SpaceX Falcon 9 Rocket Landing Prediction

Suparna Chowdhury

January 25, 2023

IBM Developer

SKILLS NETWORK

OUTLINE

Project Github link

- Executive Summary
- Introduction
- Methodology
- Results
  - Visualization – Charts
  - Dashboard
  - Predictive Analysis
- Discussion
- Conclusion

EXECUTIVE SUMMARY

- Logistic regression performs best with a prediction accuracy of 94.44%
- Rocket with lighter payload have higher success rate than rocket with heavier payload
- The landing success probability increases with time
- KSC LC 39A site has the highest launch success rate of 76%
- Orbit ES L1, SSO, HEO, and GEO has the best success rate of 100%

INTRODUCTION

- **Background**
  - SpaceX launches their rockets at a cost of $62 million which is almost one-third of other company’s cost.
  - SpaceX lowers the overall rocket launch cost by recovering and reusing the first stage boosters.
  - Successful landing of the Falcon 9 first stage is crucial
- **Problem Statement**
  - The objective of this capstone to predict if the Falcon 9 first stage will land successfully.

METHODOLOGY - SUMMARY

- Data Collection
    - Request to the SpaceX API
    - Web scrap Falcon 9 launch records HTML table from Wikipedia Data Wrangling
- Perform exploratory Data Analysis using SQL and visualization
- Innteractive visualization usinf Folium and Plotly Dashboard
- Predicative Analysis using classification algorithms
    - Logistic regression, SVM, KNN, Decision Tree
    - Find the best classifier.

METHODOLOGY

- ## Data Collection

  - We collect data from the SpaceX REST API by making a get request and receive the JSON data
  - We convert the JSON data to a dataframe by using the json_normalize function.
  - We also web scrap Wiki pages to obtain Falcon 9 launch data using Python BeautifulSoup package.
  - We parse data from the HTML tables and convert it to pandas dataframe.
  - We need to filter data which includes both Falcon 1 and Falcon 9 data to separate Falcon 9 booster info.

Data Collection from SpaceX API

Request and parse the SpaceX launch data using the GET request

```
spacex_url="https://api.spacexdata.com/v4/launches/past"
response = requests.get(spacex_url)
response.status_code
```

```
200
```

Converting to pandas Dataframe

```
data = pd.json_normalize(response.json())
```

Applying custom function to clean the data

```
# Call getLaunchSite
getLaunchSite(data)
# Call getPayloadData
getPayloadData(data)
# Call getCoreData
getCoreData(data)
```

Creating dataframe

```
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}
data = pd.DataFrame(launch_dict)
```

Data Collection from Wiki

Request the Falcon9 Launch Wiki page from its URL

```
html_data = requests.get(static_url).text
soup = BeautifulSoup(html_data, 'html.parser')
html_tables = soup.find_all('table')
```

Extracting Column name

```
column_names = []

for header in first_launch_table.find_all('th'):
    col_name = extract_column_from_header(header)
    if col_name is not None and len(col_name) > 0:
        column_names.append(col_name)

launch_dict= dict.fromkeys(column_names)
```

Appending data to key of the dictionary

```
extracted_row = 0
for table_number,table in enumerate(soup.find_all('table',"wikital
    for rows in table.find_all("tr"):
        if rows.th:
            if rows.th.string:
                flight_number=rows.th.string.strip()
                flag=flight_number.isdigit()
        else:
            flag=False
        row=rows.find_all('td')
        if flag:
            extracted_row += 1
            launch_dict['Flight No.'].append(int(flight_number))
            datatimelist=date_time(row[0])
            date = datatimelist[0].strip(',')
            launch_dict['Date'].append(date)
            time = datatimelist[1]
            launch_dict['Time'].append(time)
            bv=booster_version(row[1])
            if not(bv):
                bv=row[1].a.string
```

METHODOLOGY

- **Exploratory Data Analysis**
  - Calculate the number of launches on each site
  - Calculate the number and occurrence of each orbit
  - Calculate the number and occurance of mission outcome per orbit type
  - Create a landing outcome label from Outcome column

METHODOLOGY

- **Exploratory Data Analysis with SQL**
  - 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 successful landing outcome in ground pad was achieved.

METHODOLOGY

- **Exploratory Data Analysis with SQL**

  - List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000
  - List the total number of successful and failure mission outcomes
  - List the names of the booster_versions which have carried the maximum payload mass.
  - List the records which will display the month names, failure landing_outcomes in drone ship ,booster versions, launch_site for the months in year 2015
  - Rank the count of successful landing_outcomes between the date 04-06-2010 and 20-03-2017 in descending order

METHODOLOGY

- **Interactive Map with Folium**

  - Visualize the launch sites in an interactive map using Folium

  - Put a circle marker and label around the launch sites using lattitude and longitude coordinates data of the sites.

METHODOLOGY

- **Interactive Dashboard using Plotly Dash**

  - Pie chart showing the total launches of all sites and success rate of individual site separately.

  - Scatter plot showing the payload masses for different booster versions.

METHODOLOGY - PREDICTIVE ANALYSIS

- **Building model**

  - Load the data using Pandas
  - Split the data into training and test set
  - Choose the machine learning algorithms
  - Machine learning model applied: Logistic regression, KNN, SVM, and Decision tree.

METHODOLOGY - PREDICTIVE ANALYSIS

- We tuned the hyperparameters using GridSearchCV to obtain the most accurate model.
- With best parameters, we evaluate their score , plot confusion matrix.
- Logistic regression model has the best score accuracy of 94.44%.

```
parameters ={'C':[0.01,0.1,1],
             'penalty':['l2'],
             'solver':['lbfgs']}
```

```
lr=LogisticRegression()
logreg_cv = GridSearchCV(lr,
        param_grid=parameters, cv= 10)
logreg_cv.fit(X_train,Y_train)
```

```
GridSearchCV(cv=10, estimator=LogisticRegression(),
             param_grid={'C': [0.01, 0.1, 1], 'penalty': ['l2'],
                         'solver': ['lbfgs']})
```

```
print("tuned hpyerparameters :(best parameters) ",
      logreg_cv.best_params_)
print("accuracy :",logreg_cv.best_score_)
```

```
logreg_cv.score(X_test, Y_test)
```

```
0.9444444444444444
```

