



MÉTODOS AVANZADOS EN APRENDIZAJE ARTIFICIAL:

TEORÍA Y APLICACIONES A PROBLEMAS DE PREDICCIÓN



Manuel Sánchez-Montañés

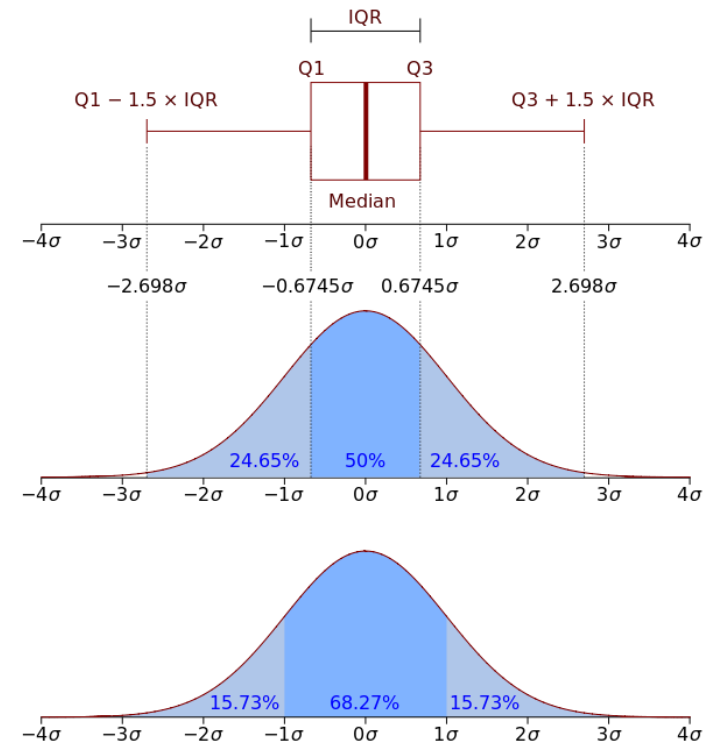
Luis Lago

Ana González

Escuela Politécnica Superior
Universidad Autónoma de Madrid

Preprocesamiento de los datos

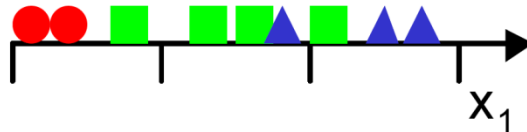
- Discretización/Segmentación de variables
- Tratamiento de fechas
- Creación de variables dicotómicas, caso particular binarias
- Gestión de missing values
- Detección de outliers
- Detección de Falso Predictores
- Balanceo de clases
- Normalización:
 - Tipificación de variables: $N(\mu, \sigma) \rightarrow N(0, 1)$
- Reducción de la base de datos



Reducción de dimensionalidad

El problema de la dimensionalidad (1)

- Se refiere a problemas asociados con el análisis de datos multivariable cuando la dimensionalidad (número de variables) es grande
- Consideremos un problema de clasificación en 3 clases:
 - Una aproximación sencilla sería:
 - Dividir el espacio de características en celdas uniformes
 - Por cada celda, calcular qué porcentaje de ejemplos de cada clase caen ahí
 - Dado un ejemplo nuevo, ver en qué celda cae, y asignarle la clase predominante ahí
 - En nuestro ejemplo de juguete decidimos empezar con un sólo atributo y dividir la recta en tres segmentos

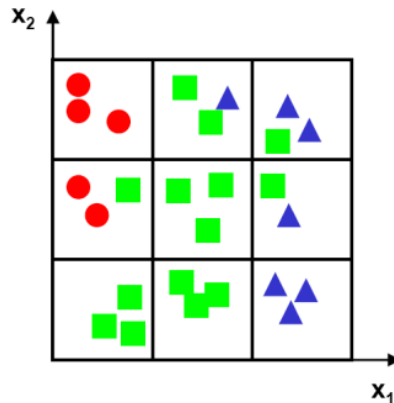


- Después de hacer esto, nos damos cuenta de que hay bastante solapamiento entre las clases, así que decidimos incluir un atributo nuevo para así intentar mejorar la separabilidad

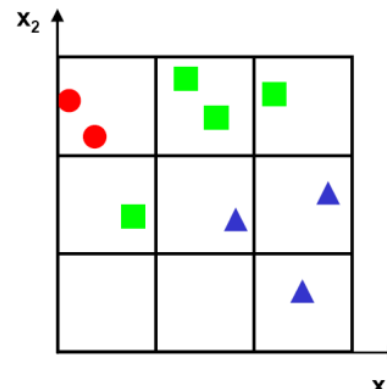
El problema de la dimensionalidad (2)

- Decidimos mantener la granularidad de cada eje, lo que aumenta el número de celdas de 3 (en 1D) a $3^2=9$ (en 2D)
 - Llegados a este punto necesitamos tomar una decisión:
¿ Mantenemos la densidad de ejemplos por celda, o mantenemos el número de ejemplos que teníamos en 1D ?
 - Si decidimos mantener la densidad, el número de ejemplos aumenta de 9 (en 1D) a 27 (en 2D)
 - Si decidimos mantener el número de ejemplos, su distribución en 2D está muy dispersada

Densidad constante



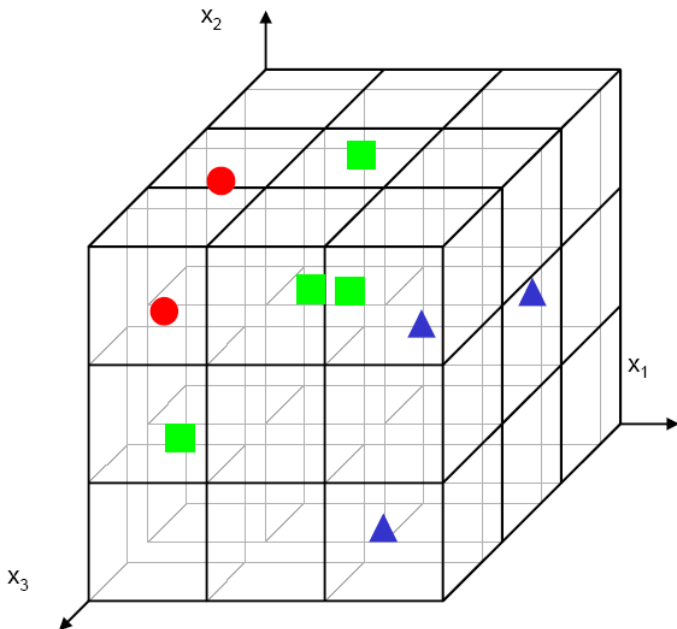
Número de ejemplos constante



El problema de la dimensionalidad (3)

- Si nos movemos a 3D, el problema empeora aún más:
 - El número de celdas aumenta a $3^3=27$
 - Si mantenemos la densidad de ejemplos, el número de ejemplos que necesitamos es 81
 - Si mantenemos el número de ejemplos, casi todas las celdas están vacías

Número de ejemplos constante

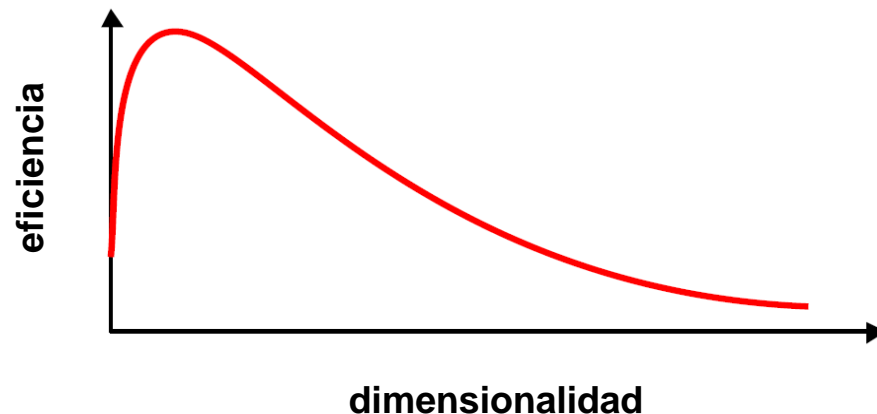


El crecimiento exponencial del número de datos necesarios para mantener la misma granulosidad en dimensiones grandes es lo que se conoce como la **maldición de la dimensionalidad**.

El problema de la dimensionalidad (4)

- En la práctica, el problema de la dimensionalidad implica que, dado un número de ejemplos fijo, hay **un número máximo de atributos** a partir del cual la eficiencia de nuestro clasificador se degrada en vez de aumentar
 - En muchos casos, la mayor calidad del clasificador con menos atributos compensa la información que perdemos descartando atributos.

Manteniendo fijo el número de observaciones



El problema de la dimensionalidad (5)

- Grado de crecimiento del problema de la dimensionalidad:
 - Crecimiento exponencial en el número de ejemplos necesarios para mantener una densidad dada:
 - Para una densidad de **N** ejemplos por celda en **D** dimensiones, el número de ejemplos es **N^D**
 - Para cajas de tamaño ϵ , el número de puntos es $\Theta(\epsilon^{-D})$
 - Crecimiento exponencial en la complejidad de la función objetivo (una estimación de la densidad) con mayor dimensionalidad

El problema de la dimensionalidad (6)

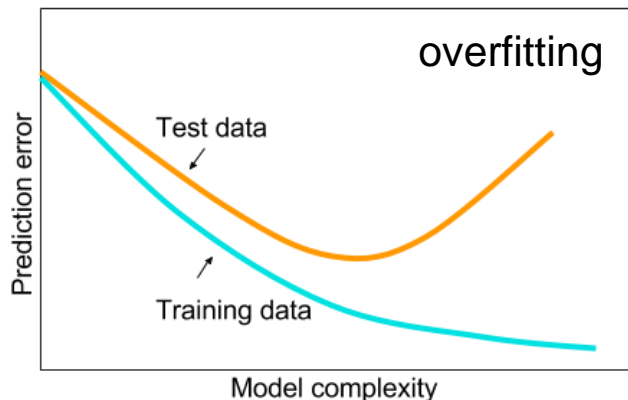
Resumiendo:

- Dentro del aprendizaje, cuánto **más compleja** (mayor cantidad de parámetros) sea nuestra función **más ejemplos** necesitaremos
- **Computacionalmente** el tiempo de entrenamiento de los algoritmos incrementa sustancialmente con el número de atributos.
- **Atributos ruidosos** o irrelevantes pueden tener el mismo peso que atributos relevantes contribuyendo negativamente en la exactitud (accuracy). → Tendencia a que “las cosas” parezcan más similares cuánto más atributos se tenga.

El problema de la dimensionalidad (7)

Resumiendo:

- Cuanto menos atributos más fácil de interpretar el modelo
 - “Una función definida en un espacio de muchas dimensiones es mucho más compleja que una función definida en un espacio de menos dimensiones, siendo estas complicaciones más difíciles de discernir” - Friedman
 - **Occam's Razor** → $f(X, \theta)$ Complejidad de un modelo depende de los atributos y de los parámetros del modelo.
 - Dado dos modelos con errores similares de generalización se debe elegir el menos complejo.
 - Modelos con alta complejidad aumentan la probabilidad de ajustar los errores de los datos.
- Incluso si sabemos (o suponemos) que las pdfs son gaussianas, para dimensiones muy altas seguimos teniendo problemas a no ser que asumamos una forma simplificada en las matrices de correlación

