Determining the Number of Hidden Units in Multi-Layer Perceptrons using F-Ratios

Ben H. Jansen and Pratish R. Desai

Department of Electrical Engineering and Bioengineering Research Center,
University of Houston, Houston, TX 77204-4793

Abstract

The hidden units in multi-layer perceptrons are believed to act as feature extractors. In other words, the outputs of the hidden units represent the features in a more traditional statistical classification paradigm. This viewpoint offers a statistical, objective approach to determining the optimal number of hidden units required. This approach is based on a F-ratio test, and proceeds in an iterative fashion. The method, and its application to simulated time-series data are presented.

1 Introduction

Artificial neural nets are increasingly being used for a variety of pattern recognition problems [1, 7, 8, 9]. Recently, Gallinari *et al.* [4] proved the formal equivalence between the linear multi-layer perceptron (MLP) and Discriminant Analysis (DA). Specifically, they noted that in a linear MLP, the first layer of weights realizes a DA of the input data, that is, projects the inputs onto a subspace so as to form well-aggregated clusters for each class. Experiments on problems with an increasing degree on nonlinearity demonstrated that DA on the hidden states gave similar performance as that of MLP. This suggests that hidden units activations can be interpreted as features. Consequently, feature selection techniques such as commonly used in statistical pattern recognition may be used to determine which hidden units are most significant, and which hidden units may be eliminated. One such method is presented here, and we show its usefulness in a problem involving the detection of specific waveforms in a time-series.

The results presented here are part of a larger study (see [2]), which investigated the use of recurrent and feed-forward neural networks for the detection of K-complexes in recordings of the electrical activity of the brain during sleep (electroencephalograms or EEGs). K-complexes are relatively large waves with a duration of between 500 and 1500 msec often seen during Sleep Stage 2. Automated detection of K-complex activity in the EEG is an important component of sleep stage EEG monitoring. Neural nets have been applied before to EEG waves with some success [3, 6].

2 Methods

The experiments described here involve the use of the multi-layer perceptron to detect bi-phasic triangular waveforms of various shapes in model-generated time-series. Both the triangular waveform and the time-series were made to resemble actual sleep EEG and K-complexes. The magnitude was extracted from segments of these time-series using the Fourier transform, and used as input to the neural nets. Once training was complete, a step-wise procedure was applied to determine the optimal number of hidden units required. The reduced net was then trained again, and tested using other data sets. The details of the data generation, net architecture and input, and net optimization procedure are provided next.

2.1 Data Generation

EEG data were obtained from six subjects. Five EEG channels (Fp1, F3, F4, T3, and T4) with observable K-complexes were used. An artificial data set was generated by producing a time series resembling actual EEG, to which a pattern representing a K-complex was added. EEG-like activity was produced through an 8th-order autoregressive (AR) model. The model coefficients were computed from actual EEG segments in the neighborhood (within 5 sec) of K-complexes (as identified by an electroencephalographer) to be used in generating "positive" examples, and from EEG taken far away from K-complexes to generate "negative" examples. Triangular patterns, resembling a K-complex, were placed in the artificial, "positive" EEG segments at various locations. No such pattern was added to the "negative" artificial EEG segments. Each positive or negative example consisted of 1000 sam-

ple points, representing 10 sec of data. The shape of the pattern differed between each of the positive examples. Specifically, the peak-to-peak amplitude of the pattern was varied in such a way that the ratio of the peak-to-peak amplitude of the pattern and the root-mean-square (rms) of the background activity would range between 0.05 and 0.15, the pattern was inserted at a random location, and the duration of the pattern varied randomly within a range similar to that of actual K-complexes. Three of such data sets were generated, referred to as the Train, Test1, and Test2 set, respectively. The Train and Test1 ("seen") data sets were generated from the same AR models, but different seed points were used to generate the EEG-like data and to control the shape and the location of the K-complex-like pattern. The Test2 data set ("unseen") was generated from the AR models obtained from EEG examples not included in the training data set.

