Data and Info Visualization - DV Exploration

Andrew Struthers, Gihane Ndjeuha, Kollin Trujillo, Nathan Chapman, Nick Haviland

January 2023

1 Part 1 - DV Introduction

id	x1	x2	x3	x4	class
0	3	4	1	1	1
1	1	2	2	3	1
2	4	5	4	1	1
3	5	4	4	1	2
4	5	3	2	2	1
5	4	4	2	1	1
6	4	4	2	3	2
7	5	2	1	4	1
8	2	3	4	5	2
9	2	4	3	4	2
10	4	2	4	2	1
11	5	5	3	4	3
12	3	4	3	3	2
13	4	1	3	1	1
14	5	1	4	5	3
15	4	3	2	3	1
16	3	5	5	3	3
17	4	1	2	3	1
18	1	2	5	4	1
19	1	2	4	5	1

Figure 1: Sample generated dataset

For this section, we generated some random data using Excel's random number generator. We have a handful of data points, 4 dimensions, and 3 classes in this dataset. Opening up DV, we can load this .csv file an view the data using the default visualization.

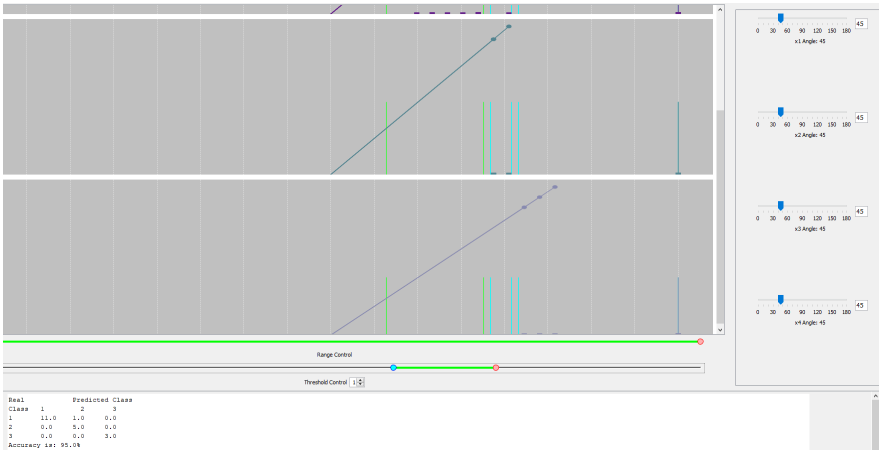

Figure 2: Default visualization

In this default visualization, we can see that each angle representing the different dimensions is initialized to 45 degrees. The data that we had actually had some nice separation with this visualization, and we were able to set some thresholds that resulted in a 95% accurate classification. This is honestly pretty rare and we were certainly not expecting that.

