Winning Space Race with Data Science

Yuexin Liu
2/6/2026

Outline

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

Executive Summary

- Summary of methodologies

In this project, I applied the data science methodologies such as data collection, data wrangling, machine learning mode training and evaluation to predict if the SpaceX Falcon 9 first stage will land successfully.

- Summary of all results

The SpaceX Falcon 9 landing outcome depends upon various factor such as launch site, boosters, customers and payload mass. Utilizing tree classifier, we can predict the landing outcome of SpaceX Falcon 9 with a predictive accuracy of around 83.3%.

Introduction

- Project background and context

In this capstone, we will predict if the Falcon 9 first stage will land successfully. 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.

- Problems you want to find answers

How can we make sure that the SpaceX Falcon 9 first stage land successfully?

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Request rocket launch data from SpaceX API
- Perform data wrangling
  - Describe how data was processed
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Build multiple machine learning models and use GridSearch to find out the best hyperparameters

Data Collection

- Describe how data sets were collected.
- You need to present your data collection process use key phrases and flowcharts

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

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

response = requests.get(spacex_url)
```

https://github.com/yx1234liu/IBM_DataScienceProject/blob/main/jupyter-labs-spacex-data-collection-api.ipynb

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

```
response = requests.get(static_url,headers = headers)

soup = BeautifulSoup(response.text,'html.parser')
