Kernel Principal Component Analysis

Alex Beeny abeeny@siue.edu

Draft: April 10, 2024

Contents

Abstract

Principal component analysis (PCA) and kernel methods are tools often used in data science. The underlying theory of these tools depend on the properties of a special type of Hilbert space called a reproducing kernel Hilbert space (RKHS). This paper explores the essence of RKHSs using data science examples, in particular, PCA and kernel PCA. When kernel methods are applied to PCA, we can analyze nonlinear data in a high-dimensional feature space with some nice properties.

1 Introduction

2 Principal Component Analysis

When analyzing data, it can be convenient to transform the given input variables to produce new features. For a well-chosen transform, these features may be approximated using fewer dimensions than the original input space [9]. This is an example of a data preprocessing technique known as *dimension reduction* and can reveal low-dimensional structure.

Principal component analysis (PCA) is an orthogonal coordinate transform that is suitable for dimension reduction if some of the inputs are linearly correlated. In this case, PCA transforms redundant variables in the input space producing uncorrelated variables in the feature space.

There are a number of ways to derive the optimal PCA transform. One approach presented in [9] is based on finding uncorrelated features. It is straightforward to show that uncorrelated features have a diagonal covariance matrix. This can be used to solve for the covariance matrix C of input variables. By asserting the orthogonality of the PCA transform, we obtain V from the diagonalization of the covariance matrix $C = VDV^\top$. Given this PCA transform, we can show that V minimizes projection residuals as in [17].

2.1 Finding uncorrelated features

The correlation between two random variables x and y is defined as

$$\operatorname{corr}(x, y) = \frac{E[(x - \mu_x)(y - \mu_y)]}{\sigma_x \sigma_y}, \tag{1}$$

where μ_x, μ_y and σ_x, σ_y are the respective means and standard deviations of x and y. We say x and y are uncorrelated when $\operatorname{corr}(x, y) = 0$. This happens if and only if

$$E[(x - \mu_x)(y - \mu_y)] = \operatorname{cov}(x, y) = 0. \tag{2}$$

The covariance matrix for a multivariate random variable $x = [x_1, x_2, \ldots, x_d]$ (as a row vector) has $\operatorname{cov}(x_i, x_j)$ in the i-th row and j-th column. Then

$$E[(x - \mu_x)^\top (x - \mu_x)] = \begin{bmatrix} \operatorname{cov}(x_1, x_1) & \operatorname{cov}(x_1, x_2) & \cdots & \operatorname{cov}(x_1, x_d) \\ \operatorname{cov}(x_2, x_1) & \operatorname{cov}(x_2, x_2) & \cdots & \operatorname{cov}(x_2, x_d) \\ \vdots & \vdots & \ddots & \vdots \\ \operatorname{cov}(x_d, x_1) & \operatorname{cov}(x_d, x_2) & \cdots & \operatorname{cov}(x_d, x_d) \end{bmatrix}. \tag{3}$$

If $x_1, x_2, \ldots, x_d$ are pairwise uncorrelated, then $\operatorname{cov}(x_i, x_j) = 0$ for all $i \neq j$. Hence, uncorrelated variables have a diagonal covariance matrix.

Now, let $a_1, a_2, \ldots, a_n \in \mathbb{R}^{1\times d}$ represent n observations in d variables. These observations can be considered points in d-dimensional space whose centroid is $\mu_a = \frac{1}{n}\sum_{i=1}^n a_i$. We want to determine a PCA transform which sends these points in the input space to points in the feature space. Moreover, the basis vectors of the feature space shall be uncorrelated. Accordingly, let $V \in \mathbb{R}^{d\times d}$ be the change of basis matrix and let

$$b_i = (a_i - \mu_a)V, \quad \text{for } i = 1, 2, \ldots, n \tag{4}$$

be observations with respect to the feature coordinates. Then

$$\mu_b = \frac{1}{n}\sum_{i=1}^n b_i = \frac{1}{n}\sum_{i=1}^n (a_i - \mu_a)V = 0. \tag{5}$$

Using equation (3), we can compute the sample covariance matrices as

$$C = \frac{1}{n-1}\sum_{i=1}^{n}(a_i - \mu_a)^\top (a_i - \mu_a), \qquad D = \frac{1}{n-1}\sum_{i=1}^{n} b_i^\top b_i. \tag{6}$$

Since D is the covariance matrix of uncorrelated features, by the argument above, it is diagonal. If we restrict V to be orthogonal, then

$$b_i^\top b_i = V^\top (a_i - \mu_a)^\top (a_i - \mu_a) V \implies D = V^\top C V \implies C = V D V^\top. \tag{7}$$

Hence, V must be a matrix of orthonormal eigenvectors $v_1, v_2, \ldots, v_d$ corresponding to eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_d$ on the diagonal of D. When the eigenvalues and eigenvectors are ordered such that

$$\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_d, \tag{8}$$

we call $v_1, v_2, \ldots, v_d$ the *principal components* of the PCA transform matrix V.

2.2 Singular value decomposition

The singular value decomposition (SVD) of a rectangular matrix generalizes the idea of diagonalization for square matrices. Moreover, this provides a connection between the matrices $A^\top A$ and $AA^\top$.

Theorem 2.1 (Singular value decomposition)**.** *[8] Let $A \in \mathbb{R}^{n\times d}$. Then there exist orthogonal matrices $U \in \mathbb{R}^{n\times n}$ and $V \in \mathbb{R}^{d\times d}$ and a diagonal matrix $S \in \mathbb{R}^{n\times d}$ such that $A = USV^\top$. We say the columns of $U = \begin{bmatrix} u_1 & u_2 & \cdots & u_n \end{bmatrix}$ and $V = \begin{bmatrix} v_1 & v_2 & \cdots & v_d \end{bmatrix}$ are the left and right singular vectors of A, respectively The diagonal entries of S are the called the singular values $\sigma_1, \sigma_2, \ldots, \sigma_r$, where $r = \operatorname{rank} A \leq \min\{n, d\}$. Then we can write*

$$A = USV^\top = \sum_{i=1}^{r} \sigma_i u_i v_i^\top. \tag{9}$$

The SVD of a matrix A can be found by diagonalizing $A^\top A$ and $AA^\top$. If $A = USV^\top$, then

$$\begin{aligned} A^\top A &= (USV^\top)^\top (USV^\top) = VSU^\top USV^\top = VS^2V^\top \\ AA^\top &= (USV^\top)(USV^\top)^\top = USV^\top VSU^\top = US^2U^\top. \end{aligned}$$

So, $\{v_j\}_{j=1}^{d}$ are the eigenvectors of $A^\top A$, $\{u_j\}_{j=1}^{n}$ are the eigenvectors of $AA^\top$, and $\{\sigma_j^2\}_{j=1}^{r}$ are the eigenvalues of both $A^\top A$ and $AA^\top$. Notice that the SVD of A will give us the projection matrix V in equation (7), provided that A is centered. In this way, we see that PCA is really just a special case of the SVD.

Theorem 2.2 (Frobenius norm)**.** *[12, 8] The Frobenius norm (or Hilbert-Schmidt norm) of a matrix $A = [a_{ij}] \in \mathbb{R}^{n\times d}$ is given by*

$$\|A\|_F = \sqrt{\operatorname{tr}(A^\top A)} = \sqrt{\sum_{i=1}^{n}\sum_{j=1}^{d} a_{ij}^2}. \tag{10}$$

Proof. Let $A = [a_{ij}] \in \mathbb{R}^{n\times d}$. Then $A^\top A = [\sum_{k=1}^n a_{ki}a_{kj}]_{ij}$. It follows that $\|A\|_F^2 = \operatorname{tr}(A^\top A) = \sum_{i=1}^n \sum_{j=1}^d a_{ij}^2$. Clearly, $\|A\|_F > 0$ whenever A is not the zero matrix and $\|A\|_F = 0$ whenever A is the zero matrix.

For the triangle inequality, consider another matrix $B = [b_{ij}] \in \mathbb{R}^{d\times m}$. Then

$$\begin{aligned}
\|AB\|_F &= \sqrt{\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{k=1}^{d}(a_{ik}b_{kj})^2} \\
&\le \sqrt{\sum_{i=1}^{n}\sum_{j=1}^{m}\left(\sum_{k=1}^{d} a_{ik}^2\right)\left(\sum_{k=1}^{d} b_{kj}^2\right)} \\
&= \sqrt{\sum_{i=1}^{n}\sum_{j=1}^{d} a_{ij}^2}\sqrt{\sum_{i=1}^{d}\sum_{j=1}^{m} b_{ij}^2} \\
&= \|A\|_F\|B\|_F.
\end{aligned}$$

Thus, $\|\cdot\|_F$ is a matrix norm. □

Combining equations (9) and (10), we have

$$\|A\|_F = \sqrt{\sum_{i=1}^{r}\sigma_i^2} = \sqrt{\sum_{i=1}^{r}\lambda_i^2}, \tag{11}$$

where $\{\sigma_i\}_{i=1}^r$ are the singular values of A and $\{\lambda_i\}_{i=1}^r$ are the eigenvalues of $A^\top A$ or $AA^\top$.

2.3 Minimizing projection residuals

Let $a_1, a_2, \ldots, a_n$ be points in $\mathbb{R}^d$. Assume that these points have zero mean, that is, $\frac{1}{n}\sum_{i=1}^n a_i$ is the zero vector in $\mathbb{R}^d$. Let A be the $n \times d$ matrix whose rows are given by $a_1, a_2, \ldots, a_n$. Then the covariance matrix from equation (6) is given by $C = \frac{1}{n-1}A^\top A$.

2.4 PCA algorithm

Let A be a data matrix whose n rows correspond to observations and d columns correspond to variables. The following algorithm demonstrates a simple method for computing the PCA of A:

1. Compute the centered matrix $A_0 = A - \operatorname{col\,mean}(A)$.
2. Compute the covariance matrix $C = \frac{1}{n-1} A_0^\top A_0$.
3. Diagonalize the covariance matrix such that $C = VDV^\top$.
4. Order the eigenvalues and eigenvectors so that $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_d$. We call the ordered eigenvalues the *principal components*.
5. Choose the dimension of the subspace $p \leq d$.
6. Construct the $d \times p$ projection matrix V_p using the first p principal components $v_1, v_2, \ldots, v_p$.

