

Modelling the Cleaner Phase of a Wet High-Intensity Magnetic Separator by Employing the Levenberg-Marquardt Algorithm

M. Potgieter, A.F. van der Merwe, F.H. Conradie

Abstract— Chrome is a fundamental commodity in our everyday life, i.e. chromium is used in turbine engines, catalyst preparation, transportation and more. Conventional beneficiation thereof can be divided into comminution and concentration. The downfall to conventional beneficiation, however, entails the low efficiency of the concentration section for particle sizes inferior to 75 μm . This inefficiency builds up to a loss of approximately 25 % of valuable materials as fines in the tailings. Alternative methods, such as the Wet High-Intensity Magnetic Separator (WHIMS), were consequently invented for further beneficiation of these tailings. This paper details an investigation into the capabilities of an artificial neural network (ANN) to predict the cleaner stage WHIMS output by employing the Levenberg-Marquardt (LM) algorithm. Both multi-input-multi-output (MIMO) and multi-input-single-output (MISO) networks were employed and the topology of each optimized. The combined performance of the MISOs proved more accurate than the MIMO network predicting with overall errors of 0.6 % mean absolute error (MAE) and 0.07 root mean square error (RMSE). However, taking the mean absolute percentage error (MAPE) and the coefficient of multiple determination (R^2) into account, the MIMO network outperformed the combined MISOs, predicting with a MAPE of 3 % with a correlation of 0.993.

Keywords— Artificial neural networks, cleaner-phase WHIMS, Levenberg-Marquardt

I. INTRODUCTION

Chrome is a fundamental mineral product that is not easily replaceable in a variety of applications [1]. During conventional beneficiation thereof, approximately 25 % of valuable material is lost as fines in the tailings [2]. A piece of beneficiation equipment invented with the purpose of further beneficiating such tailings is the WHIMS. On account of three interacting forces, i.e. magnetic, gravitational and inter particulate forces, slurry can either exit a WHIMS as a tailing (non-magnetic material), a middling or a mag (magnetic material) [3]. The WHIMS utilizes an induced gravitational field that captures magnetic material in matrices while washing off non-magnetic material. The magnetic material is hereafter washed off at the point where the magnetic field does not have an influence on the particle anymore.

The measurable cleaner stage input variables include wash water flow rate, feed flow rate, slurry density, magnetic intensity, particle size, feed chrome percentage and feed chrome to iron ratio. The efficiency of the WHIMS will be defined by (i) product grade, (ii) recovery and (iii) yield.

An ANN is used to predict these outputs, due to its tailored capability to recognize patterns that other linear fitting curves will not be able to model accurately [4]. Specifically to the extraction of chromite ore, past experiments done by Reichel [5] presented results that showed that ANNs are preferred above mechanistic and statistical methods to predict the outputs based on prediction accuracy. ANNs do not predict outcomes by means of physical coding, but rather by learning characteristics of the data as a result of a pattern recognition process called training [6].

The typically used ANN is the fully connected, feed-forward layered network undergoing supervised training [7]. This type of architecture is utilized in this study. Conventional and robust algorithms include the backpropagation, Levenberg-Marquardt and scaled conjugate algorithms, with the backpropagation method by far the most commonly used algorithm [8]. The Levenberg-Marquardt algorithm was chosen due to its ability to converge quickly and in a stable manner as well as the suitability in the use of a small data set [9].

To validate the network, different performance measurements can be used, however, for networks with relatively smooth data and for the purpose of pattern recognition, it is advised to use RMSE and MAE performance measurements [10].

Each network's topology plays a cardinal role in the behavior of the network itself [6]. Numerous factors need to be taken into account when optimizing the topology, i.e. the number of hidden layers and the number of hidden neurons per hidden layer [11]. General rules of thumb exist that estimate the number of hidden neurons [12], however, additional methods to determine the optimum number of hidden neurons include Hopfield's neural network and Akaike's information criterion [11]. Plant data can also be used to validate the network's performance. Furthermore, when comparing MIMO and MISO networks, MISO networks generally outperform MIMO networks, as confirmed by Chang et al. [13].

Manuscript received 23 October, 2019. This work was supported in part by J. Cronje with Minprotech Rustenburg (johano@minprotech.co.za)

M. Potgieter was with North West University, Potchefstroom (mirgie11@gmail.com). A.F. van der Merwe is with North West University, Potchefstroom (frikkie.vandermerwe@nwu.ac.za)

II. EXPERIMENTAL METHOD

TABLE I: A LIST OF MATERIALS AND EQUIPMENT USED.

