Model Selection for Gaussian Kernel PCA Denoising

Article *in* IEEE Transactions on Neural Networks and Learning Systems · January 2012

DOI: 10.1109/TNNLS.2011.2178325 · Source: dx.doi.org

2 authors, including:

Kasper Winther Andersen
Copenhagen University Hospital Hvidovre
28 PUBLICATIONS **365** CITATIONS

Model selection for Gaussian kernel PCA denoising

Kasper Winther Jørgensen and Lars Kai Hansen

***Abstract*—We propose kernel Parallel Analysis (kPA) for automatic kernel scale and model order selection in Gaussian kernel principal component analysis (KPCA). Parallel analysis is based on a permutation test for covariance and has previously been applied for model order selection in linear PCA, we here augment the procedure to also tune the Gaussian kernel scale of radial basis function based KPCA. We evaluate kPA for denoising of simulated data and the U.S. postal data set of handwritten digits. We find that kPA outperforms other heuristics to choose the model order and kernel scale in terms of signal-to-noise ratio of the denoised data.**

***Index Terms*—Denoising, kernel principal component analysis, model selection, parallel analysis**

I. Introduction

Kernel principal component analysis (KPCA) is of increasing interest in signal processing, in particular for non-linear signal denoising, see e.g., [1], [2]. While conventional principal component analysis (PCA) denoises signals by projecting onto a linear signal subspace, KPCA denoises by projection onto more general non-linear signal manifolds. A non-linear signal manifold is identified by first mapping the input data to feature space using a non-linear function. In feature space conventional PCA can be applied to extract the main variation in the data by projecting the data onto the subspace spanned by the eigenvectors of the q largest eigenvalues. Finally, the denoised signal is obtained by reconstructing the so-called *pre-image* in input space.

The representer theorem allows effective implementation of the non-linear mapping through inner products represented by the kernel function [3]. Here we will consider the widely used radial basis function aka Gaussian kernel defined by the function $k(x,y) = \exp(-\frac{||x-y||^2}{2\sigma^2})$. The smoothing scale parameter σ plays an important role to the quality of the pre-image as do the number of principal components retained q. Conventional linear PCA is obtained in the limit $\sigma \to \infty$.

A number of heuristics has been suggested to select the scale. Teixeira et al. [1] consider denoising of handwritten digits and they denoise each of the digits in the USPS data set [4] individually and set σ to the maximal distance between each of the training points to the average of all training points. Arias et al. [5] set σ as the average distance to the r_σ-th nearest neighbors, $r_\sigma = \{1, 5\}$. Thorstensen et al. [6] estimate σ as the median of all mutual distances between all training points. Kwok and Tsang [2] set σ to the mean of all mutual training distances. Likewise, for PCA there exist several methods to estimate the number of components q. The Guttman-Kaiser criterion [7] retains all components with eigenvalues greater than the mean. The so-called Scree criterion plots the eigenvalues in decreasing order and finds the 'elbow' of the eigenvalues spectrum. The lack of a likelihood function for KPCA prevents the use of cross-validation approaches proposed in [8], Alzate and Suykens [9] have proposed an alternative loss function that promotes sparsity, and which also with manual inspection of projection distributions allow model selection. Parallel Analysis (PA) [10], [11], [12] is a resampling based alternative for estimation of q in PCA. PA compares the eigenvalues with the distribution of eigenvalues obtained by PCA on data sets distributed according to a null hypothesis of zero covariance. The PA null distributed data sets are obtained by permuting the measurements among the data points within each feature dimension and q is determined as the set of original PCA eigenvalues greater than the 95th percentile of the corresponding null distribution of eigenvalues.

In this communication we adapt PA to KPCA to select the model order q and furthermore extend it to automatically select the smoothing scale parameter σ for Gaussian kernels. In particular we optimize σ to maximize the accumulated eigenvalue advantage of the leading q components compared with PA null data. To our knowledge this is the first general and automatic scheme for tuning q and σ for KPCA.

II. Theory

A. Kernel PCA

Let $\boldsymbol{X}$ define the set of N data points $\boldsymbol{X} = [\boldsymbol{x}_1, ..., \boldsymbol{x}_N]'$ in input space $\mathcal{X}$. Let φ be a non-linear map from $\mathcal{X}$ to feature space $\mathcal{F}$. The kernel matrix $\boldsymbol{K}$ is constructed from the inner products, i.e., $\boldsymbol{K}_{i,j} = k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \varphi(\boldsymbol{x}_j)'\varphi(\boldsymbol{x}_j)$. The eigen decomposition of the centered kernel matrix is found: $\tilde{\boldsymbol{K}} = \boldsymbol{H}\boldsymbol{K}\boldsymbol{H} = \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}'$, where $\boldsymbol{H} = \boldsymbol{I} - \frac{1}{N}\boldsymbol{1}\boldsymbol{1}'$ is the centering matrix, $\boldsymbol{I}$ is the $N \times N$ identity matrix, $\boldsymbol{1} = [1, 1, ..., 1]'$ is a $N \times 1$ vector, $\boldsymbol{U} = [\boldsymbol{\alpha}_1, ..., \boldsymbol{\alpha}_N]$ with

The research is funded by the Danish Lundbeckfonden through CIMBI Center for Integrated Molecular Brain Imaging. Kasper Winther Jørgensen was partly funded by the Lunbeckfonden (grant-nr R48 A4846).