```

https://github.com/yx1234liu/IBM_DataScienceProject/blob/main/jupyter-labs-webscraping.ipynb

Data Wrangling

- Describe how data were processed

we will perform some Exploratory Data Analysis (EDA) to find some patterns in the data and determine what would be the label for training supervised models.

- You need to present your data wrangling process using key phrases and flowcharts

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

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

- Add the GitHub URL of your completed data wrangling related notebooks, as an external reference and peer-review purpose

https://github.com/yx1234liu/IBM_DataScienceProject/blob/main/labs-jupyter-spacex-Data%20wrangling.ipynb

EDA with Data Visualization

- Summarize what charts were plotted and why you used those charts

Scatter plot

Bar chart

Line chart

- Add the GitHub URL of your completed EDA with data visualization notebook, as an external reference and peer-review purpose

https://github.com/yx1234liu/IBM_DataScienceProject/blob/main/edadataviz.ipynb

EDA with SQL

- Using bullet point format, summarize the SQL queries you performed
    - Display the names of the unique launch sites in the space mission
    - Display 5 records where launch sites begin with the string 'CCA'
    - Display the total payload mass carried by boosters launched by NASA (CRS)
    - Display average payload mass carried by booster version F9 v1.1
    - List the date when the first succesful landing outcome in ground pad was acheived.
    - List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000
- Add the GitHub URL of your completed EDA with SQL notebook, as an external reference and peer-review purpose

https://github.com/yx1234liu/IBM_DataScienceProject/blob/main/jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

