IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Santiago Wirth
14/01/2024

Outline

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

Executive Summary

- Methodologies:
  - Data Collection
  - Data Pre-processing
  - Data Visualization
  - Machine learning
- Summary of all results
  - Success rate of landing related to orbit type, pay load mass and launch site
  - Different orbit type related to different range of flight number
  - Different launch sites handle different payload mass

Introduction

- Project background and context
  - The commercial space age is here, companies are making space travel affordable for everyone. SpaceX advertises Falcon 9 rocket launches on its website with a cost of 62 million dollars; other providers cost upwards of 165 million dollars each, much of the savings is 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.
- Problems you want to find answers
  - What are the crucial features related to the success of first stage landing of Falcon 9 rocket from SpaceX?

Section 1

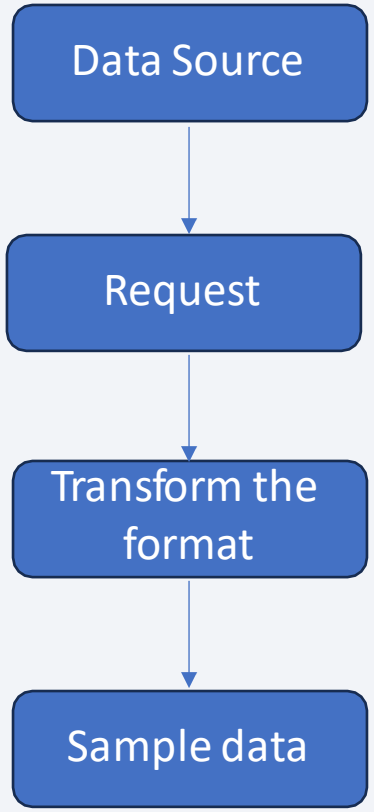
Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Describe how data was collected
- Perform data wrangling
  - Describe how data was processed
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - How to build, tune, evaluate classification models

Data Collection



Identify the Data Source (e.g. URL, database, files, etc.), in our case , we get the URLs.

Send requests to the webpage to get the raw information.

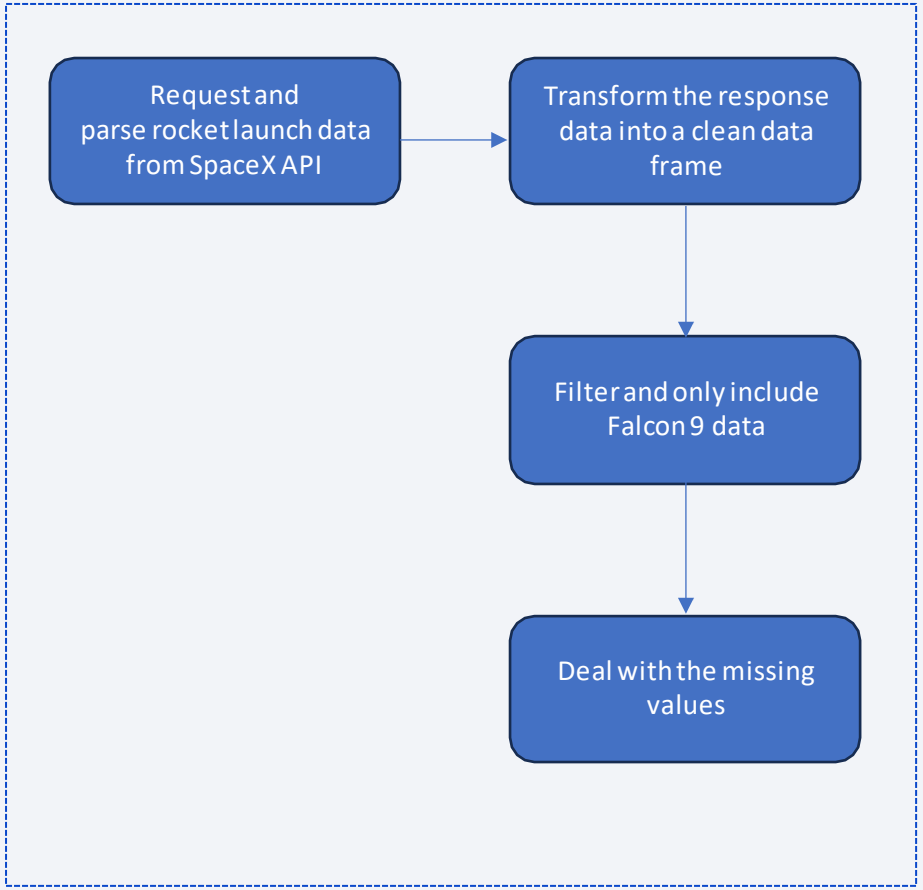
Transform the data into the format which is easier to handle, such as panda.DataFrame.

Filter or sample the data to retain what we need, for example we just focus on some tables or some booster version.

Data Collection - SpaceX API

- Send a request to SpaceX API, get the response data and transform the related contents into a data frame, and the do further filtering and pre-processing to get the data we would focus on

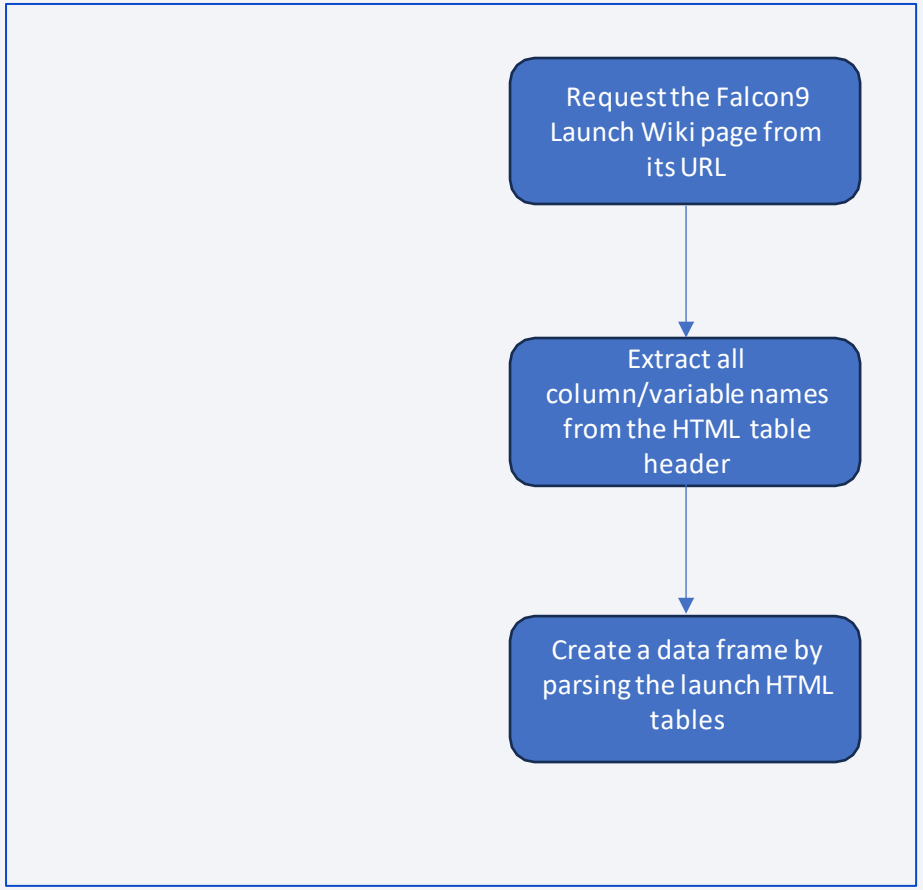
- https://github.com/JieyaoMinn/IBMds/blob/main/jupyter-labs-spacex-data-collection-api.ipynb



