IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Marc Martinho

13/10/2021

Outline

Executive Summary

Introduction

Methodology

Results

Conclusion

Appendix

Executive Summary

Space flight is the next exploration area for humanity and increasing the commercial viability of space travel is key in the progress of space exploration. In order to achieve a more commercially viable space travel the cost of space. SpaceX has at present been a pioneer of reducing the cost to space flight by reusing the stage 1 booster rocket.

Collecting data from past SpaceX flights through an API and web scraping in order to train a classification model to start to predict the reusability of the booster which improves efficiencies and reducing cost.

The data is called from the SpaceX API with multiple endpoints and web scrape from the relevant wiki page allows for a full picture of past flights. This data is cleaned and processed from nulls and setting a numerical target variable. With the combination of queries and visualizations methods the data is explored and analyzed.

The data is then presented through an interactive dashboard allowing for further data exploration and the end user to view and understand the data.

Analyzing the data shows that the SpaceX booster and flights improved over time with more expansive and heavy payloads with a higher success rate

Multiple classification methods are then trained and reviewed. The most accurate classification method is the decision tree showing that a trained classification model is viable to predict and classify a flight outcome to reduce the cost of each space flight making space flight more commercially viability of space travel.

Introduction

Commercial Space travel is rapidly developing with companies like SpaceX, Virgin Galactic and Blue Origin. However, it is considered that SpaceX is leading with missions to the International Space Station and the Starlink network. One of main reasons for this success is SpaceX has been able to reduce the cost of each flight by reusing the first stage Rocket, the Flacon 9 rocket.

As the first stage is the largest and does most of the work it is the most expensive part of the rocket and the whole trip. Thus, if we are able to recover and reuse the stage one rocket then the cost of flights can be reduced and thus space travel more commercially viable.

There are multiple factors that are used to determine if a stage one will be reused thus the aim will be to build machine model that can predict if the stage one will be recovered ad thus reused.

Section 1

Methodology

Methodology

Executive Summary

Data collection methodology:

- The data was collected from two sources. A REST API and web scraping. Both these are public data sources.

Perform data wrangling

- The data was reviewed for nulls, data types and the target variable clearly defined.

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

The data was collected from two main public sources:

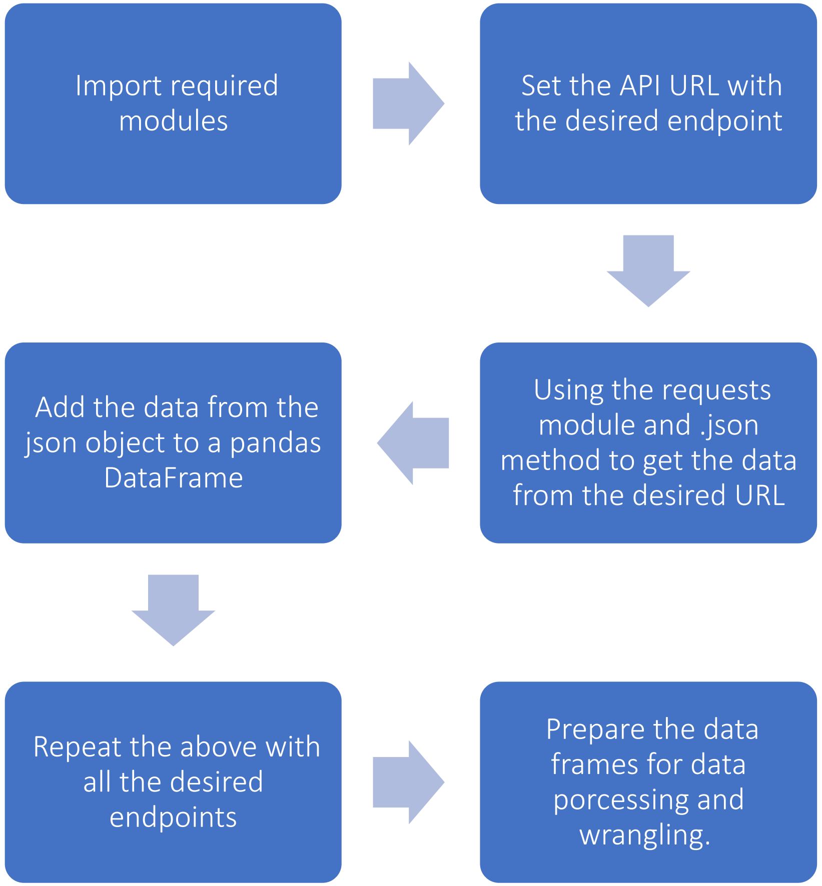
- The SpaceX REST API. This API gives the key launch data; rocket information, payloads, launch specifications, landing details and landing outcome. Multiple API end points will be used to gather all the required launch data.
- Web scaping is used as a secondary data source to retrieve launch data from the relevant wiki page.

Data Collection – SpaceX API

- SpaceX REST API was used with the following endpoints:
  - /rockets
  - /launchpads
  - /payloads
  - /cores
  - /past
- The requests module is used to get the data into a json

- The complete notebook can be found at:

https://github.com/marc-martini/Coursera_ML_Capstone/blob/4c6dc4149673fdbe56deeb61216fd47419cea2ef/Data%20Collection%20Lab.ipynb

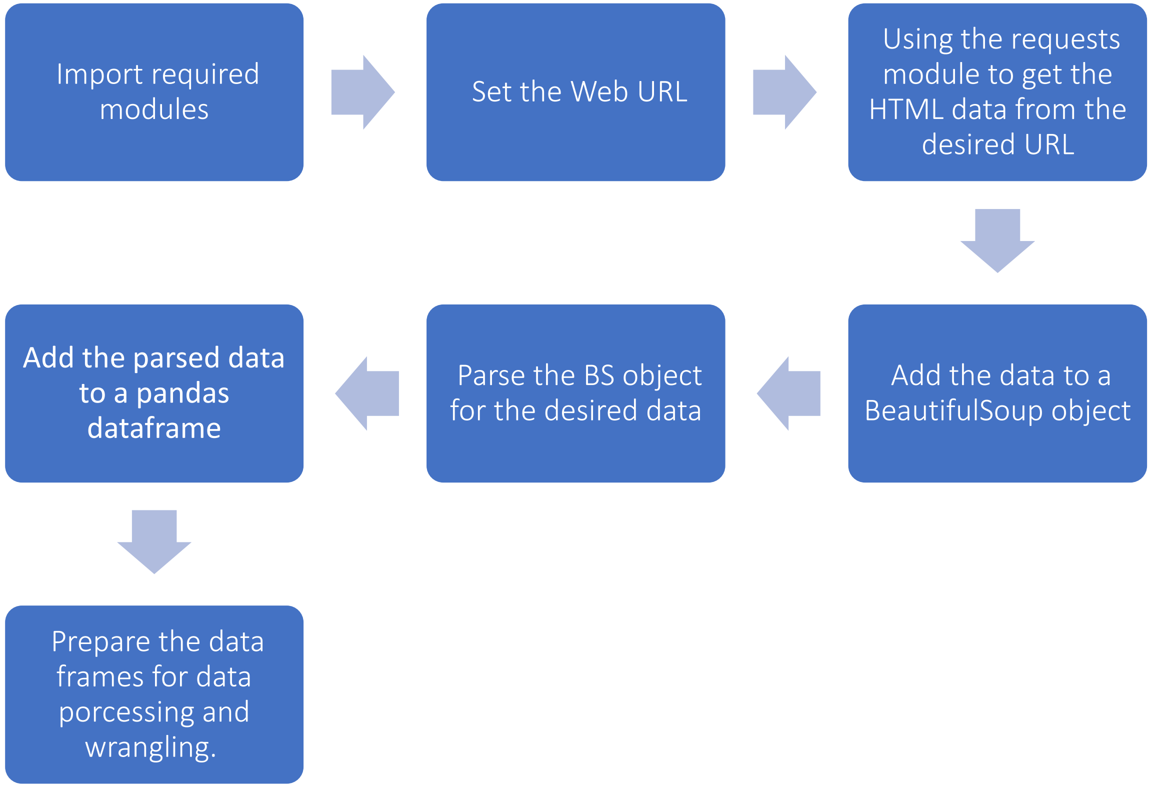


Data Collection - Scraping

- Additional Flacon 9 launch data is contained on the relent wiki page. Web scraping will be used to retrieve this data.
  - Python BeautifulSoup is used the scrape data from HTML tables.
  - This data is then parsed into pandas data structures for further processing.

