Computer Assisted Mechanics and Engineering Sciences, **18**: 265–273, 2011.

Neural identification of compaction characteristics for granular soils

Marzena Kłos, Zenon Waszczyszyn
Rzeszów University of Technology, Chair of Structural Mechanics
W. Pola 2, 35-959 Rzeszów, Poland
e-mail: marklos@prz.edu.pl, zewasz@prz.edu.pl

Maria J. Sulewska
Białystok Technical University, Faculty of Civil and Environmental Engineering
Wiejska 45A, 15-351 Białystok, Poland
e-mail: sulewska@pb.edu.pl

The paper is a continuation of [9], where new experimental data were analysed. The Multi-Layered Perceptron and Semi-Bayesian Neural Networks were used. The Bayesian methods were applied in Semi-Bayesian NNs to the design and learning of the networks. Advantages of the application of the PrincipalComponent Analysis are also discussed. Two compaction characteristics, i.e. Optimum Water Content and Maximum Dry Density of granular soils were identified. Moreover, two different networks with two and single outputs, corresponding to the compaction characteristics, are analysed.

Keywords: granular soils, compaction characteristics, Optimum Water Content (OWC), Maximum Dry Density (MDD), neural networks, Multi-Layered Perceptron (MLP), Semi-Bayesian NN (SBNN), Principal Component Analysis (PCA).

1. Introduction

Engineering structures which involve earthwork such as roadway embankment, earth dams and soil liners often require compaction to improve soil conditions. The Optimum Water Content (OWC) and Maximum Dry Density (MDD) are essential characteristics for the design of compacted earthwork. These characteristics depend on classification properties of soils. In case of granular soils classification data can be reduced to parameters corresponding to soil grain-size distribution $\{D_x\}$, see [8]. They can be used as input variables. The output variables $\{MDD, OWC\}$ can be obtained experimentally by means of the laboratory Proctor's Standard Test ASTM D558-57, cf. [5, 6].

Proctor's Test is laborious and time consuming. That is why Artificial Neural Networks (ANNs) have been used to predict compaction characteristics, cf. [4, 7–9]. The application of Standard NNs was discussed in [4, 7] for synthetic soils consisting of four different components. In paper [7], besides *MDD* and *OWC*, permeability of mixture soils was also analysed. In book [8] it was proved that granular soils can be analysed by means of classification data $\{D_x\}$. This idea was developed in [9], where new measurement data were analysed.

In the papers mentioned above it was shown that the Multi-Layered Perceptrons (MLPs) are much better than the statistical approach basing on the multiple regression method. Moreover, it was shown in [8, 9] that correlation analysis is applied only a part of data, taken from grain-size distribution, can be taken into account as the input data. Moreover, it was numerically proved in [7] that instead of two output MPNs a good approximation could be given by two networks with single outputs, corresponding to characteristics MDD and OWC. This approach was also partially adopted in book [8] and paper [9].

In the present paper, besides the standard MLP a more refined, Semi-Bayesian Neural Network (SBNN) was applied. It is based on the extended error function and application of Bayesian methods to the computation of hyperparameters in the error function. The Bayesian method was also applied to the SBNN design instead of the cross-validation method commonly used.

Additionally, in case of SBNN the Principal Component Analysis (PCA) was applied for preprocessing of input data. In the paper, the adoptions of two MPNs with single outputs are discussed. The application of SBNN with the extended error function (with a penalty term) and application of PCA enabled us to obtain satisfactory results for efficient networks without losing the approximation accuracy.

2. Adopted data

A set of pattern pairs $P = \{\mathbf{x}^p, t^p\}_{p=1}^{P}$ was composed of $P = 121$ tests corresponding to the application of Proctor's Standard Test to the postglacial soils from the north of Poland, see [9]. In this paper MPLs were applied and the set of P measured patterns was randomly split into sets of learning, testing and validation sets composed of $L = 0.5P = 61$, $T = V = 0.25P = 30$ patterns, respectively. In case of the SBNNs also the PCA transformation was used for the input data preprocessing. Then only two sets of patterns were selected, i.e. the training and testing sets composed of $L = 0.7P = 85$ and $T = 0.3P = 36$ patterns. The PCA method was also applied for input compression.