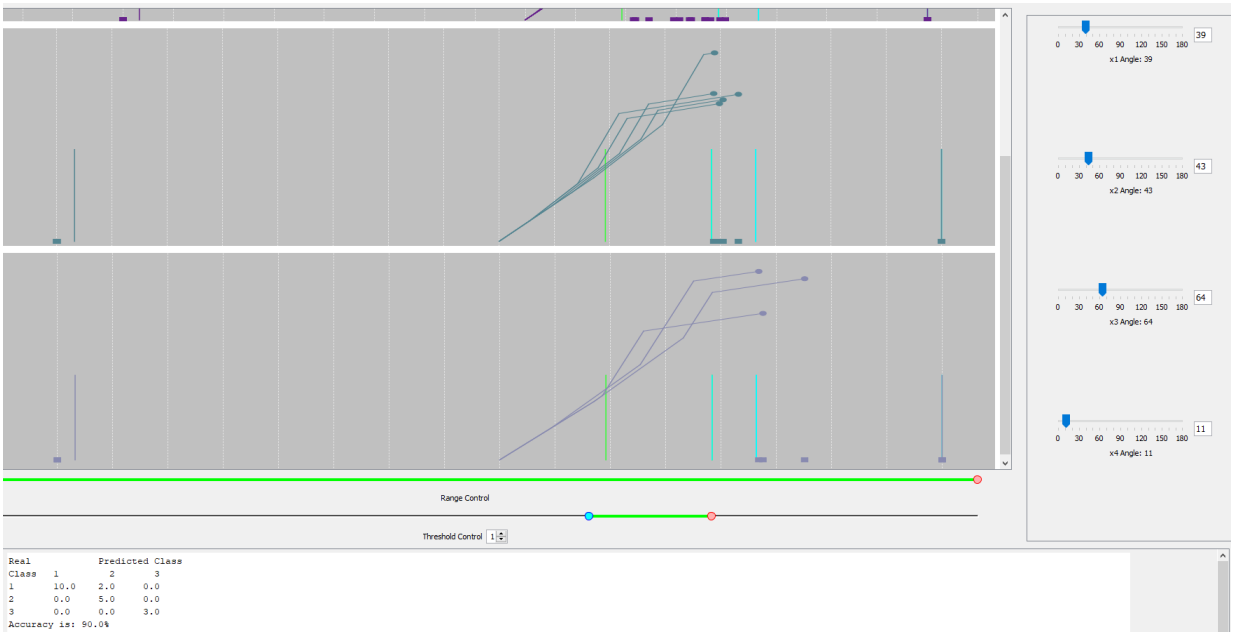

Figure 3: Better sample visualization

In fact, when we started to mess with the angles, we realized that it was actually pretty hard to mimic that 95% success we had using the default visualization. We used some of the built in functions, the randomize angle generator, and the optimize angle generator, to get us this separation. Comparing this to the default view, however, we can see that this only gave us a 90% accuracy classification rate. Regardless, we can see that each datapoint is represented by a series of line segments that have different angles to help us visualize the data. This visualization method is good for collapsing a lot of information down into easy to split categories, represented by the dots on the x-axis, in line with the starting point for each of the class visualizations.