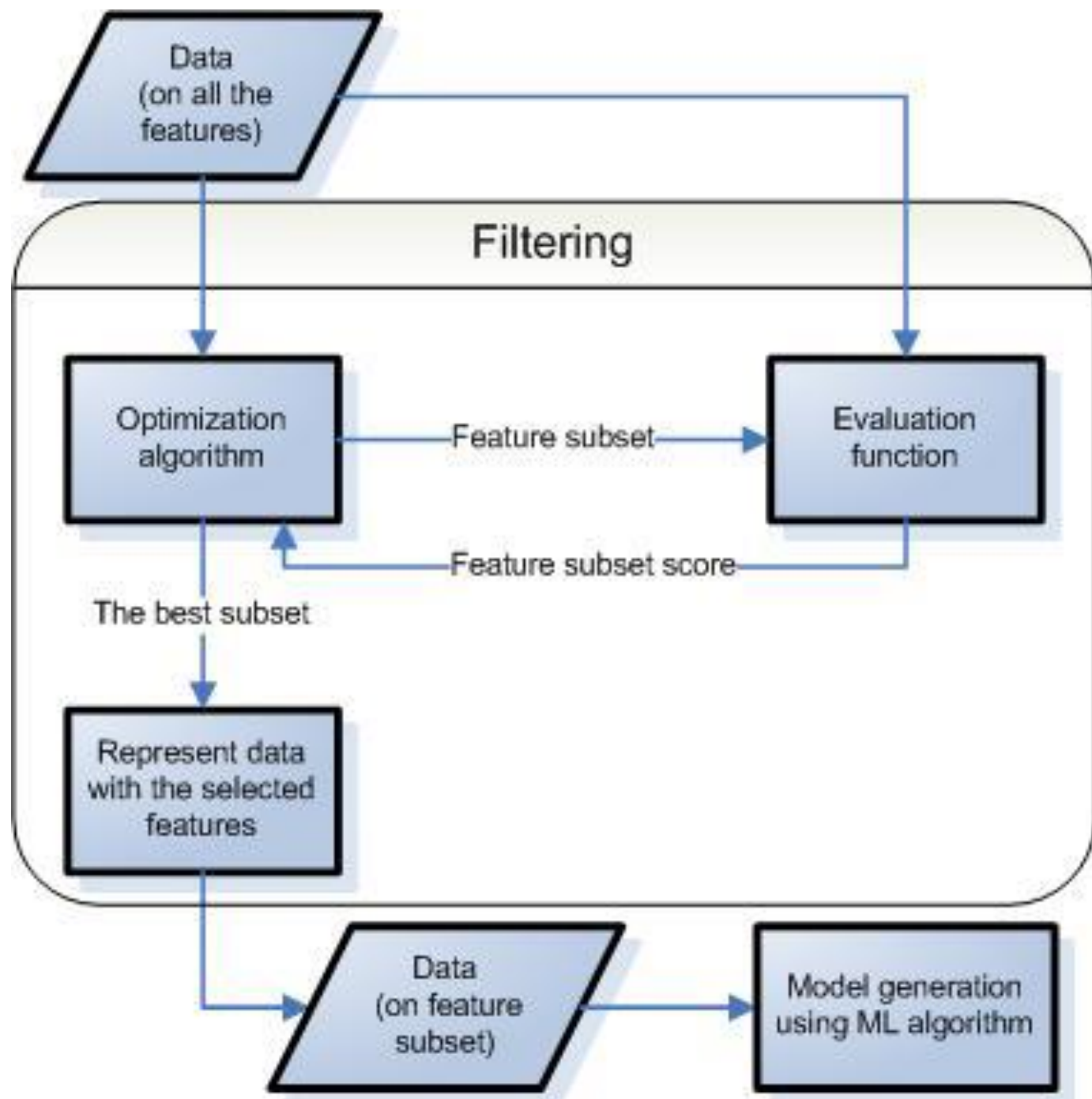


$$\begin{pmatrix} \sigma_1^2 & C_{12} & C_{13} & C_{14} \\ C_{21} & \sigma_2^2 & C_{23} & C_{24} \\ C_{31} & C_{32} & \sigma_3^2 & C_{34} \\ C_{41} & C_{42} & C_{43} & \sigma_4^2 \end{pmatrix}$$

Reducción de la dimensionalidad (8)

- Selección vs Extracción de características:
 - Selección: se selecciona un subconjunto de características a partir del conjunto original
 - Métodos de filtrado (filter methods):
 - Seleccionan el mejor conjunto de características en función de un criterio razonable.
 - El criterio es independiente del algoritmo de aprendizaje.
 - Ej: Información mutua con la clase, test múltiple hipótesis
 - Métodos envolventes (wrapper methods):
 - Selecciona el mejor conjunto de características de acuerdo al algoritmo de aprendizaje.
 - Ej: SVM-RFE (Guyon et al., 2000)
 - Extracción: las nuevas características proceden de una transformación de las originales.
 - Ej: transformación lineal $y = W^T x$, éste es el caso de LDA, PCA, ICA

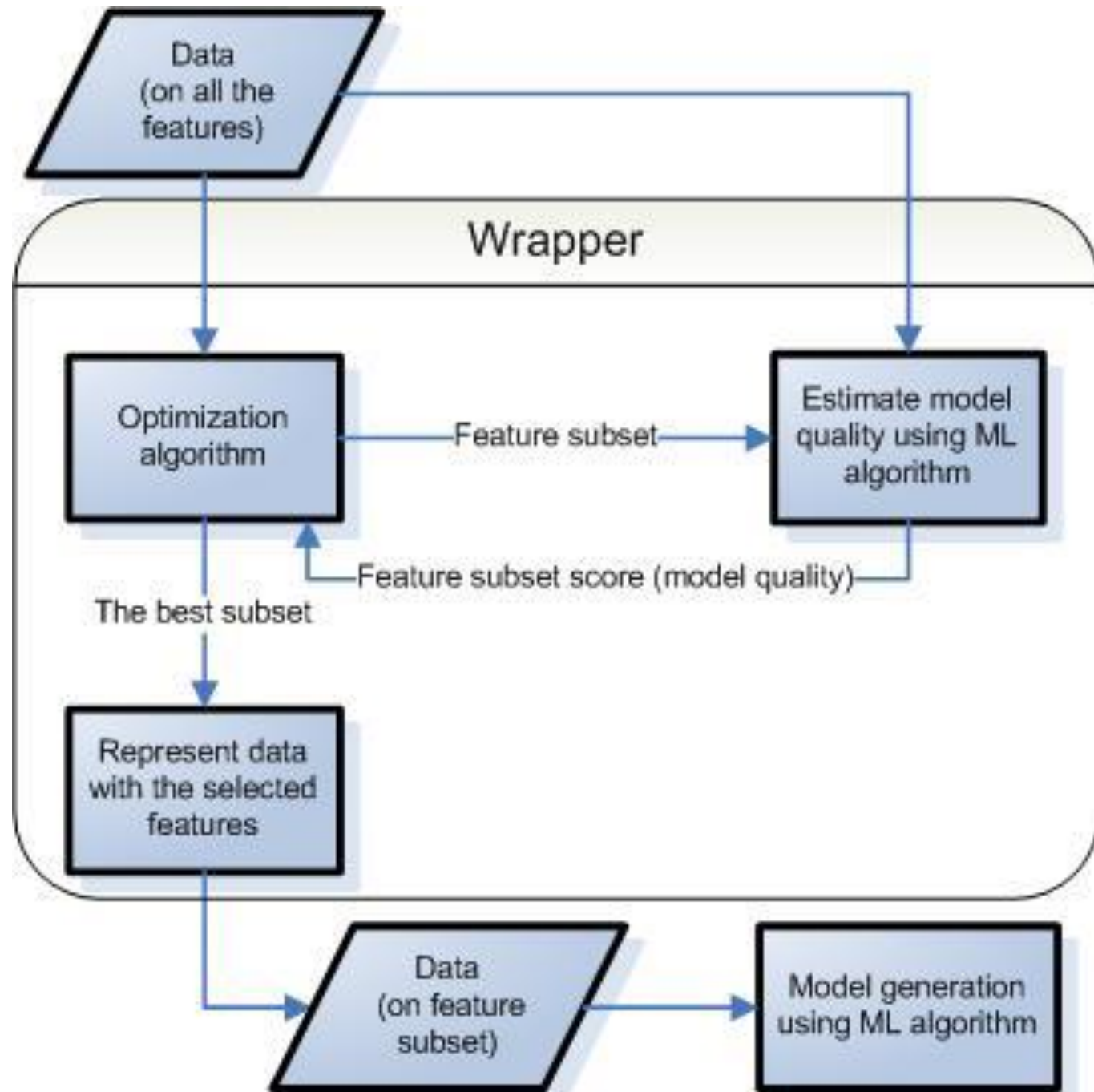
Evaluación es independiente del algoritmo de aprendizaje



Selección se basa en el algoritmo de aprendizaje.

Inconvenientes:

- Alta carga computacional
- Características seleccionadas dependen del algoritmo.



Reducción de la dimensionalidad (10)

- **Enfoques (2):**
 - **Aprendizaje supervisado:**
 - Test de múltiples hipótesis
 - Información mutua
 - LDA
 - **Aprendizaje no supervisado:**
 - PCA
 - ICA

Filtrado simple: Test múltiples hipótesis (1)

- Contraste de hipótesis: Hipótesis nula H_0 frente a la hipótesis alternativa H_1
- La hipótesis nula es elegida de tal forma que la probabilidad de cualquier resultado de un experimento puede ser calculado asumiendo que H_0 es cierta.
- Nunca se acepta H_0 :
 - se rechaza H_0 al nivel de significancia α
 - no se rechaza al nivel de significancia α
- Es preciso definir un buen test estadístico, T , y calcular los p-values

Filtrado Simple: Test múltiples hipótesis (2)

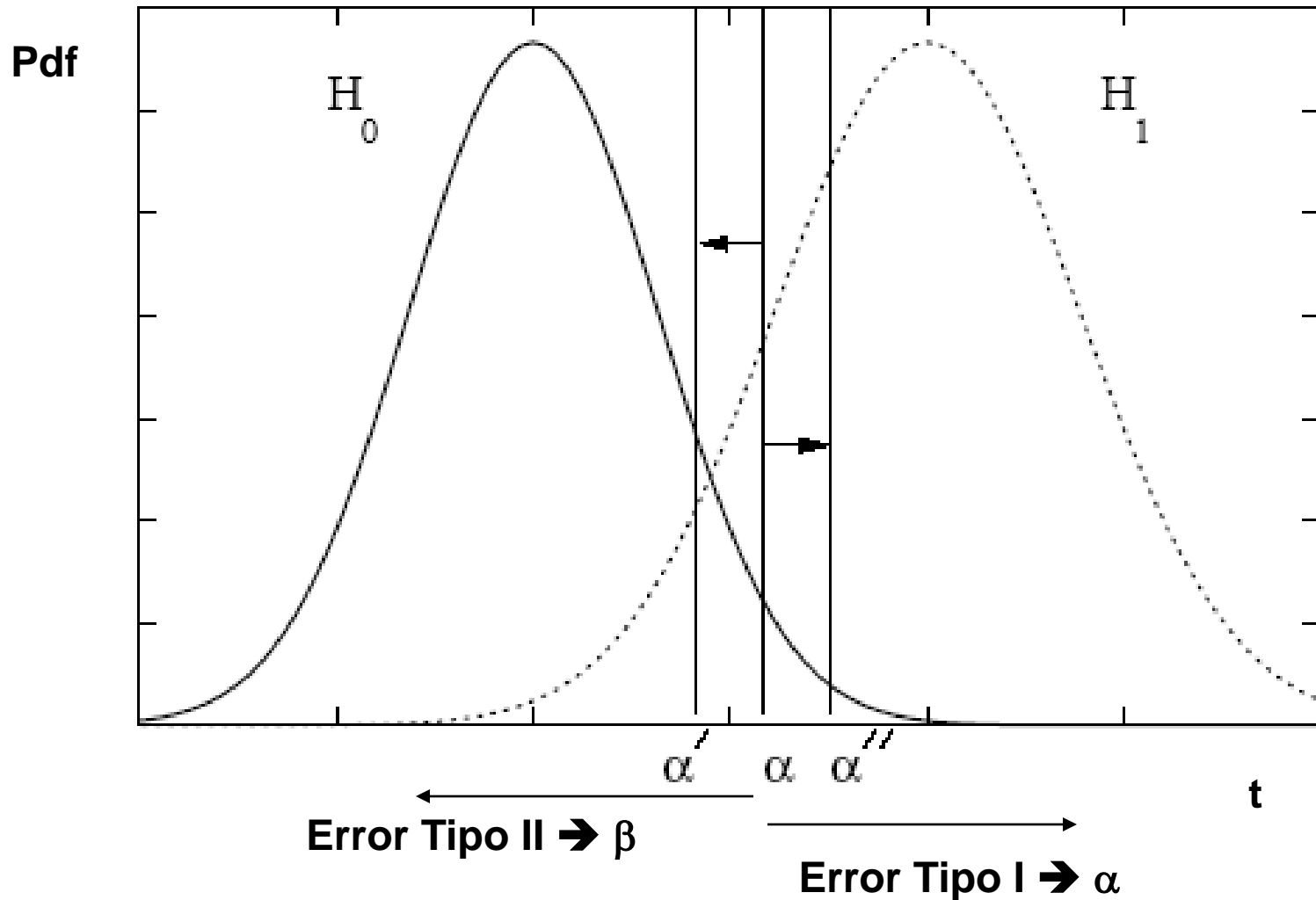
- Aplicándolo a la reducción de dimensionalidad en un problema con dos clases (ejemplo, microarray: tejidos con tumor y sin tumor)
- Objetivo: Selección de aquellos atributos (genes) que son estadísticamente diferenciados (significativos):
 - **Test estadístico** : t – student.
 - **Exigencias del test**: $p(\mathbf{x}|\omega_i) = \mathcal{N}(\mu_i, \Sigma)$
 - **Hipótesis nula**: En el atributo (gen) estudiado sigue la misma distribución en las dos clases $\Rightarrow H_0: \mu_1 = \mu_2$
 - **Hipótesis alternativa**: No siguen la misma distribución \Rightarrow
$$H_1: \mu_1 \neq \mu_2$$
 - Para cada atributo la hipótesis nula es contrastada contra la alternativa \rightarrow Filtrado simple, supone independencia de los atributos

Filtrado Simple: Test múltiples hipótesis (3)

- **Algoritmo de selección de atributos:**
 1. **Para cada atributo:**
 - **Calcular su estadístico \rightarrow valor de t_{calc}**
 - **Con el valor del estadístico calcular su p-value**
 $P(t > t_{\text{calc}})$
 2. **Seleccionar aquellos atributos cuyos p-values sean menores que el nivel de significancia $\alpha \Rightarrow$ atributos significativamente diferenciados**

Seleccionar atributos con $p_i \leq \alpha$

Filtrado Simple: Test múltiples hipótesis (4)



Filtrado Simple: Test múltiples hipótesis (5)

- Cálculo del estadístico:

μ_1 = Media clase 1

μ_2 = Media clase 2

Desviación muestral

$$s = \sqrt{\frac{(N_1 - 1)s_1^2 + (N_2 - 1)s_2^2}{(N_1 - 1) + (N_2 - 1)}}$$

$$t = \frac{\mu_1 - \mu_2}{s \sqrt{\frac{1}{N_1} + \frac{1}{N_2}}}$$

t-Student

$$s_1 = \sqrt{\frac{\sum_{i=1}^N (x_i^{\Omega_1} - \mu_1)^2}{N_1 - 1}}$$

$$s_2 = \sqrt{\frac{\sum_{i=1}^N (x_i^{\Omega_2} - \mu_2)^2}{N_2 - 1}}$$

**Suponemos
que las
desviaciones
standard de las
dos clases son
equivalentes**

Número de grados de libertad:

$$\nu = N_1 + N_2 - 2$$

Filtraado simple: Test múltiples hipótesis

(6)

Aproximación de Welch-Satterthwaite:

- Cálculo del estadístico (2):

μ_1 = Media clase 1

μ_2 = Media clase 2

$$s_1 = \sqrt{\frac{\sum_{i=1}^N (x_i^{\Omega_1} - \mu_1)^2}{N_1 - 1}}$$

$$s_2 = \sqrt{\frac{\sum_{i=1}^N (x_i^{\Omega_2} - \mu_2)^2}{N_2 - 1}}$$

$$t = \frac{\mu_1 - \mu_2}{\sqrt{\frac{s_1^2}{N_1} + \frac{s_2^2}{N_2}}}$$

t-Student

Aproximación de Welch-Satterthwaite: desviaciones standard de las dos clases NO son equivalentes (heterocedasticidad)

Número de grados de libertad:

$$\nu = \frac{\left(\frac{s_1^2}{N_1} + \frac{s_2^2}{N_2} \right)^2}{\frac{s_1^4}{N_1^2 (N_1 - 1)} + \frac{s_2^4}{N_2^2 (N_2 - 1)}}$$

Filtrado Simple: Test múltiples hipótesis (7)

- Cálculo de los p-values:
 - Tablas de la distribución de Student para el valor calculado del estadístico t y ν grados de libertad.
 - Cálculo de la densidad de probabilidad

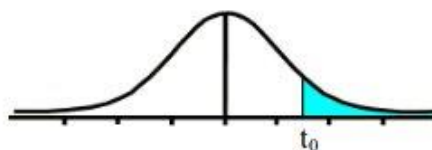
$$P(t | \nu) = \frac{1}{\nu^{1/2} B\left(\frac{1}{2}, \frac{\nu}{2}\right)} \int_{-t}^t \left(1 + \frac{x^2}{\nu}\right)^{-\frac{\nu+1}{2}} dx$$

$$\text{P-value} = 1 - P(t | \nu)$$

Filtrado Simple: Test múltiples hipótesis

(8)

Tabla t-Student



Grados de libertad	0.25	0.1	0.05	0.025	0.01	0.005
1	1.0000	3.0777	6.3137	12.7062	31.8210	63.6559
2	0.8165	1.8856	2.9200	4.3027	6.9645	9.9250
3	0.7649	1.6377	2.3534	3.1824	4.5407	5.8408
4	0.7407	1.5332	2.1318	2.7765	3.7469	4.6041
5	0.7267	1.4759	2.0150	2.5706	3.3649	4.0321
6	0.7176	1.4398	1.9432	2.4469	3.1427	3.7074
7	0.7111	1.4149	1.8946	2.3646	2.9979	3.4995
8	0.7064	1.3968	1.8595	2.3060	2.8965	3.3554
9	0.7027	1.3830	1.8331	2.2622	2.8214	3.2498
10	0.6998	1.3722	1.8125	2.2281	2.7638	3.1693
11	0.6974	1.3634	1.7959	2.2010	2.7181	3.1058
12	0.6955	1.3562	1.7823	2.1788	2.6810	3.0545
13	0.6938	1.3502	1.7709	2.1604	2.6503	3.0123
14	0.6924	1.3450	1.7613	2.1448	2.6245	2.9768
15	0.6912	1.3406	1.7531	2.1315	2.6025	2.9467
16	0.6901	1.3368	1.7459	2.1199	2.5835	2.9208
17	0.6892	1.3334	1.7396	2.1098	2.5669	2.8982
18	0.6884	1.3304	1.7341	2.1009	2.5524	2.8784
19	0.6876	1.3277	1.7291	2.0930	2.5395	2.8609
20	0.6870	1.3253	1.7247	2.0860	2.5280	2.8453
21	0.6864	1.3232	1.7207	2.0796	2.5176	2.8314
22	0.6858	1.3212	1.7171	2.0739	2.5083	2.8188
23	0.6853	1.3195	1.7139	2.0687	2.4999	2.8073
24	0.6848	1.3178	1.7109	2.0639	2.4922	2.7970
25	0.6844	1.3163	1.7081	2.0595	2.4851	2.7874

Filtrado Simple: Test múltiples hipótesis (9)

- Estudio sobre un único atributo:

N	Clase 1	Clase 2
1	32	36
2	37	31
3	35	30
4	28	31
5	41	34
6	44	36
7	35	29
8	31	32
9	34	31
10	38	
11	42	

	Clase 1	Clase 2
•Medias	36.0909	32.2222
•Desviación Standard	4.9082	2.5386
•No. Patrones por clase	11	9
•Grados libertad por clase	10	8

•Aproximación **Welch-Satterthwaite**:

$$t = 2.2694, \quad v = 15.5$$

Tabla para $\alpha=0.05 \rightarrow t_{\text{critico}} = 1.746$

$t > t_{\text{critico}} \rightarrow$ Rechazar la hipótesis nula

Atributo **estadísticamente significativo**

Filtrado simple: Información mutua (1)

- Información mutua respecto a la clase nos da una idea de cómo de independiente es un atributo respecto a la clase:
 - Mayor información mutua mayor dependencia con la clase

- Entropía de una variable aleatoria x : $H[x] = -\int p(x) \log p(x) dx$

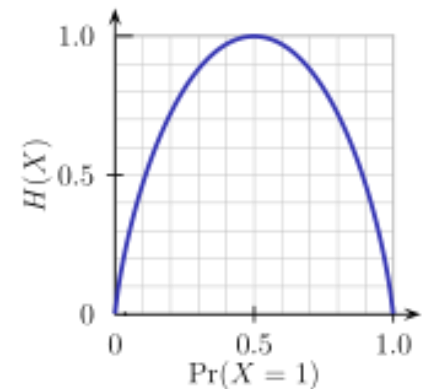
Medida de la incertidumbre, a mayor entropía mayor incertidumbre

- Información mutua

$$MI[x, \omega] = H[x] - H[x | \omega] \geq 0$$

$$H[x | \omega] = \sum_{c=1}^C p(\omega_c) H[x | \omega_c]$$

$$H[x | \omega_c] = -\int p(x | \omega_c) \log p(x | \omega_c) dx$$



Bernoulli, dos posibles estados $\{0,1\}$.
Cuando $P(X=1)=0.5$, todos los resultados posibles son igualmente probables, por lo que el resultado es poco predecible y la entropía es máxima.

Simple Filtering: Mutual Information

Two attributes (age, height), two classes

class	1	1	1	1	2	2	2	2	2	2
age	32	28	36	34	26	30	24	26	22	20
height	180	170	160	175	182	168	170	180	174	172

First step: recode the continuous attributes / attributes with many values

class	1	1	1	1	2	2	2	2	2	2
age'	a3	a2	a4	a3	a2	a3	a1	a2	a1	a1
height'	h3	h2	h1	h2	h3	h1	h2	h3	h2	h2

age' {
a1: $(-\infty, 25)$
a2: $[25, 30)$
a3: $[30, 35)$
a4: $[35, \infty)$

height' {
h1: $[160, 170)$
h2: $[170, 180)$
h3: $[180, 190)$

Simple Filtering: Mutual Information

Second step

class	1	1	1	1	2	2	2	2	2	2
age'	a3	a2	a4	a3	a2	a3	a1	a2	a1	a1
height'	h3	h2	h1	h2	h3	h1	h2	h3	h2	h2

$$\begin{aligned}
 \text{MI}[\text{age}', \text{class}] &= p(a1,1) \cdot \log \frac{p(a1,1)}{p(a1) \cdot p(1)} + p(a1,2) \cdot \log \frac{p(a1,2)}{p(a1) \cdot p(2)} + \\
 &\quad p(a2,1) \cdot \log \frac{p(a2,1)}{p(a2) \cdot p(1)} + p(a2,2) \cdot \log \frac{p(a2,2)}{p(a2) \cdot p(2)} + \\
 &\quad p(a3,1) \cdot \log \frac{p(a3,1)}{p(a3) \cdot p(1)} + p(a3,2) \cdot \log \frac{p(a3,2)}{p(a3) \cdot p(2)} + \\
 &\quad p(a4,1) \cdot \log \frac{p(a4,1)}{p(a4) \cdot p(1)} + p(a4,2) \cdot \log \frac{p(a4,2)}{p(a4) \cdot p(2)} = \\
 &= 0 + \frac{3}{10} \cdot \log \frac{3/10}{3/10 \cdot 6/10} + \\
 &\quad \frac{1}{10} \cdot \log \frac{1/10}{3/10 \cdot 4/10} + \frac{2}{10} \cdot \log \frac{2/10}{3/10 \cdot 6/10} + \\
 &\quad \frac{2}{10} \cdot \log \frac{2/10}{3/10 \cdot 4/10} + \frac{1}{10} \cdot \log \frac{1/10}{3/10 \cdot 6/10} + \\
 &\quad \frac{1}{10} \cdot \log \frac{1/10}{1/10 \cdot 4/10} + 0 = 0.3827 \text{ nats}
 \end{aligned}$$

$$\text{MI}[x, y] \equiv \sum_{i=1}^{N_x} \sum_{j=1}^{N_y} p(x_i, y_j) \cdot \log \frac{p(x_i, y_j)}{p(x_i) \cdot p(y_j)}$$

Información mutua (2)

- **Selección de atributos:**
 1. Para cada atributo calcular su información mutua, $MI[x, \omega]$
 2. Atributos con el valor de información mutua alto corresponden a los atributos más dependientes de la clase → Estos son los que interesan. Cuidado con los falsos predictores!!
- **¿Cómo hallo las funciones de densidad $p(x)$?**
 - Técnicas de estimación de densidades (pdf):
 - Paramétricas: Gaussianas, Combinación de Gaussianas, otras funciones de densidad
 - No Paramétricas: Histogramas, vecinos próximos, kernels ...

Inconvenientes métodos filtrado simple

- Combinando variables que individualmente son buenas, no siempre conduce a buenos resultados en problemas de clasificación/clustering.
- Dos o más características fuertemente correlacionadas pueden tener un alto valor en el ranking (información mutua):
 - Sólo una debería seleccionarse
 - Si correlaciones lineal → Método de Pearson $C_{ik} = \rho_{ik} \sigma_i \sigma_k$

Soluciones Propuestas:

- mRMR: mide la relevancia como la información mutua con la clase y la redundancia como la información mutua entre las variables: Peng, H.C., Long, F., and Ding, C., "Feature selection based on mutual information: criteria of max-dependency, max-relevance, and min-redundancy," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 27, No. 8, pp.1226-1238, 2005.
- CFS: Correlation Feature Selection

Reducción de la dimensionalidad con transformaciones de los atributos

- Idea: encontrar una transformación $y=f(x)$ que conserve la información acerca del problema, minimizando el número de componentes
- En general, la función óptima $y=f(x)$ será no lineal
- Sin embargo, no hay una forma de generar sistemáticamente transformaciones no lineales:
 - La selección de un subconjunto particular de transformaciones depende del problema
 - Por esta razón, la limitación a transformaciones lineales ha sido ampliamente aceptada, $y = W^T x \rightarrow$ **y es una proyección lineal de x**

Reducción de la dimensionalidad con transformaciones de los atributos (2)

$$\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{bmatrix} \xrightarrow{\text{Transformación lineal}} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_M \end{bmatrix} = \begin{bmatrix} w_{11} & w_{12} & \cdots \\ w_{21} & w_{22} & \cdots \\ \vdots & \vdots & \ddots \\ w_{M1} & w_{M2} & \end{bmatrix} \begin{bmatrix} w_{1N} \\ w_{2N} \\ \vdots \\ w_{MN} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{bmatrix}$$

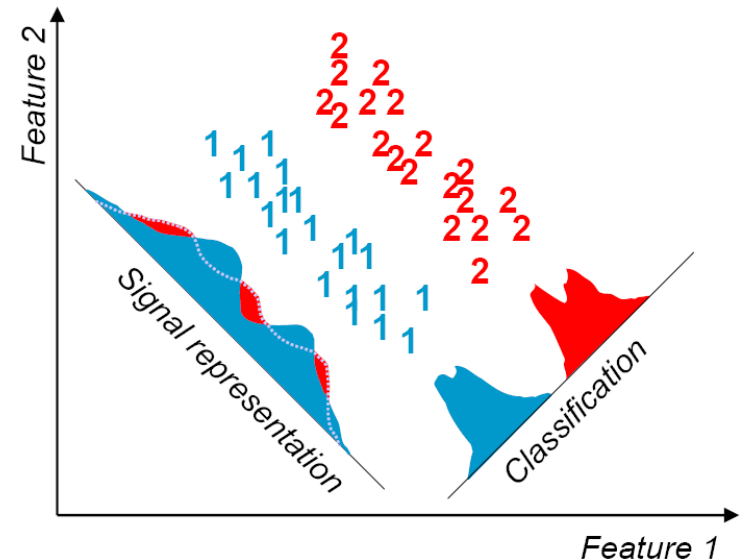
M-dimensional
 $M < N$

N-dimensional

- Por el momento nos centraremos en transformaciones lineales, y volveremos a ver transformaciones no lineales cuando hablemos de los perceptrones multicapa.

