IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Thomas Seive
05-03-2022

Outline

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

Executive Summary

Methodologies

- Need to out bid competitors
- Predict unsuccessful rocket launches
- Used SpaceX API and webscraping to collect data
- One hot encoded categorical values and identified null values
- Plotted features geospatially and in dashboard
- Ran cv = 10 model tests

Results

- Decision tree is most effective model
- Launches have become more successful over time
- Most successful launch site has sub-50% success rate

Introduction

- SpaceY has much to offer to space industry in terms of launches
- SpaceX offers launches at fraction of price
- We can outbid if we think the rocket will not success for reuse
- What features of SpaceX launches help us predict the landing success?
- Will our predictions need to change over time?

Section 1

Methodology

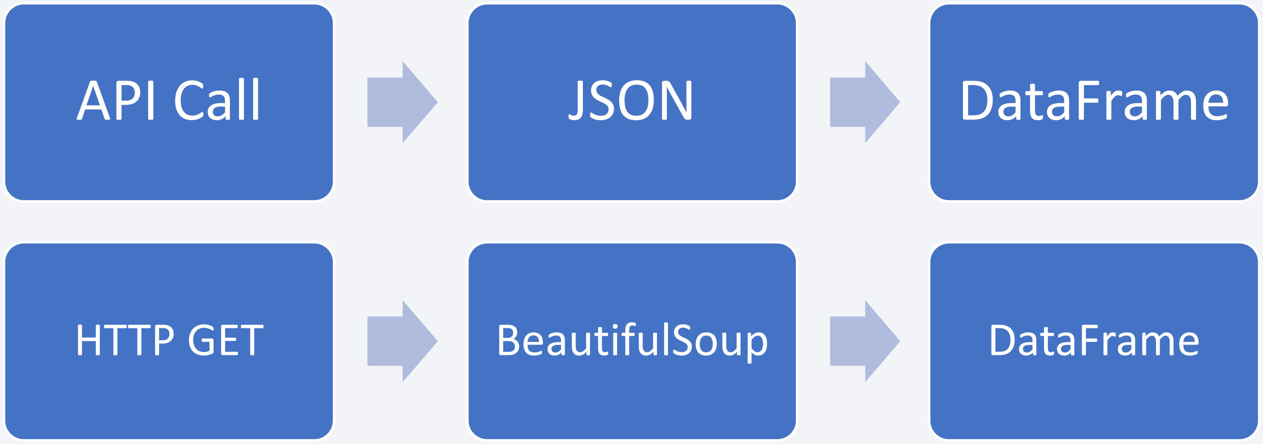
Methodology

Executive Summary

- Data collection methodology:
  - SpaceX API calls and webscraping of SpaceX related wiki HTML tables
- Perform data wrangling
  - Check for null values and consistent data types
  - One hot encoding of categorical data
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Test 4 models (Logistic Regression, SVM, Decision Tree, K Nearest Neighbors)
  - Apply Grid Search to optimize model hyperparameters

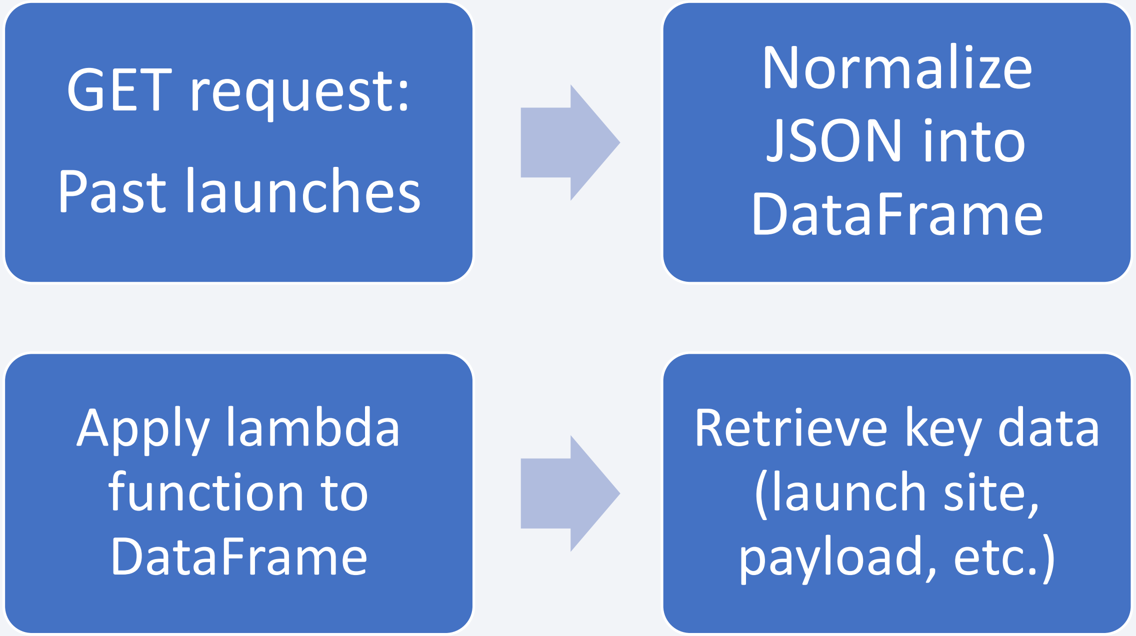
Data Collection

- Data was acquired using a combination of API calls and webscraping
  - Used the SpaceX API
  - Scraped the Wikipedia page for SpaceX launches
    - Retrieved specific rows from one table
  - Transformed data into DataFrame and a CSV (for reuse throughout project)



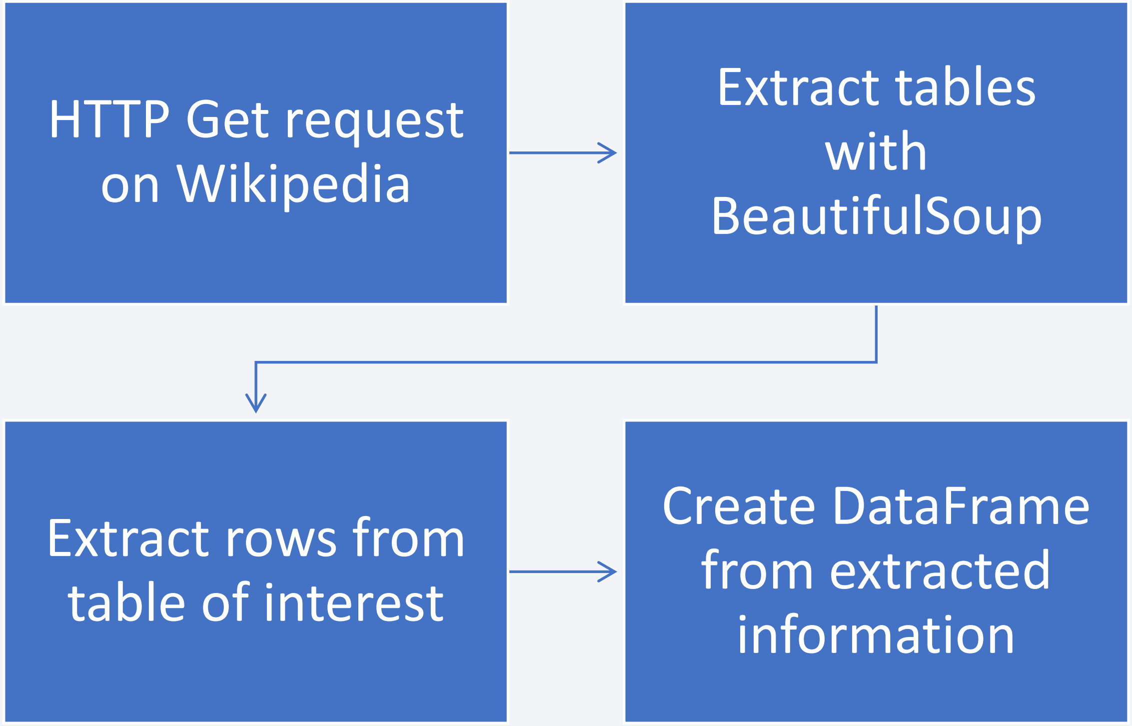
Data Collection – SpaceX API

- Use GET API call on past launches to get information directly from SpaceX API
- JSON is messy so we extract what we need: Date, launch site, payload, etc.
- Notebook for peer-review



