# PCA, kernel PCA, Laplacian eigenmaps

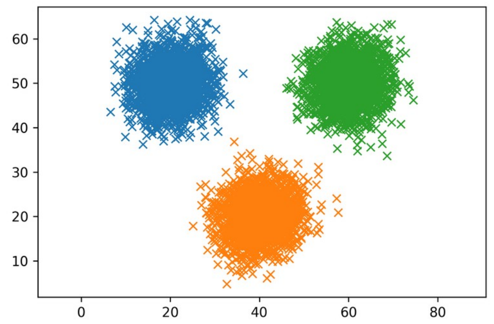
- Feature selection / generation without using labels

# PCA
## (Karhunen-Loève transform)

Samples

$$x=\begin{pmatrix} x_1 \\ x_2 \\ . \\ . \end{pmatrix}=\begin{pmatrix} x_1(1) & x_1(2) & \dots & x_1(N) \\ x_2(1) & x_2(2) & \dots & . \\ . & . & \dots & . \\ . & . & \dots & x_K(N) \end{pmatrix}$$

Features




Normalize the samples and write the covariance matrix $y_i = x_i - \mu_i$

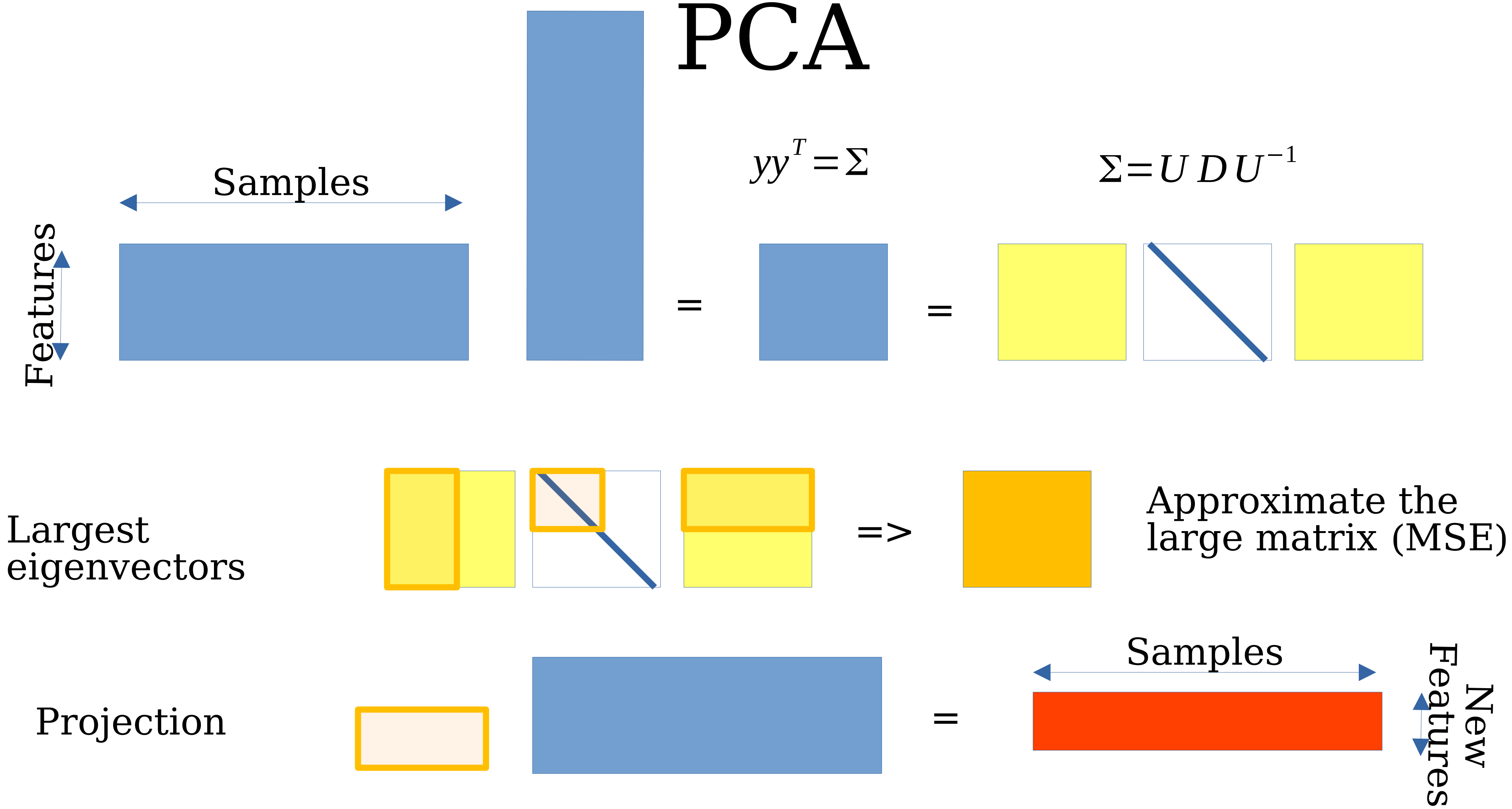
$$yy^T = \Sigma$$

Diagonalize the covariance matrix

$$\Sigma = U\,D\,U^{-1}$$

**Note:** no label information used

# PCA

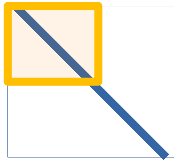

Samples
