Tina Memo No. 1997-006
Internal Memo

Autonomous Training Algorithms for Neural Networks.

N.A.Thacker, P.I.Rockett and S.W.Beet.

Last updated
5 / 5 / 2017

WWW.TINA-VISION.NET

Imaging Science and Biomedical Engineering Division,
Medical School, University of Manchester,
Stopford Building, Oxford Road,
Manchester, M13 9PT.

Foreword

The research into artificial neural networks (ANNs) received a huge boost in effort towards the end of the 1980's, following publication of Rumelhardt and McClelland's PDP books. At a time when the use of back-propagation was highly popular at computer vision and pattern recognition conferences, a core of researchers in the UK levelled a set of criticisms of this kind of work, generally focused around statistical validity, flexibility and fundamental academic value of using black-box systems to solve recognition problems.

The EPSRC initiative, Neural Networks; The Key Questions, was a program set up to address the difficulties and challenges faced by those researchers trying to use ANNs for AI tasks. I have failed to find a copy of the original call for proposals, but I seem to remember that David Bounds really nailed it. The key questions were;

- *What is the optimum strategy when applying neural computing to a given set of static data?*
- *What is the optimum strategy when applying neural computing to time-varying data?*
- *What is the best representation of data for neural computing?*
- *How can neural computing be used for data fusion?*
- *What does neural computing have to offer for optimisation and constraint satisfaction tasks?*

From memory, answering these questions involved addressing issues such as; What is the fundamental statistical basis for training? Can an ANN identify an unfamiliar pattern? Do ANNs actually embody useful (meaningful) decision systems? Can training be more flexible than batch supervision? Is it possible to directly train to optimise generalisation? Most of which still seem pertinent to more recent "deep learning" work. This has motivated me to make this document available now.

At the time I had some ideas and submitted a grant to this programme, what follows is all I have left from an early draft, but the main idea is there. Many of the individual paragraphs resurfaced in subsequent publications.

Although the project was funded the subsequent work was not a great success. In part this was due to a career move (to become a lecturer in Manchester), bad luck (a letter to the EPSRC was lost for several years down the back of a desk) and poor communication. In the years that followed however, related pieces of work were finally completed. It is useful to see how closely some of this work matched the original ideas, so I've added a short section at the end of this document, listing the related work.

Neil Thacker 2017

Autonomous Training Algorithms for Neural Networks.

1 Main Objectives.

The aim of this project would be to deliver autonomous neural network training algorithms which will provide principled solutions to problems of static pattern classification. This will include a theoretical statistical evaluation of the Bhattacharryya similarity metric for the automatic selection of computational models of data and the development of two complementary neural network architectures. One which aims to grow a neural architecture to solve a problem and the other which aims to reduce the internal complexity of the network in order to improve generalisation ability. This project will provide a practical use of these algorithms for immediate industrial exploitation.

2 Proposed Program.

The main area of research for this program comes from the development of the CLAM neural network architecture [9]. Based on the work of Grossberg, this architecture was specifically designed to address the problems of autonomous generation now identified by this initiative as of fundamental importance to the field of neural networks. In the last few years continued work on this architecture has led to a much better theoretical understanding of the CLAM algorithm. In particular those parts which allow the network to automatically generate an architecture which can be shown to be guaranteed to solve the mapping defined by any classification problem [11]. This is achieved (primarily) by use of the Bhattacharryya similarity function [3] to compare frequency coded distributions. This statistic is used to make decisions regarding modifications of the neural network architecture. Most of the remaining CLAM architecture and structure specification were required in order to make these modifications possible. This has led in turn to fundamental developments in the field of computer vision for solutions to some classes of object recognition and system identification [5, 10].

We now believe that the research has come full circle, complementary evidence on the suitability of a measure of this form for estimating the generalisation ability of a fixed network architecture has been provided independently by MacKay [8]. We believe that it would be possible to develop a training algorithm based on the Bhattacharrya statistic which would directly train an MLP based architecture to optimise its prediction ability.

3 System Identification

The problem we are addressing is one of system identification, that given a set of data we can find the most appropriate parametric description of the system which generated it. In the simplest case we can consider a set of 2D vectors which we wish to describe by a function. The standard method of least-squares fitting will give us an estimate of the optimal set of parameters to describe a given data set with a particular model but unfortunately the Chi-squared measure is not directly suitable for model selection. Standard neural network training algorithms are equivalent to high dimensional least-squares fitting for function interpolation. The inability to select the most appropriate model to describe the data has been described in the literature as the "bias/variance dilemma" [6].

