To Crash or Not to Crash, That is the Question

Gabrielle Chiarello February 2024

OUTLINE

- Introduction
- Executive Summary
- Methodology
- Results
- Discussion
- Conclusion

INTRODUCTION

As the galactic dust has settled, and The Space Race has nestled its way into conversational history, a modern era of space exploration has come to be. Companies like Rocket Lab and Virgin Galactic are making strides in a new and more affordable space travel experience. Perhaps the most successful of such companies is Space X. Space X, and their revolutionary rocket Falcon 9, has reduced rocket launch costs to as low as 62 million dollars, while competitors are reaching upwards of 165 million dollars per launch. The most significant cost reduction practice pertains to Falcon 9's ability to re-land its first stage rocket for reuse on a secondary launch. Space Y will be exploring Falcon 9 mission parameters to predict the landing outcome of stage one and determine the price of each launch.

EXECUTIVE SUMMARY

- Summary of Methodologies
  - Data Collection through API
  - Data Collection with Web Scraping
  - Data Wrangling
  - Exploratory Data Analysis with SQL
  - Exploratory Data Analysis with Data Visualization
  - Interactive Visual Analytics with Folium
  - Machine Learning Predictions
- Summary of Findings
  - Exploratory Data Analysis Results
  - Interactive Analytics Results
  - Predictive Analysis Results


Methodology
(the method behind the madness)

Data Collection

Data collection is the process of gathering and measuring information on variables of interest, in an established systematic fashion that enables one to answer stated research questions, test hypotheses, and evaluate outcomes. For this project, data was collected through the implementation of Rest API and Web scraping API. Below is a brief introduction to our collection methods. On the next two slides, we will go into greater detail on the processes

Rest API Overview

- Request to the SpaceX API
- Clean the requested data

Web Scraping API

- Extract a Falcon 9 launch records HTML table from Wikipedia
- Parse the table and convert it into a pandas data frame

Data Collection via Space X API

- Task one :
  - Request and parse the SpaceX launch data using the Get Request
- Task two:
  - Filter the data to only include Falcon 9 launches
- Task three:
  - Dealing with missing values
  - Calculate the mean for PayloadMass using .mean()
  - Use the mean and .replace() function to replace np.nan values in the data

```
static_json_url='https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/I

response.status_code

data = pd.json_normalize(response.json())
```

```
data_falcon9 = df[df['BoosterVersion']!='Falcon 1']

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

```
# Calculate the mean value of PayloadMass column
payloadmassavg = data_falcon9['PayloadMass'].mean()
# Replace the np.nan values with its mean value
data_falcon9['PayloadMass'].replace(np.nan, payloadmassavg, inplace=True)
```

Data Collection via Web Scraping

- Task one:
  - Request the falcon 9 launch wiki page from its URL
- Task two:
  - Extract all column/variable names from HTML table header
- Task three:
  - Create a data frame by parsing the launch HTML tables

```
static_url = "https://en.wikipedia.org/wiki/List_of_Falcon_9_and_Falcon_Heavy_launches"

data  = requests.get(static_url).text

soup = BeautifulSoup(data, "html.parser")
```

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

column_names = []

for row in first_launch_table.find_all('th'):
    name = extract_column_from_header(row)
    if (name != None and len(name) > 0):
        column_names.append(name)
```

```
launch_dict= dict.fromkeys(column_names)

df=pd.DataFrame(launch_dict)
```

Data Wrangling

Data wrangling is the overall process of transforming raw data into a more usable form. Our purpose for wrangling was to extract the following data

- Task one :
  - Calculate the number of launches on each site
- Task two :
  - Calculate the number and occurrence of each orbit
- Task three :
  - Calculate the number and occurrence of mission outcome of the orbits
- Task four :
  - Create a landing outcome label from Outcome column

```
df['LaunchSite'].value_counts()
```

```
df['Orbit'].value_counts()
```

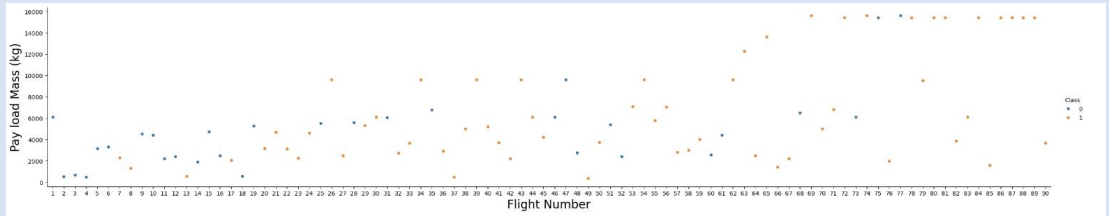
```
landing_outcomes = df['Outcome'].value_counts()
landing_outcomes
```