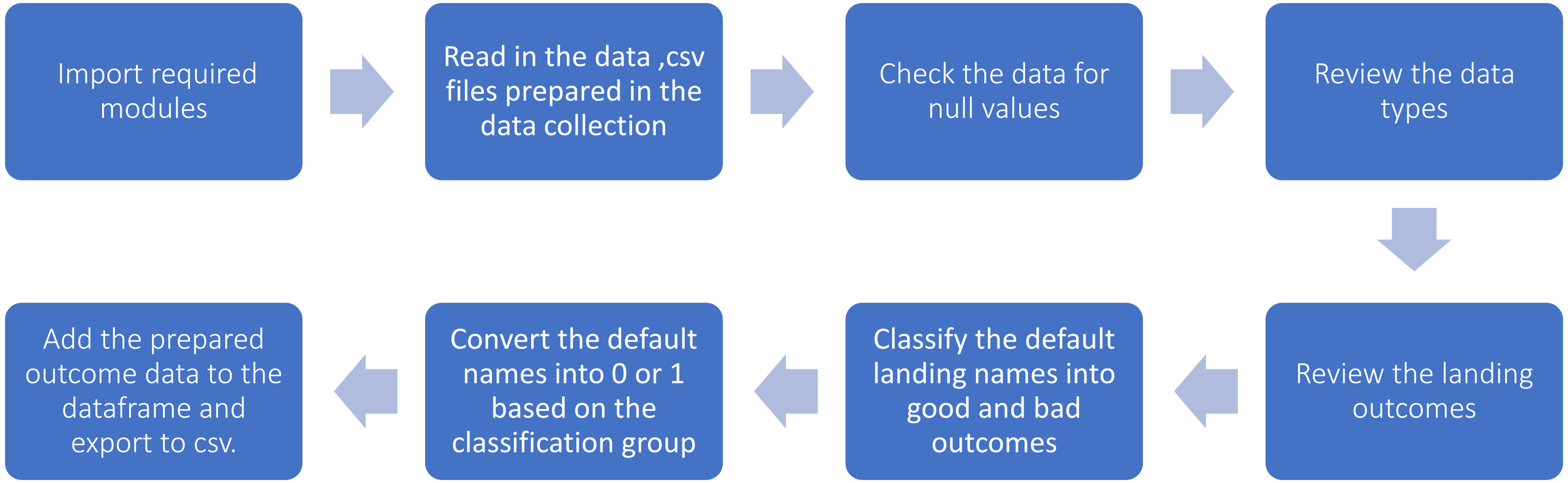
- The complete notebook can be found at:

https://github.com/marc-martini/Coursera_ML_Capstone/blob/633fce3dd1464552d5e8a57e844d6c09ab450c06/Data%20Collection%20with%20Web%20Scraping.ipynb



Data Wrangling

- The data were organized into pandas data frames with the required column names. The data is then reviewed to ensure the values are as per expected and are ready for input into a machine algorithm.
- Also, the target variable needs to be processed and put into a numerical value for use in the machine algorithm. The data wrangling process is as per the below:



- The complete notebook can be found at:

https://github.com/marc-martini/Coursera_ML_Capstone/blob/effb72c4d5ce2410bc1c5d8a9bfdd42fa8c42291/EDA%20lab.ipynb

EDA with Data Visualization

- The first step to explore the data is to use visualizations. The following visualizations were used:
  - Scatter plot of Flight Number vs. Launch Site
    - To view which sites were used for which flight and what the outcome was.
  - Scatter plot of Payload vs. Launch Site
    - View the success of the payload and how this is affected by launch site
  - Bar chart for the success rate of each orbit type
    - Compare the success rate of each orbit level
  - Scatter point of Flight number vs. Orbit type
    - View the flight numbers for each orbit and the outcome
  - Scatter point of payload vs. orbit type
    - Plot to view effect of orbit and payload on success
  - Line chart of average yearly success rate
    - Review the success rate over time

- The complete notebook can be found at:
- https://github.com/marc-martini/Coursera_ML_Capstone/blob/d475670620e8fa72b16fda27cce5da346eb79001/EDA%20with%20Visualization%20lab.ipynb

EDA with SQL

- The next step is to review key aspects of the data with SQL queries:
  - The names of the unique launch sites
  - 5 records where launch sites begin with `CCA`
  - Total payload carried by boosters from NASA
  - Average payload mass carried by booster version F9 v1.1
  - The dates of the first successful landing outcome on ground pad
  - The names of boosters which have successfully landed on drone ship and had payload mass greater than 4000 but less than 6000
  - The total number of successful and failure mission outcomes
  - The names of the booster which have carried the maximum payload mass
  - Failed landing_outcomes in drone ship, their booster versions, and launch site names for in year 2015
  - 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

- The complete notebook can be found at:
  - https://github.com/marc-martini/Coursera_ML_Capstone/blob/3e497ae0e5ccf44152f090481a4b183a0205f719/EDA%20with%20SQL%20lab.ipynb

Build an Interactive Map with Folium

- The Folium module is used to map details on a map.
  - Circle objects are used to add a circle on the map to show a location with coordinates
  - Maker object adds a text on to the map on specific coordinates
  - Marker cluster is used to add a group of markers
  - Icon object adds a custom icon onto the mad to show information
  - Mouse position add a text notification about the mouse location on the map
  - Polyline shows a line between two points on the map between two coordinates

- The complete notebook can be found at:
  - https://github.com/marc-martini/Coursera_ML_Capstone/blob/133a681f64e4babdfa9dfb90d1570f90312c5719/Interactive%20Visual%20Analytics%20with%20Folium%20lab.ipynb

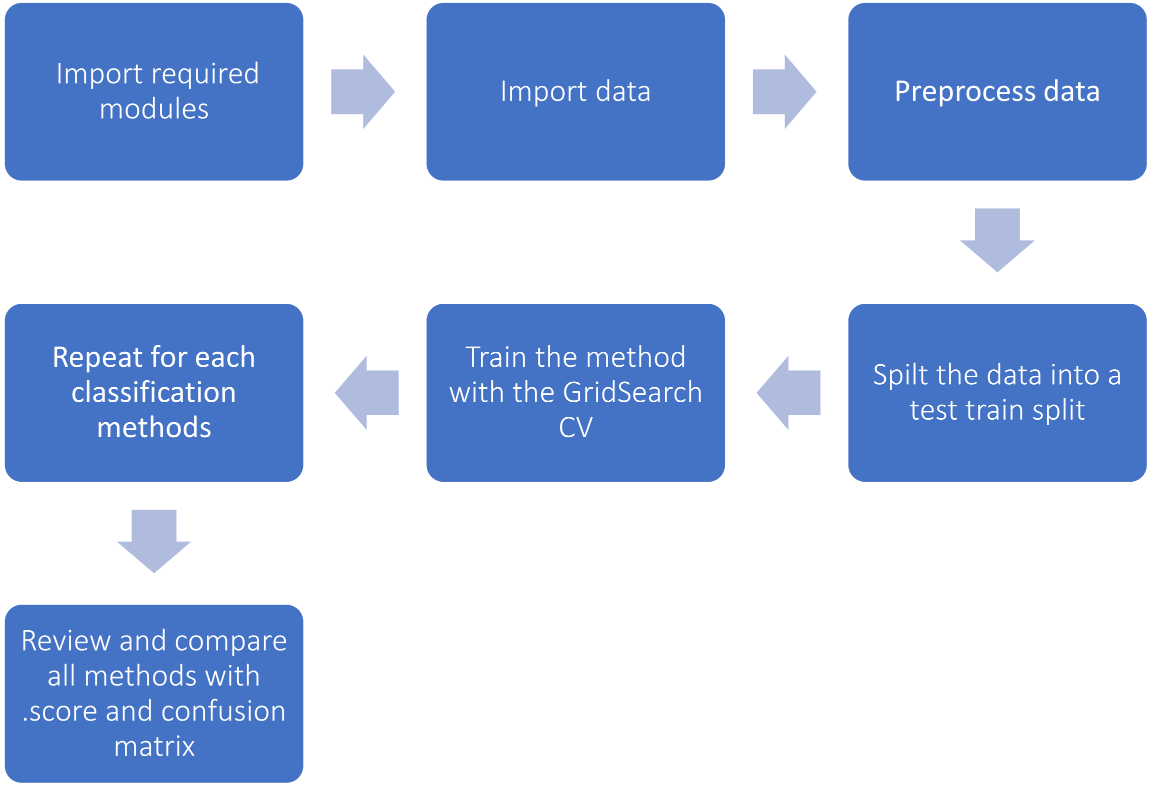
Build a Dashboard with Plotly Dash

- The use of a dashboard allows for the display of data where the user can manipulate it to show more details.
- The dashboard has the following plots:
  - A pie chart that shows the success spilt for all sites
    - This also has the option to choose a site and see the success/failure split for that site.
    - We can see the most successful sites and do a deep dive into each site.
  - A Scatter plot to payload mass for each booster type split by success class
    - This can be shown for all the sites in total or deep dive for each individual site.
- The plots can be manipulated by:
  - View site by site or all the sites.
  - Change the range of payload mass for the scatter plot

