

# KERNEL PCA WITH THE NYSTRÖM METHOD

FREDRIK HALLGREN

*Department of Statistical Science  
University College London*

**ABSTRACT.** The Nyström method is one of the most popular techniques for improving the scalability of kernel methods. However, it has not yet been derived for kernel PCA in line with classical PCA. In this paper we derive kernel PCA with the Nyström method, thereby providing one of the few available options to make kernel PCA scalable. We further study its statistical accuracy through a finite-sample confidence bound on the empirical reconstruction error compared to the full method. The behaviours of the method and bound are illustrated through computer experiments on multiple real-world datasets. As an application of the method we present kernel principal component regression with the Nyström method, as an alternative to Nyström kernel ridge regression for efficient regularized regression with kernels.

**Keywords:** Kernel methods, non-parametric statistics, confidence interval, dimensionality reduction, unsupervised learning, learning theory, PCA, functional PCA, PCR, MDS

## CONTENTS

1. Introduction	2
2. Previous work	5
3. Background	8
4. Kernel PCA with the Nyström method	12
5. Prelude: A special case	16
6. Statistical accuracy of Nyström kernel PCA	17
7. Experimental analysis	20
8. Application: Nyström principal component regression	27
9. Conclusion	30
Appendix A. Proofs	32
References	40

---

*E-mail address:* fredrik.hallgren@ucl.ac.uk.

*Acknowledgements.* The author is hugely grateful to John Shawe-Taylor, Dino Sejdinovic, Paul Northrop and Michael Arbel for invaluable comments and feedback.

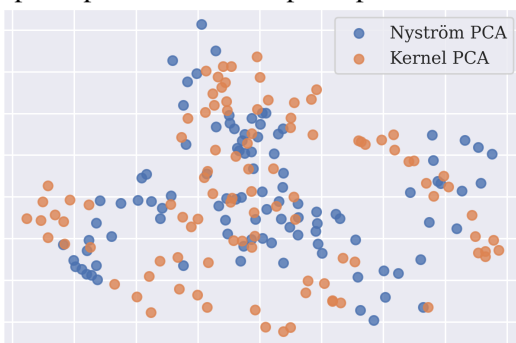
## 1. INTRODUCTION

Kernel methods generalize classical statistical methods to discover non-linear patterns in data [Hofmann et al., 2008]. They have been demonstrated to achieve excellent performance in many application domains and it is straightforward to apply them to non-numeric data, such as graphs or text [Vishwanathan et al., 2010, Lodhi et al., 2002]. Through a near arbitrary non-linear mapping of data points into a Hilbert space they offer remarkable flexibility whilst providing a precise mathematical framework for statistical analyses. A host of linear statistical methods have been adapted to be used with kernels, including Fisher discriminant analysis (FDA) [Mika et al., 1999], independent component analysis (ICA) [Bach and Jordan, 2002], instrumental variable (IV) regression [Singh et al., 2019], and many more. Kernel PCA is a non-linear version of principal component analysis (PCA), a ubiquitous method to discover the most important directions of variation in data [Pearson, 1901]. PCA may be used for dimensionality reduction, exploratory data analysis, anomaly detection, discriminant analysis, clustering, or as a general preprocessing step for regression or classification [Jolliffe, 2002, Wold et al., 1987].

The other side of the coin of kernel methods is their large computational requirements, as they generally scale in the number of data points rather than the number of data dimensions. As a remedy, various approximations have been proposed, such as the Nyström method, which randomly selects a smaller subset of data points and looks for solutions in their linear span. The Nyström method also plays an important role in recent state-of-the-art implementations of kernel methods [Rudi et al., 2017, Ma and Belkin, 2017, Meanti et al., 2020, Carratino et al., 2021, Frangella et al., 2021].

The need for approximate methods becomes particularly acute for kernel PCA, since it relies on the eigendecomposition of the kernel matrix, which requires about  $9n^3 + \mathcal{O}(n^2)$  floating-point operations, as opposed to  $\frac{1}{3}n^3 + \mathcal{O}(n^2)$  floating-point operations for the solution of a linear system by way of the Cholesky decomposition when performing regression [Golub and Van Loan, 2013, Chapters 4, 8]. Despite this fact, kernel PCA with the Nyström method has not yet been derived in line with linear PCA, significant previous research interest notwithstanding.

In this paper we derive kernel PCA with the Nyström method, providing orthonormal principal components in the span of the Nyström subset that maximize the variance of the data, without assuming that the data has zero mean, as well as the associated principal scores<sup>1</sup>. The principal scores are perhaps of particular interest, since they allow for the method to be used as a preprocessing step before applying supervised learning methods, by virtue of providing a new representation of data points in the new coordinate system defined by the principal components. The figure to the right shows the first two dimensions for these representations for an example with a dataset of images of handwritten digits, in comparison with standard full kernel PCA, where the Nyström subset uses only a tenth of the full dataset<sup>2</sup>.



The principal scores are given as follows. First let  $K_{mm}$  be  $m$  randomly subsampled rows and columns of the original kernel matrix  $K$ , and  $K_{nm}$  be the same  $m$  subsampled columns. Centring

<sup>1</sup>Different conventions exist for the terminology of PCA. Throughout this paper we will take the *principal components* to mean the vectors defining the subspaces that maximize the variance of the data i.e. the eigenvectors of the centred covariance operator or matrix. These are elsewhere sometimes referred to as the *principal axes*.

<sup>2</sup>Please see <https://github.com/fredhallgren/nystrompca> for details

the data in feature space corresponds to adjusting these matrices through

$$\begin{aligned} K'_{nm} &= K_{nm} - \mathbb{1}_n K_{nm} - \tilde{K} \mathbb{1}_n^{n,m} + \mathbb{1}_n \tilde{K} \mathbb{1}_n^{n,m} \\ K'_{mm} &= K_{mm} - \mathbb{1}_n^{m,n} K_{nm} - K_{mn} \mathbb{1}_n^{n,m} + \mathbb{1}_n^{m,n} \tilde{K} \mathbb{1}_n^{n,m} \end{aligned}$$

where  $\tilde{K} = K_{nm} K_{mm}^{-1} K_{mn}$  and  $\mathbb{1}_n$ ,  $\mathbb{1}_n^{n,m}$  and  $\mathbb{1}_n^{m,n}$  are  $n \times n$ ,  $n \times m$  and  $m \times n$  matrices respectively with all elements equal to  $\frac{1}{n}$ . Now create an approximate kernel matrix through

$$\tilde{K}' = \frac{1}{n} K_{mm}'^{-1/2} K_{mn}' K_{nm}' K_{mm}'^{-1/2}$$

and calculate its eigenvalues  $\tilde{\lambda}_j$  and eigendecomposition  $V \tilde{\Lambda} V^T$ , where  $M^{-1/2} = U D^{-1/2} U^T$  for a matrix  $M$  with eigendecomposition  $U D U^T$ . The scores are then given by  $W = K_{nm}' K_{mm}'^{-1/2} V$ , which is a new data matrix with observations along the rows, and the variances of the new data variables (in the columns) are given by  $\tilde{\lambda}_j$ . The method has time complexity  $\mathcal{O}(nm^2)$  which is the same as when the Nyström method is applied to regression. Full details may be found in Section 4 on page 12.

The method centres the data in the feature space, as is the case for linear PCA as well as the original presentation of kernel PCA [Pearson, 1901, Jolliffe and Cadima, 2016, Schölkopf et al., 1998]. Without this adjustment, the perpendicular lines defined by the principal components are forced to go through the origin, no longer minimizing the reconstruction error in an unconstrained fashion and requiring an assumption of zero-mean data in feature space. If the assumption does not hold then the results may be wildly different from the true values. The assumption may be especially contrived for kernel methods, since for any positive kernel function, like the popular radial basis functions or Cauchy kernels, it will *never hold*, since the corresponding feature maps  $\phi(x_i) = k(x_i, \cdot)$  in the reproducing kernel Hilbert space are positive everywhere. Subtracting the mean from the input data will not give zero-mean data in feature space. Sometimes the first eigenvalue of the uncentred kernel matrix will account for the mean being different from zero, and the subsequent eigenvalues approximately correspond to the full set of eigenvalues of the centred kernel matrix, but this is not always the case and the correspondence is not exact.

We further study the statistical accuracy of the proposed method. In the special case when the number of subsampled data points for the Nyström method equals the PCA dimension, then both the empirical and true reconstruction errors of the Nyström method equal the corresponding reconstruction errors for kernel PCA constructed directly using only the subset of data points. For the general case we provide a finite-sample confidence bound (a confidence interval) with  $\mathcal{O}(m^3)$  time complexity that doesn't require that we have observed the entire dataset, only the subset of data [Ramachandran and Tsokos, 2015]. In line with essentially all existing results on the accuracy of standard kernel PCA we here assume that data has zero mean. The result states that with high confidence, the difference between the empirical reconstruction errors of Nyström kernel PCA and full kernel PCA is less than or equal to a data-dependent quantity

$$R_n(\tilde{V}_d) - R_n(\hat{V}_d) \leq h \left( \sup_x k(x, x), \left\{ \hat{\lambda}_j \right\}_{j=1}^{d+1}, m, n \right)$$

which depends on the maximum value of the kernel function, the eigenvalues of the kernel matrix from the subset of randomly subsampled data points, the size of this subset  $m$  and the total size

of the dataset  $n$ , where  $h(\cdot)$  is a fixed function. All quantities in the bound are known or can be calculated from the data at hand. Please see Section 6 on page 17 for the complete result.

We illustrate and evaluate the proposed method and derived confidence bound through experimental analysis using several different datasets and kernel functions. We first compare the accuracy of Nyström kernel PCA with a number of other unsupervised learning methods, where its performance is seen to be very close to full kernel PCA, whilst being much more efficient. Then we illustrate the behaviour of the bound across different PCA dimensions. The source code for all the experiments is publicly available at <https://github.com/fredhallgren/nystrompca>.

From the proof of the confidence bound one can deduce sharper versions of several previous concentration results based on the covariance operator, including from Rosasco et al. [2010], De Vito et al. [2005b] and Blanchard and Zadorozhnyi [2019]. Please see Section 6.1 on page 19 for details.

To demonstrate the use of Nyström kernel PCA with supervised learning methods we apply it to the regression problem to present kernel principal component regression with the Nyström method. Principal component regression (PCR) performs a linear regression on the principal scores from the top principal components instead of the original data and introduces regularization for improved generalization. We also illustrate the method through experimental analysis and compare it to kernel ridge regression with the Nyström method. In summary, the prediction for a data point  $x^*$  is given by

$$\hat{y} = \bar{y} + y'^T K'_{nm} K'^{-1/2}_m V_d \tilde{\Lambda}_d^{-1} V_d^T K'^{-1/2}_m \tilde{\kappa}(x^*)$$

where  $y' = (y_1 - \bar{y}, y_2 - \bar{y}, \dots, y_n - \bar{y})^T$  and  $\tilde{\kappa}(x) = \kappa_m(x) - K_{mn} \mathbf{1}_n - \mathbf{1}_n^{m,n} K_{nm} K_{mm}^{-1} \kappa_m(x) + \mathbf{1}_n^{m,n} K \mathbf{1}_n$  with  $\mathbf{1}_n$  a length- $n$  column vector given by  $\mathbf{1}_n = (\frac{1}{n}, \frac{1}{n}, \dots, \frac{1}{n})^T$  and  $\kappa_m(x) = (k(x_1, x), k(x_2, x), \dots, k(x_m, x))^T$ . Using similar techniques we also present a novel derivation of standard kernel PCR with centred data in feature space, where a prediction is given by

$$\hat{y} = \bar{y} + y'^T Q_d \Lambda_d^{-1} Q_d^T \kappa'(x^*)$$

where  $Q_d \Lambda_d Q_d^T$  is the truncated eigendecomposition of  $K' = K - \mathbf{1}_n K - K \mathbf{1}_n + \mathbf{1}_n K \mathbf{1}_n$  and  $\kappa'(x) = \kappa(x) - \mathbf{1}_n \kappa(x) - K \mathbf{1}_n + \mathbf{1}_n K \mathbf{1}_n$  with  $\kappa(x) = (k(x_1, x), k(x_2, x), \dots, k(x_n, x))^T$ . We refer to Section 8 on page 27 for the full derivation and experimental results.

A summary of our main contributions is as follows

- (1) Deriving kernel PCA with the Nyström method
- (2) A result on the accuracy in the special case of  $d = m$  for both the empirical and true errors
- (3) A finite-sample confidence bound for the empirical error in the general case
- (4) Presenting kernel principal component regression with the Nyström method
- (5) Novel specification of kernel PCR with centred regressors
- (6) Sharper versions of some concentration results from previous literature

In the next section we give an overview of previous work (Section 2), then go through relevant background (Section 3), present the main method (Section 4), study the special case when  $d = m$  (Section 5), provide the confidence bound on the accuracy of the method (Section 6), conduct

experimental analysis of the method and bound (Section 7), present kernel principal component regression with the Nyström method (Section 8) and finally conclude with a summary and outlook (Section 9). Proofs are in the appendix.

**Notation.** Upper-case letters will be used for matrices and operators and generally for random variables, unless they represent data points before they are observed. Vectors in  $\mathbb{R}^p$  will be denoted by small letters and parameters fitted to data often by letters from the Greek alphabet. A row vector  $v$  in  $\mathbb{R}^p$  with elements  $v_1, v_2, \dots, v_p$  will be written  $(v_1, v_2, \dots, v_p)$  and its  $i$ th element will also be written  $(v)_i$ . The transpose of a vector or matrix is  $v^T$ . If not stated otherwise all Euclidean vectors will be column vectors. The arithmetic mean of a vector is denoted  $\bar{v}$ . Indices for data points will be denoted by  $i, r$ , or  $\ell$ ; indices for eigenvectors or dimensions will be denoted by  $j, k, p$  or  $q$ . Estimated quantities will often be denoted by  $\hat{\cdot}$ , approximations by  $\tilde{\cdot}$  and centred quantities by  $\cdot'$ . Empirical quantities may be superscripted or subscripted by the number of observations used in the estimate. The probability density function of a measure  $\mathbb{P}_Y$  will be denoted by  $p_Y(y)$ . The symbol  $Y$  will be used for a generic random variable; the symbols  $T$  or  $L$  for a generic operator and the symbol  $M$  for a generic matrix. The linear span of a set of vectors  $A$  is written  $\text{span}\{A\}$  or  $\langle A \rangle$ . The cardinality of a basis for the space  $V$  is written  $\dim(V)$ . The real part of a complex number is denoted  $\text{Re}\{x + iy\} = x$ .

The symbol  $\mathcal{O}(\cdot)$  denotes Big-O notation [Sipser, 2013]. The function  $\lambda_j(\cdot)$  returns the  $j$ th eigenvalue, in decreasing order, of its argument, and the symbol  $\lambda_{<d}$  denotes the sum of the largest  $d$  eigenvalues  $\lambda_1, \lambda_2, \dots, \lambda_d$ . If  $v$  is majorized by  $u$  we write  $v \succ u$ . The symbol  $:=$  denotes the introduction of new notation, i.e.  $a := b$  means that  $b$  will be denoted by  $a$ , and vice versa for  $a =: b$ . The binary operators  $\vee$  and  $\wedge$  are defined as  $a \vee b = \max\{a, b\}$  and  $a \wedge b = \min\{a, b\}$ .

The functional  $\|\cdot\|$  denotes the operator norm or the Euclidean norm, depending on the context. For other norms the space will always be specified. For an operator  $T$ , we let  $T^*$  denote its adjoint. The image of an operator is  $\text{Im}(T)$  and its null space (also called its kernel) is  $\text{Ker}(T)$ .

## 2. PREVIOUS WORK

The study of the statistical accuracy of kernel PCA, or of the related problems of functional PCA [Besse and Ramsay, 1986, Hall et al., 2006] and PCA of a Hilbert space-valued random variable [Besse, 1991], was initiated in Dauxois et al. [1982]. They demonstrated the consistency of the reconstruction error and asymptotic normality of the empirical reconstruction error and principal components about the true quantities. The asymptotics of kernel PCA was also studied in Koltchinskii and Giné [2000]. A concentration inequality for the empirical reconstruction error versus its expectation, using McDiarmid's inequality [McDiarmid, 1989], was provided in Shawe-Taylor et al. [2002] and the same authors later presented a confidence bound on the expected empirical reconstruction error versus the true error [Shawe-Taylor et al., 2005], which is based on Rademacher complexities [Bartlett and Mendelson, 2002]. In this bound the expectation is with respect to the data point to be projected and the confidence with respect to different training datasets. A similar bound, as well as a version for centred kernel PCA, was later presented in Blanchard et al. [2007]. Approximate confidence bounds for both the principal values and components were given in Hall and Hosseini-Nasab [2006] based on the bootstrap method [Davison and Hinkley, 1997]. However, these results are not immediately applicable to kernel PCA since the kernel is defined on a compact subset of  $\mathbb{R} \times \mathbb{R}$ . Error bounds for the principal components were also provided in [Zwald and Blanchard, 2005]. The first PAC-Bayes bounds for kernel PCA were recently given in Haddouche et al. [2020].

The statistical accuracy of kernel PCA has been widely studied under the assumption of zero-mean data in feature space. To the best of our knowledge, only one bound has been presented in previous literature for centred kernel PCA [Blanchard et al., 2007], but this bound is more conservative than the available bounds for uncentred kernel PCA. This is because the bound is based on a bound for uncentred kernel PCA, with an additional term to account for the error introduced by mean-adjustment. However, uncentred kernel PCA minimizes the reconstruction error under the constraint that the subspaces maximizing the variance go through the origin and so the reconstruction error will be much larger. Consequently there is strictly no bound available for standard kernel PCA without an assumption of zero-mean data in feature space, that is as accurate as the existing bounds for uncentred kernel PCA.

The Nyström method has been widely studied for different settings and assumptions. Originally developed for the discretization of integral equations [Nyström, 1930, Banach, 1932], it was adapted to kernel methods in Williams and Seeger [2001] and applied to regression. The accuracy of the approximate kernel matrix versus the full kernel matrix, considering the full dataset as fixed, has been studied in a number of papers, please see Gittens and Mahoney [2016] and references therein. The study of the accuracy of the Nyström method as applied to regression culminated in the seminal work by Rudi et al. [2015] as a bound on the expected regression error with general assumptions.

A recent paper [Sterge et al., 2020] applied the Nyström method to kernel PCA, but only explicitly derived the first principal component and also assumed data to have zero mean in feature space. In the current work we derive all the principal components without assuming that data has zero mean, and in addition we derive the principal scores and provide an empirical evaluation of our method. They also presented a probabilistic inequality for the true reconstruction error with respect to the empirical subspace, which depends on the maximum value of the kernel function  $\sup_x k(x, x)$ , the total number of data points  $n$  and the covariance operator  $C$  from the unknown true distribution  $\mathbb{P}$ . The main difference to our confidence bound is that we bound the empirical reconstruction error (the estimate) and only in terms of known quantities. As a corollary they also presented an asymptotic rate of convergence under the assumption of polynomial decay of the eigenvalues of  $C$ .

Even more recently Sterge and Sriperumbudur [2021] presented a similar analysis to Sterge et al. [2020], but with one way of centring the data. This method is different from the one considered here, with the top principal component given by

$$\tilde{\phi}_1 = \sqrt{m} G_m^* K_{mm}^{-1/2} u_1$$

where  $u_1$  is the top eigenvector of  $\frac{1}{n-1} K_{mm}^{-1/2} (K_{mn} - K_{mn} \mathbf{1}_n) K_{nm} K_{mm}^{-1/2}$  and  $G_m^*$  is given by  $\alpha \mapsto \frac{1}{\sqrt{m}} \sum_{k=1}^m \alpha_k k(x_k, x)$ . This method does not minimize the reconstruction error or maximize the variance, and it does not recover standard kernel PCA when  $m = n$ .

If we assume data to have zero mean in feature space, then our derived principal scores are somewhat similar to the *virtual samples* of Golts and Elad [2016] which were introduced in the context of dictionary learning [Aharon et al., 2006]. These are obtained through the projection of the full dataset onto the Nyström subset and may also be used as a drop-in replacement for the original data points.