The standard method for optimal model selection is that suggested by Akaike. He showed that the Chi-squared test statistic is biased towards small values due to the freedom that a model has to match the variation in the noise. An analysis for large data samples [1] shows that the bias could be estimated and compensated for using the test statistic:

$$\chi^2_C \; = \; \chi^2 \; + \; N/m$$

Where N is the quantity of data and m is the number of degrees of freedom for the parametric model. Under some limited circumstances this measure is sufficient to enable model selection but the method does have its limitations.

The limitations are directly related to the definitions of the N and m terms and can best be understood by taking some extreme cases. For example, take any model with m parameters M where m is the number that we believe we should be using in the Akaike measure. Now redefine the model to create a new parameter by splitting one of the parameters to form two new ones L_1 and L_2 such that $L_1 \; + \; L_2 \; = \; M_i$. The new number of model parameters is now $m+1$, what is to stop us doing this? This seems like a strange thing to do but there are other (less obvious) ways of introducing correlations between parameters in the definition of a model which are more difficult to spot (a 3x3 rotation matrix for example has 9 free parameters but only 3 degrees of freedom). This line of reasoning leads to the conclusion that the number of model parameters is not necessarily the number we are currently using to define the model but the number of linearly independent model parameters. However, if this is true (and it is generally accepted that it is) we now have a problem because the definition of linear independence is data dependent so we would need a different value of m for different data sets as well as for different model parameters.

The problems are not limited to the model. Consider two data fits, both from N data but one with the data well distributed throughout the measurement domain and the other set tightly clustered. A well distributed data set can strongly constrain a set of model parameters but a tightly grouped set of data may not. Again the bias correction term is data dependent in a way that is not taken into account.

Both of the above problems (and others we do not have space to mention here) have arisen because the bias correction term is only the limit of the bias and does not take account of data dependent variations, particularly for small data sets. This is unfortunate because it is generally under these circumstances that we need to use the measure, particularly in the field of neural networks.

4 Suitability of the Bhatacharrya Measure.

The Bhatacharrya measure was originally defined on the basis of a geometric argument for the comparison of two probability distributions [2]. Later it was found to provide an upper bound on the Bayes classification error rate for a two class problem. In the meantime it (and the analogous Matusita measure) has been used as an empirical favourite for probability comparison in the field of statistical pattern recognition. We believe that we are now in a position to show that the Bhatacharrya measure is the correct (maximum likelihood) similarity metric for probability distributions. The measure also can be shown to be self consistent and unbiased [12]. In addition the Bhatacharrya (or Matusita) measure can be considered as a chi-squared test statistic for a model with an infinite number of degrees of freedom. Thus in relation to the work of Akaike the measure requires no bias correction ($N/m = 0$).

5 Self Generating Networks.

The CLAM neural network [9] has been designed to provide a flexible sel f generating memory module supporting the automatic addition of nodes and connections during training. The primary aim of the original research was to develop a layered architecture capable of classifying sets of vector patterns suitable for use in object recognition. This work was later extended to provide a full probabilistic framework for the use of the network in pattern classification. The improved utility over standard networks is obtained using statistical constraints that have to be met on the form of the input data.

The probability of a node being chosen as the best representation of the input on the basis of the Bhatacharyya product measure is at the heart of the node generation algorithm which attempts to generate nodes for each distinguishable pattern presented during training. For this reason the input representations used must be statistically appropriate in order for the generation of the network architecture to proceed correctly. In particular some forms of data, such as frequency histograms and Fourier co-efficients have the correct properties. Any variable we may wish to use for input to CLAM can be re-represented into one of these forms.

The node selection process defined by the dot product is used to compute the probability $P(j|I)$ that each template stored in the representation layer is consistent with the input data. These output probabilities can be buffered by addition over a whole set of input patterns to produce outputs suitable for input to a second representation layer operating using the same statistical mechanisms. It is this process that makes multi-layering and hierarchical classification possible.

The final output classifications of the network $P(C|I)$ (the probability of the class type being C given the input data I) are computed using the probabilities that the input pattern is consistent with each of the stored templates present in the representational layer $P(j|I)$. New classification nodes can be added without affecting the existing network structure. Thus if the errors on the input pattern are properly represented the network will compute an estimate of the required probability for any defined classification problem. Other neural network architectures can be shown to approximate the same output response when trained with the appropriate algorithms [7] but cannot guarantee adequacy of the architecture to solve the problem. This probability is by definition the most useful form of the output response of any classification system, combining robustness with maximum information preservation. Thus not only is the network completely self generating but it can also be considered as optimal, the only drawback being that the method is not necessarily computationally efficient. This is our main motivation for reconsidering more popular network architectures, (in the light of what we now understand about the Bhattacharyya measure), which have the potential for being more compact.

These algorithms are now sufficiently mature that it is timely to consider moving them into a product for commercial applications.

