IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

<Muhammad Hariss Amin>
<11/08/2023>

Outline

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

Executive Summary

Summary of Methodologies

The research attempts to identify the factors for a successful rocket landing. To make this determination, the following methodologies where used:

• Collect data using SpaceX REST API and web scraping techniques

• Wrangle data to create success/fail outcome variable

• Explore data with data visualization techniques, considering the following factors: payload, launch site, flight number and yearly trend

• Analyze the data with SQL, calculating the following statistics: total payload, payload range for successful launches, and total # of successful and failed outcomes

• Explore launch site success rates and proximity to geographical markers

• Visualize the launch sites with the most success and successful payload ranges

• Build Models to predict landing outcomes using logistic regression, support vector machine (SVM), decision tree and K -nearest neighbor (KNN)

Results

Exploratory Data Analysis:

• Launch success has improved over time

• KSC LC-39A has the highest success rate among landing sites

• Orbits ES -L1, GEO, HEO, and SSO have a 100% success

Visualization/Analytics:

• Most launch sites are near the equator, and all are close to the coast

Predictive Analytics:

• All models performed similarly on the test set. The decision tree model slightly outperformed

Introduction

Background

SpaceX has pioneered reusable rocket technology that lowers launch costs, making space more accessible. Their Falcon 9 rockets cost $62 million per flight since the first stage can land and fly again, compared to $165 million for expendable rockets. By leveraging public launch data and machine learning models, we can gain insights into reusability by predicting whether Falcon 9's first stage will successfully land. Our goal is to illuminate the factors enabling SpaceX's innovative reusable rockets, which provide affordable access to space through breakthrough economics. By supporting the analysis of reusable rocket viability, our models aim to benefit the entire space community. In summary, SpaceX's innovations are opening up space exploration, and our models help uncover the mechanics of their game-changing technology.

Explore

How payload mass, launch site, number of flights, and orbits affect first-stage landing success • Rate of successful landings over time • Best predictive model for successful landing (binary classification)

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Data using SpaceX REST API and web scraping techniques
- Perform data wrangling
  - Data – by filtering the data, handling missing values and applying one hot encoding – to prepare the data for analysis and modeling
- 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 models to predict landing outcomes using classification models. Tune and evaluate models to find best model and parameters

Data Collection - API

- **Steps**
- • Request data from SpaceX API (rocket launch data)
- • Decode response using .json() and convert to a dataframe using .json_normalize()
- • Request information about the launches from SpaceX API using custom functions
- • Create dictionary from the data
- • Create dataframe from the dictionary
- • Filter dataframe to contain only Falcon 9 launches
- • Replace missing values of Payload Mass with calculated .mean()
- • Export data to csv file

```python
def getCoreData(data):
    for core in data['cores']:
            if core['core'] != None:
                response = requests.get("https://api.spacexdata.com/v4/cores/"+core['core']).json()
                Block.append(response['block'])
                ReusedCount.append(response['reuse_count'])
                Serial.append(response['serial'])
            else:
                Block.append(None)
                ReusedCount.append(None)
                Serial.append(None)
            Outcome.append(str(core['landing_success'])+' '+str(core['landing_type']))
            Flights.append(core['flight'])
            GridFins.append(core['gridfins'])
            Reused.append(core['reused'])
            Legs.append(core['legs'])
            LandingPad.append(core['landpad'])
```

Data Collection – Web Scraping

Steps

- Request data (Falcon 9 launch data) from Wikipedia
- Create BeautifulSoup object from HTML response
- Extract column names from HTML table header
- Collect data from parsing HTML tables
- Create dictionary from the data
- Create dataframe from the dictionary • Export data to csv file

Data Wrangling

Steps

- Perform EDA and determine data labels
- Calculate:
  - # of launches for each site
  - # and occurrence of orbit
  - # and occurrence of mission outcome per orbit type]
- Create binary landing outcome column (dependent variable)
- Export data to csv file

Landing Outcome

- Landing was not always successful
- True Ocean: mission outcome had a successful landing to a specific region of the ocean
- False Ocean: represented an unsuccessful landing to a specific region of ocean
- True RTLS: meant the mission had a successful landing on a ground pad
- False RTLS: represented an unsuccessful landing on a ground pad
- True ASDS: meant the mission outcome had a successful landing on a drone ship
- False ASDS: represented an unsuccessful landing on drone ship
- Outcomes converted into 1 for a successful landing and 0 for an unsuccessful landing

EDA with Data Visualization

Charts

- Flight Number vs. Payload
- Flight Number vs. Launch Site
- Payload Mass (kg) vs. Launch Site
- Payload Mass (kg) vs. Orbit type

Analysis

- View relationship by using scatter plots. The variables could be useful for machine learning if a relationship exists
- Show comparisons among discrete categories with bar charts. Bar charts show the relationships among the categories and a measured value.

EDA with SQL

Queries

Display:

- Names of unique launch sites
- 5 records where launch site begins with 'CCA'
- Total payload mass carried by boosters launched by NASA (CRS)
- Average payload mass carried by booster version F9 v1.1.

