Spatial Statistics 7 (2014) 62–78

Contents lists available at ScienceDirect

Spatial Statistics

journal homepage: www.elsevier.com/locate/spasta

Detection of tectonic faults by spatial clustering of earthquake hypocenters

Carlo Grillenzoni *

IUAV: Institute of Architecture, University of Venice, 30135 Venezia, Italy

ARTICLE INFO

Article history:
Received 21 November 2012
Accepted 12 November 2013
Available online 22 November 2013

Keywords:
Density ridges
Local Hessian
Kernel smoothing
Mean shift
Point data
Principal curves

ABSTRACT

Identification of the structure of tectonic faults from seismic data is mainly performed with clustering and principal curves techniques. In this paper we follow an approach based on the detection of the ridges of kernel densities estimated on earthquake epicenters. We use an iterative method based on the mean-shift algorithm for mode seeking, in which each step is made orthogonal to the principal direction of the local Hessian matrix. We carry out an extensive application to the historical data of San Francisco Bay area, and we compare the performance of similar methods with simulation experiments.

1. Introduction

It is well known that seismic events tend to occur along the tectonic faults of earth surface, as a consequence of frictions and breaks in the rocks caused by plate motions. Since tectonic faults are not directly observable, an attempt to identify their structure and dynamics is to fit point data corresponding to epicenters and hypocenters of earthquakes, e.g. Havskov and Ottemöller (2010).

Nonparametric smoothers are effective tools for fitting curves and surfaces to complex data (see Grillenzoni, 2005, 2008). However, they are unable to deal with *manifolds*, e.g. where to a value of the independent variable $X = x$, there correspond realizations of the dependent variable Y, which belong to different sets $S_j = [y_j, y_{j+1}]$, $j = 1, \ldots, s$. This situation often occurs in seismic data at small spatial scale, given the presence of multiple tectonic faults which run in parallel.

Useful methods to deal with such data are local principal curves, which can be defined as partial conditional expectations: $E(\mathbf{1}_{S_j} Y \mid X = x)$, where $\mathbf{1}$ is the indicator function. Several algorithms for principal curves (PC) have been defined: in the *top-down* approach a global model is iteratively

* Tel.: +39 59 350217.
E-mail address: carlog@iuav.it.

2211-6753/$ – see front matter
http://dx.doi.org/10.1016/j.spasta.2013.11.003

adapted at local level (e.g. Hastie and Stuetzle, 1989; Hastie et al., 2009; Meinicke et al., 2005; Carreira-Perpiñán and Lu, 2008), whereas in the *bottom-up* approach, principal curves are tracked starting from initial points and using only local data and constraints (e.g. Delicado, 2001; Kégl and Kryzak, 2002; Einbeck et al., 2005). However, these techniques have various difficulties in the presence of multiple or branched curves, and data-sets affected by outliers. Hence, they do not work well in fitting seismic data at small spatial scale.

As an alternative to direct curve estimation, one can model seismic data with multiple kernel densities, and define local curves as the *ridges* (crests) of such densities. In this context, there are adaptive methods which modify the basic mean-shift (MS) algorithm for mode seeking and data clustering (e.g. Cheng, 1995; Comaniciu and Meer, 2002). In particular, Bengio et al. (2006), Wang and Carreira-Perpiñán (2010), and Ozertem and Erdogmus (2011) perform projection of the data points on the ridges, using algorithms driven by covariance eigenvectors.

Other detection methods for spatial point data have been proposed by Allard and Fraley (1997), with a maximum likelihood method based on Voronoï tessellation, and by Stanford and Raftery (2000) with a two-step clustering procedure which alternate agglomeration and point relocation. Recently, *filament* classification techniques have been proposed by Hill et al. (2012) following an empirical Bayes approach, and by Genovese et al. (2012) using computationally geometry. In general, these methods require data decluttering and good initial values.

In this paper we present a MS-type estimator for local curves and surfaces of marked point processes (i.e. where the point data have different weight) and we propose an automatic method of bandwidth selection. We evaluate their performance on the seismic data of San Francisco Bay in order to detect the tectonic faults which are present in that area (San Andreas, Hayward, Calaveras, etc.). We also develop a *sequential* 2D algorithm – conditional to the levels of the depth coordinate – which can recover the 3D structure of faults in effective way.

The plan of the work is as follows: Section 2 deals with methodological aspects; Section 3 compares methods for epicenter (2D) data; Section 4 performs simulation experiments; Section 5 extends the proposed method to hypocenter (3D) data.

2. Algorithms for point data

Following a probabilistic setting, we assume that data are generated by a *marked* point process in the d-dimensional space (in seismology $d \leq 4$, if one includes time). A point process is defined as a sequence of random variables $\{\boldsymbol{x}_i, m_i\}$, $i = 1, 2, \ldots$ where $\boldsymbol{x}_i^T = [x_{1i}, \ldots, x_{di}]$ are the space–time coordinates, and $m_i > 0$ are the marks (magnitude) of the events (e.g. Grillenzoni, 2006). Under stationarity, the process is entirely described by the joint density $f(\boldsymbol{x}, m)$; in particular, principal curves of the points $\mathbf{x}_i$ correspond to the *ridges* of the marginal function $f(\boldsymbol{x})$.

To define ridge sets, we assume second order differentiability of $f(\cdot)$ and consider its Hessian matrix $\mathbf{H}_f(\boldsymbol{x})$; this describes the curvature properties of the density in every direction of the space. From differential geometry, a point $\boldsymbol{x}$ belongs to a univariate ridge set (e.g. curves in 2D space) if the gradient $\mathbf{g}_f(\boldsymbol{x})$ is orthogonal to all eigenvectors of $\mathbf{H}_f(\boldsymbol{x})$, with the sole exception of the maximal one $\mathbf{v}_1(\boldsymbol{x})$ which is called principal direction (see Eberly, 1996). In addition, as in local maximum conditions, the remaining $(d-1)$ eigenvalues $\lambda_j(\boldsymbol{x})$ must be negative:

$$\begin{aligned} &\text{ridge point} \quad \boldsymbol{x}: \quad \mathbf{v}_1^T(\boldsymbol{x})\,\mathbf{g}(\boldsymbol{x}) \neq 0, \quad \lambda_1(\boldsymbol{x}) \geq \lambda_j(\boldsymbol{x}), \quad \forall j > 1 \\ &j = 2, 3, \ldots, d: \quad \mathbf{v}_j^T(\boldsymbol{x})\,\mathbf{g}(\boldsymbol{x}) = 0, \quad \lambda_j(\boldsymbol{x}) < 0 \end{aligned} \tag{1}$$

where $\mathbf{g}(\boldsymbol{x}) = \partial f(\boldsymbol{x})/\partial \boldsymbol{x}$ and $(\lambda_1, \mathbf{v}_1)$ is the first eigenpair of $\mathbf{H}(\boldsymbol{x}) = \partial \mathbf{g}(\boldsymbol{x})/\partial \boldsymbol{x}^T$. The definition (1) extends to bivariate ridge sets (e.g. surfaces in 3D space) by considering the pair $(\lambda_2, \mathbf{v}_2)$ and the condition $\mathbf{v}_2^T(\boldsymbol{x})\,\mathbf{g}(\boldsymbol{x}) \neq 0$, etc. In this case, the eigenvectors $[\mathbf{v}_1, \mathbf{v}_2]$ span the tangent space of the ridge set (which includes the gradient), while $[\mathbf{v}_3, \ldots, \mathbf{v}_d]$ span its orthogonal subspace.

Given the sample data $\{\mathbf{x}_i^T = [\,\mathrm{x}_{1i}, \ldots, \mathrm{x}_{di}\,],\ \mathrm{m}_i\}_{i=1}^n$, the basic tool for estimating principal curves, in the form of ridge sets, is the kernel density function. Following Grillenzoni (2006), we consider a version where the observations $\mathbf{x}_i$ are weighted by their mark m_i, and each axes x_j has a specific

bandwidth $\beta_j > 0$

$$\hat{f}_n(x_1, \ldots, x_d | m = \mathrm{m}_i) = \left(\sum_{i=1}^{n} \mathrm{m}_i \prod_{j=1}^{d} \beta_j \right)^{-1} \sum_{i=1}^{n} \prod_{j=1}^{d} K\big[(\mathrm{x}_{ji} - x_j)/\beta_j\big] \mathrm{m}_i$$

$$\hat{f}_n(\boldsymbol{x}|\mathrm{m}_i) \propto \sum_{i=1}^{n} K\big[(\mathbf{x}_i - \boldsymbol{x})./\boldsymbol{\beta}\big]\, \mathrm{m}_i \tag{2}$$

where $K(\cdot)$ is the kernel function (typically a probability density), $\boldsymbol{\beta}^T = [\beta_1, \ldots, \beta_d]$ are the bandwidths and ./ is the element-wise division. In the 2D context, kernel densities on regular grids may be treated as digital images and their ridges can be detected with skeletonization techniques (see Eberly (1996) or Ozertem and Erdogmus (2007)). However, effectiveness of these methods depends on the grid resolution and they neglect the information contained in point data.

