Winning Space Race with Data Science

Danilov Aleksandr
22.11.2021

Outline

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

Executive Summary

Summary of methodologies

- Data collection
- Data wrangling
- EDA with data visualization
- Building an interactive map using Foilum
- Building a Dashboard with Plotly Dash
- Predictive analysis

Summary of all results

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

Introduction

- **Project background and context**

The Falcon 9 is a reusable, two-stage rocket designed and manufactured by SpaceX to safely and reliably transport people and goods into and out of Earth orbit. The Falcon 9 is the world's first reusable orbital-class rocket. The reusability allows SpaceX to retool the most expensive parts of the rocket, which in turn lowers the cost of accessing space. For example, a Falcon 9 rocket launch costs $ 62 million; rocket launches from other manufacturers cost more than $ 165 million each.

- **Problems you want to find answers**

➢What influences the landing of the first stage of a rocket and why?

➢How the successful return of the first stage depends on the parameters of the launch vehicle.

➢What conditions must be taken by SpaceX to minimize the risk of an unsuccessful landing of the first stage of the rocket?

Section 1

Methodology

Methodology

Executive Summary

- **Data collection methodology:**

➢ SpaceX Rest API

➢ Web scraping https://en.wikipedia.org/

- **Performed data wrangling:**

➢ One Hot Encoding data fields and dropping irrelevant columns

- **Performed exploratory data analysis (EDA) using visualization and SQL**
- **Performed interactive visual analytics using Folium and Plotly Dash**
- **Performed predictive analysis using classification models:**

➢ Algorithms K-Neighbors Classifier, SVC, Logistic Regression and Decision Tree Classifier were used. GridSearchCV from sklearn library was used to search for hyperparameters. The sklearn.metrics library was used to assess the accuracy of the models.

Data Collection

Data sets were collected:

- SpaceX REST API (URL: api.spacexdata.com/v4 / ...) - This API can provide us with all the data we need for further analysis.

- Another way to get Falcon 9 launch data is to flush Wikipedia with BeautifulSoup.

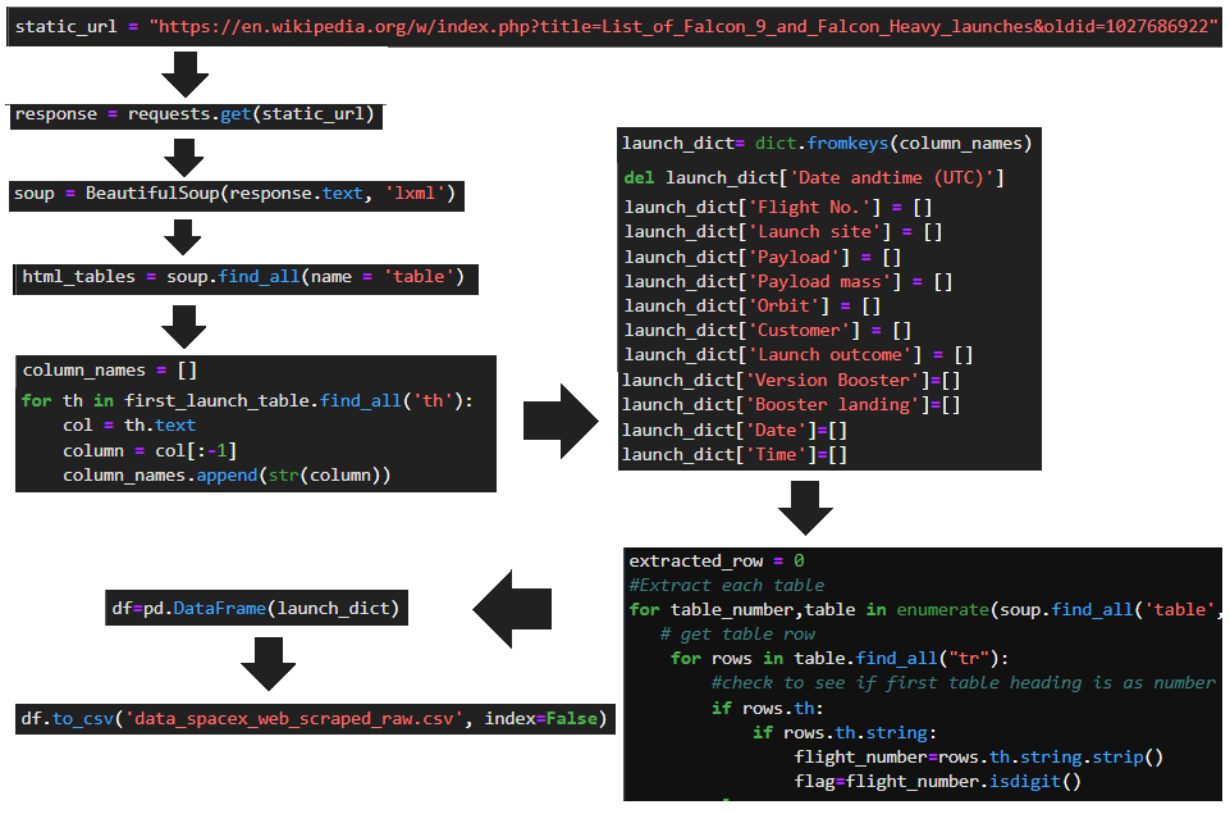
Data Collection – SpaceX API

- Getting response from API
- Converting to json and normalizing
- Apply custom functions to clean data
- Assign list to dictionary
- Assign dictionary to dataframe
- Filter the dataframe to only include Falcon 9 launches
- Export dataframe to csv file

GitHub

Data Collection - Scraping

- Getting response from wiki page
- Creating BS object
- Finding tables
- Getting column names
- Creating dictionary
- Filling dictionary with data
- Converting dictionary to dataframe
- Dataframe to .csv



GitHub

Data Wrangling

- Dealing with Missing Values
- Created a landing outcome label from Outcome column
- Features Engineering
- Dataframe to .csv

```
PayloadMass_mean = data_falcon9['PayloadMass'].mean()
# Replace the np.nan values with its mean value
data_falcon9['PayloadMass'].replace(to_replace=np.nan, value=PayloadMass_mean, inplace =True)
```

```
df_1 = pd.read_csv('data_dataset_falcon9_part_1.csv')
```

```
bad_outcomes=set(landing_outcomes.keys()[[1,3,5,6,7]])
landing_class = []
# landing_class = 0 if bad_outcome
# landing_class = 1 otherwise
for outcome in df['Outcome']:
    for bad in bad_outcomes:
        if outcome == bad:
            flag = 0
            break
        flag = 1
    landing_class.append(flag)

df['Class']=landing_class
```

```
df.to_csv("data_dataset_falcon9_part_2.csv", index=False)
```

```
features = df[['FlightNumber', 'PayloadMass', 'Orbit', 'LaunchSite', 'Flights', 'GridFins', 'Reused',
               'Legs', 'LandingPad', 'Block', 'ReusedCount', 'Serial']]
features_one_hot = pd.get_dummies(features, prefix=['Orbits', 'LaunchSite', 'LandingPad', 'Serial'])
features_one_hot.astype('float64')
```

```
features_one_hot.to_csv('data_dataset_falcon9_part_3.csv', index=False)
```

GitHub

EDA with Data Visualization

Scatter Graphs:

- Flight Number vs. Payload Mass
- Flight Number vs. Launch Site
- Payload vs. Launch Site
- Orbit vs. Flight Number
- Payload vs. Orbit Type
- Orbit vs. Payload Mass

Scatter plot shows how much one variable is affected by another.