According to [8] the input data are related to grain-size distribution corresponding to a sequence of nine grain diameters:

$$\{D_x\} = \{D_{10}, D_{20}, D_{30}, D_{40}, D_{50}, D_{60}, D_{70}, D_{80}, D_{90}\}, \tag{1}$$

where x [%] is the percentage of grain diameters D [mm] below which the soil mass is placed. Besides diameters listed in (1), the tenth input data corresponds to the uniformity coefficient

$$C_U = \frac{D_{60}}{D_{10}}. \tag{2}$$

Thus, the vector of input data has ten components

$$\mathbf{x}_{(10\times1)} = \{C_U,\, D_x \,|\, x = 10\%,\, \ldots,\, 90\%\}\ . \tag{3}$$

An extensive correlation analysis was carried out in [8, 9], which enabled us to reduce the number of important grain size diameters D_x. Due to the correlation analysis the following five input vectors were adopted:

$$\mathbf{x}_{(5\times1)} = \{C_U,\, D_{10},\, D_{20},\, D_{70},\, D_{80}\}. \tag{4}$$

The output variables correspond to two compaction characteristics, i.e.

$$\mathbf{y}_{(2\times1)} = \{OWC,\, MDD\}. \tag{5}$$

Obviously, the measurements by Proctor's Test were used as the components of the target vector $\mathbf{t}_{(2\times1)}$.

When the Principal Component Transformation (PCT) was applied, the covariance matrix of size (10×10) was computed, cf. [10]. Then PCA (PCT analysis) method was also applied for input data compression. It is shown in Point 3.2.4 that the input space can be decreased to a 4D space.

3. Neural network analysis

3.1. Application of MLP

Standard neural network analysis is related to the networks in which the Least Square method is applied with the E_{LS} error function (for the sake of clarity only a single output is considered):

$$E_{\mathrm{LS}\,i} = \frac{1}{2}\sum_{p=1}^{S}\{t_i^p - y_i(\mathbf{x}^p;\mathbf{w})\}^2, \tag{6}$$

where p – numbers of patterns, t^p, $y_i(\mathbf{x}^p;\mathbf{w})$ – target and computed outputs for pattern p and subscript i corresponds to compaction characteristics OWC, MDD, $\mathbf{x}^p \in \mathcal{R}^D$ – input vector, $\mathbf{w} \in \mathcal{R}^W$ – weight vector, $S = L$, V, T – numbers of patterns in the learning, validation and testing sets, respectively.

From among many standard neural networks we adopted the feed-forward layerd, error back propagation network which in [10] was called Multi Laeyred Perceptron (MLP). The tgh sigmoid activation functions were applied in H hidden neurons and linear outputs were assumed.

The MATLAB neural toolbox [1] was used. MLPs were trained by means of the Levenberg-Marquardt learning method. In Table 1, values of the Root Mean Square Errors $RMSE_i^S$ and determination $(R_i^S)^2$ are listed:

$$RMSE_i = \sqrt{\frac{1}{s}\sum_{p=1}^{S}(t_i^p - y_i^p)^2}, \qquad (R_i^S)^2 = 1 - \frac{SSE_i^S}{SS_{yi}^S}, \tag{7}$$

where

$$SSE_i^S = \sum_{p=1}^{S}(t_i^p - y_i^p)^2, \qquad SS_{yi}^S = \sum_{p=1}^{S}(y_i^p - \overline{y}_i)^2, \qquad \overline{y}_i = \frac{1}{S}\sum_{p=1}^{S} y_i^p.$$

Table 1. Network learning and testing errors.

