Article:
Hemingray, C and Westland, S orcid.org/0000-0003-3480-4755 (2016) A novel approach to using neural networks to predict the colour of fibre blends. Coloration Technology, 132 (4). pp. 297-303. ISSN 1472-3581

https://doi.org/10.1111/cote.12220

A novel approach to using neural networks to predict the colour of fibre blends

Caroline Hemingray[a] and Stephen Westland[b]

[a]School of Art, Architecture and Design, Leeds Beckett University, Leeds, UK

[b]School of Design, University of Leeds, Leeds, UK

This work is concerned with the colour prediction of viscose fibre blends where two conventional prediction models (Stearns-Noechel and Friele) and two neural network models were compared. A total of 333 blended samples were prepared from eight primary colours including two-, three- and four-colour mixtures. The performance of the prediction models was evaluated using 60 of the 333 blended samples. The other 273 samples were used to train the neural networks. It was found that the performance of both the neural networks exceeded that of both conventional prediction models. When the neural networks were trained using the 273 training samples the average CIELAB colour differences (between measured and predicted colour of blends) for the 60 samples in the test set were close to 1.0 for the neural network models. When the number of training samples was reduced to only 100, the performance of the neural networks degraded but still gave lower colour differences, between measured and predicted colour, than the conventional models. The first neural network was a conventional network similar to that which has been used by several other researchers; the second neural network was a novel application of a standard neural network where rather than using a single network, a set of small neural networks was used, each of which predicted reflectance at a single wavelength. The single-wavelength neural network was shown to be more robust than the conventional neural network when the number of training examples was small.

Introduction

There has been a long history of work to derive simple models that can accurately predict the reflectance factors of mixtures of coloured fibres. The simplest model is given by Equation 1 whereby the reflectance of the mixture $R_M(\lambda)$ at wavelength λ is simply the weighted average of the reflectance factors of the two components $R_1(\lambda)$ and $R_2(\lambda)$, thus

$$R_M(\lambda) = c_1R_1(\lambda) + c_2R_2(\lambda) \qquad (1)$$

where c_1 and c_2 are the fractional amounts of the two components and $c_1 + c_2 = 1$. Equation 1 can easily be extended for mixtures of more than two components. However, it is commonly acknowledged that Equation 1 is a poor predictor of the reflectance of a fibre blend and most approaches tend to seek functions that can transform the reflectance so that an additive model (such as Equation 1) can be improved.

In 1944 an empirical method for predicting the colour of wool fibre blends was proposed by Stearns and Noechel [1]. The reflectance factors were transformed using a function f_{SN} which can then be used additively to predict blending (equation 2). Thus,

$$f_{SN}(R) = (1\text{-}R)/(bR - 0.01b + 0.01) \qquad (2)$$

where the dimensionless constant b was found to be 0.15 for fine wool, but was expected to vary for different fibre types.

In 1952 Friele [2] introduced a less empirical model using a function f_F that was related to the Kubelka-Munk theory, thus,

$$f_F(R) = \exp(-\sigma(1\text{-}R)^2/2R) \qquad (3)$$

where σ is the Friele parameter and R is the reflectance factor. According to Philips-Invernizzi *et al.* different values of the Friele parameter should be assigned for different fibre types and they cite example values in the range 0.094 to 0.300 [3]. In one study of Friele's equation, Philips-Invernizzi *et al.* used 17 dyed cotton fibres as primaries and attempted to match 28 targets, defined by CIELAB values, using three-component blends [3]. The average CIELAB error of the matches was about 5.7 CIELAB units. It is possible that the constraint of using only three components in the blends could have been a factor in the relatively large errors obtained.