Example 2.3. Consider the following matrix

$$A = \begin{bmatrix} 5 & 3 & 6 & 7 & 6 \\ 4 & 5 & 7 & 1 & 3 \\ 5 & 7 & 6 & 1 & 0 \\ 6 & 10 & 12 & 12 & 11 \\ 9 & 10 & 12 & 13 & 9 \end{bmatrix}.$$

The column means are $\mu = [5.8, 7, 8.6, 6.8, 5.8]$. Then the mean-centered data becomes

$$X = A - \mu = \frac{1}{5} \begin{bmatrix} -4 & -20 & -13 & 1 & 1 \\ -9 & -10 & -8 & -29 & -14 \\ -4 & 0 & -13 & -29 & -29 \\ 1 & 15 & 17 & 26 & 26 \\ 16 & 15 & 17 & 31 & 16 \end{bmatrix}.$$

The covariance matrix is

$$C = X^T X = \frac{1}{5} \begin{bmatrix} 74 & 85 & 93 & 179 & 104 \\ 85 & 190 & 170 & 225 & 150 \\ 93 & 170 & 196 & 313 & 238 \\ 179 & 225 & 313 & 664 & 484 \\ 104 & 150 & 238 & 484 & 394 \end{bmatrix}.$$

Diagonalizing C gives

$$V = \begin{bmatrix} 0.1888 & -0.2020 & -0.6366 & 0.5495 & -0.4651 \\ 0.2755 & -0.7886 & 0.1472 & -0.4502 & -0.2791 \\ 0.3606 & -0.3464 & 0.3128 & 0.5836 & 0.5582 \\ 0.6979 & 0.2522 & -0.4422 & -0.3707 & 0.3411 \\ 0.5209 & 0.3922 & 0.5288 & 0.1316 & -0.5271 \end{bmatrix},$$

$$D = \begin{bmatrix} 264.8458 & 0 & 0 & 0 & 0 \\ 0 & 27.9766 & 0 & 0 & 0 \\ 0 & 0 & 9.3198 & 0 & 0 \\ 0 & 0 & 0 & 1.4579 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

If we keep all 5 principal component vectors, then $V_5 = V$ and the projection of X along V is

$$P = XV = \begin{bmatrix} -1.9469 & 4.3453 & -0.8756 & -0.2039 & 0 \\ -6.9742 & -0.0660 & 1.4352 & 0.7590 & 0 \\ -8.1577 & -2.6752 & -0.8063 & -0.5704 & 0 \\ 8.4282 & -0.2330 & 1.8282 & -0.4996 & 0 \\ 8.6507 & -1.3711 & -1.5815 & 0.5149 & 0 \end{bmatrix}.$$

Here, the last column of P is the zero vector because the last eigenvalue of C is zero[1]. To perfectly reconstruct A, we need $k = 4$ principal components and the row vector μ

$$A = PV^T + \mu = PV_4^T + \mu.$$

If we use $k = 3$ principal components, then the projection of X onto V_3 is

$$P = XV_3 = \begin{bmatrix} -1.9469 & 4.3453 & -0.8756 \\ -6.9742 & -0.0660 & 1.4352 \\ -8.1577 & -2.6752 & -0.8063 \\ 8.4282 & -0.2330 & 1.8282 \\ 8.6507 & -1.3711 & -1.5815 \end{bmatrix}$$

and A is approximately reconstructed by

$$A \approx PV_3^T + \mu = \begin{bmatrix} 5.1 & 2.9 & 6.1 & 6.9 & 6.0 \\ 3.6 & 5.3 & 6.6 & 1.3 & 2.9 \\ 5.3 & 6.7 & 6.3 & 0.8 & 0.1 \\ 6.3 & 9.8 & 12.3 & 11.8 & 11.1 \\ 8.7 & 10.2 & 11.7 & 13.2 & 8.9 \end{bmatrix}.$$

We can compute the reconstruction error using

$$E_k = \|A - (PV_k^T + \mu)\|_F,$$

where $\|\cdot\|_F$ is the Frobenius norm. By the SVD, we have $X = USV^T$, where $S = \sqrt{D}$. So, the projection of X onto V_k is

$$P = XV_k = US_k,$$

where S_k is the diagonal matrix of the first k singular values. Then the reconstruction error becomes

$$\begin{aligned} \|A - (PV_k^T + \mu)\|_F &= \|(A - \mu) - PV_k^T\|_F \\ &= \|X - PV_k^T\|_F \\ &= \|USV^T - US_kV^T\|_F \\ &= \|U(S - S_k)V^T\|_F \\ &= \|S - S_k\|_F \\ &= \sigma_k + \sigma_{k+1} + \cdots + \sigma_p. \end{aligned}$$

[1]Since we subtracted the column means from a square matrix A, the dimension of the row space was reduced to 4.

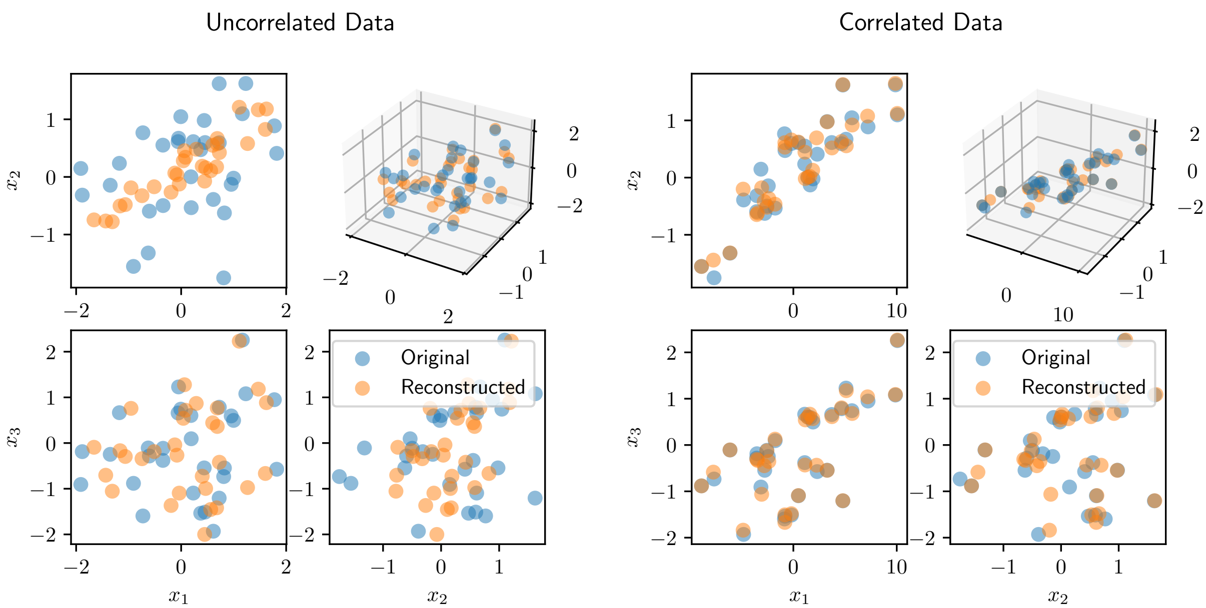

Figure 1: PCA projection of three-dimensional data onto two-dimensional subspace.

Hence,

$$E_3 = \sigma_3 + \sigma_4 = \sqrt{1.4579} + 0 = 1.2074.$$

2.5 Applications of PCA

One of the most common applications of PCA is *dimension reduction*. When variables are correlated, the observations lie in some linear subspace of the original space. In this situation, PCA can be used to project down to the lower dimension and have the smallest possible projection error. This is particularly useful when the dimension of the input space is extremely large. A number challenges arise when analyzing high-dimensional data and are collectively referred to as the *curse of dimensionality* [9]. By working in the PCA feature space, these problems may be avoided.

Example 2.4. In the following experiment, we consider two sets of three-dimensional data. Each set of vectors $x_1, x_2, x_3 \in \mathbb{R}^n$ are sampled from a standard normal distribution with sample size $n = 30$. Suppose in the first set there is no apparent relationship among variables while, in second set, we have $x_1 = 4x_2 + 2x_3$. For simplicity, we say the first set is *uncorrelated* while the second set is *correlated*. See Figure 1. In the uncorrelated data, the reconstruction error is 3.92, which does not seem to indicate the presence of a pattern. The reconstruction error for the correlated data is 0.973. This error can be explained by the variation of x_2 with x_3. Meanwhile, the correlated pair plots indicate a pattern among x_2 vs x_1 and x_3 vs x_1.

3 Reproducing Kernel Hilbert Space

In this section, our goal is to establish properties of Hilbert spaces and kernel functions that can be used to modify the PCA algorithm. To begin, we will briefly cite some definitions and results from analysis [10], [14] and matrix theory [8].

3.1 Inner products and Hilbert spaces

Definition 3.1 (Definite matrix)**.** [8] Let A be an $n \times n$ matrix over the real numbers having the form

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix} = [a_{ij}]. \tag{12}$$

The *transpose* of A is $A^\top = [a_{ji}]$. We say that A is *symmetric* whenever $A = A^\top$. A symmetric matrix A is

1. *positive definite* if $x^\top Ax > 0$, for all nonzero $x \in \mathbb{R}^n$, or
2. *positive semidefinite* if $x^\top Ax \geq 0$, for all $x \in \mathbb{R}^n$.

Negative (semi)definite matrices can be defined in a similar fashion. A matrix is *definite* if it is either positive semidefinite or negative semidefinite. Otherwise, A is an *indefinite matrix.*

Be aware that some authors use the terms positive definite ($>$) and nonnegative definite ($\geq$). Other authors use the modifier *strict* as in strict positive definite ($>$) and positive definite ($\geq$).

