IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Name: Ken Kaya
Date: JUL/2025

Outline

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

Executive Summary

The mission objective of this project is to analyse SpaceX launch data to discover insights and build models predicting a successful landing.

A summary of the methodologies and results described in this report is outlined as follows:

Summary of methodologies:

- Data Collection through API and Web Scraping
- Data Wrangling
- Exploratory Data Analysis with Data Visualization and SQL
- Building an interactive map with Folium
- Building Dashboard with Plotly Dash
- Predictive analysis (classification)

Summary of all results:

- Exploratory Data Analysis results
- Interactive Analytics Dashboard
- Predictive Analysis Result

Introduction

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.

The problems in finding the answer:

- Which launch sites perform best?
- Which orbit and payload combinations are most successful?
- Can we predict successful landings?

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - SpaceX Launch API, and Web scraping Wikipedia for additional metadata.
- Perform data wrangling
  - Handling missing values and column renaming.
  - Merging multiple datasets into one DataFrame.
  - Feature engineering, such as orbit type, booster version, and outcome mapping.
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - How to build, tune, and evaluate classification models

Data Collection

- Describe how data sets were collected.
  - Data collection was done through SpaceX Launch API
  - External CSVs for booster info and payload data.
  - Data stored and preprocessed using Pandas.
- You need to present your data collection process use key phrases and flowcharts

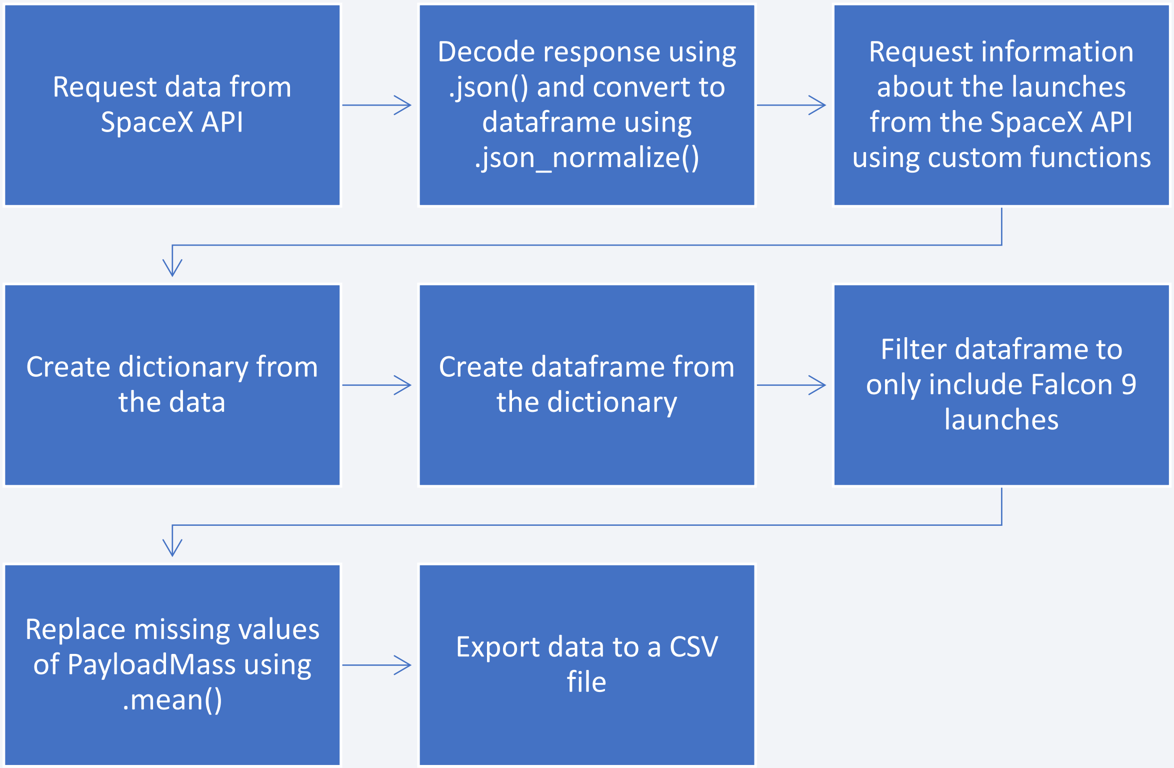
Task 1
- Request and parse the SpaceX launch data using the GET request

Task 2
- Filter the dataframe to only include “Falcon 9” launches

Task 3
- Dealing with Missing Values

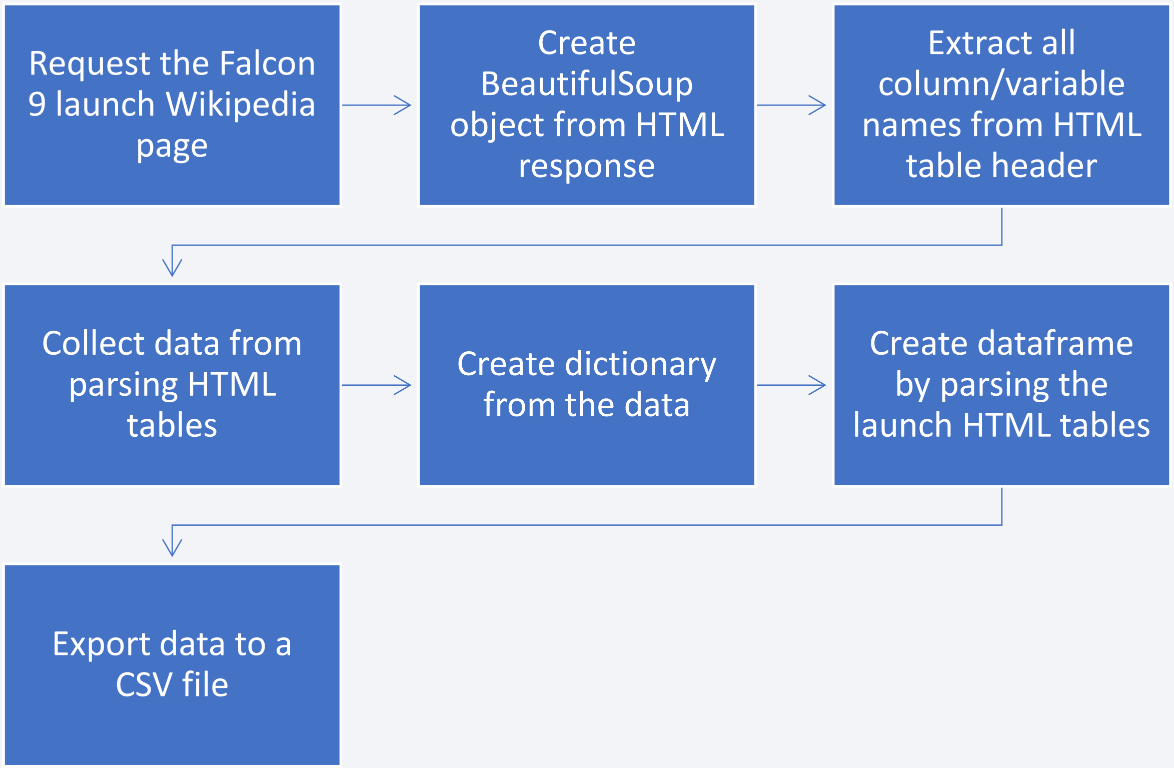
Data Collection – SpaceX API

- Request data from SpaceX API
- Decode response using .json() and convert to dataframe using .json_normalize()
- Request information about the launches from the SpaceX API using custom functions
- Create dictionary from the data
- Create dataframe from the dictionary
- Filter dataframe to only include Falcon 9 launches
- Replace missing values of PayloadMass using .mean()
- Export data to a CSV file
- GitHub URL of the completed SpaceX API calls notebook https://github.com/kayaken/IBM-Data-Science-Capstone/blob/main/1.%20Data%20Collection_API.ipynb



