IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Tom N
2021-10-09

Executive Summary

SpaceX data, acquired through their API, and by webscraping, was analysed using visualistaion and machine learning models to identify predictors for successful landings of the Falcon 9 core stage.

Result summary to be added.

Introduction

Launching to orbit is an expensive prospect. SpaceX have sought to gain a competitive advantage in the commercial space launch sector by reducing the cost of each launch. Their primary method to achieve this is by landing and re-using the booster stage of each launch vehicle.

The successful landing of a booster is not guaranteed, being influenced by several factors. Furthermore, for some launches, landing is not possible and the booster must be discarded. We seek to investigate what variables might affect attempting such a landing and the likely success of such an attempt.

Section 1

Methodology

Data Collection – SpaceX API

Using the requests library, calls were made to the SpaceX API at:
https://api.spacexdata.com/v4/launches

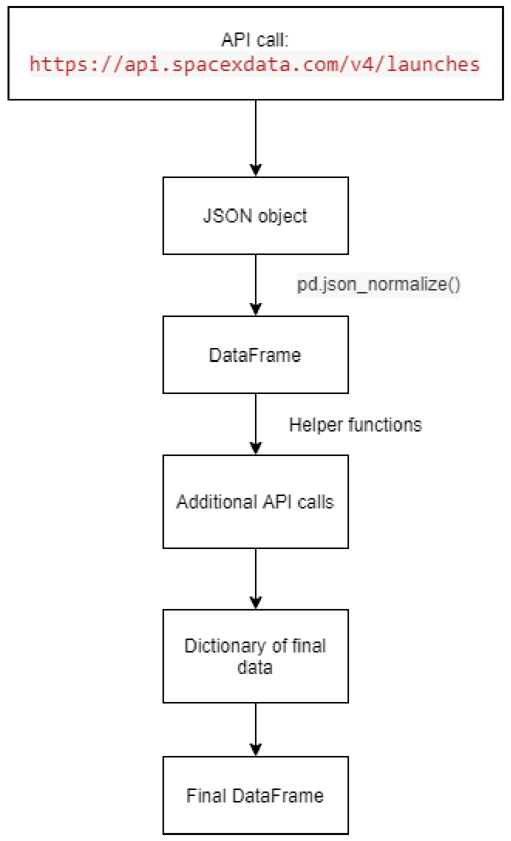
The returned JSON object was parsed into a Pandas DataFrame

Further details on data members were requested from other API endpoints:

- Rockets
- Launchpads
- Payloads
- Cores

These details were appended into the DataFrame.

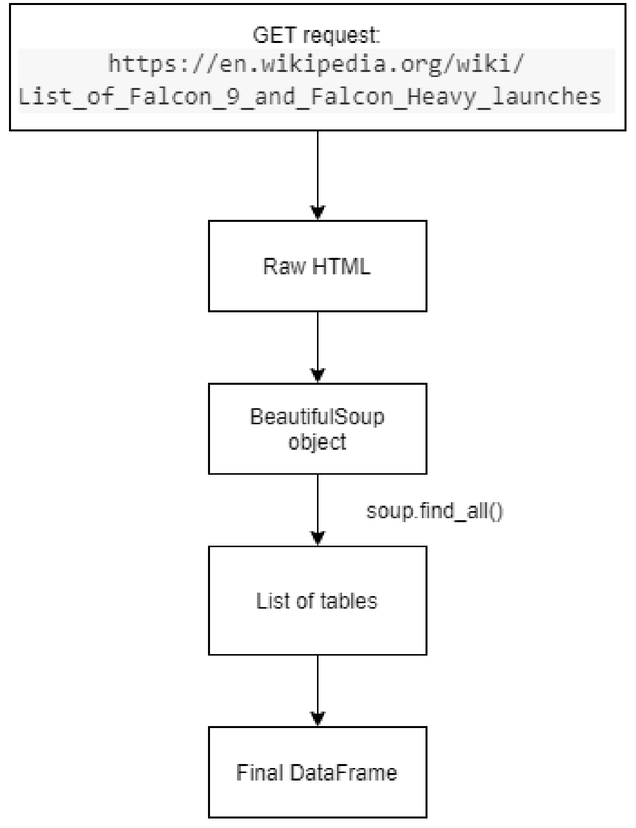
Notebook:
https://github.com/tomlnagel/ibm_capstone/blob/master/1.1%20Data%20collection.ipynb



Data Collection - Scraping

Wikipedia page https://en.wikipedia.org/wiki/List_of_Falcon_9_and_Falcon_Heavy_launches was scrpaed for data on past launches. The HTML output from the requests library was parsed by a BeutifulSoup object to exract required data from the relevant HTML tables, and loaded to a Pandas DataFrame.

Notebook: https://github.com/tomlnagel/ibm_capstone/blob/master/1.2%20Webscraping.ipynb



Data Wrangling

Several key statistics from the extracted data were computed:

- Missing value counts
- Data types
- Launches from each site
- Launches to each orbit
- Landing outcomes
  - Success or failure
  - Intended landing type

A Boolean 0/1 value (‘Class’) was computed and added to the DataFrame representing the success or failure of the landing attempt.

Notebook: https://github.com/tomlnagel/ibm_capstone/blob/master/1.3%20Data%20wrangling.ipynb

EDA with Data Visualization

The following scatter plots were generated, in each case showing success of landing as a third dimension – color.

- Flight number vs. payload mass
- Flight number vs. launch site
- Payload mass vs. launch site
- Flight number vs. orbit
- Payload mass vs. orbit

The following bar chart was generated:

- Orbit vs. landing success rate

The following line plot was generated:

- Year vs. landing success rate

All plots were generated using the Seaborn library.

