IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Sylvia Maczey
05.08.2022

Outline

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

Executive Summary

The goal of this project was to be able to predict if SpaceX will be able to reuse the first stage of a falcon 9 rocket.

We used machine learning techniques on data about past falcon 9 launches that we obtained via REST API and Scraping techniques on public web pages.

We cleaned the data, assigned labels, and performed an Exploratory Data Analysis(EDA) to find dependencies and correlations. For this we used different plots like scatter, bar, and pie charts as well as SQL. We also examined the geographical environments of the individual launch sites using folium. This enabled us to find the relevant features for training the machine learning model.

For the predictive analysis we used four different classifiers in combination with Hyper parameter tuning and compared the results.

The Decision Tree Classifier performed best with an accuracy of 0.91

Predicting the landing outcome is possible with a probability of about 90%

Introduction

The company SpaceX can perform Falcon 9 rocket launches at relatively low costs because they can reuse the first stage of the rocket. A successful landing however is not always possible.

The company SpaceX as well as it's competitors would benefit from being able to predict whether for a planned launch the landing will be successful or not.

In this project we use machine learning techniques to predict the landing outcome.

We collect available data on past rocket launches, including launch sites, payload mass, orbit types, explore the data to look for dependencies and correlations.

We prepare the data for training and testing a machine learning model. We use hyper parameter tuning to get the best results from the ML models.

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Rest API calls and Web scraping on a Wiki page
- Perform data wrangling
  - Fill missing values, determine data types, normalize data and add first labels
- 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 ML models using 4 different classifiers, Hyper parameter tuning, evaluate classification models

Data Collection

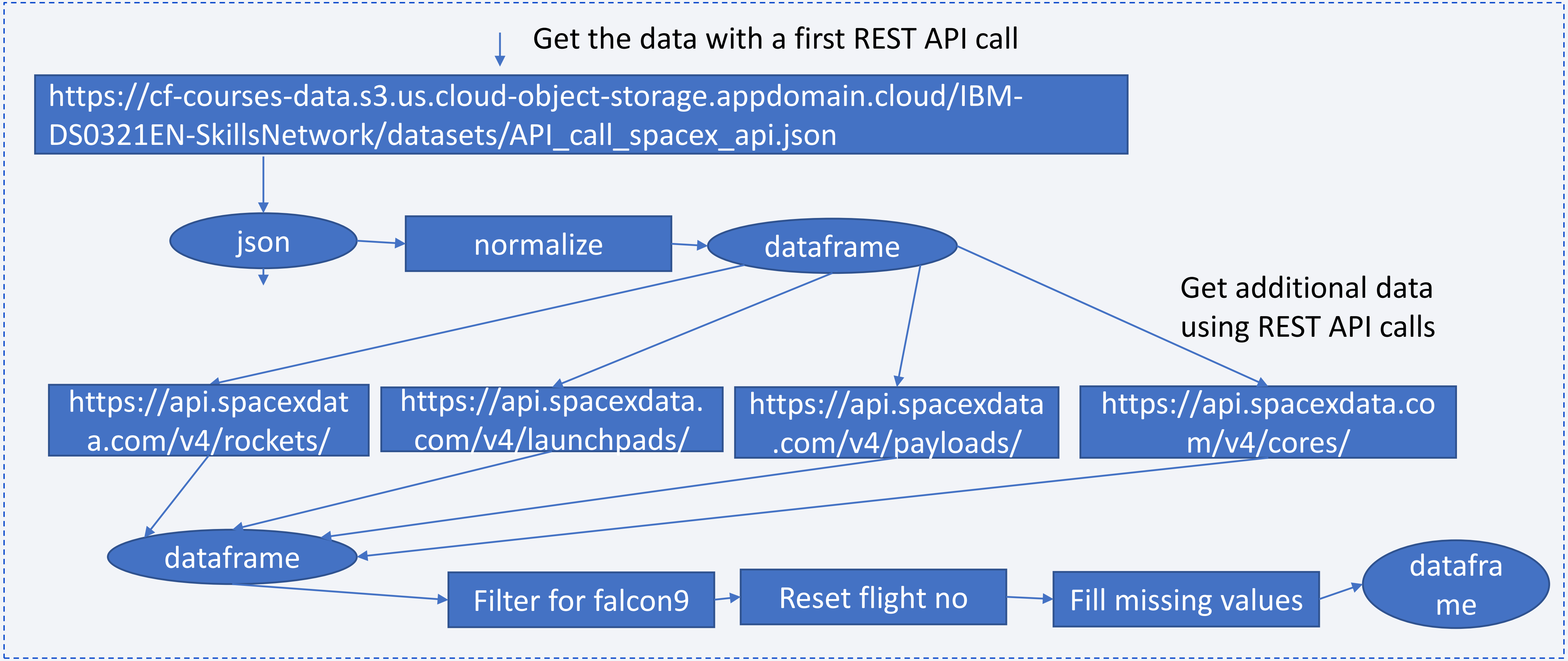
Data Collection using REST API functions

- First data about passed launches was collected from SpaceX
- The obtained data was then used to get additional data on rockets, launchpads, payloads and cores
- The data was filtered to only contain Falcon9 data, the flight number was reset, and missing values were cleaned filled by mean values or zeros

Data Collection using Web Scraping

- Additional data bout SpaceX launches was read from table contained in Wiki page

Data Collection – SpaceX API



Get the data with a first REST API call

https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DS0321EN-SkillsNetwork/datasets/API_call_spacex_api.json

Get additional data using REST API calls

https://api.spacexdata.com/v4/rockets/

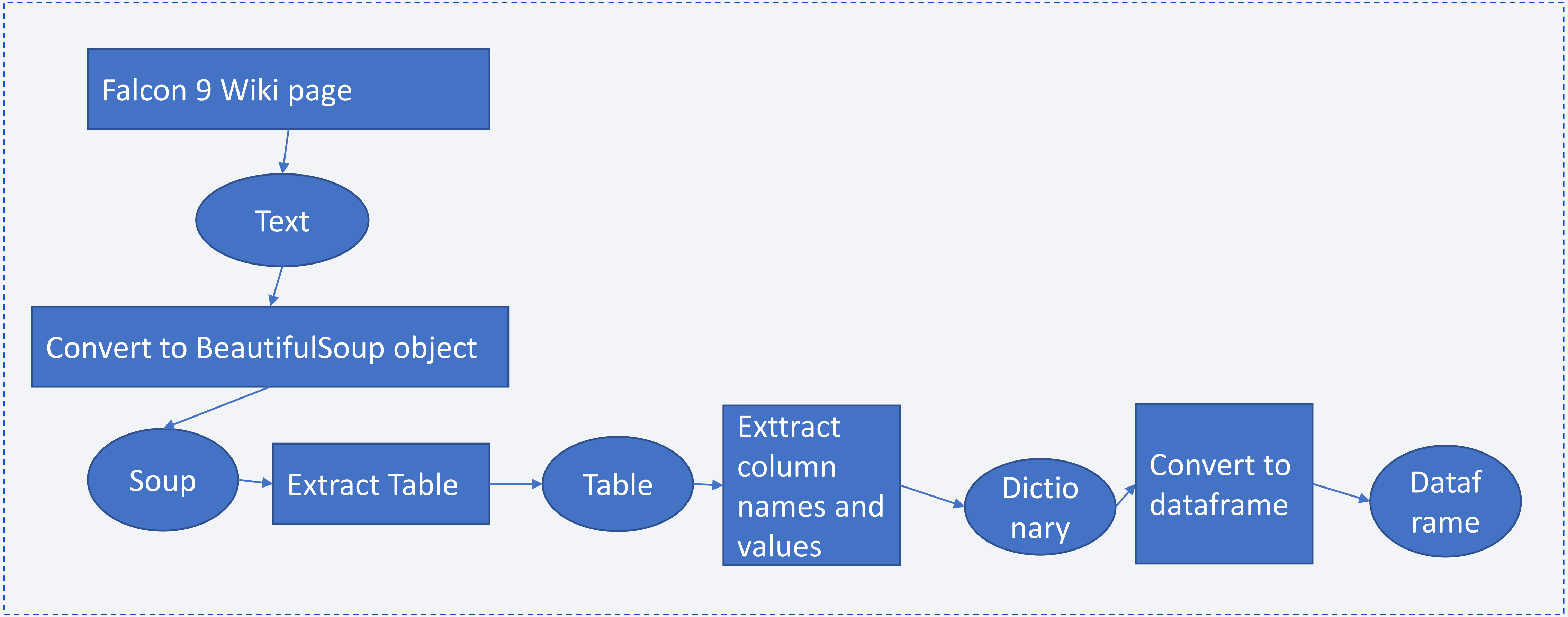
https://api.spacexdata.com/v4/launchpads/

https://api.spacexdata.com/v4/payloads/

https://api.spacexdata.com/v4/cores/

https://github.com/SylviaMaczey/courseraClass/blob/main/jupyter-labs-spacex-data-collection-api.ipynb

Data Collection - Scraping



Falcon 9 Wiki page

Text

Convert to BeautifulSoup object

Soup

Extract Table

Table

Exttract column names and values

Dictio nary

Convert to dataframe

Dataf rame

https://github.com/SylviaMaczey/courseraClass/blob/main/jupyter-labs-webscraping.ipynb

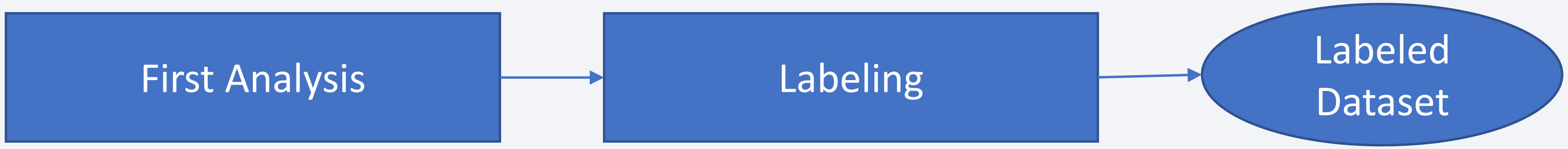
Data Wrangling

First Analysis:

- How many missing values
- What are the data types of the column values
- How many occurrences of categorical values like LaunchSite, Orbit, Outcome

Labeling:

- Label the records with value 0 for a negative outcome, with a 1 for a positive outcome



https://github.com/SylviaMaczey/courseraClass/blob/main/labs-jupyter-spacex-Data%20wrangling.ipynb

EDA with Data Visualization

For the Exploratory Data Analysis, Catplots were created to discover influences of the individual data properties on the fight outcome. One property was plotted against another, overlayed with the fight outcome.