Bar charts:

- Mean vs. Orbit

Bar chart makes it easy to compare sets of data between different groups.

Line graph:

- Success rate vs. Year

Line graphs show data variables and trends very clearly and can help to make predictions.

GitHub

EDA with SQL

Using SQL queries, the following questions were answered:

- Displaying the names of the unique launch sites in the space mission
- Displaying 5 records where launch sites begin with the string 'CCA'
- Displaying the total payload mass carried by boosters launched by NASA (CRS)
- Displaying average payload mass carried by booster version F9 v1.1
- Listing the date when the first successful landing outcome in ground pad was achieved.
- Listing the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000
- Listing the total number of successful and failure mission outcomes
- Listing the names of the booster versions which have carried the maximum payload mass
- Listing the failed landing outcomes in drone ship, their booster versions, and launch site names for in year 2015
- Ranking the count of landing outcomes between the date 2010-06-04 and 2017-03-20, in descending order

GitHub

Build an Interactive Map with Folium

- Using the circle object from the Folium library and the longitude and latitude from the dataset, the locations of the Falcon9 rocket launchers were mapped. Which allows you to quickly find the launch pads on the map.
- Added multi-colored in marker groups showing in which launches it was possible to successfully return the first stage of the carrier rocket, and which ones ended in failure. Colored markers in marker groups can easily identify which launch sites have relatively high success rates.
- Calculated the distance from the launch sites to the nearest cities, coastline, railway, and highway and plotted them on the map using the line object. This makes it possible to visually assess the logistical availability of a particular launch site.

GitHub

Build a Dashboard with Plotly Dash

- A pie chart and a scatter graph were used to create the dashboard. A Dropdown menu was used to control the pie chart, and RangeSlider was used to control the scatter graph.
- The pie chart shows the success of the return of the first stage of the launch vehicle relative to which platform the launch vehicle was launched from.
- The scatter graph shows the dependence of the return of the first stage of the launch vehicle relative to the Payload mass.

GitHub

Predictive Analysis (Classification)

Building model:

- Load data from .csv file to Pandas DataFrame
- Transform and Standardize data
- Splitting data on train and test set
- We train Logistic regression, SVM, decision tree, KNN based on train set

Model evaluating:

- Tuning model hyperparameters using GridSearchCV
- Calculate the accuracy on the test set for each model
- Plot Confusion Matrix

Model choosing:

- Choosing the model with the best accuracy score and

GitHub

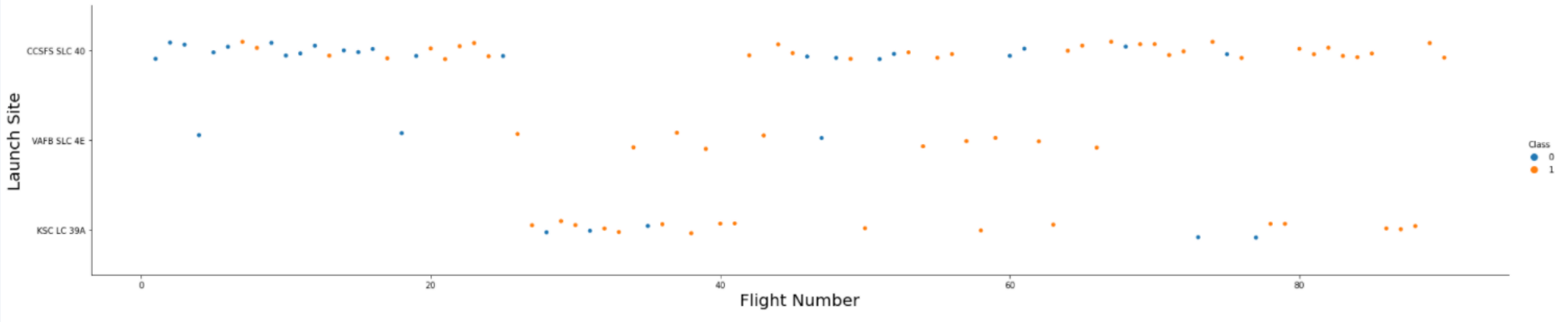
Results

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

Section 2

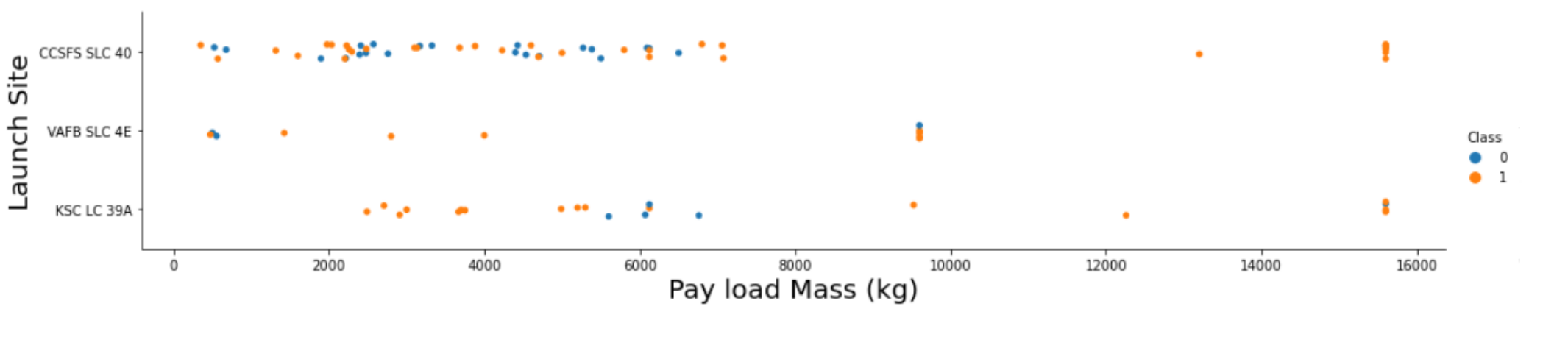
Insights drawn from EDA

Flight Number vs. Launch Site



We see that different launch sites have different success rates. CCAFS LC-40, has a success rate of 60 %, while KSC LC-39A and VAFB SLC 4E has a success rate of 77%.

