IBM Developer
SKILLS NETWORK

Data Science Professional Certificate Capstone Project

Hayford Osumanu

February 2023

OUTLINE

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

FALCON SPACEX

IBM Developer SKILLS NETWORK

EXECUTIVE SUMMARY

The primary objective of this Capstone Project is to predict if SpaceX Falcon 9 first stage will land successfully using Machine Learning Classification Algorithms. The goal is to analyze Space X Falcon 9 dataset collected from various to help provide insight for decision making.

In order to achieve the above-mentioned objective, the analyst collected using API and Web Scrapping. The data was transformed using Data Wrangling techniques.

Both descriptive and inferential data analysis was used to help understand and infer from the data. The main analytical tools used for the descriptive analysis was SQL and Python. Interactive visuals such as folium maps, dashboard plotly was built to help analyze launch site proximity. Besides, Machine Learning Classification algorithm was used to predict if the first stage of Falcon 9 will land successfully.

From the analysis, it was observed that, variable such as launch site have a significant impact with the outcome of the launches (success or failure).

Besides, Launch Site ‘KSC LC-39A’ recorded the highest launch success rate whereas Launch Site ‘CCAFS SLC- 40’ recorded the lowest launch success rate.

In addition, Orbits ES-L1, GEO, HEO, and SSO recorded the highest launch success rates whereas orbit GTO the least. Strategically, Lunch sites are located away from the cities and much closer to coastline, railroads, and highways.

More importantly, the best performing Machine Learning Classification Model was the Decision Tree with an accuracy of about 87.5%. When the models were tested on the test data, the accuracy score was about 83% for all models. Perhaps, more data may be needed to further tune the models and find a potential better fit or reduce curse of dimensionality.

This and more other interesting observations and discussions have been exhausted in this report.

INTRODUCTION

SpaceX advertises Falcon 9 rocket launches on its website with a cost of 62 million dollars; other providers cost upward of 165 million dollars each, much of the savings is because SpaceX can reuse the first stage.

Therefore, if we can determine if the first stage will land, we can determine the cost of a launch. This information can be used if an alternate company wants to bid against SpaceX for a rocket launch.

The goal of this project is to:

- Write Python code to manipulate data in a Pandas data frame
- Convert a JSON file into a Create a Python Pandas data frame by converting a JSON file
- Create a Jupyter notebook and make it sharable using GitHub
- Use data science methodologies to define and formulate a real-world business problem.
- Use your data analysis tools to load a dataset, clean it, and find out interesting insights from it.

Data Collection and ML Tools

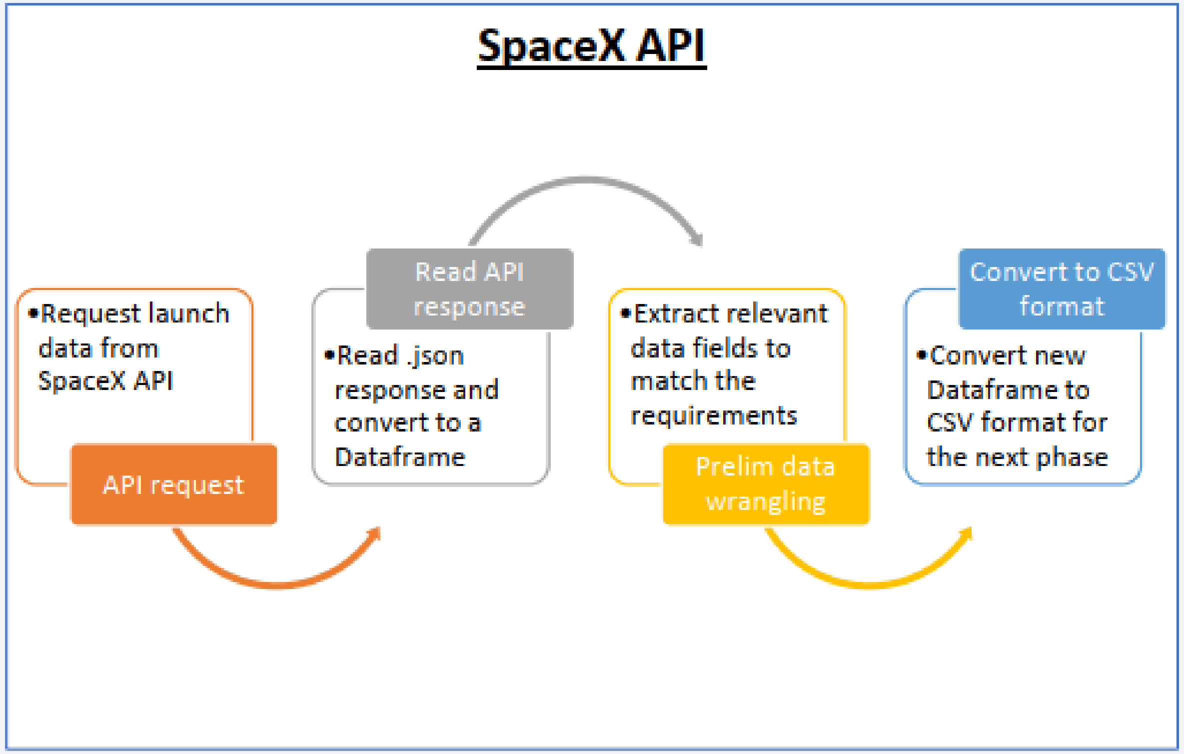
The methods used in data collection, descriptive and inferential analysis are follows:

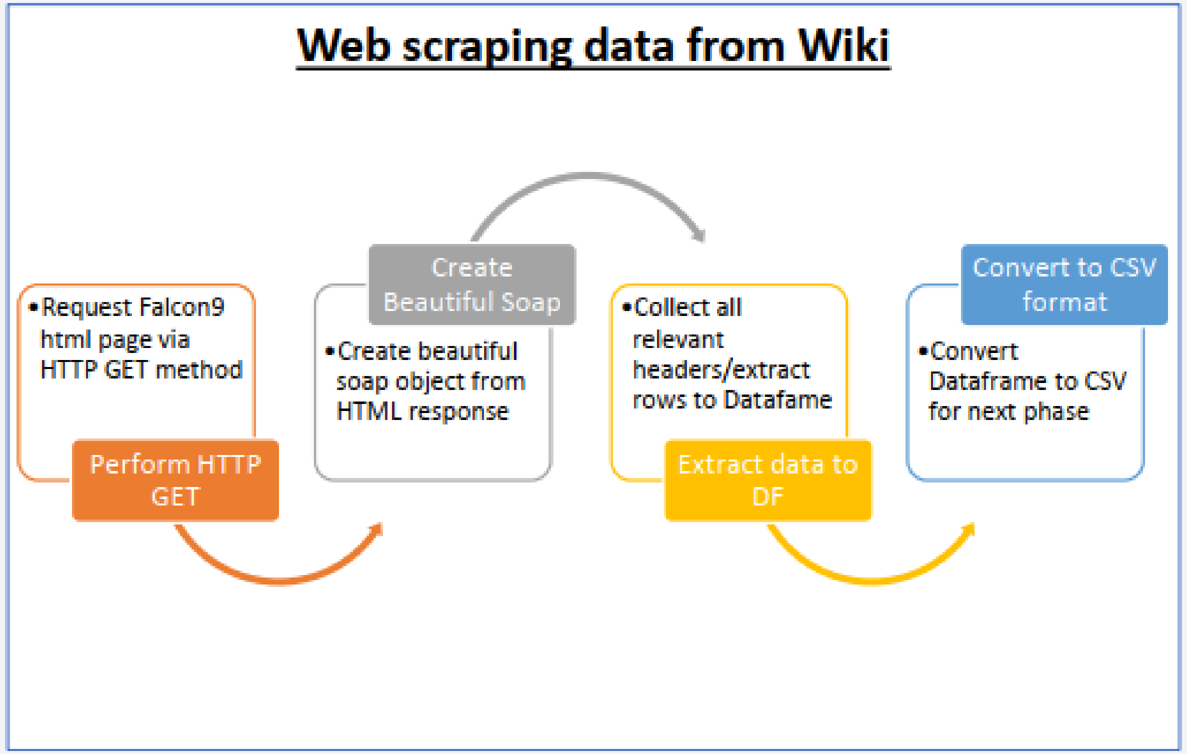
1. **Data collection, wrangling, and formatting**
    - SpaceX API
    - Web scraping
2. **Exploratory data analysis (EDA)**
    - Pandas and NumPy
    - SQL
3. **Data visualization**
    - Matplotlib and Seaborn
    - Folium
    - Dash
4. **Machine learning prediction**
    - Logistic regression
    - Support vector machine (SVM)
    - Decision tree
    - K-nearest neighbors (KNN)

METHODOLOGY

1 Data collection, wrangling, and formatting

- Data collection is the process of gathering data from available sources. This data can be structured, unstructured, or semi-structured. For this project, data was collected via SpaceX API and Web scrapping Wiki pages for relevant launch data.