Philips-Invernizzi *et al.* also explored the Stearn-Noechel model [4]. They used thirteen coloured fibres as primaries and attempted to predict the colour of 234 two-component blends. For cotton blends they recommended a b value of 0.109. In this work the coefficient b was also optimised in a number of ways. Excellent results were reported when b was optimised separately for each blend. It was noted that this was not a practical approach, but it does hint at the possibility of improved predictions if b could be shown to vary systematically with, for example, the CIELAB coordinates of the blend so that simple models that predict b based on the colour of the blend could be derived. Philips-Invernizzi *et al.* also considered an approach where b was optimised for each wavelength and this produced a systematic dependent of b on wavelength [4], thus

$$b = (0.12\lambda + 42.75)/1000 \qquad (4)$$

The traditional single-constant Kubelka-Munk model has been shown to be ineffective for predicting the colour of fibre blends [5] but it has been demonstrated that a two-constant model can be effective [6]. In one study, nine coloured fibres were denoted as primaries and were used to generate 42 arbitrary blends. A two-constant Kubelka-Munk model was used to predict formulations for each of the 42 blends and the average colour difference between the measured blends and the predicted formulations was 1.6 CIELAB units [5]. Using the same data, a Friele model (using a parameter σ of 0.22-0.24) exhibited an average colour difference of 2.7 CIELAB units and a Stearns-Noechel model (using a parameter b of 0.09) exhibited an average colour difference of 2.4 CIELAB units.

Recently, Song *et al.* developed a spectrophotometric colour-matching algorithm based on the Stearns-Noechel model and used it to predict recipes for 48 viscose blends (36 three-component blends and 12 four-component blends). The constant b was optimised. For three-component blends the best performance was found when b = 0.09 and the average colour difference between the average and predicted colours was 0.56 CIELAB units. For four-component blends the optimum value of b was in the range 0.03 to 0.06 and the average colour difference was 1.02 CIELAB units [7].

Within the field of neural networks, Thevenet *et al.* used neural networks to model colour changes that take place after the spinning process [8], and Furferi and Governi used a neural network to correct the spectral output from a Stearns-Noechel model [9]. However, the general use of neural networks to model transformations between colour spaces is more than twenty years old.

Most colour applications of neural networks are based on multi-layer perceptron feed-forward networks. These networks are described in detail by Shamey and Hussain but essentially map an input vector to an output vector via a hidden layer of processing units [10]. The values of the weights (free parameters) in the network are determined by optimisation using a training set of input-output examples. Bishop *et al.* used a neural network to predict dye concentrations (for a three-dye system) from CIELAB coordinates [11]. However, apart from in special cases, it is almost always better [12] to use the neural network to predict colour from recipes (analogous to the way in which Kubelka-Munk models, for example, operate) rather than attempting to predict recipes from colour directly. Westland used a neural network to predict spectral reflectance for mixtures of six printing inks printed on white card [12]. A total of 123 samples were used to train the neural network and the performance of the network was then tested on 40 additional samples. The network outperformed a two-constant Kubelka-Munk model when the number of units in the hidden layer was 7 and when all of the available training data were used. It was shown that performance deteriorated as the number of training samples was reduced [12].

One problem with most of this earlier work with neural networks is that the number of training cases (examples that are used to optimise the weights in the network) is less than or equal to the number of weights (free parameters in the network) in the network. Unless the number of training examples is large, the network may over-train (this is where it fits the training data well but has poor generalisation). Sarle recommends that in general there should be at least 30 times as many training cases as there are weights in the network [13]. However, Sarle also states that in some cases a ratio of 30:1 for training cases to weights may not be sufficient but that, in some situations, a ratio of 5:1 may suffice. Sarle also notes that "training a network with 20 hidden units will typically require anywhere from 150 to 2500 training cases if you do not use early stopping or regularization" [13]. Consider a network that attempts to predict spectral reflectance at 31 wavelengths for mixtures of eight colorants and which has 7 units in the hidden layer; the number of free parameters in the network would be $(9 \times 7) + (8 \times 31) = 311$ (each of the 7 hidden units receives weighted input from each of the 9 input units and the bias unit and each of the 31 output units receives weighted input from each of the 7 hidden units and the bias unit). Even using Sarle's lower limit of a ratio of 5:1 this would require over 1500 training cases [13].

