# Probabilistic Principal Surface Classifier

Kuiyu Chang[1] and Joydeep Ghosh[2]

[1] School of Computer Engineering,
Nanyang Technological University, Singapore 639798, Singapore
kuiyu.chang@pmail.ntu.edu.sg,
http://www.ntu.edu.sg/home/askychang
[2] Department of Electrical and Computer Engineering,
University of Texas at Austin,
Austin Texas 78712, USA

**Abstract.** In this paper we propose using manifolds modeled by probabilistic principle surfaces (PPS) to characterize and classify high-D data. The PPS can be thought of as a nonlinear probabilistic generalization of principal components, as it is designed to pass through the "middle" of the data. In fact, the PPS can map a manifold of any simple topology (as long as it can be described by a set of ordered vector co-ordinates) to data in high-dimensional space. In classification problems, each class of data is represented by a PPS manifold of varying complexity. Experiments using various PPS topologies from a 1-D line to 3-D spherical shell were conducted on two toy classification datasets and three UCI Machine Learning datasets. Classification results comparing the PPS to Gaussian Mixture Models and K-nearest neighbours show the PPS classifier to be promising, especially for high-D data.

## 1 Introduction

Nonlinear manifolds embedded in high-dimensional space can provide a useful low-dimensional (2-D or 3-D) summary of the data that is visualizable by humans, assuming that the intrincsic data dimensionality is much lower. Principal curves and surfaces[1] can be used to compute a manifold that generalizes the property of principal components and subspaces, respectively. A discrete non-parametric approximation[2] of principal surfaces that is much simpler to compute comes in the form of Kohonen's self-organizing map (SOM)[3]. The 2-D SOM is frequently used for visualizing high-D clusters in manifold/latent space[4][5]. Moreover, the use of 3-D manifolds is not widespread.

A novel 3-D spherical (shell) manifold is proposed for modeling and visualizing high-D data. With all latent nodes on the spherical (shell) manifold equally far away from the center, it captures nicely the sparsity and peripheral property of high-D data[6]. Using the latent nodes on the spherical manifold as class reference vectors, a template-based classifier is also proposed. It is shown that the addition of the third dimension of the spherical manifold dramatically improves classification accuracy over 1-D and 2-D manifolds on two artificial classification datasets with significant class overlap. The spherical manifold classifier is

L. Wang and Y. Jin (Eds.): FSKD 2005, LNAI 3614, pp. 1236–1244, 2005.

also evaluated against the unconstrained Gaussian mixture model (GMM) vector quantizer, and the K-nearest neighbor (KNN) classifier on three real high-D datasets. Experimental results confirm the robustness of spherical manifolds for modeling high-D data.

## 2 Spherical Manifolds

### 2.1 Curse-of-Dimensionality

Data in very high-D space tend to lie entirely at the peripheral of a sample due to the curse-of-dimensionality[6]. To appreciate that this is indeed the case, consider data uniformly distributed within a hypercube in $\mathbb{R}^D : \mathbb{R} \in [-1, 1]$, where $D$ denotes the dimensionality. For $D = 1$ (a line) the fraction $p$ of data lying within the center interval $\mathbb{R}^D : \mathbb{R} \in [-0.5, 0.5]$ is 0.5, for $D = 2$ (a square) this number decreases to 0.25, and for $D = 3$ (a cube), $p$ further decreases to 0.125. The general formula for arbitrary $D$ is $p = 2^{-D}$, from which it can be inferred that even moderate values of $D$ like 20 will result in only a single ($0.9537 \simeq 1$) point out of, say, $10^6$ samples to lie within the central region!

