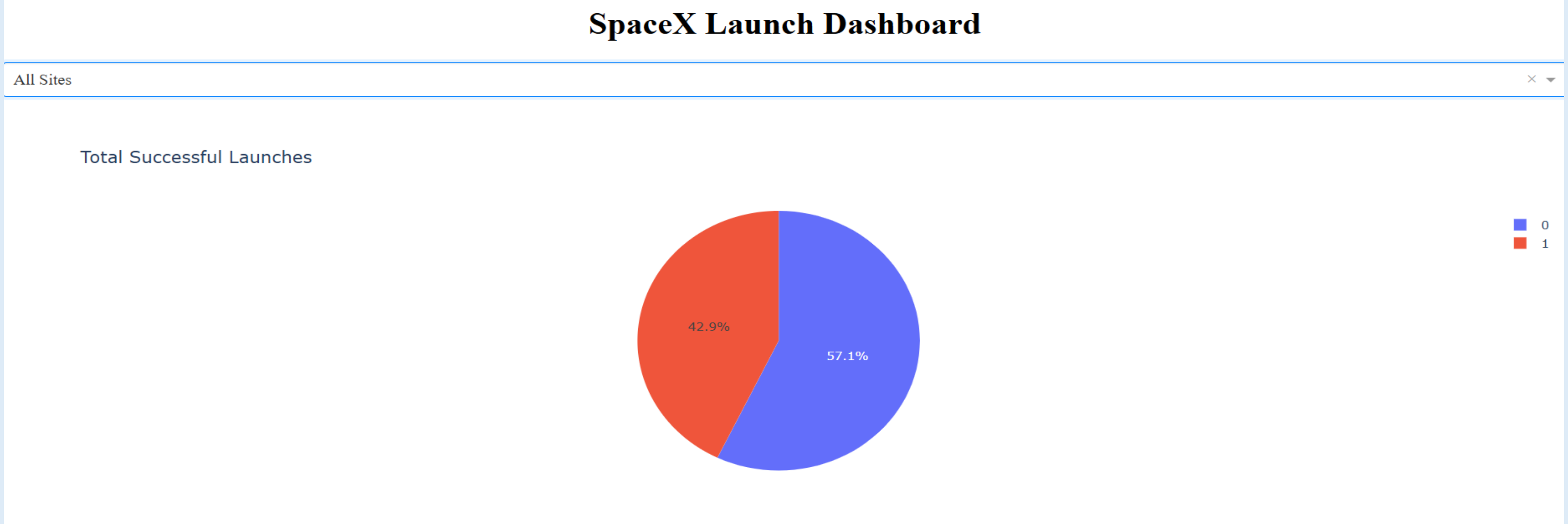


	Location	Distance to VAFB_SLC_4E
0	Coastline	1.357870
1	Railway	1.275449
2	Highway	12.446555
3	City	13.956576