Kasper Winther Jørgensen is with the DTU Informatics, Technical University of Denmark, 2800 Lyngby, Denmark and Danish Research Centre for Magnetic Resonance, Copenhagen University Hospital Hvidovre, 2650 Hvidovre, Denmark (e-mail: kwjo@imm.dtu.dk)

Lars Kai Hansen is with the DTU Informatics, Technical University of Denmark, 2800 Lyngby, Denmark (e-mail: lkh@imm.dtu.dk)

$\boldsymbol{\alpha}_i = [\alpha_{i1}, ..., \alpha_{iN}]'$ is the matrix containing the eigenvectors and $\boldsymbol{\Lambda} = \mathrm{diag}(\lambda_1, ..., \lambda_N)$ contains the corresponding eigenvalues [13].

The kth orthonormal eigenvector of the covariance matrix in the feature space can be shown to be

$$\boldsymbol{v}_k = \sum_{i=1}^{N} \frac{\alpha_{ki}}{\sqrt{\lambda_k}} \tilde{\varphi}(\boldsymbol{x}_i) = \frac{1}{\sqrt{\lambda_k}} \tilde{\boldsymbol{\Phi}} \boldsymbol{\alpha}_k, \tag{1}$$

where $\tilde{\varphi}(\boldsymbol{x}) = \varphi(\boldsymbol{x}) - \bar{\varphi}$ is the centered map with $\bar{\varphi} = \frac{1}{N}\sum_{i=1}^{N} \varphi(\boldsymbol{x}_i)$ and $\tilde{\boldsymbol{\Phi}} = [\tilde{\varphi}(\boldsymbol{x}_1), \tilde{\varphi}(\boldsymbol{x}_2), ..., \tilde{\varphi}(\boldsymbol{x}_N)]$. The projection β_k of the pattern $\boldsymbol{x}$ onto the kth component is then

$$\begin{aligned} \beta_k = \tilde{\varphi}(\boldsymbol{x})' \boldsymbol{v}_k &= \frac{1}{\sqrt{\lambda_k}} \sum_{i=1}^{N} \alpha_{ki} \tilde{\varphi}(\boldsymbol{x})' \tilde{\varphi}(\boldsymbol{x}_i) \\ &= \frac{1}{\sqrt{\lambda_k}} \sum_{i=1}^{N} \alpha_{ki} \tilde{k}(\boldsymbol{x}, \boldsymbol{x}_i), \end{aligned} \tag{2}$$

while the projection $P_q\varphi(\boldsymbol{x})$ of $\varphi(\boldsymbol{x})$ onto the subspace spanned by the first q eigenvectors can be found as

$$\begin{aligned} P_q\varphi(\boldsymbol{x}) &= \sum_{i=1}^{q} \beta_i \boldsymbol{v}_i + \bar{\varphi} \\ &= \sum_{i=1}^{q} \frac{1}{\lambda_i} (\boldsymbol{\alpha}_i' \tilde{\boldsymbol{k}}_x)(\tilde{\boldsymbol{\varphi}} \boldsymbol{\alpha}_i) + \bar{\varphi} \\ &= \tilde{\boldsymbol{\Phi}} \boldsymbol{M} \tilde{\boldsymbol{k}}_x + \bar{\varphi} \end{aligned} \tag{3}$$

with $\boldsymbol{M} = \sum_{i=1}^{q} \frac{1}{\lambda_i} \boldsymbol{\alpha}_i \boldsymbol{\alpha}_i'$ and $\tilde{\boldsymbol{k}}_x = \boldsymbol{H}(\boldsymbol{k}_x - \frac{1}{N}\boldsymbol{K}\boldsymbol{1})$, where $\boldsymbol{k}_x = [k(\boldsymbol{x}, \boldsymbol{x}_1), ..., k(\boldsymbol{x}, \boldsymbol{x}_N)]'$.

B. The pre-image problem

For denoising we are interested in projecting $P_q\varphi(\boldsymbol{x})$ back to input space to recover the denoised pattern $\boldsymbol{z}$ — the so-called pre-image. An exact pre-image of $P_q\varphi(\boldsymbol{x})$ may not exist but a least squares estimate $\boldsymbol{z}$ can be obtained by minimizing

$$||\varphi(\boldsymbol{z}) - P_q\varphi(\boldsymbol{x})||^2 = ||\varphi(\boldsymbol{z})||^2 - 2P_q\varphi(\boldsymbol{x})'\varphi(\boldsymbol{z}) + \mathrm{const.} \tag{4}$$

In this work we will use the original iterative fixed point algorithm proposed by Mika, Schölkopf et al. [3]

$$\boldsymbol{z}_{t+1} = \frac{\sum_{i=1}^{N} \tilde{\gamma}_i \exp(-||\boldsymbol{z}_t - \boldsymbol{x}_i||^2/2\sigma^2)\boldsymbol{x}_i}{\sum_{i=1}^{N} \tilde{\gamma}_i \exp(-||\boldsymbol{z}_t - \boldsymbol{x}_i||^2/2\sigma^2)} \tag{5}$$

with $\gamma_i = \sum_{k=1}^{q} \beta_k \alpha_{ki}$ and $\tilde{\gamma}_i = \gamma_i + \frac{1}{N}(1 - \sum_{j=1}^{N} \gamma_j)$.