It is clear from the above example that fitting a single multivariate Gaussian distribution to high-D data is inappropriate and actually much worse than fitting a 1-D Gaussian to a 1-D uniformly distributed data, as the Gaussian density assumes the majority of data to be concentrated at the center, contrary to the peripheral property of high-D data. A Gaussian mixture model (GMM)[7] will be able to better model the high-D data by fitting a Gaussian distribution to each dense (i.e. peripheral) regions within the space. However, the uncontrained nature of the GMM makes it very sensitive to initializations; a good fit is obtained if the Gaussian centers are initialized properly and vice-versa. Consequently, a constrained mixture model incorporating some prior knowledge of the characteristics of high-D data is clearly more desirable.

The probabilistic principal surface (PPS)[8] is one such model that explicitly contrains the Gaussians to lie in a pre-defined latent topology. A PPS with 3-D spherical latent topology is introduced for approximating high-D data. The spherical manifold is comprised of nodes evenly distributed on the surface of a sphere. It possesses two attractive characteristics: (1) nodes are distributed on the peripheral and equally far away from the center, just like high-D data, and (2) it is finite but unbounded, which is intuitively suitable for estimating the boundary of high-D data. The goal is to show that the 3-D spherical manifold is capable of modeling high-D data much more accurately then 1-D and 2-D manifolds. The next section briefly describes the probabilistic principal surface model used to construct a spherical manifold.

### 2.2 Probabilistic Principal Surfaces

Principal surfaces (curves)[1] are nonlinear generalizations of principal subspaces (components) that formalizes the notion of a low-D manifold passing through

the 'middle' of a dataset in high-D space. The probabilistic principal surface (PPS)[8], a generalization of the generative topological mapping (GTM)[9][10], is a parametric approximation of principal surfaces. The PPS manifold is comprised of $M$ nodes $\{\mathbf{x}_m\}_{m=1}^{M}$ arranged typically on a uniform topological grid in latent (low-D) space $\mathbb{R}^Q$. The topology is consistently enforced via a generalized linear mapping from each latent node $\mathbf{x}_m$ in $\mathbb{R}^Q$ to it's corresponding data node $\mathbf{f}(\mathbf{x}_m)$ in data (high-D) space $\mathbb{R}^D$ ($D$ is the data dimensionality),

$$\mathbf{f}(\mathbf{x}_m) = \mathbf{W}\phi(\mathbf{x}_m)$$

where $\mathbf{W}$ is a $D \times L$ real matrix and

$$\phi(\mathbf{x}_m) = \left[\, \phi_1(\mathbf{x}_m) \cdots \phi_L(\mathbf{x}_m) \,\right]^T,$$

is the vector containing $L$ latent basis functions $\phi_l(\mathbf{x}) : \mathbb{R}^Q \to \mathbb{R}$, $l = 1, \ldots, L$. The basis functions $\phi_l(\mathbf{x})$ are usually isotropic Gaussians with constant widths. Each data node $\mathbf{f}(\mathbf{x}_m)$ actually corresponds to the mean of a Gaussian probability distribution with noise covariance parameter,

$$\begin{aligned}\mathbf{\Sigma}_m = &\frac{\alpha}{\beta}\sum_{q=1}^{Q}\mathbf{e}_q(\mathbf{x}_m)\,\mathbf{e}_q^T(\mathbf{x}_m) \\ &+ \frac{(D-\alpha Q)}{\beta(D-Q)}\sum_{d=Q+1}^{D}\mathbf{e}_d(\mathbf{x}_m)\,\mathbf{e}_d^T(\mathbf{x}_m) \\ &0 < \alpha < D/Q,\end{aligned}$$

where $\beta^{-1}$ is the global spherical covariance, $\alpha$ is the amount of clamping in the tangential direction, $\{\mathbf{e}_q(\mathbf{x}_m)\}_{q=1}^{Q}$ and $\{\mathbf{e}_d(\mathbf{x}_m)\}_{d=Q+1}^{D}$ are the set of vectors tangential and orthogonal to the manifold at $\mathbf{f}(\mathbf{x}_m)$ in data space, respectively. Figures 1 and 2 show respectively, a 1-D PPS and its noise covariance model for different values of $\alpha$. Notice that the GTM is obtained for $\alpha$=1. It has been shown that the PPS with an orthogonal noise model ($\alpha$<1) yields better manifolds in terms of reconstruction error[8]. The PPS is iteratively computed using a maximum likelihood optimization procedure.

