Finding Curvilinear Features in Spatial Point Patterns: Principal Curve Clustering with Noise

Derek C. Stanford and Adrian E. Raftery

Abstract—Clustering about principal curves combines parametric modeling of noise with nonparametric modeling of feature shape. This is useful for detecting curvilinear features in spatial point patterns, with or without background noise. Applications include the detection of curvilinear minefields from reconnaissance images, some of the points in which represent false detections, and the detection of seismic faults from earthquake catalogs. Our algorithm for principal curve clustering is in two steps: The first is hierarchical and agglomerative (HPCC) and the second consists of iterative relocation based on the Classification EM algorithm (CEM-PCC). HPCC is used to combine potential feature clusters, while CEM-PCC refines the results and deals with background noise. It is important to have a good starting point for the algorithm: This can be found manually or automatically using, for example, nearest neighbor clutter removal or model-based clustering. We choose the number of features and the amount of smoothing simultaneously, using approximate Bayes factors.

Index Terms—Bayes factor, BIC, CEM algorithm, earthquake, EM algorithm, Hough transform, model-based clustering, smoothing, spatial point process, visual defect metrology.

1 Introduction

We would like to detect curvilinear features in spatial point processes automatically. We must deal with two kinds of noise: background noise, in the form of observed points which are not part of the features, and feature noise, which is the deviation of observed feature points from an underlying "true" feature curve. One such problem is the detection of curvilinear minefields in aerial reconnaissance images. Fig. 1a is a simulation of such an image and Fig. 1b shows the features detected by our method.

In Section 2, we give some background on principal curves and introduce our probability model and clustering algorithm. Section 2.3 presents our method for clustering on open principal curves and Section 2.4 describes our use of approximate Bayes factors to choose the number of features and the amount of smoothing simultaneously and automatically. Initialization methods, including our hierarchical principal curve clustering (HPCC) algorithm, are discussed in Section 3. Examples are presented in Section 4 and, in Section 5, we discuss other approaches and areas of further work.

2 Model, Estimation, and Inference

2.1 Principal Curves

A principal curve is a smooth, curvilinear summary of n-dimensional data; it is a nonlinear generalization of the first principal component line. Principal curves were introduced by Hastie and Stuetzle [14] and discussed in the clustering context by Banfield and Raftery [3]. The curve f is a principal curve of h if

$$E(X|\lambda_f(X) = \lambda) = f(\lambda) \qquad (1)$$

for almost all λ, where X is a random vector with density h in R^n and λ_f is the function which projects points in R^n orthogonally onto f. When this holds, f is also said to be self-consistent for h. A principal curve f is parameterized by λ, the arc length along the curve.

The algorithm for fitting a principal curve from data involves iteratively applying the definition (1), where the conditional expectation is replaced by a scatterplot smoother. The choice of the smoothing parameter is discussed in Section 2.4. Each data point, x_j, has an associated projection point $f(\lambda_j)$ on the curve, which is the point on the curve closest to x_j (see Fig. 2). The line segment from x_j to $f(\lambda_j)$ is orthogonal to the curve at $f(\lambda_j)$ unless $f(\lambda_j)$ is an endpoint of the curve. Bias correction for closed principal curves [3] can also be extended to the open principal curves that we use here.

2.2 Probability Model

We model a noisy spatial point process by making distributional assumptions about the background noise and the feature noise. Suppose that X is a set of observations, $x_1 \ldots x_N$, and C is a partition consisting of clusters, $C_0, C_1 \ldots C_K$, where cluster C_j contains N_j points. The noise cluster is denoted by C_0; we assume that the background noise is uniformly distributed over the region of the image (this is equivalent to Poisson background noise). We assume that the feature points are distributed uniformly along the true underlying feature; that is, their projections onto the feature's principal curve are drawn randomly from a uniform distribution $U(0, \nu_j)$, where ν_j

- *D.C. Stanford is with Mathsoft Incorporated, 1700 Westlake Ave. North, Seattle, WA 98109. E-mail: stanford@statsci.com.*
- *A.E. Raftery is with the Department of Statistics, Universtiy of Washington, Box 354322, Seattle, WA 98195-4322. E-Mail: raftery@stat.washington.edu.*

Manuscript received 19 Aug. 1997; revised 3 Sept. 1999; accepted 13 Dec. 1999.
Recommended for acceptance by I. Sethi.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number 107739.

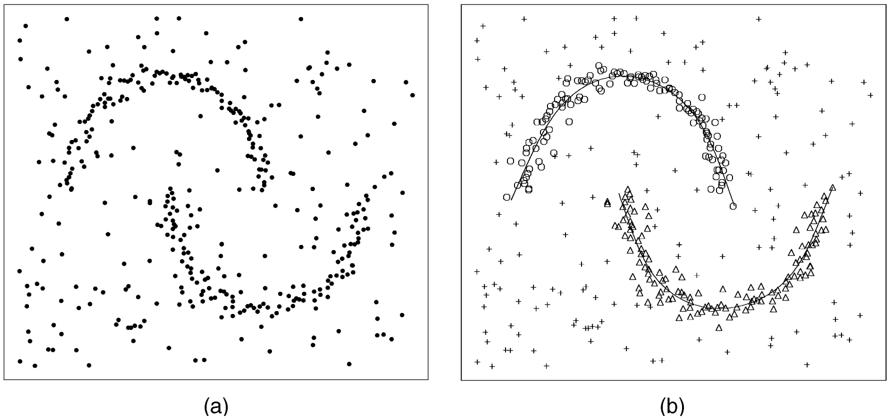


Fig. 1. (a) Simulated minefield with noise. (b) Final result.

is the length of the jth curve. We assume that the feature points are distributed normally about the true underlying feature, with mean zero and variance σ_j^2. Distance about the curve is the orthogonal distance from a point to the curve; if the point projects to an endpoint of the curve, it is simply the distance from the point to the curve endpoint. The $(K+1)$ clusters are combined in a mixture model and we denote the unconditional probability of belonging to the jth feature by π_j $(j = 0, 1 \dots K)$.