Materials	Equipment
Raw chromite ore	Containers
Water	Drying oven
Filter paper	Oven pans
WHIMS rougher product	Filter press
	Laboratory scale
	Industrial scale
	Two-way splitter
	Ten-way splitter
	WHIMS pilot plant
	Sieves and shaker
	Collection drums

Materials and equipment necessary to complete these experiments are listed in Table I. A pilot plant WHIMS with model number RW9301 102 manufactured in Australia was used to execute the experiments. In order to construct a supervised learning ANN, both input and output values need to be obtained. To quantify some of the inputs, some preparation prior to the operation of the WHIMS was required. Similar experimental steps are carried out on the product to obtain output values.

A. Sample Preparation

Raw material was received in the form of a slurry from two plants situated around Rustenburg. The first step was to leave the sample to settle in order to decant as much water as possible. A period of two hours dedicated to settling has proven to be sufficient for solid-liquid separation to take place.

The following step entailed drying of the solid material. The solids were dried in an oven at a temperature of 130 °C for a period of 24 hours. For the WHIMS operation, a minimum amount of 10 kg of solid material is required to obtain credible results. Consequently, the total weight of the dried material was split into batches with a mass slightly higher than 10 kg by using the two-way splitter multiple times.

Subsequent steps entailed determination of the particle size distribution (PSD) and x-ray fluorescence (XRF) analysis of the feed to obtain the feed particle size, chrome to iron ratio and chrome percentage. Since the bulk is now split into multiple batches with each batch representative of the bulk, material with a mass within the range of 2-3 kg was taken from one of the batches to the ten-way splitter to deliver a sample with a mass of 200-300 g. Only one of the ten parts was necessary to conduct the remainder of the sample preparation steps and the rest was added back into the batch it was taken from.

The 200-300 g sample needed to undergo separation based on particle sizes. Multiple sieves were used in the arrangement detailed in Table II. Water was used at a low flow rate for each individual sieve to aid the solid separation.

Due to fact that the previous step was conducted on a wet basis, the material needed to be dried before the material on each sieve could be weighed to produce the PSD. The material is again dried at 130 °C in the drying oven, however, 3 hours should be ample. Once the material was dry, the mass on each

TABLE II: A LIST OF SIEVES USED TOGETHER WITH THE SIEVE SIZES.

Sieve number	Sieve size (µm)
1	850
2	600
3	500
4	300
5	212
6	150
7	106
8	75
9	53
10	38

sieve was obtained, noted and separately stored to undergo XRF analysis. All the information required to construct a PSD and calculate the d_{50} (diameter at which 50 % of the material will pass through) was now known. The d_{50} was used to represent the particle size of the bulk sample. The material weighed and stored was subsequently sent for XRF analysis.

B. Operating the WHIMS

A baseline for the operating conditions of the WHIMS is provided in Table III. Step changes in each variable are implemented in order to investigate the effect of each variable on the WHIMS' operation.

TABLE III: WHIMS OPERATING CONDITIONS BASELINE.

Variable	Baseline	Unit
Wash water flow rate (single stage)	10	l/min
Wash water flow rate (double stage)	0	l/min
Feed water flow rate	7.3	l/min
Magnetic flux	13.5	A
Feed density	30	% solids (wt %)

Before the WHIMS operation could commence, a few things needed to be verified. A 2200 l tank was used to add sufficient water necessary during the WHIMS test. The level thereof needs to be verified. In addition, the amount of water needed to produce a slurry containing the mass of the test batch needed to be calculated to deliver the desired density for the run.

The rotor was started and the wash water flow rate was adjusted to the desired value prior to the run.

Next, the magnetic flux was set to the desired value and the product and tailings outlet pipes were inserted into separate bins. Hereafter, the slurries were made up to the desired feed density and added to the circulation section of the WHIMS. To ensure that the slurry was thoroughly mixed, the slurry was circulated for approximately 5 min.

After these steps have been completed, the WHIMS test can commence. The magnet was switched on and the wash water feed valve opened. The desired number of feed pipes were transferred into the hopper feeding the WHIMS. Once the circulation section containing the slurry was almost empty, water was added to flush the system.