The problem of over-training when using back-propagation neural networks for colour problems has been previously noted [14]. Table 1 reviews some colour-related uses of neural networks and shows that the number of training cases is almost always inadequate compared with the size of the network. Indeed, in only three cases [18, 20, 22] did the number of training examples exceed the number of weights in the network and these were both for colorimetric problems rather than a spectral problem. When spectral data are considered and either the input or output layer contains (typically) 31 units and the number of weights in the network is necessarily large, it is very difficult to satisfy Sarle's condition [13] that the number of training examples should be much larger than the number of weights in the network.

Another potential problem is that using networks to predict spectral data from colorant concentrations, whereby the network is treated as a 'black box', is arguably naïve. The reason for this is that almost a century of research into colour prediction (using, for example, Kubelka-Munk theory) reveals that the reflectance factors at one wavelength can be computed independently of those at other wavelengths (with notable exceptions such as fluorescence).

The novel feature of the work in this paper is to use a separate neural network for each wavelength; we will refer to this as the *single-wavelength neural network* model. Thus, rather than using a single

network to predict 31 outputs, 31 networks are used to each predict a single output.

Table 1 Examples of network sizes (and number of training samples) in the literature

Size of network	Number of weights	Number of training examples	Source
3-8-16-3	227	30	Bishop *et al.* [11]
3-41-3	290	1613	Tominaga [18]
4-10-31	391	283	De M Bezerra and Hawkyard [21]
6-7-31	297	123	Westland [12]
6-5-31	221	75	Westland *et al.* [15]
3-16-16-16-16	880	15	Dupont [19]
3-18-3	129	166	Cheung *et al.* [20]
31-25-31	1606	400	Furferi and Governi [9]
3-5-3	38	300	Jawahar *et al.* [22]
31-4-3	143	26	Sennaroglu *et al.* [23]
8-10-10-4	244	196	Kan and Song [16]
3-10-15-5	285	130	Hung *et al.* [17]

This representation has a substantial impact on the ratio of training examples to weights. Consider the simple example described earlier where there are 8 colorants and the spectra are to be predicted at 31 wavelengths, and imagine that the number of hidden units in the network is 7. The number of weights in the network is 9 × 7 + 8 × 31 = 311. Imagine now that there are 200 training examples. It is clear that the number of training examples (200) is inadequate for the size of the network (311 weights) if we accept Sarle's recommendations. Now consider what happens if 31 separate networks are used, each one predicting the reflectance factor at a single wavelength. Assuming that each of the small networks will still have 7 hidden units, the number of weights in each of the networks is now 9 × 7 + 8 × 1 = 71. However, each network still has 200 training examples and therefore the ratio of training examples to weights is now approximately 3:1. In practice, the benefit of predicting reflectance at each wavelength separately will be even greater than in this example since the number of hidden units required to predict reflectance at a single wavelength is likely to be fewer than those required to predict reflectance at 31 wavelengths.

In this paper, a set of fibre blends of known composition and known spectral reflectance is used to evaluate this new neural approach. For comparison, a standard neural network approach and variants of the Friele and the Stearns-Noechel models are also implemented to enable direct comparison.

Experimental

Sample Preparation

Pre-coloured spun-dyed viscose staple fibre (1.7 dtex with fibre length of 50 mm) was provided by Lenzing AG [24]. To produce this spun-dyed fibre, mixtures of pigments were added to the viscose spin solution during the manufacture of the fibre to become physically entrapped as the fibre is stretched and dried during the extrusion process.

Spun-dyed coloured fibre was provided by Lenzing [24] in eight different colours which will be referred to in this paper as primaries. The number of primaries used was somewhat arbitrary and was determined by the fibre made available for the project. The colours chosen were also randomly selected. Table 2 shows the CIELAB colour coordinates for each of the primaries that were used and Figure 1 shows the position of these primaries in CIELAB colour space.

