Journal of Machine Learning Research 12 (2011) 2027-2044 Submitted 1/11; Published 6/11

A Cure for Variance Inflation in High Dimensional Kernel Principal Component Analysis

Trine Julie Abrahamsen TJAB@IMM.DTU.DK
Lars Kai Hansen LKH@IMM.DTU.DK
DTU Informatics
Technical University of Denmark
Richard Petersens Plads, 2800 Lyngby, Denmark

Editor: Manfred Opper

Abstract

Small sample high-dimensional principal component analysis (PCA) suffers from variance inflation and lack of generalizability. It has earlier been pointed out that a simple leave-one-out variance renormalization scheme can cure the problem. In this paper we generalize the cure in two directions: First, we propose a computationally less intensive approximate leave-one-out estimator, secondly, we show that variance inflation is also present in kernel principal component analysis (kPCA) and we provide a non-parametric renormalization scheme which can quite efficiently restore generalizability in kPCA. As for PCA our analysis also suggests a simplified approximate expression.

Keywords: PCA, kernel PCA, generalizability, variance renormalization

1. Introduction

While linear dimensionality reduction by principal component analysis (PCA) is a trusted machine learning workhorse, kernel based methods for *non-linear* dimensionality reduction are only starting to find application. We expect the use of non-linear dimensionality reduction to expand in many applications as recent research has shown that kernel principal component analysis (kPCA) can be expected to work well as a pre-processing device for pattern recognition (Braun et al., 2008). In the following we consider non-linear signal detection by kernel PCA followed by a linear discriminant classifier.

In spite of its conceptual simplicity and ubiquitous use, principal component learning in high dimensions is in fact highly non-trivial (see, e.g., Hoyle and Rattray, 2007; Kjems et al., 2001). In the physics literature much attention has been devoted to learnability phase transitions. In PCA there is a sharp transition as function of sample size from *no learning at all* to a regime where the projections become more and more accurate. In the transition regime where learning is still incomplete there is a mismatch between the test and training projections. In Kjems et al. (2001) it was shown that this can be interpreted as a case of *over-fitting* and leads to pronounced *variance inflation* in the training set projections and results in lack of generalization to test data as illustrated in Figure 1.

Variance inflation is of particular concern if PCA is used to reduce dimensionality prior to, for example, a classifier. When the data analytic pipeline is applied to test data the reduced variance of the PCA text projections can lead to significantly reduced performance. Fortunately, the bias can

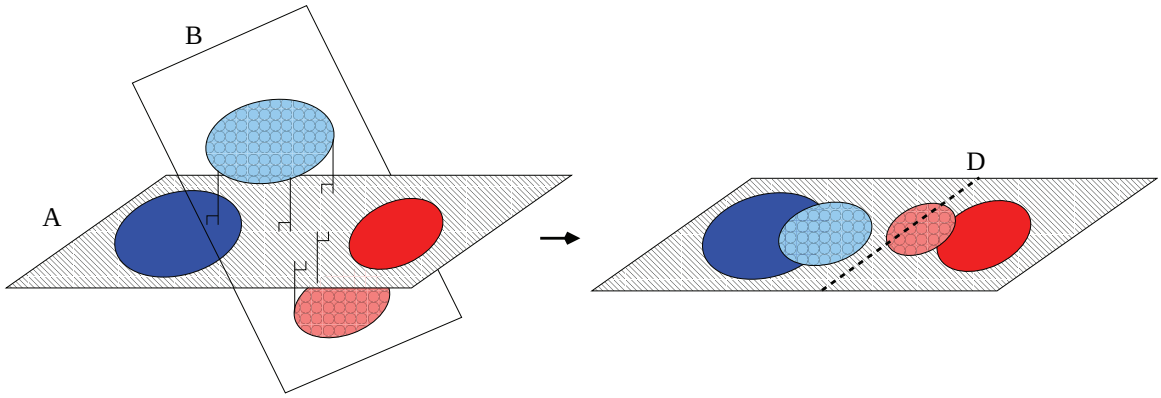

Figure 1: Illustration of the variance inflation problem in PCA. Because PCA maximizes variance, small data sets in high dimensions will be overfitted. When the PCA subspace (A) is applied to a test data set (B) the projected data will have smaller variance. This leads to lack of generalizability if the training data is used to train a classifier, say a linear discriminant (D). In Kjems et al. (2001) this problem was noted and it was shown that the necessary renormalization can be estimated in a leave-one-out procedure

be reduced effectively by a leave-one-out (LOO) scale renormalization of the PCA test projections to restore generalizability (Kjems et al., 2001). In this paper we pursue several extensions of this result. We give a straightforward geometric analysis of the projection problem that suggests a computationally less intensive approximate cure than the one originally proposed by Kjems et al. (2001). Next, we proceed to investigate the issue in the context of *kernel* based unsupervised dimensionality reduction. We show in both simulation and in real world data (USPS handwritten digits and functional MRI data) that variance inflation also happens in kPCA and basically for the same reasons as in PCA. We then provide an extension to the LOO procedure for kPCA which can cope with potential non-Gaussian distributions of the kPCA projections, and finally we propose a simplified approximate renormalization scheme.

2. Generalizability in PCA

The most complete theoretical picture of principal component learning is presented by Hoyle and Rattray (2007), which builds on and extends earlier work by, for example, Biehl and Mietzner (1994), Hoyle and Rattray (2004c), Johnstone (2001), Reimann et al. (1996), and Silverstein and Combettes (1992). Hoyle and Rattray (2007) consider a general PCA model with a multidimensional normal distributed signal that emerges from an isotropic noise background as the sample size increases. The stabilization of a given principal component happens at a given sample size and takes the form of a phase transition. For small sample sizes -below the phase transition point - the training set principal component eigenvectors are in completely random directions in space and there is no learning at all. Then, as the sample size increases, the first principal component stabilizes, and for even larger sample sizes the second, and so forth. Sharp transitions are strictly present only in a limit where both dimensionality and sample size are infinite with a finite ratio $\alpha = N/D$, but the theoretical results are very accurate at realistic dimensions as seen in Figure 2. The location of the first

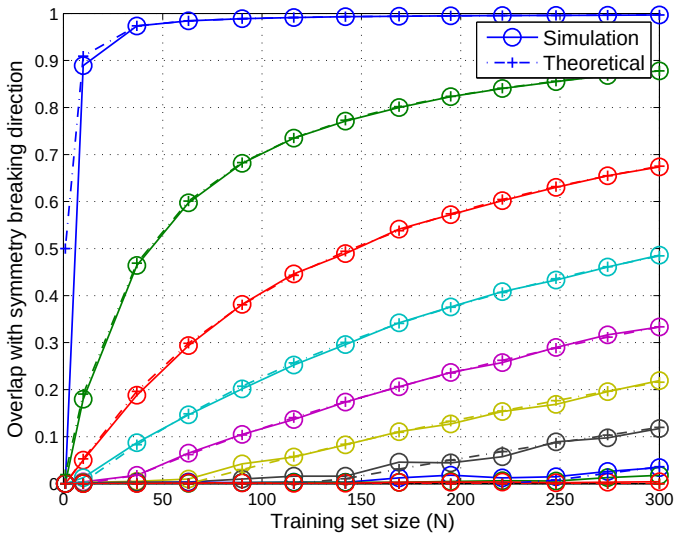

Figure 2: Phase transitions in PCA. Simulated data was created as $\boldsymbol{x} = \eta \boldsymbol{u} + \boldsymbol{\epsilon}$, with a normal distributed signal of unit strength $\eta \sim N(0,1)$, embedded in i.i.d. normal noise $\boldsymbol{\epsilon} \sim N(0, \sigma^2 \mathbf{1})$. In this simulated data set we show the phase transition like behavior of the overlap (the mean square of the projection) of the first PCA eigenvector and the signal direction $\boldsymbol{u}$. The input space has dimension $D = 1000$, and the curves are for 10 values of signal to noise within the interval $\sigma \in [0.01, 0.5]$. For a noise level of, for example, $\sigma = 0.17$ (black curves) there is a sharp transition both in the theoretical curve (dash/cross) and the experimental curve (full/circle) around $N = 120$ examples.

phase transition depends on the signal variance to noise variance ratio (SNR). The theoretical result provides a *mean bias* for a specific model, hence, cannot directly be used to restore generalizability in a given data set.

