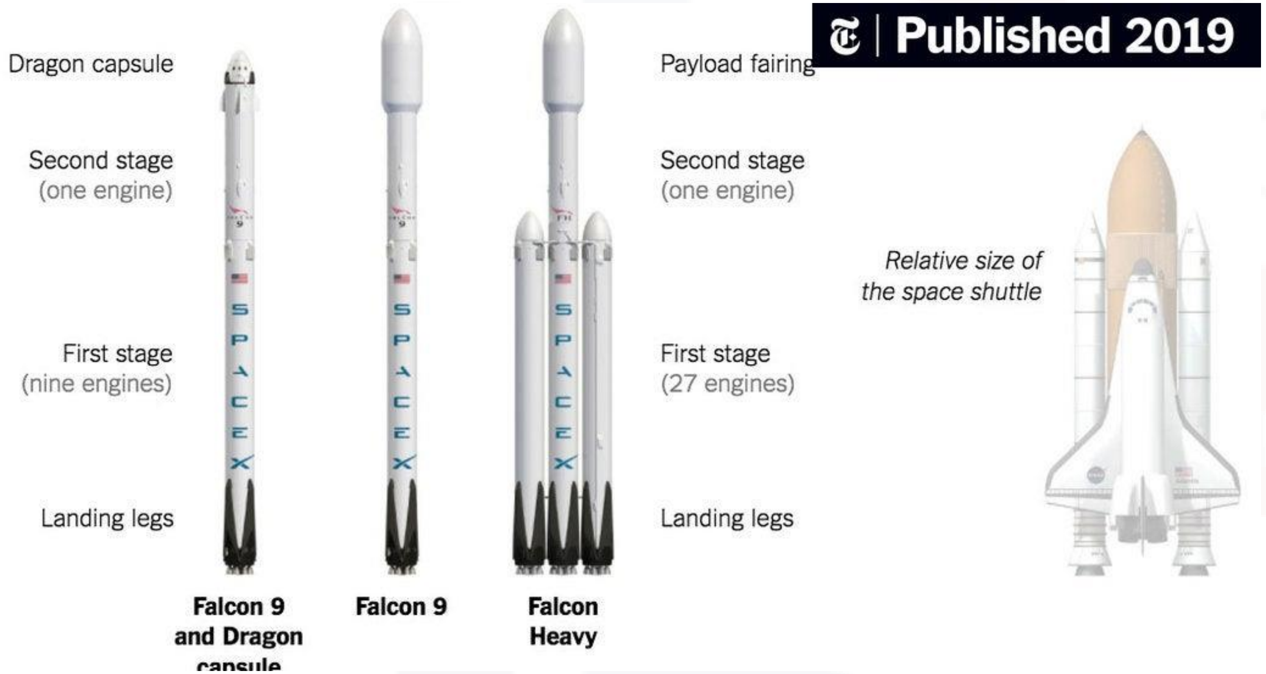
SPACE-Y INVESTIGATION INTO REUSABLE LAUNCH VEHICLES

Simbai Manyumwa

6 July 2024

OUTLINE



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

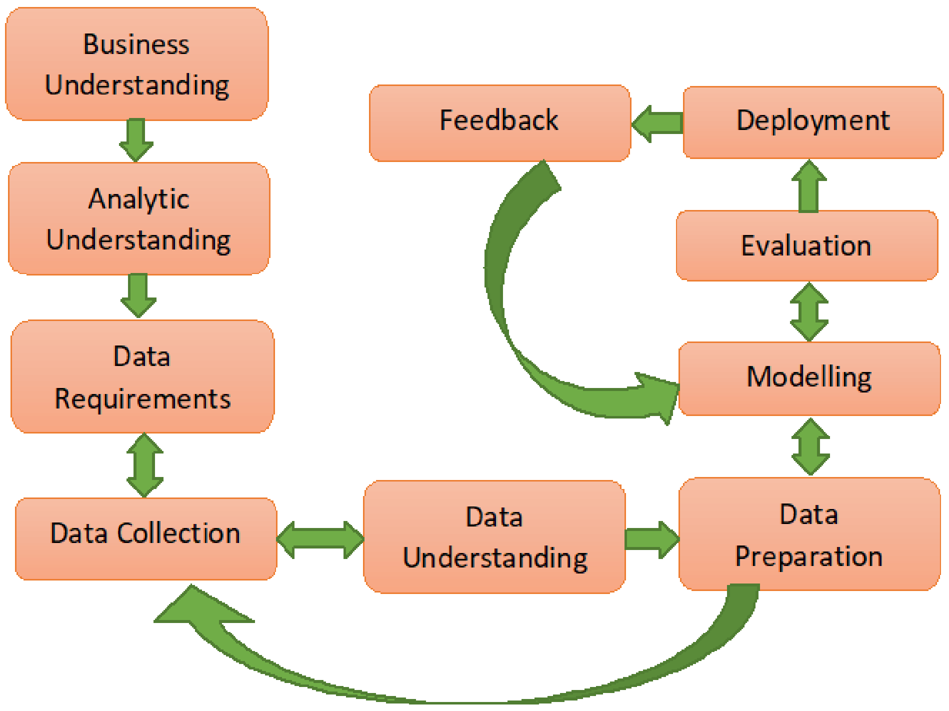
EXECUTIVE SUMMARY

- We set out to find the feasibility of Space Y using a similar version of the Falcon 9 booster used by SpaceX, specifically to see if we could predict a successful re-landing of the SpaceY first stage.
- We gathered data from SpaceX historical Falcon 9 launches including Payload Mass, Launch Site, Launch Outcomes and other factors around each launch.
- With the gathered data we applied various machine learning models and were able to come up with one that K Nearest Neighbor was the most accurate model in predicting the outcome of launches.

INTRODUCTION

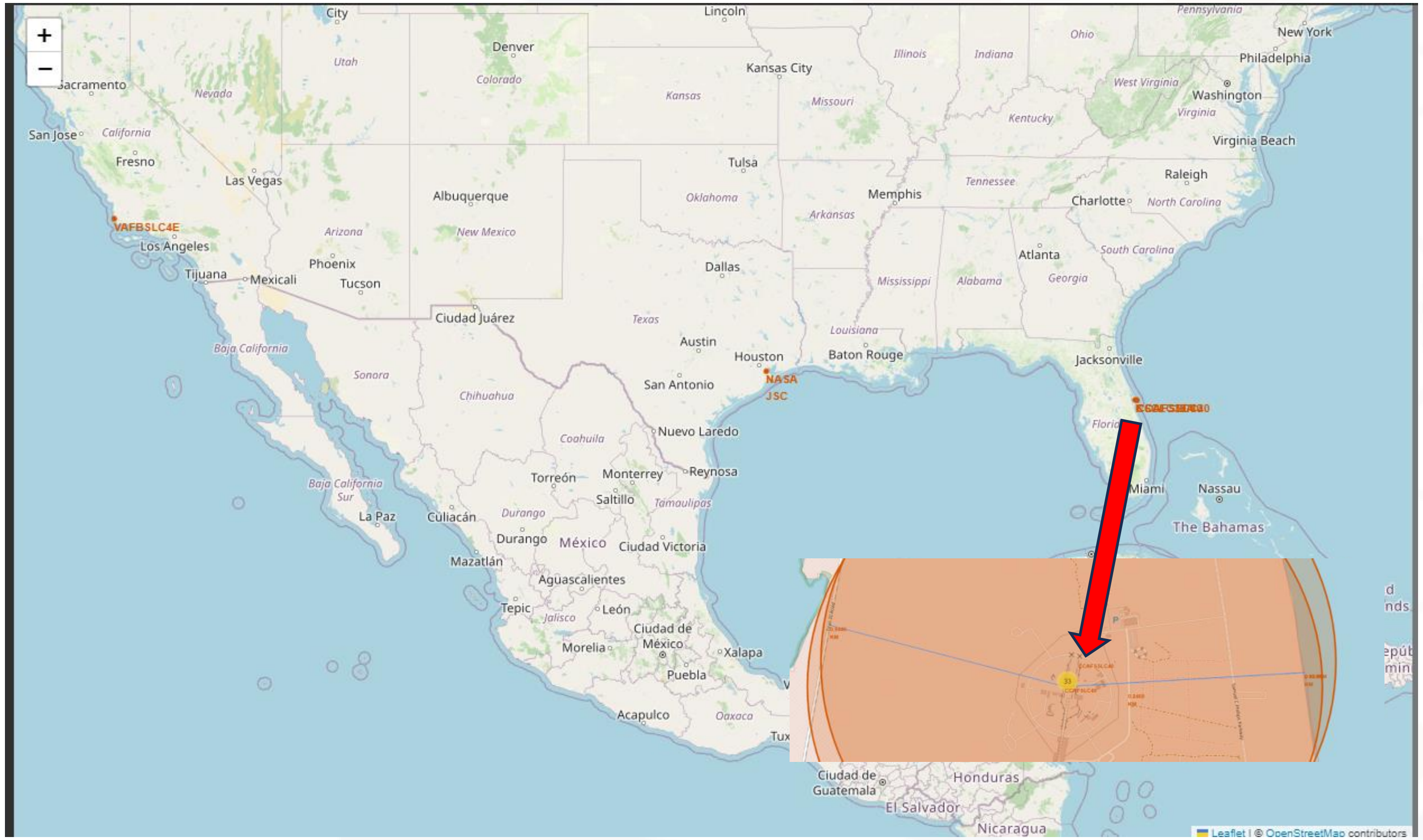
- The Falcon 9 designed and manufactured by SpaceX, following consultation Space Y tasked us with investigating what has made the Falcon 9 a success and how the success could be replicated for their own Booster Version.
- The biggest competitive advantage the Falcon9 has is its re-usability and we investigate the factors surrounding this advantage.
- We explore the results of the Falcon 9 Booster launches between 2010 and 2020, SpaceX were able to conduct a total of 90 launches over 3 sites within this period.

METHODOLOGY



- Problem - investigate the factors surrounding the Falcon 9's success so we can see it they can be replicated. This meant looking int what factors contribute to the success of the Falcon 9 and what factors contribute to its failure to land.
- Data requirements were the information around all the launches for the Falcon 9 available. This was done mostly from various web scrapes that provided the launch details of the Falcon 9.
- Applying various comparisons we were able to see the relationships between the success and failure with other flight details.
- We then standardized the data so we could apply different models and test what would best apply to this situation. We split the data into train and test data (80:20 split).
- After applying four models we found the K Nearest Neighbour model to be the most accurate. This is the model we would like to put forward your evaluation.

