IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Justine Pile
December 10, 2022

Outline

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

Executive Summary

- Summary of methodologies
  - Data was collected from various CSV files of launch data
  - Using machine learning the data was analyzed
- Summary of all results
  - Visualizations were created and coupled with the data analysis of missions, landings, and launches, the optimal model was found

Introduction

- Project background and context
  - We attempted to predict the most optimal situation for Falcon 9 landing so as to know the how to have to best chance of being able to reuse stage 1
- Problems you want to find answers
  - What conditions are most associated with failures vs successes

Section 1

Methodology

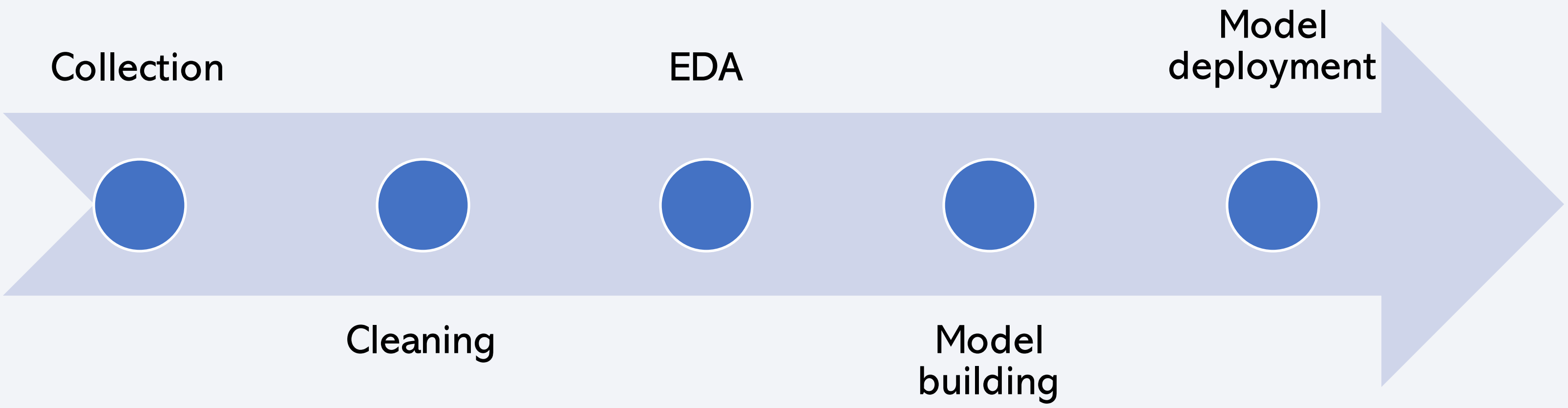
Methodology

Executive Summary

- Data collection methodology:
  - Web scraping and CSVs
- Perform data wrangling
  - Data transformations occurred for use later with visualizations and machine learning
- Perform exploratory data analysis (EDA) using visualization and SQL
  - Created charts
- Perform interactive visual analytics using Folium and Plotly Dash
  - Created a dashboard and used Folium maps
- Perform predictive analysis using classification models
  - Machine learning

Data Collection

- Data was collected from spacexdata.com in CSV form



Data Collection – SpaceX API

Create BeautifulSoup object → Obtain column names → Create the launch_dict → Convert to data frame

```
# Use BeautifulSoup() to create a BeautifulSoup object from a response text content
soup = BeautifulSoup(data, 'html.parser')
```

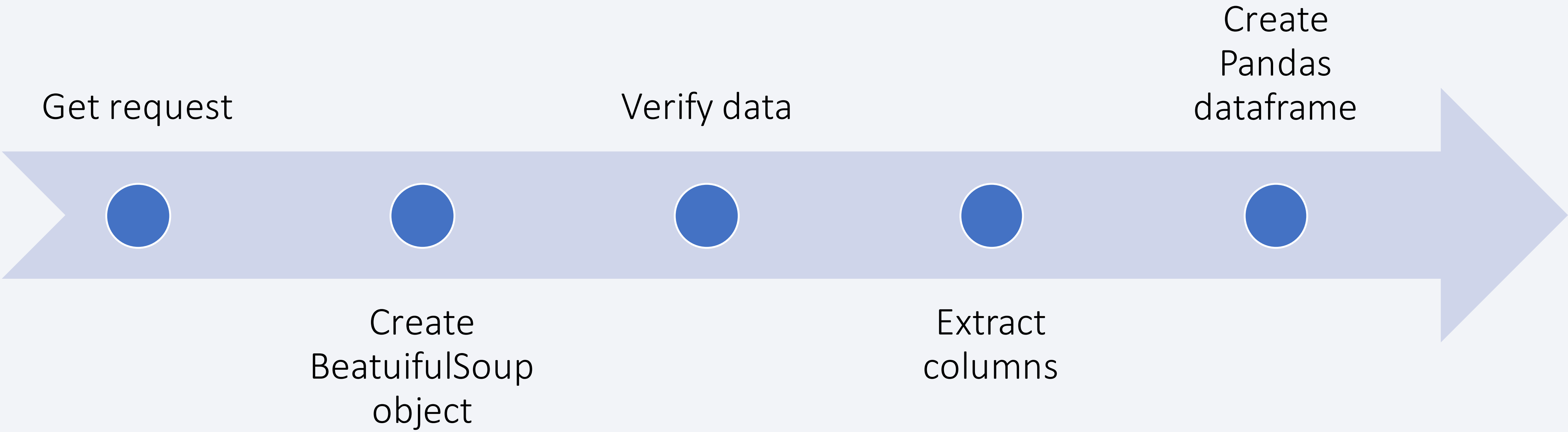
```
column_names = []

# 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 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)

# 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']=[]
```

https://github.com/JustinePile/IBM_Capstone/blob/master/Data%20Collection.ipynb

https://github.com/JustinePile/IBM_Capstone/blob/master/Data%20Collection%20with%20Web%20Scraping.ipynb

Data Collection - Scraping



https://github.com/JustinePile/IBM_Capstone/blob/master/Data%20Collection%20with%20Web%20Scraping.ipynb

Data Wrangling

```
df=pd.read_csv("https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DS0321EN-SkillsNetwork/datasets/dataset_part_1.csv")
df.head(10)
```