Now, what happens to the generalization performance of PCA in the noisy region? The PCA projections will be offset by different angles depending on how severe the given component is affected by the noise. Because of the bias the test projections will follow different probability laws than the training data, typically with much lower variance. Hence, if we train a classifier on the training projections the classifier will make additional errors on the test set as visualized in Figure 1.

In the case of PCA the subspace projections are uncorrelated, hence, it is meaningful to renormalize them independently. Assuming approximate normality, a simple affine transformation suffices. The scale factor is simply the ratio of the standard deviations of the training and test projections and can be estimated by a leave-one-out procedure (Kjems et al., 2001). However, since the LOO procedure involves the computation of N SVD's of an $(N-1) \times (N-1)$ matrix, it is of interest to find a simplified estimate.

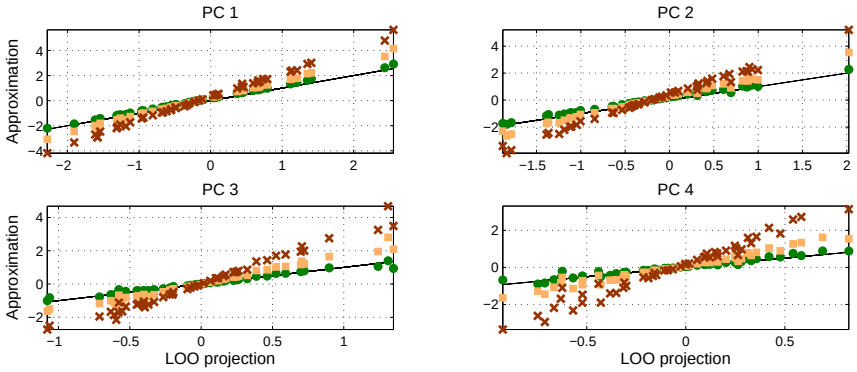

Figure 3: Approximating the leave-one-out (LOO) procedure. Here we simulate data with four normal independent signal components, $\boldsymbol{x} = \Sigma_{k=1}^{4} \eta_k \boldsymbol{u}_k + \boldsymbol{\epsilon}$ of strengths $(1.4, 1.2, 1.0, 0.8$, embedded in i.i.d. normal noise $\boldsymbol{\epsilon} \sim N(0, \sigma^2 \mathbf{1})$, with $\sigma = 0.2$. The dimension was $D = 2000$ and the sample size was $N = 50$. In the four panels we show the training set projections (red crosses), the projections corrected for the theoretical mean overlap (Hoyle and Rattray, 2007) (yellow squares) and the geometric approximation in Equation (1) (green dots) versus the exact LOO projections (black line).

Let $\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$ be N training data points in a D dimensional input space $\mathcal{X}$ (see notation),[1] we consider the case $N \ll D$. The LOO step for the N'th point $\boldsymbol{x}_N$ concerns projecting onto the PCA eigenvectors derived from the subset $\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{N-1}\}$. Define the orthogonal and parallel components of the test point, $\boldsymbol{x}_N = \boldsymbol{x}_N^{\perp} + \boldsymbol{x}_N^{\parallel}$, relative to the subspace spanned by the training data. As the PCA eigenvectors with non-zero variance are all in the span of the training data we obtain

$$\boldsymbol{u}_{N-1,k}^T \cdot \boldsymbol{x}_N = \boldsymbol{u}_{N-1,k}^T \cdot \boldsymbol{x}_N^{\parallel} \, ,$$

where $\boldsymbol{u}_{N-1,k}$ is the k'th eigenvector of the LOO training set. Assuming that the changes in the PCA eigenvectors going from sample size N to $N-1$ are small, we can approximate the test projections as

$$\boldsymbol{u}_{N-1,k}^T \cdot \boldsymbol{x}_N = \boldsymbol{u}_{N-1,k}^T \cdot \boldsymbol{x}_N^{\parallel} \approx \boldsymbol{u}_{N,k}^T \cdot \boldsymbol{x}_N^{\parallel} \, , \tag{1}$$

where $\boldsymbol{u}_{N,k}$ is the k'th eigenvector on the full sample. The approximation introduces a small error of order $1/N$ as discussed in detail in the Appendix and further illustrated in a simulation data set in Figure 3. Note that the orthogonal projections $\boldsymbol{x}_N^{\parallel}$ of the N points may be calculated from the inverse matrix of the inner products of all data points, in N steps each of a cost scaling as N^2, thereby achieving a computational burden which scales as N^3 rather than the N^4 scaling for an exact LOO procedure proposed in Kjems et al. (2001).

1. Bold uppercase letters denote matrices, bold lowercase letters represent column vectors, and non-bold letters denote scalars. $\boldsymbol{a}_j$ denotes the *j'th* column of $\boldsymbol{A}$, while a_{ij} denotes the scalar in the *i'th* row and *j'th* column of $\boldsymbol{A}$. Finally $\mathbf{1}_{NN}$ is a $N \times N$ matrix of ones.

3. Renormalization Cure for Variance Inflation in kernel PCA

The statistical properties of kernel PCA have also been studied extensively by Blanchard et al. (2007), Hoyle and Rattray (2004a), Hoyle and Rattray (2004b), Mosci et al. (2007), Shawe-Taylor and Williams (2003) and Zwald and Blanchard (2006), but to our knowledge the geometry of generalization for kPCA has not been discussed in the extremely ill-posed case $N \ll D$.

To better understand the variance inflation problem in relation to kPCA let us recapitulate some basic aspects of this non-linear dimensional reduction technique.

Let $\mathcal{F}$ be the Reproducing Kernel Hilbert Space (RKHS) associated with the kernel function $k(\boldsymbol{x},\boldsymbol{x}') = \varphi(\boldsymbol{x})^T\varphi(\boldsymbol{x}')$, where $\varphi : \mathcal{X} \mapsto \mathcal{F}$ is a possibly non-linear map from the D-dimensional input space $\mathcal{X}$ to the high dimensional (possibly infinite) feature space $\mathcal{F}$. In kPCA the PCA step is carried out in the feature space, $\mathcal{F}$, mapped data (Schölkopf et al., 1998). However, as $\mathcal{F}$ can be infinite dimensional we first apply the kernel trick allowing us to work with the Gram matrix of inner products. Let $\{\boldsymbol{x}_1,\ldots,\boldsymbol{x}_N\}$ be N training data points in $\mathcal{X}$ and $\{\varphi(\boldsymbol{x}_1),\ldots,\varphi(\boldsymbol{x}_N)\}$ be the corresponding images in $\mathcal{F}$. The mean of the φ-mapped data points is denoted $\bar{\varphi}$ and the 'centered' images are given by $\tilde{\varphi}(\boldsymbol{x}) = \varphi(\boldsymbol{x}) - \bar{\varphi}$. The kPCA is performed by solving the eigenvalue problem $\widetilde{\boldsymbol{K}}\boldsymbol{\alpha}_i = \lambda_i\boldsymbol{\alpha}_i$ where the centered kernel matrix, $\widetilde{\boldsymbol{K}}$, is defined as

$$\widetilde{\boldsymbol{K}} = \boldsymbol{K} - \frac{1}{N}\mathbf{1}_{NN}\boldsymbol{K} - \frac{1}{N}\boldsymbol{K}\mathbf{1}_{NN} + \frac{1}{N^2}\mathbf{1}_{NN}\boldsymbol{K}\mathbf{1}_{NN} \; . \tag{2}$$

