IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

<Stephen>
<2023/11/20>

Outline

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

Executive Summary

- Summary of methodologies
  - Data collection methodology:
  - Perform data wrangling
  - Perform exploratory data analysis (EDA) using visualization and SQL
  - Perform interactive visual analytics using Folium and Plotly Dash
  - Perform predictive analysis using classification models
- Summary of all results
  - The boosters (version:B5) launch at KSD LC-39A and landing on the ground has higher success rate than the other.
  - Users can apply trained model (Decision Tree model) and predict success rate with dataset from SpaceX API and web scraping .

Introduction

- Project background and context
  - Space X advertises Falcon 9 rocket launches on its website with a cost of 62 million dollars; other providers cost upward of 165 million dollars each, much of the savings is because Space X can reuse the first stage. Therefore if we can determine if the first stage will land, we can determine the cost of a launch. This information can be used if an alternate company wants to bid against space X for a rocket launch. In this context, we create a machine learning pipeline to predict if the first stage will land given the data from the preceding labs.
- Problems you want to find answers
  - Find the way to predict Falcon 9 First Stage will land successfully.

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Request to the SpaceX API
  - Web scraping Falcon 9 and Falcon Heavy Launches Records from Wikipedia
- Perform data wrangling
  - Search useful data and transfer them into meaningful data for training.
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Import machine learning models such as Logistic Regression, SVM, Decision Tree and KNN.
  - Data Wrangling : Load Data and standardize them before training.
  - Set parameters for tunning model
  - Build GridSearchCV and import model, parameters and training/test data. Finally, get the best parameter.
  - Evaluate the classification model by confusion matrix.

Data Collection

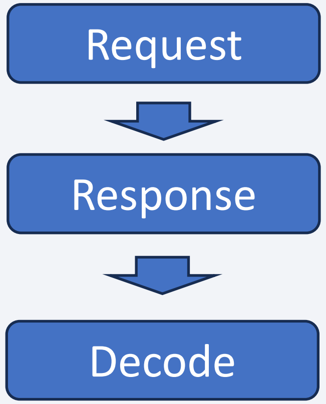
- Describe how data sets were collected.
- You need to present your data collection process use key phrases and flowcharts

Data Collection – SpaceX API

- Present your data collection with SpaceX REST calls using key phrases and flowcharts

- Add the GitHub URL of the completed SpaceX API calls notebook (must include completed code cell and outcome cell), as an external reference and peer-review purpose

https://github.com/bo73/Winning-Space-race-with-Data-Science.git



Request rocket launch data from SpaceX API with the URL.
https://api.spacexdata.com/v4/launches/past

Get the response from the server

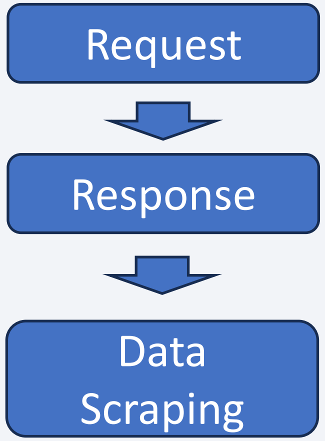
decode the response content as a Json using .json()
and turn it into a Pandas dataframe using .json_normalize()

Data Collection - Scraping

- Present your web scraping process using key phrases and flowcharts

- Add the GitHub URL of the completed web scraping notebook, as an external reference and peer-review purpose

https://github.com/bo73/Winning-Space-race-with-Data-Science.git



Request rocket launch data from SpaceX API with the URL.
https://en.wikipedia.org/w/index.php?title=List_of_Falcon_9_and_Falcon_Heavy_launches&oldid=1027686922

Get the response from the server

Create a BeautifulSoup object from the HTML response
Print the data out if the BeautifulSoup object was created properly

Data Wrangling

- Describe how data were processed
  - Search useful data and transfer them into meaningful data for training.
- You need to present your data wrangling process using key phrases and flowcharts
- Add the GitHub URL of your completed data wrangling related notebooks, as an external reference and peer-review purpose

https://github.com/bo73/Winning-Space-race-with-Data-Science.git

Discovery
- Examining and understanding your data with respect to your use case
- Creating a plan for cleaning, structuring, organizing, and mapping your data

↓

Transformation
- Structuring Data, Normalizing and Denormalizing Data, cleaning Data, Enriching Data

↓

Validation
- Checking the quality of data after transformation
- Verifying consistency, quality and security of data

↓

Publishing
- Delivering the output of the wrangled data for downstream project needs

EDA with Data Visualization

- Summarize what charts were plotted and why you used those charts
  - Scatter Chart : It can reveal relationship between:
    - Flight Number VS Payload mass
    - Flight Number VS Launch Site
    - Payload mass VS Launch Site
    - Payload mass VS Orbit type
  - Bar Chart : Visualize the relationship between Success rate of each orbit type
  - Line Chart : Plot a line chart with x axis to be the extracted year and y axis to be the Success rate
- Add the GitHub URL of your completed EDA with data visualization notebook, as an external reference and peer-review purpose

  https://github.com/bo73/Winning-Space-race-with-Data-Science.git

EDA with SQL

- Using bullet point format, summarize the SQL queries you performed
  - %sql select DISTINCT Launch_Site FROM SPACEXTABLE
  - %sql select * from SPACEXTABLE where Launch_Site like 'CCA%' limit 5
  - %sql select SUM(PAYLOAD_MASS__KG_) from SPACEXTABLE where Customer='NASA (CRS)'
  - %sql select AVG(PAYLOAD_MASS__KG_) from SPACEXTABLE where Booster_Version='F9 v1.1'
  - %sql select MIN(Date) from SPACEXTABLE where Landing_Outcome='Success (ground pad)'
  - %sql select Booster_Version from SPACEXTABLE where Landing_Outcome='Success (drone ship)' and PAYLOAD_MASS__KG_ between 4000 and 6000
  - %sql select COUNT(Mission_Outcome) FROM SPACEXTABLE WHERE Mission_Outcome like 'Success%'
  - %sql select COUNT(Mission_Outcome) FROM SPACEXTABLE WHERE Mission_Outcome like 'failure%'
  - %sql select Booster_Version from SPACEXTBL where PAYLOAD_MASS__KG_=(select MAX(PAYLOAD_MASS__KG_) from SPACEXTBL)
  - %sql select substr(Date, 0,5) as year,substr(Date, 6,2) as months,Booster_Version,Landing_Outcome,Launch_Site from SPACEXTBL where Landing_Outcome= "Failure (drone ship)" and year='2015'
  - %sql select Landing_Outcome, count(*) as count_of_landing_outcomes from SPACEXTBL where Date between '2010-06-04' and '2017-03-20' GROUP BY Landing_Outcome ORDER BY count_of_landing_outcomes desc;
- Add the GitHub URL of your completed EDA with SQL notebook, as an external reference and peer-review purpose

https://github.com/bo73/Winning-Space-race-with-Data-Science.git

Build an Interactive Map with Folium

- Summarize what map objects such as markers, circles, lines, etc. you created and added to a folium map
    - Markers
    - Circles
    - lines
- Explain why you added those objects
    - Mark all launch sites on a map
    - Add a highlighted circle area with a text label on a specific coordinate
    - Draw a line between a launch site to its closest coastline and highway
- Add the GitHub URL of your completed interactive map with Folium map, as an external reference and peer-review purpose