The product analysis is similar to steps carried out for sample preparation. The chrome percentage in the product needs to be

determined to calculate the grade, recovery (Rec) and yield with (1) [5], (2) [14] and (3) [14] with m_{feed} and $m_{product}$ referring to the mass of the feed (kg) and mass of product (kg) respectively. Furthermore, $Cr\%_{product}$ and $Cr\%_{feed}$ refer to the chrome percentage within the product and the feed respectively.

$$Grade = \frac{m_{product}(Cr\%_{product})}{m_{feed}} \times 100 \quad (1)$$

$$Rec = \frac{m_{product}(Cr\%_{product})}{m_{feed}(Cr\%_{feed})} \times 100 \quad (2)$$

$$Yield = \frac{m_{product}}{m_{feed}} \times 100 \quad (3)$$

C. Repeatability

The repeatability of the experiments carried out on the WHIMS machinery can be presented by calculating the standard deviation from the mean obtained by repeated experiments. A run with 13.18 kg of solids and 60 kg of water making up a feed that is fed at 58 l/min with magnetic flux of 11 kG and 15 l/min wash water flow rate were carried out seven times. The mass of the product produced each time were measured and the standard deviation from the mean was calculated with the use of (4)-(7) [15]. Here, μ refers to the mean value of the data, $N_{duplicates}$ the number of duplicates, SE_i the squared error of the i^{th} run, σ the standard deviation and σ_{mean} the standard deviation from the mean obtained.

$$\mu = \frac{\Sigma m_{product}}{N_{duplicates}} \quad (4)$$

$$SE_i = (m_{product_i} - \mu)^2 \quad (5)$$

$$\sigma = \sqrt{\frac{\Sigma SE_i}{N_{duplicates} - 1}} \quad (6)$$

$$\sigma_{mean} = \frac{\sigma}{\sqrt{N_{duplicates}}} \quad (7)$$

D. Outliers

To remove any possible outliers within input and output data, a simple calculation yielding the upper and lower bounds was done with the use of (8)-(10) [53]. Q_3 and Q_1 refers to the upper (75%) quartile and lower (25%) quartile of the data respectively. Furthermore, IQR refers to the inter quartile range. Any value obtained from the experiments higher than the upper bound or lower than the lower bound was regarded as an outlier and discarded.

$$IQR = Q_3 - Q_1 \quad (8)$$

$$Lower\ bound = Q_1 - 1.5 \times IQR \quad (9)$$

$$Upper\ bound = Q_3 + 1.5 \times IQR \quad (10)$$

III. MODEL DEVELOPMENT AND NETWORK OPTIMIZATION

A. Model Development

The first objective is to construct a code that can create an ANN with specific characteristics. The input and output data used in this code is obtained from experiments.

Multiple programs can be utilized to construct an ANN [16]. In this case, MATLAB R2017b was used. Each ANN consists of a fully connected feedforward layered network topology with one hidden layer, using the Levenberg-Marquardt learning algorithm. All of the networks required desired output values and were therefore considered to be supervised learning networks [11]. After the network has been trained with the training data set, the network's topology needed to be optimized as the effectiveness of the ANN lies within its topology [6].

B. Network Optimization

When constructing an ANN, there are five factors that need to be considered and/or determined. These factors include: (i) selecting the number of hidden layers, (ii) determining the number of hidden neurons in each hidden layer, (iii) find an optimal solution with the aim of avoiding local minima, (iv) take convergence time constraints into account and (v) validation of network on account of overtraining [11].

1. Selecting Number of Hidden Layers

The choice of the number of hidden layers can be made by investigating the characteristics of the data. Panchal et al. [11] explained that one hidden layer is enough for conventional problems and that the additional second layer rarely improves the network's performance. A second hidden layer should however be added when dealing with data containing discontinuities [11].

2. Determining Number of Hidden Neurons

As an initial estimation of the number of hidden neurons (N_h), rules of thumb based on number of inputs (n_{in}) and outputs (n_{out}) as given in (11)-(14) can be used [11], [12]. However, this is a mere starting point. The optimum number of hidden neurons can be determined with different selection approaches, including: (i) a simple method, (ii) Hopfield neural network or (iii) Akaike's Information Criterion (AIC) [11].

$$N_h = n_{in} + n_{out} - 0.5 \quad (11)$$

$$n_{out} < N_h < n_{in} \quad (12)$$

$$N_h = \frac{2}{3}n_{in} + n_{out} \quad (13)$$

$$N_h < 2n_{in} \quad (14)$$

(i) Simple method: The configuration for a backpropagation method can be denoted as $l-m-n$, with l referring to the number of input neurons, m the number of hidden neurons and n the number of output neurons. Here m is chosen to be smaller than the number of input neurons but larger than the number of output neurons [11].

