IBM Applied Data Science Capstone

Caroline Lim

24 Feb 2022

OUTLINE

- Executive Summary
- Introduction
- Methodology
- Results
  - Visualization – Charts
  - Dashboard
- Discussion
  - Findings & Implications
- Conclusion
- Appendix

EXECUTIVE SUMMARY

Goal – To find the best way to minimize the cost of space launch

Desired outcome – If the Stage One, which is the large and expensive component of the rocket can be recovered and reused, this would significantly reduce the cost.

We will use machine learning models to predict whether and not a SpaceX stage one recovery will be successful.

With this knowledge, we can identify the variables that affect successful recoveries.

INTRODUCTION

Project Background and Context

- SpaceX advertises that Falcon 9 rocket launches with a cost of $62M dollars whilst other providers costs more than $165M.
- If we can predict if the Falcon 9 first stage can land successfully, then this information can be used to bid against SpaceX for a rocket launch.

Problem Statements

- What are the factors that causes the rocket to land successfully?
- What are the relationships amongst the variables that affects the outcome?

METHODOLOGY

- Data collection methodology:
- Data was collected using SpaceX API and web scraping from Wikipedia.
- Perform data wrangling
- One-hot encoding was applied to categorical features
- 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

- Sources of data: SpaceX Rest API and Wikipedia
- Steps :
  - Get data from downloading data from Space Rest API and Webscraping from Wikipedia
  - Make a dataframe from data
  - Filter dataframe
  - Export to CSV flat file

DATA COLLECTION

- Data Collection via SpaceX API
  - Getting Response from API
  - Converting Response to a JSON file
  - Appy custom functions to clean data
  - Align list to dictionary
  - Create Dataframe
  - Filter dataframe and export to flat file
- Data Collection via Web Scraping
  - Getting Response from HTML
  - Creating BeautifulSoup Obj
  - Finding tables
  - Getting column names
  - Create dictionary and append data to keys
  - Convert dictionary to dataframe
  - Convert dataframe to CSV

DATA WRANGLING

- To clean messy data and organize into structured dataset for analysis
- Objective: To convert data into “1” or “0”. “1” for booster that successfully landed and “0” means booster that unsuccessfully landed.
- Steps:
  - Load Data
  - Create Dataframe
  - Clean data
  - Simplify to Boolean values
  - Export to flat file
  - Process in Detail:
  - Calculate # of launches at each site
  - Calculate # of orbits
  - Calculate # of mission outcomes per orbit type
  - Create landing outcome label from Outcome column
  - Export dataset as CSV

EDA and interactive visual analytics methodology

- Exploratory data analysis (EDA) is used to analyze and investigate data sets and summarize their main characteristics using data visualization methods. It is used to discover patterns, spot anomalies, test a hypothesis, or check assumptions.
- Steps:
  - Load Data
  - Make dataframe
  - Create Visualization
  - Collect Insights
- Output:
  - Scatter Graphs
  - Bar Graph
  - Line Graph

EDA WITH SQL RESULTS

- SQL stands for Structured Query Language which is basically a language used by databases. This language allows to handle the information using tables and shows a language to query these tables and other objects related (views, functions, procedures, etc.). Most of the databases like SQL Server, Oracle, PostgreSQL, MySQL, MariaDB handle this language (with some extensions and variations) to handle the data. For this project, IBM Db2 for Cloud was used.
- Steps:
  - Displayed names of unique launch sites
  - Displayed 5 records where launch sites begin with string 'CCA'
  - Displayed total payload mass carried by boosters by NSAS (CRS)
  - Displayed average payload mass carried by booster version F9v1.1
  - List of dates of successful landing outcome in drone ship
  - List of names of boosters which had success in ground pad and payload mass > 4000 but less than 6000
  - List of total number of successful and failure mission outcomes
  - List of names of booster versions which have carried the maximum payload mass
  - List of failed landing outcomes. Name of drone ship, booster versions and launch site names for the year of 2015.
  - Ranking of landing outcomes between Jun 2010 and Mar 2010 in descending order.

INTERACTIVE MAP WITH FOLIUM RESULTS

- Folium is a powerful Python library that helps you create several types of Leaflet maps. By default, Folium creates a map in a separate HTML file. Since Folium results are interactive, this library is very useful for dashboard building. You can also create inline Jupyter maps in Folium. Folium builds on the data wrangling strengths of the Python ecosystem and the mapping strengths of the Leaflet.js library. Using Folium, you can manipulate your data in Python, then visualize it in a Leaflet map.
- Steps:
    - Use latitude and longitude coordinates to add Circle Marker to mark and label each launch site.
    - Use different color markers to mark successful sites vs failure sites.
    - Map objects

INTERACTIVE MAP WITH FOLIUM RESULTS

Map object	Code	Result
Map Marker	folium.Marker(	Map object to make a mark on map
Icon Marker	folium.Icon(	Create an icon on map
Circle Marker	folium. Circle(	Create a circle where marker is placed
PolyLine	folium. PolyLine(	Create a line between points
Marker Cluster Object	MarkerCluster(	To simplify map containing many markers having the same coordinates
AntPath	folium. Plugins.AntPath(	Create an animated line between points

PLOTLY DASH DASHBOARD RESULTS

Definition:

- Dash is a python framework created by plotly for creating interactive web applications. It is written on the top of Flask, Plotly.js and React.js. It is open source and the application build using this framework are viewed on the web browser.

