IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Youssef Moustaid
05/08/2025

Outline

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

Executive Summary

- Summary of methodologies
- Summary of all results

Introduction

- Project background and context
- Problems you want to find answers

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
    - Data was collected through published data sets and the SpaceX api.
- Perform data wrangling
    - Data wrangling was done through multiple methods : one hot encoding, scalling, transforming types...
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
    - To build the classification models, it is necessary to try multiple hyperparamaters to find the best ones, and that is done using cross validation and grid search.

Data Collection – SpaceX API

- The API used is: https://api.spacexdata.com
- It has many endpoints such as payloads, launchpads , rockets, cores, launches and much more. Here is the result after multiple endpoint calls and merging.
  - The API provides data about many types of rocket launches done by SpaceX, the data is therefore filtered to include only Falcon 9 launches.
  - Every missing value in the data is replaced the mean the column that the missing value belongs to.

```
[86]: # Hint data['BoosterVersion']!='Falcon 1'
      data_falcon9 = df[df['BoosterVersion'] != 'Falcon 1']
      data_falcon9.head()
```

[86]:

	FlightNumber	Date	BoosterVersion	PayloadMass	Orbit	LaunchSite	Outcome	Flights	GridFins	Reused	Legs	LandingPad	Block	ReusedCount	Serial	Longitude	Latitude
4	6	2010-06-04	Falcon 9	NaN	LEO	CCSFS SLC 40	None None	1	False	False	False	None	1.0	0	B0003	-80.577366	28.561857
5	8	2012-05-22	Falcon 9	525.0	LEO	CCSFS SLC 40	None None	1	False	False	False	None	1.0	0	B0005	-80.577366	28.561857
6	10	2013-03-01	Falcon 9	677.0	ISS	CCSFS SLC 40	None None	1	False	False	False	None	1.0	0	B0007	-80.577366	28.561857
7	11	2013-09-29	Falcon 9	500.0	PO	VAFB SLC 4E	False Ocean	1	False	False	False	None	1.0	0	B1003	-120.610829	34.632093
8	12	2013-12-03	Falcon 9	3170.0	GTO	CCSFS SLC 40	None None	1	False	False	False	None	1.0	0	B1004	-80.577366	28.561857

Data Collection – Web Scrapping from Wikepidea

- The data is scraped from https://en.wikipedia.org/w/index.php?title=List_of_Falcon_9_and_Falcon_Heavy_launches&oldid=1027686922
- The website contains only the data about Falcon 9 launches.
- We end up with 121 rows or instances and 11 columns or features. The picture below shows the first few rows of the data:

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 Collection - Web Scrapping from Wikepidea

- Here is the CSV after the web scrapping process

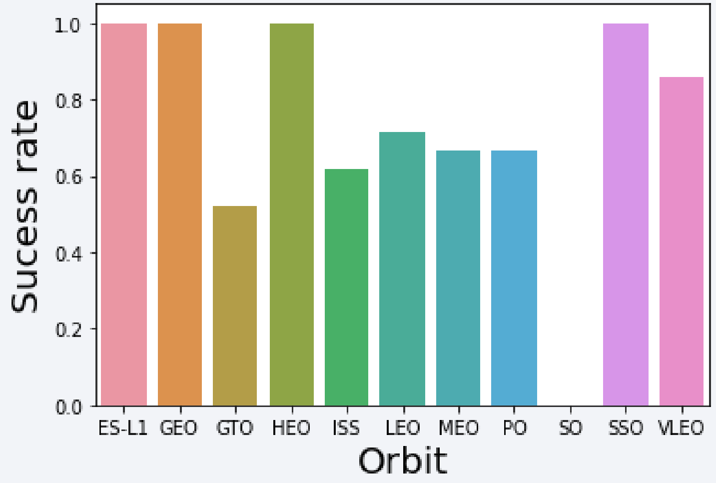
	Flight No.		Launch site	Payload	Payload mass	Orbit	Customer	Launch outcome		
1	1		CCAFS	Dragon Spacecraft Q…	0	LEO	SpaceX	Success		F9
2	2		CCAFS	Dragon	0	LEO	NASA	Success		F9
3	3		CCAFS	Dragon	525 kg	LEO	NASA	Success		F9
4	4		CCAFS	SpaceX CRS-1	4,700 kg	LEO	NASA	Success		F9
5	5		CCAFS	SpaceX CRS-2	4,877 kg	LEO	NASA	Success		F9
6	6		VAFB	CASSIOPE	500 kg	Polar orbit	MDA	Success		F
7	7		CCAFS	SES-8	3,170 kg	GTO	SES	Success		
8	8		CCAFS	Thaicom 6	3,325 kg	GTO	Thaicom	Success		
9	9		Cape Canaveral	SpaceX CRS-3	2,296 kg	LEO	NASA	Success		
10	10		Cape Canaveral	Orbcomm-OG2	1,316 kg	LEO	Orbcomm	Success		
11	11		Cape Canaveral	AsiaSat 8	4,535 kg	GTO	AsiaSat	Success		
12	12		Cape Canaveral	AsiaSat 6	4,428 kg	GTO	AsiaSat	Success		
13	13		Cape Canaveral	SpaceX CRS-4	2,216 kg	LEO	NASA	Success		
14	14		Cape Canaveral	SpaceX CRS-5	2,395 kg	LEO	NASA	Success		
15	15		Cape Canaveral	DSCOVR	570 kg	HEO	USAF	Success		
16	16		Cape Canaveral	ABS-3A	4,159 kg	GTO	ABS	Success		
17	17		Cape Canaveral	SpaceX CRS-6	1,898 kg	LEO	NASA	Success		
18	18		Cape Canaveral	TürkmenÄlem 52°E / …	4,707 kg	GTO		Success		
19	19		Cape Canaveral	SpaceX CRS-7	1,952 kg	LEO	NASA	Failure		
20	20		Cape Canaveral	Orbcomm-OG2	2,034 kg	LEO	Orbcomm	Success		
21	21		VAFB	Jason-3	553 kg	LEO	NASA	Success		
22	22		Cape Canaveral	SES-9	5,271 kg	GTO	SES	Success		
23	23		Cape Canaveral	SpaceX CRS-8	3,136 kg	LEO	NASA	Success		
24	24		Cape Canaveral	JCSAT-14	4,696 kg	GTO	SKY Perfect JSAT Gr…	Success		
25	25		Cape Canaveral	Thaicom 8	3,100 kg	GTO	Thaicom	Success		
26	26		Cape Canaveral	ABS-2A	3,600 kg	GTO	ABS	Success		

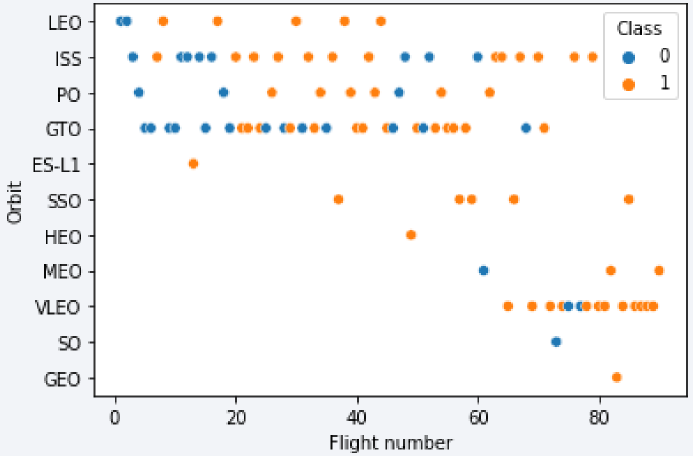
Data Wrangling

- The data is later processed so that there are no missing entries and categorical features are encoded using one-hot encoding.
- A column called ‘Class’ is was added to the data frame. The column ‘Class’ contains 0 if a given launch is failed and 1 if it is successful.
- In the end, we end up with 90 rows or instances and 83 columns or features.

