Winning Space Race with Data Science

Fernando Alfaro

June 30 2023

Table of Contents

IBM Developer SKILLS NETWORK

- Executive Summary....................Page 03
- Introduction....................Page 04
- **Section 1: Methodology**
- Methodology Summary....................Page 06
- Data Collection....................Page 07
- Data Collection – SpaceX API....................Page 09
- Data Collection – Scrapping....................Page 11
- Data Wrangling....................Page 14
- EDA with Data Visualization....................Page 15
- EDA with SQL....................Page 16
- Build an Interactive Map with Folium....................Page 17
- Build a Dashboard with Plotly Dash....................Page 18
- Predictive Analysis (Classification) ....Page 22
- **Section 2: Insights Drawn from Exploratory Data Analysis (EDA)**
- Flight Number vs. Launch Site....................Page 24
- Payload vs. Launch Site....................Page 25
- Success Rate vs. Orbit Type....................Page 26
- Flight Number vs. Orbit Type....................Page 27
- Payload vs. Orbit Type....................Page 28
- Launch Success Yearly Trend....................Page 29
- All Launch Site Names....................Page 30
- Launch Site Names Begin with ‘CCA’ ....Page 31
- Total Payload Mass....................Page 32

Table of Contents

IBM Developer SKILLS NETWORK

- Average Payload Mass by F9 v1.1..Page 33
- First Successful Ground Landing Date..Page 34
- Successful Drone Ship Landing with Payload Between 4000 and 6000....Page 35
- Total Number of Successful and Failure Mission Outcomes..................Page 36
- Boosters Carried Maximum Payload..Page 37
- 2015 Launch Records..Page 38
- Rank Landing Outcomes Between 2010/06/04 and 2017/03/20..........Page 39

- **Section 3: Launch Sites Proximities Analysis**

- Launch Sites of Falcon 9 Rockets..Page 42
- Launch Outcomes for Each Launch Site..Page 45
- Distance from Launch Site to its Proximities................................Page 48

- **Section 4: Build a Dashboard with Plotly Dash**

- Total Successful Launches by Launch Site....................................Page 50
- Successful vs Failed Landings Comparison Between Launch Sites...........Page 52
- Payload vs Launch Outcome Analysis..Page 54

- **Section 5: Predictive Analysis (Classification)**

- Classification Accuracy..Page 59
- Confusion Matrix..Page 60

- **Section 6: Results and Conclusions**

- Results...Page 62
- Conclusions..Page 63
- Other Sources...Page 64
- Thank you for Reading..Page 65

Executive Summary

The methodology used for this project is conformed of the following steps:

- Data Collection of the SpaceX Falcon 9 Launches using SpaceX API and Web scrapping through Wiki Pages to construct a Data frame.
- Transform the Data for modeling purposes through Data Wrangling.
- Perform an Exploratory Data Analysis (EDA) using Data Visualization and SQL Queries to get insights of the correlation of different variables in the Dataset, and their impact in the Success or Failure of recovering the first stage of a Falcon 9 rocket.
- For geospatial data and relationships between the outcomes and the Launch Sites, an interactive map with folium and a Dashboard with Plotly Dash were created to carry out an Interactive Data Analysis.
- To finish, four machine learning models (Logistic Regression, Support Vector Machine, Decision Tree and K-Nearest Neighbor) were trained and optimized with Grid Search to predict the outcome of future launches; finally, their accuracy was compared with each other to choose the one with better predictions.

The results obtained were:

- The most used site for launch is “CCSFS SLC 40” followed by “KSC LC 39A” and at last with “VAFB SLC 4E”, this has a relationship to the destination orbit of the payload, as LEO, ISS, GTO, HEO and ES-Lagrange points orbits are sent from “CCSFS SLC 40” and “KSC LC 39A” launch sites, while PO, SSO and LEO orbits with 60 degrees and beyond to the Earth’s equator are launched from “VAFB SLC 4E”.
- Most launches done with all models of Falcon 9 had a payload mass of less than 8,000 kg (74%), and every launch above this point used FT, B4 and B5 models, but have never gone for masses of 16,000 kg and beyond.
- There is a correlation between Success rate in each orbit and the number of launches performed to said destination, as less frequents orbits have better success rates than common ones.
- The relationship between the Payload Mass and the Orbit can be resumed as the closer the orbit the payload has to travel, the heavier it can be.
- The upwards yearly trend in Successful retrieval of first stage booster is caused by improvements done in newer Falcon 9 Models.
- Most launches are performed in a payload mass range from 1,500 kg to 5,500 kg and the success rate of the FT model have been improving with the introduction of newer Blocks (Block 4 and 5).
- Every Machine Learning model generates the same results with the actual Dataset: an accuracy of 83.34%, composed by 100% true positives and 50% true negatives, having a 50% of false positives.

This come to the conclusion that Predictive Models require to focus their training mainly in models from FT onwards, as this can provide a clearer picture of which conditions can assure that a rocket will be reused or not; also, two variables shown a pattern that could be helpful in future training of these models:

- **The Payload Mass (kg), which displays a maximum value which makes viable to recover a first stage booster or not.**
- **The times that a specific booster model have been used, as looking at the maximum reuse times of previous booster serials could give a better picture if the next use of a particular serial number will be the last or not.**

Introduction

In this capstone project for the "IBM Data Science Professional Certificate" **the public launch data of the Falcon 9 rockets from SpaceX is analyzed to understand if in future launches of the same rocket, the first stage will be reused**.

This is important to understand as a competing company in the aerospace industry, as it **lets it know which kind of launches SpaceX will spend more money by disposing the first stage of the rocket** and taking this information lets it concentrate its efforts in making this type of rocket launches cheaper than SpaceX, also this is useful if it wants to bid against SpaceX for this category of rocket launch.

But **why is it important to try to recover the first stage of a rocket launch?** Because this part is one of the most expensive parts in a rocket mission, as it is the one that gives the rocket enough propulsion to surpass the atmosphere and let the second stage put the payload into orbit; but normally this part is disposed as it is complicated to recover by the weight of it (the engines and the fuel containers are heavy and make them land without any kind of damage is difficult) and the risk it carries (it has fuel after all, a wrong landing means a high risk of an explosion).

SpaceX reduces their rocket launches costs by recovering and reusing the first stage of their Falcon 9 rockets, this means that SpaceX can advertise their rocket launches with a cost of 67 million dollars (through 2022), when other providers can cost around 165 million dollars. **But this price difference can only be achieved if a recovery is successful**, so the cost of a SpaceX Falcon 9 rocket launch can be predicted if it is known what kind of outcome a rocket will have based on the characteristics of that particular launch.

IBM Developer SKILLS NETWORK

Section 1

Methodology

Methodology Summary

Data Collection:

Data was collected by two methods: using the SpaceX Application Program Interface (API) and Web scrapping through Wiki pages to collect the rocket, payload, launchpad, launch and cores data, and assigning all of it into a Data Frame to work with it.

Data Wrangling:

The outcome of each launch was structured as a categorical variable, for modeling purposes, this data was transformed into numerical data for successful and failed Falcon 9 first stage recoveries.

Exploratory Data Analysis (EDA):

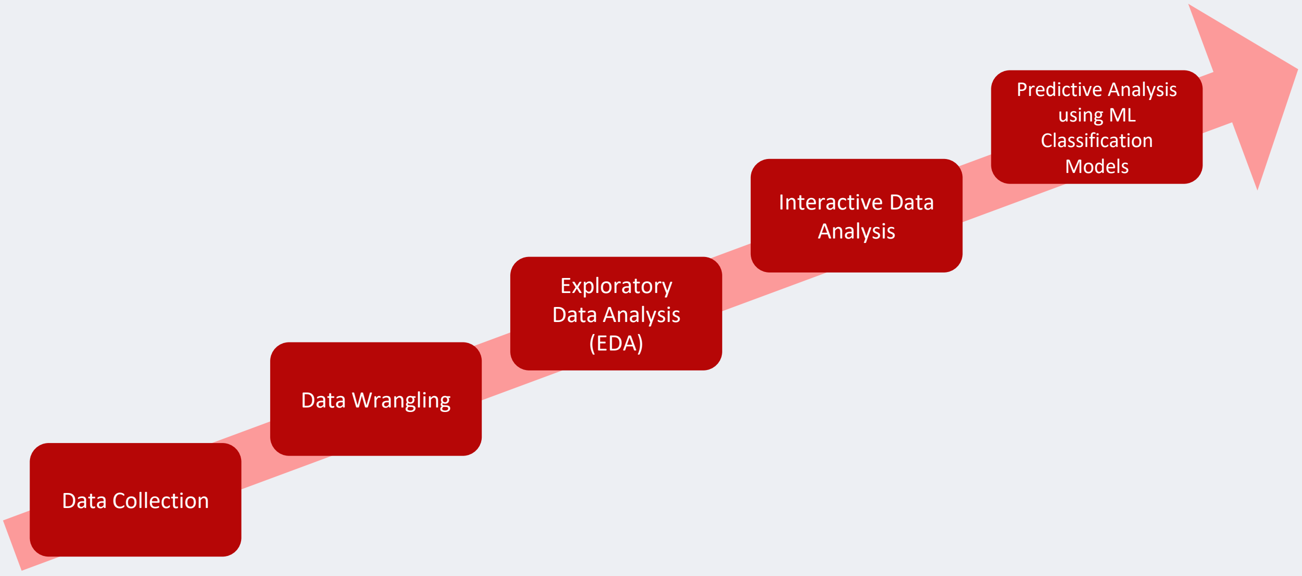
For better understanding of the data at hand, two approaches were used to see the correlation between the variables found in the launches: Structured Query Language (SQL) queries and Data Visualization. Both gave an inside look of the impact that each variable has in the outcome of the first stage recovery.

Interactive Data Analysis:

Instead of creating tons of plots or queries for specific data analysis, an interactive map with Folium was designed to review location, distances and outcomes of launches; and an interactive Dashboard with Plotly Dash to examine the total successful launches of each site, in addition to the category of booster version and the payload mass (in kg) of each launch.

Predictive Analysis using Machine Learning Classification Models:

To predict the outcome of future Falcon 9 launches, four different machine learning models (Logistic Regression, Support Vector Machines (SVM), Decision Tree and K-Nearest Neighbor (KNN)) were trained with the dataset collected previously and using the best parameters found with Grid Search for each model, and compared their results (accuracy and confusion matrix outcomes) with each other to see which one offers the best predictions.

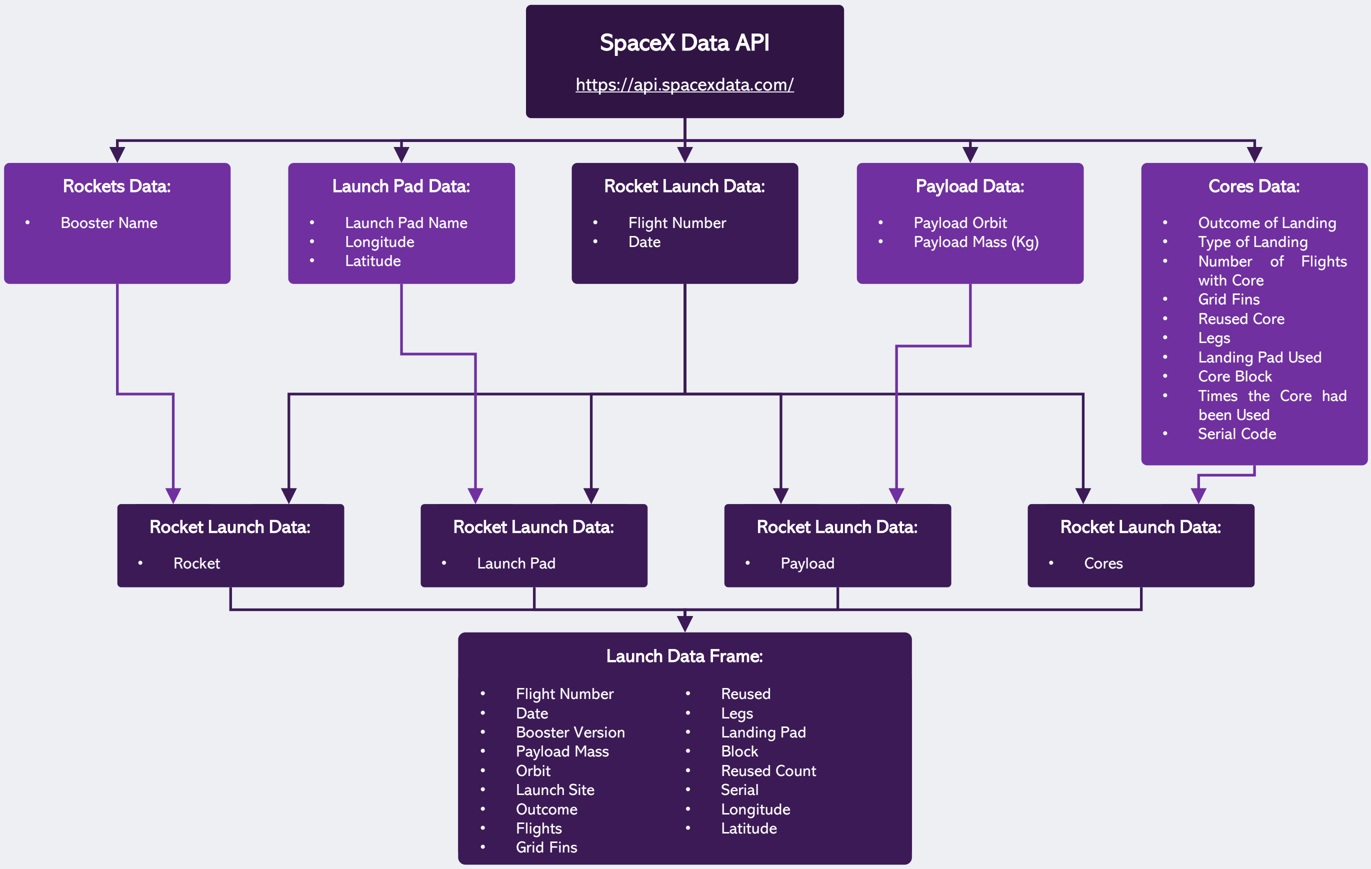


Data Collection

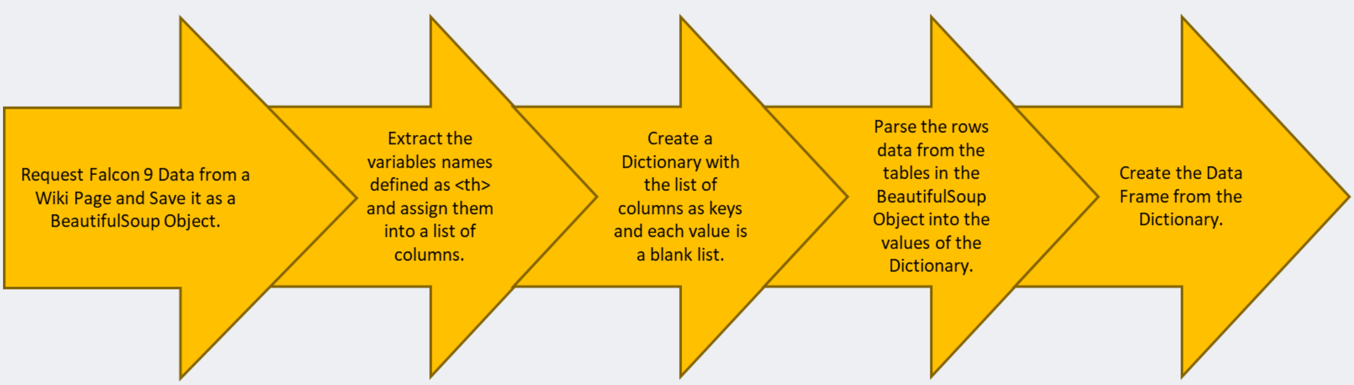
The datasets used for this project are collected from two sources:

- Requesting the data from SpaceX site using their API (https://api.spacexdata.com/) to get the following info:
  - All the rocket launch data from:
    https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DS0321EN-SkillsNetwork/datasets/API_call_spacex_api.json
    (this is done for consistency, as the data in that URL is static, if we want to use the most recent info we should use https://api.spacexdata.com/v4/launches/past).
  - Get the Booster Name by their respective ID by joining the rocket column from the rocket launches with their names in https://api.spacexdata.com/v4/rockets/.
  - Get the Launchpad Name, its Longitude and Latitude by its respective ID by joining the launchpad column from rocket launches with the Launchpad Data in https://api.spacexdata.com/v4/launchpads/.
  - Get the Payload Orbit and Mass Kg by the unique ID by joining the payloads column from rocket launches with the Payload Data in https://api.spacexdata.com/v4/payloads/.
  - Get the Outcome of the Landing and the Type of Landing, the number of flights with a respective core, if grid fins were used or not, whether the core is reused, if the core had legs or not, the Landing Pad used, the block of the core (this identifies the version of the core), the number of times the core has been used and the serial code of the core by joining the dictionary in the cores' column from rocket launches with the Cores Data in https://api.spacexdata.com/v4/cores/.

And finally, put it all together in a Data frame with only the Falcon 9 booster and without any missing values.



- Web scrapping the Falcon 9 and Heavy Launch Records from Wikipedia (https://en.wikipedia.org/wiki/List_of_Falcon_9_and_Falcon_Heavy_launches) following these steps:
    - Request the Falcon 9 data from: https://en.wikipedia.org/w/index.php?title=List_of_Falcon_9_and_Falcon_Heavy_launches&oldid=1027686922
      And save it as an HTML BeautifulSoup Object. This link refers to a snapshot from June 9, 2021; the data is collected from this wiki page for consistency with other peers' projects.
    - Extract all variable names from the HTML table header (<th> elements from the BeautifulSoup object), which will become the column names of the Data Frame.
    - Create a Data Frame by parsing all the launch tables in the BeautifulSoup Object into a Launch Dictionary with all the column names as keys and the values (rows) as Lists.



