IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Jihyoun Song
18 June 2024

Outline

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

Executive Summary

- In this capstone, we predict if the Falcon 9 first stage will land successfully with using the Space X launch history data and several machine learning algorithms
- We collect SpaceX launch history data with SpaceX REST API and Web Scraping
- We perform some Exploratory Data Analysis (EDA) to find some patterns in the data and determine training labels for supervised machine learning models
- We create a machine learning pipeline and find the classification model which performs best
- In this capstone, it is concluded that Decision Tree Model is the best machine learning algorithm to predict if the Falcon 9 first stage will land successfully

Introduction

- Space X advertises Falcon 9 rocket launches on its website with a cost of 62 million dollars
- As Space X can reuse the first stage, its cost is much more reasonable than other providers, which is upward of 165 million dollars
- If we can predict whether the first stage will land successfully, we can determine the cost of a launch

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - SpaceX REST API and Web Scraping
- Perform data wrangling
  - Convert the mission outcomes into Training Labels
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Standardize the data and split into training and test data
  - Find best Hyperparameter for models and calculate accuracy

Data Collection – SpaceX API

- End point URL
  https://api.spacexdata.com/v4/launches/past
- We perform a get request to obtain the data
- The response is in the form of a JSON
- To convert this response to a Pandas dataframe, we use .json_normalize() function

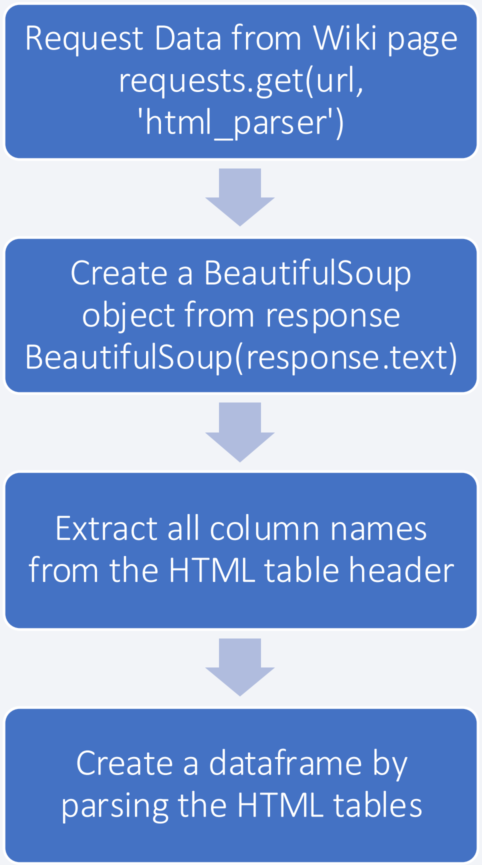
Request Data from API
response = requests.get(url)

Decode the response as Json
response.json()

Turn it into a Pandas dataframe
.json_normalize()

Data Collection - Scraping

- Used Web URL
  https://en.wikipedia.org/w/index.php?title=List_of_Falcon_9_and_Falcon_Heavy_launches&oldid=1027686922
- We perform a HTTP get request to obtain HTML pages
- We extract all relevant column names from the HTML table header
- We create an empty dictionary with keys (column names) and convert it into a Pandas dataframe



Data Wrangling

- We deal with missing values for column PayloadMass
  - We calculate the mean using .mean()
  - We replace missing values with the mean using .replace()
- In the data set, there are several cases where the Falcon 9 first stage did not land successfully such as False Ocean, False RTLS, False ASDS, None ASDS, None None
- There are different cases where it landed successfully such as True Ocean, True RTLS, True ASDS
- We convert those outcomes into Training Labels and save them in column Class
  - 1 means it successfully landed
  - 0 means it was unsuccessful.

Calculate the mean for the PayloadMass

↓

Replace missing values with the mean for PayloadMass column

↓

Calculate the occurence of mission outcome

↓

Create a landing outcome label from Outcome column

EDA with Data Visualization

- Three charts were plotted to visualize the Data
  - Scatter chart, Bar chart, Line chart
- Scatter chart is used to visualize the relationship between two features
  - Flight number vs. Launch site
  - Payload vs. Launch site
  - Flight number vs. Orbit type
  - Payload vs. Orbit type
- Bar chart is used to compare two features
  - Success rate of each orbit type
- Line chart is used to see the trend of the data
  - The launch success yearly trend

EDA with SQL

- DISTINCT query
  - Display the names of the unique launch sites in the space mission
- SUM(), AVG() query
  - Calculate the total and the average of payload mass carried by boosters
- COUNT() query
  - Count the total number of mission outcomes
- GROUP BY query
  - Group records into categories
- ORDER BY query with DESC query
  - Sort the result in descending order

Build an Interactive Map with Folium

- Circle object
  - To add a highlighted circle on a specific coordinate
  - Using folium.Popup() to add a popup label
- Marker object
  - To add a marker on a specific coordinate with an icon
  - Markers are like signposts highlighting important elements on the map
- MousePosition object
  - To get coordinate for a mouse over a point on the map
- PolyLine object
  - To draw a line between a launch site to the selected point

Build a Dashboard with Plotly Dash

- The dashboard contains a dropdown list and a range slider as input components
- Those input components interact with a pie chart and a scatter point chart
- Users use a dropdown list to see success-pie-chart based on selected site
  - The pie chart shows the successful rate on selected site
- Users use a range slider to select payload range to render the success-payload-scatter-chart
  - The scatter chart shows whether the booster has landed successfully in the selected payload range and it is categorized by booster versions

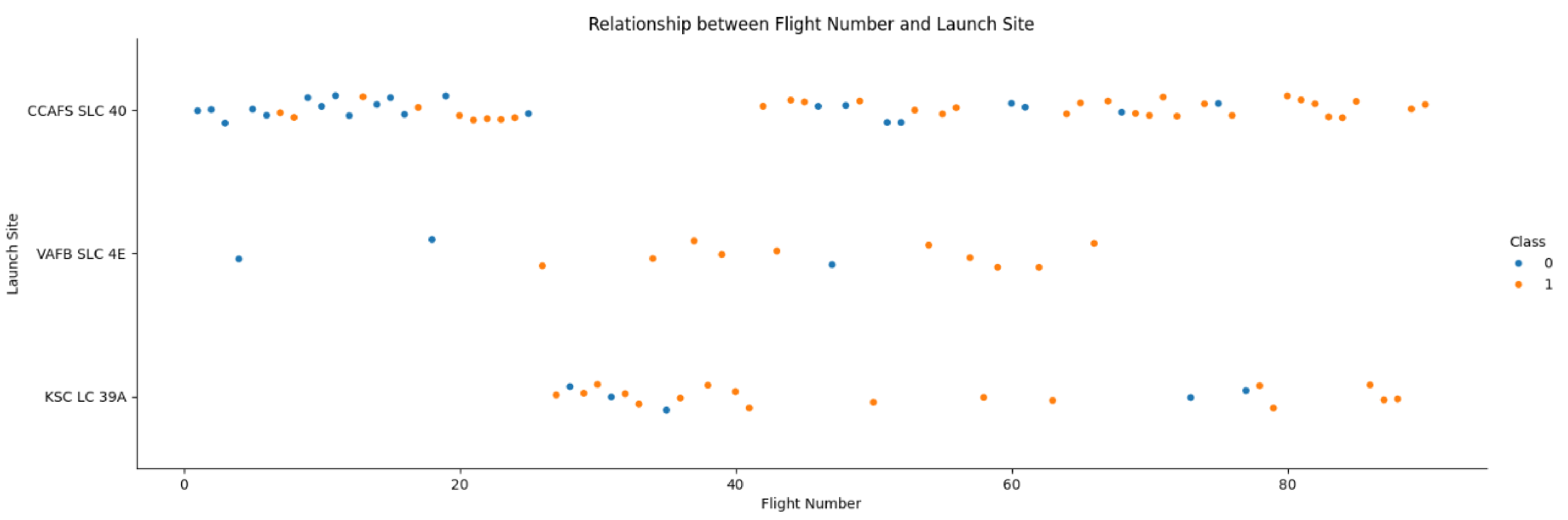
Predictive Analysis (Classification)

