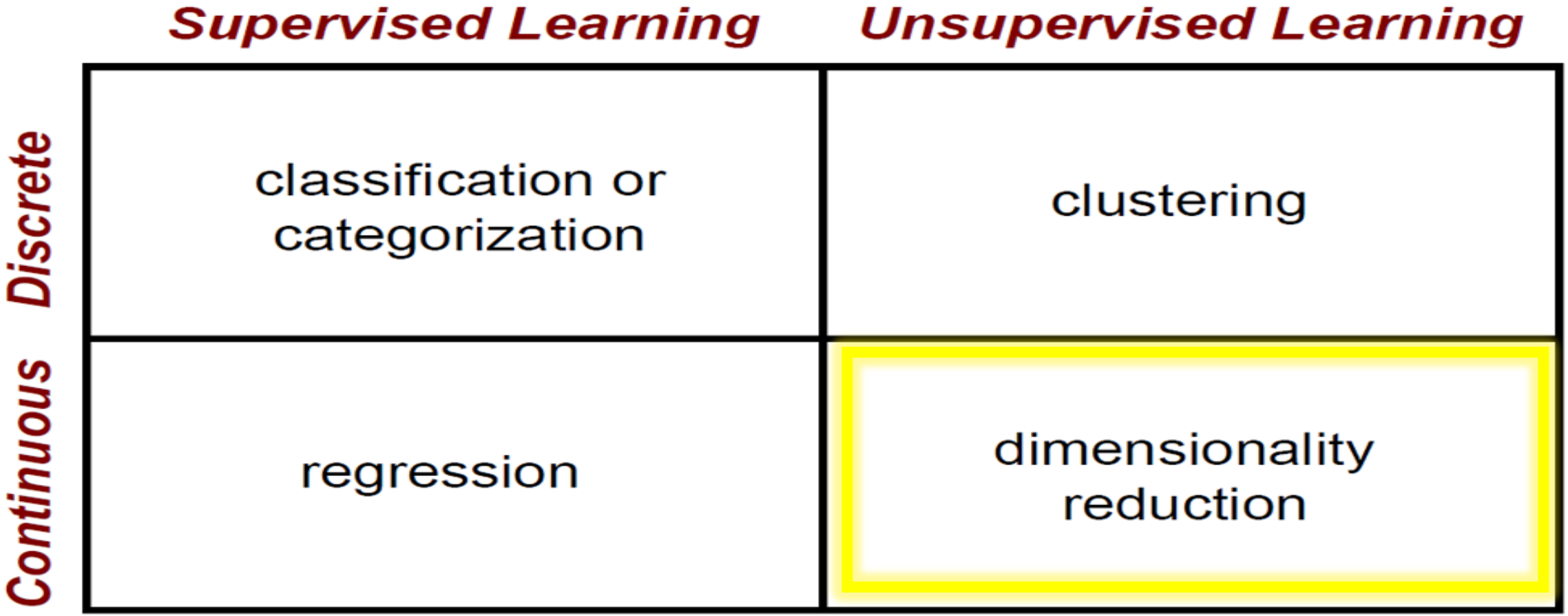
Dimensionality Reduction

Slide credits : Vineeth N Balasubramanian

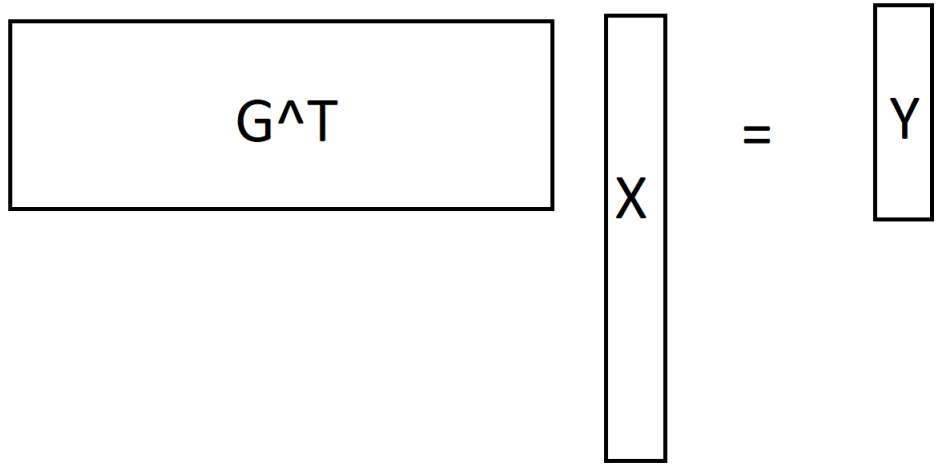
ML Problems


Supervised Learning
Unsupervised Learning
Discrete
Continuous
classification or categorization
clustering
regression
dimensionality reduction

What is Dimensionality Reduction

- Refers to the mapping of the original high-dimensional data onto a lower-dimensional space.
- Criterion for feature reduction can be different for different problems.
  - Unsupervised setting: minimize the information loss
  - Supervised setting: maximize the class discrimination
- Given a set of data points of p variables $\{x_1, x_2, \cdots, x_n\}$: Compute the linear transformation (projection)

$$G \in \Re^{p \times d} : x \in \Re^p \rightarrow y = G^T x \in \Re^d \ (d << p)$$



Dimensionality Reduction vs Feature Selection

- **Feature reduction**
  - All original features are used
  - The transformed features are linear combinations of the original features (in case of linear DR).

- **Feature selection**
  - Only a subset of the original features are used.

Why DR?

- Most machine learning techniques may not be effective for high-dimensional data
  - Curse of Dimensionality
  - Query accuracy and efficiency degrade rapidly as the dimension increases
  - Lower space and time complexity
  - Visualization, Data compression, Noise/irrelevant feature removal
- The intrinsic dimension may be small
  - For example, the number of genes responsible for a certain type of disease may be small

High-dimensional data are strange

- Consider the hypersphere of radius r on a space of dimension d

$$\mathcal{S} = \left\{ \mathbf{x} | \sum_{i=1}^{d} x_i^2 \leq r^2 \right\}$$

- Its volume is

$$V_d(r) = \frac{r^d \pi^{\frac{d}{2}}}{\Gamma\left(\frac{d}{2}+1\right)}$$

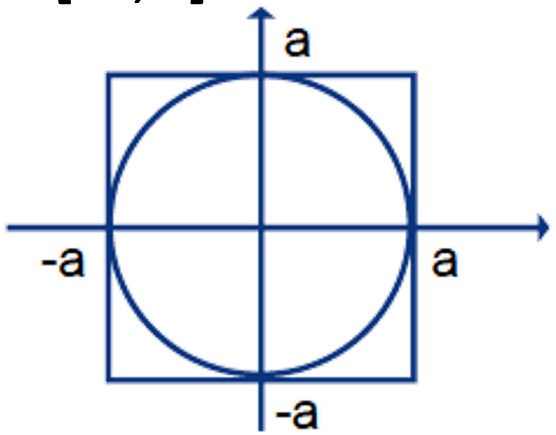
- Where Γ(n) is the Gamma function

$$\Gamma(n) = \int_0^\infty e^{-x} x^{n-1} dx$$

Source: N Vasconcelos

Hypercube vs Hypersphere

- Consider the hyper-cube $[-a,a]^d$ and the inscribed hyper-sphere, i.e.

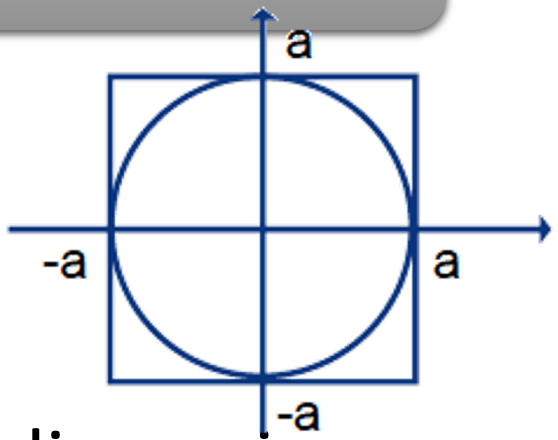


- What does your intuition tell you about the relative sizes of these two objects?
  - Volume of sphere ~= volume of cube
  - Volume of sphere >> volume of cube
  - Volume of sphere << volume of cube

Source: N Vasconcelos

Hypercube vs Hypersphere

- Let's compute the answer



$$f_d = \frac{Vol(sphere)}{Vol(cube)} = \frac{\frac{a^d \pi^{\frac{d}{2}}}{\Gamma\left(\frac{d}{2}+1\right)}}{(2a)^d} = \frac{\pi^{\frac{d}{2}}}{2^d \Gamma\left(\frac{d}{2}+1\right)}$$

- Sequence that does not depend on a, just on the dimension d!

d	1	2	3	4	5	6	7
f_d	1	.785	.524	.308	.164	.08	.037

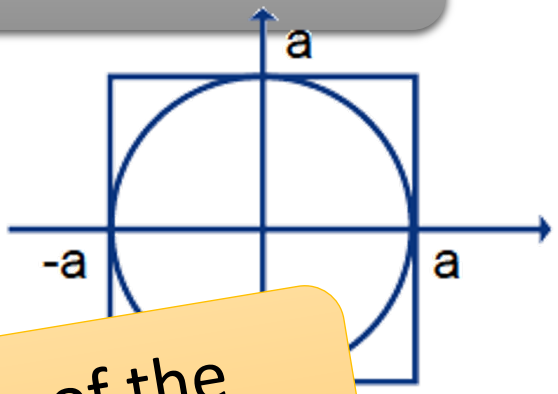
- It goes to zero, and goes to zero fast!

Source: N Vasconcelos

Hypercube vs Hypersphere

- Let's compute the answer

$$f_d = \frac{Vol(sphere)}{Vol(cube)} = \frac{\frac{a^d \pi^{\frac{d}{2}}}{\Gamma\left(\frac{d}{2}+1\right)}}{(2a)^d} = \frac{\pi^{\frac{d}{2}}}{2^d \Gamma\left(\frac{d}{2}+1\right)}$$



- Sequence that does not dep… d!

As the dimension of the space increases, the volume of the sphere is much smaller (infinitesimal) than that of the cube!

			5	6	7
	.785	.524	.308	.164	.08	.037

- It goes to zero, and goes to zero fast!

Source: N Vasconcelos

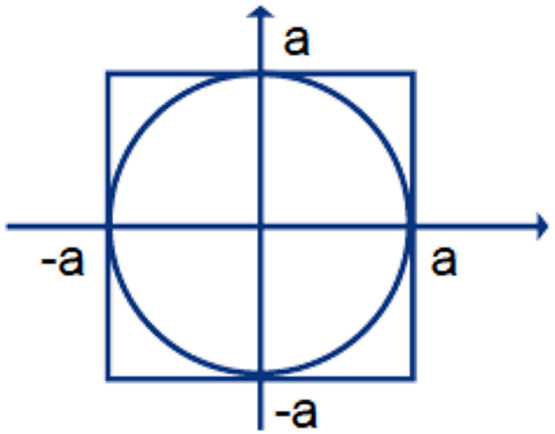
Hypercube vs Hypersphere

Actually not very surprising

1. d = 1

Volume is the same

2. d = 2

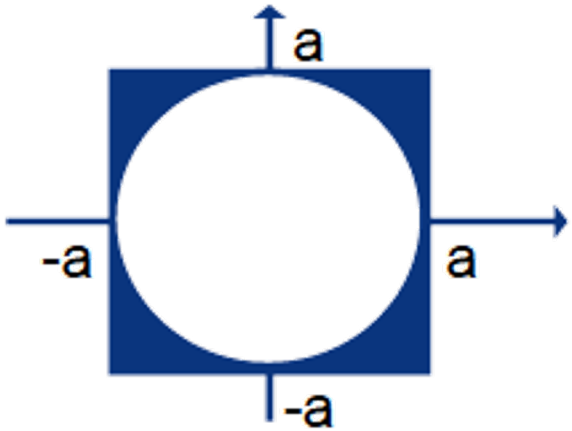


Volume of sphere is already smaller

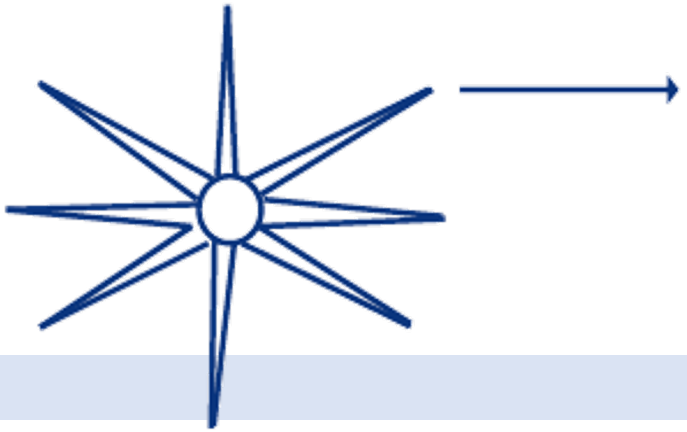
Source: N Vasconcelos

Hypercube vs Hypersphere

- As the dimension increases the volume of the shaded corners becomes larger



- In high dimensions the picture you should have in mind is

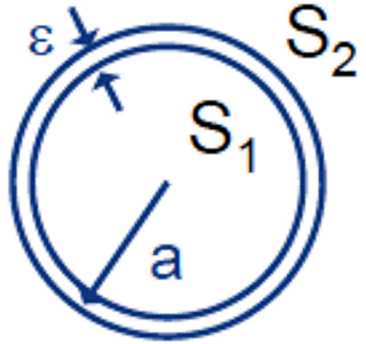


all the volume of the cube is in these spikes!