Let θ denote the entire set of parameters, π_0 and $\{\nu_j, \sigma_j^2, \pi_j : j = 1, \dots, K\}$, not including the curves themselves. Then, the likelihood is

$$L(X|\theta) = \prod_{i=1}^{N} L(x_i|\theta),$$

where $L(x_i|\theta) = \sum_{j=0}^{K} \pi_j L(x_i|\theta, x_i \in C_j)$. For feature clusters,

$$L(x_i|\theta, x_i \in C_j) = \left[\frac{1}{\nu_j}\right]\left[\frac{1}{\sqrt{2\pi}\sigma_j}\exp\left(\frac{-||x_i - f(\lambda_{ij})||^2}{2\sigma_j^2}\right)\right],$$

where $||x_i - f(\lambda_{ij})||$ is the Euclidean distance from the point x_i to its projection point $f(\lambda_{ij})$ on curve j. For the noise cluster,

$$L(x_i|\theta, x_i \in C_j) = \frac{1}{Area},$$

where $Area$ is the area of the image.

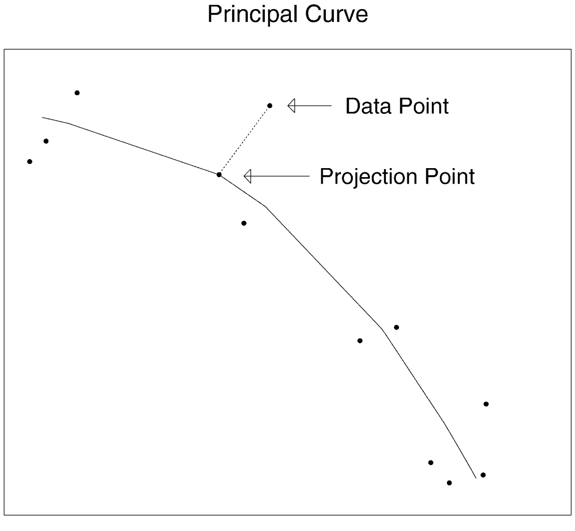


Fig. 2. Principal curve example.

2.3 Estimation: The CEM-PCC Algorithm

The CEM-PCC algorithm refines a given clustering by using the Classification EM algorithm [6], which is a version of the well-known EM algorithm [10], and the probability model of Section 2.2. We start with an initial clustering from the methods discussed in Section 3.

Overview of the CEM-PCC algorithm:

1. Begin with an initial clustering (features and noise).
2. (M-step) Conditional on the current clustering, fit a principal curve to each feature cluster and then compute estimates of the parameters (ν_j, σ_j^2, and π_j).
3. (E-step) Conditional on the current curves and parameter estimates, calculate the likelihood of each point being in each cluster.
4. (Classification step) Reclassify each point into its most likely cluster.
5. Check for convergence; end or return to Step 2.

Once we have calculated the probability of each point being in each cluster based on the estimates of the parameters at the current iteration, we reclassify each point into the cluster for which it has the highest likelihood. At the end of each iteration, we compute the overall likelihood $L(X|\theta)$. This process is executed for a predetermined number of iterations, at which point we choose, as the final result, the clustering with the highest overall likelihood (CEM iterations sometimes decrease the likelihood).

We have found that it is useful to impose a lower bound on the estimate of the variance about the curve. If the variance is allowed to decrease without bound, the likelihood can grow without bound. This can be a problem when there are small clusters since the smoothing is almost able to interpolate the data points. We impose a bound based on the assumption that the data are not known with absolute precision. For instance, if we assume the data are precise to three significant digits, then we can find a lower bound on the resolution of the data and translate this to a lower bound on the variance.

2.4 Inference: Choosing the Number of Features and Their Smoothness Simultaneously

Since the number of clusters affects the overall amount of smoothing, we select the smoothing parameter and the number of clusters simultaneously. The amount of smoothing in each feature cluster is measured by the degrees of freedom (DF) used in fitting the principal curve to that cluster. We use a cubic B-spline [37] in fitting the principal curves; specifically, we use the function *principal.curve* (obtained from Statlib) which calls the Splus function *smooth.spline*. The DF of a cubic B-spline is given by the trace of the implicit smoother matrix S; S is the matrix which yields the fitted values at the observed data points ([36],[15]).

Each combination of number of features and degrees of freedom (i.e., smoothness of a feature) considered is viewed as specifying a possible model for the data, and the competing models are compared using Bayes factors [18]. The Bayes factor for comparing two models is the Bayesian solution to the hypothesis testing and model selection problem. It is the posterior odds for one model against the other given the data, i.e., the ratio of their posterior probabilities, when their prior model probabilities are equal. Bayes factors are appealing in the present context because, unlike standard statistical significance tests, they allow us validly to compare the nonnested models that we consider and they also permit valid comparison of multiple models. Also, if viewed as the basis for a significance test, the Bayes factor automatically chooses the significance level so as to minimize the total error rate of the test (i.e., the sum of the Type I and Type II errors) [18]. This is in contrast to common practice which sets the significance level at some conventional value such as .05 and allows one to balance power and significance. This can be particularly useful in very large data sets. Bayes factors, unlike significance tests, are consistent model selection criteria in the sense that if the true model is among those considered, it will be selected by the Bayes factors with a probability that approaches one as the sample size increases.

We approximate the Bayes factor using the Bayesian Information Criterion (BIC, [32]). In regular models, the difference between the BIC values for two models is approximately equal to twice the log Bayes factor when unit information priors for the model parameters are used [19]. These are priors that contain about the same amount of information as a single typical observation. Model selection based on BIC provides (asymptotically) consistent estimators of the probability distribution generating a data set [31]. This approach has been found to work well in practice for mixture models and other model-based clustering problems [8], [11] ,[31].

The BIC for a model with K features and background noise is defined by:

$$BIC = 2\log(L(X|\theta)) - M \cdot \log(N),$$

where $M = K(DF + 2) + K + 1$ is the number of parameters. The number of feature clusters is K; for each feature cluster, we estimate ν_j and σ_j, and we fit a curve using DF degrees of freedom. The mixing proportions add K parameters and the estimate of the image area used in the noise density is one more parameter.

