An Empirically-Sourced Heuristic for Predetermining the Size of the Hidden Layer of a Multi-layer Perceptron for Large Datasets

Amanda Lunt(✉) and Shuxiang Xu

School of Engineering and ICT, University of Tasmania, Launceston, Australia
{Amanda.Lunt, Shuxiang.Xu}@utas.edu.au

Abstract. We recommend a guiding heuristic to locate a sufficiently-sized multilayer perceptron (MLP) for larger datasets. Expected to minimise the search scope, it is based on experimental research into the comparative performance of 14 existing approaches with global minimum ranges on 31 larger datasets. The most consistent performer was Baum's [1] equation that sets the number of hidden neurons equal to the square root of the number of training instances.

Keywords: Neural network · Multilayer Perceptron · Hidden layer size · Global minimum · Local minimum

1 Introduction

Trained under supervision, a 3-layer multilayer perceptron (MLP) will find 'hidden' relationships within a set of data by approximating continuous functions [2]. The trained network may then be used for prediction tasks on previously unseen data from the same domain, with the final configuration unique to each specific dataset. The size of the hidden middle layer, N^h, has a strong bearing on the prediction accuracy of the final model [3], yet the predominant technique to locating N^h is resizing through trial-and-error. Exhaustive search through a range of N^h becomes problematic with larger datasets, increasing demands on processor capacity and extending the time required for training. The usefulness of the heuristic proposed in this paper is in minimising the scope of the search to reach a suitably optimal network size. We note that a reasonable network architecture may not be limited to a single 'correct' configuration [4] so long as the underlying function can be learnt while retaining enough smallness to generalise [5].

A set of proposed mathematical relationships between N^h and the numbers of input neurons, N^i, output neurons, N^o, both fixed, and instances of the dataset used for training, N^{tr}, is summarised in Table 1. We used N^{tr} for our calculations rather than N^{TOT}, total number of instances, as it directly relates to the training process.

B.H. Kang and Q. Bai (Eds.): AI 2016, LNAI 9992, pp. 542–547, 2016.
DOI: 10.1007/978-3-319-50127-7_47

Table 1. Fourteen ways to determine N^h. Approach number was attributed randomly.

Source	Equation	Approach number	Source	Equation	Approach number
[6, 7]	$N^h \geq 2N^i + 1$	(1)	[1, 8]	$N^h = \frac{N^{tr}}{N^i}$	(8)
[9] in [10]	$N^h \leq \frac{N^{tr}}{(N^i+1)}$	(2)	[1, 8]	$N^h = \frac{N^{tr}}{(N^i+N^o)}$	(9)
[1]	$N^h = \sqrt{N^{tr}}$	(3)	[11]	$N^h = 2N^i$	(10)
[12]	$N^h = \log(N^{tr})$	(4)	[13]	$N^h > N^o$	(11)
[14]	$N^h = \sqrt{N^i N^o}$	(5)	[15, 16]	$N^h \leq N^i - 1$	(12)
[17]	$N^h = \frac{(2N^i+3)}{(N^i-2)}$	(6)	[1, 16]	$N^h = \frac{N^i}{N^{tr}}$	(13)
[18]	$N^h = C\left(\frac{N^{tr}}{N^i \log N^{tr}}\right)^{1/2}$	(7)	[15]	$N^h \geq \frac{N^i}{3}$	(14)

1.1 Research Question

Which of the existing approaches can assist the search for a suitable number of neurons in the single hidden layer of a MLP for larger datasets?

2 Experiment

Our simple experiment investigates the performance of each approach when compared with global minimum benchmarks [19]. Thirty-one datasets with many attribute-target pairs or high dimensionality were sourced [20–22] (see Table 2).

A lower and upper limit to N^h was established for the training of each dataset based on calculations from the approaches in Table 1. We set lower bound at the calculation closest to 0, while upper bound was based on a sense of being able to train to that N^h, with flexibility to extend with working processor capacity. Where an approach takes the form of a lower or upper bound, the calculated N^hat the bound was used.

