IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Richard Lam
10/12/2025

Outline

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

Executive Summary

- Summary of methodologies
  - Data Collection with API and Webscraping
  - Data Wrangling
  - Exploratory Data Analysis with Structured Query Language (SQL) and Data Visualization
  - Interactive Visual Analysis and Dashboards with Folium and Plotly
  - Predictive Analysis with Machine Learning and Classification
- Summary of all results
  - Screenshots and results of Exploratory Data Analysis
  - Screenshots and results of Interactive Visual Analysis and Dashboard
  - Screenshots and results of Predictive Analysis

Introduction

- Project background and context
  - SpaceX is the most important space company and is affordable. The space company advertises that Falcon 9 rocket launches on its website with a cost of 62 million dollars; other space provides cost upward of 165 million dollar each. There is much of the savings 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. In this data science project, we like to predict the space rocket will land successfully.
- Problems you want to find answers
  - What will happen if the rocket lands successfully?
  - How does the space company determine the rocket's success rate?
  - Why will the rocket discover the orbit?
  - What is the best result of rocket's success rate?

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Data was collected by using Space API and scrapping from Wikipedia.
- Perform data wrangling
  - Data was processed by replacing missing, unnecessary and repetitive values.
  - One-hot encoding was a part of the key features in categorical variable.
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - The machine learning allows to build, tune, evaluate classification models and to determine the best results.

Data Collection

- The goal is to use space API and to scrape from Wikipedia for the data collection.

Data Collection – SpaceX API

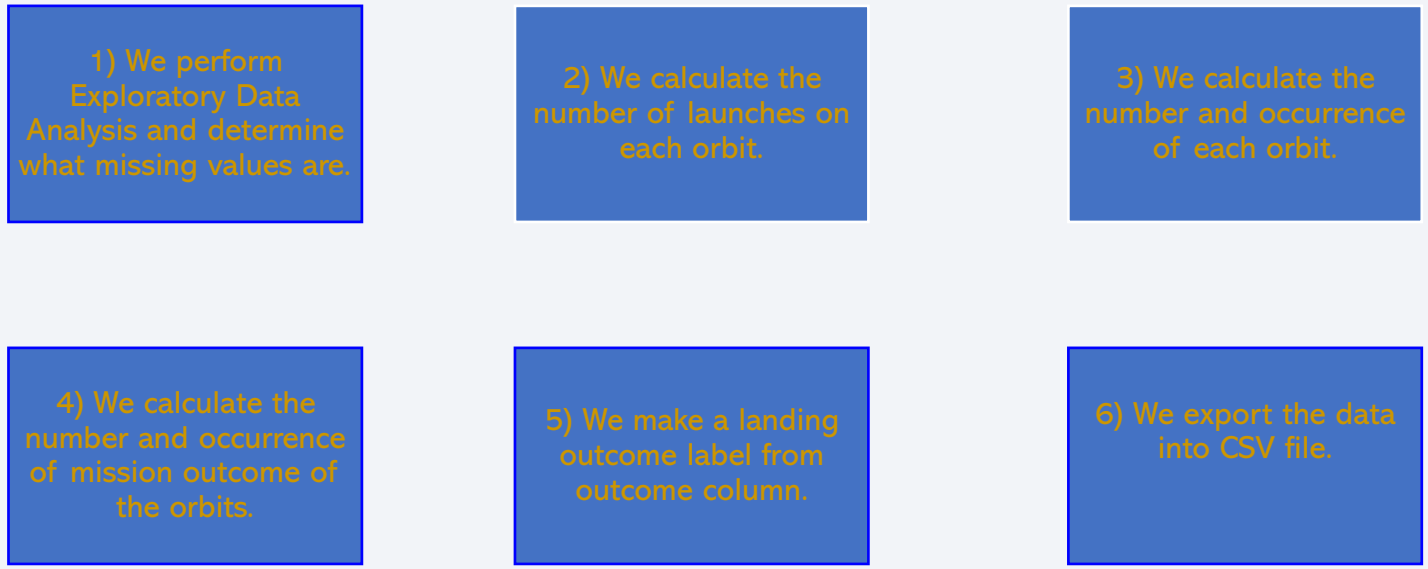
- Please see the flowchart in this slide, which is the right side.
- Link of data collection: https://github.com/richardlam4391/IBM_Data_Science_Capstone_Project/blob/main/jupyter-labs-spacex-data-collection-api.ipynb

Data Collection - Scraping

- Please see the flowchart in this slide, which is the right side.
- Link of web scraping: https://github.com/richardlam4391/IBM_Data_Science_Capstone_Project/blob/main/jupyter-labs-webscraping.ipynb

Data Wrangling

- The goal is to replace missing, unnecessary and repetitive values with numerical or categorical values. The number one represents outcome success, and the number zero also represents outcome failure.
- Please see the flowchart in this slide, which is the right side.
- Link of data wrangling: https://github.com/richardlam4391/IBM_Data_Science_Capstone_Project/blob/main/labs-jupyter-spacex-Data%20wrangling.ipynb



EDA with Data Visualization

- The scatterplot graph are Flight Number vs. Launch Site, Payload Mass vs Launch Site, Flight Number vs Orbit and Payload Mass vs Orbit. According to these scatterplot graphs, there are the relationships between variables.
- According to the bar graph “Class vs. Orbit”, there are the relationships between categorical and numerical variables.
- According to the line graph “Success Rate vs. Year”, there are the trends over time.
- Link of Exploratory Data Analysis with Data Visualization: https://github.com/richardlam4391/IBM_Data_Science_Capstone_Project/blob/main/edadataviz.ipynb

EDA with SQL

- First, The results of query show that we display the names of the unique launch sites in the space mission, five records where launch sites begin with the string 'CCA', the total payload mass carried by boosters launched by NASA (CRS), and average payload mas carried by booster version F9 v1.1. Second, The results of query also show that we list the date when the first successful landing outcome in ground pad was archieved, 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 and the records which will display the month names, failure landing_outcomes in drone ship, booster versions, launch_site for the months in year 2015. Finally, we rank the count of landing outcomes between the date 2010-06-04 and 2017-03-20 in descending order.
- Link of Exploratory Data Analysis with Structured Query Language: https://github.com/richardlam4391/IBM_Data_Science_Capstone_Project/blob/main/jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

