Winning Space Race with Data Science

Juan Carlos Galeano
Agosto 31/2024

Outline

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

Executive Summary

- Summary of methodologies
  - Data Collection using SpaceX REST API
  - Data Collection with Web Scraping
  - Data Wrangling to create success/fail outcome variable
  - Exploratory Data Analysis using SQL
  - Exploratory Data Analysis using Python Pandas and Matplotlib
  - Launch Sites Analysis with Folium-Interactive Visual Analytics and PlotyDash
  - Machine Learning Landing Prediction to predict landing outcomes using logistic regression, (SVM), decision tree and KNN

- Summary of all results
  - EDA results: Launch success has improved over time, KSC LC-39A has the highest success rate among landing sites, Orbits ES-L1, GEO, HEO, and SSO have a 100% success rate
  - Interactive Visual Analytics and Dashboards: Most launch sites are near the equator and all are close to the coast
  - Predictive Analysis(Classification): All models performed similarly on the test set.

Introduction

- ## Project background and context

Today we are living in a commercial space age in which companies are making space travel a reality for everyone, with SpaceX being the most successful with achievements such as sending spacecraft to the International Space Station, satellite Internet and sending missions manned into space. Rocket launches are relatively inexpensive like the Falcon 9 with a cost of $62 million, a value much lower than its competition because it can reuse the first stage. while other providers cost more than $165 million each.

Therefore, the work aims to determine if the first stage will land, and determine the launch price. To do this, we can use public data and machine learning models to predict whether SpaceX – or a competing company – can reuse the first stage.

- ## Problems you want to find answers

Predict if the Falcon 9 first stage will land successfully using data from Falcon 9 rocket launches

Rate of successful landings over time

Best predictive model for successful landing (classification)

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Using REST API and web scraping techniques
- Perform data wrangling
  - Filtering the data, handling missing values and preparing the data for analysis and modeling
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Predicting landing outcomes using classification models, tuning and evaluating models to find best model.

Data Collection

STAGES

- **Request data** (rocket launch data) from SpaceX API
- **Response** by using .json() and convert to a dataframe using .json_normalize()
- **Create dictionary** from the data
- **Create dataframe** from the dictionary
- **Filter dataframe by** containg only Falcon 9 launches
- **Wrangle data** by replacing missing values with calculated mean() of Payload Mass
- **Export data** to csv file

Data Collection – SpaceX API

- We collect data using the SpaceX API with Get Requests, then we decode the content of the response as a Json result and use the json_normalize method to convert the json result to a data frame. Adicionally, we filter the dataframe to only include Falcon 9 launches
- The GitHub URL of the completed SpaceX API calls notebook https://github.com/Juanchogaleano/SpaceX/blob/main/01-SpaceX-data-collection-api.ipynb

28/8/24, 9:06 p.m. jupyter-labs-spacex-data-collection-api

e":true,"tbd":false,"launch_library_id":"f33d5ece-e825-4cd8-809f-1d4c72a2e0d3","id":"62dd70d5202306255024d139"}]'

You should see the response contains massive information about SpaceX launches. Next, let's try to discover some more relevant information for this project.

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:

In [9]:
```
static_json_url='https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud
```

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

In [10]:
```
response.status_code
```

Out[10]: 200

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

In [11]:
```
# Use json_normalize meethod to convert the json result into a dataframe
data=pd.json_normalize(response.json())
```

Using the dataframe `data` print the first 5 rows

In [12]:
```
# Get the head of the dataframe
data.head()
```

Data Collection - Scraping

- We performed web scraping to collect historical Falcon 9 launch logs from Wikipedia using BeautifulSoupand request, then extracted all column/variable names from the HTML table header, create a data frame by parsing the release HTML tables
- The GitHub URL of the completed web scraping notebook https://github.com/Juanchogaleano/SpaceX/blob/main/02-SpaceX-webscraping.ipynb

29/8/24, 3:23 p.m. jupyter-labs-webscraping

```
# assign the response to a object
```

Create a `BeautifulSoup` object from the HTML `response`

```
In [7]: # Use BeautifulSoup() to create a BeautifulSoup object from a response text content

soup = BeautifulSoup(response.text, 'lxml')
```

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

```
In [9]: # Use soup.title attribute
print (soup.title.string)
```

```
List of Falcon 9 and Falcon Heavy launches - Wikipedia
```

TASK 2: Extract all column/variable names from the HTML table header

Next, we want to collect all relevant column names from the HTML table header

Let's try to find all tables on the wiki page first. If you need to refresh your memory about `BeautifulSoup`, please check the external reference link towards the end of this lab

```
In [13]: # Use the find_all function in the BeautifulSoup object, with element type `table`
# Assign the result to a list called `html_tables`

html_tables = soup.find_all('table')
```

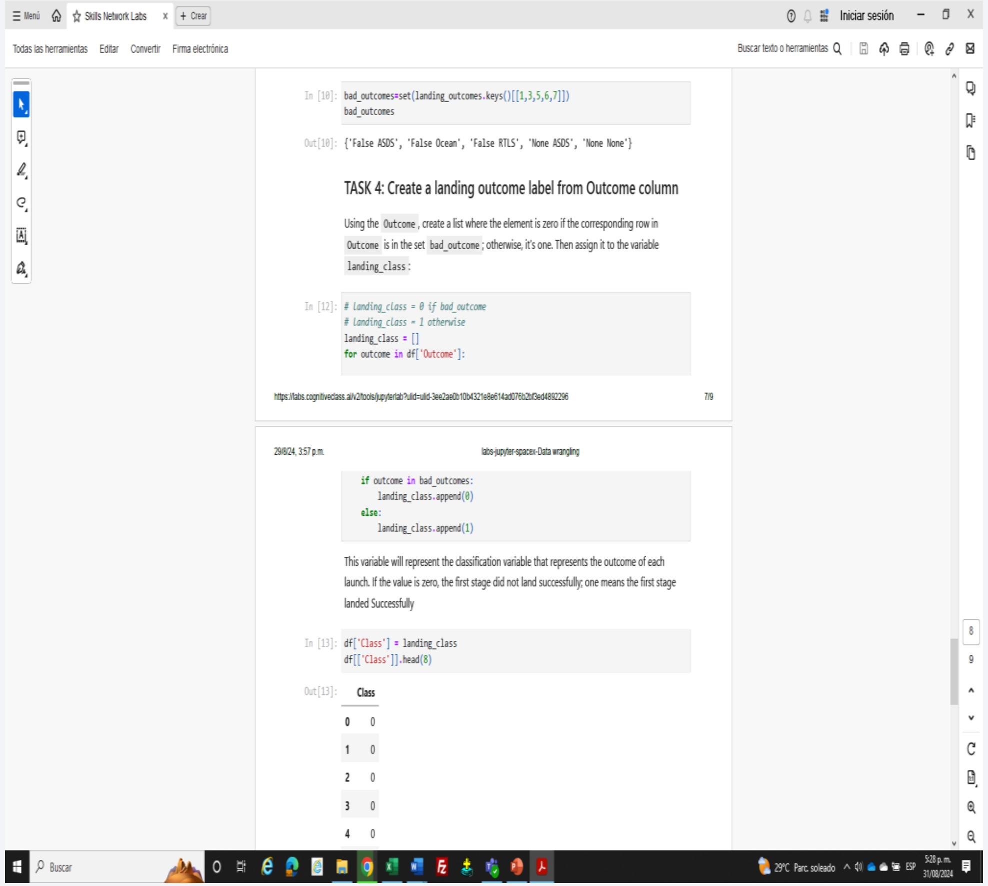
Starting from the third table is our target table contains the actual launch records.

```
In [14]: # Let's print the third table and check its content
first_launch_table = html_tables[2]
print(first_launch_table)
```

