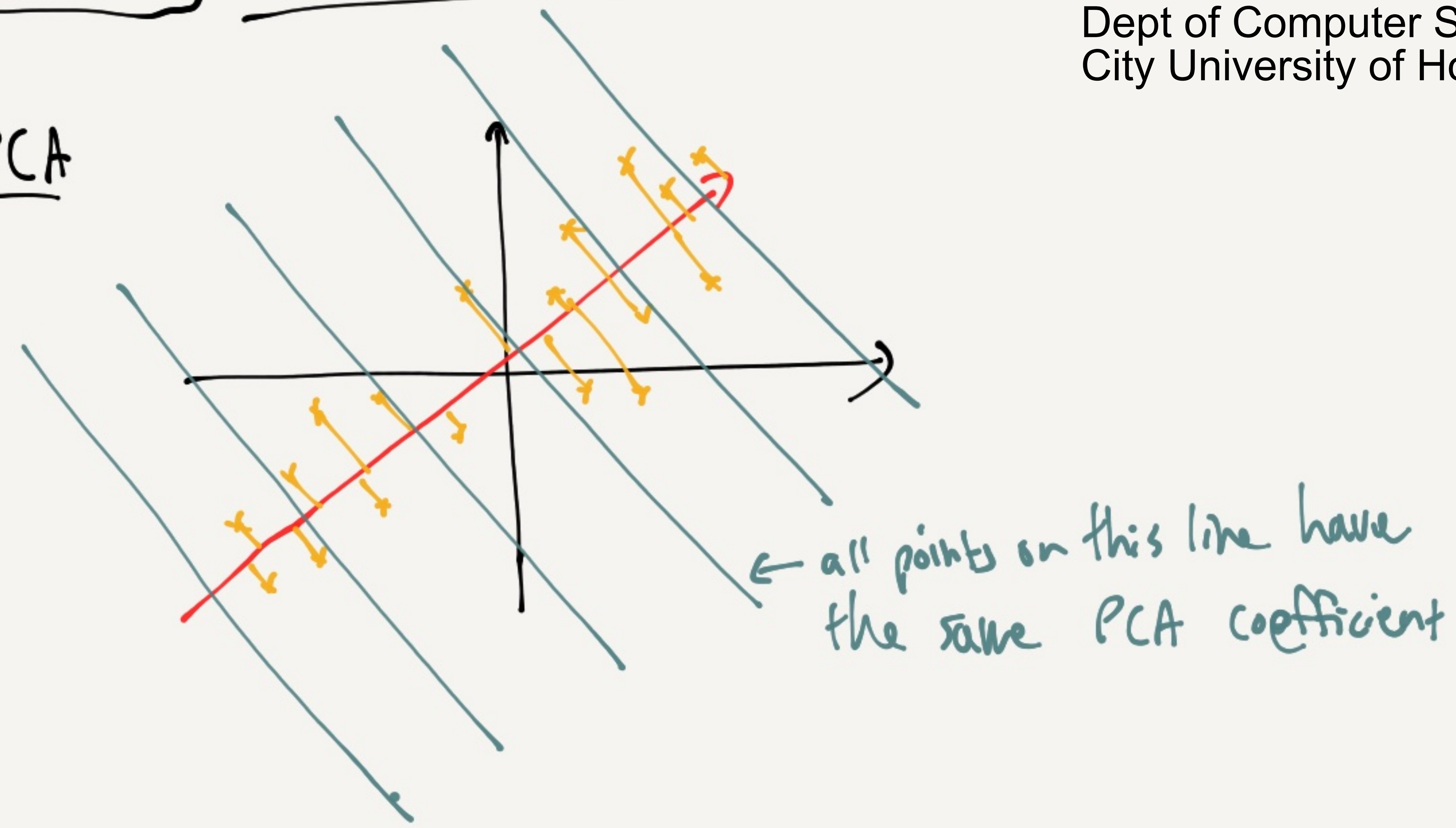
