IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Muhammad Azis Rizaldi
16th August 2023

Outline

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

Executive Summary

The workflow for gaining insights from SpaceX API data collection has long process include data collection, data wrangling, exploratory data analysis using visualization or sql query, plotting on the Dashboard, and make the predictive model analysis.

On data wrangling, we will filtering the data so we can have the beneficial information. Exploratory data analysis is useful to seek the hiden information. The best method for perform that hiden information is plotting the graph on the Dashboard. It give us simple, genuine and intuitive information for user or stakeholder.

We also purpose some classification models such as Logistic linear, Support Vector Machine, Decision Tree, and K-Nearest Neighbor. We will train the dataset and looking for the model which has the maximum accuracy. Determine the model will help us to dealing with dataset in the future.

Introduction

SpaceX advertises Falcon 9 rocket launches on its website with a cost of 62 million dollars. Many providers cost upward 165 million dollars each. It is because SpaceX can reuse the first stage.

We want to determine whether the first stage of Falcon 9 will land successfully or not by make predictions based on the training data which provided by SpaceX. In addition, we will gain some insights with Exploratory Data Analysis which displays in Dashboard.

Section 1

Methodology

Methodology

Executive Summary

The data was collected by request to SpaceX API (json). We will translate the text json context to the dataframe by applied some functions or criteria just to filtering the beneficial information.

The dataframe that we have still has some inappropriate format for further processing. Since we want to do classification, we will translate the value in “outcome” coloumn into binary value.

We can use charts visualization and SQL query for gaining the core information and correlation among the data.

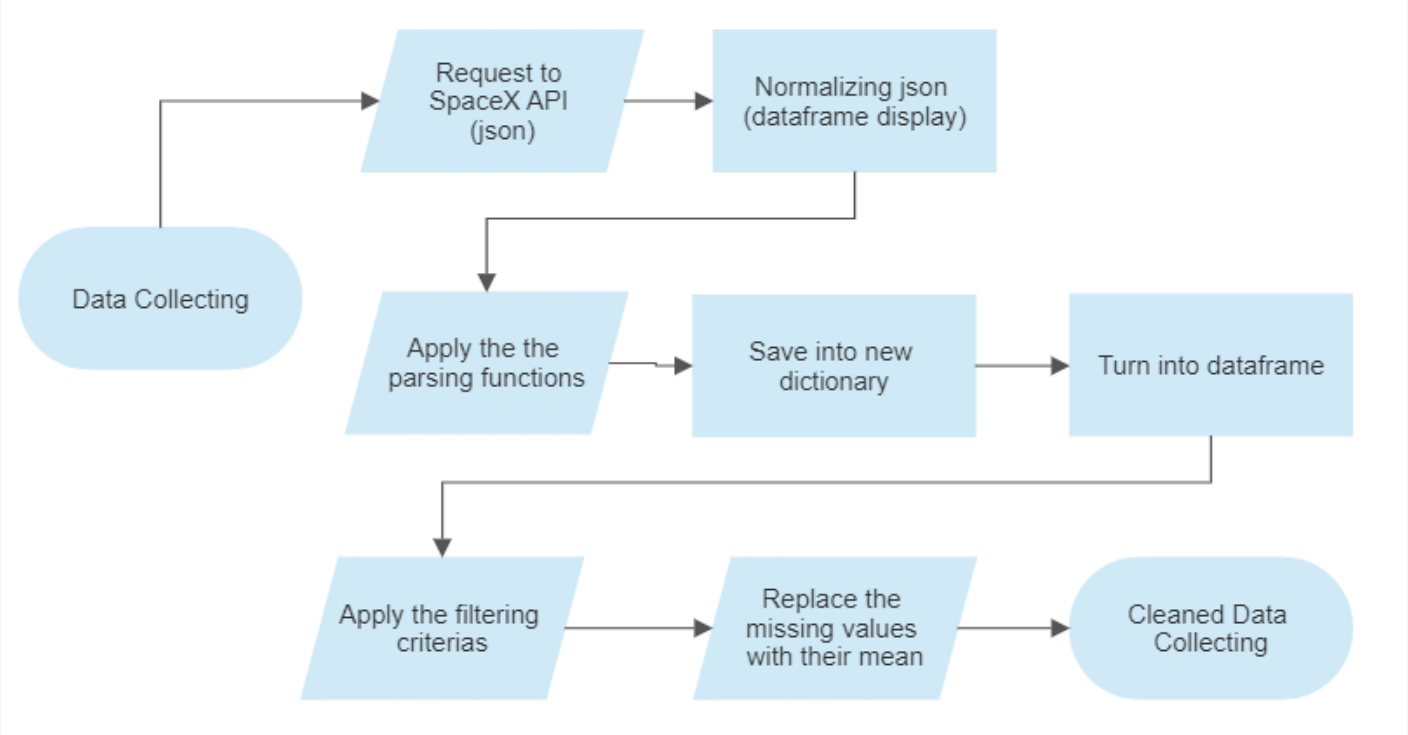
Interactive visualization analytics based on website like multiple charts with dropdown, and range slider features offer the user or stakeholder to get information and make decision in real time.

Making the model will be very beneficial for dealing the data in the future. So, we will purpose some models, training the data, and see the accuracy for testing data. At the end, we choose the model which has maximum accuracy.

Data Collection

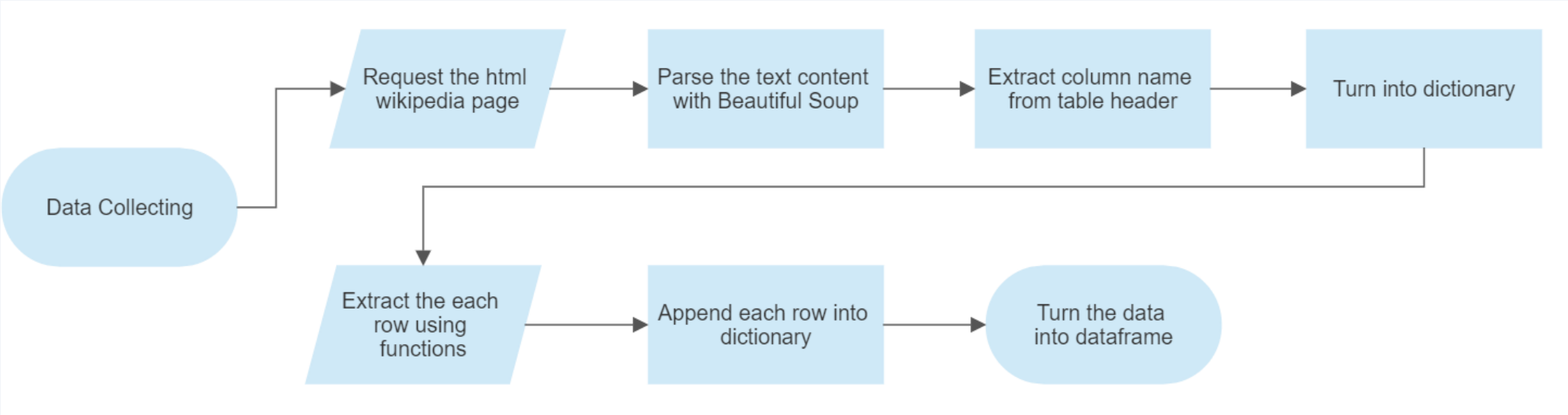
The data is collected by request to SpaceX API or using the web scrapping BeautifulSoup. For sure, we have to filtering or cleaning the collected data for better result in the next step of data processing.

Data Collection – SpaceX API



https://github.com/geez01/Python-DS-Project/blob/main/jupyter-labs-spacex-data-collection-api%20(1).ipynb

Data Collection - Scraping

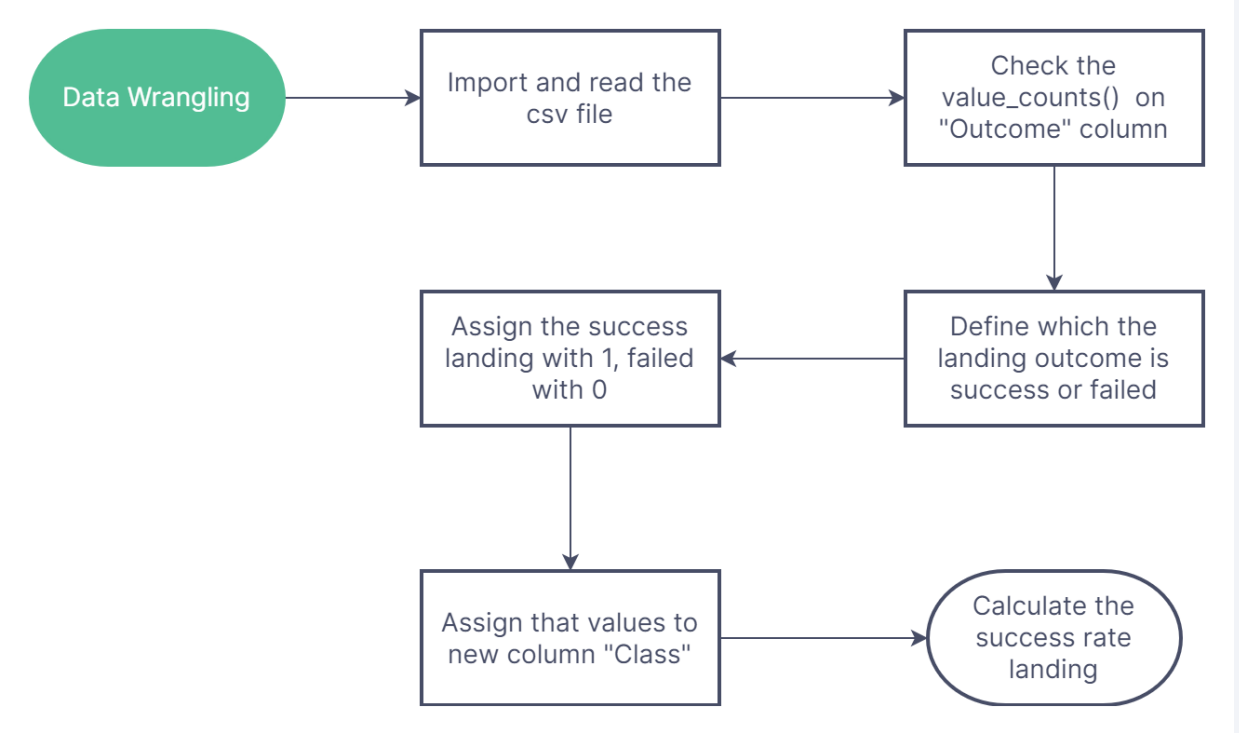


https://github.com/geez01/Python-DS-Project/blob/main/jupyter-labs-webscraping.ipynb

Data Wrangling

First, we can figure out the information on each column by the value_counts() method. After we know some information, we can do the data wrangling. In this case, we will add the dataframe column “Class” by assign the success/failed landing outcome with binary value: 0 or 1

Data Wrangling



https://github.com/geez01/Python-DS-Project/blob/main/labs-jupyter-spacex-data_wrangling_jupyterlite.jupyterlite.ipynb

EDA with Data Visualization

There are three type of charts that we will use for EDA:

- **Scatter point chart**: to find the relationship between two variables in the SpaceX Falcon 9 dataframe and which value they are fall into (in this case, success or failed outcome).

