jupyter-labs-eda-sql-coursera
Project

<CHAU VAN VAN>

<30-AUG-2021>

Space missions data understanding, visualization and analysis EDA with SQL

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

Executive Summary

- Task 1:Display the names of the unique launch sites in the space mission
- Task 2:Display 5 records where launch sites begin with the string 'CCA‘
- Task3:Display the total payload mass carried by boosters launched by NASA (CRS)
- Task 4:Display average payload mass carried by booster version F9 v1.1
- Task 5:List the date when the first successful landing outcome in ground pad was achieved.
- 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
- Task 7:List the total number of successful and failure mission outcomes.
- Task 8: List the names of the booster versions which have carried the maximum payload mass. Use a sub query.
- Task 9:List the failed landing outcomes in drone ship, their booster versions, and launch site names for the in year 2015.
- 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

Introduction

Falcon 9

- SpaceX is a private spaceflight company that puts satellites into orbit and delivers cargo and, more recently, crew to the International Space Station (ISS). It was the first private company to send a cargo ship to the ISS, doing so in 2012. The company sent its first two astronauts to the ISS on May 30, 2020 aboard the SpaceX Crew Dragon, and followed that test flight with the successful launch of four astronauts on Nov. 15, 2020. As of early 2021, it is the only commercial spaceflight company capable of sending astronauts to space, although it may soon face competition from Boeing's CST-100 Starliner.

- Since June 2010, rockets from the Falcon 9 family have been launched **127 times**, with 125 full mission successes, one partial failure and one total loss of spacecraft. In addition, one rocket and its payload were destroyed on the launch pad during the fueling process before a static fire test.

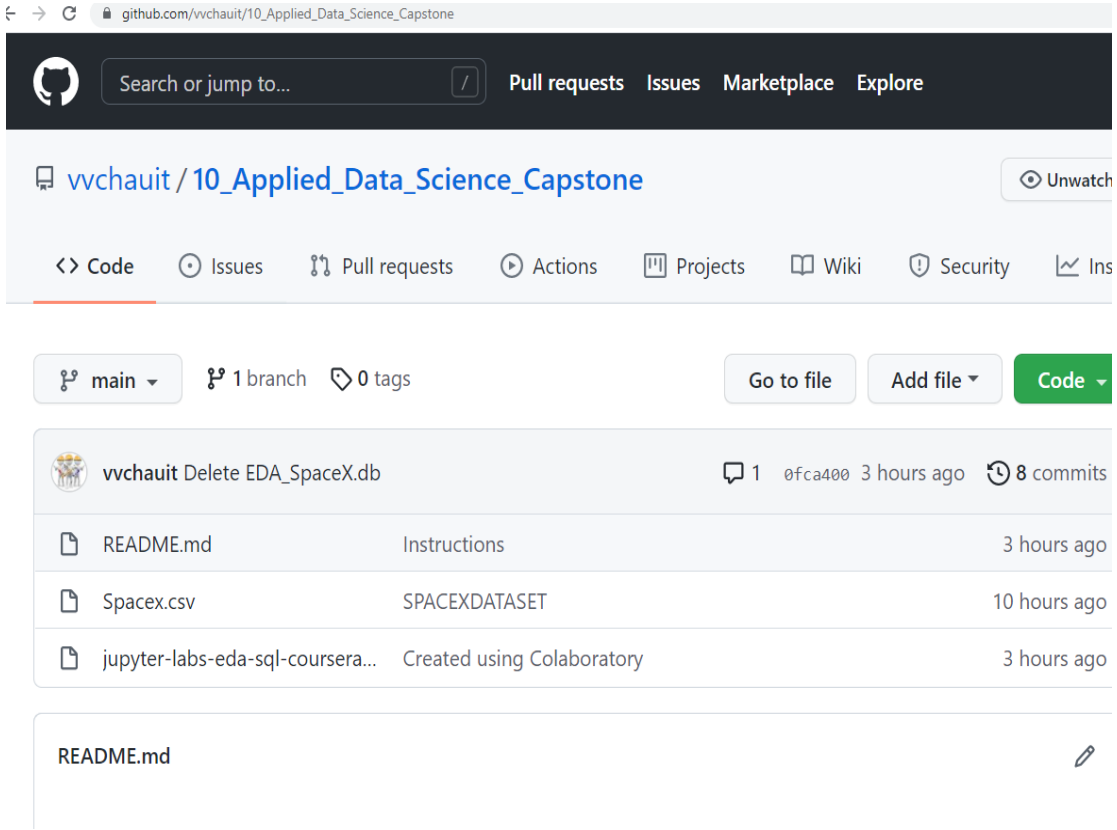
Methodology

- Data collection methodology:
  - Collect data from Dataset csv int database.
- Perform data wrangling
  - Collect and filter by SQL Select Syntax statement.
- 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

Methodology

EDA with SQL results slides

Data collection



- Data from my github account.

https://github.com/vvchauit/10_Applied_Data_Science_Capstone/blob/main/SpaceX.db.

```
import sqlite3
con = sqlite3.connect('/datalab/SpaceX.db')
cursorObj = con.cursor()
```

