Algorithms for Maximum-Likelihood Bandwidth Selection in Kernel Density Estimators

José M. Leiva-Murillo*

Antonio Artés-Rodríguez

Dept. Signal Theory and Communication, Universidad Carlos III de Madrid,
Av. Universidad, 30, Leganés 28911 (Madrid), Spain.
Ph. +34 916248450, Fax +34 916248749

Abstract

In machine learning and statistics, kernel density estimators are rarely used on multivariate data due to the difficulty of finding an appropriate kernel bandwidth to overcome overfitting. However, the recent advances on information-theoretic learning have revived the interest on these models. With this motivation, in this paper we revisit the classical statistical problem of data-driven bandwidth selection by cross-validation maximum likelihood for Gaussian kernels. We find a solution to the optimization problem under both the spherical and the general case where a full covariance matrix is considered for the kernel. The fixed-point algorithms proposed in this paper obtain the maximum likelihood bandwidth in few iterations, without performing an exhaustive bandwidth search, which is unfeasible in the multivariate case. The convergence of the methods proposed is proved. A set of classification experiments are performed to prove the usefulness of the obtained models in pattern recognition.

Key words: Kernel Density Estimation, Multivariate Density Modeling,

Email addresses: jose@tsc.uc3m.es (José M. Leiva-Murillo), antonio@tsc.uc3m.es (Antonio Artés-Rodríguez)

1. Introduction

Kernel Density Estimators (KDE) is a family of probability density function (PDF) models that have been intensively studied by the statistics community. Despite of being considered non-parametric, a *bandwidth* parameter determines the scale of the kernel function and therefore the performance of these models. Univariate KDEs have been object of much interest, whilst multivariate models have been studied mainly in low dimensional cases. For this reason, the interest of the pattern recognition and machine learning community in KDEs has been rather limited. For example, in the information-theoretic learning literature, KDEs with a Gaussian kernel are often used to estimate entropy or mutual information [18], [24], [15]. In most cases the bandwidth of the kernel is treated as a hyperparameter chosen by cross validation or a heuristic criterion.

The objective of this paper is to provide a method for the design of the kernel bandwidth based on the maximum likelihood (ML) criterion, commonly applied in parametric and semi-parametric PDF modeling. Although the idea of maximizing leave-one-out ML has been previously proposed in the literature, an exhaustive search was needed to find the optimal bandwidth. We avoid this inconvenience by providing fixed-point algorithms that converge to the ML bandwidth even when it is characterized by a number of parameters. The proof of the convergence in the general case is given by the properties of the expectation-maximization (EM) algorithm [2]. Because the spherical model is a particular case of the unconstrained case, its convergence

is automatically proved as well. However, we follow a different procedure for the convergence analysis of the spherical bandwidth, which provides us with important information about the range of values to which the bandwidth is guaranteed to belong.

In the next Section, we study the problem of PDF estimation by KDE models, and propose two novel and efficient algorithms for bandwidth selection for both the spherical and the unconstrained Gaussian kernels. In Section 3 we analyze the performance of KDEs with the obtained bandwidths on both synthetic and benchmark real data. The paper finishes in Section 4 with some conclusions about the work presented.

2. Bandwidth Selection for Kernel Density Estimators

A Kernel Density Estimator (KDE) is a non-parametric PDF model that consists of a linear combination of kernel functions centered on the data (see, for example, [9])

$$\hat{p}_{\boldsymbol{\theta}}(\mathbf{x}) = \frac{1}{N}\sum_{i=1}^{N} k(\mathbf{x} - \mathbf{x}_i|\boldsymbol{\theta}) \tag{1}$$

where $\mathbf{x} \in \mathbb{R}^D$ and $k(\mathbf{x}|\boldsymbol{\theta})$ is the kernel function with a given set of parameters $\boldsymbol{\theta}$. The kernel must be a unitary function, i.e.: $\int k(\mathbf{x}|\boldsymbol{\theta})d\mathbf{x} = 1$. The success of these models in pattern recognition and machine learning is due to several reasons. First, a-priori assumptions on the distribution of the data need not be made. Secondly, the model does not need to be trained, as it only relies on the samples. Finally, it is easy to carry out transformations on the data, such as $\mathbf{z} = \mathbf{f}(\mathbf{x})$, and estimate the PDF in the $\mathbf{z}$-space, even if the transformation is not invertible, because $\hat{p}(\mathbf{z})$ is a KDE built from the transformed sample set $\{\mathbf{z}_i = \mathbf{f}(\mathbf{x}_i)\}$.

Although the KDEs are commonly considered as non-parametric models, the kernel function has an adjustable bandwidth defined by $\boldsymbol{\theta}$ that determines the accuracy of the model, so that it can be treated as a parameter to be optimized.

The bandwidth selection problem has been a matter of intensive research for the last 30 years in the statistics community. See [25] for an exhaustive review of criteria for univariate data. The most extended criteria in bandwidth estimation are the integrated square error (ISE), the mean ISE (MISE) and the asymptotic MISE (AMISE) [1]. In the one-dimensional case, optimizing these criteria with respect to the bandwidth is not problematic as it involves a global search on one variable which is computationally feasible. This is the main reason why multivariate bandwidth selection has been addressed only in very low dimensional spaces [6]. According to the review in [25], the main categories of methods for bandwidth selection are: rule-of-thumb methods, cross-validation methods and *plug-in* methods. When applied to criteria based on asymptotic square error, these methods suffer from a serious drawback: they need to estimate not only the true density, but also its first and second derivatives for a study based on Taylor expansion. As the true density is unknown in general, the estimated one can be used instead by means of a *pilot* bandwidth [26]. Although this can be acceptable in the univariate case, in the multivariate case the inaccuracy may become dramatic. In the case of rule-of-thumb methods, data are assumed Gaussian, which leads to close-form expressions. Otherwise, the multivariate case has been addressed

[1] Criteria based on L_1-norm have been also proposed [4].

in the literature either by using a spherical kernel or by considering a scaled version of the covariance matrix of data as the kernel bandwidth [22], [20], being the scale factor the only parameter to adjust.

