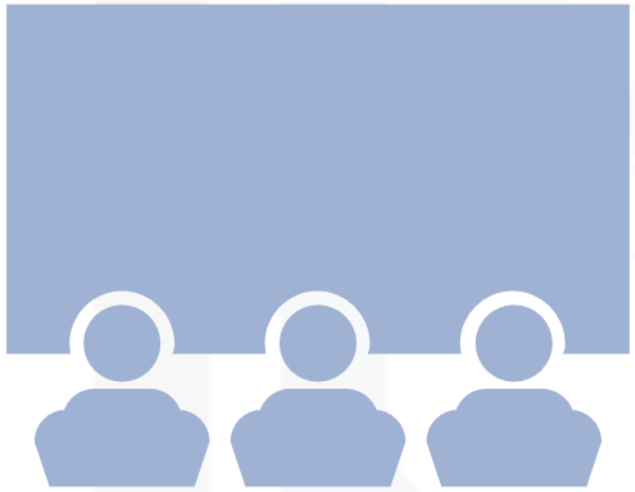
DATA SCIENCE CAPSTONE PROJECT

CRYSL LOBO

24th August 2021

OUTLINE



- Executive Summary
- Introduction
- Methodology
- Results
  - Visualization – Charts
  - Dashboard
- Discussion
  - Findings & Implications
- Conclusion
- Appendix

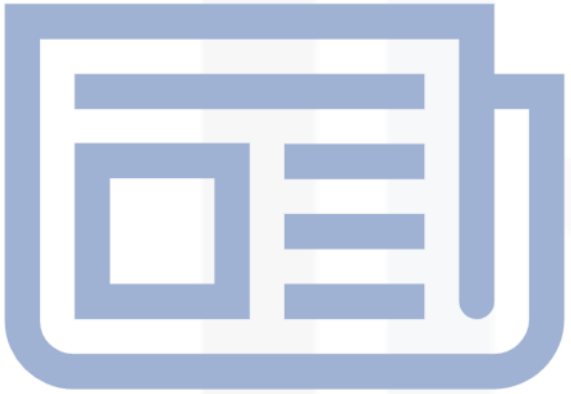
EXECUTIVE SUMMARY

- By using python, machine learning and data related to Space X rocket launches, we analysed the successful landings of the stage 1 boosters in order to predict future successful landings.
- Resuable stage 1 rockets represent significant cost savings as compared to traditional single use rockets.
- The analyzed data sets included data related to Booster Version, Payload Mass, Orbit, Landing Locations, Landing Outcomes and Flight Numbers.

INTRODUCTION

- The cost of sending payloads into space is high for many projects. Single use booster platforms carry significant overhead(estimated 165 million per launch) with a high cost per kg of weight.
- Reusable rockets represent a potential cost reduction with an average cost of 65 million per launch.
- Our aim is to understand what variables are responsible in have a successful landing of launched rockets.

METHODOLOGY



- Data collection methodology:- Using BeautifulSoup, data regarding SpaceX Falcon 9 launch records from 2010 to 2020 were obtained from Wikipedia by parsing the HTML flight table.
- Performed data wrangling:- Data from the HTML table was loaded into a pandas dataframe and each column frequencies were analyzed. Missing numbers were imputed as means of their respective columns.
- Performed exploratory data analysis (EDA) using visualization and SQL:- A combination of SQL queries and plot visualizations via Matplotlib, Plotly, Folium, and DASH integrations were used to visualize different aspects of the data.
- Performed predictive analysis using classification models:- Utilizing scikit-learn, the data was split into training and test data after variables were dummy coded. Various models were to predict success (Decision Tree, Logistic Regression, KNN, SVC).

Data Collection

- Data from the SpaceX API was requested and obtained via a JSON file. The data was appended to a table and filtered for Falcon 9 launches.
- Utilizing BeautifulSoup, we obtained the publicly available data regarding SpaceX Falcon 9 launches via Wikipedia.

Data Collection – Web Scraping

- The collected data was filtered to obtain the appropriate table and extracted and appended to separate lists. It was then concatenated into pandas dataframe.

Data Wrangling

- Each column was analyzed for missing values and were replaced with column means. A binary outcome variable was created by concatenating different outcomes (good vs bad).

EDA with Data Visualization

- Scatter plots were used to stratify data to observe relationships between continuous/desired variable.
- Histogram was used to see average success rates
- Line Graph was used to observe success rate overtime
- Pie graph was used to analyze distribution of outcomes

EDA with SQL

- Identifies different launch sites
- Reviewed different payload stratified by booster types
- Reviewed dates of successful launches
- Identifies boosters associated with drone ship launches

Interactive map with Folium

- Identifies all the launch sites on the map
- Marked the launch sites with different colors based on success and failure of the landing outcomes
- Demonstrated distance to important surrounding logistic features like railroads, coasts and highways.