Data Collection - Scraping

- Retrieve all potential information with GET HTTP request on Wikipedia
- Use BeautifulSoup to extract intended table
- Use loop to extract each row, “tr”, and create DataFrame
- Notebook for peer-review



Data Wrangling

- Read in **CSV** from our API calls and webscraping
- Data is contained in DataFrame format to allow for quick analysis:
  - Analyze **null values**
  - Verify **data types**
  - Perform **counts** for different launch sites
  - **One hot encoding** to make categorical variables useable in analysis
- Notebook for peer-review

EDA with Data Visualization

- To better understand the relations between variables, we used plots to visually inspect the relation between features
- A **scatter** plot was used to see the clustering and distribution of data: Flight Number vs Payload mass/Launch Site/Orbit type; Payload mass vs Launch Site/Orbit type;
- The **line** plot was used to study a trend against time: Year vs Success rate
- We used a **bar** plot to study how different categories compared numerically: Orbit type vs Average success rate
- Notebook for peer-review

EDA with SQL

- To get precise answers, used SQL queries ranging from:
  - Summing the mass of payloads launched for specific customers
  - Finding the boosters that can carry the maximum payload mass
  - Determining the unique launch sites
- Notebook for peer-review

Build an Interactive Map with Folium

- Mapped the main launch sites as **circles** to make their location clear
- Added **marker clusters** with color coding to indicate successful launches and reduce clutter from concentrated launch sites
- Drew **lines** to areas of interest (closest coast, city, railway, etc) to better understand the importance of these features to launch success
- Notebook for peer-review

Build a Dashboard with Plotly Dash

- Dropdown to select specific launch sites for flexibility
- Pie chart to see the percentage of launches from each launch site to understand which sites have the most use
- Slider to select the payload mass of interest for precise exploration
- Scatter plot to compare payload mass and success of the launch
- Notebook for peer-review

Predictive Analysis (Classification)

- Split data into **training and testing** sets (80/20) and **normalized** the data
- Compared 4 models (**Logistic Regression**, **SVM**, **Decision Tree**, **K Nearest Neighbors**)
- Used **Grid Search** to optimize the hyperparameters of each model
- Used a value of 10 for the **folds** on each model
- Compared the **scores** to find best model (KNN, SVM, and LR preformed very similarly)
- Notebook for peer-review

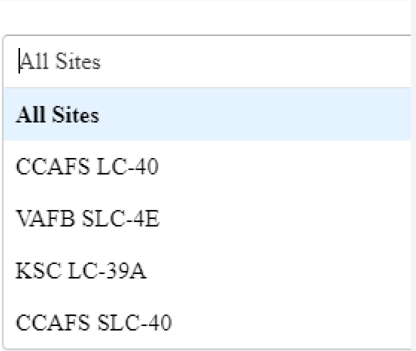
Results

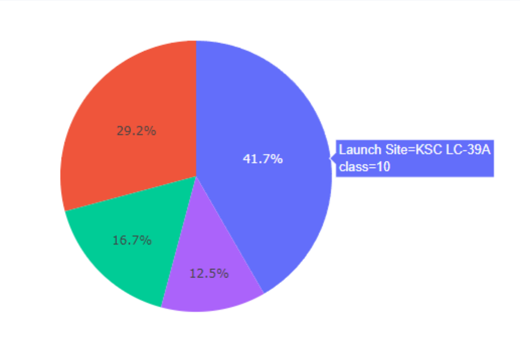
Exploratory data analysis results

```
Outcome
True ASDS      41
None None      19
True RTLS      14
False ASDS      6
True Ocean      5
False Ocean     2
None ASDS       2
False RTLS      1
dtype: int64
```

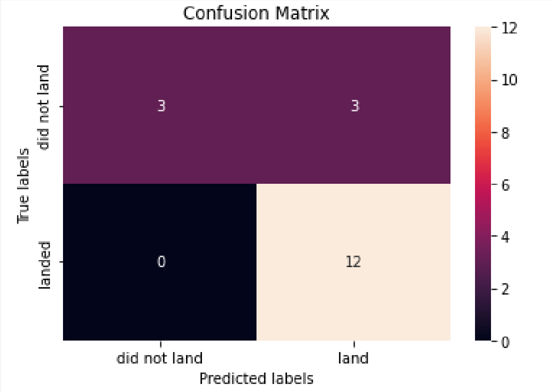
```
LaunchSite
CCAFS SLC 40    55
KSC LC 39A      22
VAFB SLC 4E     13
dtype: int64
```

Interactive analytics demo in screenshots





Predictive analysis results

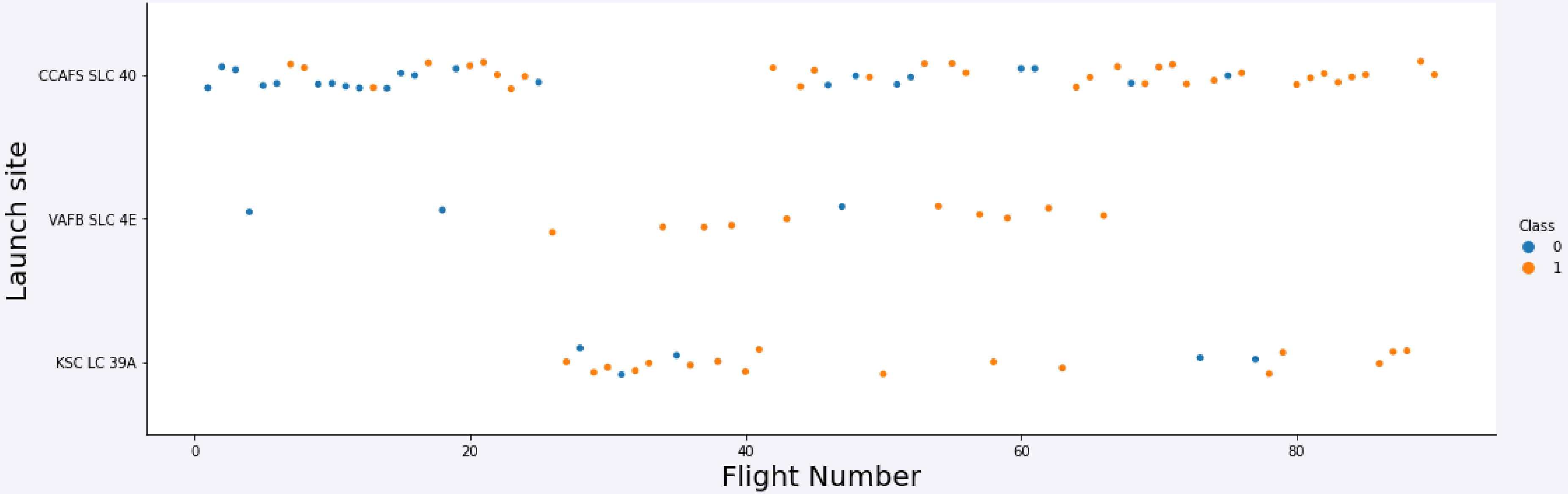



```
tuned hyperparameters SVM :(best parameters)  {'C': 1.0, 'gamma': 0.03162277660168379, 'kernel': 'sigmoid'}
accuracy : 0.8482142857142856
```