https://github.com/bo73/Winning-Space-race-with-Data-Science.git

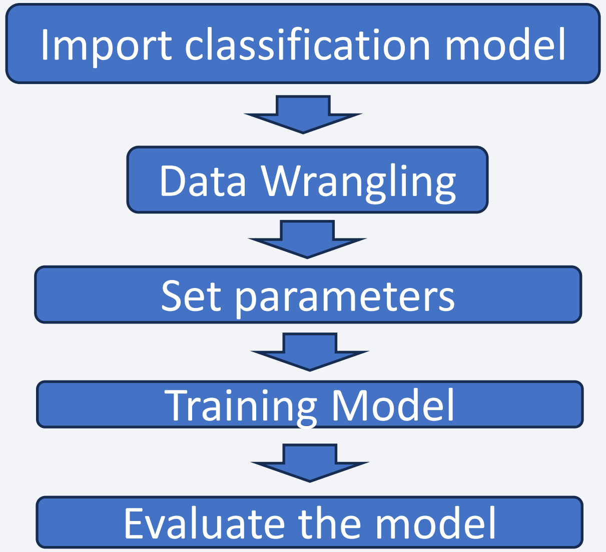
Build a Dashboard with Plotly Dash

- Summarize what plots/graphs and interactions you have added to a dashboard
  - Pie chart
  - Scatter chart
- Explain why you added those plots and interactions
  - Users can select launch site from dropdown list and interactive with the pie chart
    - ALL : Show the total Success launches count for all sites
    - CCAFS LC-40/ CCAFS SLC-40/ KSC LC-39A/ VAFB SLC-4E : Show the Success % fail launches count for specific launch site.
  - Users can select launch site from dropdown list and payload range. Then, scatter chart shows the correlation between payload and launch Success.
- Add the GitHub URL of your completed Plotly Dash lab, as an external reference and peer-review purpose

https://github.com/bo73/Winning-Space-race-with-Data-Science.git

Predictive Analysis (Classification)

- Summarize how you built, evaluated, improved, and found the best performing classification model
  - Import classification model: Import machine learning models such as Logistic Regression, SVM, Decision Tree and KNN.
  - Data Wrangling : Load Data and standardize them before training.
  - Set parameters for tunning model
  - Training Model: Build GridSearchCV and import model, parameters and training data. Furthermore, get the best parameter.
  - Evaluate the classification model by test data and confusion matrix.



- You need present your model development process using key phrases and flowchart
- Add the GitHub URL of your completed predictive analysis lab, as an external reference and peer-review purpose

https://github.com/bo73/Winning-Space-race-with-Data-Science.git

Results

- Exploratory data analysis results
  - The Success rate since 2013 kept increasing till 2020
  - SpaceX team is used to execute heavy payload missions at CCAFS SLC 40 and KSC LC-39A and it is more successful there than VAFB SLC-4E launch site.
  - The booster version F9 B5 FT executed heavy payload mission many times.
- Predictive analysis results
  - Decision Tree has the highest classification accuracy(88.9%).
  - Precision(83%)
  - Recall(83%)

Section 2

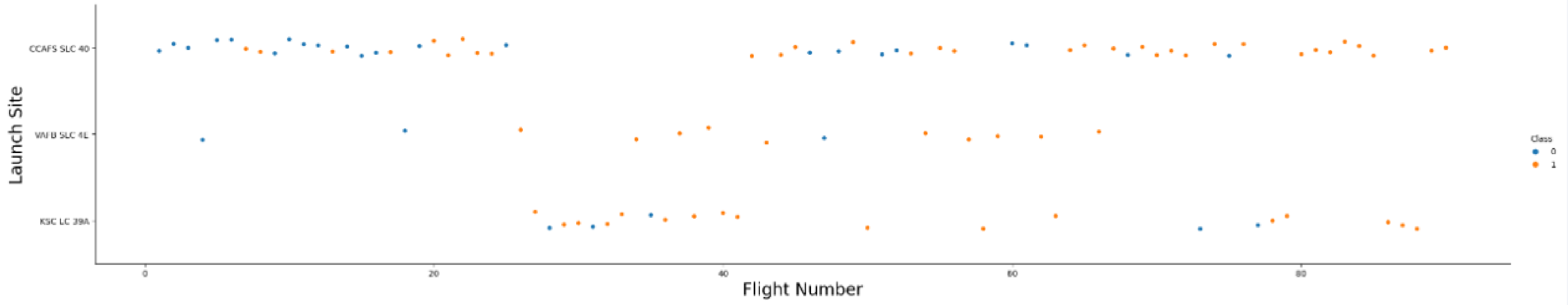
Insights drawn from EDA

Flight Number vs. Launch Site

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

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'`

[19]:
```
# Plot a scatter point chart with x axis to be Flight Number and y axis to be the launch site, and hue to be the class value
sns.catplot(y="LaunchSite", x="FlightNumber", hue='Class' , data=df, aspect = 5)
plt.ylabel("Launch Site",fontsize=20)
plt.xlabel("Flight Number",fontsize=20)
plt.show()
```



Now try to explain the patterns you found in the Flight Number vs. Launch Site scatter point plots.

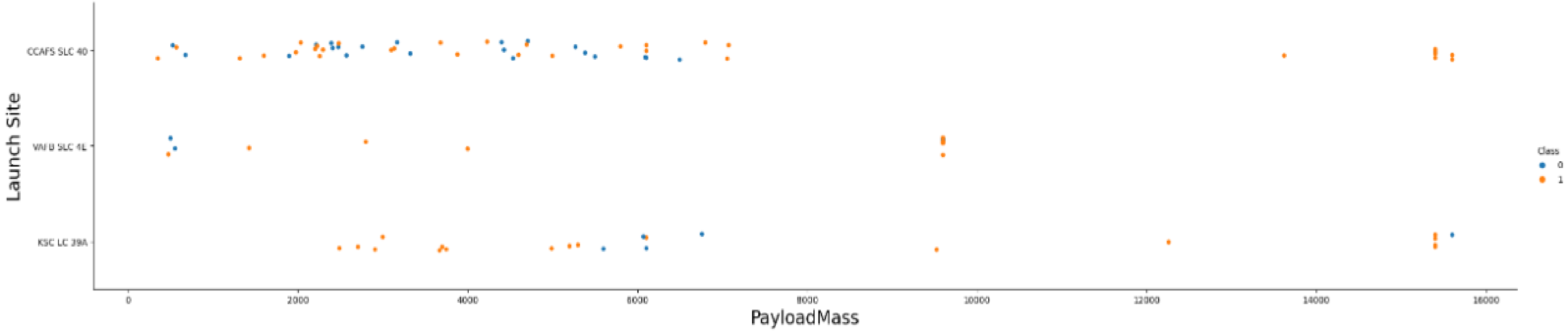
The spacex are used to carry out missions in CCAFS SLC 40 launch pad, even though it has a lower success rate. In addition, it made a progress after flight number 40.

