IBM DATA SCIENCE CAPSTONE PROJECT

SpaceX Falcon 9 Landing Analysis

KSHAMIKA GHIYA

https://github.com/Kshamika-Ghiya/IBM-Data-Science/tree/main/Applied%20Capstone%20Project

OUTLINE

1. Executive Summary
2. Introduction
3. Methodology
4. Results
5. Conclusion
6. Appendix

EXECUTIVE SUMMARY

SUMMARY OF METHODOLOGIES

- Data Collection
- Data Wrangling
- Exploratory Data Analysis (EDA)
- Interactive Visual Analytics
- Predictive Analysis (Classification)

SUMMARY OF ALL RESULTS (OUTPUTS AND VISUALIZATIONS)

- Exploratory Data Analysis (EDA) results
- Geospatial Analytics
- Interactive Dashboard
- Predictive Analysis of Classification Models

INTRODUCTION

BACKGROUND:

- The commercial space age is here, and SpaceX is Perhaps the most successful. SpaceX's accomplishments include sending spacecraft to the International Space Station and sending manned missions to Space.
- One reason SpaceX can do this is the rocket launches are relatively inexpensive. SpaceX advertises Falcon 9 rocket launches cost in a range of 62 to 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. Space X's Falcon 9 launch like regular rockets.

PROBLEM:

- Space Y that would like to compete with SpaceX founded by Billionaire industrialist Allon Musk. The job is to determine the price of each launch. This can be done by gathering information about Space X and creating dashboards. It also must be determined if SpaceX will reuse the first stage.

METHODOLOGY

- Data Collection:
  - GET requests to the SpaceX REST API
  - Web Scraping
- Data Wrangling:
  - Creating a landing outcome label where 0, when the booster did not land successfully and 1, when the booster did land successfully
- Exploratory Data Analysis (EDA) using SQL and Visualization
- Interactive Visual Analytics using Folium and Plotly Dash
- Predictive Analysis using Classification Models
  - Building, tuning and evaluating classification models

DATA COLLECTION

Data collection process involved a combination of API requests from Space X public API and web scraping data from a table in Space X's Wikipedia entry.

SPACE X API DATA COLUMNS:

FlightNumber, Date BoosterVersion, PayloadMass, Orbit, LaunchSite, Outcome, Flights, GridFins, Reused, Legs, LandingPad, Block, ReusedCount, Serial, Longitude, Latitude

WIKIPEDIA WEBSCRAPE DATA COLUMNS:

Flight No., Launch site, Payload, PayloadMass, Orbit, Customer, Launch outcome, Version Booster, Booster landing, Date, Time

DATA COLLECTION- SPACEX REST API

1. Make a GET response to SpaceX REST API and convert the response to a .json file and then to Pandas DataFrame.
2. Clean the data, define lists, retrieve and fill the lists and use lists as values in a dictionary and construct a dataset.
3. Create Pandas DataFrame from dataset
4. Filter to include only Falcon 9 launches and reset the column FlightNumber and replace the values of missing PayloadMass with the mean PayloadMass value

DATA COLLECTION– WEB SCRAPING

1. Request HTML page from the static URL and assign the response to an object.
2. Create a BeautifulSoup object from the HTML response object and find all tables in the HTML page.
3. Collect the column header names form the tables.
4. Use the column names as keys in dictionary and then parse all launch tables to fill the dictionary values
5. Finally, convert the dictionary into a Pandas DataFrame.

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

```
# use requests.get() method with the provided static_url
response = requests.get(static_url)
# assign the response to a object
data = response.text
```

```
soup = BeautifulSoup(data, 'html5lib')
html_tables = soup.find_all('table')
```

```
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)
```

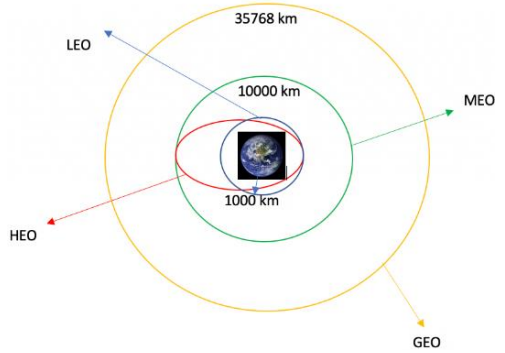
```
launch_dict= dict.fromkeys(column_names)

# Remove an irrelevant 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']=[]
```

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

BETWEEN DATA COLLECTION & DATA WRANGLING

1. Initial Data Exploration using .value_counts():

   - Number of launches on each site,
   - Number and occurrence of each orbit, and
   - Number and occurrence of landing outcomes per orbit




```
# Apply value_counts() on column LaunchSite
df['LaunchSite'].value_counts()
```

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

```
# Apply value_counts on Orbit column
df['Orbit'].value_counts()
```

```
GTO      27
ISS      21
VLEO     14
PO        9
LEO       7
SSO       5
MEO       3
ES-L1     1
GEO       1
SO        1
HEO       1
Name: Orbit, dtype: int64
```

```
# landing_outcomes = values on Outcome column
landing_outcomes = df['Outcome'].value_counts()
landing_outcomes
```

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

2. Landing outcome is shown in the outcome column.

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

```
{'False ASDS', 'False Ocean', 'False RTLS', 'None ASDS', 'None None'}
```

DATA WRANGLING

It is done to determine whether a booster successfully lands, represented by 0 and 1.

1. Define a set of unsuccessful outcomes, bad_outcomes
2. Create a list, landing_class where 0 represents unsuccessful/bad outcomes and 1 represents successful outcome.
3. Create a column names Class which contains the values from the list, landing_class
4. Export this DataFrame as a CSV file.

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

```
{'False ASDS', 'False Ocean', 'False RTLS', 'None ASDS', 'None None'}
```

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

landing_class = []

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

```
df['Class']=landing_class
```

```
df.to_csv("dataset_part\_2.csv", index=False)
```

EXPLORATORY DATA ANALYSIS (EDA)- VISUALIZATION

Exploratory Data Analysis performed on variables Flight Number, Payload Mass, Launch Site, Orbit, Class and Year.

PLOTS USED:

Flight Number vs. Payload Mass, Flight Number vs. Launch Site, Payload Mass vs. Launch Site, Orbit vs. Success Rate, Flight Number vs. Orbit, Payload vs Orbit, and Success Yearly Trend

SCATTER CHART	BAR CHART	LINE CHART
Scatter charts were produced to visualize the relationships between: • Flight Number and Launch Site • Payload and Launch Site • Orbit Type and Flight Number • Payload and Orbit Type Scatter charts are useful to observe relationships, or correlations, between two numeric variables.	Bar charts were produced to visualize the relationship between • Success rate and Orbit Type Bar charts are used to compare a numerical value to a categorical variable Horizontal or vertical bar charts can be used, depending on size of data.	Line charts were produced to visualize the relationship between: • Success Rate and Year, i.e., the launch success yearly trend. Line charts contain numerical values on both axes, and are generally used to show the change of a variable over time.