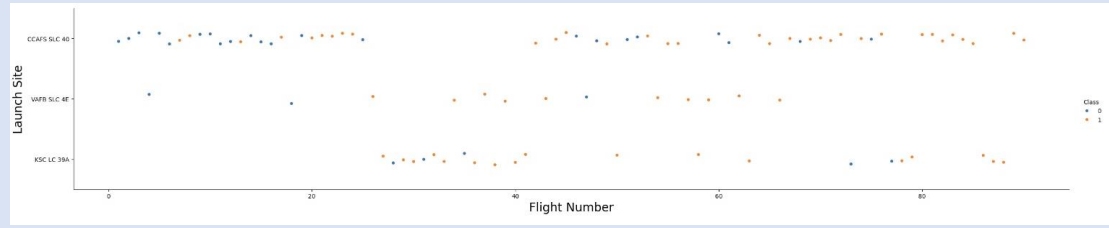
```
landing_class = []
for outcome in df['Outcome']:
    if outcome in bad_outcomes:
        landing_class.append(0)
    else:
        landing_class.append(1)
```

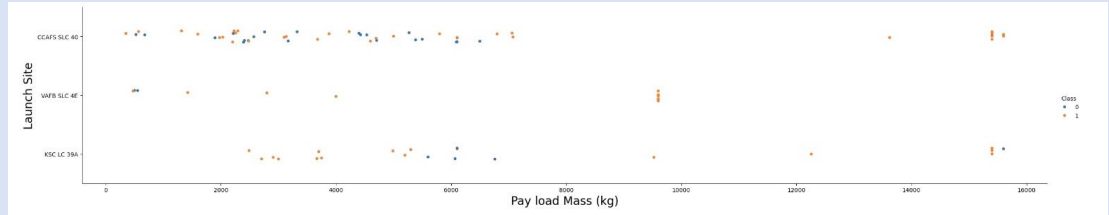
EDA with Data Visualization

Next we performed Data Analysis and Feature Engineering using Matplotlib and Pandas

- First, we created several scatter plots to visualize the relationships between different attributes, including :
  - Flight Number vs Payload Mass
  - Flight Number vs Launch Site
  - Launch Site vs Payload Mass



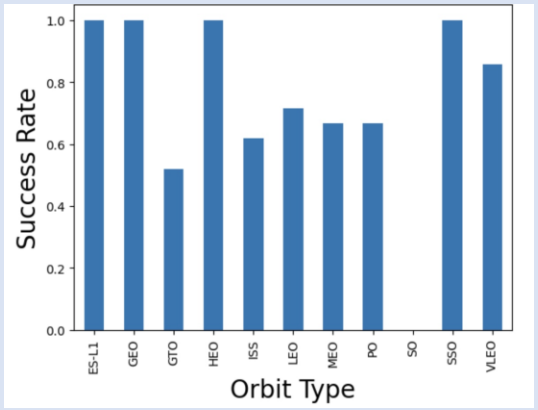


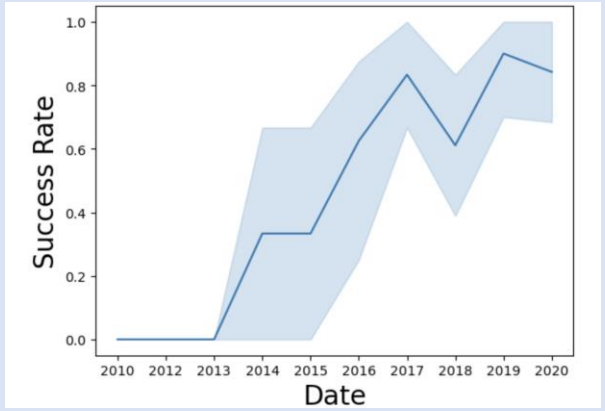


EDA with Data Visualization (cont)

- A scatter plot is crucial in identifying a possible relationship between changes observed in two sets of variables. Following the scatter plots, we used bar graphs and line graphs to further explore those relationships

- We use the bar graph to determine which orbits have the highest possibility of success
- We then use the line graph to show to show this possibility of success over time
- Next, we created dummy variables to represent categorical columns with feature engineering






```
features_one_hot = pd.get_dummies(features, columns = ['Orbit', 'LaunchSite', 'LandingPad', 'Serial'])
features_one_hot.head()
```

