A family of non-parametric density estimation algorithms

E. G. Tabak* and C. V. Turner †

December 29, 2011

Abstract

A new methodology for density estimation is proposed. The methodology, which builds on the one developed in [15], normalizes the data points through the composition of simple maps. The parameters of each map are determined through the maximization of a local quadratic approximation to the log-likelihood. Various candidates for the elementary maps of each step are proposed; criteria for choosing one includes robustness, computational simplicity and good behavior in high-dimensional settings. A good choice is that of localized radial expansions, which depend on a single parameter: all the complexity of arbitrary, possibly convoluted probability densities can be built through the composition of such simple maps.

1 Introduction

A central problem in the analysis of data is density estimation: given a set of independent observations $x_j,\ j = 1, \ldots, m$, estimate its underlying probability distribution. This article is concerned with the case in which x is a continuous, possibly multidimensional variable, typically in R^n, and its distribution is specified by a probability density $\rho(x)$. Among the many uses of density estimation are its application to classification, clustering and dimensional reduction, as well as more field-specific applications such as medical diagnosis, option pricing and weather prediction [2, 7, 13].

Parametric density estimation is often based on maximal likelihood: a family of candidate densities is proposed, $\rho(x;\beta)$, where β denotes parameters from an admissible set A. Then these parameters are chosen so as to

*E. G. Tabak, Courant Institute, New York University, tabak@cims.nyu.edu
†C. V. Turner, FaMAF, U. N. Córdoba, turner@famaf.unc.edu.ar

maximize the log-likelihood L of the available observations:

$$\beta = \arg\max_{\beta \in A} L = \sum_{j=1}^{m} \log\left(\rho(x_j;\beta)\right). \tag{1}$$

A typical example is a family $\rho(x;\beta)$ of Gaussian mixtures, with β including free parameters in the means and covariance matrices of the individual Gaussians and their weights in the mixture. Parametric density estimation is a practical tool of wide applicability, yet it suffers from the arbitrariness in the choice of the parametric family and number of parameters involved. Ideally, the form of the density function would emerge from the data, not from arbitrary a priori choices, unless these are guided by a deeper knowledge of the processes originating the probability distribution under study.

The simplest methodology for non-parametric density estimation is the histogram [16], whereby space is divided into regular bins, and the estimated density within each bin is assigned a uniform value, proportional to the number of observations that fall within. Histogram estimates are not smooth and suffer greatly from the curse of dimensionality. A smoother version, first developed in [12] and [11], uses a sum of kernel functions centered at each observation, with a bandwidth adapted to the level of resolution desired. Particular kernels have been devised to handle properties of the target distributions; for instance, when these are known to have support only in the positive half-line, Gamma kernels have been proposed as a substitute for their symmetric Gaussian counterpart [4].

In non-parametric estimation, one must be careful to to over-resolve the density, for which one needs to calibrate the smoothing parameters to the data [10]. The most universal methodology for this is cross-validation [6], in which the available data are partitioned into subsets, used alternatively for *training* and out-of-sample *testing* of the estimation procedure. A related procedure is the bootstrap [8], which creates training and testing populations by drawing samples with replacement from the data.

An alternative methodology for non-parametric density estimation was developed in [15], based on normalizing flows in feature-space (A normalization procedure in small-dimensional sections of the data also forms the basis of exploratory projection pursuit [5], a methodology originally developed for the visualization of high-dimensional data.) Normalizing the data x_j is finding a map $y(x)$ such that the $y_j = y(x_j)$ have a prescribed distribution $\mu(y)$, for which we shall adopt here the isotropic Gaussian

$$\mu(y) = \mathcal{N}(0, I_n) = \frac{1}{(2\pi)^{\frac{n}{2}}} e^{-\frac{\|y\|^2}{2}} \tag{2}$$

If such map is known, then the probability density $\rho(x)$ underlying the original data is given by

$$\rho(x) = J^y(x)\mu(y(x)), \tag{3}$$

where $J^y(x)$ is the Jacobian of the map $y(x)$ evaluated at the point x. In view of (3), density estimation can be rephrased as the search for a normalizing map.

There is more than semantics to this rephrasing: normalizing the data is often a goal per se. It allows us, for instance, to compare observations from different datasets, to define robust metrics in phase-space, and to use standard statistical tools, often applicable only to normal distributions. More important for us here, however, is that it leads to the development of a novel family of density-estimation techniques.

1.1 Density estimation through normalizing flows

The simplest idea for finding a normalizing map $y(x)$ is to propose a parametric family, $y = y_\beta(x)$, and maximize the log-likelihood of the data, combining (1), (2) and (3) into

$$\beta = \arg\max_{\beta\in A} L = \sum_{j=1}^{m}\left[\log\left(J^{y_\beta}(x_j)\right) - \frac{\|y_\beta(x)\|^2}{2}\right], \tag{4}$$

where we have omitted from the log-likelihood L the β-independent term $-\frac{n}{2}\log(2\pi)$. In particular, if $y(x)$ is chosen among all linear functions of the form

$$y_\beta(x) = A(x-b) \quad \left(\beta = \left\{A \in R^{n\times n}, b \in R^n\right\}\right), \tag{5}$$

then the output of the maximization in (4) is

$$b = \bar{x}, \quad A = \Sigma^{-\frac{1}{2}}, \tag{6}$$

where $\bar{x} = \frac{1}{m}\sum_{j=1}^{m} x_j$ is the empirical mean and $\Sigma = \frac{1}{m}\sum_{j=1}^{m} x_j x_j^t$ the empirical covariance matrix of the data. In other words, a linear choice for $y_\beta(x)$ yields the standard normalization procedure of subtracting the mean and dividing by the square-root of the covariance matrix. In terms of density estimation, it yields the Gaussian

$$\rho(x) = \frac{1}{(2\pi)^{\frac{n}{2}}|\Sigma|^{\frac{1}{2}}} e^{-\frac{1}{2}(x-\bar{x})^t\Sigma^{-1}(x-\bar{x})}. \tag{7}$$

Yet this, as all parametric procedures, suffers from the extra-structure it imposes on the data, by assuming that it has an underlying probability density of a particular form.