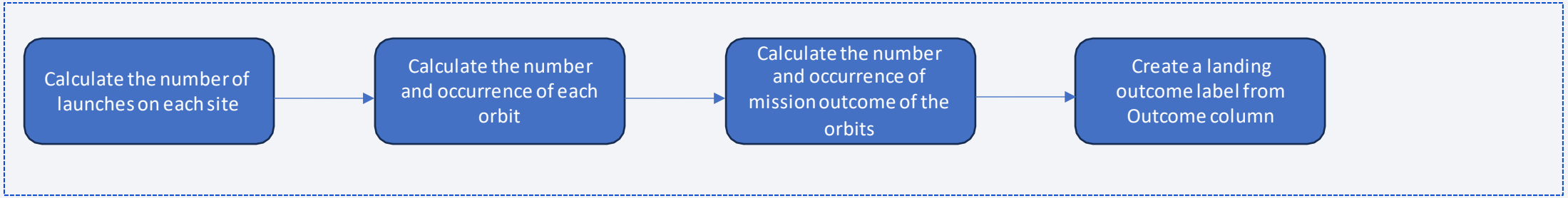
Data Collection - Scraping

- Request the Falcon9 Lauch Wiki page from the URL, extract related contents from the html data, and then transform them into a data frame

- https://github.com/JieyaoMinn/IBMds/blob/main/jupyter-labs-webscraping.ipynb



Data Wrangling



- Collect some statistics information and insights from the data
- Prepare the data for further processing and analysis

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

EDA with Data Visualization

- Scatter plot: clear visualization of the individual datapoint distribution
- Line plot: clear visualization of relationship between two variables or the trend
- Bar chart: clear visualization of the distribution on categorical factors

- https://github.com/JieyaoMinn/IBMds/blob/main/jupyter-labs-eda-dataviz.ipynb

EDA with SQL

- select * from SPACEXTABLE
- select * from SPACEXTABLE where Launch_Site like "CCA%" limit 5
- select sum(PAYLOAD_MASS__KG_) from SPACEXTABLE where Customer = "NASA (CRS)"
- select avg(PAYLOAD_MASS__KG_) from SPACEXTABLE where Booster_Version like "F9 v1.1%"
- select min(Date) from SPACEXTABLE where Landing_Outcome = "Success (ground pad)"
- select Booster_Version from SPACEXTABLE where Landing_Outcome = "Success (drone ship)" and (PAYLOAD_MASS__KG_ between 4000 and 6000)
- select distinct Mission_Outcome from SPACEXTABLE
- select count(Date) Mission_Outcome from SPACEXTABLE where Mission_Outcome like "Success%"
- select count(Date) Mission_Outcome from SPACEXTABLE where Mission_Outcome like "Failure%"
- select Booster_Version from SPACEXTABLE where PAYLOAD_MASS__KG_ = (select max(PAYLOAD_MASS__KG_) from SPACEXTABLE)
- select substr(Date, 6,2) as Month, Landing_Outcome, Booster_Version, Launch_Site from SPACEXTABLE where Landing_Outcome like "Failure%"
- select Landing_Outcome, count(Landing_Outcome) from SPACEXTABLE where Date between '2010-06-04' and '2017-03-20' group by Landing_Outcome ORDER BY count(Landing_Outcome) DESC
- https://github.com/JieyaoMinn/IBMds/blob/main/jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

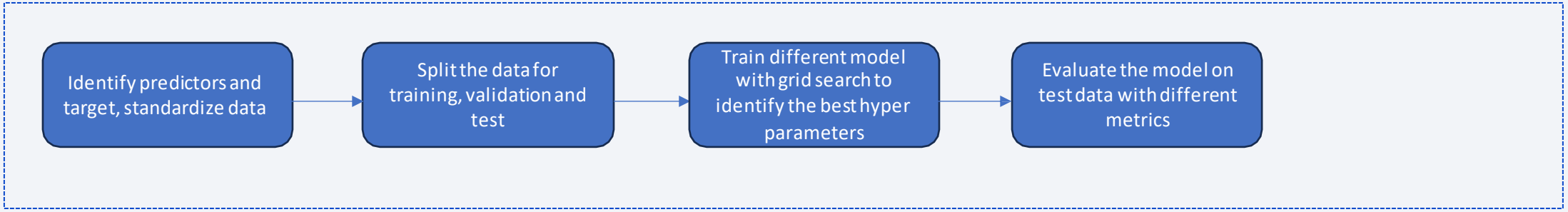
- Marker: mark the launch site on the map
- Circle: a circle around the marker indicating the impacting area
- Marker cluster: identify the makers which may have exactly same coordinate
- Line: measure the distance between markers

- https://github.com/JieyaoMinn/IBMds/blob/main/lab_jupyter_launch_site_location.ipynb

Build a Dashboard with Plotly Dash

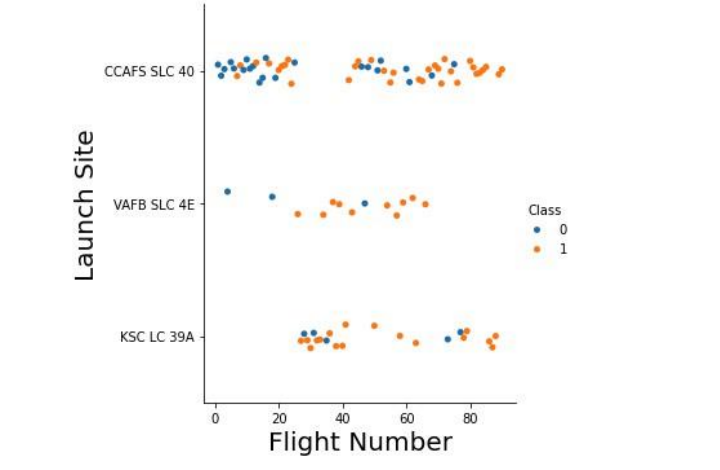
- Summarize what plots/graphs and interactions you have added to a dashboard
- Explain why you added those plots and interactions
- Add the GitHub URL of your completed Plotly Dash lab, as an external reference and peer-review purpose

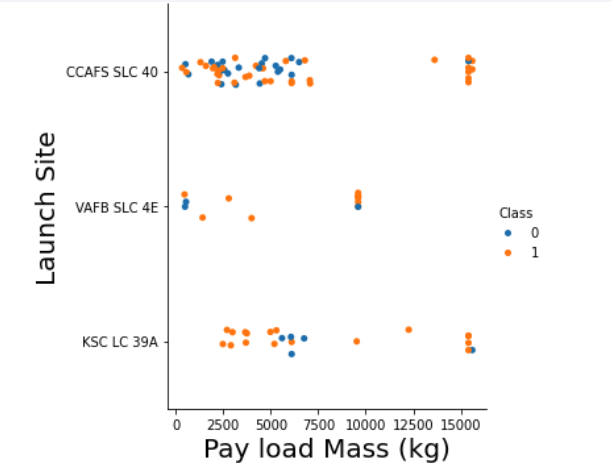
Predictive Analysis (Classification)