2 Part 2 - Real n-D Data

2.1 Dry Bean Classification - Andrew

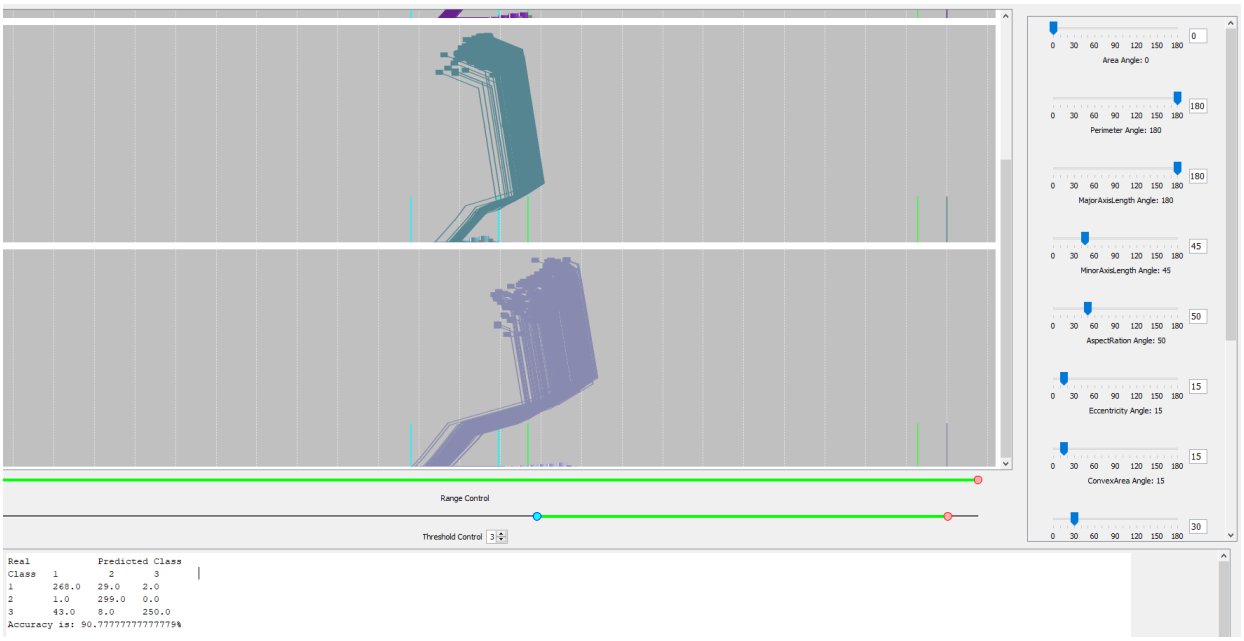

Figure 4: Separated visualization with three classes in DV

The dataset I worked with initially consisted of 16-dimension classifications, but I realized that 4 of the dimeionons were shape analysis that shouldn't be factored in to the overall classification. The new data is 12-dimensions. Initially, the dataset contained 13,611 images of 7 different bean classes, but I trimmed the data down to 3 classes with around 300 rows of data per class. This data came from UC Irvine Machine Learning Repository. Looking at the figure above, we can see that there are still many rows and a lot of different dimensions. This made classification a bit tricky, but the strategy I used to try and get the best visualization possible was to set the angle of some of the dimensions I felt would have smaller impact on the classification to around 90 degrees, which means they would have very little impact on the x-position of the final lines. After a lot of fiddling, I ended up with this visualization, which has an accuracy of 90.777777%. There is still some overlap between the full data, especially between the first and third classes, but this visualization and threshold range provides relatively good results.

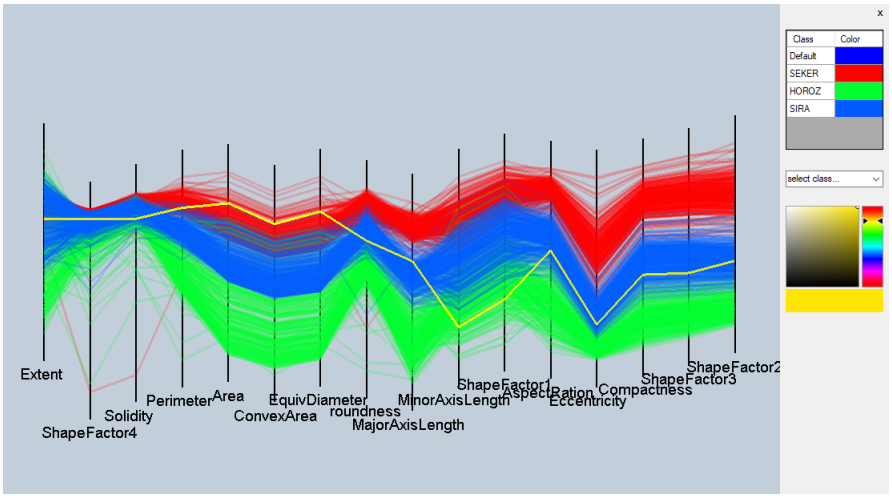

Figure 5: Cleaned up, separable classes in VisCanvas

Looking at the DV visualization compared to the VisCanvas visualization, we can see that in VisCanvas, the first two dimensions that really started showing seperation were perimeter and area. Knowing this, I set the area and perimeter angles to 0 degrees, meaning they would effectively have the largest impact on the overall shape of the line being generated. Some of the dimensions that didn't really help increase the quality of the seperation, as far as VisCanvas could tell, were set close to 90 degrees to diminish their impact on the x-position of the data. This strategy resulted in the relatively good visualization in DV, so I would say that while both of these tools aren't perfect on their own, together they can form some really solid classification information.