Source: N Vasconcelos

The Curse of Dimensionality

- Consider the crust of unit hypersphere of thickness ε
- Let's compute the ratio of volumes



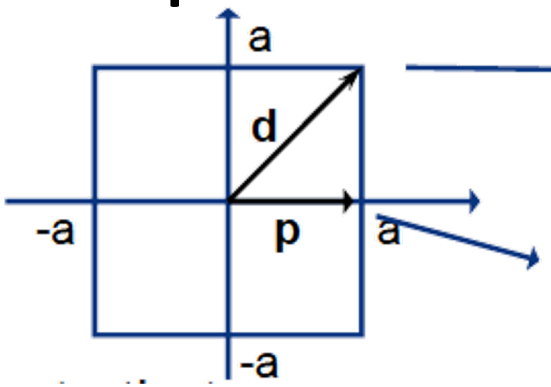
$$\frac{Vol(S_1)}{Vol(S_2)} = \frac{\frac{(a-\epsilon)^d \pi^{\frac{d}{2}}}{\Gamma\left(\frac{d}{2}+1\right)}}{\frac{a^d \pi^{\frac{d}{2}}}{\Gamma\left(\frac{d}{2}+1\right)}} = \frac{a^d \left(1-\frac{\epsilon}{a}\right)^d}{a^d} = \left(1-\frac{\epsilon}{a}\right)^d$$

- No matter how small ε is, ratio goes to zero as

d increases i. e. **"all the volume is in the crust!"**

We can check mathematically

- Consider **d** and **p**



$$\mathbf{d} = (a, a, \ldots, a) \in \mathcal{R}^d$$

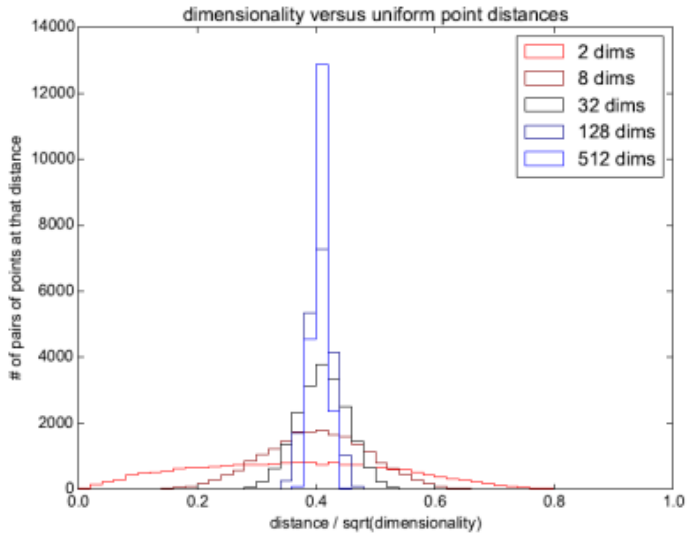
$$\mathbf{p} = (a, 0, \ldots, 0) \in \mathcal{R}^d$$

- Note that

$$\frac{||d||^2}{||p||^2} = \frac{da^2}{a^2} = d \rightarrow \infty$$

$$cos\theta = \frac{\mathbf{d}^T\mathbf{p}}{\sqrt{||d||^2||p||^2}} = \frac{a^2}{\sqrt{da^2a^2}} = \frac{1}{\sqrt{d}} \rightarrow 0$$

- **d** orthogonal to **p** as **d**
- increases and infinitely larger!!!

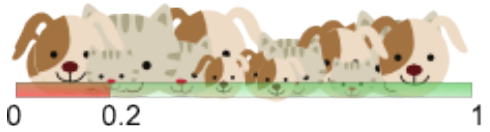


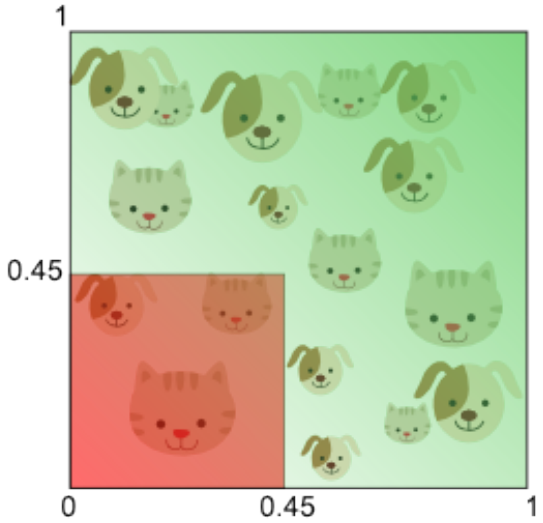
Source: N Vasconcelos

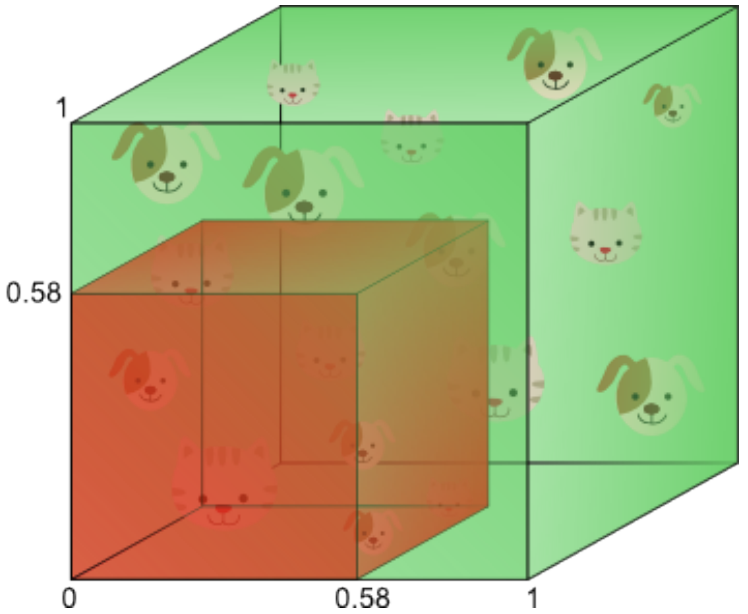
Figure 3.18: histogram of distances in D=2,8,32,128,512

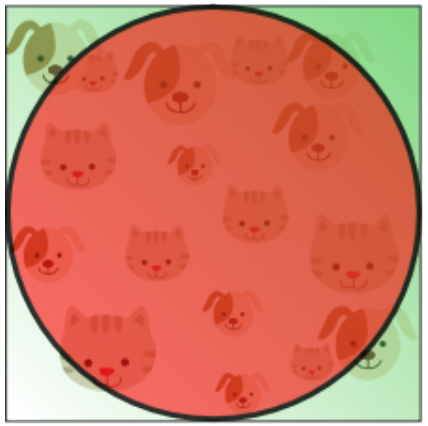
Curse of dimensionality

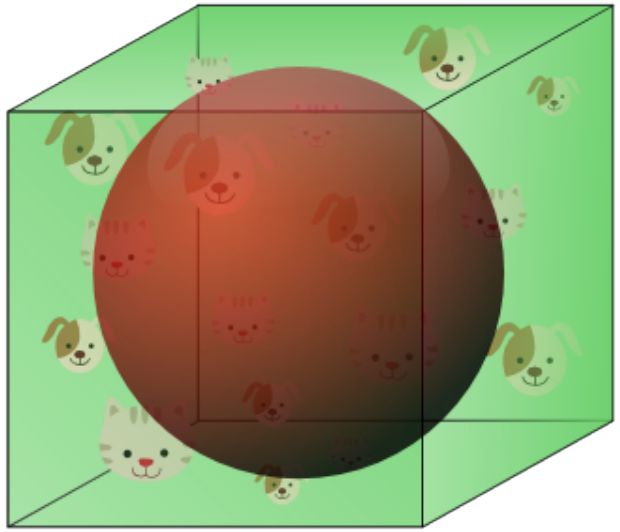
$$\lim_{d \to \infty} \frac{\text{dist}_{\text{max}} - \text{dist}_{\text{min}}}{\text{dist}_{\text{min}}} \to 0$$

