MIXTURE OF EXPERTS ARCHITECTURES FOR NEURAL NETWORKS AS A SPECIAL CASE OF CONDITIONAL EXPECTATION FORMULA

Jiří Grim

Department of Pattern Recognition
Institute of Information Theory and Automation
Academy of Sciences of the Czech Republic

Abstrakt

Recently a new interesting architecture of neural networks called "mixture of experts" has been proposed as a tool of real multivariate approximation or classification. It is shown that, in some cases, the underlying problem of prediction can be solved by estimating the joint probability density of involved variables. Assuming the model of Gaussian mixtures we can explictly write the optimal minimum dispersion prediction formula which can be interpreted as a mixture-of-experts network. In this way the optimization problem reduces to standard estimation of normal mixtures by means of EM algorithm. The computational aspects are discussed in more detail.

1 Introduction

Mixture-of-experts architecture typically consists of two parallel feedforward networks having the same real input vector $\boldsymbol{x} \in R^N$: a network of "expert" units performing prediction of some output vector $\boldsymbol{y} \in R^K$ and a gating network which weights the outputs of expert units to form the overall output.

The original heuristic idea was to simplify e.g. a complex problem of linear regression by dividing the input space into smaller regions and solving separately the presumably

[0] Early version of the paper: Grim J., "Mixture of experts architectures for neural networks as a special case of conditional expectation formula." In: Proceedings of the 1st IAPR TC1 Workshop on Statistical Techniques in Pattern Recognition. (Pudil P., Novovičová J., Grim J. eds.). ÚTIA AV ČR, Praha 1997, pp. 55-60.

less complex regression tasks within the input subsets. Thus, by proper switching between the regions, the global functioning could achieve the quality of locally optimal solutions ("local experts"). This original "divide and conquer" principle was further generalized by introducing "soft" gating allowing for "soft" partitioning of the input space [14] and also by considering hierarchical structures (cf. [7]). Optimization of the mixture-of-experts networks is a difficult problem. Roughly speaking, the recently published techniques succesfully combine EM algorithm and sofisticated gradient- and regression methods. In practical problems the reported computational results appear to be satisfactory (cf. [6], [8]).

From a statistical point of view the underlying problem can be formulated as a prediction of a real random vector $\mathbf{Y}$ given the value $\boldsymbol{x} \in R^N$ of a random vector $\mathbf{X}$. If the joint probability density functions $P(\boldsymbol{x}, \boldsymbol{y})$ is known then we can write the optimal minimum-dispersion prediction formula in terms of conditional expectation

$$\hat{\boldsymbol{y}}(\boldsymbol{x}) = E[\mathbf{Y}|\boldsymbol{x}] = \int \boldsymbol{y} P(\boldsymbol{y}|\boldsymbol{x}) d\boldsymbol{y}, \tag{1}$$

$$P(\boldsymbol{y}|\boldsymbol{x}) = \frac{P(\boldsymbol{x}, \boldsymbol{y})}{P(\boldsymbol{x})}, \quad P(\boldsymbol{x}) = \int P(\boldsymbol{x}, \boldsymbol{y}) d\boldsymbol{y}, \quad \boldsymbol{x} \in R^N. \tag{2}$$

It appears that, essentially, the optimization methods for mixture-of-experts architectures locally approximate the conditional expectation (1) by means of EM algorithm combined with different regression or gradient techniques.

An alternative possibility widely used in statistical decision-making is to estimate the unknown probability density function $P(\boldsymbol{x}, \boldsymbol{y})$ and substitute the estimate in the eqs. (2). We show that, approximating the unknown density by finite mixture of normal components, we obtain remarkable similarity between the resulting prediction formula and the functional description of the mixture-of-experts architecture. The involved parameters directly follow from the underlying mixture and therefore the optimization method reduces to standard estimation of normal mixtures by means of EM algorithm. In this sense the present approach can be viewed as a modification of probabilistic neural networks based on distribution mixtures [4] and using information preserving transforms [5]. [1]

2 Prediction Based on Normal Mixtures

We denote $\boldsymbol{z} = (\boldsymbol{x}^T, \boldsymbol{y}^T)^T$ the compound $(N+K)$-dimensional column vector and assume that the unknown probability density function $P(\boldsymbol{z})$ can be approximated by a normal mixture

$$P(\boldsymbol{z}) = \sum_{m \in \mathcal{M}} w_m F(\boldsymbol{z}|\mu_m, \Sigma_m), \quad \boldsymbol{z} \in R^{(N+K)}, \tag{3}$$

where $F(\boldsymbol{z}|\mu_m, \Sigma_m)$ are normal densities with the means μ_m and covariance matrices Σ_m, $m \in \mathcal{M}$, $\mathcal{M} = \{1, 2, \ldots, M\}$.