- Create a column for the class
  - The column Class from the data is turned to NumPy array as variable Y
- Standardize the data
  - Features to predict the result is assigned to variable X
  - preprocessing.StandardScaler().fit_transform(X) to standardize X
- Split into training and test data
  - train_test_split() to split the data X and Y into training and test data
- Find the best hyperparameter for each model
  - Create a GridSearchCV object for each model
  - Fit it with parameters in a dictionary form
  - Using the data attribute best_params_, we find the best parameters
- Find the best performing model
  - Using the data attribute best_score_, we get the accuracy
  - With confusion matrix, we visualize the accuracy on the test data set

Section 2

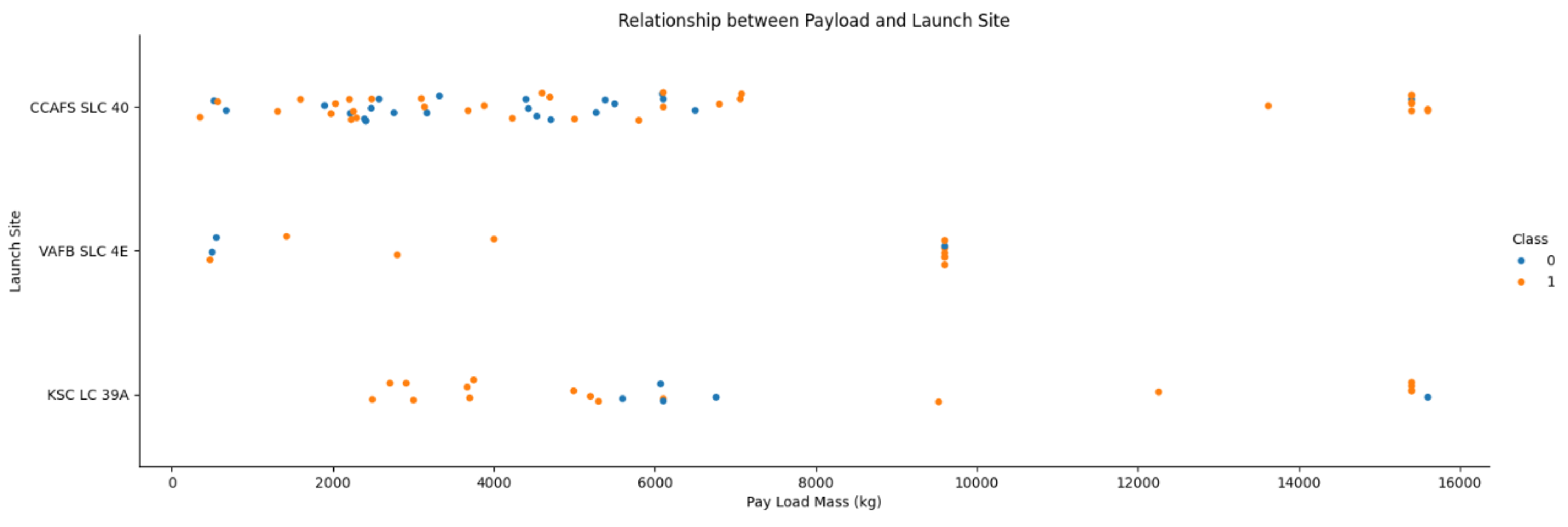
Insights drawn from EDA

Flight Number vs. Launch Site



- On every launch site, we see that as the flight number increases, the first stage is more likely to land successfully
- CCAFS SLC 40 has the most amount of flights and their success rate is 100% for recent attempts (the flight number is greater than 80)