Data Wrangling

- We first create a Pandas DF with the collected data, then calculate the number of launches on each site, the number and occurrence of each orbit and the number and occurrence of mission outcome of the orbits. Finally, we create a landing outcome label from Outcome column.

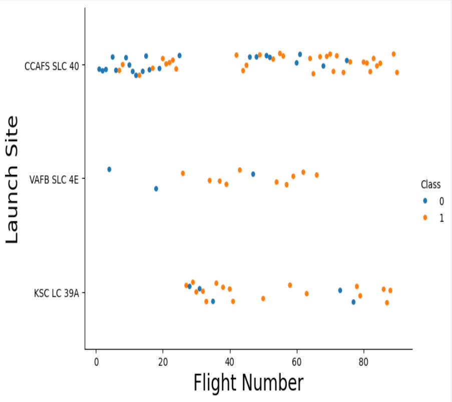
- The GitHub URL of your completed data wrangling related notebooks https://github.com/Juanchogaleano/SpaceX/blob/main/03-SpaceX-Data%20wrangling.ipynb

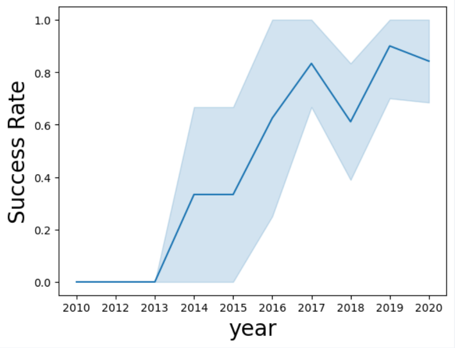


EDA with Data Visualization

Charts plotted

- **Scatter plot**: to visualize the relationship between Flight Number and Launch Site, Flight Number and. Payload Mass(Kg), Payload Mass(Kg) and Launch Site, Flight Number and Orbit type, Payload Mass(Kg) and Orbit type.
- **Bar chart:** to visualize the launch success rate(%) of each orbit type
- **Line plot**: to visualize the launch success rate of each year from 2013





- The GitHub URL of your completed EDA with data visualization notebook https://github.com/Juanchogaleano/SpaceX/blob/main/05-SpaceX-Eda-dataviz.ipynb

EDA with SQL

Querys

Display

- **The names of the unique launch sites in the space mission**
- **5 records where launch sites begin with the string 'CCA'**
- **The total payload mass carried by boosters launched by NASA (CRS)**
- **Average payload mass carried by booster version F9 v1.1**

- The GitHub URL of your completed EDA with SQL notebook https://github.com/Juanchogaleano/SpaceX/blob/main/04-SpaceX-Eda-sql.ipynb

List

- Date when the first succesful landing outcome in ground pad was acheived.
- 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
- Names of the booster_versions which have carried the maximum payload mass. Use a subquery
- Records which will display the month names, failure landing_outcomes in drone ship ,booster versions, launch_site for the months in year 2015.
- 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.

Interactive Map with Folium

Map objects such as markers, circles, lines, etc. you created and added to a folium map:

- Markers indicating Launch Sites
  - Color blue circle at NASA Johnson Space Center coordinate with a popup label showing its name using its lat and long coordinates
  - Color red circles at all launch sites coordinates with a popup label showing its name using its name using its lat and long coordinates
- Map with Folium
  - Color Markers of Launch Outcomes: successful(green) and unsuccessful(red) launch show which launch sites have high success rates
- Distances Between a Launch Site to Proximities
  - Color lines show distance from launch site CCAFS SLC-40 and its proximity to the nearest coastline, railway, highway, and city

- The GitHub URL of interactive map with Folium map https://github.com/Juanchogaleano/SpaceX/blob/main/06-SpaceX-Launch_site_location.ipynb

Build a Dashboard with Plotly Dash

Plots/graphs and interactions added to dashboard app

- **Dropdown list with Launch Sites:** Allow to user select all or someone launch sites
- **Pie-chart:** Show to user total successful launches by site as a percent of the total or by each site and class(0,1)
- **Slider:** Allow to user select payload mass range
- **Scatter Chart:** Show to user correlation between payload mass and success rate by booster version.
- The GitHub URL of Plotly Dash lab https://github.com/Juanchogaleano/SpaceX/blob/main/07-SpaceX-dash_app.py

Predictive Analysis (Classification)

Built, evaluate, improve and identify the best performing classification model with the next steps:

- Create a NumPy array from the column Class in data, by applying the method to_numpy()
- Standardize the data using the transform
- Use the function train_test_split to split the data X and Y into training and test data with the parameter test_size to 0.2
- Create object for logistic regression, support vector machine(SVC), decision tree, k nearest neighbors(KNN) with GridSearchCV and then Fit the object to find the best parameters from the dictionary parameters.
- Calculate the accuracy on the test data using the method score for all models
- Plot the confusion matrix for all models
- Find the best model using Jaccard_Score, F1_Score and Accuracy

Add the GitHub URL of your completed predictive analysis lab
https://github.com/Juanchogaleano/SpaceX/blob/main/08-SpaceX-Machine%20Learning%20Prediction.ipynb

Results

- Exploratory data analysis results
  - Launch success has improved over time
  - Orbits ES-L1, GEO, HEO and SSO have a 100% success rate
- Interactive analytics demo in screenshots
- Predictive analysis results
  - Decision Tree model is the best predictive model for the dataset

