Comparison of Nonlinear Methods for Hematocrit Estimation from the Transduced Anodic Current Curve

HIEU TRUNG HUYNH¹, JUNG-JA KIM² and YONGGWAN WON¹
¹Department of Computer Engineering, Chonnam National University
300 Yongbong-dong, Buk-gu, Gwangju 500-757
REPUBLIC OF KOREA

²Division of Bionics and Bioinformatics, Chonbuk National University
664-14 St. #1 Dukjin-dong, Dukjin-gu, Chonbuk 561-756
REPUBLIC OF KOREA

Abstract: - Hematocrit (HCT) is expressed as the percentage of red blood cells in the whole blood. It is an important factor for clinical decision marking and known as the most highly influencing factor for glucose measurements by handheld devices. Thus, estimating this factor plays an important role in improving accuracy of glucose measurements. In this paper, we present a comparison of nonlinear methods for hematocrit estimation from the transduced current curve which is produced by glucose-oxidase reaction in strip-type electrochemical biosensors.

Key-Words: - Hematocrit, hematocrit estimation, nonlinear methods, biosensors, transduced current curve

1. Introduction
Hematocrit, one of the primary characteristics in the whole blood, is not only a useful clinical indicator in surgical and hemodialysis but also a factor that significantly affects the accuracy of glucose measurements [1]-[3]. Many studies showed that the glucose results are underestimated at higher hematocrit levels and overestimated at lower hematocrit levels. Estimating hematocrit is one of the most important steps to improve the performance of glucose measurements by portable devices.

The hematocrit can manually determined by centrifugation method. In which a capillary tube called micro-hematocrit tube is filled with blood. When the tube is centrifuged at 10,000RPM for five minutes, the blood is separated into layers. The RBCs with the greatest weight are forced to the bottom of the tube, the WBCs and platelets form a thin layer between the RBCs and the plasma that is the buffy coat, and the top layer is liquid plasma. The hematocrit is measured as the percent of the RBC column to the total blood column. With modern lab equipment, the hematocrit is typically measured from a blood by automated analyzer which can make several other measurements at the same time. In the automated machines the hematocrit is not directly measured, it is calculated by multiplying the red cell count by the mean cell volume.

In addition, the dielectric spectroscopy which is called impedance spectroscopy is also applied to estimate the hematocrit [4]. However, all of the above approaches for hematocrit estimation are quite complicated or require individual devices which can not be used to reduce the effects of hematocrit in glucose measurement by handheld devices. In our studies, we investigated approaches for estimating hematocrit using electrochemical glucose biosensors which is originally designed for glucose measurement with handheld devices. These biosensors use an enzyme to break the blood glucose down and produce ions. These ions are transferred to an electrode to produce a current which is called the transduced current. Based on the calibration curve of produced current, the hematocrit is estimated. In this paper, we present a comparison of nonlinear methods for hematocrit estimation from the calibration curve.

The rest of this paper is organized as follows. Section 2 describes glucose measurements by electrochemical biosensors and the transduced current curve. Comparison of nonlinear methods for hematocrit estimation is shown in section 3, 4, 5 and 6. In section 7, we present experimental results on root mean squares error (RMSE). Finally, we make conclusion in section 8.
2. Transduced Anodic Current Curve

In the glucose measurement process by electrochemical biosensors, the glucose oxidase enzyme in biosensors (GOD) is used to catalyze the oxidation of glucose by oxygen to produce gluconic acid and hydrogen peroxide:

\[ \text{Glucose} + \text{O}_2 + \text{GO/FA} \rightarrow \text{Gluconic acid} + \text{H}_2\text{O}_2 + \text{GO/FADH}_2 \]

\[ \text{GO/FADH}_2 + \text{Ferricinium} \rightarrow \text{GO/FAD} + \text{Ferricinium} \]

Ferrocence

The reduced form of the enzyme (GO/FADH\(_2\)) is oxidized to its original state by an electron mediator (ferrocence). The resulting reduced mediator is then oxidized by the active electrode to produce the transduced anodic current (e\(^-\)). An instance of the transduced anodic current curve obtained during the first 14 seconds using a biochemical glucose biosensor is shown in Fig. 1. The first 8 seconds can be called as incubation time which is time for waiting chemical reaction producing electric signal with high-enough level.

