IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Hera Chua
November 2, 2024

Outline

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

Executive Summary

- Data collection methodology:
  - Data are collected from the web – SpaceX API and Falcon 9 launch information.
- Perform data wrangling
  - Use Python to manipulate data in a Pandas dataframe.
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Models used are Decision Tree, Logistic Regression, K Nearest Neighbors, and Support Vector Machine, with GridSearchCV() to find the best parameters for each model.
- Conclusion

Introduction

- The purpose of this project is to predict if SpaceX Falcon 9 first stage will land successfully.
- Falcon 9 rocket launches cost 62 million dollars; other providers cost 165 million dollars each, much of the savings is because SpaceX can reuse the first stage.
- If we determine if the first stage will land, we can determine the cost of a launch.
- This information can be used if an alternate company wants to bind against SpaceX for a rocket launch.

Section 1

Methodology

Methodology

Executive Summary

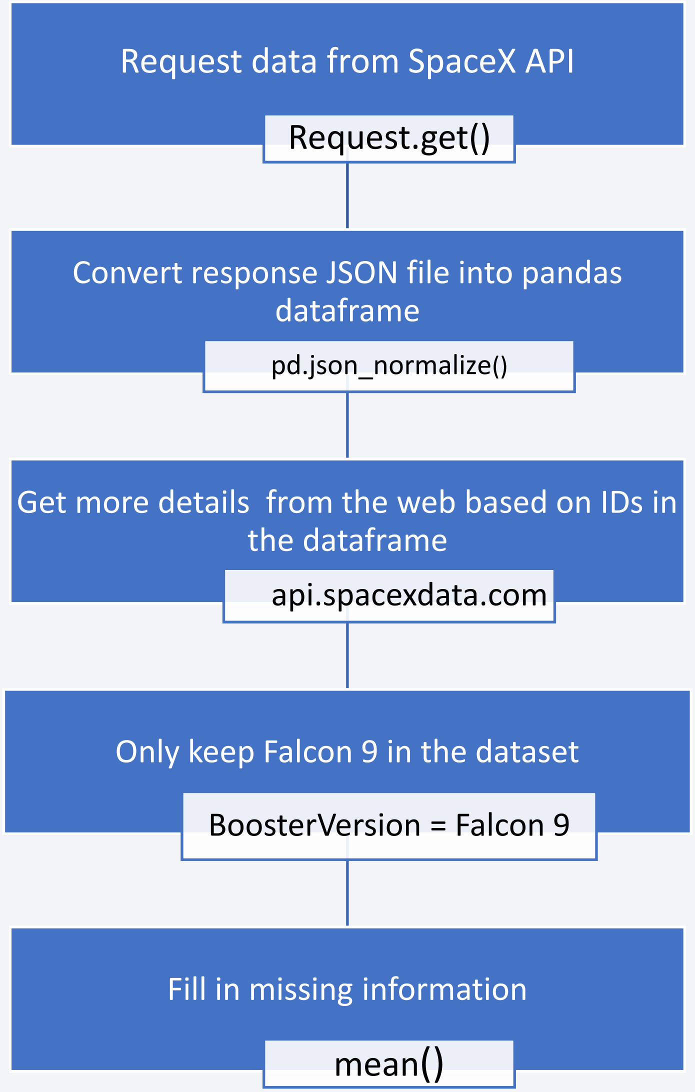
- Data collection methodology:
  - Data are collected from the web – SpaceX API and Falcon 9 launch information.
- Perform data wrangling
  - Use Python to manipulate data in a Pandas dataframe.
- Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - Models used are Decision Tree, Logistic Regression, K Nearest Neighbors, and Support Vector Machine, with GridSearchCV() to find the best parameters for each model.

Data Collection

- This section contains the following:
  - Data Collection – SpaceX.
  - Data Collection – Scraping.
  - Data Wrangling.
  - EDA with Data Visualization
  - EDA with SQL
  - Build an Interactive Map with Folium
  - Build a Dashboard with Plotly Dash
  - Predictive Analysis (Classification)

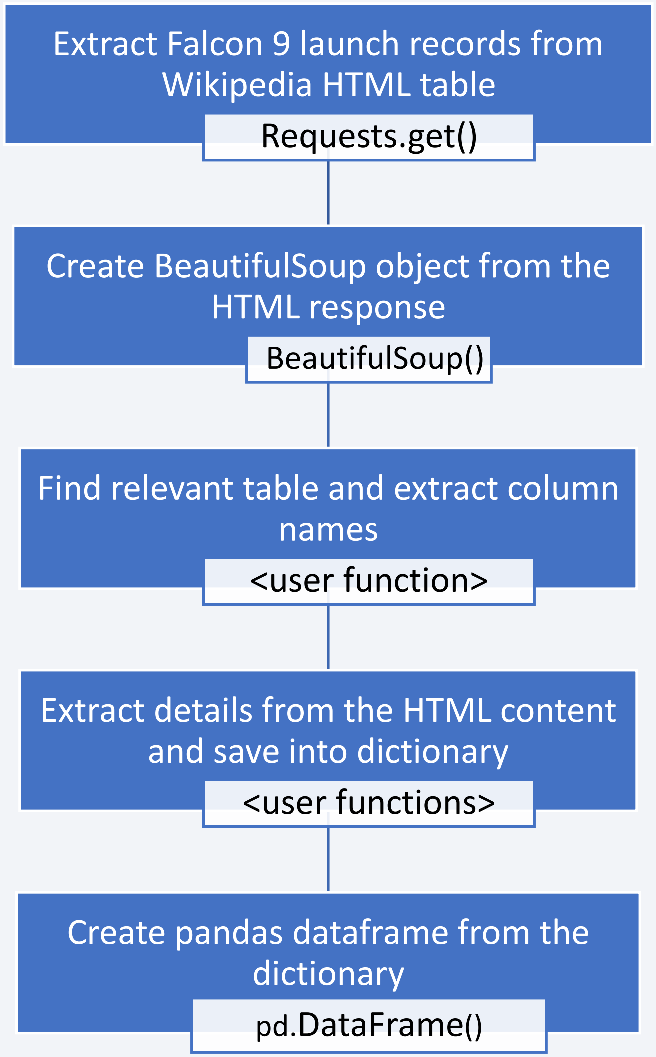
Data Collection – SpaceX API

- Data collected from SpaceX API involved five major steps.
- Details of how data was collection can be found here: [Pytrhon-for-Data-Science-Project/001 jupyter-labs-spacex-data-collection-api.ipynb at main · herachua10/Pytrhon-for-Data-Science-Project]()



Data Collection - Scraping

- Data collected from web scraping involved five major steps.
- Details of how data was collection can be found here: [Pytrhon-for-Data-Science-Project/002 jupyter-labs-webscraping.ipynb at main · herachua10/Pytrhon-for-Data-Science-Project]()



Data Wrangling

- The purpose of data wrangling is to find patterns in the data to determine what would be the label for training supervised models.
- Four main steps were performed.
- Create a landing outcome label.
- Details of data wrangling processes can be found here: Pytrhon-for-Data-Science-Project/003 labs-jupyter-spacex-data wrangling_jupyterlite.ipynb at main · herachua10/Pytrhon-for-Data-Science-Project

Identify and calculate the percentage of the missing values

Calculate the number of launches on each site

Calculate the number and occurrence of each orbit

Calculate the number and occurrence of outcome per orbit type

Create a landing outcome label

1 = success 0 = failed

EDA with Data Visualization

- The following data visualization were created by using matplotlib and seaborn for EDA:
  - Use category plot to visualize the relationship between flight number and payload to launch outcome.
  - Use a category plot to visualize the relationship between flight number and launch site to launch outcome.
  - Use a scatter plot to visualize the relationship between payload mass and launch site to launch outcome.
  - Use a bar chart to visualize the success rate of each orbit type.
  - Use a scatter plot to visualize the relationship between flight number and orbit type to launch outcome.
  - Use a scatter plot to visualize the relationship between payload mass and orbit type to launch outcome.
  - Use a line chart to visualize the launch success yearly trend.
- Some variables were deemed would affect the success rates and would be used for our prediction. They are flight number, payload mass, orbit, launch site, flights, gridfins, reused, legs, landing pad, block, reused count, and serial.
- Details of the data visualization graphs can be found here: Pytrhon-for-Data-Science-Project/005 Visualize - edadataviz.ipynb at main · herachua10/Pytrhon-for-Data-Science-Project

