SpaceX Falcon 9 first stage Landing Prediction

Jennifer Silva

02-28-2024

OUTLINE

- Executive Summary
- Introduction
- Methodology
- Results
  - Visualization – Charts
  - Dashboard
- Discussion
  - Findings & Implications
- Conclusion
- Appendix

EXECUTIVE SUMMARY

SpaceX offers a very competitively priced rocket launch service, thanks to its ability to recover and reuse the first stage of the Falcon 9. If the first stage manages to land successfully after each mission, it can be reused in future launches saving money and resources. Therefore, it is important to:

- Predict the probability of landing success, as this affects the final cost of the service.

INTRODUCTION

SpaceX has gained global attention for a series of historic milestones.

Among those it stands out for being the only private company that has returned a spacecraft from low Earth orbit, something it achieved for the first time in December 2010.

SpaceX announces on its website the launches of Falcon 9 rockets at a cost of 62 million dollars, while other suppliers cost more than 165 million dollars each, this difference is due to the fact that the first stage can be reused, representing a 62% of savings.

- If we can determine that the first stage will land, we can determine the cost of a launch.
- At the same time, this information could be useful in case an alternative company wants to bid against SpaceX for the launch of a rocket.

METHODOLOGY

- Research Question:
  - Predict the probability of landing success, as this affects the final cost of the service.
- Data Collection:
  - SpaceX API
  - Web Scrapping
    - Falcon 9 launch records with BeautifulSoup:
      - We extract an HTML table of Falcon 9 launch records from the Wikipedia website.
      - We parse the tables and convert them into a Pandas dataframe.
- Data Wrangling
- EDA: Data Analysis and Visualization through:
  - SQL
  - Pandas
  - Folium
  - Seaborn
  - Matplotlib.

METHODOLOGY

- The predictive method was used to analyze the data.
- We divide the data into 80% for training and the remaining 20% to evaluate the predictive capacity of the model.
- Machine Learning Prediction with:
  - Plotly Dash.

RESULTS

Data Collection:

- We collect data on SpaceX launches, specifically focusing on Falcon 9 launches.

Import Libraries and Define Auxiliary Functions

We will import the following libraries into the lab

[1]:
```
# Requests allows us to make HTTP requests which we will use to get data from an API
import requests
# Pandas is a software library written for the Python programming language for data manipulation and analysis.
import pandas as pd
# NumPy is a library for the Python programming language, adding support for large, multi-dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays
import numpy as np
# Datetime is a library that allows us to represent dates
import datetime

# Setting this option will print all collumns of a dataframe
pd.set_option('display.max_columns', None)
# Setting this option will print all of the data in a feature
pd.set_option('display.max_colwidth', None)
```

[2]:
```
# Takes the dataset and uses the rocket column to call the API and append the data to the list
def getBoosterVersion(data):
    for x in data['rocket']:
       if x:
        response = requests.get("https://api.spacexdata.com/v4/rockets/"+str(x)).json()
        BoosterVersion.append(response['name'])
```

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

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

RESULTS

Web Scrapping:

- Web scraping was performed to collect historical records of Falcon 9 launches from a Wikipedia page titled List of Falcon 9 and Falcon Heavy launches.

[38]:
```
static_url = "https://en.wikipedia.org/w/index.php?title=List_of_Falcon_9_and_Falcon_Heavy_launches&oldid=1027686922"
```

[16]:
```
df.head()
```

[16]:

	Flight No.	Date	Time	Version Booster	Launch Site	Payload	Payload mass	Orbit	Customer	Launch outcome	Booster landing
0	1	4 June 2010	18:45	F9 v1.0B0003.1	CCAFS	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success\n	Failure
1	2	8 December 2010	15:43	F9 v1.0B0004.1	CCAFS	Dragon	0	LEO	NASA	Success	Failure
2	3	22 May 2012	07:44	F9 v1.0B0005.1	CCAFS	Dragon	525 kg	LEO	NASA	Success	No attempt\n
3	4	8 October 2012	00:35	F9 v1.0B0006.1	CCAFS	SpaceX CRS-1	4,700 kg	LEO	NASA	Success\n	No attempt
4	5	1 March 2013	15:10	F9 v1.0B0007.1	CCAFS	SpaceX CRS-2	4,877 kg	LEO	NASA	Success\n	No attempt\n

RESULTS

Data Wrangling

- The dataset is processed and cleaned to prepare it for analysis.
- We mainly convert the results into training labels, where 1 means that the booster landed successfully and 0 means that it did not succeed.

[97]:
```
landing_class= pd.DataFrame(landing_class)
landing_class.rename(columns= {0: 'Outcome'}, inplace= True)
df['Class'] = landing_class['Outcome']
```

[98]:
```
df.head(10)
```

[98]:

	FlightNumber	Date	BoosterVersion	PayloadMass	Orbit	LaunchSite	Outcome	Flights	GridFins	Reused	Legs	LandingPad	Block	ReusedCount	Serial	Longitude	Latitude	Class
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	1.0
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	1.0
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	1.0
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	1.0
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	1.0
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	1.0
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	1.0
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	1.0
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	1.0
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	1.0

RESULTS

EDA: Data Analysis SQL

- We explore the SpaceX dataset using SQL queries and visualization techniques.

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

```
[31]: %sql select distinct "Launch_Site" from SPACEXTABLE
 * sqlite:///my_data1.db
Done.
```

[31]:

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

- Total number of successful and failure mission outcomes

```
[62]: %sql select "Landing_Outcome", count (*) as Total from SPACEXTABLE where "Landing_Outcome" in ('Succes
 * sqlite:///my_data1.db
Done.
```

[62]:

Landing_Outcome	Total
Failure	3
Success	38

- Count of landing outcomes (such as Failure (drone ship) or Success (ground pad)) between the date 2010-06-04 and 2017-03-20

```
[73]: %sql select "Landing_Outcome", count (*) as Recuento fr
 * sqlite:///my_data1.db
Done.
```

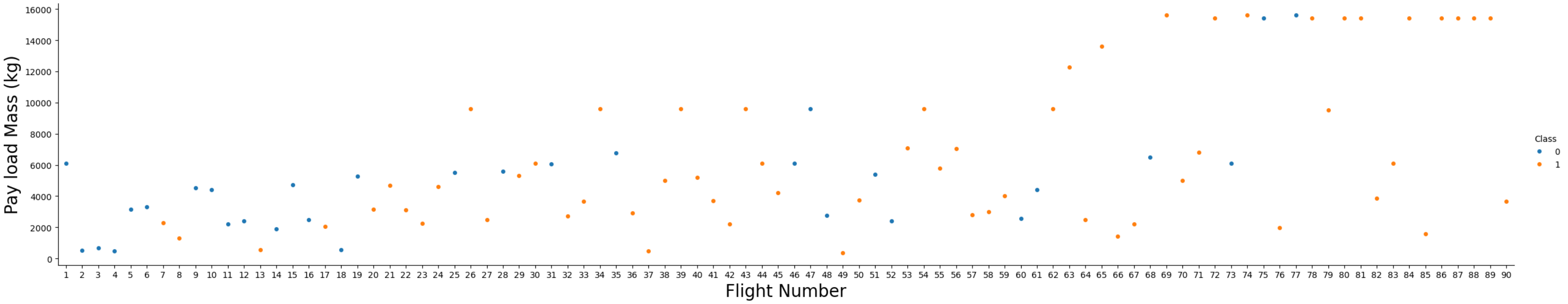
[73]:

Landing_Outcome	Recuento
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

RESULTS

EDA: Visualization

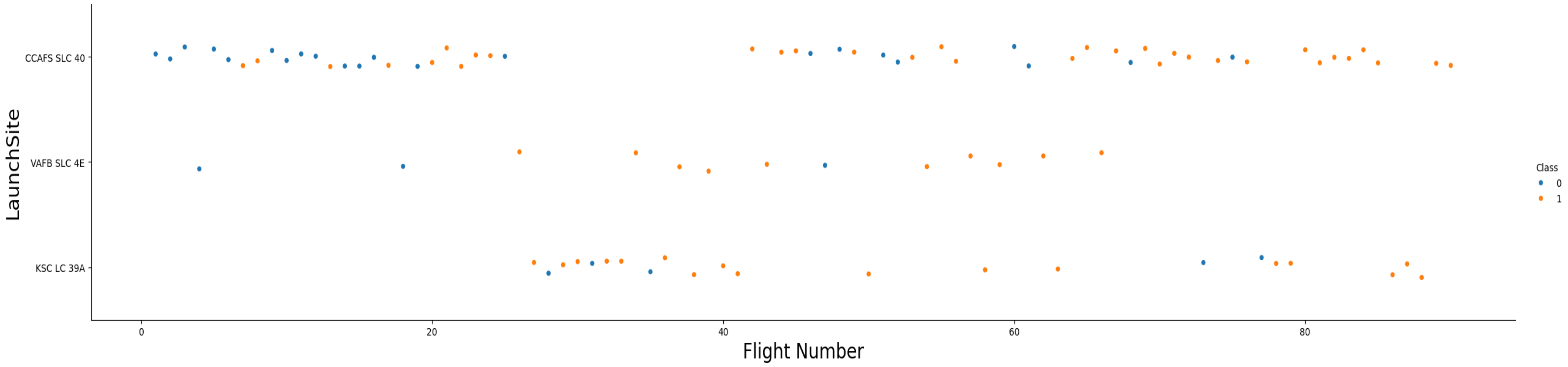
- Through this scatter plot we can see that the heavier the payload is, the less likely it is to return the first stage successfully.



RESULTS

EDA: Visualization

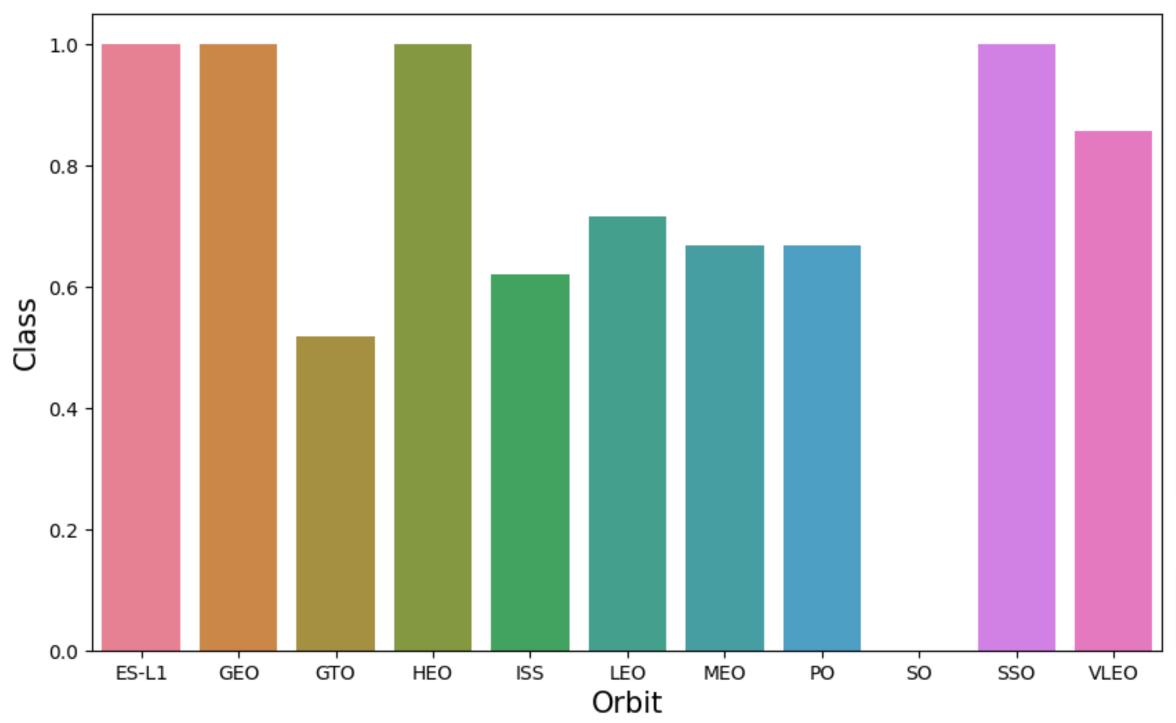
- Launch sites with the most successful flights



RESULTS

EDA: Visualization

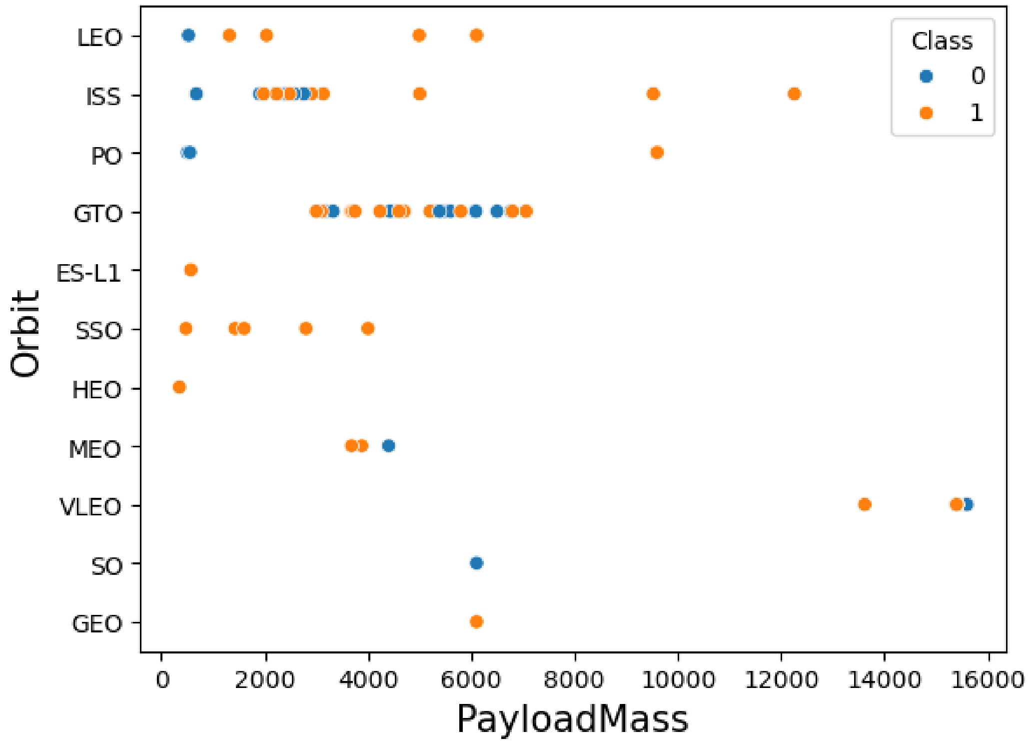
- Relationship between success rate and orbit type.