EXPLORATORY DATA ANALYSIS (EDA)- SQL

SQL QUERIES PERFORMED ON THE DATASET:

1. Display the names of the unique launch sites in the space mission
2. Display 5 records where launch sites begin with the string ‘CCA’
3. Display the total payload mass carried by boosters launched by NASA (CRS)
4. Display the average payload mass carried by booster version F9 v1.1
5. List the date when the first successful landing outcome on a ground pad was achieved
6. List the names of the boosters which had success on a drone ship and a payload mass between 4000-6000 kg
7. List the total number of successful and failed mission outcomes
8. List the names of the booster versions which have carried the maximum payload mass
9. List the failed landing outcomes on drone ships, their booster versions, and launch site names for 2015
10. Rank the count of landing outcomes (such as Failure (drone ship) or Success (ground pad)) between the date 2010-06-04 and 2017-03-20, in descending order

INTERACTIVE VISUAL ANALYSIS- FOLIUM

Geospatial analysis was done using folium maps to visualize the launch data on an interactive map.

1. Marking launch sites on a map
   - Initializing map using Folium's Map object and adding folim.Circle and folium.Marker to each launch site.
2. Marking successful and failed launches for each site.
   - If launches have same coordinates, they are clustered together after assigning class=1, as orange for successful outcome and class=0, as blue for failed/unsuccessful outcomes.
3. Calculate the distances between launch sites and proximity to key locations (City, Coast, Highway, Railway, etc.)
   - Lat and Long values can be used. To draw a line between the 2 maps, folium.PolyLine can be used.

INTERACTIVE VISUAL ANALYSIS- PLOTLY DASH

A dashboard is created using Plotly Dash which includes a Pie Chart and a Scatter Plot.

PIE CHART:

Shows all the successful launches per site.

- Makes it clear to see which sites are most successful.
- Charts can also be filtered using dcc.Dropdown() object to see success/failure ratio at individual site.

SCATTER PLOTS:

Shows the correlation between outcomes (success/failure) and payload mass (kg).

- Can be filtered using a RangeSlider() object by ranges of payload masses
- Can also be filtered by booster version.

PREDICTIVE ANALYSIS- CLASSIFICATION

MODEL DEVELOPMENT:

- Loading the dataset, performing data transformations, splitting the data into training and test datasets using the train_test_split() function, and deciding which machine learning algorithm is most appropriate.
- For each chosen algorithm, create a GridSearchCV object and a dictionary of parameters, then fit the objects to the parameters, and use the training dataset to train the appropriate machine learning model.

MODEL EVALUATION

- For each chosen algorithm using the output of GridSreachCV object, check the tuned hyperparameters, best_params and check the accuracy using score and best_score.
- Then plot and examine the Confusion Matrix.

FINDING THE BEST CLASSIFICATION MODEL

- Review the accuracy score for the algorithm.
- The model with the highest accuracy score is determined as the best performing model.

RESULTS

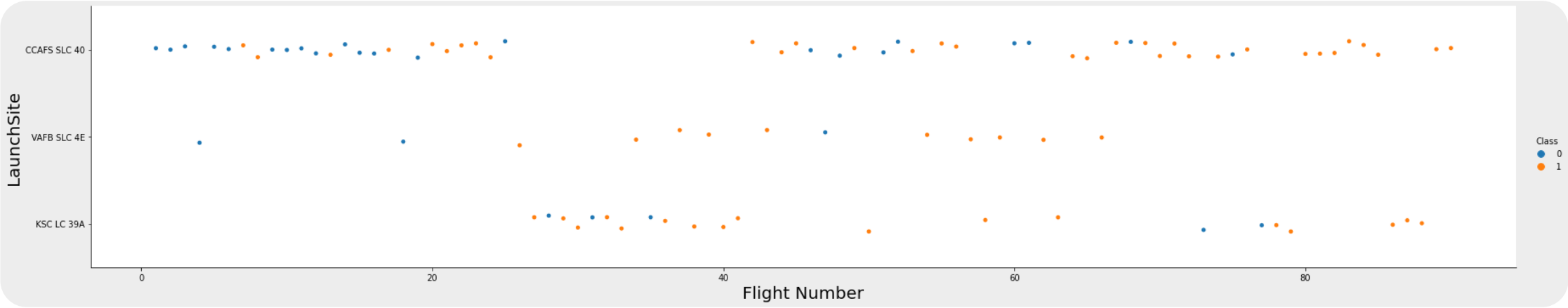
EXPLORATORY DATA ANALYSIS

INTERACTIVE ANALYSIS

PREDICTIVE ANALYSIS

LAUNCH SITE V/S FLIGHT NUMBER

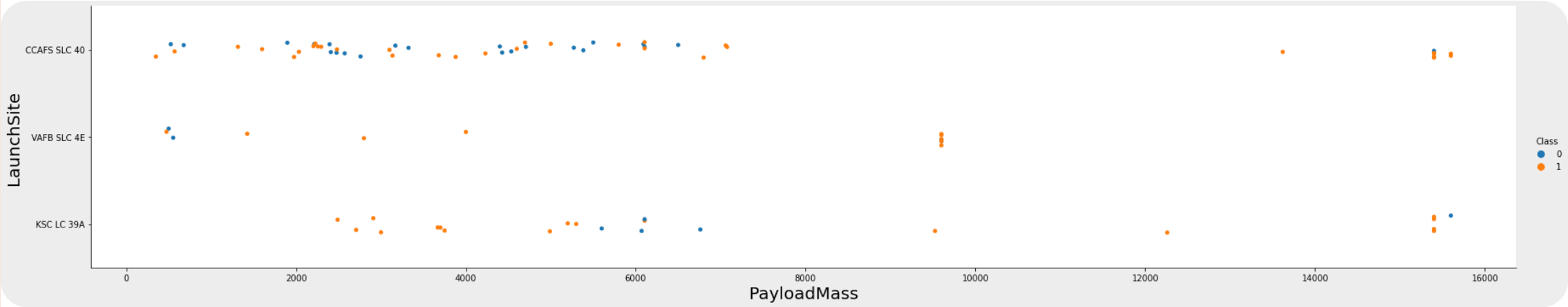
EXPLORATORY DATA ANALYSIS USING SCATTER PLOT:



- Increase in success rate over time (indicated by flight number).
- Flights with flight number<30 were launched from CCAFS SLC 40 were generally unsuccessful.
- Flights with flight number>20 contain flights with more successful landings.

LAUNCH SITE V/S PAYLOAD MASS

EXPLORATORY DATA ANALYSIS USING SCATTER PLOT:



- Payload mass falls between 0-6000 kg.
- Different sites used different payload masses
- There is no clear correlation between payload mass