EDA with SQL

- The following SQL queries were performed for EDA:
  - Display the names of the unique launch sites in the space mission.
  - Display 5 records where launch sites with string 'KSC'
  - Display the total payload mass carried by boosters launched by NASA (CRS)
  - Display average payload mass carried by booster version F9 v1.1
  - List the date where the successful landing outcome in drone ship was achieved.
  - List the names of the boosters which have success in ground pad and have payload mass greated than4000 but less than 6000.
  - List the total number of successful and failure mission outcomes
  - List the names of the booster_versions which have carried the maximum payload mass. Use a subquery
  - List the records which will display the month names, succesful landing_outcomes in ground pad ,booster versions, launch_site for the months in year 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
- Details of EDA with SQL can be found here: [Pytrhon-for-Data-Science-Project/004 sqlite - jupyter-labs-eda-sql-edx_sqllite.ipynb at main · herachua10/Pytrhon-for-Data-Science-Project]()

Build an Interactive Map with Folium

- The purpose of building an interactive map is to discover if location and proximities of a launch site i.e. the initial position of the rocket trajectories, affect the launch success rate.
- The folium maps contain the following information:
  - The four launch sites with a circle of 1000 radius, and a marker to show the launch site name.
  - MarkerCluster to show the launch outcome for each launch i.e. red = failed, green = success.
  - The distances between a launch site to its proximities i.e. closest city and ocean, by adding lines and showing the distances.
- Details of the map can be found here: Pytrhon-for-Data-Science-Project/006 lab_jupyter_launch_site_location.ipynb at main · herachua10/Pytrhon-for-Data-Science-Project

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
- The Dashboard is created to provide information regarding:
    - What site has the largest successful launches and highest launch success rate.
    - Which payload range(s) has the highest and lowest launch success rate.
    - What F9 booster version has the highest launch success rate.
- The Dashboard contains a dropdown with the option to view information for all sites or a particular site for:
    - Launch success rate with pie chart.
    - The highest and lowest launch success rate for the F9 booster version with scatter plot. There is a slicer to set the payload range (kg).
- Details of the Dashboard can be found here: Pytrhon-for-Data-Science-Project/007 Dashboard with Plotly dash.pdf at main · herachua10/Pytrhon-for-Data-Science-Project

Predictive Analysis (Classification)

- The model development process is shown below:

Create Y variable	Convert variable Class to numpy	To_numpy()
Create X variable	Standardize the data in X	Preprocessing.Standard Scaler().fit.transform()
Create training and testing data	Split the data in X to test and train. Test size = 20%	Train_test_split()

- The following four models were used:

Logistic Regression | Support Vector Machine | Decision Tree | K Nearest Neighbors

- Use GridSearchCV to find the best parameters
- Use f_score and accuracy_score to find the best models

- Details of the analysis can be found here: [Pytrhon-for-Data-Science-Project/008 SpaceX_Machine Learning Prediction_Part_5.ipynb at main · herachua10/Pytrhon-for-Data-Science-Project]

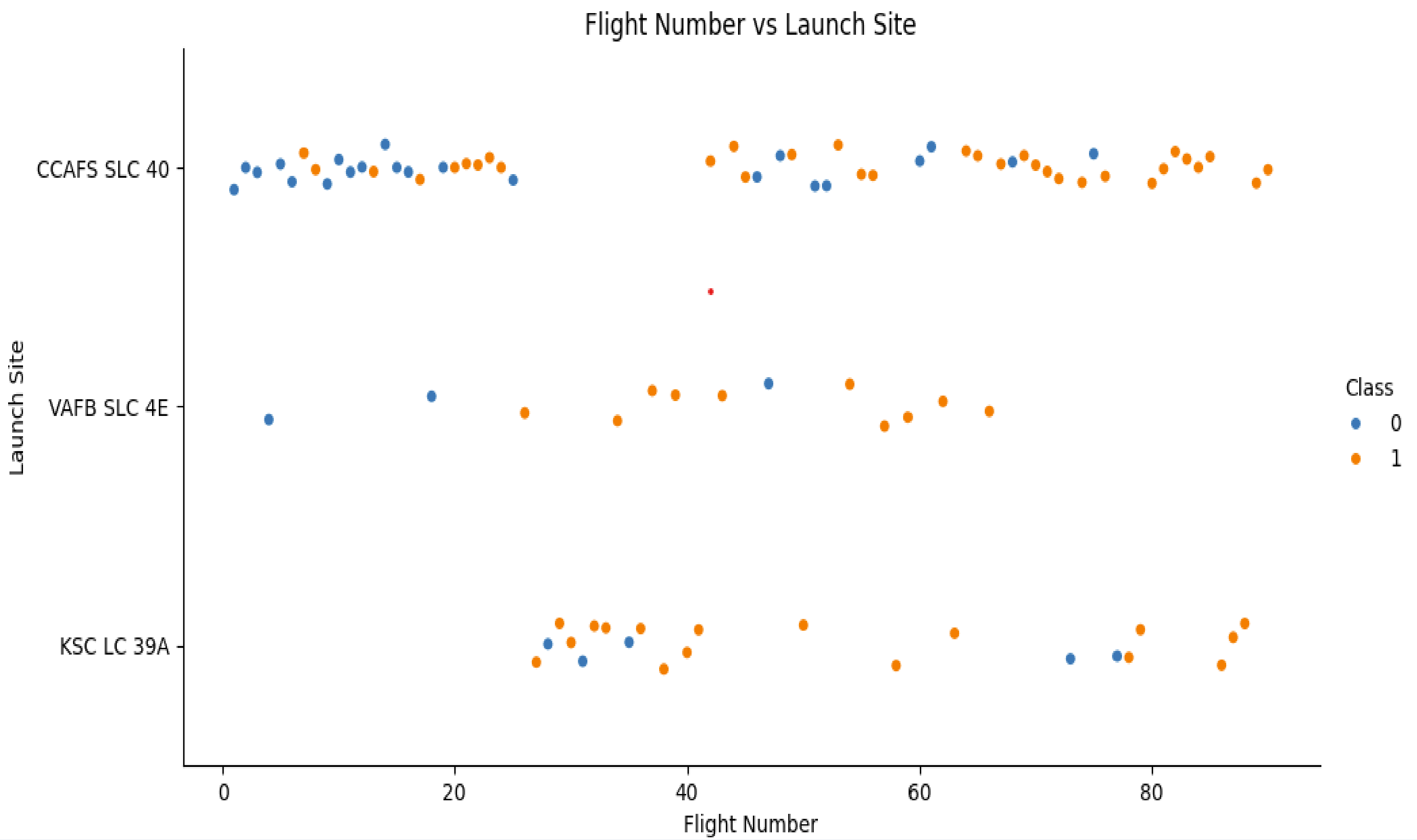
Results

- This section contains the following:
  - Exploratory data analysis results
  - Interactive analytics demo in screenshots
  - Predictive analysis results