To discover the influence of the Orbit property on the landing outcome a bar chart was created

A line plot was used to show the yearly success trend

These plots give important hints on what which features to select for machine learning tasks

https://github.com/SylviaMaczey/courseraClass/blob/main/jupyter-labs-eda-dataviz.ipynb

EDA with SQL

The SQL queries were used for the following purposes

- Find distinct values of a column
- Look at a limited amount of the data
- Compute sums and averages for column values
- Find the record with the earliest date
- Find records that that match combined conditions
- Aggregating the records to groups
- Ranking the data

https://github.com/SylviaMaczey/courseraClass/blob/main/jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

- Circles were added to the map to visualize the launch sites.
- Markers were added for each launch outcome, their color indicating success (green) or failure(red)
- Markers were added to a Marker cluster to show only aggregations on certain zoom-levels
- Lines were created to show distances of launch sites to coasts, railways, etc.

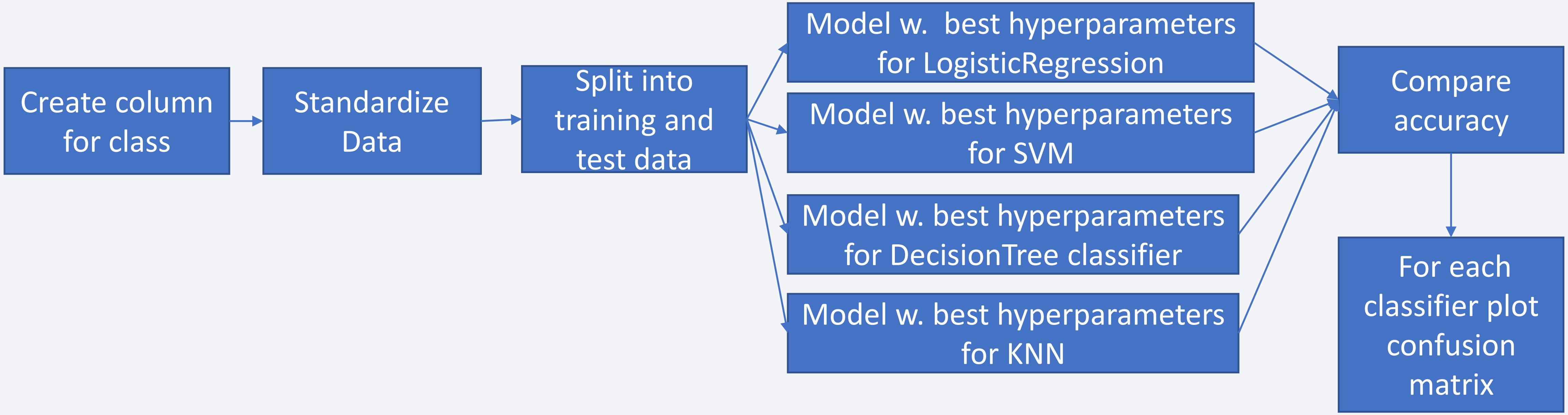
https://github.com/SylviaMaczey/courseraClass/blob/main/lab_jupyter_launch_site_location.ipynb

Build a Dashboard with Plotly Dash

- A pie chart was added to show the successful landing outcomes of all launch sites.
- A combo box was created to select one of the launch sites and make the pie chart show successful and unsuccessful landing outcomes for the selected site.
- A scatterplot was created to show the landing outcomes of a selected site in relation to the payload mass.
- A slider was added to interactively change the payload range to show on the scatter plot
- The interactions allow the user to view specific information for a launch site and specific payload ranges

https://github.com/SylviaMaczey/courseraClass/blob/main/spacex_dash_app.py

Predictive Analysis (Classification)


Create column for class
Standardize Data
Split into training and test data
Model w. best hyperparameters for LogisticRegression
Model w. best hyperparameters for SVM
Model w. best hyperparameters for DecisionTree classifier
Model w. best hyperparameters for KNN
Compare accuracy
For each classifier plot confusion matrix

A column for the landing_outcome to predict was created, the data was standardized using StandardScaler, the data was split into trainig and test data using train_test_split

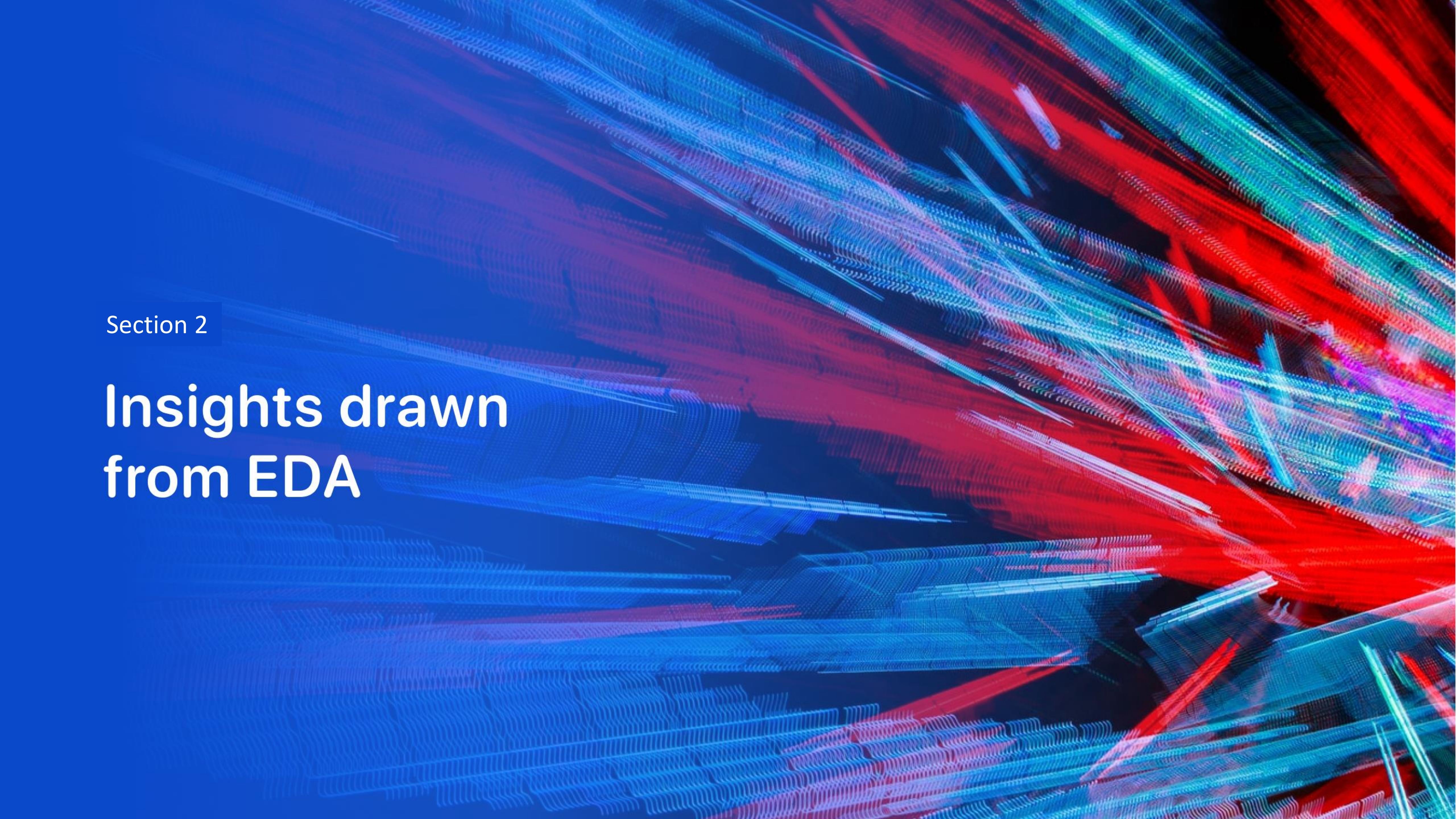
For each Classifier the best hyperparameters were used using GridSearchCV. The accuracy for the best estimators was calculated. The classifier models were tested on the test data and the results were plotted on a confusion matrix.

https://github.com/SylviaMaczey/courseraClass/blob/main/SpaceX_Machine Learning Prediction_Part_5.ipynb

Results

The exploratory data analysis showed that the success of the landing is dependent on payload mass, launch site, the year of launch and the orbit type. Screenshots will show that in later sections

- Screenshot from an interactive demos with folium and Dash will show these dependencies
- Predictive analysis results show that several classifiers give good results on predicting the landing outcome

