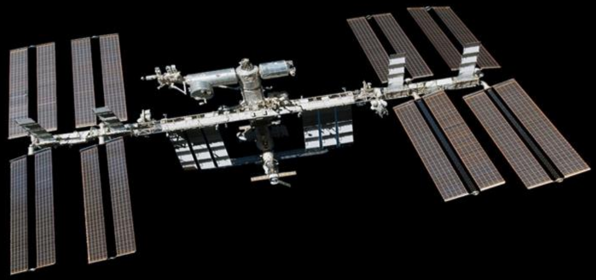
IBM Developer
SKILLS NETWORK



Space Y

First Stage Reuse

Brian Pickett
10/10/2024

Outline

EXECUTIVE SUMMARY

INTRODUCTION

METHODOLOGY

RESULTS

CONCLUSION

Executive Summary

Summary of methodologies:

Determining the likelihood of the first stage successfully landing is crucial for estimating the launch cost

1. **Data Collection**: Gather data from various sources, including transaction records, customer feedback, and demographic information.
2. **Hypothesis Formulation**: Develop hypotheses such as “Customers with low engagement are more likely to churn” or “Frequent discounts increase customer retention.”
3. **Exploratory Data Analysis (EDA)**: Use EDA techniques to uncover patterns and correlations in the data. For instance, visualize the relationship between purchase frequency and churn rate.
4. **Modeling**: Build predictive models to identify at-risk customers and test the hypotheses.
5. **Evaluation**: Assess the model’s performance and refine it based on feedback and additional data.

Summary of Findings

Exploratory Data Analysis (EDA): KSC LC-39A has the highest success rate among landing sites • Orbits ES -L1, GEO, HEO, and SSO have a 100% success rate.

Visual Analytics: You can observe that the success rate since 2013 kept increasing till 2017 (stable in 2014) and after 2015 it started increasing. Now if you observe Payload Vs. Launch Site scatter point chart you will find for the VAFB-SLC launch site there are no rockets launched for heavy payload mass(greater than 10000).

Predictive Analytics: All models performed similarly on the test set. The decision tree model slightly outperformed.

Introduction

Project background and context

We need to be able to predict if the Falcon 9 first stage will land successfully. SpaceX advertises Falcon 9 rocket launches on its website with a cost of 62 million dollars; other providers cost upward of 165 million dollars each, much of the savings is because SpaceX can reuse the first stage. Therefore, if we can 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 bid against SpaceX for a rocket launch.

Explore

How payload mass, launch site, number of flights, and orbits affect first-stage landing success.

Rate of successful landings over time.

Best predictive model for successful landing (binary classification).

Methodology

<u>Data Collection and Preparation</u>:

Once the problem is defined, the next step is to gather and prepare the relevant data.

This involves identifying data sources, ensuring data quality, and transforming raw data into a format suitable for analysis.

<u>Data Exploration and Analysis</u>:

In this phase, we explored the data to uncover patterns, trends, and insights.

This step often involves statistical analysis and visualization techniques to understand the data better and identify any anomalies or interesting features.

<u>Model Building and Evaluation</u>:

Here, we built predictive models using machine learning algorithms.

These models are then evaluated to ensure they meet the desired performance criteria.

This step may involve multiple iterations to refine the models.

<u>Deployment and Maintenance</u>:

The final step is to deploy the model into a production environment where it can be used to make real-time decisions.

Ongoing maintenance is crucial to ensure the model continues to perform well as new data becomes available.

Data Collection – SpaceX API

REQUEST: The requested JSON results we will use the following static response object for this project (seen below).

NORMALIZE: Now we decode the response content as a Json using .json() and turn it into a Pandas dataframe using .json_normalize().

REQUEST: We will now use the API again to get information about the launches using the IDs given for each launch. Specifically, we will be using columns rocket, payloads, launchpad, and cores.

DICTIONARY: Finally let's construct our dataset using the data we have obtained. We we combine the columns into a dictionary.

DATAFRAME: Pandas' data frame from the dictionary launch_dict.

FILTER: Filter the dataframe to only include Falcon 9 launches.

DATA WRANGLING: We can see below that some of the rows are missing values in our dataset. Calculate below the mean for the PayloadMass using the .mean(). Then use the mean and the .replace() function to replace np.nan values in the data with the mean you calculated

API CALLS

spacex_url=https://api.spacexdata.com/v4/launches/past

static_json_url='https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DS0321EN-SkillsNetwork/datasets/API_call_spacex_api.json'

https://github.com/brianpickett72/Capstone-IBM-DS/blob/579c54710f8681377c79f12bb6bf0c5a71eb46df/jupyter-labs-spacex-data-collection-api-v2%20(2).ipynb

Data Collection - Scraping

Request data (Falcon 9 launch data) from Wikipedia
Create BeautifulSoup object from HTML response
Extract column names from HTML table header
Collect data from parsing HTML tables
Create dictionary from the data
Create DF from the dictionary
Export data to csv file

https://github.com/brianpickett72/Capstone-IBM-DS/blob/631d9817426b42a5fdbb986c2a9fd89a07ca5dce/jupyter-labs-webscraping.ipynb

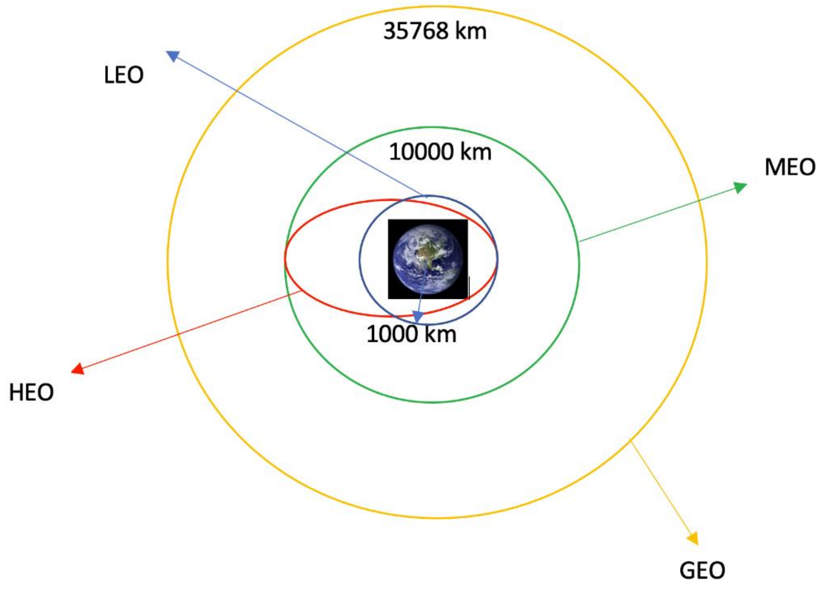
Exploratory Data Analysis (EDA)
Determine Training Labels:
Data Analysis: (Calculations)

- Landing Outcomes
- Launches on each site
- Number of occ/orbit

Create Classification Variable(Outcome)

True Ocean means the mission outcome was successfully landed to a specific region of the ocean.
False Ocean means the mission outcome was unsuccessfully landed to a specific region of the ocean.
True RTLS means the mission outcome was successfully landed to a ground pad.
False RTLS means the mission outcome was unsuccessfully landed to a ground pad.
True ASDS means the mission outcome was successfully landed on a drone ship.
False ASDS means the mission outcome was unsuccessfully landed on a drone ship.

Convert those outcomes into training labels with
1 means the booster successfully landed
0 means it was unsuccessful.



Determine the success rate: np.float64(0.6666666666666666)

https://github.com/brianpickett72/Capstone-IBM-DS/blob/ccf1d6258dede67c6ce7047e6b24498399e23bb1/labs-jupyter-spacex-Data%20wrangling.ipynb

Data Wrangling

First, let's try to see how the FlightNumber (indicating the continuous launch attempts.) and Payload variables would affect the launch outcome.

We can plot out the FlightNumber vs. PayloadMassand overlay the outcome of the launch. We see that as the flight number increases, the first stage is more likely to land successfully.