Table 2 Examples of network sizes (and number of training samples) in the literature

Primary Number	L*	a*	b*
1	95.72	-0.21	1.56
2	13.87	-0.03	-0.05
3	71.36	35.82	1.06
4	74.34	28.66	57.84
5	82.67	14.42	79.38
6	87.84	4.50	80.48
7	70.52	2.13	-23.88
8	42.48	27.89	-26.90

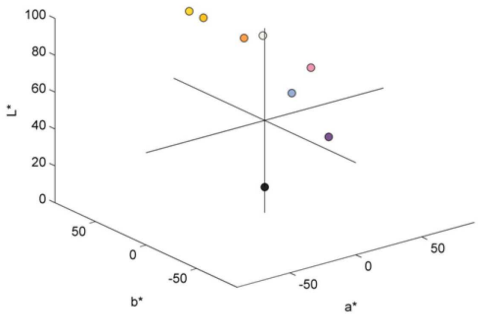

Figure 1 The position of the eight primaries in CIELAB colour space. The colours of the symbols show the approximate colours of the primaries.

A total of 333 samples (comprising of the 8 primaries and 325 blended viscose fibre samples prepared using combinations of 2, 3, and 4 primaries) were prepared. There were 52 (16%) 2-primary, 116 (36%) 3-primary and 157 (48%) 4-primary mixtures.

In making the fibre blends, careful preparation and handling was essential for accurate colour measurement [6, 25]. The fibre was first conditioned (room temperature of 20°C and relative humidity of 65 ± 5 %) for 48 hours and was then weighed to 2 decimal places. The fibre was then opened by hand, taking care to distribute the different colours of the blend evenly, in a sample area of 210 mm x 148 mm before being passed three times through a Tathams small-sample carding machine. After the first and second pass through, the parallel batt was rotated 90°. The carding machine was cleaned of loose fibre between every sample in order to minimise fibre cross-contamination between samples.

After carding, the spectral reflectance of each blended sample was measured using a Spectraflash® 600 PLUS spectrophotometer (100% UV, specular component excluded, large aperture view, optical geometry of d/8°). Reflectance factors at 35 wavelengths (10-nm intervals between 360 nm and 700 nm) were then exported into MATLAB for data analysis, comparison and conversion to CIELAB (D65 illuminant, 1964 CIE standard observer) coordinates. Figure 2 shows the positions of the 333 samples in the a*-b* plane of CIELAB colour space (the colours of the data points represent sRGB values of the samples).

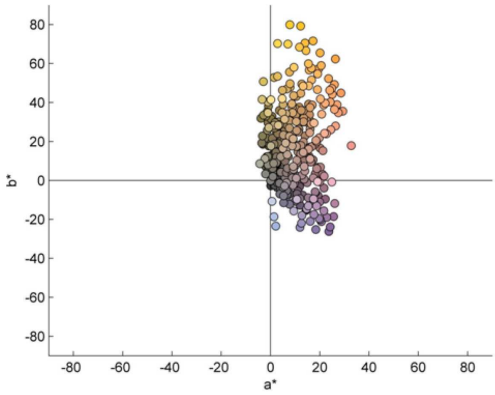

Figure 2 The position of the 333 samples in the a*-b* plane of CIELAB colour space. The colours of the symbols show the approximate colours of the samples.

Evaluation of models

The 333 samples were randomly split into two sets, a training set (273) and a test set (60). The training set was used to determine the parameters of the Stearns-Noechel and the Friele models and to optimise the weights in the artificial neural networks. Therefore, the same set of the samples (the test set) was used to evaluate performance of all of the models.