Dashboard with Plotly Dash

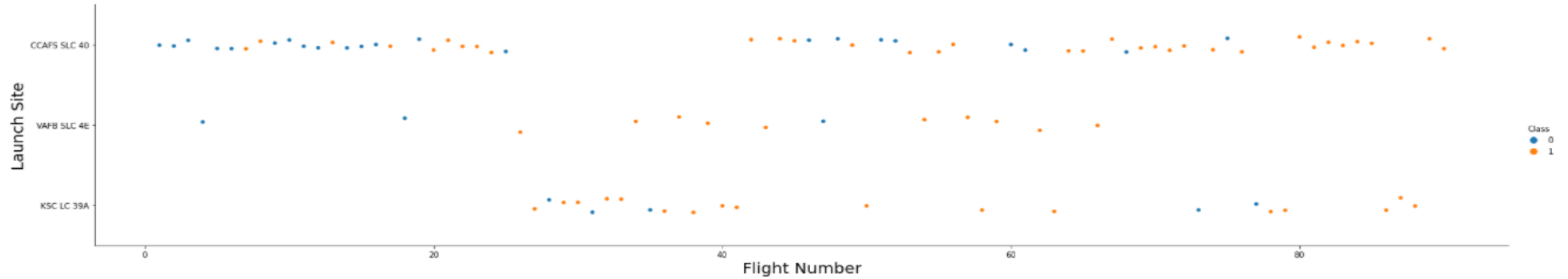
- Pie graph was used to demonstrate launch site success/failures i.e. to see if one site is more associated with successful landing
- Scatter plots were used to analyze the relationship between payload and outcome of each launch.

Predictive analysis

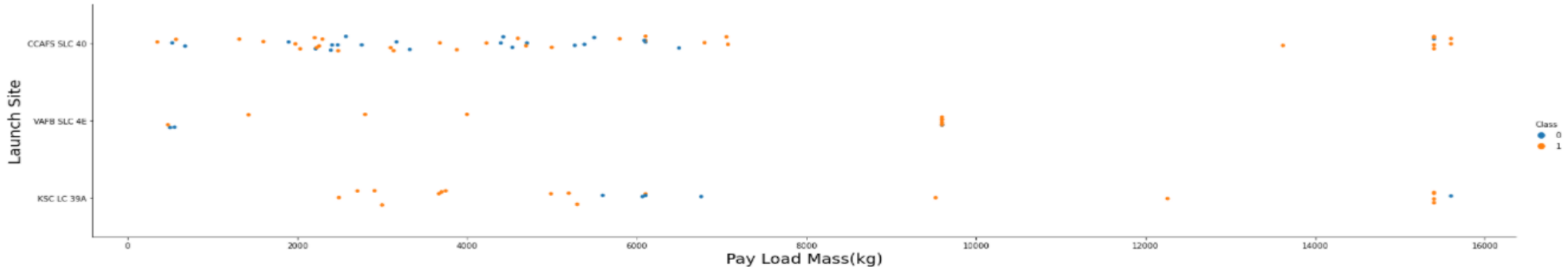
- The data was split into two parts. One part was used for training the data and the other was used for testing the data on the algorithm.
- We used Gridsearch to identify key hyperparameters and best predictive scores using logistic regression, SVM, Decision Tree and KNN.
- Each regression was fit using the training data and then the model was tested against a separate test data set.
- Confusion matrix was used to demonstrate prediction accuracy

