Xinyang Sun, xs920, 01912874; Yuancheng Shao, ys3420, 01886903

1. Section A: Data Preparation

Part 1 The original dataset is captured from PR2 robot with two Biotac tactile sensors [3]. After plenty of attempts on plotting pressure, vibration, temperature (PVT) and the Electrodes data using Matlab. We found that the holding procedure seems starting at about the 25th time step. And then after the 25th time step, we cannot find a single time step that can overtake any other time step on data differentiation. So, We thought the performance of data at each time step should be similar as long as we choose the data after 25th time step. Then we chose the 50th time step.

For visualisation of one object trials, we chose '$acrylic\_211\_04\_HOLD.mat$' and plot PVT and electrodes data separately in Figure 1 and Figure 2. The subplot of electrodes in Figure 2 clearly shows the data in Matlab but not in pdf, so we also plot Figure 3 for reference.

Part 2 As we discussed in Part 1, we saved PVT and electrodes data at the 50th time step separately in '$F0\_PVT\_50.mat$' and '$F0\_Electrodes\_50.mat$' and these two matrix have structures of 3x60 and 19x60.

Part 3 The 3D scatter plot is shown in Figure 4. For color consistency throughout this work, we used same hexadecimal codes of color for each objects which are shown in Table 1.

2. Section B: Principal Component Analysis

Part 1 First need to be mentioned here, we use two ways to preprocess data. The first is normal standardisation, the other is zero-centred approach ($x_i^{'} = x_i - \overline{x}$, where the x_i is the ith data, $\overline{x}$ is the mean of all data, $x_i^{'}$ is the zero-centred ith data) which can be used to compare the performance in following discussions [4].

Principal Component Analysis (PCA) covariance matrix which is calculated from the dataset '$F0\_PVT\_50.mat$' is shown below[5]. The eigenvectors (each column) and corresponding eigenvalues originated from the covariance matrix which are sorted in order of variance from largest to smallest. [1].

$$Covariance = \begin{bmatrix} 1 & -0.357 & -0.319 \\ -0.357 & 1 & 0.269 \\ -0.319 & 0.269 & 1 \end{bmatrix}$$

$$Eigenvector = \left[\begin{array}{c|c|c} -0.604 & -0.163 & 0.780 \\ 0.576 & 0.587 & 0.569 \\ 0.551 & -0.793 & 0.261 \end{array}\right]$$

$$Eigenvalue = \begin{bmatrix} 1.631 & 0.735 & 0.636 \end{bmatrix}$$

Figure 5 shows the standardised data with three principle components. The axes are now assigned to three principal components (The red, blue, green line represents PC1 (Principle Component 1), PC2 and PC3 respectively). Then, it needs to turn the data onto the corresponding principle components by an orthogonal projection. The projection process is done by formula: $\widetilde{x}^p = e^T x_n$ [1]. Where, the x_n is standardised data coordinate, the e^T is the eigenvector and the $\widetilde{x}^p$ is the transformed coordinate through the projection process. The PC1 is the component with largest variance, the rest PC attribute can be known in the same manner.

Figure 6 shows the standardised data reduced to two dimensions and assigned to the two most principle components (The red, green line represents PC1, PC2 respectively). It can be found that data scatters wider for PC1 than PC2 which indicates the PC1 has larger variance, the main distribution part is varies from -2.5 to 1.5 in PC1, whereas the other in PC2 is from -1 to 1 which is an obvious smaller range.

From Figure 7, the one dimension number lines show data contribution for three individual principle components, however, the standardised data can not generate an obvious wider distribution. The zero-centred data (which shown in Appendix (Figure 8)) indicates perfect performance and comparison result between three components with small outliers, the PC1 which is the component with largest variance shows the widest distribution among all three PCs, and there is a gradual decline in variance level from PC1 to PC3.

Part 2 Figure 9 shows the variance changing process with principle numbers for 19 electrodes. It can be found that the variance sharply decreases from PC1 to PC3 and shows a very small change after PC4. It is unnecessary and inconvenient to plot the rest principle components for analysing and visualizing data. Thus, the three components with largest variance are suitable to taken for plotting the electrodes data which is shown in Figure 10.

3. Section C: Linear Discriminant Analysis

Part 1-3 After applying Linear Discriminant Analysis (LDA) between every two properties in standardised data, we got three plots shown in Figure 11 with LDA lines. Then we applied LDA to the three-dimensional data shown in Figure 12.

Either property within pressure, vibration and temperature is similar between the black foam and the car sponge. In sensor readings, we could see that the range of values have overlapped between two objects within three properties, therefore, when LDA applied, the line with maximised separation cannot classify the two object classes well and usually take these two objects as one same class.

Part 4 The data overlaps a lot between the black foam and the car sponge, so we decided to choose two objects that have little or no overlap between each other to see whether the performance would be better. , After taking Figure 1 as a reference and multiple tests of two objects, we chose acrylic and black foam as the best alternative pair to repeat LDA, the results are shown in Figure 13 and Figure 14. From the new results, we could see that when two objects' physical properties varies a lot, LDA can clearly distinguish two objects through obvious different clusters.