Definition 3.2 (Inner product)**.** [19] Let $\langle \cdot, \cdot \rangle : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ be a function defined on the vector space $\mathcal{X}$. Then $\langle \cdot, \cdot \rangle$ is an *inner product* if the following properties hold for all $x, y, z \in \mathcal{X}$ and $\alpha, \beta \in \mathbb{R}$:

1. $\langle x, y \rangle = \langle y, x \rangle$; (symmetry)
2. $\langle \alpha x + \beta y, z \rangle = \alpha \langle x, z \rangle + \beta \langle y, z \rangle$; (bilinear)
3. $\langle x, x \rangle \geq 0$ and $\langle x, x \rangle = 0$ if and only if $x = 0$. (positive definite)

Note that linearity in the first argument with symmetry implies that the inner product is bilinear (linear in both arguments). Since inner products are positive definite, they induce a norm $\| \cdot \| : \mathcal{X} \to \mathbb{R}$ and metric $d : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ such that

$$\|x\| = \sqrt{\langle x, x \rangle} \quad \text{and} \quad d(x, y) = \|x - y\|. \tag{13}$$

An *inner product space*, *normed space*, and *metric space* are vector spaces along with an inner product, norm, and metric, respectively. It follows that an inner product space is also a normed space and a metric space. Then the induced norm has the following properties for all $x, y \in \mathcal{X}$ and $\alpha \in \mathbb{R}$:

1. $\|\alpha x\| = |\alpha|\|x\|$;
2. $\|x\| \geq 0$ and $\|x\| = 0$ if and only if $x = 0$;
3. $\|x + y\| \leq \|x\| + \|y\|$; (triangle inequality)
4. $\langle x, y\rangle^2 \leq \|x\|\|y\|$. (Cauchy-Schwarz inequality)

Example 3.3. The *Euclidean inner product* (or *dot product*) is the function $\langle\cdot,\cdot\rangle : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ such that

$$\langle x, y\rangle = \sum_{i=1}^{n} x_i y_i = x^\top y, \tag{14}$$

for all $x = [x_i], y = [y_i] \in \mathbb{R}^n$. Sometimes we write $x \cdot y$ to mean the Euclidean inner product. This induces the *Euclidean norm*

$$\|x\| = \sqrt{\sum_{i=1}^{n} x_i^2} \tag{15}$$

Definition 3.4 (Hilbert space)**.** [10] A metric space is *complete* if the limit of every Cauchy sequence is in the space. A complete normed space is called a *Banach space.* A complete inner product space $\mathcal{H}$ is called a *Hilbert space.* We sometimes denote the inner product and norm of $\mathcal{H}$ as $\langle\cdot,\cdot\rangle_{\mathcal{H}}$ and $\|\cdot\|_{\mathcal{H}}$ to avoid ambiguity.

A Hilbert space is said to be *separable* if it contains a dense countable subset.

Two real Hilbert spaces $\mathcal{H}$ and $\mathcal{L}$ are said to be *isomorphic* if there is a linear bijection $T : \mathcal{H} \to \mathcal{L}$ such that $\langle x, y\rangle_{\mathcal{H}} = \langle Tx, Ty\rangle_{\mathcal{L}}$, for every $x, y \in \mathcal{H}$.

The *dimension* of a Hilbert space is the cardinality of its basis.

Theorem 3.5 (Properties of Hilbert spaces)**.** *[10] Then the following properties hold for Hilbert spaces.*

1. *A Hilbert space is separable if and only if it has a countable orthonormal basis.*
2. *Two Hilbert spaces are isomorphic if and only if they have the same dimension.*
3. *Any inner product $\mathcal{H}$ space can be extended to a Hilbert space by completion and is unique up to isomorphism. We denote the completion as $\overline{\mathcal{H}}$.*
4. *A Hilbert space $\mathcal{H}$ can be decomposed into orthogonal subspaces M and $M^\perp$ such that whenever $f \in M$ and $g \in M^\perp$, then $\langle f, g\rangle = 0$ [3]. We denote the* orthogonal decomposition *of a Hilbert space as the direct sum $\mathcal{H} = M \oplus M^\perp$.*

The following example demonstrates a useful property of separable Hilbert spaces.

Example 3.6. [14] Let A be a nonempty index set. The space of square-summable indexed families is defined as

$$\ell^2(A) = \left\{ x : A \to \mathbb{R} \;\middle|\; \sum_{a \in A} x_a^2 < \infty \right\}. \tag{16}$$

Given the inner product

$$\langle x, y \rangle = \sum_{a \in A} x_a y_a, \tag{17}$$

$\ell^2(A)$ is a Hilbert space. Moreover, $\ell^2(A)$ is separable if and only if A is countable. It follows that the sequence space $\ell^2 = \ell^2(\mathbb{N})$ is the separable Hilbert space of square-summable sequences. Due to the Riesz-Fischer theorem, every infinite-dimensional Hilbert space is isomorphic to ℓ^2.

Definition 3.7 (Gram matrix)**.** [8] Let $x_1, x_2, \ldots, x_n \in \mathcal{X}$ for some inner product space $\mathcal{X}$ equipped with $\langle \cdot, \cdot \rangle$. We say G is a *Gram matrix* (or *Gramian*) for the sequence of vectors $x_1, x_2, \ldots, x_n$ with respect to $\langle \cdot, \cdot \rangle$ if $G = \big[\langle x_i, x_j \rangle\big]_{ij}$.

Example 3.8. Consider the vectors in $\mathbb{R}^3$:

$$\mathbf{v}_1 = \begin{bmatrix} v_{11} \\ v_{21} \\ v_{31} \end{bmatrix}, \qquad \mathbf{v}_2 = \begin{bmatrix} v_{12} \\ v_{22} \\ v_{32} \end{bmatrix}, \qquad \mathbf{v}_3 = \begin{bmatrix} v_{13} \\ v_{23} \\ v_{33} \end{bmatrix}, \qquad \mathbf{v}_4 = \begin{bmatrix} v_{14} \\ v_{24} \\ v_{34} \end{bmatrix}.$$

The Gram matrix for these vectors is

$$G = \begin{bmatrix} \mathbf{v}_1^\top \mathbf{v}_1 & \mathbf{v}_1^\top \mathbf{v}_2 & \mathbf{v}_1^\top \mathbf{v}_3 & \mathbf{v}_1^\top \mathbf{v}_4 \\ \mathbf{v}_2^\top \mathbf{v}_1 & \mathbf{v}_2^\top \mathbf{v}_2 & \mathbf{v}_2^\top \mathbf{v}_3 & \mathbf{v}_2^\top \mathbf{v}_4 \\ \mathbf{v}_3^\top \mathbf{v}_1 & \mathbf{v}_3^\top \mathbf{v}_2 & \mathbf{v}_3^\top \mathbf{v}_3 & \mathbf{v}_3^\top \mathbf{v}_4 \\ \mathbf{v}_4^\top \mathbf{v}_1 & \mathbf{v}_4^\top \mathbf{v}_2 & \mathbf{v}_4^\top \mathbf{v}_3 & \mathbf{v}_4^\top \mathbf{v}_4 \end{bmatrix}.$$

If V is a matrix whose columns are $\mathbf{v}_1$, $\mathbf{v}_2$, $\mathbf{v}_3$, $\mathbf{v}_4$, then we can write $G = V^\top V$.

Theorem 3.9. *[8] A matrix G is a Gram matrix if and only if G is positive semidefinite.*

Proof. ($\Rightarrow$) Suppose G is the Gram matrix of $x_1, x_2, \ldots, x_n$ with respect to $\langle \cdot, \cdot \rangle$. Let $c_1, c_2, \ldots, c_n \in \mathbb{R}$. Then G is positive semidefinite because

$$\sum_{i=1}^n \sum_{j=1}^n c_i c_j \langle x_i, x_j \rangle = \left\langle \sum_{i=1}^n c_i x_i, \sum_{j=1}^n c_j x_j \right\rangle = \left\| \sum_{i=1}^n c_i x_i \right\|^2 \geq 0. \tag{18}$$

($\Leftarrow$) Suppose G is positive semidefinite. Then G can be factored as $G = B^\top B$. Let $b_1, b_2, \ldots, b_n$, be the columns of B. Then $G = [b_i^\top b_j]_{ij}$. Hence G is the Gram matrix of $b_1, b_2, \ldots, b_n$ with respect to the dot product. □

Theorem 3.10. *[8] A Gram matrix G of $x_1, x_2, \ldots, x_n$ is positive definite if and only if $x_1, x_2, \ldots, x_n$ are linearly independent.*

Proof. □

Definition 3.11 (Symmetric bilinear form)**.** A *symmetric bilinear form* is a map $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ over a vector space $\mathcal{X}$ such that, for all $x, y, z \in \mathcal{X}$, $\alpha, \beta \in \mathbb{R}$,

1. $k(x, y) = k(y, x)$ and (symmetry)
2. $k(\alpha x + \beta y, z) = \alpha k(x, z) + \beta k(y, z)$. (bilinear)

This can be thought of as a generalization of an inner product which is symmetric and bilinear, but not necessarily positive definite.

Example 3.12. If $U = \{u_1, u_2, \ldots, u_n\}$ is a basis for $\mathcal{X}$, then we can define a matrix $K = \left[k(u_i, u_j)\right]_{ij}$. Clearly, K is symmetric since $k(u_i, u_j) = k(u_j, u_i)$. Let $v = \sum_{i=1}^{n} \alpha_i u_i$ and $w = \sum_{i=1}^{n} \beta_i u_i$ be vectors with respect to U and let $x = [\alpha_i]_{i=1}^{n}$ and $y = [\beta_i]_{i=1}^{n}$. Then

$$k(v, w) = k\left(\sum_{i=1}^{n} \alpha_i u_i, \sum_{j=1}^{n} \beta_j u_j\right) = \sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i \beta_j k\left(u_i, u_j\right) = x^\top K y. \tag{19}$$

If $K = I$, then $v = x$, $w = y$, and $k(v, w) = v^\top w$ is simply the dot product. Otherwise, if K is positive semidefinite, then $K = B^\top B$ implies

$$k(v, w) = x^\top B^\top B y = (Bx)^\top (By) \tag{20}$$

In this case, $k(v, w)$ is just the dot product after the transformation under B. Notice that if U is merely a subset of $\mathcal{X}$, then v and w no longer have unique representations, but equations (19) and (20) are still valid for all $v, w \in \operatorname{span} U$.