- Summarize what map objects such as markers, circles, lines, etc. you created and added to a folium map
  - ❖ **TASK 1:** Mark all launch sites on a map
  - ❖ **TASK 2:** Mark the success/failed launches for each site on the map
  - ❖ **TASK 3:** Calculate the distances between a launch site to its proximities

- Add the GitHub URL of your completed interactive map with Folium map, as an external reference and peer-review purpose

https://github.com/yx1234liu/IBM_DataScienceProject/blob/main/lab_jupyter_launch_site_location.ipynb

Build a Dashboard with Plotly Dash

- Summarize what plots/graphs and interactions you have added to a dashboard
  - ❖ TASK 1: Add a Launch Site Drop-down Input Component
  - ❖ TASK 2: Add a callback function to render success-pie-chart based on selected site dropdown
  - ❖ TASK 3: Add a Range Slider to Select Payload
  - ❖ TASK 4: Add a callback function to render the success-payload-scatter-chart scatter plot

- Add the GitHub URL of your completed Plotly Dash lab, as an external reference and peer-review purpose

https://github.com/yx1234liu/IBM_DataScienceProject/blob/main/spacex-dash-app.py

Predictive Analysis (Classification)

- Summarize how you built, evaluated, improved, and found the best performing classification model

❖ Perform exploratory Data Analysis and determine Training Labels

❖ create a column for the class

❖ Standardize the data

❖ Split into training data and test data

❖ -Find best Hyperparameter for SVM, Classification Trees and Logistic Regression

❖ Find the method performs best using test data

- Add the GitHub URL of your completed predictive analysis lab, as an external reference and peer-review purpose

https://github.com/yx1234liu/IBM_DataScienceProject/blob/main/SpaceX_Machine%20Learning%20Prediction_Part_5.ipynb

Results

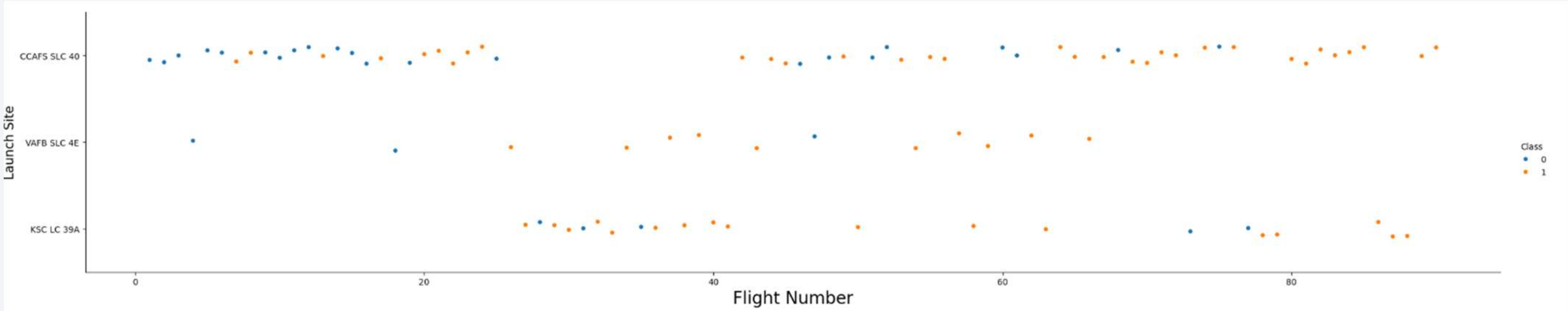