RESULTS

EDA: Visualization

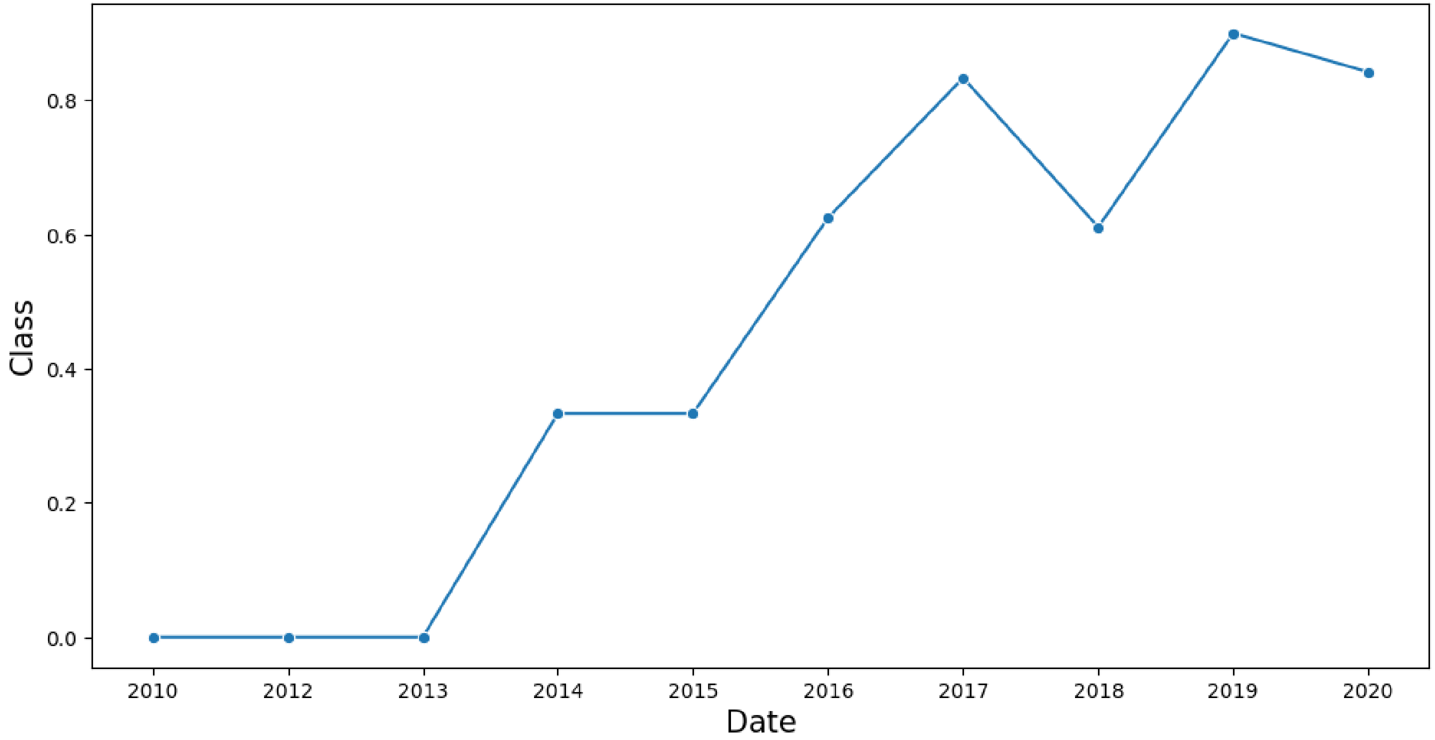
- Polar, LEO and ISIS orbits show a higher successful landing rate with heavy payloads.



RESULTS

EDA: Visualization

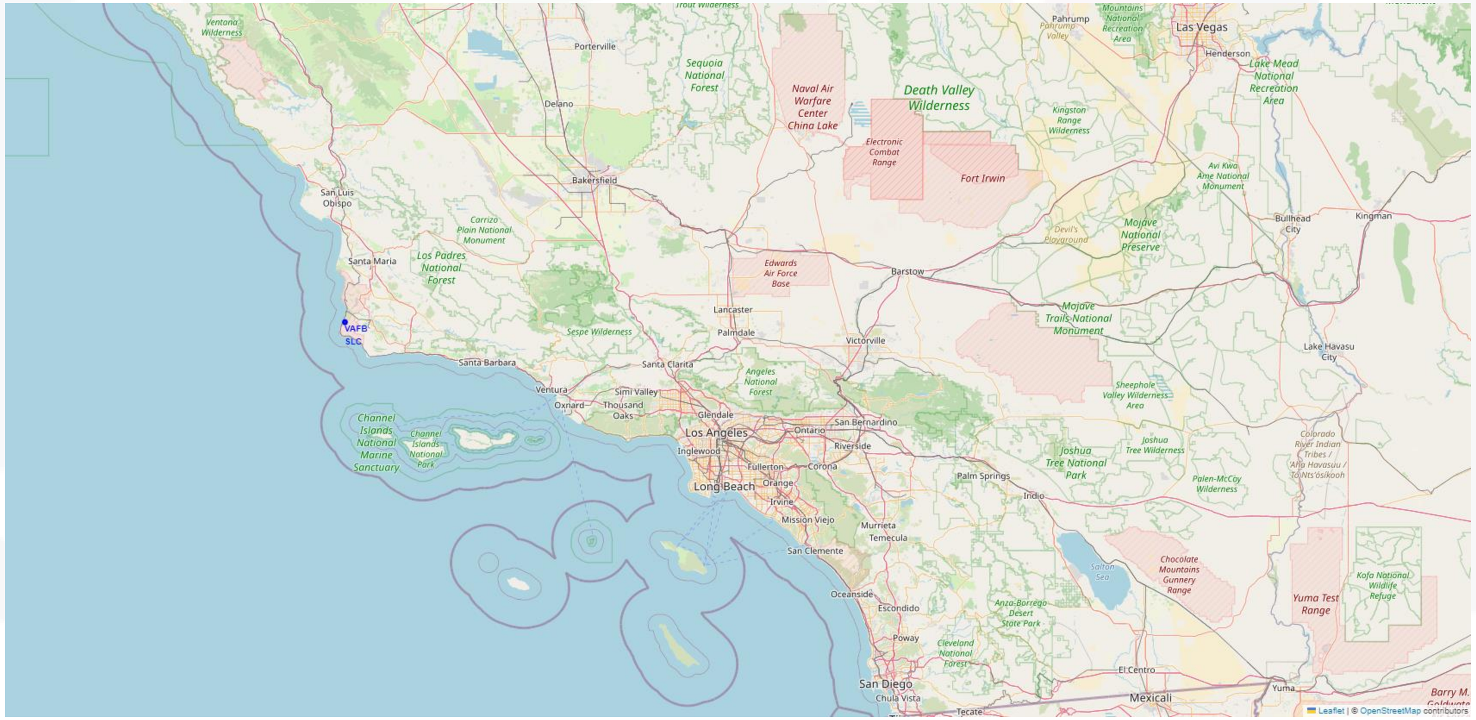
- Increase in successful flights from 2013 to 2020.