2.2 Concrete Compressive Strength - Gihane

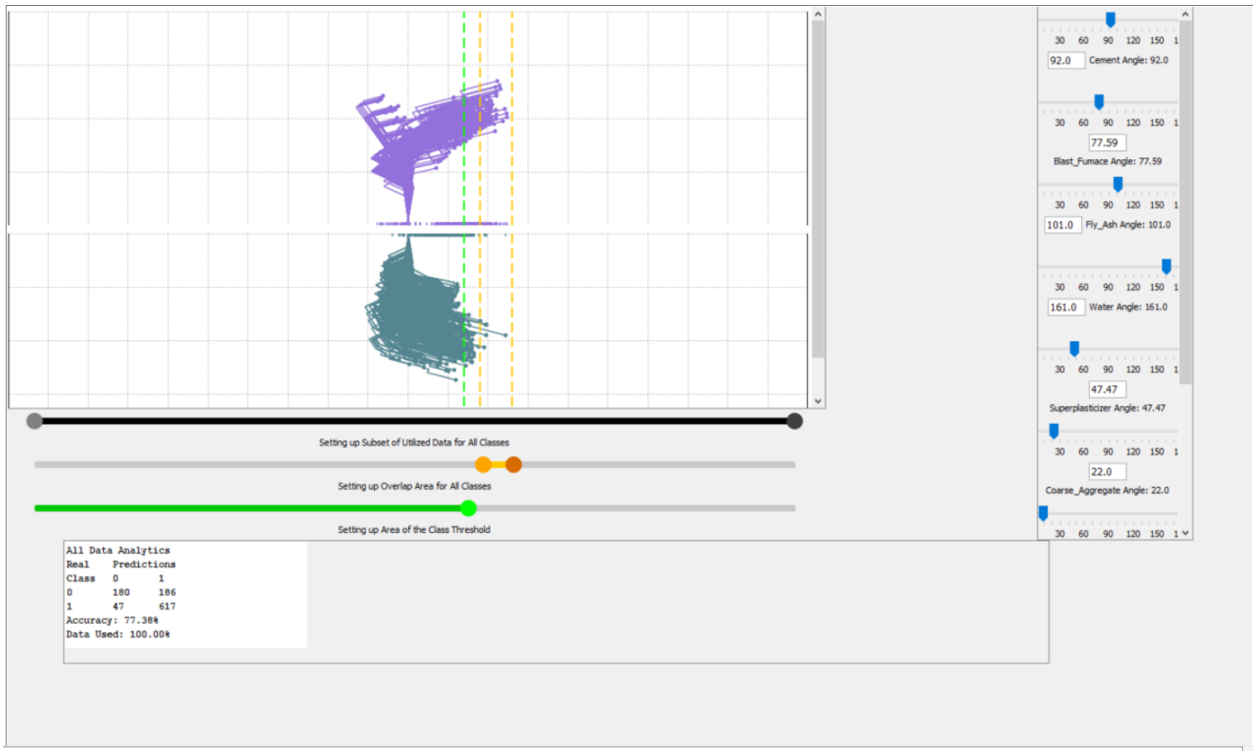

Figure 6: Concrete strength dataset

This dataset has 1030 entries, 3 classes and 9 dimensions and 8 quantitative input variables. Each variables represent a component in the mixture of the Concrete Compressive Strength. The measurement is in kg in a m^3 mixture. With the DV program, it's easy to separate the classes by adjusting angles compared to the VisCanvas where we had to move things around.

