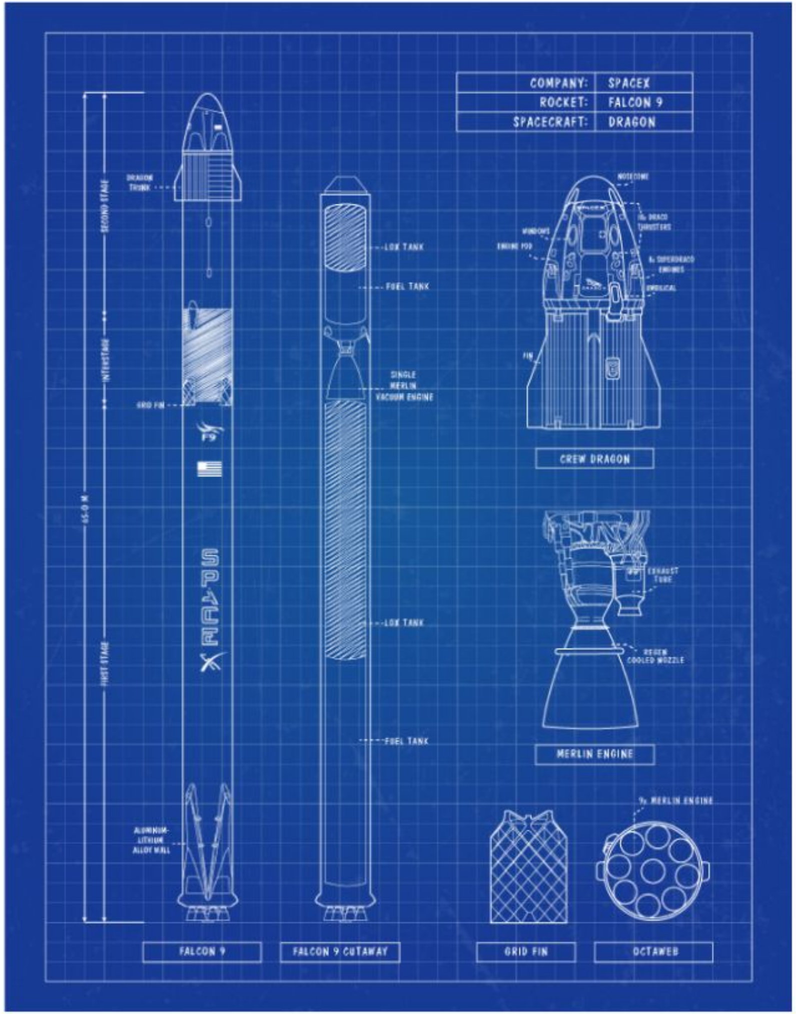
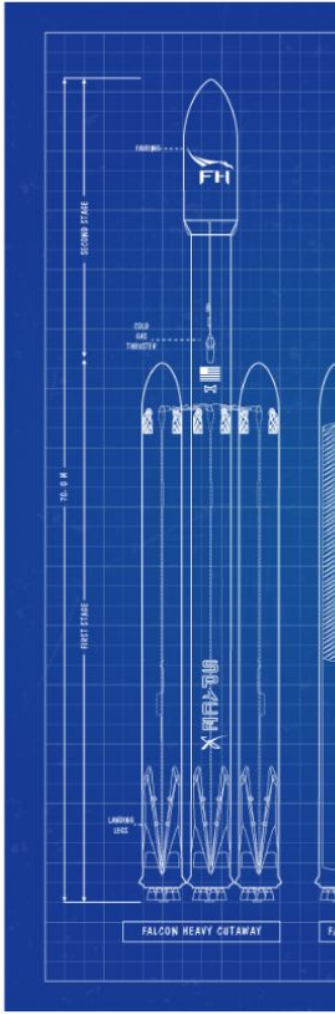
Data Findings Report: SpaceX Falcon 9 Rocket

Blake Bannon
Data Analyst

Executive Summary





Executive Summary

This report presents the findings from a comprehensive study of the SpaceX Falcon 9 rocket, focusing on its performance, reusability, and impact on the commercial space industry. The Falcon 9, a two-stage reusable rocket developed by SpaceX, has revolutionized space launch capabilities through cost efficiency and innovative design. Key findings include focus on the following elements for this analysis.

- Generalized and relevant data on data procurement and data wrangling.
- Number of Successful Landings and Failed Landings.
- Machine Learning model to accurately predict future launches with an 87.67% accuracy score.
- Launch Site location data.

This report details the data collected, methodologies used, and implications for future space exploration.