Payload vs. Launch Site

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

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

```
[21]: # Plot a scatter point chart with x axis to be Pay Load Mass (kg) and y axis to be the launch site, and hue to be the class value
sns.catplot(y="LaunchSite", x="PayloadMass", hue='Class' , data=df, aspect = 5)
plt.ylabel("Launch Site",fontsize=20)
plt.xlabel("PayloadMass",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
  - ES-L1,GEO,HEO and SSO are have high Success rate.

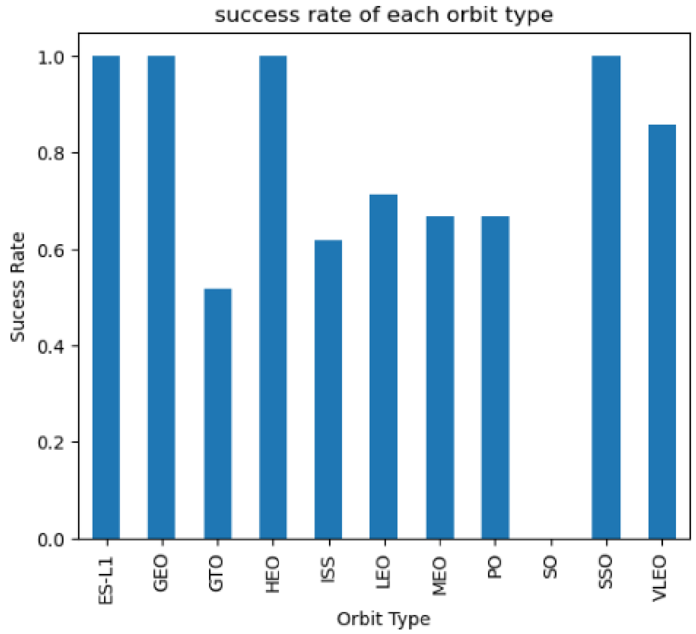
[26]:
```
### TASK  3: Visualize the relationship between success rate of each orbit type
```

Next, we want to visually check if there are any relationship between success rate and orbit type.

Let's create a `bar chart` for the sucess rate of each orbit

[27]:
```
# HINT use groupby method on Orbit column and get the mean of Class column
df = df.replace(np.nan, 0)
dfg = df.groupby(['Orbit'])['Class'].mean()
#print(dfg)
dfg.plot(kind='bar', title='success rate of each orbit type', ylabel='Sucess Rate',
         xlabel='Orbit Type', figsize=(6, 5))
```

[27]:
```
<Axes: title={'center': 'success rate of each orbit type'}, xlabel='Orbit Type', ylabel='Sucess Rate'>
```



Analyze the ploted bar chart try to find which orbits have high sucess rate. ES-L1,GEO,HEO and SSO are have high sucess rate.

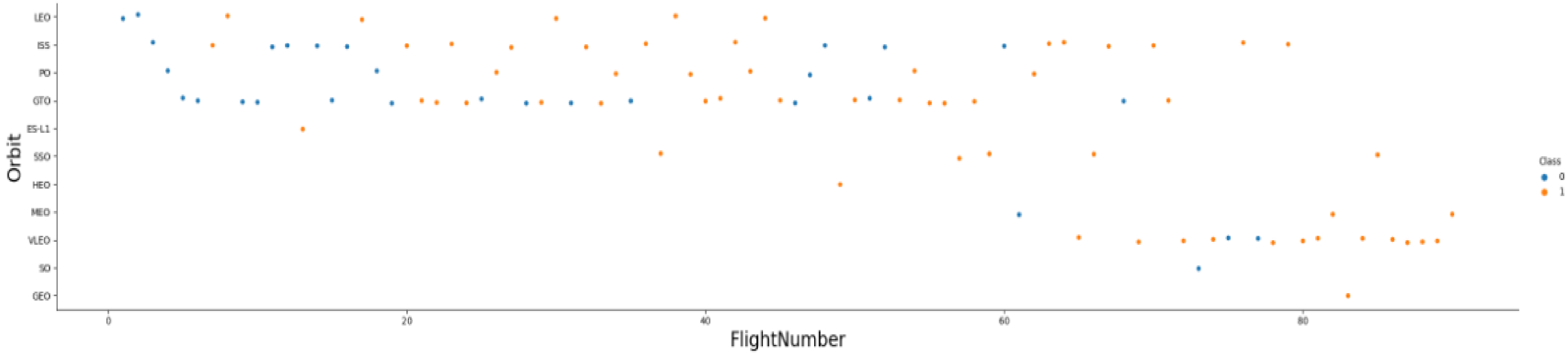
Flight Number vs. Orbit Type

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

[13]:
```
### 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.

[14]:
```
# Plot a scatter point chart with x axis to be FlightNumber and y axis to be the Orbit, and hue to be the class value
sns.catplot(y="Orbit", x="FlightNumber", hue='Class' , data=df, aspect = 5)
plt.ylabel("Orbit",fontsize=20)
plt.xlabel("FlightNumber",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.

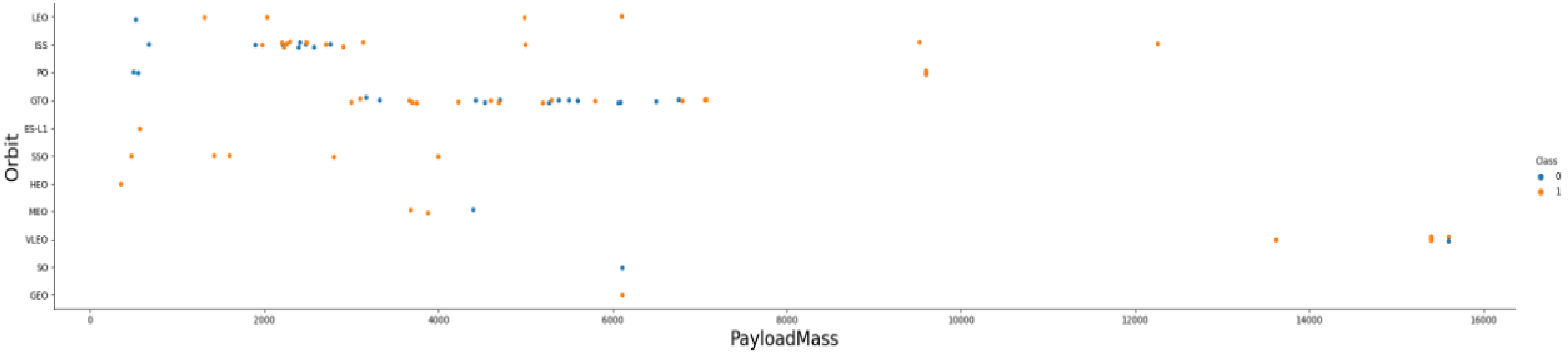
Payload vs. Orbit Type

- Show a scatter point of payload vs. orbit type
- Show the screenshot of the scatter plot with explanations

TASK 5: Visualize the relationship between Payload and Orbit type

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 to be the class value
sns.catplot(y="Orbit", x="PayloadMass", hue='Class' , data=df, aspect = 5)
plt.ylabel("Orbit",fontsize=20)
plt.xlabel("PayloadMass",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

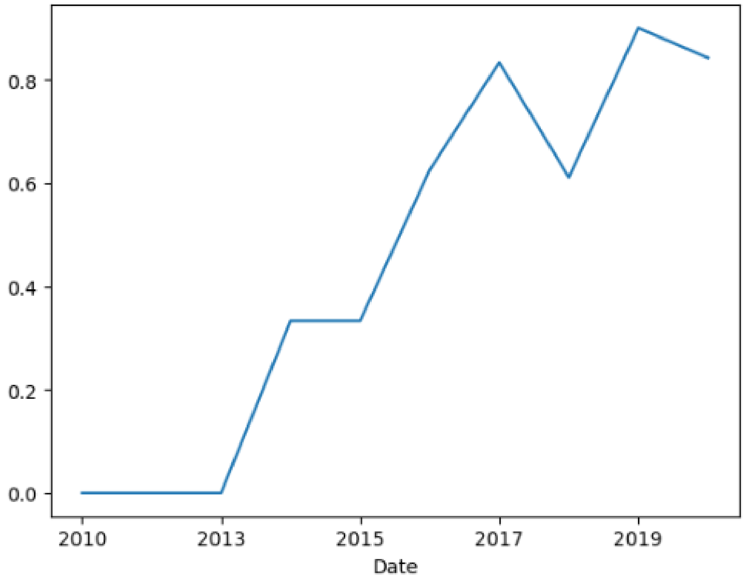
- Show a line chart of yearly average Success rate

- Show 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
dfg = df.groupby(['Date'])['Class'].mean()
print(dfg)
dfg.plot.line()
```

```
Date
2010    0.000000
2012    0.000000
2013    0.000000
2014    0.333333
2015    0.333333
2016    0.625000
2017    0.833333
2018    0.611111
2019    0.900000
2020    0.842105
Name: Class, dtype: float64
```

```
<Axes: xlabel='Date'>
```