C. Kernel Parallel Analysis

We extend the idea of Parallel Analysis (PA) to KPCA including choice of smoothing scale σ for the Gaussian kernel, and we refer to the proposed method as kernel Parallel Analysis (kPA).

In feature space the eigenvalue λ_i for component i of the PCA is compared with the distribution of eigenvalues of null data sets obtained by permuting the data in input space p times. For component i the reference threshold T_i is set to the value of the 95th percentile in the component's distribution of eigenvalues. The number of components q to retain is chosen such that the original data eigenvalues are larger than threshold for all retained components. Note, that both the original data eigenvalues, the reference thresholds, and the number of components q will depend upon the Gaussian scale σ,

$$q(\sigma) = \max_{\lambda_i(\sigma) - T_i(\sigma) > 0} i. \tag{6}$$

A conservative estimate of the signal energy can be obtained as the cumulated difference between the original data eigenvalues and reference threshold levels,

$$E(\sigma) = \sum_{i=1}^{q(\sigma)} \lambda_i(\sigma) - T_i(\sigma). \tag{7}$$

The proposed method chooses the kernel scale σ to *maximize* $E(\sigma)$. The energy is an estimate of the variance of the retained components in kernel space when accounting for the variance of null data. Thus, maximizing the energy in kernel space will maximize the variance of the true signal.

By column-wise permuting the data between samples for a given input dimension we assure that the null-data is drawn from a distribution which has the same marginal distributions as the original data. Furthermore the input dimensions of the null distribution are statistical independent, i.e. the joint probability density function is fully factorized. This means that all manifold structures in input space are destroyed. Note this is a stronger condition than necessary in PA which only requires a null distribution with no covariance. Hence, the corresponding null distribution in feature space is that of a kernel mapped fully factorized distribution in input space with the correct input space marginals. The kernel spectrum of permuted data represents this "null" information. The distribution of the null kernel spectrum, as estimated by repeated permutation, allows us to determine when structure is present - identified in kPA as eigenvalue magnitudes rejected in the distribution of the null spectrum ($p < 0.05$).

Pseudocode for the kPA algorithm is shown in Algorithm 1. The algorithm starts by making p permutations of the data matrix $\boldsymbol{X}$. Then the energy is estimated for a number N_σ of different scales σ and the scale σ_{kPA} with maximal energy is chosen with the corresponding number of components $q(\sigma_{\mathrm{kPA}})$.

The algorithm calculates the eigenvalues of p kernel matrices. This is repeated for the number of scale values investigated N_σ. The calculation of the data point distance matrix used for the kernel matrix generation can be calculated prior to the iteration over scale values and thus reduces the computations needed. The computational complexity of the eigenvalue calculation is in general $O(N^3)$, where N is the number of data points, though there exist iterative methods for finding the first few eigenvalues of large symmetric matrices which are faster [14]. Thus the worst-case time complexity of the kPA algorithm is $O(pN_\sigma N^3)$.

Algorithm 1 Kernel Parallel Analysis

1: Make p permuted replicas of the data matrix by permuting elements in the columns of $\boldsymbol{X}$ independently: $\boldsymbol{X}_j^{\mathrm{perm}} \leftarrow \mathrm{permute}(\boldsymbol{X}), j = 1, ..., p$.
2: Calculate and center the kernel matrix $\boldsymbol{K}^{\mathrm{orig}}$ corresponding to the original data matrix $\boldsymbol{X}^{\mathrm{orig}}$: $\tilde{\boldsymbol{K}}^{\mathrm{orig}} = \boldsymbol{H}\boldsymbol{K}^{\mathrm{orig}}\boldsymbol{H}$.
3: For each permuted dataset calculate and center the kernel matrix $\boldsymbol{K}_j^{\mathrm{perm}}$: $\tilde{\boldsymbol{K}}_j^{\mathrm{perm}} = \boldsymbol{H}\boldsymbol{K}_j^{\mathrm{perm}}\boldsymbol{H}, j = 1, ..., p$.
4: **for** $\sigma = \sigma_{\mathrm{start}} \rightarrow \sigma_{\mathrm{end}}$ **do**
5: Calculate eigenvalues of kernel matrix: $\lambda_i(\sigma) \leftarrow \mathrm{eigval}(\tilde{\boldsymbol{K}}^{\mathrm{orig}}, \sigma)$.
6: For the jth permutation calculate eigenvalue for component i in kernel space: $\tilde{\lambda}_{i,j}(\sigma) \leftarrow \mathrm{eigval}(\tilde{\boldsymbol{K}}_j^{\mathrm{perm}}, \sigma), i = \{1, ..., N\}, j = \{1, ..., p\}$.
7: For component i set the threshold T_i to the 95th percentile of eigenvalues of null data: $T_i \leftarrow$ 95th percentile of $\tilde{\lambda}_{i,*}(\sigma)$.
8: Use eq. (6) to estimate $q(\sigma)$.
9: Use eq. (7) to estimate $E(\sigma)$.
10: **end for**
11: Select the scale σ_{kPA} which maximizes E.
12: Set the number of components to $q(\sigma_{\mathrm{kPA}})$.