List:

- Date of first successful landing on ground pad
- Names of boosters which had success landing on drone ship and have payload mass greater than 4,000 but less than 6,000
- Total number of successful and failed missions
- Names of booster versions which have carried the max payload
- Failed landing outcomes on drone ship, their booster version and launch site for the months in the year 2015
- Count of landing outcomes between 2010-06-04 and 2017-03-20 (desc)

Build an Interactive Map with Folium

Markers Indicating Launch Sites

• Added blue circle at NASA Johnson Space Center's coordinate with a popup label showing its name using its latitude and longitude coordinates

• Added red circles at all launch sites coordinates with a popup label showing its name using its name using its latitude and longitude coordinates Map with Folium

Colored Markers of Launch Outcomes

• Added colored markers of successful (green) and unsuccessful (red) launches at each launch site to show which launch sites have high success rates

Distances Between a Launch Site to Proximities

• Added colored lines to show distance between launch site CCAFS SLC40 and its proximity to the nearest coastline, railway, highway, and city

Build a Dashboard with Plotly Dash

Dropdown List with Launch Sites

• Allow user to select all launch sites or a certain launch site

Slider of Payload Mass Range

• Allow user to select payload mass range

Pie Chart Showing Successful Launches

• Allow user to see successful and unsuccessful launches as a percent of the total

Scatter Chart Showing Payload Mass vs. Success Rate by Booster Version

• Allow user to see the correlation between Payload and Launch Success

Predictive Analysis (Classification)

Charts

- Create NumPy array from the Class column
- Standardize the data with StandardScaler. Fit and transform the data.
- Split the data using train_test_split
- Create a GridSearchCV object with cv=10 for parameter optimization
- Apply GridSearchCV on different algorithms: logistic regression (LogisticRegression()), support vector machine (SVC()), decision tree (DecisionTreeClassifier()), K-Nearest Neighbor (KNeighborsClassifier())
- Calculate accuracy on the test data using .score() for all models
- Assess the confusion matrix for all models
- Identify the best model using Jaccard_Score, F1_Score and Accuracy

Results

Exploratory Data Analysis

• Launch success has improved over time

• KSC LC-39A has the highest success rate among landing sites

• Orbits ES-L1, GEO, HEO and SSO have a 100% success rate

Visual Analytics

• Most launch sites are near the equator, and all are close to the coast

• Launch sites are far enough away from anything a failed launch can damage (city, highway, railway), while still close enough to bring people and material to support launch activities

Predictive Analytics

• Decision Tree model is the best predictive model for the dataset

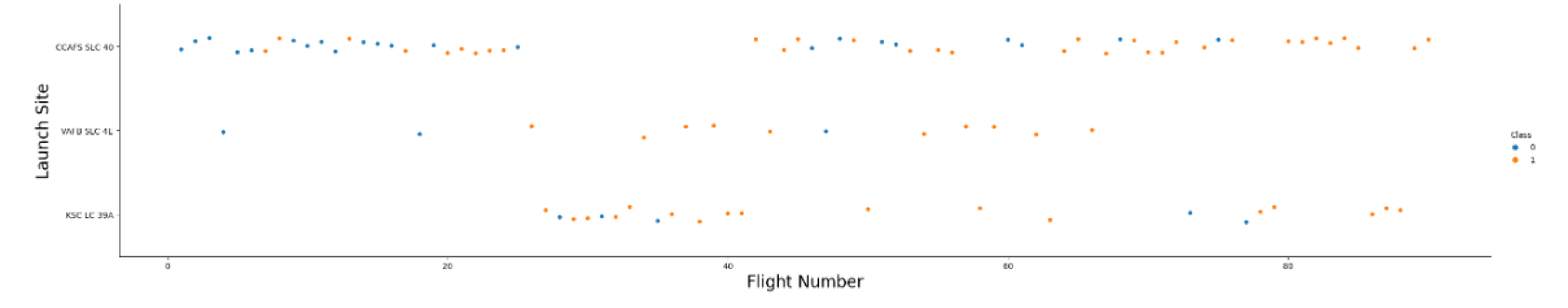
Section 2

Insights drawn from EDA

Flight Number vs. Launch Site

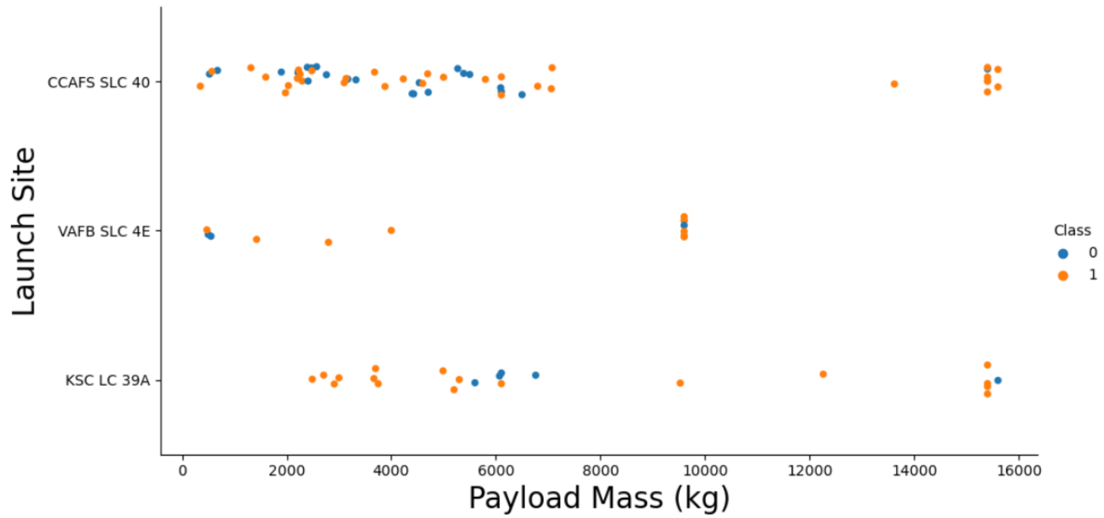
Exploratory Data Analysis

- Earlier flights had a lower success rate (blue = fail)
- Later flights had a higher success rate (orange = success)
- Around half of launches were from CCAFS SLC 40 launch site
- VAFB SLC 4E and KSC LC 39A have higher success rates
- We can infer that new launches have a higher success rate



