# An Analysis of *Locally Defined Principal Curves and Surfaces*

James McQueen

Department of Statistics, University of Washington Seattle, WA, 98195, USA

**Abstract**

Principal curves are generally defined as *smooth* curves passing through the *middle* of the data and have wide applications in machine learning for example in dimensionality reduction and feature extraction. Recently Umut Ozertem and Deniz Erdogmus (O&E) provide us with a novel approach for defining principal curves and surfaces in their paper *Locally Defined Principal Curves and Surfaces* (2011). This report aims to reproduce the results of their paper and provide critical assessment of its performance, flaws, and merits.

## 1 Introduction

Perhaps the most popular dimension reduction tool used today is known as principal components analysis (PCA). PCA is an orthogonal linear transformation which transforms (rotates) the data into a new coordinate system such that the greatest variance of the data projected on each of the new coordinates is associated with the first coordinate (first principal component), the second greatest variance with the second coordinate, etc. This is a commonly used tool in dimension reduction as we can project the original data onto the subspace spanned by the first $d$ principal components, thus preserving most of the variance of the data but reducing the dimension.

As we are projecting onto a linear space this can be describe as *linear* dimension reduction. Inherent to this method is that the first principal component is a *line*, the first and

second principal components create a plane, etc., and we rank these lines according to the amount of variance explained by each. The success of PCA as a linear dimension reduction technique begs the question: can we extend PCA to a non-linear set of *principal surfaces* that retain some of the desirable properties of principal lines?

# 2 Challenges

There are a number of challenges in non-linear dimension reduction in general, additionally, there is no agreed upon definition of a principal curve and therefore no agreed method of estimating them. Popular non-linear dimension reduction (or *manifold learning*) techniques such as Isomap (Tenenbaum et al. [2000]), local linear embedding (Roweis and Saul [2000]) , Laplacian eigenmaps (Belkin and Niyogi [2003]), and maximum variance unfolding (Weinberger and Saul [2006]) rely on generating locality information of data samples from a data proximity graph. These techniques, however, depend on careful tuning of parameters controlling graph structure as the accuracy of these methods depends on the quality of the graph. Furthermore, many of these techniques assume that the data truly lie on a manifold of inherent dimension $d$ and try to recover the underlying manifold. These methods rely on the validity of the assumption.

There is currently no agreed upon definition of principal curves as they do not come about so naturally as principal components (lines). Principal curves are generally understood to be smooth curves passing through the middle of the data, however, a more mathematically precise definition is required. Many definitions involve taking a property of the principal line and then try to find a smooth curve that fits these properties: Hastie and Stuetzle [1989] require them to be self-consistent whereas Delicado [1998] restricts their total variance and conditional means. After requiring that a principal curve satisfies some constraint, one then devises an algorithm that finds the principal curve that meets the requirement and minimizes

a criterion (such as mean-squared projection errror). Therefore, these methods aim to find a curve that best fits *the data*. This has two primary flaws. First, trying to minimize some data-weighted criterion leads to overfitting and so regularization is almost certainly required. A more philosophical issue is that principal curves ought to be thought of as inherent structures of the data generating mechanism that have to be approximated as opposed to *defined* to be the *solutions* to an algorithm.

# 3 Method

Ozertem and Erdogmus (henceforth referred to as O&E) take a novel approach to principal curves by defining them to be inherent structures of the underlying probability distribution of the data. They consider a principal surface defined such that every point on the principal surface is a local maximum (local mode) of the probability density in the local orthogonal subspace. In particular this definition implies that the principal curve (surface of dimension 1) is the "ridge" of the probability distribution – a "natural " result. Defining principal curves as structures of probability density functions leads to a differential geometric definition depending on the gradient and Hessian of probability density function.

Let $p(x)$ be the density function of $x \in \mathbb{R}^n$, let $g(\cdot)$ be the gradient of $p(\cdot)$ and $H(\cdot)$ be the Hessian. Let $(\lambda_i(x), q_i(x))$ be the $i$th eigenvalue, eigenvector pair of $H(x)$. Let $C^d$ be the set points $x$ such that there exists a set $I_\perp \subset \{1, \ldots, n\}$ with $|I_\perp| = n - d$ such that $g(x)^T q_i(X) = 0 \; \forall i \in I_\perp$. We say $x$ is a regular point of $C^d$ if the set $I_\perp$ is unique, that is, $g(x)$ is perpendicular to *exactly* $(n - d)$ eigenvectors. Any regular point $x$ of $C^d$ is perpendicular to $n - d$ orthogonal eigenvectors and therefore all such points must lie on a surface with intrinsic dimension $d$. Then define $\mathcal{P}^d$ the principal surface of dimension $d$ to be the set of regular points of $C^d$ such that $\lambda_i(x) < 0 \; \forall i \in I_\perp$, that is, $\mathcal{P}^d$ contains the local maxima of the orthogonal subsbace $C^d_\perp(x) = span\{q_i(x) | i \in I_\perp\}$ (as the gradient at these points projected

onto $C_{\perp}^{d}(x)$ is by definition zero).

To see why this definition is natural, consider $\mathcal{P}^0$. This is the set of points that are orthogonal to $n - 0 = n$ eigenvectors (that is, all of them) therefore $g(x) = 0$ so they are critical points, but additionally they are the critical points associated with negative eigenvalues and are thus local maxima. Hence $\mathcal{P}^0$ is the set of local modes, $\mathcal{P}^1$ is the *principal curve* (surface of dimension 1) and defines the ridge of the density function, i.e. there is only one direction of increase. This is a satisfying definition as it both defines principal curves as inherent structures of the data generating mechanism and it naturally extends from principal curves (of dimension 1) to principal surfaces of arbitrary dimension $d > 1$. Additionally, these surfaces exist whenever the density function permits a gradient and Hessian. In practice Kernel Density Estimation or parametric density estimation (e.g. mixture models) are used to estimate $p$ and they are composed from densities with at least second order derivatives.

