A FIXED-POINT ALGORITHM FOR FINDING THE OPTIMAL COVARIANCE MATRIX IN KERNEL DENSITY MODELING

Jose M. Leiva-Murillo and Antonio Artés-Rodríguez

Department of Signal Theory and Communications
Universidad Carlos III de Madrid
Avda. Universidad 30, 28911, Leganés (Madrid), Spain
{jose,antonio}@tsc.uc3m.es

ABSTRACT

In this paper, we apply the methodology of cross-validation Maximum Likelihood estimation to the problem of multivariate kernel density modeling. We provide a fixed point algorithm to find the covariance matrix for a Gaussian kernel according to this criterion. We show that the algorithm leads to accurate models in terms of entropy estimation and Parzen classification. By means of a set of experiments, we show that the method considerably improves the performance traditionally expected from Parzen classifiers. The accuracy obtained in entropy estimation suggests its usefulness in ICA and other information-theoretic signal processing techniques.

1. INTRODUCTION

Non-parametric density estimation, also known as *kernel* or Parzen density estimation, is a popular tool in statistics and signal processing [1]. Although non-parametric density estimation seems to refer to a free-parameter approach to the problem of density modeling, the fact is that the accuracy of the model does significantly depend on the bandwidth of the chosen kernel. Non-parametric models are constructed by means of *windows* or kernels centered on the data. The problem of searching the optimal width is commonly referred to as bandwidth selection or smoothing factor selection.

Bandwidth selection has been paid much attention by statisticians, especially in the one-dimensional case. An extensive compilation of methods can be found in [2]. In the multivariate case, each variable is often treated separately, and a model is independently built for each dimension.

Cross-validation techniques are based on a *leave-one-out* procedure, according to which the model evaluated on a training sample is built from the rest of samples [3]. A commonly used criterion for the evaluation of the model is the Integrated Square Error (ISE) $\int |\hat{p} - p|^2$, which is also used in plug-in and bootstrap methods [4].

Maximum Likelihood (ML), in a cross-validation setting, was first proposed in [5]. However, it has been criticized because of its slow rate of asymptotical convergence to the true density, when compared to ISE. In addition, some problems have been reported when applied to heavy-tailed distributions [3].

However, in the following we encourage the use of the ML criterion because of its suitability for entropy estimation and Par-zen classification. A Parzen classifier takes a ML or a maximum a-posteriori decision based on the models obtained for each class. Although not considered a state-of-the-art classifier, it is proven to provide an error tending to Bayes error, as soon as the models tend to the true densities [6]. We center our study on the Gaussian kernel, since it permits the adjustment of a covariance matrix that can take into account correlation among variables.

In Section 2 we describe the leave-one-out Maximum Likelihood (LOO_ML) procedure, and introduce two versions of a simple and efficient fixed-point algorithm to solve it, each one corresponding to different assumptions about the kernel covariance matrix. In Section 3, we analyze the performance of LOO_ML models in a set of entropy estimation and classification experiments, and compare it to other methods. The paper finishes with some remarks in Section 4

2. A FIXED-POINT ALGORITHM FOR LEAVE-ONE-OUT ML ESTIMATION

Let $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}^T$ be a set of samples from the D-dimensional random variable $\mathbf{x}$, according to which we construct our Parzen model $\hat{p}(\cdot)$. The density estimation at a given point is:

$$\hat{p}(\mathbf{x}) = \frac{1}{N} \sum_{j=1}^{N} G(\mathbf{x} - \mathbf{x}_j, \mathbf{C})$$

where $G(\cdot, \mathbf{C})$ is a centered Gaussian with a covariance matrix $\mathbf{C}$. Although other *kernel* functions can be used, the Gaussian is the most commonly employed. In this case, the bandwidth selection problem consists in finding the optimal $\mathbf{C}$.

This work is partly supported by the MCyT under grant TIC2003-02602.

In parametric modeling, $\mathbf{C}$ is given by the empirical covariance of the data. In semi-parametric models, the matrix can be obtained by the EM algorithm, which also makes use of the ML criterion. In non-parametric modeling, ML criterion can not be directly applied. In the case of one dimension, if the width of the window is to be optimized by a ML criterion, the solution converges to $\sigma = 0$, so that the likelihood becomes infinite. Thus, the model strongly overfits to data if there are not additional constraints.

