IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

<Walter Medina>
<16/12/2024>

Outline

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

Executive Summary

- **Summary of methodologies**

The present research attempts to identify the important factors for a successful landing, to approach this, the following methodologies where used:

- Collect
- Explores and Analyze
- Build Model

- **Results**

The model that performed better is the Decision tree model with 83% of accuracy for a successful landing

Introduction

- **Project background and context**

SpaceX, a one of the leaders in the space industry want to make space travel affordable for everyone and for research missions. SpaceX search for save a money because these attempts of landing are expensive. To reduce it we can predict if the landing are successful or not.

- **Problems you want to find answers**

- Factors can be affecting the successful landing.

- Best predictive model

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Data of SpaceX Rest API (Web Scraping)
- Perform data wrangling
  - Filtering data, handling missing values and applying 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
  - Tests with 5 models and use tuning for search the best parameters.

Data Collection – SpaceX API

- **Request data** from Space X website
- **Decode Response** using .json to normalize
- **Request information** related to the information
- **Create Dictionary**
- **Create Dataframe**
- **Filter Dataframe** with the Falcon 9 launches
- **Handling Missing Values**
- **Export and save data**

Data Collection - Scraping

- **Request data** from Wikipidedia Website
- **Using Beautiful Soup Library**
- **Extract columns** with some labels
- **Collect data**
- **Create diccionary**
- **Create dataframe**
- **Save and export data**

Data Wrangling

- **Perform EDA**
- **Calculations**
  - # of launches, occurrence for orbit/ site/mission
- **Creation of dependent variable**
- **Landing Outcome**
  - If landing was successful.

EDA with Data Visualization

- Charts
  - Flights Number vs Payload
  - Flights Number vs Launch Site
  - Payload Mass(Kg) vs Launch Site
  - Payload Mass(Kg) vs Orbit type
- Analysis
  - Correlation different variables.
  - Some comparisons among categories showing the relationship.

EDA with SQL

- Some queries like:
  - Distinct launch sites
  - Total payload mass carried by boosters
  - Some Average
  - Dates of successful landing
  - Names of boosters with some conditions
  - Total of failed missions

Build an Interactive Map with Folium

- Markers indicating Launch Sites with up pop up label
- Colored markers
    - Successful: Green
    - Unsuccessful: Red
- Distance of proximity
    - To coastline
    - Railway
    - Highway

Build a Dashboard with Plotly Dash

- Dropdown List with Launch Sites
- Pie Chart showing Successful Launches
- Slider of Payload Mass Range
- Scatter Chart (Payload Mass vs Booster Version)

Predictive Analysis (Classification)

- Standardize the data
- Split in train , test dataframe
- Using GridSearchCV
- Apply in different algorithms:
  - Logistic Regression
  - Support Vector Machine
  - Decision Tree Classifier
  - KNN
- Performance of models

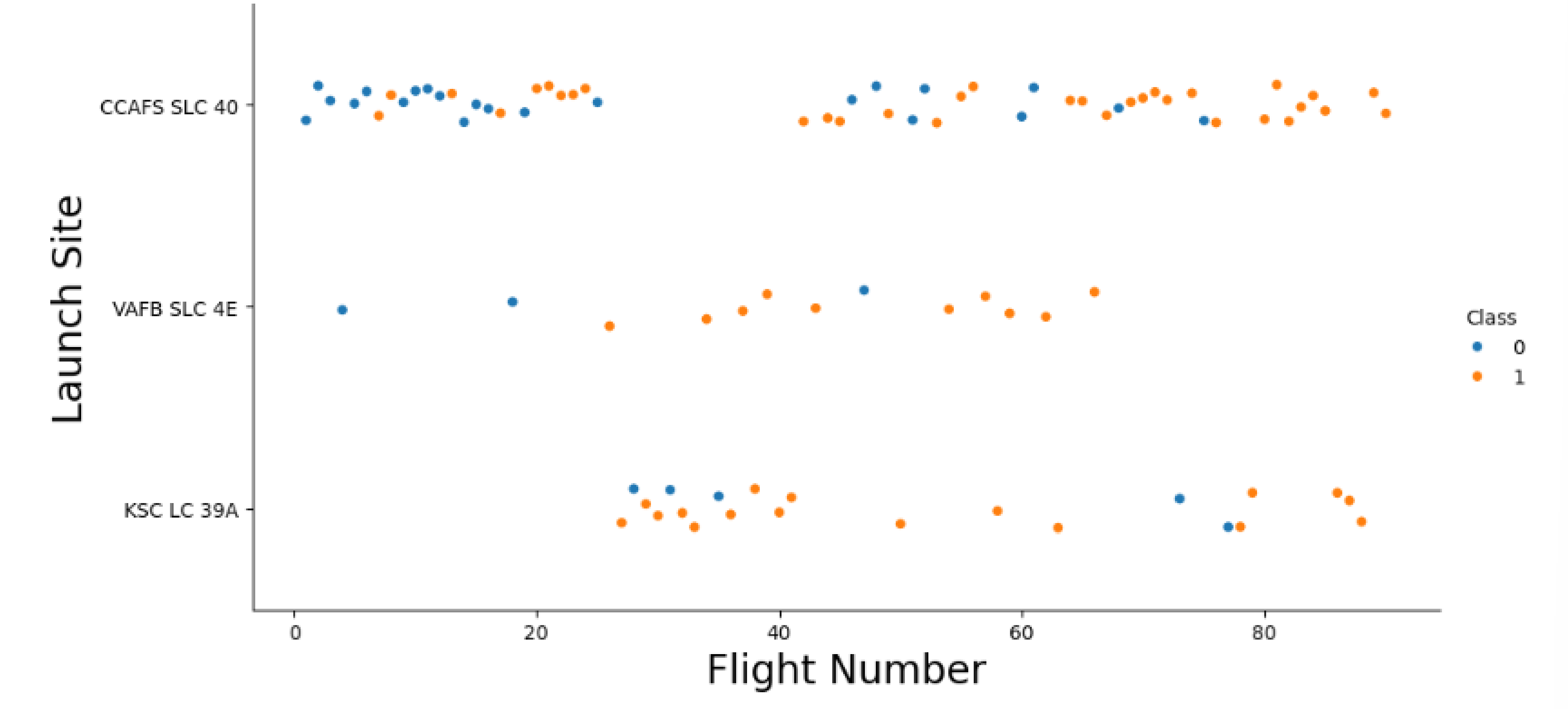
Results

- Exploratory data analysis results
  - Launch success has improved over time
  - Some orbits have a high success rate.
- Interactive analytics
  - Some launch depends of the location and how near are at the coast.
- Predictive analysis results
  - Decision tree model is the best predictive model