The second tool for principal curves detection is the mean-shift (MS) algorithm, that searches for the modal values of density functions (e.g. Carreira-Perpiñán, 2007; Comaniciu and Meer, 2002). It arises from the gradient of the function (2) where, assuming Gaussian kernels, one has

$$\hat{\mathbf{g}}_n(\boldsymbol{x}|\mathrm{m}_i) \propto \sum_{i=1}^{n} K\big[(\mathbf{x}_i - \boldsymbol{x})./\boldsymbol{\beta}\big]\, \mathrm{m}_i \,(\mathbf{x}_i - \boldsymbol{x}). \tag{3}$$

Now, equating (3) to 0 (the first order condition) and solving for $\boldsymbol{x}$ iteratively, provides the MS algorithm

$$\hat{\boldsymbol{x}}^{(k+1)} = \frac{\sum_{i=1}^{n} K\big[(\mathbf{x}_i - \hat{\boldsymbol{x}}^{(k)})./\boldsymbol{\beta}\big]\, \mathrm{m}_i\, \mathbf{x}_i}{\sum_{i=1}^{n} K\big[(\mathbf{x}_i - \hat{\boldsymbol{x}}^{(k)})./\boldsymbol{\beta}\big]\, \mathrm{m}_i}, \qquad \hat{\boldsymbol{x}}^{(0)} = \mathbf{c}_0 \tag{4}$$

where (k) is the iteration counter and $\mathbf{c}_0$ is the starting point. In practice, $\hat{\boldsymbol{x}}^{(k)}$ is a local mean of $\mathbf{x}_i$, which shifts toward the region with higher data density.

Carreira-Perpiñán (2007) showed that (4) is an expectation–maximization (EM) algorithm. Under the same regularity conditions as kernel estimators (namely, that f, K have square integrable derivatives, and $n \prod_j \beta_j \to 0$ as $n \to \infty$ and $\beta_j \to 0$), it converges to the mode of $\hat{f}_n(\boldsymbol{x})$ whose basin of attraction includes the starting point $\mathbf{c}_0$. By initializing $\hat{\boldsymbol{x}}^{(0)} = \mathbf{x}_i$, Eq. (4) provides a clustering algorithm for the data: $\hat{\boldsymbol{x}}_i^{(k)}$, $i = 1, 2, \ldots, n$ which detects all modal values of $\hat{f}_n(\boldsymbol{x})$.

The basic idea of principal curve (PC) estimation based on the MS algorithm is to divert the trajectory of every $\hat{\boldsymbol{x}}_i^{(k)}$ toward the nearest ridge point of $\hat{f}_n(\boldsymbol{x})$ in the basin of attraction. This can be done by exploiting the algebraic properties of the condition (1), in which the eigenvector $\mathbf{v}_1$ provides the principal direction of the ridge, and $\mathbf{V}_2 = [\mathbf{v}_2, \ldots, \mathbf{v}_d]$ span its orthogonal subspace. Normally, the motion of $\hat{\boldsymbol{x}}_i$ is parallel to $\mathbf{v}_1$; therefore, the deviation toward the ridge requires the projection onto the orthogonal complement: $\hat{\boldsymbol{y}}_i = \mathbf{P}_2\, \hat{\boldsymbol{x}}_i$, where $\mathbf{P}_2 = \mathbf{V}_2 \mathbf{V}_2^T$ is the projection matrix (see Ozertem and Erdogmus, 2011; Meyer, 2000, p. 430).

In this context, the fundamental step is the estimation of Hessian and its eigenvectors. A direct approach consists of computing the matrix on the kernel density (2). Assuming Gaussian kernels, differentiation of (3) provides

$$\hat{\mathbf{H}}_n(\boldsymbol{x}) \propto \big[\, \hat{\boldsymbol{\Sigma}}_n(\boldsymbol{x}) - \mathbf{I}_d \hat{f}_n(\boldsymbol{x}) \big], \tag{5}$$

$$\hat{\boldsymbol{\Sigma}}_n(\boldsymbol{x}) = \sum_{i=1}^{n} \frac{K\big[(\mathbf{x}_i - \boldsymbol{x})./\boldsymbol{\beta}\big]\, \mathrm{m}_i}{\left(\prod_{l=1}^{d} \beta_l \sum_{j=1}^{n} \mathrm{m}_j \right)} (\mathbf{x}_i - \boldsymbol{x})(\mathbf{x}_i - \boldsymbol{x})^T \tag{6}$$

where $\mathbf{I}_d$ is the identity matrix and $\hat{\boldsymbol{\Sigma}}_n(\boldsymbol{x})$ is the local covariance matrix. Notice that in scattered data, the values of $\hat{f}_n(\boldsymbol{x})$ are small for every $\boldsymbol{x}$, especially compared to the variances on the main diagonal of

(6); hence, one can assume $\hat{\mathbf{H}}_n(\boldsymbol{x}) \propto \hat{\boldsymbol{\Sigma}}_n(\boldsymbol{x})$. The exchange between Hessian and covariance is often done in the literature (e.g. Einbeck et al., 2005); however, it is only allowed by Eq. (5).

Proportionality factors do not influence the spectral properties of a matrix; hence, the eigendecomposition of the covariance (Hessian) provides

$$\hat{\boldsymbol{\Sigma}}_n = \hat{\mathbf{V}}_n \hat{\boldsymbol{\Lambda}}_n \hat{\mathbf{V}}_n^T, \qquad \hat{\mathbf{V}}_n = \left[\hat{\mathbf{v}}_1, \hat{\mathbf{V}}_2\right], \qquad \hat{\mathbf{P}}_2 = \left(\hat{\mathbf{V}}_2 \hat{\mathbf{V}}_2^T\right) \tag{7}$$

where $\hat{\boldsymbol{\Lambda}}_n = \text{diag}[\hat{\lambda}_1 > \hat{\lambda}_2 > \cdots > \hat{\lambda}_d]$ and $\hat{\mathbf{V}}_2 = [\hat{\mathbf{v}}_2, \ldots, \hat{\mathbf{v}}_d]$ (we have omitted the term $(\boldsymbol{x})$ in all matrices). Since $\boldsymbol{\Sigma}$ is symmetric and positive definite, it follows that $\mathbf{V}$ is orthogonal and $\mathbf{P}$ is idempotent, that is $\mathbf{P}\,\mathbf{P} = \mathbf{P}$. As regards PC estimation we have $\hat{\boldsymbol{y}}_i = \hat{\mathbf{P}}_2\,\hat{\boldsymbol{x}}_i$, where $\hat{\boldsymbol{x}}_i$ is the MS estimate initialized at $\mathbf{x}_i$ and the projection matrix must be computed together with the other components. However, having $\hat{\mathbf{P}}_2\,\hat{\boldsymbol{y}}_i = \hat{\boldsymbol{y}}_i$ one can also consider the expression

$$\hat{\boldsymbol{y}}_i = \hat{\boldsymbol{y}}_i + \hat{\mathbf{P}}_2\left(\hat{\boldsymbol{x}}_i - \hat{\boldsymbol{y}}_i\right) \tag{8}$$

this corresponds to the updating scheme of nonlinear optimization algorithms, and is numerically more efficient and stable than the simple projection.

The final step for obtaining the adaptive PC estimator is to insert Eq. (8) into the algorithm (4), and simultaneously computing the intermediate components (6)–(7). By defining the local weights

$$\omega_i\left(\boldsymbol{x}, \boldsymbol{\beta}\right) = \frac{K\left[(\mathbf{x}_i - \boldsymbol{x})./\boldsymbol{\beta}\right] \mathrm{m}_i}{\sum_{j=1}^{n} K\left[(\mathbf{x}_j - \boldsymbol{x})./\boldsymbol{\beta}\right] \mathrm{m}_j} \tag{9}$$

starting from (4), the integrated algorithm becomes

$$\hat{\boldsymbol{x}}_i^{(k+1)} = \sum_{j=1}^{n} \omega_j\left(\hat{\boldsymbol{y}}_i^{(k)}, \boldsymbol{\beta}\right) \mathbf{x}_j, \qquad \hat{\boldsymbol{x}}_i^{(0)} = \mathbf{x}_i, \qquad \hat{\boldsymbol{y}}_i^{(0)} = \mathbf{x}_i \tag{10}$$

$$\hat{\boldsymbol{\Sigma}}_i^{(k+1)} = \sum_{j=1}^{n} \omega_j\left(\hat{\boldsymbol{x}}_i^{(k+1)}, \boldsymbol{\beta}\right)\left(\mathbf{x}_j - \hat{\boldsymbol{x}}_i^{(k+1)}\right)\left(\mathbf{x}_j - \hat{\boldsymbol{x}}_i^{(k+1)}\right)^T \quad \mapsto \quad \hat{\mathbf{P}}_{2i}^{(k+1)} \tag{11}$$

$$\hat{\boldsymbol{y}}_i^{(k+1)} = \hat{\boldsymbol{y}}_i^{(k)} + \hat{\mathbf{P}}_{2i}^{(k+1)}\left(\hat{\boldsymbol{x}}_i^{(k+1)} - \hat{\boldsymbol{y}}_i^{(k)}\right), \qquad \hat{\boldsymbol{y}}_i^{(0)} = \mathbf{x}_i \tag{12}$$

for all $i = 1, 2, \ldots, n$, and whose stopping rule is $\max_j \left|\hat{y}_{ji}^{(k+1)} - \hat{y}_{ji}^{(k)}\right| < \varepsilon$.

Formula (12) shows that each PC estimate is updated on the basis of its previous value, by projecting the difference with the MS solution, on the subspace orthogonal to the first eigenvector of the local covariance centered on the MS itself. Thus, the above is *not* a simple principal component analysis (PCA, based on local covariance) of the original data: $\hat{\boldsymbol{z}}_i = \hat{\mathbf{V}}_i\,\mathbf{x}_i$. In fact, the transformation regards the estimate $\hat{\boldsymbol{x}}_i$ and is performed with the matrix $\hat{\mathbf{P}}_2$; once again, the MS algorithm is essential for approaching the ridge. The entire algorithm (9)–(12) is *quasi linear*, in the sense that it does not involve explicit computation of gradient and Hessian; hence, its convergence conditions could be less demanding than those established in Chacón et al. (2011).