In order to avoid this overfitting, the following model can be used when evaluated on samples $\mathbf{x}_i$ from the training set:

$$\hat{p}(\mathbf{x}_i) = \frac{1}{N-1} \sum_{\substack{j=1 \\ j \neq i}}^{N} G(\mathbf{x}_i - \mathbf{x}_j, \mathbf{C})$$

In the following, we refer to this as the leave-one-out (LOO) Parzen model. Its corresponding log-likelihood is given by:

$$\begin{aligned} \log L(\mathbf{X}, \mathbf{C}) &= \sum_i \log \hat{p}(\mathbf{x}_i) \\ &= \sum_i \log \left(\frac{1}{N-1} \sum_{j \neq i} G(\mathbf{x}_i - \mathbf{x}_j, \mathbf{C}) \right) \end{aligned} \tag{1}$$

assumed that the $\mathbf{x}_i$ are i.i.d. Now, the ML criterion can be applied on this model, and leads to a fixed-point algorithm that converges to a finite value. We present two versions of the algorithm: the first one assumes a spherical shape for $\mathbf{C}$, so that the width is the same in each dimension and there is not any local interaction. The second tackles the general case, with no constraints on $\mathbf{C}$.

2.1. Spherical covariance matrix

The use of a spherical covariance matrix is equivalent to modeling the different variables separately, with the same width. Thus, we assume the shape $\mathbf{C} = \sigma^2 \mathbf{I}_D$. In this case, the value of a Gaussian is given by:

$$\begin{aligned} G_{ij}(\sigma^2) &= G(\mathbf{x}_i - \mathbf{x}_j, \sigma^2) \\ &= (2\pi)^{-D/2} \sigma^{-D} \exp\left(- \frac{1}{2\sigma^2} \|\mathbf{x}_i - \mathbf{x}_j\|^2 \right) \end{aligned}$$

And the value of its derivative is:

$$\nabla_\sigma G_{ij}(\sigma^2) = \left(\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{\sigma^3} - \frac{D}{\sigma} \right) G_{ij}(\sigma^2)$$

The derivative of the log-likelihood, according to this, is:

$$\begin{aligned} \nabla_\sigma \log L(\mathbf{X}, \sigma^2) &= \sum_i \frac{1}{\hat{p}(\mathbf{x}_i)} \frac{1}{N-1} \sum_{j \neq i} \frac{\partial}{\partial \sigma} G_{ij}(\sigma^2) \\ &= \frac{1}{N-1} \sum_i \frac{1}{\hat{p}(\mathbf{x}_i)} \sum_{j \neq i} \quad \frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{\sigma^3} - \frac{D}{\sigma} \quad G_{ij}(\sigma^2) \end{aligned}$$

We search for the maximum of $\log L(\mathbf{X}, \sigma^2)$, so that its derivative is null. Then we have:

$$\begin{aligned} \sum_i \frac{1}{\hat{p}(\mathbf{x}_i)} \sum_{j \neq i} \frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{\sigma^3} G_{ij}(\sigma^2) &= \sum_i \frac{1}{\hat{p}(\mathbf{x}_i)} \frac{D}{\sigma} \sum_{j \neq i} G_{ij}(\sigma^2) \\ &= \frac{N(N-1)D}{\sigma} \end{aligned}$$

This leads to the following fixed-point algorithm, which is obtained by isolating the σ^2:

$$\sigma_{l+1}^2 = \frac{1}{N(N-1)D} \sum_i \frac{1}{\hat{p}_l(\mathbf{x}_i)} \sum_{j \neq i} \|\mathbf{x}_i - \mathbf{x}_j\|^2 G_{ij}(\sigma_l^2) \tag{2}$$

where $\hat{p}_l$ denotes the Parzen estimation in iteration l (i.e. the one that uses σ_l^2).

2.2. Full covariance matrix

The general expression for a Gaussian kernel with an arbitrary matrix $\mathbf{C}$ is:

$$G_{ij}(\mathbf{C}) = |2\pi \mathbf{C}|^{-1/2} \exp\left(-\frac{1}{2} (\mathbf{x}_i - \mathbf{x}_j)^T \mathbf{C}^{-1} (\mathbf{x}_i - \mathbf{x}_j) \right)$$

Its derivative is:

$$\nabla_{\mathbf{C}} G_{ij}(\mathbf{C}) = \frac{1}{2} \left(\mathbf{C}^{-1} (\mathbf{x}_i - \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)^T - \mathbf{I} \right) \mathbf{C}^{-1} G_{ij}(\mathbf{C})$$

As in the spherical case, we make the derivative of the log-likelihood from Eq. 1 null, which leads to the fixed-point algorithm:

$$\mathbf{C}_{l+1} = \frac{1}{N(N-1)} \sum_i \frac{1}{\hat{p}_l(\mathbf{x}_i)} \sum_{j \neq i} (\mathbf{x}_i - \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)^T G_{ij}(\mathbf{C}_l) \tag{3}$$

A full covariance matrix leads to more flexible models, so that it yields higher likelihood values. However, it suffers from overfitting more intensively than in the spherical case, because of the higher number of parameters (the elements of $\mathbf{C}$) involved.