Notebook:
https://github.com/tomlnagel/ibm_capstone/blob/master/2.2%20EDA%20with%20data%20visualisation.ipynb

EDA with SQL

The following SQL queries were executed:

- Find unique launch sites
- Find launches from Cape Canaveral (CCA*)
- Find total payload mass launched for NASA
- Find average payload mass for Falcon 9 booster v1.1
- Find first successful ground landing date
- Find boosters with successful landings on the drone ship having launched mid-size payloads
- Find counts of each landing outcome type
- Find booster versions that have launched with maximum payload mass
- Find failed landings in 2015
- Find and rank counts of landing outcomes 2010-2017

Notebook:
https://github.com/tomlnagel/ibm_capstone/blob/master/2.1%20Exploratory%20data%20analysis.ipynb

Build an Interactive Map with Folium

A map was created and the following markers added:

- Each launch site:
  - Cape Canaveral Launch Complex 40
  - Cape Canaveral Space Launch Complex 40
  - Kenedy Space Centre Launch Complex 39A
  - Vandeberg Airfoce Base Space Launch Complex 4E
- Cluster markers for each launch, color coded by landing outcome
- Lines and distance markers from Launch Complex 40 to nearest:
  - Coastline
  - Highway
  - Railway
  - City

Notebook:
https://github.com/tomlnagel/ibm_capstone/blob/master/3.1%20Visual%20analytics.ipynb

Build a Dashboard with Plotly Dash

An interactive dashboard was created using Plotly and Dash. The dashboard shows, for each launch site or all sites combined, specified by a dropdown:

- A pie chart of landing success rate

Additionally a scatter chart shows, for each launch site or all sites combined, and within a payload mass range specified by a slider:

- A scatter chart of payload mass against landing success
  - Marker color shows a third dimension: booster version category

Python code:
https://github.com/tomlnagel/ibm_capstone/blob/master/dashboard/spacex_dash_app.py

Predictive Analysis (Classification)

Several models were build to attempt to predict successful landing outcome. There were each built using the scikit-learn library, and in each case the optimum hyperparameters were computed using a cross-validation grid search of 10 folds.

The models were trained and tested on an 80:20 split of the source data, and were scored using the model default scoring method. A confusion matrix was generated for each model.

Models:

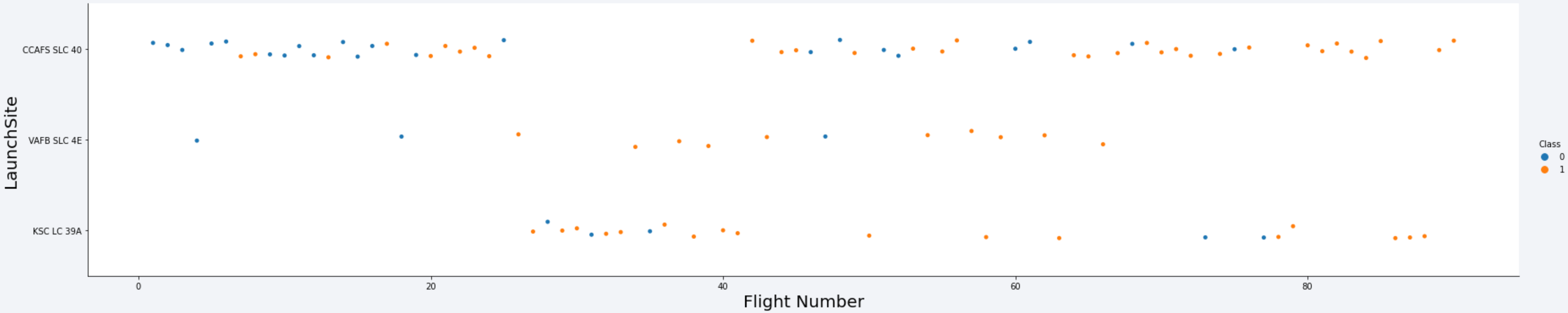
- Logistic regression
- Support vector machine
- Decision tree
- K-nearest neighbours

Notebook:
https://github.com/tomlnagel/ibm_capstone/blob/master/4.1%20Modeling.ipynb