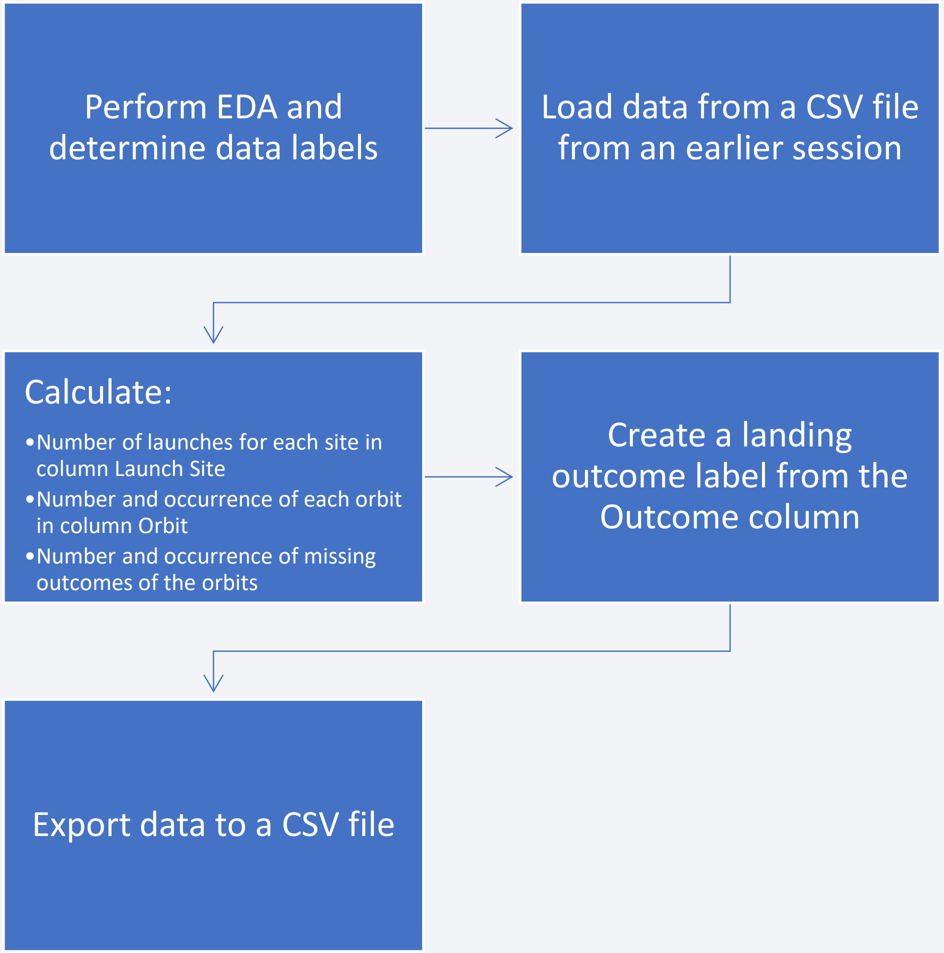
Data Collection - Scraping

- Request the Falcon 9 launch Wikipedia page
- Create BeautifulSoup object from HTML response
- Extract all column/variable names from HTML table header
- Collect data from parsing HTML tables
- Create dictionary from the data
- Create dataframe by parsing the launch HTML tables
- Export data to a CSV file
- GitHub URL of the completed web scraping notebook https://github.com/kayaken/IBM-Data-Science-Capstone/blob/main/2.%20WebScrape.ipynb



Data Wrangling

- Perform EDA and determine data labels
- Load data from a CSV file from an earlier session
- Calculate:
  - Number of launches for each site in column Launch Site
  - Number and occurrence of each orbit in column Orbit
  - Number and occurrence of missing outcomes of the orbits
- Create a landing outcome label from the Outcome column
- Export data to a CSV file
- GitHub URL of completed data wrangling-related notebooks
  https://github.com/kayaken/IBM-Data-Science-Capstone/blob/main/3.%20Data_Wrangling.ipynb



EDA with Data Visualization

- Cat Plot chart: To compare different categories or groups, often across one or more variables.
  - For “Payload Mass vs. Flight Number”
  - For “Flight Number vs. Launch Site”
- Bar Plot chart: To show and compare values across discrete categories.
  - For “Success Rate of each Orbit Type”
- Scatter Plot chart: To show relationships or correlations between two continuous variables.
  - For “Payload Mass vs. Launch Site”
  - For “Flight Number vs. Orbit Type”
  - For “Payload Mass vs. Orbit Type”
- Line Plot chart: To track changes or trends over time.
  - For “Yearly Launch Success Trend”
- GitHub URL of completed EDA with data visualization notebook, https://github.com/kayaken/IBM-Data-Science-Capstone/blob/main/5.%20EDA_with_DataViz.ipynb

EDA with SQL

- Display:
  - The names of the unique launch sites
  - 5 records where launch sites begin with the string 'CCA’
  - The total payload mass carried by boosters launched by NASA (CRS)
  - The average payload mass carried by booster version F9 v1.1
- List:
  - The date when the first successful landing outcome on the ground pad was achieved.
  - The names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000.
  - The total number of successful and failure mission outcomes.
  - All the booster_versions that have carried the maximum payload mass, using a subquery with a suitable aggregate function.
  - The records display the month names, failure landing_outcomes in drone ship ,booster versions, launch_site for the months in year 2015.
- 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.
- GitHub URL of completed EDA with SQL notebook https://github.com/kayaken/IBM-Data-Science-Capstone/blob/main/4.%20EDA_with_SQL.ipynb

Build an Interactive Map with Folium

- Launch site geolocations:
  - Added **blue** circle at NASA Jonhson Space Center's coordinate with a popup label showing its name using its latitude and longitude coordinates.
  - Added **red** circle at all launch sites coordinates with a popup label showing its name using its latitude and longitude coordinates.

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

- Proximity to infrastructure: roads, rails, coasts:
  - Added colored lines to show the distance between the launch site CCAFS SLC-40 and its proximity to the nearest coastline, railway, highway, and city

- GitHub URL of completed interactive map with Folium map https://github.com/kayaken/IBM-Data-Science-Capstone/blob/main/6.%20Visual_Analytics_with_Folium.ipynb

Build a Dashboard with Plotly Dash

- Dropdown List with Launch Sites:
  - Allow the user to select all launch sites or a certain launch site.

- Launch success Pie charts by site:
  - Allow the user to see successful and unsuccessful launches as a percentage of the total.

- Payload vs. outcome Scatter plot:
  - Allow the user to select payload mass range.

- Booster-specific filters and sliders:
  - Allow the user to see the correction between Payload and Launch Success

- GitHub URL of completed Plotly Dash lab https://github.com/kayaken/IBM-Data-Science-Capstone/blob/main/7.%20spacex-dash-app.py

Predictive Analysis (Classification)

- Data Preparation: Cleaned and split the data into training and test sets.
- Model Building and Turning: Build Logistic Regression, SVM, Decision Tree, and KNN models.
- Hyperparameter Tuning: Used GridSearchCV with 10-fold cross-validation.
- Model Evaluation: Measured test accuracy and plotted confusion matrices.
- Model Selection: Chose the best-performing model based on the highest test accuracy.

Data Preprocessing → Train-Test Split → Model Training & GridSearchCV → Test Accuracy & Confusion Matrix → Compare & Select Best Model

- GitHub URL of completed predictive analysis lab https://github.com/kayaken/IBM-Data-Science-Capstone/blob/main/8.%20Machine%20Learning_Prediction.ipynb