## 3.1 Existence and Consistency of Principal Curves

Since this definition of principal curves depends on first and second derivatives of the density as long as these exist such that the hessian is non-zero then the principal curve exist. These conditions are mild and since in practice kernel densities are used these guarantee the gradient and Hessian exist and are continuous and so the principal surfaces will exist. Chacon et al. [2011] show that under the assumption that the kernel bandwidth matrix converges to zero fast enough, that the underlying density and kernel have a sufficient number of continuous square-integrable derivatives, that the kernel has finite covariance then the integrated mean-squared-error between the vector of order-r derivatives of the KDE converge to those of the true density. Therefore for a sufficiently smooth kernel and density, the derivatives of the KDE are consistent. Consequently, since the principal surfaces are defined by the first and second derivatives they too must be consistent.

# 4 Algorithm

O&E present an adjustment to the mean shift algorithm that they claim will converge to the principal surface of dimension $d$ i.e., $\mathcal{P}^d$. In the algorithm we initialize with either a mesh of points or the data points themselves (the latter ensures the resulting principal surface will be in the support of the data and is also the projection of the data onto the surface).

## 4.1 Monotonically Increasing functions and Local Covariance

Instead of involving the Hessian $H(x)$ in their subspace constrained mean-shift algorithm, the authors define a new matrix called the "Local Covariance". In order to motivate their definition of Local Covariance, in their paper the authors show that:

**Lemma 4.1** *For strictly increasing, twice differentiable functions $f$ the principal set $\mathcal{P}^d$ of a density $p(x)$ is the same as the principal set $\tilde{\mathcal{P}}^d$ of the transformed density $f(p(x))$.*

Let $x \in \mathcal{P}^d$ with pdf $p(x)$, gradient $g(x)$ and hessian $H(x)$. Let $H(x) = Q\Lambda Q$ be the eigendecomposition. Since $x$ is a a point in $\mathcal{P}^d$ its gradient $g(x)$ is orthogonal to all eigenvectors $q_i(x)$ in the set $I_\perp$ and whose span is the *orthogonal* space. Let $Q_\perp$ be the matrix whose columns are composed of these eigenvectors, consequently let $Q_{||}$ composed of the remaining eigenvectors that span the *parallel* space. Then in this case we may write $H(x) = Q_{||}\Lambda_{||}Q_{||}^T + Q_\perp\Lambda_\perp Q_\perp^T$ where the $\Lambda$'s are the corresponding eigenvalues. Since $g(x)$ is orthogonal to the vectors in $Q_\perp$ by definition, it must be in the parallel space: $g(X) = Q_{||}\beta$ for some weight vector $\beta$. We then calculate that gradient and hessian of the transformed pdf $f(p(x))$.

$$g_f(X) = f'(p(x))g(x) = f'(p(x))Q_{||}\beta = Q_{||}(f'(p(x))\beta) \equiv Q_{||}\beta'$$

Therefore the gradient is also in the parallel space and $x \in \mathcal{C}_f^d$ as well.

$$\begin{aligned} H_f(x) &= f'(p(x))H(f(x)) + f''(p(x))g(x)g(x)^T \\ &= f'(p(x))\left[Q_{||}\lambda_{||}Q_{||}^T + Q_\perp \Lambda_\perp Q_\perp\right] + f''(p(x))Q_{||}\beta\beta^T Q_{||}^T \\ &= \left\{f'(p(x))Q_{||}\lambda_{||}Q_{||}^T + f''(p(x))Q_{||}\beta\beta^T Q_{||}^T\right\} + f'(p(x))Q_\perp\Lambda_\perp Q_\perp \end{aligned}$$

Then since $f'(x) > 0$ for all $x$ the sign eigenvalues of the orthogonal space do not change and as such $x \in \mathcal{P}_f^d$, as required.

Consider the special case of $f = log(x)$ and $p(x)$ is the Gaussian density, we have the property that:

$$H(x) = (-1/2)\Sigma^{-1}$$

Where $\Sigma$ is the covariance matrix for the Gaussian distribution. This implies that when the underlying density is assumed to be Gaussian the principal curve definition coincides with principal components. This leads O&E to define a "local covariance" for any distribution $p(x)$ based on the above. This also gives us an ordering of the eigenvectors (as in PCA) such that we select the $n-d$ eigenvectors associated with the $n-d$ largest eigen-values of the local covariance:

$$\Sigma^{-1}(x) \equiv -2H_{\log}(X) = -\frac{H(X)}{p(X)} + \frac{g(X)g(X)^T}{p(X)^2}$$

By Lemma 4.1 the principal surface defined by using the local covariance is identical to the principal surface defined by the Hessian. In fact, the eigenvalues of the Hessian are just $-p(x)$ times the eigenvalues of the local covariance thus we equally take the eigenvectors of $H(x)$ associated with the $n-d$ smallest eigenvalues of the Hessian. In Ghassabeh et al. [2013] they point out that while the motivation for the use of the so-called local inverse covariance is due to the relationship to principal components when the underlying density is Gaussian, in practice the density used will never be Gaussian and so they argue that direct

use of the Hessian or other estimates of the local covariance can be used with impunity. In fact, they prove convergence of the algorithm in all of these cases (that is, convergence in a finite number of steps not necessarily to the principal surface). Simulations in their paper also indicate significant computational-savings using two local-covariance estimates defined by Wang and Carreira-Perpiñán [2010] with no worse performance (in terms of mean square deviation from the underlying generative spiral) when compared to use of the local covariance as defined by O&E.

## 4.2 Mean-shift

In order to define an algorithm that converges to the principal surface of dimension $d$: $P^d$ the authors adjust the well-known Mean-Shift algorithm, thus we briefly review what the mean-shift algorithm does and why it makes sense to adjust it for this purpose.