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

- Present your query result with a short explanation here
    - There are 4 launch sites in US.
        - CCAFS LC-40
        - VAFB SLC-4E
        - KSC LC-39A
        - CCAFS SLC-40

Task 1

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

[8]:
```
%sql select DISTINCT Launch_Site FROM SPACEXTABLE
```

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

[8]:

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

Launch Site Names Begin with 'CCA'

- Find 5 records where launch sites begin with `CCA`
- Present your query result with a short explanation here

Task 2

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

```
%sql select * from SPACEXTABLE where Launch_Site like 'CCA%' limit 5
```

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

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

Total Payload Mass

- Calculate the total payload carried by boosters from NASA
- Present your query result with a short explanation here
  - The total payload mass for NASA(CRS) is 45596 kg.

Task 3

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

```
%sql select SUM(PAYLOAD_MASS__KG_) from SPACEXTABLE where Customer='NASA (CRS)'
```

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

SUM(PAYLOAD_MASS__KG_)
45596

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
  - Average payload mass carried by booster version F9 v1.1 is 2928.4 kg.

Task 4

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

```
%sql select AVG(PAYLOAD_MASS__KG_) from SPACEXTABLE where Booster_Version='F9 v1.1'
```

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

AVG(PAYLOAD_MASS__KG_)
2928.4

First Success Ground Landing Date

- Find the dates of the first Success landing outcome on ground pad
- Present your query result with a short explanation here
  - The first Success landing date is 2015-12-22.

Task 5

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

Hint:Use min function

```
%sql select MIN(Date) from SPACEXTABLE where Landing_Outcome='Success (ground pad)'
```

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

MIN(Date)
2015-12-22

Success Drone Ship Landing with Payload between 4000 and 6000

- List the names of boosters which have Success landed on drone ship and had payload mass greater than 4000 but less than 6000
- Present your query result with a short explanation here
  - The boosters which Success landing in drone ship are:
    - F9 FT B1022
    - F9 FT B1026
    - F9 FT B1021.2
    - F9 FT B1031.2

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

```
%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.
```

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

Total Number of Success and Failure Mission Outcomes

- Calculate the total number of Success and failure mission outcomes
- Present your query result with a short explanation here
  - Success landing counts: 100
  - Fail landing counts: 1

Task 7

List the total number of successful and failure mission outcomes

```
%sql select COUNT(Mission_Outcome) FROM SPACEXTABLE WHERE Mission_Outcome like 'Success%';
```

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

COUNT(Mission_Outcome)
100

```
%sql select COUNT(Mission_Outcome) FROM SPACEXTABLE WHERE Mission_Outcome like 'failure%';
```

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

COUNT(Mission_Outcome)
1

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
    - The names of the booster_versions which have carried the maximum payload mass
        - F9 B5 FT B1048.4
        - F9 B5 FT B1049.4
        - F9 B5 FT B1051.3
        - F9 B5 FT B1056.4
        - F9 B5 FT B1048.5
        - F9 B5 FT B1051.4
        - F9 B5 FT B1049.5
        - F9 B5 FT B1060.2
        - F9 B5 FT B1058.3
        - F9 B5 FT B1051.6
        - F9 B5 FT B1060.3
        - F9 B5 FT B1049.7

Task 8

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

```
%sql select Booster_Version from SPACEXTBL where PAYLOAD_MASS__KG_=(select MAX(PAYLOAD_MASS__KG_) from SPACEXTBL);
```

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

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

- 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
  - The Booster_versions: (ALL Failure)
    - F9 v1.1 B1012
    - F9 v1.1 B1015

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.

```
%sql select substr(Date, 0,5) as year,substr(Date, 6,2) as months,Booster_Version,Landing_Outcome,Launch_Site from SPACEXTBL where Landing_Outcome= "Failure (drone ship)" and year='2015' ;
```

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

year	months	Booster_Version	Landing_Outcome	Launch_Site
2015	10	F9 v1.1 B1012	Failure (drone ship)	CCAFS LC-40
2015	04	F9 v1.1 B1015	Failure (drone ship)	CCAFS LC-40

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

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.

```
%sql select Landing_Outcome, count(*) as count_of_landing_outcomes from SPACEXTBL where Date between '2010-06-04' and '2017-03-20' GROUP BY Landing_Outcome ORDER BY count_of_landing_outcomes des
```

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

Landing_Outcome	count_of_landing_outcomes
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