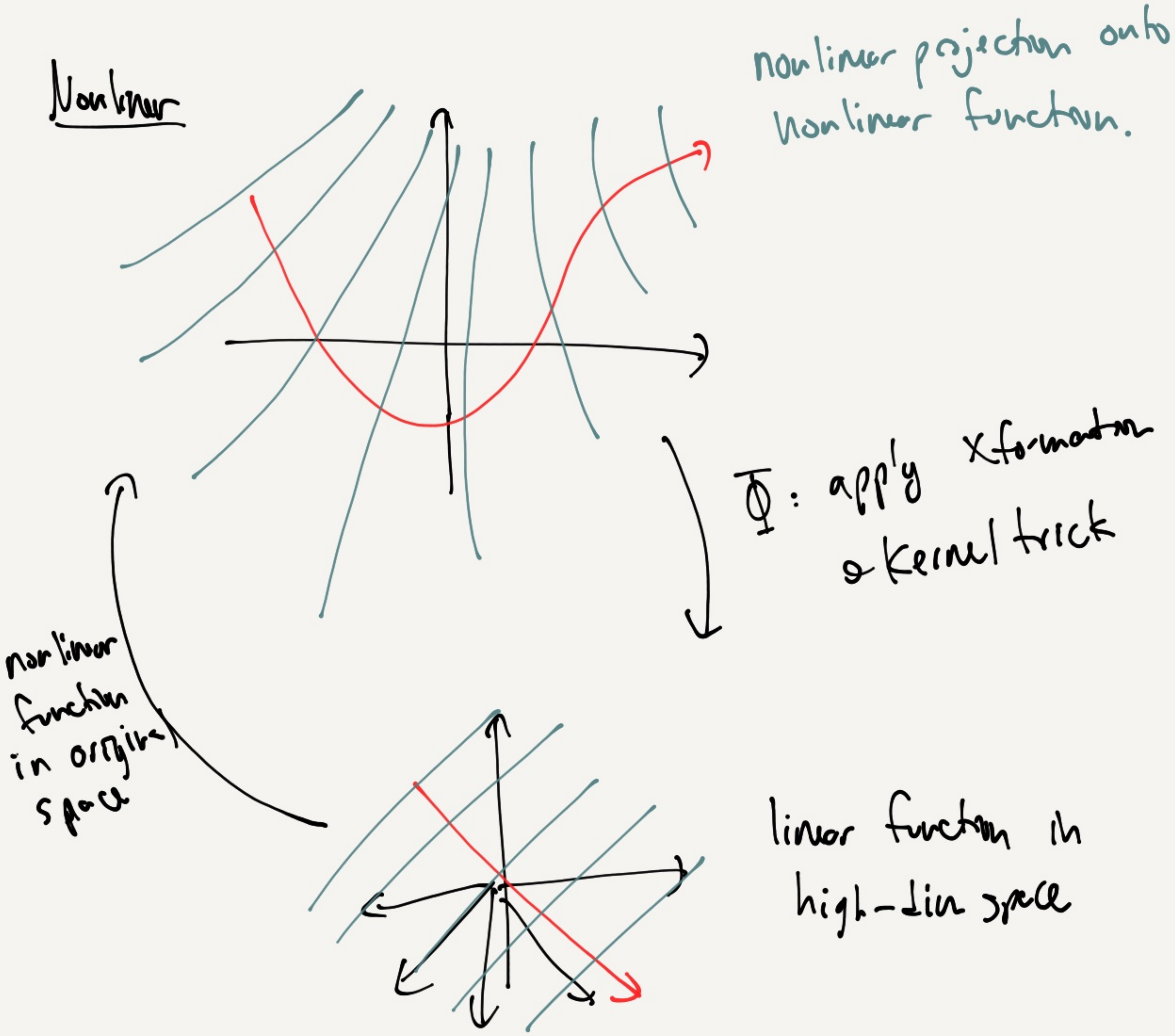
Lecture 11 - Nonlinear Dim. Reduction.