RESULTS

EDA: Visualization

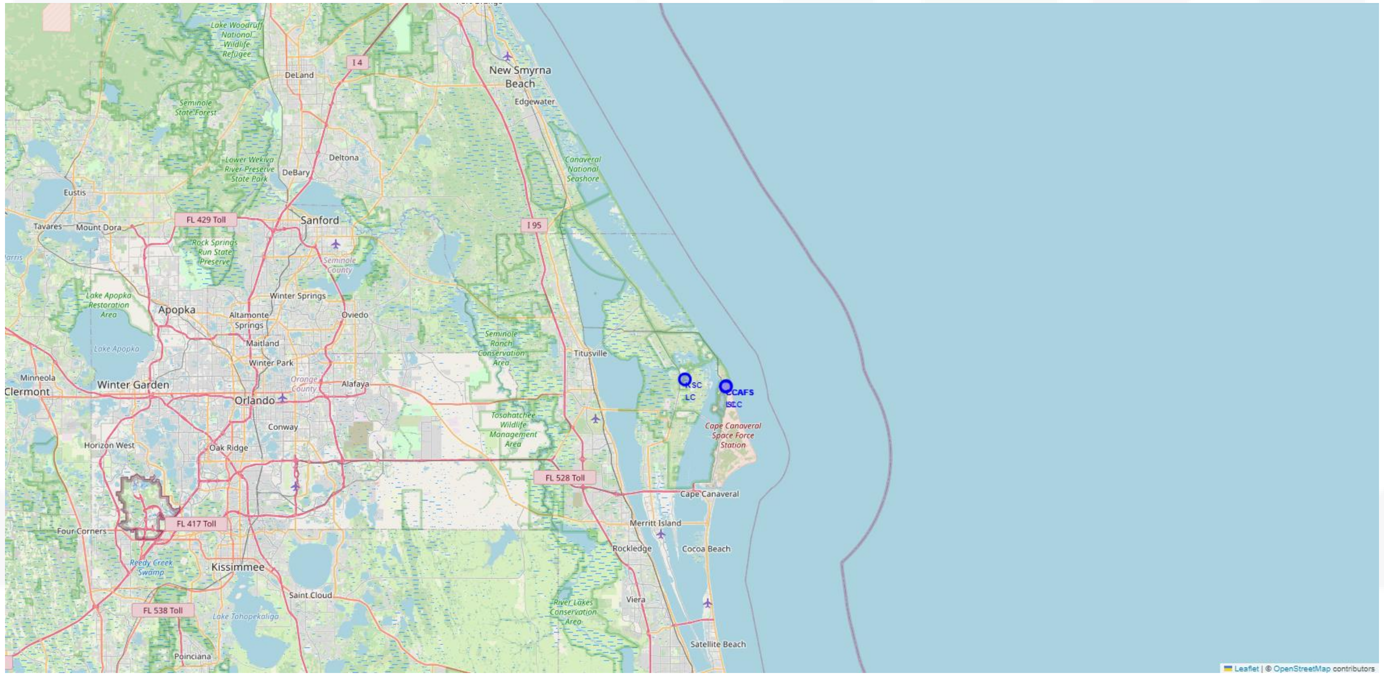
- Location of launch sites
  - VAFB SLC



RESULTS

EDA: Visualization

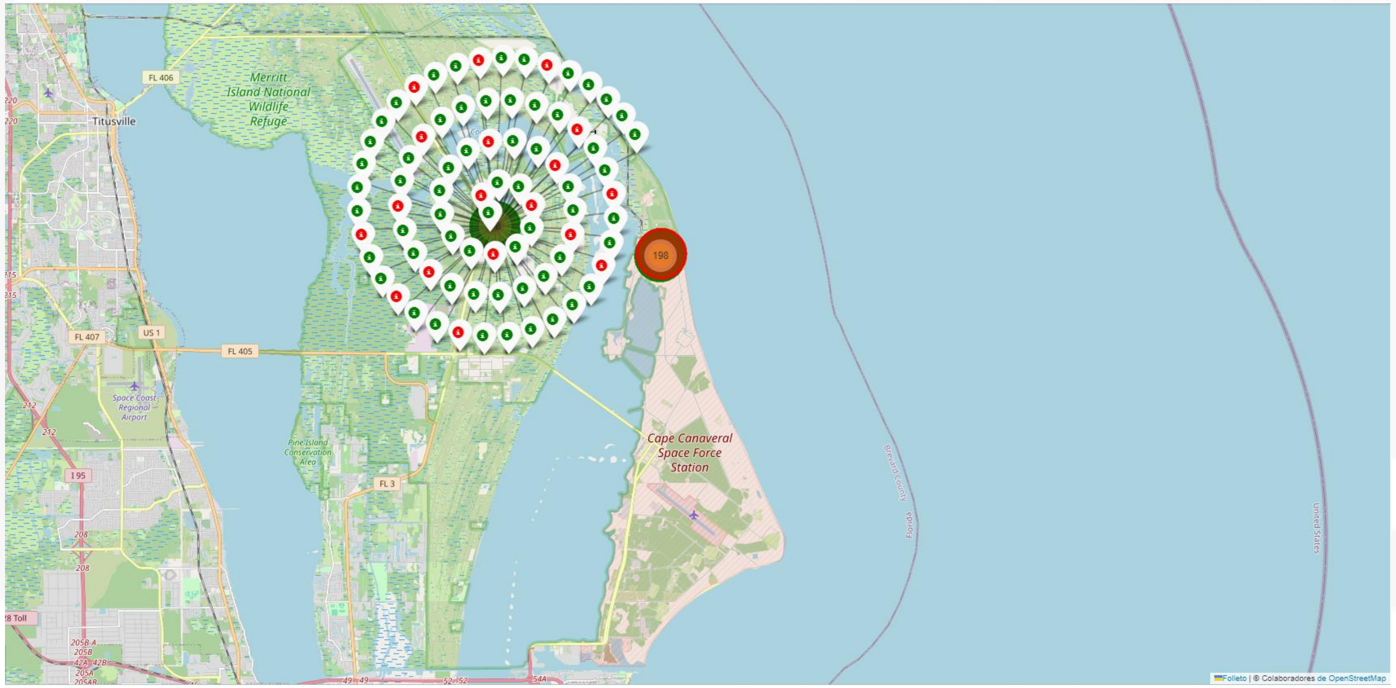
- Location of launch sites
  - CCAFS LC, CCAFS SLC and KSC LC



RESULTS

EDA: Visualization

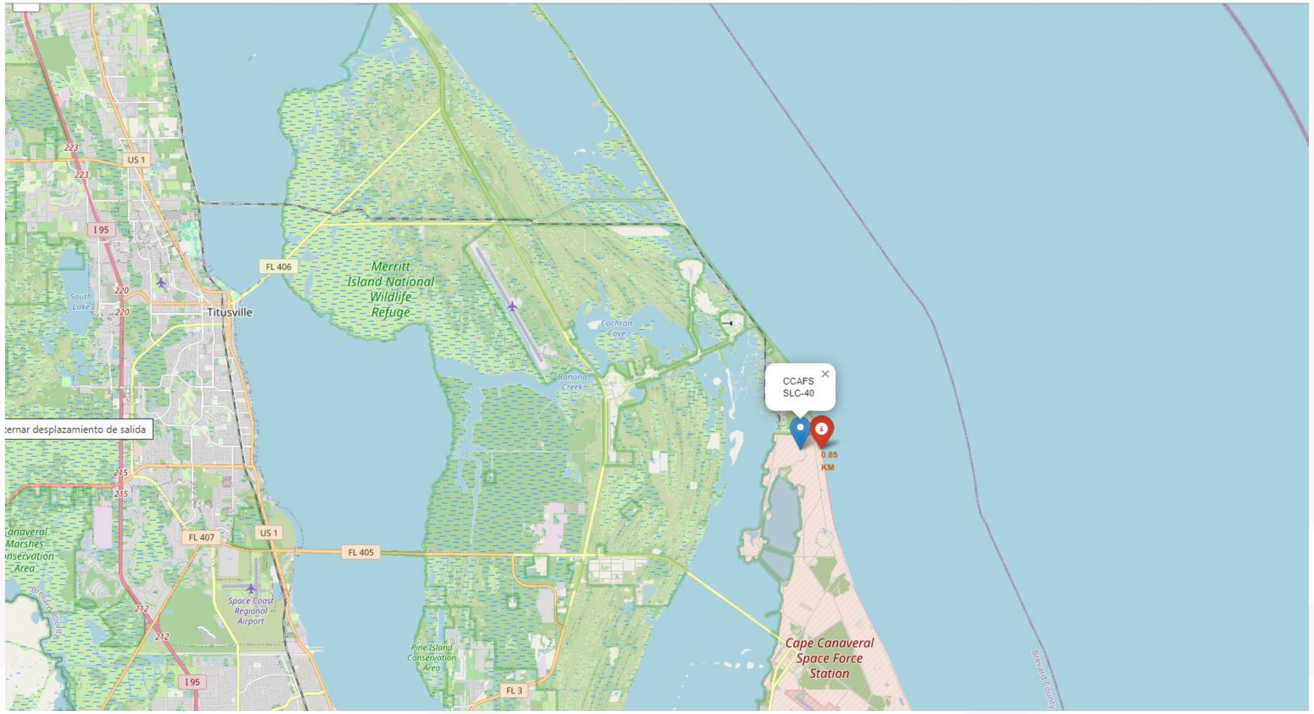
- Log of successful/failed launches by launch site
  - KSC LC-39A