Pie Chart shows the total success for all sites or by certain launch site.

Scatter Graph shows the correlation between Payload and Success for all sites or by certain launch site.

PREDICTIVE ANALYSIS METHODOLOGY

- Steps:
- Building Model
  - Load our feature engineered data into dataframe
  - Transform it into NumPy arrays
  - Standardize and transform data
  - Split data into training and test data sets
  - Check how many test samples has been created
  - List down machine learning algorithms we want to use
  - Set out parameters and algorithms to GridSearchCV
  - Fit our datasets into the GridSearchCV objects and train our model
- Evaluating Model
  - Check accuracy for each model
  - Get best hyperparameters for each type of algorithms
  - Plot Confusion Matrix
- Finding Best Performing Classification Model
  - The model with best accuracy score wins the best performing model
- Derive on the Best Model

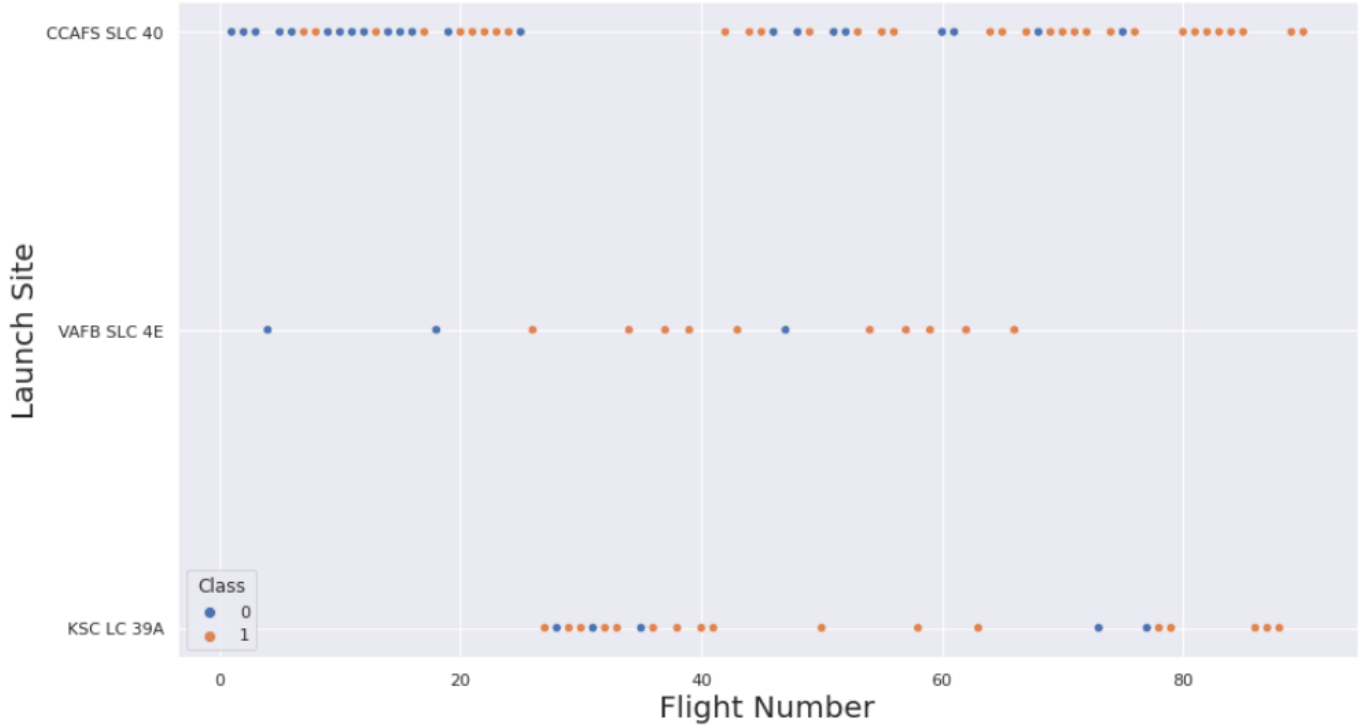
DASHBOARD

<The permanent link of the read-only view of the Cognos dashboard goes here.>

EDA WITH VISUALIZATION

Flight Number vs Launch Site

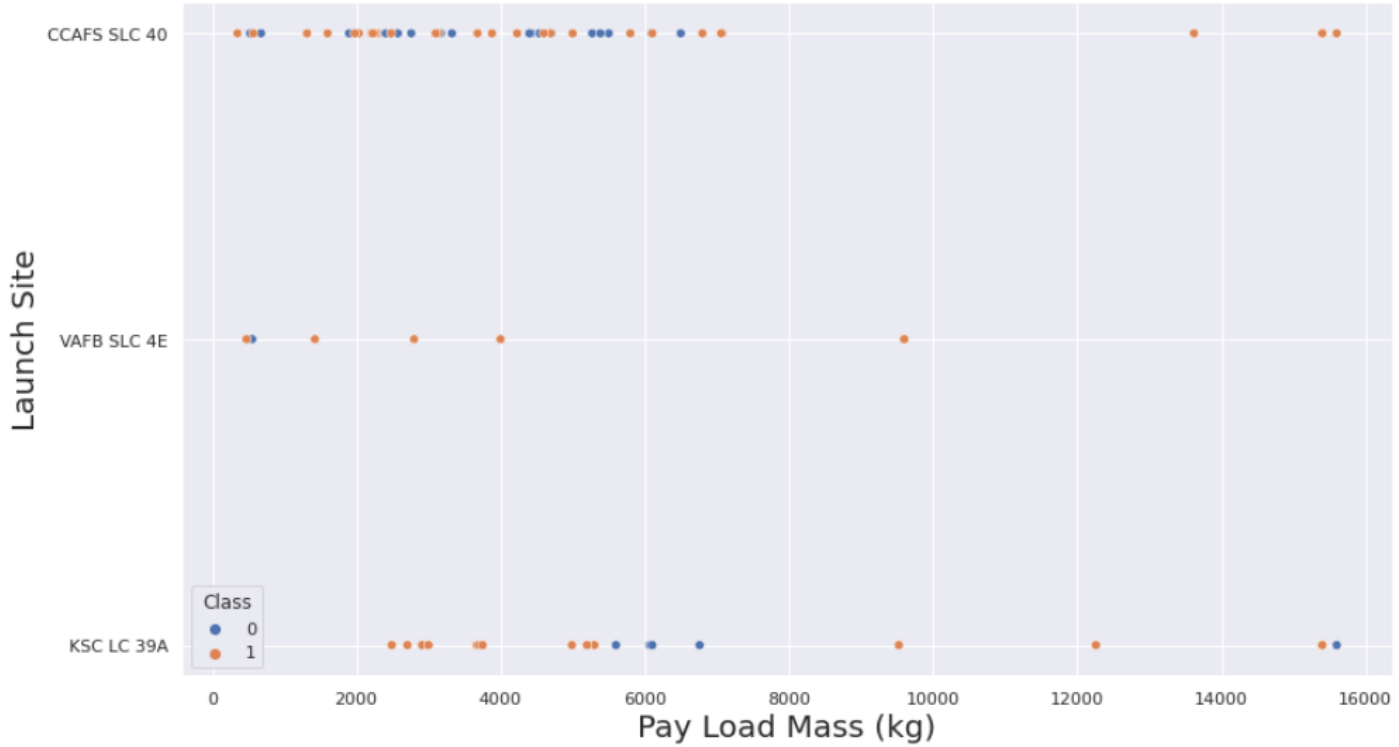
- The data suggests the higher the flight number, the success rate of launch is higher.



EDA WITH VISUALIZATION

Payload vs Launch Site

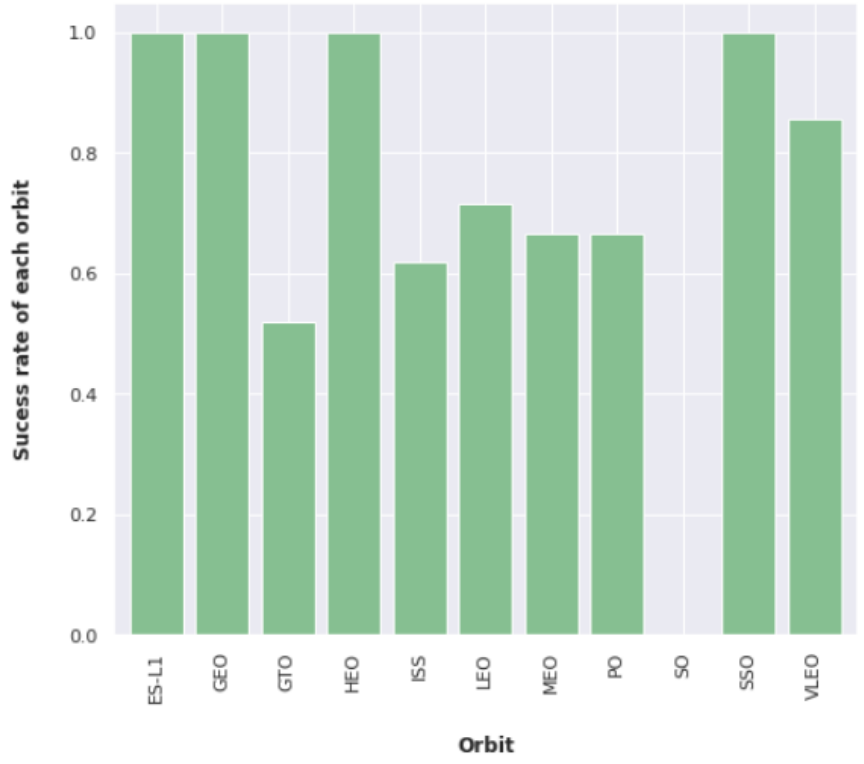
- Data suggest the higher the pay load mass, the higher the rate of successful launch.