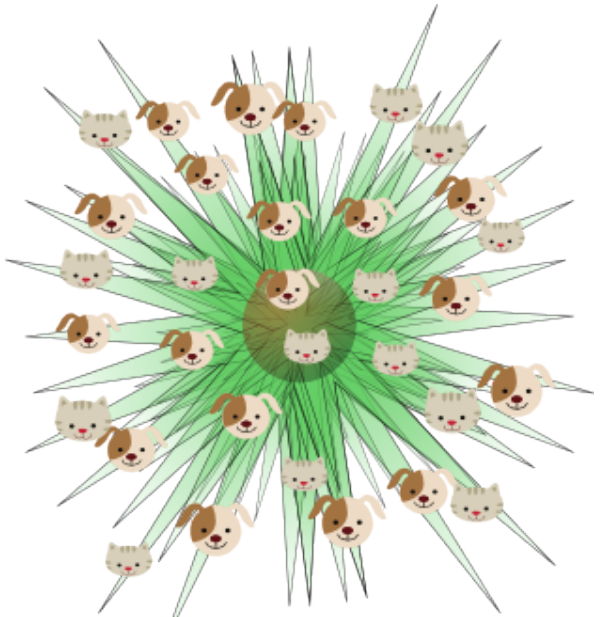










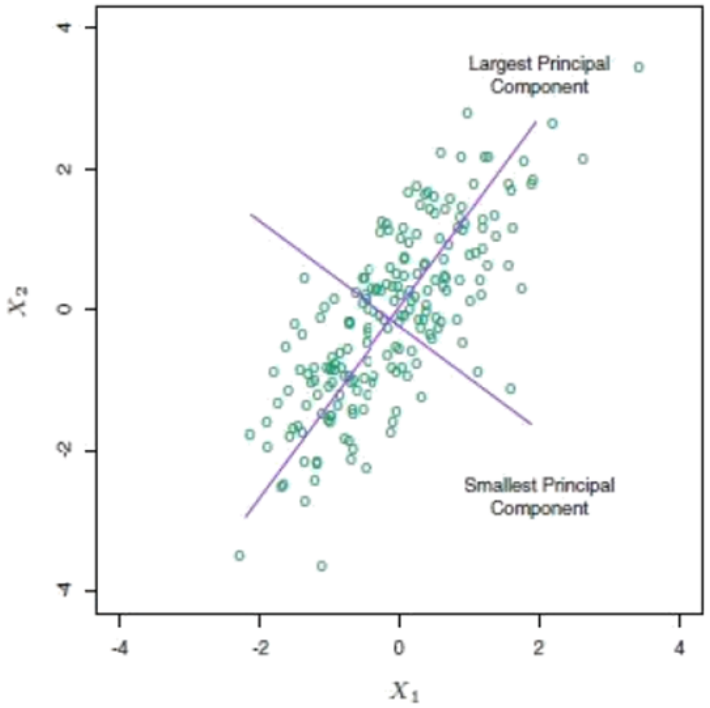


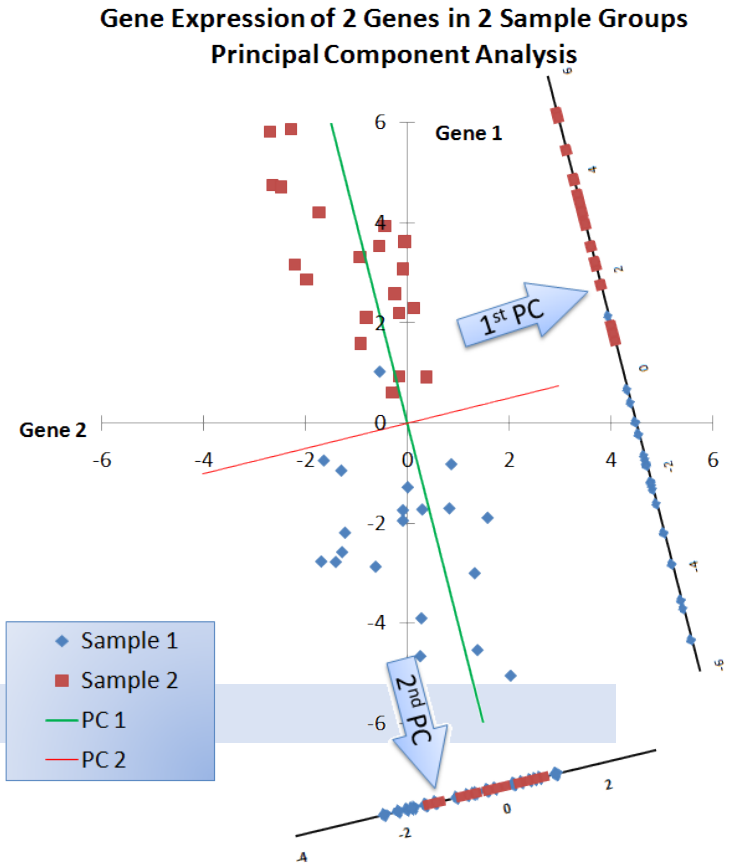
Outline

- Linear Methods
  - Principal Component Analysis (PCA)
  - Fisher (Linear) Discriminant Analysis (LDA)

Principal Component Analysis (PCA)

- Find a low-dimensional space such that when $\boldsymbol{x}$ is projected there, information loss is minimized.
- The projection of $\boldsymbol{x}$ on the direction of $\boldsymbol{w}$ is: $z = \boldsymbol{w}^T\boldsymbol{x}$
- Find $\boldsymbol{w}$ such that Var(z) is maximized,
- Our goal is to project the data onto a space having dimensionality $M < D$ while maximizing the variance





Principal Component Analysis (PCA)

- Find a low-dimensional space such that when $\boldsymbol{x}$ is projected there, information loss is minimized.
- The projection of $\boldsymbol{x}$ on the direction of $\boldsymbol{w}$ is: $z = \boldsymbol{w}^T\boldsymbol{x}$
- Find $\boldsymbol{w}$ such that Var(z) is maximized

$$\begin{aligned}\text{Var}(z) &= \text{Var}(\boldsymbol{w}^T\boldsymbol{x}) = E[(\boldsymbol{w}^T\boldsymbol{x} - \boldsymbol{w}^T\boldsymbol{\mu})^2] \\ &= E[(\boldsymbol{w}^T\boldsymbol{x} - \boldsymbol{w}^T\boldsymbol{\mu})(\boldsymbol{w}^T\boldsymbol{x} - \boldsymbol{w}^T\boldsymbol{\mu})] \\ &= E[\boldsymbol{w}^T(\boldsymbol{x} - \boldsymbol{\mu})(\boldsymbol{x} - \boldsymbol{\mu})^T\boldsymbol{w}] \\ &= \boldsymbol{w}^T E[(\boldsymbol{x} - \boldsymbol{\mu})(\boldsymbol{x} - \boldsymbol{\mu})^T]\boldsymbol{w} = \boldsymbol{w}^T \sum \boldsymbol{w}\end{aligned}$$

where $\text{Var}(\boldsymbol{x}) = E[(\boldsymbol{x} - \boldsymbol{\mu})(\boldsymbol{x} - \boldsymbol{\mu})^T] = \sum$

PCA

- Maximize Var(z) subject to $||\boldsymbol{w}||=1$

$$\max_{\mathbf{w}_1} \mathbf{w}_1^T \Sigma \mathbf{w}_1 - \alpha\left(\mathbf{w}_1^T \mathbf{w}_1 - 1\right)$$

Using Lagrange multipliers, a la SVM

$\sum \boldsymbol{w}_1 = \alpha \boldsymbol{w}_1$ that is, $\boldsymbol{w}_1$ is an eigenvector of $\sum$

Choose the one with the largest eigenvalue for Var(z) to be max

- Second principal component: Max Var(z_2), s.t., $||\boldsymbol{w}_2||=1$ and orthogonal to $\boldsymbol{w}_1$

$$\max_{\mathbf{w}_2} \mathbf{w}_2^T \Sigma \mathbf{w}_2 - \alpha\left(\mathbf{w}_2^T \mathbf{w}_2 - 1\right) - \beta\left(\mathbf{w}_2^T \mathbf{w}_1 - 0\right)$$

How?

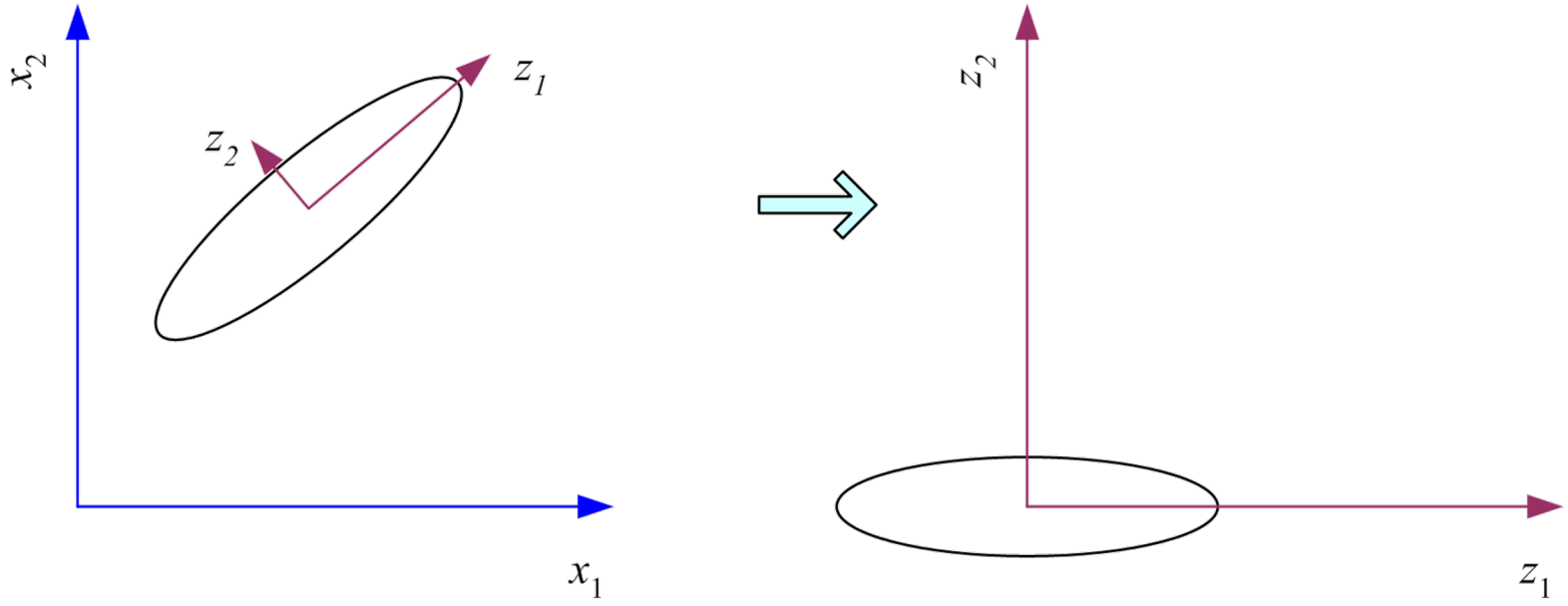
$\sum \boldsymbol{w}_2 = \alpha \boldsymbol{w}_2$ that is, $\boldsymbol{w}_2$ is another eigenvector of $\sum$

and so on.

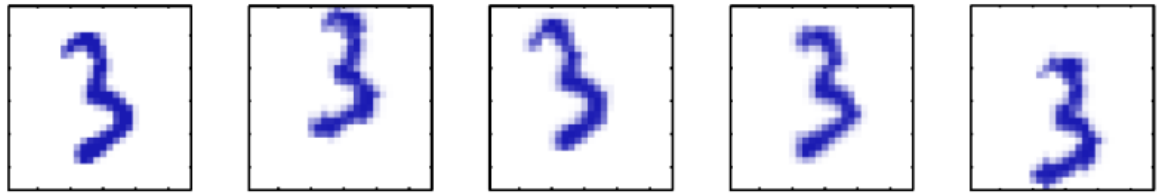
PCA

$$\mathbf{z} = \mathbf{W}^T(\boldsymbol{x} - \boldsymbol{\mu})$$

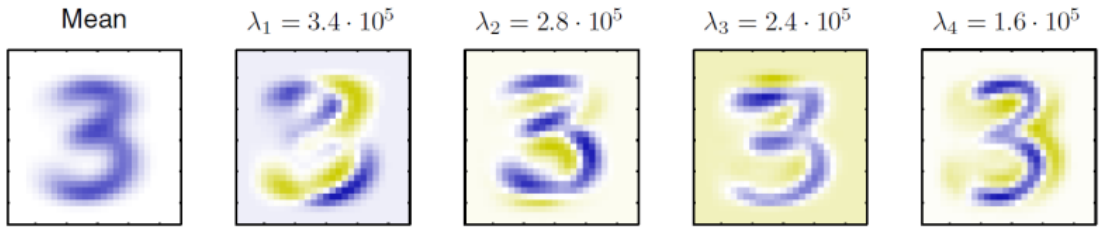
where the columns of $\mathbf{W}$ are the eigenvectors of $\sum$, and $\boldsymbol{\mu}$ is sample mean; Centers the data at the origin and rotates the axes



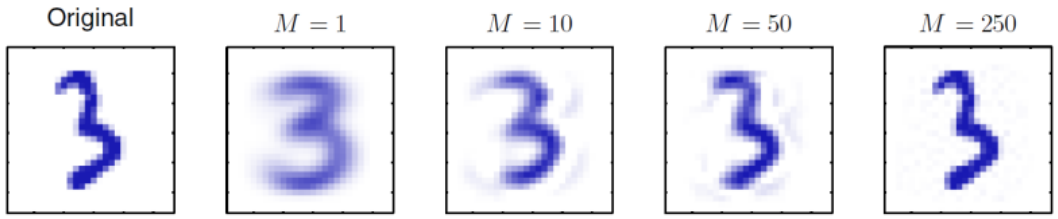
PCA



A synthetic data set obtained by taking one of the off-line digit images and creating multiple copies in each of which the digit has undergone a random displacement and rotation within some larger image field. The resulting images each have $100 \times 100 = 10,000$



The mean vector $\overline{\mathbf{x}}$ along with the first four PCA eigenvectors $\mathbf{u}_1, \ldots, \mathbf{u}_4$ for the off-line digits data set, together with the corresponding eigenvalues.



An original example from the off-line digits data set together with its PCA reconstructions obtained by retaining M principal components for various values of M. As M increases the reconstruction becomes more accurate and would become perfect when $M = D = 28 \times 28 = 784$.