Section 3

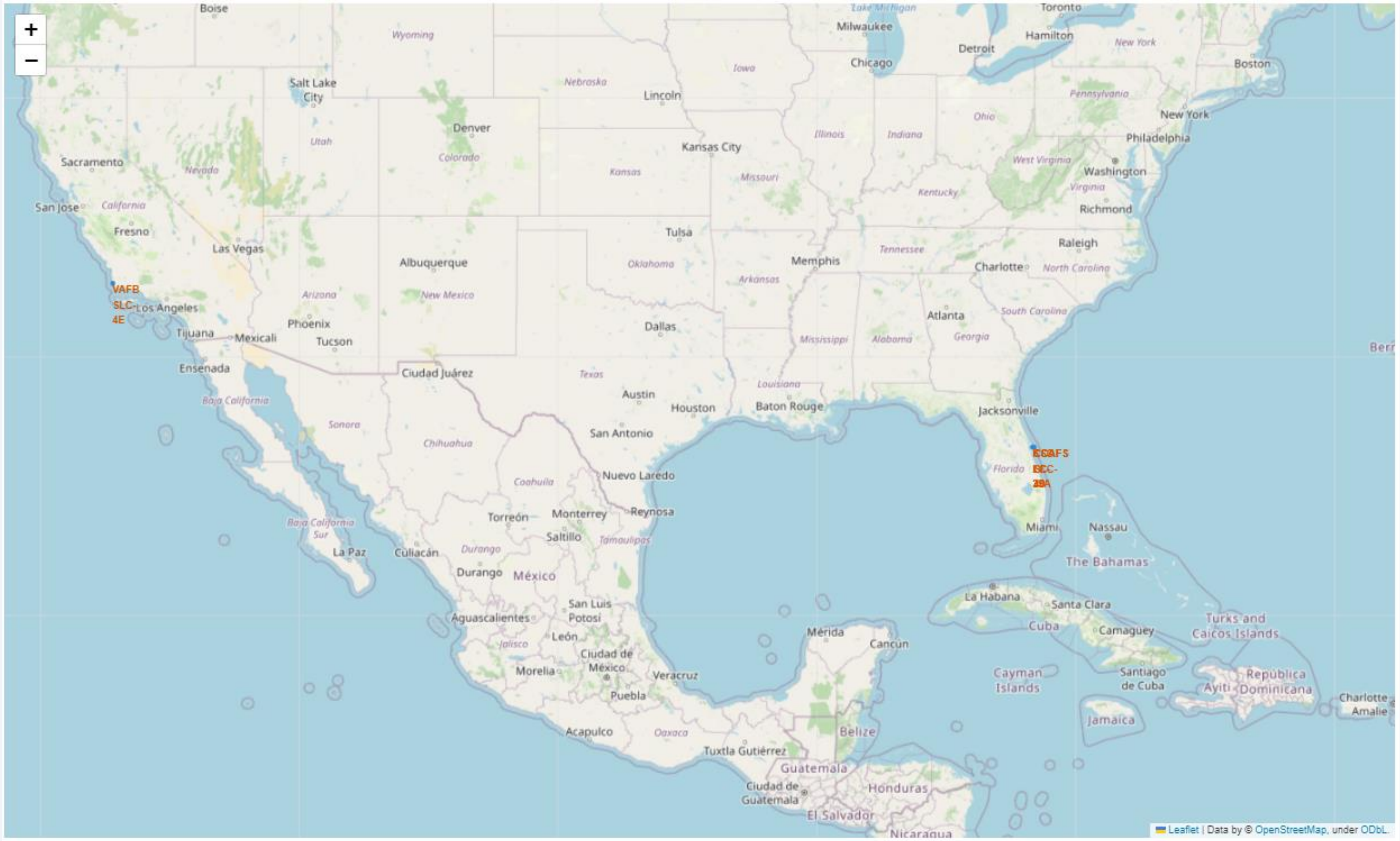
Launch Sites Proximities Analysis

<Folium Map Screenshot 1>

- Replace <Folium map screenshot 1> title with an appropriate title

- Explore the generated folium map and make a proper screenshot to include all launch sites' location markers on a global map

- Explain the important elements and findings on the screenshot
  - All launch sites are located to the east and west of U.S., and close to the coastline.



<Folium Map Screenshot 2>

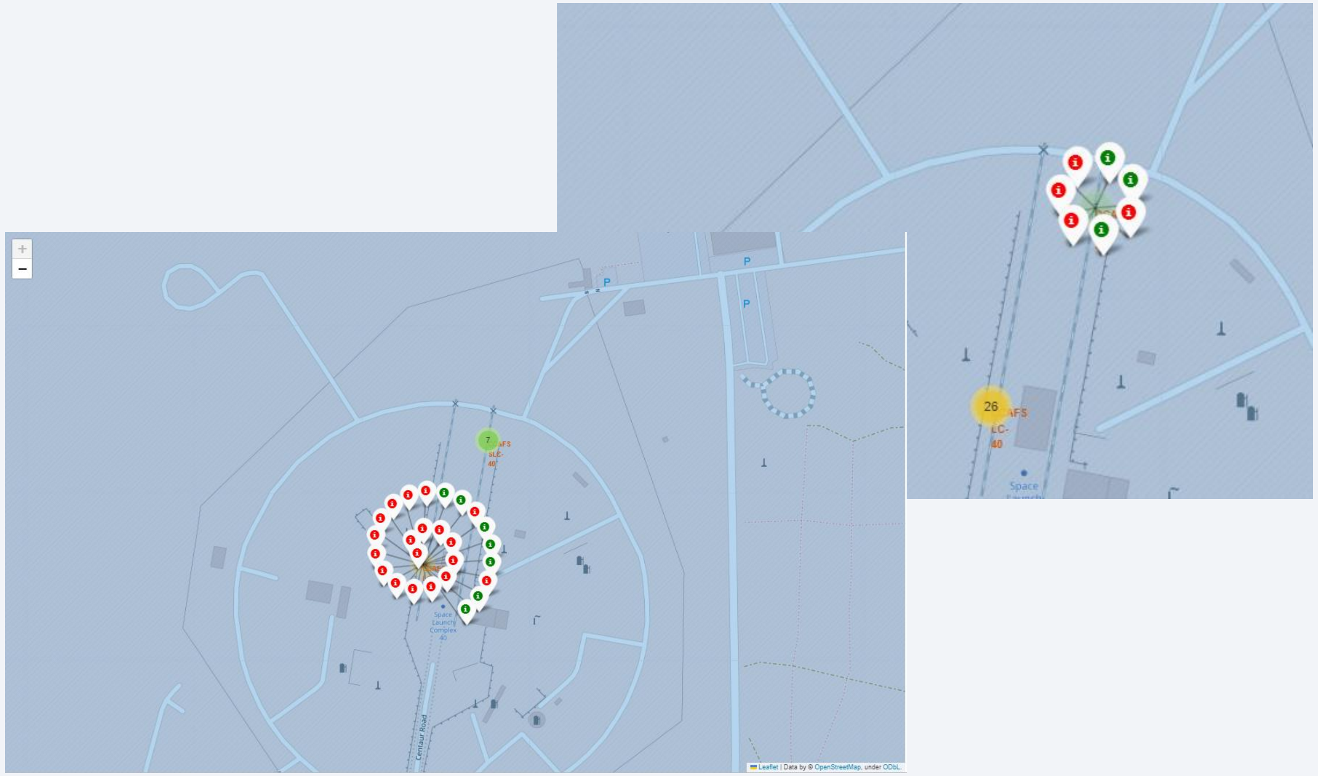
- Replace <Folium map screenshot 2> title with an appropriate title

- Explore the folium map and make a proper screenshot to show the color-labeled launch outcomes on the map

- Explain the important elements and findings on the screenshot
    - Cluster the tasks in the same launch site and display them on the map.