- Exploratory data analysis results
- Interactive analytics demo in screenshots
- Predictive analysis results

Section 2

Insights drawn from EDA

Flight Number vs. Launch Site

- Show a scatter plot of Flight Number vs. Launch Site

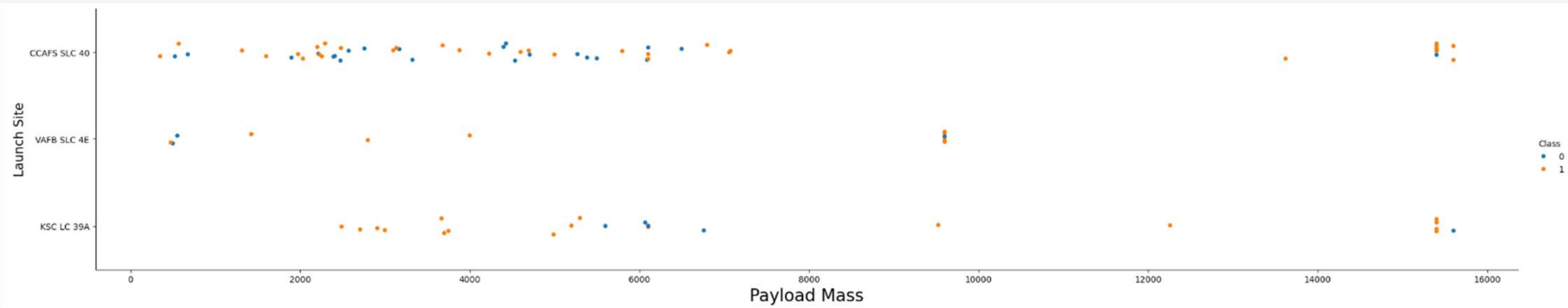


- Show the screenshot of the scatter plot with explanations

Only SLC40 has smaller flight number and the other two launch sites did not have smaller flight number. Also higher the flight number, the more the class of 1.

Payload vs. Launch Site

- Show a scatter plot of Payload vs. Launch Site

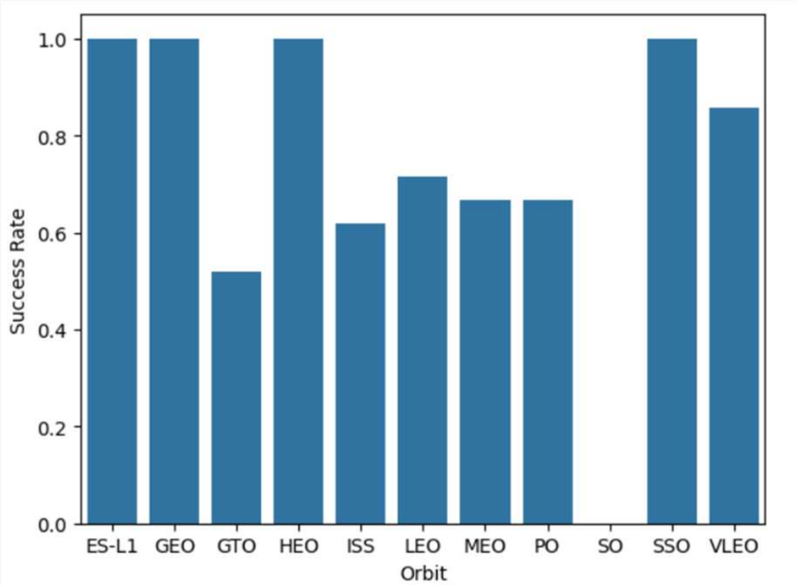


- Show the screenshot of the scatter plot with explanations

The launch sites did not have a very higher payload mass.

Success Rate vs. Orbit Type

- Show a bar chart for the success rate of each orbit type

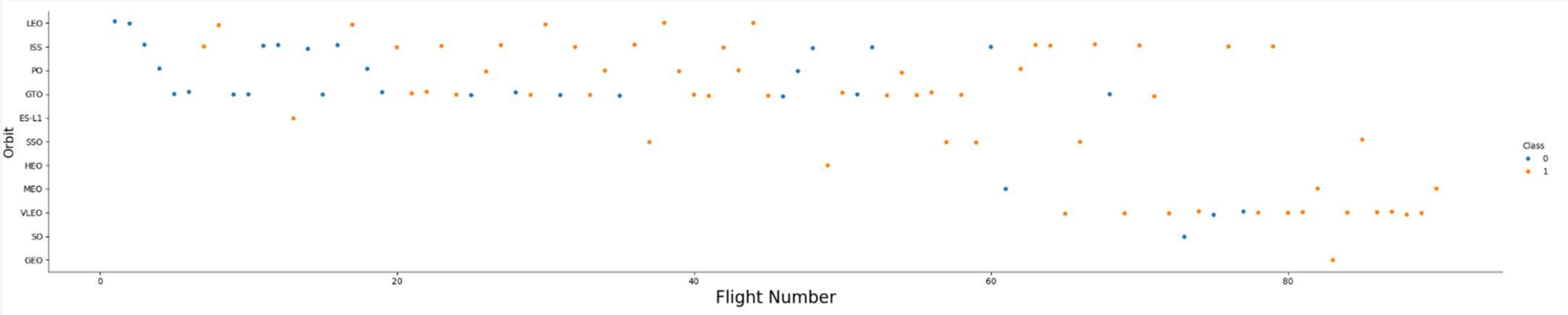


- Show the screenshot of the scatter plot with explanations

The SO orbit had zero success rate, and ES-L1, GEO, HEO, and SSO almost had a perfect success rate.

Flight Number vs. Orbit Type

- Show a scatter point of Flight number vs. Orbit type

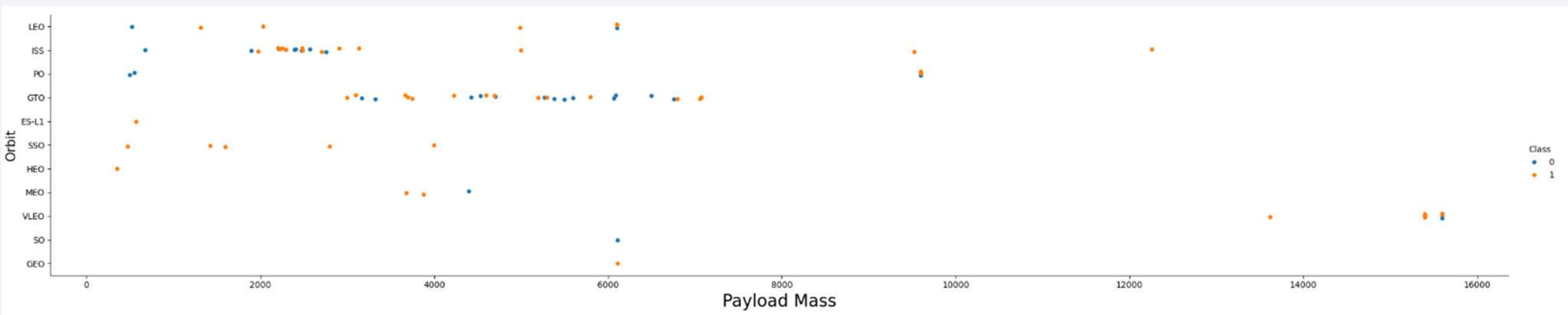


- Show the screenshot of the scatter plot with explanations

Some orbits had more flight numbers than the other orbits.

Payload vs. Orbit Type

- Show a scatter point of payload vs. orbit type



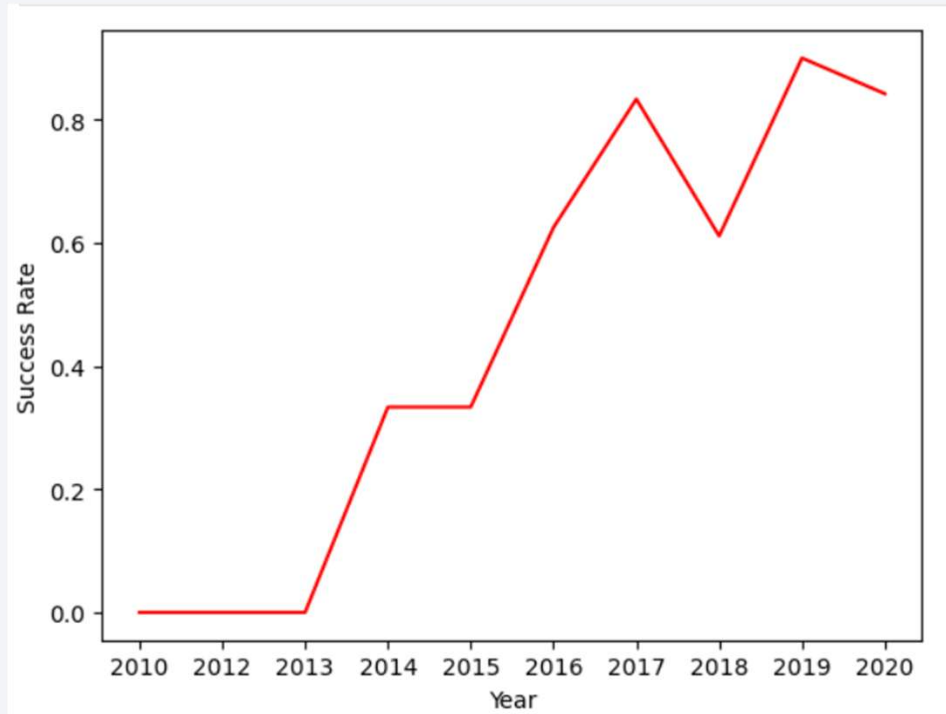
- Show the screenshot of the scatter plot with explanations

Some orbits didn't have higher payload mass.

Launch Success Yearly Trend

- Show a line chart of yearly average success rate
- Show the screenshot of the scatter plot with explanations

The sucess rate since 2013 kept increasing till 2020



All Launch Site Names

- Find the names of the unique launch sites
- Present your query result with a short explanation here