The larger the BIC, the more the model is favored by the data. Conventionally, differences of 2-6 between BIC values for models represent positive evidence, differences of 6-10 correspond to strong evidence, while differences greater than 10 indicate very strong evidence [18].

3 Initialization

3.1 Denoising and Initial Clustering

The performance of the CEM-PCC algorithm can be sensitive to the starting value, so it is important to have a good starting value. The initial clustering used to obtain this should accomplish two objectives: separate the feature points from background noise and provide an initial clustering of the feature points. The first of these can be done by a human or by various automatic methods, such as nonparametric maximum likelihood, using the Voronoi tesselation [1] or Kth nearest neighbor clutter removal [5]. This step does not need to be perfect since CEM-PCC will examine the noise points to determine if they should be included in the features and vice versa.

Once the noise points have been removed, we need an initial clustering of the feature points so that a curve can be fit to each cluster. We recommend that there be at least seven points in each cluster; when there are fewer than seven, we fit a principal component line instead of a curve. The feature points can be clustered using model-based clustering, as implemented in the MCLUST software ([4], [8], [11]). A simpler method is to fit a minimum spanning tree to the feature points and cut the longest edges, which will work well if the main clusters are well-separated ([30], [38]).

3.2 Hierarchical Principal Curve Clustering (HPCC)

Clustering on *closed* principal curves was introduced by Banfield and Raftery [3]. Their clustering criterion (V^*) is based on a weighted sum of the squared distances about the curve and the squared distances along the curve and they state that it is optimal when the data points are normally distributed about the curve (conditional on the estimated curves and assuming that α is chosen properly). It is defined by

$$V^* = V_{About} + \alpha V_{Along},$$

where

$$V_{About} = \sum_{j=1}^{N} ||x_j - f(\lambda_j)||^2,$$

$$V_{Along} = \frac{1}{2}\sum_{j=1}^{N} ||\epsilon_j - \bar{\epsilon}||^2,$$

and

$$\epsilon_j = f(\lambda_j) - f(\lambda_{j+1}).$$

The V_{About} term measures the spread of observations about the curve (in orthogonal distance to the curve), while the V_{Along} term measures the variance in arc length distances

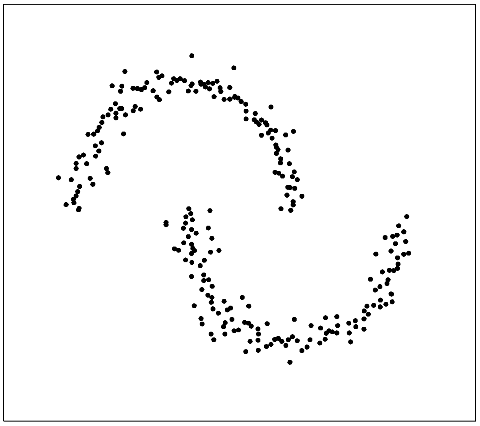


Fig. 3. Two part curvilinear minefield after denoising using nearest neighbor cleaning.

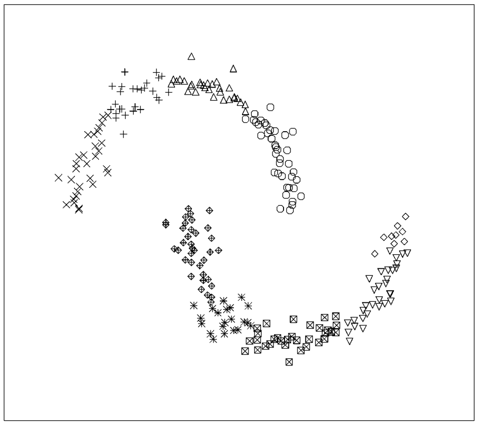


Fig. 4. Initial clustering of two part curvilinear minefield using MCLUST. There are nine clusters.

between projection points on the curve. Minimizing $\sum V^*$ (where the sum is over all clusters) will lead to clusters with points regularly spaced along the curve and tightly grouped around it. Large values of α will cause the algorithm to avoid clusters with gaps, while small values will favor thinner clusters. Clustering stops when merging clusters would lead to an increase in $\sum V^*$.

We extend the method to *open* principal curves by changing V_{Along} so that the sum goes only to $(N-1)$ instead of to N. This is because the closed curves could wrap around, whereas the open curve stops at its end points.

Overview of HPCC:

1. Make a first estimate of the noise points and remove them.
2. Form an initial clustering with at least seven points in each cluster.
3. Fit a principal curve to each cluster.
4. Calculate $\sum V^*$ for each possible merge.
5. Perform the merge which leads to the lowest $\sum V^*$.
6. Keep merging until the desired number of clusters is reached.

Deciding when to stop clustering is more difficult for open curves than for closed curves. In the closed curve case, clustering stops when any merge would lead to an increase in $\sum V^*$ [3]. For open curves, this method leads to an overfitting problem in which we end up with too many clusters. V^* can be made arbitrarily close to zero by increasing the number of clusters. We overcame this problem by using approximate Bayes factors (Section 2.4).

4 Examples

4.1 A Simulated Two-Part Curvilinear Minefield

The simulated minefield shown in Fig. 1a contains 100 points in each of the two curves and 200 points of background noise (400 points total). This simulation was created using offset semicircles as the true underlying features, and background noise was generated uniformly over the image area. Note that some of the background noise points will fall inside the regions of feature points; these noise points will be indistinguishable from feature points.

The first step is to separate the features from the noise, which we did using the ninth nearest-neighbor denoising [5]; the resulting feature points are shown in Fig. 3. We then used MCLUST [4] to provide an initial clustering into nine clusters; this is shown in Fig. 4. We used nine clusters for the initial clustering because this is the largest number of clusters for which MCLUST returns a clustering in which each cluster has at least seven points. HPCC was applied to obtain two clusters, shown in Fig. 5. The noise points were then returned to the image with the HPCC clustering and CEM-PCC was used to refine the clustering. The final result is shown in Fig. 1b.

