

An Algorithm for Active Data Collection for Learning — Feasibility Study with Neural Networks.

Tirthankar RayChaudhuri
Leonard G.C. Hamey

tirthank@mpce.mq.edu.au

This publication is available at the following URL
<ftp://ftp.mpce.mq.edu.au/pub/comp/techreports>

Macquarie University Technical Report No. 95-173C
Department of Computing
School of MPCE, Macquarie University, New South Wales, Australia

Copyright© 1995 Tirthankar RayChaudhuri and Leonard G.C. Hamey
All rights reserved

May 1995

Abstract

Statisticians have considered query-based or ‘active’ sampling of data as a means of reducing the expense of data measurement and collection in modelling tasks. The quest for more reliable neural network learning techniques has led researchers to examine statistical active querying as a means of obtaining training data that will produce greatly improved generalisation. Some algorithms evolved for active learning in neural networks perform active data subset selection. We propose using the ‘query-by-committee’ approach. This leads to an active scheme for data collection where data gathering is reduced to a minimum and yet the accuracy of modelling remains high. Our method is built around the philosophy that ‘data gathering is expensive and computation is cheap’. Our active querying criterion is determined by whether or not several models agree when they are fitted to random subsamples of a small amount of collected data. Recent experimental investigations have established the effectiveness of this algorithm for both clean and noisy data so far as neural network learning is concerned.

1 Introduction

“Data! data! data!”, he cried impatiently. “I can’t make bricks without clay.”

Sir Arthur Conan Doyle, *Sherlock Holmes, The Adventure of the Copper Beeches*

We live in a world where the growing trend is to make the most of available information. This applies to machine learning algorithms as well. The challenge is to obtain good generalisation from a limited amount of data. The traditional approach has been to study generalisation from random examples. However it has been found that random examples contain progressively less new information as learning proceeds [10, 13]. In order to improve generalisation therefore, it is necessary to make learning query-based, i.e., to set up a criterion which will select only those training examples which contain maximal information about the system being learned. Such query-based learning is often referred to as ‘active learning’ — the learner having the ability to select its own training data [3].

Apart from obtaining improved generalisation there is another major factor which motivates research in active learning methods — the high expense of data collection and measurement. Ideally an active learning method should have twin goals: to minimise both the generalisation error as well as the amount of data sampled — an apparently contradictory pair of aims. We propose a means of achieving both objectives simultaneously in this work. We have based our ideas upon concepts already introduced by researchers such as Cohn *et al* at MIT [2, 3] and Krogh and Vedelsby [7].

2 Existing Methods and a Data Minimising Approach

Recent proposed schemes of active learning in the neural network literature have covered both active data subset selection as well as active selection of unlabeled data.

2.1 Active Data Subset Selection

Tamburini and Davoli [14] have suggested that if those training patterns which exhibit the highest LMS error upon an initial classification attempt be added to the training data, then a better training set is obtained. Plutowski and White [9, 10] have implemented a technique of selecting ‘training exemplars’ which is based upon a derivation of the integrated MSE criterion — to obtain points with maximum information from already-labeled data. Their method

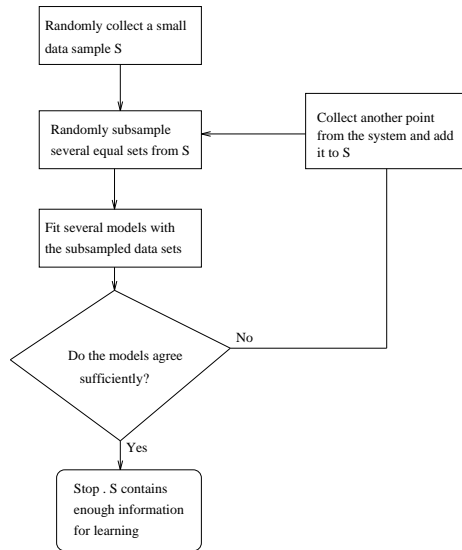


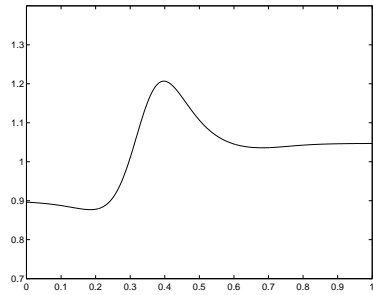
Figure 1: Active Learning Algorithm to Minimise Data Collection

is independent of the use of a neural network estimator and can generally be applied to nonlinear regression.