Section 2

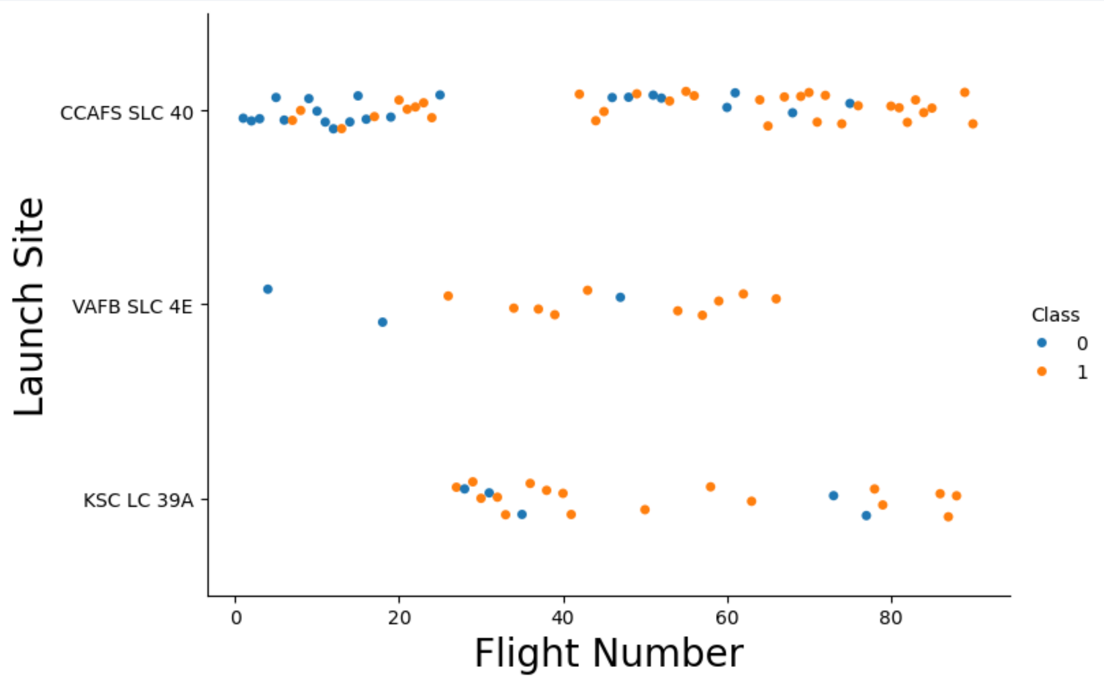
Insights drawn from EDA

Flight Number vs. Launch Site

- ➢ The new launches have a higher success rate
- ➢ Launch site from VAFB SLC 4E and KSC LC 39A have higher success rates
- ➢ The launches for the VAFB SLC 4E site above 50 number of flights have been 100% successful, for the KSC LC 39A site and for the CCAFS SLC site above 40 number of flights

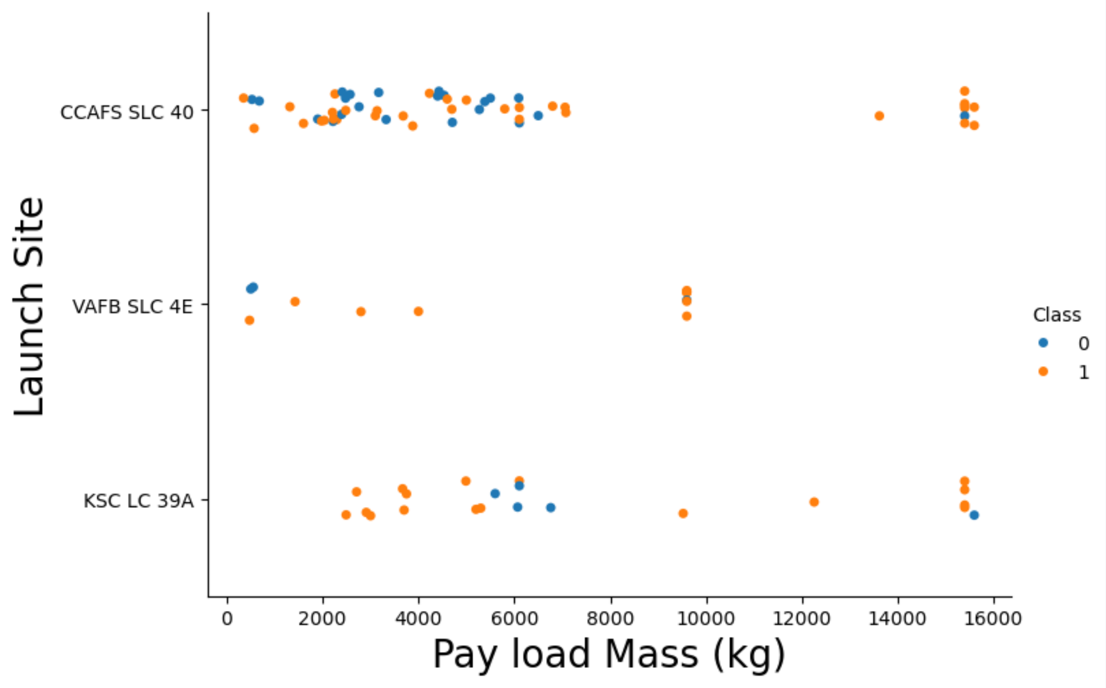
Success rate (**blue = fail**)

Success rate (**orange = success**)



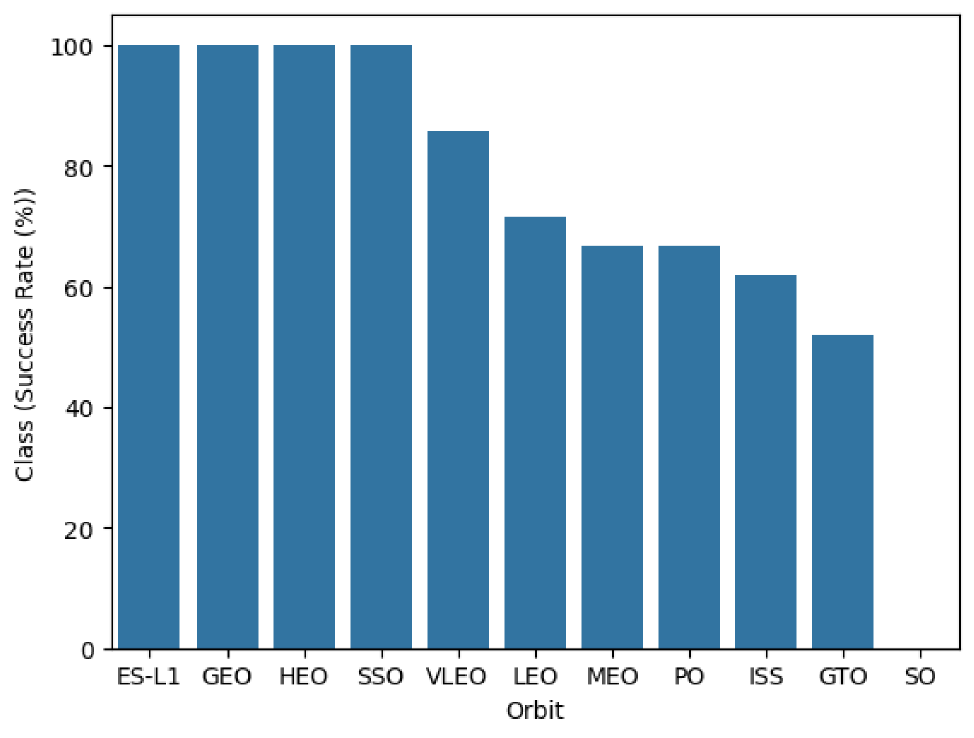
Payload vs. Launch Site

- ➢ Most lauches with payload greater than 8000 kg were successful
- ➢ Normally, higher success rate is observed at higher payload mass (kg), around 16000 kg
- ➢ Most launches from VAFB SKC 4E are successful, but none have been launched above 10,000 kg.
- ➢ There are not many launches with a payload greater than 10,000kg