```
%sql SELECT DISTINCT Launch_Site FROM SPACEXTABLE

 * sqlite:///my_data1.db
Done.
```

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`
- Present your query result with a short explanation here

```
%sql SELECT * FROM SPACEXTABLE WHERE launch_site LIKE 'CCA%' LIMIT 5
```

```
 * sqlite:///my_data1.db
Done.
```

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
- Present your query result with a short explanation here

```
In [15]: %sql SELECT SUM(PAYLOAD_MASS__KG_) AS total_payload_mass FROM SPACEXTABLE WHERE Customer = 'NASA (CRS)'

 * sqlite:///my_data1.db
Done.
```

Out[15]:

total_payload_mass
45596

Average Payload Mass by F9 v1.1

- Calculate the average payload mass carried by booster version F9 v1.1
- Present your query result with a short explanation here

```
In [16]:  %sql SELECT AVG(PAYLOAD_MASS__KG_) AS avg_payload_mass FROM SPACEXTABLE WHERE Booster_Version LIKE 'F9 v1.1%'

 * sqlite:///my_data1.db
Done.
```

Out[16]:

avg_payload_mass
2534.6666666666665

First Successful Ground Landing Date

- Find the dates of the first successful landing outcome on ground pad
- Present your query result with a short explanation here

```
In [19]: %sql SELECT MIN(Date) AS first_successful_lanidng_date FROM SPACEXTABLE WHERE Landing_Outcome = 'Success (ground pad)'

 * sqlite:///my_data1.db
Done.
```

Out[19]:

first_successful_lanidng_date
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

- Present your query result with a short explanation here

In [20]:
```
%sql SELECT Booster_Version FROM SPACEXTABLE WHERE Landing_Outcome = 'Success (drone ship)' AND PAYLOAD_MASS__KG_>4000 AND PAYLOAD_MASS__KG_<6000
```

```
 * sqlite:///my_data1.db
Done.
```

Out[20]:

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
- Present your query result with a short explanation here

```
%sql SELECT "Landing_Outcome", COUNT(*) AS total_count FROM SPACEXTABLE GROUP BY "Landing_Outcome"
```

```
 * sqlite:///my_data1.db
Done.
```

Landing_Outcome	total_count
Controlled (ocean)	5
Failure	3
Failure (drone ship)	5
Failure (parachute)	2
No attempt	21
No attempt	1
Precluded (drone ship)	1
Success	38
Success (drone ship)	14
Success (ground pad)	9
Uncontrolled (ocean)	2

Boosters Carried Maximum Payload

- List the names of the booster which have carried the maximum payload mass
- Present your query result with a short explanation here

```
%sql SELECT Booster_Version FROM SPACEXTABLE WHERE PAYLOAD_MASS__KG_=(SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTABLE);
```

```
 * sqlite:///my_data1.db
Done.
```

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

- Present your query result with a short explanation here

```
%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" = 'Failure (drone ship)';
```

```
 * sqlite:///my_data1.db
Done.
```

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

- 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

- Present your query result with a short explanation here

```
%%sql SELECT Landing_Outcome, COUNT(*) AS outcome_count
FROM SPACEXTABLE WHERE Date BETWEEN '2010-06-04' AND '2017-03-20'
GROUP BY Landing_Outcome
ORDER BY outcome_count DESC;
```

