15th Triennial World Congress, Barcelona, Spain

PREDICTION OF DYNAMIC MEDICAL DATA SERIES USING NEURAL NETWORKS METHOD

Wieslaw Wajs *

* *Institute of Automatics,*
University of Mining and Metallurgy,
al. Mickiewicza 30, 30-059 Krakow, Poland
Fax: +4812 6341568, e-mail: wwa@ia.agh.edu.pl

Abstract: The aim of the paper is to develop neural networks technique for solving some important medical problem. That problem is called "Blood gases prognosis values". The main question, that is considered, is how to predict some parameters that describe blood gases nature in the future for a nowborn based on a set of parameters that describe this child now, and in the past. We are expected to receive some parameter value on the proper level of probability. *Copyright ©2002 IFAC*

Keywords: Neural networks, Approximate analysis, Decision making, Dynamic modelling, Decision support systems

1. INTRODUCTION

The purpose of this paper is to derive an estimate of a neural network's accuracy as an empirical modelling tool. A model of a physical system has an error associated with its predictions due to the dependence of the physical system's output on uncontrollable or unobservable quantities. Neural Network models have been used as a predictor for different physical systems.

The calculation of second derivatives is required by recent training and analysis techniques of connectionist networks. That techniques can be used for the estimation of confidence intervals both for weights and network outputs. The networks of sigmoid units, exact calculation of the necessary intermediate terms requires of the order of $2h+2$ backward - forwardpropagation passes where h is the number of hidden units in the network. Those second derivatives are typically derivatives either of the network output or of the error function with respect to the weights. Second derivatives can be calculated exactly, calculated using approximation ignoring certain terms, or calculated using numerical differentiation that interprets the definition of the derivative numerically by

$$\frac{\partial E(x)}{\partial x} \approx \frac{(E(x+\Delta x)-E(x))}{\Delta x} \qquad (1)$$

Second derivatives are important in several different contexts. Simple beckpropagation is a first order gradient descent method. Learning speed can be improved if information from second derivatives is also used, for instance in a Newton-Raphson type of framework. Since the second derivatives have to be computed for each weight update, the speed of the computation is crucial here. For the huge networks, calculation of the full Hessian is considered prohibitive. There is another class of second order optimisation algorithms that do not require direct calculation of the Hessian because they operate in an iterative manner. The conjugate gradient and related algorithms are considered the most powerful all - purpose minimisation algorithms. The calculation can also be used iteratively in the power method to efficiently approximate the principle eigenvectors of the Hessian. Those method can be described briefly as: from a random initial assignment v_0, compute

$$u_{i+1} = Hv_i \tag{2}$$

then minimise u_{i+1} to a unit vector v_{i+1}, and iterating causes the eigenvectors with smaller eigenvalues to disappear. The speed of training in least mean square algorithms is related to the ratio of the largest to the smallest eigenvalues. This ratio is called the condition number and is also associated with the accuracy to which the minimum can be calculated. The condition number can be approximated by approximating the largest and smallest eigenvalues with the power method. Second derivatives are also used in a post - training phase. For example use the Hessian of the error to simplify the network by pruning weights in order to archive good generalisation performance. In Bayesian second derivatives are related to quantities such as the posterior variance of the network weights, and also to the description length of the set of weights used in evaluating the quality of the set of weights.

2. PROBLEM

Respiratory problems are the most prominent in pathology of new-borns hospitalised in neonatal intensive care unit (NICU). Arterial blood gases (ABG) are good indicators of severity of these problems. Accurate forecasting of ABG alternations caused by different factors would be of the great value in newborn intensive care. It is created a neural network structure that establishes prognosis of pH, PCO_2, PO_2, HCO_3 based on the National Information System (NIS) - a computer database used in NICU Polish - American Children's Hospital in Krakow. Currently over eight hundred patients are enrolled in the database. Most of the patient admitted in NICU suffers from respiratory distress. Respiratory insufficiency is the leading cause of hospitalisation and mortality as well. Blood gases analysed in context of actual respiratory setting to make know severity of respiratory distress. The values are forecasted with an error exceeding acceptable value. It is four neural network structures created to forecast pH, PCO_2, PO_2, and HCO_3 respectively. Input data vector is composed of eight values: surfactant (administration of the surface active agent or not), respiratory rate (RR), peak respiratory pressure (PIP), friction of inspired oxygen (FiO_2), and blood gases values: pH, PCO_2, PO_2, HCO_3. Respiratory setting and surfactant administration is shown in the form of the step function. Blood gases values are approximated by polynomials. Input data is taken with the step 0.1 [h]. Output data vector contained single blood gases parameter with a step next to input data. 15 patients' data was used for process of neural networks learning, and 9 other patients' data was used to test accuracy of prognoses. Forecast values were in the limit of acceptable error. Blood gases values depend on many factors. Presented methods is able to show a flow of varying parameters without detailed analysis of determining factors. Some factors can be constant. There is birth weight, age, Apgar score and sex. Some factors are changes step by step. There is respiratory settings, administration of medication and presence of infection. Some factors are continuous. There is: heart rate, haemoglobin saturation and blood pressure. It is considered lung mechanical parameters: tidal volume, lung compliance and airway resistance. The general conception of neural networks forecasting values of parameters important in treatment is shown as follows.

