Differential learning algorithm for Artificial Neural Networks

Manjunath R
Philips innovation Campus
Bangalore

Shyam vasudev
Philips innovation Campus
Bangalore

Narendranath Udupa
Philips innovation Campus
Bangalore

ABSTRACT

Artificial Neural Networks (ANN) are extremely useful to relate the nonlinearly depending outputs with the inputs. Various architectures are available for the ANNs to speedup the training period and reduce the square error. In this paper, new classes of neural networks with differential feedback are presented. The different orders of differential feed back form a manifold of hyperplanes. Interesting properties of this differentially fed ANN (DANN) are derived through these hyperplanes.

Categories and Subject Descriptors

I.5.1, G.0

General Terms

Algorithms

Keywords

Adaptive control, Neural networks, Differential feedback, Multiresolution.

1. INTRODUCTION

Neural networks are generally used when the input and output of a system are non-linearly related. When the data itself is noisy, the relation is described stochastically in terms of conditional probability $p(y|x)$ of the output y when x is the input. The input, output and the connecting weight matrix in a statistical neural network model are related through a family of distributions called exponential family, which has the pdf

$$P(y,\theta)=\exp\{\Sigma\theta iki(y)- \psi(\theta)\} \quad (1)$$

Where θ is the coordinate system, k= ki (y) are adequate functions of y, ψ the offset. In the context of neural networks, y is the observable state matrix ie observed p (y|x) and θ is the connecting weight matrix.[3] .For any two distributions p(y) and q(y) the geodesic connecting them [3] is given by

$$\log(p(y,t))=(1-t)\log p(y)+t \log q(y)- \psi(t) \quad (2)$$

The equation shows a linear or affine geodesic for the family of exponential distribution. It is also called flat manifold. The curvature of the geodesic connecting p1(x) and p2(x) is called Riemann-christoffel curvature for exponential family.

The distance between distributions of a manifold are parameterized as divergence. In its simplest form, it happens to be the magnitude of the geodesic. The kullback divergence of pdf p from pdf q is given by

$$D=p*\log(p/q) \quad (3)$$

The different orders of differential feedback form a manifold of hyperplanes and are related to manifolds of probability density functions(pdf). In this paper the concept of differentially fed neural networks is explored followed by the concept of hyperplanes.

2. FORMALISM OF DIFFERENTIALLY FED ANN

The output y of a neural network but for the nonlinearities can be written as

$$y=\Sigma w_i x_i. \quad (4)$$

Where xi are the inputs w_i, the corresponding weights. The space spanned by weight vector for different inputs is a hyperplane. Again the linearity of the output (4) may be viewed as a particular case of Auto Regressive Moving Average (ARMA)

$$y(n+1)=b_0 y(n)+b_1 y(n-1)+\ldots..+a_0 x(n)+\ldots \quad (5)$$

Where b_0.. and a_0.. are constants. The auto regressive terms $b_0 \ldots b_n$ may be realized using an implied differential feedback [1]. With differential feedback it has been found out[1] that the number of iterations required for training is reduced. With I order different feedback, the output reduces to

$$\Sigma w_i x_i +b_1 y_1 \quad (6)$$

y_1 being the I order differential. This equation once again represents a plane parallel to $\Sigma w_i x_i$. Thus the set of differentially fed ANNs form a manifold of parallel planes for different orders of feedback.

The two differential terms of II order differential feedback i.e., $y2-y_1$ and y_1-y_0 can be replaced by a single equivalent term y_0*W_{eq} where

$W_{eq}=(w_1*(y_1-y_0)+w_2*(y_2-y_1))/y_0$ (7)

The two parallel hyperplanes representing the two differential terms may be replaced by a single hyperplane. Extending this principle, the infinite terms of infinite order differential feedback can be replaced by a single equivalent term. The manifold of hyperplanes can be replaced by a single hyperplane. This is termed as Eigen plane.

2.1 Information geometry of differential feedback.

In order to minimize the error, the plane spanned by the weight vectors should be as close as possible to the Eigen plane. When I order differential feedback is given, the new plane is given by

$y_{new}=\sum w_i x_i+a*y_{old}$ (8)

This plane is parallel to the original plane represented by $\sum w_i x_i$ .For a given number of iterations, simulation results show that the square error is found to decrease asymptotically with the order of differential feedback. Hence, the gap between parallel planes decreases in the same way and the infinite order differential feedback plane coincides with the Eigen plane. In this case, the square error is zero.