Payload vs. Launch Site

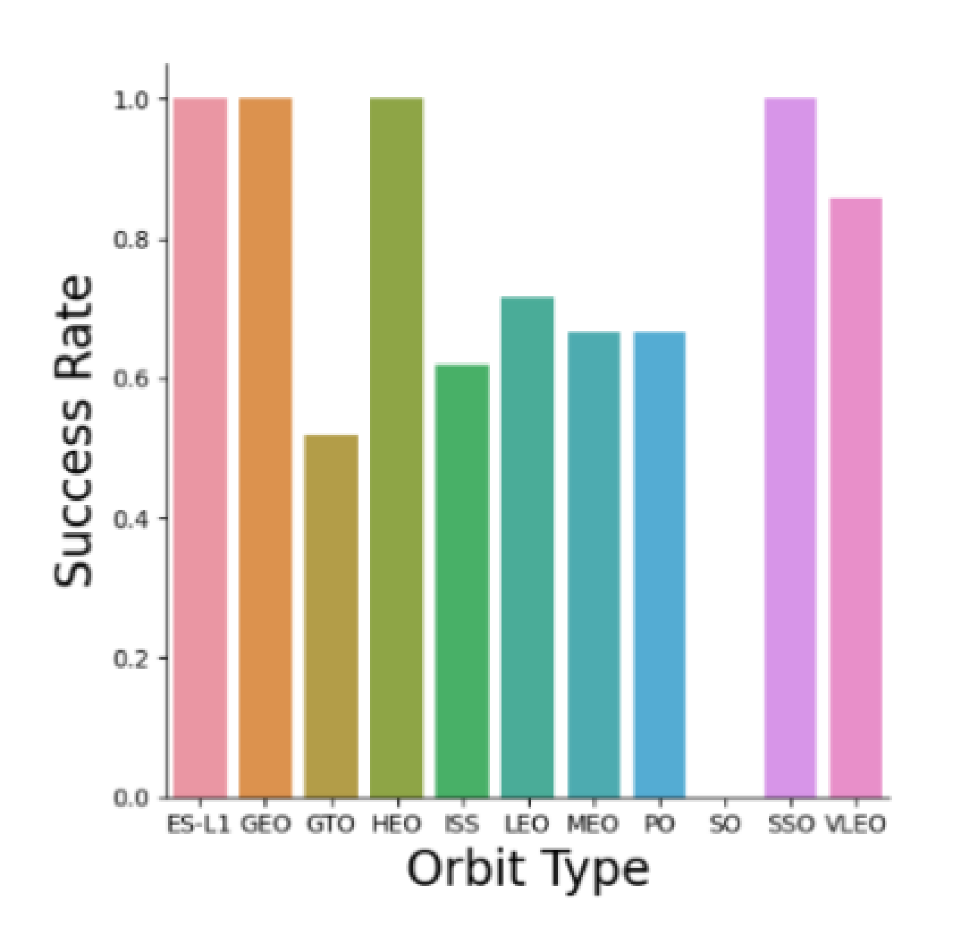
- **Exploratory Data Analysis**
- • Typically, the higher the payload mass (kg), the higher the success rate
- • Most launces with a payload greater than 7,000 kg were successful
- • KSC LC 39A has a 100% success rate for launches less than 5,500 kg
- • VAFB SKC 4E has not launched anything greater than ~10,000 kg



Success Rate vs. Orbit Type

Exploratory Data Analysis

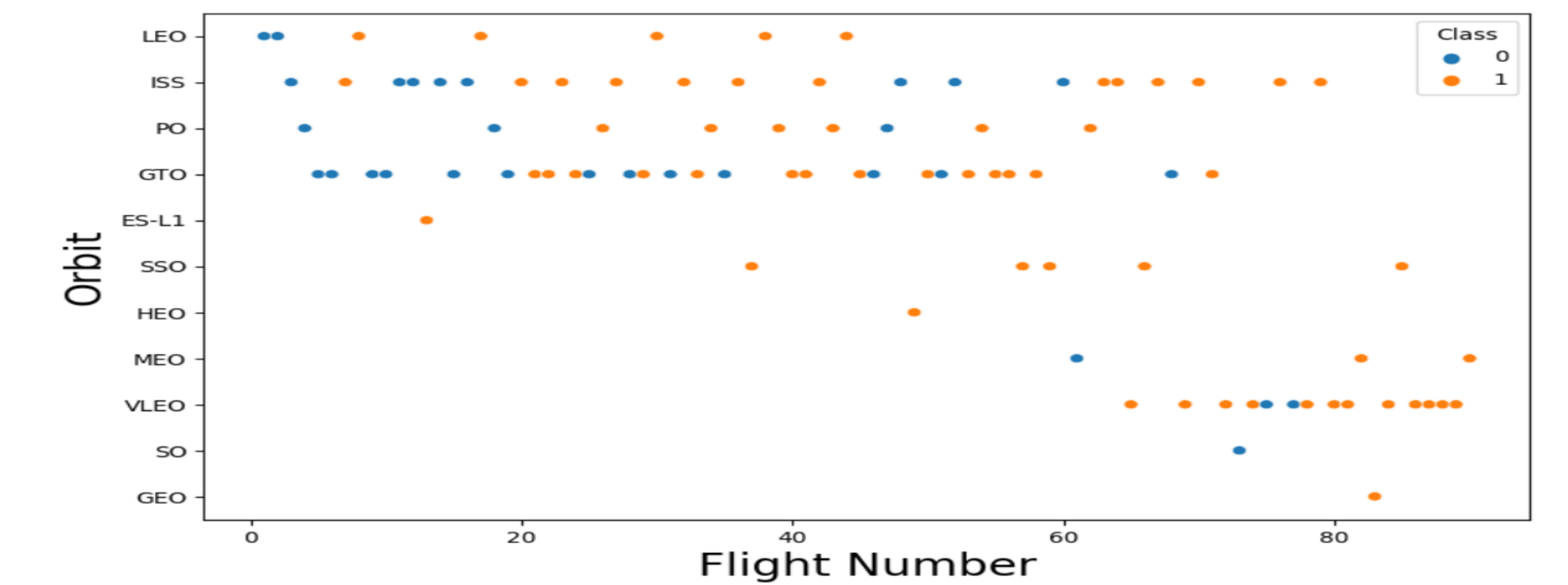
- **100% Success Rate**: ES-L1, GEO, HEO and SSO
- **50%-80% Success Rate**: GTO, ISS, LEO, MEO, PO
- **0% Success Rate**: SO