(ii) Derived from the Hopfield neural network: The Hopfield network is one of the simplest neural networks consisting of an input, output and single hidden layer with a fully connected neuron structure. It dictates that the number of hidden neurons can be set equal to the number of input neurons [11].

(iii) Akaike's Information Criterion (AIC): The AIC is a criterion based on entropy concepts that provides a measurement of the goodness of the statistical model. It describes the trade-off between bias and variance. The ANNs with different number of hidden neurons, can be ranked according to their AIC, with the lowest AIC indicating the best

configuration. The formula to calculate the AIC is given in (15) [11].

$$AIC = -2 \times \ln(\text{likelihood}) + 2 \times k \quad (15)$$

Where *likelihood* refers to maximum log likelihood of the model and *k* refers to the number of free parameters in the model. Additionally, AIC can also be calculated with (16) [17], where *n* is the number of data points.

$$AIC = n \times \ln\left(\frac{SSE}{n}\right) + 2 \times k \quad (16)$$

After the AIC has been calculated, the credibility of how good each network approximates actual data should be quantified. The first step to quantify this is to calculate how far each network's AIC is from the minimum obtained AIC as seen in (17) [11].

$$\Delta i = AIC_i - \min(AIC_i) \quad (17)$$

Next, an estimation of the likelihood of the model can be obtained with (18) [11].

$$L(\text{model}) \propto \exp(-0.5\Delta_i) \quad (18)$$

For optimal interpretation of the likelihood estimation, the values obtained in (18) should be normalized with the use of (19). This is called the Akaike's weight (AW_i) of each model [11].

$$AW_i = \frac{\exp(-0.5\Delta_i)}{\sum_{i=1}^n \exp(-0.5\Delta_i)} \quad (19)$$

The candidate models can now be filtered out by the general rule of thumb (see (20)) stating that any model with an Akaike weight greater than 10 % of the largest weight is plausible to be a candidate model [11].

$$0.1 \times \max(AW_i) < \text{Cand models} \quad (20)$$

The AIC will be considered as the determining factor when deciding on the optimum number of hidden neurons.

3. Finding an Optimal Solution with the Aim of Avoiding Local Minima

Conventional ANN architectures contain a feed forward network using a backpropagation algorithm, which is also the case in this study. However, the downfall of the backpropagation algorithm is that it may sometimes move to a local minimum and not the overall (global) minimum [18].

To overcome this occurrence, numerous studies have been conducted to propose solutions [18]. The widest used method is merely to iteratively train the network with initial random weights [19]. This, however, can be time consuming. In a paper authored by Atajulreka and Sutivong [18], the use of Resilient Backpropagation method, which is more time efficient, yet very effective, is detailed.

To simplify things, the rules of thumb are used to obtain an upper and a lower boundary for the number of hidden neurons applicable to this study. AIC results are also to be obtained within these limits, otherwise the limits are extended accordingly. Hereafter the network is trained 10 times for each number of hidden neurons until a greater certainty exists that the backpropagation algorithm focused on the global minima and not the local minima.

4. Convergence Time Constraints

In certain cases, the time the network takes to converge can play a cardinal role in the choice of number of hidden neurons

[39]. This network, however, comprises of very little data and convergence takes place in under a second for a network containing 100 hidden neurons, which therefore eliminates the need to account for a time constraint.

5. Validation of Network on Account of Overtraining

When validating an ANN, the correct error measurement should be chosen. A clear-cut theoretical approach to validate a neural network unfortunately does not exist, however, four conventionally used performance measures include the MAE, MAPE, RMSE and mean square error (MSE). Specifically, for networks containing smooth output vectors and for the purpose of pattern classification, the MAE and RMSE are usually used to validate the network's performance [10]. The MAPE, together with the coefficient of multiple determination (R^2), is used to assess the performance of the networks.

When training neural networks, it is a challenge to stop the training just before overfitting occurs and the network starts to lose its generalization abilities, yet leaving enough room for the training to learn the data set properly [20]. One approach to avoid overtraining is to investigate the number of epochs. It is treated as a hyperparameter while training the network multiple times, where-after an epoch is chosen that produces the best performance [10].