Another related method was described in Iosifidis and Gabbouj [2016]. They derive low-rank data representations from the Nyström kernel matrix, similar to the *virtual samples* above, including for novel unseen data points. However, the representations for novel data points require calculation of the top eigenvalues of the full kernel matrix, and so is  $\mathcal{O}(n^3)$  in time and leads to no improvement in computational efficiency compared to standard kernel PCA. They also propose a method to centre

the data in feature space, although this is done in order to make  $\mathbb{R}^m$  into a subspace of  $\mathcal{H}$  and the centred representations are different from the principal scores derived below. The centring of the matrices  $K_{nm}$  and  $K_{mm}$  is the same as the one used here, but these matrices are applied differently.

The above two methods represent the data points directly through their projections onto the truncated eigenspace of the Nyström subset and do not attempt to find the representations in the span of this subset that best describe the data, unlike the method presented in the current paper, which calculates the representations that have maximum variance for a given number of retained dimensions.

In Gisbrecht and Schleif [2015] the Nyström method was studied in the context of distance matrices and MDS, which is a related area to kernel PCA (also see De Silva and Tenenbaum [2004], Platt [2005]). They calculated the eigenpairs of  $\tilde{K} = K_{nm}K_{mm}^{-1}K_{mn}$ , from which the principal components, scores and values could be derived, although these specific quantities were not provided. They also applied the *double centering* procedure, which is used in MDS to convert a distance matrix into a matrix of inner products. When applied to the Nyström kernel matrix  $\tilde{K}$  this procedure becomes similar to the centring used in the current paper, although it is not quite the same and it is applied for different purposes.

In the table below we summarize previous applications of the Nyström method to kernel PCA in terms of important properties when applying the Nyström method or developing variations of PCA.

	Maximizes variance	Without zero-mean assumption	Full set of principal components	Principal scores	Scores for new data	Recovers kernel PCA when $n = m$	Experimental evaluation
This paper	✓	✓	✓	✓	✓	✓	✓
Sterge et al (2021)	✗	✓	✗	✗	✗	✗	✗
Sterge et al (2020)	✓	✗	✗	✗	✗	✓	✗
Golts and Elad (2016)	✗	✗	✗	✓	✓	✓	✓
Iosifidis and Gabbouj (2016)	✗	✓	✗	✓	✗	✓	✓

TABLE 1. Comparison with previous methods of desirable properties of the application of the Nyström method to kernel PCA

There are few other methods suggested in the literature to make kernel PCA scalable. Stochastic optimization was applied to kernel PCA in Zhang et al. [2016], but under the assumption of zero-mean data in feature space. The zero-mean assumption was also employed in Balcan et al. [2016], who presented a distributed algorithm for kernel PCA. Random features have also been applied to kernel PCA, corresponding to applying linear PCA to the approximate covariance matrix obtained from the random features, which allows for data to have non-zero mean, but assumes a shift-invariant kernel function [Sriperumbudur and Sterge, 2017]. Consequently, to the best of our knowledge, other than the Nyström method there is no other procedure proposed in previous literature to make kernel PCA scalable, for arbitrary kernel functions or without the restrictive assumption of zero-mean data.

### 3. BACKGROUND

We have a reproducing kernel Hilbert space  $\mathcal{H}$  (RKHS) of functions from a set  $\mathcal{X}$  to the real numbers. Associated with each RKHS is a symmetric positive definite kernel  $k : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$  with a reproducing property  $\langle k(x, \cdot), f \rangle_{\mathcal{H}} = f(x)$  for which the point evaluation  $f \mapsto \langle k(x, \cdot), f \rangle_{\mathcal{H}}$  is bounded. The kernel maps each element  $x \in \mathcal{X}$  to an element  $\phi(x) := k(x, \cdot) \in \mathcal{H}$ . We assume throughout that  $\mathcal{H}$  is separable, which will be the case for example if  $k$  is continuous and  $\mathcal{X}$  is compact [Paulsen and Raghupathi, 2016].

We have observations  $\{x_i\}_{i=1}^n$  of an  $\mathcal{X}$ -valued random variable  $X : (\Omega, \mathcal{A}, \mathbb{P}) \rightarrow (\mathcal{X}, \mathcal{A}_{\mathcal{X}}, \mathbb{P}_X)$  where  $\mathbb{P}_X(A) = \mathbb{P}(X^{-1}(A))$  [Cohn, 1980, Graham and Talay, 2011]. We obtain a random variable  $Z = \phi(X) \in \mathcal{H}$  with observations  $z_i = \phi(x_i)$ , assuming that  $\phi$  is measurable, which is the case for example when  $k$  is continuous. We assume  $Z$  is absolutely continuous and that it has a continuous density and so all  $z_i$  will be distinct. Its expectation in  $\mathcal{H}$  is given by  $\mathbb{E}[Z] = \int Z d\mathbb{P}$  in the sense of Bochner. We assume  $Z$  is in  $L^2(\Omega, \mathcal{A}, \mathbb{P}; \mathcal{H})$  with norm  $(\mathbb{E}[\|Z\|_{\mathcal{H}}^2])^{1/2} = (\int \|Z\|_{\mathcal{H}}^2 d\mathbb{P})^{1/2}$  [Ledoux and Talagrand, 2013].

Principal component analysis (PCA) of the zero-mean random variable  $Z \in \mathcal{H}$  constructs an optimal subspace  $V_d \subset \mathcal{H}$ , of dimension  $d$ , such that the so-called reconstruction error

$$R(V) = \mathbb{E} [\|P_V Z - Z\|_{\mathcal{H}}^2]$$

is minimized, where  $P_V : \mathcal{H} \rightarrow \mathcal{H}$  is the projection of (a realization of)  $Z$  on a subspace  $V$  [Besse, 1991]. This is termed the *true* reconstruction error [Blanchard et al., 2007]. Since  $Z$  is square-integrable the reconstruction error always exists and is finite.

In other words, the optimal  $d$ -dimensional subspace  $V_d$  is given by

$$V_d = \arg \min_{\dim(V)=d} \mathbb{E} [\|P_V Z - Z\|_{\mathcal{H}}^2]$$

An estimate of the optimal subspace  $V_d$  is obtained from the data  $\{z_i\}_{i=1}^n$  by minimizing the *empirical* reconstruction error

$$R_n(V) = \frac{1}{n} \sum_{i=1}^n \|P_V z_i - z_i\|_{\mathcal{H}}^2$$

which has a unique minimum since all eigenvalues are distinct [Blanchard et al., 2007]. We denote the estimated subspace by  $\hat{V}_d$ . One may also consider the true reconstruction error with respect to the empirical subspace, given by

$$R(\hat{V}_d) = \mathbb{E} [\|P_{\hat{V}_d} Z - Z\|_{\mathcal{H}}^2]$$

where the expectation may be taken both with respect to  $Z$  and  $\hat{V}_d$ , or treating the subspace as fixed; as well as the expected value of the empirical reconstruction error, given by



$$\mathbb{E} \left[ R_n(\hat{V}_d) \right] = \mathbb{E} \left[ \frac{1}{n} \sum_{i=1}^n \|P_{\hat{V}_d} z_i - z_i\|_{\mathcal{H}}^2 \right]$$

When the random variable  $Z$  does not have zero mean, the smallest reconstruction error is obtained from the centred random variable  $Z' = Z - \mathbb{E}[Z]$

$$R(V_d) = \min_{\dim(V)=d} \mathbb{E}[\|P_V Z' - Z'\|_{\mathcal{H}}^2]$$

and similarly for the empirical reconstruction error replacing  $z_i$  by  $z'_i = z_i - \frac{1}{n} \sum_{\ell=1}^n z_\ell$ .

Alternatively, instead of minimizing the reconstruction error of the centred random variable over  $d$ -dimensional subspaces  $V$ , one may minimize over affine subspaces with respect to the original random variable, and also optimize with respect to the term used for centring

$$R(V_d) = \min_{\substack{a \in \mathcal{H} \\ \dim(V)=d}} \mathbb{E}[\|P_{a+V} Z - Z\|_{\mathcal{H}}^2] = \min_{\substack{a \in \mathcal{H} \\ \dim(V)=d}} \mathbb{E}[\|P_V(Z - a) - (Z - a)\|_{\mathcal{H}}^2]$$

where  $a$  is the translation of the vector space  $V$ , and whose optimal value is known to equal  $\mathbb{E}[Z]$ , and  $P_{a+V} Z = a + P_V(Z - a)$  is the affine projection.

The *covariance operator* is an element  $C(u, v) \in \mathcal{H} \otimes \mathcal{H}$  in the tensor product of bilinear functionals on  $\mathcal{H}$ , given by  $C(u, v) = \mathbb{E}[Z \otimes Z]$ . The *centred* covariance operator is given by

$$C'(u, v) = \mathbb{E}[(Z - \mathbb{E}[Z]) \otimes (Z - \mathbb{E}[Z])] = \mathbb{E}[Z' \otimes Z']$$

Identifying  $\mathcal{H} \otimes \mathcal{H}$  with the space  $\text{HS}(\mathcal{H})$  of Hilbert-Schmidt operators on  $\mathcal{H}$  by way of the mapping of elementary tensors  $u \otimes v \mapsto \langle \cdot, u \rangle_{\mathcal{H}} v$  we obtain  $C' = \mathbb{E}[\langle \cdot, Z' \rangle_{\mathcal{H}} Z']$ . When we refer to the covariance operator we may either refer to the tensor in  $\mathcal{H} \otimes \mathcal{H}$  or the operator in  $\text{HS}(\mathcal{H})$ .

A Hilbert-Schmidt operator  $L$  is an operator on a Hilbert space  $\mathcal{H}$  with finite Hilbert-Schmidt norm, given by  $\|L\|_{\text{HS}(\mathcal{H})} = \sum_j \|L e_j\|_{\mathcal{H}}$  for any orthonormal basis  $\{e_j\}_j$  in  $\mathcal{H}$  [Davies, 2007, Chapter 5]. It is a Hilbert space, with inner product  $\langle L_1, L_2 \rangle_{\text{HS}(\mathcal{H})} = \sum_j \langle L_1 e_j, L_2 e_j \rangle_{\mathcal{H}}$ . The Hilbert-Schmidt norm is always larger than or equal to the operator norm,  $\|L\| \leq \|L\|_{\text{HS}(\mathcal{H})}$ , and if  $\mathcal{H}$  is finite it coincides with the Frobenius norm,  $\|L\|_{\text{HS}(\mathcal{H})} = \|M\|_F$  where  $M$  is a matrix representation of  $L$  [Kreyszig, 1989]. In the following, the use of a matrix representation for a Hilbert-Schmidt operator will often be implicit, and where applicable we pad the matrix representation with trailing zeros.

The covariance operator  $C'$  is compact, since it is Hilbert-Schmidt, and so its spectrum is countable and all spectral values are eigenvalues apart from possibly 0. Since  $C'$  is infinite-dimensional, by assumption, the value 0 is always a spectral value. Furthermore, the covariance operator is self-adjoint, and so the spectrum is real and the resolvent spectrum is empty. Finally, it is positive and so the spectrum is positive.

The sum of the smallest eigenvalues of the centred covariance operator  $C'$  equals the minimum true reconstruction error of the centred random variable  $Z' = Z - \mathbb{E}[Z]$ . The eigenvectors form a countable orthonormal basis of  $\text{Im}(C')$ , which can be extended to a countable orthonormal basis for the entire space, since  $\mathcal{H}$  is separable. Denoting the eigenvalues by  $\{\lambda_j\}_{j=1}^{\infty}$  in decreasing order, the minimum reconstruction error can be written  $R(V_d) = \sum_{j=d+1}^{\infty} \lambda_j$ .

Replacing the measure  $\mathbb{P}_Z$  on  $\mathcal{H}$  by the empirical measure  $\mathbb{P}_n = \frac{1}{n} \sum_{i=1}^n \delta_{z_i}$ , where  $\delta_x$  is the Dirac delta function, we obtain the empirical covariance operator  $C'_n : \mathcal{H} \rightarrow \mathcal{H}$

$$C'_n = \frac{1}{n} \sum_{i=1}^n \langle \cdot, z'_i \rangle_{\mathcal{H}} z'_i$$

We denote its eigenvalues by  $\hat{\lambda}_1^n, \hat{\lambda}_2^n, \dots, \hat{\lambda}_n^n$  in decreasing order and the corresponding eigenvectors by  $\hat{\phi}_1^n, \hat{\phi}_2^n, \dots, \hat{\phi}_n^n$ . It has finite rank, and so the spectrum only contains eigenvalues, and may or may not include 0. The minimum empirical reconstruction error is given by its smallest eigenvalues,  $R_n(\hat{V}_d) = \sum_{j=d+1}^n \hat{\lambda}_j^n$ , and it can be decomposed as  $C'_n = \sum_{j=1}^n \hat{\lambda}_j^n \langle \cdot, \hat{\phi}_j^n \rangle_{\mathcal{H}} \hat{\phi}_j^n$ .

If  $s : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$  is square-integrable in the second variable, then the operator given by

$$T_s f = \int_{\mathcal{X}} s(x, y) f(y) d\mathbb{P}_X(y)$$

is an isometry of  $L^2(\mathcal{X}, \mathcal{A}_X, \mathbb{P}_X; \mathbb{R})$  into the RKHS with kernel  $k(x, y) = \int_{\mathcal{X}} s(x, z) s(z, y) d\mathbb{P}_X(z)$  [Paulsen and Raghupathi, 2016]. One may also consider the integral operator

$$T_k f = \int_{\mathcal{X}} k(x, y) f(y) d\mathbb{P}_X(y)$$

which is equal to  $T_k = T_s^2$  and whose eigenvalues equal those of the covariance operator  $C$  [Shawe-Taylor et al., 2005].

If one replaces the probability measure  $\mathbb{P}_X$  by its empirical equivalent  $\mathbb{P}_n = \frac{1}{n} \sum_{i=1}^n \delta_{x_i}$  with respect to the data points  $\{x_i\}_{i=1}^n$  one again obtains an empirical operator  $T_n$

$$T_n f = \int_{\mathcal{X}} k(x, y) f(y) d\mathbb{P}_n(y) = \frac{1}{n} \sum_{i=1}^n k(x, x_i) f(x_i)$$

The sampling operator  $G_n$  is defined through  $f \mapsto \frac{1}{\sqrt{n}}(f(x_1), f(x_2), \dots, f(x_n))$  and it is an isometry of  $L^2(\mathcal{X}, \mathcal{A}_X, \mathbb{P}_X; \mathbb{R})$  into  $\mathbb{R}^n$  which identifies  $T_n$  with  $K$  [Koltchinskii and Giné, 2000]. Its adjoint  $G_n^*$  is given by  $\alpha \mapsto \frac{1}{\sqrt{n}} \sum_{i=1}^n \alpha_i k(x_i, x)$  [Rudi et al., 2015]. Furthermore,  $C_n = G_n^* G_n$  and  $\frac{1}{n} K = G_n C_n^*$ .

And so the eigenvalues of the empirical kernel integral operator  $T_n$  are the same as the eigenvalues of the kernel matrix, and its eigenvectors are given by [Bengio et al., 2004]

$$\hat{\psi}_j^n = \frac{\sqrt{n}}{\hat{\lambda}_j^n} \sum_{i=1}^n u_{j,i} k(x_i, x) = \frac{\sqrt{n}}{\hat{\lambda}_j^n} u_j^T \kappa(x)$$

The values of  $\hat{\psi}_j^n(x)$  at the points  $x_1, x_2, \dots, x_n$  equal the corresponding entries in the eigenvector of the kernel matrix  $K$ ,  $\hat{\psi}_j^n(x_i) = (u_j)_i$ , where  $u_j$  is the  $j$ th eigenvector of  $K$ .

If we randomly sample  $m < n$  indices  $S = \{r_1, r_2, \dots, r_m\}$  and then take the values of  $\hat{\psi}_r^m$ ,  $r \in S$  at all the data points  $x_1, x_2, \dots, x_n$ , and normalize by  $\frac{1}{\sqrt{n}}$ , we obtain the Nyström approximation [Williams and Seeger, 2001],

$$(1) \quad \tilde{\lambda}_j = \frac{n}{m} \hat{\lambda}_j^m$$

$$(2) \quad \tilde{u}_j = \sqrt{\frac{m}{n}} \frac{1}{\hat{\lambda}_j^m} K_{nm} u_j$$

where  $\hat{\lambda}_j^m$  are the eigenvalues of  $K_{mm}$ , which contains the  $m$  subsampled columns and rows of  $K$  corresponding to the chosen indices, and  $K_{nm}$  is the  $m$  subsampled columns. Multiplying together the approximate eigenvalues (1) and eigenvectors (2) one so obtains an approximate kernel matrix  $\tilde{K} = K_{nm} K_{mm}^{-1} K_{mn}$ , where  $K_{mn}$  is the transpose of  $K_{nm}$ . The approximate kernel matrix can serve as a replacement of the original kernel matrix for improved computational efficiency for different kernel methods.

Kernel methods in machine learning look for functions  $f$  in the reproducing kernel Hilbert space to be adapted to data

$$f(x) = \sum_{i=1}^n \alpha_i \langle \phi(x_i), \phi(x) \rangle_{\mathcal{H}} = \sum_{i=1}^n \alpha_i k(x_i, x)$$

where  $\{\alpha_i\} \in \mathbb{R}^n$  are parameters. The Nyström method may also be defined by restricting these functions to lie in the linear span of the  $m$  subsampled data points  $\{\phi(x_r)\}_{r \in S}$ , while using the full dataset of  $n$  points for estimation of the unknown parameters [Rudi et al., 2015]. For fixed  $S$  the linear span of  $\{\phi(x_r)\}_{r \in S}$  is a closed subspace of  $\mathcal{H}$  and so is a Hilbert space, which we will denote by  $\mathcal{H}_S$  [Bollobás, 1999]. In other words, one looks for functions of the form

$$f(x) = \sum_{r \in S} \alpha_r \langle \phi(x_r), \phi(x) \rangle_{\mathcal{H}} = \sum_{r \in S} \alpha_r k(x_r, x)$$

that solve an estimation problem based on all data points  $\{x_i\}_{i=1}^n$ , such as an empirical risk minimization procedure.

After drawing the  $n$  observations  $\{x_i\}_{i=1}^n$  independently from  $\mathbb{P}_X$ , the subset of  $m$  data points  $\{x_r\}_{r \in S} = \{x_{r_1}, x_{r_2}, \dots, x_{r_m}\}$  is randomly selected according to a specified distribution that may depend on the observed values  $p(S|\{x_i\}_{i=1}^n)$ . Before the data points are observed the elements in the subset are random variables  $\{X_{r_1}, X_{r_2}, \dots, X_{r_m}\}$ . For notational convenience we will assume that the data points are reordered after the subsampling so that  $\{x_r\}_{r \in S} = \{x_1, x_2, \dots, x_m\}$ .

Kernel PCA may be obtained by appealing to the  $\ell^2(\mathbb{R})$  representation of a separable real Hilbert space and arranging the data points in  $\mathcal{H}$  in a data matrix  $\Phi$  with one data point occupying a row, which may then have an infinite number of columns. With zero-mean data in feature space the principal components are then the eigenvectors of  $\frac{1}{n} \Phi^T \Phi$  and the kernel matrix can be written as  $K = \Phi \Phi^T$ . The mean can be subtracted in the RKHS (the feature space) through [Schölkopf et al., 1998]

$$K' = (\Phi - \mathbb{1}_n \Phi)(\Phi - \mathbb{1}_n \Phi)^T = K - \mathbb{1}_n K - K \mathbb{1}_n + \mathbb{1}_n K \mathbb{1}_n$$

where  $\mathbb{1}_n$  is a matrix for which  $(\mathbb{1}_n)_{i,j} = \frac{1}{n}$ . The eigenvalues of  $K' = Q \Lambda Q^T$  scaled by  $\frac{1}{n}$  then measure the variance of the data projected onto each individual principal component. Its eigenvectors

$Q$  are proportional to the principal scores – the principal scores are given by  $S = Q\Lambda^{1/2}$ . By the singular value decomposition  $\Phi - \mathbf{1}_n\Phi = Q\Sigma E^T$ , where  $\Lambda = \Sigma^2$ , the principal scores of a *new* data point  $x^*$  which is centred in feature space is given by

$$\begin{aligned} w^* &= ((\phi(x^*) - \mathbf{1}_n\Phi)E)^T = ((\phi(x^*) - \mathbf{1}_n\Phi)(\Phi - \mathbf{1}_n\Phi)^T Q\Lambda^{-1/2})^T \\ &= ((\kappa(x^*)^T - \kappa(x^*)^T \mathbf{1}_n - \mathbf{1}_n K + \mathbf{1}_n K \mathbf{1}_n) Q\Lambda^{-1/2})^T \\ &= \Lambda^{-1/2} Q^T (\kappa(x^*) - \mathbf{1}_n \kappa(x^*) - K \mathbf{1}_n + \mathbf{1}_n K \mathbf{1}_n) =: \Lambda^{-1/2} Q^T \kappa'(x^*) = S^{-1} \kappa'(x^*) \end{aligned}$$

where  $\phi(x_i)$  is an element in  $\ell^2(\mathbb{R})$  as a row vector,  $\mathbf{1}_n$  is a length- $n$  column vector with each element equal to  $\frac{1}{n}$  and  $\kappa(x) = (k(x_1, x), k(x_2, x), \dots, k(x_n, x))^T$ .