2.1. Comparisons

We now compare the above with related methods mentioned before. Eqs. (10)–(11) are similar to those of the *local principal curve* (LPC) method of Einbeck et al. (2005). However, these authors do not consider the Hessian matrix and the algebraic relationships between tangent and orthogonal subspaces of the ridge set. As a consequence, they compute the PC through the heuristic projection rule $\hat{\boldsymbol{y}}_i = \hat{\boldsymbol{x}}_i + \beta\,\hat{\mathbf{v}}_1$, where $\beta > 0$ is the kernel bandwidth. In theory, this rule is equivalent to $\hat{\boldsymbol{y}}_i = \hat{\mathbf{P}}_2\,\hat{\boldsymbol{x}}_i$ (as a demonstration multiply both sides by $\hat{\mathbf{P}}_2$ and use the property $\hat{\mathbf{P}}_2\hat{\mathbf{v}}_1 = \mathbf{0}$); in practice, however, it is not efficient and requires ad-hoc adjustments to control the convergence to the ridge.

The main difference of (10)–(12) with the *subspace constrained mean shift* (SCMS) method of Ozertem and Erdogmus (2011) is the Hessian matrix. Relying on the transformation $\log \hat{f}_n(\boldsymbol{x})$, they obtain an expression which is a nonlinear function of density, gradient and covariance. Solving for $\boldsymbol{\Sigma}_x$ they obtain

$$\boldsymbol{\Sigma}_x^{-1}(\boldsymbol{x}) = \mathbf{g}_f(\boldsymbol{x})\,\mathbf{g}_f(\boldsymbol{x})^T / f_x^2(\boldsymbol{x}) - \mathbf{H}_f(\boldsymbol{x})/f_x(\boldsymbol{x})$$

then compute the eigenvectors $\mathbf{v}_j$ on $\hat{\boldsymbol{\Sigma}}_i^{-1}$, which are calculated with the kernel estimates $\hat{f}_i$, $\hat{\mathbf{g}}_i$, $\hat{\mathbf{H}}_i$. This approach poses methodological and computational questions, because in the above expression $\mathbf{H}_f \neq -\boldsymbol{\Sigma}_x^{-1}$ and the latter can be directly estimated from the data, i.e. without involving derivative estimates and their bias and variance (see Chacón et al., 2011). In general, it should be noted that nonlinear transformations of estimators worsen their statistical properties.

Wang and Carreira-Perpiñán (2010) developed a *manifold blurred mean shift* (MBMS) algorithm in which the observations $\mathbf{x}_i$ are replaced by the estimates $\hat{\boldsymbol{x}}_i^{(k-1)}$, $k > 1$ in the formula (4). This solution strongly increases the convergence speed of the MS method applied to Gaussian mixtures, see Carreira-Perpiñán (2006). For PC estimation, the motion of $\hat{\mathbf{x}}_i$ is deviated toward the ridge with the rule

$$\hat{\boldsymbol{y}}_i = \hat{\boldsymbol{\mu}}_i + \hat{\mathbf{P}}_i(\hat{\boldsymbol{x}}_i - \hat{\boldsymbol{\mu}}_i)$$

where the projection matrix and the centroids $\boldsymbol{\mu}_i$ are computed on nearest neighbor (kNN) estimates of the covariance and the mean of $\hat{\boldsymbol{x}}_i$. The resulting method is very efficient, but may have problems of convergence, such that to avoid shrinkage (bias) of the estimated manifold, the iteration process is stopped early.

2.2. Extensions

In summary, the algorithm (9)–(12) follows the mainstream of PC estimation based on MS, and improves existing methods in the areas of weighting observations, Hessian matrix computation and projection mechanism. It also admits other useful practical variants:

(i) In the covariance (11) one could use the same weights $\omega_i(\cdot)$ as Eq. (10). In addition, if principal curves have the same direction in space, then one can compute $\hat{\boldsymbol{\Sigma}}_i$ without the local weights, i.e. letting $\omega_i = 1$. The local feature of the estimates is still allowed by the centers $\hat{\boldsymbol{x}}_i$. To emphasize this approach, one could also use a single matrix given by the global covariance or the mean of the local covariances, namely $\bar{\boldsymbol{\Sigma}}_n = n^{-1}\sum_{i=1}^n \hat{\boldsymbol{\Sigma}}_i$.

(ii) As in Wang and Carreira-Perpiñán (2010) a simple change, which boosts the performance of the algorithm, consists of replacing the observations $\mathbf{x}_j$ with the estimates $\hat{\boldsymbol{y}}_j^{(k-1)}$ for all $k > 1$. In practice, one uses original data only in the first run, then re-smoothes the previous estimates; Eq. (10) then becomes

$$\hat{\boldsymbol{x}}_i^{(k+1)} = \sum_{j=1}^{n} \hat{\omega}_j\left(\hat{\boldsymbol{y}}_i^{(k)}, \boldsymbol{\beta}\right)\hat{\boldsymbol{y}}_j^{(k-1)}, \quad k = 2, 3\ldots. \tag{13}$$

Given the inherent problems of convergence and bias, one should use this solution parsimoniously, e.g. 2–3 runs interspersed by "normal" iterations, i.e. where $\hat{\boldsymbol{y}}_j^{(k-1)}$ in (13) are held fixed as the observations $\mathbf{x}_j$ in (10).

(iii) As in *adaptive* kernel estimation (Silverman, 1986, p. 100), the performance of (10)–(12) can be improved with variable bandwidths $\hat{\boldsymbol{\beta}}_i = \boldsymbol{\beta}_0/\hat{f}_n(\mathbf{x}_i)$, which are proportional to data sparsity. This solution can also reduce the bias caused by outliers. Instead, to avoid divergence, one should use decreasing bandwidths $\boldsymbol{\beta}_k = \boldsymbol{\beta}_0/k$ (where k is the iteration step), especially in the algorithm (13).

(iv) Formula (7) generalizes to projections on subspaces of higher dimension. In particular, the method (10)–(12) can estimate principal surfaces in 3D space by using a matrix based on the smallest eigenvector of $\hat{\boldsymbol{\Sigma}}_n$ (see Meyer, 2000, p. 323). In practice, $\hat{\mathbf{P}}_3 = \left[\hat{\mathbf{v}}_3\hat{\mathbf{v}}_3^T\right]$ must replace $\hat{\mathbf{P}}_2$ in Eq. (12).

2.3. Bandwidths

Selection of the bandwidth $\boldsymbol{\beta}$ is crucial both for the convergence and performance of the algorithms. Unlike kernel smoothing, there are not automatic procedures in manifold detection. We now discuss three solutions.

Consistently with the definition of principal curves as density ridges, a natural approach is to use bandwidth selectors for kernel densities. A heuristic solution is Silverman's rule-of-thumb (SR, which is optimal when f, K are Gaussian); but a general procedure is the likelihood cross validation (LCV), see Silverman (1986, p. 52). Weighted versions, which account for the marks m_i, are given by

$$\hat{\boldsymbol{\beta}}_{\mathrm{SR}} = 0.9\, \hat{\sigma}_{\mathrm{m}} / n^{1/(d+4)}, \quad d \geq 1 \tag{14}$$

$$\hat{\boldsymbol{\beta}}_{\mathrm{LCV}} = \arg \max_{\beta} \left[\left(\sum_{i=1}^{n} \mathrm{m}_i \right)^{-1} \sum_{i=1}^{n} \log \left[\hat{f}_{-i}(\mathbf{x}_i, \boldsymbol{\beta}) \right] \mathrm{m}_i \right] \tag{15}$$

where $\hat{\boldsymbol{\sigma}}_{\mathrm{m}}$ contains the weighted standard deviations of $\mathbf{x}_i$, and $\hat{f}_{-i}(\mathbf{x}_i)$ is the leave-one-out kernel density (2) evaluated at the ith point.

The second approach exploits the clustering ability of the MS algorithm (4). Since the number of clusters (modes) n_c provided by MS is inversely proportional to the bandwidth's size, by fixing a-priori n_c one can determine the value of β. The number n_c may be established by maximizing information criteria (IC), which balance likelihood function and parametric complexity, see Stanford and Raftery (2000), or indices of goodness-of-fit adjusted by n_c, as the F-type statistic

$$\hat{F}_n(n_c) = \frac{\mathrm{tr}(\hat{\mathbf{B}})/(n_c - 1)}{\mathrm{tr}(\hat{\mathbf{W}})/(n - n_c)}, \quad n_c = 2, 3 \ldots \tag{16}$$

where $\mathbf{B}$, $\mathbf{W}$ are between-group and within-group variance matrices. However, in spatial data these methods tend to overestimate n_c for the influence of small clusters. In 2D spaces, n_c can just be determined as the number of branches which are observable in the data scatterplot. Next, by running MS over a grid of β, and counting significant clusters, one can find the corresponding bandwidth.