Section 2

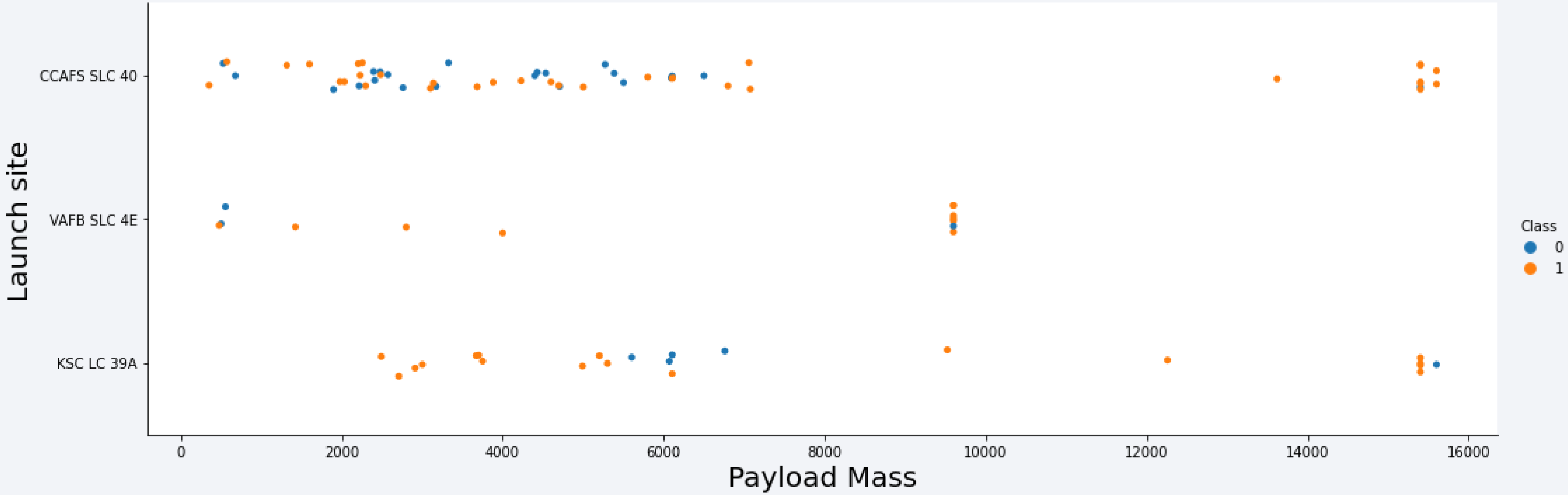
Insights drawn from EDA

Flight Number vs. Launch Site



- Scatter plot of Launch Site vs Flight Number
- Class 0 (blue) = failure
- Class 1 (orange) = success
- More successes with higher flight number
- Each launch site has same relative proportion of success to failure