```
 * sqlite:///my_data1.db
Done.
```

Landing_Outcome	outcome_count
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

<Folium Map Screenshot 1>

- Replace <Folium map screenshot 1> title with an appropriate title

- Explore the generated folium map and make a proper screenshot to include all launch sites' location markers on a global map

- Explain the important elements and findings on the screenshot

<Folium Map Screenshot 2>

- Replace <Folium map screenshot 2> title with an appropriate title
- Explore the folium map and make a proper screenshot to show the color-labeled launch outcomes on the map
- Explain the important elements and findings on the screenshot

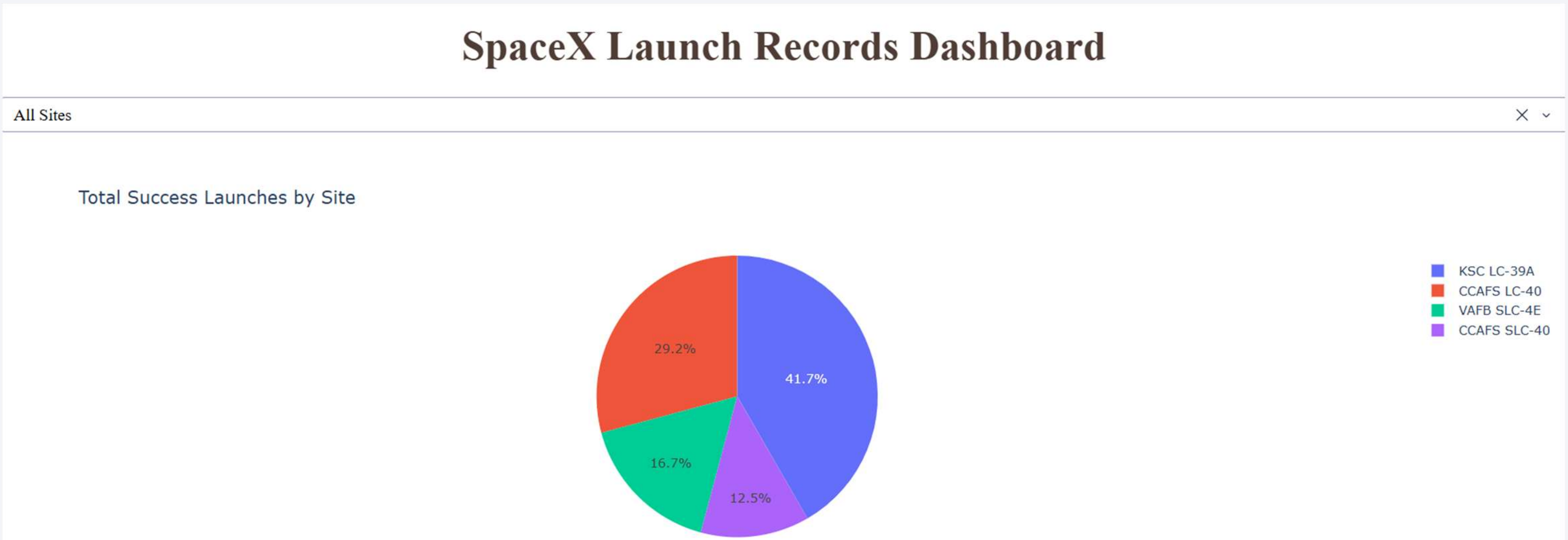
<Folium Map Screenshot 3>

- Replace <Folium map screenshot 3> title with an appropriate title
