

Formulation Blending Recognition use Artificial Intelligence (AI)

Ahmad Shafwan Zakaria¹, Dinda Clarissa Aulia¹, Leong li Yen¹, Muhammad Faris Hilmi Sukar@Yahaya¹, Nurul Suihailah Juwahir¹, Mohd. Hafis Izran Ishak^{1*}, Mahmoud Jameel Atta Daasan¹

¹Faculty of Electrical Engineering, Universiti Teknologi Malaysia, 81300 UTM, Skudai, Johor, Malaysia

Corresponding author* email: hafis@fke.utm.my

Available online 30 December 2025

ABSTRACT

This project involves with artificial intelligence to predict the blending failure for the contact lens. The formulation for the contact lens is Lotrafilcon formulation that consists of betacon macromer, TRIS, DMA, ethanol and irgacure. Each raw material in the formulation has a range of weight controlled by the Chemical Production that manually weigh by the operator. Then, the material is mixed together for about 6 hours but after 2.5 hours a sample is taken from the mixing material to be analyze by the Analytic Lab. But when the formulation undergoes FTNIR analysis, the mixture fails due to betacon macromere weight that does not equivalent with the manually weighted one. This project will use artificial intelligence to train a system with the existing data of the analysis from Analytic Lab and Chemical Production and predict a weigh data that will match the formulation specification and pass the FTNIR analysis.

Keywords – Betacon macromer, machine learning, FTNIR, artificial intelligence.

1. Introduction

In this project, the focus is to reduce the failure of Lotrafilcon formulation process. Lotrafilcon, contact lens polymers (hydrogel and silicone hydrogel materials) As an upgrade to Air Optix Aqua (lotrafilcon B), Alcon launched its monthly disposable Air Optix with HydraGylde (lotrafilcon B) to the United States market in 2016. The lens has a 25µm permanent plasma surface treatment, which the company calls 'SmartShield.' The continuous hydrophilic layer is designed to help the lens resist lipid deposition. Lotrafilcon B has a significantly lower absorption of cholesterol compared with other SiHy lens materials. [1] Lenses are made of Lotrafilcon formulation. All Lotrafilcon formulations consist of betacon macromer, TRIS, DMA, ethanol and irgacure with different types and different ratio.

The general procedure of formulation includes preparation of making formulation, which is documentation, determining formulation batch size, obtaining ingredients. Determine target weight (if required) and verify completeness of labels. Next is weighing process in which the weight of each raw material is controlled stringently by Chemical Production. The mixing process is hen carried out. Figure 1 shows the detailed procedure of mixing procedure.

- 1 Place the sealed bottle on the roller. If there is only one formulation bottle being placed on the roller, use the empty amber glass bottle identified as the “dummy bottle”.
- 2 Record roller ID, tachometer ID, Countdown Timer ID and the relevant calibration due dates for the items in batch record.
- 3 Start the roller and well positioning the tachometer optical sensor at the reflective tape on the bottle. The green light on the tachometer will glow when it is properly receiving reflections.
- 4 Adjust the mixing speed between and inclusive of 95 to 120 rpm.
- 5 Once the desired speed is achieved, activate the Timer to start the countdown time. The allowable mix time is between and inclusive of 1.5 hours to 6.0 hours.
- 6 Record set time value in the batch record form using hour: minute format. Record mixing speed display on tachometer.
- 7 Monitor roller speed in rpm (rotation per minutes) to ensure the roller speed is within the specification. Record the roller monitoring speed and remaining countdown time in the batch record.
- 8 Do not adjust and handle the glass bottles during Jar Roller unit in operation. Ensure the Jar Roller unit is properly shutdown before adjusting and handling the glass bottles.
- 9 When the mixing process is completed, remove the Parafilm after remove the bottle from the roller. Fill in “mixing completed by” in batch record.

Figure 1. Detail procedure of mixing process.