	FlightNumber	Date	BoosterVersion	PayloadMass	Orbit	LaunchSite	Outcome	Flights	GridFins	Reused	Legs	LandingPad	Block	ReusedCount	Serial	Longitude	L
0	1	2010-06-04	Falcon 9	6104.959412	LEO	CCAFS SLC 40	None None	1	False	False	False	NaN	1.0	0	B0003	-80.577366	28
1	2	2012-05-22	Falcon 9	525.000000	LEO	CCAFS SLC 40	None None	1	False	False	False	NaN	1.0	0	B0005	-80.577366	28

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

```
bad_outcomes=set(landing_outcomes.keys()[[1,3,5,6,7]])
bad_outcomes
```

Load the data

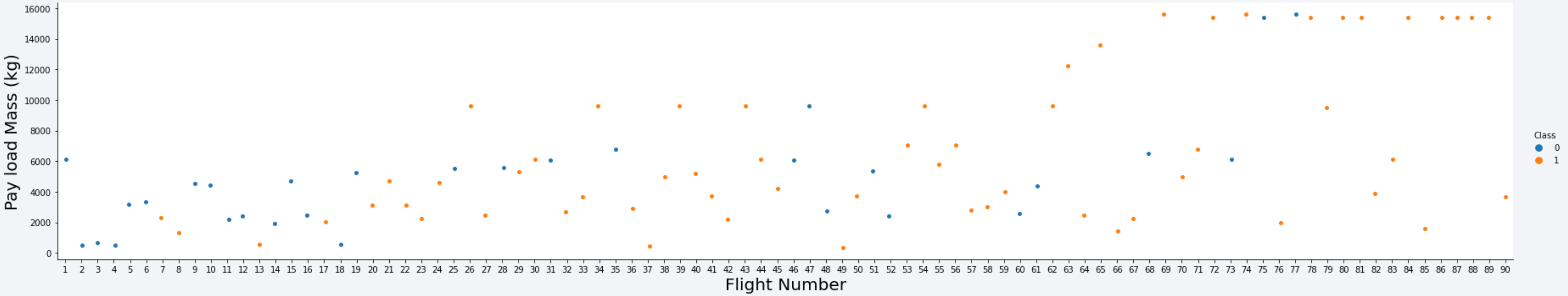
Extract the landing outcomes

successes and failures

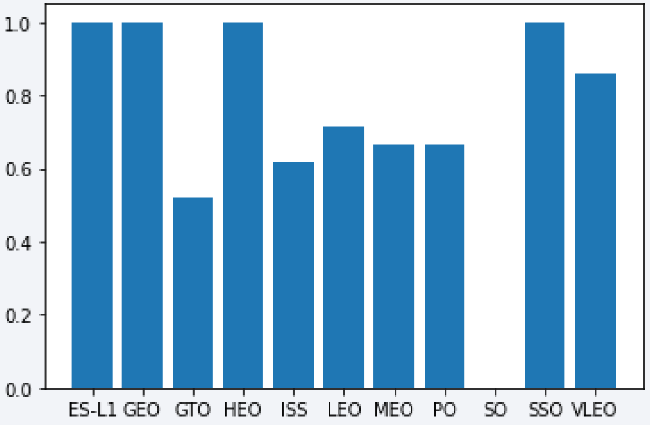
Load the data to a CSV file

https://github.com/JustinePile/IBM_Capstone/blob/master/labs_jupyter_spacex_Data_wrangling.ipynb

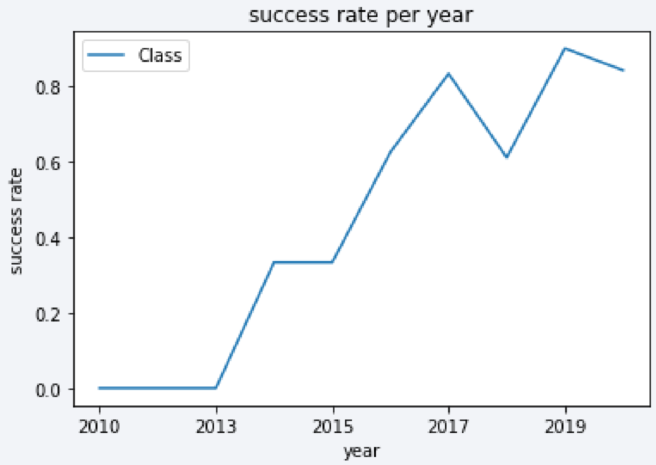
EDA with Data Visualization



Categorical plot looking at the relationship between number of flights, payload mass, and success vs failure



Bar chart looking at the success rate of each orbit type



Line chart showing the success rates per year

https://github.com/JustinePile/IBM_Capstone/blob/master/jupyter_labs_eda_dataviz.ipynb

EDA with SQL

- Display the names of the unique launch sites in the space mission
- Display 5 records where launch sites begin with the string 'CCA'
- Display the total payload mass carried by boosters launched by NASA (CRS)
- Display average payload mass carried by booster version F9 v1.1
- List the date when the first successful landing outcome in ground pad was achieved
- List the names of the boosters which have success on a drone ship and have payload mass greater than 4000 but less than 6000
- List the total number of successful and failure mission outcomes
- List the names of the `booster_versions` which have carried the maximum payload mass. Use a subquery
- List the records which will display the month names for failure `landing_outcomes` on a drone ship for the months in year 2015
- Rank the count of successful `landing_outcomes` between the date 04-06-2010 and 20-03-2017 in descending order.

https://github.com/JustinePile/IBM_Capstone/blob/master/jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

- Mark all launch sites on a map
  - Visualize the locations of launches to look for patterns
- Mark the success/failed launches for each site on the map
  - Visualize the successes and failures to easily see which sites have the best outcomes
- Calculate the distances between a launch site to its proximities
  - Visualize the distance between launch sites and railways, highways, coastline, and cities to attempt to ascertain the importance of being closer or further from any of these

https://github.com/JustinePile/IBM_Capstone/blob/master/lab_jupyter_launch_site_location.ipynb

Build a Dashboard with Plotly Dash

- Add a launch site dropdown menu
- Add a pie chart to show success based on site selected in the dropdown
- Add a range slider to select payload
- Add a pie chart showing success and payload with a scatter chart

All of this allows you to interactively determine what conditions are most likely to lead to success

https://github.com/JustinePile/IBM_Capstone/blob/master/spacex_dash_app.py

Predictive Analysis (Classification)