Section 2

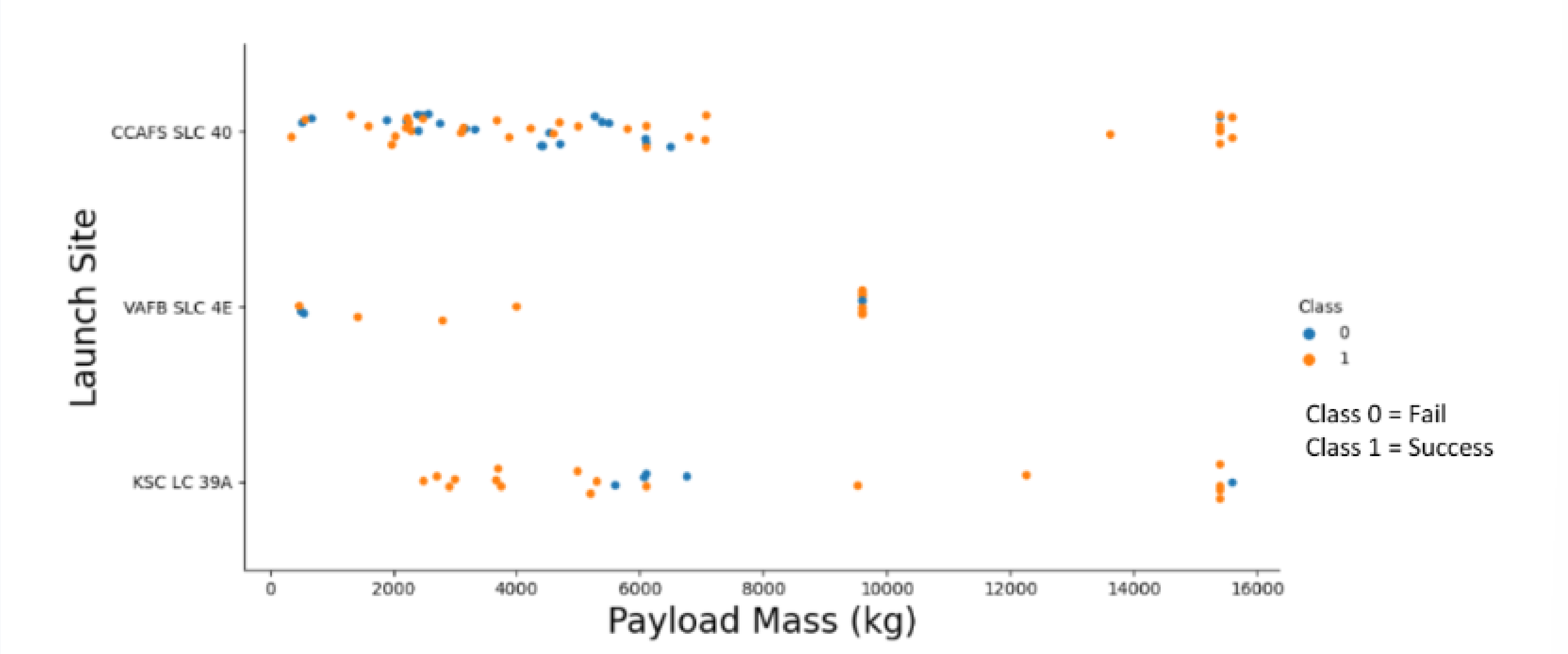
Insights drawn from EDA

Flight Number vs. Launch Site



- Later flights had a success rate
- We can consider that new launches have a higher success rate.

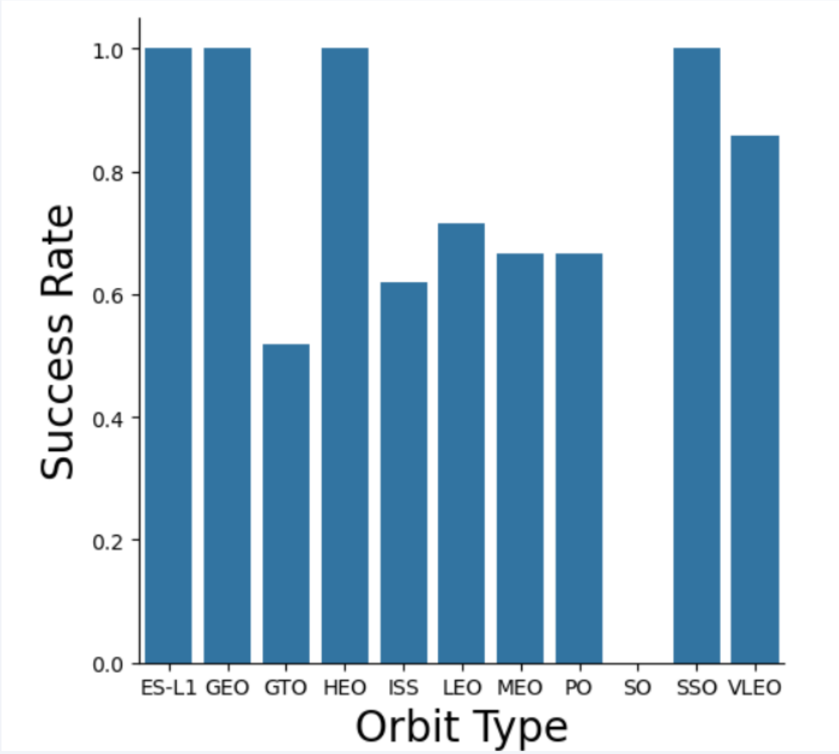
Payload vs. Launch Site



Most Launches with a payload greater than 7000 Kg were successful.

