IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Diego Fernando Rayo Cabrera.

28/12/2022

Outline

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

Executive Summary

- Summary of methodologies
- - Information assortment
- - Information fighting
- - Exploratory Information Examination with Information Representation
- - Exploratory Information Examination with SQL
- - Building an intelligent guide with Folium
- - Building a Dashboard with Plotly Run
- - Prescient examination (Arrangement)
- Summary of all results
- - Exploratory Information Examination results
- - Intelligent examination demo in screen captures
- - Prescient examination results

Introduction

- Project background and context
- SpaceX is the best organization of the business space
- age, making space travel reasonable. The organization promotes Bird of prey
- 9 rocket dispatches on its site, with an expense of 62 million bucks;
- different suppliers cost vertical of 165 million bucks every, a large part of the
- investment funds is on the grounds that SpaceX can reuse the main stage. Consequently, on the off chance that we
- can decide whether the main stage will land, we can decide the expense
- of a send off. In view of public data and AI
- models, we will anticipate on the off chance that SpaceX will reuse the primary stage.
- Problems you want to find answers
- - How do factors, for example, payload mass, send off site, number of
- flights, and circles influence the progress of the principal stage landing?
- - Does the pace of fruitful arrivals increment throughout the long term?
- - What is the best calculation that can be utilized for double grouping
- for this situation?

Section 1

Methodology

Methodology

- Executive Summary
- Data collection methodology:
  - - Utilizing SpaceX Rest Programming interface
  - - Utilizing Web Rejecting from Wikipedia
  - Perform data wrangling
  - - Sifting the information
  - - Managing missing qualities
  - - Utilizing One Hot Encoding to set up the information to a paired grouping
  - Perform exploratory data analysis (EDA) using visualization and SQL
- Perform interactive visual analytics using Folium and Plotly Dash
- Perform predictive analysis using classification models
  - - Building, tuning and assessment of characterization models to guarantee the best results

Data Collection

- Information assortment process included a blend of Programming interface demands from SpaceX REST
- Programming interface and Web Scratching information from a table in SpaceX's Wikipedia passage.
- We needed to utilize both of these information assortment techniques to get total
- data about the send-offs for a more definite examination.
- Information Segments are acquired by utilizing SpaceX REST Programming interface:
- FlightNumber, Date, BoosterVersion, PayloadMass, Circle, LaunchSite,
- Result, Flights, GridFins, Reused, Legs, LandingPad, Block, ReusedCount,
- Sequential, Longitude, Scope
- Information Segments are acquired by utilizing Wikipedia Web Scratching:
- Flight No., Send off site, Payload, PayloadMass, Circle, Client, Send off
- result, Rendition Promoter, Sponsor landing, Date, Time

Data Collection – SpaceX API

Requesting rocket launch data from SpaceX API

Decoding the response content using .json() and turning it into a dataframe using .json_normalize()

Requesting needed information about the launches from SpaceX API by applying custom functions

Constructing data we have obtained into a dictionary

Creating a dataframe from the dictionary

Filtering the dataframe to only include Falcon 9 launches

Replacing missing values of Payload Mass column with calculated .mean() for this column

Exporting the data to CSV

https://github.com/evgenyzorin/IBM-Applied-Data-Science-Capstone/blob/46bef070c13a4ddb100f78576d1e74cc0fb61c00/Data%20Collection%20API.ipynb

Data Collection - Scraping

Requesting	Creating	Extracting	Collecting	Constructing	Creating	Exporting
• Requesting Falcon 9 launch data from Wikipedia	• Creating a BeautifulSoup object from the HTML response	• Extracting all column names from the HTML table header	• Collecting the data by parsing HTML tables	• Constructing data we have obtained into a dictionary	• Creating a dataframe from the dictionary	• Exporting the data to CSV

https://github.com/evgenyzorin/IBM-Applied-Data-Science-Capstone/blob/46bef070c13a4ddb100f78576d1e74cc0fb61c00/Data%20Collection%20with%20Web%20Scraping.ipynb

Data Wrangling

In the informational collection, there are a few distinct situations where the promoter didn't land effectively. At times an arrival was endeavored yet bombed because of a mishap; for instance, Valid Sea implies the mission result was effectively landed to a particular district of the sea while Misleading Sea implies the mission result was ineffectively arrived to a particular locale of the sea. Genuine RTLS implies the mission result was effectively arrived to a ground cushion Misleading RTLS implies the mission result was fruitlessly arrived to a ground pad. True ASDS implies the mission result was effectively arrived on a robot transport Misleading ASDS implies the mission result was fruitlessly arrived on a robot transport.

We basically convert those results into Preparing Marks with "1" signifies the promoter effectively landed, "0" signifies it was fruitless.

- Perform exploratory Data Analysis and determine Training Labels
- Calculate the number of launches on each site
- Calculate the number and occurrence of each orbit
- Calculate the number and occurrence of mission outcome per orbit type
- Create a landing outcome label from Outcome column
- Exporting the data to CSV

https://github.com/evgenyzorin/IBM-Applied-Data-Science-Capstone/blob/46bef070c13a4ddb100f78576d1e74cc0fb61c00/Data%20Wrangling.ipynb

EDA with Data Visualization

- Diagrams were plotted:
- Flight Number versus Payload Mass, Flight Number versus Send off Site, Payload Mass
- versus Send off Site, Circle Type versus Achievement Rate, Flight Number versus Circle Type,
- Payload Mass versus Circle Type and Achievement Rate Yearly Pattern
- Dissipate plots show the connection between factors. In the event that a relationship exists,
- they could be utilized in AI model.
- Bar diagrams show examinations among discrete classifications. The objective is to show the
- connection between the particular classifications being looked at and a deliberate
- esteem.
- Line diagrams show patterns in information over the long haul (time series).

https://github.com/evgenyzorin/IBM-Applied-Data-Science-Capstone/blob/46bef070c13a4ddb100f78576d1e74cc0fb61c00/EDA%20with%20Data%20Visualization.ipynb

EDA with SQL

- Performed SQL questions:
- • Showing the names of the extraordinary send off destinations in the space mission
- • Showing 5 records where send off locales start with the string 'CCA'
- • Showing the absolute payload mass conveyed by supporters sent off by NASA (CRS)
- • Showing normal payload mass conveyed by sponsor variant F9 v1.1
- • Posting the date when the principal effective landing result in ground cushion was accomplished
- • Posting the names of the promoters which have progress in drone transport and have payload mass more prominent than 4000 yet
- under 6000
- • Posting the absolute number of effective and disappointment mission results
- • Posting the names of the promoter renditions which have conveyed the greatest payload mass
- • Posting the bombed landing results in drone transport, their supporter variants and send off site names for the months in
- year 2015
- • Positioning the count of landing results (like Disappointment (drone boat) or Achievement (ground cushion)) between the date
- 2010-06-04 and 2017-03-20 in dropping request

https://github.com/evgenyzorin/IBM-Applied-Data-Science-Capstone/blob/46bef070c13a4ddb100f78576d1e74cc0fb61c00/EDA%20with%20SQL.ipynb

Build an Interactive Map with Folium

- Markers of all Send off Destinations:
- - Added Marker with Circle, Popup Name and Text Name of NASA Johnson Space Center utilizing
- its scope and longitude facilitates as a beginning area.
- - Added Markers with Circle, Popup Name and Text Name of all Send off Destinations utilizing their scope
- what's more, longitude directions to show their geological areas and vicinity to Equator and
- coasts.
- Shaded Markers of the send off results for each Send off Site:
- - Added shaded Markers of achievement (Green) and fizzled (Red) dispatches utilizing Marker Group to
- recognize which send off locales have moderately high achievement rates.
- Distances between a Send off Site to its vicinities:
- - Added shaded Lines to show distances between the Send off Site KSC LC-39A (as an
- model) and its vicinities like Rail line, Parkway, Shoreline and Nearest City.

https://github.com/evgenyzorin/IBM-Applied-Data-Science-Capstone/blob/46bef070c13a4ddb100f78576d1e74cc0fb61c00/Interactive%20Visual%20Analytics%20with%20Folium.ipynb