Results of EDA with Visualization

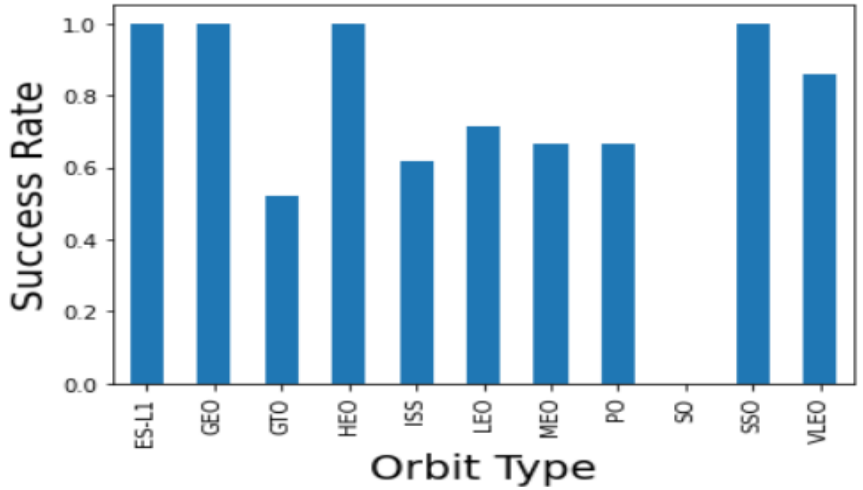
FLIGHT NUMBER AND LAUNCH SITE



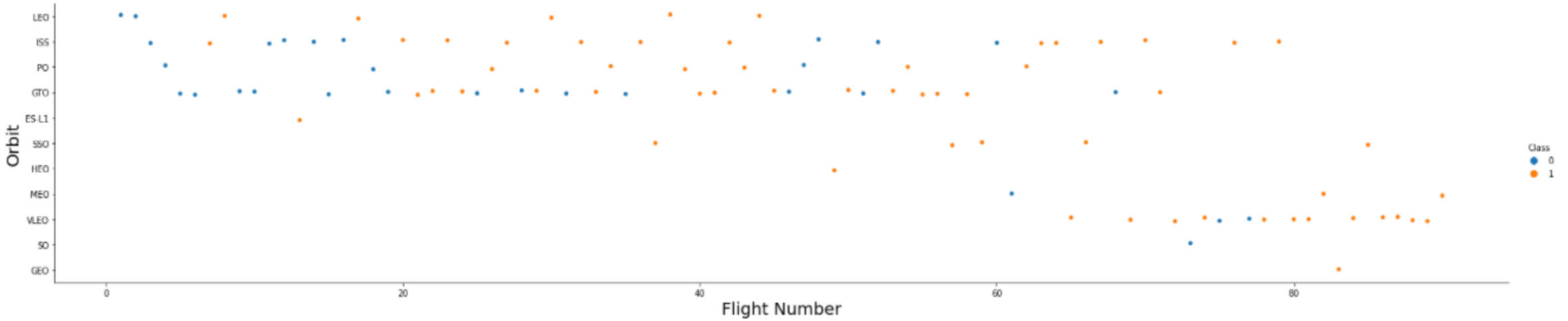
PAYLOAD AND LAUNCH SITE



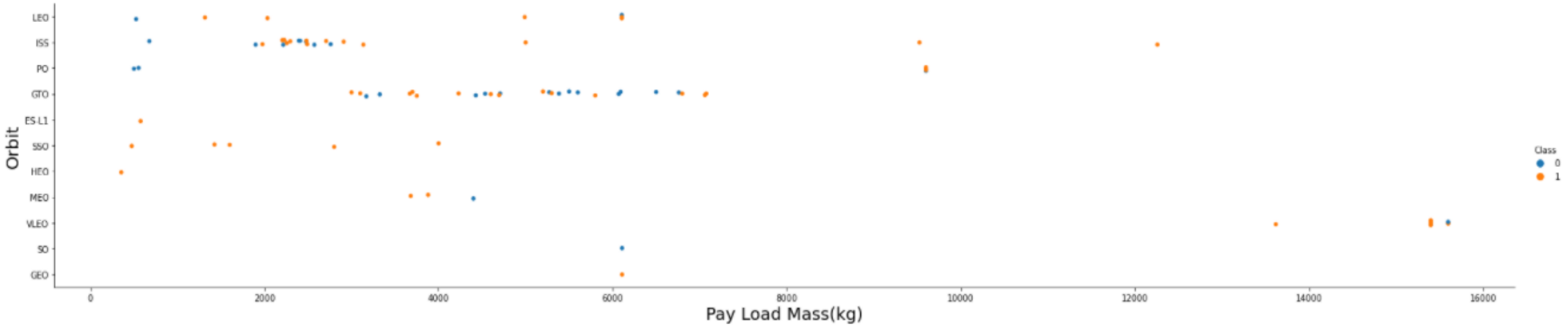
SUCCESS RATE AND ORBIT TYPE



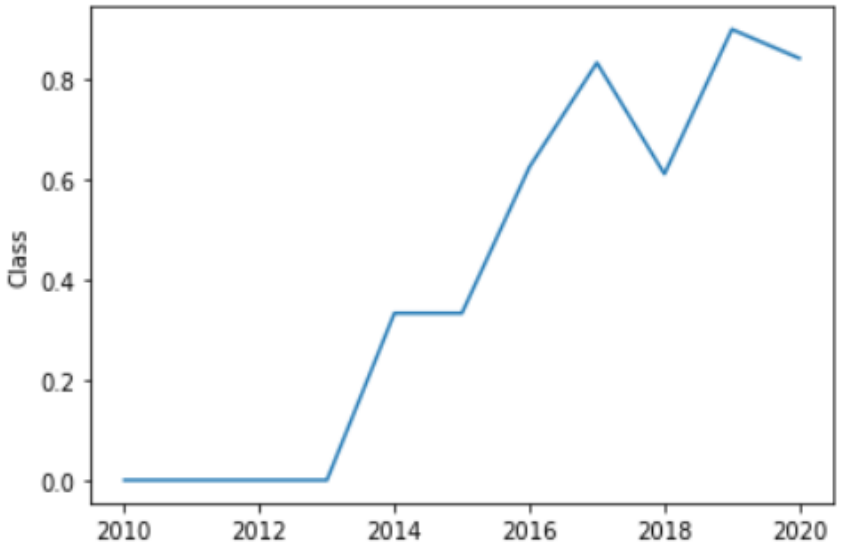
FLIGHT NUMBER AND ORBIT TYPE



PAYLOAD AND ORBIT TYPE



LAUNCH SUCCESS YEARLY TREND



Results of EDA with SQL

Task 1

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

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

```
 * ibm_db_sa://xxz12385:***@dashdb-txn-sbox-yp-dal09-12.services.dal.bluemix.net:50000/BLUDB
Done.
```

Out[4]:

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

Task 2

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

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

```
 * ibm_db_sa://xxz12385:***@dashdb-txn-sbox-yp-dal09-12.services.dal.bluemix.net:50000/BLUDB
Done.
```

Out[5]:

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

Task 3

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

In [6]:
```
%sql select sum(PAYLOAD_MASS__KG_) from SPACEXDATASET where CUSTOMER = 'NASA (CRS)'
```

```
 * ibm_db_sa://xxz12385:***@dashdb-txn-sbox-yp-dal09-12.services.dal.bluemix.net:50000/BLUDB
Done.
```

Out[6]:

1
45596

Task 4

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

In [7]:
```
%sql select avg(PAYLOAD_MASS__KG_) from SPACEXDATASET where BOOSTER_VERSION = 'F9 v1.1'
```

```
 * ibm_db_sa://xxz12385:***@dashdb-txn-sbox-yp-dal09-12.services.dal.bluemix.net:50000/BLUDB
Done.
```

Out[7]:

1
2928.400000

Task 5

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

Hint:Use min function

In [8]:
```
%sql select min(DATE) from SPACEXDATASET where LANDING__OUTCOME = 'Success (ground pad)'
```

```
 * ibm_db_sa://xxz12385:***@dashdb-txn-sbox-yp-dal09-12.services.dal.bluemix.net:50000/BLUDB
Done.
```

Out[8]:

1
2015-12-22

Task 6

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

In [9]:
```
%sql select BOOSTER_VERSION from SPACEXDATASET where LANDING__OUTCOME = 'Success (drone ship)' AND PAYLOAD_MASS__KG_ BETWEEN 4000 AND 600
0
```

```
 * ibm_db_sa://xxz12385:***@dashdb-txn-sbox-yp-dal09-12.services.dal.bluemix.net:50000/BLUDB
Done.
```

Out[9]:

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

Task 7

List the total number of successful and failure mission outcomes

In [10]:
```
%sql select count(MISSION_OUTCOME) from SPACEXDATASET
```

```
 * ibm_db_sa://xxz12385:***@dashdb-txn-sbox-yp-dal09-12.services.dal.bluemix.net:50000/BLUDB
Done.
```

Out[10]:

1
101

Task 8

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

In [11]:
```
%sql select BOOSTER_VERSION from SPACEXDATASET where PAYLOAD_MASS__KG_ in (select max(PAYLOAD_MASS__KG_) from SPACEXDATASET)
```

```
 * ibm_db_sa://xxz12385:***@dashdb-txn-sbox-yp-dal09-12.services.dal.bluemix.net:50000/BLUDB
Done.
```

Out[11]:

booster_version
F9 B5 B1048.4
F9 B5 B1049.4
F9 B5 B1051.3

Task 9

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

In [12]:
```
%sql select LANDING__OUTCOME, BOOSTER_VERSION, LAUNCH_SITE from SPACEXDATASET where LANDING__OUTCOME = 'Failure (drone ship)' and YEAR(DATE) = '2015'
```

```
 * ibm_db_sa://xxz12385:***@dashdb-txn-sbox-yp-dal09-12.services.dal.bluemix.net:50000/BLUDB
Done.
```

Out[12]:

landing__outcome	booster_version	launch_site
Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

Task 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

In [18]:
```
%sql select LANDING__OUTCOME, count(LANDING__OUTCOME) as LANDING_OUTCOME_COUNT from SPACEXDATASET where DATE between '2010-06-04' and '2017-03-20'GROUP BY LANDING__OUTCOME \
ORDER BY count(LANDING__OUTCOME) desc
```