III. Experiments

We use two data sets for illustration. In both data sets we create noisy data from a set of clean patterns which allows us to measure the quality of the denoising procedure. This experimental design is adapted from the original kernel PCA paper [3]. The average signal-to-noise ratio (SNR) over data points was calculated and used as performance metric, where SNR is defined as

$$\mathrm{SNR(dB)} = 10 \log_{10} \frac{\langle S^2 \rangle}{\sigma_{\mathrm{res}}^2}, \tag{8}$$

where S is the noise free data and σ_{res}^2 is the variance of the residual noise after de-noising.

We used $p = 49$ in the experiments for this paper, which we found resulted in satisfactory results. While increasing p sometimes give more accurate tests this comes with increased computational times.

A. Semi-circles simulation

An artificial data set was constructed as two equally populated non-intersecting semi-circles placed initially in a 2-dimensional space ($N = 500$). The two dimensions were both rotated to occupy 25 dimensions generating a $d = 50$ dimensional data set. White Gaussian noise ($\sigma_{\mathrm{noise}} = 0.5$) was added in all 50 dimensions and kPA was used to estimate q and σ before denoising. Fig. 1 shows the eigenvalues of the first 10 eigenvectors for the data and the reference threshold level T for null data using $\sigma = 4.5$ and $p = 49$. The shaded area between the two curves where

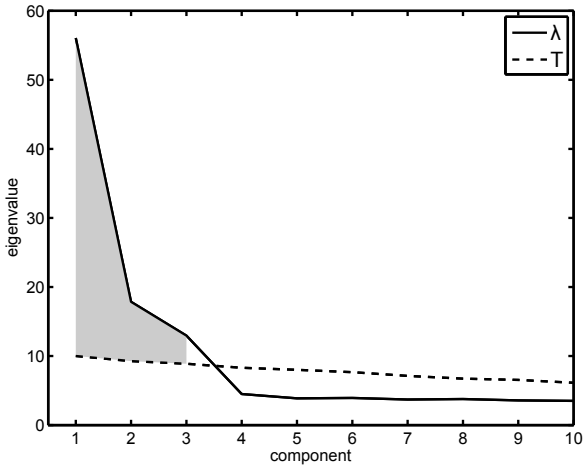

Fig. 1. Simulated data 'semi-circles' ($N = 500, d = 50$). The eigenvalues of the empirical data and the null hypothesis 95% percentile reference level. The cumulated eigenvalue difference - 'the signal energy' E - is the gray area between the two curves. In kPA this area is maximized by tuning the scale of the Gaussian kernel.

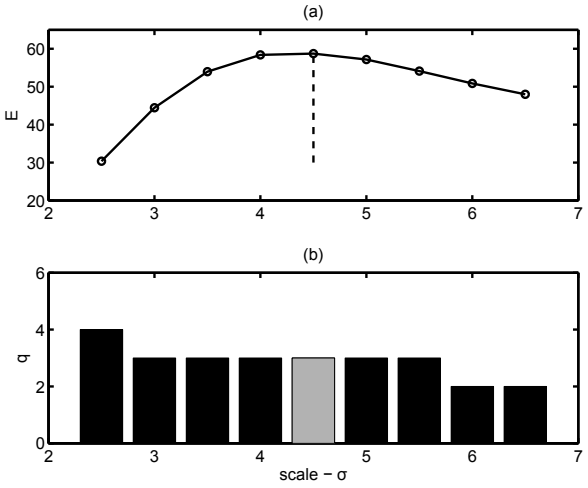

Fig. 2. Simulated data 'semi-circles' ($N = 500, d = 50$). Panel (a) shows $E(\sigma)$ vs. kernel scale σ. Panel (b) shows the corresponding model order q chosen by kPA.

$\lambda_i > T_i$ was next optimized over the single parameter σ. Fig. 2 (a) shows the cumulative eigenvalue difference $E(\sigma)$ as a function of the scale value $\sigma \in [2.5, 6.5]$. The maximum of $E(\sigma)$ is attained at $\sigma = 4.5$. Fig. 2 (b) shows the number of components $q(\sigma)$ as function of the scale, $q = 3$ components is retained for the optimal σ.

To illustrate the simulation data set we use linear PCA on the noise free data in input space and project the data onto the two first components obtained on noise-free data. Fig. 3 presents the projected data before noise was added, after noise was added, and after denoising using the optimized parameters found by kPA.