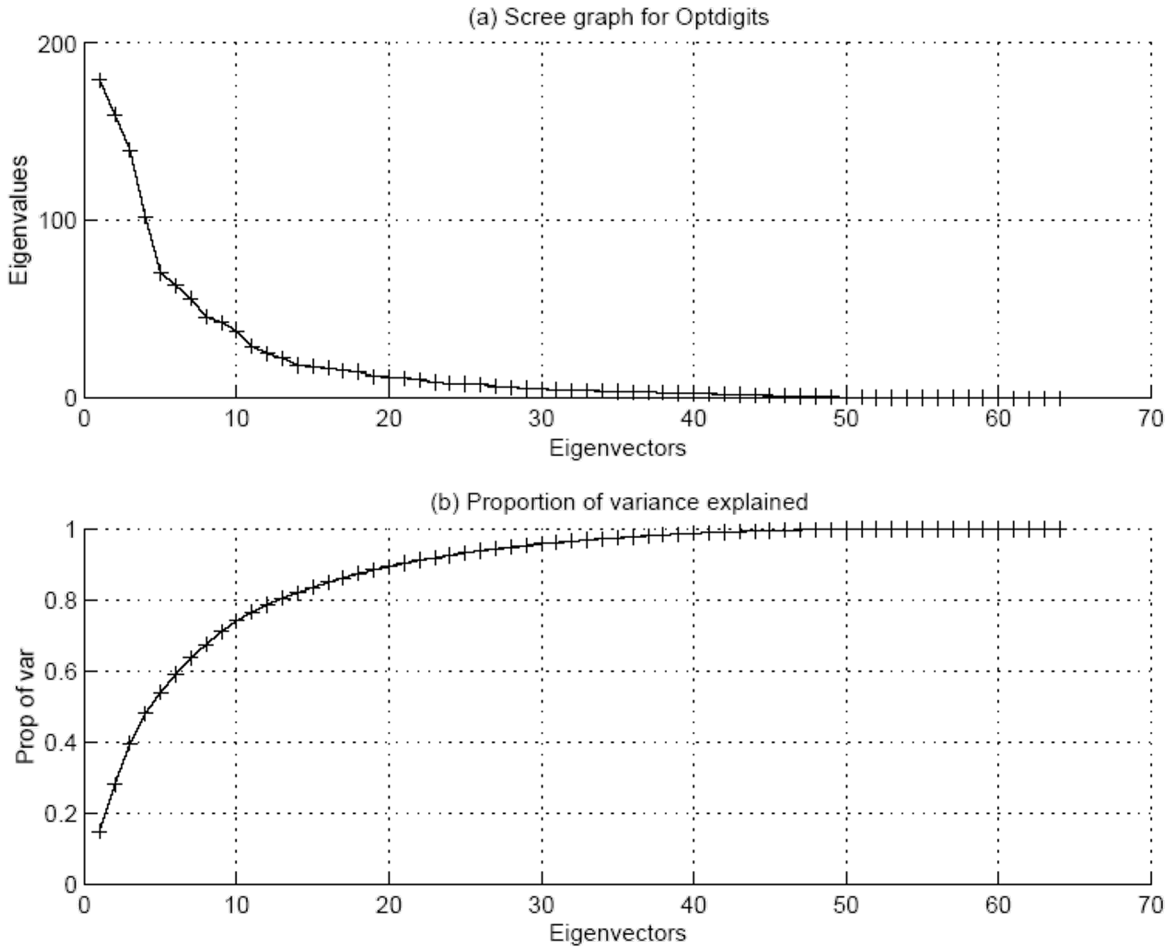
How to choose k

- Proportion of Variance (PoV) explained

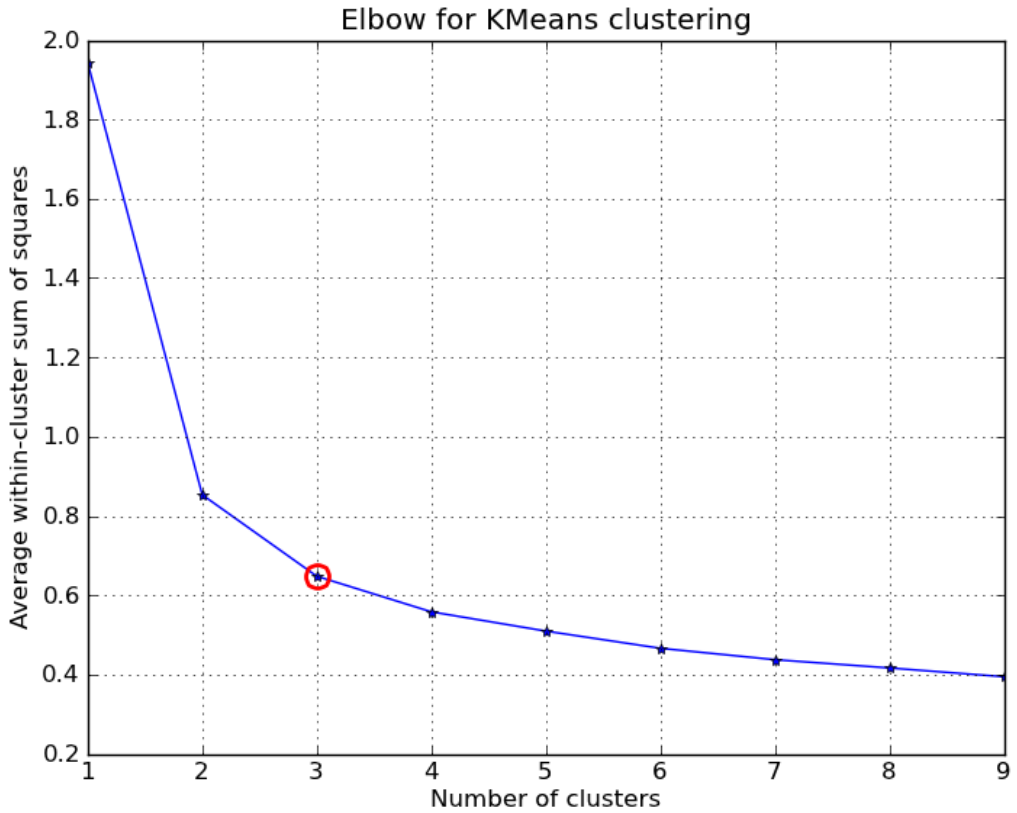
$$\frac{\lambda_1 + \lambda_2 + \cdots + \lambda_k}{\lambda_1 + \lambda_2 + \cdots + \lambda_k + \cdots + \lambda_d}$$

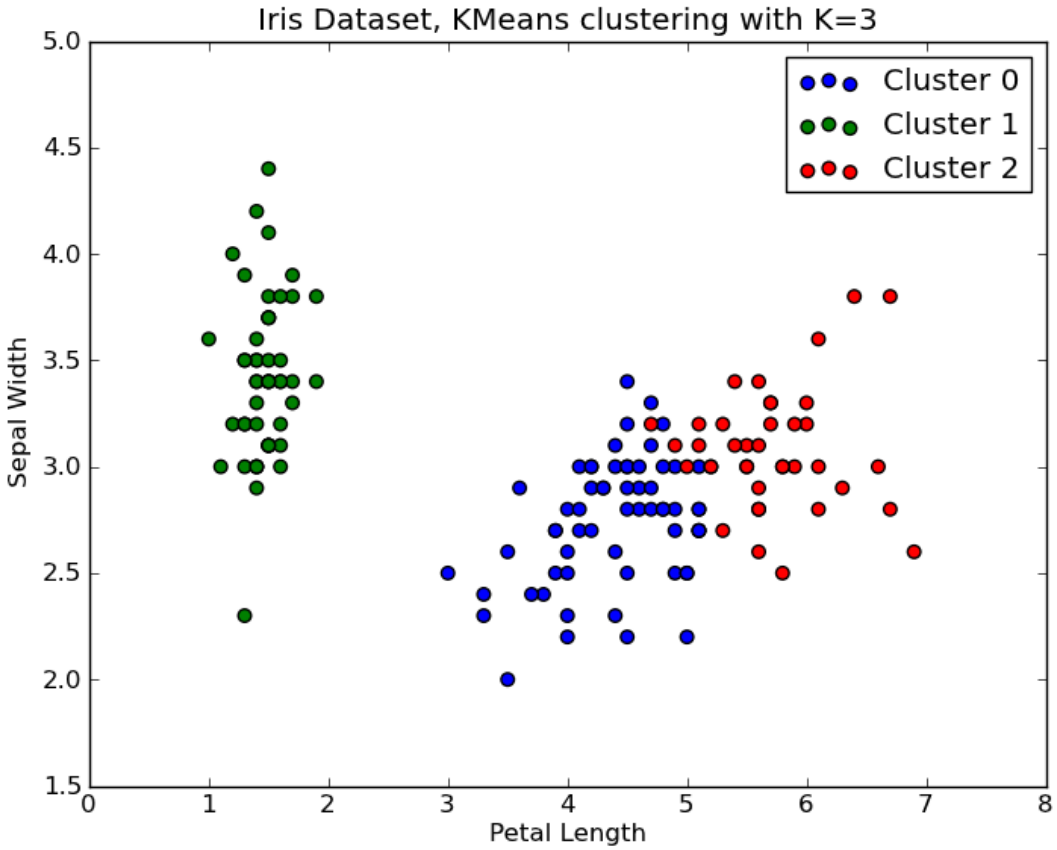
  when λ_i are sorted in descending order
- Typically, stop at PoV>0.9
- See graph plots of PoV vs *k*, stop at "elbow"

Illustration

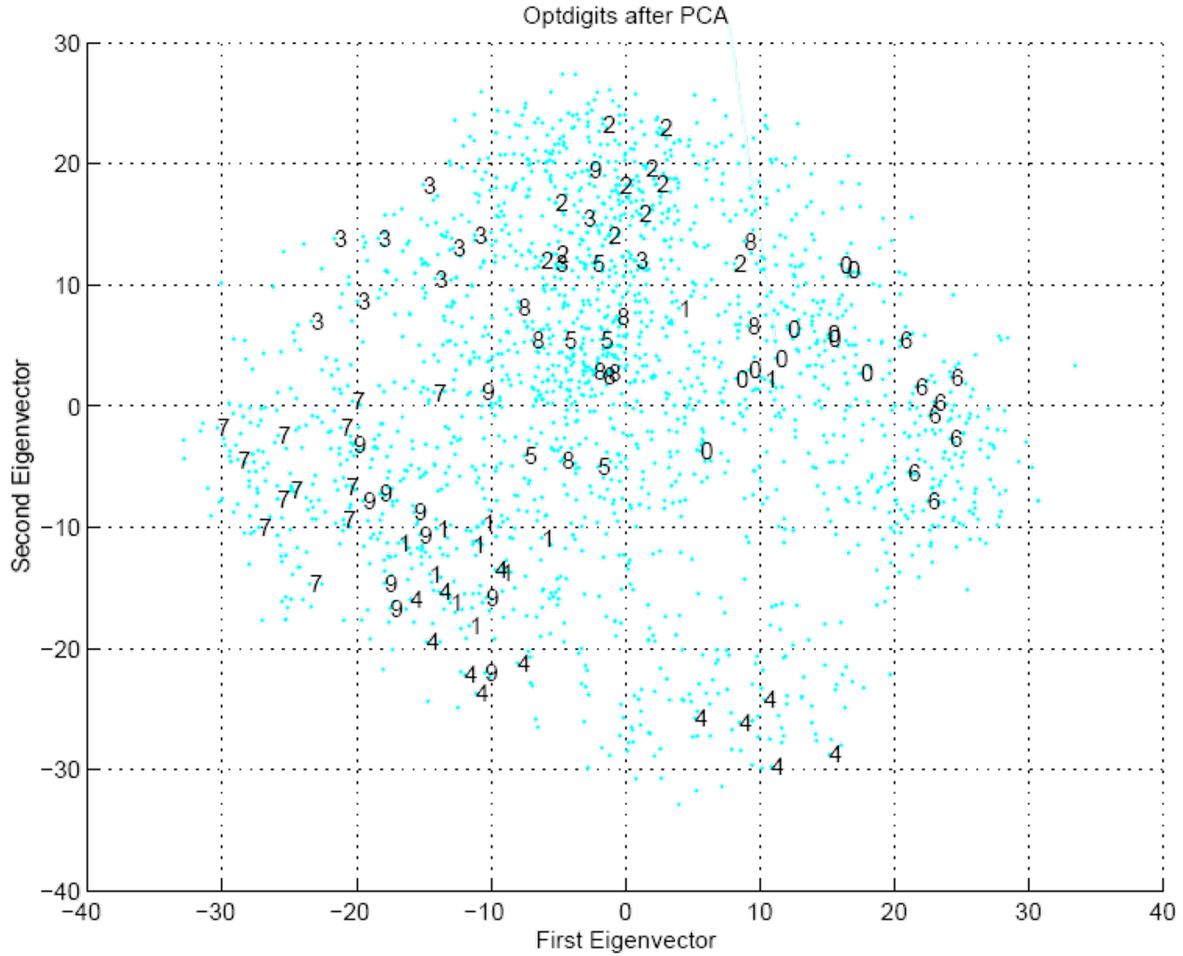


Elbow plot in Clustering





Example


Optdigits after PCA
Second Eigenvector
First Eigenvector
30
20
10
0
−10
−20
−30
−40
−40
−30
−20
−10
0
10
20
30
40