METHODOLOGY

2 Data collection, wrangling, and formatting

1. API Request and read response into DF
2. Declare global variables
3. Call helper functions with API calls to populate global vars
4. Construct data using dictionary
5. Convert Dict to Dataframe, filter for Falcon9 launches, covert to CSV

1. Create API GET request, normalize data and read in to a Dataframe:

```
spacex_url="https://api.spacexdata.com/v4/launches/past"
```

```
response = requests.get(spacex_url)
```

```
# Use json_normalize meethod to convert the json
data = pd.json_normalize(response.json())
```

2. Declare global variable lists that will store data returned by helper functions with additional API calls to get relevant data

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

3. Call helper functions to get relevant data where columns have IDs (e.g., rocket column is an identification number)
   - getBoosterVersion(data)
   - getLaunchSite(data)
   - getPayloadData(data)
   - getCoreData(data)

4. Construct dataset from received data & combine columns into a dictionary:

```
launch_dict = {'FlightNumber': list(data['flight_number']),
'Date': list(data['date']),
'BoosterVersion':BoosterVersion,
'PayloadMass':PayloadMass,
'Orbit':Orbit,
'LaunchSite':LaunchSite,
'Outcome':Outcome,
'Flights':Flights,
'GridFins':GridFins,
'Reused':Reused,
'Legs':Legs,
'LandingPad':LandingPad,
'Block':Block,
'ReusedCount':ReusedCount,
'Serial':Serial,
'Longitude': Longitude,
'Latitude': Latitude}
```

4. Create Dataframe from dictionary and filter to keep only the Falcon9 launches:

```
# Create a data from launch_dict
df_launch = pd.DataFrame(launch_dict)
```

```
# Hint data['BoosterVersion']!='Falcon 1'
data_falcon9 = df_launch[df_launch['BoosterVersion']!= 'Falcon 1']
```

```
data_falcon9.to_csv('dataset_part\_1.csv', index=False)
```

METHODOLOGY

3 Data collection, wrangling, and formatting

1. Perform HTTP GET to request HTML page
2. Create Beautiful Soap object
3. Extract column names from HTML table header
4. Create Dictionary with keys from extracted column names
5. Call helper functions to fill up dict with launch records
6. Convert Dictionary to Dataframe

1. Create API GET method to request Falcon9 launch HTML page

```
static_url = "https://en.wikipedia.org/w/index.php?title=List_of_Falcon_9_and_Falcon_Heavy_launches&oldid=1027686922"
```

```
html_data = requests.get(static_url).text
```

2. Create Beautiful Soap object

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

3. Find all the tables on the Wiki page and extract relevant column names from the HTML table header

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

```
column_names = []

# Apply find_all() function with `th` element on firs
# Iterate each th element and apply the provided extr
# Append the Non-empty column name (`if name is not N
colnames = soup.find_all('th')
for x in range (len(colnames)):
    name2 = extract_column_from_header(colnames[x])
    if (name2 is not None and len(name2) > 3):
        column_names.append(name2)
```

4. Create an empty Dictionary with keys from extracted column names:

```
launch_dict= dict.fromkeys(column_names)

# Remove an irrelvant column
del launch_dict['Date and time ( )']

# Let's initial the launch_dict with each va
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']=[]
```

5. Fill up the launch_dict with launch records extracted from table rows.
   - Utilize following helper functions to help parse HTML data

```
def date_time(table_cells):
```

```
def booster_version(table_cells):
```

```
def landing_status(table_cells):
```

```
def get_mass(table_cells):
```

6. Convert launch_dict to Dataframe:

```
df=pd.DataFrame(launch_dict)
```

METHODOLOGY

4 Data collection, wrangling, and formatting

1. Load dataset in to Dataframe → 2. Find patterns in data → 3. Create landing outcome label

1. Load SpaceX dataset (csv) in to a Dataframe

```
df=pd.read_csv("https://cf-courses-data.s3.us.cloud-object-storage.appd
art_1.csv")
```

2. Find data patterns:
   i. Calculate the number of launches on each site

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

```
CCAFS SLC 40    55
KSC LC 39A      22
VAFB SLC 4E     13
```

   ii. Calculate the number and occurrence of each orbit

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

```
GTO      27
ISS      21
VLEO     14
PO        9
LEO       7
SSO       5
MEO       3
GEO       1
HEO       1
SO        1
ES-L1     1
```

   iii. Calculate number/occurrence of mission outcomes per orbit type

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

3. Create a landing outcome label from Outcome column in the Dataframe

```
# landing_class = 0 if bad_outcome
# landing_class = 1 otherwise

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