[1] For the information preserving property and its information theoretic characterization we refer to Vajda [10].

Considering the following partition of μ_m and Σ_m in accordance with the component vectors $\boldsymbol{x}$ and $\boldsymbol{y}$

$$\mu_m = \begin{pmatrix} \boldsymbol{c}_m \\ \boldsymbol{d}_m \end{pmatrix}, \quad \Sigma_m = \begin{pmatrix} A_m & V_m^T \\ V_m & B_m \end{pmatrix}, \tag{4}$$

we can easily verify the the well known formula for the marginal density

$$P(\boldsymbol{x}) = \sum_{m \in \mathcal{M}} w_m G(\boldsymbol{x}|\boldsymbol{c}_m, A_m), \tag{5}$$

$$G(\boldsymbol{x}|\boldsymbol{c}_m, A_m) = \frac{1}{\sqrt{((2\pi)^N \det A_m)}} \exp\{-\frac{1}{2}(\boldsymbol{x} - \boldsymbol{c}_m)^T A_m^{-1}(\boldsymbol{x} - \boldsymbol{c}_m)\}$$

and for the conditional probability density

$$P(\boldsymbol{y}|\boldsymbol{x}) = \sum_{m \in \mathcal{M}} \gamma_m(\boldsymbol{x}) H(\boldsymbol{y}|\boldsymbol{u}_m, U_m), \tag{6}$$

$$H(\boldsymbol{y}|\boldsymbol{u}_m, U_m) = \frac{1}{\sqrt{((2\pi)^K \det U_m)}} \exp\{-\frac{1}{2}(\boldsymbol{y} - \boldsymbol{u}_m)^T U_m^{-1}(\boldsymbol{y} - \boldsymbol{u}_m)\},$$

whereby $\boldsymbol{u}_m$ and U_m denote means and covariance matrices respectively

$$\boldsymbol{u}_m = \boldsymbol{d}_m + V_m A_m^{-1}(\boldsymbol{x} - \boldsymbol{c}_m), \quad U_m = B_m - V_m A_m^{-1} V_m^T, \tag{7}$$

and $\gamma_m(\boldsymbol{x})$ are conditional weights

$$\gamma_m(\boldsymbol{x}) = \frac{w_m G(\boldsymbol{x}|\boldsymbol{c}_m, A_m)}{\sum_{j \in \mathcal{M}} w_j G(\boldsymbol{x}|\boldsymbol{c}_j, A_j)}. \tag{8}$$

Making substitution in the prediction formula (1) we obtain

$$\hat{\boldsymbol{y}}(\boldsymbol{x}) = \sum_{m \in \mathcal{M}} \gamma_m(\boldsymbol{x}) \int \boldsymbol{y} H(\boldsymbol{y}|\boldsymbol{u}_m, U_m) d\boldsymbol{y} =$$

$$= \sum_{m \in \mathcal{M}} \gamma_m(\boldsymbol{x}) [\boldsymbol{d}_m + V_m A_m^{-1}(\boldsymbol{x} - \boldsymbol{c}_m)]. \tag{9}$$

The last eqs.(8),(9) are similar to that arising in the mixture-of-experts architecture (cf. [8], p.705, [6], p.185, [13], [14]). The parenthesized linear expression corresponds to the locally optimal output of the m-th expert unit and is it weighted by the expression $\gamma_m(\boldsymbol{x})$ produced by gating network. In terms of the original heuristic idea the "soft" weights $\gamma_m(\boldsymbol{x})$ divide the input space into "soft" hyperellipsoids to simplify the local regression tasks.

The parameters of the prediction formula (9) directly follow from the estimated normal mixture (3) while other optimiztion methods usually solve (weighted) least squares problem for each expert unit separately. For this reason, in practical problems, the quality of results may be different because of the different criteria.

