IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Pascoal Antonio da Silva Mello
18/03/2022

Outline

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

Executive Summary

- Data Science formation

  The world needs professionals that can analyse data and presents results and solutions to the problems from all industries.

  So, I see an opportunity to learn about this new professional that so important currently and for next years too.

  I started the modules in the suggested sequence, I studied the content and did all labs.

- Summary of all results

  After each lab I can see the potencial gain that is possible to do in my work, because we can search for solutions to critical problems related to our job, reduce the impact on the environment, optimize time and resources.

  Beyound, we can see oportunities that before we couldnt:

  - Find necessities before that market can feel it.
    - the best value to apartment or cars
  - Reduce costs in production, transportation, implementation
    - the ideal consume of gas in the automotives
  - Solutions to critical problems
    - How to know the best drug we can use in patients or have success in rocket launch and first stage land

Introduction

- Project background and context
  - Some problems have big impact in results and need to be analyzed, based on all data history that could be collated, understanding and preparing it.
  - To do this, it is important we understand the business , to help clarify the goal of the entity asking the question.
  - To do the correct approach to help identify that type of patterns will be needed to address the question most effectively.
- Problems you want to find answers
  - How to know the best drug we can use in critical patients ?
  - How can I specify the value about something ?
  - How can I suggest the ideal movies, series that our client can like ?
  - How can I find suspicious activity with my credit card ?
  - How can I have success in launch rockets to the orbits ? **(SpaceX we will see more details about)**

Section 1

Methodology

Methodology

Executive Summary

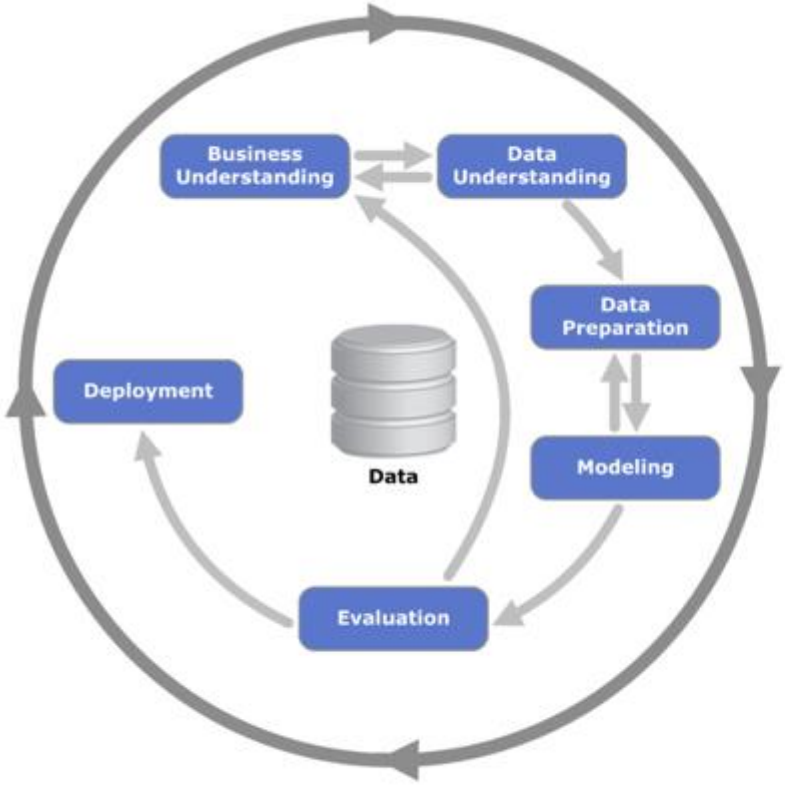
- Data collection methodology:
  - All features about the situation are collected and stored in files, identify and gather the available data resources. These can be in the form of structured, unstructured, and even semi-structured data relevant to the problem domain.
  - The data requirements are revised and decisions are made as to whether or not more data is needed.
- Perform data wrangling
  - Converting data from the initial format to a format that may be better for analysis (like csv, html, sql)
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - To do this we need to prepare data, split to train and test data, and find the best model

Data Collection

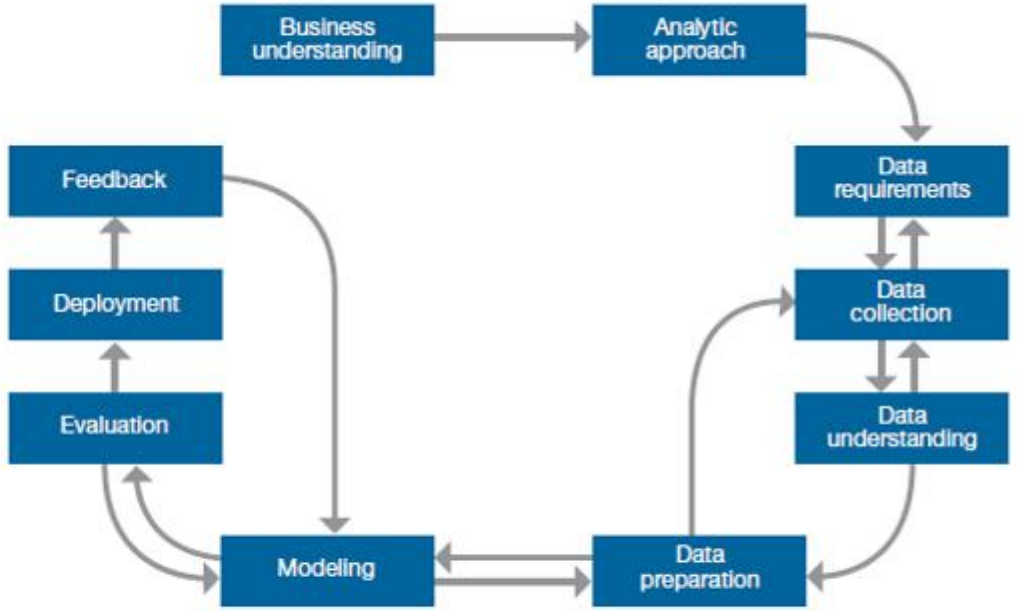
- Describe how data sets were collected.
  - In the SPACEX for example, we collected data from internet page (spacex_url), but to simplify the process we use a file in IBM site (df), and put in a variable:
    - df=pd.read_csv("https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DS0321EN-SkillsNetwork/datasets/dataset_part_2.csv")
    - spacex_url=https://api.spacexdata.com/v4/launches/past

- identify the available data resources relevant to the problem domain.
- To retrieve data, we can do web scraping on a related website, or we can use repository with premade datasets ready to use. Usually, premade datasets are CSV files or Excel; anyway,
- To collect data from any website or repository, we should use Pandas, a useful tool to download, convert, and modify datasets.

Data Collection



Cross Industry Standard Process for Data Mining



IBM Data Science Methodology

Source: https://www.ibmbigdatahub.com/blog/why-we-need-methodology-data-science

Data Collection – SpaceX API

- Present data collection with SpaceX REST
- Get, post, url, json

```
# Download and read the `spacex_launch_geo.csv`

spacex_csv_file = wget.download('https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DS0321EN-SkillsNetwork/datasets/spacex_launch_geo.csv')

spacex_df=pd.read_csv(spacex_csv_file)
```

- GitHub URL of the completed SpaceX API calls notebook :
- https://github.com/cervicalmello/datascience/blob/main/jupyter-labs-spacex-data-collection-api_SPACEX_complete_ResumoTotal.ipynb

```
1 spacex_url="https://api.spacexdata.com/v4/launches/past"
```

```
1 response = requests.get(spacex_url)
```

Check the content of the response

```
1 print(response.content)
```

```
s://live.staticflickr.com/65535/51253353892_de82b01e23_o.jpg","https:
_o.jpg","https://live.staticflickr.com/65535/51254829154_3c5980c086_
82_e59ea4df4f_o.jpg","https://live.staticflickr.com/65535/5125482913
535/51262926489_9fbce20e9c_o.jpg","https://live.staticflickr.com/655
flickr.com/65535/51262179176_e4302db116_o.jpg","https://live.staticfl
skit":null,"webcast":"https://youtu.be/QJXxVtp3KqI","youtube_id":"QJ
ipedia.org/wiki/GPS_Block_III"},"static_fire_date_utc":"2021-06-13T1
t":false,"window":900,"rocket":"5e9d0d95eda69973a809d1ec","success":t
I launch will use the first stage from the previous GPS mission. This
unch has flown on a flight proven booster. Falcon 9 will launch from
nrange on a drone ship. GPS III is the third generation of the U.S. S
llites, developed by Lockheed Martin. The GPS III constellation will
ure, allowing the entire GPS constellation to be updated simultaneous
```

