Extracción de Características

Reconocimiento de Patrones – 2013

Extracción de carácterísticas

- Encontrar una transformación de las p medidas, típicamente, a un espacio de menor dimensión

  A: conjunto de transformaciones posibles

  $$\mathbf{x}=[x_1, x_2, \ldots x_p]$$

  $$J(\mathbf{A}^*)= \text{máx. } J(\mathbf{A}(\mathbf{x}))$$

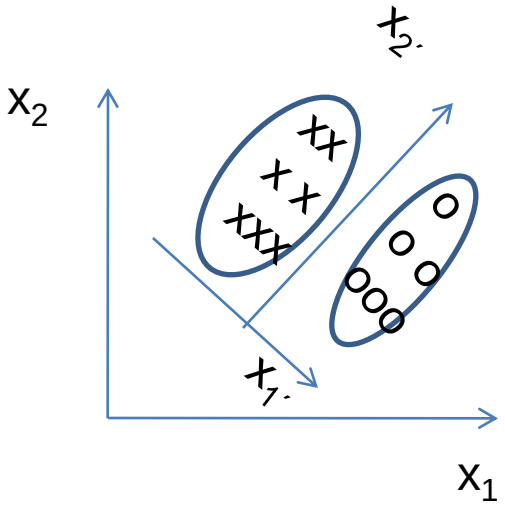
  $$\mathbf{A} \in \boldsymbol{A}$$

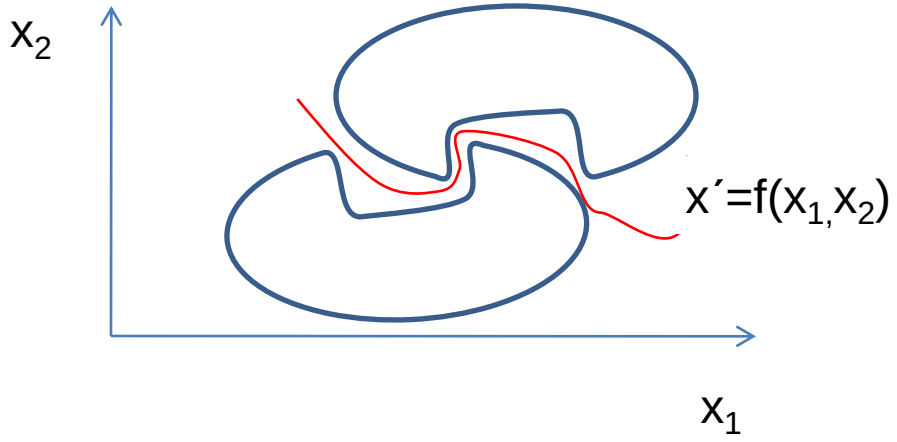
  - Optimizo sobre las transformaciones posibles
  - Nuevas características $\mathbf{y}= \mathbf{A}^*(\mathbf{x})$

Razones extracción de características

1. Proveer conjunto relevante de características al clasificador –mejora de desempeño particularmente en clasificadores simples.
2. Reducir redundancia.
3. Recuperar características latentes significativas
4. Generar mayor comprensión proceso generación de los datos.
5. Visualización de los datos.

Extracción de características





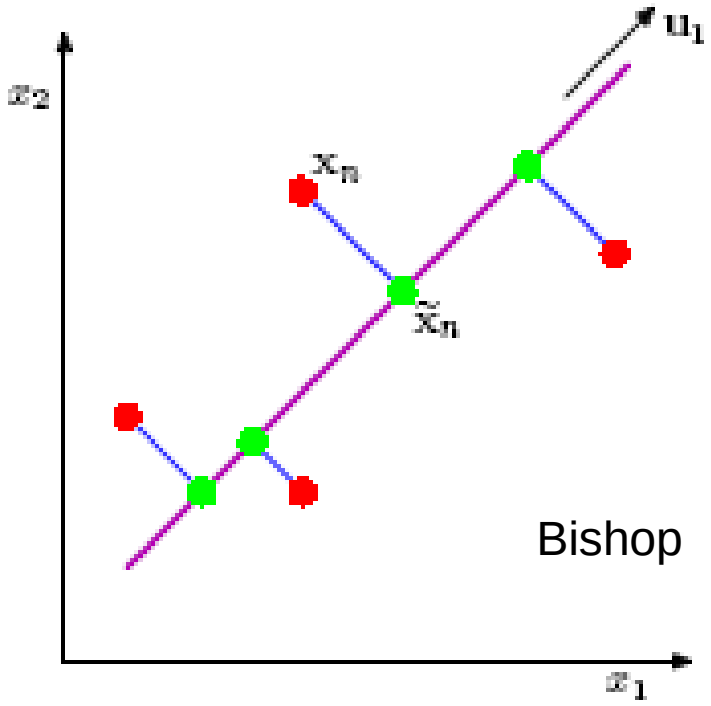
Extracción de características

- Análisis de Componentes Principales (PCA): lineal, no supervisado, basado en vectores propios
- Análisis de discriminantes lineales (LDA): lineal, supervisado, basado en vectores propios.
- Análisis de componentes independientes (ICA): lineal, iterativo, asume independencia fuerte.
- Kernel PCA: no lineal, basado en vectores propios, kernel como producto interno
- Escalado Multidimensional (MDS): no lineal, iterativo

PCA – ANALISIS DE COMPONENTES PRINCIPALES

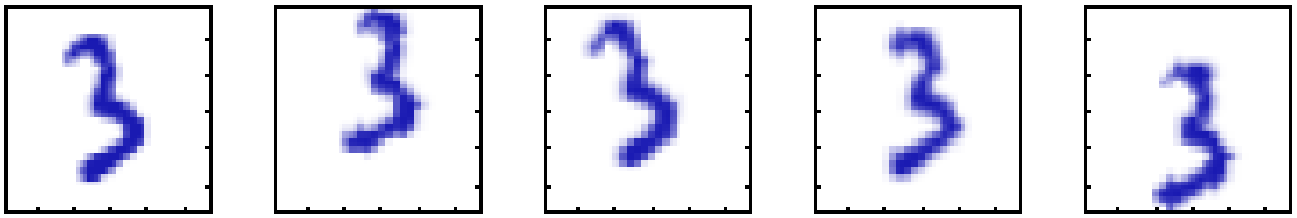
ANALISIS DE COMPONENTES PRINCIPALES –PCA

- Encontrar sub-espacio prinicipal:
- proyección de los datos maximice la varianza de los puntos proyectados
- o minimice error cuadrático entre las proyecciones y las medidas
- Ambas condiciones llevan al mismo resultado.



Transformada de Karhunen- Loève

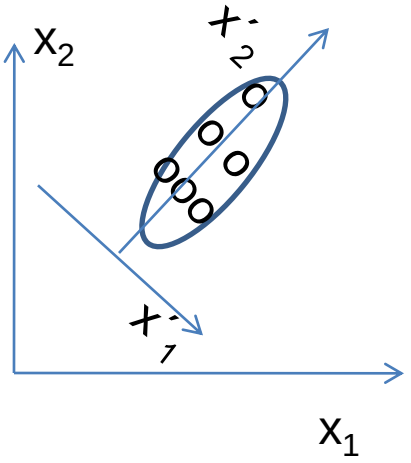
Aplicaciones PCA



Bishop

- Reducción de dimensionalidad
- Extracción de características
- Explicitar la estructura de los datos
- Filtrar ruido
- Compresión de datos con pérdidas
- Visualización de datos.