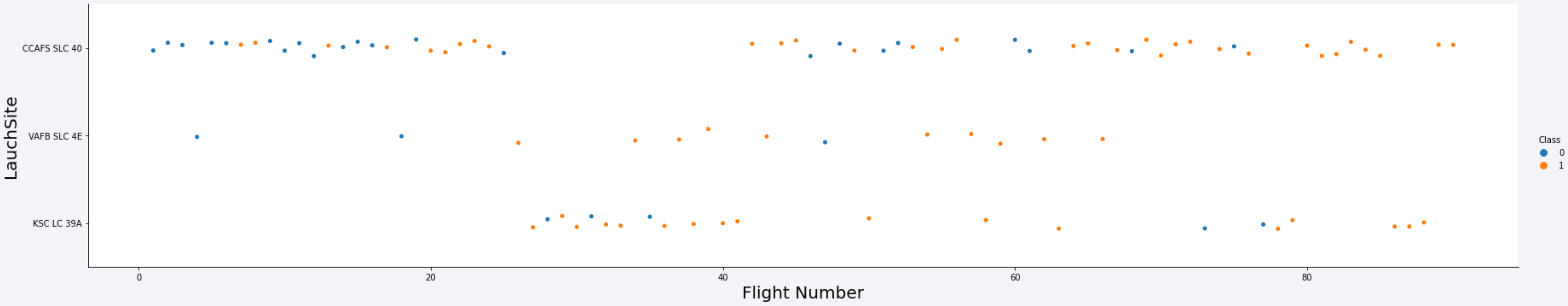


Section 2

Insights drawn from EDA

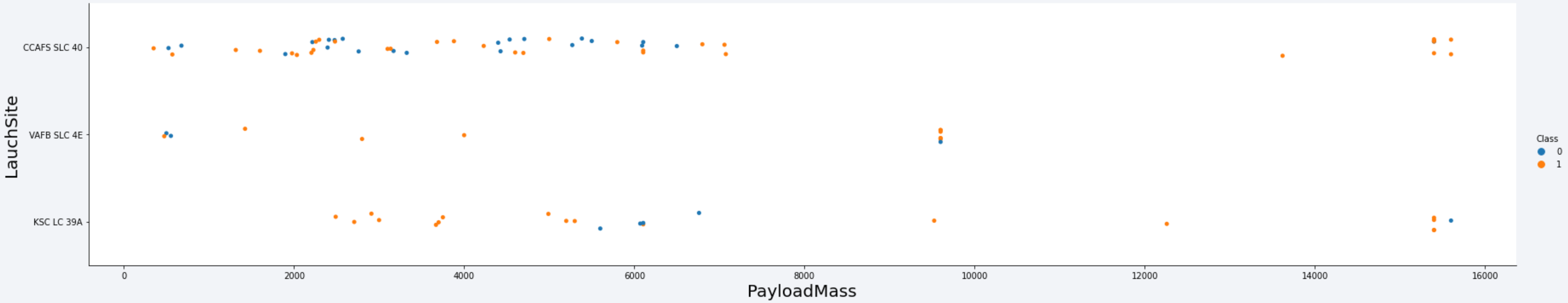
Flight Number vs. Launch Site

The following scatter plot shows that all Launch sites had more failures at the beginning, but were becoming more and more successful over time



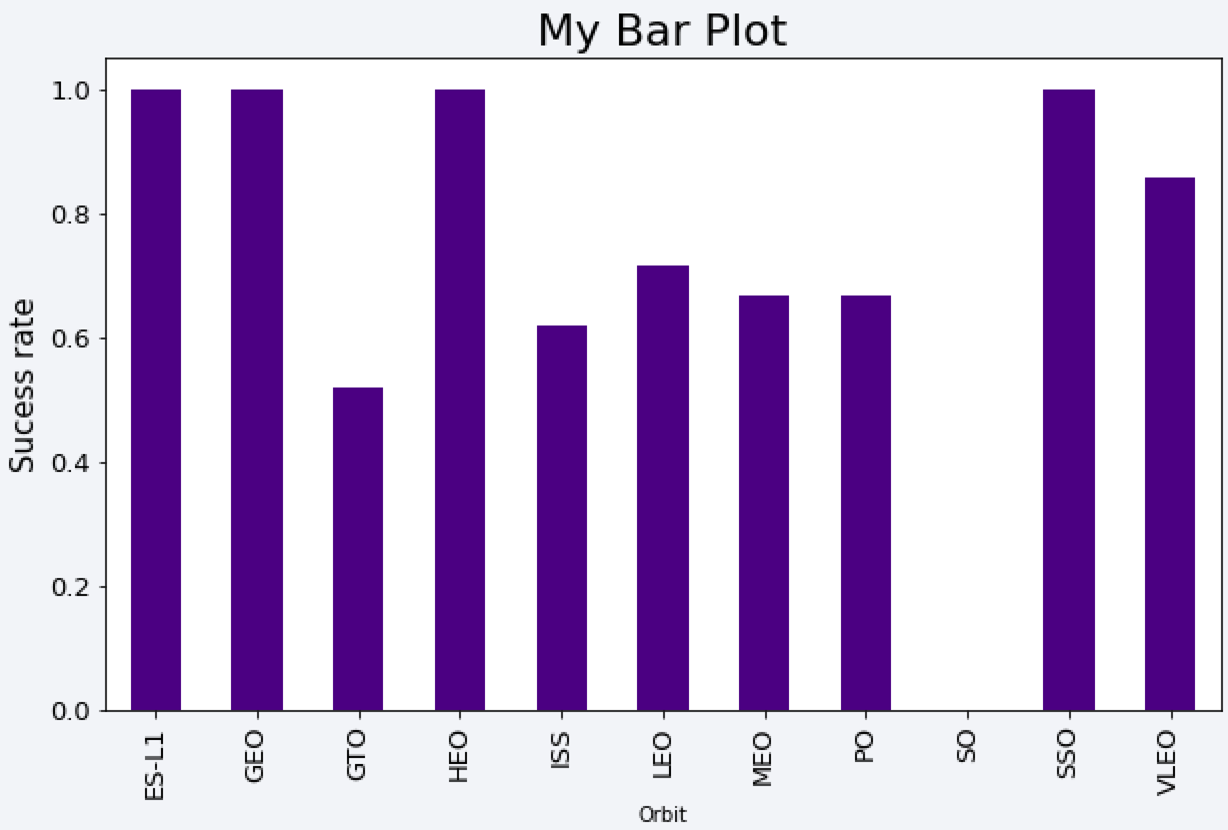
Payload vs. Launch Site

The following scatterplot shows that for the VAFB-SLC launch site there are no rockets launched for heavy payload mass(greater than 10000).



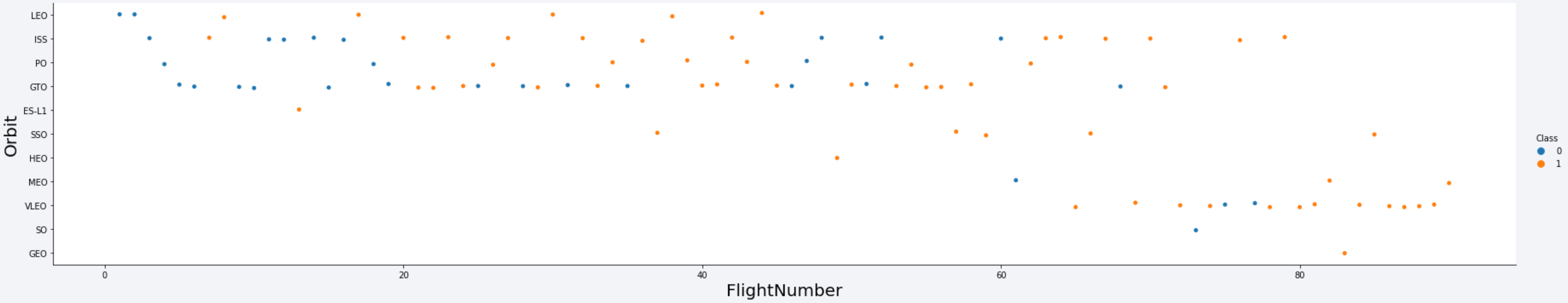
Success Rate vs. Orbit Type

The bar chart shows that there are 4 orbit types with a 100% success rate, and one orbit type with a 0% success rate



Flight Number vs. Orbit Type

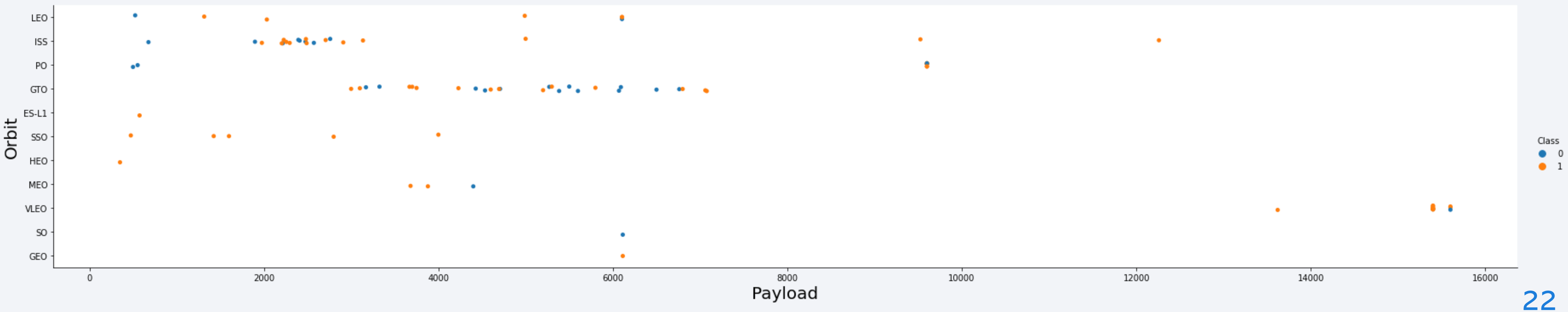
For the LEO orbit the Success appears related to the number of flights; on the other hand, there seems to be no relationship between flight number when in GTO orbit.