RESULTS- source- https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DS0321EN-SkillsNetwork/labs/module_2/data/Spacex.csv

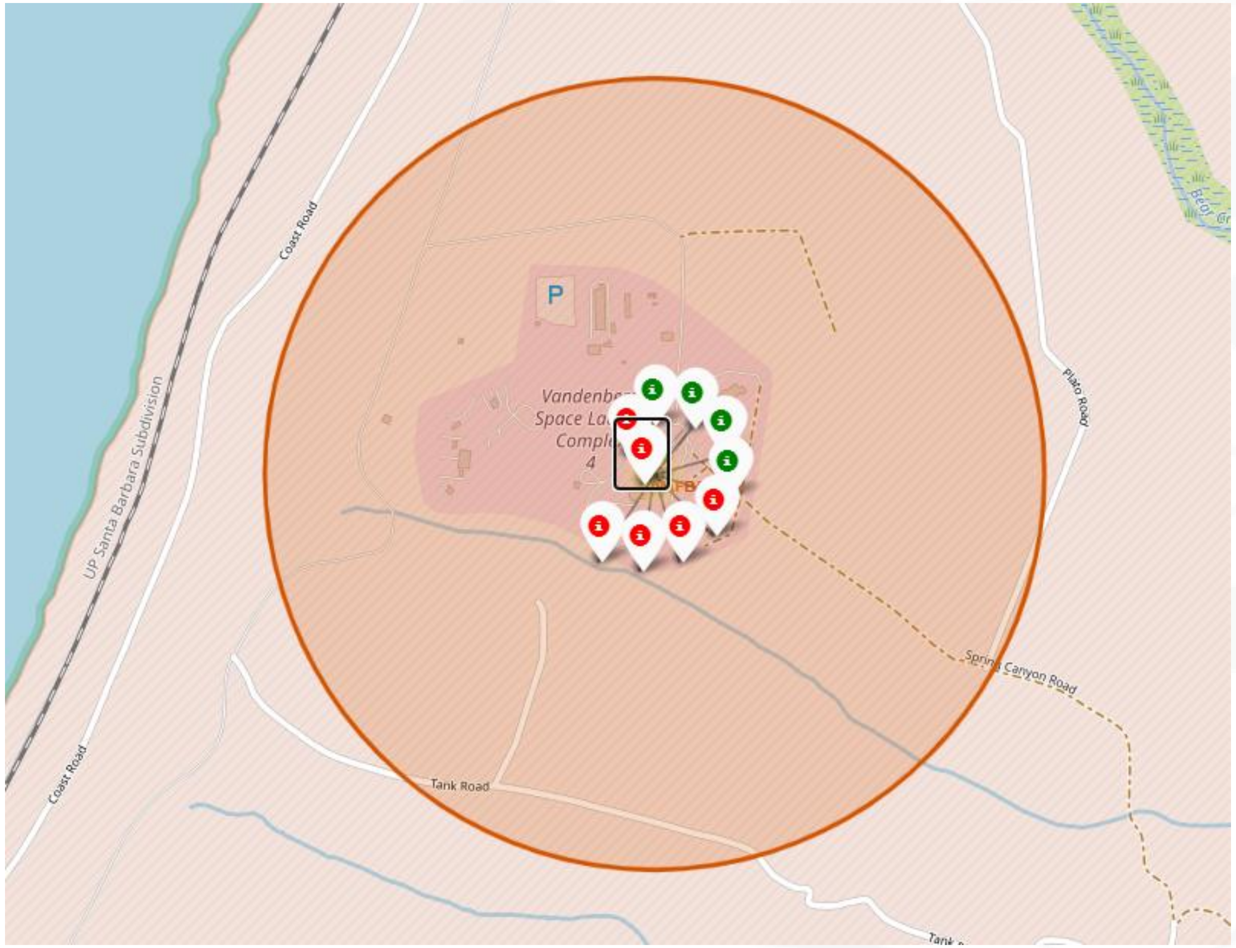


Our finding show that the Falcon 9 booster has had 90 launches from four locations:

- **VAFB SLC-4E** - Vandenberg Space Force Base, Santa Maria
- **CCAFS LC-40** - Cape Canaveral Space Force Station, Cape Canaveral
- **CCAFS SLC-40** - Cape Canaveral Space Force Station, Cape Canaveral
- **KSC LC-39A** - near Merrit Island National Wildlife Refuge

All launch sites are in close proximity to the coastline and have relatively close access to a major road or railway

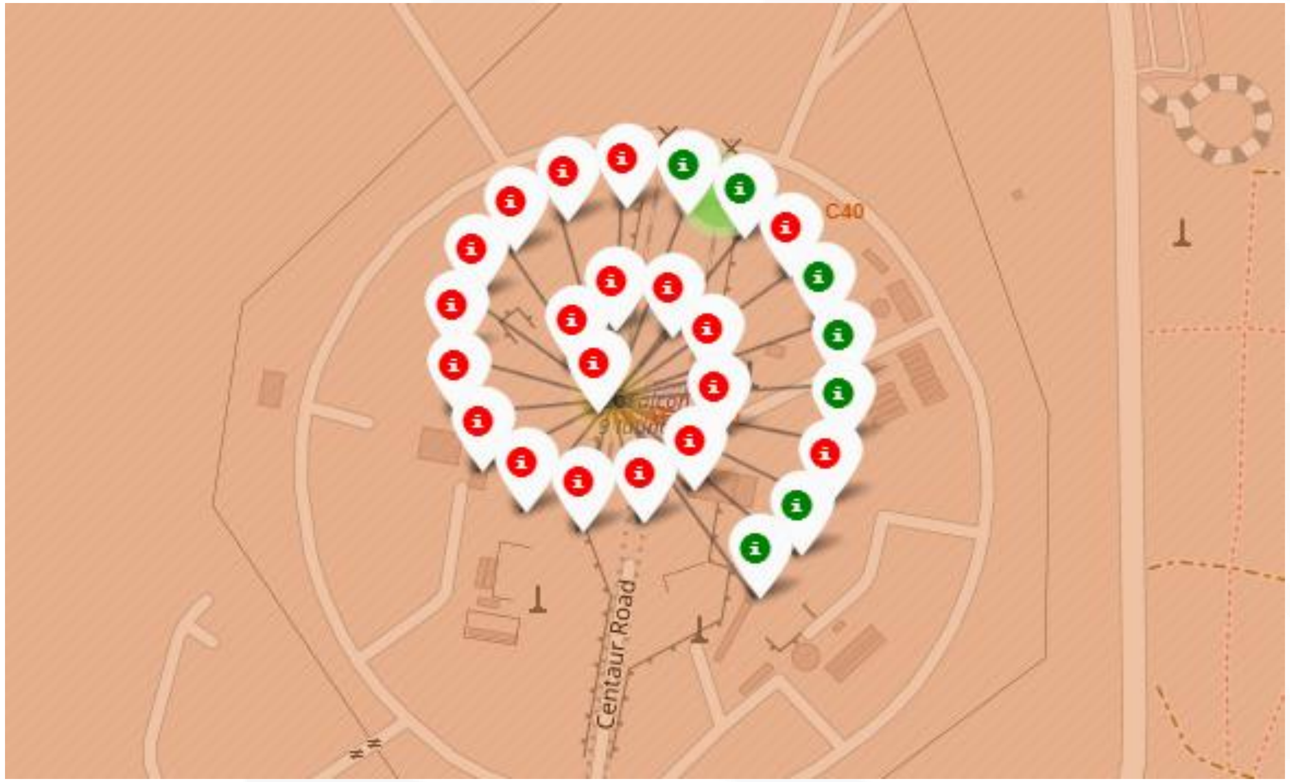
VAFB SLC-4E - Vandenberg Space Force Base, Santa Maria



In the period investigated there were 10 launches of the Falcon 9 -

- 4 resulted in a successful re-landing
- 6 were unsuccessful

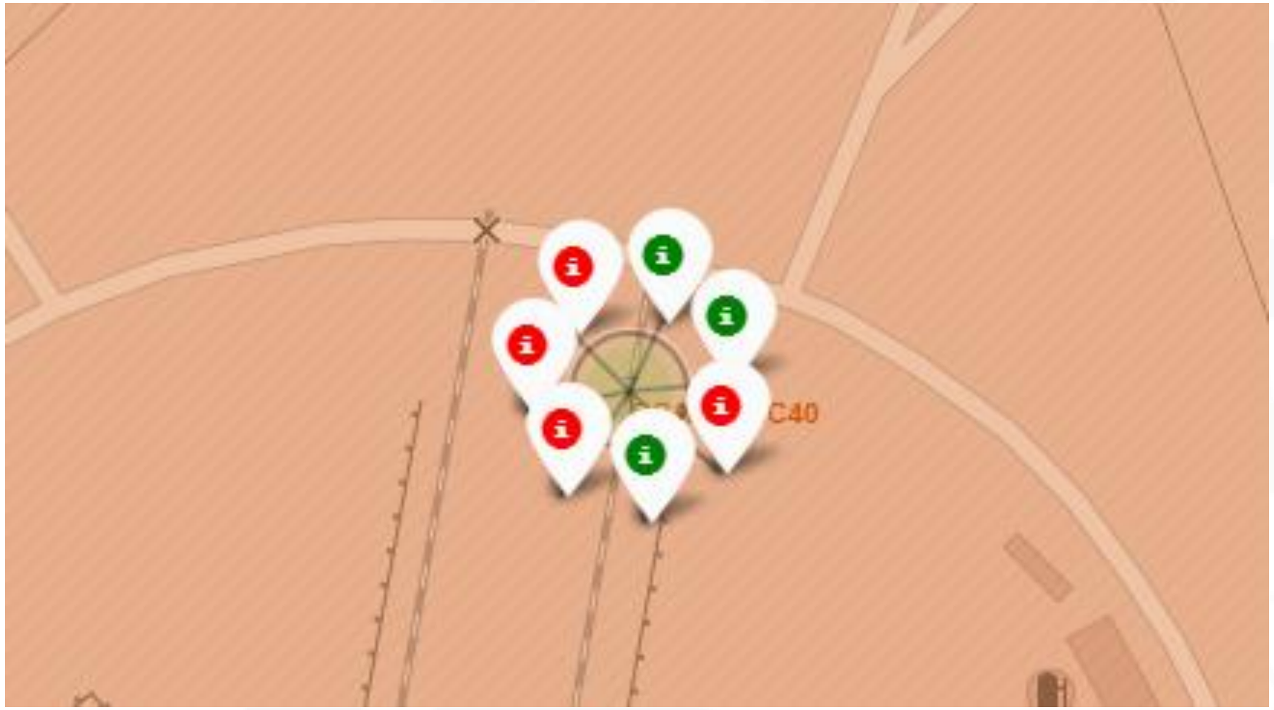
CCAFS LC-40 - Cape Canaveral Space Force Station, Cape Canaveral