In [1] it has been proved that the entropy is minimum on the Eigen plane. The error distribution is assumed to be Gaussian distributed since a large number of additions are involved with the higher order differential feedback.

The Natural learning algorithm is given by [2]

$\theta\ (t+1)=\theta\ (t)-\eta\ G-1$ (9)

This sounds as new plane=old plane + deviation which shows that the repeated learning in gradient descent algorithm shifts the planes towards the Eigen plane in the same way the differential feed back will do ,but fails to reach it because Eigen plane does not belong to the space spanned by inputs alone.

The different hyperplanes may be taught of as different observations of the output p(y|x) for the applied input x. Deviation of any plane from the Eigen plane is given by Kullback Divergence. The Kullback divergence of q from p

$D(p||q)=\int p(s)\log(p(s))/q(s)\ d\lambda(s)$ (10)

Here q(s) is the prob. Distribution of Eigen plane p(s) is the probability distribution of some plane so that D is the deviation from the Eigen plane. At Eigen plane p=q so D(p||q)=0 .

2.2 Spectrum smoothening

With differential feedback, the spectrum gets smoothened and becomes more and more flat. Suppose $X(e^{j\omega})$ is the spectrum of the signal. The spectrum of its derivative is given by $\omega^2\ X(e^{j\omega})$.This makes the high frequency signals of $X(e^{j\omega})$, which generally decrease with the frequency, to get lifted and the spectrum becomes more and more flat.

3. SUPERPOSITION OF HYPERPLANES

A neural network trained with Bayesian learning algorithm outputs entire distribution of probabilities over hypothesis set rather than a single hypothesis. In the present context each hypothesis corresponds to one hyper plane i.e., different orders of feedback. I.e. Each hyperplane may be taught of as a classifier with an associated probability density function. Degree of belief is 0 for no feedback and increases towards 1 for infinite feedback or when all classifiers merge. In such a classifier the actual output may be thought of as superposition of beliefs [11]. The addition is not simple but weighted by belief or pdf. Finally the superposed effect of all classifiers is the Eigen plane. This gives

P0*no feedback+p1*I order differential feedback=p2*II order differential feed back (11.a)

P1* I ordered differential+…infinite order =Eigen plane (11.b)

I.e. weighted sum of different ordered differentials.

P1*distance between I order and nofeedback+p2*II order and no feedback+…=1*distance between no feedback and Eigen plane (11.c)

The equations show that the learning algorithms with differential feedback do indeed resemble Bayesian learning algorithms and are hence resistant to over fitting [12,13]. Resistance for over training

The posterior has two components-a data independent Gaussian prior part and a data dependent term. Logically, the Gaussian part may be attributed to the previous or differential terms of the output since the weighted sum of any probability distribution function in general turns Gaussian. Such a Gaussian classifier is known to resistant to over fitting.

3.1 Bayesian learning

The aprior distribution $P(\lambda)$ generally encodes some prior knowledge. With the arrival of data pattern D the aprior distribution gets updated using Baye’s rule as $P\ (\lambda|D) \propto P\ (D|\lambda)\ P\ (\lambda)$.Taking Logarithm both sides, we get

$\mathrm{Log}\ (P\ (\lambda|D)) \propto \log\ (P\ (D|\lambda))+\log\ (P\ (\lambda))$ (12)

The equation has two terms-one current data dependent term and one data independent term, where the previous outputs are considered. The posterior distribution so obtained hence encodes information coming from the training set and prior knowledge.

Consider the example of II order feedback which makes use of two previous or priori terms P (λ1) and P (λ2). With this the equation may be rewritten as

$P(\lambda|D) = P(D|\lambda 1)* P(\lambda 1) + P(D|\lambda 2)* P(\lambda 2)$ (12.a)

which leads to the equation

p2*II order differential feed back =P0*no feedback+p1*I order differential feedback (13)

The equation tries to expand the (k+1) th order differential feedback plane with0,1..K th order differential feedback planes. The weighing factors may be taught of as the projection or dot product of the hyper plane over lower order hyper planes. The above equation may be rewritten as

$P(\lambda|D)=P(D)\{P(\lambda 1)+ P(\lambda 2)\}$ (14)

I.e., Output without feedback or the bias term*Gaussian like pdf. especially with higher Orders of the feedback

4. CONCEPT OF IDEAL ESTIMATOR

From [15] it is evident that the likelyhoods of different estimators form a manifold of probability distributions. Here, these distributions are mapped to form a manifold of hyperplanes each of which are formed by the different orders of differential feed back from the out put to the input. Thus the different estimators correspond to the different hyper planes with the ideal one being the Eigen plane.

