IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

<Jay Zhu>
<2022-03-06>

Outline

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

Executive Summary

- Summary of methodologies
  - data collection through REST API and Webscraping
  - data wrangling
  - Exploratory data analysis with data visualization
  - Map vasualization via Folium
  - Building a dashboard with plotly dash
  - Predictive analysis
- Summary of all results
  - the query result from database table
  - charts represents correleationship within multiple variables
  - the bar chart compares training accuracy within all four classification methods

Introduction

- Project background and context
  - SpaceX is one of the most successful commercial rocket companies today. So, it’s net profit is crucial to both its development and investors. Despite of other controllabe factors, the success of launch is much more important because minor details may cause the failure, such as payload, payload mass, orbit, version of booster, launch site. In this case, we may use the existing launch data to predict the whether or not the each launch will be successful to minimize the risks.
- Problems you want to find answers
  - Find out the relationship between each available factor and successful rate
  - Determine the best method of making decisions on rocket launch

Section 1

Methodology

Methodology

Executive Summary

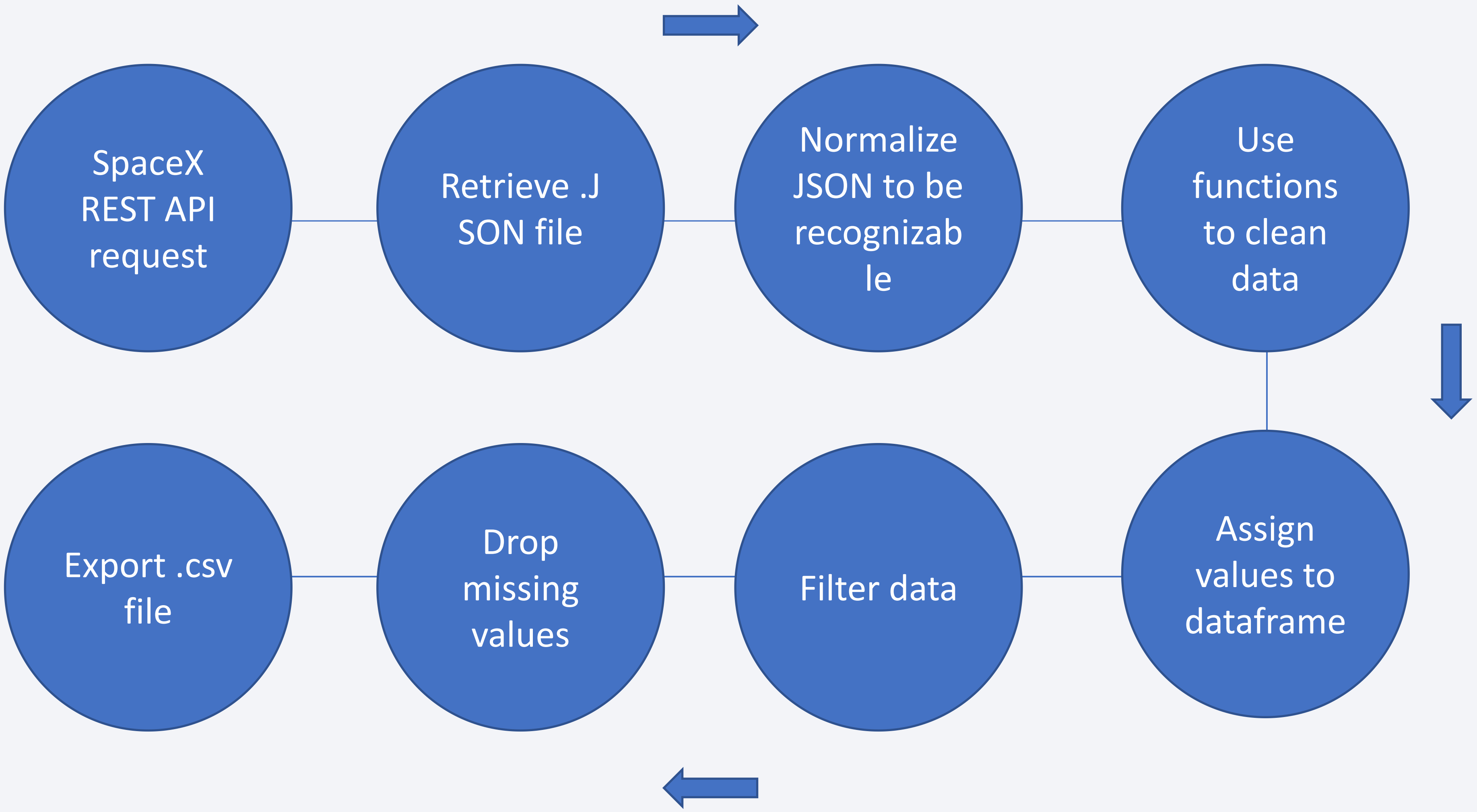
- Data collection methodology:
    - Request API response from Space X and websracpe historical data from Space X Wikipedia page
- Perform data wrangling
    - Process the data by normalizing the format and drop irrevelant values
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models

  Using logistic regression, decision tree, SVM, and KNN to evaluate the model with the best accuracy by giving cross validation method.