After 2.5 hours, a sample of mixtures will be taken out from the bottle and sent to the analytical lab. Analytical lab will undergo FTNIR test in order to check if the mixtures is correctly weighed. However, only four materials matter in this step which is all except irgacure as irgacure only 1% of the mixtures. If the formulation is approved, then it will pass to filtering process and lastly, they will store filtered formulation batches in on-hold walk-in refrigerator.

In this project, the approach to solve the problem is analyzing the database of Betacon macromer in three stages which are R&D stage, production stage and analytical stage, training its pattern and do prediction about the formulation result using neural network.

Machine learning is programming computers to optimize a performance criterion using example data or experience. Machine learning uses the theory of statistics in building mathematical models because the core task is making inference from a sample. The example of machine learning function is classification which fits a model to the past data to be able to calculate the risk for a new application and then decides to accept or refuse it accordingly. Hence, it is a suitable algorithm to be used in this project to solve the problem. This is an example of a classification problem where there are two classes which are low-risk and high-risk customers. The information about the customer makes up the inputs to classifier whose task is to assign the input of one of the two classes. After training with the past data, a classification rule learned may be of the form. The figure2 shows the classification rule trained.

IF income > θ_1 AND savings > θ_2 THEN low-risk ELSE high-risk

Figure 2. The classification rule trained from the past data.

Having a rule like this, the main application is prediction. Once having a rule that fits the past data, if the future is like the past, a correct prediction can be made for novel instances. [2]

A neural network is an interconnected assembly of simple processing elements, units or nodes, whose functionality is loosely based on the animal neuron. The processing ability of the network is stored in the interunit connection strengths, or weights, obtained by a process of adaptation to, or learning from, a set of training patterns. To flesh this out a little we first take a quick look at some basic neurobiology. The human brain consists of an estimated 1011 (100 billion) nerve cells or neurons, a highly stylized example of which is shown in Figure 3. [3]

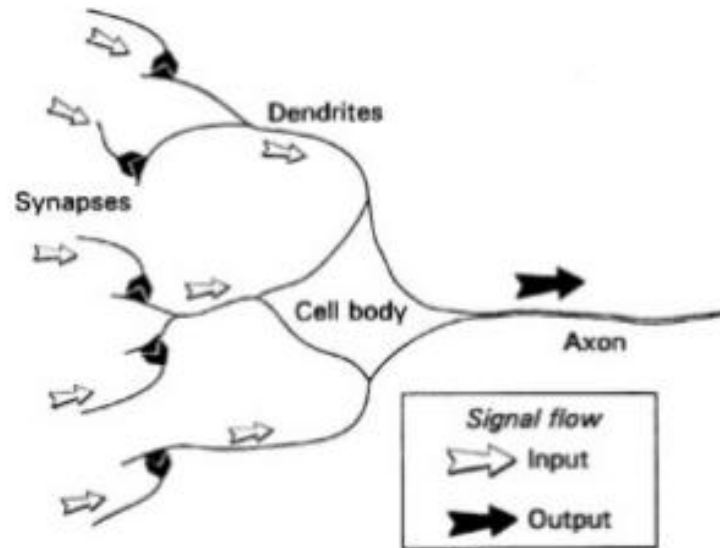


Figure 3. Essential components of a neuron shown in stylized form.

Training a neural network for classification normally involves minimizing a squared error criterion over a set of sample inputs and target vectors. During actual classification, the output neuron with the maximum signal level is used to determine the class of the input. [4]

Deep learning is about automatically learning multiple levels of representations of the underlying distribution of the data to be modeled [5]. In other words, a deep learning algorithm automatically extracts the low- and high-level features necessary for classification. By high level features, one means feature that hierarchically depends on other features. For instance, in the context of computer vision, this implies that a deep learning algorithm will learn its own low level representations from a raw image (such as edge detector, gabor filters, etc...), then build representations that depend on those low level representations (such as a linear or non-linear combinations of those low-level representations), and successively repeat the same process for higher levels. [6]