- All launch sites are located in the coast of Florida and California (the United States).
- The objects include line and circles and are determined the successful or unsuccessful launches for each site. The red circle represents launching failure, and also the green circle represents launching success. The blue line represents the distance between the launch site and the proximity.
- Link of Interactive Map with Folium: https://github.com/richardlam4391/IBM_Data_Science_Capstone_Project/blob/main/lab_jupyter_launch_site_location.ipynb

Build a Dashboard with Plotly Dash

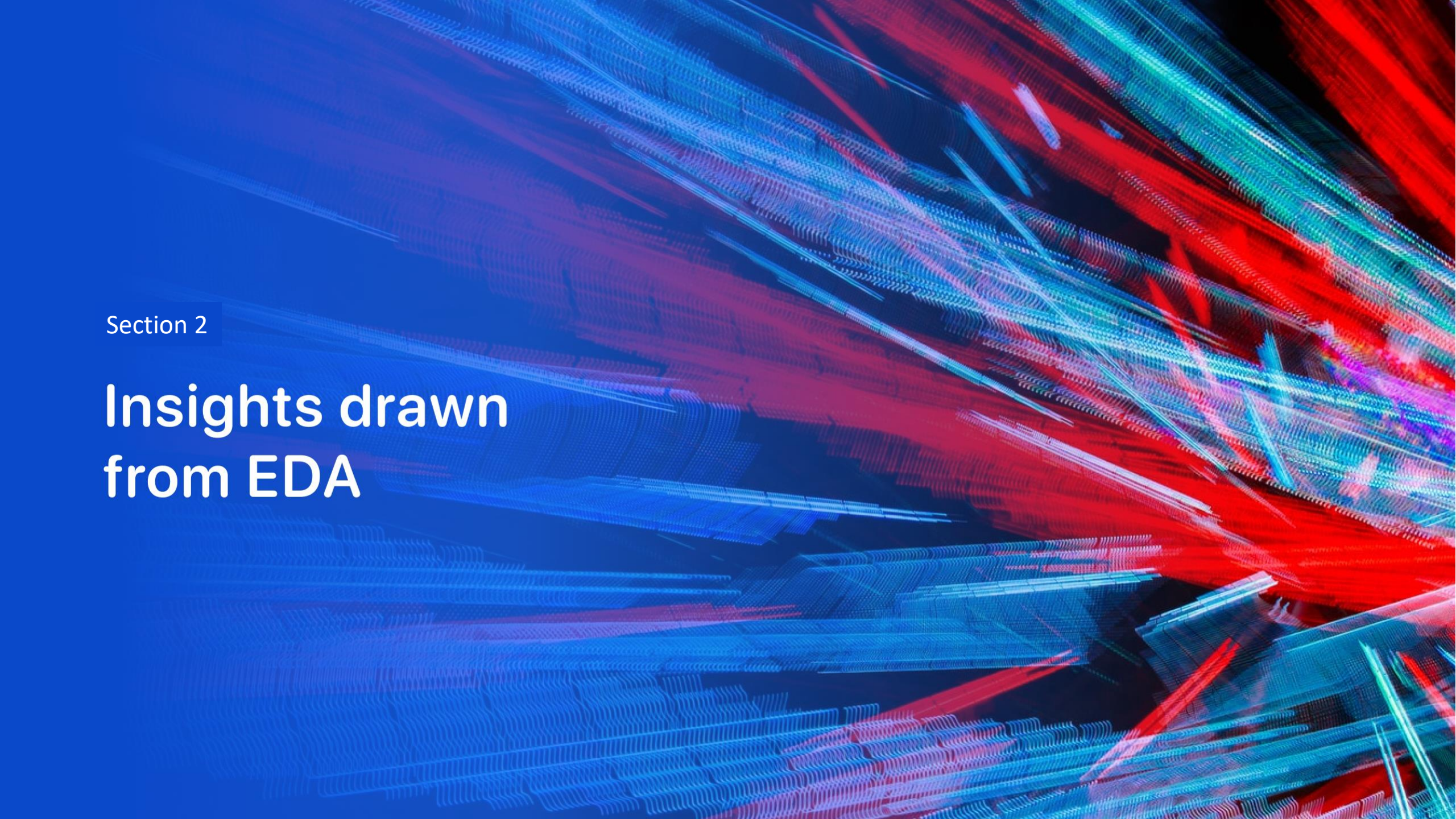
- We make the interactive dashboard with Plotly Dash, which has a dropdown menu, pie chard, slider and scatterplot.
- There are the totals of successful and unsuccessful launches for all launch sites in the pie chart.
- There are the relationships between payload mass and successful launches in the scatterplot.
- Link of Dashboard with Plotly Dash: https://github.com/richardlam4391/IBM_Data_Science_Capstone_Project/blob/main/Build%20an%20Interactive%20Dashboard%20with%20Plotly%20Dash.py

Predictive Analysis (Classification)

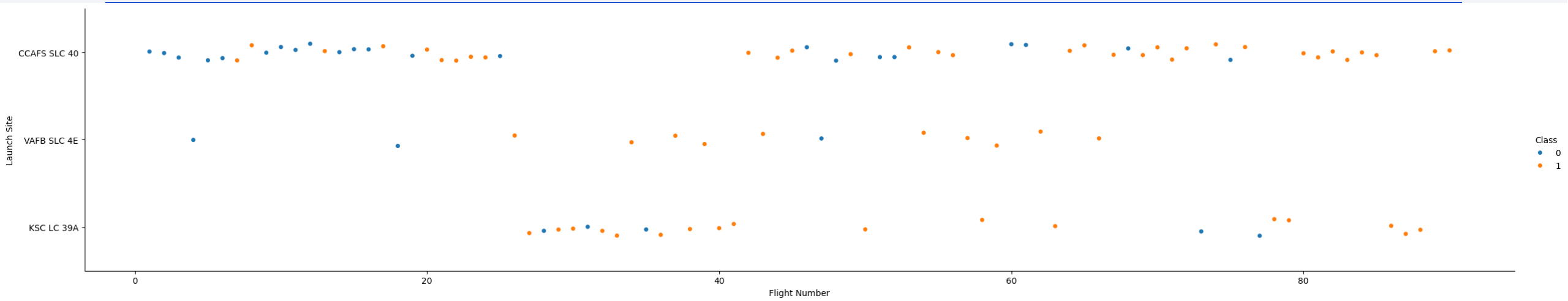
- The goal is to build, tune, evaluate classification models and to determine the best results.
- Please see the flowchart in this slide, which is the right side.
- Link of Predictive Analysis (Classification): https://github.com/richardlam4391/IBM_Data_Science_Capstone_Project/blob/main/SpaceX_Machine%20Learning%20Prediction_Part_5.ipynb