Data Collection - Scraping

- Web scraping process

- Add the GitHub URL of the completed web scraping notebook: https://github.com/cervicalmello/datascience/blob/main/Hands-on%20Lab_%20Data%20Collection%20with%20Web%20Scraping.ipynb

To keep the lab tasks consistent, you will be asked to scrape the data from a snapshot of the `List of Falcon 9 and Falcon Heavy launches` Wikipage updated on `9th June 2021`

```
static_url = "https://en.wikipedia.org/w/index.php?title=List_of_Falcon_9_and_Falcon_Heavy_launches&oldid=1027686922"
```

Next, request the HTML page from the above URL and get a `response` object

TASK 1: Request the Falcon9 Launch Wiki page from its URL

First, let's perform an HTTP GET method to request the Falcon9 Launch HTML page, as an HTTP response.

```
# use requests.get() method with the provided static_url
# assign the response to a object
data  = requests.get(static_url).text
```

Create a `BeautifulSoup` object from the HTML `response`

```
# Use BeautifulSoup() to create a BeautifulSoup object from a response text content
soup = BeautifulSoup(data, 'html5lib')
```

Print the page title to verify if the `BeautifulSoup` object was created properly

```
# Use soup.title attribute
print(soup.title)
```

```
<title>List of Falcon 9 and Falcon Heavy launches - Wikipedia</title>
```

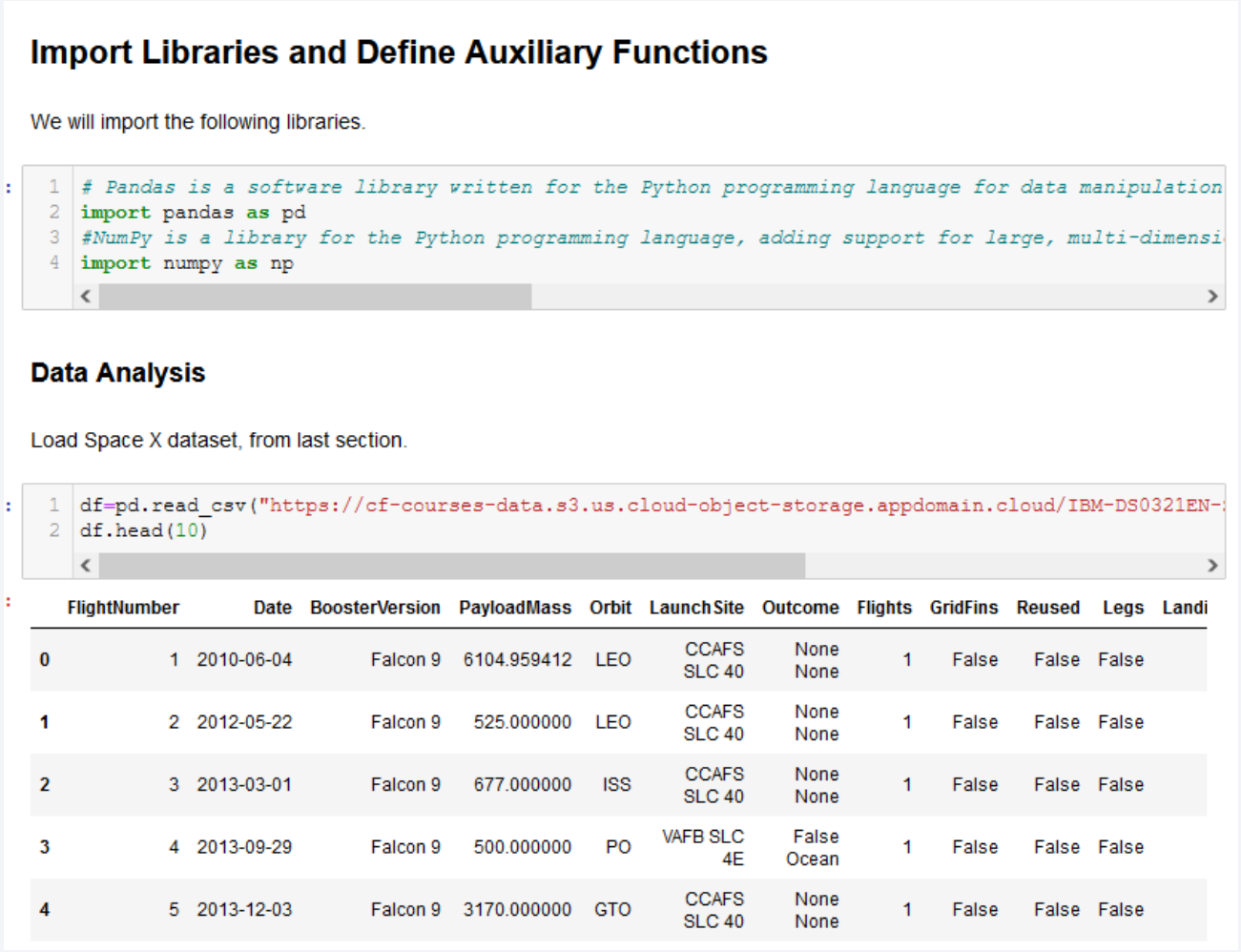
Data Wrangling

- Describe how data were processed

  - We perform some Exploratory Data Analysis (EDA) to find some patterns in the data and determine what would be the label for training supervised models.

  - mainly we convert those outcomes into Training Labels with 1 means the booster successfully landed 0 means it was unsuccessful.

- data wrangling process
- the GitHub URL:
- https://github.com/cervicalmello/datascience/blob/main/SPACEX%20Data%20Wrangling%20Overview.ipynb



Import Libraries and Define Auxiliary Functions

We will import the following libraries.

```
# Pandas is a software library written for the Python programming language for data manipulation
import pandas as pd
#NumPy is a library for the Python programming language, adding support for large, multi-dimensi
import numpy as np
```

Data Analysis

Load Space X dataset, from last section.

```
df=pd.read_csv("https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DS0321EN-
df.head(10)
```

	FlightNumber	Date	BoosterVersion	PayloadMass	Orbit	LaunchSite	Outcome	Flights	GridFins	Reused	Legs	Landi
0	1	2010-06-04	Falcon 9	6104.959412	LEO	CCAFS SLC 40	None None	1	False	False	False	
1	2	2012-05-22	Falcon 9	525.000000	LEO	CCAFS SLC 40	None None	1	False	False	False	
2	3	2013-03-01	Falcon 9	677.000000	ISS	CCAFS SLC 40	None None	1	False	False	False	
3	4	2013-09-29	Falcon 9	500.000000	PO	VAFB SLC 4E	False Ocean	1	False	False	False	
4	5	2013-12-03	Falcon 9	3170.000000	GTO	CCAFS SLC 40	None None	1	False	False	False	

EDA with Data Visualization

- Summarize what charts were plotted and why we used those charts:
  - Scatter Point – Visualize the relationship between Payload and Launch Site, FlightNumber and Orbit type, Payload and Orbit type
  - Bar chart for the success rate of each orbit type
  - Line Chart - observe that the sucess rate since 2013 kept increasing till 2020
- the GitHub URL of your completed EDA with data visualization notebook:
- https://github.com/cervicalmello/datascience/blob/main/SPACEX%20SQL%20DATAVIZ_complete_overview_bar_chart_funcrions.ipynb

EDA with SQL

- Summarize the SQL queries
  - %sql select DISTINCT LAUNCH_SITE from SPACEXDATASET
  - %sql select LAUNCH_SITE from SPACEXDATASET where LAUNCH_SITE LIKE 'CCA%'
  - %sql select SUM(payload_mass__kg_) from spacexdataset where customer like 'NASA%'
  - %sql select AVG(payload_mass__kg_) from spacexdataset where booster_version like 'F9 v1.1'
  - %sql select MIN(date) from spacexdataset where landing__outcome like 'Success'
- GitHub URL : https://github.com/cervicalmello/datascience/blob/main/Handson_Using%20SQL%20Overview_complete.ipynb