4. Section D: Clustering

Part 1 To have a proper choice on clustering algorithms, we applied Hierarchy and K-means with Euclidean and Manhattan to see which algorithm performs better, the results are shown in Figure 15, Figure 16, Figure 21 and Figure 19 (for K-means clustering figures, the cluster centroids are shown as black cross).

From the results, we could see that in same Euclidean distance metric, K-means shows a better performance on clustering than Hierarchy, but in Manhattan distance metric, Hierarchy shows a better performance on clustering than K-means, especially in distinguish black foam, car sponge, flour sack and kitchen sponge. Also, we found that no matter which distance metric we use, Hierarchy algorithm cannot distinguish between acrylic and steel vase.

Comparing the results with real-life experience, we could confused between kitchen sponge and car sponge, acrylic and steel vase. From the results of clustering, we could see that Hierarchy confused between acrylic and steel vase, K-means confused between car sponge and kitchen sponge when Manhattan distance metric applied. But Hierarchy can distinguish car sponge and kitchen sponge clearly, also K-means can distinguish acrylic and steel vase very well. So, to some extent, the results of clusters are similar with real-life experience but sometime clusters perform better than human.

By changing distance metric, we found that Manhattan in Hierarchy performs similar with Euclidean in Hierarchy, but Euclidean in K-means performs much better than Manhattan in K-means. In conclusion, we think Euclidean distance metric is better than Manhattan distance metric in clustering.

Except for changing distance metric, we also tried using zero-centered data instead of standardised data, the results are shown in Figure 17, Figure 18, Figure 22 and Figure 20. By comparing the results using zero-centered data with the results using standardised data, we found that in K-means, the results using zero-centered data show a better performance. The reason is that in standardised data, the difference in numerical value between different object is weakened, so it is hard for K-means to make a distinction between objects.

Part 2 The suitable tree number is 30. As shown in Figure 23, we first generate 100 trees for selecting an appropriate tree number. The out of bag error turns into the stable situation after ensemble tree number reaches to 30, the fluctuation of error varies from 0.2 to 0.3 which is a tolerable range for prediction. Also, when it comes to the computational convenience, too much ensemble trees will be time-consuming, the 30 trees are sufficient.

We choose first two decision trees for discussion shown in Figure 25 and Figure 26. Because each tree is grown on an independently drawn bootstrap replica of input data [2], each variables such as x_1, x_2 and outcomes can be generated repeatedly. For theses two trees, the same class data can have the same variables with different attributes. Therefore each class range in different variables is defined from each decision tree.

Figure 28 shows the confusion matrix with standardised data. All the correct predictions are on the diagonal of the matrix. The overall accuracy in this case is 91.67$\%$.

According to the object properties, the misclassification is emerged when different classes have very similar value in the same property, naive classification method usually group a number of different classes into one. Thus, the prediction can easily make wrong decisions. If PCA step is not used in bagging, the confusion matrix shown in Figure 28 indicates that the accuracy of predicting objects is much lower which is only 63%. The comparison of accuracy shows that the PCA step is helpful and improve the overall prediction accuracy when making the original 19 dimensional electrodes to three dimensions.

5. Section E: Conclusion

Part 1 In real-life, when we want to describe an object, texture, size, color, hardness, etc properties are usually to be used. But when we try to classify different objects using computer, it usually cannot get the data of all these properties. Also, the features of an object are usually abstract in our mind, for instance, we cannot tell the exact temperature of the surface of a kitchen sponge but we know it should be soft. For computer, it is easier to tell computer the exact temperature data and voltage of a tactile sensor but it is hard to let computer know what 'soft' is. With pattern recognition techniques, we could let computer get connections between numerical data and objects. Also, we could extract features from data by pattern recognition techniques which could make computer classify different objects by the features.

During our work, we applied PCA, LDA and three clustering algorithms to see how their perform. For PCA, we used covariance matrix, eigenvalues and eigenvectors to extract features of the data. With these features, we could know how different kinds of our data separated with largest variance from a numeric point of view. For LDA, with supervised learning, we could separate our data with largest separation. And for clustering algorithm, with different distance metric and different training method, we could know about cluster centriod of different kinds of objects in K-means algorithm, how different kinds of data varied from each other in Hierarchy and bagging algorithm.

Part 2 From findings above, we would say it is unlikely to distinguish all different objects only using touch. There are six objects in our dataset, and we cannot make sure these objects can be clearly distinguished from each other with all the algorithms above. In our daily life, thousands of kinds of objects are needed to be classified, and most likely there are two different objects would have similar PVT and electrode data. Then with similar data, we think it is unlikely to distinguish between this kind of two objects.

However, if the task of the sensor is to distinguish some objects whose properties varies a lot between each other, we think it is possible to distinguish them use only touch. For instance, in our supervised algorithm LDA, it is robust to distinguish between acrylic and black foam shown in Figure 13 and Figure 14.

In conclude, we think that whether we could distinguish object use only touch is determined by the task.