Section 4

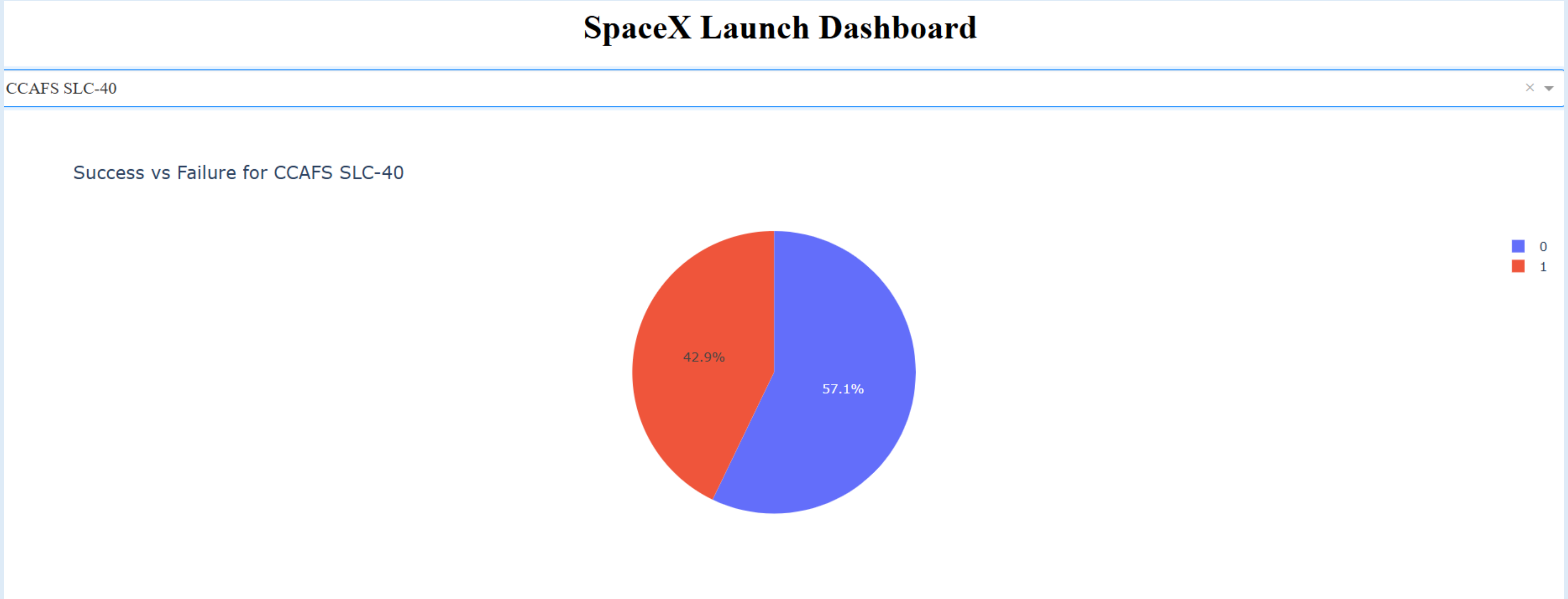
Build a Dashboard with Plotly Dash

Dashboard Total Successful Launches



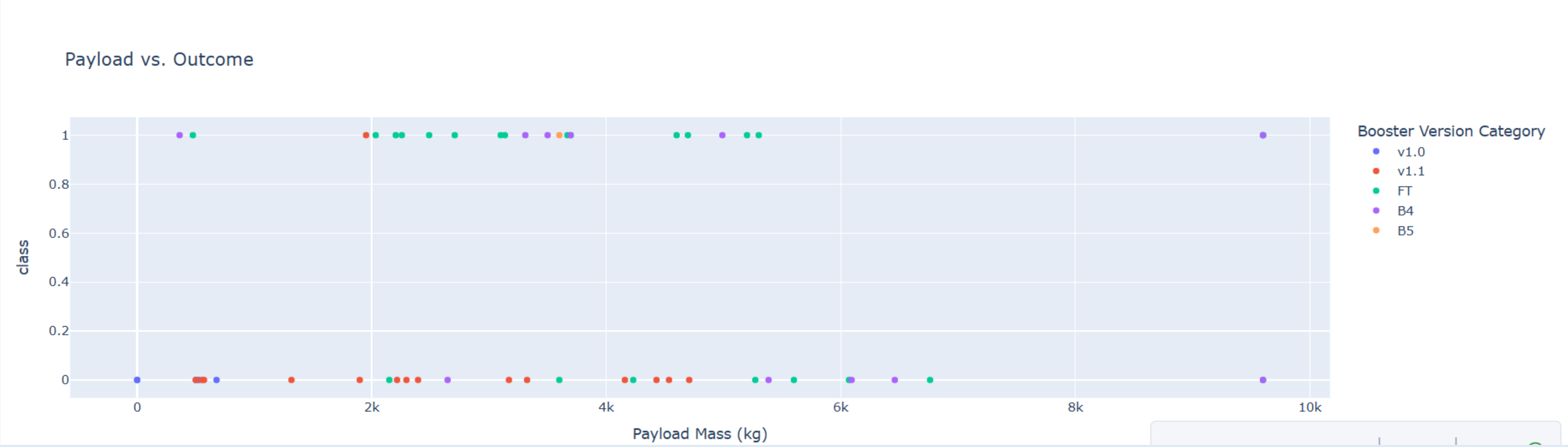
As we can see, by analyzing the launch success rate, we observe that globally, across all sites, the percentage of failed launches was higher than the percentage of successful launches, specifically 57.1% vs 42.9%.