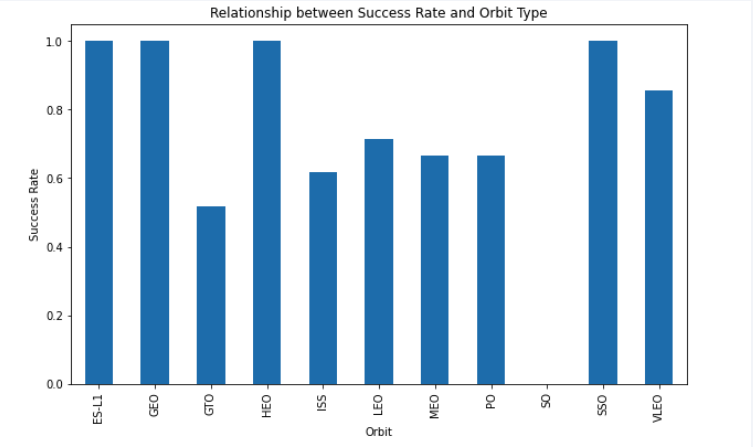


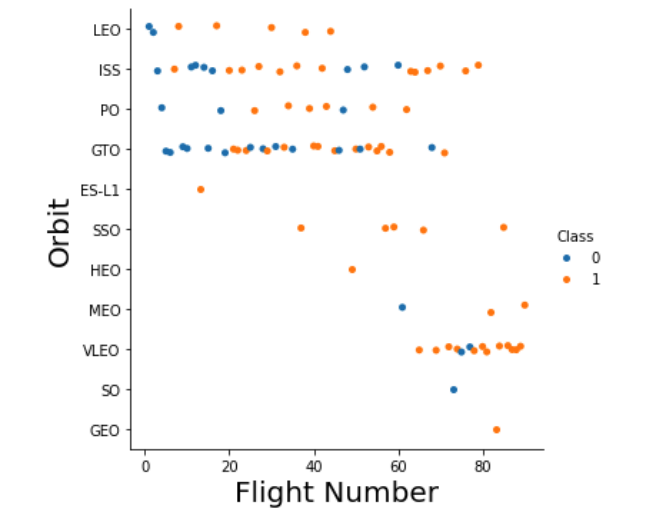
- https://github.com/JieyaoMinn/IBMds/blob/main/SpaceX_Machine_Learning_Prediction_Part_5.jupyterlite.ipynb

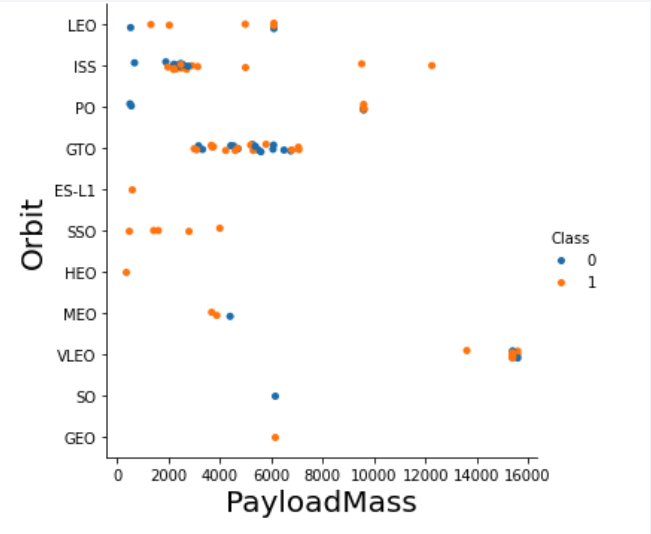
Results

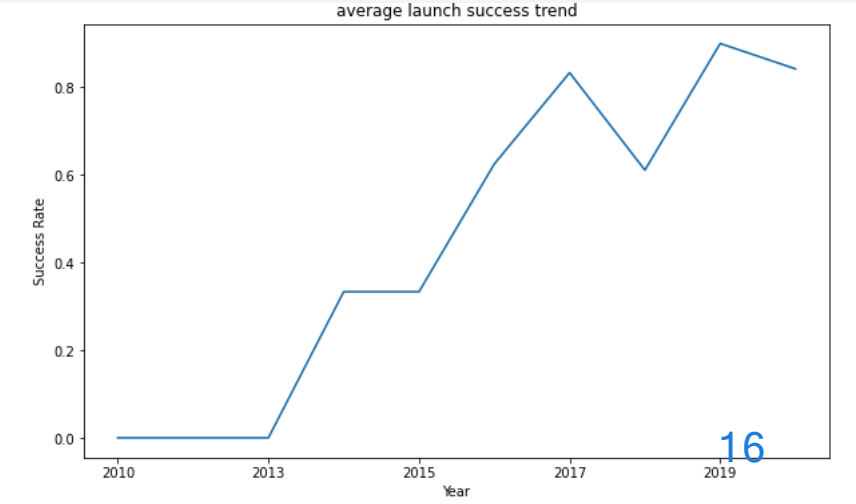

CCAFS SLC 40
VAFB SLC 4E
KSC LC 39A
Launch Site
Class
0
1
0
20
40
60
80
Flight Number


CCAFS SLC 40
VAFB SLC 4E
KSC LC 39A
Launch Site
Class
0
1
0
2500
5000
7500
10000
12500
15000
Pay load Mass (kg)


Relationship between Success Rate and Orbit Type
Success Rate
1.0
0.8
0.6
0.4
0.2
0.0
ES-L1
GEO
GTO
HEO
ISS
LEO
MEO
PO
SO
SSO
VLEO
Orbit


LEO
ISS
PO
GTO
ES-L1
SSO
HEO
MEO
VLEO
SO
GEO
Orbit
Class
0
1
0
20
40
60
80
Flight Number


LEO
ISS
PO
GTO
ES-L1
SSO
HEO
MEO
VLEO
SO
GEO
Orbit
Class
0
1
0
2000
4000
6000
8000
10000
12000
14000
16000
PayloadMass


average launch success trend
Success Rate
0.8
0.6
0.4
0.2
0.0
2010
2013
2015
2017
2019
Year

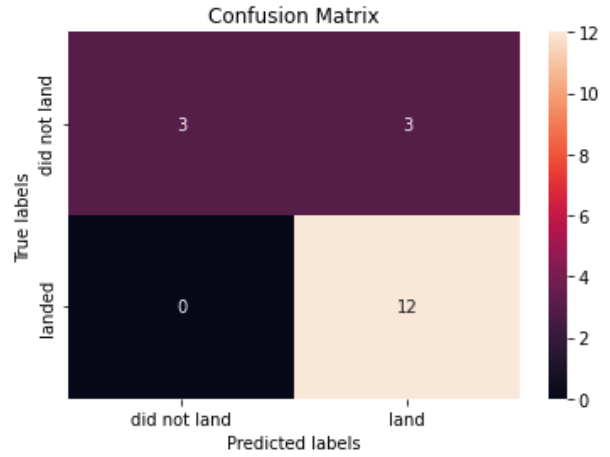
Results

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

```
In [26]: logreg_cv.score(X_test, Y_test)
Out[26]: 0.8333333333333334
```

Lets look at the confusion matrix:

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

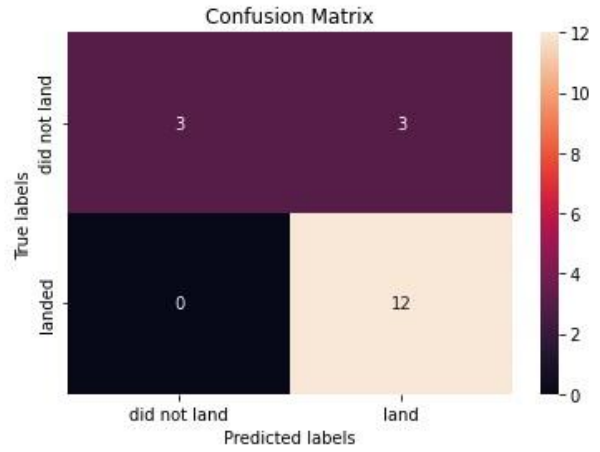


Calculate the accuracy on the test data using the method `score`:

```
In [31]: svm_cv.score(X_test, Y_test)
Out[31]: 0.8333333333333334
```