2. Problem Statement

Contact lens are made from 5 different substances which are betacon macromer, TRIS, irgacure, DMA and ethanol. To meet a specification to form a mixture of the contact lens that is made from these 5 different chemical substances through an appropriate mixing process for two and a half hours. This process is to make sure that the solution will achieve the percentage of betacon macromer for 26.25%, TRIS for 19.47%, Irgacure for 1%, DMA for 28.33% and lastly Ethanol for 24.95% respectively. Based on the different tolerance value of each substance, the percentage must be in between the upper solution limit and lower solution limit values. This is to make sure that the contact lens produced will not bring any harm to the users. It is always found that the betacon macromer substance will cause the faulty in the final mixture percentage.

3. Methodology

This method is designed to minimize the failure in formulation blending. The process of formulation consists of a few processes which start at research and development department. Each batch of Betacon Macromer is tested first before it is cleared for mixing in production area. A mini lab session is performed by taking small amount of Betacon Macromer from one batch and tested according to few specifications. After testing at mini lab, the batch is then cleared for mixing according to ranges of numbers. It is believed that there is a relationship between specifications obtained during mini lab session at R&D department with the required specifications of weight needed for large scale production. Therefore, with existing data available, a neural network with function fitting algorithm is going to be designed to predict the right amount of weight needed to mix the formula to minimize the failure. A neural network toolbox is going to be used by using MATLAB software. The system is going to be trained, and the network will be able to predict the amount of weight needed at production area just by entering specifications obtained at R&D department.

An (artificial) neural network is a network of simple elements called neurons which receive input, change their internal state(activation) according to the input and produce output depending on the input and activation. Each connection link has an appropriate weight where the transmitted signal will multiply with it. Sum of the weighted inputs

and the bias is fed into transfer function f . Every neuron uses any differentiate transfer function f to generate their output. Figure 4 shows the neuron model.

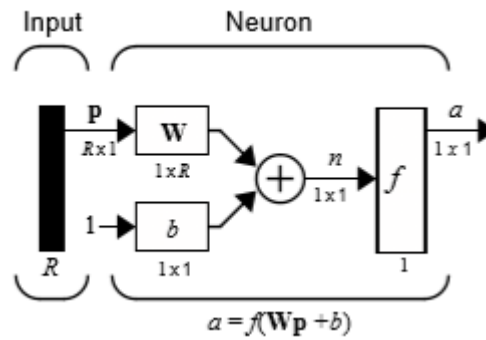


Figure 4. Neuron model.

Collection of data is the first step in the workflow while using Neural Networks. In this project, the data required will be the analytic result of each batch of Betacon Macromer tested first before it is cleared for mixing in production area, weight of Macromer before mixing and percentage of Macromer at FTNIR analysis. Next step will be creating the network. A lot of criteria will be considered in this stage including the number of layers needed for this network. Third step is configuring the network. Suitable algorithms will be chosen during this stage. Furthermore, it is required to initialise the weights and biases. Next step is training the network. The purpose of training network is to ensure the output is matched with the target as much as possible. It is followed by validating the network. Old data is used to train the network while new data is used to validate the network in order to determine the error coefficient. And finally, the network is able to used.

4. Procedure of using MATLAB Neural Network Toolbox

4.1 Extract Data from Excel to MATLAB

Firstly, two new files with name of input and target will be created in the workspace if MATLAB. The data of Excel will be copied into the input file and target file respectively. There is required to transpose the data in the files of MATLAB. Figure 5 shows the two files created in Workspace.

