Journal of Statistical
Theory & Methods
1996, Vol.7, pp 203~210

An Alternative Method of Regression: Robust Modified Anti-Hebbian Learning

Changha Hwang

Abstract A linear neural unit with a modified anti-Hebbian learning rule has been shown to be able to optimally fit curves, surfaces, and hypersurfaces by adaptively extracting the minor component of the input data set. In this paper, we study how to use the robust version of this neural fitting method for linear regression analysis. Furthermore, we compare this method with other methods when data set is contaminated by outliers.

Keywords: Hebbian learning, Minor components, Robust, Total least square method.

1. Introduction

The most commonly used regression method is that of least squares(LS). It was discovered independently by C. F. Gauss in Germany around 1795 and by A. M. Legendre in France around 1805. Early applications of the method were to astronomic and geodetic data. Its first published appearance was in 1805 in an appendix to a book by Legendre on determining the orbits of comets. Since its discovery almost 200 years ago, LS method has been the most popular method of regression analysis. Over the last two or three decades, however, there has been increasing interest in other methods. This is due partly to discoveries of deficiencies in the LS method and partly to advances in computer technology, which have made the computational complexity of other methods a relatively unimportant consideration. Numerous research articles have now been published on alternative approaches to regression analysis. In many cases, LS method is suboptimal, and the optimal LS method is the so called total least squares(TLS) method. In this paper, we will discuss in some detail that the problems of optimal fitting in TLS sense can be described as minor component analysis(MCA) problem and a linear neural unit with modified anti-Hebbian learning rule can solve the problems.

Neural networks have been vigorously promoted in the computer science literature for tackling a wide variatey of scientific problems. Recentely, investigations have started to see how useful neural networks are for tackling statistical questions in general, and for

This work was supported in part by Catholic University of Taegu-Hyosung Grant. Changha Hwang is a professor of the Department of Statistics, Catholic University of Taegu-Hyosung, Kyungbuk, 712-702.

nonlinear regression modelling in particular. By the way, when there is a strong evidence on that linear regression explains very well the given data, we had better use linear regression instead of using nonlinear regression. Therefore, it is pertinent to investigate how well a linear neural unit with modified anti-Hebbian learning(MAHL) rule and its robust version perform. In this paper we investigate the performance of robust modified anti-Hebbian learning(RMAHL) rule for the data set with or without some outliers. Section 2 is a review of TLS method and MCA. Section 3 describles a RMAHL rule and its relationship with MCA. In section 4, through three examples we compare RMAHL rule with other method: least-absolute-deviations, robust M-, nonparametric rank-based and ridge regression.

2. Total Least Squares Method

The following brief account of TLS method is intended to make this paper as self-contained as possible. However the reader may find it helpful to read Xu et al. (1992). The LS method is the most commonly used one to fit a given data set. For example, given a data set $D=\{(y_i,x_i),i=1,2,\cdots,n\}$, the problem of using a line model $y=\alpha+\beta x$ to fit D in the usual LS sense becomes the problem of finding a pair of estimates $\hat{\alpha}$, $\hat{\beta}$ such that

$$E(\hat{\alpha},\hat{\beta}) = \min_{\alpha,\beta}\sum_{i=1}^{n}(y_i-\alpha-\beta x_i)^2. \tag{1}$$

In fact, this implicitly assumes that only the measurements y_i contain errors while the measurements x_i are accurate. However, in many applications such as in image recognition and computer vision, all the measurements contain a certain degree of errors. In such cases, a line $y=\hat{\alpha}+\hat{\beta}x$ obtained in the usual LS sense is not optimal. The optimal way should be to minimize the sum of the squared lengths of all the bars which are perpendicular to the estimated line.

$$E(\hat{\alpha},\hat{\beta}) = \min_{\alpha,\beta}\sum_{i=1}^{n}\frac{(y_i-\alpha-\beta x_i)^2}{1+\beta^2}. \tag{2}$$

This is the basic idea of the so called TLS method. In comparison with the usual LS method, to obtain the solution of TLS is generally quite burdensome. The computations for TLS are more involved than for LS. Equation (1) can be reduced to a problem of solving linear equations, while equation (2) results in a problem of solving a third order nonlinear equation of β. For the more general case which involve a large number of variable, the problem becomes more complicated. This is probably why TLS has not been as widely used as the usual LS method although the basic idea of TLS was

proposed long ago.