The error at any plane i.e., corresponding to an estimator has 2 components –variance and bias related as Error2= bias2+variance2

Here the bias corresponds to its distance from the non-feedback plane (uncertainty error as the plane fixture itself is the erroneous deviation of the obtained plane from fixture of the assumed ideal plane) and variance to the approximation error (deviation of this fixture from ideal). Bias error for all degrees of differential feedback remains the same. Variance reduction increases with the order. This happens because the planes get congested as we move towards the Eigen plane as the order of differential feedback increases.

5. CONVOLUTION OF HYPERPLANES

Let y_k represent the k^{th} hyperplane corresponding to k^{th} order of differential feedback. Hence

$y_k= a*y_0+b*y_1+\dots$ (15)

In terms of the differentials of the previous output, it becomes

$y_k =b_1*y_{k-1}+c_1 d(y_{k-1})/dt$ (16)

c_1 being a constant Here the incremental portion will be approximated as the convolution of some function with y0.

Ie $\Delta y= c_1 d(y_{k-1})/dt\ y_0*f$ (17)

F may be found out by pushing both sides of the equation to frequency domain. *F*(f) should have a linear response over the points of interest .Its equivalent time domain signal may be expressed as.

$$F = x\frac{Sin^2(ax)}{ax}e^{-\frac{Nt}{2}} \qquad (18)$$

Since the response is delayed by N/2.N being the no. of points in Fourier transform. The signal is shown in fig 8. As evident from the figure, over a very small duration, the result may be approximated to a Gaussian pulse of scale factor l where the Gaussian kernel with dilation parameter l is given by G(l,x) an exponential function.

Analytically F decays as 1/t2 =(1-t2/k), as the exponential approximation of a Gaussian pulse. Hence, the envelope of f is Gaussian if the peaks are too close ie sinc function is large. The table III shows that the gap between the hyperplanes decreases with increase in the order. I.e. the information content becomes more and more abstract. Hence the kernel which is convolved with the output y is scaled with progressively increasing scale factor.

5.1 Working of the model

In the domain of learning, mixtures of Gaussians is a powerful tool for statistical modeling. Such a model can avoid the problem of over fitting [16]. The Gaussian model is regenerative with

$x=f(z)+u$ (19)

Where f(z) is the mixture of Gaussians and u is the bias. All components of x are linear combinations of Gaussian random variables. Since convolution with Gaussian function may be expressed as a linear combination of scaled and shifted Gaussians, the hyperplanes are expressible as a linear combination of Gaussian variables.

6. Hyperplane data sieves

A simple differential feedback from the input to the output gives a manifold of solutions with varying degrees of error. This spectrum of manifolds may be taught of as coarse to fine approximation of solution .i.e., as one increases the degree of feedback the fineness increases. This, by intuition, implies that the approximation set is somehow analogous to the wavelet representation of the actual solution. To explore the analogy further, consider Bohr's theorem, which relates the outputs at the levels n and n+1 as

$y(n+1)/y(n)=x(n)$ (20)

x(n) being the input. y(n+1) and y(n) also represent the (n+1)th and nth hyperplanes or degree of differential feedback. Hence x(n) may be taught of as a transform taking place at level n to level n+1

y(n+1)=x(n)*y(n) (21)

I.e. y(n) maps a certain hyperplane on to the same domain, but with a different level of abstraction. The distance between adjacent hyperplanes reduces progressively. It is this reducing distance between the hyperplanes responsible for the different abstractions of representation of the solution at different levels.

The multiresolution property of the hyperplanes has been established in[1].The simulation results actually prove that the increase in abstraction in the hyperplanes is the same as that of the wavelets.

The previous section has shown how the higher order hyperplanes may be derived from the lower order ones by convolving with a Gaussian pulse of varying degree of abstraction. Such a convolution results in multiple resolution of the resulting hyperplanes. In addition, simulation results show the orthogonality of hyperplanes. The hyperplanes thus exhibit the required characteristics of the wave let coefficients

7. SIMULATION AND RESULTS

First, the performance of DANN was tested on XOR logic function. This is followed by the different properties of the DANN imparted with the hyperplane representation. In all the cases, the output is clipped to saturation when it crosses 1. Error threshold of .1 is used as stopping criteria

7.1 XOR function learning and equivalency of hyperplanes

Gaussian distributed random signal with seed value 1000 is taken as input. The output is generated by performing XOR over the consecutive values and saturating the result between 0 and 1.. The table I shows the iterations performed and the error. The table II shows equivalency of hyperplanes. With differential feedback it has been found out [1] that the no of iterations required for training is reduced as shown in the table 1. XOR gate is considered for simulation. The equivalency of hyperplanes is given in Table 2.