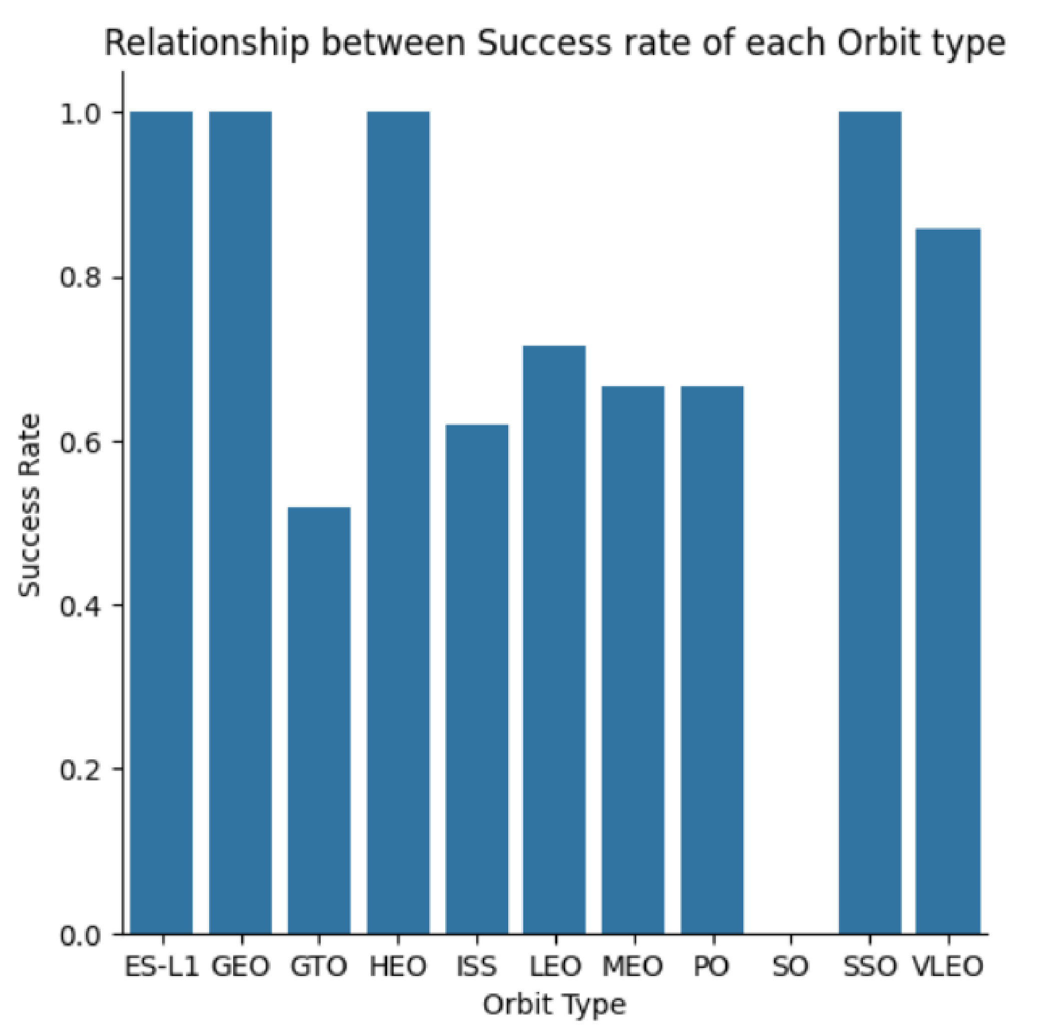
Payload vs. Launch Site



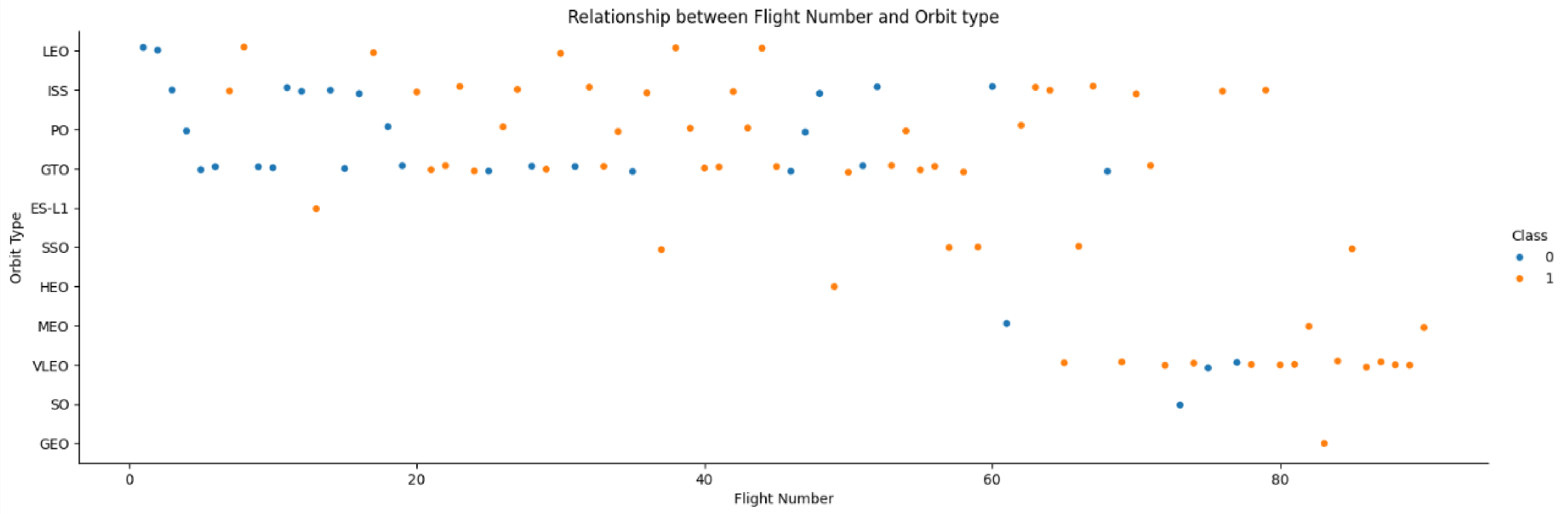
- VAFB-SLC has no rockets launched for heavy payload mass (greater than 10000)
- CCAFS SLC 40 had no rockets launched for payload mass between 8000 and 13000

Success Rate vs. Orbit Type

- Orbit type 'SO' has the lowest success rate, which is 0%
- Orbit type 'ES-L1', 'GEO', 'HEO', and 'SSO' have the highest success rate, which is100%
- Other than 'SO', success rate for each orbit type is more than 50%

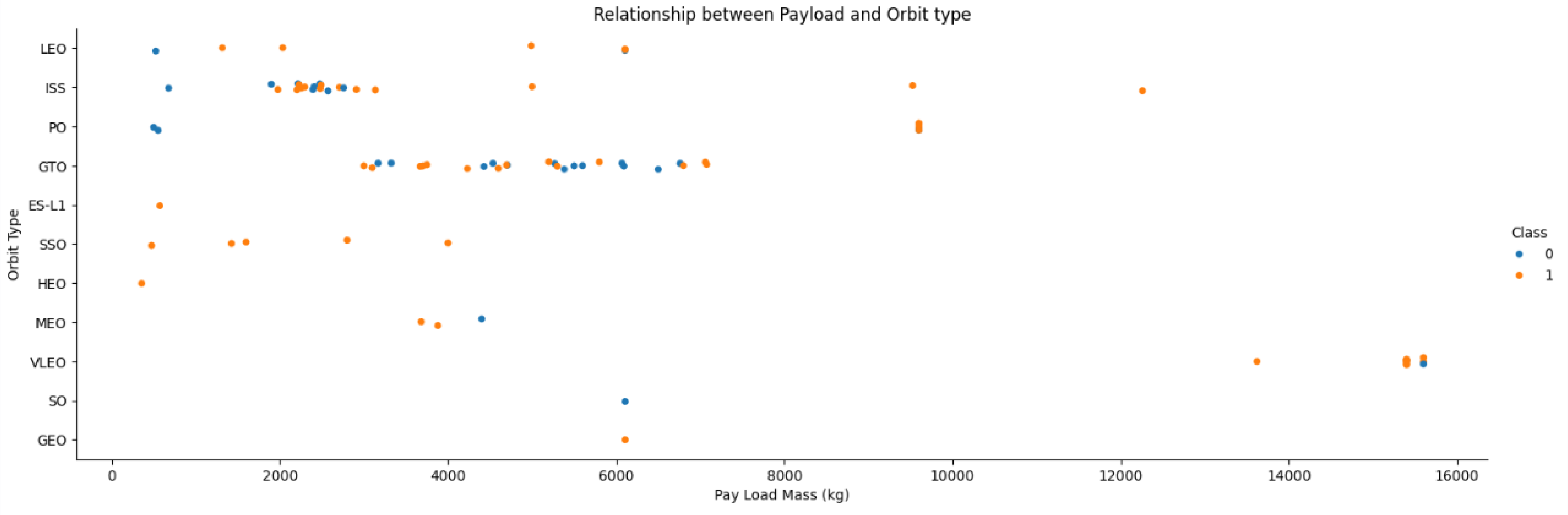


Flight Number vs. Orbit Type



- We see that in the orbit type 'LEO', as the flight number increases, the first stage is more likely to land successfully
- In the orbit type 'GTO', there seems to be no relationship between flight numbers