Flight Number vs. Orbit Type

Exploratory Data Analysis

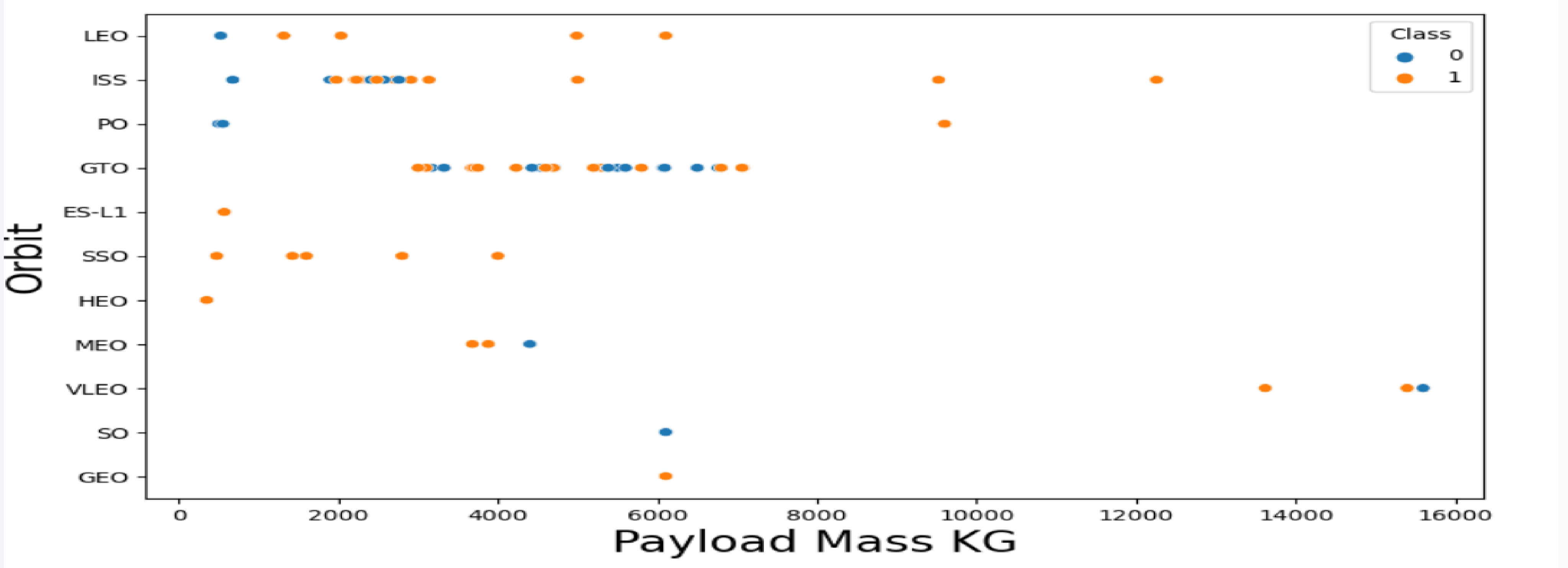
- The success rate typically increases with the number of flights for each orbit
- This relationship is highly apparent for the LEO orbit
- The GTO orbit, however, does not follow this trend



Payload vs. Orbit Type

Exploratory Data Analysis

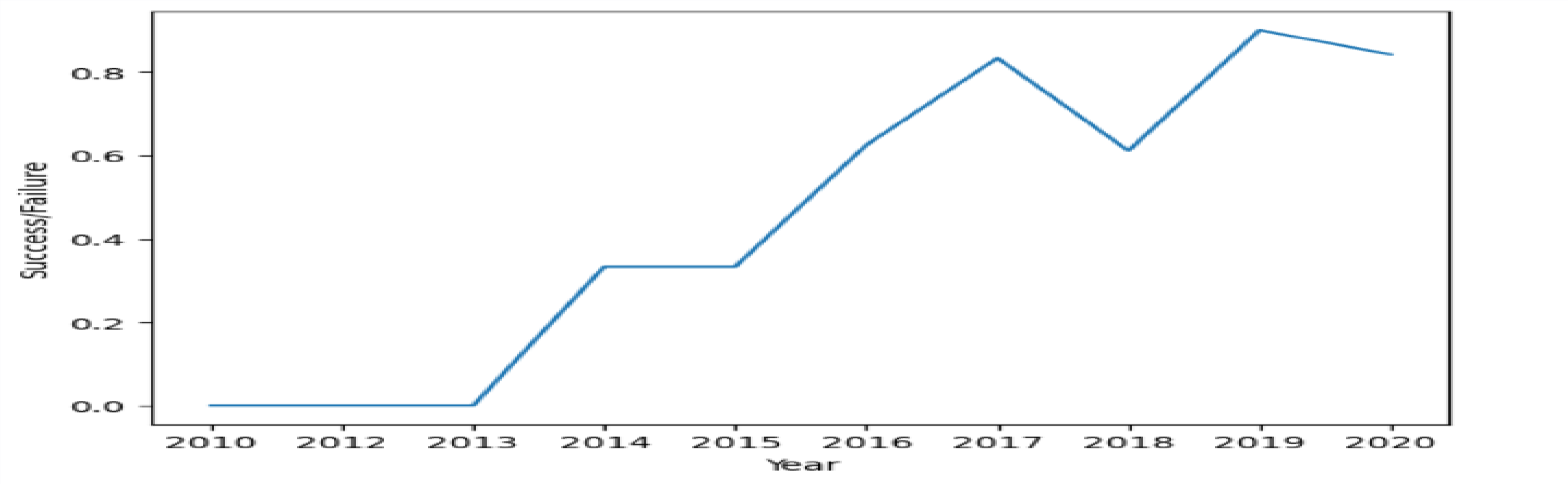
- Heavy payloads are better with LEO, ISS and PO orbits
- The GTO orbit has mixed success with heavier payloads