PCA for high dimensional data

N << D

$$\frac{1}{N}\mathbf{X}^{\mathrm{T}}\mathbf{X}\mathbf{u}_i = \lambda_i \mathbf{u}_i.$$

$$\frac{1}{N}\mathbf{X}\mathbf{X}^{\mathrm{T}}(\mathbf{X}\mathbf{u}_i) = \lambda_i(\mathbf{X}\mathbf{u}_i).$$

$$\frac{1}{N}\mathbf{X}\mathbf{X}^{\mathrm{T}}\mathbf{v}_i = \lambda_i \mathbf{v}_i$$

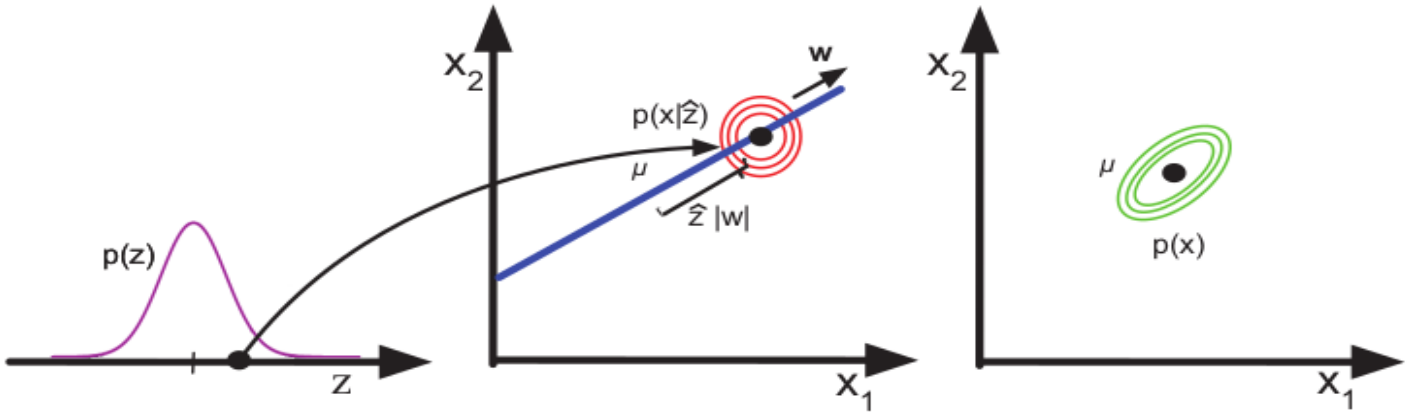
$$\left(\frac{1}{N}\mathbf{X}^{\mathrm{T}}\mathbf{X}\right)(\mathbf{X}^{\mathrm{T}}\mathbf{v}_i) = \lambda_i(\mathbf{X}^{\mathrm{T}}\mathbf{v}_i)$$

O(N^3) instead of O(D^3)

Probabilistic PCA

- Maximum likelihood solution of a probabilistic latent variable model.
- latent variable **z** corresponding to the principal-component subspace.
- Gaussian prior distribution $p(\mathbf{z})$ over the latent variable, together with a Gaussian conditional distribution $p(\mathbf{x}/\mathbf{z})$ for the observed variable **x** conditioned on the value of the latent variable.

$$p(\mathbf{z}) = \mathcal{N}(\mathbf{z}|\mathbf{0}, \mathbf{I}). \qquad p(\mathbf{x}|\mathbf{z}) = \mathcal{N}(\mathbf{x}|\mathbf{W}\mathbf{z} + \boldsymbol{\mu}, \sigma^2\mathbf{I}) \qquad \mathbf{x} = \mathbf{W}\mathbf{z} + \boldsymbol{\mu} + \boldsymbol{\epsilon}$$



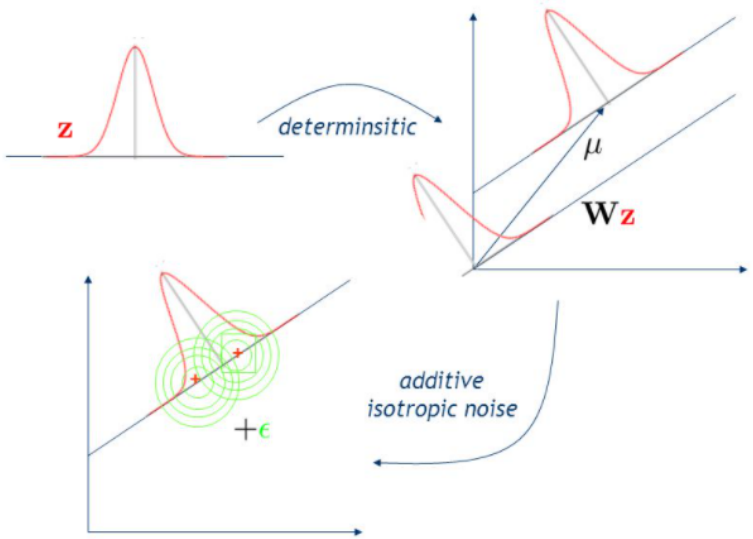


Figure 12.1 Illustration of the PPCA generative process, where we have $L = 1$ latent dimension generating $D = 2$ observed dimensions. Based on Figure 12.9 of (Bishop 2006b).

Probabilistic PCA

- Maximum likelihood solution of a probabilistic latent variable model.
- latent variable **z** corresponding to the principal-component subspace.
- Gaussian prior distribution $p(\mathbf{z})$ over the latent variable, together with a Gaussian conditional distribution $p(\mathbf{x}/\mathbf{z})$ for the observed variable **x** conditioned on the value of the latent variable.

$$p(\mathbf{z}) = \mathcal{N}(\mathbf{z}|\mathbf{0}, \mathbf{I}). \qquad p(\mathbf{x}|\mathbf{z}) = \mathcal{N}(\mathbf{x}|\mathbf{W}\mathbf{z} + \boldsymbol{\mu}, \sigma^2\mathbf{I}) \qquad \mathbf{x} = \mathbf{W}\mathbf{z} + \boldsymbol{\mu} + \boldsymbol{\epsilon}$$

- Parameter Estimation : marginal likelihood $p(\mathbf{x}) = \int p(\mathbf{x}|\mathbf{z})p(\mathbf{z})\,\mathrm{d}\mathbf{z}.$ $\boxed{p(\mathbf{x}) = \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}, \mathbf{C})}$
-

$$\begin{aligned}
\mathbb{E}[\mathbf{x}] &= \mathbb{E}[\mathbf{W}\mathbf{z} + \boldsymbol{\mu} + \boldsymbol{\epsilon}] = \boldsymbol{\mu} \\
\mathrm{cov}[\mathbf{x}] &= \mathbb{E}\left[(\mathbf{W}\mathbf{z} + \boldsymbol{\epsilon})(\mathbf{W}\mathbf{z} + \boldsymbol{\epsilon})^{\mathrm{T}}\right] \\
\mathbf{C} &= \mathbb{E}\left[\mathbf{W}\mathbf{z}\mathbf{z}^{\mathrm{T}}\mathbf{W}^{\mathrm{T}}\right] + \mathbb{E}[\boldsymbol{\epsilon}\boldsymbol{\epsilon}^{\mathrm{T}}] = \mathbf{W}\mathbf{W}^{\mathrm{T}} + \sigma^2\mathbf{I}
\end{aligned}$$

आई आई टी हैदराबाद
IIT Hyderabad

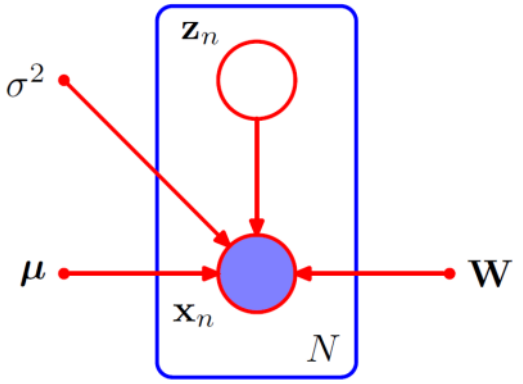
Probabilistic PCA

- Maximum likelihood solution of a probabilistic latent variable model.
- latent variable **z** corresponding to the principal-component subspace.
- Gaussian prior distribution $p(\mathbf{z})$ over the latent variable, together with a Gaussian conditional distribution $p(\mathbf{x}/\mathbf{z})$ for the observed variable **x** conditioned on the value of the latent variable.

$$p(\mathbf{z}) = \mathcal{N}(\mathbf{z}|\mathbf{0}, \mathbf{I}). \qquad p(\mathbf{x}|\mathbf{z}) = \mathcal{N}(\mathbf{x}|\mathbf{W}\mathbf{z} + \boldsymbol{\mu}, \sigma^2\mathbf{I}) \qquad \mathbf{x} = \mathbf{W}\mathbf{z} + \boldsymbol{\mu} + \boldsymbol{\epsilon}$$

- Parameter Estimation : marginal likelihood $p(\mathbf{x}) = \int p(\mathbf{x}|\mathbf{z})p(\mathbf{z})\,\mathrm{d}\mathbf{z}.$ $\boxed{p(\mathbf{x}) = \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}, \mathbf{C})}$
- Predictive distribution : finding z for an x

$$p(\mathbf{z}|\mathbf{x}) = \mathcal{N}\left(\mathbf{z}|\mathbf{M}^{-1}\mathbf{W}^{\mathrm{T}}(\mathbf{x} - \boldsymbol{\mu}), \sigma^{-2}\mathbf{M}\right).$$



Probabilistic PCA

- Gaussian prior distribution $p(\mathbf{z})$ over the latent variable, together with a Gaussian conditional distribution $p(\mathbf{x}/\mathbf{z})$ for the observed variable $\mathbf{x}$ conditioned on the value of the latent variable.

$$p(\mathbf{z}) = \mathcal{N}(\mathbf{z}|\mathbf{0}, \mathbf{I}). \qquad p(\mathbf{x}|\mathbf{z}) = \mathcal{N}(\mathbf{x}|\mathbf{W}\mathbf{z} + \boldsymbol{\mu}, \sigma^2\mathbf{I}) \qquad \mathbf{x} = \mathbf{W}\mathbf{z} + \boldsymbol{\mu} + \boldsymbol{\epsilon}$$

- Parameter Estimation : marginal likelihood $p(\mathbf{x}) = \int p(\mathbf{x}|\mathbf{z})p(\mathbf{z})\,\mathrm{d}\mathbf{z}.$ $\boxed{p(\mathbf{x}) = \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}, \mathbf{C})}$

$$\ln p(\mathbf{X}|\boldsymbol{\mu}, \mathbf{W}, \sigma^2) = \sum_{n=1}^{N} \ln p(\mathbf{x}_n|\mathbf{W}, \boldsymbol{\mu}, \sigma^2)$$

$$= -\frac{ND}{2}\ln(2\pi) - \frac{N}{2}\ln|\mathbf{C}| - \frac{1}{2}\sum_{n=1}^{N}(\mathbf{x}_n - \boldsymbol{\mu})^{\mathrm{T}}\mathbf{C}^{-1}(\mathbf{x}_n - \boldsymbol{\mu}).$$

$$\ln p(\mathbf{X}|\mathbf{W}, \boldsymbol{\mu}, \sigma^2) = -\frac{N}{2}\left\{D\ln(2\pi) + \ln|\mathbf{C}| + \mathrm{Tr}\left(\mathbf{C}^{-1}\mathbf{S}\right)\right\}$$

$$\mathbf{S} = \frac{1}{N}\sum_{n=1}^{N}(\mathbf{x}_n - \overline{\mathbf{x}})(\mathbf{x}_n - \overline{\mathbf{x}})^{\mathrm{T}}.$$

$$\mathbf{W}_{\mathrm{ML}} = \mathbf{U}_M(\mathbf{L}_M - \sigma^2\mathbf{I})^{1/2}\mathbf{R}$$

where $\mathbf{U}_M$ is a $D \times M$ matrix whose columns are given by any subset (of size M) of the eigenvectors of the data covariance matrix $\mathbf{S}$, the $M \times M$ diagonal matrix $\mathbf{L}_M$ has elements given by the corresponding eigenvalues λ_i, and $\mathbf{R}$ is an arbitrary $M \times M$ orthogonal matrix.