Table 1 Parameter pH, PCO_2, PO_2, HCO_3 mean value and close mean value

parameter	*mean value*	*close mean value*
pH	7.35-7.45	7.125-7.35; 7.45-755
PCO_2	40-55	30-40;55-70
PO_2	40-50	30-40;50-70
HCO_3	21-25	18-21;25-28

Neural networks receives of input parameter values in the real time from external devices: computer database (NIS), local computer database, and patient's monitor. Input file is characterised by: *mean* and *std*.

Table 2 Parameter pH, PCO_2, PO_2, HCO_3 mean and std value

input parameter	*mean*	*std*
pH	7.3	0.08
PCO_2	45.7	11
PO_2	55.6	21
HCO_3	22	3.7

Those parameters are used in neural networks learning process. The neural networks generates a prognosis of arterial blood gases values on users demand. The main goal of the study is to show that neural networks is able to learn forecasting blood gases values. Four neural networks are created. Each of the networks consists of three layers. The first layer formed eight non-linear neurones. The second layer is also formed of non-linear neurones. Their number of neurones in this layer is arbitrary chosen. Output layer is formed of case linear neurone. The archived result is promising. The archived results were based on assumed errors of prognosis: $pH(0.1)$, PCO_2(10mmHg), PO_2(10mmHg), HCO_3(3mmoll). Four tested patients estimated error did not reach maximum value of the assumed limit. Results were archived after input discrete parameters from the hospital database. The paper presents the application of neural predictor for Short - Term Forecasting. We investigate problem of estimating of the confidence intervals for the prediction. We have applied the neural networks model to few-hour blood gases prediction.

3. PROGNISIS OF BLOOD GASES VALUES

Blood gases values and treatment parameters flows recorded during a few days of hospitalisation. Neural networks creation learning process consists of two sets of the data. A first one contains the data introduced to the first layers of neurones. The second set contains the expected data. Analysis of the input data characteristics revealed that the data should be normalised before entering the first neurone layer. Each sequence of values of the separate input parameter recorded during several days was normalised in the simple way. All input values from the sequence are divided by the maximum value. High degree of the data discretisation and time step could also have negative impact on neural networks learning process. To reduce this phenomenon all input parameters are subject to polynomial approximation. The example of prognosis blood gases values, the values in learning vectors is with step 0.1 [h]. The example of approximated and normalised blood gases value flows without disturbing information.

Unsatisfactory result is caused by disturbances in function representing values of the parameters forming learning data sets and an inappropriate data step. An original data value from the database is recorded once or few times a day. Using polynomial method intermediate values is added. It is allowed to reduce data reading step to few hours.

The second group of input data is formed by values of factors, that has an impact on blood gases values, and they are referred as medical treatment parameters. The values of first parameter (surfactant administration) formed binary sequence, i.e. 1 means surfactant administration, 0 coded opposite situation. The values of the other parameters represented setting of respirator, and there is no need for their approximation. Eight sequences representing input parameters are created. To improve neural networks learning process each of the sequence is normalised.

Output file is characterised by *mean* and *std*.

Table 3 Output parameter pH, PCO_2, PO_2, HCO_3 mean and std value

output parameter	*mean*	*std*
pH	7.3	0.07
PCO_2	44.9	8.2
PO_2	56.1	16.8
HCO_3	22	3

Initially the period of time was equal to 1.5 hour.