Dashboard Launch Site With Highest Successful Launches



CCAFS SLC-40 has the highest percentage of successful launches. This may be because it is the launch site with the highest number of flights, and as previously observed, as the number of flights increases, the probability of a successful launch also increases.

Dashboard Payload vs Launch outcome

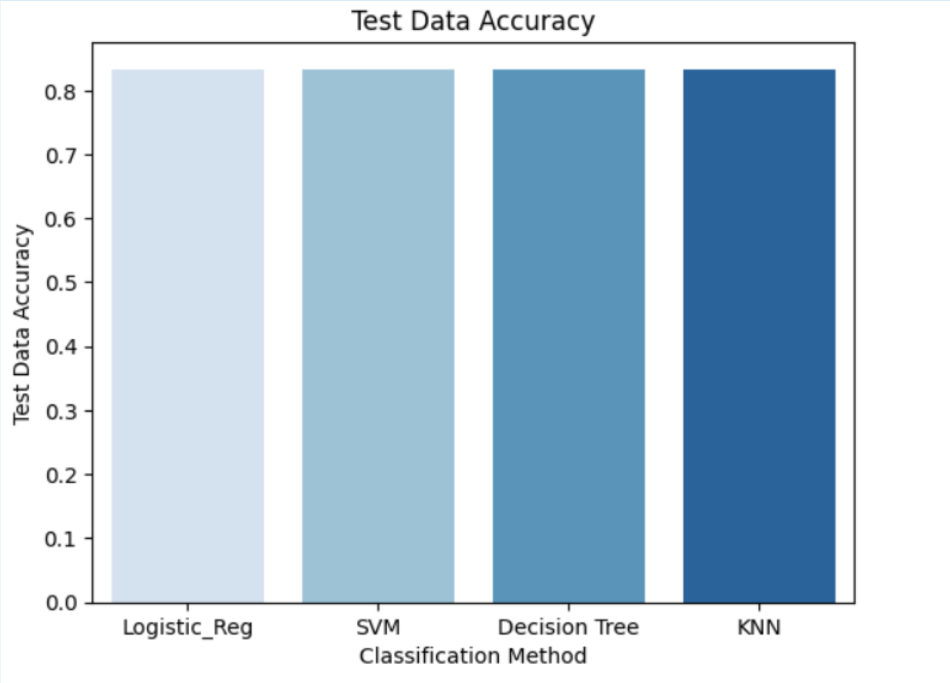


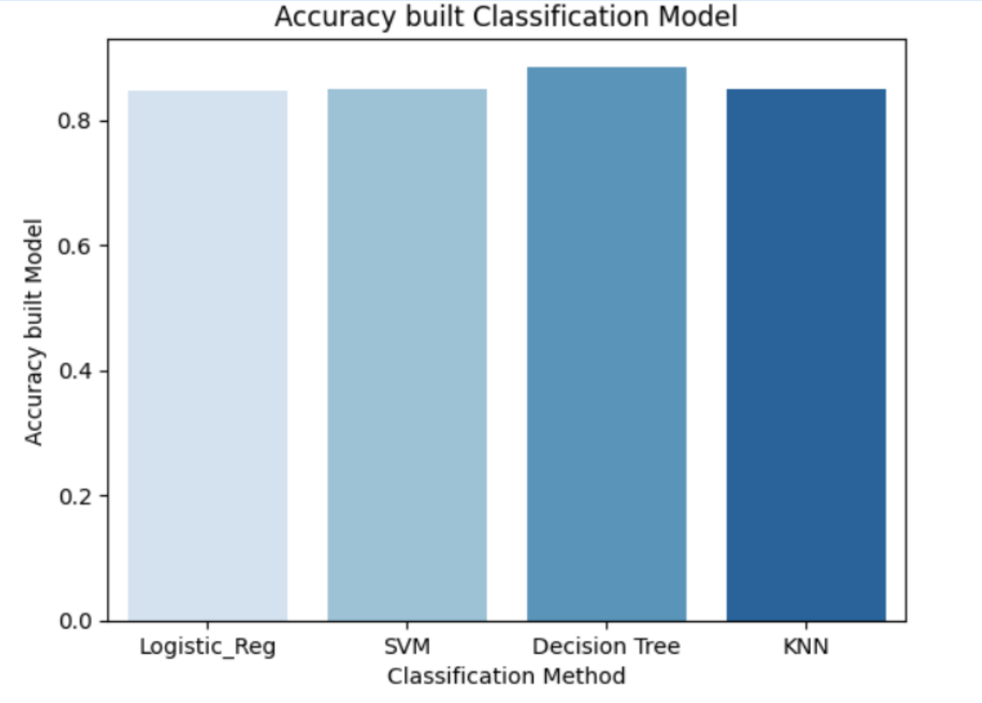
The best Booster Version Category was FT with a Payload Mass in a range of 2000K and 4000k.

Section 5

Predictive Analysis (Classification)