The spherical manifold can be trivially constructed using a PPS with latent nodes $\{\mathbf{x}_m\}_{m=1}^{M}$ arranged regularly on the surface of a sphere in $\mathbb{R}^3$. At the onset, the manifold is intialized to a hyper-sphere in dataspace via the linear transformation $\mathbf{f}(\mathbf{x}_m) = \mathbf{V}\mathbf{x}_m$ where $\mathbf{V} = \left[\, \lambda_1\mathbf{v}_1 \; \lambda_2\mathbf{v}_2 \; \lambda_3\mathbf{v}_3 \,\right]$ is comprised of the three largest eigenvectors $\{\mathbf{v}_q\}_{q=1}^{3}$ (scaled by their corresponding eigenvalues $\lambda_q$) of the data covariance matrix.

## 3 Spherical PPS Classifier

A template-based classifier using spherical manifolds is proposed. This is a significant improvement over the the previously proposed template-based classifier which uses principal curves (1-D manifolds)[11]. A template-based classifier

models data of each class independently of all other classes; a given test sample is classified according to its similarity (distance) to the class templates (reference vectors). The K-nearest neighbor (KNN) classifier can be regarded as a template-based classifier that uses all data samples as class reference vectors, and is frequently used to obtain a rough estimate of the Bayes error. Likewise, a Gaussian mixture model (GMM) can be used to compute reference vectors for each class, known as the GMM classifier. These two related classifiers are therefore used in this paper for benchmarking the proposed classifier. The main goal is to show that the spherical manifold, with its incorporated prior knowledge of high-D data, can model the data better than the unconstrained GMM, thereby contributing to better classification performance.

## 4 Experiments

The spherical manifold classifier (PPS) is compared to the GMM and KNN classifiers on the 5 datasets shown in table 1. For all experiments, a 50/50 training/test random stratified partition scheme was used, and results were obtained on the test set for 10 repeated trials. An isotropic covariance model ($\alpha = 1$) was used for both the PPS and GMM classifier, unless otherwise stated. Validation was found to be unnecessary for all three types of classifiers. All real datasets were normalized to zero mean and unit variance and the same number of reference vectors (manifold nodes) were used for all models where applicable.

**Table 1.** $N$:number of samples, $C$:number of classes, $D$:number of dimensions

| Dataset | $D$ | $C$ | $N$ |
|---|---|---|---|
| 1 gaussian | 8 | 2 | 5000 |
| 2 uniform | 8 | 2 | 5000 |
| 3 letter | 16 | 26 | 20000 |
| 4 ocr | 30 | 26 | 16280 |
| 5 satimage | 36 | 6 | 6435 |

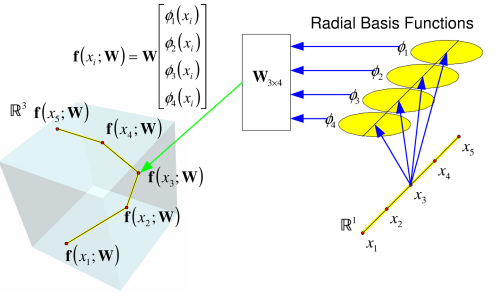


**Fig. 1.** A 1-D PPS in $\mathbb{R}^3$ with 5 nodes and 4 latent bases

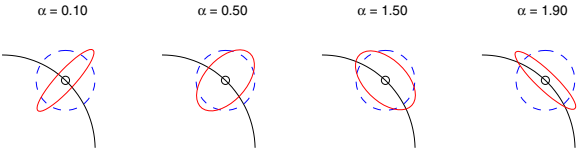


**Fig. 2.** Unoriented covariances $\alpha = 1$ (dashed line) and oriented covariances (solid line) for $\alpha = 0.10, 0.50, 1.50, 1.90$

