

CSC8003 – Machine Learning Project Part II - Final Report

Performance Evaluation of the Depth of Anaesthesia Index Model

Name: Diana Kither

Table of Contents

1. Introduction	3
2. Performance Evaluation of the DoA Index Model	3
3. Discussion	7
4. Analysis for DoA Index Model Improvement	9
5. Cybersecurity for Well Hospital	11
6. Summary	12
7. References	14
8. Appendices	17

1. Introduction

Anaesthesiologists consider multiple parameters when monitoring the depth of anaesthesia in patients (Jiang et al 2015). Amongst these parameters are data from the electroencephalogram (EEG) and the Bispectral Index (BIS) (A-2000 BIS monitor; Aspect Medical Systems Inc, Newton MA) (Jiang et al 2015). EEG monitors the electrical activity of the brain making it a very useful tool for monitoring the effects of anaesthesia in a patient (Gu et al 2019). The BIS index is calculated with EEG data and in turn has a strong relationship with anaesthetic depth (Gu et al 2019; Lee et al, 2019, Ortolani et al 2002). Accurate assessment of anaesthetic depth is crucial for patient health and safety (Gu et al 2019). Recently the use of machine learning techniques to analyse EEG data and to determine an alternative measure of anaesthetic depth have been explored (Gu et al 2019; Lee et al, 2019).

The goal of the final project for CSC8003 Machine Learning was to develop a machine learning model to determine a depth of anaesthesia (DoA) index for patients under general anaesthesia. The model for this project was built with supervised learning techniques using the EEG data of ten patients under general anaesthesia. The BIS indices from these patients were used as the target variable for the model development. The BIS index measures in units between 0 and 100 as does the new DoA index. The final model constructed in the model building process combined two machine learning algorithms, a linear model and a neural network. In this report the performance of this new DoA index model is evaluated. The model is tested using the EEG data and BIS indices collected from another five test patients under general anaesthesia. Furthermore, on consideration of model performance data and further research, amendments were made to the model in an effort improve the performance of the new DoA index model.

2. Performance Evaluation of the DoA Index Model

The machine learning model for the new DoA index was built and tested used the MATLAB platform (Version R2020a, Mathworks 2020). The code for model testing is shown in Appendix A. The model was tested using the EEG data collected from five patients whilst under general anaesthesia. Similarly to the training data set, the test data set is comprised of five features, x_1 , x_2 , x_3 , x_4 and x_5 which represent the EEG data. The newly calculated DoA index was then compared to the BIS indices collected from each patient to evaluate the performance of the new model.

In accordance with the DoA index model, feature selection methods were applied selecting features x_2 , x_4 , and x_5 from the test patient data set to be utilised as input data for the first step of the model, the linear model. The calculated results from the linear model were subsequently used as input for the neural network and the new DoA index was calculated. Performance statistics were evaluated for each individual patient as well as a combined test patient data set including the data from all five patients.

The first step in evaluating model performance was to compare the new DoA index with the BIS index graphically. Graphs of the DoA index and BIS index plotted over time were created for each patient. The distribution of DoA and BIS values for patient 2 and patient 4 are shown in Figures 1 and 2 below. The distribution of DoA and BIS values for the remaining individual test patients are shown in Appendix B (Figures 5, 6 and 7).

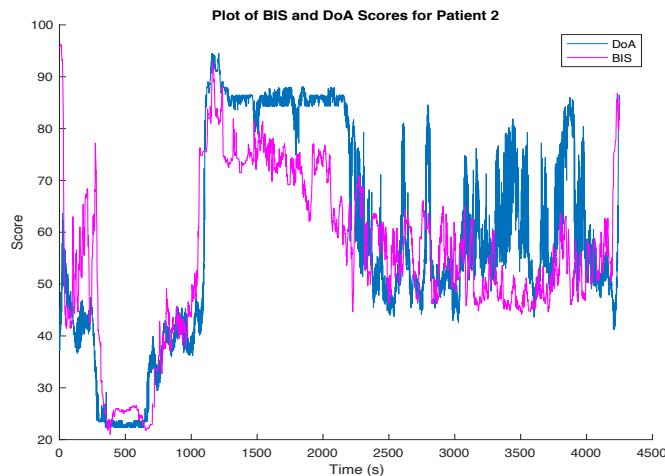


Figure 1 : Plot of BIS and DoA indices for test patient 2 over time (s)

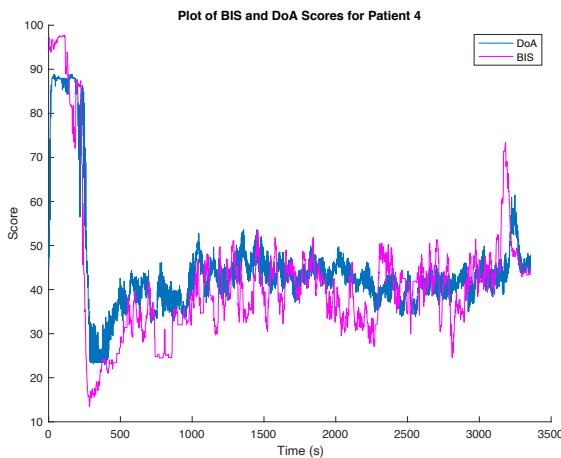


Figure 2 : Plot of BIS and DoA Indices for test patient 4 over time (s)

When examining the distributions of BIS and DoA indices for individual patients there is similarity evident between the two scores. There are however some larger differences between the two values that are also present. The greatest similarity between the two scores is evident in the distribution of test patient 4 where there is a clear association between the distribution of the DoA and BIS indices. In contrast, when examining the distribution of the DoA and BIS indices for test patient 2 similarity is evident however there is also a much more significant difference between the two values when compared to other patients. Test patients 1, 3 and 5 appear to have good similarity between the DoA and BIS indices with some variation.

The next step in assessing model performance was calculation and evaluation of test statistics to investigate the strength of the relationship between the new DoA index and the BIS index. The Pearson correlation coefficient (r) including a 95% confidence interval and the coefficient of determination (R^2) were calculated and the results are shown in Table 1 below. The test patient data was also combined into a single data set and the Pearson correlation coefficient was calculated as $r = 0.852$, $p < 0.05$ with a 95% confidence interval of 0.847-0.856. R^2 for the combined patient data set was 0.725.

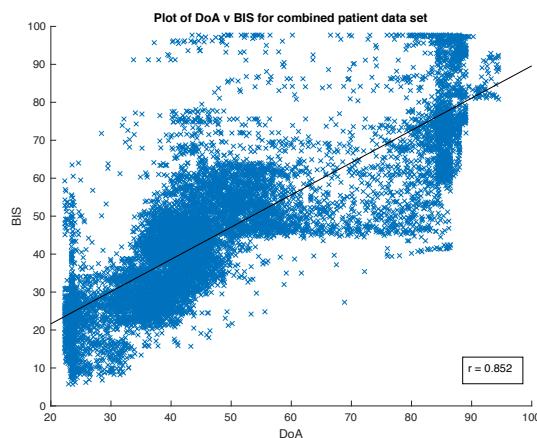
Table 1: The Pearson Correlation Coefficient and Coefficient of Determination for each test patient

	Pearson Correlation Coefficient ($p < 0.05$)	95% Confidence Interval for Correlation Coefficient	Coefficient of Determination (R^2)
Patient 1	0.880	(0.870 - 0.890)	0.774
Patient 2	0.767	(0.754 - 0.779)	0.588
Patient 3	0.909	(0.899 - 0.917)	0.826
Patient 4	0.846	(0.836 - 0.855)	0.715
Patient 5	0.879	(0.870 - 0.887)	0.772

The average Pearson correlation coefficient across the five test patients is 0.856 with a standard deviation of 0.049. The average value for R^2 across the five test patients is 0.735.

When looking at the correlation coefficients for each individual patient, all patients have moderate to high levels of correlation between the DoA index and BIS index, indicating that the model results appear to have good relationship with the BIS index. Patient 3 in particular had a very high correlation between the DoA and BIS indices at 0.909 with patient 2 having the lowest level of correlation at 0.767. On graphical examination of the distribution the DoA and BIS indices, patient 2 appeared to have the greatest difference between the two indices. Interestingly when examining the distribution of DoA and BIS indices over time, patient 4 appeared to have the greatest similarity between the two values of the all test patients. The correlation coefficient for patient 4 was high however it was slightly lower than three of the other test patients and the average across all patients.