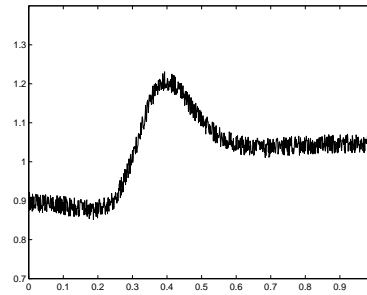
Generalisation is greatly enhanced by active data subset selection, i.e., using querying methods to resample those points which contain the most information about the system. Compared to random repetitive sampling this approach greatly reduces computational activity [13], but does little to minimise data gathering expenditure.

2.2 An Algorithm to minimise Data Collection

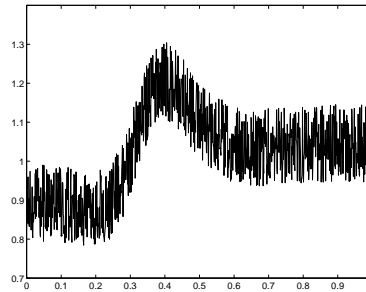
It is desirable to ‘actively’ reduce the considerable expense of data collection and at the same time not increase our generalisation error, i.e., retain modelling accuracy. Cohn *et al* [2, 3] have carried out active learning by choosing unlabeled inputs that minimise the expected value of the learner’s mean squared error. We use the ‘query-by-committee’ approach. Krogh and Vedelsby [7] have successfully applied this idea to neural network learning; however their emphasis has been upon reducing the generalisation error of a neural network ensemble rather than minimising data collection. They began training with one ‘example’ and added one point at a time corresponding to maximum ‘ensemble ambiguity’. Each of their networks were trained upon all the training data used. We introduce a querying criterion which is determined by whether or not several



(a) Nonlinear Plant

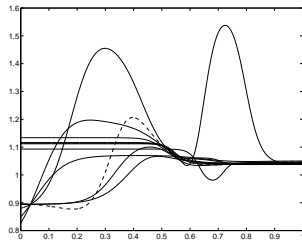


(b) Nonlinear Plant with Noise Added
Noise Uniform range ± 0.025

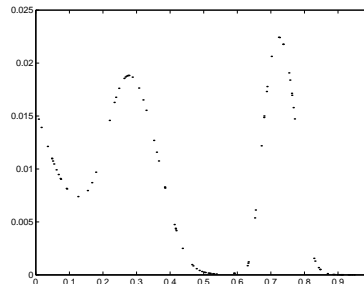


(c) Nonlinear Plant with High Noise Content
Noise Uniform range ± 0.1

Figure 2: Nonlinear Plant with Varying Noise Levels



(a) Initial Models Obtained by Training Ten Neural Nets on Random Subsamples of Collected Data.



(b) Variance Plot of the Initial Disagreeing Models at 100 Randomly-Selected Points.

Figure 3: Initial Models and their Variance. The Plant in (a) is shown in broken line

models agree when they are fitted to *random subsamples* of a small amount of data. If we sample a small number of random data points initially and fit several models with random subsamples of this data then we can compare the different outcomes of the models. If the models agree closely then the initial data samples are sufficient for the learning task. If the models disagree then we collect minimal additional data from the original system and repeat the modelling activity and re-compare the outcomes. *Thus model disparity influences the collection of further data.* This process is iterated until the models agree sufficiently (see Figure 1). Data collection and its expense are thereby minimised without having to compromise modelling accuracy.