- The complete app code can be found at:
  - https://github.com/marc-martini/Coursera_ML_Capstone/blob/133a681f64e4babdfa9dfb90d1570f90312c5719/spacex_dash_app.py

Predictive Analysis (Classification)

- The classification models were built using Scikit Learn
- There were four different types of methods used:
  - Logistics Regression
  - Support Vector Machine
  - Decision Tree
  - K-Nearnest Neighbors
- All these methods were trained and reviewed using the process on the right and the methods below:
  - Standard Scaler processer
  - Train_test_split
  - GridSearchCV
  - Score method
  - Confusion matrix
- The complete app code can be found at:
  - https://github.com/marc-martini/Coursera_ML_Capstone/blob/37f2848044de65da1d7136f481656da29031c6d5/Machine%20Learning%20Prediction%20lab.ipynb



Results

EXPLORATORY DATA ANALYSIS RESULTS

INTERACTIVE ANALYTICS DEMO IN SCREENSHOTS

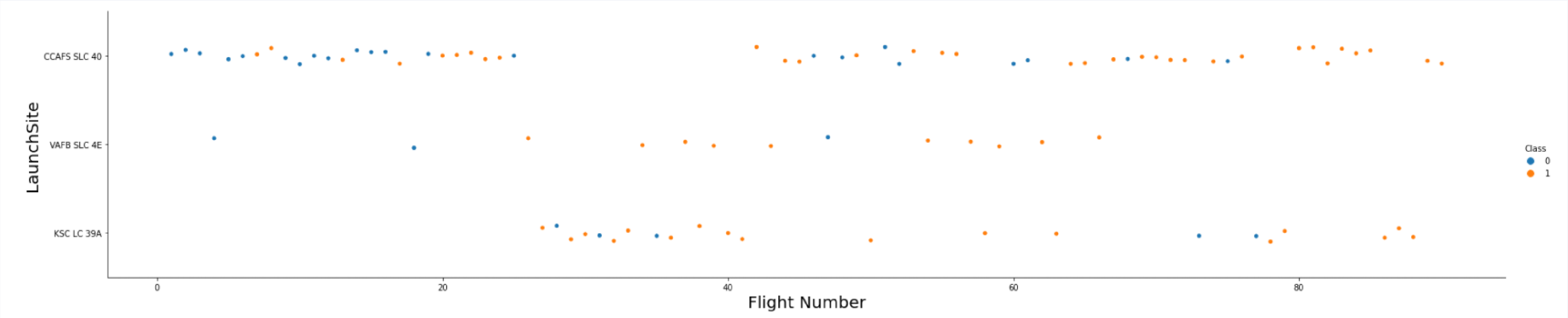
PREDICTIVE ANALYSIS RESULTS

Section 2

Insights drawn from EDA

Flight Number vs. Launch Site

Scatter plot of Flight Number vs. Launch Site

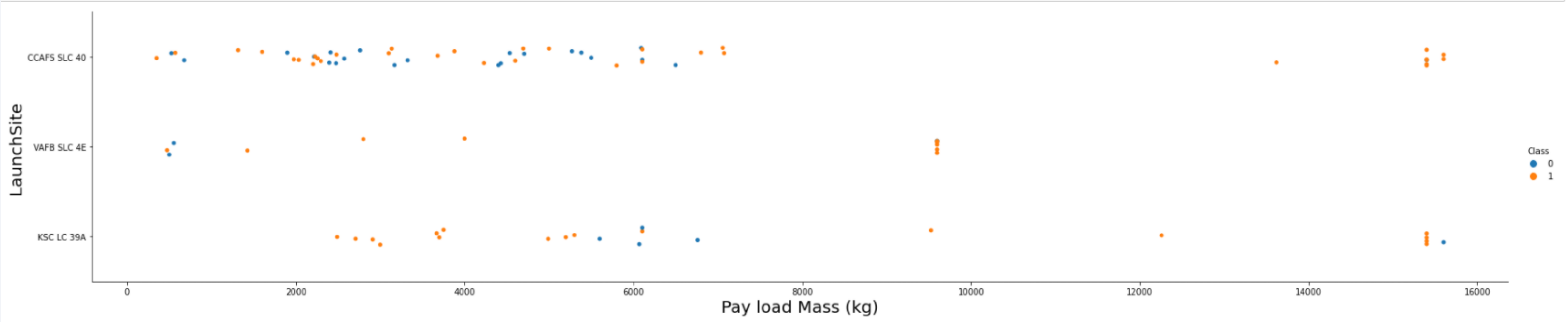


From the above we can see:

- The CCAFS site was primarily used for the early the flights
- There was a higher percent of failure for the early flights
- There have been the most flights from CCAFS

Payload vs. Launch Site

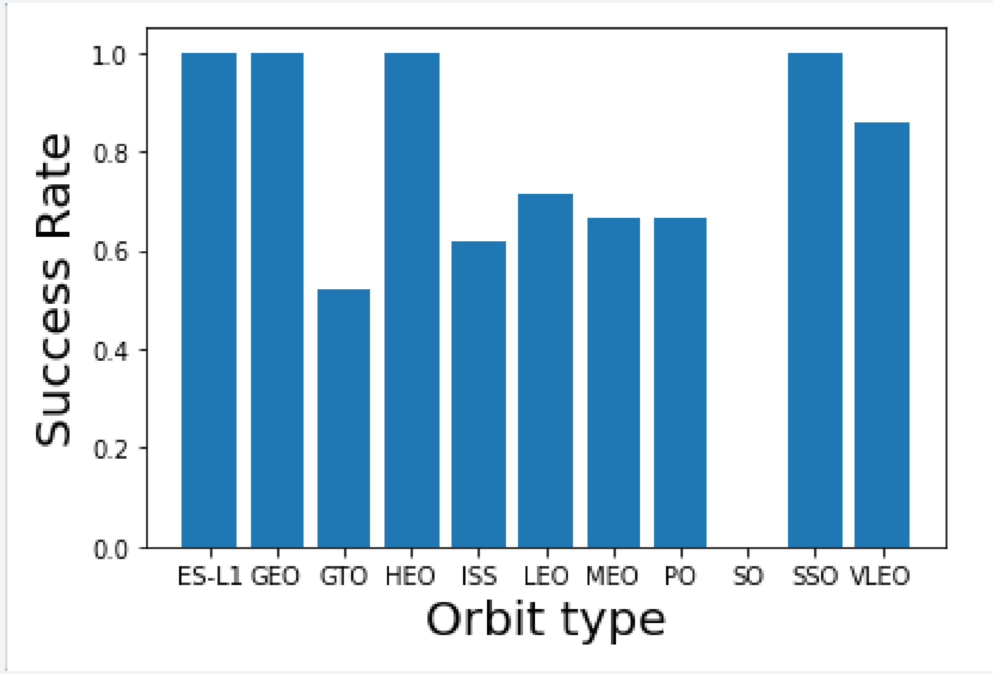
Scatter plot of Payload vs. Launch Site



- From the above we notice:
  - VAFB was only used for payloads less then 10000kg
  - No flights with a payload of between 7500kg and 13000kg were from CCAFS
  - All the flights from VAFB with a payload above 1500kg were a success
  - All flights but one above 7500kg were a success