2.3 Raisin Dataset - Kollin

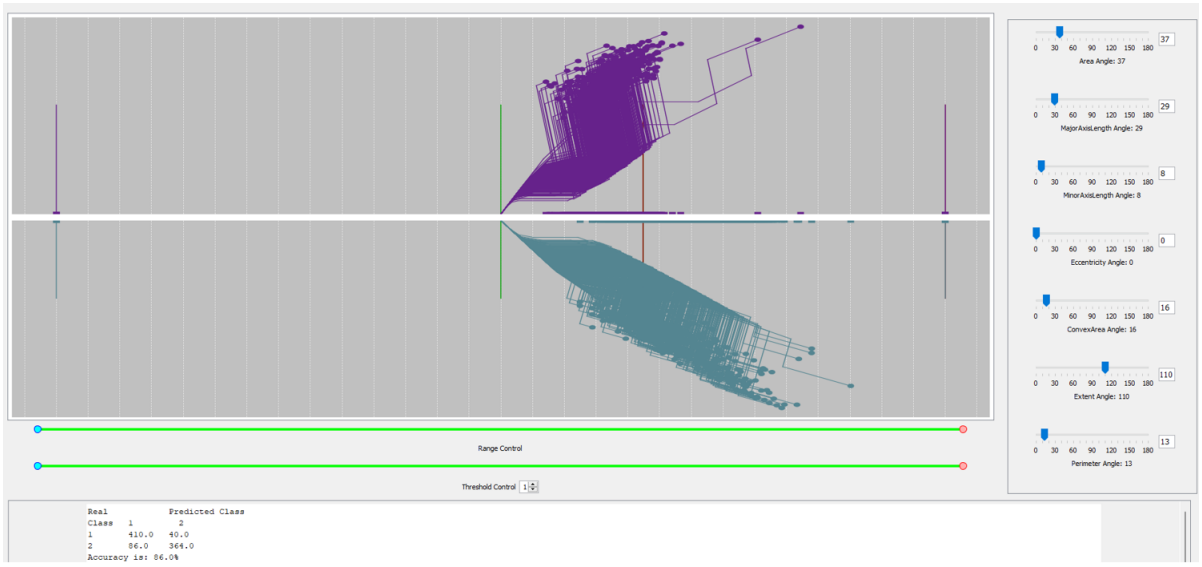

Figure 7: Raisin visualization in DV

I chose the Raisin Dataset. This dataset consists of two classes of measurements from images of Kecimen and Besni raisin varieties. There are 900 total measurements of raisins with an equal distribution between the two varieties of raisins. The measures, or dimensions, of the data include a variety of morphological measurements. These measurements include area, major axis length, minor axis length, eccentricity, convex area, extent, perimeter, and then the classification into the two varieties. This makes the dataset 7-dimensional. The figure shown above is the default GLC-L from DV whereas the figure below is the parallel line coordinates from VisCanvas.

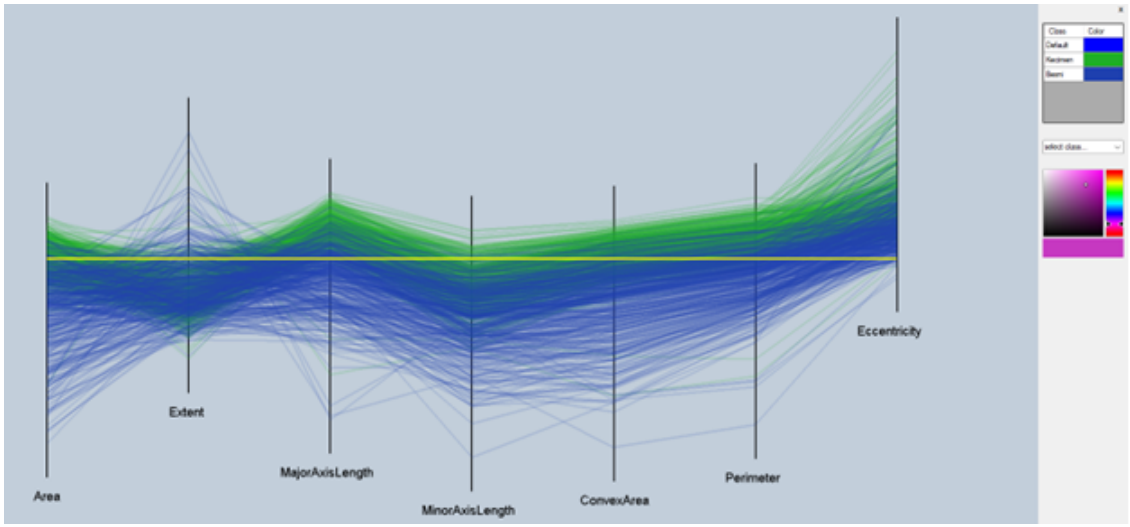

Figure 8: Same raisin dataset in VisCanvas

In comparison to the parallel line coordinates, GLC-L also suffers with difficulty from data occlusion

due to density of data points. One thing additionally that this software does is that it allows for the user to perform classification from the data that is projected to the axis. This allows the user to gauge what class the data might belong to. Due to data that sometime overlaps between the classes in this dataset it is difficult to adequately classify this dataset much beyond the 86.0% that I received after fiddling around with projects and thresholds for quite some time. Overall this methodology is useful and quite distinct from parallel lines yet also deals with basic geometric features such as lines and angles with the caveat being connects at axes versus at some user-defined point in space. I like this data visualization for more classification purposes yet since the thresholds and confusion matrix serve to give us a metric of classification and accuracy. However, where DV suffers is that I do not feel that I can get as much information "at-a-glance" as with VisCanvas.