Section 2

Insights drawn from EDA

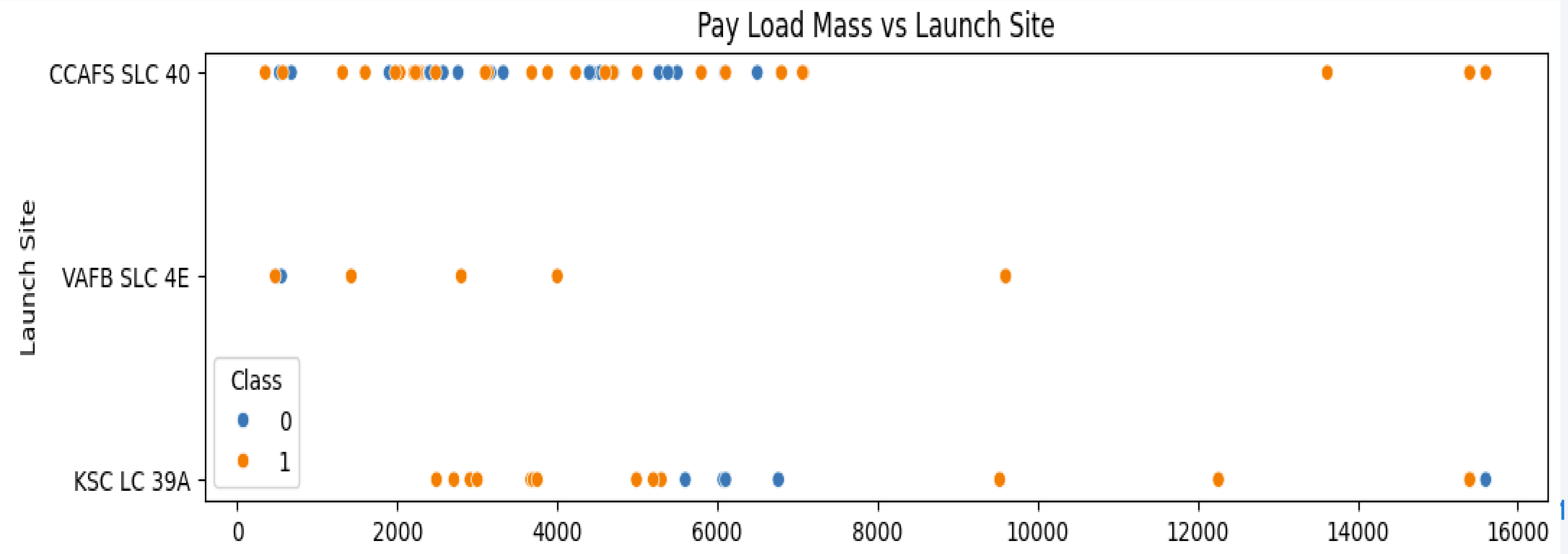
Flight Number vs. Launch Site



- CCAFS SLC 40 has the most flights.
- Most failed launches were from earlier flights.
- Since Flight number 78, all launches were success.

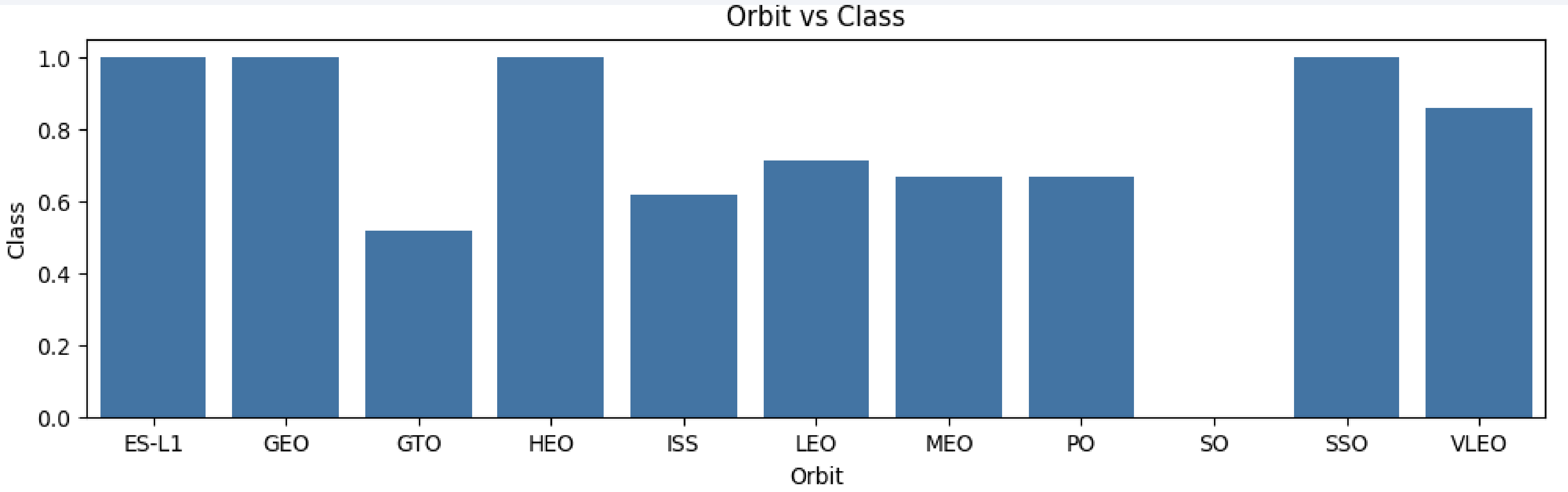
Payload vs. Launch Site

- VAFB-SLC has no rockets launched for heavy payload mass greater than 10000.
- The majority of the payload mass are below 7600.
- There are only 8 payload mass over 8500, with only one failed launch.



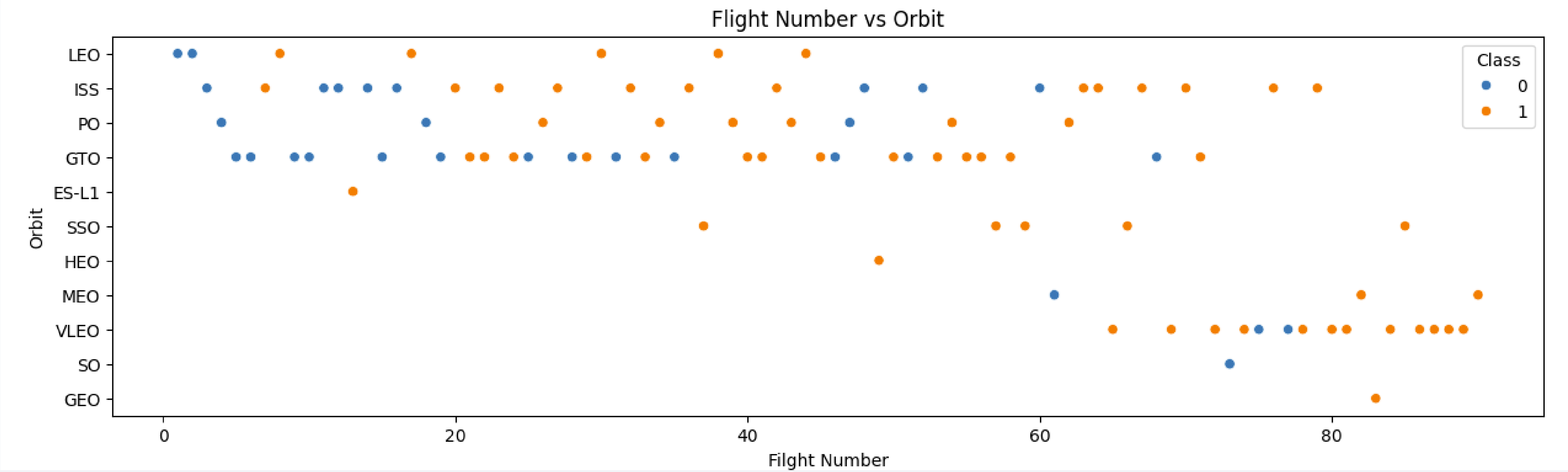
Success Rate vs. Orbit Type

- Four orbits have 100% success rate – ES-L1, GEO, HEO, and SSO.
- SO has no success launch.