The Stearns-Noechel model was implemented using Equation 2 and the value of b was optimized to give the smallest value of CIELAB ΔE for the training set. The Friele model was implemented using Equation 3 and the value of σ was also optimised to give the smallest value of CIELAB ΔE for the

training set. Optimisation was performed by exhaustive search over the range 0.01-0.4 in steps of 0.01 for both parameters.

The neural network architecture used in this study is known as a multi-layer perceptron (MLP) network and was implemented using MATLAB (2012a) and its associated neural network toolbox [27]. This architecture has an input layer of processing units, a hidden layer of processing units, and an output layer of processing units. The number of units in the input layer was determined by the dimensionality of the input vector (in this case it is 8 since there are 8 primaries). The sum of the input vector in every case was 1 because the input unit for each primary was the proportional amount of that primary in the blend. The number of units in the output layer was determined by the dimensionality of the output vector (this was 35 in the case of the standard neural network and 1 in the case of the singe-wavelength neural network). The reflectance factors at each wavelength were represented in the range 0-1. The number of units (N) in the hidden layer was varied. The standard neural network attempted to find a relationship between the 8-dimensional vector of colorant proportions (input) and the 35-dimensional vector of corresponding reflectance factors (output). The single-wavelength neural network used 35 separate networks, each one predicting a single output corresponding to one of the 35 wavelengths for the reflectance factors; each single-wavelength network therefore had 8 input units, N hidden units, and 1 output unit.

In an MLP network each of the units of a layer performs a transfer function on its input to achieve an output. The input to each unit in a layer is the weighted sum of the outputs of the units in the previous layer. The transfer function for the hidden units was the tan-sigmoid function. Specifically, for an input value x, the output $O(x)$ is given by the following expression:

$$O(x) = (e^{x} - e^{-x})/(e^{x} + e^{-x}) \qquad (5)$$

The transfer function of the input layer and the output layer is a linear identity function thus:

$$O(x) = x \qquad (6)$$

When each network is created the values of the weights are initially set to be random numbers. The weights are changed during the training process to reduce the error between the target output vectors and the actual output vectors. In this paper the Levenberg-Marquardt training method [26] was used and the network was trained until one of the progress parameters was fulfilled (Table 3). The initial settings for the adaption rate (MU) was 0.001 (the default setting in the MATLAB toolbox).

Table 3 The progress parameters for the standard neural network using the built-in MATLAB toolbox settings

Progress parameter	Stopping Criteria
Epoch	Maximum of 1000
Time	Unlimited
Performance	Unlimited
Gradient	Minimum of $1\times e^{-5}$
MU	Maximum of $1\times e^{10}$
Validation checks	Maximum of 6

The networks were trained with 1, 3, 5, 7, 9, 10, 15, 20, 25, 30, 35 and 40 hidden units. Although there were several criteria for the training (weight optimisation) of the network to stop, generally, the training would most commonly stop when the minimum gradient ($1\times e^{-05}$) was reached.

The single-wavelength neural network used 35 separate networks, each one predicting a single output corresponding to one of the 35 wavelengths for the reflectance factors. Each network therefore had 8 input units, N hidden units, and 1 output unit and was implemented using MATLAB's 'Neural Network Toolbox™' [27]. Figure 3 illustrates the network topography with 2 hidden units for one particular wavelength (in this case 360 nm).

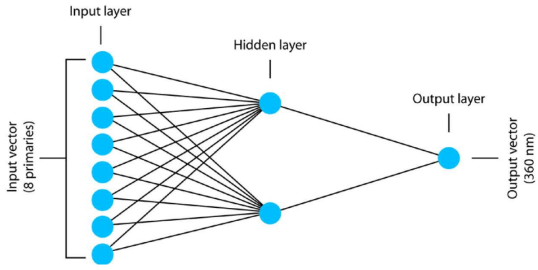

Figure 3 Schematic diagram to show the single-wavelength network structure (in this case for 360 nm)

The performance of all of the models was evaluated by calculating mean CIELAB colour differences. When this was done for the training set of samples it is referred to as a *training error* and when this was done for the test set of samples it is referred to as a *testing error*.