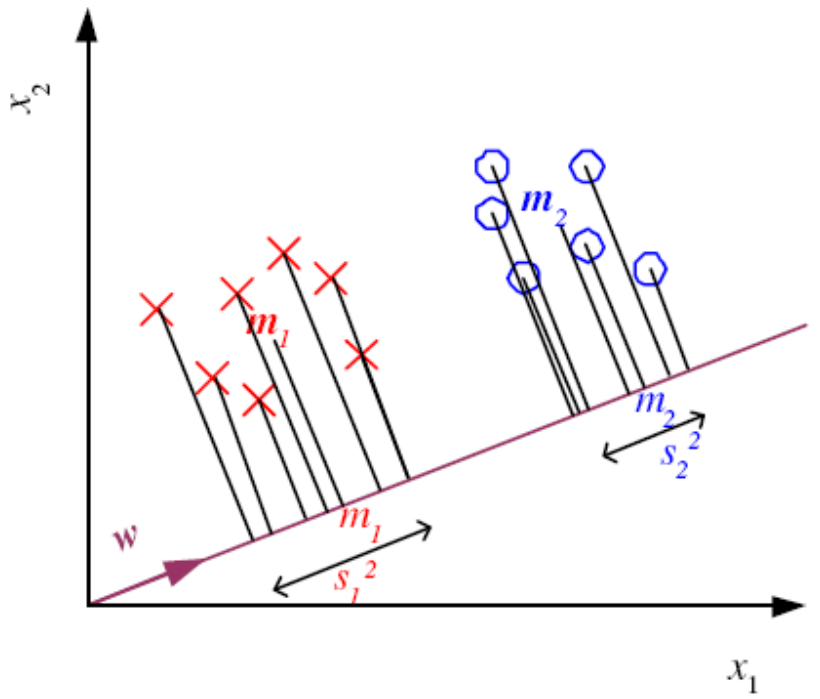
आई आई टी हैदराबाद
IIT Hyderabad

Outline

- Linear Methods
  - Principal Component Analysis (PCA)
  - Fisher (Linear) Discriminant Analysis (LDA)

Fishers Linear Discriminant

- Supervised linear DR method
- Find a low-dimensional space such that when ***x*** is projected, classes are well-separated.
- Simplest measure of the separation of the classes, when projected onto w, is the separation of the projected class means

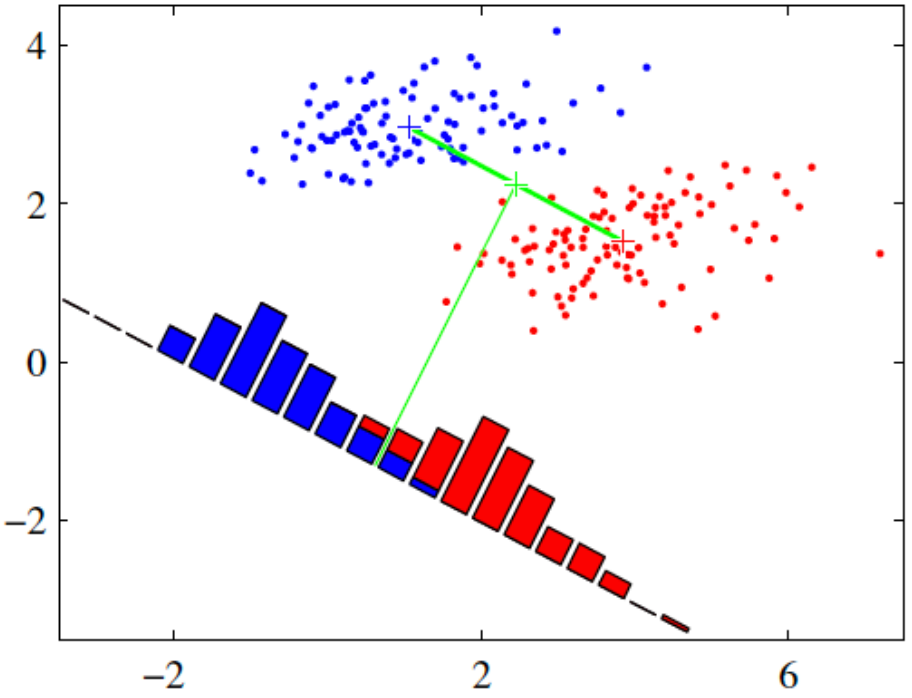


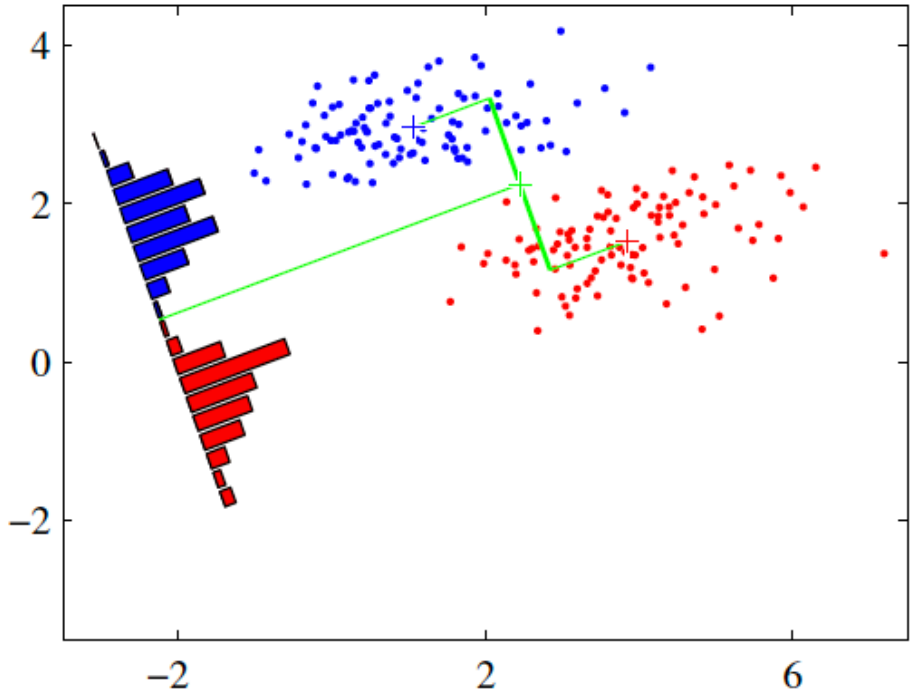
$$\mathbf{m}_1 = \frac{1}{N_1} \sum_{n \in \mathcal{C}_1} \mathbf{x}_n, \qquad \mathbf{m}_2 = \frac{1}{N_2} \sum_{n \in \mathcal{C}_2} \mathbf{x}_n.$$

Maximize $m_2 - m_1 = \mathbf{w}^{\mathrm{T}}(\mathbf{m}_2 - \mathbf{m}_1)$

Subj. to $\sum_i w_i^2 = 1.$

Fishers Linear Discriminant





$$\mathbf{m}_1 = \frac{1}{N_1} \sum_{n \in \mathcal{C}_1} \mathbf{x}_n, \qquad \mathbf{m}_2 = \frac{1}{N_2} \sum_{n \in \mathcal{C}_2} \mathbf{x}_n.$$

Maximize $m_2 - m_1 = \mathbf{w}^{\mathrm{T}}(\mathbf{m}_2 - \mathbf{m}_1)$

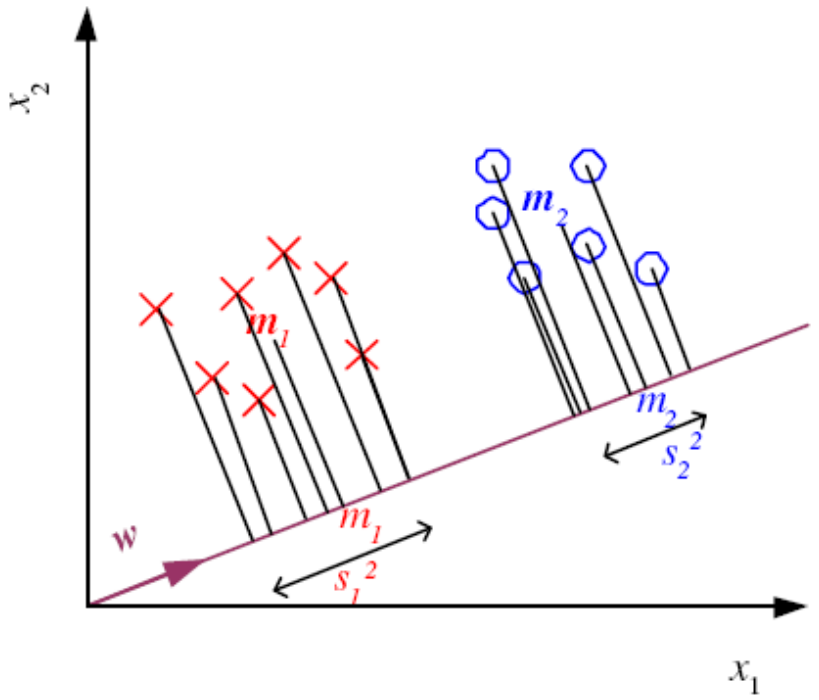
Subj. to $\sum_i w_i^2 = 1.$

Fishers Linear Discriminant

- Idea proposed by Fisher is to maximizea function that will give a large separation between the projected class means while also giving a small variance within each class, thereby minimizing the class overlap.
- Find ***w*** that maximizes

$$J(\mathbf{w}) = \frac{(m_1 - m_2)^2}{s_1^2 + s_2^2}$$

$$m_1 = \frac{\sum_t \mathbf{w}^T \mathbf{x}^t r^t}{\sum_t r^t} \qquad s_1^2 = \sum_t \left(\mathbf{w}^T \mathbf{x}^t - m_1\right)^2 r^t$$



Fishers Linear Discriminant

- Between-class scatter:

$$
\begin{aligned}
(m_1 - m_2)^2 &= \left(\mathbf{w}^T\mathbf{m}_1 - \mathbf{w}^T\mathbf{m}_2\right)^2 \\
&= \mathbf{w}^T(\mathbf{m}_1 - \mathbf{m}_2)(\mathbf{m}_1 - \mathbf{m}_2)^T\mathbf{w} \\
&= \mathbf{w}^T\mathbf{S}_B\mathbf{w} \quad \text{where} \quad \mathbf{S}_B = (\mathbf{m}_1 - \mathbf{m}_2)(\mathbf{m}_1 - \mathbf{m}_2)^T
\end{aligned}
$$

- Within-class scatter:

$$
\begin{aligned}
s_1^2 &= \sum_t \left(\mathbf{w}^T\mathbf{x}^t - m_1\right)^2 r^t \\
&= \sum_t \mathbf{w}^T(\mathbf{x}^t - \mathbf{m}_1)(\mathbf{x}^t - \mathbf{m}_1)^T\mathbf{w}r^t = \mathbf{w}^T\mathbf{S}_1\mathbf{w} \\
\text{where } \mathbf{S}_1 &= \sum_t (\mathbf{x}^t - \mathbf{m}_1)(\mathbf{x}^t - \mathbf{m}_1)^T r^t \\
s_1^2 + s_2^2 &= \mathbf{w}^T\mathbf{S}_W\mathbf{w} \quad \text{where} \quad \mathbf{S}_W = \mathbf{S}_1 + \mathbf{S}_2
\end{aligned}
$$