Part 3 We think pressure is the most important property for sensor to measure. From PVT separately shown in Figure 29, Figure 30, Figure 31 and our findings, the pressure data separates wider which is from 950 to 1400 and the overlap is smaller than in vibration and in temperature which means the differences between each object are obvious, thus according to pressure property, objects are easy to be classified [6].

Besides, temperature and vibration are also important tactical data for classification. In some of our attempts, temperature and vibration show more clear separation on raw data than pressure. We cannot deny that they are also important properties although in our work pressure shows a overwhelmingly advantage on classification.

Part 4 Instead of choosing only one time step, we could extract data every few time steps in batches and create a new list of data. Then, we could have more supported information in each trail, it is now able to get a gradient for each property and sort them with priority in each trail which could help us classify objects as a new property. With more useful data and properties in learning and training, we could have a more accurate classification result when predict. This is a kind of dynamic feature learning, other than the features of single time step, the change of properties such as pressure and temperature also give us useful features.

The drawback is that with more data, it is time-consuming on processing the preliminary result and making a final decision. Also, more classification approaches for new properties could be taken, which will cause various modalities for each results and should be modulated into one general output form.

References

[1] H. Abdi and L. J. Williams. Principal component analysis. *Wiley interdisciplinary reviews: computational statistics*, 2(4):433–459, 2010.

[2] E. Bauer and R. Kohavi. An empirical comparison of voting classification algorithms: Bagging, boosting, and variants. *Machine learning*, 36(1):105–139, 1999.

[3] V. Chu, I. McMahon, L. Riano, C. G. McDonald, Q. He, J. M. Perez-Tejada, M. Arrigo, N. Fitter, J. C. Nappo, T. Darrell, et al. Using robotic exploratory procedures to learn the meaning of haptic adjectives. In *2013 IEEE International Conference on Robotics and Automation*, pages 3048–3055. IEEE, 2013.

[4] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. PMLR, 2015.

[5] I. T. Jolliffe. Graphical representation of data using principal components. *Principal component analysis*, pages 78–110, 2002.

[6] S. Yim, S. Jeon, and S. Choi. Data-driven haptic modeling and rendering of deformable objects including sliding friction. In *2015 IEEE World Haptics Conference (WHC)*, pages 305–312. IEEE, 2015.