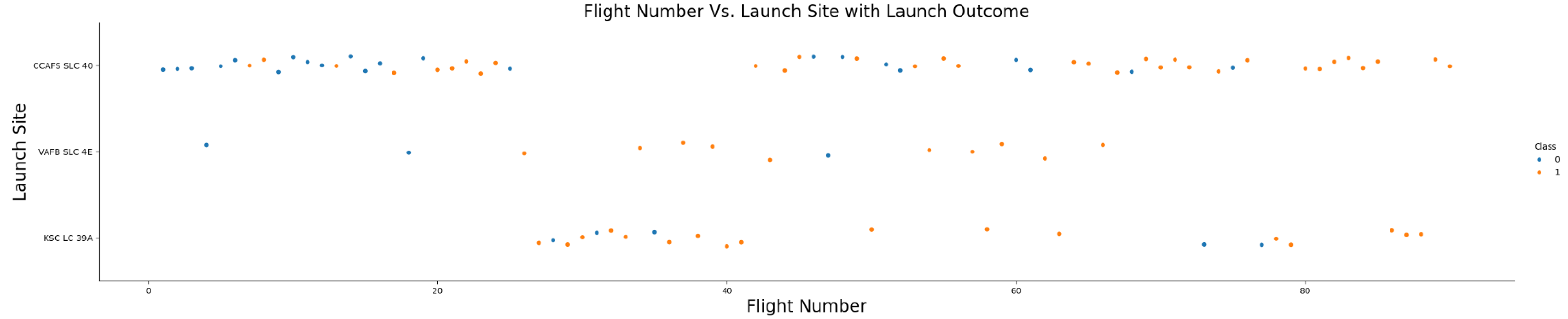
Results

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

Section 2

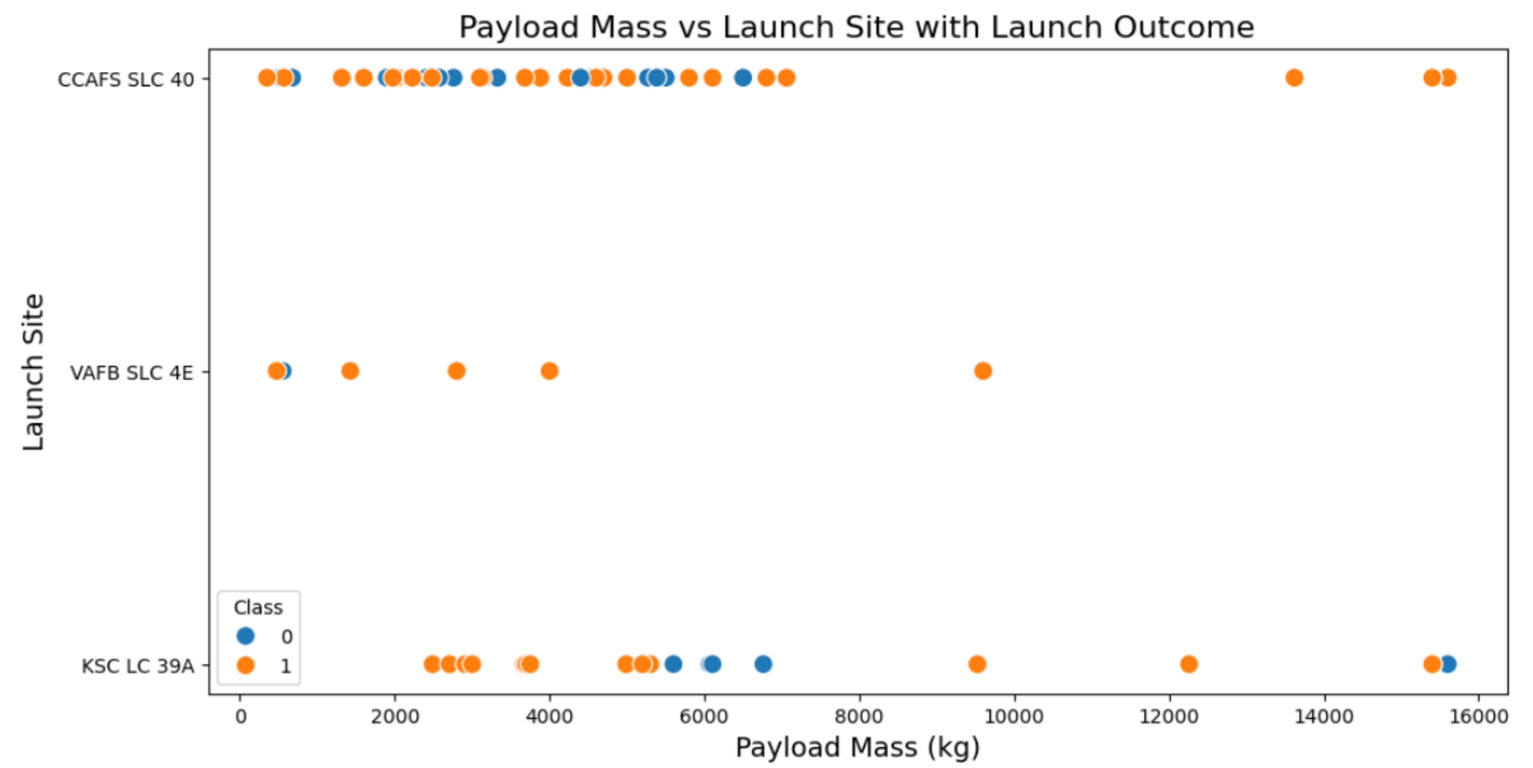
Insights drawn from EDA

Flight Number vs. Launch Site



The greater the number of flights at a launch site, the greater the success rate at a launch site.

Payload vs. Launch Site



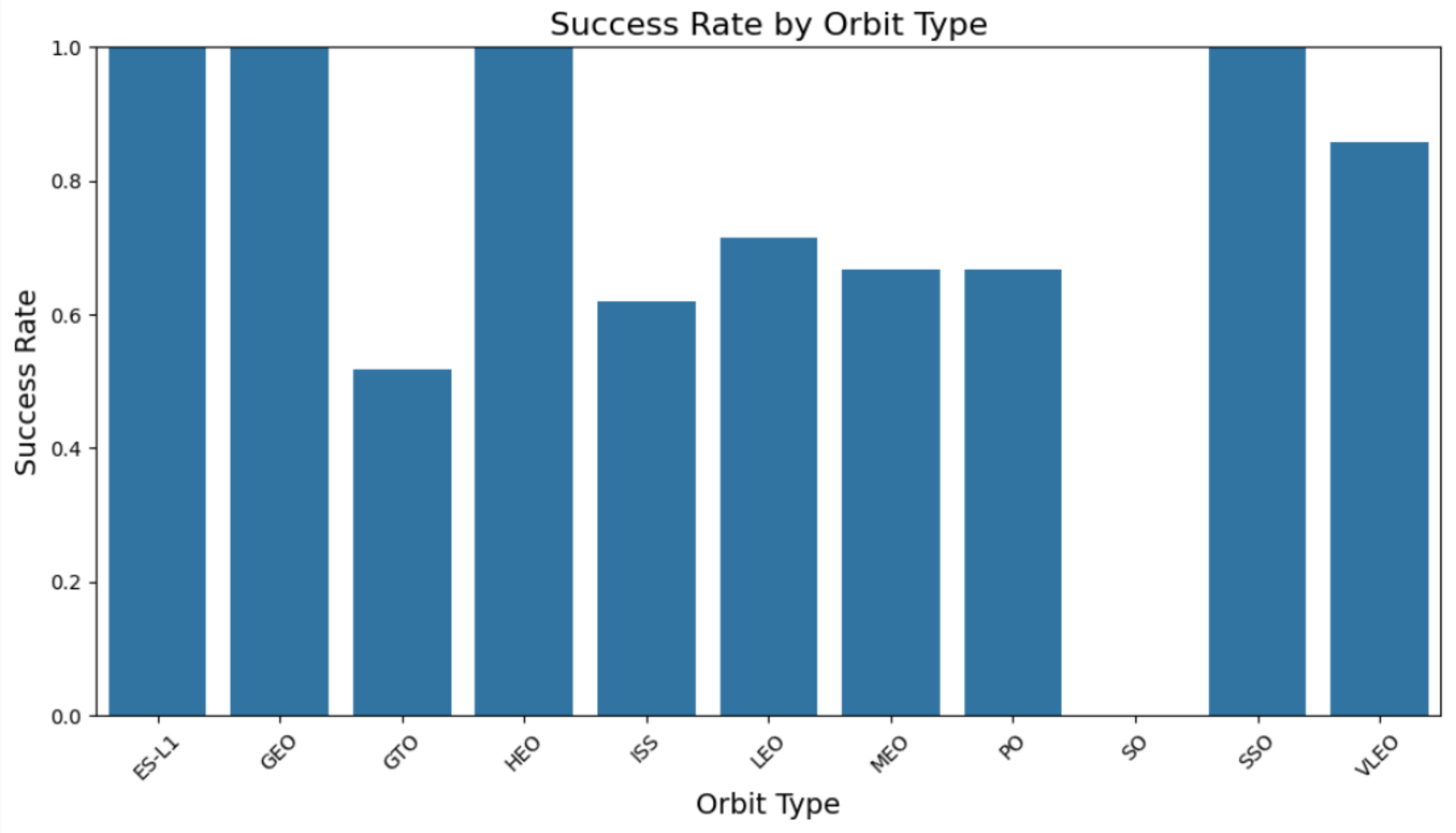
Key Findings:

- Payload 4,000 - 6,000kg is most successful.
- Site CCAFS SLC 40 had the best performance.

Recommendations:

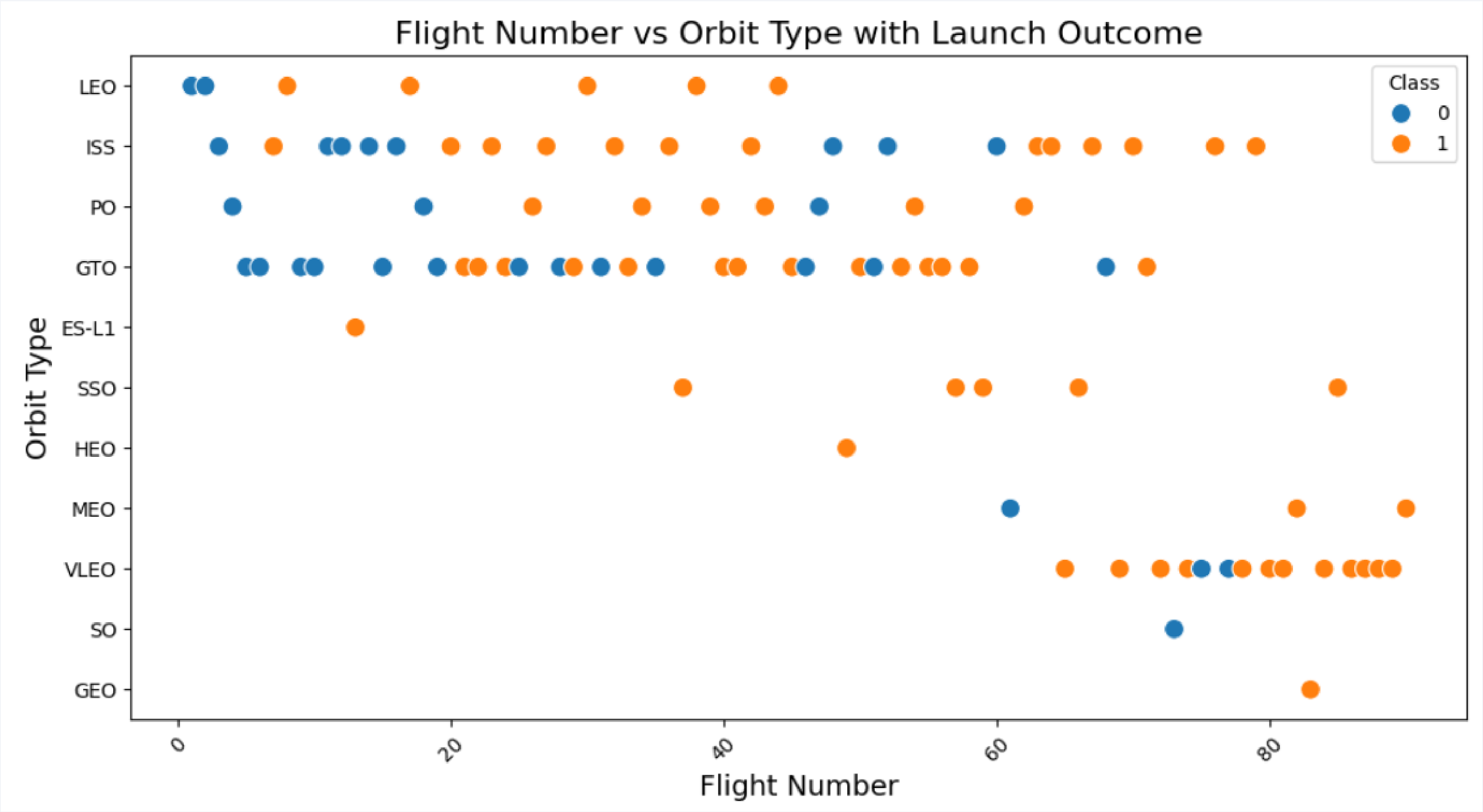
- Focus launches on high-performing sites.