Weights were initialised randomly to represent prior knowledge [23]. Training, test and validation sets (70-15-15% of N^{TOT}) were also randomly generated for the best opportunity to locate the global minimum [24]. Each-sized network was trained 10 times with cross-validation, accounting for random influences [25]. We performed our experiment using MATLAB Neural Network Toolbox version 6 add-on's `patternnet` function with the scaled conjugate backpropagation algorithm [26, 27].

Table 2. Characteristics of 31 datasets. Most are from http://archive.ics.uci.edu/ml/datasets except (b) http://mldata.org and (c) http://osmot.cs.cornell.edu/kddcup. Larger sets were excluded as too slow to train with available resources.

Working title	N^i	N^o	N^{TOT}	Working title	N^i	N^o	N^{TOT}
(b) 2Norm	20	2	7400	PokerHand	10	10	25010
Abalone	8	29	4177	(c) ProteinHomology	74	1	145751
AdultIncome	14	2	48842	PubChem362	144	2	4279
Chess	6	18	28056	PubChem456	153	2	9982
Connect4	42	3	67557	PubChem687	153	2	33067
FirstOrderTheorem	51	6	6118	(c) QuantumPhysics	78	1	50000
Gisette	5	2	7000	Shuttle	9	7	58000
LandSat	36	6	6435	Skin	3	2	245057
LetterRecognition	16	26	20000	Spambase	57	1	4601
Madelon	50	2	2600	Thyroid	21	3	7200
MagicGamma	10	2	19020	WallFollowRobot2	2	4	5456
Musk2	16	2	6598	WallFollowRobot4	4	4	5456
Nomao	17	2	34465	WallFollowRobotFull	24	4	5456
Nursery	8	5	12960	Waveform	21	3	5000
OptDigits	64	10	5620	WineQuality	11	11	4898
PageBlocks	10	5	5473				

3 Results

The global minimum was located for each dataset at the N^h with the smallest averaged performance error from the mean of squared errors comparing the actual output against the desired output [25]. Approaches (3) and (1) calculated the global minimum in one case each, *WallFollRobot2* and *AdultIncome* respectively.

Not all approaches gave us a sensible calculation for N^h for every dataset. We obtained a result for all of the 31 datasets with approaches (4), (5) and (7) only. Table 3 demonstrates a combination of this raw count [*A*] and the count of datasets where

Table 3. An excerpt of the simple ranking of approaches according to relative usefulness, ordered from 'most' useful and truncated for brevity. [***B***] was scaled in the final column to indicate its relationship to the research question, with no impact on the final rank.

Approach number	Result count [*A*]	Comparison of means [*B*]	[*A*] + [*B*]	[*A*] + 2[*B*]
(5)	31	26	57	83
(7)	31	25	56	81
(3)	29	23	52	75
(1)	28	23	51	74
(12)	28	22	50	72
(4)	31	18	49	67
(14)	22	18	40	58

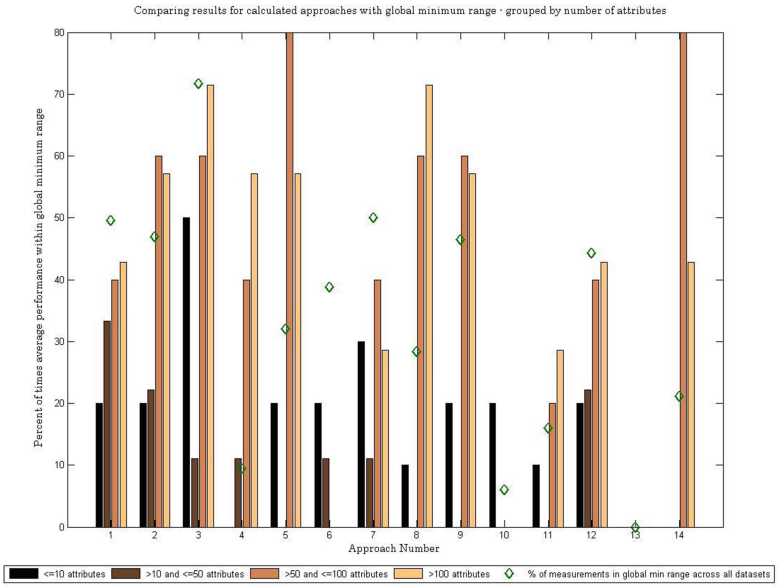

Fig. 1. Performance at calculated N^h compared with global minimum range over 31 datasets.