26 launches have been from this location

- 19 failed landings
- 7 successful

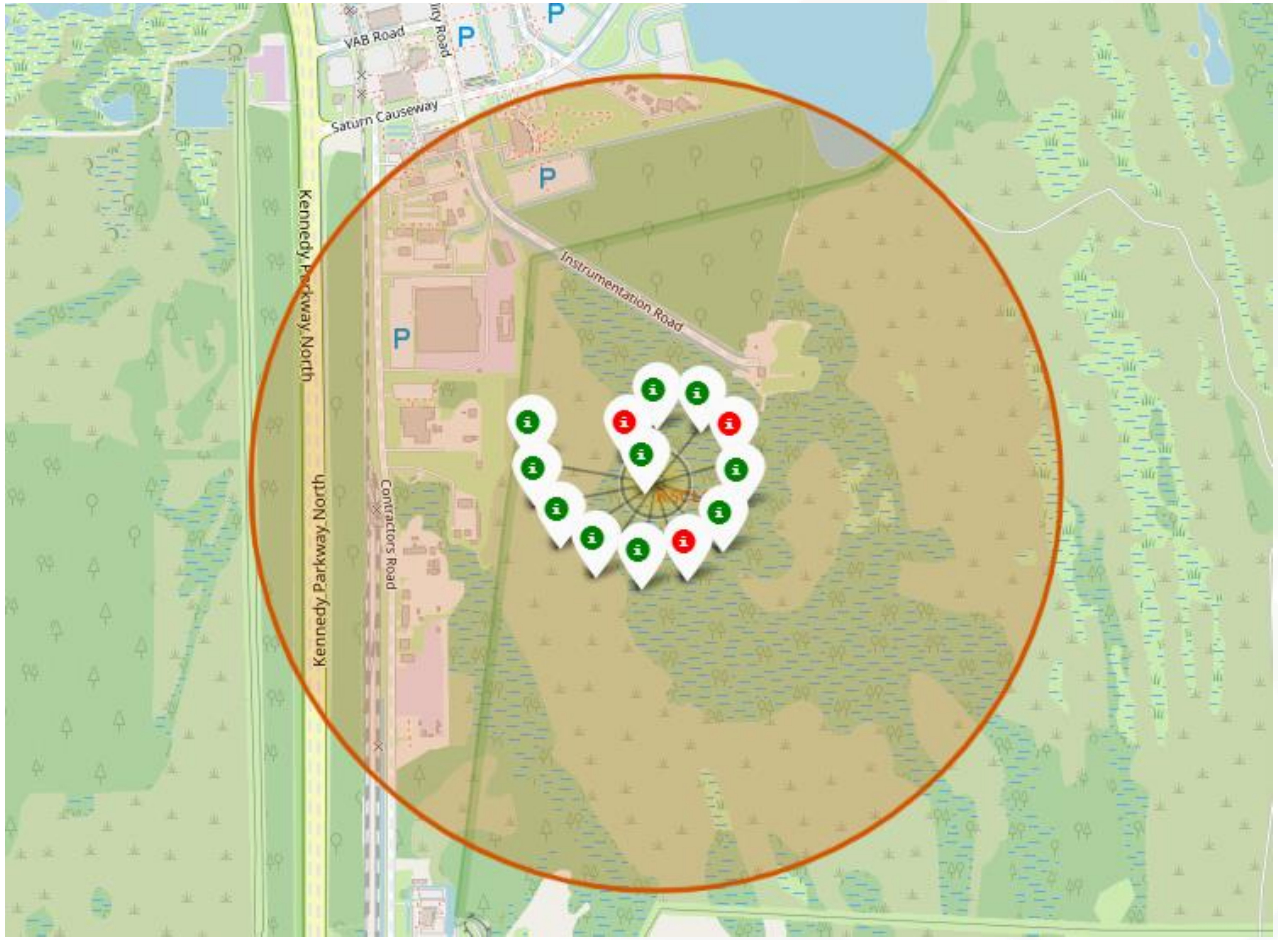
CCAFS SLC-40 - Cape Canaveral Space Force Station, Cape Canaveral



7 launches from this location

- 3 successful
- 7 failed

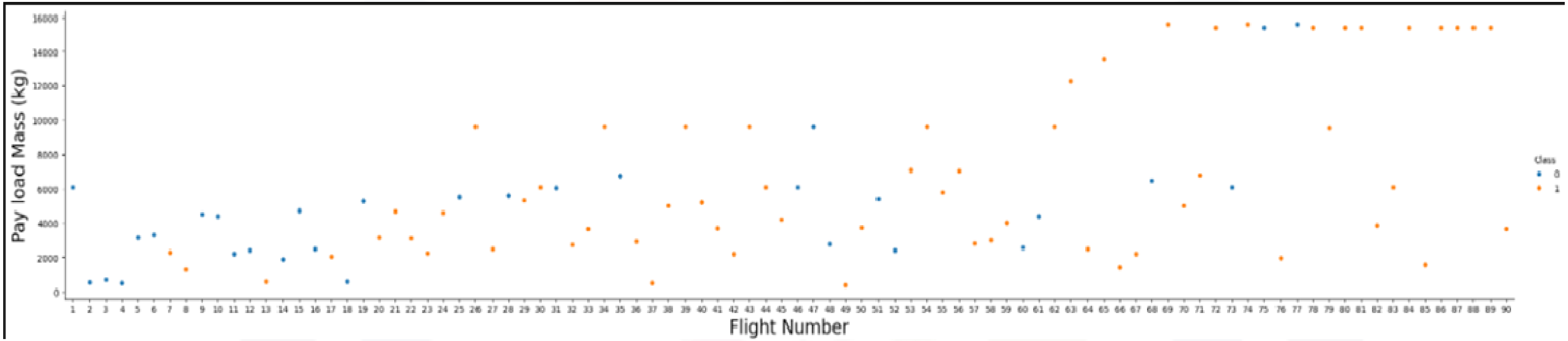
KSC LC-39A - near Merrit Island National Wildlife Refuge



X launches from this location

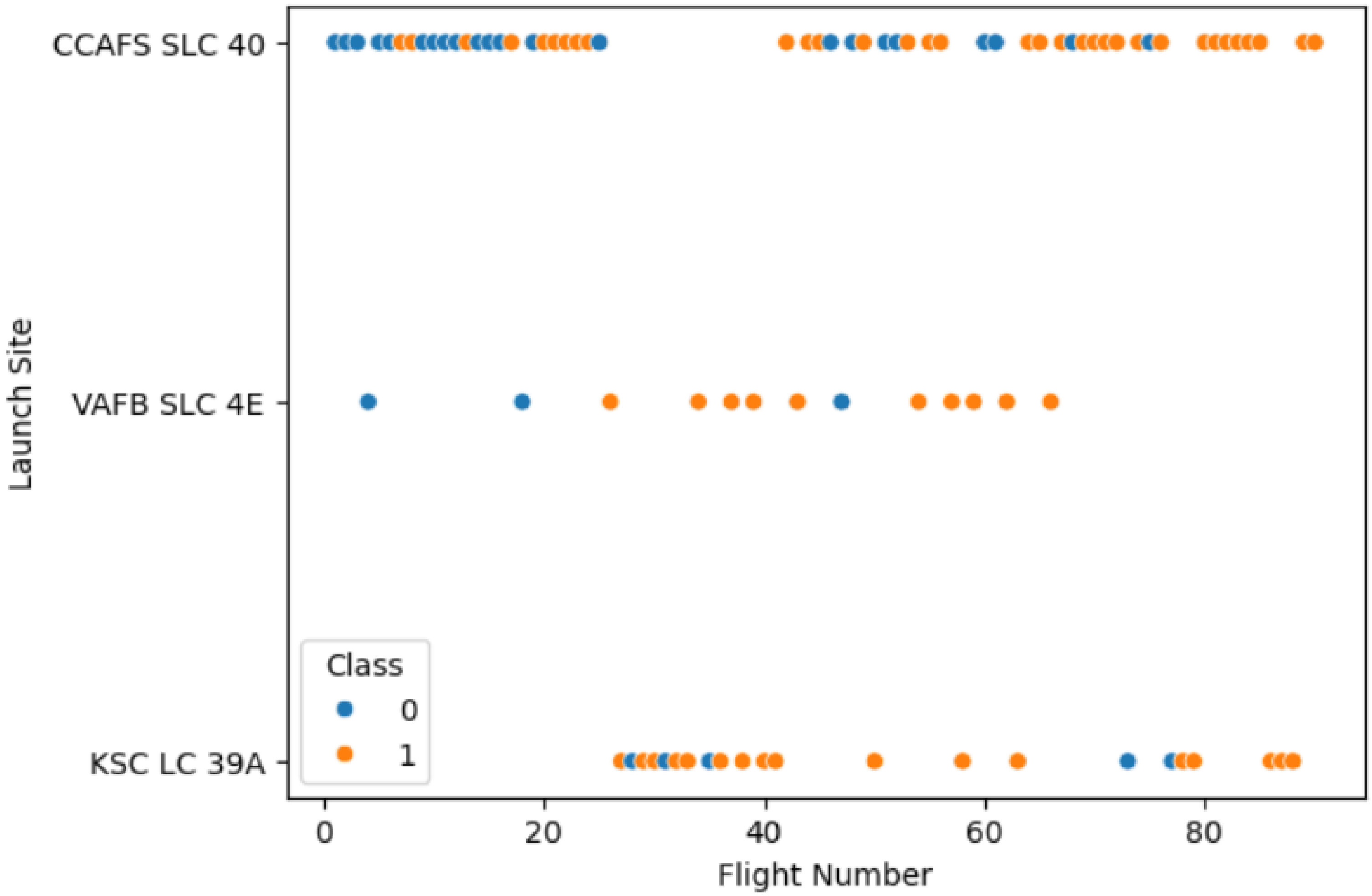
- 10 successful landings
- 3 failed

SCATTER PLOT COMPARING PAYLOAD MASS AND FLIGHT NUMBER(Class1=Success, Class 0=Failure)