We can plot the confusion matrix

```
In [32]: yhat=svm_cv.predict(X_test)
         plot_confusion_matrix(Y_test,yhat)
```



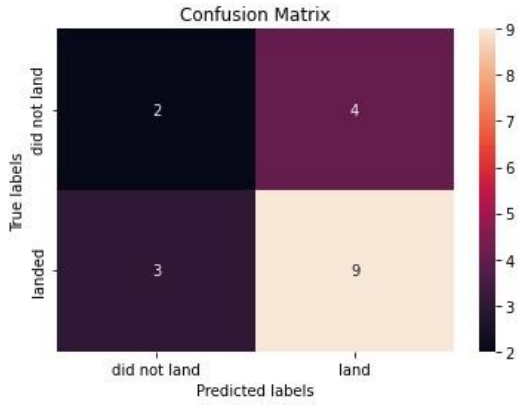
Results

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

```
In [46]: tree_cv.score(X_test, Y_test)
Out[46]: 0.6111111111111112
```

We can plot the confusion matrix

```
In [47]: yhat = tree_cv.predict(X_test)
         plot_confusion_matrix(Y_test,yhat)
```

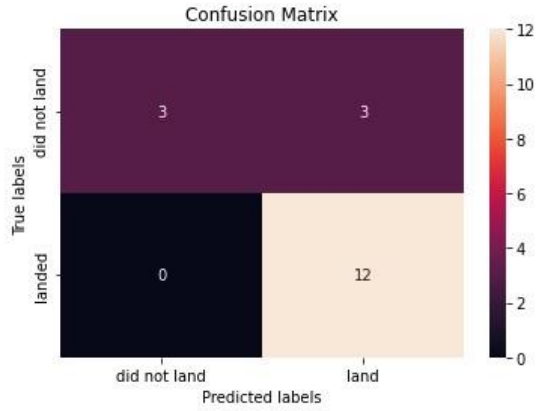


Calculate the accuracy of knn_cv on the test data using the method `score` :

```
In [41]: knn_cv.score(X_test, Y_test)
Out[41]: 0.8333333333333334
```

We can plot the confusion matrix

```
In [42]: yhat = knn_cv.predict(X_test)
         plot_confusion_matrix(Y_test,yhat)
```

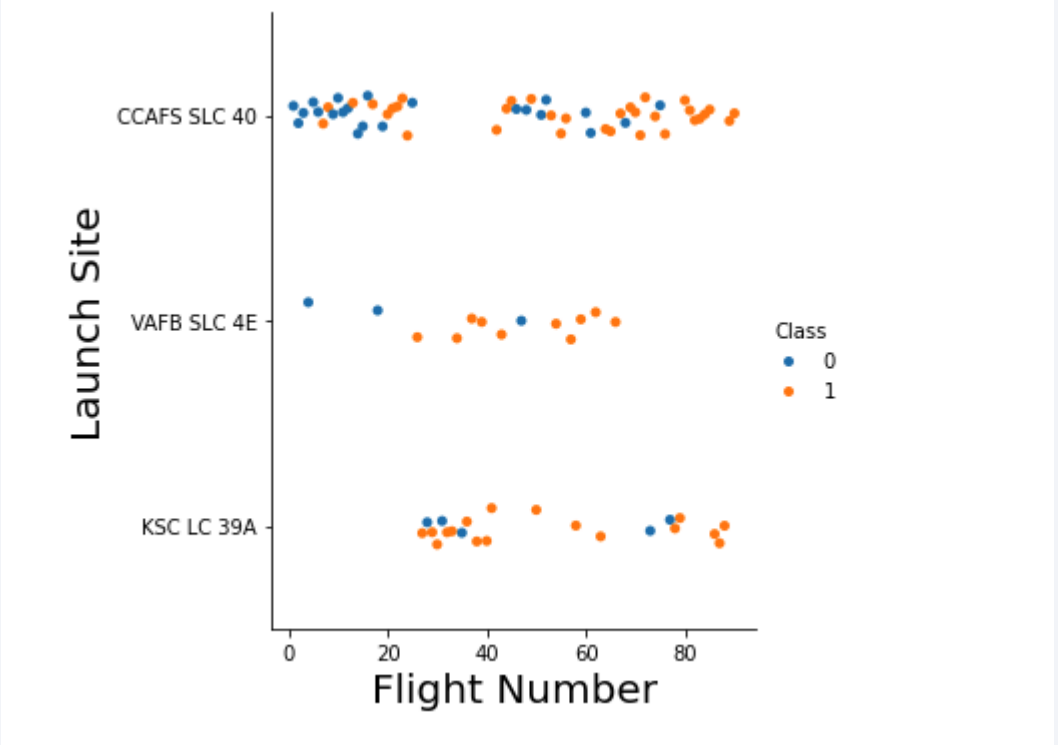


Section 2

Insights drawn from EDA

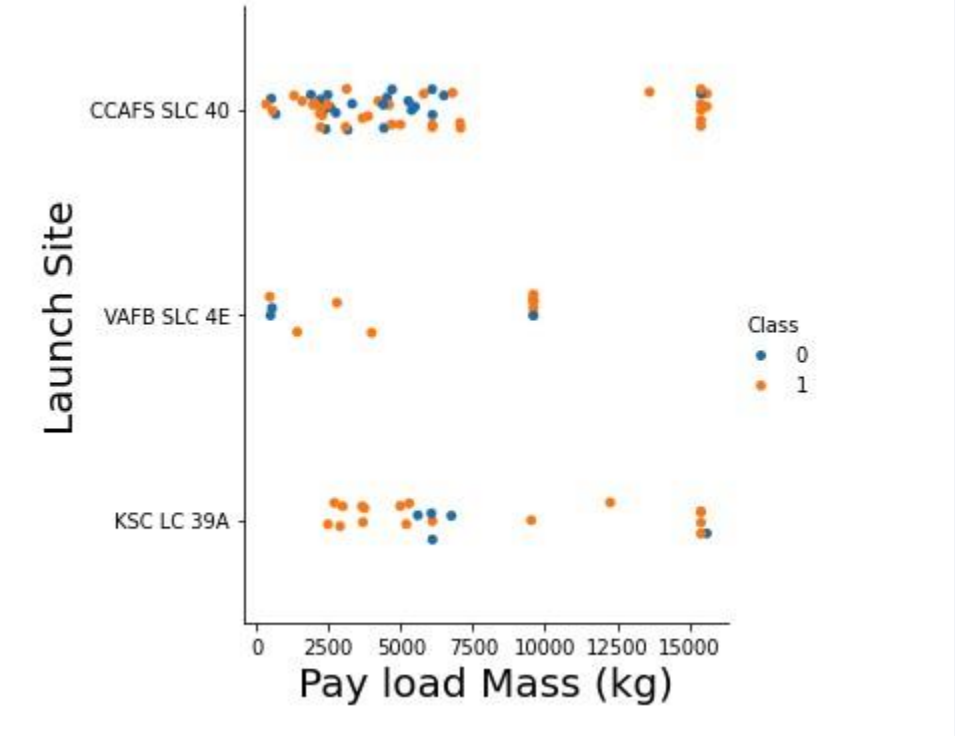
Flight Number vs. Launch Site

- Most launches are at launch site CCAFS SLC 40, especially the relatively small and large flight number
- The flight number between 20 to 40 most launched at KSCLC 39A
- The smaller flight number indicates more failure, the larger flight number indicates more success