EDA with Data Visualization

- Pandas and NumPy
  - Functions from the Pandas and NumPy libraries are used to derive basic information about the data collected, which includes:
    - The number of launches on each launch site
    - The number of occurrence of each orbit
    - The number and occurrence of each mission outcome
- SQL
  - The data is queried using SQL to answer several questions about the data such as:
    - The names of the unique launch sites in the space mission
    - The total payload mass carried by boosters launched by NASA (CRS)
    - The average payload mass carried by booster version F9 v1.1

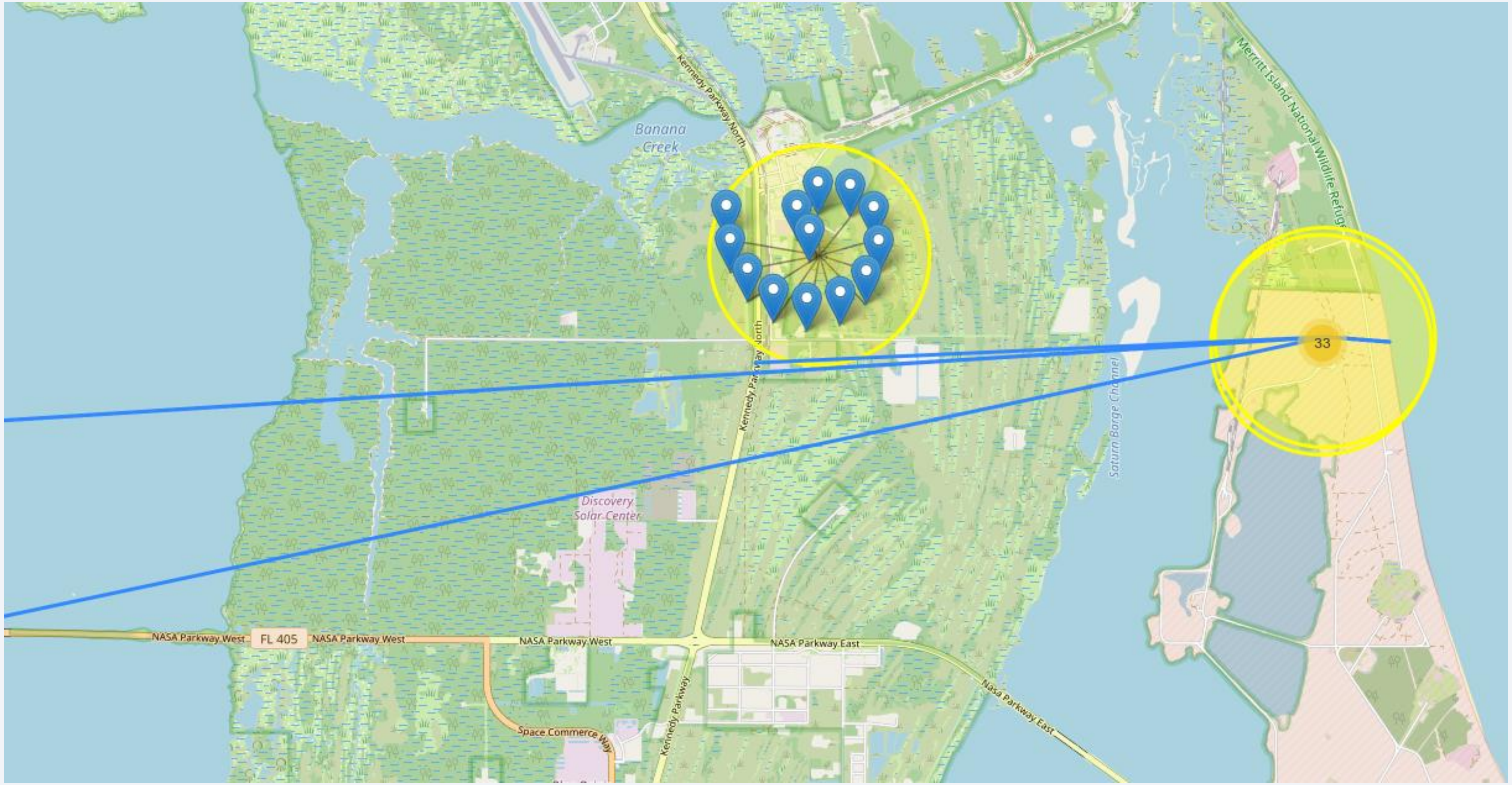




Build an Interactive Map with Folium

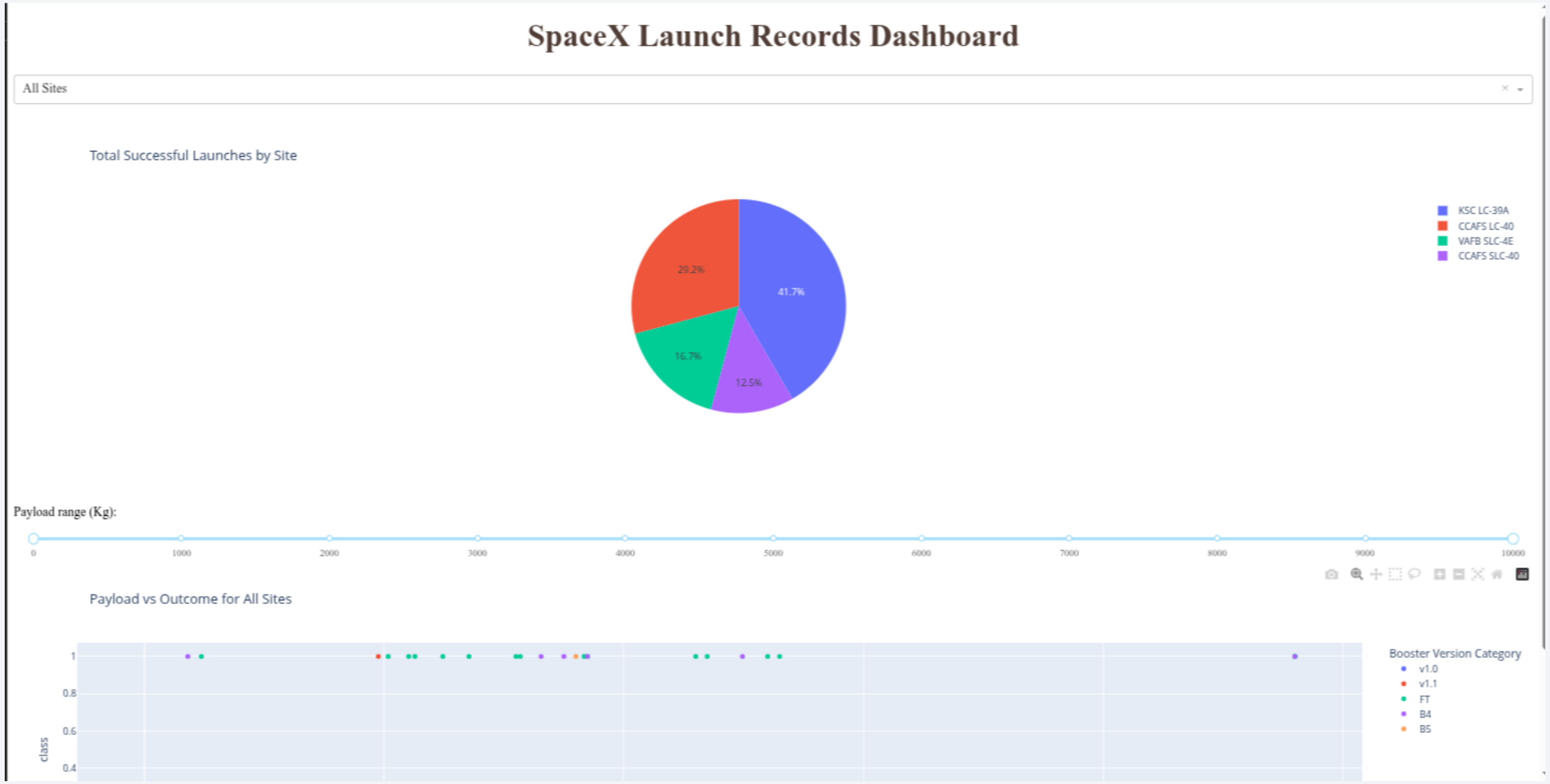
- Matplotlib and Seaborn
  - Functions from the Matplotlib and Seaborn libraries are used to visualize the data throughscatterplots, bar charts, and line charts.
  - The plots and charts are used to understand more about the relationships between severalfeatures, such as:
    - The relationship between flight number and launch site
    - The relationship between payload mass and launch site
    - The relationship between success rate and orbit type
- Folium
  - Functions from the Folium libraries are used to visualize the data through interactive maps.
  - The Folium library is used to:
    - Mark all launch sites on a map
    - Mark the succeeded launches and failed launches for each site on the map
    - Mark the distances between a launch site to its proximities such as the nearest city, railway, orhighway

