

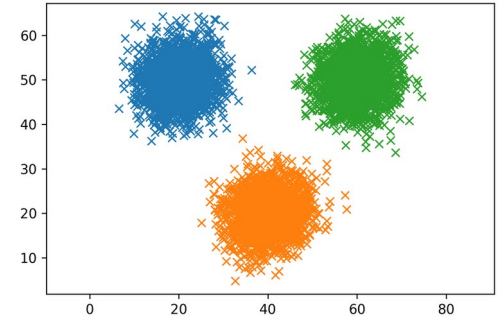
# PCA, kernel PCA, Laplacian eigenmaps

- Feature selection / generation without using labels

# PCA

## (Karhunen-Loève transform)

$$\begin{array}{c}
 \xleftarrow{\text{Samples}} \\
 \mathbf{x} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ \vdots \end{pmatrix} = \begin{pmatrix} x_1(1) & x_1(2) & \dots & x_1(N) \\ x_2(1) & x_2(2) & \dots & \cdot \\ \cdot & \cdot & \dots & \cdot \\ \cdot & \cdot & \dots & x_K(N) \end{pmatrix} \xrightarrow{\text{Features}}
 \end{array}$$



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

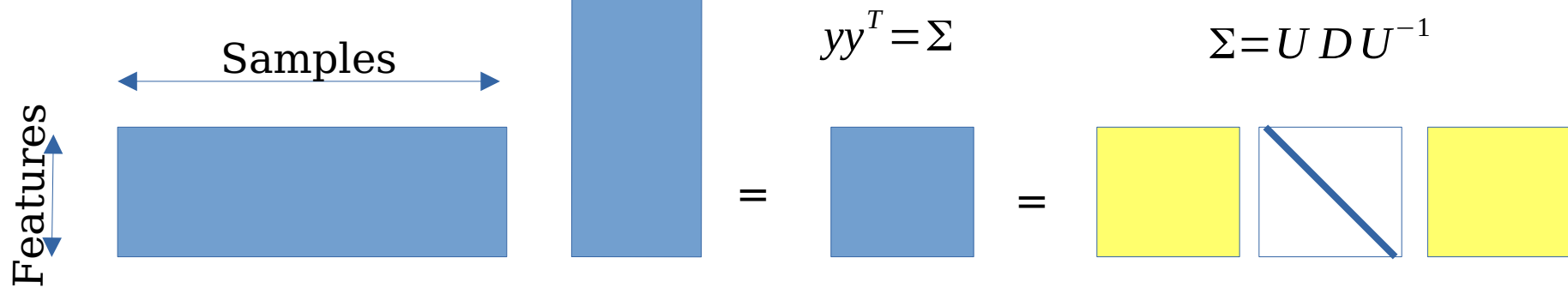
$$yy^T = \Sigma$$

Diagonalize the covariance matrix

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

**Note:** no label information used

# PCA



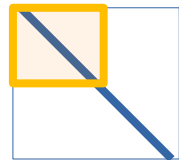
# PCA

$$Y = A^T X$$

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

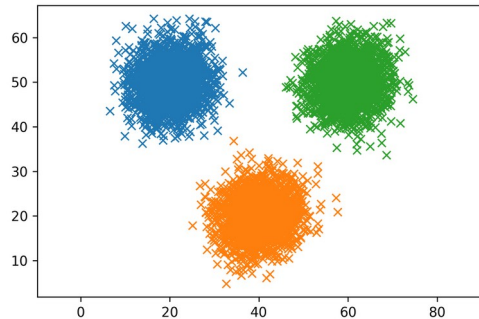
# PCA

- Some visual examples

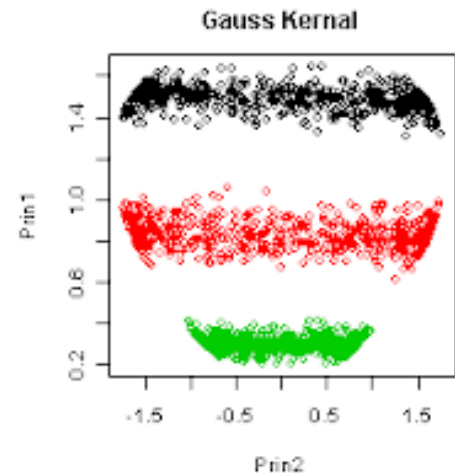
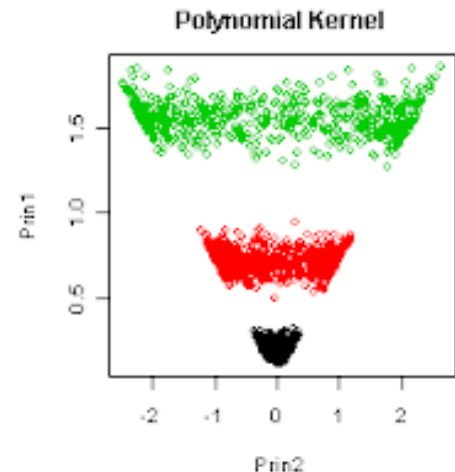
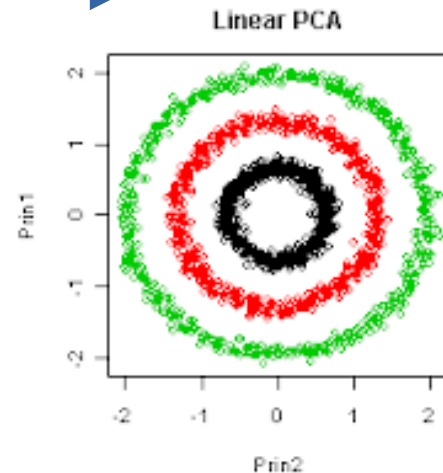
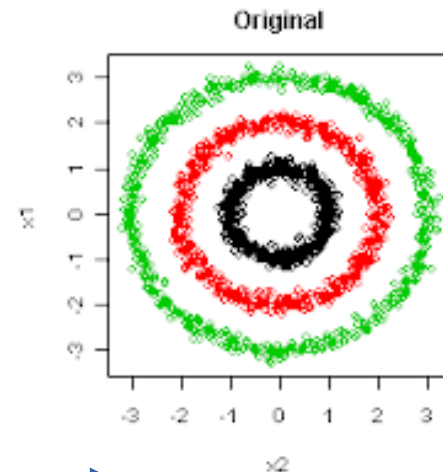
# LDA & PCA