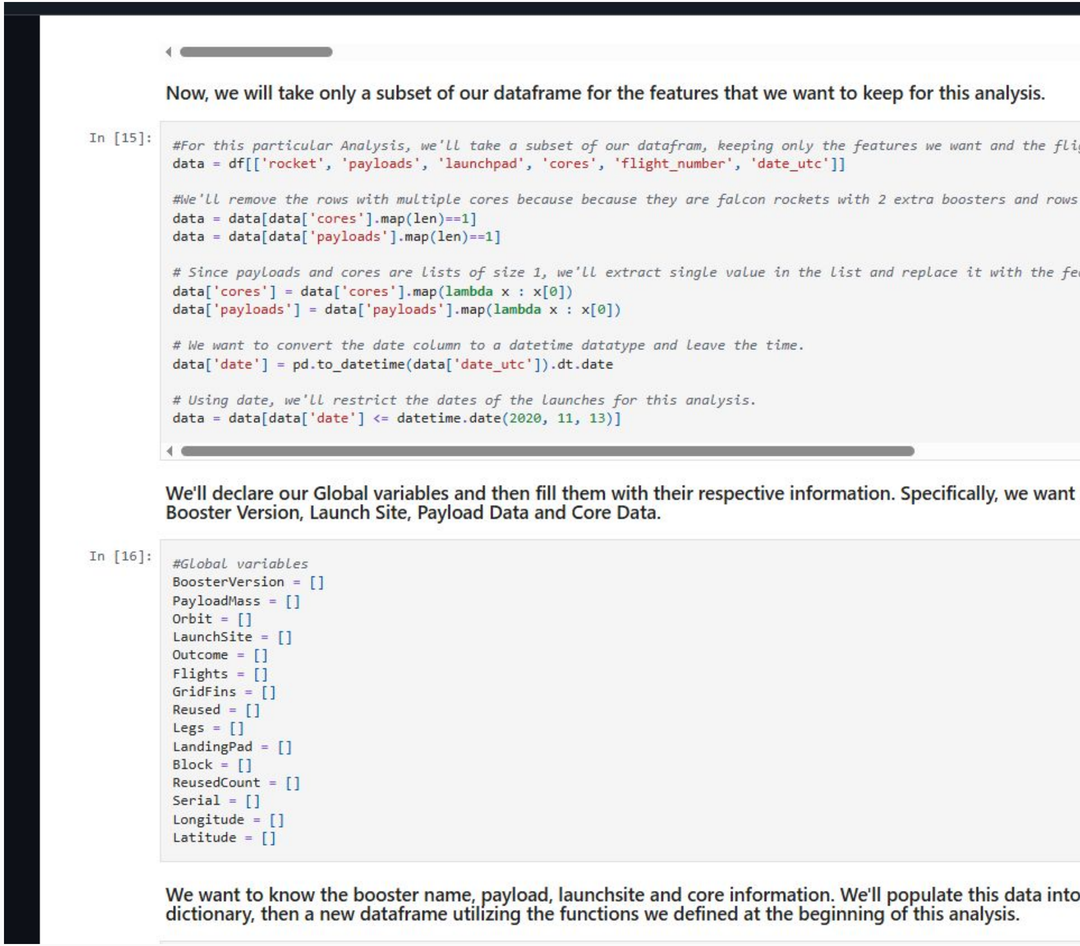
Introduction

The SpaceX Falcon 9 is a partially reusable, medium-lift launch vehicle designed to deliver payloads to low Earth orbit (LEO), geostationary transfer orbit (GTO). It is important to note in this context that the Falcon 9 is capable of missions that other orbit patterns as well. Since its debut in 2010, it has become a cornerstone of SpaceX’s mission to reduce space travel costs. This report analyzes specifically the success vs failure rates for landings and was used to train a predictive model for referencing future launches, drawing on data from an API provided by *spacexdata.com*. As well as additional data that has been webscraped from *Wikipedia*.

Methodology

Data Collection: Data was collected utilizing *spacexdata.com*'s API to specifically collect data on the Falcon 9 Rocket from 2010 to 2020. Additional data was web-scraped from Wikipedia tables as well. Data cleaning or "wrangling" was performed on each dataset, respectively, to ensure that data was useable in a meaningful way for further analysis utilizing Jupyter Notebooks, Python, and a variety of libraries to assist with different portions of this analysis.

*Photo from SpaceX_Data_Collection_via_API, URL:https://github.com/corfios/Falcon9DataExploration/blob/main/SpaceX_Data_Collection_via_API.ipynb, Blake Bannon



Now, we will take only a subset of our dataframe for the features that we want to keep for this analysis.