CCAFS SLC-40

Success Rate:42.8%



CCAFS LC-40

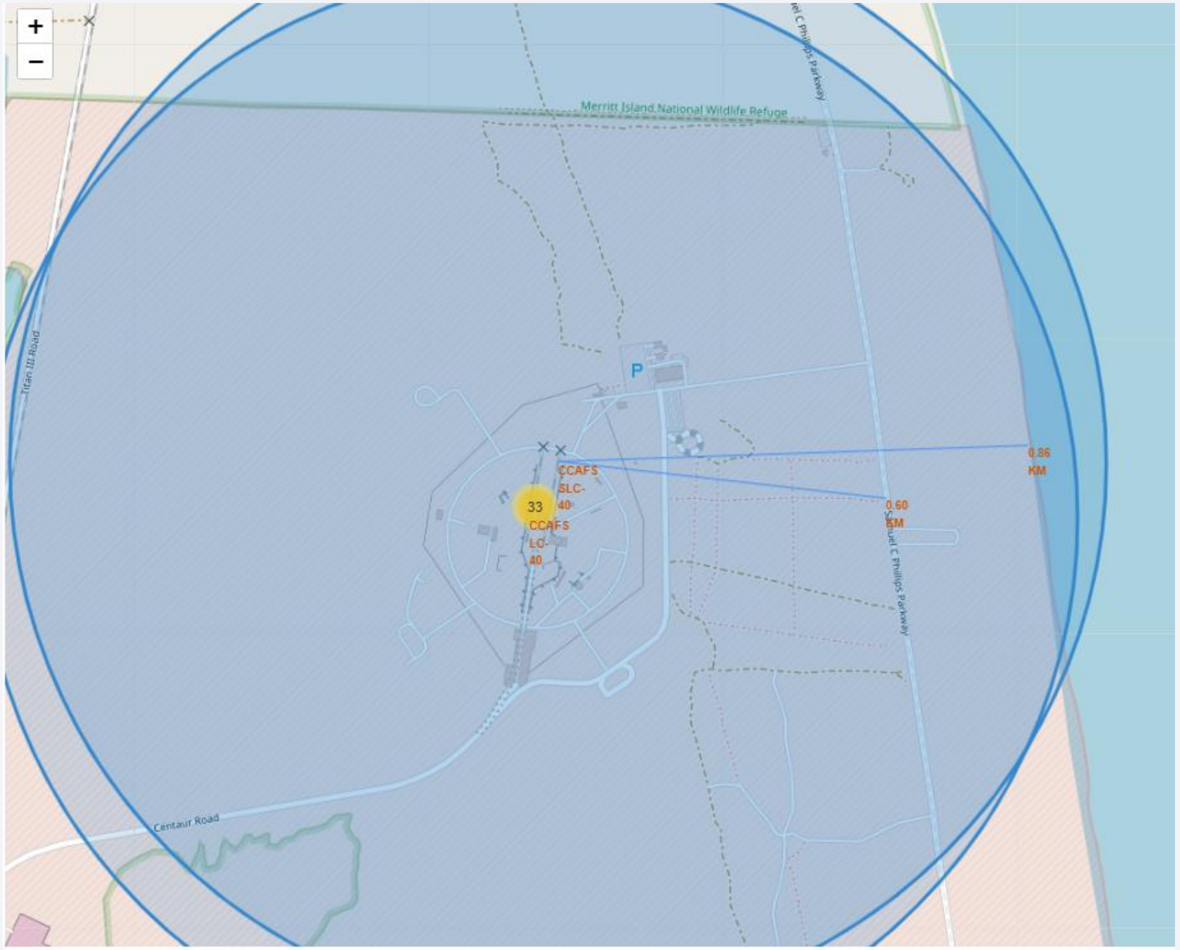
Success Rate:26.9%

<Folium Map Screenshot 3>

- Replace <Folium map screenshot 3> title with an appropriate title

- Explore the generated folium map and show the screenshot of a selected launch site to its proximities such as railway, highway, coastline, with distance calculated and displayed

- Explain the important elements and findings on the screenshot
    - The distance between CCAFS SLC-40 and coastline is 0.86 km.
    - The distance between CCAFS SLC-40 and railway is 0.6 km.



Section 4

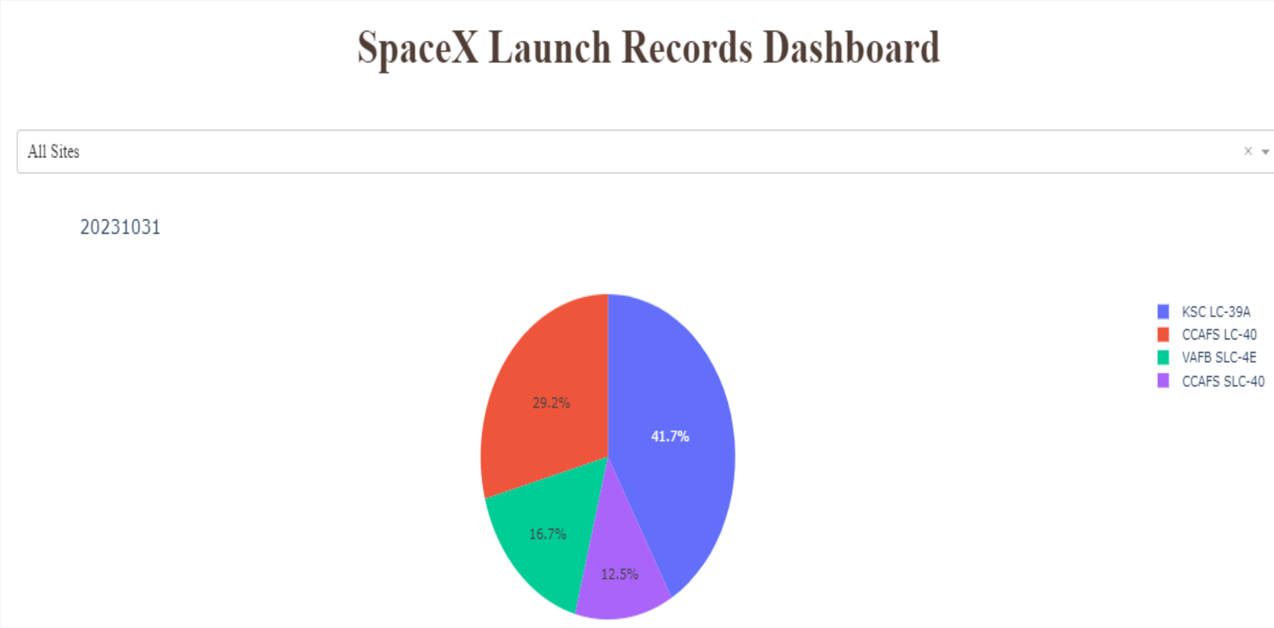
Build a Dashboard with Plotly Dash

<Dashboard Screenshot 1>

- Replace <Dashboard screenshot 1> title with an appropriate title

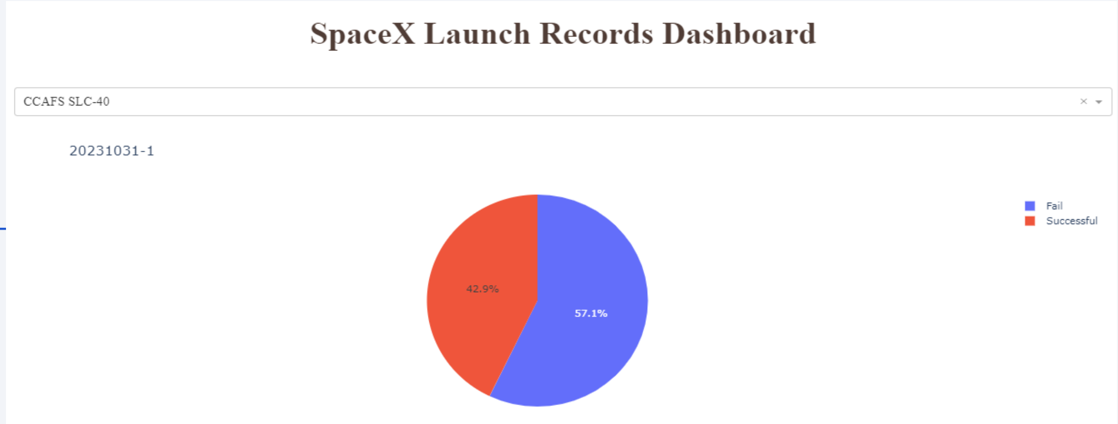
- Show the screenshot of launch Success count for all sites, in a pie chart

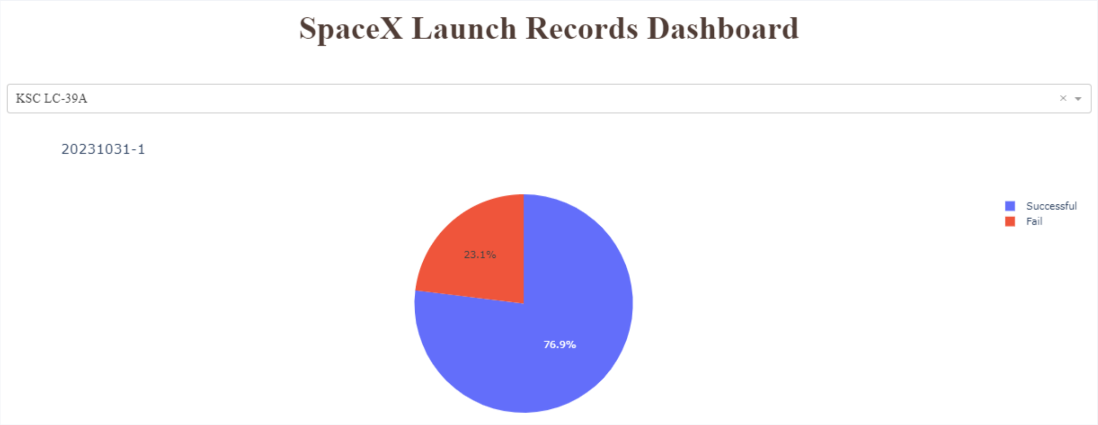
- Explain the important elements and findings on the screenshot
  - The boosters launched in KSC LC-39A had higher success times than other launch site.

