Pattern Recognition Coursework: Sebastian Steiner (ss16318 - 01516179), Kevin Wang (kw4818 - 01522932)

1. Introduction

Computerised object recognition is typically performed with computer vision algorithms; however, an alternative approach is through the interpretation of robotic touch. The accuracy of this approach will be tested using data collected from a PR2 robot with BioTac tactile sensors [1]. This equipment is expensive, costing around $500,000 in total [2] , therefore analysis will also extend to determining redundant sensors that could be removed, reducing the cost of the expensive experimental equipment.

2. Data Preparation

Tactile readings during the ”hold” phase of the 2 finger grasp were analysed for a collection of 6 objects: acrylic, black foam, a car sponge, a flour sack, a kitchen sponge and a steel vase, to see the hold phase of a grasp click here and watch from 16-18s. Each object was tested 10 times, so there were 60 grasp readings. Each measured a large amount of tactile data; however, the authors chose to perform analysis on a reduced dataset which included pressure (P), vibration (V), temperature (T) and impedance data from 19 electrodes, see Appendix A for data visualisation.

By only selecting data from a single timestep and finger, the dataset size was further reduced. An algorithm was deployed to determine this optimal timestep and finger. It calculated the weighted standard deviation of readings within each of the six object classes at each time instance. Equal weights were assigned to the electrode impedance, P, V, and T measurements. After normalizing each measurement for each finger, candidate timesteps were determined using the smallest intra-class standard deviation for six objects and two fingers. The specific timestep and finger was determined by choosing the candidate timestep with the largest inter-class standard deviation. In short, the algorithm aimed to find the timestep that minimized within class variation and maximized cross class variation, Figure 1 shows the resulting PVT data.

3. Principal Component Analysis (PCA)

PCA is a data-driven method which reprojects data along a set of axes which maximise the variance of the dataset. In doing so, the dimensionality of data may be reduced. Principal components (PCs) of the PVT were obtained using the eigenmodes from the covariance matrix of the standardised PVT data, see Appendix B for details and results.

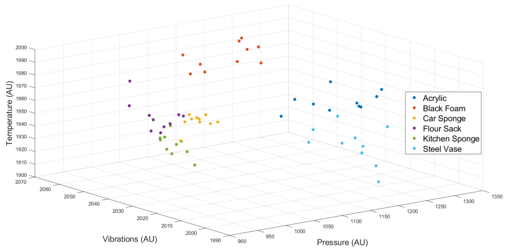

Figure 1. Raw PVT data chosen at timestep 62 from finger 1.

By reprojecting the PVT data along the first 2 PCs, the dimensionality of the data was reduced, shown in Figure 2. The data was further reduced to 1D using each PC, shown in Figure 3. As expected, the range of variation and class separation decrease with increasing PC number. Components 1 and 2 were better at separating different classes; for example component 1 successfully separates the flour sack, while component 2 separates the black foam better. As the components were able to separate classes in different ways, Figure 2 was able to show clearer groupings of object classes compared to the 3D raw PVT data. It was also noticed that each eigenvector (shown in Appendix B) predominantly pointed in the direction of a feature. The first PC pointed in the direction of pressure and the second of temperature. Therefore, these sensors should be considered more valuable for object recognition than vibration.

The electrode dataset had 19 dimensions, as 19 electrodes took measurements, so PCA was used to reduce dimensionality. The scree plot in Figure 4 shows that the first 3 PCs capture most of the dataset’s variation. Therefore, the electrode data was reprojected along the axes of the first 3 PCs, as shown in Figure 10 of Appendix B. Eigenvectors have unit length, so by squaring each eigenvector (element-wise), multiplying by its eigenvalue and summing across the weighted eigenvectors, the electrodes which captured most the data variation were determined, see Figure 5. These electrodes are the most important because by capturing data variation, they also capture class separation. From this calculation, electrodes 1 to 7 were determined to be the most important, while the other electrodes should consider being stripped. In fact, it is possible that reduction could extend to stripping electrodes 4 to 7 as they only capture 1000th the amount of information as electrode 1.