In [15]:

```
#For this particular Analysis, we'll take a subset of our datafram, keeping only the features we want and the fli
data = df[['rocket', 'payloads', 'launchpad', 'cores', 'flight_number', 'date_utc']]

#We'll remove the rows with multiple cores because because they are falcon rockets with 2 extra boosters and rows
data = data[data['cores'].map(len)==1]
data = data[data['payloads'].map(len)==1]

# Since payloads and cores are lists of size 1, we'll extract single value in the list and replace it with the fe
data['cores'] = data['cores'].map(lambda x : x[0])
data['payloads'] = data['payloads'].map(lambda x : x[0])

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

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

We'll declare our Global variables and then fill them with their respective information. Specifically, we want Booster Version, Launch Site, Payload Data and Core Data.

In [16]:

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

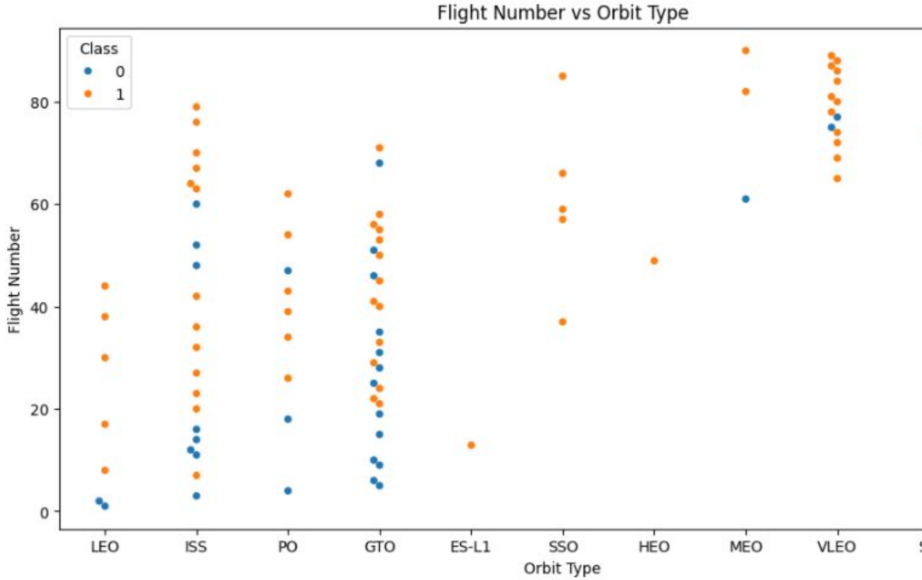
We want to know the booster name, payload, launchsite and core information. We'll populate this data into dictionary, then a new dataframe utilizing the functions we defined at the beginning of this analysis.

Methodology

Data Preparation and Analysis: Data was prepared and analyzed utilizing numpy, pandas, matplotlib and seaborn in order to create and assign relevant values that could be utilized for various seaborn plots. Additionally, analyzed data on flights and launch sites provided interesting insights into a variety of factors to be covered in more detail later in this report.

*Photo from SpaceX_Data_Collection_via_API, URL:https://github.com/corfios/Falcon9DataExploration/blob/main/Falcon9_EDA_Data_Preparation.ipynb, Blake Bannon

```
plt.title('Flight Number vs Orbit Type')
plt.xlabel('Orbit Type')
plt.ylabel('Flight Number')

plt.show()