- **Bar chart**: to know the mean of success rate of each value in a column (in this case, for each orbit)

- **Line chart**: to find the trend of success rate in the timeline (years)

https://github.com/geez01/Python-DS-Project/blob/main/jupyter-labs-eda-dataviz.ipynb.jupyterlite.ipynb

EDA with SQL

- There are four launch site. They are CCAFS LC-40, VAFB SLC-4E, KSC LC-39A, CCAFS SLC-40
- The first successful landing in ground pad was on 2015-12-22
- The booster name which have success in drone ship and payload mass between 4000 and 6000 are F9 FT B1022, B1026, B1021.2, B1031.2
- The 5th rank by the count of landing outcome between 2010-06-04 and 2017-03-20 are no attempt (10), success on ground pad (5), success on drone ship (5), failure on drone ship (5), controlled on ocean (3)

https://github.com/geez01/Python-DS-Project/blob/main/jupyter-labs-eda-sql-coursera_sqllite%20(1).ipynb

Build an Interactive Map with Folium

There are some folium map object that we use:

- **Circle**: to mark the launch site area. We can customize the radius, the color fill, and popping up the name of the launch area
- **Marker**: to mark the name of the launch site within the circle
- **Marker cluster**: to mark the success and failed outcomes either on the same launch site area
- **Mouse position**: to get the coordinates just by hover the mouse position
- **Lines**: to connect two position with the line

https://github.com/geez01/Python-DS-Project/blob/main/lab_jupyter_launch_site_location.jupyterlite%20(1).ipynb

Build a Dashboard with Plotly Dash

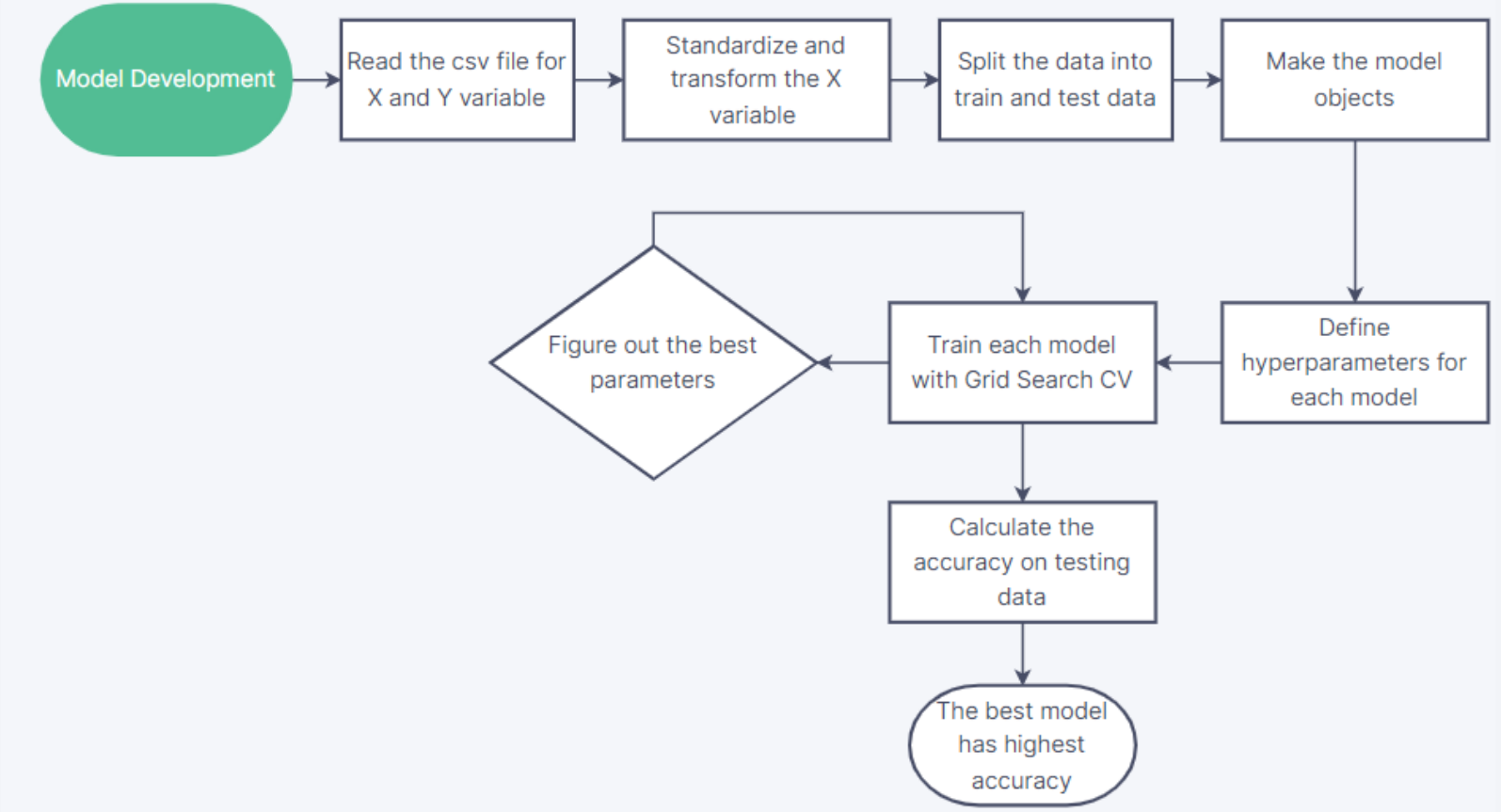
Graphics that we use in Dasboard are Pie chart and scatter chart. Interaction features that we use are Dropdown and Rangeslider

Pie and scatter chart will be showed automatically after user choosing the option on dropdown and range slider. Those interactions will give information or insights efficiently and straight-forward.

Pie chart is used for showing the percentage of successful landing on all or each site, whereas scatter chart is used for showing the relationship between the variables and what value is falling into them.

https://github.com/geez01/Python-DS-Project/blob/main/my_dash.py

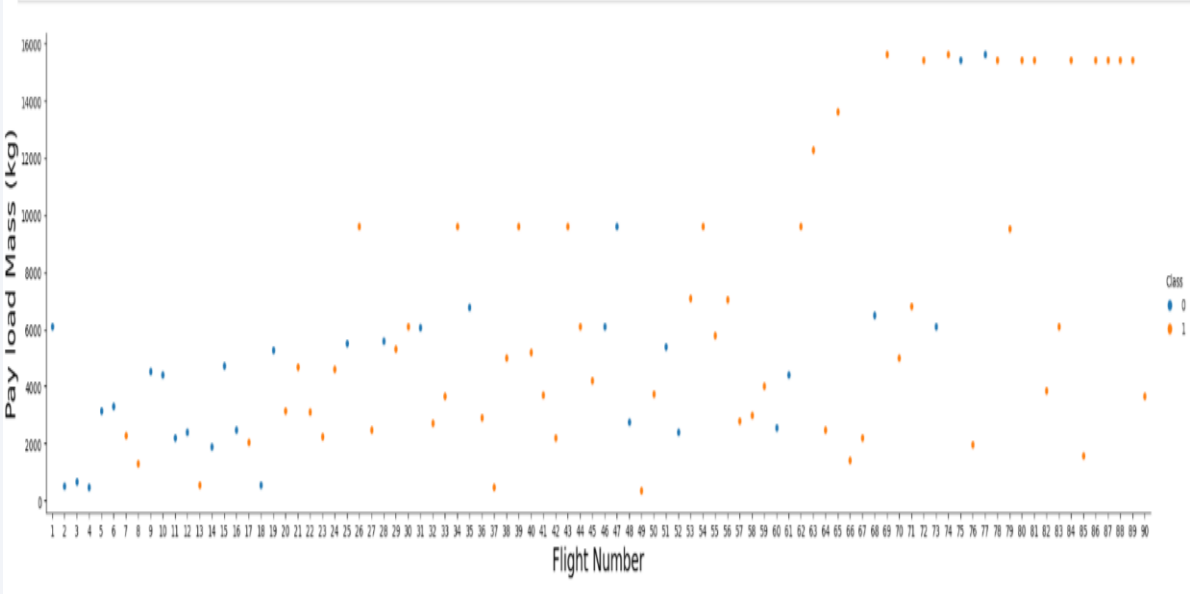
Predictive Analysis (Classification)

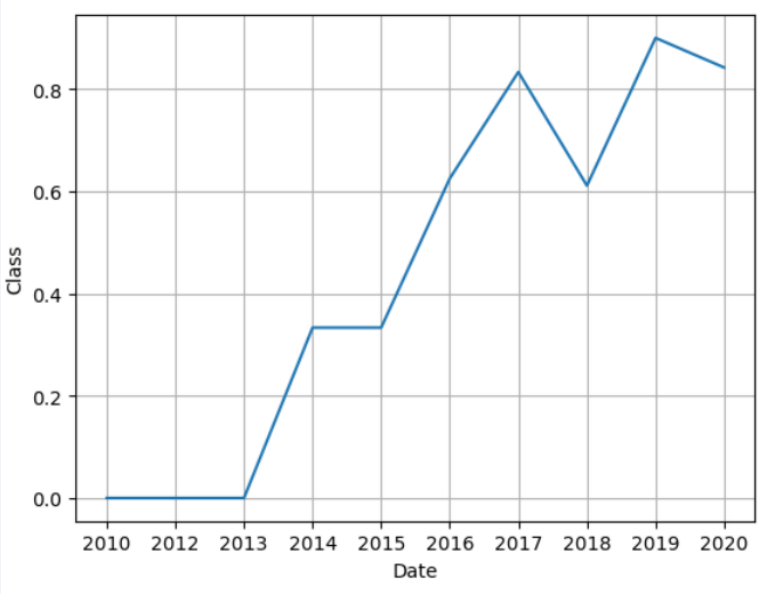


https://github.com/geez01/Python-DS-Project/blob/main/SpaceX_Machine_Learning_Prediction_Part_5.jupyterlite%20(1).ipynb

Results

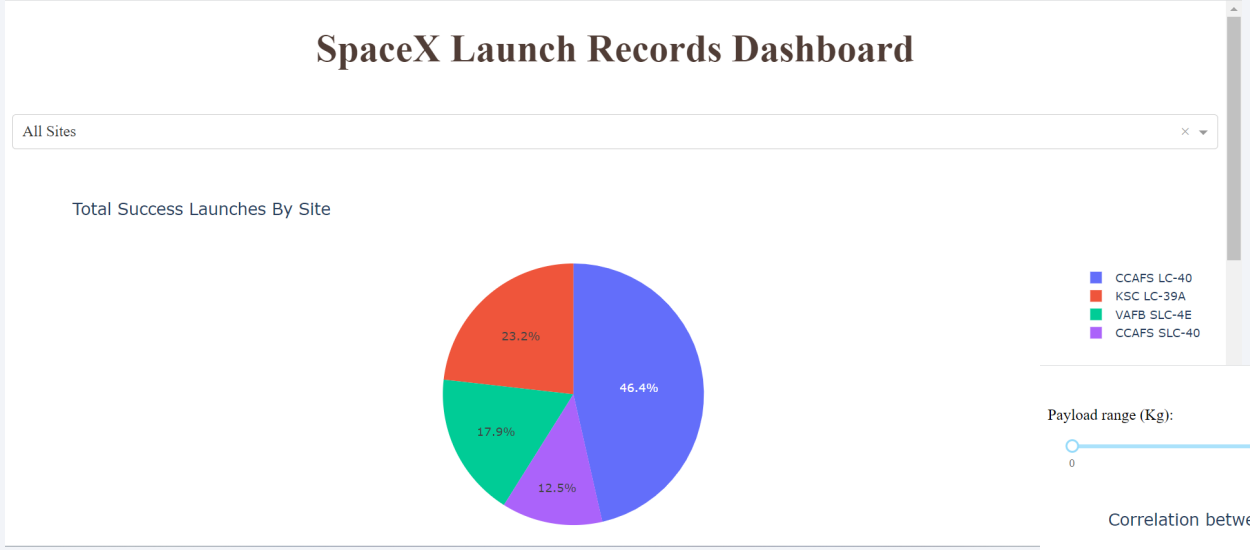
For the beginning of the launch, we face the common failure landing. But more often the flight be conducted, the probability for success landing will be clear. It is proven by the line graphic over the years show the escalation success landing trend.

