IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

KANCHANA S.
2023/09/19

Outline

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

Executive Summary

- Summary of methodologies
  - Data Collection through API
  - Data Collection with Web Scraping
  - Data Wrangling
  - Exploratory Data Analysis with SQL
  - Exploratory Data Analysis with Data Visualization
  - Interactive Visual Analytics with Folium
  - Machine Learning Prediction
- Summary of all results
  - Exploratory Data Analysis results
  - Interactive analytics demo in screenshots
  - Predictive analysis results

Introduction

- Project background and context
  - SpaceX is the most successful company of the commercial space age, making space travel affordable. The company 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. Based on public information and machine learning models, we are going to predict if SpaceX will reuse the first stage
- Problems you want to find answers
  - How do variables such as payload mass, launch site, number of flights, and orbits affect the success of the first stage landing?
  - Does the rate of successful landings increase over the years?
  - What is the best algorithm that can be used for binary classification in this case?

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Using SpaceX Rest API
  - Using Web Scrapping
- Perform data wrangling
  - Using One Hot Encoding
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Building, tuning and evaluation of classification models to ensure the best results

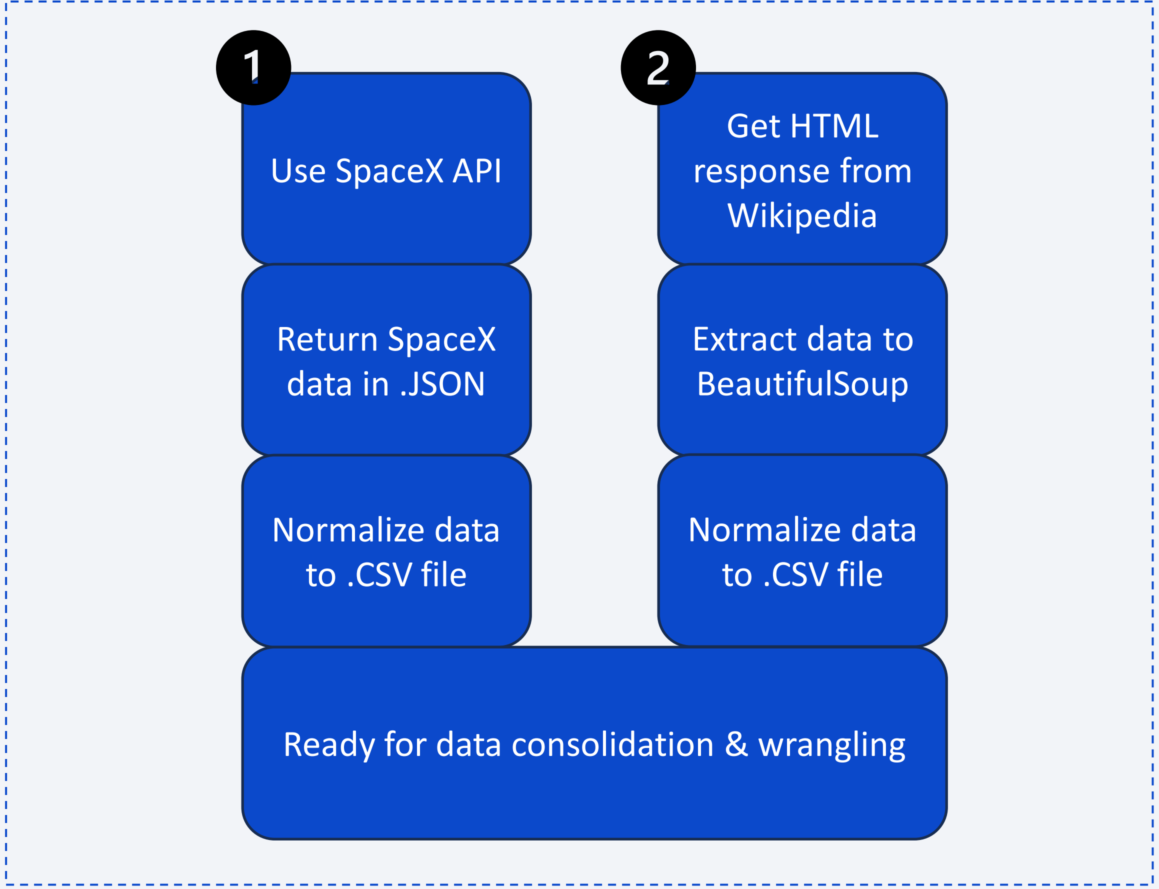
Data Collection

- Understand its content,
- Assess its quality,
- Discover any interesting preliminary insights, and,
- Determine whether additional data is necessary to fill any gaps in the data.

Data Collection – SpaceX API

- Present your data collection with SpaceX REST calls using key phrases and flowcharts

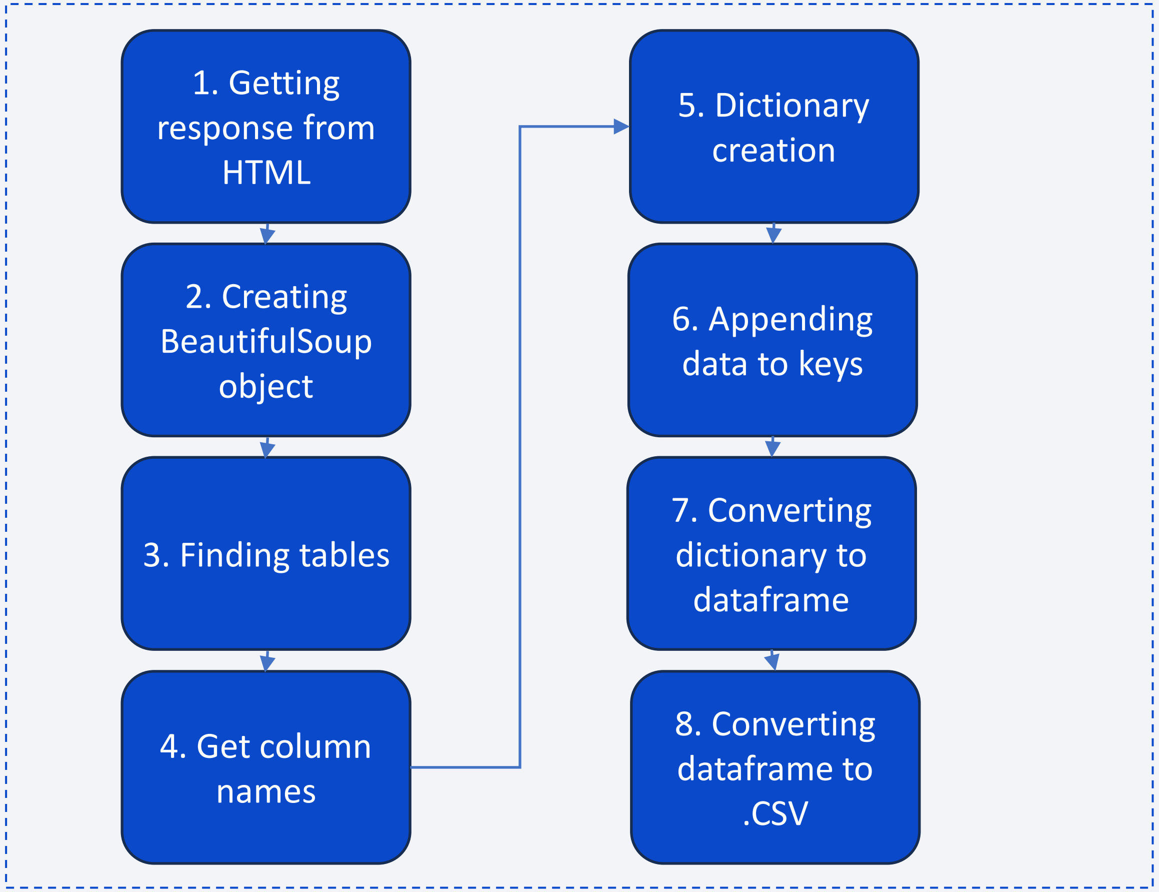
- Add the GitHub URL of the completed SpaceX API calls notebook (must include completed code cell and outcome cell), as an external reference and peer-review purpose