The Bayesian framework provides a natural way of obtaining the bandwidth by assigning it a proper prior and computing the posterior of the bandwidth given the data. However, two difficulties arise when working with KDEs. First, unlike in parametric modeling, the parameters of the bandwidth do not index any family of densities, since the centers of the kernels are needed. For this reason, a loss different than likelihood is considered [3], [7]. Secondly, the choice of the prior distribution determines the final choice of the bandwidth. The use of Monte Carlo Markov Chains (MCMC) has also been proposed [27].

In this paper, we aim at applying generative modeling to real-world machine learning and pattern recognition problems, in which the dimension is higher than usually considered by the statistics community for kernel density estimation. We base our work on the maximum likelihood (ML) criterion. The risk of overfitting is inherent to ML because the complexity of the model in not considered. However, this risk can be alleviated by performing cross-validation, since the model is evaluated in points that are different from the ones used for the KDE model. The leave-one-out (LOO) is the extreme case in which $N-1$ samples build a model evaluated on the point left

$$\hat{p}_{\boldsymbol{\theta}}(\mathbf{x}_i) = \frac{1}{N-1} \sum_{\substack{j=1 \\ j \neq i}}^{N} G(\mathbf{x}_i - \mathbf{x}_j | \boldsymbol{\theta}) \tag{2}$$

where we make explicit the use of a Gaussian kernel. Note that a ML solution is equivalent to minimum entropy when the entropy is estimated as $\hat{h}(\mathbf{X}) =$

$-\frac{1}{N}\sum_i \log \hat{p}(\mathbf{x}_i)$. It has been proven in the literature that if $\hat{p}(\mathbf{x})$ is a KDE, the entropy is overestimated [1]. Hence, a minimum entropy criterion for bandwidth choice leads to a model that provides an estimation of the entropy which is closest to the true value than any other performed with a KDE.

This procedure was first proposed in [5] and later studied by other authors [22], [12]. The study of the multivariate case has been constrained to problems in which the dimension is low, because in absence of a closed optimization procedure, an exhaustive search of the optimal bandwidth is unfeasible if the dimension of $\boldsymbol{\theta}$ is high. In this section, we present a procedure for the bandwidth selection problem that overcomes these difficulties.

We will consider two different degrees of complexity for the covariance matrix of the Gaussian kernel. In the most simple case, we assume a spherical shape so that $\mathbf{C} = \sigma^2 \mathbf{I}_D$. In this case, $\boldsymbol{\theta}$ consists of just one parameter. In the general or unconstrained case, no constraints are imposed to $\mathbf{C}$ further than its positive semi-definiteness; in this case, the number of elements in $\boldsymbol{\theta}$ is $(D-1)D/2$. In the following, we separately describe the bandwidth selection procedure for the spherical and the general case, providing two versions of what we call the Maximum Likelihood Leave-One-Out (ML-LOO) algorithm. The spherical model is a particular case of the unconstrained model, so that the convergence in the former would follow from the convergence of the latter. Although both cases can be studied through their connection to the Expectation-Maximization algorithm, we choose to separately prove the convergence in both cases. Specifically, in the spherical case we base our proof in the general properties of fixed-point algorithms. This way, interesting conclusions arise about the range of values where the optimal bandwidth

can be found. A preliminary version of this work was presented in [13].

2.1. The spherical case

The value of a kernel function centered in point $\mathbf{x}_i$ and evaluated in $\mathbf{x}_j$ is, for the spherical Gaussian case,

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

The derivative with respect to σ is given by

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

Let us consider now the log-likelihood of the data according to this model under the iid assumption: $\log\mathrm{L}(\mathbf{X}|\sigma^2) = \sum_i \log \hat{p}_\sigma(\mathbf{x}_i)$, where $\hat{p}_\sigma(\mathbf{x}_i)$ is LOO estimated as in (2). The derivative of the LOO log-likelihood, is given by

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

We now search for the maximum value of $\log L(\mathbf{X}|\sigma^2)$, and so the point that makes the derivative null. We have

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

The second equality has been obtained by the fact that, by definition, $\sum_{j\neq i} G_{ij} = (N-1)\hat{p}_\sigma(\mathbf{x}_i)$. By isolating the σ^2 we obtain the following fixed-point rule

$$\sigma^2_{n+1} = \frac{1}{N(N-1)D} \sum_i \frac{1}{\hat{p}_{\sigma_n}(\mathbf{x}_i)} \sum_{j\neq i} \|\mathbf{x}_i - \mathbf{x}_j\|^2 G_{ij}(\sigma^2_n). \tag{3}$$

We prove the convergence of the algorithm in (3) by the following Theorem:

Theorem 1. *There is a fixed point in the interval* $\left(\frac{\overline{d^2_{NN}}}{D}, \frac{\overline{d^2}}{D}\right)$, *being* $\overline{d^2_{NN}}$ *the mean quadratic distance to the nearest neighbor and* $\overline{d^2}$ *the mean distance among data points, so that* $\left(\frac{\overline{d^2_{NN}}}{D} \leq \frac{\overline{d^2}}{D}\right)$. *Besides, the fixed point is unique and the algorithm converges to it in the mentioned interval if the following condition holds*

$$\frac{1}{4\sigma^4 N(N-1)^2 D} \sum_i \frac{1}{(\sum_{l\neq i} \exp(-\frac{d^2_{il}}{2\sigma^2}))^2} \times \sum_{j\neq i} \sum_{k\neq i,j} (d^2_{ij} - d^2_{ik})^2 \exp(-\frac{d^2_{ij} + d^2_{ik}}{2\sigma^2}) < 1 \tag{4}$$

where $d^2_{ij} = \|\mathbf{x}_i - \mathbf{x}_j\|^2$.

Proof. Let $\sigma^2 = g(\sigma^2)$ be the function in (3), whose fixed point is to be obtained. The proof of the fixed point existence is based on the search of an interval (a, b) such that $a < g(\sigma^2) < b$ if $\sigma^2 \in (a, b)$, which is called a *contractive map* [8].

In order to demonstrate that the interval $\left(\frac{\overline{d^2_{NN}}}{D}, \frac{\overline{d^2}}{D}\right)$ holds the property stated by the Theorem, we need to prove these three conditions:

1. $\lim_{\sigma^2 \to 0} g(\sigma^2) = \frac{\overline{d^2_{NN}}}{D}$.

2. $\lim_{\sigma^2 \to \infty} g(\sigma^2) = \overline{d^2}$.

3. $g(\sigma^2)$ is monotonic in the interval.

These conditions are graphically summarized in Fig. 1. This way we are guaranteed that the interval is a contractive map, and there is at least one crossing point between the function $g(\sigma^2)$ and the line $g(\sigma^2) = \sigma^2$.

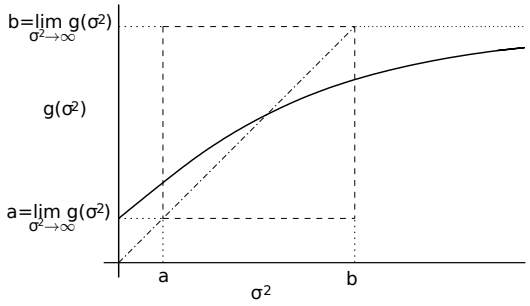

Figure 1: Contractive Map $\sigma^2 = g(\sigma^2)$

To prove the first point, we rewrite (3) as

$$g(\sigma^2) = \frac{1}{ND} \sum_i \sum_{j \neq i} d_{ij}^2 \frac{1}{1 + \sum_{k \neq i,j} \exp(\frac{d_{ij}^2 - d_{ik}^2}{2\sigma^2})}. \tag{5}$$

The limit at 0 is given by

$$\lim_{\sigma^2 \to 0} f(\sigma^2) = \frac{1}{ND} \sum_i \min_{j \neq i} d_{ij}^2 = \frac{\overline{d_{NN}^2}}{D}$$

because the elements in the denominator of (5) are null (the exponentials tend to infinite) in exception of the cases in which $d_{ij}^2 < d_{ik}^2$, $\forall k \neq j$, i.e. $\mathbf{x}_j$ is the nearest neighbor of $\mathbf{x}_i$. The first condition is then proven.

To prove the second condition, we take the limit

$$\lim_{\sigma^2\to\infty} g(\sigma^2) = \lim_{\sigma^2\to\infty} \frac{1}{ND}\sum_i\sum_{j\neq i} d_{ij}^2 \frac{\exp(-\frac{d_{ij}^2}{2\sigma^2})}{\sum_{k\neq i}\exp(-\frac{d_{ik}^2}{2\sigma^2})}$$
$$= \frac{1}{ND}\sum_i\sum_{j\neq i} d_{ij}^2 \frac{1}{N-1} = \frac{\overline{d^2}}{D}.$$

Then, the second condition is proven. The average distance can be efficiently estimated from the covariance matrix of the data as $\overline{d^2} = 2\,\mathrm{tr}\{\mathbf{\Sigma}_x\}$, by simple properties of linear algebra.

To demonstrate the last condition, we compute the derivative of $g'(\sigma^2)$ and check out that it is positive

$$\begin{aligned}\frac{dg(\sigma^2)}{d\sigma^2} &= \frac{1}{2\sigma^4 ND}\sum_i\sum_{j\neq i} d_{ij}^2 \frac{\sum_k (d_{ij}^2 - d_{ik}^2)\exp(-\frac{d_{ij}^2+d_{ik}^2}{2\sigma^2})}{(\sum_{l\neq i}\exp(-\frac{d_{il}^2}{2\sigma^2}))^2} \\ &= \frac{1}{2\sigma^4 ND}\sum_i \frac{1}{(\sum_{l\neq i}\exp(-\frac{d_{il}^2}{2\sigma^2}))^2} \times \sum_{j\neq i}\sum_{k\neq i} d_{ij}^2(d_{ij}^2 - d_{ik}^2)\exp(-\frac{d_{ij}^2+d_{ik}^2}{2\sigma^2}) \\ &= \frac{1}{4\sigma^4 N(N-1)^2 D}\sum_i \frac{1}{(\sum_{l\neq i}\exp(-\frac{d_{il}^2}{2\sigma^2}))^2} \\ &\times \sum_{j\neq i}\sum_{k\neq i,j}(d_{ij}^2 - d_{ik}^2)^2\exp(-\frac{d_{ij}^2+d_{ik}^2}{2\sigma^2}) \geq 0.\end{aligned} \tag{6}$$

In the last step, the terms with $j = k$ are removed, since they are null, and the rest have been regrouped.