The average correlation coefficient across the five patients was high at 0.856 again indicating that the new DoA index has a strong relationship with the BIS index. However when examining the data graphically with the BIS index plotted against DoA index it is evident that there is a degree of linear relationship between the two variables however there is still a wide deviation of some results from the regression line indicating a potentially significant degree of error in the model (Figure 3).

*Figure 3: Plot of DoA v BIS indices for the combined patient data set*

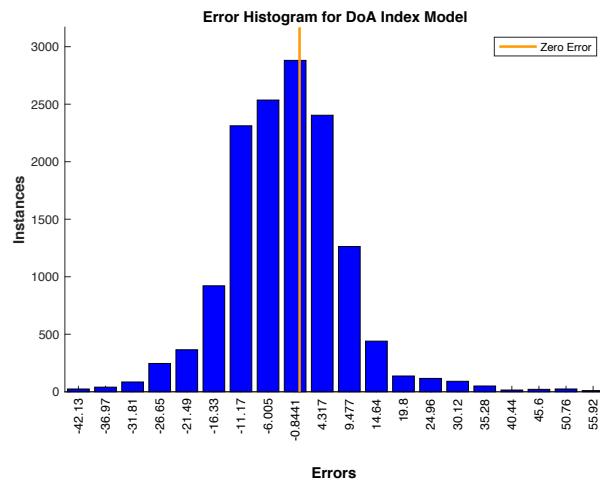
The coefficient of determination (R^2) describes the degree of variation in the DoA index that is influenced by the inputs in the DoA model. The average R^2 value across the five test patients was moderate to high at 0.735 indicating that 73.5% of the variation in the DoA index could be attributed to the feature inputs of the model. When looking at individual patients the model did not perform as well with some patients. Patient 2 had an R^2 value of 0.588 indicating only 58% of the variation in the DoA index could be explained by the model meanwhile patient 3 had an R^2 value of 0.826.

To further assess the relationship between the new DoA index and BIS index, three measures of error were calculated for each test patient, the mean absolute error (MAE), the mean squared error (MSE) and the root mean squared error (RMSE). The mean and standard deviation of each error value across the five patients was also calculated. The results are shown in Table 2 below. The combined patient data set had a MAE of 8.536, an MSE of 129.046 and an RMSE of 11.36.

Table 2: Mean Absolute Error (MAE), Mean Squared Error (MSE) and Root Mean Squared Error (RMSE) for the test patients

	MAE	MSE	RMSE
Patient 1	8.000	135.425	11.637
Patient 2	10.356	182.853	13.522
Patient 3	8.199	117.920	10.859
Patient 4	6.833	79.905	8.939
Patient 5	8.381	108.438	10.413
Mean ± SD	8.354 ± 1.138	124.908 ± 34.099	11.704 ± 1.507

A histogram was constructed to plot the frequency of the absolute errors of the model (Figure 4).



and 8.939 respectively. This is substantially less than patient 2 with the highest MSE of 182.853 and RMSE of 13.522. Given the square of the error is used to calculate MSE and RMSE both these error measurements are more sensitive to larger errors thus leading to a large result (Hiregoudar 2020). As seen in the error histogram, there are a small number of errors of up to approximately 50 index points which will increase the MSE and RMSE results. Again on an index scale of 0 to 100, an error of 50 is hugely significant and would result in a considerably different assessment of anaesthetic depth. The MSE and RMSE are considered appropriate measures to consider in this case given the consequences of major errors in the model predictions. Again when considering these results with those of other studies in the literature much lower error rates have been achieved when utilising a neural network model for assessment of depth of anaesthesia (Jiang et al 2015; Ortolani et al 2002)

The final step of performance assessment for the model was to assess the classification ability of the model for four classes of anaesthetic depth, awake (BIS 80-100), light anaesthesia (BIS 60-80), general anaesthesia (BIS 40-60) and deep anaesthesia (0-40). The sensitivity, specificity and accuracy of the new DoA index model were calculated and are displayed in Table 3 below. The confusion matrix of model classification results is shown in Appendix C, Figure 8.

Table 3: Classification performance parameters for the DoA index model for the combined test patient data set

Class of Anaesthesia	Sensitivity	Specificity	Accuracy
Awake (80-100)	81.26%	88.29%	87.84%
Light (60-80)	11.78%	94.90%	81.83%
General Anaesthesia (40-60)	66.37%	79.40%	74.78%
Deep (<40)	78.78%	87.18%	83.57%

The accuracy levels of the model are moderate to high with the lowest level of accuracy for the general anaesthesia class. The model has a very low sensitivity level for the light anaesthesia class indicating the model is poor at identifying patients that are lightly anaesthetised. The other performance parameters of the model were at moderate and occasionally high levels.

The DoA index model showed strong relationships with BIS index values when considering Pearson correlation and R^2 however the model did have a significant degree error which will affect model performance.

3. Discussion

On analysis of results of the new DoA index when compared to the BIS index there are areas where the index performs well however also areas where it does not.

The new DoA index performs well with respect to the Pearson correlation coefficient in comparison with the BIS index. The Pearson correlation coefficient measures the strength of the linear relationship between two variables (Battiti & Brunato 2017). The average correlation coefficient across the five test patients was 0.856 indicating a strong linear relationship between the new DoA index and the BIS index. The model did achieve stronger results with some test patients compared to others. In particular patient 3 had the highest correlation between the DoA index and BIS index of 0.909 indicating a strong relationship between the two variables compared with patient 2 at 0.767 indicating a moderate relationship.

The Pearson correlation coefficient is a good indicator of the strength of the relationship of the relationship between two variables however it does not take into account model bias or error (Kampakis 2016). A model may have strong relationship between the predicted and target variable

however the Pearson correlation coefficient does not indicate whether these two variables are measuring the same value which is important for the model to be a success (Kampakis 2016). As is seen with the Pearson correlation coefficient of the new DoA index model there is a strong relationship between the BIS index and the new DoA index however the goal of the model is to accurately predict a value equivalent to the BIS index. Hence further parameters need to be assessed to fully gauge the model's performance. The model has a moderately high R^2 value at 0.735 indicating a strong degree of dependence of the model result on the predictor variables however it does not give an absolute measure of the accuracy of the model. Therefore it is also important to consider the errors of the model.

Unfortunately when considering the degree of error between the predicted and target values the model does not perform well. Three measures of error were calculated for the DoA index model, the mean absolute error (MAE), the mean squared error (MSE) and the root mean squared error (RMSE). All of these values give a unit measurement of the fit of the DoA index to the BIS index (Hiregoudar 2020). The first measure MAE, is the average error measure of the absolute errors between the DoA and BIS indices (Hiregoudar 2020; Wu 2020). The average MAE across the five test patients was 8.354. The highest MAE was seen for patient 2 at 10.356 with the model having the least error for patient 4 at 6.833. This error level is significant when considering that the depth of anaesthesia could be incorrectly measured on average by 8 points when considering the scale of measurement for the index is between 0 and 100. Furthermore when looking at the maximum absolute errors as visualised in the error histogram in some cases errors were as high as 55 index points which again on a scale of 0 to 100 would result in a drastically different interpretation of the degree of anaesthetic depth between the DoA and BIS indices.

The MSE and RMSE for the model were also high. The average MSE across the five test patients was very high at 124.908. The highest level of error was seen in patient 2 with an MSE of 182.853. The model performance was much better for patient 4 as the MSE was significantly less at 79.905. The RMSE for each patient is higher than that of the MAE with an average across the five test patients of 11.704. As previously discussed MSE and RSME calculations involve the square of the error making these unit measurements more sensitive to small numbers of larger errors (Hiregoudar 2020; Wu 2020). Again as seen in the absolute error histogram there is a small distribution of large value errors of up to 50 units, a substantial error for the model which will in turn increase both MSE and RMSE. Given the serious implications of large errors in prediction of the model, assessing model performance with these measurements is particularly important. When considering these measures of error and reviewing studies in the literature it is evident that much lower levels of error are achievable using a neural network model. Jiang et al 2015 utilised two neural network models to predict a DoA index based on the BIS index with an average MSE across 24 test patients of 10.19 and 13.24 respectively. Ortolani et al 2020 also utilised a neural network model to predict a DoA index achieving a MAE of 4.016 and an RSME 5.96 with 74 test patients. Sadrawi et al 2015 achieved an average MAE of 6.61 with their neural network model when tested on 46 patients.