![Figure 1. An example of the transduced current curve](image)

We only concern the second part of the current curve during the last six seconds. During the six seconds in the second part of the curve, the anodic current curve was sampled at a frequency of 10Hz to produce current points. There are 59 current points sampled from the second part of current considered as the input pattern vector for hematocrit estimation.

3. ELM algorithm

The neural network is a powerful nonlinear method for function approximation. The architecture of network can consist of one input layer, one output layer and one or multiple hidden layers. However, in function approximation it was shown that single hidden layer feedforward neural network (SLFN) can approximate any function with arbitrary small error if activation function is chosen properly.

Recently, one of effective training algorithms for SLFNs is extreme learning machine (ELM) proposed by Huang et al. [5]. It can overcome problems in gradient-descent based algorithms such as over-fitting, local minima, learning rate, etc. A salient feature in ELM algorithm is that network weights are determined by non-iterative steps. It tries to find the network parameters that minimize error of equation defined by

\[ \mathbf{HA} = \mathbf{T}, \quad (1) \]

where \( \mathbf{A} \) is output weight matrix, \( \mathbf{T} \) is the desired target vector and \( \mathbf{H} \) is hidden layer output matrix defined by

\[ \mathbf{H} = \begin{bmatrix} f(\mathbf{w}_1 \cdot \mathbf{x}_1 + b_1) & \ldots & f(\mathbf{w}_N \cdot \mathbf{x}_1 + b_N) \\ \vdots & \ddots & \vdots \\ f(\mathbf{w}_1 \cdot \mathbf{x}_N + b_1) & \ldots & f(\mathbf{w}_N \cdot \mathbf{x}_N + b_N) \end{bmatrix}, \quad (2) \]

Note that \( f(\cdot) \) is an activation function, \( \mathbf{x}_i, i=1, 2, \ldots, N \) is input pattern, \( b_i, \mathbf{w}_i, i=1, 2, \ldots, N \), are bias and input weights connecting from the input layer to the \( i \)-th hidden unit. In ELM, the biases and input weights are assigned randomly, output weights are determined by pseudo inverse operation of hidden layer output matrix as follows:

\[ \mathbf{H} = \mathbf{H}^\dagger \mathbf{T}, \quad (3) \]

where \( \mathbf{H}^\dagger \) is Moore-Penrose (MP) generalized inverse of \( \mathbf{H} \). This is the minimum norm least-squares solution of (1). The ELM algorithm can offer good performance at high learning speed in many applications. However, because of randomly selecting, the biases and input weights are non-optimal which results in a large number of hidden units are required. This results in slow respond of trained network to new input patterns. In addition, large memory is also required to save network parameters in devices.

4. RLS-ELM algorithm

Regularized least squares ELM (RLS-ELM) is an improvement of ELM to obtain compact trained networks [6]. From (1), we can approximate \( \mathbf{H} \) by

\[ \mathbf{H} = \mathbf{TA}^\dagger, \quad (4) \]

where \( \mathbf{A}^\dagger \) is MP generalized inverse of \( \mathbf{A} \). Equation (4) can be rewritten as

\[ f(\mathbf{X}\mathbf{W}) = \mathbf{TA}^\dagger, \quad (5) \]
where \( f(XW) = f(XW) = f(w, x + b) \), matrices \( X \) and \( W \) are defined by:

\[
X = \begin{bmatrix} x_1 & x_2 & \ldots & x_N \end{bmatrix}^T,
\]

\[
W = \begin{bmatrix} w_1 & w_2 & \ldots & w_M \end{bmatrix}.
\]

If the function \( f \) is invertible, we can rewrite (5) as follows:

\[
XW = f^\dagger([TA^\dagger]),
\]

where \( f^\dagger([TA^\dagger]) = f^\dagger([TA^\dagger]) \). Let \( P \) defined by

\[
P = T^\dagger f^\dagger([TA^\dagger]),
\]

where \( T^\dagger \) is the MP generalized inverse of \( T \). Equation (6) becomes

\[
XW = TP
\]