summary = df.groupby('Orbit')['FlightNumber'].describe()
print("Summary Statistics of Flight Number by Orbit:")
print(summary)
```



Summary Statistics of Flight Number by Orbit:

Orbit	count	mean	std	min	25%	50%	75%	max
ES-L1	1.0	13.000000	NaN	13.0	13.00	13.0	13.0	13.0
GEO	1.0	83.000000	NaN	83.0	83.00	83.0	83.0	83.0
GTO	27.0	35.037037	18.842427	5.0	21.50	33.0	50.5	71.0
HEO	1.0	49.000000	NaN	49.0	49.00	49.0	49.0	49.0

Methodology

Limitations: Because our data only reflects launches from 2010 to 2020, we are missing out on several additional years worth of data and technological changes at Space X that would otherwise provide valuable insights, and as such, this data should be utilized in reflection of that understanding.

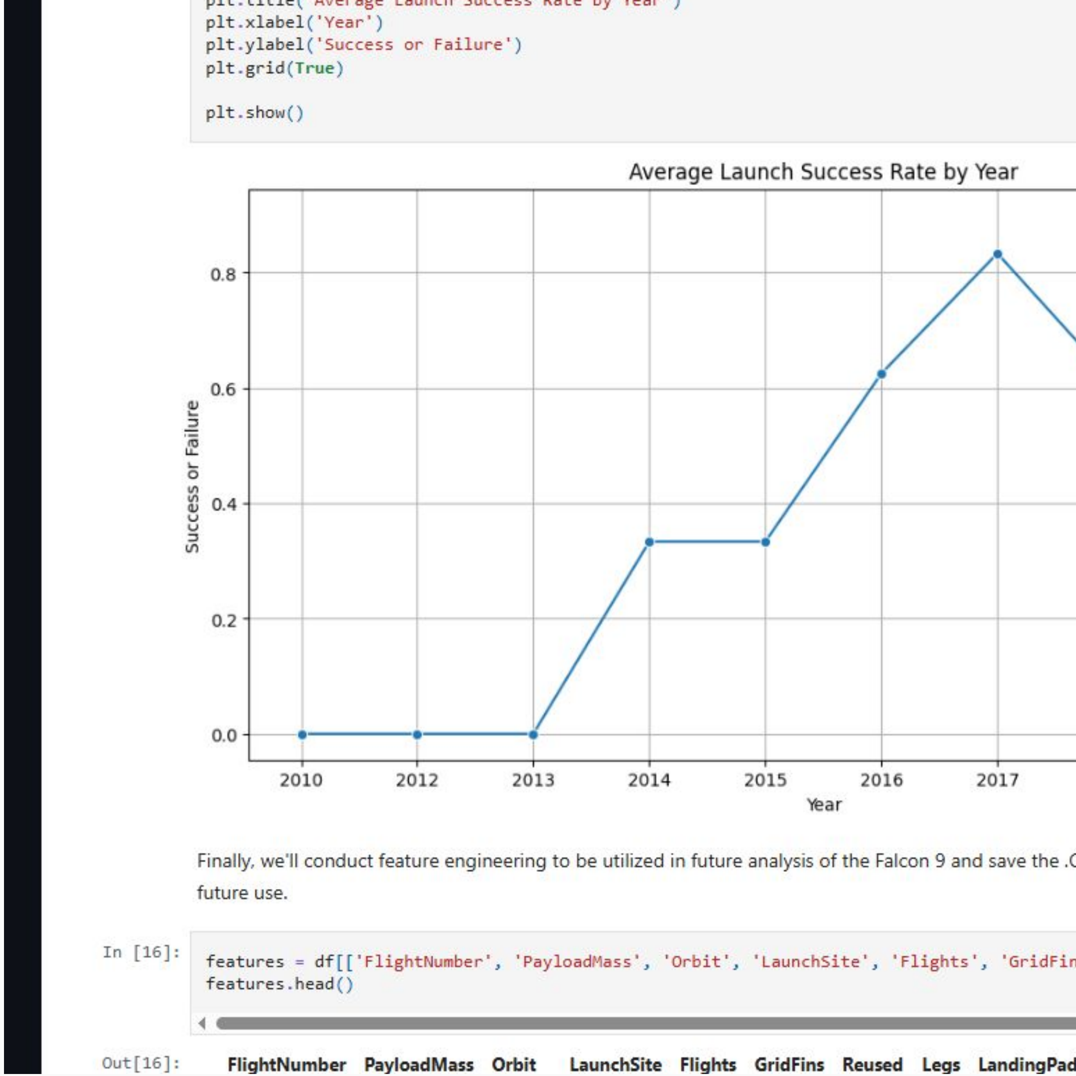
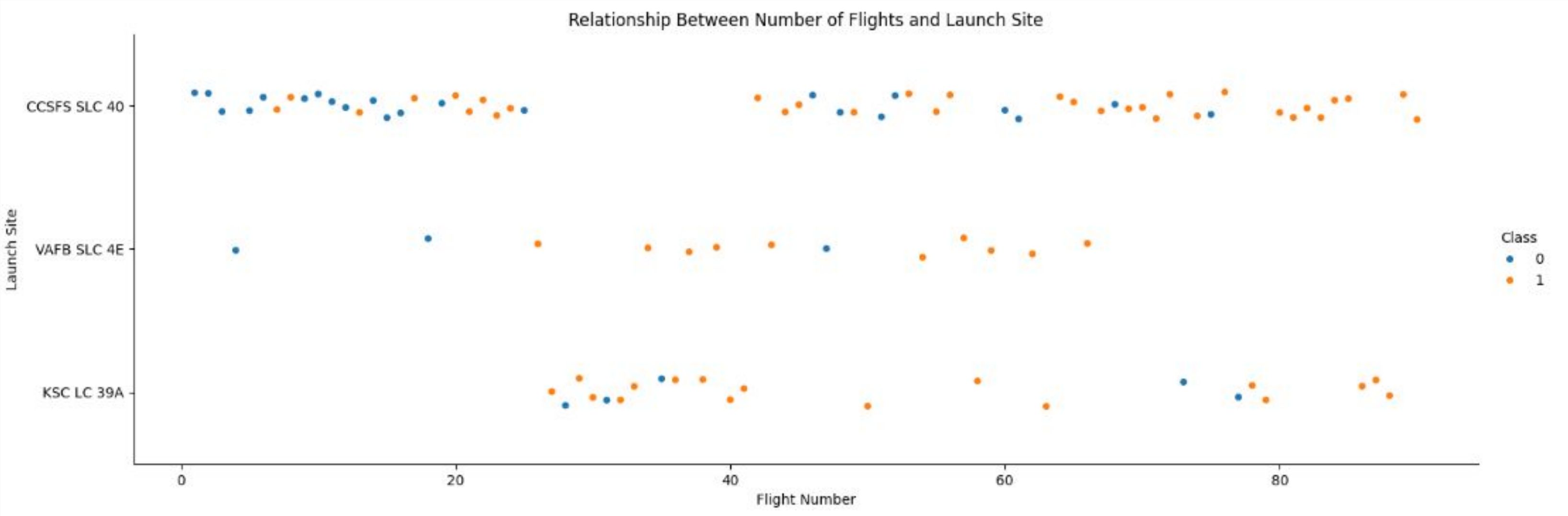


*Photo from SpaceX_Data_Collection_via_API, URL:https://github.com/corfios/Falcon9DataExploration/blob/main/Falcon9_EDA_Data_Preparation.ipynb, Blake Bannon

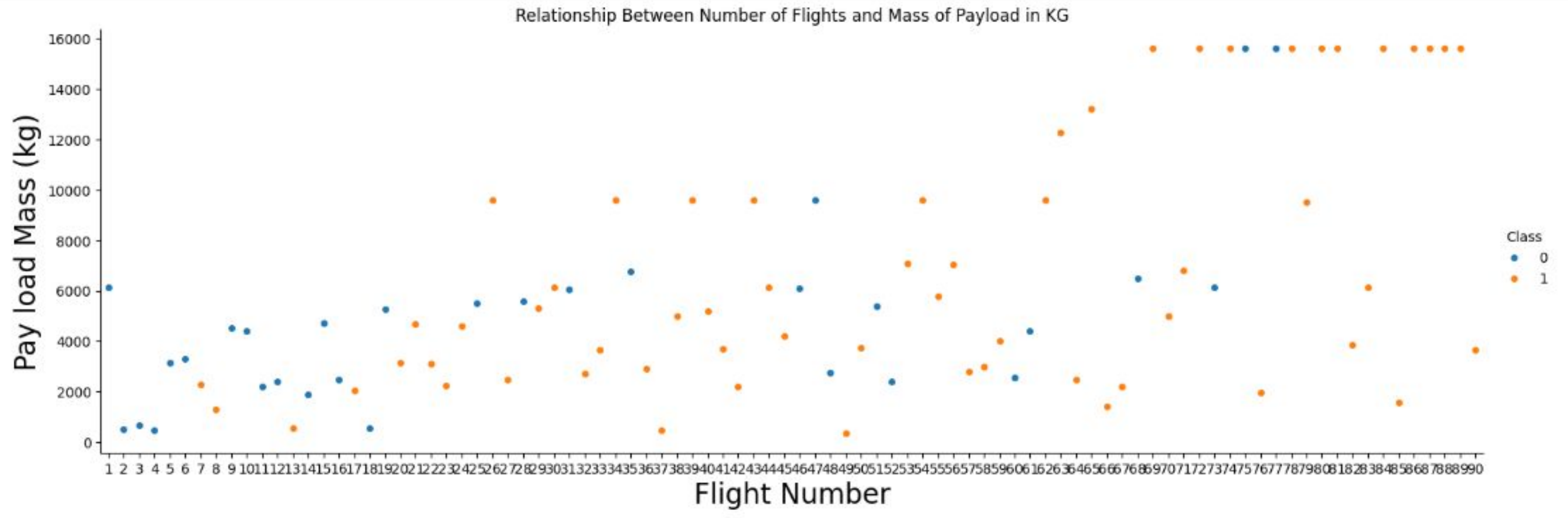
Relationship Between Number of Flights and Differing Launch Sites