One way to approach the algorithm proposed in [15] is to factor the map $y(x)$ into N parametric maps $\phi_{\beta_i}(z)$:

$$y_N(x) = \phi_{\beta_N} \circ \phi_{\beta_{N-1}} \circ \ldots \circ \phi_{\beta_1}(x), \tag{8}$$

since the composition of many simple maps can be made arbitrarily complex, thus overcoming the limitations of parametric maps. If this is considered as a function $y_\beta(x)$, depending on the indexed family of parameters $\beta = (\beta_1 \ldots \beta_N)$ on which to perform the maximization in (4), then we have just complicated matters without resolving any issue. Yet the following two realizations help us move forward:

- We can calculate the various β_i sequentially: first find β_1 using $y_1(x) = \phi_{\beta_1}(x)$ in (4), then β_2 using $y_2(x) = \phi_{\beta_2}(\phi_{\beta_1}(x))$, with β_1 fixed at the value found in the prior step, and so on. If the identity map is included in each elementary family for $\beta_i = 0$,

  $$\phi_0(z) = z,$$

  then each new step can only increase the value of the log-likelihood L, so even though we are not maximizing L over $\beta = (\beta_1 \ldots \beta_N)$, we are still ascending it through the sequence of maps.

- Switching perspective from density estimation to normalization, we can at each step i forget all prior steps, and just deal with the currently normalized states $z_j = y_{i-1}(x_j)$ of the observations as if these were the original ones. In order to be able to compute at the end the estimated density of the original variables x_j, we just need to update at each step the global Jacobian of the map, through

  $$J^{y_i}(x_j) \to J^{\phi_i}(z_j) J^{y_{i-1}}(x_j), \tag{9}$$

  i.e. by multiplying it by the Jacobian of the current elementary map. With this new perspective, all steps adopt the simple form in (4), with $\beta = \beta_i$ and each x_j replaced by the current z_j. This *duality* is the basis of our algorithm: rather than set out to estimate the density $\rho(x)$, we seek a normalizing map $y(x)$. This we factor into many elementary maps, with parameters determined through a local density estimation, in which (4) is applied not to x but to the current state $z(x)$ of the map.

Pushing this idea to the limit, we may think of a continuous flow $z = \phi_t(x)$ in an algorithmic time t, with velocity field

$$u(z) = \frac{\partial z}{\partial t} \tag{10}$$

driven by the variational gradient of the log-likelihood L. From this perspective, the observations x_j give rise to active Lagrangian markers $z_j(t)$, with $z_j(0) = x_j$, that move with the flow and guide it through their contribution to the –local in time– log-likelihood L. It was proved in [15] that, as the number of observations grows, $y(x) = \lim_{t\to\infty} z(x,t)$ converges to a normal distribution, and the density $\rho(x)$ estimated through (3) to the actual density of the data.

For the observations x_j, the active Lagrangian markers that guide the flow leading to the map $y(x)$, we know at the end their normalized values $y(x_j)$ and the corresponding estimated densities $\rho(x_j)$ from (3). Yet one is typically interested in evaluating the estimated density at other points x_i^g. These might be points on a regular grid –hence the "g" in x_i^g–, required to plot or manipulate $\rho(x)$. They can also represent events whose likelihood one would like to know, or points whose probability under various distributions is required for a classification problem. In order to evaluate the density at these extra points x_i^g, it is enough to add them as passive Lagrangian markers that move with the flow but do not influence it, since they are not included in the likelihood function.

1.2 The individual maps

The building blocks $\phi_i(x)$ proposed in [15] were simple one-dimensional maps, centered at a random point x_0, oriented in a random direction; they depended on three parameters β. These parameters were found by ascent of the log-likelihood, i.e. through

$$\beta = \nu \nabla_\beta L|_{\beta=0}, \tag{11}$$

with a learning rate ν given by

$$\nu = \frac{\epsilon}{\sqrt{\epsilon^2 + \|\nabla_\beta L\|^2}}, \tag{12}$$

and $\epsilon \ll 1$ prescribed. This simple formula for the learning rate guarantees that the size $\|\beta\|$ of all steps is bounded by ϵ and decreases near a maximum of L. It was proved in [15] that the composition of such one-dimensional maps suffices to guarantee convergence to arbitrary distributions $\rho(x)$, based on the fact that two distributions with the same marginals in all directions are necessarily identical. This procedure was further developed in [1] to address clustering and classification problems.

Yet the procedure just described suffers from some computational drawbacks:

- Exploring all directions through one-dimensional maps requires a number of steps that grows exponentially with the dimension of phase-space. In many applications, such as to microarray data, this dimension can be very large. Moreover, performing random rotations –i.e. orthogonal transformations– in high dimensions is costly.

- In order to have a smooth ascent process, the step-size ϵ needs to be small, hence requiring the algorithm to perform a large number of steps to reach convergence.

In this paper, we address both of these issues. On the one hand, we propose elementary transformations that do not deteriorate when the dimensionality of phase-space grows, the simplest and most effective of which is based on radial expansions. On the other, we exploit the fact that the elementary transformations have a very simple analytical form to go beyond straightforward gradient descent, and instead maximize in each step the local quadratic approximation to the log-likelihood in terms of the parameters β. This allows us to take much larger steps, and hence reduces significantly the total number of steps that the algorithm requires.

2 General methodological aspects

2.1 Center x_0 and length-scale α

All the elementary maps that we propose are of the form

$$y = x + \phi\left(\frac{x - x_0}{\alpha}\right),$$

centered at a random point x_0. The parameter α, measuring the length-scale of the map, has a value that depends on the selected node x_0: in areas with small probability, the length-scale must be large, not to over-fit the data. We start by choosing a number n_p of points that we would like to have within a ball or radius α around x_0. Then α is given by the expression

$$\alpha = (2\pi)^{\frac{1}{2}} \left(\Omega_n^{-1} \frac{n_p}{m}\right)^{\frac{1}{n}} e^{\frac{\|x_0\|^2}{2n}}, \tag{13}$$

which results from inverting the density of the target normal distribution. Here m is the total number of data-points, n the dimension of feature-space, and Ω_n the volume of the unit ball in R^n. The concept is illustrated in two dimensions in figure 1.