Data Collection

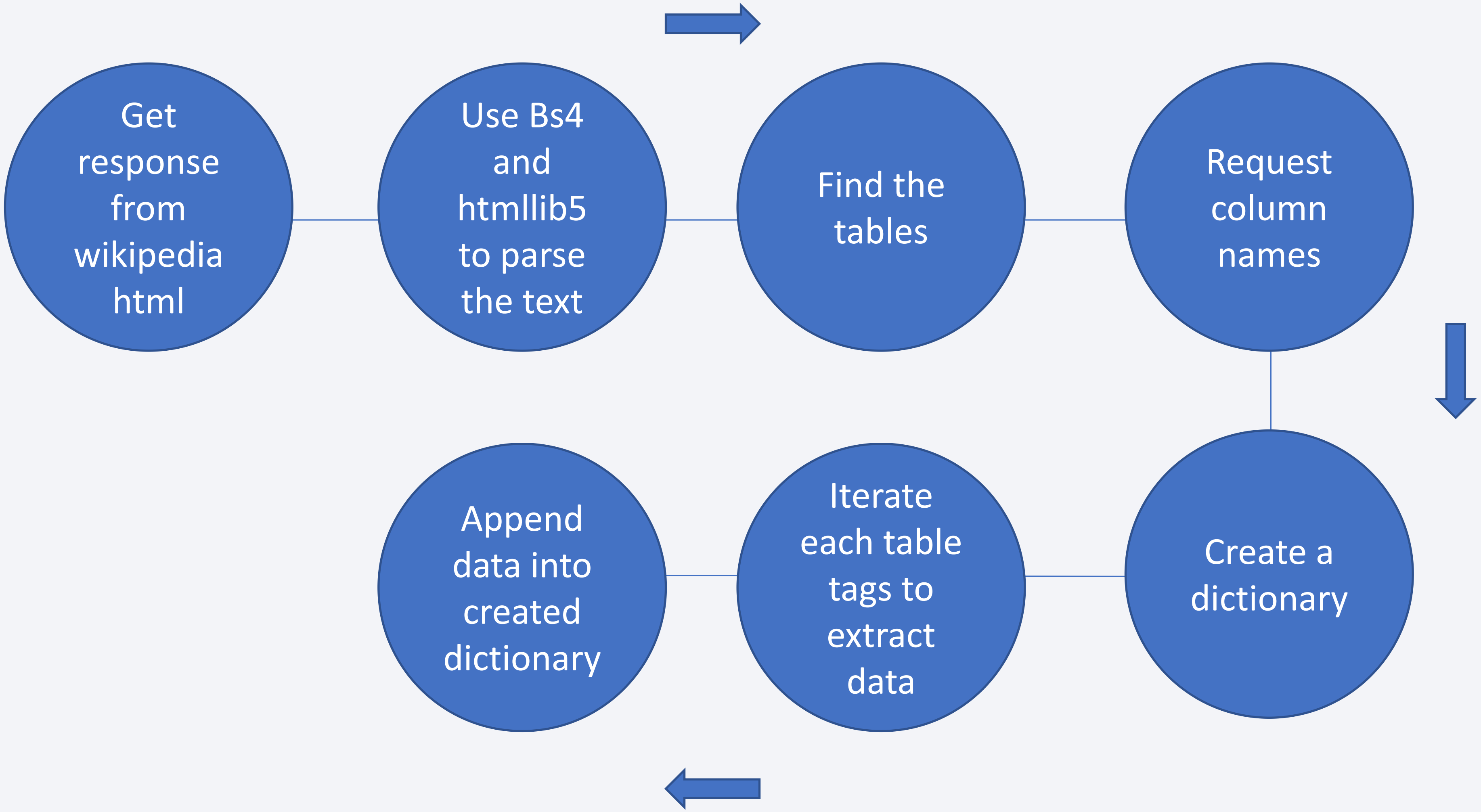
- Describe how data sets were collected.
  - The data is sourced from SpaceX REST API and Webscraping of SpaceX wikipedia page by Beautifulsoup
- The next few slides present how data collection is processed by using key phrases and flowcharts

Data Collection – SpaceX API



https://github.com/soap945/DS_CAPSTONE_FINISHED/blob/9359601386c69dce965c89f06c1b84ebd52f6e2a/DS_CAPSTONE_FINISHED/1.0%20jupyter-labs-webscraping%20(1).ipynb

Data Collection - Scraping



https://github.com/soap945/DS_CAPSTONE_FINISHED/blob/9359601386c69dce965c89f06c1b84ebd52f6e2a/DS_CAPSTONE_FINISHED/1.0%20jupyter-labs-webscraping%20(1).ipynb

Data Wrangling

- Describe how data were processed
  - Initiate the categorical data to be as binary data where success = 1, and failure = 0
  - Perform a exploratory data analysis to find out the success reate of landing and other statistical data such sum of launches, landing outcomes, correleations between different variables.

https://github.com/soap945/DS_CAPSTONE_FINISHED/blob/9359601386c69dce965c89f06c1b84ebd52f6e2a/DS_CAPSTONE_FINISHED/1.1%20labs-jupyter-spacex-Data%20wrangling%20(1).ipynb

EDA with Data Visualization

- Summarize what charts were plotted and why you used those charts
  - Flight Number vs. Launch Site (Scatter plot)
  - Payload vs. Launch Site (Scatter plot)
  - Success Rate vs. Orbit Type (Bar chart)
  - Flight Number vs. Orbit Type (Scatter plot)
  - Launch Success Yearly Trend (Line graph)
  - Payload vs. Orbit Type (Scatter plot)
- Comapred with different charts
  - Scatter plot shows the correlation between one variable and another. It also shows the frequency in a certain range that two variables are highly related.
  - Bar chart can easily campare one category to another horizontally with the change on Y-axis.
  - Line graph can tell us the change over time and help us to predict the future trend.

https://github.com/soap945/DS_CAPSTONE_FINISHED/blob/9359601386c69dce965c89f06c1b84ebd52f6e2a/DS_CAPSTONE_FINISHED/2.1%20jupyter-labs-eda-dataviz%20(1).ipynb

EDA with SQL

- Modify the data format before importing into IBM DB2 database.
- load data set from IBM DB2 database.
- Queried using IBM DB2 API and coded in Python.
- Run several queries to get understanding of the data set

https://github.com/soap945/DS_CAPSTONE_FINISHED/blob/9359601386c69dce965c89f06c1b84ebd52f6e2a/DS_CAPSTONE_FINISHED/2.0%20jupyter-labs-eda-sql-coursera%20(1).ipynb