To test kPA more extensively, conditions were varied. The number of data points was varied as $N = \{250, 500, 750\}$ with noise levels $\sigma_{\mathrm{noise}} = \{0.50, 0.75, 1.00\}$. In all cases data was distributed equally between the two semi-circles. kPA

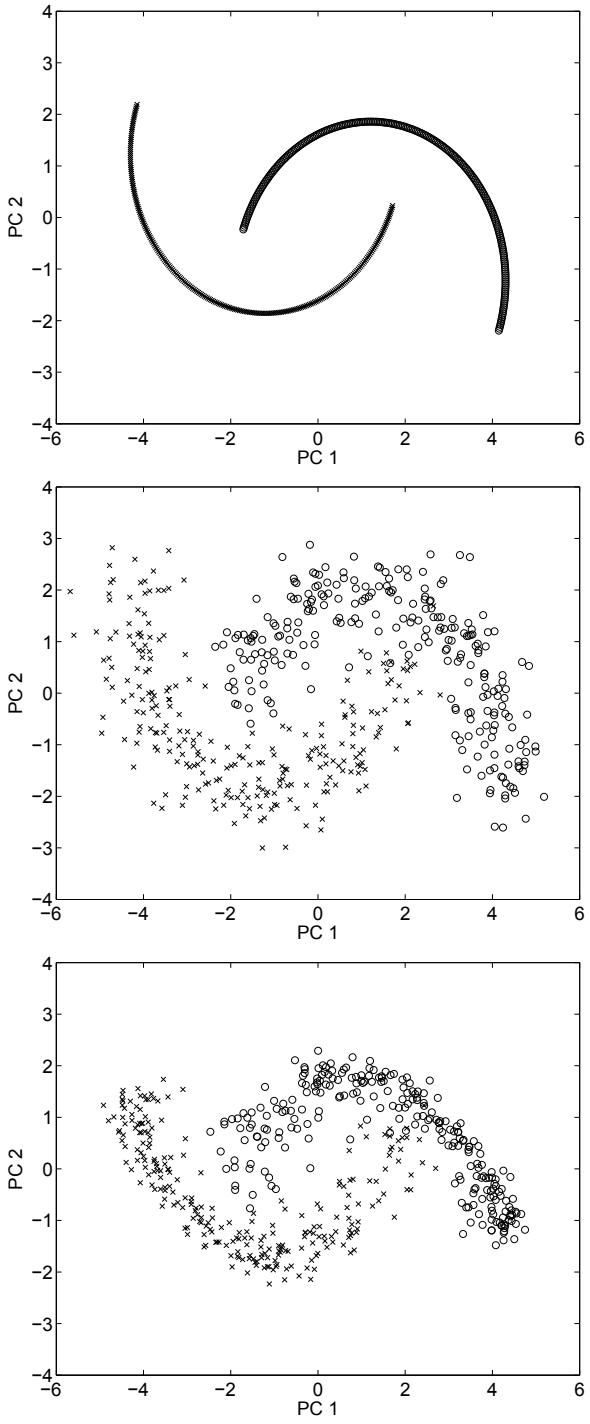

Fig. 3. Simulated data 'semi-circles' ($N = 500, d = 50$). Data in the three panels are all projections on the two first principal components from linear PCA on the noise free in input space. Panel (a) shows the projections of the noise free data set onto the two first principal components; (b) PC projections of data with Gaussian white noise added ($\sigma_{\text{noise}} = 0.5$); (c) PC projections of the denoised data using kernel PCA with $\sigma = 4.5$ and $q = 3$ as determined by kPA.

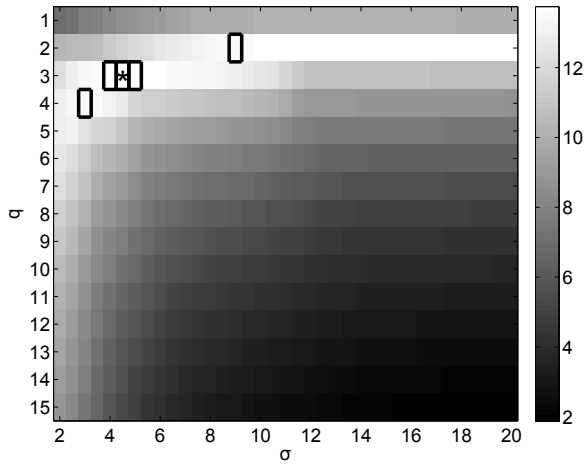

Fig. 4. Simulated data 'semi-circles'. The SNR (dB) landscape as function of scale parameter σ and the number of components q. Here $N = 500$, $d = 50$, and $\sigma_{\text{noise}} = 0.5$. The kPA solution is indicated by the asterisk at $(q, \sigma) = (3, 4.5)$ and the SNR-optimal solution is indicated with boxes.

was used to estimate σ and q and the data was denoised. Table I shows the estimated q and σ along with SNR mean and standard deviations for 100 repetitions of kPA along with the SNR-optimal combination of the parameters (q, σ) found by exhaustive grid search. For all the nine combinations of sample size and noise level the kPA estimated parameters remain constant across the 100 repetitions of the experiment. Likewise, for equal noise level the kPA estimated parameters remain constant across the sample size. Different SNR-optimal parameter combinations were found in the 100 repetitions. For $\sigma_{\text{noise}} = \{0.50, 1.00\}$ the kPA solution is included in the range of SRN-optimal solutions. For $\sigma_{\text{noise}} = 0.75$ the scale takes an intermediate value of the two optima, which causes the SNR to drop significantly compared to the SNR-optimal solution.