performance at the approach's calculated N^h intersects with the global minimum (95% CI) from a multiple comparison of means [*B*].

Figure 1 gives an overview of two further comparisons with the performance range at the global minimum N^h. Single diamonds are derived from the count of *individual* performance measures for an approach within the global minimum range over all datasets. You can clearly see the success of approach (3) $N^h = \sqrt{N^{tr}}$ in this, with occurrences 71.7% of times across all datasets.

The second set of comparisons is presented as bar graphs that have been separated into $\boldsymbol{N^i}$ groupings to allow for disparity in attribute dimensionality across the datasets: $N^i \leq 10$; $10 < N^i \leq 50$; $50 < N^i \leq 100$; and $N^i > 100$. This ratio is the per cent of times an *average* of the 10 performance measures at each pre-calculated N^h occurred within the range of performances recorded at the global minimum N^h, grouped by $\boldsymbol{N^i}$. Approaches (5) and (14) were both highly successful in the $50 < N^i \leq 100$ group (4 out of 5 cases), with (3) and (8)'s average occurring within the global minimum range for 5 out of the 7 cases in the $N^i > 100$ group.

Also of note, (2), (3), (8) and (9)'s averages placed in the global minimum range for the $50 < N^i \leq 100$ group in 3 of the 5 cases. The results for the two groups where $N^i \leq 50$ (the remaining 19 datasets) were no better than 50%.

4 Discussion and Conclusion

We empirically determined a single, optimal structure between lower and upper bounds for N^h for each dataset, comparing the performance of each approach with the range at this global minimum in several ways.

All approaches other than (3) recorded an individual measurement in all datasets' global minimum ranges in 50% or fewer cases. Approach (3)'s consistency (over 71%) is notable due to the variations between the 31 datasets.

With averaged performances, approaches (5) and (14)'s 80% success where $50 < N^i \leq 100$ is tempered by there being only 5 datasets in that group. In the initial ranking according to relative usefulness, approach (5) was ranked first, with (14) lower down. Both of these approaches consider a relationship with N^i. In the $N^i > 100$ group, approaches (8) and (3) succeeded in 5 out of the 7 datasets. Both consider a relationship with N^{rr}. In the usefulness ranking, (8) was 11$^{\text{th}}$ and (3) third. The success rate in the results grouped for all $N^i \leq 50$ was 50% or less.

On the basis of these findings, we recommend the following heuristic: in cases of more than 50 attributes in a dataset, apply the highly successful approaches (5) and (14) for $50 < N^i \leq 100$ and (8) and (3) for $N^i > 100$. For other cases, use approach (3) for an indication of reasonable network performance.

