IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Kavan J Desai
July 11th, 2023

Outline

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

EXECUTIVE SUMMARY

- **Methodologies used:**
  - Purpose of project was to collect data to determine which factors lead to successful landings for the Space X program
    - Data was collected using API and webscraping and wrangled into pass/fail outcomes
    - The data was then explored using data visualization techniques with various variables, such as payload, launch site, flight number, and yearly trend
    - This dataset was then analyzed using SQL to general statistical analyses on total payload, payload ranges for successful launches,, and total # of successful/failed outcomes
    - Exploratory measures using Folium and Geographical markers were complete along with the creation of dashboards to see which launch sites had the most success with launches and payload
    - Finally, we used machine learning classification techniques, such as logistic regression, support vector machine (SVM, decision trees, and K-nearest neighbor (HNN) to predict outcomes

- Summary of all results
  - Following output methods were used:
    1. Exploratory Data Analysis
    2. Geospatial visualizations
    3. Plotly Dashboards
    4. Classification Confusion Matrix

Introduction

- The SpaceX program was designed to launch Falcon 9 rockets. This program costs considerably much less ($62 million) compared to other providers ($165 million). These savings are achievable, because SpaceX has figured out ways to reuse their first stage launching techniques. Thus, by figuring out which 1st stage launches are successful, we can figure out how much cost it will take for future launches.
  - This cost analysis can help determine if rival companies should bid on SpaceX for their future launches.
- **Key problems addressed in this project:**
  1. **What factors determine the successful landing of a rocket?**
  2. **Interactions of various features that determine the landing of rockets**
  3. **What are the optimal operating conditions that are required for successful launches?**

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology:
  - Data was collected using SpaceX API and webscraping from Wikipedia
- Perform data wrangling
  - Categorical variables were turned into binary data using one-hot encoding
- 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

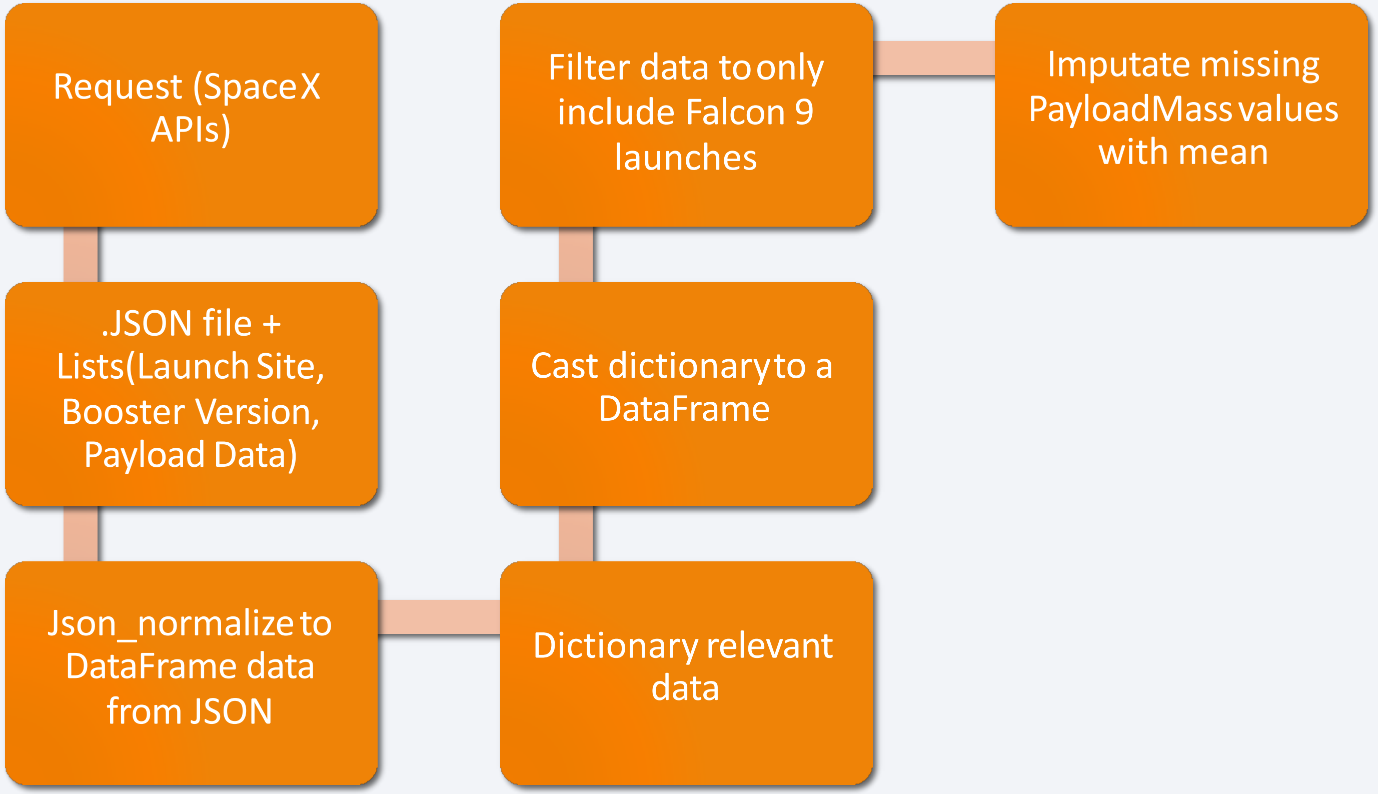
Data Collection

- Datasets were collected using get request to SpaceX Api. Then response content was decoded as Json using python .json() function. This was then placed into a pandas dataframe using .json_normalize()
- Data cleaning was then done to check for missing values, which were in turn filled if found.
- This was followed up with web scraping using BeautifulSoup to find launch records for Falcon 9
- These records were extracted to an HTML table, parsed, and then converted to a pandas dataframe for further analysis.

Data Collection – SpaceX API

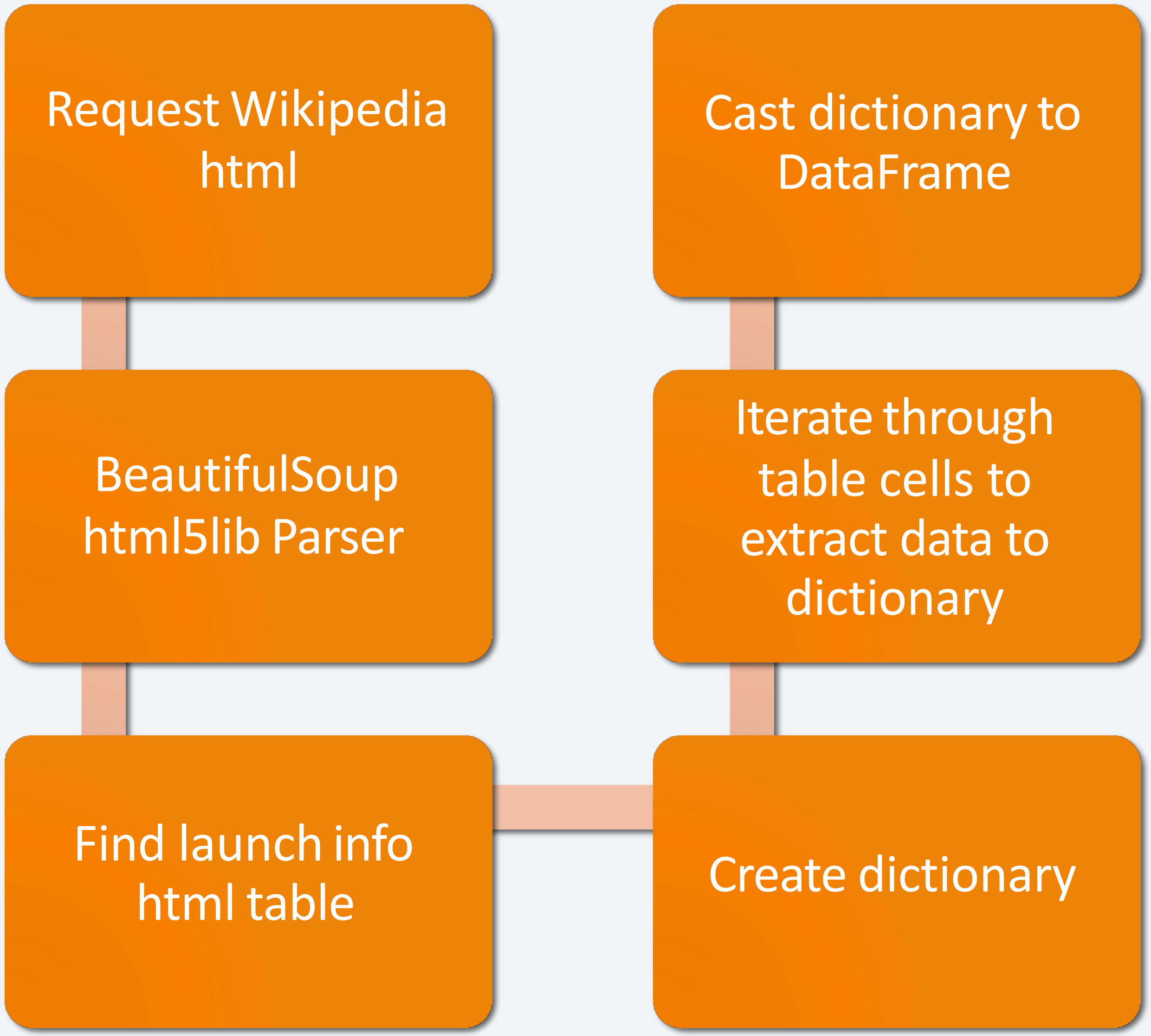
- Data was collected using SpaceX API, cleaned, and then we used based data wrangling and formatting to make it useful for further analysis.