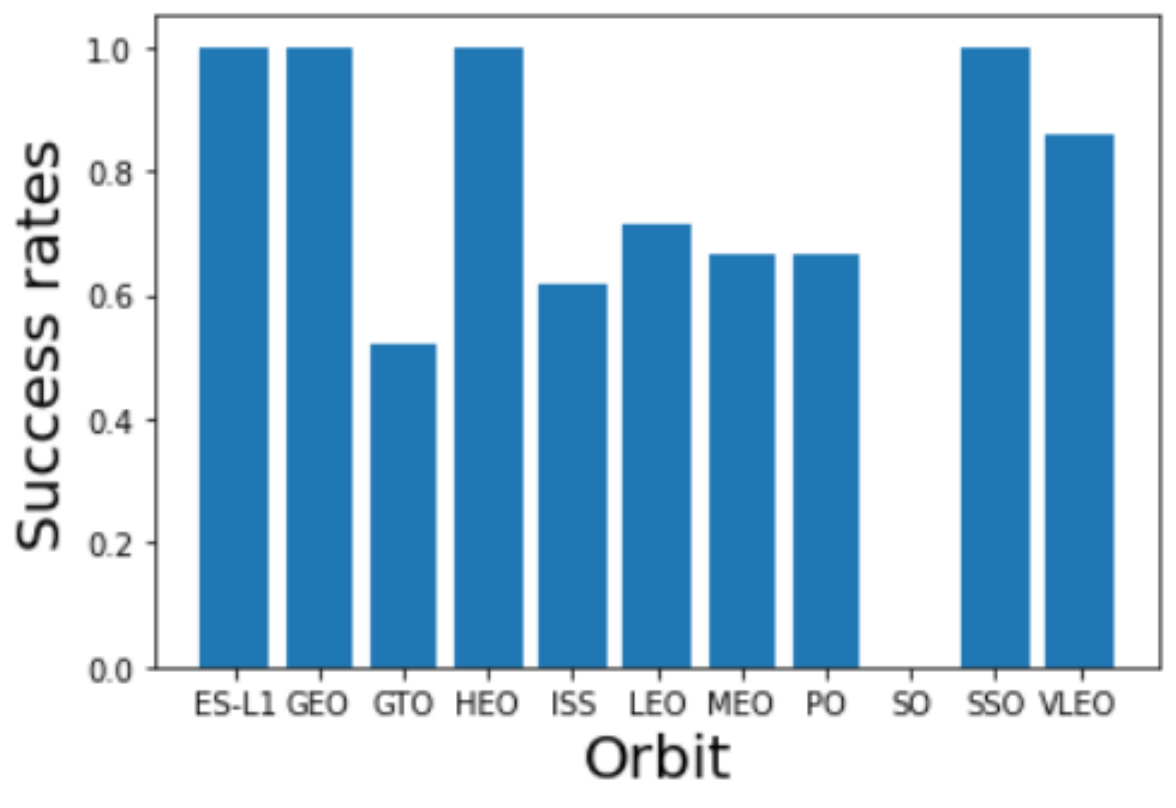
Payload vs. Launch Site



VAFB-SLC launch site there are no rockets launched for heavypayload mass(greater than 10000 kg).

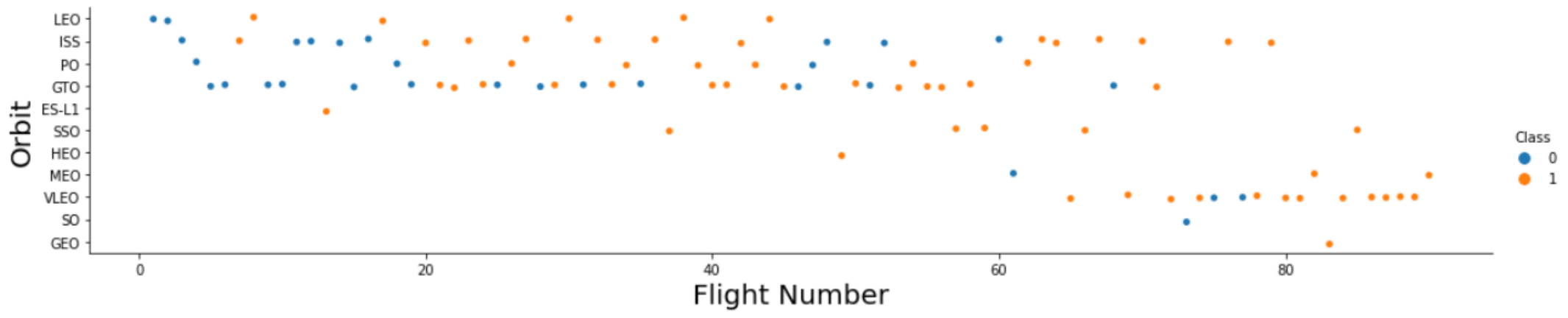
Rockets launched for heavy payload mass(greater than 10000 kg) has a very high probability of landing the first stage(more 85%).

Success Rate vs. Orbit Type



ES-L1, GEO, HEO and SSO have highest success rate

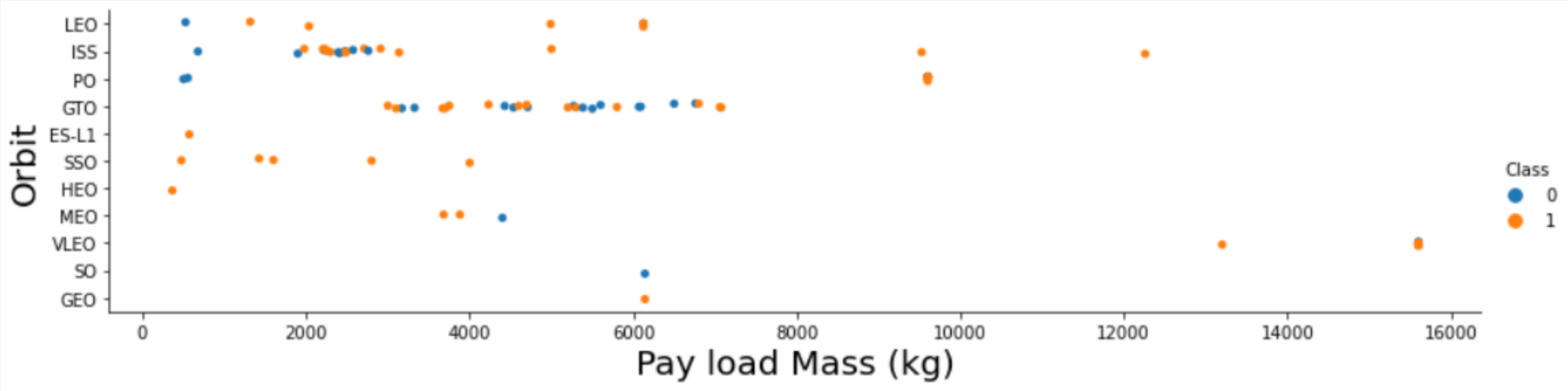
Flight Number vs. Orbit Type



In the LEO orbit the Success appears related to the number of flights.

On the other hand, there seems to be no relationship between flight number when in GTO orbit.