```
df['Class']=landing_class
df[['Class']].head(8)
```

	Class
0	0
1	0
2	0
3	0
4	0

METHODOLOGY

5 Data collection, wrangling, and formatting

SpaceX API

- The API used: https://api.spacexdata.com/v4/rockets/.
- The API provided data about many types of rocket launches done by SpaceX; the data was therefore filtered to include only Falcon 9 launches.
- Missing value in the dataset was replace by the column mean.
- We ended with 90 rows and 17 columns or features. The table below shows the first few rows of the data:

	FlightNumber	Date	BoosterVersion	PayloadMass	Orbit	LaunchSite	Outcome	Flights	GridFins	Reused	Legs	LandingPad	Block	ReusedCount	Serial	Longitude	Latitude
4	1	2010-06-04	Falcon 9	NaN	LEO	CCSFS SLC 40	None None	1	False	False	False	None	1.0	0	B0003	-80.577366	28.561857
5	2	2012-05-22	Falcon 9	525.0	LEO	CCSFS SLC 40	None None	1	False	False	False	None	1.0	0	B0005	-80.577366	28.561857
6	3	2013-03-01	Falcon 9	677.0	ISS	CCSFS SLC 40	None None	1	False	False	False	None	1.0	0	B0007	-80.577366	28.561857
7	4	2013-09-29	Falcon 9	500.0	PO	VAFB SLC 4E	False Ocean	1	False	False	False	None	1.0	0	B1003	-120.610829	34.632093
8	5	2013-12-03	Falcon 9	3170.0	GTO	CCSFS SLC 40	None None	1	False	False	False	None	1.0	0	B1004	-80.577366	28.561857

METHODOLOGY

6 Data collection, wrangling, and formatting

Web scraping

- The data is scraped from https://en.wikipedia.org/w/index.php?title=List_of_Falcon_9_and_Falcon_Heavy_launches&oldid=1027686922
- The website contains only the data about Falcon 9 launches.
- We ended up with 121 rows and 11 columns or features. The table below shows the first few rows of the dataset:

	Flight No.	Launch site	Payload	Payload mass	Orbit	Customer	Launch outcome	Version Booster	Booster landing	Date	Time
0	1	CCAFS	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success\n	F9 v1.0B0003.1	Failure	4 June 2010	18:45
1	2	CCAFS	Dragon	0	LEO	NASA	Success	F9 v1.0B0004.1	Failure	8 December 2010	15:43
2	3	CCAFS	Dragon	525 kg	LEO	NASA	Success	F9 v1.0B0005.1	No attempt\n	22 May 2012	07:44
3	4	CCAFS	SpaceX CRS-1	4,700 kg	LEO	NASA	Success\n	F9 v1.0B0006.1	No attempt	8 October 2012	00:35
4	5	CCAFS	SpaceX CRS-2	4,877 kg	LEO	NASA	Success\n	F9 v1.0B0007.1	No attempt\n	1 March 2013	15:10

METHODOLOGY

7 Exploratory Data Analysis (EDA)

Pandas and NumPy

Function from Pandas and NumPy libraries were used to obtained basic information such as:

- The number of launches on each launch site
- The number of occurrence of each orbit
- The number and occurrence of each mission outcome

SQL

Data was queried using SQL to answer several questions such as:

- The names of the unique launch sites in the space mission
- The total payload mass carried by boosters launched by NASA (CRS)