Análisis de Componentes Principales (PCA)



- Objetivo: derivar un nuevo grupo de variables:
  - combinación lineal de las originales
  - no correlacionadas
- Geométricamente: rotación de ejes a un nuevo sistema de coordenadas
- Ordenar variables según importancia: cantidad de varianza que concentran.
- Desventajas: las nuevas características pueden ser difíciles de interpretar

Deducción del método

$$\mathbf{x} = \left[x_1, x_2, \ldots\ldots\ldots x_p\right]^T \text{ caracteristicas originales}$$

$$\mathbf{y} = \left[y_1, y_2, \ldots\ldots\ldots y_p\right]^T \text{ combinación lineal}$$

$$y_i = \sum_{j=1}^{p} a_{ij} x_j \qquad \mathbf{y} = \mathbf{A}^T \mathbf{x} \qquad \mathbf{A} \text{ matriz de coef.}$$

Asumimos datos centrados en el origen

Deducción del método

- Diferentes formas de resolver PCA:

1. Maximizar señal explicada: Buscar una transformación ortogonal **A** que produzca nuevas variables y_i con valores de varianza máximos.
2. Minimizar redundancia: Buscar una transformación ortogonal **A** que devuelva variables y_i no correlacionadas.
3. Minimizar distancias: Considerar el problema geométricamente, encontrar recta para la cual la suma al cuadrado de las distancias es mínima, luego el plano que mejor ajuste y así sucesivamente.

PCA- Formulación de máxima varianza

- N puntos x_n con dimensión p
- objetivo: proyección datos en un espacio reducido d<p al maximizar la varianza de los datos proyectados.
- Por ahora supondremos d dado. Veremos como determinarlo desde los datos.

1. Inicio: Proyección sobre una dimensión d=1.

- Se $\mathbf{a}_1$ dirección proyección, sin pérdida de generalidad asumiremos $\mathbf{a_1}^{\mathbf{T}}\mathbf{a_1} = |\mathbf{a_1}|^2 = 1$

PCA- Formulación de máxima varianza

- Consideremos la variable: $$y_1 = \sum_{j=1}^{p} a_{1j} x_j$$
- Elegimos $\mathbf{a_1} = [a_{11}, a_{12}, ...a_{1p}]^T$ que maximice la varianza de y_1 sujeto a la restricción $\mathbf{a_1}^{\mathbf{T}}\mathbf{a_1} = |\mathbf{a_1}|^2 = 1$
- Asumimos media de **x**=0,

$$\text{var}(y_1) = E[y_1^2] - E[y_1]^2 = E[\mathbf{a_1^T x x^T a_1}] - E[\mathbf{a_1^T x}]E[\mathbf{x^T a_1}]$$
$$\text{var}(y_1) = \mathbf{a_1^T}\left(E[\mathbf{x x^T}] - E[\mathbf{x}]E[\mathbf{x^T}]\right)\mathbf{a_1} = \mathbf{a_1^T \Sigma a_1}$$

- Encontrar la solución a $\max(\mathbf{a_1^T \Sigma a_1}), \text{sujeto } |\mathbf{a_1}| = 1$

Primer componente principal

- Aplicando multiplicadores de Lagrange,

$$f(\mathbf{a}_1) = \mathbf{a}_1^T \boldsymbol{\Sigma} \mathbf{a}_1 - \lambda(\mathbf{a}_1^T \mathbf{a}_1 - 1)$$

$$\frac{d}{d\mathbf{x}}(\mathbf{x}^T \mathbf{A}\mathbf{x}) = 2\mathbf{A}\mathbf{x} \quad (\mathbf{A} \text{ simétrica})$$

- Diferenciando con respecto a cada componente de $\mathbf{a}_1$ e igualando a 0

$$\boldsymbol{\Sigma}\mathbf{a}_1 - \lambda\mathbf{a}_1 = 0 \qquad (\boldsymbol{\Sigma} - \lambda\mathbf{I})\mathbf{a}_1 = 0$$

- Solución no trivial (vector nulo) $\mathbf{a}_1$ debe ser vector propio de $\boldsymbol{\Sigma}$ con valor propio λ.

Primer componente principal

- $\lambda_1, \lambda_2 \ldots \lambda_p$ valores propios de $\mathbf{\Sigma}$ (no necesariamente todos $\neq$ y no nulos, $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_p \geq 0$, $\mathbf{\Sigma}$ semidefinida positiva. Como:

$$\text{var}(y_1) = \mathbf{a_1^T \Sigma a_1} = \lambda \mathbf{a_1^T a_1} = \lambda$$

- Elegimos $\lambda = \lambda_1$ mayor valor propio de modo de maximizar la varianza y_1: vector propio correspondiente.

  y_1: primer componente principal, mayor varianza de todas las combinaciones lineales posibles de las variables originales.

Segundo componente principal

- Podemos definir PCA en forma incremental eligiendo cada dirección como la de mayor varianza proyectada entre todas las posibles ortogonales a las ya consideradas.
- El segundo componente principal se elige:

$$y_2 = \mathbf{a_2^T x} \qquad \max(\mathrm{var}(y_2)) \text{ sujeto } \quad |\mathbf{a_2}| = 1 \quad \mathbf{a_2}^{\mathbf{T}}\mathbf{a_1} = 0$$

$$\left(\; y_2 \text{ no correlacionado } y_1 : \; E[y_1 y_2] - E[y_1]E[y_2] = 0 \right.$$

$$\left. \mathbf{a_2^T \Sigma a_1} = 0, \quad \mathbf{a_1} \text{ v.p. de } \mathbf{\Sigma} \Rightarrow \mathbf{a_2^T a_1} = 0 \; \right)$$

PCA

- Usando nuevamente multiplicadores de Lagrange:

$$\mathbf{a}_2^{\mathbf{T}}\boldsymbol{\Sigma}\mathbf{a}_2 - \mu(\mathbf{a}_2^{\mathbf{T}}\mathbf{a}_2 - 1) - \eta\mathbf{a}_2^{\mathbf{T}}\mathbf{a}_1 = 0$$

diferenciando respecto a a_{2j}

$$2\boldsymbol{\Sigma}\mathbf{a}_2 - 2\mu\mathbf{a}_2 - \eta\mathbf{a}_1 = 0$$

$$\times \mathbf{a}_1^{\mathbf{T}} \qquad 2\mathbf{a}_1^{\mathbf{T}}\boldsymbol{\Sigma}\mathbf{a}_2 - \eta = 0 \Rightarrow \eta = 0$$

$$\Rightarrow \boldsymbol{\Sigma}\mathbf{a}_2 = \mu\mathbf{a}_2 \quad \mathbf{a}_2 \text{ es v.p.de } \boldsymbol{\Sigma}$$

PCA

- Como queremos maximizar: $\mathbf{a}_2^T \boldsymbol{\Sigma} \mathbf{a}_2 = \mu \mathbf{a}_2^T \mathbf{a}_2 = \mu$

- $\mathbf{a}_2$:vector propio correspondiente al siguiente mayor valor propio $\mu = \lambda_2$.
- Podemos continuar hasta determinar el k-ésimo vector propio.
- Si $\lambda_i = \lambda_j$, la solución no es única. Siempre es posible encontrar un conjunto ortonormal de vectores propios para una matriz real simétrica con valores propios no negativos.
- En notación matricial: $\mathbf{y} = \mathbf{A}^T \mathbf{x}$ $\mathbf{A} = [\mathbf{a}_1, \mathbf{a}_2 ...., \mathbf{a}_p]$

  Columnas de $\mathbf{A}$ vectores propios de $\boldsymbol{\Sigma}$

Formulación de mínimo error

Basada en la minimización del error de proyección.