2.4 Pulsar Candidates in High Time Resolution Survey - Nate

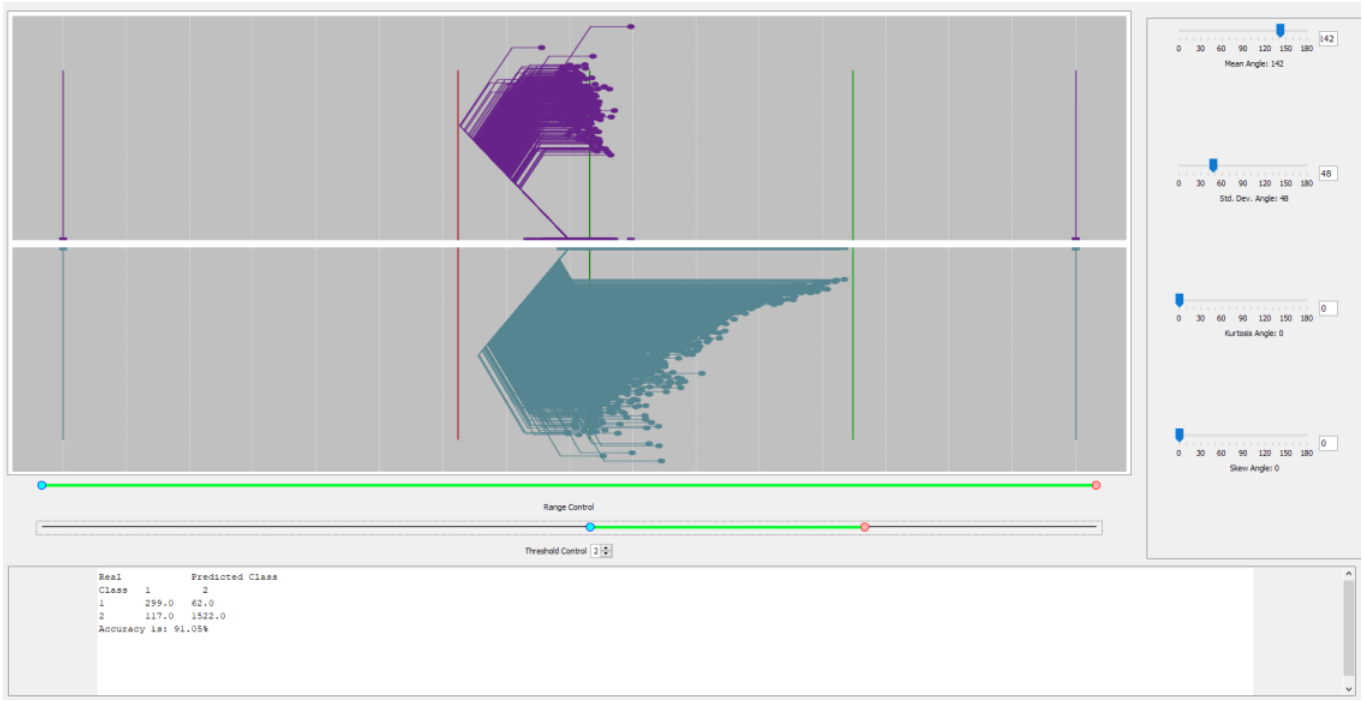

Figure 9: Separated pulsar candidate data in DV

This data set is made up of 2000 (originally 17,898) entities from 2 different classes (pulsars in green and not pulsars in red), each of which has 8 dimensions. For this analysis, only a sample of the data was considered along with a further reduction in the dimensions. The discarded data was left out for the purposes of more efficient visualization. **It is important to note that the labels of the dimensions are not referencing the data itself, but rather the integrated profile of each potential pulsar.**