CS5487 Lecture Notes (2022B)
Prof. Antoni B. Chan
Dept of Computer Science
City University of Hong Kong

PCA



Nonlinear



Kernel PCA:

Data: $\{x_1, \dots, x_N\}$

Apply feat xform: $x_i \to \phi(x_i)$

$[x_1, \dots, x_N] = X \to \Phi = [\phi(x_1) \dots \phi(x_N)]$

- Assume xformed data is centered: $\sum_i \phi(x_i) = 0$
- Covariance matrix in high-dim space:

$$C = \frac{1}{N} \sum_{i=1}^{N} \phi(x_i)\phi(x_i)^T$$

(Note: zero mean)

- Find (λ_j, v_j) eigenvalue/eigenvectors.

$$C v_j = \lambda_j v_j$$

$$\frac{1}{N} \sum_{m=1}^{N} \phi(x_m) \underbrace{\phi(x_m)^T v_j}_{a_{m,j}} = \lambda_j v_j$$

Hence, v_j has the form $v_j = \sum_{i=1}^{N} a_{ij} \phi(x_i) = \Phi a_j$

where $a_j = \begin{bmatrix} a_{1j} \\ \vdots \\ a_{Nj} \end{bmatrix}$

i.e. eigenvector is a linear combo of $\phi(x_i)$'s.

we need to find a_j

Substitute for $v_j = \Phi a_j$

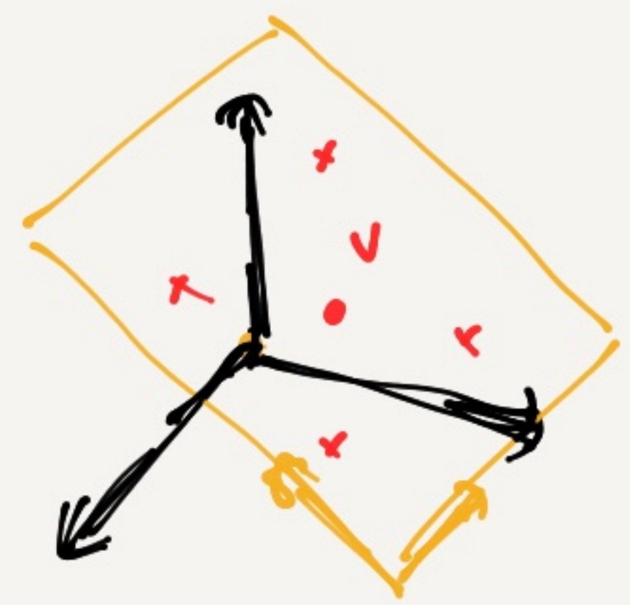
$$\frac{1}{N}\sum_n \underbrace{\phi(x_n)\,\phi(x_n)^T}\,\Phi a_j = \lambda_j \Phi a_j$$

$$\frac{1}{N}\Phi\underbrace{\Phi^T\Phi a_j} = \lambda_j \underbrace{\Phi a_j}$$

eigenvector is linear combo of

$\phi(x_i) = \mathrm{span}(\phi(x_1) \dots, \phi(x_N))$

An equivalent set of eqns is to project into the coordinates of span of Φ, & solve there.



$\Rightarrow$ premultiply by Φ^T

$$\Rightarrow \frac{1}{N}\underbrace{\Phi^T\Phi}_{K}\underbrace{\Phi^T\Phi}_{K} a_j = \lambda_j \underbrace{\Phi^T\Phi}_{K} a_j$$

kernel trick

$$\Rightarrow \boxed{KKa_j = N\lambda_j Ka_j}$$

① if K is invertible, then solve:

$$\boxed{Ka_j = N\lambda_j a_j}$$

② if K is not invertible, the only difference is there eigenvectors w/ $\lambda_j = 0$, but these are not PC's because $\lambda = 0$. Solve ① anyways.

Verify: suppose (a_j, λ_j) are eigenvector/value of K s.t. $Ka_j = N\lambda_j a_j$, $\lambda_j \neq 0$, $a_j^T a_j = 1$

original eqn:

$$K\underbrace{Ka_j}_{N\lambda_j a_j} = N\lambda_j \underbrace{Ka_j}_{N\lambda_j a_j}$$

$$N\lambda_j \underbrace{Ka_j}_{N\lambda_j a_j} = \underbrace{N\lambda_j N\lambda_j a_j}_{N^2\lambda_j^2 a_j}$$

$$N^2\lambda_j^2 a_j = N^2\lambda_j^2 a_j \quad \checkmark$$

PC should be normalized

$$\underline{v_j^T v_j = 1} = a_j^T \Phi^T \Phi a_j = a_j^T K a_j$$

$$= N\lambda_j a_j^T a_j$$

Thus, rescale

$$a_j \leftarrow \frac{1}{\sqrt{N\lambda_j}} a_j$$

Kernel Centering

How to center the feature space?