- https://github.com/Kdesai819/IBM-Capstone/blob/main/Week 1%3A Working with Data/Data Collection.ipynb



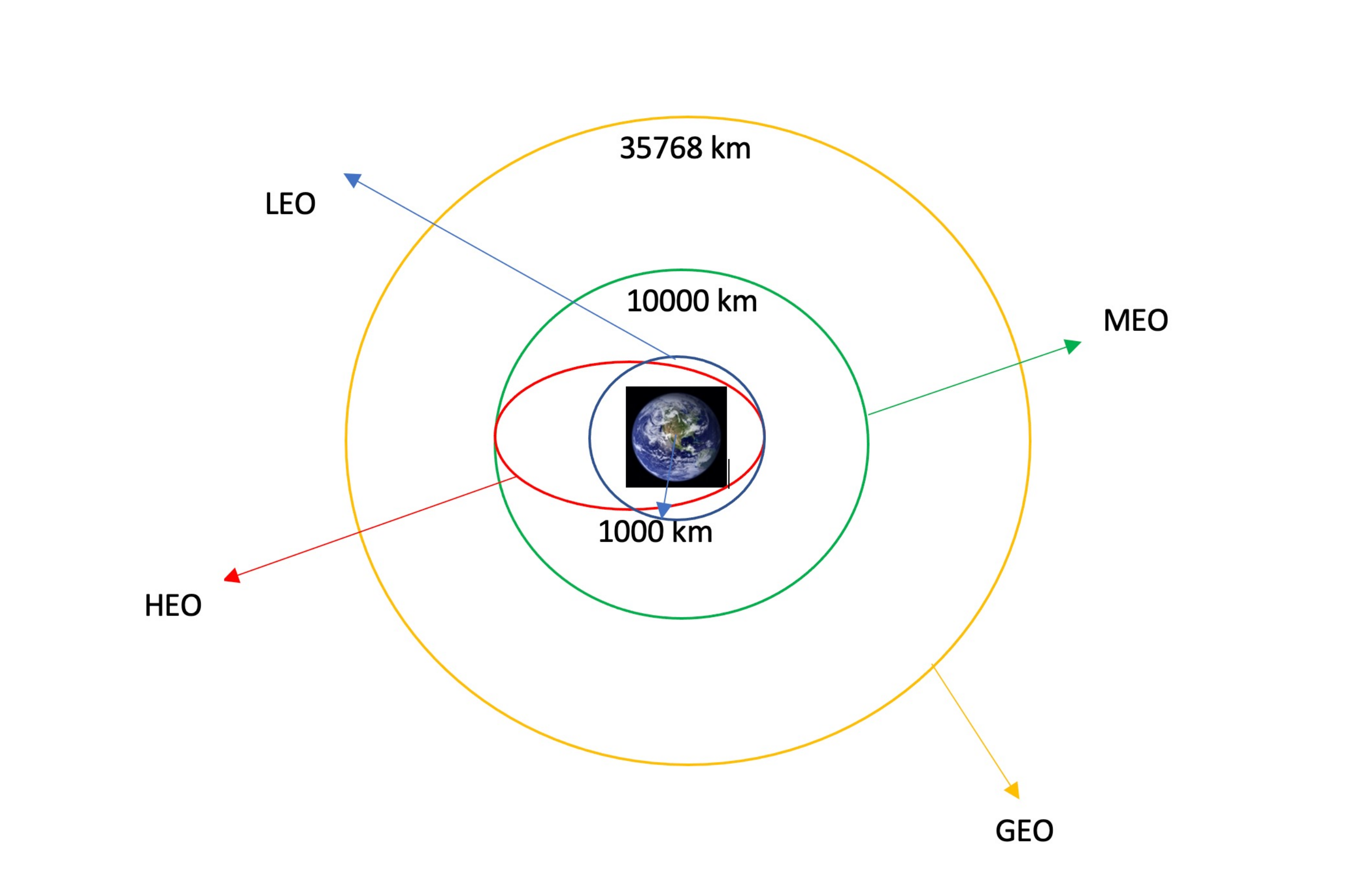
Data Collection - Scraping

- Webscraping was applied to Falcon 9 launch records using BeautifulSoup. Data was then parsed and converted to pandas dataframe.
- https://github.com/Kdesai819/IBM-Capstone/blob/main/Week 1%3A Working with Data/Data Webscraping.ipynb



Data Wrangling

- Exploratory data analysis preformed and determined labels used to train data.
- Number of launches at each site were calculated and then number and occurrence of each orbit was calculated.
- A landing outcome label was created and the results were then exported to csv
- https://github.com/Kdesai819/IBM-Capstone/blob/main/Week 1%3A Working with Data/Data Wrangling.ipynb



EDA with Data Visualization

- With data visualization techniques, we explored the SpaceX data to find relationships that existed between flight number and launch site, payload mass and launch site, the success rate of each orbit time, flight number and orbit type, and the launch success yearly trend
- Scatter plots, bar charts, and line plots were used in order to compare variables and determine if relationships existed that could then be used to train a machine learning model
- https://github.com/Kdesai819/IBM-Capstone/blob/main/Week 2%3A Exploratory Data Analysis/EDA with Visualization.ipynb

EDA with SQL

- The following SQL queries were performed

1. Display the names of the unique launch sites in the space mission
2. Display 5 records where launch sites begin with the string ‘CCA’
3. Display the total payload mass carried by boosters launched by NASA (CRS)
4. Display the average payload mass carried by booster version F9 v1.1
5. List the date when the first successful landing outcome on a ground pad was achieved
6. List the names of the boosters which had success on a drone ship and a payload mass between 4000 and 6000 kg
7. List the total number of successful and failed mission outcomes
8. List the names of the booster versions which have carried the maximum payload mass
9. List the failed landing outcomes on drone ships, their booster versions, and launch site names for 2015
10. Rank the count of landing outcomes (such as Failure (drone ship) or Success (ground pad)) between the date 2010-06-04 and 2017-03-20, in descending order

- https://github.com/Kdesai819/IBM-Capstone/blob/main/Week 2%3A

Build an Interactive Map with Folium

- The following steps were taken using Folium

1. Launch Site marked on map using the Folium Map object
   1. We then used the functions folium.Circle and folium.Marker to add each launch site to our map

2. Success and failed launches for each site were marked on the map
   - Since many launches share coordinates, it makes perfect to sense to cluster them together.
   - Launches were classified into 2 classes" class 0 (failure) and class 0 (success). Colors were then assigned for the classes: green for class 1 and red for class 0.
   - Launches were clustered for each launch by adding folium.Marker to the MarkerCluster() object.
   - An icon with a text label was then created. The icon color as the marker color that we determined earlier were then added.

3. Calculate the distances between a launch site to its proximities
   - In order to explore the proximities of the launch sizes to various points, we calculated distances between points of interest by obtain the latitude and longitudes of their locations.
   - Using folium.Marker,, we created a marker point for this location in order to show the distance.
   - To display the distance line between two points, we drew a folium.PolyLine and add this to the map.

- https://github.com/Kdesai819/IBM-Capstone/blob/main/Week 3%3A Interactive Visual analysis/SpaceX folium.ipynb

Build a Dashboard with Plotly Dash

- Plotly Dash was used in order to view, analyze, and interact with the data
- We used a pie chart and a scatter plot to visualize the data
  - A pie was used because it allows us to see clearly which launch size was the most successful with its launches. We are also able to select individual launch sites as well to assess the ratio of successful vs failed launches as well.
  - A scatter plot was created observing the relationship of outcome vs payload mass using a payload slider. This was done by filtering through the booster version.
- https://github.com/Kdesai819/IBM-Capstone/blob/main/Week 3%3A Interactive Visual analysis/spacex_dash_app.py

Predictive Analysis (Classification)

- Data was loaded using both NumPy and pandas, then transformed. This data was then split into train and test samples. This was achieved using the train_test_split() function.
- Our goal was to test different prediction models to observe which machine learning algorithm is the most appropriate for our data.
  - 4 types of algorithms were chosen: Logistic Regression, Support Vector Machines (SVM), Decision Trees, and K-Nearest Neighbor (KNN). GridSearchCV object was created for each algorithm and a directory of optimal values was chosen (cv=10).
    - The object was then fitted to these chosen parameters and the training data set was used to train the model.
      - For each output, we checked the tuned hyperparameters using best_params,. The accuracy of each parameter was then checked using score and best_score. Finally, the confusion matrix as then plotted and examined.
      - After reviewing each accuracy score, the score with the high prediction score was deemed as the best prediction model to use.
- https://github.com/Kdesai819/IBM-Capstone/blob/main/Week 4%3A Machine Learning Predictions/Machine Learning Predictions.ipynb