The payload mass is also important; it seems the more massive the payload, the less likely the first stage will return

Visualize the relationship between Flight Number and Launch Site
Visualize the relationship between Payload and Launch Site
Visualize the relationship between success rate of each orbit type
Visualize the relationship between FlightNumber and Orbit type
Visualize the relationship between Payload and Orbit type
Visualize the launch success yearly trend

https://github.com/brianpickett72/Capstone-IBM-DS/blob/6d0e3a8db7c4622d2742fb5e3bc1bea44f9b105a/jupyter-labs-eda-dataviz-v2%20(1)%20(1).ipynb

EDA with Data Visualization

EDA with SQL

SQL Queries

Display the names of the unique launch sites in the space mission
Display 5 records where launch sites begin with the string 'CCA'
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 when the first succesful landing outcome in ground pad was acheived.
List the names of the boosters which have success in drone ship and have payload mass greater than 4000 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.
List the records which will display the month names, failure landing_outcomes in drone ship ,booster versions, launch_site for the months in year 2015.
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/brianpickett72/Capstone-IBM-DS/blob/7e2769740cf98fc92b3945e69c44e6ee157aa609/jupyter-labs-eda-sql-coursera_sqllite.ipynb

Build an Interactive Map with Folium

https://github.com/brianpickett72/Capstone-IBM-DS/blob/8a7c539144d906e4fba4f06d62b09568c209381b/spacex_dash_app_bp.py

Summarize what map objects such as markers, circles, lines, etc. you created and added to a folium map

TASK 1: Mark all launch sites on a map

TASK 2: Mark the success/failed launches for each site on the map

TASK 3: Calculate the distances between a launch site to its proximities

Next, let's create markers for all launch records. If a launch was successful , then we use a **green marker** and if a launch was failed, we use a red marker.

From the color-labeled markers in marker clusters, you should be able to easily identify which launch sites have relatively high success rates.

Create a marker with distance to a closest city, railway, highway, etc.

Add the GitHub URL of your completed interactive map with Folium map, as an external reference and peer-review purpose

Build a Dashboard with Plotly Dash

https://github.com/brianpickett72/Capstone-IBM-DS/blob/8a7c539144d906e4fba4f06d62b09568c209381b/spacex_dash_app_bp.py

Add a Launch Site Drop-down Input Component

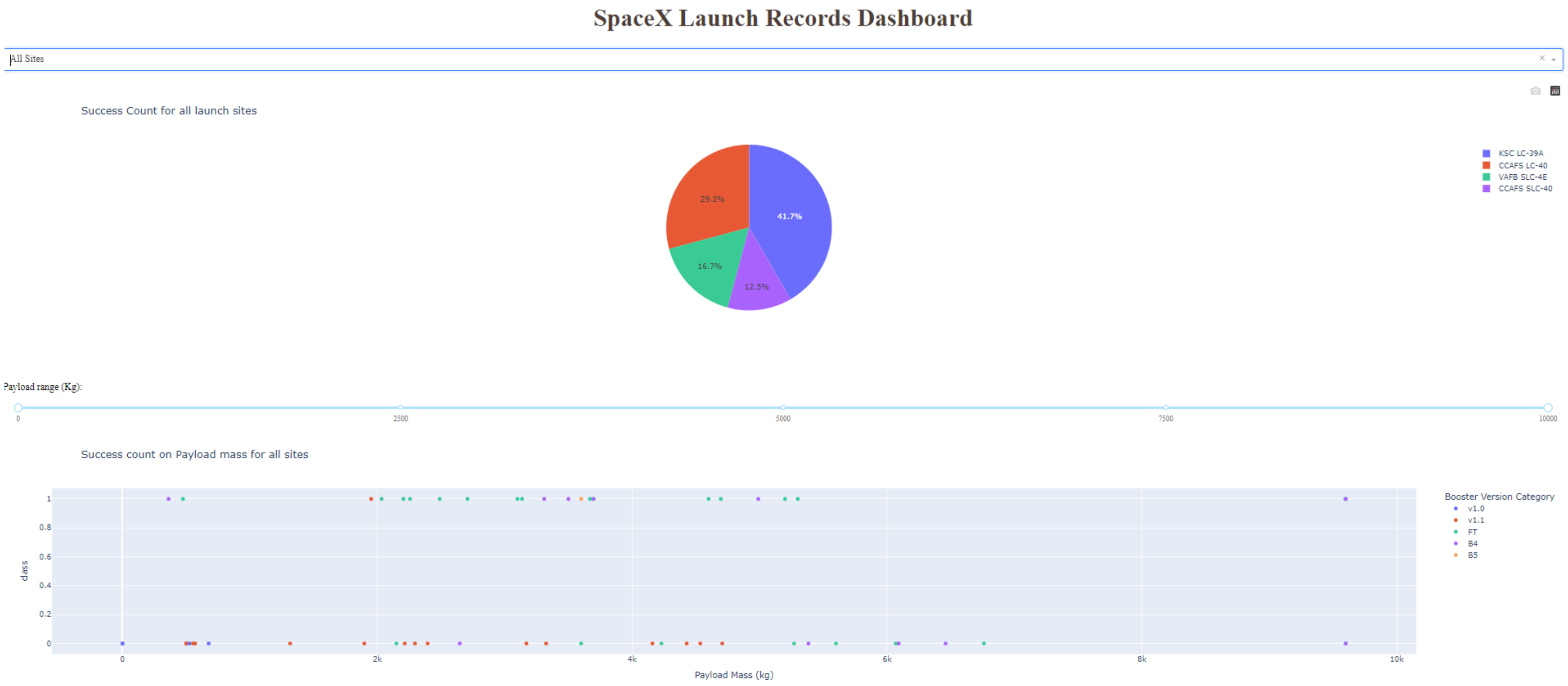
Add a callback functions

Render success-pie-chart based on selected site dropdown

Scatter Plot activation

Slider for Payload Range

Site Selections



Calculate the accuracy on the test data using the method score:

Predictive Analysis (Classification)

<u>Perform exploratory Data Analysis and determine Training Labels</u>

Create a NumPy array for the class
Standardize the data / Transform the data
Split into training data and test data
Find best Hyperparameter for SVM, Classification Trees and Logistic Regression
Find the method performs best using test data

<u>Calculate the accuracy on the test data using the method score</u>

Create a decision tree classifier object then create a GridSearchCV object tree_cv with cv = 10.
Fit the object to find the best parameters from the dictionary parameters.
Calculate the accuracy of tree_cv on the test data using the method score
Calculate the accuracy of knn_cv on the test data using the method score

https://github.com/brianpickett72/Capstone-IBM-DS/blob/901711913705794c66eaf4fd08f0065352d11d66/SpaceX-Machine-Learning-Prediction-Part-5.ipynb

Results


Wide-field Survey Explorer views the Pleiades - NASA / JPL-Caltech / UCLA / Judy Schmidt

Results Summary

EXPLORATORY DATA ANALYSIS RESULTS:

LAUNCH SUCCESS HAS IMPROVED OVER TIME

KSC LC-39A has the highest success rate among landing sites

Orbits ES-L1, GEO, HEO and SSO have a 100% success rate

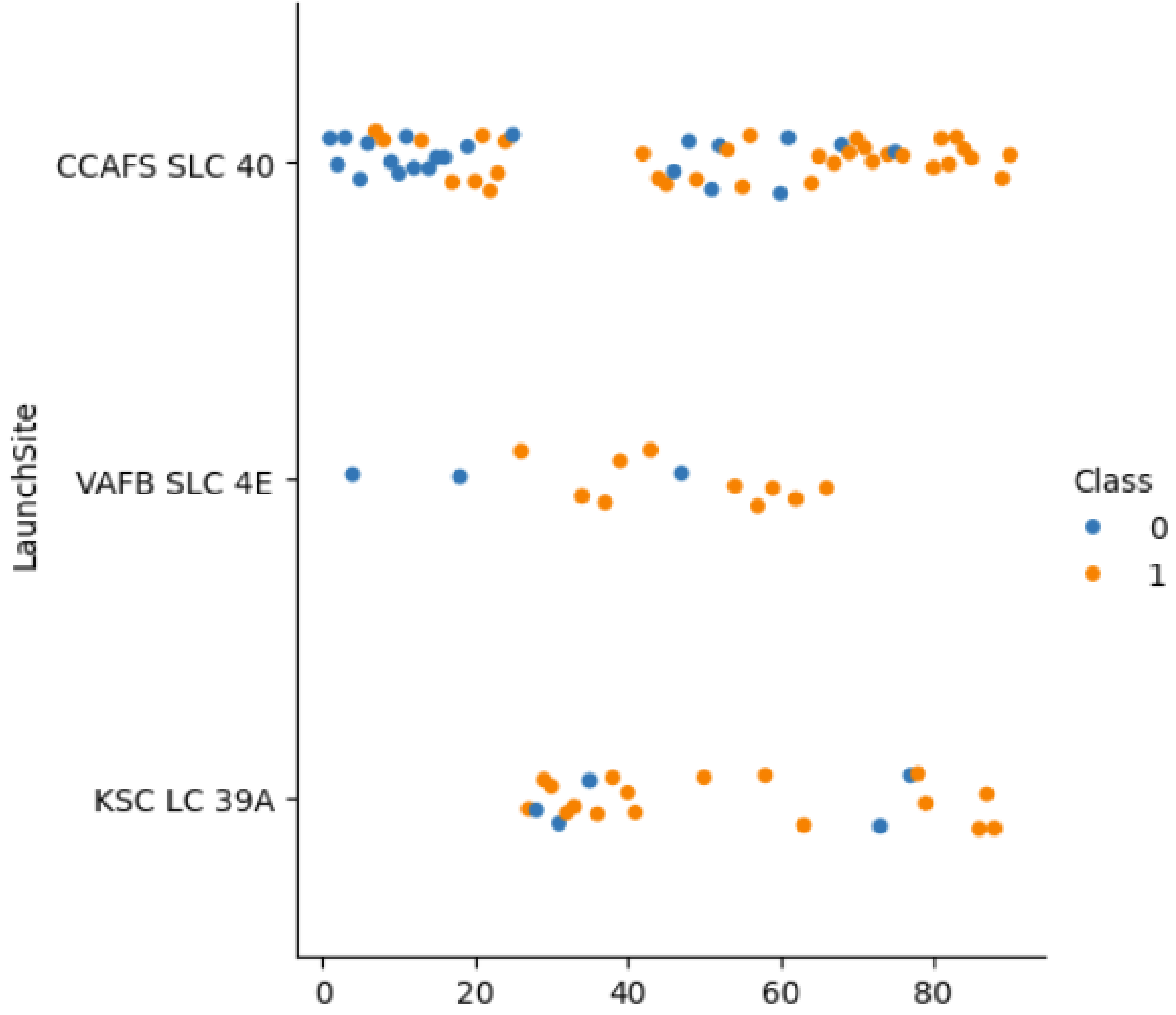
INTERACTIVE ANALYTICS DEMO

MOST LAUNCH SITES ARE NEAR THE EQUATOR, AND ALL ARE CLOSE TO THE COAST

Launch sites are far enough away from anything a failed launch can damage (city, highway, railway), while still close enough to bring people and material to support launch activities

PREDICTIVE ANALYSIS RESULTS

DECISION TREE MODEL IS THE BEST PREDICTIVE MODEL FOR THE DATASET



Scatter plot of Flight Number vs. Launch Site

Exploratory Data Analysis

- Earlier flights had a lower success rate (blue = fail)
- Later flights had a higher success rate (orange = success)

Around half of launches were from **CCAFS SLC 40 launch site**

- VAFB SLC 4E and KSC LC 39A have higher success rates

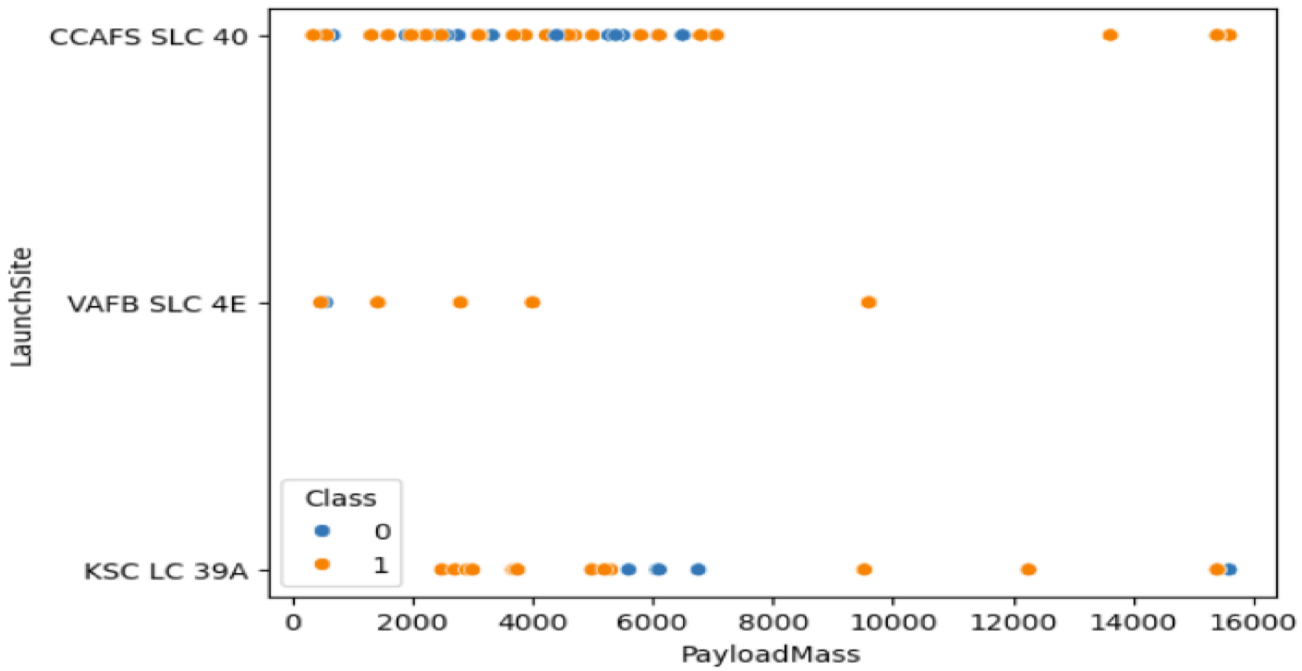
We can infer that new launches have a higher success rate

VAFB-SLC launch site there are no rockets launched for heavy payload mass(greater than 10000)