Another approach is called early stopping, where the generalization error of the network is continuously being calculated and the training is stopped when this error starts to increase [10]. Fortunately, this method is already accounted for when using MATLABTM to construct supervised feedforward neural networks [21].

IV. RESULTS AND DISCUSSIONS

A. Repeatability and Outliers

The WHIMS operation was repeated 7 times and presented a 75 g standard deviation from the mean when using 73.18 kg feed (slurry). The WHIMS therefore showed an acceptable repeatability. Furthermore, all the data were tested for outliers and none of the data points were found to be above or below the upper and lower bounds.

B. MIMO Networks

Due to the data being of a continuous nature, 1 hidden layer was used.

The general rules of thumb, together with the simple method and Hopfield's neural network results, are given in Table IV. An overall range of 3 to 14 hidden neurons was obtained.

The minimum AIC resulted in a value of -399.2 using 9 hidden neurons. On account of the rule of thumb given in (20), this network was the only viable candidate.

Furthermore, performance measurements MAPE and R^2 are investigated within the range of 3 to 14 neurons. With 9 hidden neurons, the AIC will be at its minimum, which also delivers a 3 % MAPE and 0.993 correlation. However, when only taking the MAPE together with the R^2 values into consideration, 11 hidden neurons displays the best performance. As mentioned, the AIC is chosen to be the deciding factor with regards to the optimum number of hidden neurons. Therefore, the MIMO

TABLE IV: RESULTS FOR THE NUMBER OF HIDDEN NEURONS ESTIMATED.

Rule of thumb	Minimum number of hidden neurons	Fixed number of hidden neurons	Maximum number of hidden neurons
1		10	
2	3		7
3		8	
4			14
Simple method	3		7
Hopfield		7	

TABLE V: OPTIMUM NUMBER OF HIDDEN NEURONS BASED ON PERFORMANCE MEASUREMENTS FOR ALL MISO NETWORKS.

Performance measurement	MAPE (%)	R ²	Number of hidden neurons
<u>Yield</u>			
Minimum MAPE	2	0.951	2
Maximum R ²	4	0.986	9
<u>Recovery</u>			
Minimum MAPE	1	0.983	4
Maximum R ²	4	0.998	19
<u>Grade</u>			
Minimum MAPE	1	0.980	4
Maximum R ²	3	0.997	15

network with 7 inputs, 9 hidden neurons and 3 output neurons is regarded as the optimum network structure according to AIC.

Hereafter, local minima is tested. After iteratively training the network each time with randomized weights, the smallest AIC obtained remains -399.2 with the use of 9 hidden neurons. It can therefore be stated with greater certainty that the minimum originally found represents a global minimum.

Taking local minima into account, the optimum MIMO network, therefore, contains 1 input layer with 7 neurons, 1 hidden layer with 9 hidden neurons and 1 output layer with 3 neurons. The performance of this network can be seen in Fig. 1.

This network presents an overall performance of 3 % MAPE with an R² of 0.993. The grade prediction is more accurate than the predictions for recovery and yield. This could be on account of less variance found in grade data.

C. MISO Networks

Three different types of MISO networks were constructed. All seven inputs were used in each MISO to investigate the effect that each has on grade, recovery and yield individually.

Similar to MIMO networks, the data remains to be of a continuous nature as it is the same data used. Therefore, 1 hidden layer was used for all the MISO networks.

The general rules of thumb, together with the use of the simple method and Hopfield’s neural network, resulted in an overall range of 1 to 14 hidden neurons. The minimum AIC value for the yield, recovery and grade output MISOs resulted in values of -235 with 14 hidden neurons, -549 with 5 hidden

neurons and -216 with 21 hidden neurons respectively. These networks were the only viable candidates. The optimum number of hidden neurons based on the MAPE and R² performance measurements are given in Table V.

Similar to the MIMO network, each MISO was trained 10 times, starting each time with random weights for neurons 1 to 21 to ensure that the values obtained are not representative of local minima. The hidden neuron range is extended to investigate up until 21 neurons on account of the optimal AIC found at 21 for the grade network. The minimum AICs obtained after iteratively training the networks are -502.5, -520 and -405 for the yield, recovery and grade networks respectively. Comparing these values to the values initially obtained, both the yield and the grade networks obtained smaller AICs at 6 and 8 hidden neurons. This can be an indication that the grade and yield networks initially converged to local minima. In conclusion, the optimal network structures for each MISO can now be given with greater certainty as 6, 5 and 8 hidden neurons for the yield, recovery and grade neural networks respectively. Each network’s performance is presented in Table VI. The performance graphs are shown in Fig. 2.