Results

Figure 4 shows how the training error changed with variation in the parameter b for the Stearns-Noechel model. Optimal performance (mean CIELAB ΔE = 2.92) was obtained with b = 0.10. When the Stearns-Noechel model (b = 0.10) was run with the testing samples the testing error was 3.05 CIELAB ΔE.

Figure 5 shows how the training error changed with variation in the parameter σ for the Friele model. Optimal performance (mean CIELAB ΔE = 2.13) was obtained with σ = 0.23. When the Friele model (σ = 0.23) was run with the testing samples the testing error was 2.42 CIELAB ΔE.

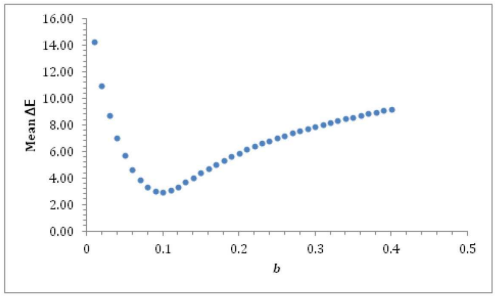

Figure 4 The training error for the changed Stearns-Noechel model with variation in the parameter b. Optimal performance was obtained with a value b = 0.10 (mean CIELAB ΔE = 2.92).

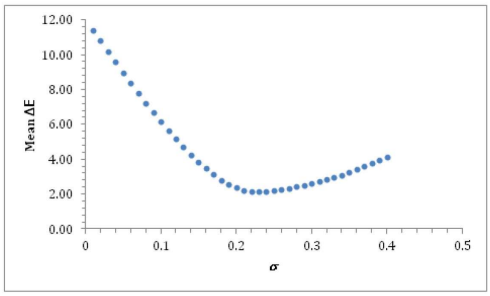

Figure 5 Training error for the Friele model with variation in the parameter σ. Optimal performance was obtained with a value σ = 0.23 (mean CIELAB ΔE = 2.13).

Figures 6 and 7 show the training and testing error for the standard and the single-wavelength neural network respectively. In each case the network was trained using the full 273 set of training data available and tested using the 60 samples of the test set; the errors reported are the mean CIELAB ΔE values. The training error reduces in each case as the number of hidden units is increased; however, the testing error reaches a minimum and after this increasing the number of hidden units tends to increase the testing error. This behaviour is typical of neural networks and indicates that above the optimal performance the network is being over-trained, resulting in poorer generalisation. The optimal testing error for the networks was 1.10 (with 15 hidden units) and 1.05 (with 3 hidden units in each of the 35 networks) for the standard and single-wavelength networks respectively.

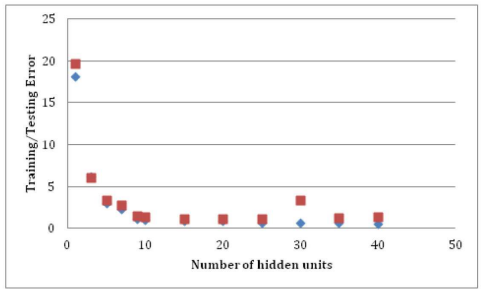

Figure 6 Mean CIELAB training error (blue diamonds) and testing error (red squares) for the standard ANN with different numbers of hidden units

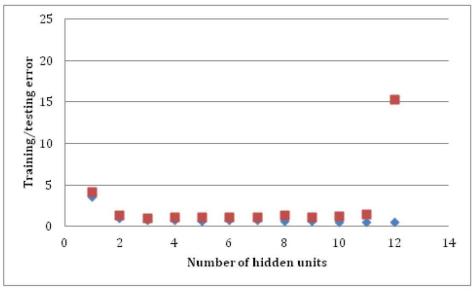

Figure 7 Mean CIELAB training error (blue diamonds) and testing error (red squares) for the single-wavelength ANN with different numbers of hidden units

Table 4 shows the optimal results (CIELAB error) for each of the systems that were investigated.