Data Collection – SpaceX API

The general rocket launch data was taken from the static URL with "requests.get" and decoded with ".json()" and then transformed into a Data Frame with "pd.json_normalize()":

```
In [11]: # Use json_normalize meethod to convert the json result into a dataframe
         StaticSpaceXData = requests.get(static_json_url)
         StaticSpaceXDataJSON = StaticSpaceXData.json()
         SpaceXDatadf = pd.json_normalize(StaticSpaceXDataJSON)
```

```
In [13]: data = SpaceXDatadf
```

Then the columns "rockets", "payloads", "launchpad" and "cores" are cleaned to just only their IDs, one of each, because when there are two or more, they refer to multiple rocket boosters, payloads, etc.

```
In [14]: # Lets take a subset of our dataframe keeping only the features we want and the flight number, and date_utc.
         data = data[['rocket', 'payloads', 'launchpad', 'cores', 'flight_number', 'date_utc']]

         # We will remove rows with multiple cores because those are falcon rockets with 2 extra rocket boosters and rows that have multiple payloads in a single rocket.
         data = data[data['cores'].map(len)==1]
         data = data[data['payloads'].map(len)==1]

         # Since payloads and cores are lists of size 1 we will also extract the single value in the list and replace the feature.
         data['cores'] = data['cores'].map(lambda x : x[0])
         data['payloads'] = data['payloads'].map(lambda x : x[0])

         # We also want to convert the date_utc to a datetime datatype and then extracting the date leaving the time
         data['date'] = pd.to_datetime(data['date_utc']).dt.date

         # Using the date we will restrict the dates of the launches
         data = data[data['date'] <= datetime.date(2020, 11, 13)]
```

After that, Blank Lists are created to store the data from the different API calls, and the requests for the "rockets", "payloads", "launchpad" and "cores" are done using functions defined previously:

```
In [16]: #Global variables
         BoosterVersion = []
         PayloadMass = []
         Orbit = []
         LaunchSite = []
         Outcome = []
         Flights = []
         GridFins = []
         Reused = []
         Legs = []
         LandingPad = []
         Block = []
         ReusedCount = []
         Serial = []
         Longitude = []
         Latitude = []
```

```
In [18]: # Call getBoosterVersion
         getBoosterVersion(data)
```

```
In [20]: # Call getLaunchSite
         getLaunchSite(data)

In [21]: # Call getPayloadData
         getPayloadData(data)

In [22]: # Call getCoreData
         getCoreData(data)
```

```
In [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'])
```

```
In [3]: # Takes the dataset and uses the launchpad column to call the API and append the data to the list
        def getLaunchSite(data):
            for x in data['launchpad']:
               if x:
                 response = requests.get("https://api.spacexdata.com/v4/launchpads/"+str(x)).json()
                 Longitude.append(response['longitude'])
                 Latitude.append(response['latitude'])
                 LaunchSite.append(response['name'])
```

```
In [4]: # Takes the dataset and uses the payloads column to call the API and append the data to the lists
        def getPayloadData(data):
            for load in data['payloads']:
               if load:
                response = requests.get("https://api.spacexdata.com/v4/payloads/"+load).json()
                PayloadMass.append(response['mass_kg'])
                Orbit.append(response['orbit'])
```

```
In [5]: # Takes the dataset and uses the cores column to call the API and append the data to the lists
        def getCoreData(data):
            for core in data['cores']:
                    if core['core'] != None:
                        response = requests.get("https://api.spacexdata.com/v4/cores/"+core['core']).json()
                        Block.append(response['block'])
                        ReusedCount.append(response['reuse_count'])
                        Serial.append(response['serial'])
                    else:
                        Block.append(None)
                        ReusedCount.append(None)
                        Serial.append(None)
                    Outcome.append(str(core['landing_success'])+' '+str(core['landing_type']))
                    Flights.append(core['flight'])
                    GridFins.append(core['gridfins'])
                    Reused.append(core['reused'])
                    Legs.append(core['legs'])
                    LandingPad.append(core['landpad'])
```

Data Collection – SpaceX API

With the variables retrieved, these are assigned as the values for a new Launch Dictionary, which in turn will become the Data Frame used next:

```python
In [23]: launch_dict = {'FlightNumber': list(data['flight_number']),
'Date': list(data['date']),
'BoosterVersion':BoosterVersion,
'PayloadMass':PayloadMass,
'Orbit':Orbit,
'LaunchSite':LaunchSite,
'Outcome':Outcome,
'Flights':Flights,
'GridFins':GridFins,
'Reused':Reused,
'Legs':Legs,
'LandingPad':LandingPad,
'Block':Block,
'ReusedCount':ReusedCount,
'Serial':Serial,
'Longitude': Longitude,
'Latitude': Latitude}
```

```python
In [24]: # Create a data from launch_dict
Launchdf = pd.DataFrame(launch_dict)
```

	FlightNumber	Date	BoosterVersion	PayloadMass	Orbit	LaunchSite	Outcome	Flights	GridFins	Reused	Legs	LandingPad	Block	ReusedCount	Serial	Longitude	Latitude
0	1	2006-03-24	Falcon 1	20.0	LEO	Kwajalein Atoll	None None	1	False	False	False	None	NaN	0	Merlin1A	167.743129	9.047721
1	2	2007-03-21	Falcon 1	NaN	LEO	Kwajalein Atoll	None None	1	False	False	False	None	NaN	0	Merlin2A	167.743129	9.047721
2	4	2008-09-28	Falcon 1	165.0	LEO	Kwajalein Atoll	None None	1	False	False	False	None	NaN	0	Merlin2C	167.743129	9.047721
3	5	2009-07-13	Falcon 1	200.0	LEO	Kwajalein Atoll	None None	1	False	False	False	None	NaN	0	Merlin3C	167.743129	9.047721
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

Finally, the Data Frame is filtered to only contain Falcon 9 Launches and the missing values in the Payload Mass is substituted with the mean value of all the Payload Masses Launched, leaving only the Launch Pad missing values as they are because these mean that it was not used.

```python
In [29]: # Hint data['BoosterVersion']!='Falcon 1'
data_falcon9 = Launchdf[Launchdf['BoosterVersion']!='Falcon 1']
data_falcon9.head()
```

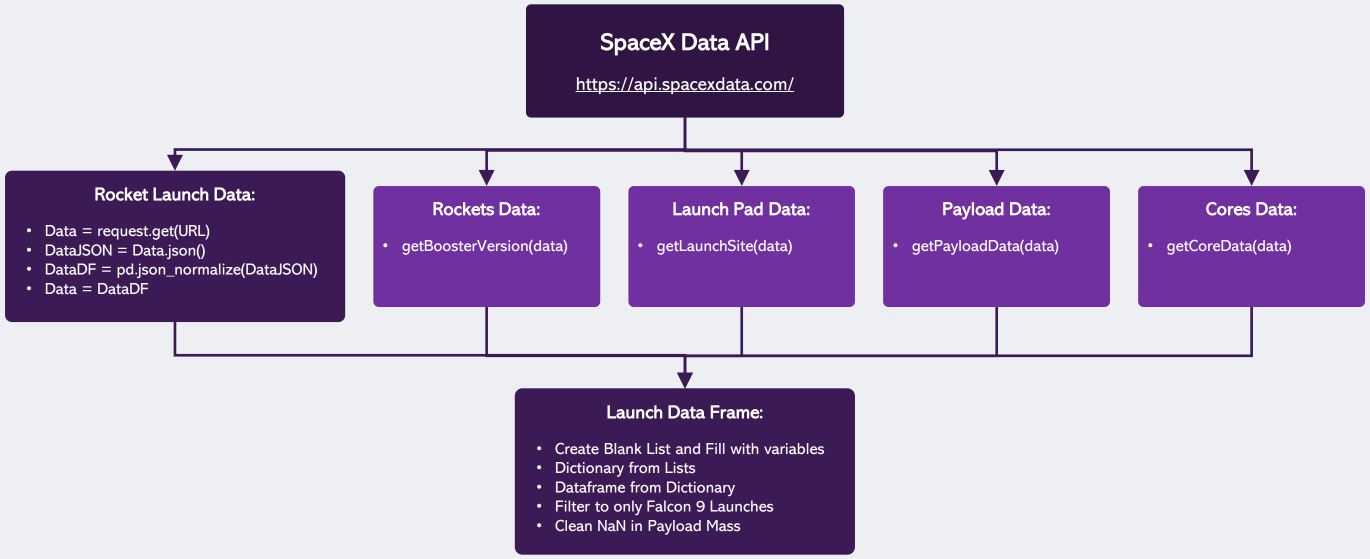
```python
In [30]: data_falcon9.loc[:,'FlightNumber'] = list(range(1, data_falcon9.shape[0]+1))
data_falcon9
```

```python
In [32]: # Calculate the mean value of PayloadMass column
PayloadMassMean = data_falcon9['PayloadMass'].mean()
print(PayloadMassMean)

6123.547647058824
```

```python
In [34]: # Replace the np.nan values with its mean value
data_falcon9['PayloadMass'].replace(np.nan, PayloadMassMean, inplace=True)
df_falcon9plm = data_falcon9
df_falcon9plm.head()
```

FlightNumber	Date	BoosterVersion	PayloadMass	Orbit	LaunchSite	Outcome	Flights	GridFins	Reused	Legs	LandingPad	Block	ReusedCount	Serial	Longitude	Latitude
1	04/06/2010	Falcon 9	6123.54765	LEO	CCSFS SLC 40	None None	1	False	False	False		1	0	B0003	-80.577366	28.5618571
2	22/05/2012	Falcon 9	525	LEO	CCSFS SLC 40	None None	1	False	False	False		1	0	B0005	-80.577366	28.5618571
3	01/03/2013	Falcon 9	677	ISS	CCSFS SLC 40	None None	1	False	False	False		1	0	B0007	-80.577366	28.5618571
4	29/09/2013	Falcon 9	500	PO	VAFB SLC 4E	False Ocean	1	False	False	False		1	0	B1003	-120.610829	34.632093
5	03/12/2013	Falcon 9	3170	GTO	CCSFS SLC 40	None None	1	False	False	False		1	0	B1004	-80.577366	28.5618571



GitHub Link for the Jupyter Notebook: [Applied_Data_Science_Capstone/01-Data Collection (with API).ipynb at main · ChillDev93/Applied_Data_Science_Capstone · GitHub]

Data Collection – Scraping

The general rocket launch data was taken from the static URL with "requests.get" and extracted as simple text with ".text", and the object was created with BeautifulSoup using the "html.parser"

```
In [8]: # use requests.get() method with the provided static_url
        falcon9launchdata = requests.get(static_url)
        # assign the response to a object
        falcon9launchdata
```

```
In [9]: # Use BeautifulSoup() to create a BeautifulSoup object from a response text content
        falcon9launchdata = falcon9launchdata.text
        soup = BeautifulSoup(falcon9launchdata, 'html.parser')
```

Then, all the tables fetched in the Soup Object are saved in a variable and the third table, which contains the data we require, is saved separately.

```
In [11]: # 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')
         html_tables
```

```
In [12]: # Let's print the third table and check its content
         first_launch_table = html_tables[2]
         print(first_launch_table)
```

From this third table, all the column names are extracted using the table header:

```
In [16]: column_names = []

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

         for header in first_launch_table.find_all('th'):
             name = extract_column_from_header(header)
             if name is not None and len(name) > 0:
                 column_names.append(name)
```

For the Data Frame, first a Dictionary is created with blank list values, and then the values are filled using the following code:

```
In [18]: launch_dict= dict.fromkeys(column_names)

         # Remove an irrelvant column
         del launch_dict['Date and time ( )']

         # Let's initial the launch_dict with each value to be an empty list
         launch_dict['Flight No.'] = []
         launch_dict['Launch site'] = []
         launch_dict['Payload'] = []
         launch_dict['Payload mass'] = []
         launch_dict['Orbit'] = []
         launch_dict['Customer'] = []
         launch_dict['Launch outcome'] = []
         # Added some new columns
         launch_dict['Version Booster']=[]
         launch_dict['Booster landing']=[]
         launch_dict['Date']=[]
         launch_dict['Time']=[]
```

Data Collection – Scraping

```
In [28]: extracted_row = 0
#Extract each table
for table_number,table in enumerate(soup.find_all('table',"wikitable plainrowheaders collapsible")):
   # get table row
    for rows in table.find_all("tr"):
        #check to see if first table heading is as number corresponding to launch a number
        if rows.th:
            if rows.th.string:
                flight_number=rows.th.string.strip()
                flag=flight_number.isdigit()
        else:
            flag=False
        #get table element
        row=rows.find_all('td')
        #if it is number save cells in a dictonary
        if flag:
            extracted_row += 1
            # Flight Number value
            # TODO: Append the flight_number into launch_dict with key `Flight No.`
            print(flight_number)
            launch_dict['Flight No.'].append(flight_number)

            datatimelist=date_time(row[0])

            # Date value
            # TODO: Append the date into launch_dict with key `Date`
            date = datatimelist[0].strip(',')
            print(date)
            launch_dict['Date'].append(date)

            # Time value
            # TODO: Append the time into launch_dict with key `Time`
            time = datatimelist[1]
            print(time)
            launch_dict['Time'].append(time)

            # Booster version
            # TODO: Append the bv into launch_dict with key `Version Booster`
            bv=booster_version(row[1])
            if not(bv):
                bv=row[1].a.string
            print(bv)
            launch_dict['Version Booster'].append(bv)

            # Launch Site
            # TODO: Append the bv into launch_dict with key `Launch Site`
            launch_site = row[2].a.string
            print(launch_site)
            launch_dict['Launch site'].append(launch_site)

            # Payload
            # TODO: Append the payload into launch_dict with key `Payload`
            payload = row[3].a.string
            print(payload)
            launch_dict['Payload'].append(payload)

            # Payload Mass
            # TODO: Append the payload_mass into launch_dict with key `Payload mass`
            payload_mass = get_mass(row[4])
            print(payload_mass)
            launch_dict['Payload mass'].append(payload_mass)

            # Orbit
            # TODO: Append the orbit into launch_dict with key `Orbit`
            orbit = row[5].a.string
            print(orbit)
            launch_dict['Orbit'].append(orbit)

            # Customer
            # TODO: Append the customer into launch_dict with key `Customer`
            customer = row[6].text.strip()
            print(customer)
            launch_dict['Customer'].append(customer)

            # Launch outcome
            # TODO: Append the launch_outcome into launch_dict with key `Launch outcome`
            launch_outcome = list(row[7].strings)[0]
            print(launch_outcome)
            launch_dict['Launch outcome'].append(launch_outcome)

            # Booster Landing
            # TODO: Append the launch_outcome into launch_dict with key `Booster landing`
            booster_landing = landing_status(row[8])
            print(booster_landing)
            launch_dict['Booster landing'].append(booster_landing)
```

Data Collection – Scraping

Finally, the Data Frame is formed from the Dictionary:

```
In [30]: df= pd.DataFrame({ key:pd.Series(value) for key, value in launch_dict.items() })

In [31]: df.head()
```

Out[31]:

	Flight No.	Launch site	Payload	Payload mass	Orbit	Customer	Launch outcome	Version Booster	Booster landing	Date	Time
0	1	CCAFS	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success\n	F9 v1.0B0003.1	Failure	4 June 2010	18:45
1	1	CCAFS	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success\n	F9 v1.0B0003.1	Failure	4 June 2010	18:45
2	1	CCAFS	Dragon	0	LEO	NASA	Success	F9 v1.0B0003.1	No attempt\n	4 June 2010	18:45
3	1	CCAFS	Dragon	525 kg	LEO	NASA	Success	F9 v1.0B0003.1	No attempt	4 June 2010	18:45
4	2	CCAFS	SpaceX CRS-1	4,700 kg	LEO	NASA	Success\n	F9 v1.0B0004.1	No attempt\n	8 December 2010	15:43

Beautiful Soup Object

- “request.get” from Static URL
- Extract the info from the URL as plain text with “.text”
- Creation of the Beautiful Soup Object with “html.parser”

Column Names

- Extracted from third table in the Beautiful Soup Object Tables using the table header.

Data Parsing

- A Dictionary is created to store the data parsed from the tables.
- Every item in the rows from the tables is extracted with an iterative code.

Data Frame Creation

- The Dataframe is created from the dictionary filled with values.

GitHub Link for the Jupyter Notebook: Applied_Data_Science_Capstone/02-Data Collection (with Web Scraping).ipynb at main · ChillDev93/Applied_Data_Science_Capstone · GitHub

Data Wrangling

As the main purpose of this project is to understand the patterns that lead to a specific landing outcome with the first stage of the Falcon 9 rockets, it is important to see the different cases that can happen during the landing of this part and transform the data into values that can be used for training for the Machine Learning step.

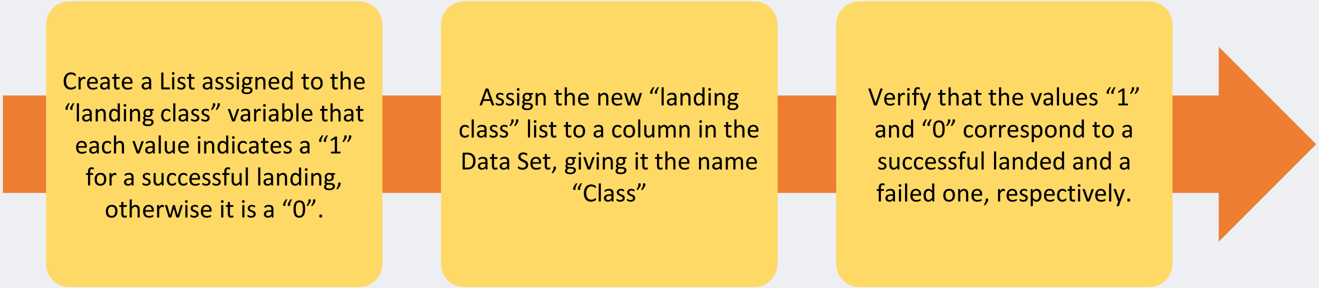
With the Data Set taken directly from the SoaceX API, the unique values in the Outcome column are:

```
In [10]: # landing_outcomes = values on Outcome column
         landing_outcomes = df['Outcome'].value_counts()
         landing_outcomes

Out[10]: True ASDS      41
         None None      19
         True RTLS      14
         False ASDS      6
         True Ocean      5
         False Ocean     2
         None ASDS       2
         False RTLS      1
         Name: Outcome, dtype: int64
```

This means that the Data Set has **three possible outcomes**: **True** for successful landings, **False** for unsuccessful landings **and None** for the cases that it did not land at all; the second part indicates where it was intended to land, “ASDS” for drone ships, “RTLS” for ground pads and “Ocean” for the ocean, “None” in this case means that the first stage was never intended to land.

Because the main concern is if the first stage landed or not, and not where it landed, the transformation of this values will consider **True as successful (in binary this is ”1”)** and **False and None as unsuccessful (in binary, “0”)**, for this transformation the following steps are done:



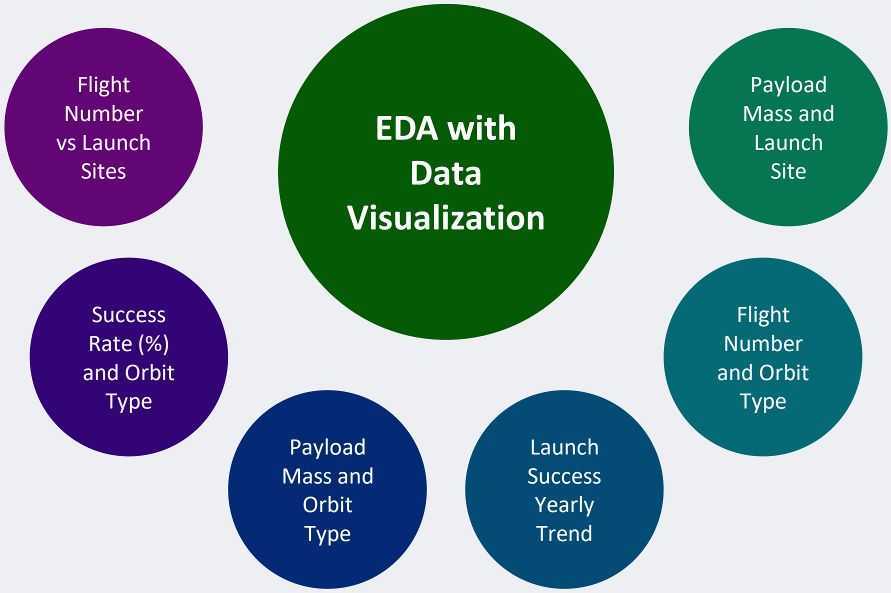
GitHub Link for the Jupyter Notebook: [Applied_Data_Science_Capstone/03-Data_Wrangling.ipynb at main · ChillDev93/Applied_Data_Science_Capstone · GitHub]

EDA with Data Visualization

To understand the Data Set in general an Exploratory Data Analysis (EDA) was performed with plots and SQL queries.

In the first instance the following charts were chose:

- **Flight Number vs Launch Site**: It is relevant to understand the tendency between the launch number, the place where it was performed and the outcome, to see if the outcomes were improving over time and which places saw a better or worse performance with every consecutive launch.
- **Payload Mass and Launch Site:** To recognize if the place where the rocket was launched and the amount of mass it must carry has an influence in the outcome that the first stage had; the launch sites having the ocean in the east have a better outcome that the ones in the west side?
- **Success Rate (%) and Orbit type:** Does a closer orbit to the planet has a higher success rate in returning the first stage that the ones with an orbit far away? or Is it the other way round?; the only way to find out is analyzing their success rate percentage.
- **Flight Number and Orbit Type:** Can we be sure that the success rate saw previously is not influenced by the number of flights that have been done to a certain orbit? What if some orbits have more launches that others, and that has repercussions in their success rate?, this chart will reflect if there is a relationship between flight numbers and success rate in a specific orbit.
- **Payload Mass and Orbit Type::** The same happens with the Payload Mass, Do carrying a heavier mass to a further orbit causes a higher failure rate in recovering the first stage?, knowing if there is a relationship between payload mass and the success rate in a precise orbit is relevant in this case.
- **Launch Success Yearly Trend:** It is important to consider if the launches have improved their outcomes each year, or if they have been having a worse performance; also, a decline in specific years can give a clearer picture in the progression of Falcon 9 Launches.



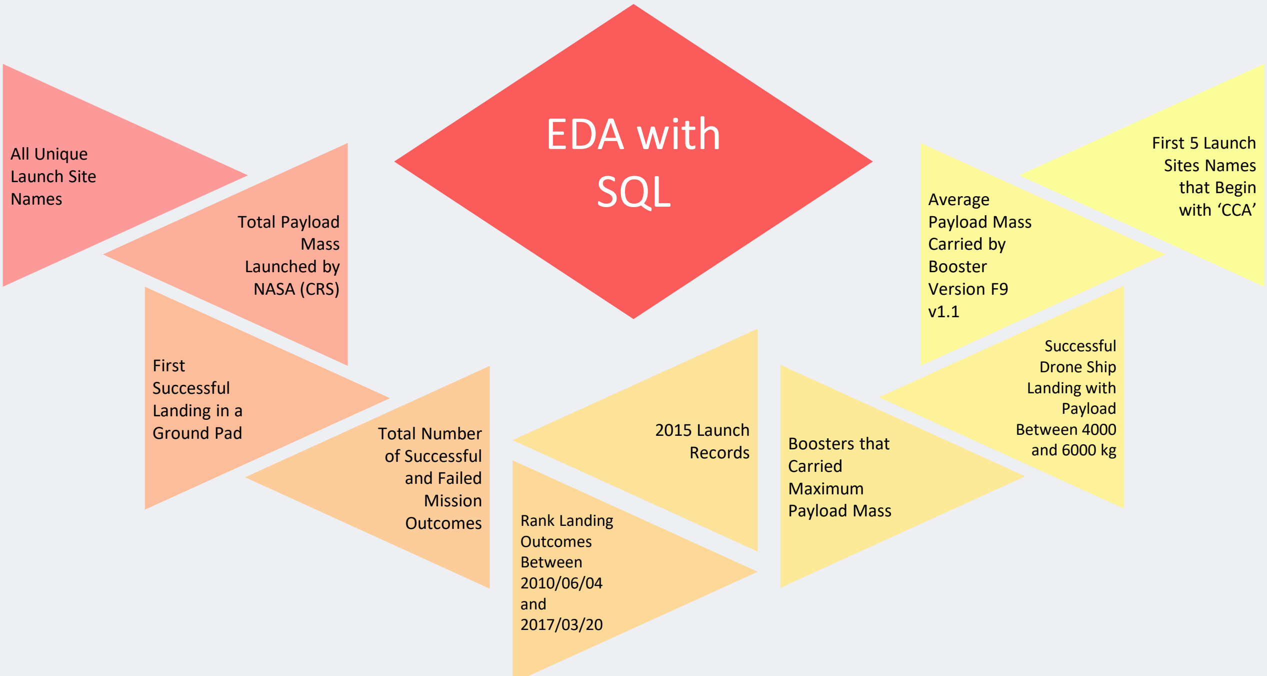
GitHub Link for the Jupyter Notebook: Applied_Data_Science_Capstone/05-Exploratory_Data_Analysis_with Pandas and Matplotlib.ipynb at main · ChillDev93/Applied_Data_Science_Capstone · GitHub

EDA with SQL

The other part of the Exploratory Data Analysis is the SQL Queries done to the Data Set, in this case, the information collected and arranged was transferred to a Database in which we could do the queries required.