Features
$yy^T = \Sigma$
$\Sigma = U\,D\,U^{-1}$
=
=
Largest
eigenvectors
=>
Approximate the
large matrix (MSE)
Projection
=
Samples
New
Features

# PCA

$$Y = A^T X$$

Maximizing the covariance

$$YY^T = \Sigma_y = A^T XX^T A = A^T \Sigma_x A$$

If $A = U$ Matrix of eigenvectors, $\Sigma_y =$ 

- Zero covariance,
- maximal variance with largest eigenvalues

**Note:** $\underset{w}{argmax} \frac{w^T M w}{w^T w}$ Solution: w eigenvector of M with largest eigenvalue

Min-max (Courant-Fisher-Weyl) theorem

# PCA

- Some visual examples

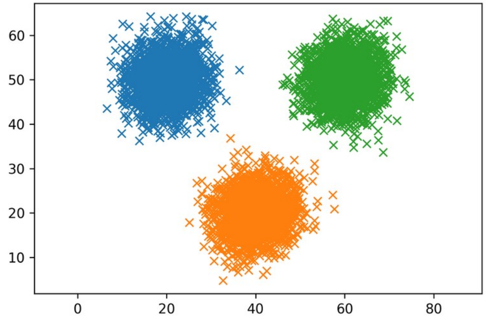
# LDA & PCA

- Remarks: complex mix of initial features, can loose interpretability / explainability