Payload vs. Orbit Type

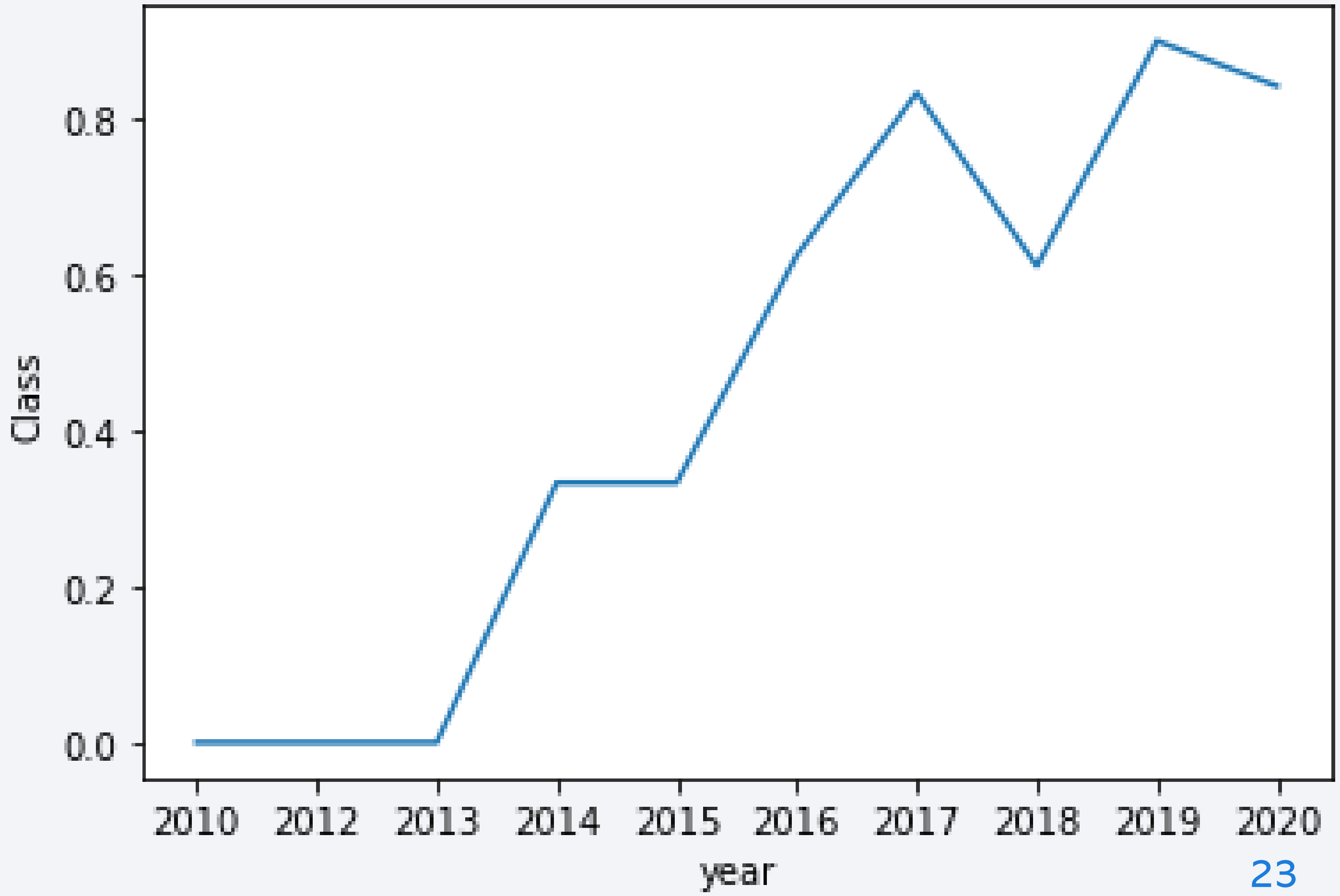
With heavy payloads the successful landing or positive landing rate are more for Polar,LEO and ISS.

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 plot on the right shows that the success rate has increased massively within 10 years, but there have also been fallbacks in between



All Launch Site Names

The Launch site names were obtained with the following query

```
select distinct Launch_Site from "SPACEXTBL"
```

The keyword “distinct” returns every value only once

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

Launch Site Names Begin with 'CCA'

The records were obtained with the following query

```
select * from "SPACEXTBL" where Launch_Site LIKE 'CCA%' LIMIT 5
```

The keywords “like” and “limit were used to return the desired once

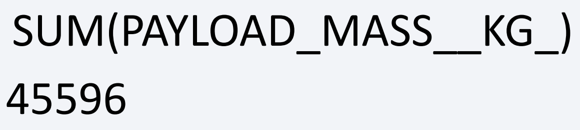
Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
04-06-2010	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
08-12-2010	15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight C1, two CubeSats, barrel of Brouere cheese	0	LEO (ISS)	NASA (COTS) NRO	Success	Failure (parachute)
22-05-2012	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
08-10-2012	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
01-03-2013	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

Total Payload Mass

The Total Payload Mass was obtained with the following query

```
select SUM(PAYLOAD_MASS__KG_) from "SPACEXTBL" where Customer
LIKE 'NASA (CRS)’
```

The keyword “SUM” sums up the values of all matching records



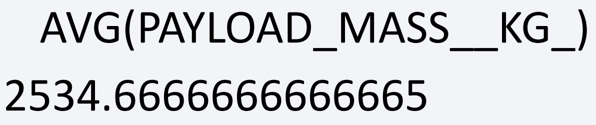
```
 SUM(PAYLOAD_MASS__KG_)
45596
```

Average Payload Mass by F9 v1.1

The Average Payload Mass was obtained with the following query

```
select AVG(PAYLOAD_MASS__KG_) from "SPACEXTBL" where
Booster_Version LIKE 'F9 v1.1%’
```

The keyword “AVG” computes the average of the values of all matching records



```
  AVG(PAYLOAD_MASS__KG_)
2534.6666666666665
```

First Successful Ground Landing Date

The First Successful Ground Landing Date was obtained with the following query

```
select MIN(DATE) from "SPACEXTBL" where Landing_Outcome LIKE
'Success (ground pad)’
```

The keyword “Min” on the Date column computes the earliest date of all matching records

```
   MIN(DATE)
01-05-2017
```

Successful Drone Ship Landing with Payload between 4000 and 6000

The result was obtained with the following query

```
select Booster_Version, Landing_Outcome, PAYLOAD_MASS__KG_
from "SPACEXTBL" where Landing_Outcome LIKE 'Success (drone
ship)' AND PAYLOAD_MASS__KG_ > 4000 AND PAYLOAD_MASS__KG_ <
6000
```

The necessary conditions were added to the WHERE-clause

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

The Total Number of Successful and Failure Mission Outcomes was obtained with the following query

```
select Mission_Outcome, COUNT(Mission_Outcome) from
"SPACEXTBL" GROUP BY  Mission_Outcome
```

The keyword “Group By” adds the values for each mission outcome

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

Boosters Carried Maximum Payload

The Boosters Carried Maximum Payload was obtained with the following query

```
select Distinct
Booster_Version from
"SPACEXTBL" where
PAYLOAD_MASS__KG_ = (select
MAX(PAYLOAD_MASS__KG_) from
"SPACEXTBL")
```

This could be achieved by using a subquery

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

The 2015 Launch Records were obtained with the following query

```
select substr(Date,4,2), Booster_Version, Launch_Site,
PAYLOAD_MASS__KG_, Landing_Outcome from "SPACEXTBL" where
Landing_Outcome LIKE 'Failure (drone ship)' AND
substr(Date,7,4)=='2015’
```

The substr function was used get the year from the Date value

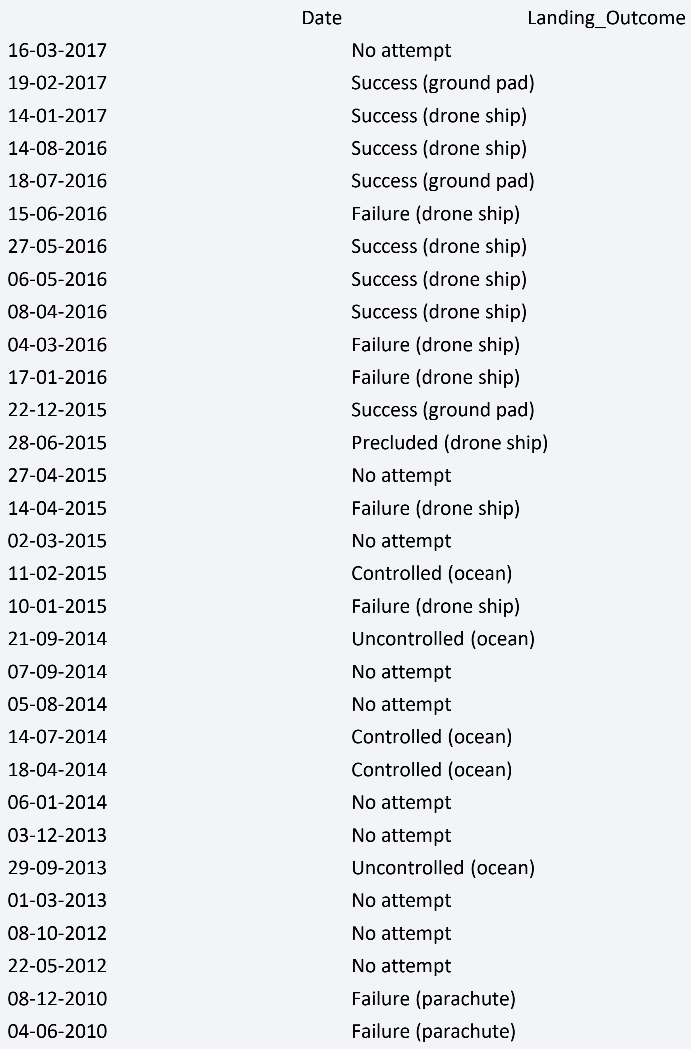
substr(Date,4,2)	Booster_Version	Launch_Site	PAYLOAD_MASS__KG_	Landing_Outcome
01	F9 v1.1 B1012	CCAFS LC-40	2395	Failure (drone ship)
04	F9 v1.1 B1015	CCAFS LC-40	1898	Failure (drone ship)