Build an Interactive Map with Folium



Build a Dashboard with Plotly Dash

- Dash
  - Functions from Dash are used to generate an interactive site where we can toggle the inputusing a dropdown menu and a range slider.
  - Using a pie chart and a scatterplot, the interactive site shows:
    - The total success launches from each launch site
    - The correlation between payload mass and mission outcome (success or failure) for each launchsite



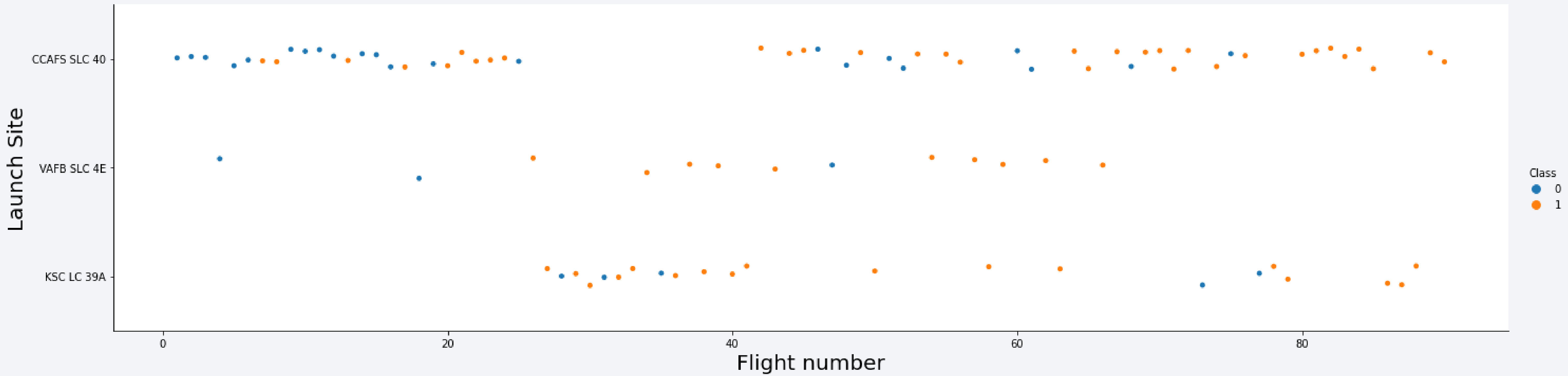
Predictive Analysis (Classification)

- Functions from the Scikit-learn library are used to create our machine learningmodels.
- The machine learning prediction phase include the following steps:
  - Standardizing the data
  - Splitting the data into training and test data
  - Creating machine learning models, which include:
    - Logistic regression
    - Support vector machine (SVM)
    - Decision tree
    - K nearest neighbors (KNN)
  - Fit the models on the training set
  - Find the best combination of hyperparameters for each model
  - Evaluate the models based on their accuracy scores and confusion matrix

Section 2

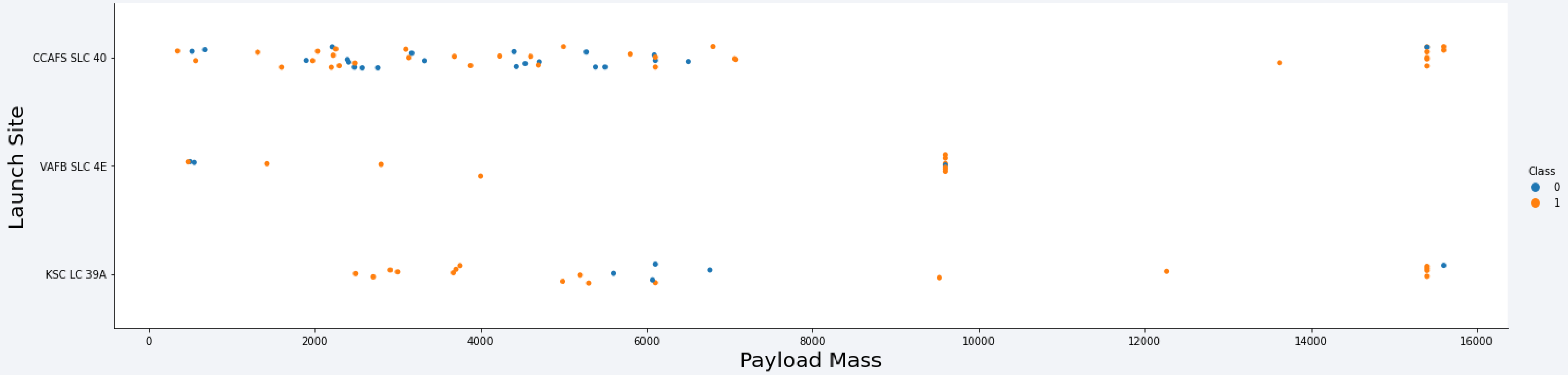
Insights drawn from EDA

Flight Number vs. Launch Site



- Show a scatter plot of Flight Number vs. Launch Site
- Show the screenshot of the scatter plot with explanations

Payload vs. Launch Site



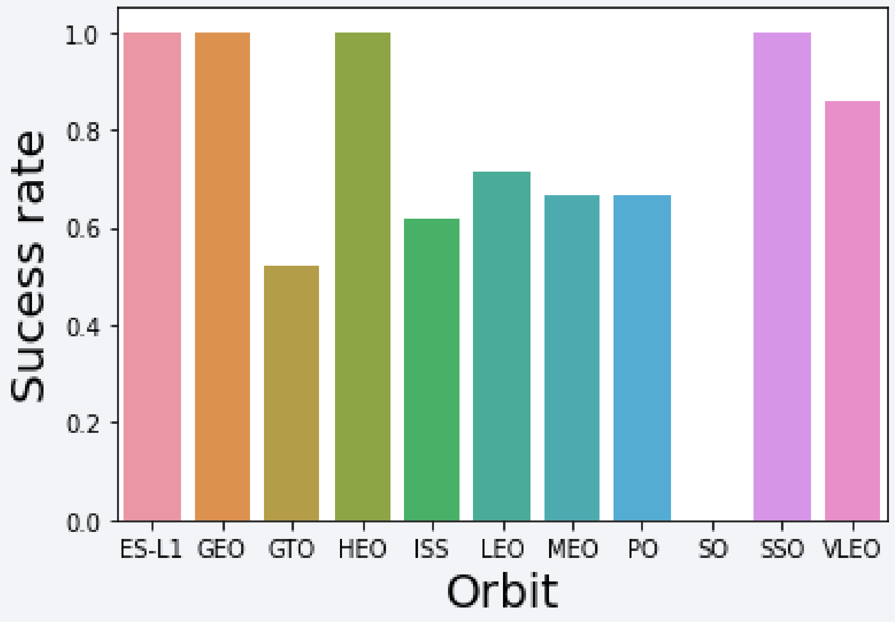
- Show a scatter plot of Payload vs. Launch Site