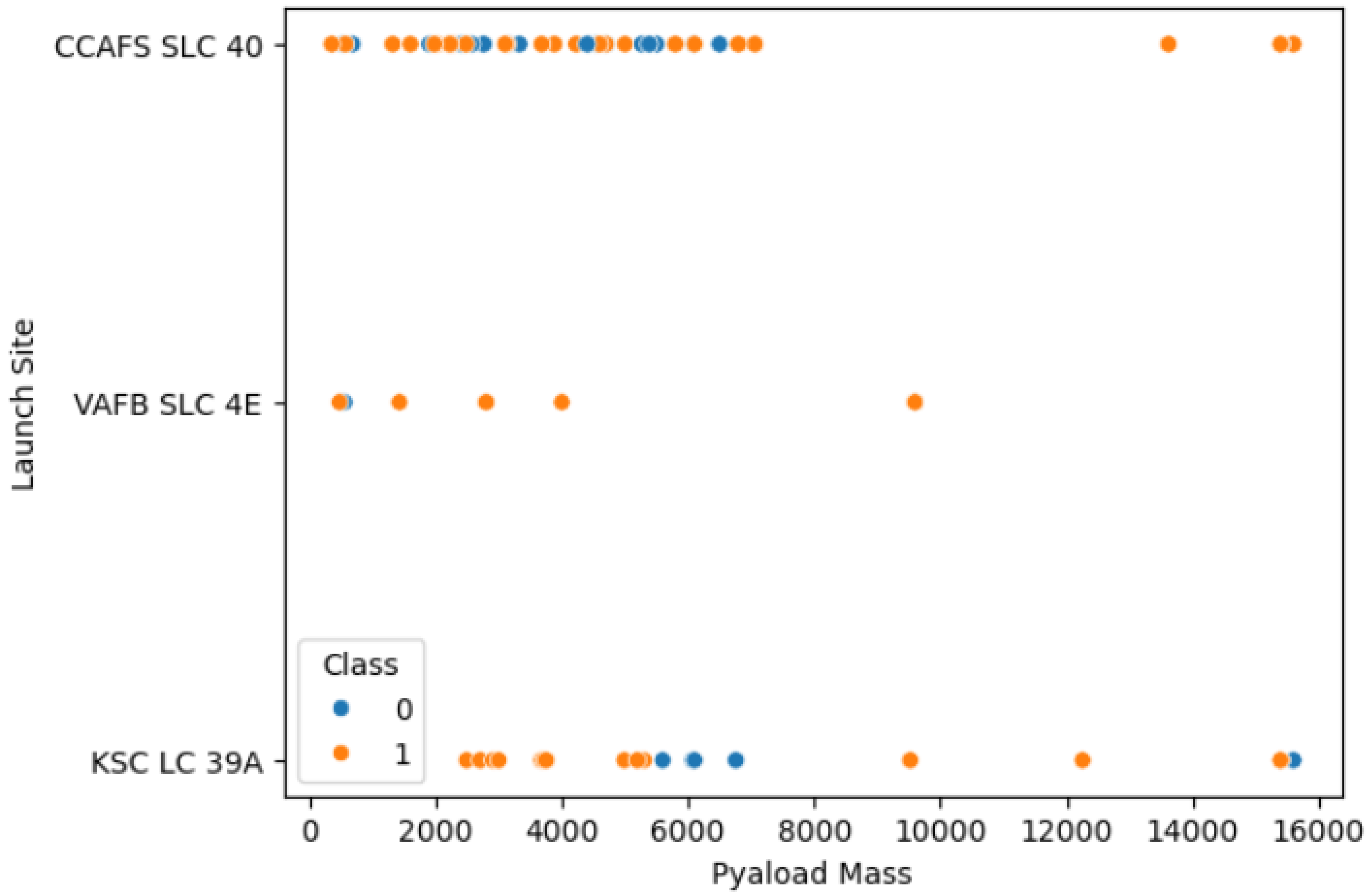
In this plot we can see the successes(class 1) have increased with time also the launches with a heavier payload mass are generally more successful. This can be attributed to the fact that Space X have been cnstantly improving on their initial design.

SCATTER PLOT COMPARING LAUNCHSITE AND FLIGHT NUMBER(Class1=Success, Class 0=Failure)



As highlighted previously the majority of launches are from CCAFS SLC 40. This plot also supports the previous statement that the success rate has been improving with time.

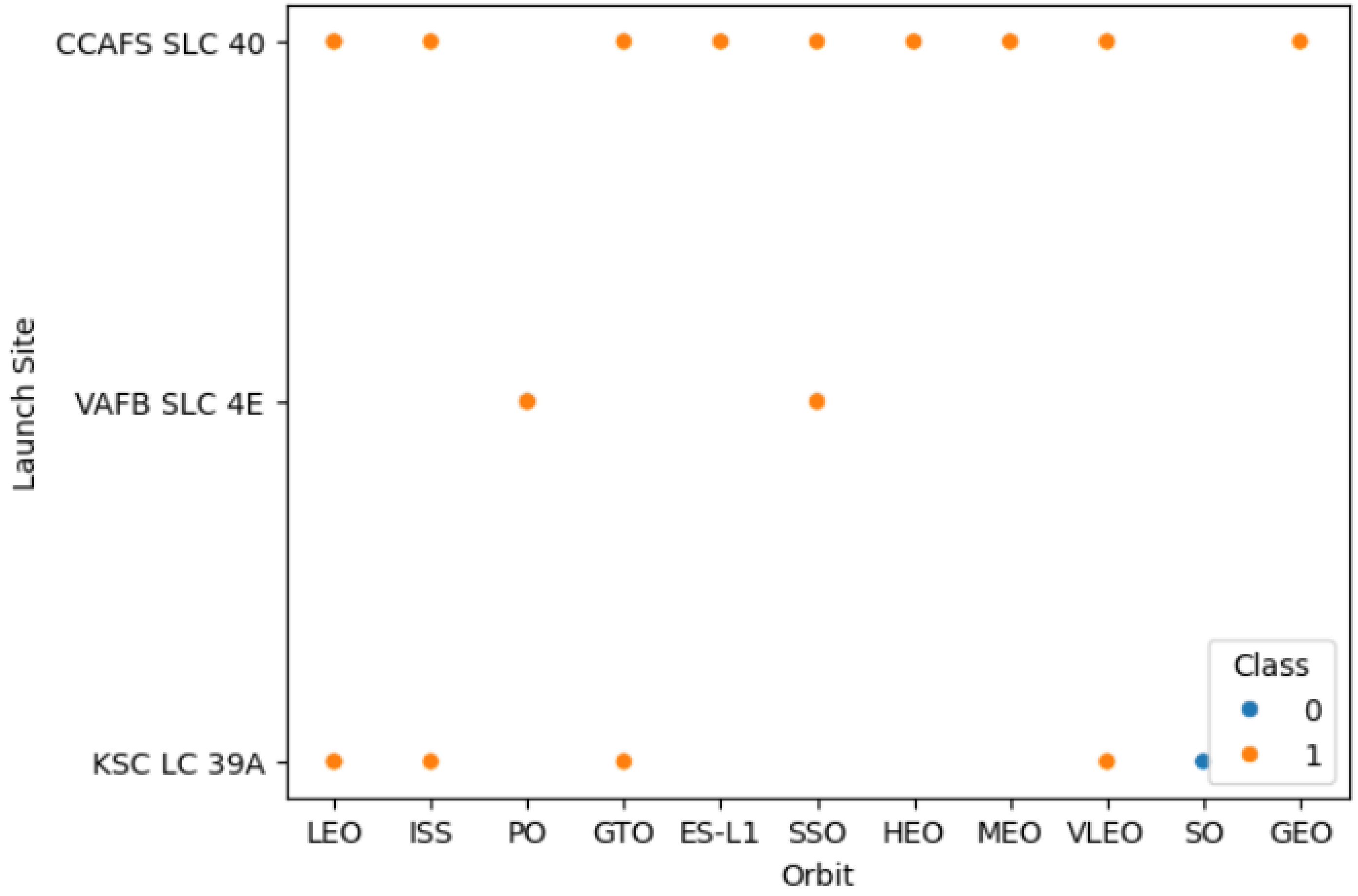
SCATTER PLOT COMPARING LAUNCHSITE AND PAYLOAD MASS(Class1=Success, Class 0=Failure)



The majority of launches
are for a payload mass below 1000kg.