Data Collection - Scraping

- Present your web scraping process using key phrases and flowcharts

- Add the GitHub URL of the completed web scraping notebook, as an external reference and peer-review purpose



Data Wrangling

1. Perform exploratory Data Analysis and determine Training Labels
2. Calculate the number of launches on each site
3. Calculate the number and occurrence of each orbit
4. Calculate the number and occurrence of mission outcome per orbit type
5. Create a landing outcome label from Outcome column
6. Exporting the data to .CSV

EDA with Data Visualization

There are various types of plots commonly used in data visualization.

- Line plots capture trends and changes over time, allowing us to see patterns and fluctuations.
- Bar plots compare categories or groups, providing a visual comparison of their values.
- Scatter plots explore relationships between variables, helping us identify correlations or trends.
- Box plots display the distribution of data, showcasing the median, quartiles, and outliers.

EDA with SQL

- Use Data Manipulation Language (DML) statements to read and modify data.
- The search condition of the WHERE clause uses a predicate to refine the search.
- COUNT, DISTINCT, and LIMIT are expressions that are used with SELECT statements.
- INSERT, UPDATE, and DELETE are DML statements for populating and changing tables.

Build an Interactive Map with Folium

- Folium is a data visualization library in Python that helps people visualize geospatial data.
- With Folium, you can create maps of different styles, such as street-level maps, stamen maps, and more.
- A feature of Folium is that you can create different map styles using the tiles parameter.
- With Folium, you can easily add markers on maps.
- The ‘location’ parameter specifies the latitude and longitude coordinates of the center point of the map.
- Markers play a vital role in enhancing interactivity and adding context to maps.
- The folium.Marker() function specifies location parameters.
- The popup parameter provides a label upon being clicked.
- Markers can be created using “feature group.”
- A choropleth map is a thematic map in which areas are shaded or patterned in proportion to the measurement of the statistical variable.
- When creating a choropleth map, Folium requires a GeoJson file that includes geospatial data of the region.
- The Mapbox Bright Tileset displays the name of every country when used on a map.

Build a Dashboard with Plotly Dash

- Dash is an Open-Source User Interface Python library for creating reactive, web-based applications.
- It is easy to build Graphical User Interfaces using Dash as it abstracts all technologies required to make the applications.
- There are two components of Dash: Core and HTML components.
- The dash_core_components describe higher-level interactive components generated with JavaScript, HTML, and CSS through the React.js library.
- The dash_html_components library has a component for every HTML tag.
- A callback function is a python function that is automatically called by Dash whenever an input component's property changes.
- The @app.callback decorator decorates the callback function in order to tell Dash to call it whenever there is a change in the input component value.
- The callback function takes input and output components as parameters and performs operations to return the desired result for the output component.

Predictive Analysis (Classification)

- **Define the explanatory variable and the response variable:** Define the response variable (y) as the focus of the experiment and the explanatory variable (x) as a variable used to explain the change of the response variable. Understand the differences between Simple Linear Regression because it concerns the study of only one explanatory variable and Multiple Linear Regression because it concerns the study of two or more explanatory variables.
- **Evaluate the model using Visualization:** By visually representing the errors of a variable using scatterplots and interpreting the results of the model.
- **Identify alternative regression approaches:** Use a Polynomial Regression when the Linear regression does not capture the curvilinear relationship between variables and how to pick the optimal order to use in a model.
- **Interpret the R-square and the Mean Square Error:** Interpret R-square (x 100) as the percentage of the variation in the response variable y that is explained by the variation in explanatory variable(s) x. The Mean Squared Error tells you how close a regression line is to a set of points. It does this by taking the average distances from the actual points to the predicted points and squaring them.

Results

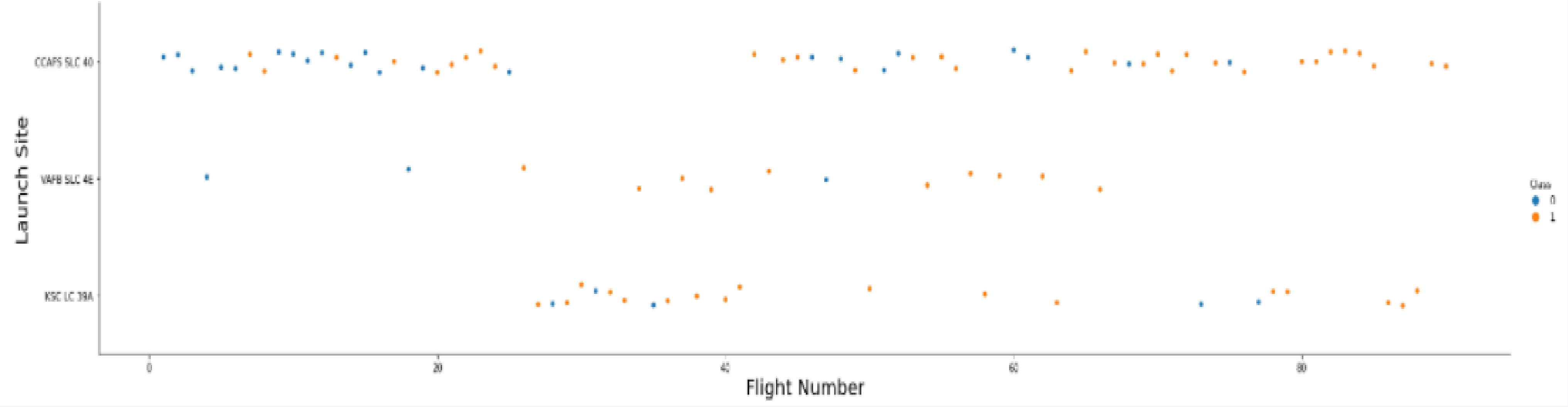
- Exploratory data analysis results
- Interactive analytics demo in screenshots
- Predictive analysis results