- **Task 1: Display the names of the unique launch sites in the space mission**

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

In [49]:
```
cursorObj.execute("SELECT Distinct Launch_site FROM Spacex;")
rows = cursorObj.fetchall()
df=pd.DataFrame(rows,columns=['Launch_site'])
df
```

Out[49]:

	Launch_site
0	CCAFS LC-40
1	VAFB SLC-4E
2	KSC LC-39A
3	CCAFS SLC-40

- **Task 2:*Display 5 records where launch sites begin with the string 'CCA'***

In [64]:
```
cursorObj.execute("SELECT * FROM Spacex WHERE Launch_site  LIKE 'CCA%' ;")
rows = cursorObj.fetchall()
df=pd.DataFrame(rows,columns=['Date','Time','Booster_Version','Launch_Site','Payload','PAYLOAD_MASS__KG_','Orbit','Customer','Mission_Outcome','Landing _Outcome'])
df.head()
```

Out[64]:

	Date	Time	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing _Outcome
0	2010-04-06	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
1	2010-08-12	15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight C1, two CubeSats, barrel of...	0	LEO (ISS)	NASA (COTS) NRO	Success	Failure (parachute)
2	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
3	2012-08-10	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
4	2013-01-03	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

- **Task 3: *Display the total payload mass carried by boosters launched by NASA (CRS)***

In [66]:
```
cursorObj.execute("SELECT Customer as [boosters launched],      SUM(PAYLOAD_MASS__KG_) AS [total payload mass] FROM Spacex WHERE Customer
='NASA (CRS)' Group by Customer ;")
rows = cursorObj.fetchall()
df=pd.DataFrame(rows,columns=['boosters launched','total payload mass'])
df
```

Out[66]:

	boosters launched	total payload mass
0	NASA (CRS)	45596

- **Task 4:*Display average payload mass carried by booster version F9 v1.1***

In [72]:
```
cursorObj.execute('''SELECT Booster_Version ,   AVG(PAYLOAD_MASS__KG_) AS [average payload mass] FROM Spacex WHERE Booster_Version ='F9 v 1.1'
                     Group by Booster_Version ;''')
rows = cursorObj.fetchall()
df=pd.DataFrame(rows,columns=['Booster_Version','average payload mass'])
df
```

Out[72]:

	Booster_Version	average payload mass
0	F9 v1.1	2928.4

- Task 5: *List the date when the first succesful landing outcome in ground pad was acheived.*

In [79]:
```
cursorObj.execute('''SELECT   min(DATE) as MinDate FROM Spacex WHERE  Landing_Outcome='Success (ground pad)'
ORDER by date ASC LIMIT 1;''')
rows = cursorObj.fetchall()
df=pd.DataFrame(rows,columns=['MinDate'])
df
```

Out[79]:

	MinDate
0	2015-12-22

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 [87]:
```
cursorObj.execute('''select "Booster_Version"
                     from Spacex where Landing_Outcome='Success (drone ship)'
                      and (PAYLOAD_MASS__KG_ BETWEEN 4000 and 6000)''')
rows = cursorObj.fetchall()
df=pd.DataFrame(rows,columns=['BOOSTER_VERSION'])
df
```

Out[87]:

	BOOSTER_VERSION
0	F9 FT B1022
1	F9 FT B1026
2	F9 FT B1021.2
3	F9 FT B1031.2

- Task 7: *List the total number of successful and failure mission outcomes*

In [89]:
```
cursorObj.execute('''select count(MISSION_OUTCOME) as missionoutcomes from SPACEX GROUP BY MISSION_OUTCOME''')
rows = cursorObj.fetchall()
df=pd.DataFrame(rows,columns=['missionoutcomes'])
df
```

Out[89]:

	missionoutcomes
0	1
1	97
2	1
3	1

- **Task 8:*List the names of the booster_versions which have carried the maximum payload mass. Use a subquery***

In [91]:
```
cursorObj.execute('''select Booster_Version, as [booster version] from SPACEX where PAYLOAD_MASS__KG_=(select max(PAYLOAD_MASS__KG_) from SPACEX)''')
rows = cursorObj.fetchall()
df=pd.DataFrame(rows,columns=['booster version'])
df
```

Out[91]:

	booster version
0	F9 B5 B1048.4
1	F9 B5 B1049.4
2	F9 B5 B1051.3
3	F9 B5 B1056.4
4	F9 B5 B1048.5
5	F9 B5 B1051.4
6	F9 B5 B1049.5
7	F9 B5 B1060.2
8	F9 B5 B1058.3
9	F9 B5 B1051.6
10	F9 B5 B1060.3
11	F9 B5 B1049.7

- Task 9: *List the failed landing_outcomes in drone ship, their booster versions, and launch site names for the in year 2015*

In [115]:
```
cursorObj.execute('''select Date, Booster_Version,Launch_Site,Landing_Outcome from SPACEX
                     where Landing_Outcome= 'Failure (drone ship)' and date like'2015%';''')

rows = cursorObj.fetchall()
df=pd.DataFrame(rows,columns=['Date','Booster_Version','Launch_Site','Landing_Outcome'])
df
```

Out[115]:

	Date	Booster_Version	Launch_Site	Landing_Outcome
0	2015-10-01	F9 v1.1 B1012	CCAFS LC-40	Failure (drone ship)
1	2015-04-14	F9 v1.1 B1015	CCAFS LC-40	Failure (drone ship)