- Show the screenshot of the scatter plot with explanations

Success Rate vs. Orbit Type

- Show a bar chart for the success rate of each orbit type

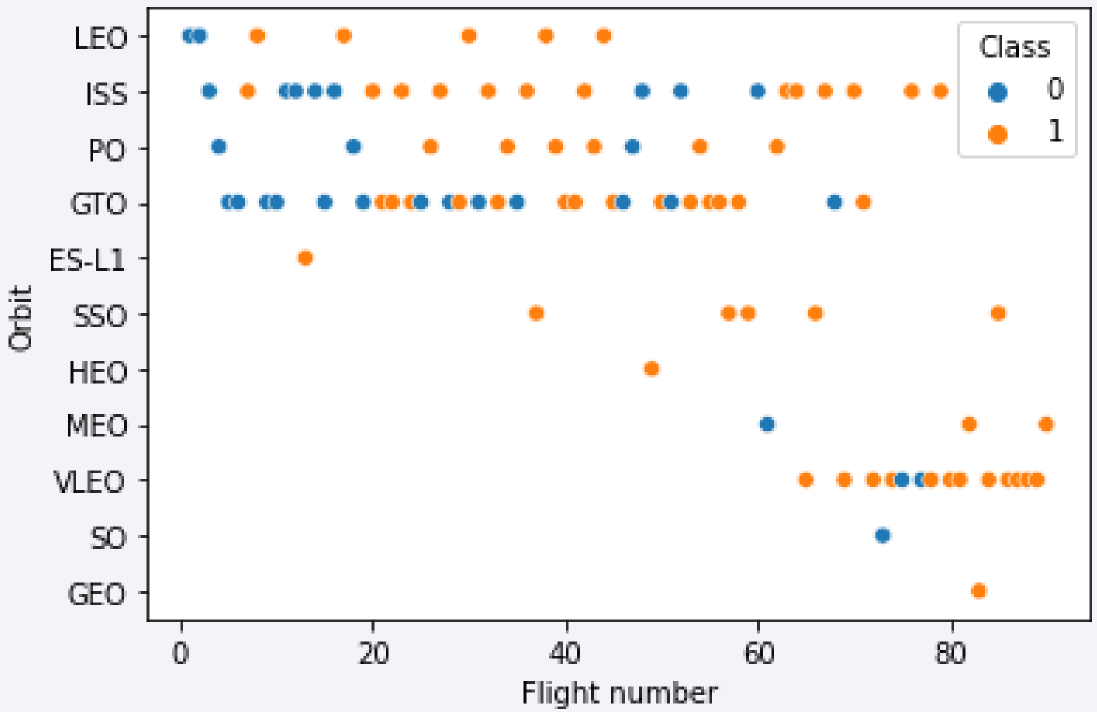
- Show the screenshot of the scatter plot with explanations



Flight Number vs. Orbit Type

- Show a scatter point of Flight number vs. Orbit type

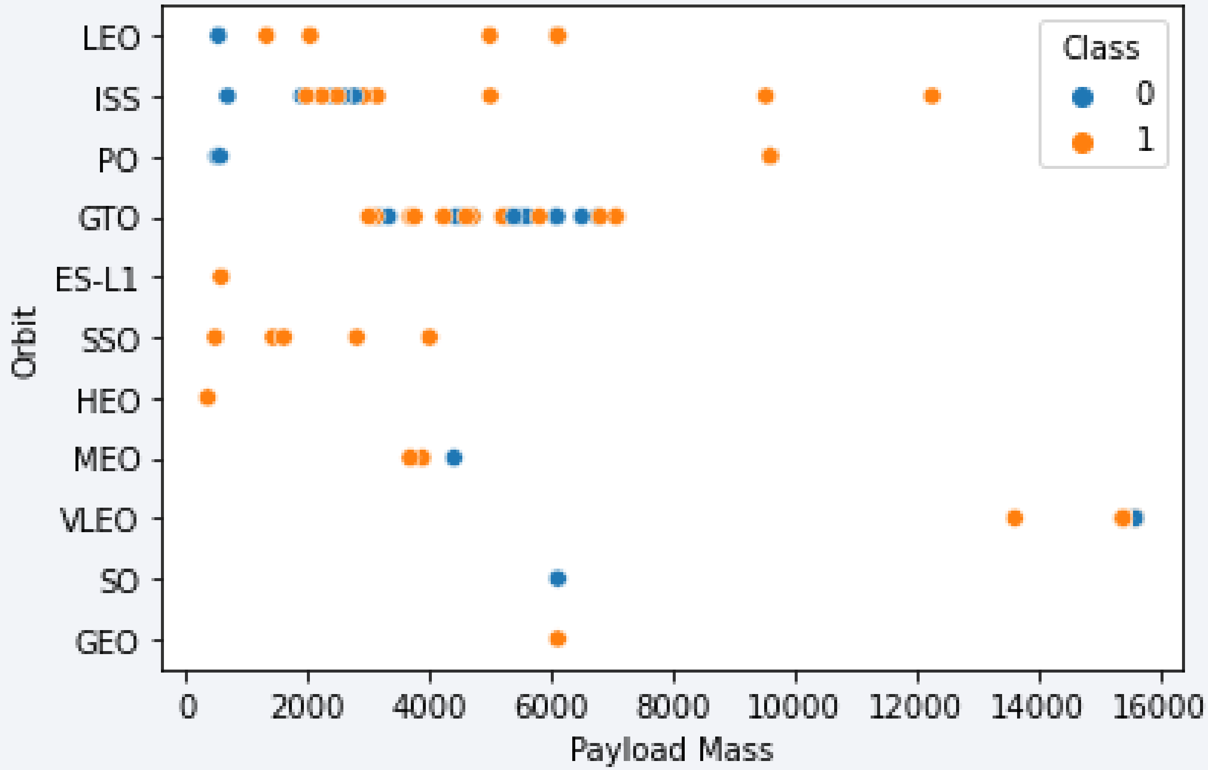
- Show the screenshot of the scatter plot with explanations



Payload vs. Orbit Type

- Show a scatter point of payload vs. orbit type

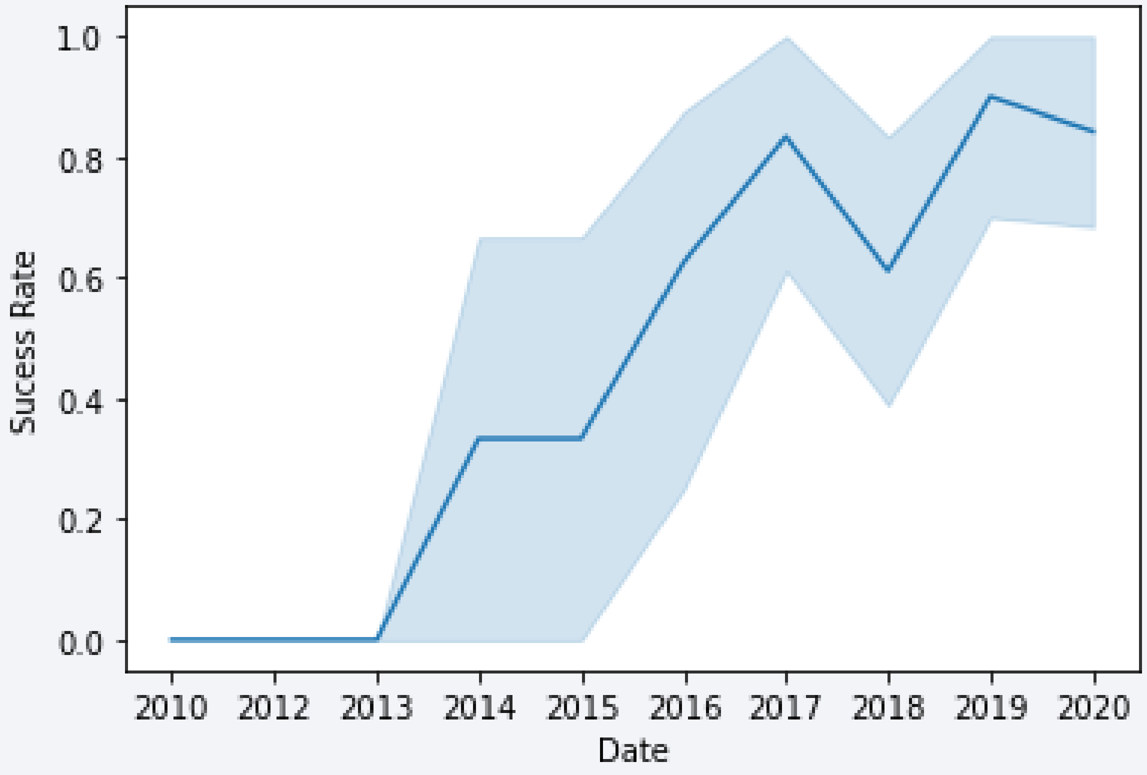
- Show the screenshot of the scatter plot with explanations