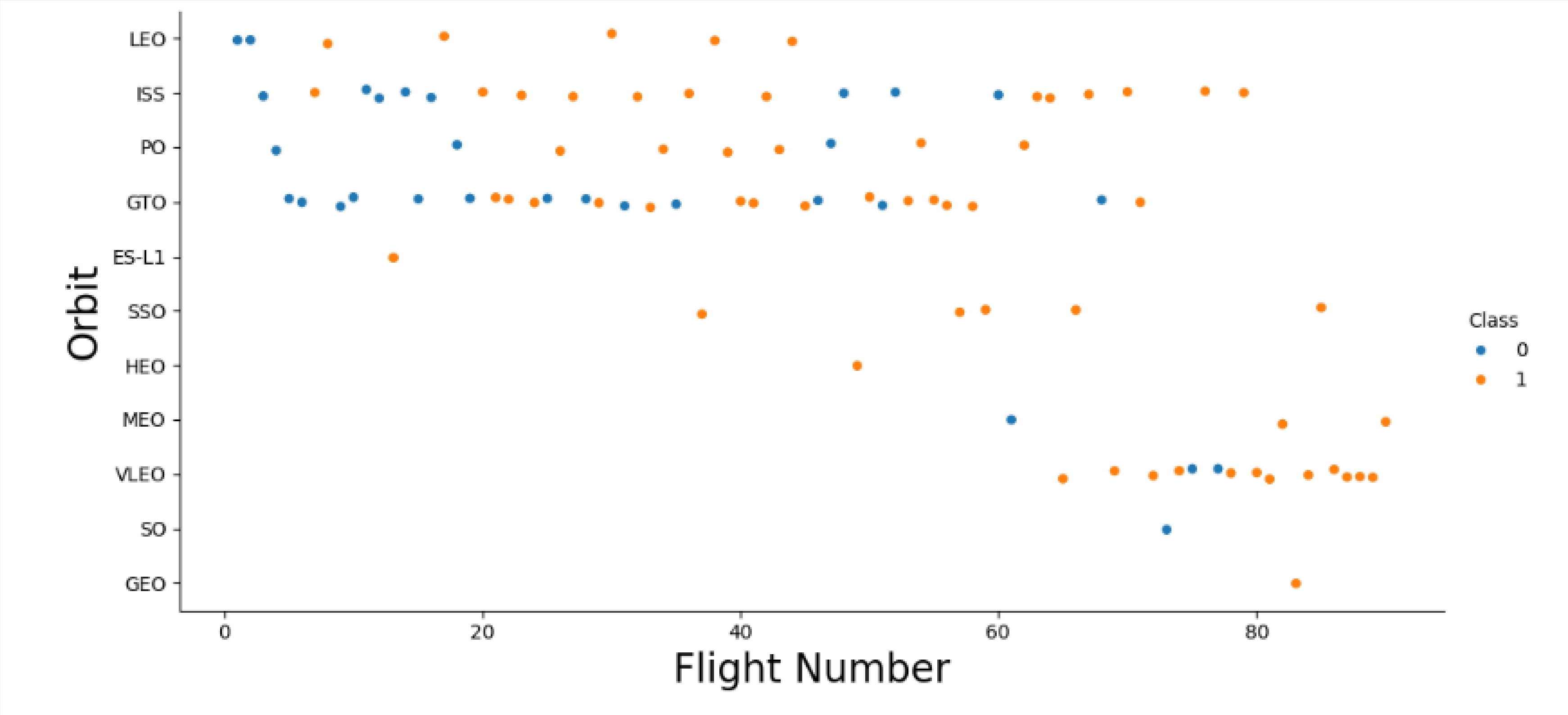
We can consider the higher the payload mass then higher the success rate.

Success Rate vs. Orbit Type



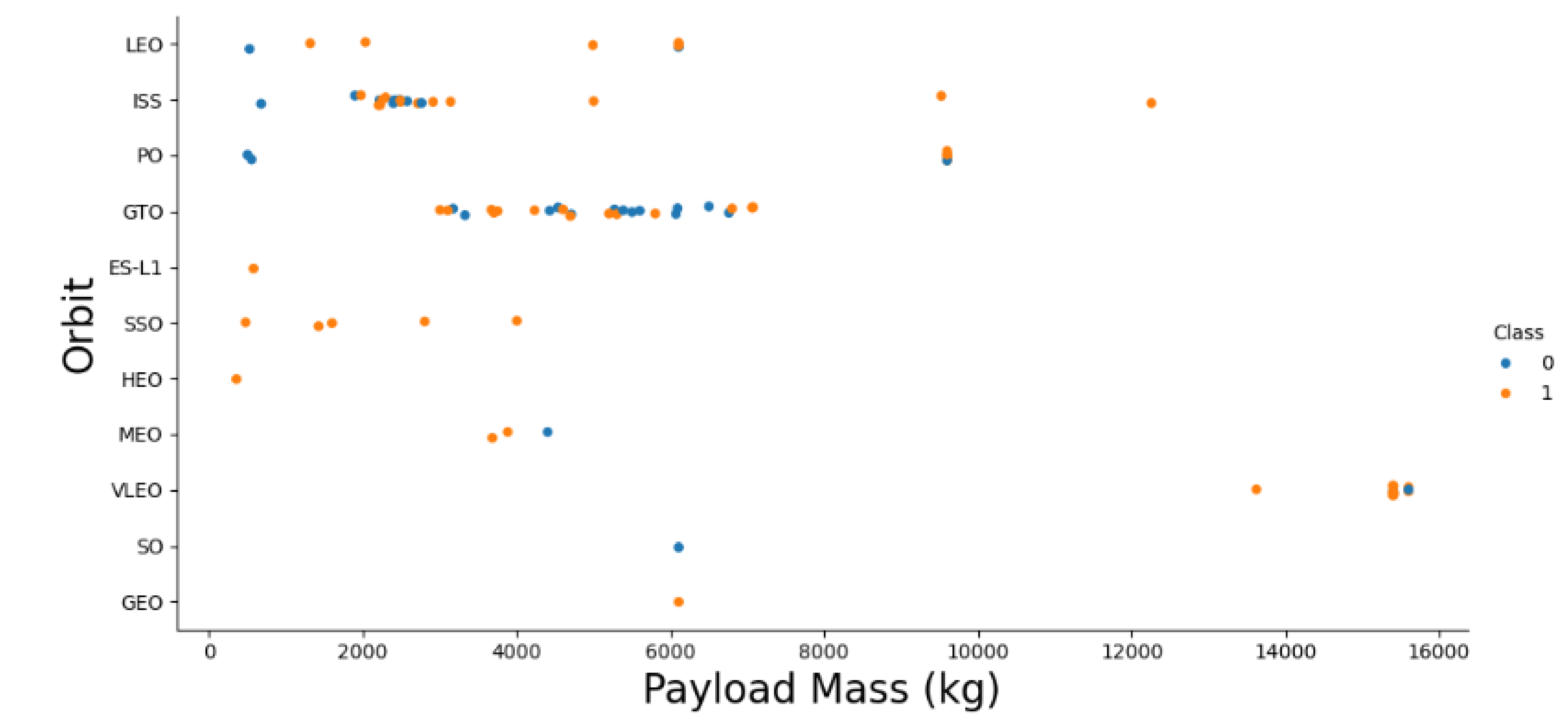
- 100% Success rate: ES- L1, GEO, HEO and SSO
- 0% Success rate: SO