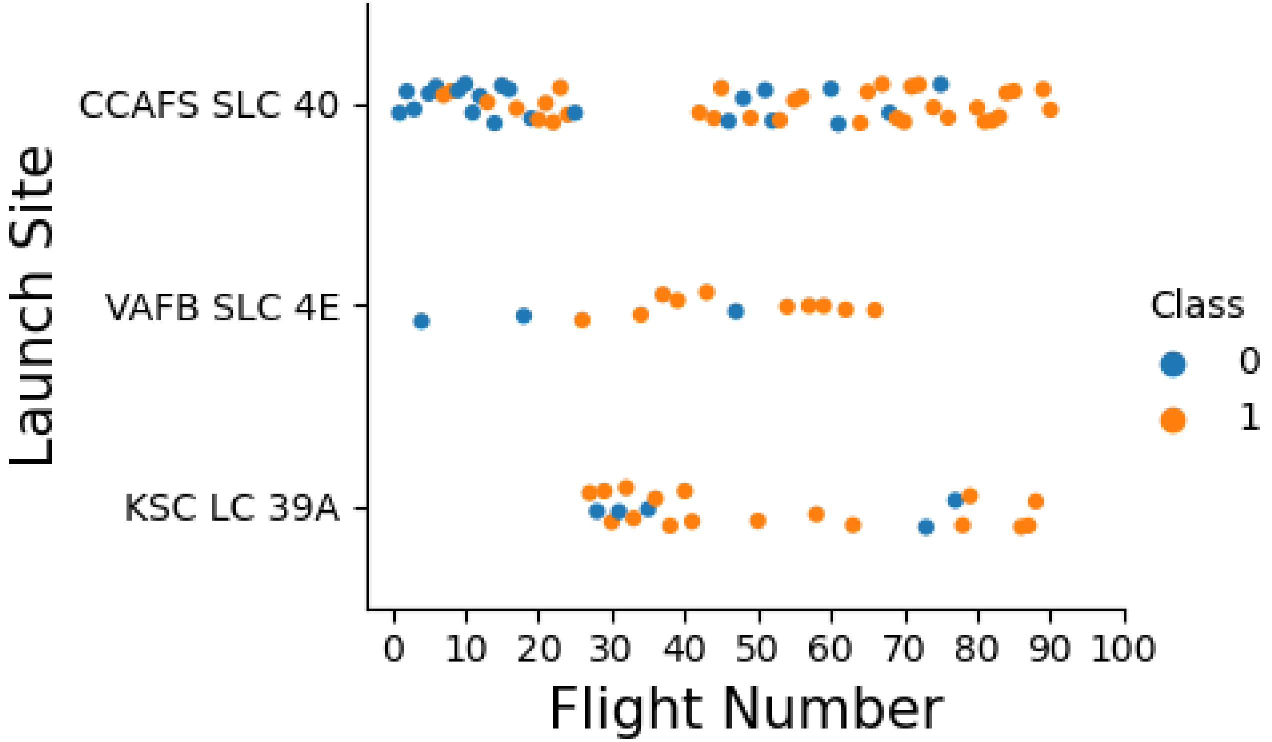
RESULTS

EXPLORATORY DATA ANALYSIS

Data visualization using seaborn

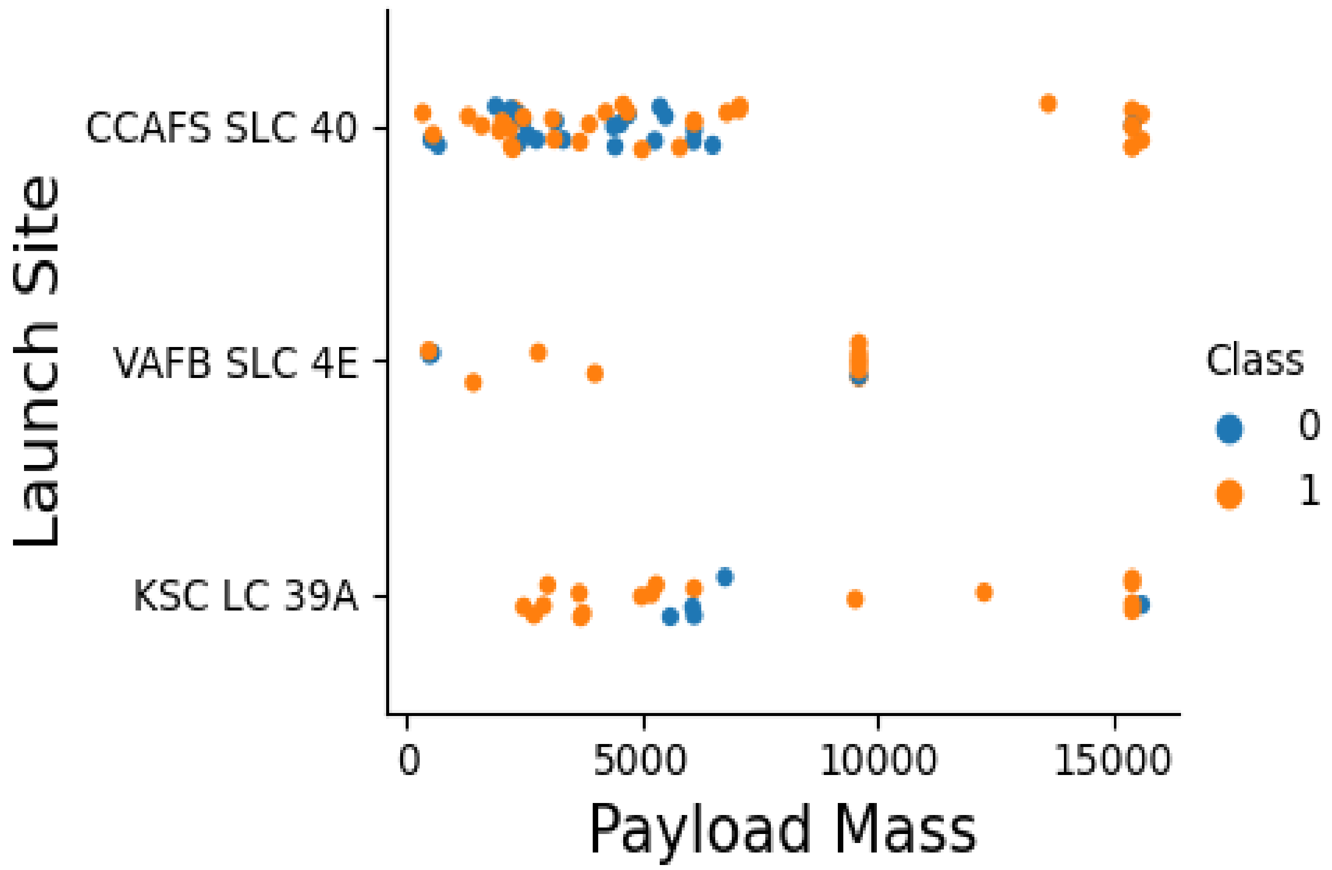
Flight Number vs Launch Site

- Most rockets are launched from CCAFS SLC 40 site.
- Fewer rockets are launched from VAFB SLC 4E SITE
- Success rate is greatest in KSC LC 39A site



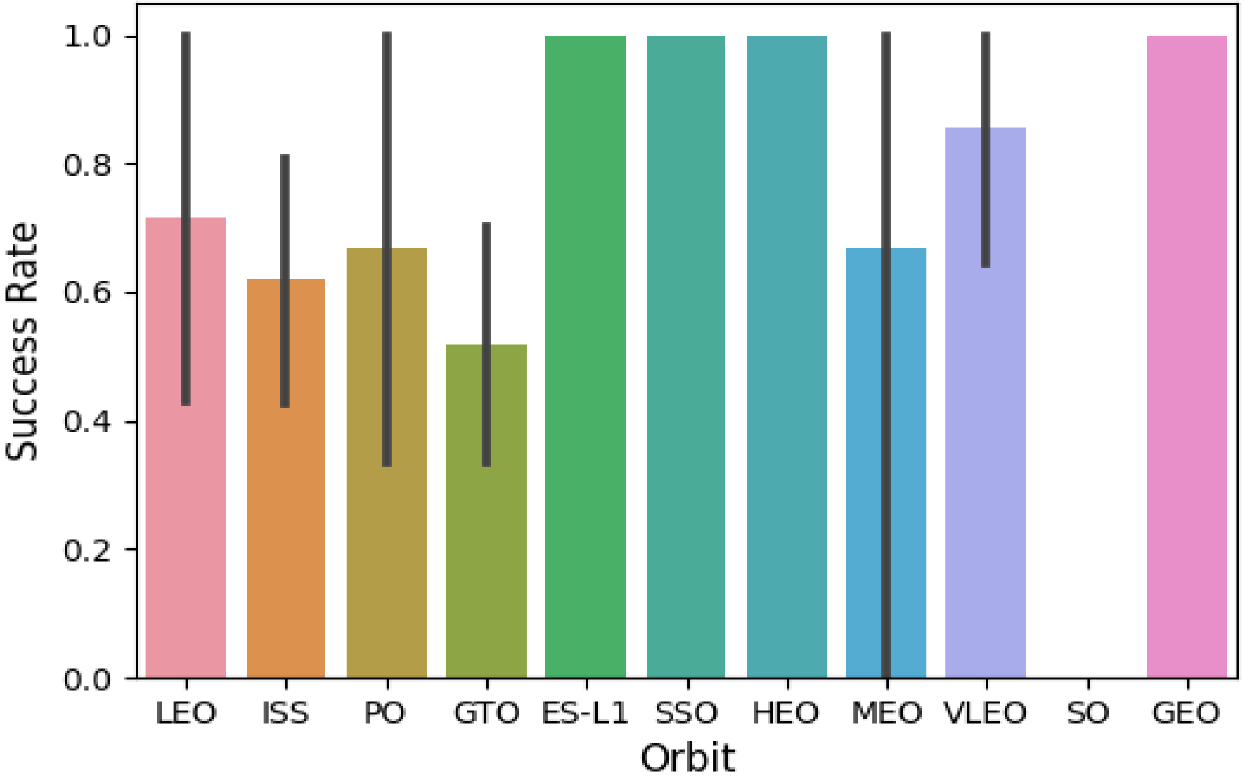
Payload Mass vs Launch Site

- No rockets greater than 10000 kg (heavy payload) launched in VAFB-SLC launch site.
- In CCAFS SLC 40 site, majority of the rocket launched are less than about 8000 kg
- In CCAFS SLC 40 site, in case of payload greater than 10000 kg, the success rate is 100%.