Results

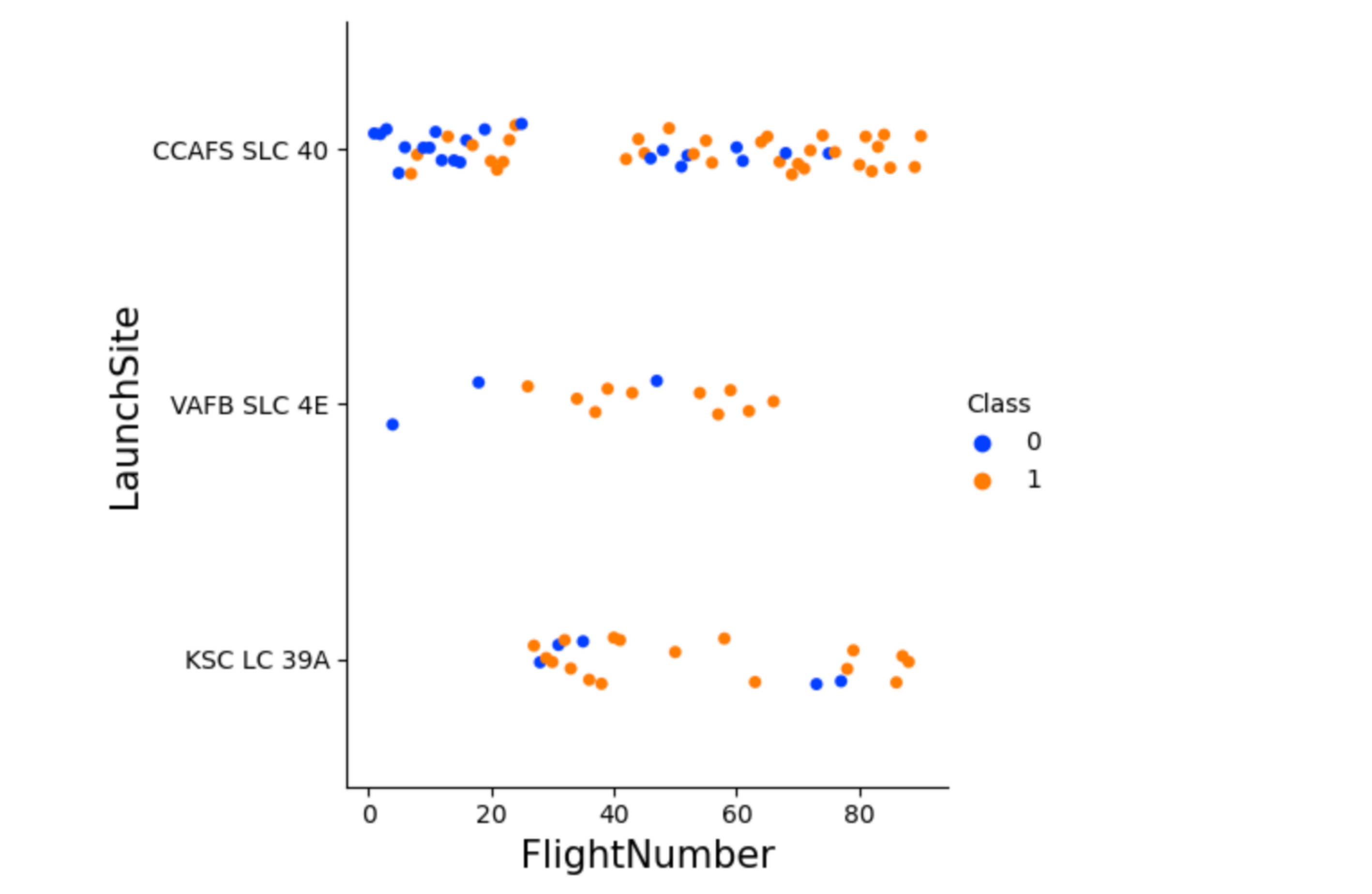
- Exploratory data analysis results
- Interactive analytics demo in screenshots
- Predictive analysis results

Section 2

Insights drawn from EDA

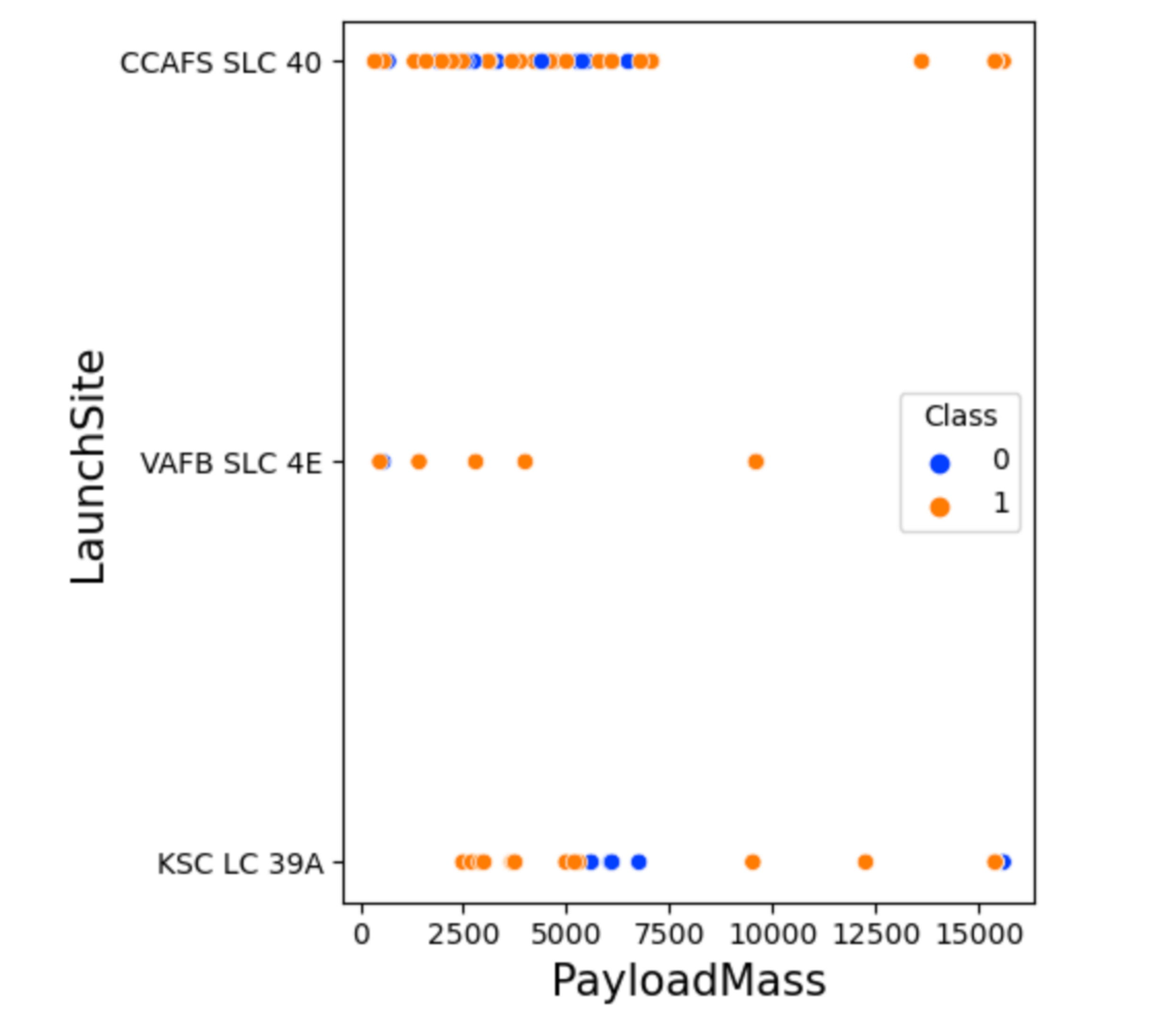
Flight Number vs. Launch Site

- AS WE CAN SEE FOR OUR SCATTER PLOT, AS WE SEE AN INCREASE IN THE NUMBER OF FLIGHTS FOR A LAUNCH SITE, THE GREATER THE NUMBER OF SUCCESSFUL LAUNCHES SEEN FOR THAT LAUNCH SITE. THIS IS ESPECIALLY SEEN AT FLGHTNUMBER>=30
- CCAFS SLC40 HAS THE MOST EARLY FLIGHT (FLIGHTNUMBER<30). MOST OF THESE EARLY FLIGHT ARE ALSO UNSUCCESSFUL FLIGHTS. THIS TREND IS ALSO SEEN WITH THE LAUNCH SITE VAFB SLC 4E.
- THERE ARE NO EARLY FLIGHTS SEEN FOR NO EARLY FLIGHTS WERE LAUNCHED FROM KSC LC 39A, SO THE LAUNCHES FROM THIS SITE ARE MORE SUCCESSFUL.



Payload vs. Launch Site

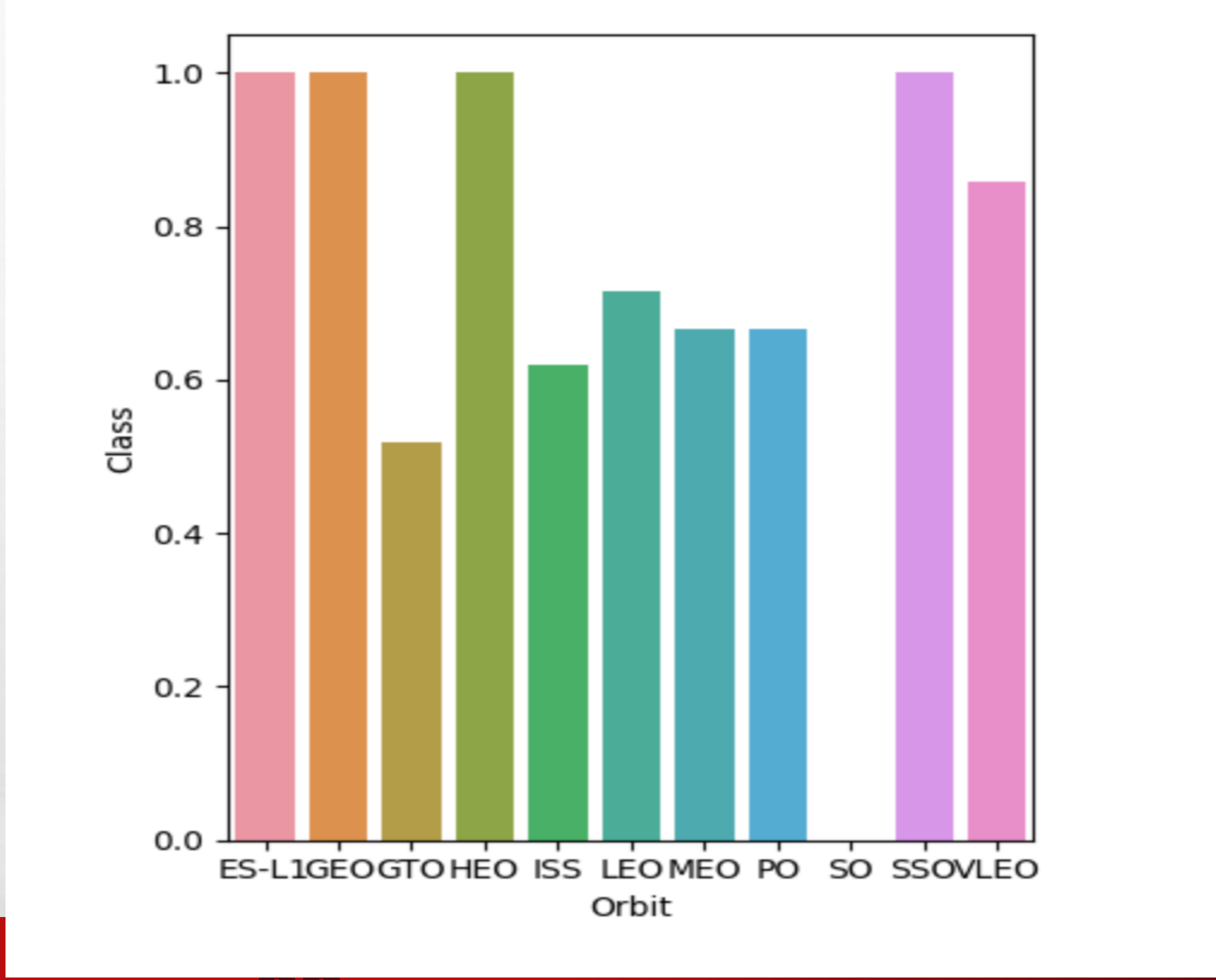
- AS WE CAN SEE IN THE SCATTERPLOT, THERE IS VERY LITTLE DATA FOR LAUNCH STATUS AT PAYLOADS>7000
- WE CAN ALSO SEE THAT THERE DOESN'T REALLY SEEM TO BE ANY CORRELATION BETWEEN PAYLOAD MASS AND RATE OF SUCCESS FOR ANY GIVEN LAUNCH SITE.
- ALL SITES LAUNCHED A VARIETY OF PAYLOAD MASSES, WITH MOST OF THE LAUNCHES FROM CCAFS SLC 40 BEING COMPARATIVELY LIGHTER PAYLOADS (WITH SOME OUTLIERS).