The queries performed to the Data Set are:

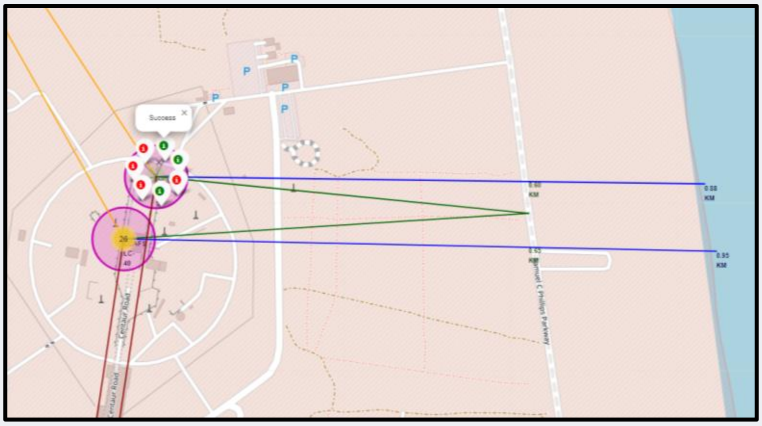
- **All Unique Launch Site Names:** To see from where the Falcon 9 rockets were launched, and if there is no duplicates in the Data that were not identified in previous steps.
- **First 5 Launch Sites Names that Begin with 'CCA':** Verifying that there are no special characters used in the launch site names that could cause a conflict in following operations.
- **Total Payload Mass Launched by NASA (CRS):** Understanding which has been the amount of payload mass that the NASA has sent using SpaceX Falcon 9 rockets.
- **Average Payload Mass Carried by Booster Version F9 v1.1:** Knowing the average amount of payload mass that this revision of the original F9 booster has carried.
- **First Successful Landing in a Ground Pad:** To check the date when the first successful landing was achieved in a ground pad.
- **Successful Drone Ship Landing with Payload Between 4000 and 6000 kg:** Listing the names of the boosters that had a successful landing in drone ships while carrying payloads from 4000 kg to 6000 kg.
- **Total Number of Successful and Failed Mission Outcomes:** Recognize how many of the missions done with the Falcon 9 rockets have been successful.
- **Boosters that Carried Maximum Payload Mass:** A list of the booster versions that have carried the maximum payload mass.
- **2015 Launch Records:** Displaying the records of the launches performed in 2015.
- **Rank Landing Outcomes Between 2010/06/04 and 2017/03/20:** Observe how many of each outcome have occurred between these dates.



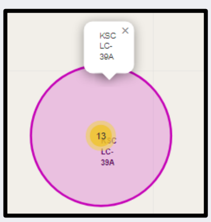
GitHub Link for the Jupyter Notebook: [Applied_Data_Science_Capstone/04-Exploratory_Data_Analysis_with SQL.ipynb at main · ChillDev93/Applied_Data_Science_Capstone · GitHub]

Build an Interactive Map with Folium

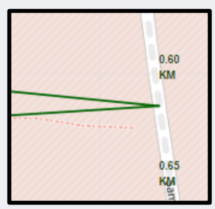
To easily visualize the localization of the launch sites in the US, how many launches has been carried in each one and the outcome from them, as well as the distance of some points of interest (for example, the closest coast, city, railway or roadway), an interactive map was built with different markers, circles and lines.



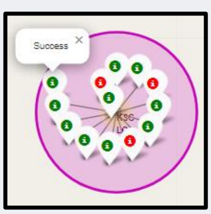
The meaning and the reason of each one of the markers, circles and lines is described next:



The purple circles indicate the area where a respective launch site is located, when it is clicked the name is displayed. The yellow circle is a cluster of all the launches done around it, this means that it groups all successful and failed launches around it, in this example from Launch Site KSC LC-39A, this means there have been 13 launches around this site.

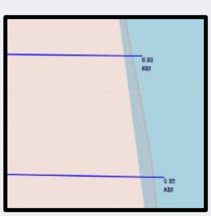


The green lines represent the distance between the launch site and the closest roadway or parkway, at the end closer to the road the exact distance is displayed in kilometers.

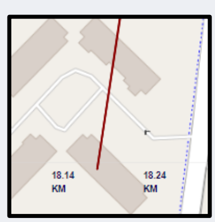


When the yellow circle is clicked, it displays all the launch markers inside it, in this case 13 of them. The launch markers are green if it was successful and red if it was a failure, also when the marker is clicked it gives a reminder of what the color means.

The yellow lines mark the distance from the launch pad to the closest railroad, and the exact distance is shown in the end closer to the railroad.



The blue lines refer to the closest coast to the launch site, and the distance from the launch site to it is marked at the end closer to the coast.



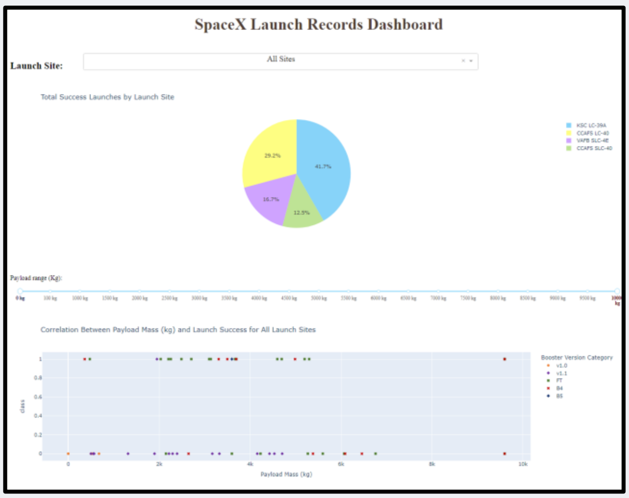
The red lines indicate the distance from a launch pad to the nearest city or town, and the exact distance in kilometers is displayed in the end of the line closer to the city or town.

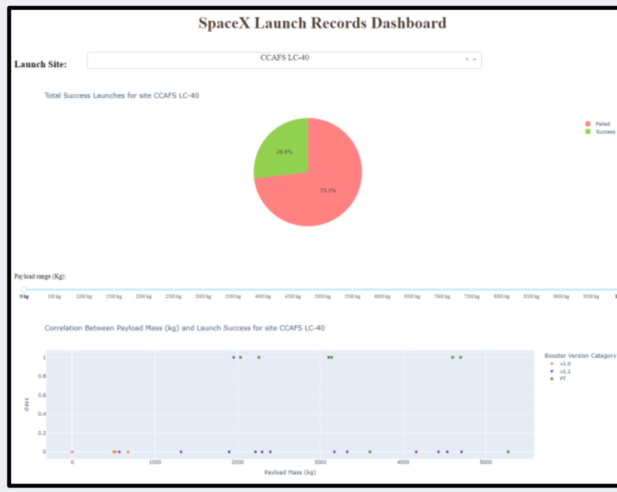
GitHub Link for the Jupyter Notebook: [Applied_Data_Science_Capstone/06-Interactive Visual Analytics with Folium.ipynb at main · ChillDev93/Applied_Data_Science_Capstone · GitHub]

If there is a problem visualizing the map in the GitHub viewer, please download the file and open it or download the html version included in the repository and open it in browser.

The **analysis of the total successful launches of each site is relevant to understand if the place where the Falcon 9 rockets take off has an influence in the success rate**; at the same time, there is a possible correlation that has not been reviewed in previous steps: **Does the version of the booster used and the payload mass that it carried have an effect in the success rate of each launch site?**

These two important elements would need various charts to explain, as there is four launch sites and five categories of booster versions to go through, and making charts for each one would make the comparison needlessly complicated so, **to facilitate the analysis of these crucial components, the total successful launches made in each site and the correlation between these, and the payload mass and booster version are explained using two interactive charts in a "Plotly Dash" Dashboard.**

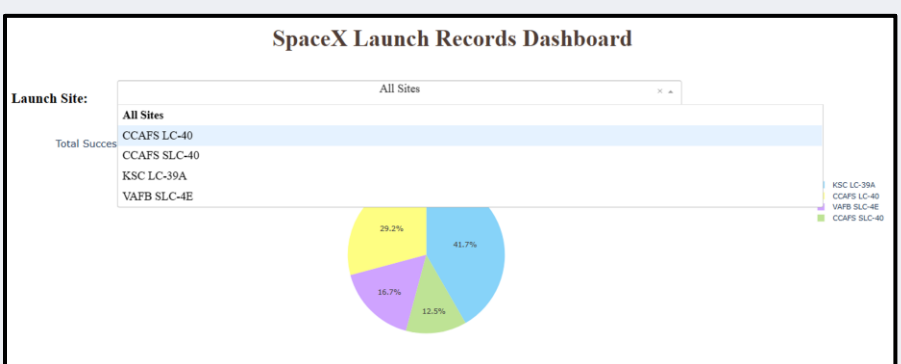




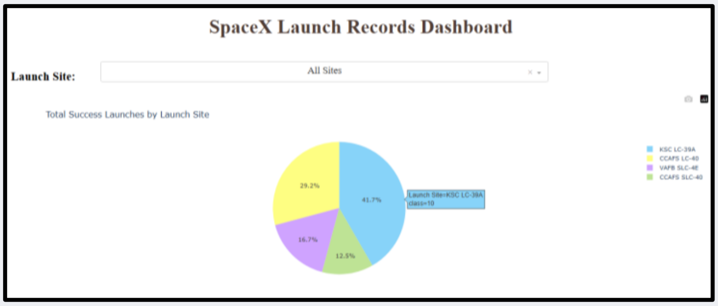
The main advantages that an interactive dashboard have are that **with simply selecting the launch site and the range of payload mass of interest, it creates the two charts** (the pie chart for the total successful launches and the scatterplot for the correlation between these and the booster version with a certain amount of mass) **in real time**, and not only that, but **they are completely interactive**, which means that it simplifies the study of these elements.

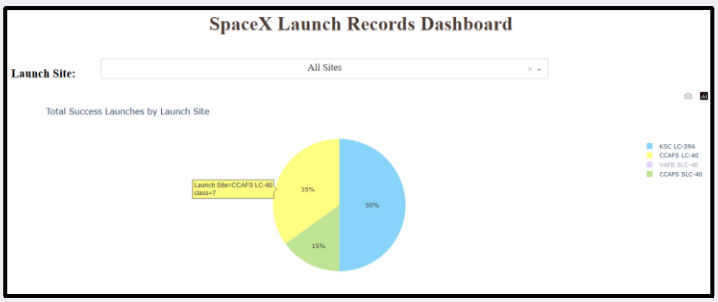
To understand this Dashboard, it is essential to explain the parts that compose it:

- The initial dropdown list presents all the options for launch sites, from all of them to specify if the information displayed should be just for one of them:

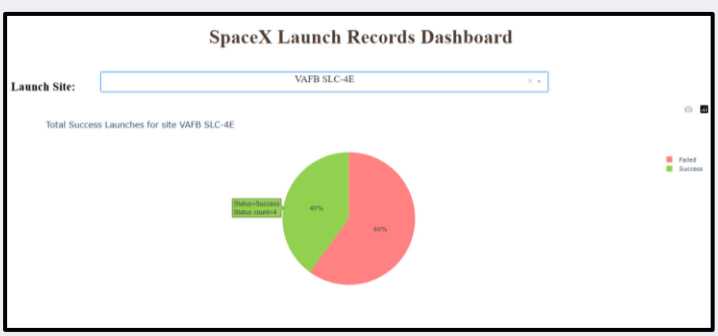


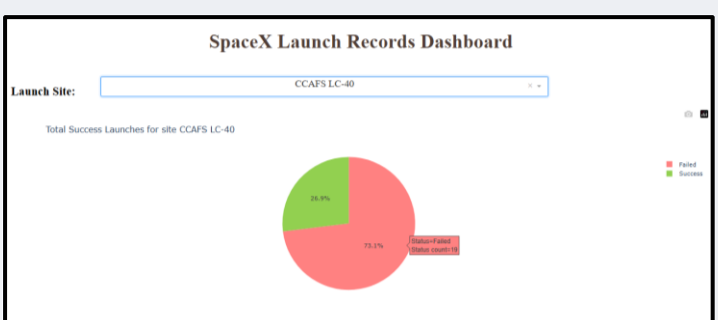
If all sites are selected, the pie chart displays the successful launches from all of them and the percentage of contribution for each site, if the mouse pointer hovers over one slice of the pie chart it displays the name of the site that it corresponds and the number of successful launches that it has; also, it is possible to select the sites to analyze by clicking on the names in the legend part, with this filter it can be chosen only the east side launch pads and compare between them.





If a specific site is chosen, the pie charts shows the percentage of successful and failed launches, and if the mouse hovers over one slice of the pie the amount that corresponds to that percentage is revealed.

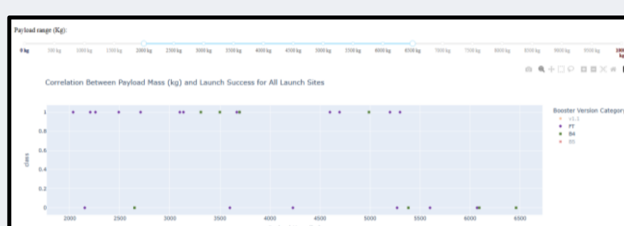




Build a Dashboard with Plotly Dash

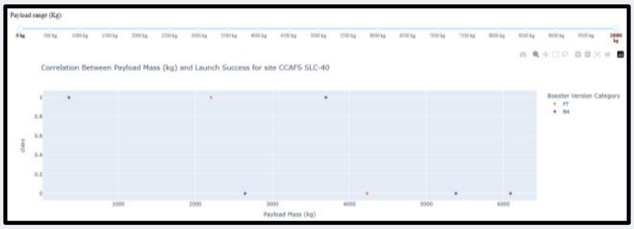
- Also, there is a **payload mass range slider for the correlation scatterplot, with steps of 500 kg to reduce or increase the range in a more precise way**. The correlation plot is affected by the dropdown list and the range slider, if all the sites are selected, it displays the correlation of all the categories of booster versions and the payload mass that they carried with the success (“1”) or failure (“0”) of the launch; if the range slider is modified, the plot reduces or increases the axis for the payload mass, making it easier to see which ranges have a better success rate.

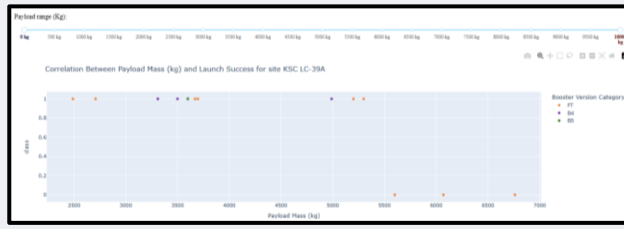
Moreover, if the mouse hovers over a point in the scatterplot it displays the info corresponding to that launch, like the precise booster version, the amount it carried and if it was a success (“1”) or a failure (“0”) in “class”; and the scatterplot can be filtered with the categories of booster version, making it clear to see the payload mass ranges that were successful for a specific booster version.

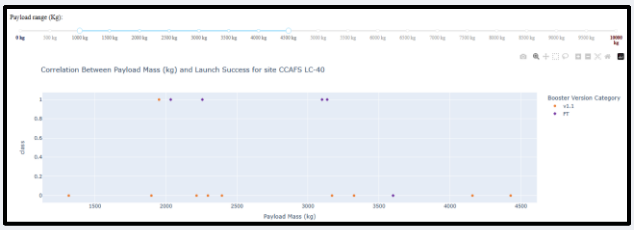


Build a Dashboard with Plotly Dash

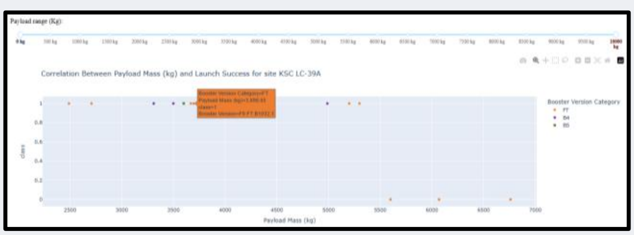
If one specific site is chosen, the scatterplot shows the correlation of all the categories of booster versions that were launch in that site and the payload mass that they carried, with the success ("1") or failure ("0") of each launch; if the range slider is modified, the plot reduces or increases the axis for the payload mass, making it easier to see which ranges have a better success rate for that precise launch site.

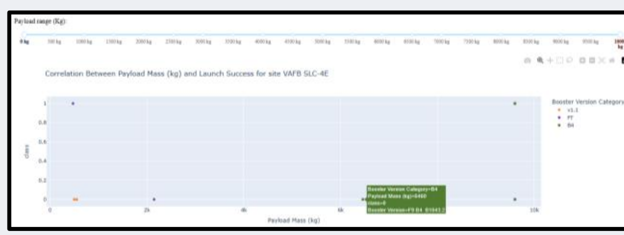


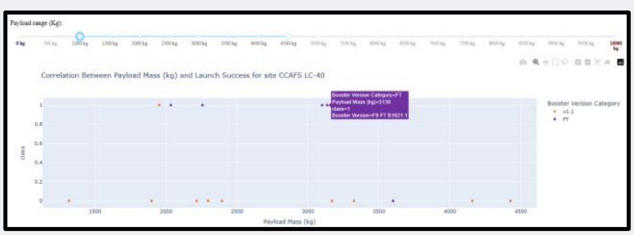




Furthermore, if the mouse hovers over a point in the scatterplot it displays the info corresponding to that launch, like the precise booster version, the amount it carried and if it was a success ("1") or a failure ("0") in "class"; and the scatterplot can be filtered with the categories of booster version, making it clear to see the payload mass ranges that were successful for a specific booster version in a launch site.