Launch Success Yearly Trend

- Show a line chart of yearly average success rate

- Show the screenshot of the scatter plot with explanations



All Launch Site Names

```
[13]: %sql SELECT DISTINCT Launch_Site from SPACEXTABLE;
 * sqlite:///my_data1.db
Done.
```

[13]:

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

Launch Site Names Begin with 'CCA'

```
[15]: %sql SELECT * FROM SPACEXTABLE WHERE Launch_Site LIKE 'CCA%' LIMIT 5 ;
 * sqlite:///my_data1.db
Done.
```

[15]:

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

Task 3

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

```
[20]: %sql SELECT sum(PAYLOAD_MASS__KG_) AS 'Life time payload by NASA(in Kg)' FROM SPACEXTABLE WHERE Customer LIKE 'NASA (CRS)' ;

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

[20]:

Life time payload by NASA(in Kg)
45596

Average Payload Mass by F9 v1.1

Task 4

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

```
[21]: %sql SELECT avg(PAYLOAD_MASS__KG_) AS 'Average payload mass by booster version F9 v1.1 (in Kg)' FROM SPACEXTABLE WHERE Booster_Versi
```

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

[21]:

Average payload mass by booster version F9 v1.1 (in Kg)
2928.4

First Successful Ground Landing Date

Task 5

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

Hint:Use min function

```
[25]: %sql SELECT Date FROM SPACEXTABLE WHERE Landing_Outcome = 'Success (ground pad)' ORDER BY Date ASC LIMIT 1 ;

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

[25]:

Date
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

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

[27]:
```
%sql SELECT DISTINCT(Booster_Version) FROM SPACEXTABLE WHERE Landing_Outcome = 'Success (drone ship)' AND PAYLOAD_MASS__KG_ BETWEEN
```

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

[27]:

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

Total Number of Successful and Failure Mission Outcomes

```
[31]: %%sql
SELECT
  CASE
    WHEN Landing_Outcome LIKE 'Success%' THEN 'Success'
    WHEN Landing_Outcome LIKE 'Failure%' OR Landing_Outcome = 'Failure' THEN 'Failure'
    ELSE 'Other'
  END AS Outcome_Category,
  COUNT(*) AS Total
FROM SPACEXTABLE
GROUP BY Outcome_Category;

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

[31]:

Outcome_Category	Total
Failure	10
Other	30
Success	61

Boosters Carried Maximum Payload

Task 8

List all the booster_versions that have carried the maximum payload mass, using a subquery with a suitable aggregate function.

[36]:
```
%%sql
SELECT DISTINCT Booster_Version, PAYLOAD_MASS__KG_
    FROM SPACEXTABLE
    WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTABLE ) ;
```

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

[36]:

Booster_Version	PAYLOAD_MASS__KG_
F9 B5 B1048.4	15600
F9 B5 B1049.4	15600
F9 B5 B1051.3	15600
F9 B5 B1056.4	15600
F9 B5 B1048.5	15600
F9 B5 B1051.4	15600
F9 B5 B1049.5	15600
F9 B5 B1060.2	15600
F9 B5 B1058.3	15600
F9 B5 B1051.6	15600
F9 B5 B1060.3	15600
F9 B5 B1049.7	15600

2015 Launch Records

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.

```
[38]: %%sql SELECT  substr(Date, 6,2) as month , Landing_Outcome
 FROM SPACEXTABLE
 WHERE Landing_Outcome LIKE '%Failure (drone ship)%'
  AND substr(Date, 0,5) = '2015';
```

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

[38]:

month	Landing_Outcome
01	Failure (drone ship)
04	Failure (drone ship)

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

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.

[39]:
```
%%sql SELECT COUNT(Landing_Outcome) as count_of_landing_outcome , Landing_Outcome , Date
   FROM SPACEXTABLE
   WHERE Date BETWEEN '2010-06-04' AND '2017-03-20'
   ORDER BY Date DESC ;