We say that k is *positive semidefinite* if $K = [k(u_i, u_j)]_{ij}$ is a positive semidefinite matrix for any finite subset $U = \{u_1, u_2, \ldots, u_n\} \subseteq \mathcal{X}$. Then K is a Gram matrix with respect to some set of transformed vectors related to U and some inner product related to k. In the next subsection, we will show that k still corresponds to some inner product even if k is not bilinear.

3.2 Kernels

The development of *kernel functions* can be traced back to the beginning of the twentieth century when David Hilbert and James Mercer were studying integral equations [7]. Hilbert proved some important results in [6] about the eigenvalues of an integral operator whose kernel function is of *definite* type. Expanding on Hilbert's work, Mercer provided the necessary conditions in [11] that allow a kernel function to be written in terms of the eigenvalues and eigenfunctions of the integral operator. This result became known as Mercer's theorem. See **??**.

A simplified version of Mercer's theorem states that a kernel function can be written as an inner product in a higher-dimensional space.

Hilbert space theory and Mercer's theorem led to a number of advances in functional analysis over the next few decades. Notably, in 1950, Nachman Aronszajn introduced reproducing kernel Hilbert spaces in [2]. This work expanded on Mercer's theorem and shows that a kernel generates a Hilbert space whose inner product agrees with the kernel.

Later, the work of Mercer and Aronszajn inspired the application of kernels in machine learning. A *kernel method* is an adaptation of a machine learning algorithm that replaces a dot product with a kernel function. The earliest research involving kernel methods was in 1964 by Mark Aizerman et al. [1]. In the 1990s, Bernhard Schölkopf et al. used Aizerman's technique to develop kernel PCA and suggested the kernel trick could work in other cases too. In Section 4, we will look at the kernel method applied to the PCA algorithm. For now, we will examine the mathematics behind kernel methods.

Definition 3.13 (Kernel)**.** [13] Let $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ be defined on a nonempty set $\mathcal{X}$. Similar to the Gram matrix, define a *kernel matrix* for a set of vectors $\{x_1, x_2, \ldots, x_n\} \subseteq X$ with respect to $k(\cdot, \cdot)$ as $K = [k(x_i, x_j)]_{ij}$. Then k is a *kernel function* (or just *kernel*) if the following hold:

1. $k(x, y) = k(y, x)$, for all $x, y \in \mathcal{X}$ and (symmetry)
2. any kernel matrix K generated by k is positive semidefinite.

We can easily show some properties that kernels have in common with inner products.

Lemma 3.14. *[13] Let k be a kernel. Then the following hold:*

1. *$k(x, x) \geq 0$ for all $x \in \mathcal{X}$ and* *(positive semidefinite)*
2. *$k(x, y)^2 \leq k(x, x)k(y, y)$.* *(Cauchy-Schwarz inequality)*

Proof. Let $x, y \in \mathcal{X}$.

1. The 1×1 kernel matrix $[k(x, x)]$ is positive semidefinite. So, $k(x, x) \geq 0$.
2. The 2×2 kernel matrix

$$K = \begin{bmatrix} k(x,x) & k(x,y) \\ k(y,x) & k(y,y) \end{bmatrix} \tag{21}$$

is positive semidefinite. Let $v = \begin{bmatrix} k(y,y) \\ -k(x,y) \end{bmatrix}$. Then

$$\begin{aligned} 0 &\leq v^\top K v \\ &= \begin{bmatrix} k(y,y) \\ -k(x,y) \end{bmatrix}^\top \begin{bmatrix} k(x,x)k(y,y) - k(x,y)^2 \\ 0 \end{bmatrix} \\ &= k(y,y)\left[k(x,x)k(y,y) - k(x,y)^2\right]. \end{aligned} \tag{22}$$

Then $v^\top K v \geq 0$ implies $k(x,y)^2 \leq k(x,x)k(y,y)$. □

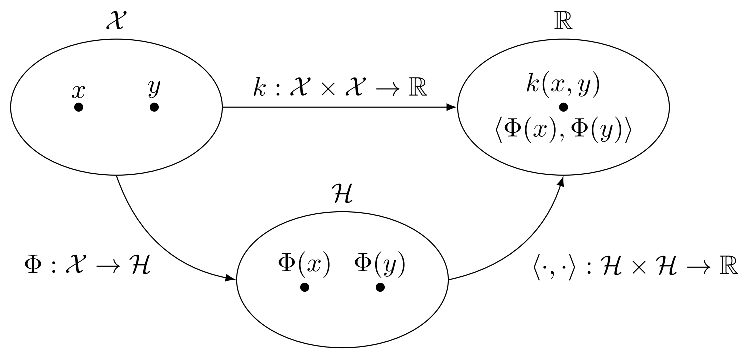

Figure 2: Kernel map diagram.

Definition 3.15 (Feature map)**.** [7] Let $\mathcal{X}$ be a nonempty set and let $\mathbb{R}^{\mathcal{X}}$ be the vector space of real-valued functions on $\mathcal{X}$. A *feature map* is a function $\Phi : \mathcal{X} \to \mathcal{H}$ for some subspace $\mathcal{H} \subseteq \mathbb{R}^{\mathcal{X}}$. In this context, $\mathcal{H}$ is referred to as the *feature space* and its elements $\Phi(x) \in \mathcal{H}$ are called *features*.

Starting with a kernel $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$, we want to construct a feature map $\Phi : \mathcal{X} \to \mathcal{H}$ and inner product $\langle \cdot, \cdot \rangle : \mathcal{H} \times \mathcal{H} \to \mathbb{R}$ which satisfies

$$k(x, y) = \langle \Phi(x), \Phi(y) \rangle , \tag{23}$$

for all $x, y \in \mathcal{X}$. Then the linear span of $\Phi(\mathcal{X}) = \{\Phi(x) \mid x \in \mathcal{X}\}$ will be an inner product space. Since inner product spaces can be completed, we can make this a Hilbert space. See Figure 2.

Constructing a feature map. Consider the map[2] $\Phi(x) = k(x, \cdot)$, for all $x \in \mathcal{X}$. Note that by the symmetry of k, we can write $\Phi(x) = k(x, \cdot) = k(\cdot, x)$. Taking the linear span of $\Phi(\mathcal{X})$ gives us

$$H_0 = \operatorname{span} \Phi(\mathcal{X}) = \left\{ f = \sum_{i=1}^{n} \alpha_i k(x_i, \cdot) \;\middle|\; \begin{array}{r} \forall n \in \mathbb{N}, \\ x_1, x_2, \ldots, x_n \in \mathcal{X}, \\ \alpha_1, \alpha_2, \ldots, \alpha_n \in \mathbb{R} \end{array} \right\}, \tag{24}$$

which forms a subspace of $\mathbb{R}^{\mathcal{X}}$. By Definition 3.15, H_0 is a feature space.

Constructing an inner product. Let $f, g \in H_0$. Then there exist $n, m \in \mathbb{N}$, $(\alpha_i)_{i=1}^n$, $(\beta_j)_{j=1}^m$, $(x_i)_{i=1}^n$, $(y_j)_{j=1}^m$ such that

$$f = \sum_{i=1}^{n} \alpha_i k(x_i, \cdot) \quad \text{and} \quad g = \sum_{j=1}^{m} \beta_j k(y_j, \cdot). \tag{25}$$

[2]Here, $\Phi : \mathcal{X} \to (\mathcal{X} \to \mathbb{R})$ is just the *curried* form of the binary function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$. For example, $\{k(x, \cdot) \mid x \in \mathcal{X}\}$ describes a set of unary functions curried from the binary function k.

Define $\langle \cdot, \cdot \rangle_{H_0} : H_0 \times H_0 \to \mathbb{R}$ as

$$\langle f, g \rangle_{H_0} = \sum_{i=1}^{n} \sum_{j=1}^{m} \alpha_i \beta_j k(x_i, y_j). \tag{26}$$

Letting $m = 1$, $\beta_1 = 1$, $y_1 = y$ in equations (25) and (26), then $g = k(y, \cdot)$. This shows that k has the *reproducing property*

$$\langle f, k(y, \cdot) \rangle_{H_0} = \sum_{i=1}^{n} \alpha_i k(x_i, y) = f(y), \tag{27}$$

for all $y \in \mathcal{X}$. Similarly, letting $n = 1$, $\alpha_1 = 1$, $x_1 = x$, we have

$$k(x, y) = \langle k(x, \cdot), k(y, \cdot) \rangle_{H_0} = \langle \Phi(x), \Phi(y) \rangle_{H_0}, \tag{28}$$

for all $x, y \in \mathcal{X}$. Now we will show that $\langle \cdot, \cdot \rangle_{H_0}$ is an inner product.