Fig. 4 shows the SNR-surface when varying q and σ for $(N, \sigma_{\text{noise}}) = (500, 0.5)$. The solution found by kPA $(q, \sigma) = (3, 4.5)$ is indicated with an asterisk while the SNR-optimal solutions are indicated with boxes.

B. USPS handwritten digits

kPA was next applied to the USPS database of handwritten digits, often used to illustrate kernel PCA [4], [3]. Images were normalized to the range [-1;1] and various number of data points $N = \{100, 200, 300, 400\}$ were used equally distributed among the ten digits. Gaussian noise with $\sigma_{\text{noise}} = \{0.75, 1.00, 1.25\}$ was added. We used kPA to determine σ and q and used these parameters to denoise the data and calculate the SNR of the denoised images. Fig. 5 presents the results (mean and standard deviation of 100 repetitions) in terms of the SNR for the kPA solution and the best solution found in exhaustive grid search. The kPA solution is seen to be robust to the number of data points and the noise level used. Table II reports the q and σ values observed in the 100 repetitions under the varying

TABLE I
SIMULATED DATA 'SEMI-CIRCLES'. THE ESTIMATED KPCA DIMENSION q, THE SCALE σ, AND MEAN AND STANDARD DEVIATION OF SNR (DB) ESTIMATED WITH KPA VS. SNR-OPTIMAL PARAMETERS FOUND BY EXHAUSTIVE GRID SEARCH. THE TEST WAS CARRIED OUT WITH DIFFERENT COMBINATIONS OF DATA SAMPLE SIZE $N = \{250, 500, 750\}$ AND NOISE STANDARD DEVIATION $\sigma_{\text{noise}} = \{0.50, 0.75, 1.00\}$. WE REPORT THE PARAMETERS FOUND IN 100 REPETITIONS.

		kPA			SNR-optimal		
N	σ_{noise}	q	σ	SNR (dB)	q	σ	SNR (dB)
250	0.50	3	4.5	12.35 ±0.28	2 3	9.0 4.0, 4.5, 5.0	12.50 ±0.27
	0.75	2	6.5	8.96 ±0.37	2 3	8.0, 8.5, 9.0 3.5	9.15 ±0.37
	1.00	2	8.0	6.44 ±0.31	1,2	7.0, 7.5, 8.0 8.5, 9.0	6.66 ±0.28
500	0.50	3	4.5	13.75 ±0.23	2 3 4	9.0 4.0, 4.5, 5.0 3.0	13.77 ±0.22
	0.75	2	6.5	10.19 ±0.23	2 3	8.5, 9.0 3.5, 4.0	10.52 ±0.23
	1.00	2	8.0	7.89 ±0.25	2	7.0, 7.5, 8.0 8.5, 9.0	7.89 ±0.25
750	0.50	3	4.5	14.42 ±0.18	3 4	4.5, 5.0, 5.5 3.0	14.46 ±0.18
	0.75	2	6.5	10.71 ±0.20	2 3	9.0 3.5, 4.0	11.17 ±0.22
	1.00	2	8.0	8.60 ±0.21	2	7.5, 8.0 8.5, 9.0	8.61 ±0.21

conditions. For both kPA and SNR-optimal solution the chosen scale value remain constant across N but increases with increasing σ_{noise}. kPA chooses larger scales for all scenarios than the SNR-optimal scale. Both the kPA and SNR-optimal q are increasing with N and decreasing with σ_{noise}. kPA's subspace dimensions q are generally, but not uniformly, smaller than the SNR-optimal solution. The possible tendency to underfit the signal subspace dimension was also noted in [12].

Next, the kPA scale estimate σ was compared with five other heuristics to set the scale: (1) maximal distance between each training point to average of all training points [1], (2) median distance between training points [6], (3) mean distance between training points [2], (4) average distance to the nearest neighbor [5], (5) average distance to the nearest 5 neighbors [5]. For this test the noise level was set to $\sigma_{\text{noise}} = 1.00$ and the number of components was fixed to the q chosen by kPA. Fig. 6 presents the mean and standard deviation of 100 repetitions for $N = \{100, 200, 300, 400\}$ and shows that kPA outperforms the other methods for all sample sizes investigated, with extremely significant p-values.

The computational complexities for the methods used here is: (1) $O(N)$, (2-5) $O(N^2)$, (6) $O(N_\sigma \hat{p} k^2 N)$. The mean computational times t_{kPA} for the kPA method in this experiment were $\{N, t_{\text{kPA}}\} = \{100, 9.7\text{s};\ 200, 27.2\text{s};\ 300, 51.9\text{s};\ 400, 92.1\text{s}\}$, with $N_\sigma = 3, \hat{p} = 50, k = 30$. For methods (1-5) the computational times were $t < 0.1\text{s}$ for all N. The experiments were done on a Intel(R) Core(TM) i7 CPU, 2.67GHz system. So, the improved performance of kPA comes with an increased computational time also. This is due to the fact that kPA is based on permutation tests in kernel space while the other methods work on distances in input space.