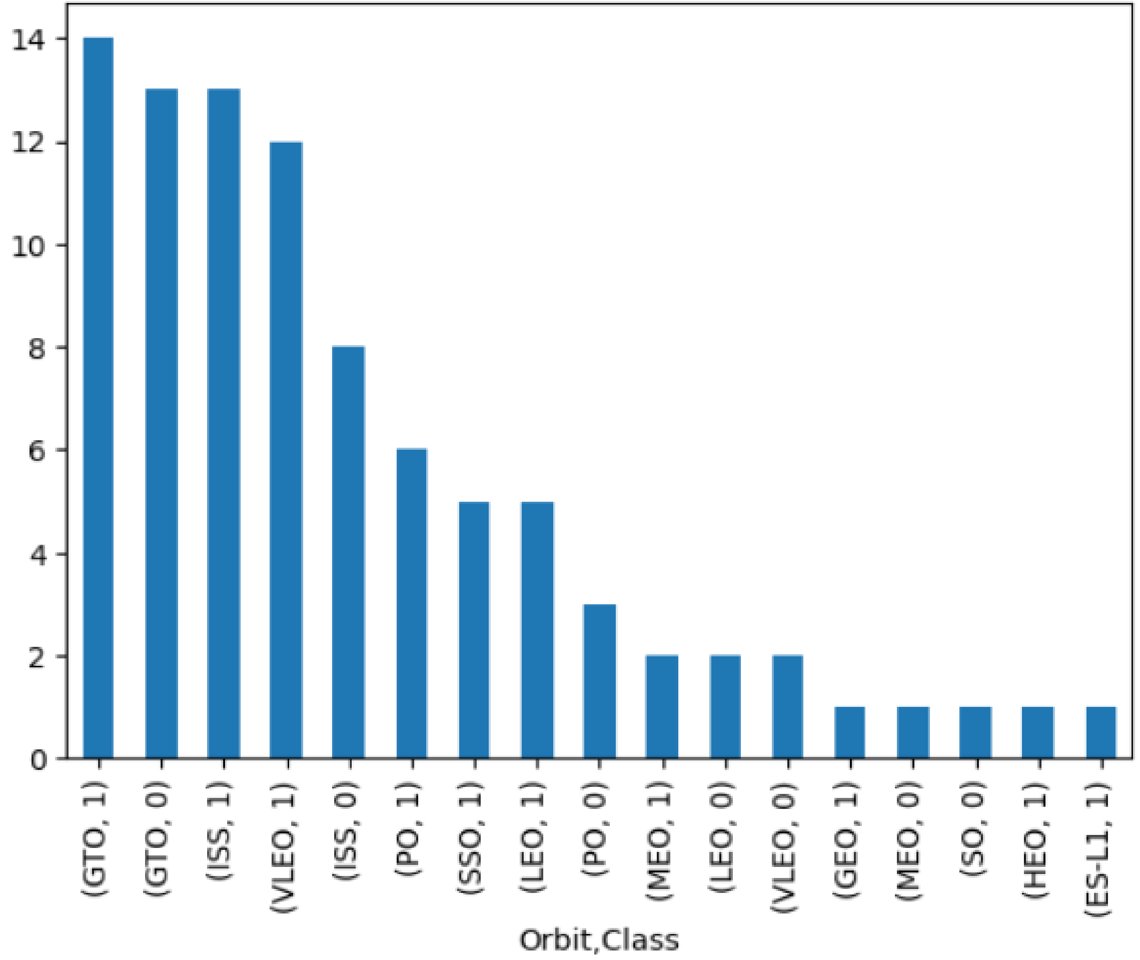
- All launches over 10000kg from site CCAFS SLC 40 were successful
- All launches
with a payload mass greater
that 1500kg from VAFB SLC 4E were successful
- The failed ;aunches
from site KSC LC 39A were when the payload mass was around 6000kg or greater than 15500kg

SCATTER PLOT COMPARING LAUNCHSITE AND ORBIT (Class1=Success, Class 0=Failure)



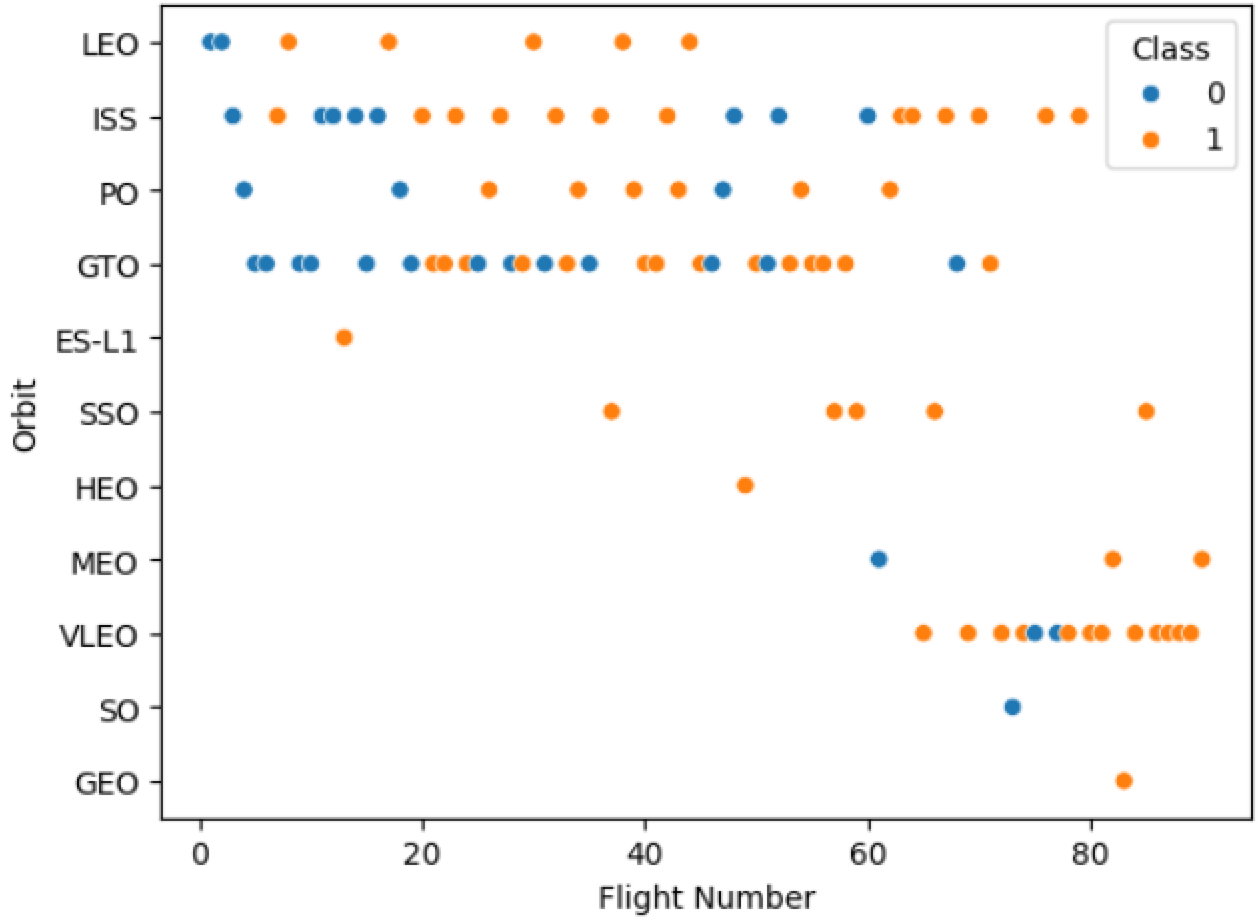
Site CCAFS SLC 40 can handle launches to any orbit. However the other 2 sites are less versatile.

BAR PLOT SHOWING ORBIT SUCCESS RATE (Class1=Success, Class 0=Failure)



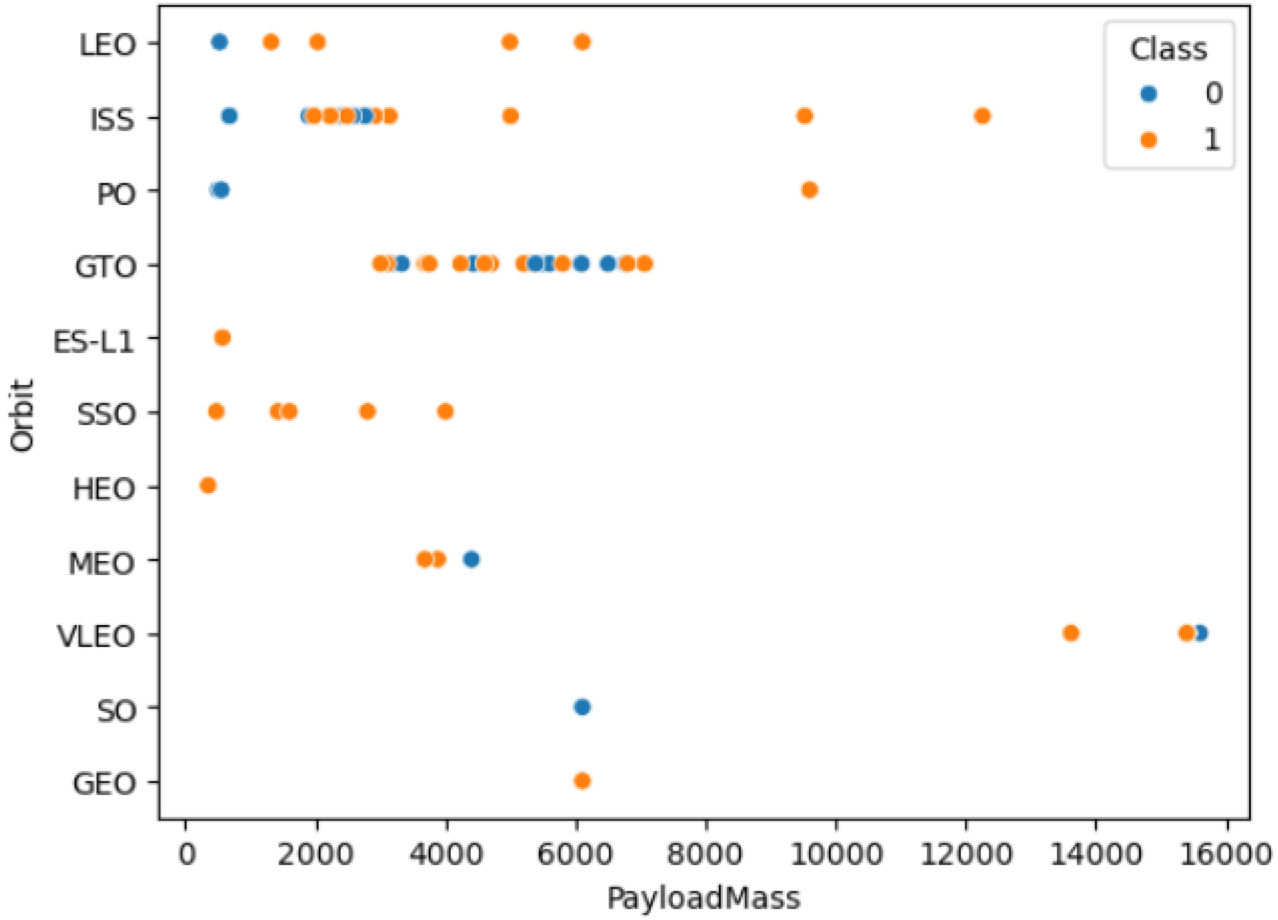
GTO orbit has the most successes 14 also the most fails 13 and is the most common orbit for launches.