- **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 [119]:
```
cursorObj.execute('''SELECT Landing_Outcome FROM SPACEX
 WHERE Landing_Outcome= 'Failure (drone ship)' or Landing_Outcome= 'Success (ground pad)'  and (DATE BETWEEN '2010-06-04' AND '2017-03-2
0') ORDER BY DATE DESC;''')
rows = cursorObj.fetchall()
df=pd.DataFrame(rows,columns=['Landing_Outcome'])
df
```

Out[119]:

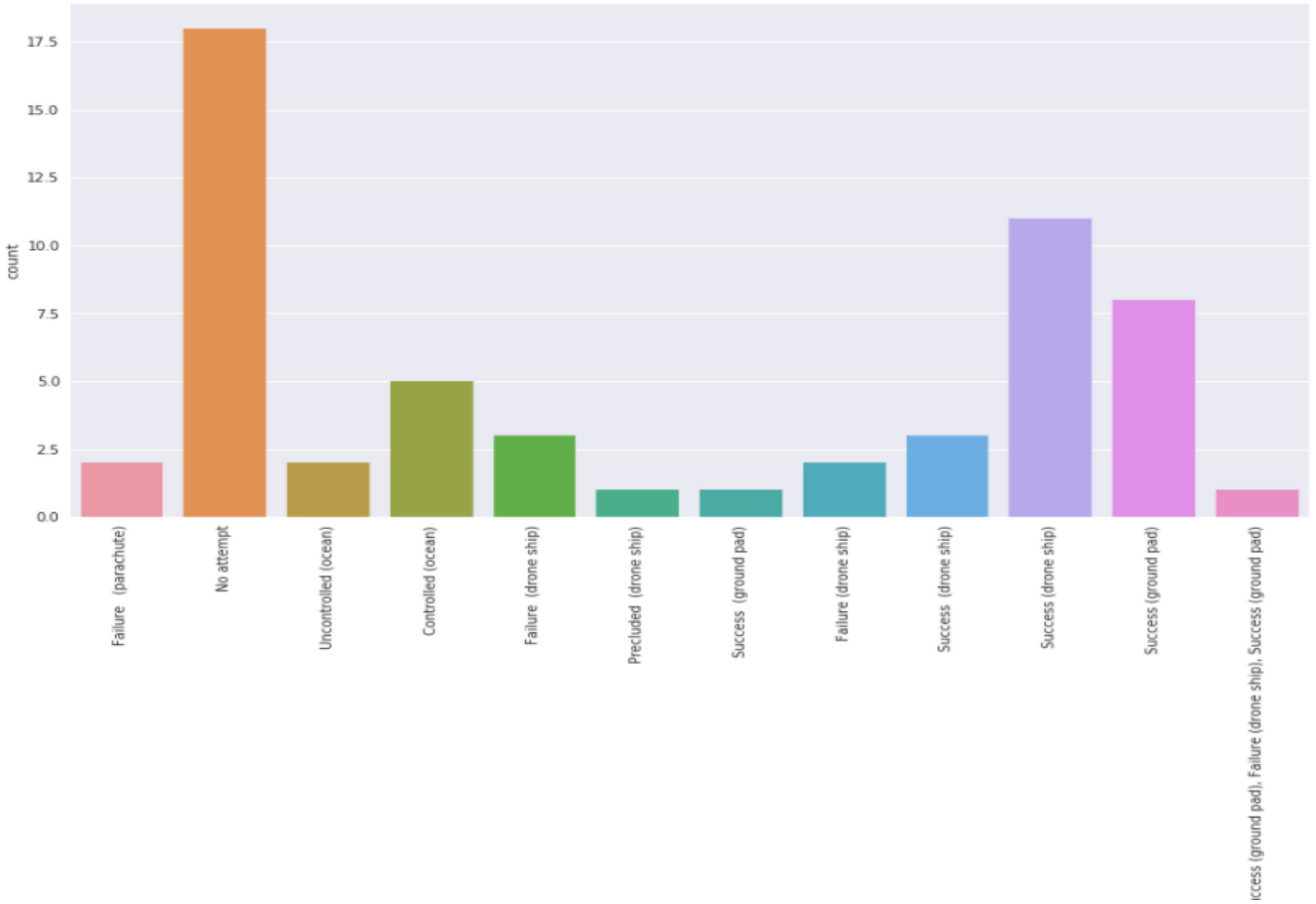
	Landing_Outcome
0	Success (ground pad)
1	Success (ground pad)
2	Success (ground pad)
3	Success (ground pad)
4	Failure (drone ship)
5	Failure (drone ship)
6	Failure (drone ship)
7	Success (ground pad)
8	Failure (drone ship)
9	Failure (drone ship)