EDA WITH VISUALIZATION

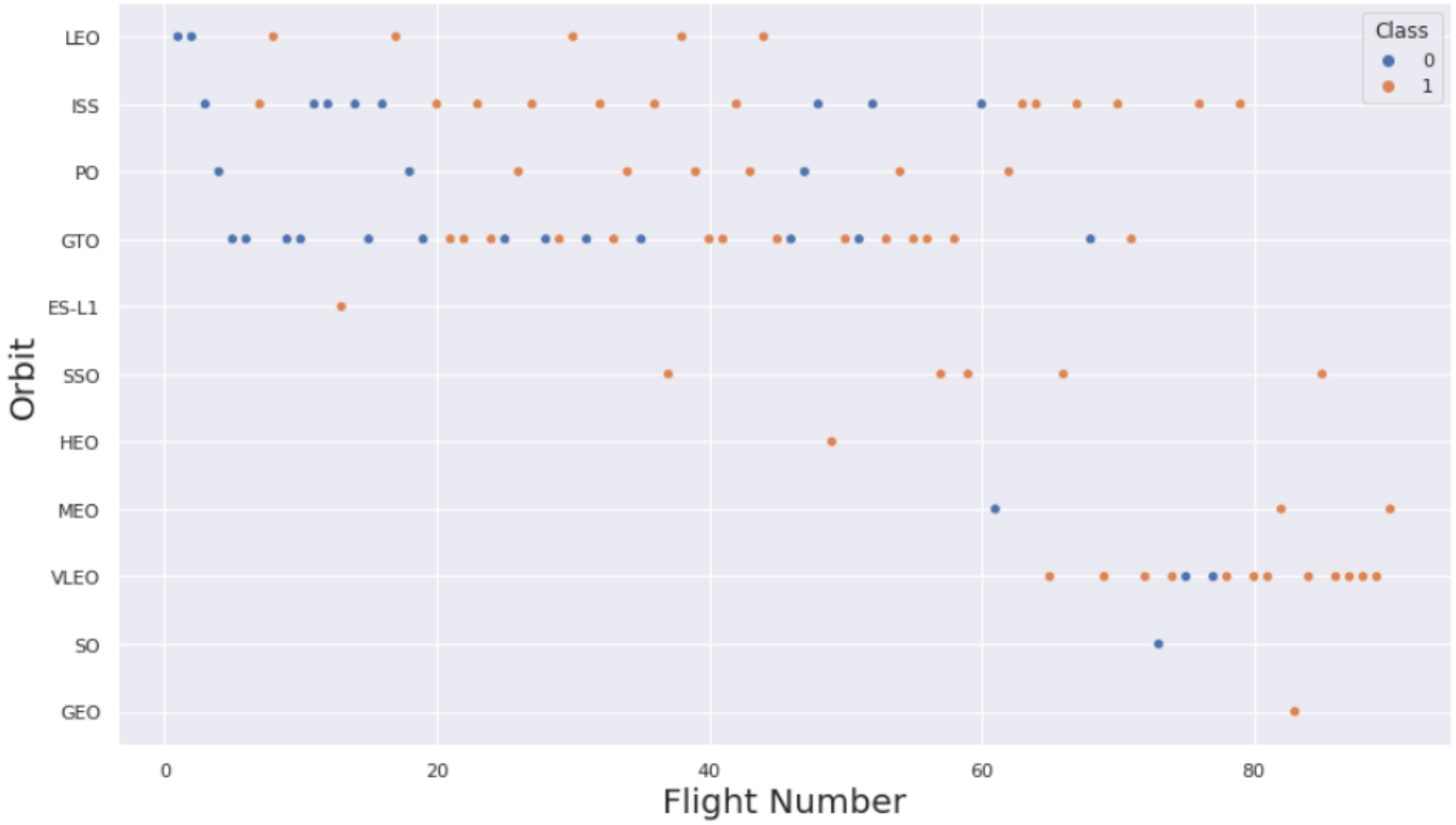
Success Rate vs Orbit Type

- ES-L1,GEO,HEO,SSO have the highest Success Rate.