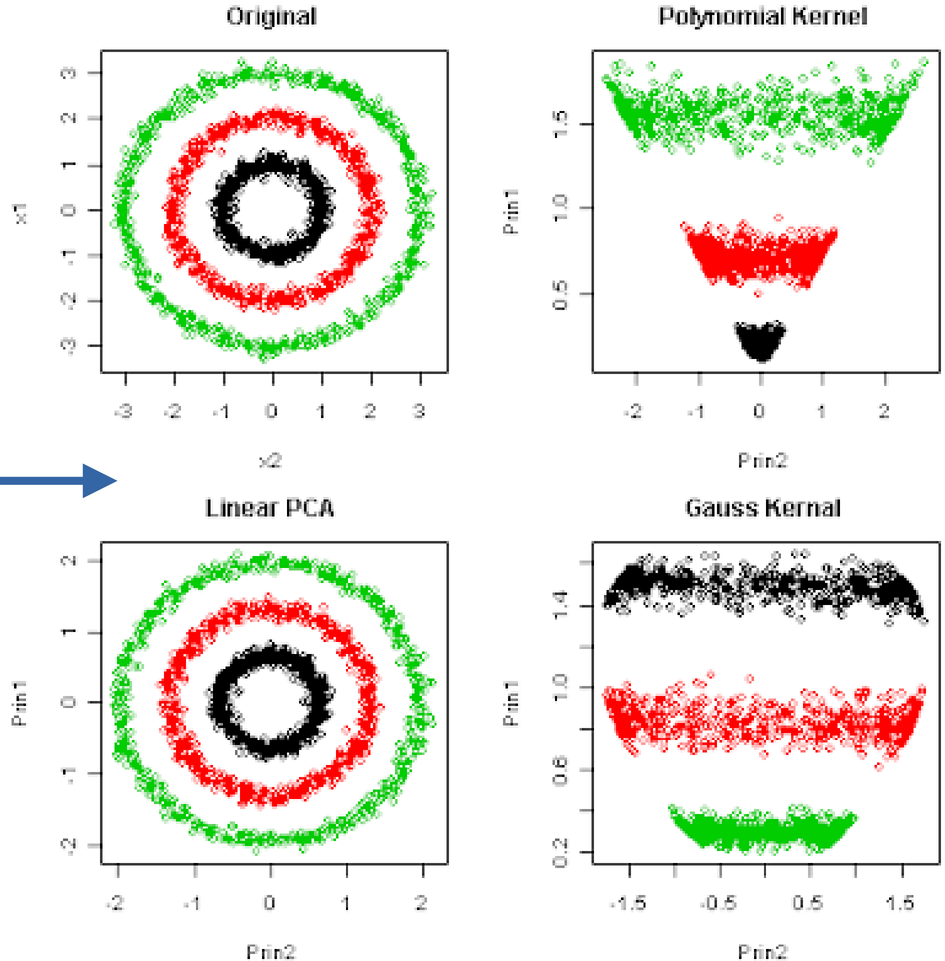
  what the important features? Difficult to answer

- Only linear separability

# Kernel PCA




Data not linearly separable:
Send the data in a higher dimensional space, nonlinearly