Using this formula to calculate the scores for the *original* data points we get that  $\kappa'(x^*)$  becomes  $K'$  and obtain  $w^{*T} = K' Q \Lambda^{-1/2} = Q \Lambda Q^T Q \Lambda^{-1/2} = Q \Lambda^{1/2}$  and so as expected we recover the previous expression for the principal scores.

When applying PCA to a real-world problem it is often appropriate to normalize the input variables to have variance 1, so as to make the analysis independent of arbitrary changes of units in the data. Otherwise the variables with higher variance will also dominate the principal components and comparisons between variables become difficult. This normalization will often also be appropriate for kernel PCA and we do this for the experimental analysis (Section 7). The centring of variables in the feature space does not guarantee that the input variables become centred.

Multi-dimensional scaling (MDS) finds a lower-dimensional representation of data from a matrix of distances between data points [Hout et al., 2013]. MDS is equivalent to kernel PCA when the kernel is *isotropic*, i.e. on the form  $f(\|x - y\|)$  for some function  $f$  [Williams, 2002]. Therefore, theoretical or practical results for kernel PCA are often also applicable to MDS.

The approximate eigenvalues and eigenvectors from the original Nyström method in Equations (1) and (2) may be used to define an approximate kernel PCA. However, these approximate eigenvectors are not orthogonal and do not yield uncorrelated principal scores, so they do not define true PCA, and the eigenvalues do not describe the variance captured by the principal components, since they are simply the eigenvalues of  $K_{mm}$  scaled by a factor  $\frac{n}{m}$ . There is a need for another way to derive kernel PCA with the Nyström method.

#### 4. KERNEL PCA WITH THE NYSTRÖM METHOD

In this section we present kernel PCA with the Nyström method, which provides an efficient and flexible technique for non-linear PCA. We present the corresponding quantities that are defined for linear PCA and are useful for data exploration and application of the method in downstream tasks

- (1) a set of orthogonal principal components with unit length in the linear span of the subsampled data points in  $\mathcal{H}$  (denoted  $\mathcal{H}_S$ ),
- (2) the variance of the data along each of these directions, termed the explained variance,
- (3) the reconstruction error of the data onto the principal components,

- (4) a set of uncorrelated principal scores with the weightings of the data points on the principal components, and,
- (5) the principal scores of a new data point with respect to the existing principal components

For standard kernel PCA (2) and (3) are the same, but with the Nyström method they are different, since the principal components will not span the entire data.

We first present the principal components, explained variance and scores for a dataset in the following theorem

**Theorem 1** (Nyström kernel PCA). *Let  $(\tilde{\lambda}_j, v_j)$  be the eigenpairs and  $V\tilde{\Lambda}V^T$  be the eigendecomposition of*

$$\tilde{K}' = \frac{1}{n} K_{mm}'^{-1/2} K_{mn}' K_{nm}' K_{mm}'^{-1/2}$$

where

$$K_{mn}' = K_{mn} - K_{mn}\mathbf{1}_n - \mathbf{1}_n^{m,n} \tilde{K} + \mathbf{1}_n^{m,n} \tilde{K} \mathbf{1}_n$$

$$K_{mm}' = K_{mm} - \mathbf{1}_n^{m,n} K_{nm} - K_{mn} \mathbf{1}_n^{n,m} + \mathbf{1}_n^{m,n} \tilde{K} \mathbf{1}_n^{m,n}$$

with  $\tilde{K} = K_{nm} K_{mm}^{-1} K_{mn}$  and where  $\mathbf{1}_n$ ,  $\mathbf{1}_n^{n,m}$  and  $\mathbf{1}_n^{m,n}$  are  $n \times n$ ,  $n \times m$  and  $m \times n$  matrices respectively with each element equal to  $\frac{1}{n}$ .

The perpendicular intersecting lines  $\phi_0 + \langle \tilde{\phi}_j \rangle$ ,  $j = 1, 2, \dots, m$  in  $\mathcal{H}_S$  along which the variance of the data is successively maximized, where the orthonormal vectors  $\{\tilde{\phi}_j\}_{j=1}^m$  are termed the principal components, are given by

$$\phi_0 = \frac{1}{n} K_{nm} K_{mm}^{-1} \kappa_m(x)$$

$$\tilde{\phi}_j = \sum_{k=1}^m u_{j,k} (k(x_k, x) - \phi_0)$$

and the variances along these directions are  $\{\tilde{\lambda}_j\}_{j=1}^m$ , termed the principal values or explained variance, where  $\kappa_m(x) = (k(x_1, x), k(x_2, x), \dots, k(x_m, x))^T$ ,  $u_j = K_{mm}'^{-1/2} v_j$  and  $U = K_{mm}'^{-1/2} V$ .

The projection coefficients of the centred data points onto the principal components, termed the principal scores, are given by

$$W = K_{nm}' U = K_{nm}' K_{mm}'^{-1/2} V$$

where each row of  $W$  contains the scores of one data point onto the principal components. The principal scores of a new data point  $x^*$  is given by

$$w^* = U^T (\kappa_m(x^*) - K_{mn} \mathbf{1}_n - \mathbf{1}_n^{m,n} K_{nm} K_{mm}^{-1} \kappa_m(x^*) + \mathbf{1}_n^{m,n} \tilde{K} \mathbf{1}_n) = U^T \tilde{\kappa}(x^*)$$

where  $\mathbf{1}_n$  is a length- $n$  column vector given by  $\mathbf{1}_n = (\frac{1}{n}, \frac{1}{n}, \dots, \frac{1}{n})^T$ .

The principal components can be seen as defining new variables through linear combinations of the existing variables that have successively maximized variance and that are uncorrelated. The values of these new variables are given by the principal scores, which represent the data in a new coordinate system defined by the principal components as a new basis for the space. As such, the principal scores can be used as a drop-in replacement for the original data in arbitrary supervised or unsupervised learning methods, including after removing the scores corresponding to principal components with smaller eigenvalues. Please see Section 8 for an example of this.

To see that these new variables are uncorrelated also with the Nyström method we note that

$$W^T W = V^T K_{mm}'^{-1/2} K_{mn}' K_{nm}' K_{mm}'^{-1/2} V = n V^T V \tilde{\Lambda} V^T V = n \tilde{\Lambda}$$

which is a diagonal matrix.

The scores of new data points are important when measuring the accuracy of PCA with a test set of hold-out data points, for example using the reconstruction error (Section 7), or when applying PCA as a preprocessing step for supervised learning methods and one wishes to create predictions for new data points, such as in principal component regression (Section 8).

The computational complexity of the method is  $\mathcal{O}(nm^2)$  in time, which is the same as the Nyström method applied to regression. Centring of the matrix  $K_{mn}$  can be accomplished in  $\mathcal{O}(m^3 + nm)$  operations, and so the centring in the proposed method adds no additional time requirements to the dominant  $\mathcal{O}(nm^2)$  factor. We refer to the software implementation for full details<sup>3</sup>.

The Nyström method approximates the corresponding full method, so when  $m = n$  we should recover standard kernel PCA. In this case  $\tilde{K} = K K^{-1} K = K$  and as expected  $K_{mm}' = K_{nm}' = K'$  and

$$K_{mm}'^{-1/2} K_{mn}' K_{nm}' K_{mm}'^{-1/2} = K'$$

and the scores are equal to  $W = K'^{1/2} V = \sqrt{n} V \tilde{\Lambda}^{1/2} V^T V = \sqrt{n} V \tilde{\Lambda}^{1/2} = Q \Lambda^{1/2}$ , which we know to be the scores for standard kernel PCA.

The smallest  $m - d$  Nyström eigenvalues  $\sum_{j=d+1}^m \tilde{\lambda}_j$  measure the residual variance of the data points *within*  $\mathcal{H}_S$  and correspond to the reconstruction error  $\frac{1}{n} \sum_{i=1}^n \|P_{\tilde{V}_d} z'_i - P_{\mathcal{H}_S} z'_i\|_{\mathcal{H}}^2$ , where  $\tilde{V}_d = \text{span}\{\{\tilde{\phi}_k\}_{k=1}^d\}$ . The full reconstruction error with respect to the top  $d$  Nyström principal components is given by

$$(3) \quad R_n(\tilde{V}_d) = \frac{1}{n} \sum_{i=1}^n \|z'_i - P_{\tilde{V}_d} z'_i\|_{\mathcal{H}}^2 = \frac{1}{n} \text{Tr}(K') - \sum_{j=1}^d \tilde{\lambda}_j$$

where  $\text{Tr}(\cdot)$  is the trace and  $\frac{1}{n} \text{Tr}(K')$  is the variance of the full dataset in  $\mathcal{H}$ . From Theorem 1 above we know that this is the smallest reconstruction error among all  $d$ -dimensional subspaces in  $\mathcal{H}_S$ .

Calculation of this quantity is  $\mathcal{O}(n^2)$  due to the centring of  $K$ . However, it can be approximated for example by subtracting the mean of  $K_{nm}$  instead of the mean of  $K$ , which becomes  $\mathcal{O}(nm)$ . This is included as an option in the software package accompanying the paper. Please see Section 7 on page 20 for further details.

<sup>3</sup>[https://github.com/fredhallgren/nystrompca/blob/main/nystrompca/algorithms/nystrom\\_KPCA.py](https://github.com/fredhallgren/nystrompca/blob/main/nystrompca/algorithms/nystrom_KPCA.py)

Note that the reconstruction error above in Equation (3) is slightly different from the reconstruction error of the uncentred data points with respect to the affine subspace  $\phi_0 + \tilde{V}_d$ , which becomes  $\frac{1}{n} \sum_{i=1}^n \|(z_i - \phi_0) - P_{\tilde{V}_d}(z_i - \phi_0)\|_{\mathcal{H}}^2 = \frac{1}{n} \sum_{i=1}^n \|(z_i - \phi_0) - P_{\tilde{V}_d} z'_i\|_{\mathcal{H}}^2$ . Both reconstruction errors are at a minimum for the proposed method.

Another quantity of interest for purposes of comparison is the reconstruction error of the full dataset on the eigenspace of the subset of  $m$  data points. Creating PCA from a random subset of  $m$  data points to describe the full dataset will be termed *Subset PCA*. We use the same subspace translation as for the Nyström method – that is to say we centre the data using the mean of the  $n$  data points projected onto  $\mathcal{H}_S$ . This ensures that the amount of variance captured is the same whether we project the centred data onto the principal components, or the uncentred data onto the lines translated from the origin. The principal components will then be given by, for  $j = 1, 2, \dots, m$

$$\hat{\phi}_j^{m,n} = \sum_{k=1}^m u_{j,k}^m (k(x_k, x) - \phi_0)$$

where  $u_j^m$  is the  $j$ th eigenvector of  $\frac{1}{m} K'_{mm}$ . The variance of the full data captured by these principal components and the associated reconstruction error are presented in the following theorem

**Theorem 2** (Subset PCA). *The variance of the dataset  $\{\phi(x_i)\}_{i=1}^n$  along the  $j$ th principal component  $\hat{\phi}_j^{m,n}$  is given by*

$$\hat{\lambda}_j^{m,n} = \frac{1}{n} \sum_{i=1}^n \|P_{\hat{\phi}_j^{m,n}} z'_i\|_{\mathcal{H}}^2 = \frac{1}{n \cdot m \hat{\lambda}_j^m} u_j^{mT} K'_{mn} K'_{nm} u_j^m$$

where  $(\hat{\lambda}_j^m, u_j^m)$  is the  $j$ th eigenpair of  $\frac{1}{m} K'_{mm}$ .

The reconstruction error of the full dataset onto the corresponding  $d$ -dimensional PCA subspace is

$$R_n(\hat{V}_d^m) = \frac{1}{n} \sum_{i=1}^n \|z'_i - P_{\hat{V}_d^m} z'_i\|_{\mathcal{H}}^2 = \frac{1}{n} \text{Tr}(K') - \frac{1}{n \cdot m} \text{Tr}(K'_{nm} U_d^m \Lambda_d^{m-1} U_d^{mT} K'_{mn})$$

where  $U_d^m \Lambda_d^m U_d^{mT}$  is the truncated eigendecomposition of  $\frac{1}{m} K'_{mm}$ .

As expected, if  $n = m = d$  then the reconstruction error is zero.

The method proposed in this section for efficient kernel PCA can also be applied to improve the scalability of MDS when these two methods are equivalent, as outlined in Section 3.

## 5. PRELUDE: A SPECIAL CASE

Before studying the statistical accuracy of kernel PCA with Nyström method through a confidence bound we present a majorization relation between Nyström and Subset PCA and consider the special case when the PCA dimension equals the number of subsampled data points,  $d = m$ . In this case the reconstruction error for the Nyström method is the same as Subset PCA, both for the empirical and true reconstruction errors.

We first present the majorization relation in the following proposition. It tells us with one concise formula that for any PCA dimension strictly less than the number of data points in the subset, kernel PCA with the Nyström method will always capture at least as much variance as PCA created directly from the subset, but that there will be no improvement when  $d = m$ .

**Proposition 1.** *We have the following majorization relation for the empirical error*

$$(\tilde{\lambda}_1, \tilde{\lambda}_2, \dots, \tilde{\lambda}_m) \succ (\hat{\lambda}_1^{m,n}, \hat{\lambda}_2^{m,n}, \dots, \hat{\lambda}_m^{m,n})$$

The majorization is strict in the sense that  $\tilde{\lambda}_{<d} > \hat{\lambda}_{<d}^{m,n}$  for  $d < m$ , by the assumption of a continuous data distribution.

A direct consequence of the proposition is that

$$R_n(\tilde{V}_m) = R_n(\hat{V}_m^m)$$

For the true reconstruction error we consider the case where the sampling of the Nyström subset occurs independently of the values of the data points

**Proposition 2.** *Let  $d = m$  and let the Nyström subset be sampled according to  $p(S | x_1, x_2, \dots, x_n)$ . Then if*

$$p(S | x_1, x_2, \dots, x_n) = p(S)$$

*i.e. the subsampling is independent of the data, we have*

$$R(\tilde{V}_m) = R(\hat{V}_m^m)$$

The above proposition includes the common case of uniform sampling for the Nyström subset. It holds whether the  $n$  data points are considered fixed or unobserved.

From the above propositions we can conclude that if retaining all the Nyström principal components then there is no gain in accuracy compared to Subset PCA from the perspective of the reconstruction error. However, for a smaller PCA dimension the Nyström method will perform strictly better than PCA directly from the subset. A more precise treatment of its accuracy for arbitrary dimensions is the subject of the next section.



## 6. STATISTICAL ACCURACY OF NYSTRÖM KERNEL PCA

In this section we provide a finite-sample confidence bound on the empirical reconstruction error of kernel PCA with the Nyström method versus the one for full kernel PCA. In line with strictly all results on the statistical accuracy on standard kernel PCA we assume that data has zero mean in feature space.

The confidence bound allows for measuring the accuracy of Nyström kernel PCA for a specific dataset, specified in the familiar language of confidence intervals, applied to the amount of variance that is left over after representing the dataset in terms of a subset of principal components.

The actual difference between the reconstruction errors of the Nyström method and standard kernel PCA for a dataset is given by

$$R_n(\tilde{V}_d) - R_n(\hat{V}_d) = \frac{1}{n} \text{Tr}(K) - \sum_{j=1}^d \tilde{\lambda}_j - \sum_{j=d+1}^m \hat{\lambda}_j^n = \hat{\lambda}_{<d}^n - \tilde{\lambda}_{<d}$$

However, the eigenvalues  $\hat{\lambda}_j^n$  of  $\frac{1}{n}K$  are not available – if they were there would be no need to apply the Nyström method. When the Nyström method is being considered for a problem then the size of the data  $n$  is very large and calculating the full kernel matrix  $K$ , let alone its eigendecomposition, is prohibitively expensive.

At a minimum, any measure of accuracy should not be more computationally demanding than the method itself, which is  $\mathcal{O}(nm^2)$ . We present a bound that does not require that we have observed the entire dataset, only the subset  $x_1, x_2, \dots, x_m$ . It takes  $\mathcal{O}(m^3)$  time to calculate and is  $\mathcal{O}(m^2)$  in memory. It holds for any subsampling distribution.

**Theorem 3** (Confidence bound). *With confidence at least  $1 - 2e^{-\delta}$  (or, with probability at least  $1 - 2e^{-\delta}$  with respect to  $\{x_i\}_{i=m+1}^n$  across repeated samples of  $\{x_r\}_{r=1}^m$ ), where  $B := \sup_x k(x, x)$ ,  $\{\hat{\lambda}_j^m\}_{j=1}^m$  are the eigenvalues of the kernel matrix  $\frac{1}{m}K_{mm}$  from the Nyström subset,  $\hat{\lambda}_0^m$  and  $\hat{\lambda}_{m+1}^m$  are defined to be  $+\infty$  and  $-\infty$  respectively, and*

$$D := \frac{n-m}{n} \frac{2B\sqrt{\delta}}{\sqrt{n-m}}$$

$$D_j := \frac{(2D)^2}{\min\{\hat{\lambda}_{j-1}^m - \hat{\lambda}_j^m, \hat{\lambda}_j^m - \hat{\lambda}_{j+1}^m\}^2} \wedge 1$$

we have

$$R_n(\tilde{V}_d) - R_n(\hat{V}_d) \leq \sum_{j=1}^d \hat{\lambda}_j^m \cdot D_j + D \cdot \max_{1 \leq k \leq d} D_k$$

The bound is stated in terms of a probability with respect to future unknown realizations of the data  $\{x_i\}_{i=m+1}^n$ , which holds across infinite repetitions of the experiment yielding the observed data  $\{x_r\}_{r=1}^m$  as in the frequentist construction of hypothesis tests or confidence intervals [Neyman and Pearson, 1933, Neyman, 1937]. Equivalently, the bound may be interpreted solely as a confidence,

both with respect to the observed data and with respect to hypothesized future realizations of the unobserved data. Stated differently, if we observe the subset  $\{x_r\}_{r=1}^m$ , calculate the bound with some specified confidence level  $\alpha$ , say 95 %, and observe the future data  $\{x_i\}_{i=m+1}^n$ , then the realized difference in reconstruction errors will lie within the bound at least 95 % of the time if we repeat this procedure indefinitely.

The behaviour of the bound is as one would expect from a measure of the statistical accuracy of the Nyström method compared to the full method – it decreases as  $m$  increases, *ceteris paribus*, and becomes zero if  $n = m$ . It increases with the dimension  $d$  of the PCA subspaces that are being compared.

Application of the bound does not require that we have observed the entire sample. For example, if data is generated sequentially and iid from  $p_X(x)$  then picking the first  $m$  points for the Nyström subset is equivalent to sampling all points and then selecting  $m$  points uniformly (in the sense that the data points in the subset have the same distribution in both instances).

If data is stored on disk, and reading from disk is expensive, then only  $m$  records need to be read in order to calculate the bound, assuming this can be done in such a way as to respect the sampling distribution of the subset of data points<sup>4</sup>.

The bound becomes infinite if  $k(x, x)$  is not bounded for all  $x$ . One may create a bounded kernel from an unbounded one through the transformation

$$(4) \quad k'(x, y) := \frac{k(x, y)}{\sqrt{k(x, x)k(y, y)}}$$

which has  $\sup_x k'(x, x) = 1$ , although the transformed kernel has somewhat different characteristics and induces a different RKHS. The transformation corresponds to scaling all feature vectors to have norm 1.

