TESTS OF DIFFERENT REGULARIZATION TERMS IN SMALL NETWORKS

Crespo, J.L.

Mora, E.

Applied Mathematics and Computer Science Department
University of Cantabria
Avda. Los Castros, s/n
39005 Santander
Spain

Abstract

Several regularization terms, some of them widely applied to neural networks, such as weight decay and weight elimination, and some others new, are tested when applied to networks with a small number of connections handling continuous variables. These networks are found when using additive algorithms that work by adding processors. First the different methods and their rationale is presented. Then, results are shown, first for curve fitting problems. Since the network constructive algorithm is being used for system modeling, results are also shown for a toy problem that includes recurrency buildup, in order to test the influence of the regularization terms in this process. The results show that this terms can be of help in order to detect unnecessary connections. No clear winner has been found among the presented terms in these tests.

Background

An automatic network construction method, described elsewhere [Crespo, 1992], has been developed for system modeling purposes. This method includes overfitting control as a way of limiting the network size, and builds up an hybrid network with feedback that is easy to interpret in a useful way. It works by adding processors and fully connecting each one before trying the new network. This may add unnecessary connections, so, in order to overcome this problem, regularization techniques have been tried.

The idea is to control the weights value in order to avoid considering variables that may be non relevant for a particular task, or a particular processor; this control would suppress non needed dependencies added during the network setup process.

One of the possibilities of the algorithm is adding feedback, and eventually it should select the right lagged terms to be considered in both independent variables and output. The regularization terms influence in this selection process have also been investigated.

Options

In order to test their efficiency, several methods are presented below. Each one adds a particular term to the objective function being minimized. The most usual methods and the corresponding terms are:

Weight decay [Hinton, 1986]:

$$\lambda \sum w^2$$

where w represents the weights.

Weight elimination [Weigend et al, 1991]:

$$\lambda \sum \frac{w^2}{1+w^2}$$

This terms can be seen [Weigend et al, 1991] as a certain assumption about the a priori distribution of the weights. In particular, weight decay assumes a normal distribution centered in 0 and with variance $1/2\lambda$. Weight elimination is assuming a combination of a uniform and a normal distributions, both centered in 0. Doing something similar to what Movellan did when studying the error distribution we can propose different a priori distributions, hence other terms, such as:

Absolute value:

$$\lambda \sum |w|$$

From a bayesian viewpoint the absolute value term conveys the assumption that the a priori weight distribution is a double exponential.

Even without assuming a distribution we can add just penalty terms to the error function. For instance we may be interested in a term that is quadratic for small weights and becomes linear as their value increases, that is, in-between absolute value and weight decay. This may be called *intermediate* term.

Intermediate:

$$\lambda \sum w \tanh(w)$$

We can also be interested in a term that makes a more clear distinction among small and big weights by considering that big weights add no cost but small non-zero ones do. Such a term could be obtained by truncating weight decay.

Truncated weight decay:

$$\lambda \sum w^2 \text{ if } w>\alpha \text{ and } 0 \text{ otherwise}$$

The intermediate, though it hasn't been derived from any a priori distribution function, this should be in between normal and double exponential, similar to Huber's. The truncated weight decay term would come from some kind of combination between a truncated distribution for small weights and a uniform distribution for large ones.

When gradient descent is applied to the modified cost function, a negative term appears for each weight. These are shown in fig. 1 as a function of weight value for all different methods. Truncated weight decay is not represented, since it would be exactly like weight decay for small weight values and zero above certain cut point.

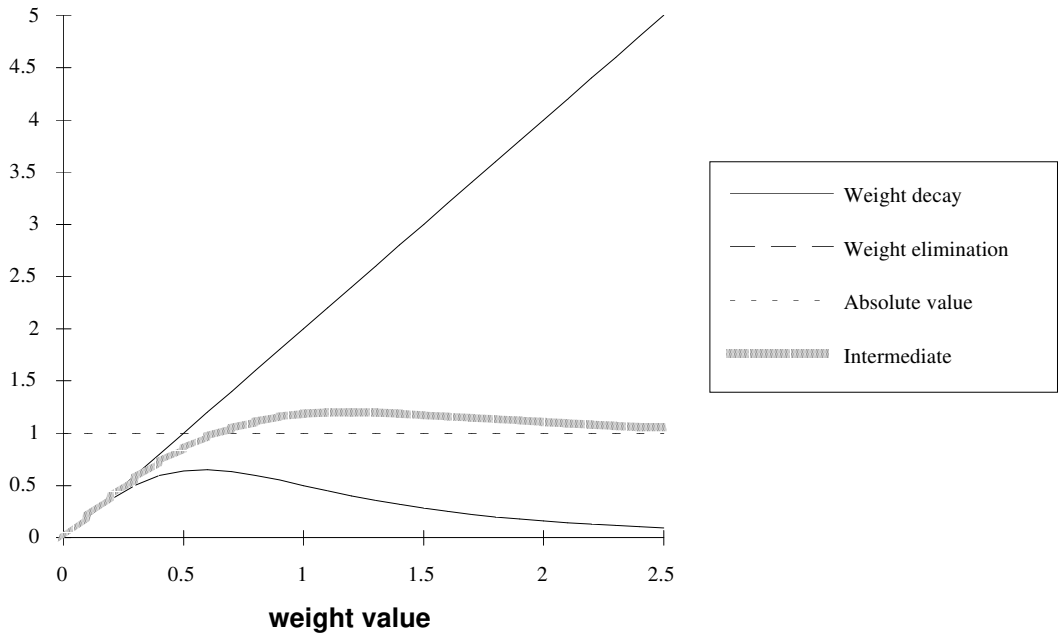

Fig. 1. Terms that affect each weight for the different methods tested here, expressed as times λ.