Results

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

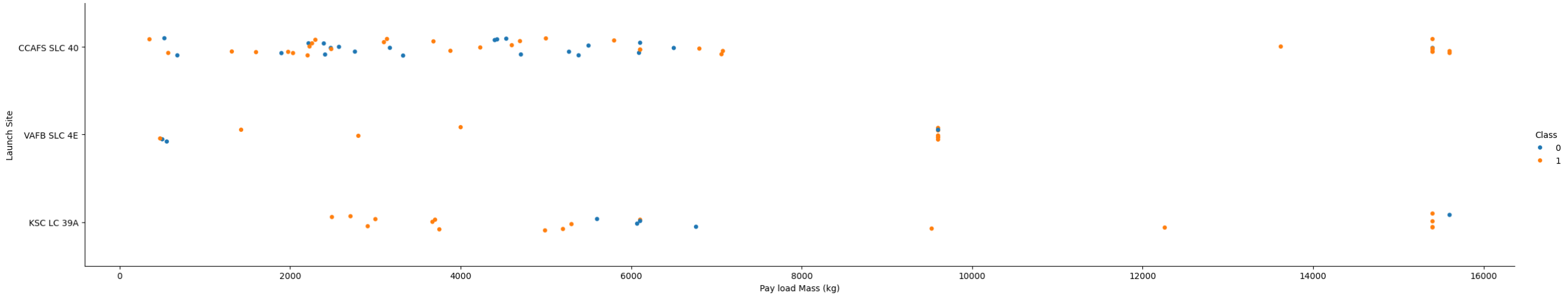

Section 2
Insights drawn from EDA

Flight Number vs. Launch Site



- The earlier flights do not land successfully while the later flights land successfully.
- The success rate of KSC LC 39A and VAFB SLC 4E is better.

Payload vs. Launch Site

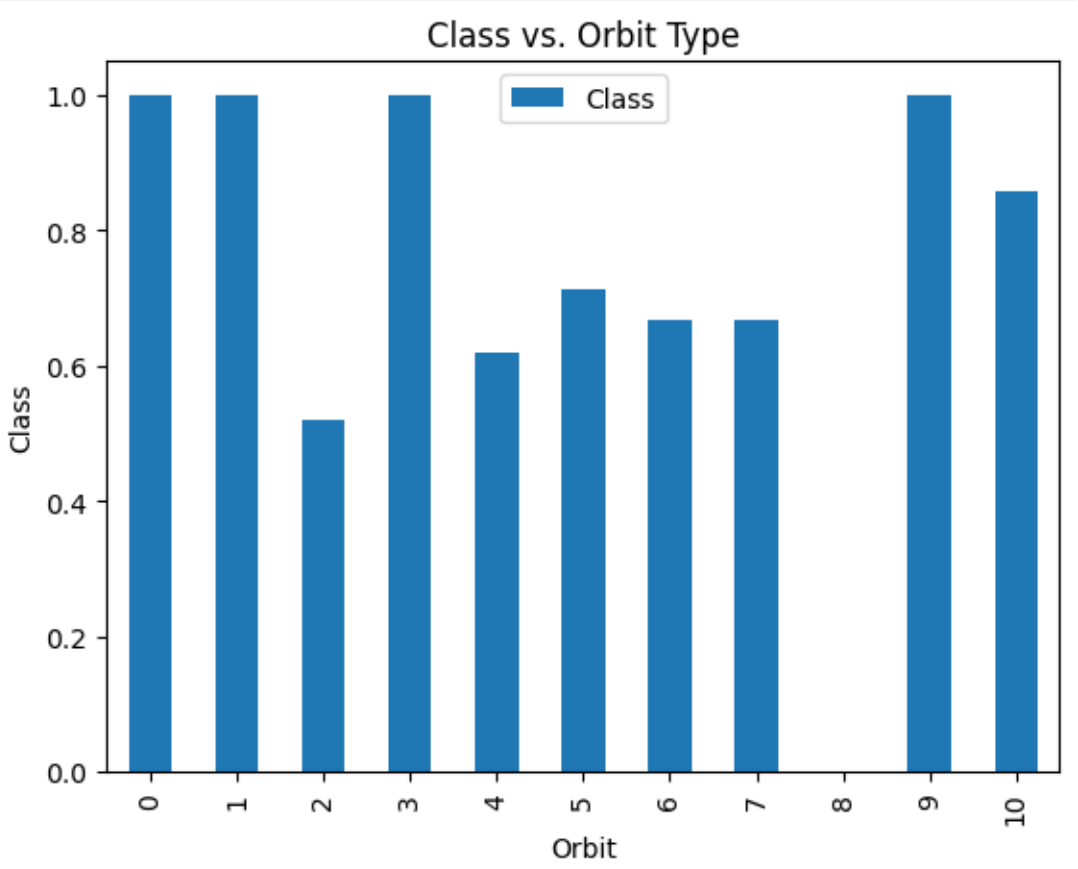


- KSC LC 39A (the launch site) have the better success rate under 5000 pay load mass (kg).
- The range of pay load mass is from light to heavy weight in all launch sites.