Exploratory Data Analysis using SQL

- In this step, we performed a series of SQL queries to better understand the SpaceX Data Set. Examples of these queries are provided below

- Display the total Payload Mass Carried by booster launched by NASA (CRS)
- List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000
- List the names of the booster versions which have carried the maximum payload mass

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

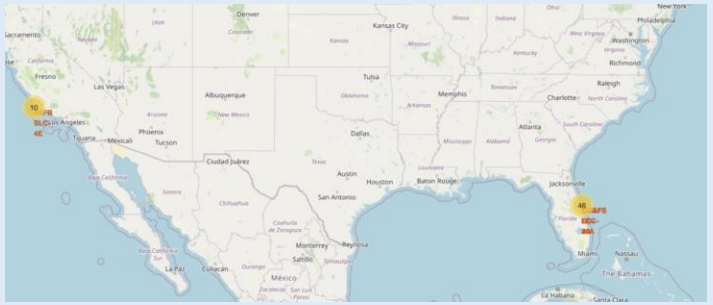
```
%%sql
SELECT BOOSTER_VERSION
FROM SPACEXTBL
WHERE LANDING_OUTCOME = 'Success (drone ship)'
    AND 4000 < PAYLOAD_MASS__KG_ < 6000;
```

```
%%sql
SELECT DISTINCT BOOSTER_VERSION
FROM SPACEXTBL
WHERE PAYLOAD_MASS__KG_ = (
    SELECT MAX(PAYLOAD_MASS__KG_)
    FROM SPACEXTBL);
```

Launch Sites Locations Analysis with Folium

Next we explored how the launch success rate may depend on locations and proximities of a launch site by performing interactive visual analytics with Folium

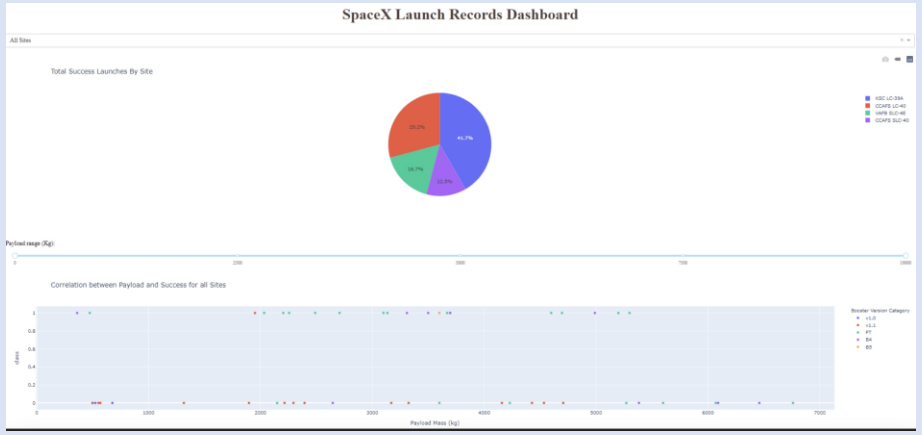
- Task One
  - Mark all launch sites on a map
- Task Two
  - Mark all success/failed launches for each site on a map
- Task Three
  - Calculate the distance between a launch site to its proximities



Build a Dashboard with Plotly Dash

Using Plotly Dash, we built an interactive dashboard to analyze launch records with pie charts and scatter plots

- Task One
  - Add a launch site Drop-Down Input Component
- Task Two
  - Add a callback function to render success pie charts based on selected site dropdown
- Task Three
  - Add a Range Slider to Select Payload
- Task Four
  - Add a callback function to render the success payload scatter chart



Predictive Analysis : Classification Model

- First, we performed EDA and determined the Training Labels
  - Created a column for class
  - Standardized the data
  - Split into training data and test data
- Next we found the best Hyperparameter for SVM, Classification Trees and Logistic regression to determine the method that performs best using the data

```
Y = data['Class'].to_numpy()
Y
```

```
transform = preprocessing.StandardScaler()

X = transform.fit_transform(X)
```

```
X_train, X_test, Y_train, Y_test = train_test_split( X, Y, test_size=0.2, random_state=2)
print ('Train set:', X_train.shape,  Y_train.shape)
print ('Test set:', X_test.shape,  Y_test.shape)
```