ANN No	Architecture	*NNP*	Outputs	$RMSE_i^L$	$RMSE_i^T$	$(R_i^L)^2$	$(R_i^T)^2$
1*	10-4-1	49	*OWC*	0.121	0.159	0.75	0.65
2*	5-4-1	29	*MDD*	0.077	0.085	0.91	0.89
3	10-3-2	41	*OWC*	0.118	0.142	0.85	0.62
			MDD	0.063	0.074	0.95	0.85
4	5-5-2	42	*OWC*	0.112	0.141	0.82	0.65
			MDD	0.061	0.074	0.93	0.85
5	10-4-2 PCA	54	*OWC*	0.098	0.112	0.89	0.79
			MDD	0.055	0.065	0.96	0.90
6	4-5-2 PCA	37	*OWC*	0.110	0.139	0.78	0.60
			MDD	0.070	0.081	0.87	0.80
7	10-2-1 PCA	25	*OWC*	0.095	0.123	0.88	0.77
8	10-4-1 PCA	49	*MDD*	0.025	0.055	0.98	0.93
9	4-2-1 PCA	13	*OWC*	0.096	0.159	0.86	0.55
10	4-2-1 PCA	13	*MDD*	0.030	0.063	0.90	0.85

* $L = 0.5P$, $T = 0.25P$

The optimum number of hidden neurons H_{opt} for the families of neural networks MLP: N-H-M with N inputs and $M = 1$ or 2 outputs were analysed in [9] applying the cross validation method. The number of learning, validation and testing patterns was $L = 61$, $V = T = 30$, respectively.

The computations carried out in [9] confirmed the conclusions from [7, 8] that the ANN of different size could be recommended for prediction of compaction characteristics. That is why from many results discussed in [9], we quote in Table 1 the errors obtained for the best networks with single outputs, i.e. the networks No 1^* and No 2^*. For these networks the optimum number of hidden neurons $H_{opt} = 4$ was adopted for both the compaction characteristics OWC and MDD.

Looking at Table 1 it can be concluded that the errors $RMSE_i^S$ and determination $(R_i^S)^2$ are higher for $i = OWC$ than for $i = MDD$, both for the learning $S = L$ and training $S = T$ numbers of patterns. These results are related to different physical phenomena related to OWC and MDD, cf. [5, 8].

3.2. Application of SBNN and PCA

3.2.1. Extended error function

In the presented paper the extended error measure E_{PLSi} is investigated, corresponding to adding the penalty term E_{Wi} . These terms are weighted by hyperparameters α_i and β_i:

$$E_{\mathrm{PLS}i}(\mathbf{w}) \equiv F_i(\mathbf{w}) = E_{\mathrm{LS}i}(\mathbf{w}) + E_{\mathrm{W}i}(\mathbf{w}) = \frac{\beta_i}{2}\sum_{p=1}^{P}\left\{t_i^p - y_i(\mathbf{x}^p;\mathbf{w})\right\}^2 + \frac{\alpha_i}{2}\sum_{j}^{W} w_j^2, \tag{8}$$

for $i = OWC,\ MDD$, where the same notation as that in formula (6) was used.

In (8) hyperparameters α_i and β_i are introduced as assumed in the Bayesian NNs.

3.2.2. Learning of SBNN

The formulas presented in the Appendix are the basis on which the Bayesian procedures were written in book [3] by Nabney. The weight vector $\mathbf{w}_{MAP}$ (the subscript MAP corresponds to the Bayesian approach Maximum A Posteriori) was computed by the conjugate gradient method joined with computing of hyperparameters α_i and β_i by means of iterative formulas, Appendix formulas (A7, A9). It was stated that the of hyperparameter can be well estimated within 3-7 iterations starting from the initial values of $\alpha_{in} \approx 0.01$ and $\beta_{in} \approx 50$.

3.2.3. Design of SBNN

In the paper only design of two networks with single outputs and architecture SBNN: D-H-1 are shown in Fig. 1. The number of hidden neurons was computed by means of procedure EVIDENCE from [3] adopting only the learning set composed of $L = 86$ patterns, see Appendix formula (A6). The Bayesian method MML and the commonly used cross-validation method were applied. In order to compare the estimated number of hidden neurons H_{opt} the testing set with $T = 35$ patterns was used as the validation set. The example graphics of functions $\ln ML(H;\ L)$ and $RMSE^T(H;\ T)$ are shown in Fig. 1 for the network with ten and four compressed inputs, respectively. As can be seen, for the family of networks SBNN: D-H-1 the optimal value of the hidden neuron numbers is $H_{opt} = 4$ for $D = 10$ and $H_{opt} = 2$ for $D = 4$.