EDA with Visualization

Data visualization of SpaceX launch data

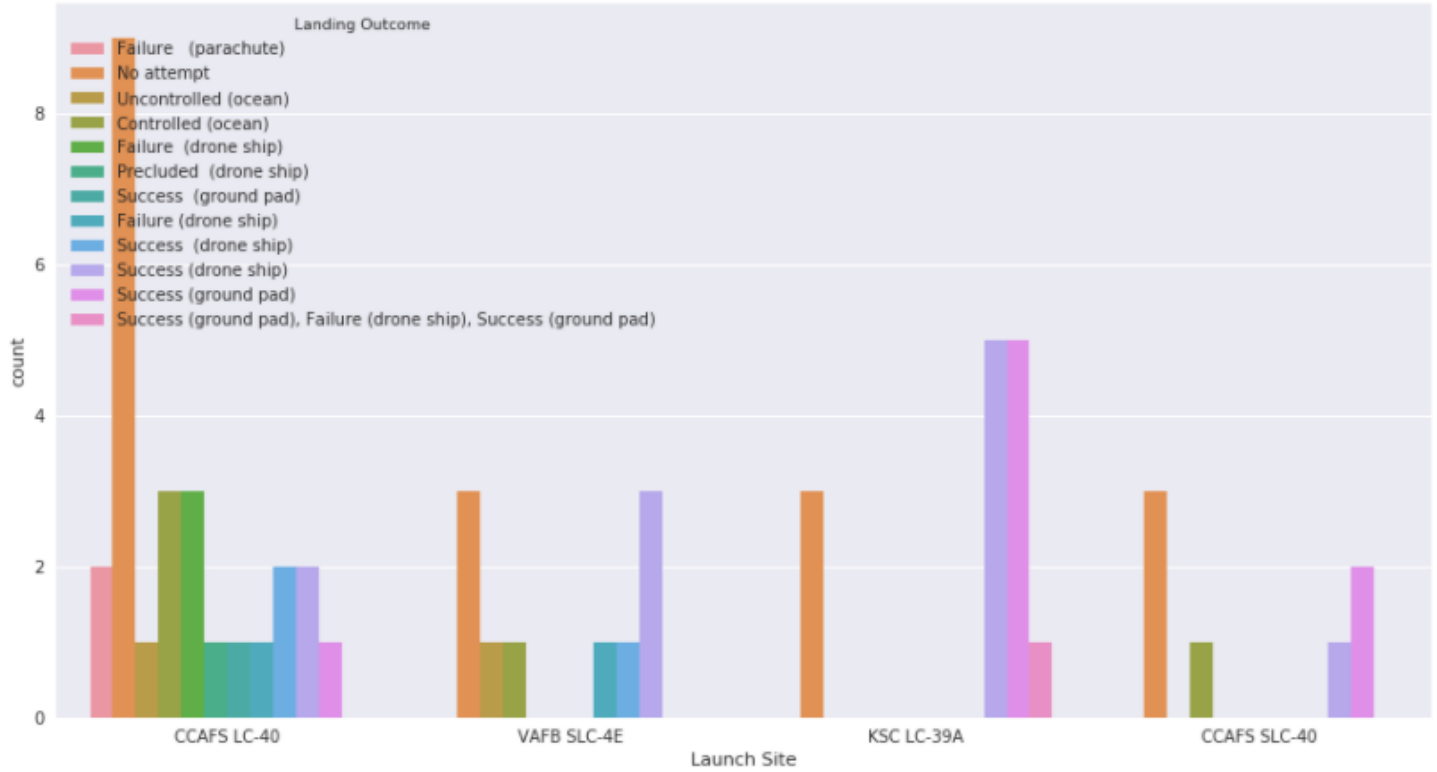
Data visualization of SpaceX launch data

```
sns.set(rc={'figure.figsize':(15,8)})
plt.xticks(rotation=90)
sns.countplot(x="Landing Outcome", data=df)
```