SUCCESS RATE V/S ORBIT TYPE

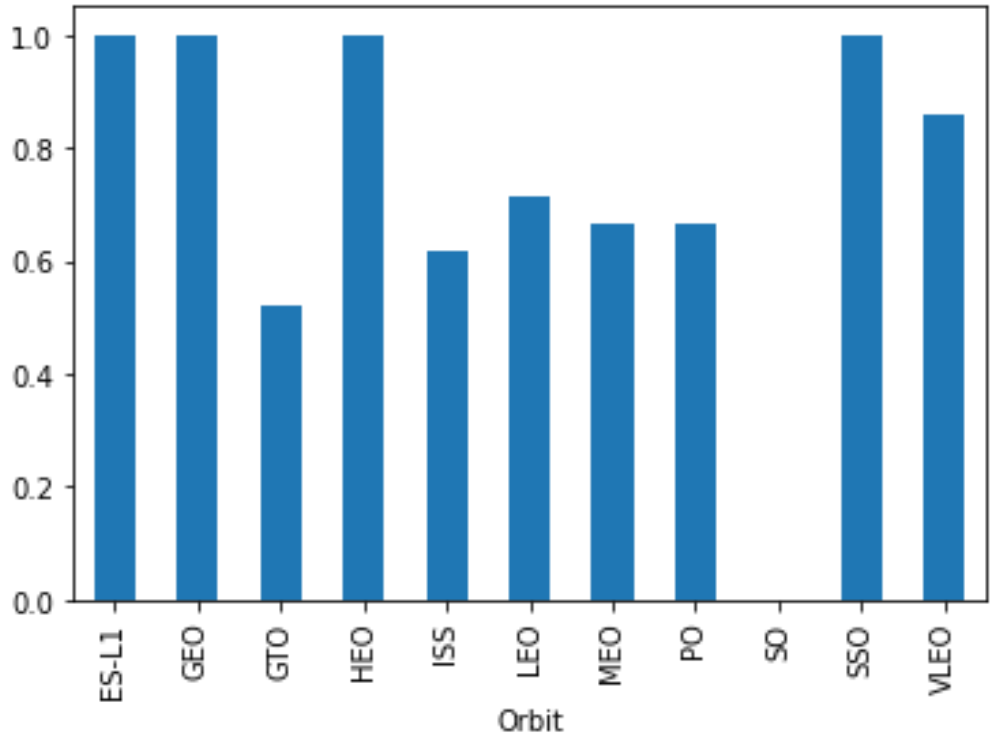
EXPLORATORY DATA ANALYSIS USING BAR CHART

The orbits having the highest success rate, 100%.

- ES-L1 → Earth-Sun First Lagrangian Point
- GEO → Geostationary Orbit
- HEO → High Earth Orbit
- SEO → Sun-Synchronous Orbit

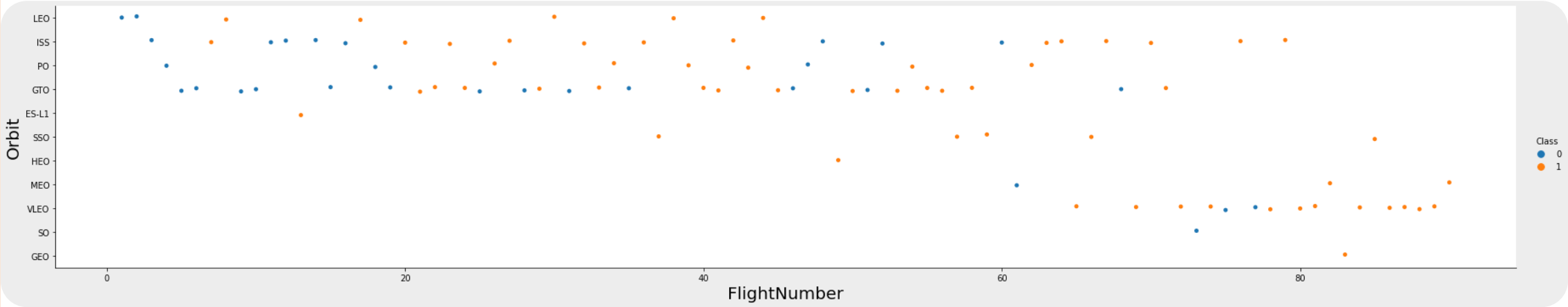
The orbit with the lowest successful rate, 0%:

- SO → Heliocentric Orbit



ORBIT TYPE V/S FLIGHT NUMBER

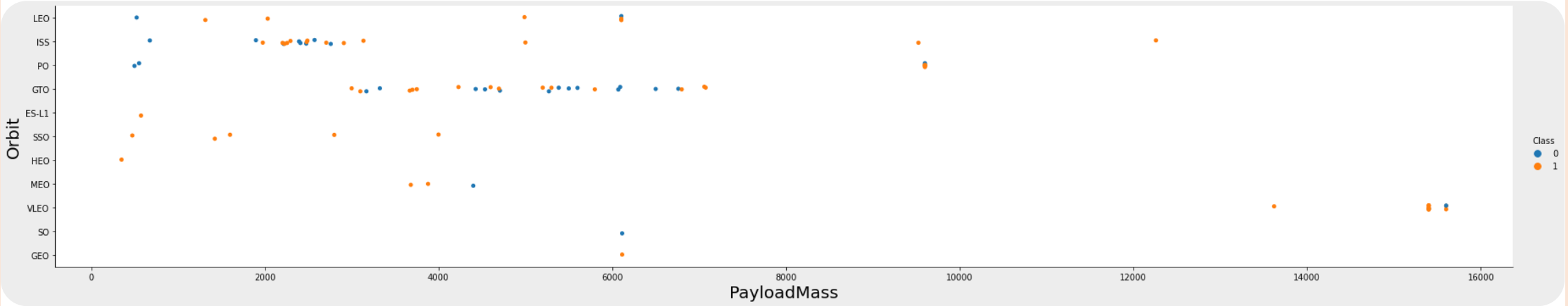
EXPLORATORY DATA ANALYSIS USING SCATTER PLOT



- SpaceX started with LEO orbits which saw moderate success.
- SpaceX performs better in lower orbits or Sun-Synchronous orbits.
- Launch orbit preferences change over flight number.

ORBIT TYPE V/S PAYLOAD MASS

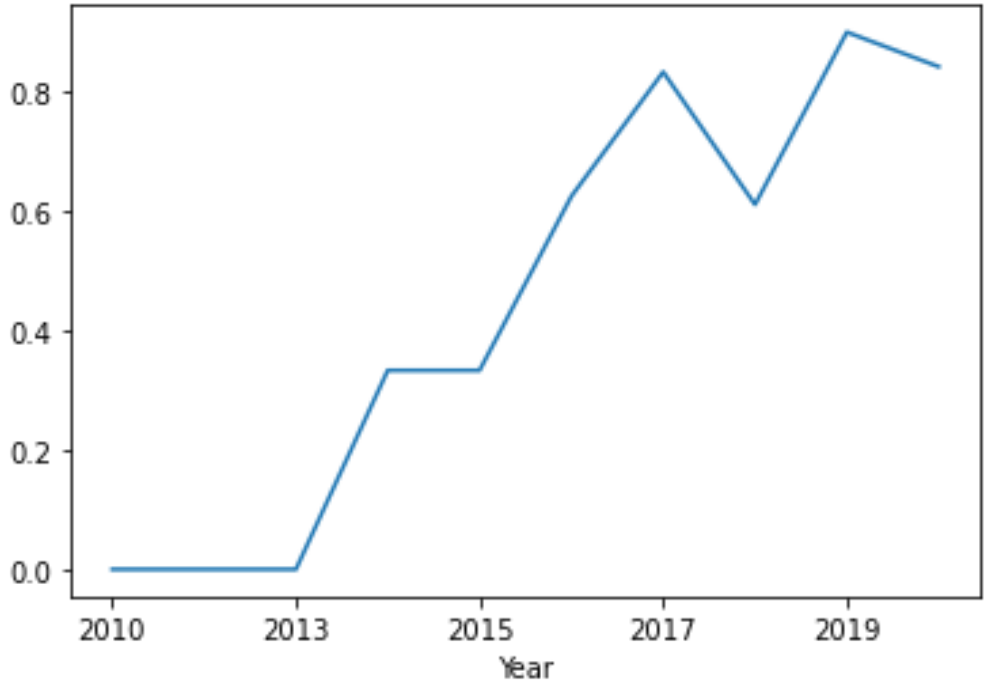
EXPLORATORY DATA ANALYSIS USING SCATTER PLOT



- Most successful orbit is VLEO.
- LEO and SSO have low payload masses.