Build an Interactive Map with Folium

- Summarize what map objects such as markers, circles, lines, etc. you created and added to a folium map:
  - folium.Circle: to make a circle area in the map
  - Folium.map.Marker: to mark the point in the map
  - Site_map.add_child: add coordenate/points in the map
- Add the GitHub URL: https://github.com/cervicalmello/datascience/blob/main/SPACEX_Maps_Overview_launch_site_location_INcompplete.ipynb

Build a Dashboard with Plotly Dash

- What plots/graphs and interactions added to a dashboard:
  - Pie chart – To see % the successful or not about launch
  - Scatter Chart – To identify how many payload mass are more success (Class)
  - Dropdown list – to choose Launch Sites
  - Range Slider – to select a payload range
- the GitHub URL: https://github.com/cervicalmello/datascience/blob/main/spacex_dash_app_DASHBOARD_overview.py

Predictive Analysis (Classification)

- Summarize how to built, evaluated, improved, and found the best performing classification model
- Load dataframe,
- standardize the data
  - preprocessing.StandardScaler()
- split data into training and testing
  - X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size=0.2, random_state=2)
- , create objects to use with GridSearchCV (logreg_cv=GridSearchCV(lr,parameters,cv=10))
- find the best parameters, calculate accuracy

  logreg_cv.best_params_

  logreg_cv.best_score_
- the GitHub URL: https://github.com/cervicalmello/datascience/blob/main/SPACEX%20ML%20Overview_Predicition_lab_INcomplete.ipynb

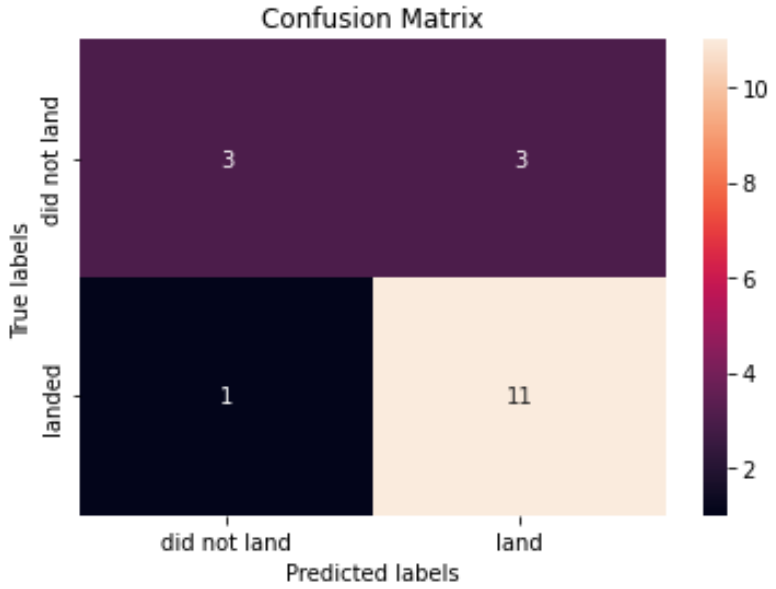
Results

```
tree_cv.score(X_train, Y_train)
```

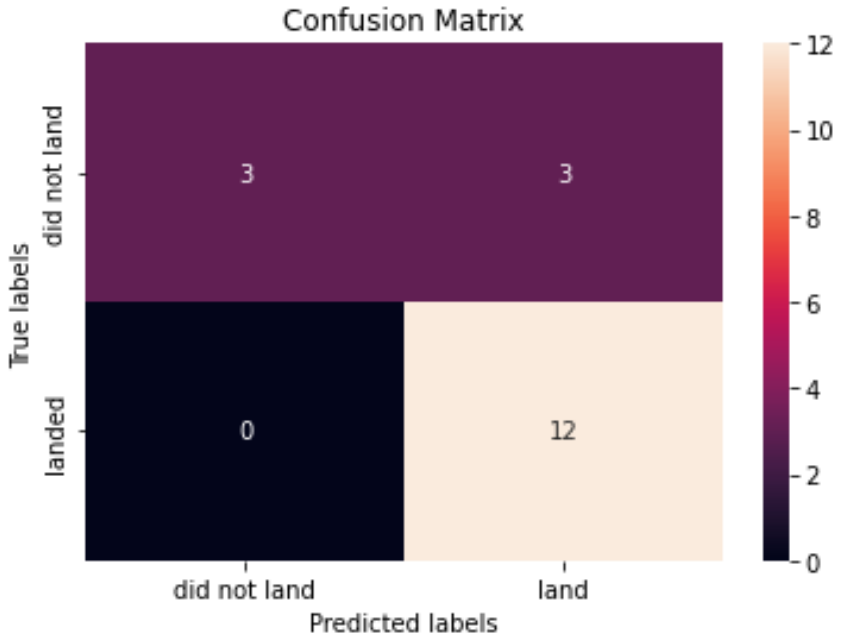
```
0.8333333333333334
```

We can plot the confusion matrix

```
yhat = tree_cv.predict(X_test)
plot_confusion_matrix(Y_test,yhat)
```




```
yhat=logreg_cv.predict(X_test)
plot_confusion_matrix(Y_test,yhat)
```



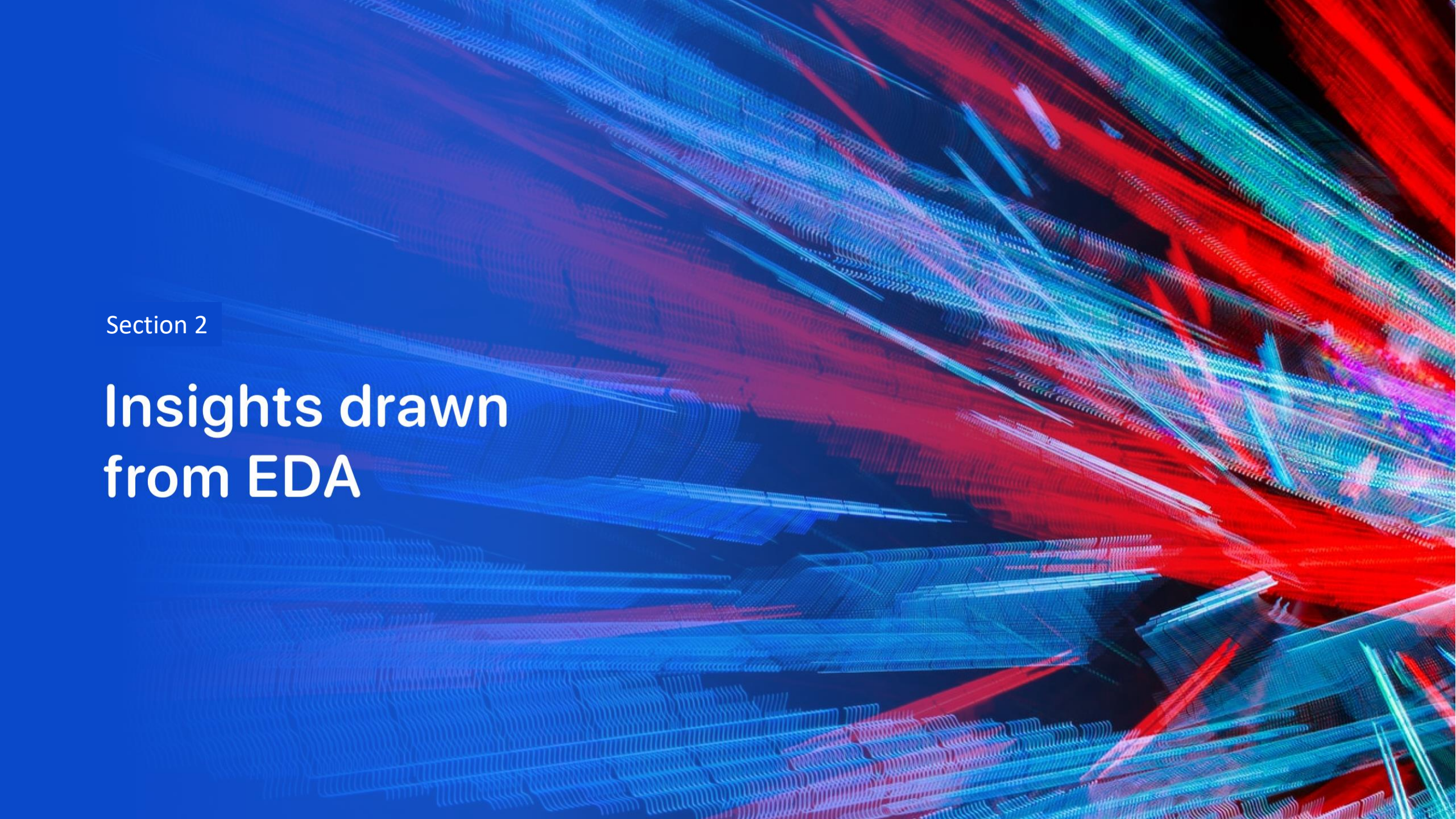
Calculate the accuracy on the test data using the method `score` :

```
logreg_cv.score(X_train, Y_train)
```

```
0.8333333333333334
```

