

# Combination of M-estimators and neural network model to analyze inside/outside bark tree diameters

Kyriaki Kitikidou<sup>1</sup>, Elias Milios<sup>2</sup>, Lazaros Iliadis<sup>3</sup>, Minas Kaymakis<sup>4</sup>

<sup>1,2,3,4</sup>Democritus University of Thrace, Department of Forestry and Management of the Environment and Natural Resources, Pandazidou 193, 68200, Orestiada, Greece

<sup>1</sup><kkitikid@fmenr.duth.gr>

**Abstract.** One of the most important statistical tools is linear regression analysis for many fields such as medical sciences, social sciences, econometrics and more. Regression techniques are commonly used for modelling the relationship between response variables and explanatory variables. In this study, inside bark tree diameter was used as the dependent variable and outside bark diameter and site type as independents. While generally it is assumed that inside and outside bark diameters are linearly correlated, linear regression application is weak in the presence of outliers. The purpose of this study was to develop a Multi-Layer Perceptron neural network model which considered significant variables from an a priori developed robust regression model. The application of robust regression could be considered in selecting the input variables in a neural network model.

**Keywords:** Artificial Neural Networks; M-Estimation; Robust Regression.

## 1 Introduction

In experimental science measurements are typically repeated for a number of times, yielding a sample size  $n$ . Then, we summarize the measurements by a central value and we measure their variability, i.e. we estimate location and scale. These estimates should preferably be robust against outliers. The estimator's stylized empirical influence function should be smooth, monotone increasing for location, and decreasing-increasing for scale. When the values of the observations tend to depart from the main body of data, maximum likelihood estimators (M-estimators) can be considered.

According to Loetsch, Zöhrer and Haller (1973), the relation between inside and outside bark tree diameter is expressed by the simple linear regression model:

$$\hat{d}_{in} = b_0 + b_1 d$$

where  $\hat{d}_{in}$  is the estimated breast height (1.3 m above the ground) diameter inside bark,  $d$  is the breast height diameter outside bark, and  $b_0, b_1$  regression coefficients.

The aim of the current study is to incorporate variables that are found to be significant in a robust regression model, into a Multi-Layer Perceptron (MLP) Artificial Neural Network (ANN) model, so as to estimate the same dependent variable (diameter inside bark). We find an attempt to combine robust estimation and neural networks in several researches (Chen 1994, Lee 1999, Suykens et al. 2002, Kuana and Whiteb 2007, Aleng et al. 2012). In this study, we used inside and outside bark tree diameters to illustrate the pre-mentioned idea, assuming that the two variables are already linearly related but simple linear regression is challenging to apply, because of the presence of outliers.

## 2 Materials and Methods

Data used in this study were collected from 30 *Populus tremula* dominant trees on Athos peninsula (northern Greece), simply randomly selected (Kitikidou et al. 2012). Trees were selected from two site types. Site I was the most productive site, while site II was the less productive. For all the analyses the SPSS v.19 statistical package was used (IBM SPSS 2010). Descriptive statistics for inside and outside bark breast height (1.3 m above the ground) diameters are given in **Table 1**.

**Table 1.** Descriptive statistics for all variables used in the analysis.

			Mean	Standard Deviation	Minimum	Maximum
Site Good	diameter outside bark (cm)		25.697	3.649	19.500	31.600
	diameter inside bark (cm)		24.825	3.471	18.950	30.250
Bad	diameter outside bark (cm)		14.195	3.121	10.350	18.900
	diameter inside bark (cm)		13.513	3.030	9.775	18.150

### 2.1 Robust Regression

Robust regression is a weighted average of independent parametric and nonparametric fits to the data. A statistical procedure is considered robust if it performs reasonably well, even when the assumptions of the statistical model are not met. If we assume that dependent and independent variables are linearly correlated, least squares estimates and tests perform quite well, but they are not robust in the presence of outliers in the data set (Rousseeuw and Leory 1987)

Robust regression analyses have been developed as an improvement to least squares estimation in the presence of outliers. This method does not exclude the outliers, which can be real observations, but tends to down-weight cases with large residuals (McKean 2004). The primary purpose of robust regression analysis is to fit a

model which represents the majority of the data. Robust regression is an important tool for analyzing data that are contaminated with outliers. This method can be used to detect outliers and to give results resistant in the presence of outliers. Many methods have been developed to deal with such problems. In this case, we proposed the robust M-estimation to model response data, since linear regression between inside and outside bark diameters detected outliers in the data set.

Robust regression is an alternative of least squares estimation regression analysis designed to avoid limitations of traditional parametric analyses. The least squares estimation in a regression model is not sufficiently robust to extreme data. It is quite common for different nature of data to separate distinctly from the data distribution. In such cases, the influence of extreme data can be minimized using robust M-estimators.