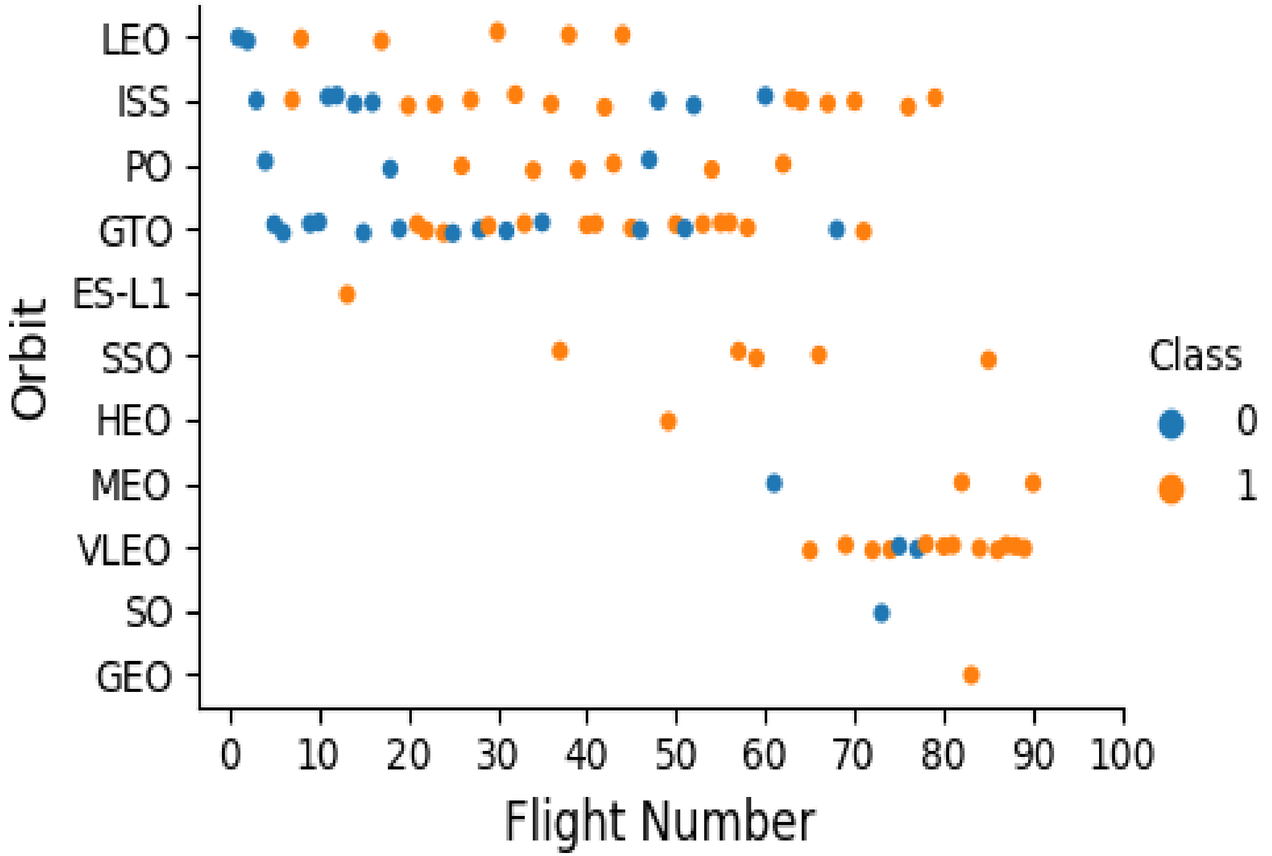
Success Rate vs Orbit

- The ES-L1, SSO, HEO, and GEO orbits have 100% success rate



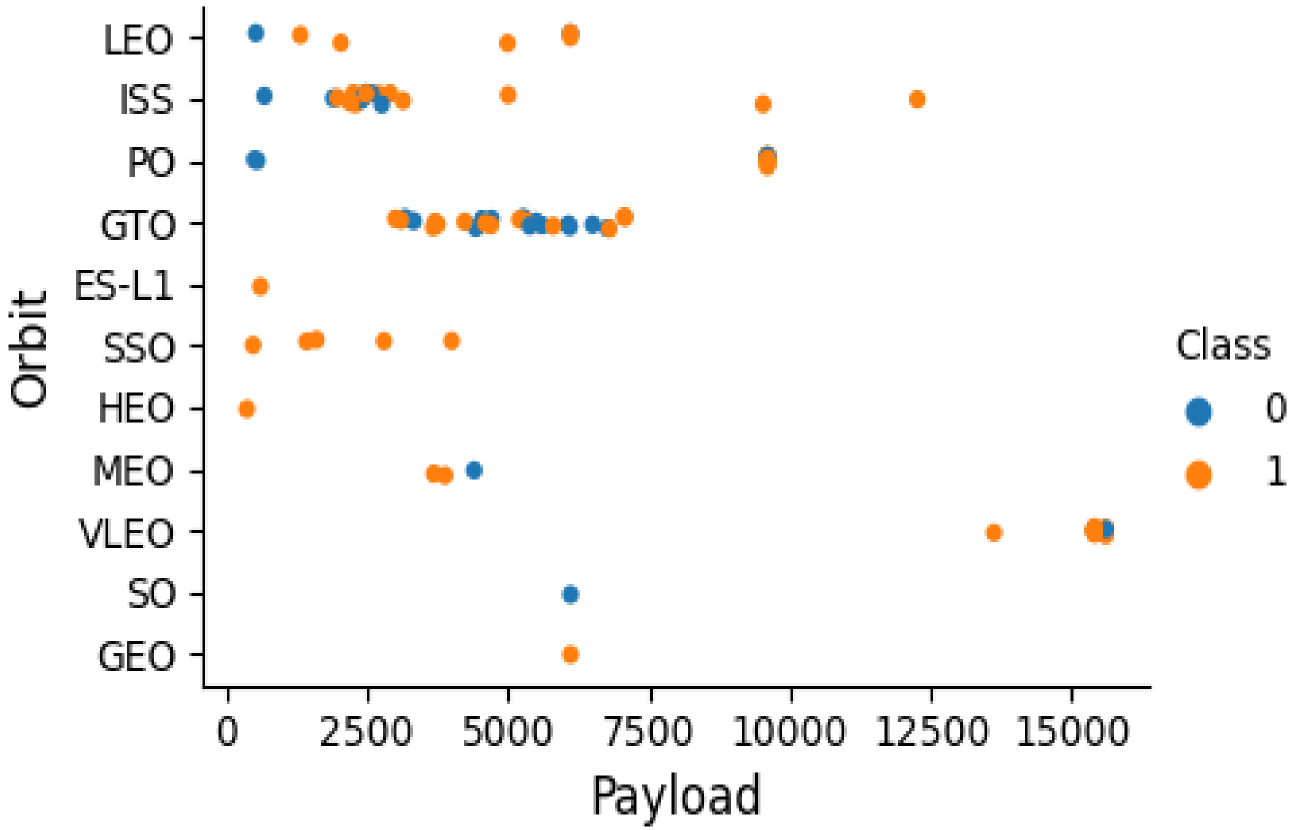
Flight Number vs Orbit

- In the LEO orbit, the initial flights are not successful but later success rate improves significantly in later flights.
- In GTO orbit, there seems to be no relationship between flight number.
- Majority of the launches in recent years are in VLEO orbit