Success Rate vs. Orbit Type



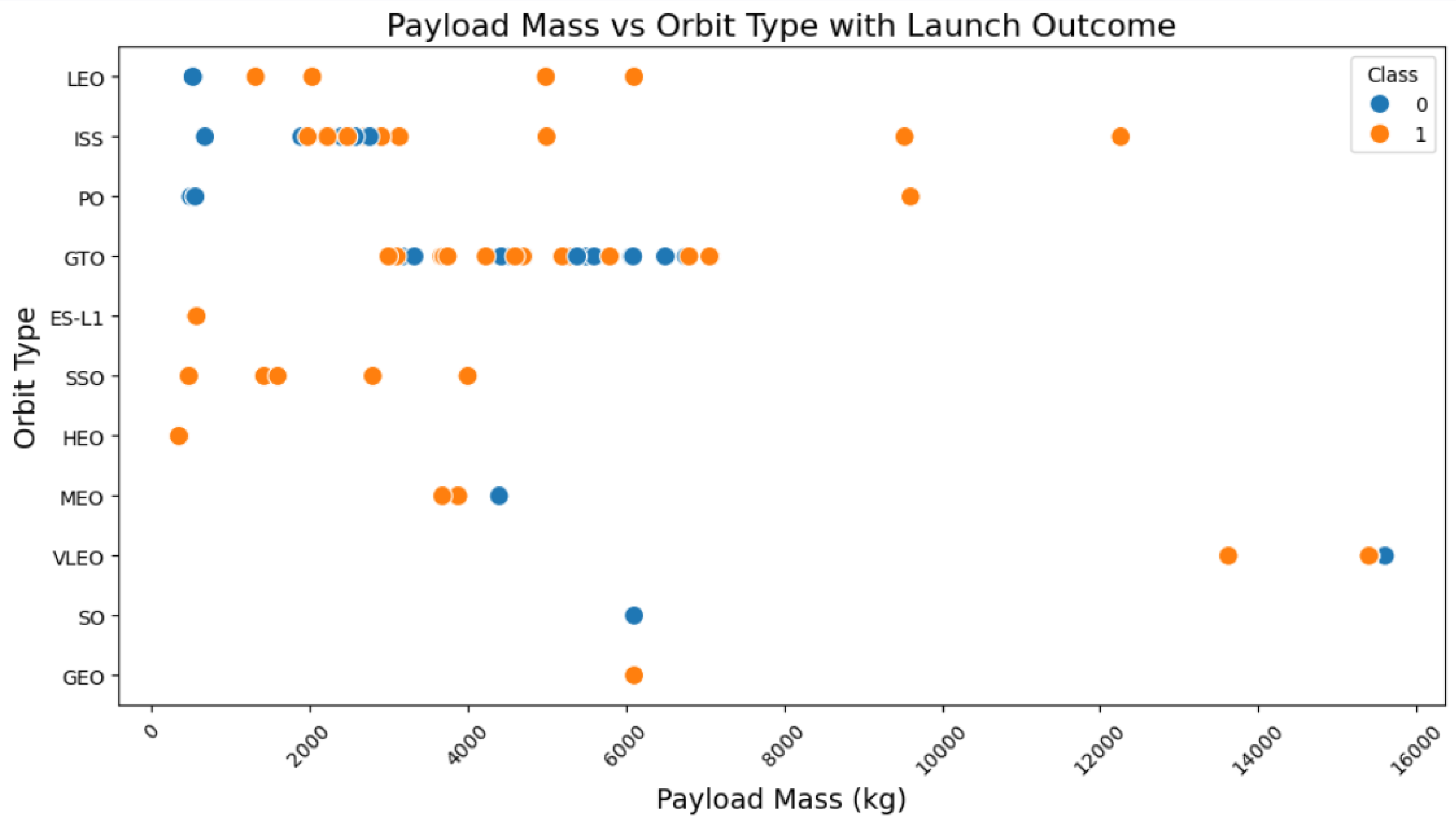
- **100% Success Rate:** GEO, HEO, SSO, ES-L1
- **50% - 90% Moderate Success Rate:** GTO, ISS, LEO, MEO, PO, VLEO
- **0% Success Rate:** SO

Flight Number vs. Orbit Type



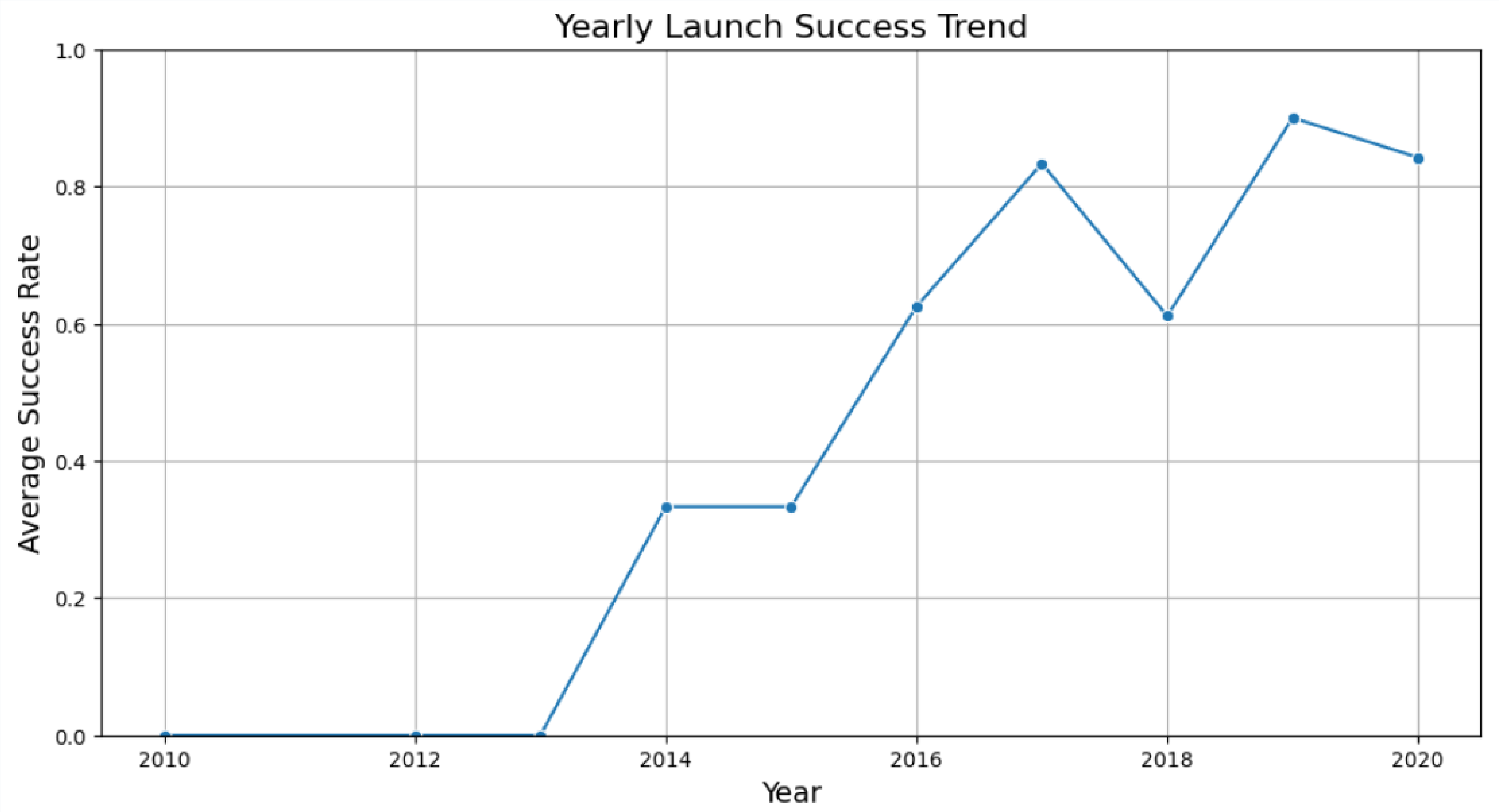
LEO orbit, success seems to be related to the number of flights. Conversely, in the **GTO** orbit, there appears to be no relationship between flight number and success.

Payload vs. Orbit Type



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

Launch Success Yearly Trend



Based on the observation that the launch success rate since 2013 has kept increasing till 2020.

All Launch Site Names

```
%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

To list all the **unique launch sites** found in the **SPACEXTABLE**, with no duplicates

Launch Site Names Begin with 'CCA'

```
%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

To retrieve the **first 5 records** from **SPACEXTABLE** where the launch site name **starts with "CCA"**.

Total Payload Mass

```
%sql select sum(PAYLOAD_MASS__KG_) from SPACEXTABLE where customer = 'NASA (CRS)'
```

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

sum(PAYLOAD_MASS__KG_)
45596

To calculate the total payload mass (in kilograms) for all launches where the customer is **"NASA (CRS)"**, using data from the **SPACEXTABLE**.

Average Payload Mass by F9 v1.1

```
%sql select avg(PAYLOAD_MASS__KG_) from SPACEXTABLE where Booster_Version='F9 v1.1'
```

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

avg(PAYLOAD_MASS__KG_)
2928.4

To calculate the **average payload mass (in kg)** for launches that used the **'F9 v1.1'** booster version, based on data from the **SPACEXTABLE**.

First Successful Ground Landing Date

```
%sql select min(date) from SPACEXTABLE where landing_outcome='Success (ground pad)'
```

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

min(date)
2015-12-22

To find the **first or earliest launch date** when the landing outcome was a **successful landing on a ground pad**, using data from the **SPACEXTABLE**.