Representación de la señal *versus* clasificación (PCA vs. LDA)

- La selección de la transformación extractora de características, $y=f(x)$, está guiada por una función objetivo que buscamos maximizar (o minimizar)
- Dependiendo del criterio usado por la función objetivo, las técnicas de extracción de características se dividen en dos categorías:
 - **Representación de la señal:** El objetivo de la transformación extractora de características es representar los vectores de atributos de manera precisa en un espacio de menos dimensiones
 - **Clasificación:** El objetivo de la transformación extractora de características es resaltar en un espacio de menos dimensiones la información discriminante de clases
- Hay dos técnicas principales en la extracción lineal de características:
 - **Análisis de Componentes Principales (PCA)**, que usa el criterio de representación de la señal
 - **Análisis Discriminante Lineal (LDA)**, que utiliza el criterio de clasificación



Análisis Discriminante Lineal (LDA)

- **Análisis Discriminante Lineal, dos clases**
- **Análisis Discriminante Lineal, C clases**
- **Limitaciones de LDA**
- **Variantes de LDA**
- **Otros métodos de reducción de la dimensionalidad basados en LDA**

Análisis Discriminante Lineal, dos clases (1)

- El objetivo de **LDA** es realizar una **reducción de la dimensionalidad**, preservando el máximo posible de **información discriminatoria**.

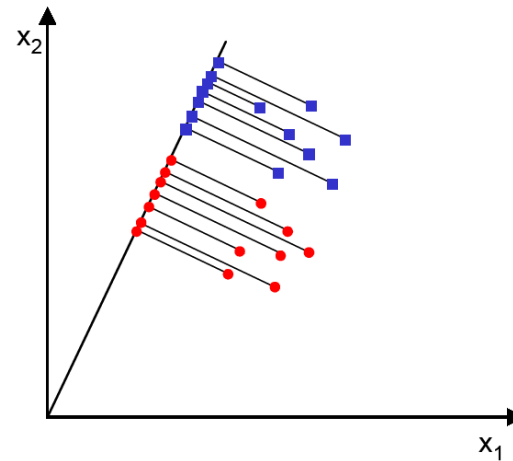
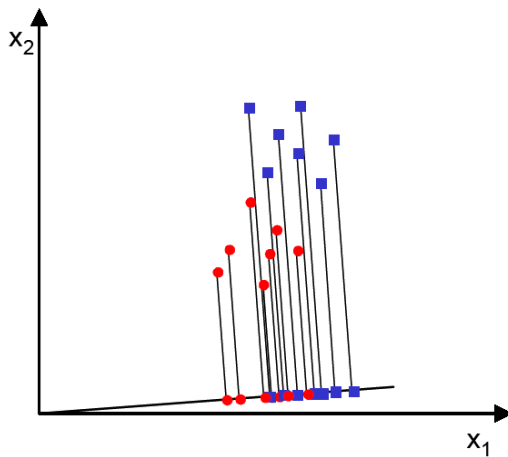
Tenemos un conjunto de vectores en **D** dimensiones $\{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N\}$, donde **N**₁ son de clase ω_1 , y **N**₂ de clase ω_2

Buscamos obtener un escalar **y** proyectando los vectores **x** en una línea:

$$\mathbf{y} = \mathbf{w}^T \mathbf{x}$$

Análisis Discriminante Lineal, dos clases (2)

- De todas las posibles líneas, nos gustaría seleccionar la que maximiza la separabilidad de los escalares $y = w^T x$
- Ilustramos a continuación esta idea para el caso de vectores x con 2 dimensiones:



Análisis Discriminante Lineal, dos clases (3)

- Para poder encontrar un buen vector de proyección, necesitaremos definir una medida de separación entre las proyecciones

- El vector promedio de cada clase en los espacios \mathbf{x} e \mathbf{y} es:

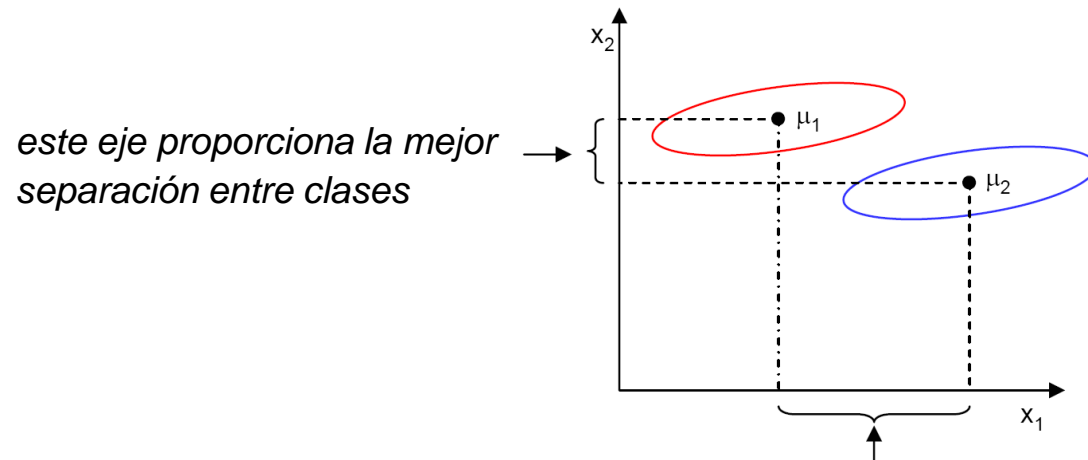
$$\mu_i = \frac{1}{N_i} \sum_{\mathbf{x} \in \omega_i} \mathbf{x} \quad \text{Espacio Original} \quad \mathbf{y} \quad \tilde{\mu}_i = \frac{1}{N_i} \sum_{\mathbf{y} \in \omega_i} \mathbf{y} = \frac{1}{N_i} \sum_{\mathbf{x} \in \omega_i} \mathbf{w}^T \mathbf{x} = \mathbf{w}^T \mu_i \quad \text{Espacio de la Proyección}$$

- Podríamos entonces elegir nuestra función objetivo como la distancia entre los promedios proyectados: clasificación por la distancia a las medias

$$J(\mathbf{w}) = \left| \tilde{\mu}_1 - \tilde{\mu}_2 \right| = \left| \mathbf{w}^T (\mu_1 - \mu_2) \right|$$

Análisis Discriminante Lineal, dos clases (4)

- Sin embargo, la distancia entre los promedios proyectados no es una buena medida ya que no tiene en cuenta la desviación standard dentro de las clases.



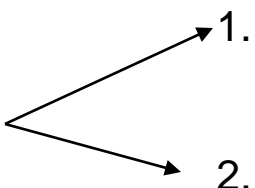
este eje maximiza la distancia entre medias

Análisis Discriminante Lineal, dos clases (5)

- La solución propuesta por Fisher es maximizar una función que representa **la diferencia entre las medias, normalizada por una medida de la dispersión dentro de las clases**
 - Por cada clase definimos la dispersión, un equivalente a la varianza, como:

$$\tilde{s}_i^2 = \sum_{y \in \omega_i} (y - \tilde{\mu}_i)^2$$

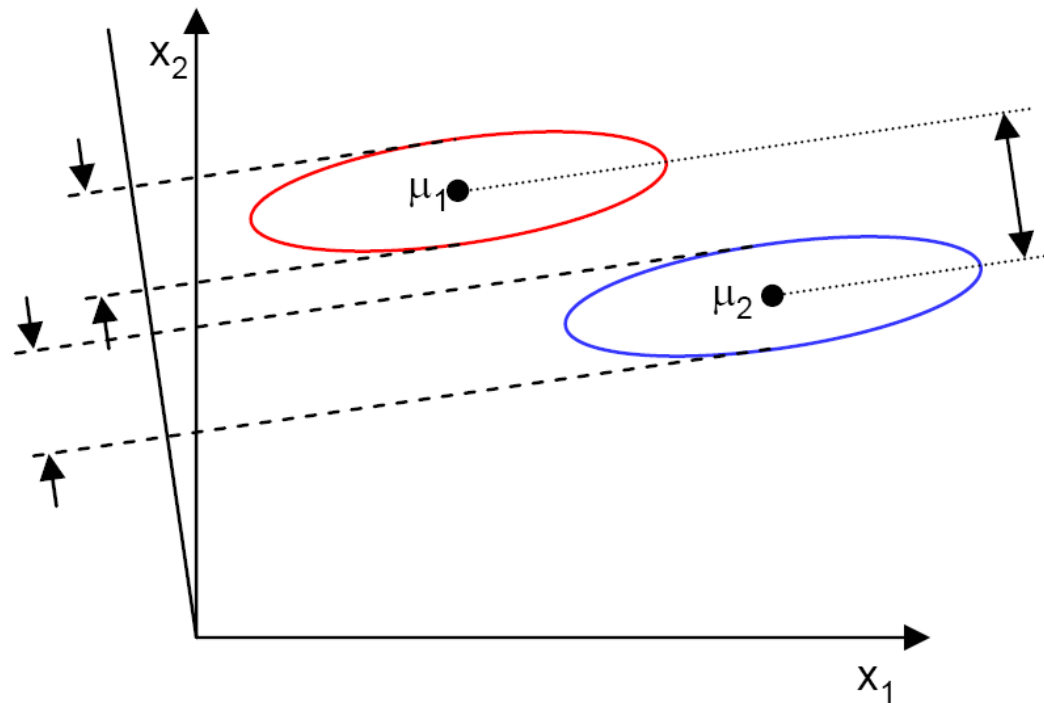
- donde la cantidad $(\tilde{s}_1^2 + \tilde{s}_2^2)$ es la **dispersión intra clase** de los ejemplos proyectados considerando igualdad en la probabilidad a priori de las clases $\rightarrow \tilde{S}_w$
- El discriminante lineal de Fisher se define como la función lineal $\mathbf{w}^T \mathbf{x}$ que maximiza la función objetivo:

$$J(\mathbf{w}) = \frac{|\tilde{\mu}_1 - \tilde{\mu}_2|^2}{\tilde{s}_1^2 + \tilde{s}_2^2}$$


1. Diferencia entre medias proyectadas aumenta
2. Dispersión intra-clases en la proyección disminuye

Análisis Discriminante Lineal, dos clases (6)

- De esta forma, estaremos buscando una proyección donde los ejemplos de la misma clase son proyectados muy cerca unos de otros (mínima dispersión), y al mismo tiempo, las medias proyectadas están lo más lejos posible.



Análisis Discriminante Lineal, dos clases (7)

- Para poder encontrar la proyección óptima \mathbf{w}^* , necesitaremos expresar $\mathbf{J}(\mathbf{w})$ como una función explícita de \mathbf{w}

- Primero definiremos las matrices de dispersión en el espacio original:

$$S_i = E[(x - \mu_i)(x - \mu_i)^T \mid x \in \omega_i]$$

$$S_w = \pi_1 S_1 + \pi_2 S_2$$

- donde \mathbf{S}_w es la llamada “matriz de dispersión intra clase”.

- La dispersión de la proyección \mathbf{y} se puede expresar en función de la matriz de dispersión en el espacio original \mathbf{x} :

$$\begin{aligned}\tilde{S}_i &= E[(y - \tilde{\mu}_i)(y - \tilde{\mu}_i)^T \mid y \in \omega_i] = E[(w^T x - w^T \mu_i)(w^T x - w^T \mu_i)^T \mid x \in \omega_i] = \\ &= E[w^T (x - \mu_i)(x - \mu_i)^T w \mid x \in \omega_i] = w^T S_i w\end{aligned}$$

$$\tilde{S}_w = \pi_1 \tilde{S}_1 + \pi_2 \tilde{S}_2 = \pi_1 w^T S_1 w + \pi_2 w^T S_2 w = w^T (\pi_1 S_1 + \pi_2 S_2) w = w^T S_w w$$

Análisis Discriminante Lineal, dos clases (8)

- De manera similar, podemos expresar la diferencia entre los promedios proyectados en función de las medias en el espacio original \mathbf{x} :

$$(\tilde{\mu}_1 - \tilde{\mu}_2)^2 = (w^T \mu_1 - w^T \mu_2)^2 = w^T \underbrace{(\mu_1 - \mu_2)(\mu_1 - \mu_2)^T}_{S_B} w = w^T S_B w$$

- S_B es la “matriz de dispersión interclase”.