6 Application to Network Training.

The relevance of the Bhattacharyya measure to system identification is due to the fact that any measured data can be represented as an estimate of a probability density distribution in the measurement domain. In order to construct the measure from a set of neural network training data we must have the expected accuracy of the output data. Then we can execute the following steps;

a) compute the co-variance of the network parameters.

b) estimate the constraint provided by the network on the training data by error propagation.

c) construct a probability distribution for the prediction of each output from the network.

d) compared with the initial data set using the Bhattacharrya measure.

Such a measure estimates the prediction accuracy of a neural network and provides an indication of the generalisation ability. Empirical proof of this has already been provided by MacKay [8]. It is this work which also shows that the problems with the conventional approach, described above, regarding the effects of numbers of degrees of freedom in the model and distribution of data are sensibly treated. Although we do not agree in detail with his derivation for the measure empirical evidence is provided that the method is a direct way of assessing the generalisation capabilities of an individual network, not just a particular architecture as with conventional cross validation techniques.

However, the benefits of the measure do not have to stop here. If we believe that the measure is a suitable indicator of prediction ability we would also be justified in constructing a training algorithm based on this measure. Neural networks are complicated non-linear functional interpolators, the weights define not only the non-linear mapping but also the degree to which the internal parameters are correlated and therefore the number of effective degrees of freedom for the mapping parameterisation. Parameters can effectively be removed from the functional mapping by, for example, driving bias weights to large values to isolate some of the nodes in a network. If the Bhattacharrya measure is an unbiased estimate of network prediction ability it should be possible to train a network to simultaneously minimise the output mapping accuracy and minimise the required internal degrees of freedom. All without the need for ad-hoc weighting of the significance of each of these to the combined cost function.

At this stage it would be right to ask the question; What extra information is being used in comparison to standard training algorithms that makes all of this possible? After-all, the bias/variance dilemma is real so we must be providing new information if we are to solve it. The answer lies in the estimate of the errors on the output data. With normal least-squared no assumption is made about the absolute error on the output data. All error scales will give the same least squares solution. With the Bhattacharyya measure the output is sensitive not only to the mean of the output distribution but also its variance. The more accurate the data the more accurately the function will be required to map it. This is an entirely reasonable requirement which we would expect to have to take into account if attempting to interpolate a data set by eye.

7 Project Outline.

The research for this project would involve;

a) Finalising a theoretical derivation of the statistical origins of the Bhattacharrya measure.

b) Developing a commercially use-able version of the CLAM algorithms.

c) Application of CLAM to multi-spectral image classification.

d) Developing a training algorithm for an MLP type architecture which will optimise its generalisation ability.

e) Evaluation of the MLP training algorithms with multi-spectral data.

f) Possible extensions to recurrent network architectures for temporal sequence classification.

8 Relevance to EPSRC.

This research program is to be based directly on a theoretical examination of the foundations of the Bhattacharrya overlap integral as a probabilistic similarity measure (note; not the upper limit on the Bayes error as is commonly quoted). As such it is rooted in a strong statistical framework. Both network architecture approaches will deliver techniques which implicitly answer questions regarding: the best representation for the input data, sensitivity and stability analysis and estimates of output performance and reliability. These properties have already been demonstrated for the CLAM network architecture and will be calculable from evaluation of the co-variance on the weights in the MLP architecture which is an integral part of the construction of the Bhattacharyya metric.

The proposed research will directly impact on the automatic application of statistical neural network pattern classification techniques in both research and industrial domains.

9 Background.

The Silicon Vision Group at the Department of Electronic and Electrical Engineering has considerable expertise in the area of neural networks for speech recognition and computer vision [4]. The principle applicants on this project already hold related grants: Peter Rockett has recently been awarded funding for CERVIP an EC funded project to develop a commercial computer vision system which includes neural network components, Steve Beet is a co-holder of the EC (TIDE) funded VAESS project to investigate speech synthesis and recognition. Both of these projects can be expected to provide synergistic research input to this project.

10 Industrial Collaboration

The National Remote Sensing Centre have a long term interest in the analysis of multispectral satellite images. They are currently designing a product called the "Land Information System". This will be a general purpose software product for analysis of images to generate statistics on land use-age. As the statistical properties of multi-spectral images varies it is important to provide trainable statistical classification systems which can be adapted to a wide range of tasks. The NRSC (via our contact Dr. Gavin Brelstaff) has identified neural networks a key analysis method for their system. Unfortunately the well known problems associated with generating well behaved neural classifiers is seen as a potential barrier to an effective product. In addition DRA (via David Whitaker) have expressed a strong interest in this project. The research outlined above would be of direct value to both DRA and NRSC who would both like to be involved in a collaborative venture and are willing to make a substantial commitment in terms of both time and money to this project. We estimate that the research program will require funding from the EPSRC corresponding to one RA fully funded for 3 years and a SUN workstation. Totalling approximately 100 thousand pounds.