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

The Ranked Landing Outcomes Between 2010-06-04 and 2017-03-20 were obtained with the following query

```
select Date, Landing_Outcome from
"SPACEXTBL" where substr(Date,7,4) ||
substr(Date,4,2) || substr(Date,1,2)
between '20100604' and '20170320'
ORDER BY substr(Date,7,4) ||
substr(Date,4,2) || substr(Date,1,2)
DESC
```

The keyword “DESC” sorts the records in descending order

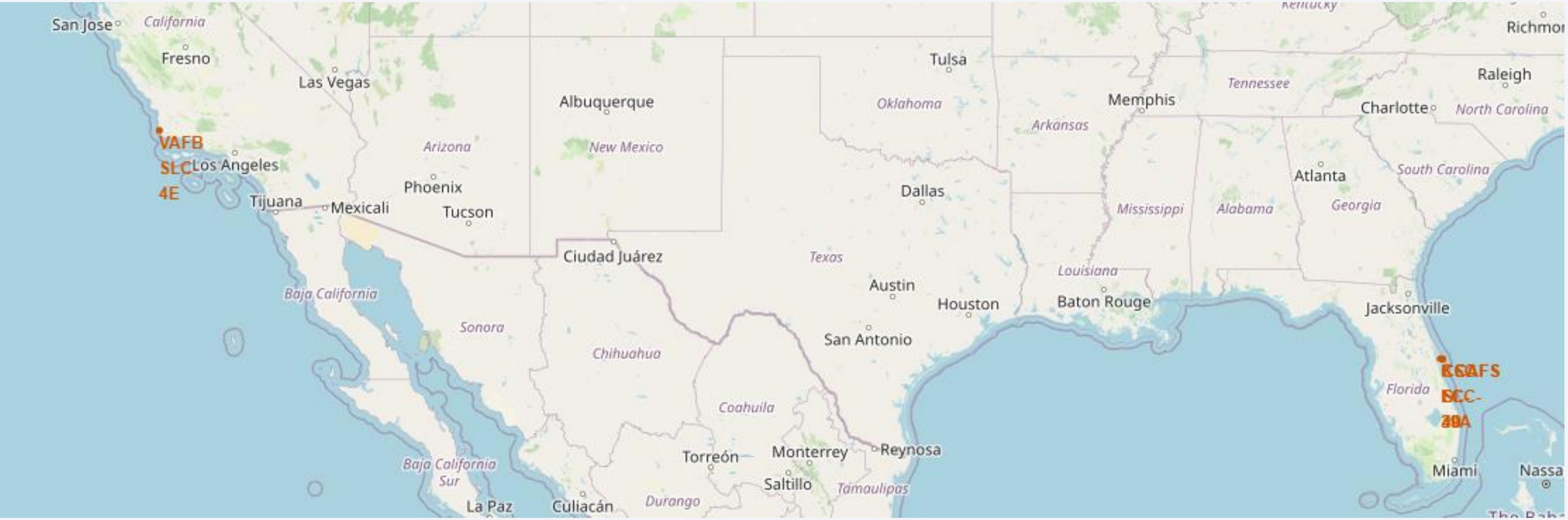


Date	Landing_Outcome
16-03-2017	No attempt
19-02-2017	Success (ground pad)
14-01-2017	Success (drone ship)
14-08-2016	Success (drone ship)
18-07-2016	Success (ground pad)
15-06-2016	Failure (drone ship)
27-05-2016	Success (drone ship)
06-05-2016	Success (drone ship)
08-04-2016	Success (drone ship)
04-03-2016	Failure (drone ship)
17-01-2016	Failure (drone ship)
22-12-2015	Success (ground pad)
28-06-2015	Precluded (drone ship)
27-04-2015	No attempt
14-04-2015	Failure (drone ship)
02-03-2015	No attempt
11-02-2015	Controlled (ocean)
10-01-2015	Failure (drone ship)
21-09-2014	Uncontrolled (ocean)
07-09-2014	No attempt
05-08-2014	No attempt
14-07-2014	Controlled (ocean)
18-04-2014	Controlled (ocean)
06-01-2014	No attempt
03-12-2013	No attempt
29-09-2013	Uncontrolled (ocean)
01-03-2013	No attempt
08-10-2012	No attempt
22-05-2012	No attempt
08-12-2010	Failure (parachute)
04-06-2010	Failure (parachute)

Section 3

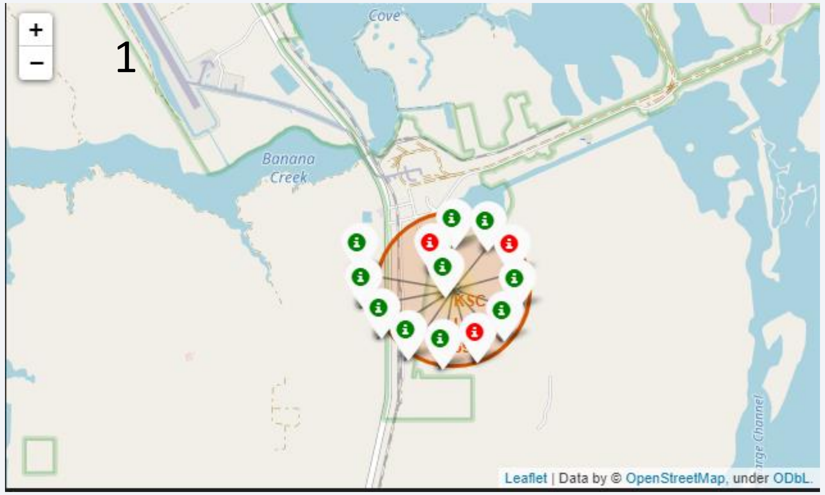
Launch Sites Proximities Analysis

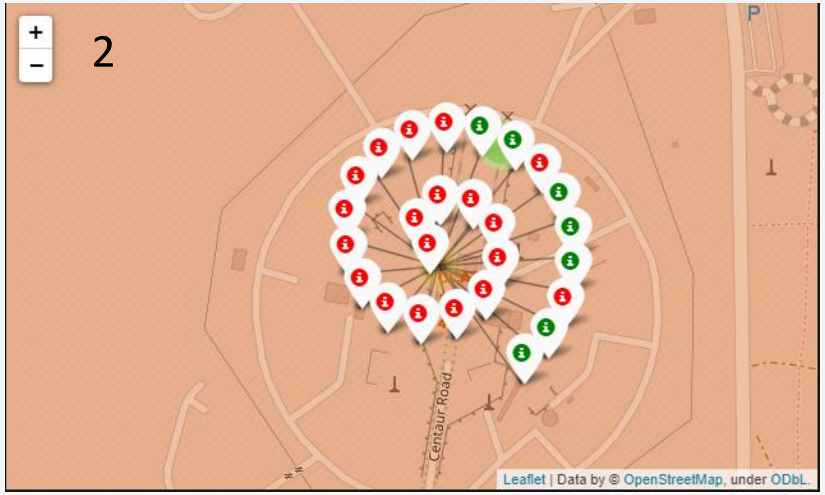
Launch Site Locations

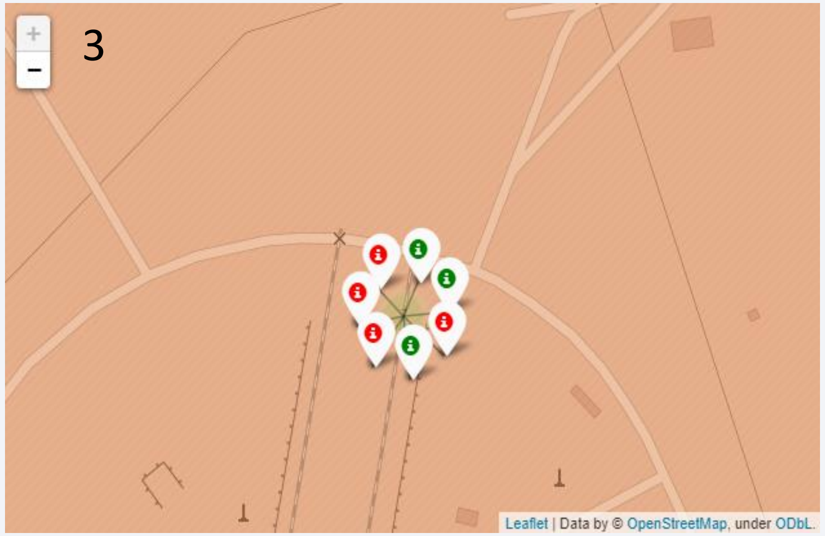


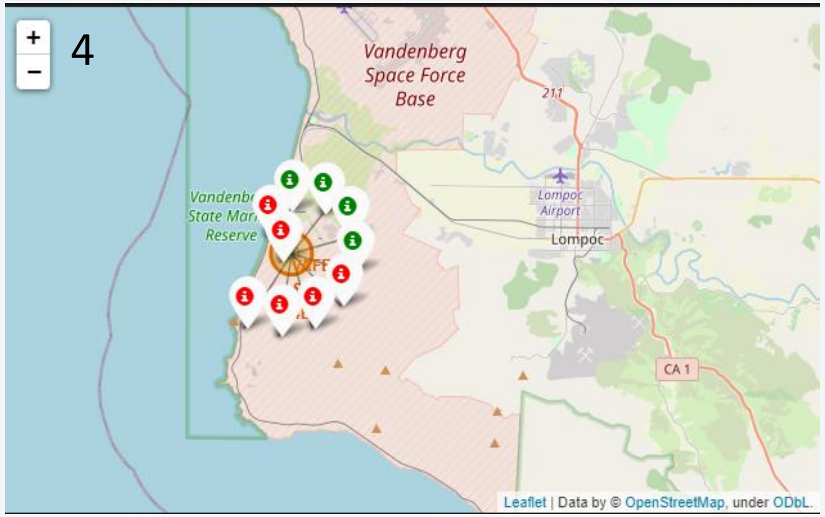
1 Launch Site can be found in California and 3 in Florida. On this zoom level the three locations in Florida are hard to distinguish.