Section 2

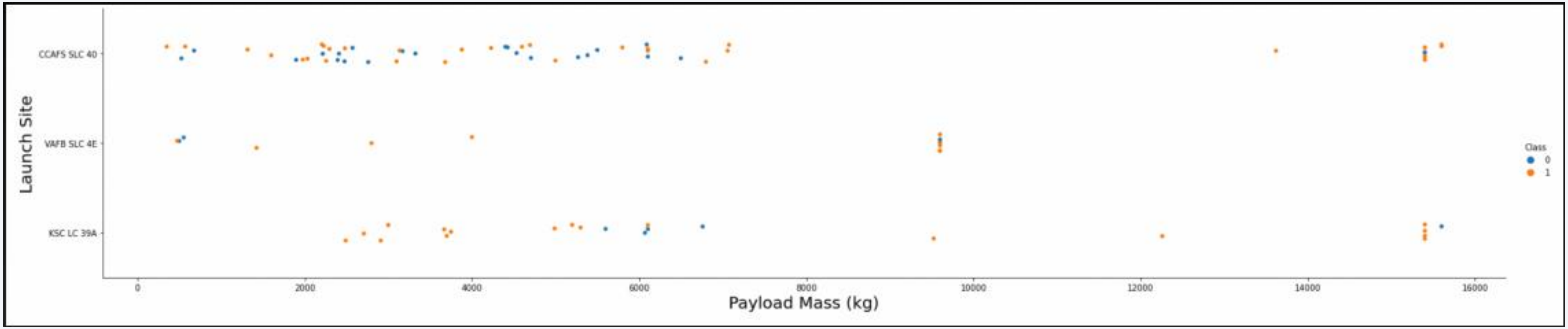
Insights drawn from EDA

Flight Number vs. Launch Site

- From the plot, we found that the larger the flight amount at a launch site, the greater the success rate at a launch site.



Payload vs. Launch Site



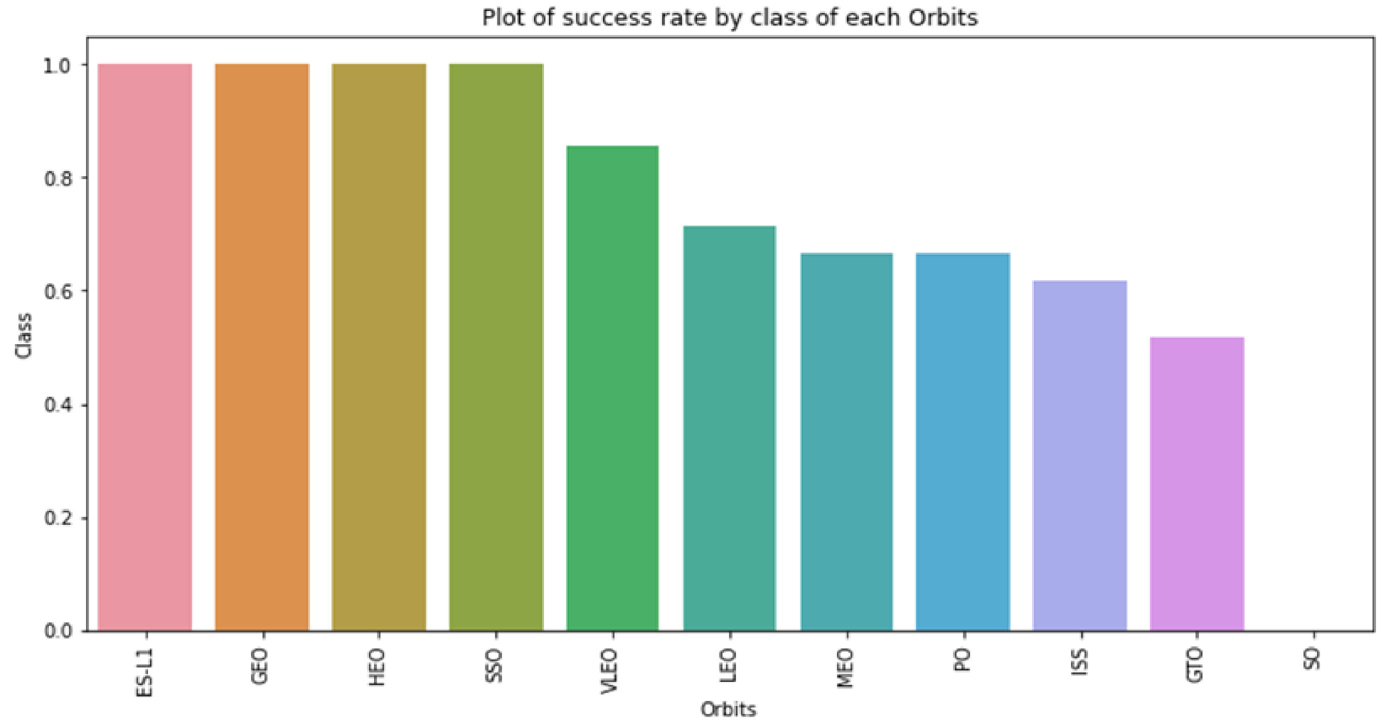
Explanation:

- For every launch site the higher the payload mass, the higher the success rate.
- Most of the launches with payload mass over 7000 kg were successful.
- KSC LC 39A has a 100% success rate for payload mass under 5500 kg too

Success Rate vs. Orbit Type

Explanation:

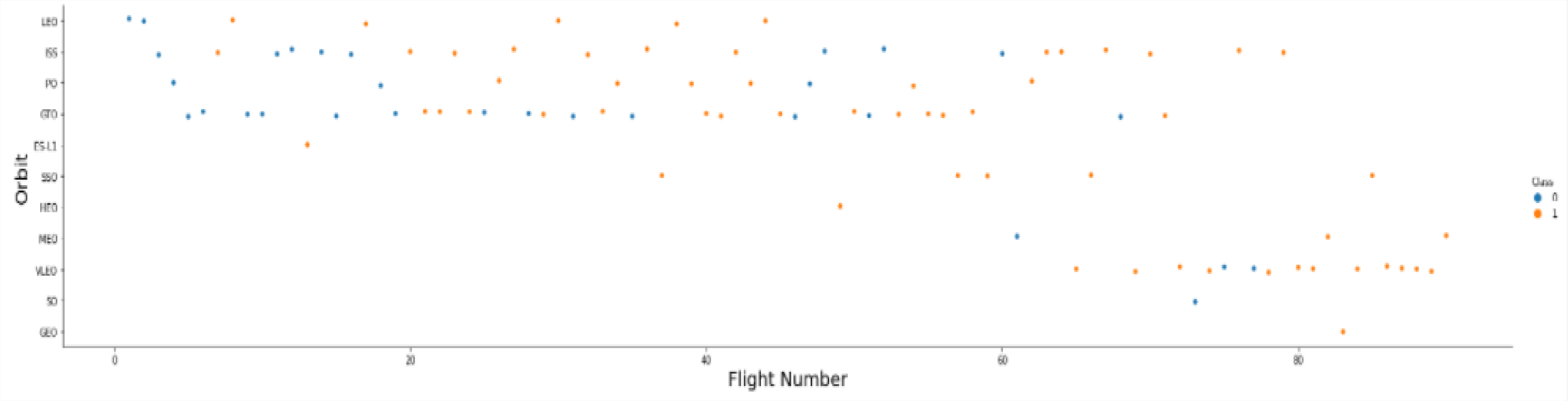
- Orbits with 100% success rate:
  - ES-L1, GEO, HEO, SSO
- Orbits with 0% success rate:
  - SO
- Orbits with success rate between 50% and 85%:
  - GTO, ISS, LEO, MEO, PO