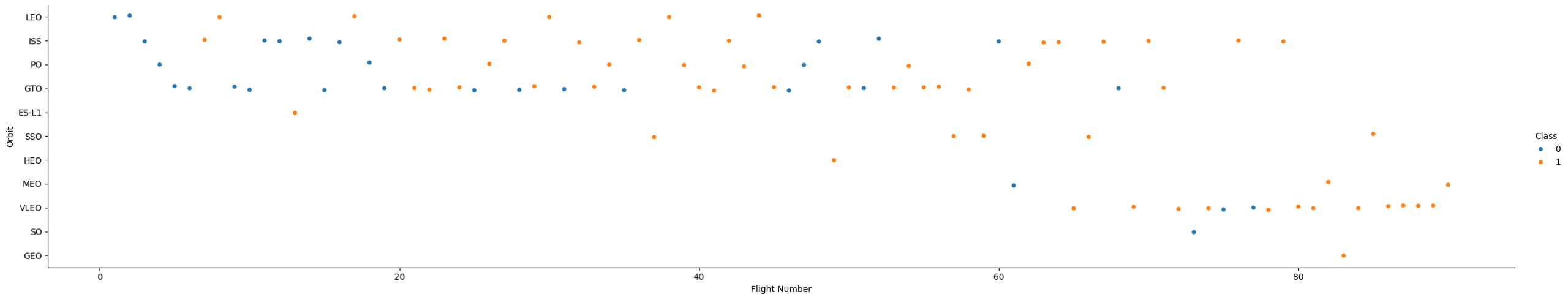
Success Rate vs. Orbit Type

- ES-L1, GEO, HEO and SSO (the orbits) have the best success rate.
- The success rate of SO (the orbit) is lower because its class is zero.

	Orbit	Class
0	ES-L1	1.000000
1	GEO	1.000000
2	GTO	0.518519
3	HEO	1.000000
4	ISS	0.619048
5	LEO	0.714286
6	MEO	0.666667
7	PO	0.666667
8	SO	0.000000
9	SSO	1.000000
10	VLEO	0.857143

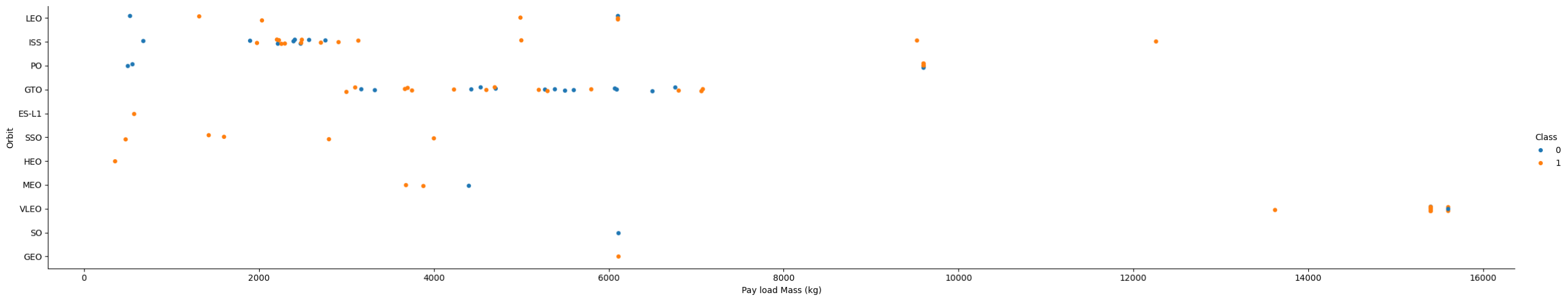


Flight Number vs. Orbit Type



- The success rate of GTO (the orbit) is lower under flight number 20.
- There are the relationships between LEO (the orbit) and number of flight.

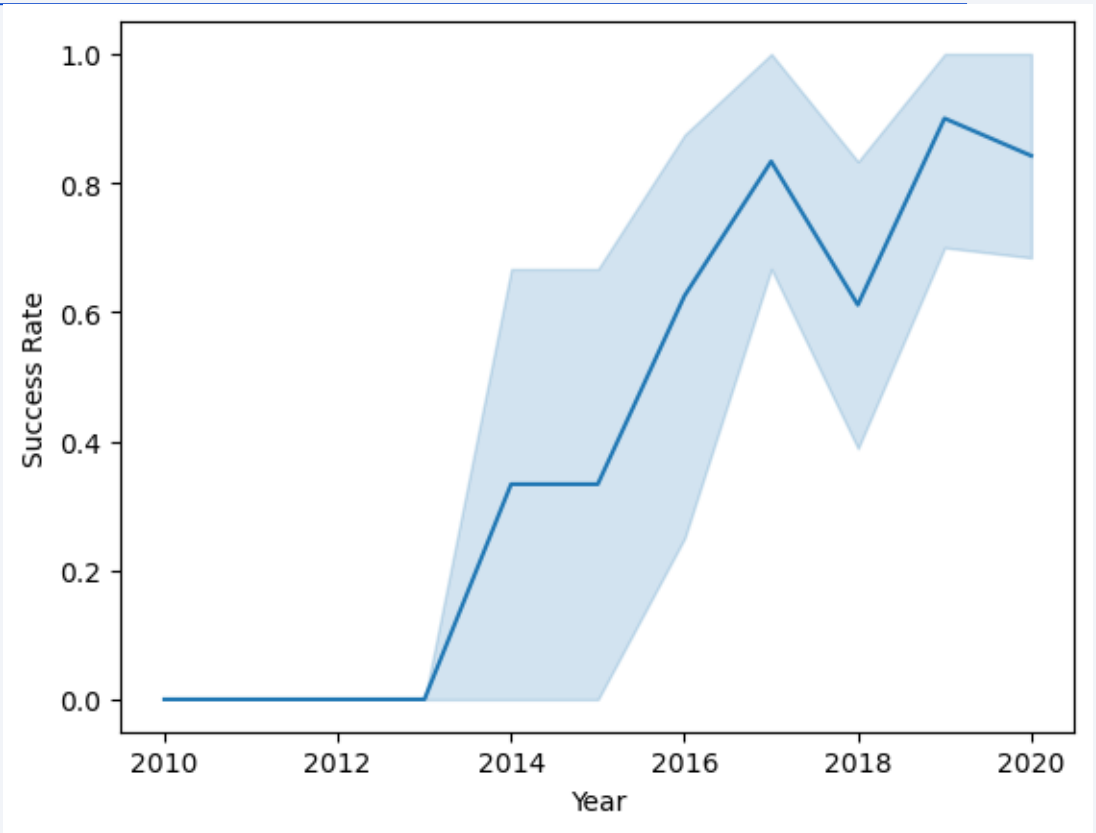
Payload vs. Orbit Type



- SO (the orbit) does not have the best success rate in the pay load mass.
- There are no relationships between pay load mass and success rate in GTO (the orbit).