Build a Dashboard with Plotly Dash

- Send off Destinations Dropdown Rundown:
- - Added a dropdown rundown to empower Send off Site choice.
- Pie Outline showing Achievement Dispatches (All Locales/Certain Site):
- - Added a pie graph to show the all out effective send-offs count for all destinations and the
- Achievement versus Bombed counts for the site, on the off chance that a particular Send off Site was chosen.
- Slider of Payload Mass Reach:
- - Added a slider to choose Payload range.
- Disperse Diagram of Payload Mass versus Achievement Rate for the different Promoter Variants:
- - Added a dissipate diagram to show the relationship among's Payload and Send off Progress.

https://github.com/evgenyzorin/IBM-Applied-Data-Science-Capstone/blob/46bef070c13a4ddb100f78576d1e74cc0fb61c00/spacex_dash_app.py

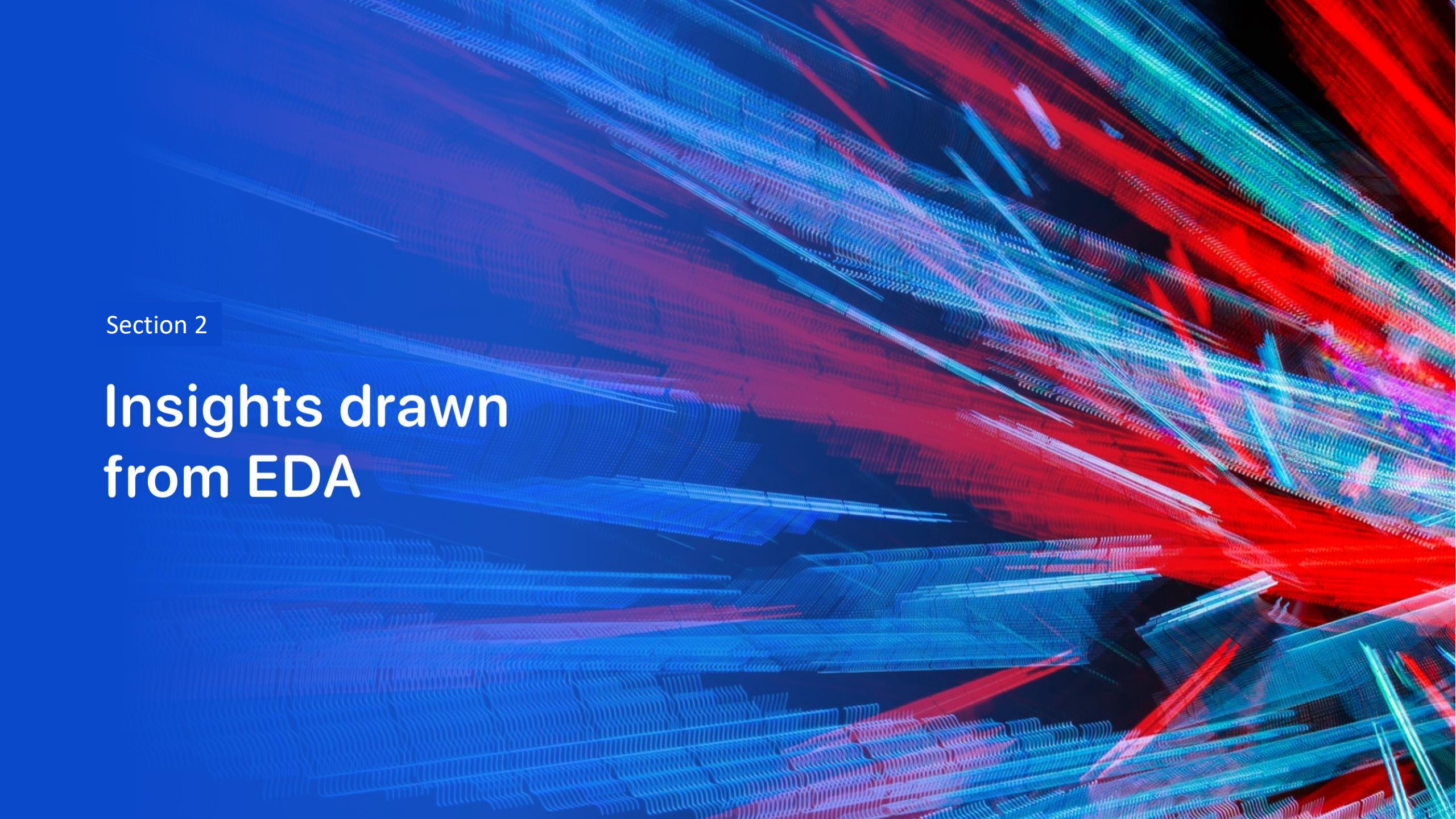
Predictive Analysis (Classification)

1	2	3	4	5	6	7	8
Creating a NumPy array from the column “Class” in data	Standardizing the data with StandardScaler, then fitting and transforming it	Splitting the data into training and testing sets with train_test_split function	Creating a GridSearchCV object with cv = 10 to find the best parameters	Applying GridSearchCV on LogReg, SVM, Decision Tree, and KNN models	Calculating the accuracy on the test data using the method .score() for all models	Examining the confusion matrix for all models	Finding the method performs best by examining the Jaccard_score and F1_score metrics

https://github.com/evgenyzorin/IBM-Applied-Data-Science-Capstone/blob/46bef070c13a4ddb100f78576d1e74cc0fb61c00/Machine%20Learning%20Prediction.ipynb

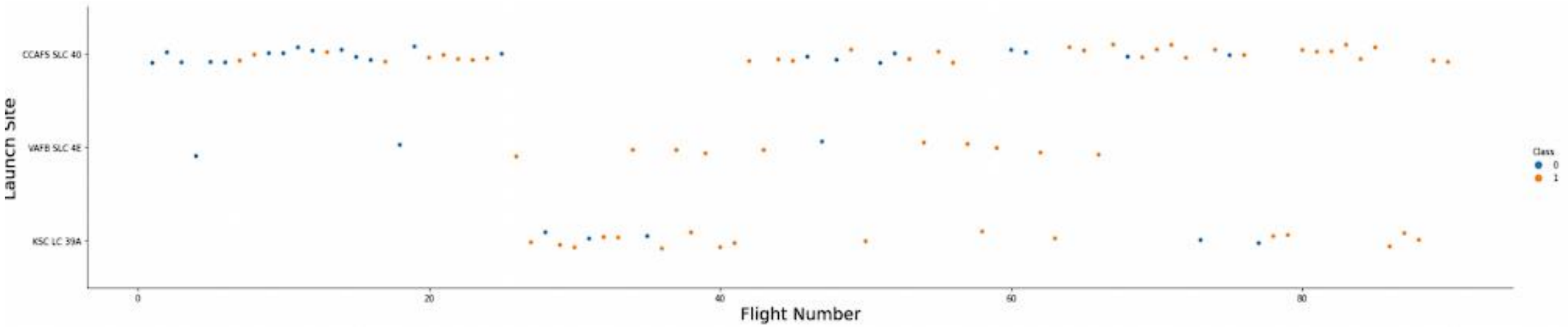
Results

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


Section 2
Insights drawn from EDA

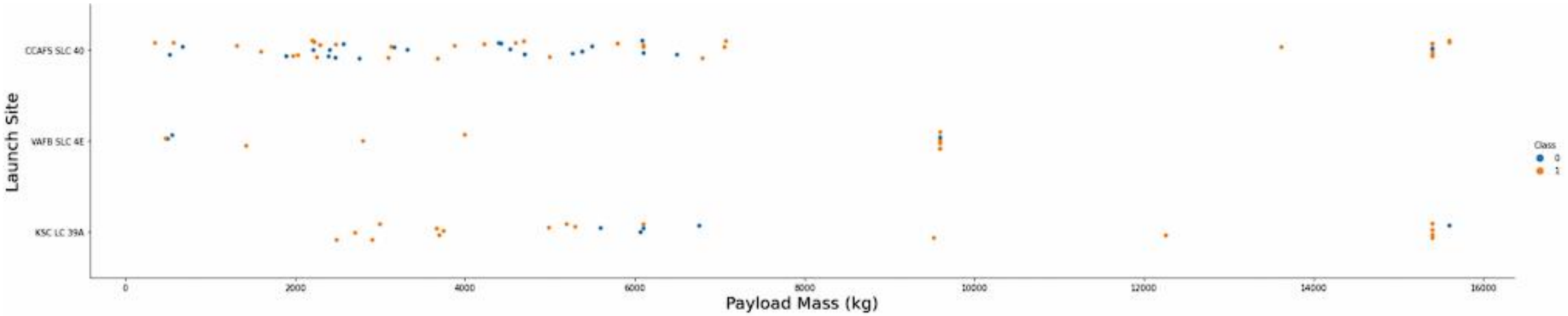
Flight Number vs. Launch Site

- Clarification:
- • The earliest flights generally fizzled while the most recent flights generally succeeded.
- • The CCAFS SLC 40 send off site has about a portion of all send-offs.
- • VAFB SLC 4E and KSC LC 39A have higher achievement rates.
- • It tends to be expected that each new send off has a higher pace of progress.