RESULTS

EDA: Visualization

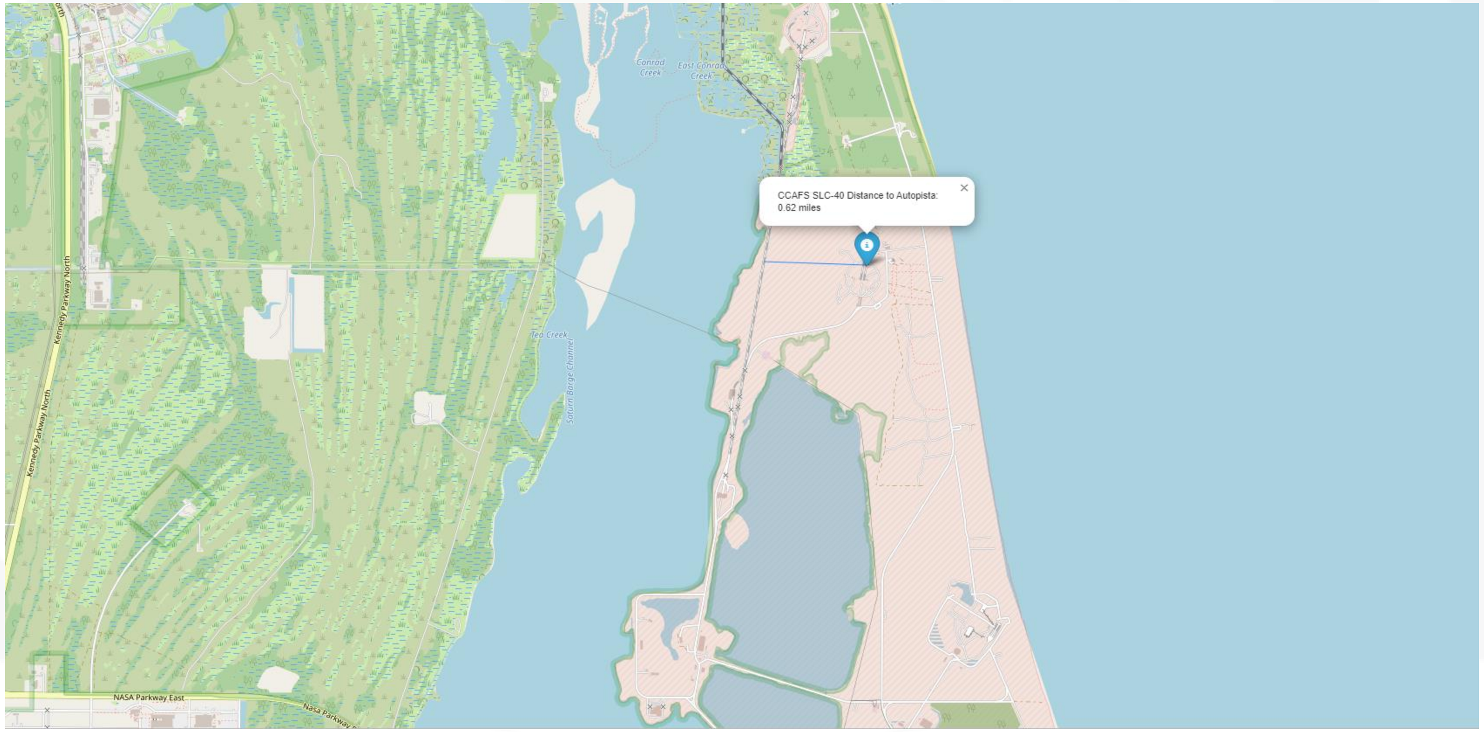
- Proximity between CCAFS SLC-40 (launch site) and the coastline



RESULTS

EDA: Visualization

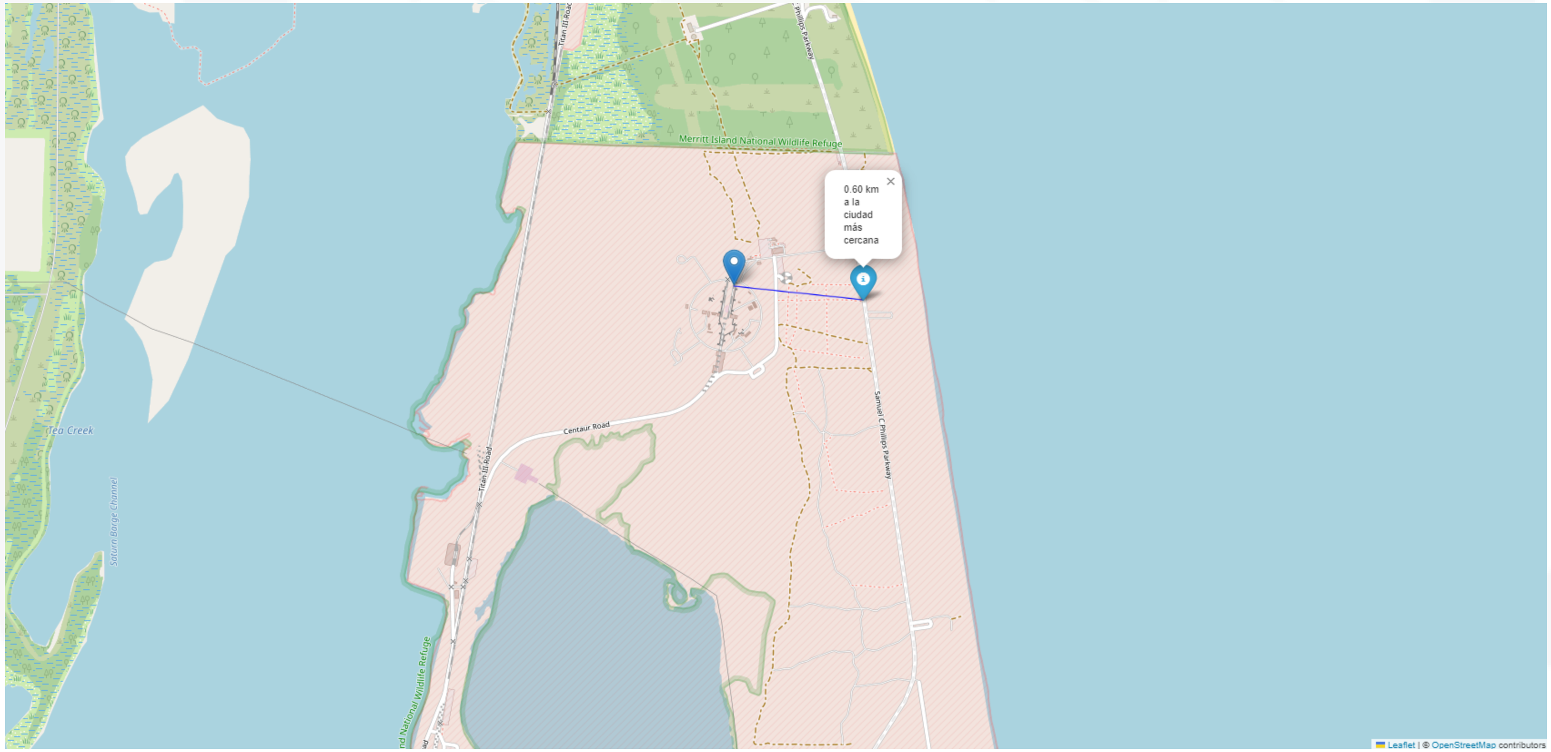
- proximity between CCAFS SLC-40 (launch site) and highway