Classification Accuracy



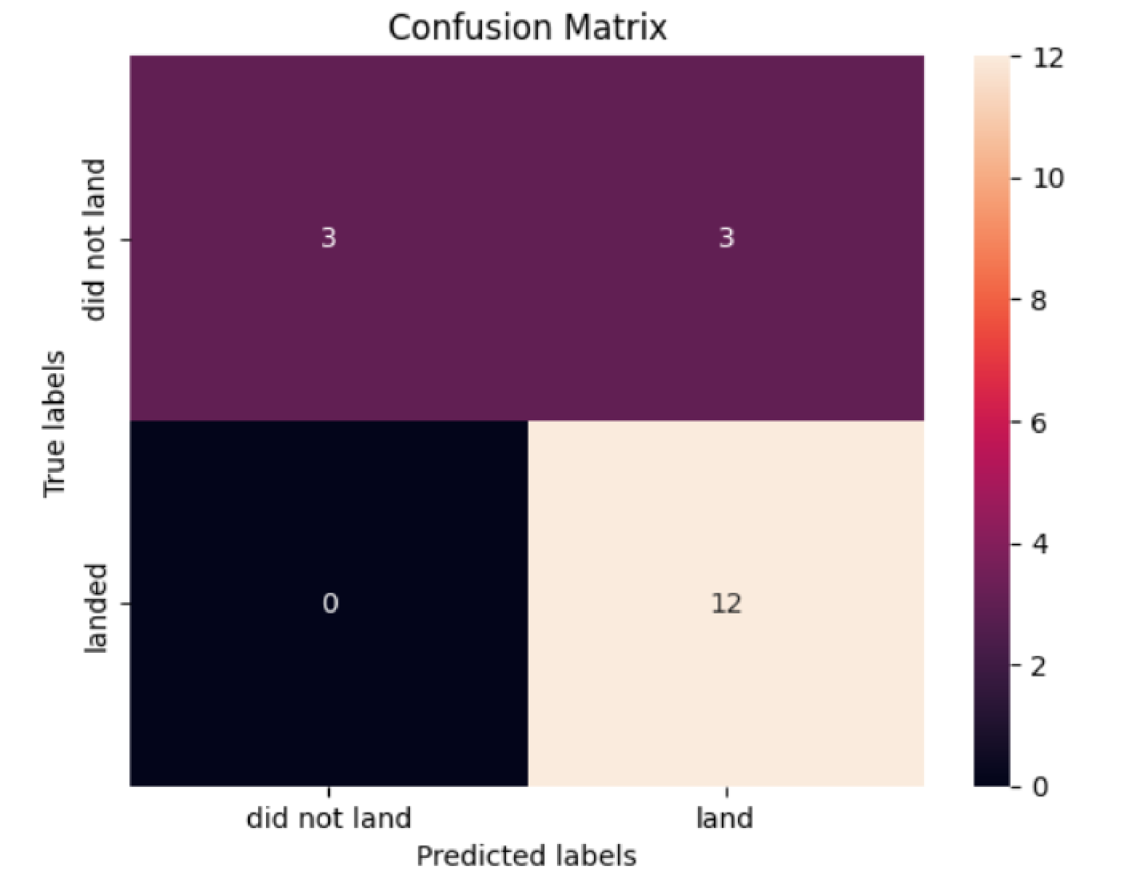


Decision Tree has the highest accuracy built model of 88% and the others classification models have the same accuracy built model of 84%. In all the methods, it has been used GirdSearch CV to find the best hyperparameters to train the models.
Conversely, all the classification methods have the same test data accuracy of 83%.

Confusion Matrix

All the models can distinguish between the different classes. But the problem is false positives:

- True positives: 12
- False positives: 3
- True negatives: 3
- False negatives: 0



Conclusions

1. Success rates launches for SpaceX has been increasing with time and it looks like soon they will reach the target.

2. Orbits ES-L1, GEO, HEO, SSO have the highest success rates.

3. KSC LC-39A had the highest launch success rate, although it was observed that with high payloads (>5000 kg), the launches failed.

4. In general, the optimal payload ranges between 2000 kg and 4000 kg, and FT was the best booster.

5. The Decision Tree Classifier algorithm is the best predictive classification model for this dataset.

Appendix

1. Data Preprocessing

```python
# students get this
transform = preprocessing.StandardScaler()

X = transform.fit_transform(X)
```

2. GridSearchcv to find the best hyperparameters to train Logistic Regresion

```python
parameters ={'C':[0.01,0.1,1],