Launch Success Yearly Trend

- According to the line graph, the slope did not change until 2013.
- The success rate was increasing since 2013.



All Launch Site Names

Task 1

Display the names of the unique launch sites in the space mission

In [10]:
```
%sql SELECT DISTINCT "Launch_Site" FROM SPACEXTABLE
```

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

Out[10]:

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

- We use DISTINCT clause to determine the four unique launch sites.

Launch Site Names Begin with 'CCA'

Task 2

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

In [11]:
```
%sql SELECT "Launch_Site" FROM SPACEXTABLE WHERE Launch_Site LIKE 'CCA%' LIMIT 5
```

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

Out[11]:

Launch_Site
CCAFS LC-40
CCAFS LC-40
CCAFS LC-40
CCAFS LC-40
CCAFS LC-40

- We use WHERE, LIKE and LIMIT clauses to determine the maximum five launch sites for the beginning of ‘CCA’.

Total Payload Mass

Task 3

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

In [12]:
```
%sql SELECT SUM (PAYLOAD_MASS__KG_) FROM SPACEXTABLE WHERE CUSTOMER = 'NASA (CRS)'
```

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

Out[12]:

SUM (PAYLOAD_MASS__KG_)
45596

- We use WHERE clause to calculate the total payload mass with the customers from NASA (CRS).

Average Payload Mass by F9 v1.1

Task 4

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

In [13]:
```
%sql SELECT AVG (PAYLOAD_MASS__KG_) FROM SPACEXTABLE WHERE Booster_Version = 'F9 v1.1'
```

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

Out[13]:

AVG (PAYLOAD_MASS__KG_)
2928.4

- We use WHERE clause to calculate the average of payload mass with the Booster Version F9 v1.1.

First Successful Ground Landing Date

Task 5

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

Hint:Use min function

In [14]:
```
%sql SELECT MIN ("Date") AS "First Successful Landing" FROM SPACEXTABLE WHERE "Landing_Outcome" = 'Success (ground pad)'
```

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

Out[14]:

First Successful Landing
2015-12-22

- We use MIN, AS and WHERE clause to determine the oldest date when the rocket landed in ground pad successfully.

Successful Drone Ship Landing with Payload between 4000 and 6000

Task 6

List the names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000

In [15]:
```
%sql SELECT "Booster_Version" FROM SPACEXTABLE WHERE "Landing_Outcome" = 'Success (drone ship)' AND PAYLOAD_MASS__KG_ BETWEE
```

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

Out[15]:

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

- We use WHERE clause to determine the booster versions that have success in drone ship between 4000 kg and 6000 kg.

Total Number of Successful and Failure Mission Outcomes

Task 7

List the total number of successful and failure mission outcomes

In [16]:
```
%sql SELECT COUNT ("Mission_Outcome") AS "Mission Success" FROM SPACEXTABLE WHERE "Mission_Outcome" LIKE 'Success%'
```

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

Out[16]:

Mission Success
100

In [17]:
```
%sql SELECT COUNT ("Mission_Outcome") AS "Mission Failure" FROM SPACEXTABLE WHERE "Mission_Outcome" LIKE 'Failure%'
```

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

Out[17]:

Mission Failure
1

- We use WHERE, AS and LIKE clauses to count the mission successes and failures.

Boosters Carried Maximum Payload

Task 8

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

In [18]:
```
%sql SELECT DISTINCT ("Booster_Version") AS "Booster Versions Where Payload Mass is Maxed" FROM SPACEXTABLE \
    WHERE "PAYLOAD_MASS__KG_" = (SELECT MAX ("PAYLOAD_MASS__KG_") FROM SPACEXTABLE)
```

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

Out[18]:

Booster Versions Where Payload Mass is Maxed
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

- We use DISTINCT, WHERE, and AS clauses to determine the booster versions that maximize the payload mass.

2015 Launch Records

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.

Note: SQLLite does not support monthnames. So you need to use substr(Date, 6,2) as month to get the months and substr(Date,0,5)='2015' for year.

In [19]:
```
%sql SELECT substr(Date, 6,2) AS "Month", "Date", "Booster_Version", "Launch_Site", "Mission_Outcome", "Landing_Outcome" FRO
WHERE "Date" LIKE  '2015-%' AND "Landing_Outcome" = 'Failure (drone ship)'
```

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

Out[19]:

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

- We use WHERE, AND and AS clauses to determine the booster versions, month, date, launch site and mission outcome and landing outcome in year 2015.

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

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.

In [20]:
```
%sql SELECT "Landing_Outcome", COUNT("Landing_Outcome") AS "Total Count" FROM SPACEXTABLE \
WHERE "Date" BETWEEN '2010-06-04' AND '2017-03-20' GROUP BY "Landing_Outcome" ORDER BY COUNT("Landing_Outcome") DESC
```

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

Out[20]:

Landing_Outcome	Total Count
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