Table 1 Performance of DANN

Order of differential	Square error	Iterations
No feedback	18	1156
I order	18	578
II order	18	289

Table 2 Equivalence of hyperplanes

Order of differential	Square error	Iterations
II order Feedback	18	578
Equivalent Output feed back	18	578

7.2 Spectral smoothing

The error signal generated in XOR learning is used for the simulation. The spectrum of error signal and its different derivatives are plotted in figure 1.

Here the DANN learns the spectrum of the error signal generated. It may be seen in the figure that the spectrum becomes more and more flat when higher order differential feedback is given. Also, magnitude of the error signal reduces and the error energy decreases with increase in the order of the differential.

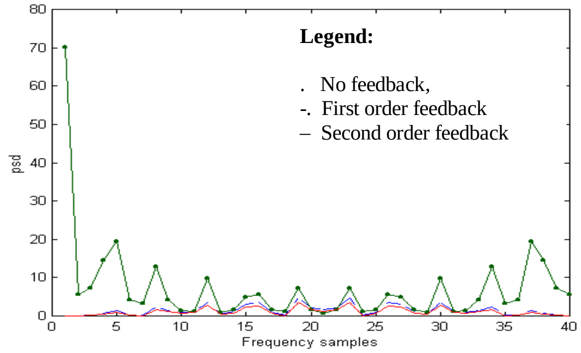

Figure 1. Spectrum of the error in the output and its derivatives

7.3 Bayesian learning and ideal estimation

The differentially fed artificial neural networks are made to learn the power spectral density (psd) of the Normal distributed data in the interval (0, 1). The network is trained up to second order feedback The error after learning and the differentials of the error are stored The probability distribution of each of them is computed using Parzen equation

$$p(x) = \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{(mean-x)^2}{\sigma^2}}$$

(22)

σ being the standard deviation of the distribution. The weighted sum of the zero order feedback and I order feedback data with their corresponding probability distribution functions (pdfs) is found identical to the weighted second ordered differential feedback with the corresponding pdf as given in the equation In figure 2 signals of first and zero order weighed with pdf and weighed second order signal are shown. From the table 3 it is clear that With I order feedback bias or the mean value remains the same. Variance is reduced. They are further reduced with II order differential feedback. I.e. the output plane has been lifted towards the Eigen plane. The mean and variance of the error are shown in table 3.

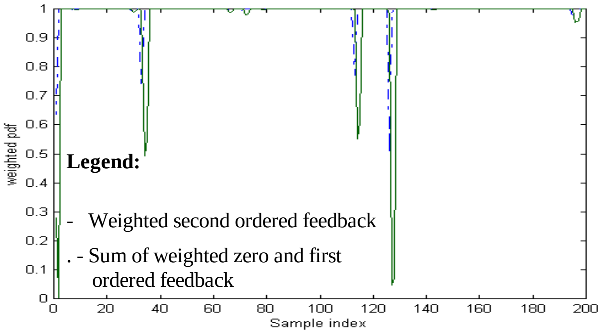

Figure 2. signals of I and zero order weighed with pdf and weighed II order signal

Table 3 Mean and Variance of the error

Order of differential	Variance	Mean
No feedback	6.9132e-005	0.998
I order	6.8614e-005	0.998
II order	6.3371e-005	0.998

7.4 Multiresolution of hyperplanes

The same data of previous experiment has been made use here. The normalized PSDs of convolution of outputs without feedback and Gaussian pulse and normalized power spectral density (PSD) of the first derivatives with feedback are shown in the figure 3. It may be seen from the figure that the derivatives are formed by the convolution of the output with Gaussian functions of different scales

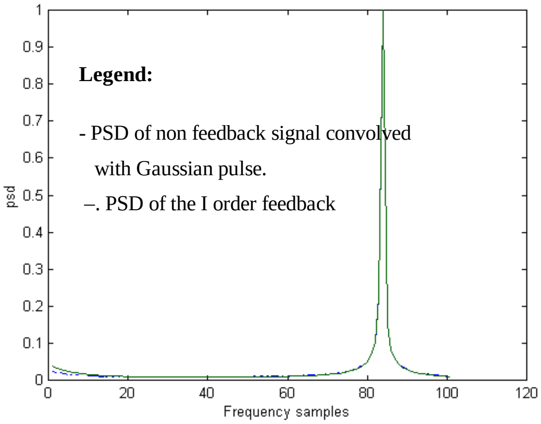

Figure 3. PSDs of Differential signals and the convolved signals

It has been shown in table 4 that outputs at different hyperplanes are orthogonal. The differentially fed Artificial neural networks are made to learn the PSD of random data .The Normal distributed data is generated using Matlab. The error after learning and the differentials of the error are computed. This satisfies the condition that the basis functions have to be orthogonal.