- Explore the generated folium map and show the screenshot of a selected launch site to its proximities such as railway, highway, coastline, with distance calculated and displayed
- Explain the important elements and findings on the screenshot

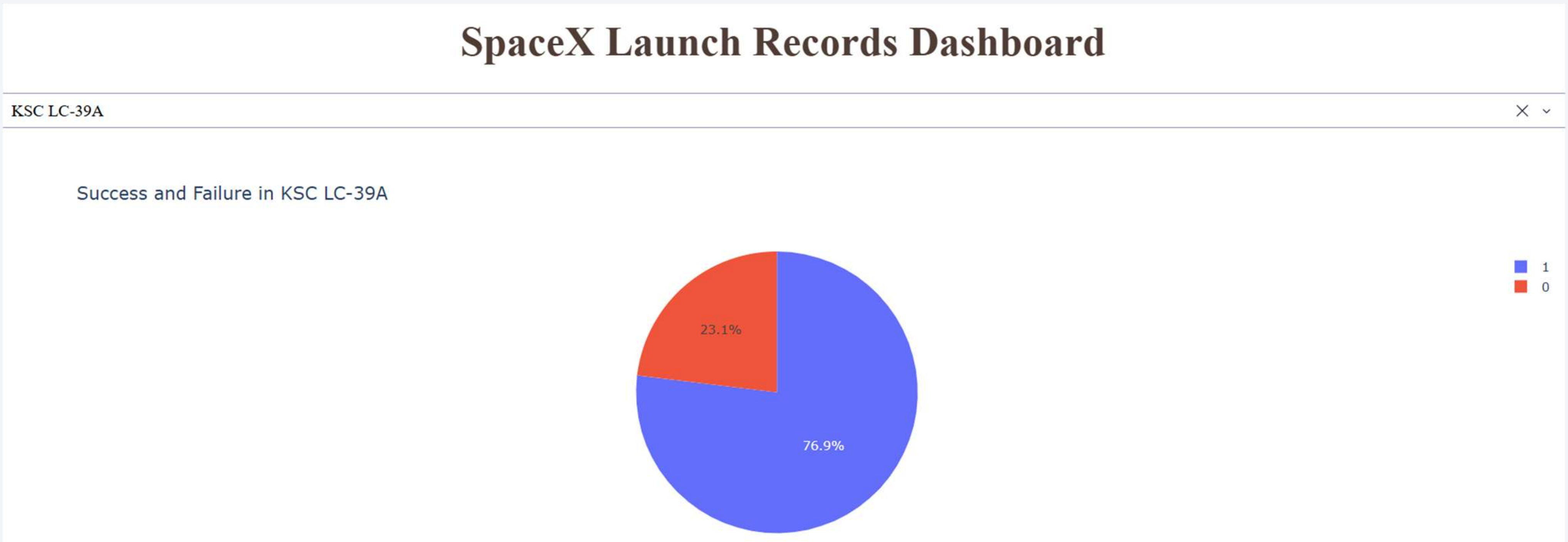
Section 4

Build a Dashboard with Plotly Dash

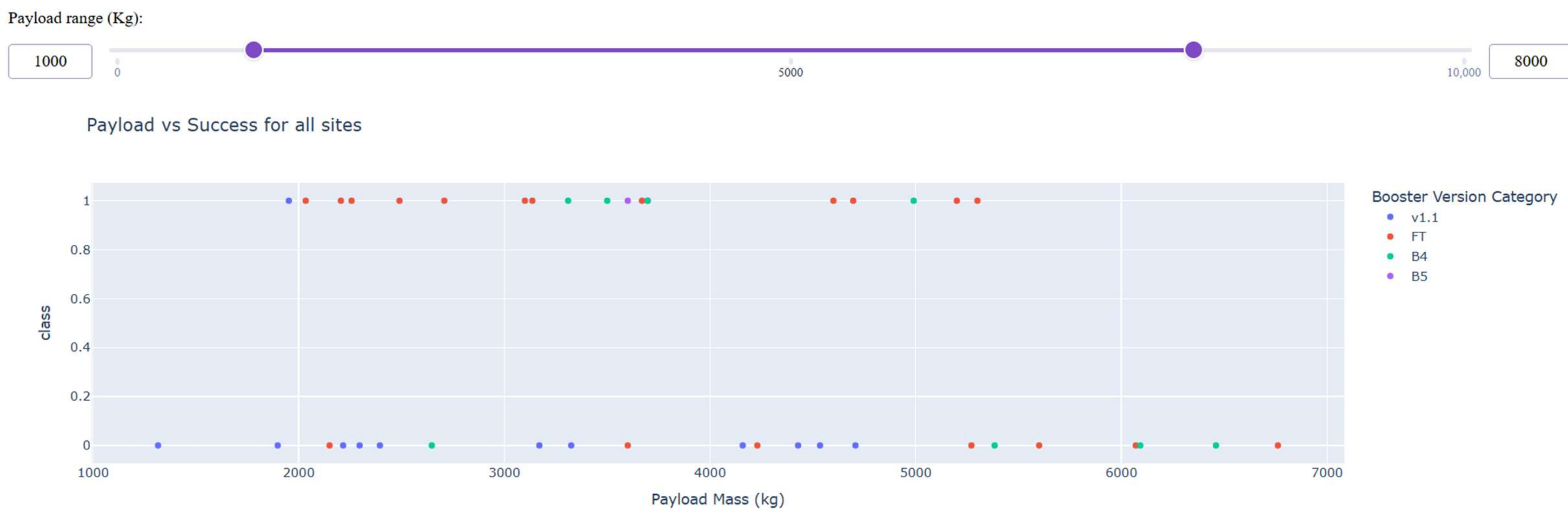
Total Success Launches by Site



Success and Failure in KSC LC-39A site



Payload vs Success for all sites



Section 5

Predictive Analysis (Classification)

Classification Accuracy

- Visualize the built model accuracy for all built classification models, in a bar chart

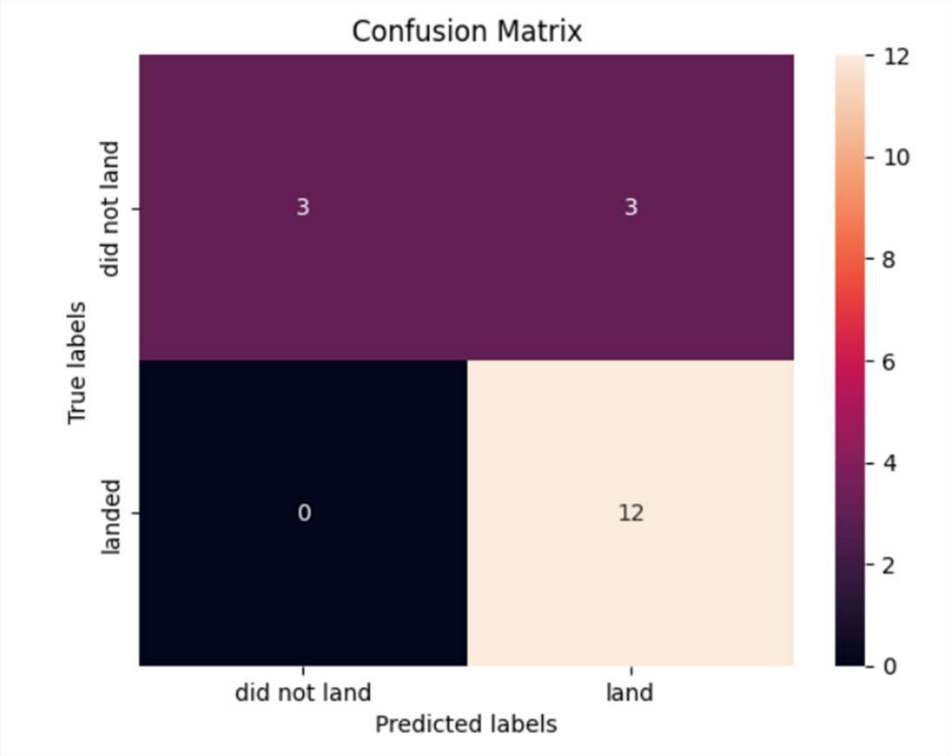
- Find which model has the highest classification accuracy

The decision tree classifier had the highest classification accuracy.

Models	accuracy
Logistic regression	84.60%
Support Vector Machine	84.80%
Decision Tree	87.50%
K Nearest Neighbors	84.80%

Confusion Matrix

- Show the confusion matrix of the best performing model with an explanation



Conclusions

- Utilizing data science methodologies, we will be able to predict if the SpaceX Falcon 9 first stage will land successfully with a high accuracy.
- Several features or factors were identified to be significantly associated the SpaceX landing and used as predictors in building the machine learning models.
- Multiple exploratory data analysis techniques such as Dashboard and interactive website were employed to examine and help understand the data set.

Appendix

- Include any relevant assets like Python code snippets, SQL queries, charts, Notebook outputs, or data sets that you may have created during this project

All the python code were included in the GitHub:

https://github.com/yx1234liu/IBM_DataScienceProject


Thank you!