             'penalty':['l2'],
             'solver':['lbfgs']}
```

```python
parameters ={"C":[0.01,0.1,1],'penalty':['l2'], 'solver':['lbfgs']}# l1 lasso l2 ridge
lr=LogisticRegression()
logreg_cv = GridSearchCV(lr, parameters, cv=10)
logreg_cv.fit(X_train, Y_train)
```

```
GridSearchCV
best_estimator_:
LogisticRegression
LogisticRegression
```

We output the `GridSearchCV` object for logistic regression. We display the best parameters using the data attribute `best_params_` and the accuracy on the validation data using the data attribute `best_score_`.

```python
print("tuned hpyerparameters :(best parameters) ",logreg_cv.best_params_)
print("accuracy :",logreg_cv.best_score_)
```

```
tuned hpyerparameters :(best parameters)  {'C': 0.01, 'penalty': 'l2', 'solver': 'lbfgs'}
accuracy : 0.8464285714285713
```

Appendix

3. This query display the month names, failure landing_outcomes in drone ships, booster version, launch_site for the months inyear 2015

```
%sql select substr(Date, 6, 2) as month, Landing_Outcome, Booster_Version, Launch_Site from SPACEXTABLE\
where substr(Date, 1, 4) = '2015' and Landing_Outcome = 'Failure (drone ship)'
```

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

month	Landing_Outcome	Booster_Version	Launch_Site
01	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
04	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