GitHub Link for the Jupyter Notebook: [Applied_Data_Science_Capstone/07-Interactive_Dashboard_with_Plotly_Dash.py at main · ChillDev93/Applied_Data_Science_Capstone · GitHub]

Predictive Analysis (Classification)

To determine if the first stage of the Falcon 9 rocket will land or not it is important to **analyze all the relevant variables that can have an influence and the outcome of every launch in the dataset so a prediction can be made for future launches**; because many variables could be related to the success or failure, the **most appropriate way to tackle this matter is by the creation of a machine learning (ML) model that can predict, with a certain level of confidence, the upcoming results of each new launch taking into account the preceding data.**

But which ML model is the most suitable for the data and the problem at hand?, to answer this question multiple ML models were tested, which include:

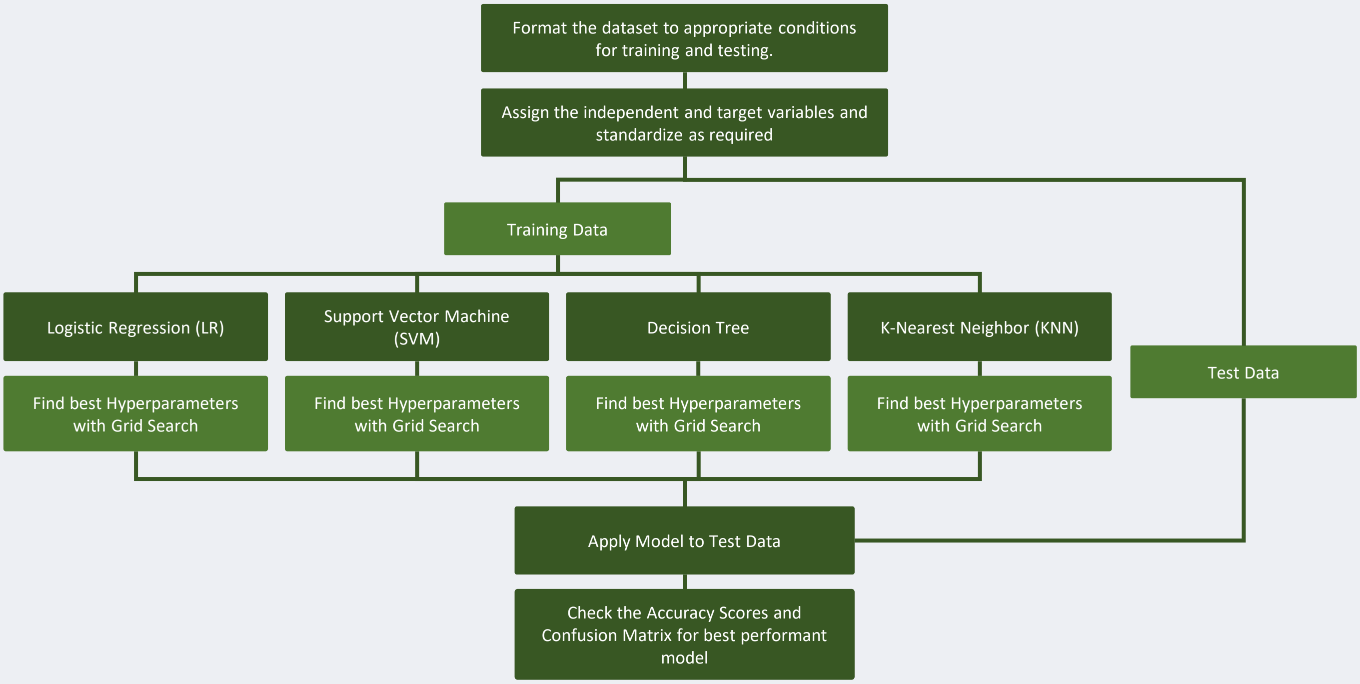
- **Logistic Regression.**
- **Support Vector Machines (SVM).**
- **Decision Tree.**
- **K-Nearest Neighbors (KNN).**

All this models went through this steps before being compared:

- Use a properly formatted dataset, which include transforming categorical variables into numerical variables (like the orbit taken), the creation of dummy variables (for true and false variables) and assigning a column for the success and failure of each launch (the "class" column).
- Standardize the variables that will be used for the prediction (the "X" variables).
- Assigning which will be the data wanted to be predicted (the "Y" variable), in this case the "class" column which refers to the outcome of the launches.
- Split the data into training and test data, to prevent overfitting the models to the dataset and get more reliable predictions.

To find which parameters for each model fitted the most to the dataset, a **Grid Search object was created to analyze and compare a list of parameters specific to each ML model and assign the best performing parameters correspondingly**, which was validated by the accuracy obtained by the tuned hyperparameters.

And last, **all the previously mentioned models were compared by their accuracy score** between predicted and actual test data, **and with the results of a confusion matrix** (to analyze false negatives and false positives) as well to confirm which one is the best performing model.



GitHub Link for the Jupyter Notebook: [Applied_Data_Science_Capstone/08-Machine_Learning_Predictive_Analysis.ipynb at main · ChillDev93/Applied_Data_Science_Capstone · GitHub]

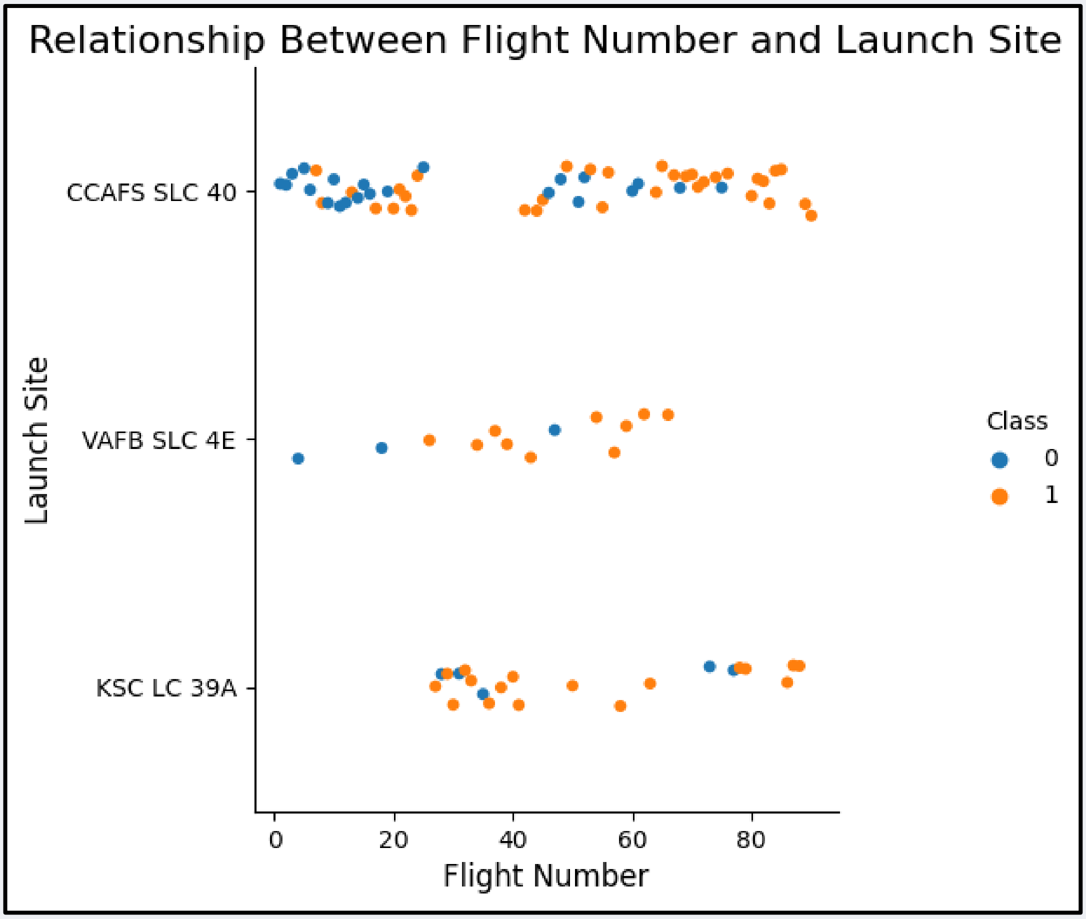
IBM Developer
SKILLS NETWORK

Section 2

Insights Drawn from Exploratory Data Analysis (EDA)

Flight Number vs. Launch Site

IBM Developer SKILLS NETWORK

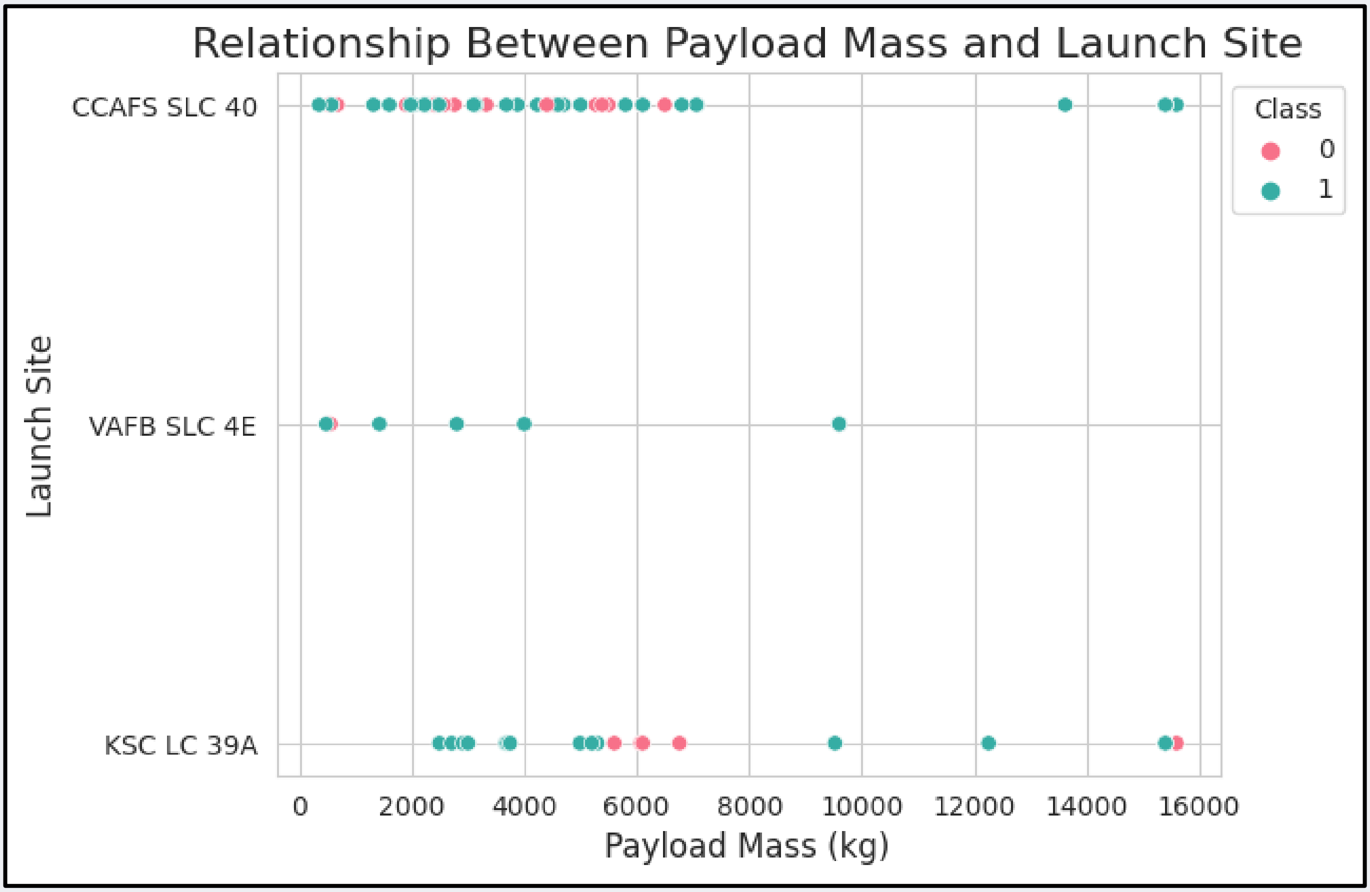


It can be seen in this relationship chart that, **from the first 90 launches, most of them were done in Cape Canaveral Space Force Station Space Launch Complex 40** (**"CCAFS SLC 40"**, because it was known as Cape Canaveral Air Force Station until December 2020) **being 61% of them**, followed by the Kennedy Space Center Launch Complex 39A (**"KSC LC 39A"**) with 25% and finally the Vandenberg Space Force Base Space Launch Complex 4E (**"VAFB SLC 4E"**, previously known as Vandenberg Air Force Base until May 2021) with 14%. **Also, the CCAFS is the one with the most quantity of failed outcomes (40%)**, while VAFB and KSC have both 23%, which makes sense as most of this results were in the first launches of the Falcon 9 rocket done by SpaceX.

Furthermore, there is a gap in the CCAFS from launches 26 to 41, when no launches were done in this launch site (from January 14 to October 30, 2017) and there have no been more launches in VAFB after the flight number 66 (June 12, 2019), after that point, 71% of the Falcon 9 launches were done in CCAFS and the rest in KSC.

In conclusion, for the first 90 flights of the Falcon 9 rockets, 22 of the 30 failures come from CCAFS, but 14 of them are from 2016 or prior, which is reasonable as **future launches prove to have better success rates, also, CCAFS is the most common place were Falcon 9 rockets are launched**.

Payload vs. Launch Site

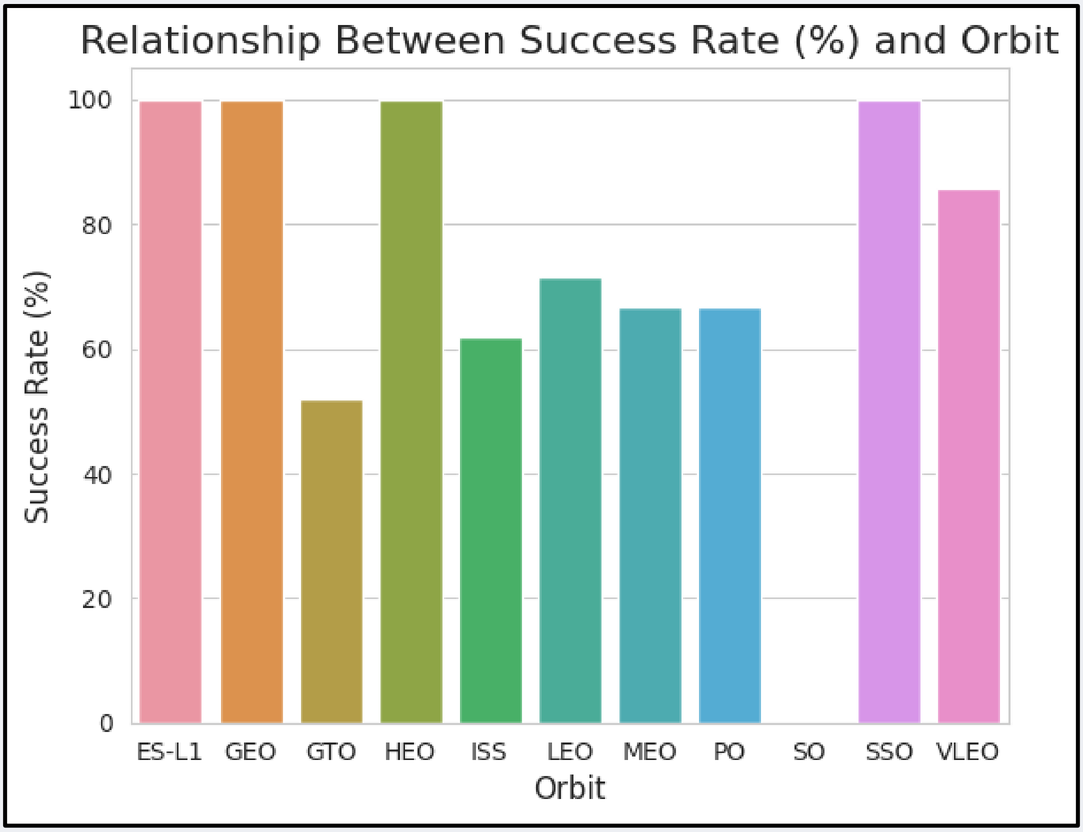


In this plot it is very evident that **most of the launches done with a Falcon 9 rocket have a payload of less than 8000 kg, being a 74% of the first 90 launches**; for this reason, it is not a surprise that most of the failed outcomes are also found in this range (40% of all the launches with a payload of less than 8000, 90% of all the unsuccessful outcomes).

But this chart **should not be used to identify in a precise manner the failed launches done with a specific payload mass**, as launches done with a same amount of mass can hide others behind them, for example, in VAFB it appears to be only one failure, when there are 3, two of them in payloads below 8000 kg and one above.

This chart is only to see what is the most common range of payload mass launched from each site, and a good insight was found previously.

Success Rate vs. Orbit Type



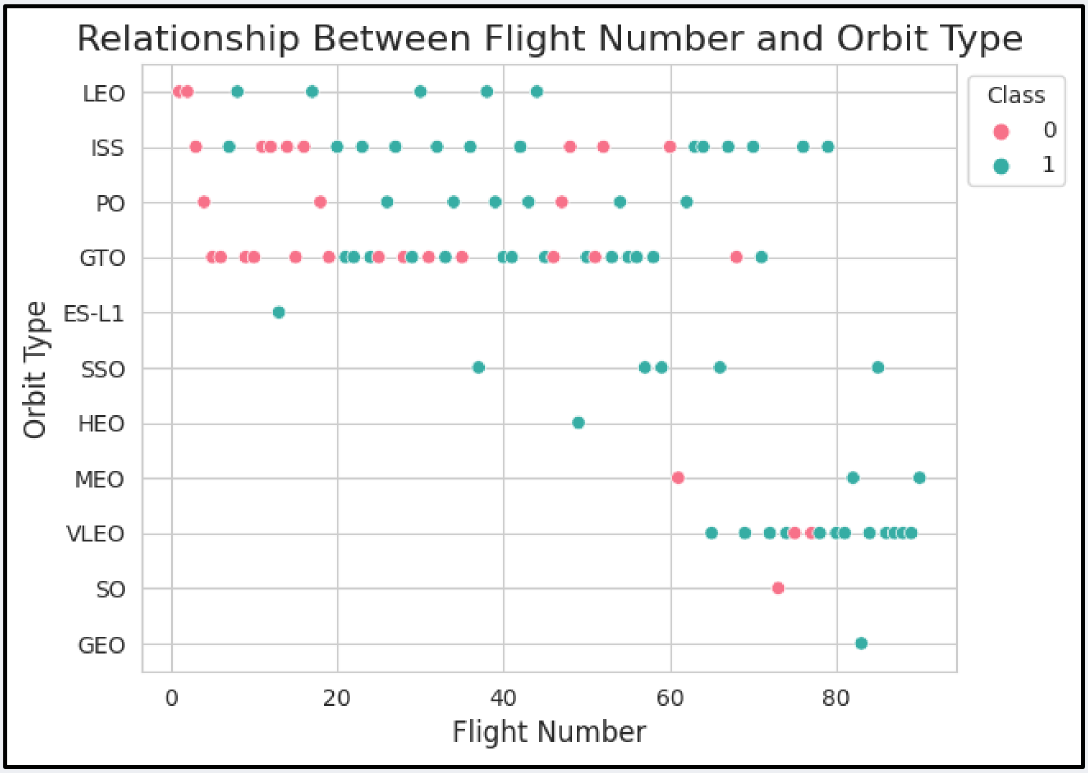
This plot shows that **the orbits that had a perfect success rate (100%) in recovering the first stage of the rocket have been the Earth-Sun Lagrange Point 1 (ES-L1), the Geosynchronous Equatorial Orbit (GEO), the High Earth Orbit (HEO) and the Sun Synchronous Orbit (SSO)**; while the Geosynchronous Transfer Orbit (GTO), the International Space Station Orbit (ISS) and both the Medium Earth Orbit (MEO) and Polar Orbit (PO) have the lowest success rates, with 52%, 62% and 67% respectively.

The Sub-Orbital (SO) is not considered as it consists in only one launch, and it was done to test a launch abort system for emergencies inflight on January 19, 2020; which of course ended in not recovering the Falcon 9 first stage as it was destroyed intentionally (success rate of 0%).

While this may suggest that ES-L1, GEO, HEO and SSO are the orbits where it is easier to recover the first stage, it can also mean that fewer launches have been performed to this locations and can skew the results, as less frequent payload launches sent into a specific orbit usually have more cautions taken into consideration that orbits regarded with lesser risk.

At the same time, the lower success rates may indicate that they have more launches: furthermore, some of those orbits could be used for testing or planned disposal of the first stage (as previously discussed with the SO orbit).