The projection of a φ-mapped test point onto the *i'th* component is given by

$$\beta_i = \tilde{\varphi}(\boldsymbol{x})^T\boldsymbol{v}_i = \sum_{n=1}^{N}\alpha_{in}\tilde{\varphi}(\boldsymbol{x})^T\tilde{\varphi}(\boldsymbol{x}_n) = \sum_{n=1}^{N}\alpha_{in}\tilde{k}(\boldsymbol{x},\boldsymbol{x}_n) \; , \tag{3}$$

where $\boldsymbol{v}_i$ is the *i'th* eigenvector of the feature space covariance matrix and the $\boldsymbol{\alpha}_i$'s have been normalized. The centered kernel function can be found as $\tilde{k}(\boldsymbol{x},\boldsymbol{x}') = k(\boldsymbol{x},\boldsymbol{x}') - \frac{1}{N}\mathbf{1}_{1N}\boldsymbol{k}_{\boldsymbol{x}} - \frac{1}{N}\mathbf{1}_{1N}\boldsymbol{k}_{\boldsymbol{x}'} + \frac{1}{N^2}\mathbf{1}_{1N}\boldsymbol{K}\mathbf{1}_{N1}$, where $\boldsymbol{k}_{\boldsymbol{x}} = [k(\boldsymbol{x},\boldsymbol{x}_1),\ldots,k(\boldsymbol{x},\boldsymbol{x}_N)]^T$. The projection of $\varphi(\boldsymbol{x})$ onto the first q principal components will in be denoted $P_q(\boldsymbol{x})$.

In the following we focus on a Gaussian kernel of the form $k(\boldsymbol{x},\boldsymbol{x}') = \exp(-\frac{1}{c}||\boldsymbol{x}-\boldsymbol{x}'||^2)$, where c is the scale parameter controlling the non-linearity of the kernel map. By the centering operation, PCA is the obtained in the limit when $c \to \infty$. Thus for large values we expect variance inflation to be present due the reasons discussed above. What happens in the non-linear regime with a finite c? To answer this question we analyze the LOO scenario for kPCA.

Consider the squared distance $||\boldsymbol{x}_n - \boldsymbol{x}_N||^2$ in the exponent in the Gaussian kernel for some training set point $\boldsymbol{x}_n$ and a test point $\boldsymbol{x}_N$. If we split the test point in the orthogonal components as above with respect to the subspace spanned by the training set we obtain,

$$||\boldsymbol{x}_n - \boldsymbol{x}_N||^2 = ||\boldsymbol{x}_n - \boldsymbol{x}_N^{||}||^2 + ||\boldsymbol{x}_N^{\perp}||^2 \; .$$

Inserting this expression in the Gaussian kernel in Equation (3) it is seen that the test projection acquire a common factor $\exp\left(-\frac{1}{c}||\boldsymbol{x}_N^{\perp}||^2\right)$:

$$\beta_i(\boldsymbol{x}_N) = \sum_{n=1}^{N-1}\alpha_{in}\tilde{k}(\boldsymbol{x}_N,\boldsymbol{x}_n) = \exp\left(-\frac{1}{c}||\boldsymbol{x}_N^{\perp}||^2\right)\sum_{n=1}^{N-1}\alpha_{in}\tilde{k}(\boldsymbol{x}_N^{||},\boldsymbol{x}_n) \; ,$$

which can be arbitrary small for small values c, that is, in the non-linear regime.

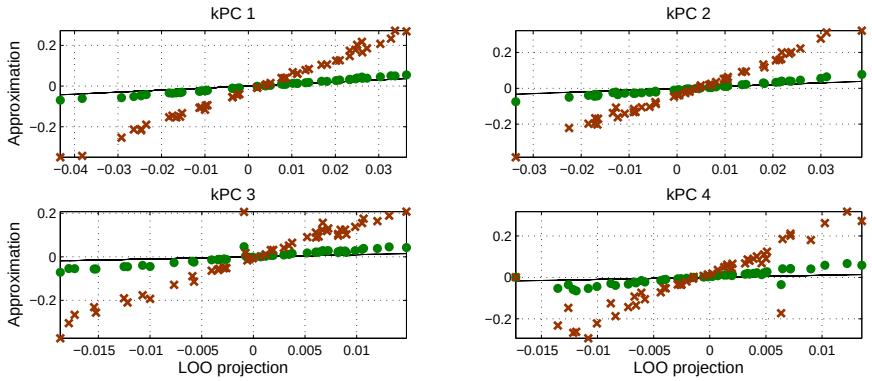

Figure 4: Approximating the leave-one-out (LOO) procedure for kPCA. We simulate a data set with four normal independent signal components, $\boldsymbol{x} = \Sigma_{k=1}^{4} \eta_k \boldsymbol{u}_k + \boldsymbol{\epsilon}$ of strengths $(1.4, 1.2, 1.0, 0.8$, embedded in i.i.d. normal noise $\boldsymbol{\epsilon} \sim \mathcal{N}(0, \sigma^2 \mathbf{1})$, with $\sigma = 0.2$. The dimension was chosen $D = 2000$ and the sample size was $N = 50$. In the four panels we show the four kPCA component's training set projections (red crosses), and the result of applying the point wise correction factor $\exp(\frac{1}{c}||\boldsymbol{x}_N^{\perp}||^2)$ for the lost orthogonal projection (green dots) versus the exact LOO kPCA test projections (black).

For a coordinate-wise LOO renormalization procedure we thus propose to compute N test projections by repeated kPCA on the $N-1$ sized sub training sets. However, compared to the PCA case we face two additional challenges, namely the potentially strongly non-Gaussian distributions and component dependencies.

To check for dependency we appeal to simple pairwise permutation test of significant mutual information measure (see, e.g., Moddemeijer, 1989). If the null hypothesis is rejected for a given set of components we cannot expect coordinate-wise renormalization to be effective. If, on the other hand, the kernel PCA projections pass the independence test we can proceed to renormalize the components individually. In the following we will assume that a coordinate-wise approach is acceptable. First, as a simple approximation to the full LOO we consider adjusting for the common scaling factor due to the lost orthogonal projection. This may indeed provide for viable approximation as seen in Figure 4.

To address the second challenge, namely the potential non-normality we propose to generalize the affine scaling method of Kjems et al. (2001) by a non-parametric procedure. Assume that there exists a monotonic transformation between the N training and N LOO test set projections. The problem of calibrating for an unknown monotone transformation is a common operation in image processing, and is used, for example, to transform the gray scale of an image in order to standardize the pixel histogram (Gonzalez and Wintz, 1977). Equalizing two equal sized samples, simply involves sorting both and assigning the sorted test projections the sorted values of the training projections, this procedure is easily seen to equalize the histograms without changing the level sets (relative ordering) of the LOO test projections. In Figure 5 a simple 1-dimensional data set is used to illustrate the equalization procedure. The training set clearly contains two classes. However, due to variance inflation (induced by, for example, kernel PCA) the test set does not follow the same

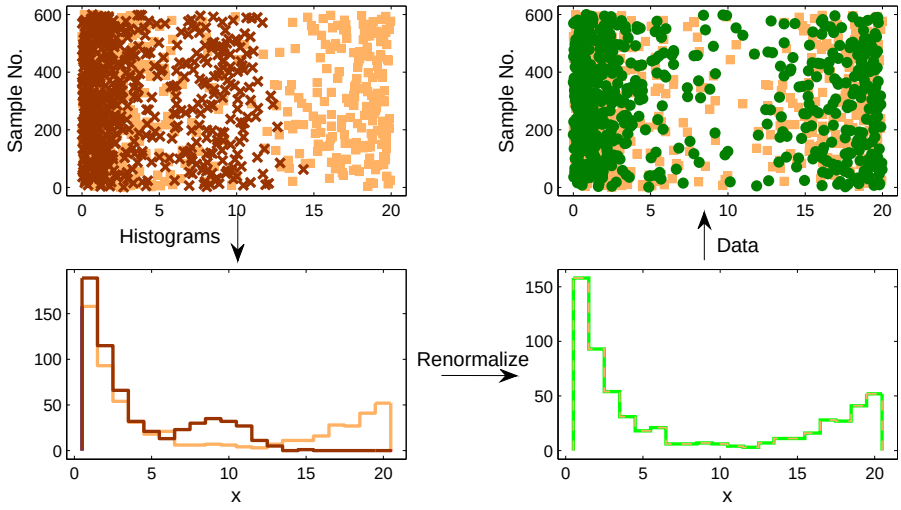

Figure 5: Illustration of renormalization by histogram equalization. The left panel shows the training set (yellow squares) and original test set (red crosses) and their respective histograms. The histograms are then equalized as seen in the right panel, where the green dots are the renormalized test data. The renormalization clearly restores the variation of the test set.

distribution, and may potentially lead to a high misclassification rate. The right panel of the figure shows how histogram equalization restores generalizability.