Table 1 shows the BIC values for one to three features with a variety of DF values. The BIC is maximized for two features with five DF. The approximate Bayes factors identified the correct number of features quite decisively in this example.

4.2 A Simulated Curvilinear Minefield

We simulated a curvilinear minefield by generating 100 points uniformly along a sine wave and then adding Gaussian jitter; 100 background noise points are then generated uniformly over the image region. The data are shown in Fig. 6. The BIC results from our method are given in Table 2. Here, there is just one feature and the BIC strongly favors one feature with 8 degrees of freedom.

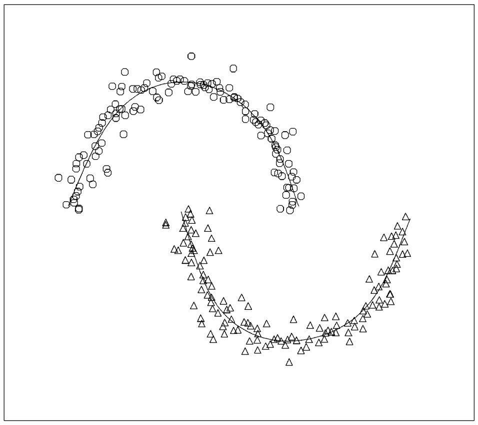


Fig. 5. HPCC applied to the two-part curvilinear minefield.

TABLE 1
BIC Results for the Two-Part Curvilinear Minefield

	BIC			
DF	0 Features	1 Feature	2 Features	3 Features
2	-1984	-1880	-1846	-1745
3	-1984	-1850	-1748	-1721
4	-1984	-1861	-1648	**-1648**
5	-1984	-1845	**-1628**	-1658
6	-1984	-1803	-1632	-1670
7	-1984	-1761	-1641	-1685
8	-1984	-1726	-1643	-1693
9	-1984	-1702	-1648	-1703
10	-1984	-1692	-1660	-1718
11	-1984	-1689	-1671	-1735
12	-1984	-1689	-1688	-1749
13	-1984	**-1680**	-1703	-1770
14	-1984	-1721	-1718	-1792
15	-1984	-1748	-1727	-1810
16	-1984	-1777	-1733	-1807
17	-1984	-1755	-1739	-1827

Figs. 7, 8, and 9 show the steps in our method. The HPCC step is not shown since there is only one feature cluster (all the points in Fig. 8 are classified into one cluster).

This is a rather difficult example for two reasons: First, there are as many noise points as feature points. Second, generating points uniformly along a sine wave yields dense regions at the peak and trough of the wave, while points are relatively sparse in between. Visually, the correct number of features are not unambiguously clear from Fig. 6, but our method finds it quite decisively.

4.3 New Madrid Seismic Region

TABLE 2
BIC Results for Simulated Sine Wave Minefield

DF	0 Features	1 Feature	2 Features	3 Features
2	-1031	-1034	-1018	-991
3	-1031	-1016	-938	-941
4	-1031	-975	-889	-908
5	-1031	-911	**-883**	-904
6	-1031	-884	-887	-901
7	-1031	-873	-894	**-901**
8	-1031	**-868**	-901	-907
9	-1031	-869	-908	-921
10	-1031	-872	-913	-929
11	-1031	-873	-919	-940
12	-1031	-877	-926	-951

Data on 219 earthquakes in the New Madrid seismic region were obtained from the Center for Earthquake Research and Information (CERI) World Wide Web site http://samwise.ceri.memphis.edu. We have included all earthquakes in the New Madrid catalog from 1974 to 1992 with a magnitude of 2.5 and above. This time period was chosen because the data collection methods were consistent; data prior to this period are available, but become sparser and less reliable as one goes back in time. The New Madrid region extends from Illinois to Arkansas: latitude 35 to 38 and longitude -91 to -88. These data are displayed in Fig. 10 and the BIC results are shown in Table 3. Figs. 11, 12, 13, and 14 show each step of our process; the final result (Fig. 14) corresponds to the parameters which yield the maximum BIC value (three features, each with 10 degrees of freedom).

This example illustrates some strengths and limitations of our method. We can see in Fig. 11 that the most striking features in this dataset are a combination of lines and blobs. While our method does a good job of picking out the curvilinear features, blobs are not very well modeled by curves, causing the rather awkward looking result for the rightmost feature.

5 Discussion

We have introduced a probability model for noisy spatial point process data with curvilinear features. We use the CEM algorithm to estimate it and classify the points, and we use approximate Bayes factors to find the number of features and the optimal amount of smoothing, simultaneously and automatically. The hierarchical principal curve clustering method of Banfield and Raftery [3] is extended to

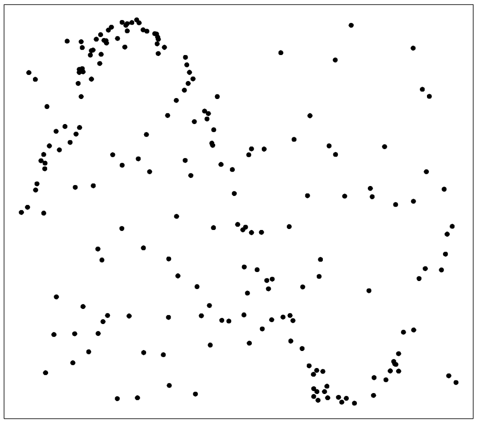


Fig. 6. Simulated curvilinear minefield.

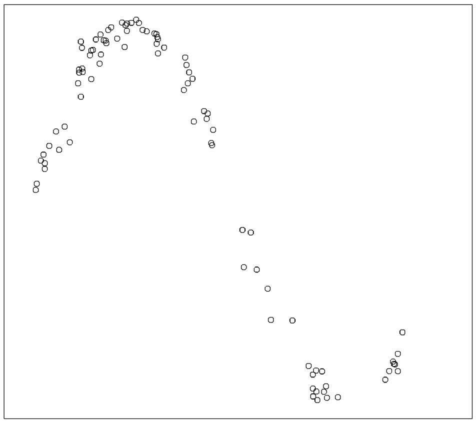


Fig. 7. Simulated curvilinear minefield after denoising using nearest-neighbor cleaning.