Standardize data

Split data into training and test sets

Create a regression object

Build the model

Calculate the accuracy

Evaluate with confusion matrix

Evaluate the model

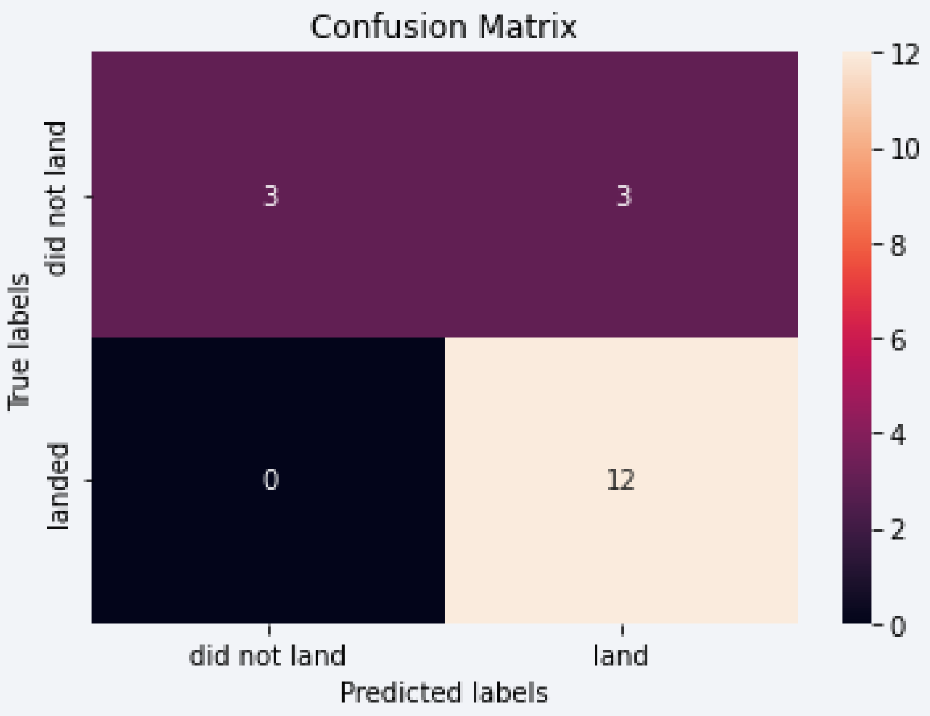
Repeat the process until you find the model with the highest accurary

Find the optimal model

https://github.com/JustinePile/IBM_Capstone/blob/master/SpaceX_Machine_Learning_Prediction_Part_5.ipynb

Results

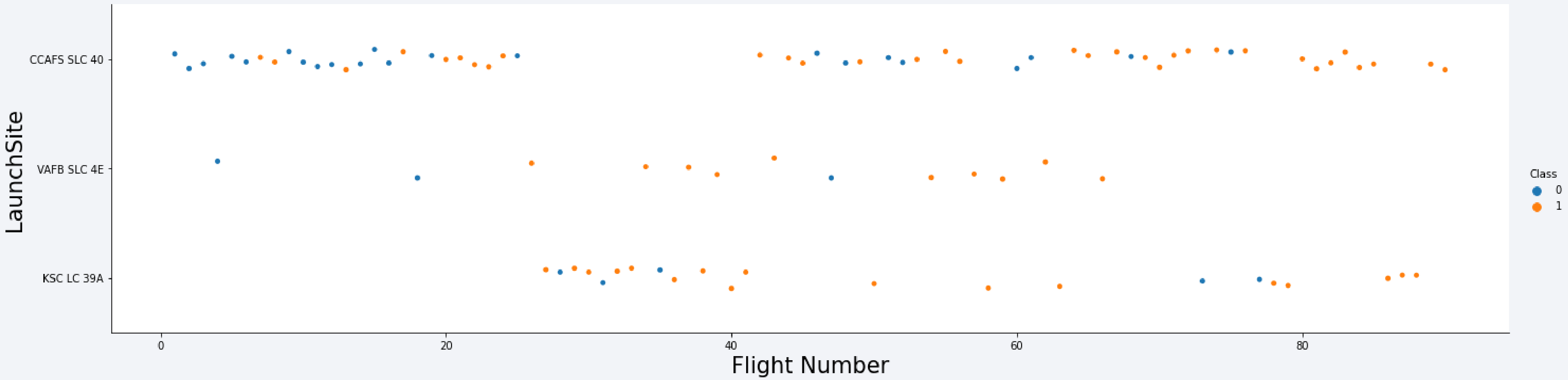
- We were able to conclude that most models had equal accuracy and that the error most likely to occur was the model predicting it had landed when it had actually not