Como es el producto externo de un vector consigo mismo, tiene rango ≤ 1

Análisis Discriminante Lineal, dos clases (9)

- Con lo que hemos visto, podemos expresar $\mathbf{J}(\mathbf{w})$ como una función explícita de \mathbf{w} :

$$\mathbf{J}(\mathbf{w}) = \frac{|\tilde{\mu}_1 - \tilde{\mu}_2|^2}{\tilde{\mathbf{S}}_1^2 + \tilde{\mathbf{S}}_2^2} \longrightarrow \boxed{\mathbf{J}(\mathbf{w}) = \frac{\mathbf{w}^T \mathbf{S}_B \mathbf{w}}{\mathbf{w}^T \mathbf{S}_W \mathbf{w}}}$$

- Maximizar $\mathbf{J}(\mathbf{w})$ respecto a \mathbf{w} tiene una solución analítica sencilla:

$$\mathbf{w}^* = \underset{\mathbf{w}}{\operatorname{argmax}} \left\{ \frac{\mathbf{w}^T \mathbf{S}_B \mathbf{w}}{\mathbf{w}^T \mathbf{S}_W \mathbf{w}} \right\} = \mathbf{S}_W^{-1} (\mu_1 - \mu_2)$$

- donde el módulo de \mathbf{w}^* es indiferente

Esta solución es el famoso **Discriminante Lineal de Fisher** (1936), aunque en realidad no es un discriminante sino **la elección de una dirección específica para la proyección de los datos a una dimensión**

Ejemplo de LDA

- Calcular la proyección LDA para el siguiente conjunto de datos en dos dimensiones:

$$X_1 = (x_1, x_2) = \{(4, 1), (2, 4), (2, 3), (3, 6), (4, 4)\}$$

$$X_2 = (x_1, x_2) = \{(9, 10), (6, 8), (9, 5), (8, 7), (10, 8)\}$$

- Solución** (a mano):

- Las estadísticas de las clases son:

$$S_1 = \begin{bmatrix} 0.80 & -0.40 \\ -0.40 & 2.60 \end{bmatrix}; S_2 = \begin{bmatrix} 1.84 & -0.04 \\ -0.04 & 2.64 \end{bmatrix}$$

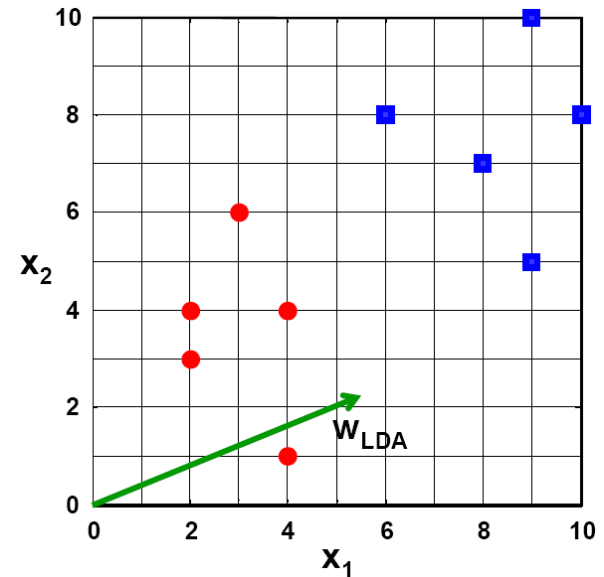
$$\mu_1 = [3.00 \quad 3.60]; \quad \mu_2 = [8.40 \quad 7.60]$$

- Las matrices de dispersión inter- e intra-clase son:

$$S_B = \begin{pmatrix} 25.2 & 22.8 \\ 30.24 & 27.36 \end{pmatrix} \quad S_W = \begin{pmatrix} 1.32 & -0.22 \\ -0.22 & 2.64 \end{pmatrix}$$

- La proyección óptima viene entonces dada por:

$$w^* = S_W^{-1}(\mu_1 - \mu_2) = [-0.91 \quad -0.39]^T$$



$$S_i = E[(x - \mu_i)(x - \mu_i)^T \mid x \in \omega_i]$$

$$S_W = \pi_1 S_1 + \pi_2 S_2$$

$$S_B = (\mu_1 - \mu_2)(\mu_1 - \mu_2)^T$$

Análisis Discriminante Lineal, C clases (1)

- El Discriminante de Fisher se puede generalizar a problemas con **C** clases (arbitrario)
- En vez de buscar una proyección **y** (escalar), buscamos **(C-1)** proyecciones **[y₁, y₂, ..., y_{C-1}]** por medio de **(C-1)** vectores de proyección **w_i**.
- Definimos por conveniencia la matriz de proyección **W** con **(C-1)** columnas:

$$\mathbf{W} = [\mathbf{w}_1 \mid \mathbf{w}_2 \mid \dots \mid \mathbf{w}_{C-1}]$$

$$y_i = \mathbf{w}_i^T \mathbf{x} \Rightarrow \mathbf{y} = \mathbf{W}^T \mathbf{x}$$

Análisis Discriminante Lineal, C clases (2)

Espacio Original

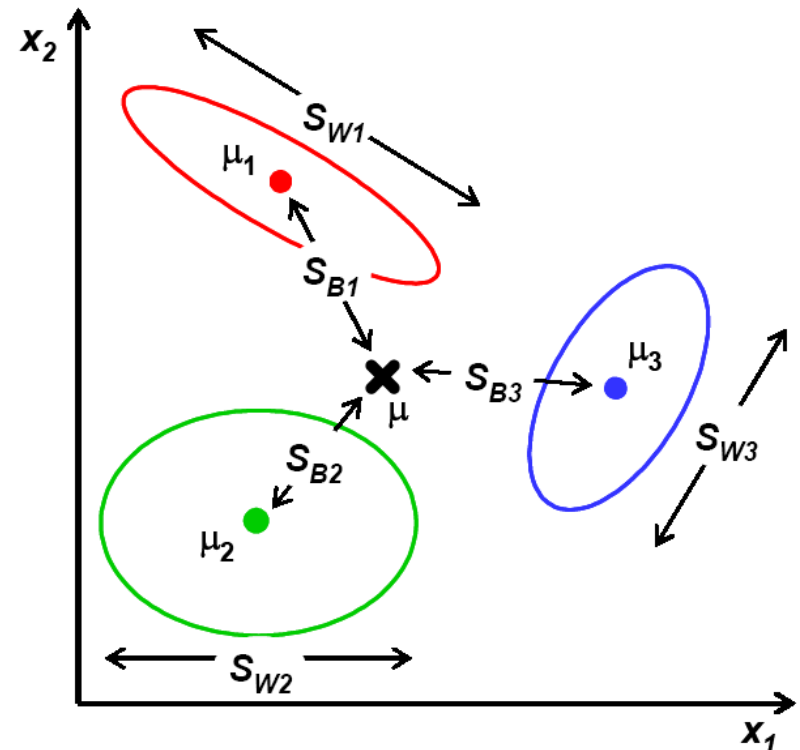
- Matriz de dispersión intra-clase

$$S_W = \sum_{i=1}^C \pi_i S_i \quad S_i = E[(x - \mu_i)(x - \mu_i)^T \mid x \in \omega_i] \quad \mu_i = \frac{1}{N_i} \sum_{x \in \omega_i} x$$

- Matriz de dispersión inter-clase

$$S_B = \sum_{i=1}^C \pi_i (\mu_i - \mu)(\mu_i - \mu)^T$$

$$\mu = \frac{1}{N} \sum_{\forall x} x = \frac{1}{N} \sum_{x \in \omega_i} N_i \mu_i$$



Análisis Discriminante Lineal, C clases (3)

- De manera similar, definimos el vector promedio y las matrices de dispersión de los ejemplos **proyectados** como:

$$\tilde{\mu}_i = E[y \mid y \in \omega_i]$$

$$\tilde{\mu} = E[y] = \sum_{i=1}^C \pi_i \tilde{\mu}_i$$

$$\tilde{S}_W = \sum_{i=1}^C \pi_i \tilde{S}_i$$

$$\tilde{S}_B = \sum_{i=1}^C \pi_i (\tilde{\mu}_i - \tilde{\mu})(\tilde{\mu}_i - \tilde{\mu})^T$$

- De manera análoga a cuando teníamos 2 clases, podemos escribir:

$$\tilde{S}_W = W^T S_W W$$

$$\tilde{S}_B = W^T S_B W$$

Análisis Discriminante Lineal, C clases (4)

- Estamos buscando una proyección que maximice la dispersión inter-clase y minimice la dispersión intra-clase.
- Ya que ahora la proyección no es un escalar (tiene C-1 dimensiones), usamos el determinante de las matrices de dispersión para obtener escalares:

$$J(W) = \frac{|\tilde{S}_B|}{|\tilde{S}_W|} = \frac{|W^T S_B W|}{|W^T S_W W|}$$

Función criterio de Fisher es una función escalar que es grande cuando:

- S_B es grande
- S_W es pequeña

- De esta forma, buscamos la matriz de proyecciones W^* que maximiza $J(W)$.

Análisis Discriminante Lineal, C clases (5)

- Se puede demostrar analíticamente que la matriz óptima \mathbf{W}^* es la que en sus columnas contiene los **(C-1)** autovectores de la matriz $\mathbf{S}_W^{-1} \mathbf{S}_B$ correspondientes a los **(C-1)** autovalores más grandes:

$$\mathbf{W}^* = [\mathbf{w}_1^* | \mathbf{w}_2^* | \cdots | \mathbf{w}_{C-1}^*] = \operatorname{argmax} \left\{ \frac{|\mathbf{W}^T \mathbf{S}_B \mathbf{W}|}{|\mathbf{W}^T \mathbf{S}_W \mathbf{W}|} \right\} \Rightarrow (\mathbf{S}_B - \lambda_i \mathbf{S}_W) \mathbf{w}_i^* = 0$$

$$(\mathbf{S}_W^{-1} \mathbf{S}_B) \mathbf{W}^* = \lambda \mathbf{W}^*$$

Análisis Discriminante Lineal, C clases (6)

- ¿Por qué **(C-1)**?

- \mathbf{S}_B es la suma de **C** matrices de orden 1 o menos

$$S_B = \sum_{i=1}^C \pi_i (\mu_i - \mu)(\mu_i - \mu)^T$$

y los vectores media están restringidos por: $\frac{1}{C} \sum_{i=1}^C \mu_i = \mu$

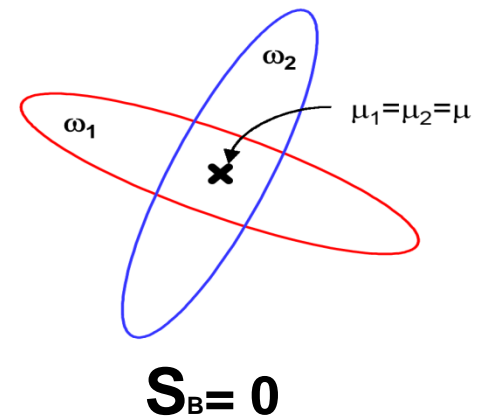
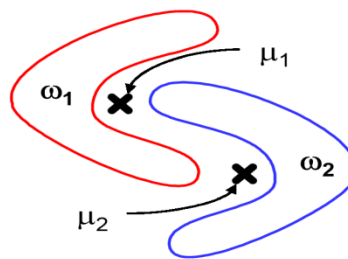
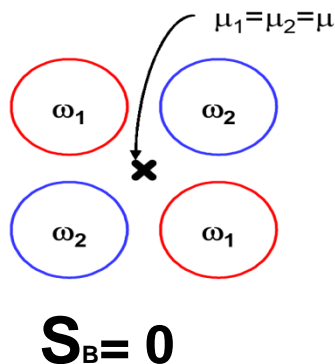
- De esta forma, \mathbf{S}_B es de rango menor o igual que **(C-1)**

- Esto significa que hay como mucho **(C-1)** autovalores λ_i que no son cero

- LDA se puede también derivar del método de Máxima Verosimilitud para el caso en el que las densidades condicionadas a la clase son gaussianas con las mismas matrices de covarianza.

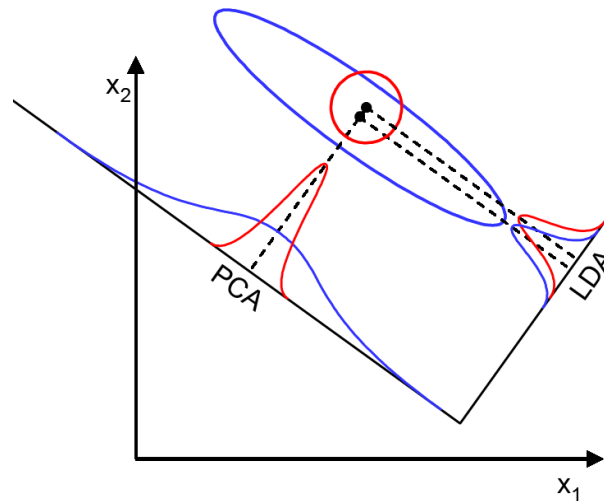
Limitaciones de LDA

- LDA produce como mucho **C - 1** características proyectadas
 - Si el error de clasificación estimado es demasiado alto, necesitaremos más características, con lo que deberemos utilizar otro método que proporcione esas características adicionales.
- LDA es un **método paramétrico** ya que asume implícitamente distribuciones **unimodales gaussianas**.
 - Si las distribuciones distan de ser gaussianas, las proyecciones LDA no serán capaces de preservar ninguna estructura compleja en los datos, lo que puede ser necesario para la clasificación.



Limitaciones de LDA

- LDA falla cuando la información discriminatoria no está en la media sino en la varianza de los datos:



- Precisa que S_W sea no singular $\rightarrow (S_W^{-1} S_B) W^* = \lambda W^*$
 - No es aplicable a datos altamente dimensionados donde el número de patrones es menor que el número de características.
- Como discriminante será lineal

Variantes de LDA

- **LDA no paramétrico**, “NPLDA” (Fukunaga)
 - Este método no necesita la suposición de gaussianidad en las distribuciones. Para ello, calcula la matriz de dispersión inter-clase \mathbf{S}_B usando información local y la regla de \mathbf{K} vecinos más próximos.
 - Como resultado de esto:
 - La matriz \mathbf{S}_B tiene orden máximo, permitiéndonos extraer más de $\mathbf{C}-1$ características.
 - Las proyecciones son capaces de preservar la estructura de los datos de una manera más precisa.
- **LDA ortonormal** (Okada y Tomita)
 - Se computan proyecciones que maximizan $J(w)$ y a la vez son ortonormales entre sí.
 - Se combina lo obtenido con Fisher con el proceso de ortonormalización de Gram-Schmidt
 - Es capaz de encontrar más de $\mathbf{C}-1$ características.
- **LDA generalizado** (Lowe)
 - Se generaliza lo desarrollado con Fisher incluyendo funciones de costo similares a las usadas al calcular el Riesgo de Bayes.
 - El efecto es una proyección LDA cuya estructura está sesgada por la función de costo.
 - Las clases con costos \mathbf{C}_{ij} mayores se separarán más en el espacio de proyecciones.
- **Perceptrones multicapa** (Webb y Lowe)
 - Estos autores demostraron que las capas ocultas de perceptrones multi-capas (**MLP**) efectúan un **análisis discriminante no lineal** maximizando $\text{Tr} [\mathbf{S}_B \mathbf{S}_T^+]$, donde las matrices de dispersión se miden a la salida de la última capa oculta. [Nota: $\mathbf{S}_T = \mathbf{S}_W + \mathbf{S}_B$].