Success Rate vs. Orbit Type

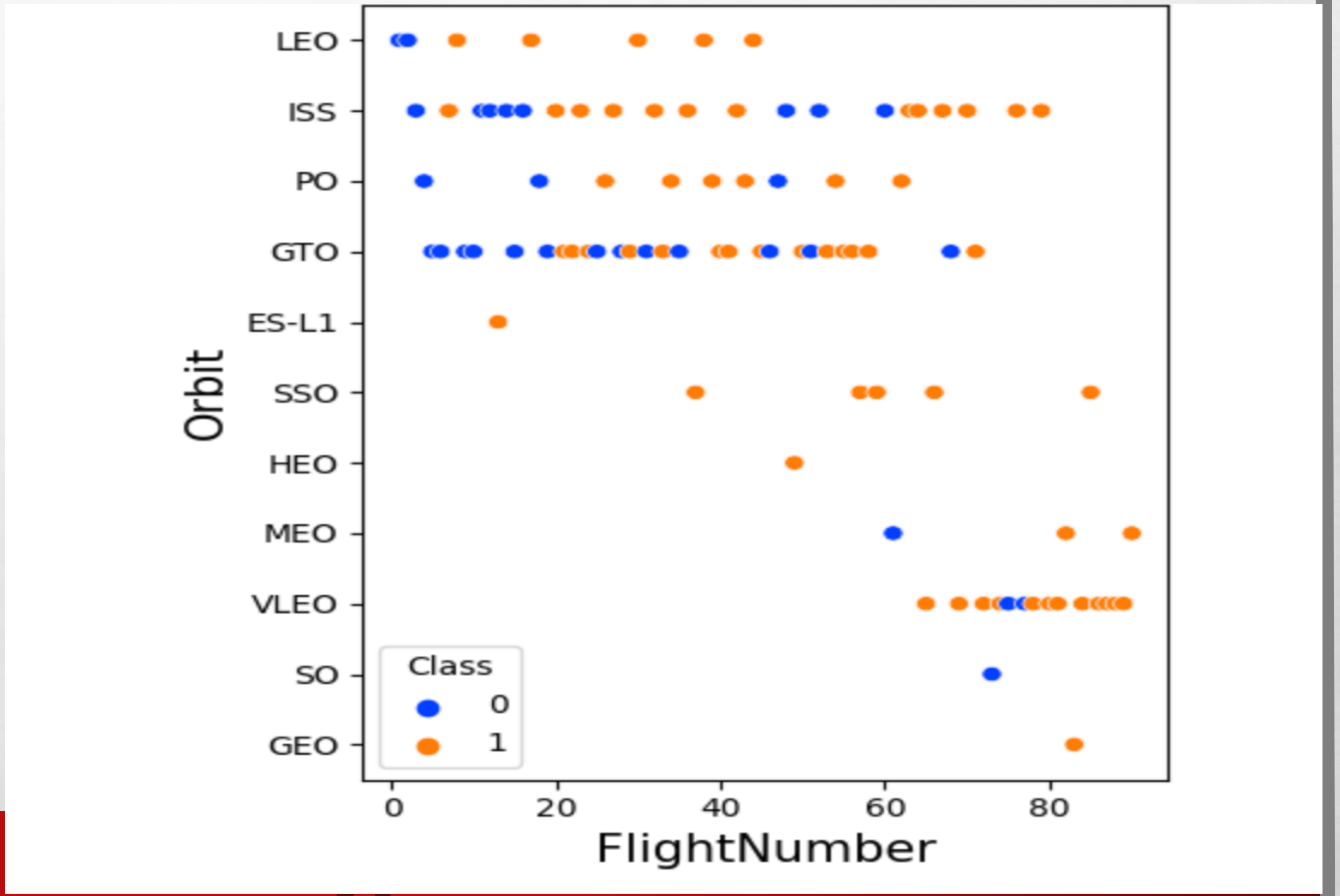
- SITES WITH THE HIGHEST SUCCESS RATES (100%) ARE:
- ES-L1 (EARTH-SUN FIRST LAGRANGIAN POINT)
- GEO (GEOSTATIONARY ORBIT)
- HEO (HIGH EARTH ORBIT)
- SSO (SUN-SYNCHRONOUS ORBIT)

- THE ORBIT WITH THE LOWEST (0%) SUCCESS RATE IS:
- SO (HELIOCENTRIC ORBIT)



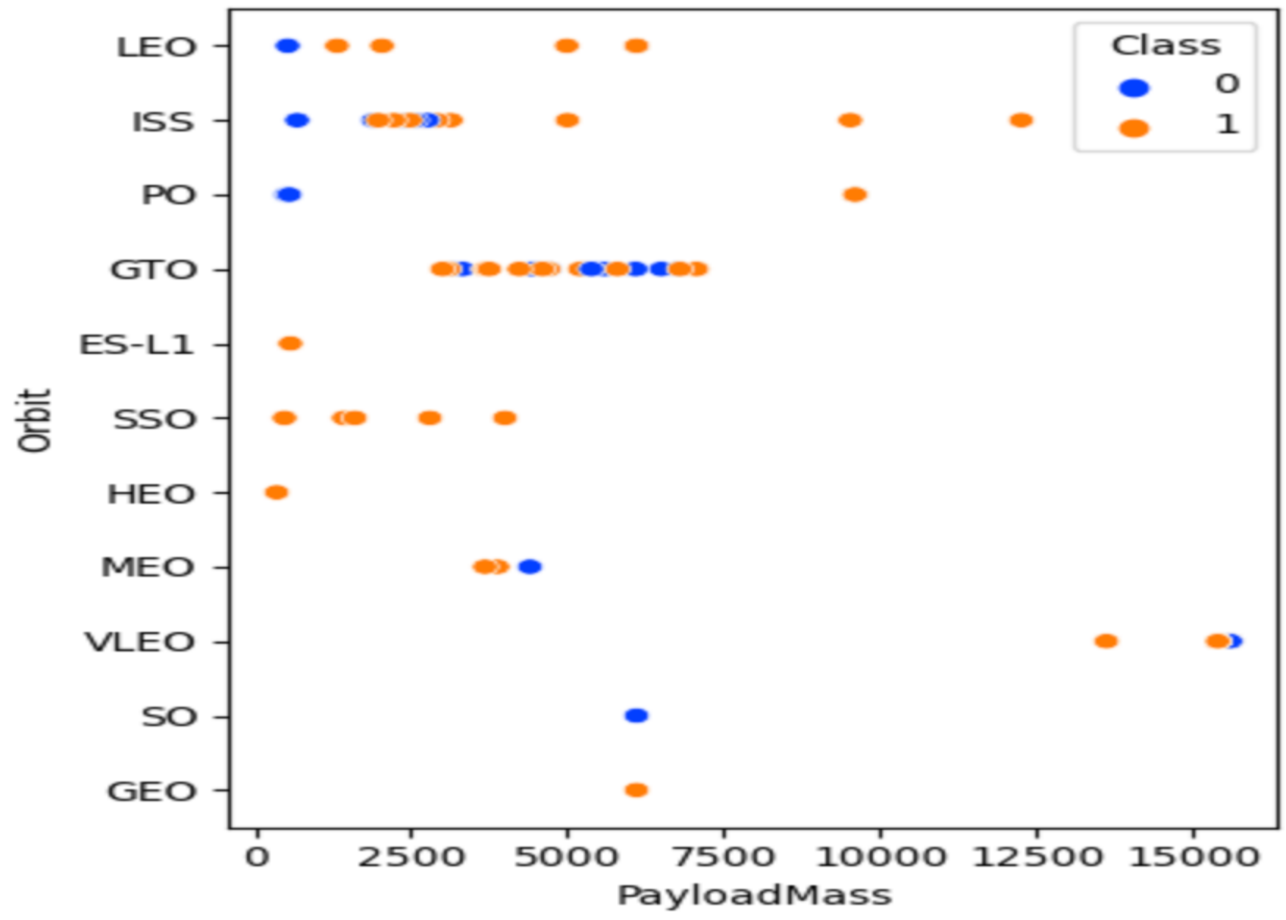
Flight Number vs. Orbit Type

- THIS SCATTERPLOT GIVES US MANY INSIGHTS THAT WE DID RECEIVE FROM THE PREVIOUS PLOTS.
  - THE 100% SUCCESS RATE OF THE GEO, ES-L1, AND HEO ORBITS FROM THE PREVIOUS SLIDE CAN BE ATTRIBUTED TO THE FACT THAT THERE WAS ONLY 1 LAUNCH FOR THESE ORBITS.
  - THE 100% SUCCESS RATE OF SSO IS MOST IMPRESSIVE DUE TO THE OBSERVATION OF 5 SUCCESSFUL LAUNCHES.
  - GTO SHOWS THAT THERE IS LITTLE RELATIONSHIP BETWEEN SUCCESS RATE AND FLIGHT NUMBER
  - GENERALLY, AS FLIGHT NUMBER INCREASES, THE SUCCESS RATE INCREASES. THIS IS MOST EXTREME FOR LEO, WHERE UNSUCCESSFUL LANDINGS ONLY OCCURRED FOR THE LOW FLIGHT NUMBERS (EARLY LAUNCHES).