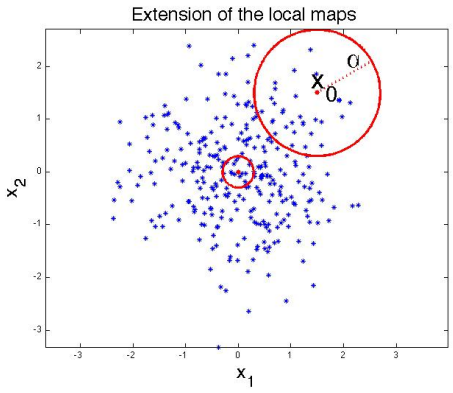

Figure 1: Dependence of the length scale α on the center x_0. In order not to over-resolve the estimation, the maps need to include a sufficient number of observations within their typical length scale. Thus, for maps centered in relatively unpopulated areas, the radius α must be larger than in areas with high probability density, so as to encompass a similar number of points. The two circles in the figure exemplify this: a point in the tail of the distribution is assigned a larger domain of influence than one in the core.

We can think of two candidate methodologies for selecting the point x_0: to pick it at random from the actual observations -at their current normalized state- or to sample the normal distribution to which $y(x)$ is converging. The former choice has the advantage of sampling the actual current density, not the estimated one; on the negative side, it never picks points away from the observations, so it may be ineffective at reducing over-estimated densities at points far from the observed set. The latter choice, on the other hand, will sample all points proportionally to their current estimated density, so it will detect and help correct points with over-estimated probability, yet it may fail to sample points in areas with under-estimated probability density, so these may never be corrected. We have implemented a balanced solution, whereby we randomly alternate between the two sampling methodologies just described.

2.2 Local ascent

In [15], we proposed picking the parameters β of each time-step by gradient ascent of the log-likelihood, through (11) and (12). Yet such procedure does not exploit to its full extent the simplicity of each elementary map. The explicit nature of these maps allows us to compute analytically not just the first but also the second derivatives of the log-likelihood L with respect to β.

With this is hand, we can take larger and more effective steps by maximizing the quadratic local approximation to L

$$L \approx L_0 + G\beta + \frac{1}{2}\beta' H\beta, \tag{14}$$

where

$$L_0 = L|_{\beta=0}, \quad G^j = \frac{\partial L}{\partial \beta_j}\Big|_{\beta=0} \quad \text{and} \quad H_i^j = \frac{\partial^2 L}{\partial \beta_i \partial \beta_j}\Big|_{\beta=0} \tag{15}$$

are the log-likelihood, its gradient and its Hessian matrix evaluated at $\beta = 0$. Maximizing (14) yields

$$\beta = -H^{-1}G. \tag{16}$$

A little care is required though not to take steps that are too long, incompatible with the local quadratic approximation. Firstly, if the Hessian matrix H were not negative-definite, the quadratic form (14) would have no maximum, and regular gradient descent would be called upon. Luckily, this is never the case with the maps that we propose, whose Hessian H is always negative definite. It may happen though that L is locally quite flat, leading to a large value of $\|\beta\|$ from (16). To avoid pushing the quadratic approximation too far from its domain of validity, we limit the step-size to a maximum learning rate ϵ. We adopt, if $\|H^{-1}G\| > \epsilon$, the capped step

$$\beta = -\epsilon \frac{H^{-1}G}{\|H^{-1}G\|}. \tag{17}$$

2.3 Preconditioning

It may be convenient to do some simple initial transformations that map the data points toward a normal distribution in the bulk. Reasons for this range from the general to the specific:

- For data points far from the origin, the gradient of their likelihood under a normal distribution could reach machine zero, at which point the algorithm will lack any guidance as to how to move them to improve their likelihood.

- Movements in the bulk may require a coarse resolution, as measured by the length scale α, at odds with the finer one needed for a more detailed resolution of the probability density.

- We may have some a priori knowledge of a family of distributions that should capture much of the data's variability. Using this to do first a simple parametric estimation may save much computational time.

- In some cases, we might be interested in how much the actual distribution differs from a conventional one, such as the log-norm for investment returns. Then we can first do a fit to the conventional distribution, and then quantify the extent and nature of the subsequent maps.

This first set of maps can be thought of as a preconditioning step of the algorithm, which only differs from the subsequent steps in the form of the proposed maps or in the scale adopted. Two preconditioning steps that we include by default in the algorithm are substracting the mean of the data,

$$x \to x - \mu, \quad \mu = \frac{1}{m}\sum_{j=1}^{m} x_j, \tag{18}$$

and dividing by the average standard deviation:

$$x \to \frac{1}{\sigma}x, \quad \sigma = \sqrt{\frac{1}{mn}\sum_{j=1}^{m} \|x_j\|^2}, \tag{19}$$

with corresponding initial estimation

$$\rho_0(x) = \left(2\pi\sigma^2\right)^{-\frac{n}{2}} e^{-\frac{\|x-\mu\|^2}{2\sigma^2}}. \tag{20}$$

Proposing a general Gaussian as in (7) is not generally advisable in high dimensions, unless the sample size m is big enough to allow for a robust estimation of the covariance matrix Σ.

Another preconditioning candidate generally applicable consists of carrying out a few steps of the regular core procedure, but with coarser resolution, i.e. with larger n_p in (13). More generally, we can have a value of n_p that decreases monotonically throughout the procedure, from an initial coarse value to the finest resolution desired or allowed by the data, thus blurring the boundary between preconditioning and the algorithm's core.

In specific cases, where a family of probability densities of specific form $\rho_0(x, \beta)$ is known or conjectured to provide a sensible fit of the data, and a map $y(x, \beta)$ is known such that

$$\rho_0(x, \beta) = J^y(x)\mu(y(x, \beta)), \tag{21}$$

then the preconditioning step should consist of a parametric fit of these parameters β followed by the map. The popular procedure of taking the log of a series of returns fits within this framework, where the conjecture $\rho_0(x)$ is a log-normal distribution.