EDA WITH VISUALIZATION

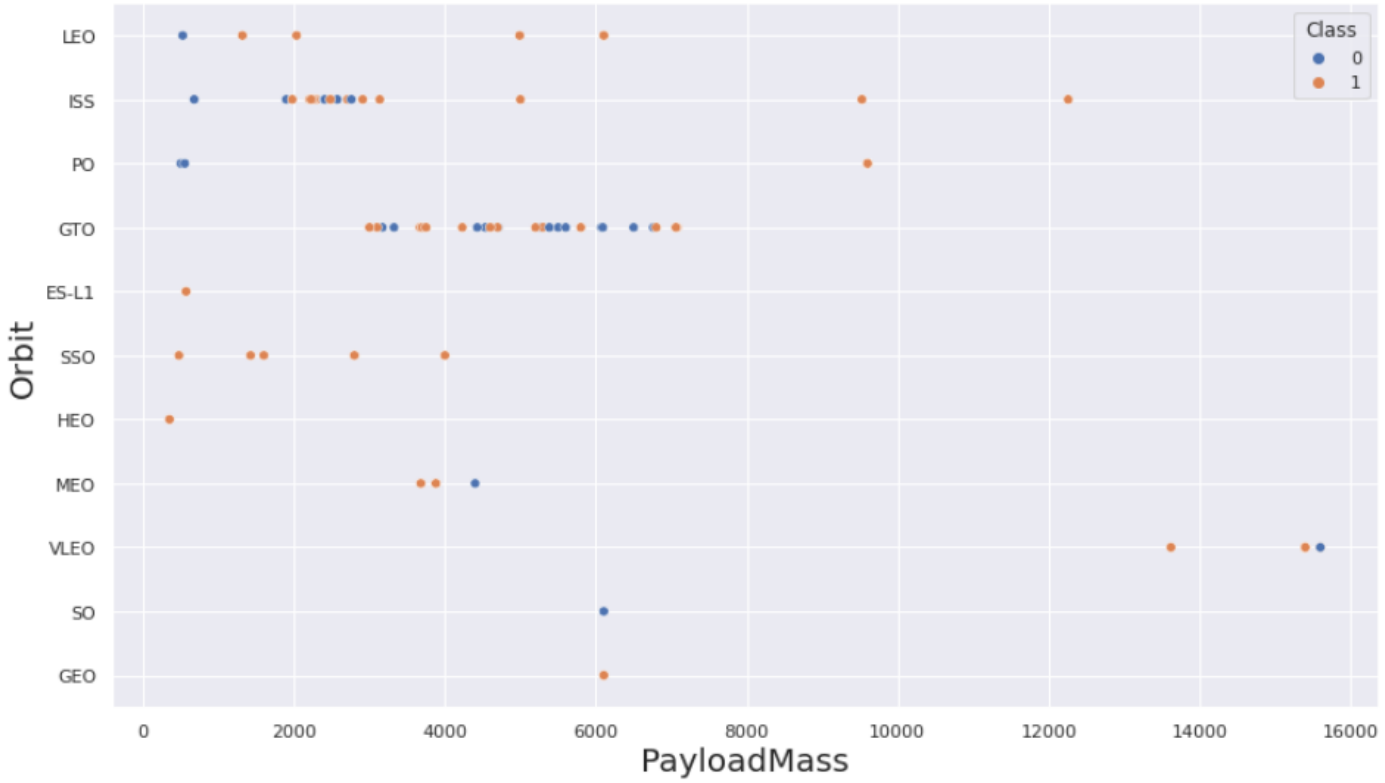
- Flight Number vs Orbit Type



EDA WITH VISUALIZATION

Payload vs Orbit Type

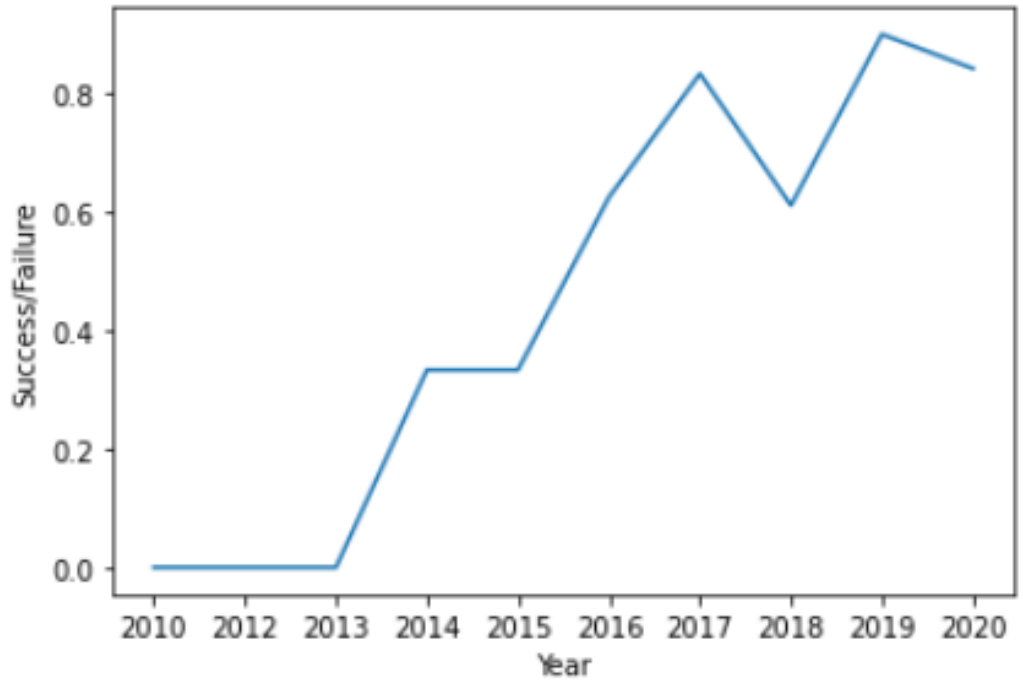
- LEO orbit the Success appears related to the number of flights; on the other hand, there seems to be no relationship between flight number when in GTO orbit.
- With heavy payloads the successful landing or positive landing rate are more for Polar,LEO and ISS.
- However for GTO we cannot distinguish this well as both positive landing rate and negative landing(unsuccessful mission) are both there here.



EDA WITH VISUALIZATION

Launch Success Yearly Trend

- the sucess rate since 2013 kept increasing till 2020



EDA with SQL

- All Launch Site Names

```
# Apply value_counts() on column LaunchSite
df.LaunchSite.value_counts()
```

```
CCAFS SLC 40    55
KSC LC 39A      22
VAFB SLC 4E     13
Name: LaunchSite, dtype: int64
```

- Launch Site Names begin with “CCA”

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

```
%sql SELECT * FROM SPACEXTBL WHERE LAUNCH_SITE LIKE 'CCA%' LIMIT 5
```

```
 * ibm_db_sa://rwk40946:***@21fecfd8-47b7-4937-840d-d791d0218660.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31864/bludb
Done.
```

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

EDA with SQL

- Total Payload Mass

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

```
%sql SELECT SUM(PAYLOAD_MASS__KG_) FROM SPACEXTBL WHERE CUSTOMER='NASA (CRS)'
```

```
 * ibm_db_sa://rwk40946:***@21fecfd8-47b7-4937-840d-d791d0218660.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31864/bludb
Done.
```

1
45596

- Average Payload Mass by F9 v1.1