Most launces with a payload greater than 7,000 kg were successful

KSC LC 39A has a 100% success rate for launches less than 5,500 kg

VAFB SKC 4E has not launched anything greater than ~10,000 kg



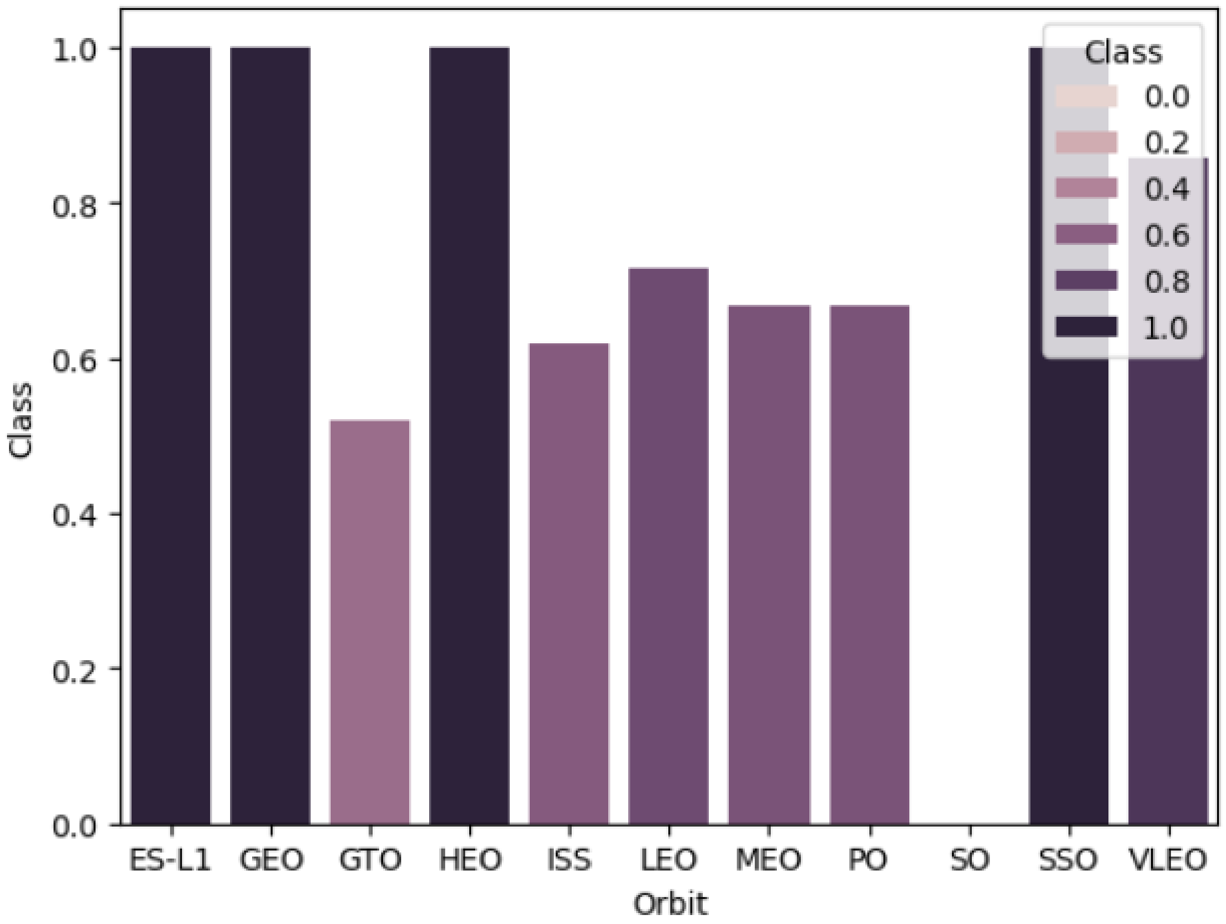
Typically, the higher the payload mass (kg), the higher the success rate

Payload vs. Launch Site

100% Success Rate: ES-L1, GEO, HEO and SSO

50%-80% Success Rate: GTO, ISS, LEO, MEO, PO

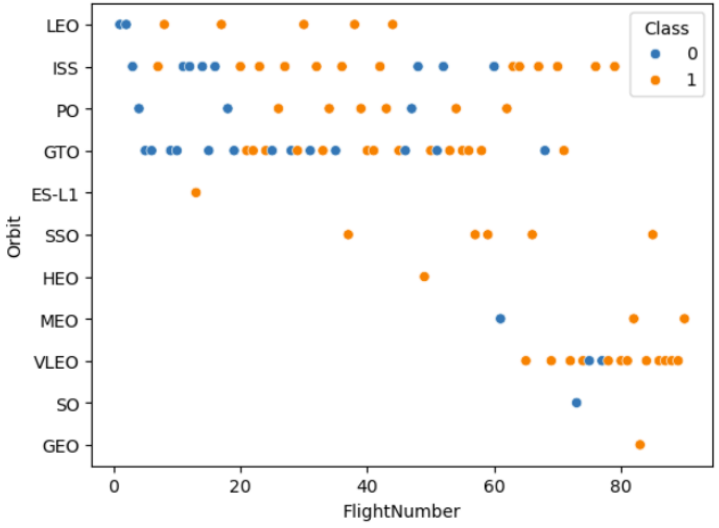
0% Success Rate: SO

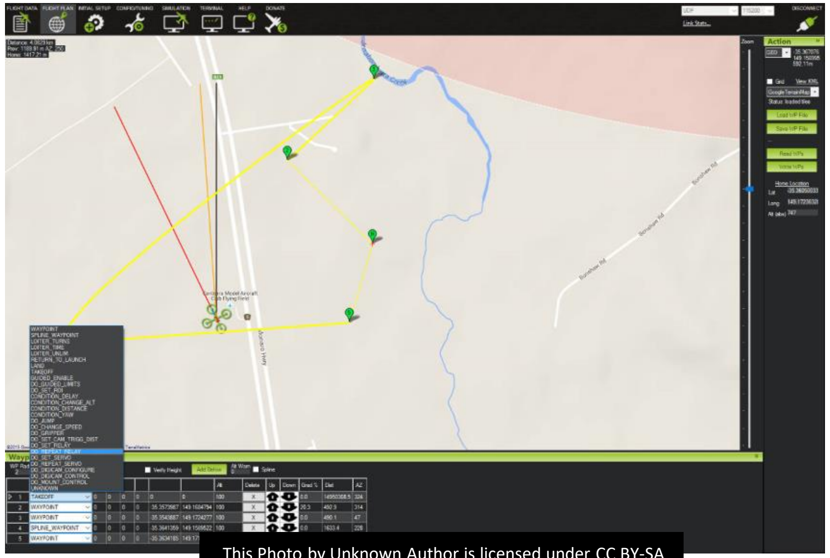


Success Rate vs. Orbit Type

Analyze the ploted bar chart try to find which orbits have high sucess rate.

Flight Number vs. Orbit Type



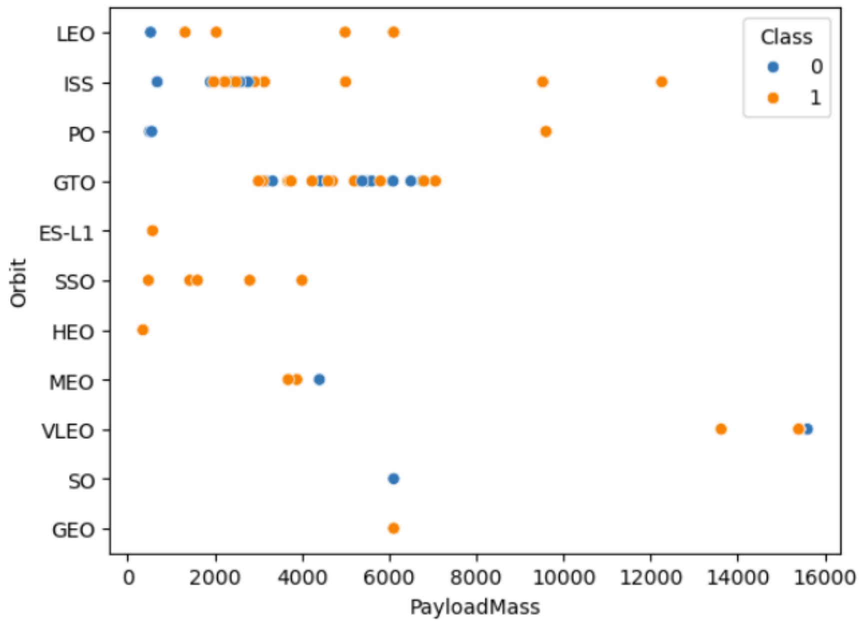


You should see that in the LEO orbit the Success appears related to the number of flights; on the other hand, there seems to be no relationship between flight number when in GTO orbit.