### 4.1 Artificial Dataset

In this section, the effect of dimensionality (varied from 3 to 8) on the spherical manifold classifier is examined using two artificial datasets with highly overlapping classes. The gaussian dataset[12] is comprised of two equal-sized classes drawn from overlapping Gaussian distributions with zero mean and isotropic variances of 1 and 4, respectively. For comparison, the 1-D and 2-D PPS classifiers were also simulated. Results averaged over 10 trials are given in figure 3. It can be seen that the 1-D and 2-D PPS classifiers were the two worst performers due to their inability to model the complete overlap of the 2 classes in high-D space. While the spherical manifold (PPS-3D) classifier performed much better than the other two PPS classifiers, it was still worse than the GMM and KNN classifiers. This is expected because at lower dimensions, the Gaussian data is mostly concentrated at the core, thereby defying the peripheral assumption of the spherical manifold classifier. However, as data gets increasingly sparse at higher dimensions, both the KNN (for D>6) and GMM (for D>7) classifiers start to deteriorate, whereas the spherical manifold classifier is seen to consis-

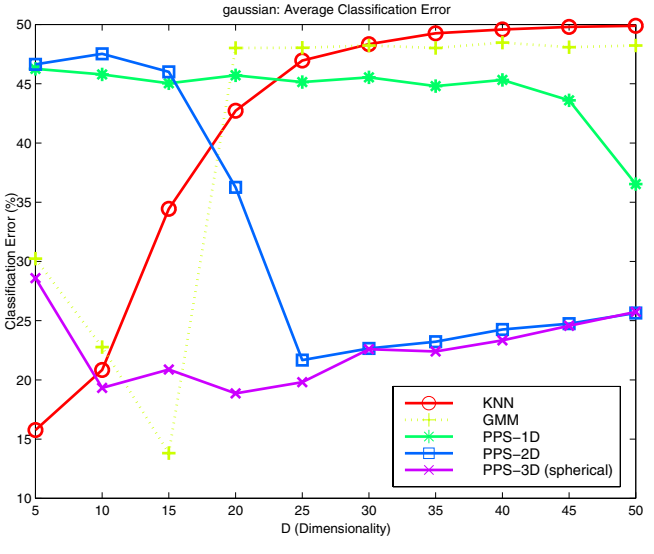


**Fig. 3.** Gaussian: average classification error versus dimensionality

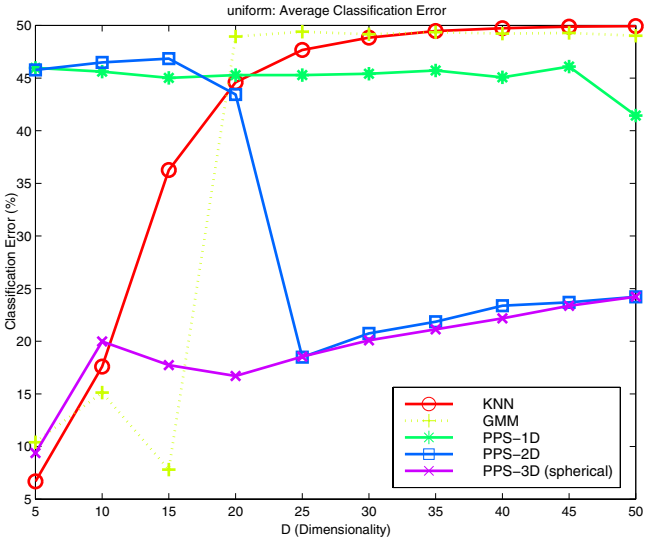


**Fig. 4.** uniform: average classification error versus dimensionality

tently improve with increasing dimensionality! This demonstrates the robustness of the spherical manifold classifier with respect to the curse-of-dimensionality, even where Gaussian data is concerned.