Flight Number vs. Orbit Type

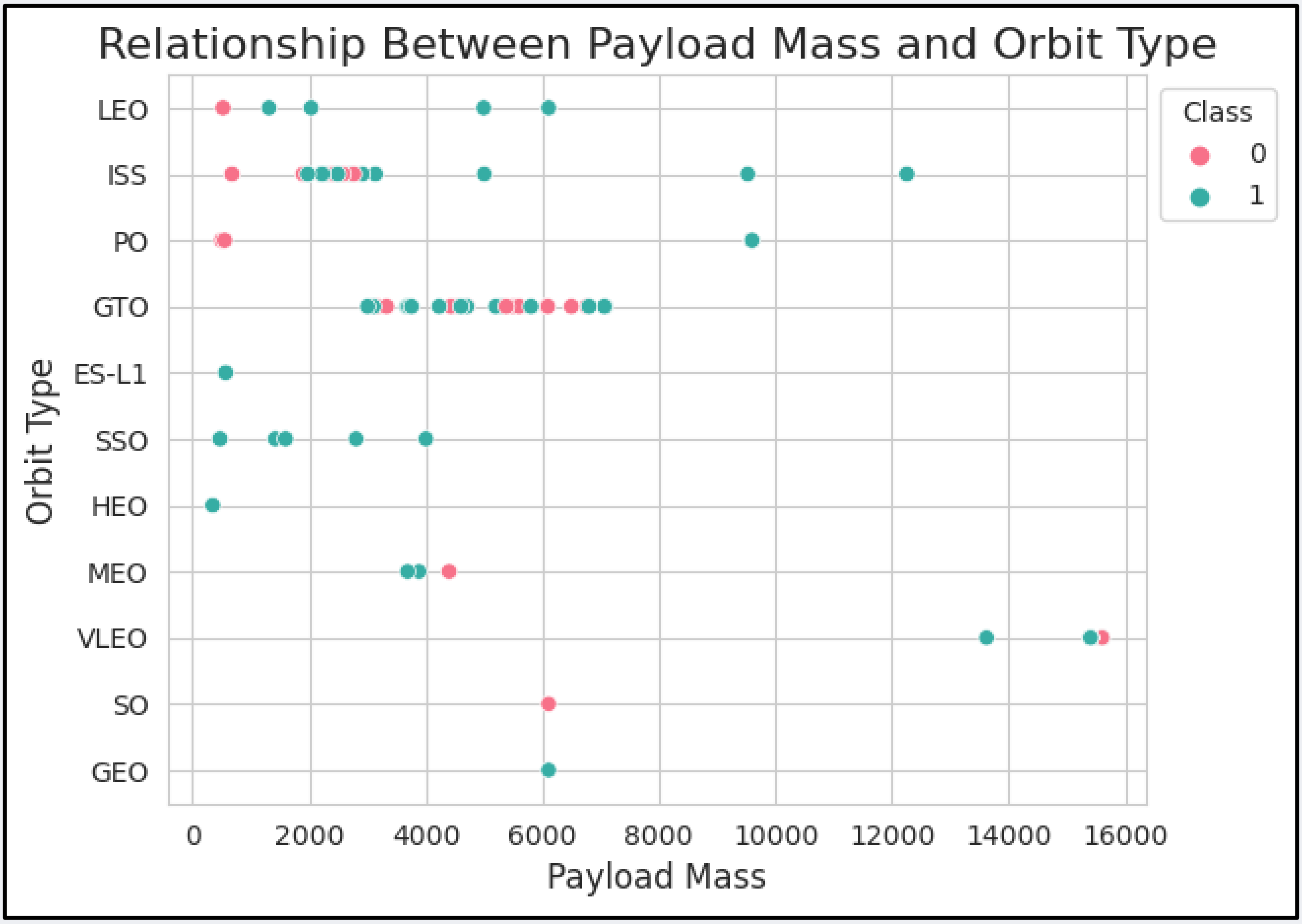


As suspected previously, **the orbits that had a perfect success rate are some of the less frequent destinations for payloads; from the first 90 launches, the ES-L1 has only one** which was for the DSCVR space weather satellite on February 11, 2015; **likewise, the HEO and the GEO have one** launch for the NASA Transiting Exoplanet Survey Satellite (TESS) in 2018 and the ANASIS-II (a South Korean Military Satellite) in 2020 **respectively**; **the only one that has multiple launches and every single one had a successful booster landing is the SSO orbit, all consisting in earth observation satellites or satellite dispenser.**

Two of the orbits with lower success rate are precisely some of the most common locations for the payloads, being GTO with a total of 27 launches (13 of which resulted in failure) and ISS with 21 (8 ended unsuccessfully), but **if attention is brought to the flight number, these also correlate with being some of the first orbits that Falcon 9 rockets were launch including LEO and PO, this can explain the high number of failed booster landings.**

The one orbit that has a lot of launches and is of special interest as it can give a general panorama of the success in recovering the first stage is the VLEO, which has a total of 14 launches and a success rate of 86%, the two failed landings had Starlink satellites as payload mass.

Payload vs. Orbit Type

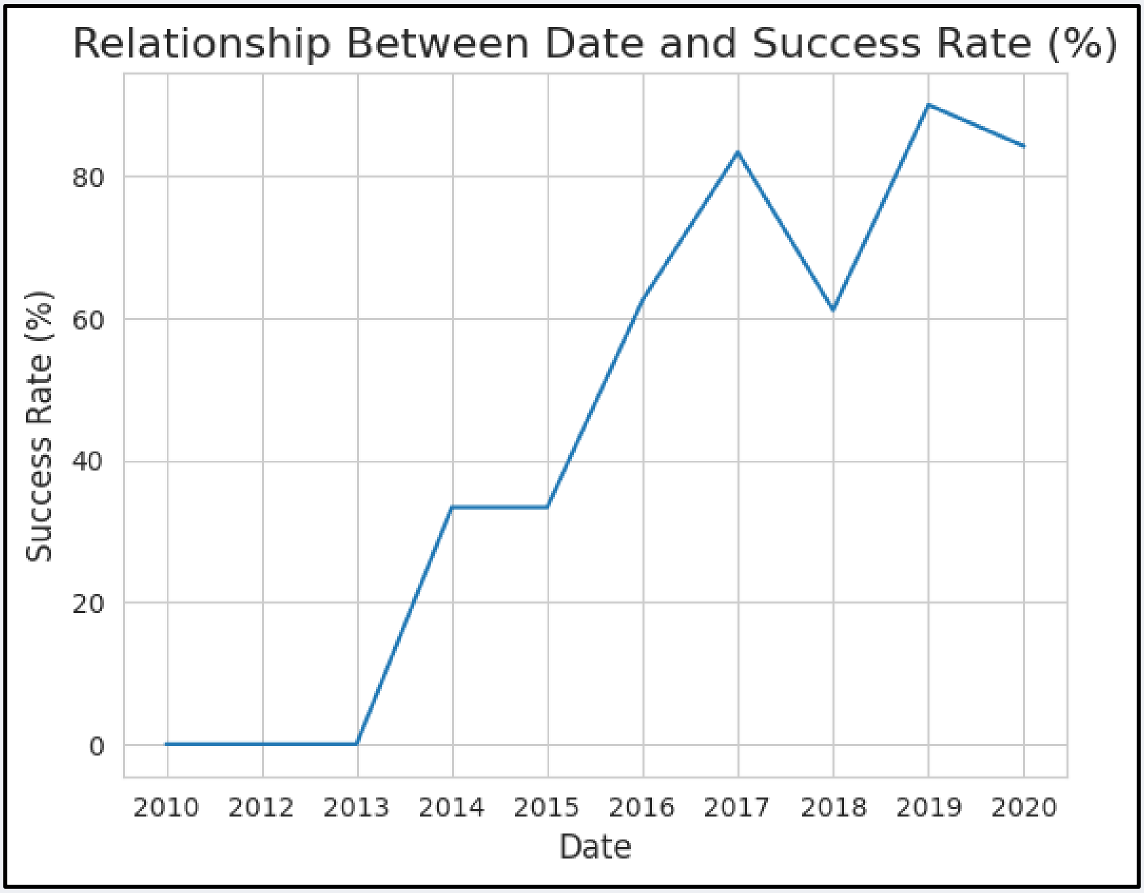


As seen in the Payload vs Launch Site, **the vast majority of launches have a payload mass below 8000 kg and they go to most of the orbits, only being the VLEO orbit the exception**.

Excluding one launch orbits, the orbits that show a bigger variation in payload mass are the GTO, ISS, MEO and SSO; basically, every launch in GTO orbit has a payload mass between 3000 and 7100 kg and the ISS orbit usually receives payloads of 1800 to 3200 kg, while every launch in MEO had a different mass from 3600 to 4400 kg, and SSO from 400 to 4000 kg. The other orbits usually repeat same mass payload launches, for example, PO is either around 500 kg or exactly 9600, and VLEO is around 13600 to 15600 kg.

The only three orbits that had payload masses above 8000 kg are the ISS, the PO and the VLEO orbits, and these ones have a very high success rate at higher amount of mass, 100% of the launches done at ISS orbit over 8000 kg of mass have been successful, and 86% for both PO and VLEO, the former with one booster landing that was not attempted (9600 kg) and the latter with two failed outcomes (15,400 and 15,600 kg)

Launch Success Yearly Trend



The success rate at recovering the first stage of the Falcon 9 rockets has been increasing from 2010 to 2020.

During the first three years (2010-2013) all launches either did not try to land the first stage or failed trying to do so (0%), in 2014 and 2015 the success rate increased to 33%, only one booster landed in conditions to be reused, but it was not as it is in display outside SpaceX headquarters.

In 2016 the rate increased to 62.50%, all of these successful lands were later reused, **in 2017 the success rate went up to 83.33%**, with only one booster not being reusable.

2018 was the first year with a dip in the success rate, with 61.11%. One failed landing and six no attempts increased the failure percentage, and from the ones that landed, only one cannot be reused. **2019 saw its highest increase, to 90%**, with just one booster that was not attempted its recovery. **Until November 5, 2020, the percentage was at 84.21%**, with two failed landings and one not attempted.

All Launch Site Names

```
In [8]: %%sql

        SELECT DISTINCT "Launch_Site" FROM SPACEXTBL;

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

Out[8]:

Launch_Site
CCAFS LC-40
VAFB SLC-4E
KSC LC-39A
CCAFS SLC-40
None

All the unique launch sites where the Falcon 9 rockets had take-off are 3:

- **Cape Canaveral Space Force Station, Space Launch Complex 40** : which includes two acronyms, **“CCAFS SLC-40” and “CCAFS LC-40”**, this is because this launch pad had its named changed twice, from “Air Force Station” to “Space Force Station” and “Launch Complex” to “Space Launch Complex”, both refers exactly to the same launch pad, but to preserve the original registry of the data, both will be considered.
- **Vandenberg Space Force Base, Space Launch Complex 4E** : represented in the data base by its acronym **“VAFB SLC-4E”**, which was formed from its previous name being “Air Force Base”.
- **Kennedy Space Center, Launch Complex 39A** : known by its acronym **“KSC LC-39A”**.

The “None” refers to empty rows in the Database, as we are only considering Falcon 9 Launches up to December 6, 2020, for consistency with other projects. This can be ignored as it has no effect in future results.

Launch Site Names Begin with 'CCA'

IBM Developer SKILLS NETWORK

Date	Time (UTC)	Booster Version	Launch Site	Payload	PAYLOAD _MASS__ KG_	Orbit	Customer	Mission Outcome	Landing Outcome
06/04/2010	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0.0	LEO	SpaceX	Success	Failure (parachute)
12/08/2010	15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight C1, two CubeSats, barrel of Brouere cheese	0.0	LEO (ISS)	NASA (COTS) NRO	Success	Failure (parachute)
22/05/2012	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525.0	LEO (ISS)	NASA (COTS)	Success	No attempt
10/08/2012	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500.0	LEO (ISS)	NASA (CRS)	Success	No attempt
03/01/2013	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677.0	LEO (ISS)	NASA (CRS)	Success	No attempt

Just for testing purposes, a query requesting 5 launches with launch sites that begin with "CCA" was performed. The purpose of the test is to verify the integrity of the names in the launch site (they don't include non-conventional or blank characters) and see that all the columns display the intended rows.

Taking advantage of this check, all variables can be seen in a comprehensible table, which includes the Date of the launch (in "DD/MM/AAAA" format), the time it was performed (24-hour format), the booster version used, the launch site, the payload that the rocket included, the payload mass registered (in kg), the destination orbit, the customer, the mission outcome (if the payload arrived to its target point or not) and the landing outcome (if there was an attempt to recover the first stage of the rocket, the place or the mechanism used to land and if it was successful or not).

Total Payload Mass


```
In [10]: %%sql

         SELECT SUM(PAYLOAD_MASS__KG_) AS 'TOTAL_NASA_PAYLOAD_MASS' FROM SPACEXTBL
         WHERE Customer='NASA (CRS)';

 * sqlite:///my_data1.db
Done.

Out[10]: TOTAL_NASA_PAYLOAD_MASS
         45596.0
```

The NASA has executed a series of programs with the Falcon 9 rockets from 2010 to 2020, which includes:

- **NASA (CCD)** : Development of the Commercial Crew Program.
- **NASA (CCDev)**: Commercial Crew Development.
- **NASA (CCP)**: Commercial Crew Program.
- **NASA (COTS)**: Commercial Orbital Transportation Services.
- **NASA (COTS) NRO**: Commercial Orbital Transportation Services / National Reconnaissance Office.
- **NASA (CRS)**: Commercial Resupply Services.
- **NASA (CRS)**, **Kacific 1**: Commercial Resupply Services / Kacific Broadband Satellites Group.
- **NASA (CTS)**: Crew Transportation Service, part of the Commercial Crew Program (CCP).
- **NASA (LSP)**: Launch Services Program.
- **NASA (LSP) NOAA CNES**: Launch Services Program / National Oceanic and Atmospheric Administration / National Centre for Space Studies.
- **NASA / NOAA / ESA / EUMETSAT**: National Aeronautics and Space Administration / National Oceanic and Atmospheric Administration / European Space Agency / European Organization for the Exploitation of Meteorological Satellites.

If there is a special interest in one of these programs, it is needed to specify the information from which is going to be collected.

In this case, its important to known the total payload mass for the NASA (CRS) launches, as these ones correspond to the delivery of cargo and supplies to the International Space Station (ISS), which consist of payloads of different masses in basically each launch. And **the result of the query is that, between 2012 (the first launch under this program) and June 2020, 45,596kg has been sent to the ISS using Falcon 9 rockets.**

Average Payload Mass by F9 v1.1


```
In [11]: %%sql

         SELECT AVG(PAYLOAD_MASS__KG_) AS 'AVG_PAYLOAD_MASS_F9_V1.1' FROM SPACEXTBL
         WHERE Booster_Version = 'F9 v1.1';

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

Out[11]:

AVG_PAYLOAD_MASS_F9_V1.1
2928.4

There have been multiple versions of the booster used in the Falcon 9 Rockets, which include:

- **F9 v1.0**: The original version of the Falcon 9, it made 5 flights from 2010 to 2013, powered with nine SpaceX Merlin 1C rocket engines that could carry **payload masses to LEO of 9,000 kg and to GTO of 3,400 kg**.
- **F9 v1.1**: The second version of the Falcon 9, it made 15 total launches from 2013 to 2016, where 14 were successful and 1 failed; the modifications consisted in 60% more thrust thanks to its nine Merlin 1D engines and 60% longer fuel tanks, which increased its weight 60% from the previous model, this means it could **take payload masses to LEO from 10,454 kg to 13,150 kg, and to GTO of 4,850 kg.**
- **F9 FT**: also known as Falcon 9 Full Thrust or Falcon 9 v1.2, the third version of the rocket started launch operations in 2015 and continues its use up to the last point in this dataset (December 6, 2020), this rocket can **carry up to 22,800 kg to LEO and 8,300 kg to GTO**, and is represented in the dataset in the following manner:
    - **F9 FT**: this name includes all the block variants from 1 to 3, it included a set of modifications from the v.1.1 that increased the height to 70 m and the performance to 33% compared to the last model.
    - **F9 B4**: this acronym corresponds to Full Thrust Block 4, being a minor improvement from Block 3 including incremental engine thrust upgrades in first and second stage, effectively being a transitional version.
    - **F9 B5**: Full Thrust Block 5, or also known as Falcon 9 Block 5, it is comprised with a noticeable number of upgrades, especially higher thrust on all engines and improvements on the landing legs.

These names can be also in conjunction with the construction serial number (for example, B1019) and an optional flight number when the same booster serial number has been reused (e.g., B1029.1, B1029.2).

Taking into account this information, to find out the average payload mass that carried the Falcon 9 v1.1 in its 15 launches, a query must specify exactly this booster version, and as a result it returns the average as 2,928.4 kg, which is consistent with the information in the Database and means that every launch used the GTO reference as a limit in the payload mass, even if a mission required a shorter orbit.

First Successful Ground Landing Date

```
In [14]: %%sql

SELECT MIN(substr(Date,7,4) || substr(Date,4,2) || substr(Date,1,2)) AS 'FIRST_SUCCESSFUL_LANDING' FROM SPACEXTBL
WHERE Landing_Outcome = 'Success (ground pad)';

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

Out[14]:

FIRST_SUCCESSFUL_LANDING
20151222

All the landing outcomes from the in the database are registered in the following manner:

- **Controlled (Ocean):** Referring to soft landings on the ocean, the first stage was recovered but not in a reusable state.
- **Uncontrolled (Ocean):** When a soft landing at sea fails by any reason, it is considered as uncontrolled, as the first stage crashed into the ocean.
- **Failure (Drone Ship):** as it states, when the landing was intended on a drone ship at sea, but it failed in the end.
- **Failure (Parachute):** When the attempt to land the first stage was done by using parachutes to reduce the speed and make soft contact with the ground, nonetheless it failed to do so.
- **No Attempt:** In this circumstances, SpaceX did not take an effort to recover the first stage, this is by design as certain payloads do not allow to try a landing of the first stage without risk.
- **Precluded (Drone Ship)**: When originally a landing onto a drone ship is planned but is canceled by unforeseen conditions in a launch.
- **Success (Drone Ship):** This signifies that the first stage of a Falcon 9 rocket successfully landed onto a drone ship at sea and in conditions to be used again in future launches.
- **Success (Ground Pad):** This means that the first stage achieved its landing on ground and fulfilling all requirements to be reused in the future.

So, to retrieve with a query the first successful landing, it is important to specify the method in which the first stage landed, by using the "Success (Ground Pad)" outcome **the date obtained was December 22, 2015, which is in fact the first successful landing ever with a Falcon 9 First Stage Booster, this was achieved with the F9 FT booster version sent to a Low Earth Orbit (LEO) with 2,034 kg of payload mass**.

The previous models (F9 v1.0 and v1.1) did not achieve a successful first stage landing in conditions that could allow it to be reused (no attempts, failed outcomes, uncontrolled landings in the ocean or controlled landings at sea at most), and after the last attempt to land a v1.1 model on January 17, 2016 (which ended in failure), all future models were Falcon 9 Full Thrust up to 2020 (the last year recorded in this dataset).

Successful Drone Ship Landing with Payload Between 4000 and 6000

```
In [15]: %%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.
```

Out[15]:

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

The booster versions that fulfill the required conditions (successfully landing on a drone ship and having carried a payload mass between 4,000 and 6,000 kg) are:

- **F9 FT B1022:** performed on May 6, 2016. It carried a payload mass of 4,696 kg sent to GTO (this being the first successful recovery with this destination) and was the first and only launch of the booster with serial number 1022, even if it landed successfully and in conditions at the time, later it was determined to have suffered a lot of damage by the speed that it re-entered Earth atmosphere and the heat that it implies, deeming it unusable for future launches.
- **F9 FT B1026:** done on August 14, 2016. It delivered 4,600 kg of payload to GTO and was the only launch of serial number 1026, as it suffered structural stress by burning only one engine for landing, even though it successfully landed.
- **F9 FT B1021.2:** launched on March 30, 2017, and sending 5,300 kg to GTO, was the first time a first stage booster was reused to launch a payload and the first time the first stage landed successfully and intact twice as well.
- **F9 FT B1031.2:** sent also to GTO on October 11, 2017, with a payload of 5,200 kg. This was the third time a first stage from a Falcon 9 was reused for a launch.

There have been other successful landings on drone ships, but they don't comply with the payload mass range, but if it is expanded a little bit, it can be seen that GTO orbits that have payloads from 3,000 to 6,000 kg have landed successfully on drone ships; if the orbit is closer (like LEO), it can carry up to 9,600 kg and at further orbits (HEO) it is limited to 500 kg or less.

Total Number of Successful and Failure Mission Outcomes

```
In [16]: %%sql

SELECT Mission_Outcome, COUNT(*) FROM SPACEXTBL
GROUP BY Mission_Outcome;

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

Out[16]:

Mission_Outcome	COUNT(*)
None	898
Failure (in flight)	1
Success	98
Success	1
Success (payload status unclear)	1

Excluding the row with “None” (caused by empty rows in the database) and merging both “Success” rows (this could be caused by a blank character in one of the registries, putting it into another group), there are three outcomes for the launch missions:

- **Success:** The payload has been delivered successfully to the intended orbit; this has been achieved 99 times.
- **Success (Payload Status Unclear):** This refers to a particular case that happened on January 8, 2018, where there is a controversy in the launch of the Zuma US government classified satellite, in which is uncertain if the payload actually separated from the second stage, or it was lost. As it is a classified mission, no concrete information has been provided, but SpaceX mentions that the Falcon 9 rocket performed correctly, so it will be taken as a Successful Launch.
- **Failure (in flight):** As it states, it refers to a situation where the mission failed either by an accident, unforeseen circumstances or a design flaw, and the payload was lost when the rocket was in flight.