A method used for the regularization of ill-posed problems is Tikhonov regularization [7], in which the solution for \( W \) of (8) can be found by minimizing

\[
||XW-TP||^2 + \lambda ||W||^2,
\]

where \( \lambda \) is a positive constant. The solution for (9) is given by

\[
\hat{W} = (X^TX + \lambda I)^{-1}X^TP.
\]

This is the direct solution for \( W \). We can also use an indirect solution given by

\[
\tilde{W} = X^TY,
\]

where

\[
Y = (XX^T + \lambda I)^{-1}TP.
\]

In RLS-ELM, matrix \( P \) is randomly assigned and then it can be used to estimate input weights and hidden layer biases by Eq. (10) or Eq. (11). Whenever these network parameters are computed, the output weights are determined by the MP generalized inverse as given by Eq. (3).

RLS-ELM still assigns random values for matrix \( P \), however, in many real-world applications, the number of outputs \( C \) is much smaller than the number of inputs \( d \). Thus, the number of randomly chosen values for matrix \( P \) \((C \times \bar{N})\) much smaller than \((d+1)\times \bar{N}\) that is used for the original ELM. Especially, in hematocrit estimation, the number of sampled current points is \( d=59 \) while \( C=1 \). Therefore, RLS-ELM assigns randomly \( \bar{N} \) values for matrix \( P \) which is reduced sixty times in comparison with the original ELM.

5. ELS-ELM algorithm

Evolutionary least squares ELM (ELS-ELM) algorithm is another improvement of ELM in order to obtain better performance with compact networks [9]. It is combination of ELM and differential evolution (DE) [10]. Set of input weights and hidden layer biases forms an individual in population:

\[
\theta = \{ w_1^T, w_2^T, \ldots, w_M^T, b_1, b_2, \ldots, b_N \}.
\]

Firstly, the initial generation is generated in which each individual is determined by:

\[
\hat{W} = X^TTP,
\]

where \( X^T \) is MP generalized inverse of \( X \).

The output weights corresponding to each individual are computed by MP generalized inverse. Three steps of DE process are used and individuals with better fitness values are retained for the next generation. The fitness function is chosen as the root mean squares error (RMSE) on the whole training set or the validation set. In summary, the ELS-ELM for SLFNs can be described as follows:

a) Initialization: Generate the initial generation being composed of parameter vectors \( \{ \theta_{i,G} \} \) \( i=1, 2, \ldots, NP \) as the population, where \( NP \) is the population size.

For each individual \( \theta \) in the population, we do

i) Randomly assign the values for the matrix \( P \).

ii) Estimate input weights and hidden layer biases of \( \theta \) by using Eq. 13.

iii) Calculate the hidden-layer output matrix \( H \).

iv) Determine the output weights \( A \) by Eq. 3.

v) Calculate the fitness value.

b) Training process:

At each generation \( G \), we do:

i) Mutation: the mutant vector is generated as

\[
\nu_{l,G+1} = \theta_{l,G} + F(\theta_{l,G} - \theta_{j,G}), \quad \text{where} \ r1, r2, r3 \ \text{are different random indices and} \ F \ \text{is a constant factor used to control the amplification of the differential variation.}
\]

ii) Crossover: the trial vector is formed so that

\[
\mu_{l,G+1} = \begin{cases} 
\nu_{l,G+1} & \text{if} \ rand \ b(j) \leq CR \ \text{or} \ j = rnbv(i) \\
\theta_{l,G} & \text{if} \ rand \ b(j) > CR \ \text{and} \ j \neq rnbv(i)
\end{cases}
\]

where \( rand \ b(j) \) is the \( j \)-th evaluation of a uniform random number generator, \( CR \) is the crossover constant and \( rnbv(i) \) is a randomly chosen index which ensures at least one parameter from \( v_{l,G+1} \).
iii) Determine the output weights by Eq.3. iv) Evaluate the fitness for each individual. v) Selection: The new generation is determined by:

\[
\theta_{i,j}\in \begin{cases} 
\mu_{i,j} & \text{if } \varphi(\theta_{i,j}) - \varphi(\mu_{i,j}) > \epsilon \varphi(\theta_{i,j}), \\
\mu_{i,j} & \text{if } \varphi(\theta_{i,j}) - \varphi(\mu_{i,j}) < \epsilon \varphi(\theta_{i,j}) \\
\theta_{i,j} & \text{otherwise},
\end{cases}
\]

where \(\varphi(.)\) is the fitness function and \(\epsilon\) is a predefined tolerance rate. The DE process in the training process is repeated until the goal is met or a maximum learning epochs is completed. This algorithm can obtain compact networks and the performance for hematocrit estimation can be improved.