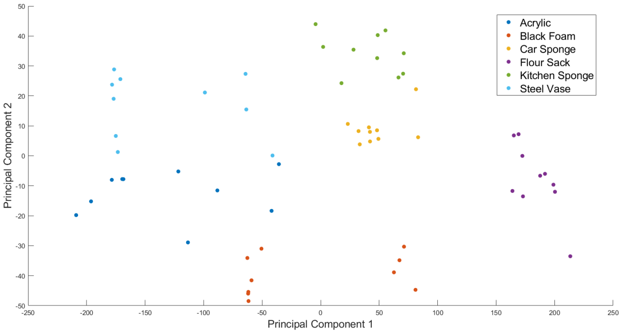

Figure 2. PVT data reduced to 2D by reprojecting data using first 2 PCs.

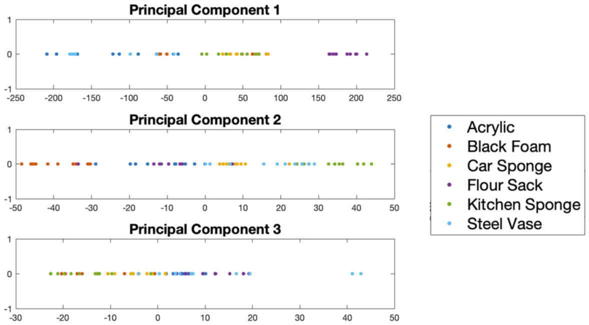

Figure 3. PVT data reduced to 1D by reprojecting data using each PC.

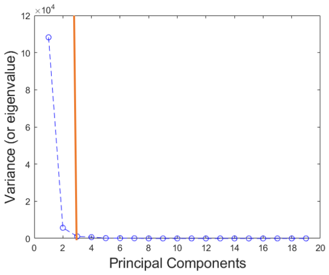

Figure 4. Scree Plot showing the variance captured by each PC.

4. Linear Discriminant Analysis (LDA)

LDA aims to find linear relationships between features that maximise class separation on a dataset. This technique was implemented on 2 pairs of similar objects to identify features in the PVT data that maximise separation. The first object pair was the black foam and the car sponge, which have similar deformable and porous properties. The second pair was the steel vase and acrylic, which are rigid objects and do not have obvious separation. Figure 6 shows that temperature was the most important attribute to maximise class separability for the first object pair; the same result was found for the second object pair as shown in Figure 12 in Appendix C. When LDA was performed on the 3D PVT data, as shown in Figures 13 and 14 , it was again found for both object pairs that temperature was the most feature. Figure 6 also shows the object pairs were replotted against this temperature correlated LDA line and there is clear separation of classes.

The LDA results indicate temperature is solely required for class separation. However, in both experiments objects had similar stiffness properties. As the pressure and vibration measurements depend on stiffness, it should be expected that these features were unable to separate classes. Therefore, it should be considered a tool for distinguishing objects with very similar stiffness properties. Objects pairs with different stiffnesses should be analysed; for example, LDA should be performed with a car sponge and a steel vase. If the results remain consistent, an investigation could be opened into temperature as a sole object recognition feature.

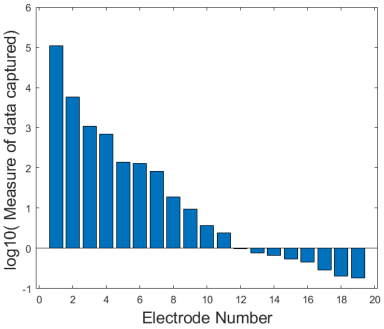

Figure 5. Logarithm of data variation measured by different electrodes, see Appendix B for calculation.

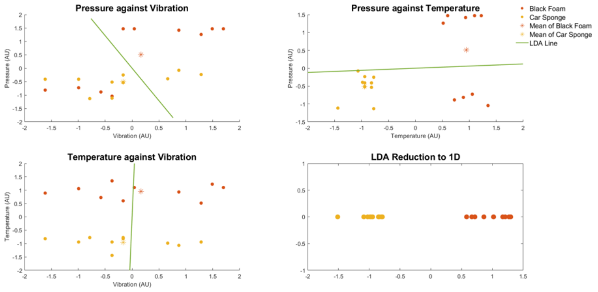

Figure 6. LDA lines plotted against all three combinations of P, V and T features. 1D plot of LDA line with largest eigenvalue.