LAUNCH SUCCESS YEARLY TREND

EXPLORATORY DATA ANALYSIS USING A LINE CHART



- Success is at 80% in the recent years.
- Since 2013, success has increased with a small dip seen in 2018.

ALL LAUNCH SITE NAMES

EXPLORATORY DATA ANALYSIS USING SQL

```
1  %sql SELECT UNIQUE(LAUNCH_SITE) FROM SPACEXTBL;
```

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

LAUNCH SITE NAMES WHICH BEGIN WITH 'CCA'

EXPLORATORY DATA ANALYSIS USING SQL

```
1  %sql SELECT LAUNCH_SITE FROM SPACEXTBL WHERE LAUNCH_SITE LIKE 'CCA%' LIMIT 5;
```

launch_site
CCAFS LC-40
CCAFS LC-40
CCAFS LC-40
CCAFS LC-40
CCAFS LC-40

TOTAL PAYLOAD MASS

EXPLORATORY DATA ANALYSIS USING SQL

```
%sql SELECT SUM(PAYLOAD_MASS__KG_) AS TOTAL_PAYLOAD_MASS FROM SPACEXTBL \
    WHERE CUSTOMER = 'NASA (CRS)';
```

total_payload_mass
45596

AVERAGE PAYLOAD MASS BY F9 v1.1

EXPLORATORY DATA ANALYSIS USING SQL

```
%sql SELECT AVG(PAYLOAD_MASS__KG_) AS AVERAGE_PAYLOAD_MASS FROM SPACEXTBL \
    WHERE BOOSTER_VERSION = 'F9 v1.1';
```

average_payload_mass
2928

FIRST SUCCESSFUL GROUND LANDING DATE

EXPLORATORY DATA ANALYSIS USING SQL

```
%sql SELECT MIN(DATE) AS FIRST_SUCCESSFUL_GROUND_LANDING FROM SPACEXTBL \
    WHERE LANDING__OUTCOME = 'Success (ground pad)';
```

first_successful_ground_landing
2015-12-22

SUCCESSFUL DRONE SHIP LANDING PAYLOAD (4000-6000)

EXPLORATORY DATA ANALYSIS USING SQL

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

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

TOTAL NUMBER OF SUCCESSFUL & FAILURE MISSION OUTCOMES

EXPLORATORY DATA ANALYSIS USING SQL

```
1  %sql SELECT MISSION_OUTCOME, COUNT(MISSION_OUTCOME) AS TOTAL_NUMBER FROM SPACEXTBL GROUP BY MISSION_OUTCOME;
```

mission_outcome	total_number
Failure (in flight)	1
Success	99
Success (payload status unclear)	1

BOOSTERS WHICH CARRIED MAXIMUM PAYLOAD

EXPLORATORY DATA ANALYSIS USING SQL

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

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

EXPLORATORY DATA ANALYSIS USING SQL

```
%sql SELECT BOOSTER_VERSION, LAUNCH_SITE FROM SPACEXTBL \
    WHERE (LANDING__OUTCOME = 'Failure (drone ship)') AND (EXTRACT(YEAR FROM DATE) = '2015');
```

booster_version	launch_site
F9 v1.1 B1012	CCAFS LC-40
F9 v1.1 B1015	CCAFS LC-40

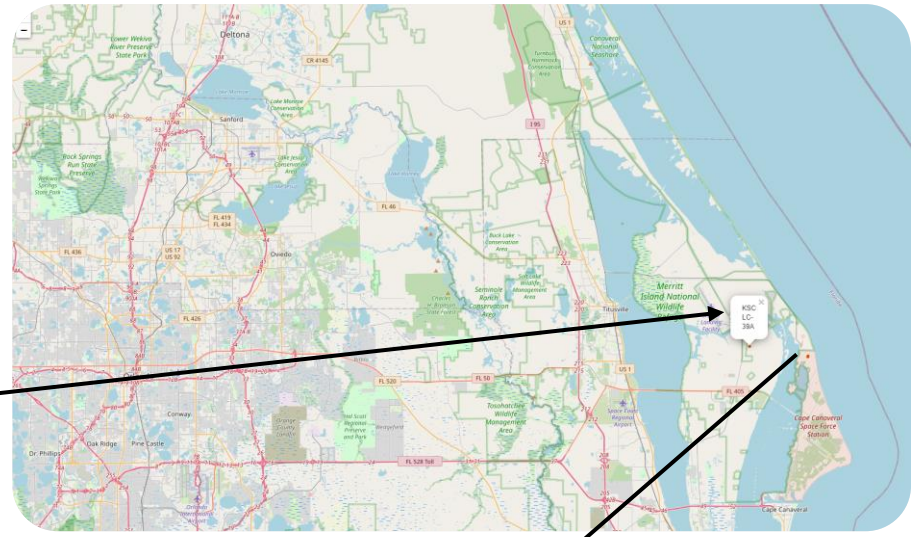
RANK LANDING OUTCOMES B/W 2010/06/04 TO 2017/03/20

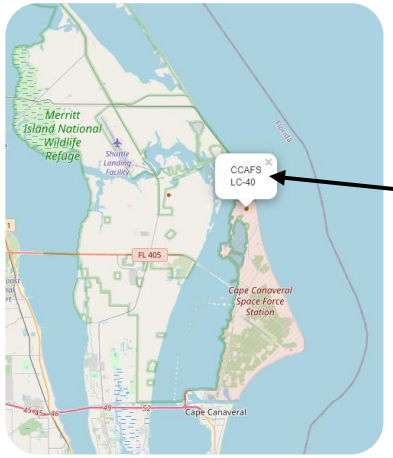
EXPLORATORY DATA ANALYSIS USING SQL

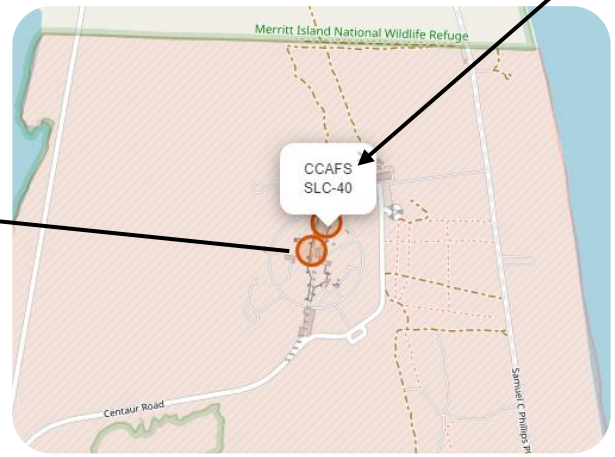
```
%sql SELECT LANDING__OUTCOME, COUNT(LANDING__OUTCOME) AS TOTAL_NUMBER FROM SPACEXTBL \
    WHERE DATE BETWEEN '2010-06-04' AND '2017-03-20' \
    GROUP BY LANDING__OUTCOME \
    ORDER BY TOTAL_NUMBER DESC;
```

landing__outcome	total_number
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