SCATTER PLOT COMPARING ORBIT AND FLIGHT NUMBER (Class1=Success, Class 0=Failure)



In recent launches the most common orbit attempted has been for VLEO and ISS orbits. Though the most common remains GTO.

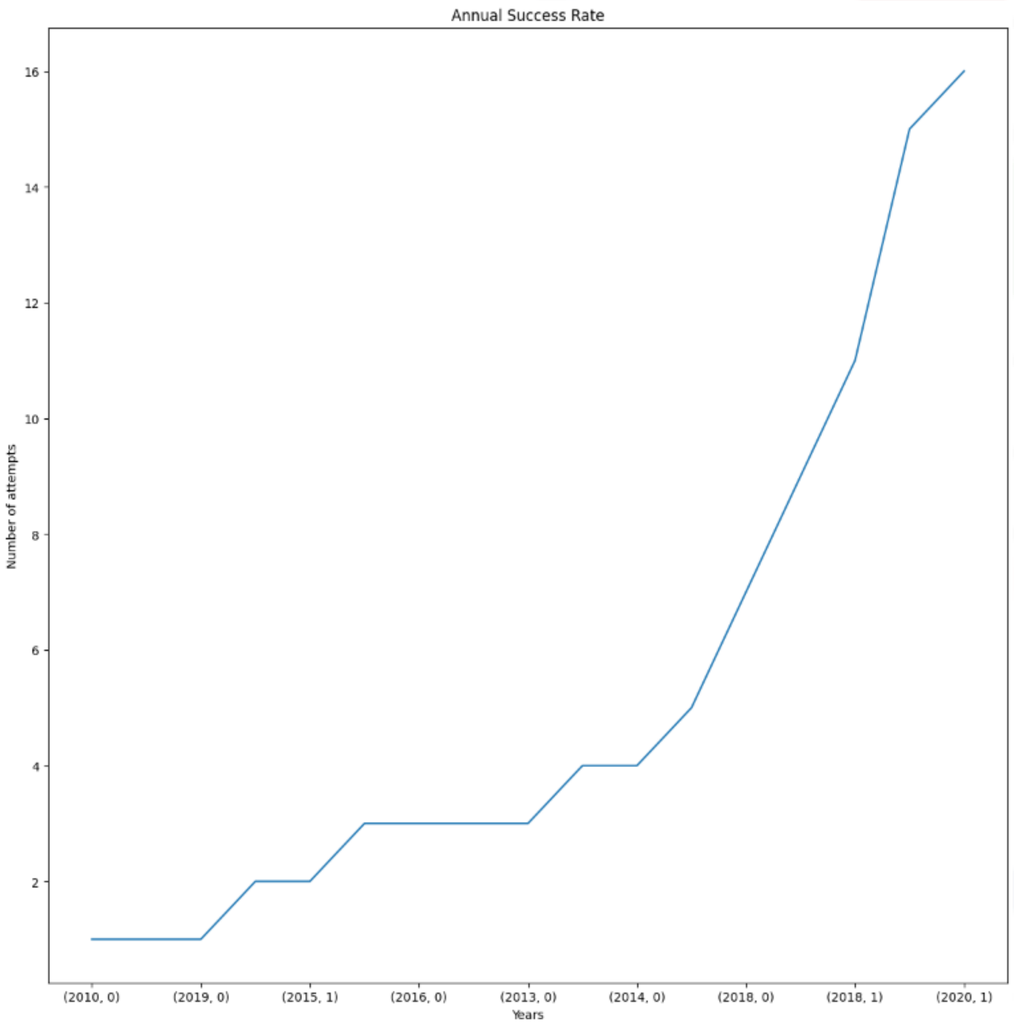
SCATTER PLOT COMPARING ORBIT AND PAYLOAD MASS (Class1=Success, Class 0=Failure)



In this plot we see that most launches have a payload mass between 2000-8000kg.

- ISS orbit success is 100% over 3000kg
- LEO orbit success rate is 100% over 1500kg
- SSOorbit has a 100% success rate

ANNUAL SUCCESS RATE



We can clearly see in this line graph that the launches have been far more successful as time has passed. The first spike is in 2013 reaching 16 successful launches in 202. This again supports the fact that the modifications to the booster have been very successful.

THE DATA

	FlightNumber	Date	BoosterVersion	PayloadMass	Orbit	LaunchSite	Outcome
1	1	2010	Falcon 9	6104.959411764706	LEO	CCAFS SLC 40	None None
2	2	2012	Falcon 9	525.0	LEO	CCAFS SLC 40	None None
3	3	2013	Falcon 9	677.0	ISS	CCAFS SLC 40	None None
4	4	2013	Falcon 9	500.0	PO	VAFB SLC 4E	False Ocean
5	5	2013	Falcon 9	3170.0	GTO	CCAFS SLC 40	None None
6	6	2014	Falcon 9	3325.0	GTO	CCAFS SLC 40	None None
7	7	2014	Falcon 9	2296.0	ISS	CCAFS SLC 40	True Ocean
8	8	2014	Falcon 9	1316.0	LEO	CCAFS SLC 40	True Ocean
9	9	2014	Falcon 9	4535.0	GTO	CCAFS SLC 40	None None
10	10	2014	Falcon 9	4428.0	GTO	CCAFS SLC 40	None None
11	11	2014	Falcon 9	2216.0	ISS	CCAFS SLC 40	False Ocean
12	12	2015	Falcon 9	2395.0	ISS	CCAFS SLC 40	False ASDS
13	13	2015	Falcon 9	570.0	ES-L1	CCAFS SLC 40	True Ocean
14	14	2015	Falcon 9	1898.0	ISS	CCAFS SLC 40	False ASDS
15	15	2015	Falcon 9	4707.0	GTO	CCAFS SLC 40	None None
16	16	2015	Falcon 9	2477.0	ISS	CCAFS SLC 40	None ASDS
17	17	2015	Falcon 9	2034.0	LEO	CCAFS SLC 40	True RTLS
18	18	2016	Falcon 9	553.0	PO	VAFB SLC 4E	False ASDS
19	19	2016	Falcon 9	5271.0	GTO	CCAFS SLC 40	False ASDS

Using a jupyter notebook we were able to scrape various sources and organise it into meaningful data. Links of the sorted data are below:

- [Space X data]

THE DATA -SQLITE3

	FlightNumber	Date	BoosterVersion	PayloadMass	Orbit	LaunchSite	Outcome
1	1	2010	Falcon 9	6104.959411764706	LEO	CCAFS SLC 40	None None
2	2	2012	Falcon 9	525.0	LEO	CCAFS SLC 40	None None
3	3	2013	Falcon 9	677.0	ISS	CCAFS SLC 40	None None
4	4	2013	Falcon 9	500.0	PO	VAFB SLC 4E	False Ocean
5	5	2013	Falcon 9	3170.0	GTO	CCAFS SLC 40	None None
6	6	2014	Falcon 9	3325.0	GTO	CCAFS SLC 40	None None
7	7	2014	Falcon 9	2296.0	ISS	CCAFS SLC 40	True Ocean
8	8	2014	Falcon 9	1316.0	LEO	CCAFS SLC 40	True Ocean
9	9	2014	Falcon 9	4535.0	GTO	CCAFS SLC 40	None None
10	10	2014	Falcon 9	4428.0	GTO	CCAFS SLC 40	None None
11	11	2014	Falcon 9	2216.0	ISS	CCAFS SLC 40	False Ocean
12	12	2015	Falcon 9	2395.0	ISS	CCAFS SLC 40	False ASDS
13	13	2015	Falcon 9	570.0	ES-L1	CCAFS SLC 40	True Ocean
14	14	2015	Falcon 9	1898.0	ISS	CCAFS SLC 40	False ASDS
15	15	2015	Falcon 9	4707.0	GTO	CCAFS SLC 40	None None
16	16	2015	Falcon 9	2477.0	ISS	CCAFS SLC 40	None ASDS
17	17	2015	Falcon 9	2034.0	LEO	CCAFS SLC 40	True RTLS
18	18	2016	Falcon 9	553.0	PO	VAFB SLC 4E	False ASDS
19	19	2016	Falcon 9	5271.0	GTO	CCAFS SLC 40	False ASDS

using our jupyter notebook with SQLite3 we were able to extract further insights.

THE DATA -SQLITE3

```
%sql select max(PAYLOAD_MASS__KG_) from SPACEXTABLE
```

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

max(PAYLOAD_MASS__KG_)
15600

```
%sql select Booster_Version from SPACEXTABLE where PAYLOAD_MASS__KG_ = '15600'
```

```
 * 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
F9 B5 B1060.2
F9 B5 B1058.3
F9 B5 B1051.6
F9 B5 B1060.3
F9 B5 B1049.7

Which Booster versions were able to handle the maximum payload(12 version could handle 15600kg).

Or what the average payload mass is for the F9 1v1 Booster- 2928.4kg

```
%sql select avg(PAYLOAD_MASS__KG_) from SPACEXTABLE where Booster_Version ='F9 v1.1'
```

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

avg(PAYLOAD_MASS__KG_)
2928.4

THE DATA -SQLITE3

```
%sql select Date, Landing_Outcome, Booster_Version, Launch_Site from SPACEXTABLE where Landing_Outcome ='Failure (drone ship)'
 * sqlite:///my_data1.db
Done.
```

Date	Landing_Outcome	Booster_Version	Launch_Site
2015-01-10	Failure (drone ship)	F9 v1.1 B1012	CCAFS LC-40
2015-04-14	Failure (drone ship)	F9 v1.1 B1015	CCAFS LC-40
2016-01-17	Failure (drone ship)	F9 v1.1 B1017	VAFB SLC-4E
2016-03-04	Failure (drone ship)	F9 FT B1020	CCAFS LC-40
2016-06-15	Failure (drone ship)	F9 FT B1024	CCAFS LC-40

Using a SQLite3 we were able to extract other data like;

- The booster versions used for drone landings, with a certain payload mass
- Along with the Booster versions that failed to land on a drone ship
- The 4 launch sites studied