Se considera una base ortonormal $\mathbf{a}_i^{\mathrm{T}}\mathbf{a}_j = \delta_{ij}$ completa

$\mathbf{x}_n = \sum_{i=1}^{p} \left(\mathbf{x}_n{}^T \mathbf{a}_i\right)\mathbf{a}_i$ si aproximamos $\tilde{\mathbf{x}}_n = \sum_{i=1}^{d} \left(\mathbf{x}_n{}^T \mathbf{a}_i\right)\mathbf{a}_i$

$$J = \frac{1}{N}\sum_{n=1}^{N} \left\|\mathbf{x}_n - \tilde{\mathbf{x}}_n\right\|^2 = \sum_{i=d+1}^{p} \mathbf{a}_i^{\mathrm{T}} \boldsymbol{\Sigma} \mathbf{a}_i$$

La minimización de J para un p y d arbitario

es la determinada por los vectores propios de $\boldsymbol{\Sigma}$.

$J = \sum_{i=d+1}^{p} \lambda_i$ J es mínimo eligiendo los λ_i más grande para

determinar el subespacio.

Si p = d no hay reducción de dimensionalidad - rotación de ejes.

Reducción de dimensión con PCA

- ¿Cómo determino d?
- Para que la distorsión J esté por debajo de un cierto valor :

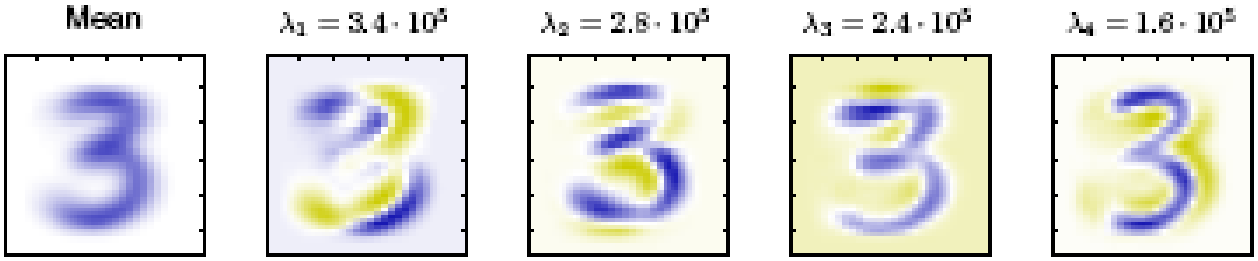
$$\sum_{i=1}^{p} \mathrm{var}(y_i) = \sum_{i=1}^{p} \lambda_i$$

$$\sum_{i=1}^{d} \lambda_i \geq \alpha \sum_{i=1}^{p} \lambda_i \geq \sum_{i=1}^{d-1} \lambda_i$$

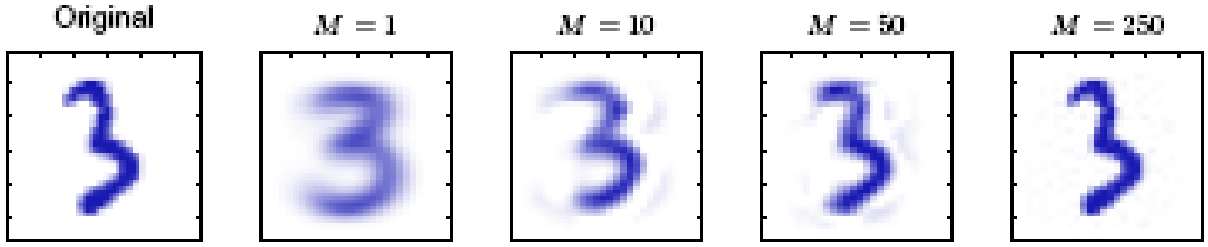
$$\mathbf{y} = [y_1, y_2 .... y_d]^T \qquad \mathbf{y} = \mathbf{A}_d^{\mathbf{T}} \mathbf{x} \qquad \mathbf{A_d} : p \times d$$

- El valor de α depende del problema (α: 70%-90%)

Reducción de dimensión con PCA



The mean vector $\bar{x}$ along with the first four PCA eigenvectors $u_1, \ldots, u_4$ 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 $M = D = 28 \times 28 = 784$.

Bishop

Reducción de dimensión con PCA

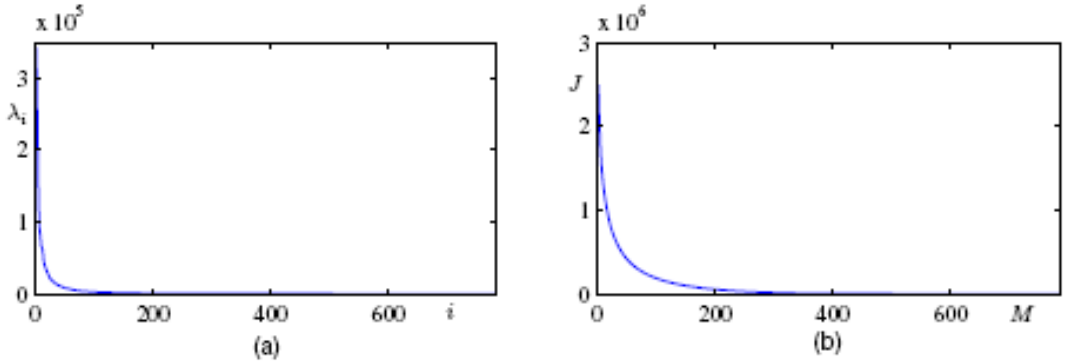


Figure 12.4 (a) Plot of the eigenvalue spectrum for the off-line digits data set. (b) Plot of the sum of the discarded eigenvalues, which represents the sum-of-squares distortion J introduced by projecting the data onto a principal component subspace of dimensionality M.

Bishop

- Analizar como varían los valores propios: buscar corte abrupto.
- Distorsión J cuando elijo d-dimensión. Grafico paras distintos valores de d.
- Asumimos $\boldsymbol{\Sigma}$ conocido. En la practica tenemos una estimación a partir de un conjunto de muestras.
- PCA es dependiente de la escala usada para medir las variables originales, es necesario estandarizar (media nula, varianza unidad)

PCA- Minimizar correlación

- Transformada A que devuelva variables no correlacionadas.

$$\mathbf{y} = \mathbf{A}^T\mathbf{x} \quad \boldsymbol{\Sigma} = \mathbf{y}\mathbf{y}^T = \mathbf{A}^T\mathbf{x}(\mathbf{A}^T\mathbf{x})^T = \mathbf{A}^T(\mathbf{x}\mathbf{x}^T)\mathbf{A}$$

$$\boldsymbol{\Sigma}_\mathbf{x} = \mathbf{B}\mathbf{D}\mathbf{B}^T \quad \mathbf{D}: \text{diagonal} \quad \mathbf{B}: \text{matriz} \quad \text{v.p.}$$

$$\mathbf{A} = \mathbf{B} \quad \boldsymbol{\Sigma}_\mathbf{y} = \mathbf{A}^T\boldsymbol{\Sigma}_\mathbf{x}\mathbf{A} = \mathbf{A}^T\mathbf{A}\mathbf{D}\mathbf{A}^T\mathbf{A} = \mathbf{D}$$

- Suposiciones:
  - Linealidad: busco cambio de base.
  - Extensión: transformación no lineal
  - Componentes principales ortogonales.
  - Media y varianza describen enteramente la distribución (ej: gaus)

PCA con dimensión de datos muy alta

- Costo computacional calculo vectores propios de una matriz *dxd* es $O(d^3)$, existen algoritmos más eficientes.
- $\mathbf{X}$: pxn , $\mathbf{X}\mathbf{X}^T$:pxp, puede ocurrir que $n<p$, n puntos definen un subespacio de dimensión a lo sumo $n-1$. Al aplicar PCA $p-n+1$ valores propios son 0. Costo $O(p^3)$.