Show a scatter point of payload vs. orbit type

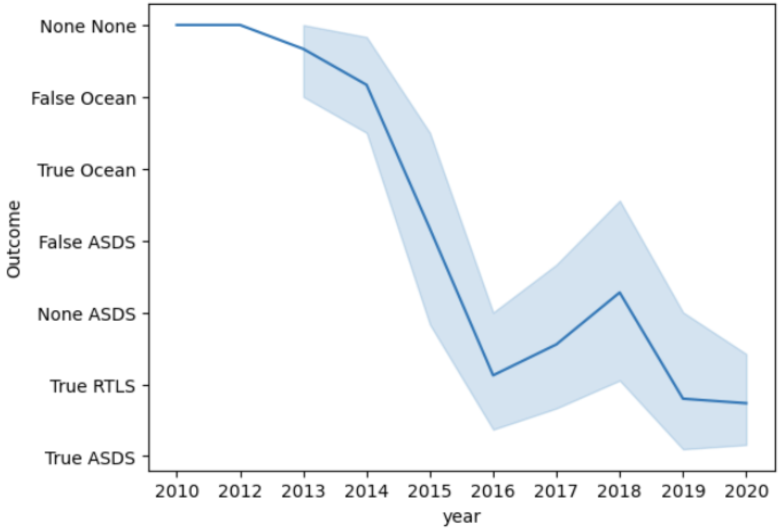
With heavy payloads the successful landing or positive landing rate are more for Polar, LEO and ISS.

However, for GTO we cannot distinguish this well as both positive landing rate and negative landing(unsuccessful mission) are both there here.



Payload vs. Orbit Type

Line chart of yearly average success rate



- The success rate improved from 2013-2017 and 2018-2019
- The success rate decreased from 2017-2018 and from 2019-2020
- Overall, the success rate has improved since 2013

All Launch Site Names

Find the names of the unique launch sites

```
[22]: %sql select distinct "Launch_Site" from SPACEXTABLE
```

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

[22]:

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

Present your query result with a short explanation here

```
[57]: # Select relevant sub-columns: `Launch Site`, `Lat(Latitude)`, `Long(Longitude)`, `class`
spacex_df = spacex_df[['Launch Site', 'Lat', 'Long', 'class']]
launch_sites_df = spacex_df.groupby(['Launch Site'], as_index=False).first()
launch_sites_df = launch_sites_df[['Launch Site', 'Lat', 'Long']]
launch_sites_df
```

[57]:

	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	-120.610745

Launch Site Names Begin with 'CCA'

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

```
[42]: %sql select * from SPACEXTABLE where "Launch_Site" like 'CCA%' limit 10
```

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

[42]:

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	7:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
2012-10-08	0:35:00	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
2013-12-03	22:41:00	F9 v1.1	CCAFS LC-40	SES-8	3170	GTO	SES	Success	No attempt
2014-01-06	22:06:00	F9 v1.1	CCAFS LC-40	Thaicom 6	3325	GTO	Thaicom	Success	No attempt
2014-04-18	19:25:00	F9 v1.1	CCAFS LC-40	SpaceX CRS-3	2296	LEO (ISS)	NASA (CRS)	Success	Controlled (ocean)
2014-07-14	15:15:00	F9 v1.1	CCAFS LC-40	OG2 Mission 1 6 Orbcomm-OG2 satellites	1316	LEO	Orbcomm	Success	Controlled (ocean)
2014-08-05	8:00:00	F9 v1.1	CCAFS LC-40	AsiaSat 8	4535	GTO	AsiaSat	Success	No attempt

Total Payload Mass

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

```
%sql select sum(PAYLOAD_MASS__KG_) from SPACEXTABLE where "Customer"
```

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

sum(PAYLOAD_MASS__KG_)
45596

Calculate the total payload carried by boosters from NASA

Present your query result with a short explanation here

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

```
[48]: %sql select avg(PAYLOAD_MASS__KG_) from SPACEXTABLE where "Booster_Version" = 'F9 v1.1'

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

[48]:

avg(PAYLOAD_MASS__KG_)
2928.4

CALCULATE THE AVERAGE PAYLOAD MASS CARRIED BY BOOSTER VERSION F9 V1.1

PRESENT YOUR QUERY RESULT WITH A SHORT EXPLANATION HERE

Average Payload Mass by F9 v1.1

First Successful Ground Landing Date

Task 5

List the date when the first succesful landing outcome in ground pad was acheived.

Hint:Use min function

```
%sql select min(Date) from SPACEXTABLE where Landing_Outcome like 'Success %ground%'
```

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

min(Date)
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 distinct Booster_Version from SPACEXTABLE where Landing_Outcome like 'Success %drone%' and PAYLOAD_MASS__KG_ between 4000 and 6000
```

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

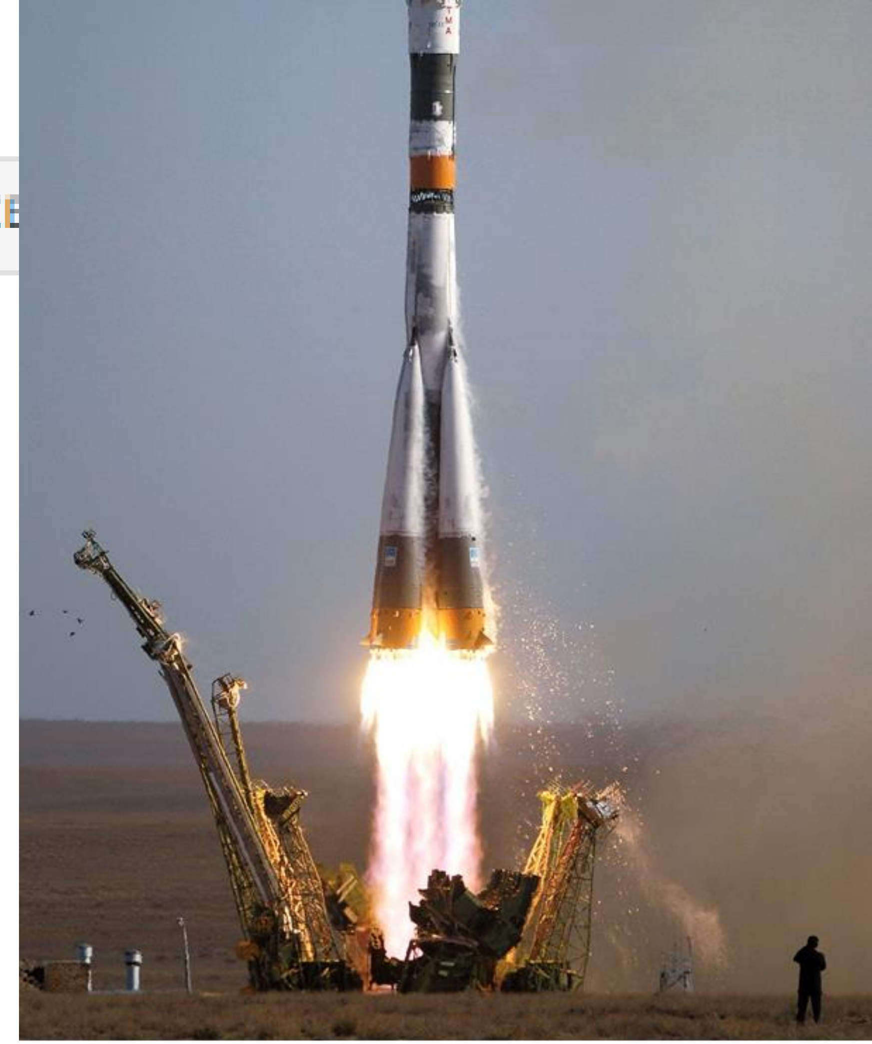
Booster_Version
F9 FT B1022
F9 FT B1026
F9 FT B1021.2
F9 FT B1031.2

List the total number of successful and failure mission outcomes

```
%sql SELECT MISSION_OUTCOME, COUNT(*) as total_number FROM SPACE
```

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

Mission_Outcome	total_number
Failure (in flight)	1
Success	98
Success	1
Success (payload status unclear)	1