centered feature:

$$\tilde{\phi}(x_i) = \phi(x_i) - \underbrace{\frac{1}{N}\sum_{k=1}^{N}\phi(x_k)}_{\text{mean in f.s.}}$$

$$= \phi(x_i) - \frac{1}{N}\Phi\mathbf{1}$$

$\mathbf{1}$ vector $= \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}$

centered data:

$$\tilde{\Phi} = \Phi - \left[\frac{1}{N}\Phi\mathbf{1}\right]\mathbf{1}^T = \Phi - \frac{1}{N}\Phi\mathbf{1}\mathbf{1}^T = \Phi\left(I - \frac{1}{N}\mathbf{1}\mathbf{1}^T\right)$$

Centered kernel matrix

$$\tilde{K} = \tilde{\Phi}^T\tilde{\Phi} = \left(I - \frac{1}{N}\mathbf{1}\mathbf{1}^T\right)^T \underbrace{\Phi^T\Phi}_{K}\left(I - \frac{1}{N}\mathbf{1}\mathbf{1}^T\right)$$

$$\boxed{\tilde{K} = \left(I - \frac{1}{N}\mathbf{1}\mathbf{1}^T\right)K\left(I - \frac{1}{N}\mathbf{1}\mathbf{1}^T\right)}$$

test kernel:

$$\tilde{k}(x_*, x_j) = k(x_*, x_j) - \frac{1}{N}k_*^T\mathbf{1} - \frac{1}{N}k_j^T\mathbf{1} + \frac{1}{N^2}\mathbf{1}^TK\mathbf{1}$$

j^{th} row of K

Summary: KPCA

1) Calculate kernel matrix: $K = [k(x_i, x_j)]_{ij}$
2) center the kernel: $\tilde{K} = (I - \frac{1}{N}\mathbf{1}\mathbf{1}^T)K(I - \frac{1}{N}\mathbf{1}\mathbf{1}^T)$
3) Find the top D eigenvectors: $\tilde{K}a_j = \lambda_j a_j$, $j = 1 \ldots D$
4) Scale: $a_j \leftarrow \frac{1}{\sqrt{\lambda_j}}a_j$
5) project data x_*: $z_{*j} = \tilde{k}_*^T a_j$

$$z_{*j} = \sum_{i=1}^{N} a_{i,j}\tilde{k}(x_*, x_i)$$

Note: original problem needs d-dim eigenvector
kernel problem " N-dim eigenvector

Pick problem that is more efficient.

PS 10-15: Kernelized FLD $\rightarrow$ Kernel disc. analysis.

Pre-image Problem

Given PCA coeff z, we can reconstruct $\hat{x}$, e.g. denoising x.

$$\hat{x} = \sum_j v_j z_j$$

What about KPCA?