**Proof outline.** A proof outline is as follows. Please see the appendix for a full proof.

1. Rewrite the difference in reconstruction errors in terms of the eigenpairs of the empirical operators  $C_n$  and  $C_m$ , to obtain

$$R_n(\tilde{V}_d) - R_n(\hat{V}_d) \leq \sum_{j=1}^d \hat{\lambda}_j^n \left(1 - \langle \hat{\phi}_j^n, \hat{\phi}_j^m \rangle_{\mathcal{H}}^2\right)$$

2. Apply the Davis-Kahan theorem to convert the angle between the eigenvectors into a difference between successive eigenvalues of  $C_m$  and the norm of the difference between the empirical operators  $\|C_n - C_m\|_{\text{HS}(\mathcal{H})}$
3. Convert the unknown eigenvalues  $\hat{\lambda}_j^n$  into the ones based on the observed data  $\hat{\lambda}_j^m$  plus the difference  $\|C_n - C_m\|_{\text{HS}(\mathcal{H})}$ , using Lidskii's inequality

<sup>4</sup>In many implementations of the SQL query language, including MySQL and PostgreSQL, this would correspond to appending `LIMIT (m)` to the end of the query, which interrupts it after finding the first  $m$  records [Beaulieu, 2020]

4. Now  $\|C_n - C_m\|_{\text{HS}(\mathcal{H})}$  is the only unknown quantity left. Split up the empirical operators into two independent ones through

$$\|C_n - C_m\|_{\text{HS}(\mathcal{H})} = \frac{n-m}{n} \|C_{n-m} - C_m\|_{\text{HS}(\mathcal{H})}$$

where  $C_{n-m} = \frac{1}{n-m} \sum_{i=m+1}^n z_i \otimes z_i$

5. Apply Hoeffding's inequality in Banach spaces, obtaining

$$\mathbb{P}\left(\|C_{n-m} - C_m\|_{\text{HS}(\mathcal{H})} \leq 2B\sqrt{\delta}/\sqrt{n-m}\right) \geq 1 - 2e^{-\delta}$$

6.1. **A corollary.** From Lemma 1 on page 32 which is used in the proof of the confidence bound one can deduce sharper versions of Theorem 7 and Propositions 10 and 11 from Rosasco et al. [2010], by a factor  $1/\sqrt{2}$  or  $1/2$ . These follow since the covariance operator  $C$  and its empirical equivalent  $C_n$  are positive, and then by the lemma their difference  $\|C - C_n\|_{\text{HS}(\mathcal{H})}$  is bounded by  $\sqrt{2} \sup_x k(x, x)$ , rather than  $2 \sup_x k(x, x)$ .

For Theorem 7, the sharper result states that with probability at least  $1 - 2e^{-\delta}$  we have

$$\|C - C_n\|_{\text{HS}(\mathcal{H})} \leq \frac{2B\sqrt{\delta}}{\sqrt{n}}$$

The sharper version of Proposition 10 states that with probability  $1 - 2e^{-\delta}$

$$\sum_{j=1}^{\infty} \left(\lambda_j - \hat{\lambda}_j^n\right)^2 \leq \frac{4B^2\delta}{n} \qquad \sup_j |\lambda_j - \hat{\lambda}_j^n| \leq \frac{2B\sqrt{\delta}}{\sqrt{n}}$$

And for Proposition 11 we obtain that also with probability  $1 - 2e^{-\delta}$

$$\left| \sum_{j=1}^{\infty} \lambda_j - \sum_{j=1}^n \hat{\lambda}_j^n \right| = |\text{Tr}(C) - \text{Tr}(C_n)| \leq \frac{2B\sqrt{\delta}}{\sqrt{n}}$$

A number of other results can also be sharpened using the same technique, including, but not limited to, Theorem 2 in De Vito et al. [2005b], Theorem 1 in De Vito et al. [2005a], Lemma 1 in Zwald and Blanchard [2005], Theorem 6.2 in Giraldo et al. [2014], Theorem 4.2 in Bouvrie and Hamzi [2012] and Lemma 4.1 in Blanchard and Zadorozhnyi [2019].

## 7. EXPERIMENTAL ANALYSIS

In this section we illustrate the method and bound through experiments on real-world datasets with different kernel functions. We first compare the proposed method to a number of other unsupervised learning methods by measuring the reconstruction error on hold-out datasets. We then evaluate the bound and compare it to the actual errors and the errors for Subset PCA.

The methods and experiments are implemented in the Python programming language and the source code is available at <https://github.com/fredhallgren/nystrompca>. The package can be installed with one simple command using the Python package manager<sup>5</sup>. It includes a command-line tool to run the different experiments with different parameter values and kernel functions.

For purposes of reproducibility the computer experiments allow for setting the random seed of the pseudo-random number generator [Robert and Casella, 2004], to produce exactly the same results every time the experiments are run. Other than the random sampling of the Nyström subset, randomness is also present in the splitting of data into training and test sets.

The principal components are unique only up to a sign, so in the package we switch the sign of the scores and components such that the range of values in each dimension of the scores is mostly positive. This will ensure that we will get exactly the same values for the scores and components every time we run the algorithm.

We use different datasets from the UCI Machine Learning Repository [Dua and Graff, 2017]. Dimensionality reduction can be particularly important for high-dimensional data, so we include a number of such datasets. We use the simulated `magic` gamma telescope dataset, the `yeast` dataset, containing cellular protein location sites for fungi, the `cardiotocography` dataset, with heart measurements, the `segmentation` dataset containing various data on images, the `drug` dataset with personality traits and drug consumption, the `digits` dataset with flattened  $8 \times 8$  pixel grayscale images, and two bag-of-words datasets with bag-of-words vectors of articles from `www.dailykos.com` and NIPS papers, respectively. We tabulate some information on the datasets used in one or both of the experiments below in Table 2, where the number of attributes is before any data transformation. For comparability we cut each dataset to 1000 data points when running the experiments. For both experiments we sample the Nyström subset uniformly without replacement and we use the same sampled subset for both Nyström PCA and Subset PCA.

	<i>Dataset</i>	<i>Data size</i>	<i>Number of attributes</i>
1	<code>magic</code>	19020	11
2	<code>yeast</code>	1484	8
3	<code>cardiotocography</code>	2126	23
4	<code>segmentation</code>	2310	19
5	<code>drug</code>	1885	32
6	<code>digits</code>	5620	64
7	<code>dailykos</code>	3430	6906
8	<code>nips</code>	1500	12419

TABLE 2. Datasets used

<sup>5</sup>`pip install nystrompca`

We convert ordinal variables to integers and categorical variables to discrete ones through one-hot encoding. We treat discrete numerical variables in the data as continuous for the purposes of PCA. We remove any date or time variables. We also remove variables that are constant. These will differ depending on how many data points we include in the total dataset when we run the experiments.

We normalize the input data to have mean zero and variance one. Note that this does not mean that data has zero mean in the feature space. As previously mentioned, normalizing the input data makes the analysis independent of the units used to measure the variables and unaffected by the scale of the variables, which may otherwise dominate the PCA results. Furthermore, it makes it easier to compare results across different datasets and kernel functions and can make the same kernel parameters appropriate for different datasets.

We cut eigenvalues that are smaller than  $10^{-12}$  when performing matrix inversions to improve the condition number of the matrix. We also remove any negative eigenvalues – in theory all kernel matrices will be positive definite, however numerical inaccuracies may occasionally lead to small negative eigenvalues in practice.

We use three different kernel functions, the radial basis functions (RBF), polynomial and Cauchy kernels, summarized below in Table 3. The software package includes a number of additional kernel functions that can be used when running either of the experiments.

<i>Kernel</i>	<i>Functional form</i> $k(x, y)$	<i>Parameters</i>	<i>Bound</i> $\sup_x k(x, x)$
RBF	$\exp\left\{-\frac{\ x-y\ ^2}{\sigma^2}\right\}$	$\sigma \in \mathbb{R}_+ \setminus \{0\}$	1
Polynomial	$(\langle x, y \rangle + R)^d$	$R \in \mathbb{R}, d \in \mathbb{N}$	$\infty$
Cauchy	$\frac{1}{1+\ x-y\ ^2/\sigma^2}$	$\sigma \in \mathbb{R}_+ \setminus \{0\}$	1

TABLE 3. Kernel functions used

**7.1. Methods comparison.** We compare the proposed method to other unsupervised learning techniques to evaluate its behaviour. We compare with linear PCA, full kernel PCA, subset PCA, sparse PCA, which finds sparse eigenvectors [Wang et al., 2016], locally linear embeddings (LLE), that fits a lower-dimensional manifold to the data [Roweis and Saul, 2000] and independent component analysis (ICA), which extracts signals that are independent [Hyvärinen and Oja, 2000]. We run the methods for all the datasets in Table 2 above. We split each dataset randomly in half, fitting the methods on one half and then evaluating them on the other half. We compare the fraction of variance captured for the different methods for different dimensions. Note that kernel PCA and Nyström kernel PCA measure the variances captured in the RKHS and not in the input space.

For this experiment we only display the results for the RBF kernel and we use a Nyström subset of size  $m = 100$ . For the first six datasets we calculate the bandwidth parameter as the median distance between pairs of data points, which is a common heuristic for the RBF kernel [Garreau et al., 2017]. Using all pairs of data points is quadratic in the total number of data points, so we only use the data points in the Nyström subset. For the last two datasets we set the bandwidth parameter to  $\sigma = 500$ , which was manually tuned. Please see Table 4 for the full results, where we have set the random seed to 1. Sparse PCA is computationally demanding for very high-dimensional data, so we don't run it for all the datasets.

<i>Dataset</i>	<i>d</i>	<i>Subset PCA</i>	<i>Nyström PCA</i>	<i>Kernel PCA</i>	<i>Linear PCA</i>	<i>Sparse PCA</i>	<i>LLE</i>	<i>ICA</i>
magic								
	1	0.2401	0.2494	0.2500	0.5089	0.5027	0.0660	0.0937
	2	0.3612	0.3746	0.3760	0.6223	0.6353	0.1719	0.1798
	3	0.4187	0.4417	0.4450	0.7170	0.7303	0.2828	0.2630
	4	0.4760	0.5063	0.5102	0.7926	0.7341	0.4151	0.3488
	5	0.5387	0.5653	0.5690	0.8575	0.7798	0.5243	0.3488
	6	0.5731	0.6052	0.6093	0.9249	0.8395	0.6210	0.5535
	7	0.6047	0.6372	0.6423	0.9648	0.8658	0.7499	0.6962
	8	0.6243	0.6622	0.6688	0.9808	0.8981	0.7954	0.6962
	9	0.6450	0.6843	0.6911	0.9979	0.9588	0.8997	0.6962
	10	0.6613	0.7042	0.7115	1.0000	0.9803	1.0000	1.0000
yeast								
	1	0.1184	0.1400	0.1401	0.1182	0.1150	0.0527	0.0431
	2	0.2350	0.2562	0.2566	0.2207	0.2132	0.1367	0.0911
	3	0.3419	0.3714	0.3727	0.3144	0.3050	0.1978	0.1223
	4	0.4138	0.4418	0.4437	0.4179	0.4111	0.2752	0.1977
	5	0.4532	0.4901	0.4922	0.5058	0.4963	0.2892	0.2598
	6	0.5066	0.5362	0.5385	0.5876	0.5992	0.3815	0.3149
	7	0.5438	0.5681	0.5716	0.6462	0.6472	0.4350	0.3655
	8	0.5724	0.5985	0.6020	0.6877	0.7012	0.5215	0.4145
	9	0.5997	0.6329	0.6364	0.7520	0.7665	0.5371	0.4569
	10	0.6279	0.6475	0.6512	0.7949	0.8026	0.6000	0.5056
cardiotocography								
	1	0.1398	0.1422	0.1430	0.2173	-	0.0251	0.0260
	2	0.2100	0.2351	0.2370	0.3687	-	0.0645	0.0521
	3	0.2946	0.3105	0.3133	0.4636	-	0.0862	0.0765
	4	0.3496	0.3674	0.3714	0.5364	-	0.1163	0.1019
	5	0.3897	0.4091	0.4139	0.5831	-	0.1524	0.1264
	6	0.4240	0.4439	0.4501	0.6282	-	0.1704	0.1539
	7	0.4446	0.4752	0.4826	0.6678	-	0.2064	0.1790
	8	0.4668	0.5033	0.5103	0.7076	-	0.2212	0.2051
	9	0.4899	0.5299	0.5380	0.7452	-	0.2540	0.2296
	10	0.5233	0.5561	0.5658	0.7770	-	0.2931	0.2576
segmentation								
	1	0.2491	0.2546	0.2548	0.4044	0.3969	0.0366	0.0387
	2	0.3747	0.3775	0.3782	0.5055	0.4934	0.0799	0.1223
	3	0.4801	0.4864	0.4872	0.6077	0.5800	0.1330	0.1573
	4	0.5224	0.5340	0.5350	0.6519	0.6236	0.2121	0.1934
	5	0.5645	0.5827	0.5840	0.7135	0.6746	0.2502	0.2429
	6	0.6003	0.6318	0.6338	0.7626	0.7464	0.3072	0.3109
	7	0.6478	0.6657	0.6687	0.8730	0.8374	0.3475	0.3676
	8	0.6707	0.6877	0.6908	0.9098	0.8635	0.4149	0.4284
	9	0.6895	0.7091	0.7128	0.9316	0.8904	0.4833	0.4739
	10	0.7060	0.7341	0.7380	0.9623	0.9121	0.5491	0.6188

*Table continues on the next page*

<i>Dataset</i>	<i>d</i>	<i>Subset PCA</i>	<i>Nyström PCA</i>	<i>Kernel PCA</i>	<i>Linear PCA</i>	<i>Sparse PCA</i>	<i>LLE</i>	<i>ICA</i>
drug								
	1	0.1338	0.1395	0.1422	0.2316	-	0.0374	0.0278
	2	0.1684	0.1787	0.1833	0.3031	-	0.0381	0.0573
	3	0.2005	0.2214	0.2279	0.3594	-	0.0919	0.0874
	4	0.2256	0.2458	0.2532	0.4060	-	0.0997	0.1149
	5	0.2440	0.2728	0.2821	0.4463	-	0.1810	0.1418
	6	0.2766	0.2999	0.3098	0.4847	-	0.1936	0.1699
	7	0.2962	0.3209	0.3331	0.5087	-	0.2579	0.1905
	8	0.3153	0.3478	0.3623	0.5511	-	0.2817	0.2276
	9	0.3321	0.3639	0.3796	0.5795	-	0.3198	0.2530
	10	0.3474	0.3797	0.3968	0.6037	-	0.3618	0.2766
digits								
	1	0.0754	0.0704	0.0721	0.0253	-	0.0109	0.0148
	2	0.1299	0.1491	0.1528	0.0455	-	0.0234	0.0289
	3	0.1796	0.2057	0.2123	0.0644	-	0.0360	0.0442
	4	0.2256	0.2522	0.2614	0.0801	-	0.0492	0.0595
	5	0.2733	0.2978	0.3090	0.0951	-	0.0580	0.0772
	6	0.3008	0.3258	0.3389	0.1239	-	0.0626	0.0914
	7	0.3233	0.3550	0.3716	0.1566	-	0.0805	0.1067
	8	0.3494	0.3807	0.3997	0.1791	-	0.0952	0.1221
	9	0.3766	0.4054	0.4274	0.3674	-	0.1160	0.1373
	10	0.3952	0.4261	0.4498	0.3755	-	0.1289	0.1534
dailykos								
	1	0.0014	0.0043	0.0047	0.0046	-	0.0025	0.0033
	2	0.0016	0.0100	0.0122	0.0073	-	0.0046	0.0040
	3	0.0023	0.0104	0.0131	0.0080	-	0.0081	0.0048
	4	0.0025	0.0107	0.0138	0.0090	-	0.0111	0.0054
	5	0.0049	0.0111	0.0142	0.0093	-	0.0139	0.0058
	6	0.0054	0.0115	0.0145	0.0097	-	0.0152	0.0062
	7	0.0056	0.0116	0.0148	0.0100	-	0.0168	0.0065
	8	0.0058	0.0120	0.0152	0.0103	-	0.0170	0.0069
	9	0.0062	0.0122	0.0156	0.0108	-	0.0198	0.0071
	10	0.0063	0.0125	0.0160	0.0112	-	0.0222	0.0074
nips								
	1	0.0010	0.0026	0.0020	0.0008	-	0.0028	0.0046
	2	0.0018	0.0034	0.0062	0.0013	-	0.0049	0.0114
	3	0.0035	0.0043	0.0083	0.0016	-	0.0068	0.0164
	4	0.0038	0.0050	0.0092	0.0017	-	0.0077	0.0187
	5	0.0038	0.0067	0.0095	0.0020	-	0.0095	0.0195
	6	0.0041	0.0095	0.0098	0.0023	-	0.0101	0.0216
	7	0.0045	0.0098	0.0107	0.0025	-	0.0143	0.0242
	8	0.0046	0.0100	0.0110	0.0027	-	0.0152	0.0267
	9	0.0051	0.0102	0.0115	0.0029	-	0.0155	0.0281
	10	0.0053	0.0105	0.0118	0.0032	-	0.0158	0.0305

TABLE 4. Comparison of the variance captured by different dimensionality reduction methods across the maximum dimension  $d$

To run these experiments using the supplied command-line tool one would do

```
> nystrompca methods --seed 1
```

Note that the purpose of each of these methods is not necessarily to capture as much variance as possible, however it can still be enlightening to contrast this quantity between different methods. Furthermore, since linear PCA acts in the input space and kernel PCA and its derivations act in the feature space, comparisons of the amount of variance captured are not necessarily clear-cut. Even when linear PCA captures more variance than kernel PCA, the latter may give better performance in downstream tasks, for example when dimensionality reduction is used for preprocessing before carrying out regression or classification.

For all datasets the performance of Nyström kernel PCA is very close to the method it is attempting to approximate, despite being many times more efficient. Nyström kernel PCA also almost always captures more variance than Subset PCA, in particular for the two high-dimensional bag-of-words datasets (*dailykos* and *nips*). Only for the *digits* dataset with a PCA dimension of 1 is the opposite true. Since we are calculating the reconstruction error on a hold-out dataset it's possible that Subset PCA achieves better performance – we know this to be impossible for the training dataset by Proposition 1. For datasets with a small number of dimensions standard linear PCA captures the most amount of variance whilst being simpler and more computationally efficient, and so may be the preferred method. The results for sparse PCA are very similar to linear PCA, despite having fewer non-zero entries in the eigenvectors. LLE and ICA generally capture the least amount of variance. LLE is often a good method when the data really lies close to a low-dimensional manifold, which may not be the case for any of the datasets included above. ICA does not attempt to maximize the variance of the new representations and may be preferred when other advantages are being sought. All methods struggle to explain much of the data with only 10 dimensions for the last two datasets that have the highest dimensionality.

Calculation of Nyström kernel PCA takes on average 0.988 seconds across the eight datasets on an AWS EC2 m5.large instance with an Intel Xeon® Platinum 8175M CPU<sup>6</sup> running Ubuntu Server 20.04 with Linux kernel version 5.4, versus 2.753 seconds for full kernel PCA ( $n = 500$ ,  $m = 100$ ). In both instances the kernel matrices are created in Python whilst the eigendecomposition uses built-in LAPACK routines written in Fortran<sup>7</sup>. For these values of  $n$  and  $m$  the cubic time complexity is not attained and the constant, linear and quadratic factors are still important.

**7.2. Bound evaluation.** To demonstrate and evaluate the confidence bound as applied to data we compare it to the actual difference between the Nyström reconstruction error and the standard one, as well to the difference between the standard reconstruction error and the reconstruction error for Subset PCA. These quantities are generally not available when applying the Nyström method since they depend on the eigenvalues of the full kernel matrix, but we calculate them here for purposes of illustration.

Here we use a Nyström subset of size  $m = 50$ . We calculate the bound for PCA dimensions 1 through 10 and use a confidence level of 0.9 when calculating the bound. We run the experiments for multiple samples of the Nyström subset and plot the averages for the relevant quantities using 100 samples. The individual runs for different samples are run in parallel to leverage multi-core CPUs.