4. CONFIDENCE INTERVAL PREDICTION

The aim of this section is to derive an estimate of a neural network's accuracy. Neural networks models have been used as a predictor. It is also be used as a method to quantify the confidence intervals of the predictions from neural networks models. For a desired degree of confidence (for a given probability), a confidence interval is a prediction of the range of a output model where the actual value exists. With the assumption of a normal distribution of the errors, confidence intervals can be calculated for neural networks.

The analysis is extended to include the calculation of confidence intervals for models obtained from noisy data. For the given system with output $y = y_1, y_2, ..., y_n$ the model for the system is given as $f(x, w^*)$, where: $x = x_1, x_2, ..., x_n$ is the set of inputs, and w^* represents the real values of the set of parameters w, for the function that models the system. The error ε_i associated with the function in modelling the system is assumed to be independently and identically distributed with the variance σ^2, where the distribution has the form $N(0, \sigma^2)$. With n observations, where $i = 1, 2, ..., n$ the system is represented by

$$y_i = f(x_i, w^*) + \varepsilon_i, \quad i = 1, 2, ..., n. \tag{3}$$

The least squares estimate of w^* is $\hat{w}$, that is obtained by minimising the error function that is used for neural networks Backpropagation (BP) algorithm. BP is a common method for minimising the error function. The predicted outputs from the model is $\hat{y}$

$$\hat{y}_i = f(x_i, \hat{w}). \tag{4}$$

The error function is given in the form:

$$\sum_{i=1}^{i=n} (f(x_i, w^*) + \varepsilon_i - f(x_i, w))^2 \tag{5}$$

It is assumed that the model gives a good prediction of the actual system behavior. It means that $\hat{w}$ is close the real value of the set of parameters

w^*, and a Taylor expression to the first order can be used to approximate $f(x_i, \hat{w})$ in terms of $f(x_i, w^*)$ where:

$$f(x_i, \hat{w}) \approx f(x_i, w^*) + \mathbf{f}_o\ (\hat{w} - w^*) \qquad (6)$$

where

$$\mathbf{f}^{\mathrm{T}}{}_o = \left(\frac{\partial f(x_i, w^*)}{\partial w_1^*}, \frac{\partial f(x_i, w^*)}{\partial w_2^*}, ..., \frac{\partial f(x_i, w^*)}{\partial w_p^*}\right). (7)$$

The difference between the real value y of the system and the predicted value $\hat{y}$ gives the expected value of the error.

The subscript value of o is given to denote the set of points other than that used for the least squares estimation of w^*. The difference

$$y_o - \hat{y}_o \approx y_o - f(x, w^*) - \mathbf{f}_o^T\ (\hat{w} - w^*) = \varepsilon_o - \mathbf{f}_o^T\ (\hat{w} - w^*). \qquad (8)$$

For an error ε_o with a normal distribution with a mean of zero and a variance of $\sigma^2(N(0, \sigma^2 \mathbf{I}_{\mathrm{n}}))$, the distribution of difference $\hat{w} - w^*$ can be approximated to have the distribution

$$N_p(0.\sigma^2\ [\mathbf{F}\ (\hat{w})^T\ \mathbf{F}\ (\hat{w})]^{-1})$$

where: Jacobian matrix $\mathbf{F}\ (\hat{w})$ has the form

$$\mathbf{F}\ (\hat{w}) = \frac{\partial f(\mathbf{x}, \hat{w})}{\partial \hat{w}} = \begin{bmatrix} \frac{\partial f(x_1, \hat{w})}{\partial \hat{w}_1} & \frac{\partial f(x_1, \hat{w})}{\partial \hat{w}_2} & ... & \frac{\partial f(x_1, \hat{w})}{\partial \hat{w}_p} \\ \frac{\partial f(x_2, \hat{w})}{\partial \hat{w}_1} & \frac{\partial f(x_2, \hat{w})}{\partial \hat{w}_2} & ... & \frac{\partial f(x_2, \hat{w})}{\partial \hat{w}_p} \\ ... & ... & ... & ... \\ \frac{\partial f(x_n, \hat{w})}{\partial \hat{w}_1} & \frac{\partial f(x_n, \hat{w})}{\partial \hat{w}_2} & ... & \frac{\partial f(x_n, \hat{w})}{\partial \hat{w}_p} \end{bmatrix} \qquad (9)$$

$$\mathrm{var}[y_o - \hat{y}_o] \approx \sigma^2 + \sigma^2\ \mathbf{f}_o^T\ [\mathbf{F}\ (\hat{w})^T\ \mathbf{F}\ (\hat{w})]^{-1})\mathbf{f}_o \qquad (10)$$