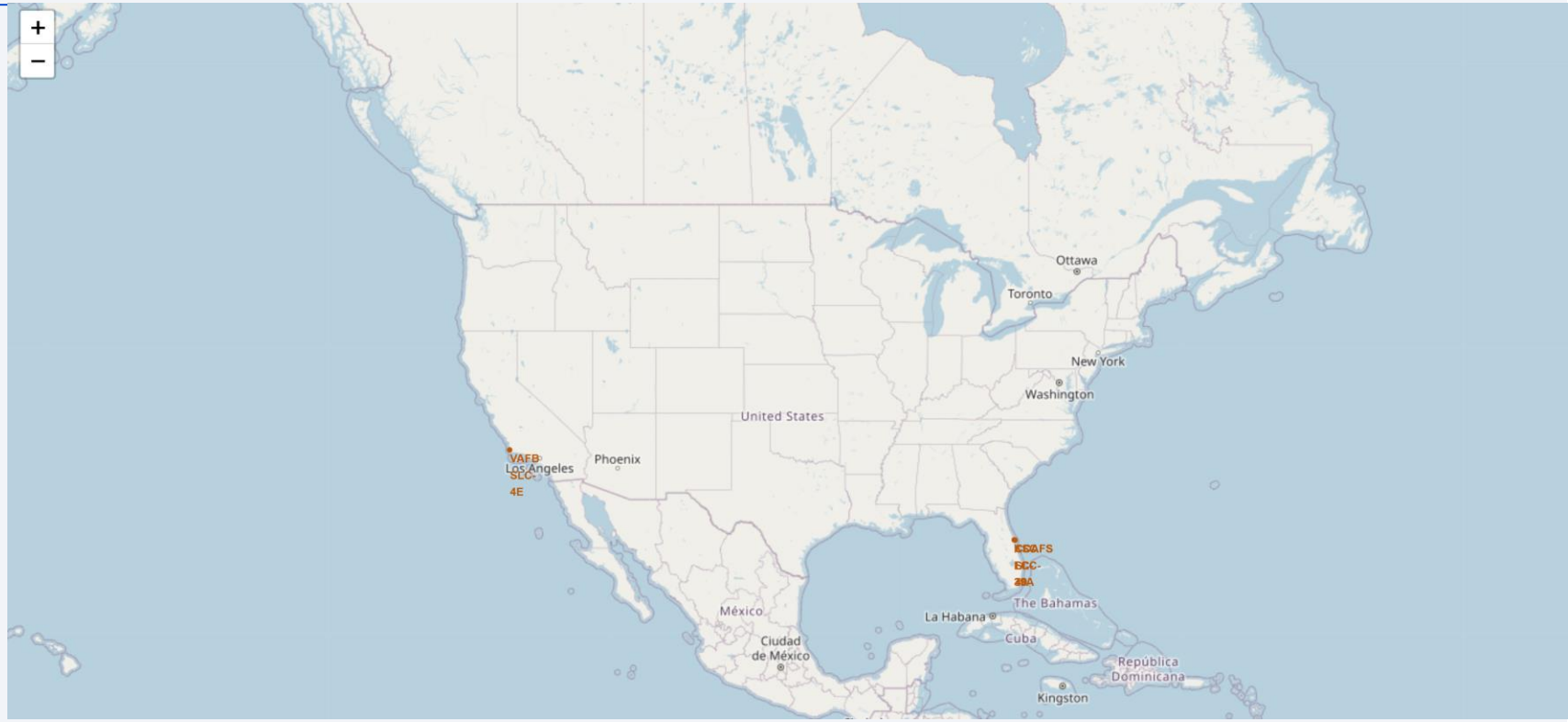
- We use COUNT, and AS clauses and descending order to count the landing outcomes between 2010-06-04 and 2017-03-20.

Section 3

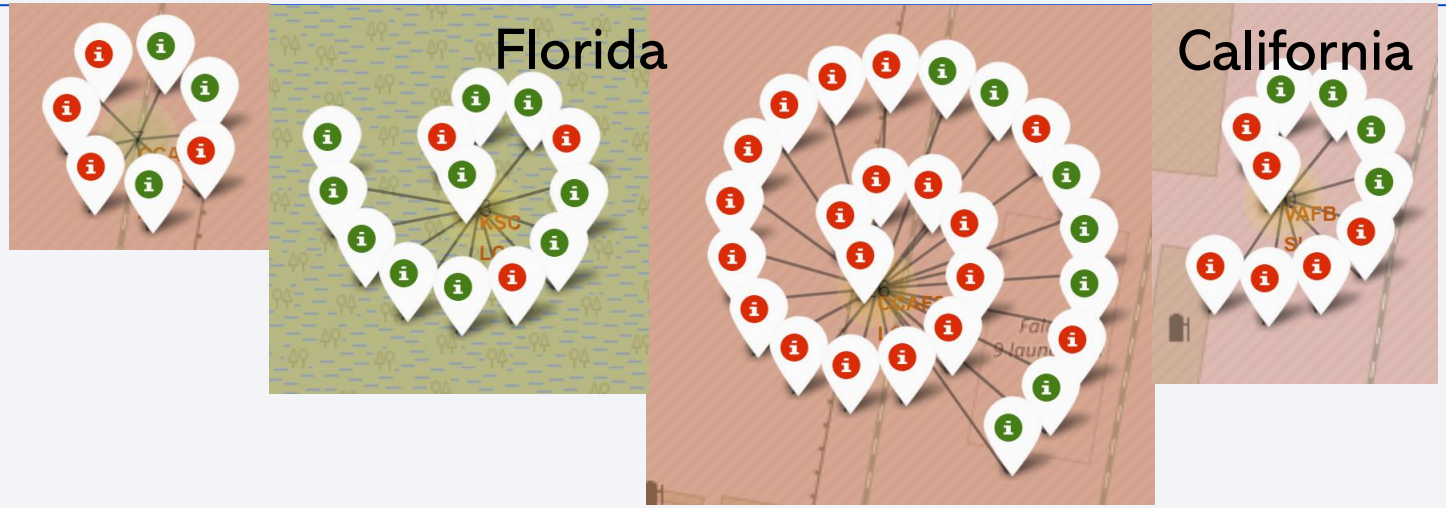
Launch Sites Proximities Analysis

Launches Sites in California and Florida (United States)

- All launch sites are found in the coast of Florida and California (the United States).