The third approach follows the philosophy of kernel regression and spline smoothing, where it must be reached a compromise between fitting (bias) and smoothness (variance) of the estimates. As $\boldsymbol{\beta}$ increases, $\hat{\boldsymbol{y}}_i$ become more smooth, i.e. closer to each other, but more distant from the data $\mathbf{x}_i$, i.e. non representative. Therefore, the bandwidth selection should minimize the composite objective function

$$J_n(\boldsymbol{\beta}) = \frac{1}{n} \sum_{i=1}^{n} \left(\left\| \hat{\boldsymbol{y}}_i - \mathbf{x}_i \right\| + \alpha \frac{1}{n_k} \sum_{k=1}^{n_k} \left\| \hat{\boldsymbol{y}}_{ik}^{*} - \hat{\boldsymbol{y}}_i \right\| \right) \tag{17}$$

where $\| \cdot \|$ is the Euclidean norm, $\hat{\boldsymbol{y}}_{ik}^{*}$ is the kth nearest neighbor of $\hat{\boldsymbol{y}}_i$, and $\alpha > 0$ is a scale factor which balances the two components. The function J_n is not very sensitive to n_k and the coefficient α can be set equal 1 by standardizing the two components computed over the same grid of values. This approach can also be applied to the smoothing coefficients of other algorithms (see next section).

The third method is preferable since it is entirely automatic and is the most general, being applicable to the coefficients of any PC estimator. Given the regression nature of MS-based algorithms, it is also a valid substitute of least squares CV criteria which cannot be applied to manifolds.

3. Application to San Francisco data

To illustrate and test the methods discussed so far, we consider an application to the seismic data of San Francisco (SF) bay area. From the Northern California earthquake catalog (NCES, http://quake.geo.berkeley.edu/ncedc), we consider the events with longitude $x_1 \in [-123, -121.5]$, latitude $x_2 \in [37, 38.5]$, depth $x_3 \in [-25, 0)$, time $x_4 \in$ (March 1, 1968; May 31, 2012), and magnitude $m \geq 2.3$. This choice allows a reasonable sample size $n = 5197$, whereas the number of events

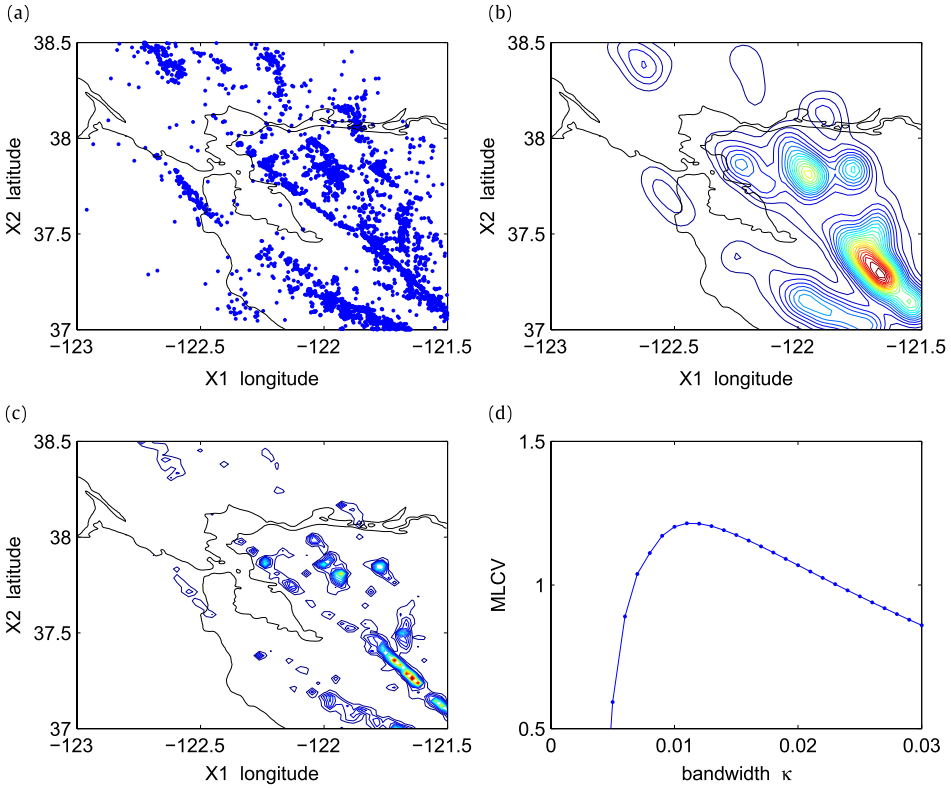

Fig. 1. (a) Epicenters of earthquakes with magnitude $m \geq 2.3$ in the SF bay area, in the period [1968.03, 2012.05]; (b) Iso-contours of kernel density (2) obtained with SR bandwidths [0.054, 0.072]; (c) Density estimated with the method of Botev et al. (2010); (d) Likelihood cross-validation function (15) with $\beta = (\beta_1 = \beta_2)$.

with $m \geq 2.0$ was nearly twice. Epicenters (x_{1i}, x_{2i}) are displayed in Fig. 1a, together with the coastal lines; they show the existence of about 10 clusters/branches which correspond to the main tectonic faults (San Andreas, Hayward, Calaveras, etc.; see http://fieldguides.gsapubs.org/content/7/215/F4.large.jpg).

Fig. 1b shows the kernel density (2) with bandwidths $\hat{\boldsymbol{\beta}}_{\text{SR}} = [0.054, 0.072]$, selected with (14). As a comparison, Fig. 1c provides the estimate obtained with the diffusion method of Botev et al. (2010), where $\hat{\boldsymbol{\beta}} = [0.0071, 0.0078]$. The latter is close to LCV solution in Fig. 1d: $\hat{\beta}_{\text{LCV}} = 0.011$, and to the value which minimizes the mean integrated squared error (MISE): $\hat{\beta}_{\text{MISE}} = 0.015$. However, all these bandwidths yield kernel densities which are clearly under-smoothed.

Methods (14) and (15) provide substantially different bandwidths; we now apply the other approaches discussed in Section 2. Fig. 2a shows the inverse relationship between the value of β and the number of clusters n_c provided by MS (4). Notice that for the mean value $\bar{\beta}_{\text{SR}} = 0.063$ one has $n_c = 10$, which is the rough number of branches in Fig. 1a. Instead, the maximization of statistic (16) yields $n_c = 24$, and $\hat{\beta}_{\text{LCV}} = 0.011$ strongly increases n_c. Fig. 2b shows the clustering activity of MS (4) with $\beta = 0.063$, at different steps of the iteration process.

As regards the method (17) (which aims to balance fitting and smoothness), we obtain the results in Fig. 3. Panel (c) shows that the optimal value of β for the PCMS (10)–(12) is not far from that indicated by Silverman's rule. Panel (f) provides the best value of Knn (the number of nearest neighbors for estimating the local covariance matrix) for the MBMS algorithm of Wang and Carreira-Perpiñán (2010). Suitable values of Knn are in the interval 10–20.

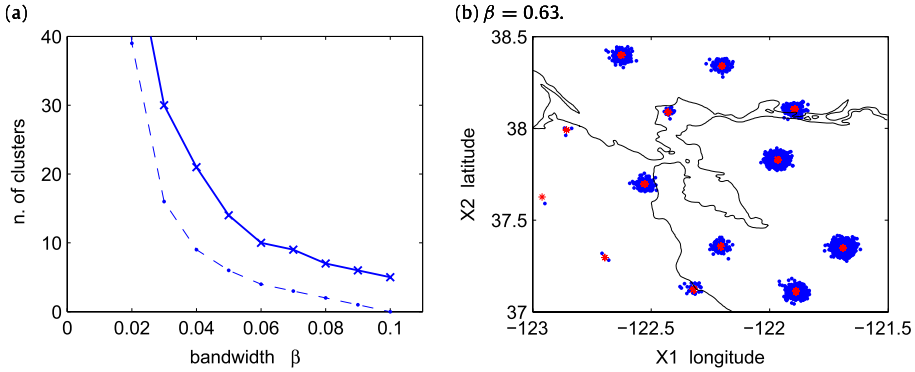

Fig. 2. (a) Relationship between the bandwidth β and the number of clusters n_c provided by MS algorithm (4) in the SF data. The dashed line shows the number of negligible clusters (with less than 10 units); (b) Clustering performance of (4), with $\beta = 0.063$, at iterations $k = 10$ (blue) and $k = 20$ (red). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

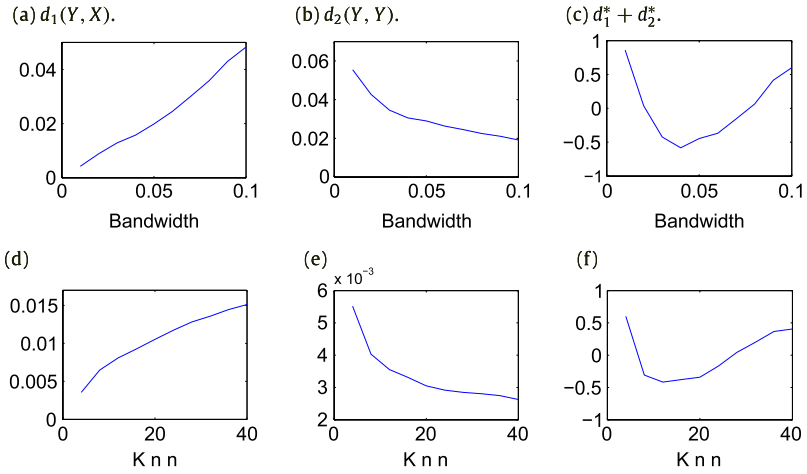

Fig. 3. Selection of the smoothing coefficients of ridge estimators by means of the criterion (17) rewritten as $J_n = d_1(Y, X) + \alpha\, d_2(Y, Y)$: (a)–(c) results for β of PCMS (10)–(12); (d)–(f) results for Knn of MBMS. In particular: (a), (d) mean distance between data and estimates: $d_1(Y, X)$; (b), (e) mean nearest neighbor distance between the estimates: $d_2(Y, Y)$; (c), (f) sum of the two distances standardized.