# Kernel PCA

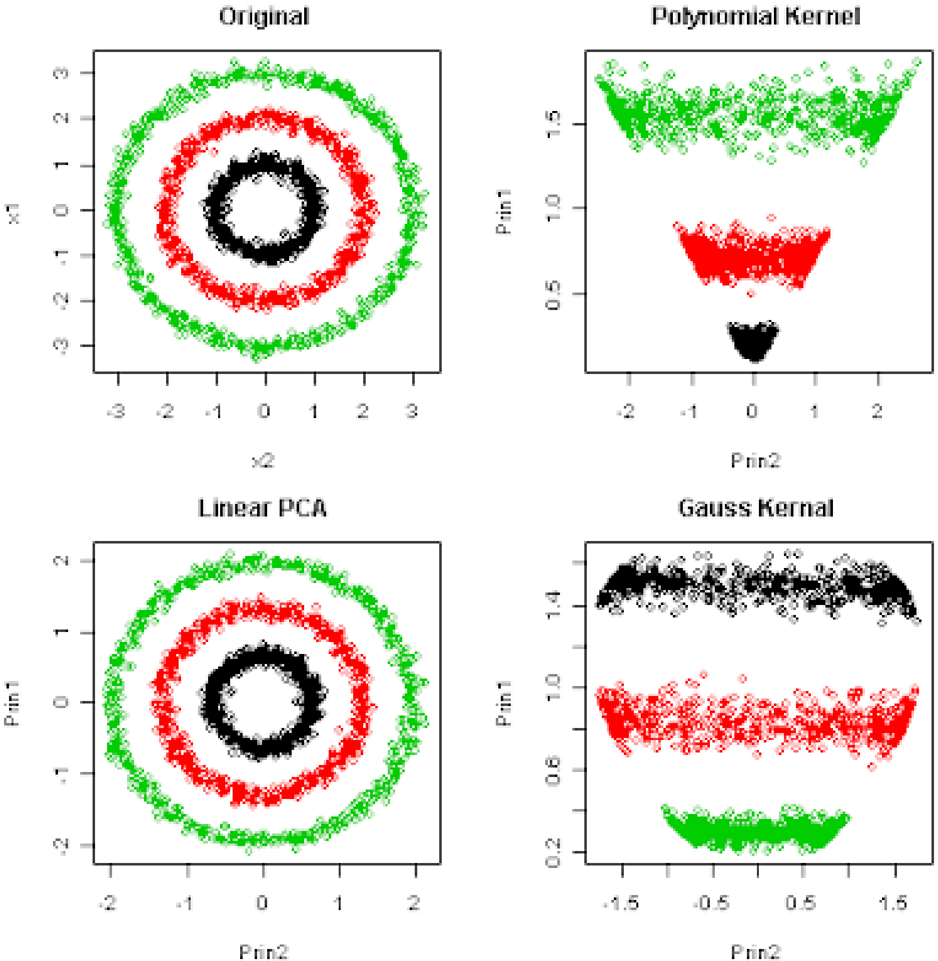
$$Y = A^T X$$

Matrix, linear transformation:
Rotation, translation, reflection, shear

Nonlinear transformation + high dimensional

$$\phi : \mathbb{R}^n \rightarrow \mathbb{R}^d$$
$$\phi : x \rightarrow \phi(x)$$

PCA in the high dimensional space ?
→ No, untractable



# Going in high dimension

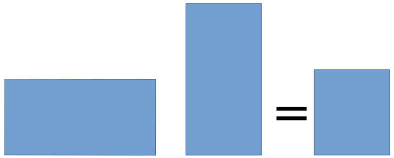


**Singular Value Decomposition** (SVD):
Same non-zero eigenvalues

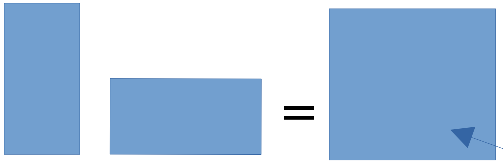
$$X = U \Lambda V^H$$

# Going in high dimension

$X^T X$



$XX^T$



Covariance matrix,
Correlation matrix,
Gram matrix
In high dim space

Eigenvector v

$$X^T X v = \lambda v \qquad XX^T X v = \lambda X v$$

Eigenvector $u = Xv$ of the covariance matrix!