Launch Success Yearly Trend

Exploratory Data Analysis

- The success rate improved from 2013-2017 and 2018-2019
- The success rate decreased from 2017-2018 and from 2019-2020
- Overall, the success rate has improved since 2013



All Launch Site Names

Launch Site Names

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

Records with Launch Site Starting with CCA

- Displaying 5 records below

In [19]:

```
%sql select * from SPACEXTBL where Launch_Site like 'CCA%' limit 5;
```

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

Out[19]:

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
2010-04-06	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
2010-08-12	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
2012-08-10	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
2013-01-03	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

Total Payload Mass

Payload Mass

- 61,997 kg (total) carried by boosters launched by NASA (CRS)

Average Payload Mass

- 2,928 kg (average) carried by booster version F9 v1.1

```
In [21]: %sql select sum(PAYLOAD_MASS__KG_) as Sum_of_Payload from SPACEXTBL

 * sqlite:///my_data1.db
Done.

Out[21]: Sum_of_Payload
         619967
```

```
In [22]: %sql select avg(PAYLOAD_MASS__KG_) as AVG_PAYLOAD from SPACEXTBL where Booster_Version='F9 v1.1';

 * sqlite:///my_data1.db
Done.

Out[22]: AVG_PAYLOAD
         2928.4
```

First Successful Ground Landing Date

1st Successful Landing in Ground Pad

- 12/22/2015

```
In [30]: %sql select Min(Date) from SPACEXTBL where Landing_Outcome like '%ground pad%';

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

Out[30]:

Min(Date)
2015-12-22

Booster Drone Ship Landing

- Booster mass greater than 4,000 but less than 6,000
- JSCAT-14, JSCAT-16, SES-10, SES-11 / EchoStar 105 B

```
In [35]: %sql select Payload from SPACEXTBL where Landing_Outcome like '%drone ship%' and PAYLOAD_MASS__KG_ between 4000 and 6000;

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

Out[35]:

Payload
SES-9
JCSAT-14
JCSAT-16
SES-10
SES-11 / EchoStar 105

Total Number of Successful and Failed Mission Outcomes

- 1 Failure in Flight
- 99 Success
- 1 Success (payload status unclear)

```
In [42]: %sql select Mission_Outcome,count(*) from SPACEXTBL group by Mission_Outcome;

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

Out[42]:

Mission_Outcome	count(*)
Failure (in flight)	1
Success	98
Success	1
Success (payload status unclear)	1

Boosters Carried Maximum Payload

- 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

In [47]:

```
%sql select Booster_Version, MAX(PAYLOAD_MASS__KG_) from SPACEXTBL group by Booster_Version
```

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

Out[47]:

Booster_Version	MAX(PAYLOAD_MASS__KG_)
F9 B4 B1039.2	2647
F9 B4 B1040.2	5384
F9 B4 B1041.2	9600
F9 B4 B1043.2	6460
F9 B4 B1039.1	3310
F9 B4 B1040.1	4990
F9 B4 B1041.1	9600
F9 B4 B1042.1	3500
F9 B4 B1043.1	5000
F9 B4 B1044	6092
F9 B4 B1045.1	362
F9 B4 B1045.2	2697
F9 B5 B1046.1	3600
F9 B5 B1046.2	5800
F9 B5 B1046.3	4000
F9 B5 B1046.4	12050

2015 Launch Records

2015

- Showing month, date, booster version, launch site and landing outcome

```
%sql select substr(Date,6,2) as month, Date, Booster_Version, Launch_Site, [Landing_Outcome] from SPACEXTBL where [Landing_(
```

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

month	Date	Booster_Version	Launch_Site	Landing_Outcome
10	2015-10-01	F9 v1.1 B1012	CCAFS LC-40	Failure (drone ship)
04	2015-04-14	F9 v1.1 B1015	CCAFS LC-40	Failure (drone ship)
01	2016-01-17	F9 v1.1 B1017	VAFB SLC-4E	Failure (drone ship)
04	2016-04-03	F9 FT B1020	CCAFS LC-40	Failure (drone ship)
06	2016-06-15	F9 FT B1024	CCAFS LC-40	Failure (drone ship)

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

- Count of landing outcomes between 2010-06-04 and 2017-03-20 in descending order