3 Experiments with Neural Networks

In order to study the feasibility of the algorithm several experiments were performed using neural network models. Ten different networks were used. Training data was obtained by subsampling randomly from an initial small random data sample collected from a nonlinear system (Figure 2(a)) which we will refer to hereafter as the ‘plant’. This plant was artificially generated with a feedforward neural network with one input node, three hidden nodes and one output node. The architectures of the ten identifying networks were identical with that used to generate the plant, but their weights were initialised with random values every time. Using the Matlab Neural Network Toolbox, fast backpropagation with the Levenberg-Marquardt algorithm [4] performed quick training. The initial data consisted of ten points (input-output pairs) collected randomly from the plant. A subsample (selected randomly) consisting of half the number of

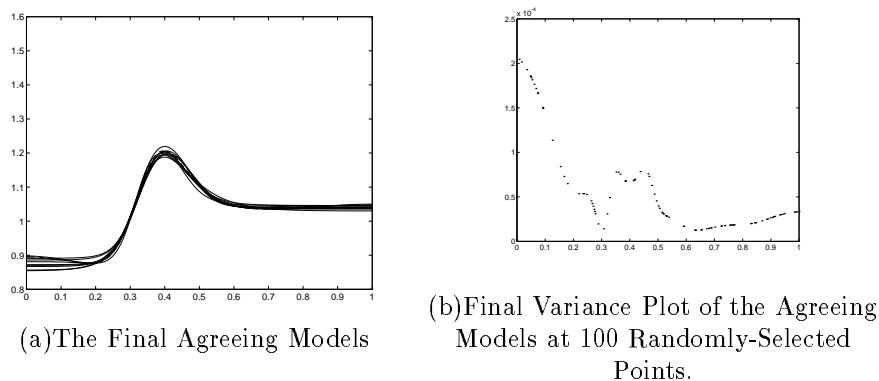


Figure 4: Final Models and their Variance.

sampled data points was used to train one of the networks. Another nine such random subsamples of equal size were subsequently used to train the other nine networks respectively. The outcome of the ten models is shown in Figure 3(a). Clearly there was disagreement at the initial stage.

The variance of the models was then computed at 100 randomly chosen points and a plot of the variance is shown in Figure 3(b). The points of higher variance are the points where more information is required — the *querying criterion* for resampling.

The point corresponding to the maximum variance value was then resampled from the plant and added to the initial data sample. Again ten sets of random subsamples of half (rounded to the floor integral value) the data were used to train ten neural networks. The models were re-compared by computing the variance as before. As long as the maximum variance remained above a threshold value of 0.001 the process of resampling the additional point of maximum variance from the plant was repeated, added to the earlier sampled data and the nets were retrained with half the updated sample each time. When the maximum variance dropped below the set threshold the iterations were ceased. The final variance plot is shown in Figure 4(b). Figure 4(a) indicates how closely the models agreed with one another. Figure 5 shows the final sampled data points and their distribution.

3.1 Random Selection of Additional Points: Passive Learning

We ran the experiments again and on this occasion, instead of adding the point corresponding to the maximum variance value, we added a point selected randomly from the plant. The algorithm converged, but as is only to be expected,

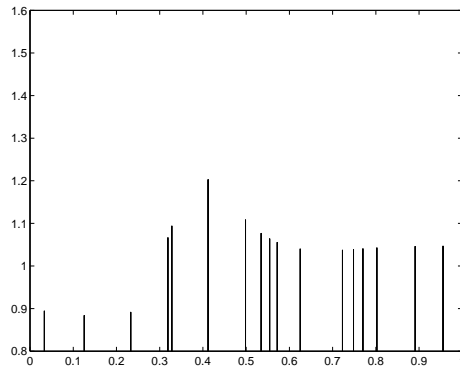


Figure 5: Data Points for Correct Modelling found by Active Experimentation

more points were collected than in the experiment with active learning. Figure 6(a) shows the final sampled points and Figure 6(b) the corresponding plot of final variance. This is a case of ‘passive’ learning [7] — the selection of the additional labeled sample each time is unbiased.