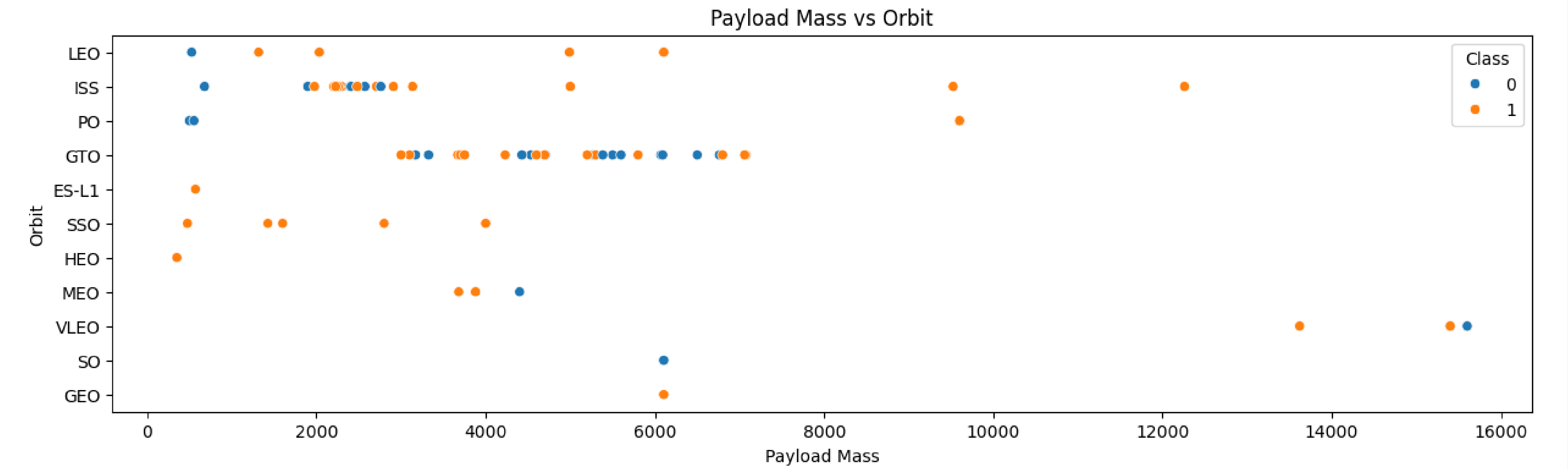
Flight Number vs. Orbit Type

- Historically, GTO and ISS have the most flights, but the majority of the most recent flights went to VLEO.
- The success rates improve overtime.



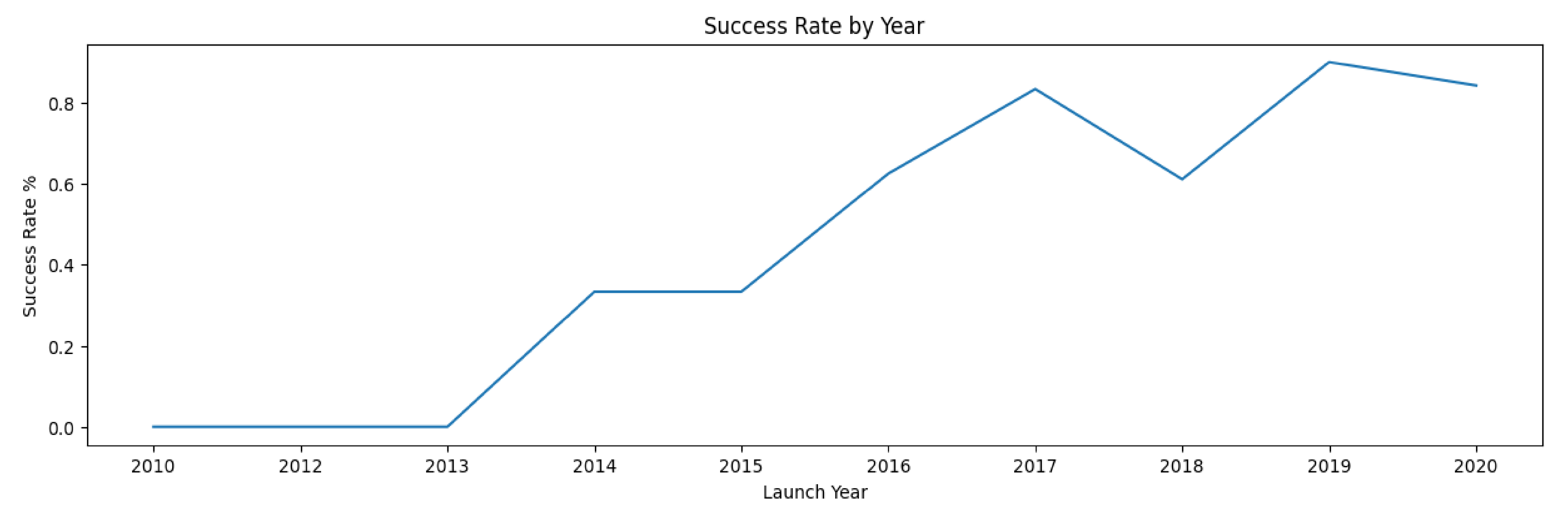
Payload vs. Orbit Type

- With heavy payloads, the success rates are more for PO, LEO, and ISS.
- For GTO, the success rate are mixed.
- SSO only has payload mass on and below 4000 with 100% success rate.
- VLEO has the most heavier payloads with the heaviest one failed.



Launch Success Yearly Trend

- From 2010 to 2013, the success rate is zero.
- Since 2013, the success rate kept increasing.



All Launch Site Names

- There are four unique launch sites. See below:

```
%sql select distinct Launch_Site from spacextable
```

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

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

Launch Site Names Begin with 'KSC'

- Below is the 5 records with launch site names begin with ‘KSC’
- Pyaload mass range from 2490 to 6070.

```
%sql select * from spacextable where launch_site like 'KSC%' limit 5
```

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

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_	Orbit	Customer	Mission_Outcome	Landing_Outcome
2017-02-19	14:39:00	F9 FT B1031.1	KSC LC-39A	SpaceX CRS-10	2490	LEO (ISS)	NASA (CRS)	Success	Success (ground pad)
2017-03-16	6:00:00	F9 FT B1030	KSC LC-39A	EchoStar 23	5600	GTO	EchoStar	Success	No attempt
2017-03-30	22:27:00	F9 FT B1021.2	KSC LC-39A	SES-10	5300	GTO	SES	Success	Success (drone ship)
2017-05-01	11:15:00	F9 FT B1032.1	KSC LC-39A	NROL-76	5300	LEO	NRO	Success	Success (ground pad)
2017-05-15	23:21:00	F9 FT B1034	KSC LC-39A	Inmarsat-5 F4	6070	GTO	Inmarsat	Success	No attempt

Total Payload Mass

- The total payload carried by boosters from NASA is 45596.

```
%sql select sum(PAYLOAD_MASS__KG_) from spacextable where customer = 'NASA (CRS)'
```

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

sum(PAYLOAD_MASS__KG_)
45596

Average Payload Mass by F9 v1.1

- There are five payload mass carried by booster version F9 V1.1.
- The average payload mass is 2928.4.

```
%sql select avg(PAYLOAD_MASS__KG_), count(*) from spacextable where booster_version = 'F9 v1.1'
```

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

avg(PAYLOAD_MASS__KG_)	count(*)
2928.4	5

First Successful Ground Landing Date

- There are 14 success landing from drone ship from 2016 to 2018.

```
%sql select Date, landing_outcome from spacextable where landing_outcome ='Success (drone ship)'
```

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

Date	Landing_Outcome
2016-04-08	Success (drone ship)
2016-05-06	Success (drone ship)
2016-05-27	Success (drone ship)
2016-08-14	Success (drone ship)
2017-01-14	Success (drone ship)
2017-03-30	Success (drone ship)
2017-06-23	Success (drone ship)
2017-06-25	Success (drone ship)
2017-08-24	Success (drone ship)
2017-10-09	Success (drone ship)
2017-10-11	Success (drone ship)
2017-10-30	Success (drone ship)
2018-04-18	Success (drone ship)
2018-05-11	Success (drone ship)

Successful ground pad Landing with Payload between 4000 and 6000

- There are three boosters which have successfully landed on ground pad and had payload mass greater than 4000 but less than 6000

-

```
%sql select booster_version, landing_outcome, PAYLOAD_MASS__KG_ from spacextable where landing_outcome = 'Success (ground pad)' and PAYLOAD_MASS__KG_ > 4000 and PAYLOAD_M
```

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

Booster_Version	Landing_Outcome	PAYLOAD_MASS__KG_
F9 FT B1032.1	Success (ground pad)	5300
F9 B4 B1040.1	Success (ground pad)	4990
F9 B4 B1043.1	Success (ground pad)	5000

Total Number of Successful and Failure Mission Outcomes

- In total, there are 61 successful mission outcomes and 10 failure mission outcomes.

```
%sql select sum(case when landing_outcome  like 'Success%' then 1 else 0 end) as Success, sum(case when landing_outcome like 'Failure%' then 1 else 0 end) as Failure from
```