<sup>6</sup><https://aws.amazon.com/ec2/instance-types/>

<sup>7</sup><https://numpy.org/devdocs/reference/generated/numpy.linalg.eigh.html>



We plot the results of the experiments for the first four datasets in Table 2 and the kernels in Table 3 for different PCA dimensions below in Figures 1, 2 and 3. For the RBF and Cauchy kernels we set the bandwidth to  $\sigma = 1$  and for the polynomial kernel we use  $R = 1$  and  $d = 2$ . The RBF and Cauchy kernels are bounded by  $\sup_x k(x, x) = 1$  and we normalize the polynomial kernel according to Equation (4) before applying it in the experiments. Each plot contains

- (1) The values of the confidence bound (“Conf. bound”)
- (2) The difference between the Nyström PCA and standard errors  $R_n(\tilde{V}_d) - R_n(\hat{V}_d^n)$  (“Nyström diff.”)
- (3) The difference between the Subset PCA and standard errors  $R_n(\hat{V}_d^m) - R_n(\hat{V}_d^n)$  (“Subset diff.”)

Both the Nyström difference, the subset difference and the bound increase as the PCA dimension increases. The bound increases more rapidly as the PCA dimension increases from low values, but levels out for larger values as the tail eigenvalues decrease.

The bound seems fairly conservative for these datasets and these choices of hyperparameters. In real-life applications of the Nyström method the datasets are usually much larger, with the number of data points sometimes in the millions, and then the bound will be significantly smaller. The main purpose of the current experiments is rather to investigate differences between datasets and kernel functions and across PCA dimensions.

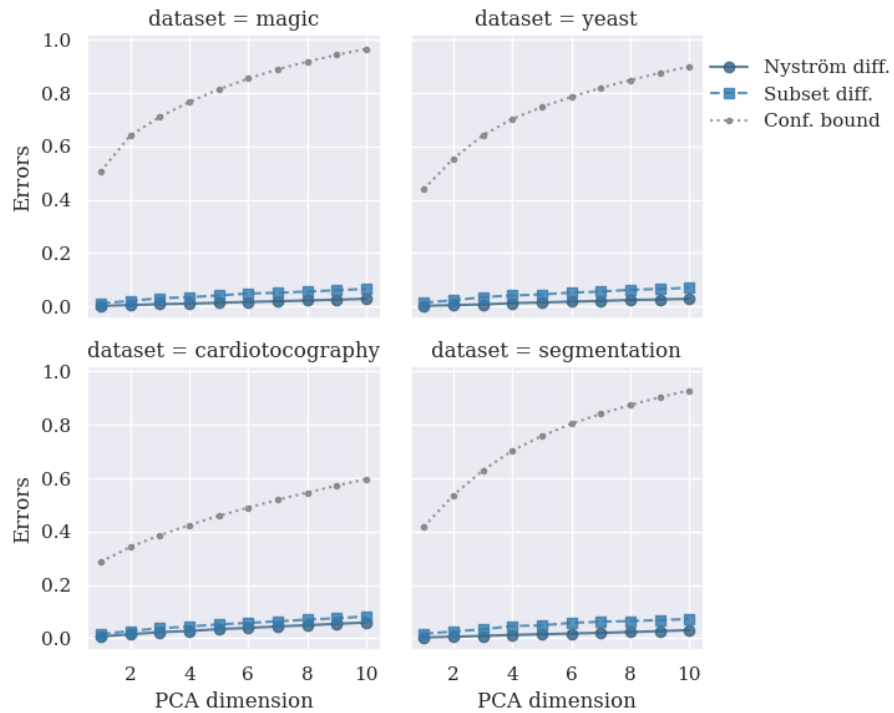


FIGURE 1. Error comparison with the RBF kernel

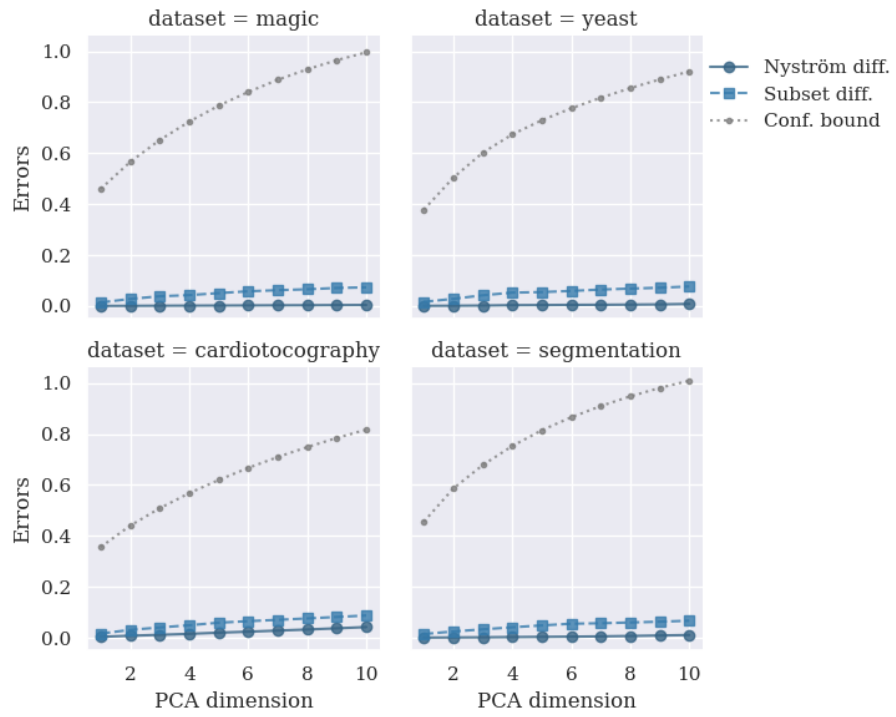


FIGURE 2. Error comparison with the polynomial kernel

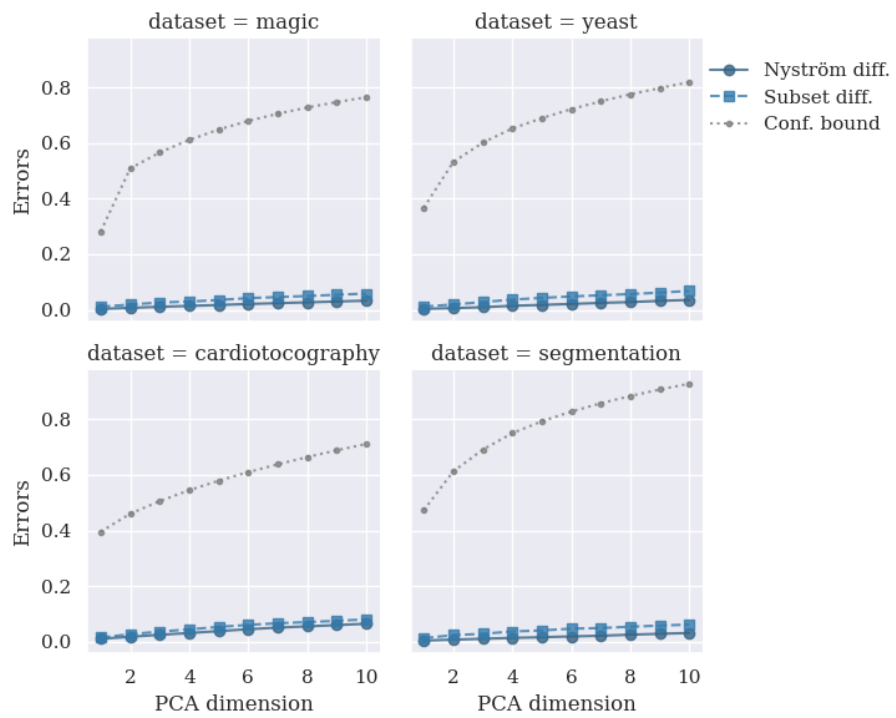


FIGURE 3. Error comparison with the Cauchy kernel

Running the bound evaluation experiments with the command-line tool can be accomplished with the following command

```
> nystrompca bound -m 50
```

In this section we have presented the experimental results for a few select parameter values. Other combinations can easily be tried after downloading the supplied software package.

## 8. APPLICATION: NYSTRÖM PRINCIPAL COMPONENT REGRESSION

As an application of Nyström kernel PCA we present kernel principal component regression with the Nyström method, or *Nyström kernel PCR*. The proposed method may be used for efficient regularized kernel regression, for example as an alternative to kernel ridge regression with the Nyström method [Liang and Rakhlin, 2020]. Its derivation demonstrates how the principal scores from Nyström kernel PCA may be used as new data points for supervised learning methods.

Principal component regression performs a regression of a target variable onto the principal scores from a subset of the principal components, instead of using the original data as regressor variables [Jolliffe, 2002, Chapter 8]. Principal component regression introduces regularization and ameliorates collinearity of the regressors, which leads to high variances for the coefficient estimates and may especially be a problem for kernel methods. It is known to correspond to the *errors-in-variables* regression model under certain circumstances, where the dependent and independent variables are assumed to contain measurement noise [Fuller, 1980].

We first derive standard kernel PCR, without the Nyström method. This derivation appears to be novel, as previous presentations of kernel principal component regression assumed data to have zero mean in feature space [Rosipal et al., 2000, 2001, Wibowo and Yamamoto, 2012].

Suppose thus that each data point  $x_i$  is paired with an observation of a target variable  $y_i$  in  $\mathbb{R}$  which we wish to predict using a new observation  $x^*$  of the independent variable. The regression model is

$$y = \alpha + S_d \beta + \varepsilon$$

with parameters  $\alpha$  and  $\beta = (\beta_1, \beta_2, \dots, \beta_d)^T$ , where  $y = (y_1, y_2, \dots, y_n)^T$ ,  $S_d$  are the principal scores from kernel PCA with respect to the top  $d$  principal components, and  $\varepsilon$  is a noise vector  $\varepsilon = (\varepsilon_1, \varepsilon_2, \dots, \varepsilon_n)^T$ , whose components we assume are generated from a zero-mean distribution with finite variance  $\text{Var}(\varepsilon_i)$ . The intercept is given by  $\alpha = \bar{y}$  since the scores have zero mean in each dimension. From Section 3 the principal scores are given by  $S_d = Q_d \Lambda_d^{1/2}$ , where  $Q_d \Lambda_d Q_d^T$  is the truncated eigendecomposition of  $K'$ . Since we assumed  $Z$  to be square-integrable we may apply least squares estimation to obtain that [Sen et al., 2010]

$$\hat{\beta} = (S_d^T S_d)^{-1} S_d^T y' = \Lambda_d^{-1/2} Q_d^T y' = S_d^{-1} y'$$

where  $y' = (y_1 - \bar{y}, y_2 - \bar{y}, \dots, y_n - \bar{y})^T$ . We recall that the principal scores of a new data point  $x^*$ , which we centre since we estimated the regression for zero-mean data points, are given by, with

respect to the top  $d$  principal components

$$w_d^* = \Lambda_d^{-1/2} Q_d^T \kappa'(x^*) = \Lambda_d^{-1/2} Q_d^T (\kappa(x^*) - \mathbf{1}_n \kappa(x^*) - K \mathbf{1}_n + \mathbf{1}_n K \mathbf{1}_n)$$

and so the prediction for a new data point becomes

$$\hat{y} = \bar{y} + \beta^T w_d^{*T} = \bar{y} + y'^T Q_d \Lambda_d^{-1} Q_d^T \kappa'(x^*)$$

For the Nyström method, the principal scores are given by  $W = K'_{nm} K'^{-1/2}_{mm} V = K'_{nm} U$ , and so the principal scores with respect to the top  $d$  principal components are given by  $W_d = K'_{nm} K'^{-1/2}_{mm} V_d = K'_{nm} U_d$  where  $V_d \tilde{\Lambda}_d V_d^T$  is the truncated eigendecomposition of  $\frac{1}{n} K'^{-1/2}_{mm} K'_{mn} K'_{nm} K'^{-1/2}_{mm}$  and  $U_d = K'^{-1/2}_{mm} V_d$ . The regression model then becomes

$$y = \alpha + W_d \beta + \varepsilon = \alpha + K'_{nm} U_d \beta + \varepsilon = \alpha + K'_{nm} K'^{-1/2}_{mm} V_d \beta + \varepsilon$$

The least squares parameter estimates are  $\hat{\alpha} = \bar{y}$  and

$$\begin{aligned} \hat{\beta} &= (W_d^T W_d)^{-1} W_d^T y' = \left( V_d^T K'^{-1/2}_{mm} K'_{mn} K'_{nm} K'^{-1/2}_{mm} V_d \right)^{-1} V_d^T K'^{-1/2}_{mm} K'_{mn} y' \\ &= \left( (V_d^T V \tilde{\Lambda} V^T V_d) \right)^{-1} V_d^T K'^{-1/2}_{mm} K'_{mn} y' = \tilde{\Lambda}_d^{-1} V_d^T K'^{-1/2}_{mm} K'_{mn} y' = \tilde{\Lambda}_d^{-1} U_d^T K'_{mn} y' \end{aligned}$$

And so the prediction becomes

$$\hat{y} = \bar{y} + y'^T K'_{nm} U_d \tilde{\Lambda}_d^{-1} U_d^T \tilde{\kappa}(x^*)$$

We implement kernel principal component regression with the Nyström method (Nyström KPCR) in computer experiments and compare it with Nyström kernel ridge regression (Nyström KRR) [Rudi et al., 2015], which is given by<sup>8</sup>

$$\begin{aligned} \hat{y} &= \bar{y} + \beta^T \kappa(x^*) \\ \hat{\beta} &= (K_{mn} K_{nm} + \gamma K_{mm})^{-1} K_{mn} y' \end{aligned}$$

where  $\gamma \geq 0$  is a regularization parameter, called the ridge parameter.

We use the `airfoil` dataset from the UCI machine learning repository [Dua and Graff, 2017], which describes aerodynamic tests of blades in a wind tunnel from NASA and contains 1503 data points and 6 attributes.

Again we normalize the attributes to have mean 0 and variance 1. Note that we must not normalize the entire dataset at once so as to not introduce look-ahead bias in the regression – when creating a prediction for a new data point we need to normalize this data point using the mean and variance from the training set.

<sup>8</sup>This is a slightly different specification than in Rudi et al. [2015], where we have demeaned the target variable and subsumed a factor  $n$  into the ridge parameter

For this experiment we use the radial basis functions kernel with parameter  $\sigma = 1$ . The source code for these experiments is available in the same package at <https://github.com/fredhallgren/nystrompca>. We estimate the regression on a training dataset with a random sample of 75 % of all data points, and evaluate the method on a test set with the remaining data points.

We first plot the  $R^2$  for the regression on the test set for different subset sizes  $m$ , ridge parameters  $\gamma$  and PCA dimensions  $d$  below in Figure 4. We do not specify a random seed for these plots and for each parameter combination a different subset is used.

For Nyström kernel PCR the regression accuracy improves as we increase the number of principal components used in the regression and as the size of the subset increases. For Nyström KRR the accuracy also improves with a larger subset, but the pattern is less clear as we change the regularization parameter.

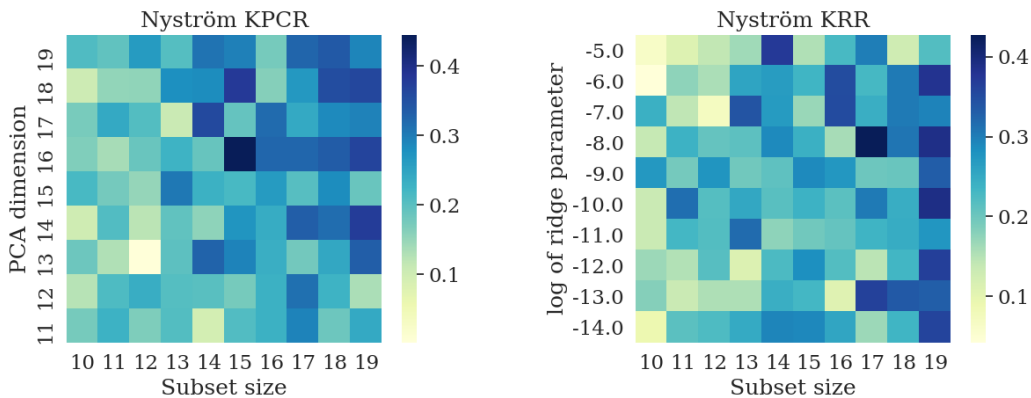


FIGURE 4. Heat maps with regression  $R^2$

To further elucidate the behaviour of the methods we also plot the actual target values versus the predicted ones on the test set for one instance of the parameters. Please see below Figure 5. Here we use  $m = 100$ ,  $d = 90$  and  $\gamma = 10^{-11}$ . The parameters  $d$  and  $\gamma$  were manually tuned. In this particular example Nyström KPCR obtained an  $R^2$  of 0.74 and Nyström KRR 0.72 with a seed of 1.

The scatter plots of the predictions versus the actual targets look as expected for an  $R^2$  of around 0.7. The predictions for the two methods look quite similar, but slightly different characteristics are exhibited by the plots due to the different regularization methodologies.

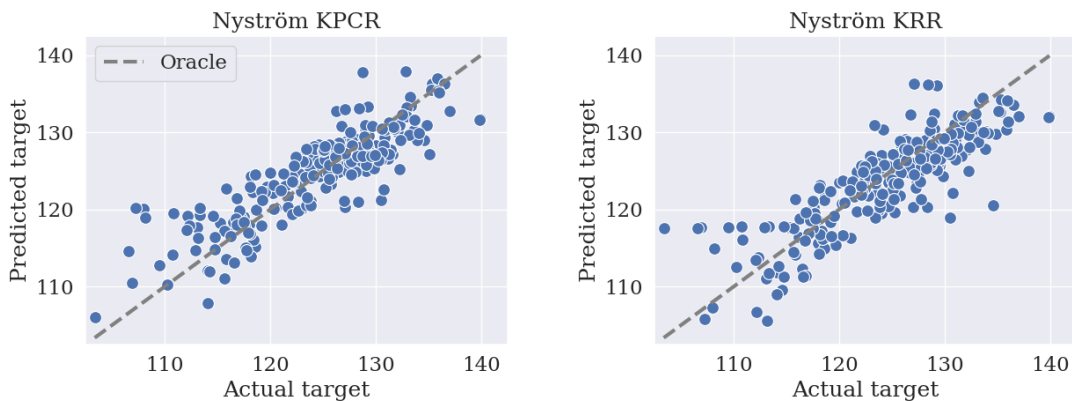


FIGURE 5. Scatter plot with regression predictions

These experiments can also be run with the command-line tool, using the below command

```
> nystrompca regression -m 100 -d 90
```

where the `-m` flag changes the size of the subset for the second set of plots, and `-d` selects the number of PCA dimensions for Nyström kernel PCR in the same plots. Please see the software documentation for a full list of available command-line options.

## 9. CONCLUSION

In this paper we have presented an efficient method for non-linear PCA by deriving the Nyström method for kernel PCA, providing the principal components, explained variance, principal scores and reconstruction error, without assuming that data has zero mean. The Nyström method has been shown in theory and practice to be effective for improving the scalability of kernel methods, but it had not yet been derived for kernel PCA in line with classical PCA. The method presented in this paper is one of the few options available to reduce the computational requirements for kernel PCA while maintaining good accuracy.

We also provided a finite-sample confidence bound on the empirical reconstruction error of the method, which allows us to measure its statistical accuracy for a specific dataset and which can be applied before the entire dataset has been observed. As a corollary to the confidence bound we presented a brief lemma by which a large number of previous results can be sharpened, and of potentially wide application in the future.

The principal scores from the method may be used instead of the original data matrix in any supervised learning method, in order for example to achieve regularization and denoising. We demonstrated this for linear regression by presenting Nyström kernel principal component regression. This derivation also led to a novel specification for standard kernel PCR, where the regressors are centred, as is the case for linear PCR.

We thank you for your interest in this work. We hope that what we have presented in this paper will be useful to the academic community and to industry practitioners, and that it may give ideas about future directions of research.

In addition to linear regression, there are many other methods based on PCA where kernel PCA with the Nyström method could be analyzed and explored, such as when PCA is applied in discriminant analysis, outlier detection or dictionary learning. The latter could be achieved for example along the lines of Golts and Elad [2016].