Often $\rho(x)$ has bounded or semi-infinite support, which may be known even though ρ itself is not. For instance, some components of x may be known to be positive or, if x denotes geographical location, $\rho(x)$ may be known to vanish over seas or in other unpopulated areas [14]. When this is the case, it may be convenient to perform as preconditioning a first map that fills out all of space, such as

$$x \to \mathrm{erf}^{-1}\left(1 - 2e^{-x}\right) \tag{22}$$

for one-dimensional data with $x \geq 0$. The advantage of such preconditioning step goes beyond moving the data toward Gaussianity: it also guarantees that the estimated $\rho_{est}(x)$ will vanish outside the support of $\rho(x)$.

3 Elementary building blocks

In order to complete the description of the algorithm, we need to provide a form for the elementary maps of each computational step, the "building blocks" of the general map $y(x)$ defining the estimated density $\rho(x)$ through (3). In order to be useful, these elementary maps must satisfy some properties:

- They must include the identity map for $\beta = 0$.
- They must constitute a basis, through composition, for quite general maps. Thus linear maps are not good, since their composition never leaves the group of linear transformations.
- For robustness, they should have a simple spatial structure, without unnecessary oscillations. Our choices below are local, typically the identity plus localized bumps times linear functions.
- They must have a simple analytical dependence on the parameters β, leading to first and second derivatives of the likelihood function with respect to β that are not computationally intensive. We find below that a scalar β, the simplest of all choices, works best, since no complexity is needed at the level of the elementary maps: any complexity of the actual $y(x)$ can be built by the composition of simple maps.
- They should not deteriorate in high dimensions. The maps proposed in [15] require the laborious construction of general maps through the composition of one-dimensional transformations. This is always doable, as proved in [15], but not computationally efficient in high dimensions.

3.1 Radial expansions

Among the simplest elementary transformations suitable for building general maps are isotropic local expansion centered at a random point x_0, of the form

$$y = x + \beta f(\|x - x_0\|)(x - x_0), \tag{23}$$

depending on the single parameter β, positive for local expansions and negative for contractions.

A typical localization function f is given by

$$f(r) = \frac{1}{\alpha} \frac{\operatorname{erf}\left(\frac{r}{\alpha}\right)}{\frac{r}{\alpha}}, \tag{24}$$

where $r = \|x - x_0\|$. Another choice is

$$f(r) = \frac{1}{\alpha + r}. \tag{25}$$

Even though the two are similar in shape, each choice has its advantages: the former is smoother and more localized, while the latter is faster to compute and, more importantly, the corresponding map (23) can be inverted in closed form, yielding

$$\frac{x - x_0}{r} = \frac{y - x_0}{s},$$

where $s = \|y - x_0\|$ and

$$r = \frac{s - (\alpha + \beta)}{2} + \sqrt{\left(\frac{s - (\alpha + \beta)}{2}\right)^2 + \alpha s}.$$

This is useful in a number of applications that involve finding the inverse $x(y)$ of the normalizing map $y(x)$: producing synthetic extra sample points x_j from $\rho(x)$, for instance, can be achieved by obtaining samples y_j from the Gaussian $\mu(y)$, and writing $x_j = x(y_j)$).

Still one more choice is

$$f(r) = \frac{\left(1 - \frac{r}{\alpha}\right)^2}{\alpha} \quad \text{for } r < \alpha, \quad f(r) = 0 \text{ otherwise.} \tag{26}$$

This has the advantage of its compact support, which permits the easy superposition of various such maps simultaneously. All three families require $\beta > -\alpha$ for the maps to be one–to–one; the last one requires also that $\beta < 3\alpha$. Figure 2 compares the three functions, for $x_0 = 0$ and $\alpha = 1$.

The map in (23) has Jacobian

$$J = \left|(1 + \beta f)^{n-1}(1 + \beta(f + r f'))\right|$$

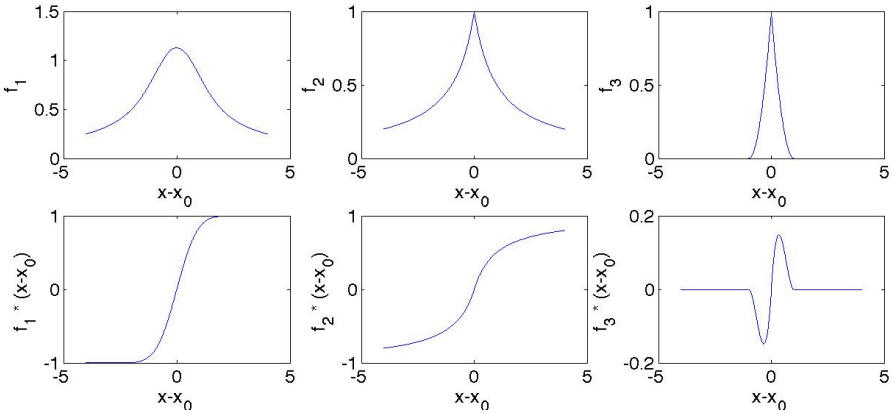

Figure 2: Three radial building blocks. The upper panels display $f(|x|)$, the lower ones $xf(|x|)$. On the left, a smooth, analytic block based on the error function, in the center, one with algebraic decay –and closed-form inversion– and, on the right, one with compact support.

and corresponding log-likelihood function L

$$\sum_j \log(\rho(x_j)) = \sum_j \Big\{ -\frac{1}{2}\Big[|x_0|^2 + 2(x_0, (1+\beta f)(x_j - x_0)) + ((1+\beta f) r_j)^2\Big] \\ +(n-1)\log(1+\beta f) + \log(1+\beta(f + r_j f'))\Big\}.$$

Then

$$\frac{\partial L}{\partial \beta}\Big|_{\beta=0} = \sum_j \Big\{\Big[n - (x_0, x_j - x_0) - r_j^2\Big] f + r_j f'\Big\}$$

and

$$\frac{\partial^2 L}{\partial \beta^2}\Big|_{\beta=0} = -\sum_j \Big[(n + r_j^2) f^2 + 2 r_j f f' + (r_j f')^2\Big] < 0,$$

so we may replace L by its quadratic approximation at $\beta = 0$, yielding the following approximation to the maximizer:

$$\beta = -\frac{\frac{\partial L}{\partial \beta}|_{\beta=0}}{\frac{\partial^2 L}{\partial \beta^2}|_{\beta=0}}.$$