To see if the spherical manifold classifier actually performs better then GMM or KNN on high-D uniformly distributed data, a uniform dataset with features similar to the gaussian dataset was created. The first class is comprised of 2500 samples uniformly drawn from $\mathbb{R}^D : \mathbb{R} \in [-1, 1]$ and the second class contains 2500 samples drawn from $\mathbb{R}^D : \mathbb{R} \in [-2, 2]$, where $D = 8$. The results in figure 4 confirm the superiority of the spherical manifold classifier over other classifiers for high-D ($D > 6$) uniformly distributed data.

### 4.2 Real Dataset

In this section, the performance of the spherical manifold classifier is evaluated on three real high-D datasets–the letter (letter-recognition) dataset from the UCI machine learning database[13], the ocr (handwritten character) dataset provided by the National Institute of Science and Technology, and the remote sensing

**Table 2.** Letter: average classification error

| Classifier | Error (%) | Std. Dev. |
|---|---|---|
| PPS-3D ($\alpha = 0.1$) | 8.08 | 0.17 |
| PPS-3D ($\alpha = 0.5$) | 7.84 | 0.26 |
| PPS-3D ($\alpha = 1$) | **7.82** | 0.24 |
| KNN ($k = 1$) | 8.21 | 0.16 |
| GMM | 13.76 | 0.29 |

**Table 3.** ocr: average classification error

| Classifier | Error (%) | Std. Dev. |
|---|---|---|
| PPS-3D ($\alpha = 0.1$) | 10.68 | 0.36 |
| PPS-3D ($\alpha = 0.5$) | **10.56** | 0.34 |
| PPS-3D ($\alpha = 1$) | 10.60 | 0.29 |
| KNN ($k = 5$) | 11.23 | 0.43 |
| GMM | 16.84 | 1.95 |

**Table 4.** satimage: average classification error

| Classifier | Error (%) | Std. Dev. |
|---|---|---|
| PPS-3D ($\alpha = 0.1$) | 11.36 | 0.35 |
| PPS-3D ($\alpha = 0.5$) | **11.03** | 0.57 |
| PPS-3D ($\alpha = 1$) | 11.16 | 0.50 |
| KNN ($k = 1$) | 10.76 | 0.28 |
| GMM | 14.89 | 0.73 |

satimage dataset from the Elena database[12]. Averaged results on the three datasets are shown in tables 2 to 4. From the tables, it can be concluded that the constrained nature of the spherical manifold results in a much better set of class reference vectors compared to the GMM. Further, its classification performance was comparable to, if not occasionally better than the best KNN classifier.

## 5 Conclusion

From the observation that high-D data lies almost entirely at the peripheral, a 3-D spherical manifold based on probabilistic principal surfaces is proposed for modeling very high-D data. A template-based classifier using spherical manifolds as class templates is subsequently described. Experiments demonstrated the robustness of the spherical manifold classifier to the curse-of-dimensionality. In fact, the spherical manifold classifier performed better with increasing dimensionality, contrary to the KNN and GMM classifiers which deteriorates with increasing dimensionality. The spherical manifold classifier also performed significantly better than the unconstrained GMM classifier on three real datasets, confirming the usefulness of incorporating prior knowledge (of high-D data) into the manifold. In addition to giving comparable classification performance to the KNN on the real datasets, it is important to note that the spherical manifold classifier possess 2 important properties absent from the other two classifiers:

1. It defines a parametric mapping from high-D to 3-D space, which is useful for function estimation within a class, e.g object pose angles (on a viewing sphere) can be mapped to the spherical manifold[14].
2. High-D data can be visualized as projections onto the 3-D sphere, allowing discovery of possible sub-clusters within each class[15]. In fact, the PPS has been used to visualize classes of yeast gene expressions[16].