Technically, the transformation may be described as follows. Let $H(f)$ be the cumulative distribution of values f of a given kPCA projection of the training set. Let the test set projections on the same component for N_{test} samples take values $g(m)$. Let $I(m)$ be the index of sample m in a sorted list of the test set values. Then the renormalized value of the test projection m is

$$\widetilde{g(m)} = H^{-1}(I(m)/N_{\text{test}}) \, .$$

The test set projections can be obtained by the simple relation

$$\widetilde{g(m)} = f_{\text{sort}}(I(m)) \, , \tag{4}$$

where f_{sort} is the sorted list of training set projections. The algorithm for approximate renormalization is summarized in Algorithm 1.[2]

4. Evaluation of the Proposed Cure in Classification Problems

In the following we evaluate the non-parametric exact LOO correction scheme when kPCA is used as a dimensional reduction step in simulated and real classification data sets.

2. We thank the reviewers for pointing out that while non-normality is expected in the case of kPCA, non-normality may also appear in PCA calling for application of the proposed non-parametric renormalization scheme in this case.

Algorithm 1 Approximate renormalization in kernel PCA

Require: $\boldsymbol{X}_{tr}$ and $\boldsymbol{X}_{te}$to be $N_{tr} \times D$ and $N_{te} \times D$ respectively
  Compute $\widetilde{\boldsymbol{K}}_{tr}$ using Equation (2) and find the eigenvectors, $\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_q$
  for $i = 1$ to N_{tr} **do**
    $\boldsymbol{f}_{tr}^{i,:} \leftarrow P_q(\boldsymbol{x}_{tr}^{i,:}) = \tilde{\boldsymbol{k}}_{x_i}^T \boldsymbol{\alpha}^{i:q}$ {see Equation (3)}
  end for
  for $j = 1$ to N_{te} **do**
    $\boldsymbol{f}_{te}^{j,:} \leftarrow P_q(\boldsymbol{x}_{te}^{j,:}) = \tilde{\boldsymbol{k}}_{x_j}^T \boldsymbol{\alpha}^{i:q}$ {see Equation (3)}
  end for
  for $d = 1$ to q **do**
    $[\boldsymbol{f}_{sort}, \ ] \leftarrow \text{sort}(\boldsymbol{f}_{tr}^{:,d})$ {ascending order}
    $[\ , I] \leftarrow \text{sort}(\boldsymbol{f}_{te}^{:,d})$ {ascending order}
    if $N_{tr} = N_{te}$ **then**
      $\boldsymbol{h} \leftarrow \boldsymbol{f}_{sort}$
    else $\{N_{tr} \neq N_{te}\}$
      $\boldsymbol{h} \leftarrow \text{spline}\big([1 : N_{tr}], \boldsymbol{f}_{sort}, \text{linspace}(1, N_{tr}, N_{te})\big)$ {interpolate to create N_{te} values of $\boldsymbol{f}_{sort}$ in the interval $[1 : N_{tr}]$}
    end if
    for $n = 1$ to N_{te} **do**
      $\tilde{\boldsymbol{g}}_{te}^{I(n),d} \leftarrow \boldsymbol{h}^{n,d}$ {renormalized test data in the principal subspace, see Equation (4)}
    end for
  end for

4.1 Simulated Data

To get some insight into the non-linear regime, we design a synthetic data set containing two 2-dimensional semi-circular clusters which cannot be separated linearly (cf., Jenssen et al., 2006). Gaussian noise is added to one of the clusters, and the data is further embedded in 1000 'noise dimensions'. The basis is changed so that the 2D signal space occupies a general position. The noise is as earlier assumed i.i.d. with variance σ^2. The assignment variable is $t = 0, 1$, and in the experiments the data set is assumed unbalanced with $p(t = 0) = 0.6$.

In Figure 6 we show in the left panel a linear discriminant trained on the training set projections in a data set of $N = 500$ in $D = 1000$ dimensions. The role of the non-linearity as controlled by the parameter c in the Gaussian kernel is investigated in Figure 7 for a simulation setup similar to Figure 6. As seen the inflation problem dramatically amplifies as non-linearity increases. Finally, Figure 8 shows how renormalization improves the learning curve for the same problem.

4.2 USPS Handwritten Digit Data

The USPS handwritten digit benchmark data set is often used to illustrate unsupervised and supervised kernel methods. The USPS data set consists of $D = 16 \times 16 = 256$ pixels handwritten digits.[3] For each digit we randomly chose 10 examples for training and another 10 examples for testing. The scale was chosen as the 5th percentile of the mutual distances of the data points leading to $c \approx 120$,

3. The USPS data set is described by Hull (1994) and can be downloaded from `www.kernel-machines.org`.

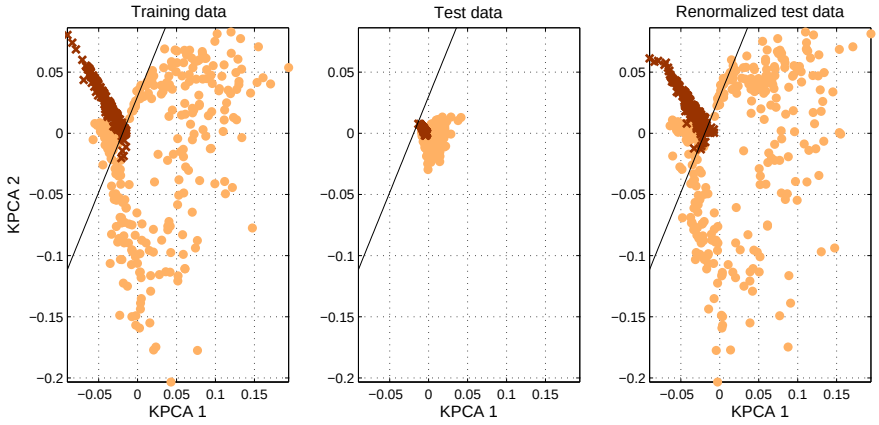

Figure 6: An unbalanced two cluster data set showing a pronounced variance inflation problem in the projections of the test data in the middle panel. In the right panel we have applied the cure based on non-parametric renormalization to equalize training and test projections using histogram equalization. The linear discriminant performs close to the optimal Bayes rate after non-parametric renormalization. The sample size is $N = 500$ in $D = 1000$ dimensions and the SNR is 10. The training error rate is 0.002 while the uncorrected test error rate is 0.4. Renormalization reduces the test error to 0.002.

and the number of principal components was chosen so 85% of the variance was contained in the principal subspace leading to around $q = 57$ PCs to be included.

The first step is to submit the data to the mutual information permutation test. For every pair of principal components a permutation test with 1000 permutations was performed in order to test the null hypothesis of the two given components being independent. Using a $\rho = 0.05$ significance level, we find that the null hypothesis can only be rejected for approximately 2% of the principal component pairs when not using Bonferroni correction. The combinations for which the null hypothesis can be rejected are equally distributed across the principal components. Since the expected number of rejected tests at the given confidence level is 5%, hence, we can safely proceed with the coordinate-wise renormalization process.

In the q dimensional principal subspace the projections of the test set are renormalized to follow the training set histogram. We chose in these experiments for demonstration to classify digit 8 versus the rest. A linear discriminant classifier was trained on the kernel PCA projections of the training set, and the classification error was found using both the conventional kernel PCA projections of the test set and their renormalized counterparts. In order to compare the two methods, the procedure was repeated 300 times using random training and test sets. While classification based on the conventional projections resulted in a mean classification error rate ($\pm$ 1 std) of 0.06 ± 0.01, using the renormalized projections lowered the error rate to 0.05 ± 0.02. A paired t-test showed that this reduction is highly significant ($p = 2.0875 \cdot 10^{-11}$).