Flight Number vs. Orbit Type



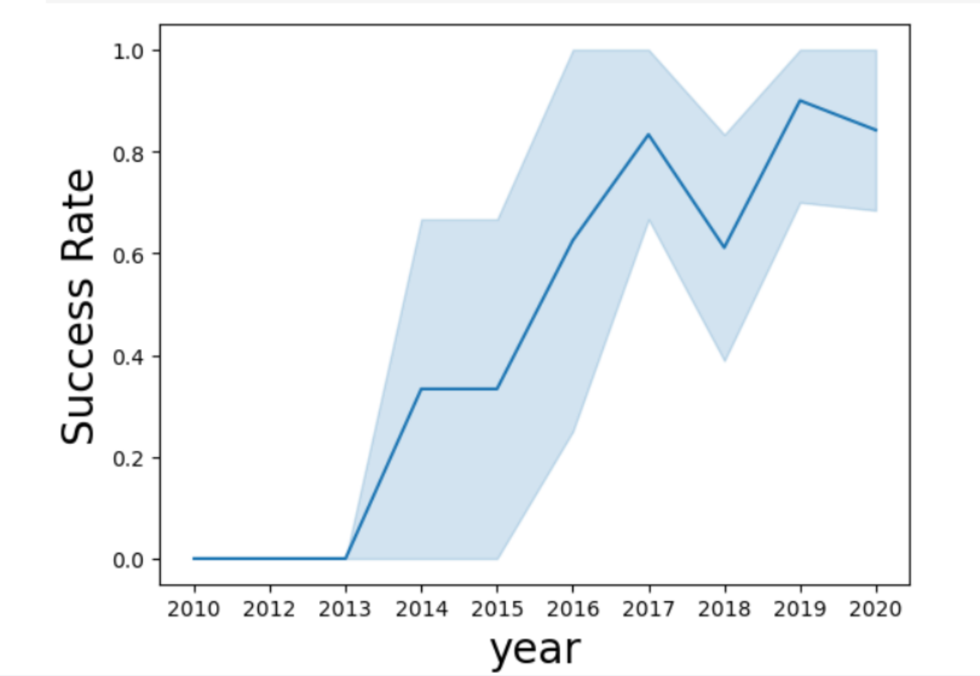
- In Leo orbit the relationship is highly apparent
- Success rate increases with the number of flights

Payload vs. Orbit Type



- With heavy payloads the successful landing or positive landing rate are more for Polar,LEO and ISS.
- For GTO, it's difficult to distinguish between successful and unsuccessful landings as both outcomes are present.

Launch Success Yearly Trend



- We can observe that the success rate since 2013 kept increasing till 2020

All Launch Site Names

- Distinct Launch Sites:

In [13]:

```
%sql SELECT DISTINCT(Launch_Site)  FROM SPACEXTBL
```

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

Out[13]:

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

Launch Site Names Begin with 'CCA'

- 5 records where launch sites begin with CCA:

```
%sql SELECT * \
    FROM SPACEXTBL \
    WHERE LAUNCH_SITE LIKE'CCA%' LIMIT 5
```

```
 * sqlite:///my_data1.db
Done.
Out[14]:
```

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	M
2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	
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	
2012-05-22	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	
2012-10-08	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	
2013-03-01	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	

Total Payload Mass

The total payload carried by boosters from NASA:

In [15]:

```
%sql SELECT SUM(PAYLOAD_MASS__KG_) \
    FROM SPACEXTBL \
    WHERE CUSTOMER = 'NASA (CRS)'
```

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

Out[15]:

SUM(PAYLOAD_MASS__KG_)
45596

Average Payload Mass by F9 v1.1

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

```
In [16]:
%sql SELECT AVG(PAYLOAD_MASS__KG_) \
    FROM SPACEXTBL \
    WHERE BOOSTER_VERSION = 'F9 v1.1';
```

```
 * sqlite:///my_data1.db
Done.
Out[16]:
```

AVG(PAYLOAD_MASS__KG_)
2928.4

First Successful Ground Landing Date

- Firs Susccesful using min function:

```
In [18]:
%sql SELECT MIN(DATE) \
FROM SPACEXTBL \
WHERE Landing_Outcome = 'Success (ground pad)'

 * sqlite:///my_data1.db
Done.
Out[18]:
```