Table 4 Training and Test results – CIELAB colour differences

	Training average ΔE	max ΔE	Testing average ΔE	max ΔE
Friele	2.13	6.08	2.42	6.05
Stearns-Noechel	2.92	7.08	3.05	6.55
Standard ANN	0.88	5.39	1.10	4.44
Single-wavelength ANN	0.80	6.66	1.05	3.52

In the context of the earlier comments about the impact of the number of training examples for a neural network, it is interesting to consider how the networks would perform if fewer training samples were available. The training set was randomly sub-sampled to reduce it to 100 but the test set was not changed. Standard and single-wavelength networks were trained with varying numbers of hidden units, as before, and the performance of the best-performing networks is summarised in Table 5.

In Table 5 it is evident that the testing performance of both networks degrades as the number of training samples is reduced; however, the single-wavelength neural network is more robust and the mean CIELAB testing error rises from 1.05 to 1.39 whereas for the standard neural network the testing error rises from 1.10 to 1.86.

Table 5 Training and Test results (CIELAB colour differences) for the neural networks when only 100 training samples were used. The optimal performance shown was obtained with 15 hidden units for the standard ANN and 2 hidden units for the single-wavelength ANN.

	Training		Testing	
	mean ΔE	max ΔE	mean ΔE	max ΔE
Standard ANN	0.70	1.88	1.86	23.06
Single-wavelength ANN	0.94	5.77	1.39	9.44

Conclusions

This work has compared the performance of two conventional models (Stearns-Noechel model and Friele model) for the prediction of the colour of fibre blends and also two neural network models (standard and single-wavelength neural networks). The performance of both the neural networks exceeded that of both conventional models. The average ΔE (between predicted and measured spectra) for the test set of data was 3.05, 2.42, 1.10 and 1.05 for the Stearns-Noechel, Friele, standard neural network and single-wavelength neural network respectively. Note that the Stearns-Noechel and Friele models were fully optimised (to give b = 0.10 for the Stearns-Noechel model and σ = 0.23 for the Friele model). The neural network methods, however, were not fully optimised in the sense that not all possible configurations of the networks and their training parameters were explored; nevertheless the neural networks gave much lower error scores (between predicted and measured spectra) than either of the two conventional models.

The performance of the two neural network models was similar but the mean ΔE on the test set of data was 1.10 for the standard neural network and 1.05 for the single-wavelength neural network. Note that the best performance for the standard neural network was for a

structure with 15 hidden units (693 weights) and the best performance for the single-wavelength neural network was with a population of networks each with 3 hidden units (each with 31 weights). The ratio of training samples to weights was therefore 273/693 and 273/31 for the two network models respectively.

The above results were all obtained using 273 training samples. When the training set was sub-sampled to produce a smaller training set with only 100 samples in it, the differential in performance between the standard neural network and the single-wavelength neural network increased. Now, the mean□E error on the test set was 1.86 for the standard neural network and 1.39 for the single-wavelength neural network. The maximum error score on the test set was almost three times larger for the standard neural network than for the single-wavelength neural network. Note that in this condition, the ratio of training samples to weights was 100/693 for the standard neural networks and 100/31 for the single-wavelength neural network.

These results support the notion that the ratio of training samples to weights in a neural network is a key factor in their performance. It is not entirely clear whether the ratio of training samples to weights needs to be greater than one, or greater than five, or some other number in order to produce acceptable results. Nevertheless, this paper has shown than the larger this ratio is, in general the better the generalisation performance of the neural networks. When the number of training samples available for a problem is very large, it may be that the single-wavelength network may offer no significant advantage over the standard neural network. However, as has been discussed earlier, in many cases the number of training samples is limited and may be insufficient to allow a high-enough ratio of training samples to weights; this may be especially true when the problem is spectral rather than colorimetric (because when data are being predicted at many wavelengths the networks become necessarily large). In such cases, the single-wavelength neural network approach is advantageous because each network has many fewer weights than in the standard network. As Anderson and Rosenfeld [28] put it: "A good representation does most of the work."