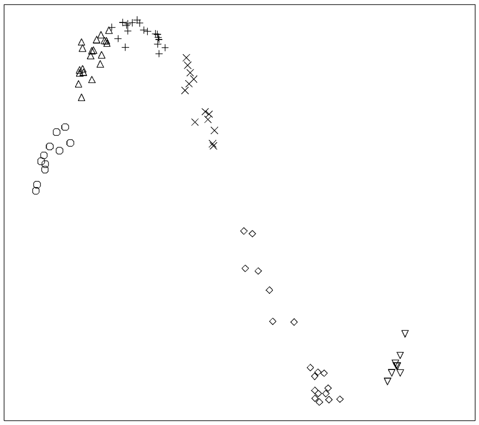


Fig. 8. Initial clustering of denoised curvilinear minefield using MCLUST. There are seven clusters.

open principal curves (HPCC) and we describe an iterative relocation method (CEM-PCC) for refining a principal curve clustering based on the Classification EM algorithm [6]. In combination with the denoising method of Byers and Raftery [5] and an initial clustering method such as MCLUST [4], we have an approach which takes noisy spatial point process data and automatically extracts curvilinear features. The method appears to work well in simulated and real examples.

One way that this kind of data may arise is from image processing. There are many methods for edge detection in images, but most of these methods yield noisy results. These edge detector results can be viewed as a point process and analyzed with principal curve clustering; this would enhance edge and boundary detection in images by reducing noise and looking for larger scale structures. Note that this requires an important additional step to convert the edge detector results into a binary image which can then be regarded as a point process. This is illustrated with ice floe images in Banfield and Raftery [3].

The Hough transform is a well-known and widely used method for fitting a parameteric curve to a point set [17], [16]. One limitation of the Hough transform is that it fits only parametric curves so that the form of the curve must be specified in advance. We would like to address the situation in which little is known about the true shape of the features in a point set, so we want to avoid assumptions about the parametric form of the features. In this paper, we use open principal curves to model underlying curves in the data. Principal curves provide a data-driven, nonparametric summary of feature shape; they are characterized by the number of degrees of freedom allowed. For example, a principal curve with 15 degrees of freedom could look like a line, an arc, a sinusoid, a spiral, or some combination of these. In order to give a parametric curve as much flexibility as a principal curve, many parameters would be needed, which would greatly increase the already large computational requirements of the Hough transform.

Preconditioning on the parameter domain is used in [13] to improve the speed of the Radon transform, a generalization of the Hough transform. Although this approach can greatly reduce the size of the parameter space, it has two drawbacks. First, several sensitivity parameters must be specified by the user. This means that the use of this method

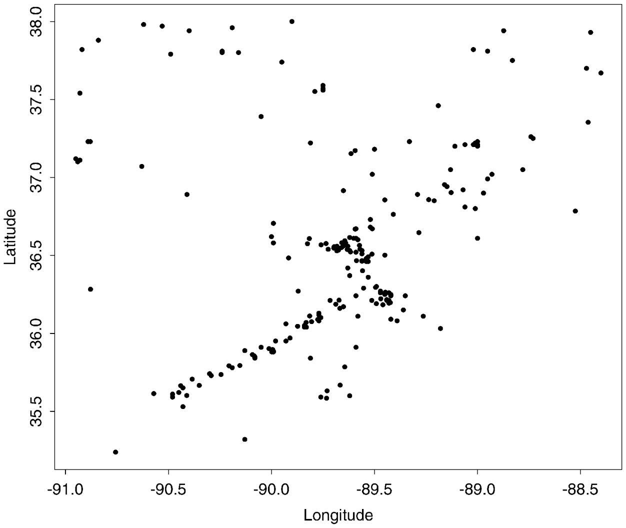


Fig. 10. New Madrid earthquakes 1974-1992.

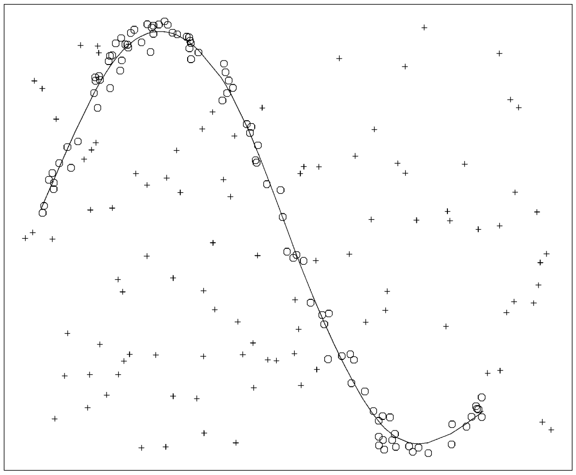


Fig. 9. CEM-PCC applied to the curvilinear minefield.

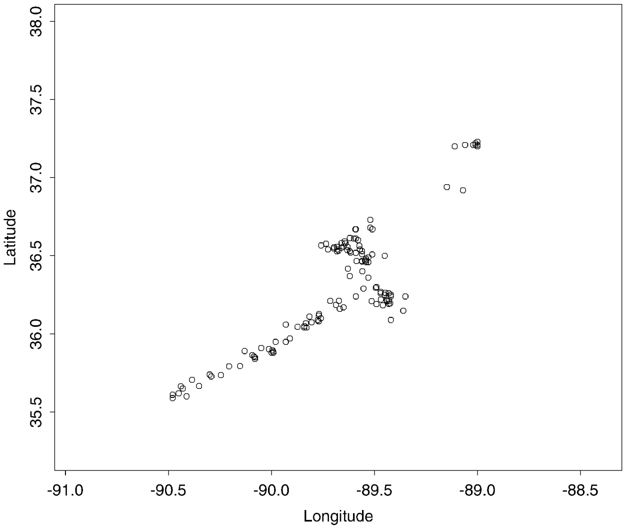


Fig. 11. New Madrid data after denoising.

