Machine Learning Methods for Orbital Debris Characterization: Report 1

Anson Biggs

2022-02-14

Gathering Data

To get started on the project before any scans of the real debris are made available I opted to find similar 3D models online and to process them as if they were data collected by my team. GrabCad is an excellent source of high quality 3D models and all of the models have at worst a non-commercial license making them suitable for this study. To start I downloaded a high quality model of a 6U CubeSat, which coincidentally enough was designed for detection of orbital debris. This model consists of 48 individual parts most of which are unique.

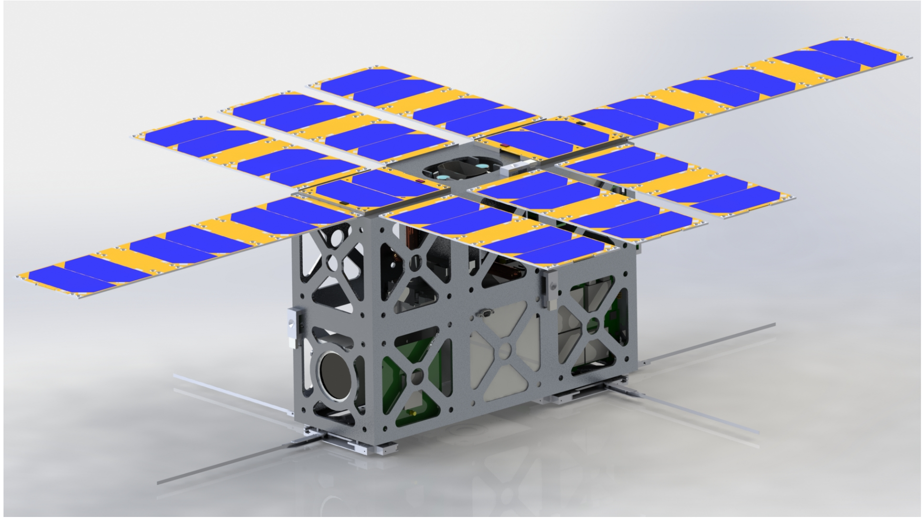

Figure 1: CubeSat Used for Analysis

Data Preparation

To begin analysis of the pieces of the satellite key properties from each piece needed to be gathered. The most accessible CAD software for me at the moment is Fusion 360, but most any CAD software is capable of giving properties for a model. The only way to export the properties is to the clipboard of my computer and required clicking on each individual part of the CubeSats assembly. This task was easily automated using an autohotkey script to automatically push my computers clipboard to a file. Below is an example of what each file looks like, but only a portion of the file for brevity. The text file is generated in a way that makes it difficult to parse so a separate piece of code was used to collect all of the data from all of the part files and then turn it into a `.csv` file for easy import into Matlab.

```
...
Physical
    Mass    108.079 g
    Volume  13768.029 mm^3
    Density 0.008 g / mm^3
    Area    19215.837 mm^2
    World X,Y,Z 149.00 mm, 103.80 mm, 41.30 mm
    Center of Mass  101.414 mm, 102.908 mm, -0.712 mm
...
```

The full file of the compiled parts properties from Fusion 360 can be seen here. This method gave 22 columns of data but most of the columns are unsuitable for characterization of 3D geometry. Its important that the only properties considered are scalars that are independent of a models orientation of position in space. Part of the data provided was a moment of inertia tensor. This was computed down to I_x, I_y, and I_z, which was then used to compute an $\bar{I}$. Then bounding box length, width, and height were used to compute a total volume that the object takes up. In the end the only properties used in the analysis of the parts were: mass, volume, density, area,bounding box volume, $\bar{I}$. Some parts also had to be removed due to being outliers to the final dataset is 44 rows and 6 columns. Below is a splom plot which is a great way to visualize data of high dimensions. As you can see most of the properties correlate with one another.