The same number of optimal hidden neurons is given by the cross-validation method. What is worth emphasising again is that the curve marked as $\ln ML$ was obtained for the learning set of input data set. In case of the cross-validation method, the curve RMSET corresponds to a randomly selected validation (testing) set of patterns.

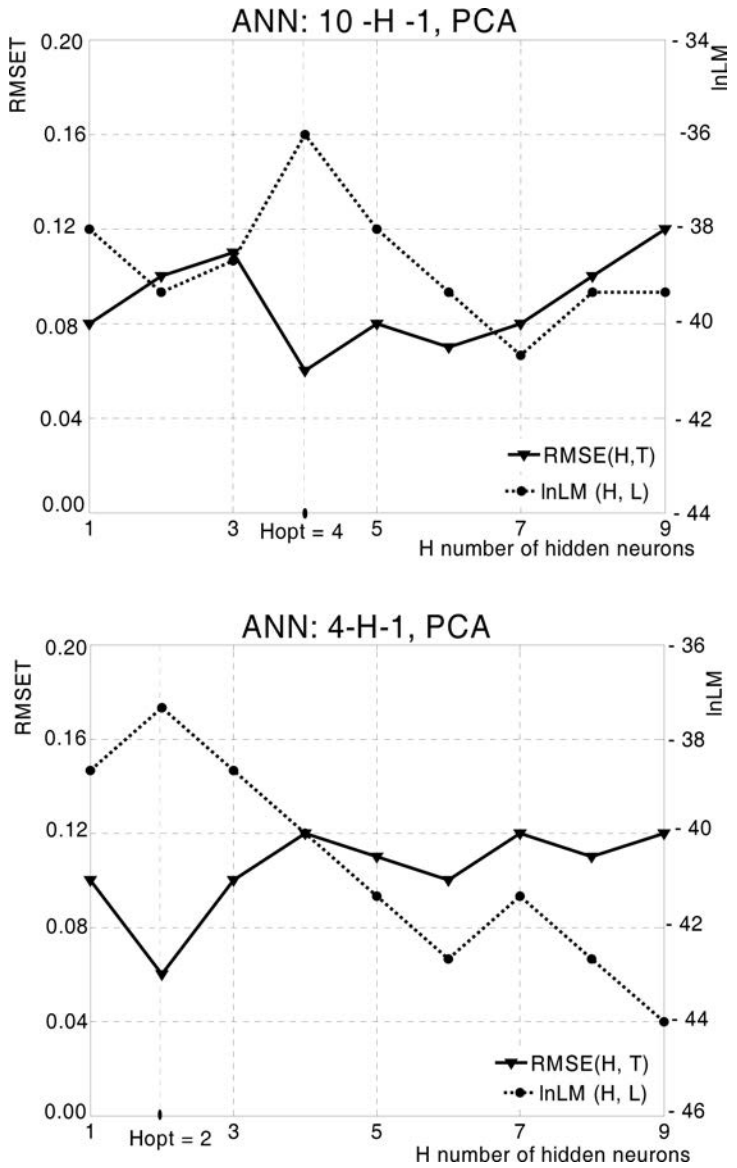

Fig. 1. Curves log $ML(H;\, L)$ and $RMSE^T(H;\, T)$ for design of neural networks by means of MNN and cross-validation method for network 10-H-1, PCA.

3.2.4. Application of PCA method

In the present paper the PCA (Principal Component Analysis) was applied. Following the PCA algorithm, cf. [2, 10], the covariance matrix $\mathbf{C}_{(10\times10)}$, computed for the input data set of $L = 85$ learning patterns with components (3), was formulated. Proncipal eigenvalues of this matrix have the following values:

$$\{\lambda_j\}_{(10\times1)} = \{20.7105, 4.6436, 0.4818, 0.0744, 0.0070, 0.0035, 0.0007, 0.0006, 0.0005, 0.0000\}. \quad (9)$$

The PCA bases on linear transformation of input patterns to the principal coordinates $\xi_j \in \mathcal{R}^{10}$, corresponding to the eigenvectors $\mathbf{q}_j$. This transformation can be treated as preprocessing of input data.