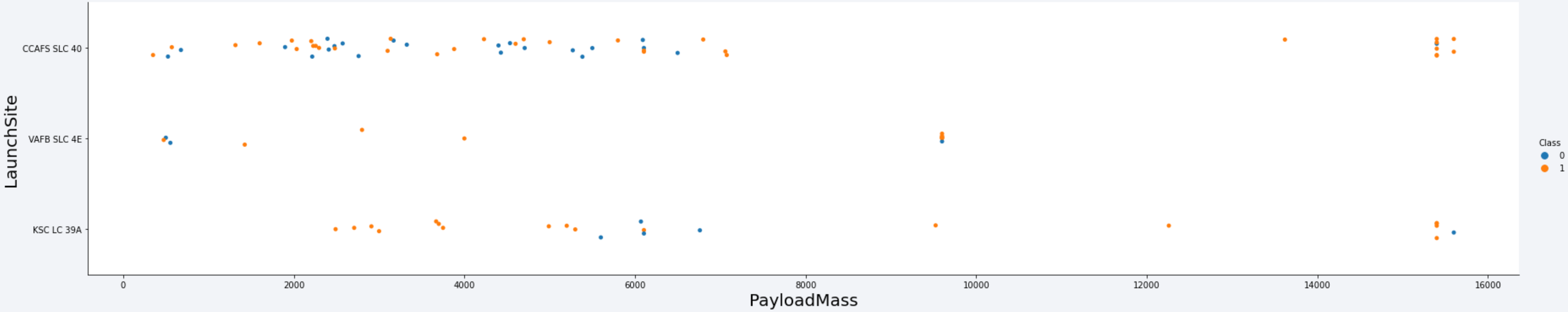
Section 2

Insights drawn from EDA

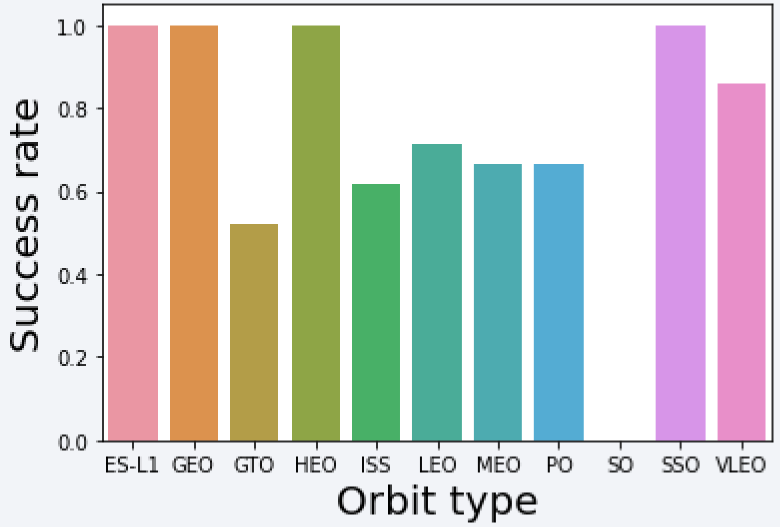
Flight Number vs. Launch Site



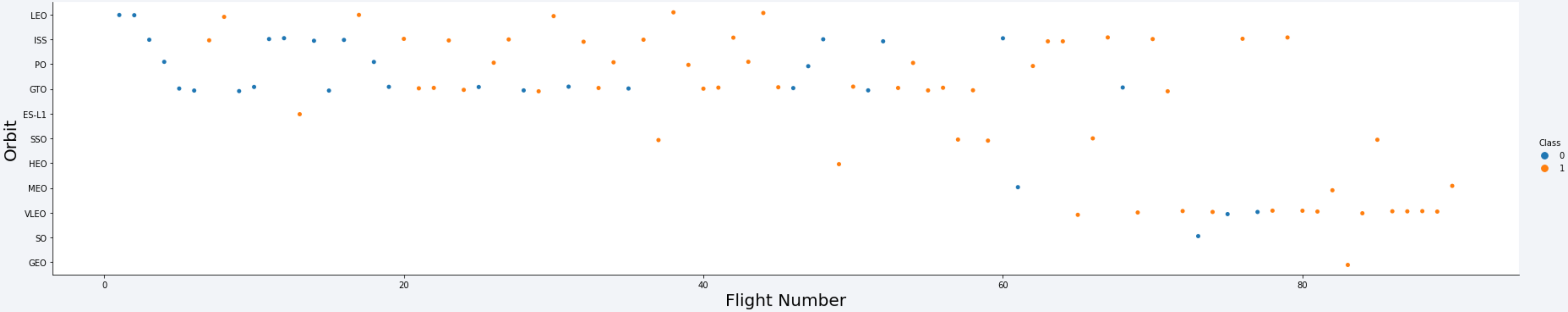
Payload vs. Launch Site



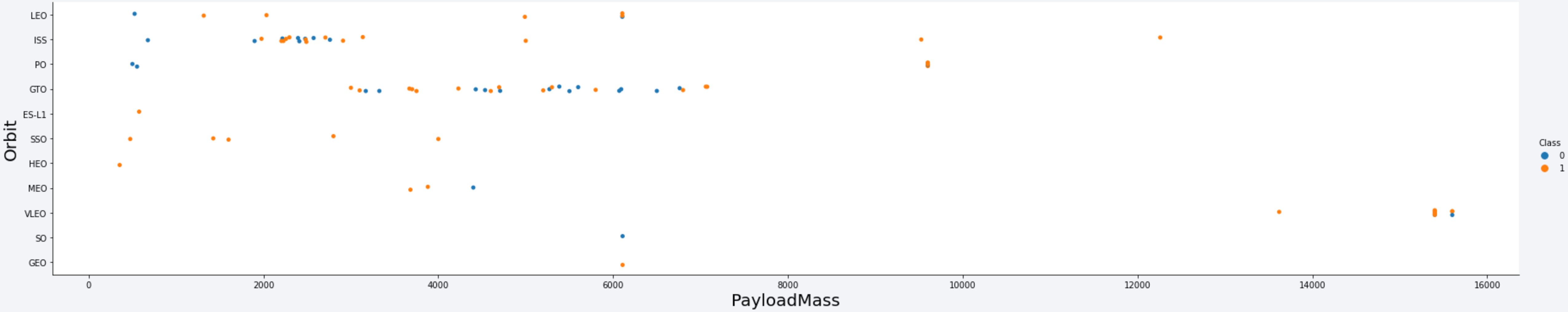
Success Rate vs. Orbit Type



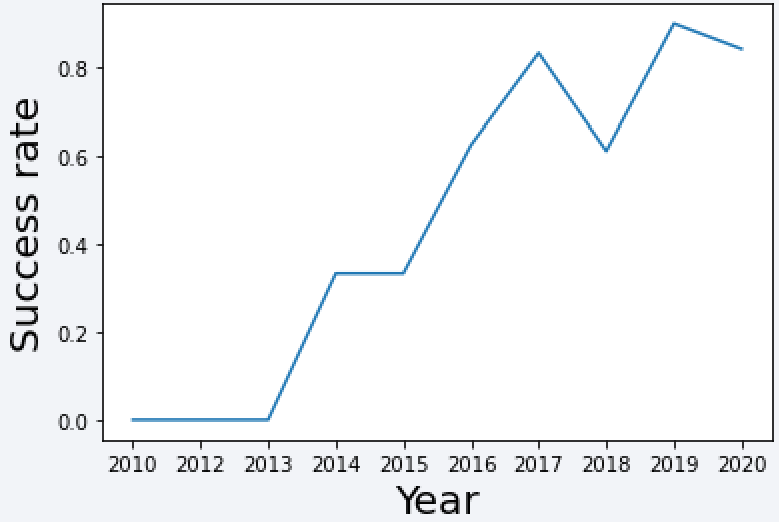
Flight Number vs. Orbit Type



Payload vs. Orbit Type



Launch Success Yearly Trend



All Launch Site Names

Display the names of the unique launch sites in the space mission

In [8]:
```
%%sql
select distinct launch_site from spacexdataset
```

```
 * ibm_db_sa://dzw73829:***@764264db-9824-4b7c-82df-40d1b13897c2.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:32536/bludb
Done.
```