- The average payload mass carried by booster version F9 v1.1

METHODOLOGY

8 Data Visualization

Matplotlib and Seaborn

Functions from the Matplotlib and Seaborn libraries were used to visualize the data using scatterplots, bar charts, and line charts to help answer questions such as.

- The relationship between flight number and launch site
- The relationship between payload mass and launch site
- The relationship between success rate and orbit type

Folium

- Functions from the Folium libraries were used to visualize the data through interactive maps to help:
    - Mark all launch sites on a map
    - Mark the succeeded launches and failed launches for each site on the map
    - Mark the distances between a launch site to its proximities such as the nearest city, railway, or highway

METHODOLOGY

9 Data Visualization

Dash

Functions from Dash were used to generate an interactive site where we can toggle the input using a dropdown menu and a range slider. The interactive site shows:

- The total success launches from each launch site
- The correlation between payload mass and mission outcome (success or failure) for each launch site

METHODOLOGY

10 Machine Learning Prediction

Machine Learning Modeling

Functions from the Scikit-learn library were also used to create machine learning models.

The machine learning prediction include the following steps:

- Standardizing the data
- Splitting the data into training and test data
- Creating machine learning models using:
    - Logistic regression
    - Support vector machine (SVM)
    - Decision tree
    - K nearest neighbors (KNN)
- Fit the models on the training set
- Find the best combination of hyperparameters for each model
- Evaluate the models based on their accuracy scores and confusion matrix

RESULTS

The results of the analysis were grouped into 5 parts:

- SQL (EDA with SQL)
- Matplotlib and Seaborn (EDA with Visualization)
- Folium
- Dash
- Predictive Analysis

The target variable was Outcome with class **0** represents a failed launch outcome while class **1** represents a successful launch outcome.

RESULTS

1 SQL (EDA with SQL)

SQL Query Results:

List of unique launch sites in the space mission

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

Five records where launch sites begin with 'CCA'

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	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
2012-10-08	00:35:00	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

RESULTS

2 SQL (EDA with SQL): Payload Mass Carried by Boosters

The total payload mass carried by boosters launched by NASA (CRS)

Total payload mass by NASA (CRS)
45596

The average payload mass carried by booster version F9 v1.1

Average payload mass by Booster Version F9 v1.1