To show the goodness of bandwidth selections (14) and (17), we run the algorithm (10)–(12) with different values of β (a single coefficient is used). The estimates for $\beta = 0.030, 0.063, 0.096$ are given in Fig. 4; the best visual result, in terms of trade-off between fitting and smoothness, is the intermediate one.

To improve the results of Fig. 4b, we run the algorithm (12) without the local weighting in the covariance matrix (11) (i.e. letting $\omega_i = 1$), and we apply the smoothed version (13). The latter is very efficient at the beginning, but it yields oversmoothed and biased solutions. To stop it at a useful point, we have adopted the decreasing bandwidth $\beta_k = 0.07/k$; the results are reported in Fig. 5b.

The estimates which better represent the actual tectonic faults are those in Fig. 5a. They can be further improved by using a constant covariance matrix, as

$$\hat{\boldsymbol{\Sigma}}_{\mathrm{m}} = \left(\sum_{i=1}^{n} \mathrm{m}_i\right)^{-1} \sum_{i=1}^{n} \left(\mathbf{x}_i - \bar{\mathbf{x}}_{\mathrm{m}}\right)\left(\mathbf{x}_i - \bar{\mathbf{x}}_{\mathrm{m}}\right)^T \mathrm{m}_i \quad \mapsto \quad \hat{\mathbf{P}}_2.$$

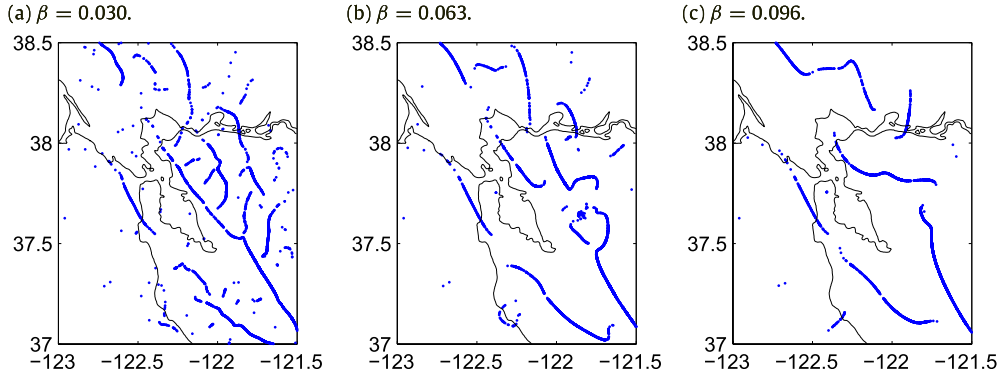

Fig. 4. Ridge estimates provided by the algorithm (10)–(12) on the data of Fig. 1a with different values of the bandwidth $\beta = (\beta_1 = \beta_2)$.

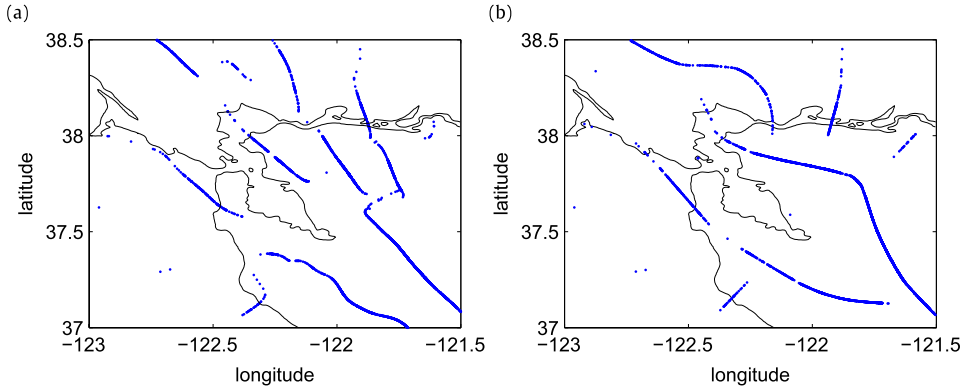

Fig. 5. Estimates provided by: (a) (11) with $\omega_i = 1$; (b) (13) with $\beta_k = 0.07/k$.

By replacing Eq. (11) with $\hat{\mathbf{\Sigma}}_{\mathrm{m}}$ provides the estimates in Fig. 6. It shows several branches that closely correspond to the ridges of the observable clusters; they also agree with the tectonic faults identified by geologists, as those at the link http://fieldguides.gsapubs.org/content/7/215/F4.large.jpg. This results is allowed by the uniform direction (NW-SE) of the faults; however, the estimated curves are not straight lines and admit bifurcations. Compared with Allard and Fraley (1997, p. 1493), who have treated the same case study at different spatial scale, the estimates in Figs. 4–6 are sharper.

3.1. Comparisons

To evaluate the proposed method, we compare its results with those of other approaches; we begin with methods based on the MS algorithm. Bengio et al. (2006) represent data with Gaussian mixture models (GMM), estimate them with the expectation–maximization (EM) algorithm and then extract their ridges. The weakness of this solution is in the lack of parametric identifiability of the coefficients (means, covariances an weights) of the mixture, even for a small number of components n_f. This generates multiple solutions, which are oversmoothed and seldom coincide with the actual ridges. Estimates obtained with $n_f = 8$ (the number of branches of Fig. 6) are provided in Fig. 7a.

The second approach is the subspace constrained mean shift (SCMS) of Ozertem and Erdogmus (2011), which computes the covariance matrix in a complex way. Fig. 7b displays the estimates generated with $\beta = 0.063$ and a maximum number of iterations $n_h = 100$. It resembles Fig. 4b, but

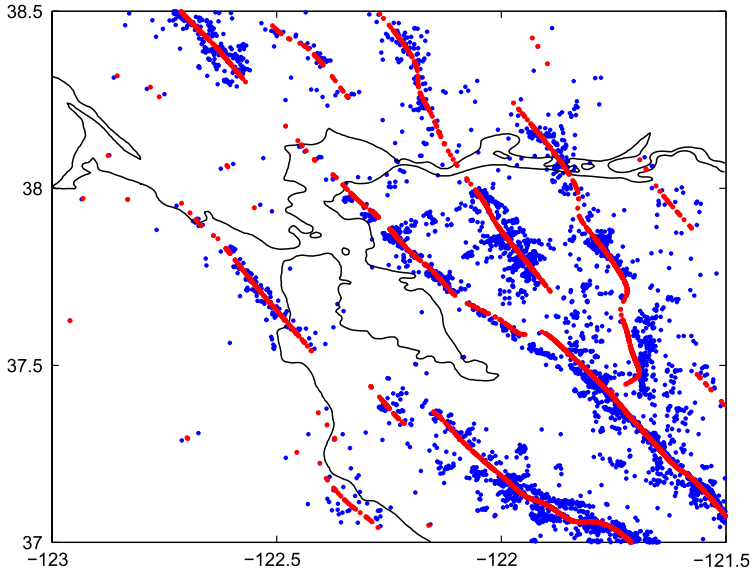

Fig. 6. Results of (10)–(12) with $\hat{\boldsymbol{\Sigma}}_i = \hat{\boldsymbol{\Sigma}}_{\mathrm{m}}$: Ridges (red), Data (blue). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

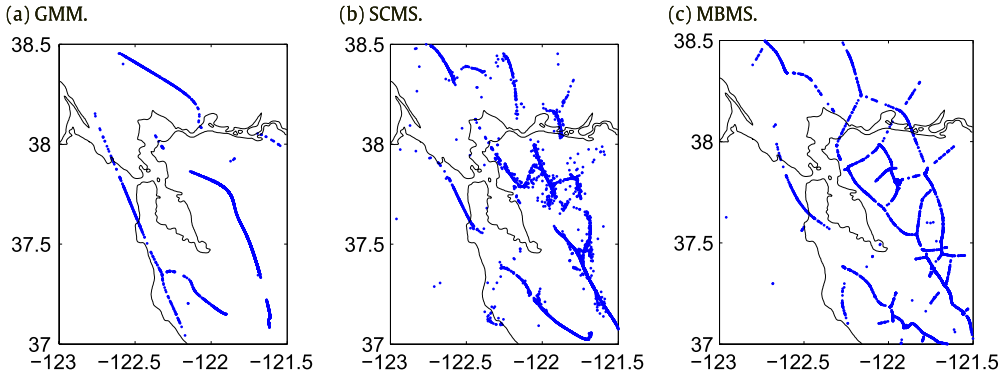

Fig. 7. Results provided by other approaches: (a) GMM with $n_f = 8$; (b) SCMS with $\beta = 0.063$ and $n_h = 100$; (c) MBMS with $\beta = 0.063$ and Knn $= 20$.

the curves have greater variability and are affected by many unclustered data points. These problems are probably caused by the covariance estimator.

Finally, we apply the manifold blurred mean shift (MBMS) of Wang and Carreira-Perpiñán (2010), which is based on a smoothed version of the MS component. For the selection of the bandwidth β and the number of nearest neighbors Knn, we use the results of Fig. 3. We adopt the version with *partial graph* and Knn $=$ 20 since provide more realistic estimates; however, greater values of Knn yield shrinkage and problems of convergence. Results are reported in Fig. 7c.

As a comparison, one can see that results in Fig. 7 are inferior to those in Figs. 4b, 5a and 6. The evaluation can be extended to methods which fit connected curves to the data. However, since the case study is characterized by many disconnected branches (which correspond to separate faults), the results are very disappointing. For this reason, we report them in the Appendix.