Flight Number vs. Orbit Type

Explanation:

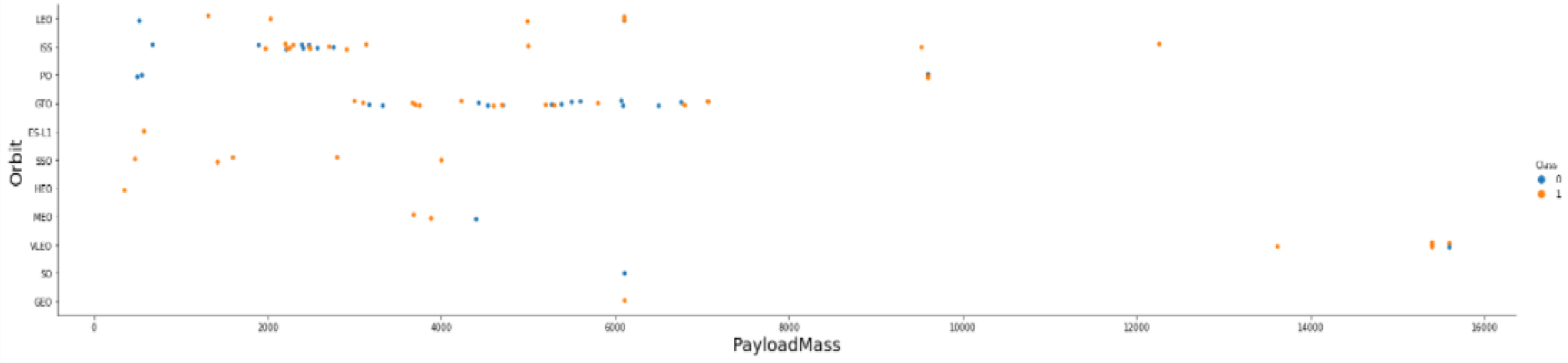
- In the LEO orbit the Success appears related to the number of flights; on the other hand, there seems to be no relationship between flight number when in GTO orbit. Show the screenshot of the scatter plot with explanations



Payload vs. Orbit Type

Explanation:

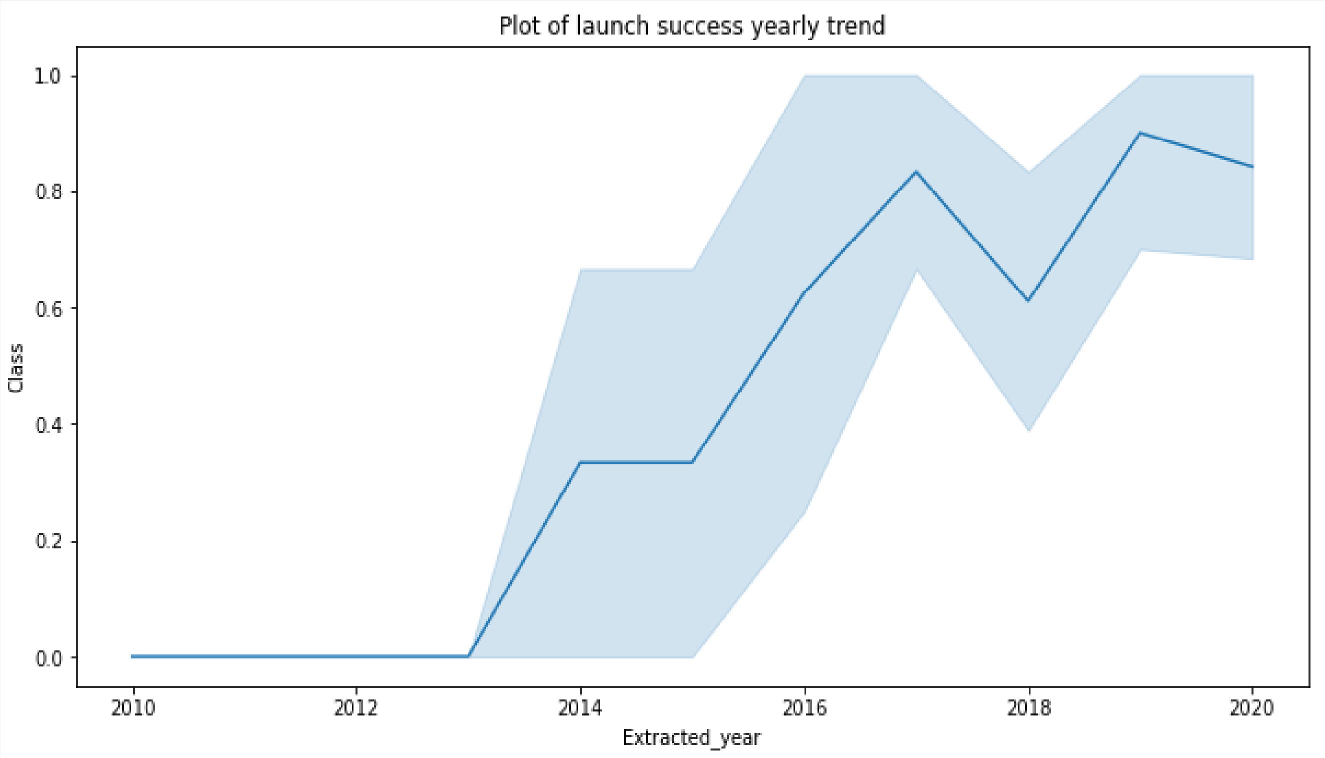
- Heavy payloads have a negative influence on GTO orbits and positive on GTO and Polar LEO (ISS) orbits. Show the screenshot of the scatter plot with explanations



Launch Success Yearly Trend

Explanation:

- The success rate since 2013 kept increasing till 2020.