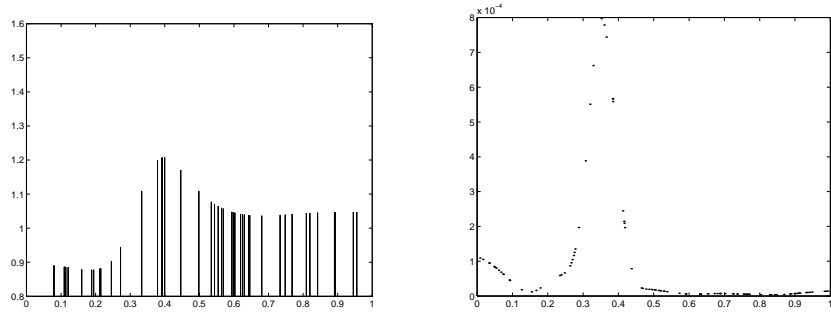
3.2 Experiments with Noisy Data

So far we had worked with clean data which is mathematically easier to handle. Noise was now added to the output of the plant (Figure 2(b)). The same algorithm was used to successfully identify the plant. Figure 7(b) is a final plot of the variance values at 100 random points and Figure 7(a) shows the actively collected data. The algorithm sampled more points in order to accommodate noise.

3.3 A Highly Noisy Plant, Multiple Experiments and Histograms of Collected Data

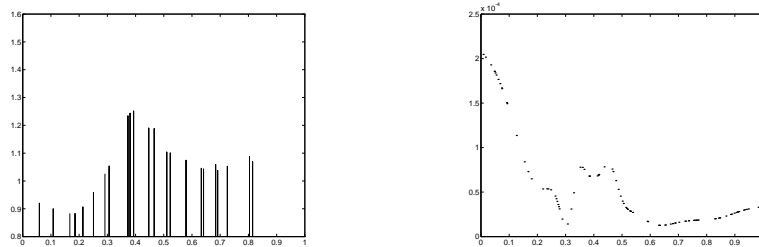
We decided to run each of the previous experiments 20 times and to examine the number of sampled data points in each case. Also a plant with high noise content (Figure 2(c)) was generated — the noise level being around four times that of the previous noisy plant (Figure 2(b)). Consequently we had three plants: with zero noise content, with relatively low noise content and relatively high noise. For each of these plants we ran 40 sets of experiments of which 20 used the active-learning algorithm and the other 20 performed passive learning. Thus a total of 120 experiments were performed.

The results are shown in Figures 8, 9 and 10 in the form of histograms of collected data. It is clear from these results that the active version of the algo-



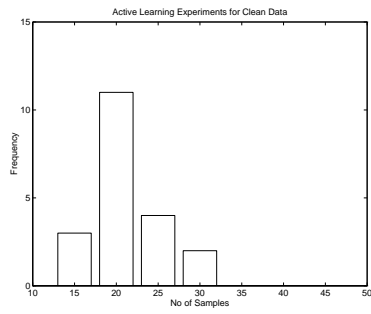
(a) Data Points for Correct Modelling found by Random Resampling (Passive Learning). (b) Final Variance Plot of the Agreeing Models at 100 Randomly-Selected Points with Random Resampling (Passive Learning).

Figure 6: Random Resampling (Passive Learning) Experiment — Sampled Data and Variance.

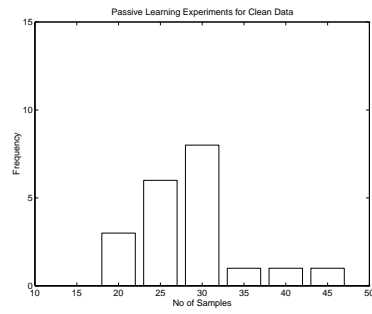


(a) Data Points for Correct Modelling found by Active Experimentation using Noisy Data. (b) Final Variance Plot of the Agreeing Models of Noisy Data at 100 Randomly-Selected Points.

Figure 7: Active Experimentation using Noisy Data — Data Samples and Final Model Variance.