Success Rate vs. Orbit Type

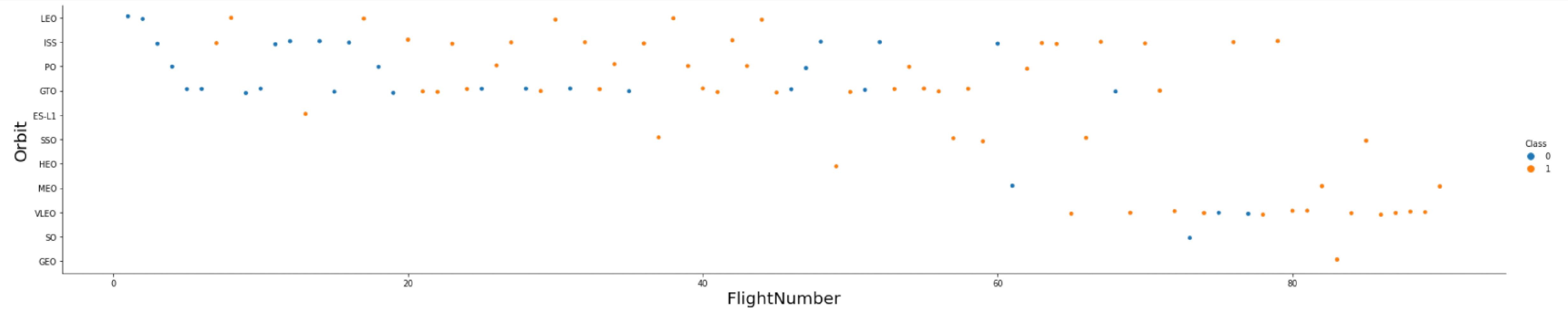
Bar chart for the success rate of each orbit type



- The bar chart shows:
  - There were no successful flights to the SO orbit.
  - Of the orbits that had flights, GTO flights has the lowest success rate
  - Four orbit flights; ES-L1, GEO, HEO and SSO has a 100% success rate

Flight Number vs. Orbit Type

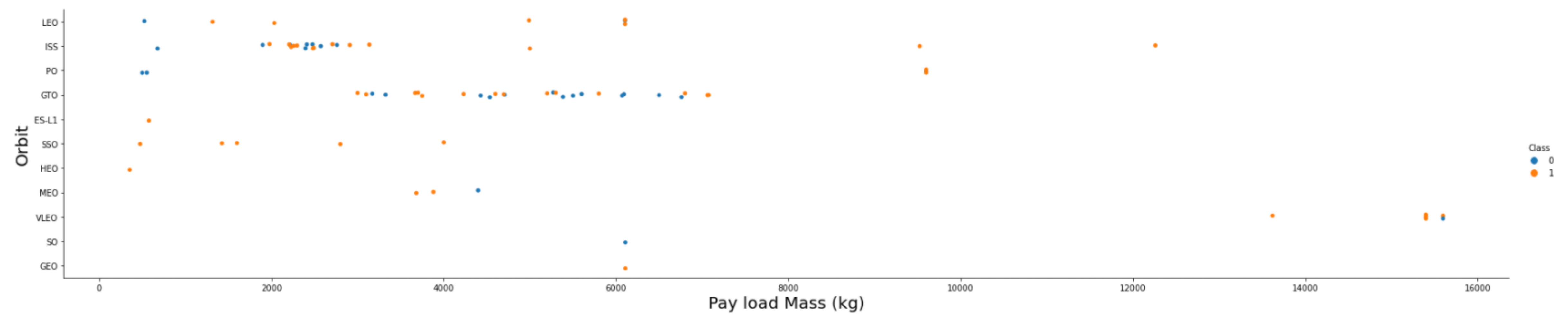
Scatter point of Flight number vs. Orbit type



- This scatter plot shows:
  - The early flights were only to LEO, ISS, PO and GEO
  - The most recent flights are mainly to VLEO
  - All flights to SSO were successful
  - Only one flight each to SO and GEO

Payload vs. Orbit Type

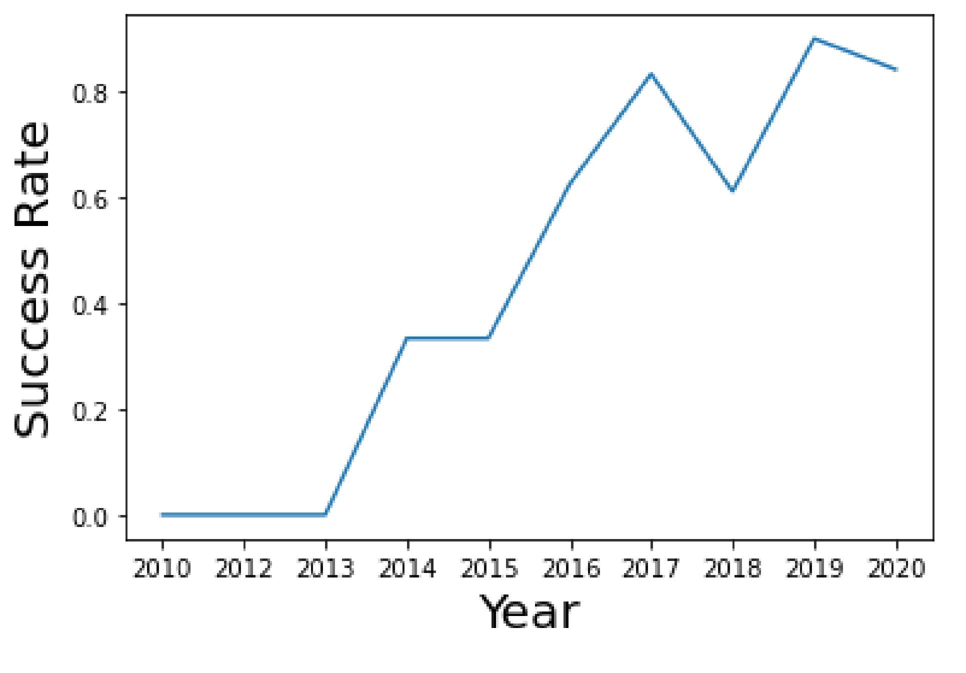
Scatter point of payload vs. orbit type



- The plot shows:
  - There is a higher success rate for heavier payloads
  - VLEO flights only take heavy pay loads and the only orbit with payloads over 13 000kg
  - Flights with heavier payloads have a higher success rate

Launch Success Yearly Trend

Line chart of average yearly success rate



- The line chart shows
  - Success rate for the first 3 years was 0
  - The success rate climbs from 2013 till 2019 but a small dip in 2018

All Launch Site Names

The names of the unique launch sites

```
%sql SELECT DISTINCT(LAUNCH_SITE) FROM SPACEX
```

```
 * ibm_db_sa://nkm90241:***@764264db-9824-4b7c
Done.
```

5]:

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

- There are 4 unique launch sites

Launch Site Names Begin with 'CCA'

5 records where launch sites begin with `CCA`

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

```
 * ibm_db_sa://nkm90241:***@764264db-9824-4b7c-82df-40d1b13897c2.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:32536/bludb
Done.
```

7]:

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 these 5 sites were used to launch to LEO orbit

Total Payload Mass

The total payload carried by boosters from NASA

```
: %sql SELECT SUM(payload_mass__kg_) FROM SPACEX where customer like 'NASA (CRS)'

   * ibm_db_sa://nkm90241:***@764264db-9824-4b7c-82df-40d1b13897c2.bs2io90l08kqb1
  Done.

10]:     1
     45596
```

- The total payload mass for NASA is 45 596 kg

Average Payload Mass by F9 v1.1

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

```
%sql SELECT AVG(payload_mass__kg_) FROM SPACEX where booster_version like '%F9 v1.1%'

 * ibm_db_sa://nkm90241:***@764264db-9824-4b7c-82df-40d1b13897c2.bs2io90l08kqb1od8lc
Done.
```

2]:

1
2534

- The average payload mass for the F9 v1.1 booster is 2534 kg

First Successful Ground Landing Date

The date of the first successful landing outcome on ground pad

```
%sql SELECT MIN(DATE) FROM SPACEX where landing__outcome like '%Success (ground pad)%'
```

```
 * ibm_db_sa://nkm90241:***@764264db-9824-4b7c-82df-40d1b13897c2.bs2io90l08kqb1od8lcg.
Done.
```

1
2015-12-22

- The first successful ground pad landing was on 2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

The names of boosters which have successfully landed on drone ship and had payload mass greater than 4000 but less than 6000

```
%sql SELECT booster_version FROM SPACEX WHERE landing__outcome LIKE '%Success (drone ship)%' AND payload_mass__kg_ > 4000 AND payload_mass__kg_ < 6000
```

```
 * ibm_db_sa://nkm90241:***@764264db-9824-4b7c-82df-40d1b13897c2.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:32536/bludb
Done.
```

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

- Four booster versions had successful outcomes with payloads between 4000kg and 6000kg.