All Launch Site Names

- Used the key word DISTINCT to show only unique launch sites from the SpaceX data.

```
In [4]: %sql select distinct launch_site from SPACEXDATASET;

 * ibm_db_sa://wzf08322:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/bludb
Done.
Out[4]:
```

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

Launch Site Names Begin with 'CCA'

- 5 records where launch sites begin with `CCA`

In [5]:
```
%sql select * from SPACEXDATASET where launch_site like 'CCA%' limit 5;
```

```
 * ibm_db_sa://wzf08322:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/bludb
Done.
```

Out[5]:

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

Total Payload Mass

- The total payload carried by boosters from NASA

Display the total payload mass carried by boosters launched by NASA (CRS)

```
task_3 = '''
        SELECT SUM(PayloadMassKG) AS Total_PayloadMass
        FROM SpaceX
        WHERE Customer LIKE 'NASA (CRS)'
        '''
create_pandas_df(task_3, database=conn)
```

	total_payloadmass
0	45596

Average Payload Mass by F9 v1.1

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

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

In [13]:

```
task_4 = '''
        SELECT AVG(PayloadMassKG) AS Avg_PayloadMass
        FROM SpaceX
        WHERE BoosterVersion = 'F9 v1.1'
        '''
create_pandas_df(task_4, database=conn)
```

Out[13]:

	avg_payloadmass
0	2928.4

First Successful Ground Landing Date

- The dates of the first successful landing outcome on ground pad

In [14]:
```
task_5 = '''
        SELECT MIN(Date) AS FirstSuccessfull_landing_date
        FROM SpaceX
        WHERE LandingOutcome LIKE 'Success (ground pad)'
        '''
create_pandas_df(task_5, database=conn)
```

Out[14]:

	firstsuccessfull_landing_date
0	2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

- List the names of boosters which have successfully landed on drone ship and had payload mass greater than 4000 but less than 6000

In [15]:

```
task_6 = '''
        SELECT BoosterVersion
        FROM SpaceX
        WHERE LandingOutcome = 'Success (drone ship)'
            AND PayloadMassKG > 4000
            AND PayloadMassKG < 6000
        '''
create_pandas_df(task_6, database=conn)
```

Out[15]:

	boosterversion
0	F9 FT B1022
1	F9 FT B1026
2	F9 FT B1021.2
3	F9 FT B1031.2

Total Number of Successful and Failure Mission Outcomes

- The total number of successful and failure mission outcomes

List the total number of successful and failure mission outcomes

In [16]:
```
task_7a = '''
        SELECT COUNT(MissionOutcome) AS SuccessOutcome
        FROM SpaceX
        WHERE MissionOutcome LIKE 'Success%'
        '''

task_7b = '''
        SELECT COUNT(MissionOutcome) AS FailureOutcome
        FROM SpaceX
        WHERE MissionOutcome LIKE 'Failure%'
        '''
print('The total number of successful mission outcome is:')
display(create_pandas_df(task_7a, database=conn))
print()
print('The total number of failed mission outcome is:')
create_pandas_df(task_7b, database=conn)
```

```
The total number of successful mission outcome is:
```

	successoutcome
0	100

```
The total number of failed mission outcome is:
```

Out[16]:

	failureoutcome
0	1

Boosters Carried Maximum Payload

- The names of the booster which have carried the maximum payload mass

List the names of the booster_versions which have carried the maximum payload mass. Use a subquery

In [17]:

```
task_8 = '''
        SELECT BoosterVersion, PayloadMassKG
        FROM SpaceX
        WHERE PayloadMassKG = (
                                SELECT MAX(PayloadMassKG)
                                FROM SpaceX
                                )
        ORDER BY BoosterVersion
        '''
create_pandas_df(task_8, database=conn)
```

Out[17]:

	boosterversion	payloadmasskg
0	F9 B5 B1048.4	15600
1	F9 B5 B1048.5	15600
2	F9 B5 B1049.4	15600
3	F9 B5 B1049.5	15600
4	F9 B5 B1049.7	15600
5	F9 B5 B1051.3	15600
6	F9 B5 B1051.4	15600
7	F9 B5 B1051.6	15600
8	F9 B5 B1056.4	15600
9	F9 B5 B1058.3	15600
10	F9 B5 B1060.2	15600
11	F9 B5 B1060.3	15600

2015 Launch Records

- The failed landingoutcomes in drone ship, their booster versions, and launch site names for in year 2015

List the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015

In [18]:
```
task_9 = '''
        SELECT BoosterVersion, LaunchSite, LandingOutcome
        FROM SpaceX
        WHERE LandingOutcome LIKE 'Failure (drone ship)'
            AND Date BETWEEN '2015-01-01' AND '2015-12-31'
        '''
create_pandas_df(task_9, database=conn)
```

Out[18]:

	boosterversion	launchsite	landingoutcome
0	F9 v1.1 B1012	CCAFS LC-40	Failure (drone ship)
1	F9 v1.1 B1015	CCAFS LC-40	Failure (drone ship)

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

- 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

Rank the count of landing outcomes (such as Failure (drone ship) or Success (ground pad))

In [19]:

```
task_10 = '''
        SELECT LandingOutcome, COUNT(LandingOutcome)
        FROM SpaceX
        WHERE DATE BETWEEN '2010-06-04' AND '2017-03-20'
        GROUP BY LandingOutcome
        ORDER BY COUNT(LandingOutcome) DESC
        '''
create_pandas_df(task_10, database=conn)
```

Out[19]:

	landingoutcome	count
0	No attempt	10
1	Success (drone ship)	6
2	Failure (drone ship)	5
3	Success (ground pad)	5
4	Controlled (ocean)	3
5	Uncontrolled (ocean)	2
6	Precluded (drone ship)	1
7	Failure (parachute)	1