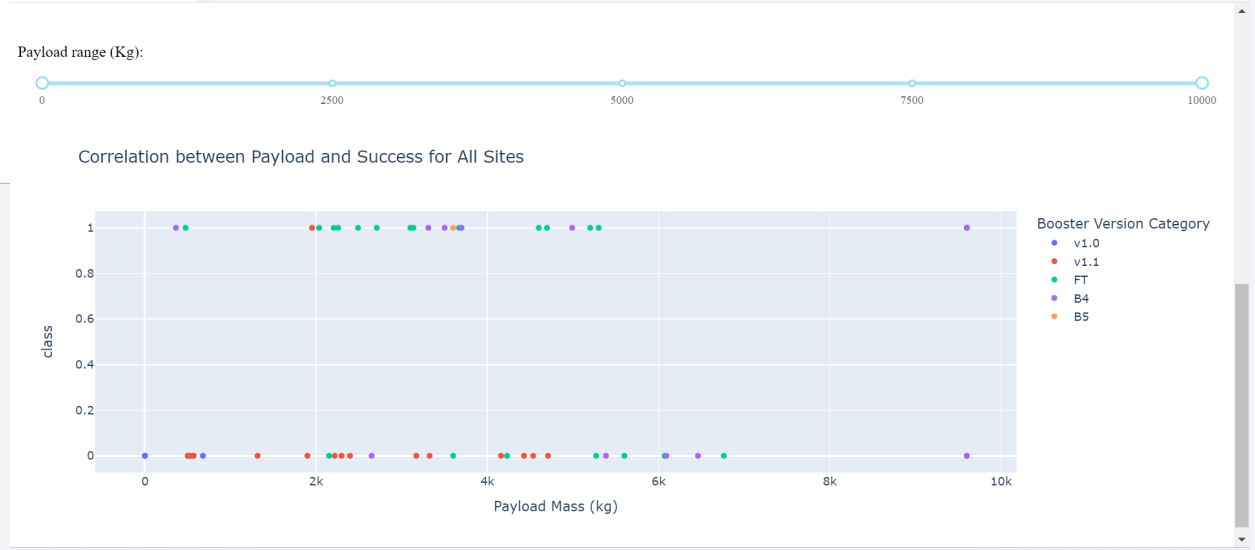




Results

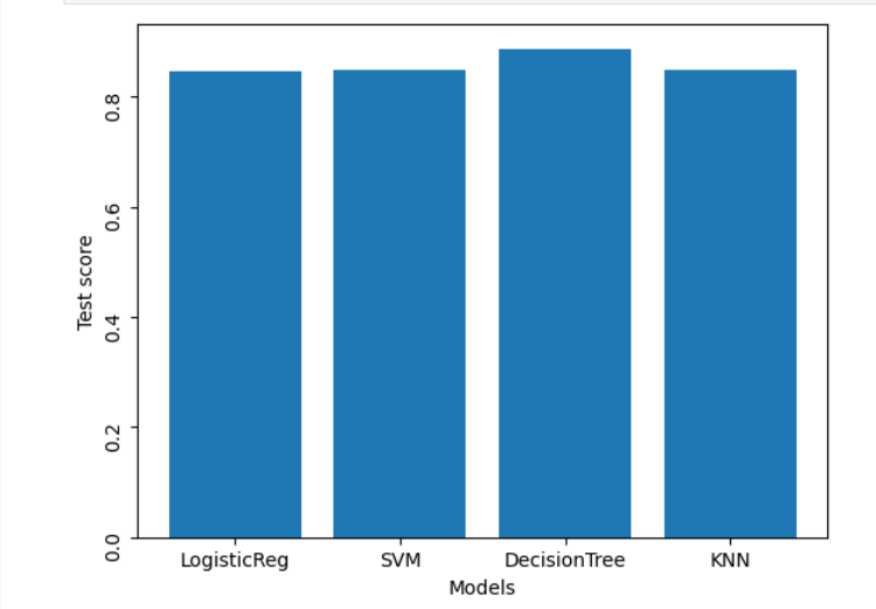
Based on Dashboard, CCAFS LC-40 has the highest success landing rate. More heavy the payload mass that being carried by rocket, more high risk to be failed too.

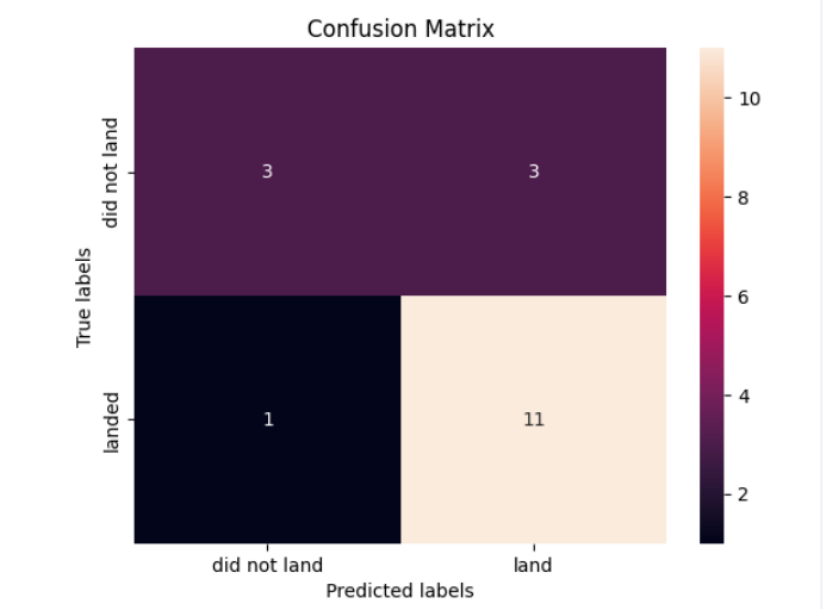




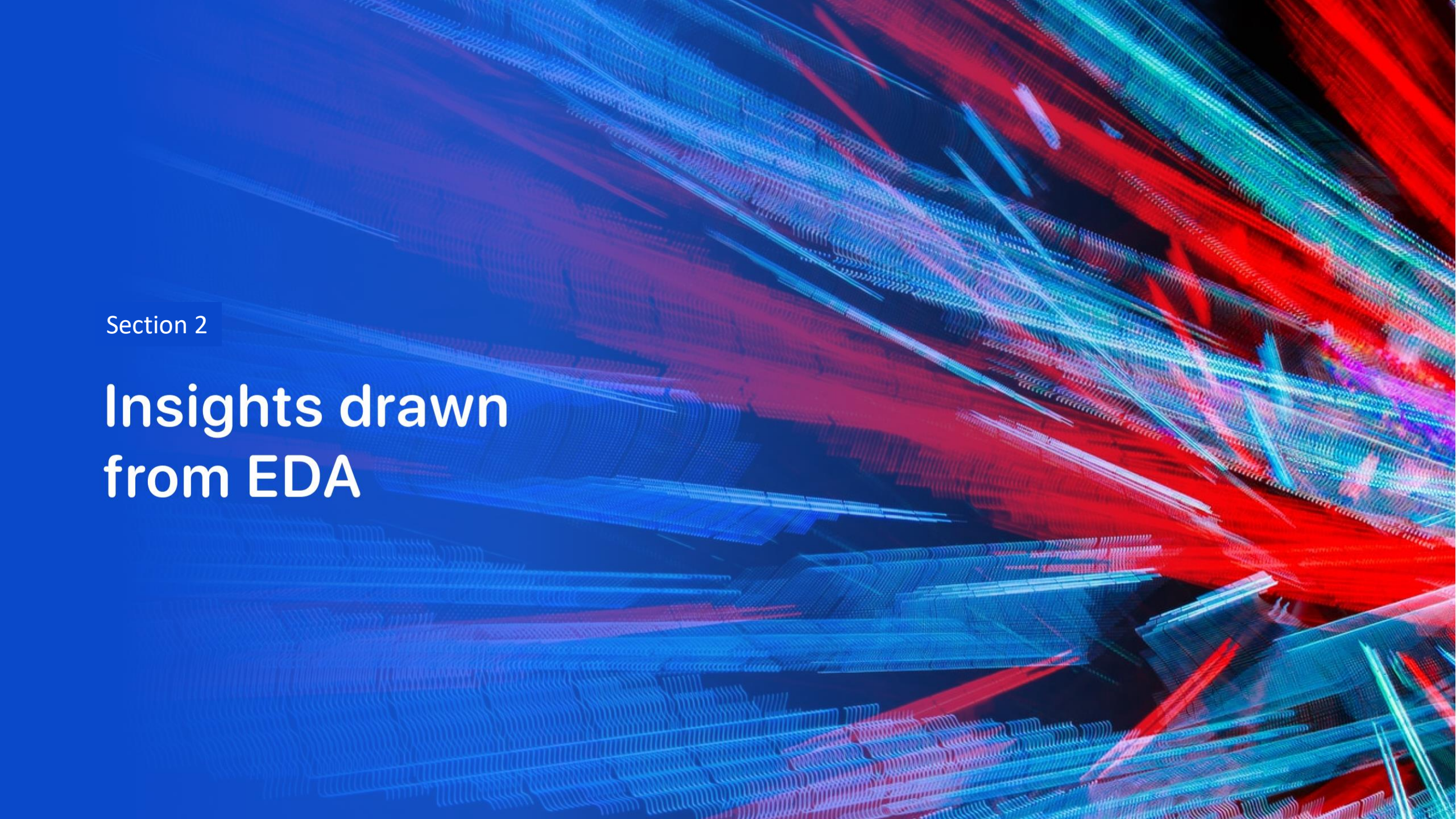
Results

The best model for classification is Decision Tree which has the accuracy 0.8875. Although the accuracy on testing data is not satisfy, because it is depend on the size of training and testing data.