Successful Drone Ship Landing with Payload between 4000 and 6000

```
%sql select * from SPACEXTABLE where landing_outcome='Success (drone ship)' and PAYLOAD_MASS__KG_ > 4000 and PAYLOAD_MASS__KG_ < 6000
```

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

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
2016-05-06	5:21:00	F9 FT B1022	CCAFS LC-40	JCSAT-14	4696	GTO	SKY Perfect JSAT Group	Success	Success (drone ship)
2016-08-14	5:26:00	F9 FT B1026	CCAFS LC-40	JCSAT-16	4600	GTO	SKY Perfect JSAT Group	Success	Success (drone ship)
2017-03-30	22:27:00	F9 FT B1021.2	KSC LC-39A	SES-10	5300	GTO	SES	Success	Success (drone ship)
2017-10-11	22:53:00	F9 FT B1031.2	KSC LC-39A	SES-11 / EchoStar 105	5200	GTO	SES EchoStar	Success	Success (drone ship)

To retrieve **all records** from **SPACEXTABLE** where the **landing was successful** on a **drone ship**, and the **payload mass** is **between 4000 and 6000 kilograms**.

Total Number of Successful and Failure Mission Outcomes

```
%sql select mission_outcome, count(*) as 'count' from SPACEXTABLE group by mission_outcome
```

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

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

To count how many times each **mission outcome** occurred in the **SPACEXTABLE**, and show the **number of launches** for each outcome.

Boosters Carried Maximum Payload

```
%sql select * from SPACEXTABLE where PAYLOAD_MASS__KG_ in (select max(PAYLOAD_MASS__KG_) from SPACEXTABLE)
```

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

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
2019-11-11	14:56:00	F9 B5 B1048.4	CCAFS SLC-40	Starlink 1 v1.0, SpaceX CRS-19	15600	LEO	SpaceX	Success	Success
2020-01-07	2:33:00	F9 B5 B1049.4	CCAFS SLC-40	Starlink 2 v1.0, Crew Dragon in-flight abort test	15600	LEO	SpaceX	Success	Success
2020-01-29	14:07:00	F9 B5 B1051.3	CCAFS SLC-40	Starlink 3 v1.0, Starlink 4 v1.0	15600	LEO	SpaceX	Success	Success
2020-02-17	15:05:00	F9 B5 B1056.4	CCAFS SLC-40	Starlink 4 v1.0, SpaceX CRS-20	15600	LEO	SpaceX	Success	Failure
2020-03-18	12:16:00	F9 B5 B1048.5	KSC LC-39A	Starlink 5 v1.0, Starlink 6 v1.0	15600	LEO	SpaceX	Success	Failure
2020-04-22	19:30:00	F9 B5 B1051.4	KSC LC-39A	Starlink 6 v1.0, Crew Dragon Demo-2	15600	LEO	SpaceX	Success	Success
2020-06-04	1:25:00	F9 B5 B1049.5	CCAFS SLC-40	Starlink 7 v1.0, Starlink 8 v1.0	15600	LEO	SpaceX, Planet Labs	Success	Success
2020-09-03	12:46:14	F9 B5 B1060.2	KSC LC-39A	Starlink 11 v1.0, Starlink 12 v1.0	15600	LEO	SpaceX	Success	Success
2020-10-06	11:29:34	F9 B5 B1058.3	KSC LC-39A	Starlink 12 v1.0, Starlink 13 v1.0	15600	LEO	SpaceX	Success	Success
2020-10-18	12:25:57	F9 B5 B1051.6	KSC LC-39A	Starlink 13 v1.0, Starlink 14 v1.0	15600	LEO	SpaceX	Success	Success
2020-10-24	15:31:34	F9 B5 B1060.3	CCAFS SLC-40	Starlink 14 v1.0, GPS III-04	15600	LEO	SpaceX	Success	Success
2020-11-25	2:13:00	F9 B5 B1049.7	CCAFS SLC-40	Starlink 15 v1.0, SpaceX CRS-21	15600	LEO	SpaceX	Success	Success

To retrieve the **full record(s)** from **SPACEXTABLE** where the **payload mass is the highest** among all launches.

2015 Launch Records

```
%sql select substr(Date, 6, 2), landing_outcome, booster_version, launch_site from SPACEXTABLE where landing_outcome = 'Failure (drone ship)' and substr(Date, 0, 5)='2015'
```

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

substr(Date, 6, 2)	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

To get the **month** (from the Date field), along with **landing outcome**, **booster version**, and **launch site** for launches in the **year 2015** that had a **failed landing on a drone ship**.

- SUBSTR(Date, 6, 2) extracts the **month** from the Date (assuming Date is in YYYY-MM-DD format).
- SUBSTR(Date,_0, 5) = '2015' filters only the records from the **year 2015**.

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

```
%sql select landing_outcome, count(landing_outcome) as 'landing_outcomes' from SPACEXTABLE where date between '2010-06-04' and '2017-03-20' group by landing_outcome order by 2 desc
```

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

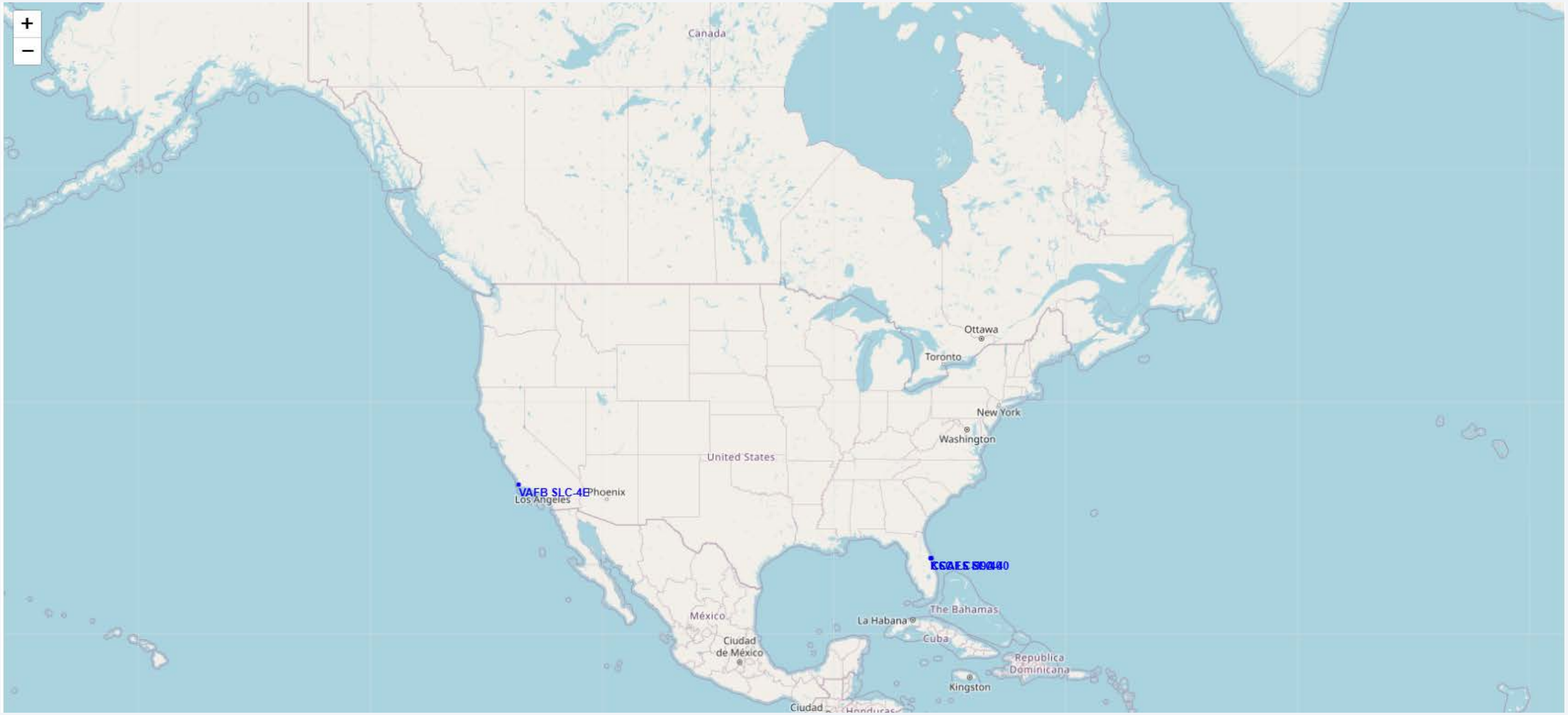
Landing_Outcome	landing_outcomes
No attempt	10
Success (drone ship)	5
Failure (drone ship)	5
Success (ground pad)	3
Controlled (ocean)	3
Uncontrolled (ocean)	2
Failure (parachute)	2
Precluded (drone ship)	1

To count how many times each **landing outcome** occurred **between June 4, 2010, and March 20, 2017**, and list them in **descending order of frequency**.