Total Number of Successful and Failure Mission Outcomes

The total number of successful and failure mission outcomes

```
%sql SELECT COUNT(mission_outcome), mission_outcome FROM SPACEX GROUP BY mission_outcome
```

```
1]:  * ibm_db_sa://nkm90241:***@764264db-9824-4b7c-82df-40d1b13897c2.bs2io90l08kqb1od8lcg.da
Done.
```

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

- There have been 99 complete successes, 1 success without a clear status payload and 1 failure.

Boosters Carried Maximum Payload

The names of the booster which have carried the maximum payload mass

```
%sql SELECT DISTINCT(booster_version) FROM SPACEX WHERE payload_mass__kg_ = (SELECT MAX(payload_mass__kg_) FROM SPACEX)
```

```
 * ibm_db_sa://nkm90241:***@764264db-9824-4b7c-82df-40d1b13897c2.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:32536/
Done.
```

6]:

booster_version
F9 B5 B1048.4
F9 B5 B1048.5
F9 B5 B1049.4
F9 B5 B1049.5
F9 B5 B1049.7
F9 B5 B1051.3
F9 B5 B1051.4
F9 B5 B1051.6
F9 B5 B1056.4
F9 B5 B1058.3
F9 B5 B1060.2
F9 B5 B1060.3

- There are 12 booster versions that have carried the max payload

2015 Launch Records

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

```
%sql SELECT landing__outcome, booster_version, launch_site  FROM SPACEX WHERE landing__outcome LIKE '%Fail%' AND DATE LIKE '%2015%'

 * ibm_db_sa://nkm90241:***@764264db-9824-4b7c-82df-40d1b13897c2.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:32536/bludb
Done.
```

[6]:

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

- There are 2 failed landings in 2015

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

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(landing__outcome) FROM SPACEX WHERE DATE BETWEEN '2010-06-04' AND '2017-03-20' GROUP BY landing__outcome ORDER BY COUNT(landing__outcome) desc
```