Out[8]:

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

Launch Site Names Begin with 'CCA'

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

In [10]:
```
%%sql
select * from spacexdataset
where launch_site like 'CCA%'
limit 5
```

```
 * ibm_db_sa://dzw73829:***@764264db-9824-4b7c-82df-40d1b13897c2.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:32536/bludb
Done.
```

Out[10]:

DATE	time__utc_	booster_version	launch_site	payload	payload_mass__kg_	orbit	customer	mission_outcome	landing__outcome
2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
2010-12-08	15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight C1, two CubeSats, barrel of Brouere cheese	0	LEO (ISS)	NASA (COTS) NRO	Success	Failure (parachute)
2012-05-22	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
2012-10-08	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
2013-03-01	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

Total Payload Mass

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

In [13]:
```
%%sql
select sum(payload_mass__kg_) from spacexdataset
where customer = 'NASA (CRS)'
```

```
 * ibm_db_sa://dzw73829:***@764264db-9824-4b7c-82df-40d1b13897c2.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:32536/bludb
Done.
```

Out[13]:

1
45596

Average Payload Mass by F9 v1.1

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

In [16]:

```
%%sql
select avg(payload_mass__kg_) from spacexdataset
where booster_version like 'F9 v1.1%'
```

```
 * ibm_db_sa://dzw73829:***@764264db-9824-4b7c-82df-40d1b13897c2.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:32536/bludb
Done.
```

Out[16]:

1
2534

First Successful Ground Landing Date

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

Hint:Use min function

In [21]:
```
%%sql
select min(date) from spacexdataset
where landing__outcome = 'Success (ground pad)'
```

```
 * ibm_db_sa://dzw73829:***@764264db-9824-4b7c-82df-40d1b13897c2.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:32536/bludb
Done.
```

Out[21]:

1
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

In [31]:
```
%%sql
select booster_version from spacexdataset
where landing__outcome like 'Success%drone%'
and payload_mass__kg_ between 4000 and 6000
```

```
 * ibm_db_sa://dzw73829:***@764264db-9824-4b7c-82df-40d1b13897c2.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:32536/bludb
Done.
```

Out[31]:

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

Total Number of Successful and Failure Mission Outcomes

List the total number of successful and failure mission outcomes

In [33]:
```
%%sql
select landing__outcome, count(*) from spacexdataset
group by landing__outcome
```

```
 * ibm_db_sa://dzw73829:***@764264db-9824-4b7c-82df-40d1b13897c2.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:32536/bludb
Done.
```

Out[33]:

landing__outcome	2
Controlled (ocean)	5
Failure	3
Failure (drone ship)	5
Failure (parachute)	2
No attempt	22
Precluded (drone ship)	1
Success	38
Success (drone ship)	14
Success (ground pad)	9
Uncontrolled (ocean)	2

Boosters Carried Maximum Payload

List the names of the booster_versions which have carried the maximum payload mass. Use a subquery

In [38]:
```
%%sql
select booster_version from spacexdataset
where payload_mass__kg_ = (select max(payload_mass__kg_) from spacexdataset)
```

```
 * ibm_db_sa://dzw73829:***@764264db-9824-4b7c-82df-40d1b13897c2.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:32536/bludb
Done.
```

Out[38]:

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

2015 Launch Records

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

In [42]:

```
%%sql
select * from spacexdataset
where landing__outcome like 'Fail%drone%'
and year(date) = '2015'
```

```
 * ibm_db_sa://dzw73829:***@764264db-9824-4b7c-82df-40d1b13897c2.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:32536/bludb
Done.
```

Out[42]:

DATE	time__utc_	booster_version	launch_site	payload	payload_mass__kg_	orbit	customer	mission_outcome	landing__outcome
2015-01-10	09:47:00	F9 v1.1 B1012	CCAFS LC-40	SpaceX CRS-5	2395	LEO (ISS)	NASA (CRS)	Success	Failure (drone ship)
2015-04-14	20:10:00	F9 v1.1 B1015	CCAFS LC-40	SpaceX CRS-6	1898	LEO (ISS)	NASA (CRS)	Success	Failure (drone ship)

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

In [48]:
```
%%sql
select landing__outcome, count(*) from spacexdataset
where date between '2010-06-04' and '2017-03-20'
group by landing__outcome
order by 2 desc
```

```
 * ibm_db_sa://dzw73829:***@764264db-9824-4b7c-82df-40d1b13897c2.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:32536/bludb
Done.
```

Out[48]:

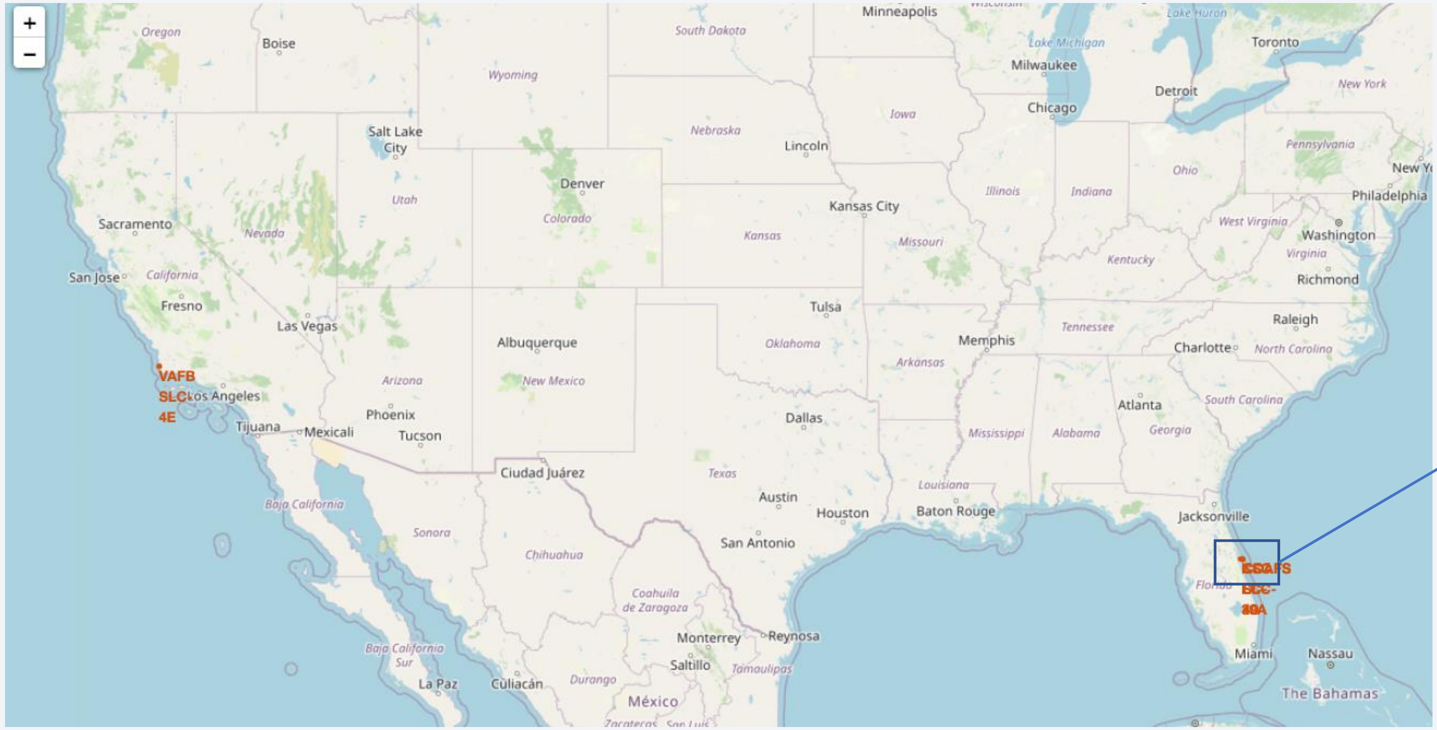
landing__outcome	2
No attempt	10
Failure (drone ship)	5
Success (drone ship)	5
Controlled (ocean)	3
Success (ground pad)	3
Failure (parachute)	2
Uncontrolled (ocean)	2
Precluded (drone ship)	1

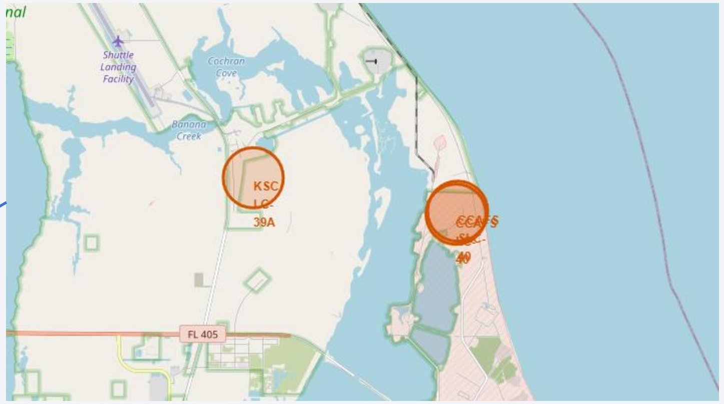
Section 4

Launch Sites Proximities Analysis

Launch site map

The four launch sites were plotted on a map. Three are in close proximity, with two being different labels for the same site.