6. Appendix

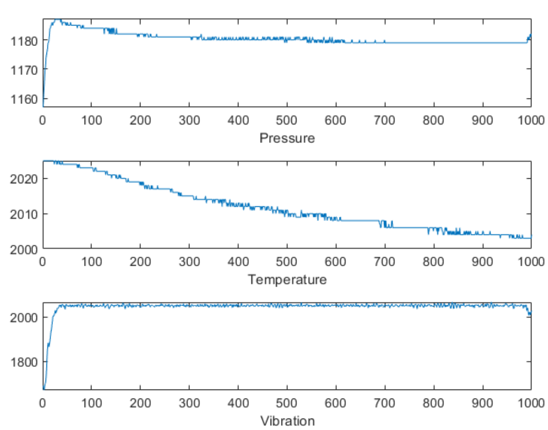

Figure 1. PVT data from $acrylic\_211\_04\_HOLD.mat$

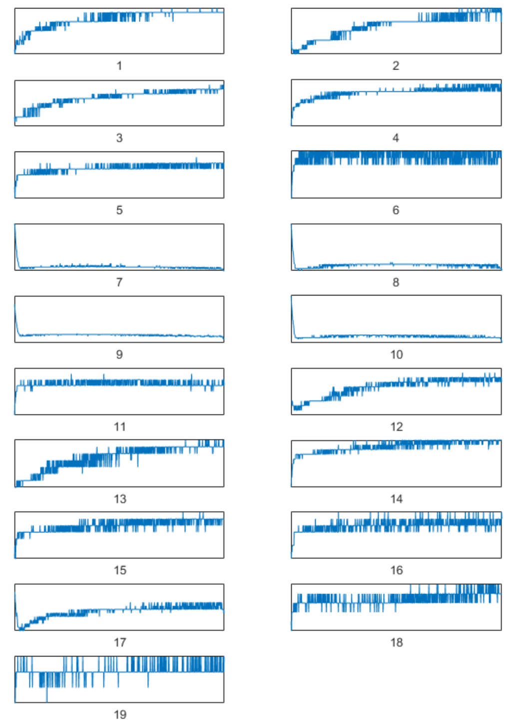

Figure 2. electrodes data from $acrylic\_211\_04\_HOLD.mat$

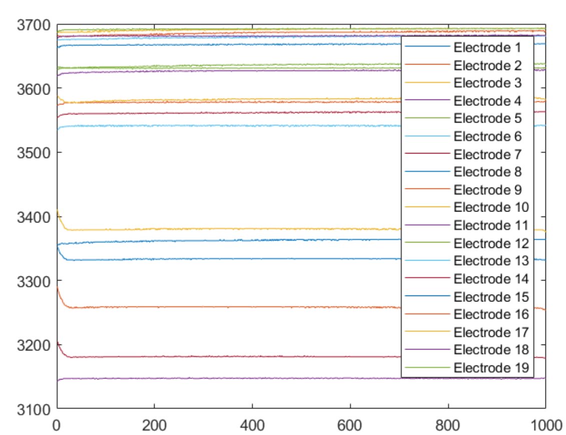

Figure 3. Re-plot electrodes data from $acrylic\_211\_04\_HOLD.mat$

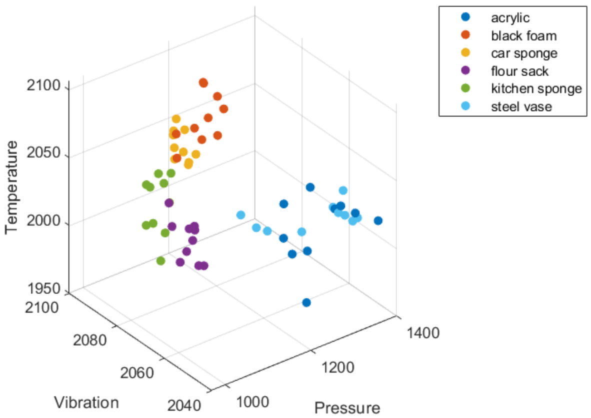

Figure 4. PVT data 3D scatter plot

Object	Hexadecimal code for color	Color
acrylic	#0072BD	
black foam	#D95319	
car sponge	#EDB120	
flour sack	#7E2F8E	
kitchen sponge	#77AC30	
steel vase	#4DBEEE	

Table 1. Color of objects

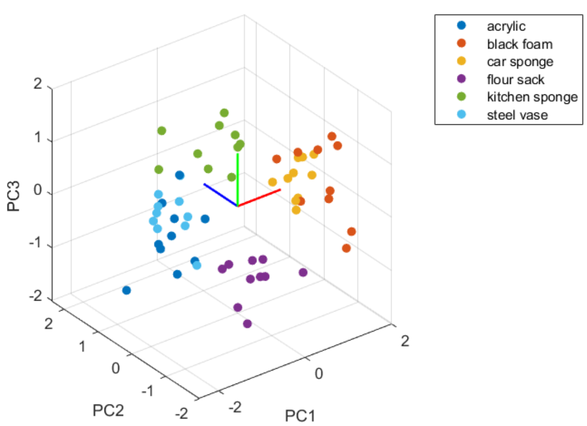

Figure 5. 3D standardised data with principle component analysis

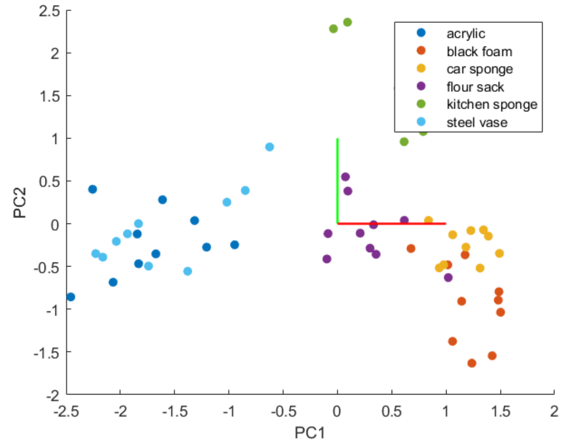

Figure 6. 2D standardised data with principle component analysis

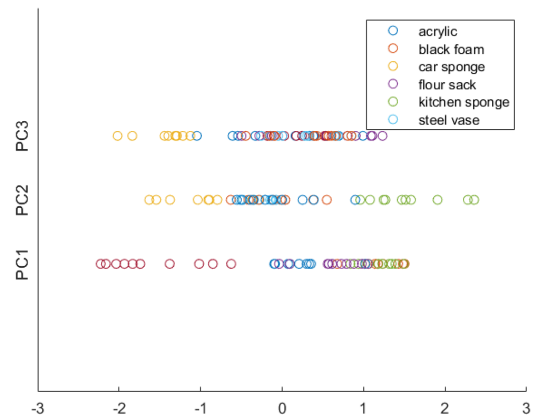

Figure 7. 1D number lines with principle component analysis (standardised data)

Figure 8. 1D number lines with principle component analysis using zero-centred data