Section 2

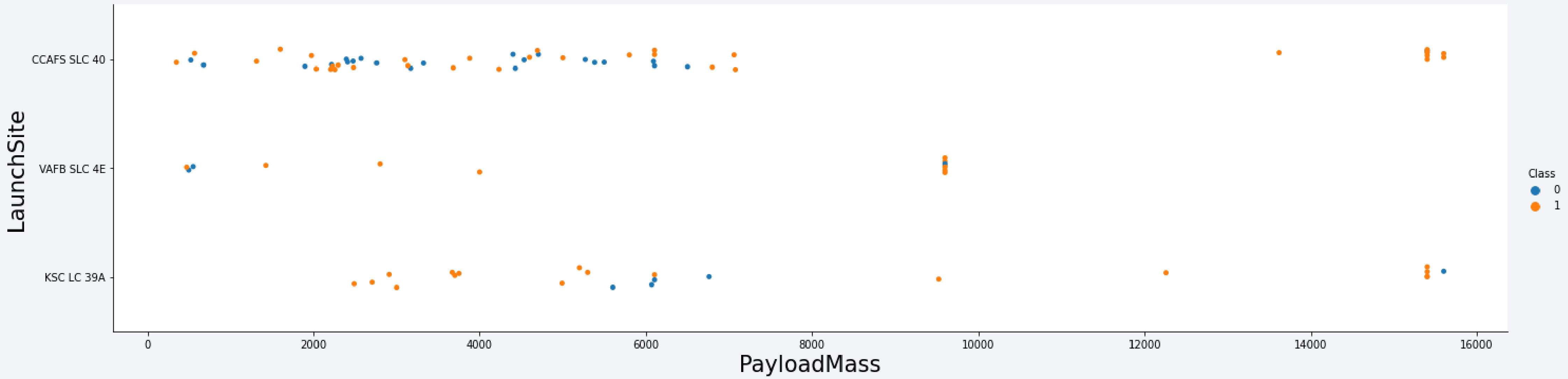
Insights drawn from EDA

Flight Number vs. Launch Site



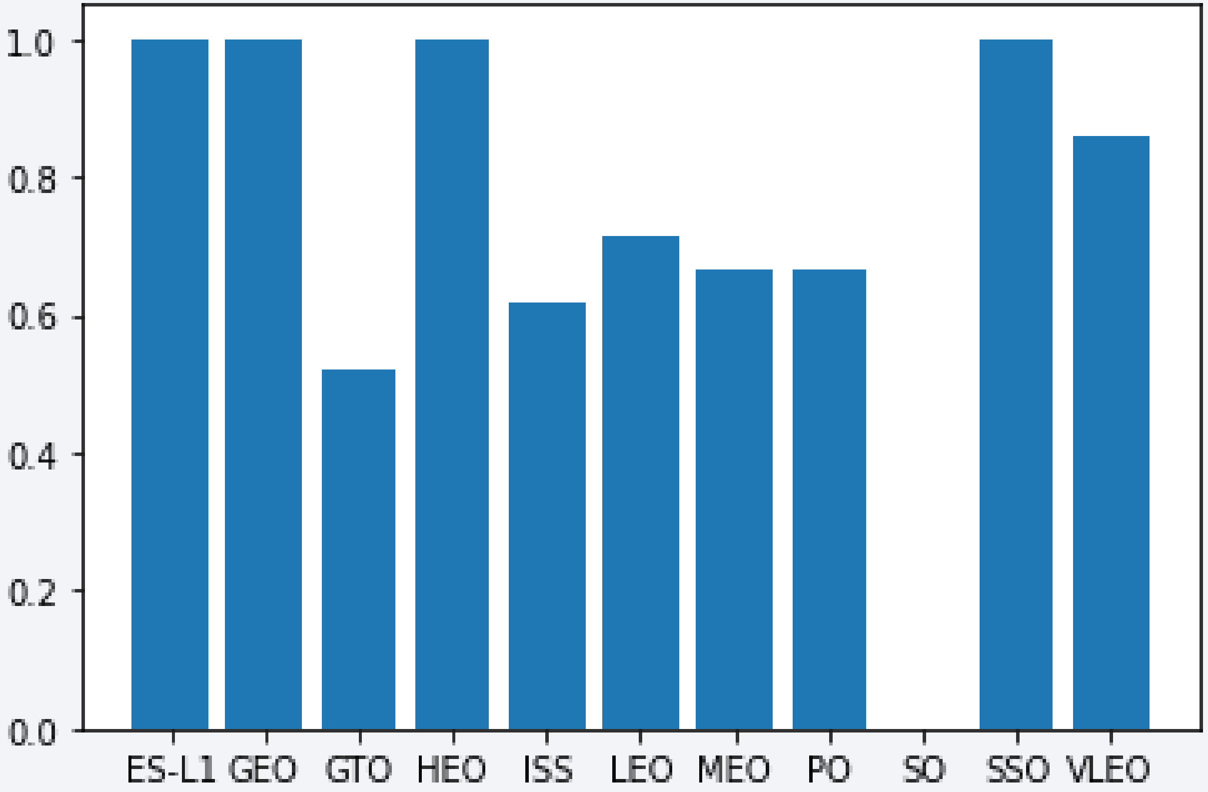
As flight number increases, success rate does as well

Payload vs. Launch Site



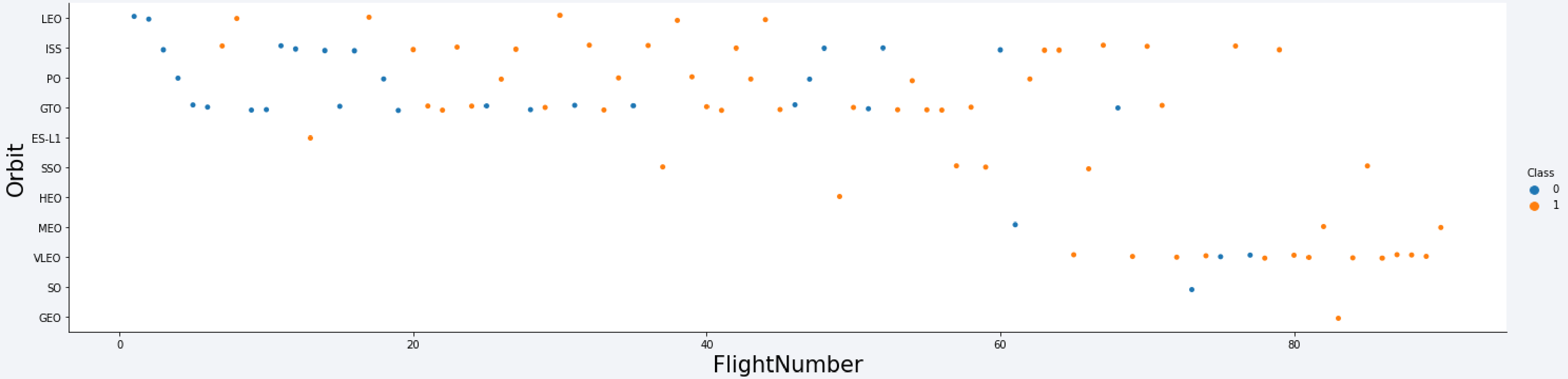
As payload mass increases, so does the success rate