5. Classification & Clustering

Clustering To determine whether the PVT data collected from the robot could be automatically classified, unsupervised clustering algorithms were implemented to group the data. K-means and hierarchical clustering were tested using euclidean and city block distance measures, see Appendix D for details. Before clustering, the PVT data was normalised to remove dimensionality bias. However, the results from PCA found that PCs were essentially parallel to feature axes, hence dimensionality should be biased for classification. So, k-means clustering was also performed on the data with zero-mean but unnormalised variance.

From Table 1 , it is clear that k-means with unnormalised variance yields the best classification results, confirming that dimensionality bias was important for clustering in this case, as suggested by PCA. Moreover, k-means performs slightly better than hierarchical clustering. The distance metric minimally effects the classification accuracy, so city block metric was preferred as it is less computationally expensive.

Clustering Method	**Distance Metric**	**Accuracy**
Hierarchical	Euclidean	58%
Hierarchical	City Block	62%
k-means	Euclidean	63%
k-means	City Block	63%
k-means (unnormalised)	Euclidean	77%
k-means (unnormalised)	City Block	73%

Table 1. Classification accuracy of PVT data clustering. In Appendix D, see Figure 15 for graphical representation of clustering and classification accuracy calculation.

Bootstrap Aggregation for an ensemble of decision trees A bagging algorithm was built to perform supervised classification, with the expectation that by labelling the training data, the classification accuracy would increase. To assess this hypothesis, the electrode data, which was reduced to 3 dimensions in section 3, was split into testing and training, with a ratio of 40:60, respectively . The bagging algorithm used the labelled training data to create decision trees for classification, as shown in Appendix D. The number of decision trees was user specified and chosen to be 8, as more decision trees would not significantly reduce the out-of-bag error and would increase computational costs, see Figure 7. The accuracy of the model was then determined using the test data by comparing the models predicted classifications to the real classifications. The model was 98.6% accurate, see the confusion matrix in Appendix D. This proves that despite reducing dimensioanlity, PCA preserves the structure of electrode data. It was also found that 8 decision trees were sufficient and that supervised classification performs significantly better than unsupervised. Furthermore, this result demonstrates that it is possible for robotic touch to correctly recognise objects and that only electrode data is required for object recognition.

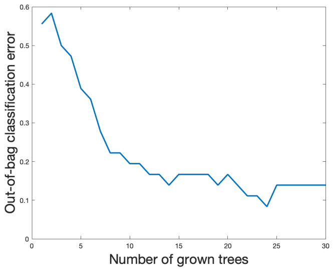

Figure 7. Out-of-bag error plotted against number of decision trees. Out-of-bag error measures the misclassification for a given set of decision trees.

6. Conclusion

Data Preparation was successful as the selection of timestep and finger permitted subsequent analytical tasks to separate classes and recognise objects. However, alternative methods to minimising within class variation and maximising cross class variation should be explored. For example, by calculating the feature average during periods of low change, more indicative features of objects could be extracted. Further work could also extend to analysing changes in data and using dynamic time warping to capture time varying features that cannot be extracted from a single timestep, see Appendix A.

PCA proved to be a useful technique to reduce dimensionality, especially for the electrode data, which was reduced from 19 dimensions to 3 dimensions. The data reduction did not affect classification as supervised classification was 98.6% accurate using the 3 dimensions, indicating that fewer sensors could be used for robotic touch classification. Further work should include repeating all electrode experiments, only using electrodes 1 to 7 (or even 1 to 4), which have been identified as the most important electrodes.

LDA revealed that for object pairs with similar deformable properties, temperature was the most important feature for class separation. However, LDA should also be performed on objects with different stiffnesses to confirm that temperature is the most important class separating attribute. PCA found pressure to be the most important feature and Figure 2 essentially plots of T against P, see Appendix B for eigenvectors directions, with resolvable class clustering. Therefore, the stripping of vibration sensors should be considered.

Classification was performed using unsupervised clustering and supervised bagging. The classification accuracy ranged between 58%-77% for clustering. Indicating that if the goal is to automatically classify robotic touch, more features must be measured by the robot. However, if labelling all training data is a viable option, bagging has proven to be 98.6% accurate, so successful object recognition was possible using only touch. Hence, further work into sensor stripping should be carried out to reduce the cost of the robot.

References

[1] 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.

[2] A. Spiers. Pattern recognition coursework. 2021.