The mean-shift algorithm is a general-purpose algorithm for finding local-modes in data. It is often used (Comaniciu and Meer [2002]) for clustering. It is a non-parametric method that assumes an underlying kernel density estimate. As we will be using the Gaussian Kernel throughout the paper we will specialize to this case. If we define the $k(x)$ be the Gaussian "profile": $k(x) = \exp\left(-\frac{1}{2}x\right)$ (different from the Kernel in that the squaring operation is done before passing it to the function $k$). Then we define the kernel density estimate of $p(\mathbf{x}^t)$ the underlying distribution of $\mathbf{x}^t \in \mathbb{R}^n$ based on $N$ i.i.d samples $\mathbf{x}_i \sim p(\cdot)$ as:

$$\hat{p}(\mathbf{x}^t) \equiv \frac{1}{Nh^n}\left(\frac{1}{2\pi}\right)^{n/2}\sum_{i=1}^{N} k\left(\left|\left|\frac{\mathbf{x}^t - \mathbf{x}_i}{h}\right|\right|^2\right)$$

A local mode is a local maximum in the density function, thus, to find a local maximum we take the gradient of the density function with respect to the point of interest $\mathbf{x}^t$ and set it

to zero:

$$\begin{aligned}\nabla\hat{p}(x) \equiv g(x) &= \frac{2}{Nh^{n+2}}\left(\frac{1}{2\pi}\right)^{n/2}\sum_{i=1}^{N}(\mathbf{x}-\mathbf{x}_i)k'\left(\left|\left|\frac{\mathbf{x}^t-\mathbf{x}_i}{h}\right|\right|^2\right) \\ &= \frac{1}{Nh^{n+2}}\left(\frac{1}{2\pi}\right)^{n/2}\sum_{i=1}^{N}(\mathbf{x}_i-\mathbf{x})k\left(\left|\left|\frac{\mathbf{x}^t-\mathbf{x}_i}{h}\right|\right|^2\right) \\ &= \frac{1}{Nh^{n+2}}\left(\frac{1}{2\pi}\right)^{n/2}\left[\sum_{i=1}^{N}k\left(\left|\left|\frac{\mathbf{x}^t-\mathbf{x}_i}{h}\right|\right|^2\right)\right]\left[\frac{\sum_{i=1}^{N}k\left(\left|\left|\frac{\mathbf{x}^t-\mathbf{x}_i}{h}\right|\right|^2\right)\mathbf{x}_i}{\sum_{i=1}^{N}k\left(\left|\left|\frac{\mathbf{x}^t-\mathbf{x}_i}{h}\right|\right|^2\right)}-\mathbf{x}^t\right]\end{aligned}$$

The step in the second line follows as $k'(x) = -\frac{1}{2}k(x)$ for the special case of Gaussian profile. The quantity in square brackets:

$$\left[\frac{\sum_{i=1}^{N}k\left(\left|\left|\frac{\mathbf{x}^t-\mathbf{x}_i}{h}\right|\right|^2\right)\mathbf{x}_i}{\sum_{i=1}^{N}k\left(\left|\left|\frac{\mathbf{x}^t-\mathbf{x}_i}{h}\right|\right|^2\right)}-\mathbf{x}^t\right] \equiv m(x)$$

Is called the "mean-shift" and we iteratively set the gradient to zero by setting:

$$\mathbf{x}^{t+1} = \mathbf{x}^t + m(\mathbf{x}^t)$$

A so-called "mean-shift update". When this becomes a fixed point we have found a local mode.

## 4.3 Deriving the Gradient and Hessian

As $p(x)$ is generally unknown, O&E assume an underlying Kernel Density Estimate. The definition applies for any estimate $p(x)$, however, presently we will consider fixed bandwidth $h$ Kernel Density Estimators with Gaussian Kernel. Since we are using the same KDE as in

the Mean-shift the gradient is the same:

$$\nabla\hat{p}(x) \equiv g(x) = \frac{1}{Nh^{n+2}}\left(\frac{1}{2\pi}\right)^{n/2}\left[\sum_{i=1}^{N} k\left(\left|\left|\frac{\mathbf{x}^t - \mathbf{x}_i}{h}\right|\right|^2\right)\right]\left[\frac{\sum_{i=1}^{N} k\left(\left|\left|\frac{\mathbf{x}^t-\mathbf{x}_i}{h}\right|\right|^2\right)\mathbf{x}_i}{\sum_{i=1}^{N} k\left(\left|\left|\frac{\mathbf{x}^t-\mathbf{x}_i}{h}\right|\right|^2\right)} - \mathbf{x}^t\right]$$

We can also simplify the gradient to:

$$g(x) = \frac{p(\mathbf{x}^t)}{h^2}\left[m(\mathbf{x}^t) - \mathbf{x}^t\right]$$

Taking the second derivative we get the Hessian:

$$\begin{aligned}
H(\mathbf{x}^t) &= \frac{1}{Nh^{n+2}}\left(\frac{1}{2\pi}\right)^{n/2}\sum_{i=1}^{N}\left\{k\left(\left|\left|\frac{\mathbf{x}^t - \mathbf{x}_i}{h}\right|\right|^2\right)\left[\frac{1}{h^2}(\mathbf{x}^t - \mathbf{x}_i)(\mathbf{x} - \mathbf{x}_i)^T - I_n\right]\right\} \\
&= \frac{1}{Nh^{n+4}}\left(\frac{1}{2\pi}\right)^{n/2}\sum_{i=1}^{N} k\left(\left|\left|\frac{\mathbf{x}^t - \mathbf{x}_i}{h}\right|\right|^2\right)\left\{\frac{\sum_{i=1}^{N} k\left(\left|\left|\frac{\mathbf{x}^t-\mathbf{x}_i}{h}\right|\right|^2\right)(\mathbf{x}^t - \mathbf{x}_i)(\mathbf{x} - \mathbf{x}_i)^T}{\sum_{i=1}^{N} k\left(\left|\left|\frac{\mathbf{x}^t-\mathbf{x}_i}{h}\right|\right|^2\right)} - h^2 I_n\right\} \\
&= \frac{1}{h^4}p(\mathbf{x}^t)\left\{\frac{\sum_{i=1}^{N} k\left(\left|\left|\frac{\mathbf{x}^t-\mathbf{x}_i}{h}\right|\right|^2\right)(\mathbf{x}^t - \mathbf{x}_i)(\mathbf{x}^t - \mathbf{x}_i)^T}{\sum_{i=1}^{N} k\left(\left|\left|\frac{\mathbf{x}^t-\mathbf{x}_i}{h}\right|\right|^2\right)} - h^2 I_n\right\} \\
&\equiv \frac{p(\mathbf{x}^t)}{h^4}\left\{v(\mathbf{x}^t) - h^2 I_n\right\}
\end{aligned}$$