2.2 Net Input and Architecture

Our basic approach was to compute the magnitude spectrum of 10 sec signal segments (using a FFT routine). These data were input to a multi-layer perceptron, which was trained using the backpropagation algorithm. Unless otherwise stated, the inputs to the net consisted of the magnitude at each of 64 frequency bins. A 512-point Fast Fourier Transform (FFT) was computed to obtain the magnitude, which was subsequently smoothed and reduced to 64 sample values by averaging over 8 adjacent points. These smoothed magnitude and phase values were then normalized between 0 and 1 for use as inputs to the neural network input nodes. Experiments with the hidden unit selection technique were performed on nets with 64 input units, one hidden layer with 8 units, and one or two output units.

2.3 Optimizing using Discriminant Analysis

The core of the optimization procedure derives from stepwise feature selection methods often used in statistical pattern recognition. In these approaches, the 'best' feature is selected from a pool of features using some criterion. All the pair-wise combinations of this best feature with any of the remaining features are explored to determine which is the 'best' pair, and if this additional feature has any discriminating power. If the answer to the last question is yes, triplets are formed by combining the best pair with any of the remaining

features. This process is repeated until it is found that adding a feature to the ones already selected does not lead to significant improvements in the criterion function.

In the present application, the outputs (activations) of the hidden units are treated as features. The Wilks' Λ is used as the criterion function to determine which feature should be selected. The Wilks' Λ is a multi-variate statistic that tests the equality of group means for the selected features [5]. The Λ may be converted to an approximate F-ratio. In the present method, the conditional F-ratio is used. The latter measures how much a given feature contributes to the group differences given the variables already selected. At each step the conditional F-ratios are computed for each feature. If a feature which has already been selected has a non-significant F-ratio, it is removed. If none of the features are removed, then the feature which creates the largest change in the criterion function is added to the selection. If none of the remaining features have a significant F-ratio, the procedure halts.

3 Results

In the first experiment, magnitude data were used to train a single output net with the Train data set. Upon convergence, training was halted, and the Train, Test1, and Test2 data sets were input to determine the classification performance of the net. A correct classification rate of 100% was found for Train, 92% for Test1, and 87% for Test2, respectively. Following this stage, the activations of the 8 hidden units for each example in the Train data set were recorded and subjected to the F-ratio test. The results shown in Table 1. Hidden units are listed in the order in which they were selected, together with their F-value at the time of selection.

The relatively large difference in F-value between unit 3 and 7 suggests that unit 3 is a very important feature. The scatter plot of the activations of unit 3 and 7, in response to the presentation of the training examples, is shown in Figure 1. It can be observed that the two classes are very well separated, except for a few positive examples that fall in the negative class cluster.

Mamelak, *et al.* [7] found that the overall performance of a single output net is usually worse than a 2 output net for a two-class problem. Even though each example can be assigned an unique pattern, with no indeterminate pat-