The approximate Nyström kernel matrix  $\tilde{K} = K_{nm}K_{mm}^{-1}K_{mn}$  may often be used as a drop-in replacement for the original kernel matrix to speed up kernel machines. However, for many methods, like kernel PCA, more work is needed for a complete treatment. There are still many kernel methods where application of the Nyström method is not necessarily trivial and has not been fully derived, including potentially kernel FDA or kernel PLS [Mika et al., 1999, Rosipal and Trejo, 2001]

Kernel PCA is closely related to functional PCA [Amini and Wainwright, 2012]. Functional PCA may also suffer from scalability issues if the individual functions are sampled at a large number of points. It's possible that there are settings where the Nyström method could be successfully applied to functional data analysis for improved computational efficiency.

As outlined in Section 2 on page 5, since the one existing bound for centred kernel PCA is more conservative than the available bounds for uncentred kernel PCA, there no statistical analysis of kernel PCA available in its full generality that is as precise as with an assumption of zero-mean data. Closing this gap would be a major achievement and in our view one of the most significant potential research contributions for kernel methods in the near term.

In this paper the data points in the Nyström subset are selected randomly from the full dataset to provide a subspace in which the PCA solution is sought, but the selection of these data points is not optimized in any way to best suit the problem at hand. In the Gaussian process literature, the so-called *inducing points*, which are analogous to the Nyström subset, are often selected to be optimal in some sense [Wild et al., 2021]. Other ways to select the Nyström subset for our method could also be explored, which may achieve some measure of optimality, or benefit from improved performance.

In an ideal world one would not randomly constrain some subspace in pursuit of improved computational performance, a crude proxy without doubt, but instead optimize with respect to the computational resources directly, where some measure of computational complexity is seamlessly included in the estimation procedure. Sadly, such theory does not yet exist. The theory of statistical estimation is conspicuously detached from that of computation, so given some utility of improved statistical accuracy, and some model of computation, there is no way to objectively, or even formally, trade accuracy against computational cost. Until such time, we are reduced to considering a plethora of different models, each with its own separate measures of accuracy and complexity, but with little recourse when attempting to choose between them.

## APPENDIX A. PROOFS

In this section we present the proofs of Propositions 1 and 2, and Theorems 1, 2 and 3.

First we state and prove a lemma that is used in the proof of Theorem 3. It shows that the bound of the difference between two positive operators is smaller than the sum of the bounds for the individual operators by a factor  $1/\sqrt{2}$ . For wider applicability we state this lemma for Hilbert spaces over both the real or complex numbers.

**Lemma 1** (Bound of positive operators). *Let  $L_1$  and  $L_2$  be positive operators in  $\text{HS}(\mathcal{H})$  over  $\mathbb{R}$  or  $\mathbb{C}$  with  $\|L_1\|_{\text{HS}(\mathcal{H})} \leq B$  and  $\|L_2\|_{\text{HS}(\mathcal{H})} \leq B$ . Then  $\|L_1 - L_2\|_{\text{HS}(\mathcal{H})} \leq \sqrt{2}B$ .*

*Proof.*  $\|L_1 - L_2\|_{\text{HS}(\mathcal{H})}^2 = \|L_1\|_{\text{HS}(\mathcal{H})}^2 + \|L_2\|_{\text{HS}(\mathcal{H})}^2 - 2\text{Re}\{\langle L_1, L_2 \rangle_{\text{HS}(\mathcal{H})}\} \leq 2B^2$  since  $\langle L_1, L_2 \rangle_{\text{HS}(\mathcal{H})}$  is real and positive. To see this, we note that  $\langle L_1, L_2 \rangle_{\text{HS}(\mathcal{H})} = \sum_{i=1}^{\infty} \langle L_1 e_i, L_2 e_i \rangle_{\mathcal{H}}$  for any basis  $\{e_i\}$  in  $\mathcal{H}$  and take  $\{e_i\}$  to be the eigenvectors of  $L_1$ , arbitrarily extended to a basis for the entire space if  $\text{Ker}(L_1) \neq \{0\}$ , obtaining, for each  $i$ , that  $\langle L_1 e_i, L_2 e_i \rangle_{\mathcal{H}} = \langle \lambda_i e_i, L_2 e_i \rangle_{\mathcal{H}} = \lambda_i \langle e_i, L_2 e_i \rangle_{\mathcal{H}} \in \mathbb{R}_+$  since  $L_2$  is positive and  $\lambda_i \in \mathbb{R}_+$ .  $\square$

**Proof of Theorem 1.** Standard principal component analysis finds the perpendicular intersecting lines in  $\mathbb{R}^d$  along which the variance of the data is successively maximized [Pearson, 1901]. These lines are affine subspaces of  $\mathbb{R}^d$  which are orthogonal with respect to the associated vector space. To derive kernel PCA with the Nyström method we apply PCA in the span of the subset of data points  $\mathcal{H}_S$ , i.e. finding the orthogonal one-dimensional affine subspaces of  $\mathcal{H}_S$  where the projected data has maximum variance. These are on the form

$$\phi_0 + \langle f_j \rangle = \phi_0 + \{ a f_j \mid a \in \mathbb{R} \}$$

where  $\phi_0 \in \mathcal{H}_S$  is the translation of the vector space  $\langle f_j \rangle$ , and the  $f_j \in \mathcal{H}_S$ , taken to have norm one, are the principal components. It is known from standard PCA and kernel PCA that the translation vector is given by the mean of the data points, which in our case is the mean of the data points projected onto  $\mathcal{H}_S$ . Using  $P_{\mathcal{H}_S} = G_m^* (G_m G_m^*)^{-1} G_m = m \cdot G_m^* K_{mm}^{-1} G_m$ , where  $G_m$  is the sampling operator [Rudi et al., 2015], we obtain

$$\begin{aligned} \phi_0 &= \frac{1}{n} \sum_{r=1}^n P_{\mathcal{H}_S} \phi(x_r) = \frac{1}{n} \sum_{r=1}^n m \cdot G_m^* K_{mm}^{-1} G_m \phi(x_r) \\ &= \frac{1}{n} \sum_{r=1}^n \sqrt{m} \cdot G_m^* K_{mm}^{-1} \kappa_m(x_r) = \frac{1}{n} K_{nm} K_{mm}^{-1} \kappa_m(x) \end{aligned}$$

Any element  $\phi \in \mathcal{H}_S$  can be written as  $\phi = \phi_0 + \sum_{k=1}^m a_k \cdot (\phi(x_k) - \phi_0)$  for some coefficients  $a_1, a_2, \dots, a_m$  and so the principal components are on the form  $f_j = \sum_{k=1}^m u_{j,k} (\phi(x_k) - \phi_0)$  with coefficients  $u_{j,1}, u_{j,2}, \dots, u_{j,m}$ . The affine projection of a data point  $\phi(x)$  onto  $\phi_0 + \langle f_j \rangle$  is then

$$P_{\phi_0 + \langle f_j \rangle} \phi(x) = \phi_0 + \langle \phi(x) - \phi_0, f_j \rangle_{\mathcal{H}} f_j$$



The variance of the full dataset along  $\phi_0 + \langle f_j \rangle$  then becomes

$$\begin{aligned}
\text{Var}_{f_j}(\{\phi(x_i)\}_{i=1}^n) &= \frac{1}{n} \sum_{i=1}^n \left( \phi_0 + \langle \phi(x_i) - \phi_0, f_j \rangle_{\mathcal{H}} - \frac{1}{n} \sum_{\ell=1}^n (\phi_0 + \langle \phi(x_\ell) - \phi_0, f_j \rangle_{\mathcal{H}}) \right)^2 \\
&= \frac{1}{n} \sum_{i=1}^n \left\langle \phi(x_i) - \frac{1}{n} \sum_{\ell=1}^n \phi(x_\ell), \sum_{k=1}^m u_{j,k} (\phi(x_k) - \phi_0) \right\rangle_{\mathcal{H}}^2 \\
&= \frac{1}{n} \sum_{i=1}^n \left( \sum_{k=1}^m u_{j,k} \left( k_{k,i} - \frac{1}{n} \sum_{\ell=1}^n k_{k,\ell} - \langle \phi(x_i), \phi_0 \rangle_{\mathcal{H}} + \frac{1}{n} \sum_{\ell=1}^n \langle \phi(x_\ell), \phi_0 \rangle_{\mathcal{H}} \right) \right)^2
\end{aligned}$$

Using

$$\begin{aligned}
\langle \phi(x_i), P_{\mathcal{H}_S} \phi(x_r) \rangle_{\mathcal{H}} &= \langle \phi(x_i), m \cdot G_m^* K_{mm}^{-1} G_m \phi(x_r) \rangle_{\mathcal{H}} \\
&= \sqrt{m} \langle \phi(x_i), G_m^* K_{mm}^{-1} \kappa_m(x_r) \rangle_{\mathcal{H}} = \kappa_m(x_i)^T K_{mm}^{-1} \kappa_m(x_r)
\end{aligned}$$

where  $\kappa_m(x) = (k(x_1, x), k(x_2, x), \dots, k(x_m, x))^T$ , and setting  $\kappa_m(x_a) = \kappa_{m,a}$ , we obtain

$$\begin{aligned}
&\frac{1}{n} \sum_{i=1}^n \left( \sum_{k=1}^m u_{j,k} \left( k_{k,i} - \frac{1}{n} \sum_{\ell=1}^n k_{k,\ell} - \langle \phi(x_i), \phi_0 \rangle_{\mathcal{H}} + \frac{1}{n} \sum_{\ell=1}^n \langle \phi(x_\ell), \phi_0 \rangle_{\mathcal{H}} \right) \right)^2 \\
&= \frac{1}{n} \sum_{i=1}^n \left( \sum_{k=1}^m u_{j,k} \left( k_{k,i} - \frac{1}{n} \sum_{\ell=1}^n k_{k,\ell} - \frac{1}{n} \sum_{r=1}^n \kappa_{m,i}^T K_{mm}^{-1} \kappa_{m,r} + \frac{1}{n^2} \sum_{\ell=1}^n \sum_{r=1}^n \kappa_{m,\ell}^T K_{mm}^{-1} \kappa_{m,r} \right) \right)^2 \\
&= \frac{1}{n} u_j^T K'_{mn} K'_{nm} u_j
\end{aligned}$$

where  $K'_{mn} = K_{mn} - K_{mn} \mathbf{1}_n - \mathbf{1}_n^{m,n} K_{nm} K_{mm}^{-1} K_{mn} + \mathbf{1}_n^{m,n} K_{nm} K_{m,m}^{-1} K_{mn} \mathbf{1}_n$ , with  $\mathbf{1}_n^{m,n}$  an  $m \times n$  matrix with each element equal to  $\frac{1}{n}$ , and  $K'_{nm} = K'_{mn}{}^T$ .

The principal components are then given by the orthonormal vectors  $f_j = \sum_{k=1}^m u_{j,k} (\phi(x_k) - \phi_0)$ ,  $j = 1, 2, \dots, m$  that successively maximize the variance. The inner product between two principal components is

$$\begin{aligned}
\langle f_j, f_p \rangle_{\mathcal{H}} &= \left\langle \sum_{k=1}^m u_{j,k} (\phi(x_k) - \phi_0), \sum_{q=1}^m u_{p,q} (\phi(x_q) - \phi_0) \right\rangle_{\mathcal{H}} \\
&= \sum_{\substack{k=1 \\ q=1}}^m u_{j,k} u_{p,q} \left( k_{k,q} - \frac{1}{n} \sum_{r=1}^n \kappa_{m,r} K_{mm}^{-1} \kappa_{m,k} - \frac{1}{n} \sum_{\ell=1}^n \kappa_{m,\ell} K_{mm}^{-1} \kappa_{m,q} + \frac{1}{n^2} \sum_{\substack{r=1 \\ \ell=1}}^n \kappa_{m,r} K_{mm}^{-1} \kappa_{m,\ell} \right) \\
&= u_j^T K'_{mm} u_p
\end{aligned}$$

where  $K'_{mm} = K_{mm} - \mathbb{1}_n^{m,n} K_{nm} - K_{mn} \mathbb{1}_n^{n,m} + \mathbb{1}_n^{m,n} K_{nm} K_{mm}^{-1} K_{mn} \mathbb{1}_n^{m,n}$ . Maximizing the variance therefore becomes a generalized eigenvalue problem. We have

$$\langle f_j, f_p \rangle_{\mathcal{H}} = u_j^T K'_{mm} u_p = \left( K_{mm}'^{1/2} u_j \right)^T \left( K_{mm}'^{1/2} u_p \right) := v_j^T v_p$$

where  $K_{mm}'^{1/2}$  is the unique positive semi-definite square root of  $K'_{mm}$  given by  $m \cdot U^m \Lambda^{m1/2} U^{mT}$ , where  $U^m \Lambda^m U^{mT}$  is the eigendecomposition of  $\frac{1}{m} K'_{mm}$ . Therefore the variance can be written

$$\frac{1}{n} v_j^T K_{mm}'^{-1/2} K'_{mn} K'_{nm} K_{mm}'^{-1/2} v_j = \left\langle v_j, \frac{1}{n} K_{mm}'^{-1/2} K'_{mn} K'_{nm} K_{mm}'^{-1/2} v_j \right\rangle_{\mathbb{R}^m}$$

Then by the Courant-Fischer-Weyl theorem [Bhatia, 1997, Corollary III.1.2] the maximum values over successively orthonormal vectors  $v_j$  are given by the eigenvalues of  $\frac{1}{n} K_{mm}'^{-1/2} K'_{mn} K'_{nm} K_{mm}'^{-1/2}$ , and they occur at its eigenvectors. These eigenvectors will be unique (up to a sign), since all data points are different by assumption.

The principal components are then given by

$$\tilde{\phi}_j = \sum_{k=1}^m u_{j,k} (\phi(x_k) - \phi_0) \quad j = 1, 2, \dots, m$$

where  $u_j = K_{mm}'^{-1/2} v_j$ , and the affine subspaces with maximum variances are  $\{\phi_0 + t \tilde{\phi}_j \mid t \in \mathbb{R}\}$ ,  $j = 1, 2, \dots, m$ .

The principal score of a centred data point  $i$  with respect to the principal component  $j$  is given by

$$\begin{aligned}
w_{j,i} &= \left\langle \phi(x_i) - \frac{1}{n} \sum_{\ell=1}^n \phi(x_\ell), \sum_{k=1}^m u_{j,k} (\phi(x_k) - \phi_0) \right\rangle_{\mathcal{H}} \\
&= \sum_{k=1}^m u_{j,k} \left( k_{k,i} - \frac{1}{n} \sum_{\ell=1}^n k_{k,\ell} - \frac{1}{n} \sum_{r=1}^n \kappa_{m,i}^T K_{mm}^{-1} \kappa_{m,r} + \frac{1}{n^2} \sum_{\substack{\ell=1 \\ r=1}}^n \kappa_{m,\ell}^T K_{mm}^{-1} \kappa_{m,r} \right)
\end{aligned}$$

for  $j = 1, 2, \dots, n$ . Or in matrix format

$$(w_{i,j}) = W = K'_{nm} U$$

where  $U = K'^{-1/2}_{mm} V$  and  $\frac{1}{n} K'^{-1/2}_{mm} K'_{mn} K'_{nm} K'^{-1/2}_{mm} = V \tilde{\Lambda} V^T$ , and so  $W = K'_{nm} K'^{-1/2}_{mm} V$ .

The scores of a *new* data point  $x^*$  which is centred in feature space, i.e. the coordinates of  $\phi(x^*) - \frac{1}{n} \sum_{\ell=1}^n \phi(x_\ell)$  in terms of the principal components, are given by

$$\begin{aligned} w_j^* &= \left\langle \phi(x^*) - \frac{1}{n} \sum_{\ell=1}^n \phi(x_\ell), \sum_{k=1}^m u_{j,k} (\phi(x_k) - \phi_0) \right\rangle_{\mathcal{H}} \\ &= \sum_{k=1}^m u_{j,k} \left( k(x_k, x^*) - \frac{1}{n} \sum_{\ell=1}^n k_{k,\ell} - \frac{1}{n} \sum_{r=1}^n \kappa_{m,r}^T K_{mm}^{-1} \kappa_m(x^*) + \frac{1}{n^2} \sum_{\ell=1}^n \kappa_{m,r}^T K_{mm}^{-1} \kappa_{m,\ell} \right) \end{aligned}$$

or in matrix format

$$w^* = U^T \left( \kappa_m(x^*) - K_{mn} \mathbf{1}_n - \mathbf{1}_n^{m,n} K_{nm} K_{mm}^{-1} \kappa_m(x^*) + \mathbf{1}_n^{m,n} K_{nm} K_{mm}^{-1} K_{mn} \mathbf{1}_n \right) := U^T \tilde{\kappa}(x^*)$$

where  $\mathbf{1}_n$  is a length- $n$  column vector given by  $\mathbf{1}_n = (\frac{1}{n}, \frac{1}{n}, \dots, \frac{1}{n})^T$ .

□

**Proof of Theorem 2.** The projection of a data point  $\phi(x_i)$  onto a principal component is given by

$$\begin{aligned} P_{\hat{\phi}_j^{m,n}} \phi(x_i) &= \frac{1}{\sqrt{m \hat{\lambda}_j^m}} \sum_{k=1}^m u_{j,k}^m \langle \phi(x_i), \phi(x_k) - \phi_0 \rangle_{\mathcal{H}} \hat{\phi}_j^{m,n} \\ &= \frac{1}{\sqrt{m \hat{\lambda}_j^m}} \sum_{k=1}^m u_{j,k}^m \left( k(x_k, x_i) - \frac{1}{n} K_{nm} K_{mm}^{-1} \kappa_m(x_i) \right) \hat{\phi}_j^{m,n} \end{aligned}$$

where  $(\hat{\lambda}_j^m, u_j^m)$  is the  $j$ th eigenpair of  $\frac{1}{m} K'_{mm}$  and  $u_{j,k}^m$  is the  $k$ th element of  $u_j^m$  [Shawe-Taylor et al., 2005].

The projection of a centred data point  $\phi'(x_i)$  is then, similarly to Theorem 1, with  $k_{a,b} := k(x_a, x_b)$  and  $\kappa_m(x_a) = \kappa_{m,a}$

$$P_{\hat{\phi}_j^{m,n}} \phi'(x_i) = \frac{1}{\sqrt{m \hat{\lambda}_j^m}} \sum_{k=1}^m u_{j,k}^m \left\langle \phi(x_i) - \frac{1}{n} \sum_{\ell=1}^n \phi(x_\ell), \phi(x_k) - \phi_0 \right\rangle_{\mathcal{H}} \hat{\phi}_j^{m,n} =$$

$$= \frac{1}{\sqrt{m\hat{\lambda}_j^m}} \sum_{k=1}^m u_{j,k}^m \left( k_{k,i} - \frac{1}{n} \sum_{\ell=1}^n k_{k,\ell} - \frac{1}{n} \sum_{r=1}^n \kappa_{m,i}^T K_{mm}^{-1} \kappa_{m,r} + \frac{1}{n^2} \sum_{\ell=1}^n \kappa_{m,\ell}^T K_{mm}^{-1} \kappa_{m,r} \right) \hat{\phi}_j^{m,n}$$

Taking the norm and summing over  $\phi(x_1), \phi(x_2), \dots, \phi(x_n)$  we obtain

$$\begin{aligned} & \frac{1}{n} \sum_{i=1}^n \|P_{\hat{\phi}_j^{m,n}} \phi'(x_i)\|_{\mathcal{H}}^2 = \\ & \frac{1}{n \cdot m \hat{\lambda}_j^m} \sum_{i=1}^n \left( \sum_{k=1}^m u_{j,k}^m \left( k_{k,i} - \frac{1}{n} \sum_{\ell=1}^n k_{k,\ell} - \frac{1}{n} \sum_{r=1}^n \kappa_{m,i}^T K_{mm}^{-1} \kappa_{m,r} + \frac{1}{n^2} \sum_{\ell=1}^n \kappa_{m,\ell}^T K_{mm}^{-1} \kappa_{m,r} \right) \right)^2 \\ & = \frac{1}{n \cdot m \hat{\lambda}_j^m} u_j^{mT} K'_{mn} K'_{nm} u_j^m =: \hat{\lambda}_j^{m,n} \end{aligned}$$

For the reconstruction error we have

$$\begin{aligned} R_n(\hat{V}_d^m) &= \frac{1}{n} \sum_{i=1}^n \|\phi'(x_i) - P_{\hat{V}_d^m} \phi'(x_i)\|_{\mathcal{H}}^2 = \frac{1}{n} \sum_{i=1}^n \|\phi'(x_i)\|_{\mathcal{H}} - \frac{1}{n} \sum_{i=1}^n \|P_{\hat{V}_d^m} \phi'(x_i)\|_{\mathcal{H}} \\ &= \frac{1}{n} \text{Tr}(K') - \frac{1}{n} \sum_{i=1}^n \|P_{\hat{V}_d^m} \phi'(x_i)\|_{\mathcal{H}} \end{aligned}$$

And so similarly to above, the second term becomes

$$\begin{aligned} & \frac{1}{n} \sum_{i=1}^n \|P_{\hat{V}_d^m} \phi'(x_i)\|_{\mathcal{H}}^2 = \\ & \frac{1}{n} \sum_{i=1}^n \left( \sum_{j=1}^d \frac{1}{\sqrt{m\hat{\lambda}_j^m}} \sum_{k=1}^m u_{j,k}^m \left( k_{k,i} - \frac{1}{n} \sum_{\ell=1}^n k_{k,\ell} - \frac{1}{n} \sum_{r=1}^n \kappa_{m,i}^T K_{mm}^{-1} \kappa_{m,r} + \frac{1}{n^2} \sum_{\ell=1}^n \kappa_{m,\ell}^T K_{mm}^{-1} \kappa_{m,r} \right) \right)^2 \\ & = \frac{1}{n \cdot m} \text{Tr}(K'_{nm} U_d^m \Lambda_d^{m-1} U_d^{mT} K'_{mn}) \end{aligned}$$

with  $U_d^m \Lambda_d^m U_d^{mT}$  the truncated eigendecomposition of  $\frac{1}{m} K'_{mm}$ .