Then we are trying to find points $\mathbf{x}^t$ whose gradient is orthogonal to exactly $n-d$ eigenvectors of the Hessian, that is, we are looking for local modes in the *orthogonal* subspace as defined in §3. This leads to an adjustment to the Mean-shift algorithm that the authors name the subspace-constrained mean-shift algorithm. It is similar to a projected gradient (Goldstein [1964] and Levitin and Polyak [1966]) version of the mean shift, where the mean shift update $m(x)$ is projected into the local orthogonal space before being used to update the trajectory $\mathbf{x}^t$.

## 4.4 Subspace Constrained Mean-shift

Given the above idea of constraining the mean-shift update into the orthogonal subspace, the authors adjust the mean-shift algorithm to create the "subspace constrained mean-shift algorithm".

---

**Subspace Constrained Mean Shift** (SCMS) for Gaussian KDE

---

**Input**: denisty estimate $p(\mathbf{x})$, desired dimension $d$, tolerance $\epsilon > 0$.
**Initialize**: Trajectories $\mathbf{x}_1^0, \ldots, \mathbf{x}_K^0$ to a mesh or data points.
**for** $k = 1$ to $k = K$ **do**
  **while** not converged **do**

1. $m(\mathbf{x}_k^t) \leftarrow \left[ \frac{\sum_{i=1}^N k\left(\left|\left|\frac{\mathbf{x}^t - \mathbf{x}_i}{h}\right|\right|^2\right)\mathbf{x}_i}{\sum_{i=1}^N k\left(\left|\left|\frac{\mathbf{x}^t - \mathbf{x}_i}{h}\right|\right|^2\right)} - \mathbf{x}^t \right]$ evaluate mean-shift
2. $g(\mathbf{x}_k^t) \leftarrow \frac{p(\mathbf{x}_k^t)}{h^2}\left[m(\mathbf{x}_k^t) - \mathbf{x}_k^t\right]$ evaluate gradient
3. $v(\mathbf{x}_k^t) \leftarrow \frac{\sum_{i=1}^N k\left(\left|\left|\frac{\mathbf{x}^t - \mathbf{x}_i}{h}\right|\right|^2\right)(\mathbf{x}^t - \mathbf{x}_i)(\mathbf{x}^t - \mathbf{x}_i)^T}{\sum_{i=1}^N k\left(\left|\left|\frac{\mathbf{x}^t - \mathbf{x}_i}{h}\right|\right|^2\right)}$
4. $H(\mathbf{x}_k^t) \leftarrow \frac{p(\mathbf{x}^t)}{h^4}\left\{v(\mathbf{x}^t) - h^2 I_n\right\}$ evaluate Hessian
5. $\mathbf{\Sigma}^{-1}(\mathbf{x}_k^t) \leftarrow -\frac{1}{p(\mathbf{x}_k^t)}H(\mathbf{x}_k^t) + \frac{1}{p(\mathbf{x}_k^t)^2}g(\mathbf{x}_k^t)g(\mathbf{x}_k^t)^T$ evaluate local covariance
6. perform the eigendecomposition: $\mathbf{\Sigma}^{-1}(\mathbf{x}_k^t) = \mathbf{V}\mathbf{\Lambda}\mathbf{V}$
7. $\mathbf{V}_\perp \leftarrow [\mathbf{v}_1, \ldots \mathbf{v}_n]$ eigenvectors with the $(n-d)$ largest eigenvalues of $\mathbf{\Sigma}^{-1}(\mathbf{x}_k^t)$.
8. $\hat{m}(\mathbf{x}_k^t) \leftarrow \mathbf{V}_\perp \mathbf{V}_\perp^T m(\mathbf{x}_k^t)$ project $m(\mathbf{x}_k^t)$ onto the orthogonal sub-space
9. $\tilde{x}_k^t \leftarrow \hat{m}(\mathbf{x}_k^t) + \mathbf{x}_k^t$ projected/subspace constrained mean-shift update

  **if** $|g^T(\mathbf{x}_k^t)\mathbf{V}_\perp^T g(\mathbf{x}_k^t)|/(||g(\mathbf{x}_k^t)|| \cdot ||\mathbf{V}_\perp^T g(\mathbf{x}_k^t)||) < \epsilon$ **then**
    declare converged
  **else**
    $\mathbf{x}_k^{t+1} \leftarrow \tilde{\mathbf{x}}_k^t$
  **end if**
  **end while**
**end for**

---

Note that each trajectory $k$ can be run individually without knowledge of the others (thus the for loop can be run in parallel). The parallelization of the algorithm will decrease computation time, however, the procedure is still inherently iterative in each trajectory and requires evaluating the kernel density as well as an eigendecomposition at each step. The algorithm is $\mathcal{O}(N^2 \times n^3)$ where $N$ is the number of data points and $n$ is the dimension of

the data. Thus, even when run in parallel, for large data sets (especially of large dimension) this algorithm can be slow.

### 4.5 On the convergence of SCMS

It should be noted that the authors claim convergence of the SCMS algorithm by relation to the convergence of the Mean-shift algorithm proposed in Comaniciu and Meer [2002], however, Li et al. [2007] pointed out a fundamental mistake in the proof of the MS algorithm in Comaniciu and Meer [2002], thus there are no proofs of the optimality of the algorithm and whether or not it converges to the principal curve/surface. Recently, Ghassabeh et al. [2013] investigated the convergence properties of the SCMS algorithm. It was shown in Carreira-Perpiñán [2007] that if a Gaussian profile is used then the MS algorithm reduces to an EM algorithm and thus converges, however, use of other profiles do not guarantee convergence. In Ghassabeh et al. [2013] they point out that even if the MS converges it is not obvious that this implies convergence of the SCMS let alone to the desired principal surface. They do show, however, that the algorithm will converge (i.e. it will end) in a finite number of steps though not necessarily to the correct surface.