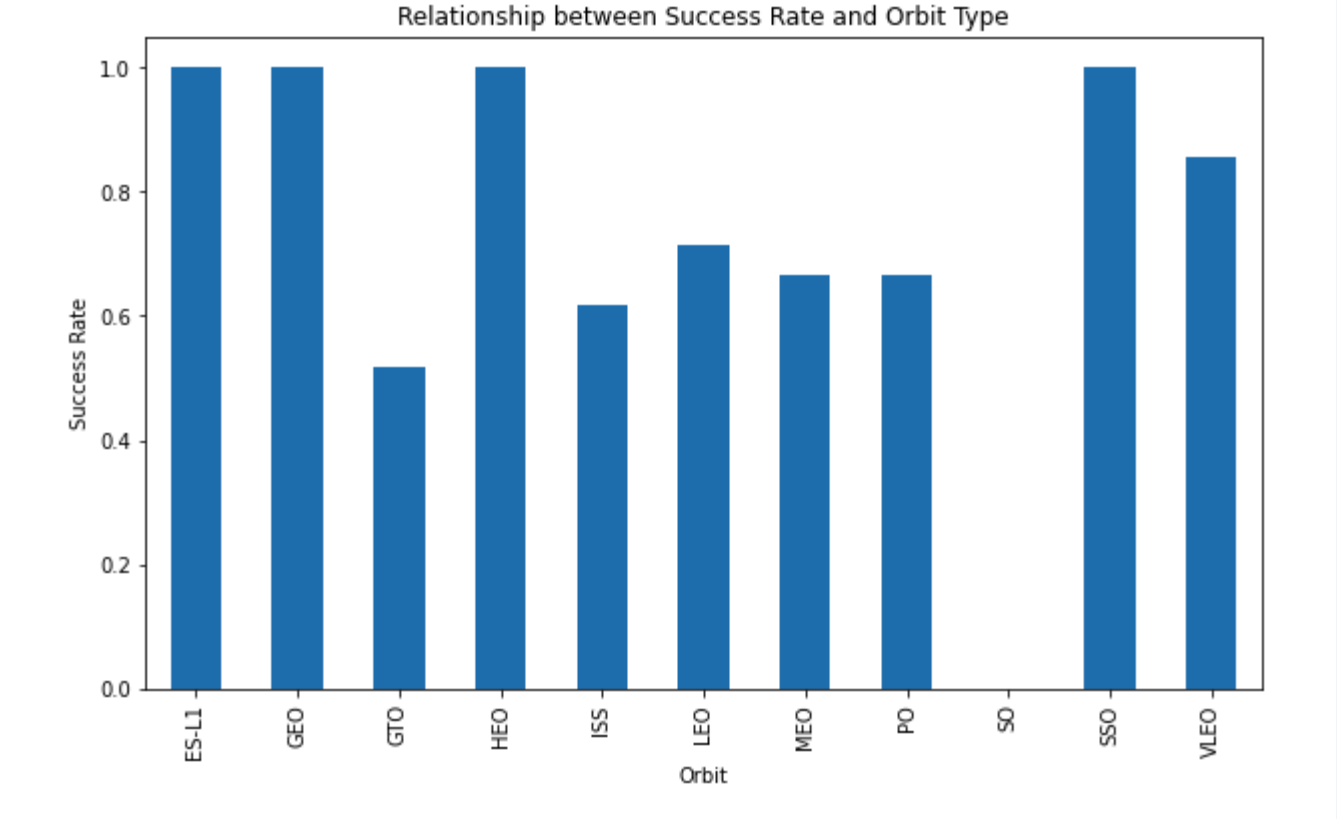
Payload vs. Launch Site

- For the VAFB-SLC launch site there are no rockets launched for heavy payload mass(greater than 10000)
- Heavy payload have more success cases
- For the CCAFS SLC 40 launch site there are no median payload
- Most payload mass is 10000 launched at VAFB SLC launch site



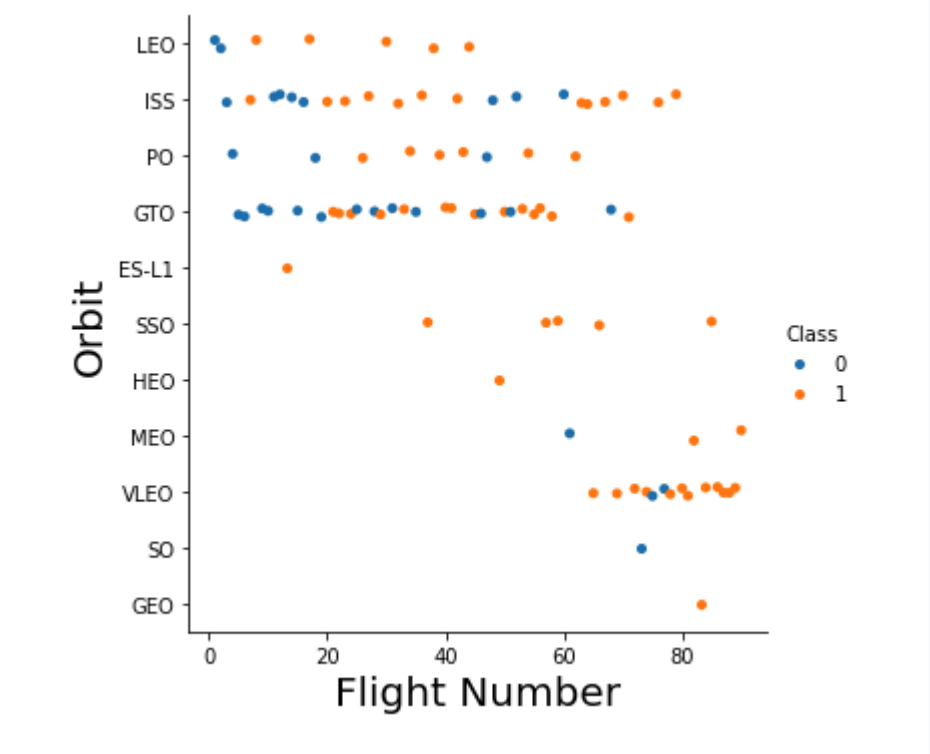
Success Rate vs. Orbit Type

- For orbit type ES-L1, GEO, HEO, SSO, there are all success
- For orbit type GTO the success rate is 0.5, which is the lowest
- For orbit type SO, there is no data