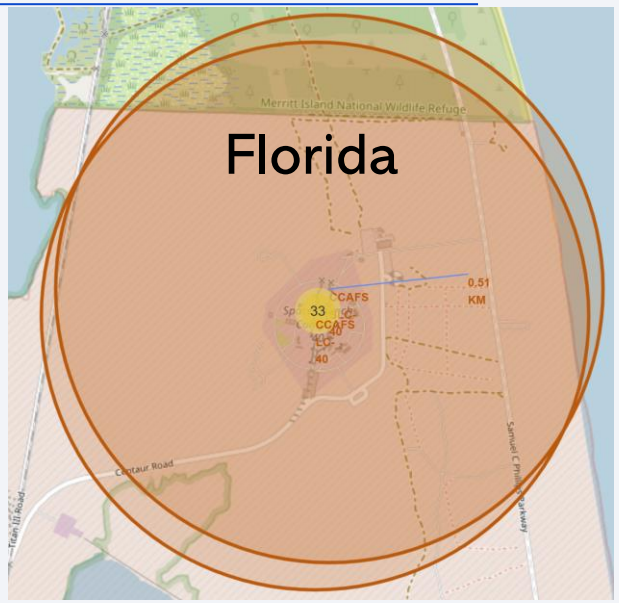
Color Markers in Launch Sites

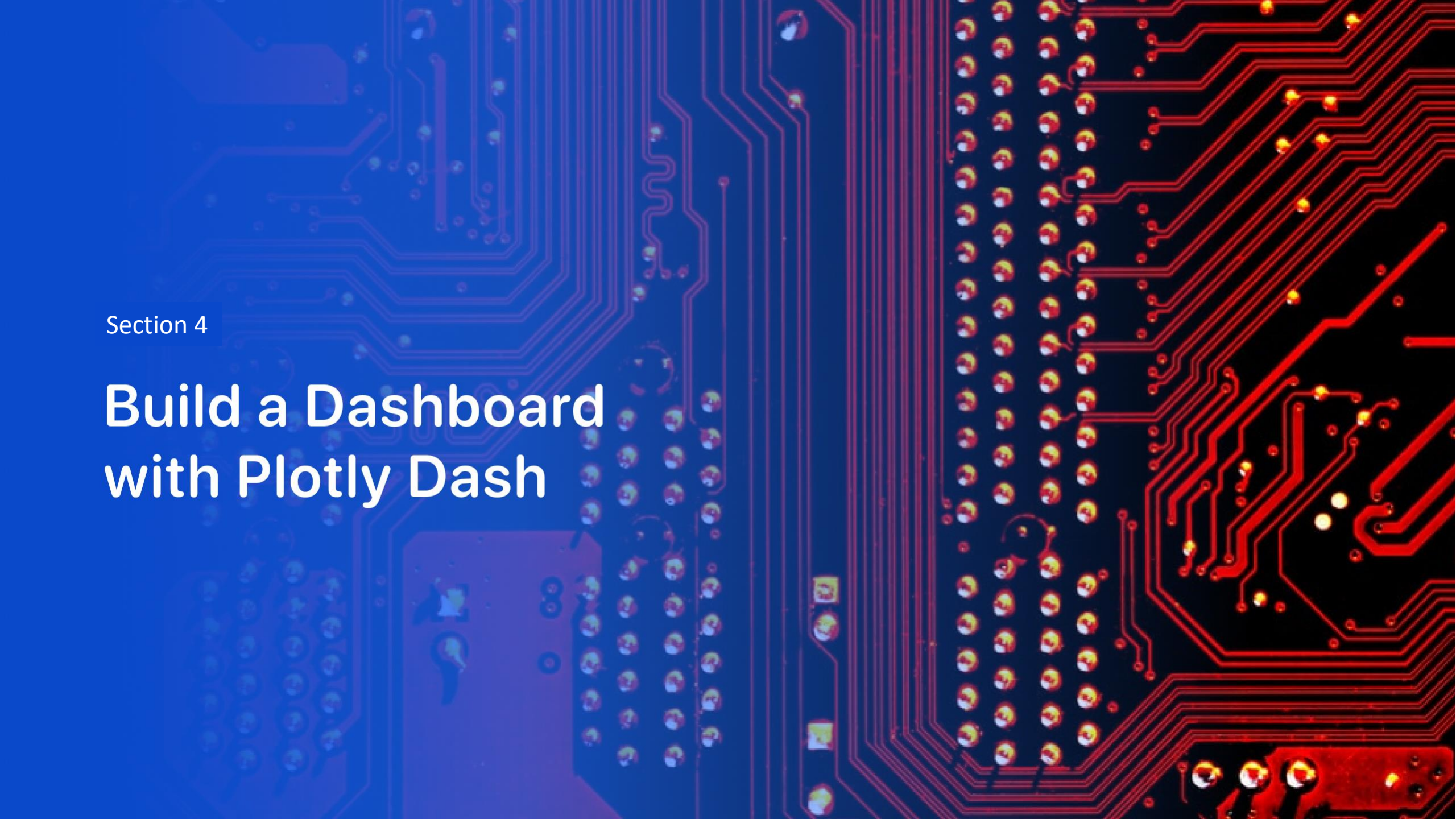


- We use zoom-in to see a lot of red markers and a few green makers.
- In both Florida and California, the red marker represents the unsuccessful launches for the sites, and also the green marker represents the unsuccessful launches for the sites.

Distance Between Launch Sites and Proximity

- The blue line represents the distance between launch site and proximity to one of areas.
- The launch sites are not in proximity to railroad, highway, city and coastline.




Section 4
Build a Dashboard with Plotly Dash

<Dashboard Screenshot 1>

- Replace <Dashboard screenshot 1> title with an appropriate title
- Show the screenshot of launch success count for all sites, in a piechart
- Explain the important elements and findings on the screenshot

<Dashboard Screenshot 2>

- Replace <Dashboard screenshot 2> title with an appropriate title
- Show the screenshot of the piechart for the launch site with highest launch success ratio
- Explain the important elements and findings on the screenshot

<Dashboard Screenshot 3>

- Replace <Dashboard screenshot 3> title with an appropriate title
- Show screenshots of Payload vs. Launch Outcome scatter plot for all sites, with different payload selected in the range slider
- Explain the important elements and findings on the screenshot, such as which payload range or booster version have the largest success rate, etc.

Section 5

Predictive Analysis (Classification)

Classification Accuracy

- The accuracy on the data test classification is between 0.80 and 0.90.
- The decision tree has the highest accuracy on the data test classification.