## 5 Experiments

In order to examine the principal curve method proposed by O&E we perform a number of experiments. In the first two sections we compare the O&E's princpal curve method to the original Hastie & Stuetzle principal curve method as well as the method proposed by Kegl [1999]. In the third section we display the robustness of O&E's principal curve method and SCMS algorithm to handle more complicated data sets without changes. Finally, we perform a simulation study to compare the principal curve and wavelet denoising methods.

## 5.1 Standard Principal Curve Data Sets

In this section we examine how the principal curve algorithm defined by O&E performs on some standard principal curve data sets. This is primarily to ensure that the methods have been replicated accurately and the results should look similar to the original paper. We compare this method to the methods proposed by Hastie & Stuetzle and Kegl.

### 5.1.1 Zig-Zag Data set

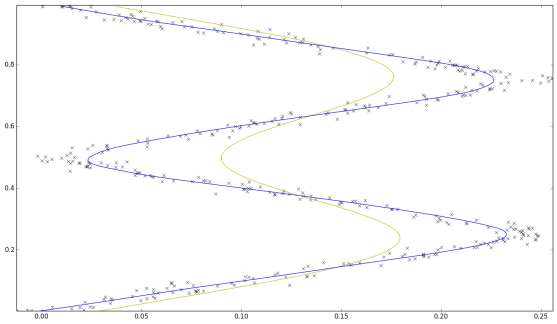


Figure 1: The Zig-Zag data set is plotted in (×), Hastie & Stuetzle's principal curve curve in yellow, and Kegl's Polygonal Line in blue.[1]

Figure 1 plots the Hastie and Stuetzle and Kegl's polygonal line algorithm with the zig-zag data set[2]. Figure 2 plots the principal curve for a variety of values of the bandwidth parameter. Once a bandwidth is selected the pricipal curve is found using the SCMS algorithm presented in §4.4. The algorithm is initialized on the original data points such that the resulting curve is the projection of the data onto the principal curve as defined by O& E. The bandwidth parameters were chosen to display the importance of appropriate selection of the bandwidth as the results can vary heavily from small changes.

[1]Computed using Kegl's Java application code:
[2]Data set provided by Kegl

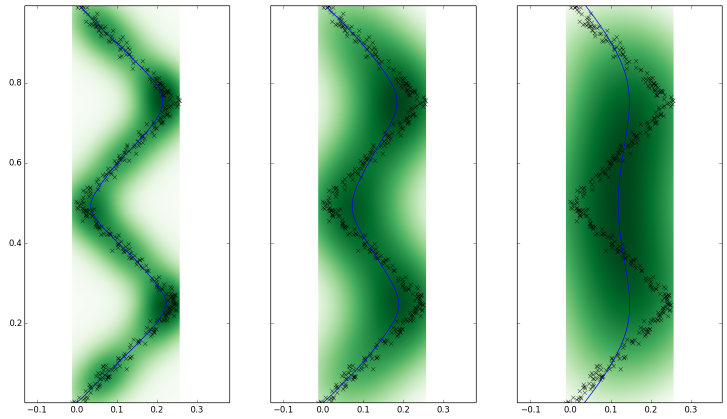

Figure 2: Zig-zag data set (×), O&E Principal Curve (blue), an intensity map of the associated KDE estimate is in green indicating the different curves for different bandwidths.

### 5.1.2 Spiral Data Set

Here we take the spiral data set and compare the three methods of finding principal curves. Figure 3 plots the Hastie & Stuetzle line as well as the Polygonal Line. Figure 4 plots the principal curve solution for different values of the bandwidth parameter $h$. It should be noted that the bandwidth that performs well on one data set (e.g. zig-zag data) does not necessarily perform well on another data set (e.g. spiral data). Therefore, in practice it is advisable to use a data-dependent kernel bandwidth, for example selecting by leave-one out maximum likelihood as we will below.

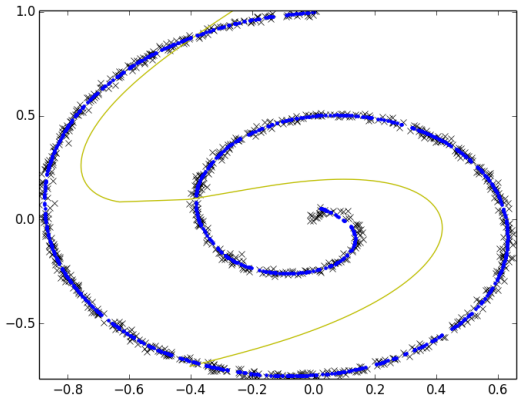

Figure 3: The Spiral data set is plotted in (×), Hastie & Stuetzle's principal curve in yellow, as well as Kegl's Polygonal Line in blue

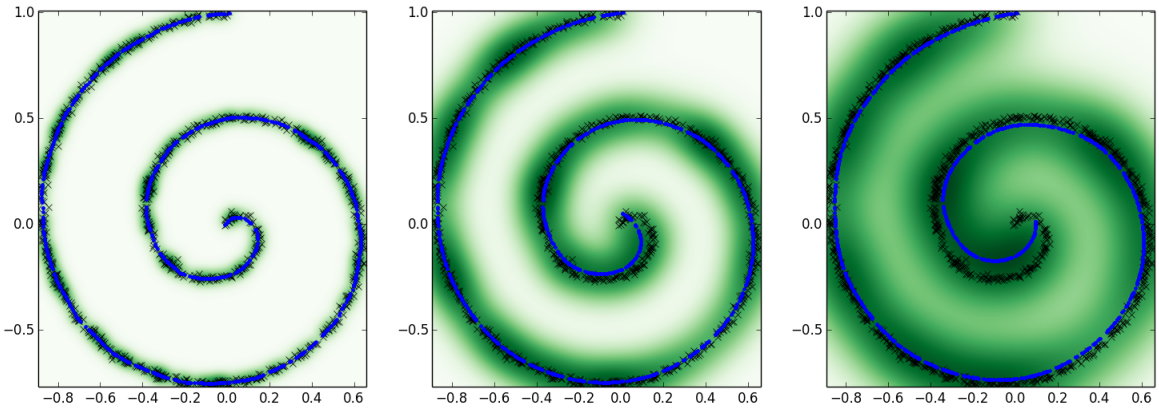

Figure 4: Spiral data set (×), O&E Principal Curve (blue), an intensity map of the associated KDE estimate is in green indicating the different curves for different bandwidths.