```
 * ibm_db_sa://nkm90241:***@764264db-9824-4b7c-82df-40d1b13897c2.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:32536/bludb
Done.
```

4]:

landing__outcome	2
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

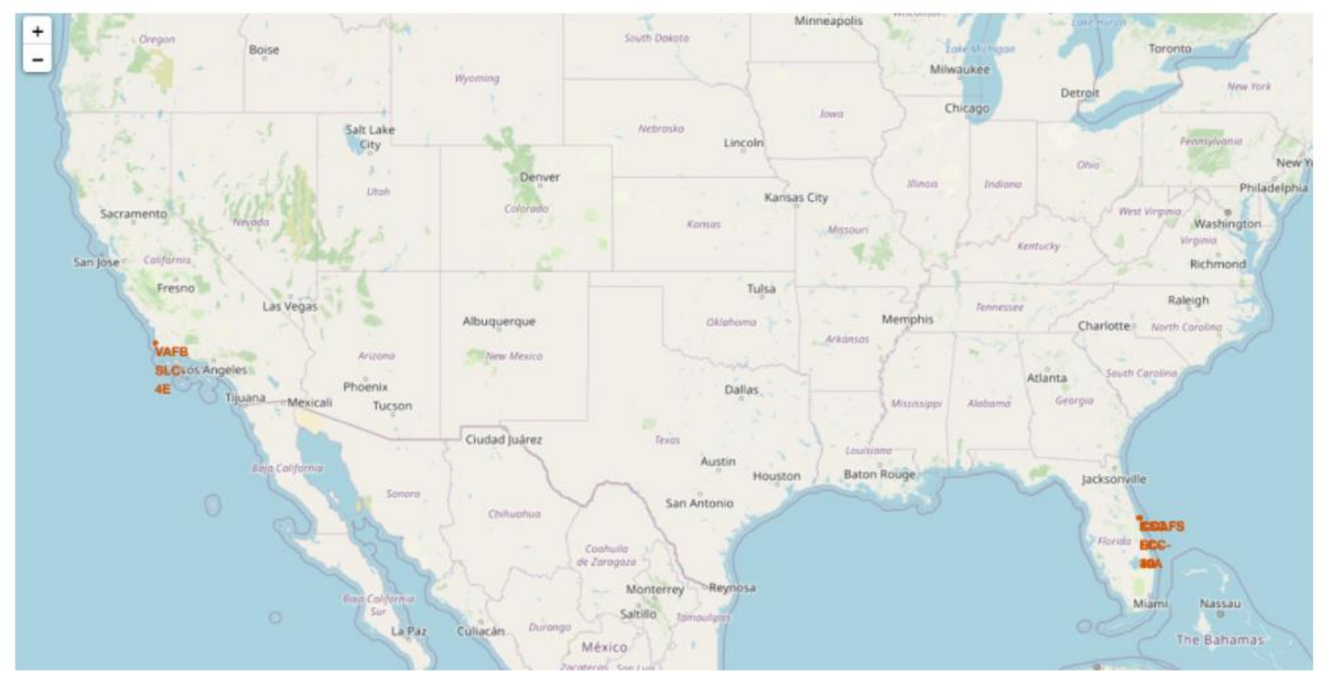
- There were 10 no attempt ladings, 7 total failures and 11 total successes between the time period.

Section 4

Launch Sites Proximities Analysis

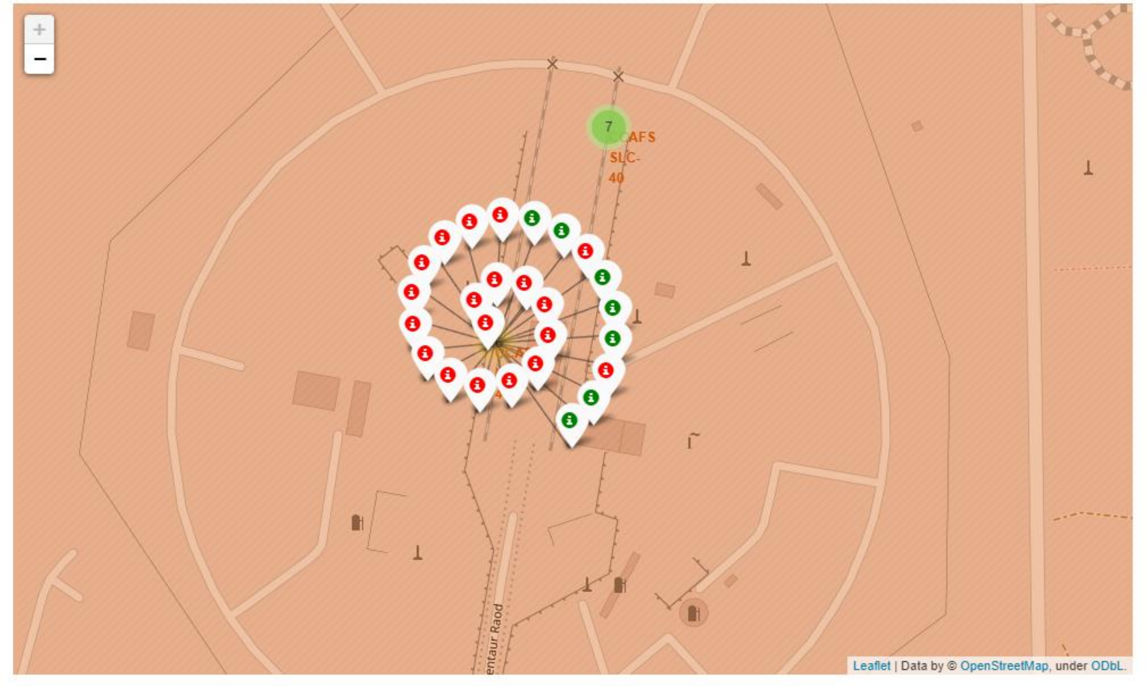
Map of Launch Sites for SpaceX

- This map shows the launch sites for SpaceX across the United States
- The launch sites are all near the coast
- The locations of the launch sites are in the south of America and near the equator
- These locations are warm and have moderate climate to facilitate better landings
- The launch near the ocean to allow for ocean landings



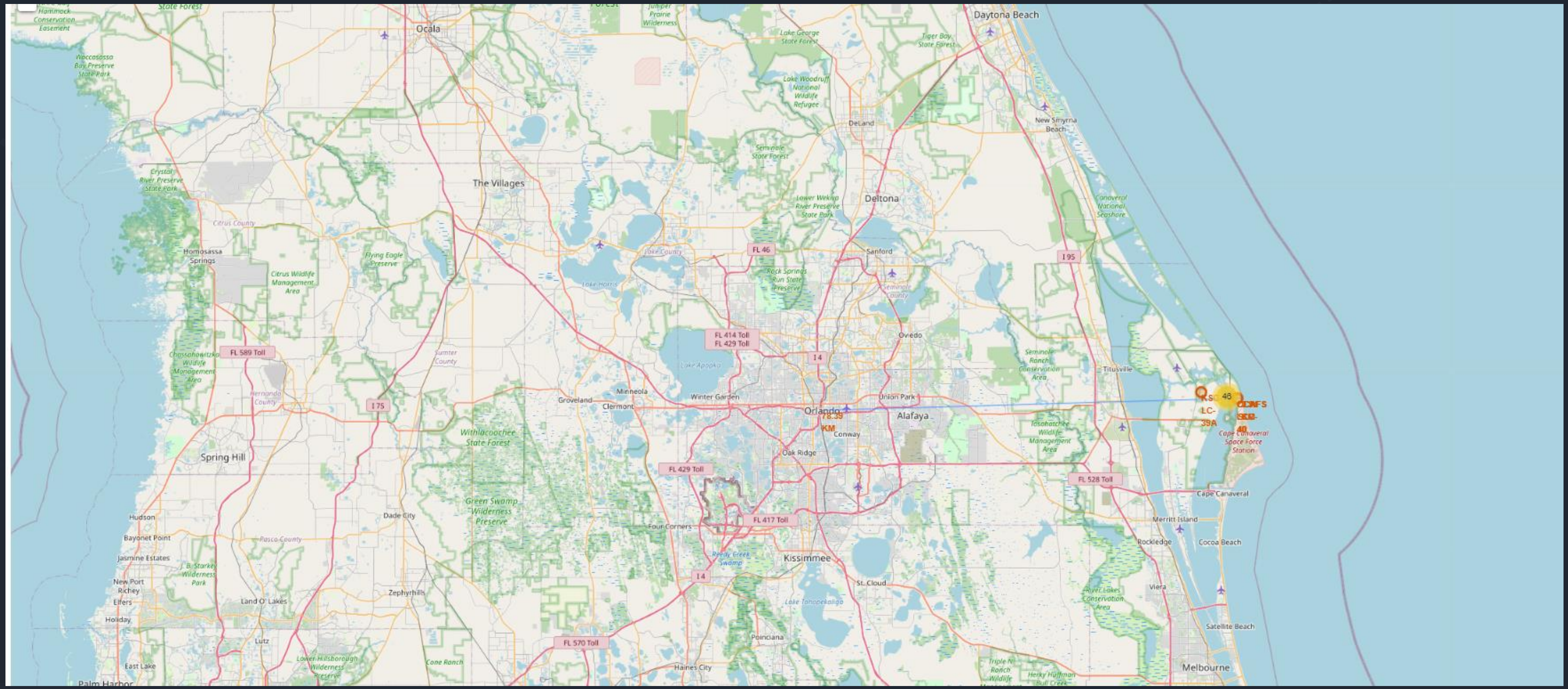
Map of launch site CCAFS SLC site

- This map shows the launch details for launch site CCAFS SLC
- The pins show the successful launches from this site



Map of CCAFS in relation to Orlando

- Orlando, Florida is 78.39 km from CCAFS launch site

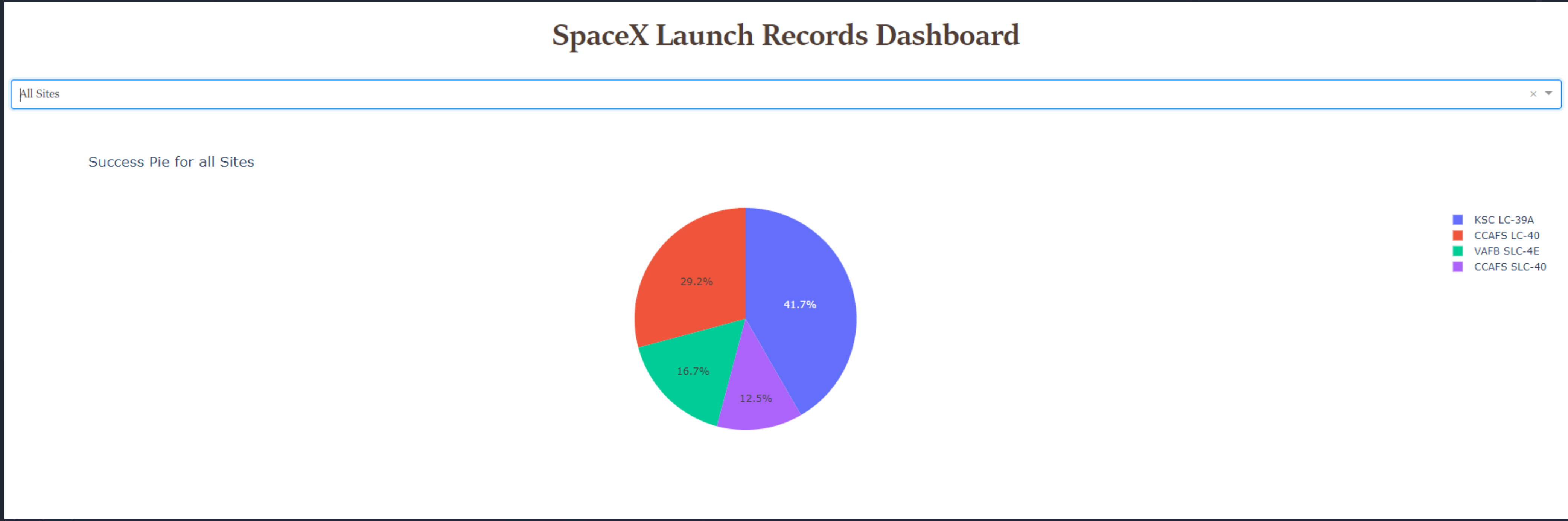


Section 5

Build a Dashboard with Plotly Dash

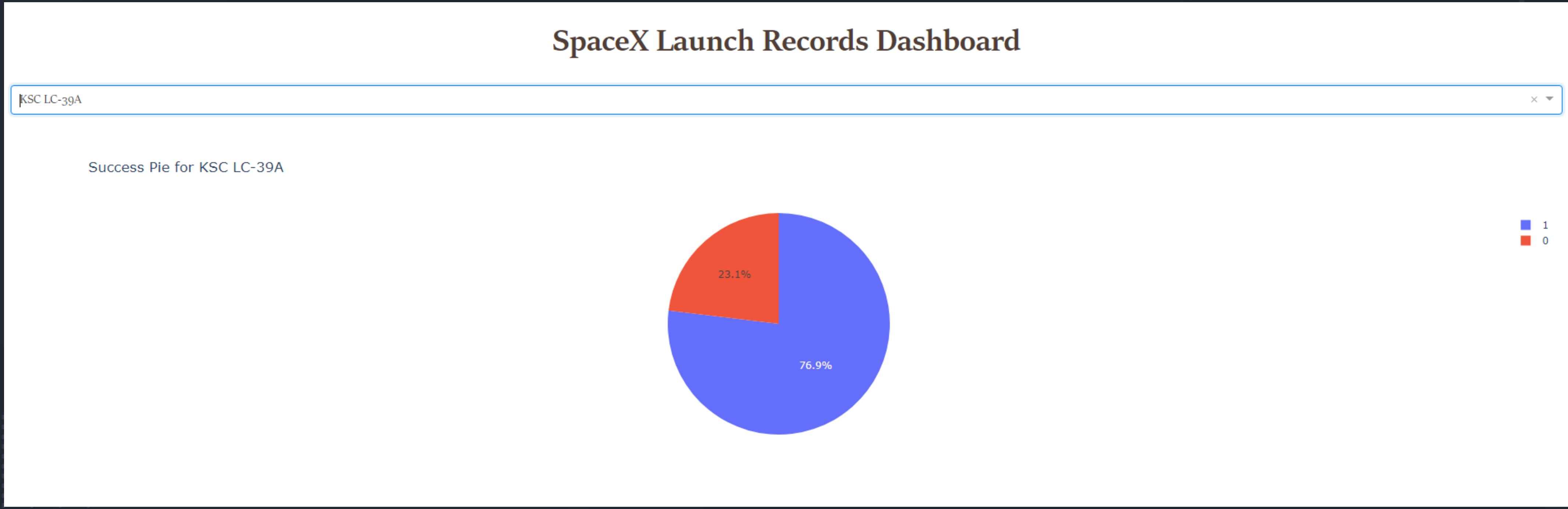
Pie chart for launch success count for all sites

- KSC LC is the most successful site
- The CCAFS is the least successful site



Piechart for the launch site with highest launch success ratio -

- KSC LC was the site with highest success ratio
- This site had a 76,9% success ratio and a 23,1% failure.



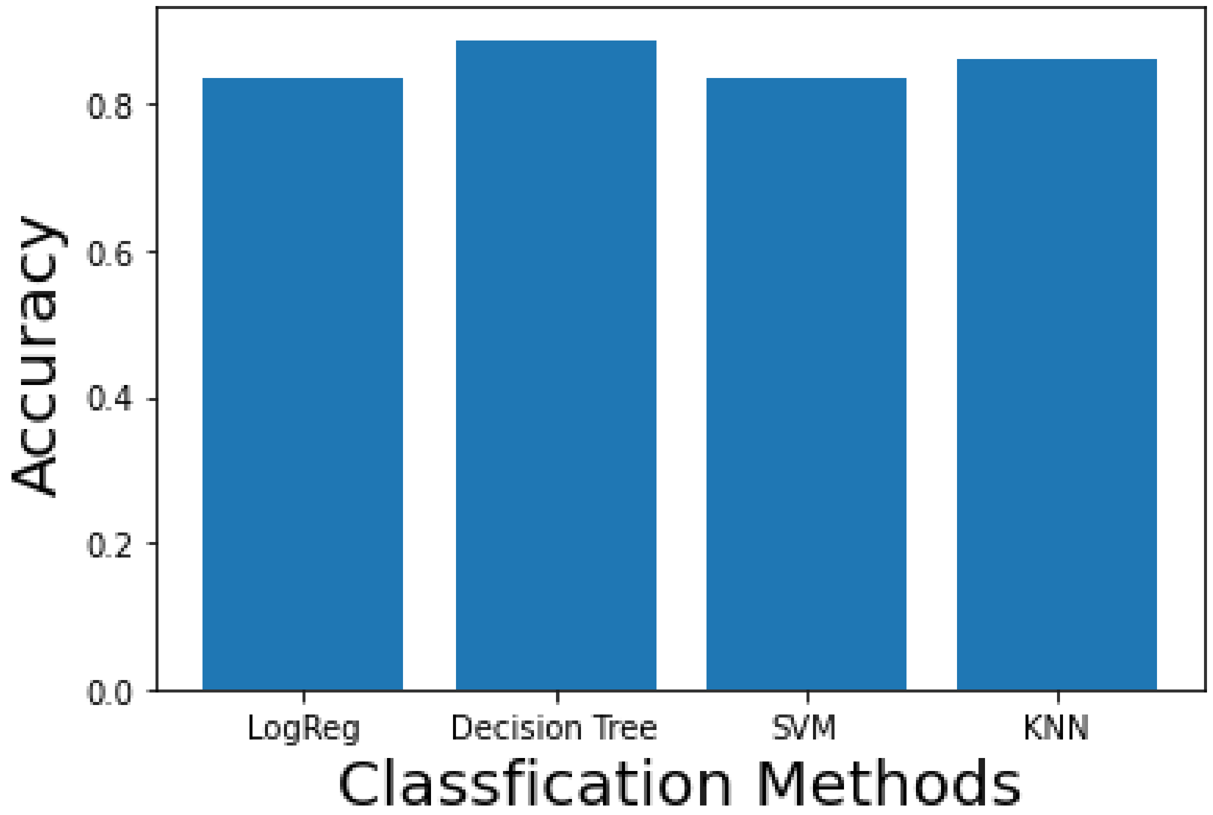
Payload vs. Launch Outcome scatter plot for all sites with a range of 2000-7000kg

- The v1.1 Booster version has the most failures
- This Booster is only used for payloads lees then 5000 kg
- There were no successes between 5500kg and 7000kg

Section 6

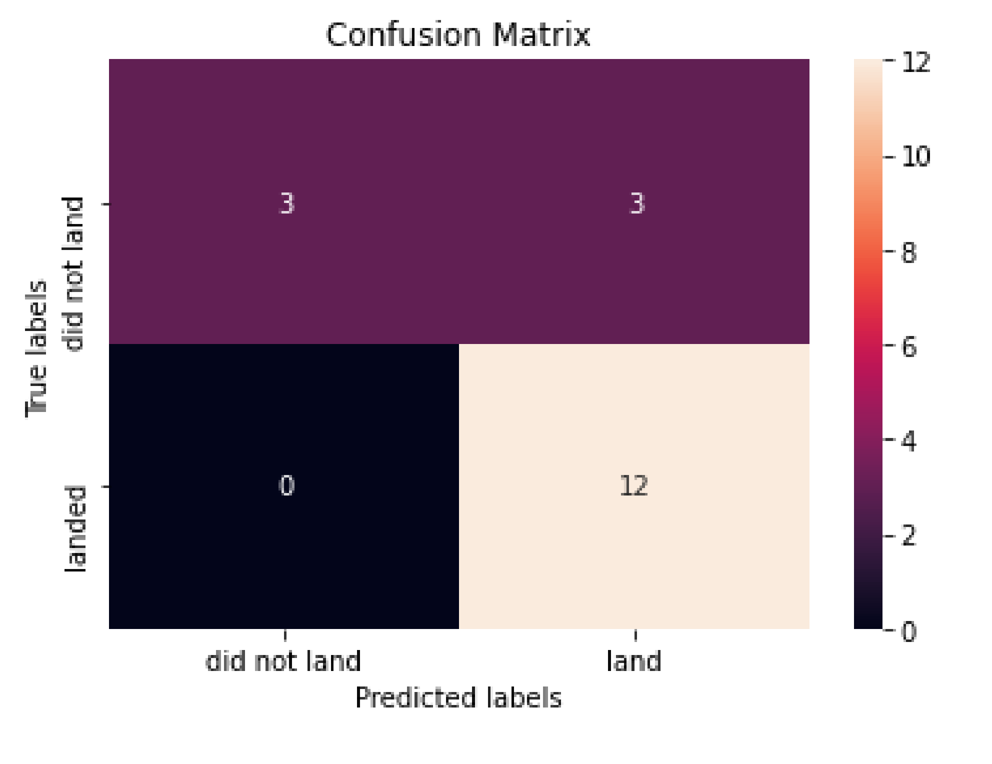
Predictive Analysis (Classification)

Classification Accuracy



- The Decision Tree Classification method has the highest accuracy

Confusion Matrix



- The Confusion Matrix for the decision tree classification show:
- There were 3 false positive classifications
- There were no false negative classfications

Conclusions

- The combination of queries and visualizations are the most effective methods to eplore and analyze the data
- The SpaceX booster and flights improved over time with more expansive and heavy payloads with a higher success rate
- Building an interactive dashboard allows for further data exploration and for the end user to view and understand the data further.
- The most accurate classification method is the decision tree
- A trained classification model is viable to predict and classify a flight outcome
- Using a viable classification model can reduce the cost of each space flight
- More cost-effective space flight increases the commercial viability of space travel
- More commercial space travel increases the technological advancement and exploration into space

Appendix 1

- The full Github repository for the notebooks and code can be found at:
    - https://github.com/marc-martini/Coursera_ML_Capstone

Appendix 2

- Snippet from main data set

	Flight Number	Date	Time (UTC)	Booster Version	Launch Site	Payload	Payload Mass (kg)	Orbit	Customer	Mission Outcome	Landing Outcome	class	Lat	Long
5	7	2013-12-03	22:41:00	F9 v1.1	CCAFS LC-40	SES-8	3170.0	GTO	SES	Success	No attempt	0	28.562302	-80.577356
6	8	2014-01-06	22:06:00	F9 v1.1	CCAFS LC-40	Thaicom 6	3325.0	GTO	Thaicom	Success	No attempt	0	28.562302	-80.577356
7	9	2014-04-18	19:25:00	F9 v1.1	CCAFS LC-40	SpaceX CRS-3	2296.0	LEO (ISS)	NASA (CRS)	Success	Controlled (ocean)	0	28.562302	-80.577356
8	10	2014-07-14	15:15:00	F9 v1.1	CCAFS LC-40	OG2 Mission 1 6 Orbcomm-OG2 satellites	1316.0	LEO	Orbcomm	Success	Controlled (ocean)	0	28.562302	-80.577356
9	11	2014-08-05	8:00:00	F9 v1.1	CCAFS LC-40	AsiaSat 8	4535.0	GTO	AsiaSat	Success	No attempt	0	28.562302	-80.577356
10	12	2014-09-07	5:00:00	F9 v1.1 B1011	CCAFS LC-40	AsiaSat 6	4428.0	GTO	AsiaSat	Success	No attempt	0	28.562302	-80.577356
11	13	2014-09-21	5:52:00	F9 v1.1 B1010	CCAFS LC-40	SpaceX CRS-4	2216.0	LEO (ISS)	NASA (CRS)	Success	Uncontrolled (ocean)	0	28.562302	-80.577356
12	14	2015-01-10	9:47:00	F9 v1.1 B1012	CCAFS LC-40	SpaceX CRS-5	2395.0	LEO (ISS)	NASA (CRS)	Success	Failure (drone ship)	0	28.562302	-80.577356
14	16	2015-03-02	3:50:00	F9 v1.1 B1014	CCAFS LC-40	ABS-3A Eutelsat 115 West B	4159.0	GTO	ABS Eutelsat	Success	No attempt	0	28.562302	-80.577356
15	17	2015-04-14	20:10:00	F9 v1.1 B1015	CCAFS LC-40	SpaceX CRS-6	1898.0	LEO (ISS)	NASA (CRS)	Success	Failure (drone ship)	0	28.562302	-80.577356
16	18	2015-04-27	23:03:00	F9 v1.1 B1016	CCAFS LC-40	TürkmenÄlem 52°E / MonacoSAT	4707.0	GTO	Turkmenistan National Space Agency	Success	No attempt	0	28.562302	-80.577356
17	19	2015-06-28	14:21:00	F9 v1.1 B1018	CCAFS LC-40	SpaceX CRS-7	1952.0	LEO (ISS)	NASA (CRS)	Failure (in flight)	Precluded (drone ship)	1	28.562302	-80.577356
18	20	2015-12-22	1:29:00	F9 FT B1019	CCAFS LC-40	OG2 Mission 2 11 Orbcomm-OG2 satellites	2034.0	LEO	Orbcomm	Success	Success (ground pad)	1	28.562302	-80.577356
19	22	2016-03-04	23:35:00	F9 FT B1020	CCAFS LC-40	SES-9	5271.0	GTO	SES	Success	Failure (drone ship)	0	28.562302	-80.577356
20	23	2016-04-08	20:43:00	F9 FT B1021.1	CCAFS LC-40	SpaceX CRS-8	3136.0	LEO (ISS)	NASA (CRS)	Success	Success (drone ship)	1	28.562302	-80.577356
21	24	2016-05-06	5:21:00	F9 FT B1022	CCAFS LC-40	JCSAT-14	4696.0	GTO	SKY Perfect JSAT Group	Success	Success (drone ship)	1	28.562302	-80.577356
22	25	2016-05-27	21:39:00	F9 FT B1023.1	CCAFS LC-40	Thaicom 8	3100.0	GTO	Thaicom	Success	Success (drone ship)	1	28.562302	-80.577356
23	26	2016-06-15	14:29:00	F9 FT B1024	CCAFS LC-40	ABS-2A Eutelsat 117 West B	3600.0	GTO	ABS Eutelsat	Success	Failure (drone ship)	0	28.562302	-80.577356
24	27	2016-07-18	4:45:00	F9 FT B1025.1	CCAFS LC-40	SpaceX CRS-9	2257.0	LEO (ISS)	NASA (CRS)	Success	Success (ground pad)	1	28.562302	-80.577356
25	28	2016-08-14	5:26:00	F9 FT B1026	CCAFS LC-40	JCSAT-16	4600.0	GTO	SKY Perfect JSAT Group	Success	Success (drone ship)	1	28.562302	-80.577356

Appendix 2

- Snippet from dashboard code