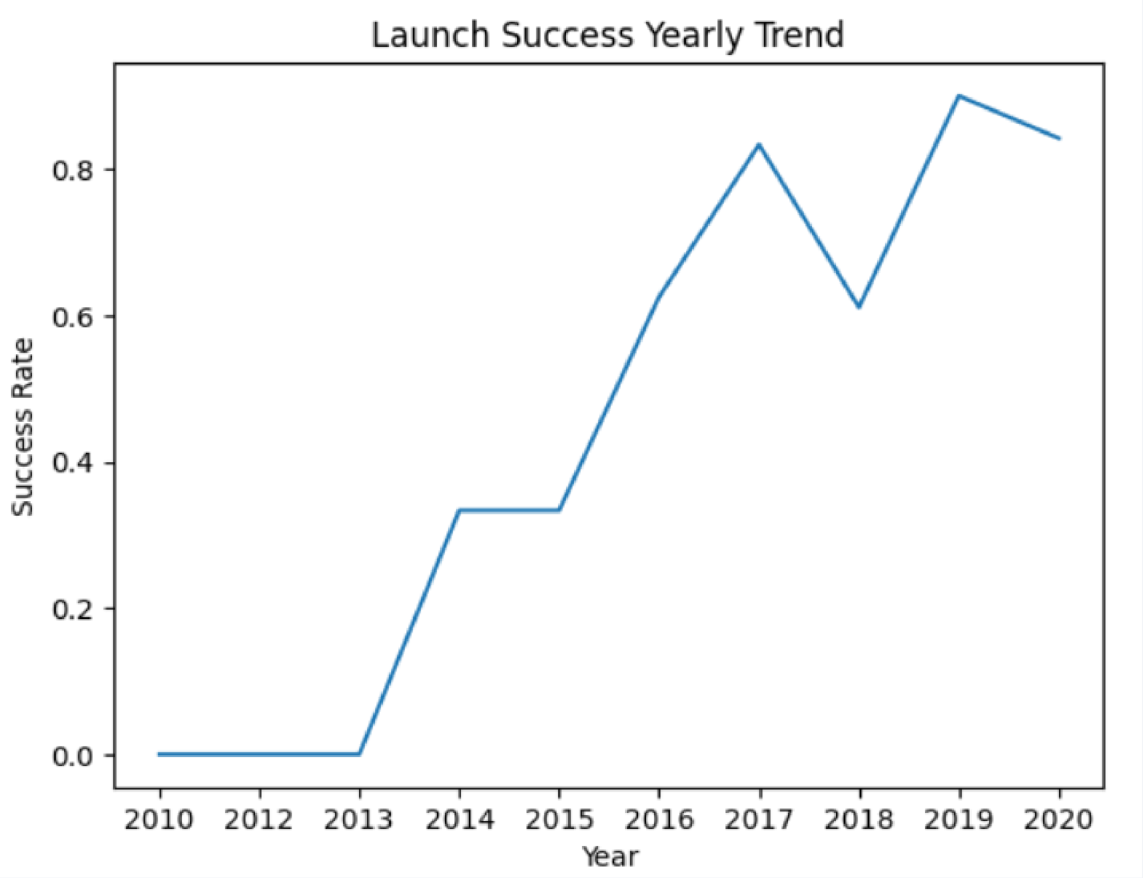
Payload vs. Orbit Type



- With heavy payload mass, the success rate increases for the orbit type 'PO', 'LEO', and 'ISS'
- In the orbit type 'GTO', there seems to be no relationship between payload mass

Launch Success Yearly Trend

- In general, the success rate since 2013 kept increasing till 2020
- Between 2017 and 2018, the success rate has decreased (80% in 2017 and 60% in 2018)
- In 2014 and 2015, the success rate was same (33%)



All Launch Site Names

- Find the names of the unique launch sites using SQL query DISTINCT

```
%sql select distinct Launch_Site from SPACEXTABLE
```

- All the unique launch sites are shown below

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

Launch Site Names Begin with 'CCA'

- Find 5 records where launch sites begin with `CCA`
  - Launch_Site like 'CCA%' to find launch sites beginning with 'CCA'
  - limit 5 to see only 5 records

```
%sql select * from SPACEXTABLE where Launch_Site like 'CCA%' limit 5
```

- the query result is shown below

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
2010-12-08	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)
2012-05-22	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
2012-10-08	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
2013-03-01	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

Total Payload Mass

- Calculate the total payload carried by boosters from NASA (CRS)
  - sum(PAYLOAD_MASS__KG_) to calculate the total payload
  - where Customer = 'NASA (CRS)' to select the boosters from NASA (CRS)

```
%sql select sum(PAYLOAD_MASS__KG_) as `total payload mass (kg)` from SPACEXTABLE where Customer = 'NASA (CRS)'
```

- The result is shown below

total payload mass (kg)
45596

Average Payload Mass by F9 v1.1

- Calculate the average payload mass carried by booster version F9 v1.1
  - avg(PAYLOAD_MASS__KG_) to calculate the average payload mass
  - where Booster_Version like 'F9 v1.1%' to find booster version F9 v1.1

```
%sql select avg(PAYLOAD_MASS__KG_) as `average payload mass (kg)` from SPACEXTABLE where Booster_Version like 'F9 v1.1%'
```

- The query result is shown below

average payload mass (kg)
2534.6666666666665

First Successful Ground Landing Date

- Find the dates of the first successful landing outcome on ground pad
  - min(Date) to see the date of the first successful landing outcome on ground pad
  - where Landing_Outcome like '%Success (ground%' to find only the successful landing outcome on ground pad

```
%sql select min(Date) as Date from SPACEXTABLE where Landing_Outcome like '%Success (ground%'
```

- the query result is shown below

Date
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

- - select Booster_Version to list the names of the boosters
  - Landing_Outcome like 'Success%drone%' to find records which have successfully landed on drone ship
  - PAYLOAD_MASS__KG_ between 4000 and 6000 to find records which had payload mass greater than 4000 but less than 6000
  - Combining two conditions with and query

```
%sql select Booster_Version from SPACEXTABLE where (Landing_Outcome like 'Success%drone%') and (PAYLOAD_MASS__KG_ between 4000 and 6000)
```

- The result is shown in the table

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

Total Number of Successful and Failure Mission Outcomes

- Calculate the total number of successful and failure mission outcomes
  - count(*) to calculate the total number of records
  - group by Mission_Outcome to group records into success and failure

```
%sql select Mission_Outcome, count(*) as `total number` from SPACEXTABLE group by Mission_Outcome
```

- The query result is shown below

Mission_Outcome	total number
Failure (in flight)	1
Success	98
Success	1
Success (payload status unclear)	1

Boosters Carried Maximum Payload

- List the names of the booster which have carried the maximum payload mass
  - distinct Booster_Version to list the names of the boosters
  - using a subquery to find the records which have carried the maximum payload mass

```
%sql select distinct Booster_Version from SPACEXTABLE
where PAYLOAD_MASS__KG_ = (select max(PAYLOAD_MASS__KG_) from SPACEXTABLE)
```

- The result is shown in the table

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

2015 Launch Records

- List the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015
  - SUBSTR(Date, 6, 2) to get the month and SUBSTR(Date, 0, 5) to get the year
  - Two conditions combined with and query

```
%sql select SUBSTR(Date,6,2) as Month, Landing_Outcome, Booster_Version, Launch_Site From SPACEXTABLE
where (SUBSTR(Date, 0, 5) = '2015') and (Landing_Outcome like 'Failure%drone%')
```

- The result is shown below

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

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

- \- Date between '2010-06-04' and '2017-03-20' to get records for those periods
  \- order by count(Landing_Outcome) desc to sort the result in descending order

```
%sql select Landing_Outcome, count(Landing_Outcome) from SPACEXTABLE
where Date between '2010-06-04' and '2017-03-20' group by Landing_Outcome order by count(Landing_Outcome) desc
```