$$\mathbf{\Sigma} = \frac{1}{n-1}\mathbf{X}\mathbf{X}^T \qquad \frac{1}{n-1}\mathbf{X}\mathbf{X}^{\mathbf{T}}\mathbf{u} = \lambda\mathbf{u}$$

$$\frac{1}{n-1}\mathbf{X}^{\mathbf{T}}\mathbf{X}(\mathbf{X}^{\mathbf{T}}\mathbf{u}) = \lambda(\mathbf{X}^{\mathbf{T}}\mathbf{u}) \Rightarrow \frac{1}{n-1}\mathbf{X}^{\mathbf{T}}\mathbf{X}\mathbf{v} = \lambda\mathbf{v}$$

$$\mathbf{\Sigma}(\mathbf{X}\ \mathbf{v}) = \lambda(\mathbf{X}\ \mathbf{v}) \quad \mathbf{u} = \frac{1}{\left((\mathrm{n}-1)\lambda\right)^{1/2}}\mathbf{X}\ \mathbf{v}$$

- Calculo los valores y vectores propios de $\mathbf{X}\mathbf{X}^{\mathrm{T}}$ y luego determino los vectores propios $\mathbf{u}$ en el espacio original de datos normalizando.

LDA- ANÁLISIS DISCRIMINANTES LINEALES

Funciones discriminantes

$y(\mathbf{x}) = \mathbf{w}^T \mathbf{x} + \mathbf{w}_0$

$\mathbf{w}$: vector de pesos

w_0 : bias o threshold

asigno x a c_1 si $y(\mathbf{x}) > 0$ y c_2 en otro caso.

La superficie de decisión corresponde a un hiperplano de dimensión d-1 si el espacio de entrada es de dimensión d.

LDA- Análisis Discriminante Lineal

Discriminante Lineal de Fisher

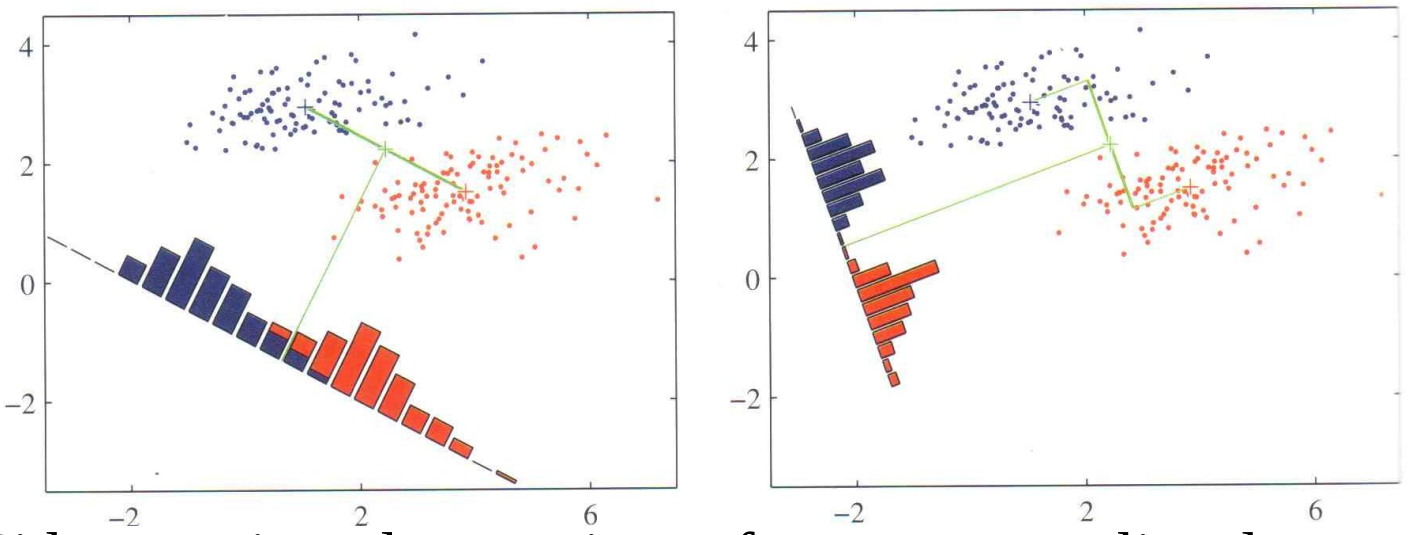
- Objetivo: Seleccionar una proyección que maximice separabilidad entre clases. Método supervisado.
- Ejemplo- 2 clases: c_1 y c_2 con N_1 y N_2 patrones.
- Medias:

$$\mathbf{m_1} = \frac{\sum_{n \in c_1} \mathbf{x_n}}{N_1} \qquad \mathbf{m_2} = \frac{\sum_{n \in c_2} \mathbf{x_n}}{N_2}$$

- Si consideramos la proyección sobre una dimensión, una medida de separabilidad es la separación de las medias luego de proyectar:

$$y = \mathbf{w}^{\mathbf{T}}\mathbf{x} \qquad m_2 - m_1 = \mathbf{w}^{\mathbf{T}}(\mathbf{m_2} - \mathbf{m_1})$$

- Restringiendo a los **w** con módulo 1 y usando multiplicadores de Lagrange: **w** proporcional **$(m_2 - m_1)$.**



- Si las matrices de covarianza fuertemene no lineal, no son diagonales puede existir solapamiento considerable luego de proyectar y a pesar de ser linealmente separables.
- Discriminante de Fisher busca maximizar separación de medias proyectadas, pequeña varianza dentro de cada clase, minimizar solapamiento entre clases.

Varianza intra- clase:

$$s_k^2 = \sum_{n \in c_k} (y_n - m_k)^2 \qquad y_n = \mathbf{w}^T \mathbf{x}_n$$

Criterio de Fisher:

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

Varianza entre clases

Varianza intra clases

$$J(\mathbf{w}) = \frac{\mathbf{w}^T \mathbf{S}_B \mathbf{w}}{\mathbf{w}^T \mathbf{S}_w \mathbf{w}} \qquad \text{con} \qquad \mathbf{S}_B = (\mathbf{m}_2 - \mathbf{m}_1)(\mathbf{m}_2 - \mathbf{m}_1)^T$$

$$\mathbf{S}_w = \sum_{n \in c_1} (\mathbf{x}_n - \mathbf{m}_1)(\mathbf{x}_n - \mathbf{m}_1)^T + \sum_{n \in c_2} (\mathbf{x}_n - \mathbf{m}_2)(\mathbf{x}_n - \mathbf{m}_2)^T$$

Diferenciando respecto a $\mathbf{w}$: $J(\mathbf{w})$ máximo si:

$$(\mathbf{w}^T \mathbf{S}_B \mathbf{w}) \mathbf{S}_w \mathbf{w} = (\mathbf{w}^T \mathbf{S}_w \mathbf{w}) \mathbf{S}_B \mathbf{w} \qquad \mathbf{S}_B \mathbf{w} = \lambda \mathbf{S}_w \mathbf{w}$$

- $\mathbf{S}_w$: simétrica y semidefinida positiva, usualmente no singular si n>p.

$$\mathbf{S}_w^{-1}\mathbf{S}_B\mathbf{w} = \lambda\mathbf{w}$$

- En este caso particular no es necesario resolver el problema de vectores y valores propios ya que :

$$\mathbf{S}_B\mathbf{w} = (\mathbf{m}_2 - \mathbf{m}_1)(\mathbf{m}_2 - \mathbf{m}_1)^T\mathbf{w} \propto (\mathbf{m}_2 - \mathbf{m}_1)$$

$$\mathbf{w} \propto \mathbf{S}_w^{-1}(\mathbf{m}_2 - \mathbf{m}_1)$$

- No importa la magnitud de **w**. Determino una dirección para la proyección de los datos proyectados.