In the case of line or hyperplane fitting, when the line or hyperplane models can be reexpressed as

$$a_1x_1 + a_2x_2 + c_0 = 0, \tag{3}$$

$$a_1x_1 + a_2x_2 + \cdots + a_px_p + c_0 = 0 \tag{4}$$

where x_i, $i = 1,\ldots,p$ are variables and c_0 is an arbitrary constant. Again let us take the problem of line fitting as an example. For equation (3), the TLS fitting problem is to minimize the following total least square error

$$E = \sum_{i=1}^{n} r_i^2, \; r_i = \left|a_1x_1^{(i)} + a_2x_2^{(i)} + c_0\right| / \sqrt{a_1^2 + a_2^2} \tag{5}$$

Let $\mathbf{a} = [a_1, a_2]^T$ and $\mathbf{x}_i = [x_1^{(i)}, x_2^{(i)}]^T$. Then E can be further reexpressed as

$$\begin{aligned} E &= \sum_{i=1}^{n} \frac{\left(\mathbf{a}^T\mathbf{x}_i + c_0\right)^2}{\mathbf{a}^T\mathbf{a}} = n\frac{\mathbf{a}^TR\mathbf{a} + 2c_0\mathbf{a}^T\mathbf{e} + c_0^2}{\mathbf{a}^T\mathbf{a}}, \\ R &= \frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i\mathbf{x}_i^T, \; \mathbf{e} = \frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i, \end{aligned} \tag{6}$$

where $\mathbf{e}, R$ are the mean vector and the autocorrelation matrix of data set D. From $dE/d\mathbf{a} = \mathbf{0}$, the critical points of E should satisfy

$$R\mathbf{a} + c_0\mathbf{e} - \lambda\mathbf{a} = \mathbf{0}, \; \lambda = \frac{\mathbf{a}^TR\mathbf{a} + 2c_0\mathbf{a}^T\mathbf{e} + c_0^2}{\mathbf{a}^T\mathbf{a}}. \tag{7}$$

In general, equation (7) is difficult to solve because it is a third order matrix equation. Here, we use a special strategy for solving the equation. First, by taking expectation on both sides of equation (4), we can obtain

$$c_0 = -\mathbf{a}^T\mathbf{e}, \tag{8}$$

Then, substituting (8) into (7) and simplifying yield

$$\sum\mathbf{a} - \lambda\mathbf{a} = 0, \; \lambda = \frac{\mathbf{a}^T\sum\mathbf{a}}{\mathbf{a}^T\mathbf{a}} \tag{9}$$

where $\sum = R - \mathbf{e}\mathbf{e}^T$ is the covariance matrix of data set D. So, we see that the TLS porblem can be reduced to the problem of finding the minimum eigenvalue and its corresponding normalized eigenvector of matrix $\sum$, or in other words, finding the first minor component of data set D.

It is not difficult to see that for plane and hyperplane expressed by equation (4), if we let $\mathbf{a} = [a_1, a_2, \ldots, a_p]^T$ and $\mathbf{x}_i = [x_1^{(i)}, x_2^{(i)}, \ldots, x_p^{(i)}]$, equations (6), (7), (8), and (9) will also

hold. In general, the same technique applies to curves and hypersurfaces expressed as

$$a_1 f_1(\mathbf{x}) + a_2 f_2(\mathbf{x}) + \cdots + a_m f_m(\mathbf{x}) + c_0 = 0, \quad \mathbf{x} = [x_1, x_2, \ldots, x_p]^T,$$

where $f_i(\mathbf{x})$ is a function of $\mathbf{x}$. For example, quadratic curves are expressed as

$$a_1 x_1^2 + a_2 x_1 x_2 + a_3 x_2^2 + a_4 x_1 + a_5 x_2 + c_0 = 0.$$

If we first transform each $\mathbf{x}_i$ into $f_i = [f_1(\mathbf{x}_i), f_2(\mathbf{x}_i), \ldots, f_m(\mathbf{x}_i)]^T$, we can obtain the same equations as equations (6), (7), (8), and (9) for the problems of TLS hypersurface fitting.