Perhaps somewhat predictably, as the number of flights conducted by Space X went up, the number of successful landings increased as well, as visualized in the plot below. We can also see an interesting pattern of distribution in terms of which launch sites Space X conducted their launches at. **Blue dots** represent failure to land, while **orange dots** represent successful landings.



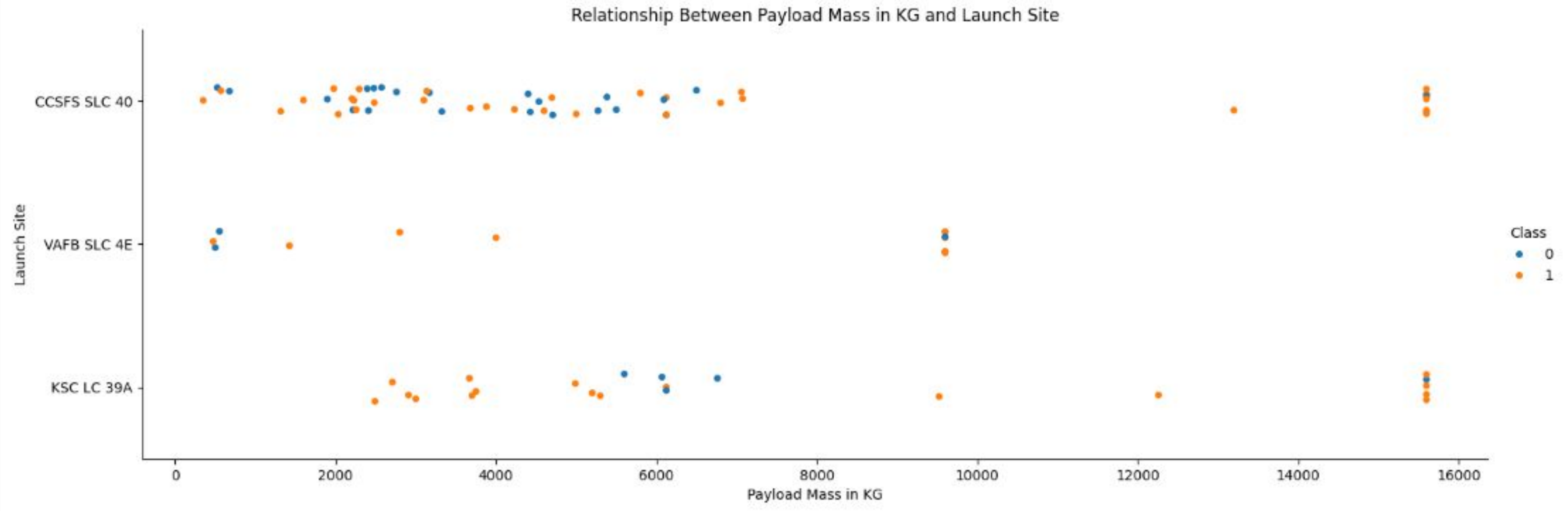
Relationship Between Number of Flights and Mass of Payload

As indicated by the plot below, it becomes fairly apparent that, as the number of flights begins to increase, the tolerance for the amount of weight in each launch increases. Similarly, we can see that the number of failed landings begins to somewhat decline, particularly after roughly 70 launches.



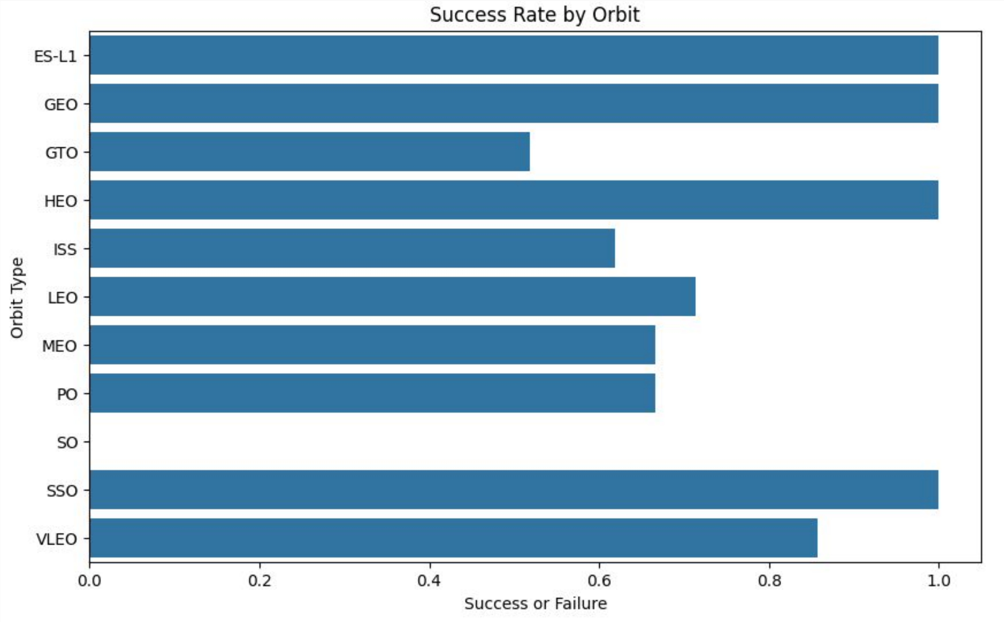
Relationship Between Payload Mass and Launch Site

We can fairly clearly see that with higher Payload Mass, the number of successful flights begins to increase. This is likely due to lower payload flights failing during testing and refining of SpaceX technology, where as that technology gradually improves not only the Payload Capacity, but also the number of successful flights.