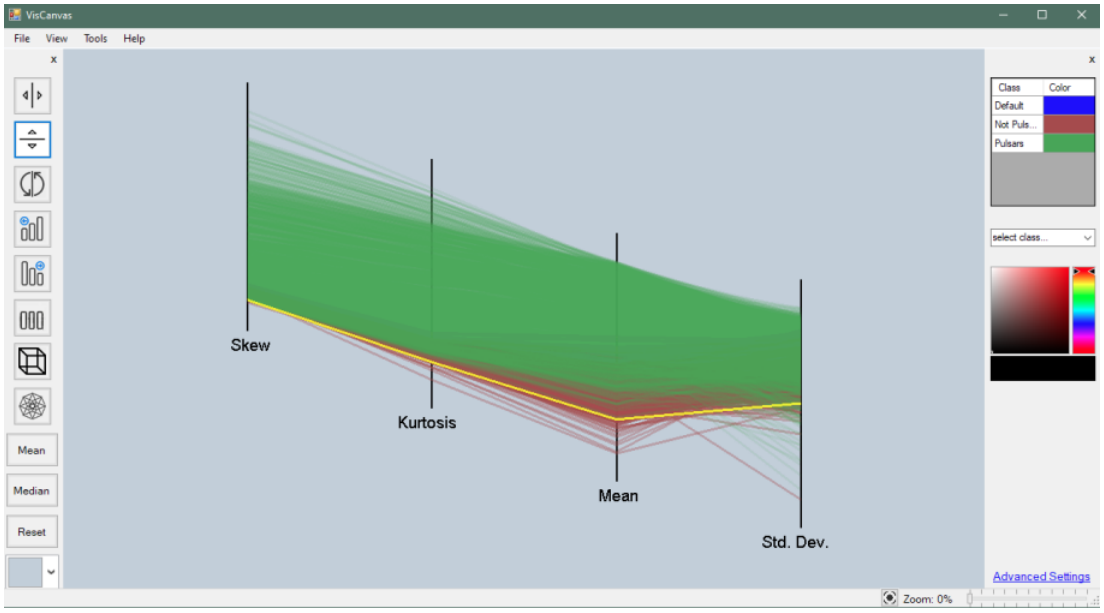

Figure 10: VisCanvas correlation data

Because the skew angle had the greatest effect on the clustering of the vectors and the overlap of the class clusters, this visual analysis suggests that the skew angle has the greatest physical effect as well. Visualizing the same data in VisCanvas also shows a strong correlation between skew and class. Between these two visual analyses, we can be more confident in concluding that pulsar candidacy heavily relies on the skew angle.

2.5 Wireless Indoor Localization - Nick

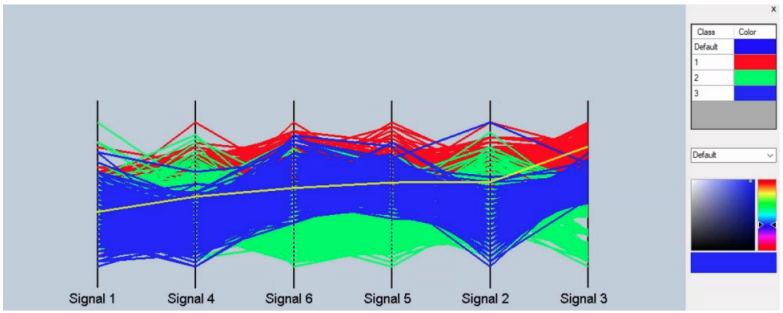

Figure 11: Raw localization dataset

The data used here is Wireless Indoor Localization. Originally, this data set contained 2,000 observations from 4 different classes. I trimmed this data set down to include only 3 classes, and 1,500 observations. This data set measures Wi-Fi signal strength from 6 different smartphones, measured in an unmentioned unit. These 6 different smartphones and 3 rooms make this a 6-dimensional data set with three classes.

The figure above shows that room one, shown in red, tends to have a better overall signal strength than either of the other two rooms. It also shows that the highest overall signal strength tends toward the 3rd smartphone. This figure is presented after changes made to it in VisCanvas to explore the data set.

The figure below shows the data imported and unchanged in the DV Program, with an accuracy of 68.6%, with room 1 first, room 2 second, and room 3 third.