Build an Interactive Map with Folium

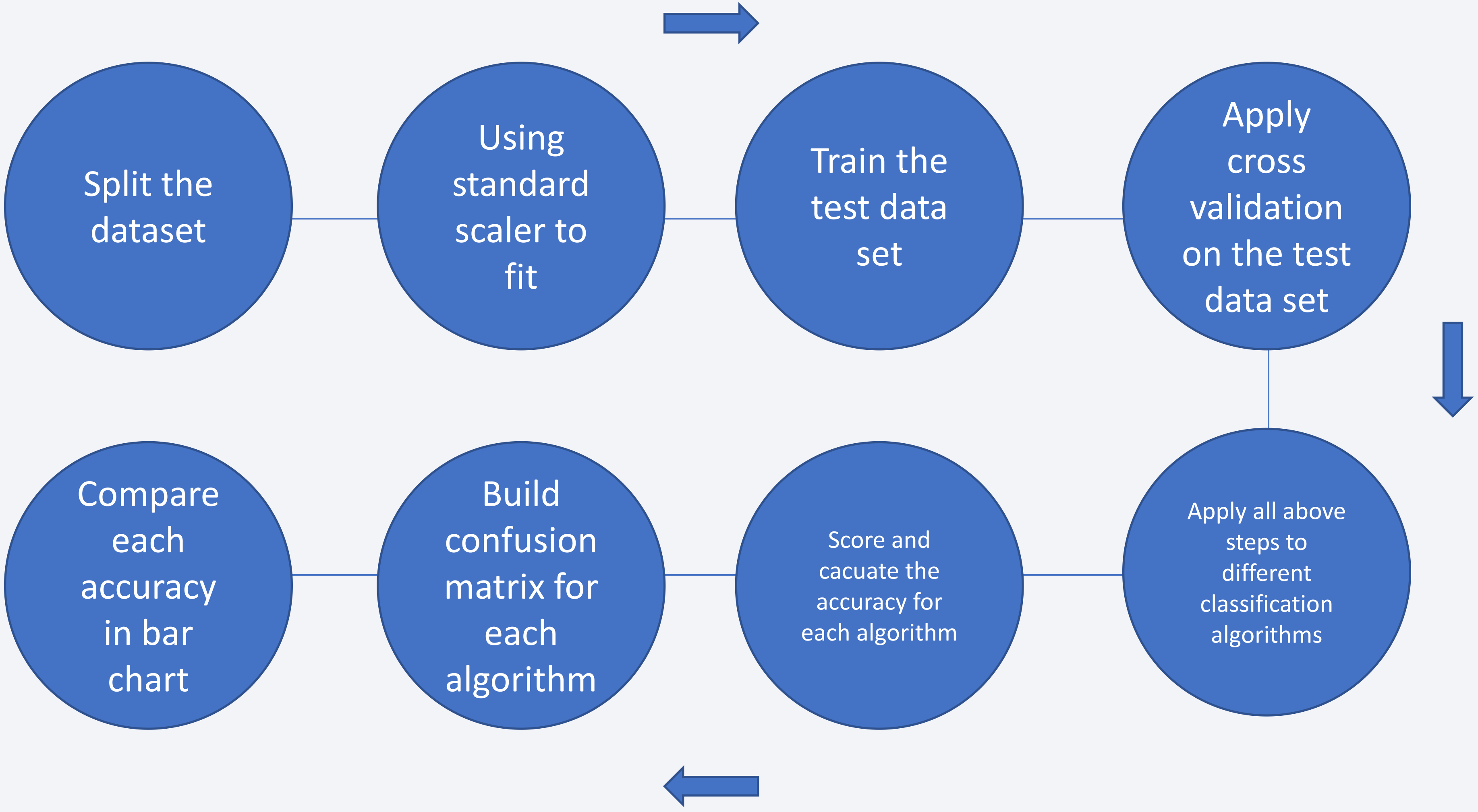
- Folium is a tool that helps us to visualize the location and distance on the map through analysis. Ther markers identifies landing position either success or failed ones by using green and red colors.

https://github.com/soap945/DS_CAPSTONE_FINISHED/blob/9359601386c69dce965c89f06c1b84ebd52f6e2a/DS_CAPSTONE_FINISHED/3.0%20lab_jupyter_launch_site_location_dashboard_vis.ipynb

Build a Dashboard with Plotly Dash

- Two charts added: scatter plot and pie chart
- Scatter plot indicates each success varies across launch sites and payload mass
- Pie chart can easily tell the portion of success landing on each site

Predictive Analysis (Classification)



https://github.com/soap945/DS_CAPSTONE_FINISHED/blob/9359601386c69dce965c89f06c1b84ebd52f6e2a/DS_CAPSTONE_FINISHED/4.0%20SpaceX_Machine%20Learning%20Prediction_Part_5%20(1).ipynb

Results (avaliable on the following slides)

Results includes

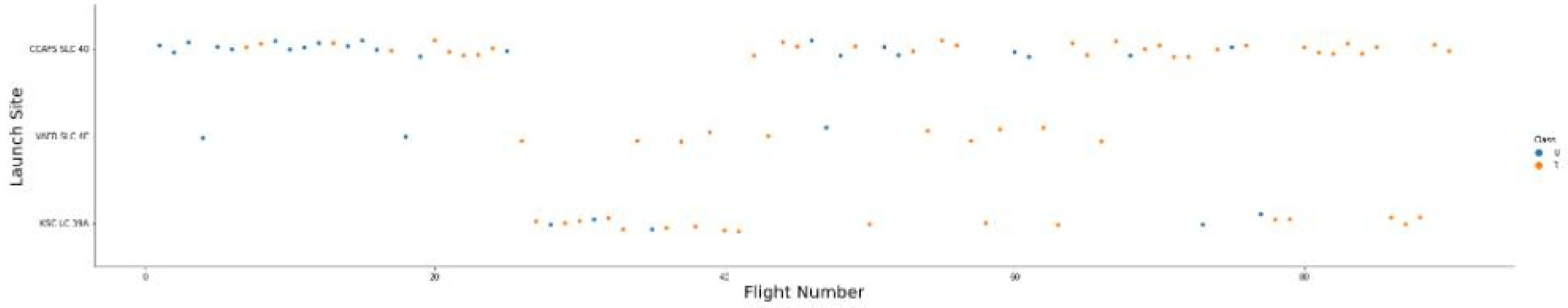
- Exploratory data analysis results
- Interactive analytics demo in screenshots
- Predictive analysis results