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

Success	Failure
61	10

Boosters Carried Maximum Payload

- The max payload mass is 15600.
- There are 12 boosters that carried it from 2019 to 2020.

```
%sql select * from spacextable where PAYLOAD_MASS__KG_ = (select max(PAYLOAD_MASS__KG_) from spacextable)
```

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

Date	Time (UTC)	Booster_Version	Launch_Site	Payload	PAYLOAD_MASS__KG_
2019-11-11	14:56:00	F9 B5 B1048.4	CCAFS SLC-40	Starlink 1 v1.0, SpaceX CRS-19	15600
2020-01-07	2:33:00	F9 B5 B1049.4	CCAFS SLC-40	Starlink 2 v1.0, Crew Dragon in-flight abort test	15600
2020-01-29	14:07:00	F9 B5 B1051.3	CCAFS SLC-40	Starlink 3 v1.0, Starlink 4 v1.0	15600
2020-02-17	15:05:00	F9 B5 B1056.4	CCAFS SLC-40	Starlink 4 v1.0, SpaceX CRS-20	15600
2020-03-18	12:16:00	F9 B5 B1048.5	KSC LC-39A	Starlink 5 v1.0, Starlink 6 v1.0	15600
2020-04-22	19:30:00	F9 B5 B1051.4	KSC LC-39A	Starlink 6 v1.0, Crew Dragon Demo-2	15600
2020-06-04	1:25:00	F9 B5 B1049.5	CCAFS SLC-40	Starlink 7 v1.0, Starlink 8 v1.0	15600
2020-09-03	12:46:14	F9 B5 B1060.2	KSC LC-39A	Starlink 11 v1.0, Starlink 12 v1.0	15600
2020-10-06	11:29:34	F9 B5 B1058.3	KSC LC-39A	Starlink 12 v1.0, Starlink 13 v1.0	15600
2020-10-18	12:25:57	F9 B5 B1051.6	KSC LC-39A	Starlink 13 v1.0, Starlink 14 v1.0	15600
2020-10-24	15:31:34	F9 B5 B1060.3	CCAFS SLC-40	Starlink 14 v1.0, GPS III-04	15600
2020-11-25	2:13:00	F9 B5 B1049.7	CCAFS SLC-40	Starlink 15 v1.0, SpaceX CRS-21	15600

2015 Launch Records

- Below is the month names, succesful landingoutcomes in ground pad ,booster versions, launchsite for the months in year 2017

```
%sql select case strftime('%m', date) when '02' then 'Feb' when '05' then 'May' when '06' then 'Jun' when '08' then 'Aug' when '09' then 'Sep' when '12' then 'Dec' else d
    booster_version, launch_site, date, landing_outcome from spacextable where strftime('%Y', date)  = '2017' and landing_outcome = 'Success (ground pad)' order by Date
```

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

Mth_name	Booster_Version	Launch_Site	Date	Landing_Outcome
Feb	F9 FT B1031.1	KSC LC-39A	2017-02-19	Success (ground pad)
May	F9 FT B1032.1	KSC LC-39A	2017-05-01	Success (ground pad)
Jun	F9 FT B1035.1	KSC LC-39A	2017-06-03	Success (ground pad)
Aug	F9 B4 B1039.1	KSC LC-39A	2017-08-14	Success (ground pad)
Sep	F9 B4 B1040.1	KSC LC-39A	2017-09-07	Success (ground pad)
Dec	F9 FT B1035.2	CCAFS SLC-40	2017-12-15	Success (ground pad)

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

- Below is the ranking of landing outcomes between the date 2010-06-04 and 2017-03-20, in descending order

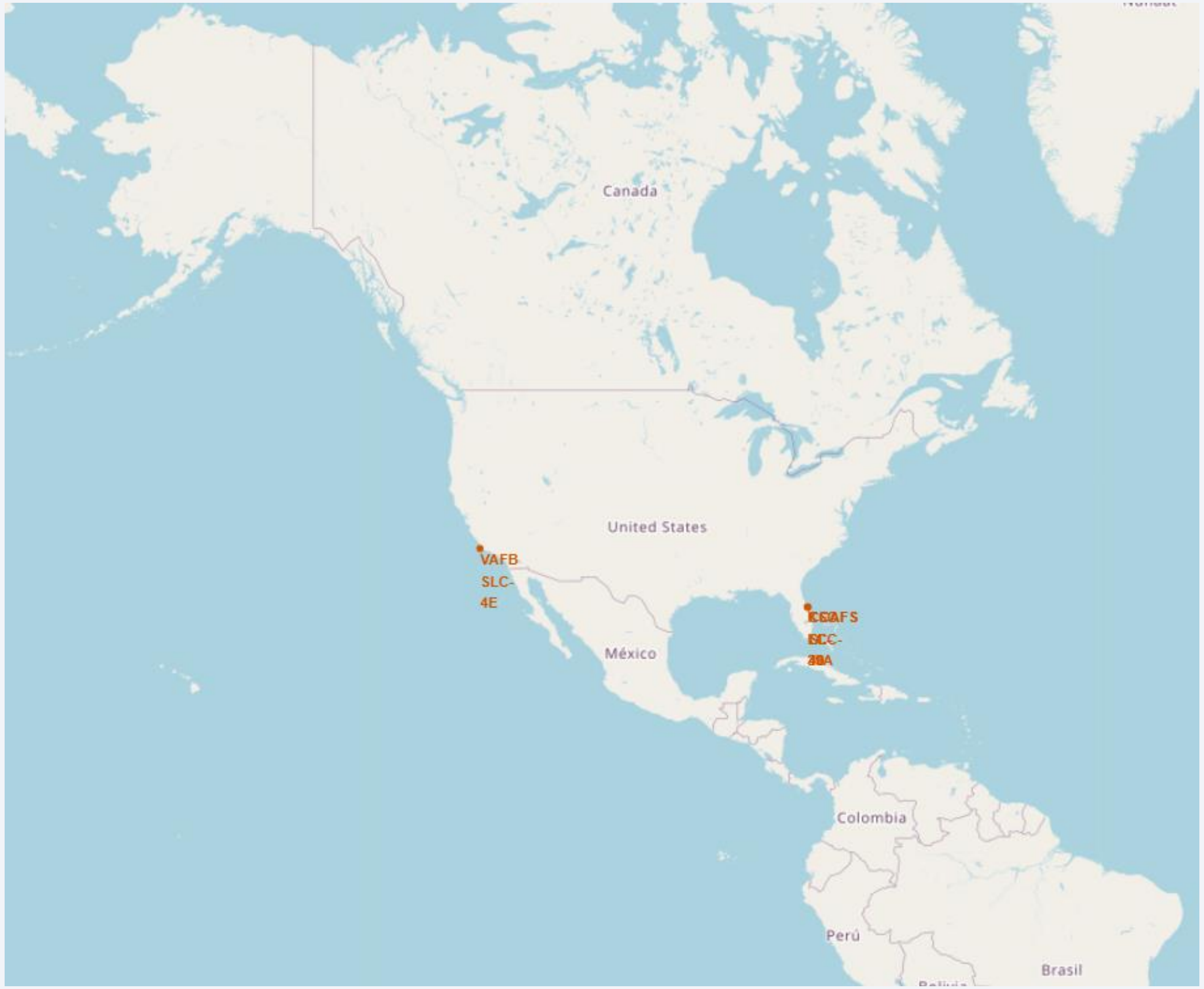
```
%sql select landing_outcome, count(*) from spacextable where date between '2010-06-04' and '2017-03-20' group by landing_outcome order by 2 desc
```