MIN(DATE)
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

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

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 [19]:

```
%sql SELECT PAYLOAD \
FROM SPACEXTBL \
WHERE Landing_Outcome = 'Success (drone ship)' \
AND PAYLOAD_MASS__KG_ BETWEEN 4000 AND 6000;
```

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

Out[19]:

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

Total Number of Successful and Failure Mission Outcomes

- Total number of successful and failure mission outcomes

In [20]:

```
%sql SELECT MISSION_OUTCOME, COUNT(*) as total_number \
FROM SPACEXTBL \
GROUP BY MISSION_OUTCOME
```

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

Out[20]:

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

Boosters Carried Maximum Payload

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

```
In [21]:
%sql SELECT BOOSTER_VERSION \
FROM SPACEXTBL \
WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTBL);

 * sqlite:///my_data1.db
Done.
Out[21]:
```

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

- Failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015

In [23]:

```
%sql SELECT substr(Date,6,2) as month, DATE,BOOSTER_VERSION, LAUNCH_SITE, Landing_Outcome \
FROM SPACEXTBL \
where Landing_Outcome = 'Failure (drone ship)' and substr(Date,0,5)='2015';
```

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

Out[23]:

month	Date	Booster_Version	Launch_Site	Landing_Outcome
01	2015-01-10	F9 v1.1 B1012	CCAFS LC-40	Failure (drone ship)
04	2015-04-14	F9 v1.1 B1015	CCAFS LC-40	Failure (drone ship)

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

- Rank of 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 [25]:

```
%sql SELECT Landing_Outcome, count(*) as count_outcomes \
FROM SPACEXTBL \
WHERE DATE between '2010-06-04' and '2017-03-20' group by Landing_Outcome order by count_out
```

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

Out[25]:

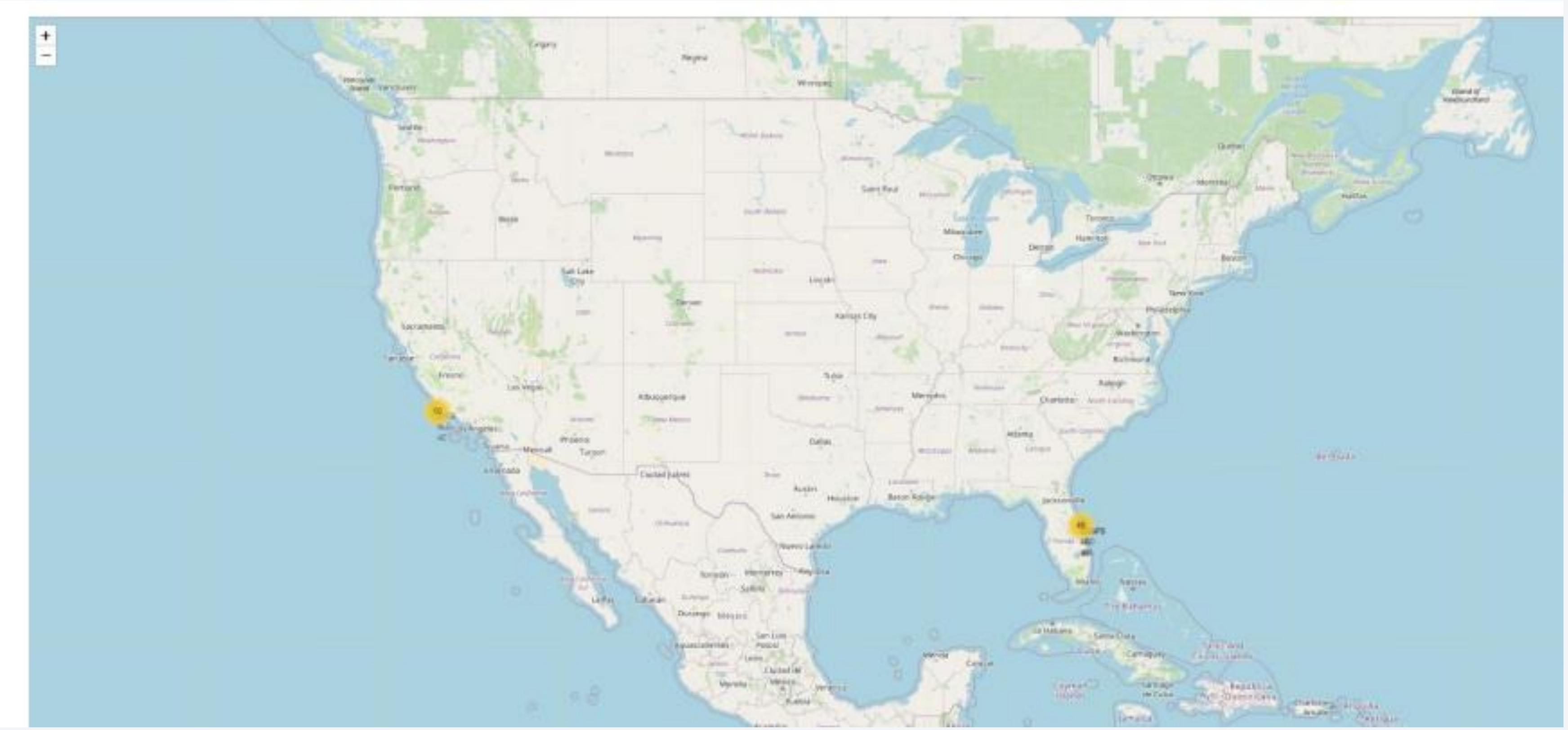
Landing_Outcome	count_outcomes
No attempt	10
Success (drone ship)	5
Failure (drone ship)	5
Success (ground pad)	3
Controlled (ocean)	3
Uncontrolled (ocean)	2
Failure (parachute)	2
Precluded (drone ship)	1