3.2 One-dimensional maps

One may gain extra degrees of freedom by changing the map above into the component-wise

$$y^i - x_0^i = \left(1 + \beta^i f_i(|x^i - x_0^i|)\right)\left(x^i - x_0^i\right), \tag{27}$$

the composition of n one-dimensional maps, each depending on a parameter β^i. In this case,

$$L = \sum_j \sum_i \log\left[1 + \beta^i\left(f_i + \left|x_j^i - x_0^i\right| f_i'\right)\right]$$
$$-\frac{1}{2}\left[{x_0^i}^2 + 2x_0^i\left(1 + \beta^i f_i\right)\left(x_j^i - x_0^i\right) + \left(x_j^i - x_0^i\right)^2\left(1 + \beta^i f_i\right)^2\right],$$

$$\frac{\partial L}{\partial \beta_i}\Big|_{\beta=0} = \sum_j \left\{f_i + \left|x_j^i - x_0^i\right| f_i' - x_0^i f_i\left(x_j^i - x_0^i\right) - f_i\left(x_j^i - x_0^i\right)^2\right\}$$

and

$$\frac{\partial^2 L}{\partial \beta_i^2}\Big|_{\beta=0} = -\sum_j \left(f_i + \left|x_j^i - x_0^i\right| f_i'\right)^2 + f_i^2\left(x_j^i - x_0^i\right)^2,$$

so we must pick

$$\beta_i = -\frac{\frac{\partial L}{\partial \beta_i}|_{\beta=0}}{\frac{\partial^2 L}{\partial \beta_i^2}|_{\beta=0}}.$$

This family of maps is not isotropic, since it privileges the coordinate axes. To restore isotropy, one can rotate the axes every time-step, through a random orthogonal matrix. With this extra ingredient, this building block agrees with the one originally implemented in [15]; the only differences are the specific form of the stretching function, which in [15] was a more complex function depending on three parameters per dimension, and the maximization procedure, which is carried out here through a local quadratic approximation, not by first-order ascent of the log-likelihood.

3.3 Localized linear transformations

The radial expansions in (23) and, except for a minor twist, also the one-dimensional maps in (27) can be thought of as particular instances of a more general localized linear transformation of the form

$$y = x + f(\|x - x_0\|)A(x - x_0). \tag{28}$$

For the radial expansions, we have $A = \beta I$ and, for each one-dimensional map, $A = \beta nn^t$, where n is the column vector of direction cosines of the direction considered; in the latter case f applies not to $\|x - x_0\|$, but to $|n \cdot (x - x_0)|$.

For a general matrix A in (28), we have the following quadratic approximation to the logarithm of the density ρ at each point x:

$$\log(\rho(x)) \approx -|x|^2/2 + \sum_{i,j} L_i^j(x) A_i^j + \sum_{i,j,k,l} Q_{ij}^{kl}(x) A_i^j A_k^l, \tag{29}$$

where

$$L_i^j(x) = \left[\frac{\partial f}{\partial x^i} - f(x)x^i\right](x^j - x_0^j) + f(x)\delta_i^j$$

and

$$Q_{ij}^{kl}(x) = f^2(x)\left(\delta_i^l\delta_j^k + \delta_i^k(x^j - x_0^j)(x^l - x_0^l)\right) + \delta_k^j\left(2f(x)\frac{\partial f}{\partial x^i}(x^l - x_0^l)\right) + \frac{\partial f}{\partial x^i}\frac{\partial f}{\partial x^k}(x^j - x_0^j)(x^l - x_0^l).$$

Hence maximizing over A the quadratic approximation to the log-likelihood $L = \sum_m \log(\rho(x_m))$ yields the system

$$\sum_{kl}\left\{\sum_m \left(Q_{ij}^{kl}(x_m) + Q_{kl}^{ij}(x_m)\right)\right\} A_k^l = -\sum_m L_i^j(x_m).$$

Notice that this building block requires much more computational work than the isotropic expansions, hence its use would only be justified if it yielded better accuracy in a much smaller number of steps. We found in the experiments below that this is not typically the case, so we conclude that simpler maps, with only a handful of parameters β –such as the single one for the radial expansions– are to be preferred.

4 Examples

In this section, we use some synthetic examples to illustrate the procedure and to compare the efficiency of the various building blocks proposed above. In all examples, we have used for preconditioning only the two steps in (18,19), which re-center the observations at $x = 0$ and stretch them isotropically so as to produce a unitary average standard deviation.

As a first example, consider the two-dimensional probability density displayed in figure 3, given by

$$\rho(x, y) = e^{-\frac{1}{2}\theta^2} e^{-\frac{1}{2}\left(\frac{r-1}{0.1}\right)^2}, \tag{30}$$

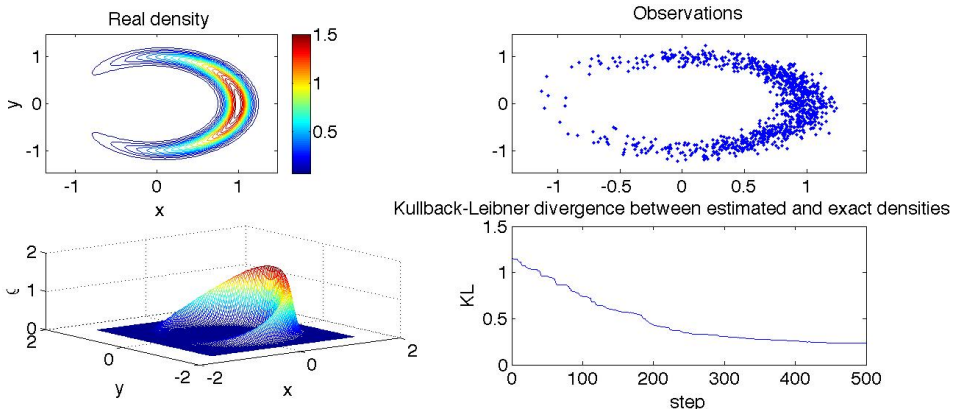

Figure 3: A synthetic two-dimensional example. On the left, the proposed probability density, displayed through contours and in perspective. On the right, the 1000 point sample used to test the procedure, and the evolution of Kullback-Leibler divergence between the analytical density and the one discovered by the algorithm.

where r and θ are the radius and angle in polar coordinates: a distribution concentrated in a small neighborhood of the unit circle, with maximal density at $(x, y) = (1, 0)$. Such distribution, with thin support and pronounced curvature, would be hard to capture with any parametric approach. Yet the proposed algorithm does a very good job, as shown in figure 4.