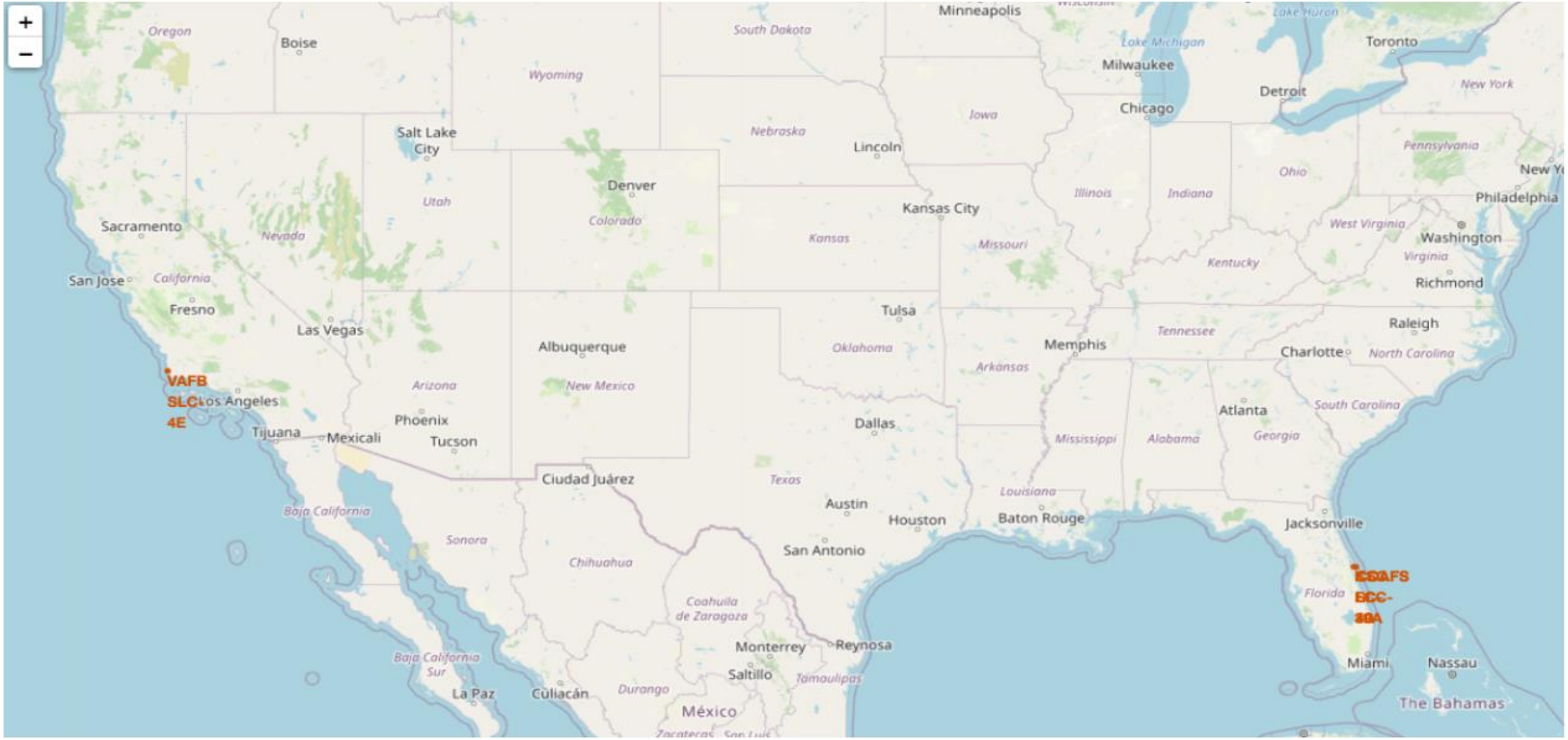
```
 * ibm_db_sa://xxz12385:***@dashdb-txn-sbox-yp-dal09-12.services.dal.bluemix.net:50000/BLUDB
Done.
```

Out[18]:

landing__outcome	landing_outcome_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 of Folium