Análisis de Discriminantes Lineales Múltiples

- La generalización a un problema de C clases involucra c-1 funciones discriminantes. Se proyecta de un espacio de dimensión p a un espacio de dimensión c-1, asumiendo p≥c.
- Generalización de $\mathbf{S}_w$ es directa:

$$\mathbf{S}_w = \sum_{i=1}^{c} \mathbf{S}_i \qquad \mathbf{S}_i = \sum_{n \in c_i} (\mathbf{x}_n - \mathbf{m}_i)(\mathbf{x}_n - \mathbf{m}_i)^T \qquad \mathbf{m}_i = \frac{1}{N_i} \sum_{n \in c_i} \mathbf{x}_n$$

- $\mathbf{S}_B$: consideramos media y covarianza global de los datos

$$\mathbf{m} = \frac{1}{N} \sum_j \mathbf{x}_j = \frac{1}{N} \sum_{i=1}^{c} N_i \mathbf{m}_i \qquad \mathbf{S}_T = \sum_j (\mathbf{x}_j - \mathbf{m})(\mathbf{x}_j - \mathbf{m})^T$$

LDA Múltiples Generalización

$$\mathbf{S}_T = \sum_{i=1}^{c} \sum_{x \in c_i} (\mathbf{x} - \mathbf{m_i} + \mathbf{m_i} - \mathbf{m})(\mathbf{x} - \mathbf{m_i} + \mathbf{m_i} - \mathbf{m})^T$$

$$= \sum_{i=1}^{c} \sum_{x \in c_i} (\mathbf{x} - \mathbf{m}_i)(\mathbf{x} - \mathbf{m_i})^T + \sum_{i=1}^{c} \sum_{x \in c_i} (\mathbf{m_i} - \mathbf{m})(\mathbf{m_i} - \mathbf{m})^T$$

$$\mathbf{S}_T = \mathbf{S}_w + \sum_{i=1}^{c} N_i (\mathbf{m}_i - \mathbf{m})(\mathbf{m}_i - \mathbf{m})^T = \mathbf{S}_w + \mathbf{S}_B$$

LDA Múltiples Generalizado

$$\mathbf{W} = [\mathbf{w}_1, \mathbf{w}_2 ..... \mathbf{w}_{c-1}] \quad \mathbf{W} : c-1 \times p \quad \mathbf{y} = \mathbf{W}^T \mathbf{x}$$

$$\text{Def :} \quad \boldsymbol{\mu}_i = \frac{1}{N_i} \sum_{n \in c_n} \mathbf{y}_n \qquad \boldsymbol{\mu} = \frac{1}{N} \sum_{i=1}^{c} N_i \boldsymbol{\mu}_i$$

$$\tilde{\mathbf{S}}_{\mathbf{W}} = \sum_{i=1}^{c} \sum_{n \in c_n} (\mathbf{y} - \boldsymbol{\mu}_i)(\mathbf{y} - \boldsymbol{\mu}_i)^T \quad \tilde{\mathbf{S}}_B = \sum_{i=1}^{c} N_i (\boldsymbol{\mu}_i - \boldsymbol{\mu})(\boldsymbol{\mu}_i - \boldsymbol{\mu})^T$$

$$\tilde{\mathbf{S}}_B = \mathbf{W}^T \mathbf{S}_B \mathbf{W} \qquad \tilde{\mathbf{S}}_{\mathbf{W}} = \mathbf{W}^T \mathbf{S}_{\mathbf{W}} \mathbf{W}$$

- Necesitamos un medida de la razón entre la varianza entre clases y la varianza intra-clases. Una medida escalar simple es el determinante de la matriz otra la traza de las matrices.

$$J(\mathbf{w}) = \frac{\det(\tilde{\mathbf{S}}_B)}{\det(\tilde{\mathbf{S}}_\mathbf{W})} = \frac{\det(\mathbf{W}^T \mathbf{S}_B \mathbf{W})}{\det(\mathbf{W}^T \mathbf{S}_\mathbf{W} \mathbf{W})}$$

$$\mathbf{S}_B \mathbf{w}_i = \lambda_i \mathbf{S}_\mathbf{W} \mathbf{w}_i$$

- Problema de vectores propios generalizado, las columnas de **W** son los vectores propios generalizados correspondientes a los valores propios mayores de:
- Si $\mathbf{S}_\mathbf{w}$ no es singular:

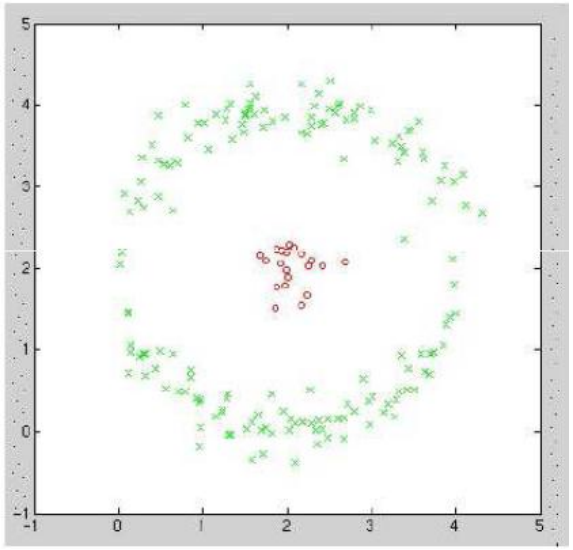
$$\mathbf{S}_W^{-1} \mathbf{S}_B \mathbf{w}_i = \lambda_i \mathbf{w}_i$$

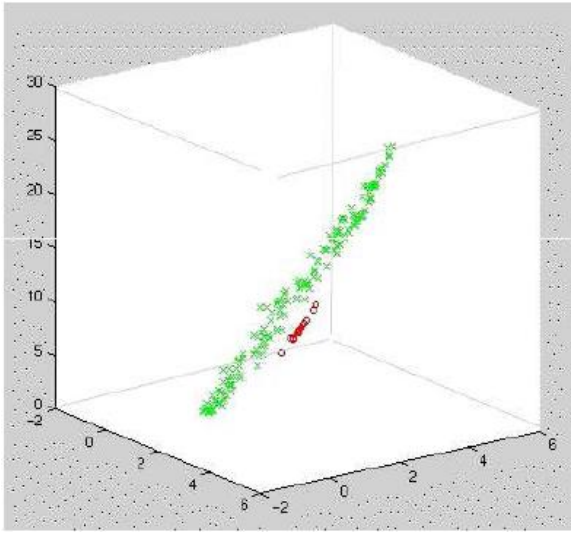
LDA Múltiples Generalizados

- Nota: $\mathbf{S}_B$ es la suma de c matrices que son el producto de dos vectores por ende de rango 1, solo (c-1) son independientes. Por lo tanto $\mathbf{S}_B$ es a lo sumo de rango c-1 y consecuentemente tengo a lo sumo c-1 vectores propios no nulos.
- Obs: Las clases no tienen porqué ser linealmente separables. Alternativa SVM: mapeo no lineal y discriminantes lineales.

KERNEL PCA

La magia de la alta dimensión





$$\Phi : \mathbf{R}^2 \rightarrow \mathbf{R}^3$$

$$(x_1, x_2) \mapsto (x_1, x_2, x_1^2 + x_2^2)$$

La teoría de VC (Vapnik-Chervonenkis) dice que a menudo un mapeo a un espacio de mayor dimensión tiene mayor poder de clasificación que el espacio de entrada

Kernel Trick

- Un mapeo de alta dimensión puede incrementar el tiempo computacional en forma seria.
- ¿Podemos evitar ese costo y tener el beneficio de la alta dimensión ?