Number of Successful and Unsuccessful Launches








4
Vandenberg Space Force Base
Vandenberg State Marine Reserve
Lompoc Airport
Lompoc
CA 1
Leaflet | Data by © OpenStreetMap, under ODbL.

1:KSC LC-39A

successful: 10

unsuccessful:3

2:CCAFS LC-40

successful: 7

unsuccessful:19

3:CCAFS SLC-40

successful: 3

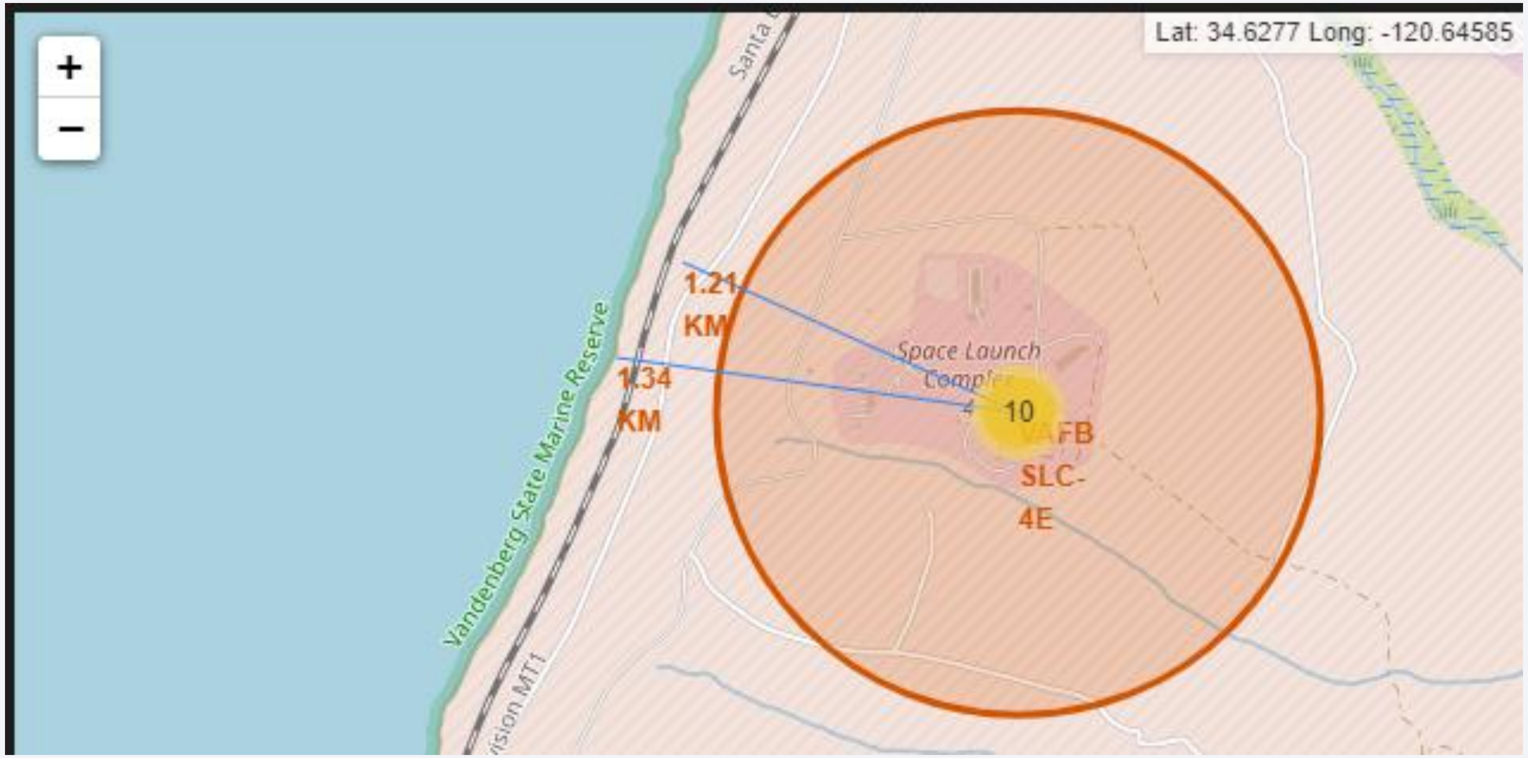
unsuccessful:4

4:VAFB SLC-4E

successful: 4

unsuccessful:6

Distance to Coast and Railways

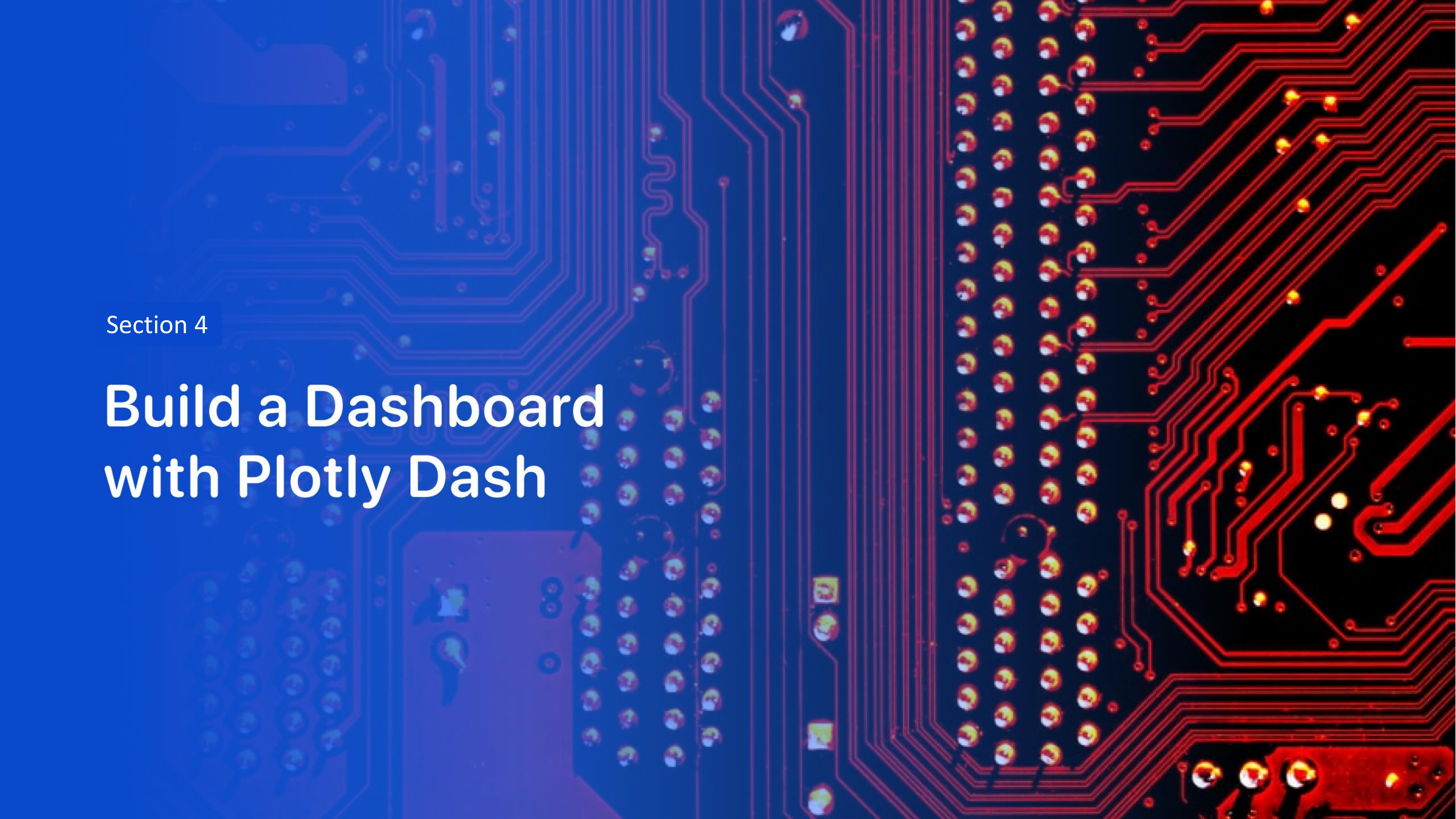


The straight blue lines indicate the distances of the launch site to the coast (1,34 km) and to the railways (1,21 km)