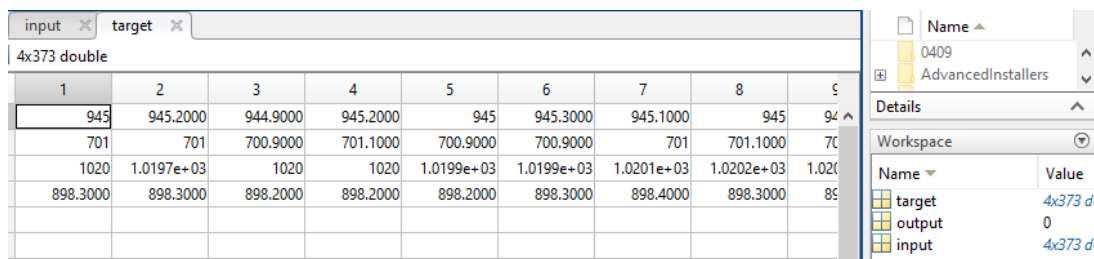


Figure 5. Two files are created in Workspace

4.2 Start the Neural Network Toolbox

To open the Neural Network toolbox is simply typing 'nntool' in command window or choosing the Neural Network toolbox icon from the APPS column above the MATLAB. Figure 6 shows the command to open the Neural Network toolbox.

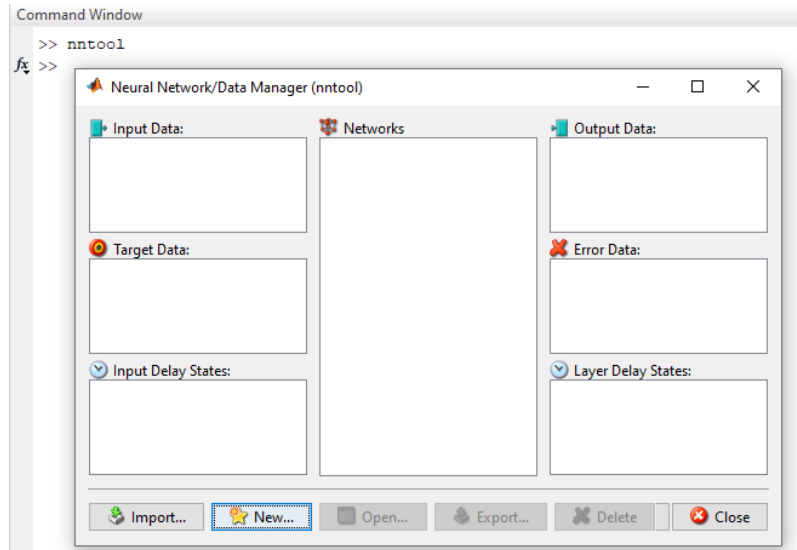


Figure 6. Command to open the Neural Network toolbox.

4.3 Start Create, Train, Configure and Validate the Network

Firstly, the data is imported to Network/Data Manager by choosing the respective files in Workspace to the desired destination and ticking the type of data wished to be imported. The network can be created by clicking the 'New' icon below. The network properties can be customized by choosing the network type and followed by input data, target data, training function, adaption learning function, performance function and the numbers of layers. Besides, each layer can be customized by choosing the number of neurons and transfer function. This also included the process of configuring the network. The way of choosing these must depend on the main target, which the output matches the target as much as possible. Figure 7 show the window of create network or Data.