PCA: DEFINICIÓN Y OBJETIVO

Busca comprimir la información que hay en los datos



Construcción de **componentes o factores más relevantes** a partir de las variables originales

resumen la información

representan los datos

Simplificar la estructura de los datos transformando las variables originales en otras llamadas componentes principales a través de combinaciones lineales de las mismas: $Y=W^T x$

Definición formal de PCA. 1

- El objetivo de PCA es realizar una reducción de la dimensionalidad preservando lo máximo posible la aleatoriedad (varianza) original en el espacio de alta dimensión

- Sea un vector \mathbf{x} de \mathbf{N} dimensiones, y una base de vectores **ortonormales** $\{\boldsymbol{\varphi}_1, \boldsymbol{\varphi}_2, \dots, \boldsymbol{\varphi}_N\}$:

$$\boldsymbol{\varphi}_i \cdot \boldsymbol{\varphi}_j = \begin{cases} 0 & i \neq j \\ 1 & i = j \end{cases}$$

- Entonces podemos expresar \mathbf{x} como una combinación lineal de estos vectores:

$$\mathbf{x} = \sum_{i=1}^N y_i \boldsymbol{\varphi}_i$$

- Supongamos que queremos representar \mathbf{x} con solamente \mathbf{M} ($\mathbf{M} < \mathbf{N}$) de los vectores de la base. Podemos hacer esto si sustituimos los componentes $[\mathbf{Y}_{\mathbf{M}+1}, \dots, \mathbf{Y}_{\mathbf{N}}]^T$ por unas constantes preseleccionadas \mathbf{b}_i :

$$\hat{\mathbf{x}}(\mathbf{M}) = \sum_{i=1}^{\mathbf{M}} y_i \boldsymbol{\varphi}_i + \sum_{i=\mathbf{M}+1}^{\mathbf{N}} \mathbf{b}_i \boldsymbol{\varphi}_i$$

Definición formal de PCA. 2

$$\hat{\mathbf{x}}(M) = \sum_{i=1}^M y_i \boldsymbol{\varphi}_i + \sum_{i=M+1}^N b_i \boldsymbol{\varphi}_i$$

- El error que tenemos en esta representación de \mathbf{x} es:

$$\Delta \mathbf{x}(M) = \mathbf{x} - \hat{\mathbf{x}}(M) = \sum_{i=1}^N y_i \boldsymbol{\varphi}_i - \left(\sum_{i=1}^M y_i \boldsymbol{\varphi}_i + \sum_{i=M+1}^N b_i \boldsymbol{\varphi}_i \right) = \sum_{i=M+1}^N (y_i - b_i) \boldsymbol{\varphi}_i$$

- $\Delta \mathbf{x}(M)$ es un **vector** diferencia. La magnitud del error es el módulo de este vector.
- Podemos cuantificar el error que cometemos en promedio para cualquier \mathbf{x} a través del promedio de la magnitud al cuadrado: error cuadrático medio.
- El objetivo es entonces encontrar los vectores base $\boldsymbol{\varphi}_i$ y las constantes b_i que minimizan este error cuadrático medio:

$$\bar{\varepsilon}^2(M) = E[|\Delta \mathbf{x}(M)|^2] = E\left[\sum_{i=M+1}^N \sum_{j=M+1}^N (y_i - b_i)(y_j - b_j) \boldsymbol{\varphi}_i^T \boldsymbol{\varphi}_j \right] = \sum_{i=M+1}^N E[(y_i - b_i)^2]$$

Definición formal de PCA. 3

- El problema es entonces encontrar los vectores base $\boldsymbol{\varphi}_i$ y las constantes \mathbf{b}_i que minimizan el error cuadrático medio, bajo la restricción de que los vectores $\boldsymbol{\varphi}_i$ deben formar una base ortonormal.
- Este problema se puede resolver analíticamente, llegando a una solución cerrada: los $\boldsymbol{\varphi}_i$ son los autovectores de $\boldsymbol{\Sigma}_x$, siendo ésta la matriz de covarianza de \mathbf{x} . $\text{Cov}(\mathbf{X}_i, \mathbf{X}_k) = \mathbf{E} [(\mathbf{X}_i - \mu_i)(\mathbf{X}_k - \mu_k)]$
- Entonces, el error cuadrático medio es:
$$\bar{\varepsilon}^2(M) = \sum_{i=M+1}^N \lambda_i$$

autovalores de los autovectores que no escogemos.

Esto quiere decir que debemos escoger los M autovectores cuyos autovalores son los mayores

- **Resumiendo:** si queremos aproximar con el mínimo error cuadrático medio un vector aleatorio de N dimensiones, a través de una combinación lineal de M vectores, ($M < N$), debemos escoger que esos M vectores sean los autovectores de $\boldsymbol{\Sigma}_x$ con mayores autovalores.

Comentarios sobre PCA

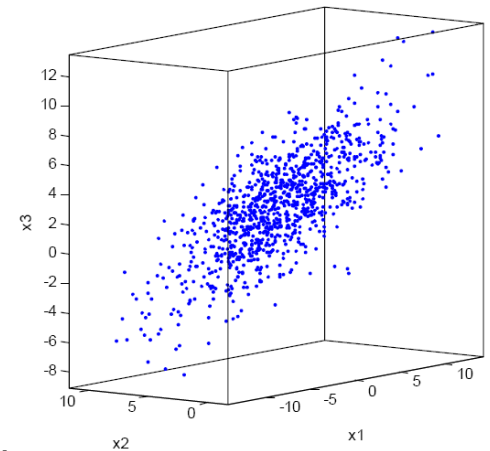
- Ya que PCA elige los autovectores de la matriz de covarianza Σ_x , es capaz de encontrar los ejes independientes de los datos cuando éstos están distribuidos gaussianamente
 - Para distribuciones no Gaussianas (multimodales, por ejemplo), PCA simplemente *decorrelaciona* los ejes (las nuevas variables tienen correlación 0 entre ellas).
- La principal limitación de PCA es que no tiene en cuenta la separabilidad de las clases ya que no tiene en cuenta las clases de los vectores \mathbf{x} → **Método No supervisado**
 - PCA simplemente realiza una rotación de coordenadas que alinea los ejes transformados con las direcciones de máxima varianza.
 - **No hay garantía alguna de que los ejes de máxima varianza contengan una buena información para la clasificación**
- Comentarios históricos:
 - PCA es la técnica más antigua de análisis multivariable
 - Se conoce también como “transformada de Karhunen-Loève” en otros campos como la Teoría de la Comunicación y la Física.

Ejemplo 1

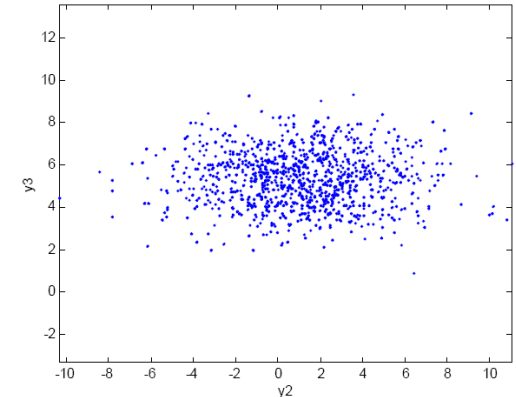
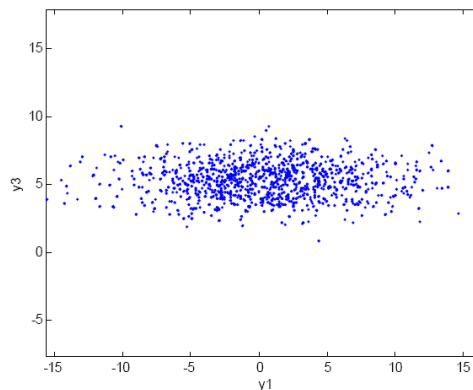
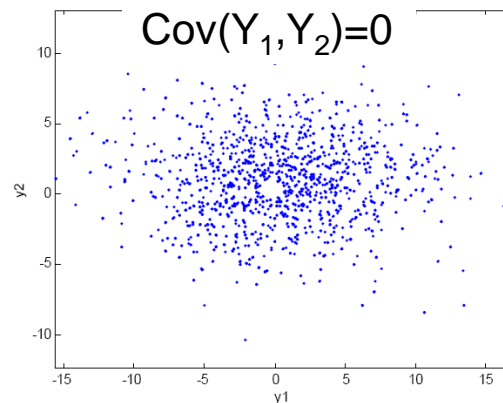
- En este ejemplo tenemos una distribución gaussiana en tres dimensiones con los siguientes parámetros:

$$\text{Cov}(X_i, X_k) = E [(X_i - \mu_i)(X_k - \mu_k)]$$

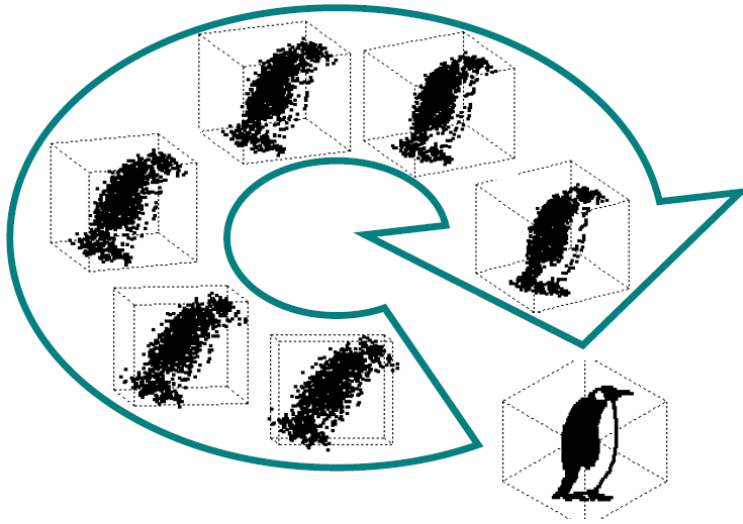
$$\mu = [0 \ 5 \ 2]^T \quad \Sigma = \begin{bmatrix} 25 & -1 & 7 \\ -1 & 4 & -4 \\ 7 & -4 & 10 \end{bmatrix}$$



- A continuación mostramos los tres pares de proyecciones en los componentes principales
 - La primera proyección tiene la mayor varianza, seguida por la segunda
 - Las proyecciones PCA “decorrelacionan” los ejes. $\text{Cov}(Y_i, Y_k) = 0$

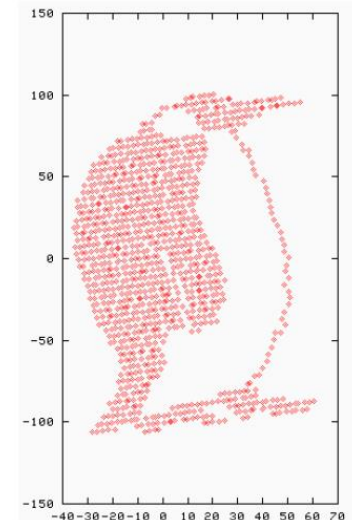


Ejemplo 2



- Ahora tenemos una nube de datos en 3 dimensiones
- Inicialmente, excepto por un alargamiento en la nube de puntos, no hay estructura aparente
- Elegir una rotación apropiada nos permite descubrir la estructura que hay por debajo (podemos pensar en esta rotación como el “caminar” en 3 dimensiones, buscando el mejor punto de vista).