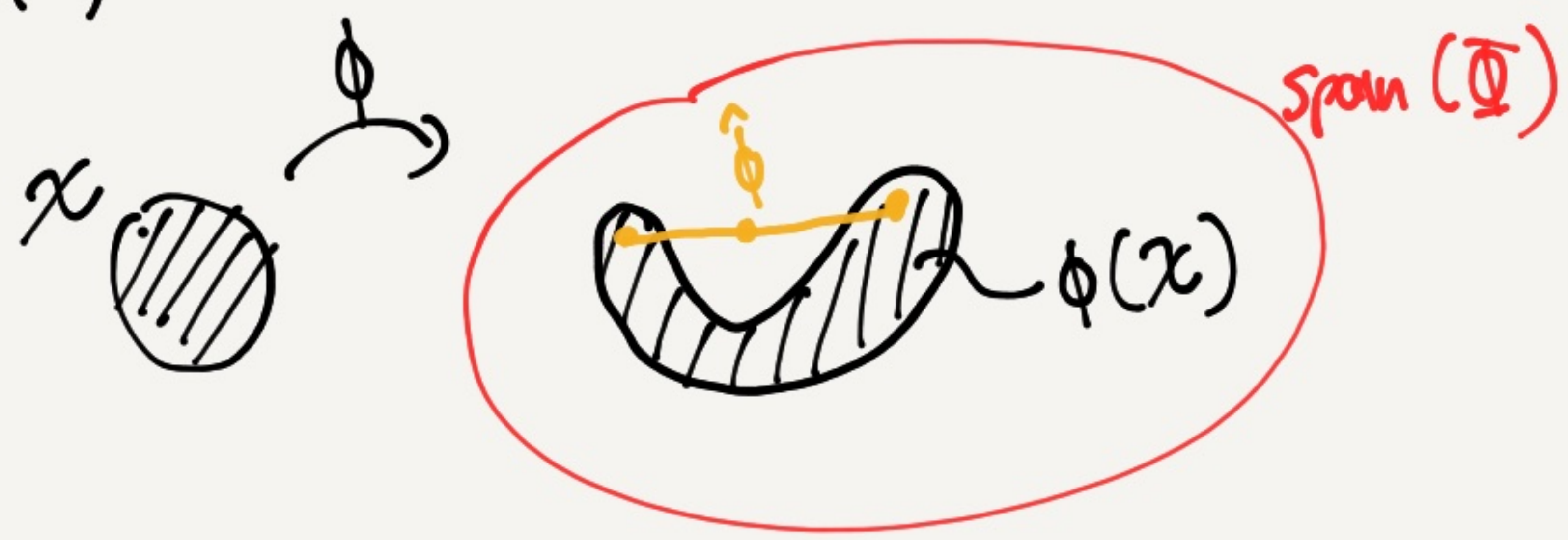
- Given z, what is the high-dim feature?

$$\hat{\phi} = \sum_j v_j z_j = \sum_j \Phi a_j z_j = \Phi\Big(\sum_j a_j z_j\Big)$$

linear combo of $\phi(x_i)$

$\hat{\phi} \in \mathrm{span}(\Phi)$

BUT, not all points $\mathrm{span}(\Phi)$ have a corresponding x!



Example: let $\phi(x) = \begin{bmatrix} x^2 \\ x \end{bmatrix}$

$x_1 = 1 \Rightarrow \phi(x_1) = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$

$x_2 = 2 \Rightarrow \phi(x_2) = \begin{bmatrix} 4 \\ 2 \end{bmatrix}$

let $\hat{\phi} = \phi(x_1) + \phi(x_2) = \begin{bmatrix} 5 \\ 3 \end{bmatrix}$, what is $\hat{x}$ s.t. $\hat{\phi} = \phi(\hat{x})$?

there is no $\hat{x}$ possible.

Approximate Pre-Image

general problem: $\hat{\phi} = \Phi \alpha = \sum_i \alpha_i \phi(x_i)$

given

Find: $\hat{x} = \arg\min_x \|\phi(x) - \hat{\phi}\|^2$ ← find the x that gives the closest ϕ in the f.s.

$$\hat{x} = \arg\min_x k(x,x) - 2\sum_i \alpha_i k(x,x_i) + \underbrace{\sum_i \sum_j \alpha_i \alpha_j k(x_i,x_j)}_{\times}$$

$$\boxed{\hat{x} = \arg\min_x k(x,x) - 2\sum_i \alpha_i k(x,x_i)}$$

Soln 1: nearest neighbors
select $x \in X$ (training data)

Soln 2: solve optimization problem numerically w/ package.

Soln 3: Suppose $k(x,x) = 1$, $k(x_i,x_j) \geq 0$ (e.g. Gaussian)

$$\hat{x} = \arg\min_x -\sum_i \alpha_i k(x,x_i) = \arg\max_x \underbrace{\sum_i \alpha_i k(x,x_i)}_{\langle \phi(x), \hat{\phi} \rangle} = \arg\max_x \Big(\sum_i \alpha_i k(x,x_i)\Big)^2$$

assume homogeneous kernel, $k(\|x - x'\|^2)$

iterative algo.

$$\boxed{\hat{x} \leftarrow \frac{\sum_i \alpha_i k'(\|x_i - \hat{x}\|^2) x_i}{\sum_i \alpha_i k'(\|x_i - \hat{x}\|^2)}}$$

Similar to mean-shift w/ weights α_i on each point x_i