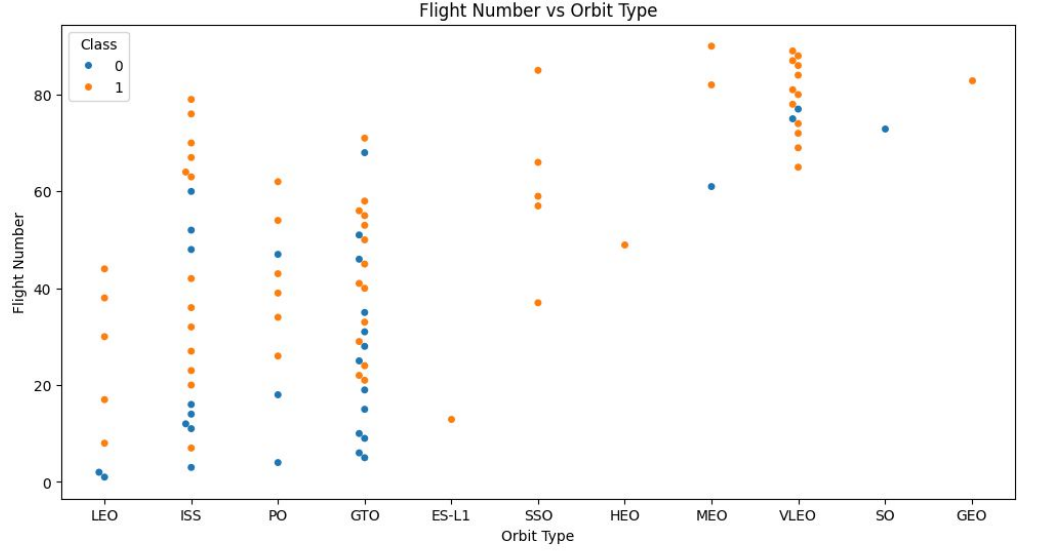
Success Rate by Orbit Type of Mission

To the right is an interesting comparison, where we can visualize the number of successful flights based on the type of orbit type the launch was intended for. ES-L1, GEO, HEO and SSO orbit launches all have successful landing rates of 100%.



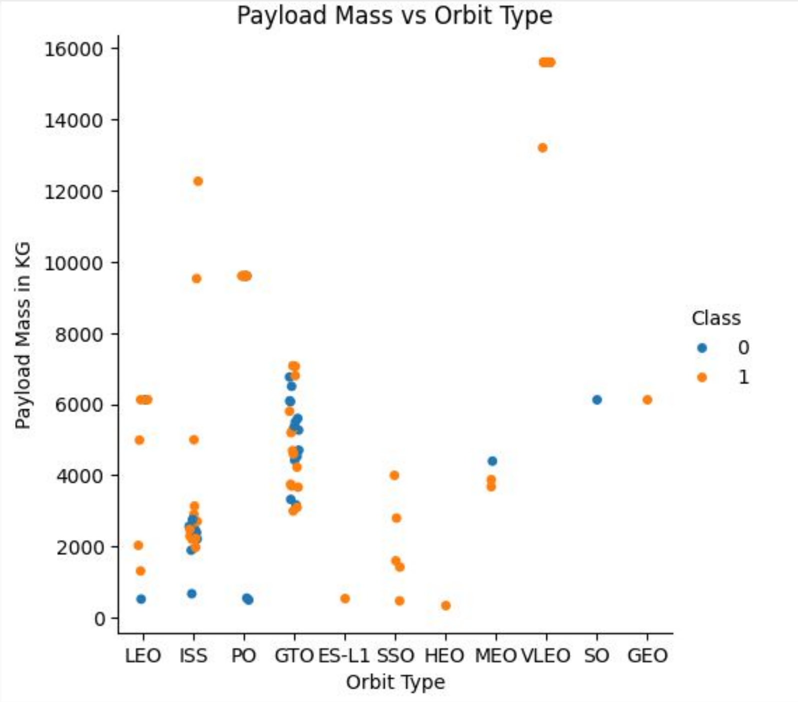
Success Rate by Orbit Type of Mission

Now we'll analyze the Number of Flights vs the Orbit Type. Interestingly, (or perhaps not so interestingly if you are a rocket scientist), the number of successful flights tends to increase for SSO, MEO and VLEO type orbits as the number of flights increase. Additionally we can see that these types of orbital launches were not even attempted until Space X had conducted over 40 launches.



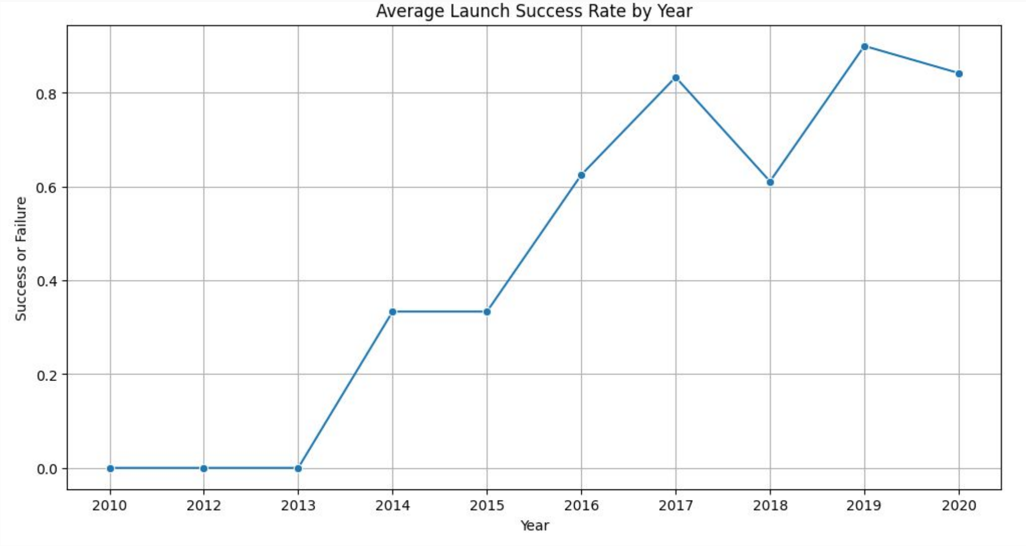
Relationship Between Payload Mass and Orbit Type

Here we can visualize the orbit type of a launch and the payload of each launch. We can see some interesting differences within LEO, for example, where more successes occur as the payload increases. Conversely, we cannot gather much relevant information for GTO type orbital launches as the cluster is compact.