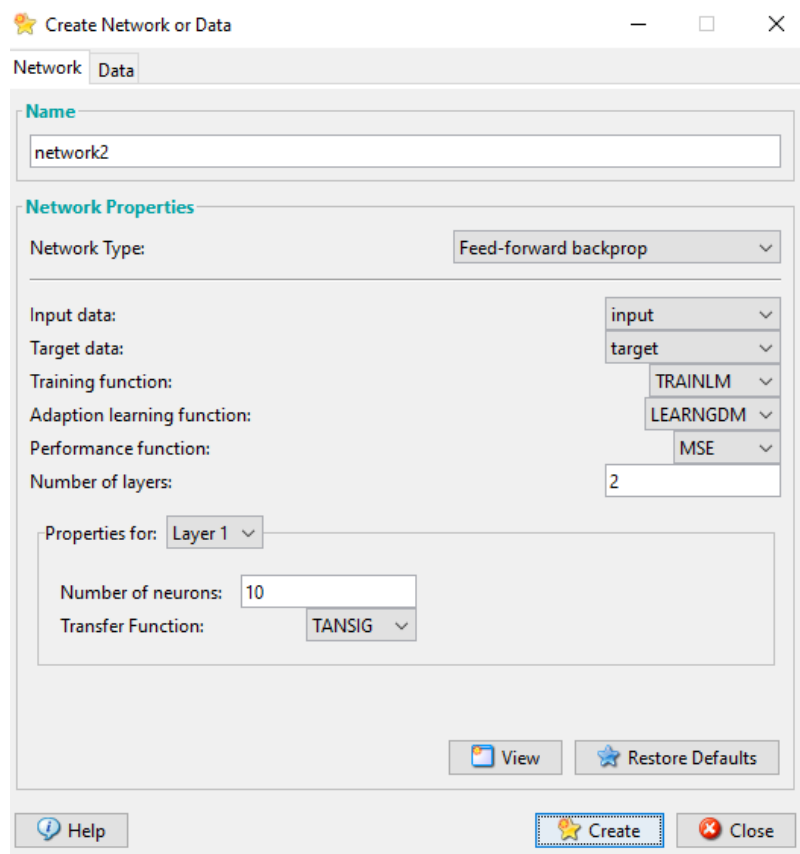


Figure 7. Window of create network or Data.

The neuron model can be viewed by clicking in the Network created. Figure 8 shows the neural model created.

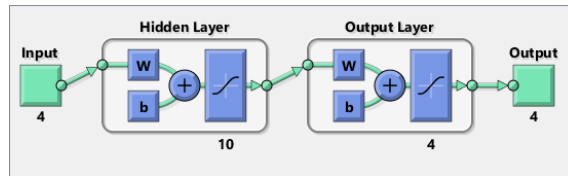


Figure 8. The neural model created.

The network can be trained by setting the input and target and adjusting desired training parameters. The progress can be viewed in the NN training tool window. Some plots like performance plot, training state plot and regression plot can be shown. For regression plot, the three plots represent the training, validation, and testing data. The dashed line in each plot represents the perfect result – outputs = targets. The solid line represents the best fit linear regression line between outputs and targets. The R value is an indication of the relationship between the outputs and targets. If R = 1, this indicates that there is an exact linear relationship between outputs and targets. If R is close to zero, then there is no linear relationship between outputs and targets. [7] Figure 9 shows the bad fitting for regression plot while Figure 10 show the good fitting for regression plot.