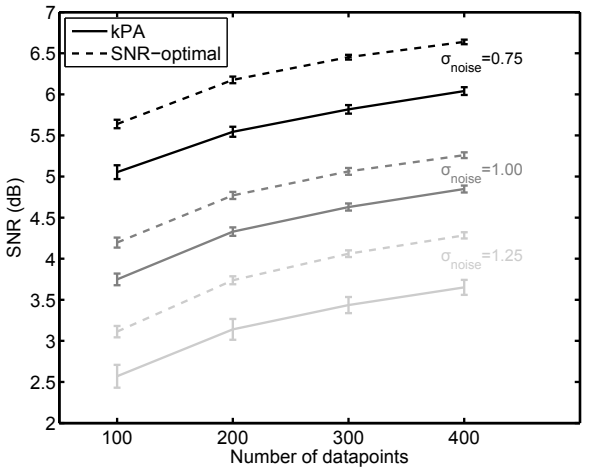

Fig. 5. Mean and standard deviations of SNR (dB) in 100 repetitions of the denoised USPS digits obtained from exhaustive search (SNR-ptimal) and by kPA for three different noise levels.

Finally, we compare methods to choose the number of components q. We compare the kPA solution to the Scree criterion and the Guttman-Kaiser criterion. The Scree criterion was implemented as the first point where the difference between two consecutive eigenvalues in the sorted eigenspectrum was less than 5% of the largest consecutive difference. The Guttman-Kaiser method estimated q as the number of eigenvalues greater than the mean. Fig. 7 plots means and standard deviations of 100 repetitions. kPA significantly outperforms both the Scree and Guttman-Kaiser criteria. In all cases the differences between kPA and the other methods are extremely significant.

TABLE II
THE USPS DATASET. DENOISING WITH DIFFERENT COMBINATIONS OF NUMBER OF DATA POINTS $N = \{100, 200, 300, 400\}$ AND ADDITIVE NOISE STD. $\sigma_{\text{noise}} = \{0.75, 1.00, 1.25\}$. THIS TABLE PRESENTS THE OBSERVED q AND σ IN 100 REPETITIONS OF THE EXPERIMENT USING KPA. IN COMPARISON THE SNR-OPTIMAL SOLUTIONS OBTAINED FROM EXHAUSTIVE GRID SEARCH ARE SHOWN.

		kPA		SNR-optimal	
N	σ_{noise}	q	σ	q	σ
100	0.75	9-14	15-16	16-21	9-11
	1.00	8-14	19	9-14	10-11
	1.25	6-13	21-23	4-10	11-13
200	0.75	14-18	16	27-32	9-10
	1.00	12-17	19	15-20	10-11
	1.25	9-17	22-23	7-13	11-12
300	0.75	16-20	16	34-42	9
	1.00	14-19	19	18-26	9-10
	1.25	11-19	22-23	9-16	10-13
400	0.75	18-22	16	40-46	9
	1.00	15-20	19	21-28	9-10
	1.25	14-20	22-23	11-18	10-12

IV. CONCLUSION

We propose kPA, a generalization of PA to KPCA. kPA completes the widely used Gaussian KPCA as an algorithm,

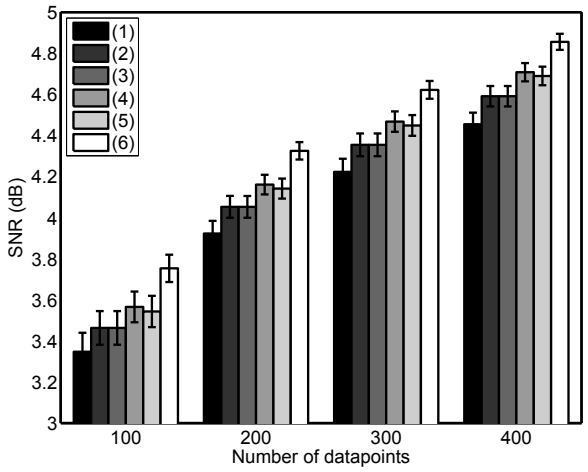

Fig. 6. Mean and standard deviations of SNR (dB) in 100 repetitions of the denoised USPS digits obtained by choosing scale according to the six methods: (1) maximal distance between each training point to average of all training points, (2) median distance between training points, (3) mean distance between training points, (4) average distance to the nearest neighbor, (5) average distance to the nearest 5 neighbors, (6) kPA. Here $\sigma_{\text{noise}} = 1$, results were similar at other noise-levels. For all data cases kPA significantly outperforms the other 5 competing methods.