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

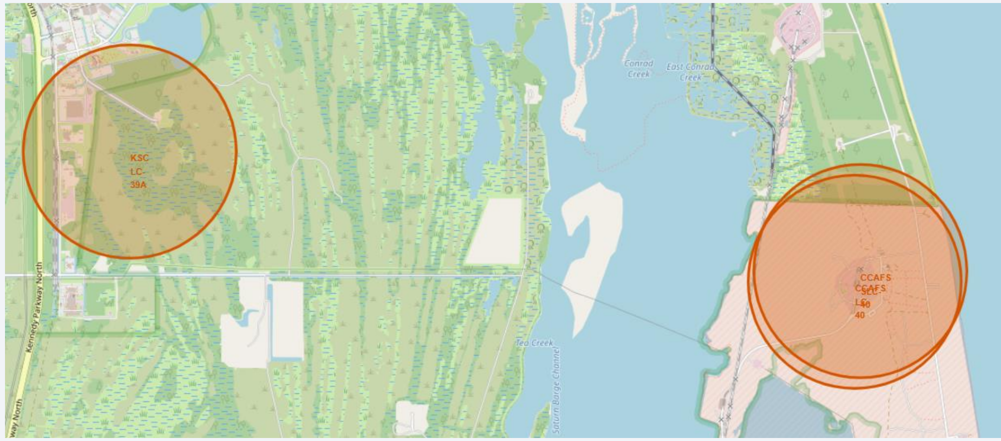
Landing_Outcome	count(*)	min(date)	max(date)
No attempt	10	2012-05-22	2017-03-16
Success (drone ship)	5	2016-04-08	2017-01-14
Failure (drone ship)	5	2015-01-10	2016-06-15
Success (ground pad)	3	2015-12-22	2017-02-19
Controlled (ocean)	3	2014-04-18	2015-02-11
Uncontrolled (ocean)	2	2013-09-29	2014-09-21
Failure (parachute)	2	2010-06-04	2010-12-08
Precluded (drone ship)	1	2015-06-28	2015-06-28


Section 3
Launch Sites
Proximities Analysis

The four launch sites on folium

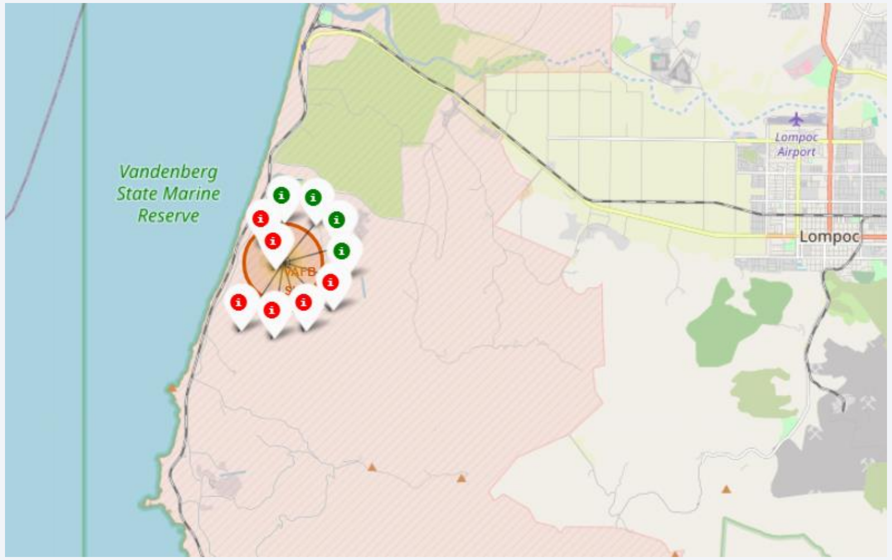


- There are one launch site on US west coast and three launch sites on US east coast (see the map to the left).
- The three launch sites on US east coast are shown as below with a zoom in view.

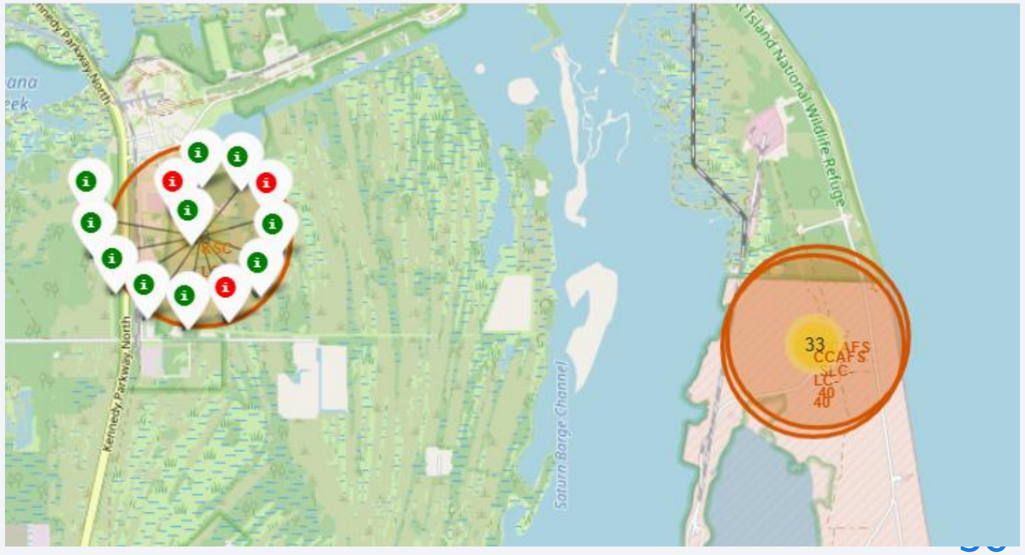


Color-labeled launch outcomes

- Success and failed launch outcomes for each site are showed with color-labeled.
- Below is the color-labeled outcome for VAFB SLC-4E site, which is the site on US west coast.

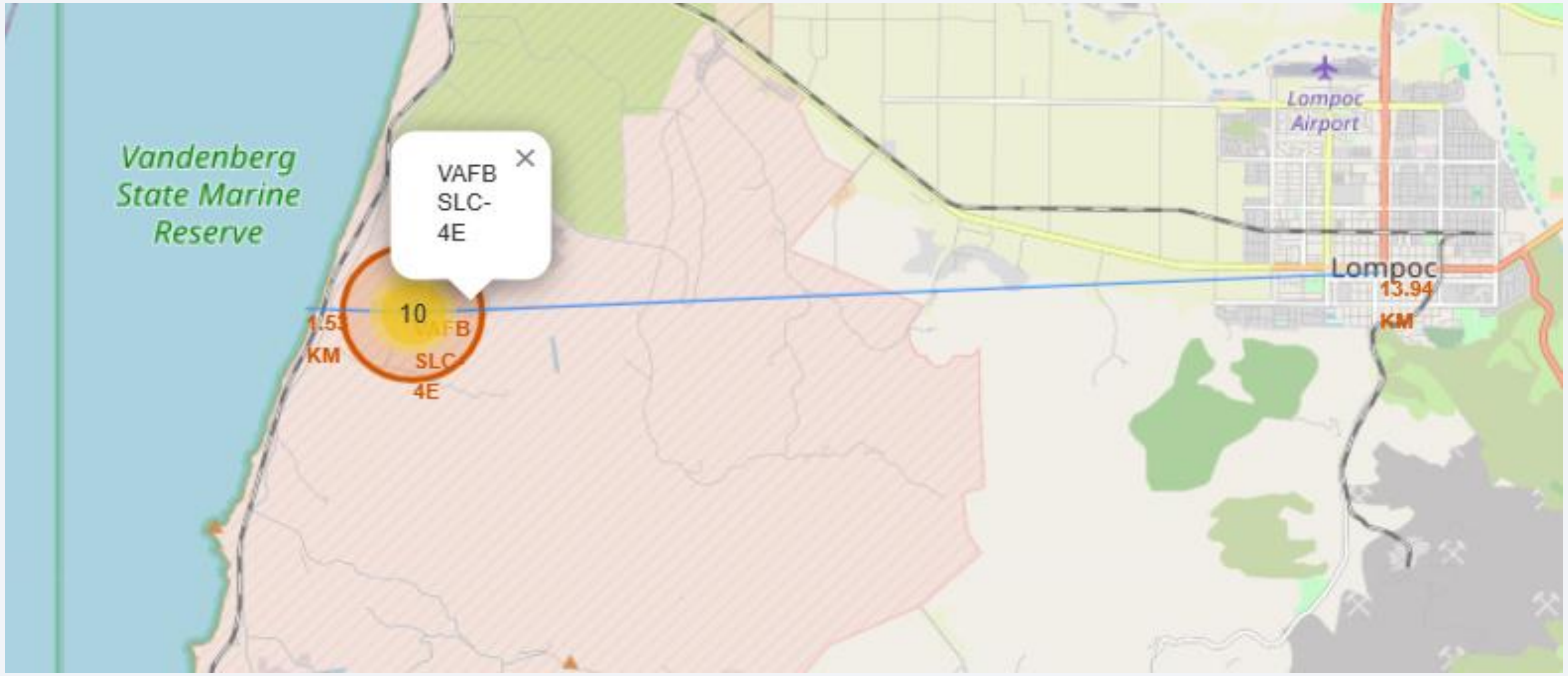


- Below is the color-labeled outcome for KSC LC-39A site, which is one of the sites on US east coast.
- To see the color-label for all sites, go to this page: [Pytrhon-for-Data-Science-Project/006 lab_jupyter_launch_site_location.ipynb at main · herachua10/Pytrhon-for-Data-Science-Project]