ALL LAUNCH SITES ON A MAP

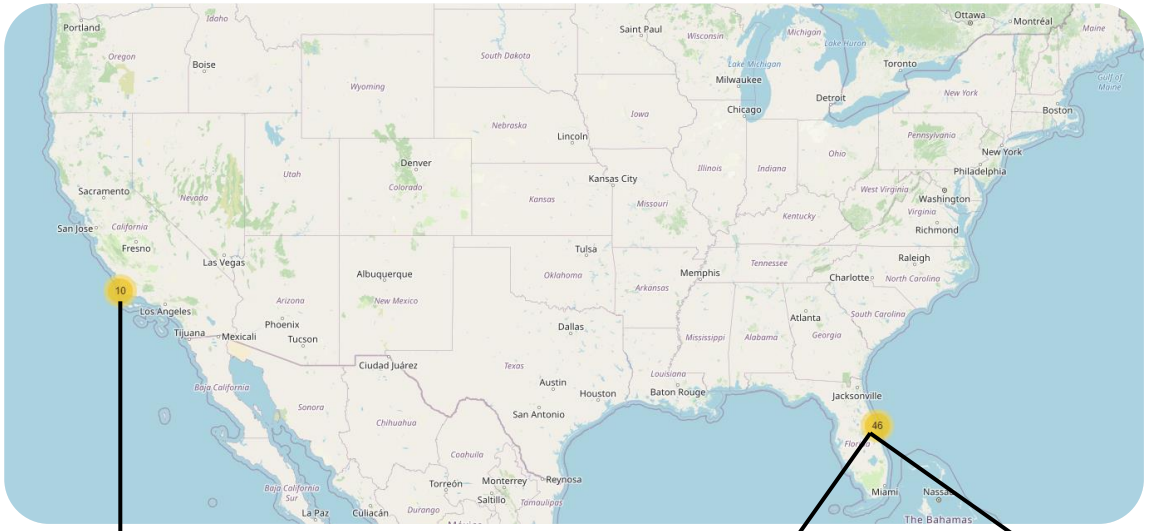


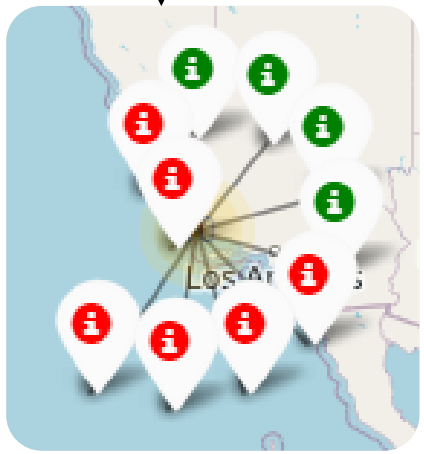


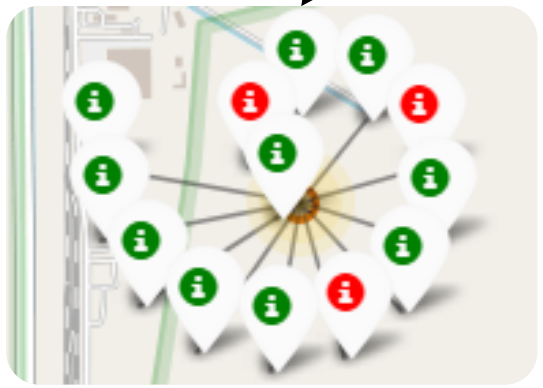


All SpaceX launch sites are on the coast of the United States of America, mainly Florida and California

SUCCESS / FAILED LAUNCHES FOR EACH SITE



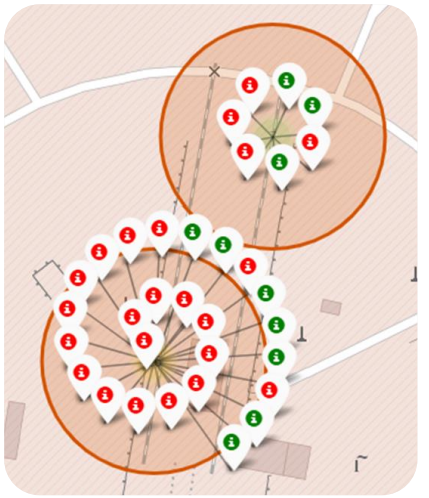




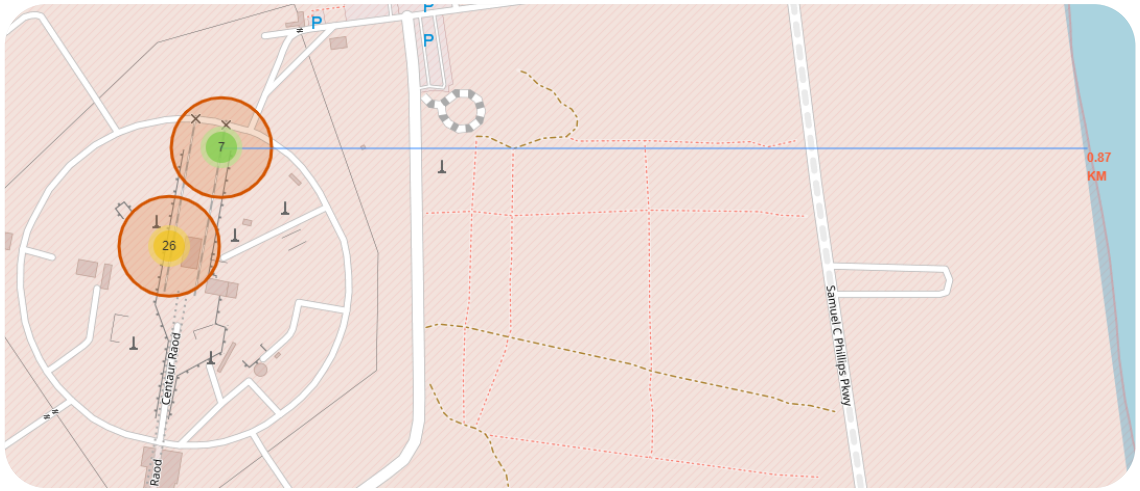
Launches have been grouped into clusters and annotated with green markers for successful launches and red markers for unsuccessful launches.

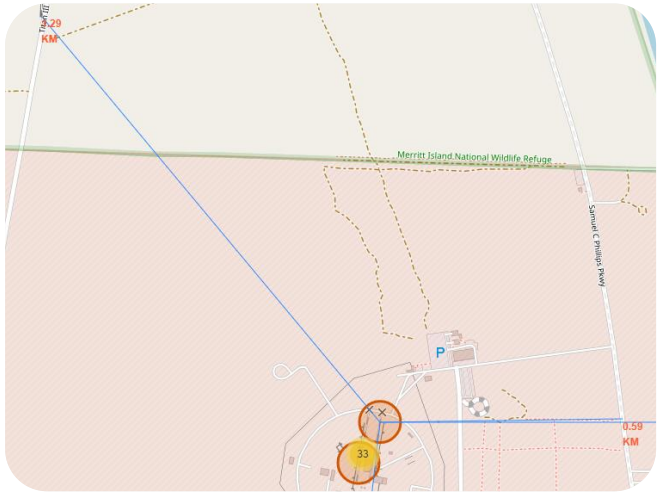
CCAFS SLC-40 and CCAFS LC-40

=

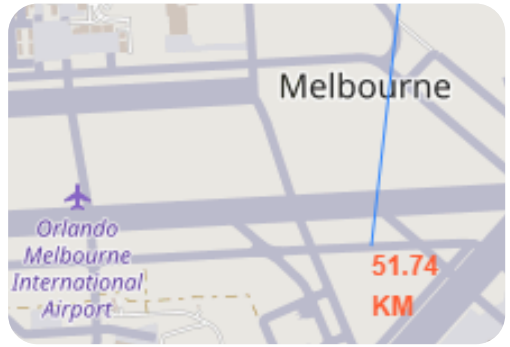


PROXIMITY OF LAUNCH SITES TO OTHER POINTS OF INTEREST





Using CCAFS SLC-40 launch site as an example, understanding the placement of launch sites.