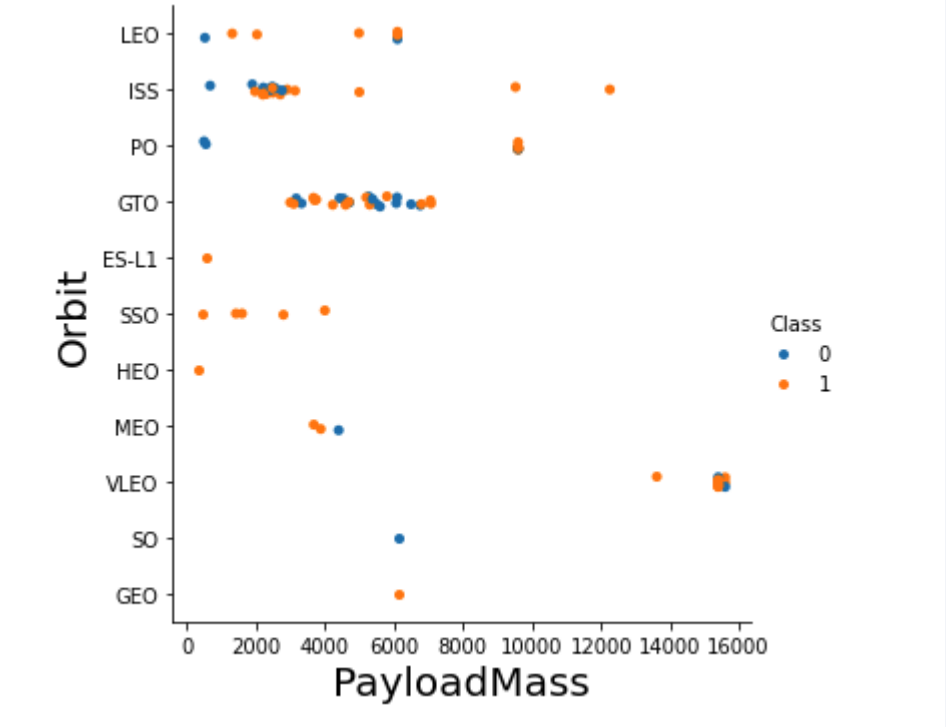
Flight Number vs. Orbit Type

- Orbit type VLEO related to relatively large flight number
- There are no explicit relationship between orbit type GTO and the success of landing
- Flight number less than 60 mostly related to orbit type LEO, ISS, PO, GTO



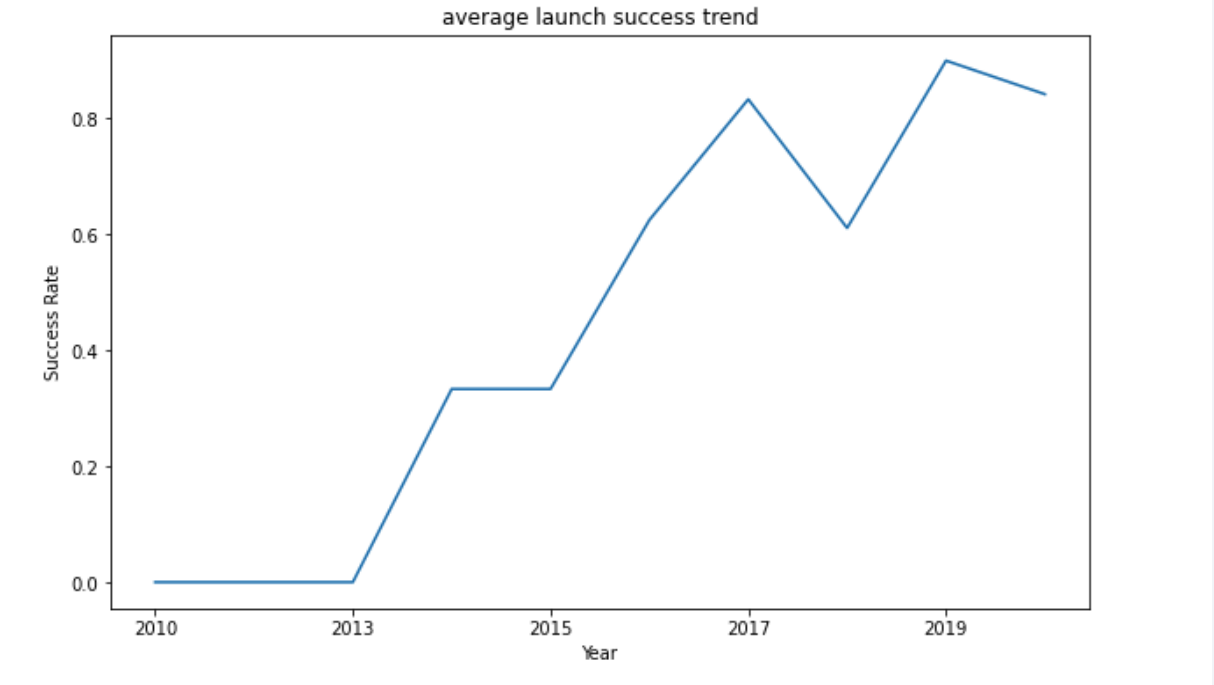
Payload vs. Orbit Type

- Heavy payload mass related to orbit type VLEO
- Small payload mass (0) related to orbit type LEO, ISS, PO are failed to land, related to ES-L1, SSO, HEO are successful to land
- Payload mass related to GTO are between 3000-8000
- Small payload mass related to SSO



Launch Success Yearly Trend

- The overall trend is that the success rate increasing
- There is a little bit success rate drop between 2017-2018, and 2019-2020



All Launch Site Names

- The unique launch sites are:
  - CCAFS LC-40
  - VAFB SLC-4E
  - KSC LC-39A
  - CCAFS SLC-40

```
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

Launch Site Names Begin with 'CCA'

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

```
In [12]: %sql select * from SPACEXTABLE where Launch_Site like "CCA%" limit 5
 * sqlite:///my_data1.db
Done.
```

Out[12]:

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
2010-12-08	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	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
2012-10-08	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
2013-03-01	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

- Here presenting first five record the launch site begin with string "CCA" from the table SPACEXTABLE

Total Payload Mass

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

```
In [13]: %sql select sum(PAYLOAD_MASS__KG_) from SPACEXTABLE where Customer = "NASA (CRS)"
 * sqlite:///my_data1.db
Done.
```

Out[13]:

sum(PAYLOAD_MASS__KG_)
45596

- The total mass carried by boosters launched by NASA (CRS) is 45596kg

Average Payload Mass by F9 v1.1

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

```
In [14]: %sql select avg(PAYLOAD_MASS__KG_) from SPACEXTABLE where Booster_Version like "F9 v1.1%"
 * sqlite:///my_data1.db
Done.
```

Out[14]:

avg(PAYLOAD_MASS__KG_)
2534.6666666666665

- The average payload mass carried by booster version F9 v1.1 is about 2534.67kg

First Successful Ground Landing Date

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

Hint:Use min function

```
In [16]: %sql select min(Date) from SPACEXTABLE where Landing_Outcome = "Success (ground pad)"
```

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

Out[16]:

min(Date)
2015-12-22

- The dates of the first successful landing outcome on ground pad is 2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

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

```
In [18]: %sql select Booster_Version from SPACEXTABLE where Landing_Outcome = "Success (drone ship)" and (PAYLOAD_MASS__KG_ b
```

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

Out[18]:

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

- The names of boosters which have successfully landed on drone ship and had payload mass greater than 4000 but less than 6000 are:
  - F9 FT B1022
  - F9 FT B1026
  - F9 FT B1021.2
  - F9 FT B1031.2

Total Number of Successful and Failure Mission Outcomes

- There are 1 failure and 100 success of mission

List the total number of successful and failure mission outcomes

```
In [20]: %sql select distinct Mission_Outcome from SPACEXTABLE
```

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

Out[20]:

Mission_Outcome
Success
Failure (in flight)
Success (payload status unclear)
Success

```
In [21]: %sql select count(Date) Mission_Outcome from SPACEXTABLE where Mission_Outcome like "Success%"
```

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

Out[21]:

Mission_Outcome
100