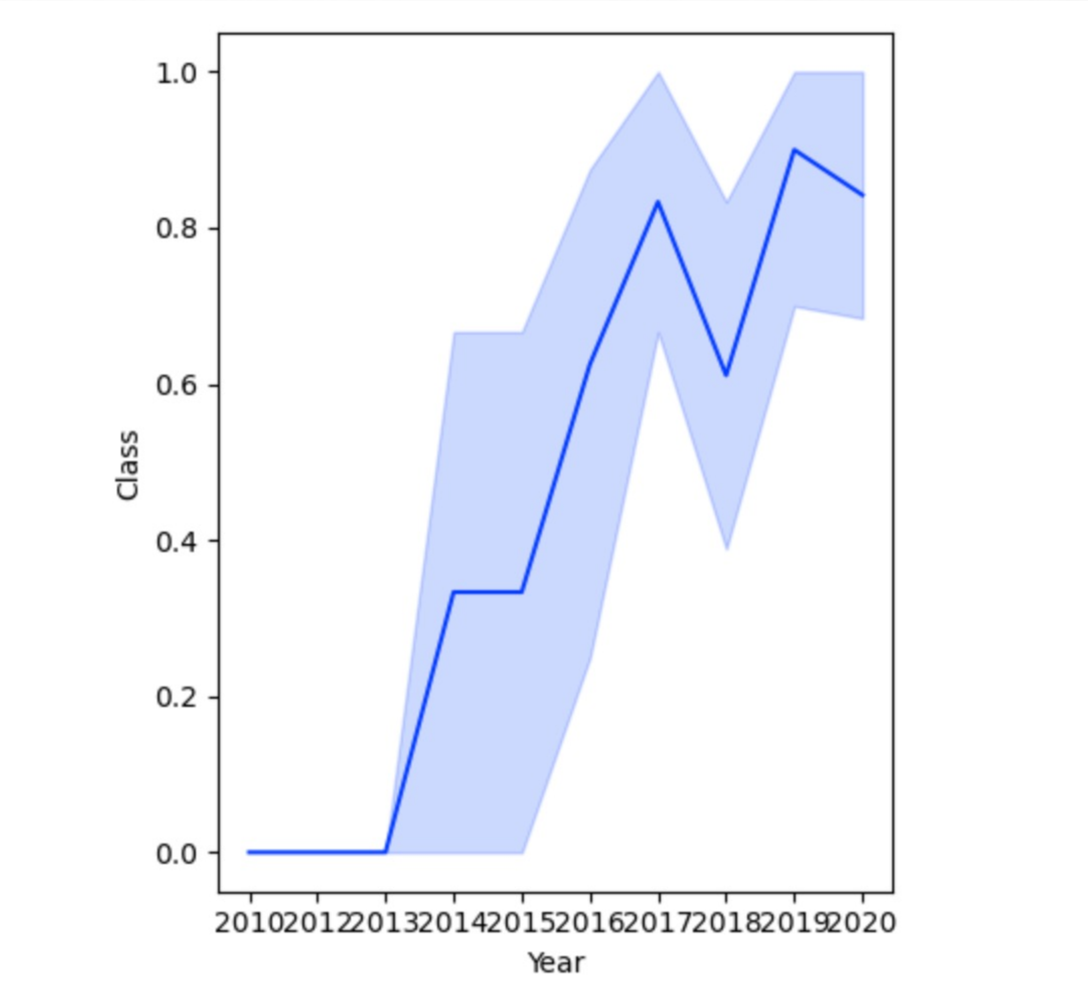
Payload vs. Orbit Type

- BASED ON THE SCATTERPLOT, THE FOLLOWING ORBITS HAVE THE HIGHEST SUCCESS RATE WITH HEAVY PAYLOADS: PO (DATA IS LOW THOUGH), ISS, AND LEO
- RELATIONSHIP BETWEEN SUCCESS RATE AND PAYLOADMASS IS UNCLEAR FOR GEO.



Launch Success Yearly Trend

- SUCCESS RATES ARE PREDOMINATELY UNSUCCESSFUL FOR YEARS 2010 T0 2013.
- AFTER 2013, WE SEE A RISING TREND IN SUCCESS RATES DESPITE A DIP IN 2018 AND 2020.
- POST 2016, SUCCESS RATE HAS BEEN MAINTAINED ABOVE 60%



All Launch Site Names

```
In [43]: %sql select DISTINCT LAUNCH_SITE from SPACEXTBL;

 * ibm_db_sa://wlh40668:***@1bbf73c5-d84a-4bb0-85b9-ab1a4
86/bludb
   sqlite:///my_data1.db
Done.
```

Out[43]:

launch_site
CCAFS LC-40
CCAFS SLC-40
KSC LC-39A
VAFB SLC-4E
None

- WE USED DISTINCT SQL FUNCTION TO SEE THE LAUNCH SITES UNIQUE TO THE SPACE X PROGRAM

Launch Site Names Begin with 'CCA'

```
[44]: %sql select * from SPACEXtbl where launch_site like 'CCA%' limit 5
```

```
 * ibm_db_sa://wlh40668:***@1bbf73c5-d84a-4bb0-85b9-ab1a4348f4a4.c3n41cmd0nqnrk39u98g.databases.appdomain.cl
286/bludb
   sqlite:///my_data1.db
Done.
```

[44]:

DATE	time__utc_	booster_version	launch_site	payload	payload_mass__kg_	orbit	customer	mission_outcome	Landing outcome
2010-04-06	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
2010-08-12	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	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
2012-08-10	0:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
2013-01-03	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

- LIMIT 5 FETCHES ONLY 5 RECORDS, AND THE LIKE KEYWORD IS USED WITH THE WILD CARD 'CCA%' TO RETRIEVE STRING VALUES BEGINNING WITH 'CCA'.

Total Payload Mass

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

```
[45]: %sql select sum(payload_mass__kg_) as sum from SPACEXTBL where customer like 'NASA (CRS)';

 * ibm_db_sa://wlh40668:***@1bbf73c5-d84a-4bb0-85b9-ab1a4348f4a4.c3n41cmd0nqnrk39u98g.databases.
286/bludb
   sqlite:///my_data1.db
Done.
```

[45]:

SUM
45596

- WE CALCULATED THE TOTAL PAYLOAD MASS AS 45596 USING THE QUERY SHOW IN THE ABOVE IMAGE

Average Payload Mass by F9 v1.1

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

```
[48]: %sql select avg(payload_mass__kg_) as Average_payloadmass from SPACEXTBL\
where Booster_Version= 'F9 v1.1';

 * ibm_db_sa://wlh40668:***@1bbf73c5-d84a-4bb0-85b9-ab1a4348f4a4.c3n41cmd0nqnrk39u98g.databases.appdomain.cloud:32286/bludb
   sqlite:///my_data1.db
Done.
```

[48]:

average_payloadmass
2928

- QUERY CALCULATES THE AVERAGE PAYLOAD MASS OR LAUNCHES WHICH USED BOOSTER VERSION F9 V1.1
- AVERAGE PAYLOAD MASS OF F9 1.1 IS ON THE LOW END OF OUR PAYLOAD MASS RANGE

First Successful Ground Landing Date

Hint:Use min function

In [49]:
```
%sql SELECT MIN(DATE) AS FIRST_SUCCESSFUL_GROUND_LANDING FROM SPACEXTBL \
    WHERE LANDING_OUTCOME like 'Success (ground pad)';
```

```
 * ibm_db_sa://wlh40668:***@1bbf73c5-d84a-4bb0-85b9-ab1a4348f4a4.c3n41cmd0nqnrk39u98g.databases.appdomain.cloud:322
86/bludb
   sqlite:///my_data1.db
Done.
```

Out[49]:

first_successful_ground_landing
2015-12-22

- THE ABOVE QUERY WAS RUN TO FIND THE 1ST DATE FOR A SUCCESSFUL GROUND LANDING. WE CAN SEE THAT THIS OCCURRED ON 12/22/20215.

Successful Drone Ship Landing with Payload between 4000 and 6000

In [50]:

```
%sql SELECT BOOSTER_VERSION FROM SPACEXTBL \
    WHERE (LANDING_OUTCOME = 'Success (drone ship)') AND (PAYLOAD_MASS__KG_ BETWEEN 4000 AND 6000);
```

```
 * ibm_db_sa://wlh40668:***@1bbf73c5-d84a-4bb0-85b9-ab1a4348f4a4.c3n41cmd0nqnrk39u98g.databases.appdomain.cloud:32286/bludb
   sqlite:///my_data1.db
Done.
```

Out[50]:

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

- WE USED THE **WHERE** CLAUSE TO FILTER FOR BOOSTERS WHICH HAVE SUCCESSFULLY LANDED ON DRONE SHIP AND APPLIED THE **AND** CONDITION TO DETERMINE SUCCESSFUL LANDING WITH PAYLOAD MASS GREATER THAN 4000 BUT LESS THAN 6000

Total Number of Successful and Failure Mission Outcomes

In [51]:
```
%sql SELECT MISSION_OUTCOME, COUNT(MISSION_OUTCOME) AS TOTAL_NUMBER FROM SPACEXTBL GROUP BY MISSION_OUTCOME;
```

```
 * ibm_db_sa://wlh40668:***@1bbf73c5-d84a-4bb0-85b9-ab1a4348f4a4.c3n41cmd0nqnrk39u98g.databases.appdomain.cloud:3
86/bludb
   sqlite:///my_data1.db
Done.
```

Out[51]:

mission_outcome	total_number
Failure (in flight)	1
Success	99
Success (payload status unclear)	1
None	0

- AS WE CAN SEE, THE SPACE X HAS BEEN QUITE SUCCESSFUL WITH 100 OVERALL SUCCESSFUL MISSIONS. ONLY 1 MISSION HAS BEEN A FAILURE WHILE IN FLIGHT

Boosters Carried Maximum Payload

```
In [52]:  %sql SELECT DISTINCT(BOOSTER_VERSION) FROM SPACEXTBL \
            WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTBL);

 * ibm_db_sa://wlh40668:***@1bbf73c5-d84a-4bb0-85b9-ab1a4348f4a4.c3n41cmd0nqnrk39u98g.databases.appd
86/bludb
   sqlite:///my_data1.db
Done.
```

Out[52]:

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

- WE DETERMINED THE BOOSTER THAT HAVE CARRIED THE MAXIMUM PAYLOAD USING A SUBQUERY IN THE **WHERE** CLAUSE AND THE **MAX()** FUNCTION.