The matrix $\mathbf{F}\ (\hat{w})$ has the dimensions n by p where: n is the number of samples that is used to obtain $\hat{w}$, and p is the number of parameters w_i that composes $\hat{w}$. The unbiased estimator of σ^2 is s^2 as

$$s^2 = \frac{||\ y - f(x, \hat{w})\ ||^2}{n - p} \qquad (11)$$

The Student t- distribution is given in the form

$$t_{n-p} \sim \frac{y_o - \hat{y}_o}{\sqrt{\mathrm{var}[y_o - \hat{y}_o]}} \approx \frac{y_o - \hat{y}_o}{\sqrt{s^2 + s^2 \mathbf{f}_o^T (\mathbf{F}^T \mathbf{F})^{-1} \mathbf{f}_o}} \qquad (12)$$

Hence, the equation

$$\hat{y}_o \pm t_{n-p}^{\alpha/2} s \sqrt{1 + \mathbf{f}_o^T (\mathbf{F}^T \mathbf{F})^{-1} \mathbf{f}_o)} \qquad (13)$$

gives the confidence interval $100(1 - \alpha)$ for the predicted value $\hat{y}$. The term $t_{n-p}^{\alpha/2}$ can be found for a given α and the degrees of freedom $n-p$ where p is the number of weights and bias terms employed by the neural network.

Confidence Interval (CI) for the error is shown below.

Table 4 Confidence Interval (70 % and 90 %) for pH, PCO_2, PO_2, HCO_3

confidence interval (CI)	70 % CI	90 % CI
pH	0;0.35	0;0.6
PCO_2	0;5.3	0;10.3
PO_2	0;10.2	0;21.9
HCO_3	0;1.5	0;2.9

The forecast of pH, PCO_2, PO_2, and HCO_3 for newborn is presented below.

The patient, for which result is shown (see Appendix), did not require artificial ventilation. The difference between forecasted and the real values were bigger in other newborn who has required artificial ventilation.

5. RESULTS

It can be seen that results are not quite satisfactory. Such results might be attributed to bad selection of input parameters.

Improvement of the results could be achieved by adding more parameters influencing blood gases factors. Polynomial values of the gases factors could be taken with smaller step. The learning process can be improved due to the fact that decreasing step for polynomial values implies more data vectors. In this case there is about 2500 data vectors. It should be also consider the correctness for selecting patients whose data were used as a sample for the learning process of the neural networks. There is a tendency to increase prognostic error at the end of the approximated sequence, that is related to the nature of approximation methods implemented for the time series. We achieved good prognoses with the error below assumed value for HCO_3 and PCO_2, PO_2. In the case of pH prognosis is under of expectation. This factor is strongly related to PCO_2 and HCO_3. It can be concluded that we have received a good prognosis for two or three parameters. The table shows results of simulation of neural networks predictions for next 6 hours.

Table 5 Mean and std parameter value of pH, PCO_2, PO_2, HCO_3

parameter	*mean* of error	*std* of error	right results (%)
pH	0.03	0.03	76
PCO_2	4.9	6.3	82
PO_2	9.6	14.0	55
HCO_3	1.4	1.4	69

REFERENCES

Bartlett and Downs (1990). *Training a neural networks with a genetic algorithm*, Dept. of Elec. Eng., University of Queensland 1990

Chryssolouris, G. Lee, M. Pierce, J. and Ramsey, W. *Use of neural nelworks for the design of manufacturing systems* Manufacturing Rev. vol. 3, no. 3, pp, l87-194, Sep. l990

Chryssolouris, G., Lee, M. and Ramsey, A. *Confidence Interval Prediction for Neural Network Models* IEEE Tran. on Neural Networks, vol.7 No 1, Jan. l996.

Donaldson, R. and Schnabel, R. B. *Computational experience with confidence regions and confidence intervals for nonlinear least squares* Technometrics, vol. 29, no. 1, pp. 67-82. Feb. 1987

Freund, J. C. and Walpole, R. E. *Mathemutical Statistics* Englewood Cliffs, NJ: Prentice-Hall, 1987.