7. Appendices

7.1. Appendix A: Data Preparation

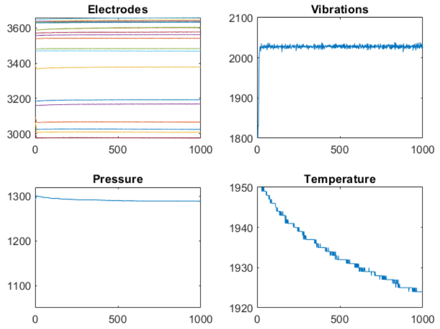

Figure 8. Raw PVT and Electrode data plotted against time for the 5th acrylic grasp trial.

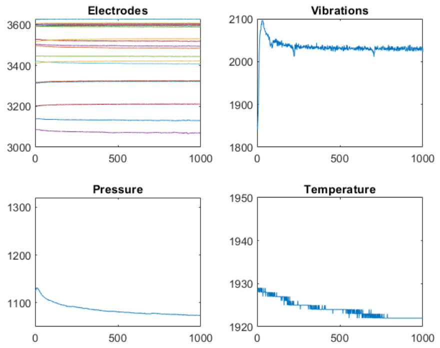

Figure 9. Raw PVT and Electrode data plotted against time for the 5th kitchen sponge grasp trial.

Figure 8 and Figure 9 shows raw data examples that were reduced during data preparation. During this process, only P, V, T and electrode data was analysed; however, trends in other features such as temperature and pressure change or robot joint effort should also analysed. In addition, dynamic time warping could be performed to align data trajectories of the same attribute over 10 trials of the same object class. The process makes the trajectories comparable with each other. It would strengthen characteristics from object classes while making any random noise less influential for analysis. This would, however, increase dimensionality of data preparation potentially making the analysis more difficult.

7.2. Appendix B: PCA

PCA calculation on PVT data To perform PCA on the data, the following steps were taken (note the same steps were taken to perform PCA on the electrode data):

- Standardise the data so that the average of each feature (P, V and T) had zero-mean.
- Calculate the covariance matrix of the standardised data, see below:

$$CovarianceMatrix = \begin{pmatrix} 1.43e4 & -317 & -728 \\ -317 & 195 & -14.3 \\ -728 & -14 & 665 \end{pmatrix}$$

- Perform single value decomposition on the covariance matrix to get the eigenvalues and eigenvectors, see below:

$$Eigenvalues = \begin{pmatrix} 1.43e4 & 0 & 0 \\ 0 & 628 & 0 \\ 0 & 0 & 185 \end{pmatrix}$$

$$Eigenvectors = \begin{pmatrix} -0.998 & -0.052 & -0.026 \\ 0.022 & 0.071 & -0.997 \\ 0.053 & -0.996 & -0.069 \end{pmatrix}$$

Figure 10 shows the eigenvectors plotted as principal components with the standardised PVT data. Interestingly, the eigenvectors predominantly point in directions of the P, V and T features. This can be explained as the off diagonal elements in the covariance matrix are small relative to their diagonal counterparts, indicating weak correlations between the P,V and T features.

The first 2 eigenvectors were used to plot the 2D PVT reduction plot in Figure 2. Also, Figure 3 shows the data reprojected along each eigenvector. Figure 3 shows the replotted along all 3 eigenvectors.

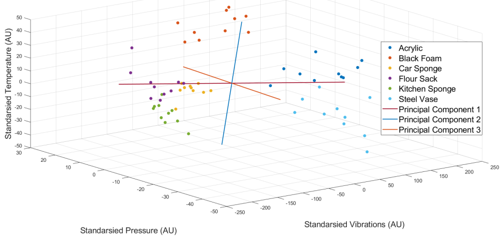

Figure 10. PVT data reprojected along PC axes.

Electrode Dimensionality Reduction Figure 11 shows the electrode data reduced from 19 dimensions to 3 dimensions. Considering the large dimensionality reduction, the resolution of class clusters is impressive.

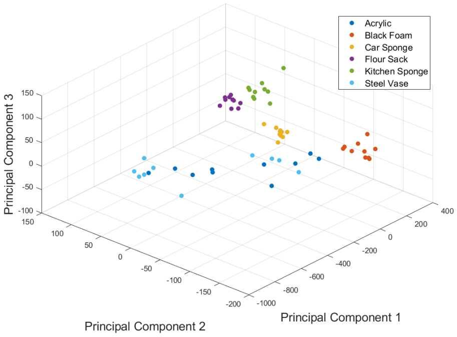

Figure 11. Electrode data reduced from 19 dimensions to 3 dimensions using the first 3 PCs.