Looking at the eigenvalues (9) it is evident that the first four eigenvalues dominate so the inputs can be compressed to four principal components which can be adopted for the network family of architecture SBNN: 4-H-1.

3.3. Comparison of compaction parameters identification by SBNNs

Networks with two outputs. The SBNN networks with and without PCA were applied to identification of compaction characteristics. In networks Nos 3 and 4, shown in Table 1, the input data were assumed according to formulas (3) and (4), adopting two outputs, i.e. networks of architecture SBNN: D-H-2, where $D = 10$, 5 and $H = 3$, 5. The application of Bayesian methods for the design and learning gave the networks which helped overcome the difficulties mentioned in [8, 9] concerning the learning networks with two outputs. It is visible that identification by means of reducing the number of inputs from ten to five gave results comparable to those without data compressing. This was caused by the application of networks of nearly the same number of network parameters (i.e. number of synaptic weights and biases) $NNP = 42$ and 41 for the networks Nos 3 and 4, respectively.

Networks with single output. The standard networks of architectures MLP: D-H-1 Nos 1^* and 2^* were applied in [9]. They were learned and tested by a different number of patterns $L = 0.5 \times 121 = 61$ and $T = 0.25 \times 121 = 30$, respectively, than other networks listed in Table 1. In case of SBNNs the corresponding sets were composed of $L = 86$ and $T = 35$ patterns. The networks were designed and learned by Baysian methods. The corresponding errors are shown in Table 1 for the networks Nos 7–10. Both in ten principal components and four compressed components the PCA method was applied.

As indicated by Table 1, for the networks mentioned above the errors *RMSE* are nearly the same. In order to compare the errors obtained for four compressed inputs (networks SBNN: 4-2-1, PCA) the computations were also carried out for the transformed data, but without the input compression (networks SBNN: 10-H-1, PCA). An advantage of the input data compression lies in decreasing the networks Nos 9 and 10 sizes to $NNP = 13$, vs. $NNP = 25$ and 49 for the networks Nos 7 and 8 in which the input data were not compressed.

4. Prediction properties of networks with applied PCA

The distributions of points $(y^p,\ t^p)_i$ are shown in Fig. 2, separately for $i = OWC$ and $i = MDD$. The points related to the target values, taken from Proctor's Standard Test, are distributed around the diagonals $y_i^p = t_i^p$.

For the prediction purposes the relative error Re_i [%] is introduced:

$$Re_i = (y_i^p / t_i^p - 1) \times 100\%, \quad (y_i^p = t_i^p), \tag{10}$$

where coordinates y_i^p, t_i^p determine pattern points p on the planes $i = OWC,\ MCC$. In Fig. 2 the lines $Re = \pm C\%$ bound the area in which prediction points with the relative errors $|Re| \leq C$ are placed.

The bounded area corresponds to the cumulative parameter SR [%], called Success Ratio, defined in [2] as:

$$SR = \frac{SRe}{S} \times 100\% \qquad \text{for} \quad S = L, T, \tag{11}$$

where: SRe – number of prediction points in the Re area, S – total number of points of the data sets $S = L$, T. In Fig. 3 the cumulative curves SR_i^S (Re_i^S) are shown for the learning and testing sets computed by the networks 10-4-1, PCA and 4-2-1, PCA. It is evident that the PCA transformation improves the accuracy of neural approximation. The network SBNN: 10-4-1, PCA gives the best results from all the networks presented in Table 1. This also concerns the testing determination $(R_i^T)^2$.

The network SBNN: 4-2-1 PCA with compressed inputs gives slightly higher values of errors, than the network of architecture 10-4-1, see Table 1. The Success Ratio curves $SC(Re)$ shown in Fig. 3 are close to each other for the networks discussed above both for the training and testing