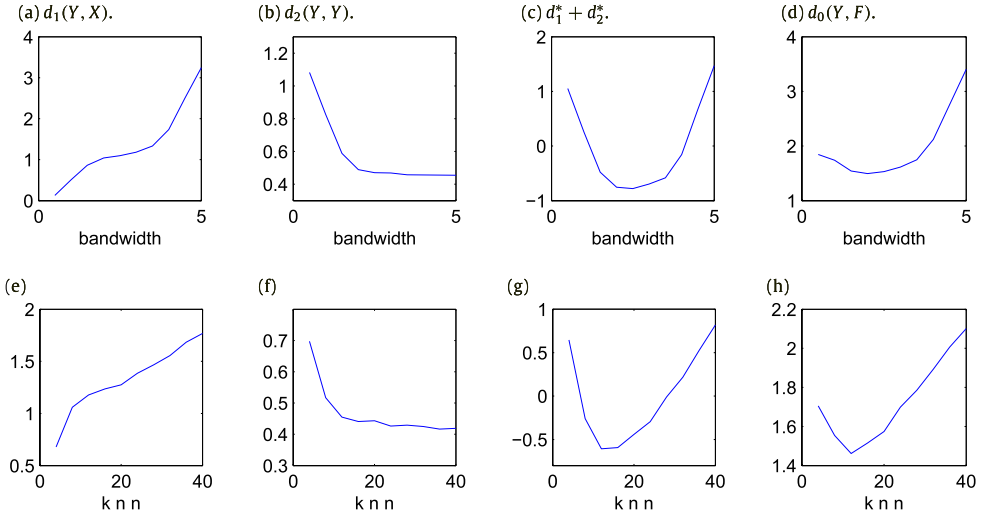

Fig. 8. Application of the criterion (17) to the data of Fig. 9a: (a)–(d) Results for the bandwidth of PCMS, (e)–(f) results for the Knn factor of MBMS. As in Fig. 3: (a), (e) distance between data and estimates d_1; (b), (f) distance between the estimates d_2; (c), (g) sum of the two distances standardized; (d), (h) distance between estimates and ground model d_0. The minima of the last two graphs should coincide.

4. Simulation experiments

In this section we carry out simulation experiments to test the performance of MS based algorithms on complex data. We consider a 2D geometric structure which overlaps a circle and two straight lines: $\mathbf{F} = [\,\mathbf{c}_0;\ \mathbf{l}_1;\ \mathbf{l}_2\,]$, originating five crossings. Each point $\mathbf{f}_i$ of the model is blurred with a Gaussian noise $\boldsymbol{e}_i \sim \text{IN}(\mathbf{0}, \mathbf{I}_2\ 1.5^2)$, generating the observed process $\mathbf{x}_i = \mathbf{f}_i + \mathbf{e}_i$. We consider a sample size $n = 600$ (300 for the circle) and $N = 200$ replications; an example is given in Fig. 9a.

We apply the PCMS algorithm (10)–(12), with variable and constant covariance matrix, the MBMS of Wang and Carreira-Perpiñán (2010), the SCMS of Ozertem and Erdogmus (2011), and the GMM method. Selection of smoothing coefficients is performed as in Section 3. In particular, the bandwidth follows Silverman's rule (14) and the Knn coefficient of MBMS is selected with the approach (17). The results for a single sample are provided in Fig. 8c and g: It can be seen that $\hat{\beta}_{\text{SR}} = 2.73$ is close to the minimum of the standardized J-function (J^*, see panel (c)), and Knn $= 10$ is a reasonable value (see panel (g)). More importantly, in both cases the minima of J^* are close to the optimal values which minimize the distance between estimates and theoretical model (see panels (d) and (h)). This result fully legitimates the approach (17) in real data where $\mathbf{f}$ is actually unknown.

After performing N replications, the estimates are summarized with average mean (M) and maximum (X) squared errors (SE) as follows

$$\text{MSE}_1 = \frac{1}{N}\sum_{j=1}^{N}\left(\frac{1}{n}\sum_{i=1}^{n}\left\|\hat{\boldsymbol{y}}_{ij} - \mathbf{f}_i\right\|^2\right), \qquad \text{XSE}_1 = \frac{1}{N}\sum_{j=1}^{N}\left(\max_i\left\|\hat{\boldsymbol{y}}_{ij} - \mathbf{f}_i\right\|^2\right) \tag{18}$$

where $\|\cdot\|$ is the Euclidean norm. Similar statistics for average estimates are

$$\text{MSE}_2 = \frac{1}{n}\sum_{i=1}^{n}\left\|\frac{1}{N}\sum_{j=1}^{N}\hat{\boldsymbol{y}}_{ij} - \mathbf{f}_i\right\|^2, \qquad \text{XSE}_2 = \max_i\left\|\frac{1}{N}\sum_{j=1}^{N}\hat{\boldsymbol{y}}_{ij} - \mathbf{f}_i\right\|^2 \tag{19}$$

which correspond to single estimations computed with $n\,N$ data.

Numerical results are reported in Fig. 9 and Table 1: It can be noted that statistics (18)–(19) of PCMS and MBMS are similar, and outperform the others. The worst method is GMM, which was

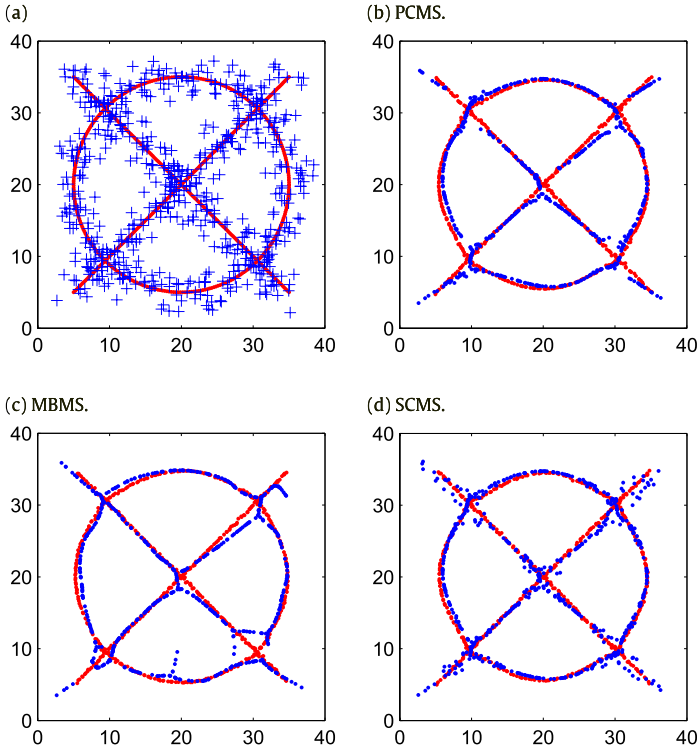

Fig. 9. (a) Theoretical model (solid red), and a sample of size $n = 600$ obtained by adding a noise $N(0, 1.5^2)$ to its coordinates. (b)–(d) Average PCMS, MBMS, SCMS estimates over $N = 100$ replications (red) and a single estimate (blue). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 1
Results of the simulation experiment with the manifold represented in Fig. 9. The entries are the values of the statistics (18)–(19) over $N = 200$ replications. $PCMS_0$ corresponds to the algorithm (11) with constant (non-local) covariance matrix.

Stat.	Eq.	n	σ_e	PCMS	MBMS	SCMS	$PCMS_0$	GMM
MSE_1	(18a)	600	1.5	**1.54**	1.56	1.71	1.99	2.92
MSE_1	(18a)	300	2	2.26	**2.23**	2.58	2.60	3.71
XSE_1	(18b)	600	1.5	6.26	**6.23**	9.31	6.69	14.49
XSE_1	(18b)	300	2	**7.78**	8.25	10.98	8.05	15.57
MSE_2	(19a)	600	1.5	0.42	**0.34**	0.64	0.78	1.44
MSE_2	(19a)	300	2	**0.66**	0.70	1.04	0.89	1.50
XSE_2	(19b)	600	1.5	0.93	**0.91**	1.30	2.52	4.77
XSE_2	(19b)	300	2	**1.38**	1.85	2.19	3.07	5.19

implemented with 10 Gaussian components. Further experiments have shown that MBMS with Knn = 5, 15 does not improve; hence, the selection of Knn described in Fig. 8g is satisfactory. As shown by the XSE statistics and Fig. 9d, the SCMS method is sensitive to the greater variability of estimates yielded by its covariance matrix. Indeed, the results of $PCMS_0$ (which has constant covariance) may be better. Finally, these conclusions partly depend on the simulated model, which contains various crossings and has a low signal-to-noise ratio. In standard manifolds, such as circles and spirals, the performance of MBMS tends to improve (see Wang and Carreira-Perpiñán (2010)).

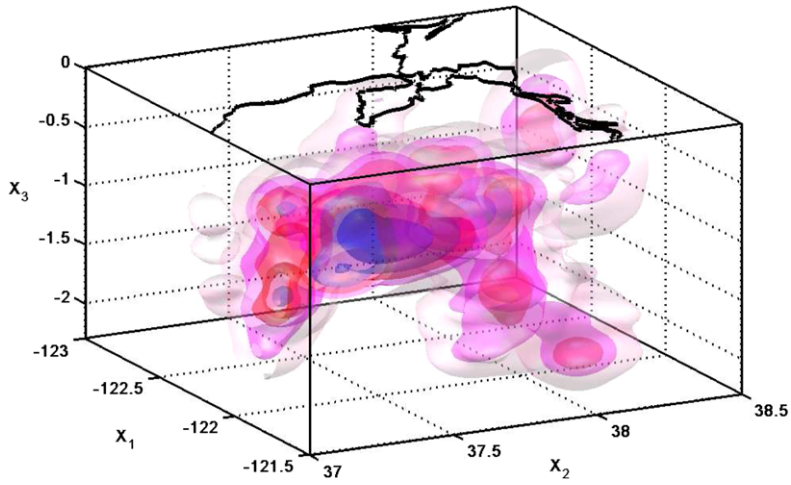

Fig. 10. Iso-surfaces of the 3D kernel density of San Francisco data. The depth coordinate has been divided by 10 (km).