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

```
%sql SELECT AVG(PAYLOAD_MASS__KG_) FROM SPACEXTBL WHERE BOOSTER_VERSION='F9 v1.1'
```

```
 * ibm_db_sa://rwk40946:***@21fecfd8-47b7-4937-840d-d791d0218660.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31864/bludb
Done.
```

1
2928

EDA with SQL

- First Successful Ground Landing Date

```
%sql SELECT min(DATE) FROM SPACEXTBL WHERE LANDING__OUTCOME='Success (ground pad)'
```

```
 * ibm_db_sa://rwk40946:***@21fecfd8-47b7-4937-840d-d791d0218660.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31864/bludb
Done.
```

1
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

```
%sql SELECT BOOSTER_VERSION FROM SPACEXTBL WHERE PAYLOAD_MASS__KG_ between 4000 and 6000 AND LANDING__OUTCOME='Success (drone ship)'
```

```
 * ibm_db_sa://rwk40946:***@21fecfd8-47b7-4937-840d-d791d0218660.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31864/bludb
Done.
```

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

EDA with SQL

- Total number of Successful and Failure Mission Outcomes

List the total number of successful and failure mission outcomes

```
%sql SELECT COUNT(*) FROM SPACEXTBL WHERE MISSION_OUTCOME LIKE '%Success%' OR MISSION_OUTCOME LIKE '%Failure%'
```

```
 * ibm_db_sa://rwk40946:***@21fecfd8-47b7-4937-840d-d791d0218660.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31864/bludb
Done.
```

1
101

EDA with SQL

- Boosters carried Maximum Payload

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)
```

```
 * ibm_db_sa://rwk40946:***@21fecfd8-47b7-4937-840d-d791d0218660.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31864/bludb
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