3. Robust Modified Anti-Hebbian Learning

In the previous section, it is illustrated that the TLS method reduces to finding the minimum eigenvalue and its corresponding normalizaed eigenvector for the covariance matrix of the data. In other words, the problem reduces to finding the minor component of the data set. Here, we explain how to obatin the minor component by neural network.

Consider the liner neural unit with inputs $\mathbf{x}(t) = [\xi_1(t), \cdots, \xi_p(t)]^T$, weights $\mathbf{m}(t) = [\mu_1(t), \cdots, \mu_p(t)]^T$ and output

$$y(t) = \sum_{i=1}^{p} \mu_i(t)\xi_i(t) = \mathbf{m}^T(t)\mathbf{x}(t)$$

It has been shown (Oja 1982) that by using a constrained Hebbian learning rule as follows

$$\mu_i(t+1) = \mu_1(t) + \alpha(t)(y(t)\xi_i(t) - y^2(t)\mu_i(t)) \text{ or}$$

$$\mathbf{m}(t+1) = \mathbf{m}(t) + \alpha(t)(y(t)\mathbf{x}(t) - y^2(t)\mathbf{m}(t)) \tag{10}$$

the unit learns to function as a principal component analyzer of the stationary input vector stream $\mathbf{x}(t)$, in the sense that the weight vector $\mathbf{m}(t)$ tends asymptotically to the principal eigenvector of the input data correlation matrix. In (10), $\alpha(t)$ is a positive scalar gain parameter that must be chosen in a suitable way.

Here, we still use the same linear unit but change equation (10) into a constrained anti-Hebbian learning rual given either by

$$\mathbf{m}(t+1) = \mathbf{m}(t) - \alpha(t)y(t)[\mathbf{x}(t) - y(t)\mathbf{m}(t)] \tag{11}$$

or its normalized version

$$\mathbf{m}(t+1) = \mathbf{m}(t) - \alpha(t)y(t)[\mathbf{x}(t) - \frac{y(t)\mathbf{m}(t)}{\mathbf{m}^T(t)\mathbf{m}(t)}]. \tag{12}$$

By expanding $\mathbf{m}(t)$ in terms of eigenvectors, the following theorem can be proven. See for details Xu et al. (1992).

Theorem 1. Let $R = E(\mathbf{x}^T\mathbf{x})$ be positive semidefinite autocorrelation matrix with the minimum eigenvalue of multiplicity one, and let $\lambda_{\min}$ and $\mathbf{c}_{\min}$ be the minimum eigenvalue and its corresponding normalized eigenvector of R. If $\mathbf{m}(0)^T\mathbf{c}_{min} \neq 0$, then

$$\lim_{t\to\infty}\mathbf{m}(t) = \mathbf{c}_{\min} \text{ or } -\mathbf{c}_{\min}$$

$$\lim_{t\to\infty}\mathbf{m}(t)^T R\,\mathbf{m}(t) = \lambda_{\min} = \min\{\mathbf{m}^T R\,\mathbf{m}\}.$$

We will see that the neural unit with learning equation either equation (11) or (12) can be directly used as an optimal TLS fitting analyzer if $\mathbf{e} = E(\mathbf{x}) = \mathbf{0}$ for input data set D. In this case, equation (8) shows that $\mathbf{e} = \mathbf{0}$ results in $c_0 = 0$.

In the case that $\mathbf{e} = E(\mathbf{x}) \neq \mathbf{0}$, the above unit can not be used directly. However, noticing that $\sum = R - \mathbf{e}\mathbf{e}^T = E[(\mathbf{x}-\mathbf{e})(\mathbf{x}-\mathbf{e})^T]$, we see that a slight preprocessing of subtracting $\mathbf{e}$ from each data point can make all the above discussions remain true. The only extra issue here is that the representation of the fitted hyperplane needs not only the obtained final solution $\mathbf{a}$ alone but also an accompanied parameter $c_0 = -\mathbf{m}^T\mathbf{e} \neq 0$.