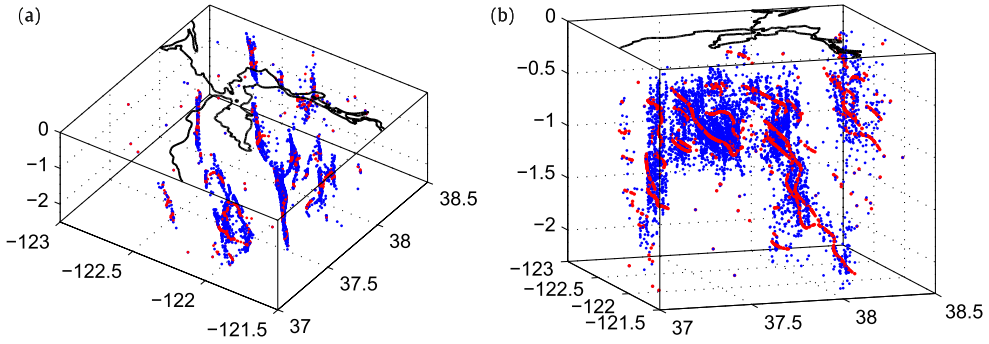

Fig. 11. Principal surfaces (blue) and 3D ridges (red) of San Francisco data obtained with the algorithm (10)–(12): (a) SE-NW view; (b) East view. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

5. The 3D structure of SF faults

A more interesting issue is the analysis of the 3D structure of San Francisco faults. This requires the inclusion of the depth coordinate x_3 in the data vector $\mathbf{x}_i$. The 3D kernel density, estimated with bandwidth selection (14), is shown in Fig. 10; the SE-NW direction offers the most clear viewpoint.

The estimation of principal surfaces can be carried out with the algorithm (10)–(12) by using the data vector $\mathbf{x}_j^T = [\,\mathrm{x}_{1j}, \mathrm{x}_{2j}, \mathrm{x}_{3j}]$ and the matrix $\mathbf{P}_3 = \left[\mathbf{v}_3\mathbf{v}_3^T\right]$. Instead, density ridges require the projection matrix $\mathbf{P}_2$ built with two minor eigenvectors. As in Section 3, locally constant covariance matrices provide a superior clustering performance in terms of a smaller number of sparse points; further, they allow ridges to have a uniform direction SE/top–NW/bottom (see Fig. 11). This is consistent with the S–N scrolling motion of the Pacific plate in California, which has also a W–E *subduction* component with respect to the North American plate.

The extraction of principal surfaces from 3D point data is a challenging issue. The proposed algorithm performs better than other methods discussed in Section 2, but the results in Fig. 11b are not satisfactory. In order to improve this situation, we fit the 3D cloud with a 2D algorithm which works *sequentially* with respect to the levels of the depth coordinate. In practice, we define a regular

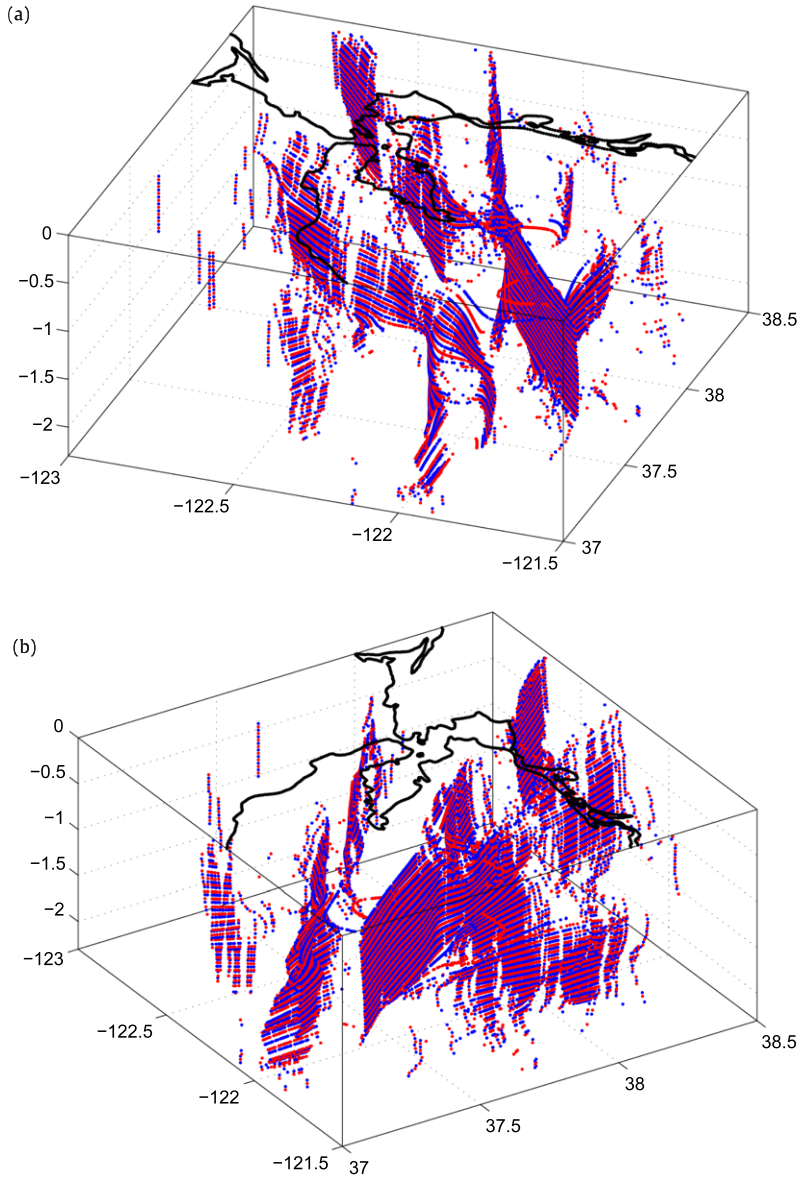

Fig. 12. Slice 2D ridge estimates with respect to the depth coordinate. The depth coordinate has been divided by 10 (km).

grid of values for $x_3 \in [\,x_{3\,\min}, 0)$, we insert the kernel $K[\,(x_{3i} - x_3)/\beta_3]$ in Eq. (9), with $\mathbf{x}_i^T = [x_{1i}, x_{2i}]$, and we perform the estimates (10)–(12) for each x_3.

This approach is similar to multi-slice scanning of volumes and is capable of detecting multiple faults. In addition, it is more sensitive and stable than the direct extraction of surfaces by means of 3D algorithms. Results for San Francisco data are shown in Fig. 12, where horizontal layers are displayed with different colors for enhancing the vertical structure. One can see that the surfaces are clearer than those in Fig. 11b and enable to understand the tectonic structure of that area. In particular, the fact that the faults are not *oblique* is consistent with the main motion (N–S scrolling) of Pacific and American plates.

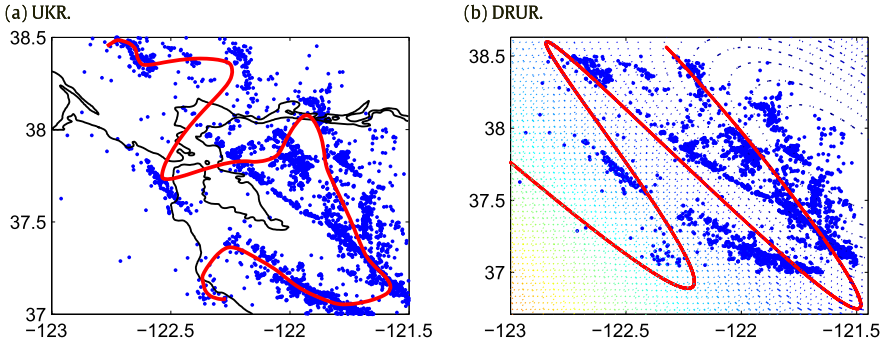

Fig. 13. Results provided by other approaches: (a) UKR (with default coefficients) (b) DRUR with $n_f = 4$.

6. Conclusions

In this paper we have addressed the problem of detecting the structure of tectonic faults by spatial clustering of epi/hypocenters of seismic events. By defining local curves and surfaces as the ridges of kernel densities of point data, we have exploited mean-shift based algorithms. These are mode-seeking methods in which the iterative steps are driven toward the density ridges by means of the eigenvectors of the local Hessian matrix.

We have checked that the step updating mechanism (8), the kind of covariance weighting and the iterative management (13) are important factors that determine the performance of the algorithm. Also the choice of the bandwidth is important, but optimal selection methods developed in kernel density estimation may not provide suitable solutions. Methods based on the clustering ability of MS and the trade-off between fitting and smoothness of the estimated manifold may be preferable.

Focusing on the San Francisco case-study, we have performed several comparisons with methods discussed in the statistical learning literature. These have been proved effective in simulation experiments or real data having simple structures; in particular, SCMS and MBMS outperform GMM and LPC algorithms. However, in complex seismic data, with multiple branches that cross each other and affected by outliers, they exhibit some difficulties.