□

**Proof of Proposition 1.** Since  $\hat{V}_d^m \subset \mathcal{H}_S$  for any  $d$  and by Theorem 1

$$\begin{aligned}\tilde{\lambda}_{<d} &= \max_{\substack{\dim(V)=d \\ a+V \subset \mathcal{H}_S}} \frac{1}{n} \sum_{i=1}^n \|P_{a+V} z_i\|_{\mathcal{H}}^2 = \max_{\substack{\dim(V)=d \\ V \subset \mathcal{H}_S \\ a \in \mathcal{H}_S}} \frac{1}{n} \sum_{i=1}^n \|P_V(z_i - a)\|_{\mathcal{H}}^2 \\ &\geq \frac{1}{n} \sum_{i=1}^n \|P_{\hat{V}_d^m}(z_i - \phi_0)\|_{\mathcal{H}}^2 = \hat{\lambda}_{<d}^{m,n}\end{aligned}$$

The case  $d = m$  follows since both  $\langle \{\hat{\phi}_j^{m,n}\}_{j=1}^m \rangle$  and  $\langle \{\tilde{\phi}_j\}_{j=1}^m \rangle$  capture the full variance of the data in  $\mathcal{H}_S$ . □

**Proof of Proposition 2.** By the previous proposition we have  $\tilde{V}_m = \hat{V}_m^m$  for a fixed  $\omega$  and so we will have  $\tilde{V}_m \stackrel{d}{=} \hat{V}_m^m$  if  $\{X_{i_1}, X_{i_2}, \dots, X_{i_m}\} \stackrel{d}{=} \{X_1, X_2, \dots, X_m\}$ , where  $S = \{i_1, i_2, \dots, i_m\}$  are the indices for the subsampled data points. By the law of total probability

$$\begin{aligned}\mathbb{P}(\{X_{i_1} \leq a_1, X_{i_2} \leq a_2, \dots, X_{i_m} \leq a_m\}) \\ &= \sum_S \mathbb{P}(\{X_{i_1} \leq a_1, X_{i_2} \leq a_2, \dots, X_{i_m} \leq a_m\} | S) \mathbb{P}(S) \\ &= \sum_S \mathbb{P}(\{X_1 \leq a_1, X_2 \leq a_2, \dots, X_m \leq a_m\} | S) \mathbb{P}(S)\end{aligned}$$

since conditional on the sample  $S$ , we have  $m$  random variables generated according to  $\mathbb{P}_X$ , which we can take to be  $X_1, X_2, \dots, X_m$ . If the subsampling is independent of the data then

$$\begin{aligned}\sum_S \mathbb{P}(\{X_1 \leq a_1, X_2 \leq a_2, \dots, X_m \leq a_m\} | S) \mathbb{P}(S) \\ = \mathbb{P}(\{X_1 \leq a_1, X_2 \leq a_2, \dots, X_m \leq a_m\}) \sum_S \mathbb{P}(S) = \prod_{k=1}^m \mathbb{P}(\{X_k \leq a_k\})\end{aligned}$$

so the subsampled data points are generated i.i.d. from  $\mathbb{P}_X$ . We can therefore conclude that  $\tilde{V}_m \stackrel{d}{=} \hat{V}_m^m$ . Since  $Z$  has the same distribution  $\mathbb{P}_Z$  regardless of the subspace and since  $\tilde{V}_m \stackrel{d}{=} \hat{V}_m^m$  we have  $P_{\tilde{V}_m} Z' \stackrel{d}{=} P_{\hat{V}_m^m} Z'$  and can conclude that, since  $Z$  is square-integrable

$$\mathbb{E}[\|P_{\tilde{V}_m} Z' - Z'\|_{\mathcal{H}}^2] = \mathbb{E}[\|P_{\hat{V}_m^m} Z' - Z'\|_{\mathcal{H}}^2]$$

and so  $R(\tilde{V}_m) = R(\hat{V}_m^m)$  when  $p(S | x_1, x_2, \dots, x_n) = p(S)$ . □

**Proof of Theorem 3.** The difference in errors can be rewritten through

$$\begin{aligned}
R_n(\tilde{V}_d) - R_n(\hat{V}_d) &= \min_{\substack{\dim(V)=d \\ V \subset \mathcal{H}_S}} \frac{1}{n} \sum_{i=1}^n \|P_V z_i - z_i\|_{\mathcal{H}}^2 - \min_{\dim(V)=d} \frac{1}{n} \sum_{i=1}^n \|P_V z_i - z_i\|_{\mathcal{H}}^2 \\
&= \max_{\dim(V)=d} \frac{1}{n} \sum_{i=1}^n \|P_V z_i\|_{\mathcal{H}}^2 - \max_{\substack{\dim(V)=d \\ V \subset \mathcal{H}_S}} \frac{1}{n} \sum_{i=1}^n \|P_V z_i\|_{\mathcal{H}}^2 \\
&= \frac{1}{n} \sum_{i=1}^n \|(P_{\mathcal{H}_S} + P_{\mathcal{H}_S^\perp})P_{\hat{V}_d} z_i\|_{\mathcal{H}}^2 - \max_{\substack{\dim(V)=d \\ V \subset \mathcal{H}_S}} \frac{1}{n} \sum_{i=1}^n \|P_V z_i\|_{\mathcal{H}}^2 \\
&\leq \frac{1}{n} \sum_{i=1}^n \|P_{\mathcal{H}_S} P_{\hat{V}_d} z_i\|_{\mathcal{H}}^2 + \frac{1}{n} \sum_{i=1}^n \|P_{\mathcal{H}_S^\perp} P_{\hat{V}_d} z_i\|_{\mathcal{H}}^2 - \max_{\substack{\dim(V)=d \\ V \subset \mathcal{H}_S}} \frac{1}{n} \sum_{i=1}^n \|P_V z_i\|_{\mathcal{H}}^2 \\
&\leq \frac{1}{n} \sum_{i=1}^n \|P_{\mathcal{H}_S^\perp} P_{\hat{V}_d} z_i\|_{\mathcal{H}}^2
\end{aligned}$$

Expanding the projection operator  $P_{\hat{V}_d}$  we obtain

$$\begin{aligned}
\frac{1}{n} \sum_{i=1}^n \|P_{\mathcal{H}_S^\perp} P_{\hat{V}_d} z_i\|_{\mathcal{H}_S}^2 &= \frac{1}{n} \sum_{i=1}^n \left\| P_{\mathcal{H}_S^\perp} \sum_{j=1}^d \langle z_i, \hat{\phi}_j^n \rangle_{\mathcal{H}} \hat{\phi}_j^n \right\|_{\mathcal{H}}^2 \\
&= \frac{1}{n} \sum_{i=1}^n \left\| \sum_{j=1}^d \langle z_i, \hat{\phi}_j^n \rangle_{\mathcal{H}} P_{\mathcal{H}_S^\perp} \hat{\phi}_j^n \right\|_{\mathcal{H}}^2 \leq \frac{1}{n} \sum_{i=1}^n \sum_{j=1}^d \left\| \langle z_i, \hat{\phi}_j^n \rangle_{\mathcal{H}} P_{\mathcal{H}_S^\perp} \hat{\phi}_j^n \right\|_{\mathcal{H}}^2
\end{aligned}$$

The last inequality is fairly sharp. It becomes an equality without the projection  $P_{\mathcal{H}_S^\perp}$ , and the further the projection is from the identity, the smaller the norm of  $P_{\mathcal{H}_S^\perp} \hat{\phi}_j^n$ . Now we have

$$\begin{aligned}
\frac{1}{n} \sum_{i=1}^n \sum_{j=1}^d \left\| \langle z_i, \hat{\phi}_j^n \rangle_{\mathcal{H}} P_{\mathcal{H}_S^\perp} \hat{\phi}_j^n \right\|_{\mathcal{H}}^2 &= \frac{1}{n} \sum_{i=1}^n \sum_{j=1}^d |\langle z_i, \hat{\phi}_j^n \rangle_{\mathcal{H}}|^2 \left\| P_{\mathcal{H}_S^\perp} \hat{\phi}_j^n \right\|_{\mathcal{H}}^2 \\
&= \sum_{j=1}^d \left( \frac{1}{n} \sum_{i=1}^n |\langle z_i, \hat{\phi}_j^n \rangle_{\mathcal{H}}|^2 \right) \left\| P_{\mathcal{H}_S^\perp} \hat{\phi}_j^n \right\|_{\mathcal{H}}^2 = \sum_{j=1}^d \hat{\lambda}_j^n \left\| P_{\mathcal{H}_S^\perp} \hat{\phi}_j^n \right\|_{\mathcal{H}}^2
\end{aligned}$$

Expanding the other projection operator we get

$$\sum_{j=1}^d \hat{\lambda}_j^n \left\| P_{\mathcal{H}_S^\perp} \hat{\phi}_j^n \right\|_{\mathcal{H}}^2 = \sum_{j=1}^d \hat{\lambda}_j^n \left\| \hat{\phi}_j^n - P_{\mathcal{H}_S} \hat{\phi}_j^n \right\|_{\mathcal{H}}^2 = \sum_{j=1}^d \hat{\lambda}_j^n \left\| \hat{\phi}_j^n - \sum_{k=1}^m \langle \hat{\phi}_j^n, \hat{\phi}_k^m \rangle_{\mathcal{H}} \hat{\phi}_k^m \right\|_{\mathcal{H}}^2$$

We have, for any  $j$

$$\left\| \hat{\phi}_j^n - \sum_{k=1}^m \langle \hat{\phi}_j^n, \hat{\phi}_k^m \rangle_{\mathcal{H}} \hat{\phi}_k^m \right\|_{\mathcal{H}}^2 \leq \left\| \hat{\phi}_j^n - \langle \hat{\phi}_j^n, \hat{\phi}_j^m \rangle_{\mathcal{H}} \hat{\phi}_j^m \right\|_{\mathcal{H}}^2 = 1 - \langle \hat{\phi}_j^n, \hat{\phi}_j^m \rangle_{\mathcal{H}}^2 = \sin^2 \theta_j$$

Then by the Davis-Kahan theorem [Yu et al., 2015, Corollary 1] (also see Davis and Kahan [1970]), defining  $\hat{\lambda}_0^m := +\infty$  and  $\hat{\lambda}_{m+1}^m := -\infty$

$$\sin \theta_j \leq \frac{2\|C_n - C_m\|_{\text{HS}(\mathcal{H})}}{\min\{\hat{\lambda}_{j-1}^m - \hat{\lambda}_j^m, \hat{\lambda}_j^m - \hat{\lambda}_{j+1}^m\}} \wedge 1 =: \sqrt{D_j}$$

Then by Lidskii's inequality [Kato, 2013, Chapter 3, Theorem 6.11]

$$\sum_{j=1}^d \hat{\lambda}_j^n \cdot D_j = \sum_{j=1}^d \hat{\lambda}_j^m \cdot D_j + \sum_{j=1}^d (\hat{\lambda}_j^n - \hat{\lambda}_j^m) \cdot D_j \leq \sum_{j=1}^d \hat{\lambda}_j^m \cdot D_j + \|C_n - C_m\|_{\text{HS}(\mathcal{H})} \max_{1 \leq k \leq d} D_k$$

Now the only unknown and random quantity is  $\|C_n - C_m\|_{\text{HS}(\mathcal{H})}$ . It depends both on the unobserved data points  $z_{m+1}, z_{m+2}, \dots, z_n$  and the observed ones  $z_1, z_2, \dots, z_m$ . We split these up into two terms

$$\begin{aligned} \|C_n - C_m\|_{\text{HS}(\mathcal{H})} &= \left\| \frac{1}{n} \sum_{i=1}^n z_i \otimes z_i - \frac{1}{m} \sum_{r=1}^m z_r \otimes z_r \right\|_{\mathcal{H} \otimes \mathcal{H}} \\ &= \left\| \frac{1}{n} \sum_{i=m+1}^n z_i \otimes z_i - \frac{n-m}{nm} \sum_{r=1}^m z_r \otimes z_r \right\|_{\mathcal{H} \otimes \mathcal{H}} \\ &= \frac{n-m}{n} \left\| \frac{1}{n-m} \sum_{i=m+1}^n z_i \otimes z_i - \frac{1}{m} \sum_{r=1}^m z_r \otimes z_r \right\|_{\mathcal{H} \otimes \mathcal{H}} \\ &= \frac{n-m}{n} \|C_{n-m} - C_m\|_{\text{HS}(\mathcal{H})} \end{aligned}$$

If we let  $Y_i = z_i \otimes z_i - C_m$ , then  $\frac{1}{n-m} \sum_{i=m+1}^n Y_i = C_{n-m} - C_m$  and the random variables  $Y_i$  have zero expectation with respect to  $C_{n-m}$  across hypothesized repeated realizations of  $C_m$ , and they are bounded by  $\sqrt{2}B := \sqrt{2} \sup_x k(x, x)$  by Lemma 1 since both  $z_i \otimes z_i$  and  $C_m$  are positive. Then by Hoeffding's inequality in Banach spaces [Pinelis, 1994, Theorem 3.5], we have that with probability at least  $1 - 2e^{-\delta}$ , over infinite repetitions of the experiment yielding  $C_m$

$$\frac{n-m}{n} \left\| \frac{1}{n-m} \sum_{i=m+1}^n z_i \otimes z_i - \frac{1}{m} \sum_{r=1}^m z_r \otimes z_r \right\|_{\mathcal{H} \otimes \mathcal{H}} \leq \frac{n-m}{n} \frac{2B\sqrt{\delta}}{\sqrt{n-m}}$$

We recall that the eigenvalues of the empirical covariance operator equal the eigenvalues of the kernel matrix  $\frac{1}{m} K_{mm}$ , which completes the proof. □

## REFERENCES

- M. Aharon, M. Elad, and A. Bruckstein. K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Transactions on Signal Processing*, 54(11):4311–4322, 2006.  
<https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=1710377>.
- A. A. Amini and M. J. Wainwright. Sampled forms of functional PCA in reproducing kernel Hilbert spaces. *The Annals of Statistics*, 40(5):2483–2510, 2012.  
<https://projecteuclid.org/journals/annals-of-statistics/volume-40/issue-5/Sampled-forms-of-functional-PCA-in-reproducing-kernel-Hilbert-spaces/10.1214/12-AOS1033.pdf>.
- F. R. Bach and M. I. Jordan. Kernel independent component analysis. *The Journal of Machine Learning Research*, 3(Jul):1–48, 2002.  
<http://www.jmlr.org/papers/volume3/bach02a/bach02a.pdf>.
- M. F. Balcan, Y. Liang, L. Song, D. Woodruff, and B. Xie. Communication efficient distributed kernel principal component analysis. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 725–734, 2016.  
<https://dl.acm.org/doi/pdf/10.1145/2939672.2939796>.
- S. Banach. Théorie des opérations linéaires. *Monografie Matematyczne*, 1932.  
[http://kielich.amu.edu.pl/Stefan\\_Banach/pdf/teoria-operacji-fr/banach-teorie-des-operations-lineaires.pdf](http://kielich.amu.edu.pl/Stefan_Banach/pdf/teoria-operacji-fr/banach-teorie-des-operations-lineaires.pdf).
- P. L. Bartlett and S. Mendelson. Rademacher and Gaussian complexities: risk bounds and structural results. *The Journal of Machine Learning Research*, 3(Nov):463–482, 2002.  
<https://www.jmlr.org/papers/volume3/bartlett02a/bartlett02a.pdf>.
- A. Beaulieu. *Learning SQL: generate, manipulate, and retrieve data*. O’Reilly Media, 3rd edition, 2020.  
<https://www.oreilly.com/library/view/learning-sql-3rd/9781492057604/>.
- Y. Bengio, O. Delalleau, N. L. Roux, J.-F. Paiement, P. Vincent, and M. Ouimet. Learning eigenfunctions links spectral embedding and kernel PCA. *Neural Computation*, 16(10):2197–2219, 2004.  
<https://www.mitpressjournals.org/doi/pdfplus/10.1162/0899766041732396>.
- P. Besse. Approximation spline de l’analyse en composantes principales d’une variable aléatoire hilbertienne. In *Annales de la Faculté des sciences de Toulouse: Mathématiques*, volume 12, pages 329–349. Université Paul Sabatier, 1991.  
[https://afst.centre-mersenne.org/article/AFST\\_1991\\_5\\_12\\_3\\_329\\_0.pdf](https://afst.centre-mersenne.org/article/AFST_1991_5_12_3_329_0.pdf).
- P. Besse and J. O. Ramsay. Principal components analysis of sampled functions. *Psychometrika*, 51(2): 285–311, 1986.  
<https://link.springer.com/content/pdf/10.1007/BF02293986.pdf>.
- R. Bhatia. *Matrix analysis*, volume 169 of *Graduate texts in mathematics*. Springer Science & Business Media, 1st edition, 1997.  
<https://link.springer.com/book/10.1007/978-1-4612-0653-8>.
- G. Blanchard and O. Zadorozhnyi. Concentration of weakly dependent Banach-valued sums and applications to statistical learning methods. *Bernoulli*, 25(4B):3421–3458, 2019.  
<https://arxiv.org/pdf/1712.01934.pdf>.