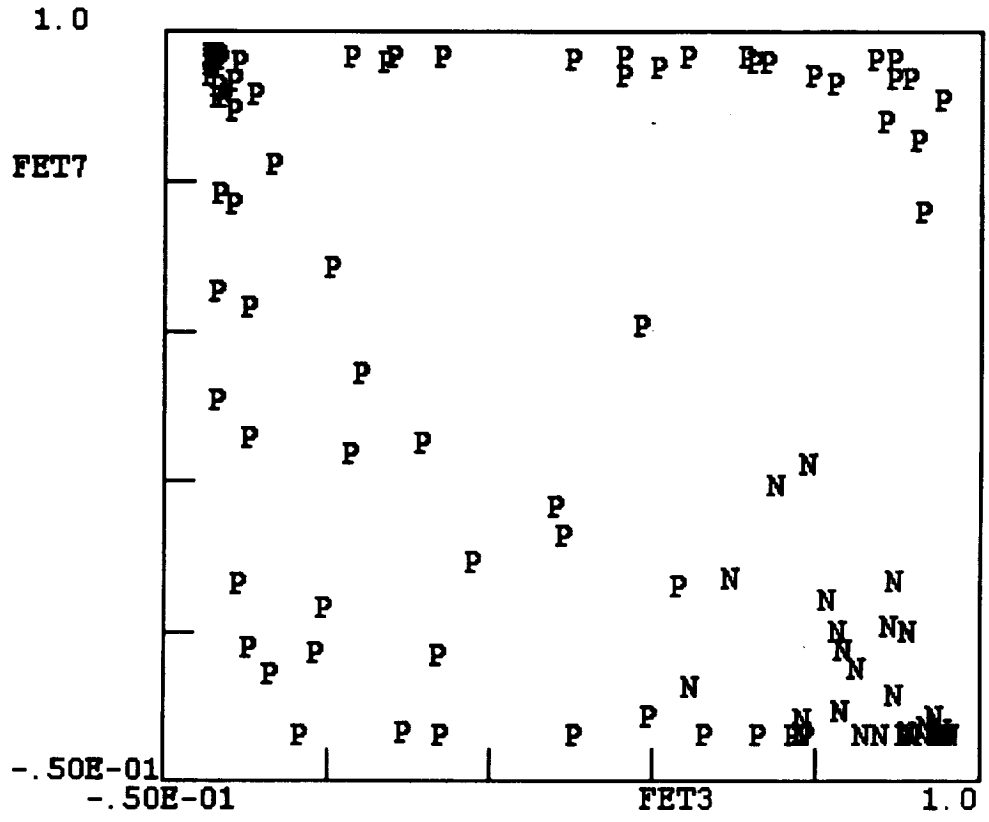

Figure 1: *Scatter plot of the activations of 2 hidden units (3rd and 7th), for the net with 8 hidden units and 1 output unit trained on the power spectra of exp.4.*

Table 1: *F-values obtained by performing an F-test on the 8 hidden unit outputs of a single-output net .*

Hidden Unit	F-value
3	155.88
7	37.77
8	68.73
2	43.43
5	43.51
6	34.28
1	4.25

terns, if a single output unit is used for a two-class problem, they found that the mapping between input and output patterns is actually too restricted, limiting the ability of the single-output net to fine-tune the threshold levels for all remaining patterns. We decided to explore this issue by applying the same training set as used above to a net with 8 hidden units and 2 output units. The net converged in 1187 cycles. The results of the F-test on the 8 hidden unit outputs are presented in Table 2.

Table 2: *F-values obtained by performing an F-test on the 8 hidden units activations of a net with 2 output units*

Hidden Unit	F-value
5	203.22
8	106.47
1	193.73
7	12.12
3	34.13
2	9.66

Observe that units 5, 8, and 1 produce large F-values, indicating their relative importance. Figure 2 shows the scatter plot for the first two selected hidden units. As shown, both classes are well clustered and are sitting well

in the corners of the square box. Compared to the results obtained with the net with one output unit (see Figure 1), the separation between the two classes is better defined. This confirms the observations made by Mamelak *et al.*.