Launch Successes by Year

This line plot is showing the success rate of the landings of the Falcon 9 rocket. In a previous portion of this analysis (see Falcon9_Data_Wrangling) we were able to conclude that the average success rate for the Falcon 9 was about 66.67%. When looking at 2017 to 2020, we can see that rate climbs nearer to above 85% success.

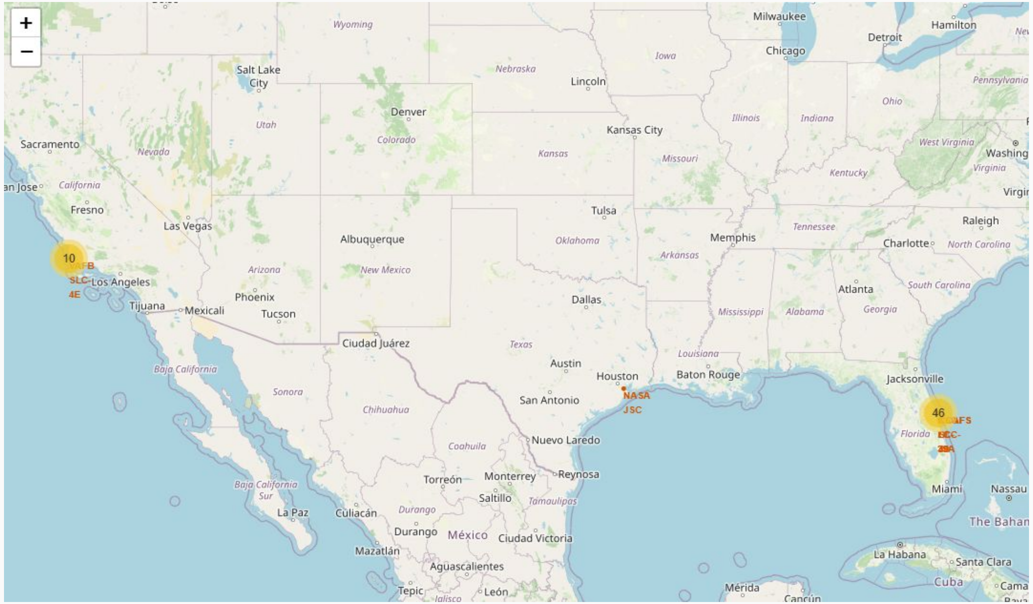


Launch Site Interactive Map

Utilizing Folium, an interactive map was produced utilizing the data gathered at the beginning of this map.

This map can be viewed in Jupyter Notebook via downloading from the link below to my Github. Note the interactive maps will not work without downloading the notebook and running it locally.

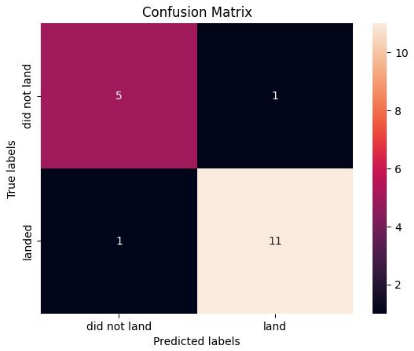
https://github.com/corfios/Falcon9DataExploration/blob/main/Falcon%209%20Interactive%20Visuals%20w%20Folium.ipynb



Machine Learning

Utilizing Different Machine Learning Models to Predict Landing Success