Are launch sites near railways?

- YES. The coastline is only 0.87 km due East.

Are launch sites near highways?

- YES. The nearest highway is only 0.59km away.

Are launch sites near railways?

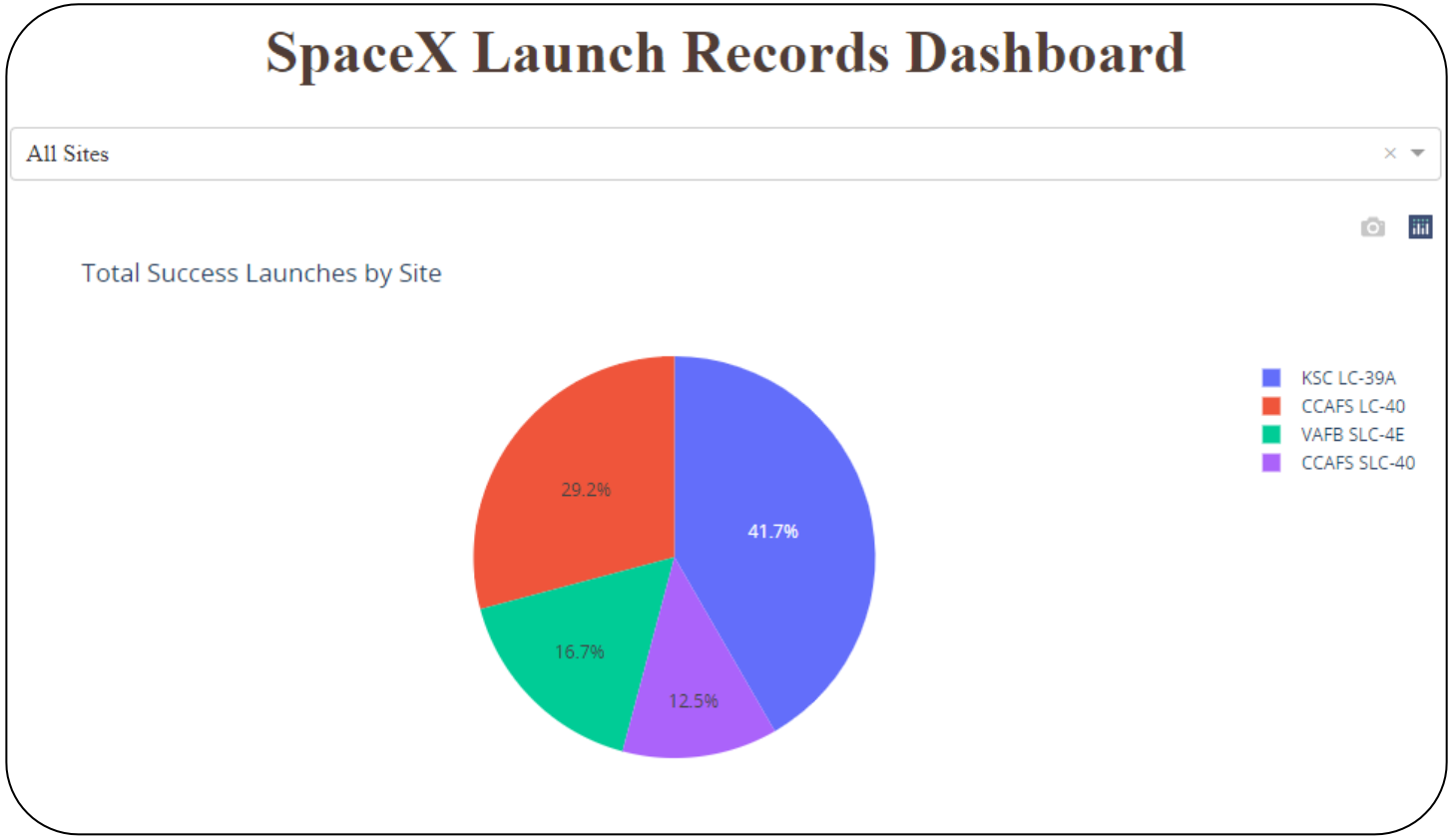
- YES. The nearest railway is only 1.29 km away.

Do launch sites keep certain distance away from cities?

- YES. The nearest city is 51.74 km away.