Payload vs. Launch Site

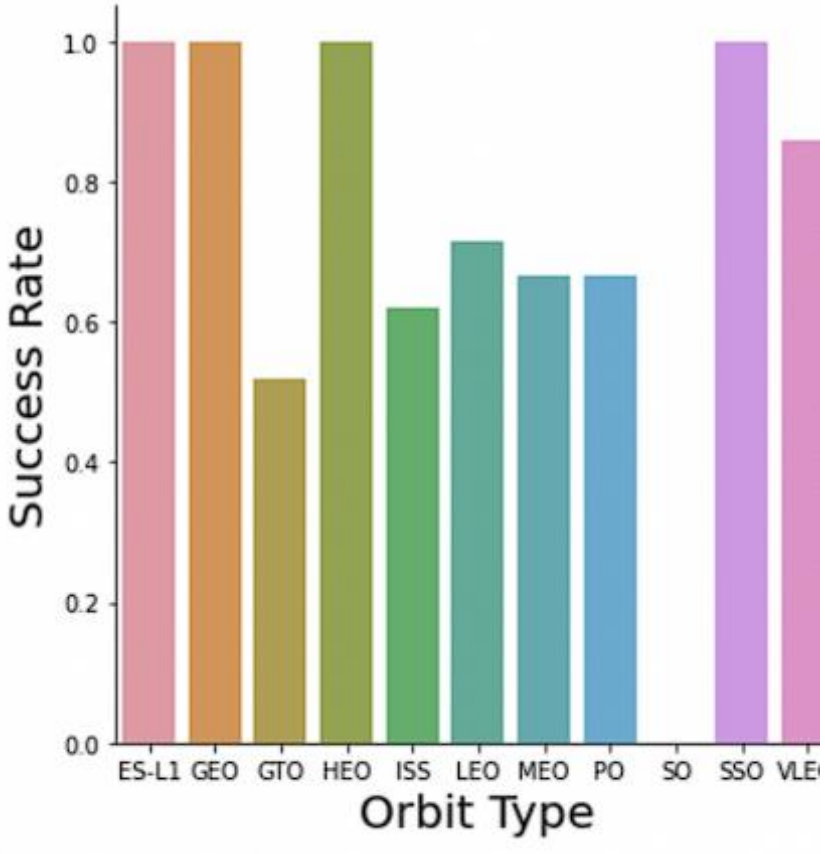
- Clarification:
- • For each send off site the higher the payload mass, the higher the achievement
- rate.
- • The greater part of the send-offs with payload mass north of 7000 kg were fruitful.
- • KSC LC 39A has a 100 percent achievement rate for payload mass under 5500 kg as well.



Success Rate vs. Orbit Type

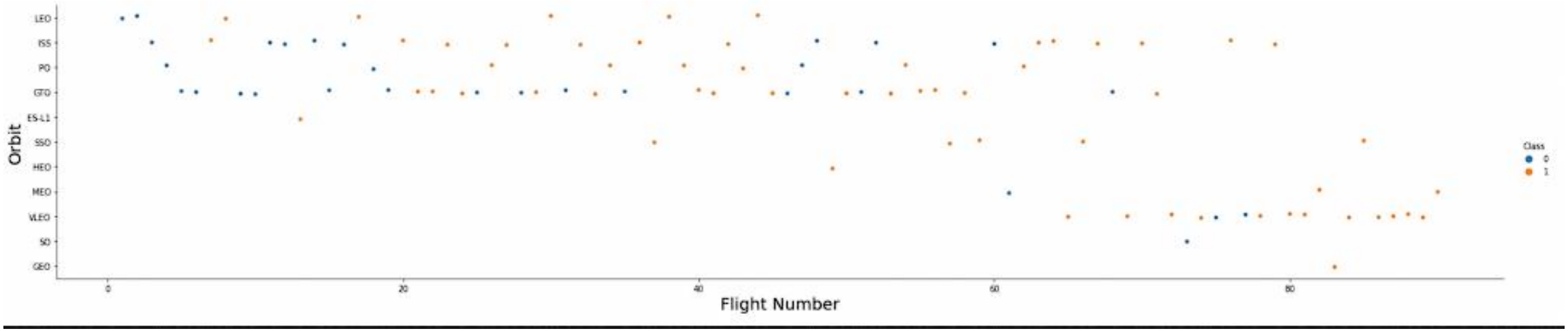
Clarification:

- Orbits with 100% success rate: - ES-L1, GEO, HEO, SSO
- Orbits with 0% success rate: - SO
- Orbits with success rate between 50% and 85%: - GTO, ISS, LEO, MEO, PO



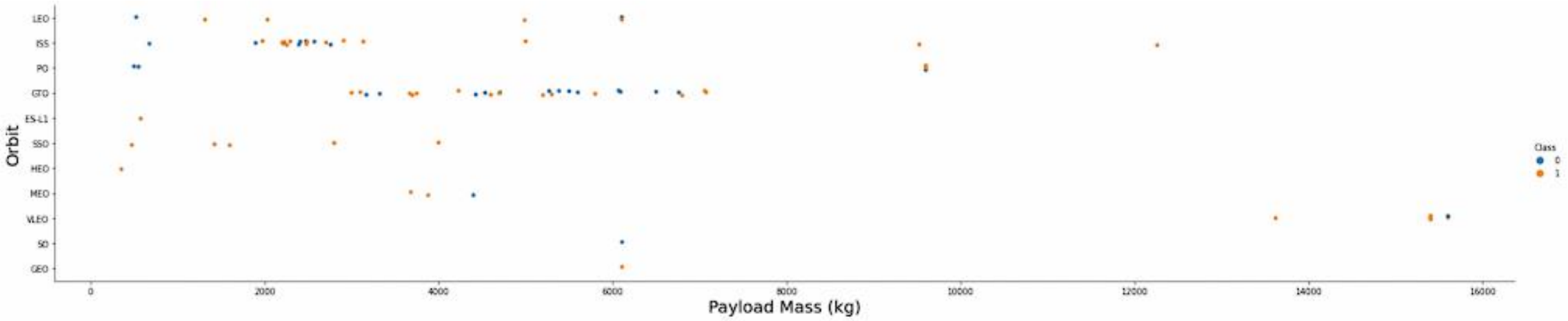
Flight Number vs. Orbit Type

- Clarification:
- • In the LEO circle the Achievement seems connected with the quantity of flights;
- then again, there is by all accounts no connection between flight
- number when in GTO circle.