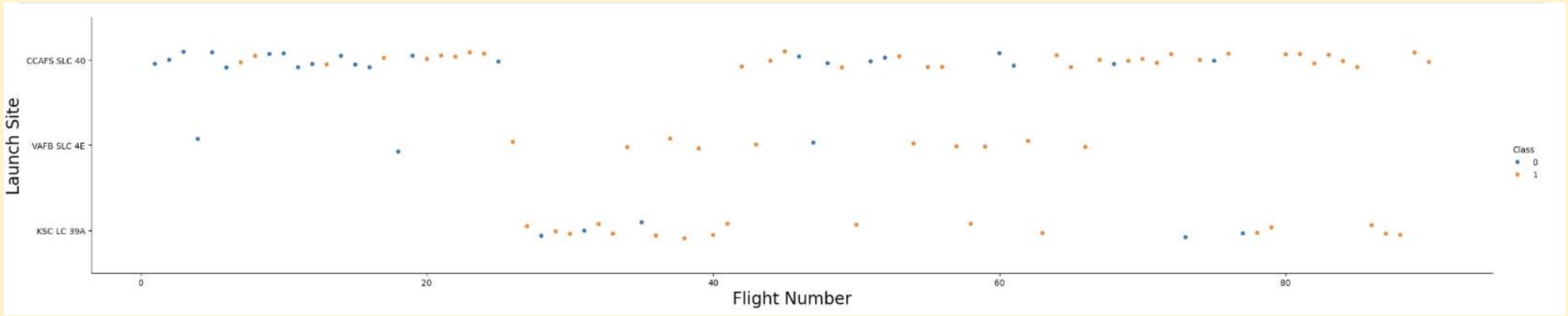
```
parameters = {'criterion': ['gini', 'entropy'],
     'splitter': ['best', 'random'],
     'max_depth': [2*n for n in range(1,10)],
     'max_features': ['auto', 'sqrt'],
     'min_samples_leaf': [1, 2, 4],
     'min_samples_split': [2, 5, 10]}

tree = DecisionTreeClassifier()
```

```
tree_cv = GridSearchCV(tree,parameters,cv=10)
tree_cv.fit(X_train, Y_train)
```

Results

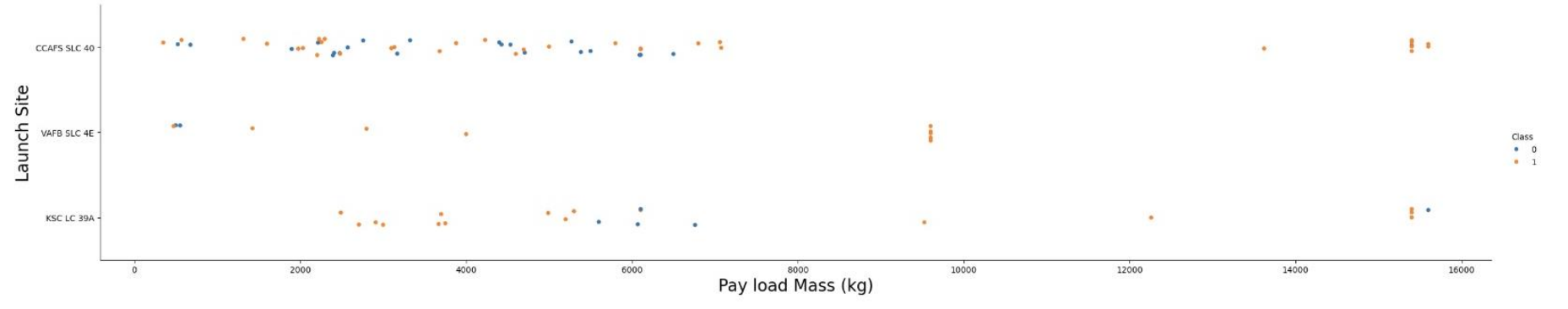
(1) Exploratory Data Analysis Results with Visualizations
(2) Exploratory Data Analysis Results using SQL
(3) Interactive Results with Folium and Plotly Dash
(4) Predictive Analysis Results