EDA with SQL

- 2015 Launch Records
- Rank Landing Outcomes between June 2010 and Mar 2017
- Rank Success Count between Jun 2010 and Mar 2017

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

```
%sql SELECT TO_CHAR(TO_DATE(MONTH("DATE"), 'MM'), 'MONTH') AS MONTH_NAME, \
    LANDING__OUTCOME AS LANDING__OUTCOME, \
    BOOSTER_VERSION AS BOOSTER_VERSION, \
    LAUNCH_SITE AS LAUNCH_SITE \
    FROM SPACEXTBL WHERE LANDING__OUTCOME = 'Failure (drone ship)' AND "DATE" LIKE '%2015%'
```

```
 * ibm_db_sa://rwk40946:***@21fecfd8-47b7-4937-840d-d791d0218660.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31864/bludb
Done.
```

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

EDA with SQL

- Rank Landing Outcomes between June 2010 and Mar 2017
- Rank Success Count between Jun 2010 and Mar 2017

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 "DATE", COUNT(LANDING__OUTCOME) as COUNT FROM SPACEXTBL \
    WHERE "DATE" BETWEEN '2010-06-04' and '2017-03-20' AND LANDING__OUTCOME LIKE '%Success%' \
    GROUP BY "DATE" \
    ORDER BY COUNT(LANDING__OUTCOME) DESC
```

```
 * ibm_db_sa://rwk40946:***@21fecfd8-47b7-4937-840d-d791d0218660.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31864/bludb
Done.
```

DATE	COUNT
2015-12-22	1
2016-04-08	1
2016-05-06	1
2016-05-27	1
2016-07-18	1
2016-08-14	1
2017-01-14	1
2017-02-19	1

Interactive with Folium

- All Launch Sites on Folium Map: Near coastline of Florida and California

	Launch Site	Lat	Long
0	CCAFS LC-40	28.562302	-80.577356
1	CCAFS SLC-40	28.563197	-80.576820
2	KSC LC-39A	28.573255	-80.646895
3	VAFB SLC-4E	34.632834	-120.610746

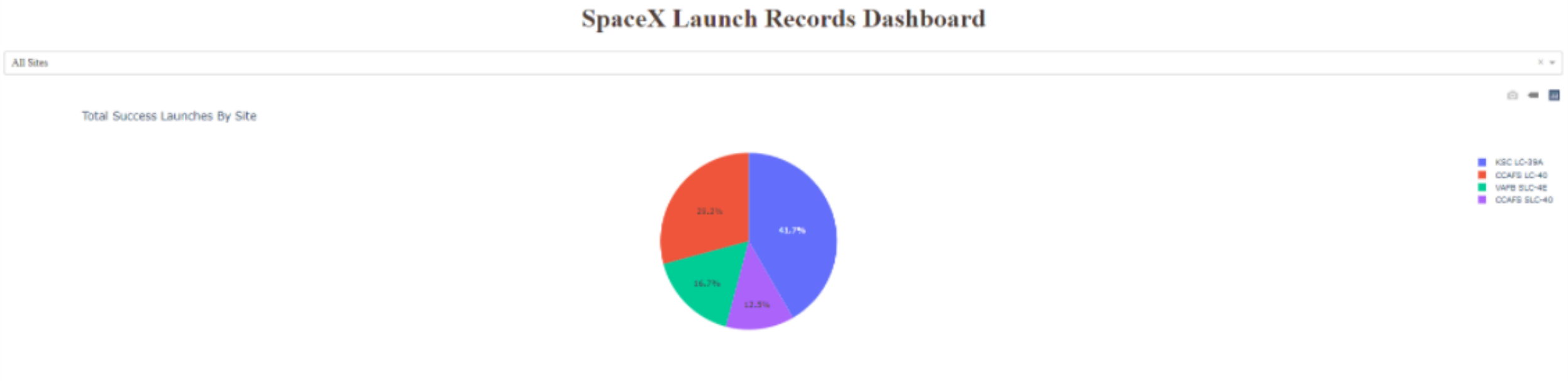
Interactive with Folium

- Launch Site Distances from Equator & Railways: Not near
- Launch Site Distances from Coastlines & Cities: Near to coastline
- Launch Site Distances from Highways: Not near