2015 Launch Records

```
In [53]: %sql select Monthname(DATE) as Month,landing_outcome,booster_version,launch_site from SPACEXTBL\
         where DATE like '2015%' and landing_outcome like 'Failure (drone ship)';

          * ibm_db_sa://wlh40668:***@1bbf73c5-d84a-4bb0-85b9-ab1a4348f4a4.c3n41cmd0nqnrk39u98g.databases.app
         86/bludb
            sqlite:///my_data1.db
         Done.
```

Out[53]:

MONTH	landing_outcome	booster_version	launch_site
October	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
April	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40

- WE USED WHERE, LIKE, AND, AND BETWEEN CONDITIONS TO FILTER FAILED LANDING OUTCOMES TO FIND FAILED OUTCOMES IN DRONE SHIP, BOOSTERS, AND LAUNCH SITES IN 2015. AS WE CAN SEE, THESE FAILED OUTCOMES CAME FROM CCAFS LC-40 IN OCTOBER AND APRIL.

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

```
In [54]:  %sql SELECT LANDING_OUTCOME, COUNT(LANDING_OUTCOME) AS TOTAL_NUMBER FROM SPACEXTBL \
              WHERE DATE BETWEEN '2010-06-04' AND '2017-03-20' \
              GROUP BY LANDING_OUTCOME \
              ORDER BY TOTAL_NUMBER DESC;

 * ibm_db_sa://wlh40668:***@1bbf73c5-d84a-4bb0-85b9-ab1a4348f4a4.c3n41cmd0nqnrk39u98g.dat
86/bludb
   sqlite:///my_data1.db
Done.
```

Out[54]:

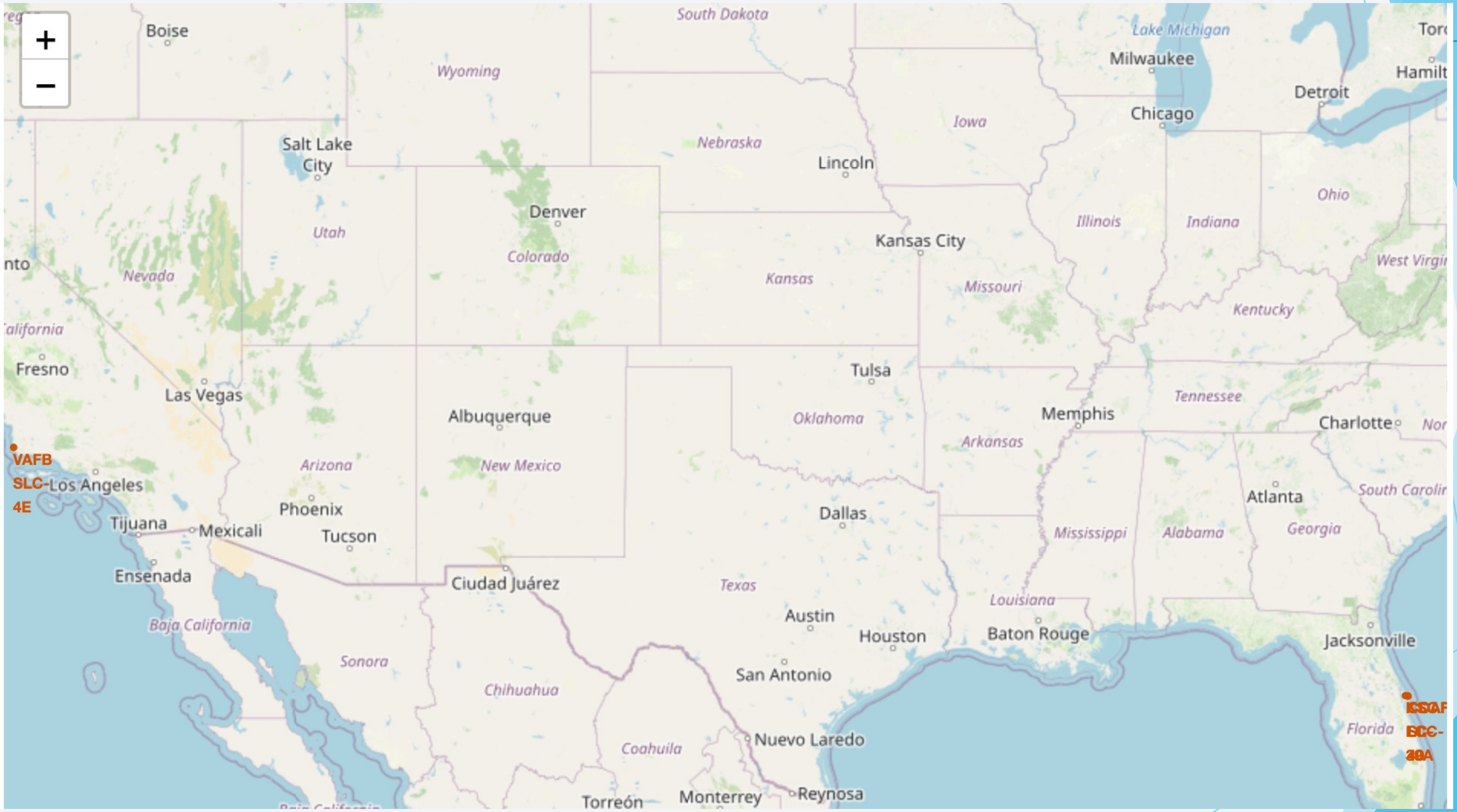
landing_outcome	total_number
No attempt	10
Failure (drone ship)	5
Success (drone ship)	5
Success (ground pad)	5
Controlled (ocean)	3
Uncontrolled (ocean)	2
Failure (parachute)	1
Precluded (drone ship)	1

- LANDING OUTCOMES WERE SELECTED USING THE **COUNT** FUNCTION FROM THE DATA AND THE WHERE CLAUSE WAS APPLIED TO CHOOSE LANDING OUTCOMES **BETWEEN** 2010-06-04 TO 2010-03-20.
- **GROUP BY** CLAUSE WAS THEN USED TO GROUP THE LANDING OUTCOMES. **ORDER BY** CLAUSE WAS USED TO ORDER THE GROUPED LANDING OUTCOME IN DESCENDING ORDER.

Section 3

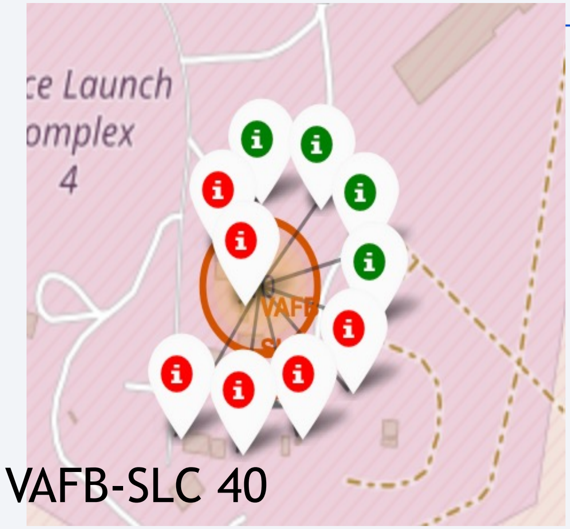
Launch Sites Proximities Analysis

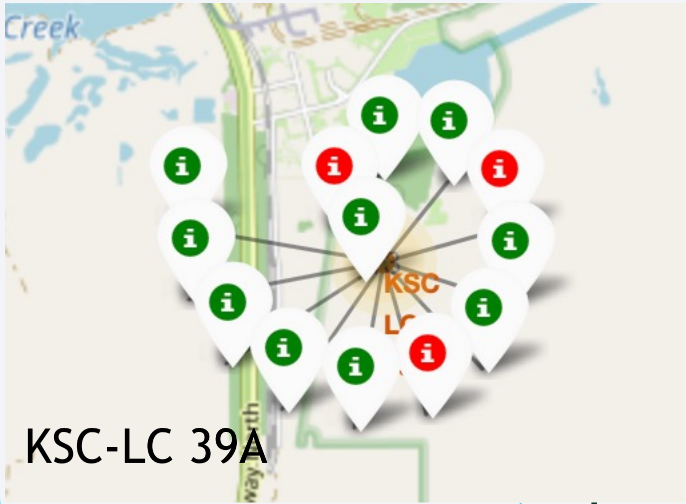
All Space X Launch sites on the map of USA