Let us note further that, by grouping the components of the mixture $P(\boldsymbol{y}|\boldsymbol{x})$, we obtain prediction formula which corresponds to the hierarchical mixture-of-experts architecture. Indeed, considering a partition of the index set $\mathcal{M}$

$$\mathcal{M} = \bigcup_{j\in\mathcal{J}} \mathcal{M}_j, \quad \mathcal{J} = \{1, 2, \dots, J\} \tag{10}$$

we may define the first- and second level weights $g_{jm}(\boldsymbol{x})$ and $g_j(\boldsymbol{x})$

$$g_{jm}(\boldsymbol{x}) = \frac{\gamma_m(\boldsymbol{x})}{g_j(\boldsymbol{x})}, \quad g_j(\boldsymbol{x}) = \sum_{m\in\mathcal{M}_j} \gamma_m(\boldsymbol{x}), \quad j \in \mathcal{J} \tag{11}$$

and the prediction formula corresponding to a hierarchical mixture of experts

$$\hat{\boldsymbol{y}}(\boldsymbol{x}) = \sum_{j\in\mathcal{J}} g_j(\boldsymbol{x}) \sum_{m\in\mathcal{M}_j} g_{jm}(\boldsymbol{x})[\boldsymbol{d}_m + V_m A_m^{-1}(\boldsymbol{x} - \boldsymbol{c}_m)] \tag{12}$$

which is similar to that of Jordan et al. [7], p.185. Obviously, in our case, this formula is equivalent to that of the nonhierarchical mixture of experts (cf. (9)). Nevertheless, the hierarchical structure may become advantageous and nontrivial in case of separate solutions of the local regression subtasks.

3 Optimization of Parameters

Let us recall that all the parameters involved in the prediction formula (9) can be deduced from the mixture (3) and therefore the optimization procedure reduces to standard estimation of a normal mixture by means of EM algorithm (cf. [1], [3], [12]). A learning algorithm can be obtained by using a sequential modification of EM algorithm (cf. Titterington et al. [9], Chapter 6.)

Particularly let $\mathcal{S} = \{\boldsymbol{z}_1, \boldsymbol{z}_2, \dots\}$ be a finite set of $|\mathcal{S}|$ independent observations of the random vector $\mathbf{Z}$ identically distributed according to some unknown probability density function of the form (3). To estimate the unknown parameters we maximize log-likelihood function

$$L = \frac{1}{|\mathcal{S}|} \sum_{z\in\mathcal{S}} \log P(\boldsymbol{z}) = \frac{1}{|\mathcal{S}|} \sum_{z\in\mathcal{S}} \log \left[\sum_{m\in\mathcal{M}} w_m F(\boldsymbol{z}|\mu_m, \Sigma_m) \right] \tag{13}$$

by the two iterative steps of EM algorithm (cf. Grim (1996)):

E-step: $(m \in \mathcal{M},\ \boldsymbol{z} \in \mathcal{S},\ t = 0, 1, \dots)$

$$h^{(t)}(m|\boldsymbol{z}) = \frac{w_m^{(t)} F(\boldsymbol{z}|\mu_m^{(t)}, \Sigma_m^{(t)})}{\sum_{j\in\mathcal{M}} w_j^{(t)} F(\boldsymbol{z}|\mu_j^{(t)}, \Sigma_j^{(t)})}, \tag{14}$$

M-step: $(m \in \mathcal{M})$

$$w_m^{(t+1)} = \frac{1}{|\mathcal{S}|} \sum_{z\in\mathcal{S}} h^{(t)}(m|\boldsymbol{z}), \tag{15}$$

$$\mu_m^{(t+1)} = \frac{1}{|\mathcal{S}|w_m^{(t+1)}} \sum_{z\in\mathcal{S}} \boldsymbol{z} h^{(t)}(m|\boldsymbol{z}), \tag{16}$$

$$\Sigma_m^{(t+1)} = \frac{1}{|\mathcal{S}|w_m^{(t+1)}} \sum_{z\in\mathcal{S}} (\boldsymbol{z}-\mu_m^{(t+1)})(\boldsymbol{z}-\mu_m^{(t+1)})^T h^{(t)}(m|\boldsymbol{z}), \tag{17}$$

The iterative equations (14) - (17) generate a nondecreasing sequence of values $L^{(t)}, t = 1, 2, ...$ converging to a possibly local maximum of the likelihood function (13). A detailed discussion of the convergence properties can be found e.g. in Wu [11], Titterington et al. [9], Xu and Jordan [12] and others.