Test results

In order to test their effect on distinguishing non relevant variables, all methods have been tested with a combination of two straight lines with a null coefficient in each one, so the correct result would have been to assign a zero weight to the connections corresponding to those input elements representing the non relevant variables. These lines are:

$$2x \text{ for } 0 < x < 0.5$$

$$1 \text{ for } 0.5 < x < 1$$

The sample size was 600 points from a uniform distribution.

The results obtained so far are presented in tables 1 and 2. The constructive algorithm built 4 connections for the cases presented in table 1 and 6 for the ones in table 2. Ideally two of these connections should be discarded.

Table 1. Results achieved with different methods in a problem with a noise standard deviation of 0.028.

Term	λ	residual error	coefficient 1	coefficient 2
Weight decay	0.01	0.034	0.17	0.010
Weight decay truncated at 0.01	0.01	0.028	0.0027	< 0.001
Weight decay truncated at 0.1	0.01	0.028	0.010	< 0.001
Absolute value	0.01	0.029	0.027	0.0048
Weight elimination	0.01	0.029	0.033	0.0046
Intermediate	0.01	0.031	0.13	< 0.001
None	NO	0.028	0.0046	0.0015
Weight decay	0.0001	0.028	0.0025	< 0.001
Weight decay truncated at 0.01	0.0001	0.028	0.0028	< 0.001
Absolute value	0.0001	0.028	0.010	0.0012
Weight elimination	0.0001	0.028	0.020	< 0.001

Next, in order to test the effects on recurrency selection, a toy problem is setup following the equation:

$$y_n = 0'5(y_{n-1} + x_{n-2})$$

The sample includes 200 points starting x=0, y=0 and keeping x between 0 and 1. No noise is added.

Table 2. Results achieved by different methods when applied to a sample a noise standard deviation of 0.12.

Term	λ	residual error	coefficient 1	coefficient 2
Weight decay	0.01	0.12	0.055	>0.2
Weight decay truncated at 0.01	0.01	0.12	0.039	0.16
Weight decay truncated at 0.1	0.01	0.12	0.044	0.31
Absolute value	0.01	0.12	0.072	0.048
Weight elimination	0.01	0.12	0.034	0.090
Intermediate	0.01	0.12	0.053	>0.2
None	NO	0.12	0.019	0.017
Weight decay	0.0001	0.12	0.021	0.015
Weight decay truncated at 0.01	0.0001	0.12	0.039	0.16
Absolute value	0.0001	0.12	0.0089	0.0034
Weight elimination	0.0001	0.12	0.019	0.019

The results are shown in table 3 for different methods used along the whole construction process.

In order to perform a more difficult test, another equation is chosen, including a larger dead time. This is:

$$y_n = 0.5(y_{n-1} + x_{n-6})$$

The results achieved with all methods show a residual error equal to the noise standard deviation, but none is similar to the proposed equation. This is due to autorregresive relationships found along the step by step procedure.

Table 3. Results achieved with different methods, using them along the whole construction process.

Term	residual error	final models
None	0.0051	○ $0.59x_{n-2}+0.045x_n-0.12x_{n-1}+0.48y_{n-1}$ ○ $0.19x_{n-2}-0.19x_n+0.44x_{n-1}+0.55y_{n-1}$
Weight decay	0.0013	○ $0.54x_{n-2}+0.024x_n-0.060x_{n-1}+0.49y_{n-1}$
Weight decay truncated at 0.01	0.027	○ $1.19x_{n-1}-0.59x_n-0.028y_{n-1}+0.42y_{n-2}$ ○ $1.26x_{n-1}-0.63x_n-0.093y_{n-1}+0.46y_{n-2}$ ○ $1.15x_{n-1}-0.62x_n-0.002y_{n-1}+0.45y_{n-2}+0.060$
Weight elimination	0.0013	○ $0.55x_{n-2}+0.026x_n-0.063x_{n-1}+0.49y_{n-1}$
Absolute value	0.0016	○ $0.55x_{n-2}+0.026x_n-0.066x_{n-1}+0.49y_{n-1}$

Further work is necessary, including some way of avoiding autorregresive terms, what may be related to sample setup.

Conclusions

For a constructive method such as the one being used here, regularization methods do not show very glossy results. This is due to the fact that there are not many extra connections.

This methods do not totally cancel weights, but reduce the value of the non necessary ones. Certainly they are a clever choice, but user interpretation is still necessary.

Considering the results achieved so far, there is no clear winner for all cases. As it was already known the regularization coefficient should be kept very small. Truncated terms have not showed advantages in the test cases.

When it comes to large lags, none of the approaches used so far has been successful at eliminating autorregresive effects. This means that the network is not capable yet of finding the right lag by itself. The user analysis is again necessary.

References

Crespo, J.L. "Procesamiento paralelo y distribuido aplicado a la simulación de sistemas". Ph.D. Thesis, Universidad de Cantabria, Santander (1992)

Hinton, G.E. ."Learning Distributed Representations of Concepts". Proceedings of the Eighth Annual Conference of the Cognitive Science Society (Amherst 1986), 1-12. Hillsdale: Erlbaum.(1986)

Movellan, J.R. "Error functions to improve noise resistance and generalization in backpropagation networks". (1989)

Weigend, A.S.; Rumelhart, D.E. y Huberman, B.A. . "Generalization by Weight-Elimination with Application to Forecasting". Advances in Neural Information Processing 3. ed.: R.P. Lippmann, J. Moody and D.S. Touretzky. Morgan Kaufmann, San Mateo, CA.(1991)