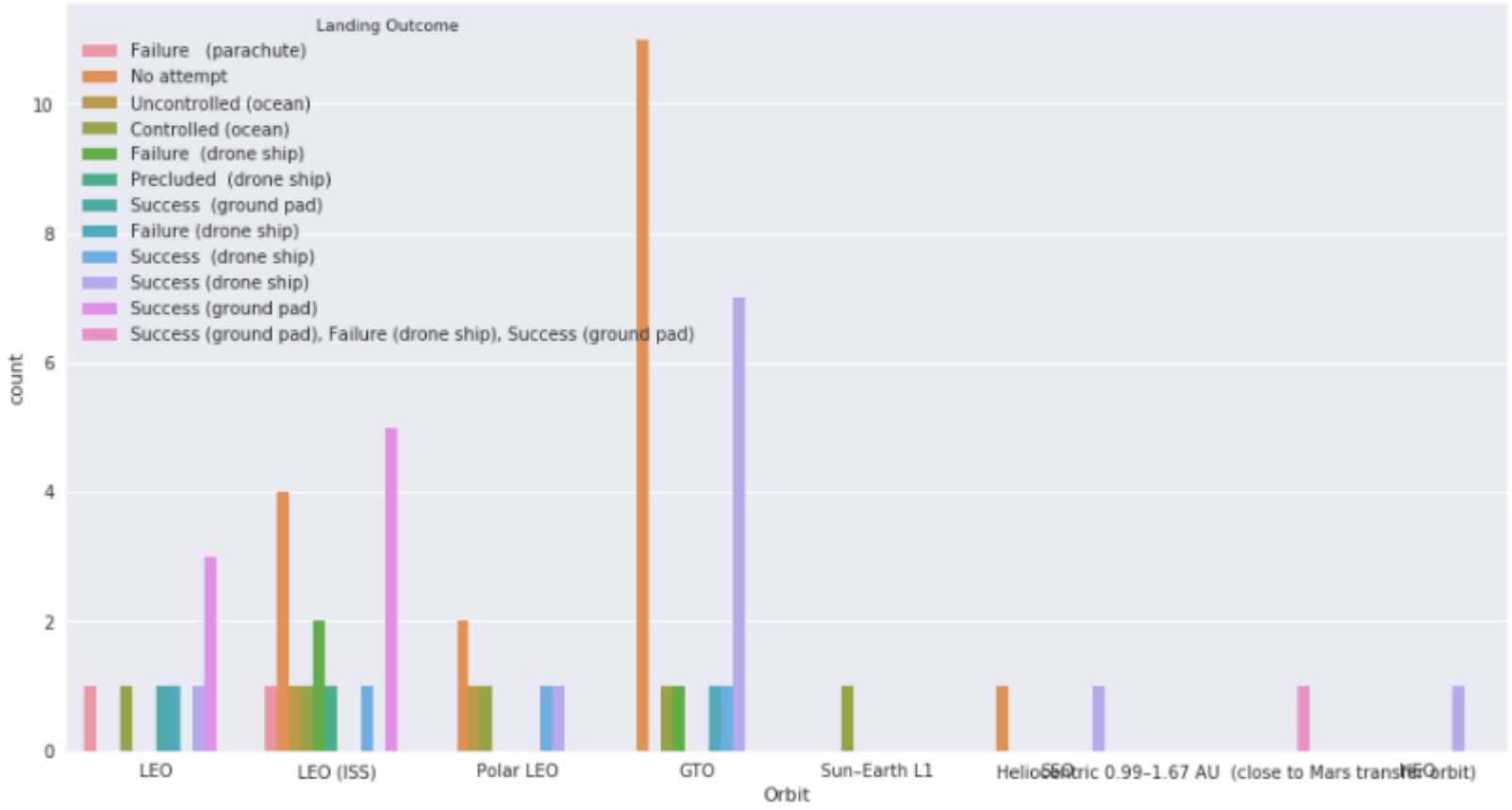
Data visualization of SpaceX launch data

```
g=sns.countplot(x="Launch Site",
data=df,hue="Landing Outcome");
```



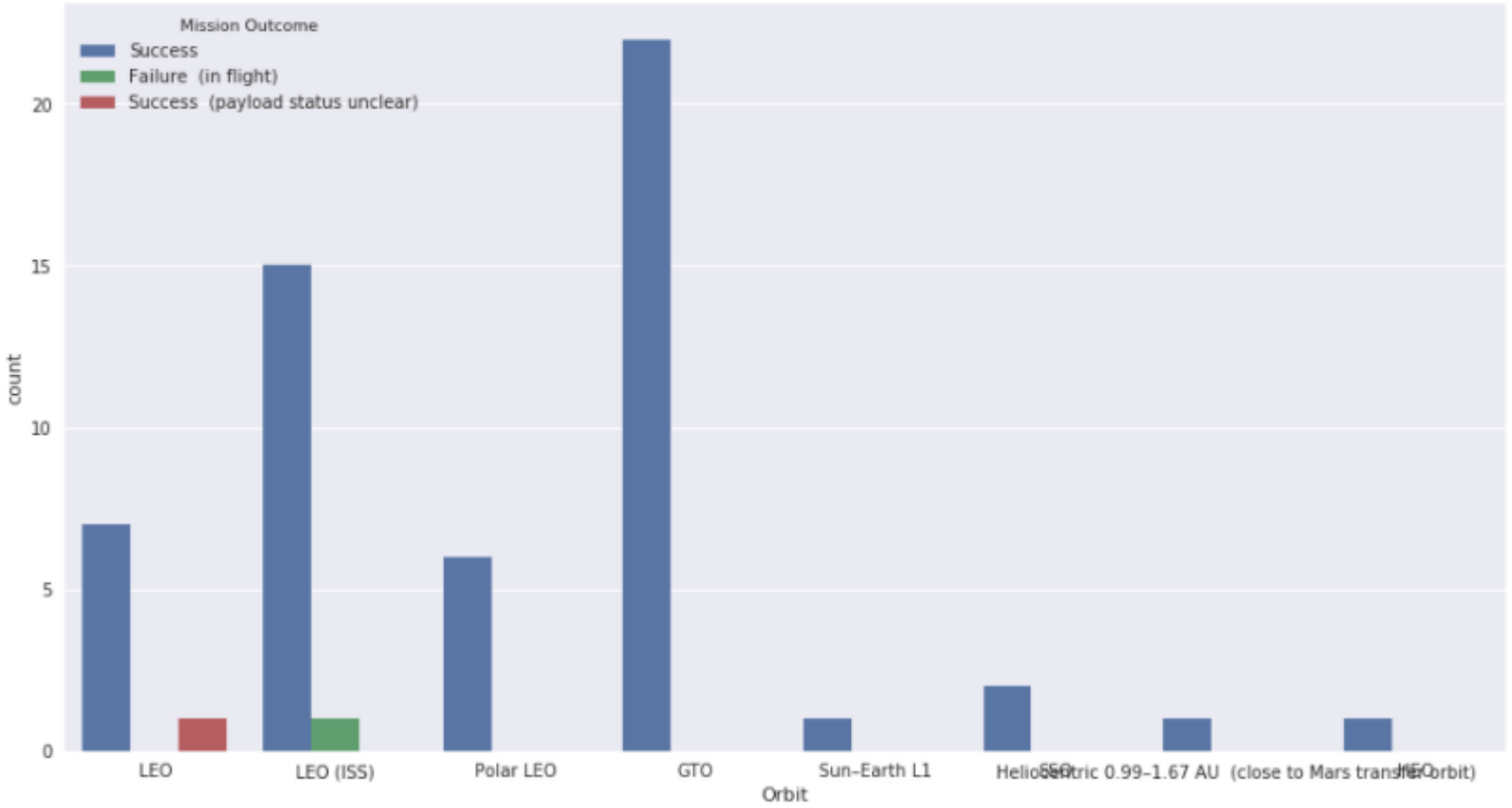
Data visualization of SpaceX launch data

```
g=sns.countplot(x="Orbit", data=df,hue="Landing Outcome");
```