RESULTS

EDA: Visualization

- Proximity between CCAFS SLC-40 (launch site) and nearest city

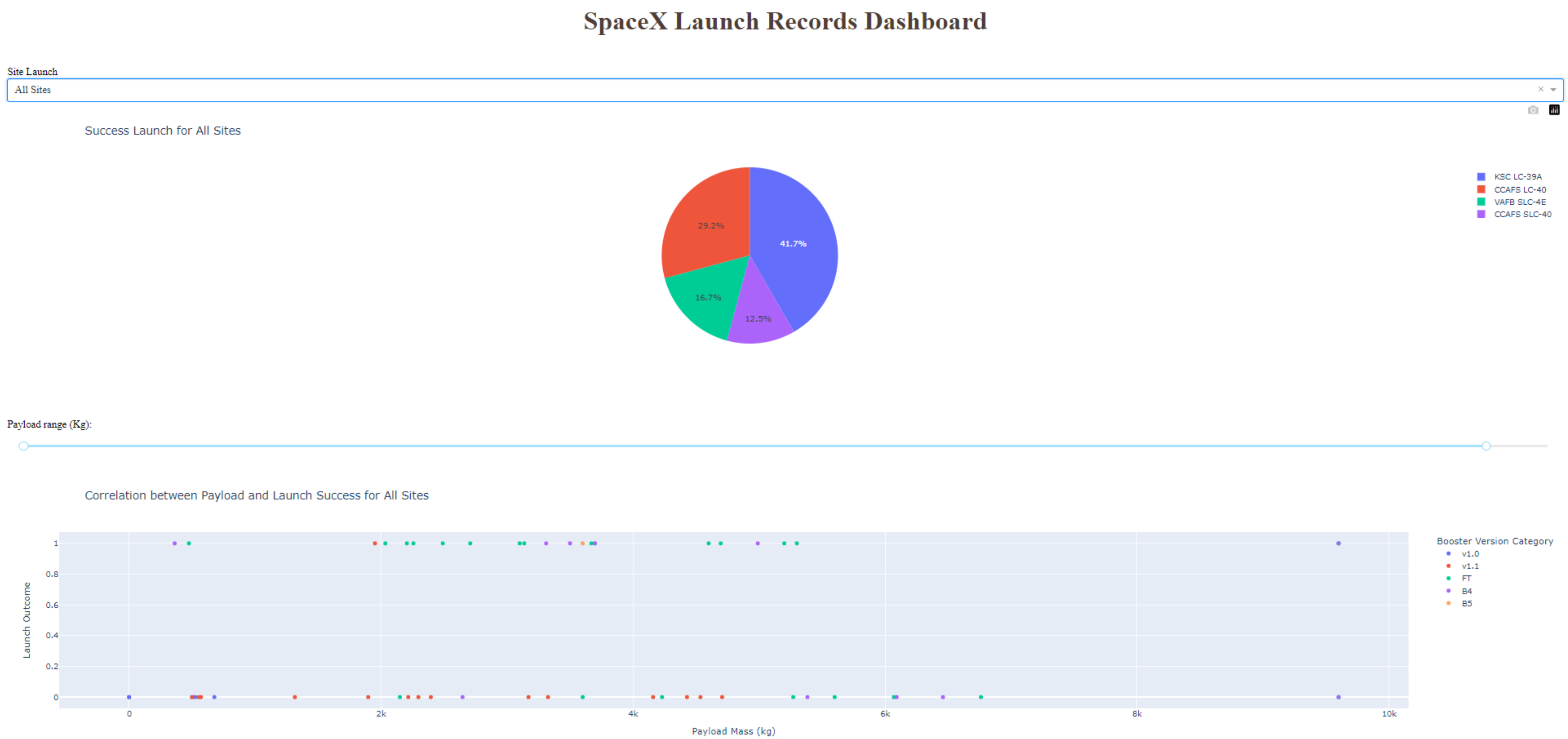


DASHBOARD

<The GitHub link of the Cognos dashboard goes here.>

DASHBOARD TAB 1

SpaceX Launch Records Dashboard 1/3

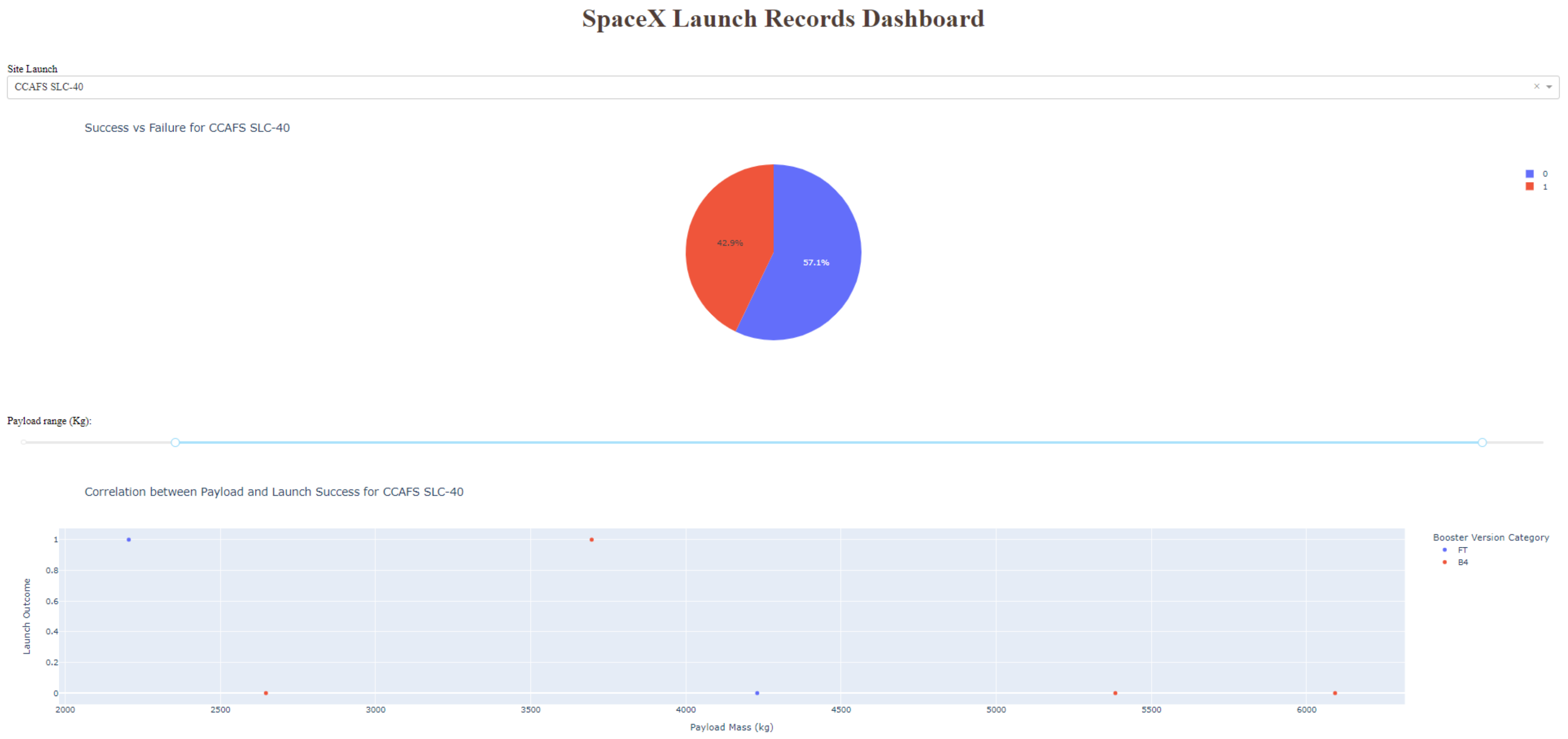


IBM Developer

SKILLS NETWORK

DASHBOARD TAB 2

SpaceX Launch Records Dashboard 2/3

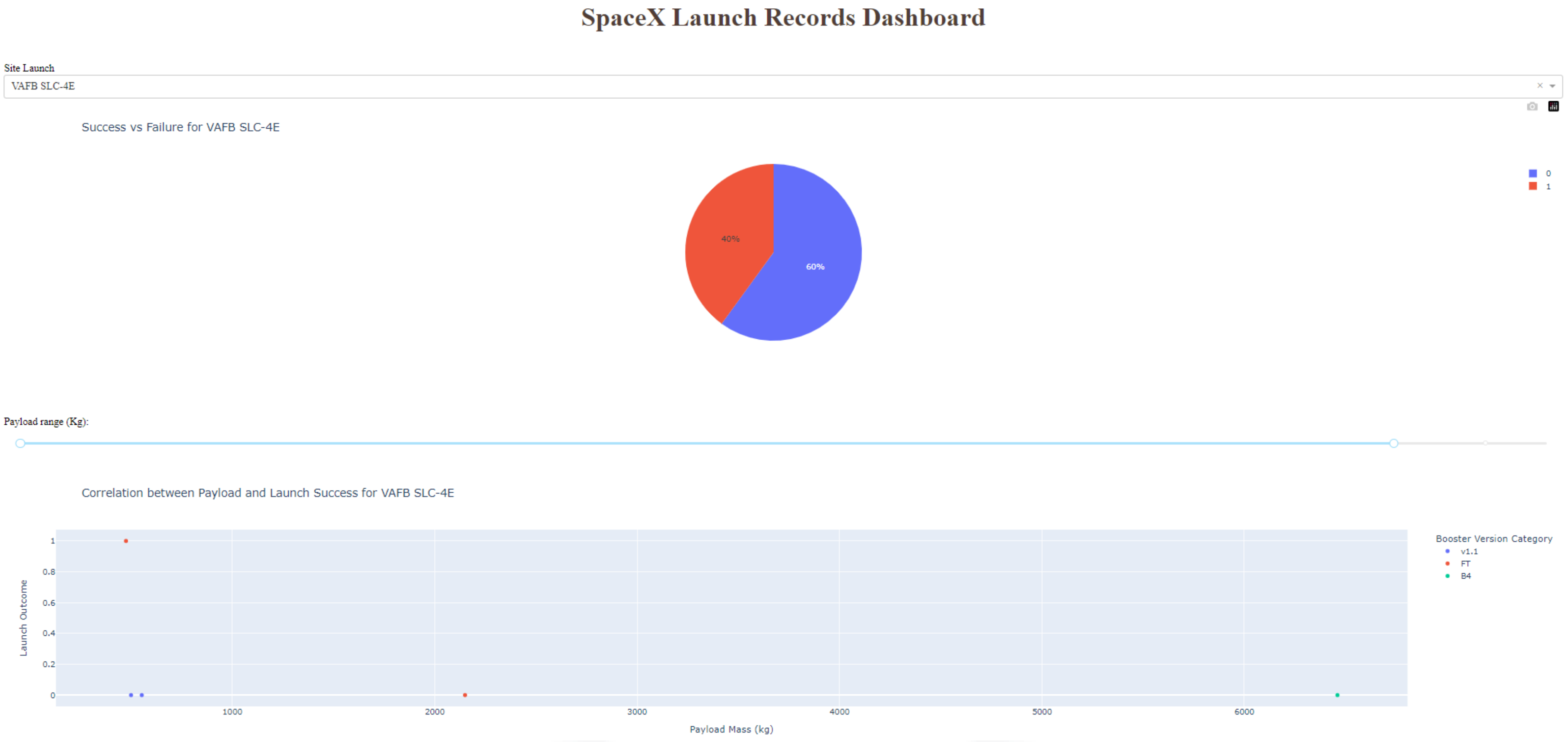


IBM Developer

SKILLS NETWORK

DASHBOARD TAB 3

SpaceX Launch Records Dashboard 3/3



RESULTS:

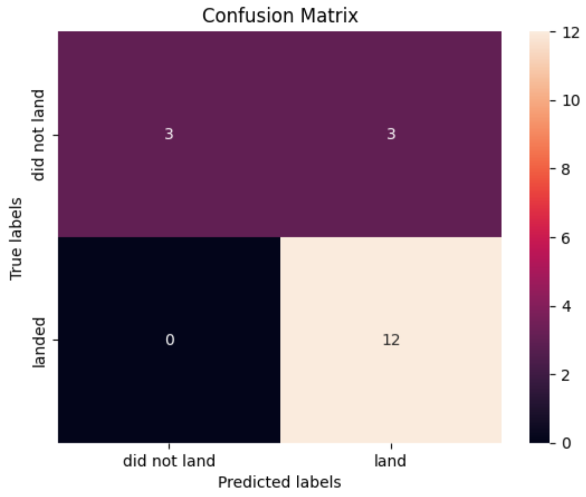
Predictive Analysis Classification:

- Logistic Regression: The precision of the test data was calculated using the score method

```
[16]: accuracy=logreg_cv.score(X_test, Y_test)
      accuracy
[16]: 0.8333333333333334