- The result is shown in the table

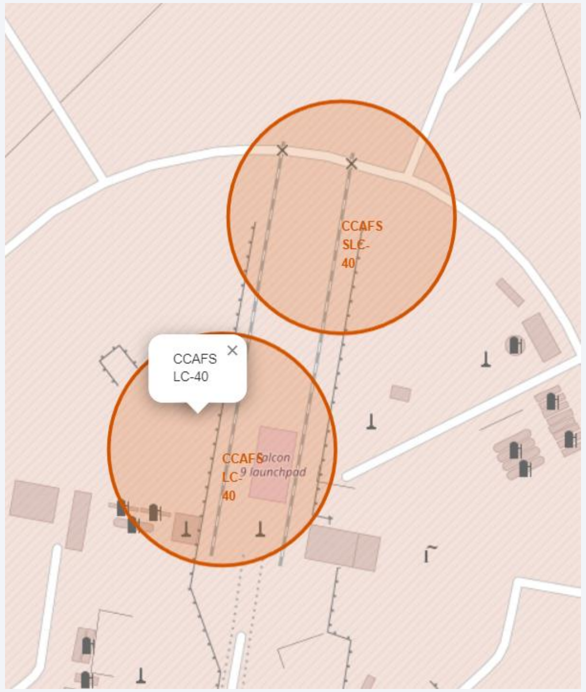
Landing_Outcome	count(Landing_Outcome)
No attempt	10
Success (drone ship)	5
Failure (drone ship)	5
Success (ground pad)	3
Controlled (ocean)	3
Uncontrolled (ocean)	2
Failure (parachute)	2
Precluded (drone ship)	1

Section 3

Launch Sites Proximities Analysis

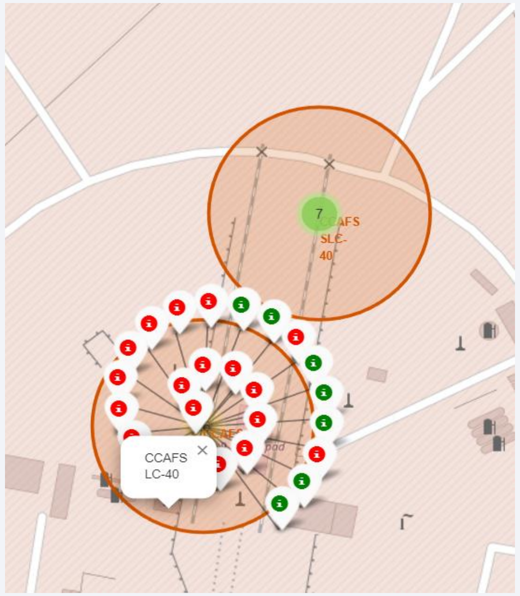
Task 1: Mark all launch sites on a map

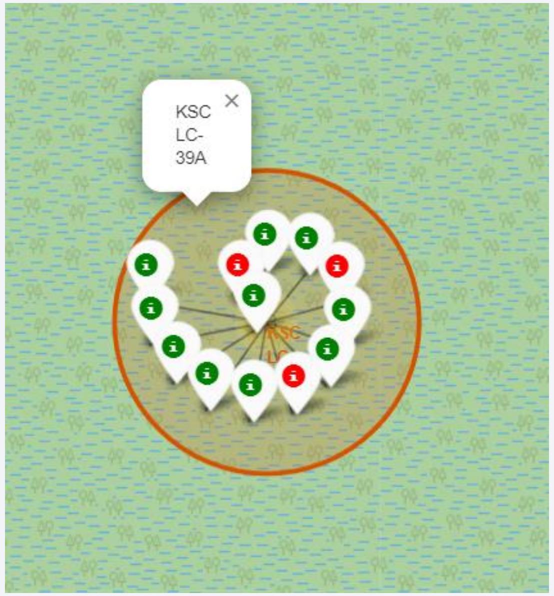
- We see that all launch sites are in proximity to the Equator line
- We see that all launch sites are in very close proximity to the coast

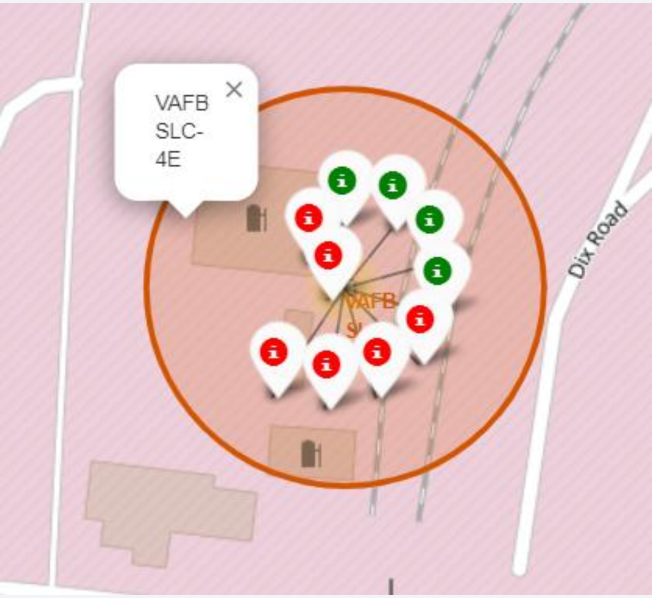


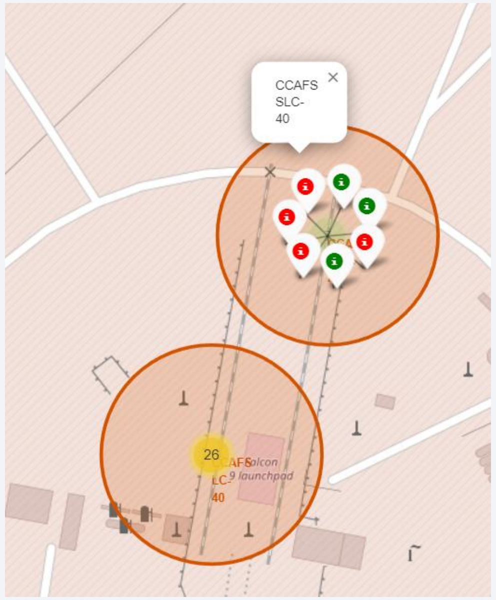
Task 2: Mark the success/failed launches for each site on the map

- We are able to easily see which launch sites have relatively high success rates from the color-labeled markers
  - KSC LC-39A has the highest success rate