1. Since k is symmetric, we have

$$\langle f, g \rangle_{H_0} = \sum_{i=1}^{n} \sum_{j=1}^{m} \alpha_i \beta_j k(x_i, y_j) = \sum_{j=1}^{m} \sum_{i=1}^{n} \beta_j \alpha_i k(y_j, x_i) = \langle g, f \rangle_{H_0}. \tag{29}$$

2. By rearrangement, equation (26) becomes

$$\langle f, g \rangle_{H_0} = \sum_{j=1}^{m} \beta_j \sum_{i=1}^{n} \alpha_i k(x_i, y_j) = \sum_{j=1}^{m} \beta_j f(y_j). \tag{30}$$

   Then for all $f_1, f_2 \in \operatorname{span} \Phi(\mathcal{X})$ and $\alpha, \gamma \in \mathbb{R}$,

$$\begin{aligned} \langle \alpha f_1 + \gamma f_2, g \rangle_{H_0} &= \sum_{j=1}^{m} \beta_j (\alpha f_1 + \gamma f_2)(y_j) \qquad (31) \\ &= \alpha \sum_{j=1}^{m} \beta_j f_1(y_j) + \gamma \sum_{j=1}^{m} \beta_j f_2(y_j) \\ &= \alpha \langle f_1, g \rangle_{H_0} + \gamma \langle f_2, g \rangle_{H_0}. \end{aligned}$$

3. Since k is positive semidefinite, by Lemma 3.14 part 1

$$\langle f, f \rangle_{H_0} = \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j k(x_i, x_j) \geq 0. \tag{32}$$

   Now let $f_1, f_2, \ldots, f_p$ be functions and $\gamma_1, \gamma_2, \ldots, \gamma_p \in \mathbb{R}$. Then by bilinearity,

$$\sum_{i=1}^{p} \sum_{j=1}^{p} \gamma_i \gamma_j \langle f_i, f_j \rangle_{H_0} = \left\langle \sum_{i=1}^{p} \gamma_i f_i, \sum_{j=1}^{p} \gamma_j f_j \right\rangle_{H_0} \geq 0. \tag{33}$$

   Thus, $\langle \cdot, \cdot \rangle_{H_0}$ is a kernel. Then by equation (27) and Lemma 3.14 part 2,

$$f(x)^2 = \langle f, k(x, \cdot) \rangle_{H_0}^2 \leq \langle f, f \rangle_{H_0} \langle k(x, \cdot), k(x, \cdot) \rangle_{H_0}, \tag{34}$$

   for all $x \in \mathcal{X}$. If $\langle f, f \rangle_{H_0} = 0$, then $f(x)^2 = 0$ implies $f = 0$.

Constructing a Hilbert space. Now that we know $\langle \cdot, \cdot \rangle_{H_0}$ is an inner product, H_0 is an inner product space. By [10], this can be completed with respect to the induced metric. Then

$$\mathcal{H} = \overline{\operatorname{span} \Phi(\mathcal{X})} = \overline{\operatorname{span}\{k(x, \cdot) \mid x \in \mathcal{X}\}} \tag{35}$$

is a Hilbert space with inner product $\langle \cdot, \cdot \rangle_{\mathcal{H}} : \mathcal{H} \times \mathcal{H} \to \mathbb{R}$. Since $\mathcal{H}$ contains all its limit points, functions in this space have the form the sequences $(\alpha_i)_{i=1}^{\infty}$ and $(x_i)_{i=1}^{\infty}$ determine a function $f \in \mathcal{H}$ such that

$$f = \sum_{i=1}^{\infty} \alpha_i k(x_i, \cdot), \tag{36}$$

provided the series converges.

Definition 3.16 (Reproducing kernel Hilbert space)**.** [7] Let $\mathcal{H}$ be a Hilbert space of functions $\mathcal{X} \to \mathbb{R}$ on some nonempty set $\mathcal{X}$. Then $\mathcal{H}$ is a *reproducing kernel Hilbert space* (RKHS) if there exists a function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ such that for all $f \in \mathcal{H}$ and $x \in \mathcal{X}$,

1. $f(x) = \langle f, k(x, \cdot) \rangle$ and (reproducing property)
2. $\mathcal{H} = \overline{\operatorname{span}\{k(x, \cdot) \mid x \in \mathcal{X}\}}$. (spanning property)

Fix $y \in \mathcal{X}$ and treat $k(x, y)$ as a univariate function of $x \in \mathcal{X}$. By the reproducing property, $k(x, y) = \langle k(x, \cdot), k(y, \cdot) \rangle$. Define $\Phi(x) = k(x, \cdot)$ to give the desired result in equation (23).

3.3 Related notions of an RKHS

In the previous section, we showed that a (symmetric positive semidefinite) kernel defines an RKHS. Presently, we will look at three alternative methods for constructing an RKHS.

1. **Positive semidefinite kernels.** Due to Aronszajn [2], a reproducing kernel will generate a unique RKHS. Moreover, a kernel is unique to its RKHS. See Theorems 3.17 and 3.18.

2. **Continuous linear functionals.** By the Riesz representation theorem, if every evaluation functional is continuous, then every function in the Hilbert space can be reproduced at every point. In this way, a reproducing kernel can be defined.

3. **Feature maps.** A explicit feature map with an inner product can be used to define a kernel as in equation (23). Alternatively, by Mercer's theorem 3.21, a kernel has a series expansion which allows us to define a feature map in terms of eigenvalues and eigenfunctions.

3.3.1 Positive semidefinite kernels.

Theorem 3.17 (Moore-Aronszajn theorem)**.** *[2] Let $\mathcal{X}$ be a nonempty set and $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ be a kernel. Then there exists a unique RKHS for which k is a reproducing kernel.*

Proof. For existence, we summarize the construction provided in Section 3.2.

1. k defines a feature map $\Phi : \mathcal{X} \to \mathbb{R}^{\mathcal{X}}$ such that $\Phi(x) = k(x, \cdot)$ for all $x \in \mathcal{X}$.
2. The linear span of $\Phi(\mathcal{X})$ is a feature space.
3. Φ defines an inner product $\langle \cdot, \cdot \rangle_{H_0} : H_0 \times H_0 \to \mathbb{R}$ in equation (26).
4. Completing the feature space yields a Hilbert space $\mathcal{H}$ with inner product $\langle \cdot, \cdot \rangle : \mathcal{H} \times \mathcal{H} \to \mathbb{R}$.
5. k has the reproducing property $\langle f, k(x, \cdot) \rangle = f(x)$ shown by equation (27).

For uniqueness, suppose k is a reproducing kernel for Hilbert spaces $\mathcal{H}$ and $\mathcal{L}$.

Now let $f \in \mathcal{L}$. Then we can write $\mathcal{L} = \mathcal{H} \oplus \mathcal{H}^{\perp}$ as the orthogonal decomposition of $\mathcal{L}$. There exist $g \in \mathcal{H}$ and $g^{\perp} \in \mathcal{H}^{\perp}$ such that $f = g + g^{\perp}$. Let $x \in \mathcal{X}$. Then $k(x, \cdot) \in \mathcal{H}$ implies $\left\langle g^{\perp}, k(x, \cdot) \right\rangle_{\mathcal{L}} = 0$ Thus

$$f(x) = \langle g, k(x, \cdot) \rangle_{\mathcal{L}} + \left\langle g^{\perp}, k(x, \cdot) \right\rangle_{\mathcal{L}} = \langle g, k(x, \cdot) \rangle_{\mathcal{L}} = \langle g, k(x, \cdot) \rangle_{\mathcal{H}} = g(x). \quad (37)$$

Thus $f = g \in \mathcal{H}$ implies $\mathcal{L} \subseteq \mathcal{H}$. $\square$

Theorem 3.18. *[2] A reproducing kernel for an RKHS is unique.*

Proof. Let $\mathcal{H}$ be an RKHS of functions $\mathcal{X} \to \mathbb{R}$ for some set $\mathcal{X} \neq \emptyset$. Suppose k and ℓ reproducing kernels for $\mathcal{H}$. Denote $k_x = k(x, \cdot)$ and $\ell_x = \ell(x, \cdot)$, for all $x \in \mathcal{X}$. By the reproducing property,

$$\begin{aligned} \|k_x - \ell_x\|^2 &= \langle k_x - \ell_x, k_x - \ell_x \rangle \qquad (38) \\ &= \langle k_x - \ell_x, k_x \rangle - \langle k_x - \ell_x, \ell_x \rangle \\ &= k_x(x) - \ell_x(x) - k_x(x) + \ell_x(x) \\ &= 0. \end{aligned}$$

It follows that $k_x - \ell_x$ is the zero function. Hence, $k(x, \cdot) = \ell(x, \cdot)$ for all $x \in \mathcal{X}$. By symmetry, $k(\cdot, x) = \ell(\cdot, x)$. Therefore, $k = \ell$. $\square$

3.3.2 Continuous linear functionals.

In the reverse direction, suppose we have a Hilbert space and we want to know if it is an RKHS.

Example 3.19. Consider a Hilbert space $\mathbb{R}^n$ with the dot product. Note that the vectors in $\mathbb{R}^n$ are actually just sequences $\mathbb{N} \to \mathbb{R}$ with some vector operations. It is straightforward to show that the reproducing kernel is the Kroenecker delta $k(i, j) = \delta_{ij}$ for all $i, j \in \{1, 2, \ldots, n\}$. This generates the standard basis $\{e_i\}_{i=1}^n$, where $e_i = k(i, \cdot)$. So, $\mathbb{R}^n$ is an RKHS.

We can replace $\mathbb{N}$ with any other index set with cardinality n, say $I_n = \{\frac{0}{n}, \frac{1}{n}, \frac{2}{n}, \ldots, \frac{n-1}{n-1}\}$. Then an indexed family $f_n : I_n \to \mathbb{R}$ has a vector representation in $\mathbb{R}^n$. Letting n tend to infinity, we have $I_\infty = \mathbb{Q} \cap [0, 1]$. By completion, this is the space of functions $\{f \mid f : [0, 1] \to \mathbb{R}\}$. In one sense, $f_n \in \mathbb{R}^n$ is a point in n-dimensional space and, in another sense, $f_n : I_n \to \mathbb{R}$ is the discretization of a function $f : [0, 1] \to \mathbb{R}$. This way, real-valued functions on $[0, 1]$ can be interpreted as infinite-dimensional vectors.