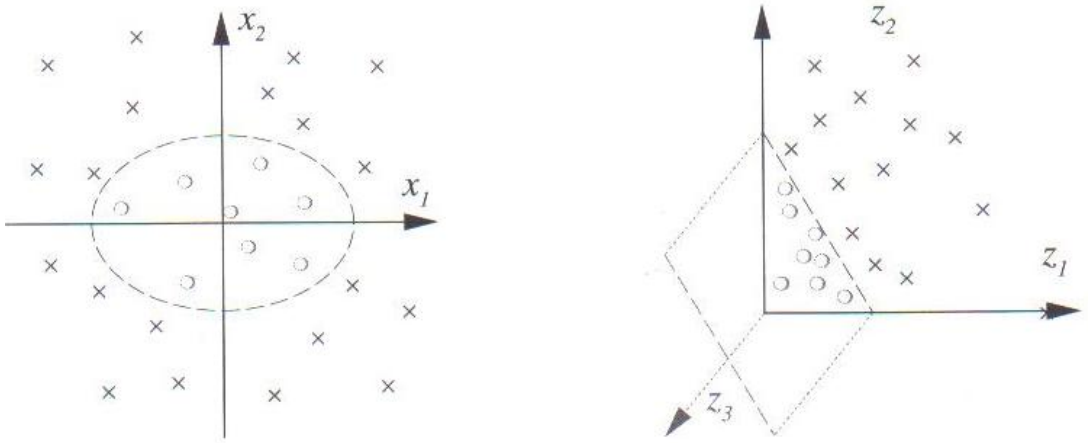
⟹ Usar algoritmos que solo requieran producto escalar entre vectores de H y elegir el mapeo de forma que se puedan calcular en el espacio original mediante una función kernel.

$\phi : \mathrm{X} \to H \quad \mathbf{x} \to \phi(\mathbf{x})$ función kernel dada por:

$k(\mathbf{x}, \mathbf{x}') = \phi(\mathbf{x})^T \phi(\mathbf{x}')$ función simétrica.

Kernel

- Métodos de Kernel: mapea los patrones en un espacio de alta dimensión. Se espera lograr ventajas por ejemplo en la separabilidad o la linealidad de los datos.



- La elección de ϕ (típicamente no lineal) permite buscar una representación adecuada para un problema determinado, definir nuevas medidas de similitud y algoritmos de aprendizaje.
- Función kernel: puede expresarse como un producto interno usualmente en un espacio diferente.

Kernel

- El vector característica solo entra en forma de producto escalar al algoritmo. En lugar de elegir $\phi(x)$ en forma explicita, elijo $k(x,x')$.
- Hay numerosas funciones comúnmente usadas:

$k(\mathbf{x},\mathbf{x}') = k(\mathbf{x}-\mathbf{x}')$: kernels estacionarios

invariantes a traslaciones

$k(\mathbf{x},\mathbf{x}') = k(\|\mathbf{x}-\mathbf{x}'\|)$: kernels homogeneos

funciones base radiales, solo dependen de distancia.

Construcción directa del kernel

- Debemos asegurar que es una función válida, que corresponde a un producto escalar en un espacio de característica (de dimensión finita o infinita)
- Ej: $k(\mathbf{x},\mathbf{z}) = (\mathbf{x}^T\mathbf{z})^2$ en 2 dimensiones $\mathbf{x} = (x_1, x_2)$

  vamos a buscar mapeo no lineal.

  $k(\mathbf{x},\mathbf{z}) = (\mathbf{x}^T\mathbf{z})^2 = (x_1 z_1 + x_2 z_2)^2$

  $= (x_1^2 z_1^2 + 2x_1 z_1 x_2 z_2 + x_2^2 z_2^2)$

  $= (x_1^2, \sqrt{2}x_1x_2, x_2^2)(z_1^2, \sqrt{2}z_1z_2, z_2^2)^T = \phi(\mathbf{x})^T \phi(\mathbf{z})$

  El mapeo lleva : $\phi(\mathbf{x}) = (x_1^2, \sqrt{2}x_1x_2, x_2^2)^T$

  Todos los términos de segundo orden con pesos específicos.

Kernel trick:

- Necesitamos forma más fácil de saber si una función es un kernel válido sin construirlo en forma explícita. Condición necesaria y suficiente para saber si k(**x**,**x**´) es un kernel válido es que la matríz GRAM **K** cuyos elementos están definidos por $k(\mathbf{x}_n,\mathbf{x}_m')$ sea semidefinida positiva.

- Relevante: Seleccionar el kernel apropiado y parámetros

Kernels utilizados:

- Polinómico: $k(\mathbf{x},\mathbf{x}') = \langle \mathbf{x},\mathbf{x}' \rangle^d$

- Funciones de base radial: $k(\mathbf{x},\mathbf{x}') = f(d(\mathbf{x},\mathbf{x}'))$

  d: típicamente proviene del producto interno

$$d(\mathbf{x},\mathbf{x}') = |\mathbf{x}-\mathbf{x}'| = \sqrt{\langle \mathbf{x}-\mathbf{x}',\mathbf{x}-\mathbf{x}' \rangle}$$

- Funciones de base radial Gaussiana:

$$k(\mathbf{x},\mathbf{x}') = e^{-\gamma|\mathbf{x}-\mathbf{x}'|^2} = e^{-\gamma\langle \mathbf{x}-\mathbf{x}',\mathbf{x}-\mathbf{x}' \rangle}$$

Kernel PCA

- Extender el análisis convencional de PCA a un espacio de características de mayor dimensión usando “kernel trick”

- Se pueden extraer hasta m (número de muestras) componentes principales no lineales sin costo computacional de trabajar en espacio de mayor dimensión.

Kernel PCA

- Versión no lineal de PCA:
  - Mapeo a un espacio de dimensión mayor. $\phi(x)$: X→H
  - Aplico PCA en ese nuevo espacio, PCA en H
- Formulamos PCA en términos de productos internos:

$$\Sigma = \frac{1}{m}\sum_{j=1}^{m}\mathbf{x}_j\mathbf{x}_j^T \qquad \lambda\mathbf{v} = \mathbf{\Sigma}\mathbf{v} \qquad \sum_{i=1}^{m}\mathbf{x}_i = 0$$

$$\mathbf{\Sigma}\mathbf{v} = \left(\frac{1}{m}\sum_{j=1}^{m}\mathbf{x}_j\mathbf{x}_j^T\right)\mathbf{v} = \lambda\mathbf{v}$$

$$\mathbf{v} = \frac{1}{\lambda m}\sum_{j=1}^{m}\langle\mathbf{x}_j,\mathbf{v}\rangle\mathbf{x}_j \qquad (\mathbf{x}_j\mathbf{x}_j^T)\mathbf{v} = \langle\mathbf{x}_j,\mathbf{v}\rangle\mathbf{x}_j$$

$$\mathbf{v} = \sum_{j=1}^{m} \alpha_j \mathbf{x}_j$$

- Todo **v** es combinación lineal de los $\mathbf{x}_j$, todo **v** con $\lambda \neq 0$, pertenece al espacio generado por $\mathbf{x}_1, \mathbf{x}_2, \ldots \mathbf{x}_m$

$$\lambda \langle \mathbf{x}_i, \mathbf{v} \rangle = \langle \mathbf{x}_i, \boldsymbol{\Sigma} \mathbf{v} \rangle \qquad \forall i = 1, \ldots, m$$