Lets look at the confusion matrix:

```
yhat=logreg_cv.predict(X_test)
plot_confusion_matrix(Y_test,yhat)
```


Section 2
Insights drawn from EDA

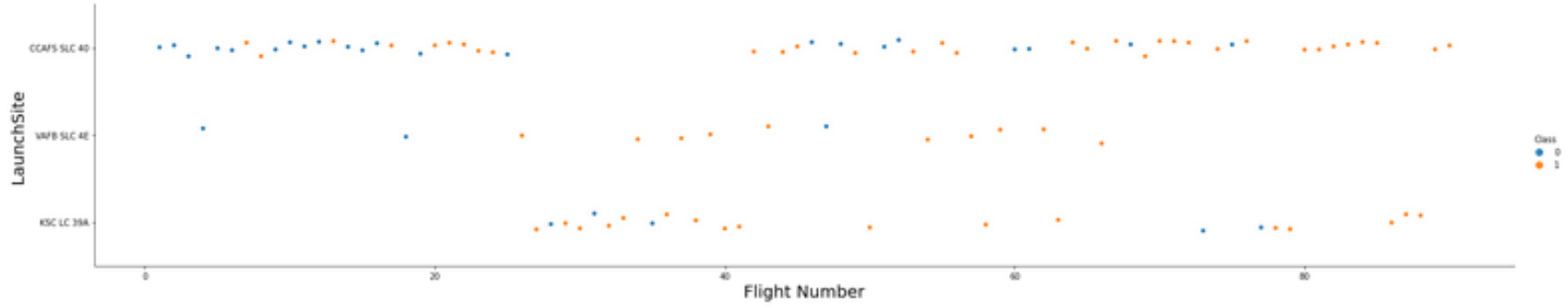
Flight Number vs. Launch Site

- Scatter plot of Flight Number vs. Launch Site
- The screenshot of the scatter plot with explanations

TASK 1: Visualize the relationship between Flight Number and Launch Site

Use the function `catplot` to plot `FlightNumber` vs `LaunchSite`, set the parameter `x` parameter to `FlightNumber`,set the `y` to `Launch Site` and set the parameter `hue` to `'class'`

In [4]:
```
# Plot a scatter point chart with x axis to be Flight Number and y axis to be the launch site, a
sns.catplot(y="LaunchSite", x="FlightNumber", hue="Class", data=df, aspect = 5)
plt.xlabel("Flight Number",fontsize=20)
plt.ylabel("LaunchSite",fontsize=20)
plt.show()
```



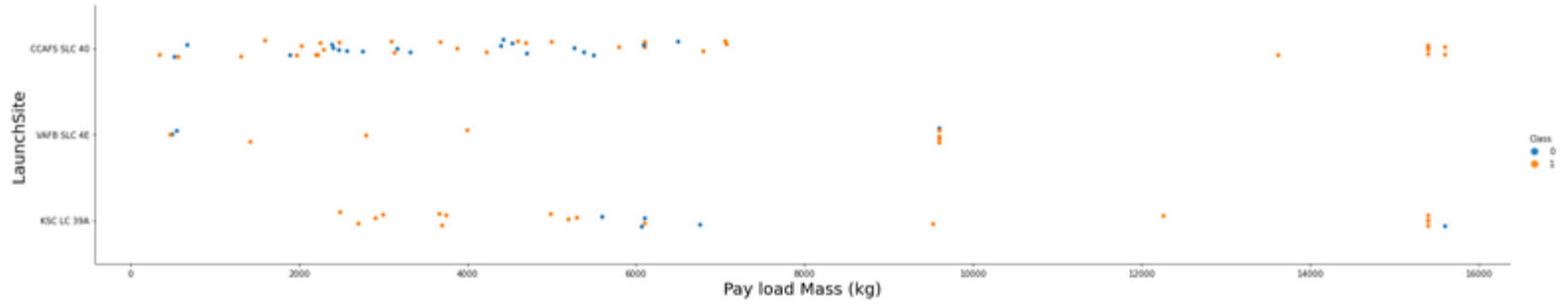
Payload vs. Launch Site

- Show a scatter plot of Payload vs. Launch Site
- Show the screenshot of the scatter plot with explanations

TASK 2: Visualize the relationship between Payload and Launch Site

We also want to observe if there is any relationship between launch sites and their payload mass.

In [5]:
```
# Plot a scatter point chart with x axis to be Pay Load Mass (kg) and y axis to be the launch si
sns.catplot(x="PayloadMass", y="LaunchSite", hue="Class", data=df, aspect = 5)
plt.ylabel("LaunchSite",fontsize=20)
plt.xlabel("Pay load Mass (kg)",fontsize=20)
plt.show()
```



Now if you observe Payload Vs. Launch Site scatter point chart you will find for the VAFB-SLC launchsite there are no rockets launched for heavypayload mass(greater than 10000).

Success Rate vs. Orbit Type

- Show a bar chart for the success rate of each orbit type
- Show the screenshot of the scatter plot with explanations

```
# Import required libraries
import pandas as pd
import plotly.express as px
import plotly.graph_objects as go
fig = px.bar(bar_data, x="Orbit", y="Class", title='Orbit x Class')
fig.show()
```

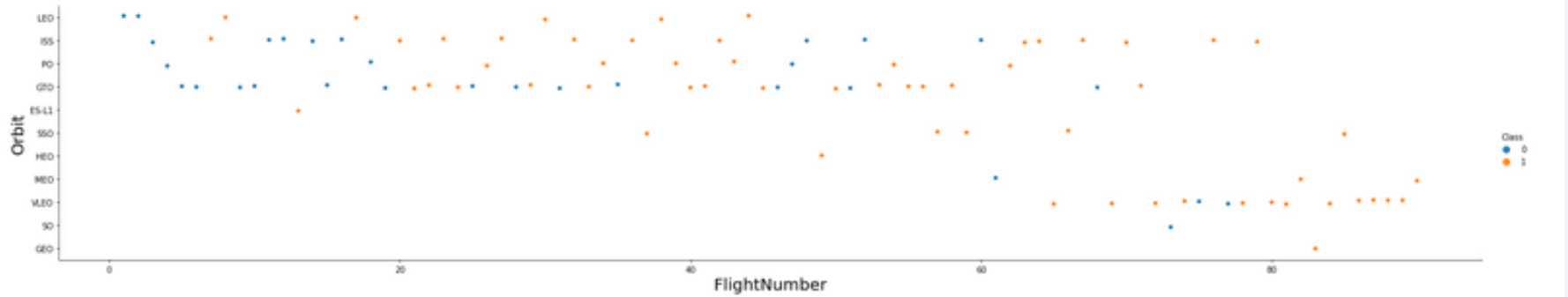
Flight Number vs. Orbit Type

- Show a scatter point of Flight number vs. Orbit type
- Show the screenshot of the scatter plot with explanations

TASK 4: Visualize the relationship between FlightNumber and Orbit type

For each orbit, we want to see if there is any relationship between FlightNumber and Orbit type.

```
# Plot a scatter point chart with x axis to be FlightNumber and y axis to be the Orbit, and hue
sns.catplot(x="FlightNumber", y="Orbit", hue="Class", data=df, aspect = 5)
plt.xlabel("FlightNumber",fontsize=20)
plt.ylabel("Orbit",fontsize=20)
plt.show()
```



