Probabilistic Interpretation of Feedforward Classification Network Outputs, with Relationships to Statistical Pattern Recognition

JOHN S. BRIDLE
Speech Research Unit
Royal Signals and Radar Establishment
St. Andrews Road Great Malvern
UK WR14 3PS

Abstract

We are concerned with feed-forward non-linear networks (multi-layer perceptrons, or MLPs) with multiple outputs. We wish to treat the outputs of the network as probabilities of alternatives (*e.g.* pattern classes), conditioned on the inputs. We look for appropriate output non-linearities and for appropriate criteria for adaptation of the parameters of the network (*e.g.* weights). We explain two modifications: probability scoring, which is an alternative to squared error minimisation, and a normalised exponential (**softmax**) multi-input generalisation of the logistic non-linearity. The two modifications together result in quite simple arithmetic, and hardware implementation is not difficult either. The use of **radial units** (squared distance instead of dot product) immediately before the **softmax** output stage produces a network which computes posterior distributions over class labels based on an assumption of Gaussian within-class distributions. However the training, which uses cross-class information, can result in better performance at class discrimination than the usual within-class training method, unless the within-class distribution assumptions are actually correct.

1 Networks and Probabilistic Models

1.1 The stochastic model paradigm

Currently the most successful approach to automatic speech recognition is based on *Stochastic Models.* The approach generalises parametric statistical pattern recognition, and is also important in other application domains, such as computational vision.

NATO ASI Series, Vol. F 68
Neurocomputing
Edited by F. Fogelman Soulié and J. Hérault
© Springer-Verlag Berlin Heidelberg 1990

The method consists in treating the data as if it were the output of a stochastic system (one governed by probabilistic laws) [1]. The structure of the model is a pattern of direct dependencies between internal (hidden) and external (visible) random variables, and it is usually informed by insight into the structure of the data or of the real generator (*e.g.* vocal tracts in the case of speech). There is also a (possibly large) set of parameters of the model, which need to be estimated (such as conditional probabilities, means and variances). There are usually separate algorithms for recognition (interpreting the data in terms of the model) and for learning (estimating parameters.)

The basis of the recognition process is often to compute the likelihood of the data conditioned on some unseen random variable of interest, such as the class of the pattern: $P(\text{Data} \mid \text{Class})$. Using Bayes rule, the class conditional probabilities are then

$$P(\text{Class} \mid \text{Data}) = P(\text{Data} \mid \text{Class})P(\text{Class}) \Big/ P(\text{Data}),$$

where $P(\text{Class})$ are the prior probabilities (often assumed equal) and

$$P(\text{Data}) = \sum_{\text{classes}} P(\text{Data} \mid \text{Class})P(\text{Class}).$$

If we only require the most likely class then this denominator can be ignored.

One of the simplest stochastic models (and one we shall return to below) is a set of Gaussian distributions, one for each class. In this case the parameters are the priors, the means, and the covariance matrices. We may choose to restrict the covariance matrices in various ways, such as zero except on the diagonals (independence assumption), or still more by assuming the covariance matrix is unit diagonal times a constant. We might also assume that the covariance matrices for the different classes are related (*e.g.* equal). We would usually estimate the means etc. from the sample means, etc. (*i.e.* maximum likelihood estimation for each class separately).

1.2 Feed-forward nonlinear "neural" networks

In contrast to the stochastic model approach, a neural network approach typically starts with assumptions about the form of a suitable recognition process, such as a non-linear feedforward network, and we adjust its parameters (*e.g.* weights) to optimise some measure of performance.

The standard "multi-layer perceptron" (MLP) can be taken as a semi-linear logistic feedforward network with squared error minimisation, trained using some optimisation technique exploiting the error back-propagation method for computing partial derivatives with respect to the weights. Such MLPs are often used for pattern classification, using one-from-N coding at the output. (*i.e.* One output for each class, targets are zero except for unity at the true class line.) However, this structure is really only suitable for cases where the outputs correspond to (approximately) independent properties of the input, such presence/absence of objects which may appear together, or in the use of componential representations, *e.g.* [2]. Later we consider a more appropriate structure for 1-from-N classification problems.