Determining important electrodes from PCA results During PCA, the eigenvectors and corresponding eigenvalues are calculated. The eigenvectors describe the directions which maximise the variation of the data, while the eigenvalues describe how much of the variation each direction captures. Therefore, by multiplying each eigenvector by a loading equal to its eigenvalue, weighted eigenvectors are obtained. Summing these weighted eigenvectors together reveals which electrodes capture the most data variation, see equation below:

$$VariationMeasure(j) = \sum_{i=1}^{N} S(i) \cdot V(j,i)^2$$

where j represents the electrode of interest, S represents eigenvalues, V represents eigenvectors, i represents a particular eigenmode number and N represents the total number of eigenmodes. (Note that each element of each eigenvector is squared as only magnitude is of interest. Also, the sum of the squared eigenvector elements is always 1, as eigenvectors have unit length).

Figure 5 shows the logarithm of the variance captured by each electrode and from this it is clear that electrodes 1 to 4 capture the most data variation, so should be kept while removal of the other electrodes should be considered.

7.3. Appendix C: LDA

As mentioned, LDA maximises class separation. Figure 12 shows LDA performed on acrylic and steel vase yielding the same results as black foam and car sponge: temperautre was the most important feature. However, the 1D plot in the predominantly temperature direction shows smaller class separation, which was also seen during PCA for these objects, see Figure 10 and Figure 11. Figure 13 and Figure 14 show the LDA lines plotted in 3D with the PVT data. The LDA 1 line found in these plots were used during reprojection to 1D in Figure 6 and Figure 12. Interestingly, Figure 13 does not have a second LDA line because the eigenvalue of the second LDA line is -2.2 e-16, which should not be possible as eigenvalues from LDA cannot be negative. However, the e-16 shows this eigenvalue is essentially zero, indicating that it is negative due to computer precision error rather than implementation error. Therefore, the second eigenvalue has been set to zero and consequently its eigenvector was not plotted in Figure 13. This result strengthens the claim that temperature is the most important differentiating feature for objects with similar stiffness properties.

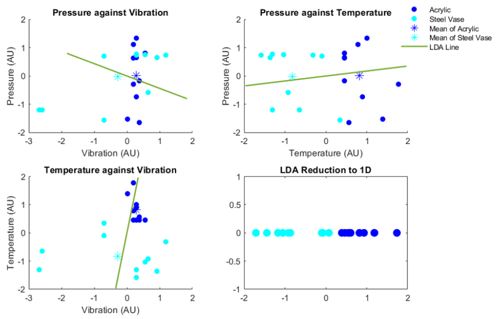

Figure 12. LDA lines plotted against all three combinations of P, V and T features. 1D plot of LDA line with largest eigenvalue (this line predominantly points in the temperature direction.

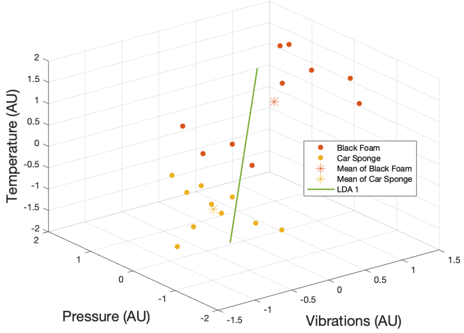

Figure 13. LDA lines from 3D PVT data for black foam and car sponge. Note only one LDA is shown in the temperautre direction as the eigenvalues corresponding to the second LDA line was set to zero.

7.4. Appendix D: Classification and Clustering

Clustering The raw PVT data, shown in Figure 1, was stored in an array with 60 rows, where every 10 rows repre-

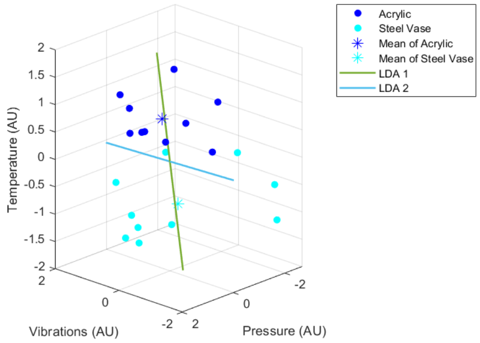

Figure 14. LDA lines from 3D PVT data for acrylic and steel vase.