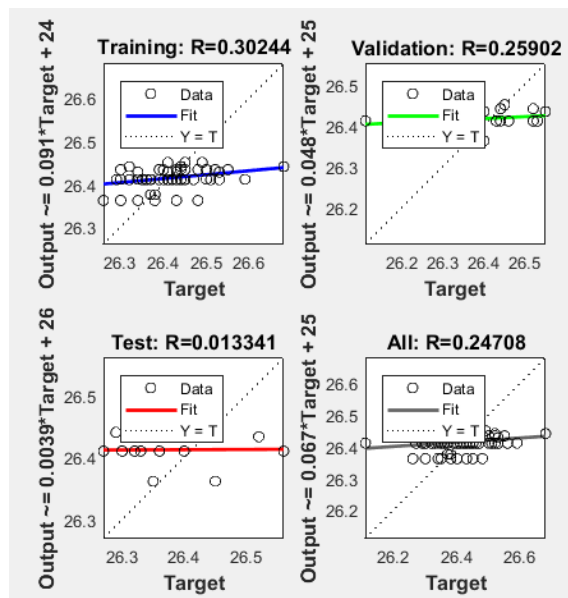


Figure 9. Bad fitting for regression plot

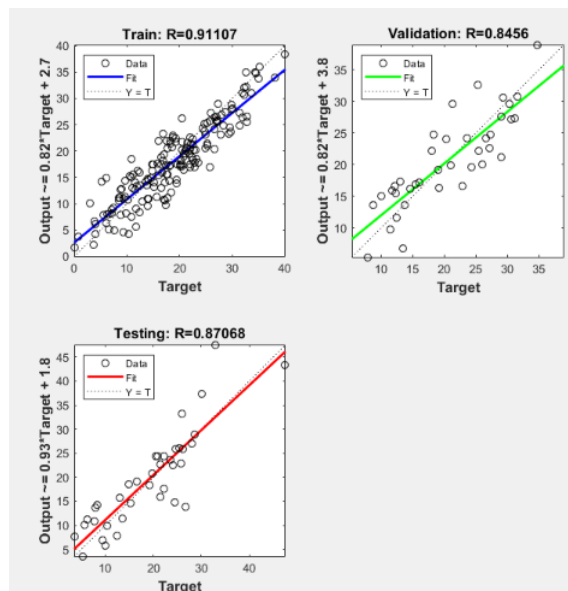


Figure 10. Good fitting for regression plot

5. Ways to Improve

There are some ways to improve the result. if the network is not sufficiently accurate, it is good to try initializing the network and the training again. Each time initialize a feedforward network, the network parameters are different and might produce different solutions. As a second approach, you can increase the number of hidden neurons above 20. Larger numbers of neurons in the hidden layer give the network more flexibility because the network has more parameters it can optimize. (Increase the layer size gradually. If you make the hidden layer too large, you might cause the problem to be under-characterized and the network must optimize more parameters than there are data vectors to constrain these parameters.)

A third option is to try a different training function. Bayesian regularization training with `trainbr`, for example, can sometimes produce better generalization capability than using early stopping. Finally, try using additional training data. Providing additional data for the network is more likely to produce a network that generalizes well to new data [7].

6. Prediction and Data Export

After the network is fully trained, the prediction can be made by importing the inputs. Then, using the simulation of trained data, select the desired input then simulate the network. Lastly, the predicted data can be exported to the Workspace by pressing export icon below the window of Neural Network/Data Manager. Figure 11 shows the example of predicted output.

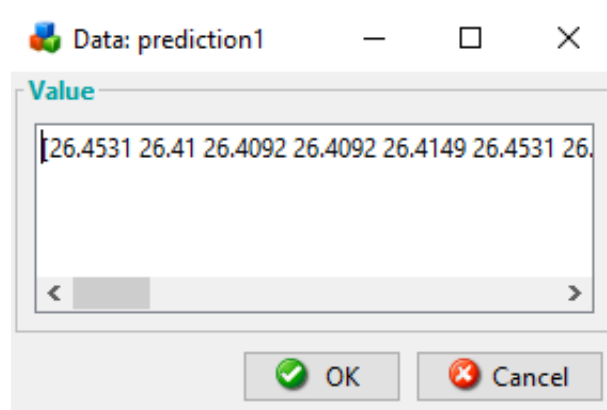


Figure 11. The example of predicted output.

7. Results

7.1 Regression Plot of Neural Network

The training of neural network stops when the Regression coefficient reached close to 1. Figure 12 shows the regression plot of neural network built.

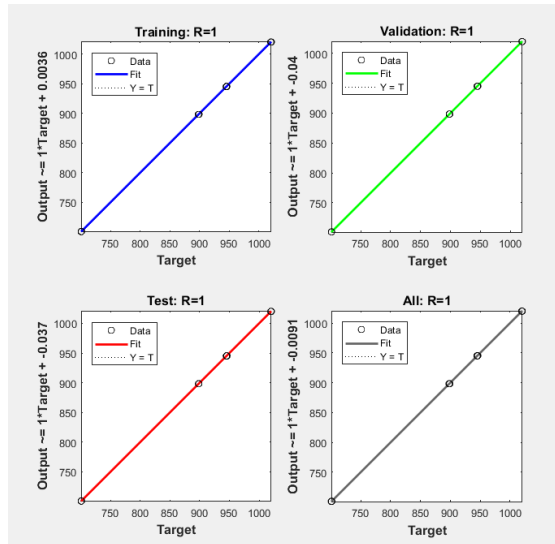


Figure 12. The regression plot of neural network built.