```

Veamos la matriz de confusión:

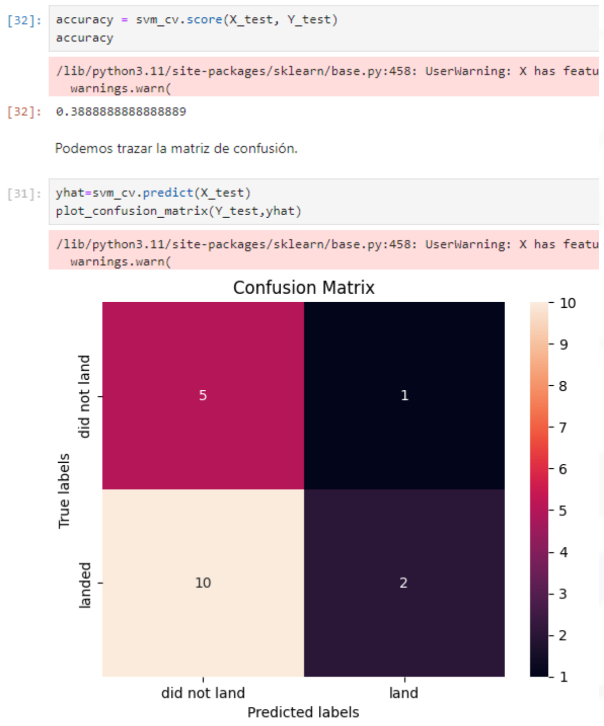
```
[17]: yhat=logreg_cv.predict(X_test)
      plot_confusion_matrix(Y_test,yhat)
```



RESULTS:

Predictive Analysis Classification:

- Support Vector Machine: The precision of the test data was calculated using the score method



RESULTS:

Predictive Analysis Classification:

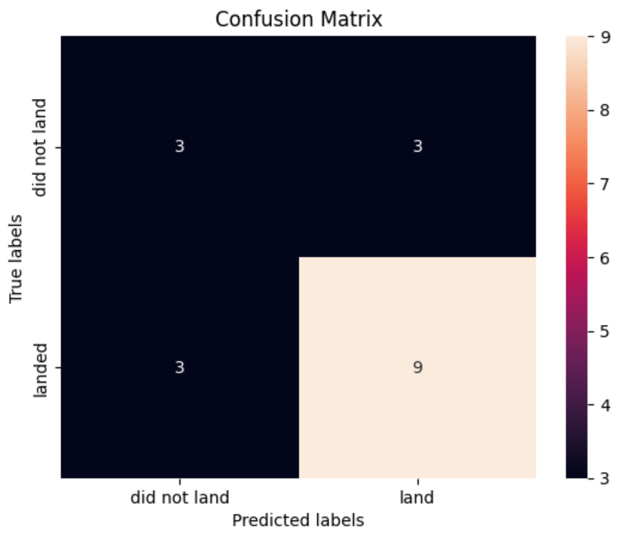
- Decision Tree: The precision of the test data was calculated using the score method

```
[38]: accuracy_tree=tree_cv.score(X_test, Y_test)
      accuracy_tree
```

```
[38]: 0.6666666666666666
```

Podemos trazar la matriz de confusión.

```
[37]: yhat = tree_cv.predict(X_test)
      plot_confusion_matrix(Y_test,yhat)
```



RESULTS:

Predictive Analysis Classification:

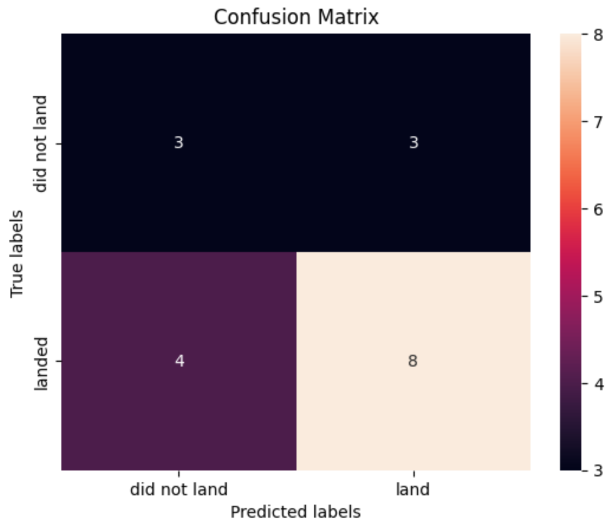
- K-Nearest Neighbors: The precision of the test data was calculated using the score method

```
[42]: accuracy_knn = knn_cv.score(X_test, Y_test)
      accuracy_knn
```

```
[42]: 0.6111111111111112
```

Podemos trazar la matriz de confusión.

```
[43]: yhat = knn_cv.predict(X_test)
      plot_confusion_matrix(Y_test,yhat)
```



RESULTS :

Predictive Analysis Classification:

- As we can see in the previous confusion matrices, the model that yields a precision of 83.33% is the Logistic Regression model.
- It should be noted that the confusion matrix is a very useful tool to assess the performance of a classification model based on machine learning.
- In this case, the confusion matrix is used to evaluate the performance of the model and analyze how it performs in classification.

CONCLUSION

- Exploratory data analyzes performed on the SpaceX launch data set have provided valuable insights into correlations between launch site and flight success rates. The results reveal that as the number of flights increases, the probability of a successful recovery of the first stage also increases, highlighting the importance of accumulated experience.
- A significant finding is the crucial influence of payload mass on first-stage recovery. The graphs clearly indicate that more massive loads are associated with a lower probability of successful recovery. This finding could be vital for planning and decision-making in future missions.

CONCLUSION

- Disparities in success rates were identified between different launch sites, with CCAFS LC-40 and KSC LC-39A leading with rates over 80%, while VAFB SLC 4E clocks in at around 67%. Visualizing this data through scatter plots highlights specific patterns, such as the relationship between VAFB-SLC launch site and launch success as a function of payload mass.
- It is relevant to note that although the launch site can influence the success rate, other factors, such as the number of flights and the payload of the rocket, also play a crucial role.
- The positive trend in success rate from 2013 to 2020 reflects the continued refinement of SpaceX's operations, supporting the idea of constant progress in the efficiency and reliability of its launches.

APPENDIX

- Sample of total payload mass carried by NASA-launched boosters (CRS)

```
[39]: %sql select sum("PAYLOAD_MASS__KG_") as Total_Mass from SPACEXTABLE where "Customer" = 'NASA (CRS)'
 * sqlite:///my_data1.db
Done.
```

[39]:

Total_Mass
45596

- Sample of the average payload mass carried by the F9 v1.1 booster version

```
[47]: %sql select avg("PAYLOAD_MASS__KG_") as Average_Payload_Mass from SPACEXTABLE where "Booster_Version" = 'F9 v1.1'
 * sqlite:///my_data1.db
Done.
```

[47]:

Average_Payload_Mass
2928.4

APPENDIX

- List of thrusters that are successful in unmanned spacecraft and have a charge mass greater than 4000 but less than 6000.

```
[53]: %sql select Booster_Version from SPACEXTABLE where "PAYLOAD_MASS__KG_" > 4000 and "PAYLOAD_MASS__KG_" < 6000 and "Landing_Outcome" = 'Success (drone ship)';
 * sqlite:///my_data1.db
Done.
```

[53]:

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

APPENDIX

- List of the booster_versions which have carried the maximum payload mass

```
[68]: %sql select distinct "Booster_Version" from SPACEXTABLE where "PAYLOAD_MASS__KG_" = (select max("PAYLOAD_MASS__KG_") from SPACEXTABLE);
 * sqlite:///my_data1.db
Done.
```

[68]:

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

APPENDIX

- List of the booster_versions which have carried the maximum payload mass

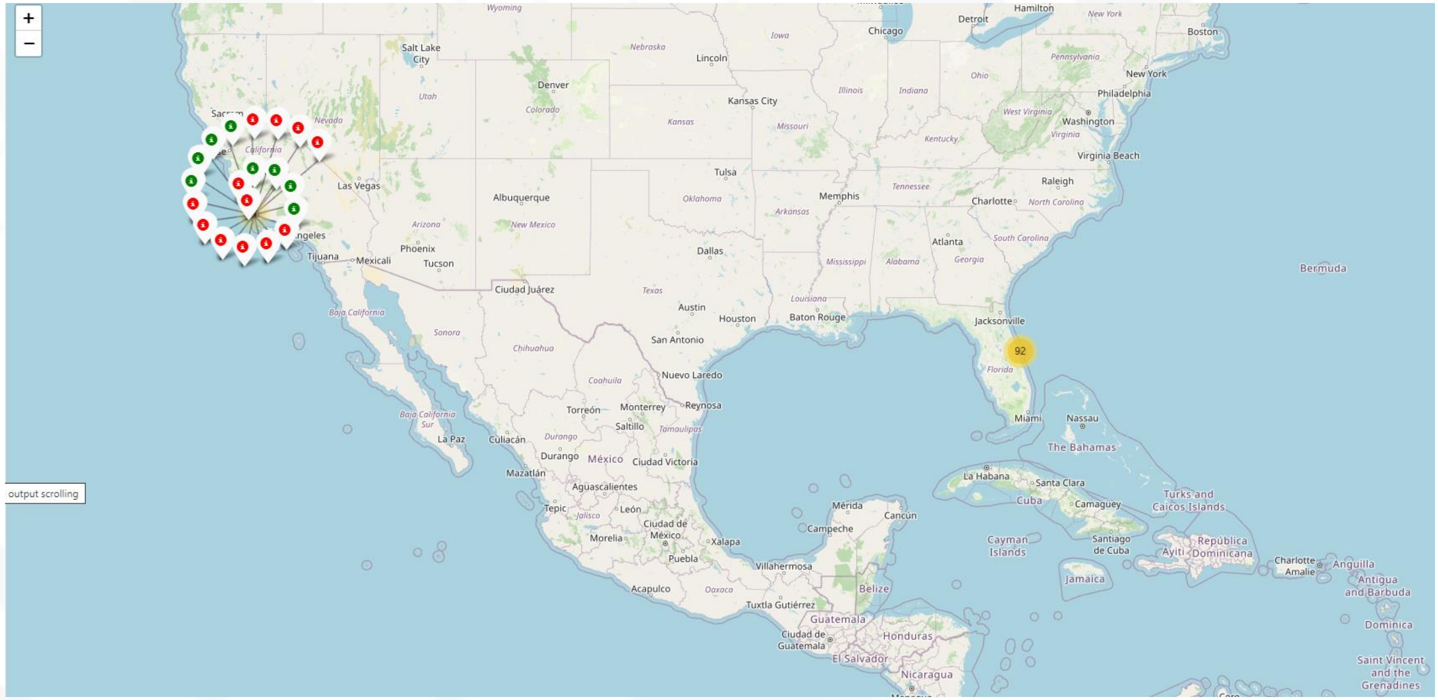
```
[68]: %sql select distinct "Booster_Version" from SPACEXTABLE where "PAYLOAD_MASS__KG_" = (select max("PAYLOAD_MASS__KG_") from SPACEXTABLE);
 * sqlite:///my_data1.db
Done.
```

[68]:

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

APPENDIX

- Launch result in spacex_df data frame (VAFB SLC-4E)



APPENDIX

- Launch result in spacex_df data frame (KSC LC-39A)