Section 3

Launch Sites Proximities Analysis

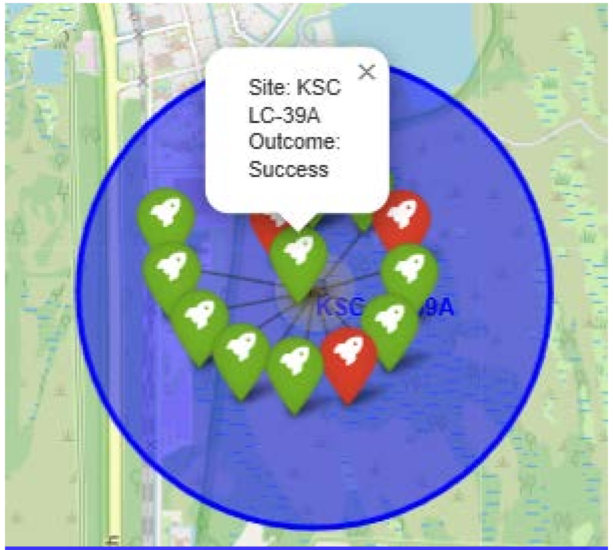
Launch Sites Global Map



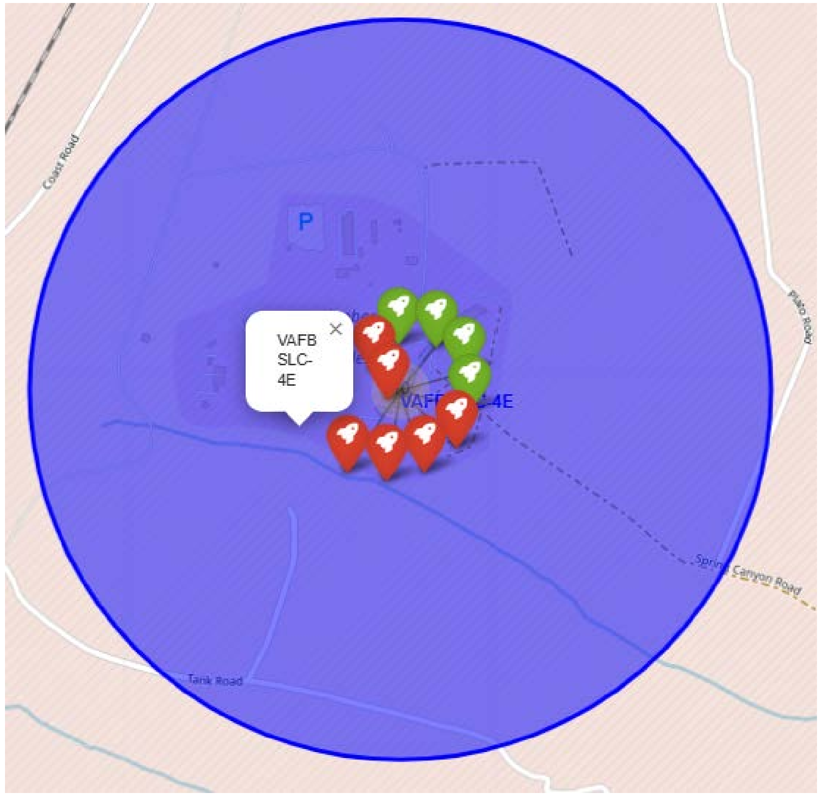
Geographic map showing SpaceX launch sites (blue text labels) in North America, which are:

- East Coast - Florida (CCAFS SLC-40 and KSC LC-39A)
- West Coast - California (VAFB SLC-4E)

Launch Sites Outcome with Markers Color Labels



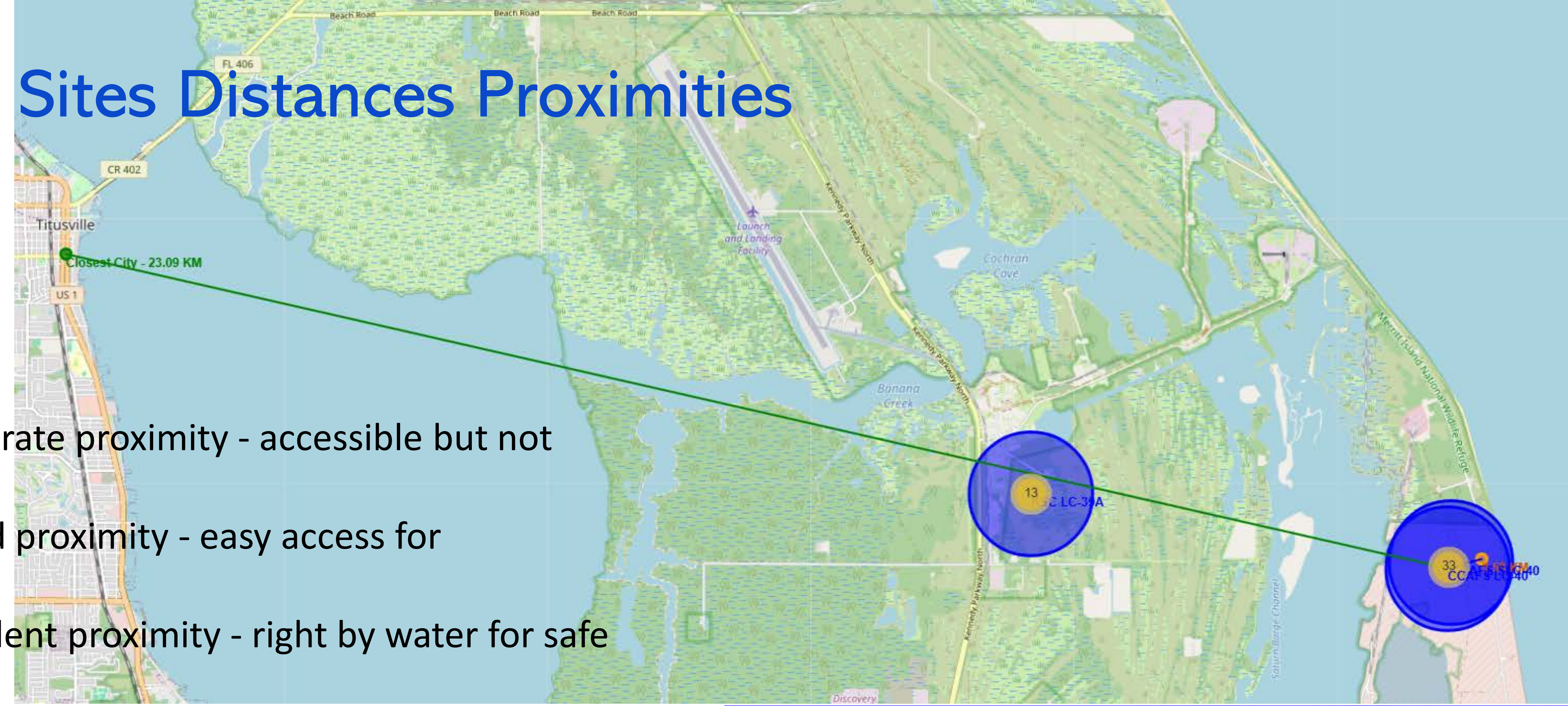
East Coast – Florida launch Sites (CCAFS SLC-40 and KSC LC-39A)



West Coast - California launch site (VAFB SLC-4E)

Green markers for Successful and Red markers for Unsuccessful launches

Launch Sites Distances Proximities

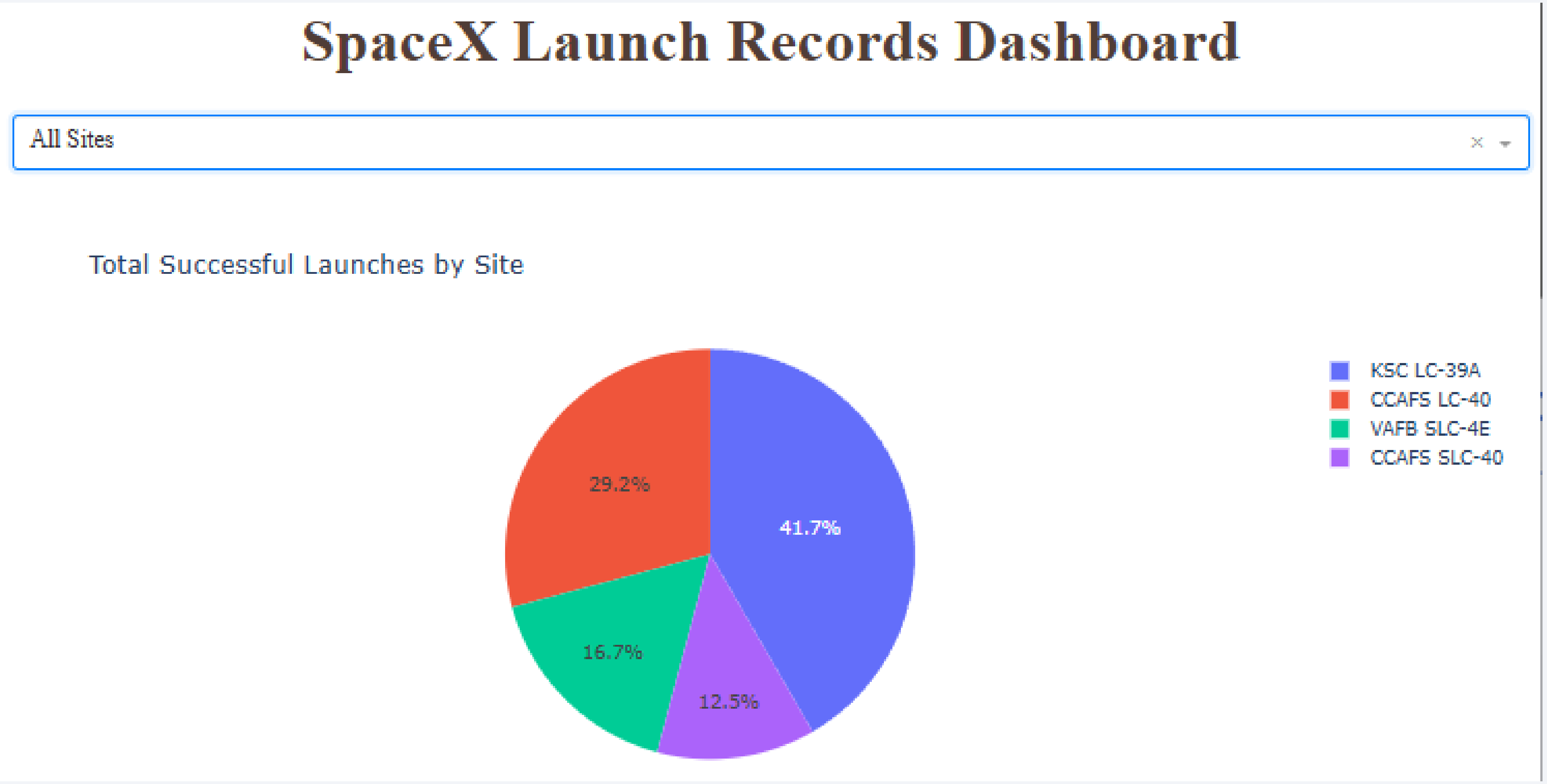


- **Railways:** Moderate proximity - accessible but not too close.
- **Highways:** Good proximity - easy access for transportation.
- **Coastline:** Excellent proximity - right by water for safe launches.
- **Cities:** Safe distance - far from populated areas like Titusville.
- **Summary:** Launch sites are strategically placed near transportation and coastline for operational needs, but away from cities for safety. A Perfect balance of access and protection.