7.2 Testing Data

Testing data which is the FTNIR analysis result in November is imported into the neural network. Figure 13 shows the testing data folder in MATLAB workspace.

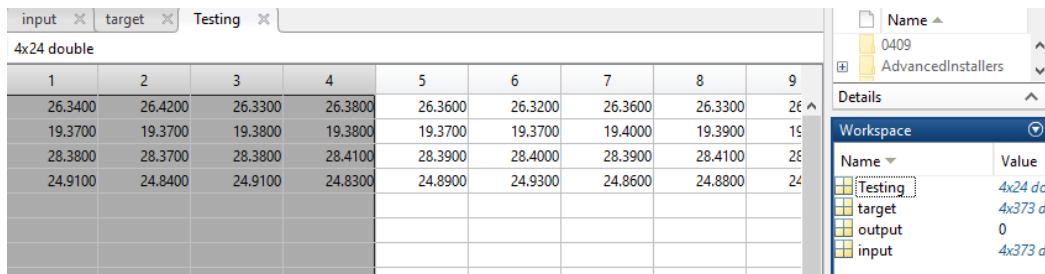


Figure 13. The testing data folder in MATLAB workspace.

7.3 Testing Output

The network simulates the testing input and predicts the testing output which is the weight of each material. Figure 14 shows the testing output obtained from the neural network.

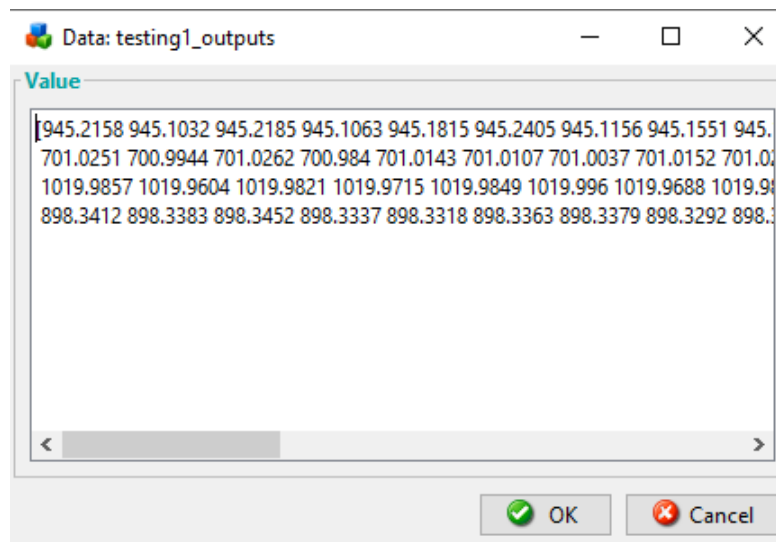


Figure 14. The testing output obtained from the neural network.

7.4 Comparison of Testing Data and Actual Data

The testing output will be compared to the actual data in November, and the percentage of errors will be calculated. Figure 15 shows the formula to calculate the percentage of error. Table 1 shows the percentage error of each material in November.

$$\%_{\text{error}} = \left| \frac{\#_{\text{experimental}} - \#_{\text{theoretical}}}{\#_{\text{theoretical}}} \right| \times 100$$

Figure 15. The formula to calculate the percentage of error.

Table 1. The percentage error of each material in November.

Material	Percentage of error (%)
Macromer	0.0073
TRIS	0.0099
DMA	0.0040
EtOH	0.0119

7.5 Predicted Input

The predicted input is the specification of FTNIR analysis to each material in percentage. Figure 16 shows the predicted input data folder in MATLAB workspace.