Section 2

Insights drawn from EDA

Flight Number vs. Launch Site

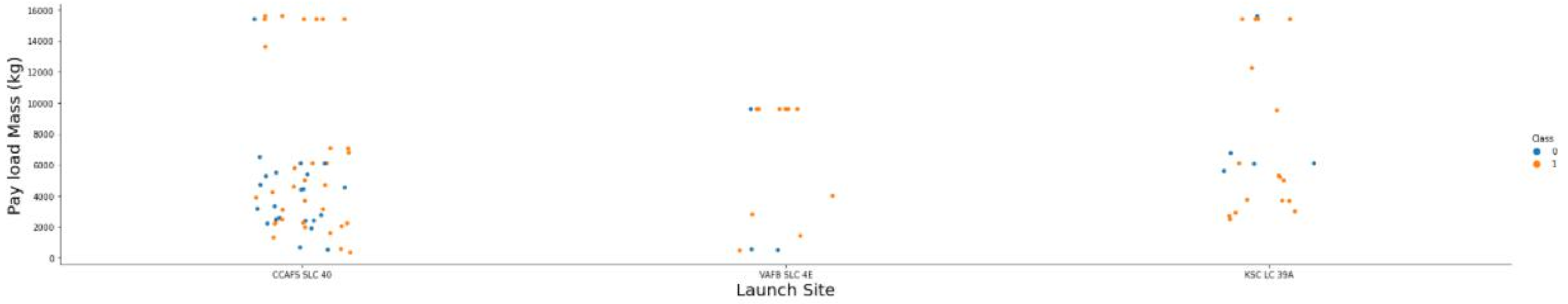
In [45]:
```
# Plot a scatter point chart with x axis to be Flight Number and y axis to be the launch site, and hue to be the class value
sns.catplot(x="FlightNumber", y="LaunchSite", hue="Class", data=df, aspect = 5)
plt.xlabel("Flight Number",fontsize=20)
plt.ylabel("Launch Site",fontsize=20)
plt.show()
```



We can tell from the chart that CCAFS SLC 40 has the most flight numbers.

Payload vs. Launch Site

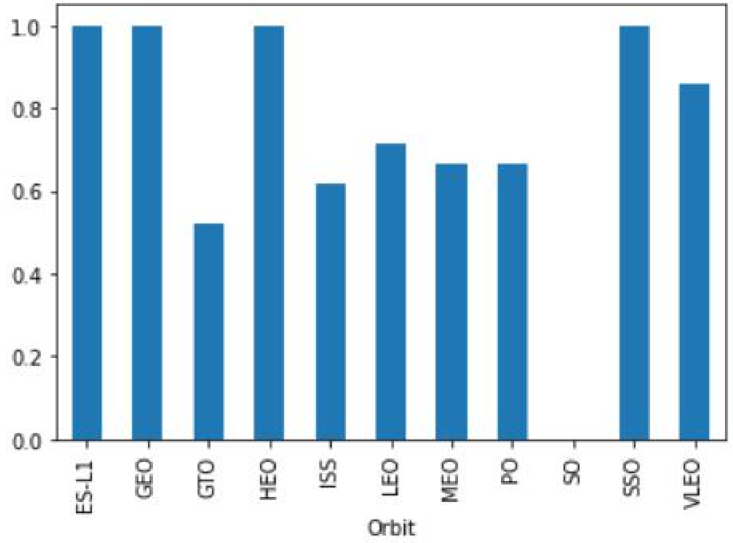
```
In  [46]: # Plot a scatter point chart with x axis to be Pay Load Mass (kg) and y axis to be the launch site, and hue to be the class value
sns.catplot( x="LaunchSite", y="PayloadMass", hue="Class", data=df, aspect = 5)
plt.xlabel("Launch Site",fontsize=20)
plt.ylabel("Pay load Mass (kg)",fontsize=20)
plt.show()
```



We can tell from the chart that CCAFS SLC 40 not only launches the most rocket, but also is capable of launching the heaviest rocket. The payload mass is ranging from lowest to the highest.

Success Rate vs. Orbit Type

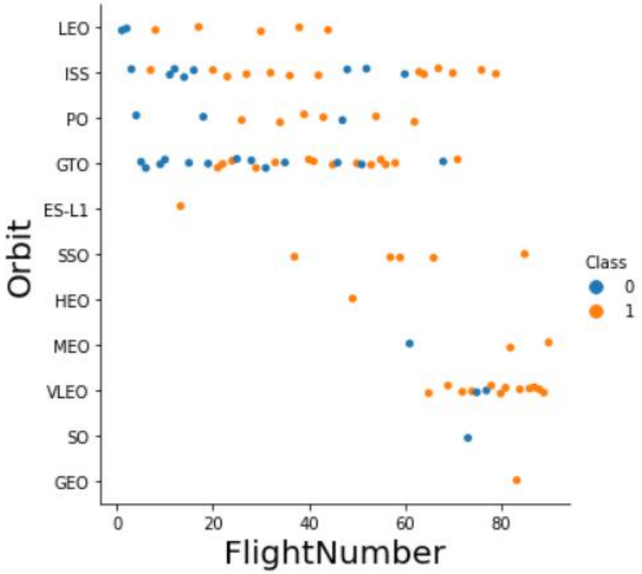
In [51]:
```
# HINT use groupby method on Orbit column and get the mean of Class column
df.groupby(['Orbit']).mean()['Class'].plot(kind='bar')

plt.show()
```



Over half of orbit is able to obtain rocket over 60% success rate

Flight Number vs. Orbit Type

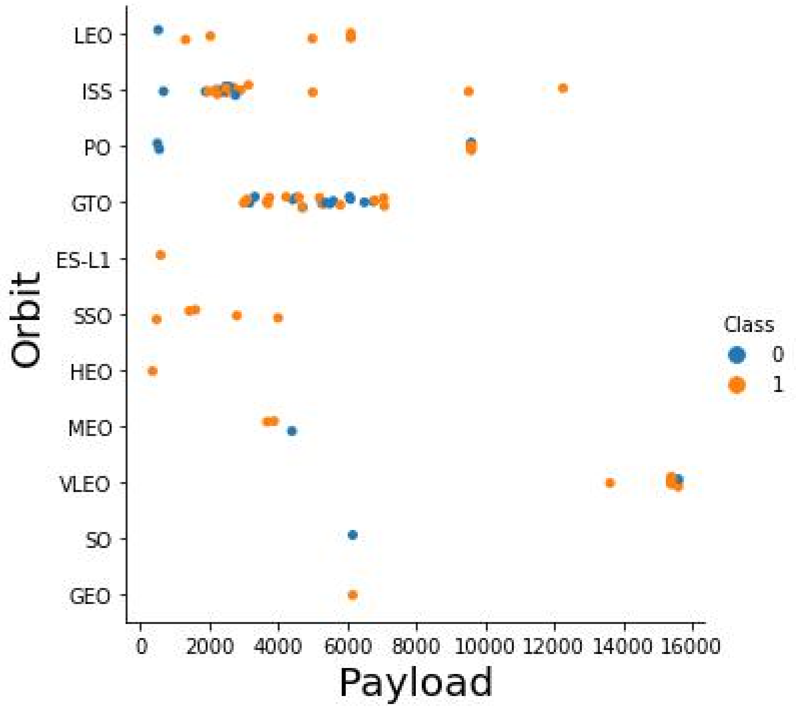
In [48]:
```
# Plot a scatter point chart with x axis to be FlightNumber and y axis to be the Orbit, and hue to be the class value
sns.catplot(x="FlightNumber", y="Orbit", hue="Class", data=df)
plt.xlabel("FlightNumber", fontsize=20)
plt.ylabel("Orbit", fontsize=20)
plt.show()
```