The existence of the fixed point is then proved. To demonstrate the convergence of the algorithm in such interval, we need to check out the condition $|g'(\sigma^2)| < 1$ [8]. In that case, we are guaranteed that only a crossing point between $g(\sigma^2)$ and the line $g(\sigma^2) = \sigma^2$ exists. The convergence condition (4) means that the value of the derivative obtained in (6) is lesser than 1. □

Unfortunately, the computational complexity of obtaining the bound (4) is $O(N^3)$. In practice we have found the first condition as the most critical. The reason is that for quantized data it is possible that $\overline{d_{NN}^2}/D \approx 0$. It is easy to see from Fig. 1 that, in that case, the fixed-point location dangerously approaches to zero. In that case, the data should be treated as discrete instead, and a distribution model based on the histogram can be a better choice than a KDE.

The advantage of using the fixed-point iteration instead of an alternative optimization procedure is double. First, the practitioner does not have to worry about the optimization procedure for the log-likelihood, which is already implicit in the fixed-point algorithm. Secondly, Theorem 1 provides us with a range of values where the optimal bandwidth is guaranteed to be found, which is of both theoretical and practical interest.

2.2. The unconstrained case

The general expression for a Gaussian kernel is

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

and its derivative w.r.t. $\mathbf{C}$

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

As in the previous cases, we take the derivative of the log-likelihood and make it equal to zero

$$\sum_i \frac{1}{\hat{p}_{\mathbf{c}}(\mathbf{x}_i)} \frac{1}{N-1} \sum_{j\neq i} \frac{1}{2}\mathbf{C}^{-1}(\mathbf{x}_i - \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)^T\mathbf{C}^{-1}G_{ij} = \sum_i \frac{1}{\hat{p}_{\mathbf{c}}(\mathbf{x}_i)} \frac{1}{N-1} \sum_{j\neq i} \frac{1}{2}\mathbf{C}^{-1}G_{ij}.$$

By multiplying both members by $\mathbf{C}$, we obtain

$$\sum_i \frac{1}{\hat{p}_{\mathbf{c}}(\mathbf{x}_i)} \sum_{j \neq i} (\mathbf{x}_i - \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)^T G_{ij} = \mathbf{C} \sum_i \frac{1}{\hat{p}_{\mathbf{c}}(\mathbf{x}_i)} \sum_{j \neq i} G_{ij}.$$

After some simplifications as in the spherical case, we have

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

leading to the following fixed-point rule

$$\mathbf{C}_{n+1} = \frac{1}{N(N-1)} \sum_i \frac{1}{\hat{p}_{\mathbf{c}_n}(\mathbf{x}_i)} \sum_{j \neq i} (\mathbf{x}_i - \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)^T G_{ij}(\mathbf{C}_n). \tag{7}$$

The expression in (7) suggests a relationship with the expectation maximization (EM) result for Gaussian mixture models (GMM). A GMM is a PDF estimator given by the expression $p(\mathbf{x}) = \sum_{k=1}^{K} \alpha_k G(x|\boldsymbol{\mu}_k, \mathbf{C}_k)$. The weights of the K components of the mixture are given by the α_k, and each Gaussian is characterized by its mean vector $\boldsymbol{\mu}_k$ and its covariance matrix $\mathbf{C}_k$. The solution provided by the EM algorithm consists of an iterative procedure where the values at step t are estimated from their corresponding values at step $t-1$. To do so, a matrix of auxiliary variables r_{ki} is used, each of which stands for the likelihood that the i-th sample has been generated by the k-th component of the mixture, with $\sum_k r_{ki} = 1$. The EM solution establishes the following updating rule for the covariance matrix at step t

$$\mathbf{C}_k^t = \sum_k \sum_i r_{ki}^t \frac{(\mathbf{x}_i - \boldsymbol{\mu}_k^t)(\mathbf{x}_i - \boldsymbol{\mu}_k^t)^T}{N} \tag{8}$$

where the r_{ki}^t and $\boldsymbol{\mu}_k^t$ are also iteratively updated. Note that our KDE model can be considered as a special case of GMM where there are as many mixtures as samples ($K = N$) with the same weights ($\alpha_k = 1/N$) and fixed

mean vectors: $\boldsymbol{\mu}_k = \mathbf{x}_k$. The covariance matrix is the same for each of the components. Besides, the leave-one-out scheme imposes the values $r_{ki} = 0$ if $k = i$ and $r_{ki} = \dfrac{1}{N-1}$ if $k \neq i$. Then, the updating rule in (8) is equal to the one given by the iteration (7).

We conclude that the optimization for the unconstrained case can be set up as a particular case of the EM algorithm. The EM algorithm is known to monotonically increase the likelihood, so that its convergence to a local minimum has been proven in the literature [14]. According to this, the algorithm given in (7) is immediately shown to converge.