However,almost all the MCA algorithms are based on the assumption that data have not been spoiled by outliers. In practice, real data often contain from the data set. In what follows, a robust version of modified anti-Hebbian learning rule will be discussed. See for details Xu and Yuille(1995). Here we still use the same linear unit but different learning rule given by

$$\mathbf{m}(t+1) = \mathbf{m}(t) - \alpha(t)\frac{1}{1+\exp(\beta(z(\mathbf{x}(t),\mathbf{m}(t)) - \eta))}[y(t)\mathbf{x}(t) - y^2(t)\mathbf{m}(t)]$$

where $z(\mathbf{x}(t),\mathbf{m}(t)) = \mathbf{x}^T(t)\mathbf{x}(t) - (\mathbf{m}^T(t)\mathbf{x}(t)\mathbf{x}^T(t)\mathbf{m}(t))/\mathbf{m}^T(t)\mathbf{m}(t)$. Notice that, as $t\to\infty$, $\alpha(t)\to 0$ and $\beta\to\infty$.

4. Numerical Illustrations

In Section 2 we see that the TLS method reduces to finding the eigenvector corresponding to the minimum eigenvalue for the covariance matrix of the data. In addition, we realize that we can use the commonly used methods in Statistics to obtain this eigenvector, even when there are some outliers in the data set. The preliminary simulation study was conducted to see how well the RMAHL rule works for the artificial data with zero mean vector. Here, this artificial data set contained some outliers. According to this study, this learning rule worked reasonably well compared with the widely used methods in Statistics.

To get a more concrete idea of how the RMAHL rule performs in the regression problem on real data, let us look at three examples. For the comparisons on three data sets, we use LS regression, least-absolute-deviations(LAD) regression, M-regression, nonparametric regression and ridge regression besides the robust modified anti-Hebbian learning. Three data sets are acid content data, turnip green data and stack loss data. See for details Birkes and Dodge(1993). Some results are taken from their book.

By the way, we should do a slight preprocessing of subtracting mean vector from each data point since it is not zero for three data sets used for the comparisons. We use a 10% trimmed mean vector to apply RMAHL rule. See for the reason Seber(1984).

Example 1. Consider the acid contend data with no outliers. We see from Birkes and Dodge (1993) that all the data points fall closely around a straight line. For such a well-behaved data set, all the regression methods give very similar results. LS estimates of β_0, β_1 are 35.46 and 0.3216, respectively. Compared to the LS estimates of β_0 and β_1, the LAD estimates were within 2%, the M-estimates were exactly the same, the nonparametric estimates were within 1%, and the ridge estimates differed only in the fourth significant digit. RMAHL estimates of β_0, β_1 are 35.61 and 0.3219, respectively. MAHL estimates are very similar to RMAHL estimates. Here, $a = 0.01, \beta = 500$, and $\eta = 0.02$ have been used to obtain RMAHL estimates. The estiamtes of σ were not as close; they were 1.230, 1.433, 1.595, and 1.364 for LS, LAD, M-, and nonparametric regression, respectively.

Example 2. Let us apply all six regression methods to turnip green data. Table 1 lists the estiamtes $\hat{\beta}_0, \hat{\beta}_1, \hat{\beta}_2, \hat{\beta}_3, \hat{\beta}_4$ of the regression coefficients, the estimate $\hat{\sigma}$ of the standard deviation of the error population, and the number N_0 of standardized residuals with absolute value large than 2.5.

Table 1. Results on Turnip Green Data

	$\hat{\beta}_0$	$\hat{\beta}_1$	$\hat{\beta}_2$	$\hat{\beta}_3$	$\hat{\beta}_4$	$\hat{\sigma}$	N_0
Least square	119.6	-0.03367	5.425	-0.5026	-0.1209	6.104	0
LAD	133.8	-0.05301	6.635	-0.6974	-0.1460	4.140	4
M-regression	122.7	-0.03967	5.763	-0.5443	-0.1282	4.177	4
Nonparametric	123.7	-0.04478	6.043	-0.5583	-0.1339	4.509	3
Ridge	115.9	-0.02805	4.807	-0.4363	-0.1089		
RMAHL	665.6	0.77801	-5.112	-9.0240	0.0752		