To further assess the level of error of the model, MSE values were compared between model training and model test data sets. On examination of the performance plot of training of the neural network in the DoA index model the MSE measured at 84.536 which is again high (Appendix C, Figure 9). Given that the average MSE of the testing patients is also high this indicates that the model has a high level of bias and as result the model is underfitting the data set.

It is also interesting to compare the performance results for individual patients. In terms of error the model performed much better with predicted results for patient 4 compared other patients in the testing data set however patient 4 also had the second lowest value for Pearson correlation. Patient 3 had the highest level of correlation between the DoA Index and BIS index at 0.9 however had a

higher level of error than patient 4, indicating that higher levels of correlation do not necessarily indicate the model will also predict an accurate result.

On examination of the classification results of the model the outcomes are mixed. The model proved to be very poor at correctly identifying patients that were lightly anaesthetised with a sensitivity value of just 11% however it was very good at correctly classifying patients that were not in this category with a very high specificity of 94%. Excluding the awake category there were significantly less observations within the light anaesthesia category which may have influenced these results. Additionally the degree of error in this category is likely to be a major contributing factor to the overall error of the model. The average accuracy of the model in correctly classifying the level of anaesthesia in patients was good at 82.01% with the model having the best accuracy for awake patients despite having the lowest number of observations in this category. The other levels performance parameters for classification of the model were moderate.

On assessment of model performance in some areas the model does perform well however the model does have a high level of error indicating that the model is underfitting the data set and thus there is room for improvement.

4. Analysis for DoA Index Model Improvement

As seen in the performance assessment of the model, the new DoA index has strong correlation with the BIS index however the DoA index had a very high degree of error when compared with BIS scores. This high degree of error was seen when model performance was evaluated in both training and testing indicating that the model is underfitting the data set. Furthermore the sensitivity level for detecting patients under light anaesthesia was very low thus errors in this category are likely to also have significant influence on the overall error level of the model.

In response to these results, several methods were tried in an effort to improve model performance. The first method involved using the data of all ten training patients as part of the training patient data set in an effort to increase the volume of data to train the model. In the initial model building process, seven patients were used for model training and the remaining three were used for model testing. This strategy however did little to change the performance of the existing model.

Given various researchers had success with using a neural network as part of a DoA index model, it was elected trial applying other machine learning techniques to the input data for the neural network to see this may improve model performance (Gu et al 2019; Jiang et al 2015; Liu et al 2015 Ortolani et al 2002; Sadwari et al 2015). Adjusting the parameters of the neural network model did not significantly change model results during the initial model building process and combined with the fact that the model was underfitting the data set led to the proposal that revising the handling of the input data to the neural network may improve model performance (Labs 2018; Mahmood 2016).

The next method trialled to improve the DoA index model was to develop a decision tree model using the EEG training patient data set and use the results of the decision tree as input for the neural network. A study performed by Liu et al (2015) showed promising results with the use of random forests in predicting depth of anaesthesia with EEG data. When examining the performance of this updated neural network model after model training, this new model showed promise. As indicated by the performance plot shown in Appendix C, Figure 10, the updated model had an MSE of 18.0267, a significant improvement on the original model.

The regression plot for the newly trained neural network model (Appendix C, Figure 11) also highlights a high degree of correlation between the target and predicted variables in model building,

validation and testing of greater than 0.95. However the performance of the model when applied to the testing data set was mixed as shown by the results in Table 4 below.

Table 4: Performance results for neural network with decision tree data when applied to combined test patient data set

Pearson Correlation Coefficient ($p < 0.05$)	MAE	MSE	RMSE
0.7963 (0.7902 – 0.8023)	10.2219	184.7716	13.5931

The DoA index calculated by this modified model has a moderate to high correlation with the BIS index however this is slightly reduced from the results of the original model which had a correlation of 0.852. Furthermore, the degree of error in the model when applied to the combined patient test data set is increased from the results of the original model which had a MAE of 8.536, a MSE of 129.046 and a RMSE of 11.36 respectively. Therefore this modified model is overfitting the training data set with a low MSE for the training data set compared to a very high MSE for the test data set.

The next technique used in an effort to improve the original DoA index model was k-means clustering. K-means was applied to the EEG data set to calculate the centroids for each observation. These results were then used as input for the neural network. This technique however led to results similar to that of the original model. Similar levels of correlation for both training and test sets were achieved compared to the original model. Again there was a high level of error for the training and test data sets in line with the original model, resulting in an updated model that also underfits the data set thus leading to no improvement in the original model.

The final technique trialled in an effort to improve performance of the DoA index model was to apply a support vector regression (SVR) model to the data set and then use these results as input for the neural network. SVR enables a model to be developed to fit the data set in more than one dimension which may make it more appropriate than the linear model for the input EEG data (Sharp 2020). Research conducted by Hernandez-Meza et al (2017) and Jahanseir et al (2018) in the use of support vector machines (SVM) and SVR in prediction of anaesthetic depth using EEG data yielded promising results. In this case the SVR model was used with a Gaussian kernel and the data set was standardised. The results of the SVR model from the training data set were then used to train the neural network model. The performance and regression plots of model training of this updated model are shown Appendix C, Figures 12 and 13.

This new model was then applied to the combined patient test data set and the indicators of model performance, Pearson's correlation coefficient, MAE, MSE and RSME were calculated. Model classification performance was examined by calculating the sensitivity, specificity and accuracy of the model predictions for the four classes of anaesthesia. These results are shown in the Table 5 and 6 below. The confusion matrix of the updated model's classification results is shown in Appendix C, Figure 14.

Table 5: Performance results for the updated SVR/neural network DoA index model when applied to the combined test patient data set

Pearson Correlation Coefficient ($p < 0.05$)	MAE	MSE	RMSE
0.8405 (0.8356-0.8453)	8.4114	127.6102	11.2965

Table 6: Model classification parameters for the SVR/neural network DoA index model for the test patient data set

Class of Anaesthesia	Sensitivity	Specificity	Accuracy
Awake (80-100)	72.94%	98.01%	90.86%
Light (60-80)	31.57%	88.13%	84.83%
General Anaesthesia (40-60)	74.01%	84.93%	76.81%
Deep (<40)	76.02%	89.61%	83.66%

This updated model had very similar levels of error and correlation between the new DoA index and the BIS index when compared with the original model. Although the level of correlation between the new DoA index and the BIS index was high at 0.84, the degree of error of the updated model was high in both training and testing indicating that this updated model is still underfitting the data set. The overall accuracy of the updated model was 84.08%, a slight improvement from the original model at 82.01%. The accuracy of the model in predicting each class of anaesthesia was improved. The updated model also did improve its ability to correctly recognise patients that were lightly anaesthetised, with a sensitivity of 31.57% increased from 11.78%, although sensitivity for the model in this category is still very low. When examining other categories some parameters did improve slightly and others did not.

In summary, despite trialling alternative measures to modify the DoA index model there was minimal improvement in model performance. One the main limitations on model performance acknowledged by other authors is the volume of data (Gu et al 2019; Jiang et al 2015). Neural network models do require a large volume of data for model training which may have influenced model performance in this case. There were lower number of observations for both the awake and light anaesthesia categories which may have influenced results. In case of Gu et al (2019) the neural network model did not perform as well in predicting deep anaesthesia however the writers also acknowledged the smaller amount of data in this category in the training data set which may have affected model training and results. Shalbaf et al (2018) also acknowledges the limitations with using the BIS index as the control variable in model building. It was acknowledged that patients may be considered awake with BIS values that indicate anaesthesia and that time delays in receiving BIS results for a patient can also influence its use as the control variable (Shalbaf et al 2013). A final consideration for the data set is the effect of differing drug protocols for anaesthesia as different medications can affect EEG results differently and this may also influence modelling results (Shalbaf et al 2013).