```
%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

```
%sql select Booster_Version from SPACEXTABLE where (landing_outcome='Success (drone ship)') 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

SKILLS NETWORK

MACHINE LEARNING

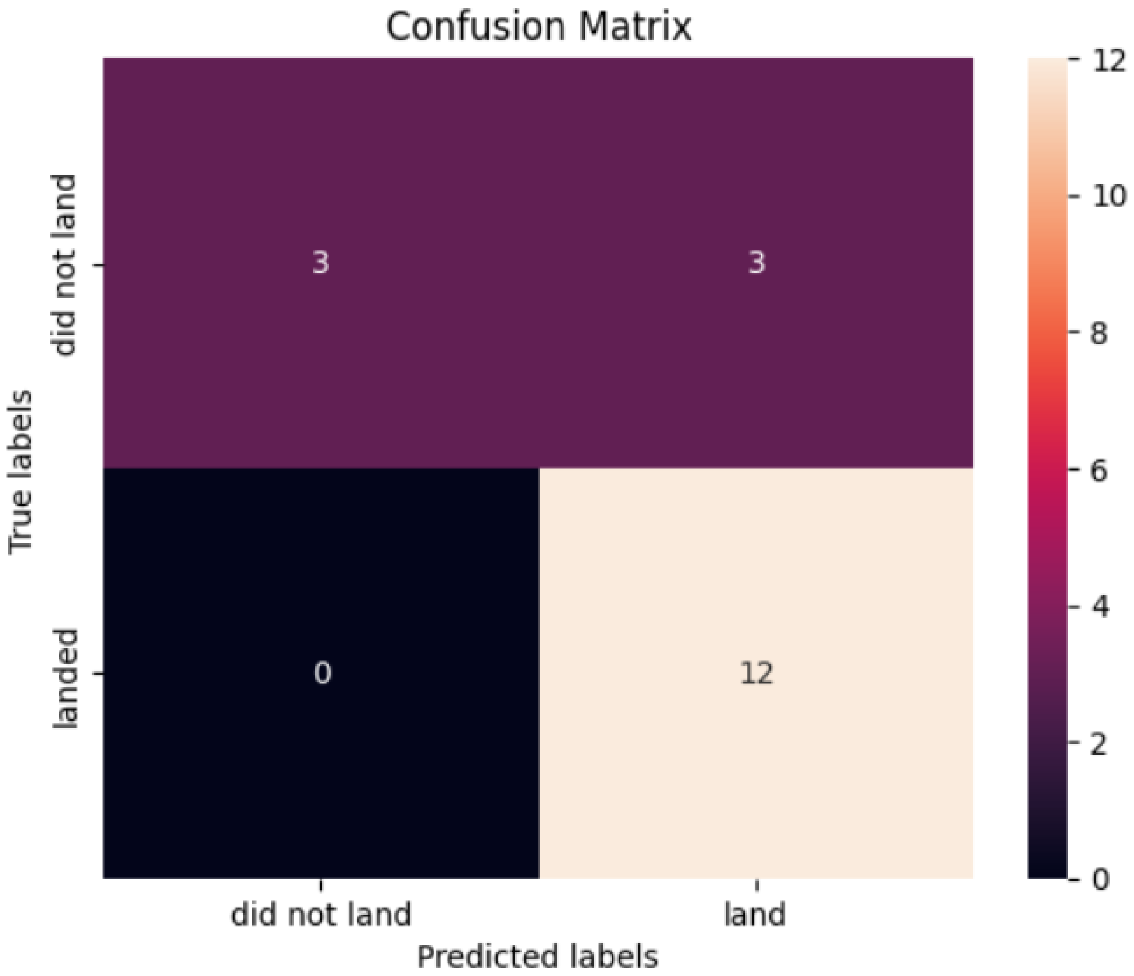
As discussed earlier we applied 4 models to the data to try and see which would yield the most accurate results. We then split our data into train and test data (80 % to train the model and 20% to test it) and applied each model and drew up a confusion matrix.

The four models were:

1. A logistic Regression Model
2. Support Vector Machine Model
3. Decision Tree Classifier model
4. K Nearest Neighbour model

MACHINE LEARNING - LOGISTIC REGRESSION

- Logistic regression estimates the probability of an event occurring, such as booster landing or booster not landing, based on a given data set of independent variables



- we can see from the confusion matrix under this model
  - It correctly predicted that 3 would not land and that 12 would land
  - It incorrectly predicted that 3 would land
  - The main concern is the bottom left had no predictions at all.

```
logreg_cv=GridSearchCV(lr,parameters,scoring='accuracy',cv=10)
```

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

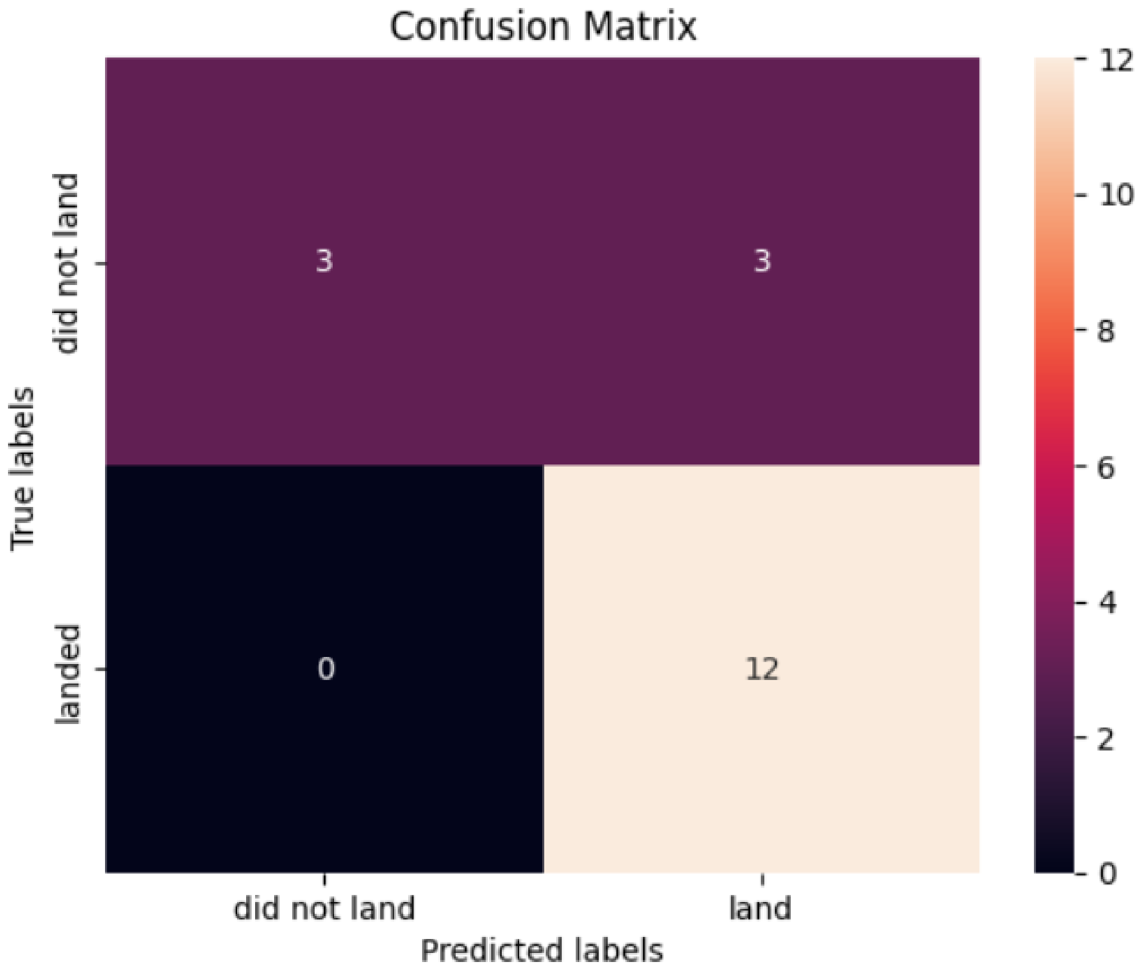
```
tuned hpyerparameters :(best parameters)  {'C': 0.01, 'penalty': 'l2', 'solver': 'lbfgs'}
accuracy : 0.8464285714285713
```