Total Number of Successful and Failure Mission Outcomes

Boosters Carried Maximum Payload

```
%sql SELECT BOOSTER_VERSION FROM SPACEXTABLE WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_) FROM SPACEXTABLE)
```

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

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

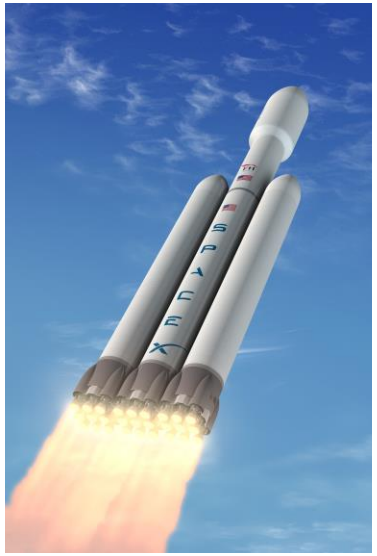
2015 Launch Records

```
%sql SELECT substr(Date,1,4) as month, DATE, BOOSTER_VERSION, LAUNCH_SITE, [Landing_Outcome] FROM SPACEXTABLE where [Landing_Outcome] = 'Failure (drone ship)' and substr(Date,1,4)='2015'
```

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

month	Date	Booster_Version	Launch_Site	Landing_Outcome
2015	2015-01-10	F9 v1.1 B1012	CCAFS LC-40	Failure (drone ship)
2015	2015-04-14	F9 v1.1 B1015	CCAFS LC-40	Failure (drone ship)

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



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

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 [Landing_Outcome], count(*) as count_outcomes FROM SPACEXTABLE WHERE DATE between '2010-06-04' and '2017-03-20' group by [Landing_Outcome] order by count_outcomes DESC
```

* sqlite:///my_data1.db
Done.

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