## 5.2 Other Data Sets

In addition to the standard principal curve data sets, the principal curve method defined by O&E can handle arbitrarily complicated data sets including those with self-intersections, bifrucations and loops. These data sets do not require alteration of the algorithm. Both of these are improvements over existing principal curve methods. This data set has many self-loops and bifurcations but can be handled by the SCMS algorithm with no additional changes. Figure 5 shows the resulting principal curve on two complicated data sets with self-intersections. The first is a star shape and the second is a epitrochoid.

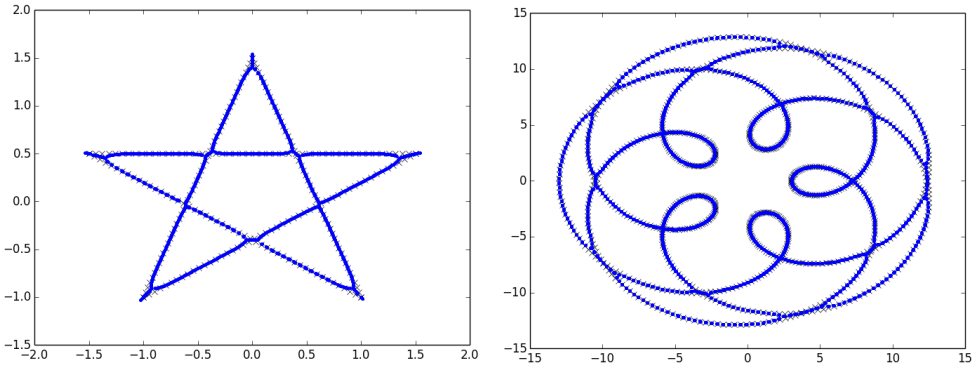

Figure 5: The underlying data are × and the resulting O&E principal curve is plotted in blue. The left is a star and the right is an epitrochoid. These plots display the ability of principal curves as defined by the authors to handle complex data sets

## 5.3 Signal Denoising

The authors claim that the principal curve method can be applied to solve the problem of denoising a signal. In Ozertem et al. [2008] they apply their method to piece-wise linear functions that have been corrupted by white where they achieve some level of success but do not compare to other more robust denoising methods. In this case we will consider a deterministic one-dimensional time signal $D$ which has been corrupted by some form of i.i.d mean-zero noise $\epsilon$. In this case we will let $\epsilon$ be Gaussian white noise. We define the signal $D$ deterministically as a function of sinusoids. The goal of any denoising method is, given a corrupted signal $X = D + \epsilon$ to estimate $D$. Naturally, as variability of the noise increases this task becomes more challenging. This leads to defining a notion of Signal-To-Noise-Ratio:

$$SNR = \frac{||D||^2}{E[||\epsilon||^2]}$$

It is common in the signal processing literature to write the SNR in terms of decibels. SNR (in dB) $= 10 \times \log_{10}(SNR)$. As such we will follow this standard.

### 5.3.1 Principal Curve Denoising

Here we follow Ozertem et al. [2008] in their paper on applying principal curve methods to denoise piece-wise linear signals. That is, we use the Gaussian kernel with single bandwidth parameter $h$ that we will select by leave-one-out maximum likelihood cross-validation as in Leiva-Murillo and Rodríguez [2012] That is, we select the bandwidth that maximizes $Pr(x_i|\mathbf{x}_{-i})$ over the entire data set where $\mathbf{x}_{-i}$ is the data set excluding the point $i$. The estimated true signal $D$ will be $\mathcal{P}^1 = \hat{D}$ i.e. the principal curve for the data set under the Gaussian KDE.

### 5.3.2 Wavelet Denoising

As much of the details in wavelet denoising theory are beyond the scope of this paper, our discussion will be brief. The Discrete Wavelet Transformation (DWT) is an orthonormal transformation, such that, for an orthonormal matrix $\mathcal{W}$ (defined by a choice of wavelet filter) we define the wavelet coefficients to be: $\mathbf{W} = \mathcal{W}\mathbf{X}$. Since $\mathcal{W}$ is an orthonormal transformation this representation $\mathbf{W}$ both preserves energy (i.e. $||\mathbf{W}||^2 = ||\mathbf{X}||^2$ but also is an exact (alternate) representation of the signal in that we can invert the transformation to recover the data set: $\mathbf{X} = \mathcal{W}^{-1}\mathbf{W} = \mathcal{W}^T\mathbf{W}$. Where the last step follows from the orthonormality of $\mathcal{W}$.

The method of Wavelet denoising comes from the more general orthonormal transformation denoising. The methodology is simple:

1. Given a signal $\mathbf{X}$ and a choice of wavelet filter we calculate the wavelet coefficients $\mathbf{W}$
2. Given a threshold $\delta$ set to zero all wavelet coefficients $W_t$ such that $|W_t| < \delta$
3. Given thresholded coefficients $\mathbf{W}^{(T)}$ calculate the inverse transformation to arrive at new signal $\mathbf{X}^{(T)}$

Then we take $\mathbf{X}^{(T)}$ to be our estimate of $D$. Based on the work of Donoho and Johnstone [1994] we use the "universal threshold" which is derived from assuming Guassian white noise (as in this case): $\delta^{(U)} = \sqrt{2\sigma_e^2 \log(N)}$. Since the variance of the noise is (typically) unknown we use the Median Absolute Deviation (MAD) estimate $\sigma^2_{(MAD)} = \frac{\text{median}\{W_{1,1},\ldots,W_{1,N/2}\}}{0.6745}$ which under certain assumptions (Donoho and Johnstone [1994]) is an unbiased estimate of $\sigma_e^2$.