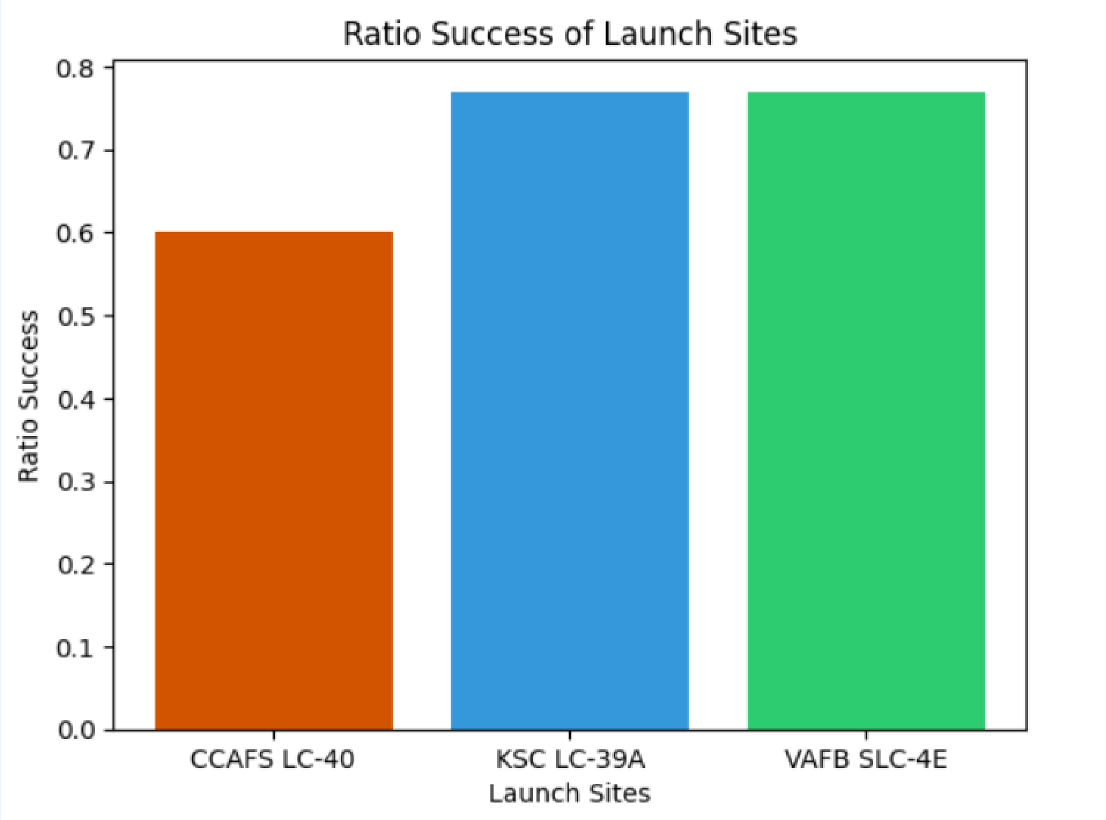
Appendix

4. Dataset with selected features that will be used in success prediction in the future model prediction.

	FlightNumber	PayloadMass	Orbit	LaunchSite	Flights	GridFins	Reused	Legs	LandingPad	Block	ReusedCount	Serial
0	1	6104.959412	LEO	CCAFS SLC 40	1	False	False	False	NaN	1.0	0	B0003
1	2	525.000000	LEO	CCAFS SLC 40	1	False	False	False	NaN	1.0	0	B0005
2	3	677.000000	ISS	CCAFS SLC 40	1	False	False	False	NaN	1.0	0	B0007
3	4	500.000000	PO	VAFB SLC 4E	1	False	False	False	NaN	1.0	0	B1003
4	5	3170.000000	GTO	CCAFS SLC 40	1	False	False	False	NaN	1.0	0	B1004