```
%sql SELECT [Landing _Outcome], count(*) as count_outcomes \
FROM SPACEXTBL \
WHERE DATE between '04-06-2010' and '20-03-2017' group by [Landing _Outcome] order by count_outcomes DESC;
```

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

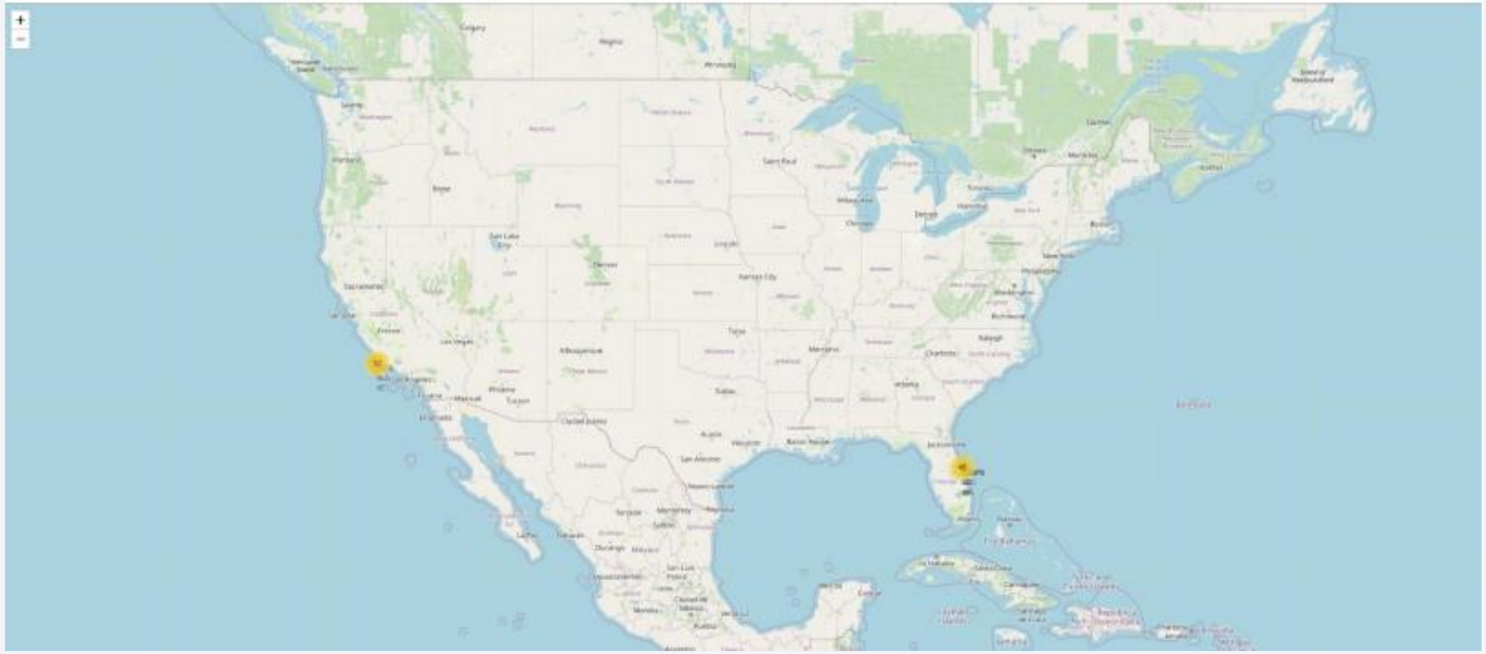
Landing _Outcome	count_outcomes
Success	20
No attempt	10
Success (drone ship)	8
Success (ground pad)	6
Failure (drone ship)	4
Failure	3
Controlled (ocean)	3
Failure (parachute)	2
No attempt	1

Section 3

Launch Sites Proximities Analysis

Launch Site with Markers

- Near Equator: the closer the launch site to the equator, the easier it is to launch to equatorial orbit, and the more help you get from Earth's rotation for a prograde orbit. Rockets launched from sites near the equator get an additional natural boost - due to the rotational speed of earth - that helps save the cost of putting in extra fuel and boosters



Launch Outcomes

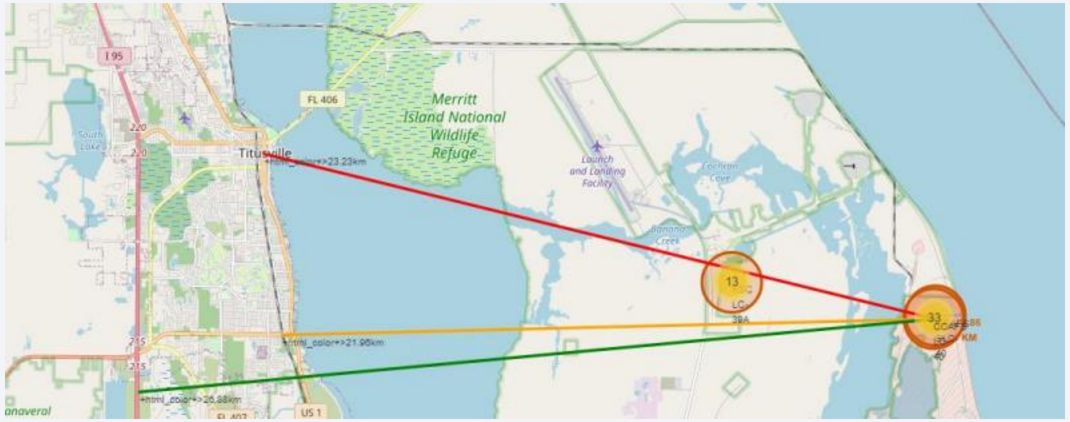
At Each Launch Site

- Outcomes:
- Green markers for successful launches
- Red markers for unsuccessful launches
- Launch site CCAFS SLC-40 has a 3/7 success rate (42.9%)

Distance to Proximities

CCAFS SLC-40

- .86 km from nearest coastline
- 21.96 km from nearest railway
- 23.23 km from nearest city
- 26.88 km from nearest highway



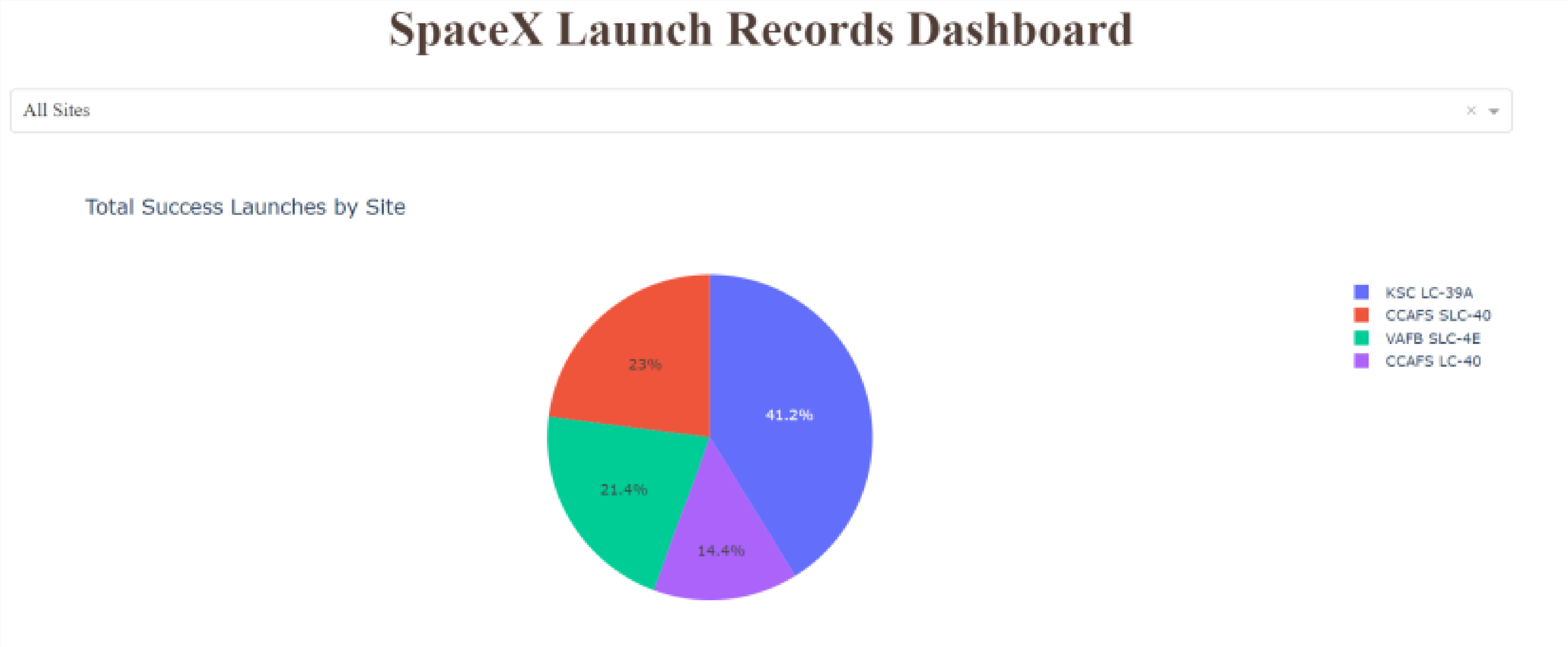
Section 4