Table 4 Orthogonality property of hyperplanes

Order	Order	Sum of product
0	1	0.0082798
1	2	0.0008682
2	0	0.081904

.

8. CONCLUSIONS

With differential feedback from the output to input of an ANN, it has been found out that the number of iterations required for training is reduced. The set of differentially fed outputs form a manifold of parallel planes, with infinite order feedback being the plane with zero error. Simulation results show that error varies asymptotically with order and the gap between parallel planes decreases with order. The entropy is the minimum when infinite feedback is given.

Differential feedback, when applied over a neural networks leads to a manifold of affinely transported hyperplanes. These hyper planes are actually formed by the convolution of the non feedback output with Gaussian kernels of different scales.

From the simulation results it is clear that the classifier represented by a certain hyper pane is the weighted sum of the hyper planes or classifiers below. This way, ideal classifier is the weighted sum of all the classifier. The differential feed back represents a signal with the same level of abstraction as that of wavelets. It provides a crucial link between wavelets and information geometry.

The performance of an artificial neural network as an estimator has been examined. It has been found out that the estimator performs better if higher order differential feedback is given, reducing the errors due to variance of the measurement.

9. ACKNOWLEDGMENTS

Our thanks to Philips Innovation Campus, Bangalore for the support rendered to publish this work..

10. REFERENCES

[1] Manjunath.R and K.S.Gurumurthy,2002. System design using differentially fed Artificial Neural networks, TENCON’02.

[2] Aarts, E.H.L. and Korst, J.H.M., 1989. Simulated Annealing and Boltzmann Machines, Chichester: Wiley.

[3] S.Amari , 1995. Information Geometry of the EM and em algorithms for neural networks, Neural networks, 8,No.9.

[4] Amari.s., 1982. Differential geometry of curved exponential families –curvatures and Information loss. Annals of statistics.

[5] Amari.s. Information Geometry of neural networks-New Bayesian Duality theory—

[6] S.Amari,1998. Natural gradient works efficiently in learning, neural computation 10, pp..251-276

[7] S.Amari and H.Nagaoka,2000. Methods of information geometry,AMS and Oxford university press.

[8] Amari,1995. Information geometry of EM and em algorithms for neural networks, Neural networks 8(9),pp. 1379-1408

[9] Amari.s, 1990.mathematical foundations of neurocomputing, Proceedings of IEEE, 78, pp 1446-1463

[10] D.J.C.Mac Kay, 1992.A practical Bayesian framework for back propagation networks Neural computation 4,pp. 448-472

[11] Radford Neal, 1996. Bayesian learning in neural networks springer verlag.

[12] D.J.C. MacKay. 1992.The evidence framework applied to classification networks.Neural computation, 4, pp720-736

[13] D.J.C.MacKay,1992. A practical Bayesian framework for backpropagation networks.Neural computation, 4, pp.448-472

[14] Zhu, H. and Rohwer, R. 1995. Bayesian invariant measurements of generalisation for continuous distributions. Technical Report NCRG/4352,Aston University. ftp://cs.aston.ac.uk/neural/zhuh/ continuous .ps.Z.

[15] Rao, C. R. 1962. Efficient estimates and optimum inference procedures in large samples J. R. Statist. Soc., B, 24, pp.46–72

[16] Peter Meinicke and Helge Ritter.1999. ,Resolution based complexity control for Gaussian mixture models. Technical report, Faculty of Technology, University of Bielefeld.

[17] McKenzie, P. and Alder. M. , 1994. Initializing the EM algorithm for use in Gaussian mixture modelling. In Gelsema, E. S. and Kanal, L. N., editors, Pattern Recognition in Practice IV, Amsterdam: Elsevier., pp.91-105

[18] M.Frazier and B.Jawerth. 1985.Decomposition of Besov spaces Indiana univ. of Maths journal 34(4),pp.777-779

[19] S.Mallat and W.L.Hwang, 1991. Singularity detection and processing with wavelets. Preprint Courant Institute of Mathematical sciences, New York University.

[20] B. Jawerth and G. peters,1993. Wavelets on non smooth sets of Rn. Reprint.