TABLE 3
BIC Results for New Madrid Seismic Data

DF	0 Features	1 Feature	2 Features	3 Features	4 Features
2	-861	-548	-447	-334	-349
3	-861	-544	-403	-336	-361
4	-861	-524	-380	-335	-365
5	-861	-492	-367	-328	-360
6	-861	-458	-363	-330	-363
7	-861	-426	-362	-337	-367
8	-861	-401	**-362**	-337	-362
9	-861	-386	-363	-332	**-347**
10	-861	-378	-364	**-306**	-365
11	-861	-375	-372	-320	-383
12	-861	-372	-377	-333	-400
13	-861	-369	-381	-345	-417
14	-861	-367	-387	-355	-431
15	-861	-364	-391	-366	-446
16	-861	-363	-404	-375	-458
17	-861	-360	-408	-383	-468
18	-861	-361	-410	-393	-482
19	-861	-362	-414	-404	-495
20	-861	**-359**	-420	-410	-507
21	-861	-361	-418	-416	-518
22	-861	-363	-424	-426	-532
23	-861	-365	-424	-430	-541
24	-861	-371	-426	-439	-554

needs to be interactive, with the user trying various parameter values until a good result is obtained. Second, only parametric curves are allowed. Our principal curve clustering method uses BIC to automatically choose the number of features and the amount of smoothing; the curve shape is estimated adaptively and nonparametrically, so there is no need to search over a large parameter space.

A curve detection method which makes no parametric assumptions about the curve shape is given in [34]. Candidate points for the curves underlying the features are detected using local differential operators. These points are then linked into curves; a user-specified threshold is used to determine which candidates are used. The curves are subsequently modeled as two edges with an interior region. This allows determination of curve width and a bias reduction step is used to improve the result. Because the curves detected by this method are nonparameteric, they are much more general than curves which can be fitted

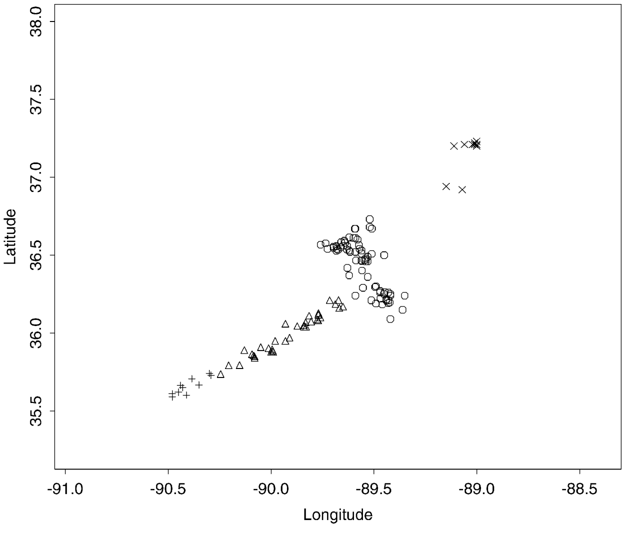


Fig. 12. Initial clustering of denoised New Madrid earthquake data using MCLUST. There are four clusters.

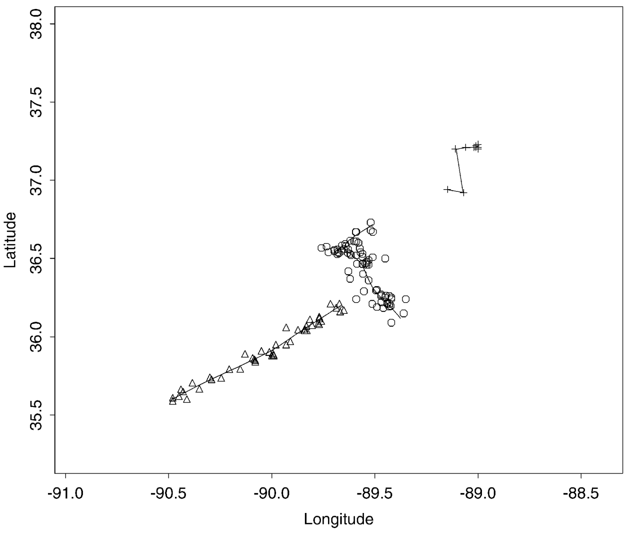


Fig. 13. HPCC applied to the New Madrid data.

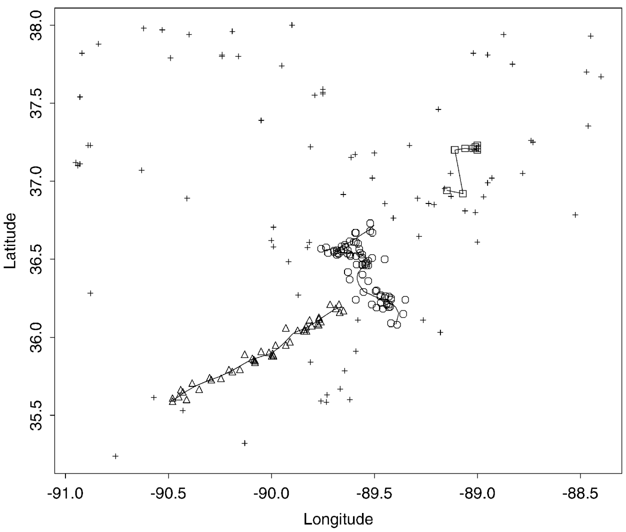


Fig. 14. CEM-PCC applied to the New Madrid data.

using a Hough transform type of approach. The examples in [34] show that the curves fit the data quite well, but it is still up to the user to interactively choose the sensitivity parameter. The method does not include a formal way to choose the number of features.

Spatial point process data arise in visual defect metrology and the Hough transform has been used previously to detect linear features in these data [7]. In this application of the Hough transform, several parameters and thresholds must be specified in advance by the user. Although users experienced with this technique may well be able to find reasonable values for all of the needed parameters, it seems more satisfactory to estimate parameters from the data. Principal curve clustering allows automatic detection of both linear and nonlinear features without the need for ad hoc parameter specification.

The Kohonen self-organizing feature map (SOFM) is another data-driven approach to feature detection ([2], [22], [28], [23]). Neither principal curve clustering nor the SOFM approach requires prior specification of feature shape and both algorithms are hierarchical in nature. Like principal curves, the SOFM can be combined with further clustering methods to produce a more powerful clustering algorithm [28]. Examination of the use of the SOFM in place of principal curves or vice versa is an interesting area for further research.

Tibshirani [35] proposes an alternate definition of principal curves based on mixture models and a new algorithm for fitting principal curves based on the EM algorithm. It is argued that this definition avoids the bias problems inherent in the approach of Hastie and Stuetzle [14], and an example is presented showing that these principal curves can be different in practice from curves of the Hastie and Stuetzle type. Other definitions of principal curves are given by Delicado [9] and by Kegl et al. ([20], [21]). It would be of interest to see what effect these alternate definitions would have on our results.

The examples we have presented in this paper consist of two-dimensional point patterns, but our methods can be generalized to higher dimensions. Principal curves could be used in higher dimensions or our approach could be modified to use a different model as the basis for features, such as principal surfaces [14] or adaptive principal surfaces [26].

Many variations of the EM algorithm are available. Green [12] introduced the One Step Late (OSL) algorithm, a version of EM for use with penalized likelihoods. Silverman et al. [33] added a smoothing step to the EM algorithm and similarities between this approach and maximum penalized likelihood were discussed by Nychka [29]. Lu [27] replaced the M-step with a smoothing step. Theoretical properties of smoothing in the EM algorithm were discussed by Latham and Anderssen [24] and Latham [25]. Our use of principal curves in CEM-PCC is similar to these ideas of smoothing. Although the curves themselves are not smoothed across CEM iterations, the curves can be viewed as smoothing the pointwise likelihoods and thus indirectly smoothing the parameter estimates.

In addition to approximate Bayes factors, we explored cross-validation as a method for choosing the number of clusters and amount of smoothing. This involves iteratively leaving out one data point, recomputing the entire clustering, and then calculating the likelihood for the left out point. We found that the results are quite similar to the Bayes factor results, but that the cross-validation approach involves much more computation.

Because the HPCC algorithm is hierarchical, the result with K clusters and D degrees of freedom can be used as a starting point in computing the result for $K - 1$ clusters and D degrees of freedom. The results cannot be reused for a different number of degrees of freedom. The computational complexity of the HPCC step depends on the computational complexity of the algorithm used to fit principal curves. As mentioned above, there are several different definitions of principal curves and multiple fitting algorithms as well. On the other hand, the CEM-PCC algorithm is iterative; determining the conditions necessary to guarantee convergence is a topic for further study.

Our model assumes that the principal curve underlying each feature has the same smoothness. This may not always be realistic and it would be possible and worthwhile to relax this assumption. Furthermore, as seems to be the case in the New Madrid data set, one or more of the features might be circular (or hyperspherical in higher dimensions), concentrated about a point rather than a curve. Extending the method to accommodate this possibility explicitly would also be worthwhile.

Splus source files for HPCC and CEM-PCC are available at http://www.isomorphic.org/software/princlust.html. Statlib has Splus functions available for fitting principal curves and Kth nearest neighbor denoising: http://lib.stat.cmu.edu/S/principal.curve and http://lib.stat.cmu.edu/S/nnclean.

Acknowledgments

The authors would like to thank Simon Byers, Gilles Celeux, and Christian Posse for their helpful discussions. This research was supported by the U.S. Office of Naval Research grants N00014-96-10192 and N00014-96-1-0330.

References

[1] D. Allard and C. Fraley, "Nonparametric Maximum Likelihood Estimation of Features in Spatial Point Processes Using Voronoi Tesselation," *J. Am. Statistical Assoc.,* vol. 92, pp. 1,485-1,493, 1997.

[2] C. Ambroise and G. Govaert, "Constrained Clustering and Kohonen Self-Organizing Maps," *J. Classification,* vol. 13, pp. 299-313, 1996.

[3] J.D. Banfield and A.E. Raftery, "Ice Floe Identification in Satellite Images Using Mathematical Morphology and Clustering about Principal Curves," *J. Am. Statistical Assoc.,* vol. 87, pp. 7-16, 1992.

[4] J.D. Banfield and A.E. Raftery, "Model-Based Gaussian and Non-Gaussian Clustering," *Biometrics,* vol. 49, pp. 803-821, 1993.

[5] S.D. Byers and A.E. Raftery, "Nearest Neighbor Clutter Removal for Estimating Features in Spatial Point Processes," *J. Am. Statistical Assoc.,* vol. 93, pp. 557-584, 1998.

[6] G. Celeux and G. Govaert, "A Classification EM Algorithm and Two Stochastic Versions," *Computational Statistics and Data Analysis,* vol. 14, pp. 315-332, 1992.

[7] S. Cunningham and S. MacKinnon, "Statistical Methods for Visual Defect Metrology," *IEEE Trans. Semiconductor Manufacturing,* vol. 11, pp. 48-53, 1998.

[8] A. Dasgupta and A.E. Raftery, "Detecting Features in Spatial Point Processes with Clutter via Model-Based Clustering," *J. Am. Statistical Assoc.,* vol. 93, pp. 294-302, 1998.

[9] P. Delicado, "Another Look at Principal Curves and Surfaces," Working Paper 309, Department d'Economia i Empresa, Universitat Pompeu Fabra, 1998.

[10] A. Dempster, N. Laird, and D. Rubin, "Maximum Likelihood from Incomplete Data via the EM Algorithm (with Discussion)," *J. Royal Statistical Soc., Series B,* vol. 39, pp. 1-38, 1977.

[11] C. Fraley and A.E. Raftery, "How Many Clusters? Which Clustering Method?—Answers via Model-Based Cluster Analysis," *Computer J.,* vol. 41, pp. 578-588, 1998.

[12] P. Green, "On the Use of the EM Algorithm for Penalized Likelihood Estimation," *J. Royal Statistical Soc., Series B,* vol. 52, pp. 443-452, 1990.

[13] K.V. Hansen and P.A. Toft, "Fast Curve Estimation Using Preconditioned Generalized Radon Transform," *IEEE Trans. Image Processing,* vol. 5, pp. 1,651-1,661, 1996.

[14] T. Hastie and W. Stuetzle, "Principal Curves," *J. Am. Statistical Assoc.,* vol. 84, pp. 502-516, 1989.

[15] T. Hastie and R. Tibshirani, *Generalized Additive Models.* New York: Chapman and Hall 1990.

[16] P.V.C. Hough, *A Method and Means for Recognizing Complex Patterns,* U.S. Patent 3,069,654, 1962.

[17] J. Illingworth and J. Kittler, "A Survey of the Hough Transform," *Computer Vision, Graphics, and Image Processing,* vol. 44, pp. 87-116, 1988.

[18] R.E. Kass and A.E. Raftery, "Bayes Factors," *J. Am. Statistical Assoc.,* vol. 90, pp. 773-795, 1995.

[19] R.E. Kass and L. Wasserman, "A Reference Bayesian Test for Nested Hypotheses and Its Relationship to the Schwarz Criterion," *J. Am. Statistical Assoc.,* vol. 90, pp. 928-934, 1995.

[20] B. Kegl, A. Krzyzak, T. Linder, and K. Zeger, "Principal Curves: Learning and Convergence," *Proc. IEEE Int'l Symp. Information Theory,* p. 387, 1998.

[21] B. Kegl, A. Krzyzak, T. Linder, and K. Zeger, "A Polygonal Line Algorithm for Constructing Principal Curves," *Proc. Neural Information Processing Systems, NIPS '98,* 1998.

[22] T. Kohonen, "Self-Organized Formation of Topologically Correct Feature Maps," *Biological Cybernetics,* vol. 43, pp. 59-69, 1982.

[23] T. Kohonen, *Self-Organizing Maps.* Berlin: Springer-Verlag, 1995.

[24] G. Latham and R. Anderssen, "Assessing Quantification for the EM Agorithm," *Linear Algebra and Its Applications,* vol. 210, pp. 89-122, 1994.

[25] G. Latham, "Existence of EMS Solutions and A-Priori Estimates," *SIAM J. Matrix Analysis and Applications,* vol. 16, pp. 943-953, 1995.

[26] M. LeBlanc and R. Tibshirani, "Adaptive Principal Surfaces," *J. Am. Statistical Assoc.,* vol. 89, pp. 53-64, 1994.

[27] W. Lu, "The Expectation-Smoothing Approach for Indirect Curve Estimation," *ASA Proc. Statistical Computing Section,* pp. 57-62, 1995.

[28] F. Murtagh, "Interpreting the Kohonen Self-Organization Feature Map Using Contiguity Constrained Clustering," *Pattern Recognition Letters,* vol. 16, pp. 399-408, 1995.

[29] D. Nychka, "Some Properties of Adding a Smoothing Step to the EM Algorithm," *Statistics and Probability Letters,* vol. 9, pp. 187-193, 1990.

[30] R. Prim, "Shortest Connection Networks and Some Generalizations," *Bell System Technical J.,* pp. 1,389-1,401, 1957.

[31] K. Roeder and L. Wasserman, "Practical Bayesian Density Estimation Using Mixtures of Normals," *J. Am. Statistical Assoc.,* vol. 92, pp. 894-902, 1997.

[32] G. Schwarz, "Estimating the Dimension of a Model," *Annals of Statistics,* vol. 6, pp. 461-464, 1978.

[33] B. Silverman, M. Jones, J. Wilson, and D. Nychka, "A Smoothed EM Approach to Indirect Estimation Problems, with Particular Reference to Stereology and Emission Tomography (with Discussion)," *J. Royal Statistical Soc., Series B,* vol. 52, pp. 271-324, 1990.

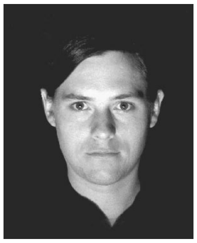
[34] C. Steger, "An Unbiased Detector of Curvilinear Structures," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 20, pp. 113-125, 1998.

[35] R. Tibshirani, "Principal Curves Revisited," *Statistics and Computing,* vol. 2, pp. 183-190, 1992.

[36] R. Tibshirani and T. Hastie, "Local Likelihood Estimation," *J. Am. Statistical Assoc.,* vol. 82, pp. 559-568, 1987.

[37] S. Wold, "Spline Functions in Data Analysis," *Technometrics,* vol. 16, pp. 1-11, 1974.

[38] C. Zahn, "Graph-Theoretical Methods for Detecting and Describing Gestalt Structures," *IEEE Trans. Computers,* vol. 20, no. 1, pp. 68-86, Jan. 1971.



Derek C. Stanford received the BS and MS degrees from Harvey Mudd College in 1993 and the PhD degree from the University of Washington in 1999. From 1996 to 1998, he worked as a statistical consultant in both the University of Washington Department of Ophthalmology and the University of Washington Division of Pulmonary and Critical Care. In 1999, he joined Mathsoft, Incorporated, as a research scientist, and he now pursues interests in wavelets, Markov models, statistical image analysis, and automatic modeling methods for large databases.

Adrian E. Raftery received the MA and MSc degrees from Trinity College, Dublin, Ireland, in 1976 and 1977, and his doctorate from the University of Paris 6 in 1980. He is a professor of statistics and sociology at the University of Washington, Seattle. He is currently the coordinating editor of the *Journal of the American Statistical Association* and the director of the Center for Statistics and the Social Sciences. His research focuses on Bayesian statistics, model-based clustering, spatial point processes, image analysis, and the statistical analysis of deterministic models.