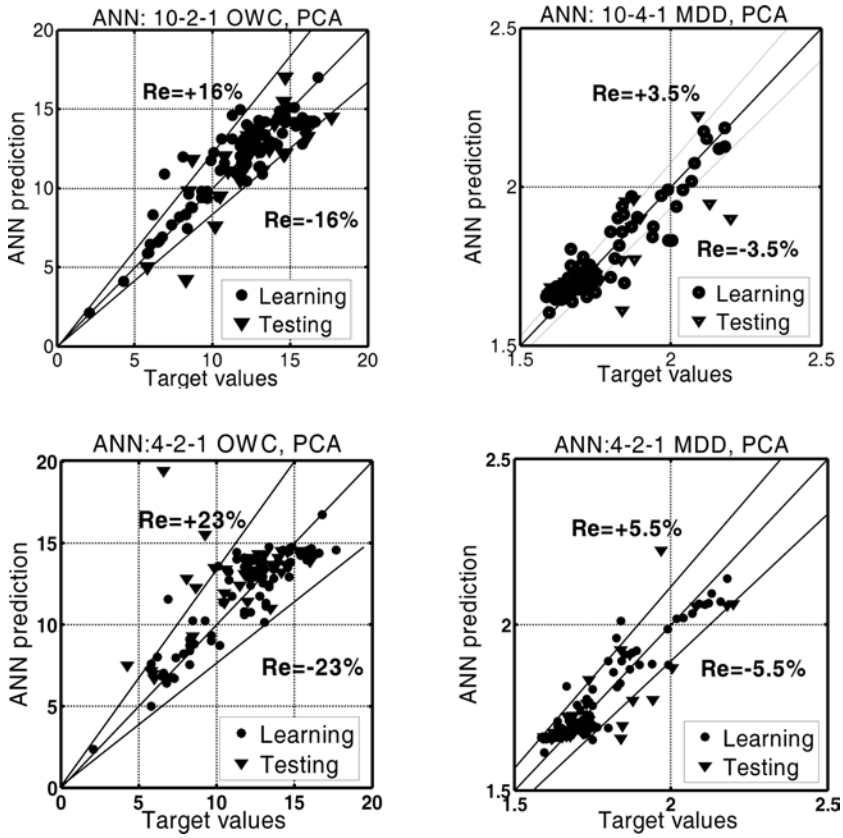

Fig. 2. Areas of $SR = 80\%$ with error bounds Re shown for networks ANN: 10-4-1 and 4-2-1 , PCA and compaction parameters OWC and MDD.

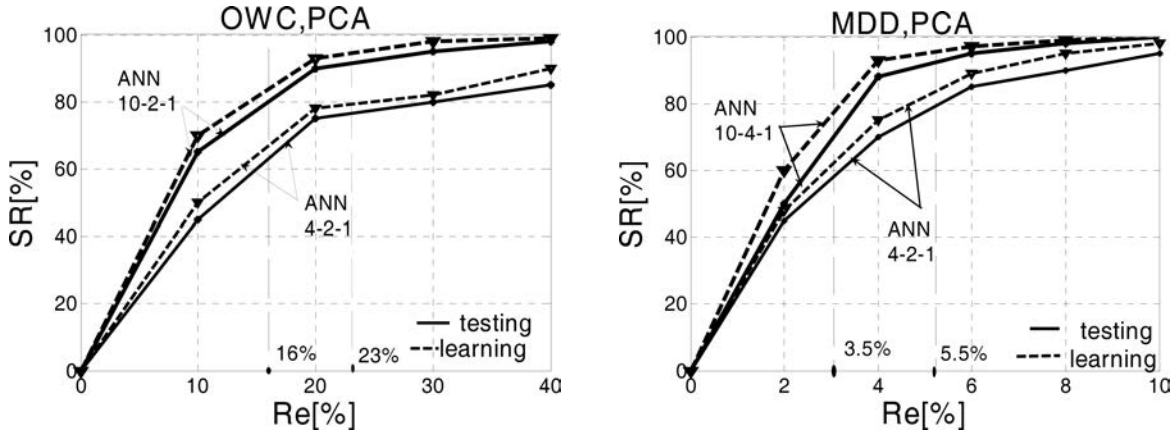

Fig. 3. Cumulative curves of Success Ratios SR (Re) for the Re percent of correctly predicted compaction characteristics OWC and MDD and networks ANN: 10-4-1. 4-2-1, PCA.

patterns. If we assume that about 80% of testing patterns are correctly predicted with $SR \approx 80\%$, then the error area bounds are $|Re| \approx$16–23% and $|Re| \approx$3.5–5.3% for the compaction characteristics OWC and MDD, respectively.