Section 3

Launch Sites Proximities Analysis

All launch sites global map markers

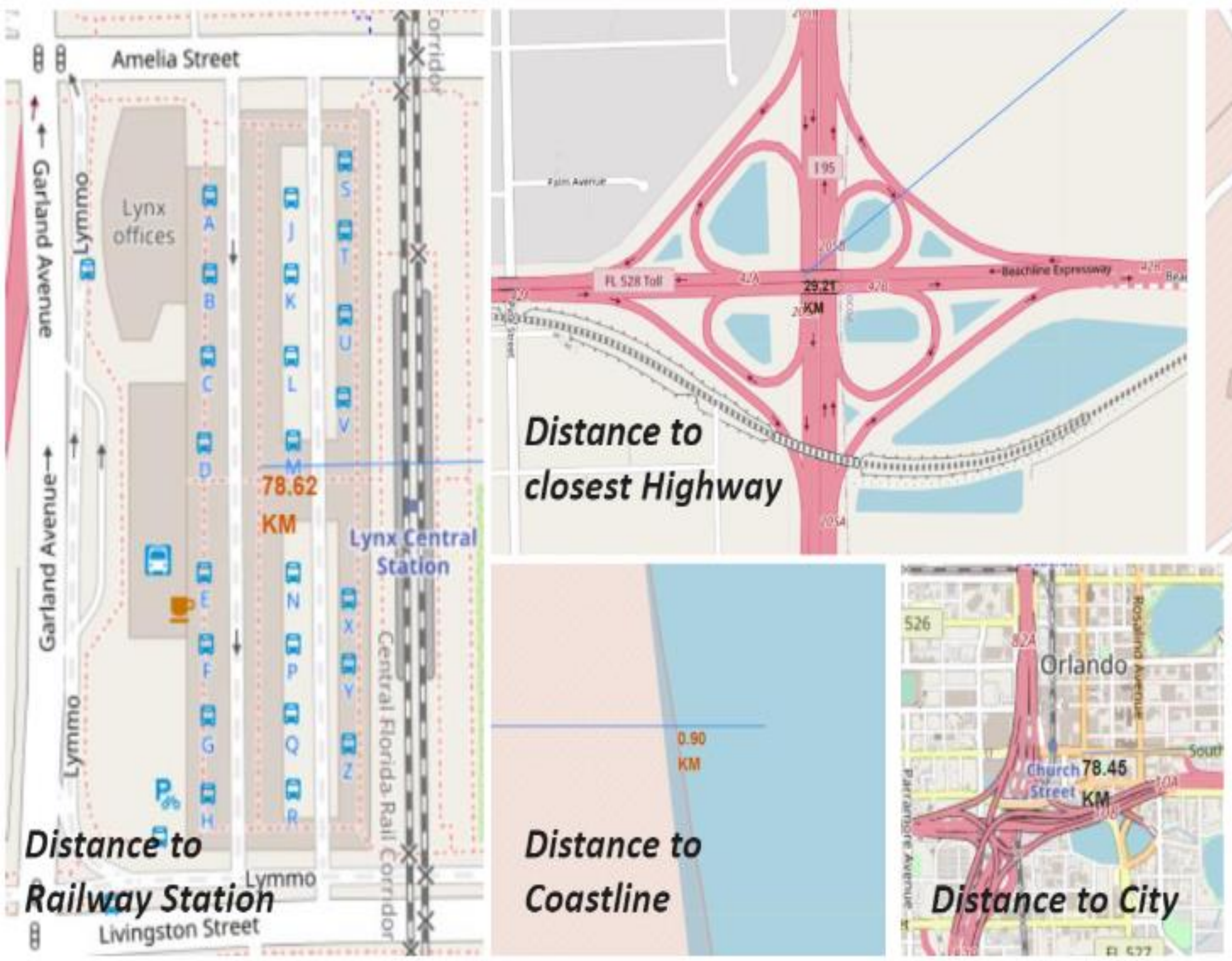
Markers showing launch sites with color labels

Florida Launch Sites

Green Marker shows successful Launches and Red Marker shows Failures

California Launch Site

Launch Site distance to landmarks

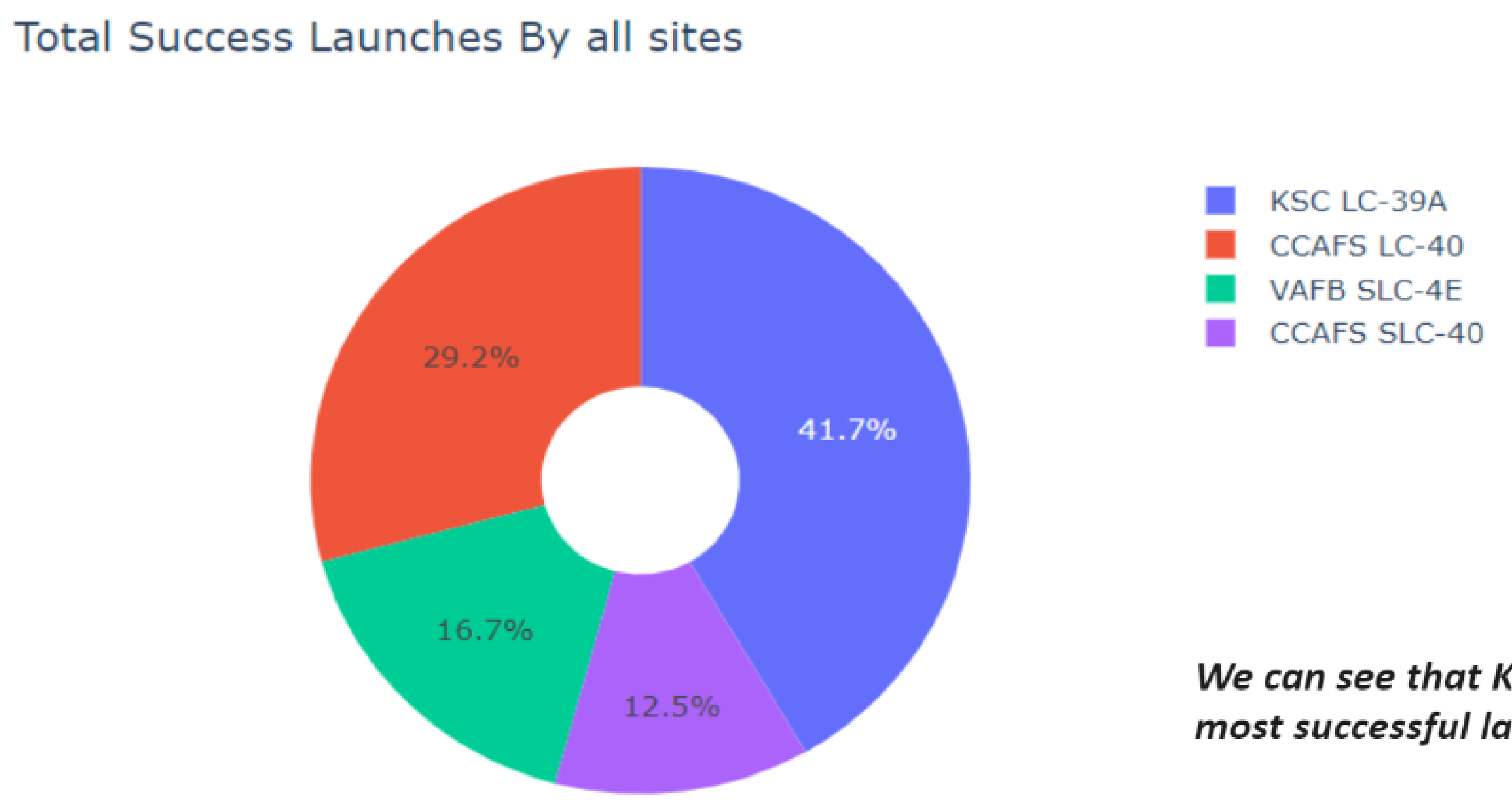


- Are launch sites in close proximity to railways? No
- Are launch sites in close proximity to highways? No
- Are launch sites in close proximity to coastline? Yes
- Do launch sites keep certain distance away from cities? Yes