You should see that 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

- Scatter point of payload vs. orbit type
- The screenshot of the scatter plot with explanations

Similarly, we can plot the Payload vs. Orbit scatter point charts to reveal the relationship between Payload and Orbit type

```
# Plot a scatter point chart with x axis to be Payload and y axis to be the Orbit, and hue t
sns.catplot(x="PayloadMass", y="Orbit", hue="Class", data=df, aspect = 5)
plt.xlabel("PayloadMass",fontsize=20)
plt.ylabel("Orbit",fontsize=20)
plt.show()
```

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

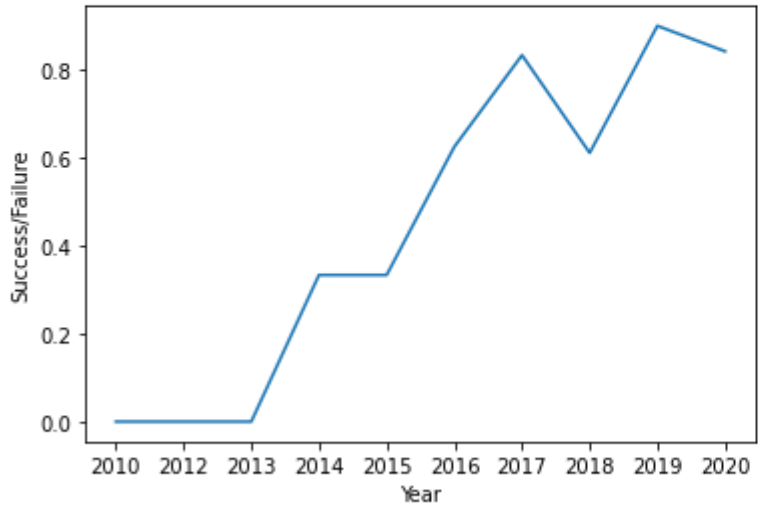
However for GTO we cannot distinguish this well as both positive landing rate and negative landing(unsuccessful mission) are both there here.

Launch Success Yearly Trend

- A line chart of yearly average success rate
- The screenshot of the scatter plot with explanations

```
# Plot a line chart with x axis to be the extracted year and y axis to be the success rate

plt.plot(average_by_year["Year"],average_by_year["Class"])
plt.xlabel("Year")
plt.ylabel("Success/Failure")
plt.show()
```



you can observe that the sucess rate since 2013 kept increasing till 2020

All Launch Site Names

- Find the names of the unique launch sites
- Query result with a short explanation: we need the “distinct” to find to launch sites names

```
1 %sql select DISTINCT LAUNCH_SITE from SPACEXDATASET
```

```
 * ibm_db_sa://tqg67218:***@125f9f61-9715-46f9-9399-c8177b21803b.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:30426/bludb
Done.
```

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

Launch Site Names Begin with 'CCA'

- Find 10 records where launch sites begin with `CCA`
- Query result with a short explanation: we use “like” to name and “limit” to limit rows, to do que query

```
1 %sql select LAUNCH_SITE from SPACEXDATASET where LAUNCH_SITE LIKE 'CCA%' limit 10
```

```
 * ibm_db_sa://tqg67218:***@125f9f61-9715-46f9-9399-c8177b21803b.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:30426/bludb
Done.
```

launch_site
CCAFS LC/40
CCAFS LC/40
CCAFS LC/40
CCAFS LC/40
CCAFS LC/40
CCAFS LC/40
CCAFS LC/40
CCAFS LC/40
CCAFS LC/40
CCAFS LC/40

Total Payload Mass

- Calculate the total payload carried by boosters from NASA
- Query result with a short explanation: we use “SUM” to get a total payload carried, and “LIKE” to do this for NASA only.

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

```
1 %sql select SUM(payload_mass__kg_) from spacexdataset where customer like 'NASA%'
```

```
 * ibm_db_sa://tqg67218:***@125f9f61-9715-46f9-9399-c8177b21803b.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:30426/bludb
Done.
```

1
36679

Average Payload Mass by F9 v1.1

- Calculate the average payload mass carried by booster version F9 v1.1
- Present your query result with a short explanation here

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

```
1 %sql select AVG(payload_mass__kg_) from spacexdataset where booster_version like 'F9 v1.1'
```

```
 * ibm_db_sa://tqg67218:***@125f9f61-9715-46f9-9399-c8177b21803b.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:30426/bludb
Done.
```

1
3676

First Successful Ground Landing Date

- Find the dates of the first successful landing outcome on ground pad
- Present your query result with a short explanation here

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

Hint:Use min function

```
1 %sql select MIN(date) from spacexdataset where landing__outcome like 'Success'
```

```
 * ibm_db_sa://tqg67218:***@125f9f61-9715-46f9-9399-c8177b21803b.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:30426/bludb
Done.
```

1
2018-03-12

Successful Drone Ship Landing with Payload between 4000 and 6000

- List the names of boosters which have successfully landed on drone ship and had payload mass greater than 4000 but less than 6000
- Present your query result with a short explanation here

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

```
1 %sql select booster_version from spacexdataset where landing__outcome like 'Success%drone%'
```

```
 * ibm_db_sa://tqg67218:***@125f9f61-9715-46f9-9399-c8177b21803b.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:30426/bludb
Done.
```

booster_version
F9 FT B1021.1
F9 FT B1022
F9 B4 B1041.1
F9 FT B1031.2
F9 B5 B1046.1

Total Number of Successful and Failure Mission Outcomes

- Calculate the total number of successful and failure mission outcomes
- Present your query result with a short explanation here

List the total number of successful and failure mission outcomes

```
1 %sql select COUNT(mission_outcome) from spacexdataset
```

```
 * ibm_db_sa://tqg67218:***@125f9f61-9715-46f9-9399-c8177b21803b.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:30426/bludb
Done.
```

1
45

Boosters Carried Maximum Payload

- List the names of the booster which have carried the maximum payload mass
- Present your query result with a short explanation here

List the names of the booster_versions which have carried the maximum payload mass. Use a subquery ¶

```
1 %sql select booster_version,payload_mass__kg_ from spacexdataset where payload_mass__kg_ = (sele
```

```
 * ibm_db_sa://tqg67218:***@125f9f61-9715-46f9-9399-c8177b21803b.c1ogj3sd0tgtu0lqde00.databases.ap
pdomain.cloud:30426/bludb
Done.
```

booster_version	payload_mass__kg_
F9 B5 B1048.4	15600
F9 B5 B1049.4	15600
F9 B5 B1049.5	15600
F9 B5 B1060.2	15600
F9 B5 B1058.3	15600

%sql select booster_version,payload_mass__kg_ from spacexdataset where payload_mass__kg_ = (select MAX(payload_mass__kg_) from spacexdataset)

2015 Launch Records

- List the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015
- Present your query result with a short explanation here

List the failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015

```
%sql select booster_version,launch_site,landing__outcome,date \
    from spacexdataset where landing__outcome   \
    like 'Failure%drone ship%' and date like '2015%'
```

```
 * ibm_db_sa://tqg67218:***@125f9f61-9715-46f9-9399-c8177b21803b.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:30426/bludb
Done.
```

booster_version	launch_site	landing__outcome	DATE
F9 v1.1 B1012	CCAFS LC/40	Failure (drone ship)	2015-10-01

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

- 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
- Present your query result with a short explanation here

Rank the count of landing outcomes (such as Failure (drone ship) or Success (ground pad)) between the date 2010-06-04 and 2017-03-20, in descending order

```
%sql select COUNT(landing__outcome) \
    from spacexdataset where ( landing__outcome like 'Failure%drone ship%' \
    or landing__outcome like 'Success%ground%' ) \
    and ( date > '2010-06-04' and date < '2017-03-20' )
```

```
 * ibm_db_sa://tqg67218:***@125f9f61-9715-46f9-9399-c8177b21803b.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud:30426/bludb
Done.
```