5. Final remarks

1. The application of the penalised network error function (8) enables formulation of efficient networks with two outputs for the compaction characteristics OWC (Optimum Water Content) and MDD (Maximum Dry Density) prediction.

2. The compressed input vector (4), corresponding to four selected components of Proctor's Test, enables formulation of a small network 4-2-1 of accuracy comparable with the network 10-4-1 with the non-compressed number of inputs.

3. The application of PCA (Principal Component Analysis) and MML (Maximum Marginal Likelihood) criterion makes it possible to formulate the PCA networks 10-4-1 and 4-2-1 with a high accuracy of approximation. The PCA approach, mathematically well grounded, seems to be simpler than selection of inputs on the basis of the correlation analysis.

Appendix

MML, design of SBNN, computation of α and β

The criterion of Maximum Marginal Likelihood (MML) is supported on the Bayes' theorem, which is written with respect to conditional probabilities $p(A|B)$, where A and B are specified to variables used in the present paper:

$$p(\mathbf{w}|\mathbf{t},\alpha,\beta) = \frac{p\,(\mathbf{t}|\mathbf{w},\beta)\,p(\mathbf{w}|\alpha)}{p(\mathbf{t}|\alpha,\beta)}, \tag{A1}$$

where $p\,(\mathbf{t}|\alpha,\beta)$ is the Marginal Likelihood ML (also called Evidence).

$$p(\mathbf{t}|\alpha,\beta) = \int\limits_{\mathcal{R}^W} p(\mathbf{t}|\mathbf{w},\beta)p(\mathbf{w}|\alpha)d\mathbf{w}. \tag{A2}$$

In the theorem (A1) the neural weight vector $\mathbf{w} = \{w_i\}_{i=1}^{W}$ is used, where W is the number of weights. This number is linearly related to the number of hidden neurons H in feed-forward, single layer network of structure D-H-M with D inputs and M outputs. The number of weights is $W = (D + M + 1) \times H + M$ so we have $H = (W - M)/(D + M + 1)$.

Applying Gaussian approximation of Marginal Likelihood function $ML = p(\mathbf{t}|\alpha,\beta)$ the following equation for ln ML can be derived, cf. [10, 11]

$$\boxed{\ln ML \equiv \ln p\,(\mathbf{t}|\alpha,\beta) \cong -\widehat{E}(\mathbf{w}_{\rm MAP}) - \frac{1}{2}\ln|\mathbf{A}| + \frac{W}{2}\ln\alpha + \frac{N}{2}(\ln\beta - \ln 2\pi)}\,, \tag{A3}$$

where

$$\begin{aligned} &\widehat{E}(\mathbf{w}) \cong \widehat{E}(\mathbf{w}_{\rm MAP}) + \frac{1}{2}(\mathbf{w} - \mathbf{w}_{\rm MAP})^T\mathbf{A}(\mathbf{w} - \mathbf{w}_{\rm MAP}),\\ &\widehat{E}(\mathbf{w}_{\rm MAP}) = \beta E_D(\mathbf{w}_{\rm MAP}) + \alpha E_W(\mathbf{w}_{\rm MAP}) = \frac{\beta}{2}\sum_{n=1}^{N}\{t^n - y(\mathbf{x}^n;\mathbf{w})\}^2 + \frac{\alpha}{2}\mathbf{w}_{\rm MAP}^T\mathbf{w}_{\rm MAP}, \end{aligned} \tag{A4}$$

$$\mathbf{A} = \alpha\mathbf{I} + \beta\mathbf{H}, \qquad \mathbf{H} = \nabla_{\mathbf{w}}\nabla_{\mathbf{w}}E_D. \tag{A5}$$

A data set of patterns, applied in (A4), corresponds to the pairs $\{\mathbf{x}^n,\ t^n\}_{n=1}^{N}$. The subscript $_{\rm MAP}$ means Maximum 'A Posterior', related to the left-hand side of the Bayes' theorem (1). Matrix $\mathbf{A}$ is expressed by the Hessian matrix $\mathbf{H}$.