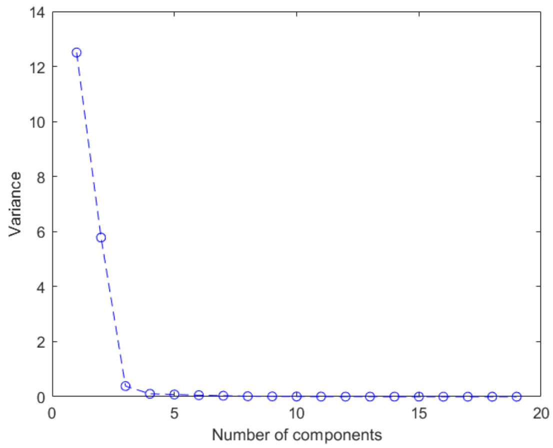

Figure 9. Scree plot with variance versus principal component numbers

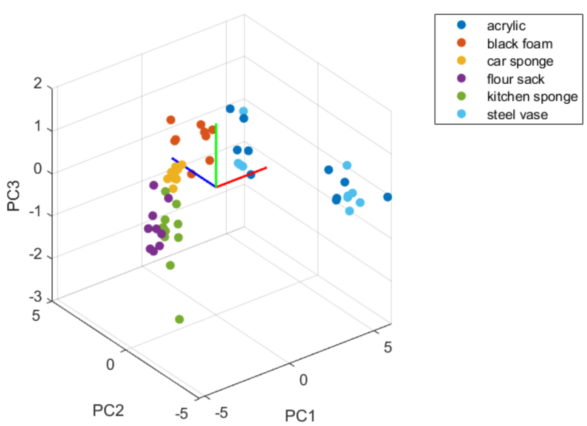

Figure 10. Electrode data with three principle components

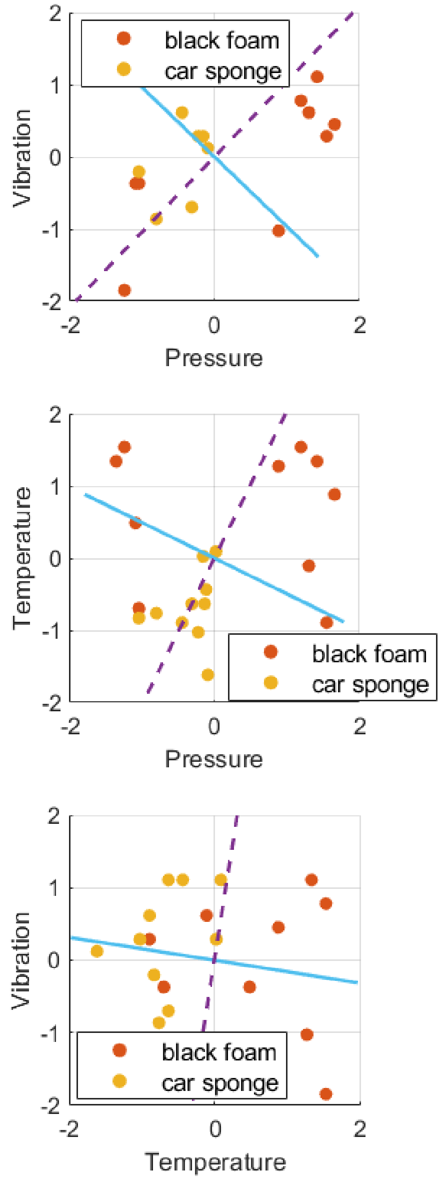

Figure 11. Spilted data by LDA line

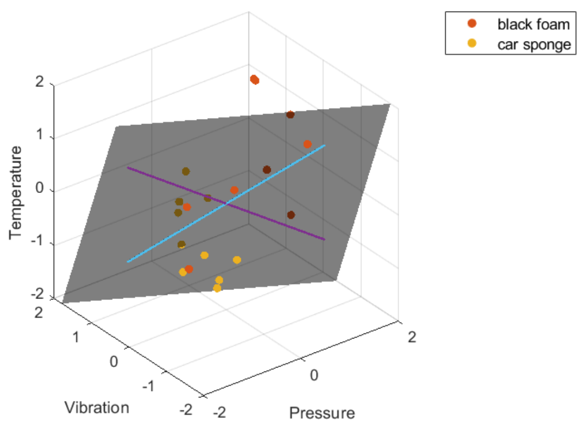

Figure 12. Spilted data by LDA line P-V-T and hyperplane

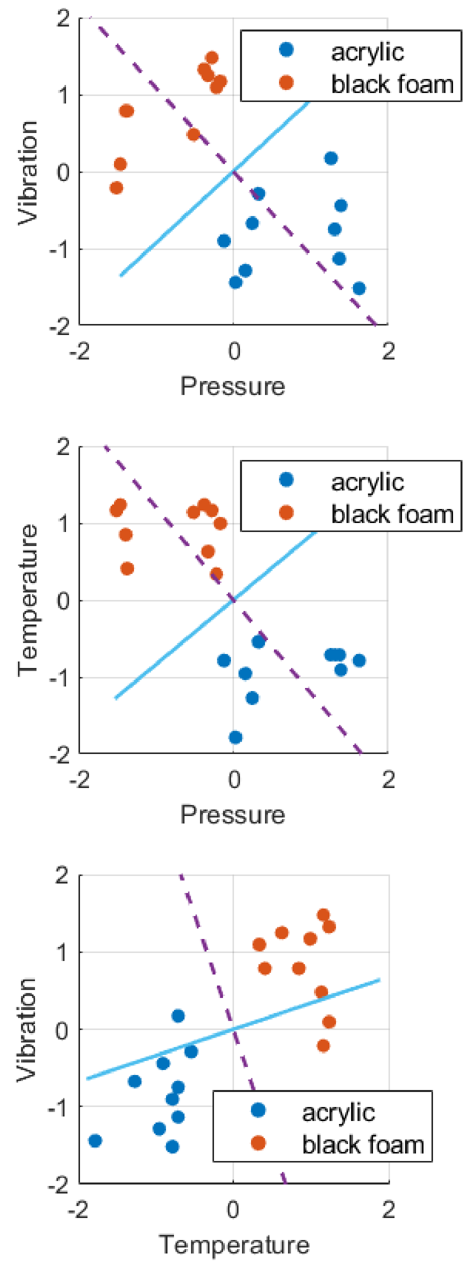

Figure 13. Spilted data by LDA line using acrylic and black foam

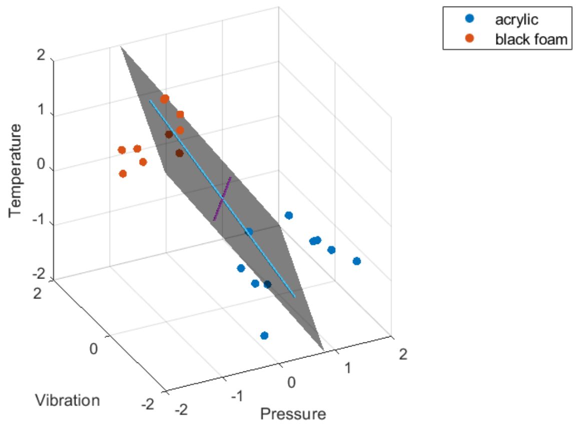

Figure 14. Spilted data by LDA line P-V-T and hyperplane using acrylic and black foam