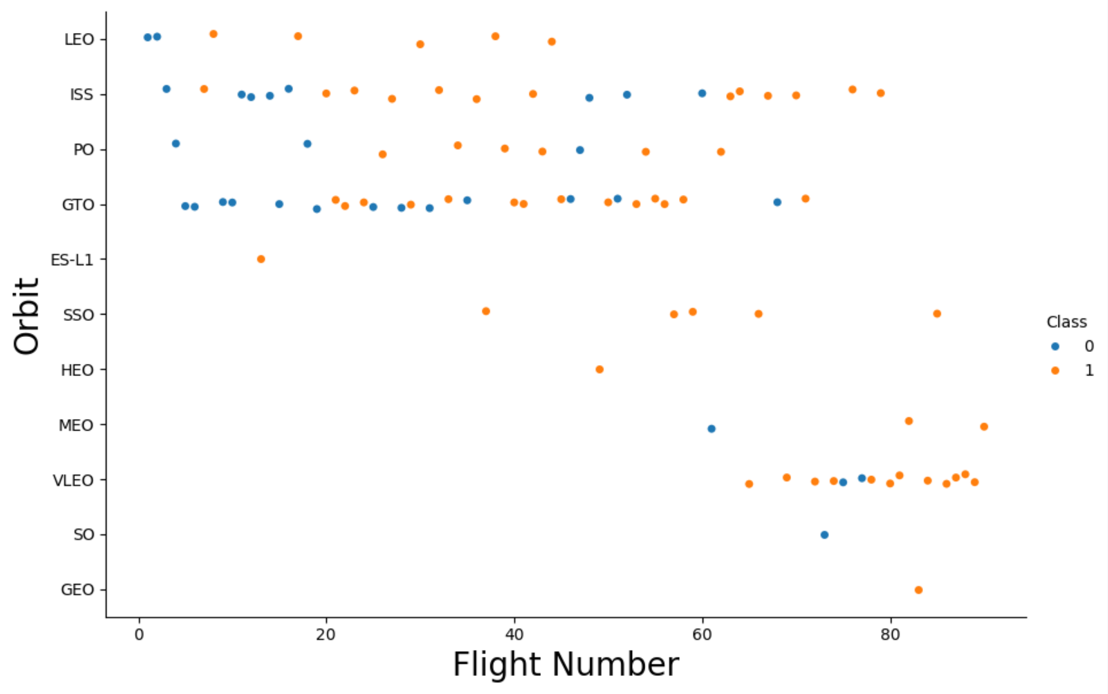
Success Rate vs. Orbit Type

- The orbit type ES-L1, GEO, HEO and SSO was 100% Success Rate
- The orbit type SO was 0% Success Rate
- The other types of orbits (GTO, ISS, LEO, MEO, PO) was between 50%-80% success rate



Flight Number vs. Orbit Type

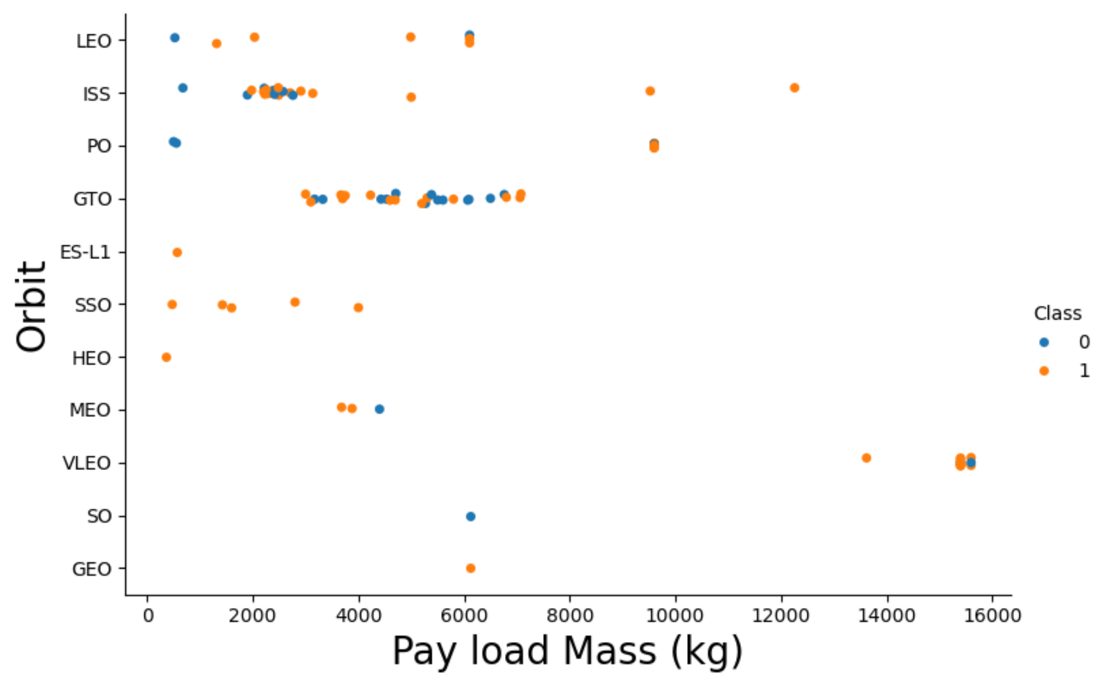
- ➢ Generally, the greater the number of launches, the greater the success rate of the launches in any of the types of orbit, except GTO orbit.
- ➢ Some orbits have had 100% success rates for any number of flights such as SSO, HEO y GEO