(1) Flight Number vs Launch Site



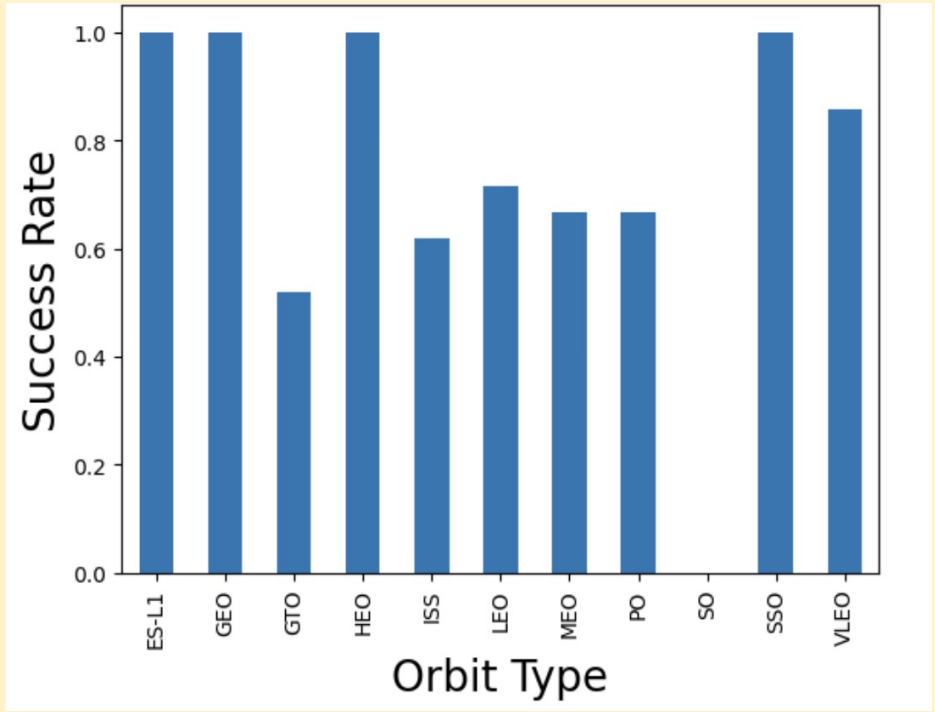
Analyzing this scatter plot we can deduce that that as the Flight Number increases, so does the Success Rate

(1) Payload Mass vs Launch Site



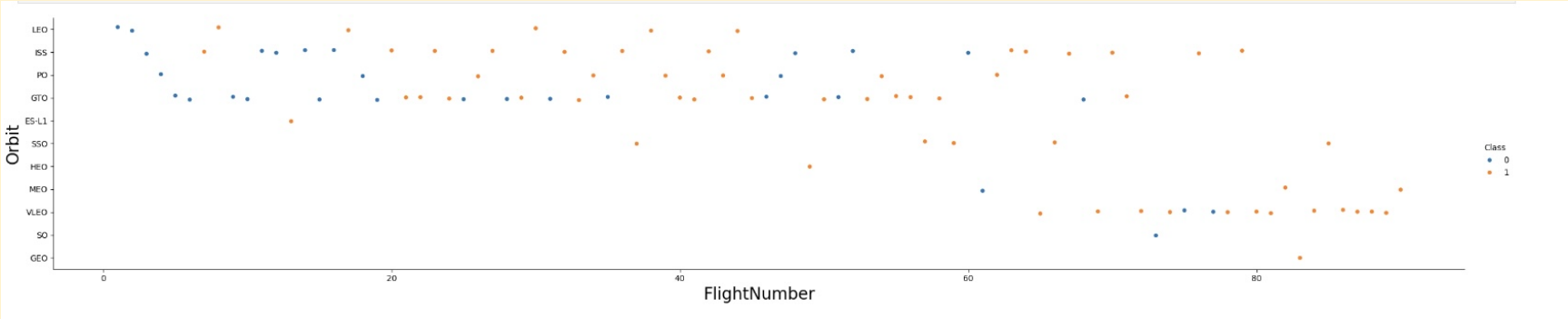
This scatter plot shows several things. First, there are no launches heavier than 16000 kg. Second, launches over 7000 kg have a higher chance of success.