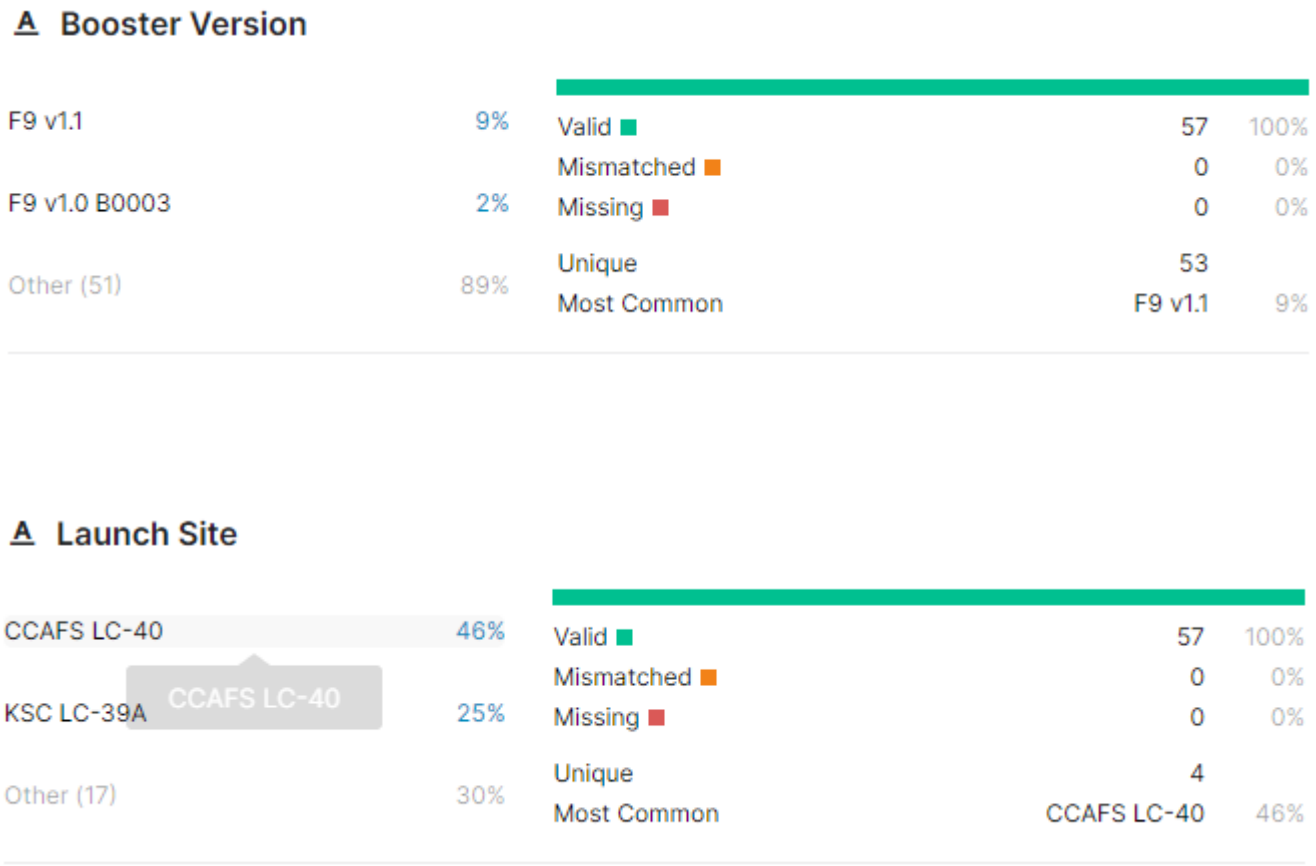


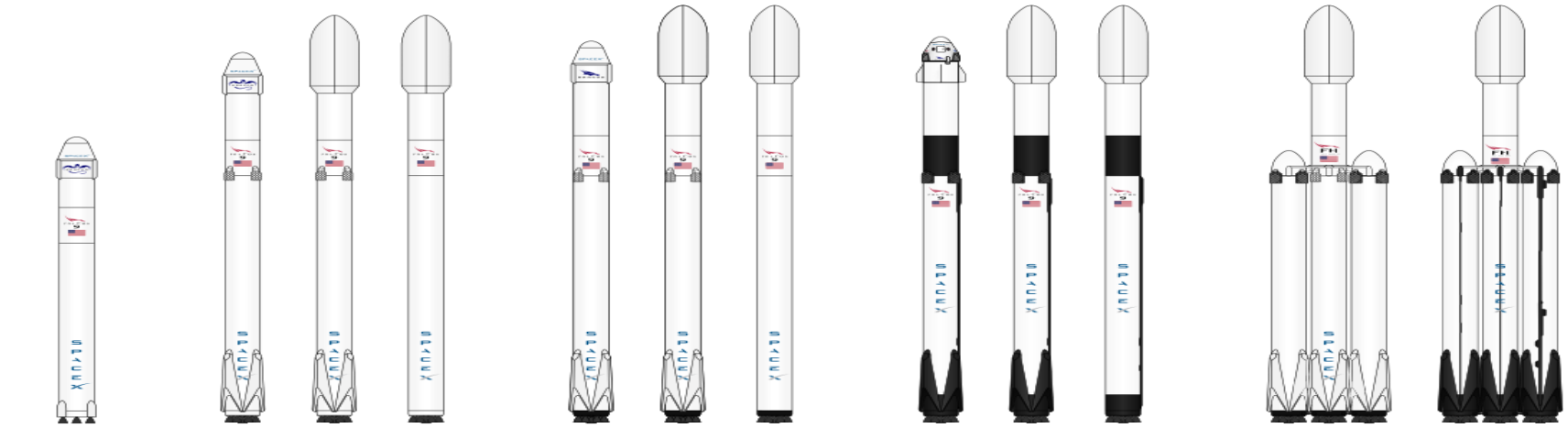
Data visualization of SpaceX launch data

```
g=sns.countplot(x="Orbit",
data=df,hue="Landing Outcome");
```