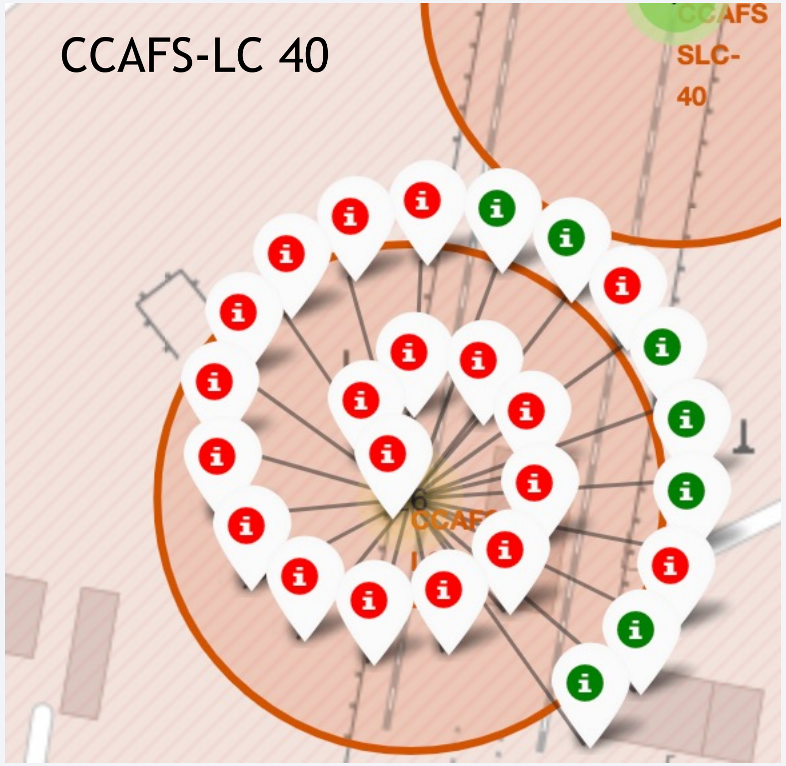


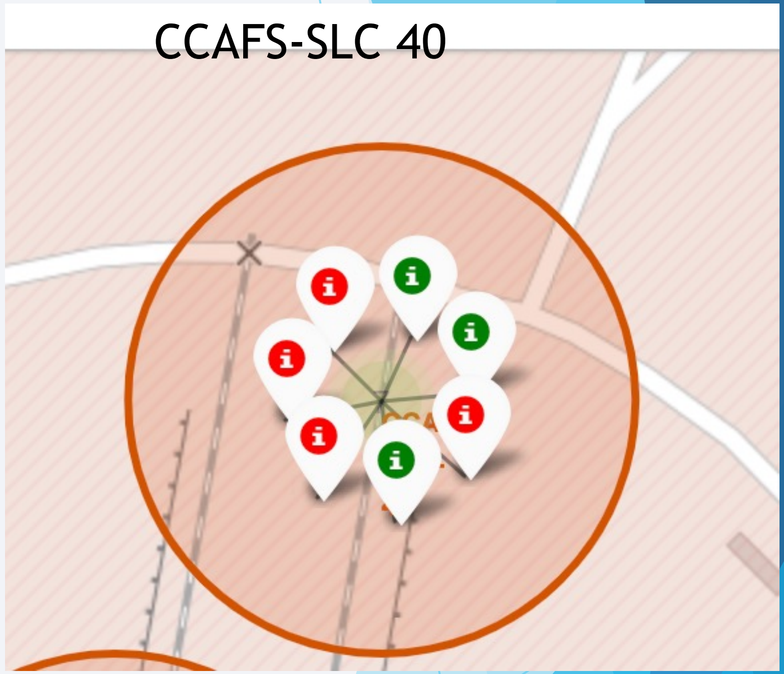
- Space X launch sites are located along the coasts of the United States of America. As we can see, they have predominately launched from Florida and California

Successful/Failed Launches per site



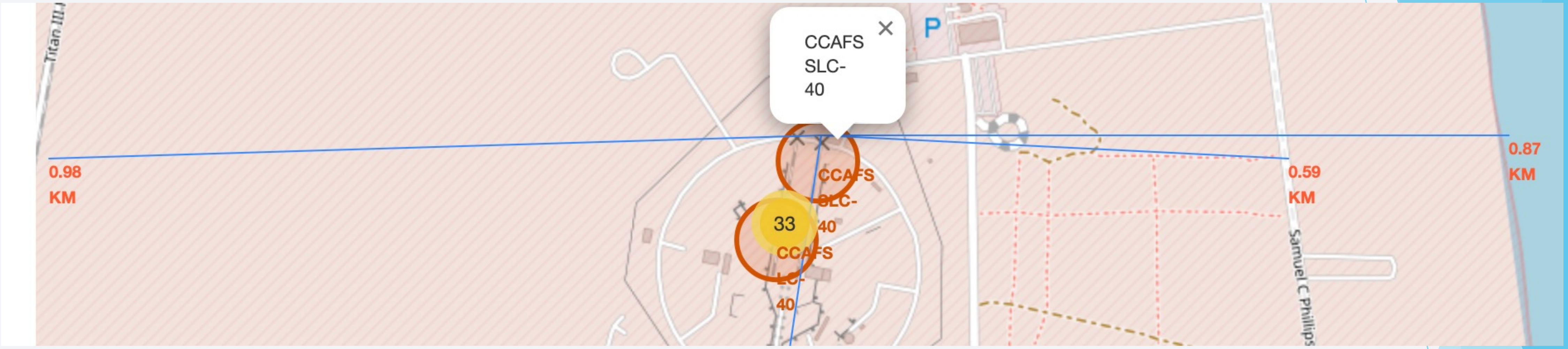






- Launches are grouped in clusters. Green markers shows successful launches and red markers show unsuccessful launches.

Launch Site Proximities to other Points of Interest



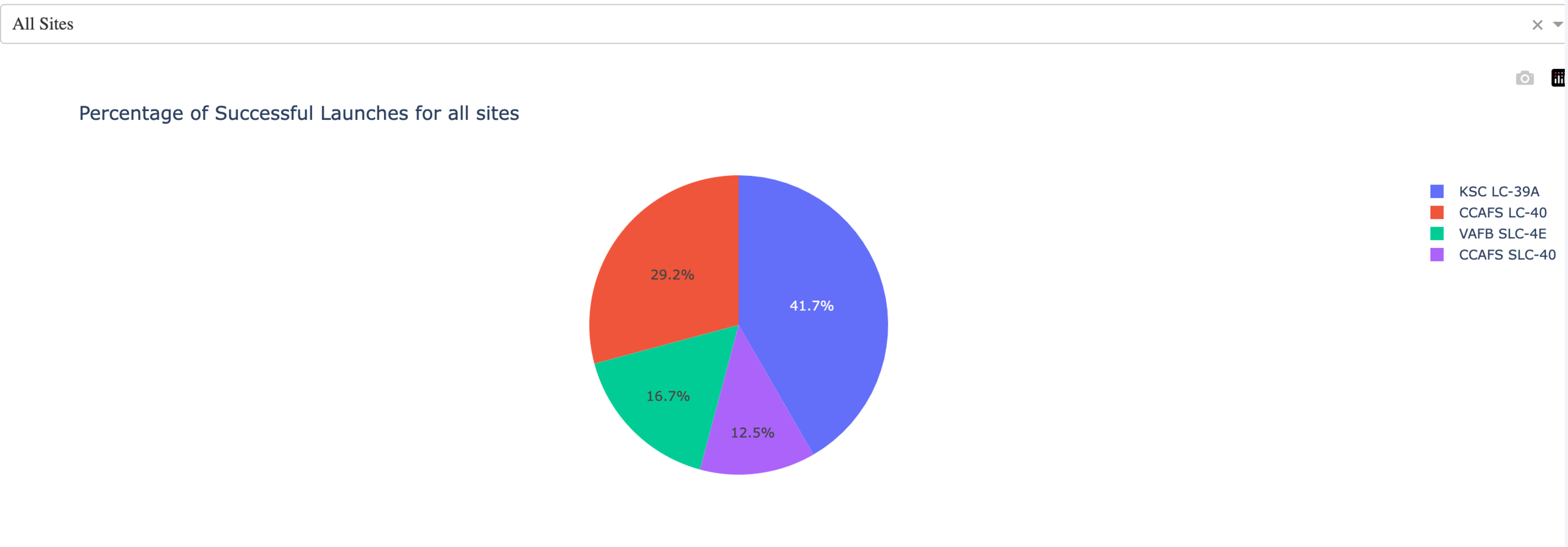
Using CCAFS SLC-40 as our chosen site, we can answer the following question:

- Are launch sites in close proximity to railways? Yes, the nearest railway is only 0.96 km away.
- Are launch sites in close proximity to highways? Yes, the nearest highway is 0.59km away.
- Are launch sites in close proximity to coastline? Yes, our chosen launch site is only 0.87km from the coastline.
- Do launch sites keep certain distance away from cities? Yes, the nearest city, Cape Canaveral, is 18.21km from the launch site.