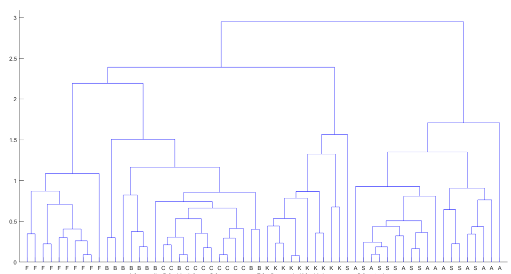

Figure 15. Hierarchical Clustering with Euclidean using standardised data

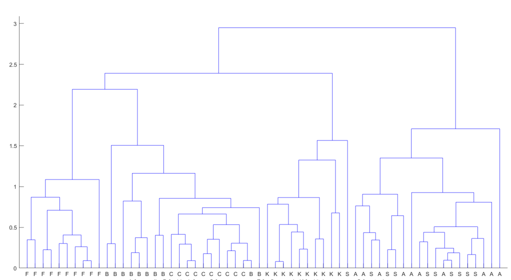

Figure 16. Hierarchical Clustering with Manhattan using standardised data

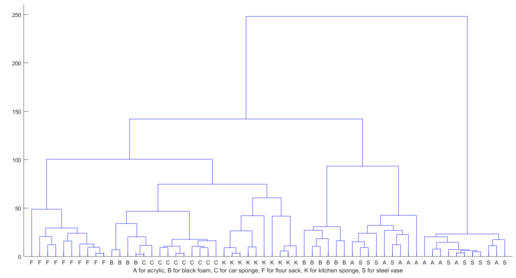

Figure 17. Hierarchical Clustering with Euclidean using zero-centered data

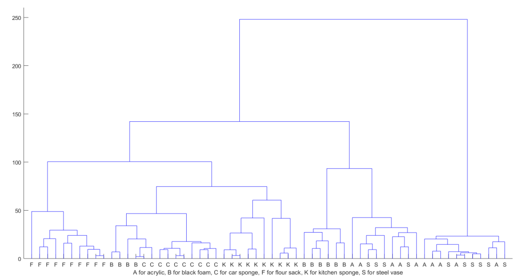

Figure 18. Hierarchical Clustering with Manhattan using zero-centered data

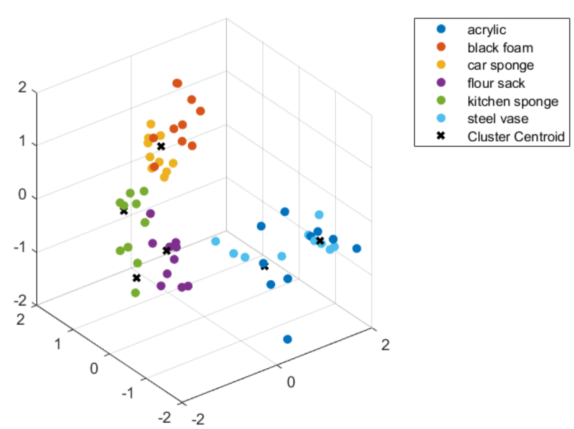

Figure 19. K-Means Clustering with Manhattan using standardised data