- Introducimos el mapeo:

$$\phi : \mathrm{X} \to H, \qquad \mathbf{x} \to \phi(\mathbf{x}) \qquad \sum \phi(\mathbf{x}_i) = 0$$

$$\mathbf{C} = \frac{1}{m} \sum_{j=1}^{m} \phi(\mathbf{x}_j) \phi(\mathbf{x}_j)^T \qquad \lambda \mathbf{V} = \mathbf{C} \mathbf{V}$$

$$\mathbf{V} = \sum_{i=1}^{m} \alpha_i \phi(\mathbf{x}_i)^T$$

$$\lambda \langle \phi(\mathbf{x}_n), \mathbf{V} \rangle = \langle \phi(\mathbf{x}_n), \mathbf{C} \mathbf{V} \rangle \qquad \forall n = 1, \ldots., m$$

Combinando las 2 últimas ecuaciones:

$$\lambda\left\langle \phi(\mathbf{x}_n), \sum_{i=1}^{m} \alpha_i \phi(\mathbf{x}_i) \right\rangle = \left\langle \phi(\mathbf{x}_n), \frac{1}{m} \sum_{j=1}^{m} \phi(\mathbf{x}_i)\phi(\mathbf{x}_i)^T \sum_{i=1}^{m} \alpha_i \phi(\mathbf{x}_i) \right\rangle$$

$$\lambda \sum_{i=1}^{m} \alpha_i \left\langle \phi(\mathbf{x}_n), \phi(\mathbf{x}_i) \right\rangle = \frac{1}{m} \sum_{i=1}^{m} \alpha_i \left\langle \phi(\mathbf{x}_n), \sum_{j=1}^{m} \phi(\mathbf{x}_j) \left\langle \phi(\mathbf{x}_j), \phi(\mathbf{x}_j) \right\rangle \right\rangle$$

$$\forall n = 1..., m$$

Definimos matríz Gram K: $k_{ij} = \left\langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \right\rangle$

$m\lambda \mathbf{K}\boldsymbol{\alpha} = \mathbf{K}^2\boldsymbol{\alpha}$ $\qquad \boldsymbol{\alpha} = \left(\alpha_1, ... \alpha_m\right)^T$

equivalente: $m\lambda\boldsymbol{\alpha} = \mathbf{K}\boldsymbol{\alpha}$

- Sean $\lambda_1 \geq \lambda_2 \geq ..... \geq \lambda_m$ valores p.de $\mathbf{K}$

  $\boldsymbol{\alpha}^1, \boldsymbol{\alpha}^2, ... \boldsymbol{\alpha}^m$ vectores p. de $\mathbf{K}$

- λ_p último valor propio no cero
- Se normalizan los vectores $\boldsymbol{\alpha}^i$ para que los vectores $\mathbf{V}$ en H estén normalizados.

$$\left\langle \mathbf{V}^n, \mathbf{V}^n \right\rangle = 1 \qquad \forall n = 1 \ldots p$$

$$1 = \sum_{i,j}^{m} \alpha_i^n \alpha_j^n \left\langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \right\rangle = \sum_{i,j}^{m} \alpha_i^n \alpha_j^n \mathbf{K}_{ij} =$$

$$= \left\langle \boldsymbol{\alpha}^n, \mathbf{K}\boldsymbol{\alpha}^n \right\rangle = \lambda \left\langle \boldsymbol{\alpha}^n, \boldsymbol{\alpha}^n \right\rangle$$

Kernel PCA

- Para la extracción de los componentes principales necesitamos calcular las proyecciones sobre los vectores propios $\mathbf{V}^n$ en H (n=1…p)
- Si **x** es un punto de test con imagen $\phi(\mathbf{x})$ en H

$$y_n = \left\langle \mathbf{V}^n, \phi(\mathbf{x}) \right\rangle = \sum_{i=1}^{m} \alpha_i^n \left\langle \phi(\mathbf{x}_i), \phi(\mathbf{x}) \right\rangle = \sum_{i=1}^{m} \alpha_i^n k(\mathbf{x}_i, \mathbf{x})$$

- Son los componentes principales no lineales o características correspondientes al mapeo ϕ.

Resumen del algoritmo Kernel PCA

- Calcular la matriz de Gram $\mathbf{K}_{ij}=k(x_i, x_j)$
- Diagonalizar **K** y normalizar los vectores propios $\boldsymbol{\alpha}^n$,

- Extraer los componentes principales de un punto **x**.

$$\left\langle \mathbf{V}^n, \phi(\mathbf{x}) \right\rangle = \sum_{i=1}^{m} \alpha_i^n k(\mathbf{x_i}, \mathbf{x}) \qquad n = 1,.., p$$

- Centrado: asumimos datos centrados en H. Centrado no explicito, sino implícito modificando levemente las ecuaciones, logra invarianza a traslaciones.

$$\tilde{\mathbf{K}} = (\mathbf{K} - \mathbf{1}_m\mathbf{K} - \mathbf{K}\mathbf{1}_m + \mathbf{1}_m\mathbf{K}\mathbf{1}_m) \quad con \quad (\mathbf{1}_m)_{ij} = 1/m \;\; \forall i, j$$

Resumen de Kernel PCA

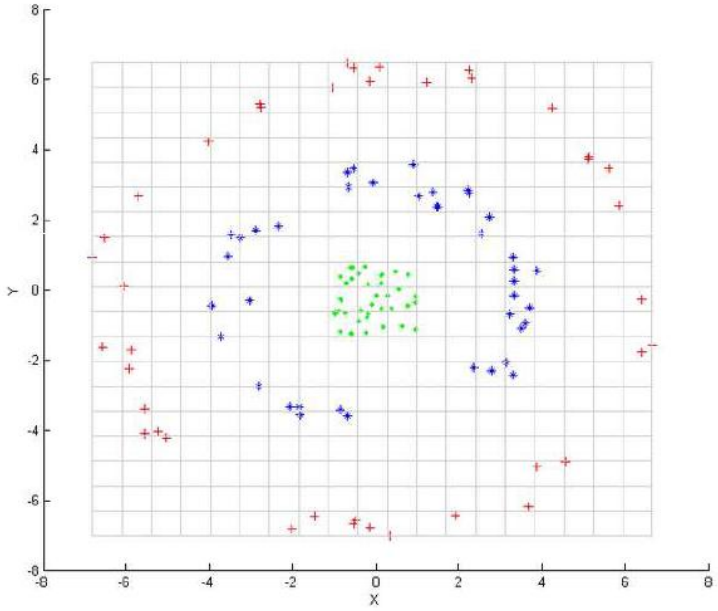
- Elegir un kenel k,
- Construir la matriz Gram $\mathbf{K}_{ij}=k(x_i, x_j)$
- Encontrar los vectores y valores propios:

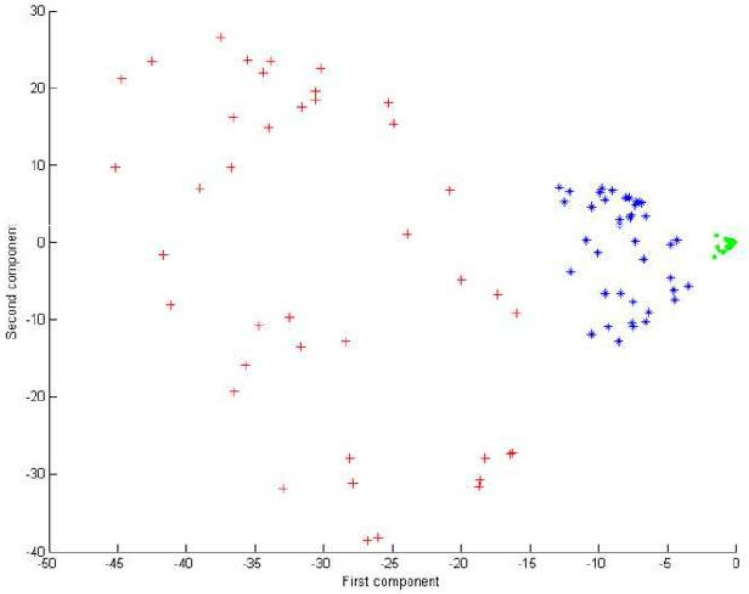
$$\tilde{\mathbf{K}}\boldsymbol{\alpha}^n = \lambda_n \boldsymbol{\alpha}^n$$