### 5.3.3 Results

We fix a sample size $N$ and generate corrupted signal $X = D + \epsilon$ varying the SNR. We then apply both wavelet and principal curve methods to estimate $D$. We evaluate their

performance based on Mean-Squared-Error from the true signal $D$. We then repeat this procedure 100 times for each sample size $N$ and each $SNR$ to provide Monte-Carlo standard deviation bounds. In figure 6 we consider small sample sizes $N = 32$ and $N = 64$. We plot log MSE to make the plots easier to read. In both cases here we see that in terms of mean-squared error for any signal to noise ratio that the principal curve method is out-performing the wavelet method.

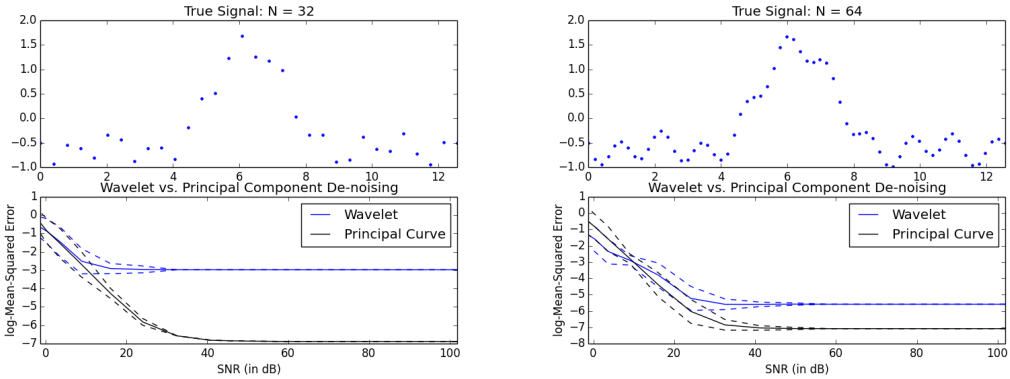


Figure 6: The top on both sides plots the underlying signal (uncorrupted) for data sizes 32 and 64. The bottom plots the resulting log MSE for the principal curve method (black) and wavelet method (blue) for varying SNRs.

What we see in figures 7 and 8, however, is a change as we increase the sample size. For larger data sets the more theoretically well-founded wavelet denoising method vastly outperforms the principal curve method which appears to stagnate around -8 log mean-squared error regardless of sample size. There might be a variety of reasons for this. First, there exists a great deal of theory on wavelet denoising (and orthonormal transforms in general), whereas there is none for the principal curves. In particular, the threshold $\delta$ was found assuming an underlying Gaussian white noise process but the bandwidth selection method for the principal curve was a general method for selecting bandwidths for density estimation. That being said, the principal curve method is still performing generally well and with additional theory (in particular on kernel selection and bandwidth selection in this particular instance of signal denoising) could lead to an improved principal curve denoising method that may be more comparable to state of the art methods.

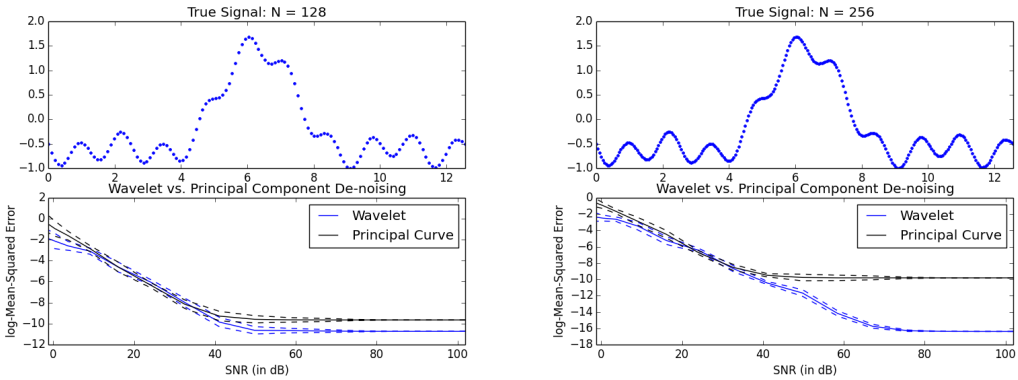


Figure 7: The top on both sides plots the underlying signal (uncorrupted) for data sizes 128 and 256. The bottom plots the resulting log MSE for the principal curve method (black) and wavelet method (blue) for varying SNRs.

In order to assess whether this test was biased towards the wavelet method due to use of a true signal based on sinusoids, the experiment was re-run using a piece-wise linear signal as in Ozertem and Erdogmus [2008]. The resulting plots (similar to those in Figures 6 through 8) are in the supplementaray appendix. As is in this case, the principal curve method outperforms the wavelet method for data sets of small sizes $N < 32$ but by the time $N > 64$ the wavelet method outperforms the principal curve method (in terms of MSE). It should be noted that the KDE-SCMS is $\mathcal{O}(N^2)$ whereas the DWT is $\mathcal{O}(N)$ (faster than the fast Fourier transform) though in principle the KDE-SCMS can be parallelized reducing the computation load.

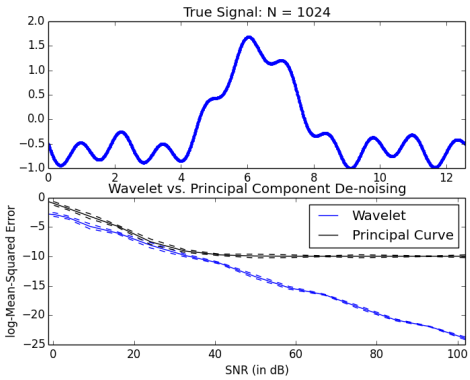


Figure 8: The top plots the underlying signal (uncorrupted) for data sized 1024. The bottom plots the resulting log MSE for the principal curve method (black) and wavelet method (blue) for varying SNRs.