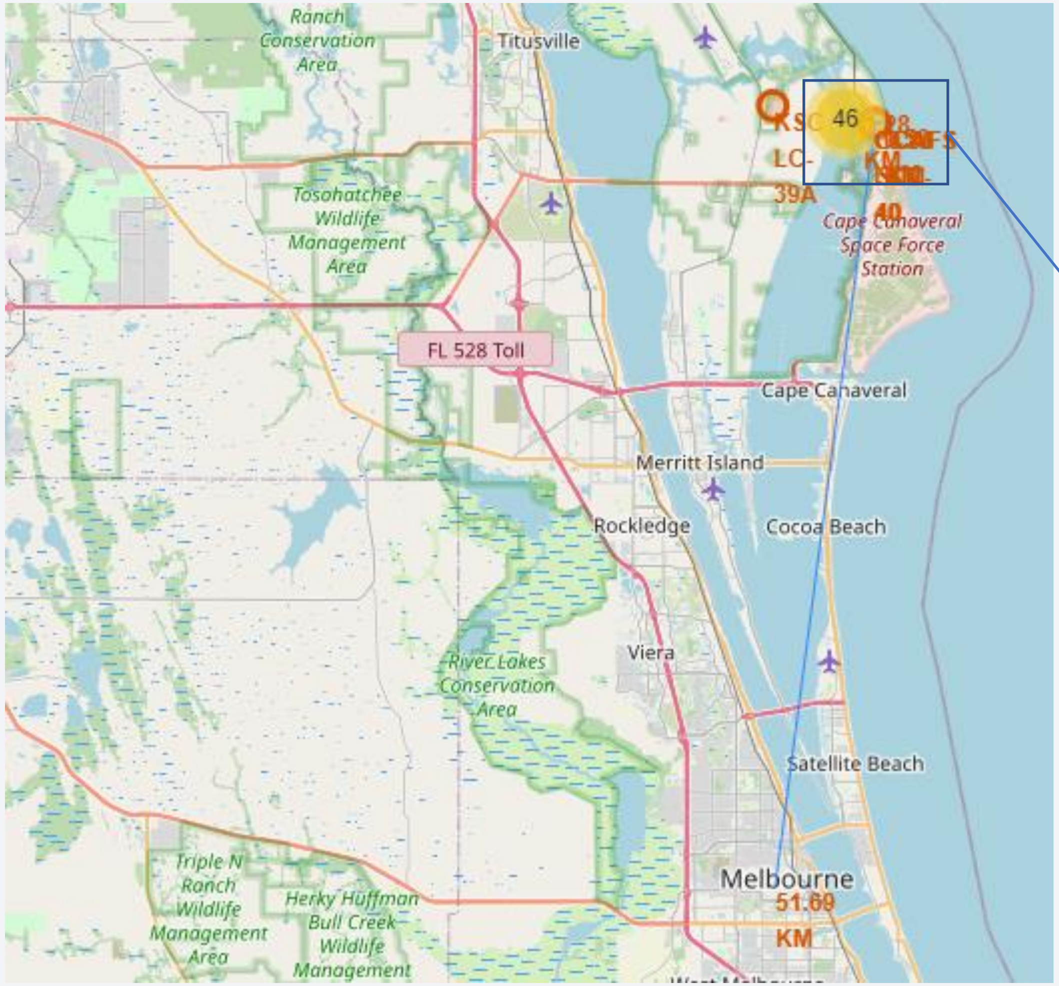


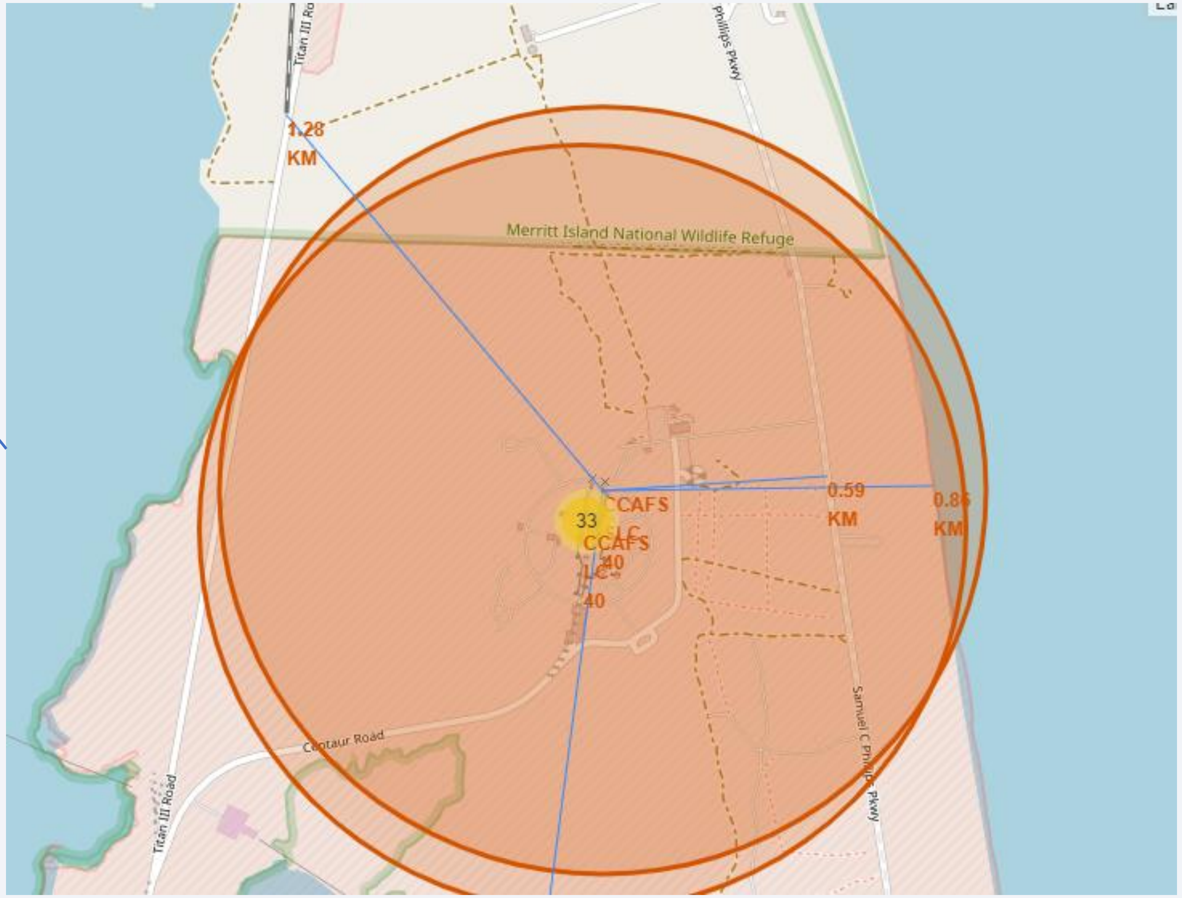
Landing success indicators

Each launch was plotted in a marker cluster. Red markers indicate failed landings, green markers indicate successes.

Distances to features

Distances were calculated from one launch site to nearby features: coast, highway, railway, and a city.