In [19]:
```
print("tuned hpyerparameters :(best parameters) ",logreg_cv.best_params_)
print("accuracy :",logreg_cv.best_score_)
```

```
tuned hpyerparameters :(best parameters)  {'C': 0.01, 'penalty': 'l2', 'solver': 'lbfgs'}
accuracy : 0.8464285714285713
```

In [25]:
```
print("tuned hpyerparameters :(best parameters) ",svm_cv.best_params_)
print("accuracy :",svm_cv.best_score_)
```

```
tuned hpyerparameters :(best parameters)  {'C': 1.0, 'gamma': 0.03162277660168379, 'kernel': 'sigmoid'}
accuracy : 0.8482142857142856
```

In [31]:
```
print("tuned hpyerparameters :(best parameters) ",tree_cv.best_params_)
print("accuracy :",tree_cv.best_score_)
```

```
tuned hpyerparameters :(best parameters)  {'criterion': 'gini', 'max_depth': 6, 'max_features': 'sqrt', 'min_samples_leaf':
1, 'min_samples_split': 5, 'splitter': 'best'}
accuracy : 0.8607142857142855
```

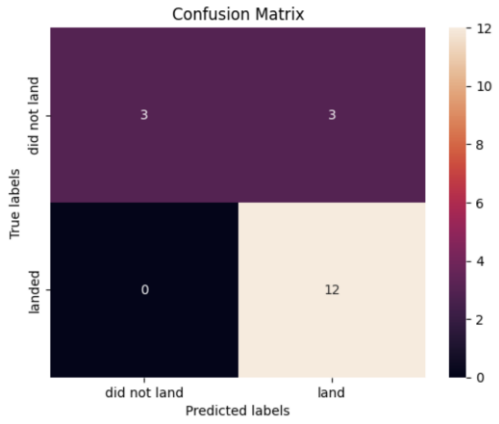
In [37]:
```
print("tuned hpyerparameters :(best parameters) ",knn_cv.best_params_)
print("accuracy :",knn_cv.best_score_)
```

```
tuned hpyerparameters :(best parameters)  {'algorithm': 'auto', 'n_neighbors': 10, 'p': 1}
accuracy : 0.8482142857142858
```

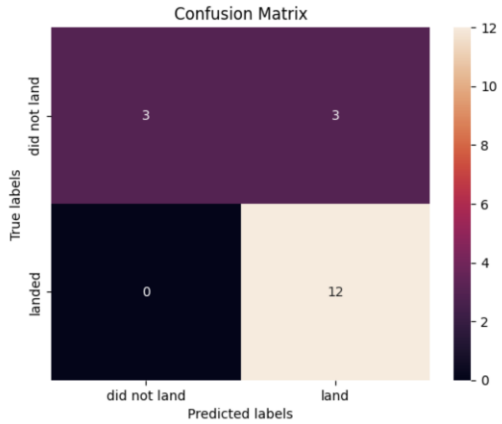
Confusion Matrix

- The confusion matrix on the test data classification, except decision tree, are same.
- The main problem is the false positive because the classifier marks the successful landing as the unsuccessful landing.

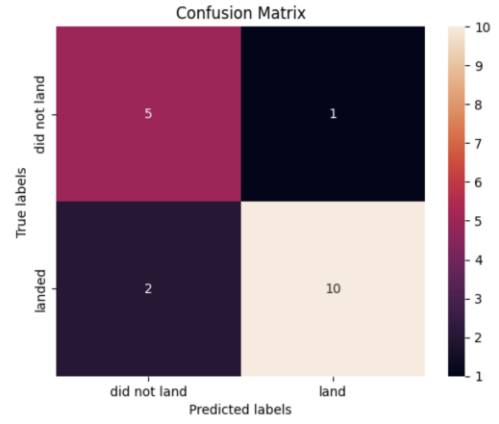
```
In [22]: yhat=logreg_cv.predict(X_test)
         plot_confusion_matrix(Y_test,yhat)
```



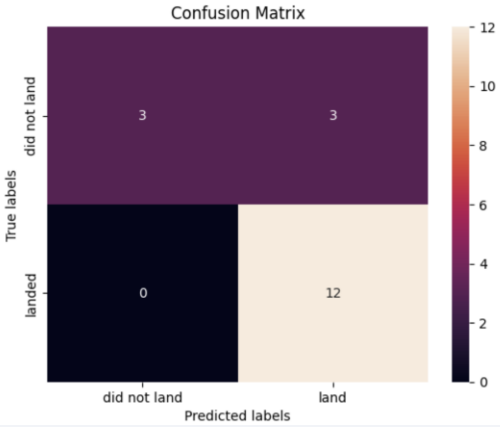

```
In [27]: yhat=svm_cv.predict(X_test)
         plot_confusion_matrix(Y_test,yhat)
```




```
In [33]: yhat = tree_cv.predict(X_test)
         plot_confusion_matrix(Y_test,yhat)
```




```
In [39]: yhat = knn_cv.predict(X_test)
         plot_confusion_matrix(Y_test,yhat)
```



Conclusions

- The decision tree is the highest result in the machine learning project.
- In the line graph, the success rate will increase since 2013.
- There are many successful and unsuccessful markers in California and Florida (the United States).
- The success rate of ES-L1, GEO, HEO and SSO (the orbits) is better because the class of these orbits are 1.0.
- KSC LC 39A is the number one launch site and has the best success rate.

Appendix

- Data Source:
- API: https://api.spacexdata.com/v4/launches/past
- Webscrapping: https://en.wikipedia.org/wiki/List_of_Falcon_9_and_Falcon_Heavy_launches
- Data Wrangling: https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DS0321EN-SkillsNetwork/datasets/dataset_part_1.csv
- Exploratory Data Analysis and Visualization: https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DS0321EN-SkillsNetwork/datasets/dataset_part_2.csv
- Exploratory Data Analysis and SQL: https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DS0321EN-SkillsNetwork/labs/module_2/data/Spacex.csv
- Interactive Visual Analytics with Folium: https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DS0321EN-SkillsNetwork/datasets/spacex_launch_geo.csv
- Interactive Visual Analytics with Plotly: https://www.kaggle.com/datasets/tliolopes/spacex-launch-dash
- Predictive Analysis (Classifcation): https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DS0321EN-SkillsNetwork/datasets/dataset_part_2.csv

Thank you!