GTO orbit shows fifty-fty chance of successful landing. So, it is hard to tell the correlationship between GTO orbit and Flight numbers

Payload vs. Orbit Type

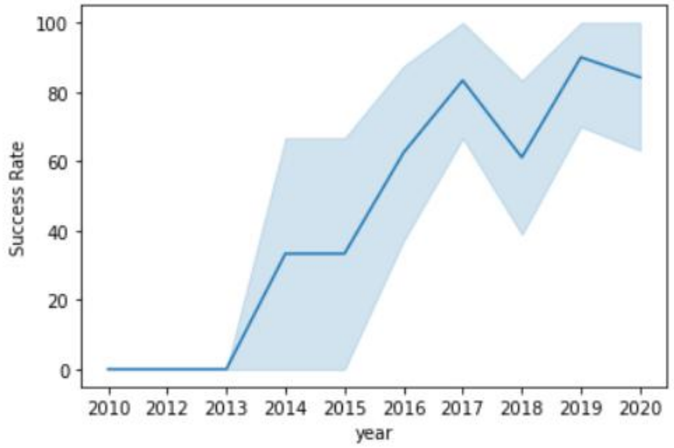
In [49]:
```
# Plot a scatter point chart with x axis to be Payload and y axis to be the Orbit, and hue to be the class value
sns.catplot(x="PayloadMass", y="Orbit", hue="Class", data=df)
plt.xlabel("Payload", fontsize=20)
plt.ylabel("Orbit", fontsize=20)
plt.show()
```



VLEO orbit needs highest payload of rocket. Most launches happened between 2000 - 10000 payload mass. GTO orbit has most failures.

Launch Success Yearly Trend

```
In [32]: # Plot a line chart with x axis to be the extracted year and y axis to be the success rate
         year = []
         df["year"] = Extract_year(year)
         sns.lineplot(data = df, x = "year", y = "Success Rate")

Out[32]: <AxesSubplot:xlabel='year', ylabel='Success Rate'>
```



Overall, the success rate is increasing over the past 7 years can close to 80%. There is a sudden drop within 2017-2018 that we cannot tell what cause it from our existing data.

All Launch Site Names

Out[28]:

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

Four unique launch site names are available in the dataset

Launch Site Names Begin with 'CCA'

Out[32]:

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	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
2012-10-08	00:35:00	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

All 5 CCA launch sites either fail on landing or the attempt were cancelled. It launched F9 version booster over from 2010-2013

Total Payload Mass

Out[40]:

SUM
48213

The total of payload is 48,213 kg.

Average Payload Mass by F9 v1.1

Done.

Out[41]:

average
2928

Average payload mass by F9 v1.1 is 2,928 kg.

First Successful Ground Landing Date

```
In  [21]: %%sql
select min(date) as Date from SPACEXTBL
where mission_outcome like 'Success'

 * ibm_db_sa://lvr03390:***@6667d8e9-9d4d-4ccb-ba32-21da3bb5aafc.clogj3sd0tgtu0lqde00.databases.appdomain.cloud:3
Done.
```

Out[21]:

DATE
2010-06-04

Successful Drone Ship Landing with Payload between 4000 and 6000

In [62]:

```
%%sql
select booster_version from SPACEXTBL
where (landing__outcome like 'Success (drone ship)') and (payload_mass__kg_ between 4000 and 6000)
```

```
 * ibm_db_sa://lvr03390:***@6667d8e9-9d4d-4ccb-ba32-21da3bb5aafc.clogj3sd0tgtu0lqde00.databases.appdomain.cloud:30376/bludb
Done.
```

Out[62]:

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

Total Number of Successful and Failure Mission Outcomes

In [59]:
```
%%sql
select mission_outcome, count(*) as total_number from SPACEXTBL
group by mission_outcome
```

```
 * ibm_db_sa://lvr03390:***@6667d8e9-9d4d-4ccb-ba32-21da3bb5aafc.clogj3sd0tgtu0lqde00.databases.appdomain.cloud:30376/bludb
Done.
```

Out[59]:

mission_outcome	total_number
Failure (in flight)	1
Success	99
Success (payload status unclear)	1

Boosters Carried Maximum Payload

In [66]:

```
%%sql
select booster_version from SPACEXTBL
where payload_mass__kg_ = (select max(payload_mass__kg_) from SPACEXTBL)
```

```
 * ibm_db_sa://lvr03390:***@6667d8e9-9d4d-4ccb-ba32-21da3bb5aafc.clogj3sd0tgtu0lqde00.databases.appdomain.cloud:30376/bludb
Done.
```

Out[66]:

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

In [70]:
```
%%sql
select landing__outcome, booster_version, launch_site from SPACEXTBL
where DATE like '2015%' AND landing__outcome like 'Failure (drone ship)'
```

```
 * ibm_db_sa://lvr03390:***@6667d8e9-9d4d-4ccb-ba32-21da3bb5aafc.clogj3sd0tgtu0lqde00.databases.appdomain.cloud:30376/bludb
Done.
```

Out[70]:

landing__outcome	booster_version	launch_site
Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

Both two launched in 2015 were all failed.