Huber (1973) proposed an M-estimator that uses maximum likelihood formulations by deriving optimal weighting for the data set in non-normal conditions. In other words, the M-estimator minimizes the role of the residuals. As in the case of M-estimation of location, the robustness of the estimator is determined by the choice of weight function.

Consider the linear model

$$Y_i = a + \beta_1 x_{i1} + \beta_2 x_{i2} + \dots + \beta_n x_{in} + \varepsilon_i = x_i' \beta + \varepsilon_i \quad (1)$$

for the  $i$ th of  $n$  observations. The fitted model is

$$Y_i = b_0 + b_1 x_{i1} + b_2 x_{i2} + \dots + b_n x_{in} + e_i = x_i' b + e_i \quad (2)$$

The Huber M-estimators minimize the objective functions

$$\sum_{i=1}^n \rho(e_i) = \sum_{i=1}^n \rho(Y_i - x_i' b) \quad (3)$$

where the function  $\rho(\cdot)$  is usually chosen in a way that represents some weighting of the  $i$ th residual. This weighting means that outlying observations have their weights reduced and thus the estimates are affected less by such noise. A weighting of zero means that the observation is classified as an outlier.

Differentiating (1) with respect to the regression coefficients  $\hat{\beta}_j$  yields

$$\sum_{i=1}^n \psi(e_i) x_i = 0 \quad (4)$$

where  $\psi(\cdot)$  is the derivative of  $\rho(\cdot)$  and the corresponding M-estimator is the maximum likelihood estimator. If we define the weight function  $w(e_i) = \frac{\psi(e_i)}{e_i}$ , then the

estimating equations can be written as

$$\sum_{i=1}^n w(e_i) x_i = 0 \quad (5)$$

Solving the estimating equations is similar to a weighted least-squares estimation. The weights, however, depend upon the residuals, the residuals depend upon the esti-

mated coefficients, and the estimated coefficients depend upon the weights. An iterative solution (called iteratively reweighted least-squares (IRLS)) is therefore required.

## 2.2 Multi-Layer Perceptron (MLP) Artificial Neural Network (ANN)

Consider a neural model, which consists of an input layer, one or several hidden layers and an output layer. The neurons in the neural network are generally grouped into layers. Signals flow in one direction from the input layer to the next, but not within the same layer. Success of the application of a neural network relies on the training network. Among the several learning algorithms available, back-propagation has been the most popular and most widely implemented. Basically, the back-propagation training algorithm with three-layer architecture means that, the network has an input layer, one hidden layer and an output layer (Bishop 1995; Ripley 1996; Haykin 1998; Fine 1999).

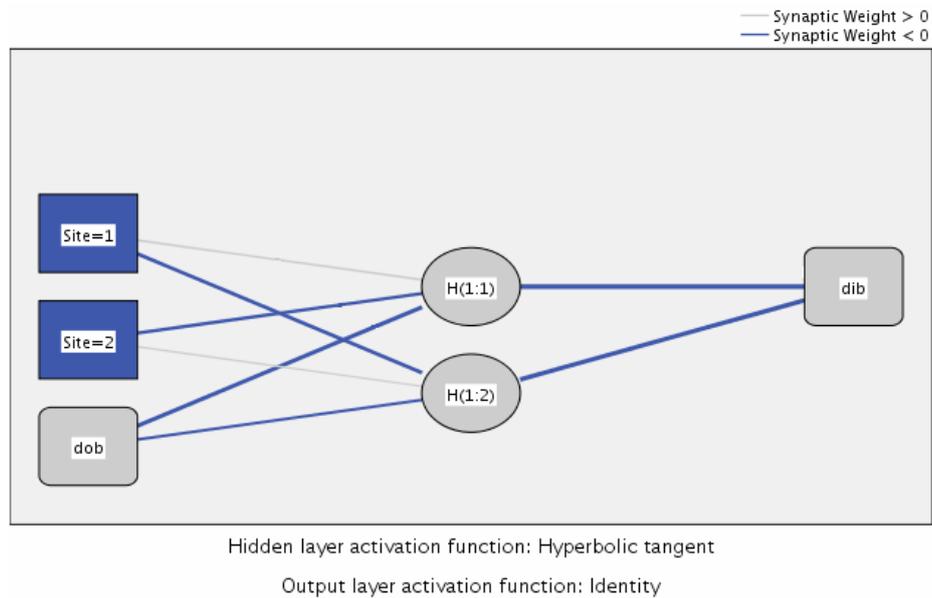
In this research the hidden layer is fixed at one in order to reduce the complexity of the network, and to increase its computational efficiency (Haykin 1998). For the network with  $N$  input nodes,  $H$  hidden nodes and one output node, the values  $\hat{Y}$  are given by:

$$\hat{Y} = g_2 \left( \sum_{j=1}^H w_j h_j + w_0 \right) \quad (6)$$

where  $w_j$  is an output weight from hidden node  $j$  to output node,  $w_0$  is the bias for output node, and  $g_2$  is an activation function. The values of the hidden nodes  $h_j$ ,  $j=1, \dots, H$  are given by:

$$h_j = g_1 \left( \sum_{i=1}^N v_{ji} X_i + v_{j0} \right), j=1, \dots, H \quad (7)$$

where  $v_{ji}$  is the input weight from input node  $i$  to hidden node  $j$ ,  $v_{j0}$  is the bias for hidden node  $j$ ,  $X_i$  are the independent variables where  $i=1, \dots, N$  and  $g_1$  is an activation function. The schematic representation of the neural network used in this study is given in **Fig. 1**.



**Fig. 1.** Multi-layer perceptron network structure with one hidden layer including two neurons (H(1:1), H(1:2)), two input variables (Sites 1 and 2, dob: diameter outside bark) and one output variable (dib: diameter inside bark).

From the initial 30 data records (30 *Populus tremula* trees), 70% were used for training (21 records) and 30% for testing (9 records).

### 3 Results-Discussion

#### 3.1 Results of robust regression analysis

We applied robust regression analysis as an alternative to the least squares regression model, because fundamental assumptions weren't met due to the nature of the data. R squared value was calculated at 0.9996, indicating a great ability of the model to predict a trend. **Table 2** displays the final weighted least square estimates. The weighted estimates provide resistant results against outliers. As we see in this Table, both independent variables differ significantly from zero (sig. <0.05). That leads us in using both variables for the inside bark diameter estimation through the MLP neural network.

**Table 2.** Parameter estimates from robust regression model.

Parameter	B	Std. Error	95% Wald		Hypothesis Test		
			Confidence Interval		Wald Chi-Square	df	Sig.
			Lower	Upper			
(Intercept)	-0.098	0.1080	0-.310	0.114	0.825	1	0.364
[Site=1]	0.283	0.0923	0.102	0.464	9.392	1	0.002
[Site=2]	0 <sup>a</sup>	.	.	.	.	.	.
dob	0.959	0.0077	0.944	0.974	15558.112	1	0.000
(Scale)	0.018						

Dependent Variable: diameter inside bark (cm)

Model: (Intercept), Site, dob (diameter outside bark)

a. Set to zero because this parameter is redundant.

### 3.2 Results of MLP neural network

The input variables are selected based on the significant variables of the robust regression model (sites and diameter outside bark). Back propagation algorithm was applied and by following trial and error approach we ended up in two neurons included in the single hidden layer. Hence, the appropriate neural network architecture which results in the best multilayer neural network model is composed of two input variables, two hidden nodes and one output node. This can be represented by equations (6) and (7).

Training this architecture using the Levenberg-Marquardt back propagation training algorithm and using hyperbolic tangent transfer function in the hidden layer and identity function in the output layer, the  $R^2$  for training and testing data were equal to 0.99964 and 0.99955, respectively, showing that the model gives very accurate estimations of the inside bark diameter (**Table 3**). This procedure was completed in just one step, as we can see in iteration history (**Table 4**).

**Table 3.** Model summary of the MLP ANN model.

Training	Sum of Squares Error	0.012
	Root Mean Squared Error (RMSE)	0.032
	R square	0.99964
Testing	Sum of Squares Error	0.003
	Root Mean Squared Error (RMSE)	0.045
	R square	0.99955

Dependent Variable: diameter inside bark (cm)

**Table 4.** Iteration history of the MLP ANN model.

Iteration	Update Type	Parameter			
		(Intercept)	[Site=1]	dob	(Scale)
0	Initial	-0.098241	0.282914	0.958909	0.016168
1	Scoring <sup>a</sup>	-0.098143	0.282807	0.958892	0.017964

Redundant parameters are not displayed. Their values are always zero in all iterations.

Dependent Variable: diameter inside bark (cm)