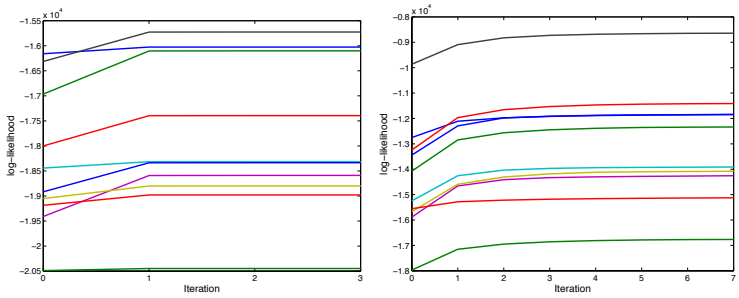

Figure 2: Log-likelihood values across iteration for the 10 different classes in Optdigits dataset. Left: spherical model; right: full (unconstrained) model.

In Fig. 2 we illustrate the convergence speed of both versions of ML-LOO on real data from the Optdigits dataset from the UCI repository. When the bandwidth is initialized with Scott's rule [20], it only takes one iteration to get the optimal bandwidth when the spherical model is considered. In the unconstrained case, the likelihood values are higher than in the spherical one as expected, and the convergence is slower, but we note that no relevant increase in the log-likelihood is obtained beyond the 5th or 6th iteration.

A graphical example of the performance of both algorithms in (3) and (7) is displayed in Fig. 3 for a small set of synthetic data. Note that, in spite of using few samples to build the model, the proposed algorithm reaches solutions that are smooth yet descriptive. Besides, the unconstrained kernel is more flexible than the spherical one, in the sense that it is more accurate for distributions in which the covariance matrix of data is far from spherical.

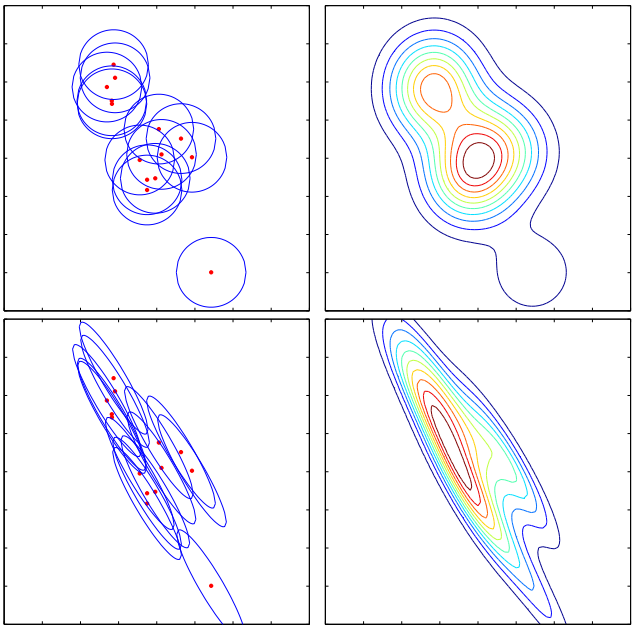

Figure 3: Examples of models obtained for the spherical (top) and the general (bottom) cases. Left: individual kernels, plotted at their -3dB level w.r.t. the mode; right: KDE models as the average of kernels on the left.

3. Experimental Results

Probability density estimation belongs to the family of unsupervised learning methods. This means that it is not possible to compare different methods

unless the experiment has been designed on synthetic data whose distribution is known. The estimated density is then compared to the true one according to some criterion. Obviously, the criterion chosen for comparison benefits the methods that have made use that same criterion for bandwidth selection.

As an alternative, we propose to apply the bandwidth selection criteria to models that are used in supervised learning tasks. In particular, we consider a set of classification or pattern recognition problems. In classification, the general problem is to estimate the relationship between an input $\mathbf{x}$ and an output y from a dataset $\{\mathbf{x}_i, y_i\}$, $i = 1, \ldots, L$, $\mathbf{x}_i \in \mathbb{R}^D$, $y \in \{1, 2, \ldots, N_c\}$. When the density of the data is modeled, classification is said to be *generative*. In particular, when KDEs are used for density modeling, the procedure is referred to as Parzen classification.

We first describe the concept of Parzen classification, and then show the performance of the classifier on both synthetic and real data.

3.1. Parzen Classification

A Parzen classifier assigns a sample $\mathbf{x}$ a label according to a criterion based either on the likelihood $p(\mathbf{x}|y)$ or the posterior $p(y|\mathbf{x})$, using the density models estimated for each class. In the former case, the decision is taken according to

$$\hat{y} = \arg\max_l \hat{p}(\mathbf{x}|c_l) \tag{9}$$

where $\hat{p}(\mathbf{x}|c_l)$ is a Parzen or KDE model built from the training data belonging to class l [9]. In the literature, generative models have rarely been used in real world problems due to their high dimensionality, unless some

factorization is imposed on the variables to decompose the problem in low dimensional tasks [16].

When choosing between a spherical or a full Gaussian kernel to perform the classification rule (9), a tradeoff between advantages and weaknesses of each option must be considered. In the spherical case, the model can be inaccurate if the training data are distributed according to a strongly non-spherical pattern. If the components of $\mathbf{x}$ are very different in their range or variance, taking the same bandwidth for each dimension may be not reasonable. On the other hand, if we use a full (or unconstrained) model we must be aware that the number of parameters involved can be very high and lead to overfitting.