(1) Success Rate vs Orbit Type



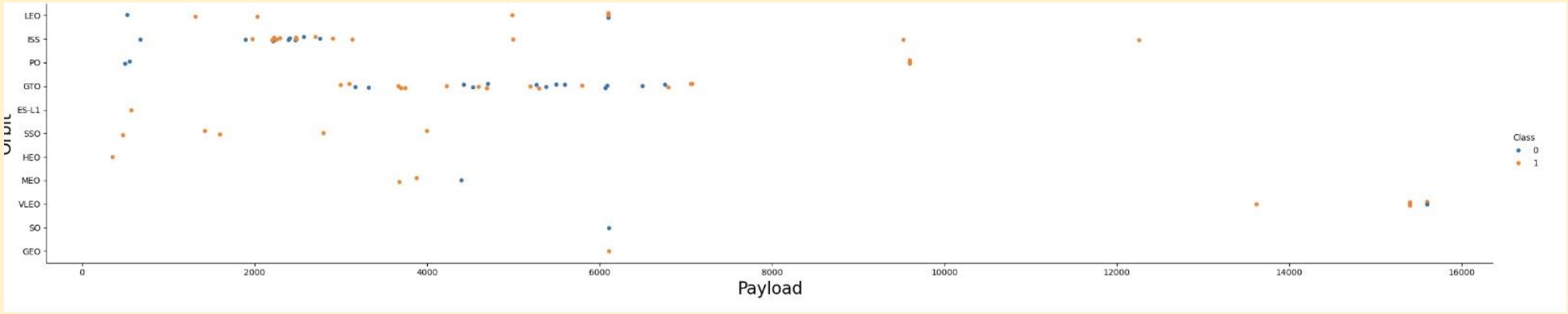
- This bar chart visualizes the relationship between success rate and orbit type
- ES-L1, GEO, HEO, SSO and VLEO all have a 100% success rate
- Although SO has a success rate of 0%, there was only one occurrence
- More data would be needed to draw an accurate conclusion

(1) Flight Number vs Orbit Type



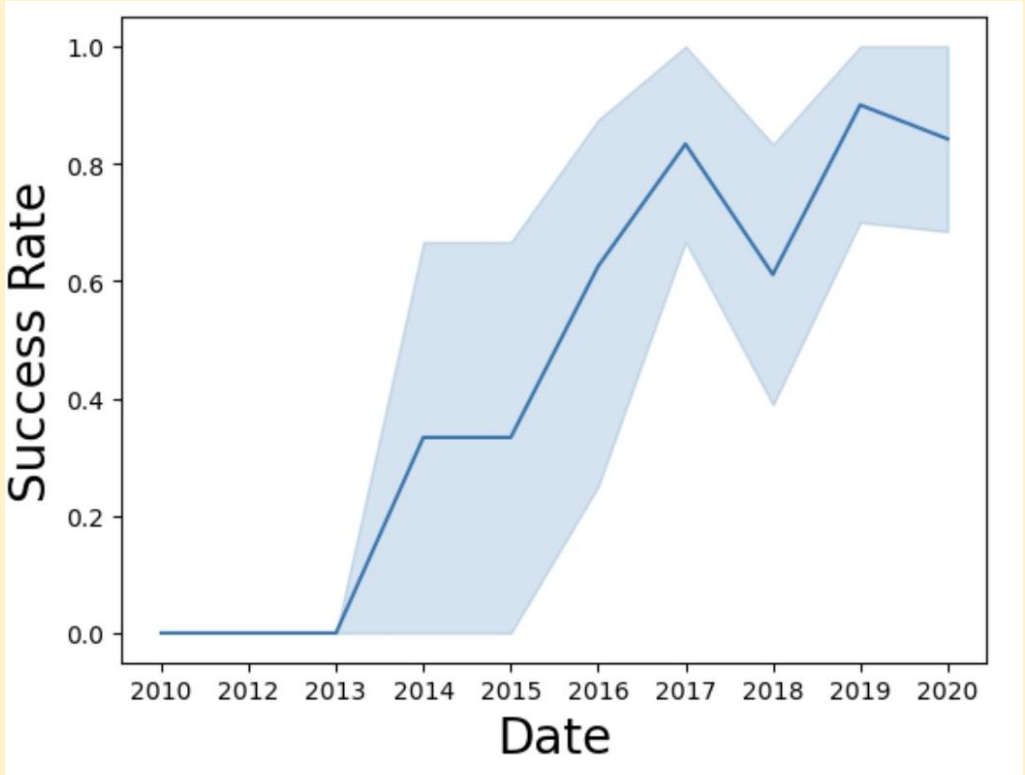
- In the LEO orbit, it seems a higher flight number is related to a higher success rate
- For GTO orbit, it seems that there is no relation
- Similar to the previous slide, orbits with only one occurance do not have enough data for an insightful conclusion

(1) Payload Mass vs Orbit Type



- There is a positive relationship between higher payload mass and success rate for LEO, ISS and PO
- There is a negative relationship between higher payload mass and success rate for MEO, and VLEO
- There seems to be no relationship between higher payload mass and success rate for GTO