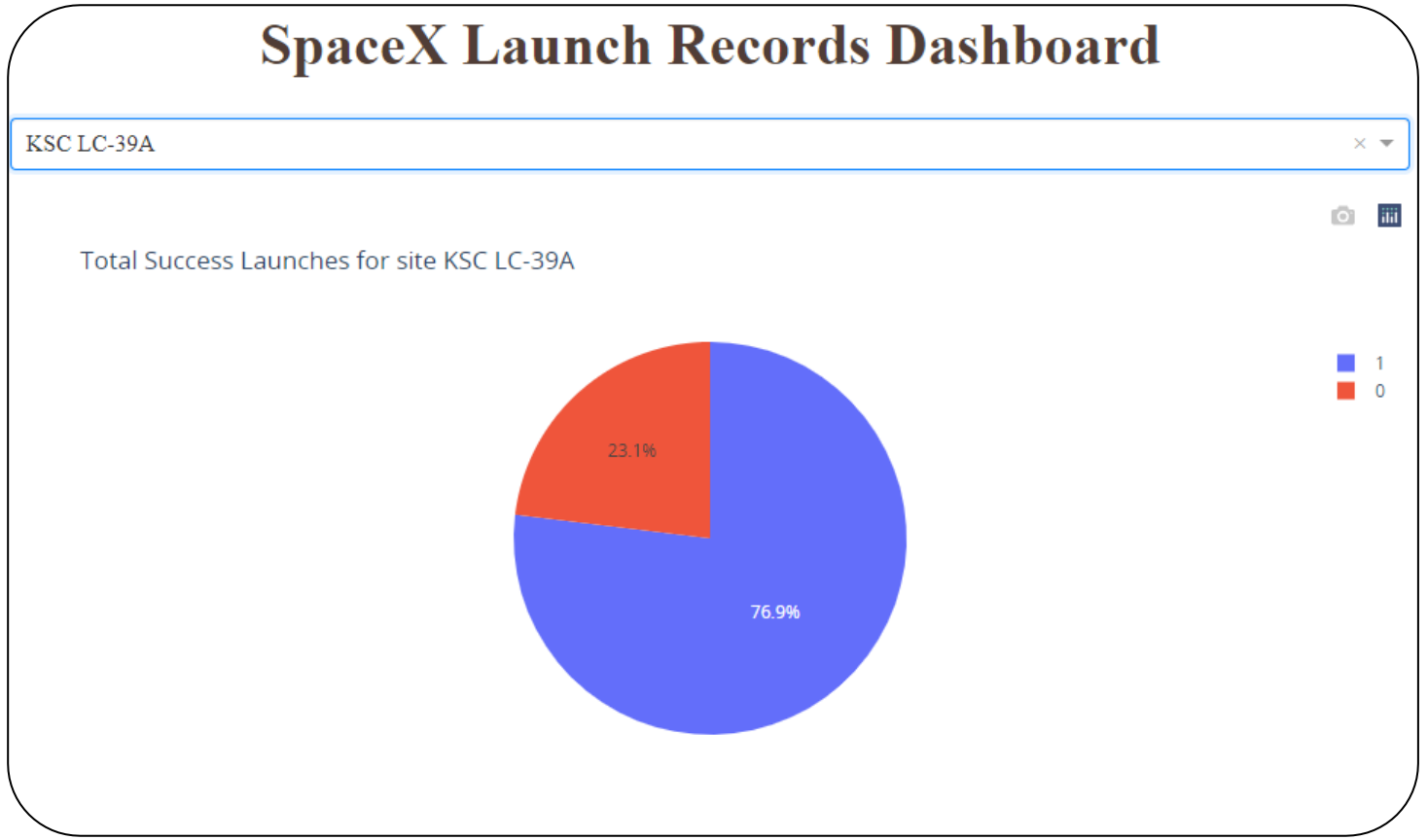
LAUNCH SUCCESS COUNT FOR ALL SITES

The launch site KSC LC-39A has the most successful launches with a total of 41.7% of successful launches



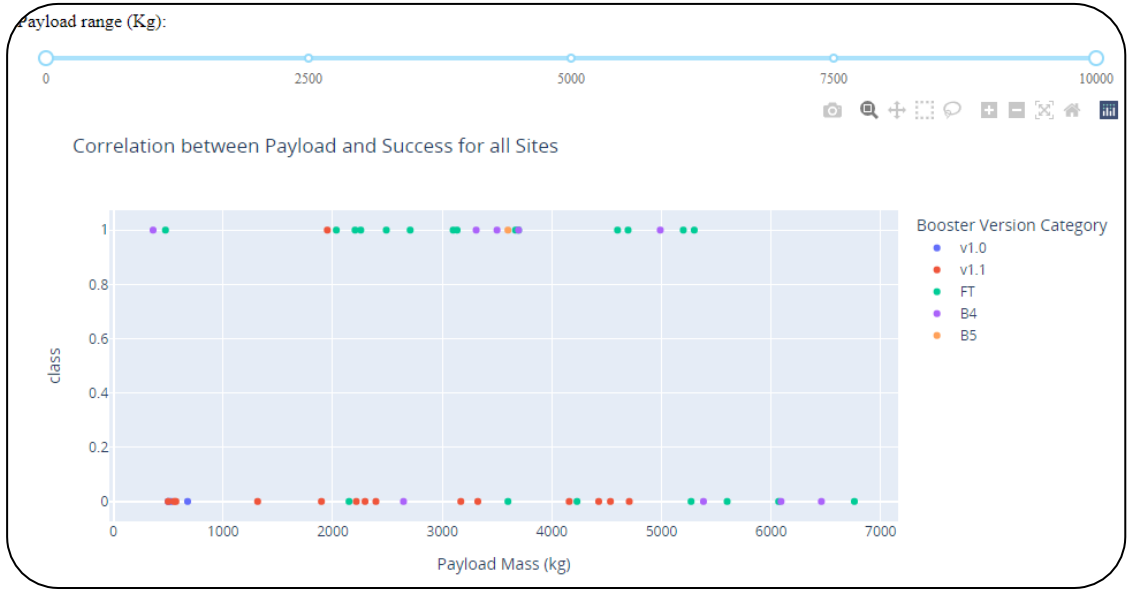
PIE CHART FOR LAUNCH SITE HAVING HIGHEST SUCCESS RATIO

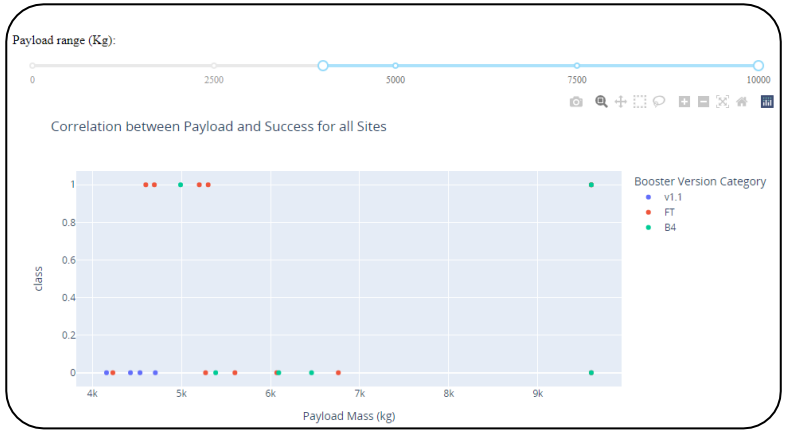
The launch site KSC LC-39A has the highest rate of successful launches with a 76.9% success rate.



LAUNCH OUTCOME V/S PAYLOAD SCATTER PLOT

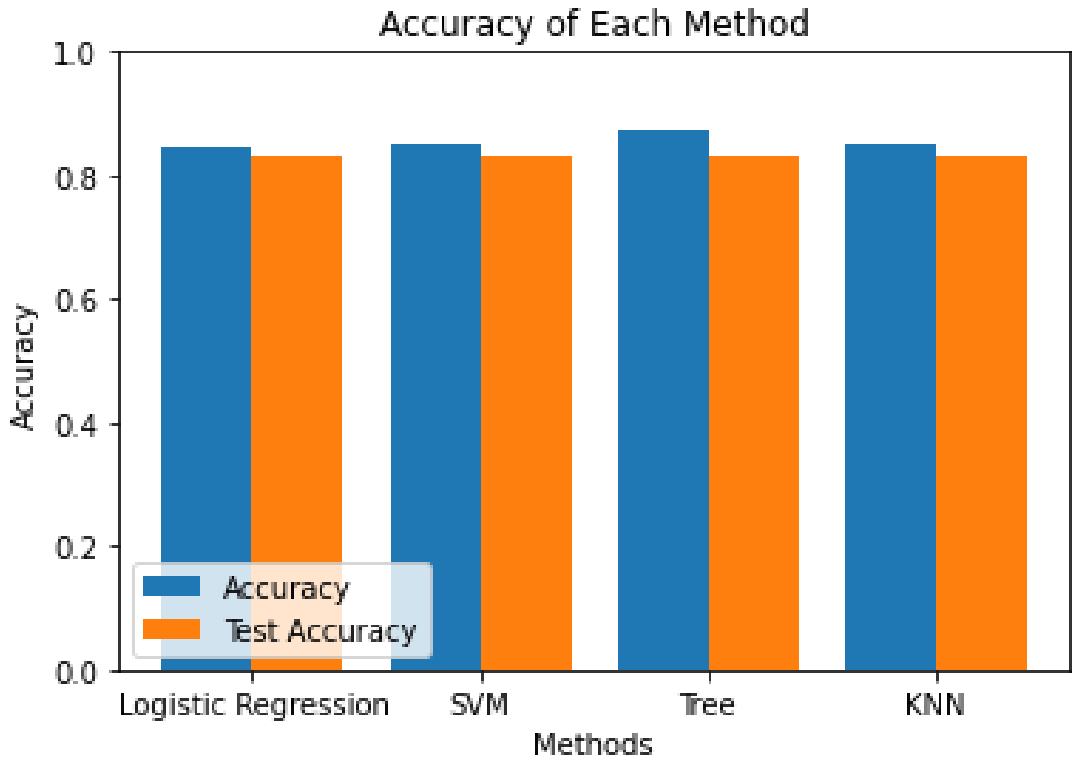
2 ranges of payloads, 0-4000 kg (low payloads) and 4000-10000 kg (massive payloads) are created which shows that the success rate for massive payloads is lower than the rate of low payloads.