Data visualization of SpaceX launch data

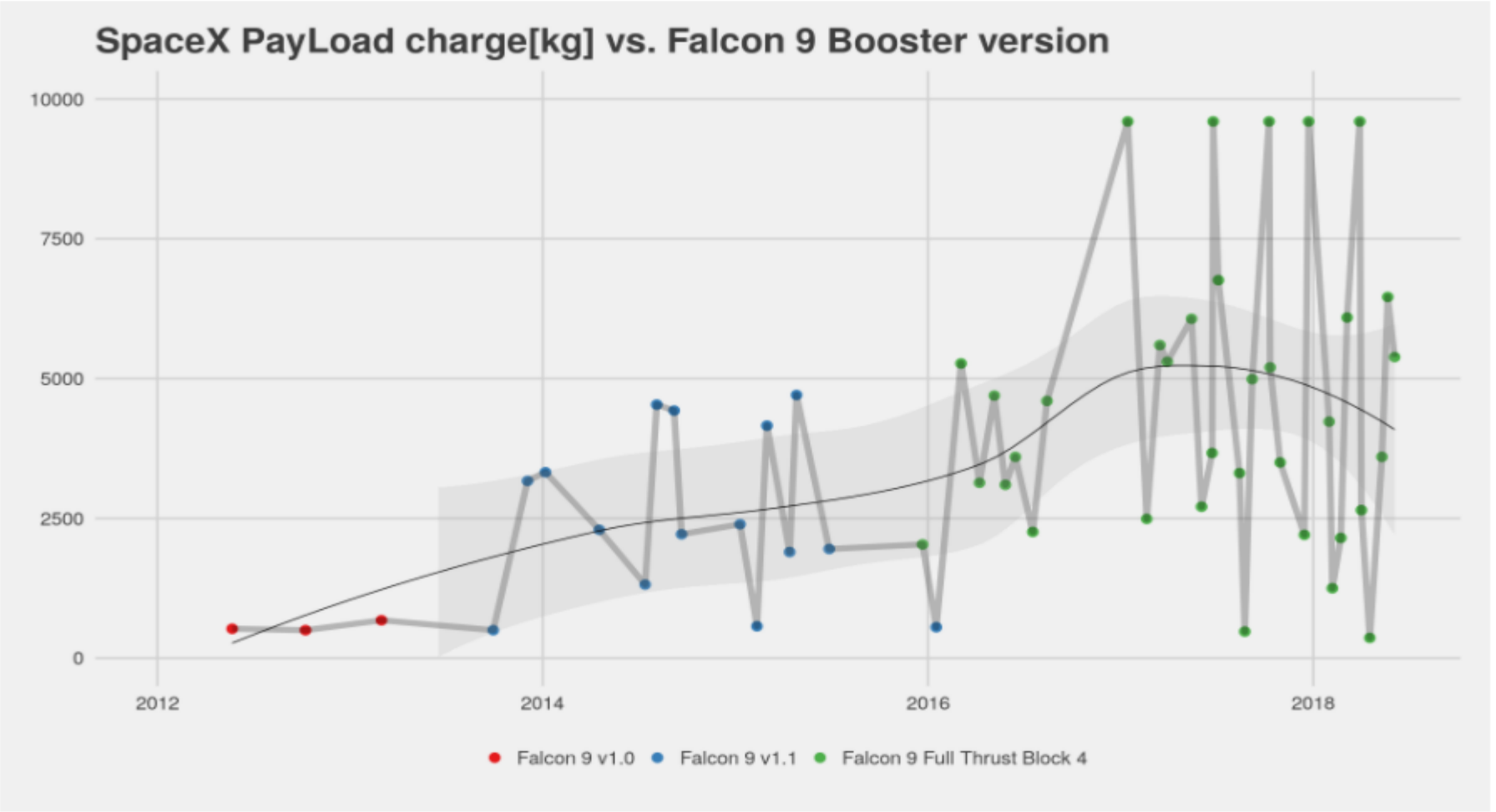




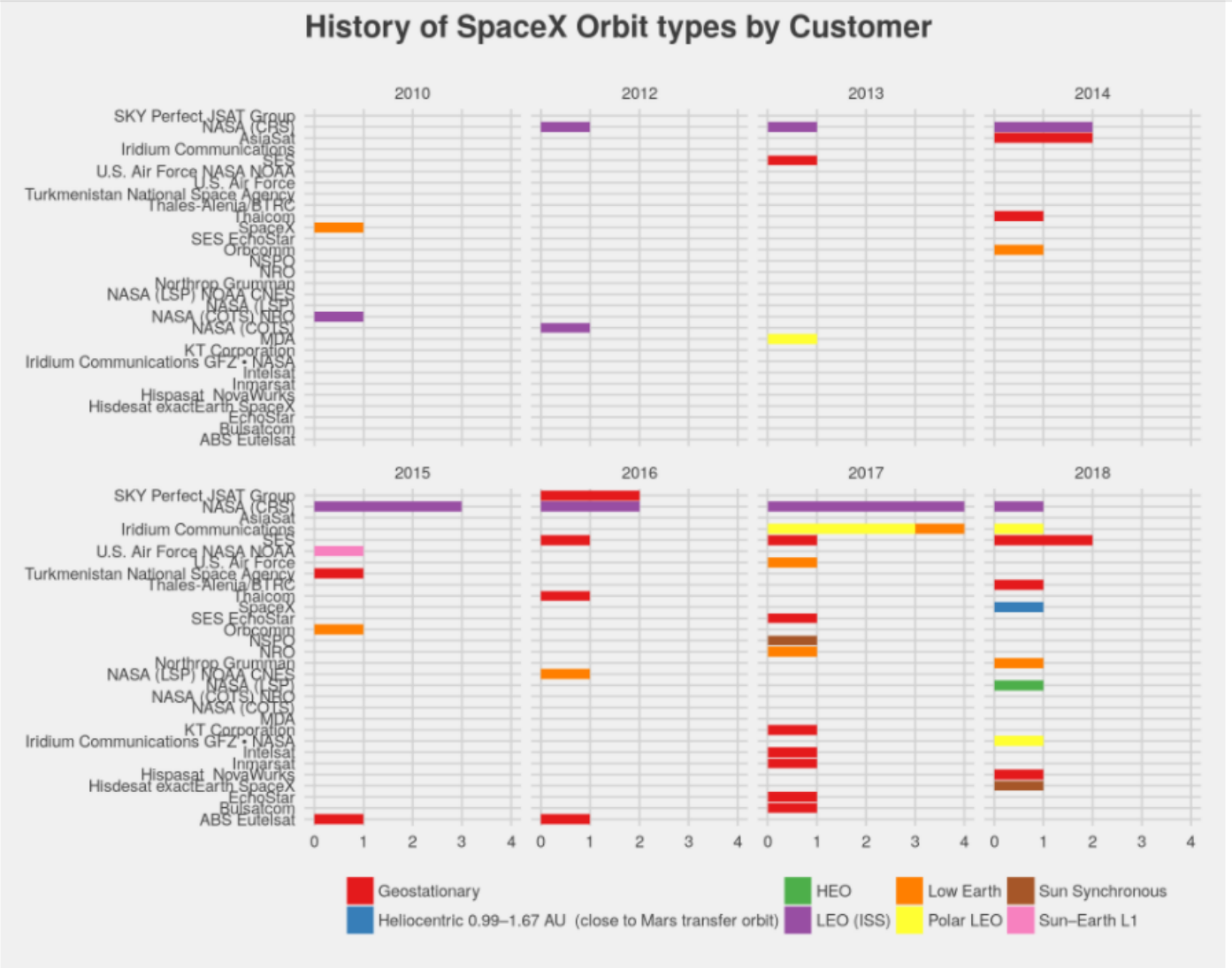
Interactive map with Folium

Payload vs. Booster version

Folium map screenshot 1



Folium map screenshot 2



CONCLUSION

- Data science is new, and we are still working out what it is. At the moment, it is best defined by what a data scientist does. IBM Data Science is a data scientist uses programming as the basis for a deeper, more flexible approach to data analysis

APPENDIX

- Include any relevant assets like Python code snippets, SQLite queries, charts, Colab Notebook outputs,and data simulation of SpaceX that I used during this project.

Author: Chau Van

Email:vvchauit@gmail.com