For the 101 launches performed from June 4, 2010, to December 6, 2020, SpaceX reports 100 successful launches and only one failure, the latter happened on June 28, 2015, when an overpressure incident in the second stage provoked a vehicle breakup; the payload survived the explosion, but it was destroyed when it impacted the ocean

Boosters Carried Maximum Payload

```
In [17]: %%sql

SELECT Booster_Version, PAYLOAD_MASS__KG_ FROM SPACEXTBL
WHERE PAYLOAD_MASS__KG_=(SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTBL);

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

Out[17]:

Booster_Version	PAYLOAD_MASS__KG_
F9 B5 B1048.4	15600.0
F9 B5 B1049.4	15600.0
F9 B5 B1051.3	15600.0
F9 B5 B1056.4	15600.0
F9 B5 B1048.5	15600.0
F9 B5 B1051.4	15600.0
F9 B5 B1049.5	15600.0
F9 B5 B1060.2	15600.0
F9 B5 B1058.3	15600.0
F9 B5 B1051.6	15600.0
F9 B5 B1060.3	15600.0
F9 B5 B1049.7	15600.0

Looking at the boosters that have carried the highest payload mass can be concluded **that all of them correspond to the Falcon 9 Full Thrust Block 5 version**, which relate to previous information as this model have the improvements required to sent that amount of mass comfortably (higher thrust from the engines and better landing legs) and earlier versions could not carry this load, even if it was intended to not recover the first stage **(v1.0 had a limit of 9,000 kg to LEO, and v1.1 was 13,150 kg to the same orbit)**.

Even though these launches were performed with the Block 5 model, **they are limited to 68% of the maximum payload mass permitted (stated in its site as 22,800 kg), which indicates the weight that SpaceX can assure the recovery of the first stage**. All these launches were performed after 2019 and sent to LEO, and **the payload of them are Starlink Satellites**, part of the Satellite Internet Constellation that SpaceX wants to form to provide satellite internet access and global mobile phone service. This confirms that **usually these closer to maximum mass limit launches are usually done with payloads that belong to SpaceX**.

And finally, the fact that **all of these launches were performed with reused first stage boosters** (from one use up to six previous launches) indicates the confidence that SpaceX have in their performance, even though two of these twelve launches resulted in a failed landing.

In [18]:
```
%%sql

SELECT SUBSTR(Date, 4, 2) AS 'MONTH', Landing_Outcome, Booster_Version, Launch_Site FROM SPACEXTBL
WHERE substr(Date,7,4)='2015' AND Landing_Outcome='Failure (drone ship)';
```

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

Out[18]:

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

If the data of the same year that the first successful landing was achieved with a Full Thrust is analyzed, **it can be seen that this year was basically a rollercoaster, as 2015 reported 5 different landing outcomes for just 7 total launches**, which includes controlled ocean landing (1 launch), precluded drone ship landing (1), no attempts at recovering the first stage (2), and finally, besides the successful ground pad landing (1), the failed drone ship landing (2).

Moreover, **all the other landing outcomes where it was not possible to reuse the first stage the Falcon 9 v1.1 was involved, which confirms that the improvements performed in the FT model made it viable for successful landings in conditions to use the first stage in future launches**.

The **two more interesting landings are the failures at arriving at the drone ship, as these involved mechanical blunders that made these boosters crash at the last minute**, these occurred in the launches made in January and April (sadly the query reported January erroneously as October, caused by a Date Format error that will be explained in the rank landing outcomes) and having the same launch site as the successful one, so it makes it clear that this model have specifications that complicated its goal of landing and reusing the first stage.

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

There is a mistake in the Database, where the first launch date appears as "06/04/2010", when in reality it was performed on "04/06/2010"; for this reason, the range of dates will be from "06/04/2010" to "20/03/2017" to include this launch.

In [20]:
```
%%sql