Build a Dashboard with Plotly Dash

Launch Success by Site

Success as Percent of Total

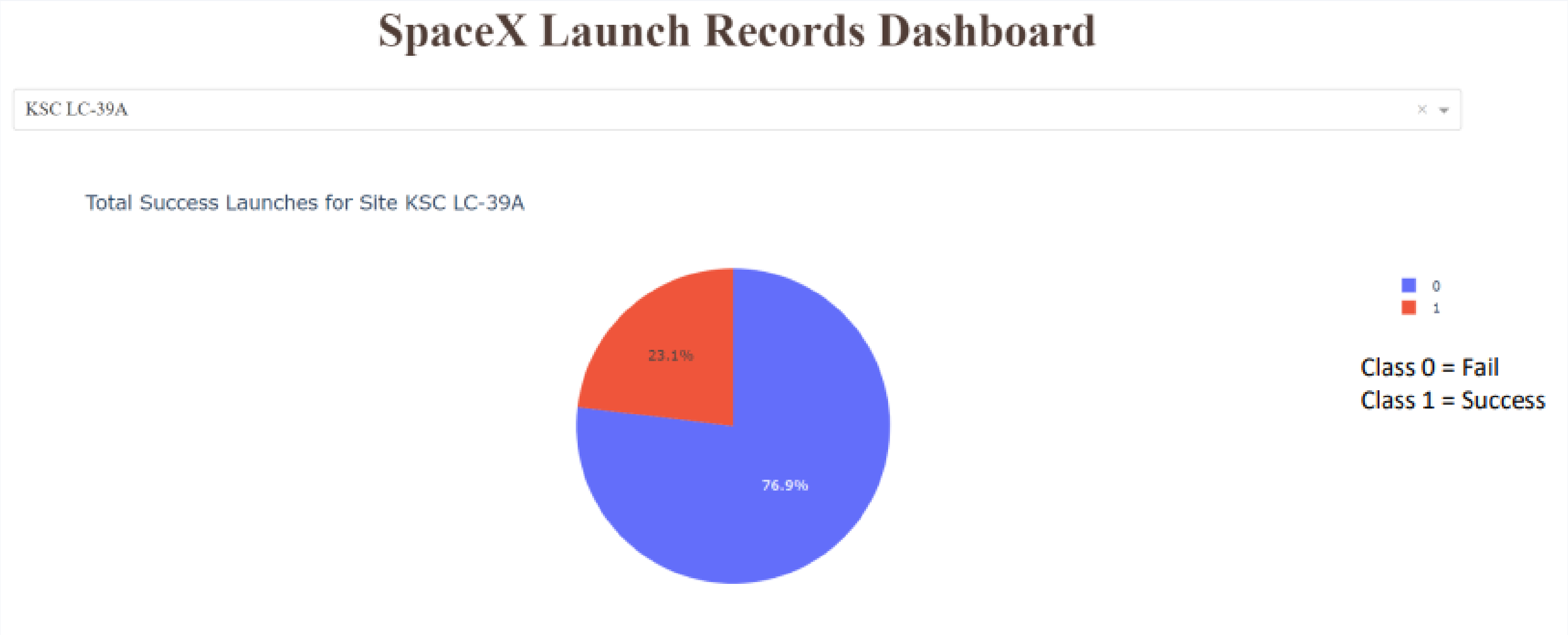
- KSC LC-39A has the most successful launches amongst launch sites (41.2%)



Launch Success (KSC LC-29A)

Success as Percent of Total

- KSC LC-39A has the highest success rate amongst launch sites (76.9%)
- 10 successful launches and 3 failed launches



Payload Mass and Success

By Booster Version

- Payloads between 2,000 kg and 5,000 kg have the highest success rate
- 1 indicating successful outcome and 0 indicating an unsuccessful outcome

Section 5

Predictive Analysis (Classification)

Classification Accuracy

Accuracy

- All the models performed at about the same level and had the same scores and accuracy. This is likely due to the small dataset. The Decision Tree model slightly outperformed the rest when looking at .best_score_
- .best_score_ is the average of all cv folds for a single combination of the parameters

	LogReg	SVM	Tree	KNN
Jaccard_Score	0.800000	0.800000	0.800000	0.800000
F1_Score	0.888889	0.888889	0.888889	0.888889
Accuracy	0.833333	0.833333	0.833333	0.833333

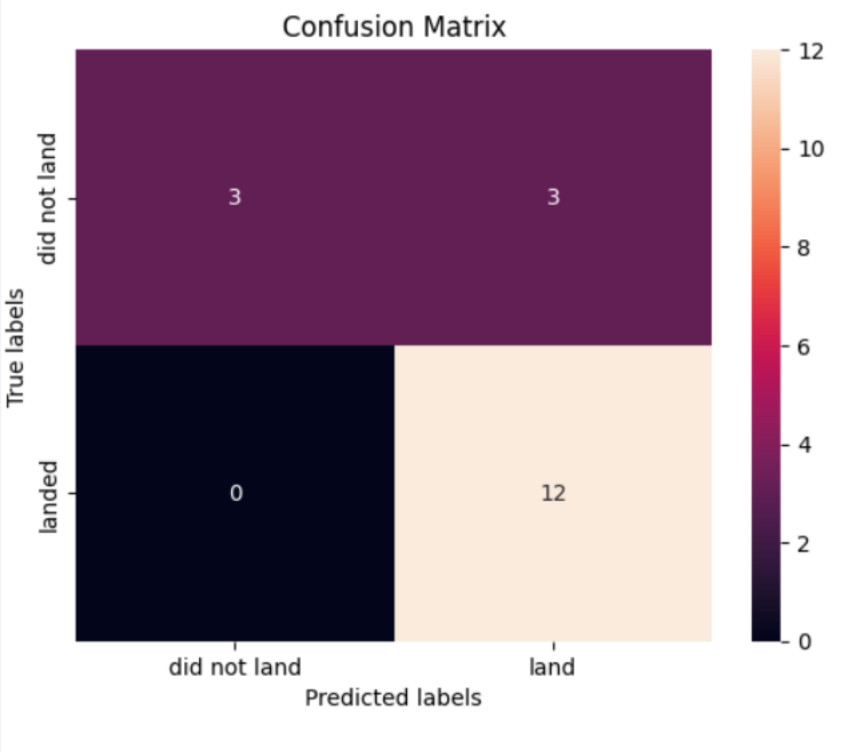
```python
models = {'KNeighbors':knn_cv.best_score_,
              'DecisionTree':tree_cv.best_score_,
              'LogisticRegression':logreg_cv.best_score_,
              'SupportVector': svm_cv.best_score_}