Section 4

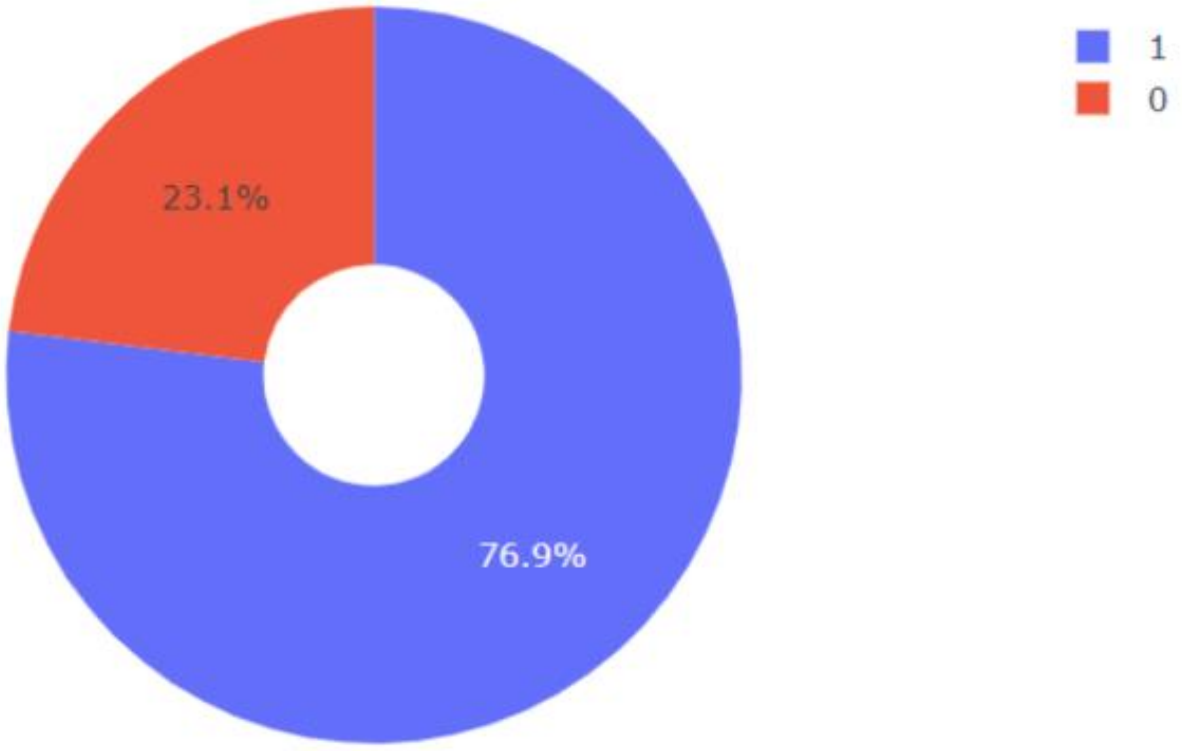
Build a Dashboard with Plotly Dash

Pie chart showing the success percentage achieved by each launch site



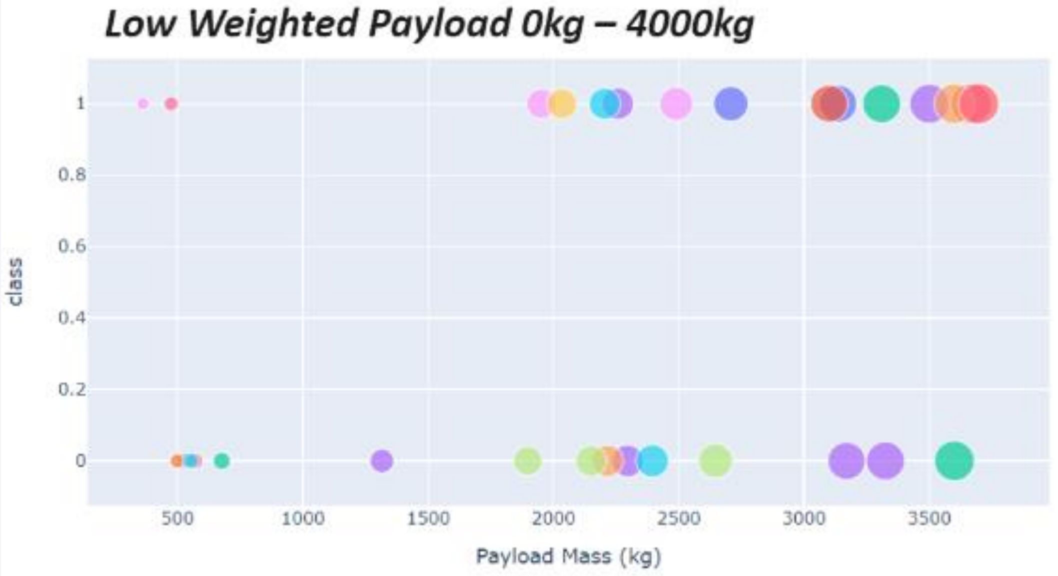
We can see that KSC LC-39A had the most successful launches from all the sites

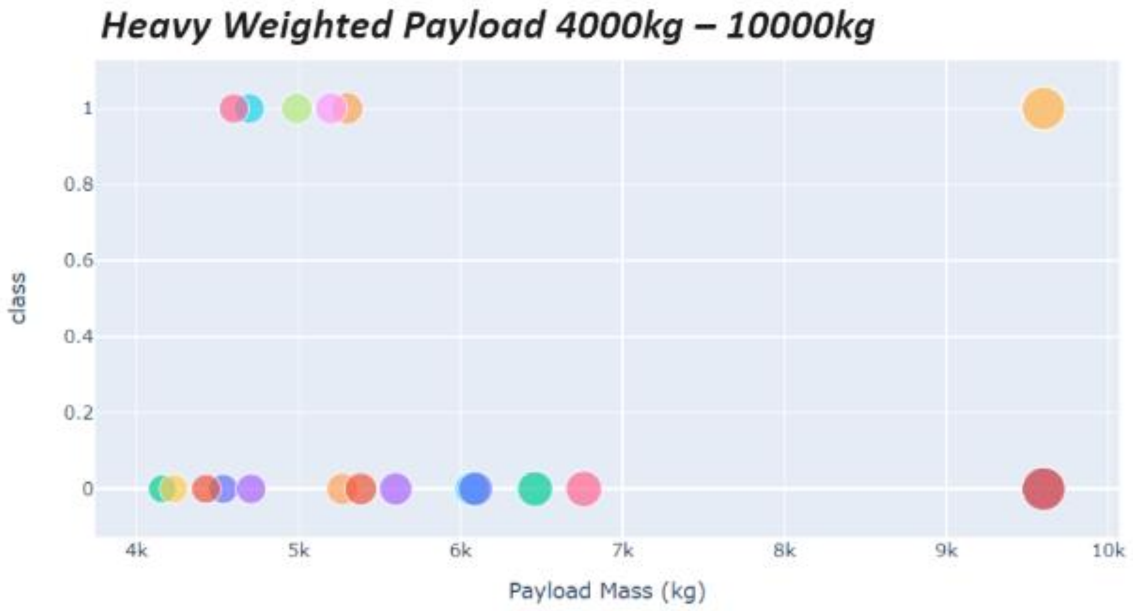
Pie chart showing the Launch site with the highest launch success ratio