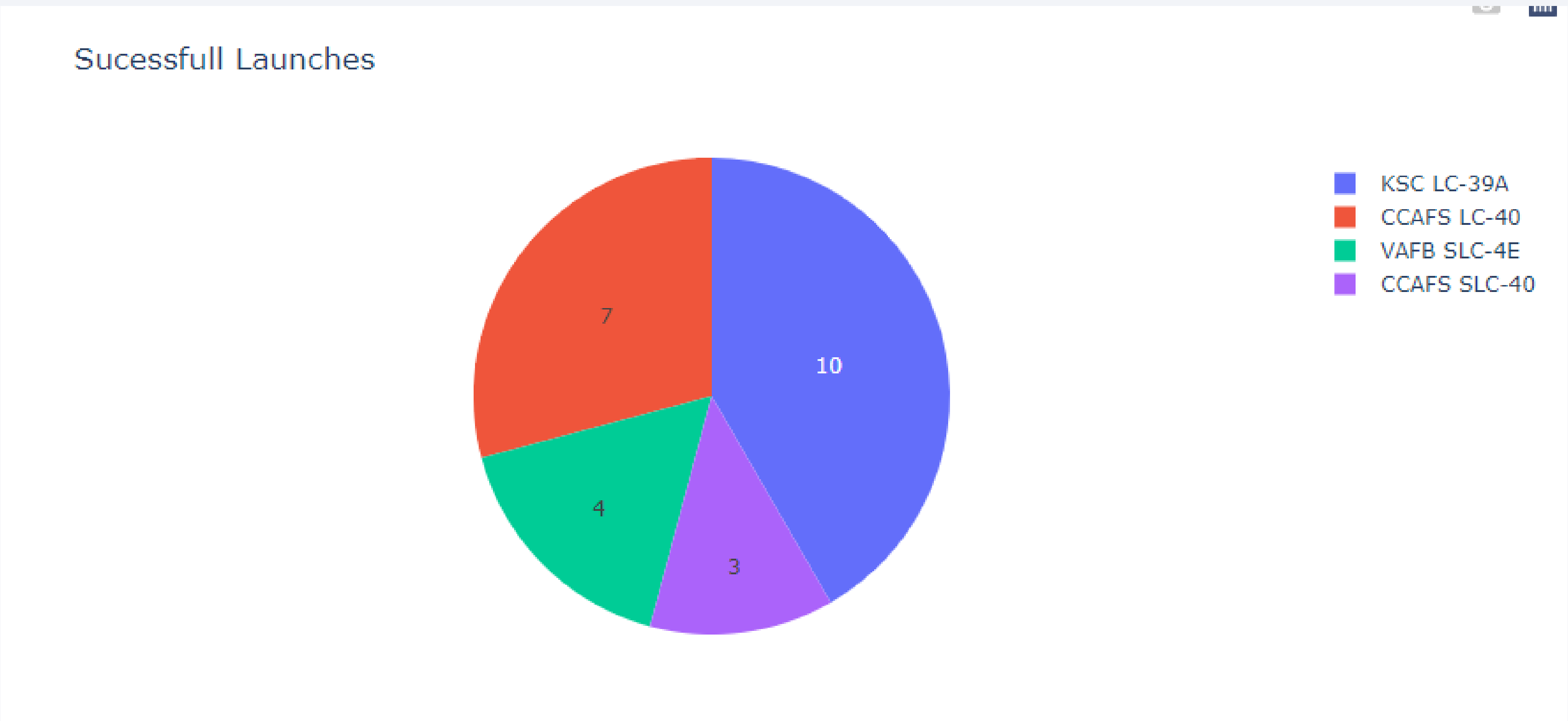
Short distances to coast and railways seem to be an locational advantage

Section 4

Build a Dashboard with Plotly Dash



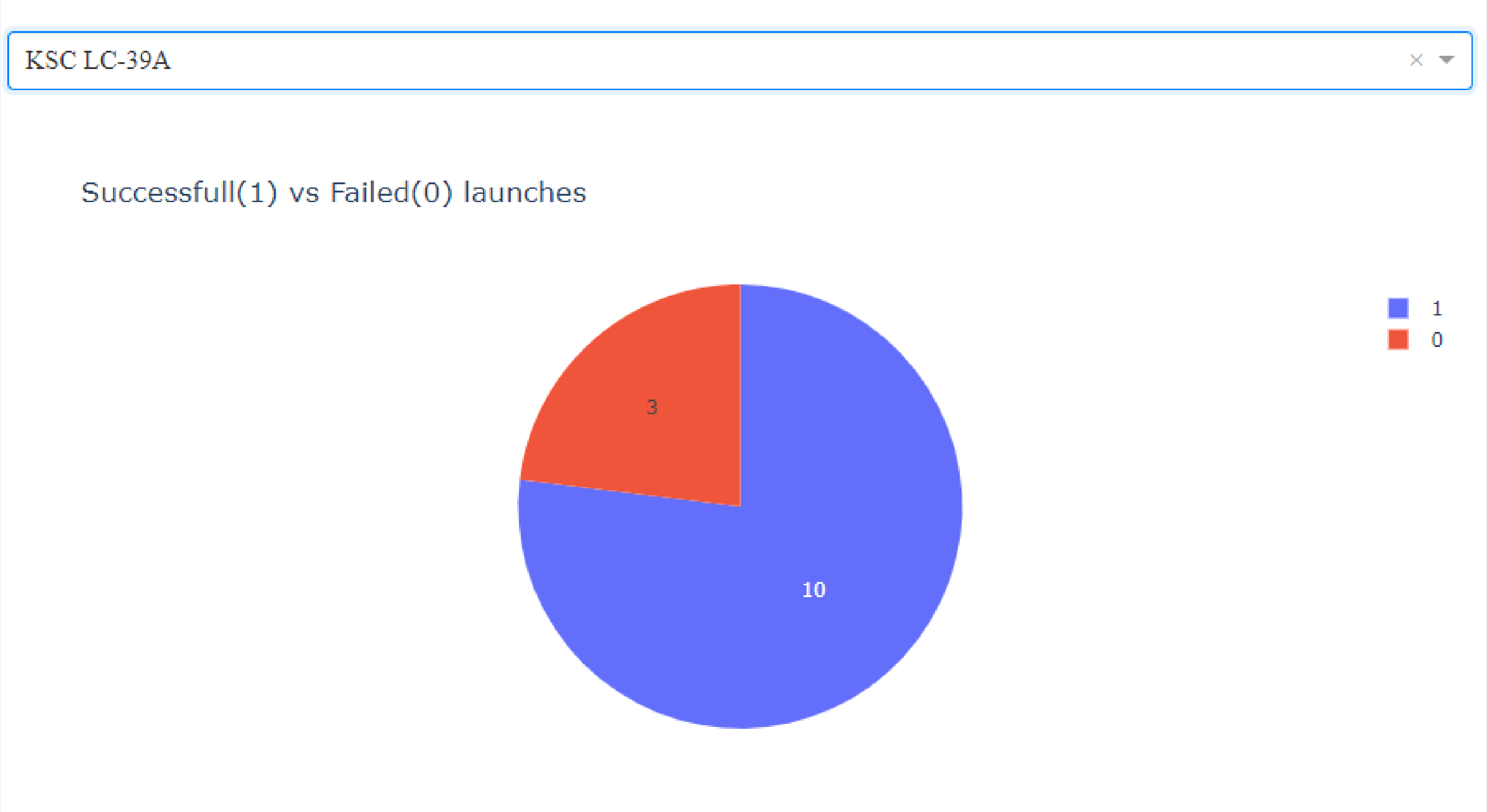
Successful Launches



Highest number of successful launches by KSC LC-39A.

To determine if it is the most successful launch site, we need to compare it to the unsuccessful launches

KSC LC-39A: Successes and Failures

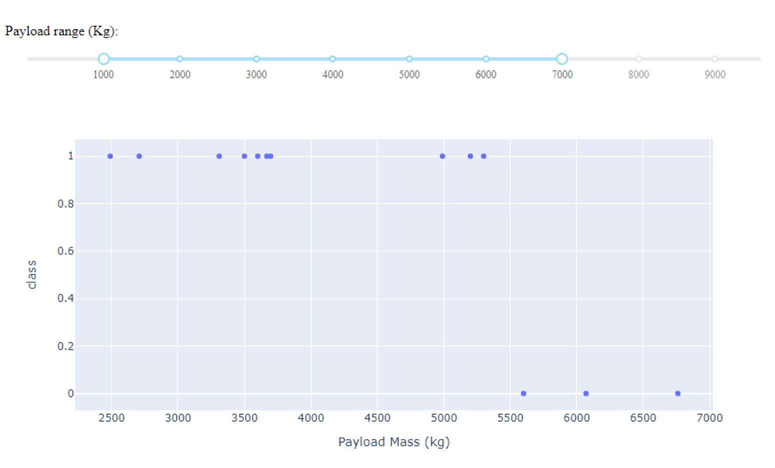


KSC LC-39A

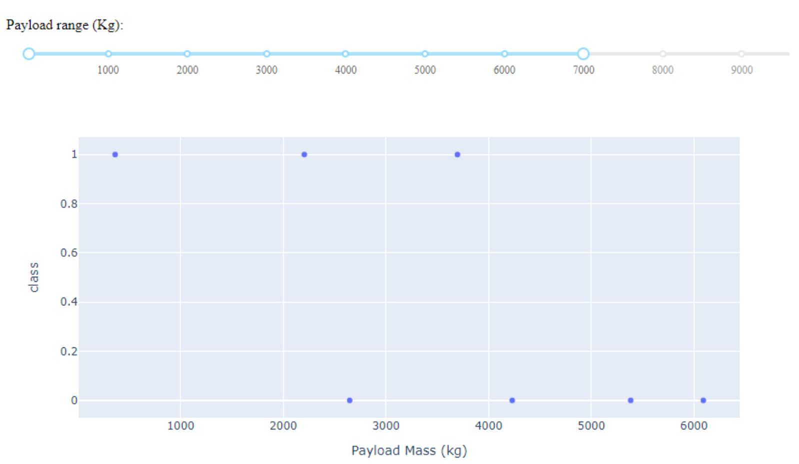
- Ten successes versus three fails: The success rate for this launch site is very high.