Figure 9 shows an example of the projections before and after renormalization. The axis are fixed across the two methods. The top row clearly illustrates the inflation problem for conventional kPCA. Furthermore, due to the imbalanced nature of the data set, the inflation causes a high misclas-

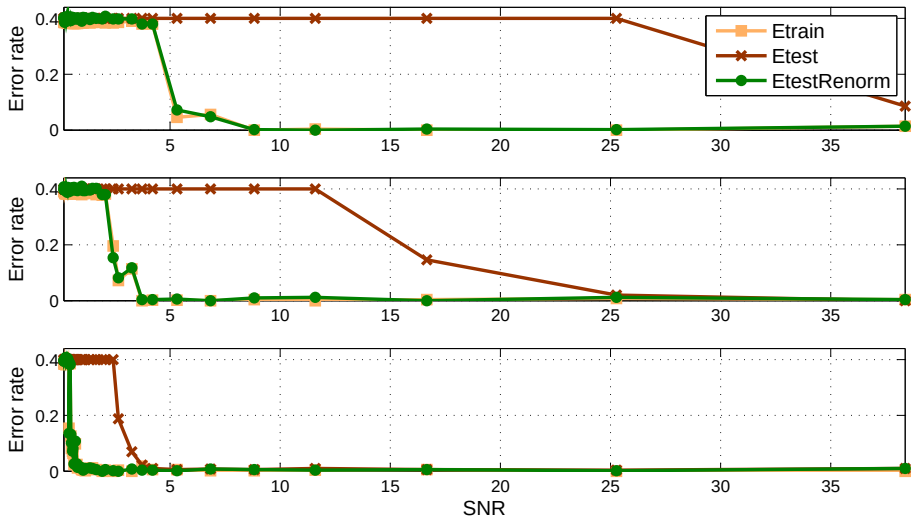

Figure 7: The role of non-linearity on the variance inflation problem. We carry out three experiments at different values of the Gaussian kernel scale parameter (top to bottom: $c = 0.05$, $c = 0.1$, $c = 0.5$). We show classification errors as a function of SNR. The linear discriminant performs close to the optimal Bayes rate after the renormalization operation in all cases, while the un-renormalized systems suffers from poor generalizability. The sample size is $N = 500$ and the number of dimensions is $D = 1000$.

sification rate. The bottom row illustrates how renormalization overcomes the distortions induced by the variance inflation. The discriminant line is seen to separate the two classes appropriately.

To gain a better understanding of how the variance inflation and quality of the renormalization are effected by noise, we added Gaussian noise ($\mathcal{N}(0,\sigma_e^2)$) with $\sigma_e \in [0,5]$. For every noise level, 300 random training and test sets where drawn as explained above and kPCA was performed. Once again our goal was to classify digit 8 versus the rest by a linear classifier in the principal subspace. The results are summarized in Figure 10 where we show the error rate before and after renormalization as well as the result based on renormalizing according to the leave-one-out error. In the last case, the N projections determined from leave-one-out cross validation (LOOCV) are renormalized to follow the entire training set histogram. Renormalization is then only applied to the test set when this renormalized LOOCV error is less than the estimated baseline error. In the right panel of Figure 10 it is seen how renormalizing the projections leads to a much improved classifier as long as the SNR is 'reasonable'. Even when $\sigma_e = 0$ there is some inherent noise in the data, which explains why renormalization still improves the classification. As σ_e reaches 1 it is no longer possible to identify the digits by visual inspection, and classification becomes increasingly difficult.

The left panel of Figure 10 shows how the conventional error rate converges to the baseline of 0.1 (misclassifying all digits 8), for high noise levels. Basically, increasing the noise result in a more skewed test set subspace in relation to the subspace spanned by the training set (see Figure 1). At a given threshold this causes all the projections to lie on the same side of the discrimination function due to the imbalanced composition, leading to a misclassifications rate of $1/10$. As the idea of renormalization by histogram equalization is to restore the variation in the test set, this be-

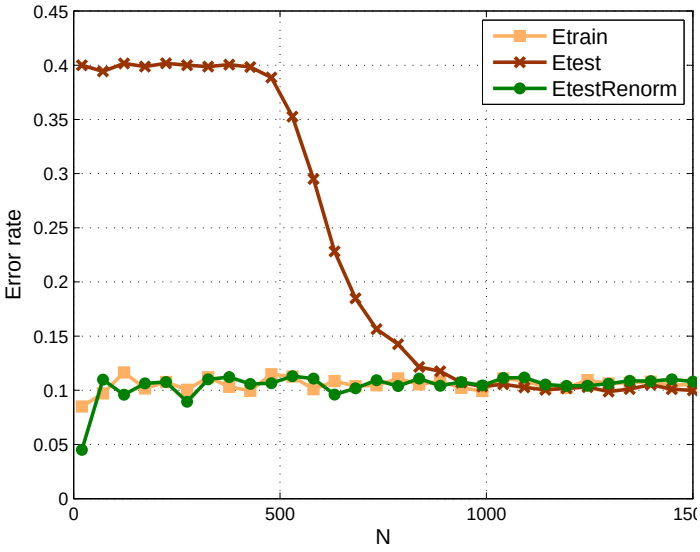

Figure 8: Classification error learning curves for the two semicircular clusters in i.i.d. noise setup. The signal to noise ratio was $SNR = 60$. The linear discriminant performs close to the optimal Bayes rate after the renormalization operation in all cases, while the conventional system suffers from poor generalizability, and requires about ten times as many examples to reach the same error level as the renormalized classifier. The experiment was carried out with $D = 2000$.

havior is naturally not encountered for the renormalized projections. Instead, as the SNR decreases, renormalization increases the error rate, as the test set observations are forced to be distributed on both sides of the discrimination line - which leads to many misclassifications when the signal is suppressed by the noise. However, using LOOCV based renormalization prevents the error rate from blowing up while at the same time improving the classification in the more sensible SNR regime as compared to conventional kPCA.

4.3 Functional MRI Data

As a second high dimensional real data example, functional magnetic resonance imaging (fMRI) data was used to illustrate the effect of renormalization. The fMRI data set was acquired by Dr. Egill Rostrup at Hvidovre Hospital on a 1.5 T Magnetom Vision MR scanner. The scanning sequence was a 2D gradient echo EPI (T2- weighted) with 66 ms echo time and 50° RF flip angle. The images were acquired with a matrix of $D = 128 \times 128 = 16,384$ pixels, with FOV of 230 mm, and 10 mm slice thickness, in a para-axial orientation parallel to the calcarine sulcus. The visual paradigm consisted of a rest period of 20 sec of darkness using a light fixation dot, followed by 10 sec of full-field checkerboard reversing at 8 Hz, and ending with 20 sec of rest (darkness). In total, 150 images were acquired in 50 sec, corresponding to a period of approximately 330 msec per image. The experiment was repeated in 10 separate runs containing 150 images each. In order to reduce saturation effects, the first 29 images were discarded, leaving 121 images for each run. We use a