```
In [28]: print("tuned hpyerparameters :(best parameters) ",tree_cv.best_params_)
         print("accuracy :",tree_cv.best_score_)

tuned hpyerparameters :(best parameters)  {'criterion': 'entropy', 'max_depth': 6, 'max_features': 'sqrt', 'min_samples_lea
f': 2, 'min_samples_split': 2, 'splitter': 'best'}
accuracy : 0.8767857142857143

In [29]: test_accuracy_2 = tree_cv.score(X_test, Y_test)

         print("Test Accuracy is:", test_accuracy_2)

Test Accuracy is: 0.8888888888888888

In [30]: yhat = tree_cv.predict(X_test)
         plot_confusion_matrix(Y_test,yhat)
```



Machine Learning Models

This analysis utilized four separate Machine Learning Models, including Logistic Regression, Support Vector Machine(SVM), Decision Tree and a K-Nearest Neighbor (KNN) model

*Photo from SpaceX_Data_Collection_via_API, URL:https://github.com/corfios/Falcon9DataExploration/blob/main/Falcon9_EDA_Data_Preparation.ipynb, Blake Bannon

```
X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size=0.2, random_state=2)
```

```
Y_test.shape
```

```
(18,)
```

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

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

```
GridSearchCV(cv=10, estimator=LogisticRegression(),
             param_grid={'C': [0.01, 0.1, 1], 'penalty': ['l2'],
                         'solver': ['lbfgs']})
```

In a Jupyter environment, please rerun this cell to show the HTML representation or trust the notebook.
On GitHub, the HTML representation is unable to render, please try loading this page with nbviewer.org.

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

```
tuned hpyerparameters :(best parameters)  {'C': 0.1, 'penalty': 'l2', 'solver': 'lbfgs'}
accuracy : 0.8214285714285714
```

As we can see, using Logistic Regression provides underwhelming results and indicates it's likely not a great model to be using for predictive analysis in the case of Falcon 9 successful landings. With only an 82.14% accuracy, we'll see if we can generate a better model for this.

Machine Learning Model Takeaways

After training, testing, and changing sample sizes several times, the Decision Tree Classifier ended up being the most accurate predictive model for this particular analysis. With this, we can say that for future launches from Space X, we can effectively predict with 87.67% accuracy weather that launch will end in a successful landing or not.

*Photo from SpaceX_Data_Collection_via_API, URL:https://github.com/corfios/Falcon9DataExploration/blob/main/Falcon9_EDA_Data_Preparation.ipynb, Blake Bannon

```
tuned hpyerparameters :(best parameters)  {'criterion': 'entropy', 'max_depth': 6, 'max_features': 'sqrt',
f': 2, 'min_samples_split': 2, 'splitter': 'best'}
accuracy : 0.8767857142857143
```

“"In God we trust; all others must bring data."

– W. Edwards Deming

Overall, a variety of factors play into a successful Falcon 9 landing, and predictive modeling helps us see what those factors are.

Thanks!

Contact:

Blake Bannon
Denver, CO
https://www.linkedin.com/in/blake-bannon/