- PCA nos puede ayudar en encontrar la estructura implícita en nuestros datos. Selecciona una rotación tal que casi toda la variabilidad de los datos es representada en las primeras componentes principales.
 - En nuestro ejemplo no parece de mucha ayuda.
 - Sin embargo, cuando tenemos docenas de dimensiones, PCA es muy potente



Ejemplo 3

- Finalmente, resolveremos “a mano” un problema de PCA.
- Computar los componentes principales de los siguientes datos en dos dimensiones:

$$X=(x_1,x_2)=\{(1,2),(3,3),(3,5),(5,4),(5,6),(6,5),(8,7),(9,8)\}$$

- La estimación (sesgada) de Σ_x , es:

$$\text{Cov}(X_i, X_k) = E [(X_i - \mu_i)(X_k - \mu_k)] \quad \Sigma_x = \begin{bmatrix} 6.25 & 4.25 \\ 4.25 & 3.5 \end{bmatrix}$$

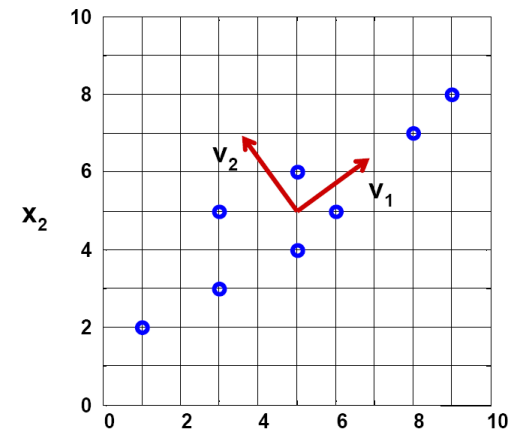
- Los autovalores son los ceros de la ecuación característica:

$$\Sigma_x v = \lambda v \Rightarrow |\Sigma_x - \lambda I| = 0 \Rightarrow \begin{vmatrix} 6.25 - \lambda & 4.25 \\ 4.25 & 3.5 - \lambda \end{vmatrix} = 0 \Rightarrow \lambda_1 = 9.34; \lambda_2 = 0.41;$$

- Los autovectores son las soluciones del sistema:

$$\begin{bmatrix} 6.25 & 4.25 \\ 4.25 & 3.5 \end{bmatrix} \begin{bmatrix} v_{11} \\ v_{12} \end{bmatrix} = \begin{bmatrix} \lambda_1 v_{11} \\ \lambda_1 v_{12} \end{bmatrix} \Rightarrow \begin{bmatrix} v_{11} \\ v_{12} \end{bmatrix} = \begin{bmatrix} 0.81 \\ 0.59 \end{bmatrix}$$

$$\begin{bmatrix} 6.25 & 4.25 \\ 4.25 & 3.5 \end{bmatrix} \begin{bmatrix} v_{21} \\ v_{22} \end{bmatrix} = \begin{bmatrix} \lambda_2 v_{21} \\ \lambda_2 v_{22} \end{bmatrix} \Rightarrow \begin{bmatrix} v_{21} \\ v_{22} \end{bmatrix} = \begin{bmatrix} -0.59 \\ 0.81 \end{bmatrix}$$



Nota: para resolver este sistema, suponemos que una de las variables vale 1 (p. ej., $v_{i1}=1$), y entonces calculamos la otra (en este caso, v_{i2}). Finalmente, las dividimos por un mismo factor para que la norma sea 1.

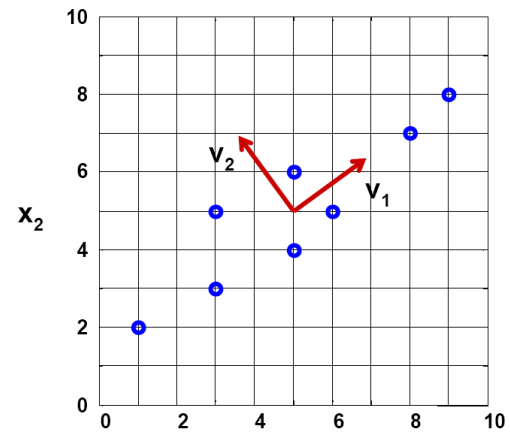
```
octave:1> A=[6.25 4.25;4.25 3.5]
A =
```

```
6.2500  4.2500
4.2500  3.5000
```

```
octave:2> [V,D]=eig(A)
V =
```

```
0.58829 -0.80865
-0.80865 -0.58829
```

$$\Sigma_x = \begin{bmatrix} 6.25 & 4.25 \\ 4.25 & 3.5 \end{bmatrix}$$



D =

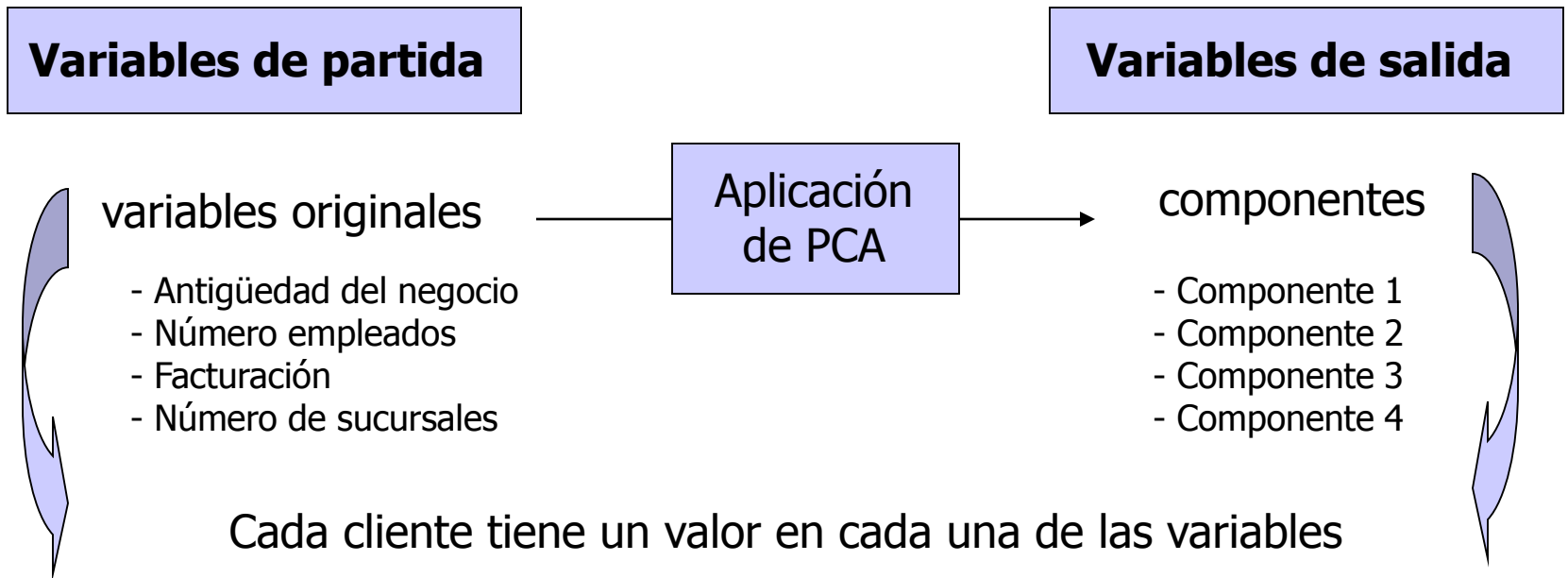
Diagonal Matrix

```
0.40811  0
0  9.34189
```

**Primera Componente explica
el 95.81%**

$$Y = \begin{pmatrix} 0.81 & 0.59 \end{pmatrix} \begin{pmatrix} X_1 \\ X_2 \end{pmatrix}$$

USO PRÁCTICO DE PCA



Cada cliente tiene un valor en cada una de las variables

Las variables originales tienen sentido de negocio

Las variables de salida, componentes, sintetizan la información de las variables originales

└─ es difícil darles un sentido de negocio

CONDICIONES PARA SU APLICACIÓN (I)

Tipo de las variables

Por las propiedades y características del modelo **sólo** está permitido el uso de **variables numéricas**

Si se dispone de **variables categóricas** que describan directamente el problema



Para que intervengan en el análisis de componentes principales, es necesario convertirlas en numéricas



Transformar la variable creando variables dicotómicas

CONDICIONES PARA SU APLICACIÓN (II)

Escala de medida → PCA no es invariante de escala!!!

La escala en la que esté medida la variable influye en esta técnica

Si no se quiere dar importancia a una variable por la escala en la que viene medida



Normalización (o “estandarización”) de los datos

Correlaciones

PCA es una técnica que **tiene sentido aplicarse** en el caso de existir **altas correlaciones entre las variables** (indicio de que existe información redundante) como consecuencia, pocos factores explicarán gran parte de la variabilidad total.

APLICACIÓN DE PCA (III)

Matemáticamente: Partiendo de K variables iniciales



Construcción de una matriz a partir de los datos de partida

└─ **Matriz de Covarianza** si los datos no están estandarizados
└─ **Matriz de Correlaciones** si los datos están estandarizados.

Cálculo de los valores propios y vectores propios de la matriz

└─ λ_j y $(a_{j1}, a_{j2}, \dots, a_{jk})^T$ para $j = 1, \dots, k$

Cálculo de los nuevos atributos ("componentes")

└─
$$\left. \begin{array}{l} C_1 = a_{11}X_1 + \dots + a_{1k}X_k \\ \vdots \\ C_k = a_{k1}X_1 + \dots + a_{kk}X_k \end{array} \right\} \rightarrow \text{Las componentes sintetizan la información de las variables}$$

APLICACIÓN DE PCA (IV)

Características de las componentes o factores

La correlación entre factores diferentes es 0 $\text{Cov}(Y_i, Y_k) = 0$

Los primeros tienen más relevancia (más información) que los últimos

Los factores sintetizan la información de las variables originales

Si los atributos antiguos estaban normalizados:

Los nuevos atributos ("componentes" o "factores") tienen media 0

La varianza de cada factor es igual al valor propio asociado

El Análisis de Componentes Principales estudia las relaciones que las variables tienen entre sí, descubriendo grupos de variables muy correlacionadas entre sí

APLICACIÓN DE PCA (V)

Elección de componentes

Si nos quedamos con todas las componentes calculadas, no se gana nada con respecto a la situación original del problema

Criterios para la ayuda a la elección de factores o componentes

- 1.- Seleccionar tantos factores necesarios para sumar al menos el 80% del valor total de la varianza
- 2.- Seleccionar todos los factores con valor propio ≥ 1 en el caso de Matriz de correlación. Con covarianza seleccionar aquellas componentes cuyos autovalores superen el promedio de todos los autovalores

$$\sum_{i=1}^p \frac{\lambda_i}{p}$$

- 3.- Representar en una gráfica los valores propios y seleccionar el número de factores en función de un cambio brusco

APLICACIÓN DE PCA (VI)

Ejemplo 1: calificación de 15 alumnos en distintas asignaturas

Se dispone de la información de 15 alumnos en cuanto a distintas materias: lengua, matemáticas, física, inglés, filosofía, historia, química y gimnasia.

Factor	Variabilidad explicada (autovalor / suma de los 8 autovalores)	Variabilidad acumulada
1	0.464	0.464
2	0.358	0.821 (0.464+0.358)
3	0.119	0.941 (0.464+0.358+0.119)
4	0.027	0.968 (0.464+ ... + 0.027)
...
8	0.002	1

Construcción de 8 factores porque hay 8 variables independientes

Con 8 factores se consigue explicar toda la información

Cada factor aporta menos información que el anterior

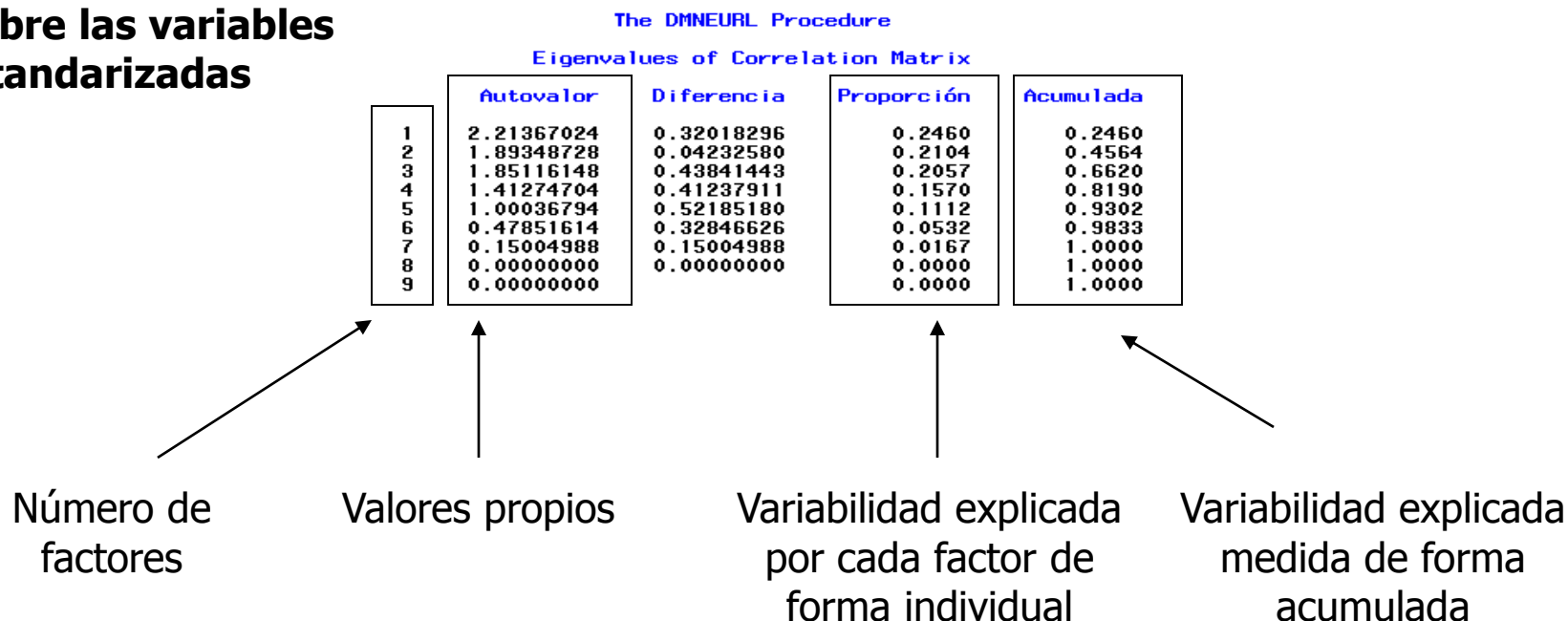
Con el primer factor se explica casi el 50% de la información

Con 3 factores se consigue explicar el 94% de la información

APLICACIÓN DE PCA (VII)

Ejemplo 2: eliminación de variables redundantes en el problema de impago en PYMES

PCA sobre las variables estandarizadas



¿Por qué hay 9 factores? → Porque hay 9 variables independientes

¿PCA detecta redundancia en las variables? → SÍ

¿QUÉ APLICACIONES TIENE PCA? (I)

1. Reducir el número de variables del problema mediante **extracción de características**: $Y = W^T X$

Nos **olvidamos** de las antiguas variables y trabajamos en adelante con las variables sintéticas obtenidas de los componentes principales. Por ejemplo, entrenamos un árbol de decisión con ellas.