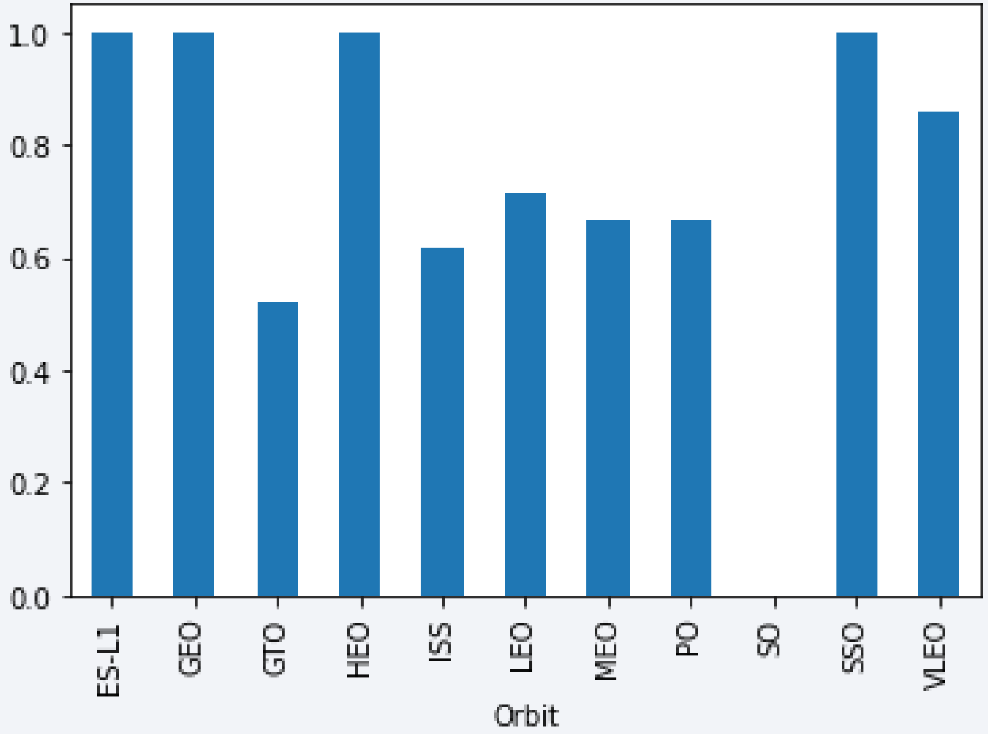
Payload vs. Launch Site



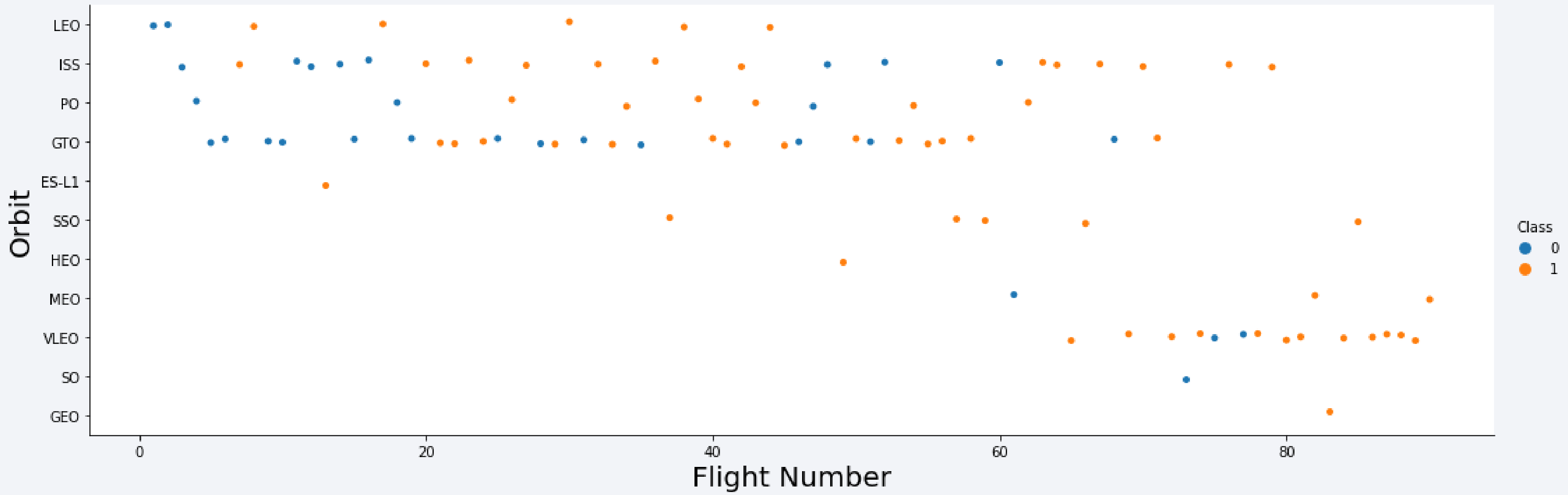
- Scatter plot of Payload vs. Launch Site
- Higher payload masses have a higher success rate.
- CCAFS SLC 40 has most launches
- VAFB SLC 4E has a payload mass limit

Success Rate vs. Orbit Type

- Bar chart for the average success rate of each orbit type
- GTO has the lowest success rate for rocket reuse
- SSO, GEO, HEO, and ES-L1 all have 100% success

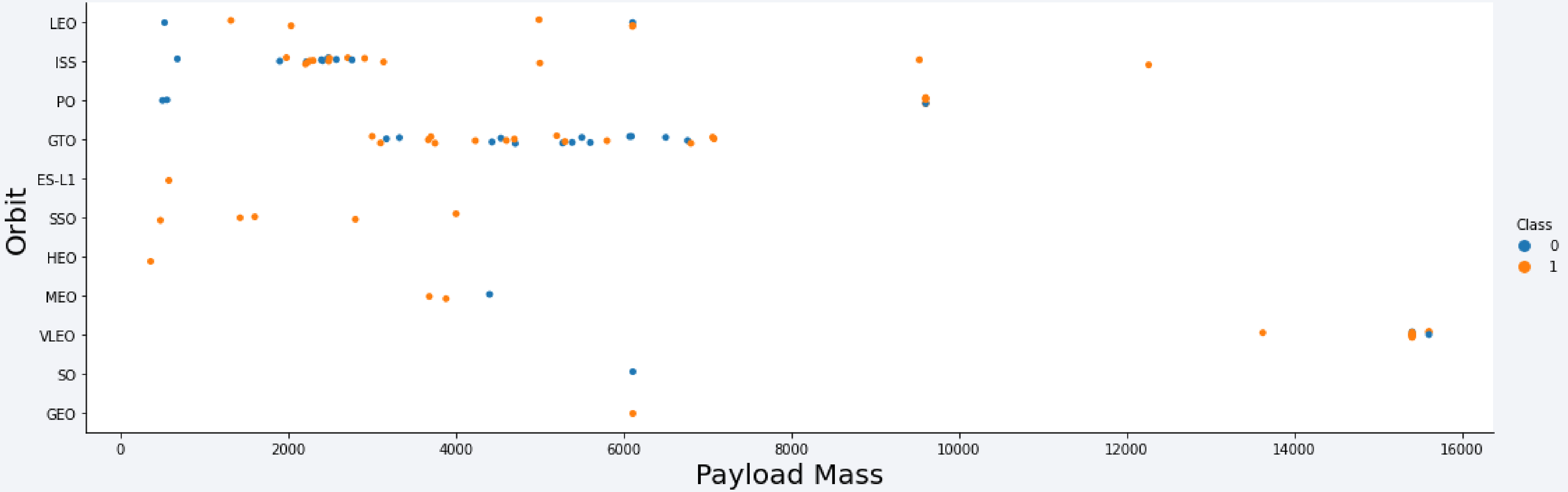


Flight Number vs. Orbit Type



- Scatter plot of Flight number vs. Orbit type
- No obvious correlation beyond higher flight numbers going to new orbit types

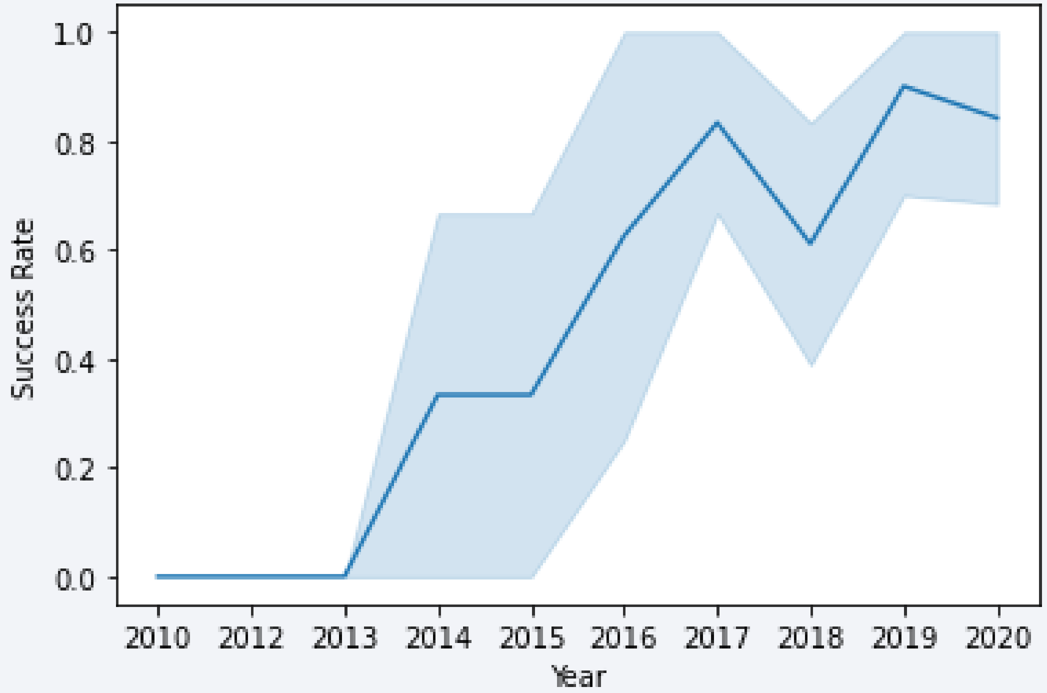
Payload vs. Orbit Type



- Scatter plot of Payload mass vs Orbit type
- Successful and unsuccessful missions are close together, meaning there is little correlation

Launch Success Yearly Trend

- A line chart of yearly average success rate
- The shaded region indicates the upper and lower bounds of the success rate for a specific year
- Clear trend up with a mild dip in recent years



All Launch Site Names

- In order to find all the unique launch site names, I used the **Distinct** SQL command. There are only 4 unique sites used by SpaceX
- Result:

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

Launch Site Names Begin with 'KSC'

- In order to find 5 records where launch sites' names start with `KSC`, I used the **like** SQL function. All of these records were successful launches in 2017.
- Result:

DATE	time__utc_	booster_version	launch_site	payload	payload_mass__kg_	orbit	customer	mission_outcome	landing__outcome
2017-02-19	14:39:00	F9 FT B1031.1	KSC LC-39A	SpaceX CRS-10	2490	LEO (ISS)	NASA (CRS)	Success	Success (ground pad)
2017-03-16	06:00:00	F9 FT B1030	KSC LC-39A	EchoStar 23	5600	GTO	EchoStar	Success	No attempt
2017-03-30	22:27:00	F9 FT B1021.2	KSC LC-39A	SES-10	5300	GTO	SES	Success	Success (drone ship)
2017-05-01	11:15:00	F9 FT B1032.1	KSC LC-39A	NROL-76	5300	LEO	NRO	Success	Success (ground pad)
2017-05-15	23:21:00	F9 FT B1034	KSC LC-39A	Inmarsat-5 F4	6070	GTO	Inmarsat	Success	No attempt

Total Payload Mass

- In order to calculate the total payload carried by boosters from NASA, I used the **Sum** and **Where** SQL functions. I found that NASA is an important customer for heavy SpaceX launches and has had 45,596 kg in payloads launched.
- Result:

1
45596

Average Payload Mass by F9 v1.1

- In order to calculate the average payload mass carried by booster version F9 v1.1, I used the **AVG** and **Where** SQL functions. I found that F9 v1.1 carries an average payload of 2,928 kg.
- Result:

1
2928

First Successful Ground Landing Date

- To find the dates of the first successful landing outcome on ground pad, I used the **min** SQL command. The first successful ground pad landing was in December 2015 which was a few months before the first successful drone ship landing.
- Result:

1
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

- To list the names of boosters which have successfully landed on drone ship and had payload mass greater than 4000 but less than 6000, I used **Where** and **Between** … **And**. In this payload range, we see that SpaceX employs the F9 FT.
- Results:

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

Total Number of Successful and Failure Mission Outcomes

- In order to calculate the total number of successful and failure mission outcomes I had to use a **subquery** along with the **as** command. I found that there has only ever been one mission failure!
- Result:

mission_failure	mission_success
1	99

Boosters Carried Maximum Payload

- To list the names of the booster which have carried the maximum payload mass, I used a subquery "**where** payload_mass__kg_ = (**select** **max**(payload_mass__kg_) **from** SPACEXTBL)". Noticeably, the F9 B5 is responsible for the heavy lifting.

Result:

booster_version
F9 B5 B1048.4
F9 B5 B1048.5
F9 B5 B1049.4
F9 B5 B1049.5
F9 B5 B1049.7
F9 B5 B1051.3
F9 B5 B1051.4
F9 B5 B1051.6
F9 B5 B1056.4
F9 B5 B1058.3
F9 B5 B1060.2
F9 B5 B1060.3

2015 Launch Records

- To list the records which will display the month names, successful landing outcomes in ground pad, booster versions, launch site for the months in year 2017, I used **MonthName**, **Where … And**. The launches were concentrated in one launch site and where a few months apart.
- Result:

1	landing__outcome	booster_version	launch_site
February	Success (ground pad)	F9 FT B1031.1	KSC LC-39A
May	Success (ground pad)	F9 FT B1032.1	KSC LC-39A
June	Success (ground pad)	F9 FT B1035.1	KSC LC-39A
August	Success (ground pad)	F9 B4 B1039.1	KSC LC-39A
September	Success (ground pad)	F9 B4 B1040.1	KSC LC-39A
December	Success (ground pad)	F9 FT B1035.2	CCAFS SLC-40

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

- In order to rank the count of successful landing outcomes between the date 2010-06-04 and 2017-03-20 in descending order, I used **Where**, **Between** … **And**, and **Order By**. The dates do not extend past December 2015 and most successes are with a drone ship.
- Result:

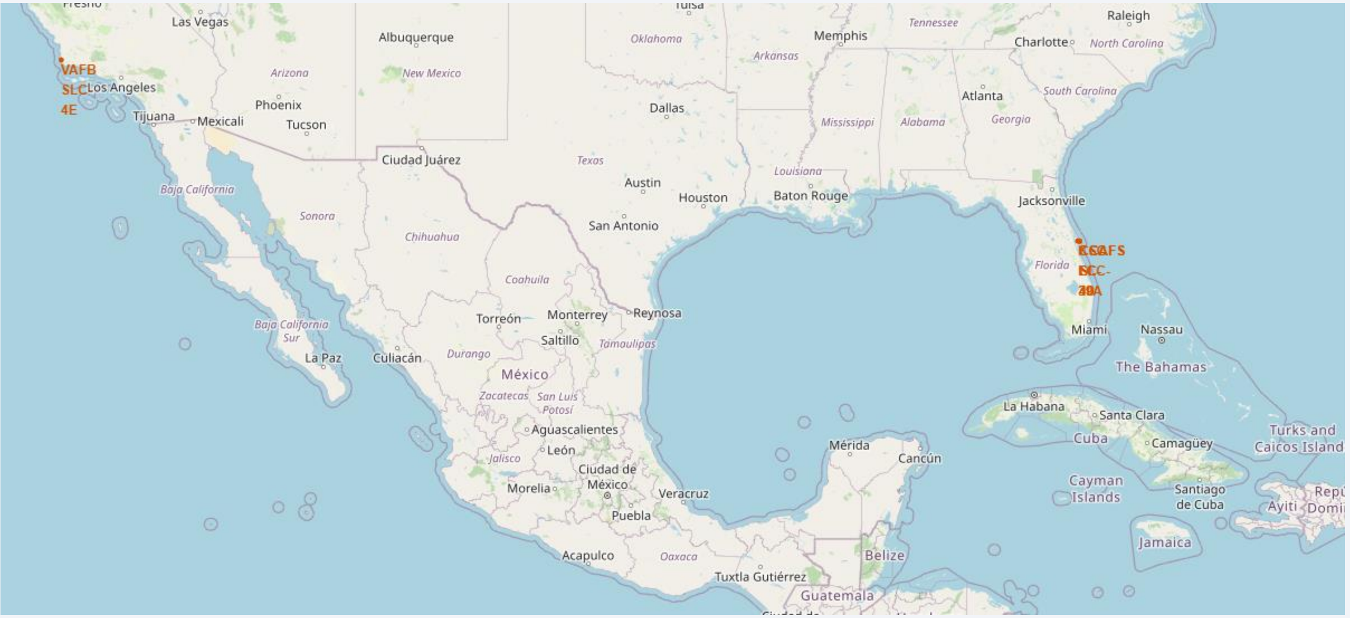
landing__outcome	DATE
Success (ground pad)	2017-02-19
Success (drone ship)	2017-01-14
Success (drone ship)	2016-08-14
Success (ground pad)	2016-07-18
Success (drone ship)	2016-05-27
Success (drone ship)	2016-05-06
Success (drone ship)	2016-04-08
Success (ground pad)	2015-12-22