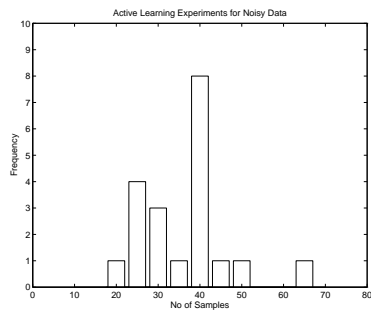


(a) Active Learning

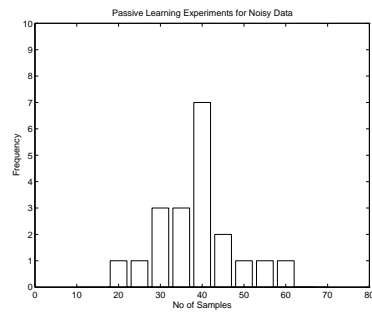


(b) Passive Learning

Figure 8: Histograms of Experimental Results for Clean Data

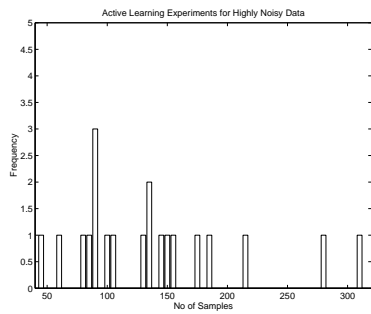


(a) Active Learning

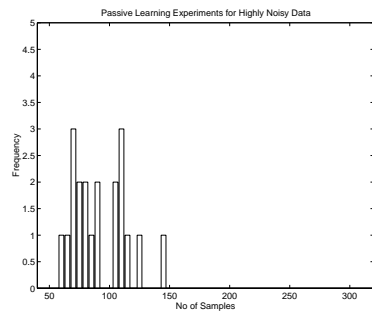


(b) Passive Learning

Figure 9: Histograms of Experimental Results for Noisy Data



(a) Active Learning



(b) Passive Learning

Figure 10: Histograms of Experimental Results for Highly Noisy Data

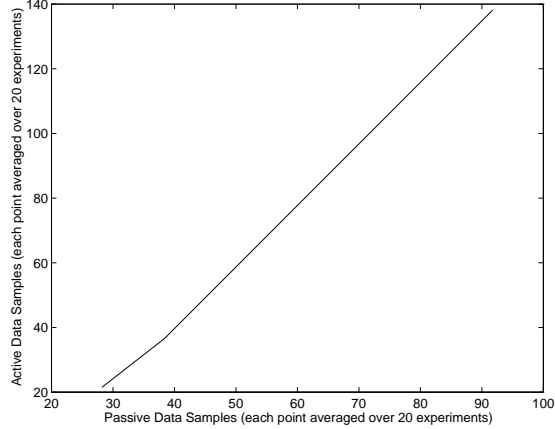


Figure 11: Active Learning Performance versus Passive

rithm converges with less data samples at zero noise level. When dealing with data of higher noise content the difference in the number of samples collected between active and passive learning is less significant.

4 Conclusion and Future Directions

From our experimental studies the following conclusions may be drawn

1. The proposed algorithm is feasible — most definitely in the case of Neural Network Learning.
2. The algorithm works in the case of both clean and noisy data — but more points need to be collected for identifying noisy data.
3. Although random resampling (or passive learning) increases computation and results in more samples being collected, it still causes the algorithm to converge.
4. The advantage of active learning over passive in terms of the number of labeled data points sampled (Figure 11), tends to be less apparent at higher noise levels. In fact in our experiments with the highly noisy data the performance of passive learning was superior. This is interesting and the subject of ongoing investigation.

The algorithm has been tested only upon a static plant so far. It needs to be considered whether a dynamic plant (with time-varying characteristics) can be identified with a modified version of this algorithm. Our feasibility experiments have been restricted to neural network modelling. Statistical nonlinear

regression techniques should also be applied to establish the versatility of this method.