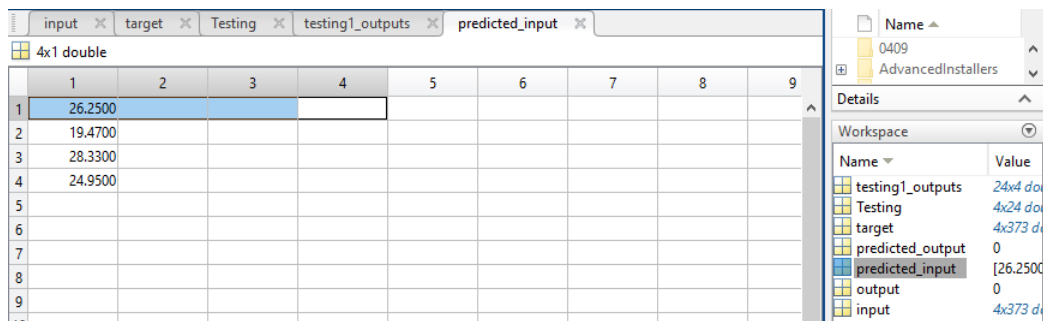


Figure 16. The predicted input data folder in MATLAB workspace.

7.6 Predicted Output

The predicted output will be weight of each material which will pass the FTNIR analysis. Figure 17 shows the predicted output produced by the neural network trained.

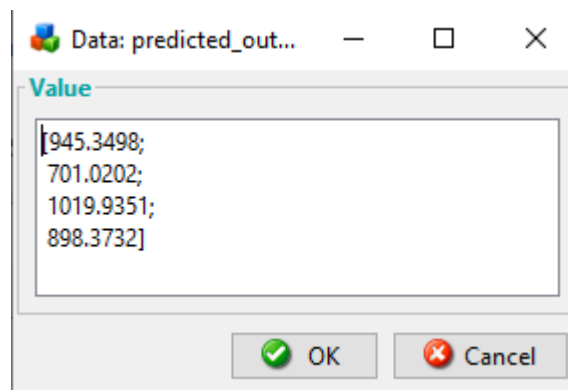


Figure 17. The predicted output produced by the neural network trained.

7.7 Details of Data Usage

All data used to build, test and predict the neural network are recorded in excel file. The data included are input data, data in waiting list, target data, testing input data, testing output data, actual testing data, percentage error, prediction input data (targeted percentage) and prediction output data (targeted weight).

8. Discussion

Two methods were used to verify the correctness of network. Firstly, it depends on the regression plot by observing the value of R. This network passed the verification with value of R of 1. Secondly, it is verified by calculating the percentage error between testing output data and actual testing output data. The output passed the verification with all percentage of error below 0.02%. However, the neural network can be further improved by adding more data and optimization

9. Conclusion

After passing the verification, the output is predicted by importing the prediction input data which is the targeted percentage of each material in FTNIR analysis. The predicted output is 945.35g for Macromer, 701.02g for TRIS, 1.19.94g for DMA and 898.37g for EtOH.

References

- [1] Mehrdad Mohammad pou, "Comparison of Lotrafilcon B and Balafilcon A silicone hydrogel bandage contact lenses in reducing pain and discomfort after photorefractive keratectomy: A contralateral eye study", June 2015
- [2] Ethem Alpaydin, "Introduction to Machine Learning", 2010
- [3] Kevin Gurney, "An introduction to Neural Network", 2005
- [4] Eric A. Wan, "Neural Network Classification: A Bayesian Interpretation", 1990
- [5] Yixing Wang, Meiqin Liu, Zhejing Bao, "Deep learning neural network for power system fault diagnosis", *Control Conference (CCC) 2016 35th Chinese*, pp. 6678-6683, 2016.
- [6] Francis Quintal Lauzon, "An introduction to deep learning", 2012
- [7] "Analyse Shallow Neural Network Performance After Training" , 2019, retrieved from: <https://uk.mathworks.com/help/deeplearning/ug/analyze-neural-network-performance-after-training.html>.