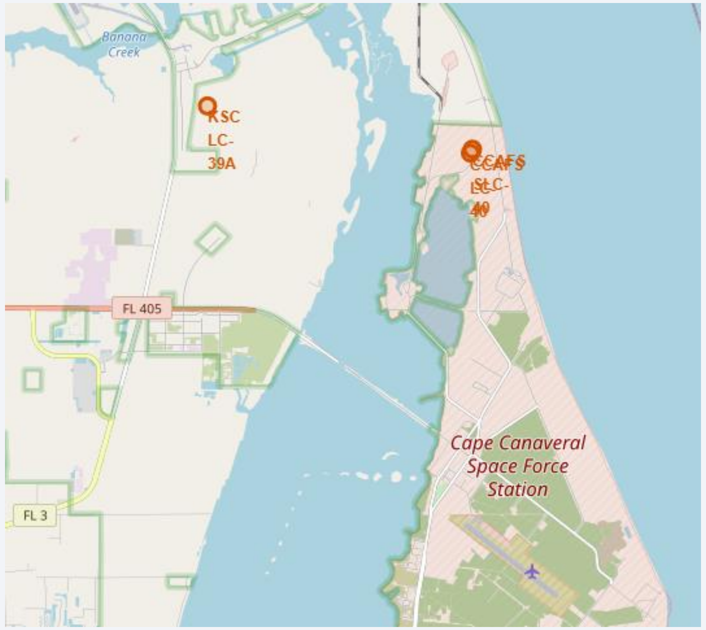
Section 3

Launch Sites Proximities Analysis

Launch Site Location Markers

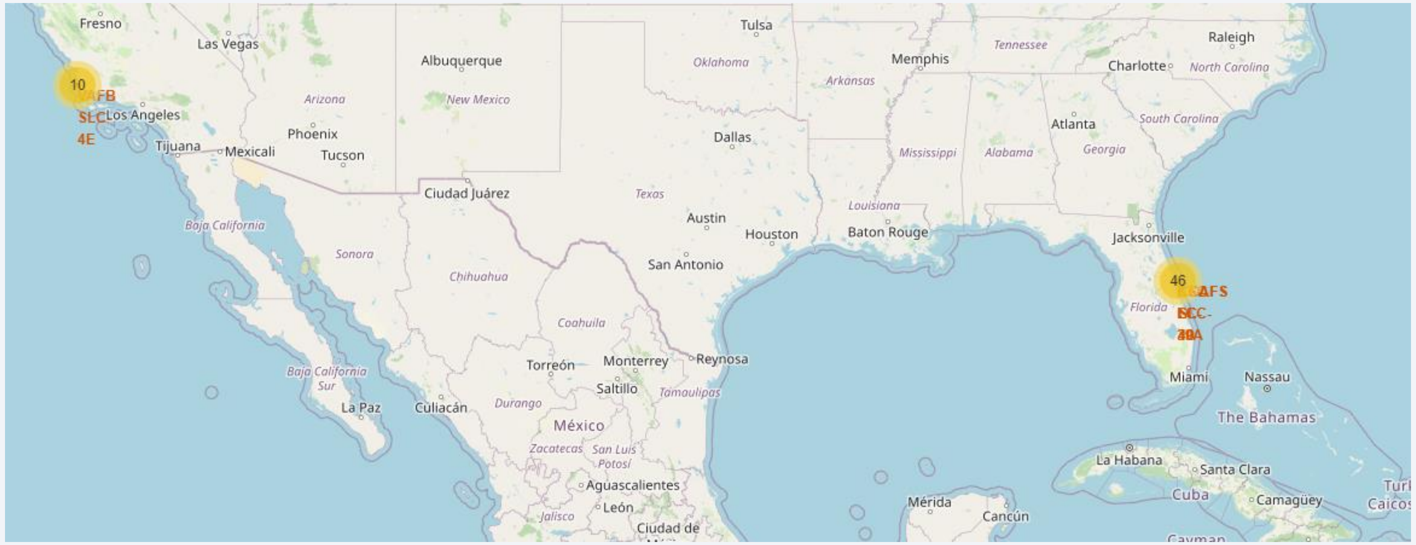
- The launch sites are represented by orange circles
- The launch sites are located on both the east and west coast of the United States
- A few of the sites are located close together

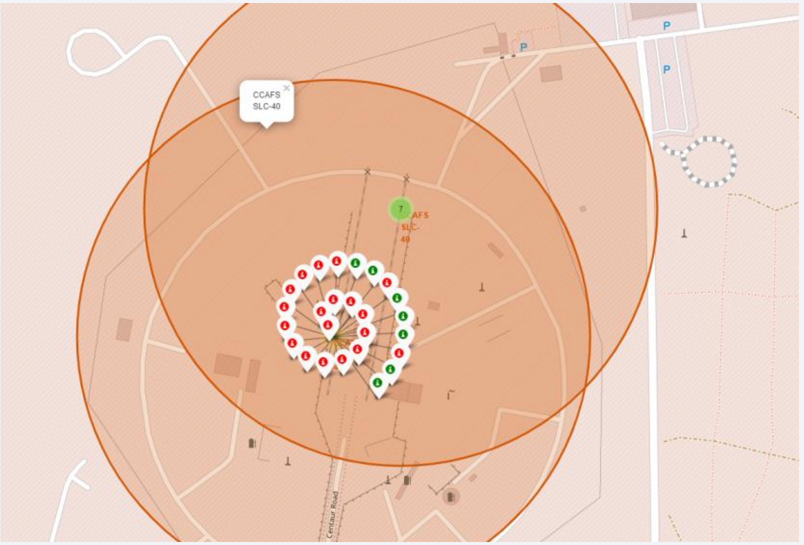




Successful Launch Marker Clusters

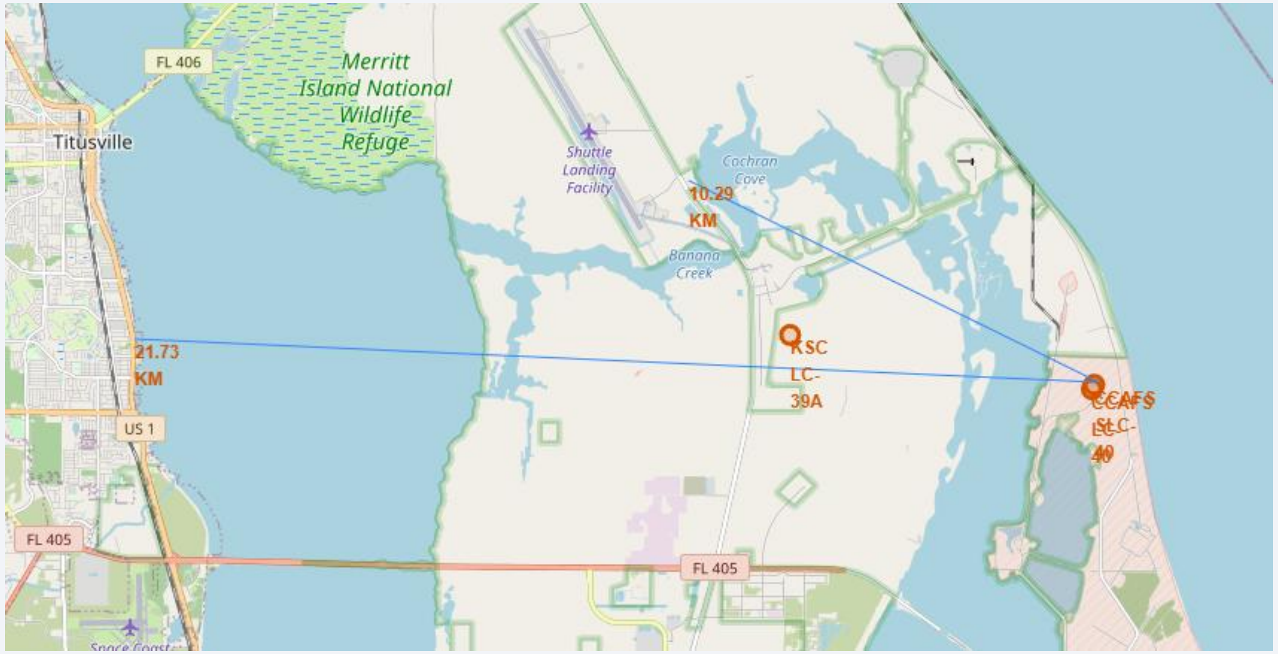
- Markers were placed in clusters to reduce clutter
- A green label indicates a successful launch
- Launches are concentrated on the east coast
- Some launch sites have a large concentration of unsuccessful launches