5. Cybersecurity for Well Hospital

With the increase in the volume of data available to us and the valuable information it contains, cybersecurity is becoming an issue of great significance. Big data in the field of health holds great value for clinicians, research and can lead to improvement in patient care (Abouelmehdi et al 2018). However when utilising this information, patient consent and privacy are of the utmost importance (Abouelmehdi et al 2018; Lee et al 2016; Offner et al 2020). In terms of cybersecurity, health data is considered to be highly valuable given the level of personal information it contains, making it highly sought after and vulnerable to cybercrime (Coventry & Branley 2018; Lee et al 2016; Offner et al 2020). Furthermore the use of interconnecting systems and cloud computing are making cybersecurity systems even more vulnerable to malicious attacks (Abouelmehdi et al 2018; Offner et al 2020). Therefore it is highly important for Well Hospital to have a sound cybersecurity system.

There are several important issues to contemplate when developing a cybersecurity system. The first issue is data privacy which considers who has the authority retrieve a person's data, use the

information it contains and potentially share that information with other parties (Lawler 2019; Lee et al 2016). Data privacy also takes into account the extent to which we own our own individual information (Lee et al 2016). Security measures must ensure that data privacy of individuals is protected to prevent misuse of highly personal information potentially illegally. Data privacy is also governed by privacy law so it is imperative that we are adhering to legal regulations within our jurisdiction (Lawler 2019; Lee et al 2016). However not only do we need to consider our use of personal data in ordinance with the law, we also need to consider ethics and our moral principles in our decision making, does our use of information fit with our values, the values of those whose information we are utilising and what are the potential effects for all those involved (Lawler 2019; Lee et al 2016).

In accordance with our considerations for cybersecurity there are several aspects of security that need to be managed as part of the Well Hospital system. The first is intrusion detection, that is having the ability to detect abnormal activity within the hospital network (Buczak & Guven 2016; Ford & Siraj 2015). There are several machine learning techniques that have shown to be effective intrusion detection. In the review conducted by Handa et al (2019), K-means clustering combined with SVM were discussed as a potentially effective system for intrusion detection. Artificial neural networks have also shown good success when used for intrusion detection systems (Buczak & Guven 2016). Subbulakshmi et al (2010) demonstrated greater accuracy with SVM when compared to a neural network for intrusion detection. Buczak & Guven (2016) also highlighted the efficacy for clustering techniques in combination with other machine learning techniques and also highlighted that ensemble methods using a combination of machine learning techniques appeared to yield better results.

The next aspect of cybersecurity to consider is phishing detection. During data transfer our information is vulnerable to those attempting to gain access to confidential information. Abu-Nimeh et al (2007) compared several techniques including SVM, neural networks and random forests and found Logistic Regression to be the most effective technique for phishing detection. Cryptography will also protect our data from security threats. As discussed the review by Ford and Siraj (2015) of machine learning techniques in cybersecurity, neural networks were demonstrated by both Yu & Cao (2006) and Kinzel & Canter (2002) were effective for cryptography. Password management is also integral to managing cybersecurity of the hospital system. The password is temporary however implementing a system for keystroke dynamics to monitor user activity will further assist hospital security measures. Using a neural network model, Revett et al (2007) demonstrated the model to be successful at identifying a new user. Cybersecurity is of vital importance to Well Hospital and should be of the highest priority in data management systems. As discussed there are a number of machine learning techniques we can incorporate in development of this system.

6. Summary

The goal of this project was to develop a machine learning model to predict a DoA index determined from patient EEG data using the BIS index as a target variable. The model developed in the model building process was comprised of a linear model and a neural network model. This DoA index model was then tested with the EEG data of five test patients to measure model performance.

The new DoA index model had strong correlation between the predicted DoA index and the BIS index with an average across the five test patients of 0.856 indicating a strong linear relationship between the two variables. Unfortunately the new DoA index had a high degree of error with an average MAE, MSE and RMSE of 8.354, 124.908 and 11.701 respectively. The greatest error in classification was present in the light anaesthesia category (BIS 60-80). When examining model

error in model training this too was also high, confirming that the new DoA index model was underfitting the patient data set.

Given the success shown by several authors using a neural network model for depth of anaesthesia prediction using EEG data and the fact the model was underfitting the data set, attempts to improve model prediction results were focused on the input data of the neural network model (Gu et al 2019; Jiang et al 2015; Labs 2018; Mahmood 2016; Ortolani et al 2002; Sadrawi et al 2015). However this did not prove to be successful. Updated versions of the model still had a strong correlation between the new DoA index and BIS values however the level of error was persistently high and results did not improve by increasing the amount of data in the training data set.

The next attempt to improve the model involved applying a decision tree model to the input data prior to the neural network. This resulted in an overfitting DoA index model with a low MSE of 18.027 on the training data set and a high MSE of 184.772 for the test data set. Furthermore the MSE of this new model was even higher than the original model for the test data set. Applying an SVR model to the input data prior to the neural network was also trialled however this resulted in an underfitting model similar to the original model with similar levels of correlation and error between the new DoA index and BIS index. Model accuracy improved slightly with this updated model to 84.08% compared with 82.01% from the original model. This updated model also improved its ability to correctly classify patients as lightly anaesthetised, improving sensitivity to 31.57% however there is still great room for further improvement.

Although initial attempts were unsuccessful, further model development in addressing feature selection and neural network input may improve model performance. Several studies have used sample entropy calculations of EEG data as the input for neural network model with very promising results (Gu et al 2019, Jiang et al 2016; Sadwari et al 2015; Shalbaf et al 2013). Given then variation in EEG data and also being time dependant data, using measurements of entropy of an EEG time series data has shown significant promise for use with machine learning techniques for prediction of depth of anaesthesia and may prove useful with this data set (Gu et al 2019, Jiang et al 2016; Sadwari et al 2015; Shalbaf et al 2013).

The new DoA index model developed in this project showed strong correlation with the BIS index however had a high degree of error.

7. References

1. Abouelmehdi K, Abderrahim BH, Khaloufi H, 2018, 'Big Health Care Data: preserving security and privacy, *Journal of Big Data*, Vol 5 No 1, pp. 1-18
2. Abu-Nimeh S, Nappa D, Wang X, Nair S, 2007 'A Comparison of Machine Learning Techniques for Phishing Detection', *APWG eCrime Researchers Summit*, October 4-5 Pittsburg PA
3. Battiti R & Brunato M 2017 '*The Lion Way – Machine Learning plus intelligent optimization*', University of Trento, Italy, viewed 13th August 2020, <http://intelligent-optimization.org/LIONbook>
4. Coventry L, Branley D, 2018, 'Cybersecurity in healthcare: A narrative review of trends, threats and ways forward,' *Maturitas*, Vol 113, pp. 48-52
5. Ford V, Siraj A 2015 'Applications of Machine Learning in Cybersecurity' *Proceedings of the 27th International Conference on Computer Applications in Industry and Engineering*, October 2014
6. Gu Y, Liang Z & Hagihira S, 2019, 'Use of Multiple EEG Features and Artificial Neural Network to Monitor the Depth of Anaesthesia' *Sensors*, Vol 19, No 11:2499 pp. 1-12
7. Hernandez-Meza G, Izzetoglu M, Sacan A, Green M & Izzetoglu K (2017) 'Investigation into data-driven optical neuromonitoring approach to general anaesthesia with sevoflurane' *Neurophotonics*, 4(4), pp 1-10
8. Hiregoudar S 2020 'Ways to evaluate regression models' *Towards Data Science*, weblog post, 5th August, viewed 4th October 2020, <https://towardsdatascience.com/ways-to-evaluate-regression-models-77a3ff45ba70>
9. Jahanseir M, Setarehdan SK & Momenzadah S 2018 'Estimation of depth of anaesthesia by using a multioutput least square support vector regression' *Turkish Journal of Electrical Engineering and Computer Sciences*, Vol 26, pp 2792-2801
10. Jiang GJA, Fan SZ, Abbod MF, Huang HH, Lan JY, Tsai FF, Chang HC, Yang YW, Chuang FL, Chiu YF, Jen KK & Wu JF 2015 'Sample Entropy Analysis of EEG Signals via Artificial Neural Networks to Model Patients' Consciousness Level Based on Anaesthesiologists Experience' *Biomed Research International* Vol 2015: 343478, pp 1-8
11. Kampakis S 2016 'Performance Measures: The Concordance Correlation Coefficient' *The Data Scientist*, weblog post, 5th September, viewed 16th October 2020, <https://thedataScientist.com/performance-measures-concordance-correlation-coefficient/#:~:text=A%20common%20way%20to%20measure,true%20and%20the%20predicted%20values.&text=We%20can%20treat%20X%20as,1%2C%20so%20it%20is%20perfect.>
12. Kinzel W, Kanter I, 2002 'Neural Cryptography' *Proceedings of the 9th Conference on Neural Information Processing* Vol 3, pp 1351-1354, 18th-22nd November
13. Lawler B, 2019 'Five Global Trends in Data Ethics and Privacy in 2019' *Looker*, weblog, 28th January, viewed 24th October 2020, <https://looker.com/blog/big-data-ethics-privacy#:~:text=Data%20privacy%20is%20responsibly%20collecting,based%20on%20your%20brand%20values.>