3. EXPERIMENTS

Here, first we use a set of examples to show the performance of the LOO_ML algorithm of Eqs. 2 and 3 when applied to entropy estimation. Secondly, we provide a set of classification experiments to compare the performance of the method to other classifiers'.

3.1. Entropy Estimation

Here, the validity of LOO_ML for entropy estimation is explored by its application to synthetic Gaussian and Uniform multidimensional distributions. The estimation of the entropy is given by:

$$\hat{h}(\mathbf{x}) = -\frac{1}{N}\sum_i \log \hat{p}(\mathbf{x}_i)$$

3.1.1. Multidimensional Gaussian

Our estimator has been applied to two multidimensional Gaussian distributions with covariance matrixes $\begin{pmatrix} 2 & 0.5 \\ 0.5 & 1 \end{pmatrix}$ and $\begin{pmatrix} 3 & 0.7 & 0.2 \\ 0.7 & 2 & 0.5 \\ 0.2 & 0.5 & 1 \end{pmatrix}$. In Table 1, the value of the entropies estimated by the two versions of our method are compared to the one given by another bandwidth selector, and an entropy estimator that is not based on Parzen. The bandwidth selector is a multivariate generalization of Scott's rule. It chooses $\mathbf{C} = N^{\frac{-2}{D+4}}\mathbf{\Sigma}$, where $\mathbf{\Sigma}$ is the empirical covariance of $\mathbf{X}$. This estimation is optimal in terms of ISE for a Gaussian distribution [7]. The other method for entropy estimation is due to Kozachenko-Leonenko, and it is based on a nearest-neighbour procedure [8]. In this case, full LOO_ML provides the best estimations, although the one by Scott's rule is very close. This fact proves the validity of Scott's rule even from a ML point of view.

Table 1. Entropy estimation of Gaussian distributions. 500 samples have been used

D	Sph. $\hat{h}$	Full $\hat{h}$	Scott	K. L.	Real h
2	3.197	3.189	3.191	4.142	3.118
3	5.135	5.107	5.115	6.251	5.043

3.1.2. Multidimensional Uniform

Here, we apply the spherical LOO_ML to the modeling of data generated by a uniform distribution in 1, 2, 5 and 10 dimensions, and three different sample sizes. The width has been tuned around the width obtained by LOO_ML. We have plotted the result of estimating the entropy from a Parzen model with the different widths considered, and stressed the point found by LOO_ML. The results are shown in Fig. 1. In all cases LOO_ML reaches the minimum, so that it obtains the value that is closest to the true value.

Two interesting conclusions from the curves are:

- The accuracy in the entropy estimation is higher as the sample size becomes larger.
- The behavior of the optimal width with respect to N and D does not seem to disagree with the asymptotic factor traditionally modeled by $N^{\frac{-2}{D+4}}$ [7].

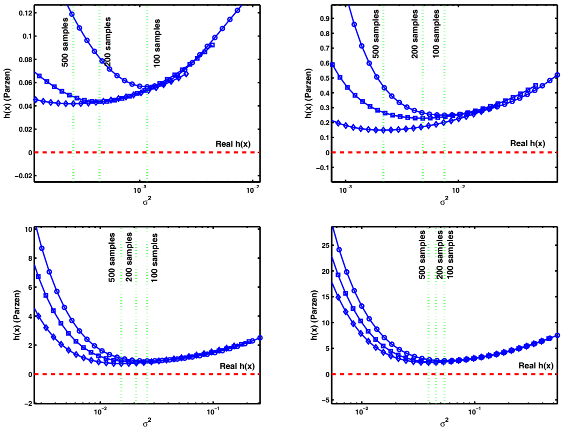

Fig. 1. Window width and estimated entropy from 100 (circles), 200 (squares) and 500 (diamonds) samples from multidimensional uniform distributions of 1, 2, 5 and 10 dimensions in [0,1]. The vertical dashed lines mark the LOO_ML bandwidth

3.2. Parzen Classification

Here we display the result of applying the LOO_ML models to Parzen classification. A Parzen classifier assigns an incoming sample $\mathbf{x}$ the class k given by a ML criterion on the models built for each class:

$$\hat{y} = \arg\max_k \hat{p}_k(\mathbf{x}) \tag{4}$$

where $\hat{p}_k(\cdot)$ is the Parzen model of class k. We have carried out two experiments. First, we tune the window width of a spherical Parzen model in a toy example, to show the suitability of LOO_ML in terms of classification error. After that, we display the performance of a Parzen classifier when applied to some public datasets.