TABLE VI: PERFORMANCES OF EACH OPTIMUM MISO NETWORK.

Output	MAPE (%)	R ²
Yield	3	0.971
Recovery	5	0.952
Grade	3	0.981

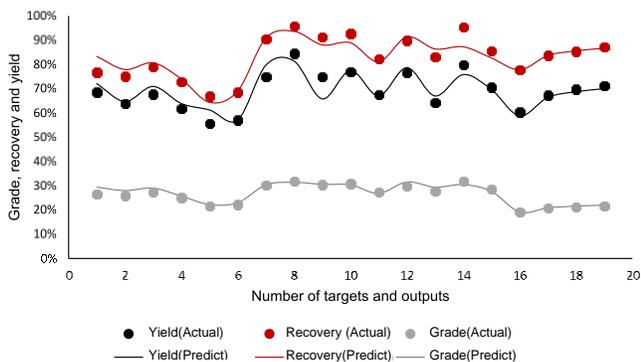


Fig. 1. Fit of optimum MIMO neural network with actual experimental output data.

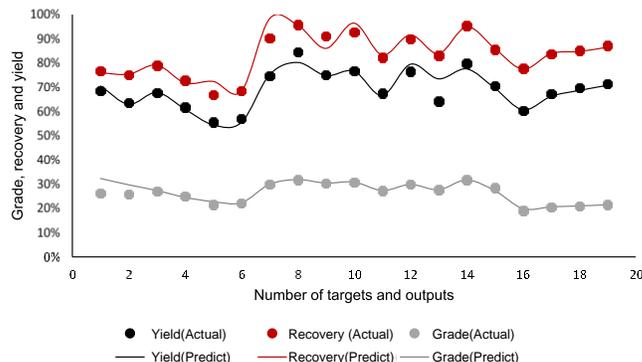


Fig. 2. Fit of optimum MISO networks to experimental data.

TABLE VII: COMPARISON OF PERFORMANCE MEASUREMENTS MIMO NETWORKS WITH MISO NETWORKS.

Output	MAE		RMSE		MAPE		R ²	
	%				%			
	M	S	M	S	M	S	M	S
Yield	0.1	0.1	0.03	0.02	3		0.97	
Recovery	0.4	0.3	0.03	0.05	5		0.95	
Grade	0.5	0.2	0.01	0.004	3		0.98	
Overall	1.0	0.6	0.07	0.07	3	3.7	0.99	0.97

A. Comparison Between MIMO and MISO Networks

The optimum MIMO and MISO network performances are presented in Table VII. When considering MAE and RMSE, the MISO networks primarily outperforms the MIMO, confirming findings from literature [13]. However, when considering the MAPE and R² values, the MIMO network perform better than the combined performance of the MISO networks.

V. VALIDATION OF NETWORK TOPOLOGY WITH PLANT DATA

In order to inspect the validity of the chosen network and its topology, three months' plant data was used as input and output values. The combined MAPE for the MIMO and MISO networks were 10 % and 4 % respectively. The MIMO and MISO obtained an R² of 0.97 and 0.79 respectively.

VI. CONCLUSIONS AND RECOMMENDATIONS

The findings correspond with work done by Reichel [7] that showed the suitability of an ANN to model the rougher phase of a WHIMS. It is also fit to use an ANN to model the cleaner phase of the WHIMS as it models experimental data with overall errors of 0.2 % MAE, 0.02 RMSE and 3.7 % MAPE while displaying a correlation of 0.97 when using MISO networks. MISO neural networks are deemed optimum between MIMO and MISO networks considering performance criteria MAE and RMSE. Contrastingly, the MIMO network slightly outperformed the MISOs when taking the MAPE and R² into consideration. Optimum number of hidden neurons for the MISO networks are 6, 5 and 8 hidden neurons for yield, recovery and grade respectively, while the MIMO network performed optimally with 9 hidden neurons.