(1) Success Rate vs Date



- This graph portrays the relationship between success rate and date of lauch
- We can easily say that as time passed, the success rate of all laucnhes steadily increased between the years 2013 – 2020
- We can assume that as more time passes, the success rate will continue to rise

(2) EDA with SQL Results

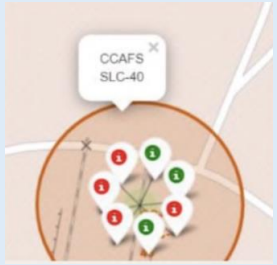
- List of Distinct Launch Site Names
  - CCAFS LC-40, VAFB SLC-4E, KSC LC-39A, CCAFS SLC-40
- 5 Launch Sites that begin with the string 'CCA'
  - CCAFS LC-40 was listed 5 times
- Total Payload Mass carried by boosters launched by NASA (CRS)
  - 45596
- Average Payload Mass carried by booster F9 v1.1
  - 340.4
- List the date when the first successful landing outcome in ground pad was achieved
  - 2015-12-22
- List the names of the boosters which have success in drone ship and have a payload mass greater than 4000 but less than 6000
  - F9 FT B1021.1, F9 FT B1022, F9 FT B1023.1, F9 FT B1026, F9 FT B1029.1, F9 FT B1021.2, F9 FT B1029.2, F9 FT B1036.1, F9 FT B1038.1, F9 B4 B1041.1, F9 FT B1031.2, F9 B4 B1042.1, F9 B4 B1045.1 F9 B5 B1046.1

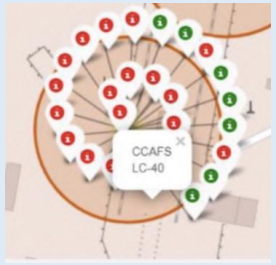
(2) EDA with SQL Results continued

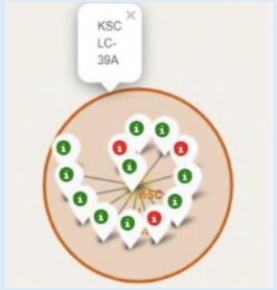
- List the total number of successful and failure mission outcomes
  - Failure in flight (1), success (100)
- List the names of the booster versions which have carried the maximum payload mass
  - F9 B5 B1048.4, F9 B5 B1049.4, F9 B5 B1051.3, F9 B5 B1056.4, F9 B5 B1048.5, F9 B5 B1051.4, F9 B5 B1049.5, F9 B5 B1060.2, F9 B5 B1058.3, F9 B5 B1051.6, F9 B5 B1060.3, F9 B5 B1049.7
- List the records which will display month names, failure landing outcome in drone ship, booster versions and launch site for the months in the year 2015
  - [failure drone ship, F9 v1.1 B1012, CCAFS LC-40, month 01], [failure drone ship, F9 v1.1 B1015, CCAFS LC-40, month 04]
- Rank the count of landing outcomes or success between 2010-06-04 and 2017-03-20 in descending order
  - No attempt (10), success drone ship (5), failure drone ship (5), success ground pad (3), controlled ocean (3), uncontrolled ocean (2), failure parachute (2), precluded drone ship (1)

(3) Interactive Results: Folium

- Here, we see that all launch site are within the United States



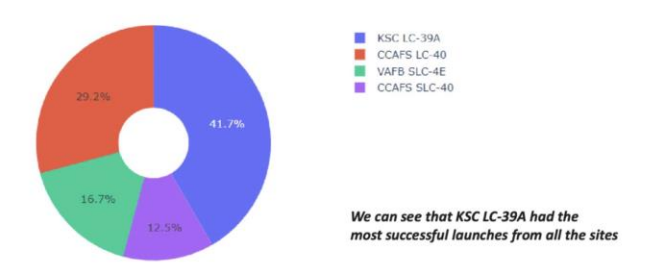




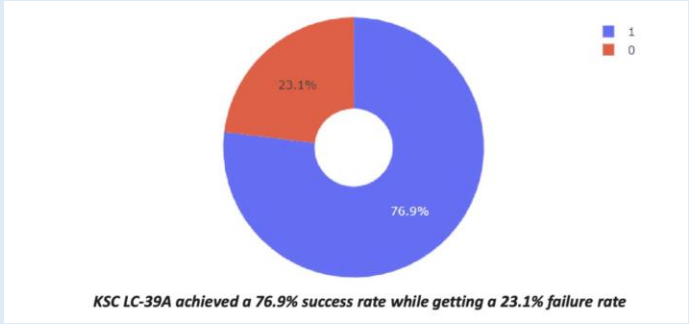
- Here, the red markers shows launches than failed, while green markers shows the successful launches