Let us note that, in case of general covariance matrices, it is alvays posible to obtain ill conditioned matrices $\Sigma_m^{(t+1)}$ during computation. Typically, this situation would occur whenever a deterministic functional relation is to be approximated by picewise linear regression. To avoid numerical problems we can remove the singular components or regularize the obtained matrices, e.g. by adding small positive constants to eigenvalues of matrices (to preserve the covariance structure of data). However, any such manipulation may violate the monotone behaviour of EM algorithm in the immediately following iteration.

4 Estimation of Initial Components of Mixtures

A difficult point in application of EM algorithm is to specify the number of components of the estimated mixture and to choose their initial parameters. This problem can be solved by optimizing the nonparametric kernel estimates of Parzen. Let us recall that all the good properties of Parzen estimates are guarranteed only asymptotically for infinitely large data sets and therefore, in any practical application, the smoothing has to be individually optimized.

In our case the kernel estimate can be interpreted as a uniformly weighted mixture of normal densities with equal covariance matrices Σ_0

$$P(\boldsymbol{z}) = \sum_{u\in\mathcal{S}} F(\boldsymbol{z}|\boldsymbol{u}, \Sigma_0) w(\boldsymbol{u}), \quad w(\boldsymbol{u}) = \frac{1}{|\mathcal{S}|}, \quad \boldsymbol{z} \in \mathbf{X} \tag{18}$$

whereby the components are positioned at data vectors $\boldsymbol{u} \in \mathcal{S}$. For this reason the component weights $w(\boldsymbol{u})$ and the common covariance matrix Σ_0 as a smoothing parameter can be optimized by means of the EM algorithm.

The log-likelihood function (18) is known to atain a “singular” maximum for the kernel density shrinking to delta function ($\det \Sigma_0 \to 0$). This undesirable property can simply be removed by the following modification of the criterion (18) (cf. Duin [2])

$$L = \sum_{z\in\mathcal{S}} \log[\sum_{u\in\mathcal{S}, u\neq z} F(\boldsymbol{z}|\boldsymbol{u}, \Sigma_0) w(\boldsymbol{u})], \quad w(\boldsymbol{u}) = \frac{1}{(|\mathcal{S}|-1)} \tag{19}$$

In order to maximize the criterion (19) we can modify the EM algorithm from Sec.4 again.

E-step: $(\boldsymbol{z}, \boldsymbol{u} \in \mathcal{S}, \boldsymbol{z} \neq \boldsymbol{u})$

$$q^{(t)}(\boldsymbol{u}|\boldsymbol{z}) = \frac{F(\boldsymbol{z}|\boldsymbol{u}, \Sigma_0^{(t)})w^{(t)}(\boldsymbol{u})}{\sum_{u \in \mathcal{S}, u \neq z} F(\boldsymbol{z}|\boldsymbol{u}, \Sigma_0^{(t)})w^{(t)}(\boldsymbol{u})}, \tag{20}$$

M-step: $(\ \boldsymbol{z} \in \mathcal{S})$

$$w^{(t+1)}(\boldsymbol{z}) = \frac{1}{|\mathcal{S}| - 1} \sum_{u \in \mathcal{S}, u \neq z} q^{(t)}(\boldsymbol{z}|\boldsymbol{u}), \tag{21}$$

$$\Sigma_0^{(t+1)} = \frac{1}{|\mathcal{S}|} \sum_{z \in \mathcal{S}} \sum_{u \in \mathcal{S}, u \neq z} (\boldsymbol{z} - \boldsymbol{u})(\boldsymbol{z} - \boldsymbol{u})^T q^{(t)}(\boldsymbol{u}|\boldsymbol{z}), \tag{22}$$

The method makes use of the empirical fact that many component weights $w(\boldsymbol{z})$ tend to vanish in the course of optimization and therefore the result can be used as an initial estimate of the approximating normal mixture.