Payload vs. Orbit Type

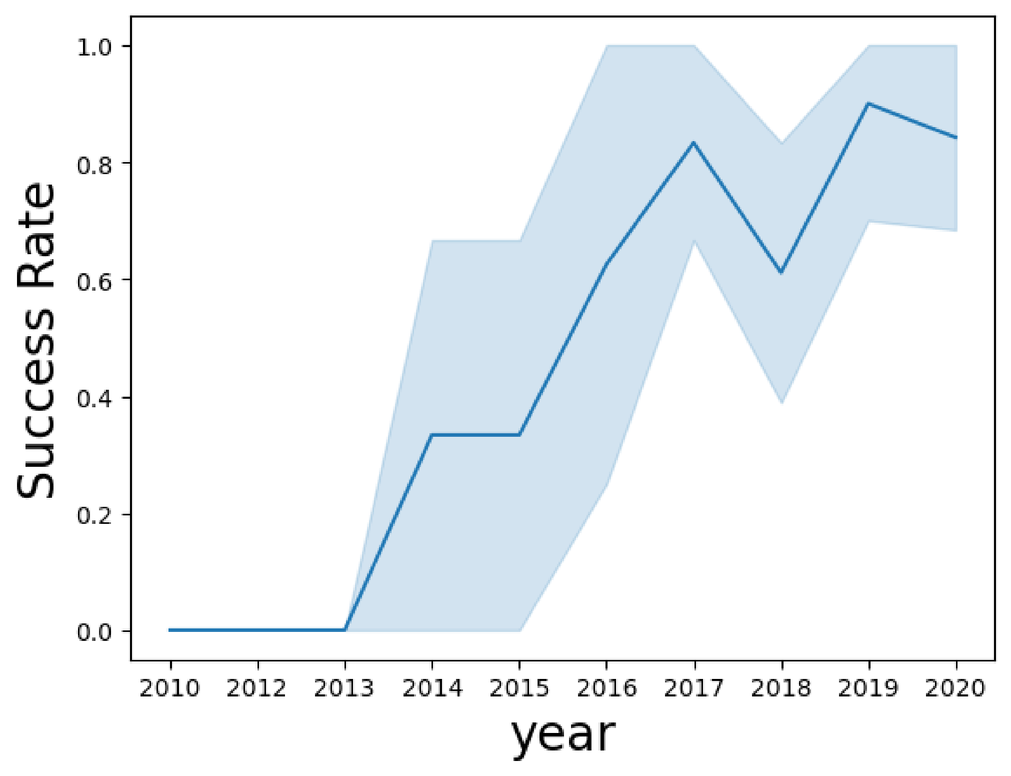
- With heavy payloads, the successful landing rate is higher for PO, LEO and ISS orbits

- The GTO orbit has not had a stable trend. Their results are dissimilar for any payloads mass.



Launch Success Yearly Trend

- Since 2013, the success rate has been significantly increasing; However, the success rate decreased in 2018 and 2020
- Since 2016 the success rate is above 50%



All Launch Site Names

- The names of the unique launch sites in the space mission are:
  - CCAFS LC-40
  - VAFB SLC-4E
  - KSC LC-39A
  - CCAFS SLC-40

- I used SELECT DISTINCT statement to obtain the unique launch sites using the 'LAUNCH_SITE' column of the SPACEXTBL table

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

Launch Site Names Begin with 'CCA'

- 5 records where launch sites begin with `CCA`

- I used SELECT statement and LIKE command to obtain the launch sites using the LAUNCH_SITE column begin with the string CCA% of the SPACEXTBL table

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**
Cu

2010- 06-04 18:45:00 F9 v1.0 B0003 CCAFS LC- 40 Dragon
Spacecraft Qualification Unit 0 LEO
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)
2012- 05-22 7:44:00 F9 v1.0 B0005 CCAFS LC- 40 Dragon demo
flight C2 525 LEO (ISS)
2012- 10-08 0:35:00 F9 v1.0 B0006 CCAFS LC- 40 SpaceX CRS-1
500 LEO (ISS)
2013- 03-01 15:10:00 F9 v1.0 B0007 CCAFS LC- 40 SpaceX CRS-2
677 LEO (ISS)

Total Payload Mass

- Total payload carried by boosters from NASA was 45.596 Kg
- I used SELECT statement and SUM function to obtain 'Total payload' using the PAYLOAD_MASS__KG_ column of the SPACEXTBL table where CUSTOMER = 'NASA (CRS)'

Task 3

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

```
%sql SELECT
SUM(PAYLOAD_MASS__KG_) \ FROM
SPACEXTBL \
WHERE CUSTOMER = 'NASA (CRS)';

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

SUM(PAYLOAD_MASS__KG_)

45596

Average Payload Mass by F9 v1.1

- The average payload mass carried by booster version F9 v1.1 was 2928.4 Kg

- I used SELECT statement and AVG function to obtain ‘Average payload’ using the PAYLOAD_MASS__KG_ column of the SPACEXTBL table where BOOSTER_VERSION = 'F9 v1.1'

Task 4
Display average payload mass carried by booster version F9 v1.1

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

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

AVG(PAYLOAD_MASS__KG_)
2928.4

First Successful Ground Landing Date

- The dates of the first successful landing outcome on ground pad was 2018-07-22

- I used SELECT statement and MIN function to obtain 'the dates of the first successful landing outcome on ground pad' using the `DATE` column of the SPACEXTBL table where `Landing_Outcome = 'Success'`

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 SPACEXTBL \ WHERE Landing_Outcome = 'Success';

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

min(DATE)
2018-07-22

Successful Drone Ship Landing with Payload between 4000 and 6000

- Names of booster version which have successfully landed on drone ship and had payload mass greater than 4000 but less than 6000
  - ✓ F9 FT B1022
  - ✓ F9 FT B1026
  - ✓ F9 FT B1021.2
  - ✓ F9 FT B1031.2
- I used SELECT statement to obtain ‘names of booster version’ using the ‘Booster_Version‘ column of the SPACEXTBL table where Landing_Outcome = 'Success (drone ship)' and PAYLOAD_MASS__KG_ was 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

```
%sql select Booster_Version from SPACEXTBL
\ where Landing_Outcome = 'Success (drone
ship)' \ and PAYLOAD_MASS__KG_ BETWEEN
4000 and 6000;

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

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

Total Number of Successful and Failure Mission Outcomes

- The total number of successful and failure mission outcomes was:
  - ✓ **Mission_Outcome total_number**
  - ✓ Failure (in flight) 1
  - ✓ Success 98
  - ✓ Success 1
  - ✓ Success (payload status unclear) 1
- I used SELECT statement and COUNT function to obtain 'Total successful and failure mission outcomes' using the 'MISSION_OUTCOME' column of the SPACEXTBL table grouping by the same column

Task 7

List the total number of successful and failure mission outcomes

```
%sql SELECT MISSION_OUTCOME, COUNT(*) as total_number \ FROM SPACEXTBL \ GROUP BY MISSION_OUTCOME;

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

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

Boosters Carried Maximum Payload

- Names of the booster version which have carried the maximum payload mass was:
  - ✓ 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
- I used SELECT statement and MAX function to obtain 'Names of the booster version ' using the 'BOOSTER_VERSION' column of the SPACEXTBL table with the maximum payload mass kg

Task 8

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

```
%sql select BOOSTER_VERSION as boosterversion \ from
SPACEXTBL \ where PAYLOAD_MASS__KG_=(select
max(PAYLOAD_MASS__KG_) from SPACEXTBL);

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

boosterversion

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

- The failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015 was:

month	Date	Booster_Version	Launch_Site	Landing_Outcome
5-	2015-01-10	F9 v1.1 B1012	CCAFS LC-40	Failure (drone ship)
5-	2015-04-14	F9 v1.1 B1015	CCAFS LC-40	Failure (drone ship

- I used SELECT statement to obtain for 2015 and failed landing outcome in dron ship using month, date, Booster version, Launch site and landing outcome column of the SPACEXTBL table.