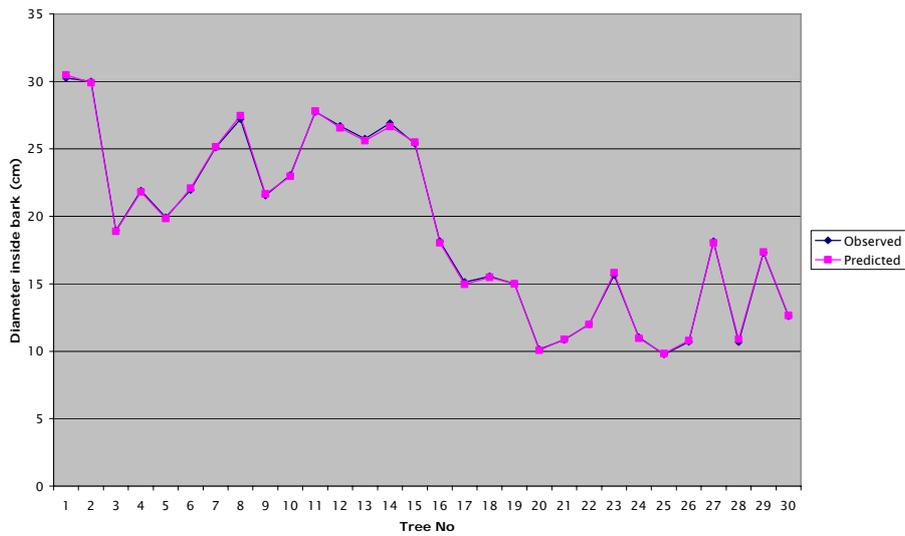
Model: (Intercept), Site, dob

a. All convergence criteria are satisfied.

Parameter estimates of the MLP neural network are given in **Table 5**. The observed and predicted values of the dependent variable in **Fig. 2** show clearly that the MLP network does an exceptional job in predicting inside bark diameters.

**Table 5.** Parameter estimates from the MLP ANN model.

Predictor		Predicted		
		Hidden Layer 1		Output Layer
		H(1:1)	H(1:2)	dib
Input Layer	(Bias)	-0.061	0.226	
	[Site=1]	0.187	-0.386	
	[Site=2]	-0.328	0.162	
	dob	-0.638	-0.306	
Hidden Layer 1	(Bias)			-0.046
	H(1:1)			-1.264
	H(1:2)			-1.108



**Fig. 2.** Observed and predicted values of the MLP ANN.

## 4 Conclusions

The MLP neural network model for inside bark diameter estimation, with selection of input variables from a robust regression model, discovered two input variables;

diameter outside bark and site type. The  $R^2$  for training and testing were 0.99964 and 0.99955, respectively, which are quite large. Hence, significant variables from a robust regression, when we prefer to avoid ordinary least squares estimation in the presence of outliers, could be considered in selecting the input variables of a neural network model.

Artificial neural networks have shown a great promise in predicting variables in many scientific fields, because of their fast learning capacity. However, when the training patterns incur large errors caused by outliers, the networks interpolate the training patterns incorrectly. Using significant variables from robust procedures as input variables will result in better capability of approximation to underlying functions and faster learning speed (Lee 1999). In this way, a neural network can be robust against gross errors and have an improved rate of convergence, since the influence of incorrect samples is notably suppressed (Chen 1994)

## 5 References

1. Aleng, N., Mohamed, N., Ahmad, W., Naing, N.: A new strategy to analyze medical data using combination of M-estimator and Multilayer Feed-Forward Neural Network Model. *European Journal of Scientific Research* 1: 79-85 (2012)
2. Bishop, C.: *Neural Networks for Pattern Recognition*, 3<sup>rd</sup> ed. Oxford University Press, Oxford (1995)
3. Chen, D.: A robust backpropagation learning algorithm for function approximation. *IEEE Transactions on Neural Networks* 5(3): 467-479 (1994)
4. Fine, T.: *Feedforward Neural Network Methodology*, 3<sup>rd</sup> ed. Springer-Verlag, New York (1999)
5. Haykin, S.: *Neural Networks: A Comprehensive Foundation*, 2<sup>nd</sup> ed. Macmillan College Publishing, New York (1998)
6. Huber, P.: Robust regression: Asymptotics, conjectures and Monte Carlo. *The Annals of Statistics* 1: 799-821 (1973).
7. IBM SPSS Neural Networks 19. SPSS Inc (2010)
8. Kitikidou, K., Kaymakis, M., Milios, E.: Site index curves for young *Populus tremula* stands on Athos peninsula (northern Greece). *Turkish Journal of Agriculture and Forestry* 36: 55-63 (2012)
9. Kuana, Ch., Whiteb, H.: Artificial neural networks: an econometric perspective. *Econometric Reviews* 13(1): 1-91 (1994)
10. Lee, Ch.: Robust radial basis function neural networks. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics* 29(6): 674-685 (1999)
11. Loetsch, F., Zöhrer, F, Haller, K.: *Forest Inventory, Vol. II*. BLV Verlagsgesellschaft, Munchen (1973)
12. McKean, J.: Robust Analysis of Linear Models". *Statistical Science* 19(4): 562-570 (2004)

13. Ripley, B.: Pattern Recognition and Neural Networks. Cambridge University Press, Cambridge (1996)
14. Rousseeuw, P., Leory, A.: Robust regression and outlier detection. Wiley Series in Probability and Statistics (1987)
15. Suykens, J., De Brabanter, J., Lukas, L., Vandewalle, J.: Weighted least squares support vector machines: robustness and sparse approximation. Neurocomputing 48(1-4): 85-105 (2002)