- After finding the distance to different land marks be found:
  - Launch sites avoid being close to railways, highways, and cities
  - Launch sites are usually close to coast lines, which makes sense that all the launch sites are in California and Florida

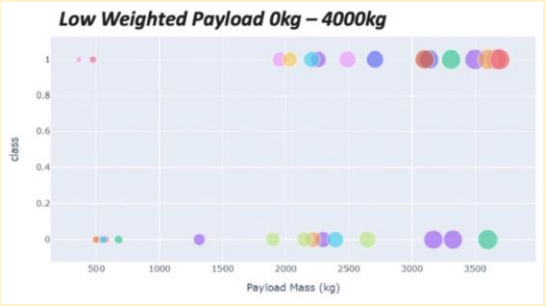
(3) Interactive Results : Plotly Dash

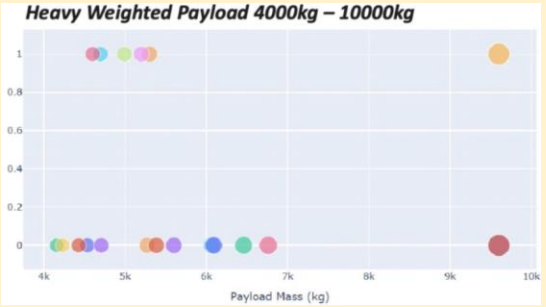


- Here we can see the success rate for each site



- Here, we see than KSC LG-39A has the highest launch success rate





- Here, we see than a lower payload has a higher success rate

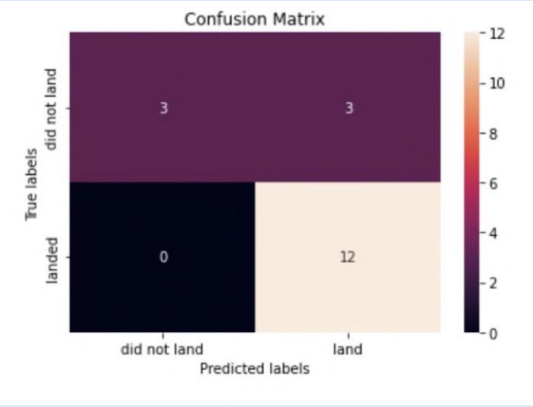
(4) Predictive Analysis Results

- After, finding the hyperparameter for SVM, Classification Tree and Logistic Regression Models, we determined that the Classification Tree Model was the most accurate with an accuracy of .875
- Analyzing the Confusion Matrix we find
  - We have 0 type 2 errors, ie predicting the rocket did not land when it landed
  - We have only 3 type 1 errors, ie predicting the rocket landed when it did not land
  - We are the most successful (12) in predicting the rocket landed when it did in fact land

```
parameters = {'criterion': ['gini', 'entropy'],
     'splitter': ['best', 'random'],
     'max_depth': [2*n for n in range(1,10)],
     'max_features': ['auto', 'sqrt'],
     'min_samples_leaf': [1, 2, 4],
     'min_samples_split': [2, 5, 10]}

tree = DecisionTreeClassifier()
```

```
tree_cv = GridSearchCV(tree,parameters,cv=10)
tree_cv.fit(X_train, Y_train)
```

```
tuned hpyerparameters :(best parameters)  {'criterion': 'entropy', 'max_depth': 6, 'max_features': 'sqrt', 'min_samples_leaf': 1, 'min_samples_split': 2, 'splitter': 'best'}
accuracy : 0.875
```



Conclusion

Throughout the course of our analysis on SpaceX flight launch data, we have discovered a plethera of interesting insights. With the help of our interactive dash boards and machine learning algorithms we have concluded the following to be of the most significance:

- The Classification Tree Machine Learning Model is the most accurate method for predicting flight launch success with an accuracy of .875
- The SSO orbit has the highest success rate with a perfect 100%
- The KSC LC-39A Launch Site has the highest amount of successful launches with nearly 80%
- Payloads with a lighter weight had a higher chance of success
- Between the years 2013 and 2020 the overall rate of success has continued to increase, most likely due to a steady increase trial and error correction of launch technology by Space X

Thank you for joining Space Y on this rocket launch exploration journey. After thorough analysis we may now have a greater understanding on the success of Space X and their Falcon 9 rocket.