References

- [1] D. Cohn, L. Atlas, and L. Ladner. Training connectionist networks with queries and selective sampling. In D. Touretzky, editor, *Advances in Neural Information Processing Systems 2*. Morgan Kaufmann, San Mateo, California, 1990.
- [2] David A. Cohn. Neural network exploration using optimal experiment design. Technical Report AIM-1491, Department of Brain and Cognitive Sciences, Massachusetts Institute of Technology, June 1994.
- [3] David A. Cohn, Zoubin Ghahramani, and Michael I. Jordan. Active learning with statistical models. In G. Tesauro, D. Touretzky, and T.K. Leen, editors, *Advances in Neural Information Processing Systems 7*, Cambridge, MA, 1995. MIT Press.
- [4] Howard Demuth and Mark Beale. *Neural Network Toolbox for Use with MATLAB*. The Math Works Inc., January 1994. User's Guide.
- [5] Y. Freund, H.S. Seung, E. Shamir, and N. Tishby. Information, prediction, and query by committee. In *Advances in Neural Information Processing Systems*, volume 5. Morgan Kaufmann, San Mateo, California, 1993.
- [6] L.K. Hansen and P. Salamon. Neural network ensembles. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(10):993–1001, October 1990.
- [7] Anders Krogh and Jesper Vedelsby. Neural network ensembles, cross validation and active learning. In G. Tesauro, D.S. Touretzky, and T.K. Leen, editors, *Advances in Neural Information Processing Systems 7*. MIT Press, Cambridge MA, 1995.
- [8] D. MacKay. Information-based objective functions for active data selection. *Neural Computation*, 4(4):590–604, 1992.
- [9] M. Plutowski and H. White. Selecting concise training sets from clean data. *IEEE Transactions on Neural Networks*, 4(2):305–318, 1993.
- [10] Mark Plutowski. *Selected Training Exemplars for Neural Network Learning*. PhD thesis, University of California, San Diego, 1994.
- [11] H.S. Seung, M. Opper, and H. Sompolinsky. Query by committee. In *Proceedings of the Fifth Workshop on Computational Learning Theory*, pages 287–294, San Mateo, California, 1992. Morgan Kaufmann.

- [12] Peter Sollich. Query construction, entropy and generalisation in neural network models. *Physical Review E*, 49:4637–4651, 1994.
- [13] Peter Sollich and David Saad. Learning from queries for maximum information gain in imperfectly learnable problems. In G. Tesauero, D.S. Touretzky, and T.K. Leen, editors, *Advances in Neural Information Processing Systems 7*. MIT Press, Cambridge MA, 1995.
- [14] Fabio Tamburini and Renzo Davoli. An algorithmic method to build good training sets for neural-network classifiers. Technical Report UBLCS-94-18, Laboratory for Computer Science, University of Bologna, July 1994.
- [15] Sebastian B. Thrun and Knut Möller. Active exploration in dynamic environments. In John E. Moody, Steven J. Hanson, and Richard P. Lippman, editors, *Advances in Neural Information Processing Systems 4*. Morgan Kaufmann, San Mateo, California, 1992.
- [16] D.H. Wolpert. Stacked generalisation. *Neural Networks*, 5(2):241–259, 1992.

About the Authors



Tirthankar RayChaudhuri was born in Calcutta, India in 1959. He received the B.E. and M.E. degrees in Electrical Engineering from Jadavpur University, Calcutta in 1981 and 1984 respectively. He worked for several years in industrial positions and is now a Ph.D candidate in Computing at Macquarie University. He is a member of the Australian Pattern Recognition Society, a graduate of the Institution of Engineers, Australia and a student member of the IEEE. His current research interests are neural networks, digital control systems and computer vision.



Leonard G. C. Hamey is a Lecturer in Computing at Macquarie University. His research interests centre on computer vision and artificial neural networks. Dr. Hamey received his BSc(hons) in Statistics from Macquarie University in 1982 and his PhD in Computer Science from Carnegie Mellon University in 1988. He is a member of IEEE and the Australian Pattern Recognition Society. Besides his professional involvements, Leonard is a Christian and active in his local church.