Section 4

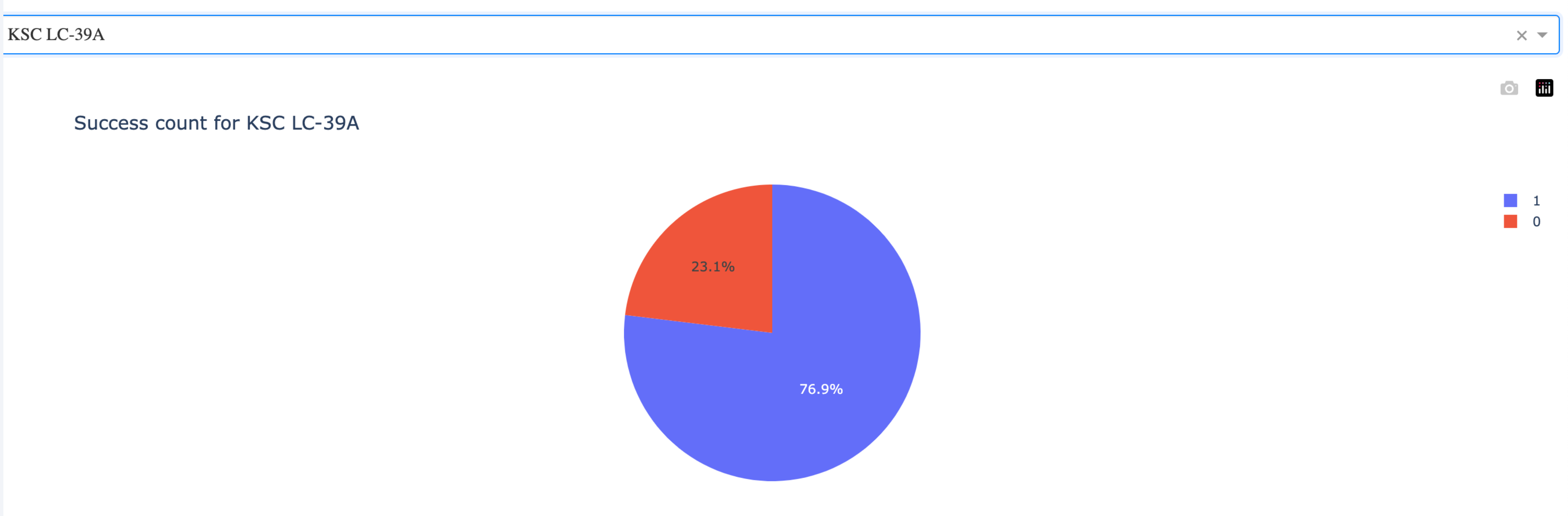
Build a Dashboard with Plotly Dash

Pie Chart showing Success Rate of all Launch Site



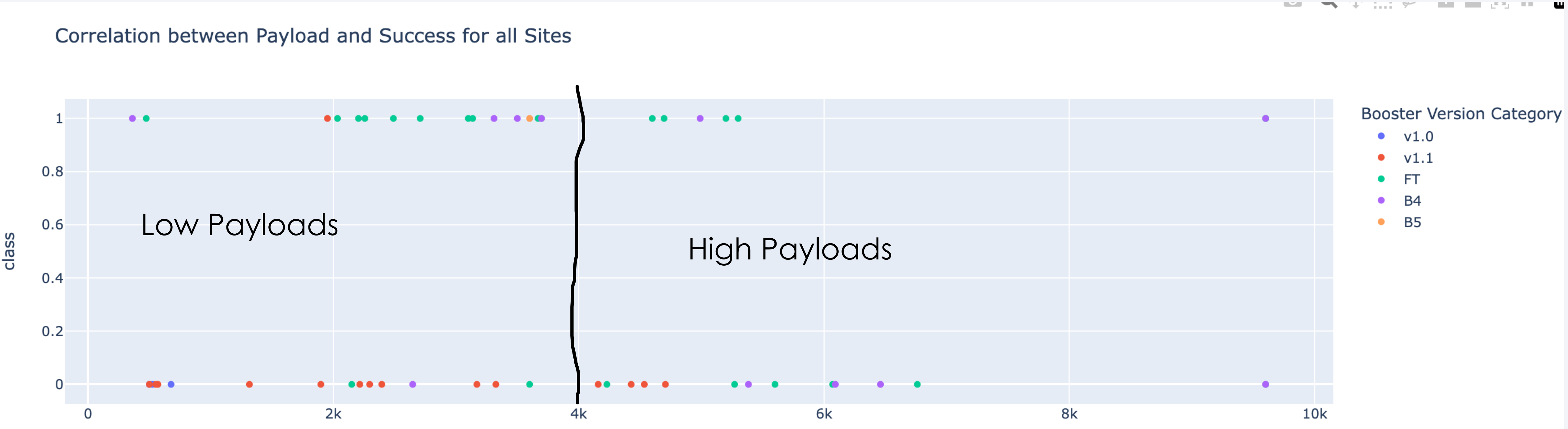
- Based on our pie chart, we can see that KSC LC-39A has had the most successful launches at 41.7%

Pie chart of Launch Site with the highest launch success ratio



- The same launch site, KSC LC-39A also has the high launch success ratio (success vs failure) with 76.9% of launches being successful.

Launch Outcome vs Payload Scatter Plot for all Launch Sites.



- Payload of 4000 shows that there is a gap in outcomes, so this can be used to split our data in two parts: Low payloads (0-4000) and High payloads (4000-10,000)
- We can see that the success rate is much high at low payloads when compared to high payloads. We can also see that some booster versions are also not launched at low payloads as well.

Section 5

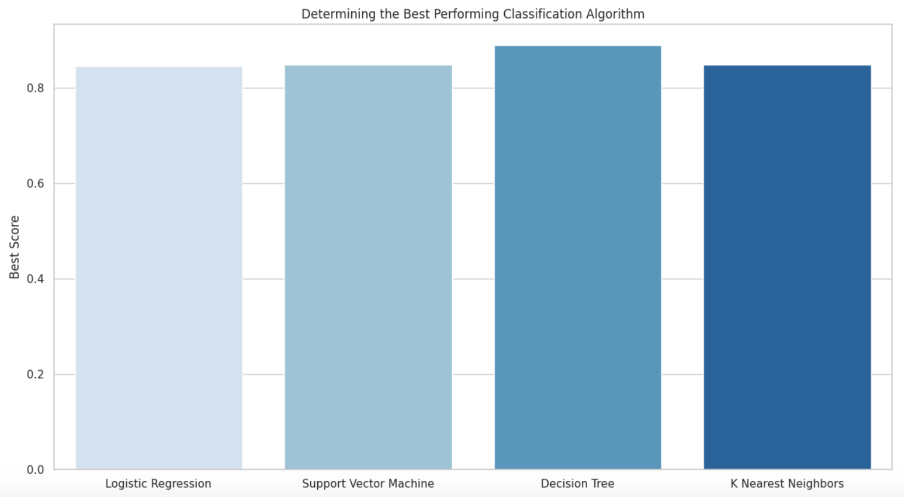
Predictive Analysis (Classification)

Classification Accuracy

- Based on the table above with the best score and the best score for accuracy, we can see that the decision tree model is the best for classification with the highest accuracy. The accuracy level for Decision Tree is 94.44%

```
sns.set(style="whitegrid")

plt.figure(figsize=(15,8))
sns.barplot(x=various_algorithms, y=all_best_scores, palette="Blues")
plt.title("Determining the Best Performing Classification Algorithm")
plt.ylabel("Best Score")
plt.show()
```



Find the method performs best:

In [87]:
```
various_algorithms=['Logistic Regression','Support Vector Machine','Decision Tree','K Nearest Neighbors']
various_scores=[lr_score,svm_score,tree_score,knn_score]
all_best_scores=[lr_best_score,svm_best_score,tree_best_score,knn_best_score]
column_names=['Algorithm Name','Accuracy','Best Score']


df = pd.DataFrame(list(zip(various_algorithms, various_scores, all_best_scores)),columns = column_names)
df
```

Out[87]:

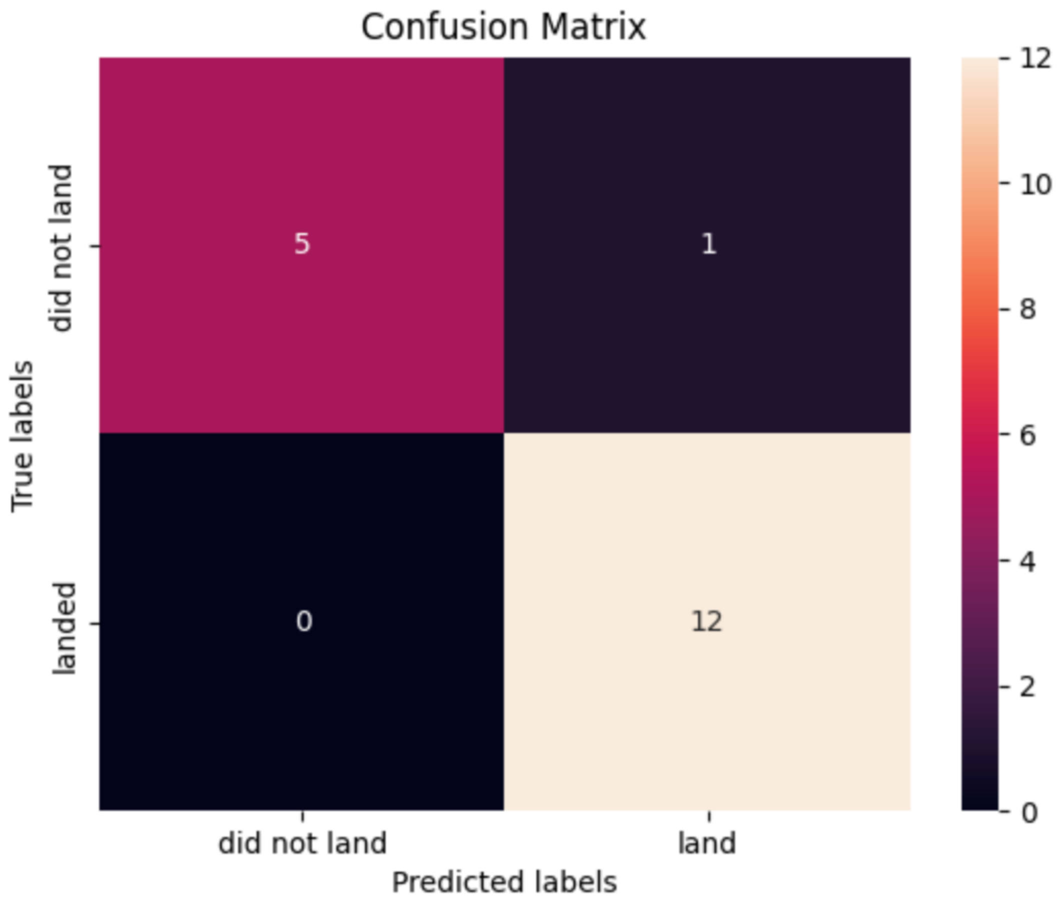
	Algorithm Name	Accuracy	Best Score
0	Logistic Regression	0.833333	0.846429
1	Support Vector Machine	0.833333	0.848214
2	Decision Tree	0.833333	0.889286
3	K Nearest Neighbors	0.833333	0.848214

Confusion Matrix

- The confusion matrix further shows why the Decision Tree is the best model to make predictions.
- Only 1 of the total 18 results has been incorrectly classified as a false positive (shown as a landed even when it should have not landed.
- The remaining 17 results are correctly classified.

19]:

```
yhat = tree_cv.predict(X_test)
plot_confusion_matrix(Y_test,yhat)
```



Conclusions

- To summarize our conclusions, we can see:

1. The greater the number of flights (flight number), the greater the success rate at each launch site.
2. Launch Site success rates have steadily risen from the years 2013 to 2020 with only 2 dips in 2018 and 2020.
3. The orbits with the highest success rate are those of ES-L1, GEO, HEO, SSO. All of these at 100%.
4. The launch site KSC Lc-39A has the most success with its launches (41.7%). Of these launches, 76.9% have been successful.
5. The Decision Tree Machine Learning model is the best to use when making predictions with this data as its accuracy score is 94.44%

Thank you!