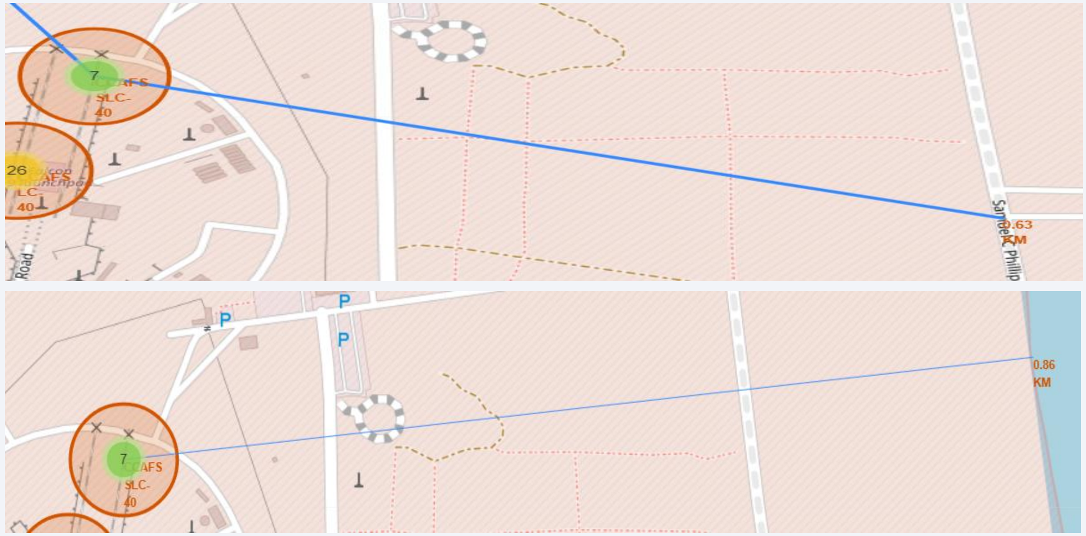


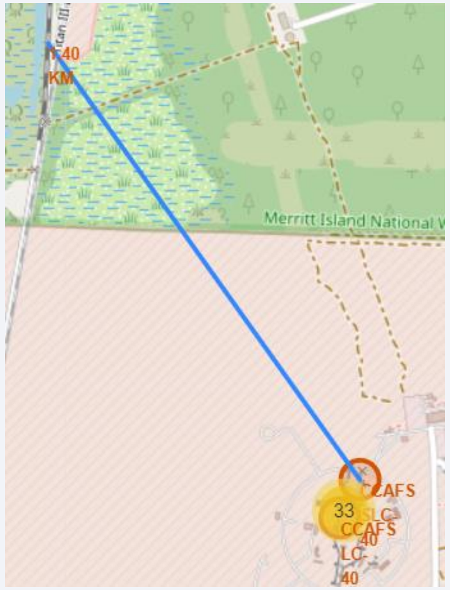


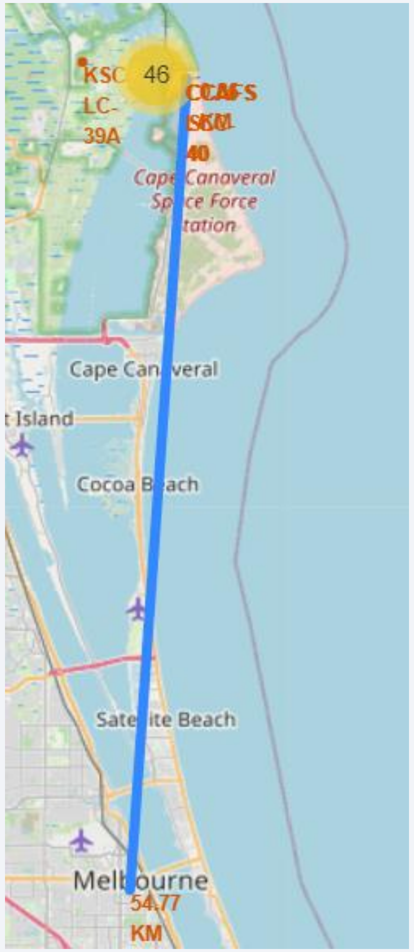


TASK 3: Calculate the distances between a launch site to its proximities

- CCAFS SLC-40 is in close proximity to railways, highways, and coastline
  - railways (1.4km), highways (0.63km), coastline (0.86km)
- CCAFS SLC-40 keeps distance away from cities
  - 54.77km away from Melbourne