```
logreg_cv.score(X_test, Y_test)
```

```
0.8333333333333334
```

MACHINE LEARNING - SUPPORT VECTOR MACHINE

- A support vector machine (SVM) is a supervised machine learning algorithm that classifies data by finding an optimal line or hyperplane that maximizes the distance between each class in an N-dimensional space.

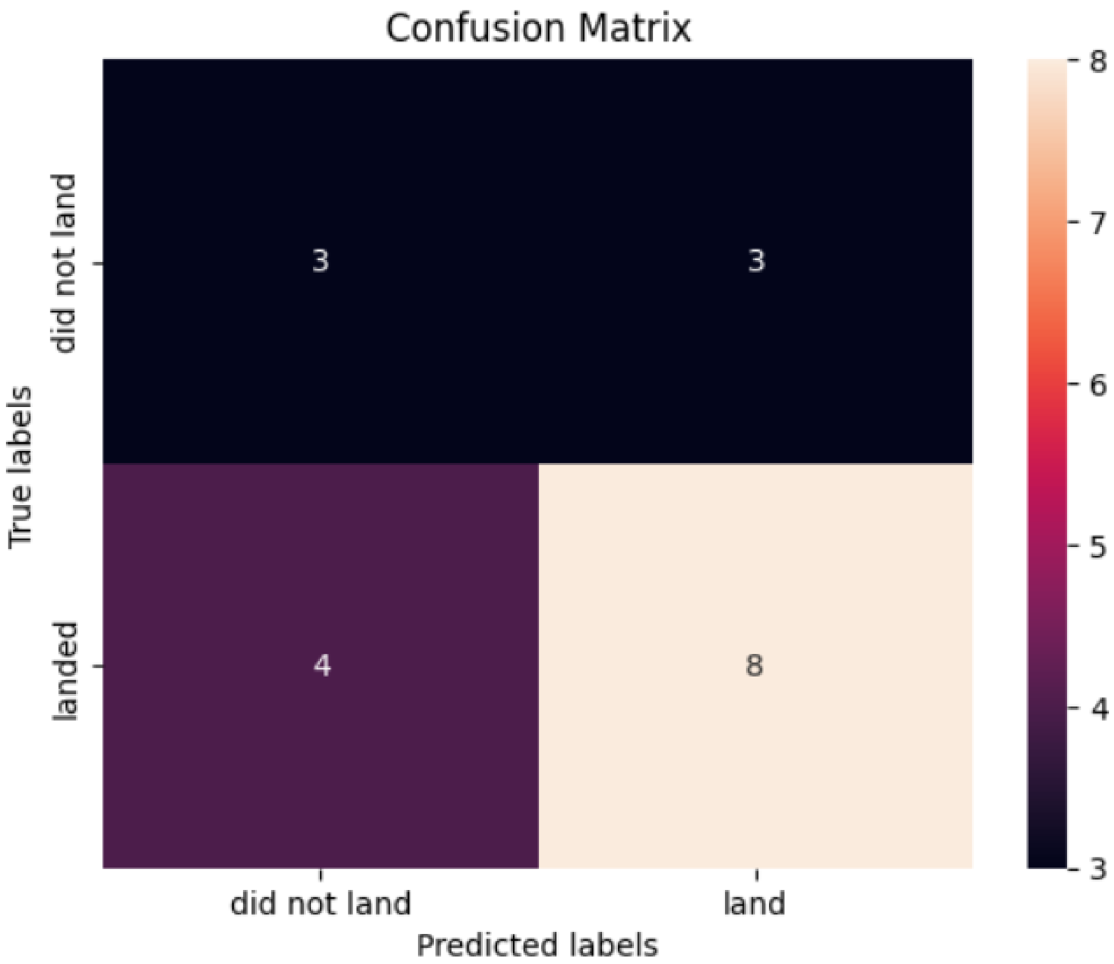


- We can see from the confusion matrix under this model
  - It correctly predicted that 3 would not land and that 12 would land
  - It incorrectly predicted that 3 would land
  - Again the main concern is the bottom left had no predictions at all.

```
gscv2 = GridSearchCV(svm,parameters2,scoring='accuracy',cv=10)
svm_cv = gscv2.fit(X_train,Y_train)

svm_cv.score(X_test, Y_test)

0.8333333333333334

svm_cv.fit(X_train,Y_train)
print("tuned hpyerparameters :(best parameters) ",svm_cv.best_params_)
print("accuracy :",svm_cv.best_score_)

tuned hpyerparameters :(best parameters)  {'C': 1.0, 'gamma': 0.03162277660168379, 'kernel': 'sigmoid'}
accuracy : 0.8482142857142856

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

MACHINE LEARNING - DECISION TREE CLASSIFIER

- Decision Tree is a Supervised Machine Learning Algorithm that uses a set of rules to make decisions, similarly to how humans make decisions



- We can see from the confusion matrix under this model
    - It correctly predicted that 3 would not land and that 8 would land
    - It incorrectly predicted that 3 would land that 4 would not land

```
parameters3 = {'criterion': ['gini', 'entropy'],
     'splitter': ['best', 'random'],
     'max_depth': [2*n for n in range(1,10)],
     'max_features': ['auto', 'sqrt'],
     'min_samples_leaf': [1, 2, 4],
     'min_samples_split': [2, 5, 10]}

tree = DecisionTreeClassifier()
```

```
gscv3 = GridSearchCV(tree,parameters3,scoring='accuracy',cv=10)
tree_cv = gscv3.fit(X_train,Y_train)
print("tuned hpyerparameters :(best parameters) ",tree_cv.best_params_)
print("accuracy :",tree_cv.best_score_)
```

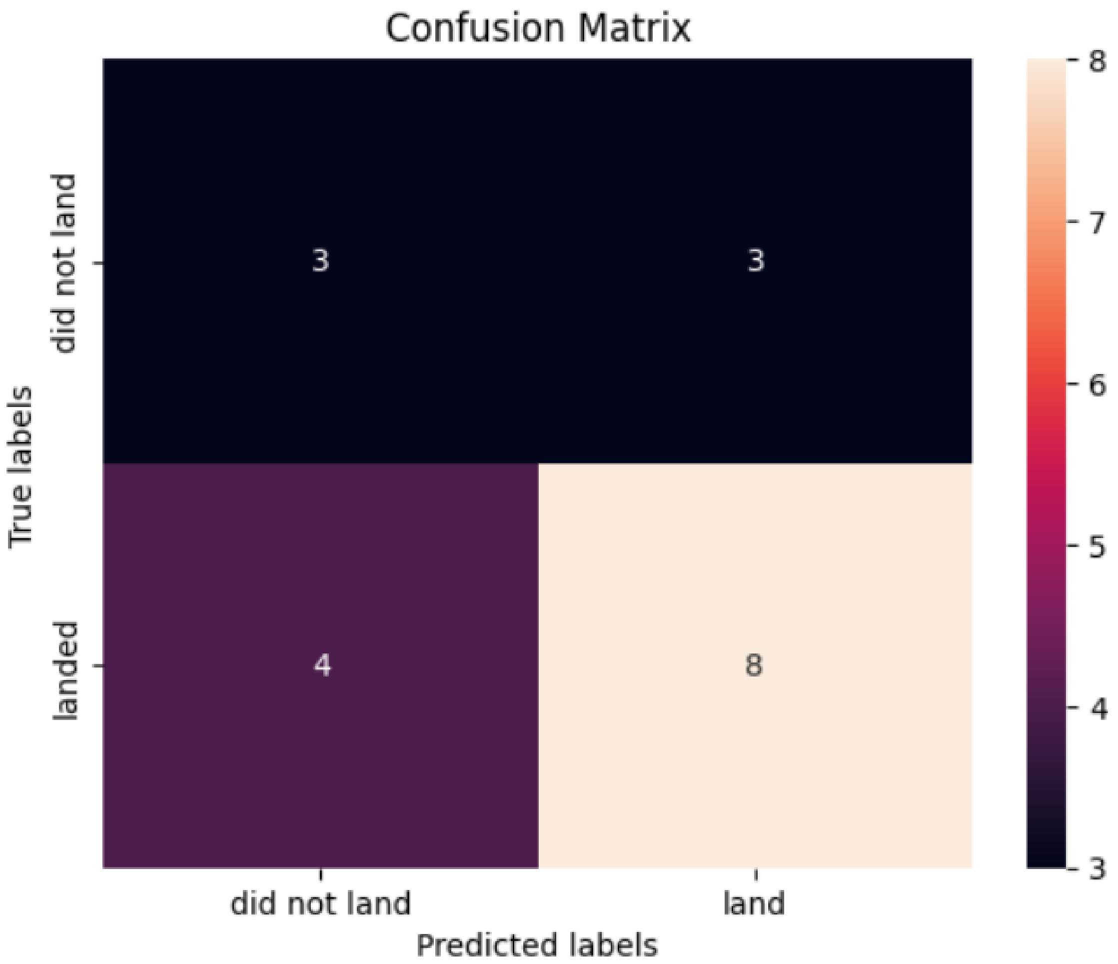
...

```
tree_cv.score(X_test, Y_test)
```

```
0.6111111111111112
```

MACHINE LEARNING - K NEAREST NEIGHBOR

- k-nearest neighbors (KNN) algorithm is a non-parametric, supervised learning classifier, which uses proximity to make classifications or predictions about the grouping of an individual data point



- We can see from the confusion matrix under this model
  - It correctly predicted that 3 would not land and that 8 would land
  - It incorrectly predicted that 3 would land that 4 would not land

```
parameters4 = {'n_neighbors': [1, 2, 3, 4, 5, 6, 7, 8, 9, 10],
              'algorithm': ['auto', 'ball_tree', 'kd_tree', 'brute'],
              'p': [1,2]}

KNN = KNeighborsClassifier()
```

```
gscv4= GridSearchCV(KNN,parameters4,scoring='accuracy',cv=10)
knn_cv = gscv4.fit(X_train,Y_train)
print("tuned hpyerparameters :(best parameters) ",knn_cv.best_params_)
print("accuracy :",knn_cv.best_score_)
```

```
tuned hpyerparameters :(best parameters)  {'algorithm': 'auto', 'n_neighbors': 10, 'p': 1}
accuracy : 0.8482142857142858
```

MACHINE LEARNING -

- When we look at the accuraccy and prediction scores of the models we can confidently advise proceeding with the KNearest Neighbors Classifier to appl y to the unknown data.

	Model	Accuracy	Prediction score
0	LogisticRegression(penalty={'C': [0.01, 0.1, 1...	0.8464285714285713	0.8333333333333334
1	SVC()	0.8482142857142856	0.8333333333333334
2	DecisionTreeClassifier()	0.8642857142857142	0.6111111111111112
3	KNeighborsClassifier()	0.8482142857142858	0.8333333333333334

DASHBOARD

<The GitHub link of the Cognos dashboard goes here.>

DASHBOARD TAB 1

Screenshot of dashboard tab 1 goes here

DASHBOARD TAB 2

Screenshot of dashboard tab 2 goes here

DASHBOARD TAB 3

Screenshot of dashboard tab 3 goes here

DISCUSSION

1. Data sources
2. Visualisations
3. Model selection
4. Additional data
5. Next steps

OVERALL FINDINGS & IMPLICATIONS

Findings

- It is possible to predict whether a launch will be successful
- SapceX's biggest client is NASA
- The most common orbit is GTO but in recent time this has shifted to VLEO and ISS orbits

Implications

- We need more data on the site locations and weather patterns related to the landings and launches
- It is possible to enter this market

CONCLUSION

- WE CAN PREDICT SUCCESS THOUGH MORE DATA WOULD ASSIST THE MODEL
- WE NEED TO SECURE SITES IN CLOSE PROXIMITY TO THE EQUATOR, THE COASTLINE, WHERE THERE IS RAILWAY AND ROAD ACCESS WITHIN 1KM
- Point 3
- Point 4

APPENDIX

- Falcon 9 - Wikipedia
- MODELLING DATA SET 1
- MODEELING DATA SET 2
- DATASET 1
- DATASET 2
- DATASET 3
- DATASET 4