- Representar cada punto como:

$$y_n = \sum_{i=1}^{m} \alpha_i^n k(\mathbf{x}, \mathbf{x}_i) \quad \text{n} = 1 \ldots \text{p}$$

Ejemplo: Kernel PCA





Ejemplo: Eliminación de ruido

Original data

Data corrupted with Gaussian noise

Result after linear PCA

Result after kernel PCA, Gaussian kernel

ICA- ANÁLISIS COMPONENTES INDEPENDIENTES

Análisis de Componentes Independientes – ICA

- Busca las direcciones más independientes en lugar de minimizar los errores de representación (suma del error cuadrático) como PCA.
- Se aplica por ejemplo a la separación de mezclas en voz
- Separación de canales EEG. Se presenta una letra (B, H, J, C, F, o K) cada intervalos de 1500-ms. Responde pulsando uno de dos pulsadores indicando si la letra es la misma que la vista hace dos intervalos. Dependiendo de la respuesta, el sujeto gana o pierde puntos. Señales de la de sensores cercanos se parecen mucho. Suposición :señales registradas en cada electrodo del cuero cabelludo son una mezcla de los potenciales independientes.

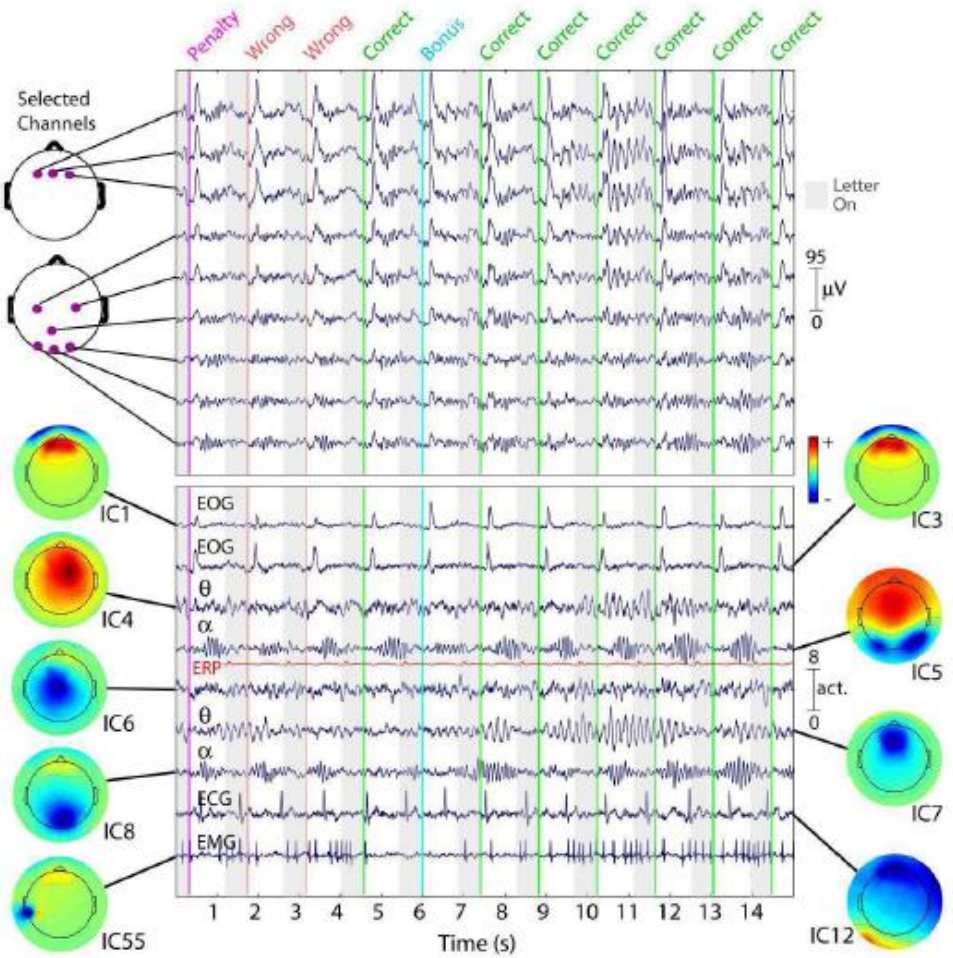


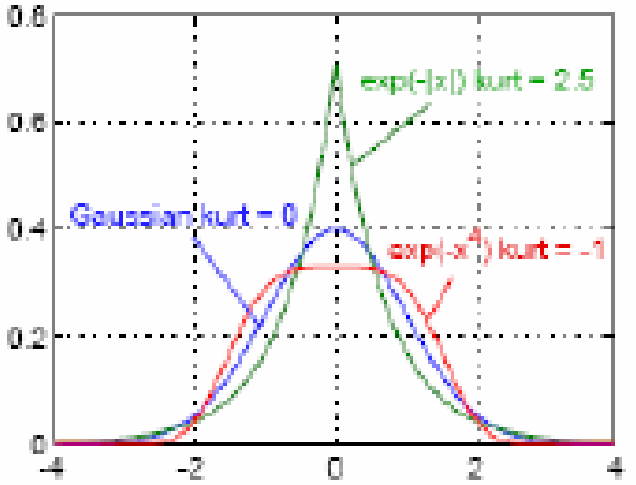
FIGURE 14.41. *Fifteen seconds of EEG data (of* 1917 *seconds) at nine (of* 100*) scalp channels (top panel), as well as nine ICA components (lower panel). While nearby electrodes record nearly identical mixtures of brain and non-brain activity, ICA components are temporally distinct. The colored scalps represent the ICA unmixing coefficients* $\hat{a}_j$ *as a heatmap, showing brain or scalp location of the source.*

Hastie

ICA - Divergencia de Gaussianidad

- Encontrar los mejores pesos para que los componentes de salida sean independientes.
- ¿Cómo medir la independencia?
- Combinación lineal de variables randómicas llevan a una distribución normal
- Se usa estadística de alto- orden para medir divergencia de la Gaussiana: Momento de cuarto orden, Kurtosis.

$$kurt(y) = E\left[\left(\frac{y-\mu}{\sigma}\right)^4\right] - 3$$



ICA- Maximizar entropía conjunta

$$H(\mathbf{y}) = -E\left[\ln\left(p_y(\mathbf{y})\right)\right] = E\left[\ln|\mathbf{J}|\right] - E\left[\ln\left(p_s(\mathbf{s})\right)\right]$$

$$p_y(\mathbf{y}) = \frac{p_s(\mathbf{s})}{|\mathbf{J}|} \qquad \mathbf{J} = \begin{bmatrix} \frac{\delta y_1}{\delta s_1} & \frac{\delta y_d}{\delta s_1} \\ \frac{\delta y_1}{\delta s_d} & \frac{\delta y_d}{\delta s_d} \end{bmatrix}$$

- Ventajas: No orotgonales, número variable, combinado con funciones no lineales.
- Contras: No importa el orden, sensible al ruido, no considera dimensión temporal