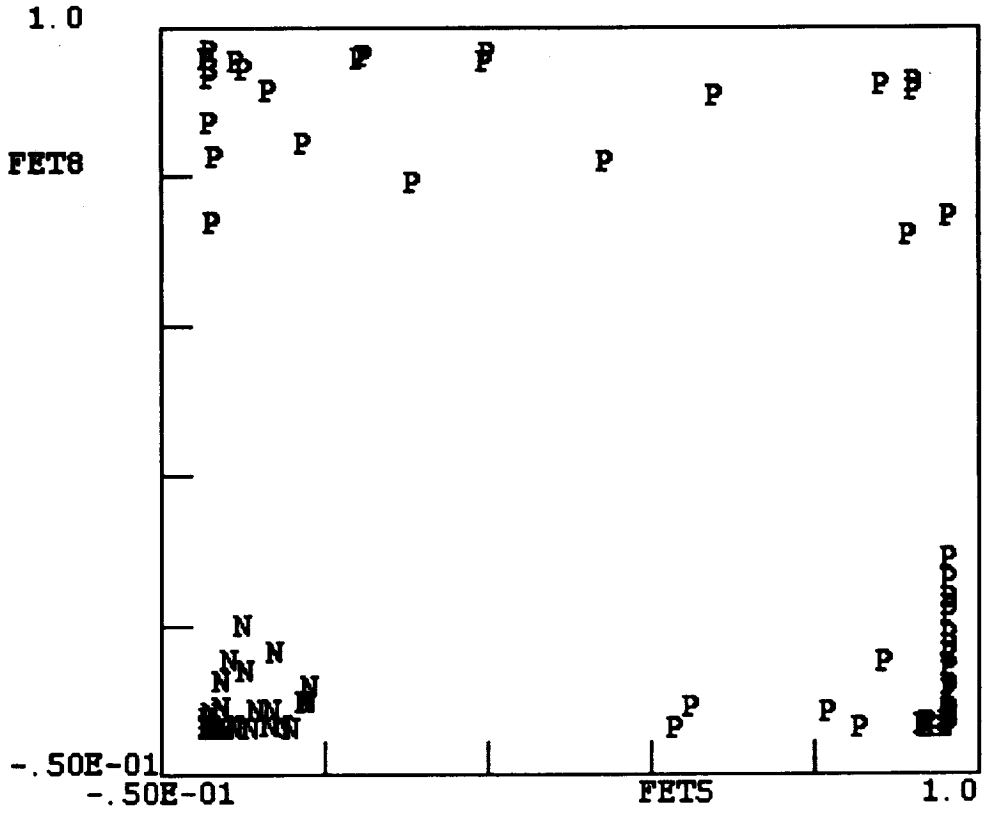

Figure 2: *Scatter plot of the activations of 2 hidden units (5th and 8th), for the net with 8 hidden units and 2 output units.*

Both of the aforementioned experiments suggest that a net with just two hidden units would perform as well as a net with 8 hidden units. This was explored in the next experiment involving a net with 2 hidden units and 2 output units. Again, training was done using the magnitude data, and it was found that the net converged in 1503 cycles. The scatter diagram of the