Fisher's Linear Discriminant

- Find ***w*** that max

$$J(\mathbf{w}) = \frac{\mathbf{w}^T \mathbf{S}_B \mathbf{w}}{\mathbf{w}^T \mathbf{S}_W \mathbf{w}} = \frac{\left|\mathbf{w}^T (\mathbf{m}_1 - \mathbf{m}_2)(\mathbf{m}_1 - \mathbf{m}_2)^T \mathbf{w}\right|}{\mathbf{w}^T \mathbf{S}_W \mathbf{w}}$$

- FLD soln:

$$\mathbf{w} = c \cdot \mathbf{S}_W^{-1}(\mathbf{m}_1 - \mathbf{m}_2)$$

- Alternative parametric soln:

$$\mathbf{w} = \Sigma^{-1}(\mu_1 - \mu_2)$$

when $p(\mathbf{x}|C_i) \sim \mathcal{N}(\mu_i, \Sigma)$

Fisher's Linear Discriminant

- Within-class scatter:

$$\mathbf{S}_W = \sum_{i=1}^{K} \mathbf{S}_i \qquad \mathbf{S}_i = \sum_t r_i^t \left(\mathbf{x}^t - \mathbf{m}_i\right)\left(\mathbf{x}^t - \mathbf{m}_i\right)^T$$

- Between-class scatter:

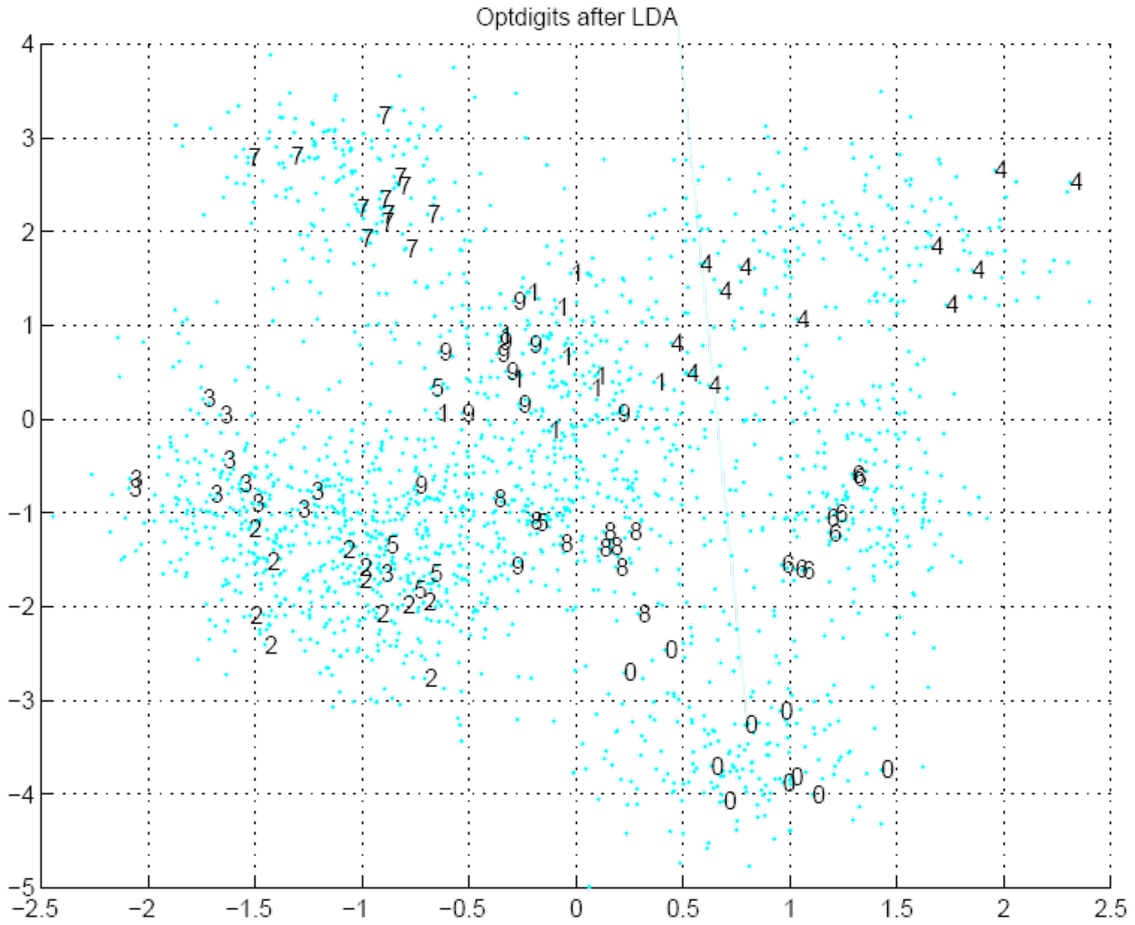
$$\mathbf{S}_B = \sum_{i=1}^{K} N_i \left(\mathbf{m}_i - \mathbf{m}\right)\left(\mathbf{m}_i - \mathbf{m}\right)^T \qquad \mathbf{m} = \frac{1}{K}\sum_{i=1}^{K} \mathbf{m}_i$$

- Find $\mathbf{W}$ that max

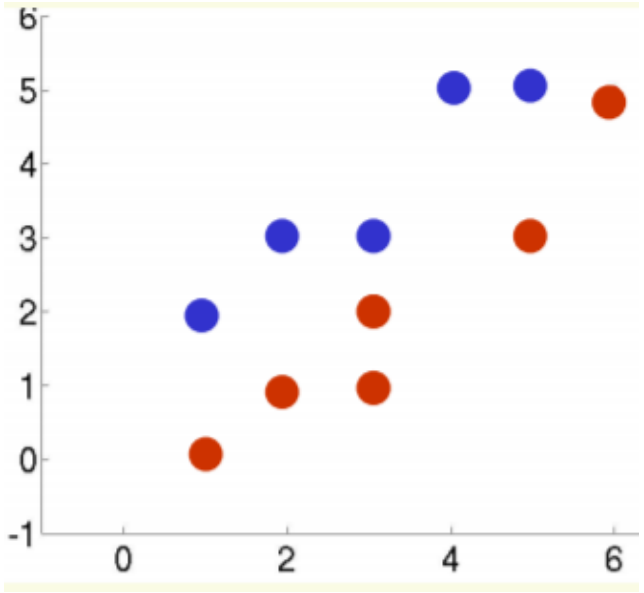
$$J(\mathbf{W}) = \frac{\left|\mathbf{W}^T \mathbf{S}_B \mathbf{W}\right|}{\left|\mathbf{W}^T \mathbf{S}_W \mathbf{W}\right|}$$

The largest eigenvectors of $\mathbf{S}_W^{-1}\mathbf{S}_B$

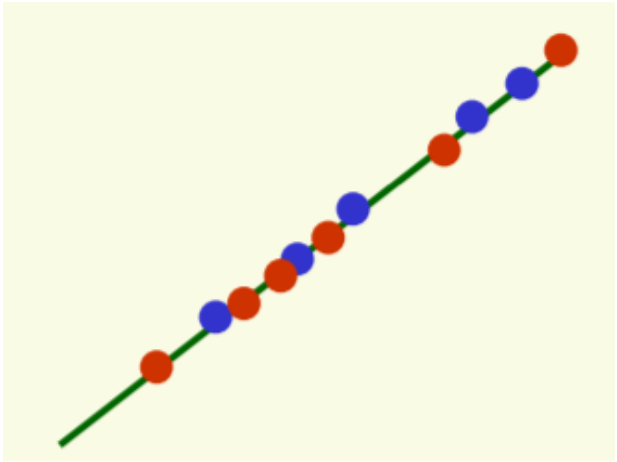
FLD: Illustration



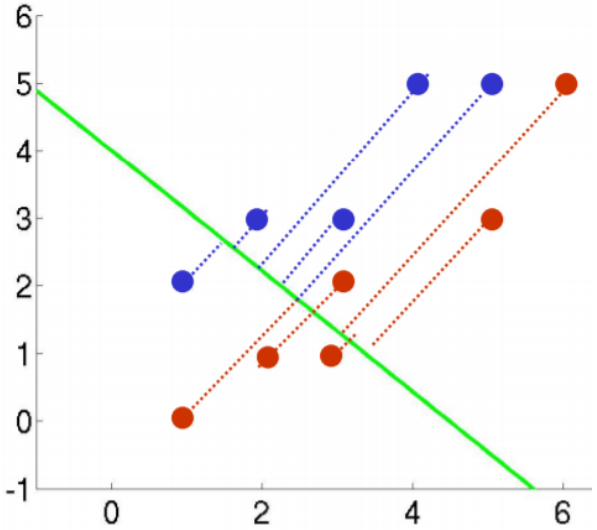
LDA: Illustration



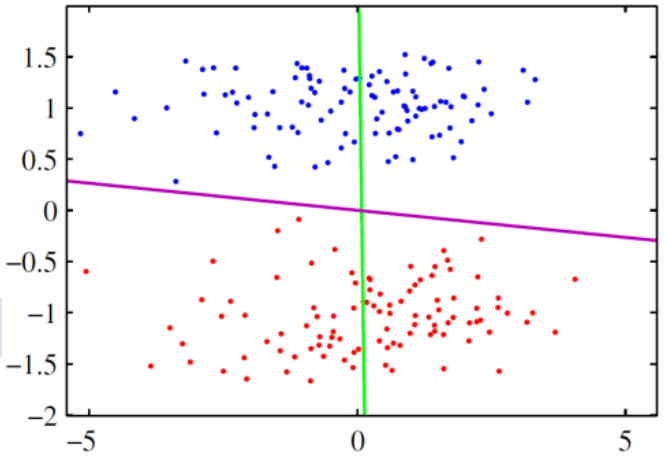
Data



PCA projection



LDA projection



Kernel PCA

- PCA

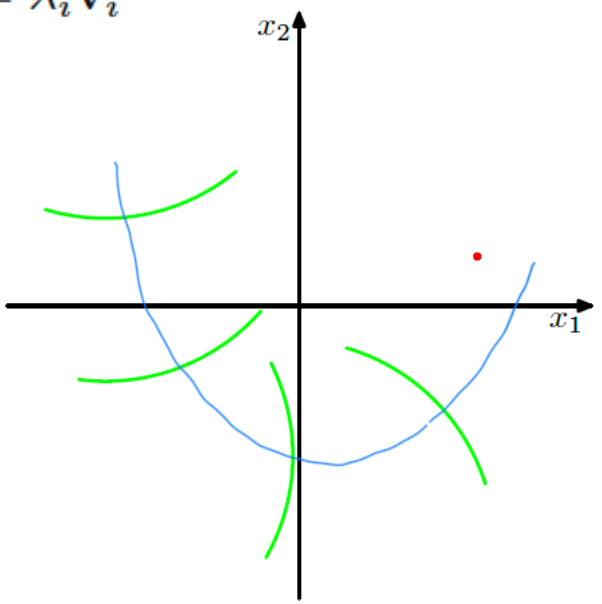
$$\mathbf{S} = \frac{1}{N}\sum_{n=1}^{N}\mathbf{x}_n\mathbf{x}_n^{\mathrm{T}}, \qquad \mathbf{S}\mathbf{u}_i = \lambda_i\mathbf{u}_i$$

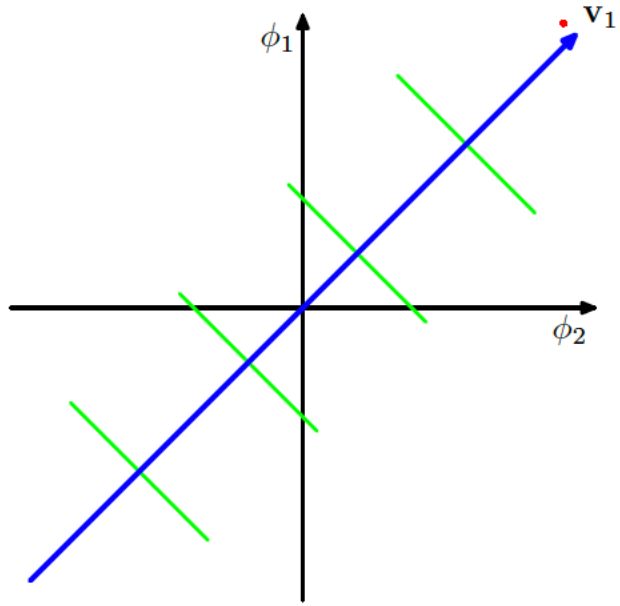
- nonlinear transformation ϕ(x) into an M-dimensional feature space, so that each data point xn is thereby projected onto a point ϕ(xn).