In order to overcome the disadvantages of both the spherical and the full kernels, we propose a hybrid scheme that combines the advantages from both approaches. The idea behind this procedure is to first whitening the data to make the spherical approach appropriate per-class. Then, we evaluate the value of the probability density back in the original feature space by means of the linear transformation property of PDFs: $p(\mathbf{A}^T\mathbf{x}) = |\mathbf{A}|^{-1}p(\mathbf{x})$. Let $\mathbf{B}_l$ be the whitening matrix for each class, so that the components of $\mathbf{B}_l^T\mathbf{x}$ are uncorrelated and normalized. The classification algorithm according to this procedure is described by the algorithm in Fig. 4.

If the empirical covariance matrix of any of the classes is singular, we can consider a non square matrix $\mathbf{B}_l$, with as many rows as non negative eigenvalues of the covariance matrix. In order to compute the determinant of $\mathbf{B}_l$ as required by the algorithm, we propose to consider the heuristic estimation $|\mathbf{B}| \approx \sqrt{|\mathbf{B}^T\mathbf{B}|}$.

Obtain whitening matrices $\{\mathbf{B}_l\}$, $l = 1, \ldots, N_c$

Obtain spherical models $\hat{p}(\tilde{\mathbf{x}}|c_l)$ for $\tilde{\mathbf{x}} = \mathbf{B}_l^T \mathbf{x}_i$.

For each sample $\mathbf{x}_i$ in the test dataset:

    Project the sample according to $\tilde{\mathbf{x}}_i = \mathbf{B}_l^T \mathbf{x}_i$

    Obtain $\hat{p}(\tilde{\mathbf{x}}_i|c_l)$

    Obtain $\hat{p}(\mathbf{x}_i|c_l) = |\mathbf{B}_l| \hat{p}(\tilde{\mathbf{x}}_i|c_l)$

    Decide the class according to: $\hat{y}_i = \max_l \hat{p}(\mathbf{x}_i|c_l)$

End

Figure 4: Hybrid method for Parzen classification.

Given that the three methods described have a different complexity, an objective criterion is needed to choose a priori among the models. In addition to the classification performance, we have evaluated the leave-one-out likelihood, i.e. the criterion according to which the models have been build. Also, we have evaluated the Bayesian Information Criterion (BIC) and the Akaike Information Criterion (AIC) to penalize the likelihood according to the complexity of the model. Although originally developed for models of the exponential family [21], BIC is usually applied to mixture models too [23]. The BIC suggests choosing the model for which the following expression is maximum

$$BIC_j = \log L(\mathbf{X}) - \frac{K_j}{2} \log N$$

where j indexes the model, K_j is the number of parameters of the model and N is the number of observations. In our case, the number of parameters is $K_{sph} = K_{hyb} = 1$ in the spherical and hybrid models, and $K_{full} = D(D-1)/2$

for the unconstrained case, which is the number of elements in the bandwidth matrix.

AIC, on the other hand, chooses the model based on a different penalization of the likelihood. In particular, the *constrained* AIC or AIC^c accounts for finite sample sizes

$$AIC_j^c = \log L(\mathbf{X}) - K_j - \frac{2K_j(K_j+1)}{N-K_j-1}$$

In addition to the classification results on data under different dimension reduction degrees and the use of a spherical kernel, in this section we explore the performance of KDE-based classifiers with different kernel degrees. First, we use a simple yet illustrative synthetic experiment to measure the dependency of the classification performance with respect to the bandwidth. Secondly, we provide a set of results on real data, using the spherical and full models provided by ML-LOO, as well as the hybrid one, as described above.

3.2. Experiment on Synthetic Data

We illustrate the importance of an appropriate bandwidth choice by a simple synthetic binary classification experiment. Data have been generated following a 5 dimensional normal distribution. The labels have been generated according to the classification function $y = \text{sign}(x_4 x_5)$. By means of 1000 training data, the spherical bandwidth has been obtained from these data according to the rule (3). The $\hat{\sigma}$ obtained is shared by both classes. In Fig. 5, the classification performance is shown on a test dataset with 500 samples in an interval of bandwidth values ranging from $\hat{\sigma}^2/10$ y $10\hat{\sigma}^2$. Note

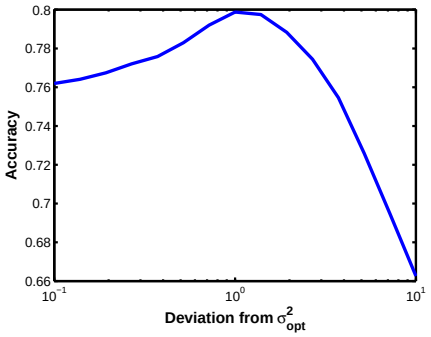

Figure 5: Classification performance averaged over 100 trials as a function of the bandwidth used.

that the best performance is reached when the bandwidth used is equal to $\hat{\sigma}$. The performance is shown to be very sensitive to the bandwidth choice.

3.3. Experiments on Real Data

Dataset	N_c	D	Size
Landsat	6	36	6435
Optdigits	10	64 (40)	5620
Letter	26	16	20000
Isolet	26	617 (40)	7797
Waveform	3	21	800
Segmentation	7	19 (8)	2310

Table 1: Characteristics of the public Datasets. The numbers between brackets express the dimension after dimension reduction by principal component analysis.

We have tested several Parzen classifiers, with spherical, unconstrained and hybrid kernels, on the same public datasets described in Table 1. We

have made sure that the convergence conditions hold for all the datasets, including the more complex and computationally costly condition (6). We show the results for both the whitened and the raw data in the cases in which this is possible: those in which the covariance matrix of each per-class dataset is non-singular. Otherwise, only the result on whitened data is reported.