2928

The date when the first successful landing outcome in ground pad was achieved

Date of first successful landing outcome in ground pad
2015-12-22

RESULTS

3 SQL (EDA with SQL): Boosters and Payload Mass

List of boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000

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

The total number of successful and failure mission outcomes

number_of_success_outcomes	number_of_failure_outcomes
100	1

RESULTS

4 SQL (EDA with SQL): Booster Versions with Maximum Payload Mass

List of booster versions which have carried the maximum payload mass

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

RESULTS

5 SQL (EDA with SQL): Failed Landing Outcomes and Dates

Failed landing outcomes in drone ship, their booster versions, and launch site names for in year 2015

DATE	booster_version	launch_site
2015-01-10	F9 v1.1 B1012	CCAFS LC-40
2015-04-14	F9 v1.1 B1015	CCAFS LC-40

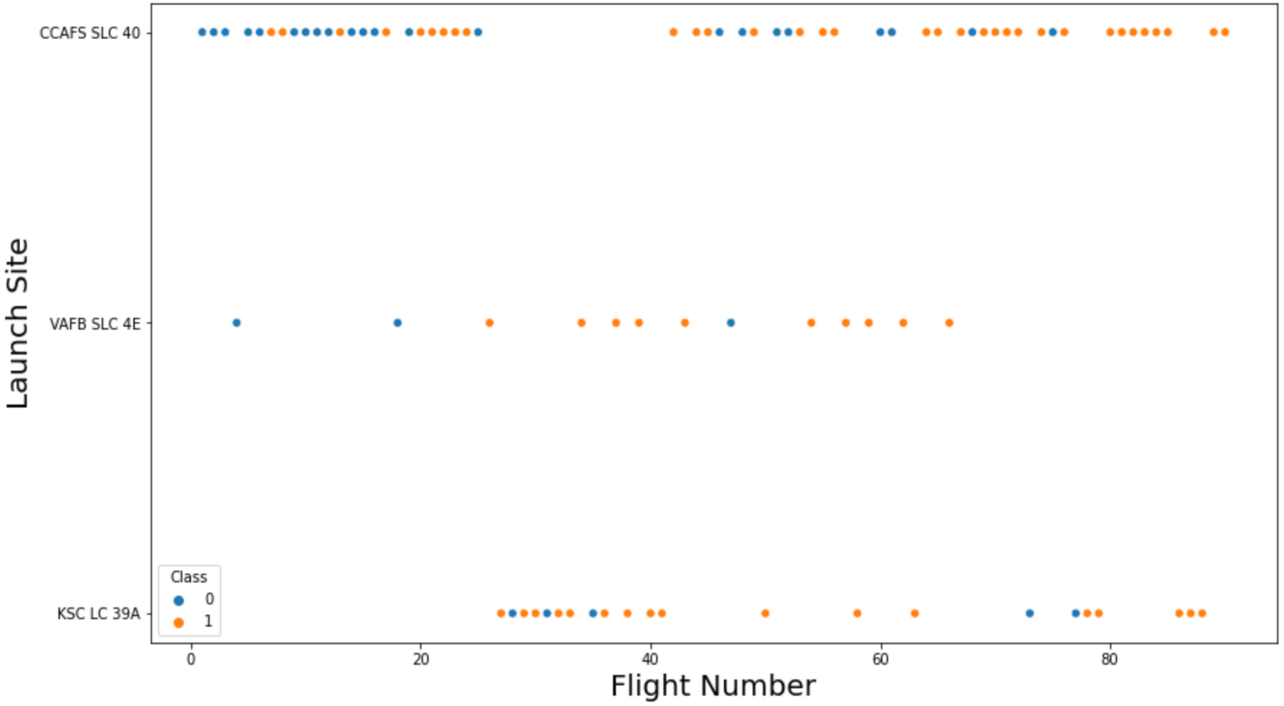
Count of landing outcomes between the date 2010-06-04 and 2017-03-20, in descending order

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

RESULTS

6 **EDA Visualization: Flight Number and Launch Site**

The relationship between flight number and launch site

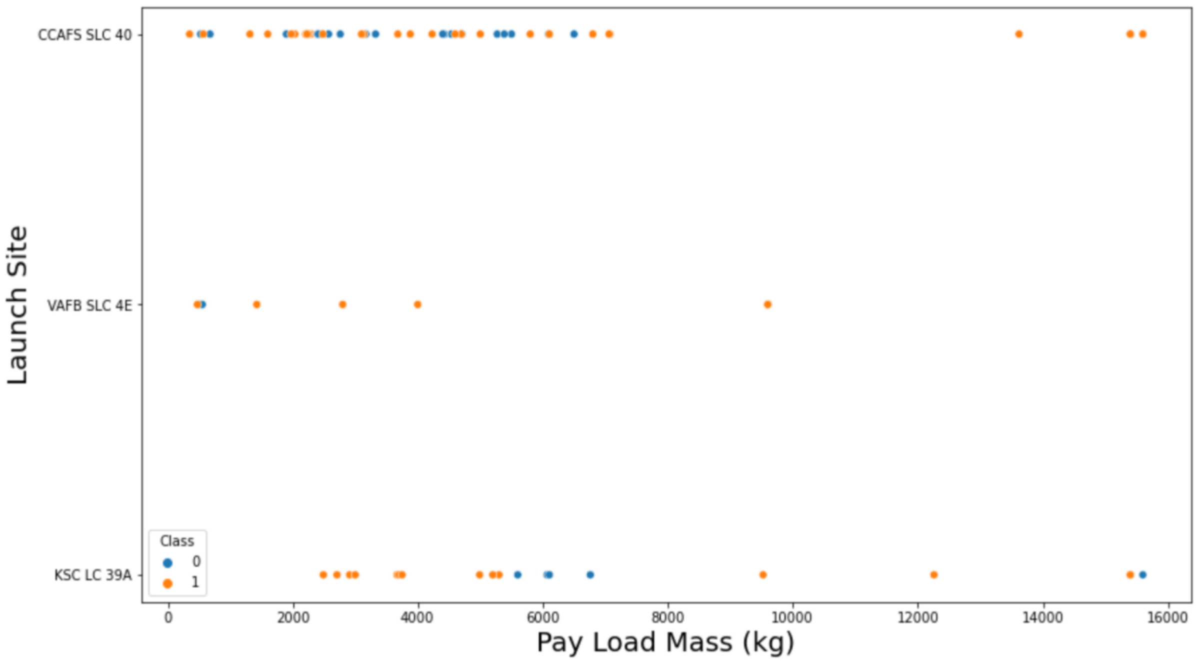


- **Observation**
  - Success rates (***Class=1***) increases as the number of flights increase
  - For launch site 'KSC LC 39A', it takes at least approximately 25 launches before a first successful launch.

RESULTS

7 EDA with Visualization: Relationship Between Payload Mass and Launch Site

- The relationship between payload mass and launch site

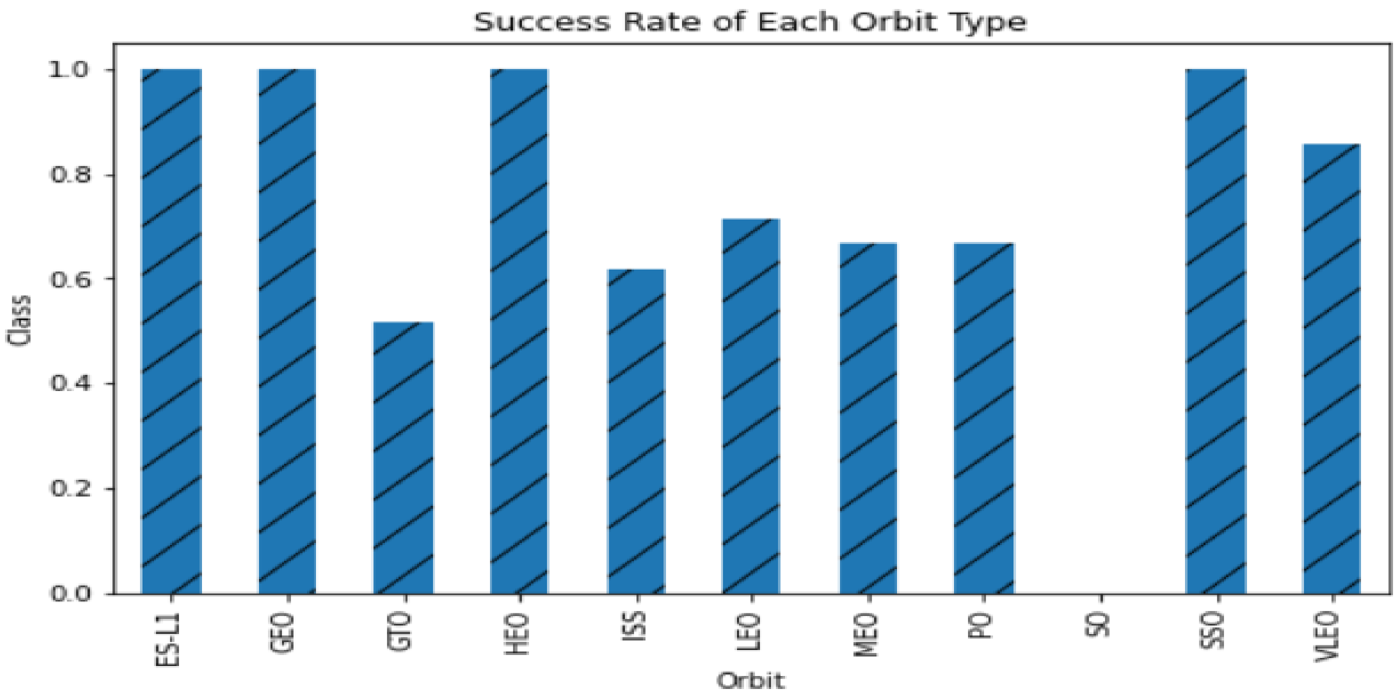


- **Observation**
  - For launch site 'VAFB SLC 4E', it appears there is no rockets launched for payload greater than 10,000 kg
  - The proportion of successful launch (***Class=1***) increases for launch site 'VAFB SLC 4E' as the payload mass increases
  - There is little or no correlation or pattern between launch site and payload mass

RESULTS

8 **EDA with Visualization: Success Rate and Orbit Type**

Relationship between success rate and orbit type

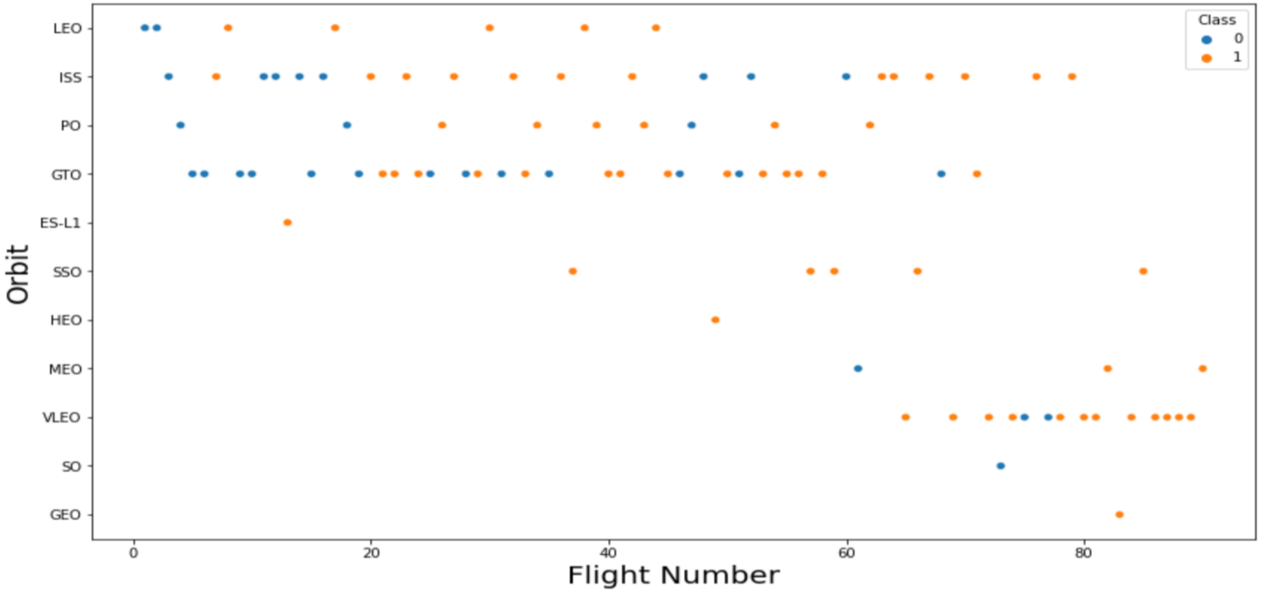


- **Observation**
  - Orbits ES-LI, GEO, HEO, and SSO have the highest success rates whereas GTO orbit has the least success rate.

RESULTS

2 EDA with Visualization: Flight Number and Orbit Type

Relationship between flight number and orbit type

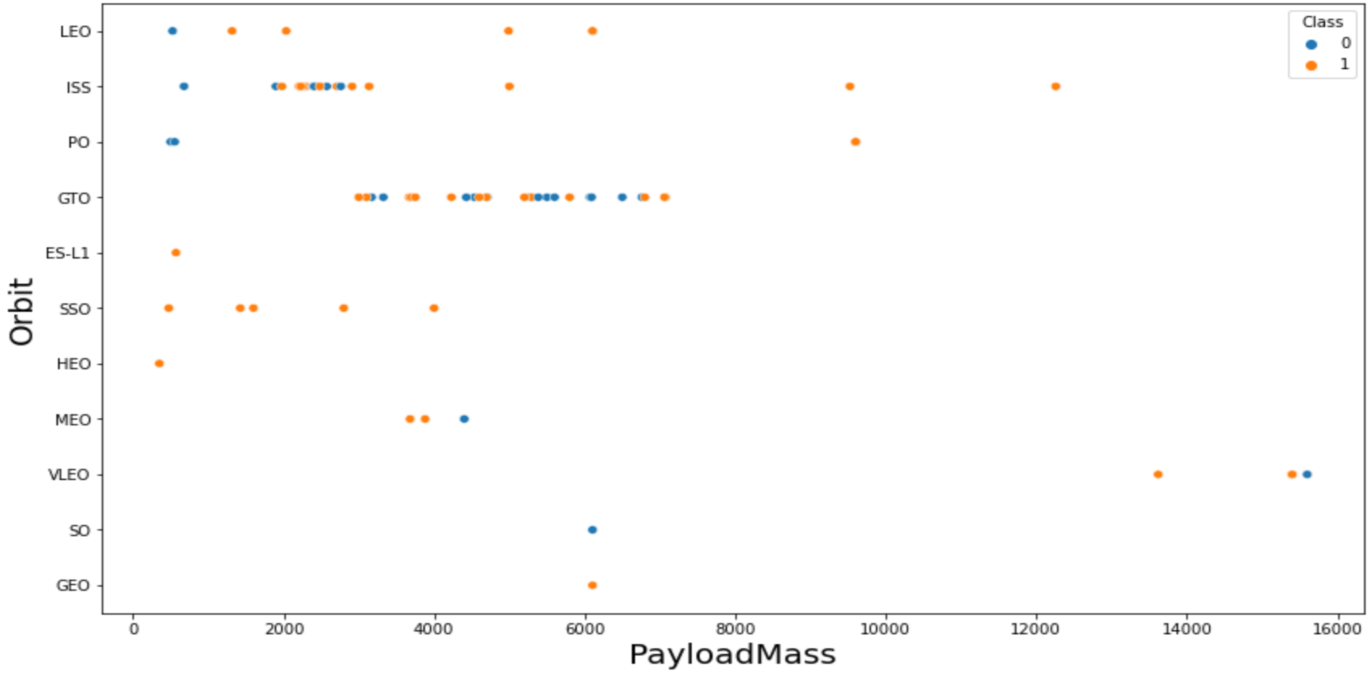


- **Observation**
  - Orbit VLEO first successful landing (***class=1***) doesn't occur until 60+ number of flights
  - For most orbits (LEO, ISS, PO, SSO, MEO, VLEO) successful landing rates appear to increase with flight numbers
  - It looks like there is no relationship between flight number and orbit for GTO