KSC LC-39A achieved a 76.9% success rate while getting a 23.1% failure rate

Scatter plot of Payload vs Launch Outcome for all sites, with different payload selected in the range slider



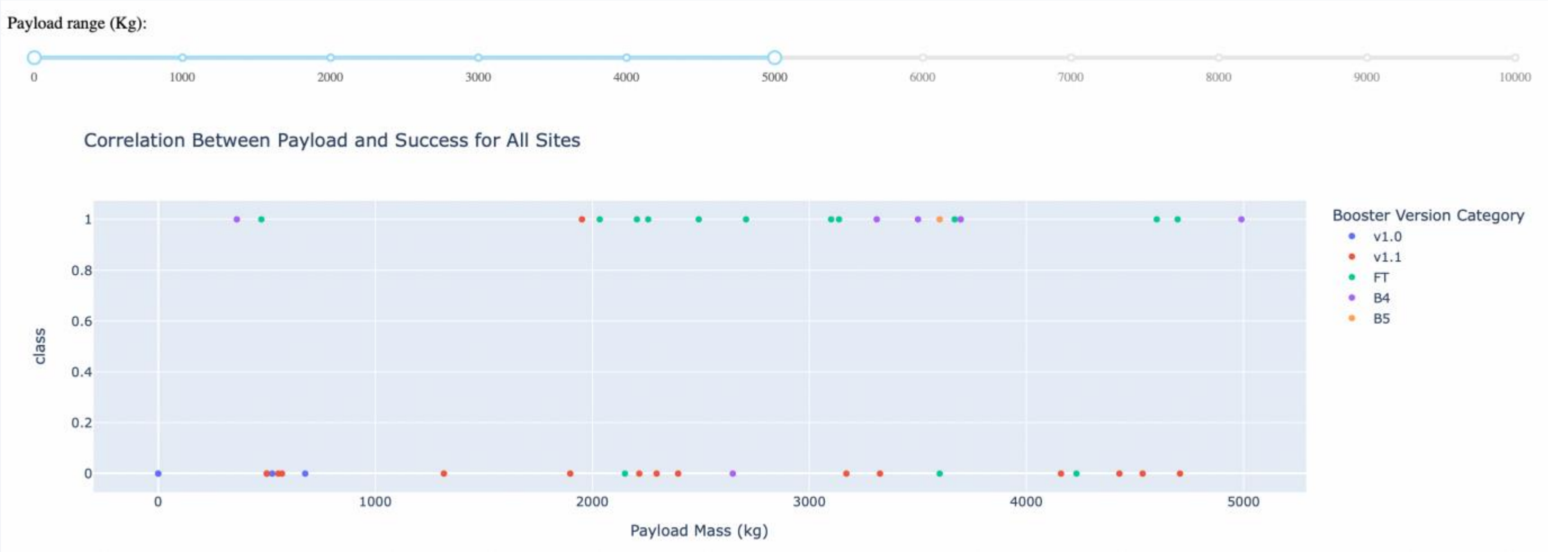


We can see the success rates for low weighted payloads is higher than the heavy weighted payloads

Section 5

Predictive Analysis (Classification)

Classification Accuracy



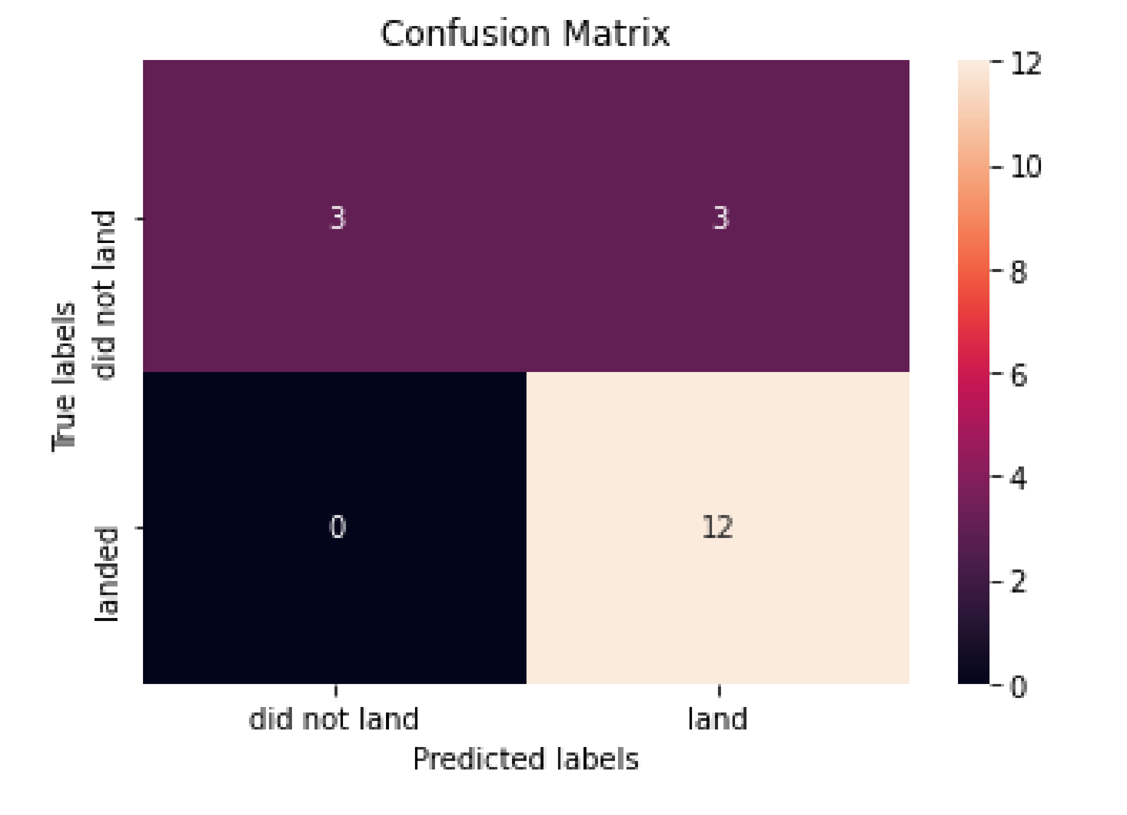
Explanation:

- The charts show that payloads between 2000 and 5500kg have the highest success rate.

Confusion Matrix

Explanation:

- Examining the confusion matrix, we see that logistic regression can distinguish between the different classes. We see that the major problem is false positives



Conclusions

- Decision Tree Model is the best algorithm for this dataset.
- Launches with a low payload mass show better results than launches with a larger payload mass.
- Most of launch sites are in proximity to the Equator line and all the sites are in very close proximity to the coast.
- The success rate of launches increases over the years.
- KSC LC-39A has the highest success rate of the launches from all the sites.
- Orbits ES-L1, GEO, HEO and SSO have 100% success rate.

Appendix

- https://github.com/kanchanaduck/Applied-Data-Science-Capstone.git

Thank you!