Graves, S. C. and Redfield, C. H. *Equipment selection and task assignment for multiproduct assembly systcm design* Int. J. Flexib. Manufacturing Syst., vol. 1, pp. 31-50. 1988

Hogg, R. V. and Ledolter, J. *Engineering Statistics* New York Macmillan, 1987

Lippman, R.P. *An introduction to computing with neural nets.* IEEE ASSP Magazine, s. 2-22

Seber, G. A. and Wild, C. J. *Nonlinear Regression* New York: Wiley 1989

APPENDIX

time (min)	forecast value HCO_3	real value HCO_3	forecast value pH	real value pH
10	23	23	7.35	7.35
20	22	22	7.36	7.36
30	20	20	7.28	7.26
40	22	22	7.35	7.3
50	23	23	7.35	7.34
60	22	22	7.36	7.3
70	19	19	7.27	7.24
80	18	18	7.25	7.23
90	19	19	7.33	7.36
100	20	20	7.33	7.36

time (min)	forecast value PO_2	real value PO_2	forecast value PCO_2	real value PCO_2
10	51	61	39	39
20	51	51	38	38
30	46	34	47	50
40	51	51	40	39
50	85	92	44	44
60	51	51	48	48
70	78	80	52	52
80	80	79	47	47
90	44	44	62	68
100	45	44	57	58

time (min)	forecast value HCO_3	real value HCO_3	forecast value pH	real value pH
110	20	20	7.36	7.24
120	17	18	7.19	7.2
130	22	22	7.19	7.2
140	22	22	7.25	7.24
150	22	22	7.26	7.23
160	20	20	7.26	7.24
170	22	22	7.29	7.29
180	20	19	7.33	7.34
190	22	22	7.3	7.27
200	22	22	7.36	7.36
210	23	23	7.31	7.32
220	23	23	7.34	7.34
230	23	23	7.23	7.25
240	20	20	7.26	7.26
250	20	20	7.25	7.25
260	20	20	7.25	7.26
270	21	22	7.29	7.29
280	20	20	7.34	7.3
290	23	24	7.22	7.19
300	24	23	7.26	7.26
310	19	19	7.31	7.25
320	24	24	7.27	7.27
330	23	24	7.36	7.36
340	25	25	7.37	7.37
350	25	24	7.37	7.37
360	23	23	7.36	7.36
370	24	24	7.35	7.35
380	24	24	7.34	7.34
390	22	21	7.31	7.3
400	23	24	7.3	7.27
410	23	24	7.27	7.27
420	24	24	7.3	7.3
430	27	27	7.28	7.25
440	27	28	7.42	7.42
450	24	24	7.36	7.36
460	25	25	7.32	7.32
470	24	24	7.38	7.38
480	25	25	7.34	7.34
490	23	23	7.4	7.4
500	24	24	7.37	7.37
510	20	20	7.35	7.35
520	24	24	7.35	7.32
530	25	26	7.38	7.38
540	23	23	7.38	7.38
550	26	26	7.35	7.35
560	23	23	7.34	7.34
570	24	24	7.33	7.33
580	28	28	7.26	7.26
590	31	28	7.35	7.35
600	31	29	7.39	7.34
610	26	22	7.35	7.34

time (min)	forecast value PO_2	real value PO_2	forecast value PCO_2	real value PCO_2
110	50	46	59	66
120	50	56	45	45
130	80	81	51	59
140	51	51	56	57
150	48	48	49	49
160	37	37	46	46
170	72	72	44	44
180	39	39	48	49
190	42	42	43	39
200	48	49	49	49
210	70	70	40	39
220	46	46	48	48
230	45	45	45	43
240	63	63	54	58
250	50	48	49	49
260	40	36	47	51
270	40	44	50	57
280	58	50	49	53
290	53	52	50	54
300	41	42	50	55
310	50	50	47	50
320	57	57	53	60
330	65	65	48	57
340	44	44	50	62
350	52	52	50	54
360	31	31	44	45
370	63	63	45	44
380	60	53	39	39
390	50	50	42	42
400	53	53	43	43
410	60	63	48	50
420	80	80	42	49
430	45	45	52	54
440	45	45	51	56
450	45	40	56	65
460	43	40	43	43
470	42	42	50	53
480	45	45	44	52
490	52	52	41	45
500	51	51	48	48
510	40	40	40	40
520	40	40	42	42
530	60	62	46	46
540	66	66	39	39
550	46	46	49	51
560	50	50	52	56
570	47	47	43	43
580	50	50	50	52
590	38	39	48	49
600	37	38	57	62
610	39	39	54	56