Section 3

Launch Sites Proximities Analysis

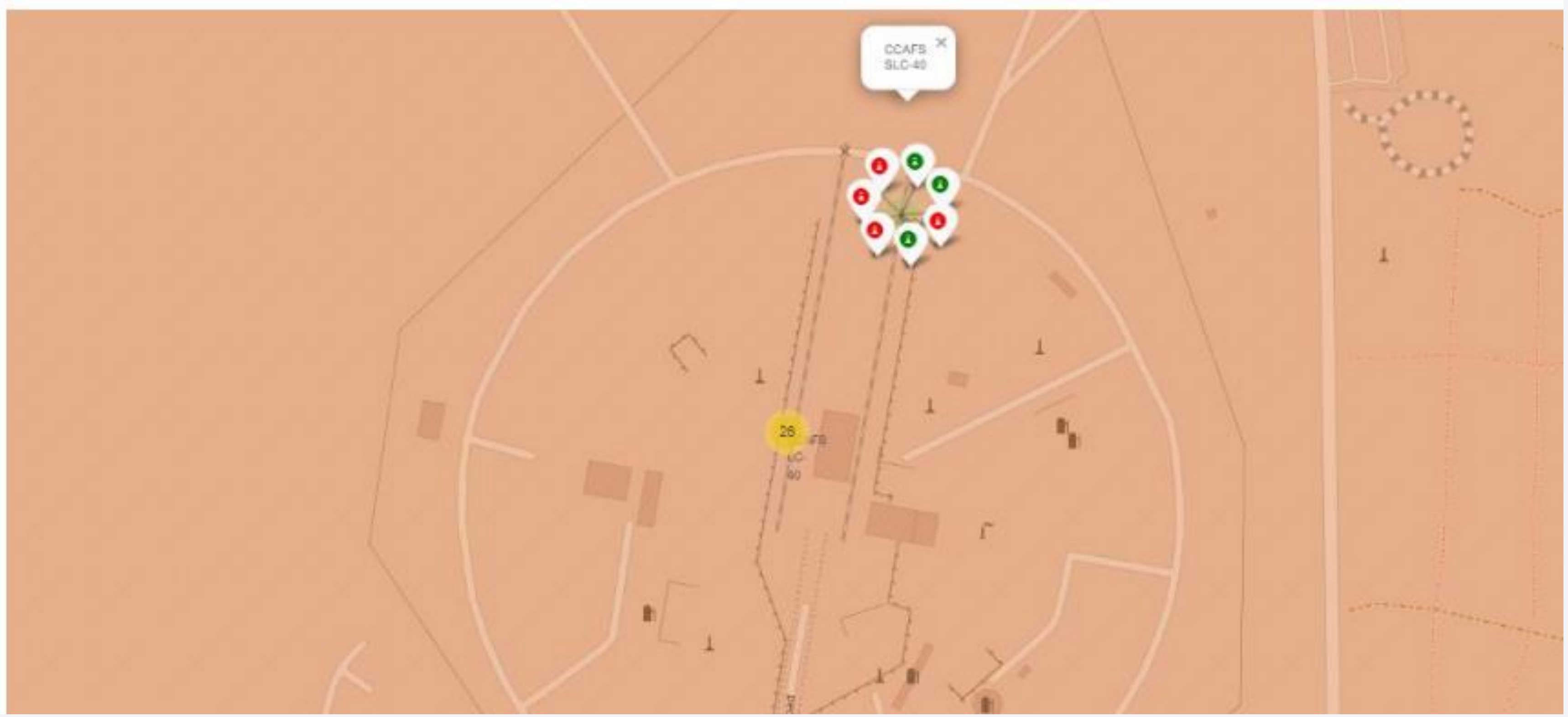
Launch Sites

- With Markets, we can indicate the location of launch sites(successful or unsuccesful)



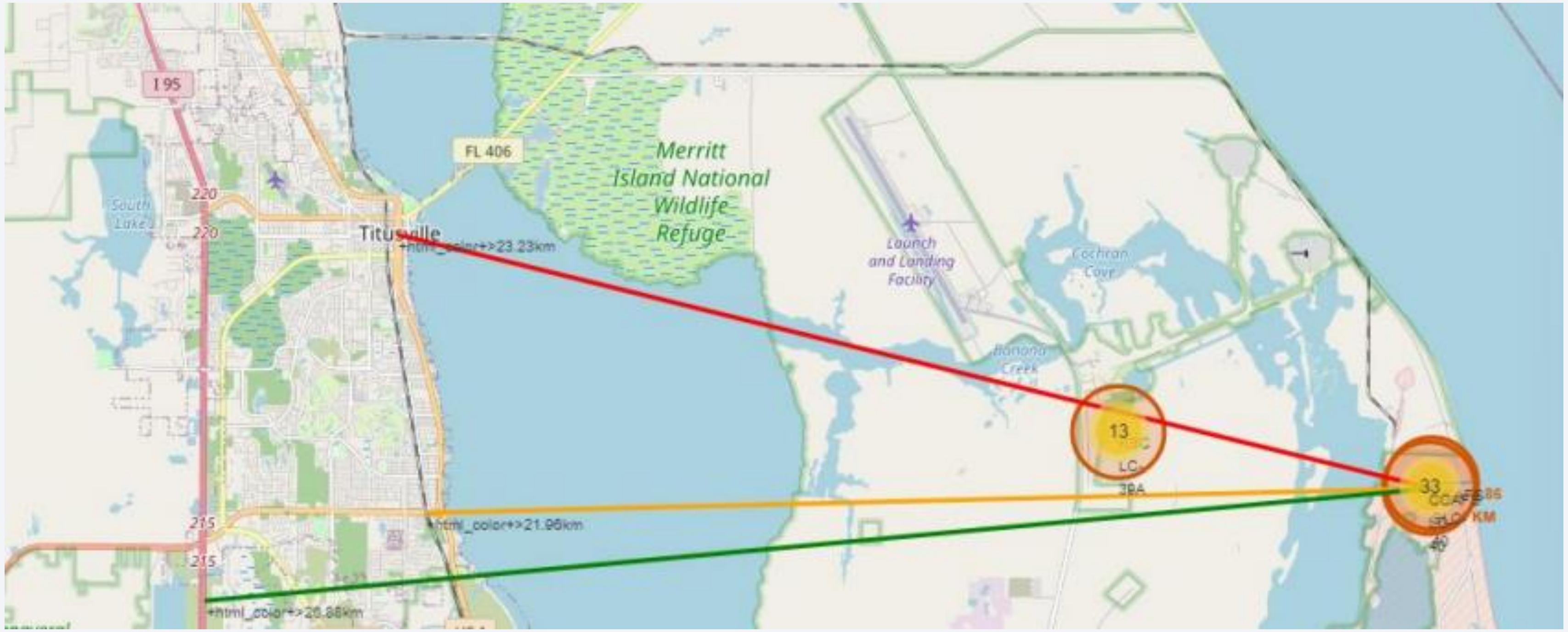
<Launch Outcomes>

- Green markers for successful launches.
- Red markers for unsuccessful launches.