14. Labs M 2018 ‘Tackle High Bias and Other Problems/Solutions in Machine Learning Models’, *Medium*, weblog, 17th December, viewed 19th October 2020,
<https://medium.com/hackeroon/tackle-bias-and-other-problems-solutions-in-machine-learning-models-f4274c5fe538>
15. Lee HC, Ryu HG, Park Y, Yoon SB, Yang SM, Oh HW & Jung CW, 2019, ‘Data Driven Investigation of Bispectral Index Algorithm, *Nature: Scientific Reports*, Vol 9:13769, pp 1-12
16. Lee WW, Zanki W, Chang H, 2016 ‘An Ethical Approach to Data Privacy Protection’, *ISACA Journal*, Vol 6, pp 1-9
17. Mahmood B, 2016 ‘4 reasons your machine learning model is wrong and how to fix it’, *KDnuggets*, weblog, 21st December, viewed 19th October 2020,
<https://www.kdnuggets.com/2016/12/4-reasons-machine-learning-model-wrong.html#:~:text=One%20straightforward%20method%20is%20to,sets%20and%20has%20High%20Bias.>
18. Mathworks Inc, 1994-2020 ‘Support’ Mathworks Inc, Natick MA, USA viewed 16th August 2020,
https://au.mathworks.com/support.html?s_tid=gn_supp
19. Offner KL, Sitnikova E, Joiner K, MacIntyre CR 2020 ‘Towards Understanding Cybersecurity Capability in Australian Healthcare Organisations: A Systemic review of recent trends, threats and mitigation, *Journal Intelligence and National Security* Vol 35 No 4, pp556-585
20. Ortolani O, Conti A, Fillipo D, Adembri C, Moraldi E, Evangelisti A, Maggini M & Roberts SJ 2002, ‘EEG signal processing in anaesthesia. Use of neural network technique for monitoring general anaesthesia’, *British Journal of Anaesthesia*, Vol 88, No 5, pp 644-648
21. Revett K, Gorunescu F, Gorunescu M, Ene M 2007 ‘A machine learning approach to keystroke dynamics based on user authentication’ *International Journal of Electronic Security and Digital Forensics*, Vol 1, No 1, pp 55-70
22. Sadrawi M, Fan SZ, Abbad MF, Jen KK & Sheih JS 2015 ‘Computation Depth of Anaesthesia via Multiple Vital Signs based on Artificial Neural Networks’ *Biomed Research International*, Vol 2015: 536683 pp 1-13
23. Shalbaf R, Behnam H, Sleigh JW, Steyn-Ross A, Voss LJ 2013, ‘Monitoring the depth of anaesthesia using entropy features and an artificial neural network’ *Journal of Neuroscience Methods*, Vol 218, pp 17-24
24. Sharp T 2020 ‘An Introduction to Support Vector Regression (SVR), *Towards Data Science*, weblog post 4th March, viewed 24th October 2020, <https://towardsdatascience.com/an-introduction-to-support-vector-regression-svr-a3ebc1672c2>
25. Subbulakshmi T, Shalinie SM, Ramamoorthi 2010, ‘Detection of Classification of DDoS Attacks Using Machine Learning Algorithms’ *European Journal of Scientific Research*, Vol 74 No 3, pp 334-346
26. Wu S 2020 ‘3 Best Metrics to evaluate a regression model?’ *Towards Data Science*, weblog post, 24th May, viewed 16th October 2020, <https://towardsdatascience.com/what-are-the-best-metrics-to-evaluate-your-regression-model-418ca481755b>

27. Yu W, Cao J, 2016 ‘Cryptography based on delayed chaotic neural networks’ *Physics Letters A*, Vol 356 Issues 4-5 pp 333-338

8. Appendices

Appendix A:

MATLAB Code for applying DoA Index Model to Test Patient Data Set and Measuring Performance

```

%%
%%%PROJECT FINAL REPORT
%%% Load test data set

test1 = readtable('Data set (stage 2) with BIS value','Sheet','Test1');
test2 = readtable('Data set (stage 2) with BIS value','Sheet','Test2');
test3 = readtable('Data set (stage 2) with BIS value','Sheet','Test3');
test4 = readtable('Data set (stage 2) with BIS value','Sheet','Test4');
test5 = readtable('Data set (stage 2) with BIS value','Sheet','Test5');

%%
%%%Total data set
test = readtable('Data set (stage 2) with BIS value','Sheet','TestTotal');

%%
%%% Apply feature selection and linear model to test data

X = [test.x2, test.x4, test.x5];

linResult2 = predict(lin,X);

%%
%%%Test Model on combined test data set
%%%Input data for model testing from combined test data set
PInput = transpose(linResult2);

%%
%%%Target values from combined test data set
Q = table2array(test)
PResult = transpose(Q(:, 1));

%%
%%%Calculate model performance for combined test data set
predP = sim(net,PInput);

[RP,PP,RLP,RUP] = corrcoef(predP,PResult);

eP = PResult-predP;
perfP = mae(eP); %%MAE

errP = mse(net,PResult,predP) %%MSE

%%
%%%Plot Results of Test
%%%BIS an predP values for whole data set
figure; hold on;
plot (predP);
plot(PResult,'m');

%%
%%%Scatter plot of NN results and BIS scores
figure; hold on;
plot(predP, PResult, 'x');
h = lsline;
h.Color = 'k';

```

```

%%
%% Error histogram
figure;
ploterrhist(eP);

%%
% Apply model to test1

%Calculate inputs from linear model
TP1 = [test1.x2, test1.x4, test1.x5]
TR1 = [test1.BIS];

linResultTP1 = predict(lin,TP1);

TI1 = transpose(linResultTP1);
TR1 = transpose(TR1);

predT1 = sim(net, TI1);

[RP,PP,RLP,RUP] = corrcoef(predT1,TR1);

eT1 = TR1-predT1;
perfT1 = mae(eT1); %%MAE

errT1 = mse(net,TR1,predT1) %%MSE

%%Plot Results of Test 1
figure; hold on;
plot (predT1);
plot(TR1, 'm');

%% %%
%% Apply model to test2

%Calculate inputs from linear model
TP2 = [test2.x2, test2.x4, test2.x5]
TR2 = [test2.BIS];

linResultTP2 = predict(lin,TP2);

TI2 = transpose(linResultTP2);
TR2 = transpose(TR2);

predT2 = sim(net, TI2);

[RP,PP,RLP,RUP] = corrcoef(predT2,TR2);

eT2 = TR2-predT2;
perfT2 = mae(eT2); %%MAE

errT2 = mse(net,TR2,predT2) %%MSE

%%Plot Results of Patient test2
figure; hold on;
plot (predT2);
plot(TR2, 'm');

%%
% Apply model to test3