Thus, criterion MML (Maximum Marginal Likelihood) corresponds to:

$$\boxed{\max_{w} \ln ML \to W_{opt} \sim H_{opt}}\,. \tag{A6}$$

The hyperparameters α and β are used in the extended neural network error (A4). These hyperparameters can be computed iteratively using the following formulas derived on the basis of minimisation of the function $\ln ML$ with respect to either α and β, see [11]:

$$\boxed{\alpha_{in} = \frac{\gamma}{\mathbf{w}_{\mathrm{MAP}}^T \, \mathbf{w}_{\mathrm{MAP}}}, \qquad \frac{1}{\beta_{in}} = \frac{1}{N-\gamma} \sum_{n=1}^{N} \left\{ t^n - y\left(\mathbf{x}^n;\, \mathbf{w}_{\mathrm{MAP}}\right) \right\}^2} , \tag{A7}$$

where parameter γ is related to the eigenvalues λ_i of the Hessian matrix $\mathbf{H}$:

$$\gamma = \sum_{i=1}^{W} \frac{\lambda_i}{\alpha + \lambda_i}. \tag{A8}$$

Updated values of the hyperparameters are estimated by the following formulas:

$$\boxed{\alpha_{\mathrm{new}} = \frac{\gamma}{2\, E_W(\mathbf{w}_{\mathrm{MAP}}^{\mathrm{old}})}, \qquad \beta_{\mathrm{new}} = \frac{N-\gamma}{2\, E_D(\mathbf{w}_{\mathrm{MAP}}^{\mathrm{old}})}} , \tag{A9}$$

where the weight vector $\mathbf{w}_{\mathrm{MAP}}^{\mathrm{old}}$ is computed for the networks investigated.

The formulas presented above are the basis of an algorithm called in [3], pp. 341–349, the EVIDENCE procedure.

Acknowledgments

The authors would like to acknowledge the financial support from the Polish Ministry of Science and Higher Education, Grant No. N N506 4326 36, "Joining of artificial neural networks and Bayesian reasoning for the analysis of identification problems in structural dynamics and geotechnics".

References

[1] H. Demuth, M. Beale. *Neural Network Toolbox for Use with MATLAB*, User's Guide, Version 3, The Mathworks Inc., 1998.
[2] S.S. Haykin. *Neural Networks – A Comprehensive Introduction*, 2^{nd} Ed. Prentice Hall, 1999.
[3] I.T. Nabney. *Netlab – Algorithms for Pattern Recognition*. Springer, London, 2004.
[4] Y.M. Najjar. On the identification of compaction characteristics by neuronets. *Computers and Geotechnics*, **18**: 167–187, 1996.
[5] R.R. Proctor. Fundamental principles of soil compaction. *Eng. News Records*, **111**: 245–248, 1993.
[6] R.R. Proctor. Description of field and laboratory methods. *Eng. News Records*, **111**: 286–289, 1993.
[7] S.K. Sinha, M.C. Wang. Artificial neural network prediction models for soil compaction and permeability. *Geotech. Geol. Eng.*, **26**: 47–64, 2008.
[8] M. Sulewska. *Artificial Neural Networks for Evaluation of Compaction Parameters of Cohesionless Soils* (in Polish). Committee Civil Eng. of the Polish Acad. Sci., Warsaw – Białystok, 2009.
[9] M. Sulewska. Artificial Neural Modelling of Compaction Characteristics of Cohesionless Soil. *Comp. Aided Mech. Eng. Sci.*, **17**: 27–40, 2010.
[10] Z. Waszczyszyn, M. Słoński. Maximum of marginal likelihood criterion instead of cross-validation for designing of artificial neural networks, In: L. Rutkowski et al., [Eds.], 9th *Artificial Intelligence and Soft Computing ICAISC 2008*, LNAI 5097, pp. 186–194. Springer, Berlin – Heidelberg – New York, 2008.
[11] Z. Waszczyszyn, M. Słoński. Selected problems of artificial neural network development, Ch. 5 in: Z. Waszczyszyn [Ed.], *Advances of Soft Computing in Engineering*, CISM Courses and Lectures, vol. 512, pp. 237–316. Springer, Wien – New York, 2010.