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

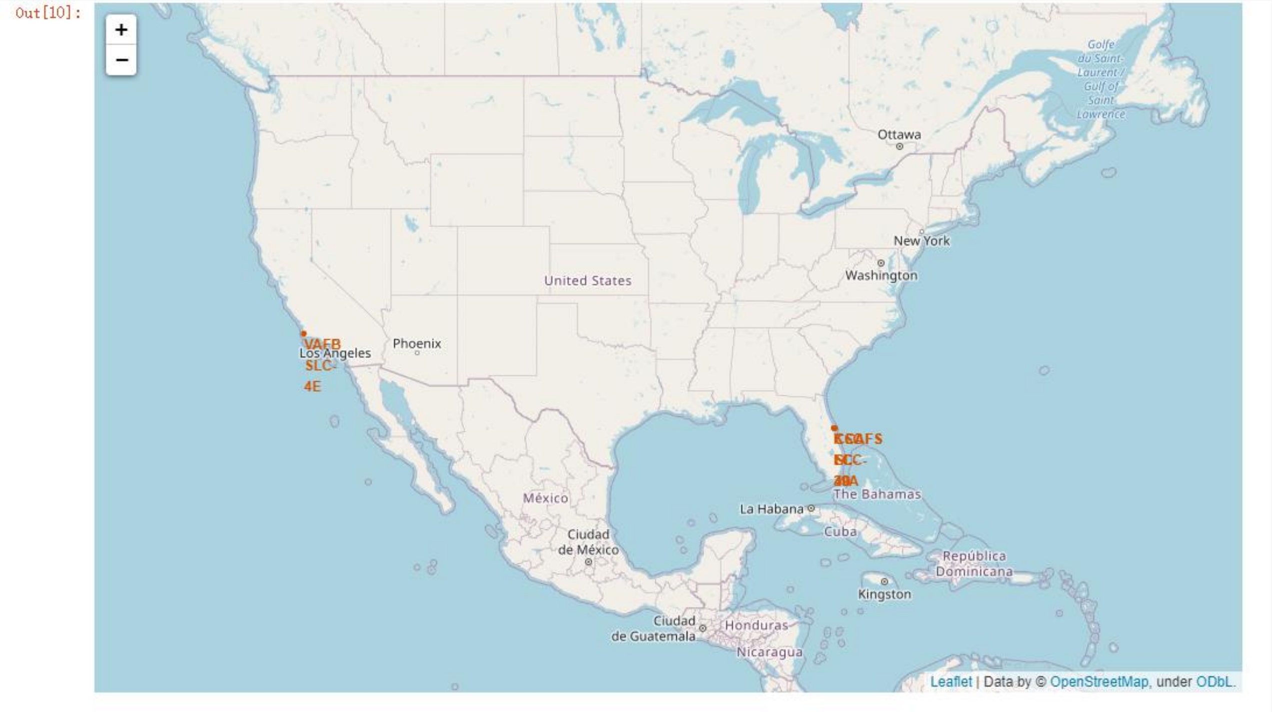
Out[71]:

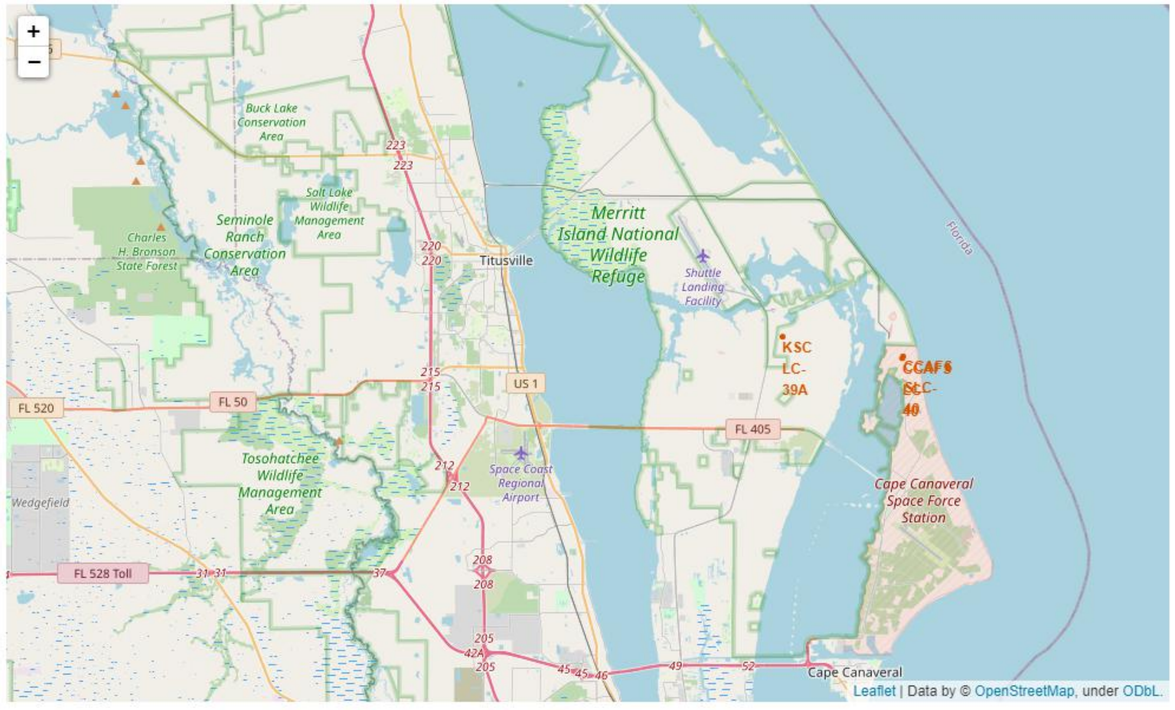
landing__outcome	COUNT
No attempt	10
Failure (drone ship)	5
Success (drone ship)	5
Controlled (ocean)	3
Success (ground pad)	3
Failure (parachute)	2
Uncontrolled (ocean)	2
Precluded (drone ship)	1

Section 3

Launch Sites Proximities Analysis

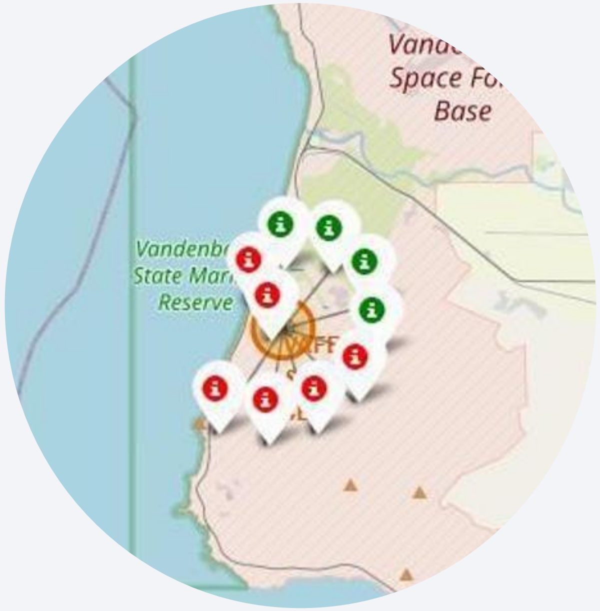
<Overview of launch sites>





There are three launch sites, two on the right hand side are close to each other on the US south eastern coastline.