3.2.1. Synthetic Data

We have generated data according to a five-dimensional spherical Gaussian distribution $N(0, I_5)$. The labels have been generated by the function $\mathrm{sign}(x_4 x_5)$. The performance of the Parzen classifier has been plotted for the optimal bandwidth, together with the performance obtained by a set of values around it. The results in Fig. 2 show that the width found by LOO_ML is the optimal one in terms of classification performance.

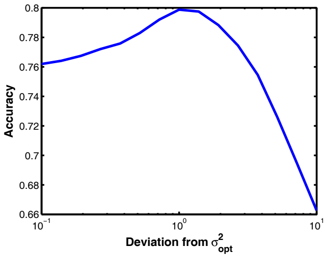

Fig. 2. Average (100 trials) accuracy of the Parzen classifier as a function of the window width in the synthetic problem. The datasets have 200 samples

3.2.2. Real Data

Although the Parzen classifier is a simple and sub-optimal one, here we compare the performance of the two versions of LOO_ML with:

- Another Parzen classifier where the Scott's rule has been used for bandwidth-selection, as described above.
- Two discriminative-type classifiers: the K-nearest-neighbour (KNN) [9] and the Support Vector Machine (SVM) [10]. Note that these methods do not provide probabilistic values at their output.

The experiments have been carried out on the Pima, Wine and Landsat datasets from the UCI public repository [1]. The performance of Parzen classification is tested for the two versions of LOO_ML. Each training dataset is used for both training the model and the classifier, which are then used to classify the testing set.

The results are displayed in Table 2. Although a full shape is expected to provide the best result, it does not hold in the case of Landsat. This dataset consists of hyperspectral images from satellites, according to which the kind of soil must be determined. The fact that the spherical approach performs better in this case is due to the number of free parameters involved, which is far higher in the full approach than in the spherical one. The number of parameters is 1 in the spherical case, and $\frac{D(D+1)}{2}$ in the full case. Thus, in such high-dimensional problems, the full approach suffers from a stronger overfitting, which suggests the use of the spherical approach instead.

Although a purely discriminative method like SVM obtains a better performance, Parzen provides a-posteriori probabilities. There are many situations in which a probabilistic interpretation of the result is required. An advantage of Parzen classifiers is that it provides such interpretation.

Table 2. Classification accuracy (in percentage) on public datasets. N_c is the number of classes

Dataset	N_c/D	Sph. P.	Full P.	Scott	KNN	SVM
Pima	2 / 8	71.22	75.00	73.05	73.18	76.47
Wine	3 / 13	78.10	99.44	99.44	76.97	100
Landsat	6 / 36	89.45	86.10	84.85	90.60	90.90

4. DISCUSSION AND FUTURE WORK

We have shown the suitability of a Maximum Likelihood choice of the covariance matrix in Gaussian Parzen models, by means of a fixed-point algorithm based on a leave-one-out procedure. Although the algorithm has converged in all the cases explored so far, a future effort must be carried out to prove its convergence. The accuracy of the method when estimating multivariate entropy suggests a promising application in the ICA framework and other information-theoretic approaches, as for example feature extraction. The good performance obtained by Parzen classifiers trained with LOO_ML makes this method appropriate for classification problems in which probabilistic interpretations are needed.

5. REFERENCES

[1] D. Erdogmus and J. C. Principe, "Generalized information potential criterion for adaptive system training," *IEEE Trans. on Neural Networks*, vol. 13, no. 5, pp. 1035–1044, 2002.

[2] L. Devroye, "Universal smoothing factor selection in density estimation: Theory and practice," *Test*, vol. 6, no. 2, pp. 223–320, 1997.

[3] A. Cuevas R. Cao and W. González, "A comparative study of several smoothing methods in density estimation," *Computational Statistics and Data Analysis*, vol. 17, pp. 153–176, 1994.

[4] M. P. Wand and M. C. Jones, *Kernel Smoothing*, Chapman & Hall, London, 1995.

[5] R. P. Duin, "On the choice of smoothing parameters for parzen estimators of probability density functions," *IEEE Transactions on Computers*, vol. 25, no. 11, pp. 1175–1179, 1976.

[6] K. Fukunaga, *Introduction to Statistical Pattern Recognition*, New York: Academic Press, 1990.

[7] David W. Scott, *Multivariate Density Estimation*, Wiley-Interscience, 1992.

[8] L. F. Kozachenko and N. N. Leonenko, "Sample estimate of entropy of a random vector," *Problems of Information Transmition*, vol. 23, no. 9, pp. 95–101, 1987.

[9] L. Devroye, L. Györfi, and G. Lugosi, *A Probabilistic Theory of Pattern Recognition*, Springer, New York, 1996.

[10] B. Schölkopf and A. Smola, *Learning with Kernels*, MIT Press, Cambridge, MA, 2002.

[1] http://www.ics.uci.edu/ mlearn/MLRepository.html