- 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

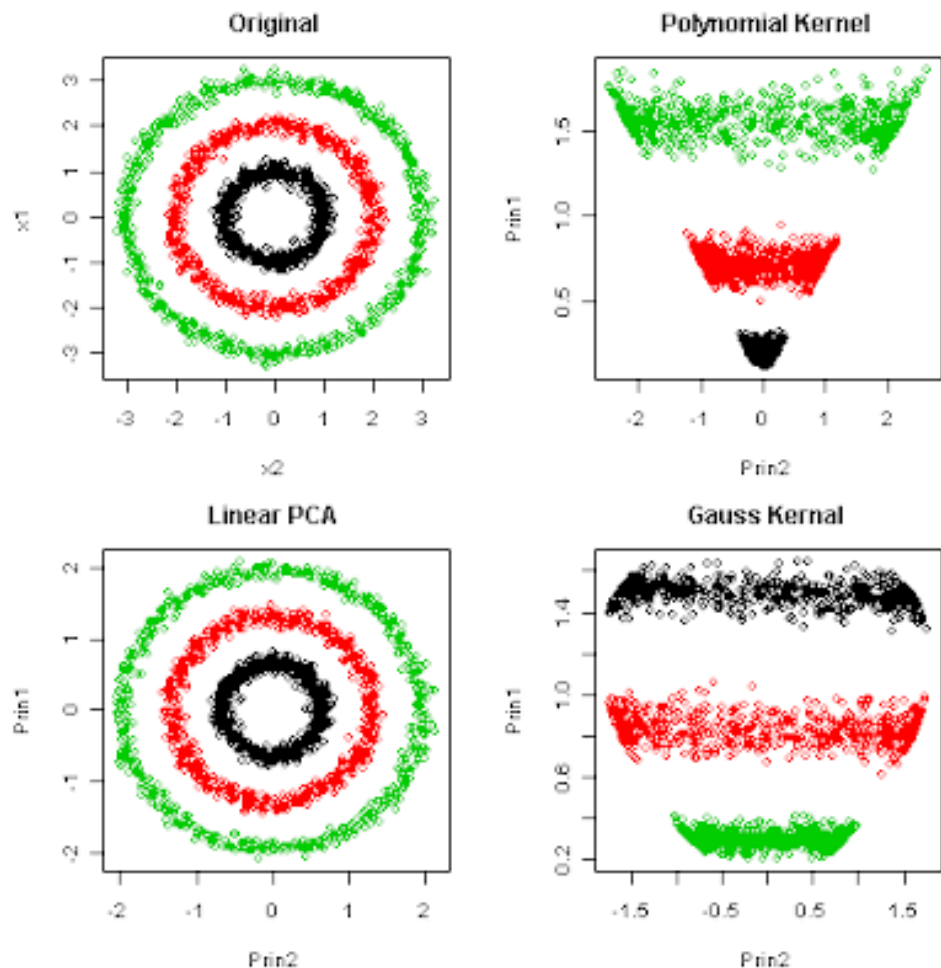
$$Y = A^T X$$

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

Nonlinear transformation + high dimensional

$$\begin{aligned}\phi: \mathbb{R}^n &\rightarrow \mathbb{R}^d \\ \phi: x &\rightarrow \phi(x)\end{aligned}$$

PCA in the high dimensional space ?  
→ No, untractable





# Going in high dimension



$$XX^T$$

A diagram showing the matrix product  $XX^T$ . It consists of a small blue rectangle, followed by a taller blue rectangle, an equals sign, and a small blue square.

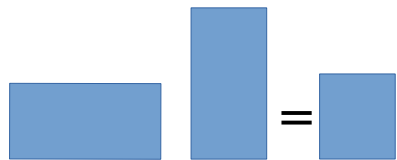
$$X^T X$$

A diagram showing the matrix product  $X^T X$ . It consists of a tall blue rectangle, followed by a small blue rectangle, an equals sign, and a blue square.

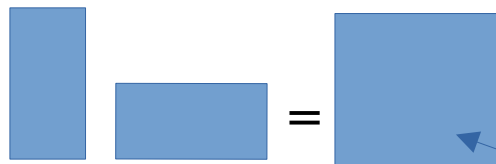
**Singular Value Decomposition (SVD):**  
Same non-zero eigenvalues

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

# Going in high dimension

$$X^T X$$


A diagram illustrating the multiplication of two matrices. On the left, a wide blue rectangle (representing  $X^T$ ) is followed by a tall blue rectangle (representing  $X$ ). An equals sign follows, and then a small blue square (representing the resulting  $X^T X$  matrix).

$$X X^T$$


A diagram illustrating the multiplication of two matrices. On the left, a tall blue rectangle (representing  $X$ ) is followed by a wide blue rectangle (representing  $X^T$ ). An equals sign follows, and then a large blue square (representing the resulting  $X X^T$  matrix).

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

Eigenvector  $v$

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

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

Eigenvector  $u = X v$  of the covariance matrix!