Success Rate vs. Orbit Type



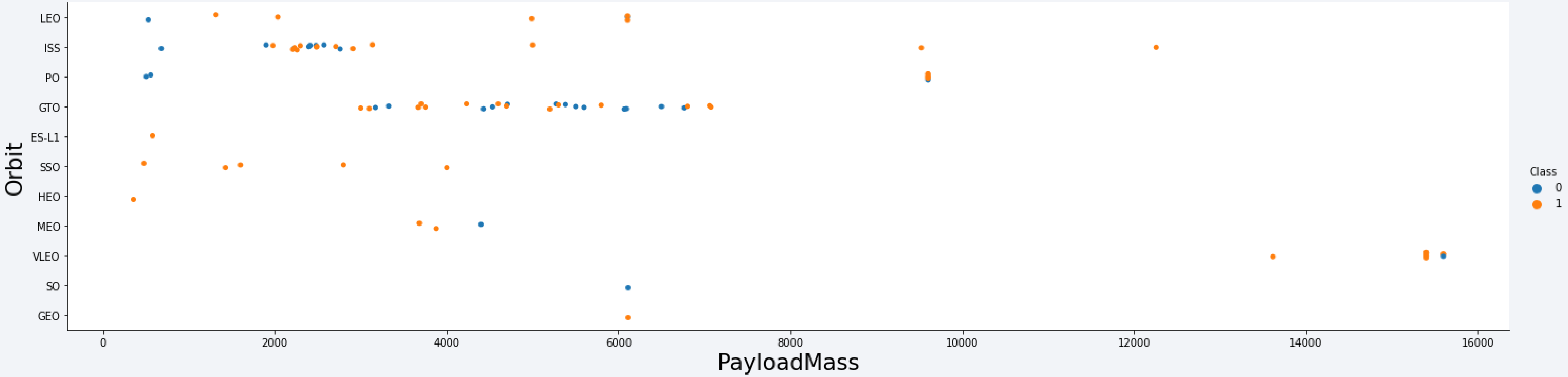
- We can see which orbits types have the highest and lowest success rates
- From the graph we can determine that four orbits have 100% success rate while one has 0% success rate

Flight Number vs. Orbit Type



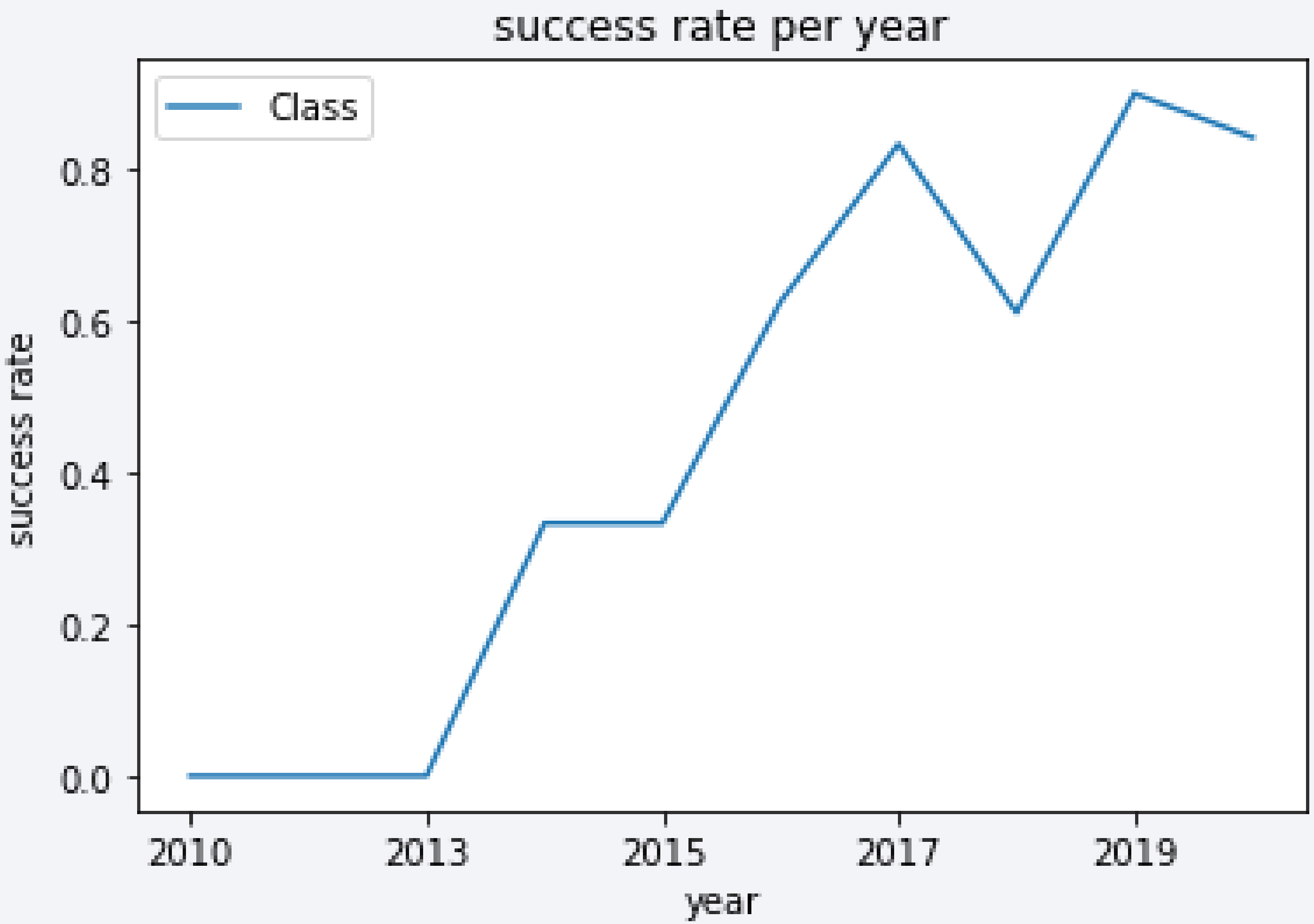
In LEO (low earth orbit) success seems to be tied to the number of flights while most other orbits do not seems to have the same strong correlation

Payload vs. Orbit Type



For the first three orbits types (LEO, ISS, and PO) as payload mass increases so does the likelihood of success

Launch Success Yearly Trend



As time has progressed the number of successes per year has too, aside from a dip in the year 2018

All Launch Site Names

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

We ran a SQL query to determine which launch sites were in the data

Launch Site Names Begin with 'CCA'

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing _Outcome
04-06-2010	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
08-12-2010	15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight C1, two CubeSats, barrel of Brouere cheese	0	LEO (ISS)	NASA (COTS) NRO	Success	Failure (parachute)
22-05-2012	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
08-10-2012	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
01-03-2013	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

We performed a SQL query to find five launch sites that with names that began with CCA