```
%sql distinct (
```

* sqlite:///my_data1.db
Done.

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

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

Launch Site Analysis using Folium

As mentioned before, launch sites are in close proximity to equator to minimize fuel consumption by using Earth's ~ 30km/sec eastward spin to help spaceships get into orbit. Launch sites are in close proximity to coastline so they can fly over the ocean during launch, for at least two safety reasons-- (1) crew has option to abort launch and attempt water landing (2) minimize people and property at risk from falling debris. Launch sites are in close proximity to highways, which allows for easily transport required people and property. Launch sites are in close proximity to railways, which allows transport for heavy cargo. Launch sites are not in close proximity to cities, which minimizes danger to population dense areas.

Site Location information

Each Launch Site

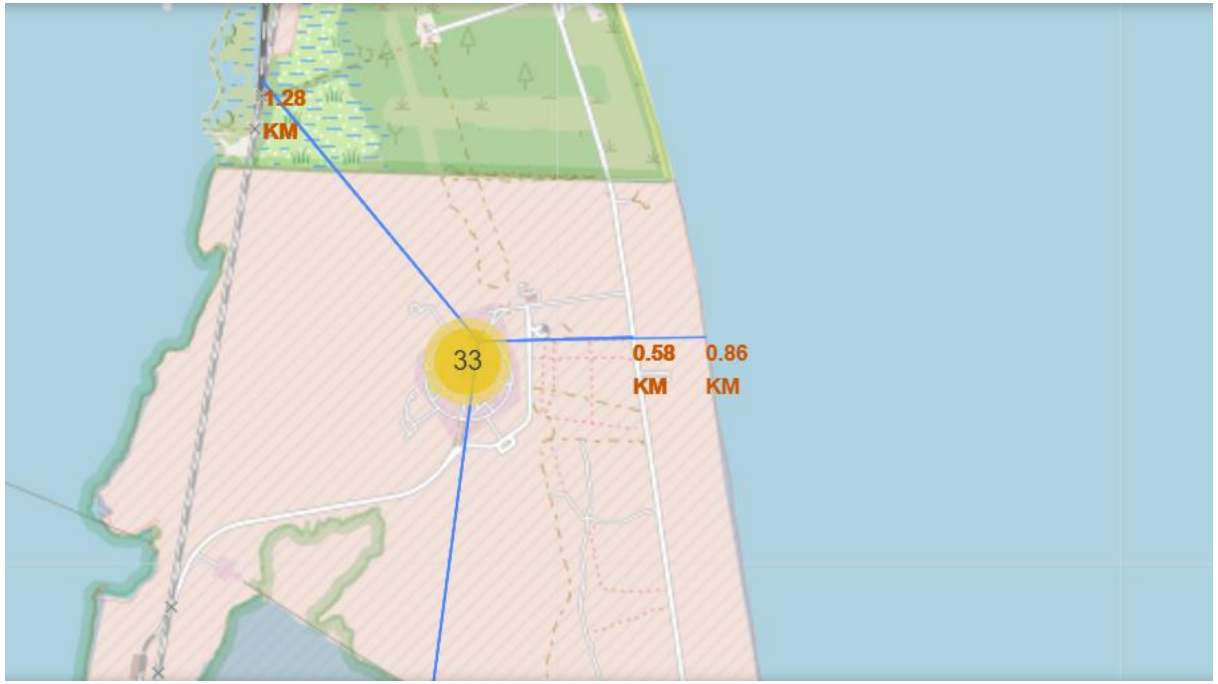
Outcomes:

Green markers for successful launches

Red markers for unsuccessful launches

Launch site CCAFS SLC-40 has a 3/7 success rate (42.9%)

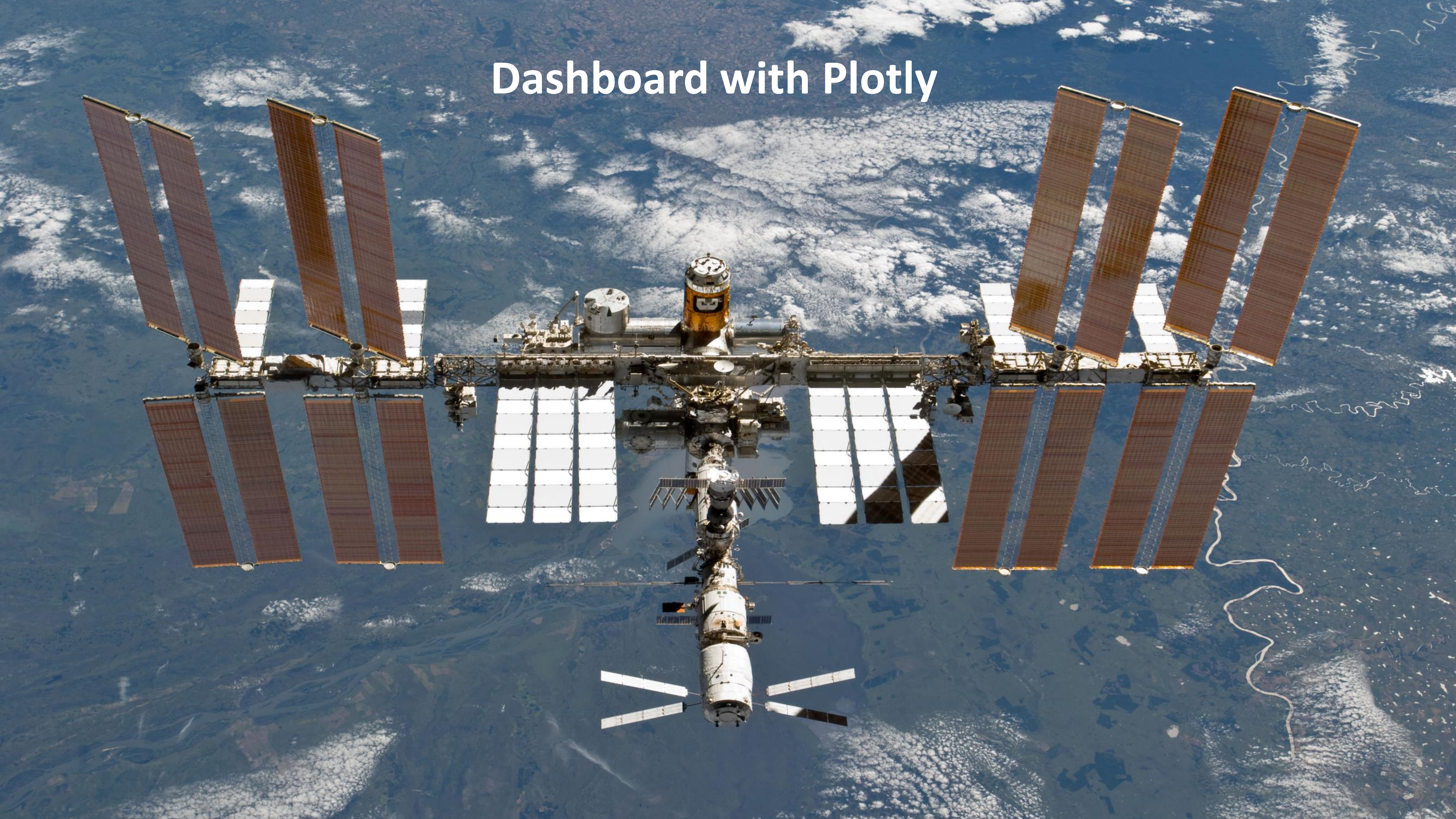
Launch Site Infomation

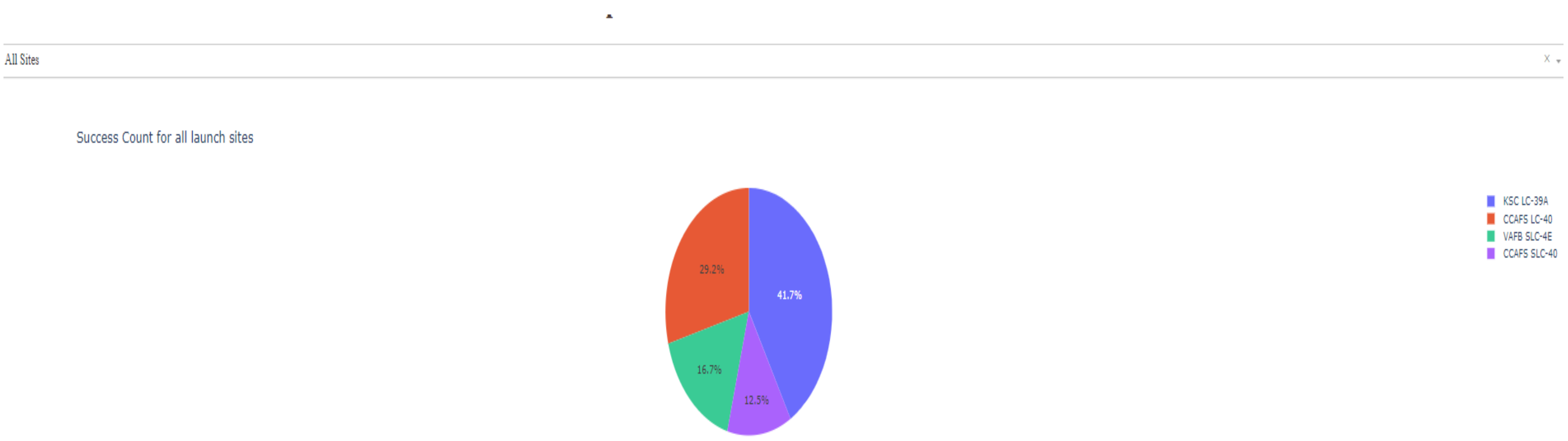


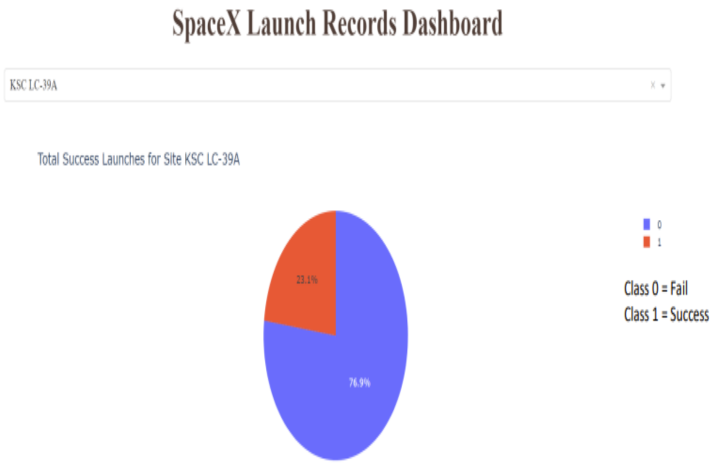
distance_highway = 0.5834695366934144 km
distance_railroad = 1.2845344718142522 km
distance_city = 51.434169995172326 km

Site Location Information

Dashboard with Plotly







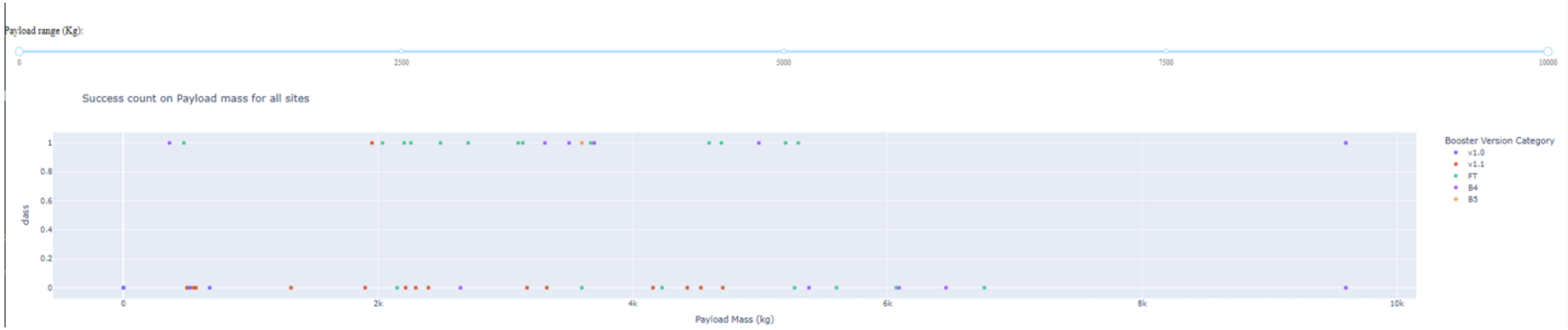
Success as Percent of Total

KSC LC-39A has the highest success rate amongst launch sites (76.9%)

10 successful launches and 3 failed launches

Launch Success by Site

Show screenshots of Payload vs. Launch Outcome scatter plot for all sites, with different payload selected in the range slider



KSC LC-39A has the most successful launches amongst launch sites (41.2%)

Payload Mass and Success By Booster Version

Payloads between 2,000 kg and 5,000 kg have the highest success rate

1 indicating successful outcome and 0 indicating an unsuccessful outcome

Visualize the relationship between Payload and Launch Site

Predictive Analytics

Classification Accuracy

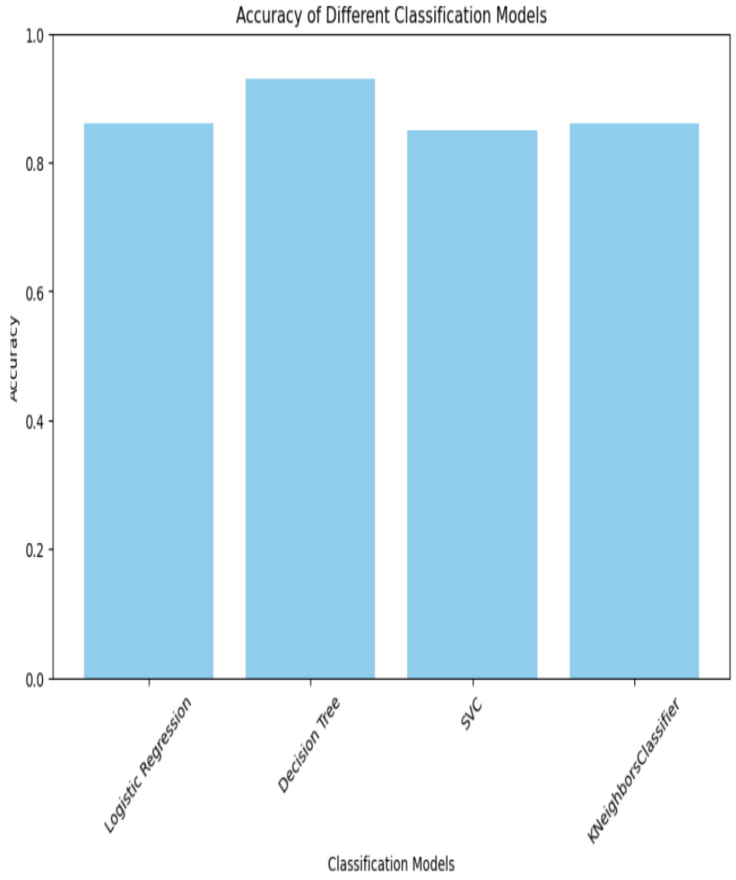
Find best Hyperparameter for SVM, Classification Trees and Logistic Regression