RESULTS

9 EDA with Visualization: Payload Mass and Orbit Type

Relationship between payload mass and orbit type

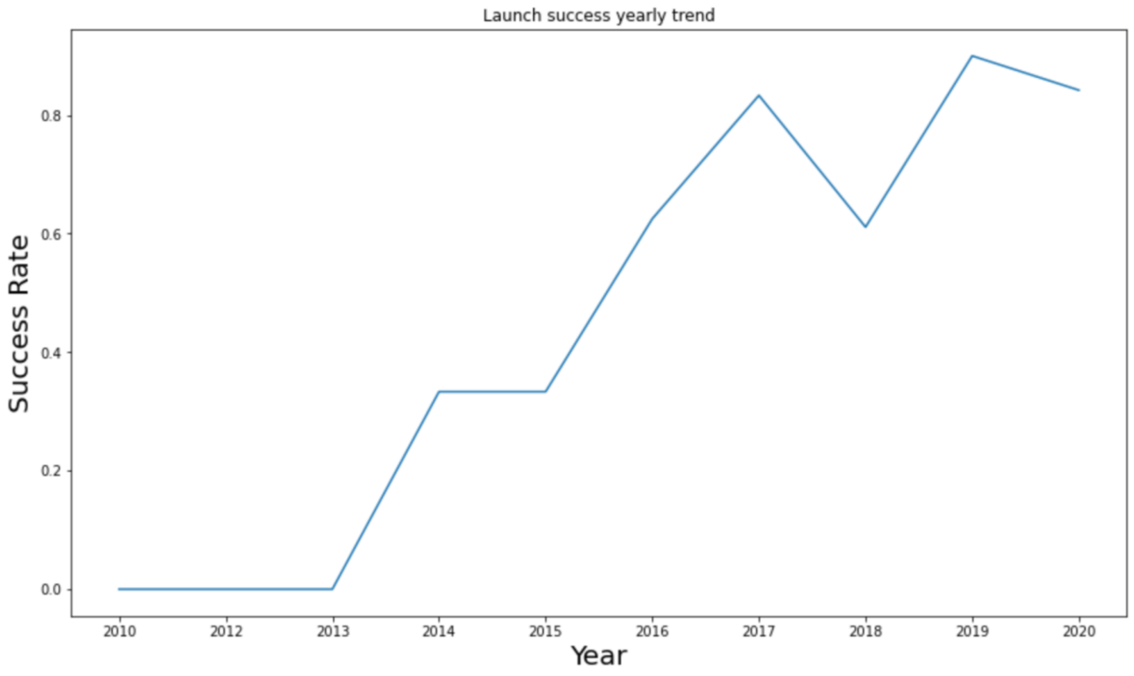


- **Observation**
  - Successful landing rates (***Class=1***) appears to be slightly increased with pay load for orbits LEO, ISS, PO, and SSO
  - For GEO orbit, there is no clear pattern between payload and orbit for successful or unsuccessful landing

RESULTS

10 EDA with Visualization: Launch success yearly trend (2010-2020)

- The launch success yearly trend

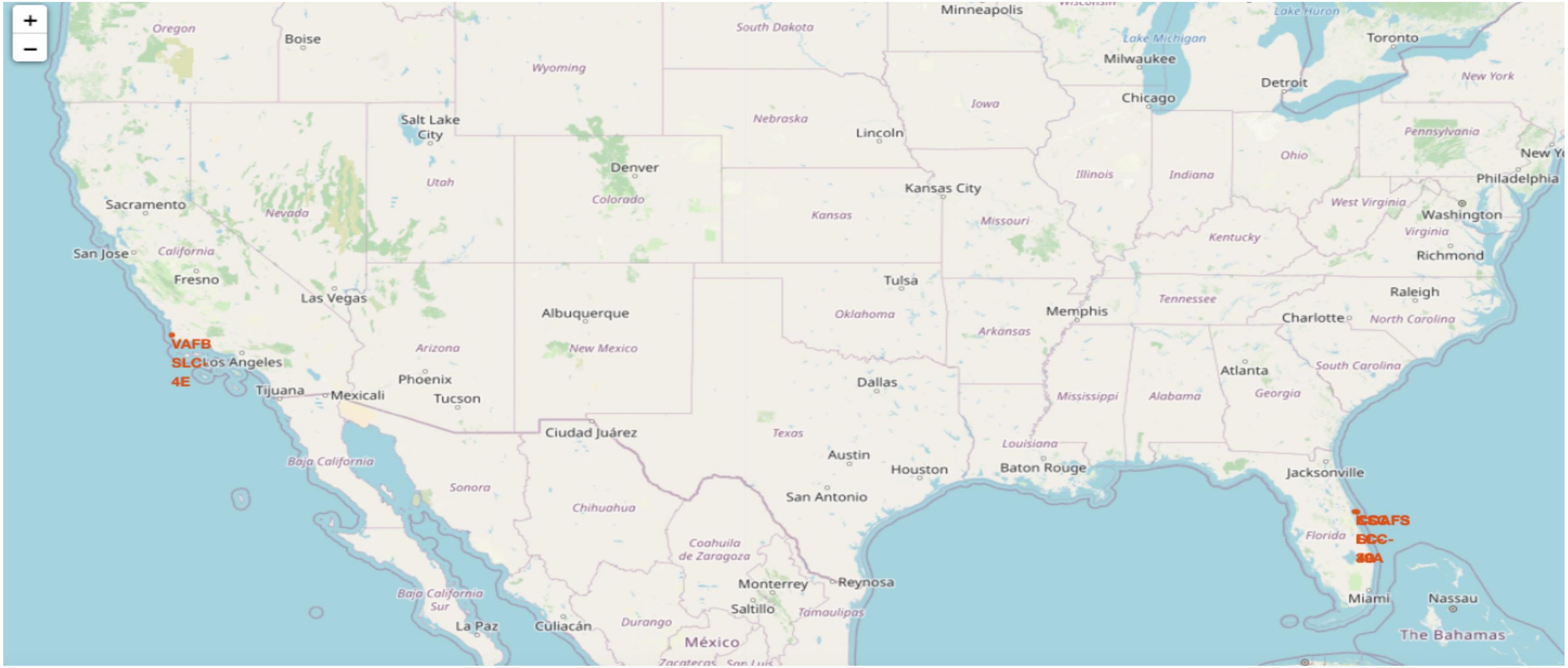


- **Observation**
  - Success rate (***Class=1***) increased by about 80% between 2013 and 2020
  - Success rates was virtually the same between 2010 to 2013 and between 2014 to 2015.
  - Success rates decreased between 2017 to 2018 and between 2019 to 2020, respectively.
  - There is a linear (***upward trend***) relationship between success rate and year of launch.

RESULTS

① Folium: Map of All Space X Falcon 9 Launch Sites

All launch sites on map



RESULTS

2 Folium: Map of All Space X Falcon 9 Launch Sites

- All launch sites on map

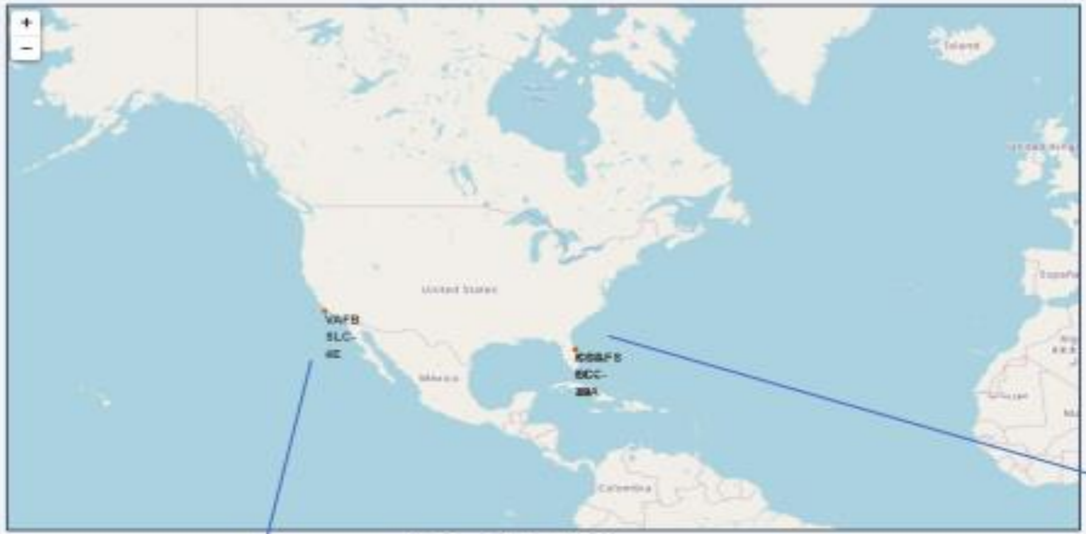


Fig 1 – Global Map

Figure 1 on left displays the Global map with Falcon 9 launch sites that are located in the United States (in California and Florida). Each launch site contains a circle, label, and a popup to highlight the location and the name of the launch site. It is also evident that all launch sites are near the coast.

Figure 2 and Figure 3 zoom in to the launch sites to display 4 launch sites:

- VAFB SLC-4E (CA)
- CCAFS LC-40 (FL)
- KSC LC-39A (FL)
- CCAFS SLC-40 (FL)

Fig 2 – Zoom 1

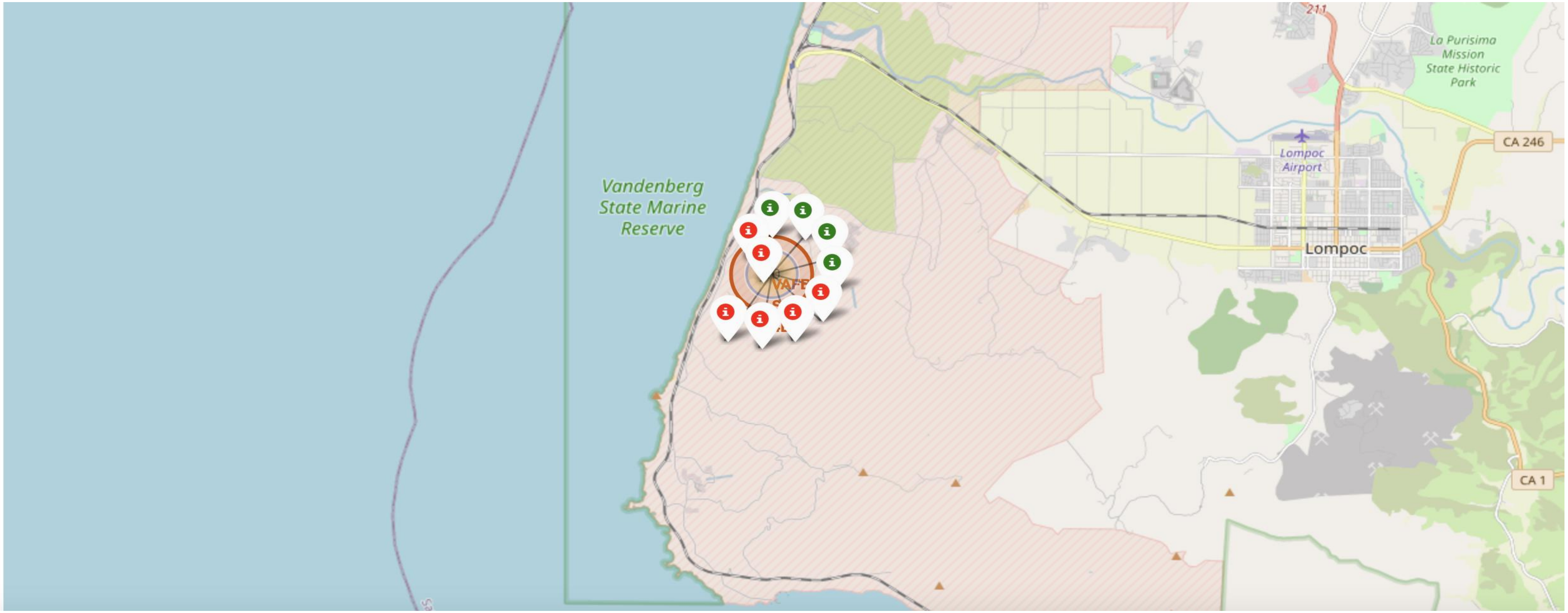
Fig 3 – Zoom 2

RESULTS

3 Folium: Map of Successful and Failed Launch Sites

The succeeded launches and failed launches for each site on map

- If we zoom in on one of the launch site, we can see green and red tags. Each green tag represents a successful launch while each red tag represents a failed launch



RESULTS

4 Folium: Map of Successful and Failed Launch Sites

Fig 1 – US map with all Launch Sites

- **Observation**
  - Figure 1 shows the US map with all the Launch Sites. The numbers on each site displays the total number of successful and failed launches.
  - Figure 2, 3, 4, and 5 below shows a zoom picture of each site and displays the success/fail markers with green as success and red as failed
  - KSC LC-39A Launch Site has the greatest number of successful launches

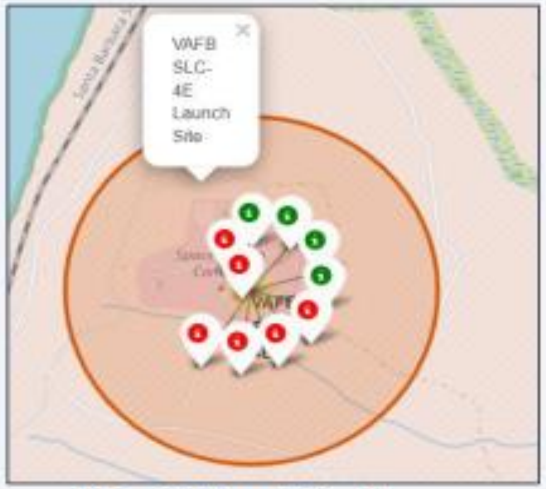


Fig 2 – VAFB Launch Site with success/failed markers

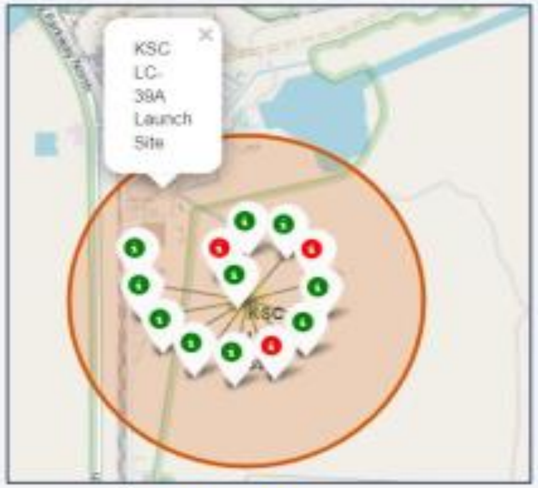


Fig 3 – KSC LC-39A success/failed markers

Fig 4 – CCAFS SLC-40 success/failed markers