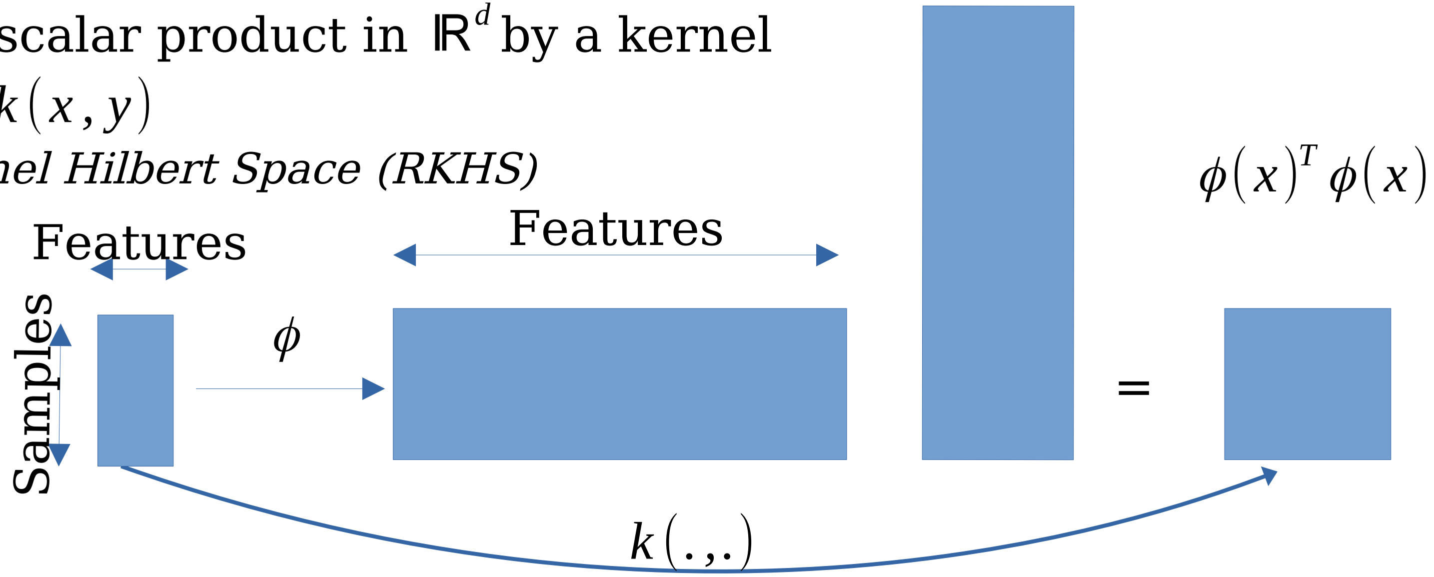
$$XX^T u = \lambda u$$

# The kernel trick!

We replace the scalar product in $\mathbb{R}^d$ by a kernel

$$\langle \phi(x), \phi(y) \rangle = k(x, y)$$

*Reproducible Kernel Hilbert Space (RKHS)*




Examples:

Gaussian kernel (RBF) $K(\mathbf{x}, \mathbf{y}) = \exp\left(-\frac{\|\mathbf{x}-\mathbf{y}\|^2}{2\sigma^2}\right).$

Polynomial kernel $K(x, y) = (x^\top y + c)^d$

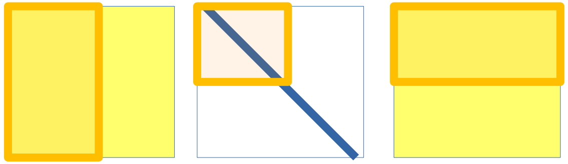
See Moore-Aronszajn theorem for conditions on k

# The kernel trick!

$$\phi(x)^T\phi(x)=\begin{pmatrix} k(x_1,x_1) & k(x_1,x_2) & \dots \\ k(x_2,x_1) & k(x_2,x_2) & \dots \\ \vdots & . & . \\ . & . & k(x_N,x_N) \end{pmatrix}$$

Let us diagonalize this matrix!

Largest eigenvectors



$$\phi(x)^T\phi(x)v_k=\lambda_k v_k$$

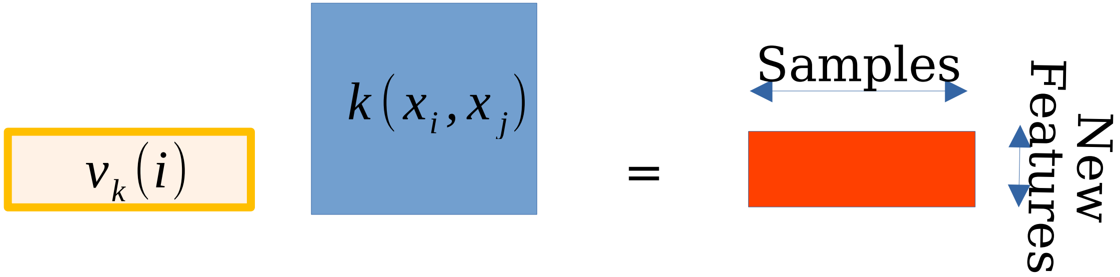
$$u_k=\phi(x)v_k$$

BUT: we cannot directly project, we have to pass through the high dim. space

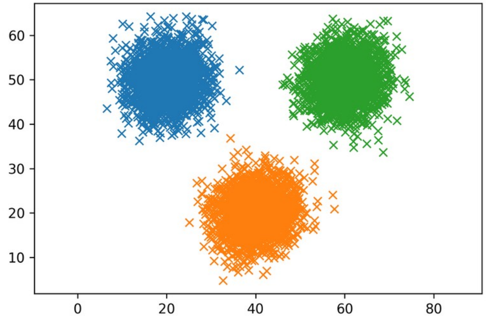
# The kernel trick!

$$\underbrace{(u_k,\phi(x_1))}_{\text{In the high dim space}}=(\phi(x)v_k,\phi(x_1))=v_k^T\underbrace{\phi(x)^T\phi(x_1)}_{\text{Replace by kernel}}=\underbrace{\sum_{i=1}^{N}v_k(i)k(x_i,x_1)}_{\text{In the original space}}$$

Projection



# Kernel PCA




Data not linearly separable:
Send the data in a higher dimensional space, nonlinearly