```
In [22]: %sql select count(Date) Mission_Outcome from SPACEXTABLE where Mission_Outcome like "Failure%"
```

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

Out[22]:

Mission_Outcome
1

2015 Launch Records

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 [21]:
```
%sql select substr(Date, 6,2) as Month, Landing_Outcome, Booster_Version, Launch_Site from SPACEXTABLE where substr(
```

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

Out[21]:

Month	Landing_Outcome	Booster_Version	Launch_Site
01	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
04	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

- F9 v1.1 B1012 launched at CCAFS LC-40 failed to land in drone ship in Jan 2015
- F9 v1.1 B1015 launched at CCAFS LC-40 failed to land in drone ship in Apr 2015

Boosters Carried Maximum Payload

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

In [20]:
```
%sql select Booster_Version from SPACEXTABLE where PAYLOAD_MASS__KG_ = (select max(PAYLOAD_MASS__KG_) from SPACEXTAB
```

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

Out[20]:

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

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.

In [22]:
```
%sql select Landing_Outcome, count(Landing_Outcome) from SPACEXTABLE where Date between '2010-06-04' and '2017-03-20
```

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

Out[22]:

Landing_Outcome	count(Landing_Outcome)
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

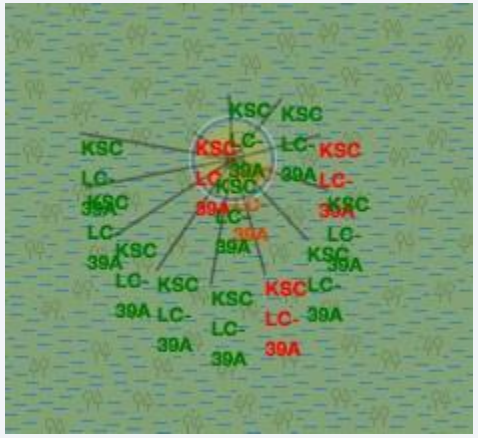
Section 3

Launch Sites Proximities Analysis

Folium Map with launch site makers

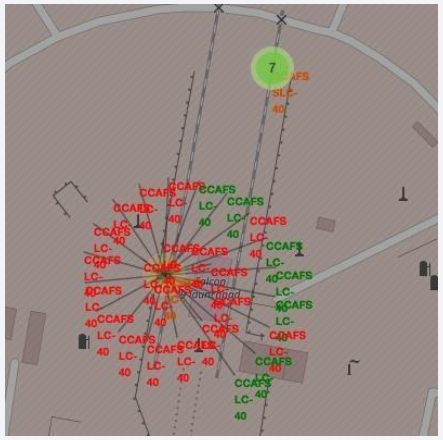
- KSC LC-39A, CCAFS SLC-40, CCAFS LC-40 are in Florida, VAFB SLC-4E is in Califonia
- CCAFS SLC-40 and CCAFS LC-40 are highly overlap
- These four launch sites are near the sea

Folium Map with launch outcomes at each site



- The 13 launches outcome at KSC LC-39A

The 7 launches outcome at CCAFS SLC-40



The 26 launches outcome at CCAFS LC-40

The 10 launches outcome at VAFB SLC-4E

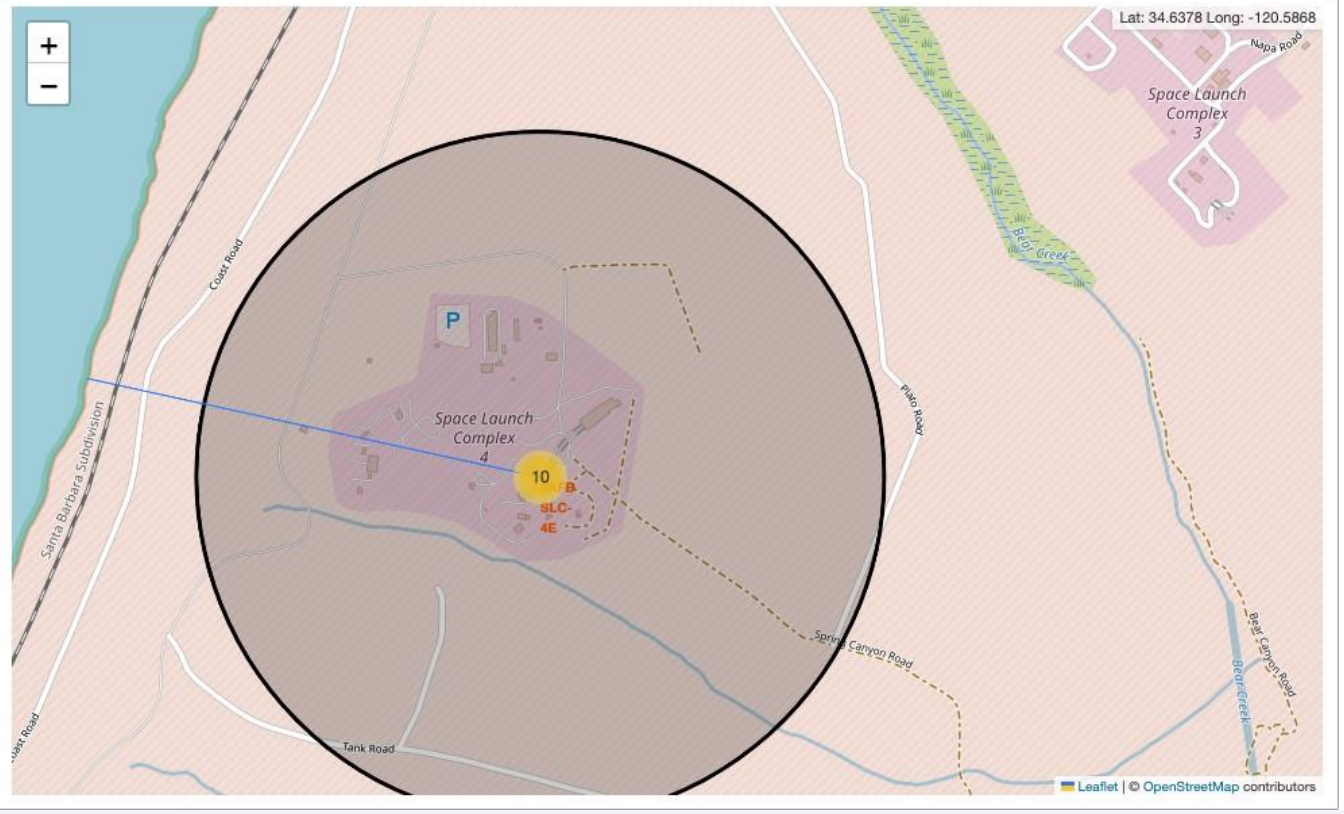
- Color green related to success, red related to fail
- High success rate at KSC LC-39A

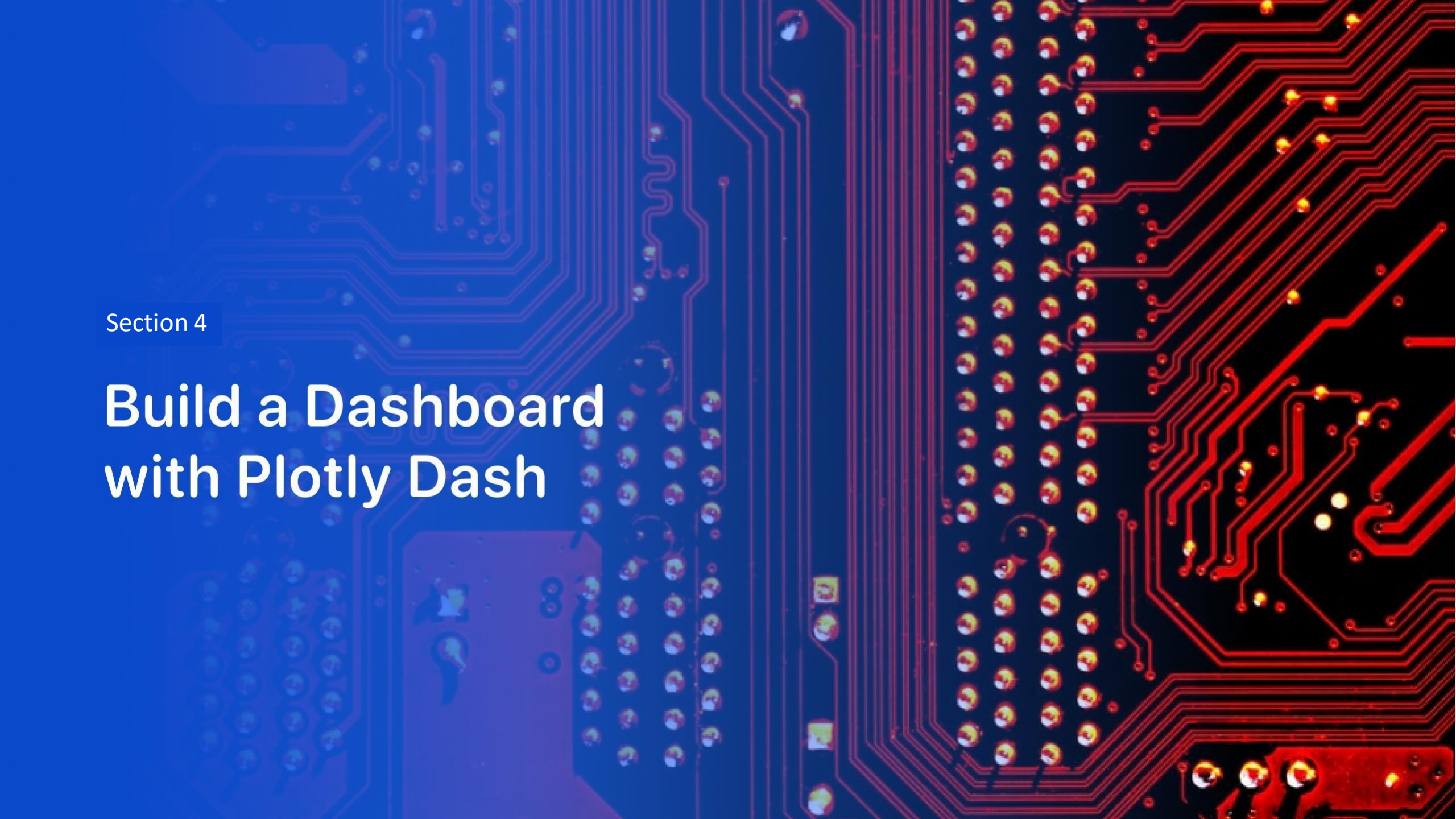
Folium Map with proximities around launch site