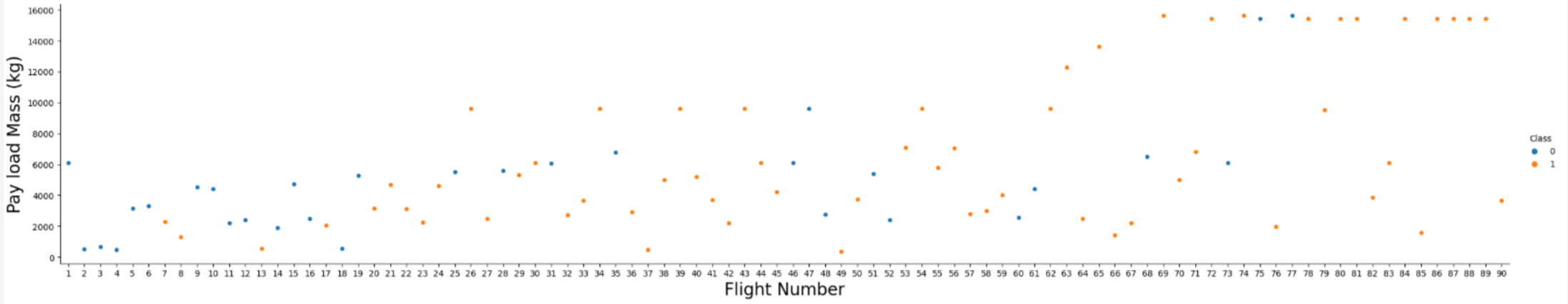


The model will be used for determining the success of Falcon 9 landing for the next data collection.


Section 2
Insights drawn from EDA

Flight Number vs. Launch Site

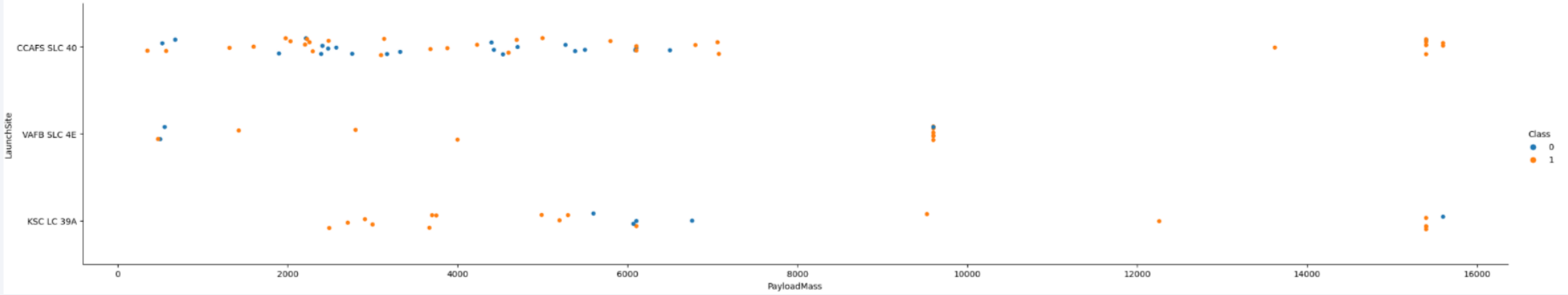
```
sns.catplot(y="PayloadMass", x="FlightNumber", hue="Class", data=df, aspect = 5)
plt.xlabel("Flight Number",fontsize=20)
plt.ylabel("Pay load Mass (kg)",fontsize=20)
plt.show()
```



For the beginning flight number, it seems has high failed landing outcome. As flight number increasing as payload mass too, it tends to give high successful landing outcome.

Payload vs. Launch Site

```
# Plot a scatter point chart with x axis to be Pay Load Mass (kg) and y axis to be the launch site, and hue
sns.catplot(data=df, x='PayloadMass', y='LaunchSite', hue='Class', aspect=5)
plt.show()
```

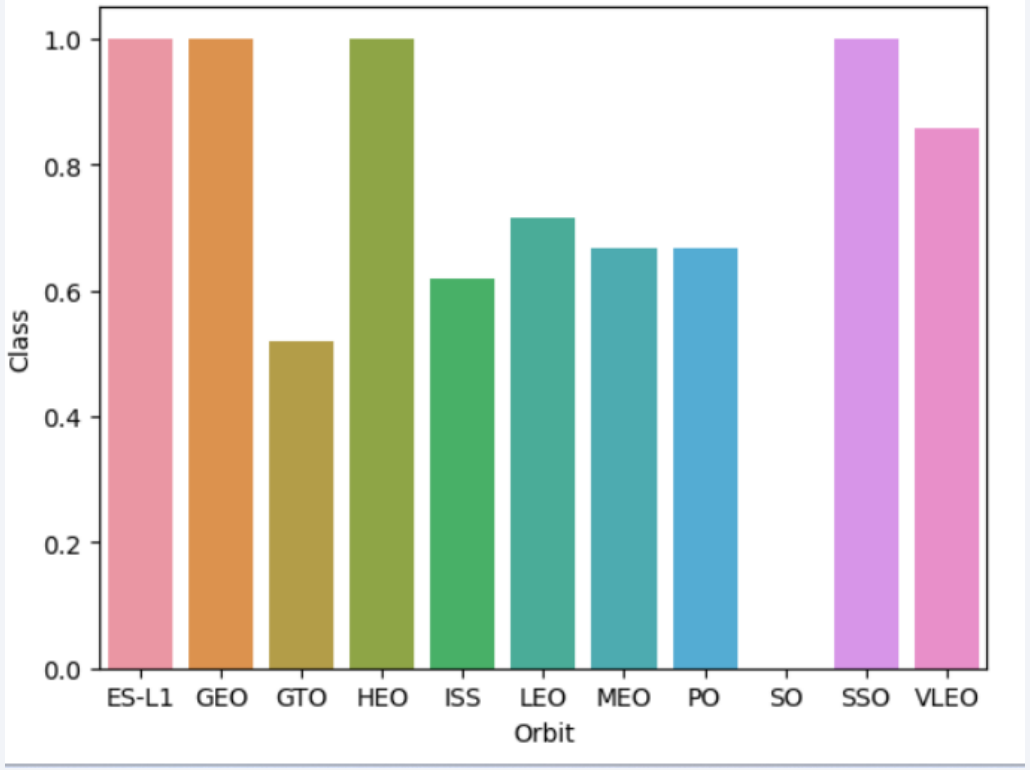


For different launch site, it may have or not correlation between payload mass and successful landing outcome. **VAFB SLC 4E** has payload mass lesser than 10000 and it gives higher successful rate. **CCAFS SLC-40** absolute correlation between payload mass and its outcome. **KSC LC 39A** has high successful rate when carries lighter payload mass.

Success Rate vs. Orbit Type

```
# HINT use groupby method on Orbit column and get the mean of Class column
orbit_success_rate = df.groupby('Orbit')['Class'].mean().reset_index()

sns.barplot(data=orbit_success_rate, x='Orbit', y='Class')
plt.show()
```

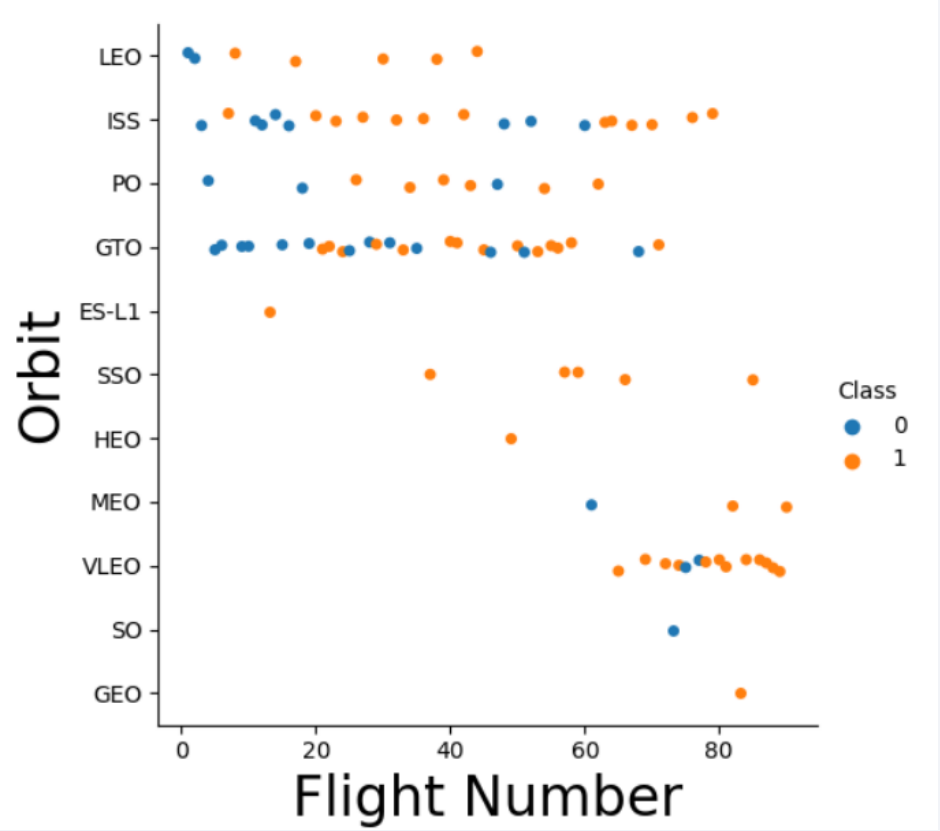


Each orbit has own successful landing rate. The most higher rate is GEO, ES-L1, HEO, and SSO.

They are have different radius orbit and the sync type. So, the radius orbit and sync type doesn't have correlation for successful landing.

Flight Number vs. Orbit Type

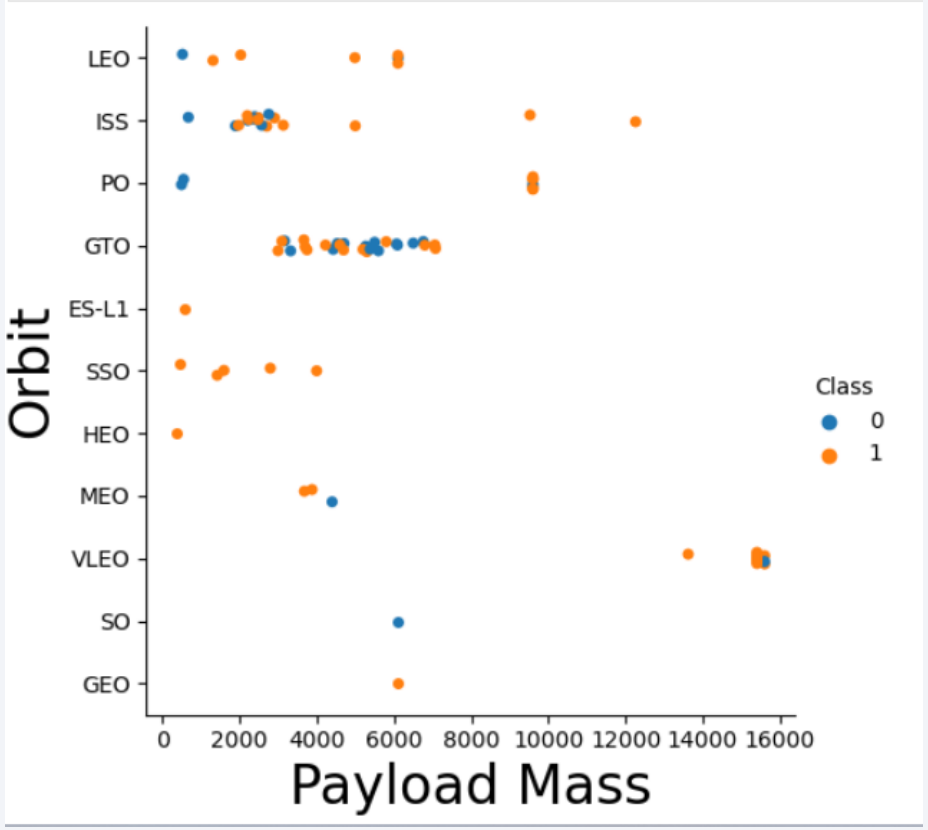
```
sns.catplot(data=df, x='FlightNumber', y='Orbit', hue='Class')
plt.xlabel("Flight Number", fontsize=24)
plt.ylabel("Orbit", fontsize=24)
plt.show()
```



Whatever the type of orbit, there is dominant successful landing outcome when flight number increasing.