```

```
%Calculate inputs from linear model
TP3 = [test3.x2, test3.x4, test3.x5]
TR3 = [test3.BIS];

linResultTP3 = predict(lin,TP3);

TI3 = transpose(linResultTP3);
TR3 = transpose(TR3);

predT3 = sim(net, TI3);

[RP,PP,RLP,RUP] = corrcoef(predT3,TR3);

eT3 = TR3-predT3;
perfT3 = mae(eT3); %%MAE

errT3 = mse(net,TR3,predT3) %%MSE

%%Plot Results of Patient test3
figure; hold on;
plot (predT3);
plot(TR3, 'm');

%%
%%% Apply model to test4

%Calculate inputs from linear model
TP4 = [test4.x2, test4.x4, test4.x5]
TR4 = [test4.BIS];

linResultTP4 = predict(lin,TP4);

TI4 = transpose(linResultTP4);
TR4 = transpose(TR4);

predT4 = sim(net, TI4);

[RP,PP,RLP,RUP] = corrcoef(predT4,TR4);

eT4 = TR4-predT4;
perfT4 = mae(eT4); %%MAE

errT4 = mse(net,TR4,predT4) %%MSE

%%Plot Results of Patient test4
figure; hold on;
plot (predT4);
plot(TR4, 'm');

%%
% Apply model to test5

%Calculate inputs from linear model
TP5 = [test5.x2, test5.x4, test5.x5]
TR5 = [test5.BIS];

linResultTP5 = predict(lin,TP5);

TI5 = transpose(linResultTP5);
TR5 = transpose(TR5);
```

```
predT5 = sim(net, TI5);

[RP, PP, RLP, RUP] = corrcoef(predT5, TR5);

eT5 = TR5 - predT5;
perfT5 = mae(eT5); %%MAE

errT5 = mse(net, TR5, predT5) %%MSE

%%Plot Results of Patient test5
figure; hold on;
plot (predT5);
plot(TR5, 'm');

%%
%%%Confusion Matrix - Total patient test set

predP = transpose(predP);
PResult = transpose(PResult);
%%
%%%Convert predicted results to range 0-40, 40-60, 60-80, 80-100
A = predP(:,1) < 100;
predP(A,2) = 100;

A = predP(:,1) < 80;
predP(A,2) = 80;

A = predP(:,1) < 60;
predP(A,2) = 60;

A = predP(:,1) < 40;
predP(A,2) = 40;
%%
%%% Convert actual results to range 0-40, 40-60, 60-80, 80-100
A = PResult(:,1) < 100;
PResult(A,2) = 100;

A = PResult(:,1) < 80;
PResult(A,2) = 80;

A = PResult(:,1) < 60;
PResult(A,2) = 60;

A = PResult(:,1) < 40;
PResult(A,2) = 40;

BIS1 = PResult(:,2);
DoA1 = predP(:,2);
CM = confusionmat(BIS1, DoA1);
CC = confusionchart(BIS1, DoA1);
```

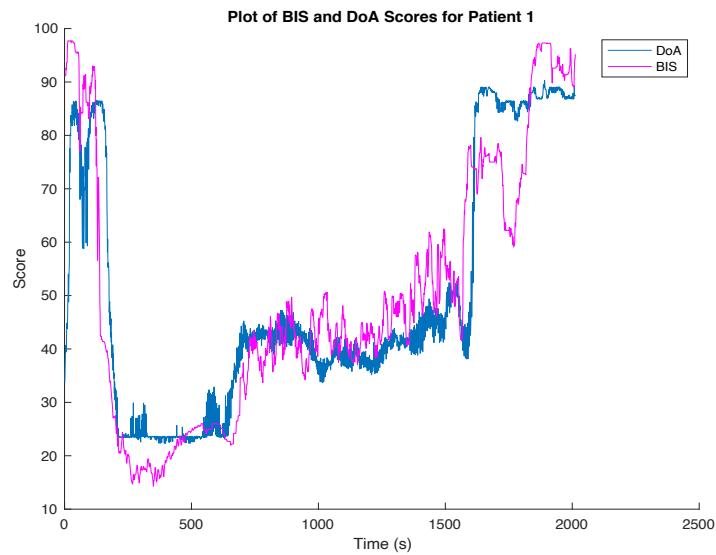
Appendix B:

Figure 5: Plot of BIS and DoA indices for test patient 1 over time (s)

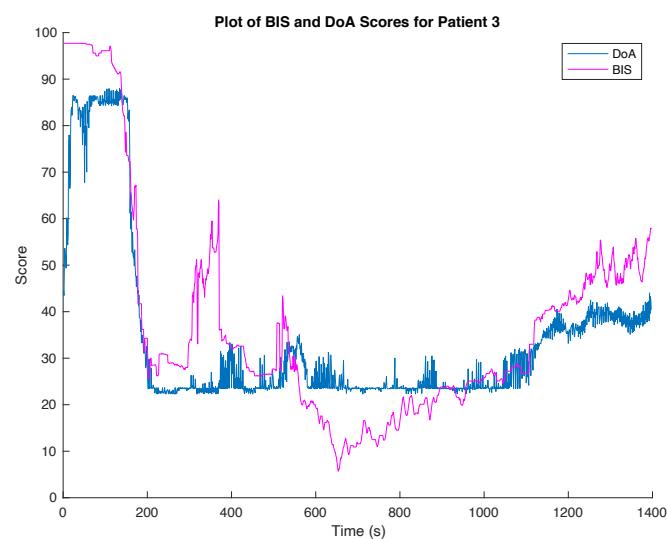


Figure 6: Plot of BIS and DoA indices for test patient 3 over time (s)

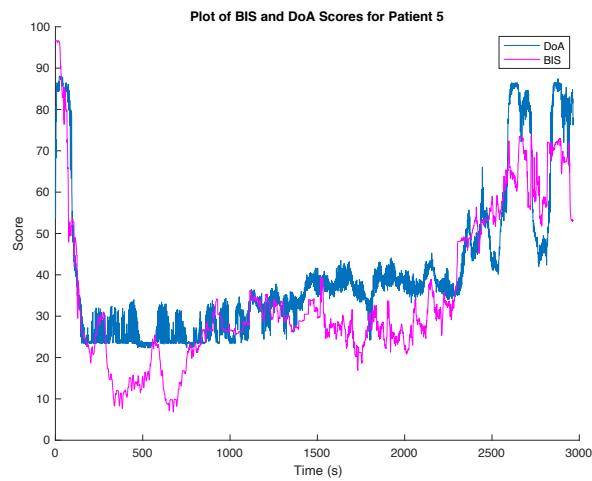


Figure 7: Plot of BIS and DoA indices for test patient 5 over time (s)

Appendix C

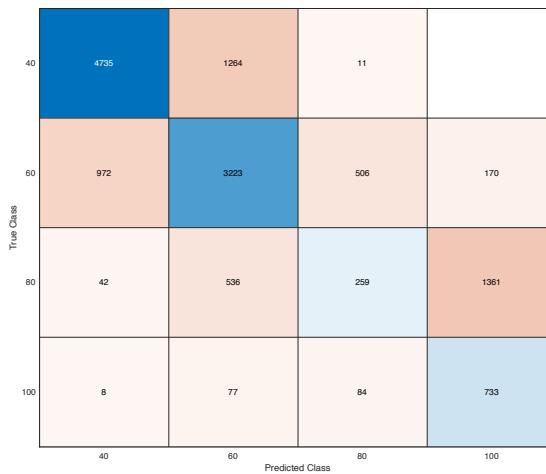


Figure 8: Confusion matrix for DoA index model for test patient data set

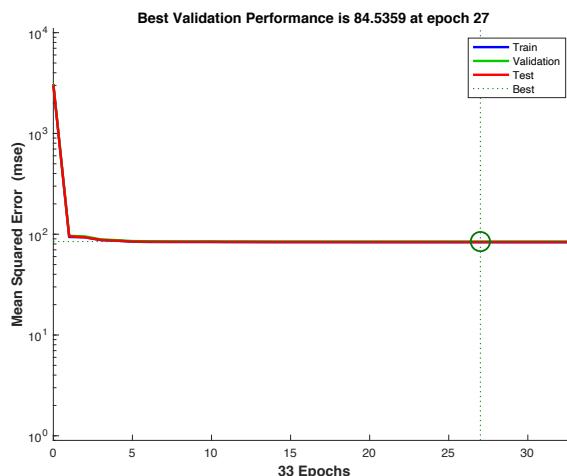


Figure 9: Performance plot for training of neural network in original DoA index model