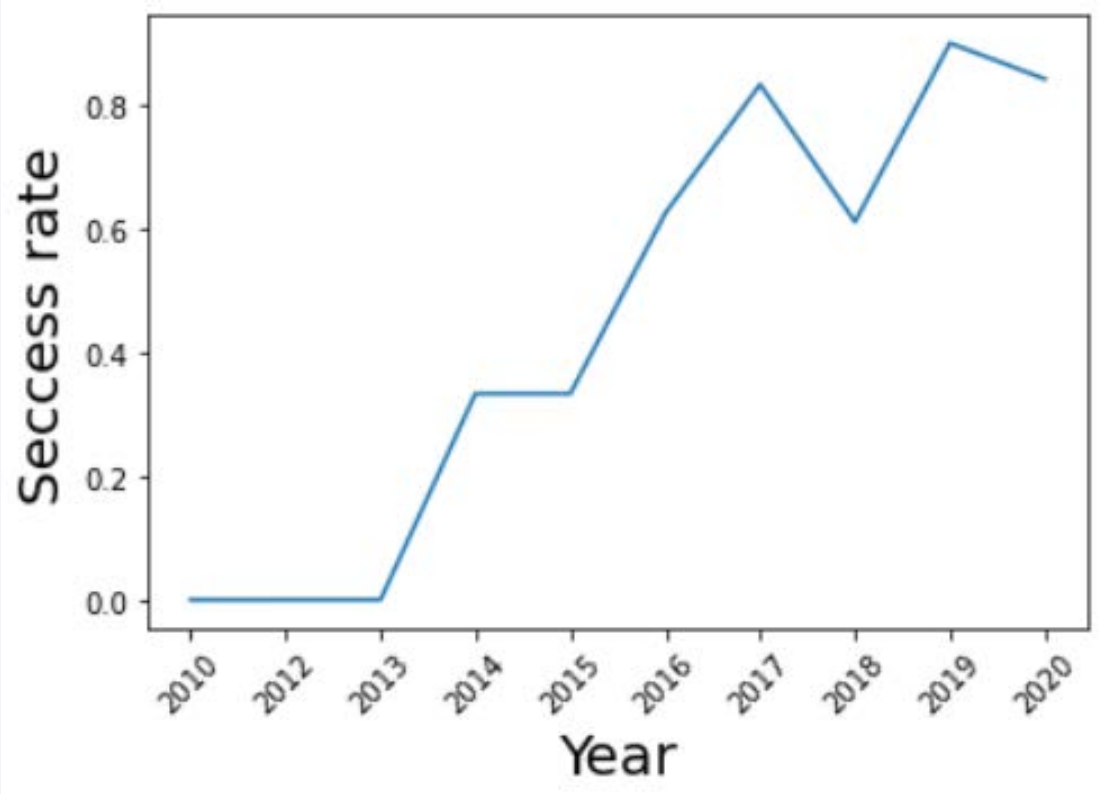
Payload vs. Orbit Type



With heavy payloads the successful landing or positive landing rate are more for Polar, LEO and ISS.

However for GTO we cannot distinguish this well as both positive landing rate and negative landing(unsuccessful mission) are both there here.

Launch Success Yearly Trend



The success rate since 2013 kept increasing till 2020.

All Launch Site Names

SELECT DISTINCT launch_site
FROM spacextbl

```
%%sql   SELECT DISTINCT launch_site
        FROM spacextbl
```

```
 * postgresql://postgres:***@localhost:5432/Sqltest
4 rows affected.
```

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

Launch Site Names Begin with 'CCA'

```
SELECT *
FROM spacextbl
WHERE launch_site LIKE 'CCA%'
LIMIT 5
```

Using the word LIMIT 5 in the query means that it will only show 5 records from table and LIKE keyword has a wild card with the words 'CCA%' the percentage in the end suggests that the launch_site name must begin with 'CCA'

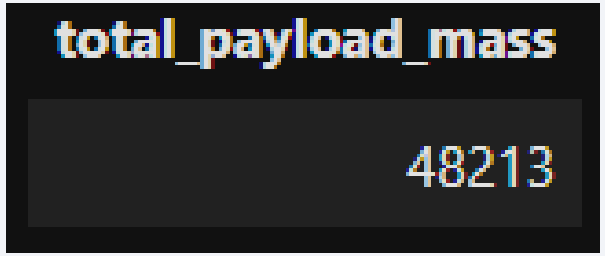
index	date	time	booster_version	launch_site	payload	payload_mass_kg	orbit	customer	mission_outcome	landing_outcome
0	2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
1	2010-12-08	15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight C1, two CubeSats, barrel of Brouere cheese	0	LEO (ISS)	NASA (COTS) NRO	Success	Failure (parachute)
2	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
3	2012-10-08	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
4	2013-03-01	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

Total Payload Mass

```
SELECT SUM(payload_mass_kg) AS total_payload_mass
FROM spacextbl
WHERE customer LIKE '%NASA (CRS)%'
```

Using the function SUM summates the total in the column payload_mass_kg.

LIKE '%NASA (CRS)%' is used to find all customers containing NASA (CRS).

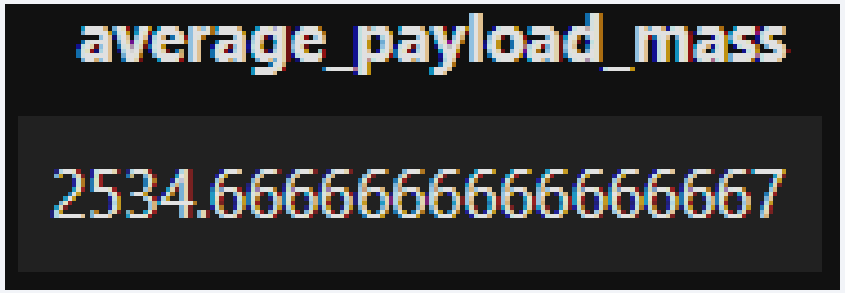


Average Payload Mass by F9 v1.1

```
SELECT AVG(payload_mass_kg) AS average_payload_mass
FROM spacextbl
WHERE booster_version LIKE '%F9%v1.1%'
```

Using the function AVG which allows you to find the average value for the column payload_mass_kg.

LIKE '%F9%v1.1%' is used to find all booster_version containing F9 v1.1.

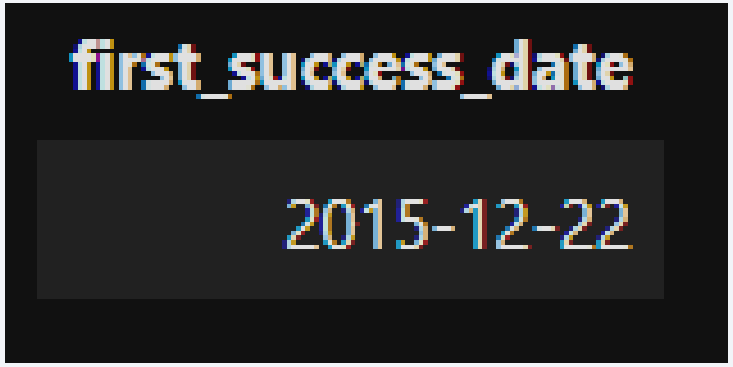


First Successful Ground Landing Date

```
SELECT MIN(date) AS first_success_date
FROM spacextbl
WHERE landing_outcome LIKE '%Success%'
```

Using the function MIN to find the smallest date.

LIKE '%Success%' is used to find all Success landing outcome.



Successful Drone Ship Landing with Payload between 4000 and 6000

```
SELECT booster_version, landing_outcome, payload_mass_kg
FROM spacextbl
WHERE landing_outcome LIKE '%Success%drone%' AND
                          payload_mass_kg BETWEEN 4000 AND 6000
```

LIKE '%Success%drone%' is used to find all Successful Drone Ship Landing . payload_mass_kg BETWEEN 4000 AND 6000 used to select all rockets with a mass between 4000 and 6000

booster_version	landing_outcome	payload_mass_kg
F9 FT B1022	Success (drone ship)	4696
F9 FT B1026	Success (drone ship)	4600
F9 FT B1021.2	Success (drone ship)	5300
F9 FT B1031.2	Success (drone ship)	5200

Total Number of Successful and Failure Mission Outcomes

```
SELECT mission_outcome, COUNT(mission_outcome) as total
FROM spacextbl
GROUP BY mission_outcome
```

We group table by mission outcome and count amount of outcomes per group.

mission_outcome	total
Success (payload status unclear)	1
Success	99
Failure (in flight)	1

Boosters Carried Maximum Payload

```
SELECT DISTINCT (booster_version), payload_mass_kg
FROM spacextbl
WHERE payload_mass_kg IN (SELECT MAX(payload_mass_kg)
                          FROM spacextbl)
```

Using the word DISTINCT in the query means that it will only show unique values in the booster_version column from the table. Using sub query for retrieving maximum payload mass.