We have taken ten independent samples for each experiment, where 75% of the data are used for training, and the rest are used for test.

	Generative				Discriminative	
Dataset	Spherical KDE	Full KDE	Hybrid KDE	Scott	KNN	Reference
Landsat (raw)	90.11± 0.74	86.34± 0.90 $^{ML}_{PL}$	84.87± 0.73	84.34± 0.75	**90.61± 0.56**	97.58± 0.84
Landsat (whitened)	67.82± 0.44	**85.85± 0.42** $^{ML}_{PL}$	84.41± 0.43	84.00± 0.48	66.44± 0.84	94.77± 1.05
Optdigits (whitened)	97.95± 0.33	98.26 ± 0.26 $^{ML}_{PL}$	**98.80± 0.23**	97.83± 0.23	97.39± 0.39	99.98± 0.05
Letter (raw)	95.58± 0.28	93.50 ± 0.60 $^{ML}_{PL}$	95.34± 0.39	95.35± 0.29	**95.65± 0.24**	99.92 ± 0.11
Letter (whitened)	94.71± 0.25	94.30± 0.38 $^{ML}_{PL}$	**95.29± 0.34**	95.25± 0.28	94.80 ± 0.25	99.93± 0.11
Isolet (whitened)	86.52 ± 0.50	90.86± 0.72 $^{ML}_{PL}$	**91.90± 0.67**	91.69± 0.69	85.65± 0.55	99.74± 0.22
Waveform (raw)	**78.50± 2.95** $_{PL}$	74.45± 3.45 ^{ML}	**78.50± 3.20**	77.95 ± 3.72	75.55± 2.77	86.77± 1.33
Waveform (whitened)	62.45± 3.07 $_{PL}$	74.20± 3.54 ^{ML}	**78.90± 1.95**	78.40± 1.97	57.25± 4.03	89.65± 3.89
Segmentation (whitened)	87.69± 1.19	93.29± 1.1613 $^{ML}_{PL}$	92.46 ± 0.93	90.85± 1.05	**95.67 ± 0.54**	98.71 ± 0.58

Table 2: Classification performance on public datasets. The best models according to log-likelihood are marked with the "ML" label; those chosen by BIC or AIC are marked with "PL".

We have compared these results to the ones obtained by the generalization of Scott's rule proposed in [11], which establishes the following covariance matrix for the kernel: $\mathbf{C} = N^{\frac{-2}{D+4}}\mathbf{\Sigma}_x$, where $\mathbf{\Sigma}_x$ is the empirical covariance matrix of $\mathbf{x}$. We have also compared the generative KDE-based methods with the discriminative K-nearest neighbors (KNN, with $K = 1$). We also provide a reference value that can be considered as the highest achievable performance on each dataset. These reference values have been obtained by a state-of-the-art support vector machine (SVM) with radial basis function

as a kernel [19], where the hyperparameters C and σ have been chosen by a grid-search and a 5-fold cross-validation on the training set.

The classification results are shown in Table 2. Boldfaced numbers mark the highest performance on each dataset. Also, the model providing the highest likelihood value is marked with "ML" label, and the one chosen by the BIC or AIC^c criteria (which have agreed in all cases) is marked with a "PL" (from *penalized likelihood*) label. According to the results, when using generative modeling there is a version of ML-LOO that performs better than Scott's rule for all cases, although the difference is not statistically significant in some cases. Furthermore, better classification performance is obtained by generative methods when compared to a discriminative method as KNN in most cases. Note that, when performance is similar, generative models have the advantage that they provide values of the joint distribution, which allows both sampling from the distribution and obtaining likelihood values, as opposed to discriminative methods like KNN or SVM.

Regarding the comparison between the bandwidth of the spherical KDE-based classifier and the RBF width of the SVM, the latter is larger than the former in all the cases evaluated (details are not given for lack of space). We can find the reason in the inverse proportionality between number of samples and optimal bandwidth observed in both KDEs and SVMs. Therefore, because the SVM provides a sparse solution (only a subset of the data is involved), its optimal bandwidth becomes larger.

The results are not concluding regarding whether the spherical or the full model provides better results. However, the fact that the hybrid approach provides the best result in the majority of the experiments reveals

that the method incorporates advantages from both the spherical and the unconstrained methods, as described in Section 3.1. Unfortunately, neither BIC nor AIC seem to give a conclusive clue about the kind of method that provides the best performance - they agree only in two of the nine cases. From this, we conclude that the ML-LOO procedure, in its different versions, provides competitive classification performance although neither likelihood, BIC nor AIC provide an useful criterion to choose a priori among the different options.

Surprisingly, in some cases the performance of spherical Parzen is worse on the whitened data than in the original datasets, while the performance with full, hybrid and Scott bandwidth is almost unchanged. This suggests that i) whitening the data only helps in cases in which is unavoidable, i.e. when the covariance matrix of the data is singular and a dimension reduction is needed; and ii) whitening the whole dataset does not imply that the data belonging to each class are whitened. The bad results of the spherical KDE on the whitened versions of Landsat and Waveform are two examples of that.