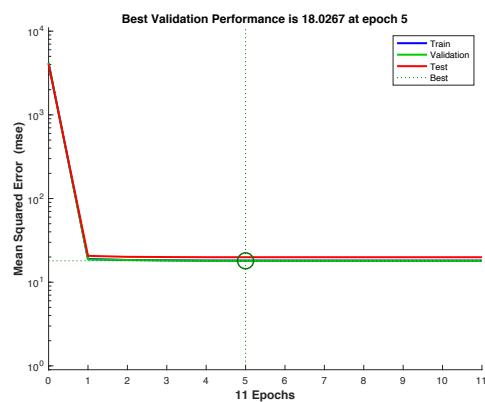


Figure 10: Performance plot for training of updated decision tree/neural network DoA index model

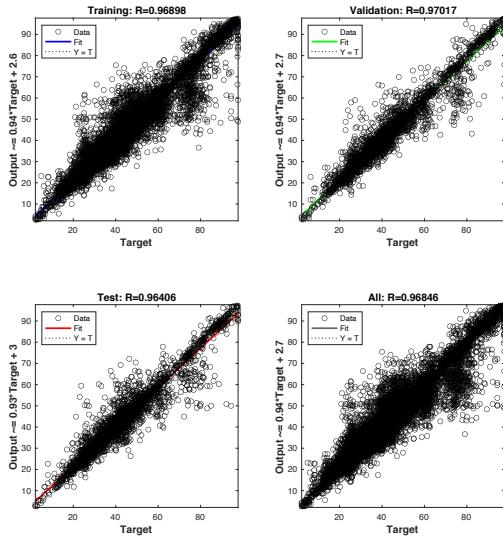


Figure 11: Regression plots of performance of updated decision tree/neural network DoA index model

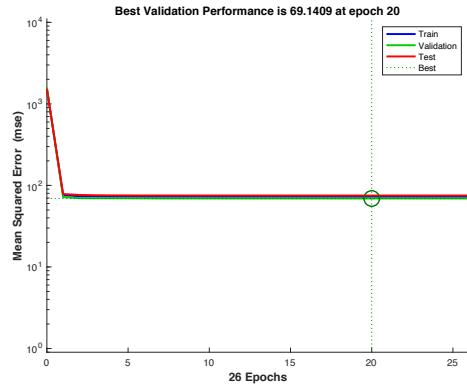


Figure 12: Performance plot for training of the updated SVR/neural network DoA index model

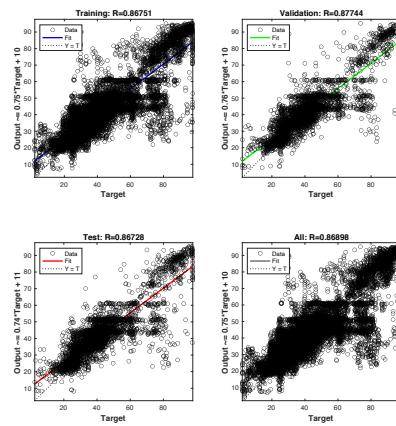


Figure 13: Regression plot for training of the updated SVR/neural network DoA index model

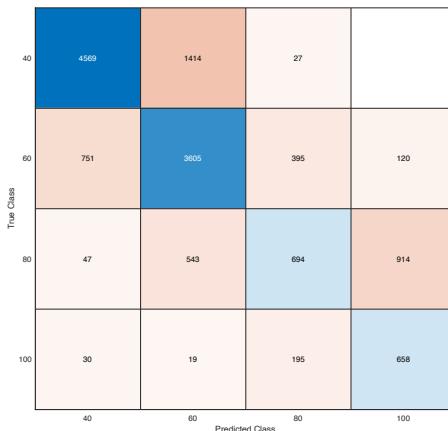


Figure 14: Confusion matrix for predicted DoA values and actual BIS values for the updated SVR/neural network DoA index model for test patient data set

Appendix D:*MATLAB Code for model improvements*

```

%%
%%% FINAL PROJECT IMPROVE MODEL
%%%Updated model #1
%%%Combine use 10 training patients instead of 7 and increase number of
%%%data

X = [eeg.x2, eeg.x4, eeg.x5];

%%%Target values from eeg data

TrainResult = g(:, 1);

%%
%%%DECISION TREE + NEURAL NETWORK
%%% Build regression tree with training data and apply tree to data to
%%% obtain prediction for neural network input

tree = fitrtree(X, TrainResult);

Pred = predict(tree,X)

Pred = transpose(Pred);
TrainResult = transpose(TrainResult);

%%

%%%Build neural Network
netconf = [10]

net = feedforwardnet(netconf, 'trainlm');
net = train(net,Pred,TrainResult);

net.trainParam.epochs=1000;
net.trainParam.goal=0.05;
net.trainParam.show=50;
net.trainParam.lr=0.05;
net.trainParam.mc=0.8;
net.divideFcn= 'dividerand';

%%

%%% Apply new model to test data set

XT = [test.x2, test.x4, test.x5];

%%
%%%Test Model on combined test data set
%%%Input data for model testing from combined test data set
PTest = predict(tree,XT)
%%

PTest = transpose(PTest);

%%%Target values from combined test data set
Q = table2array(test)
PResult = transpose(Q(:, 1));

%%

%%%Calculate model performance for combined test data set

```

```

predTest = sim(net, PTest);

[RP, PP, RLP, RUP] = corrcoef(predTest, PResult);

eP = PResult - predTest;
perfP = mae(eP); %%MAE

errP = mse(net, PResult, predTest); %%MSE

%%%
%%% UPDATED MODEL #2 - SVR + NEURAL NETWORK
%%%Build neural Network

%%% Input into SVR

Input = X;
%%

Input = transpose(Input);
TrainResult = transpose(TrainResult);
%%

%%%%%
%%Build SVM Model

mdl2 = fitrsvm(Input, TrainResult, 'Standardize', true, 'KernelFunction',
'gaussian');

%%

TT = predict(mdl2, Input);

%%

TT = transpose(TT);

%%

netconf = [10]

net = feedforwardnet(netconf, 'trainlm');
net = train(net, TT, TrainResult);

net.trainParam.epochs=1000;
net.trainParam.goal=0.05;
net.trainParam.show=50;
net.trainParam.lr=0.05;
net.trainParam.mc=0.8;
net.divideFcn= 'dividerand';

%%%
%%%PERFORMANCE OF MODEL
%%%Test Data

%%%Apply SVMR model
XT = transpose(XT);
predT2 = predict(mdl2, XT);
%%
```

```

predT2 = transpose(predT2);

%%
%%Apply NN Model
IT = sim (net, predT2);

%%
%%Performance

[RP,PP,RLP,RUP] = corrcoef(IT,PResult);

eP = PResult-IT;
perfP = mae(eP); %%MAE

errP = mse(net,PResult,IT); %%MSE

R2 = corrcoef(IT,PResult);

%%
%%UPDATED MODEL #3 - K MEANS + NEURAL NETWORK
X = transpose(X);
[idx,C] = kmeans(X,20);

%%
%% Assign k menas values to variables in data set
XK = X;

XK = [XK, idx];
%%
%% Assign kmeans values to observations

idx2 = XK(:,4) == 1;
XK(idx2,5) = C(1,1);
XK(idx2,6) = C(1,2);
XK(idx2,7) = C(1,3);

idx2 = XK(:,4) == 2;
XK(idx2,5) = C(2,1);
XK(idx2,6) = C(2,2);
XK(idx2,7) = C(2,3);

idx2 = XK(:,4) == 3;
XK(idx2,5) = C(3,1);
XK(idx2,6) = C(3,2);
XK(idx2,7) = C(3,3);

idx2 = XK(:,4) == 4;
XK(idx2,5) = C(4,1);
XK(idx2,6) = C(4,2);
XK(idx2,7) = C(4,3);

idx2 = XK(:,4) == 5;
XK(idx2,5) = C(5,1);
XK(idx2,6) = C(5,2);
XK(idx2,7) = C(5,3);

idx2 = XK(:,4) == 6;
XK(idx2,5) = C(6,1);