$$\mathbf{C} = \frac{1}{N}\sum_{n=1}^{N}\boldsymbol{\phi}(\mathbf{x}_n)\boldsymbol{\phi}(\mathbf{x}_n)^{\mathrm{T}} \qquad \mathbf{C}\mathbf{v}_i = \lambda_i\mathbf{v}_i$$

$$\frac{1}{N}\sum_{n=1}^{N}\boldsymbol{\phi}(\mathbf{x}_n)\left\{\boldsymbol{\phi}(\mathbf{x}_n)^{\mathrm{T}}\mathbf{v}_i\right\} = \lambda_i\mathbf{v}_i$$

$$\mathbf{v}_i = \sum_{n=1}^{N}a_{in}\boldsymbol{\phi}(\mathbf{x}_n).$$





Kernel PCA

- Standard PCA

$$\mathbf{S} = \frac{1}{N}\sum_{n=1}^{N}\mathbf{x}_n\mathbf{x}_n^{\mathrm{T}}, \qquad \mathbf{S}\mathbf{u}_i = \lambda_i\mathbf{u}_i$$

- Nonlinear transformation ϕ(x) into an M-dimensional feature space, so that each data point xn is thereby projected onto a point ϕ(xn). Aim is to find non-linear principal components using kernels without requiring to explicitly project to the M-dimensional feature space

$$\mathbf{C} = \frac{1}{N}\sum_{n=1}^{N}\boldsymbol{\phi}(\mathbf{x}_n)\boldsymbol{\phi}(\mathbf{x}_n)^{\mathrm{T}} \qquad \mathbf{C}\mathbf{v}_i = \lambda_i\mathbf{v}_i$$

$$\frac{1}{N}\sum_{n=1}^{N}\boldsymbol{\phi}(\mathbf{x}_n)\left\{\boldsymbol{\phi}(\mathbf{x}_n)^{\mathrm{T}}\mathbf{v}_i\right\} = \lambda_i\mathbf{v}_i$$

$$\mathbf{v}_i = \sum_{n=1}^{N}a_{in}\boldsymbol{\phi}(\mathbf{x}_n).$$

$$\frac{1}{N}\sum_{n=1}^{N}\boldsymbol{\phi}(\mathbf{x}_n)\boldsymbol{\phi}(\mathbf{x}_n)^{\mathrm{T}}\sum_{m=1}^{N}a_{im}\boldsymbol{\phi}(\mathbf{x}_m) = \lambda_i\sum_{n=1}^{N}a_{in}\boldsymbol{\phi}(\mathbf{x}_n).$$

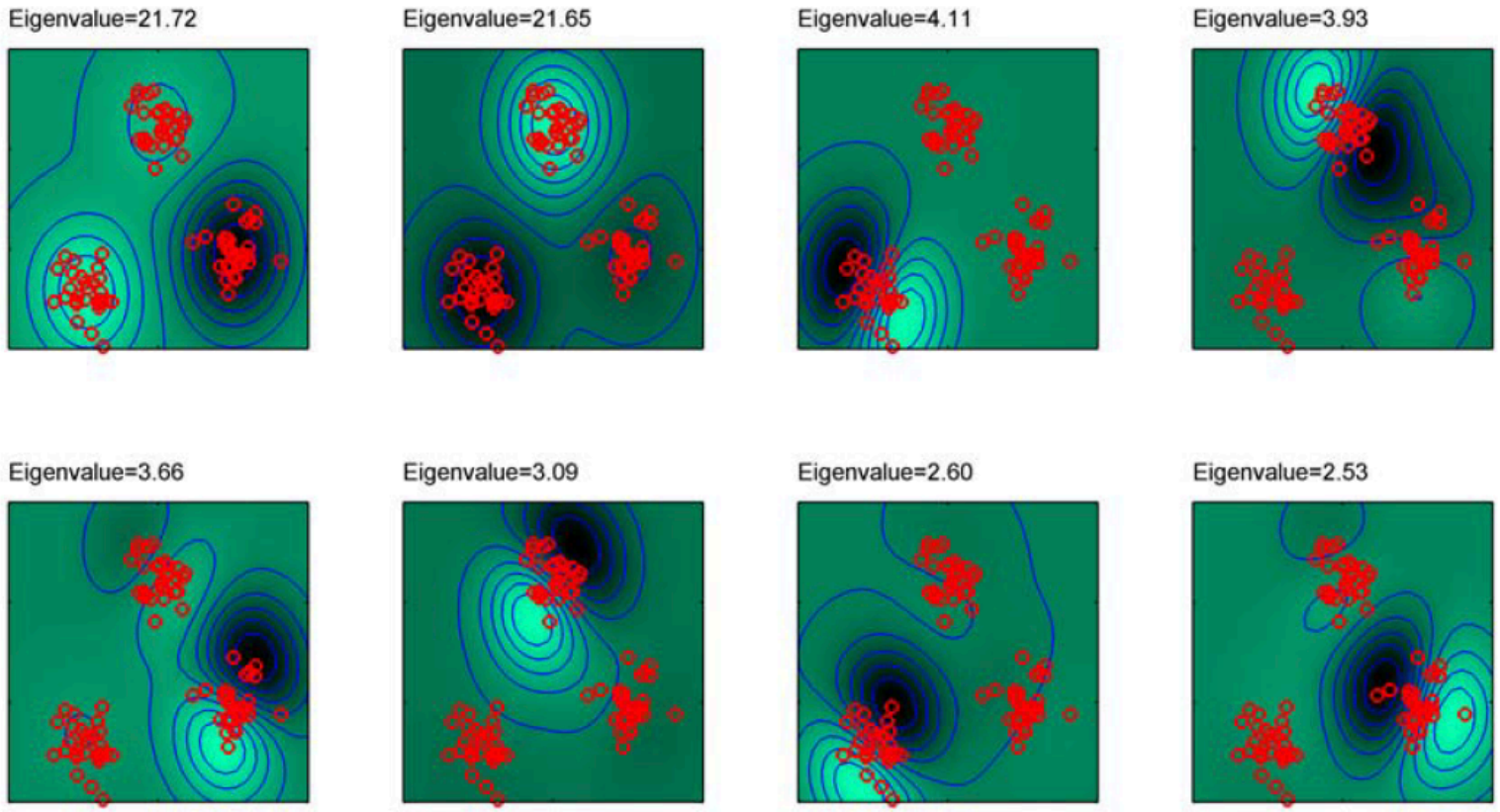
Assuming k(xn, xm) = ϕ(xn)Tϕ(xm)

$$\frac{1}{N}\sum_{n=1}^{N}k(\mathbf{x}_l,\mathbf{x}_n)\sum_{m=1}^{m}a_{im}k(\mathbf{x}_n,\mathbf{x}_m) = \lambda_i\sum_{n=1}^{N}a_{in}k(\mathbf{x}_l,\mathbf{x}_n).$$

$$\mathbf{K}^2\mathbf{a}_i = \lambda_i N\mathbf{K}\mathbf{a}_i \qquad \mathbf{K}\mathbf{a}_i = \lambda_i N\mathbf{a}_i$$

$$y_i(\mathbf{x}) = \boldsymbol{\phi}(\mathbf{x})^{\mathrm{T}}\mathbf{v}_i = \sum_{n=1}^{N}a_{in}\boldsymbol{\phi}(\mathbf{x})^{\mathrm{T}}\boldsymbol{\phi}(\mathbf{x}_n) = \sum_{n=1}^{N}a_{in}k(\mathbf{x},\mathbf{x}_n)$$

Kernel PCA


Eigenvalue=21.72
Eigenvalue=21.65
Eigenvalue=4.11
Eigenvalue=3.93
Eigenvalue=3.66
Eigenvalue=3.09
Eigenvalue=2.60
Eigenvalue=2.53

Kernel Regression

- Many linear models for regression and classification can be reformulated in terms of a dual representation in which the kernel function arises

$$J(\mathbf{w}) = \frac{1}{2}\sum_{n=1}^{N}\left\{\mathbf{w}^{\mathrm{T}}\boldsymbol{\phi}(\mathbf{x}_n) - t_n\right\}^2 + \frac{\lambda}{2}\mathbf{w}^{\mathrm{T}}\mathbf{w}$$

$$\mathbf{w} = -\frac{1}{\lambda}\sum_{n=1}^{N}\left\{\mathbf{w}^{\mathrm{T}}\boldsymbol{\phi}(\mathbf{x}_n) - t_n\right\}\boldsymbol{\phi}(\mathbf{x}_n) = \sum_{n=1}^{N} a_n\boldsymbol{\phi}(\mathbf{x}_n) = \boldsymbol{\Phi}^{\mathrm{T}}\mathbf{a} \qquad a_n = -\frac{1}{\lambda}\left\{\mathbf{w}^{\mathrm{T}}\boldsymbol{\phi}(\mathbf{x}_n) - t_n\right\}.$$

Kernel Regression

- Many linear models for regression and classification can be reformulated in terms of a dual representation in which the kernel function arises

$$J(\mathbf{w}) = \frac{1}{2}\sum_{n=1}^{N}\left\{\mathbf{w}^{\mathrm{T}}\boldsymbol{\phi}(\mathbf{x}_n) - t_n\right\}^2 + \frac{\lambda}{2}\mathbf{w}^{\mathrm{T}}\mathbf{w}$$

$$\mathbf{w} = -\frac{1}{\lambda}\sum_{n=1}^{N}\left\{\mathbf{w}^{\mathrm{T}}\boldsymbol{\phi}(\mathbf{x}_n) - t_n\right\}\boldsymbol{\phi}(\mathbf{x}_n) = \sum_{n=1}^{N} a_n\boldsymbol{\phi}(\mathbf{x}_n) = \boldsymbol{\Phi}^{\mathrm{T}}\mathbf{a} \qquad a_n = -\frac{1}{\lambda}\left\{\mathbf{w}^{\mathrm{T}}\boldsymbol{\phi}(\mathbf{x}_n) - t_n\right\}.$$

$$J(\mathbf{a}) = \frac{1}{2}\mathbf{a}^{\mathrm{T}}\boldsymbol{\Phi}\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Phi}\boldsymbol{\Phi}^{\mathrm{T}}\mathbf{a} - \mathbf{a}^{\mathrm{T}}\boldsymbol{\Phi}\boldsymbol{\Phi}^{\mathrm{T}}\mathbf{t} + \frac{1}{2}\mathbf{t}^{\mathrm{T}}\mathbf{t} + \frac{\lambda}{2}\mathbf{a}^{\mathrm{T}}\boldsymbol{\Phi}\boldsymbol{\Phi}^{\mathrm{T}}\mathbf{a}$$

$$\textit{Gram}\ \text{matrix}\ \mathbf{K} = \boldsymbol{\Phi}\boldsymbol{\Phi}^{\mathrm{T}}, \qquad K_{nm} = \boldsymbol{\phi}(\mathbf{x}_n)^{\mathrm{T}}\boldsymbol{\phi}(\mathbf{x}_m) = k(\mathbf{x}_n, \mathbf{x}_m)$$

$$J(\mathbf{a}) = \frac{1}{2}\mathbf{a}^{\mathrm{T}}\mathbf{K}\mathbf{K}\mathbf{a} - \mathbf{a}^{\mathrm{T}}\mathbf{K}\mathbf{t} + \frac{1}{2}\mathbf{t}^{\mathrm{T}}\mathbf{t} + \frac{\lambda}{2}\mathbf{a}^{\mathrm{T}}\mathbf{K}\mathbf{a}. \qquad \mathbf{a} = (\mathbf{K} + \lambda\mathbf{I}_N)^{-1}\mathbf{t}.$$

$$y(\mathbf{x}) = \mathbf{w}^{\mathrm{T}}\boldsymbol{\phi}(\mathbf{x}) = \mathbf{a}^{\mathrm{T}}\boldsymbol{\Phi}\boldsymbol{\phi}(\mathbf{x}) = \mathbf{k}(\mathbf{x})^{\mathrm{T}}(\mathbf{K} + \lambda\mathbf{I}_N)^{-1}\mathbf{t}$$

we determine the parameter vector a by inverting an N * N matrix, whereas in the original parameter space formulation we had to invert an M * M matrix in order to determine w.