4. Conclusions

We have provided a methodology for bandwidth selection in kernel density estimators. Although the spherical Gaussian is the most widespread kernel in these models, we have also described the procedure for a full Gaussian kernel. The application of a maximum likelihood leave-one-out criterion has led to a set of fixed-point algorithms that prevent us from carrying out an exhaustive search of the optimal bandwidth parameter. The conditions for the convergence of the algorithms proposed have also been established. In

particular, for the spherical approach the range of bandwidth values to which the optimum belongs has been provided. The convergence in that range is guaranteed.

Generative classification methods are usually in disadvantage compared to discriminative methods for which the classification performance is the objective to maximize. However, we have explored the performance of a Parzen classifier and obtained a performance that is less sensitive to the *curse of the dimensionality* than traditionally attributed to KDE-based classifiers. Moreover, higher performance than discriminative KNN is obtained in most of the cases explored. Although both spherical and full versions of ML-LOO converge in few iterations, we have also observed that the alternative and computationally cheap bandwidth criterion based on Scott's rule provides reasonable results which can justify its use in large datasets. Finally, although some clues have been given about the conditions in which the different versions ML-LOO perform better in classification, a criterion based on likelihood penalized by the number of parameters does not give successful clues about the best model for classification. This is explained by the fact that penalized likelihood evaluates a *full picture*, while the classification accuracy is mainly determined by the accuracy of the models only close to the classification boundaries. The hybrid method seems to incorporate positive elements from both the spherical and the unconstrained approaches, and therefore it is the methods that have obtained the best result in a larger number of experiments.

References

[1] Ahmad, I., Lin, P., 1976. A nonparametric estimation of the entropy for absolutely continuous distributions. IEEE Transactions on Information Theory 22 (3), 372–375.

[2] Bilmes, J. A., 1997. A gentle tutorial of the EM algorithm and its application to parameter estimation for gaussian mixture and hidden markov models. Tech. Rep. tr-97-021, International Computer Science Institute, Berkeley, California, USA.

[3] de Lima, M. S., Atuncar, G. S., 2011. A bayesian method estimate optimal bandwidth multivariate kernel estimator. Journal of Nonparametric Statistics 23 (1), 137–148.

[4] Devroye, L., Lugosi, G., Dec. 1996. A universally acceptable smoothing factor for kernel density estimates (24), 2499–2512.

[5] Duin, R., 1976. On the choice of smoothing parameters for Parzen estimators of probability density functions. IEEE Transactions on Computers 25 (11).

[6] Duong, T., Hazelton, M. L., 2005. Cross-validation bandwidth matrices for multivariate kernel density estimation. Scandinavian Journal of Statistics 32, 485–506.

[7] Filippone, M., Sanguinetti, G., 2011. Approximate inference of the bandwidth in multivariate kernel density estimation. Computational Statistics & Data Analysis 55 (12), 3104–3122.

[8] Fletcher, R., 1995. Practical Methods of Optimization (2nd Edition). John Wiley & Sons, New York.

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

[10] Goldberger, J., Roweis, S., Hinton, G., Salakhutdinov, R., 2005. Neighbourhood components analysis. In: Advances in Neural Information Processing Systems, NIPS 17. Vancouver, Canada, pp. 513–520.

[11] Haerdle, W., Mueller, M., Sperlich, S., Werwatz, A., 2004. Nonparametric and Semiparametric Models. Springer, New York.

[12] Hall, P., 1982. Cross-validation in density estimation. Biometrika 69 (2), 383–390.

[13] Leiva-Murillo, J., Artés-Rodríguez, A., 2008. Algorithms for gaussian bandwidth selection in kernel density estimators. In: Advances in Neural Information Processing System, NIPS, Optimization Workshop. Wistler, Canada.

[14] McLachlan, G., 1997. The EM algorithm and extensions. John Wiley & Sons, New York.

[15] Peltonen, J., Kaski, S., 2005. Discriminative components of data. IEEE Transactions on Neural Networks 16 (1), 68–83.

[16] Pérez, A., Larrańaga, P., Inza, I., 2009. Bayesian classifiers based on kernel density estimation: Flexible classifiers. International Journal of Approximate Reasoning (50), 341–362.

[17] Platt, J. C., Platt, J. C., 1999. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In: Advances in Large Margin Classifiers. MIT Press, pp. 61–74.

[18] Principe, J., Xu, D., Fischer, J., 2000. Information-Theoretic Learning. Vol. 1 of Unsupervised Adaptive Filtering. John Wiley & Sons, New York.

[19] Scholkopf, B., Smola, A., 2002. Learning with Kernels. The MIT Press, Cambridge, MA.

[20] Scott, D., 1992. Multivariate Density Estimation. John Wiley & Sons, New York.

[21] Schwarz, G., 1978. Estimating the dimension of a model. Ann. Stat. 14, 461–464.

[22] Silverman, B. W., 1986. Density Estimation for Statistics and Data Analysis. Chapman & Hall, Londres.

[23] Steele, R. J., Raftery, A. E. Performance of Bayesian Model Selection Criteria for Gaussian Mixture Models. Technical Report no. 559. Department of Statistics, University of Washington.

[24] Torkkola, K., 2003. Feature extraction by non-parametric mutual information maximization. Journal on Machine Learning Research 3, 1415–1438.

[25] Turlach, B. A., 1993. Bandwidth selection in kernel density estimation: A review. In: CORE and Institut de Statistique. pp. 1–33.

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

[27] Zhang, X., King, M. L., Hyndman, R. J., 2006. A bayesian approach to bandwidth selection for multivariate kernel density estimation. Computational Statistics & Data Analysis 50 (11), 3009–3031.