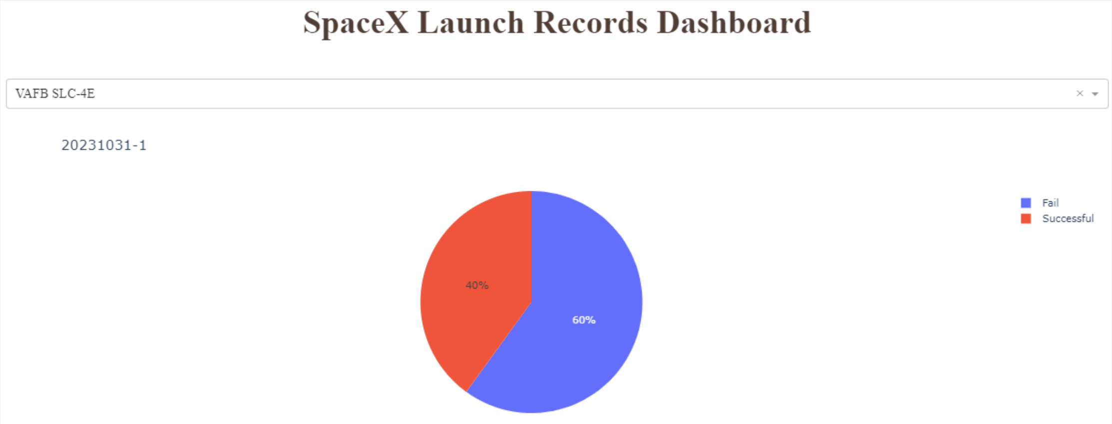


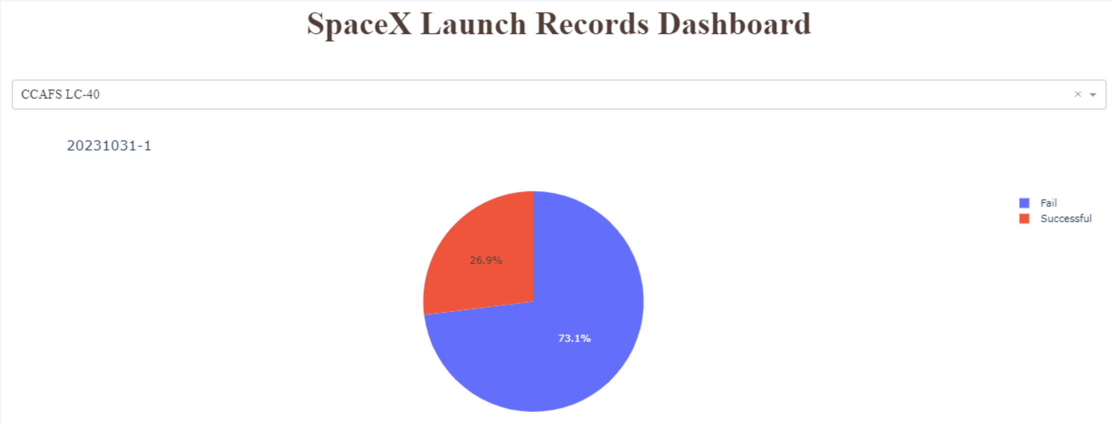
<Dashboard Screenshot 2>

- Replace <Dashboard screenshot 2> title with an appropriate title
- Show the screenshot of the pie chart for the launch site with highest launch Success ratio
- Explain the important elements and findings on the screenshot
  - The boosters launched in KSC LC-39A had higher Success rate than other launch site.



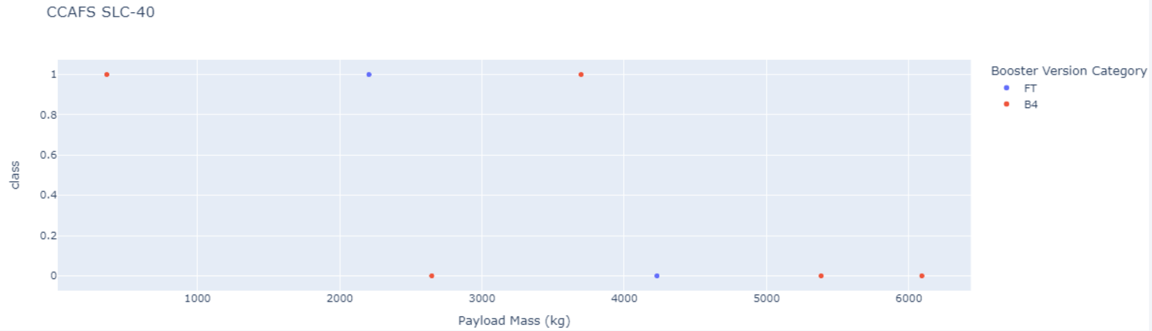


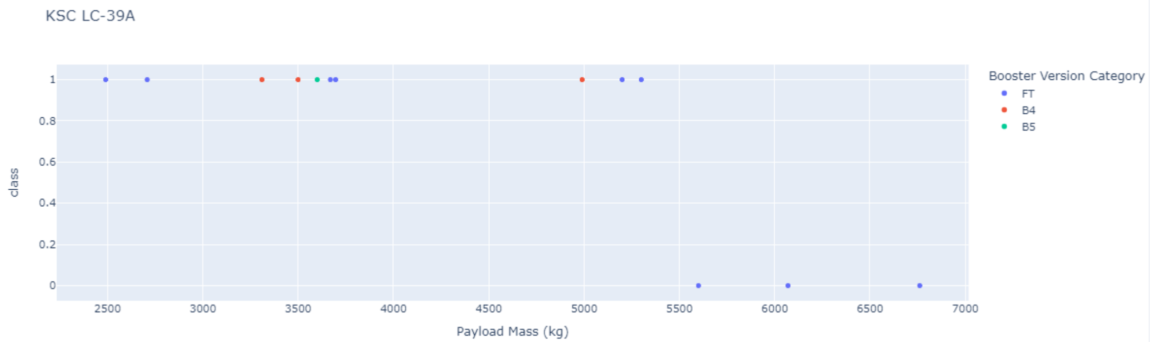


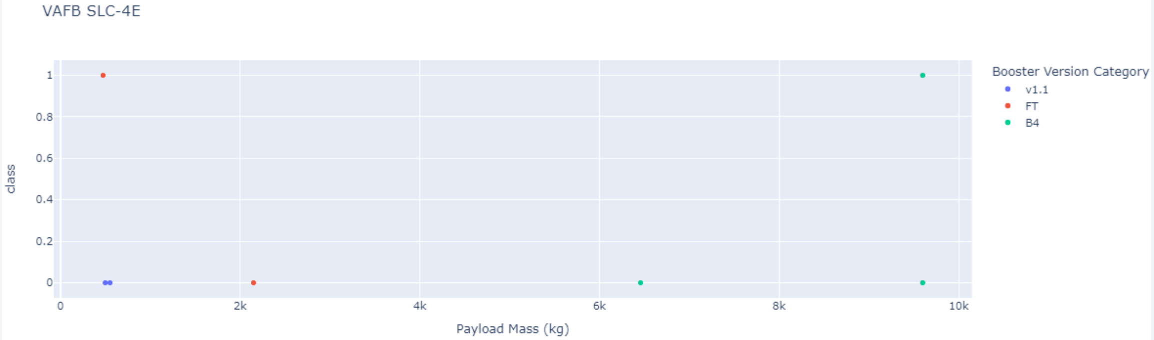


<Dashboard Screenshot 3>

- 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.
    - Apparently, boosters (version:FT/B4/B5) launched at KSC LC-39A were all Success when the payload under 5500 kg. In addition, they had higher success than other launch site.
    - It seems SpaceX team isn’t used to execute missions at CCAFS SLC-40 and VAFB SLC-4E.