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,4,2) as month, DATE,BOOSTER_VERSION,
LAUNCH_SITE, Landing_O FROM SPACEXTBL \ where Landing_Outcome
= 'Failure (drone ship)' and substr(Date,0,5)='2015';
```

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

month	Date	Booster_Version	Launch_Site	Landing_Outcome
5-	2015-01-10	F9 v1.1 B1012	CCAFS LC-40	Failure (drone ship)
5-	2015-04-14	F9 v1.1 B1015	CCAFS LC-40	Failure (drone ship)

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

- Rank the count of landing outcomes between the date 2010-06-04 and 2017-03-20 was
  - **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
- I used SELECT statement and COUNT function to obtain between 2010-06-04 and 2017-03-20 the count outcome in landing outcome using Landing_Outcome column of the SPACEXTBL table by count descending.

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 SPACEXTBL \ WHERE DATE BETWEEN '2010-06-04' AND
'2017-03-20' \ group by Landing_Outcome order by
count_outcomes DESC;

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

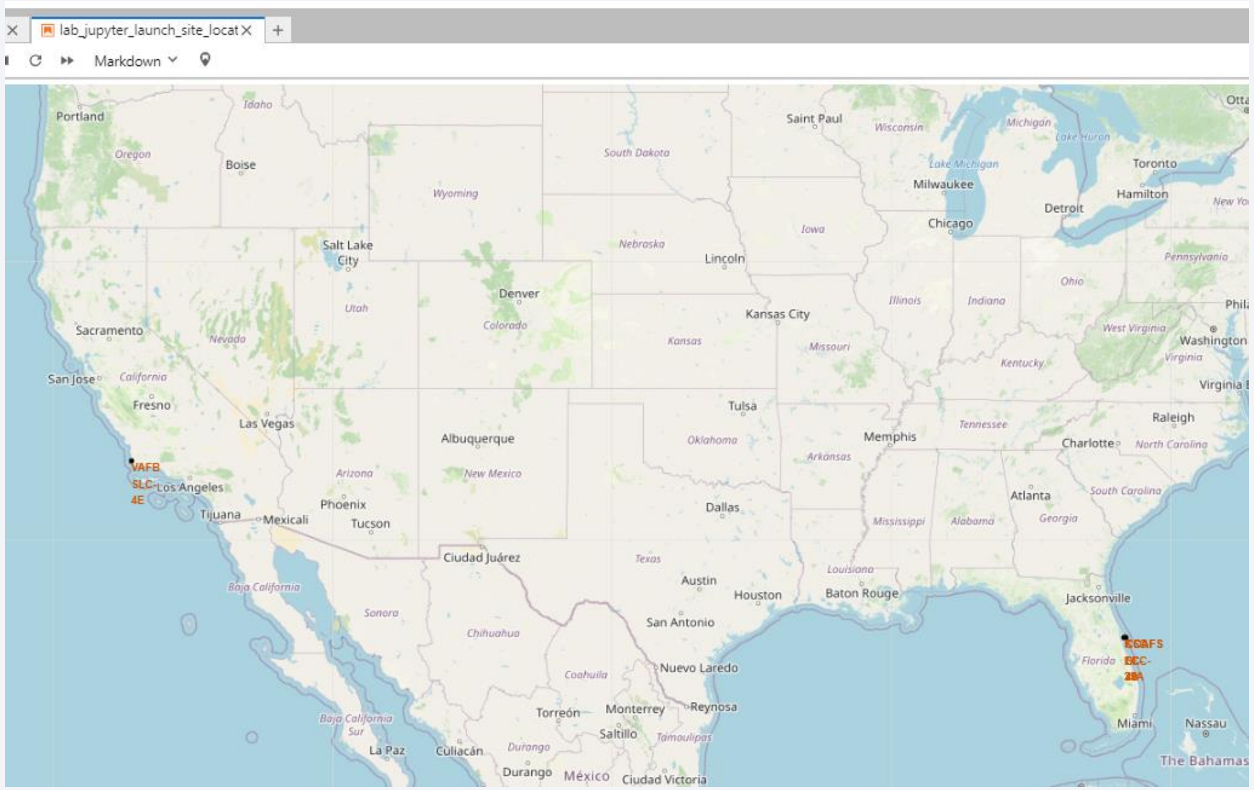
Landing_Outcome	count_outcomes
No attempt	10
Success (drone ship)	5
Failure (drone ship)	5
Success (ground pad)	3
Controlled (ocean)	3
Uncontrolled (ocean)	2
Failure (parachute)	2
Precluded (drone ship)	1

Section 3

Launch Sites Proximities Analysis

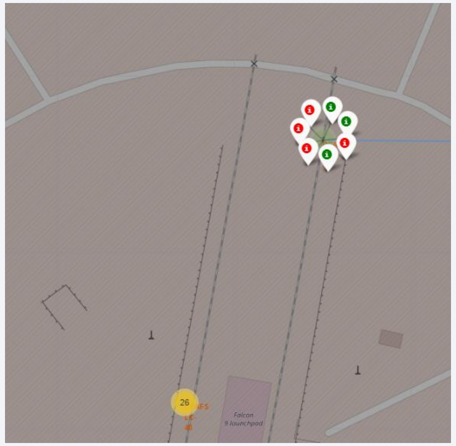
Markers of all SpaceX launch sites

- All launch sites are in near to the Equator, located southwards of the US.
- All the launch sites are very close to the coast.
- Launches from sites close to the equator and due to the speed of the Earth's rotation help to have lower fuel consumption.



Launch outcomes for each site

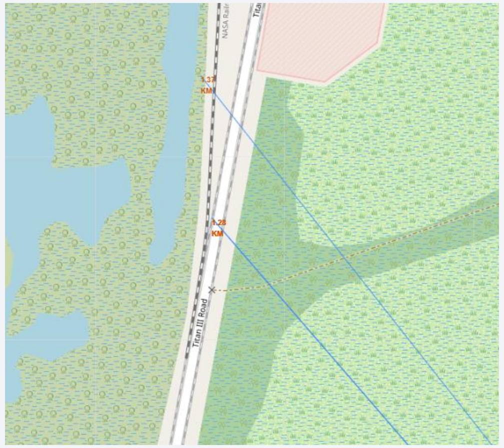