<Launch sites with colored markers>



Green stands for success launch

Red stands for failed launch

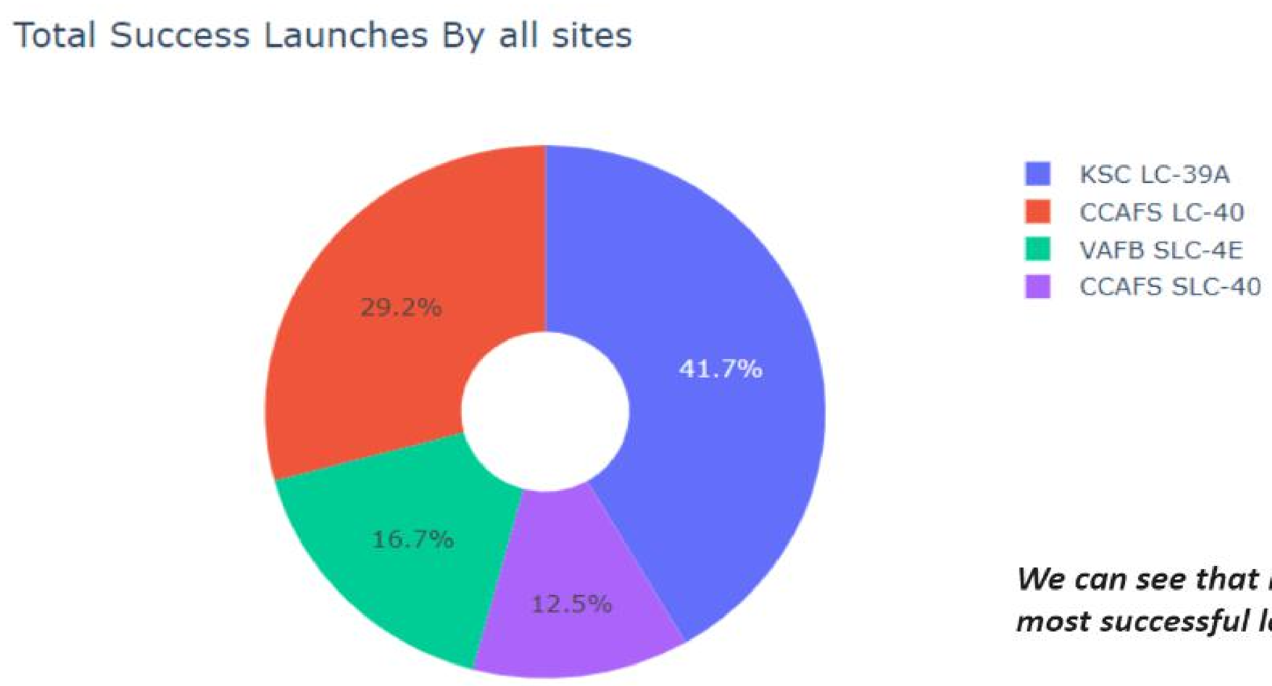
<Folium Map Screenshot 3>

The launch site is 900 meters to the nearest costaline.