bestalgorithm = max(models, key=models.get)
print('Best model is', bestalgorithm,'with a score of', models[bestalgorithm])
if bestalgorithm == 'DecisionTree':
    print('Best params is :', tree_cv.best_params_)
if bestalgorithm == 'KNeighbors':
    print('Best params is :', knn_cv.best_params_)
if bestalgorithm == 'LogisticRegression':
    print('Best params is :', logreg_cv.best_params_)
if bestalgorithm == 'SupportVector':
    print('Best params is :', svm_cv.best_params_)
```

```
Best model is DecisionTree with a score of 0.8875
Best params is : {'criterion': 'gini', 'max_depth': 4, 'max_features': 'sqrt', 'min_samples_leaf': 2, 'min_samples_split':
10, 'splitter': 'random'}
```

Confusion Matrix

Performance Summary

- A confusion matrix summarizes the performance of a classification algorithm
- All the confusion matrices were identical
- The fact that there are false positives (Type 1 error) is not good
- Confusion Matrix Outputs:
    - 12 True positive
    - 3 True negative
    - 3 False positive
    - 0 False Negative
- Precision = TP / (TP + FP)
    - 12 / 15 = .80
- Recall = TP / (TP + FN)
    - 12 / 12 = 1
- F1 Score = 2 * (Precision * Recall) / (Precision + Recall)
    - 2 * (.8 * 1) / (.8 + 1) = .89
- Accuracy = (TP + TN) / (TP + TN + FP + FN) = .833



Conclusions

Research

- **Model Performance**: The models performed similarly on the test set with the decision tree model slightly outperforming
- **Equator**: Most of the launch sites are near the equator for an additional natural boost - due to the rotational speed of earth - which helps save the cost of putting in extra fuel and boosters
- **Coast**: All the launch sites are close to the coast
- **Launch Success**: Increases over time
- **KSC LC-39A**: Has the highest success rate among launch sites. Has a 100% success rate for launches less than 5,500 kg
- **Orbits**: ES-L1, GEO, HEO, and SSO have a 100% success rate
- **Payload Mass**: Across all launch sites, the higher the payload mass (kg), the higher the success rate

Conclusion - Things to consider

- **Dataset:** A larger dataset will help build on the predictive analytics results to help understand if the findings can be generalizable to a larger data set
- **Feature Analysis / PCA:** Additional feature analysis or principal component analysis should be conducted to see if it can help improve accuracy
- **XGBoost**: Is a powerful model which was not utilized in this study. It would be interesting to see if it outperforms the other classification models

Thank you!