Payload vs. Orbit Type

```
sns.catplot(data=df, x='PayloadMass', y='Orbit', hue='Class')
plt.xlabel("Payload Mass", fontsize=24)
plt.ylabel("Orbit", fontsize=24)
plt.show()
```

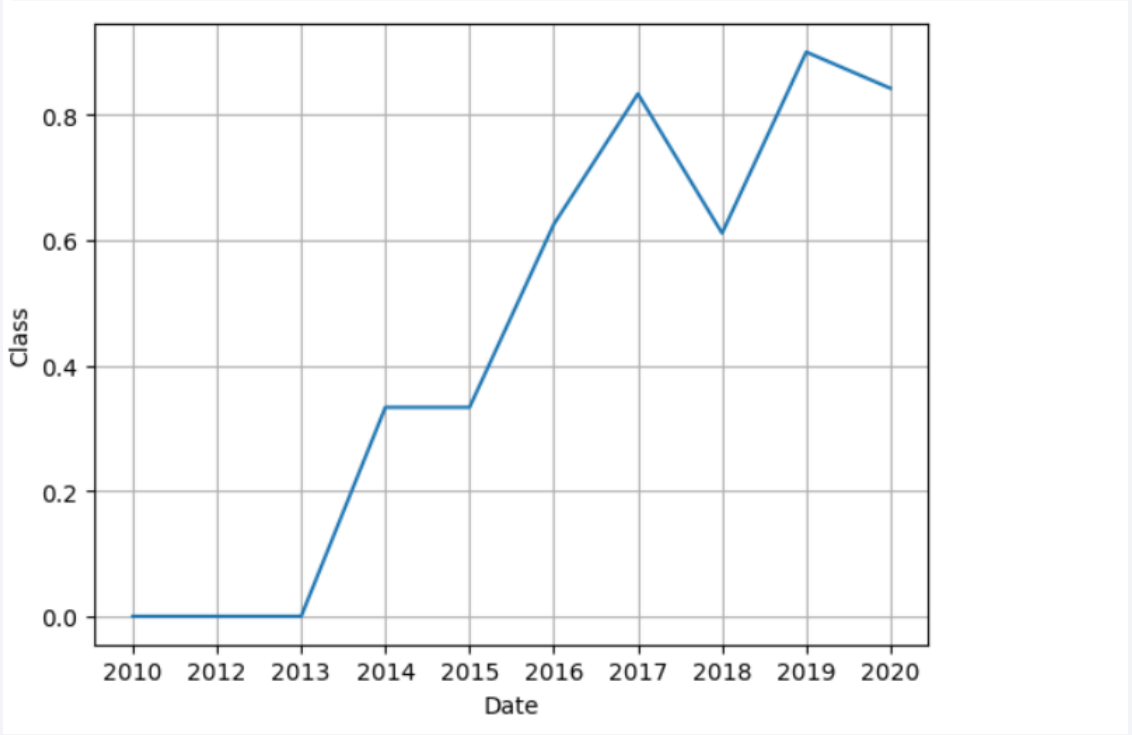


For GTO and ISS orbit there is no conclusion at some range of payload mass and successful landing. For others, the increasing payload mass doesn't impact for the failed landing outcome.

Launch Success Yearly Trend

```
# Plot a line chart with x axis to be the extracted year and y axis to be the success rate
year_success_rate = df.groupby('Date')['Class'].mean().reset_index()

sns.lineplot(data=year_success_rate, x='Date', y='Class')
plt.grid()
plt.show()
```



There is significant increasing successful rate after launching in 2013

All Launch Site Names

```
In [8]: %sql SELECT DISTINCT("Launch_Site") FROM SPACEXTABLE
 * sqlite:///my_data1.db
Done.
```

Out[8]:

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

On SpaceX report table, there are four launch site which are CCAFS LC-40, VAFB SLC-4E, KSC LC-39A, CCAFS SLC-40

Launch Site Names Begin with 'CCA'

In [9]:
```
%sql SELECT * FROM SPACEXTABLE WHERE "Launch_Site" LIKE 'CCA%' LIMIT 5
```

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

Out[9]:

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcor
2010-04-06	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachu
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 (parachu
2012-05-22	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No atten
2012-08-10	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No atten
2013-01-03	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No atten

IN CCAFS LC-40, It has earlier launching, light payload mass, and the type of orbit is LEO (Low Earth Orbit)

Total Payload Mass

```
In [10]: %sql SELECT SUM("PAYLOAD_MASS__KG_") FROM SPACEXTABLE WHERE "Customer" = "NASA (CRS)"
 * sqlite:///my_data1.db
Done.
Out[10]: SUM("PAYLOAD_MASS__KG_")
                                45596
```

The launching rocket which is requested by NASA has the total payload mass 45596 kg.

Average Payload Mass by F9 v1.1

```
In [11]: %sql SELECT AVG("PAYLOAD_MASS__KG_") FROM SPACEXTABLE WHERE "Booster_Version" = "F9 v1.1"
 * sqlite:///my_data1.db
Done.
Out[11]: AVG("PAYLOAD_MASS__KG_")
         2928.4
```

Average payload mass kg which carried by F9 v1.1 is 2928.4 kg

First Successful Ground Landing Date

```
In [12]: %sql SELECT MIN(DATE) FROM SPACEXTABLE WHERE "Landing_Outcome" = "Success (ground pad)"
 * sqlite:///my_data1.db
Done.
Out[12]: MIN(DATE)
         2015-12-22
```

The first launching rocket date and has successful landing on ground pad is 22nd December 2015

Successful Drone Ship Landing with Payload between 4000 and 6000

In [13]:

```
%%sql
SELECT "Booster_Version"
FROM SPACEXTABLE
WHERE "Landing_Outcome" = 'Success (drone ship)' AND ("PAYLOAD_MASS__KG_" BETWEEN 4000 AND 6000)
```

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

Out[13]:

Booster_Version
F9 FT B1022
F9 FT B1026
F9 FT B1021.2
F9 FT B1031.2

Booster versions which can carries payload mass between 4000 and 6000 are F9 FT B1022, B1026, B1021.2, B1031.2

Total Number of Successful and Failure Mission Outcomes

```
[16]: %%sql
SELECT COUNT(*) FROM SPACEXTABLE
WHERE "Mission_Outcome" =  "Success" AND "Landing_Outcome" LIKE 'Failure%'
```

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

[16]:

COUNT(*)
10

There are 10 occurrence that has success launch but failure landing

Boosters Carried Maximum Payload

In [17]:
```
%%sql
SELECT "Booster_Version"
FROM SPACEXTABLE
WHERE (SELECT MAX("PAYLOAD_MASS__KG_") FROM SPACEXTABLE)
```

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

Out[17]:

Booster_Version
F9 v1.0 B0003
F9 v1.0 B0004
F9 v1.0 B0005
F9 v1.0 B0006
F9 v1.0 B0007
F9 v1.1 B1003
F9 v1.1
F9 v1.1
F9 v1.1
F9 v1.1
F9 v1.1
F9 v1.1 B1011
F9 v1.1 B1010
F9 v1.1 B1012
F9 v1.1 B1013

Many of booster version can carries maximum payload mass

2015 Launch Records

In [16]:
```
%%sql
SELECT "Booster_Version", "Launch_Site" FROM SPACEXTABLE
WHERE "Landing_Outcome" = 'Failure(drone ship)' AND substr(Date,7,4) = '2015'
```

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

Out[16]:

Booster_Version	Launch_Site

There is no record in 2015 and has failure landing outcome on drone ship

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

In [17]:
```
%%sql
SELECT "Landing_Outcome", COUNT(*) AS 'COUNT' FROM SPACEXTABLE
WHERE substr(Date,1,4)||substr(Date,6,2)||substr(Date,9,2)
    BETWEEN '20100604' AND '20170320'
GROUP BY "Landing_Outcome"
ORDER BY "COUNT" DESC
```

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

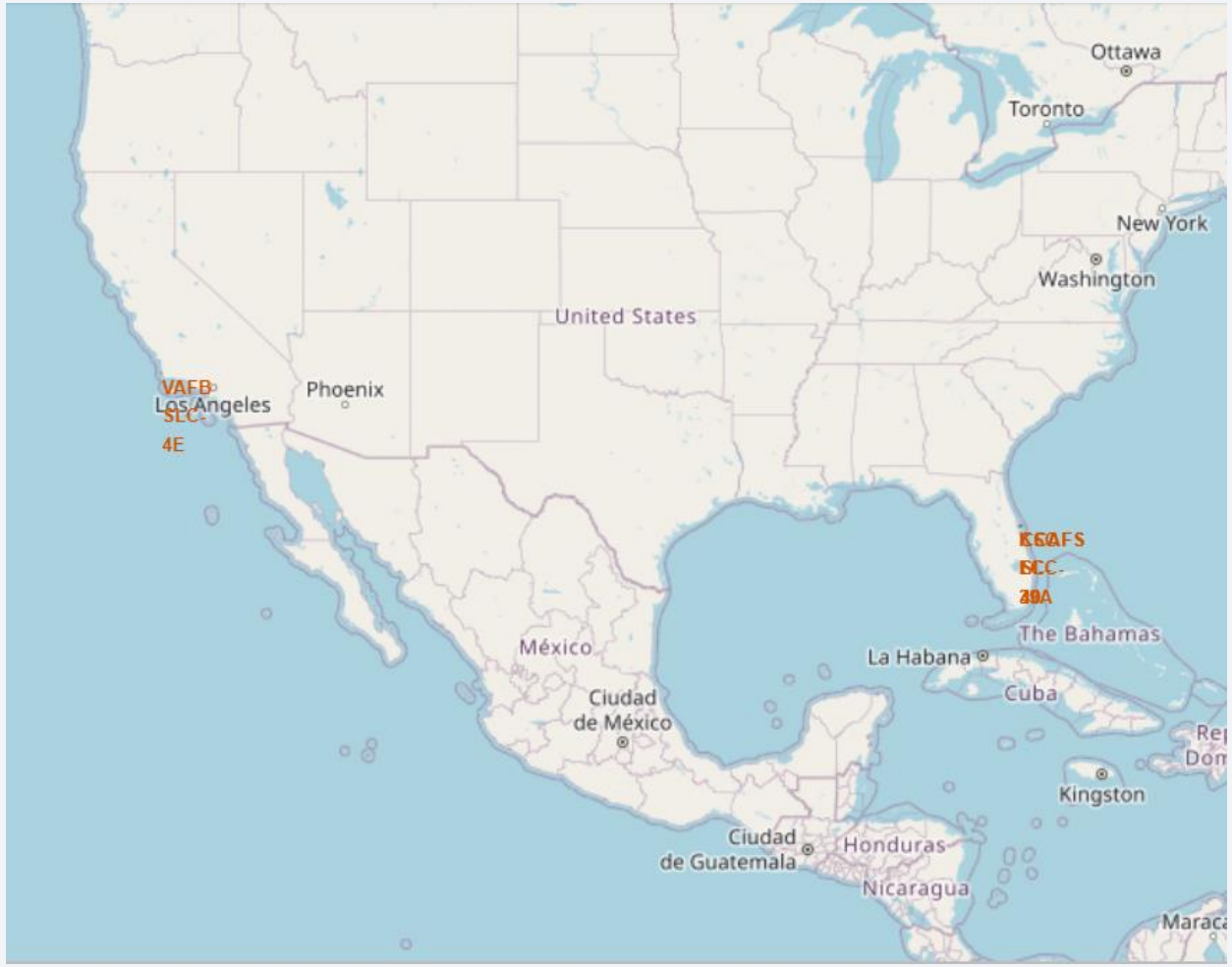
Out[17]:

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

There are 10 no attempt record,
5 success record on ground pad,
5 success record on drone ship,
5 failure record on drone ship,
and so on.


Section 3
Launch Sites
Proximities Analysis

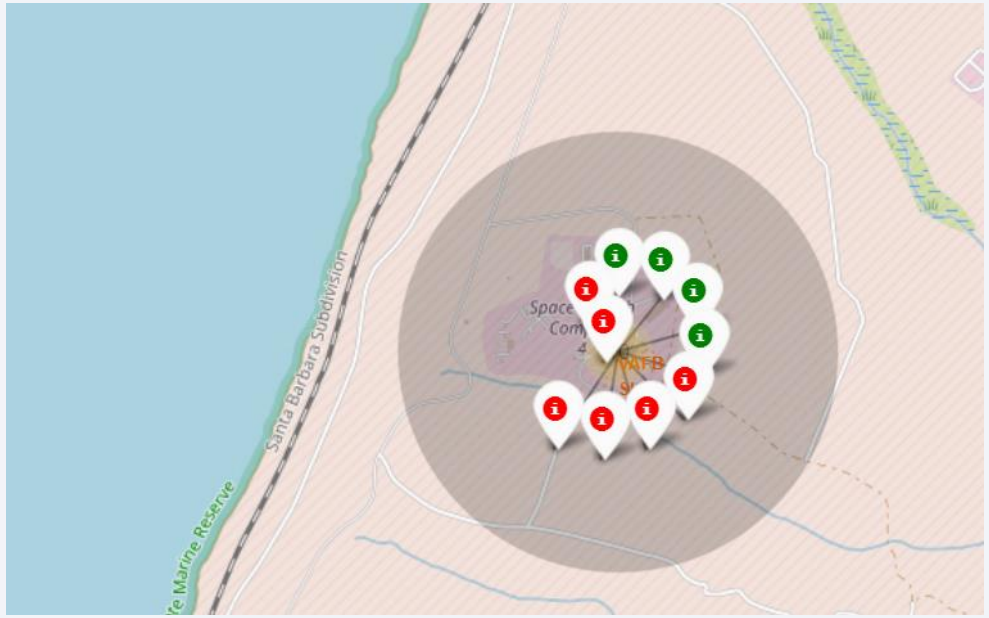
Mark all launch sites on a map

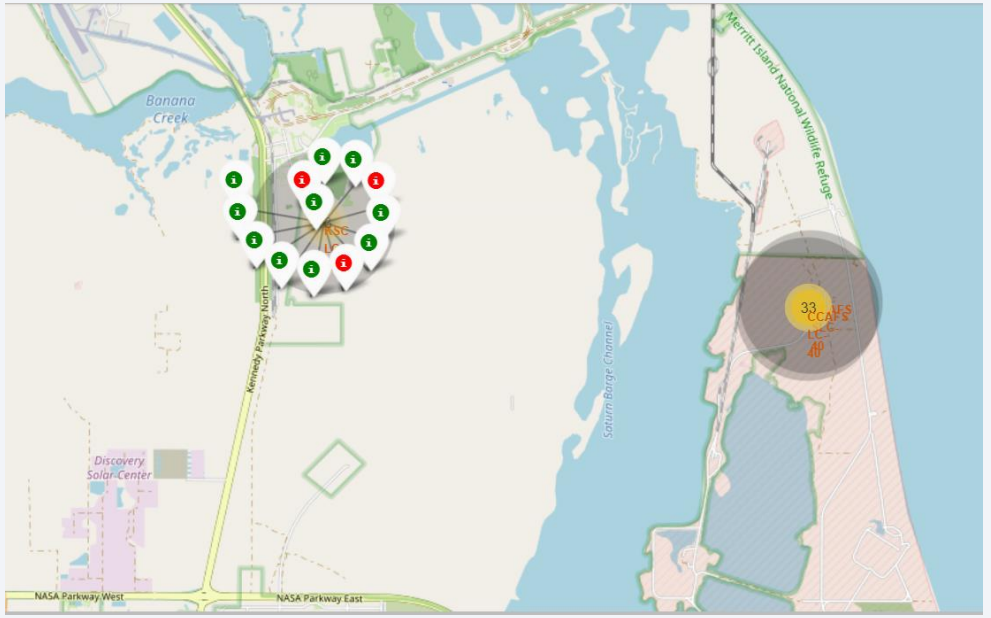


Mark All Launch Sites on a Map

All the launch site are near the coastline, either eastern or western USA. VAFB SLC-40 is on the western, whereas KSC LC-39A, CCAFS LC-40, and CCAFS SLC-40 are on the eastern. CCAFS LC-40, and CCAFS SLC-40 are not 1000m apart.

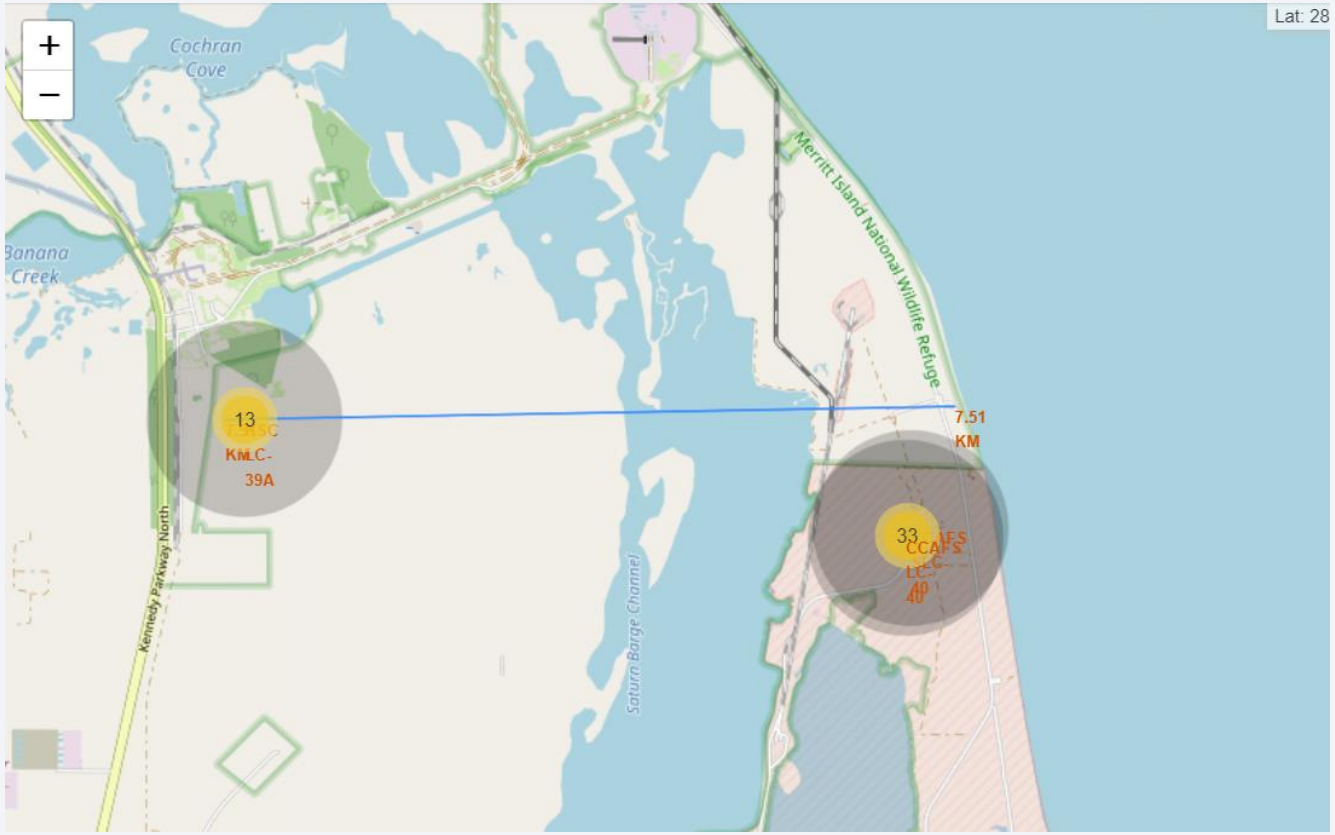
Mark the success/failed launches for each site