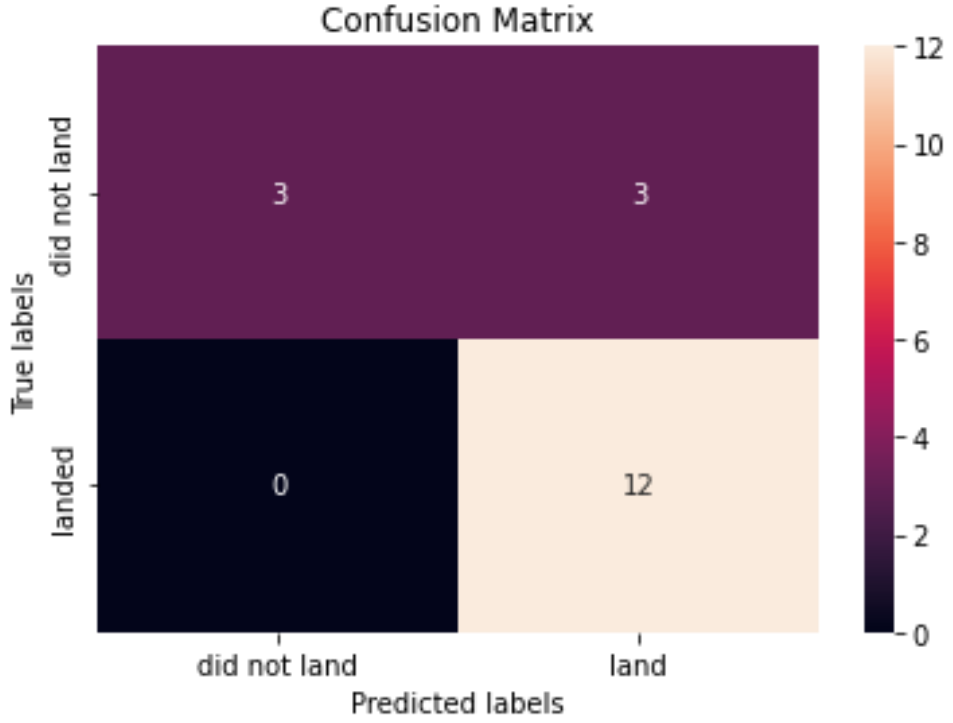
CLASSIFICATION ACCURACY

- All models have same accuracy on the test set at 83.33%.
- Sample size of the test size is small, i.e., 18.



CONFUSION MATRIX

- The confusion matrix is same across all models.
- Models predicted 12 successful landings when the true label was successful landing.
- Models predicted 3 successful landings when the true label was unsuccessful landings (false positive).
- Models predicted 3 unsuccessful landings when the true label was unsuccessful landing.



Correct predictions are on a diagonal from top left to bottom right

CONCLUSION

As the number of flights increases, the rate of success at a launch site increases, with most early flights being unsuccessful, i.e., with more experience, the success rate increases.

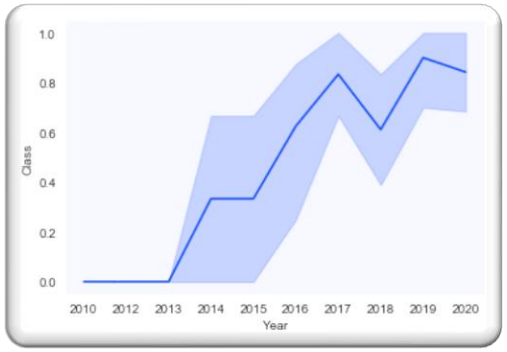
- Between 2010 and 2013, all landings were unsuccessful (as the success rate is 0).
- After 2013, the success rate generally increased, despite small dips in 2018 and 2020.
- After 2016, there was always a greater than 50% chance of success.

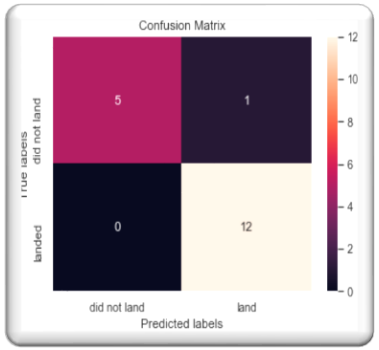
Orbit types ES-L1, GEO, HEO, and SSO, have the highest (100%) success rate.

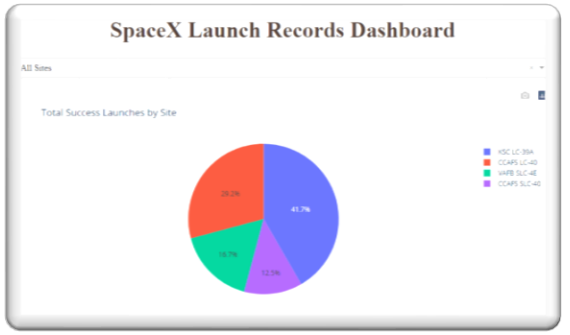
- The 100% success rate of GEO, HEO, and ES-L1 orbits can be explained by only having 1 flight into the respective orbits.
- The 100% success rate in SSO is more impressive, with 5 successful flights.
- The orbit types PO, ISS, and LEO, have more success with heavy payloads.
- VLEO (Very Low Earth Orbit) launches are associated with heavier payloads, which makes intuitive sense.

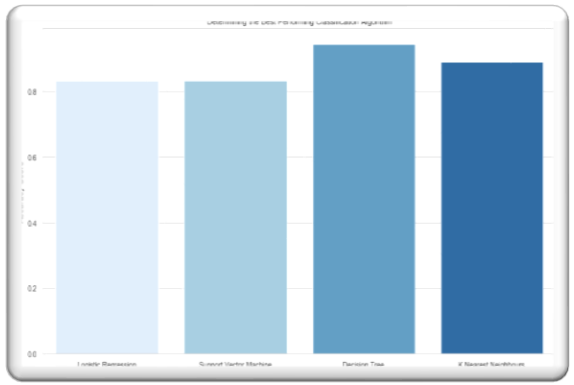
CONCLUSION

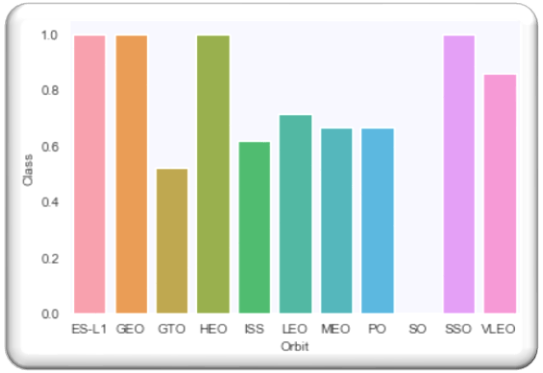
The launch site KSC LC-39 A had the most successful launches, with 41.7% of the total successful launches, and the highest rate of successful launches, with a 76.9% success rate.
The success for massive payloads (over 4000kg) is lower than that for low payloads.
The best performing classification model is the Decision Tree model, with an accuracy of 94.44%.











THANK YOU