Launch Site Proximity to Infrastructure

- Displayed the distance to the closest highway/coast and the closest railway
- The east coast launch sites have twice the distance to the closest highway and city
- The launch sites are very close to railways

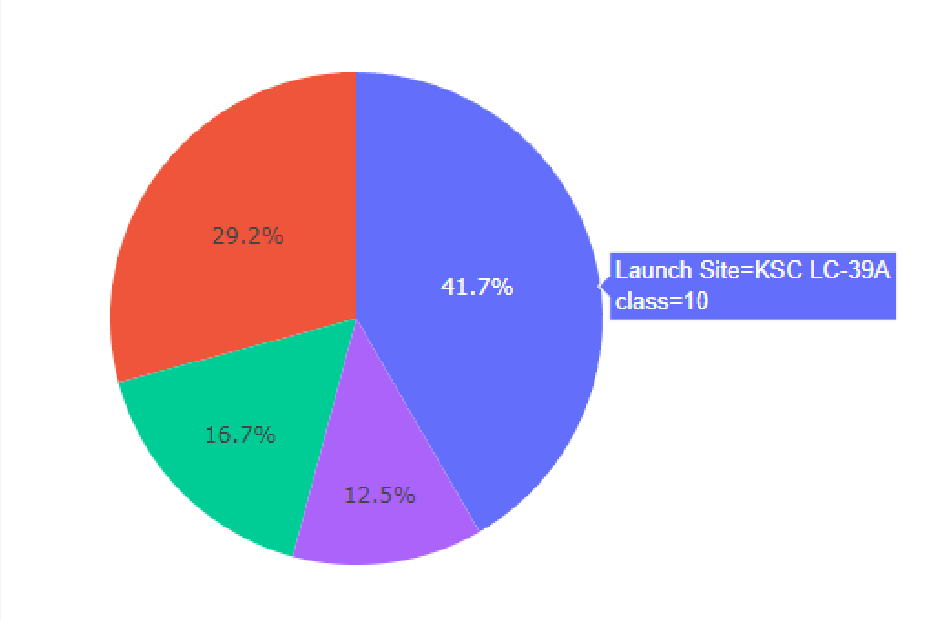


Section 4

Build a Dashboard with Plotly Dash

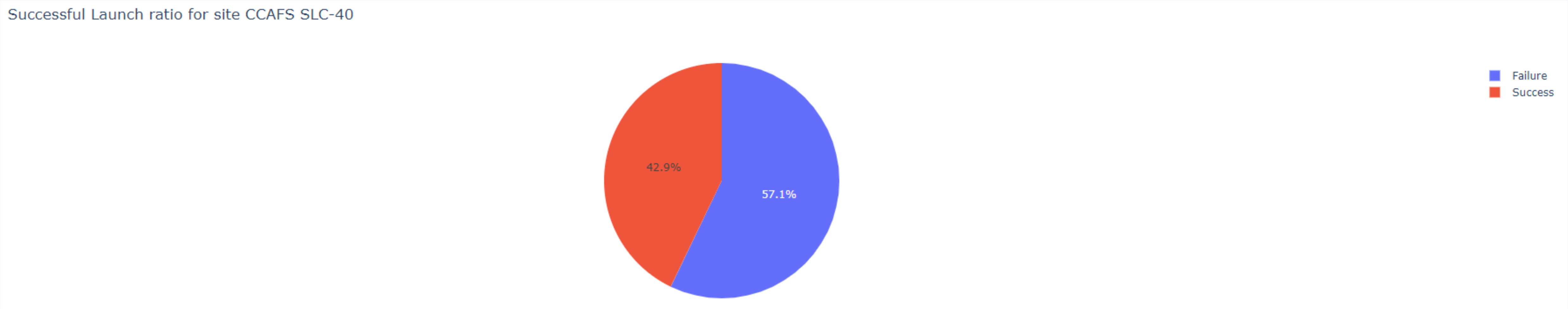
Successful Launches Pie Chart

- This pie chart highlights that over 40% of successful launches came from the KSC LC-39A launch site
- This launch site also has the most launches



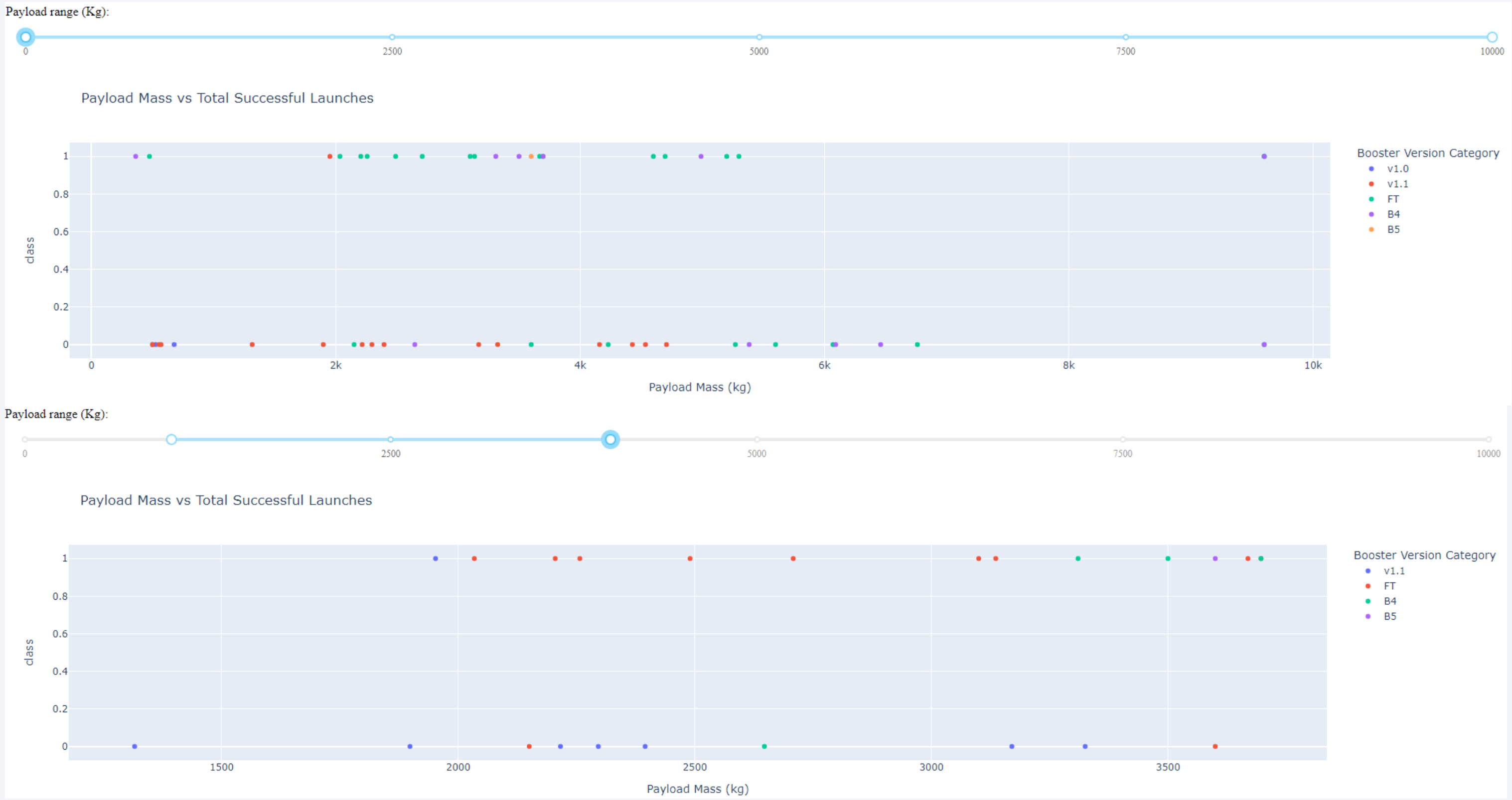
Launch Site with Highest Launch Success Ratio