- G. Blanchard, O. Bousquet, and L. Zwald. Statistical properties of kernel principal component analysis. *Machine Learning*, 66(2-3):259–294, 2007.  
<https://link.springer.com/content/pdf/10.1007/s10994-006-6895-9.pdf>.
- B. Bollobás. *Linear analysis*. Cambridge mathematical textbooks. Cambridge University Press, 2nd edition, 1999.  
<https://www.cambridge.org/core/books/linear-analysis/E43EE4282F2D8636117A47A4F110E8FE>.
- J. Bouvrie and B. Hamzi. Kernel methods for the approximation of some key quantities of nonlinear systems. In *Proc. American Control Conference*, 2012.  
<https://arxiv.org/pdf/1204.0563.pdf>.
- L. Carratino, S. Vigogna, D. Calandriello, and L. Rosasco. ParK: Sound and efficient kernel ridge regression by feature space partitions. *Advances in Neural Information Processing Systems*, 34: 6430–6441, 2021.  
<https://proceedings.neurips.cc/paper/2021/file/32b9e74c8f60958158eba8d1fa372971-Paper.pdf>.
- D. L. Cohn. *Measure theory*, volume 165 of *Birkhäuser Advanced Texts Basler Lehrbücher*. Springer Science & Business Media, 2nd edition, 1980.  
<https://link.springer.com/book/10.1007/978-1-4614-6956-8>.
- J. Dauxois, A. Pousse, and Y. Romain. Asymptotic theory for the principal component analysis of a vector random function: some applications to statistical inference. *Journal of Multivariate Analysis*, 12(1):136–154, 1982.  
<https://core.ac.uk/download/pdf/82501258.pdf>.
- E. B. Davies. *Linear operators and their spectra*, volume 106 of *Cambridge studies in advanced mathematics*. Cambridge University Press, 1st edition, 2007.  
<https://www.cambridge.org/core/books/linear-operators-and-their-spectra/6DDA33D1D7032F9EBB41194F33C18A69>.
- C. Davis and W. M. Kahan. The rotation of eigenvectors by a perturbation. III. *SIAM Journal on Numerical Analysis*, 7(1):1–46, 1970.  
<https://epubs.siam.org/doi/pdf/10.1137/0707001>.
- A. C. Davison and D. V. Hinkley. *Bootstrap methods and their application*. Cambridge series in statistical and probabilistic mathematics. Cambridge University Press, 1st edition, 1997.  
<https://www.cambridge.org/core/books/bootstrap-methods-and-their-application/ED2FD043579F27952363566DC09CBD6A>.
- V. De Silva and J. B. Tenenbaum. Sparse multidimensional scaling using landmark points. Technical report, Stanford University, 2004.  
[http://graphics.stanford.edu/courses/cs468-05-winter/Papers/Landmarks/Silva\\_landmarks5.pdf](http://graphics.stanford.edu/courses/cs468-05-winter/Papers/Landmarks/Silva_landmarks5.pdf).
- E. De Vito, A. Caponnetto, and L. Rosasco. Model selection for regularized least-squares algorithm in learning theory. *Foundations of Computational Mathematics*, 5(1):59–85, 2005a.  
[https://web.mit.edu/lrosasco/www/publications/model\\_focm.pdf](https://web.mit.edu/lrosasco/www/publications/model_focm.pdf).
- E. De Vito, L. Rosasco, A. Caponnetto, U. De Giovannini, and F. Odone. Learning from examples as an inverse problem. *The Journal of Machine Learning Research*, 6(5), 2005b.  
<https://www.jmlr.org/papers/volume6/devito05a/devito05a.pdf>.

- D. Dua and C. Graff. UCI machine learning repository, 2017.  
<http://archive.ics.uci.edu/ml>.
- Z. Frangella, J. A. Tropp, and M. Udell. Randomized Nyström preconditioning. *arXiv preprint arXiv:2110.02820*, 2021.  
<https://arxiv.org/pdf/2110.02820.pdf>.
- W. A. Fuller. Properties of some estimators for the errors-in-variables model. *The Annals of Statistics*, pages 407–422, 1980.  
[https://projecteuclid.org/download/pdf\\_1/euclid.aos/1176344961](https://projecteuclid.org/download/pdf_1/euclid.aos/1176344961).
- D. Garreau, W. Jitkrittum, and M. Kanagawa. Large sample analysis of the median heuristic. *arXiv preprint arXiv:1707.07269*, 2017.  
<https://arxiv.org/pdf/1707.07269.pdf>.
- L. G. S. Giraldo, M. Rao, and J. C. Principe. Measures of entropy from data using infinitely divisible kernels. *IEEE Transactions on Information Theory*, 61(1):535–548, 2014.  
<https://ieeexplore.ieee.org/document/6954500>.
- A. Gisbrecht and F.-M. Schleich. Metric and non-metric proximity transformations at linear costs. *Neurocomputing*, 167:643–657, 2015.  
<https://arxiv.org/pdf/1411.1646>.
- A. Gittens and M. W. Mahoney. Revisiting the Nyström method for improved large-scale machine learning. *The Journal of Machine Learning Research*, 17(1):3977–4041, 2016.  
<http://www.jmlr.org/papers/volume17/gittens16a/gittens16a.pdf>.
- A. Golts and M. Elad. Linearized kernel dictionary learning. *IEEE Journal of Selected Topics in Signal Processing*, 10(4):726–739, 2016.  
<https://arxiv.org/pdf/1509.05634.pdf>.
- G. H. Golub and C. F. Van Loan. *Matrix computations*. Johns Hopkins University Press, 4th edition, 2013.  
<https://jhupbooks.press.jhu.edu/title/matrix-computations>.
- C. Graham and D. Talay. *Simulation stochastique et méthodes de Monte-Carlo*. École Polytechnique, Département de Mathématiques Appliquées, 2011.  
<https://hal.archives-ouvertes.fr/hal-00602795>.
- M. Haddouche, B. Guedj, O. Rivasplata, and J. Shawe-Taylor. Upper and lower bounds on the performance of kernel PCA. *arXiv preprint arXiv:2012.10369*, 2020.  
<https://arxiv.org/pdf/2012.10369.pdf>.
- P. Hall and M. Hosseini-Nasab. On properties of functional principal components analysis. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(1):109–126, 2006.  
<https://rss.onlinelibrary.wiley.com/doi/pdf/10.1111/j.1467-9868.2005.00535.x>.
- P. Hall, H.-G. Müller, and J.-L. Wang. Properties of principal component methods for functional and longitudinal data analysis. *The Annals of Statistics*, pages 1493–1517, 2006.  
[https://projecteuclid.org/download/pdfview\\_1/euclid.aos/1152540756](https://projecteuclid.org/download/pdfview_1/euclid.aos/1152540756).

- T. Hofmann, B. Schölkopf, and A. J. Smola. Kernel methods in machine learning. *The Annals of Statistics*, pages 1171–1220, 2008.  
<https://projecteuclid.org/journals/annals-of-statistics/volume-36/issue-3/Kernel-methods-in-machine-learning/10.1214/009053607000000677.pdf>.
- M. C. Hout, M. H. Papesh, and S. D. Goldinger. Multidimensional scaling. *Wiley Interdisciplinary Reviews: Cognitive Science*, 4(1):93–103, 2013.  
<https://onlinelibrary.wiley.com/doi/pdf/10.1002/wcs.1203>.
- A. Hyvärinen and E. Oja. Independent component analysis: algorithms and applications. *Neural Networks*, 13(4-5):411–430, 2000.  
<https://www.sciencedirect.com/science/article/pii/S0893608000000265>.
- A. Iosifidis and M. Gabbouj. Nyström-based approximate kernel subspace learning. *Pattern Recognition*, 57:190–197, 2016.  
<https://www.sciencedirect.com/science/article/pii/S0031320316300036>.
- I. T. Jolliffe. *Principal component analysis*. Springer Science & Business Media, 2nd edition, 2002.  
[http://cda.psych.uiuc.edu/statistical\\_learning\\_course/Jolliffe%20I.%20Principal%20Component%20Analysis%20\(2ed.,%20Springer,%202002\)\(518s\)\\_MVsa\\_.pdf](http://cda.psych.uiuc.edu/statistical_learning_course/Jolliffe%20I.%20Principal%20Component%20Analysis%20(2ed.,%20Springer,%202002)(518s)_MVsa_.pdf).
- I. T. Jolliffe and J. Cadima. Principal component analysis: a review and recent developments. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 374(2065): 20150202, 2016.  
<https://royalsocietypublishing.org/doi/pdf/10.1098/rsta.2015.0202>.
- T. Kato. *Perturbation theory for linear operators*, volume 132 of *Classics in mathematics*. Springer Science & Business Media, 2013.  
<https://link.springer.com/book/10.1007/978-3-642-66282-9>.
- V. Koltchinskii and E. Giné. Random matrix approximation of spectra of integral operators. *Bernoulli*, 6(1):113–167, 2000.  
[https://projecteuclid.org/download/pdf\\_1/euclid.bj/1082665383](https://projecteuclid.org/download/pdf_1/euclid.bj/1082665383).
- E. Kreyszig. *Introductory functional analysis with applications*. Wiley, 1st edition, 1989.  
<https://www.wiley.com/en-gb/Introductory+Functional+Analysis+with+Applications-p-9780471504597>.
- M. Ledoux and M. Talagrand. *Probability in Banach spaces: isoperimetry and processes*. Springer Science & Business Media, 2013.  
<https://www.springer.com/gp/book/9783642202117>.
- T. Liang and A. Rakhlin. Just interpolate: kernel “ridgeless” regression can generalize. *The Annals of Statistics*, 48(3):1329–1347, 2020.  
<https://arxiv.org/pdf/1808.00387.pdf>.
- H. Lodhi, C. Saunders, J. Shawe-Taylor, N. Cristianini, and C. Watkins. Text classification using string kernels. *The Journal of Machine Learning Research*, 2(Feb):419–444, 2002.  
<http://www.jmlr.org/papers/volume2/lodhi02a/lodhi02a.pdf>.

- S. Ma and M. Belkin. Diving into the shallows: a computational perspective on large-scale shallow learning. In *Advances in Neural Information Processing Systems*, pages 3778–3787, 2017.  
<https://papers.nips.cc/paper/6968-diving-into-the-shallows-a-computational-perspective-on-large-scale-shallow-learning.pdf>.
- C. McDiarmid. On the method of bounded differences. *Surveys in Combinatorics*, 141(1):148–188, 1989.  
<https://www.cambridge.org/no/academic/subjects/mathematics/discrete-mathematics-information-theory-and-coding/surveys-combinatorics-1989-invited-papers-twelfth-british-combinatorial-conference?format=PB&isbn=9780521378239>.
- G. Meanti, L. Carratino, L. Rosasco, and A. Rudi. Kernel methods through the roof: handling billions of points efficiently. *arXiv preprint arXiv:2006.10350*, 2020.  
<https://arxiv.org/abs/2006.10350>.
- S. Mika, G. Rätsch, J. Weston, B. Schölkopf, and K.-R. Müller. Fisher discriminant analysis with kernels. In *Neural Networks for Signal Processing IX: Proceedings of the 1999 IEEE Signal Processing Society Workshop (cat. no. 98th8468)*, pages 41–48. IEEE, 1999.  
<https://ieeexplore.ieee.org/abstract/document/788121>.
- J. Neyman. Outline of a theory of statistical estimation based on the classical theory of probability. *Philosophical Transactions of the Royal Society of London. Series A, Mathematical and Physical Sciences*, 236(767):333–380, 1937.  
<https://royalsocietypublishing.org/doi/pdf/10.1098/rsta.1937.0005?download=true>.
- J. Neyman and E. S. Pearson. On the problem of the most efficient tests of statistical hypotheses. *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character*, 231(694-706):289–337, 1933.  
<https://royalsocietypublishing.org/doi/pdf/10.1098/rsta.1933.0009?download=true>.
- E. J. Nyström. Über die praktische auflösung von integralgleichungen mit anwendungen auf randwertaufgaben. *Acta Mathematica*, 54(1):185–204, 1930.  
<https://link.springer.com/content/pdf/10.1007/BF02547521.pdf>.
- V. I. Paulsen and M. Raghupathi. *An introduction to the theory of reproducing kernel Hilbert spaces*, volume 152 of *Cambridge studies in advanced mathematics*. Cambridge University Press, 2016.  
<https://www.cambridge.org/core/books/an-introduction-to-the-theory-of-reproducing-kernel-hilbert-spaces/C3FD9DF5F5C21693DD4ED812B531269A>.
- K. Pearson. On lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 6(2):559–572, 1901.  
<https://www.tandfonline.com/doi/abs/10.1080/14786440109462720>.
- I. Pinelis. Optimum bounds for the distributions of martingales in Banach spaces. *The Annals of Probability*, pages 1679–1706, 1994.  
[https://projecteuclid.org/download/pdf\\_1/euclid.aop/1176988477](https://projecteuclid.org/download/pdf_1/euclid.aop/1176988477).
- J. Platt. FastMap, MetricMap, and Landmark MDS are all Nyström algorithms. In *International Conference on Artificial Intelligence and Statistics*, pages 261–268. PMLR, 2005.  
<http://proceedings.mlr.press/r5/platt05a/platt05a.pdf>.

- K. M. Ramachandran and C. P. Tsokos. *Mathematical statistics with applications in R*. Academic Press, 2nd edition, 2015.  
<https://www.elsevier.com/books/mathematical-statistics-with-applications-in-r/ramachandran/978-0-12-417113-8>.
- C. Robert and G. Casella. *Monte Carlo statistical methods*. Springer Science & Business Media, 2nd edition, 2004.  
<https://www.springer.com/gp/book/9780387212395>.
- L. Rosasco, M. Belkin, and E. D. Vito. On learning with integral operators. *The Journal of Machine Learning Research*, 11(Feb):905–934, 2010.  
<http://www.jmlr.org/papers/volume11/rosasco10a/rosasco10a.pdf>.
- R. Rosipal and L. J. Trejo. Kernel partial least squares regression in reproducing kernel Hilbert space. *The Journal of Machine Learning Research*, 2(Dec):97–123, 2001.  
<https://www.jmlr.org/papers/volume2/rosipal01a/rosipal01a.pdf>.
- R. Rosipal, L. J. Trejo, and A. Cichocki. *Kernel principal component regression with EM approach to nonlinear principal components extraction*. University of Paisley, 2000.  
[http://aiolos.um.savba.sk/~roman/Papers/tr00\\_2.pdf](http://aiolos.um.savba.sk/~roman/Papers/tr00_2.pdf).
- R. Rosipal, M. Girolami, L. J. Trejo, and A. Cichocki. Kernel PCA for feature extraction and de-noising in nonlinear regression. *Neural Computing & Applications*, 10(3):231–243, 2001.  
[https://www.researchgate.net/profile/Leonard-Trejo/publication/243134486\\_Kernel\\_PCA\\_for\\_Feature\\_Extraction\\_and\\_De-Noising\\_in\\_Nonlinear\\_Regression/links/583f5da508ae8e63e6182cbf/Kernel-PCA-for-Feature-Extraction-and-De-Noising-in-Nonlinear-Regression.pdf](https://www.researchgate.net/profile/Leonard-Trejo/publication/243134486_Kernel_PCA_for_Feature_Extraction_and_De-Noising_in_Nonlinear_Regression/links/583f5da508ae8e63e6182cbf/Kernel-PCA-for-Feature-Extraction-and-De-Noising-in-Nonlinear-Regression.pdf).
- S. T. Roweis and L. K. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500):2323–2326, 2000.  
<https://science.sciencemag.org/content/290/5500/2323.full>.
- A. Rudi, R. Camoriano, and L. Rosasco. Less is more: Nyström computational regularization. In *Advances in Neural Information Processing Systems*, pages 1657–1665, 2015.  
<https://arxiv.org/pdf/1507.04717.pdf>.
- A. Rudi, L. Carratino, and L. Rosasco. Falkon: an optimal large scale kernel method. In *Advances in Neural Information Processing Systems*, pages 3888–3898, 2017.  
<http://papers.nips.cc/paper/6978-falkon-an-optimal-large-scale-kernel-method.pdf>.
- B. Schölkopf, A. Smola, and K.-R. Müller. Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation*, 10(5):1299–1319, 1998.  
<https://www.mitpressjournals.org/doi/pdfplus/10.1162/089976698300017467>.
- P. K. Sen, J. M. Singer, and A. C. P. De Lima. *From finite sample to asymptotic methods in statistics*, volume 28 of *Cambridge series in statistical and probabilistic mathematics*. Cambridge University Press, 2010.  
<https://www.cambridge.org/core/books/from-finite-sample-to-asymptotic-methods-in-statistics/07DFA2860E18EDB1A9FE1FF3B4E07F0C>.

- J. Shawe-Taylor, C. K. Williams, N. Cristianini, and J. Kandola. On the eigenspectrum of the Gram matrix and its relationship to the operator eigenspectrum. In *International Conference on Algorithmic Learning Theory*, pages 23–40. Springer Science & Business Media, 2002.  
[https://link.springer.com/chapter/10.1007/3-540-36169-3\\_4](https://link.springer.com/chapter/10.1007/3-540-36169-3_4).
- J. Shawe-Taylor, C. K. Williams, N. Cristianini, and J. Kandola. On the eigenspectrum of the Gram matrix and the generalization error of kernel PCA. *IEEE Transactions on Information Theory*, 51(7): 2510–2522, 2005.  
<https://homepages.inf.ed.ac.uk/ckiw/postscript/gram.pdf>.
- R. Singh, M. Sahani, and A. Gretton. Kernel instrumental variable regression. In *Advances in Neural Information Processing Systems*, pages 4593–4605, 2019.  
<http://papers.nips.cc/paper/8708-kernel-instrumental-variable-regression.pdf>.
- M. Sipser. *Introduction to the theory of computation*. Cengage Learning, 3rd edition, 2013.  
<https://www.cengagebrain.co.uk/shop/isbn/9780357670583>.
- B. Sriperumbudur and N. Sterge. Approximate kernel PCA using random features: computational vs. statistical trade-off. *arXiv preprint arXiv:1706.06296*, 2017.  
<https://arxiv.org/pdf/1706.06296.pdf>.
- N. Sterge and B. Sriperumbudur. Statistical optimality and computational efficiency of Nyström kernel PCA. *arXiv preprint arXiv:2105.08875v1*, 2021.  
<https://arxiv.org/pdf/2105.08875v1>.
- N. Sterge, B. Sriperumbudur, L. Rosasco, and A. Rudi. Gain with no pain: efficiency of kernel-PCA by Nyström sampling. In *International Conference on Artificial Intelligence and Statistics*, pages 3642–3652. PMLR, 2020.  
<http://proceedings.mlr.press/v108/sterge20a/sterge20a.pdf>.
- S. V. N. Vishwanathan, N. N. Schraudolph, R. Kondor, and K. M. Borgwardt. Graph kernels. *The Journal of Machine Learning Research*, 11:1201–1242, 2010.  
<http://www.jmlr.org/papers/volume11/vishwanathan10a/vishwanathan10a.pdf>.
- T. Wang, Q. Berthet, and R. J. Samworth. Statistical and computational trade-offs in estimation of sparse principal components. *The Annals of Statistics*, 44(5):1896–1930, 2016.  
[https://projecteuclid.org/download/pdfview\\_1/euclid.aos/1473685263](https://projecteuclid.org/download/pdfview_1/euclid.aos/1473685263).
- A. Wibowo and Y. Yamamoto. A note on kernel principal component regression. *Computational Mathematics and Modeling*, 23(3):350–367, 2012.  
<https://link.springer.com/content/pdf/10.1007/s10598-012-9143-0.pdf>.
- V. Wild, M. Kanagawa, and D. Sejdinovic. Connections and equivalences between the Nyström method and sparse variational Gaussian processes. *arXiv preprint arXiv:2106.01121*, 2021.  
<https://arxiv.org/pdf/2106.01121.pdf>.
- C. K. Williams. On a connection between kernel PCA and metric multidimensional scaling. *Machine Learning*, 46(1):11–19, 2002.  
<https://link.springer.com/content/pdf/10.1023/A:1012485807823.pdf>.

- C. K. Williams and M. Seeger. Using the Nyström method to speed up kernel machines. In *Advances in Neural Information Processing Systems*, pages 682–688, 2001.  
<http://papers.nips.cc/paper/1866-using-the-nystrom-method-to-speed-up-kernel-machines.pdf>.
- S. Wold, K. Esbensen, and P. Geladi. Principal component analysis. *Chemometrics and Intelligent Laboratory Systems*, 2(1-3):37–52, 1987.  
<https://www.sciencedirect.com/science/article/abs/pii/0169743987800849>.
- Y. Yu, T. Wang, and R. J. Samworth. A useful variant of the Davis–Kahan theorem for statisticians. *Biometrika*, 102(2):315–323, 2015.  
<https://arxiv.org/pdf/1405.0680.pdf>.
- L. Zhang, T. Yang, J. Yi, R. Jin, and Z.-H. Zhou. Stochastic optimization for kernel PCA. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, 2016.  
<https://www.aaai.org/ocs/index.php/AAAI/AAAI16/paper/viewFile/12072/11878>.
- L. Zwald and G. Blanchard. On the convergence of eigenspaces in kernel principal component analysis. In *Advances in Neural Information Processing Systems*, 2005.  
<https://hal.archives-ouvertes.fr/file/index/docid/373803/filename/Nips2005mod.pdf>.