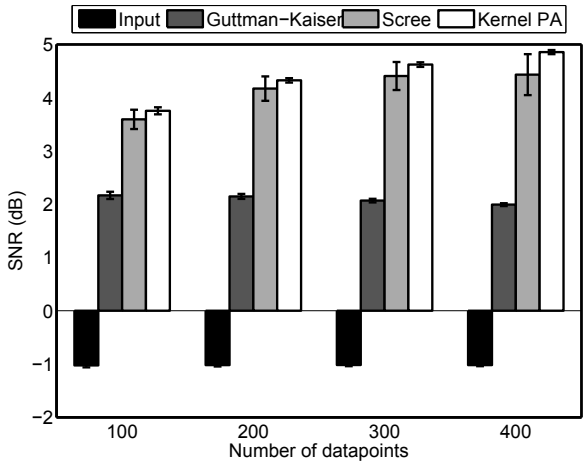

Fig. 7. The USPS dataset. SNR (dB) for the input images and the denoised images using the Guttman-Kaiser criterion, the Scree criterion and kPA for choosing the number q of components to retain. The scale σ was chosen by kPA. Here $\sigma_{\text{noise}} = 1$, results were similar at other noise-levels. kPA is better than the Scree and clearly superior to the Guttman-Kaiser criterion.

as it both solves the subspace dimensionality problem and tunes the smoothing scale parameter. The method optimizes the energy function, which is the accumulated eigenvalue advantage of the leading q components compared with null data. The energy is only a function of the Gaussian kernel smoothing scale, thus the optimization is one dimensional. We used two datasets to extensively test the proposed method, namely the artificial semi-circles data and the USPS dataset of handwritten digits. For the semi-circles data the kPA obtained parameters were shown to be constant across 100 repetitions of the same noise-level and number of data points. Except for $\sigma_{\text{noise}} = 0.75$ the chosen parameters are in the range of SNR-optimal solutions. For $\sigma_{\text{noise}} = 0.75$ the kPA solution takes an intermediate value for the scale parameter. For the USPS dataset we show that the SNR obtained using the kPA solution is robust to the sample size and noise level compared with the SNR-optimal solution. When compared with other heuristics to chose the scale we show that kPA significantly outperform all other methods. Also, when compared to other methods to select the subspace dimensionality the kPA parameter estimates result in significantly higher SNR on the denoised data.

Since kPA is based on permutation tests of the eigenspectra in kernel space the computational time is larger than the other methods used for comparison in this paper. Future work will focus on improving the computational complexity and test kPA with other noise sources.

Acknowledgement

The authors would like to thank M. N. Schmidt for helpful discussions and the anonymous reviewers for their valuable comments and suggestions to improve the manuscript.

REFERENCES

[1] A. Teixeira, A. Tomé, K. Stadlthanner, and E. Lang, "KPCA denoising and the pre-image problem revisited," *Digital Signal Processing*, vol. 18, no. 4, pp. 568 – 580, 2008.

[2] J. T. Y. Kwok and I. W. H. Tsang, "The pre-image problem in kernel methods," *Neural Networks, IEEE Transactions on*, vol. 15, no. 6, pp. 1517–1525, 2004.

[3] S. Mika, B. Schölkopf, A. Smola, K.-R. Müller, M. Scholz, and G. Rätsch, "Kernel PCA and de-noising in feature spaces," in *Advances in Neural Information Processing Systems 11*. MIT Press, 1999, pp. 536–542.

[4] J. J. Hull, "A database for handwritten text recognition research," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 16, no. 5, pp. 550–554, 1994.

[5] P. Arias, G. Randall, and G. Sapiro, "Connecting the Out-of-Sample and Pre-Image Problems in Kernel Methods," *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8, 2007.

[6] N. Thorstensen, F. Segonne, and R. Keriven, "Normalization and preimage problem in Gaussian kernel PCA," in *Image Processing, 2008. ICIP 2008. 15th IEEE International Conference on*. IEEE, 2008, pp. 741–744.

[7] K. A. Yeomans and P. A. Golder, "The Guttman-Kaiser criterion as a predictor of the number of common factors," *Journal of the Royal Statistical Society. Series D (The Statistician)*, vol. 31, no. 3, pp. 221–229, 1982.

[8] L. K. Hansen, J. Larsen, F. Å. Nielsen, S. C. Strother, E. Rostrup, R. Savoy, N. Lange, J. Sidtis, C. Svarer, and O. B. Paulson, "Generalizable patterns in neuroimaging: How many principal components?" *NeuroImage*, vol. 9, no. 5, pp. 534–544, 1999.

[9] C. Alzate and J. Suykens, "Kernel component analysis using an epsilon-insensitive robust loss function," *Neural Networks, IEEE Transactions on*, vol. 19, no. 9, pp. 1583–1598, 2008.

[10] J. L. Horn, "A rationale and test for the number of factors in factor analysis," *Psychometrika*, vol. 30, no. 2, pp. 179 – 185, 1965.

[11] R. D. Ledesma and P. Valero-Mora, "Determining the number of factors to retain in EFA: An easy-to-use computer program for carrying out parallel analysis," *Practical Assessment, Research & Evaluation*, vol. 12, no. 2, 2007.

[12] A. Beauducel, "Problems with parallel analysis in data sets with oblique simple structure," *Methods of Psychological Research*, vol. 6, no. 2, 2001.

[13] B. Schölkopf, A. Smola, and K.-R. Müller, "Nonlinear component analysis as a kernel eigenvalue problem," *Neural Computation*, vol. 10, no. 5, pp. 1299–1319, 1998.

[14] J. Baglama, D. Calvetti, and L. Reichel, "Iterative methods for the computation of a few eigenvalues of a large symmetric matrix," *BIT Numerical Mathematics*, vol. 36, no. 3, pp. 400–421, 1996.