References

1. Baum, E.B.: On the capabilities of multilayer perceptrons. J. Complex. **4**, 193–215 (1988)
2. Cybenko, G.: Approximation by superpositions of a sigmoidal function. Math. Control Signals Syst. **2**, 303–314 (1989)
3. Hornik, K.: Approximation capabilities of multilayer feedforward networks. Neural Netw. **4**, 251–257 (1991)
4. Zeng, X., Yeung, D.S.: Hidden neuron pruning of multilayer perceptrons using a quantified sensitivity measure. Neurocomputing **69**, 825–837 (2006)
5. Aran, O., Yildiz, O.T., Alpaydin, E.: An incremental framework based on cross-validation for estimating the architecture of a multilayer perceptron. Int. J. Pattern Recogn. Artif. Intell. **23**, 159–190 (2009)
6. Hecht-Nielsen, R.: Kolmogorov's mapping neural network existence theorem. In: Proceedings of IEEE First Annual International Conference on Neural Networks, pp. III-11–III-14. (1987)
7. Sprecher, D.A.: A universal mapping for kolmogorov's superposition theorem. Neural Netw. **6**, 1089–1094 (1993)
8. Barron, A.R.: Approximation and estimation bounds for artificial neural networks. Mach. Learn. **14**, 115–133 (1994)
9. Rogers, L.L., Dowla, F.U.: Optimization of groundwater remediation using artificial neural networks with parallel solute transport modeling. Water Resour. Res. **30**, 457–481 (1994)
10. Somaratne, S., Seneviratne, G., Coomaraswamy, U.: Prediction of soil organic carbon across different land-use patterns. Soil Sci. Soc. Am. J. **69**, 1580–1589 (2005)
11. Denker, J.S., Schwartz, D., Wittner, B., Solla, S., Howard, R., Jackel, L., Hopfield, J.: Large automatic learning, rule extraction and generalization. Complex Syst. **1**, 877–922 (1987)

12. Wanas, N.M., Auda, G.A., Kamel, M.S., Karray, F.O.: On the optimal number of hidden nodes in a neural network. In: IEEE Canadian Conference on Electrical and Computer Engineering 1998, vol. 2, pp. 918–921 (1998)
13. Gallinari, P., Thiria, S., Soulie, F.F.: Multilayer perceptrons and data analysis. In: IEEE International Conference on Neural Networks 1988, vol.391, pp. 391–399 (1988)
14. Shibata, K., Ikeda, Y.: Effect of number of hidden neurons on learning in large-scale layered neural networks. In: ICROS-SICE International Joint Conference 2009, pp. 5008–5013. SICE, Fukuoka International Congress Center, Japan (2009)
15. Arai, M.: Bounds on the number of hidden units in binary-valued three-layer neural networks. Neural Netw. **6**, 855–860 (1993)
16. Huang, S.-C., Huang, Y.-F.: Bounds on the number of hidden neurons in multilayer perceptrons. IEEE Trans. Neural Netw. **2**, 47–55 (1991)
17. Deepa, S.N., Sheela, K.G.: Estimation of number of hidden neurons in back propagation networks for wind speed prediction in renewable energy systems. Draft (2013)
18. Xu, S., Chen, L.: A novel approach for determining the optimal number of hidden layer neurons for FNNs and its application in data mining. In: Proceedings the 5th International Conference on Information Technology and Applications 23–26 June 2008, Cairns, Qld, pp. 683–686 (2008)
19. Gorman, R.P., Sejnowski, T.J.: Analysis of hidden units in a layered network trained to classify sonar targets. Neural Netw. **1**, 75–89 (1988)
20. Bache, K., Lichman, M.: UCI Machine Learning Repository. School of Information and Computer Science, University of California, Irvine (2013)
21. Hoyer, P.O., Ong, C.S., Henschel, S., Braun, M.L., Sonnenburg, S.: IDA Benchmark Repository, vol. 0.1.6. ML Group, Berlin (2013)
22. ACM Special Interest Group on Knowledge Discovery and Data Mining: KDD Cup 2004: Particle physics; plus protein homology prediction. ACM (2004). http://www.kdd.org
23. Dayhoff, J.: Neural Network Architectures: An Introduction. International Thomson Computer Press, Boston (1996)
24. http://ulcar.uml.edu/∼iag/CS/Intro-to-ANN.html
25. Flexer, A.: Statistical evaluation of neural network experiments: minimum requirements and current practice, pp. 1005–1008. The Austrian Research Institute for Artificial Intelligence, Schottengasse 3, A-1010 (1994)
26. Demuth, H., Beale, M., Hagan, M.: Neural Network Toolbox 6 User's Guide. The MathWorks Inc., Natick (2009)
27. Møller, M.F.: A scaled conjugate gradient algorithm for fast supervised learning. Neural Netw. **6**, 525–533 (1993)