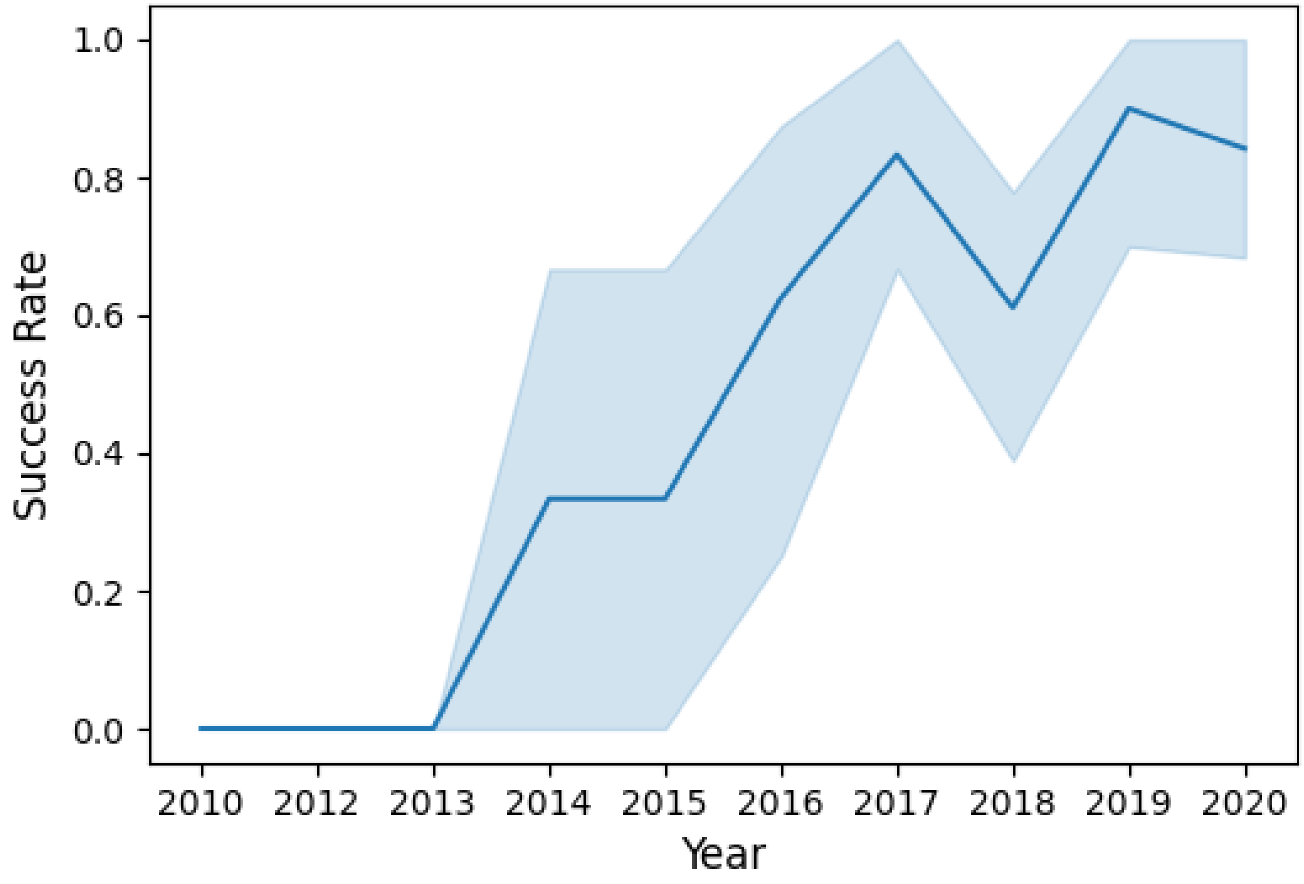
Payload Mass vs Orbit

- For Polar, LEO and ISS orbit, heavy payloads have higher success rate.
- In case of GTO orbit success rate are about 50%



Success Rate with time

- Success probabililty increases significantly with time since 2013 due to maybe technological advancement.



RESULTS

EXPLORATORY DATA ANALYSIS

Data exploration with SQL

Data Exploration using SQL

- Display the names of the unique launch sites in the space mission

	Launch_Site
0	CCAFS LC-40
1	VAFB SLC-4E
2	KSC LC-39A
3	CCAFS SLC-40

- Display 5 records where launch sites begin with the string 'CCA'

	Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing _Outcome
0	04-06-2010	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
1	08-12-2010	15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight C1, two CubeSats, barrel of...	0	LEO (ISS)	NASA (COTS) NRO	Success	Failure (parachute)
2	22-05-2012	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
3	08-10-2012	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
4	01-03-2013	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

Data Exploration using SQL

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

	SUM("PAYLOAD_MASS__KG_")
0	45596

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

	AVG("PAYLOAD_MASS__KG_")
0	2928.4

Data Exploration using SQL

- List the date when the first succesful landing outcome in ground pad was acheived.

	MIN(Date)
0	01-05-2017

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

	Booster_Version
0	F9 FT B1022
1	F9 FT B1026
2	F9 FT B1029.1
3	F9 FT B1021.2
4	F9 FT B1036.1

Data Exploration using SQL

- List the total number of successful and failure mission outcomes

	Count("Mission_Outcome")
0	100

- List the names of the booster_versions which have carried the maximum payload mass.

	Booster_Version
0	F9 B5 B1048.4
1	F9 B5 B1049.4
2	F9 B5 B1051.3
3	F9 B5 B1056.4
4	F9 B5 B1048.5

Data Exploration using SQL

- List the records which will display the month names, failure landing_outcomes in drone ship ,booster versions, launch_site for the months in year 2015

	Month_names	Landing _Outcome	Booster_Version	Launch_Site
0	01	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
1	04	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

- Rank the count of successful landing_outcomes between the date 04-06-2010 and 20-03-2017 in descending order.

	COUNT("Landing _Outcome")
0	34

RESULTS

INTERACTIVE DATA ANALYSIS

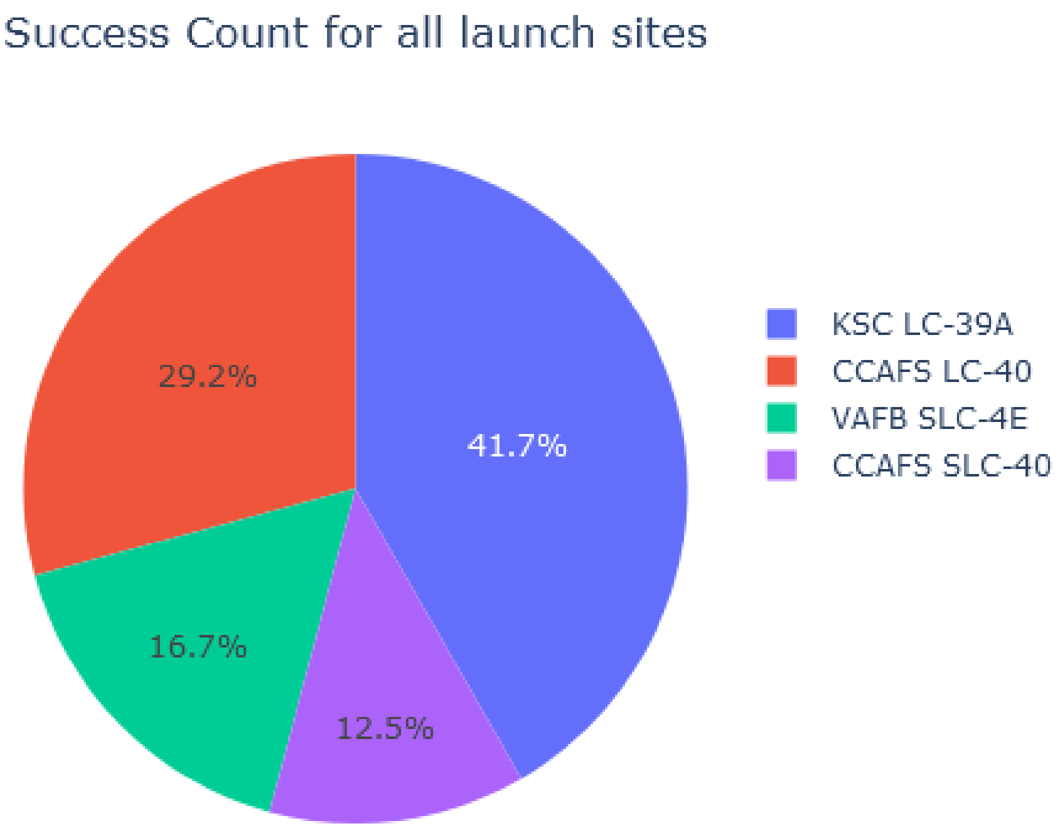
- Launch Site map using Folium
- Dashboard with Plotly Dash