%%sql
SELECT
  Date,
  Landing_Outcome,
  COUNT(Landing_Outcome) AS count_of_landing_outcome
FROM SPACEXTABLE
WHERE Date BETWEEN '2010-06-04' AND '2017-03-20'
GROUP BY Date, Landing_Outcome
ORDER BY Date DESC;
```

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

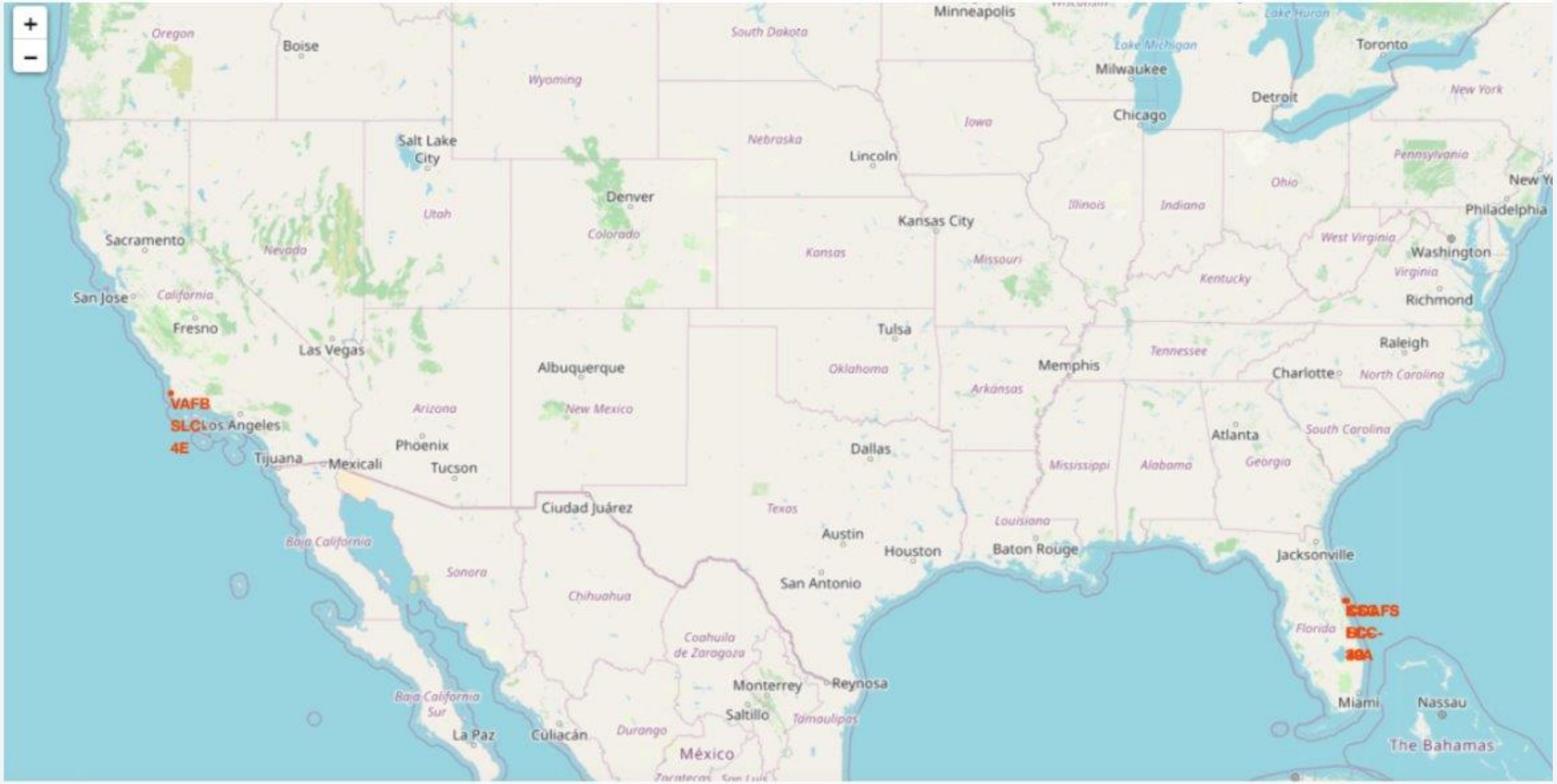
[39]:

count_of_landing_outcome	Landing_Outcome	Date
31	Failure (parachute)	2010-06-04

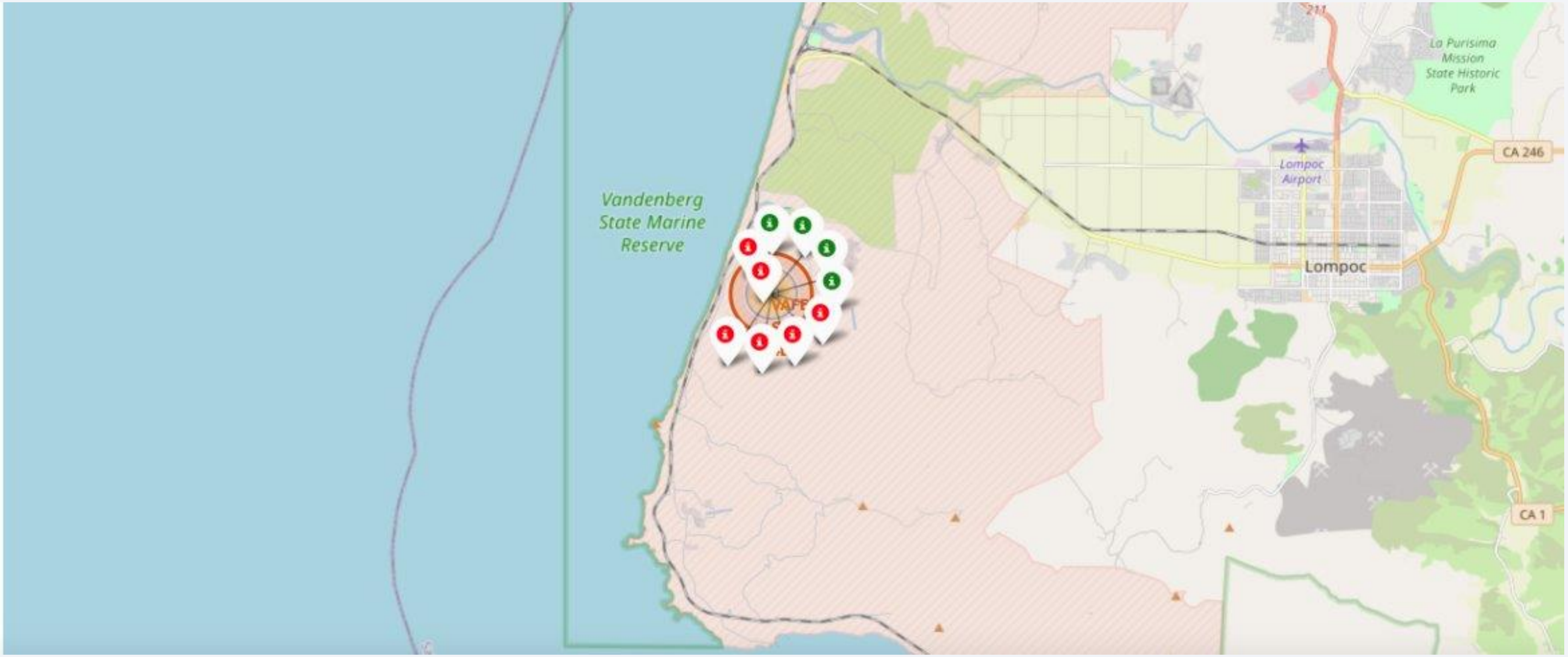
Section 3

Launch Sites Proximities Analysis

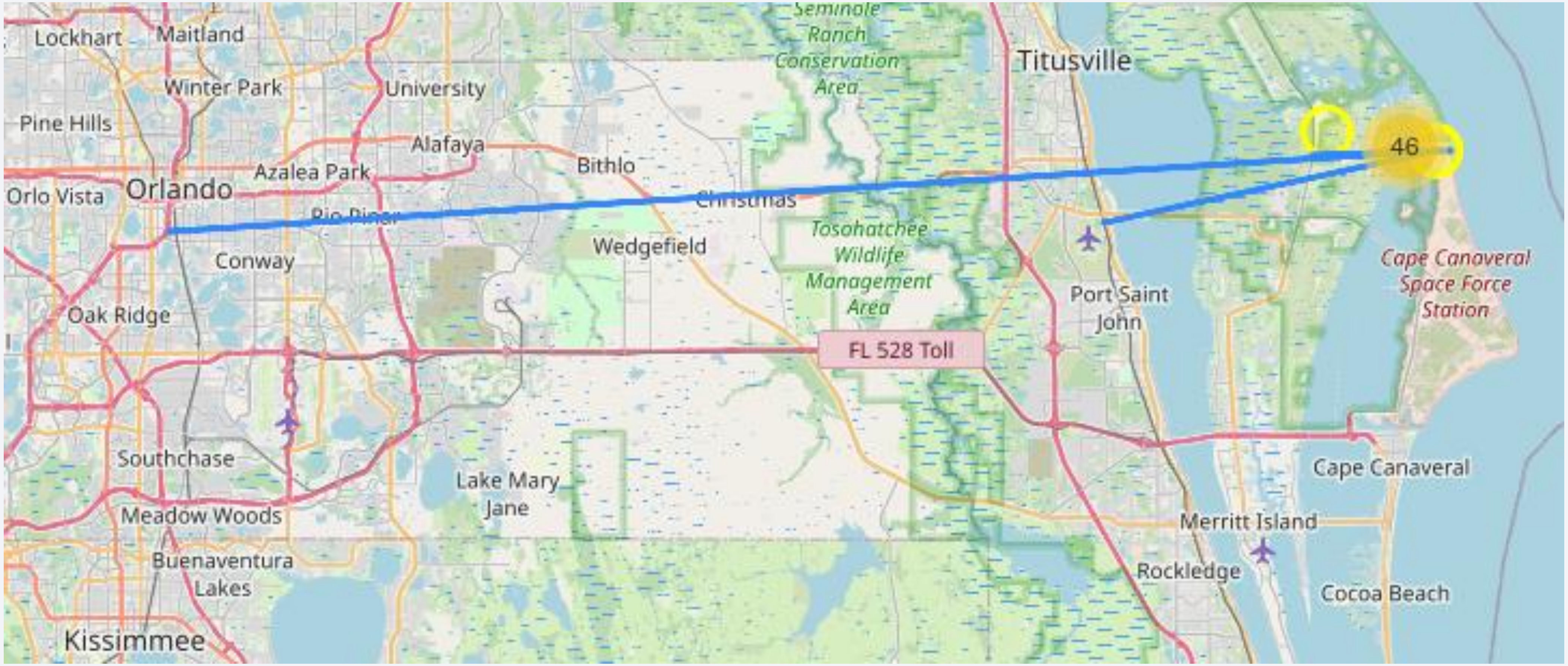
All Launch Sites on Map



Success launch sites



Approximation to nearest ciry, railway and highway



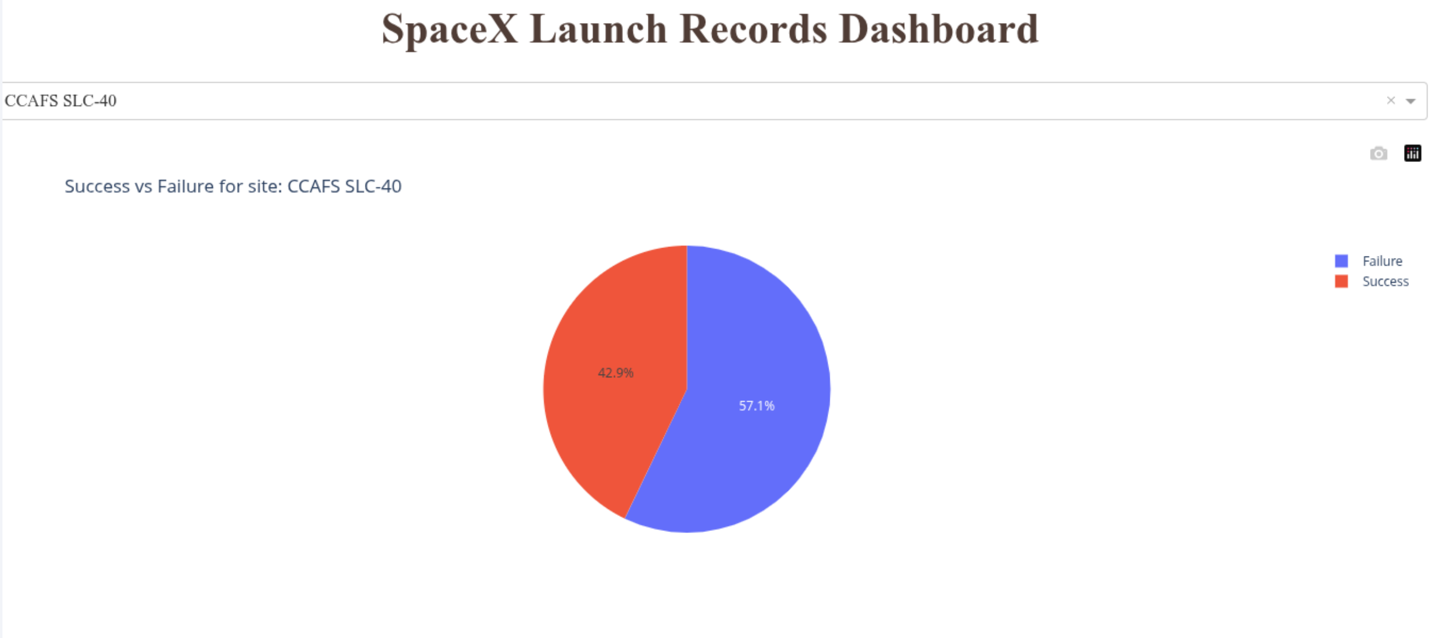
Section 4

Build a Dashboard with Plotly Dash

Dashboard

- As shown , we can clearly see that KSL LC-39 A has the highest succesull launches being 41.7%.

- It is also possible to select a launch site and have the pie chart updated



Success of laning to payload mass

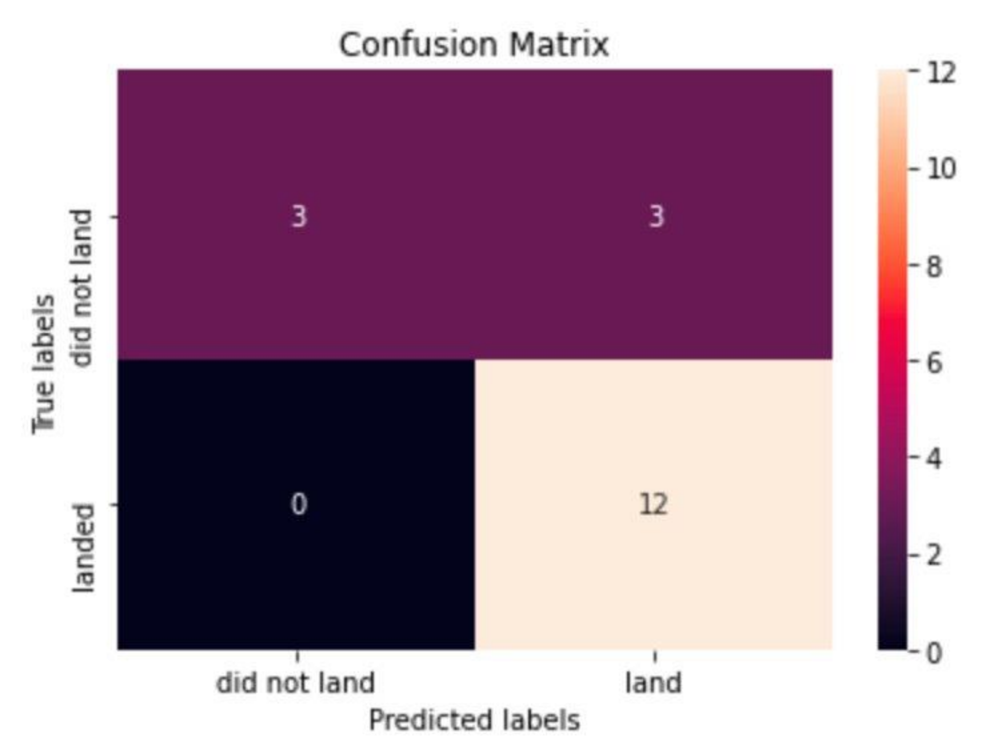
- For the payload selection to success rates, we can select a mass range and have the scatter plot updated in real time thanks to dash.