SELECT Landing_Outcome, COUNT(Landing_Outcome) FROM SPACEXTBL
WHERE (substr(Date,7,4) || substr(Date,4,2) || substr(Date,1,2)) BETWEEN '20100406' AND '20170320'
GROUP BY Landing_Outcome ORDER BY COUNT(Landing_Outcome) DESC;
```

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

Out[20]:

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

The total amount of launches performed between June 4, 2010, and March 20, 2017, are 31 (there would be 32 if it wasn't for the pre-flight test explosion on September 3, 2016, which SpaceX doesn't count in their launch totals), in the query appear as 33 because a slight misconfiguration is present in the database, where multiple date formats are mixed (it was found that some are in "DD/MM/AAAA" and others in "MM/DD/AAAA" format), this is problematic as Months and Days between 01 and 12 are easily confused, reminding the importance of verifying the date format in Datasets.

Due to time constraints and because this is the last query performed with this dataset, this erroneous query result will be fixed manually, and the results shown below, but the original outcome is displayed above; the right way of fixing this would be by changing the date format from the date column in the Database. The correct ranked landing outcomes are:

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

In the last table the difficulty of recovering the first stage of a rocket in conditions to be reused are revealed, **in this period of almost 7 years, from the 30 successful rocket launches (the one with the "precluded" detail was in fact a failed launched, as the vehicle broke by overpressure in its second stage and the payload was lost) only 8 could achieve to recover the first stage in a reusable state, either by pulling of the landing on a ground pad (3 of them) or a drone ship (5)**.

From the other 22, **three landings recovered the first stage but not in conditions to be used again (the controlled ocean landings)**. The last 19 could be said that they were failed outcomes as the first stage did not properly land.

10 of the 19 failed outcomes SpaceX did not try to recover the first stage (no attempt), almost all of them are pre-2016, when the F9 v.1.0 and the v.1.1 were used, with just one launch not making an effort to recover it after the first successful landing of a first stage (March 16, 2017).

Five of those 19 were failed drone ship landings (The first stage crashed into the drone ship and exploded in this cases, technically landing in multiple pieces), **these failed attempts appear from January 10, 2015, until June 15, 2016**.

And finally, **there is two landings outcomes that happened twice, the uncontrolled ocean ones that refer to literally crashing into the ocean, when a soft landing was intended; and the failed landings using parachutes, in the case of these ones basically the first stage burned upon reentry into atmosphere, so they did not even have a chance to deploy the parachutes**.

Something to clarify with **the failed landings with parachutes is that these are the first two launches ever done with the Falcon 9**, so using this configuration in future launches is very unlikely when the use of grid fins, the booster engines and the landing legs have proven to be the way to go.

Section 3

Launch Sites Proximities Analysis

Launch Sites of Falcon 9 Rockets

For the analysis of the launch site locations the first 56 launches have been chosen, as the launch sites do not change in future launches and to speed up the generation of the interactive maps, since the main purpose is to know if the location data can be displayed well in an interactive map, for more precise examination of successful and failed launches the most recent data would be required, but this test will confirm if the results shown make it worth the extra time spend in that endeavor.

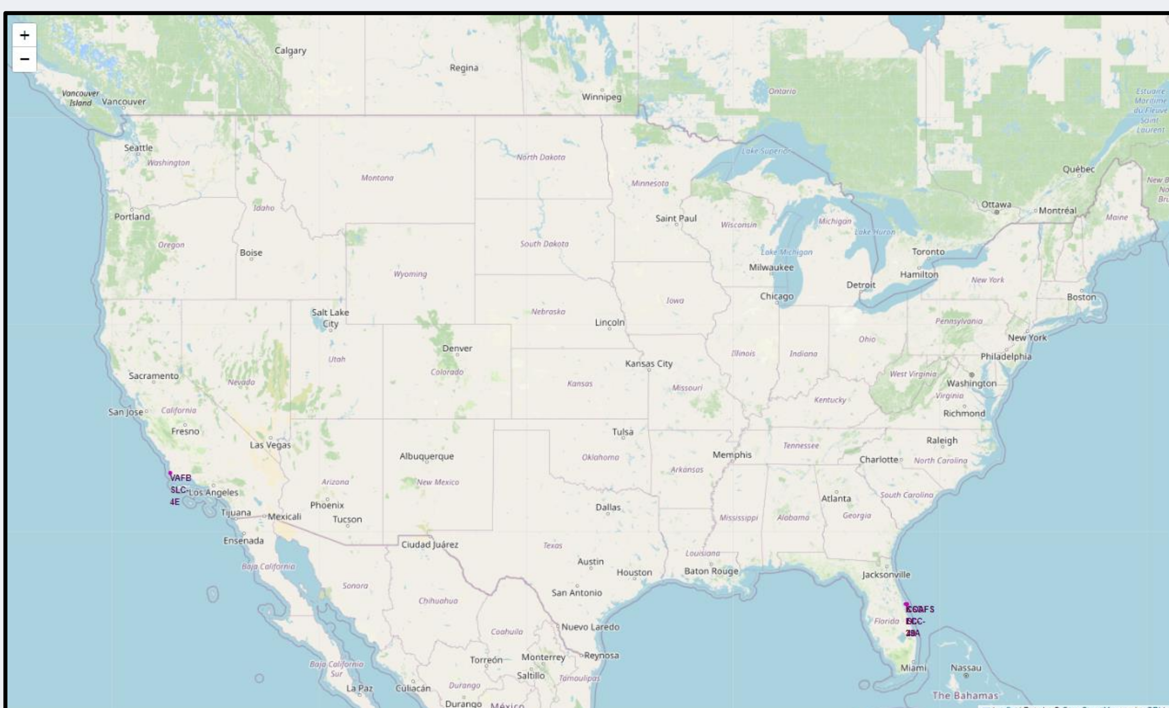
All the places that have been used for Falcon 9 rocket launches between 2010 and June 4, 2018, are three space launch bases located in two states of the United States: California and Florida.

The ones located in Florida are two:

- **Kennedy Space Center, Launch Complex 39A ("KSC LC 39A")**.
- **Cape Canaveral Space Force Station, Space Launch Complex 40** : which includes two acronyms, **"CCAFS SLC-40" and "CCAFS LC-40"**, both refers exactly to the same launch pad, but to preserve the original registry of the data, both will be considered.

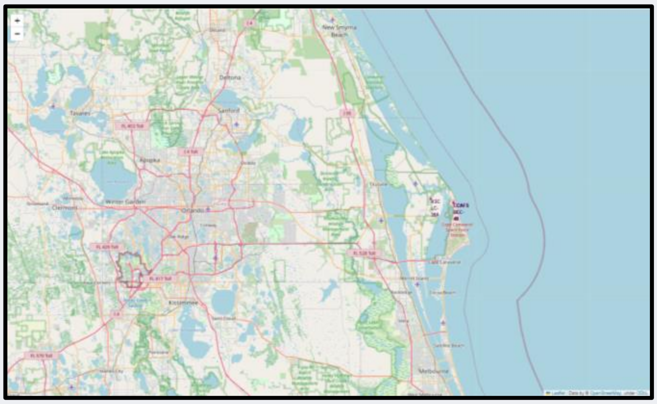
And the only one in California is:

- **Vandenberg Space Force Base, Space Launch Complex 4E ("VAFB SLC 4E").**

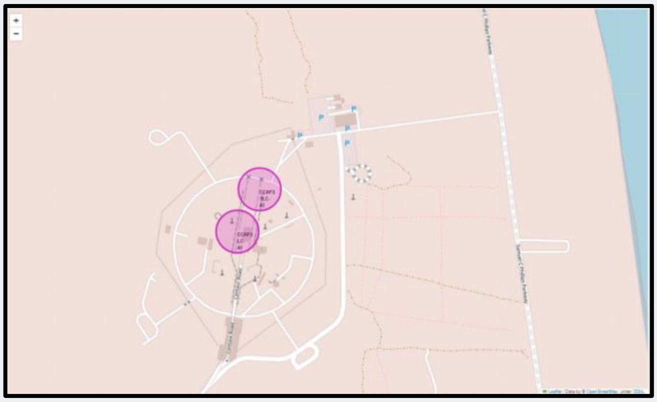


Launch Sites of Falcon 9 Rockets

The Kennedy Space Center and the Cape Canaveral Space Force Station are located close to Orlando, and being more specific, the KSC is located on Merritt Island, in Brevard County; and the CCSFS is located on Cape Canaveral, also in Brevard County.

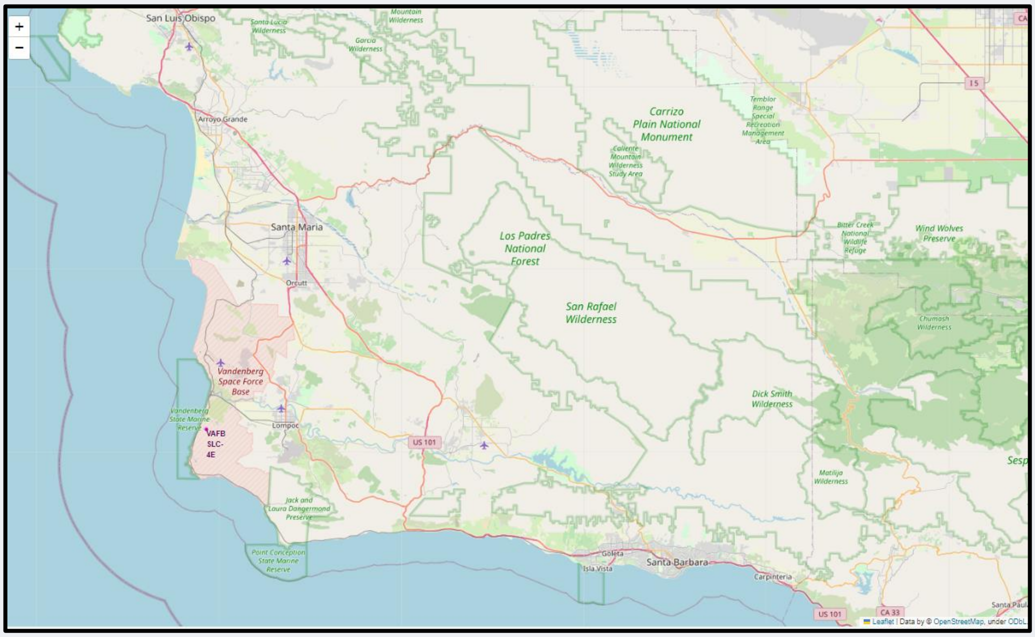


To see the specific location of the launch bases, as they are not necessarily close to the main complex, purple markers with the specific names were used, as shown next:

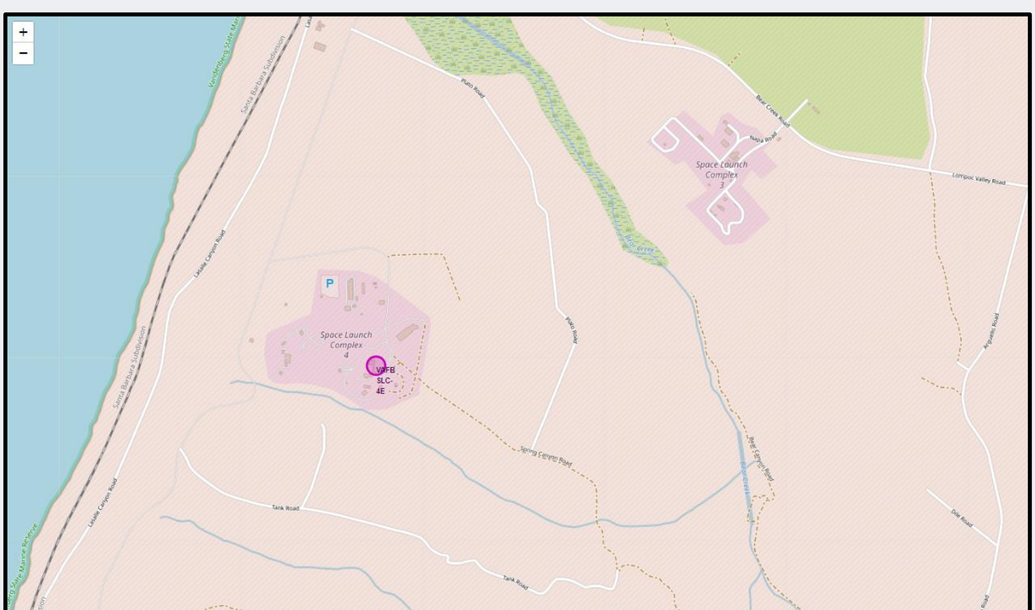


Even though the markers for “CCAFS SLC-40” and “CCAFS LC-40” appear as they are separated on the map, both share the same coordinates.

And for the Vandenberg Space Force Base, it is close to three cities: Arroyo Grande, Santa Maria and Santa Barbara; its precise location is in Santa Barbara County.

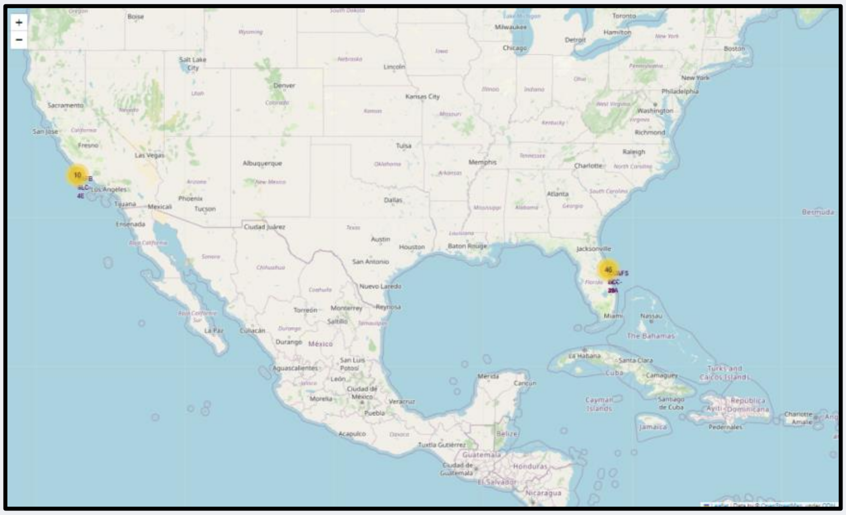


To see the specific location of the launch base, as it is not necessarily close to the main complex, a purple marker with the specific name is used, as shown next:

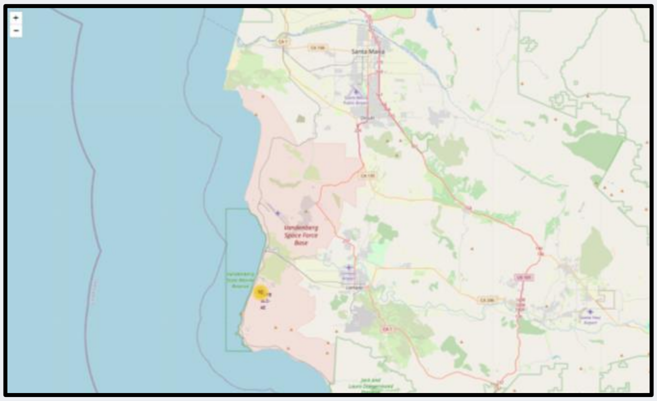


Launch Outcomes for Each Launch Site

To localize all the 56 launches in the map without anyone being on top of each other, a marker cluster was created for each launch site, in the nationwide map, it can be seen that 46 were performed in Florida and 10 in California.

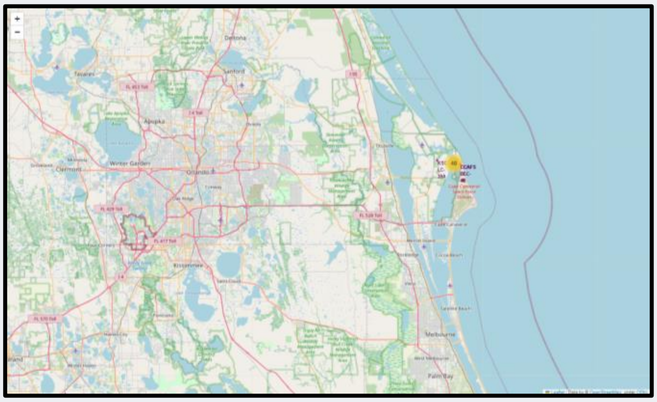


A closer look in each of the states:



The VSFB with their 10 launches in Santa Barbara County, California.

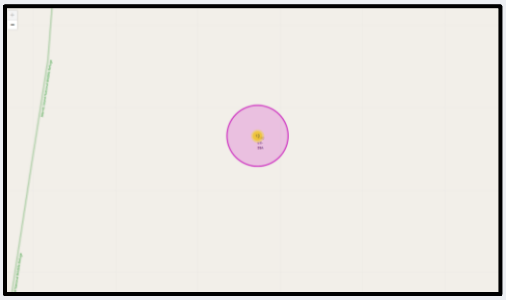
The CCSFS and the KSC in Brevard County, Florida. Both sum the 46 launches made in this state.

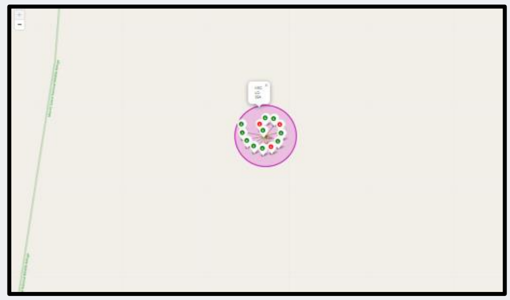


Launch Outcomes for Each Launch Site

Looking at the marker cluster **in KSC, it includes 13 launches, from February 19, 2017, to May 11, 2018, where all of them used the F9 FT booster** (including Block 4 and 5) with payload masses between 2,400 and 6,800 kg sent to LEO, ISS and GTO orbits.

From those launches, **10 were successful and 3 failed in the landing of the first stage**, the successful ones (ground pad and drone ship) are indicated with green markers and the failed ones (no attempt to recover the first stage) with red pointers.

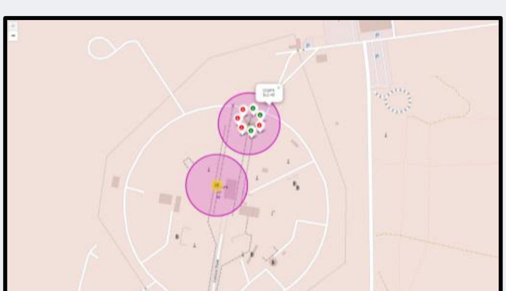


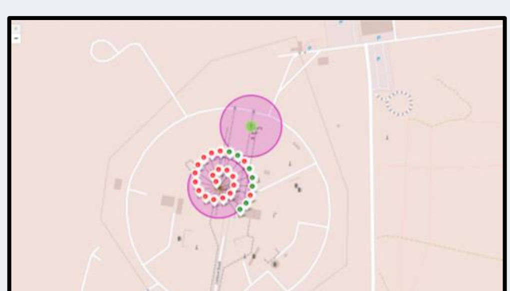


And now, with the **CCSFS cluster markers, 26 launches were made under the name "CCAFS LC-40", from June 4, 2010, to August 14, 2016, and 7 launches under the name "CCAFS SLC-40" from December 15, 2017, to June 4, 2018**. The year that this launch base was not used (and when it changed its named) was because of repairs being done to it after the explosion of September 3, 2016.

These launches used the F9 v1.0 (5 launches) with payloads between 0 and 700 kg and sent to LEO and ISS orbits; **F9 v1.1 (13)** with payloads from 500 to 4,800 kg and having LEO, ISS, GTO and ES-L1 orbits as destinations; **FT (10)** and **FT B4 (5)** with masses between 300 and 6,100 kg and launched to LEO, ISS, GTO and HEO orbits.

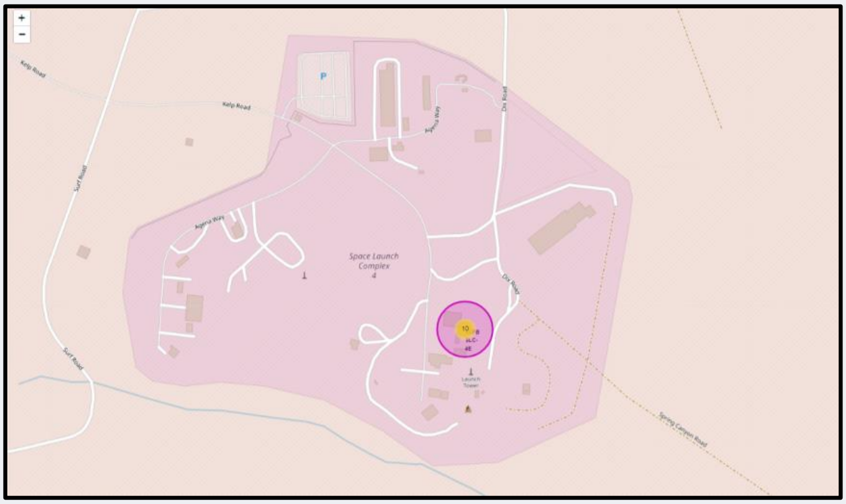
From the 26 launches under "CCAFS LC-40" name, 7 were successful landings of the first stage and 16 failed; and from the 7 launches with the name "CCAFS SLC-40", 3 recovered the first stage and 4 did not. The successful landings are marked with green markers and the failed ones with red pointers.





Launch Outcomes for Each Launch Site

And finally, **with VSFB cluster marker, 10 launches were done in this site, from September 29, 2013, to May 22, 2018, where 2 launches were done with the F9 v1.1 booster**, with payloads from 500 to 600 kg sent to LEO and Polar LEO (PO) orbits; **5 launches with the F9 FT**, with masses between 400 and 9,600 kg and launched to LEO, Polar LEO (PO) and SSO orbits; **and ultimately 3 launches with the Block 4**, with payloads ranging from 6,400 to 9,600 kg and all having Polar LEO orbits as destination.

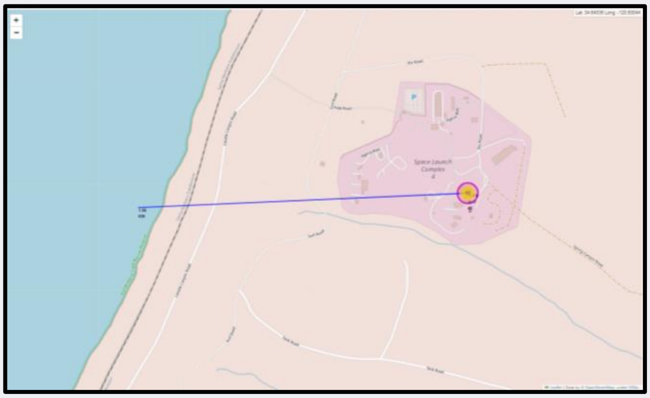


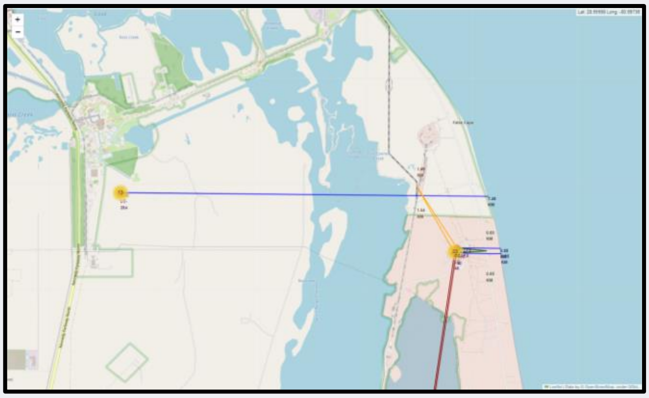
From these 10 launches, 4 landings of the first stage were successful (all of them in drone ships) and the other 6 were considered as a failure, either because it cannot be reused, or it wasn't intended to be recovered after the launch; but one of this failed landings is interesting as it did achieve a soft landing on the drone ship, but one of the legs failed and exploded after it fell over (the launch of January 17, 2016).

All the successful launches are indicated with a green marker on the map, while the unsuccessful ones are shown with a red pointer.

Distance from Launch Site to its Proximities

Looking at the distance from launch sites to their proximities, **it can be concluded that they usually are located next to the ocean; the VSFB at 1.56 km from the west coast, and both the KSC and the CCSFS at 7.48 km and 0.9 km from the east coast, respectively.** As rocket launches aim to follow an east trajectory to take advantage from the Earth Rotation to get away from the atmosphere, **it is no surprise that most launches to LEO, ISS, GTO, HEO and ES-L1 orbits have been done in KSC and CCSFS, while VSFB is primarily used for orbits that need to go from 60 to 90 degrees to the Earth's equator (SSO, PO and some LEO orbits). By the orbit that the launches require to follow, it also explains why successful landings from VSFB launches are more commonly done in drone ships, while KSC and CCSFS have less of a difference between ground pad and drone ship landings.**

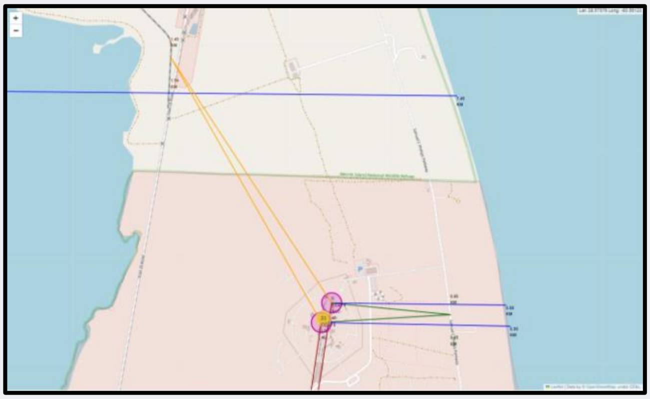




Analyzing **the distance from CCSFS (the most common launch base used by SpaceX) from human-made proximities, it is located approximately at 0.62 km from the nearest parkway** (Samuel C. Phillips Parkway), **at around 1.51 km from the closest railroad** (NASA Railroad) **and roughly at 18.20 km from the next city** (Cape Canaveral).

It is completely understandable the distance between a launch base and civilian settlements, as this is made by safety reasons in case of an unexpected failure, even if most launches are directed with an east direction (the nearest road used by citizens is the Kennedy Parkway North at about 7 km West).

But the distance can be relatively close from roads and train tracks if they are used for the Space Station, as the distances are not above 2 km from each, with makes it easier to transport parts and vehicles to the launch pad, and the restricted access to those sites facilitate the prevention of casualties in the worst-case scenario.



IBM Developer
SKILLS NETWORK

Section 4

Build a Dashboard with Plotly Dash

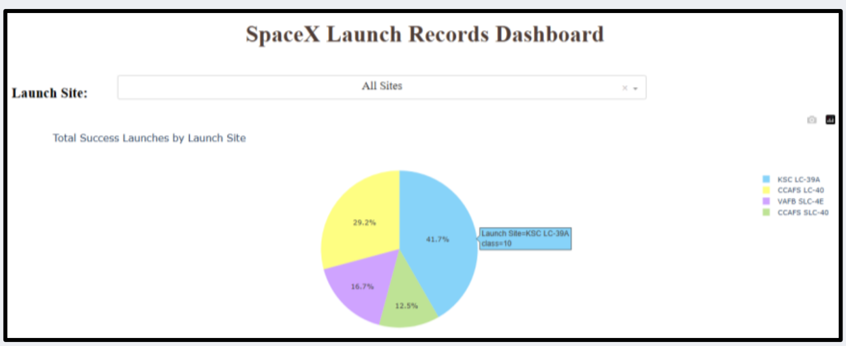
Total Successful Launches by Launch Site

For the development of the Dashboard **the first 56 launches have been chosen (from June 4, 2010, to June 4, 2018), this is done to speed up the testing of the interactive environment and obtain insights of the type of results that can be obtained using this tool**; so, for a more precise analysis the most recent data from SpaceX would be required.

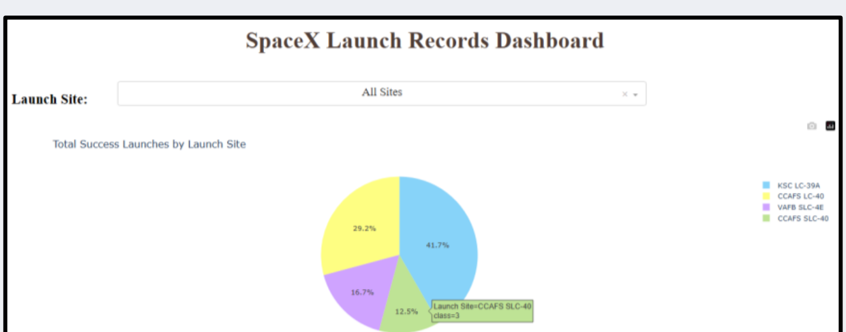
With the Dashboard, the first point of interest is the pie chart that can give in a simple observation how many of the landings of the first stage were successful for each site and how it corresponds to the total achieved landings.

Looking at the all-sites comparison of achieved landings, **the total successful launches refer to the Falcon 9 launches that could recover the first stage in proper conditions so that it can be reused in future launches**, and with that in mind the following conclusions can be made:

- **The Kennedy Space Center, Launch Complex 39A ("KSC LC 39A") has successfully landed the first stage on 10 of their total launches, which corresponds to the 41.7% of all the achieved landings made with the Falcon 9 rockets.**

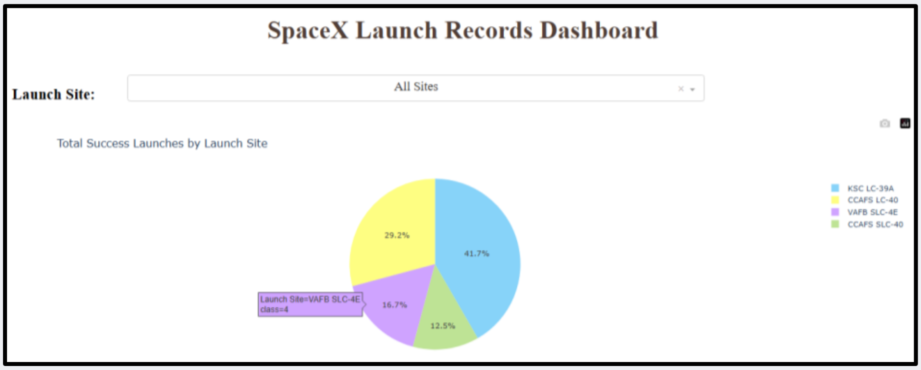


- **The Cape Canaveral Space Force Station, Space Launch Complex 40, under the acronym "CCAFS SLC-40" from the launch No. 45 (December 15, 2017) onwards, has successfully landed the first stage on 3 of all their launches, which makes 12.5% of all the accomplished landings.**

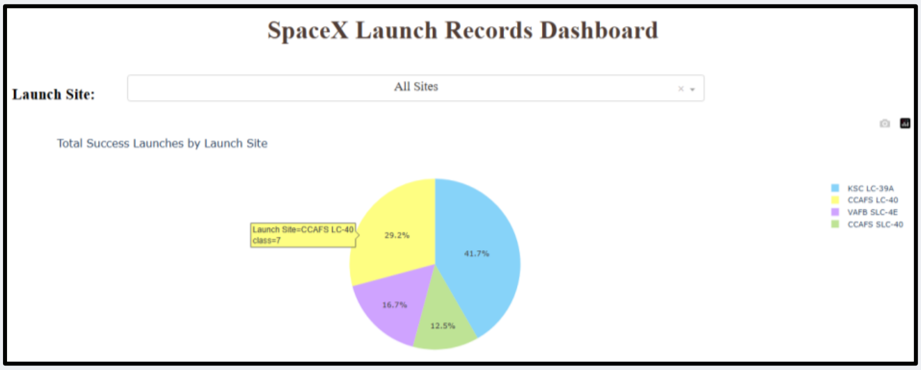


Total Successful Launches by Launch Site

- For the Vandenberg Space Force Base, Space Launch Complex 4E ("VAFB SLC 4E") 4 of all their launches ended in recovering the first stage for reuse in the future, this quantity makes up to 16.7% of all the landings done successfully.



- And finally, the Cape Canaveral Space Force Station, Space Launch Complex 40, under the acronym "CCAFS LC-40" from the launch No. 1 (June 4, 2010) up to the 28 (August 14, 2016), successfully landed the first stage on 7 of all their launches made, which makes 29.2% of all the accomplished landings.

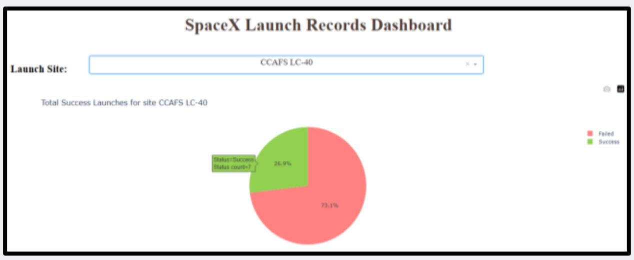


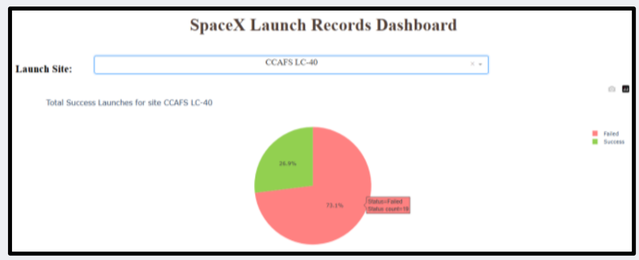
In general, the launch base that contributed the most to the total successful landings has been the KSC LC-39A, followed by the CCSFS under both acronyms and lastly the VSFB, but to understand if this means that their launches during these years were more successful overall, it is needed to examine the successful and failed launches of each site.

Successful vs Failed Landings Comparison Between Launch Sites

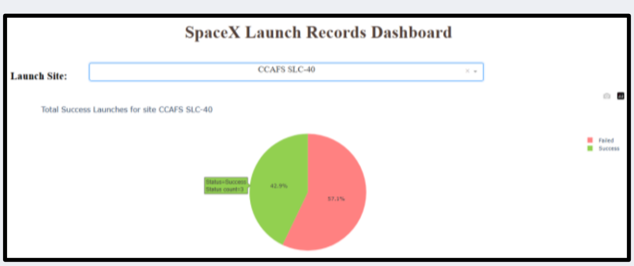
If the landing outcome of all the launches is analyzed for each site, the subsequent findings are:

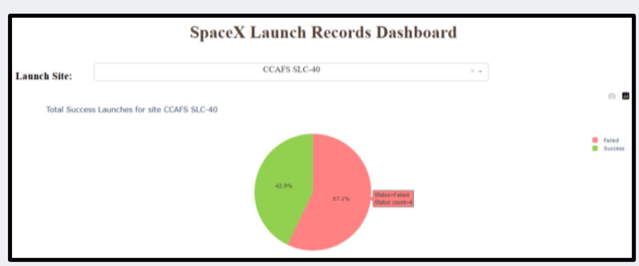
- The CCSFS under the acronym **“CCAFS LC-40” has made 26 launches from 2010 to 2016, in which 7 were successful (26.9%) and 19 failed in the landing (73.1%)**, the main reason of this high number of failed recoveries of the first stage is that 18 of the 26 were made with Falcon 9 v1.0 and v1.1, which none of them achieved to retrieve the booster in conditions to reuse it in future launches.





- The CCSFS under the acronym **“CCAFS SLC-40” launched 7 rockets from 2017 and beyond, in which 3 were successful (42.9%) and 4 failed in the landing (57.1%)**; as all this launches were made with the F9 Full Thrust (FT) and Block 4 versions, they are more interesting to review because this model was the first where the first stage could be recovered and reused. One point of interest is that three of the failed landings were never intended to land (it wasn’t made an attempt) and one is a controlled landing on the sea, which means that the booster cannot be used further.

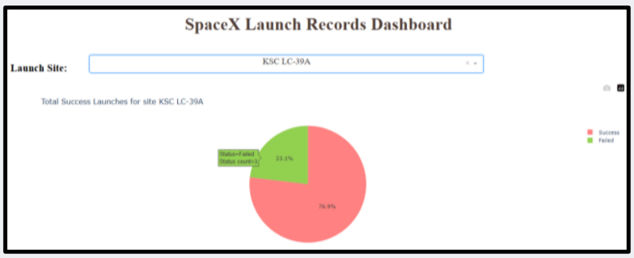


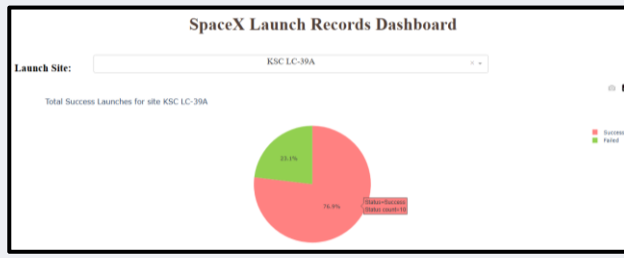


If both acronyms are considered as one, from June 4, 2010, to June 4, 2018, **10 of the 33 launches achieved recovering the first stage (30.3%) and 23 could not reuse the booster (69.7%)**

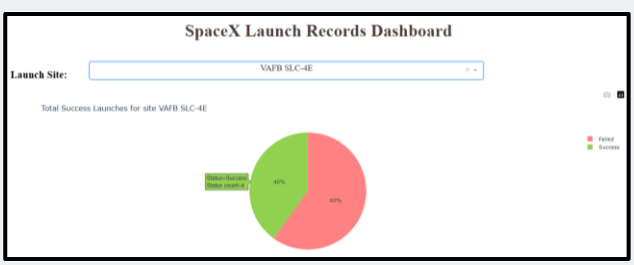
Successful vs Failed Landings Comparison Between Launch Sites

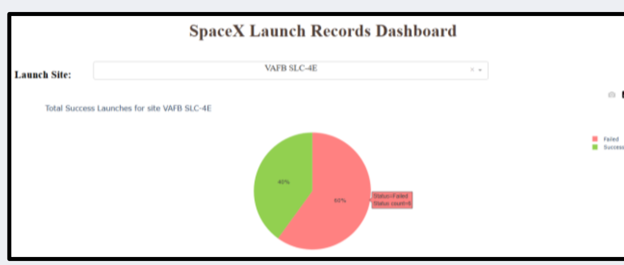
- The **"KSC LC 39A" has made 13 launches in total from February 19, 2017, to May 11, 2018, in which 10 were successful (76.9%) and 3 failed in the landing (23.1%)**, all of this launches were made with F9 FT, Block 4 and Block 5 so this recovery rate isn't strange, but something truly remarkable is that the three failed landings were intended (this means that SpaceX did not have the intention to retrieve the first stage for future uses), which means that the requirements of the launch are more related to the not retrieval of the booster rather than launch conditions that made it unviable.