Now that the data is processed and clean characterization in Matlab can begin. The original idea was to perform *PCA*, but the method had difficulties producing meaninful results. This is likely due the dataset being very small for machine learning, and that the variation in the data is high. Application of *PCA* will be visited again once the dataset grows. The first step for characterization is importing our data into Matlab.

```
data = readmatrix('prepped.csv');
```

Next *k-means* will be used to cluster the data. Since it is hard to represent data in higher dimensions than two only two columns of data will be provided for

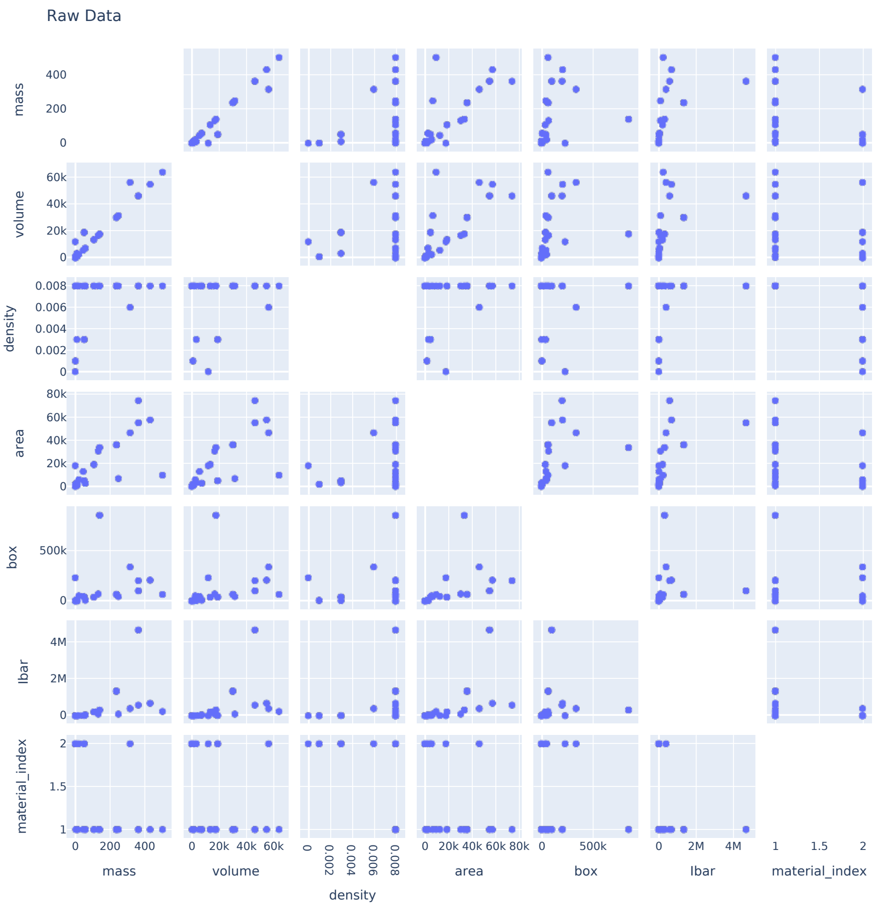

Figure 2: Data before clustering

the clustering. For now I think it makes most intuitive sense to treat volume and mass as the most important columns, and the volume vs mass plot shows 3 fairly distinct groups.

```
[idx,C] = kmeans(data(:,1:2),3);
```

We can look at the distribution of parts in each cluster to ensure that the each cluster has a good amount of data. Since *k-means* is an iterative method, relies on a user guess, and randomness its important to make sure that the clusters make some sense.

```
histcounts(idx) =

    22    13     9
```

Then we plotting Volume vs. Mass using our clusters we get the following plot. These make intuitive sense, but it is clear that the dataset needs much more data for Cluster 3.