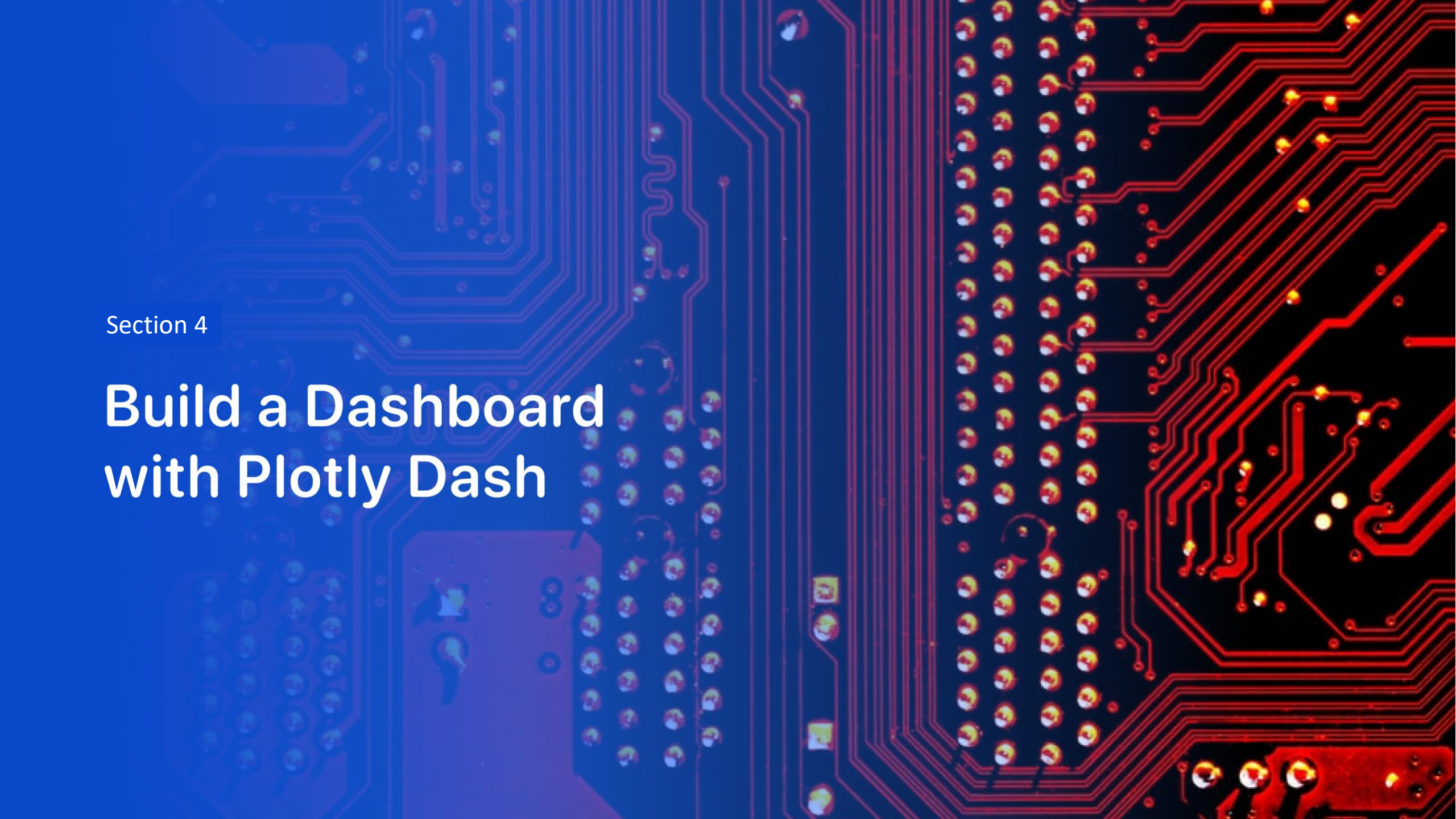
There are mark cluster on each launch site which represented the number of launching. As we zoom in, it is showed the number success with green pin or failed occurrence with red pin.

Calculate the distance from a map

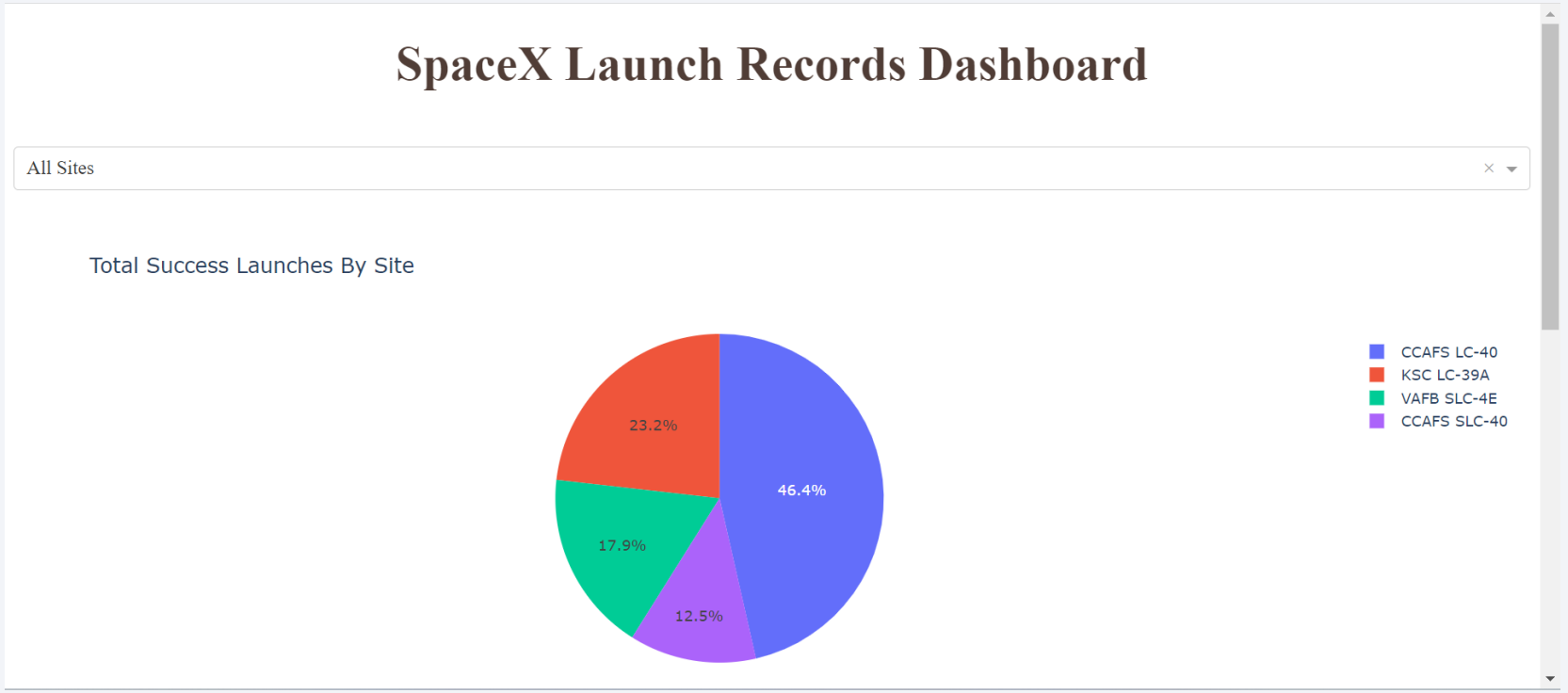


We can count the distance by knowing the launch site and coastline coordinate and apply them to the mathematical function.

There is a close distance between launch site and coastline may close than 7 km.

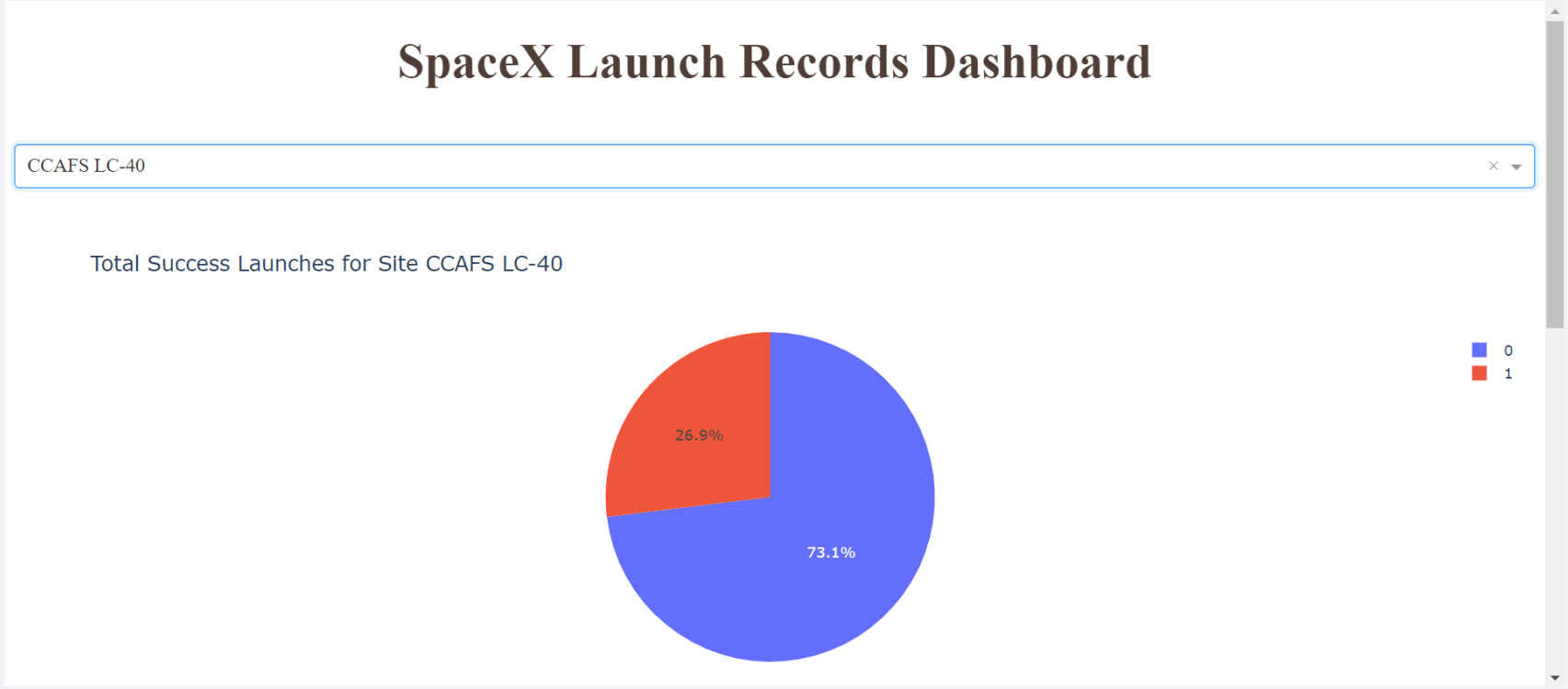

Section 4
Build a Dashboard with Plotly Dash

Overall Successful Launches for All Sites



If we compare each launch site, CCAFS LC-40 has greatest successful landing rate among others