Map with marked launch sites

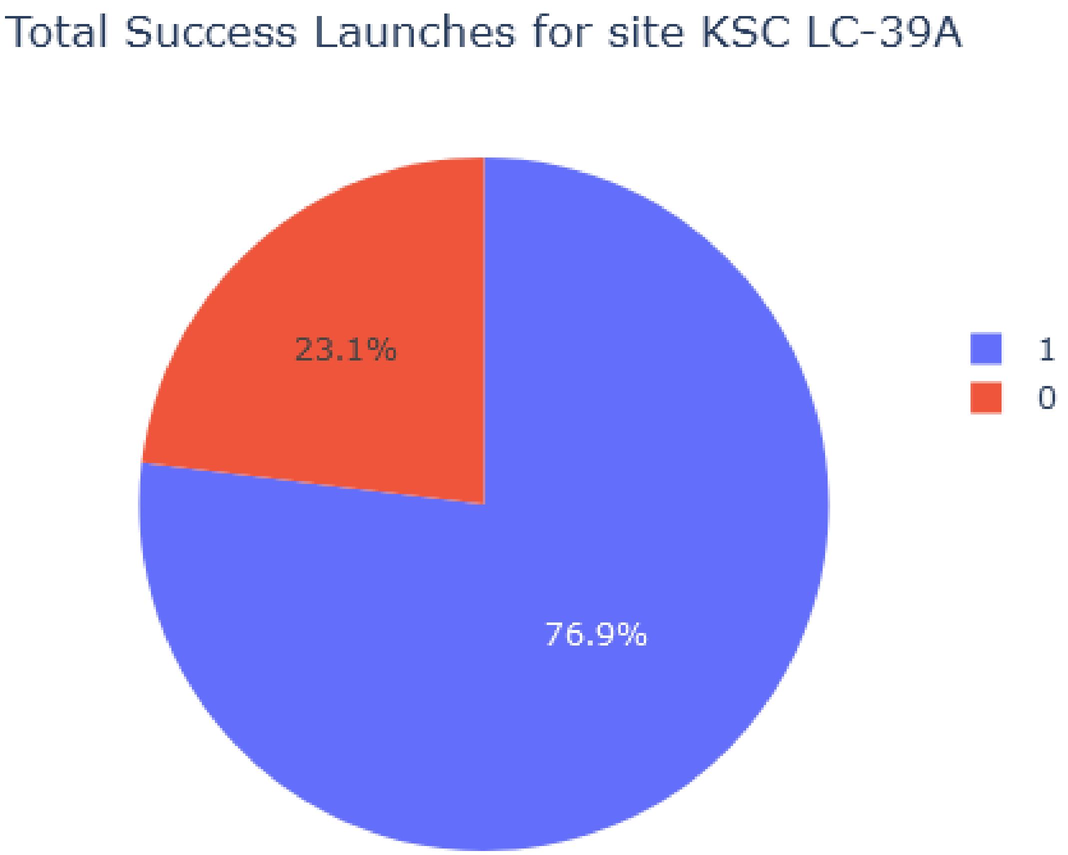
Success/ Faliure Launches

Dashboard: Success Rate for all launch sites

Payloads between 2000 and 5000 kg have the highest success rate.



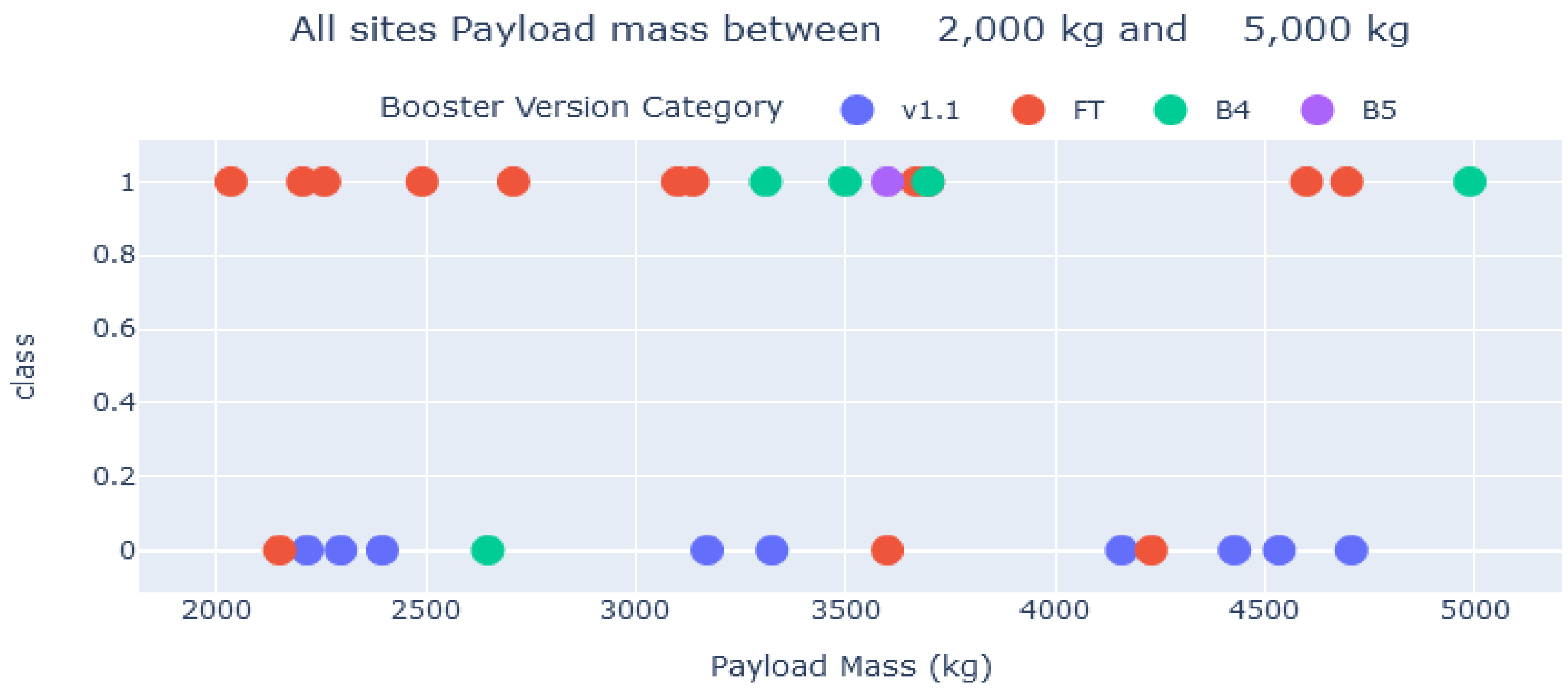
Dashboard: Which site has the highest launch success rate?