Now consider the Hilbert space $L^2([0, 1]) = \{f \mid \int_{[0,1]} f^2 < \infty\}$ with inner product $\langle f, g\rangle = \int_{[0,1]} fg$. Then for all $x \in [0, 1]$,

$$f(x) = \int_{[0,1]} \delta(x - t) f(t) \, \mathrm{d}\, t, \tag{39}$$

where δ is the Dirac delta. If $L^2([0, 1])$ is an RKHS, then by Theorem 3.18, $k(x, t) = \delta(x - t)$ is the unique reproducing kernel. But $\int_{\mathcal{X}} \delta^2 = \infty$ implies $\delta \notin L^2([0, 1])$. Therefore, $L^2([0, 1])$ is not an RKHS.

The Kroenecker delta and Dirac delta in the example reproduce functions in the Hilbert space with the inner product.

Definition 3.20 (Evaluation functional)**.** Let $\mathcal{X}$ be a nonempty set and $\mathcal{H}$ be a Hilbert space of functions $\mathcal{X} \to \mathbb{R}$. Then for all $x \in \mathcal{X}$, let $\delta_x : \mathcal{H} \to \mathbb{R}$ such that $\delta_x(f) = f(x)$, for each $f \in \mathcal{H}$.

3.3.3 Mercer's theorem

Mercer's theorem provides a result similar to Aronszajn's, but without the context of an RKHS. Rather, the focus is the decomposition of an integral operator.

Theorem 3.21 (Mercer's theorem)**.** *[11] Let $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ be a continuous bounded kernel on a compact set $\mathcal{X}$. Define the Hilbert-Schmidt integral operator $T_k : L^2(\mathcal{X}) \to L^2(\mathcal{X})$ as*

$$(T_k f)(x) = \int_{\mathcal{X}} k(x, t) f(t) \, \mathrm{d}t. \tag{40}$$

Then there exists an orthonormal basis $\{\psi_i\}_{i=1}^\infty$ of eigenfunctions of T_k and corresponding eigenvalues $(\lambda_i)_{i=1}^\infty$ with $\lambda_i \geq 0$, for all $i \in \mathbb{N}$. Moreover, for all $x, y \in \mathcal{X}$,

$$k(x, y) = \sum_{i=1}^\infty \lambda_i \psi_i(x) \psi_i(y), \tag{41}$$

where convergence is uniform.

Proof. See [4] for a sketch of this proof. Otherwise, this is main result proved in [11]. $\square$

3.4 Constructing kernels

Theorem 3.22. *[13, 18] Suppose k_1 and k_2 are kernels on $\mathcal{X}\times\mathcal{X}$. The following functions kernels.*

1. $k(x,y) = a_1k_1(x,y) + a_2k_2(x,y)$ *for all* $a_1, a_2 \geq 0$.
2. $k(x,y) = k_1(x,y)k_2(x,y)$.
3. $k(x,y) = a_0 + a_1k_1(x,y) + a_2k_1(x,y)^2 + \cdots + a_nk_1(x,y)^n$ *for all* $n \in \mathbb{N}$ *and* $a_0, \ldots, a_n \geq 0$.
4. $k(x,y) = k_1(h(x), h(y))$ *for all* $h : \mathcal{X} \to \mathcal{X}$.
5. $k(x,y) = g(x)g(y)$ *for all* $g : \mathcal{X} \to \mathbb{R}$.
6. $k(x,y) = \exp(k_1(x,y))$.

Proof. Let $x_1, \ldots, x_n \in \mathcal{X}$ and $c_1, \ldots, c_n \in \mathbb{R}$.

1. Let $k = a_1k_1 + a_2k_2$ for $a_1, a_2 \geq 0$. Since k_1 and k_2 are symmetric,

   $$k(x,y) = a_1k_1(x,y) + a_2k_2(x,y) = a_1k_1(y,x) + a_2k_2(y,x) = k(y,x),$$

   for all $x, y \in \mathcal{X}$. So, k is symmetric.

   Since k_1 and k_2 are positive semidefinite and $a_1, a_2 \geq 0$,

   $$\begin{aligned}\sum_{i=1}^{n}\sum_{j=1}^{n} c_ic_jk(x_i,x_j) &= \sum_{i=1}^{n}\sum_{j=1}^{n} c_ic_j(a_1k_1(x_i,x_j) + a_2k_2(x_i,x_j)) \\ &= a_1\sum_{i=1}^{n}\sum_{j=1}^{n} c_ic_jk_1(x_i,x_j) + a_2\sum_{i=1}^{n}\sum_{j=1}^{n} c_ic_jk_2(x_i,x_j) \\ &\geq 0.\end{aligned}$$

   So, k is positive semidefinite.

2. Let $k = k_1k_2$. Define K so that $[K]_{ij} = k(x_i,x_j) = k_1(x_i,x_j)k_2(x_i,x_j)$. Let K_1 and K_2 be the Gram matrices for k_1 and k_2, respectively. Then K_1, K_2 have orthonormal eigenvectors and nonnegative eigenvalues such

that

$$
\begin{aligned}
K_1 &= VLV^\top \\
&= \begin{bmatrix} v_{11} & \cdots & v_{1n} \\ \vdots & \ddots & \vdots \\ v_{n1} & \cdots & v_{nn} \end{bmatrix} \begin{bmatrix} \lambda_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \lambda_n \end{bmatrix} \begin{bmatrix} v_{11} & \cdots & v_{n1} \\ \vdots & \ddots & \vdots \\ v_{1n} & \cdots & v_{nn} \end{bmatrix} \\
&= \begin{bmatrix} \sum_{j=1}^{n} \lambda_j v_{1j} v_{1j} & \cdots & \sum_{j=1}^{n} \lambda_j v_{nj} v_{1j} \\ \vdots & \ddots & \vdots \\ \sum_{j=1}^{n} \lambda_j v_{1j} v_{nj} & \cdots & \sum_{j=1}^{n} \lambda_j v_{nj} v_{nj} \end{bmatrix} \\
&= \sum_{j=1}^{n} \lambda_j \begin{bmatrix} v_{1j} v_{1j} & \cdots & v_{nj} v_{1j} \\ \vdots & \ddots & \vdots \\ v_{1j} v_{nj} & \cdots & v_{nj} v_{nj} \end{bmatrix}
\end{aligned}
$$

and

$$
K_2 = UMU^\top = \sum_{j=1}^{n} \mu_j \begin{bmatrix} u_{1j} u_{1j} & \cdots & u_{nj} u_{1j} \\ \vdots & \ddots & \vdots \\ u_{1j} u_{nj} & \cdots & u_{nj} u_{nj} \end{bmatrix}.
$$

Let $\mathbf{v}_i = \begin{bmatrix} v_{1i} & \cdots & v_{ni} \end{bmatrix}^\top$ and $\mathbf{u}_j = \begin{bmatrix} u_{1j} & \cdots & u_{nj} \end{bmatrix}$, for all $i, j = 1, 2, \ldots, n$. Then

$$
\begin{aligned}
K &= K_1 \circ K_2 \\
&= \sum_{i=1}^{n} \lambda_i \begin{bmatrix} v_{1i} v_{1i} & \cdots & v_{ni} v_{1i} \\ \vdots & \ddots & \vdots \\ v_{1i} v_{ni} & \cdots & v_{ni} v_{ni} \end{bmatrix} \circ \sum_{j=1}^{n} \mu_j \begin{bmatrix} u_{1j} u_{1j} & \cdots & u_{nj} u_{1j} \\ \vdots & \ddots & \vdots \\ u_{1j} u_{nj} & \cdots & u_{nj} u_{nj} \end{bmatrix} \\
&= \sum_{i=1}^{n} \sum_{j=1}^{n} \lambda_i \mu_j \begin{bmatrix} v_{1i} u_{1j} v_{1i} u_{1j} & \cdots & v_{1i} u_{1j} v_{ni} u_{nj} \\ \vdots & \ddots & \vdots \\ v_{ni} u_{nj} v_{1i} u_{1j} & \cdots & v_{ni} u_{nj} v_{ni} u_{nj} \end{bmatrix} \\
&= \sum_{i=1}^{n} \sum_{j=1}^{n} \lambda_i \mu_j \begin{bmatrix} v_{1i} u_{1j} \\ \vdots \\ v_{ni} u_{nj} \end{bmatrix} \begin{bmatrix} v_{1i} u_{1j} & \cdots & v_{ni} u_{nj} \end{bmatrix} \\
&= \sum_{i=1}^{n} \sum_{j=1}^{n} \lambda_i \mu_j (\mathbf{v}_i \circ \mathbf{u}_j)(\mathbf{v}_i \circ \mathbf{u}_j)^\top,
\end{aligned}
$$

where $\circ$ is the Hadamard product. Each $(\mathbf{v}_i \circ \mathbf{u}_j)(\mathbf{v}_i \circ \mathbf{u}_j)^\top$ is a symmetric positive semidefinite matrix. Since K_1, K_2 are positive semidefinite, we have $\lambda_i, \mu_i > 0$. Then K is symmetric positive semidefinite.

3. By part 2, $k_1, k_1^2, \ldots, k_1^n$ are kernels. By part 1, $a_0 + a_1 k_1 + a_2 k_1^2 + \cdots + a_n k_1^n$ is a kernel.

4. Since $y_i = h(x_i) \in \mathcal{X}$ for all $i = 1, 2, \ldots, n$, we have

$$\begin{aligned}\sum_{i=1}^n \sum_{j=1}^n c_i c_j k(x_i, x_j) &= \sum_{i=1}^n \sum_{j=1}^n c_i c_j k_1(h(x_i), h(x_j)) \\ &= \sum_{i=1}^n \sum_{j=1}^n c_i c_j k_1(y_i, y_j) \\ &\geq 0.\end{aligned}$$

5. Let $g : \mathcal{X} \to \mathbb{R}$ and let $c_i g(x_i) = y_i \in \mathbb{R}$. If $k(x, y) = g(x)g(y)$, then

$$\begin{aligned}\sum_{i=1}^n \sum_{j=1}^n c_i c_j k(x_i, x_j) &= \sum_{i=1}^n \sum_{j=1}^n c_i g(x_i) c_j g(x_j) \\ &= \sum_{i=1}^n \sum_{j=1}^n y_i y_j \\ &= \left(\sum_{i=1}^n y_i\right)^2 \\ &\geq 0.\end{aligned}$$

6. Let K_1 be the Gram matrix for k_1. If $K_1 v = \lambda v$, then $K_1^m = \lambda^m v$ for all $m \in \mathbb{N}$. So,

$$(\exp K_1)v = \sum_{m=0}^{\infty} \frac{K_1^m v}{m!} = \sum_{m=0}^{\infty} \frac{\lambda^m v}{m!} = e^{\lambda} v.$$

Then $K = \exp K_1$ has eigenvalues e^{λ}. Since K_1 is positive semidefinite, it has real eigenvalues so that $e^{\lambda} > 0$. It follows that K is positive definite.