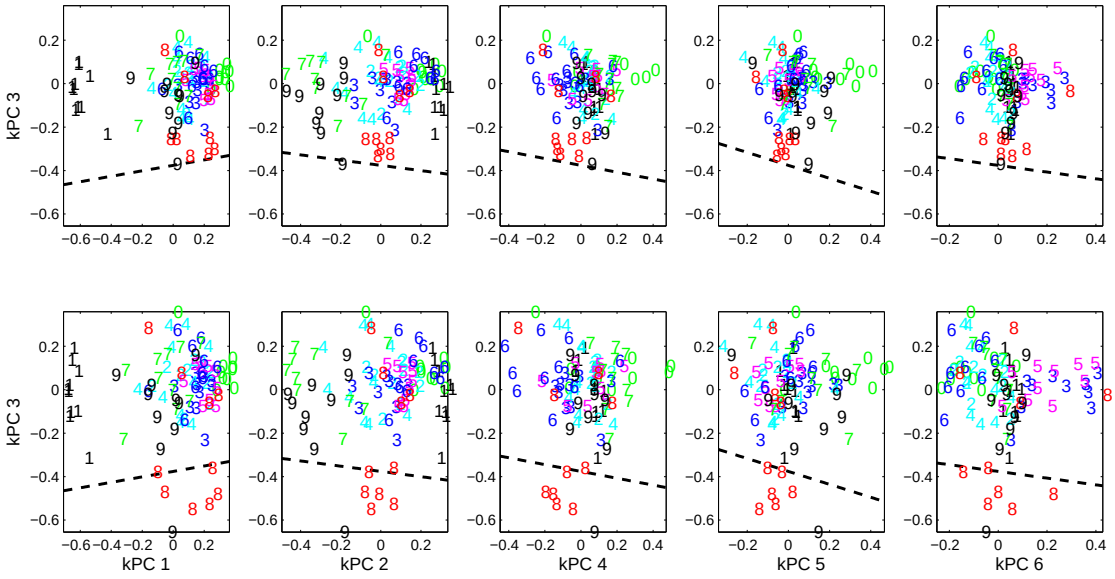

Figure 9: USPS handwritten digits test set projections. The top row shows the conventional projections, while the bottom row shows the projections after renormalization. In this example the third kPC carries a large part of the signal, and hence this component is shown versus the other five first PCs. The variance reduction and the consequent shift is evident from the top row. The dashed line indicates the linear discriminant function for classifying digit 8 vs the rest.

simple on-off activation reference function for supervision of the classifier. The reference function is off-set by 4 seconds to emulate the hemodynamic delay.

The data set is split in two equal sized subsets: Five runs for training and five runs for testing. As the test and training data are independent, the test error estimate is an unbiased estimator of performance. The scale of the Gaussian kernel was chosen as the 5th percentile of the mutual distances leading to $c \approx 15000$, while the dimension of the principal subspace is chosen as $q = 20$.

Again the principal components are tested for independence by a mutual information permutation test. Using 1000 permutations and a $\rho = 0.05$ significance level, we find that the null hypothesis is rejected for approximately 1% of the principal component pairs.

Similar to the handwritten digit data we perform linear classification in the kernel principal subspace. This was repeated 300 times using random splits for different noise levels. The results are summarized in Figure 11. Again renormalization is seen to decrease the error rate significantly, while the LOOCV based scheme furthermore prevents the increase in error rate for high noise levels (low SNR).

Figure 12 shows the projection of the data onto the first kPC's before and after renormalization.

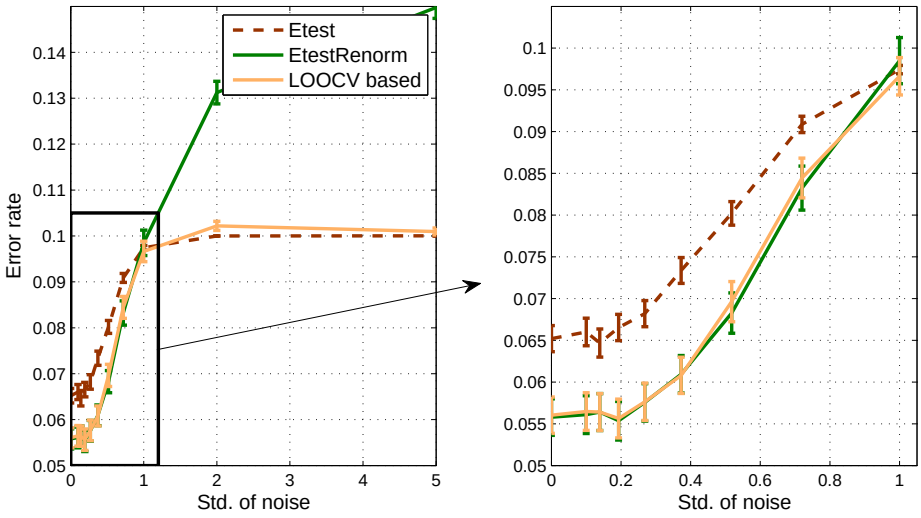

Figure 10: Mean error rates $\pm$ 1 standard deviation as a function of the noise level. The test error based on conventional kernel PCA projections, renormalized projections, and a LOOCV scheme is shown. Renormalization is seen to improve the performance, while LOOCV based renormalization prevents the classification error to blow up in the very low SNR regime.

5. Conclusion

Dimensionality reduction by PCA and kPCA can lack generalization due to training set variance inflation in the extremely ill-posed case when the sample size is much smaller than the input space dimension. In this work we have provided a simple geometric explanation for the main effect, namely that test points 'loose' their orthogonal projections, when their embedding is computed. This insight allowed for a speed-up of a previously proposed LOO scheme for renormalization. For kPCA we showed that the effects can be even more dramatic than in PCA, and we proposed a scheme for exact LOO renormalization of the embedding, and an approximate expression at lower cost. The viability of the new scheme was demonstrated for kPCA when used for dimensionality reduction both in simple synthetic data, in the USPS digit classification problem, and for fMRI brain state decoding.

Acknowledgments

We thank the reviewers of this manuscript and earlier versions for many useful comments. This research was supported by the Danish Lundbeckfonden through the Center for Integrated Molecular Brain Imaging (`www.cimbi.dk`).

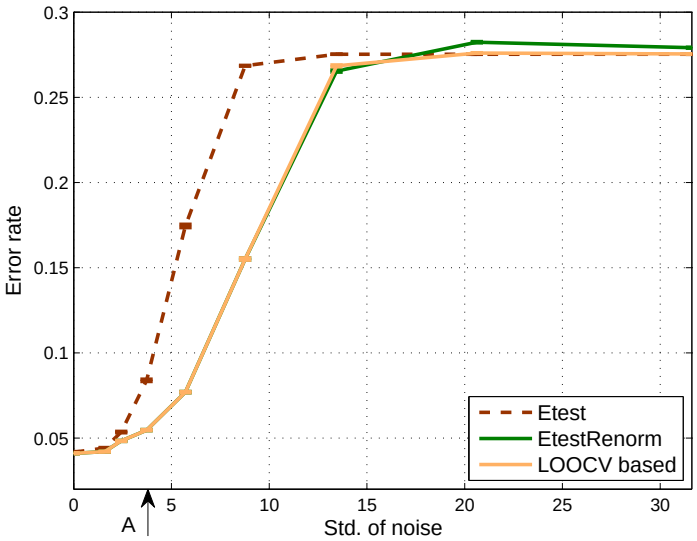

Figure 11: Mean error rates $\pm$ 1 standard deviation as a function of the noise level for fMRI data ($D = 16,384, N = 605$) . The test error based on conventional kernel PCA projections, renormalized projections, and a LOOCV scheme is shown. Renormalization is seen to clearly improve the performance. Arrow 'A' indicates the noise level used in Figure 12

Appendix A.