- **Green** markers for successful launches
- **Red** markers for unsuccessful launches
- The map shows that 3 of 7 launches for the CCAFS SLC-40 site have been successful
- Launch site KSC LC-39A has high success rates compared to CCAFS SLC-40 & CCAFS LC-40 in the Eastern coast (Florida)

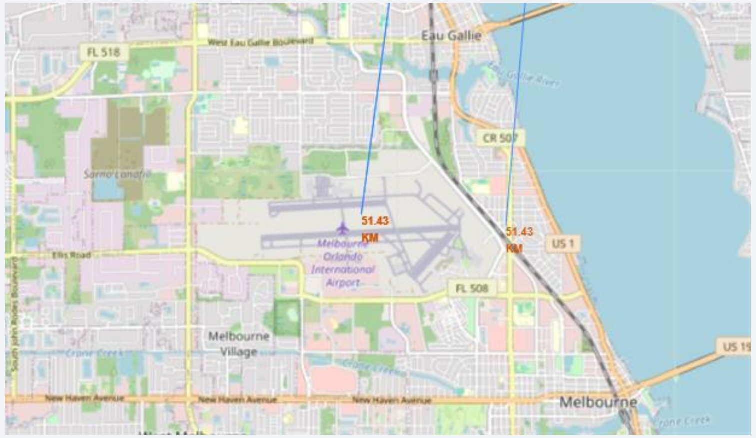


Distances between a launch site to its proximities

Site CCAFS SLC-40

- City Distance(Melbourne) 51.43 km
- Railway Distance(Nasa raillroad) 1.37 km
- Highway Distance(Samuel P HW) 0.67 km
- Coastline Distance 1.45 km





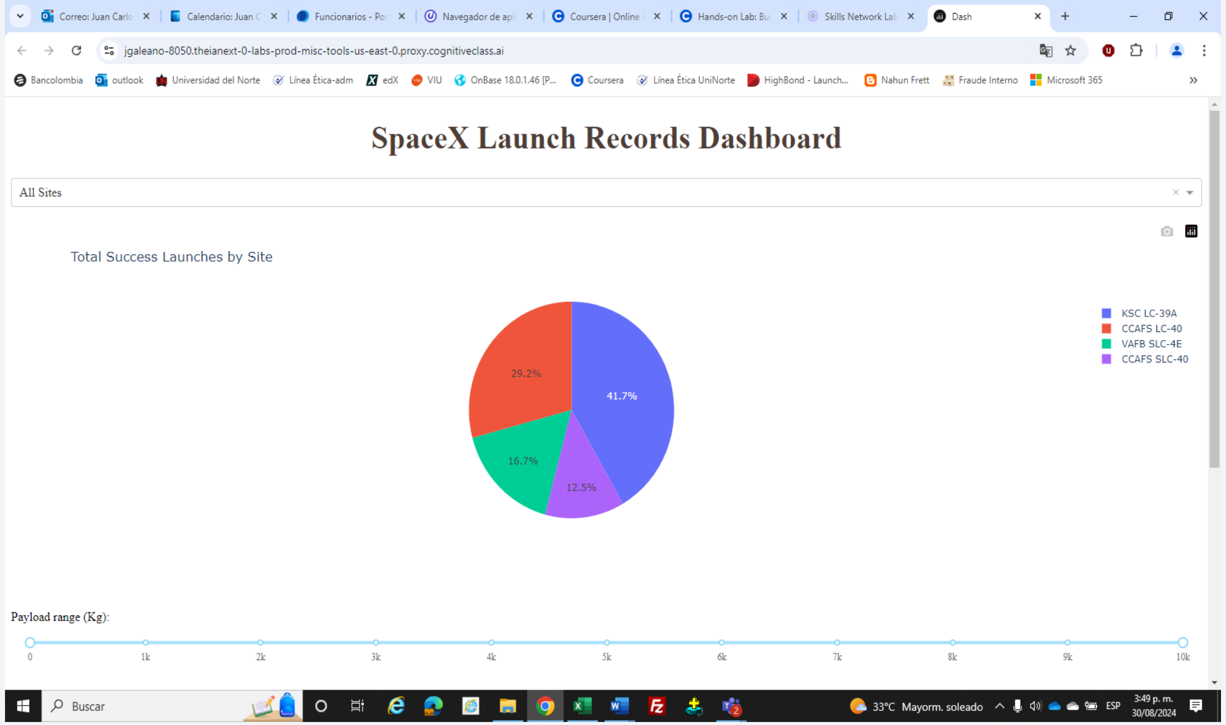
Section 4

Build a Dashboard with Plotly Dash

Launch success by site

Success as Percent of Total

- Launch site KSC LC-39A has the highest success rate at 41.7. The site CCAFS LC-40 has at 29.2%, VAFB SLC-4E has at 16.7% and CCAFS SLC-40 with a success rate of 12.5%

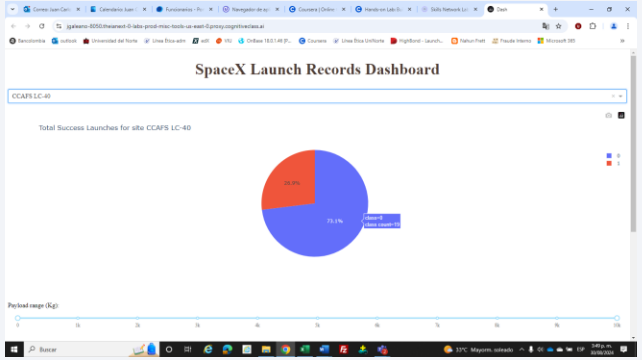


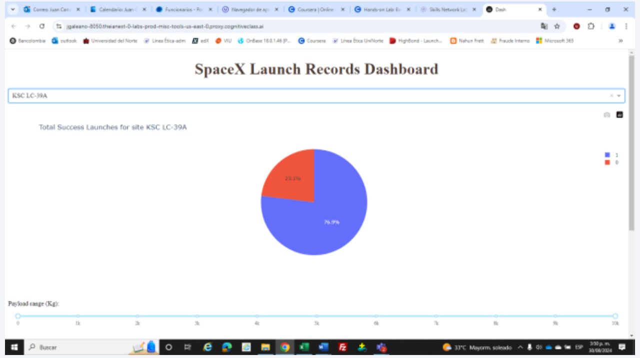
Launch site with success ratio

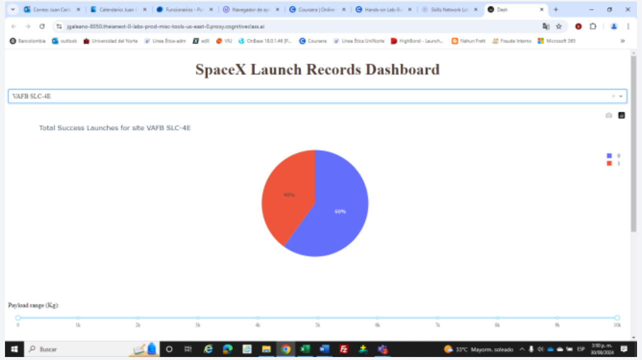
Success as Percent of Total by site:

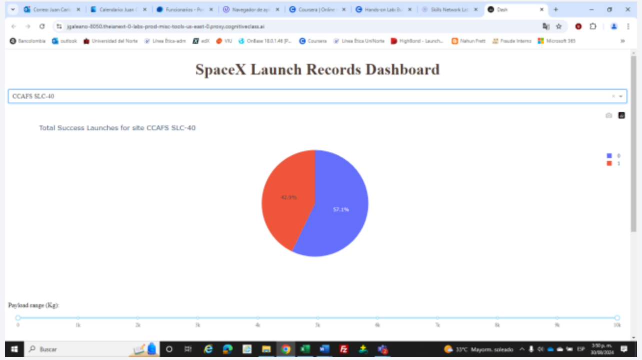
- KSC LC-39A : 76.9%
- CCAFS SLC-40 : 42.9%
- VAFB SLC-4E : 40.0%
- CCAFS SL-40 : 26.9%

The launch site highest success rate was KSC LC-39A 76.9%









Payload mass and success by bosster version

- Launch by booster version has the largest success rate from a payload mass greater than 2000 kg
- The highest success rate had by booster version F1

Section 5

Predictive Analysis (Classification)

Classification Accuracy

All the models (SVM, LR,KNN, decision tree) performed at about the same level and had the same scores and accuracy on the test data.

The accuracy score was 83.33%

TASK 12

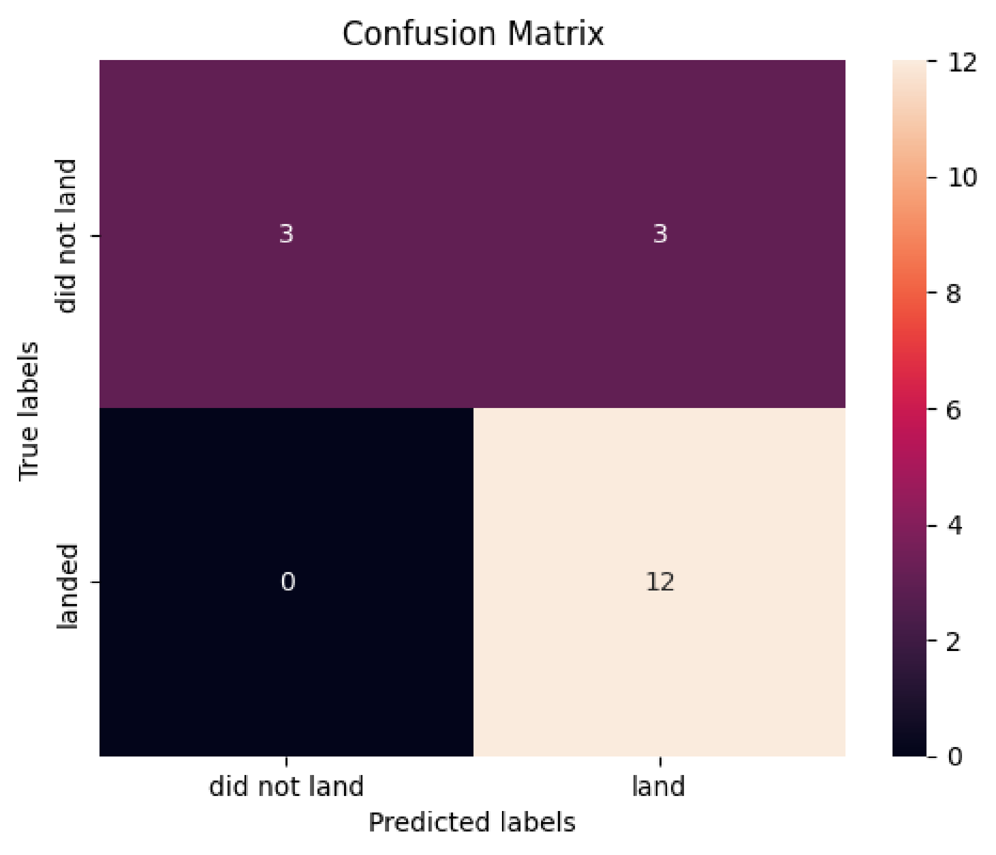
Find the method performs best:

```
accuracy = [svm_cv_score, logreg_score, knn_cv_score, tree_cv_score]

accuracy = [i * 100 for i in accuracy]

method = ['Support Vector Machine', 'Logistic Regression', 'K Nearest Neighbour',
'Decision Tree']

models = {'ML Method':method, 'Accuracy Score (%)':accuracy}

ML_df = pd.DataFrame(models)

ML_df
```

	ML Method	Accuracy Score (%)
0	Support Vector Machine	83.333333
1	Logistic Regression	83.333333
2	K Nearest Neighbour	83.333333
3	Decision Tree	83.333333

Confusion Matrix

- All the 4 classification model had the same confusion matrixes.
- A confusion matrix summarizes the performance of a classification algorithm
- Confusion Matrix Outputs were:
    - 12 True positive
    - 3 True negative
    - **3 False positive**
    - 0 False Negative



Conclusions

- The launch site are CCAFS LC-40 , VAFB SLC-4E, KSC LC-39A and CCAFS SLC-40. All launch sites are in near to the Equator, located southwards of the US and are very close to the coast.
- Launch site KSC LC-39A has the highest success rate at 41.7 of all site and your Success as Percent of Total by site was 76.9%
- Launch by booster version has the largest success rate from a payload mass greater than 2000 kg and the highest success rate had by booster version F1
- Since 2013, the success rate has been significantly increasing; However, the success rate decreased in 2018 and 2020. Since 2016 the success rate is above 50%
- Generally, the greater the number of launches, the greater the success rate of the launches in any of the types of orbit, except GTO orbit. Some orbits have had 100% success rates for any number of flights such as SSO, HEO y GEO. With heavy payloads, the successful landing rate is higher for PO, LEO and ISS orbits. The GTO orbit has not had a stable trend. Their results are dissimilar for any payloads mass. The orbit type SO was 0% Success Rate
- Normally, higher success rate is observed at higher payload mass (kg), around 16000 kg and most launches from VAFB SKC 4E are successful, but none have been launched above 10,000 kg.
- The launches for the VAFB SLC 4E site above 50 number of flights have been 100% successful, for the KSC LC 39A site and for the CCAFS SLC site above 40 number of flights
- All the models (SVM, LR,KNN, decision tree) performed at about the same level and had the same scores and accuracy on the test data. The accuracy score of the models was 83.33% and all models had the same confusion matrices. The main problem is false positives for all the models.

Thank you!