□

Theorem 3.23 (Gaussian kernel)**.** *The function $k : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ defined by*

$$k(x, y) = \exp\left(\frac{-\|x - y\|_2^2}{\sigma^2}\right), \tag{42}$$

is a kernel.

Proof. □

4 Kernel PCA

Recall that linear PCA generates new features from a linear combination of the input variables. PCA is an orthogonal projection that rotates the data witin the original space of input variables. These components provide a new basis that may provide more information about the structure of high-dimensional data. The work of Schölkopf, Smola, and Müller [15, 16] generalized PCA based on the successful application of kernel methods in support vector machines by Aizerman [1]. In kernel PCA, the inner product of the input space is replaced with the inner product of a feature space. As such, the principal components, or features, of kernel PCA are nonlinear transformations of input variables.

4.1 Covariance matrix and kernel matrix

Consider a centered data matrix $A = [a_{ij}]^{n \times d}$ whose columns represent input variables. In the linear case, PCA decomposes the covariance matrix $C = \frac{1}{n-1} A^\top A$ into a basis of eigenvectors $\{v_i\}_{i=1}^d$ with descending eigenvalues $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_d$.

Using results from the previous section, a kernel function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ defines a unique reproducing kernel Hilbert space $\mathcal{H}$ and feature map $\Phi : \mathcal{X} \to \mathcal{H}$ such that $k(x, y) = \langle \Phi(x), \Phi(y) \rangle$, for all $x, y \in \mathcal{H}_k$.

4.2 Centering in the feature space

Let $\Phi : \mathcal{X} \to \mathcal{H}_k$ be a feature map determined by a kernel k. Since Φ may be nonlinear, the image $\Phi(x)$ of a centered vector $x \in \mathcal{X}$ is not guaranteed to be centered. For an effective PCA algorithm, it is necessary to compute the kernel matrix of centered vectors in the feature space. [16]

Given $x_1, \ldots, x_n \in \mathcal{X}$, the points

$$\Phi_0(x_i) = \Phi(x_i) - \frac{1}{n} \sum_{i=1}^{n} \Phi(x_i), \quad \text{for } i = 1, \ldots, m \tag{43}$$

are the centered feature vectors in H_k. Then the centered kernel matrix becomes

$$\begin{aligned}
[K_0]_{ij} &= \langle \Phi_0(x_i), \Phi_0(x_j) \rangle \\
&= \left\langle \Phi(x_i) - \frac{1}{n}\sum_{p=1}^{n} \Phi(x_p), \Phi(x_j) - \frac{1}{n}\sum_{q=1}^{n} \Phi(x_q) \right\rangle \\
&= \langle \Phi(x_i), \Phi(x_j) \rangle \; - \frac{1}{n}\sum_{p=1}^{n} \langle \Phi(x_p), \Phi(x_j) \rangle \\
&\qquad\qquad\qquad\quad - \frac{1}{n}\sum_{q=1}^{n} \langle \Phi(x_i), \Phi(x_q) \rangle \\
&\qquad\qquad\qquad\quad + \frac{1}{n^2}\sum_{p=1}^{n}\sum_{q=1}^{n} \langle \Phi(x_p), \Phi(x_q) \rangle \\
&= [K]_{ij} - \frac{1}{n}\sum_{p=1}^{n} [K]_{pj} - \frac{1}{n}\sum_{q=1}^{n} [K]_{iq} + \frac{1}{n^2}\sum_{p=1}^{n}\sum_{q=1}^{n} [K]_{pq},
\end{aligned}$$

where K is the uncentered kernel matrix given by $[K]_{ij} = \langle \Phi(x_i), \Phi(x_j) \rangle$. Then the formula for the centered kernel matrix can be written as

$$K_0 = K - \operatorname{col\,mean}(K) - \operatorname{row\,mean}(K) + \operatorname{mean}(K). \tag{44}$$

See notes in Appendices A.1 and A.2.

4.3 Kernel PCA algorithm

Let $x_1, x_2, \dots, x_n \in \mathbb{R}^d$ be input vectors and $k : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ be a kernel. The kernel PCA algorithm outputs the transformed input vectors $\tilde{x}_1, \tilde{x}_2, \dots, \tilde{x}_n \in \mathbb{R}^n$.

1. Compute the kernel matrix $K = [k(x_i, x_j)]_{ij}^{n \times n}$.
2. Center kernel matrix $K_0 = K - \operatorname{col\,mean}(K) - \operatorname{row\,mean}(K) + \operatorname{mean}(K)$.
3. Compute eigenvalues $(\lambda_i)_{i=1}^n$ and eigenvectors $(\alpha_i)_{i=1}^n$ of K_0 so that $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$.

Example 4.1. Consider the problem of classifying points based on their radii. These points cannot be separated using a linear classifier in the two dimensions. However, by mapping them to a three-dimensional space, they can be separated by planes. Applying kernel PCA, these points can be sent to the RKHS associated with a Gaussian kernel without using an explicit feature map. The points in this high-dimensional feature space can then be projected onto the first three principal components to find separation boundaries. See Figure 3.

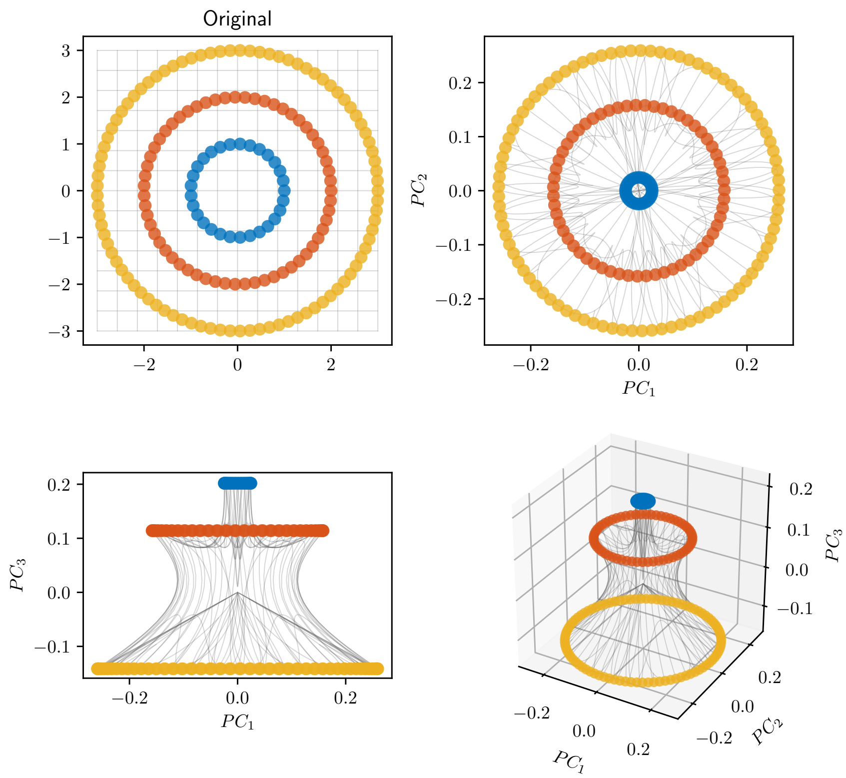

Figure 3: An idealized set of points in the plane are classified based on their radius. Kernel PCA with the Gaussian kernel is applied to find separation boundaries using the first three principal components.

5 Conclusion

A Linear Algebra

A number of matrix definitions and results are presented without proof. These can be found in [8].

1. A square matrix A is *normal* if $AA^\top = A^\top A$.
2. Symmetric matrices are normal.
3. Symmetric matrices have orthogonal eigenvectors and real eigenvectors.
4. Positive semidefinite matrices have nonnegative eigenvalues.
5. A is positive semidefinite if and only if there exists a matrix B such that $A = B^\top B$. We say B is the *square root* of A and write $A^{1/2} = B$.
6. Positive definite matrices have positive eigenvalues.
7. $A^\top A$ and $AA^\top$ are symmetric positive semi-definite.

Lemma A.1. *Let A be a positive semidefinite matrix. Then A has the factorization $A = B^\top B$. We call*

Proof. Since A is symmetric, it is diagonalizable and we can write $A = V^\top DV$. Since A is positive semidefinite, □

A.1 Matrix operations and notation

Let A be an $n \times d$ matrix. We write $[A]_{ij}$ to indicate the matrix entry in the i-th row and the j-th column.