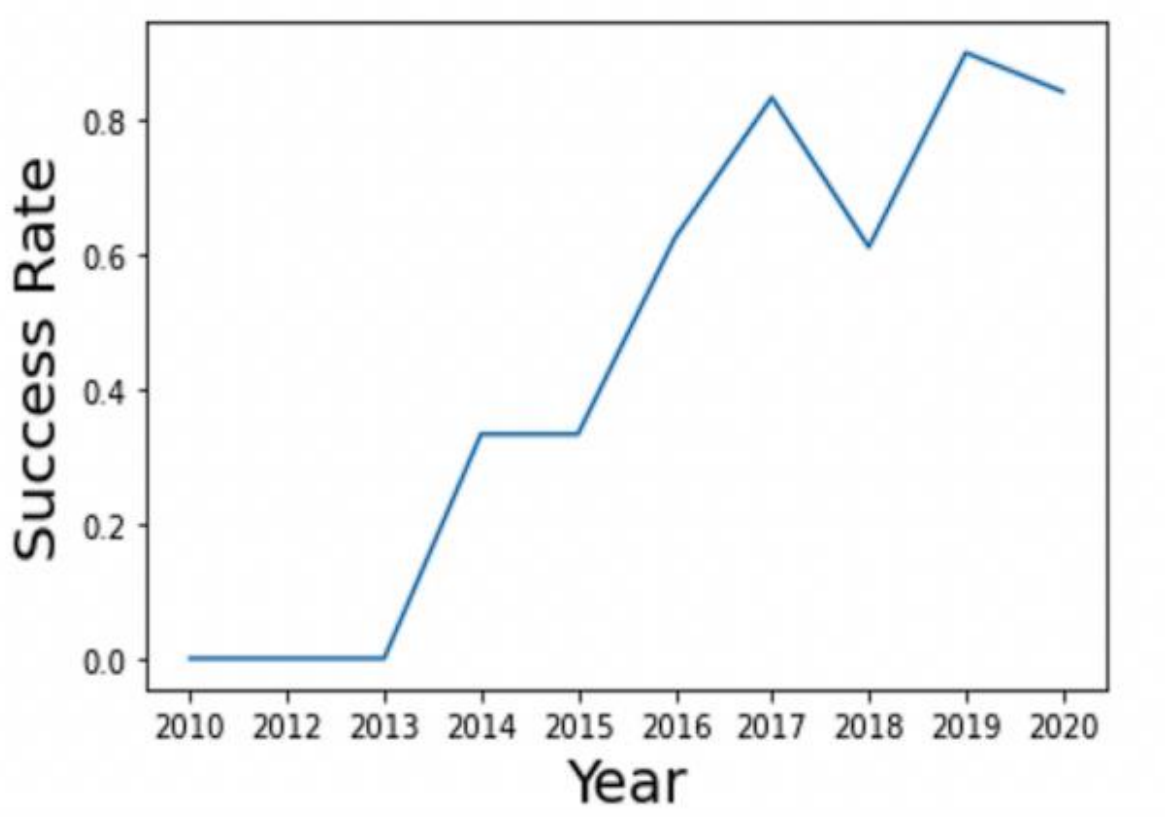
Payload vs. Orbit Type

- Clarification:
- • Weighty payloads impact GTO circles and positive
- on GTO and Polar LEO (ISS) circles.



Launch Success Yearly Trend

- Clarification:
- • The achievement rate
- beginning around 2013 kept
- expanding till 2020



EDA with SQL

All Launch Site Names

- Clarification:
- • The achievement rate
- beginning around 2013 kept
- expanding till 2020

```
In [4]: %sql select distinct launch_site from SPACEXDATASET;
```

```
 * ibm_db_sa://wzf08322:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/bludb
Done.
```

Out[4]:

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

Launch Site Names Begin with 'CCA'

- Clarification:
- Showing 5 records where send off locales start with the string 'CCA'.

```
In [5]: %sql select * from SPACEXDATASET where launch_site like 'CCA%' limit 5;
 * ibm_db_sa://wzf08322:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90108kqblod8lcg.databases.appdomain.cloud:31198/bludb
Done.
```

Out[5]:

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

- Clarification:
- • Showing the complete payload mass conveyed by sponsors sent off by
- NASA (CRS).

```
In [6]: %sql select sum(payload_mass__kg_) as total_payload_mass from SPACEXDATASET where customer = 'NASA (CRS)';
 * ibm_db_sa://wzf08322:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/bludb
Done.
```

Out[6]:

total_payload_mass
45596

Average Payload Mass by F9 v1.1

- Clarification:
- Showing normal payload mass conveyed by promoter adaptation F9 v1.1.

```
In [7]: %sql select avg(payload_mass__kg_) as average_payload_mass from SPACEXDATASET where booster_version like '%F9 v1.1%';
 * ibm_db_sa://wzf08322:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/bludb
Done.
```

Out[7]:

average_payload_mass
2534

First Successful Ground Landing Date

- Clarification:
- • Posting the date when the main fruitful landing result in ground
- cushion was accomplished.

```
In [8]: %sql select min(date) as first_successful_landing from SPACEXDATASET where landing__outcome = 'Success (ground pad)';
 * ibm_db_sa://wzf08322:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90108kqblod8lcg.databases.appdomain.cloud:31198/bludb
Done.
```

Out[8]:

first_successful_landing
2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

- Clarification:
- Posting the names of the sponsors which have progress in drone transport furthermore, have payload mass more noteworthy than 4000 yet under 6000.

```
In [9]: %sql select booster_version from SPACEXDATASET where landing__outcome = 'Success (drone ship)' and payload_mass__kg_ between 4
        000 and 6000;
```

```
 * ibm_db_sa://wzf08322:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90108kqblod8lcg.databases.appdomain.cloud:31198/bludb
Done.
```

Out[9]:

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

Total Number of Successful and Failure Mission Outcomes

- Explanation:
- Listing the total number of successful and failure mission outcomes.

```
In [10]: %sql select mission_outcome, count(*) as total_number from SPACEXDATASET group by mission_outcome;
 * ibm_db_sa://wzf08322:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:31198/bludb
Done.
```

Out[10]:

mission_outcome	total_number
Failure (in flight)	1
Success	99
Success (payload status unclear)	1

Boosters Carried Maximum Payload

- Clarification:
- Posting the names of the promoter variants which have conveyed the most extreme payload mass.

```
In [11]: %sql select booster_version from SPACEXDATASET where payload_mass__kg_ = (select max(payload_mass__kg_) from SPACEXDATASET);
 * ibm_db_sa://wzf08322:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90108kqb1od8lcg.databases.appdomain.cloud:31198/bludb
Done.
```

Out[11]:

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

- Clarification:
- • Posting the bombed landing results in drone transport, their supporter
- adaptations and send off site names for the months in year 2015.

```
In [12]: %%sql select monthname(date) as month, date, booster_version, launch_site, landing__outcome from SPACEXDATASET
             where landing__outcome = 'Failure (drone ship)' and year(date)=2015;
```

```
 * ibm_db_sa://wzf08322:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90108kqb1od8lcg.databases.appdomain.cloud:31198/bludb
Done.
```

Out[12]:

MONTH	DATE	booster_version	launch_site	landing__outcome
January	2015-01-10	F9 v1.1 B1012	CCAFS LC-40	Failure (drone ship)
April	2015-04-14	F9 v1.1 B1015	CCAFS LC-40	Failure (drone ship)

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

- Clarification:
- • Positioning the count of landing results (like Disappointment (drone boat) or Achievement (ground cushion)) between the date 2010-06-04 and 2017-03-20 in plummeting request.

```
In [13]: %%sql select landing__outcome, count(*) as count_outcomes from SPACEXDATASET
             where date between '2010-06-04' and '2017-03-20'
             group by landing__outcome
             order by count_outcomes desc;
```

```
 * ibm_db_sa://wzf08322:***@0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90108kqblod8lcg.databases.appdomain.cloud:31198/bludb
Done.
```

Out[13]:

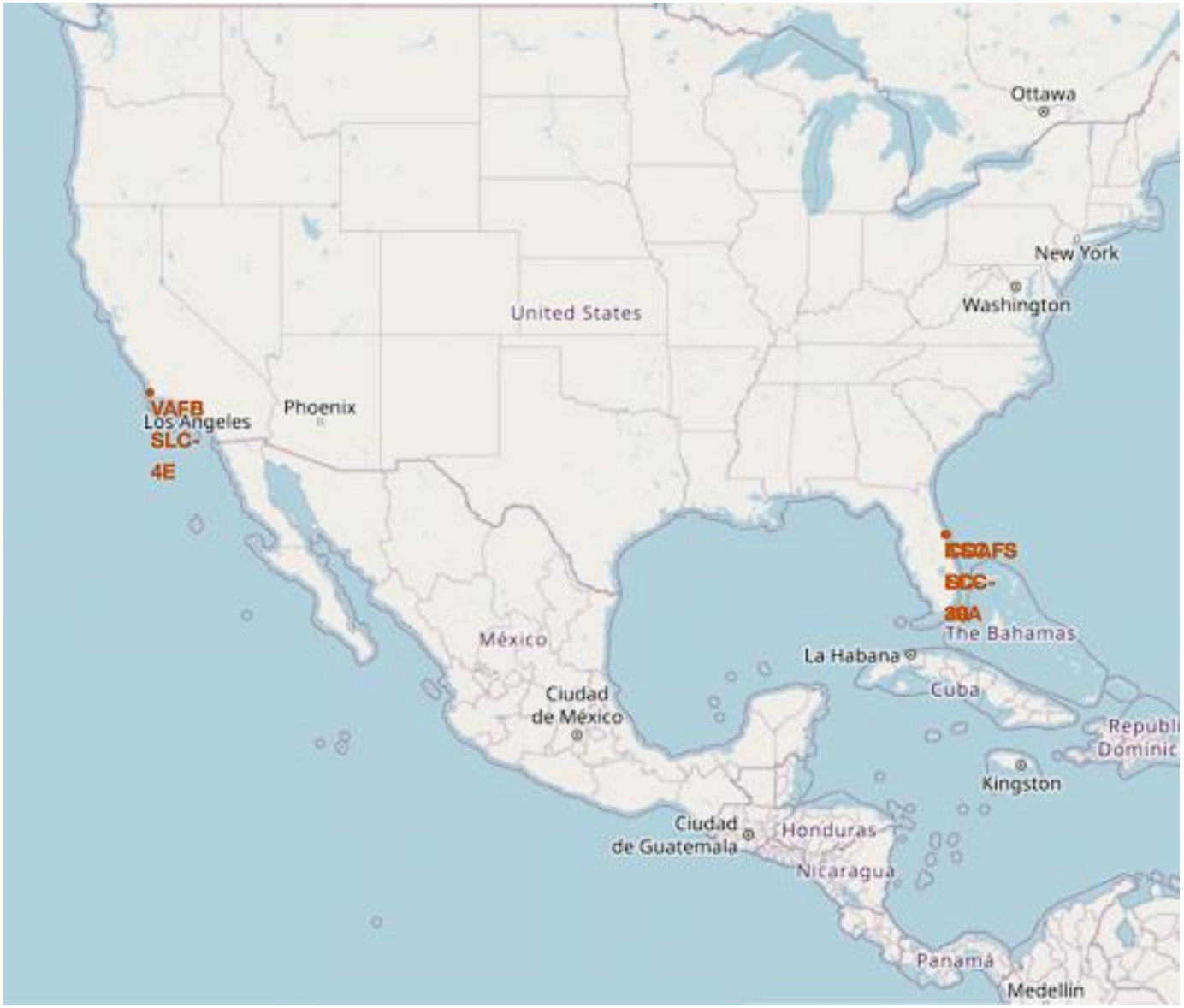
landing__outcome	count_outcomes
No attempt	10
Failure (drone ship)	5
Success (drone ship)	5
Controlled (ocean)	3
Success (ground pad)	3
Failure (parachute)	2
Uncontrolled (ocean)	2
Precluded (drone ship)	1

Section 3

Launch Sites Proximities Analysis

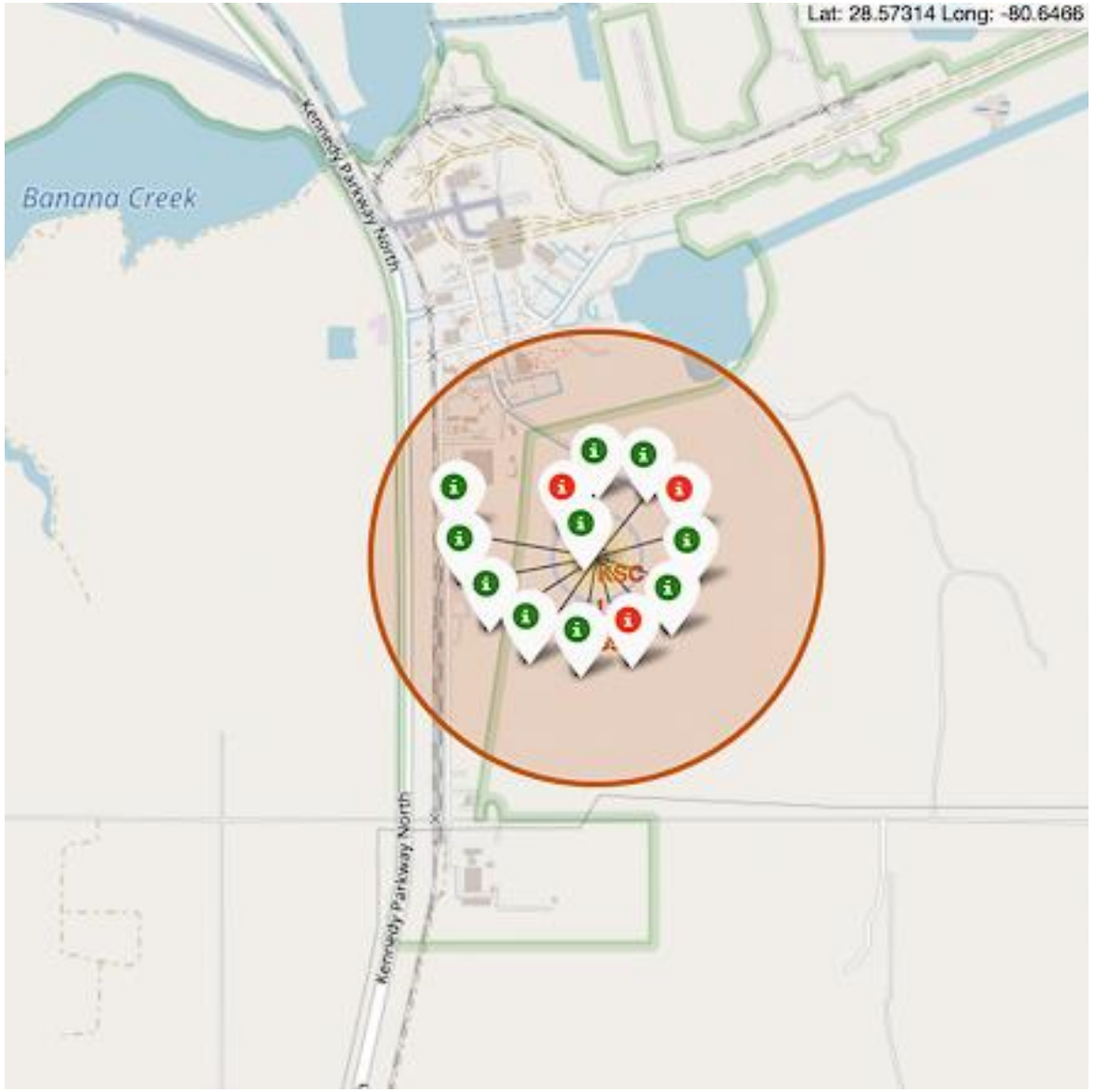
All launch sites' location markers on a global map

- Clarification:
- The vast majority of Send off destinations are in vicinity to the Equator line. The land is moving quicker at the equator than some other put on the outer layer of the Earth. Anything on the outer layer of the Earth at the equator is now moving at 1670 km/hour. On the off chance that a boat is sent off from the equator it goes up into space, and it is likewise moving around the
- Earth at a similar speed it was moving prior to sending off. This is a result of dormancy.
- This speed will assist the space apparatus with keeping up a sufficient speed to remain in circle.
- All send off locales are in exceptionally closeness to the coast, while sending off rockets towards the sea it limits the gamble of having any garbage dropping or xploding close to individuals.