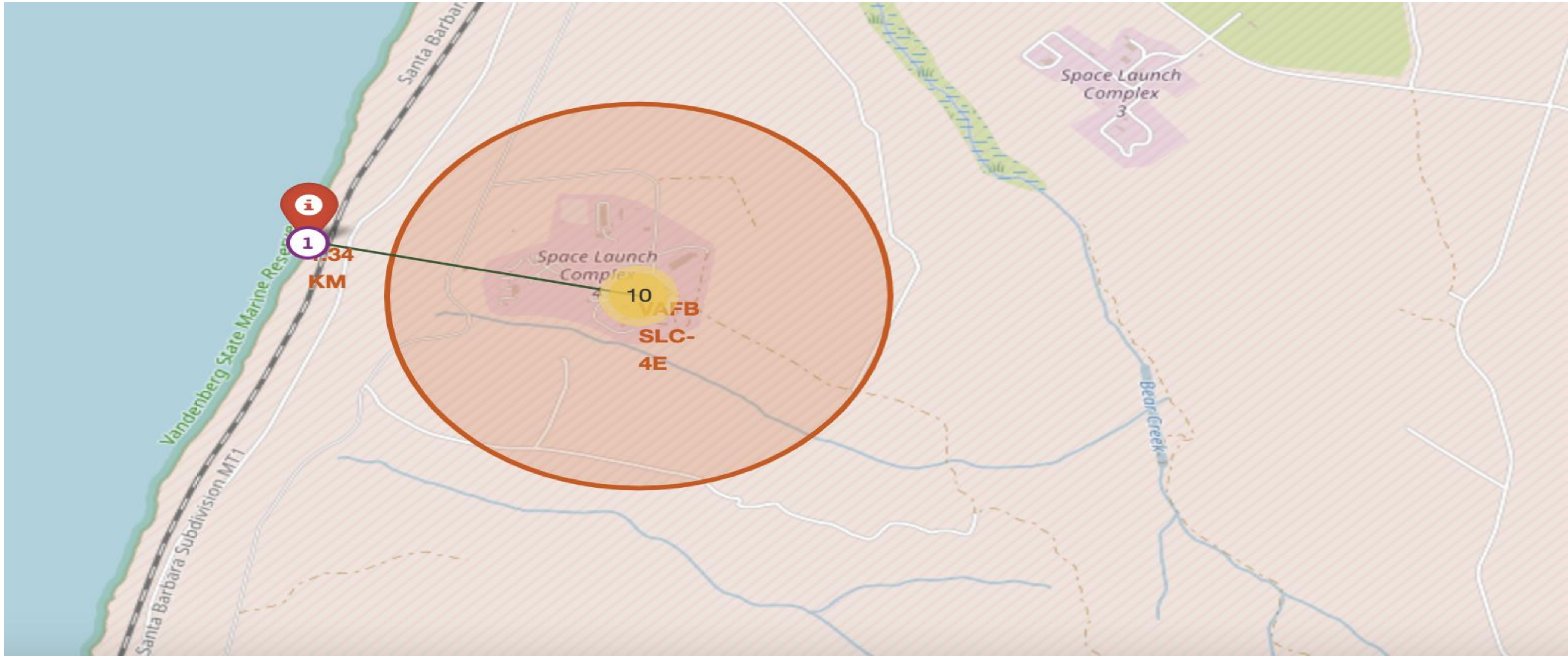
Fig 5 – CCAFS SLC-40 success/failed markers

RESULTS

5 Folium: Distances Between a Launch Site and its Proximities to City, Railway, or Highway

Distances between a launch site to its proximities such as the nearest city, railway, or highway

- The picture below shows the distance between the VAFB SLC-4E launch site and the nearest coastline



RESULTS

5 Folium: Distances Between a Launch Site and its Proximities to City, Railway, or Highway

Fig 1 – Proximity site map for VAFB SLC-4E

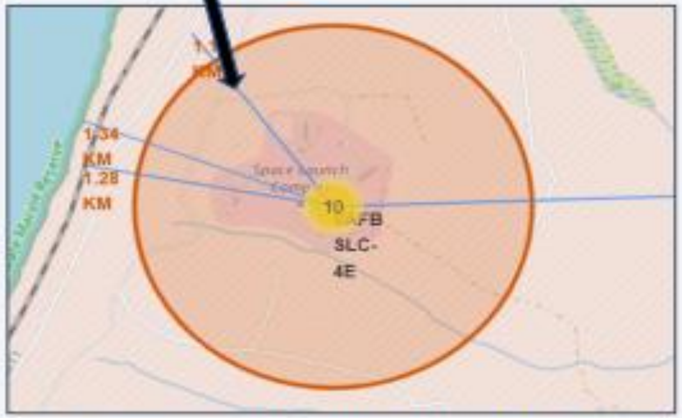


Fig 2 – Zoom in for sites – coastline, railroad, and highway

Observation

- Figure 1 shows all the proximity sites marked on the map for Launch Site VAFB SLC-4E. City Lompoc is located further away from Launch Site compared to other proximities such as coastline, railroad, highway, etc. The map also exhibit a marker with city distance from the Launch Site (14.09 km)
- Figure 2 also provides a zoom in view into other proximities such as coastline, railroad, and highway with respective distances from the Launch Site
- Cities or residential areas are located away from the Launch Sites to avoid the impacts of any accidental to the general public and infrastructure. Usually, Launch Sites are strategically located near the coastline, railroad, and highways to provide easy access to resources

RESULTS

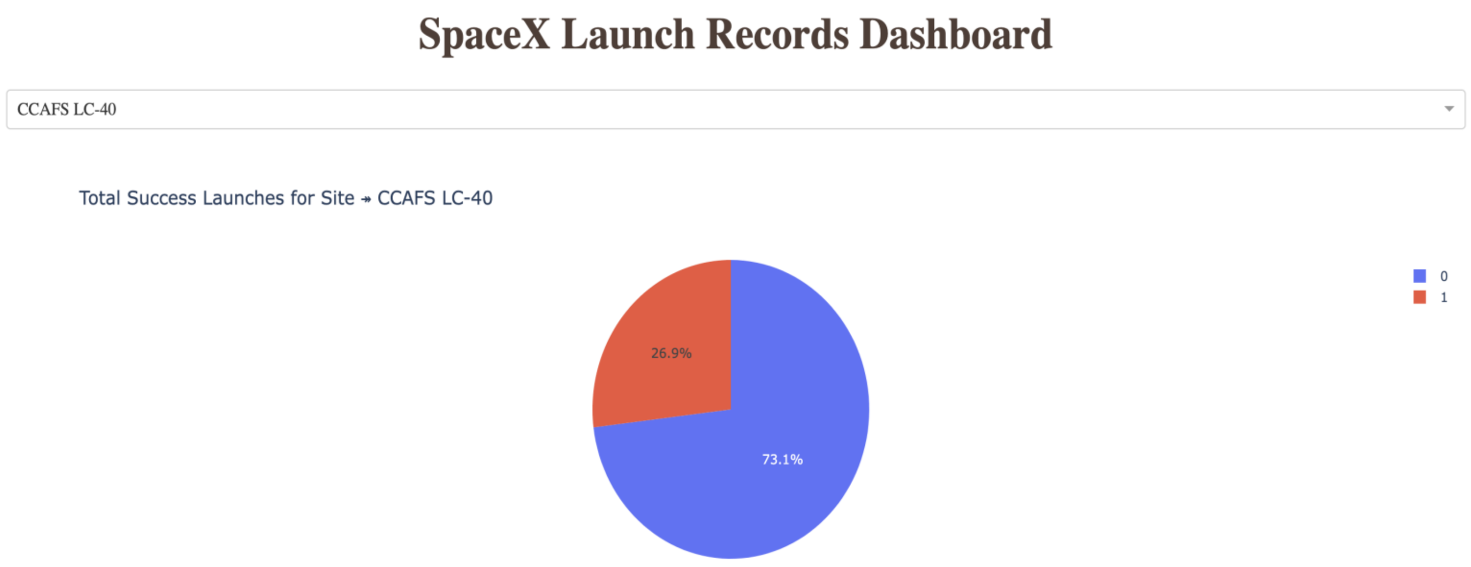
1 Dash: Launch Success Counts for all Sites

- **Observation**
  - Comparatively, Launch Site ‘KSC LC-39A’ has the highest launch success rate
  - Launch Site ‘CCAFS SLC- 40’ has the least launch success rate

RESULTS

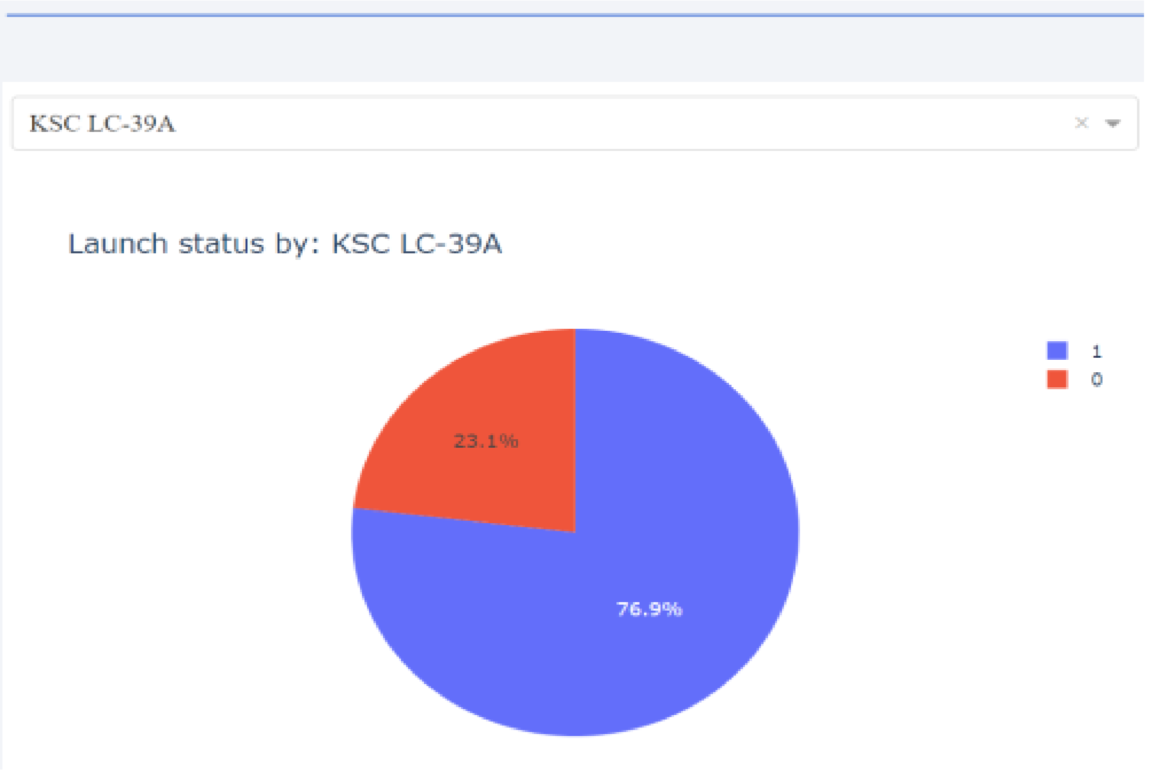
2 Dash : Successful Count for Launch site CCAFS LC-40

Launch site CCAFS LC-40 : 73.1% of launches done at CCAFS LC-40 was successful.



RESULTS

3 Dash : Successful Count for Launch site KSC LC-39A Lauch Site



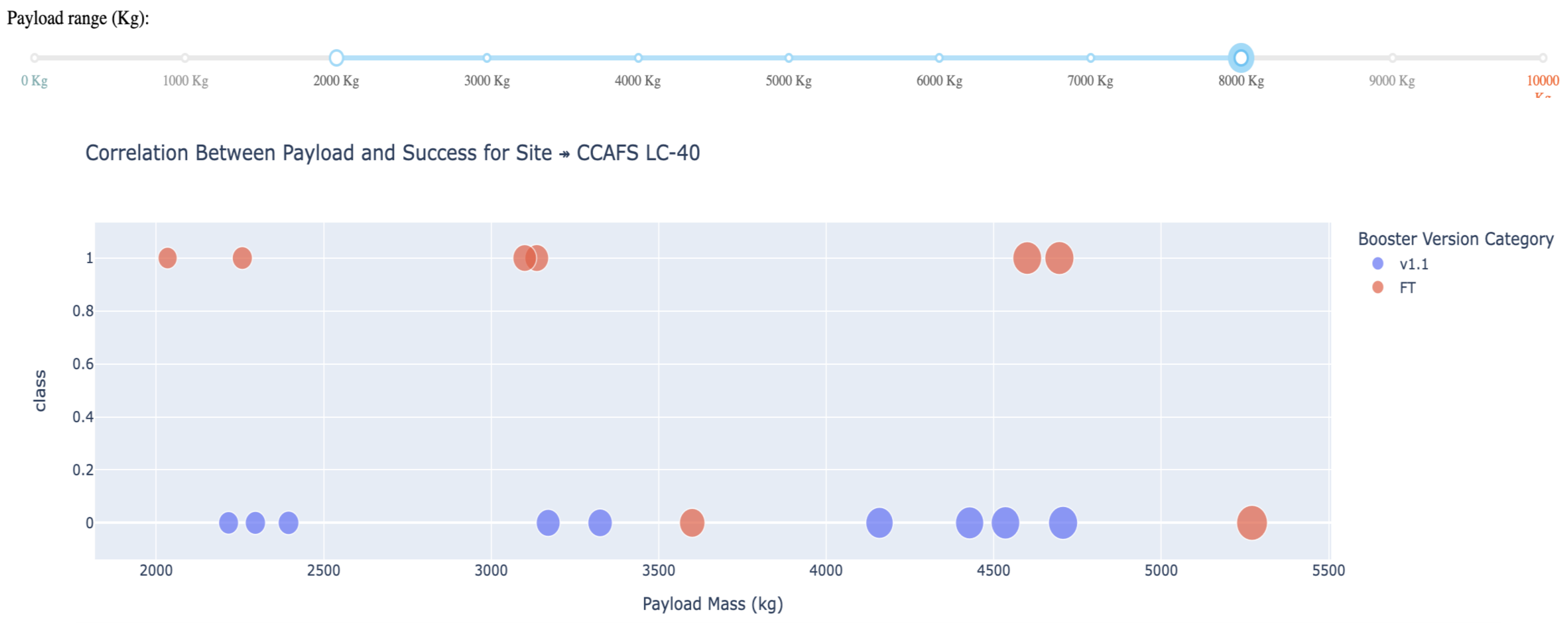
Observation

- Comparatively, KSC LC-39A Launch Site has the highest launch success rate and count
- Launch success rate is approximately 76.9%
- Launch success failure rate is about 23.1%

RESULTS

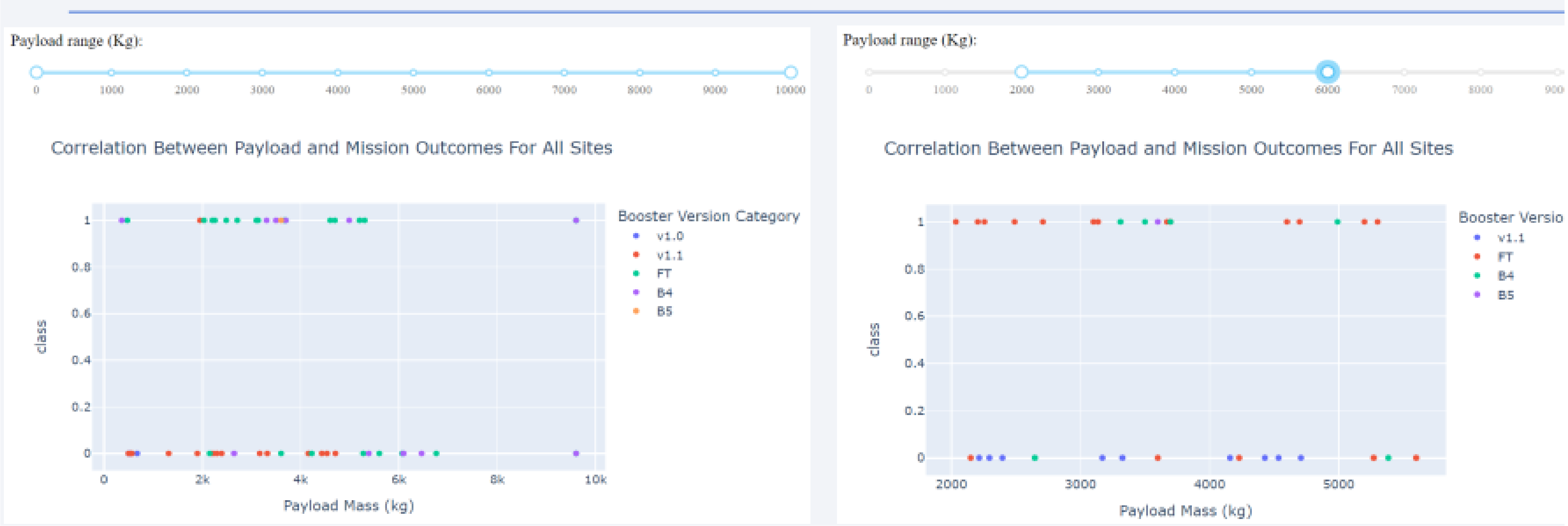
4 Dash: Scatterplot of Payload Mass Between 2000kg and 8000kg

The picture below shows a scatterplot when the payload mass range is set to be from 2000kg to 8000kg.



RESULTS

4 Dash: Scatterplot of All and Payload Mass Between 2000kg and 8000kg



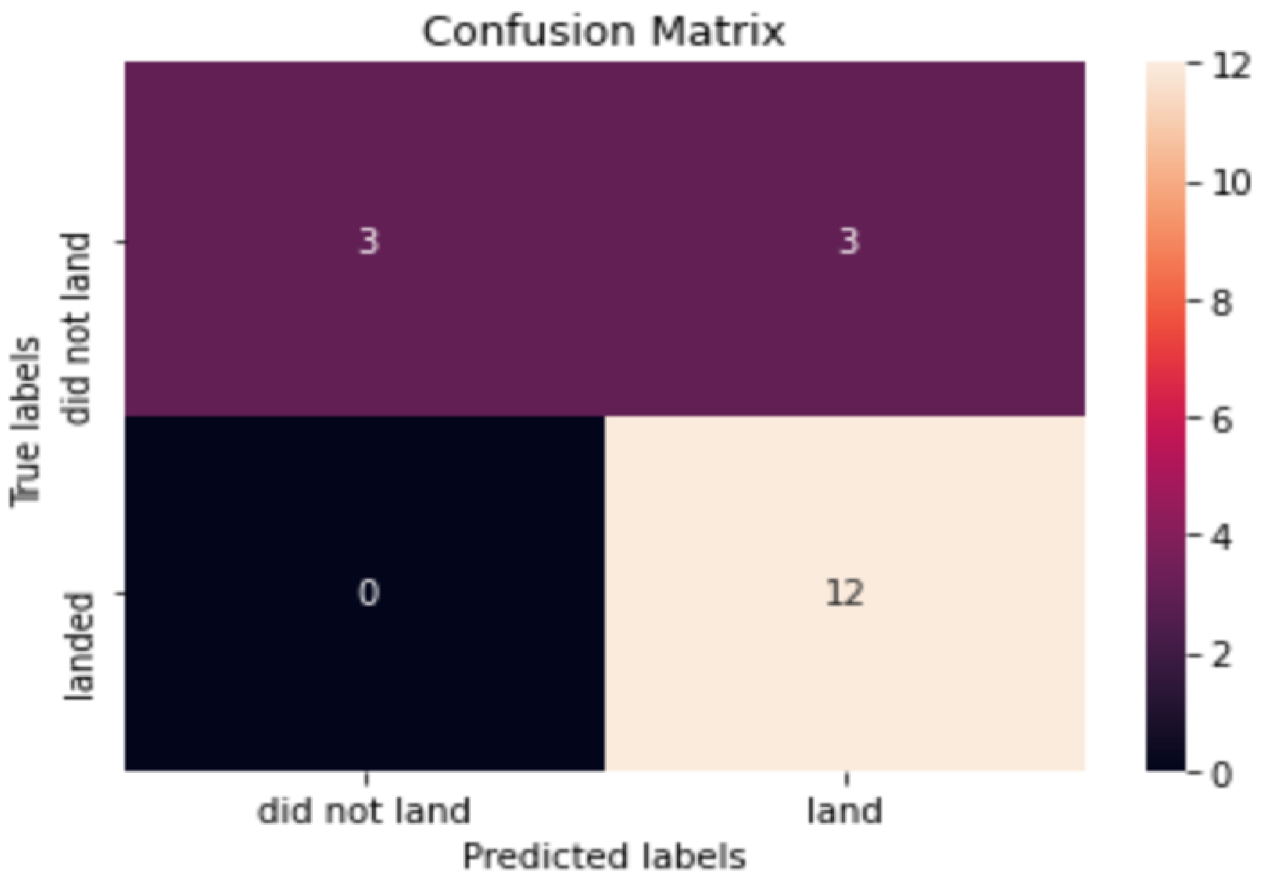
Observation

- Most successful launches are in the payload range from 2000 to about 5500
- Booster version category ‘FT’ has the most successful launches
- Only booster with a success launch when payload is greater than 6k is ‘B4

RESULTS