Definition A.2. Define the *entry-wise mean* of A as

$$\operatorname{mean}(A) = \frac{1}{nd} \sum_{i=1}^{n} \sum_{j=1}^{d} [A]_{ij}. \tag{45}$$

Define the *column-wise mean* of A as a $1 \times d$ row vector whose j-th entry is the mean of column j given by the formula

$$\begin{aligned} \operatorname{col\,mean}(A) &= \left[\tfrac{1}{n} \textstyle\sum_{i=1}^{n} [A]_{i1}, \quad \tfrac{1}{n} \sum_{i=1}^{n} [A]_{i2}, \quad \ldots, \quad \tfrac{1}{n} \sum_{i=1}^{n} [A]_{id} \right] \\ &= \frac{1}{n} \sum_{i=1}^{n} \left[[A]_{i1}, \quad [A]_{i2}, \quad \ldots, \quad [A]_{id} \right]. \end{aligned} \tag{46}$$

Define the *row-wise mean* of A as an $n \times 1$ column vector whose i-th entry is the mean of row i given by the formula

$$\operatorname{row\,mean}(A) = \begin{bmatrix} \frac{1}{d}\sum_{j=1}^{d}[A]_{1j} \\ \frac{1}{d}\sum_{j=1}^{d}[A]_{2j} \\ \vdots \\ \frac{1}{d}\sum_{j=1}^{d}[A]_{nj} \end{bmatrix} = \frac{1}{d}\sum_{j=1}^{d} \begin{bmatrix} [A]_{1j} \\ [A]_{2j} \\ \vdots \\ [A]_{nj} \end{bmatrix}. \tag{47}$$

Let $[a]_{p\times q}$ denote the $p \times q$ repeated matrix whose entries are all a. Then equations (45) to (47) can be written as

$$\operatorname{mean}(A) = \left[\tfrac{1}{n}\right]_{1\times n} \cdot A \cdot \left[\tfrac{1}{d}\right]_{d\times 1} \tag{48}$$

$$\operatorname{col\,mean}(A) = \left[\tfrac{1}{n}\right]_{1\times d} \cdot A \tag{49}$$

$$\operatorname{row\,mean}(A) = A \cdot \left[\tfrac{1}{d}\right]_{d\times 1}. \tag{50}$$

A.2 Broadcasting

Consider the sum of two real matrices $A + B$. By definition, A and B must both have size $n \times d$. This means we cannot add a 2×3 matrix A and a 2×1 vector $\mathbf{b}$. However, in many programming languages the sum $A + \mathbf{b}$ would be handled using *broadcasting* [5]. In this case, $\mathbf{b}$ is converted to a 2×3 matrix $\begin{bmatrix}\mathbf{b} & \mathbf{b} & \mathbf{b}\end{bmatrix}$ so that normal matrix addition applies. Generally, broadcasting a vector $\mathbf{b} \in \mathbb{R}^n$ to an $n \times d$ matrix can be represented as the matrix product

$$\mathbf{b} \cdot [1]_{1\times d} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix} \cdot \begin{bmatrix} 1 & 1 & \cdots & 1 \end{bmatrix} = \begin{bmatrix} b_1 & b_1 & \cdots & b_1 \\ b_2 & b_2 & \cdots & b_2 \\ \vdots & \vdots & & \vdots \\ b_n & b_n & \cdots & b_n \end{bmatrix}, \tag{51}$$

where the notation $[a]_{n\times d}$ represents an $n \times d$ matrix whose entries are all a.

Definition A.3. For an $n \times d$ matrix A, we can define addition by an $n \times 1$ column vector $\mathbf{c}$ as

$$A + \mathbf{c} := A + \mathbf{c} \cdot [1]_{1\times d}. \tag{52}$$

Similarly, addition by a $1 \times d$ row vector $\mathbf{r}$ can be defined as

$$A + \mathbf{r} := A + [1]_{n\times 1} \cdot \mathbf{r} \tag{53}$$

and addition by a scalar a can be defined as

$$A + a := A + a \cdot [1]_{n\times d}. \tag{54}$$

The left hand sides of equations (52) to (54) are more concise and intuitive than the right hand sides. Provided that the vector types are clearly defined and compatible, there should be no ambiguity when adding column vectors, row vectors, and scalars to matrices. Moreover, this method of broadcasting is consistent with scientific programming languages.

B Code

Listing 1: PCA example

```python
import numpy as np
from pca import *

# Random number generator for repeatability
rng = np.random.default_rng(12)
# Set data dimensions
d = 3  # number of columns (variables)
n = 30 # number of rows (observations)
# Generate uncorrelated and correlated data
A = rng.standard_normal((n,d)) # uncorrelated
B = A.copy() # correlated
B[:,0] += 4*B[:,1] + 2*B[:,2]  # correlated
# PCA projections
coeff, score, latent, mu = pca(A, n_components=2)
Ahat = score @ coeff.T + mu # reconstruct uncorrelated data
coeff, score, latent, mu = pca(B, n_components=2)
Bhat = score @ coeff.T + mu # reconstruct correlated data
# Reconstruction error
err_uncorr = np.linalg.norm(A - Ahat)
err_corr = np.linalg.norm(B - Bhat)
print(f"Uncorrelated reconstruction error: {err_uncorr}")
print(f"Correlated reconstruction error: {err_corr}")
# Plot Scatter
pairwise3dscatter(A, Ahat, "Uncorrelated Data")
pairwise3dscatter(B, Bhat, "Correlated Data")
```

Listing 2: PCA source functions

```python
import numpy as np
import matplotlib.pyplot as plt

def pca(data, n_components=None):
    '''
    Perform principal component analysis.

    Parameters
    ----------
    data: array_like
        Input data.
    n_components: int
        Number of principal components to keep.

    Returns
    -------
    V: array_like
        Principal component vectors (aka coeff).
```

```
    A: array_like
        Transformed datta (aka score).
    D: array_like
        Explained variance for each principal component (aka latent).
    '''
    # Copy data as numpy array
    A = np.copy(data)
    # Center data matrix
    col_mean = A.mean(0)
    A -= col_mean
    # Compute covariance matrix
    C = A.T @ A / (A.shape[0]-1)       # C = A'A/(n-1)
    # Get eigenvalues D and eigenvectors V
    D, V = np.linalg.eigh(C)
    # Assert sign convention for eigenvectors
    V *= np.sign(V.min(0) + V.max(0)) # change sign where |min|>|max|
    # Sort eigenvalues and eigenvectors
    sort_index = D.argsort()[::-1]     # descending
    D = D[sort_index]                  # sort eigenvalues
    V = V[:,sort_index]                # sort eigenvectors by columns
    # Get principal component coefficient matrix
    V = V if n_components is None else V[:,:n_components]
    # Transform data
    A = A @ V
    return V, A, D, col_mean

def pairwise3dscatter(A, B, title=None):
    '''
    Helper function for pairwise plots.
    '''
    fig = plt.figure(figsize=(4,4))
    fig.suptitle(title)
    ax = [
        fig.add_subplot(2, 2, 2, projection='3d'),
        fig.add_subplot(2, 2, 1),
        fig.add_subplot(2, 2, 3),
        fig.add_subplot(2, 2, 4)
    ]
    a1, a2, a3 = A.T
    b1, b2, b3 = B.T
    ax[0].scatter(a1, a2, a3, alpha=.5)
    ax[0].scatter(b1, b2, b3, alpha=.5)
    ax[1].scatter(a1, a2, alpha=.5)
    ax[1].scatter(b1, b2, alpha=.5)
    ax[2].scatter(a1, a3, alpha=.5)
    ax[2].scatter(b1, b3, alpha=.5)
    ax[3].scatter(a2, a3, label="Original", alpha=.5)
    ax[3].scatter(b2, b3, label="Reconstructed", alpha=.5)
    ax[1].set_ylabel("$x_2$")
    ax[2].set_ylabel("$x_3$")
```

```
    ax[2].set_xlabel("$x_1$")
    ax[3].set_xlabel("$x_2$")
    ax[3].legend()
    plt.show()
```

References

[1] Mark Aronovich Aizerman, Emmanuel Markovich Braverman, and Lev Ilyich Rozonoer. Theoretical foundations of the potential function method in pattern recognition learning. *Automation and remote control*, 25:821–837, 1964.

[2] Nachman Aronszajn. Theory of reproducing kernels. *Transactions of the American mathematical society*, 68(3):337–404, 1950.

[3] G. Bachman and L. Narici. *Functional Analysis*. Number v. 10 in Academic Press textbooks in mathematics. Academic Press, 1966.

[4] Benyamin Ghojogh, Ali Ghodsi, Fakhri Karray, and Mark Crowley. Reproducing kernel Hilbert space, Mercer's theorem, eigenfunctions, Nyström method, and use of kernels in machine learning: Tutorial and survey. *arXiv preprint arXiv:2106.08443*, 2021.

[5] Charles R Harris, K Jarrod Millman, Stéfan J Van Der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J Smith, et al. Array programming with numpy. *Nature*, 585(7825):357–362, 2020.

[6] D. Hilbert. *Grundzüge einer allgemeinen Theorie der linearen Integralgleichungen*. Cornell University Library historical math monographs. B. G. Teubner, 1912.

[7] Thomas Hofmann, Bernhard Schölkopf, and Alexander J. Smola. Kernel methods in machine learning. *The Annals of Statistics*, 36(3), jun 2008.

[8] R.A. Horn and C.R. Johnson. *Matrix Analysis*. Cambridge University Press, 2013.

[9] K. Koutroumbas and S. Theodoridis. *Pattern Recognition*. Elsevier Science, 2008.

[10] E. Kreyszig. *Introductory Functional Analysis with Applications*. Wiley Classics Library. Wiley, 1991.

[11] James Mercer. Xvi. functions of positive and negative type, and their connection the theory of integral equations. *Philosophical transactions of the royal society of London. Series A, containing papers of a mathematical or physical character*, 209(441-458):415–446, 1909.

[12] Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. *Foundations of Machine Learning*. Adaptive Computation and Machine Learning series. MIT Press, 2012.

[13] Cynthia Rudin. Intuition for the Algorithms of Machine Learning. Self-pub, 2020.

[14] Walter Rudin. *Real and Complex Analysis*. Higher Mathematics Series. McGraw-Hill Education, 1987.

[15] Bernhard Schölkopf, Alexander Smola, and Klaus-Robert Müller. Kernel principal component analysis. In *International conference on artificial neural networks*, pages 583–588. Springer, 1997.

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

[17] Cosma Rohilla Shalizi. Advanced Data Analysis from an Elementary Point of View. Draft textbook, 2021.

[18] John Shawe-Taylor and Nello Cristianini. *Kernel Methods for Pattern Analysis*. Cambridge University Press, 2004.

[19] Christopher G. Small and D.L. Mcleish. *Hilbert Space Methods in Probability and Statistical Inference*. John Wiley & Sons, Inc, 1994.