It is possible within the probabilistic formulation of the spherical manifold to use a Bayesian framework for classification (i.e. classifying a test sample to the class that gives the maximum *a posteriori* probability), thereby coming up with a rejection threshhold. However, this entails evaluating $\mathcal{O}(M)$ multivariate Gaussians, and can be computationally intensive. The PPS classifier has recently been extended to work in a committee, which was shown to improve classification rate on astronomy datasets[17]. Further studies are being done on using the spherical manifold to model data from all classes for visualization of class structure on the sphere, and also for visualizing text document vectors.

## Acknowledgments

This research was supported in part by Army Research contracts DAAG55-98-1-0230 and DAAD19-99-1-0012, NSF grant ECS-9900353, and Nanyang Technological University startup grant SUG14/04.

## References

1. Hastie, T., Stuetzle, W.: Principal curves. Journal of the American Statistical Association **84** (1988) 502–516
2. Mulier, F., Cherkassky, V.: Self-organization as an iterative kernel smoothing process. Neural Computation **7** (1995) 1165–1177
3. Kohonen, T.: Self-Organizing Maps. Springer, Berlin Heidelberg (1995)
4. Jain, A.K., Mao, J.: Artificial neural network for nonlinear projection of multivariate data. In: IEEE IJCNN. Volume 3., Baltimore, MD (1992) 335–340
5. Mao, J., Jain, A.K.: Artificial neural networks for feature extraction and multivariate data projection. IEEE Transactions on Neural Networks **6** (1995) 296–317
6. Friedman, J.H.: An overview of predictive learning and function approximation. In Cherkassky, V., Friedman, J., Wechsler, H., eds.: From Statistics to Neural Networks, Proc. NATO/ASI Workshop, Springer Verlag (1994) 1–61
7. Bishop, C.M.: Neural Networks for Pattern Recognition. 1st edn. Clarendon Press, Oxford. (1995)
8. Chang, K.y., Ghosh, J.: A unified model for probabilistic principal surfaces. IEEE Transactions on Pattern Analysis and Machine Intelligence **23** (2001) 22–41
9. Bishop, C.M., Svensén, M., Williams, C.K.I.: GTM: The generative topographic mapping. Neural Computation **10** (1998) 215–235
10. Bishop, C.M., Svensén, M., Williams, C.K.I.: Developments of the generative topographic mapping. Neurocomputing **21** (1998) 203–224
11. Chang, K.y., Ghosh, J.: Principal curve classifier – a nonlinear approach to pattern classification. In: International Joint Conference on Neural Networks, Anchorage, Alaska, USA, IEEE (1998) 695–700
12. Aviles-Cruz, C., Guérin-Dugué, A., Voz, J.L., Cappel, D.V.: Enhanced learning for evolutive neural architecture. Technical Report Deliverable R3-B1-P, INPG, UCL, TSA (1995)
13. Blake, C., Merz, C.: UCI repository of machine learning databases (1998)

14. Chang, K.y., Ghosh, J.: Three-dimensional model-based object recognition and pose estimation using probabilistic principal surfaces. In: SPIE:Applications of Artificial Neural Networks in Image Processing V. Volume 3962., San Jose, California, USA, SPIE, SPIE (2000) 192–203
15. Staiano, A., Tagliaferri, R., Vinco, L.D.: High-d data visualization methods via probabilistic principal surfaces for data mining applications. In: International Workshop on Multimedia Databases and Image Communication, Salerno, Italy (2004)
16. Staiano, A., Vinco, L.D., Ciaramella, A., Raiconi, G., Tagliaferri, R., Longo, G., Miele, G., Amato, R., Mondo, C.D., Donalek, C., Mangano, G., Bernardo, D.D.: Probabilistic principal surfaces for yeast gene microarray data mining. In: International Conference on Data Mining. (2004) 202–208
17. Staiano, A., Tagliaferri, R., Longo, G., Benvenuti, P.: Committee of spherical probabilistic surfaces. In: International Joint Conference on Neural Networks, Budapest, Hungary (2004)