For the ridge regression and RMAHL methods, only the estimates of the regression coefficients are given because these methods are intended to be used only for estimation. Here, we took these solutions as estimates after 9,000 iterations with $a = 0.01, \beta = 3.5$, and $\eta = 0.6502$ since RMAHL estimates oscilated in the range of

some values. The five estimation methods in the table produce estimated coefficients that are noticeably different although "in the same ballpark". LAD, M-, and nonparametric regression are especially suitable when there are outliers in the data. another. Ridge regression is especially suitable when there is collinearity among the explanatory variables. Because of the high correlation of 0.997 between X_2 and $X_4(=X_2^2)$, we expect ridge regression to give more accurate estimates of β_2 and β_4 than LS. Note that ridge estimates $\hat{\beta}_1, \hat{\beta}_2, \hat{\beta}_3$, and $\hat{\beta}_4$ are all closer to 0 than the corresponding LS estimates. This agrees with the description of ridge regression as a procedure that shrinks the LS estimates. RMAHL rule gives erroneous resuls. We guess it is because of the collinearity among the explantory variables. We also think this agrees with theory. So, we should be cautious in using RMAHL rule when there is collinearity among the explantory variables.

Example 3. Next we consider a data set that has appeared as example in many books and articles. The data set consist of measurements from a factory for the oxidation of ammonia to nitric acid. On 21 different days, measurements were taken of the flow(X_1), the temperature of cooling water(X_2), the concentration of acid(X_3), and the amount of ammonia that escaped before being oxidized, called stack loss(y). All six regression methods were applied using the model $y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3$. Table 2 shows the estimates $\hat{\beta}_0, \hat{\beta}_1, \hat{\beta}_2, \hat{\beta}_3$, and $\hat{\sigma}$, the number N_0 of standardized residuals with absolute value large than 2.5. There are significant differences in the estimates of β_0, β_1, β_2, and β_3 for the six methods. This is at least partly due to outliers. As in Example 2, the M- and nonparametric estimates are similar to one another. Here, $a = 0.01, \beta = 125$, and $\eta = 1.3366$ have been used to obtain RMAHL estimates.

Table 2. Results on Stack Loss Data

	$\hat{\beta}_0$	$\hat{\beta}_1$	$\hat{\beta}_2$	$\hat{\beta}_3$	$\hat{\sigma}$	N_0
Least square	-39.92	0.7156	1.295	-0.1521	3.243	0
LAD	-39.69	0.8319	0.574	-0.0609	2.171	3
M-regression	-41.17	0.8133	1.000	-0.1324	2.661	1
Nonparametric	-40.16	0.8155	0.888	-0.1202	2.920	1
Ridge	-40.62	0.6861	1.312	-0.1273		
RMAHL	-44.31	0.4091	2.515	-0.1869		

In each example above we have applied six different mothods of regression to the same set of data for the purpose of comparing the methods. If our only purpose had been to analyze the data, it would still be good practice to apply more than one (but maybe not as many as six) regression methods. If you use several methods to analyze a data set

and they all lead to similar results, you can feel confident about your conclusion. If there are serious disagreements between the results of the different methods, you should examine the data to see why.

TLS method is the one of using the so called minor component in order to fit regression model. It is well known that TLS method is very sensitive to outliers since covariance matrix is so. The linear neural unit with RMAHL can be applied to linear regression analysis. It has turned out that this method is also very sensitive to outliers and collinearity among the explanatory variables. Therefore, we need to be cautious in applying this method to regression analysis.

References

Birkes, D. and Dodge, Y. (1993), *Alternative methods of regression*, John Wiley and Sons, Inc., New York.

Seber, G. A. F. (1984), *Multivariate observations*, John Wiley and Sons, Inc., New York

Xu, L., Oja, E. and Suen, C. Y. (1992), *Modified Hebbian learning for curve and surface fitting*, Neural Networks, **5**, 441-457.

Xu, L., and Yuille, A. L. (1995), *Robust principal component analysis by self-organizing rules based on statistical physics approach*, IEEE Transactions on Neural Networks, **6**, 131-143.