```
app.layout = html.Div(children=[html.H1('SpaceX Launch Records Dashboard',
                                        style={'textAlign': 'center', 'color': '#503D36',
                                               'font-size': 40}),
                                # TASK 1: Add a dropdown list to enable Launch Site selection
                                # The default select value is for ALL sites
                                # dcc.Dropdown(id='site-dropdown',...)
                                dcc.Dropdown(id='site-dropdown',
                                             options=[
                                                 {'label': 'All Sites', 'value': 'ALL'},
                                                 {'label': 'CCAFS LC-40', 'value': 'CCAFS LC-40'},
                                                 {'label': 'VAFB SLC-4E', 'value': 'VAFB SLC-4E'},
                                                 {'label': 'KSC LC-39A', 'value': 'KSC LC-39A'},
                                                 {'label': 'CCAFS SLC-40', 'value': 'CCAFS SLC-40'},
                                             ],
                                             value='ALL',
                                             placeholder="Select a Launch Site",
                                             searchable=True
                                             ),
                                html.Br(),

                                # TASK 2: Add a pie chart to show the total successful launches count for all sites
                                # If a specific launch site was selected, show the Success vs. Failed counts for the site
                                html.Div(dcc.Graph(id='success-pie-chart')),
                                html.Br(),

                                html.P("Payload range (Kg):"),
                                # TASK 3: Add a slider to select payload range
                                #dcc.RangeSlider(id='payload-slider',...)
                                html.Div(dcc.RangeSlider(id='payload-slider',
                                                 min=0, max=10000, step=1000,
                                                 marks={0: '0',
                                                     1000: '1000',
                                                     2000: '2000',
                                                     3000: '3000',
                                                     4000: '4000',
                                                     5000: '5000',
                                                     6000: '6000',
                                                     7000: '7000',
                                                     8000: '8000',
                                                     9000: '9000',
                                                     10000: '10000',
                                                     },
                                                 value=[min_payload, max_payload])),

                                # TASK 4: Add a scatter chart to show the correlation between payload and launch success
                                html.Div(dcc.Graph(id='success-payload-scatter-chart')),
                                ])
```

Appendix 3

- Snippet from dashboard code

```
app.layout = html.Div(children=[html.H1('SpaceX Launch Records Dashboard',
                                        style={'textAlign': 'center', 'color': '#503D36',
                                               'font-size': 40}),
                                # TASK 1: Add a dropdown list to enable Launch Site selection
                                # The default select value is for ALL sites
                                # dcc.Dropdown(id='site-dropdown',...)
                                dcc.Dropdown(id='site-dropdown',
                                             options=[
                                                 {'label': 'All Sites', 'value': 'ALL'},
                                                 {'label': 'CCAFS LC-40', 'value': 'CCAFS LC-40'},
                                                 {'label': 'VAFB SLC-4E', 'value': 'VAFB SLC-4E'},
                                                 {'label': 'KSC LC-39A', 'value': 'KSC LC-39A'},
                                                 {'label': 'CCAFS SLC-40', 'value': 'CCAFS SLC-40'},
                                             ],
                                             value='ALL',
                                             placeholder="Select a Launch Site",
                                             searchable=True
                                             ),
                                html.Br(),

                                # TASK 2: Add a pie chart to show the total successful launches count for all sites
                                # If a specific launch site was selected, show the Success vs. Failed counts for the site
                                html.Div(dcc.Graph(id='success-pie-chart')),
                                html.Br(),

                                html.P("Payload range (Kg):"),
                                # TASK 3: Add a slider to select payload range
                                #dcc.RangeSlider(id='payload-slider',...)
                                html.Div(dcc.RangeSlider(id='payload-slider',
                                                 min=0, max=10000, step=1000,
                                                 marks={0: '0',
                                                     1000: '1000',
                                                     2000: '2000',
                                                     3000: '3000',
                                                     4000: '4000',
                                                     5000: '5000',
                                                     6000: '6000',
                                                     7000: '7000',
                                                     8000: '8000',
                                                     9000: '9000',
                                                     10000: '10000',
                                                     },
                                                 value=[min_payload, max_payload])),

                                # TASK 4: Add a scatter chart to show the correlation between payload and launch success
                                html.Div(dcc.Graph(id='success-payload-scatter-chart')),
                                ])
```

Thank you!