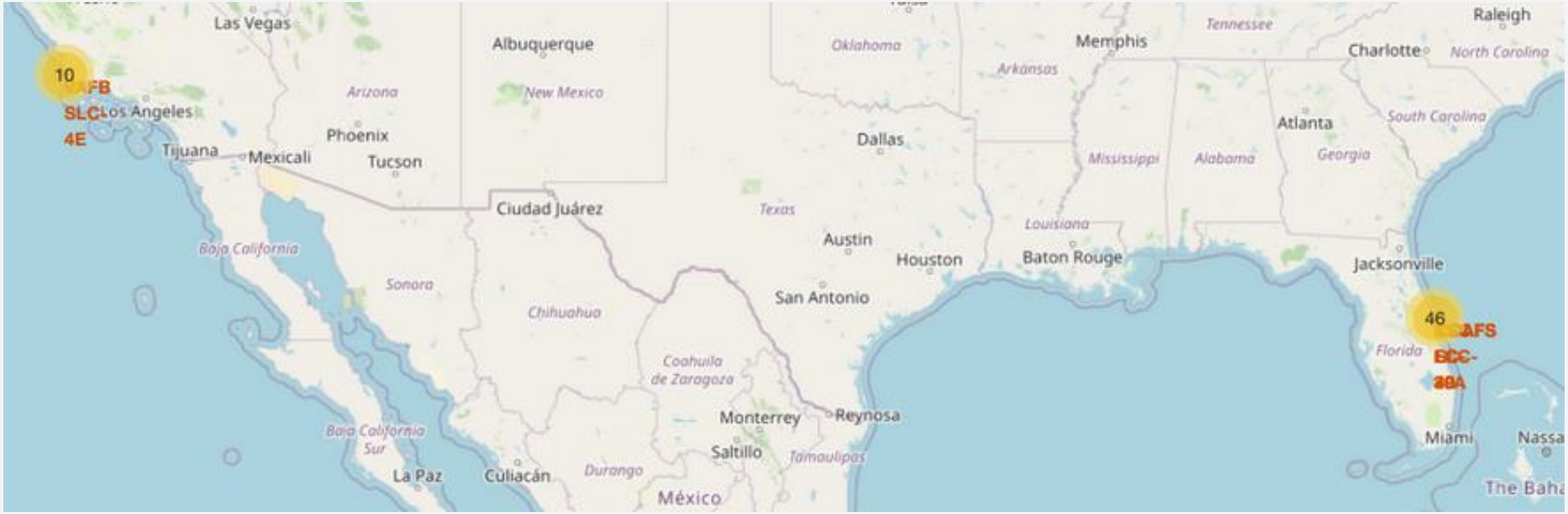
1
4

Section 3

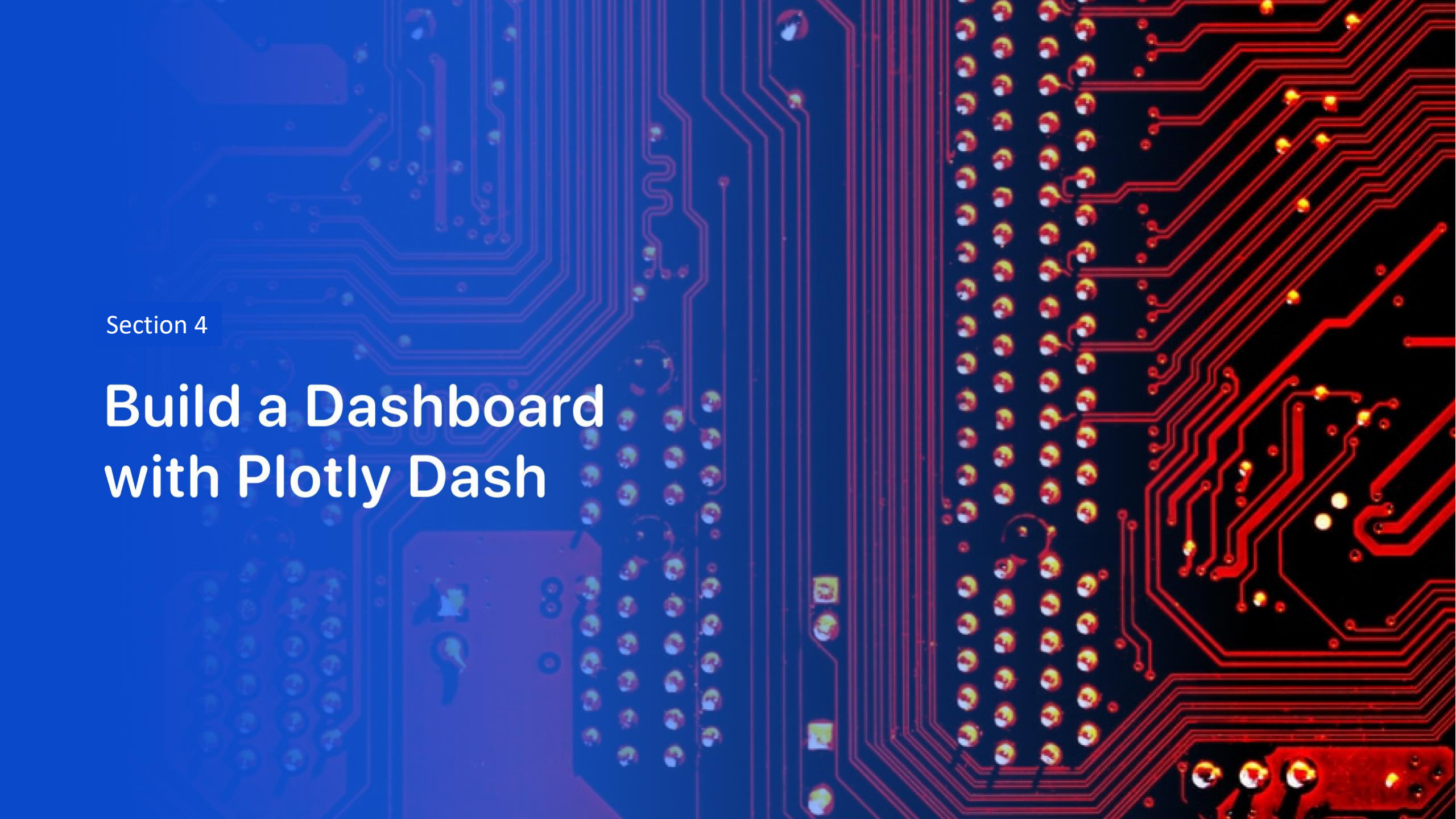
Launch Sites Proximities Analysis

Launch Sites

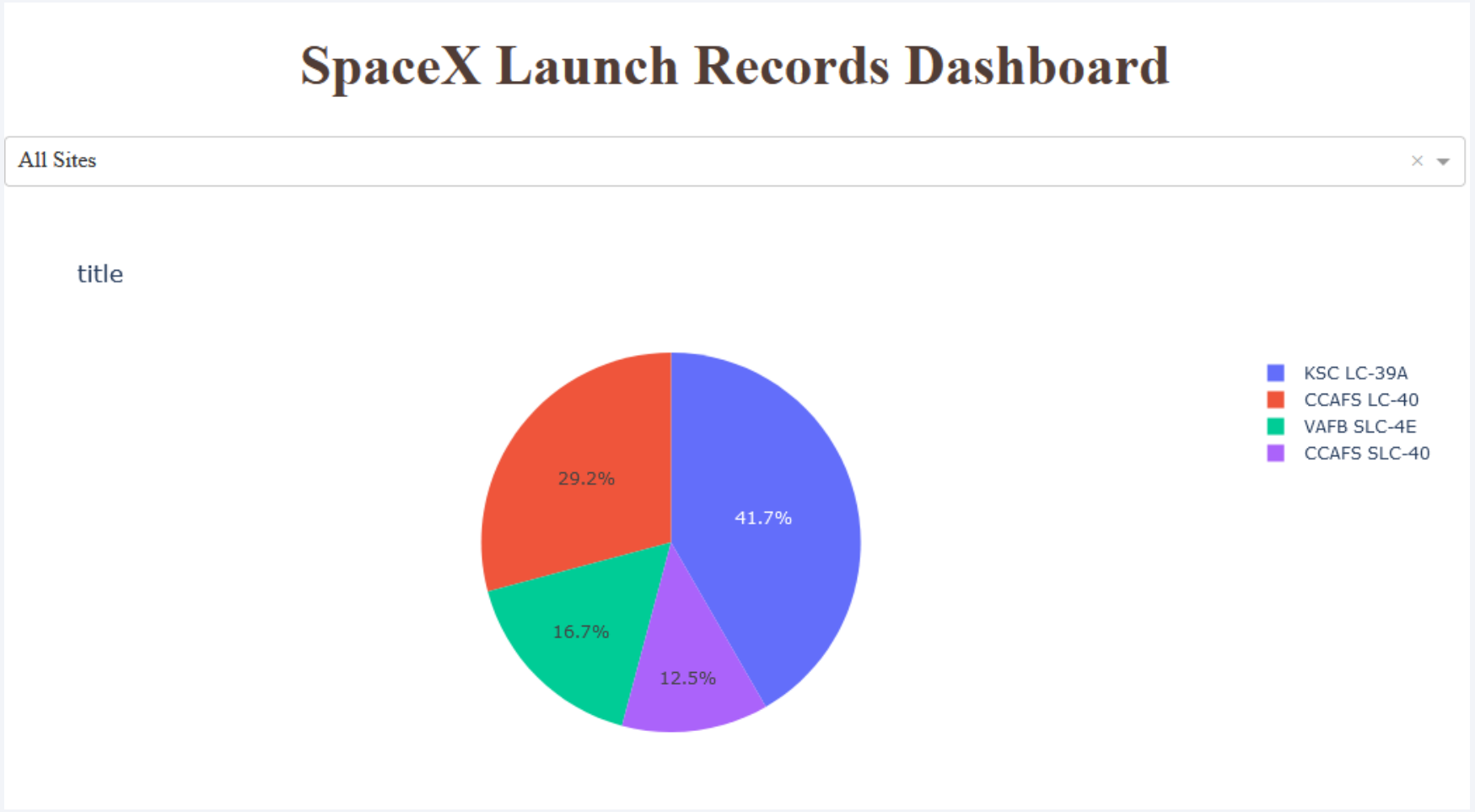
Launch Outcomes



Launch site to its proximities