- The CCAFS SLC-40 has the highest successful launch ratio with 43%
- This statistic is surprising given the success values measured in the other visual analyses



Payload Mass and Launch Outcome: Slider + Scatter Plot

- For the complete range of payload masses and all launch sites, there are a lot of failures for the v1.1 booster
- There is a slightly higher amount of successful launches in a payload mass range of 1200-4000kg

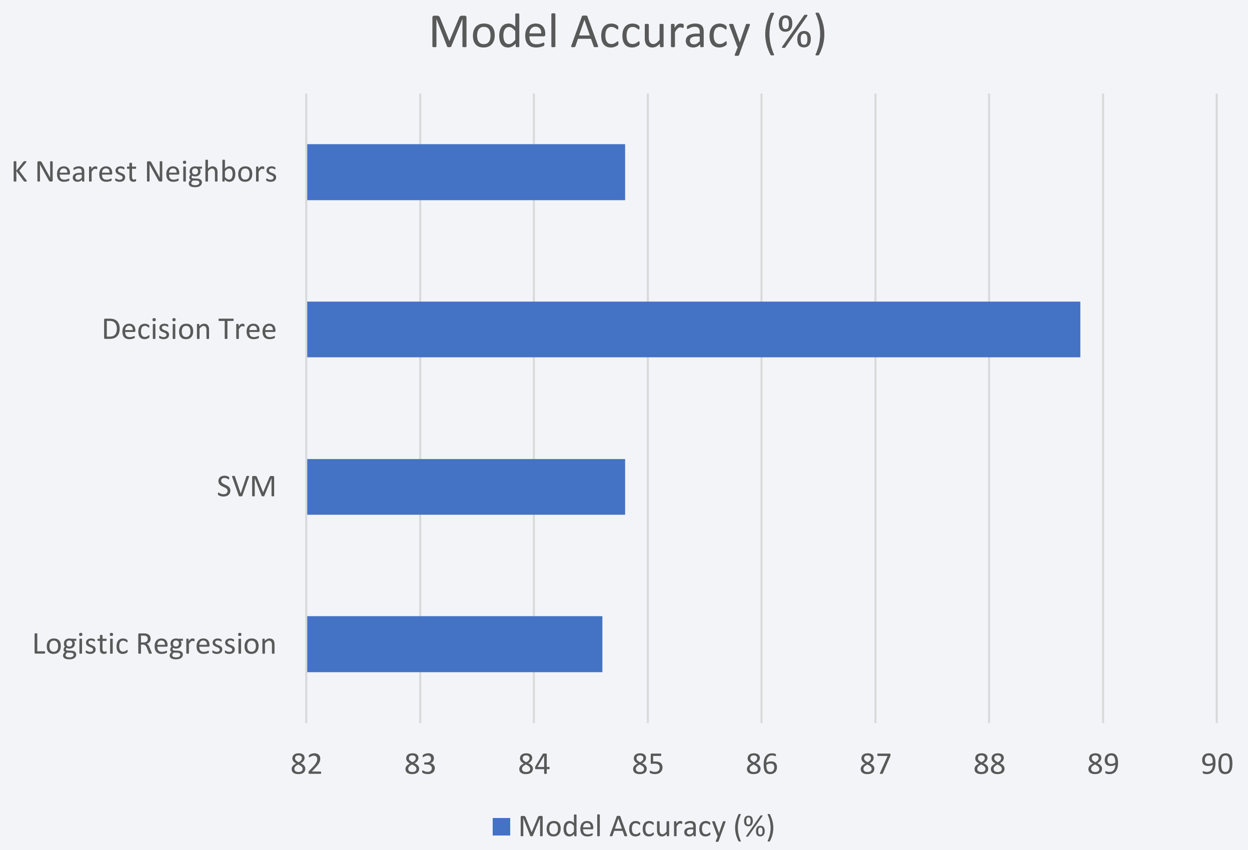


Section 5

Predictive Analysis (Classification)

Classification Accuracy

- The models all performed well in the cross validation (cv) tests
- The **Decision Tree** model outperformed the rest with an accuracy of 88.8%



Confusion Matrix

- Decision Tree Confusion Matrix for the testing data: there are 3 false negatives (3 “did not land” labeled as “land”)
- Decision Tree had the highest score on the cv test but it had an identical confusion matrix as the other models

Conclusions

- Decision Tree model can predict successful landings with 88.8% accuracy
- The most successful launch sites have sub-50% success rates
- Success of SpaceX launches has increased with time
- Launch sites are located near railways and the ocean and far from cities

Appendix

- Included here are the callback for the dashboard pie chart and the SQL command for successful missions, which were complex operations

```
@app.callback(Output(component_id='success-pie-chart', component_property='figure'),
              Input(component_id='site-dropdown', component_property='value'))
def get_pie_chart(entered_site):
    filtered_df = spacex_df
    if entered_site == 'ALL':
        fig = px.pie(filtered_df, values='class',
        names='Launch Site',
        title='Total Successful Launches by Site')

        return fig
    else:
        # return the outcomes piechart for a selected site
        filtered_df_site = filtered_df[filtered_df['Launch Site'] == entered_site]
        class_names = []
        class_values = []
        for output in filtered_df_site['class']:
            if output == 1:
                class_names.append('Success')
                class_values.append(1)
            elif output == 0 :
                class_names.append('Failure')
                class_values.append(1)
        filtered_df_site['class name'] = class_names
        filtered_df_site['class value'] = class_values
        fig = px.pie(filtered_df_site, values='class value',
        names='class name',
        title=f'Successful Launch ratio for site {entered_site}',color_discrete_map={'Success':'lightcyan','Failure':'red'})
        return fig
```

%sql select count(mission_outcome) as "Mission_Failure", (select count(mission_outcome) as "Mission_Success" from SPACEXTBL where mission_outcome = 'Success') from SPACEXTBL where mission_outcome like 'Failure%'

Thank you!