LAUNCH SITE DISTRIBUTION

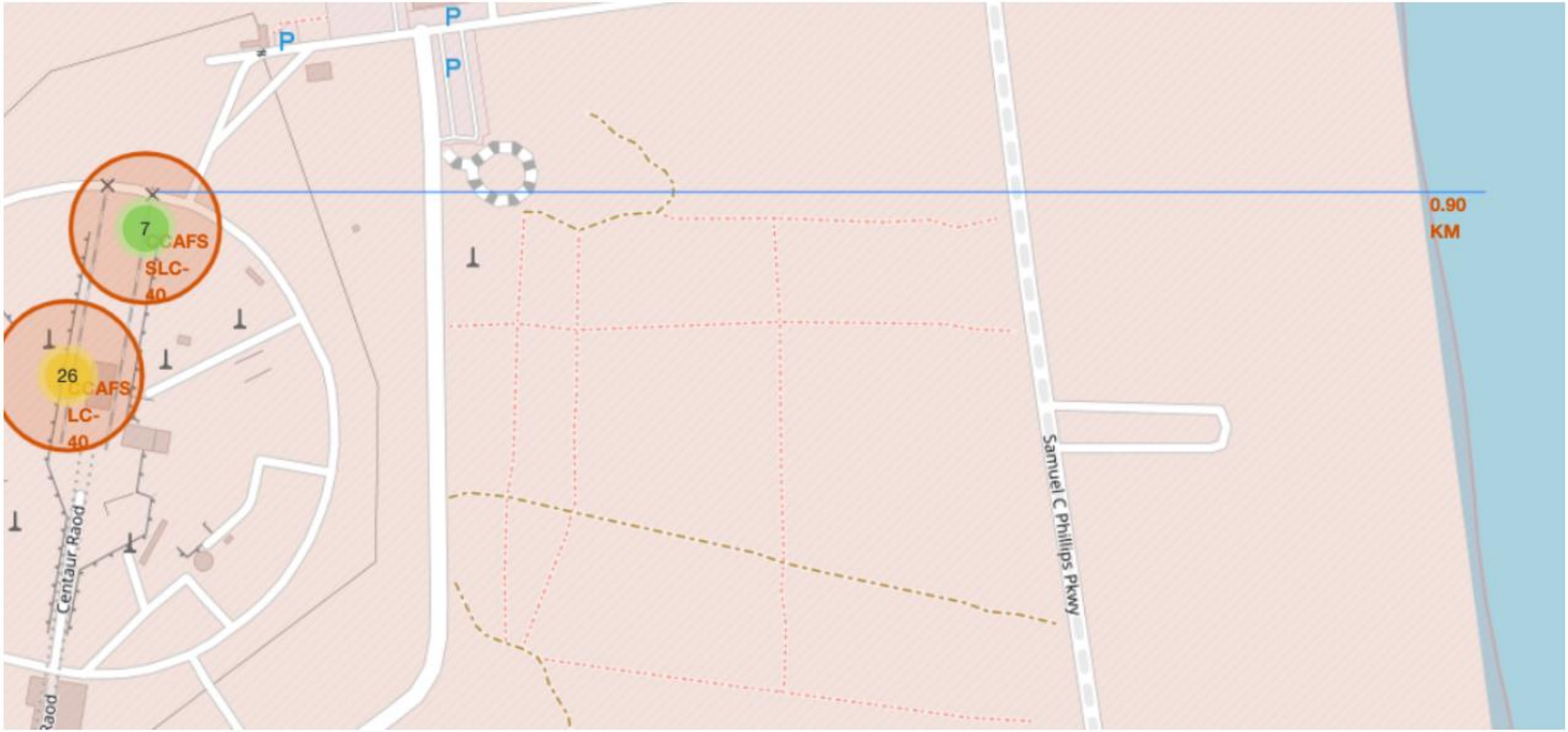


IBM Developer

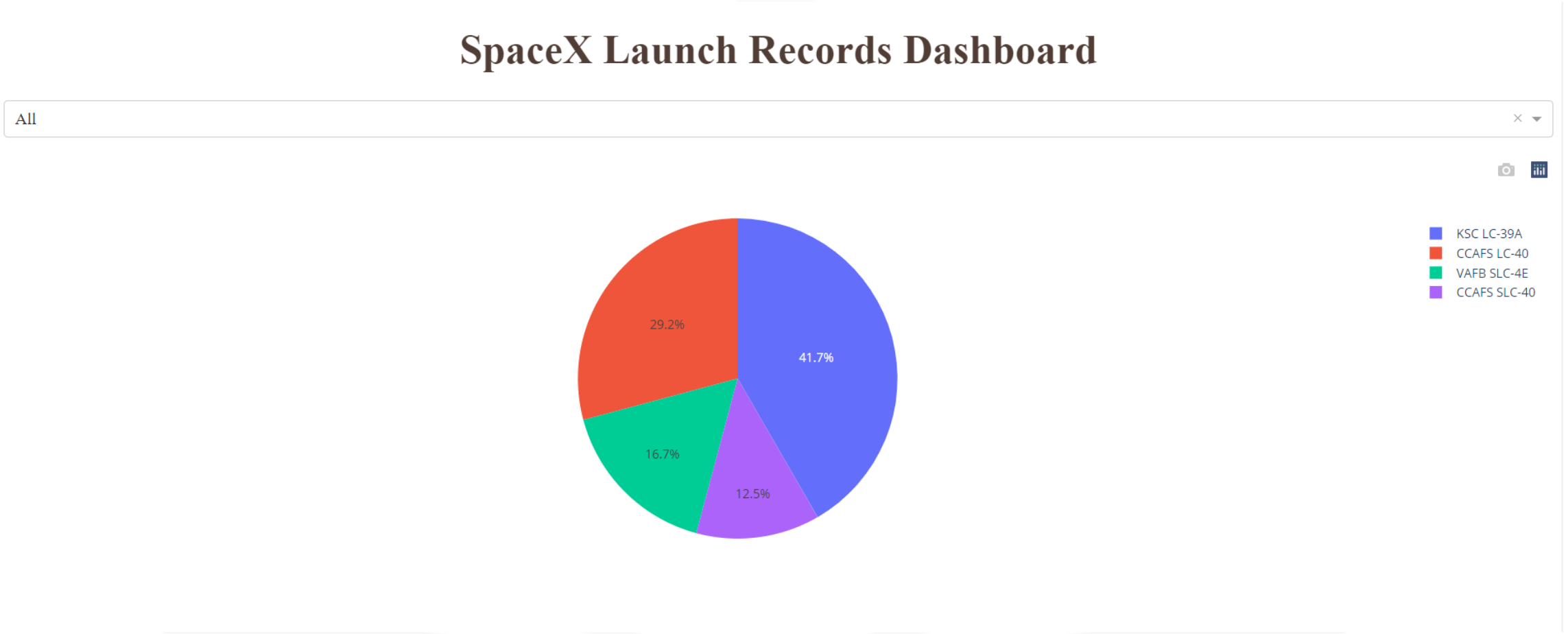
SKILLS NETWORK

SUCCESSFUL LAUNCHES BY SITE

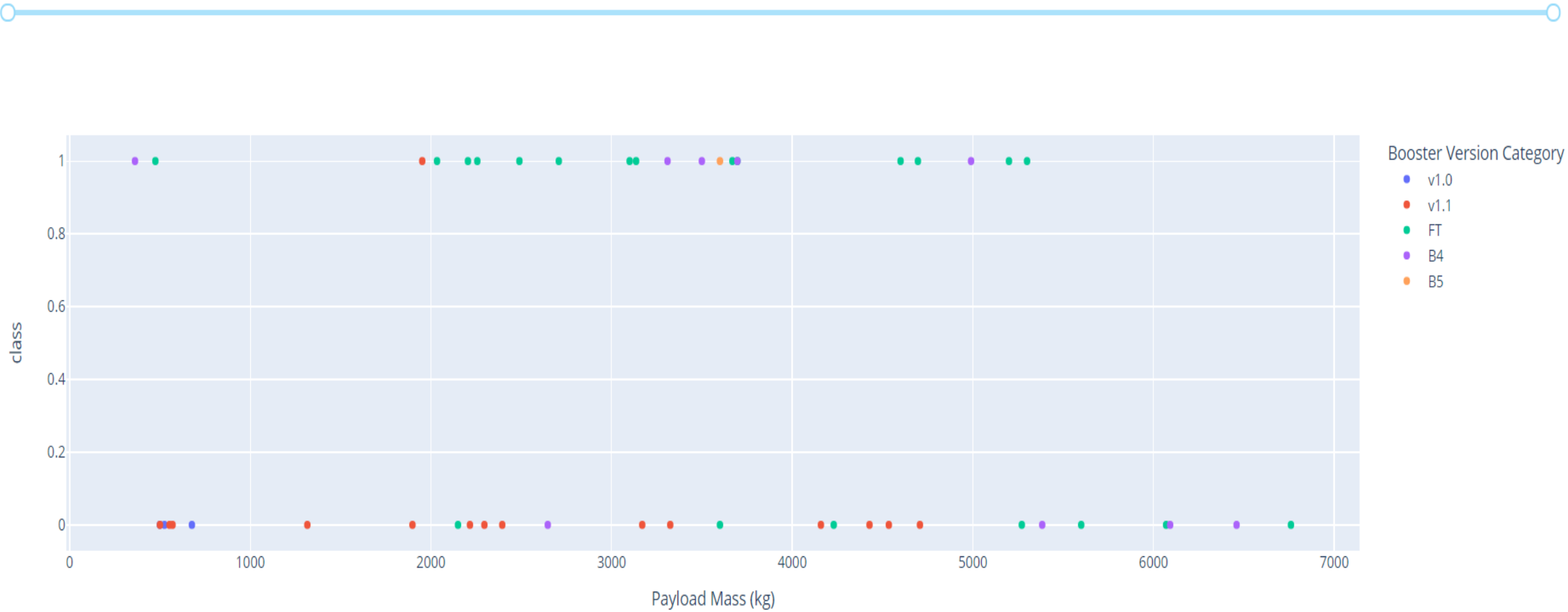
DISTANCE BETWEEN LAUNCH SITE AND LOGISTIC FEATURES



Results of Plotly Dash



Payload range (Kg):



Results of Predictive Analysis