booster_version	payload_mass_kg
F9 B5 B1048.4	15600
F9 B5 B1048.5	15600
F9 B5 B1049.4	15600
F9 B5 B1049.5	15600
F9 B5 B1049.7	15600
F9 B5 B1051.3	15600
F9 B5 B1051.4	15600
F9 B5 B1051.6	15600
F9 B5 B1056.4	15600
F9 B5 B1058.3	15600
F9 B5 B1060.2	15600
F9 B5 B1060.3	15600

2015 Launch Records

```
SELECT booster_version, launch_site, date, landing_outcome
FROM spacextbl
WHERE landing_outcome LIKE '%Failure%drone%' AND
                                    date BETWEEN '2014-12-31' AND '2016-01-01'
```

LIKE '%Failure%drone%' is used to find all Failure Drone Ship Landing .
data BETWEEN '2014-12-31' AND '2016-01-01' used to select all dates in 2015 year.

booster_version	launch_site	date	landing_outcome
F9 v1.1 B1012	CCAFS LC-40	2015-01-10	Failure (drone ship)
F9 v1.1 B1015	CCAFS LC-40	2015-04-14	Failure (drone ship)

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

```
SELECT landing_outcome, COUNT(mission_outcome) AS total
FROM spacextbl
WHERE date BETWEEN '2010-04-06' AND '2017-03-20'
GROUP BY landing_outcome
ORDER BY total DESC
```

Count the total number of landing results grouped by landing_outcomes.
data BETWEEN '2014-12-31' AND '2016-01-01' used to select all dates between 2010-06-04 and 2017-03-20.
Sort the received data using ORDER BY and DESC

landing_outcome	total
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 4

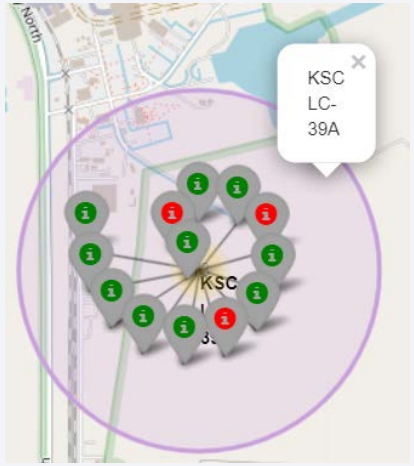
Launch Sites Proximities Analysis

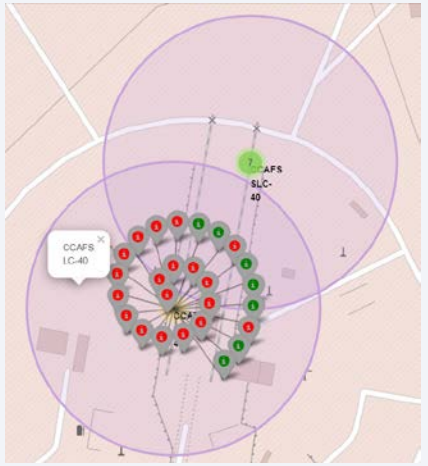
SpaceX launch sites

All launch sites are located in the United States close enough to the equator, which saves fuel when launching rockets.