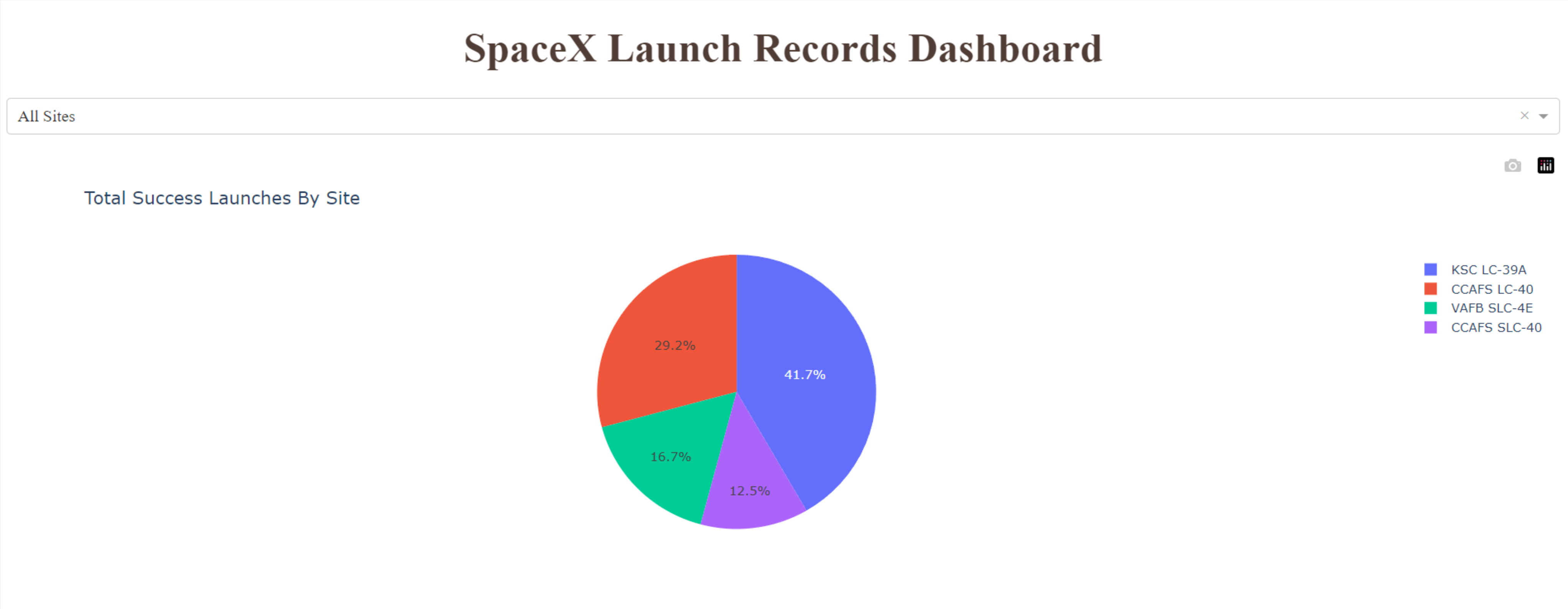


Section 4

Build a Dashboard with Plotly Dash

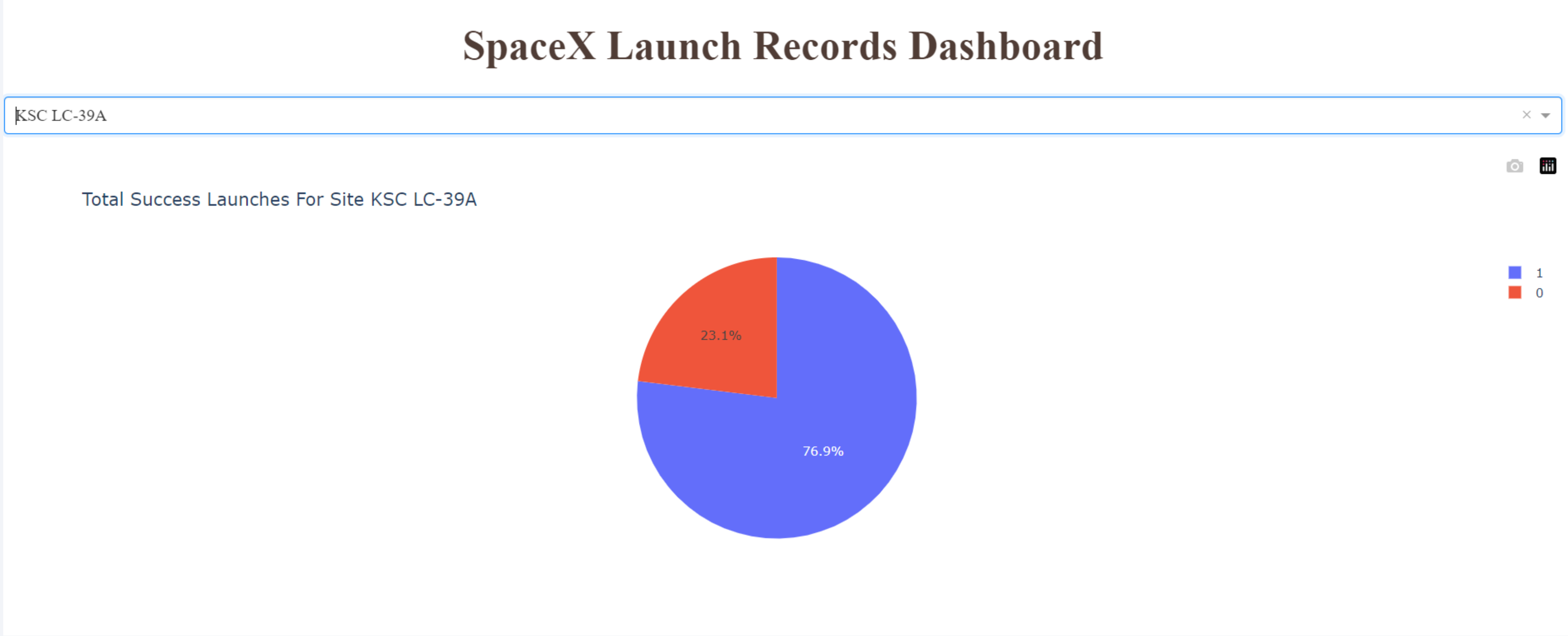
Launch Success Count for All Sites

- We see that KSC LC-39A has the largest successful launches (41.7%)



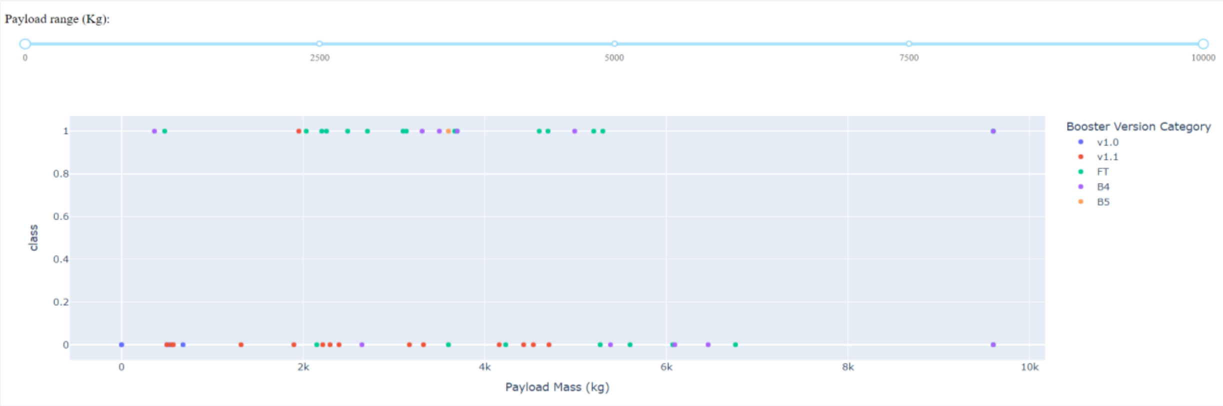
Total Success Launches for the Site

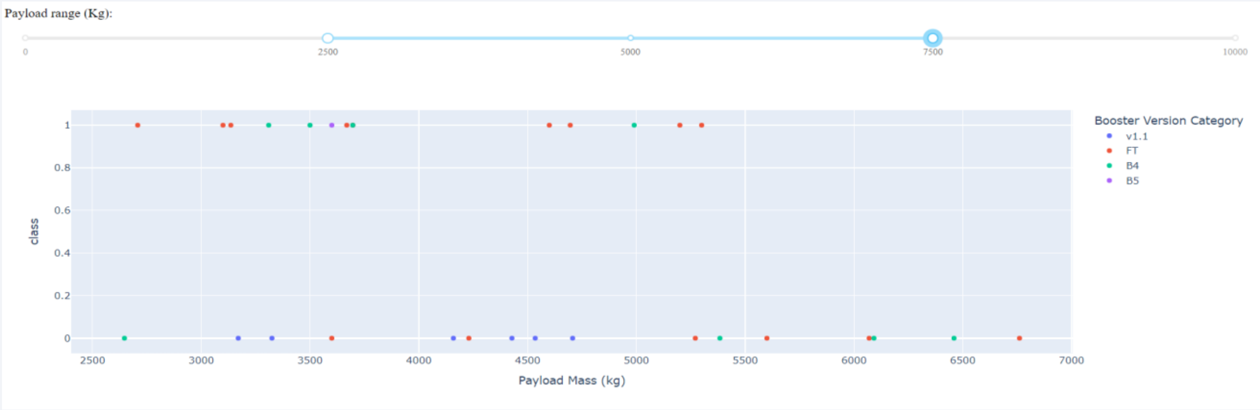
- KSC LC-39A has the highest launch success rate (76.9%)
  - CCAFS LC-40 (26.9%), VAFB SLC-4E (40%), CCAFS SLC-40 (42.9%)



Payload vs. Launch Outcome

- Payload range between 2k and 4k has the highest launch success rate
- Payload range between 8k and 10k has the lowest launch success rate
- Booster Version FT has the highest launch success rate