Total Payload Mass

sum(payload_mass__kg_)
45596

This SUM query obtained the sum of all payloads from NASA

Average Payload Mass by F9 v1.1

avg(payload_mass__kg_)
2928.4

This query shows us the average payload mass for F9 v1.1 using a SUM query

First Successful Ground Landing Date

min(`Date`)
01-05-2017

By using SQL MIN function we can see that the first ground landing date in the data is 1/5/2017

Successful Drone Ship Landing with Payload between 4000 and 6000

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

We used the SQL WHERE clause to see which drone ship landings were a success with a criteria of only looking at payload mass between 4000 and 6000

Total Number of Successful and Failure Mission Outcomes

Mission Success
98

Mission Failure
3

Using two SQL queries with a WHERE clause we can see how many missions were successful and how many were failures

Boosters Carried Maximum Payload

Booster_Version
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

Using a SQL WHERE clause we obtain the names of the boosters that had the highest payload by using the SQL MAX function

2015 Launch Records

Month	Landing _Outcome	Booster_Version	Launch_Site
01	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
04	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

We use a SQL WHERE clause to query multiple fields in the table so that we can see all records for 2015

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

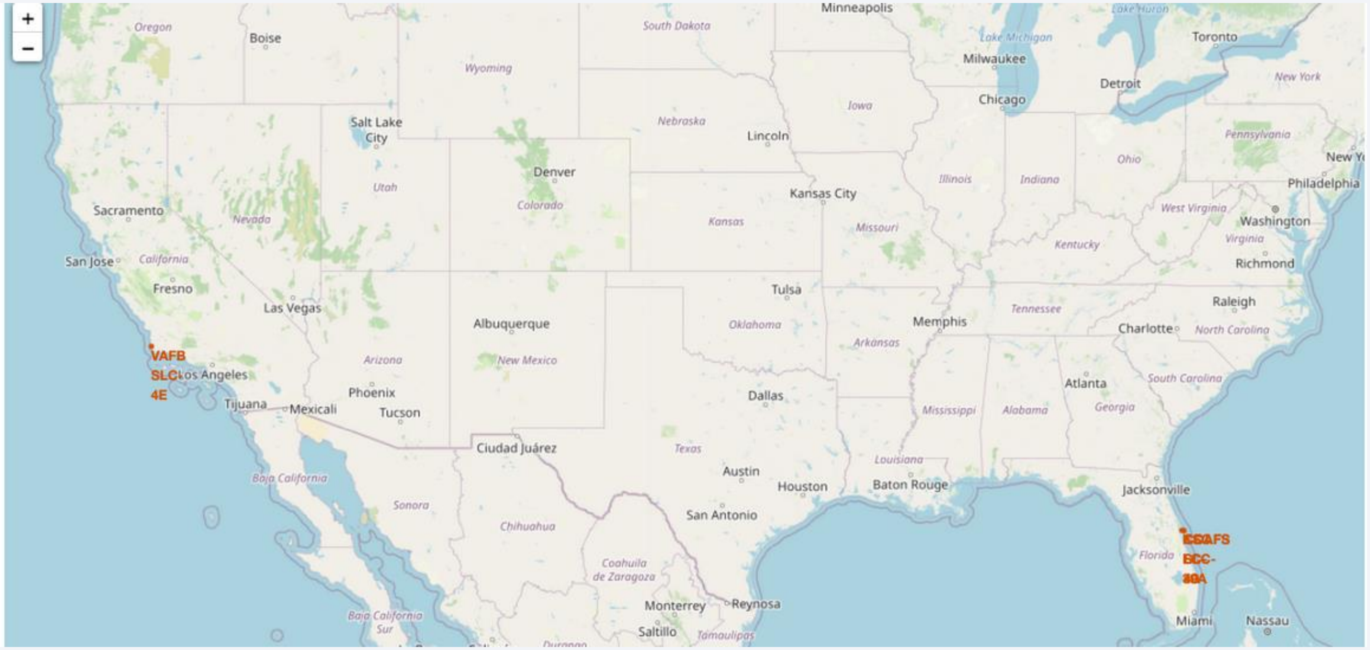
Landing _Outcome	COUNT(`landing _outcome`)
Success	20
Success (drone ship)	8
Success (ground pad)	6

We ran a query to determine the successful landing outcomes using a WHERE clause to limit to the data to the time period 4/6/2010-3/20/2017

Section 3

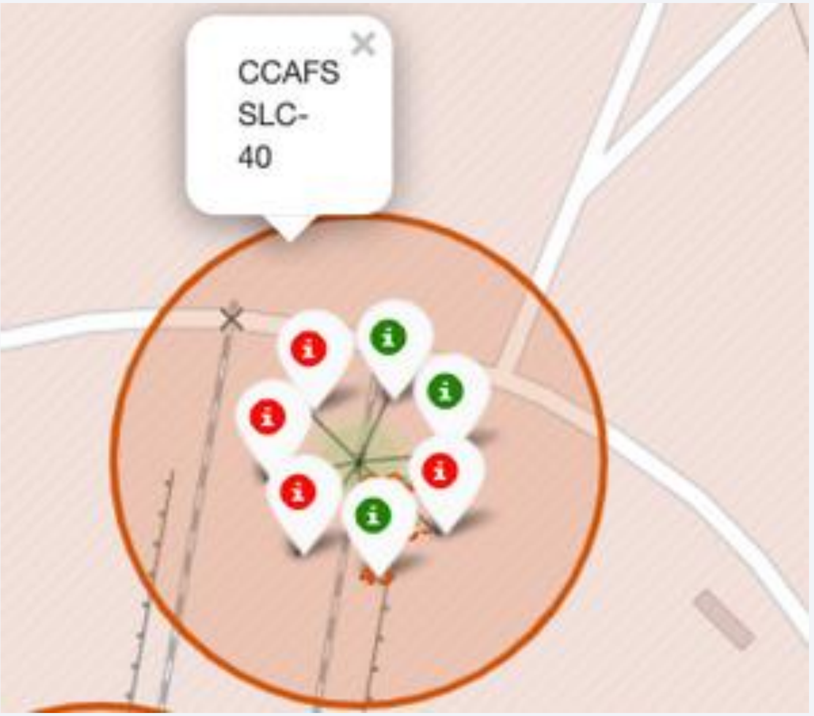
Launch Sites Proximities Analysis

Map of launch sites



By generating a folium map we can see all launch sites' location with markers on a global map