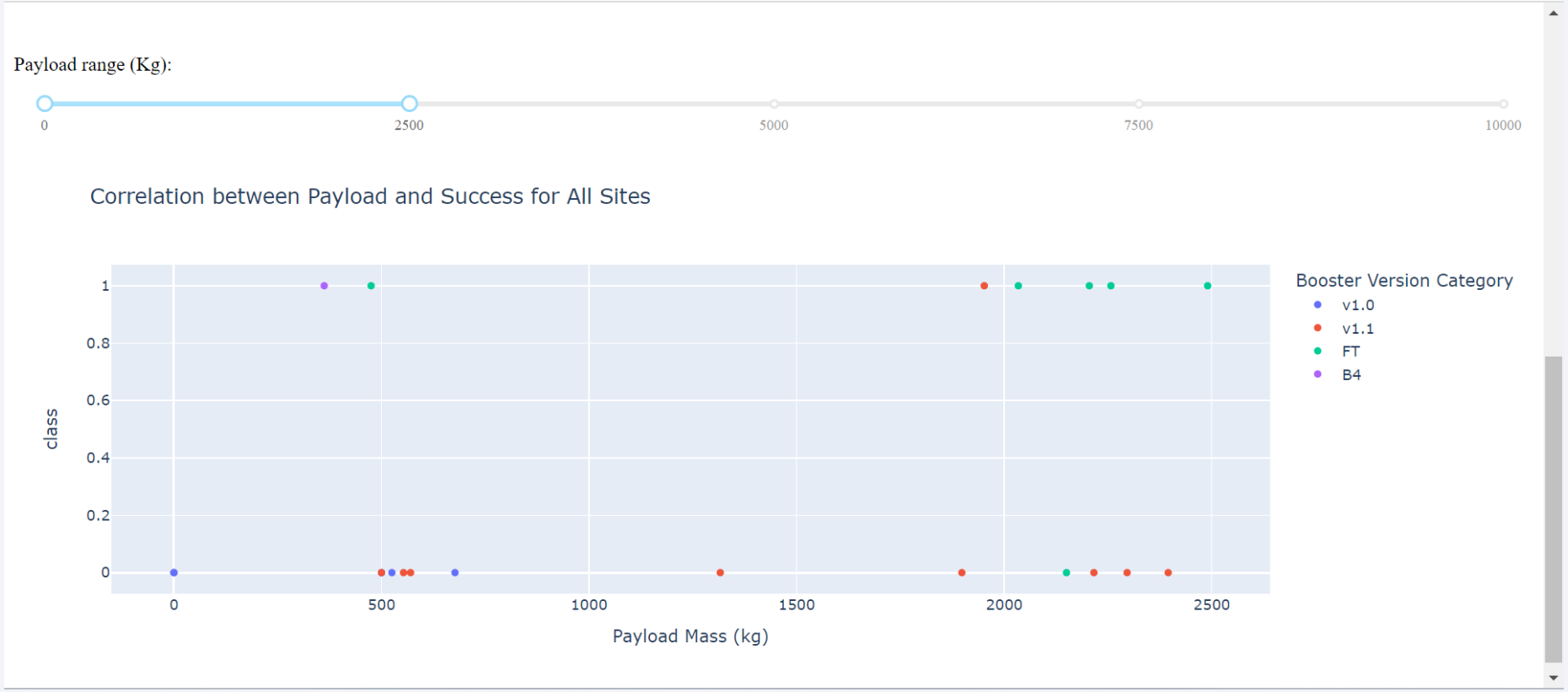
The percentage of successful on CCAFS LC-40



If we focus at CCAFS LC-40, we found that only 26.9% successful launch.

We can conclude each of other launchsite will have lower successful launch rate.

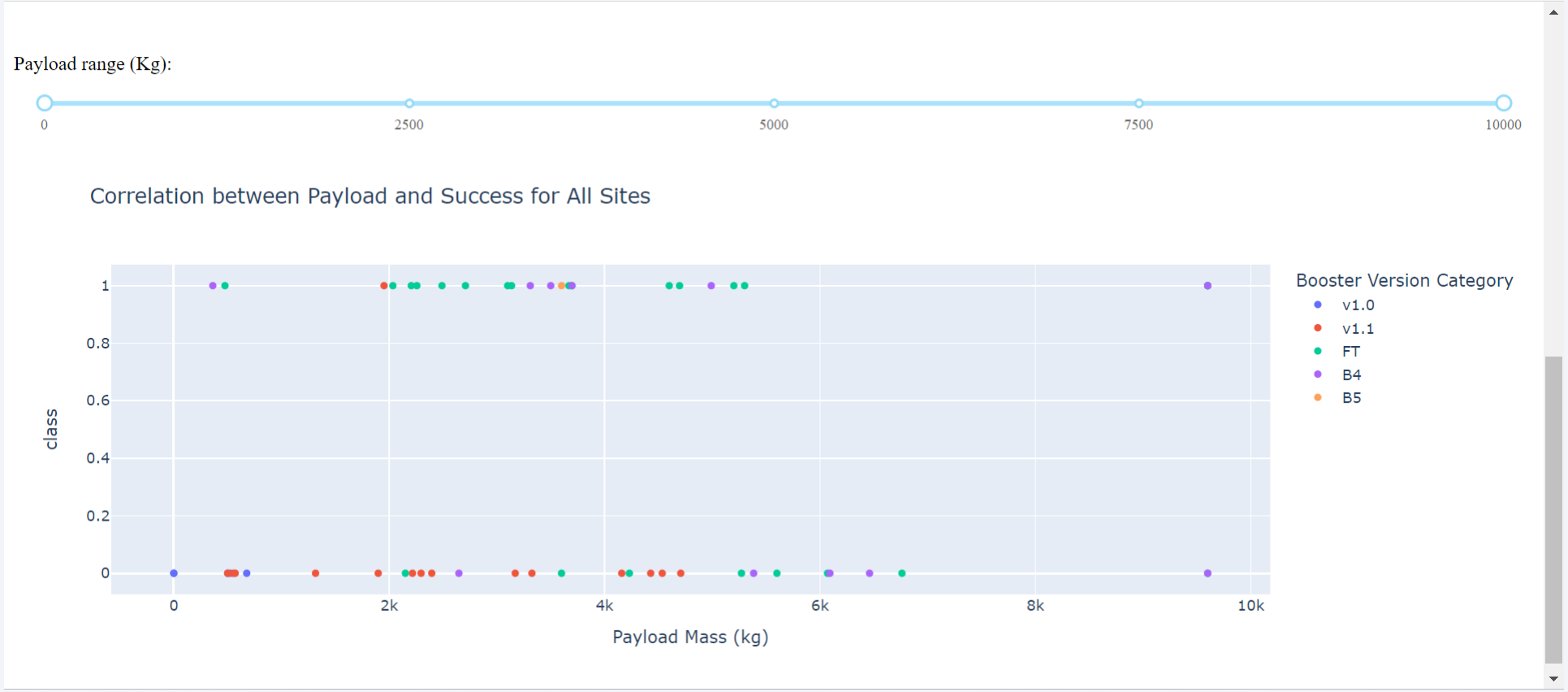
Payload Mass in range 0-2500 vs class



Payload Masss in range 0-5000 vs class

Payload Mass in range 0-7500 vs class

Payload Mass in range 0-10000 vs class



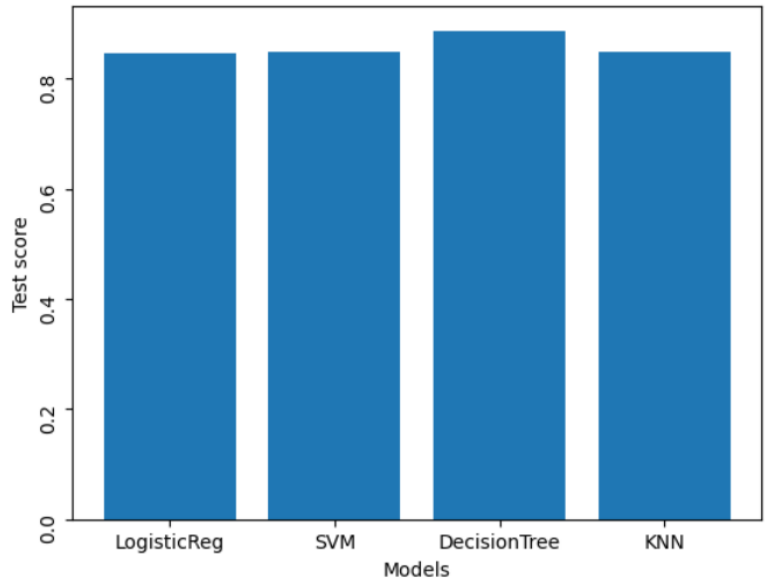
As we increase the range of payload mass, success and failure landing record increasing.

Section 5

Predictive Analysis (Classification)

Classification Accuracy

```
[33]: plt.bar(df_performance["models"], df_performance["score_test"])
      plt.xlabel("Models")
      plt.ylabel("Test score")
      plt.yticks(rotation=90)
      plt.show()
```



From four classification models we purpose, they have almost similar accuracy. But, Decision Tree has highest best score with best parameters we input such as:
'criterion': 'entropy',
'max_depth': 14,
'max_features': 'sqrt',
'min_samples_leaf': 1,
'min_samples_split': 10,
'splitter': 'random'

```
[34]: best_performance = df_performance[df_performance["score_test"] == df_performance["score_test"].max()]
      print(best_performance[["models", "score_test"]])
```

```
          models  score_test
2  DecisionTree      0.8875
```

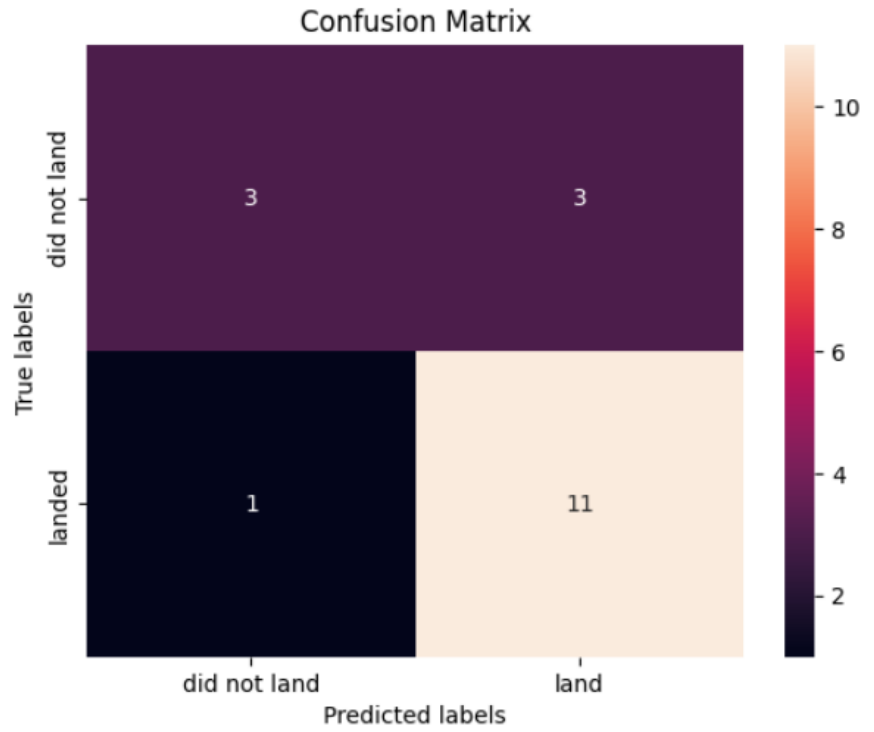
Confusion Matrix

```
[25]: tree_cv.score(X_test, Y_test)

[25]: 0.7777777777777778
```

We can plot the confusion matrix

```
[26]: yhat = tree_cv.predict(X_test)
      plot_confusion_matrix(Y_test,yhat)
```



From the confusion matrix, we see the true positive is more dominant and having score 0.778. It is lower than best accuracy score. It can cause by the size of test sample

Conclusions

- Most of the launch site is near the coastline either on eastern or western USA.
- There is significant correlation among success rate, flight number, and payload mass.
- There is not a significant correlation between the type of orbit with success rate. But, some orbit with high success rate have restricted payload mass.
- Four classification models: Logistic Regression, Support Vector Machine, Decision Tree, and K-Nearest Neighbor has similar accuracy. So, as complex as possible, the best model may be change.

Appendix

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

https://github.com/geez01/Python-DS-Project.git

Thank you!