```
In [29]: distance_coastline
Out[29]: 1.349777616302522
```

```
In [27]: # Create a `folium.PolyLine` object using the coastline coordinates and launch site coordinate
         coordinates = [(launch_site_lat,launch_site_lon), (coastline_lat, coastline_lon)]
         lines=folium.PolyLine(locations=coordinates, weight=1)
         site_map.add_child(lines)
```

Out[27]:




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

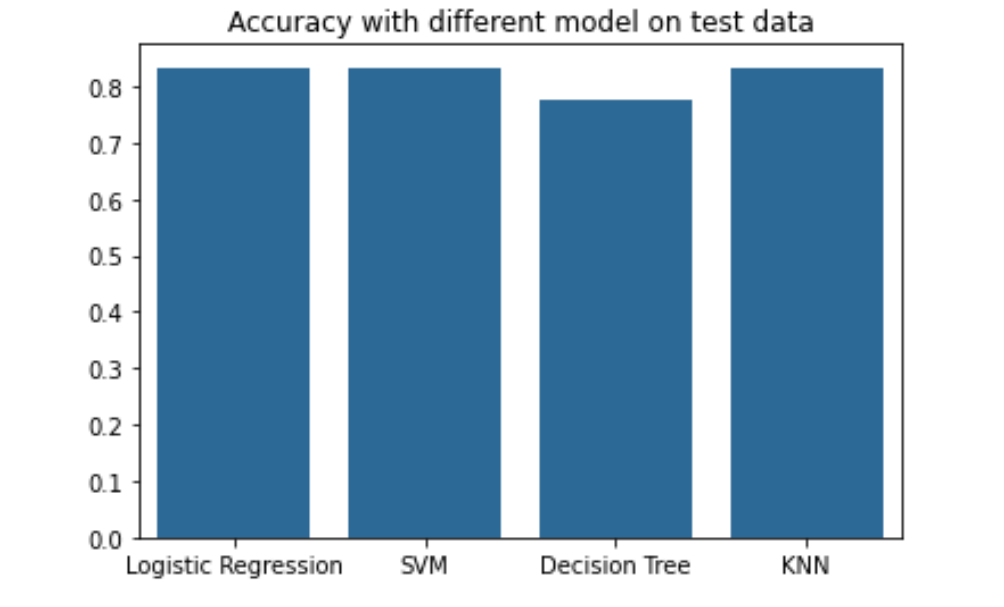
<Dashboard Screenshot 3>

- 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)

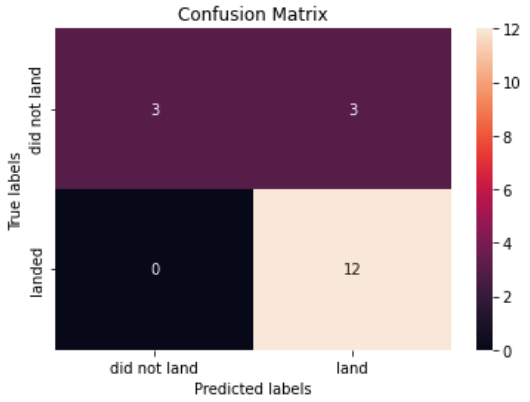
Classification Accuracy



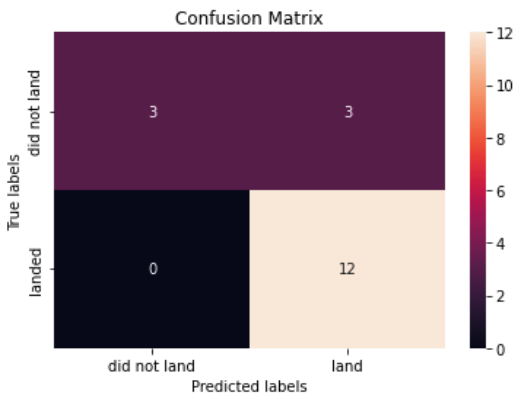
- Logistic Regression, SVM, KNN have similar accuracy

Confusion Matrix

```
In [17]: yhat=logreg_cv.predict(X_test)
         plot_confusion_matrix(Y_test,yhat)
```




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




```
In [32]: yhat = knn_cv.predict(X_test)
         plot_confusion_matrix(Y_test,yhat)
```

- The models have good TPR, which can successfully predict all ture landed correctly
- The models cannot identify very well on did not land situation

Conclusions

- Success rate of landing related to orbit type, pay load mass and launch site
- Success rate of landing is improving overtime
- Lauch sites should be near coastline
- Different orbit type related to different range of flight number
- Different launch sites handle different payload mass

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!