For the experiment displayed in figures 3 and 4, we have taken a sample of size $m = 1000$, used the radial expansion in (23) with $f(r)$ from (24), and adopted a value $n_p = 500$ for the calculation of the lengh-scale α in (13). The Kullback-Leibler divergence [9] between the exact and the estimated distributions, displayed in the last panel of figure 3, is given by

$$KL = \int \log\left(\frac{\rho_{ex}(y)}{\rho_{est}(y)}\right) \rho_{ex}(y)\ dy, \tag{31}$$

which is integrated numerically on the same grid used for the plots, a set of points carried passively by the algorithm, where ρ_{est} is known. Another possibility, much more efficient in high dimensions, is to estimate KL through Monte Carlo simulation:

$$KL \approx \frac{1}{N} \sum_{j=1}^{n} \log(\rho_{ex}(x_j) - \log(\rho_{est}(x_j)). \tag{32}$$

This also reveals the connection between the Kullback-Leibler divergence between estimated and exact densities and the log-likelihood of the estimated density, which the algorithm ascends.

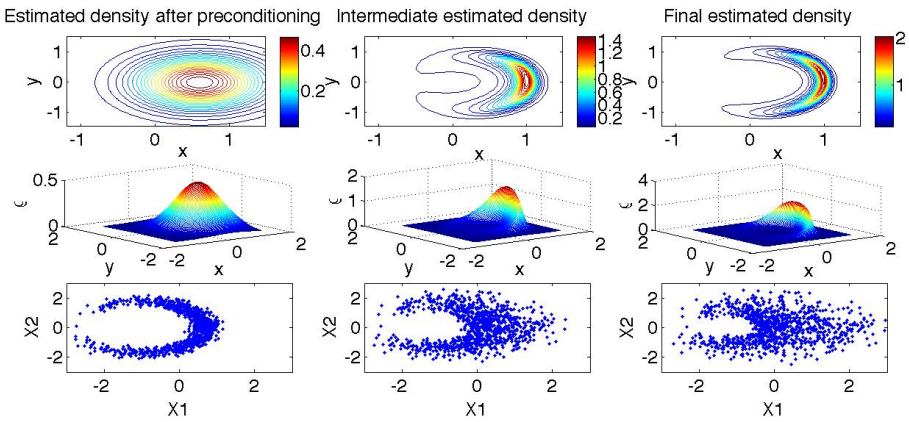

Figure 4: Evolution of the estimated density and normalized observations, through three snap-shots: on the left, the onset of the algorithm, after a pre-conditioning step that re-centers the observations and rescales them isotropically; in the center, the situation after 200 steps and, on the right, a final estimation after 600 steps. The top two rows display the estimated density; the third row the normalized observations.

Experimenting with the other building blocks proposed above yields entirely similar results. We conclude that the radial expansions are to be preferred, since their use is much less computationally intensive. Moreover, the simplicity of the radial expansions brings in an extra degree of robustness, as revealed by a much smaller sensitivity to the choice of n_p, the only free parameter of the algorithm.

Next we compare the procedure developed here with Kernel density estimation, the most popular non-parametric methodology in use [16]. We have adopted Gaussian kernels of the form

$$K_h(x,y) = \frac{1}{(2\pi)^{\frac{n}{2}}\ h^n} e^{-\frac{1}{2}\left(\frac{\|y-x\|}{h}\right)^2}, \tag{33}$$

and proposed the estimate

$$\rho(y) = \frac{1}{m}\sum_{j=1}^{m} K_h\left(x_j, y\right). \tag{34}$$

Hence each observation x_j contributes to the local density within a neighborhood whose size scales with h. This *bandwidth* plays a similar role to the n_p of our procedure, the typical number of points affected by each map.

Figure 5 displays the results of applying both procedures, at different bandwidths, to a sample with $m = 500$ points from the distribution in

(30). We have picked two values for h and n_p, one too large, slightly under-resolving the distribution, and one too small, slightly over-resolving it. Comparing the results from the two procedures, we can make the following observations:

- Both procedures are robust, capturing the main features of the probability density $\rho(x)$, whereas most parametric approaches would have done poorly.
- The mapping procedure yields smoother and tighter density profiles, and correspondingly smaller values of the Kullback–Leibler divergence between the exact and estimated densities.

The computational costs of both procedures are comparable: estimating the density at q points requires $m \times q$ evaluations of the kernel and $n_s \times q$ map applications respectively. Since the number n_s of iterations before convergence scales with the number m of observations, these two numbers of evaluations are of the same order. The mapping procedure has the additional cost of determining the optimal parameter β for each step, but this is comparatively unimportant when q is much larger than m.

Beyond the comparison of effectiveness, which depends on the actual problem in hand, one can describe the main differences between the two procedures:

- The estimated density is expressed in terms of the sum of kernel functions in one case and of the composition of elementary maps in the other.
- In the implementations discussed here, Kernel density estimation is explicit and deterministic, while there is a stochastic element to the choice of the centers for the elementary maps.
- Kernel density estimation is conceptually simpler, while the normalizing maps have a richer structure and more versatility.
- The kernels provide just an estimated density, while the new procedure also produces a normalizing map. This can be used for a variety of purposes, such as sampling.