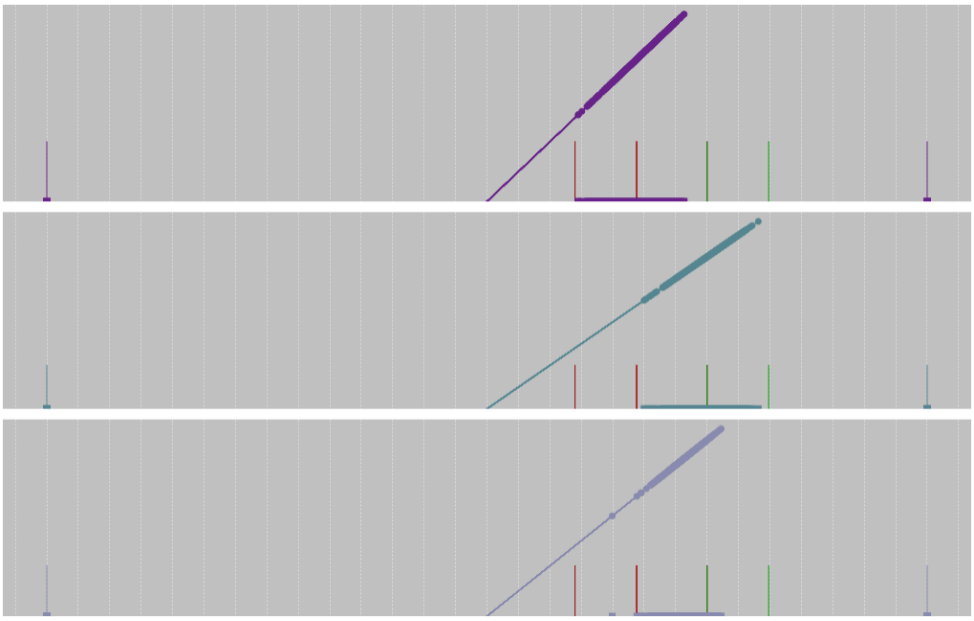

Figure 12: Default DV visulized signal data

To make changes to the angles in the DV program, I used the figure from Vis Canvas as a basis for the angles in each dimension. After using the DV program for a while, I decided to change the angles to try to differentiate between rooms 2 and 3 the most, since they are most similar. The third signal separates the three rooms the most cleanly, so I emphasized it, placing it at 0 degrees. I placed additional emphasis on signals 5 and 6, putting them at 30 degrees. I placed signals 2 and 4 at 45 degrees since they are still differentiating rooms 2 and three, and since signal 1 had almost no difference between the 3 rooms, I put it and 0 degrees. With this, I ended up with the figure below at 71.5% accuracy.

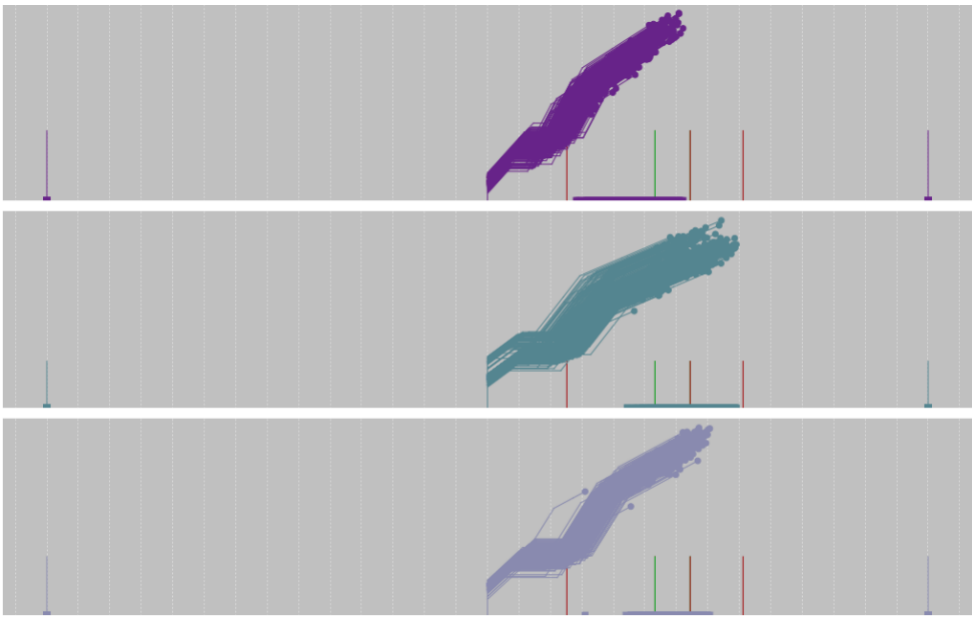

Figure 13: Manipulated DV signal data

In the exploration between the two programs, I noticed that VisCanvas does a better job at differentiating between classes at each dimension, so for this data set it is easier to how signal strength compares with each room. The DV program does a better job at showing the general trend of the data. From the DV program, one can see that room 2 has the best signal strengths overall, while room 1 has the worst. With the DV program, it is also apparent that rooms 2 and 3 have a lot of overlap in their data, making differentiating between them visually difficult. I believe that they work well in conjunction with one another, especially since the DV program requires setting angles for visualization. I could not get the optimization for the angles to work in the program, so using the VisCanvas program as a basis for the angles made the job easier than just guess and check.