TASK 5

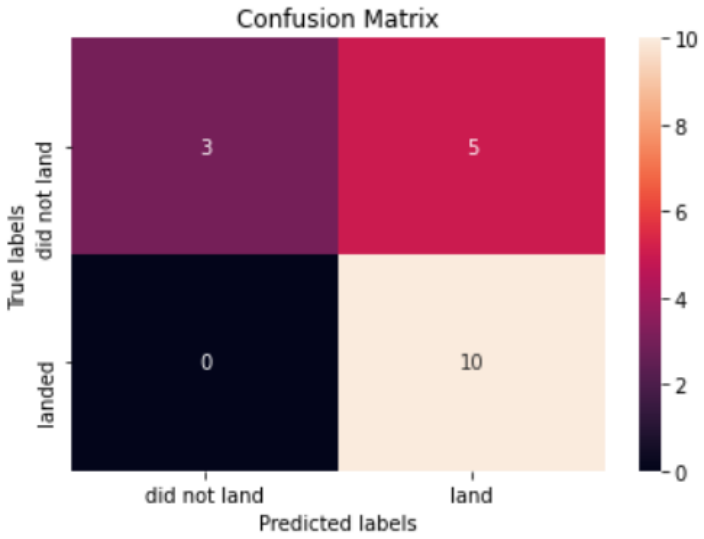
Calculate the accuracy on the test data using the method `score`:

```
In [12]: logreg_cv.score(X_test,Y_test)
```

```
Out[12]: 0.7222222222222222
```

Lets look at the confusion matrix:

```
In [13]: yhat=logreg_cv.predict(X_test)
         plot_confusion_matrix(Y_test,yhat)
```