Color coded outcome markers



- With color coded markers we can see successes in green and failures in red to more easily identify successes vs failures at a given site

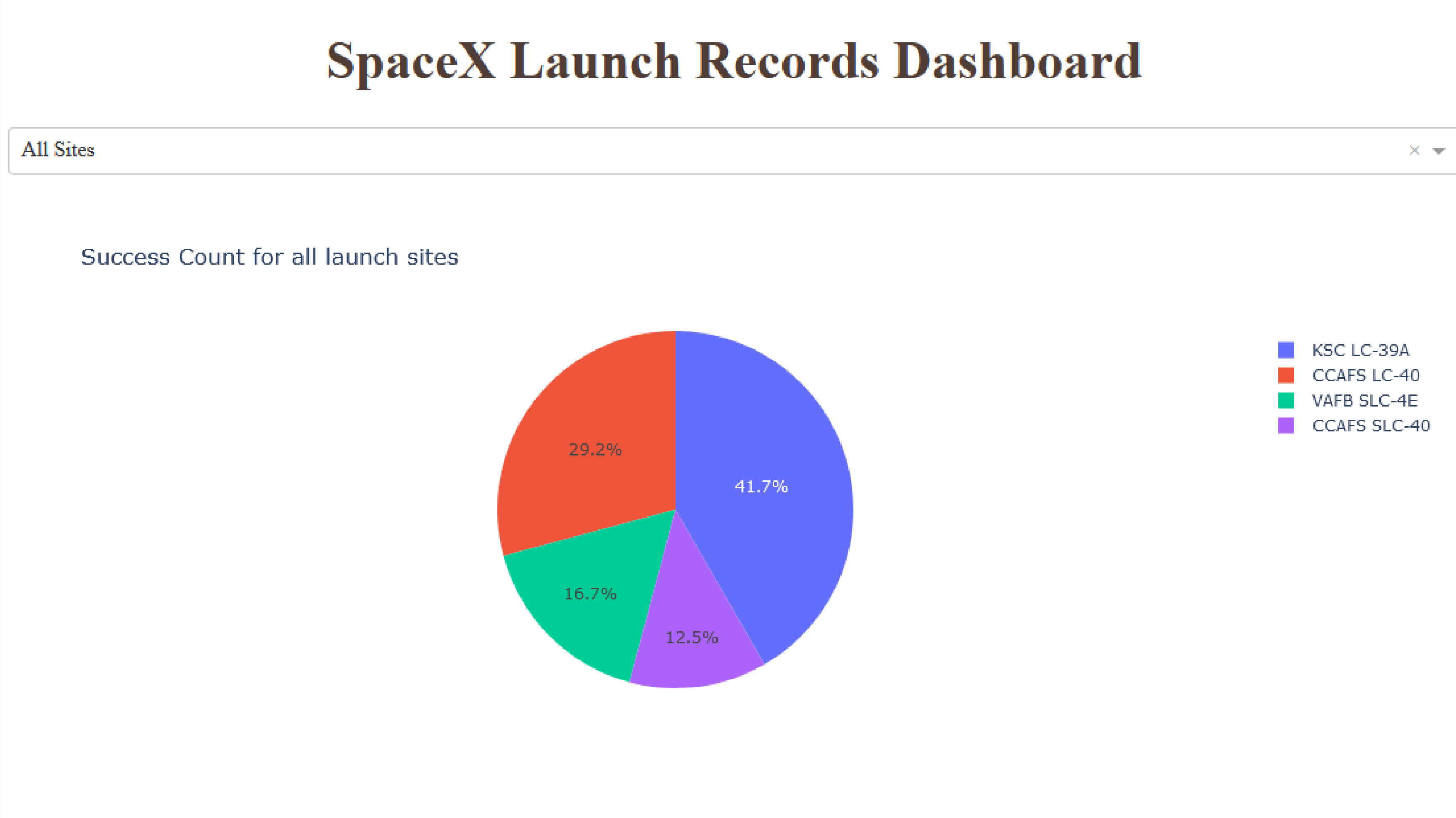
Launch site proximities

By creating markers on the map for railways, coastline, cities, and highways we can attempt to ascertain their importance to launch sites