activations of the two hidden units is shown in Figure 3. As one can see,

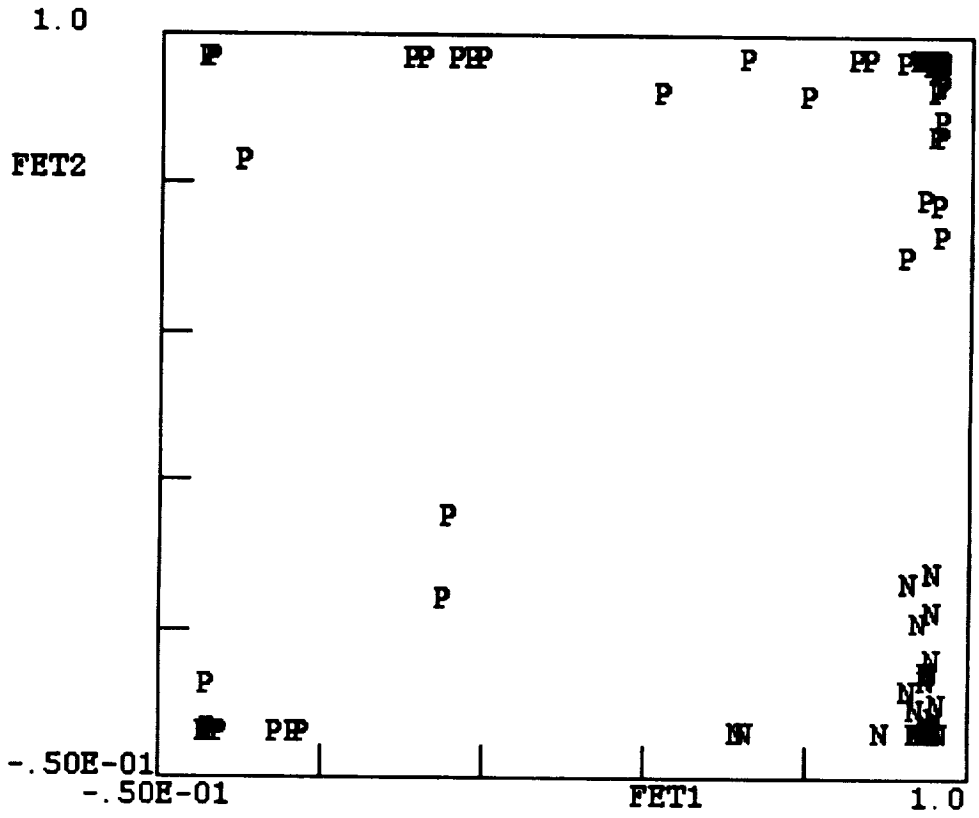

Figure 3: *Scatter plot of the activations of 2 hidden units for the net with 2 hidden units and 2 output units.*

the two classes are well-separated and occupying the corners of the feature space. The negative examples (N) are grouped into one corner, whereas the positive examples (P) are distributed over the other 3 corners. There was no specific relationship between the positive examples within one corner. This strongly suggests that a net with two hidden units should be sufficient to classify all the examples correctly. This was tested on the Train, Test1, and Test2 data sets, and although not perfect classification results were obtained for the two testing sets, the results were not significantly different from those

obtained with a net with 8 hidden units and 2 output units, and with a net with 8 hidden units and a single output.

4 Conclusions

We have presented a simple technique for the *a posteriori* determination of the hidden units required in a multi-layer perceptron. The method uses the fact that the hidden units appear to perform a discriminant analysis, essentially extracting features from the neural net input. The relative importance of each hidden unit can be assessed using an F-ratio test. In addition, the absolute value of the F-ratio provides insight in the degree of confidence one may place in the classifications produced by the net. For example, if the most significant hidden units have F-values barely above the level of significance, the classifying power of the net will be small.

The method described here is part of most widely available software packages for multi-variate data analysis, including BMDP and SPSS, making it very easy to apply this method.

References

[1] E. Barnard, R.A. Cole, M.P. Vea, and F.A. Alleva, "Pitch detection with a Neural-net classfier". *IEEE Transactions on Signal Processing*, vol. SP-39, pp. 298-307, 1991.

[2] P. R. Desai, *Waveform Detection Using Artificial Neural Networks*, M.Sc.-Thesis, Department of Electrical Engineering, University of Houston, 1991.

[3] R.C. Eberhart, R.W. Dobbins, W.R.S. and Webber, "EEG waveform analysis using casenet". *Proceedings of IEEE Engineering in Medicine and Biology Society 11th Annual International Conference*, pp. 2046, 1989.

[4] P. Gallinari, S. Thiria, F. Badran and F. Fogelman-Soulie, "On the relations between discriminant analysis and multi-layer perceptrons". *Neural Networks*, vol. 4, pp. 349-360, 1991.

[5] M. James, *Classification Algorithms*, John Wiley & Sons, New York, 1985.

[6] B.H. Jansen, "Artificial Neural Nets for K-Complex Detection". *IEEE Engineering in Medicine and Biology*, vol. 9, n. 3, pp. 50-52, 1990.

[7] A.N. Mamelak, J.J. Quattrochi, and J.A. Hobson, "Automated staging of sleep in cats using neural networks". *Electroencephalography and clinical Neurophysiology*, 79, pp. 52-61, 1991.

[8] T.J. Sejnowski, and R.P. Gorman, "Analysis of hidden units in a layered network trained to classify sonar targets". *Neural Networks*, vol. 1, pp. 75-89, 1988.

[9] A.H. Waibel, and K.J. Lang, "A time delay neural network architecture for isolated word recognition". *Neural Networks*, vol. 3, pp. 23-43, 1990.