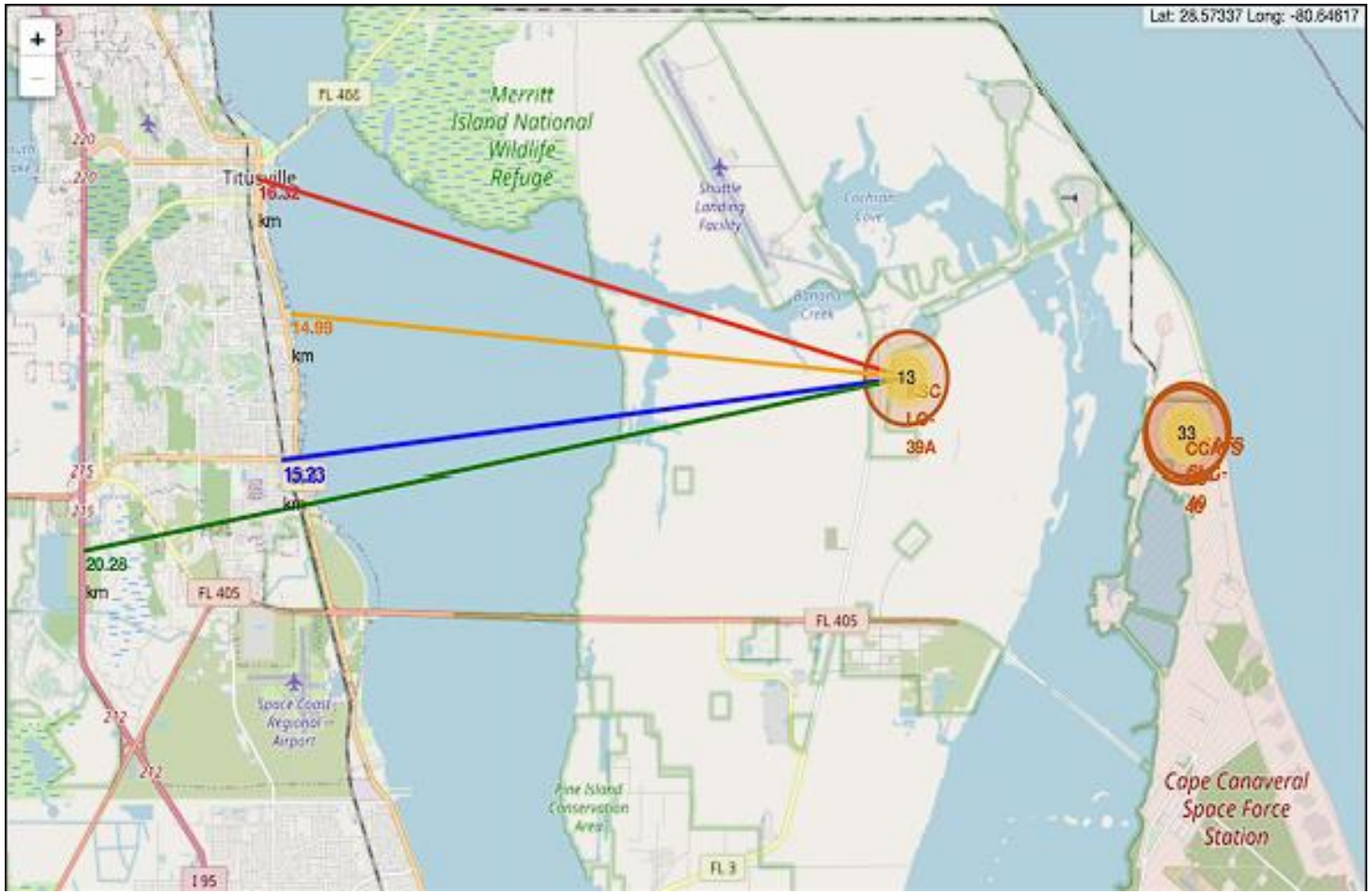
Colour-labeled launch records on the map

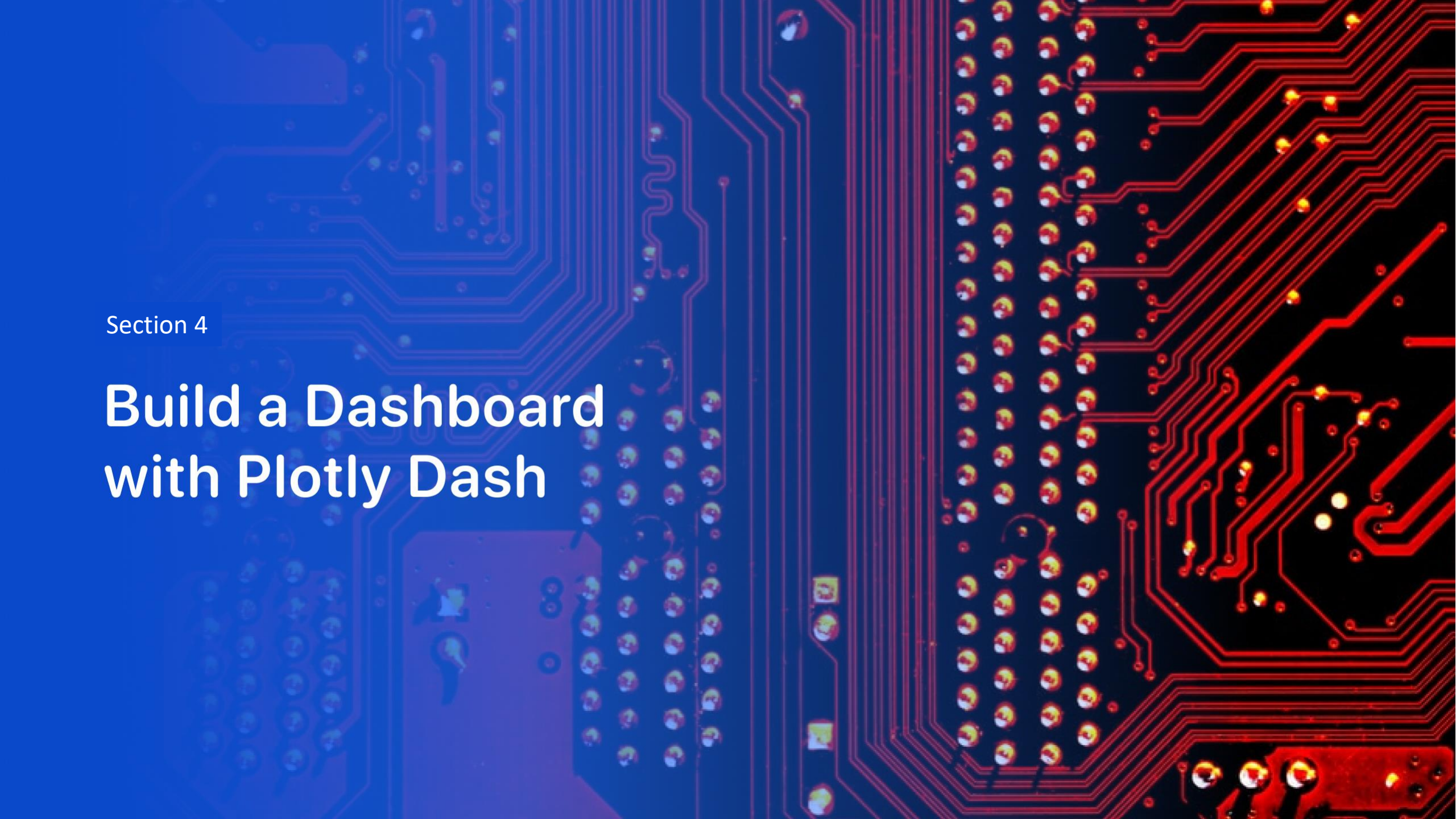
- Clarification:
- • From the variety named markers we ought to have the option to without any problem
- recognize which send off destinations have moderately high achievement rates.
- - Green Marker = Fruitful
- Send off
- - Red Marker = Bombed Send off
- • Send off Site KSC LC-39A has an exceptionally high Achievement Rate.