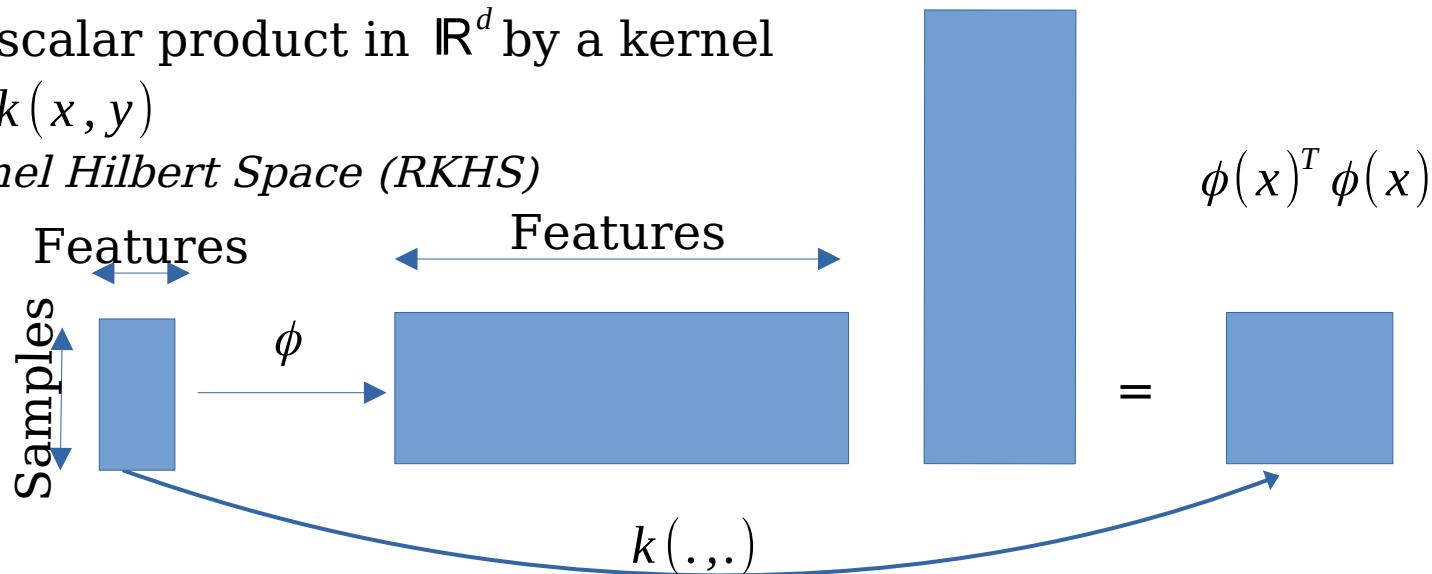
$$X X^T u = \lambda u$$

# The kernel trick!

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

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

*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

$$K(x, y) = (x^T y + c)^d$$

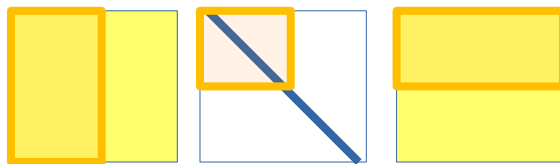
See Moore-Aronszajn theorem for conditions on  $k$

# The kernel trick!

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

Let us diagonalize this matrix!

Largest  
eigenvectors



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

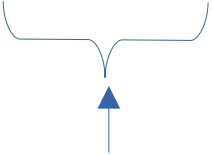
$$u_k = \phi(x) v_k$$

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

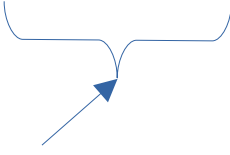
# The kernel trick!

$$(u_k, \phi(x_1)) = (\phi(x) v_k, \phi(x_1)) = v_k^T \phi(x)^T \phi(x_1) = \sum_{i=1}^N v_k(i) k(x_i, x_1)$$

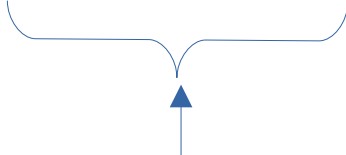
In the high dim  
space



Replace by  
kernel

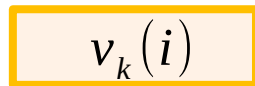


In the original  
space

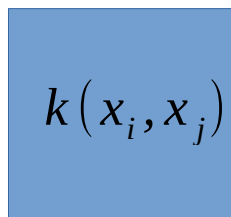


Projection

$v_k(i)$



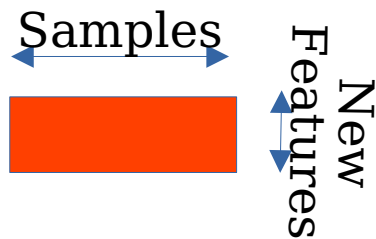
$k(x_i, x_j)$



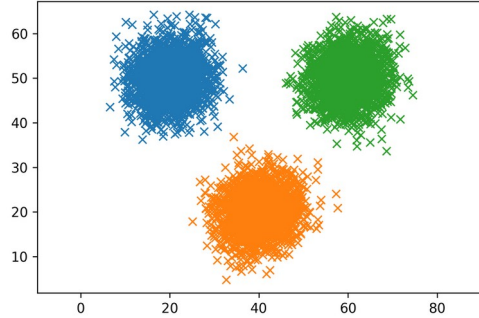
=

Samples

New  
Features



# Kernel PCA



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