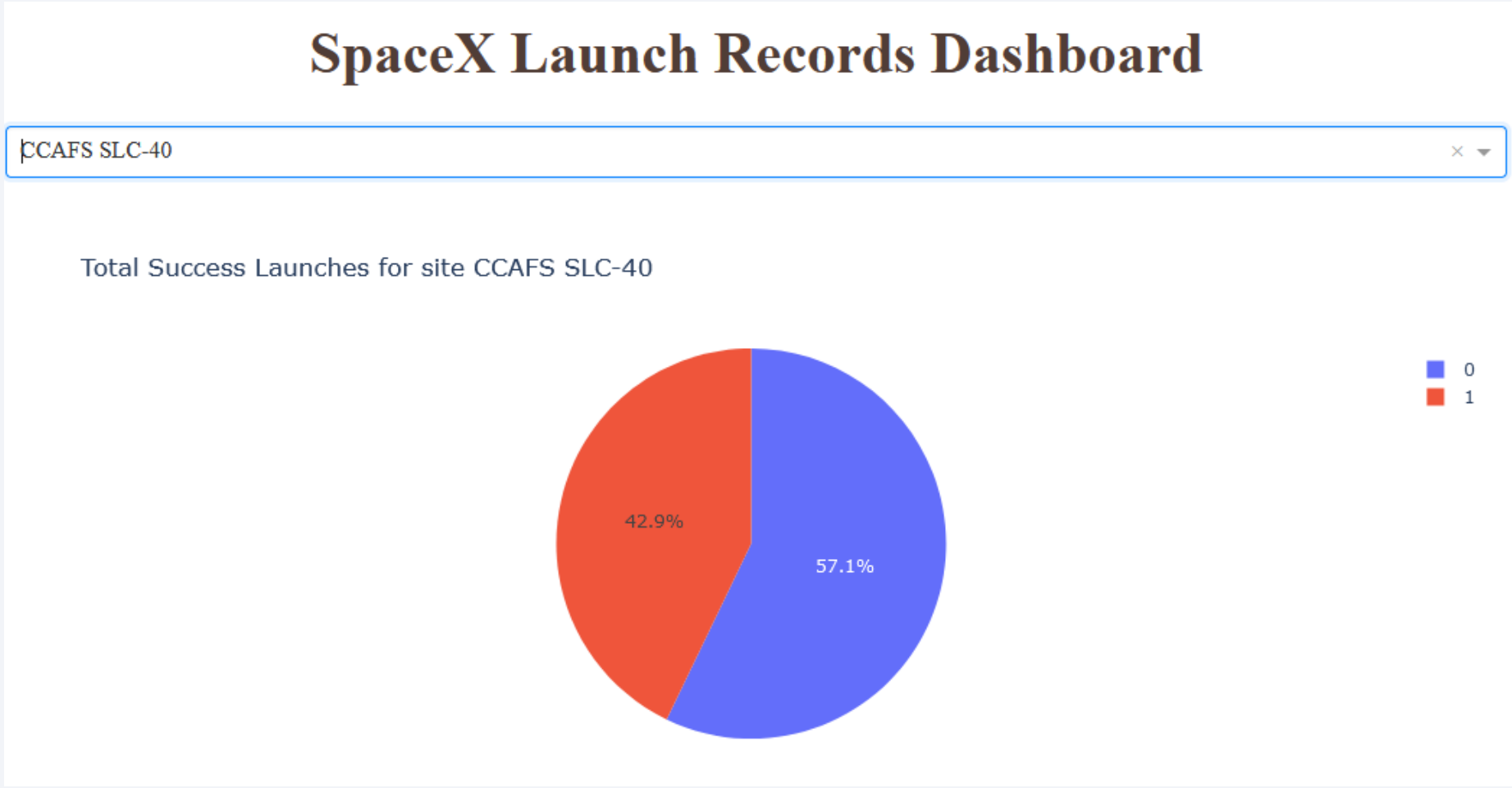

Section 4
Build a Dashboard with Plotly Dash

Launch success count for all sites

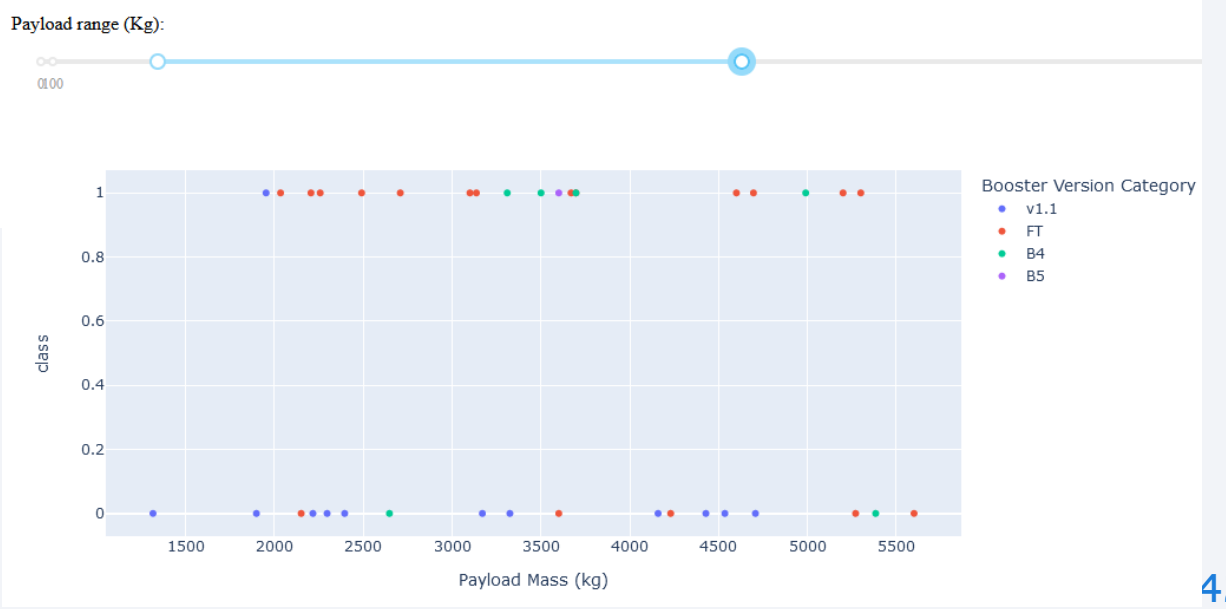


- We see the most percentage is from KSC LC-39A site.