Section 4

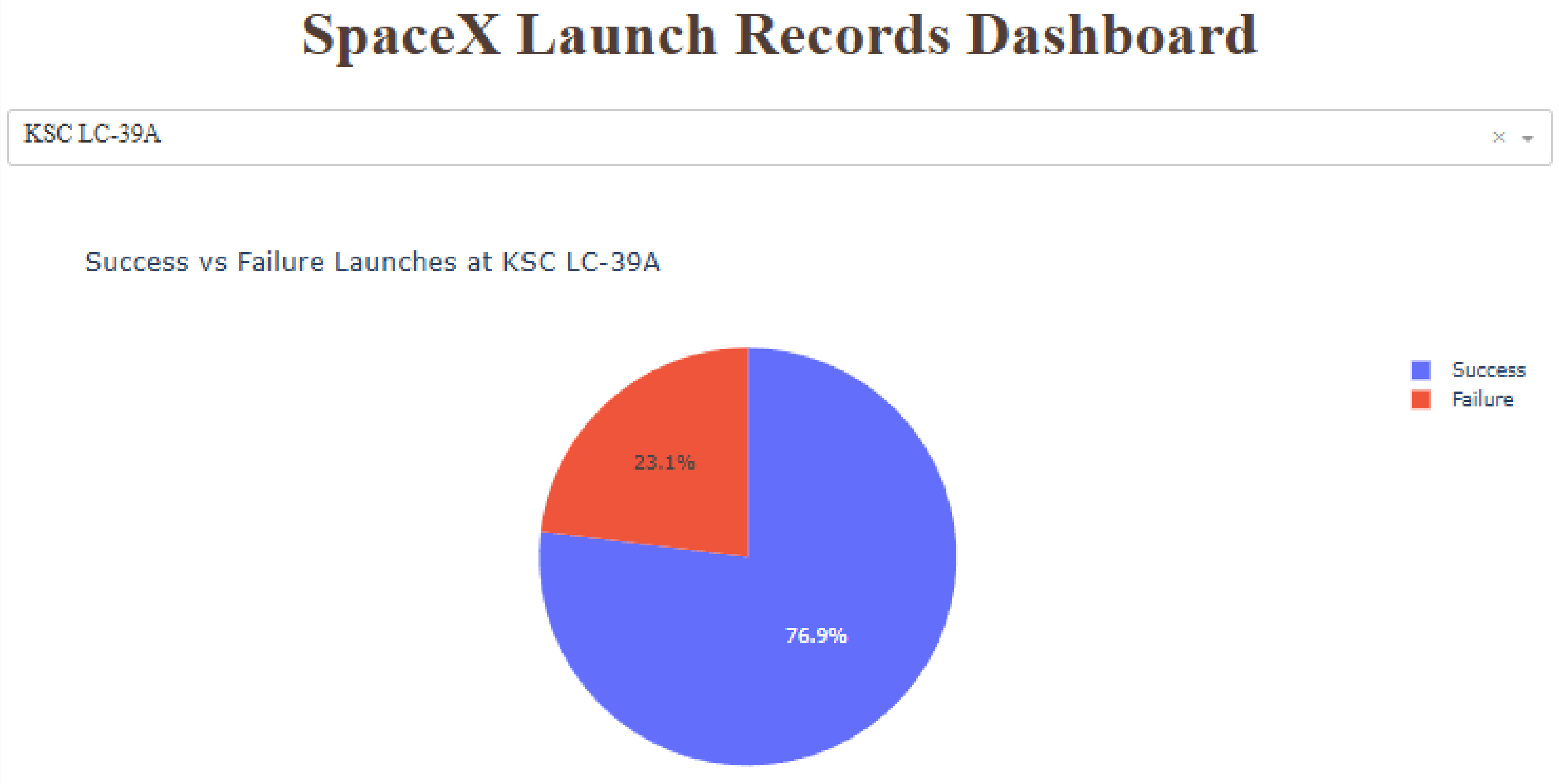
Build a Dashboard with Plotly Dash

Total Successful Launches by Site



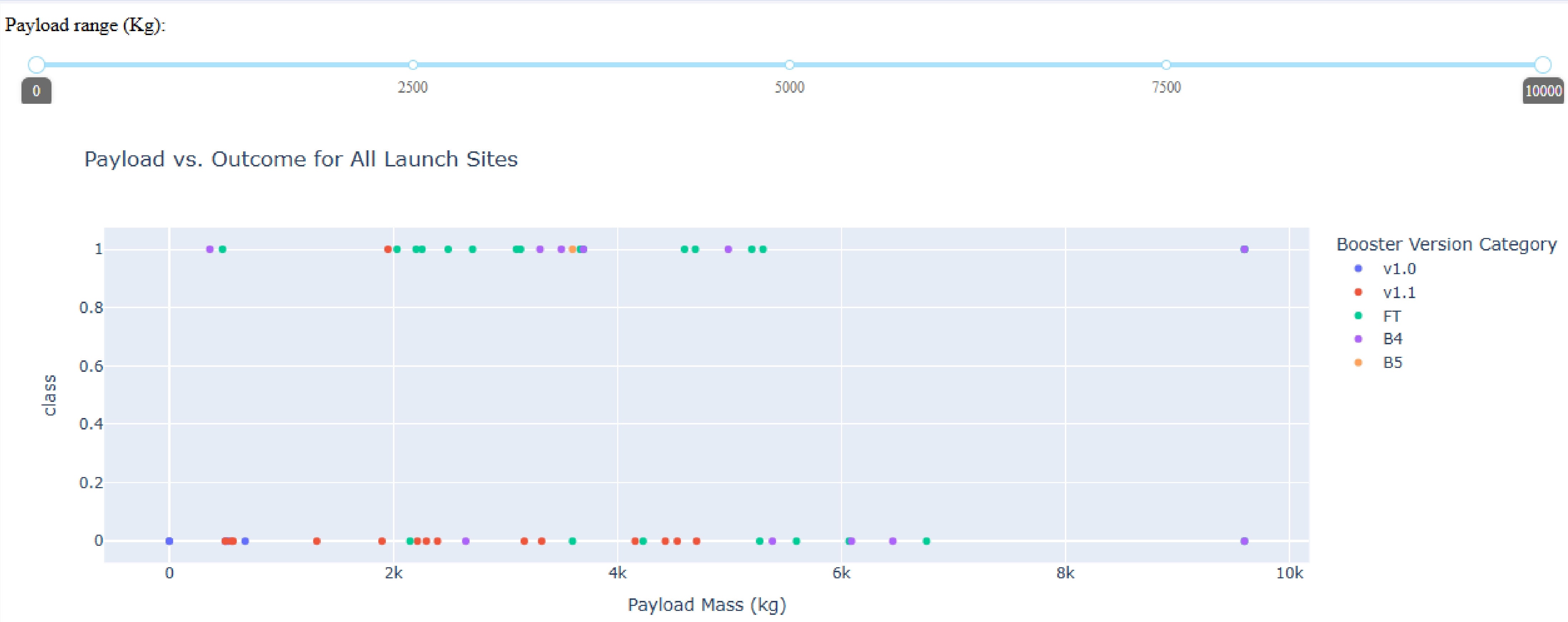
KSC LC-39A had the **most successful launches** (41.8%) among the launch sites.

Launch Sites with the Highest Launch Success Ratio



KSC LC-39A achieved a 76.9% success rate, while getting a 23.1% failure rate.