TASK 7

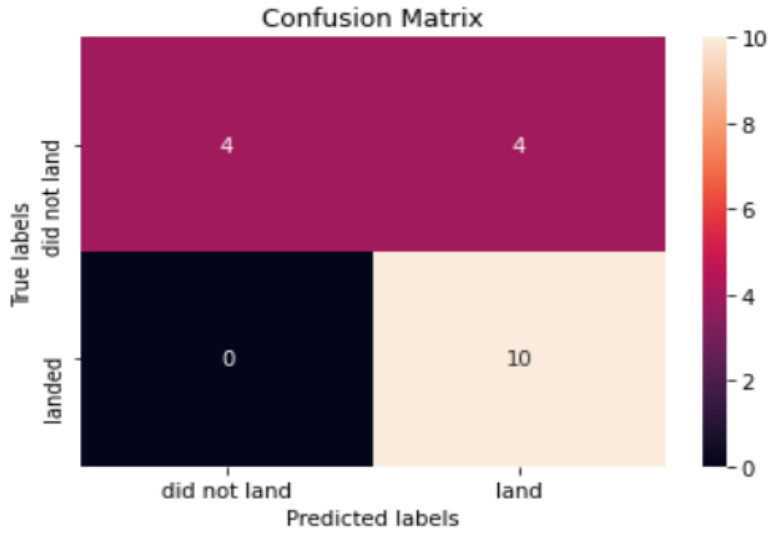
Calculate the accuracy on the test data using the method `score`:

```
In [17]: svm_cv.score(X_test,Y_test)
```

```
Out[17]: 0.7777777777777778
```

We can plot the confusion matrix

```
In [18]: yhat=svm_cv.predict(X_test)
         plot_confusion_matrix(Y_test,yhat)
```



TASK 9

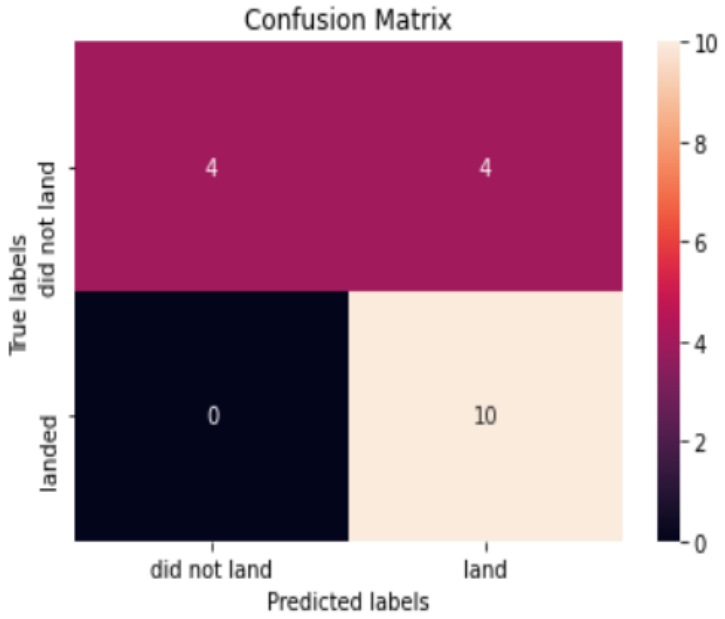
Calculate the accuracy of tree_cv on the test data using the method `score`:

In [22]:
```
print("test score:",tree_cv.score(X_test,Y_test))
```

```
test score: 0.7222222222222222
```

We can plot the confusion matrix

In [23]:
```
yhat = svm_cv.predict(X_test)
plot_confusion_matrix(Y_test,yhat)
```



TASK 11

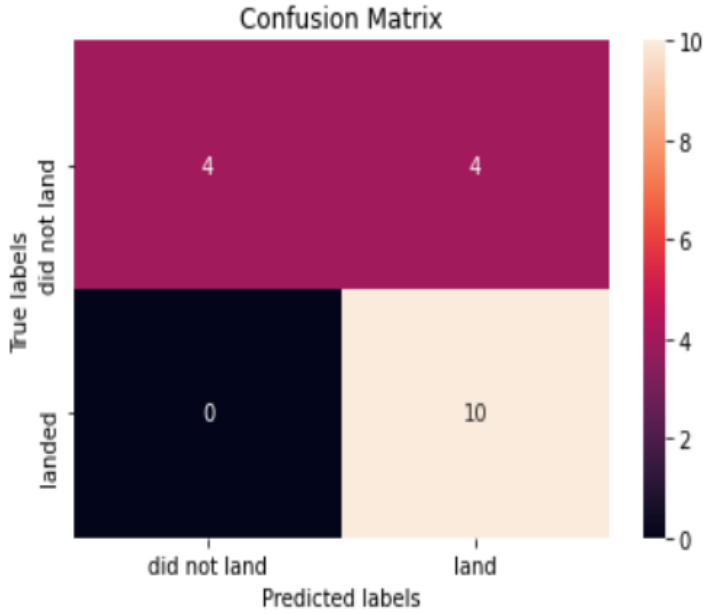
Calculate the accuracy of tree_cv on the test data using the method `score`:

In [27]:
```
knn_cv.score(X_test,Y_test)
```

Out[27]:
```
0.7777777777777778
```

We can plot the confusion matrix

In [28]:
```
yhat = knn_cv.predict(X_test)
plot_confusion_matrix(Y_test,yhat)
```



CONCLUSION

- Important factore to predict success include: Launch Number, Desired Orbit, Booster Version and Payload Mass.
- Overall east coast launches are more successful compared to west coast.
- Given the high success rate, SpaceX Falcon 9 booster represents a reliable and cheaper alternative to single use rockets.