Long Term Outputs

Our main collaborative contact at NRSC left to live in Sardiania within a month of the project starting. So we didn't get to see any multi-spectral data.

The assertion that the Bhattacharrya measure did not require bias correction when used to compare probability densities was documented in [12], but not published, as it was consided too trivial.

The derivation of the Bhattacharyya measure as a way of comparing distributions was published in [13]. The work has a rather obvious extension for application to a probability density which was not included in the paper because of the contention it seemed to arouse in reviewers.

The construction of a similarity measure which optimised generalisation was published in [14]. The idea was only tried out for polynomials, ideally we would have taken this forward into an ANN.

Our attempt to use these insights during network training was submitted to ICANN 1999, but rejected [16]. By then the manpower for the project had evaporated and we all had to move on to other things.

The supervised extensions to the CLAM network were finally published in 1997 [11].

The relationship of the square-root homoscedastic transform to more conventional statistical similarity scores was explored in [17]. This was a good piece of work, but by the end we felt that the statistics community must have known this for 50 years. So, beyond the document on our own web pages, no attempt was made to publish.

The temporal recognition work ultimately resulted in [15], a nice (well cited) paper, but not what I originally expected.

References

[1] Akaike H, A New Look at the Statistical Model Identification. IEEE Trans. Automatic Control 1974, 19(6):716–723.

[2] A. BHATTACHARYYA, On a Measure of Divergence Between Two Statistical Populations Defined by their Probability Distributions, Bull. Calcutta Math. Soc., 35, pp. 99-110, 1943.

[3] A.C.Evans, N.A.Thacker and J.E.W.Mayhew, 'The Use of Geometric Histograms for Model Based Object Recognition.' Proc. 4th. BMVC, Guildford, p 429-438 Sept. 1993.

[4] S.Evans, R.B.Yates, and N.A.Thacker. 'The application of Neural Networks to Object Recognition.',I.E.E. Col. Neural Networks for Image Processing Applications, October 1992.

[5] A.J.Lacey, N.A.Thacker and N.L.Seed, 'Feature Tracking and Motion Classification Using a Switchable Model Kalman Filter.' Proc. BMVC, York, Sept. 1994.

[6] S.Geman, E.Bienenstock, R.Doursat. Neural Networks and the Bias Variance Dilemma, Neural Computation, Vol4. No1, pp 1-58. 1992.

[7] Richard, M.D., & Lippmann, R.P. (1991). Neural Network Classifiers Estimate Bayesian a posterioi Probabilities. *Neural Computation*, **3**, 461-483.

[8] D.J.C.MacKay, 'Bayesian Modelling and Neural Networks', Research Fellow Dissertation, Trinity College, Cambridge (1991).

[9] N.A.Thacker and J.E.W.Mayhew, 'Designing a Network for Context Sensitive Pattern Classification.' Neural Networks 3,3, 291-299, 1990.

[10] N.A.Thacker, P.A.Riocreux, and R.B.Yates, 'Assessing the Completeness Properties of Pairwise Geometric Histograms", Accepted for Publication in Image and Vision Computing 1994.

[11] N.A.Thacker, I.A.Abraham and P.Courtney, 'Supervised Learning Extensions to the CLAM Network.' originally submitted to Neural Networks Journal 1994, published 10, 2, pp.315-326, 1997.

[12] N.A.Thacker, The Bhattacharyya Measure Requires No Bias Correction., Tina-memo 1999-001, 1999.

[13] N.A.Thacker, F.Ahearne and P.I.Rockett, 'The Bhattacharryya Metric as an Absolute Similarity Measure for Frequency Coded Data.' Kybernetika, 34, 4, 363-368, 1997.

[14] N.A.Thacker, D.Prendergast and P.I.Rockett,'B-Fitting: A Statistical Estimation Technique with Automatic Parameter Selection.',Proc, BMVC, 283-292, Edinburgh, 1996.

[15] J.Luettin and N.A.Thacker. "Speechreading Using Probabilistic Models". Computer Vision and Image Understanding, 65(2),163-178, 1997.

[16] P. Rockett and N. A. Thacker. The Application of the Bhattacharra Measure as a Selector of the Point of Optimal Generalisation During Neural Network Training. Tina-memo 1999-006, 1999.

[17] N.A. Thacker and P.A. Bromiley, The Effects of a Square Root Transform on a Poisson Distributed Quantity. Tina-memo 2001-001, 2001.