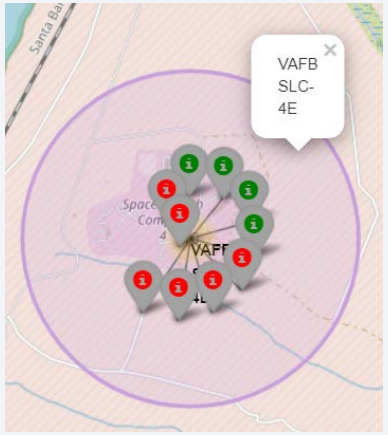
Color labeled markers

Florida launch site





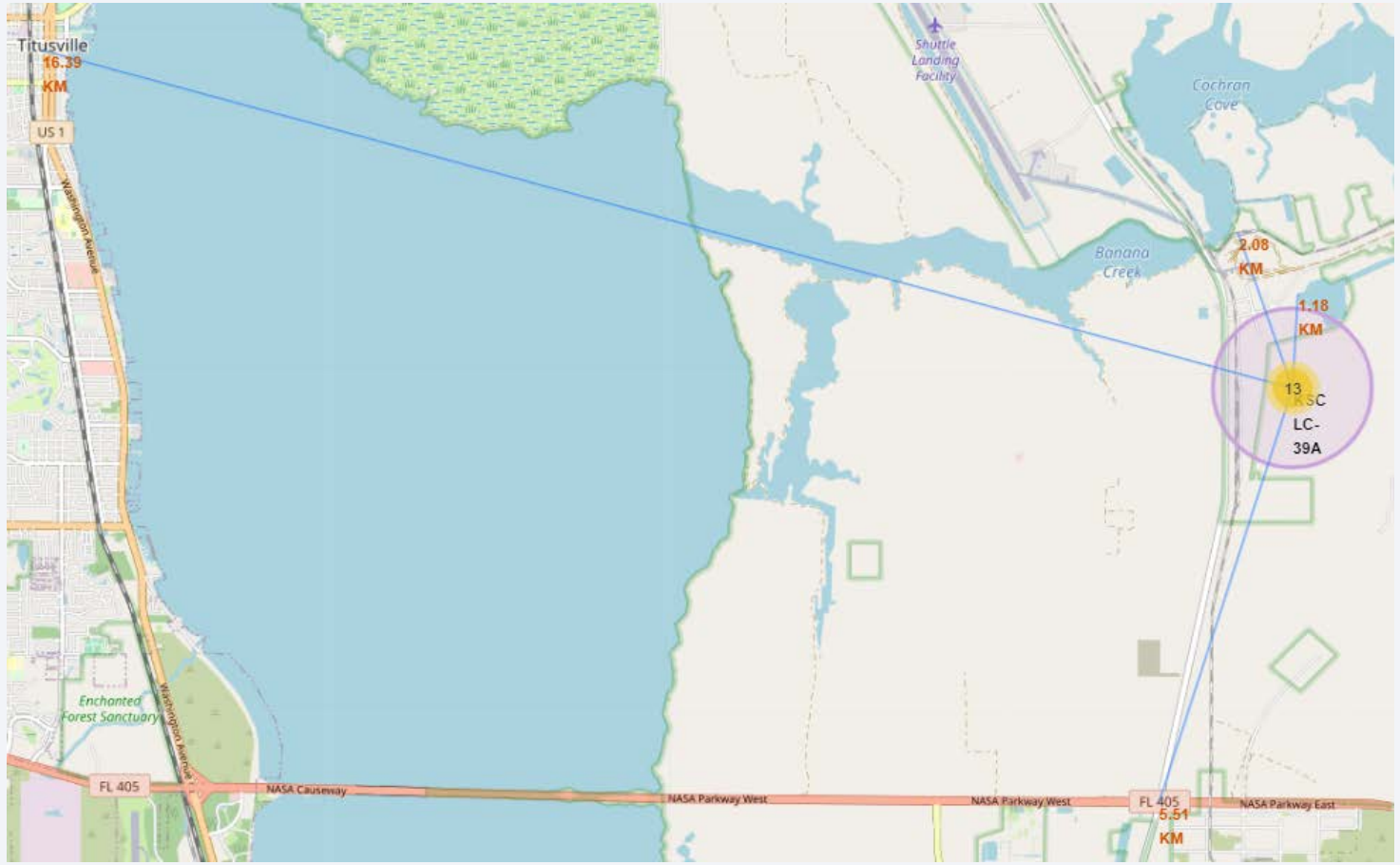
California launch site

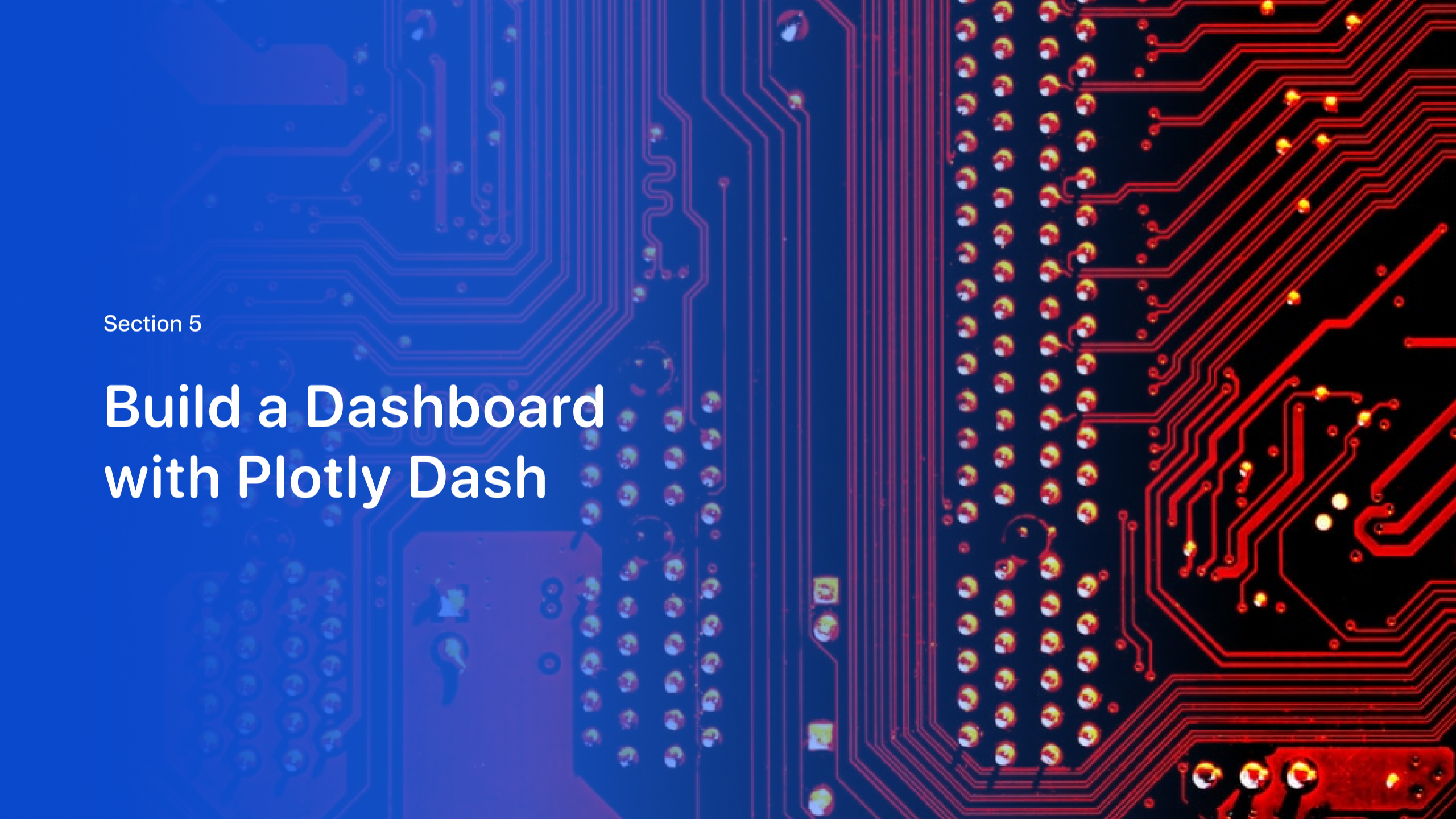


- Red marker - failures
- Green marker - success

Distance to the proximities

All launch sites are located close to supply routes such as highways, railways and coastlines. The proximity to the coasts and remoteness from cities is due to the safety of the inhabitants of these cities during launches and possible emergency situations, for example, noise, falling fragments, a blast wave. However, the nearest cities are located at such a distance that the launch site employees can comfortably get to their place of work.