Section 5

Predictive Analysis (Classification)

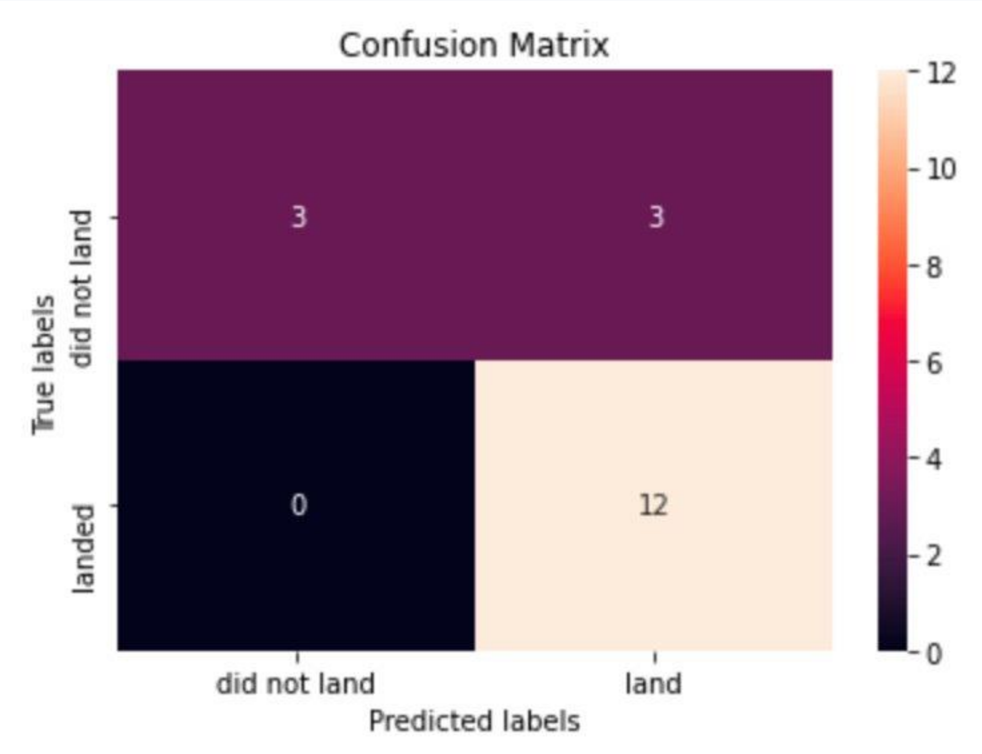
Classification Accuracy

- Logistic regression
- GridSearchCV best score: 0.8464285714285713
- Accuracy score on test set: 0.8333333333333334
- Confusion matrix:



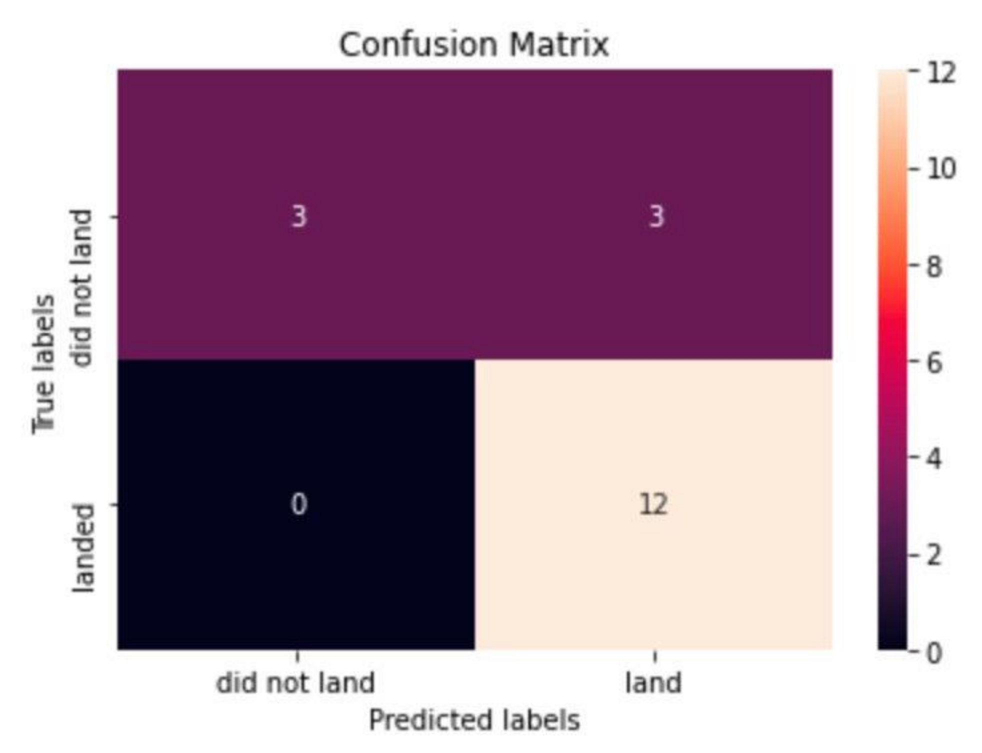
Classification Accuracy

- Support vector machine (SVM)
  - GridSearchCV best score: 0.8482142857142856
  - Accuracy score on test set: 0.8333333333333334
  - Confusion matrix:



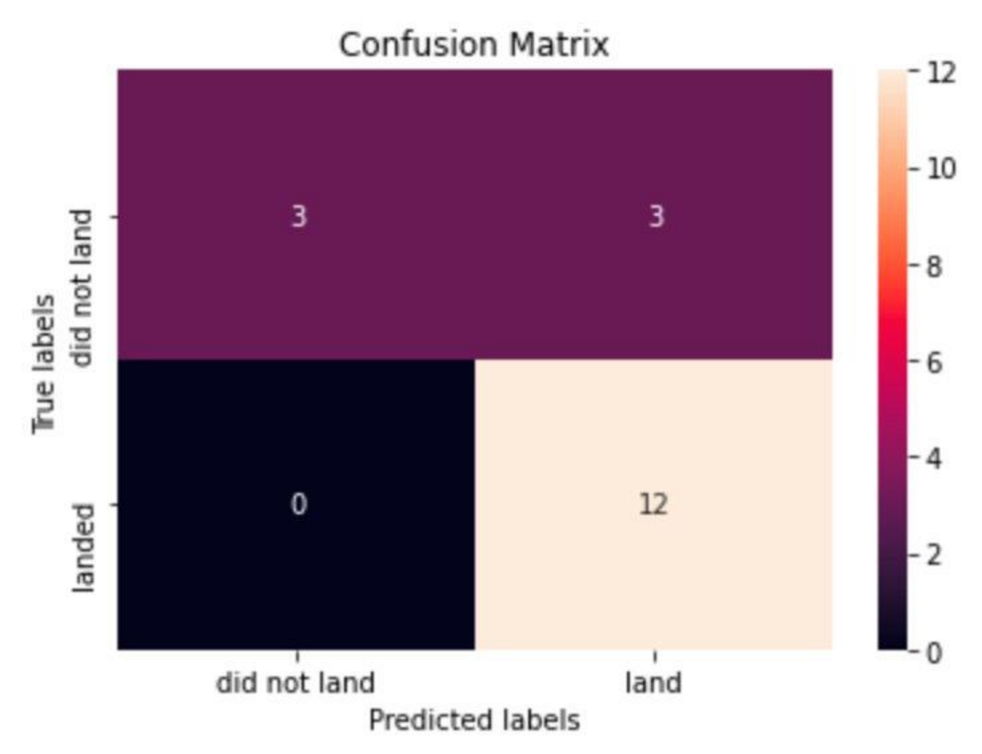
Classification Accuracy

- Decision tree
  - GridSearchCV best score: 0.8892857142857142
  - Accuracy score on test set: 0.8333333333333334
  - Confusion matrix:



Classification Accuracy

- K nearest neighbors (KNN)
  - GridSearchCV best score: 0.8482142857142858
  - Accuracy score on test set: 0.8333333333333334
  - Confusion matrix:



Compairason of the models

•After evaluating the performance of the four models using confusion matrices, we decided to rank them based on their **best scores from GridSearchCV**, as it provides a more objective comparison. The models are ranked below from best to worst according to their GridSearchCV best scores:

1. **Decision Tree** – Best Score: **0.889**
2. **K-Nearest Neighbors (KNN)** – Best Score: **0.848**
3. **Support Vector Machine (SVM)** – Best Score: **0.848**
4. **Logistic Regression** – Best Score: **0.846**

Conclusion

- This project focuses on predicting whether the **first stage of a Falcon 9 rocket** will successfully land, a factor that significantly influences **launch cost efficiency**.
- Key launch features—such as **payload mass**, **orbit type**, and other mission-specific variables—are analyzed to understand their impact on the landing outcome.
- To uncover meaningful patterns in historical launch data, we applied multiple **machine learning algorithms**. These models were trained to recognize correlations and trends that could predict future outcomes.
- Among the four algorithms tested, the **Decision Tree model** delivered the highest performance, making it the most effective approach for this classification task.

My Github repo

Thank you!