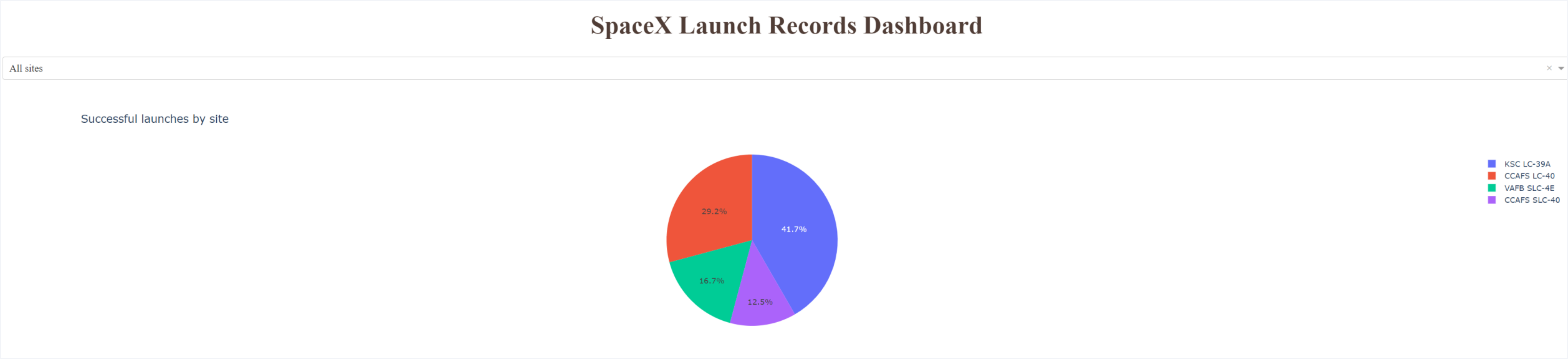


Section 5

Build a Dashboard with Plotly Dash

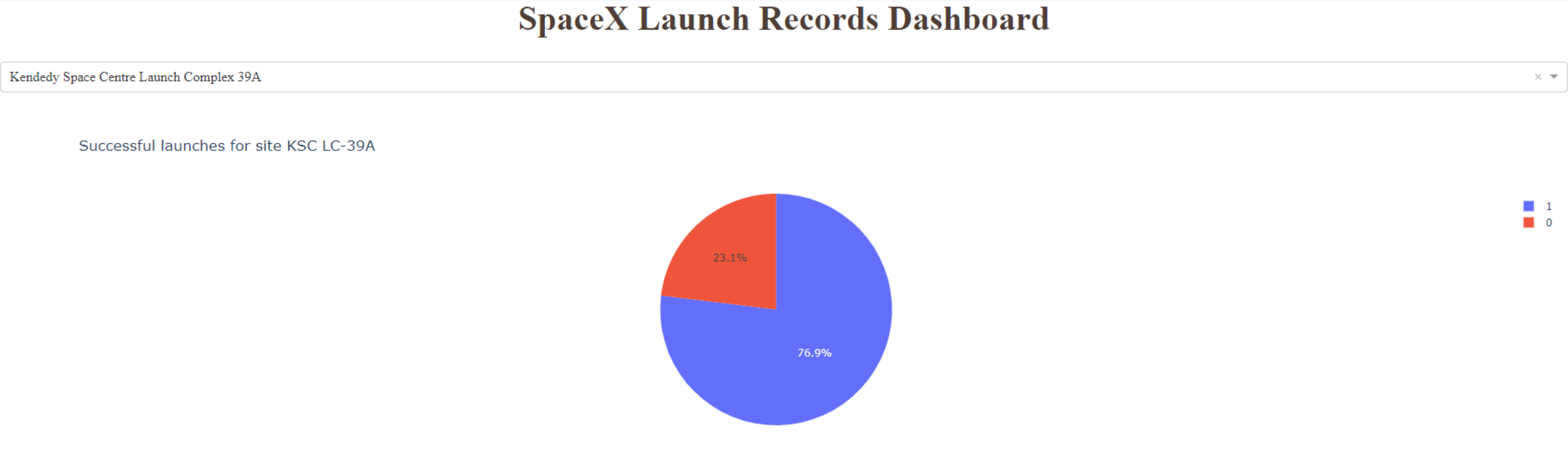
Dashboard – launch successes

The pie chart for ‘All sites’ displays the proportion of successful landings launched from each site.



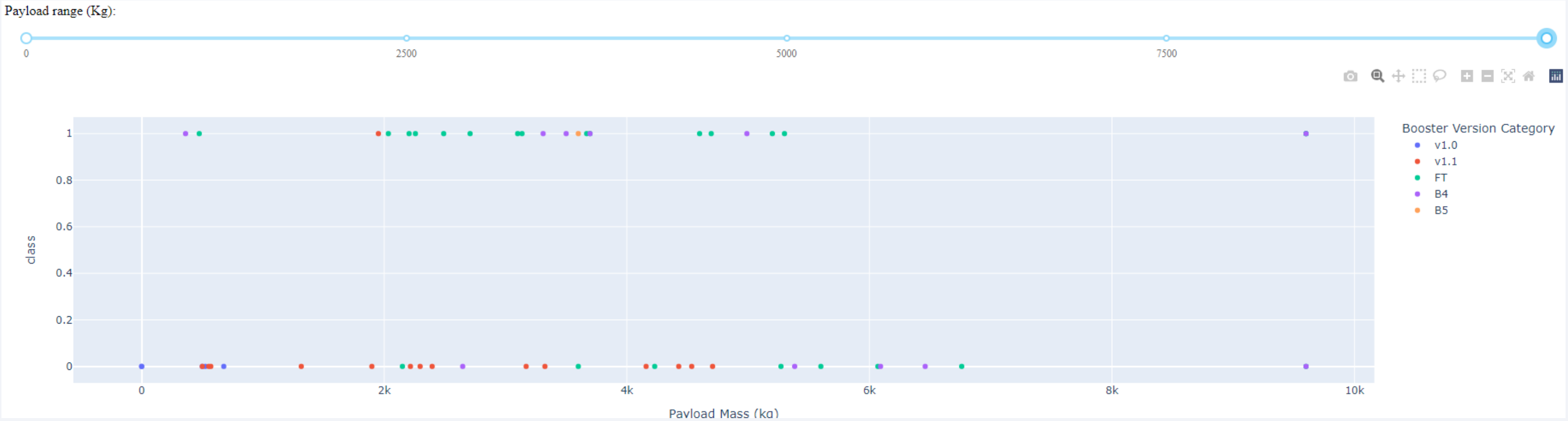
Dachboard – KSC success rate

The pie chart for Kennedy Space Centre shows the high rate of successful landings launched from that site:

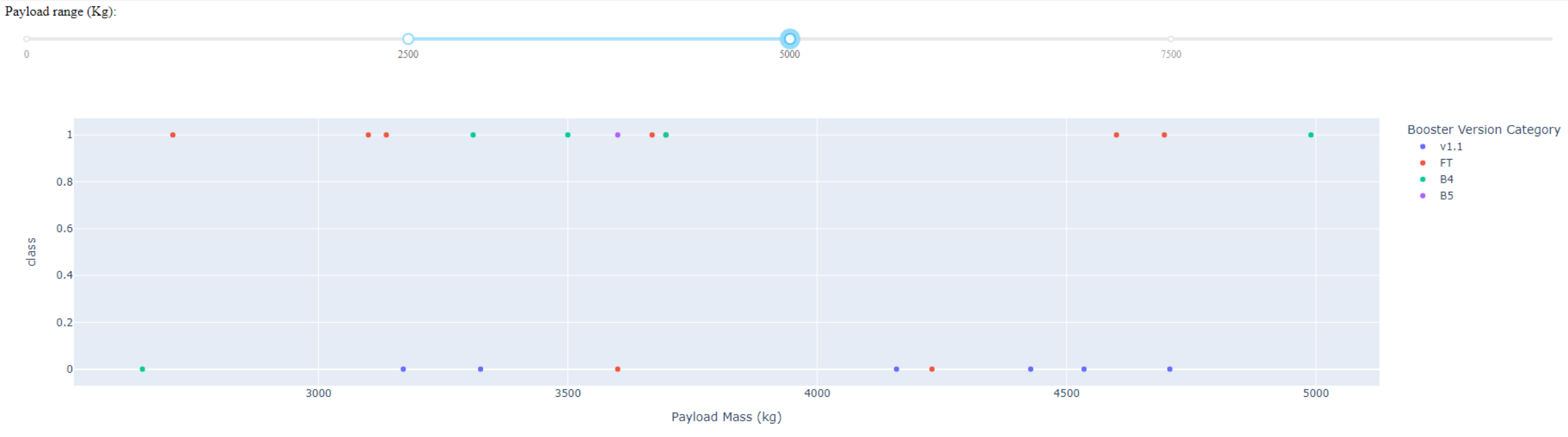


Dashboard – successes by payload mass

The scatter chart shows the landing successes and failures for each launch by payload mass:



Adjusting the range slider shows only launches in that payload range:



Section 6

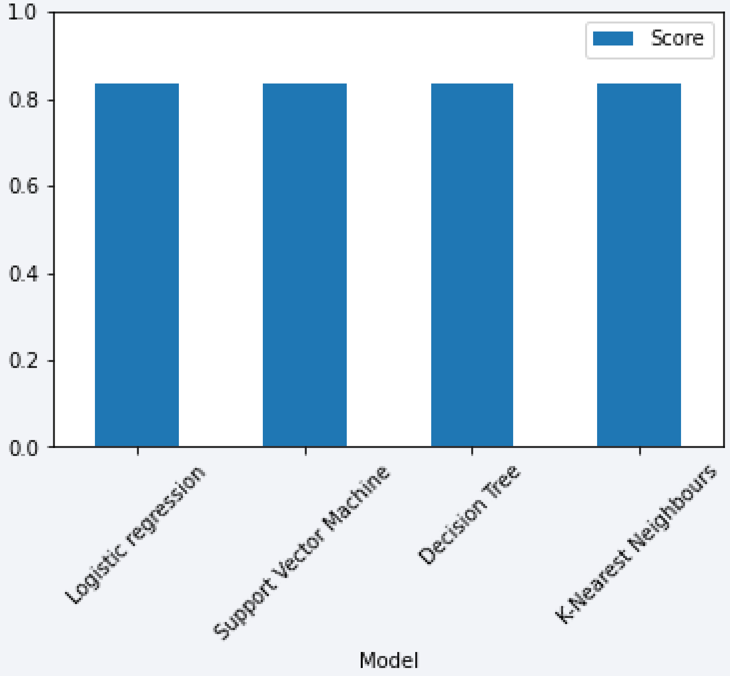
Predictive Analysis (Classification)

Classification Accuracy

Each model was trained, evaluated for best hyperparameters using 10-fold cross validation, and tested against a pre-split test set. The models were scored using the default method for each within scikit-learn.

The scores were:

- Logistic regression: 83.3%
- Decision tree: 83.3%
- Support vector machine: 83.3%
- K-nearest neighbors: 83.3%



As can be seen, each model performs equally well for this data set.

Confusion Matrix

In addition to the overall score, a confusion matrix was generated for each model. As this the scoring, all models produced the same confusion matrix.

This shows that the only errors in the models were type I – false positives. Each model incorrectly predicted three of the test set would successfully land, when they did not do so.

Conclusions

Any model seems to work fine for predicting whether a launch will successfully land.

There were no interesting findings from any of the other assignments – they were box-checking exercises.

Thank you!