- And lastly, the **"VAFB SLC 4E" has made 10 launches from 2013 to 2018, in which 4 were successful (40.0%) and 6 failed in the landing (60.0%)**, this is interesting as F9 v1.1, FT and Block 4 versions were used, and only two of the unsuccessful recovers were using the version v1.1, the other 4 were with the Full Thrust, but they were either controlled landings on the sea or a retrieval wasn't attempted at all.





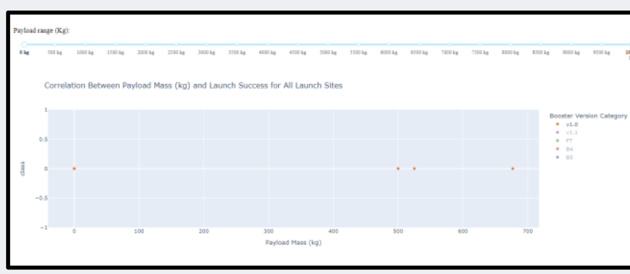
In the end, the contribution made by the **"KSC LC 39A" site in the total successful landings really corresponds to the recovery rate that it had, and not because of the total quantity of launches; nonetheless, this amount comes from the exclusive use of the Full Thrust version of the booster**, so eventually other sites should catch up or even surpass this recovery rate of the first stage booster as they continue using this model in future launches.

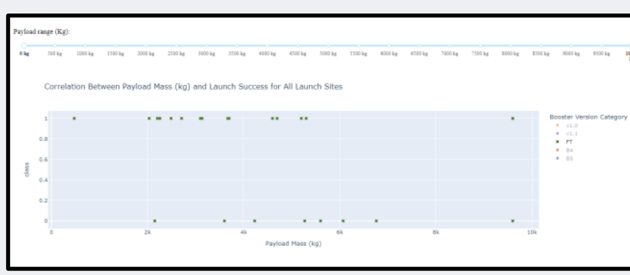
Payload vs Launch Outcome Analysis

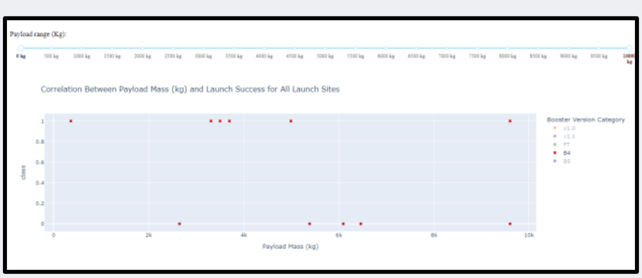
The second point of interest in the Dashboard is the Payload vs Launch Outcome comparison between booster versions at different ranges of mass, this can explain the evolution of the Falcon 9 models and confirm the payload mass range where boosters could be retrieved.

Using the slider to define different ranges for all the sites, these are the results:

- **Leaving the range from 0 to 10,000 kg shows that from the 56 launches**, 24 (43%) of the launches are considered successful **(this is a mistake as it appears as one v1.1 booster landed in conditions, when not one achieved this feat, so in reality the number is 23 (41%))** and 32 (57%) failed **(considering the mistake, it should be 33 (59%))**.
- **The version v1.0 was launched 5 times and not one recovered the first stage booster (100% failed recoveries).**
- **The version v.1.1 took off 15 times, and every single one did not recover the first stage (100% failed retrievals)**, even if the graph shows otherwise.
- **The FT version was sent 24 times to space, 16 of those were successful (67%) and 8 failed in the recovery (33%).**
- **The Block 4 (B4) was launched 11 times, 6 landed successfully (55%) and 5 could not be recovered (45%).**
- **The Block 5 (B5) only was launched once** (as the continuous use of this version started on July 22, 2018 and the Dataset covers to June 4, 2018) **and it was recovered (100% recovery rate)**.

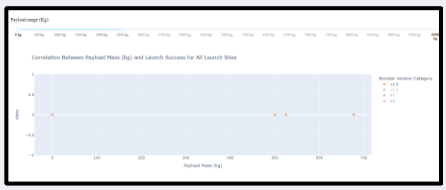




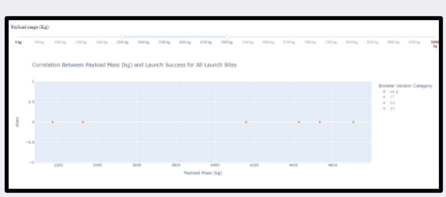
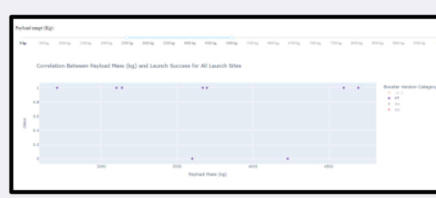


Payload vs Launch Outcome Analysis

- From the **21 launches done in the range from 0 to 2,500 kg, 6 launches were successful landing the first stage (29%) and 15 were not able to use again the booster (71%).**
- **All launches made during the lifespan of the v1.0 booster (5 in total) were done in this range of payload mass, and all of them couldn't retrieve the first stage booster**, either by not trying to recover it or by failing the landing completely (100% failure rate).
- **Nine (9) launches were done with the v1.1 booster, and all of them failed in landing the first stage** (100% failed landings), even though the graph shows 1 of them was retrieved (this is an error on the dataset).
- **Six (6) of the launches were made with the FT version (Block 1 through 3), 5 of them successfully recover the first stage (83%) while only 1 was not able to be reused (17%)**, and this one was because SpaceX did not try to recuperate it.
- **Only one launch (1) was done with the Block 4 version (B4), and it landed successfully to be reused again on June 29, 2018** (100% recover rate).

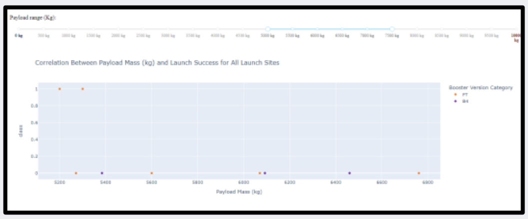



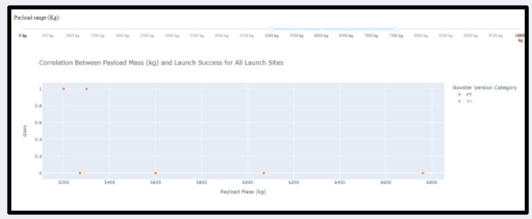

- Likewise, **21 launches were made in the range from 2,500 to 5,000 kg, and inside this range, 12 launches were successful landing the first stage (57%) and 9 were not able to use again the booster (43%).**
- The **rest of the launches made during the lifespan of the v1.1 booster (6) were done in this range of payload mass, and all of them failed retrieving the first stage booster**, since SpaceX did not try to land the first stage of this version at all (100% failure rate).
- **Nine (9) launches were done with the FT version (Block 1 through 3), where 7 of them were able to retrieve the first stage (78%) and 2 failed in retrieving them to be reused (22%)**, from these two, one landed in the ocean, and one crashed on the drone ship.
- **Five (5) of the launches correspond to the Block 4 version (B4), 4 of them successfully recover the first stage (80%) while only 1 was not able to be used again (20%)**, as SpaceX did not attempt to recuperate it.
- **Only one launch (1) was done with the Block 5 booster (B5), and it landed successfully to be reused again** on August 7, 2018 (100% recover rate).

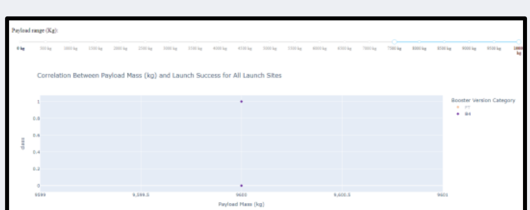
Payload vs Launch Outcome Analysis

- **At the 5,000 to 7,500 kg range can be seen that fewer launches have been performed (9 in total)** as these require FT booster models onwards (including Block 4 and 5), **inside this range, 2 launches were successful landing the first stage (22%) and 7 were not able to use again the booster (78%)**.
- **Six (6) launches were done with the FT version (Block 1 through 3), where 2 of them were able to retrieve the first stage (33%) and 4 failed in retrieving them to be reused (67%)**, from these four, three weren't intended to land, and one "landed hard" on the drone ship.
- **The last three (3) launches correspond to the Block 4 version (B4), and all of them were not recovered for future uses** as SpaceX did not intend to retrieve them (100% failure rate)





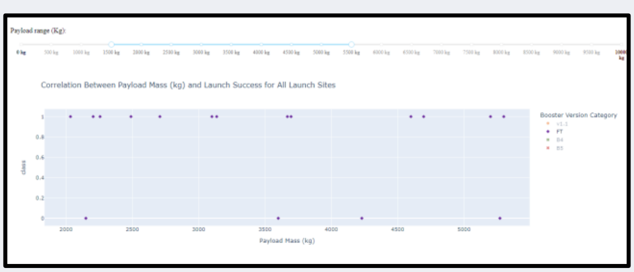
- **And finally, 5 launches were made in the range from 7,500 to 10,000 kg, where 3 launches were successful landing the first stage (60%) and 2 were not able to use again the booster (40%)**.
- **Three (3) take offs were done with the FT version (Block 1 through 3), where 2 of them were able to retrieve the first stage (67%) and one cannot be retrieved to be reused (33%) as it landed in the ocean**.
- **The other two (2) launches are from the Block 4 version (B4), one of them successfully recover the first stage (50%) while the other was not able to be used again (50%)**, as SpaceX did not attempt to recuperate it.



Payload vs Launch Outcome Analysis

- The range where most of the launches were performed from June 4, 2010, to June 4, 2018, was from 1,500 to 5,500 kg, with a total of 35 launches; from these launches, 18 were successful landing the first stage (51%) and 17 were not able to use again the booster (49%).
- Many of the launches made during the lifespan of the v1.1 booster (11) were done in this range of payload mass, and all of them failed retrieving the first stage booster (100% failure rate).
- Seventeen (17) launches were done with the FT version (Block 1 through 3), where 13 of them were able to retrieve the first stage (76%) and 4 failed in retrieving them to be reused (24%), from these four, one landed in the ocean, one wasn't attempted to be recovered and two crashed on the drone ship.
- Six (6) of the launches correspond to the Block 4 version (B4), 4 of them successfully recover the first stage (67%) while 2 were not able to be used again (33%), as SpaceX did not attempt to recuperate it.
- The only launch done with the Block 5 booster (B5) was performed in this range, and it landed successfully to be reused again on August 7, 2018 (100% recover rate).

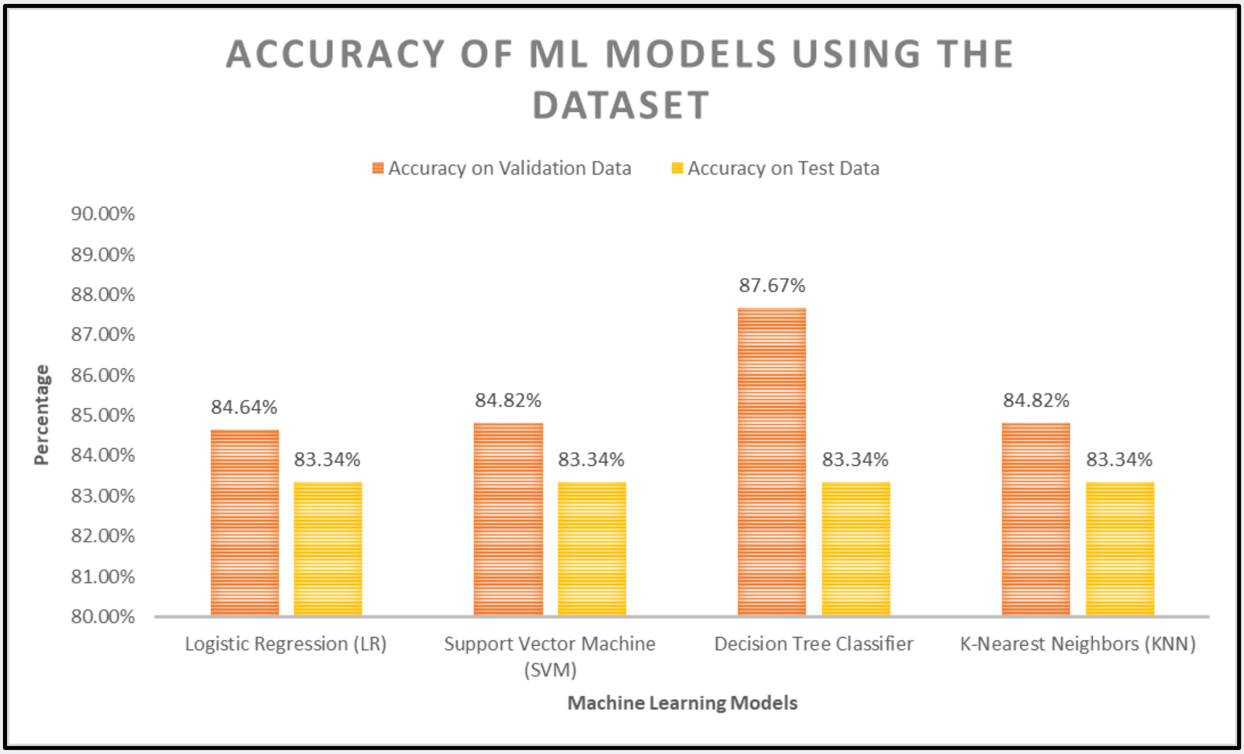
Even though it may seem that a middling percentage (51%) of the Falcon 9 boosters could be reused, it must be considered that many launches were done with a version that never landed successfully in its lifespan (v1.1) and the range of dates only take into account one Block 5 launch, if this data included newer launches, it would be less skewed to retired models, but as an informative tool of all the models, it accomplished its purpose, as it can define the range were it is most likely to recover a first stage booster.



Section 5

Predictive Analysis (Classification)

Classification Accuracy

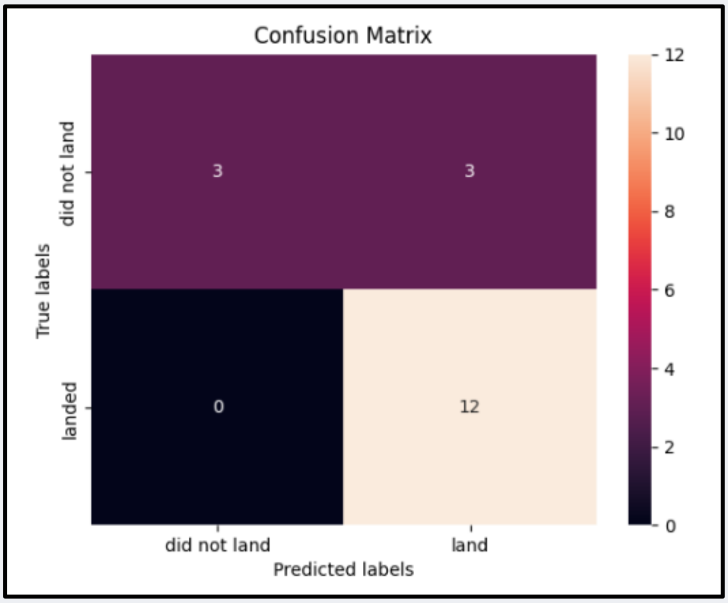


Looking at this bar chart, **the Decision Tree Classifier got the highest accuracy on the validation data used by Grid Search with the best parameters selected (87.67%), but when every Model is tried with the Test portion of the SpaceX Falcon 9 Dataset, every single one achieved the same accuracy (83.34%)**.

The Decision Tree depends on the seed that it uses at the start, which could improve or worsen the accuracy with the test data, but this most likely signifies that more data is required in the training set for the machine learning models to find a pattern between successful and failed landing outcomes of the first stage.

But this alone don't provide enough information to see where the inaccuracies occur in the predictions, for this reason, **the confusion matrix of the models must be analyzed to find if this impreciseness is because of the presence of false positives** (meaning that first stages that did not land were predicted as if they landed) **or false negatives** (boosters that did land were predicted as if they didn't land).

Method	Logistic Regression (LR)	Support Vector Machine (SVM)	Decision Tree Classifier	K-Nearest Neighbors (KNN)
Accuracy on Validation Data	84.64%	84.82%	87.67%	84.82%
Accuracy on Test Data	83.34%	83.34%	83.34%	83.34%
Test Samples	18	18	18	18
Land (Test Positives)	12	12	12	12
Did not Land (Test Negatives)	6	6	6	6
True Positives (Quantity)	12	12	12	12
True Positives (Percentage)	100%	100%	100%	100%
True Negatives (Quantity)	3	3	3	3
True Negatives (Percentage)	50%	50%	50%	50%
False Positives (Quantity)	3	3	3	3
False Positives (Percentage)	50%	50%	50%	50%
False Negatives (Quantity)	0	0	0	0
False Negatives (Percentage)	0%	0%	0%	0%



In every single ML Model used with this Dataset the predictions produced would have a 100% accuracy in the landing of the first stage of future launches using the variables assigned, but at the cost of 50% of false positives. In other words, every Model generate the same results.

This can mean that in the training set most launches assigned were from the FT, B4 and B5 Models, which have a better success rate than previous models and if the test set includes a mixture of all models, could skew the results to a favorable outcome.

IBM Developer SKILLS NETWORK

Section 6

Results and Conclusions

Results

Exploratory Data Analysis (EDA) Results:

- **The most used site for launch is "CCSFS SLC 40" followed by "KSC LC 39A" and at last with "VAFB SLC 4E"**, as most launches done are for an orbit with an angle from 0 to 60° from the equator, and the location of the first two bases simplify sending the rockets to the east, following Earth's rotation.
- **Most launches done with all models of Falcon 9 had a payload mass of less than 8,000 kg (74%), and every launch above this point used FT, B4 and B5 models, but have never gone for masses of 16,000 kg and beyond**.
- **There is a correlation between Success rate in each orbit and the number of launches performed to said destination, as less frequents orbits have better success rates than common ones**; also, **the more frequent orbits had launches from v1.0 and v1,1 models**, which never achieved a successful landing and can skew the results against them.
- **The relationship between the Payload Mass and the Orbit can be resumed as the closer the orbit the payload has to travel, the heavier it can be**, which means that only PO, ISS and VLEO had masses above 8,000 kg.
- **The upwards yearly trend in Successful retrieval of first stage booster is caused by improvements done in newer Falcon 9 Models, as SpaceX could reuse its boosters with the arrival of the FT model**, that could land the first stage in conditions for future uses, **and the dip saw in 2018 can be explained by the transition between FT, B4 and B5 rockets**, in which many test and changes were done to the Falcon 9 until arriving at the current Block 5.
- **SpaceX had a really high success rate in their Falcon 9 missions, with only one unclear status and one failed mission in all of their launches**.

Interactive Analytics Results:

- Falcon 9 boosters with destination to **LEO, ISS, GTO, HEO and ES-Lagrange points orbits are sent from "CCSFS SLC 40" and "KSC LC 39A" launch sites, while PO, SSO and LEO orbits with 60 degrees and beyond to the Earth's equator are launched from "VAFB SLC 4E"**.
- **Most launches are performed in a payload mass range from 1,500 kg to 5,500 kg and the success rate of the FT model have been improving with the introduction of newer Blocks (Block 4 and 5)**, as most failed recoveries are planned and tend to be less frequent in later years.

Predictive Analysis Results:

- While the **Decision Tree had the highest accuracy in the validation data (87.67%), when it was used with the test samples from the Dataset, it provide the same results** as the Support Vector Machine (SVM), Logistic Regression (LR) and the K-Nearest Neighbor (KNN) models: **an accuracy of 83.34%, composed by 100% in true positives and 50% in true negatives, having a 50% of false positives; meaning that every model generates the same results.**

Conclusions

All this project made easy to see the improvements made in the Falcon 9 boosters with each new model, starting from zero first stage recoveries from 2010 to 2014 in V1.0 and v1.1 versions and changing this drastically from 2015 with the introduction of the Full Thrust (FT) Model, and improving to a high success rate with the Block 5 (B5), only having a dip in 2018 by the transition from FT to B4 and B4 to B5 version.

This means that Predictive Models require to focus their training mainly in models from FT onwards, as this can provide a clearer picture of which conditions can assure that a rocket will be reused or not; moreover, some conditions have shown a pattern that could be helpful in future training of these models:

- The Payload Mass (kg) in each launch has a bigger influence in determining if a retrieval attempt is going to be performed by SpaceX; as seen with the LEO, VLEO and ISS orbits, they have a maximum payload mass of around 15,600 kg, even though it is stated that FT models including B4 and B5 can carry up to 22,800 kg to these orbits; which means that there is a maximum mass in which the booster has enough fuel to sent it to orbit and return successfully for another use. If it is analyzed further, it can be determined if boosters with payloads bigger than this maximum will not be recovered purposefully.
- The times that a specific booster model have been used can determine if it will be recovered for future launches, as a higher amount makes it more likely that SpaceX will not land the first stage, for this reason, looking at the maximum reuse times of previous booster serials could give a better picture if the next use of a particular serial number will be the last or not.

Other Sources

- Falcon 9 Capabilities & services, SpaceX, retrieved 30 June 2023.
- Falcon 9: first orbital class rocket capable of reflight, SpaceX, retrieved 30 June 2023.
- James Cawley (January 19, 2020), "NASA, SpaceX Complete Final Major Flight Test of Crew Spacecraft", NASA Blogs, retrieved 30 June 2023.
- "L1, the first Lagrangian Point", European Space Agency (ESA), retrieved 30 June 2023.
- List of Falcon 9 and Falcon Heavy launches (2010–2019), Wikipedia, retrieved 30 June 2023.
- List of Falcon 9 and Falcon Heavy launches, Wikipedia, retrieved 30 June 2023.
- List of orbits, Wikipedia, retrieved 30 June 2023.
- Types of orbits, European Space Agency (ESA), 30 March 2020, retrieved 30 June 2023.

IBM Developer
SKILLS NETWORK

Thanks for Reading