Section 4

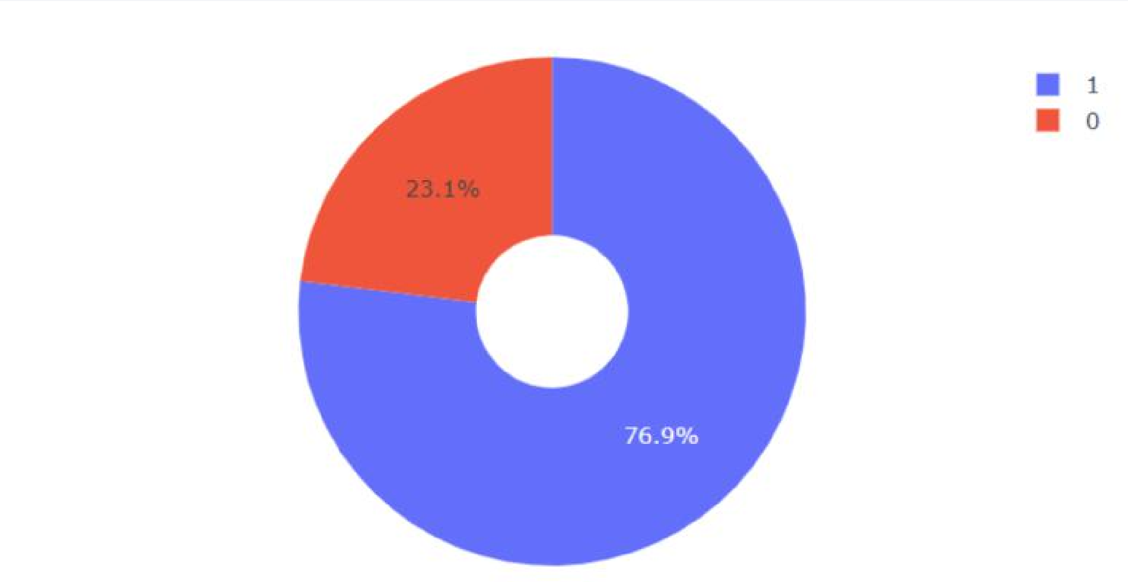
Build a Dashboard with Plotly Dash

<Total Success Launches By all sites>



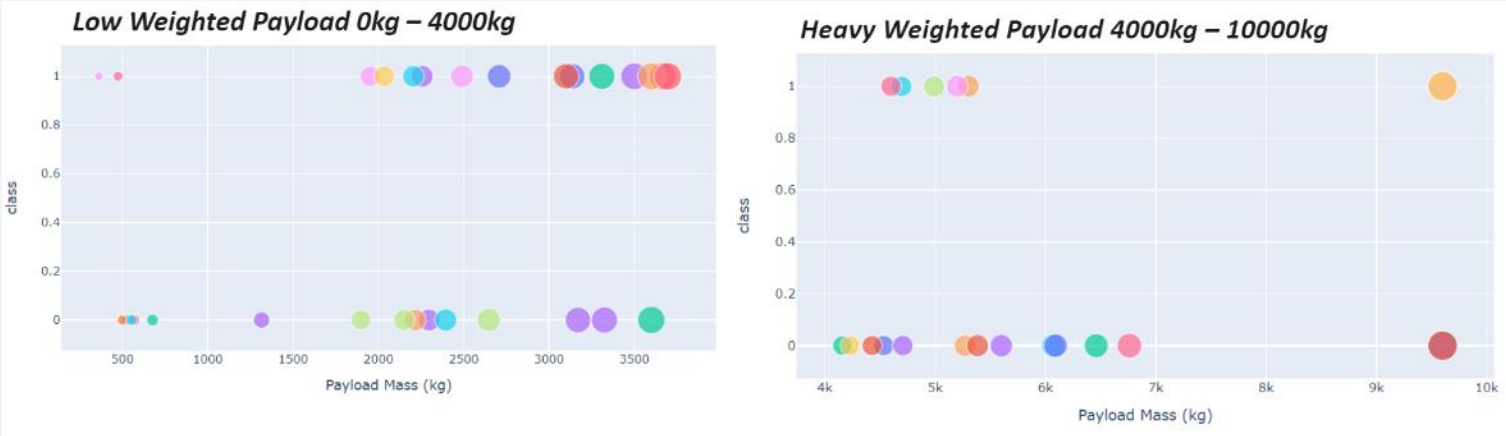
We can see that KSC LC-39A had the most successful launches from all the sites

<KSC LC39A launch site with the highest success rate>



KSC LC-39A achieved a 76.9% success rate while getting a 23.1% failure rate

<Launch outcomes between different payload mass>



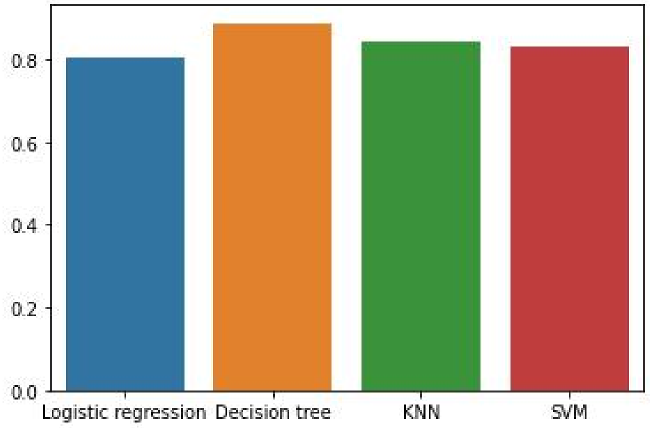
We can see the success rates for low weighted payloads is higher than the heavy weighted payloads

Section 5

Predictive Analysis (Classification)

Classification Accuracy

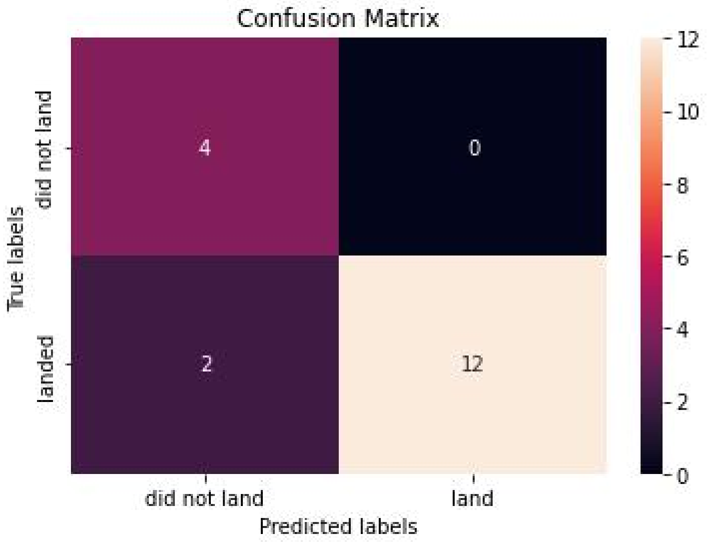
```
In  [22]: algorithm = {"Logistic regression": [0.8035714285714285],
                      "Decision tree":  [0.8875],
                     "KNN": [0.8446428571428569],
                     "SVM": [0.83214285714285]}
w
algorithm_com = pd.DataFrame(data=algorithm)

sns.barplot(data=algorithm_com)

Out[22]: <AxesSubplot:>
```



In this case, decision tree has the highest accuracy.

Confusion Matrix

```
In  [51]: yhat_tree = tree_cv.predict(X_test)
          plot_confusion_matrix(Y_test, yhat_tree)
```



According to the confusion matrix, Precision = 1, Recall = 0.857

Conclusions

- According to the charts, the larger the payload mass, the higher the success rate will be.
- GTO orbit has the lowest success rate of landing success
- The success rate is propotional to the year of development
- The decision is the best algorithm for the data set, however, the precision is “perfect” 1. Probably, the training data set is too small to predict. The recall is optimal meaning the algorithm has 85.7% correctness.

Appendix

- Github links:
  - https://github.com/soap945/DS_CAPSTONE_FINISHED/blob/9359601386c69dce965c89f06c1b84ebd52f6e2a/DS_CAPSTONE_FINISHED/1.0%20jupyter-labs-webscraping%20(1).ipynb
  - https://github.com/soap945/DS_CAPSTONE_FINISHED/blob/9359601386c69dce965c89f06c1b84ebd52f6e2a/DS_CAPSTONE_FINISHED/1.1%20labs-jupyter-spacex-Data%20wrangling%20(1).ipynb
  - https://github.com/soap945/DS_CAPSTONE_FINISHED/blob/9359601386c69dce965c89f06c1b84ebd52f6e2a/DS_CAPSTONE_FINISHED/2.1%20jupyter-labs-eda-dataviz%20(1).ipynb
  - https://github.com/soap945/DS_CAPSTONE_FINISHED/blob/9359601386c69dce965c89f06c1b84ebd52f6e2a/DS_CAPSTONE_FINISHED/2.0%20jupyter-labs-eda-sql-coursera%20(1).ipynb
  - https://github.com/soap945/DS_CAPSTONE_FINISHED/blob/9359601386c69dce965c89f06c1b84ebd52f6e2a/DS_CAPSTONE_FINISHED/3.0%20lab_jupyter_launch_site_location_dashboard_vis.ipynb
  - https://github.com/soap945/DS_CAPSTONE_FINISHED/blob/9359601386c69dce965c89f06c1b84ebd52f6e2a/DS_CAPSTONE_FINISHED/4.0%20SpaceX_Machine%20Learning%20Prediction_Part_5%20(1).ipynb

Thank you!