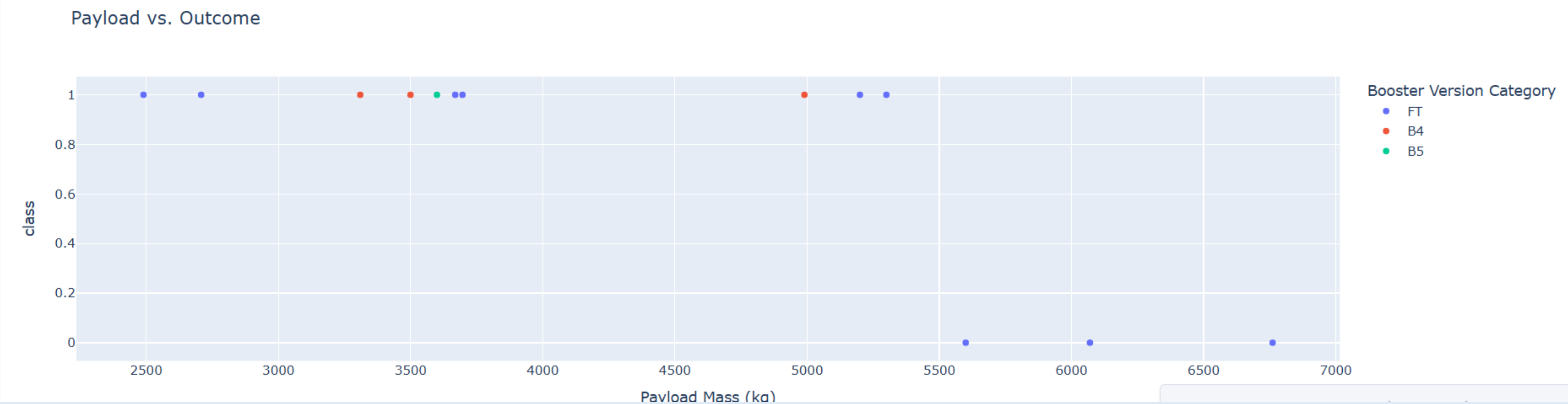
Appendix

5. Ratio Success of launch sites chart



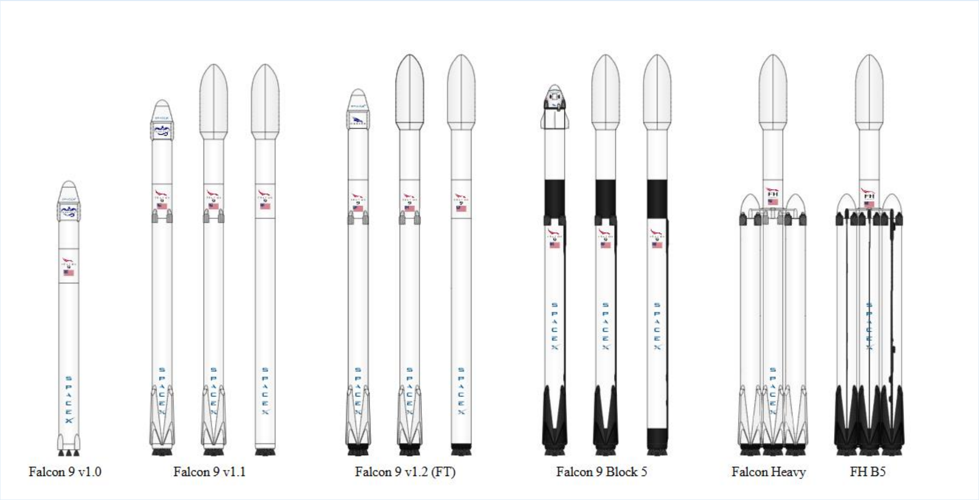
Appendix

6. Payload vs Outcome with different kind of booster version in KSC LC 39A



Appendix

6. Different kind of Falcon rockets and a tables with information about Falcon launches



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)

Thank you!