Figures 6 and 7 show another two-dimensional experiment. In this case, the proposed density is the mixture of two anisotropic Gaussians, and, for illustration, the building block utilized is the general localized linear transformation in (28). Notice in figure 7 a feature associated with the dual nature of the algorithm: since the normalizing procedure cannot fully eliminate the gap between the two Gaussians without over-resolving, as shown

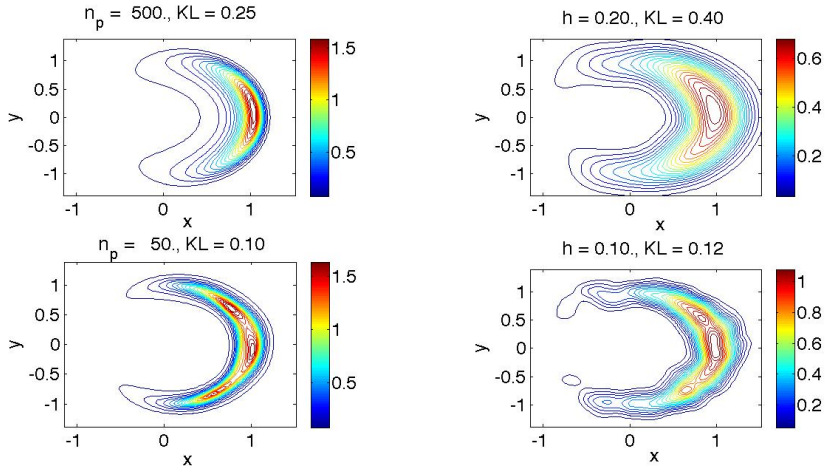

Figure 5: Comparison between density estimation through the mapping procedure and through Gaussian kernels at various bandwidths. On the left, two estimates performed using radial expansions; on the right, two Gaussian kernel density estimations. The top row uses values of n_p and h that slightly under-resolve the distribution, while the corrresponding values on the bottom row slightly over-resolve it. Both methodologies are robust and yield comparable results, yet the mapping procedure gets estimates that are both tighter and smoother, with corresponding lower values for the Kullback-Leibler divergence with the exact density underlying the sample.

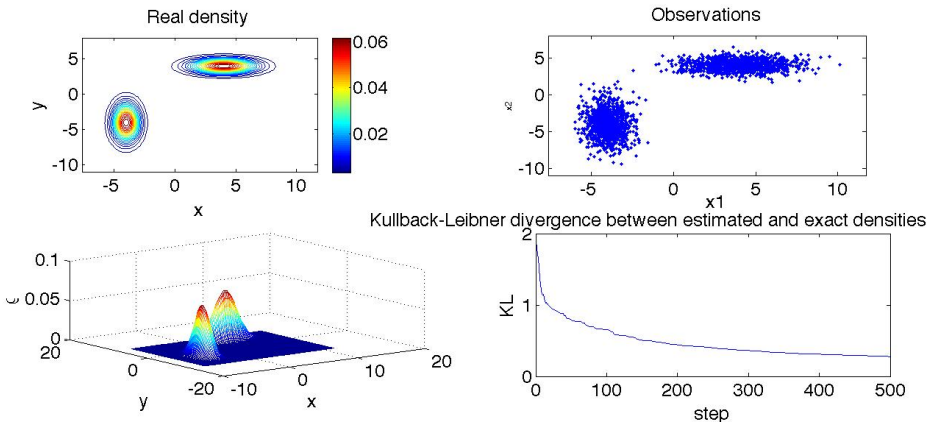

Figure 6: A second synthetic two-dimensional example, the mixture of two Gaussians. On the left, the proposed probability density, displayed through contours and in perspective. On the right, the 2000 point sample used to test the procedure, and the evolution of Kullback-Leibler divergence between the analytical density and the one discovered by the algorithm.

in the three bottom panels, the corresponding density estimation, displayed in the top panels, cannot fully separate the two. A kernel estimator would also be unable to fully separate the two components, but here the reason would be more straightforward: a bandwidth h small enough to separate them would over-resolve the estimation, particularly at the less populated tails of the distribution.

The examples above are two-dimensional to facilitate their display, yet the full power of the algorithm manifests itself in high-dimensional situations. Thus we consider next the equal-weight mixture of two n-dimensional normal distributions, centered at $x = \pm 2\mathbf{e_1}$. Here we have used a sample of $m = 1000$ points, $n_p = 500$, and again the isotropic radial expansion in (23,24). Figure 8 compares the evolution of the Kullback-Leibler divergence between the exact and estimated density in dimensions $n = 2$, $n = 5$ and $n = 10$. In order to enable a meaningful comparison between problems in different dimensions, the KL from (31) in the plots is normalized by A_n, the surface area of the n-dimensional unit sphere.

Notice that the rate of convergence does not deteriorate significantly with the dimension n –nor does the time per step, which is nearly independent of n for radial expansions. The value of $n = 10$ is beyond the largest one might have hoped to resolve with a sample of size $m = 1000$, since $2^{10} = 1024$: one has in average one observation per the 10-dimensional equivalent of a quadrant! Thus it is surprising that the algorithm resolves this density so

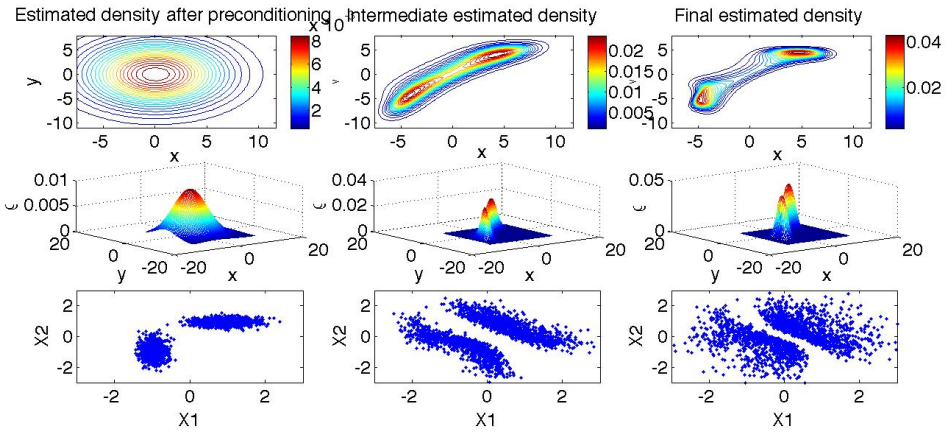

Figure 7: Evolution of the estimated density and normalized observations, through three snap-shots: on the left, the onset of the algorithm, after a pre-conditioning step that re-centers the observations and rescales them isotropically; in the center, the situation after 100 steps and, on the right, a final estimation after 500 steps. The top two rows display the estimated density; the third row the normalized observations.

well.