KSC LC 39A has the highest success rate of 76.9%

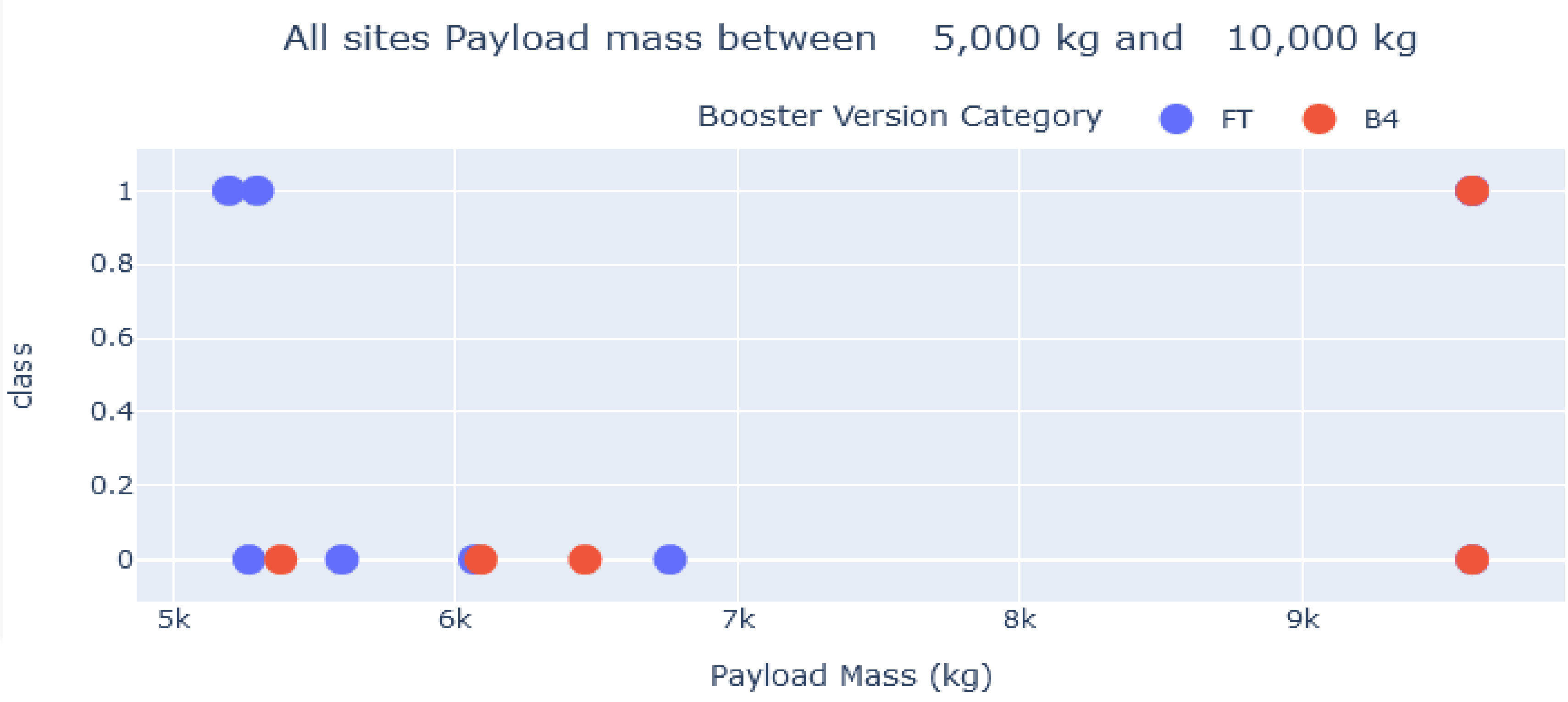
Dashboard: Which payload range(s) has the highest launch success rate?

Payloads between 2000 and 5000 kg have the highest success rate.



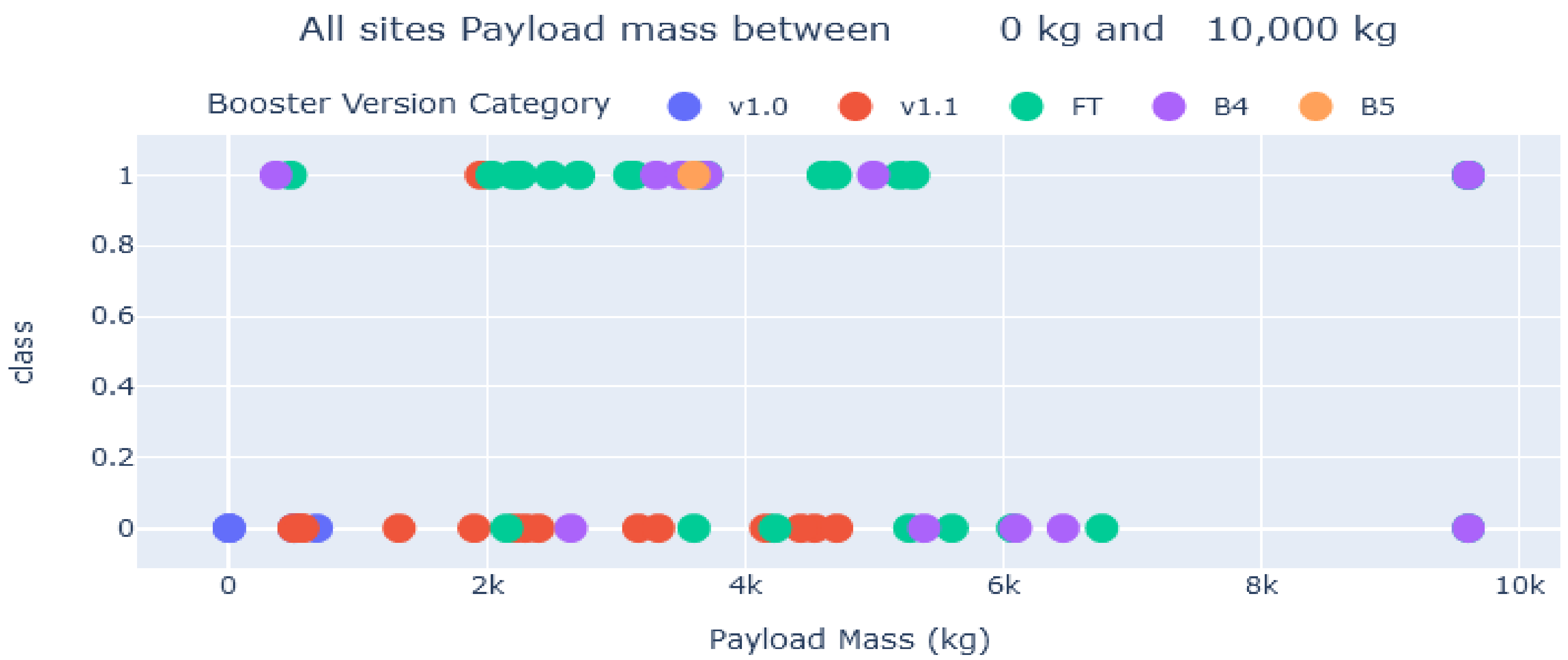
Dashboard: Which payload range(s) has the lowest launch success rate?

Payloads between 5000 and 10000 kg have the lowest success rate.



Dashboard: Which F9 Booster version (v1.0, v1.1, FT, B4, B5, etc.) has the highest launch success rate?

F9 Booster version FT has the highest success rate.



RESULTS

PREDICTIVE ANALYSIS

- Different machine learning algorithm performance to find the best performed method