Let us note that, applying the modified log-likelihood function (19), we cannot refer to some known properties of the m.-l. estimates. However, this fact is of minor importance because the EM procedure above is used only to compute initial estimates of a finite mixture.

5 Concluding remarks

In the last section the initial components of a mixture are estimated by means of a rather complex "top down" method. In practical situations a "bottom up" solution based on successive adding of components could be more efficient: For a given M we iterate the EM algorithm until reasonable convergence. Then we add a new sufficiently "flat" randomly placed component with a relatively high initial weight (e.g. $w_{M+1} = 0.5$). Continuing computation we obtain again a monotonely converging sequence $L^{(t)}$ for the new enlarged mixture. In this way there is a chance to find out the data regions not sufficiently covered by the previous set of component densities. The increased initial weight w_{M+1} helps the new component to "survive" in competition with the old well "fitted" components. Note that, again, any interrupt of regular iterations may disturb the monotone convergency. The adding of components may be continued until the weight of the new component is repeatedly suppressed despite the increased initial value.

Another difficulty arises if a large number of parameters involved in the multivariate mixture (3) is to be estimated from a limited data set $\mathcal{S}$. One way is to reduce the complexity of the mixture, e.g. by considering diagonal- or identical covariance matrices.

Acknowledgments

The present project has been supported by the Grant of the Academy of Sciences No. A2075703, by the Grant of the Ministry of Education No. VS 96063 and partially by the Complex research project of the Academy of Sciences of the Czech Republic No. K1075601.

Reference

[1] Dempster, A.P., Laird, N.M. and Rubin, D.B. 1977. Maximum likelihood from incomplete data via the EM algorithm. *J.Roy.Statist.Soc.* , Sec. B 39, pp.1–38.

[2] Duin, P.W. 1994. On the choice of smoothing parameters for Parzen estimates of probability density functions. *IEEE Trans. on Computers*, C-25, No.11, pp. 1175-1179.

[3] Grim, J. 1982. On numerical evaluation of maximum - likelihood estimates for finite mixtures of distributions. *Kybernetika*, Vol.18, No.3, pp.173–190.

[4] Grim, J. 1996. Maximum Likelihood Design of Layered Neural Networks. In *IEEE Proceedings of the 13th International Conference on Pattern Recognition*, pp. 85–89, IEEE Press.

[5] Grim, J. 1996a. Design of multilayer neural networks by information preserving transforms. In *Proc. 3rd Systems Science European Congress*, E. Pessa, M.B. Penna, A. Montesanto, eds., pp. 977–982, Edizzioni Kappa, Roma 1996.

[6] Jacobs, R.A., Jordan, M.I., Nowlan, S.J. and Hinton, G.E., 1991. Adaptive mixtures of local experts. *Neural Comp.*, Vol. 3. pp. 79 - 87.

[7] Jordan, M.I. and Jacobs, R.A. 1994. Hierarchical mixtures of experts and the EM algorithm. *Neural Comp.*, Vol. 6. pp. 181–214.

[8] Ramamurti, V. and Ghosh, J., 1996. Structural adaptation in mixtures of experts. In *IEEE Proceedings of the 13th International Conference on Pattern Recognition*, pp. 704–708, IEEE Press.

[9] Titterington, D.M., Smith, A.F.M. and Makov, U.E. 1985. *Statistical analysis of finite mixture distributions*, John Wiley & Sons: Chichester, Singapore, New York.

[10] Vajda, I. 1992. *Theory of Statistical Inference and Information.* Kluwer: Boston.

[11] Wu, C.F.J., 1983. On the convergence properties of the EM algorithm. *Ann. Statist.*, Vol. 11, pp. 95–103.

[12] Xu, L. and Jordan, M.I., 1996. On convergence properties of the EM algorithm for Gaussian mixtures. *Neural Comp.*, Vol. 8. pp. 129–151.

[13] Xu, L., Jordan, M.I. and Hinton, G.E. 1994. A modified gating network for the mixtures of experts architecture. In *Proc. WCNN'94*, San Diego, Vol. 2, pp. 405–410.

[14] Xu, L., Jordan, M.I. and Hinton, G.E. 1995. An alternative model for mixture of experts. In *Advances in Neural Information Processing Systems*, G. Tesauro, D.S. Touretzky and T.K. Leen eds., Vol. 7. pp. 633–640, MIT Press, 1995.