Let $\boldsymbol{u}_{N,k}$ be the k'th eigenvector of the covariance matrix on the full sample $\boldsymbol{\Sigma}_N$ and $\boldsymbol{u}_{N-1,k}$ be the corresponding eigenvector of LOO training set covariance matrix $\boldsymbol{\Sigma}_{N-1}$. In the following we use first order perturbation theory to show that

$$\boldsymbol{u}_{N-1,k}^T \cdot \boldsymbol{x}_N \approx \boldsymbol{u}_{N,k}^T \cdot \boldsymbol{x}_N^{\parallel} \ ,$$

where the data vector $\boldsymbol{x}$ has been split in its orthogonal and parallel components, $\boldsymbol{x}_N = \boldsymbol{x}_N^{\perp} + \boldsymbol{x}_N^{\parallel}$, relative to the subspace spanned by the training data. Thus, we are interested in the difference between $\boldsymbol{u}_{N,k}$ and $\boldsymbol{u}_{N-1,k}$. Simple manipulations of the covariance matrices lead to

$$\boldsymbol{\Sigma}_{N-1} = \boldsymbol{\Sigma}_N + \underbrace{\frac{1}{N-1}\boldsymbol{\Sigma}_N - \frac{1}{N}(\boldsymbol{x}_N - \boldsymbol{\mu}_{N-1})(\boldsymbol{x}_N - \boldsymbol{\mu}_{N-1})^T}_{O(\frac{1}{N})} \ .$$

By introducing the shorthand $\boldsymbol{A} = \boldsymbol{\Sigma}_{N-1}$ and $\boldsymbol{B} = \boldsymbol{\Sigma}_N$ we get

$$\boldsymbol{A} = \boldsymbol{B} + \delta \boldsymbol{C} \ , \tag{5}$$

where δ is of order $\frac{1}{N}$. Note that all matrices are symmetric. We now look at the k'th eigenvector of $\boldsymbol{A}$ and $\boldsymbol{B}$:

$$\boldsymbol{B}\boldsymbol{u}_k = \lambda_k \boldsymbol{u}_k \ , \tag{6}$$

$$\boldsymbol{A}\boldsymbol{v}_k = \nu_k \boldsymbol{v}_k \ . \tag{7}$$

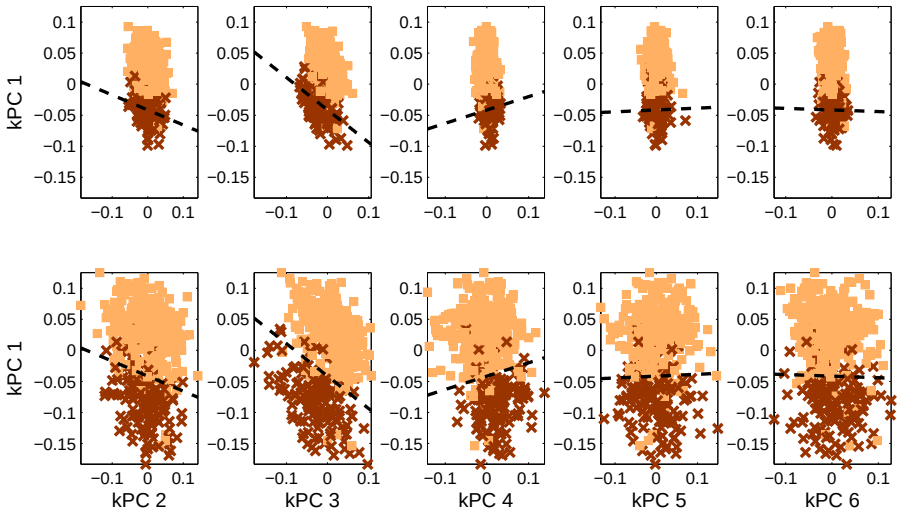

Figure 12: Test set projections of the fMRI data with Gaussian noise added as marked on Figure 11 ($\varepsilon_i = \mathcal{N}(0, 3.8^2)$). The top row shows the conventional projections, while the bottom row shows the projections after renormalization. The 'red class' indicates activation, while the blue observations are acquired during rest. The dashed line marks the linear discriminant. The scale is chosen as the 5th percentile of the mutual distances.

First order perturbation theory posits

$$\nu_k = \lambda_k + \delta\xi_k \,, \tag{8}$$
$$\boldsymbol{v}_k = \boldsymbol{u}_k + \delta\boldsymbol{w}_k \,. \tag{9}$$

That is, when going from N to $N-1$ samples we only have a small ($O(\frac{1}{N})$) change in eigenvalues and rotation of eigenvectors. Since all eigenvectors are orthonormal it follows that $\boldsymbol{u}_k \perp \boldsymbol{w}_k$, c.f.,

$$||\boldsymbol{v}_k||^2 = ||\boldsymbol{u}_k + \delta\boldsymbol{w}_k||^2 = \underbrace{||\boldsymbol{u}_k||^2}_{=1} + \underbrace{\delta^2}_{\approx 0}||\boldsymbol{w}_k||^2 + 2\delta\boldsymbol{u}_k^T\boldsymbol{w}_k = 1$$
$$\delta\boldsymbol{u}_k^T\boldsymbol{w}_k = 0 \,.$$

We now expand Equation (7) using Equation (5), (8) and (9)

$$\boldsymbol{A}\boldsymbol{v}_k = \nu_k\boldsymbol{v}_k \qquad \Rightarrow$$
$$(\boldsymbol{B} + \delta\boldsymbol{C})(\boldsymbol{u}_k + \delta\boldsymbol{w}_k) = (\lambda_k + \delta\xi_k)(\boldsymbol{u}_k + \delta\boldsymbol{w}_k) \,,$$

ignoring higher order terms of δ gives

$$\boldsymbol{B}\boldsymbol{u}_k + \delta\boldsymbol{C}\boldsymbol{u}_k + \delta\boldsymbol{B}\boldsymbol{w}_k = \lambda_k\boldsymbol{u}_k + \delta\lambda_k\boldsymbol{w}_k + \delta\xi_k\boldsymbol{u}_k \,,$$

Finally, exploiting Equation (6) reduces the above to

$$\boldsymbol{C}\boldsymbol{u}_k + \boldsymbol{B}\boldsymbol{w}_k = \lambda_k \boldsymbol{w}_k + \xi_k \boldsymbol{u}_k \ . \tag{10}$$

We now look for an estimate of ξ_k by left multiplying with $\boldsymbol{u}_k^T$

$$\boldsymbol{u}_k^T \boldsymbol{C}\boldsymbol{u}_k + \boldsymbol{u}_k^T \boldsymbol{B}\boldsymbol{w}_k = \lambda_k \boldsymbol{u}_k^T \boldsymbol{w}_k + \xi_k \boldsymbol{u}_k^T \boldsymbol{u}_k \ ,$$

using $||\boldsymbol{u}_k||^2 = 1$ and $\boldsymbol{u}_k \perp \boldsymbol{w}_k$ gives

$$\boldsymbol{u}_k^T \boldsymbol{C}\boldsymbol{u}_k + \boldsymbol{u}_k^T \boldsymbol{B}\boldsymbol{w}_k = \xi_k \ ,$$

since $\boldsymbol{B}$ is symmetric, $\boldsymbol{u}_k$ is both a left and right singular vector. Hence, $\boldsymbol{u}_k^T \boldsymbol{B}\boldsymbol{w}_k = \lambda_k \boldsymbol{u}_k^T \boldsymbol{w}_k$=0. Thus finally, it follows that

$$\boldsymbol{u}_k^T \boldsymbol{C}\boldsymbol{u}_k = \xi_k \ . \tag{11}$$

Next, we find an estimate of $\boldsymbol{w}_k$ by left multiplying Equation (10) with $\boldsymbol{u}_j^T \ j \neq k$.

$$\boldsymbol{u}_j^T \boldsymbol{C}\boldsymbol{u}_k + \boldsymbol{u}_j^T \boldsymbol{B}\boldsymbol{w}_k = \lambda_k \boldsymbol{u}_j^T \boldsymbol{w}_k + \xi_k \boldsymbol{u}_j^T \boldsymbol{u}_k \ ,$$

again we exploit the fact that $\boldsymbol{B}$ is symmetric and that $\boldsymbol{u}_j$ is orthogonal to $\boldsymbol{u}_k$, which gives

$$\boldsymbol{u}_j^T \boldsymbol{C}\boldsymbol{u}_k + \lambda_j \boldsymbol{u}_j^T \boldsymbol{w}_k = \lambda_k \boldsymbol{u}_j^T \boldsymbol{w}_k \ . \tag{12}$$

Assuming that span$\{\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_D\}$ = span$\{\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_D\}$, that is, the $\boldsymbol{v}$-basis is a rotation of the $\boldsymbol{u}$-basis, which implies that $\boldsymbol{w}_k$ can be represented as a linear combination of the $\boldsymbol{u}$-vectors (or $\boldsymbol{v}$-vectors), leads to

$$\boldsymbol{w}_k = \sum_{m=1}^{D} h_{km} \boldsymbol{u}_m \ .$$

Due to orthonormality of the eigenvectors, we now realize that $h_{kk} = 0$ and $\boldsymbol{u}_j^T \boldsymbol{w}_k = \boldsymbol{u}_j^T \sum_{m=1}^{D} h_{km} \boldsymbol{u}_m$ will only be non-zero for $m = j$. Hence, Equation (12) reduces to

$$\begin{aligned} \boldsymbol{u}_j^T \boldsymbol{C}\boldsymbol{u}_k + \lambda_j h_{kj} = \lambda_k h_{kj} \qquad & \Rightarrow \\ h_{kj} = \frac{\boldsymbol{u}_j^T \boldsymbol{C}\boldsymbol{u}_k}{\lambda_k - \lambda_j} \quad & k \neq j \\ h_{kk} = 0 \ . & \end{aligned}$$

In the above we have assumed a nondegenerate system, that is, $\lambda_k \neq \lambda_j \ \forall k \neq j$. Thus, $\boldsymbol{w}_k$ can be expressed as

$$\boldsymbol{w}_k = \sum_{m=1 \neq k}^{N} \frac{\boldsymbol{u}_m^T \boldsymbol{C}\boldsymbol{u}_k}{\lambda_k - \lambda_m} \boldsymbol{u}_m \ , \tag{13}$$

where we used that $\boldsymbol{C}\boldsymbol{u}_k$ is only non-zero for $k \leq N$. We are now ready to return to Equation (8) and (9) inserting the expressions derived for ξ_k and $\boldsymbol{w}_k$ in Equation (11) and (13) respectively:

$$\nu_k = \lambda_k + \delta \boldsymbol{u}_k^T \boldsymbol{C} \boldsymbol{u}_k \tag{14}$$

$$\boldsymbol{v}_k = \boldsymbol{u}_k + \delta \sum_{m=1 \neq k}^{N} \frac{(\boldsymbol{u}_{\boldsymbol{m}}^T(\boldsymbol{x}_N - \boldsymbol{\mu}_{N-1}))(\boldsymbol{u}_{\boldsymbol{k}}^T(\boldsymbol{x}_N - \boldsymbol{\mu}_{N-1}))}{\lambda_k - \lambda_m} \boldsymbol{u}_m \; . \tag{15}$$

Equation (14) shows that the change in eigenvalue is indeed small ($O(\frac{1}{N})$) when going from N to $N-1$ samples. For the eigenvector perturbation, Equation (15), we can bound the squared length of the sum and obtain a similar result,

$$\begin{aligned} \left|\left| \frac{1}{N} \sum_{m=1 \neq k}^{N} \frac{(\boldsymbol{u}_{\boldsymbol{m}}^T(\boldsymbol{x}_N - \boldsymbol{\mu}_{N-1}))(\boldsymbol{u}_{\boldsymbol{k}}^T(\boldsymbol{x}_N - \boldsymbol{\mu}_{N-1}))}{\lambda_k - \lambda_m} \boldsymbol{u}_m \right|\right|^2 \leq \\ \frac{1}{N^2} ||\boldsymbol{x}_N - \boldsymbol{\mu}_{N-1}||^2 \left|\left| \sum_{m=1 \neq k}^{N} \frac{(\boldsymbol{u}_{\boldsymbol{m}}^T(\boldsymbol{x}_N - \boldsymbol{\mu}_{N-1}))}{\lambda_k - \lambda_m} \boldsymbol{u}_m \right|\right| = \\ \frac{1}{N^2} ||\boldsymbol{x}_N - \boldsymbol{\mu}_{N-1}||^2 \sum_{m=1 \neq k}^{N} \frac{|(\boldsymbol{u}_{\boldsymbol{m}}^T(\boldsymbol{x}_N - \boldsymbol{\mu}_{N-1}))|^2}{|\lambda_k - \lambda_m|^2} \leq \\ \frac{1}{N^2} \frac{2||\boldsymbol{x}_N - \boldsymbol{\mu}_{N-1}||^4}{|\Delta\lambda_k|^2} \; , \end{aligned}$$

where $\Delta\lambda_k$ is the spacing between the k'th eigenvalue and the closest neighbor, and the factor of two compensates for the missing k'th term in the sum, that is, the perturbation is of order $O(1/N)$

References

Michael Biehl and Andreas Mietzner. Statistical mechanics of unsupervised structure recognition. *Journal of Physics A-Mathematical and General*, 27(6):1885–1897, 1994.

Gilles Blanchard, Olivier Bousquet, and Laurent Zwald. Statistical properties of kernel principal component analysis. *Machine Learning*, 66(2-3):259–294, 2007.

Mikio L. Braun, Joachim M. Buhmann, and Klaus-Robert Müller. On relevant dimensions in kernel feature spaces. *Journal of Machine Learning Research*, 9:1875–1908, 2008.

Rafael C. Gonzalez and Paul Wintz. *Digital Image Processing*. 1977. ISBN 0-201-02596-5 (hardcover), 0-201-02597-3 (paperback).

David C. Hoyle and Magnus Rattray. A statistical mechanics analysis of gram matrix eigenvalue spectra. In *Lecture Notes in Computer Science, 17th Annual Conference on Learning Theory*, volume 3120, pages 579–593. Springer Verlag, 2004a.

David C. Hoyle and Magnus Rattray. Limiting form of the sample covariance eigenspectrum in pca and kernel pca. In *Advances in Neural Information Processing Systems 16*, pages 16–23. MIT Press, 2004b.

David C. Hoyle and Magnus Rattray. Principal-component-analysis eigenvalue spectra from data with symmetry-breaking structure. *Physical Review E*, 69(2):026124, 2004c.

David C. Hoyle and Magnus Rattray. Statistical mechanics of learning multiple orthogonal signals: Asymptotic theory and fluctuation effects. *Physical Review E (Statistical, Nonlinear, and Soft Matter Physics)*, 75(1):016101, 2007.

Jonathan J . Hull. A database for handwritten text recognition research. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(5):550–554, 1994.

Robert Jenssen, Torbjörn Eltoft, Deniz Erdogmus, and Jose C. Principe. Some equivalences between kernel methods and information theoretic methods. *Journal of VLSI Signal Processing*, 45:49–65, 2006.

Iain M. Johnstone. On the distribution of the largest eigenvalue in principal components analysis. *Annals of Statistics*, 29(2):295–327, 2001.

Ulrik Kjems, Lars K. Hansen, and Stephen C. Strother. Generalizable singular value decomposition for ill-posed datasets. In *Advances in Neural Information Processing Systems 13*, pages 549–555. MIT Press, 2001.

Rudy Moddemeijer. On estimation of entropy and mutual information of continuous distributions. *Signal Processing*, 16(3):233–246, 1989.

Sofia Mosci, Lorenzo Rosasco, and Alessandro Verri. Dimensionality reduction and generalization. In *Proceedings of the 24th International Conference on Machine Learning*, pages 657–664, 2007.

Peter Reimann, Chris Van den Broeck, and Geert J. Bex. A Gaussian scenario for unsupervised learning. *Journal of Physics A - Mathematical and General*, 29(13):3521–3535, 1996.

Bernhard Schölkopf, Alex Smola, and Klaus-Robert Müller. Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation*, 10(5):1299–1319, 1998.

John Shawe-Taylor and Christopher K. I. Williams. The stability of kernel principal components analysis and its relation to the process eigenspectrum. In *Advances in Neural Information Processing Systems 15*, pages 367–374. MIT Press, 2003.

Jack W. Silverstein and Patrick L. Combettes. Signal-detection via spectral theory of large dimensional random matrices. *IEEE Transactions on Signal Processing*, 40(8):2100–2105, 1992.

Laurent Zwald and Gilles Blanchard. On the convergence of eigenspaces in kernel principal component analysis. In *Advances in Neural Information Processing Systems 18*, pages 1649–1656. MIT Press, 2006.