sented 1 object. So in Figure 15, the class of each object was also stored in an array of 60 rows; however, the class numbers were not correlated to the raw data. Therefore, rather than comparing cluster colours in Figure 15 to Figure 1, groups of clusters should be compared. This concept was applied when measuring the classification accuracy whose procedure was as follows:

1. For every 10 rows in the class labelling array of the clustering algorithm, count the frequency of each class.

2. For each of the 10 rows propose the class that appears the most.

3. If all of the proposed classes are different, move to step 4. If the same class is proposed multiple times, from the 10 rows with the lower proposed class frequency re-propose the second most common class. Continue this process until all proposed classes are different. (Note that if the proposed classes are the same and their frequencies are the same, change the proposed class of the 10 rows with the higher second most common frequency).

4. Sum the frequencies of the proposed classes. Divide this number by 60 and multiply by 100 to get the classification accuracy.

Bagging Figure 16 and Figure 17 shows example decision trees, which were generated by the bagging the algorithm. x1, x2 and x3 represent the reprojected electrode data along PCs 1, 2, and 3, respectively. The values of the x's were compared to a threshold at each node and the result determined which branch of the tree to follow. Note that the branches always end with a class classification decision and that the depth of the decision tree may vary.

Below shows a confusion matrix where each row represents true class and each column represents predicted class. Counts along diagonals along represent correct predictions, counts elsewhere represent incorrect predictions. The sum of all elements in the confusion matrix equals the number of test data points. 2 out of 3 test runs yielded confusion matrices showing perfect classification. As each test run contained 24 trials, 1 in 72 classifications was incorrect mean-

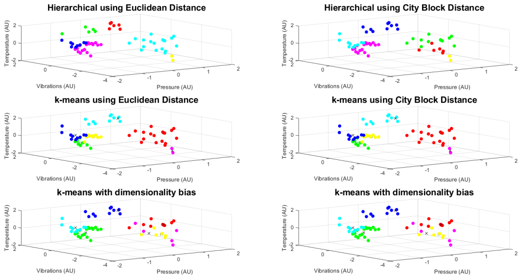

Figure 15. Resulting clusters using hierarchical, k-means clustering and using different distance metrics: euclidean and city block. Note the 'x' in the k-means plots are cluster centroids. It is worth mentioning that re-running the clustering part of the code referenced in Appendix E will yield the same results for hierarchical clustering. However, this cannot be guaranteed for k-means clustering, as clusters are randomly initialised each experiment, so final results differ.

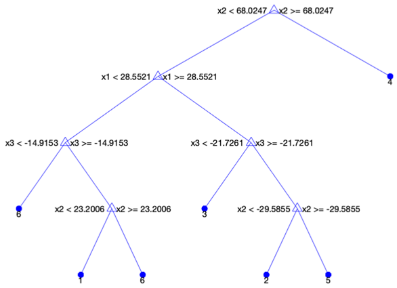

Figure 16. Example decision tree from bagging. x1, x2, x3 represents decision based off electrode data reprojection value along PC 1, PC 2 and PC 3, respectively.

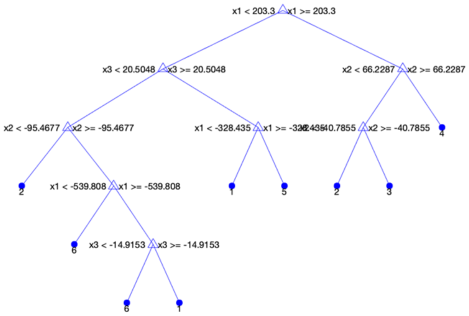

Figure 17. Example decision tree from bagging. x1, x2, x3 represents decision based off electrode data reprojection value along PC 1, PC 2 and PC 3, respectively.

ing the model was 98.6% accurate.

$$ConfusionMatrix = \begin{pmatrix} 4 & 0 & 0 & 0 & 0 & 0 \\ 0 & 4 & 0 & 0 & 0 & 0 \\ 0 & 0 & 3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 4 & 0 & 0 \\ 0 & 0 & 0 & 0 & 5 & 0 \\ 0 & 0 & 0 & 0 & 0 & 4 \end{pmatrix}$$

7.5. Appendix E: Code

Click here to access GitHub repository with the code used to complete the coursework.