Distance from the launch site KSC LC-39A to its proximities

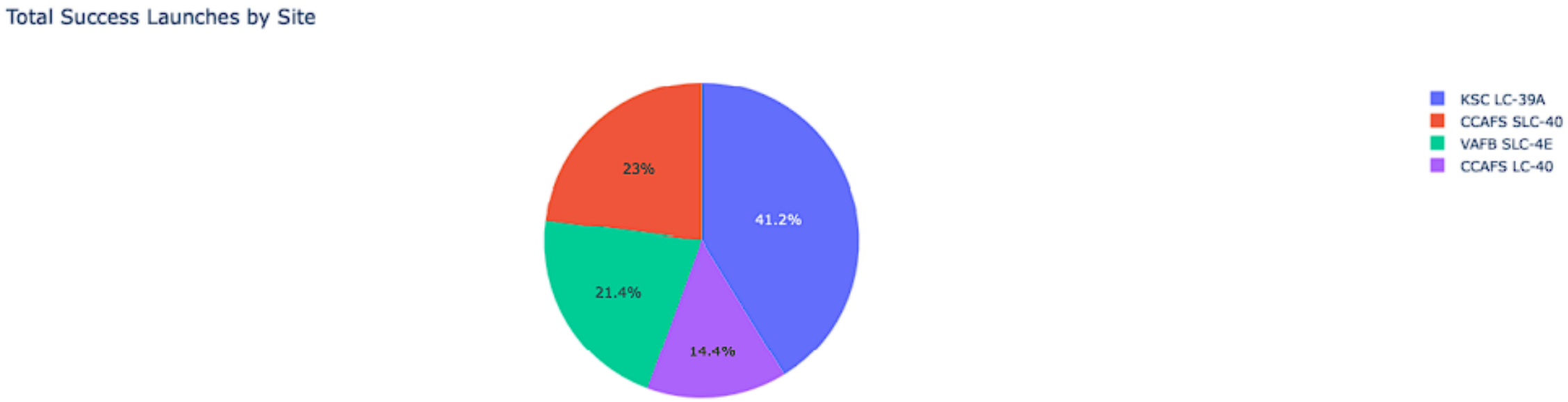
- Clarification:
- • From the visual investigation of the send off site KSC LC-39A we can obviously see thatit is:
- - relative near railroad (15.23 km)
- - relative near expressway (20.28 km)
- - relative near shore (14.99 km)
- • Likewise the send off site KSC LC-39A is relative near its nearest city
- Titusville (16.32 km).
- • Bombed rocket with its fast can cover distances like 15-20 km in couple of moments. It very well may be possibly perilous to populated regions.




Section 4
Build a Dashboard with Plotly Dash

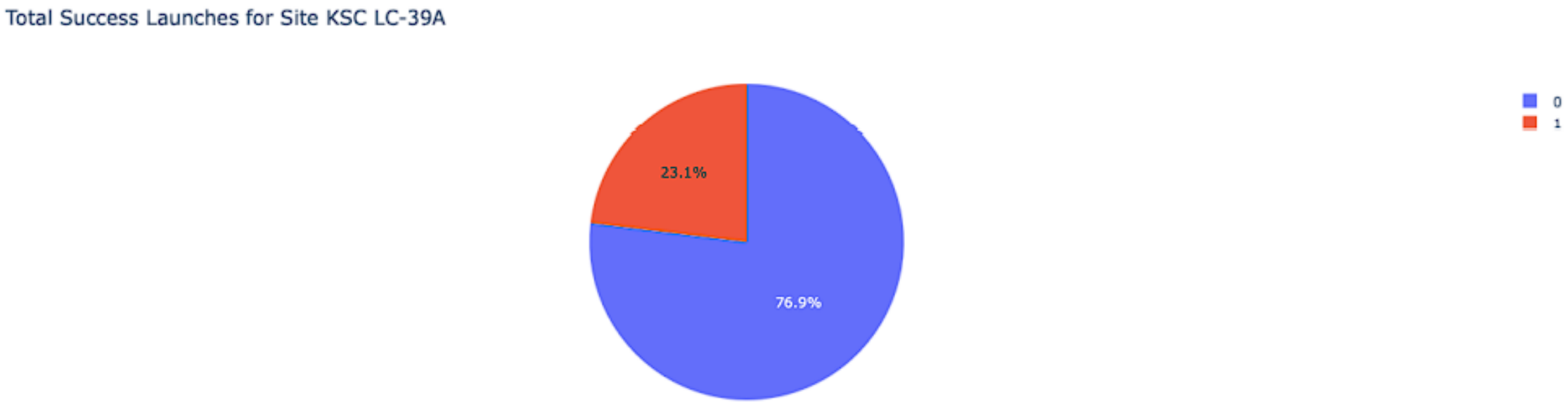
Launch success count for all sites

- Clarification:
- • The diagram obviously shows that from every one of the locales, KSC LC-39A has the most effective send-offs.



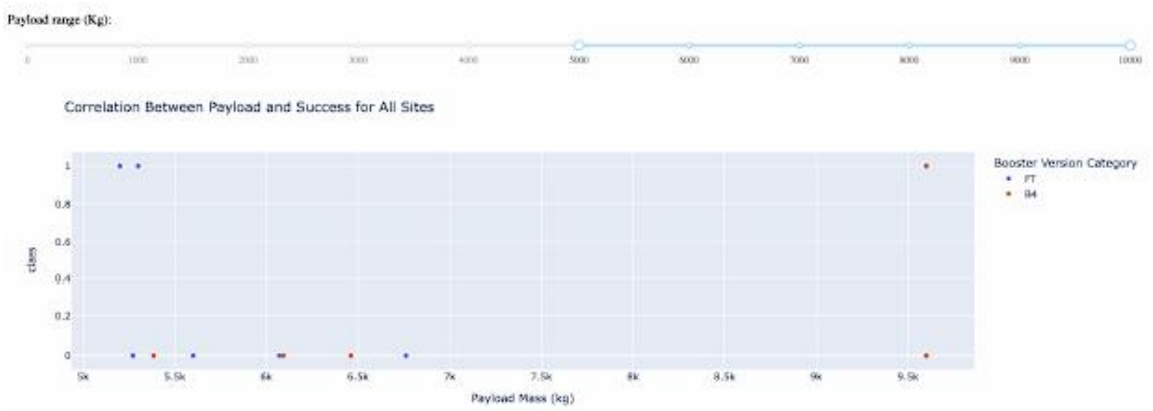
<Dashboard Screenshot 2>

- Clarification:
- • KSC LC-39A has the most noteworthy send off progress rate (76.9%) with 10 fruitful and just 3 bombed arrivals.



Payload Mass vs. Launch Outcome for all sites

- Clarification:
- • The outlines show that payloads somewhere in the range of 2000 and 5500 kg have the most elevated achievement rate.



Section 5

Predictive Analysis (Classification)

Classification Accuracy

Clarification:

- In view of the scores of the Test Set, we can not affirm which technique performs best.
- Same Test Set scores might be because of the little test size (18 examples). Subsequently, we tried all strategies in light of the entire Dataset.
- The scores of the entire Dataset affirm that the best model is the Choice Tree Model. This model has higher scores, yet additionally the most elevated precision.

Scores and Accuracy of the Test Set

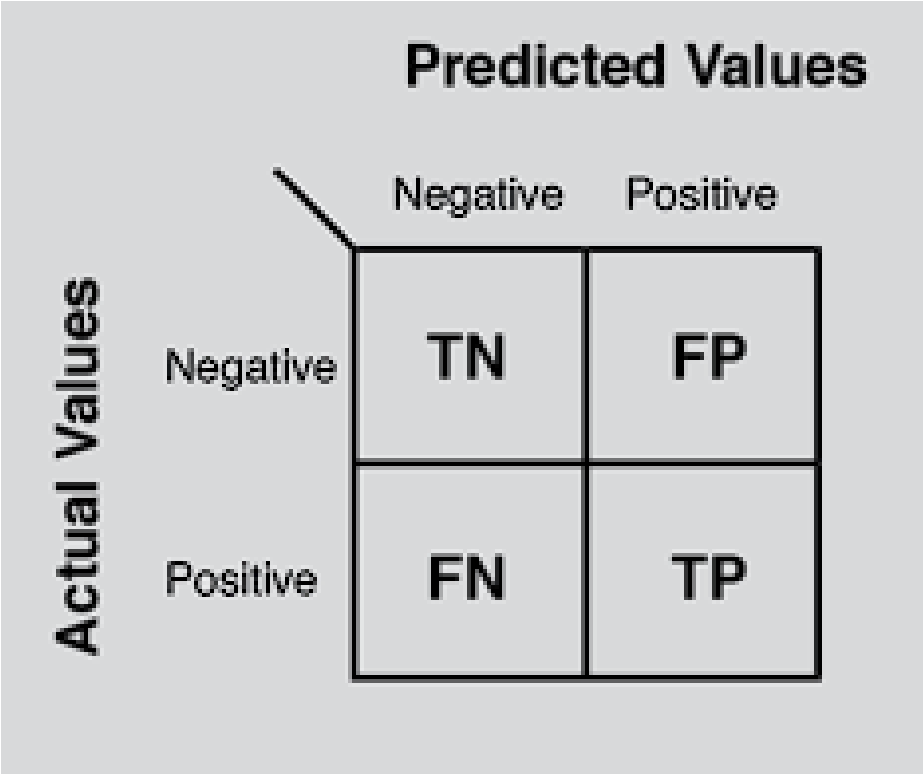
	LogReg	SVM	Tree	KNN
Jaccard_Score	0.800000	0.800000	0.800000	0.800000
F1_Score	0.888889	0.888889	0.888889	0.888889
Accuracy	0.833333	0.833333	0.833333	0.833333