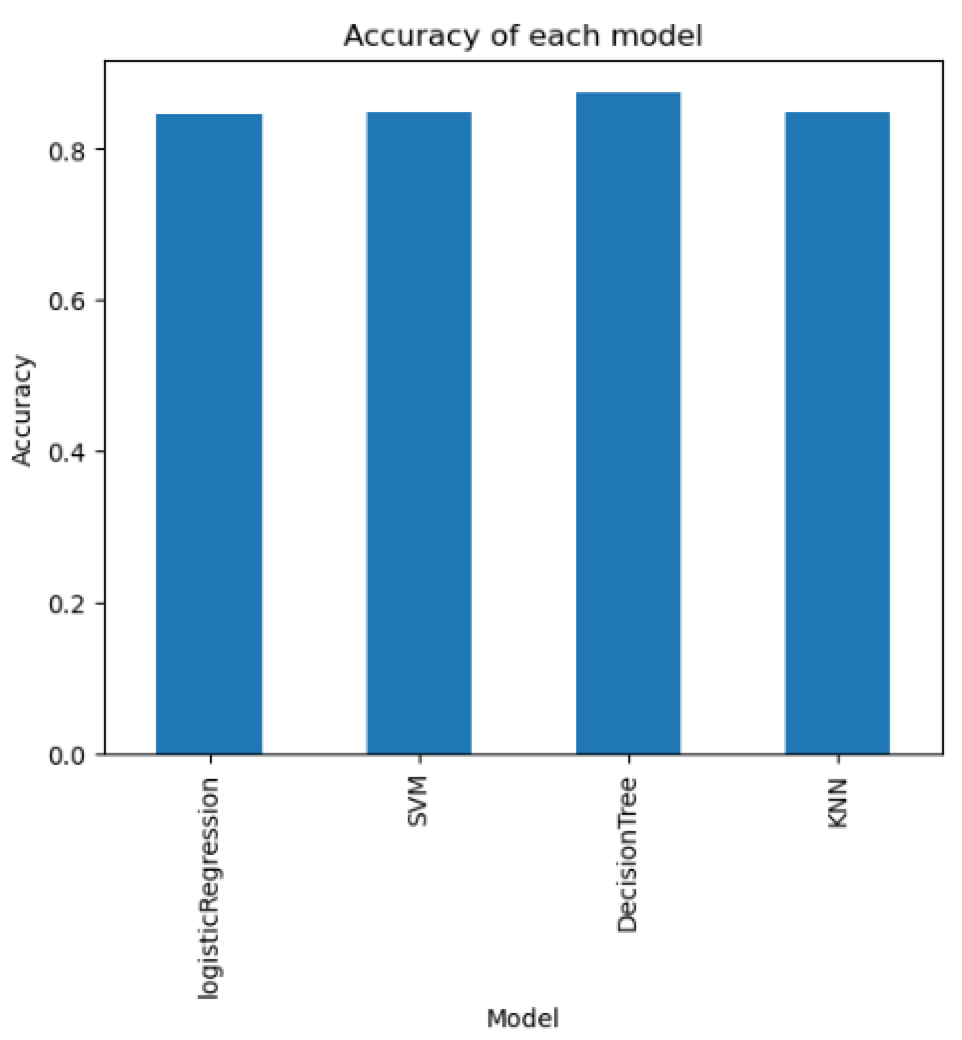
Section 5

Predictive Analysis (Classification)

Classification Accuracy

- Visualize the built model accuracy for all built classification models, in a bar chart

- Find which model has the highest classification accuracy
  - Decision Tree has the highest classification accuracy(88.9%).



Confusion Matrix

- Show the confusion matrix of the best performing model with an explanation
  - In confusion matrix, it has higher recall(83%) and precision(83%) value than other model such as Logistic Regression(Recall:50%/Precision:0%), SVM(Recall:50%/Precision:0%) and KNN(Recall:50%/Precision:0%).

TASK 9

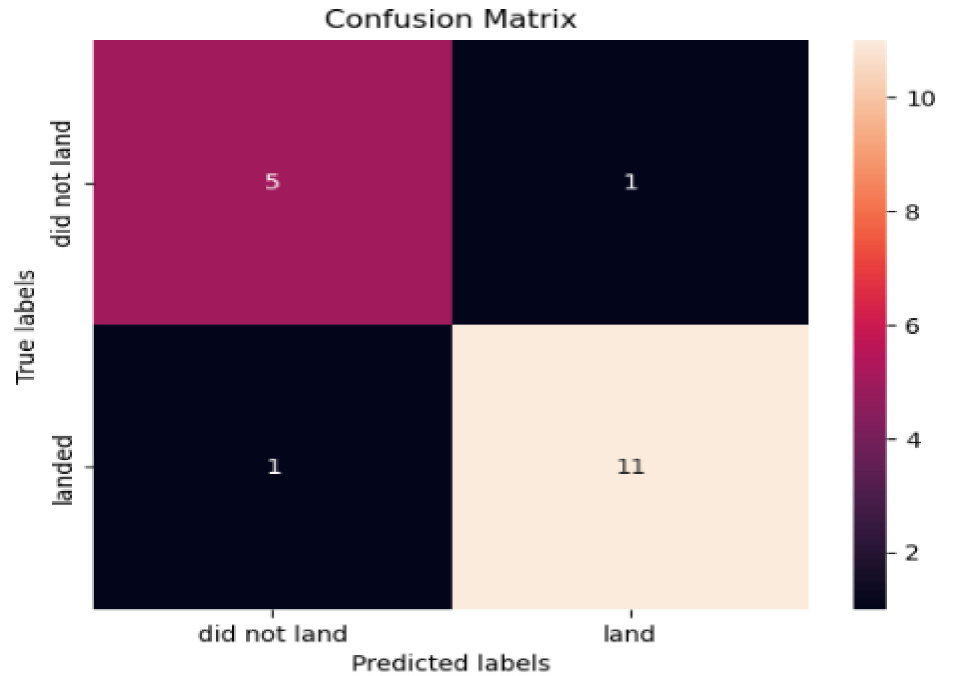
```
Calculate the accuracy of tree_cv on the test data using the method <code>score</code>:
```

```
tree_cv.score(X_test,Y_test)
```

```
0.8888888888888888
```

We can plot the confusion matrix

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



Conclusions

- The Success rate since 2013 kept increasing till 2020
- The boosters (version:B5) launch at KSD LC-39A and landing on the ground has higher success rate than the other.
  - SpaceX team is used to execute heavy payload missions at CCAFS SLC 40 and KSC LC-39A and it is more successful there than VAFB SLC-4E launch site.
  - The booster version F9 B5 FT executed heavy payload mission many times.
  - The rate of landing at the drone ship is 50% between 2010 to 2017. However, the rate of landing at the ground is 100%.
  - The boosters launched in KSC LC-39A had higher success times than other launch site and also had higher success rate than other launch sites. (Payload under 5500 kg are all success in chart of dashboard.)
- We can predict the success rate by decision tree model.

Appendix

- Include any relevant assets like Python code snippets, SQL queries, charts, Notebook outputs, or data sets that you may have created during this project

Thank you!