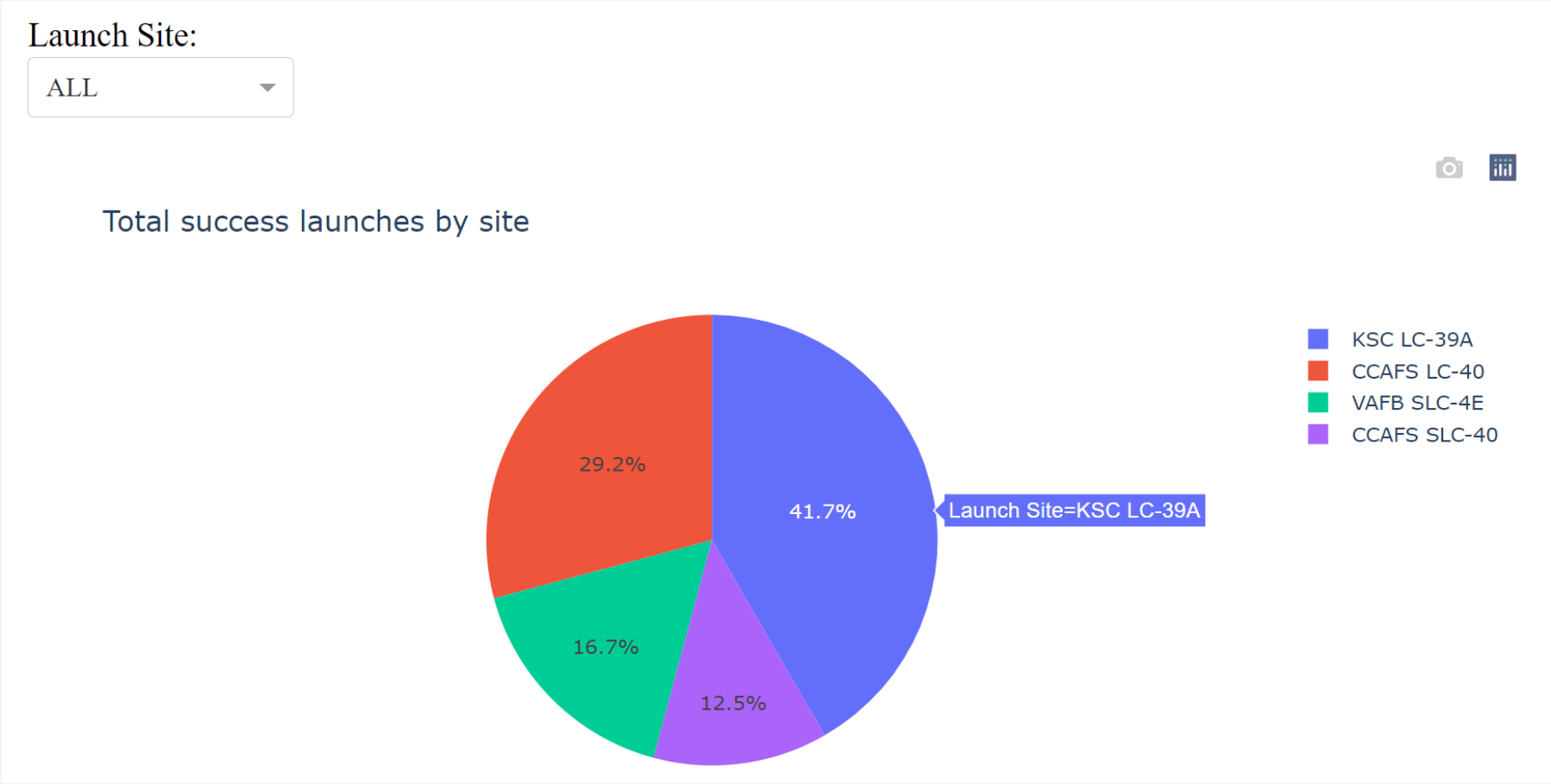




Section 5

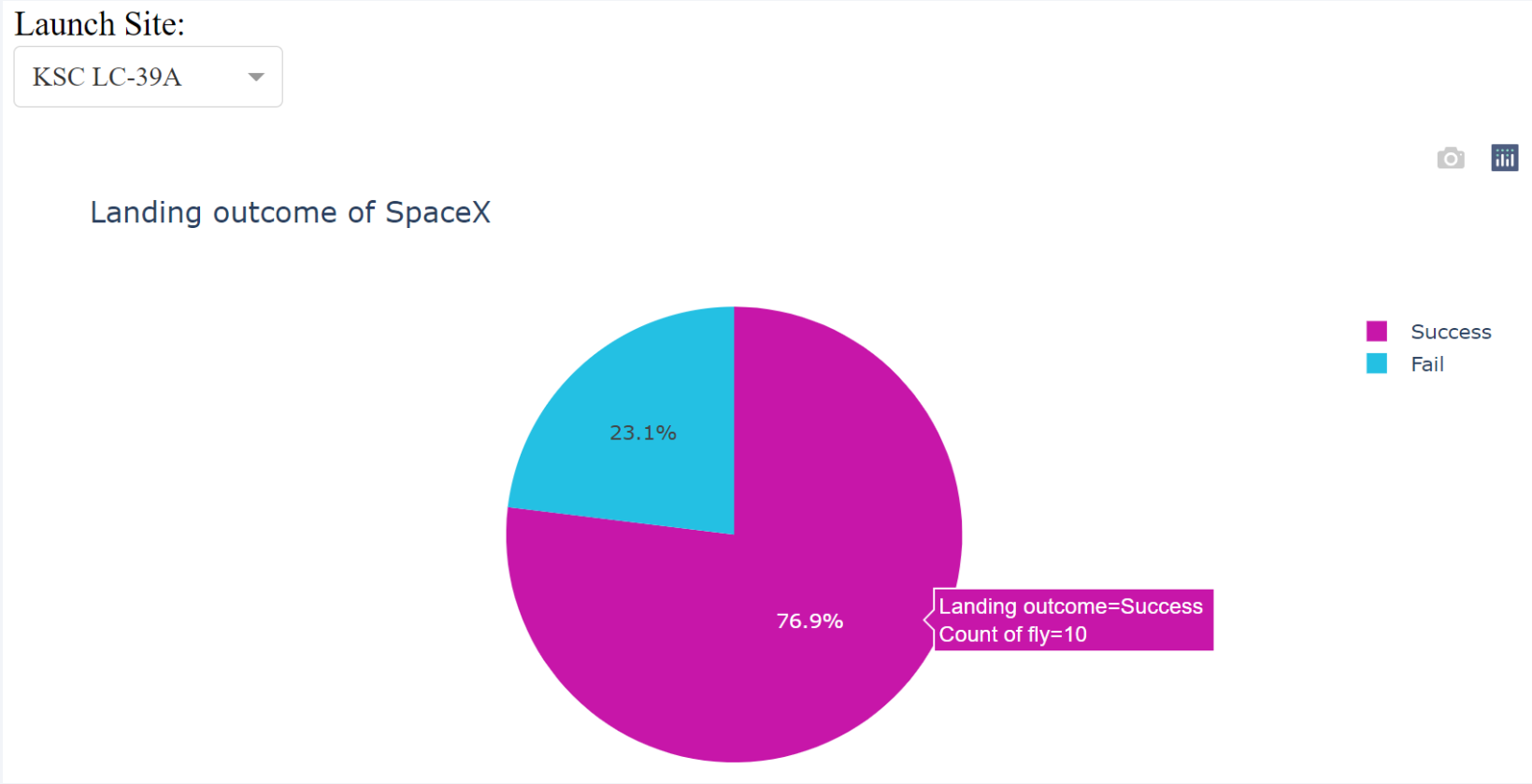
Build a Dashboard with Plotly Dash

Total success launchers by site



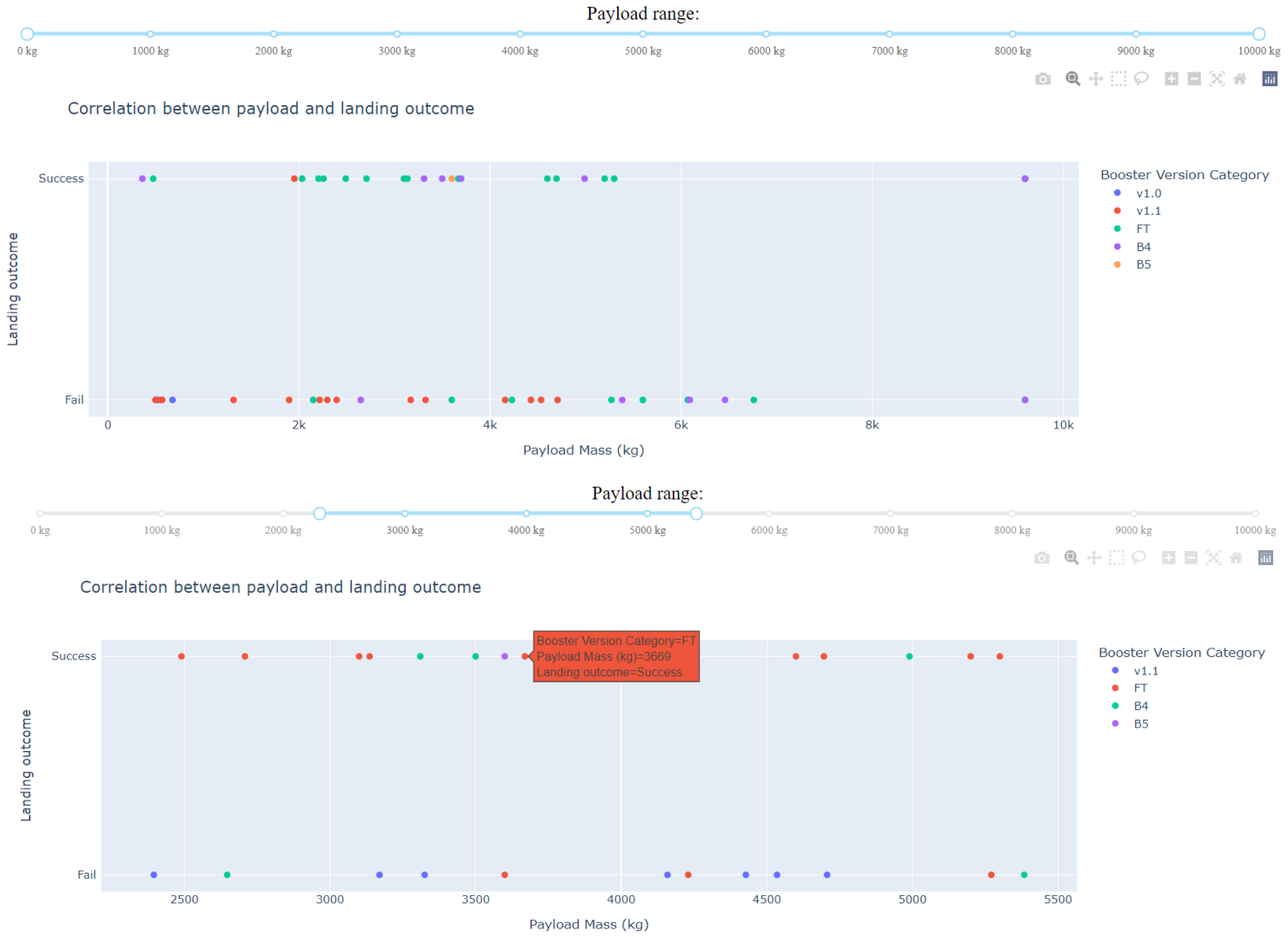
KSC LC-39A has the most successful first stage landings of any launch site

Landing outcome for KSC LC-39A launch site



The landing success rate of the first stage for the KSC LC-39A launch site was 76.9%.