Payload Mass (kg) vs. Outcome for all Launch Sites



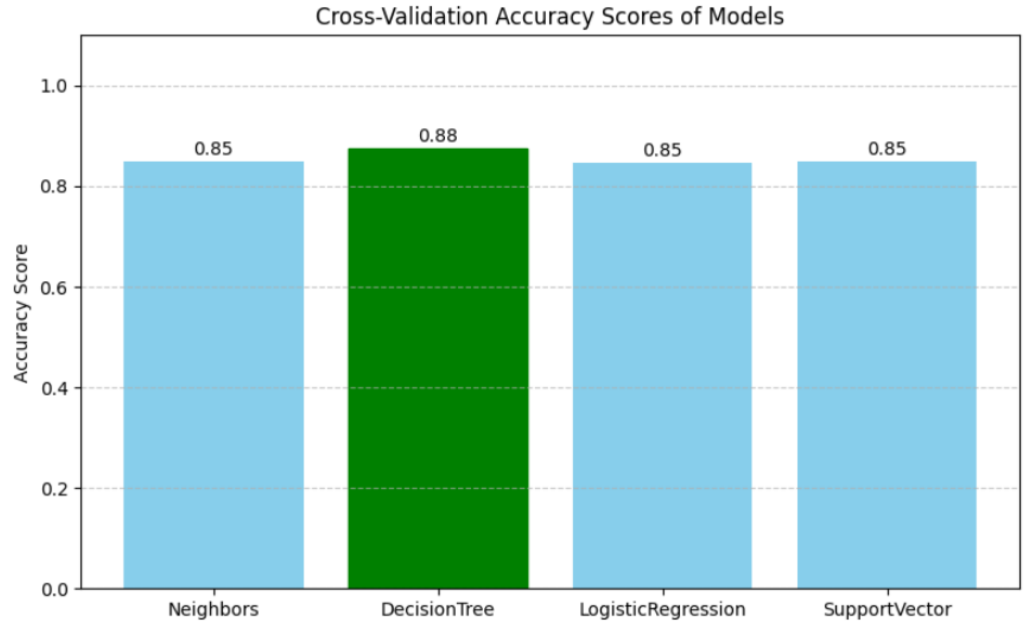
- Most Successful Payload Range: Between **2,000** - **6,000** kg, especially with **FT** and **B5** boosters.
- Best Booster Versions: **FT** and **B5** - lots of successes, very few or no failures.
- Older Boosters (**v1.0**, **v1.1**): More failures, mostly with lower payloads **(<4,000 kg**).
- Highest Payload (**~10,000 kg**): Rare but possible with **B4** and can still succeed.

Section 5

Predictive Analysis (Classification)

Classification Accuracy

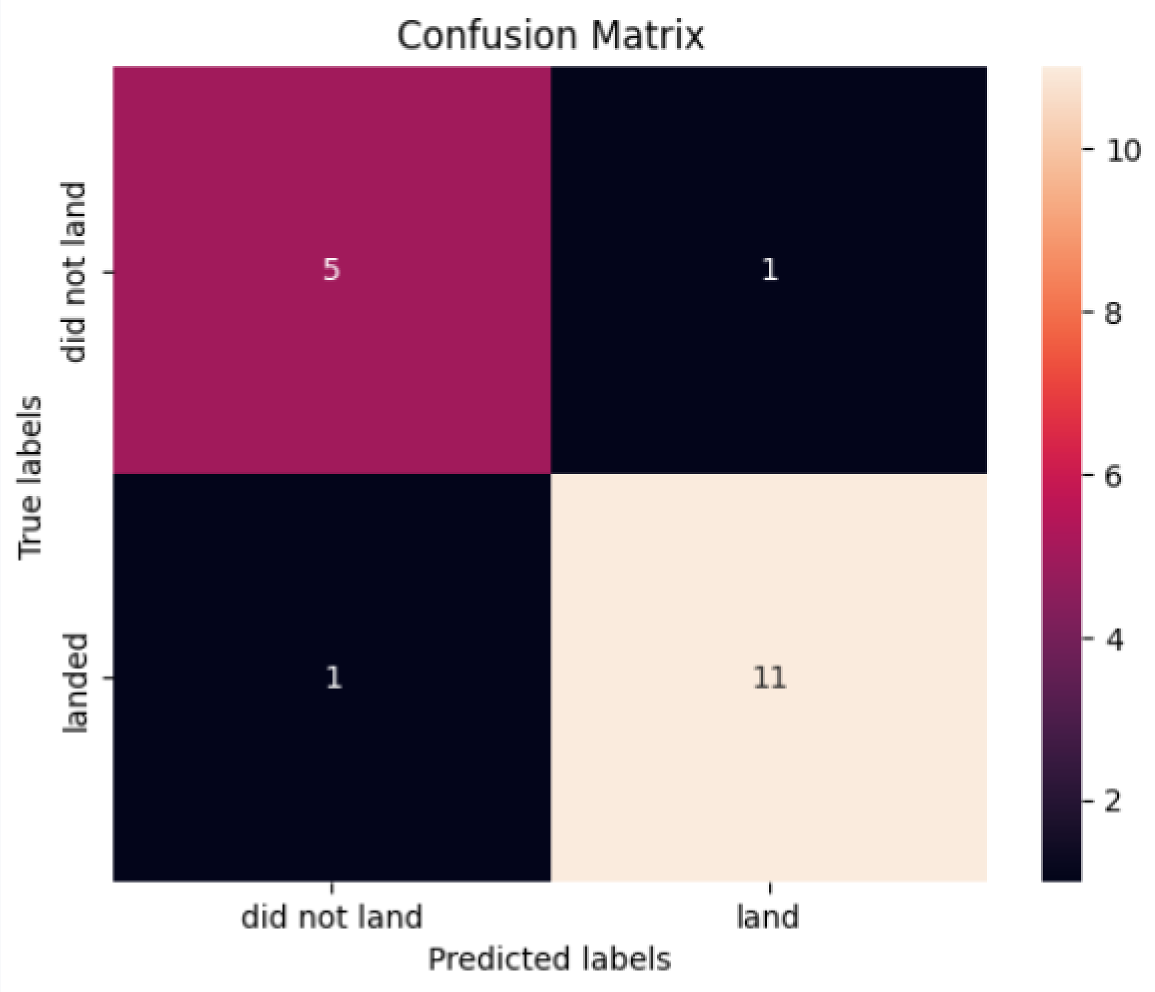
```
models = {'Neighbors': knn_cv.best_score_,
          'DecisionTree': tree_cv.best_score_,
          'LogisticRegression': logreg_cv.best_score_,
          'SupportVector': svm_cv.best_score_}

bestalgorithm = max(models, key=models.get)
print('Best algorithm is', bestalgorithm, 'with a score of', models[bestalgorithm])
if bestalgorithm == 'DecisionTree':
    print('Best params is :', tree_cv.best_params_)
if bestalgorithm == 'Neighbors':
    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 algorithm is DecisionTree with a score of 0.875
Best params is : {'criterion': 'entropy', 'max_depth': 4, 'max_features': 'sqrt', 'min_samples_leaf': 2, 'min_samples_split': 10, 'splitter': 'best'}
```

The **Decision Tree** is the model that has the **highest classification accuracy 88%**.

Confusion Matrix



Results:

- True Negatives (TN): 5 (correct "did not land")
- False Positives (FP): 1 (wrong "land" prediction)
- False Negatives (FN): 1 (missed "land" prediction)
- True Positives (TP): 11 (correct "land")

Performance:

- Accuracy: 88% (16/18 correct predictions)
- Precision: 91% (11/12 "land" predictions were right)
- Recall: 91% (11/12 actual landings detected)

Summary:

- Excellent performance with only 2 errors out of 18 predictions. The model is particularly strong at identifying successful landings.

Conclusions

Key Findings:

- The greater the number of flights at a launch site, the greater the success rate at a launch site.
- Payload **4,000** - **6,000kg** is most successful.
- Site **CCAFS SLC 40** had the best performance.
- 100% Success Rate: **GEO, HEO, SSO, ES-L1**
- With heavy payloads, the successful landing or positive landing rate is higher for **Polar**, **LEO** and **ISS**. However, for **GTO**, it's difficult to distinguish between successful and unsuccessful landings as both outcomes are present.
- Launch success rate started to increase in **2013** till **2020**.
- **Coastline**: Excellent proximity - right by water for safe launches
- **KSC LC-39A** had the most successful launches (41.8%) among the launch sites.
- The **Decision Tree** is the model that has the **highest classification accuracy (88.9%).**

Appendix

SQL Queries:

Task 1: Display the names of the unique launch sites in the space mission

- %sql select distinct(launch_Site) from SPACEXTABLE

Task 2: Display 5 records where launch sites begin with the string 'CCA'

- %sql select * from SPACEXTABLE where Launch_Site like 'CCA%' limit 5

Task 3: Display the total payload mass carried by boosters launched by NASA (CRS)

- %sql select sum(PAYLOAD_MASS__KG_) from SPACEXTABLE where customer = 'NASA (CRS)’

Task 4: Display average payload mass carried by booster version F9 v1.1

- %sql select avg(PAYLOAD_MASS__KG_) from SPACEXTABLE where Booster_Version='F9 v1.1’

Task 5: List the date when the first succesful landing outcome in ground pad was acheived.

- %sql select min(date) from SPACEXTABLE where landing_outcome='Success (ground pad)’

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

- %sql select * from SPACEXTABLE where landing_outcome='Success (drone ship)' and PAYLOAD_MASS__KG_ > 4000 and PAYLOAD_MASS__KG_ < 6000

Task 7: List the total number of successful and failure mission outcomes

- %sql select mission_outcome, count(*) as 'count' from SPACEXTABLE group by mission_outcome

Task 8: List all the booster_versions that have carried the maximum payload mass, using a subquery with a suitable aggregate function.

- %sql select * from SPACEXTABLE where PAYLOAD_MASS__KG_ in (select max(PAYLOAD_MASS__KG_) from SPACEXTABLE)

Task 9: 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

- %sql select substr(Date, 6, 2), landing_outcome, booster_version, launch_site from SPACEXTABLE where landing_outcome = 'Failure (drone ship)' and substr(Date, 0, 5)='2015’

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.

- %sql select landing_outcome, count(landing_outcome) as 'landing_outcomes' from SPACEXTABLE where date between '2010-06-04' and '2017-03-20' group by landing_outcome order by 2 desc

Thank you!