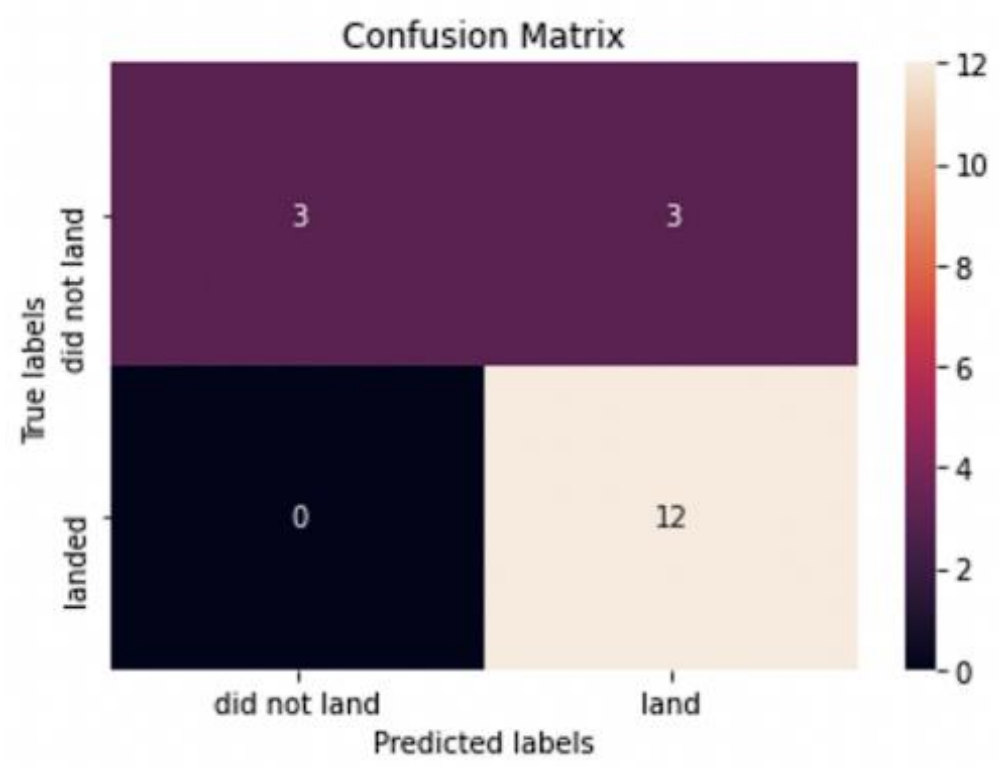
Scores and Accuracy of the Entire Data Set

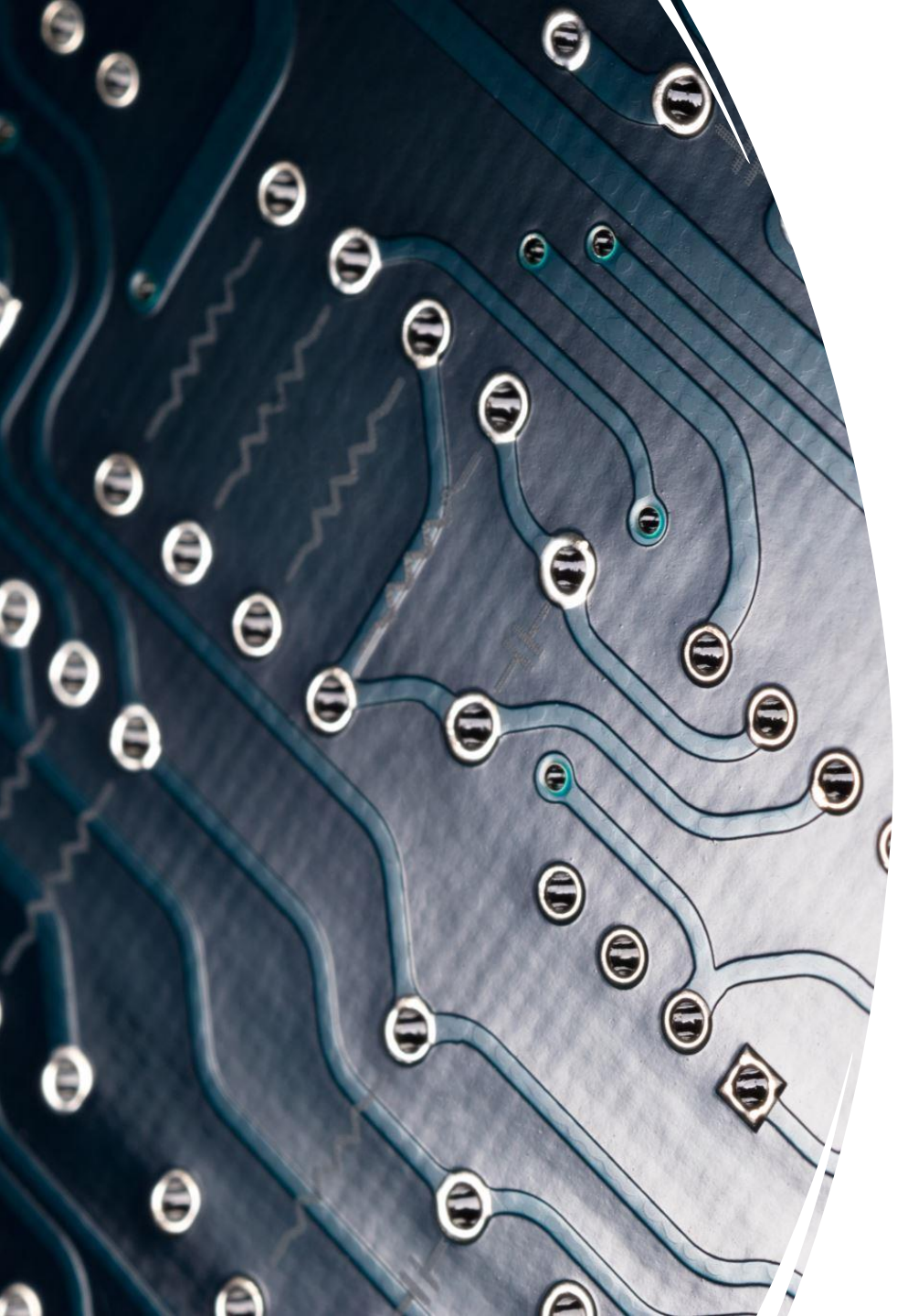
	LogReg	SVM	Tree	KNN
Jaccard_Score	0.833333	0.845070	0.882353	0.819444
F1_Score	0.909091	0.916031	0.937500	0.900763
Accuracy	0.866667	0.877778	0.911111	0.855556

Confusion Matrix

- Clarification:
- Inspecting the disarray lattice, we see that strategic relapse can recognize
- between the various classes. We see that the serious issue is misleading up-sides.







Conclusions

- Choice Tree Model is the best calculation for this dataset.
- Dispatches with a low payload mass show improved results than dispatches with a bigger payload mass.
- The majority of send off locales are in nearness to the Equator line and every one of the destinations are in exceptionally closeness to the coast.
- The achievement pace of dispatches increments throughout the long term.
- KSC LC-39A has the most elevated achievement pace of the send-offs from every one of the locales.
- Circles ES-L1, GEO, HEO and SSO have 100 percent achievement rate.

Appendix

Special Thanks to:

- Instructors
- Coursera
- IBM

Thank you!