Correlation between payload and landing outcome



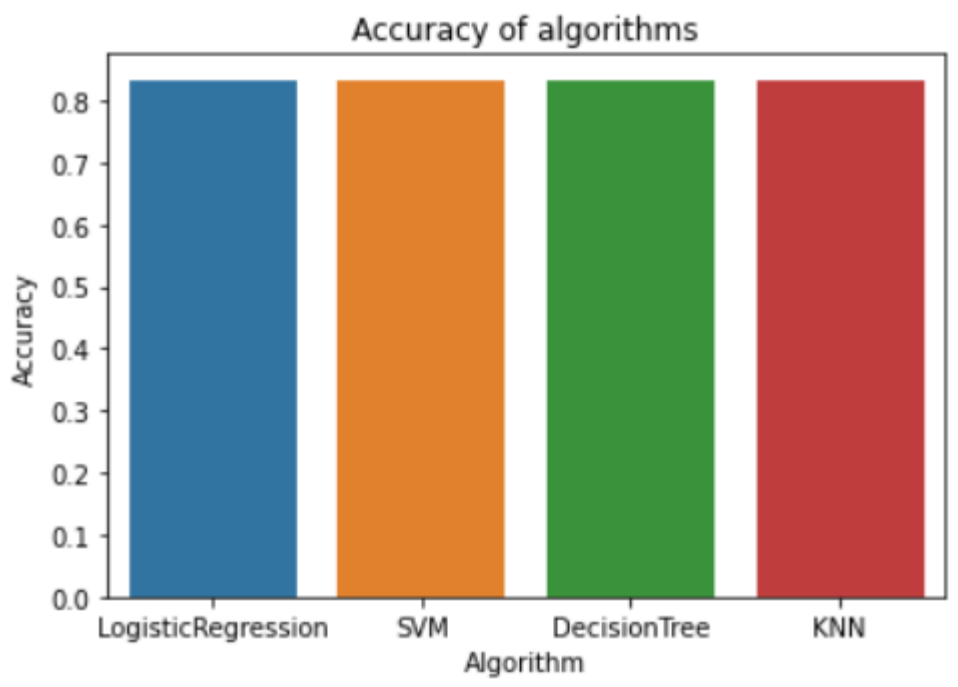
The first step landing success rate for Booster v1.0 and v1.1 is minimal.

Section 6

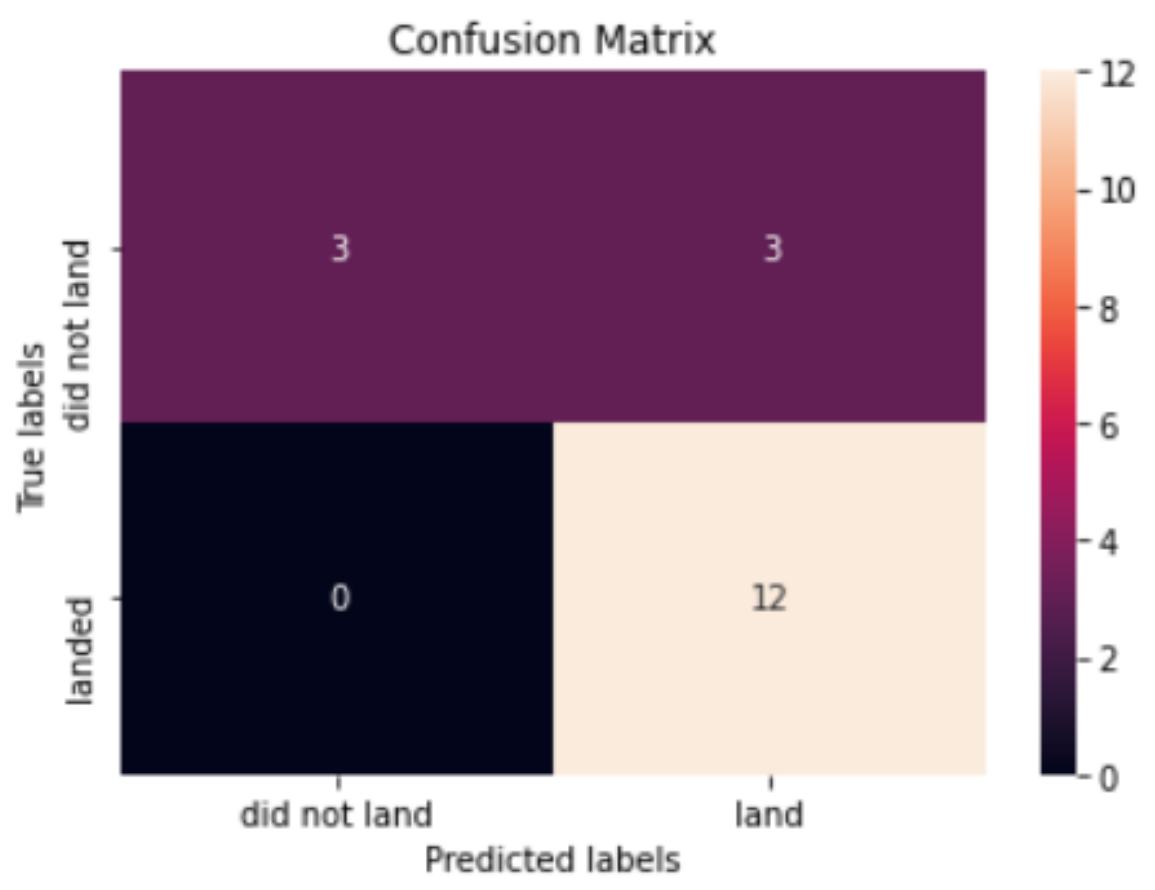
Predictive Analysis (Classification)

Classification Accuracy

Thanks to GridSearchCV, the optimal parameters were selected for all predictive models. And the prediction accuracy on the test set for all models was 83.3%. I chose the SVM model because this model has fewer type 2 errors compared to other models.



Confusion Matrix of SVM



Main problem of model is false positive

Conclusions

- SpaceX continually improves its launch vehicles, resulting in more successful first-stage rocket landings and lower costs for subsequent launches.
- For launch vehicles with a mass of more than 10,000 kg, the probability of successful landings of the first stage of the rocket is higher than 85%.
- The v1.0 and v1.1 launch vehicles showed only one successful landing of the first stage of the rocket.
- Orbit GEO, HEO, SSO, ES-L1 has the best landing success rate for the first stage of the rocket.
- The location of launch sites helps to reduce the cost of launches due to their proximity to the equator and logistical accessibility.

Thank you!