1 Predictive Analysis: Logistic Regression GridSearchCV Evaluation Metrics

- **Logistic regression**
  - GridSearchCV best score: **0.846**
  - Accuracy score on test set: **0.833**

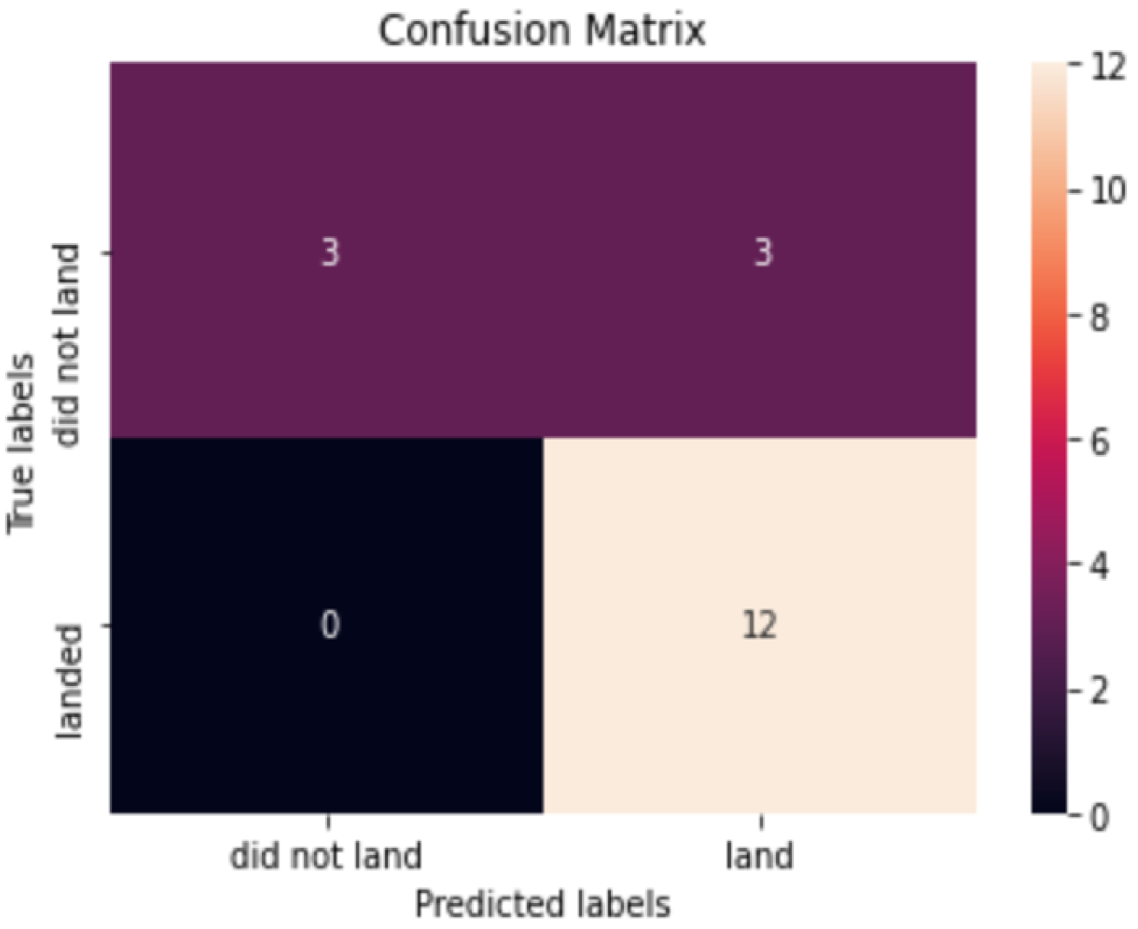


RESULTS

2 Predictive Analysis: Support Vector Machine(SVM) Evaluation Metrics

Support vector machine (SVM)

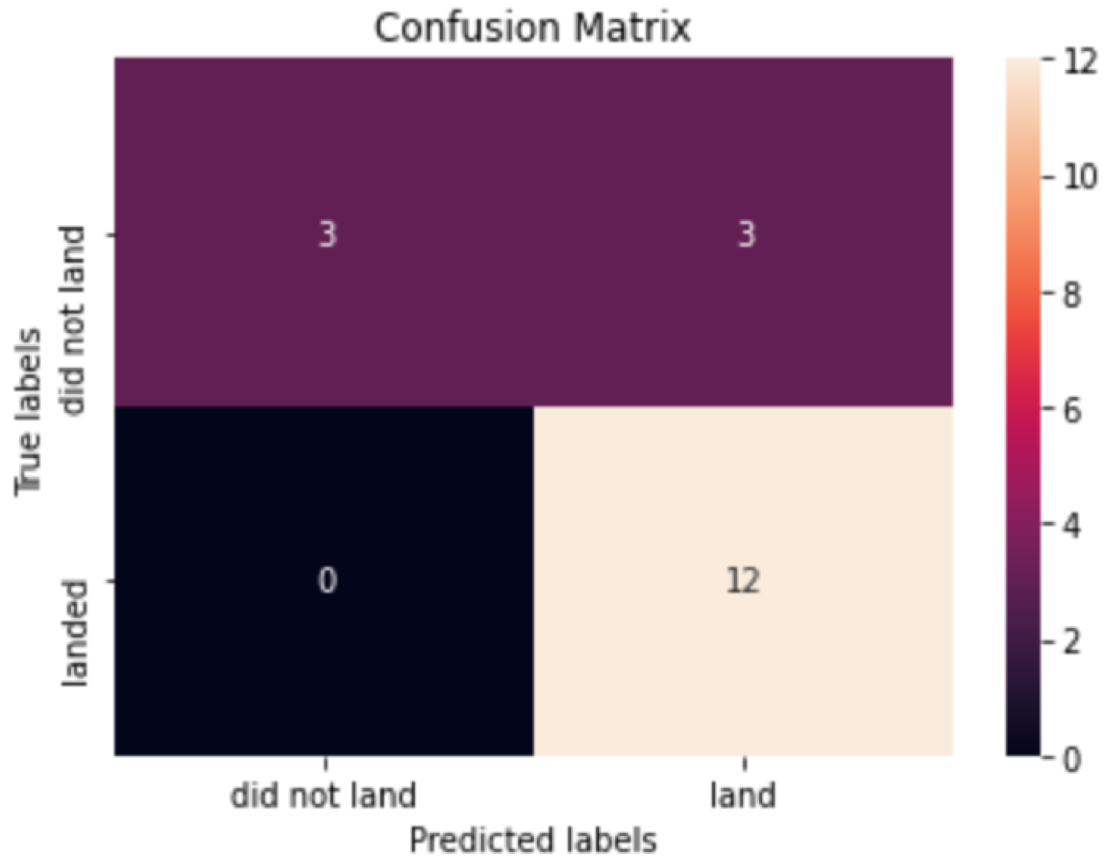
- GridSearchCV best score: 0.848
- Accuracy score on test set: 0.833



RESULTS

3 Predictive Analysis: Decision Tree Evaluation Metrics

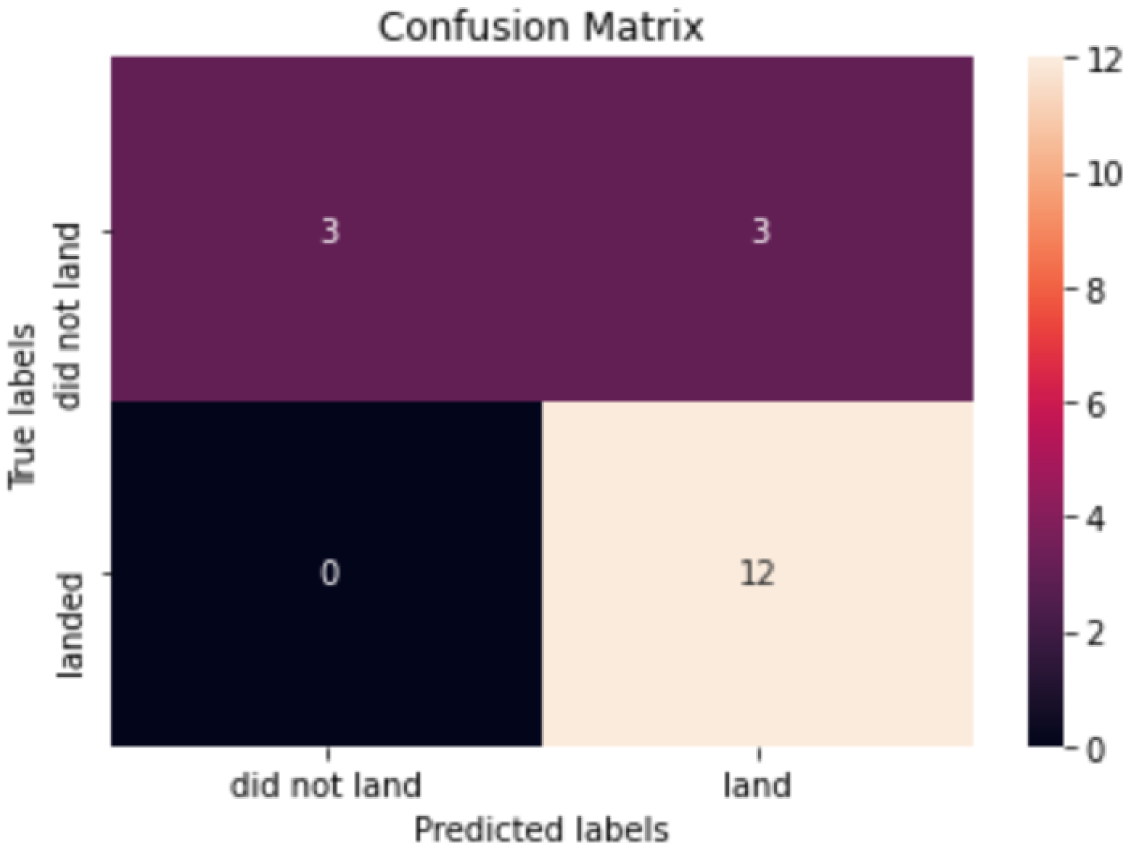
- Decision tree
  - GridSearchCV best score: 0.889
  - Accuracy score on test set: 0.833



RESULTS

5 Predictive Analysis: KNN Evaluation Metrics

- K nearest neighbors (KNN)
  - GridSearchCV best score: 0.848
  - Accuracy score on test set: 0.833



DISCUSSION

From the Analysis above it was discovered/observed that:

- The confusion matrix obtained was virtually the same for all the other models (LR, SVM, Decision Tree, KNN)
- the classifier made 18 Predictions and 12 scenarios were predicted correctly (Yes for landing), and they did land successfully (True positive)
- Three scenarios (top left) were predicted No for landing, and they did not land (True negative)
- Three scenarios (top right) were predicted Yes for landing, but they did not land successfully (False positive)
- Overall, the classifier predicted correctly for 83% of the time ((TP + TN) / Total) with a misclassification or error rate ((FP + FN) / Total) of about 16.5%
- Positive relationship between number of flights and success rate. As the numbers of flights increase, the first stage is more likely to land successfully.
- Success rates appear go up as Payload increases but there is no clear correlation between Payload mass and success rates
- Launch success rate increased by about 80% from 2013 to 2020

DISCUSSION

From the Analysis above it was discovered/observed that:

- Launch Site ‘KSC LC-39A’ recorded the highest launch success rate whereas Launch Site ‘CCAFS SLC- 40’ recorded the least launch success rate.
- Orbits ES-L1, GEO, HEO, and SSO also recorded the highest launch success rates whereas orbit GTO as the least.
- Strategically, lunch sites are located away from the cities and mostly closer to coastline, railroads, and highways.
- The best performing Machine Learning Classification Model was Decision Tree with accuracy of about 87.5%. When data was tested on the models, the accuracy score was about 83% for all models. Perhaps more data may be needed to further tune the models and find a potential better fit.
- Based on the Accuracy scores and EDA analysis, Decision Tree algorithm recorded the highest classification score with a value of .8750.


IBM Developer
SKILLS NETWORK
Thank you
FALCON
SPACEX
IBM