Results: Find the method performs best

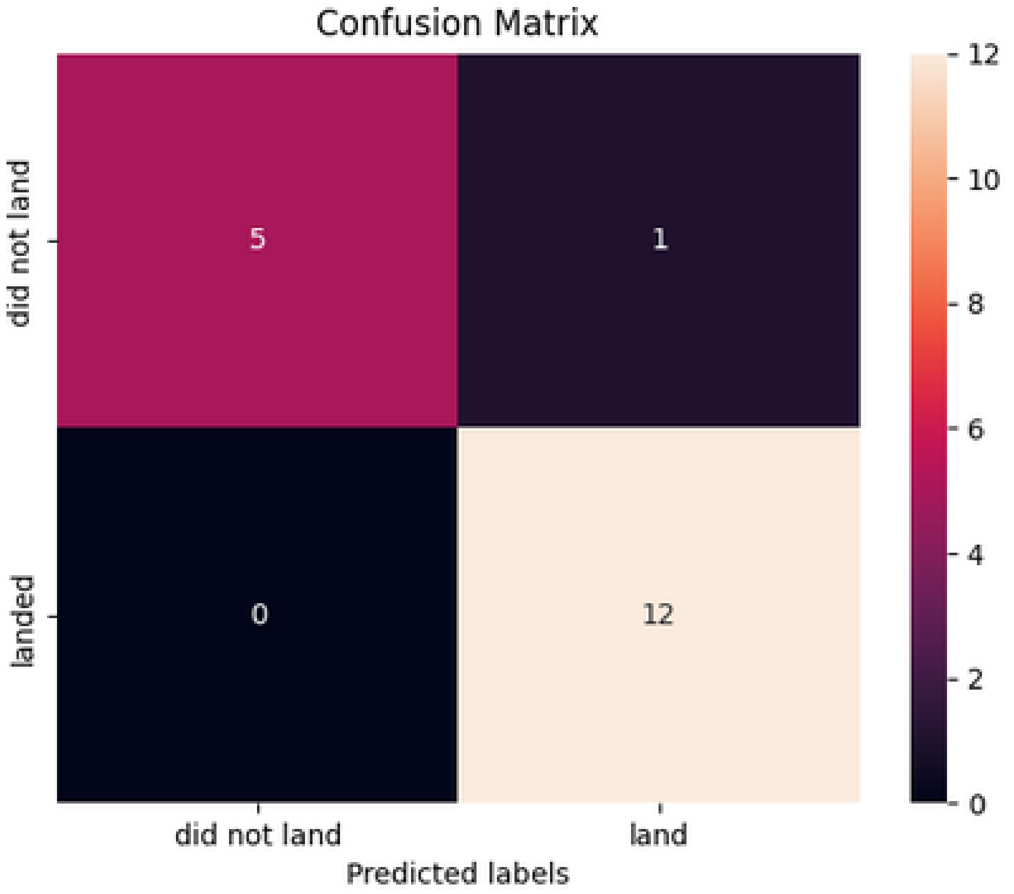
```
print("Logistic Regression: ",logreg_cv.score(X_test, Y_test))
print("SVM: ",svm_cv.score(X_test, Y_test))
print("Decision Tree: ", tree_cv.score(X_test, Y_test))
print("KNN: ", knn_cv.score(X_test, Y_test))
```

```
Logistic Regression:  0.9444444444444444
SVM:  0.8333333333333334
Decision Tree:  0.8888888888888888
KNN:  0.8333333333333334
```