Some recommendations to improve on the accuracy of the model are (i) to obtain a larger dataset to allow training to take place with more data points, (ii) a more in depth study to ensure that all possible local minima are avoided and (iii) to combine the performance measurements with the AIC results to make a decision regarding the optimum number of hidden neurons

VII. REFERENCES

- [1] J. O. Nriago and E. Nieboer, *Chromium in the natural & human environments*. Wiley, 1988.
- [2] N. Gence, "Beneficiation of elazig-kefdag chromite by multi-gravity separator," *J. Eng. Environ. Sci.*, vol. 23, pp. 473–475, 1999.
- [3] J. A. Oberteuffer, "Magnetic Separation: A Review of Principles, Devices, and Applications," *IEEE Trans. Magn.*, vol. 10, no. 2, pp. 223–238, 1974.
- [4] C. Stergiou and D. Siganos, "Neural Networks." 2019.
- [5] C. R. M. Reichel, A. F. Van Der Merwe, and J. Cronje, "A Predictive Model for Wet High Intensity Magnetic Separator (WHIMS) using Artificial Neural Networks," 2018.
- [6] R. Beresford and S. Agatonovic-Kustrin, "Basic concepts of artificial neural network (ANN) modeling and its application in pharmaceutical research," *J. Pharm. Biomed. Anal.*, vol. 22, no. 5, pp. 717–727, 2000.
- [7] R. Aggarwal and Y. Song, "Artificial neural networks in power systems. Part I: General introduction to neural computing," *Power Eng. J.*, vol. 11, no. 3, pp. 129–134, 2005.
- [8] Ö. Kişi, "Streamflow Forecasting Using Different Artificial Neural Network Algorithms," *J. Hydrol. Eng.*, vol. 12, no. 5, pp. 532–539, 2007.
- [9] H. Yu and B. M. Wilamowski, "Levenberg–Marquardt Training," *Intell. Syst.*, pp. 12-1-12–16, 2019.
- [10] J. M. Twomey and A. E. Smith, "Validation and verification," *Artif. Neural Networks Civ. Eng. Fundam. Appl.*, no. January 1995, pp. 44–64, 1997.
- [11] G. Panchal, A. Ganatra, Y. P. Kosta, and D. Panchal, "Behaviour Analysis of Multilayer Perceptrons with Multiple Hidden Neurons and Hidden Layers," *Int. J. Comput. Theory Eng.*, vol. 3, no. 2, pp. 332–337, 2011.
- [12] S. Trenn, "Multilayer perceptrons: approximation order and necessary number of hidden units," *IEEE Trans. Neural Networks*, vol. 19, no. 5, pp. 836–844, 2008.
- [13] F. J. Chang, Y. M. Chiang, and L. C. Chang, "Multi-step-ahead neural networks for flood forecasting," *Hydrol. Sci. J.*, vol. 52, no. 1, pp. 114–130, 2007.
- [14] B. A. Wills and T. Napier-Munn, *Wills' Mineral Processing Technology*, no. October. 2005.
- [15] C. Deziel, "How do I calculate repeatability?," *Sciencing*, 2018. [Online]. Available: <https://sciencing.com/do-calculate-repeatability-7446224.html>. [Accessed: 10-Oct-2019].
- [16] R. M. Yoo, H. Lee, K. CHow, and H. S. Lee, "Constructing a Non-Linear Model with Neural Networks for Workload Characterization," *IEEE*, pp. 150–160, 2006.
- [17] H. Akaike, "Fitting Autoregressive Models for Prediction," *Ann. Inst. Stat. Math.*, vol. 21, pp. 243–247, 1969.
- [18] A. Atakulreka and D. Sutivong, "Avoiding local minima in feedforward neural networks by simultaneous learning," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 4830 LNAI, pp. 100–109, 2007.
- [19] M. S. Iyer and R. R. Rhinehart, "A Method to Determine the Required Number of Neural- Network Training Repetitions," *IEEE Trans. Neural Networks*, vol. 10, pp. 427–432, 1999.
- [20] J. Brownlee, "A Gentle Introduction to Early Stopping to Avoid Overtraining Neural Networks," *Machine Learning Mastery*, 2019. [Online]. Available: <https://machinelearningmastery.com/early-stopping-to-avoid-overtraining-neural-network-models/>.
- [21] (MathWorks), "Improve Shallow Neural Network Generalization and Avoid Overfitting," 2019. [Online]. Available: <https://www.mathworks.com/help/deeplearning/ug/improve-shallow-neural-network-generalization-and-avoid-overfitting.html>. [Accessed: 08-Oct-2019].