Section 4

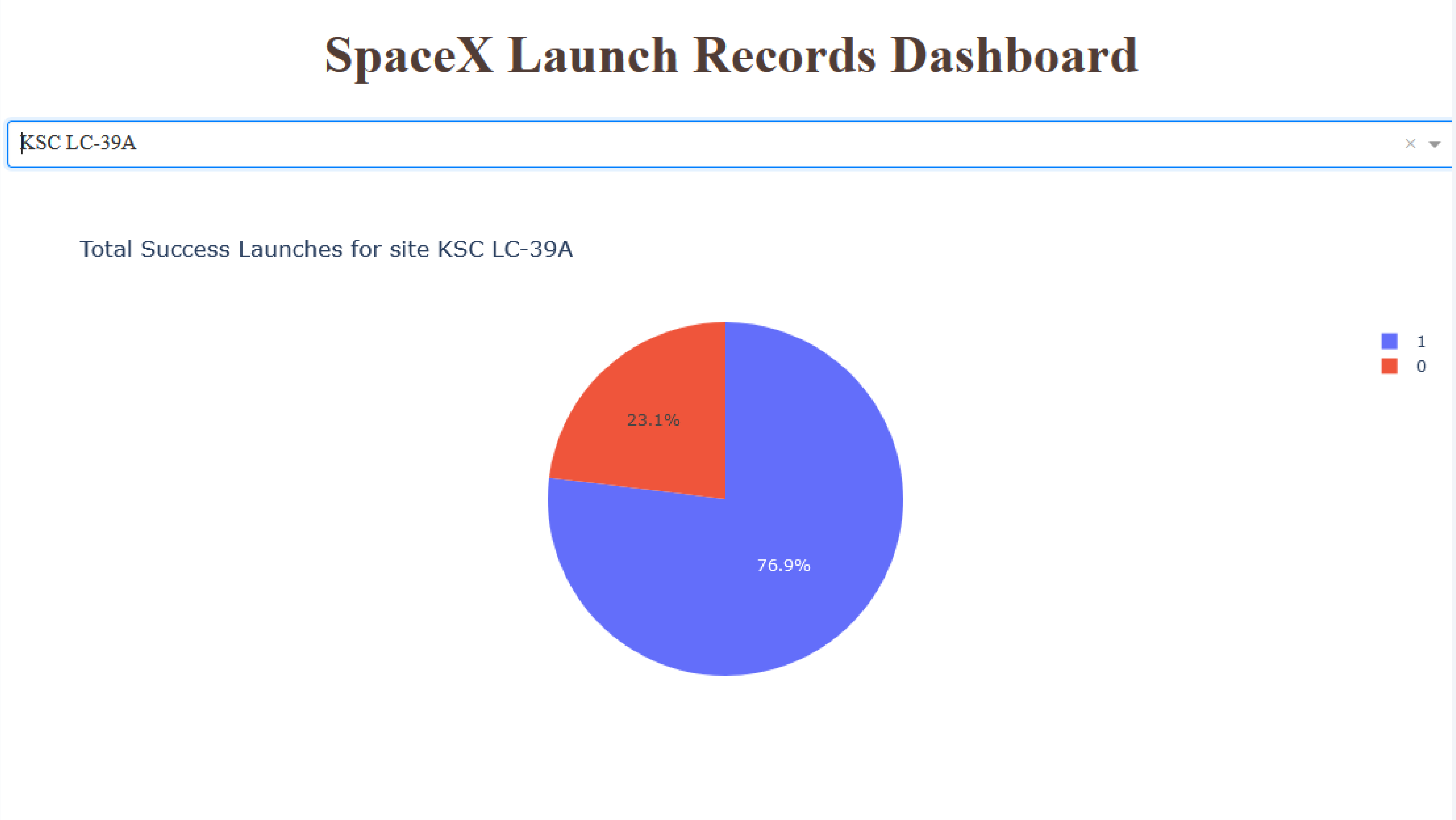
Build a Dashboard with Plotly Dash

Total successful launches for all site



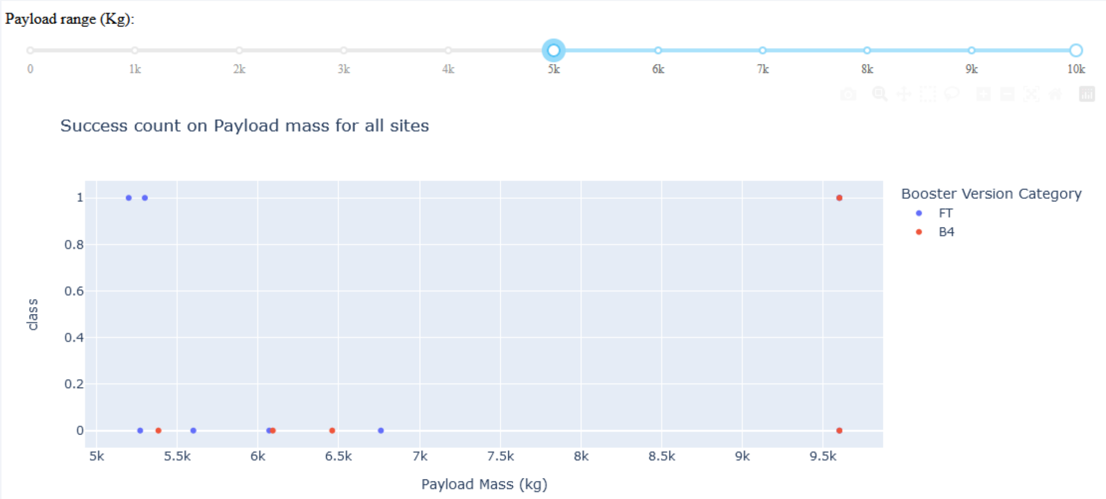
We have an interactive dashboard which shows successful launches for all sites

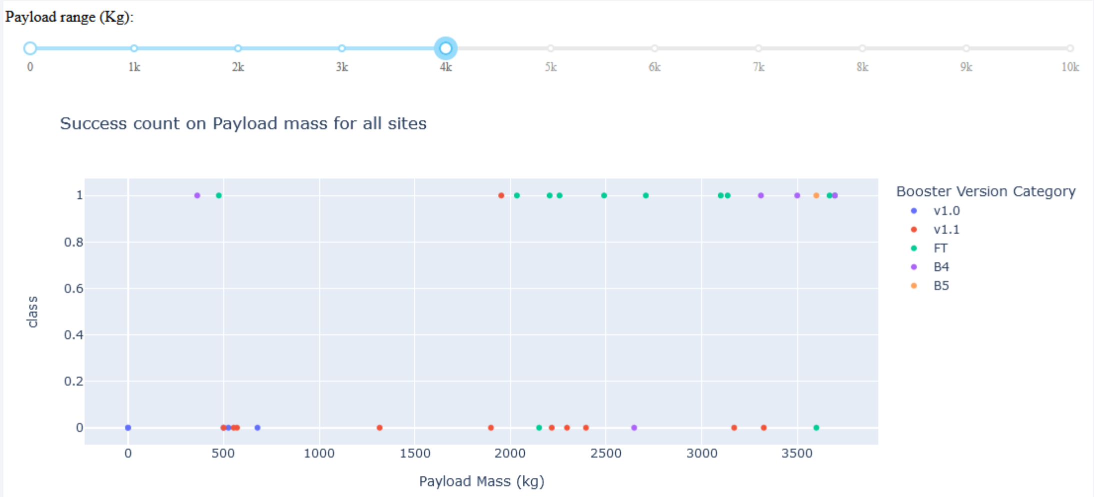
Launch site with the most successes



- We can select the KSC LC-39A site which is the site with the most successful launches
- We can see the ratio of successes to failures for this site

Payload and launch outcomes





Here we can drag the slider to filter the payload mass displayed in the graph

Section 5

Predictive Analysis (Classification)

Confusion Matrix

- We can see that this model which had a score of 0.94 has the best accuracy
- As discussed earlier, inaccuracies in this model tended to be predicting a success when it was actually a failure

Conclusions

- KSC LC 39-A had the highest amount of successes
- There was more data (i.e. more launches) for lower payload masses
- The decision tree model has the highest accuracy
- It is important to consider the impact on mission success of what is nearby to a launch site and where the launch site is located

Thank you!