Payload versus Launch Outcome

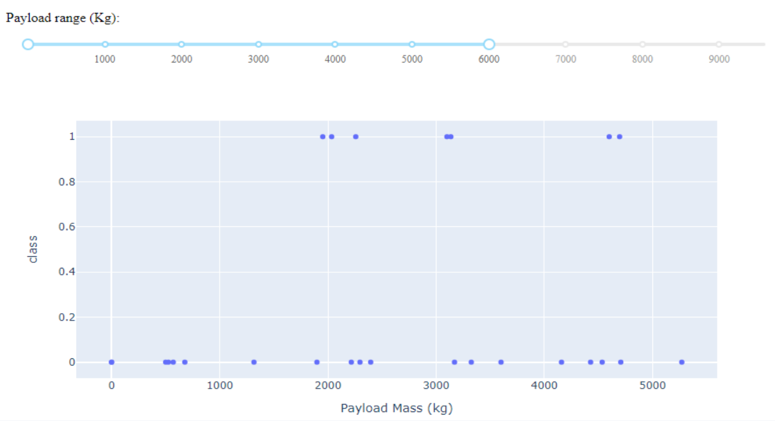
KSC LC-39A



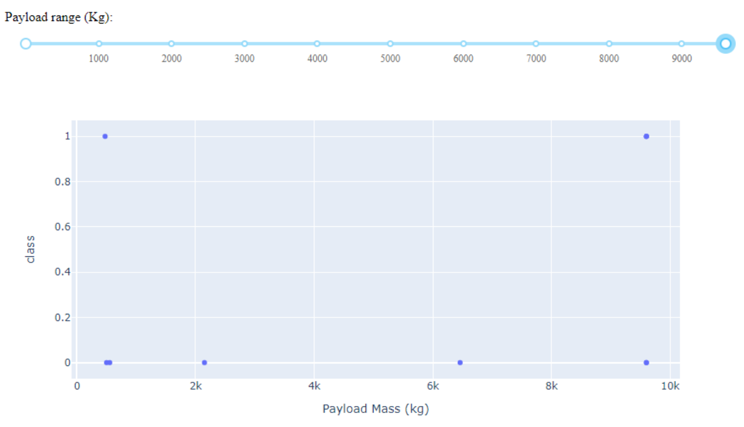
CCAFS SLC-40



CCAFS LC-40



VAFB SLC-4E



For KSC LC-39A and for CCAFS SLC-40 the success rate highly depends on the payload

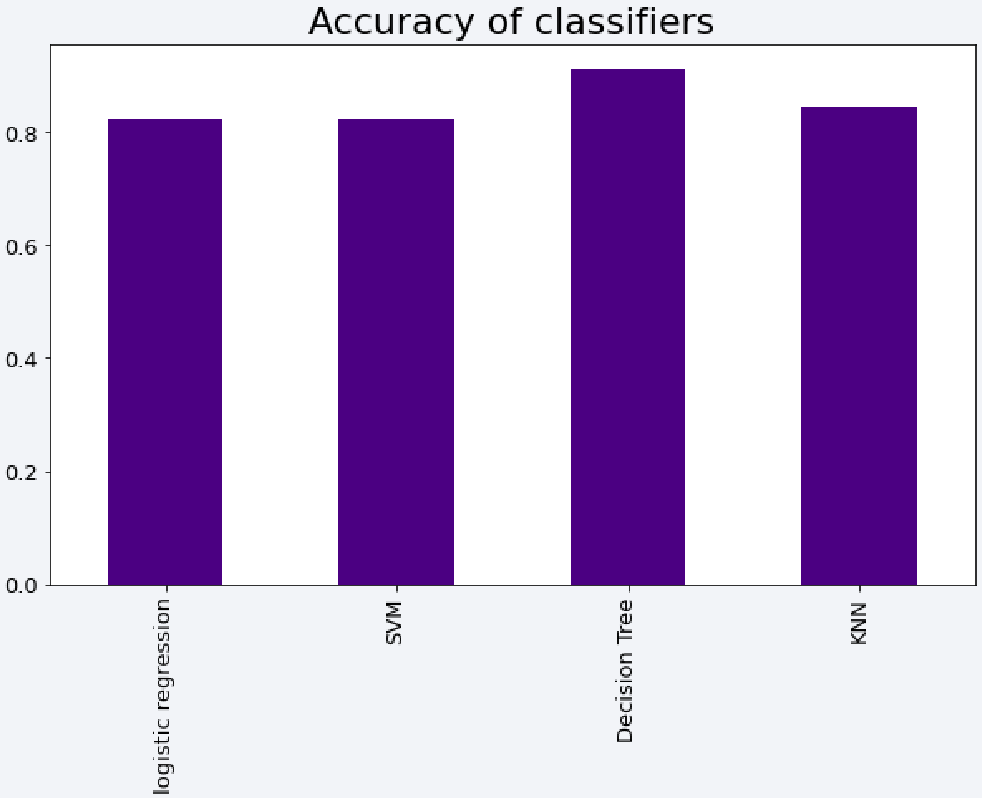
For the other launch sites payload does not seem to be the outstanding problem.

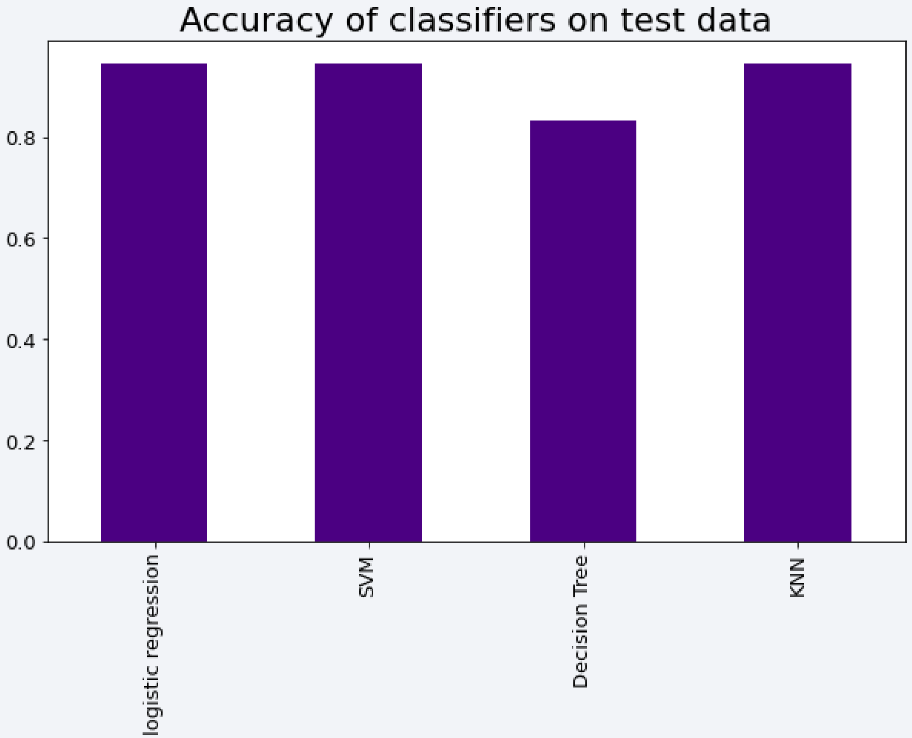
Note: For VAFB SLC-4E not all points are visible due to overlapping payload values

Section 5

Predictive Analysis (Classification)

Classification Accuracy

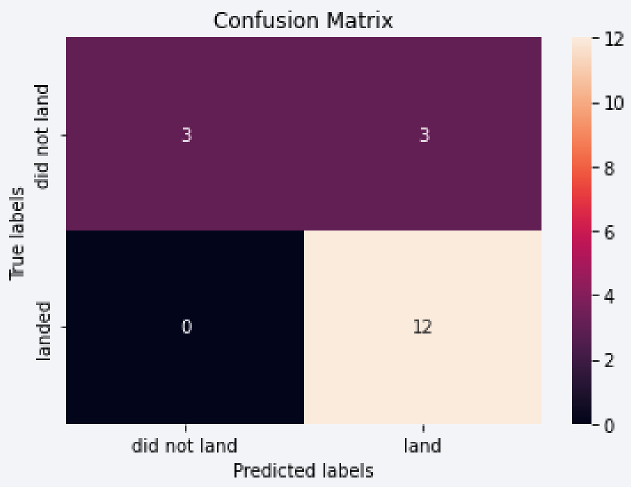


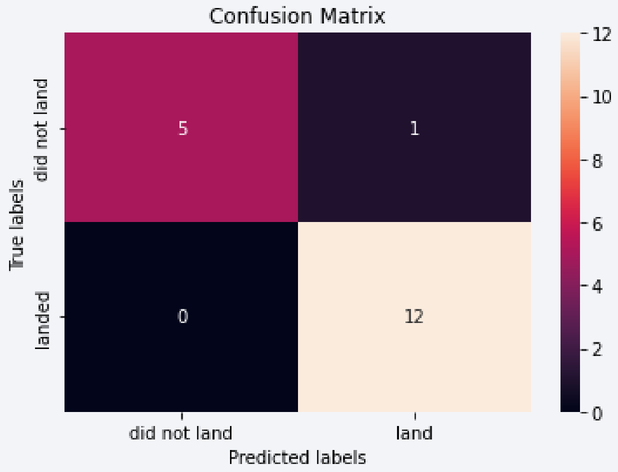


According to GridSearchCV. best_score_ The DecisionTree Classifier performs best, but using the score method on the test data the others are better.

- Find which model has the highest classification accuracy

Confusion Matrix





The confusion matrix for the

Decision Tree Cassifier:

Three records of the 18 records of the test set were missplaced:

They were predicted to land, but did not land.

The confusion matrix for the other three classifiers show that only 1 record of the 18 records of the test set was missplaced.

It was predicted to land, but did not land.

Conclusions

- A 100% percent reliable prediction on whether a launch will successfully land is not possible
- For a large number of planned launches, it is possible to predict an overall probability of the mission
- Visual plots help a lot in finding the relevant features and dependencies
- Preprocessing of the data is crucial for the success of each data science project

Appendix

- All notebooks and python files can be found here:
- https://github.com/SylviaMaczey/courseraClass

Thank you!