6. Support Vector Machine (SVM)

Support vector (SV) algorithm is a nonlinear generalization based the framework of statistical learning theory. One of approaches for function approximation based on SVM is \(\varepsilon\)-SV regression. The main goal in \(\varepsilon\)-SVR is to find a function \(\varepsilon\) that has most \(\varepsilon\) deviation from the actually obtained targets \(t_i\) for all training data, and as flat as possible. In the cases where \(\varepsilon\) is a linear function, the problem of finding \(\varepsilon\) can be written formally as:

\[
\begin{align*}
\min_{\theta} & \quad \frac{1}{2} \|w\|^2 \\
\text{subject to} & \quad t_j - <w, x_j> - b \leq \varepsilon \\
& \quad <w, x_j> + b - t_j \leq \varepsilon
\end{align*}
\]

(14)

In situations where some errors in approximation are allowed, a method based on “soft margin” is proposed [11], in which slack variables \(\xi_j, \xi^*_j\) are introduced to cope with otherwise infeasible constraints of the optimization (14). The problem can be expressed by:

\[
\begin{align*}
\min_{\theta} & \quad \frac{1}{2} \|w\|^2 + C \sum_{j=1}^{N} (\xi_j + \xi^*_j) \\
\text{subject to} & \quad t_j - <w, x_j> - b \leq \varepsilon + \xi_j \\
& \quad <w, x_j> + b - t_j \leq \varepsilon + \xi^*_j
\end{align*}
\]

(15)

where \(C>0\) is a constant. This problem can be converted to the dual optimization problem that

\[
\begin{align*}
\max_{\alpha} & \quad \frac{1}{2} \sum_{j=1}^{N} (\alpha_j - \alpha^*_j)(\alpha_j - \alpha^*_j) <x_j, x_j> \\
\text{subject to} & \quad \sum_{j=1}^{N} (\alpha_j - \alpha^*_j) = 0 \\
& \quad 0 \leq \alpha_j, \alpha^*_j \leq C
\end{align*}
\]

(16)

and \(b\) can be computed by:

\[
\begin{align*}
b &= t_j - <w, x_j> - \varepsilon & \text{for } \alpha_j \in (0, C) \\
b &= t_j - <w, x_j> + \varepsilon & \text{for } \alpha^*_j \in (0, C)
\end{align*}
\]

(17)

In the cases which are not possible to have a linear function on the training data, a nonlinear mapping can be applied in order to map the data into other feature space where the linear model can be used. In addition, we can use an approach via kernels defined by:

\[
K(x_1,x_2) = <\Phi(x_1), \Phi(x_2)>
\]

(18)

The solution for \(w\) in this case is given by

\[
w = \sum_{j=1}^{N} (\alpha_j - \alpha^*_j)\Phi(x_j)
\]

(19)

7. Experimental Results (Comparison of RMSE)

In this section, we present comparison on RMSE of methods for hematocrit estimation. The dataset was obtained from 199 blood samples that were randomly selected volunteers. The desired hematocrit values are measured from accurate machine in hospital. The distribution of accurate hematocrit values is shown in Fig. 2 with mean value of 36.02 and deviation of 6.39. This distribution is fairly representing the trend of hematocrit values for human. The input features were normalized into the range [0, 1]. The dataset was divided into two subsets: the training set was 40 percent of the data set and the remaining 60 percent was used for blind test.
8. Conclusion

In this paper, a comparison of nonlinear methods for hematocrit estimation is presented. They can achieve acceptable performance for hematocrit estimation from the transduced current curve. In comparison with ELM, RLS-ELM and ELS-ELM can obtain good performance with compact networks which can save memory in hardware implementation and make the trained networks responding fast to the new input patterns. Experimental results also show that SVM can obtain slightly better RMSE than SLFNs.

References:


