12-1999

Self-Consistency Algorithms

Thaddeus Tarpey
Wright State University - Main Campus, thaddeus.tarpey@wright.edu

Self-Consistency Algorithms

Thaddeus TARPEY

The k-means algorithm and the principal curve algorithm are special cases of a self-consistency algorithm. A general self-consistency algorithm is described and results are provided describing the behavior of the algorithm for theoretical distributions, in particular elliptical distributions. The results are used to contrast the behavior of the algorithms when applied to a theoretical model and when applied to finite datasets from the model. The algorithm is also used to determine principal loops for the bivariate normal distribution.

Key Words: Conditional expectation; Elliptical distributions; k-means algorithm; Principal components; Principal curves; Principal loops; Principal points; Self-consistent points.

DEDICATION

This article is dedicated to the memory of the author's former Ph.D. advisor, Bernhard Flury. Professor Flury died July 6, 1999, in a tragic accident in the Dolomite Mountains in Italy. He made significant contributions to multivariate analysis with publications of numerous articles and three books. The following article is about principal points and self-consistency, topics to which the author was introduced by Bernhard Flury.

—*The Author, the Editor*

1. INTRODUCTION

Tarpey and Flury (1996), hereafter known as TF,
defined a random vector $\mathbf{Y}$ to be self-consistent for a random vector $\mathbf{X}$ if

$$\mathcal{E}[\mathbf{X}|\mathbf{Y}] = \mathbf{Y} \text{ a.s.} \tag{1.1}$$

The term self-consistency was inspired by the self-consistent curves (or principal curves) defined by Hastie and Stuetzle (1989), hereafter known as HS. Typically, self-consistency is used to find a simpler approximation to a given distribution. For example, for multivariate normal populations, a principal component axis provides a lower dimensional self-consistent approximation to the distribution. A self-consistent curve is a nonlinear

Thaddeus Tarpey is Assistant Professor, Wright State University Department of Mathematics and Statistics Dayton, OH 45435 E-mail: ttarpey@paladin.wright.edu).

Journal of Computational and Graphical Statistics, Volume 8, Number 4, Pages 889–905

generalization of a principal component axis having the self-consistency property that each point on the curve is the average of all observations projecting onto it. Principal points (Flury 1990) provide another example of self-consistency where the goal is to optimally approximate a distribution by a finite set of representative points.

Finding a self-consistent approximation to a theoretical distribution can be very difficult. Algorithms have been proposed to determine self-consistent approximations from a finite set of data. In Section 2, we take a look at one of these algorithms: the k-means algorithm. In Section 3, we define a general self-consistency algorithm that can be used to find self-consistent approximations to theoretical distributions. The behavior of the self-consistency algorithm for elliptical distributions is described in Section 4. Principal components and principal curves are examined in terms of the self-consistency algorithm in Section 5. In Section 6, the self-consistency algorithm is used to find principal loops (Duchamp and Stuetzle 1993, 1996) for bivariate normal distributions. We conclude the article in Section 7.

2. k-MEANS ALGORITHM

In this section we consider the well-known k-means algorithm (Hartigan 1975; MacQueen 1967) which is a special case of a self-consistency algorithm. The k-means algorithm illustrated here is an example of a "batch" algorithm where each iteration uses the entire dataset (e.g., Lloyd 1982). Given a set of data $\mathbf{x}_1, \dots, \mathbf{x}_n$, the goal is to find a set of k cluster means which optimally represent the data in terms of mean squared error. The algorithm begins with an initial set $S_0 = \{\mathbf{y}_1, \dots, \mathbf{y}_k\}$ of k distinct points. For each point $\mathbf{y}_j \in S_0$, define the *domain of attraction* of $\mathbf{y}_j$ as

$$D_{\mathbf{y}_j} = \{\mathbf{x}_i : \|\mathbf{y}_j - \mathbf{x}_i\| < \|\mathbf{y}_l - \mathbf{x}_i\|, \mathbf{y}_l \in S_0, \mathbf{y}_l \neq \mathbf{y}_j\};$$

that is, $D_{\mathbf{y}_j}$ is the set of points in the sample closer to $\mathbf{y}_j$ than any other point in S_0. The k-means algorithm iterates between the following two steps: (1) Compute the means of each cluster $D_{\mathbf{y}_j}$ to obtain a new set of k points denoted by S_1; and (2) reassign the data points to the cluster mean in S_1 according to minimal distance; that is, compute the domains of attraction $D_{\mathbf{y}}$ for each of the points in S_1. Repeat steps (1) and (2) obtaining $S_1, S_2, \dots$ until convergence is reached: $S_m = S_{m+1}$.

The cluster means produced at convergence of the k-means algorithm are estimators of the k principal points of the distribution which generated the data. The k principal points of a p-variate random vector $\mathbf{X}$ are defined to be the set of k distinct points $\{\boldsymbol{\xi}_1, \dots, \boldsymbol{\xi}_k\} \subset \Re^p$ so that

$$\mathcal{E}\left[\min_{j=1,\dots,k} \|\mathbf{X} - \boldsymbol{\xi}_j\|^2\right] \leq \mathcal{E}\left[\min_{j=1,\dots,k} \|\mathbf{X} - \mathbf{y}_j\|^2\right]$$

for all $\{y_1, \dots, y_k\} \subset \Re^p$. Principal points are a special case of self-consistent points (Flury 1993). For a set of k distinct points $\{\mathbf{y}_1, \dots, \mathbf{y}_k\} \subset \Re^p$, define

$$\mathbf{Y} = \mathbf{y}_j \quad \text{if} \quad \mathbf{X} \in D_{\mathbf{y}_j},$$

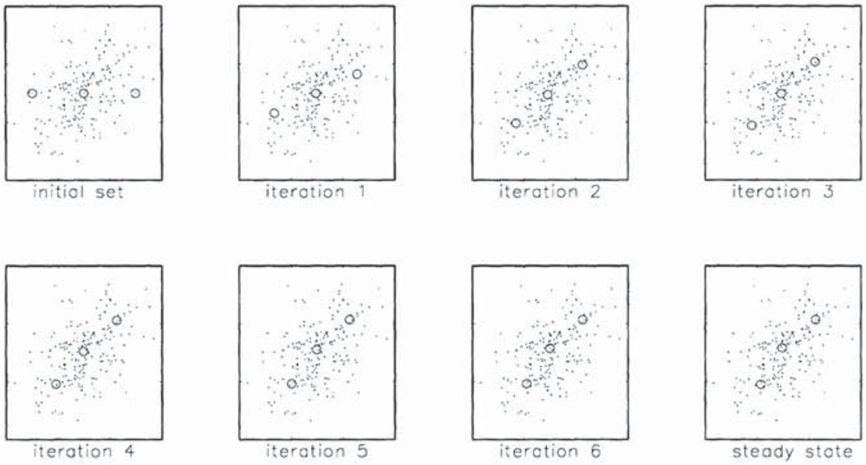

Figure 1. Swiss Head Dimension Data. Each frame shows a scatterplot of $n = 200$ observations on two head measurements. The large open circles represent $k = 3$ cluster means from iterations of the k-means algorithm which begins with three points along the x_1-axis. The $k = 3$ points are pulled towards the first principal component axis and remain roughly collinear through each iteration.

where $D_{\mathbf{y}_j}$ is the domain of attraction of $\mathbf{y}_j$, $j = 1, \ldots, k$,; that is, the set of all points in $\Re^p$ closer to $\mathbf{y}_j$ than any of the other $\mathbf{y}_i, i \neq j$. The points $\mathbf{y}_1, \ldots, \mathbf{y}_k$ are called k self-consistent points of $\mathbf{X}$ if $\mathbf{Y}$ is self-consistent for $\mathbf{X}$ according to Equation (1.1).

The k-means algorithm has been applied to provide estimates of principal points in a problem of determining optimal sizes and shapes of gas masks for soldiers in the Swiss Army (Flury 1993). To illustrate the behavior of the k-means algorithm, consider the Swiss head dimension data consisting of two head measurements—minimal frontal breadth (MFB) and breadth of angulus mandibulae (BAM)—on a sample of $n = 200$ Swiss soldiers (Flury and Riedwyl 1988, p. 219). In Figure 1, the k-means algorithm was run using $k = 3$ and beginning with the three points lying on the MFB-axis. Fitting $k = 3$ sizes corresponds to determining a small, medium, and large size gas mask. It took the algorithm nine iterations to converge.

Figure 1 shows that the three points remain roughly collinear at each iteration of the algorithm. Also, the three points are pulled closer and closer to the first sample principal component axis with each iteration. Assuming an approximate bivariate normal population, the $k = 3$ principal points should lie along the first principal component axis (Tarpey 1998).

Next, we simulate bivariate normal data to illustrate how the k-means algorithm behaves for a known underlying distribution. Using the software GAUSS, $n = 10,000$ bivariate normal variates were generated from the model $N_2(\mathbf{0}, \text{diag}(\sigma^2, 1))$. For large values of σ ($\sigma > 1.54$ approximately), most of the variability is along the first principal component axis and, consequently, the $k = 3$ principal points lie on the first principal component axis. However, for $\sigma \leq 1.54$, the three principal points form a triangle pattern (Tarpey 1998). Figure 2 shows the $k = 3$ cluster means for several iterations of the k-means algorithm using $\sigma = 1.2$ for the simulations. The algorithm begins with three

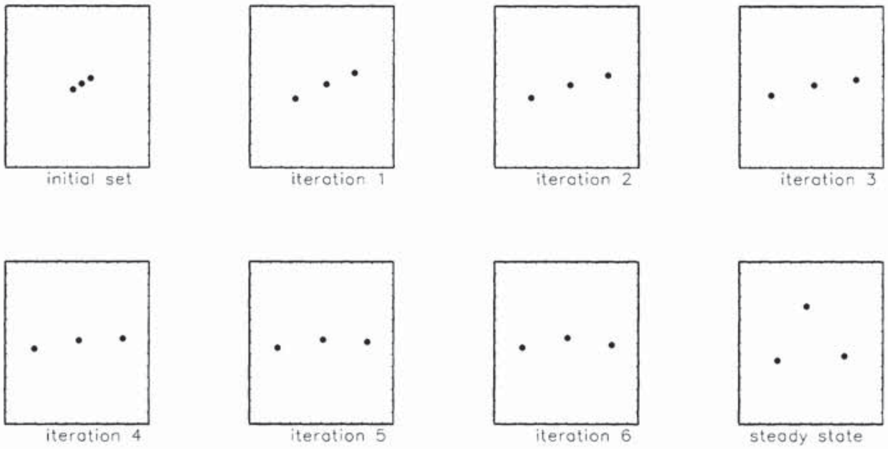

Figure 2. Iterations of the k-means algorithm for simulated bivariate normal data. The algorithm begins with $k = 3$ collinear points on the 30° diagonal line.

collinear points along a line making an angle of 30° with the horizontal axis. Once again, each of the first several iterations of the algorithm yield three collinear points and the line containing these points is pulled closer and closer to the first principal component axis. Once the points are pulled onto the first principal component axis, however, they eventually form the optimal triangle pattern when the steady state is reached after 65 iterations as shown in the last frame of Figure 2.

The foregoing simulation was repeated, using $k = 3$ collinear points lying on the second principal component axis (the vertical axis). The results of the first several iterations as well as the steady state are shown in Figure 3. The first few iterations of the algorithm leave the $k = 3$ points lying on the second principal component approximately. In other words, the three points remain initially stuck on the second principal component axis, before eventually being pulled into the optimal triangle pattern.

The k-means algorithm begins with an initial set S_0 of k distinct points. Suppose now that our initial set consisted of all points on a curve. If each iteration of the algorithm yields another curve, then at convergence one will have a self-consistent or principal curve (Hastie and Stuetzle 1989). For finite samples, we can approximate a smooth curve using a large number of points. To illustrate, we ran the k-means algorithm using $k = 20$ points on a circle (shown in the first frame of Figure 4) for a trivariate normal distribution. In GAUSS, we simulated $n = 15{,}000$ trivariate normal variates with mean zero and a diagonal covariance matrix $\mathrm{diag}(16, 9, 1)$. The k-means algorithm took 83 iterations to converge. The first iteration of the algorithm, as shown in the second frame of Figure 4, shows that the original $k = 20$ points in the circle have been rotated and transformed to lie roughly in an ellipse. Thus, the initial set of points lie in a plane, and the first iteration rotates the points into another plane. The third frame of Figure 4 shows the points in the final state of the algorithm. The points still lie approximately in a two-dimensional plane (a principal component analysis of the 20 points in the final state shows that the first two principal components account for over 99.99% of the total variability). Note that the algorithm eventually pulls the points out of the ellipse and into

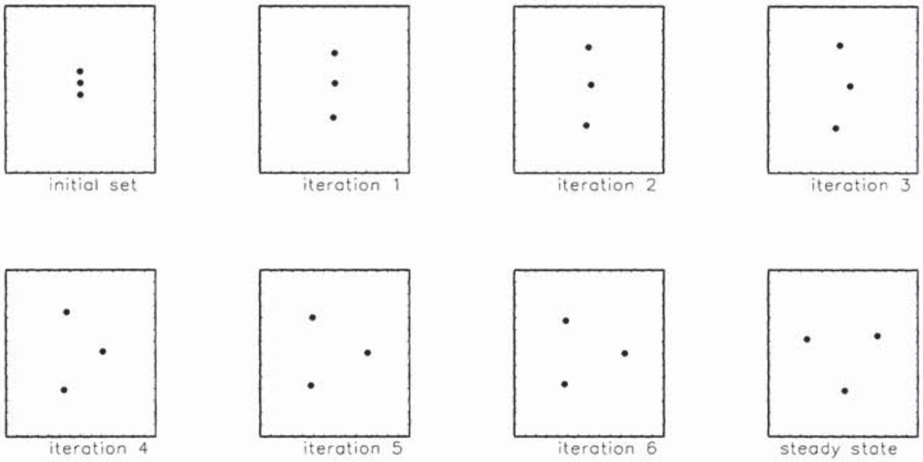

Figure 3. Iterations of the k-means algorithm for simulated bivariate normal data. The algorithm begins with $k = 3$ collinear points on the second principal component axis.

a pattern where the boundaries of the domains of attraction of the 20 points form roughly a hexagonal tessalation of the plane (Su 1997).

In this section we have used the k-means algorithm to find self-consistent approximations to a distribution using large simulated datasets. In the following sections, we give results that govern the behavior of the algorithm when applied to a theoretical model, and not just a finite sample from the model.

3. THE SELF-CONSISTENCY ALGORITHM

We now describe the self-consistency algorithm which is a generalization of the k-means algorithm described in Section 2. The following notation will be used throughout the article. Let $S \subset \Re^p$ denote a measurable set. For each $\mathbf{y} \in S$, define $D_{\mathbf{y}}(S) = \{\mathbf{x} \in \Re^p : \|\mathbf{x} - \mathbf{y}\| < \|\mathbf{x} - \mathbf{z}\|, \mathbf{z} \in S, \mathbf{z} \neq \mathbf{y}\}$ to be the *domain of attraction* of the point $\mathbf{y}$. Define the random vector $\mathbf{Y}$ as follows:

$$\mathbf{Y} = \mathbf{y} \text{ if } \mathbf{X} \in D_{\mathbf{y}}(S). \tag{3.1}$$

Thus, in Equation (3.1), $\mathbf{Y}$ is defined as a minimal distance projection of $\mathbf{X}$ onto the set S. If the random vector $\mathbf{Y}$ defined by Equation (3.1) is self-consistent according to the TF condition in Equation (1.1), then we say $\mathbf{Y}$ (or the corresponding set S) is self-consistent for $\mathbf{X}$ in the HS sense.

The HS definition of self-consistency immediately brings to mind the problem of *ambiguity points*. A point $\mathbf{x}$ is an ambiguity point if it has more than one point in S to which it is closest. HS (1989) showed that for an absolutely continuous distribution, the set of ambiguity points for a smooth curve has measure zero.

The self-consistency algorithm is similar to the EM algorithm (Dempster, Laird, and Rubin 1977) consisting of two steps: an E-step and an M-step. We will assume implicitly that moments exist as required.

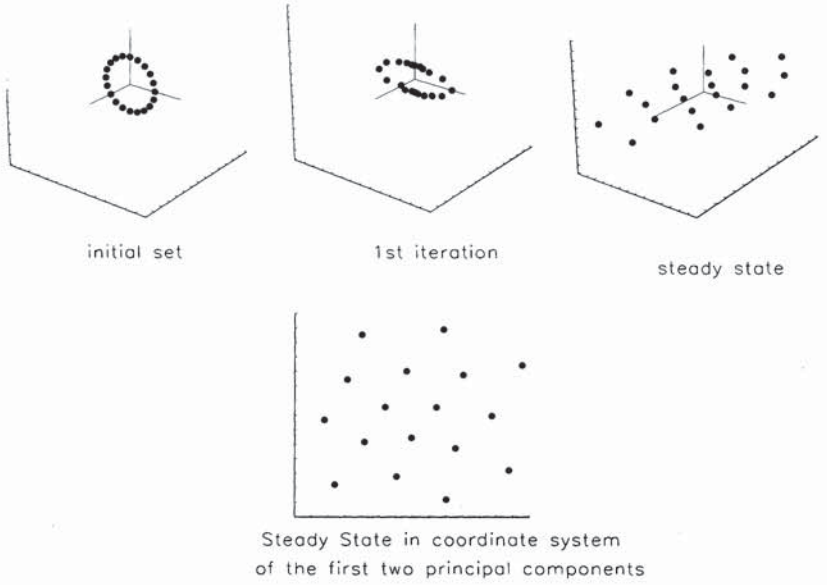

Figure 4. Frame one shows the initial set for the k-means algorithm beginning with k = 20 points on a circle. The algorithm was run on a sample 15,000 simulated trivariate normal observations. The second frame shows the first iteration of the algorithm. The third frame shows the steady state of the algorithm and the last frame shows the steady state of the algorithm in the subspace of the first two principal components of the model.

3.1 THE SELF-CONSISTENCY ALGORITHM

Let S_0, a measurable subset of $\Re^p$, denote our initial set. For each $\mathbf{y} \in S_0$ define $\mathbf{Y}_0$ by Equation (3.1).

1. E-step: Compute the conditional expectation $\tilde{\mathbf{Y}}_0 = \mathcal{E}[\mathbf{X}|\mathbf{Y}_0]$. Let S_1 denote the support of $\tilde{\mathbf{Y}}_0$.
2. M-step: Define $\mathbf{Y}_1$ according to minimal distances: for $\mathbf{y} \in S_1$, let $\mathbf{Y}_1 = \mathbf{y}$ if $\mathbf{X} \in D_{\mathbf{y}}(S_1)$.
3. Repeat steps 1 and 2 obtaining $S_1, S_2, S_3, \ldots$, and the corresponding $\mathbf{Y}_1, \mathbf{Y}_2, \mathbf{Y}_3, \ldots$, until convergence is reached.

For the purposes of this article, we shall consider only S_j as described in the algorithm which yield sets of ambiguity points of measure zero for continuous distributions.

We note that the TF notion of self-consistency is more general than the HS notion of self-consistency. Step 1 of the self-consistency algorithm will produce a sequence of random vectors $\tilde{\mathbf{Y}}_0, \tilde{\mathbf{Y}}_1, \ldots$ which will be self-consistent for $\mathbf{X}$ in the TF sense (see Tarpey and Flury 1996, lemma 2.5). However, the $\tilde{\mathbf{Y}}_j$ are not necessarily self-consistent in the HS sense. Let $\mathbf{Y}_c$ denote the steady state of the algorithm at convergence. That is, $\mathbf{Y}_c = \mathcal{E}[\mathbf{X}|\mathbf{Y}_c]$. Since $\mathbf{Y}_c$ is defined in terms of minimal distance of the $\mathbf{y} \in S_c$, it follows that $\mathbf{Y}_c$ is self-consistent in the HS sense. Hence, the algorithm produces a self-consistent approximation to $\mathbf{X}$ in both the TF and the HS senses. We shall return to this idea in Section 6.

We conclude this section with a result that shows that successive iterations of the self-consistency algorithm provide better approximations to a distribution in terms of mean squared error.

Proposition 1. *Let $\mathbf{Y}_j,\ j = 1, 2, \ldots,$ denote the random vectors from successive iterations of the self-consistency algorithm for a random vector $\mathbf{X}$. Then $\mathcal{E}\|\mathbf{X} - \mathbf{Y}_j\|^2$ is monotonically decreasing in j.*

Proof:

$$\begin{aligned}
\mathcal{E}[\|\mathbf{X} - \mathbf{Y}_j\|^2] &\geq \mathcal{E}[\|\mathbf{X} - \mathcal{E}[\mathbf{X}|\mathbf{Y}_j]\|^2] \\
&= \mathcal{E}[\|\mathbf{X} - \tilde{\mathbf{Y}}_j\|^2] \\
&\geq \mathcal{E}\left[\inf_{\mathbf{y}\in S_{j+1}} \|\mathbf{X} - \mathbf{y}\|^2\right] \quad \text{since } S_{j+1} \text{ is the support of } \tilde{\mathbf{Y}}_j \\
&= \mathcal{E}[\|\mathbf{X} - \mathbf{Y}_{j+1}\|^2].
\end{aligned}$$

□

Note that this proposition does not guarantee that the sequence $\mathcal{E}[\|\mathbf{X} - \mathbf{Y}_j\|^2]$ is strictly decreasing.

4. SELF-CONSISTENCY ALGORITHM FOR ELLIPTICAL DISTRIBUTIONS

Versions of the self-consistency algorithm are used on sample data for various statistical applications, such as estimating principal curves (HS 1989) and principal points (Flury 1993; Tarpey 1997). Of course, the statistician is not interested in the dataset, but in the model that generated the data. Thus, it is of interest to see how the self-consistency algorithm behaves when applied to the model. In this section we shall give two results that determine the behavior of the algorithm for elliptical distributions.

Proposition 2. *Suppose the self-consistency algorithm is run for a p-variate elliptically distributed random vector $\mathbf{X}$ with mean zero and a covariance matrix of full rank. Let S_0 denote the initial set whose support lies in a linear subspace of dimension $q < p$. Then at each iteration, the set of means $S_1, S_2, \ldots,$ will have a span whose dimension is at most q.*

Proof: Let Ψ denote the covariance matrix of $\mathbf{X}$. Let S_0 denote the initial set. Without loss of generality, assume S_0 lies in the subspace of the first q coordinates $X_1, \ldots, X_q$. Partition $\mathbf{X}$ as $\mathbf{X} = (\mathbf{X}_1', \mathbf{X}_2')'$, where $\mathbf{X}_1$ has q components and $\mathbf{X}_2$ has $p-q$ components. Partition $\mathbf{Y}_0$ analogously as $\mathbf{Y}_0 = \left(\mathbf{Y}_0^{(1)'}, \mathbf{Y}_0^{(2)'}\right)'$. Then the components of $\mathbf{Y}_0^{(2)}$ are zero. Partition Ψ accordingly:

$$\Psi = \begin{pmatrix} \Psi_{11} & \Psi_{12} \\ \Psi_{12}' & \Psi_{22} \end{pmatrix}.$$

Note that $\mathcal{E}[\mathbf{X}_2|\mathbf{X}_1] = \Psi_{12}'\Psi_{11}^{-1}\mathbf{X}_1$ by the linearity of the conditional expectation for elliptical distributions (e.g., see Fang, Kotz, and Ng 1990, p. 45). Then the E-step of the algorithm gives

$$\mathcal{E}[\mathbf{X}|\mathbf{Y}_0] \quad = \quad \mathcal{E}[\mathbf{X}|\mathbf{Y}_0^{(1)}]$$

$$\begin{aligned} &= \begin{pmatrix} \mathcal{E}[\mathbf{X}_1|\mathbf{Y}_0^{(1)}] \\ \mathcal{E}[\mathbf{X}_2|\mathbf{Y}_0^{(1)}] \end{pmatrix} \\ &= \begin{pmatrix} \tilde{\mathbf{Y}}_0^{(1)} \\ \mathcal{E}[\mathcal{E}[\mathbf{X}_2|\mathbf{X}_1]|\mathbf{Y}_0^{(1)}] \end{pmatrix} \\ &= \begin{pmatrix} \tilde{\mathbf{Y}}_0^{(1)} \\ \Psi_{12}'\Psi_{11}^{-1}\mathcal{E}[\mathbf{X}_1|\mathbf{Y}_0^{(1)}] \end{pmatrix} \\ &= \begin{pmatrix} \tilde{\mathbf{Y}}_0^{(1)} \\ \Psi_{12}'\Psi_{11}^{-1}\tilde{\mathbf{Y}}_0^{(1)} \end{pmatrix}. \end{aligned}$$

Let $\mathbf{A} = \Psi_{12}'\Psi_{11}^{-1}$, and let $\mathbf{a}_j$, $j = 1, \dots, q$, denote the columns of $\mathbf{A}$. Then each point in S_1, the support of $\tilde{\mathbf{Y}}_0$, can be expressed as a linear combination of:

$$(1, 0, \dots, 0, \mathbf{a}_1')',\ (0, 1, 0, \dots, 0, \mathbf{a}_2')', \dots, (0, \dots, 0, 1, \mathbf{a}_q')'.$$

Repeating this argument for each subsequent step of the algorithm proves the proposition. □

Our next proposition states that if we run the self-consistency algorithm for an elliptically distributed random vector and the initial set S_0 lies in a principal component subspace, then subsequent iterations of the algorithm produce sets $S_1, S_2, \dots$, each of which lies in the same principal component subspace. For example, if $\mathbf{X}$ is trivariate and we start with a circle that lies in the plane spanned by the second and third principal component axes, then the self-consistency algorithm converges to an object of dimension at most two (by Proposition 4.1) which must lie in the principal component subspace spanned by the second and third principal component axes. In other words, the algorithm is trapped in this two-dimensional principal component subspace.

Proposition 3. *Suppose we start the self-consistency algorithm for an elliptically distributed p-variate random vector $\mathbf{X}$ with mean zero from S_0 whose support lies completely in a principal component subspace of dimension $q < p$. Then the jth iteration of the self-consistency algorithm produces a set S_j which also lies in the same q-dimensional principal component subspace.*

Proof: Let Ψ denote the covariance matrix of $\mathbf{X}$ and let $\mathbf{H}\Lambda\mathbf{H}' = \Psi$ denote the spectral decomposition of Ψ where the matrix $\mathbf{H}$ is a $p \times p$ orthogonal matrix of eigenvectors of Ψ. Then the components of $\mathbf{Z} = \mathbf{H}'\mathbf{X}$ are the principal components of $\mathbf{X}$. Let S_0 be the initial set which lies in the principal component subspace spanned by the first $q < p$ columns of $\mathbf{H}$. Partition $\mathbf{Z}$ into q and $p - q$ components as $\mathbf{Z} = (\mathbf{Z}_1', \mathbf{Z}_2')'$. Since S_0 lies in the subspace spanned by the first q columns of $\mathbf{H}$, then $\mathbf{Y}_0$ is a measurable function of $\mathbf{Z}_1$ only, say $\mathbf{Y}_0 = f(\mathbf{Z}_1)$. Then

$$\begin{aligned} \tilde{\mathbf{Y}}_0 &= \mathcal{E}[\mathbf{X}|\mathbf{Y}_0] \\ &= \mathbf{H}\mathbf{H}'\mathcal{E}[\mathbf{X}|\mathbf{Y}_0] \quad \text{since } \mathbf{H} \text{ is orthogonal} \\ &= \mathbf{H}\mathcal{E}[\mathbf{Z}|f(\mathbf{Z}_1)] \\ &= \mathbf{H}\begin{pmatrix} \mathcal{E}[\mathbf{Z}_1|f(\mathbf{Z}_1)] \\ \mathcal{E}[\mathbf{Z}_2|f(\mathbf{Z}_1)] \end{pmatrix} \end{aligned}$$

$$\begin{aligned} &= \mathbf{H}\begin{pmatrix} \tilde{\mathbf{Z}}_1 \\ \mathcal{E}[\mathcal{E}[\mathbf{Z}_2|\mathbf{Z}_1]|f(\mathbf{Z}_1)] \end{pmatrix} \\ &= \mathbf{H}\begin{pmatrix} \tilde{\mathbf{Z}}_1 \\ \mathbf{0} \end{pmatrix} \quad \text{since the covariance matrix of } \mathbf{Z} \text{ is diagonal} \end{aligned}$$

Thus, the support of $\tilde{\mathbf{Y}}_0$ is spanned by the first q columns of $\mathbf{H}$. Repeating this argument for each iteration of the algorithm demonstrates that the sets $S_1, S_2, \ldots,$ lie in the principal component subspace spanned by the first q columns of $\mathbf{H}$. □

Propositions 2 and 3 help explain the behavior of the k-means algorithm in Figures 1, 2, and 3. In each of these examples, the initial set consisted of $k = 3$ collinear points. If the algorithm is run for the model as opposed to a finite sample produced from the model, then according to Proposition 2, each iteration of the algorithm will produce cluster means which must lie on a straight line. The first few iterations of the k-means algorithm shown in Figures 1, 2, and 3 show that the points remain approximately collinear. In Figure 3, the $k = 3$ points in the initial set lie on the second principal component axis. According to Proposition 3, if the algorithm were run for the model, then each iteration would produce points along the second principal component axis. The first few iterations for a finite sample shown in Figure 3 shows that the three points remain stuck on the second principal component axis initially before finally being pulled into the optimal triangle pattern.

5. PRINCIPAL COMPONENTS AND PRINCIPAL CURVES

HS (1989) introduced principal curves as a nonlinear generalization of principal component axes in terms of self-consistency. HS showed that if a line is self-consistent (in the HS sense), then the line must be a principal component axis. For elliptical distributions, principal component axes are always self-consistent (Tarpey 1999).

Using Propositions 2 and 3, we can characterize principal component axes as steady states of the self-consistency algorithm for elliptical distributions. To illustrate, consider a bivariate elliptical distribution. Begin the algorithm with an arbitrary line S_0 and let $\mathbf{Y}_0$ denote the projection of $\mathbf{X}$ onto this line. By Proposition 2, each step of the algorithm will yield another line. According to Proposition 1, the MSE of the resulting distribution $\mathbf{Y}_1$ must be smaller than or equal to that of $\mathbf{Y}_0$ which means that the line is pulled towards the first principal component axis at each iteration of the algorithm. At convergence, the algorithm produces a line which is self-consistent in both the HS and TF senses. Recall, that if a line is self-consistent in the HS sense, then it must be a principal component axis. Therefore, the self-consistency algorithm produces a principal component axis at the steady state when the initial set is a line for elliptical distributions. We can generalize this argument to p-variate elliptical distributions and initial sets which are planes of arbitrary dimension less than p.

The rate at which the self-consistency algorithm converges when beginning with a line for an elliptical distribution depends on the eccentricity of the contours of equal density. To illustrate, consider a bivariate elliptical distribution with mean zero, and covariance matrix $\Psi = \text{diag}(\sigma^2, 1)$. The algorithm begins with a line S_0 through the origin given by the equation $x_2 = mx_1$. Figure 5 shows the iterations of the algorithm

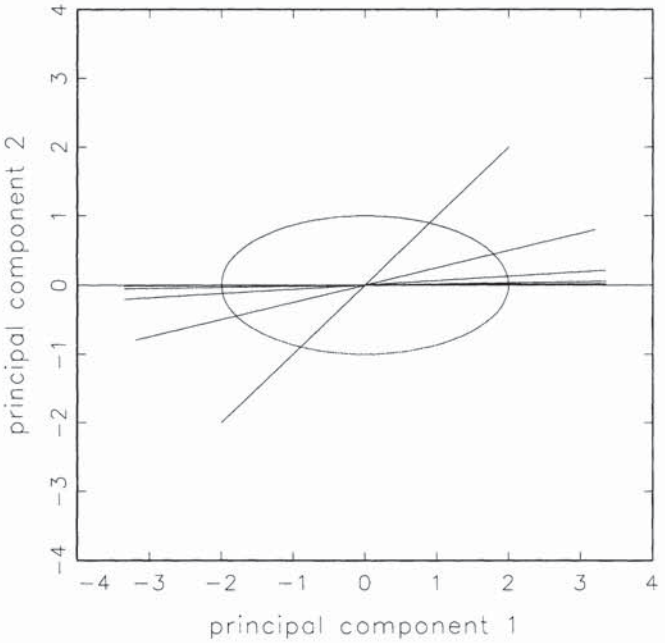

Figure 5. Iterations of the self-consistency algorithm beginning with a 45° diagonal line for the bivariate normal distribution centered at zero with covariance matrix diag (4,1). The ellipse represents a contour of constant density.

when $\sigma^2 = 4$ and the slope of the line is $m = 1$. One contour of constant density is also provided. As can be seen, the algorithm approaches the first principal axis quite quickly.

Figure 6 shows the iterations of the self-consistency algorithm again with the same initial starting line except this time the contours of equal density are less eccentric with $\sigma^2 = 1.1$. Since the distribuiton is nearly spherical (the contours of equal density are nearly perfect circles) the algorithm converges much more slowly to the first principal component axis. In fact, if we take $\sigma^2 = 1$, then any line through the origin is a self-consistent principal component axis in which case the algorithm will not produce distinct lines on subsequent iterations—the algorithm begins with a steady state.

The behavior of the self-consistency algorithm for nonelliptical distributions when the initial set is a straight line may differ drastically than for elliptical distributions. As a simple example, consider the uniform distribution in the top half of the unit circle. Then the horizontal line through the point (0,.5) is the first principal component axis—see Figure 7(a). However, the first principal component axis is not self-consistent. If we start the self-consistency algorithm beginning with the first principal component axis, the next step produces not a straight line, but the top half of the ellipse given by $x^2/4 + y^2 = .25$ as shown in Figure 7(b).

In practice, given a dataset produced from an unknown model, it is not known if the first principal component axis is self-consistent, or if a nonlinear principal curve should instead be estimated. Figures 5 and 6 indicate how the principal curve algorithm behaves when run for an elliptical distribution. To illustrate the self-consistency algorithm for finite datasets beginning with a line, we implement Hastie and Stuetzle's principal curve

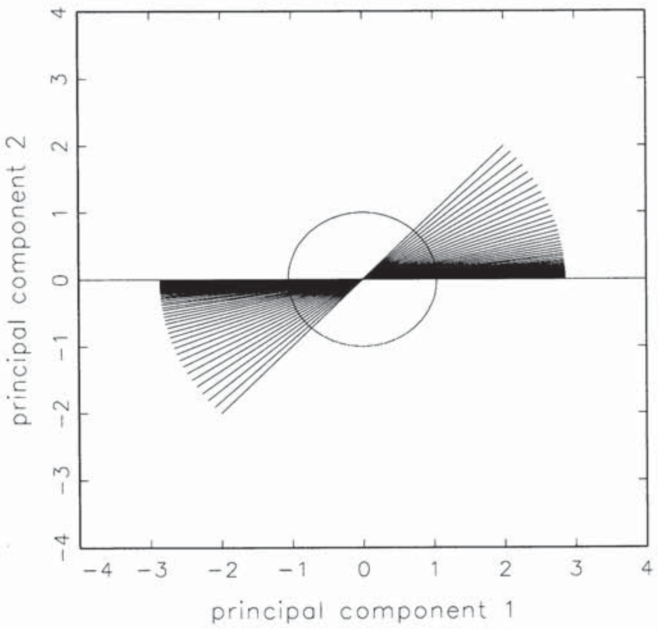

Figure 6. Iterations of the self-consistency algorithm beginning with a 45° diagonal line for the bivariate normal distribution centered at zero with covariance matrix diag (1.1,1). The ellipse represents a contour of constant density.

algorithm (1989, p. 506). The principal curve algorithm is a special case of the self-consistency algorithm where the E-step is based on scatterplot smoothing. In particular, HS considered the locally weighted running-lines smoother. Figure 8 shows the results of running the principal curve algorithm for $n = 200$ simulated bivariate normal variates with mean zero and diagonal covariance matrix $\text{diag}(1.1^2, 1)$. The left frame of Figure 8 shows the first principal component axis of the sample (the straight line) and the steady state based on using local averaging in the E-step with a span of .5. That is, for each data point, the expectation step of the algorithm uses a neighborhood based on half the data points that are nearest neighbors in terms of projections onto the curve. The right frame of Figure 8 shows the results of the principal curve algorithm using a span of .75. Typically, the larger the span, the smoother the resulting curve at convergence.

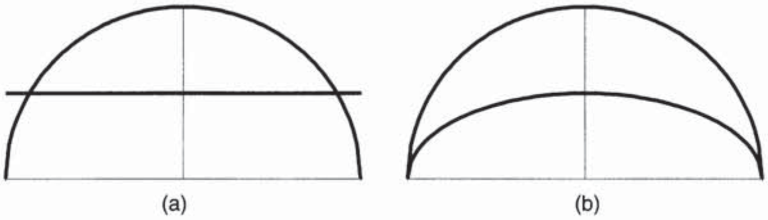

Figure 7. The uniform distribution in the top-half of the unit circle. Figure 7(a) shows the first principal component axis (the horizontal line) which is not self-consistent. Figure 7(b) shows what happens to the first principal component axis after the first iteration of the self-consistency algorithm.

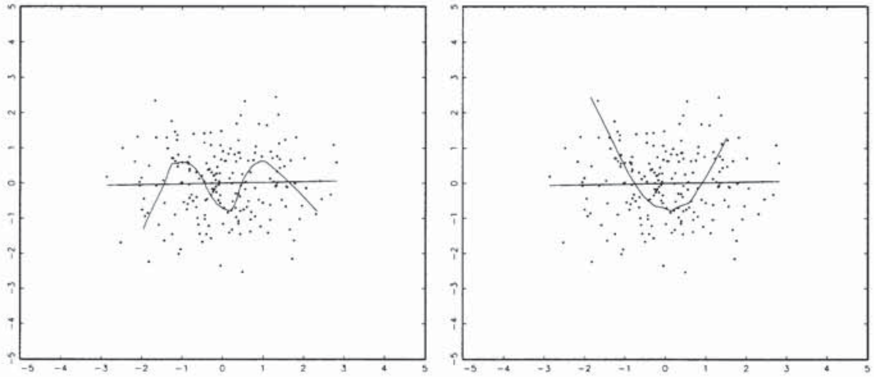

Figure 8. The steady state of the principal curve algorithm run on n = 200 simulated bivariate normal observations. The first frame shows the result of the algorithm using a span of .5 in the scatterplot smoother and the second frame shows the resulting curve useing a span of .75.

Contrasting Figures 5 and 6 with Figure 8, one can see once again that the finite sample behavior of the self-consistency algorithm is markedly different than the behavior of the algorithm for the model in the case of estimating principal curves. Beginning the principal curve algorithm with a straight line for a finite sample from a multivariate normal distribution, the algorithm bends the line drastically indicating that for a nearly spherical normal distribution, the first principal component is not the optimal self-consistent curve approximation to the distribution in terms of mean squared error.

6. SELF-CONSISTENT LOOPS

In the previous section we examined the behavior of the self-consistency algorithm when the initial set consisted of a straight line. Now we look for self-consistent approximations when the algorithm begins with a closed loop. We shall call a closed curve that is self-consistent in the HS sense a self-consistent loop. First, we give an example contrasting the difference between the TF and HS notions of self-consistency for loops.

Example 1. Let $\mathbf{X}$ denote a continuous random vector with a spherical distribution. Then $\mathbf{X}$ has a stochastic representation $\mathbf{X} = R\mathbf{U}$, where $R = \|\mathbf{X}\|$ and $\mathbf{U} = \mathbf{X}/\|\mathbf{X}\|$. $\mathbf{U}$ is uniformly distributed on the surface of the unit sphere, and R and $\mathbf{U}$ are independent (e.g., see Fang, Kotz, and Ng 1990). Let $r = \mathcal{E}[R]$. Consider the circle S centered at the origin with radius r: $S = \{\mathbf{x} \in \Re^2 : \mathbf{x}'\mathbf{x} = r^2\}$. For each point $\mathbf{y} \in S$, $D_{\mathbf{y}}(S)$, the domain of attraction of $\mathbf{y}$, is the ray extending from the origin through the point $\mathbf{y}$. Let $\mathbf{Y} = r\mathbf{U}$. Then the support of $\mathbf{Y}$ is the sphere S and $\mathbf{Y}$ is self-consistent for $\mathbf{X}$ in both the TF and the HS senses (TF 1996).

Now define a linear transformation of $\mathbf{X}$ to get $\mathbf{W} = \mathbf{A}'\mathbf{X}$ for some $p \times p$ diagonal matrix $\mathbf{A}$ of full rank which is not a multiple of the identity matrix. Then $\mathbf{W}$ has an elliptical distribution. Define $\mathbf{Y}_{\mathbf{A}} = r\mathbf{A}'\mathbf{U}$. Then the support of $\mathbf{Y}_{\mathbf{A}}$ is the (nonspherical) ellipsoid $E = \mathbf{A}'\mathbf{S}$ which, as we will see in the following, is not self-consistent in the

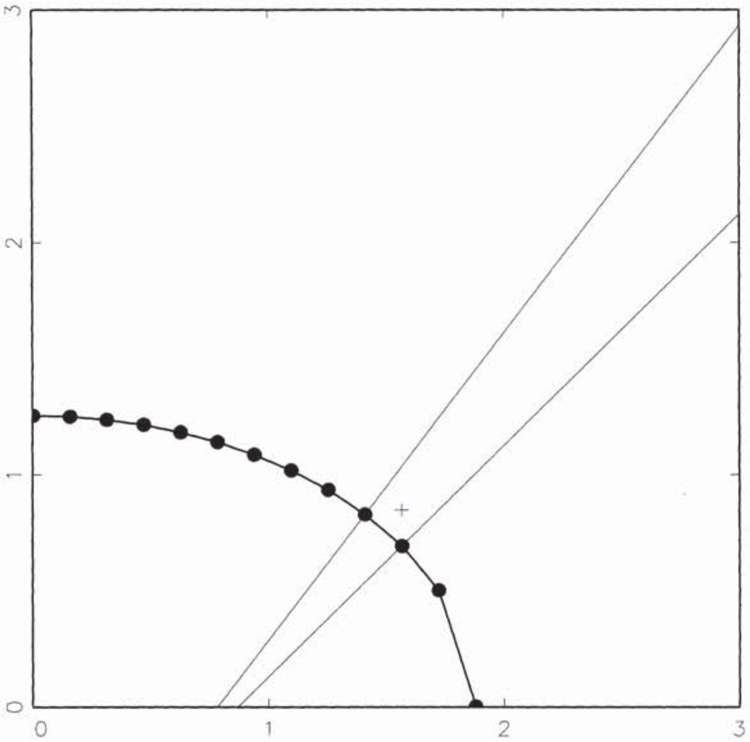

Figure 9. An iteration of the self-consistency algorithm for a principal loop. The solid points are used to approximate the curve. The two orthogonal lines to the curve at two of the points form a wedge shaped region. The conditional mean is computed numerically in this wedge region giving the value indicated by the + sign.

HS sense. However, $\mathbf{Y}_\mathbf{A}$ is self-consistent for $\mathbf{W}$ in the TF sense because

$$\begin{aligned} \mathcal{E}[\mathbf{W}|\mathbf{Y}_\mathbf{A}] &= \mathcal{E}[R\mathbf{A}'\mathbf{U}|r\mathbf{A}'\mathbf{U}] \\ &= \mathbf{A}'\mathbf{U}\mathcal{E}[R|\mathbf{A}'\mathbf{U}] \\ &= \mathbf{A}'\mathbf{U}\mathcal{E}[R] \quad \text{since } R \text{ is independent of } \mathbf{U} \\ &= r\mathbf{A}'\mathbf{U} \\ &= \mathbf{Y}_\mathbf{A}. \end{aligned}$$

In this case, $\mathbf{Y}_\mathbf{A}$ has an elliptical distribution on the surface of the ellipsoid E.

Duchamp and Stuetzle (1993, 1996) studied nonlinear principal curves and loops for the simple case of the uniform distribution on a rectangle and an ellipse. The principal curves were found by solving an ordinary differential equation which result from the self-consistency condition (see also Salinelli 1998). Duchamp and Stuetzle (1993, p. 149) noted that for nonspherical elliptical distributions it is not known if there exist principal curves other than the principal component axes. They did show that the uniform distribution on an ellipse (noncircular) does not have an elliptical principal loop. There do exist elliptical distributions with elliptical self-consistent loops in the simple case where $\mathbf{W}$ in Example 1 is self-consistent for itself and $\mathbf{W}$ has an elliptical distribution on the surface of an ellipse; that is, $\mathbf{W} = R\mathbf{A}'\mathbf{U}$ and $R =$ constant.

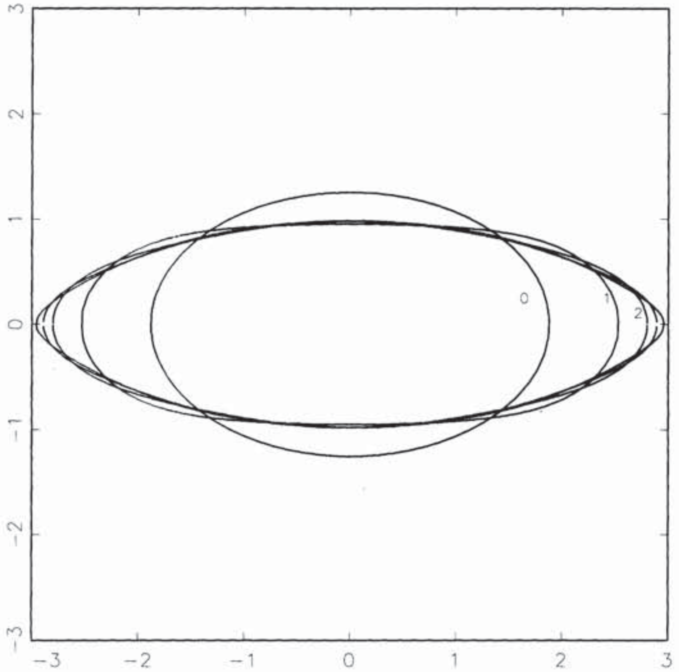

Figure 10. Principal Loop. The first four iterations of the self-consistency algorithm beginning with an ellipse (marked with the 0) for the bivariate normal distribution centered at zero with covariance matrix diag (1.5^2,1). The curves that result from the first and second iteration are indicated by 1 and 2, respectively.

The self-consistency algorithm will now be applied to the bivariate normal distribution to determine a self-consistent loop. In Example 1, if the ellipse E were self-consistent in the HS sense, then it would correspond to a steady state of the self-consistency algorithm. We shall run the self-consistency algorithm beginning with the ellipse E and show that it produces other loops upon subsequent iterations of the algorithm. The algorithm was run as follows: the initial ellipse E was approximated by 4,000 points on E. The line orthogonal to the curve at each point was computed. The conditional mean of $\mathbf{W}$ given that $\mathbf{W}$ lies in the wedge shaped area between two adjacent orthogonal lines was then computed by numerical integration in GAUSS. Figure 9 illustrates the procedure in the first quadrant.

The conditional mean over the wedge shaped area produced a new point denoted by the $+$ symbol in Figure 9. The set of these new points were then used to approximate the curve resulting from the E-step of the algorithm. On subsequent iterations of the algorithm, a curve through the points was fitted using least squares for the purpose of estimating the slope of the orthogonal line through each point. Figure 10 shows the results of running the algorithm through four iterations beginning with an initial ellipse E and $\sigma = 1.5$. The curves from the first two iterations are marked 1 and 2, respectively.

A limitation of this procedure is that it gives only approximations to the actual curves and the approximation will be less accurate in later iterations of the algorithm. However, the process was repeated for 40,000 points which essentially reproduced the same loops as those shown in Figure 10. As can be seen in Figure 10, it appears that the

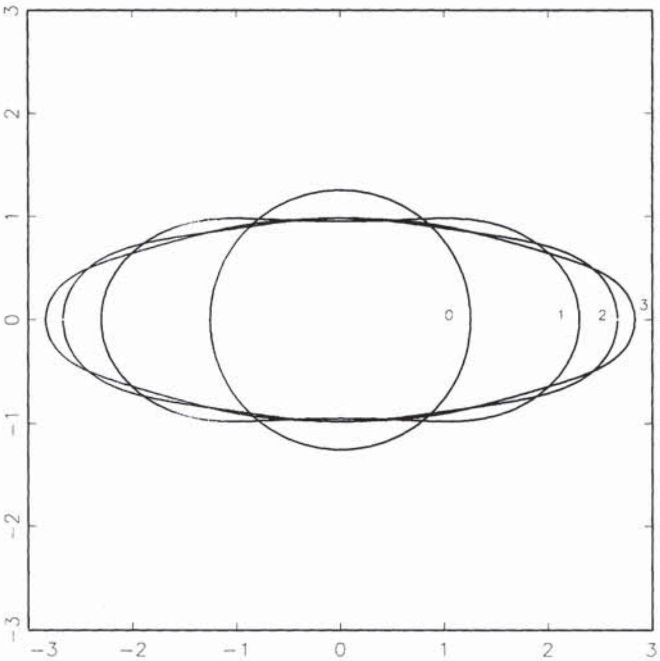

Figure 11. Principal Loop: Three iterations of the self-consistency algorithm beginning with a circle of radius $(\pi/2)^{1/2}$ for the bivariate normal distribution centered at zero with covariance matrix diag(1.5^2,1).

loops are converging to a principal loop for the bivariate normal distribution. We also ran the algorithm for the same distribution but instead of beginning the algorithm with an ellipse, we used a circle centered at the origin with radius $r = (\pi/2)^{1/2}$ as the initial loop. Figure 11 shows the initial circle along with three iterations of the algorithm. It appears from the figures that the algorithm may be converging to the same loop whether one begins with a circle or an ellipse but we have no proof that this is the case.

7. DISCUSSION

The self-consistency algorithm has been used to illustrate the TF and the HS notions of self-consistency. It should be noted that the term self-consistency was originally introduced by Efron (1967) to describe an estimator of a distribution function in survival analysis with censored data. See TF (1996) for an illustration of the relationship between the Efron and the TF notions of self-consistency.

We have described a general self-consistency algorithm and provided results that determine its behavior when applied to a theoretical model. Special cases of the algorithm are the k-means algorithm for estimating principal points and the principal curve algorithm for estimating principal curves.

The behavior of the algorithm applied to a theoretical model can differ dramatically from results obtained for finite samples from the model, even if the sample size is very large. When applying the algorithm to a model, the results of Section 4 show that suboptimal solutions may be produced at convergence unless care is taken in choosing the initial set. For finite samples, one may want to choose initial sets for the self-consistency algorithm whose span is of high dimension in order to speed convergence. Otherwise, the first several iterations of the algorithm may produce sets which are stuck in a linear subspace of low dimension. On the other hand, if one begins the self-consistency algorithm with an initial set that lies in a two-dimensional plane say, and the first iteration of the algorithm produces a set of points that also lie in a two-dimensional plane, then this is evidence that the underlying distribution may be elliptical.

Analytical solutions for self-consistent approximations to theoretical distributions can be very difficult to determine even for relatively simple distributions. Applying the self-consistency algorithm is a practical way to solve the problem of determining self-consistent approximations as in the case of principal loops in Section 6.

This article shows what to expect from the self-consistency algorithm when applied to the class of elliptical distributions. It would be interesting to expand this work to other multivariate distributions.

ACKNOWLEDGMENTS

The author thanks two anonymous referees and an associate editor for many helpful suggestions which have improved this article.

[Received August 1998. Revised August 1999.]

REFERENCES

Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977), "Maximum Likelihood Estimation from Incomplete Data Via the EM Algorithm" (with discussion), *Journal of the Royal Statistical Society*, Series B, 39, 1–38.

Duchamp, T., and Stuetzle, W. (1993), "Geometric Properties of Principal Curves in the Plane," in *Robust Statistics, Data Analysis, and Computer Intensive Methods*, ed. Helmut Rieder, *Springer Lecture Notes in Statistics*, 109, pp. 135–152.

——— (1996), "Extremal Properties of Principal Curves in the Plane," *The Annals of Statistics*, 24, 1511–1520.

Efron, B. (1967), "The Two Sample Problem with Censored Data," in *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability* (vol. 4), Berkeley, CA: University of California Press, pp. 831–853.

Fang, K., Kotz, S., and Ng, K. (1990), *Symmetric Multivariate and Related Distributions*, New York: Chapman and Hall.

Flury, B. (1990), "Principal Points," *Biometrika*, 77, 33–41.

——— (1993), "Estimation of Principal Points," *Applied Statistics*, 42, 139–151.

Flury, B., and Riedwyl, H. (1988), *Multivariate Statistics: a Practical Approach*, London: Chapman and Hall.

Hartigan, J. A. (1975), *Clustering Algorithms*, New York: Wiley.

Hastie, T., and Stuetzle, W. (1989), "Principal Curves," *Journal of the American Statistical Association*, 84, 502–516.

Lloyd, S. P. (1982), "Least Sqaures Quantization in PCM," *IEEE Transactions on Information Theory, Special Issue on Quantization*, IT–28, 129–137.

MacQueen, J. (1967), "Some Methods for Classification and Analysis of Multivariate Observations," in *Proceedings 5th Berkeley Symposium on Mathematics, Statistics and Probability*, 3, pp. 281–297.

Salinelli, E. (1998), "Nonlinear Principal Components I. Absolutely Continuous Random Variables with Positive Bounded Densities," *The Annals of Statistics*, 26, 596–616.

Su, Y. (1997), "On Asymptotics of Quantizers in Two Dimensions," *Journal of Multivariate Analysis*, 61, 67–85.

Tarpey, T. (1997), "Estimating Principal Points of Univariate Distributions," *Journal of Applied Statistics*, 24, 499–512.

——— (1998), "Self-Consistent Patterns for Symmetric Multivariate Distribution," *The Journal of Classification*, 15, 57–79.

——— (1999), "Self-Consistency and Principal Component Analysis," *Journal of the American Statistical Association*, 94, 456–467.

Tarpey, T., and Flury, B. (1996), "Self-consistency: A Fundamental Concept in Statistics," *Statistical Science*, 11, 229–243.

Tarpey, T., Li, L., and Flury, B. (1995), "Principal Points and Self-Consistent Points of Elliptical Distributions," *The Annals of Statistics*, 23, 103–112.