CCAFS SLC-40


SpaceX Launch Records Dashboard
CCAFS SLC-40
Total Success Launches for site CCAFS SLC-40
42.9%
57.1%
0
1

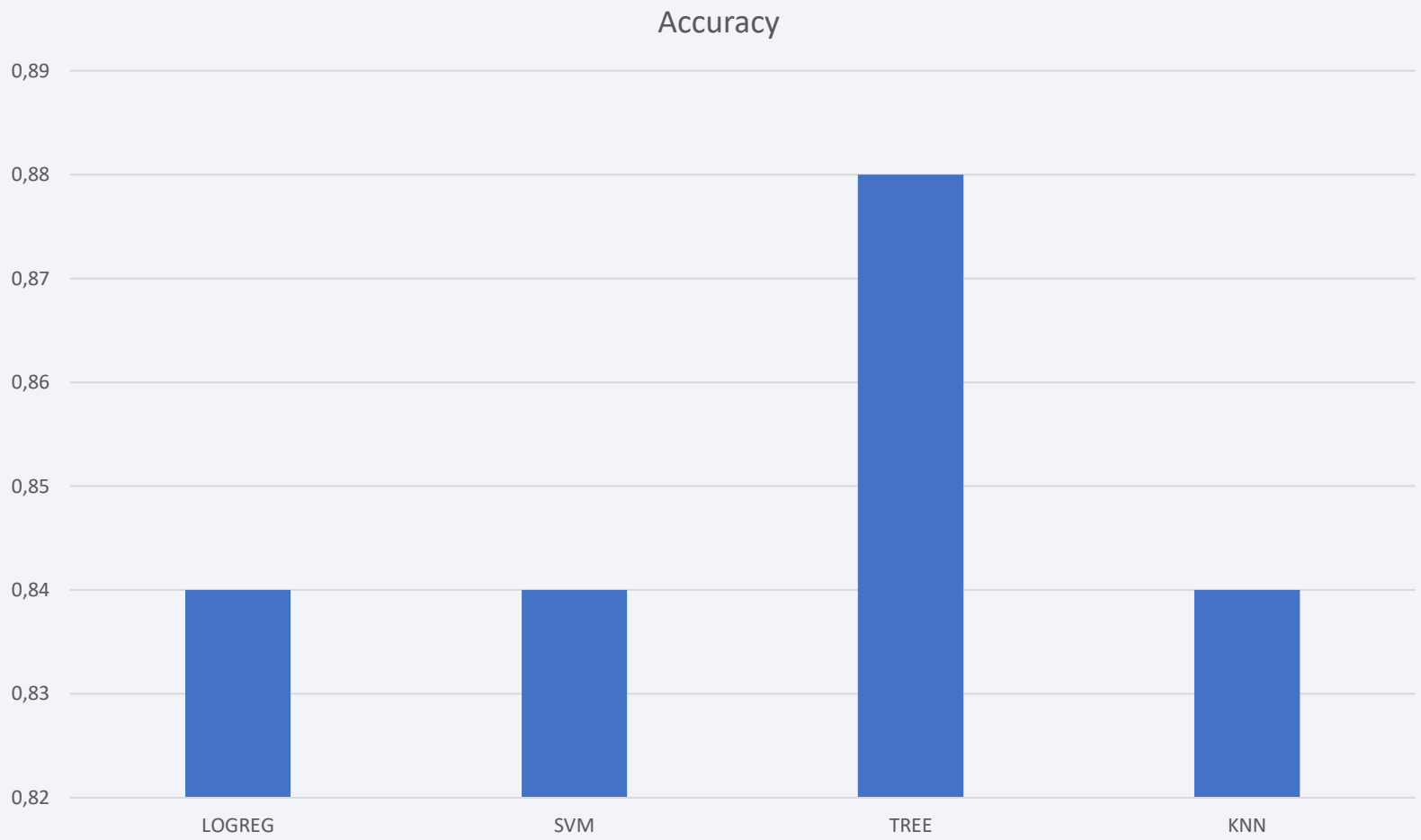
Payload Mass Range



Section 5

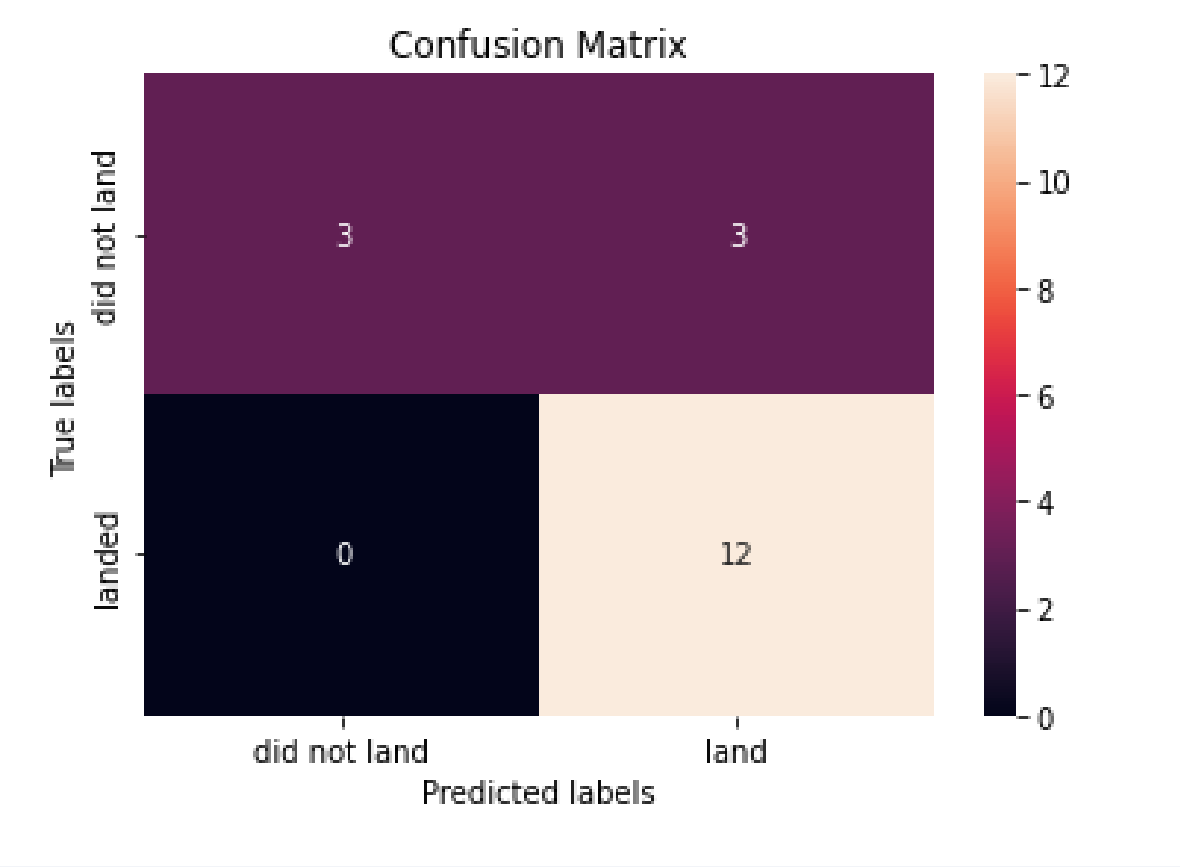
Predictive Analysis (Classification)

Classification Accuracy



Confusion Matrix

- This is the confusion matrix with the best acurracy



Conclusions

- Data Science can help to conclude a lot situations, to solve problems, or to show solutions
- In SPACE X case:
  - we can see the Payload Mass has relation with success or not about launch
  - there is a Launch site with more success than others
  - 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%.
  - if you observe Payload Vs. Launch Site scatter point chart you will find for the VAFB-SLC launchsite there are no rockets launched for heavypayload mass(greater than 10000)
  - 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.

Appendix

- Relevant assets like Python code snippets, SQL queries, charts, Notebook outputs, or data sets that you may have created during this project:
- Linear Regression: https://didatica.tech/matematica-para-machine-learning-curso/
- Matplot: https://matplotlib.org/2.0.2/api/pyplot_api.html
- Folium: https://orbeai.medium.com/gr%C3%A1ficos-de-informa%C3%A7%C3%B5es-geogr%C3%A1ficas-com-folium-a68225d42b2c

Thank you!