Distance between launch site to nearest town and coastline

- The launch site keeps certain distance away from cities and in close proximity to coastline. See the screenshot below regarding VAFB SLC-4E:
  - The distance to Lompoc, the nearest town, is 13.94 KM.
  - The distance to the coastline is 1.53 KM

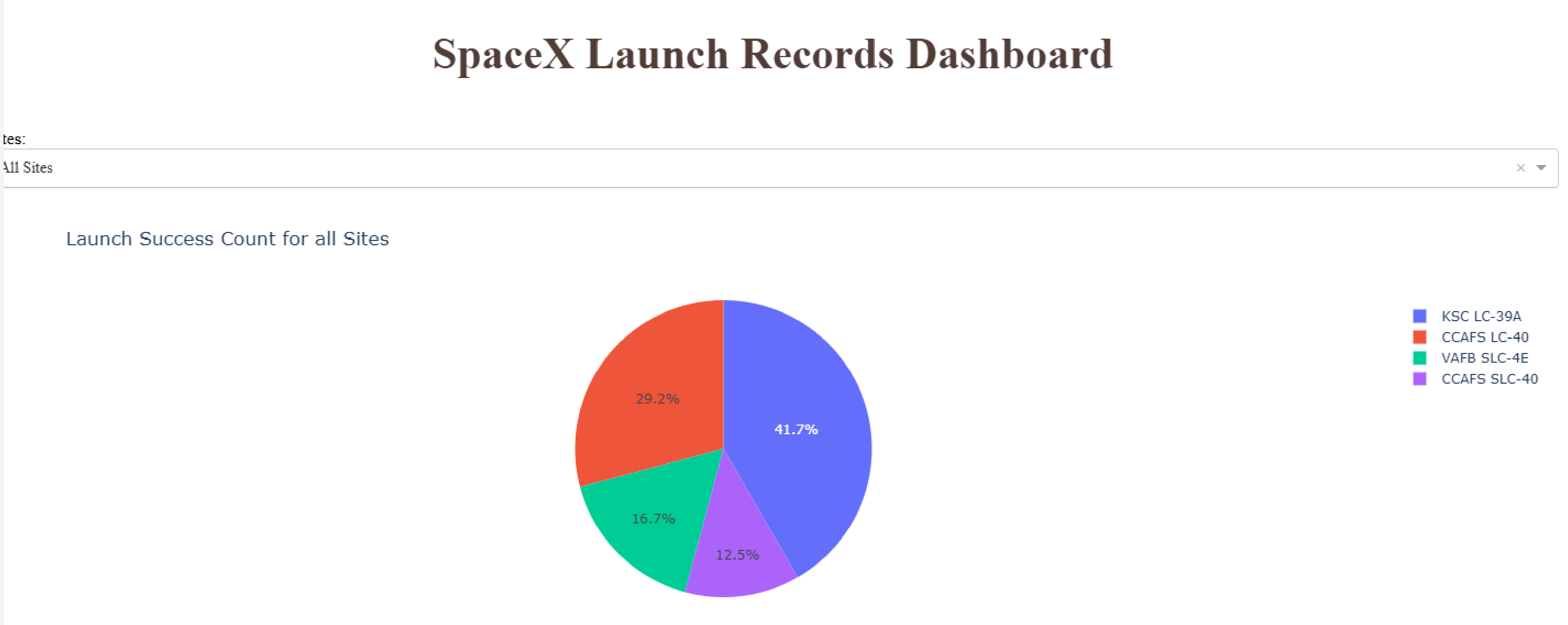


Section 4

Build a Dashboard with Plotly Dash

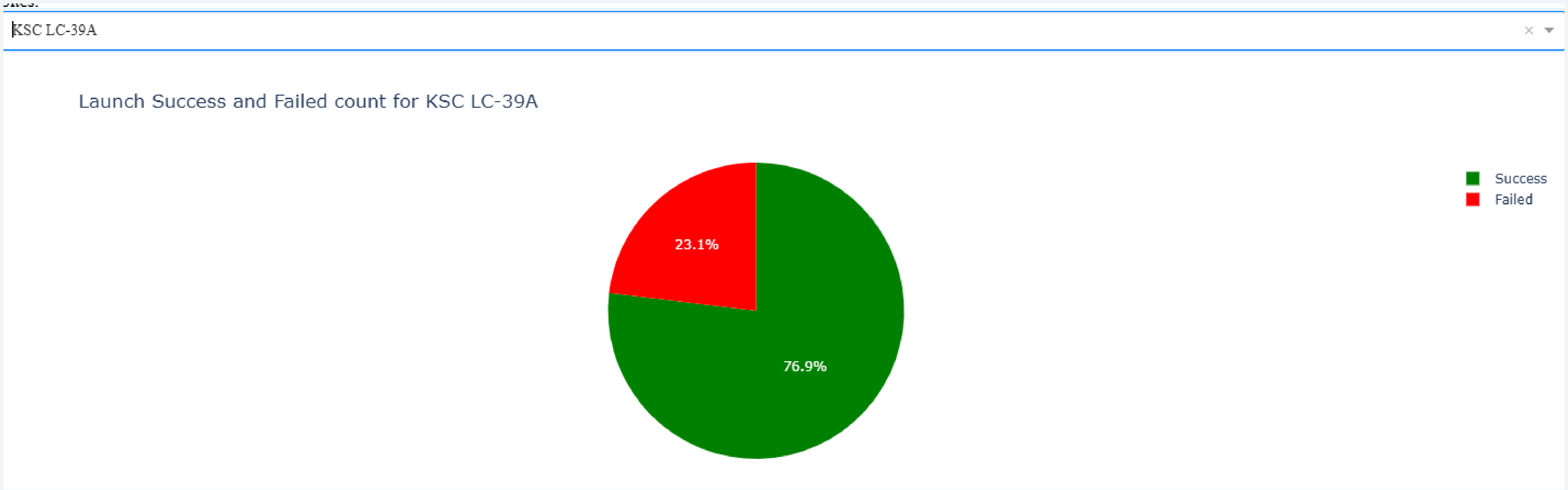
Success Rate for Launch Sites

- By comparing the four launch sites:
  - KSC LC-39A has the highest launch success rate.
  - CCAFS SLC-40 has the lowest launch success rate.



The success and failed rates for KSC LC-39A

- KSC LC-39A is the site with the most successful launch, with 76.9% success rate.