```
]: Logreg_cv_fit.best_params_

   Logreg_cv_fit.best_score_
   #Logreg_cv.best_score_

   print("tuned hpyerparameters :(best parameters) ",logreg_cv.best_params_)
   print("accuracy :",logreg_cv.best_score_)
```

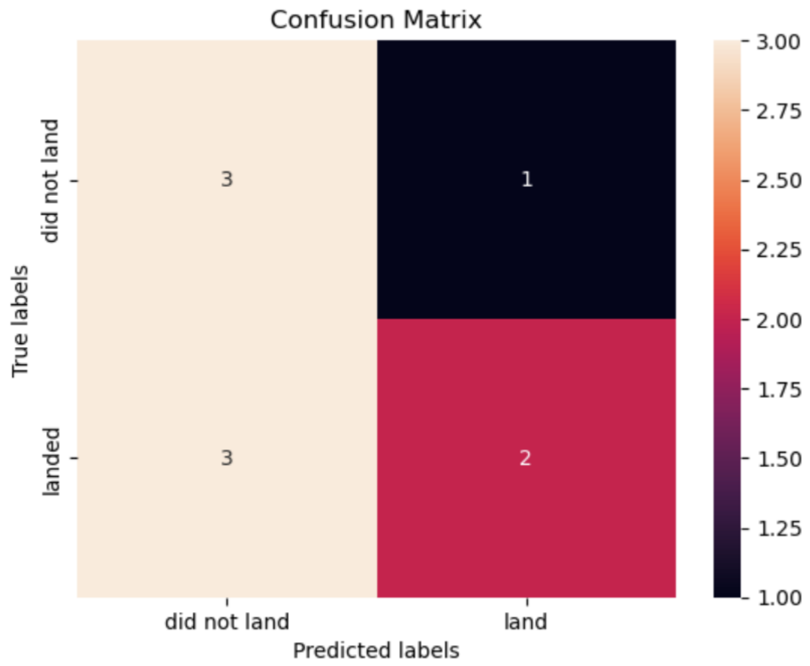
```
tuned hpyerparameters :(best parameters)  {'C': 0.1, 'penalty': 'l2', 'solver': 'lbfgs'}
accuracy : 0.8638888888888889
```

```
models = {
'Logistic Regression': LogisticRegression(),
'Decision Tree': DecisionTreeClassifier(),
'Support Vector Machine': SVC()
                    }
```

The best performing model is Decision Tree with an accuracy of 0.8888888888888888



- **Components of a Confusion Matrix**
  - **True Positives (TP)**: Correctly predicted positive instances.
  - **True Negatives (TN)**: Correctly predicted negative instances.
  - **False Positives (FP)**: Incorrectly predicted positive instances (Type I error).
  - **False Negatives (FN)**: Incorrectly predicted negative instances (Type II error).
- **Key Metrics Derived from a Confusion Matrix**
  - **Accuracy**: Proportion of total correct predictions. [\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}]
  - **Precision**: Proportion of true positive predictions among all positive predictions. [\text{Precision} = \frac{TP}{TP + FP}]
  - **Recall (Sensitivity)**: Proportion of true positive predictions among all actual positives. [\text{Recall} = \frac{TP}{TP + FN}]
  - **F1 Score**: Harmonic mean of precision and recall. [\text{F1 Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}]
- **Interpreting the Confusion Matrix**
  - Analysis of the balance between TP, TN, FP, and FN.
  - Discussion on the trade-offs between precision and recall.
  - Importance of context in interpreting these metrics (e.g., medical diagnosis vs. spam detection).
- **Conclusion**
  - Recap of the importance of confusion matrices.
  - Final thoughts on leveraging confusion matrices for better model evaluation.

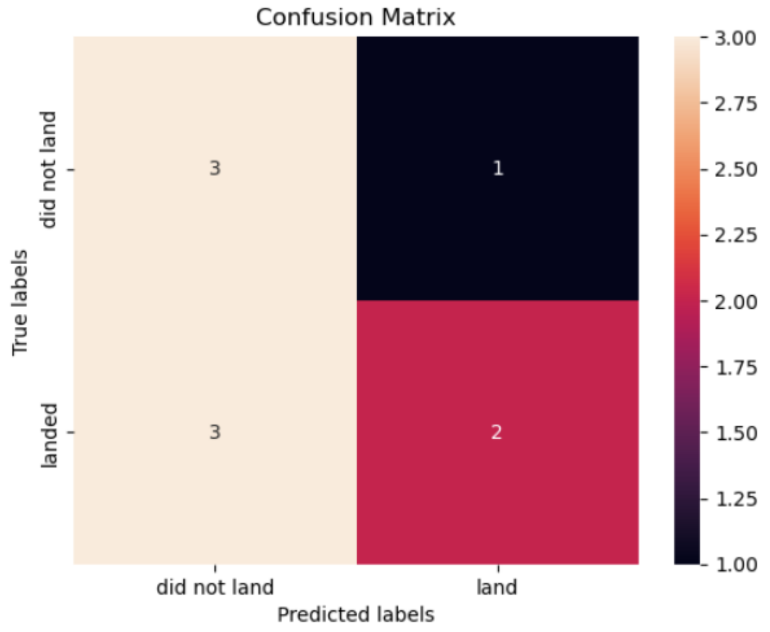


Confusion Matrix Explained

Confusion Matrix cont.

Examining the confusion matrix, we see that logistic regression can distinguish between the different classes.
We see that the problem is false positives.
Overview:
True Positive - 12 (True label is landed, Predicted label is also landed)
False Positive - 3 (True label is not landed, Predicted label is landed)

```python
def plot_confusion_matrix(y,y_predict):
    "this function plots the confusion matrix"
    from sklearn.metrics import confusion_matrix

    cm = confusion_matrix(y, y_predict)
    ax= plt.subplot()
    sns.heatmap(cm, annot=True, ax = ax); #annot=True to annotate cells
    ax.set_xlabel('Predicted labels')
    ax.set_ylabel('True labels')
    ax.set_title('Confusion Matrix');
    ax.xaxis.set_ticklabels(['did not land', 'land']); ax.yaxis.set_ticklabels(['did not land', 'landed'])
    plt.show()
```



Conclusions

Research

Model Performance: The models performed similarly on the test set with the decision tree model slightly outperforming.
Equator: Most of the sites are near the equator for an additional boost … saving fuel and boosters.
Launch near the coast
KSC LC-39A: Had the highest success rate. It also has a 100% success rate with payload <5,500 kg
Orbits ES-L1, GEO, HEO and SSO also have 100% success rates
Payload Mass: Across all launch sites, the higher the payload mass (kg) the higher the successs rate.