```

```
XK(idx2,6) = C(6,2);
XK(idx2,7) = C(6,3);

idx2 = XK(:,4) == 7;
XK(idx2,5) = C(7,1);
XK(idx2,6) = C(7,2);
XK(idx2,7) = C(7,3);

idx2 = XK(:,4) == 8;
XK(idx2,5) = C(8,1);
XK(idx2,6) = C(8,2);
XK(idx2,7) = C(8,3);

idx2 = XK(:,4) == 9;
XK(idx2,5) = C(9,1);
XK(idx2,6) = C(9,2);
XK(idx2,7) = C(9,3);

idx2 = XK(:,4) == 10;
XK(idx2,5) = C(10,1);
XK(idx2,6) = C(10,2);
XK(idx2,7) = C(10,3);

idx2 = XK(:,4) == 11;
XK(idx2,5) = C(11,1);
XK(idx2,6) = C(11,2);
XK(idx2,7) = C(11,3);

idx2 = XK(:,4) == 12;
XK(idx2,5) = C(12,1);
XK(idx2,6) = C(12,2);
XK(idx2,7) = C(12,3);

idx2 = XK(:,4) == 13;
XK(idx2,5) = C(13,1);
XK(idx2,6) = C(13,2);
XK(idx2,7) = C(13,3);

idx2 = XK(:,4) == 14;
XK(idx2,5) = C(14,1);
XK(idx2,6) = C(14,2);
XK(idx2,7) = C(14,3);

idx2 = XK(:,4) == 15;
XK(idx2,5) = C(15,1);
XK(idx2,6) = C(15,2);
XK(idx2,7) = C(15,3);

idx2 = XK(:,4) == 16;
XK(idx2,5) = C(16,1);
XK(idx2,6) = C(16,2);
XK(idx2,7) = C(16,3);

idx2 = XK(:,4) == 17;
XK(idx2,5) = C(17,1);
XK(idx2,6) = C(17,2);
XK(idx2,7) = C(17,3);

idx2 = XK(:,4) == 18;
XK(idx2,5) = C(18,1);
XK(idx2,6) = C(18,2);
XK(idx2,7) = C(18,3);
```

```

idx2 = XK(:,4) == 19;
XK(idx2,5) = C(19,1);
XK(idx2,6) = C(19,2);
XK(idx2,7) = C(19,3);

idx2 = XK(:,4) == 20;
XK(idx2,5) = C(20,1);
XK(idx2,6) = C(20,2);
XK(idx2,7) = C(20,3);

%%
%%% Select variables from XK for neural network model
XK2 = XK(:,5:7);

%%
XK2 = transpose(XK2);

%%
netconf = [10]

net = feedforwardnet(netconf, 'trainlm');
net = train(net,XK2,TrainResult);

net.trainParam.epochs=1000;
net.trainParam.goal=0.05;
net.trainParam.show=50;
net.trainParam.lr=0.05;
net.trainParam.mc=0.8;
net.divideFcn= 'dividerand';

%%
%%%Performance of model on test set data

[idxT,C] = kmeans(XT,20);

%%
%%% Assign k menas values to variables in data set
XKT = XT;

XKT = [XKT, idxT];

%%
%%% Assign k mean values
idxT = XKT(:,4) == 1;
XKT(idxT,5) = C(1,1);
XKT(idxT,6) = C(1,2);
XKT(idxT,7) = C(1,3);

idxT = XKT(:,4) == 2;
XKT(idxT,5) = C(2,1);
XKT(idxT,6) = C(2,2);
XKT(idxT,7) = C(2,3);

idxT = XKT(:,4) == 3;
XKT(idxT,5) = C(3,1);
XKT(idxT,6) = C(3,2);
XKT(idxT,7) = C(3,3);

idxT = XKT(:,4) == 4;
XKT(idxT,5) = C(4,1);

```

```
XKT(idxT,6) = C(4,2);
XKT(idxT,7) = C(4,3);

idxT = XKT(:,4) == 5;
XKT(idxT,5) = C(5,1);
XKT(idxT,6) = C(5,2);
XKT(idxT,7) = C(5,3);

idxT = XKT(:,4) == 6;
XKT(idxT,5) = C(6,1);
XKT(idxT,6) = C(6,2);
XKT(idxT,7) = C(6,3);

idxT = XKT(:,4) == 7;
XKT(idxT,5) = C(7,1);
XKT(idxT,6) = C(7,2);
XKT(idxT,7) = C(7,3);

idxT = XKT(:,4) == 8;
XKT(idxT,5) = C(8,1);
XKT(idxT,6) = C(8,2);
XKT(idxT,7) = C(8,3);

idxT = XKT(:,4) == 9;
XKT(idxT,5) = C(9,1);
XKT(idxT,6) = C(9,2);
XKT(idxT,7) = C(9,3);

idxT = XKT(:,4) == 10;
XKT(idxT,5) = C(10,1);
XKT(idxT,6) = C(10,2);
XKT(idxT,7) = C(10,3);

idxT = XKT(:,4) == 11;
XKT(idxT,5) = C(11,1);
XKT(idxT,6) = C(11,2);
XKT(idxT,7) = C(11,3);

idxT = XKT(:,4) == 12;
XKT(idxT,5) = C(12,1);
XKT(idxT,6) = C(12,2);
XKT(idxT,7) = C(12,3);

idxT = XKT(:,4) == 13;
XKT(idxT,5) = C(13,1);
XKT(idxT,6) = C(13,2);
XKT(idxT,7) = C(13,3);

idxT = XKT(:,4) == 14;
XKT(idxT,5) = C(14,1);
XKT(idxT,6) = C(14,2);
XKT(idxT,7) = C(14,3);

idxT = XKT(:,4) == 15;
XKT(idxT,5) = C(15,1);
XKT(idxT,6) = C(15,2);
XKT(idxT,7) = C(15,3);

idxT = XKT(:,4) == 16;
XKT(idxT,5) = C(16,1);
XKT(idxT,6) = C(16,2);
XKT(idxT,7) = C(16,3);
```

```

idxT = XKT(:,4) == 17;
XKT(idxT,5) = C(17,1);
XKT(idxT,6) = C(17,2);
XKT(idxT,7) = C(17,3);

idxT = XKT(:,4) == 18;
XKT(idxT,5) = C(18,1);
XKT(idxT,6) = C(18,2);
XKT(idxT,7) = C(18,3);

idxT = XKT(:,4) == 19;
XKT(idxT,5) = C(19,1);
XKT(idxT,6) = C(19,2);
XKT(idxT,7) = C(19,3);

idxT = XKT(:,4) == 20;
XKT(idxT,5) = C(20,1);
XKT(idxT,6) = C(20,2);
XKT(idxT,7) = C(20,3);

%%
XKT2 = XKT(:,5:7);

%%
XKT2 = transpose(XKT2);

%%% Test model performance in NN

predTestK = sim(net,XKT2);

[RP,PP,RLP,RUP] = corrcoef(predTestK,PResult);

eP = PResult-predTest;
perfP = mae(eP); %%MAE

errP = mse(net,PResult,predTest); %%MSE

%%
%%%Confusion Matrix - Total patient test set

IT = transpose(IT);
PResult = transpose(PResult);
%%
%%%Convert predicted results to range 0-40, 40-60, 60-80, 80-100
A = IT(:,1) < 100;
IT(A,2) = 100;

A = IT(:,1) < 80;
IT(A,2) = 80;

A = IT(:,1) < 60;
IT(A,2) = 60;

A = IT(:,1) < 40;
IT(A,2) = 40;
%%
%%% Convert actual results to range 0-40, 40-60, 60-80, 80-100
A = PResult(:,1) < 100;
PResult(A,2) = 100;

A = PResult(:,1) < 80;
PResult(A,2) = 80;

```

```
A = PResult(:,1) < 60;  
PResult(A,2) = 60;  
  
A = PResult(:,1) < 40;  
PResult(A,2) = 40;  
  
BIS1 = PResult(:,2);  
DoA1 = IT(:,2);  
CM = confusionmat(BIS1, DoA1);  
CC = confusionchart(BIS1,DoA1);
```