Their problems can be overcome in part by using constant covariance matrices and double iteration (smoothing) of the estimates. The latter, in particular, can reduce the presence of sparse (unclassified) points. We have also provided a sliced implementation of the 2D algorithm which yields more reliable 3D surfaces.

Open issues are related to multi-dimensional (3D) fitting and to the selection of the smoothing coefficients. The approach of variable bandwidths (either at spatial and iterative level) can make this issue less urgent.

Acknowledgments

Sincere thanks to the editors and the reviewers for many helpful suggestions.

Appendix A

The unsupervised kernel regression (UKR) method of Meinicke et al. (2005) assumes that principal curves are locally driven by latent explanatory variables. It first fits the data with a kernel regression smoother, then estimates the latent variables by minimizing the reconstruction errors. The method selects the bandwidth by cross-validation and is iterated until convergence.

The dimensionality reduction unsupervised regression (DRUR) approach of Carreira-Perpiñán and Lu (2008) faces the same problem, but from a double mapping viewpoint. Given the data-set, it first performs dimensionality reduction as in PCA, then optimizes a regularized objective function for dimensionality reconstruction. Similarly to UKR, it alternately optimizes over the latent coordinates given reconstruction and projection mappings respectively.

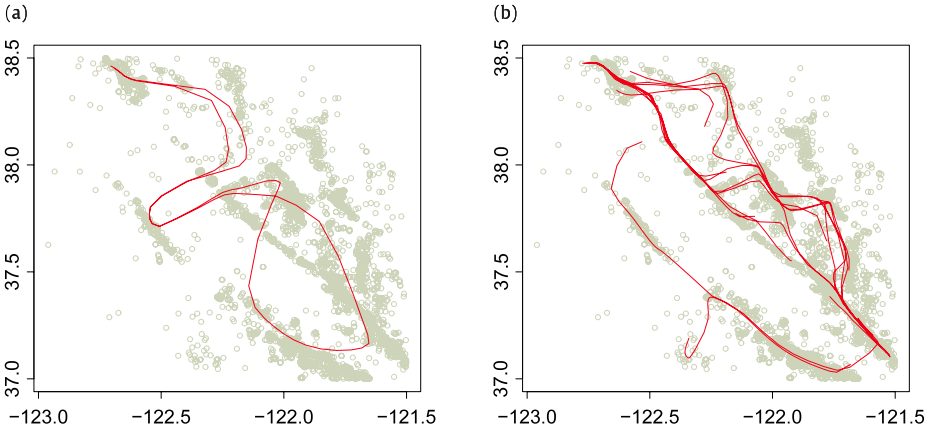

Fig. 14. Results provided by the LPC method: (a) default options; (b) ad hoc options.

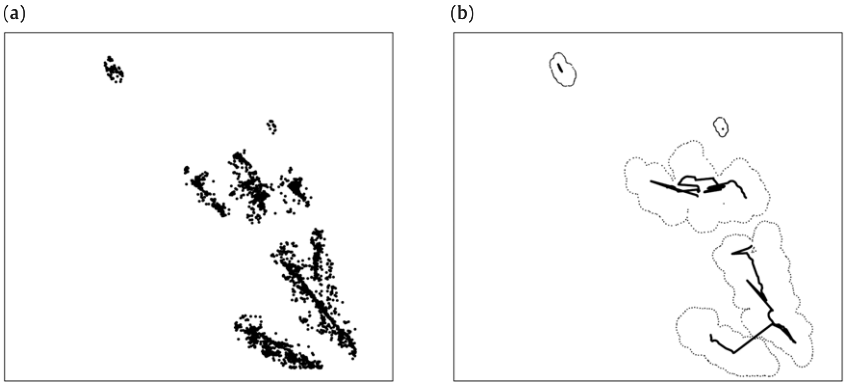

Fig. 15. Filament estimation method: (a) Decluttered data; (b) Medial estimates.

As the GMM approach, DRUR requires the definition of the number of radial basis functions (RBF) n_f; the risk of instability and divergence are proportional to it. The application to San Francisco data provides the results in Fig. 13; they clearly show the limits of fitting complex data with a single curve.

Similar problems are shared by other approaches implemented with the R-package. The local principal curve (LPC) method of Einbeck et al. (2005, 2010) tracks a connected curve through the local centers of mass by driving the MS solution with the eigenvectors of the local covariance. Specific constraints are used in correspondence of crossings and bifurcations. The application to San Francisco data, by using default and ad hoc options, provides the results in Fig. 14.

Finally, we consider the *filament* estimation method of Genovese et al. (2012), which merges nonparametric smoothing and computational geometry. The method considers the bivariate kernel density, performs decluttering for deleting density tails, then estimates the density ridges by using morphological operators. The method requires the definition of bandwidth correction, decluttering rate and other tuning coefficients. The results for San Francisco data are displayed in Fig. 15.

Appendix B. Supplementary data

Supplementary material related to this article can be found online at http://dx.doi.org/10.1016/j.spasta.2013.11.003.

References

Allard, D., Fraley, C., 1997. Nonparametric maximum likelihood estimation of features in spatial point processes using Voronoi tessellation. J. Amer. Statist. Assoc. 92 (440), 1485–1493.

Bengio, Y., Larochelle, H., Vincent, P., 2006. Non-local manifold Parzen windows. In: Weiss, Y., SchÏkopf, B., Platt, J. (Eds.), Advances in Neural Information Processing Systems. Vol. 18. MIT Press, pp. 115–122.

Botev, Z.I., Grotowski, J.F., Kroese, D.P., 2010. Kernel density estimation via diffusion. Ann. Statist. 38 (5), 2916–2957.

Carreira-Perpiñán, M.Á., 2006. Fast nonparametric clustering with Gaussian blurring mean-shift. In: 23rd International Conference on Machine Learning, ICML 2006, pp. 153–160.

Carreira-Perpiñán, M.Á., 2007. Gaussian mean shift is an EM algorithm. IEEE Trans. Pattern Anal. Mach. Intell. 29 (5), 767–776.

Carreira-Perpiñán, M.Á., Lu, Z., 2008. Dimensionality reduction by unsupervised regression. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2008, pp. 1–8.

Chacón, J.E., Duong, T., Wand, M.P., 2011. Asymptotics for general multivariate kernel density derivative estimators. Statist. Sinica 21 (2), 807–840.

Cheng, Y., 1995. Mean shift, mode seeking, and clustering. IEEE Trans. Pattern Anal. Mach. Intell. 17 (8), 790–799.

Comaniciu, D., Meer, P., 2002. Mean shift: a robust approach toward feature space analysis. IEEE Trans. Pattern Anal. Mach. Intell. 24 (5), 603–619.

Delicado, P., 2001. Another look at principal curves and surfaces. J. Multivariate Anal. 77 (1), 84–116.

Eberly, D., 1996. Ridges in Image and Data Analysis. Kluwer.

Einbeck, J., Tutz, G., Evers, L., 2005. Local principal curves. Stat. Comput. 15 (4), 301–313.

Einbeck, J., Evers, L., Powell, B., 2010. Data compression and regression, through local principal curves and surfaces. Int. J. Neural Syst. 20 (3), 177–192.

Genovese, C.R., Perone-Pacifico, M., Verdinelli, I., Wasserman, L., 2012. The geometry of nonparametric filament estimation. J. Amer. Statist. Assoc. 107 (498), 788–799.

Grillenzoni, C., 2005. Non-parametric smoothing of spatio-temporal point processes. J. Statist. Plann. Inference 128 (1), 61–78.

Grillenzoni, C., 2006. Sequential kernel estimation of the conditional intensity of nonstationary point processes. Stat. Inference Stoch. Process. 9 (2), 135–160.

Grillenzoni, C., 2008. Design of kernel M-smoothers for spatial data. Stat. Methodol. 5 (3), 220–237.

Hastie, T., Stuetzle, W., 1989. Principal curves. J. Amer. Statist. Assoc. 84 (406), 502–516.

Hastie, T., Tibshirani, R., Friedman, J., 2009. The Elements of Statistical Learning: Data Mining, Inference and Prediction, second ed. Springer.

Havskov, J., Ottemöller, L., 2010. Routine Data Processing in Earthquake Seismology. Springer.

Hill, B.J., Kendall, W.S., Thönnes, E., 2012. Fibre-generated point processes and fields of orientations. Ann. Appl. Stat. 6 (3), 994–1020.

Kégl, B., Kryzak, A., 2002. Piecewise linear skeletonization using principal curves. IEEE Trans. Pattern Anal. Mach. Intell. 24 (1), 59–74.

Meinicke, P., Klanke, S., Memisevic, R., Ritter, H., 2005. Principal surfaces from unsupervised kernel regression. IEEE Trans. Pattern Anal. Mach. Intell. 27 (9), 1379–1391.

Meyer, C.D., 2000. Matrix Analysis and Applied Linear Algebra. SIAM Books.

Ozertem, U., Erdogmus, D., 2007. Nonparametric snakes. IEEE Trans. Image Process. 16 (9), 2361–2368.

Ozertem, U., Erdogmus, D., 2011. Locally defined principal curves and surfaces. J. Mach. Learn. Res. 12, 1249–1286.

Silverman, B.W., 1986. Density Estimation for Statistical Data Analysis. Chapman & Hall.

Stanford, D.C., Raftery, A.E., 2000. Finding curvilinear features in spatial point patterns: principal curve clustering with noise. IEEE Trans. Pattern Anal. Mach. Intell. 22 (6), 601–609.

Wang, W., Carreira-Perpiñán, M.Á., 2010. Manifold blurring mean shift algorithms for manifold denoising. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2010, 1759–1766.