5 Conclusions

We have developed a methodology for non-parametric density estimation. Based on normalizing flows, the new procedure improves on the one developed in [15], in that it is more robust and efficient in high dimensions, and ascends the log-likelihood function through larger steps, based on a quadratic approximation rather than gradient ascent. It requires only one external parameter, n_p, with a clear interpretation: the level of resolution sought, measured in number of observations per localized feature of the estimated density. We have found that the simplest elementary transformations, such as localized radial expansions, are also the most efficient and robust building blocks from which to form the map that normalizes the data points.

Density estimation appears often in applications as a tool for more specific tasks. One advantage of the methodology developed here is its flexibility, which allows for easy adaptation to such tasks. Thus, in [1], we have adapted the algorithm from [15], a direct ancestor to the one in this paper, to do classification and clustering. Along similar lines, projects under way employ variations of the methodology proposed here to perform tasks as varied as medical diagnosis, relating behavioral traits to neuron classes in worms,

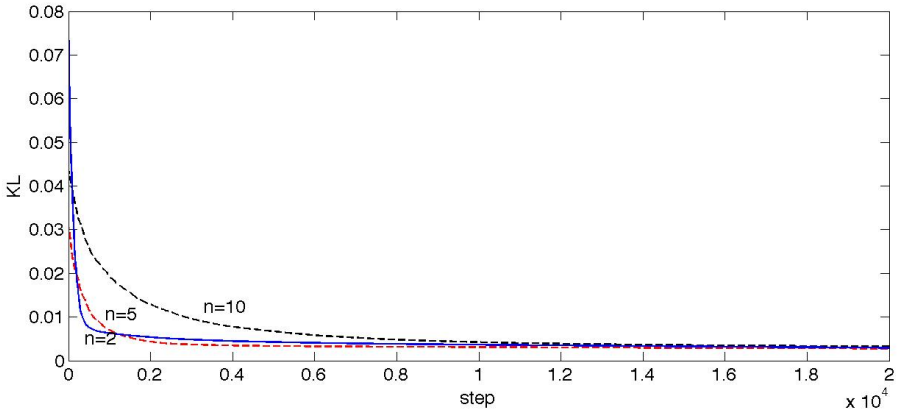

Figure 8: Evolution of the Kullback-Leibler divergence between the real and the estimated density for a Gaussian mixture in dimensions 2 (solid blue), 5 (dashed red) and 10 (dashed black.)

Montecarlo simulation, time series analysis, estimation of risk-neutral measures, and transportation theory. It is in the context of these more specific procedures that examples with real data make sense. In this paper, we have purposefully concentrated instead on "pure" density estimation, illustrating the new procedure only with synthetic examples. The advantage of these is that the knowledge of the precise distribution from which the observations are drawn allows us to quantify the accuracy of the estimated distribution, both visually, for small dimensional problems, and quantitatively, through the Kullback-Leibler divergence between the two.

Acknowledgments

The work of C. V. Turner was partially supported by grants from CONICET, SECYT-UNC and PICT-FONCYT, and the work of E. G. Tabak was partially supported by the National Science Foundation under grant number DMS 0908077.

References

[1] J. P. Agnelli, M. Cadeiras, E. G. Tabak, C. V. Turner and E. Vanden-Eijnden, "Clustering and classification through normalizing flows in feature space", *SIAM MMS*, **8**, 1784–1802, 2010.

[2] Bishop, C. M., **Pattern recognition and machine learning**, Springer, 2006.

[3] Botev, Z. I., Grotowski, J. F. and Kroese, D. P., "Kernel density estimation via diffusion", *Annals of Statistics*, **38**, 2916–2957, 2010.

[4] Chen, S. X., "Probability density function estimation using Gamma kernels", *Ann. Inst. Statist. Math.*, **52**, 471–480, 2000.

[5] Friedman, J., "Exploratory projection pursuit", *J. Amer. Statist. Assoc.*, **82**, 249–266, 1987.

[6] Hall, P., Racine, J. and Li, Q., "Cross-validation and the estimation of conditional probability densities", *J. Amer. Statist. Assoc.*, **99**, 1015–1026, 2004.

[7] Hastie, T., Tibshirani, R. and Friedman, J., **The elements of statistical learning**, Springer, 2001.

[8] Kohavi, R., "A study of cross-validation and bootstrap for accuracy estimation and model selection", *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence* **2**, 11371143, 1995.

[9] Kullback S. and Leibler, R. A. , "On information and sufficiency", *Annals of Math. Statistics* **22**, 79–86, 1951.

[10] Park, B. U. and Marron, J. S., "Comparison of data-driven bandwidith selectors", *J. Amer. Statist. Assoc.* **85**, 6672, 1990.

[11] Parzen, E., "On estimation of a probability density function and mode", *Annals of Math. Stat.*, **33**, 1065–1076, 1962.

[12] Rosenblatt, M., "Remarks on some nonparametric estimates of a density function", *Annals of Math. Stat.*, **27**, 832–837, 1956.

[13] Silverman, B.W., **Density Estimation for Statistics and Data Analysis**, London: Chapman & Hall/CRC, 1998.

[14] L. M. Smith, M. S. Keegan, T. Wittman, G. O. Mohler, and A. L. Bertozzi, "Improving Density Estimation by Incorporating Spatial Information", *EURASIP Journal on Advances in Signal Processing*, **2010**, 2010.

[15] Tabak, E. and Vanden-Eijnden, E., " Density estimation by dual ascent of the log-likelihood", *Comm. Math. Sci.* **8** , 217-233, 2010.

[16] Wasserman, L., **All of nonparametric statistics**, Springer, 2006.