Payload Mass vs Launch Outcome

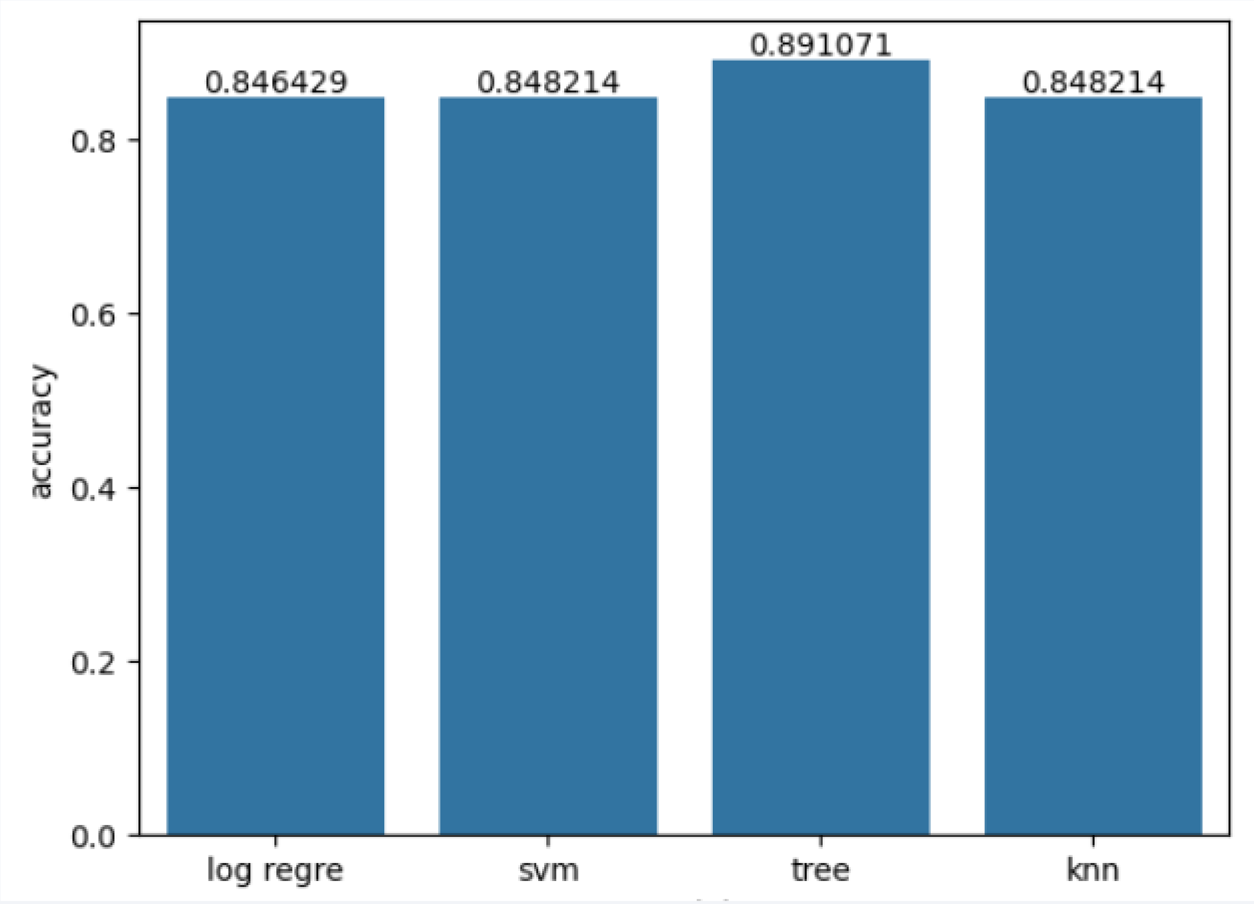
- FT booster has the highest success rate.
- V1.0 never success.
- B4 has the heaviest payload mass.
- The majority of the boosters have payload mass below 6K.

Section 5

Predictive Analysis (Classification)

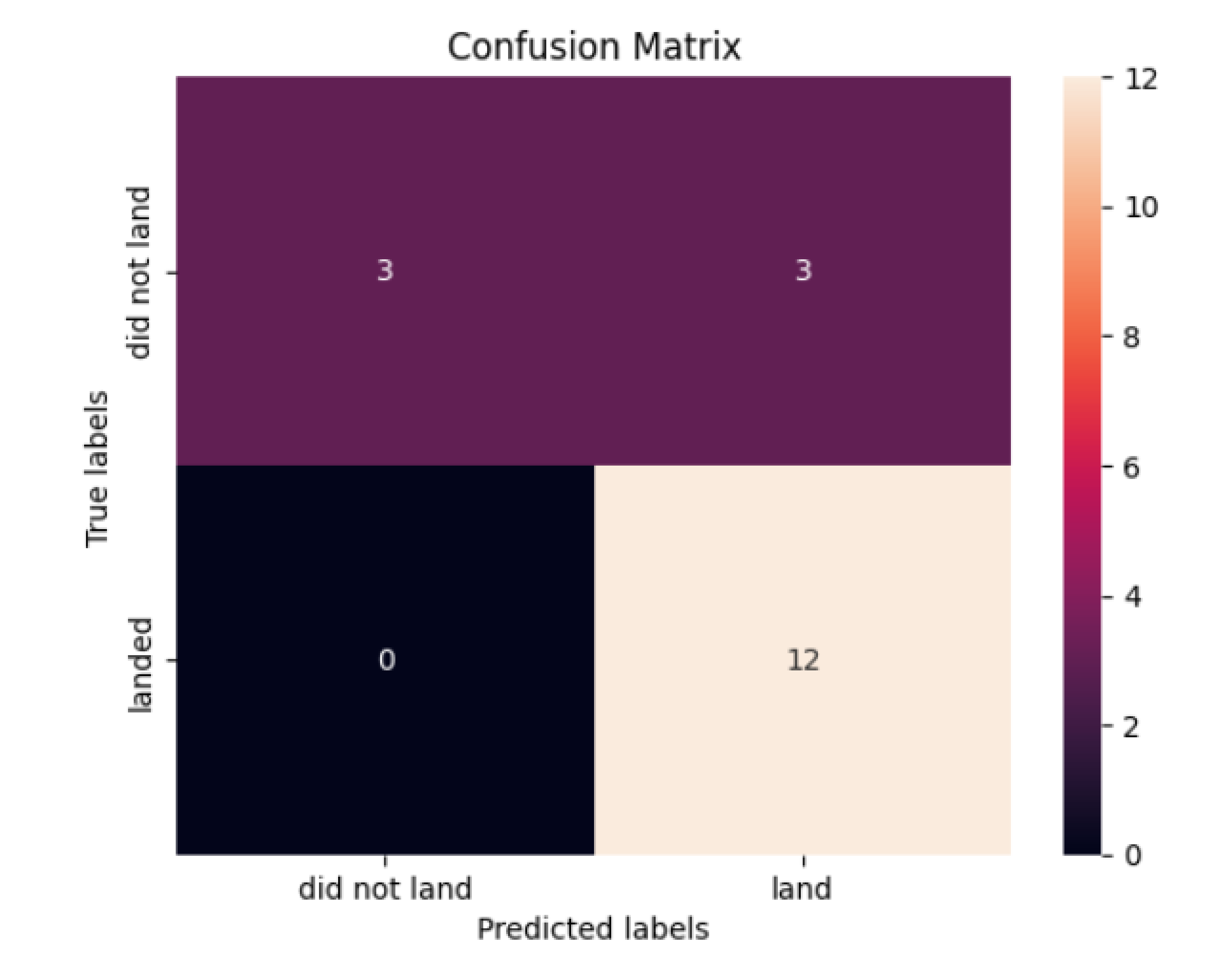
Classification Accuracy

- Decision tree has the highest accuracy of 0.89.



Confusion Matrix

- There are 12 True Positive.
- There are 3 False Positive.



Conclusions

- Most of the launches carried out at US east coast.
- The launch went to eleven orbit types.
- The success rate of launch increased significantly overtime.

Appendix

f1_score for the 4 predicted models.

```
from sklearn import metrics

f1_logre = metrics.f1_score(Y_test, logreg_cv.predict(X_test))
f1_svm = metrics.f1_score(Y_test, svm_cv.predict(X_test))
f1_tree = metrics.f1_score(Y_test, tree_cv.predict(X_test))
f1_knn = metrics.f1_score(Y_test, knn_cv.predict(X_test))

print(f'logistic regression f1_score: {f1_logre:.4f}')
print(f'support vector machine f1_score: {f1_svm:.4f}')
print(f'decision tree f1_score: {f1_tree:.4f}')
print(f'knn f1_score: {f1_knn:.4f}')

print('based on f1_score, both logictic regression, support vector machine, and knn perfom equally well')
```

```
logistic regression f1_score: 0.8889
support vector machine f1_score: 0.8889
decision tree f1_score: 0.7407
knn f1_score: 0.8889
based on f1_score, both logictic regression, support vector machine, and knn perfom equally well
```

Thank you!