Logistic Regression

```
1  logreg_cv.score(X_test, Y_test)
```

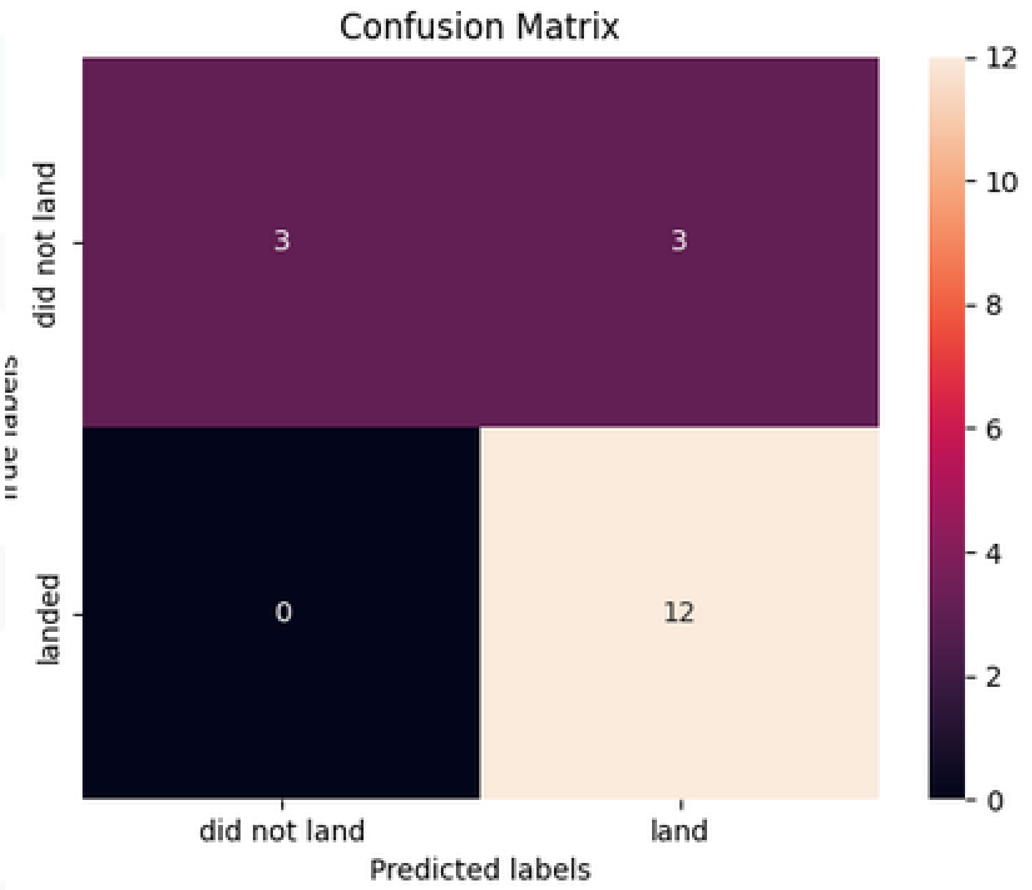
0.9444444444444444



Support Vector Machine

```
svm_cv.score(X_test, Y_test)
```

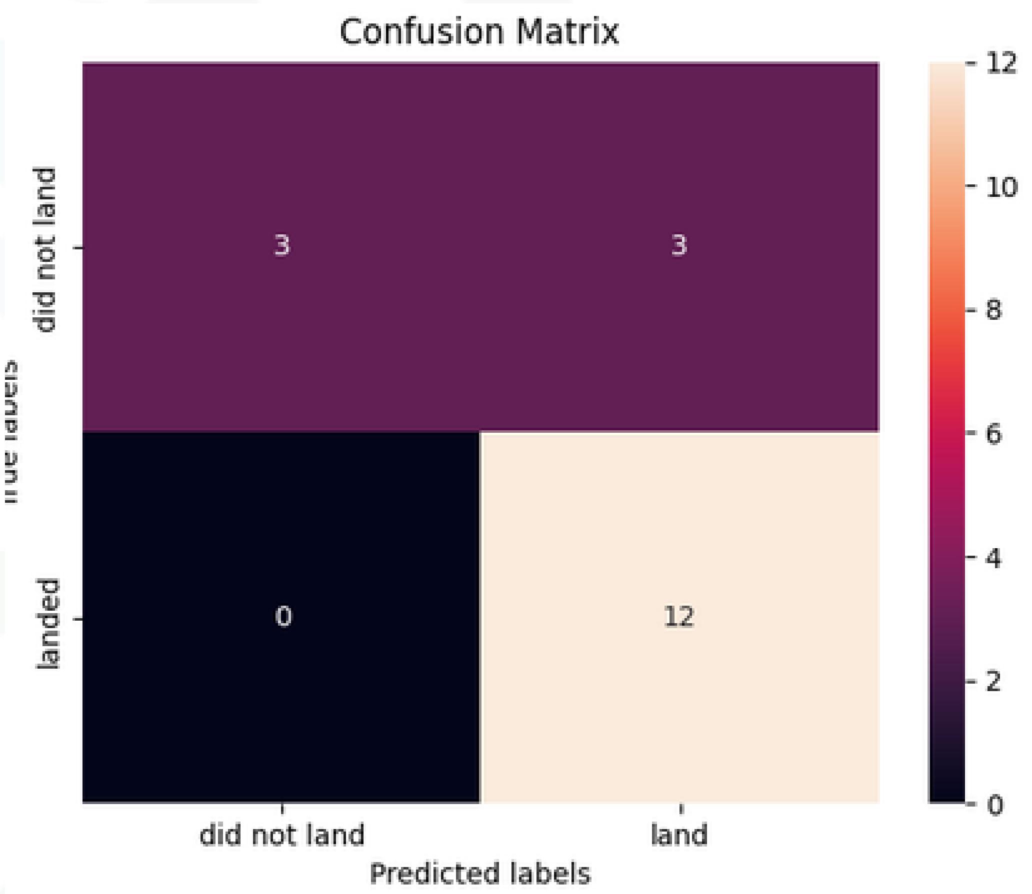
0.8333333333333334



Decision Tree Classifier

```
1  tree_cv.score(X_test, Y_test)
```

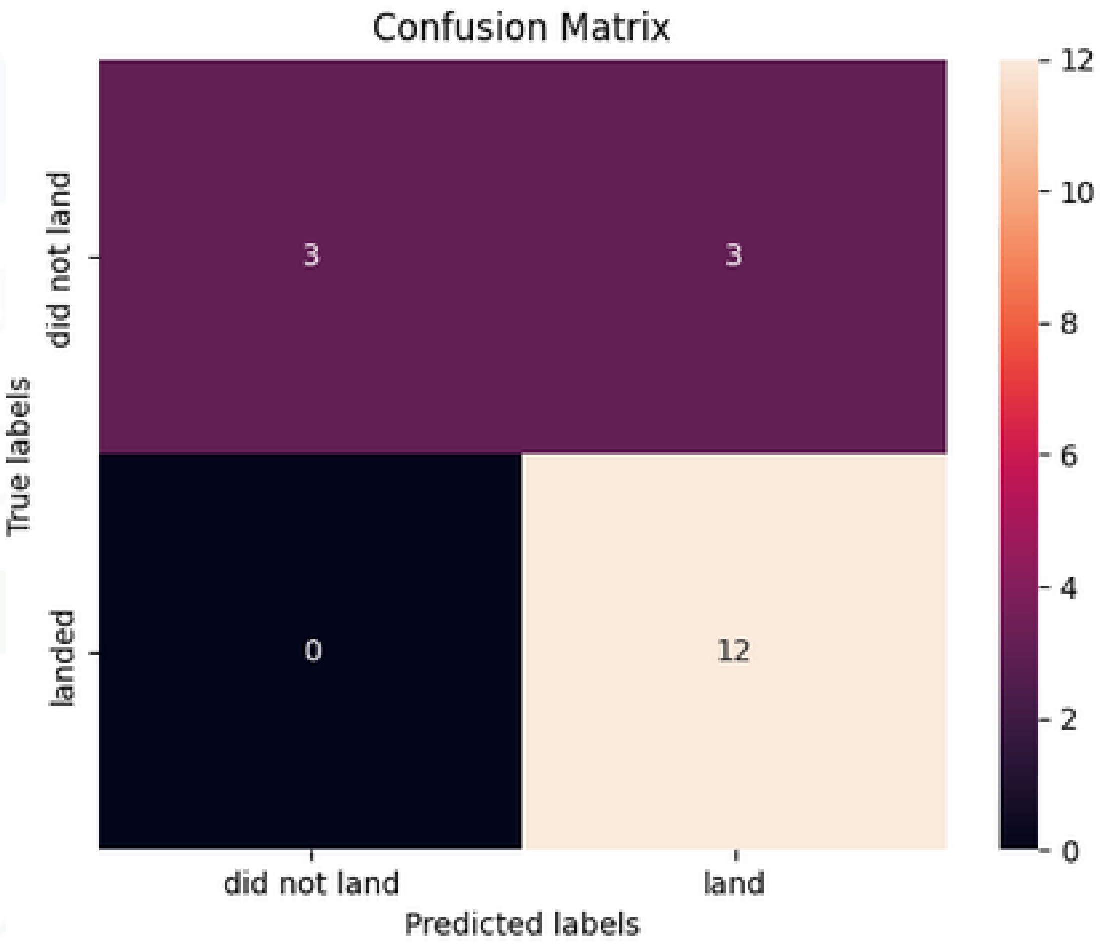
0.8888888888888888



K Nearest Neighbors

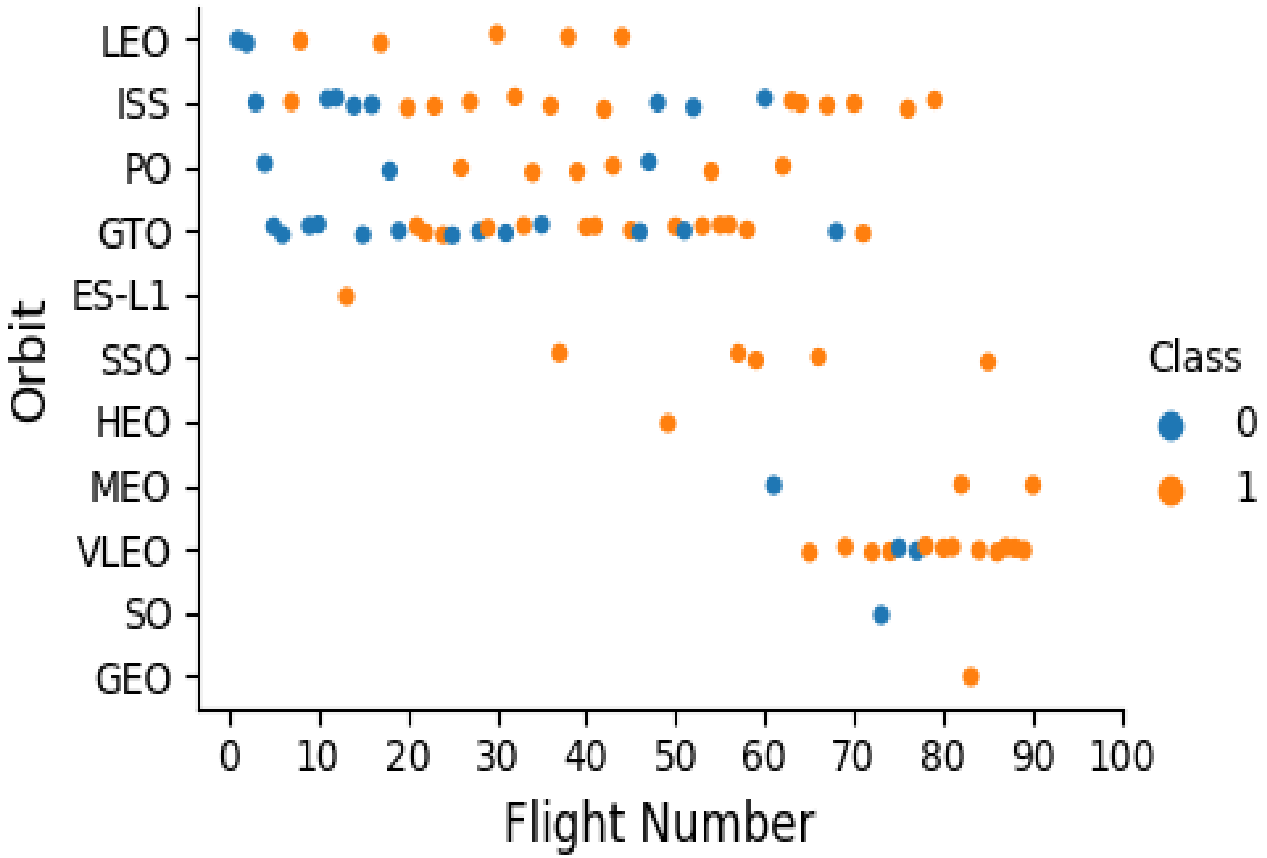
```
knn_cv.score(X_test, Y_test)
```

0.8333333333333334



DISCUSSION

- Earlier flight launches are mostly unsuccessful.
- Success of launching and landing improves with time and experiences.



Success Rate Of Different Launch Site

Launch Site	Success rate (%)	Failure rate (%)
KSC LC 39A	76.9	23.1
CCAFS LC 40	73.1	26.9
VAFB SLC 4E	60	40
CCAFS SLC 40	57.1	42.9

CONCLUSION

- Logistic regression performs best with a prediction accuracy of 94.44%
- Rocket with lighter payload have higher success rate than rocket with heavier payload
- The landing success probability increases with time.
- KSC LC 39A site has the highest launch success rate of 76%
- Orbit ES L1, SSO, HEO, and GEO has the best success rate of 100%