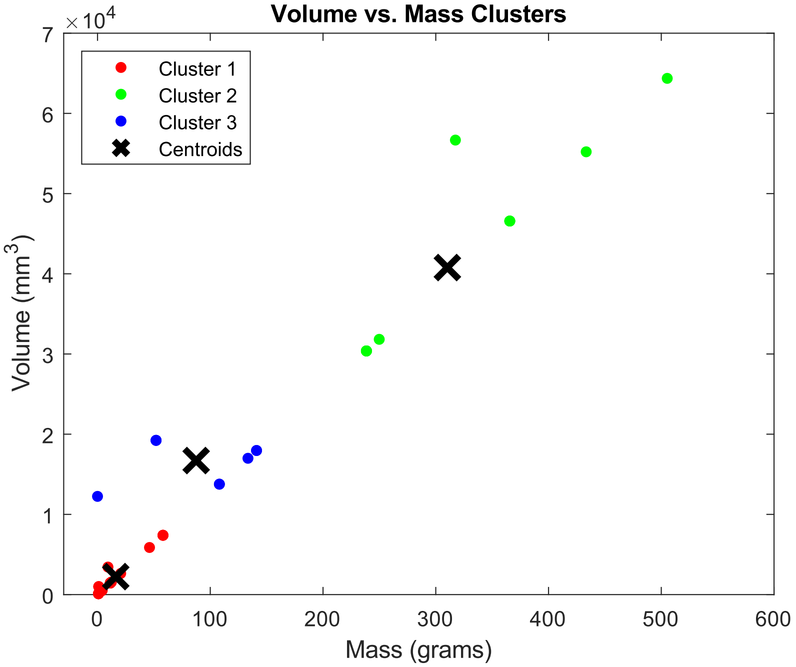

Figure 3: Volume and Mass clusters

Below is all of the data clustered. Since the *k-means* only used Mass and Volume to come up with its clusters some of the properties do not cluster well against eachother. This is also a powerful cursory glance at what properties are correlated.

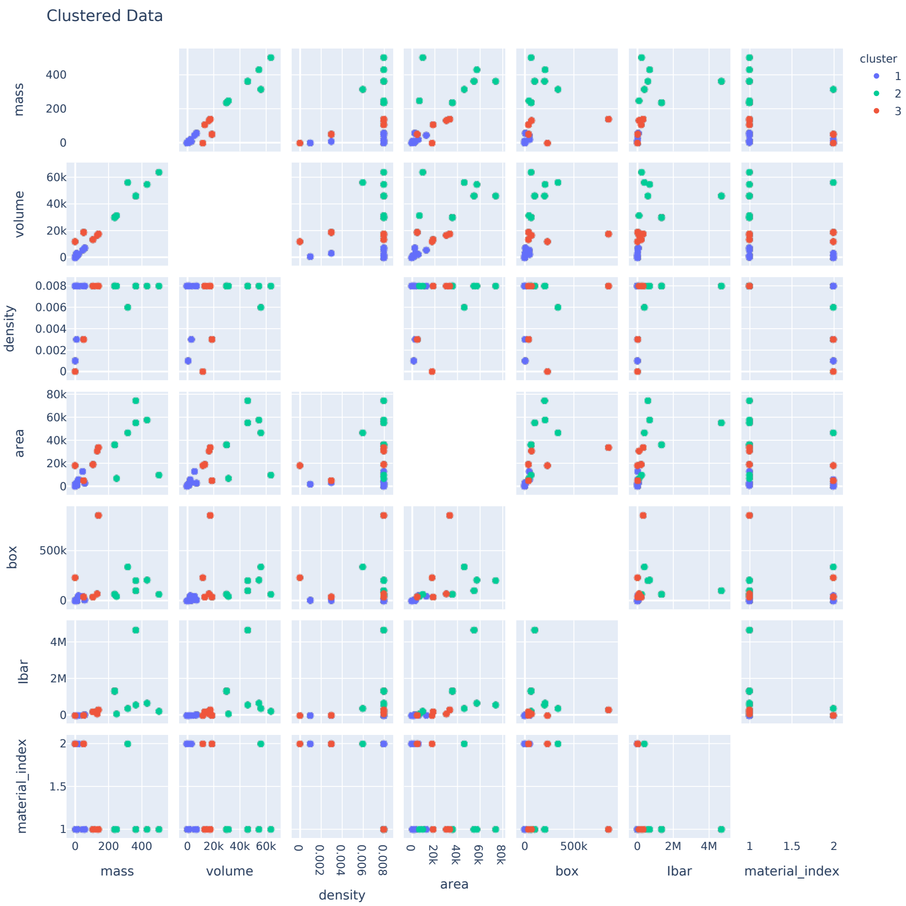

Figure 4: Clusters for all data

Next Steps

The current dataset needs to be grown in both the amount of data aswell ass the variety of data. The most glaring issue with the current dataset is that there are only two different material types. Modern satellites, and therefore their debris, are composed of dozens of unique materials. The other and harder to fix issue is finding more properties of the data. Properties such as cross sectional are or aerodynamic drag would be very insightful but at the moment there is no good way to collect that data. Thankfully with the 3D scanner methods to obtain more properties can be developed and then applied over the entire dataset.

Once the dataset is grown more advanced analysis can begin. PCA is the current goal, and can hopefully be applied by the next report.