# 6 Significance

This method of defining principal curves and surfaces offers a number of advantages over existing methods. By defining principal surfaces as inherent structures of the geometry rather than solutions to an optimizing criterion O&E allow for a richer definition of principal curves. Furthermore, the method extends the existing principal curves literature by defining "principal surfaces" in such a manner that can be naturally extended from principal curves of 1 dimension to principal surfaces of arbitrary dimension. Currently no other method of defining principal curves allows for this. Additionally, this definition allows finding principal curves in data with loops, bifurcations and self-intersections without any additional changes in the definition or algorithm.

In their definition of principal curves O&E rely on a known density function $p(x)$. In practice, of course, this is not available and so must be estimated from data. O&E take the approach of approximating $p(x)$ via Kernel Density Estimation. This allows them to do a number of things. First, the smoothness constraints that are usually placed on principal curves can be removed by assuming that $p(x)$ itself is smooth resulting in inherently smooth principal curves. If $p(x)$ is estimated via KDE the result will be inherently smooth. Furthermore, these Kernel Density estimates always have second order derivatives and so principal curves (as defined by O&E) are well defined, and, under certain regularity conditions they are consistent. Finally, issues of overfitting and outlier robustness can be handled in the density estimation phase (which has a much larger existing literature) than in the principal curve approximation phase.

While this method offers a new insight into principal curves, its impact in manifold learning is less supported. The Subspace Constrained Mean-Shift (SCMS) algorithm presented by the authors may converge to a principal surface of dimension $d$, however, the vector of values will still be in the ambient, larger, dimension $D > d$. Thus, while the points are guaranteed to

lie on a surface of lower *inherent* dimension, this method alone cannot be used for dimension reduction unless paired with another suitable algorithm to parametrize the principal surface or approximate it by projecting the points onto vectors of lower dimension.

Much of the literature on manifold learning assumes that there is an underlying true manifold from which the data is generated. These methods can assess their quality by determining if it will recover the true manifold given sufficient data. The principal surface method defined by O&E does not assume an underlying manifold (in fact there is no guarantee that the resulting principal surface itself is indeed a manifold), and it is unknown whether the method would recover the underlying manifold if the data were generated from one, or if the principal surface of appropriate dimension can be used as a reasonable estimate of the underlying manifold.

Nevertheless, principal curves as defined by O&E have enjoyed success in signal processing. In particular, they have been used in vector quantization Ghassabeh et al. [2012], as well as in signal denoising Ozertem et al. [2008]. Recently, Zhang and Pedrycz [2014]t proposed extending principal curves to Granular principal curves in order to do apply principal curves to large data sets by granulating the data. The method proposed by O&E opens the door to a new (potentially rich) principal curve/surface framework to be studied and applied.

# References

M. Belkin and P. Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Computation, 15(6)*, pages 1373–1396, 2003.

M.A. Carreira-Perpiñán. Gaussian mean shift is an em algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence 29*, pages 767–776, 2007.

J. E. Chacon, T. Duong, and M.P. Wand. Asymptotics for general multivariate kernel density derivative estimators. *Statistica Sinica, in press*, 2011.

D. Comaniciu and P. Meer. Mean shift: a robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(5):603–619, 2002.

P. Delicado. Principal curves and principal oriented points, 1998. URL `http://www.econ.upf.es/deehome/what/wpapers/postscripts.309.pdf`.

D.L Donoho and I.M. Johnstone. Ideal spatial adaptation by wavelet shrinkage. *Biometrika, 81*, pages 425–455, 1994.

Y.A. Ghassabeh, T. Linder, and G. Takahara. On noisy source vector quantization via subspace constrained mean shift algorithm. *Proceedings of the 26th Biennial Symposium on Communications, Kingston, Canada*, pages 107–110, 2012.

Y.A. Ghassabeh, T. Linder, and G. Takahara. On some convergence properties of the subspace constrained mean shift. *Pattern Recognition 46*, pages 3140–3147, 2013.

A.A. Goldstein. Convex programming in hilbert spaces. *Bulletin of the American Mathematical Society:70*, pages 709–710, 1964.

T. Hastie and W. Stuetzle. Principal curves. *Journal of American Statistical Association*, 84:502–516, 1989.

B. Kegl. Principal curves; learning, design, and applications. *PhD thesis, Concordia University, Montreal, Canada*, 1999.

J.P. Leiva-Murillo and A.A. Rodríguez. Algorithms for gaussian bandwidth selection in kernel density estimators. *Pattern Recognition Letters, Vol 33. Issue 13*, pages 1717–1724, 2012.

E.S. Levitin and B.T. Polyak. Constrained minimization problems. *USSR Computational Mathematics and Mathematical Physics 6*, pages 1–50, 1966.

X. Li, Z. Hu, and F. Wu. A note on the convergence of the mean shift. *Pattern Recognition 40*, pages 1756–1762, 2007.

U. Ozertem and D. Erdogmus. Local conditions for critical and principal manifolds. *IEEE Int. Conf on Acoustics Speech and Signal Processing*, pages 1893–1896, 2008.

U. Ozertem, D. Erdogmus, and O. Arikan. Piecewise smooth signal denoising via priciple curve projections. *IEEE International Conference on Machine Learning for Signal Processing*, pages 426–431, 2008.

S.T. Roweis and L.K. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science, 290(55500)*, pages 2323–2326, 2000.

J.B Tenenbaum, V. de Silva, and J.C Langford. A global geometric framework for nonlinear dimensionality reduction. *Science, 290(5500)*, pages 2319–2323, 2000.

W. Wang and M.A. Carreira-Perpiñán. Manifold blurring mean shift algorithsm for manifold denoising. *Proceeding of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2010)*, pages 1759–1766, 2010.

K.Q. Weinberger and L.K. Saul. Unsuperviased learning of image manifolds by semidefinite programming. *International Journal of Computer Vision 70(1)*, pages 77–90, 2006.

H. Zhang and W. Pedrycz. From principal curves to granular principal curves. *IEE Transactions on Cybernetics Vol 44. N0. 6*, pages 748–760, 2014.