Proximity

- We can draw lines of distance to coastline , railway, highway.

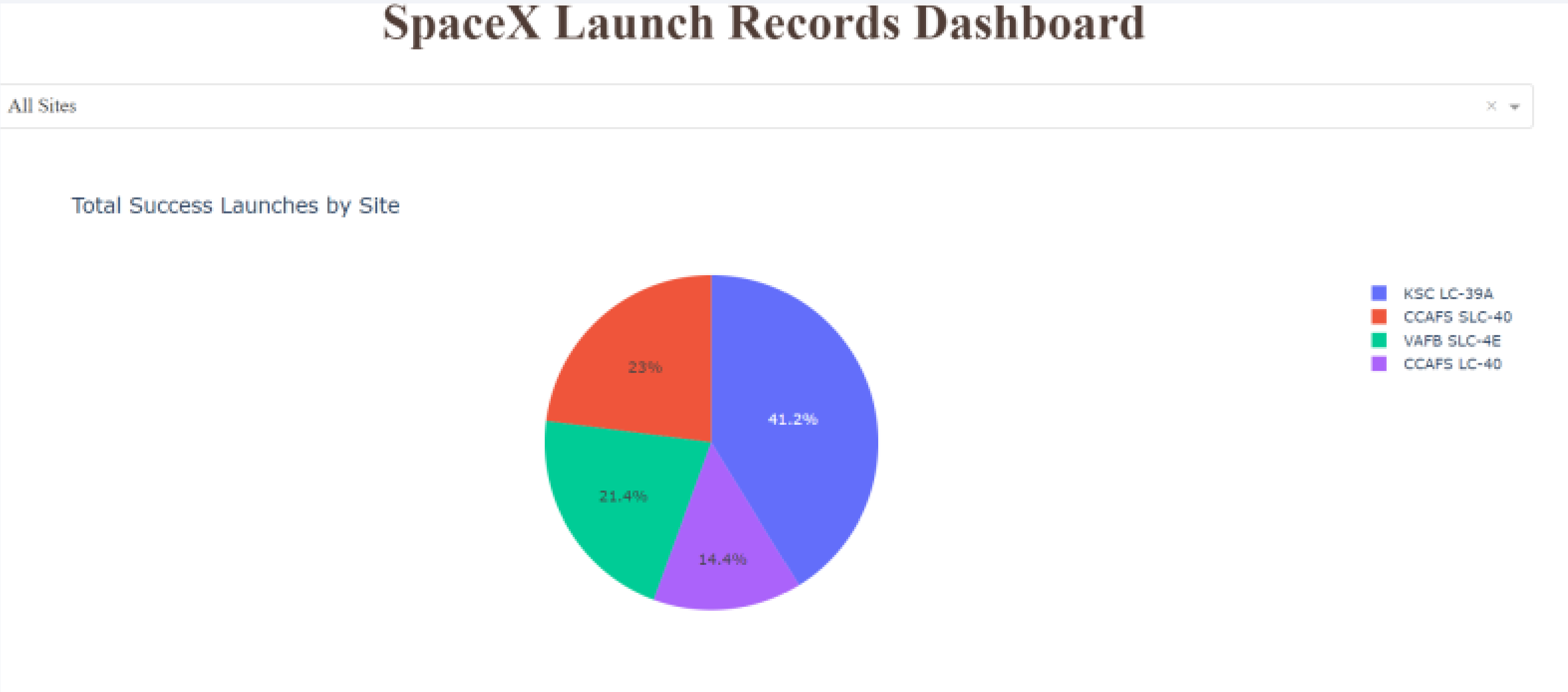


Section 4

Build a Dashboard with Plotly Dash

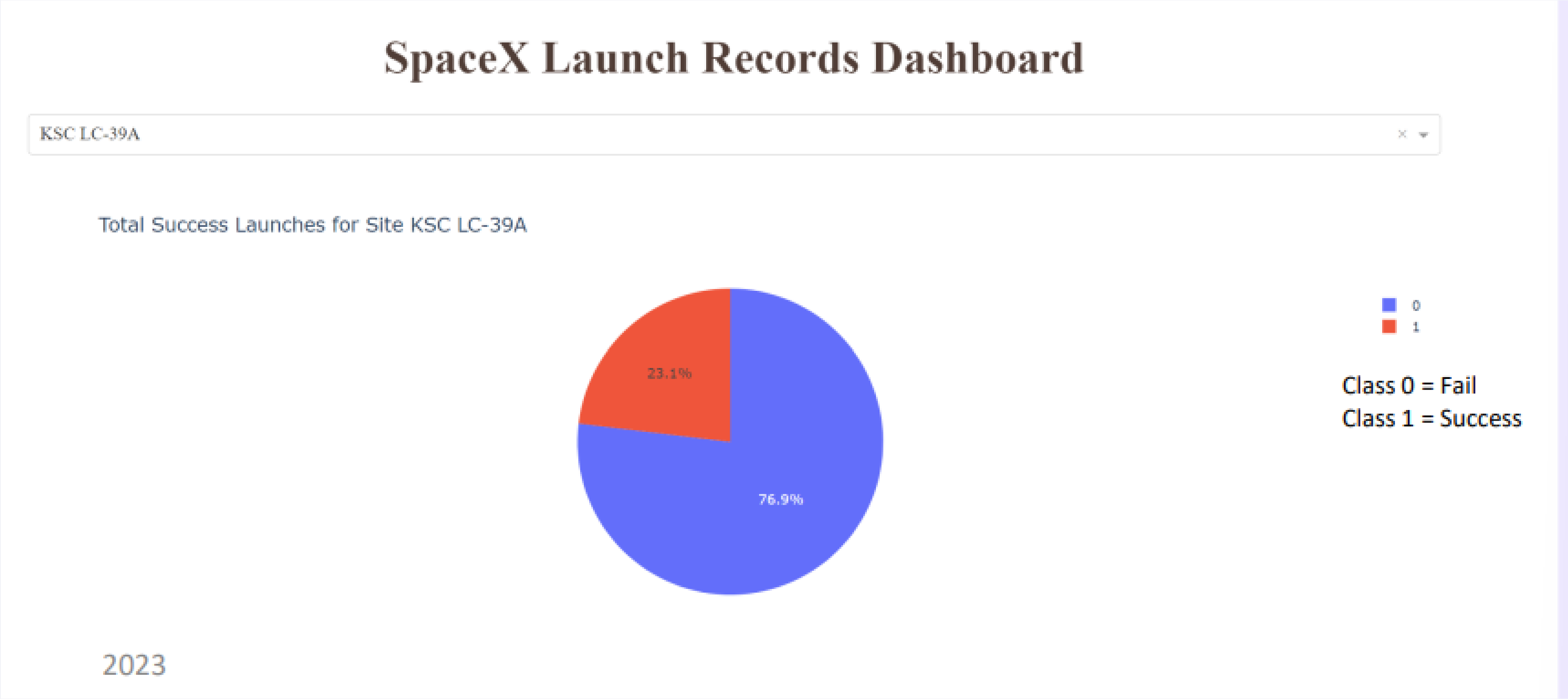
Launches Success by Site

- Success as percent
  - We can see the most successful launches and the most unsuccessful.



Launch Success(LC 29 A)

- It has a 76.9% of success rate.



Payload Mass and Success

- We can see relationship by Payload and Success:

Section 5

Predictive Analysis (Classification)

Classification Accuracy

- The best model is Decision Tree model because the best_score is a mean of 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

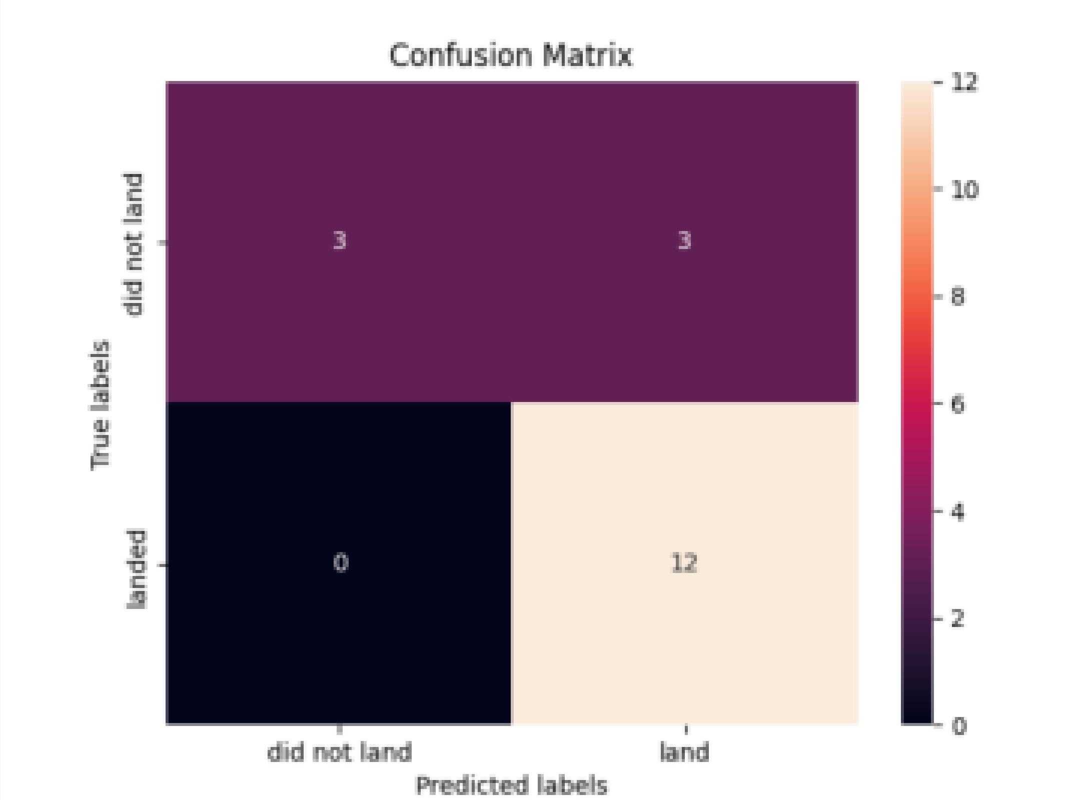
```
models = {'KNeighbors':knn_cv.best_score_,
              'DecisionTree':tree_cv.best_score_,
              'LogisticRegression':logreg_cv.best_score_,
              'SupportVector': svm_cv.best_score_}

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 0.9017857142857142
Best params is : {'criterion': 'gini', 'max_depth': 16, 'max_features': 'auto', 'min_samples_leaf': 4, 'min_samples_split': 10, 'splitter': 'random'}
```

Confusion Matrix

- 12 True Positive
- 3 True Negative
- 3 False positive
- 0 False Negative
- Precision: 12/15
- Recall = 12/12



Conclusions

- Models are similarly, but Decision Tree is better.
- Launch Sites increases over time
- Orbits like ES L1, GEO, HEO have a 100% success rate
- When Payload Mass increases the success rate increases too.
- Is recommendable has a dataset with large records
- And also is recommendable try with another classification models

Thank you!