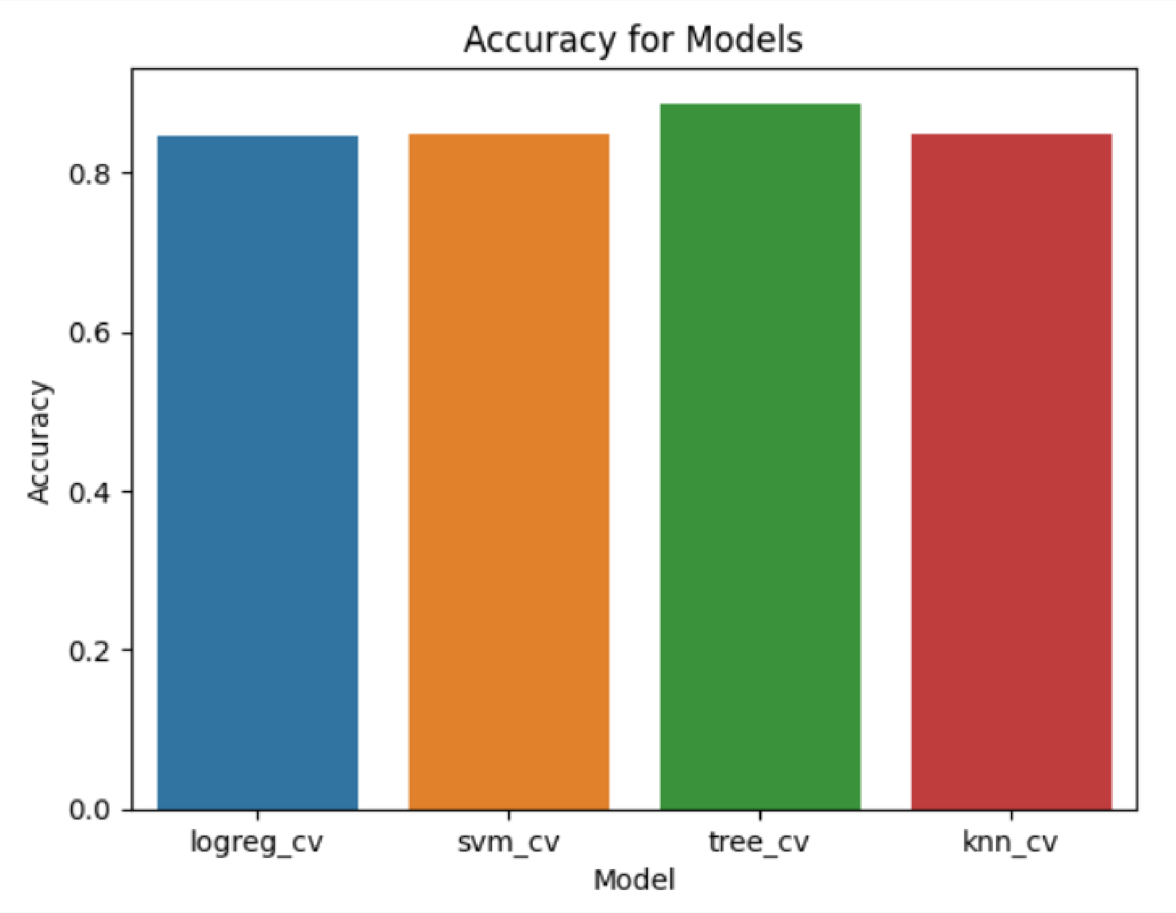


Section 5

Predictive Analysis (Classification)

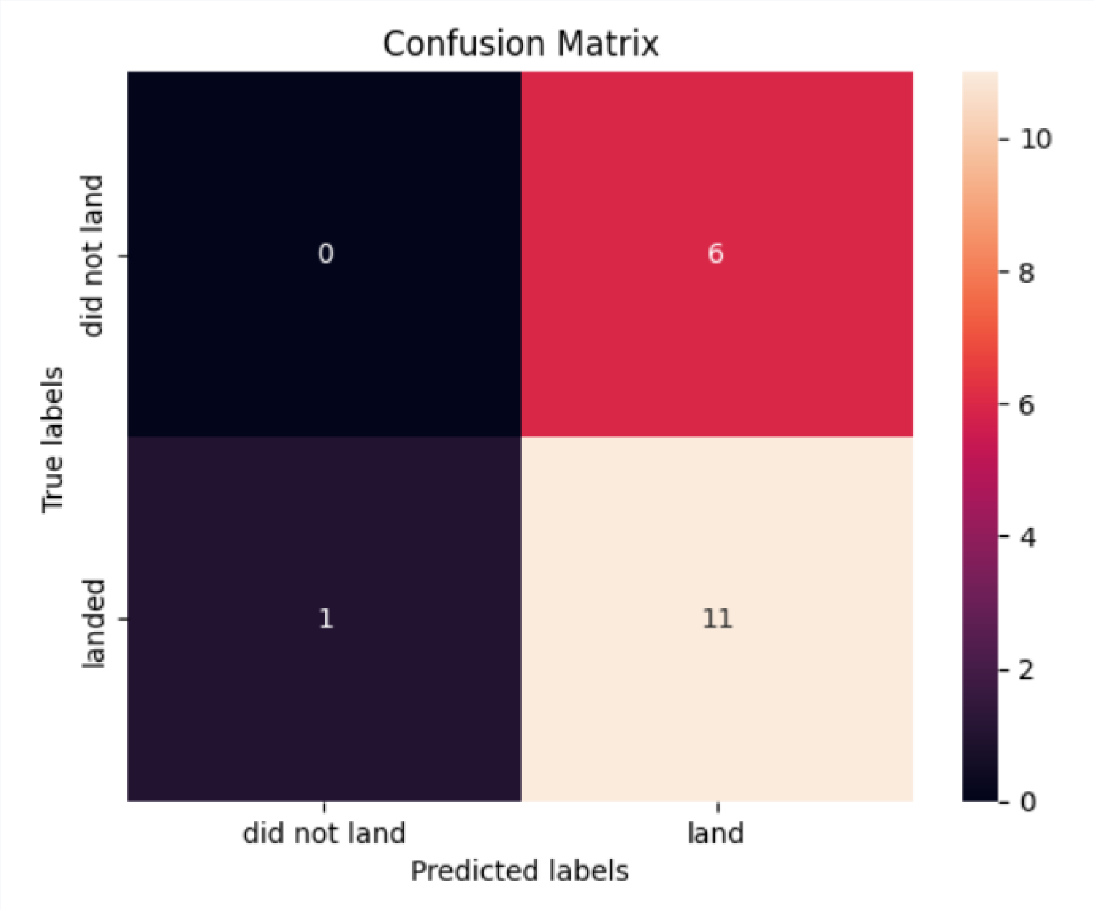
Classification Accuracy

- Decision Tree Model has the highest classification accuracy (0.8875)
  - Logistic Regression Model (0.8464)
  - Support Vector Machine(SVM) Model (0.8482)
  - K-Nearest Neighbors (KNN) Model (0.8482)



Confusion Matrix

- The best performing model is Decision Tree Model
- We see that Decision Tree Model can distinguish between the different classes
- When Falcon 9 landed successfully, this model could not predict correctly once out of 12 test sets (False Negative problem)



Conclusions

- We found some patterns in the data using visualization and SQL
  - KSC LC-39A has the highest success rate among all the launch sites during whole period
  - CCAFS SLC 40 has the highest success rate for recent attempts
  - The Falcon 9 first stage is more likely to land successfully when its payload is between 2k and 4k
  - Booster Version FT is more likely to land successfully
  - In Orbit type 'SO', The Falcon 9 first stage never landed successfully
- Decision Tree Model predicts if the Falcon 9 first stage will land successfully with the highest accuracy

Thank you!