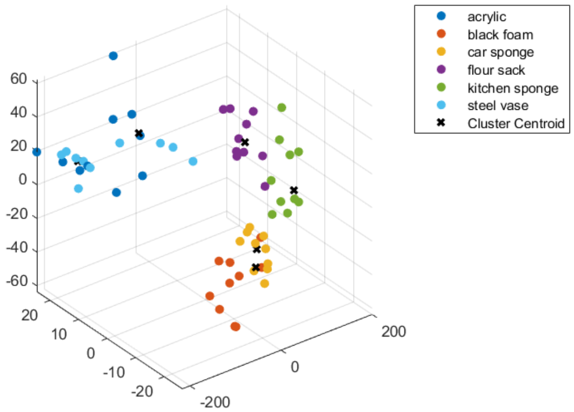

Figure 20. K-Means Clustering with Manhattan using centered data

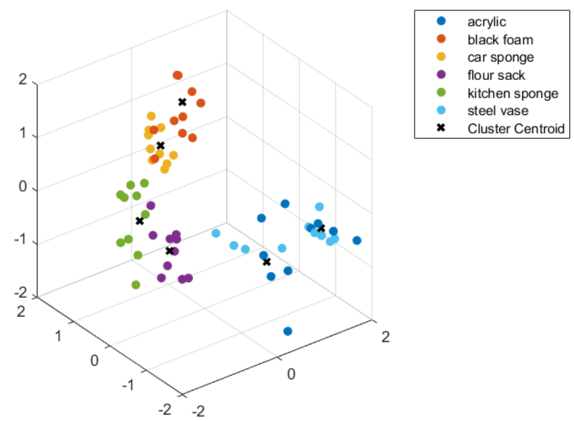

Figure 21. K-Means Clustering with Euclidean using standardised data

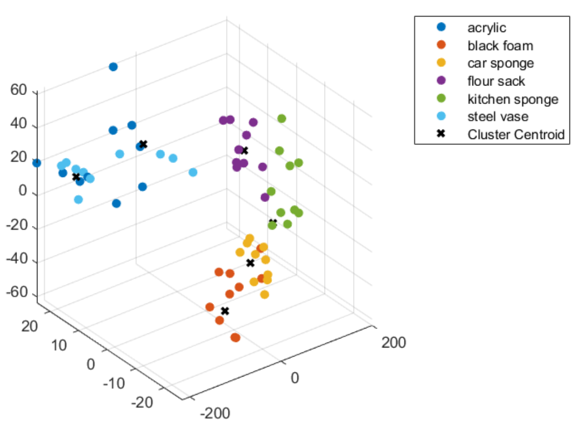

Figure 22. K-Means Clustering with Euclidean using centered data

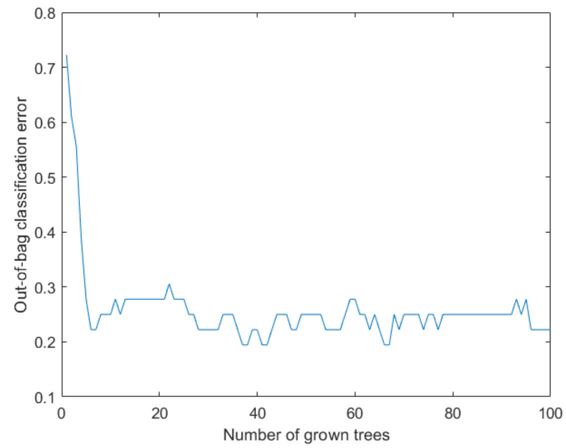

Figure 23. Out-of-bag error vs. trees grown with standardised data

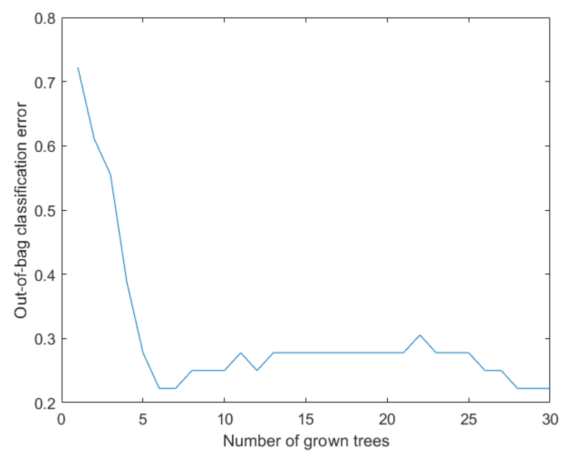

Figure 24. Out-of-bag error vs. trees grown with standardised data