```
distance_highway = 0.5834695366934144  km
distance_railroad = 1.2845344718142522  km
distance_city = 51.43416999517233  km
```

Build a Dashboard with Plotly Dash

- Launch Success Count for all sites
- Launch Site with highest launch success ratio: KSC-LC-39A



Build a Dashboard with Plotly Dash

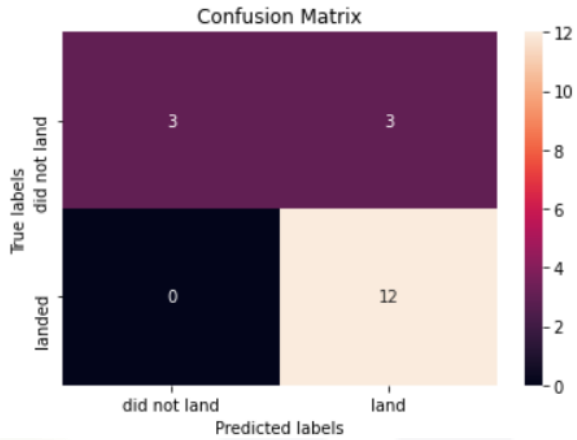
- Payload vs Launch Outcome Scatter Plots for all sites
  - 2000Kg-10000Kg payload range has the highest launch success rate
  - 0-1000Kg has the lowest launch success rate
  - FT has the highest launch success rate

Predictive analysis (Classification)

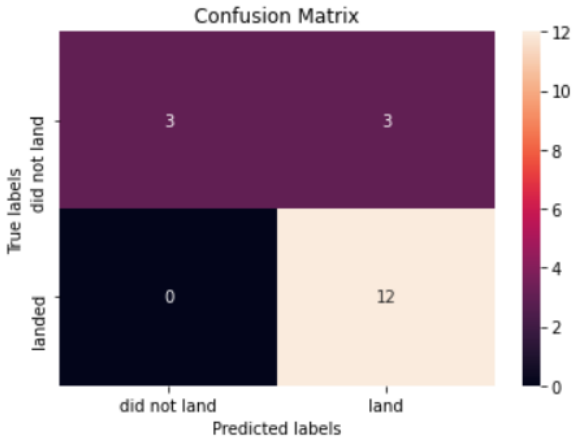
- ## Confusion Matrix

- We see that logistic regression can distinguish between the different classes.
- The major problem observed is false positives.
- Accuracy: (TP+TN)/Total; =12+3/18 =0.83333
- Misclassification Rate: (FP+FN)/Total =3+0/18 =0.166
- True Positive Rate: TP/Actual Yes =12/12 =1
- False Positive Rate: FP/Actual No =3/6 =0.5
- True Negative Rate: TN/Actual No =3/6 =0.5
- Precision: TP/Predicted Yes = 12/15=0.8
- Prevalence: Actual Yes/Total =12/18 =0.6667

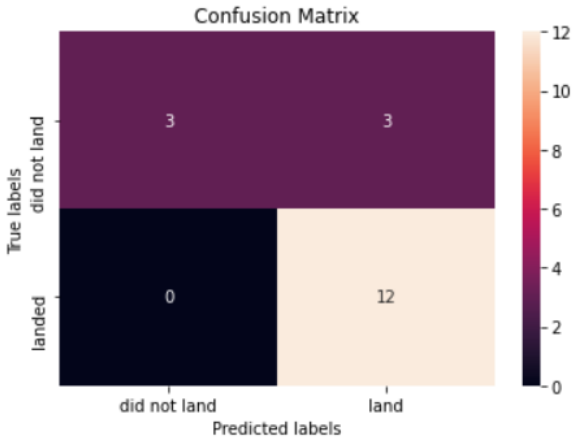
Logistic Regression Model



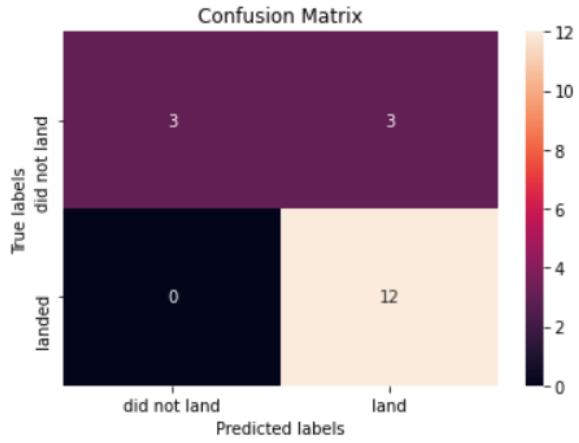
SVM Model



Decision Tree Model



KNN Model



Predictive analysis (Classification)

- Classification Accuracy

```
Accuracy for Logistics Regression method: 0.8333333333333334
Accuracy for Support Vector Machine method: 0.8333333333333334
Accuracy for Decision tree method: 0.8333333333333334
Accuracy for K nearsdt neighbors method: 0.8333333333333334
```

No clear winner as all 4 methods yielded 83.33% accuracy rate

DISCUSSION

CONCLUSION

- We have identified that payload and payload mass are primary variables that affect successful recoveries.
- The higher the flight number and payload mass, the higher the success rate of launch.
- The success rate differs on different orbit types. ES-L1,GEO,HEO,SSO have the highest Success Rate