Whilst all of this work has been carried out using a set of eight primary colours it is likely that similar findings would result if a greater number (or indeed a smaller number) of primaries was used. However, further work is required to extend this analysis to larger scale problems. The work was also conducted with viscose fibre blends but since the neural network models are non-parametric there is no reason to doubt that similar performance for the neural network models would be achievable with other fibre types.

Acknowledgement
The authors would like to thank Lenzing AG and The Worshipful Company of Clothworkers for their generous financial support for this project.

References

1) E I Stearns and F Noechel, *Am. Dyest. Rep.*, **33** (1944) 177.
2) L F C Friele, *J Text Inst*, **43** (1952) 604.
3) B Philips-Invernizzi, D Dupont and C Cazé, *CRA*, **27** (2002) 191.
4) B Philips-Invernizzi, D Dupont, A-M Jolly-Desodt and C Cazé, *CRA*, **27** (2002) 100.
5) D A Burlone, *CRA*, **9** (1984) 213.
6) D A Burlone, *CRA*, **8** (1983) 114.
7) R Li, Y Song, F Gu and W Pan, *CRA*, **34** (2009) 108.
8) L Thevenet, D Dupont and M Jolly-Desodt, *CRA*, **28** (2003) 50.
9) R Furferi and L Governi, *CRA*, **36** (2011) 179.
10) R Shamey and T Hussain, *Rev. Prog. Col.*, **33** (2003) 33.
11) J M Bishop, M J Bushnell and S Westland, *CRA*, **16** (1991) 3.
12) S Westland, *Proceedings of Colour Science 98*, University of Leeds, **3** (2001) 225.
13) W S Sarle (2002), ftp://ftp.sas.com/pub/neural/FAQ3.html , (last accessed 20th Aug, 2014).
14) D H Alman and L Ningfang, *CRA*, **27** (2002) 122.
15) S Westland, L Iovine and J M Bishop, *Proceedings of the 9th Congress of the International Color Association*, (2001) 745.
16) C W Kan and L J Song, *Neural Process Lett.*, DOI 10.1007/s11063-015-9485-7 (2015).
17) N Hung, C K Chan, C W Kan, C W M Yuen and L J Song, *Fibres and Polymers*, **15** (2014)1330.
18) S Tominaga, *CRA*, **18** (1993) 253.
19) D Dupont, *CRA*, **27** (2002) 88.
20) V Cheung, S Westland, D Connah and C Ripamonti, *Coloration Technology*, **120** (2004) 19.
21) C De M Bezerra and C J Hawkyard, *Coloration Technology*, **116** (2000) 163.
22) M Jawahar, C B N Kannan and M K Manobhai, *Coloration Technology*, **131** (2015) 48.
23) B Sennaroglu, E Öner and Ö Şenvar, *Industria Textilă*, **65** (2014), 22.
24) Lenzing (2014), Lenzing Viscose: specifications. [Online] [Accessed 5th May 2014] Available from http://www.lenzing.com/en/fibers/lenzing-viscose/specifications.html
25) J C Guthrie and P H Oliver, *JSDC*, **73** (1957) 533.
26) M T Hagan, H B Demuth and M H Beale (1996), *Neural Network Design*, Boston, MA: PWS Publishing.
27) MathWorks (2014), Neural Network Toolbox. [Online]. [Accessed 31 July 2014]. Available from: http://uk.mathworks.com/products/neural-network/
28) J A Anderson and E Rosenfeld, *Neurocomputing, foundations of research*, (1988) p. 587, MIT Press, Cambridge MA.