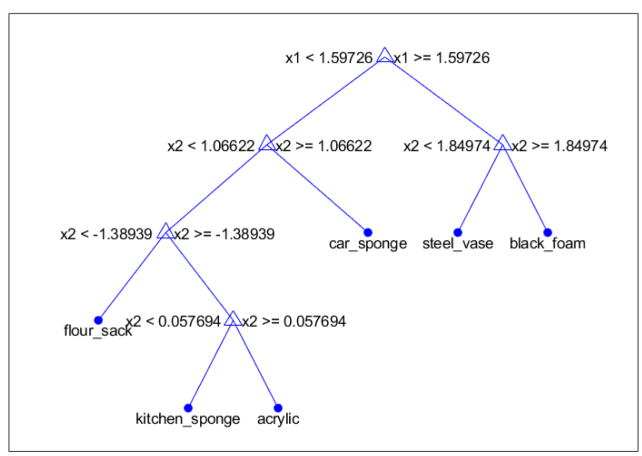

Figure 25. Grown tree 1 using standardised data

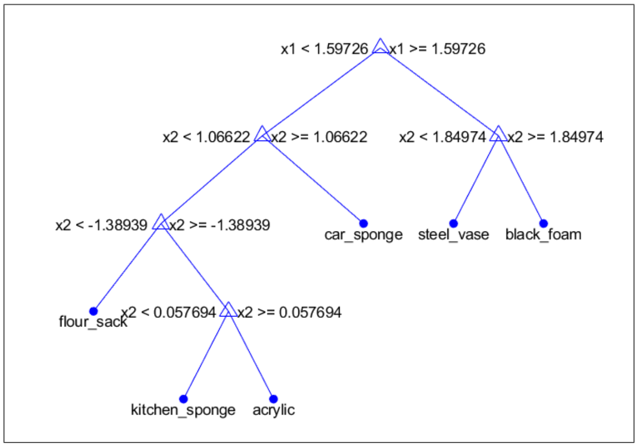

Figure 26. Grown tree 1 using standardised data

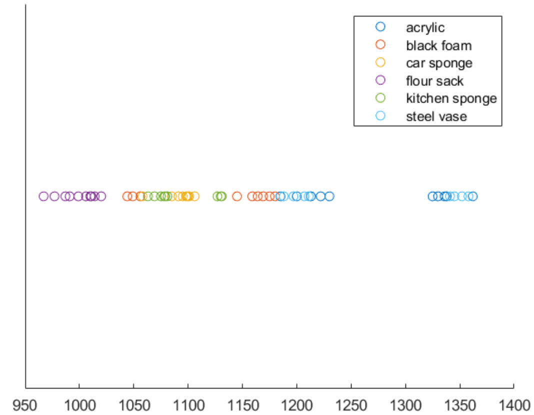

Figure 29. Pressure data

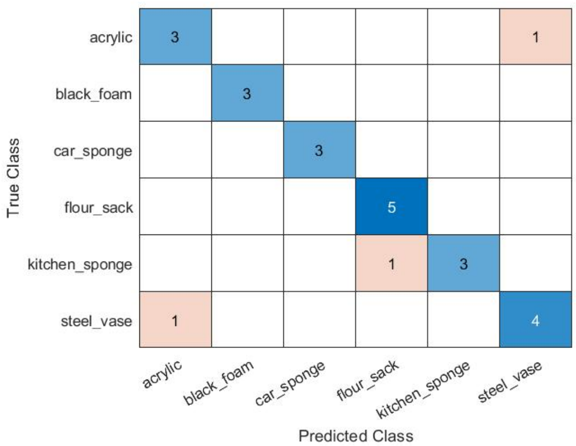

Figure 27. Confusion matrix with standardised data

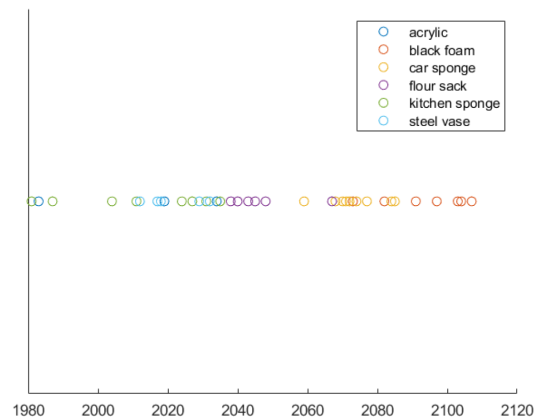

Figure 30. Vibration data

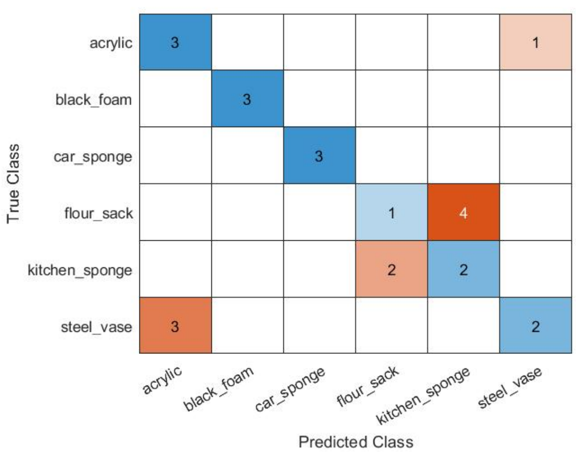

Figure 28. Confusion matrix without PCA

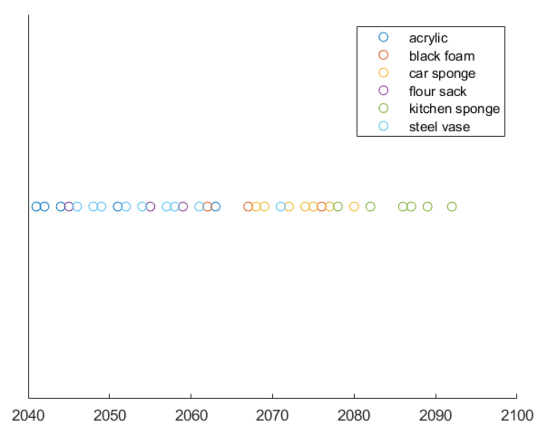

Figure 31. Temperature data