2. Reducir el número de variables del problema mediante **selección** de variables:

Usamos PCA para que nos “diga” qué variables antiguas podemos “tachar” (no aportan información relevante o son redundantes).

Aquellas características o variables cuyos coeficientes sean elevados para los primeros componentes principales, serán las que aporten la mayor capacidad discriminativa

A partir de ese momento, trabajamos sólo con las **variables antiguas** relevantes.

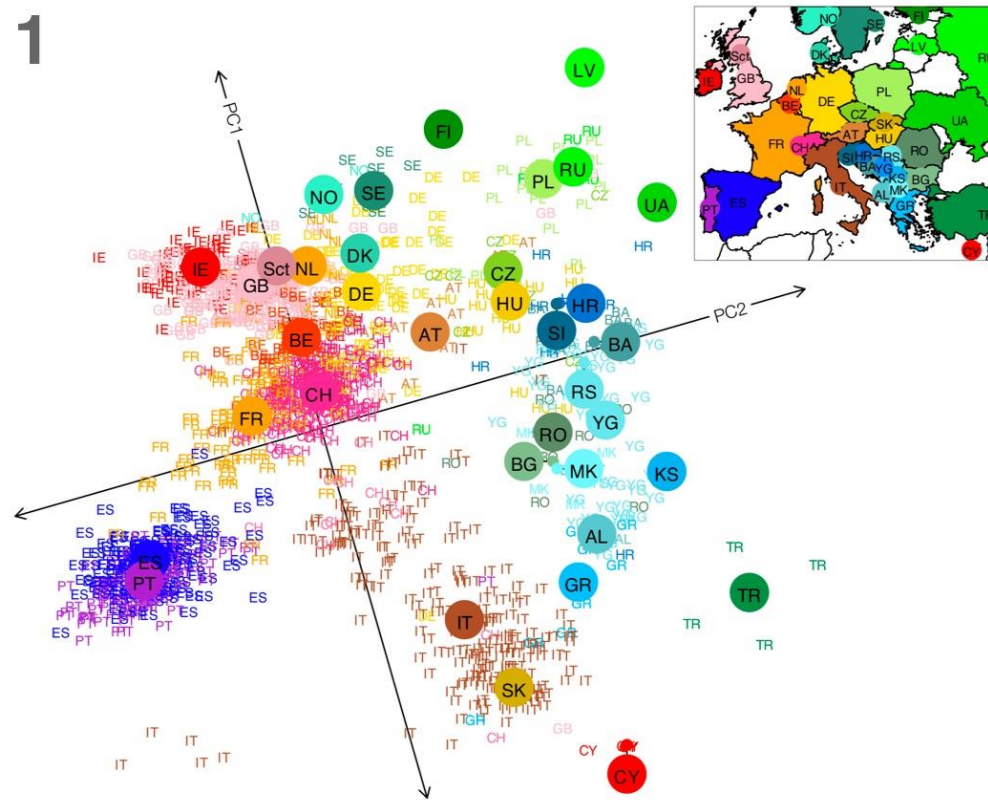
3. Detección de “**outliers**”:

Usamos PCA para detectar qué ejemplos son atípicos. Diagramas de dispersión en los factores.

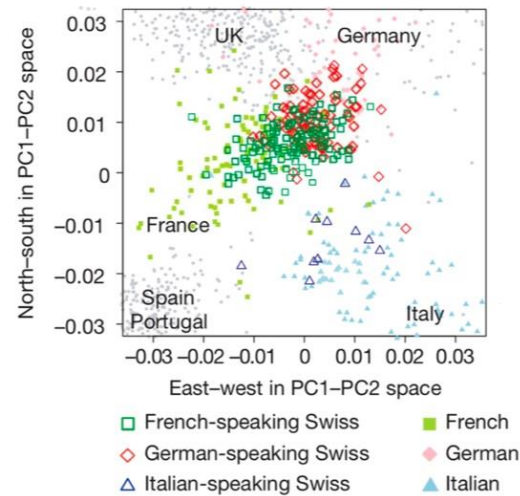
4. Descubrimiento de “**clusters**”:

Usamos PCA para detectar grupos diferentes de ejemplos.

1

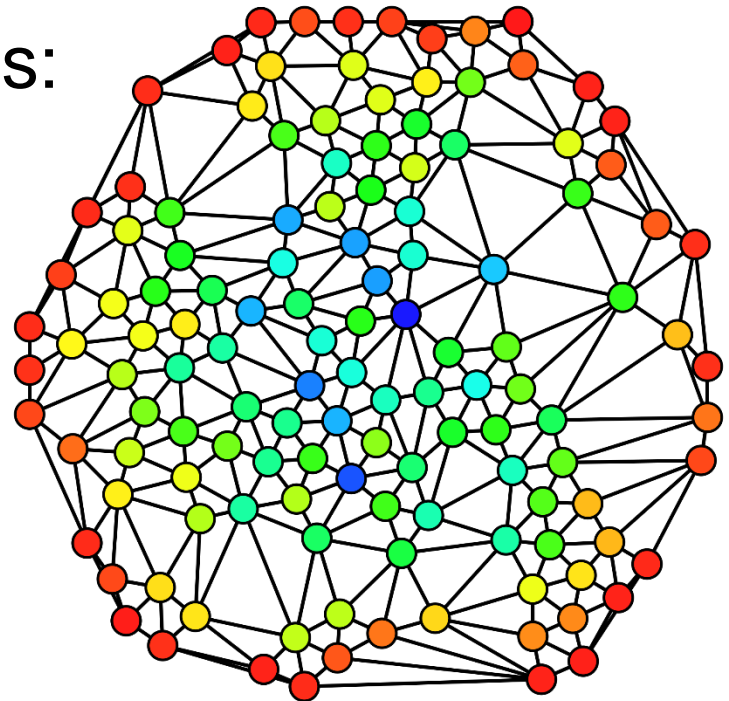


2



Métodos espectrales para reducción de dimensionalidad

- Métodos lineales:
 - LDA
 - PCA
 - MDS (Metric multiDimensional Scaling): Conserva la distancia entre pares de patrones. Similar a PCA
- Métodos basados en grafos:
 - Isomap
 - Maximum variance unfolding
 - Locally Linear Embedding (LLE)
 - Laplacian Eigenmaps
- Kernel Methods
 - Kernel PCA
 - Graph-Based Kernel



Locally Linear Embedding

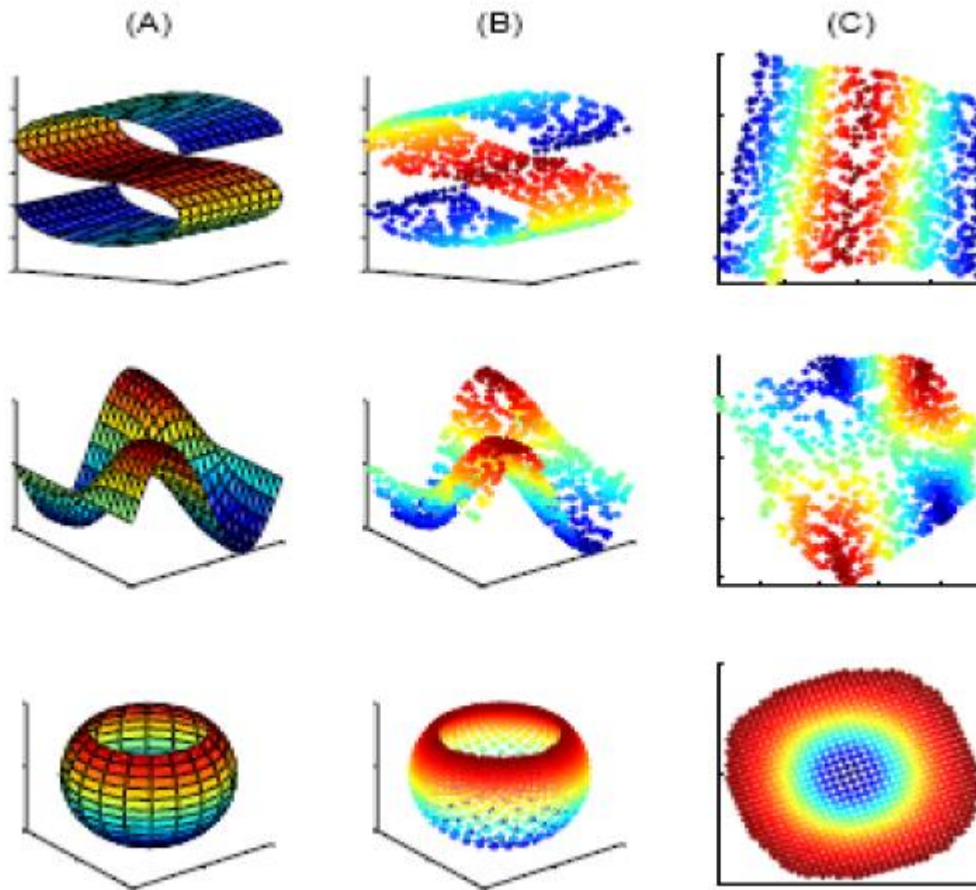
LLE:

- algoritmo de reducción de dimensionalidad de estructura **no lineal**, sin embargo parte de un ajuste lineal.
- algoritmo no supervisado basado en teoría de grafos

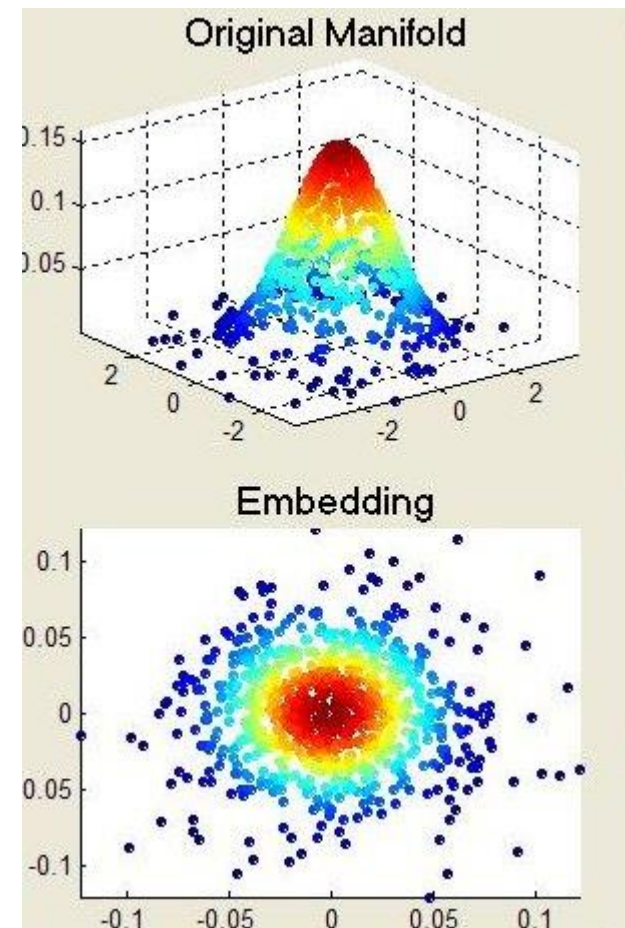
Necesidades:

- Variedad = Manifold = Espacio Original
- $S = \{x_i\}$ D-dimensional
- Número suficiente de datos = Manifold is well-sampled

Motivación: Conservar en el nuevo espacio reducido la geometría existente en el espacio original. ➔ Capturar la geometría intrínseca del vecindario



Embedding



Locally Linear Embedding

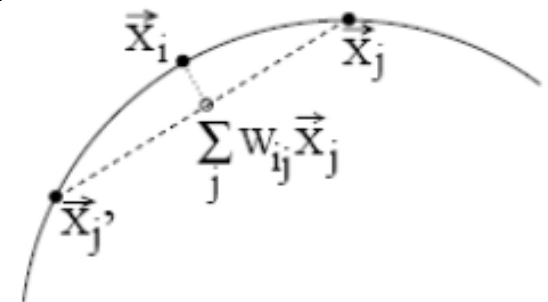
1. Reconstrucción de un punto x_i con los puntos de su vecindario:

$$\hat{x}_i = \sum_{j=1}^K W_{ij} x_j$$

K número de vecinos
próximos al punto x_i

2. Error cometido para el punto x_i : $\varepsilon_i = \left| x_i - \sum_{j=1}^K W_{ij} x_j \right|^2$

3. Función de coste: $\varepsilon(W) = \sum_{i=1}^N \left| x_i - \sum_{j=1}^K W_{ij} x_j \right|^2$



Locally Linear Embedding

- Calcular los pesos W :

$\min(\varepsilon(W))$ sujeto a:

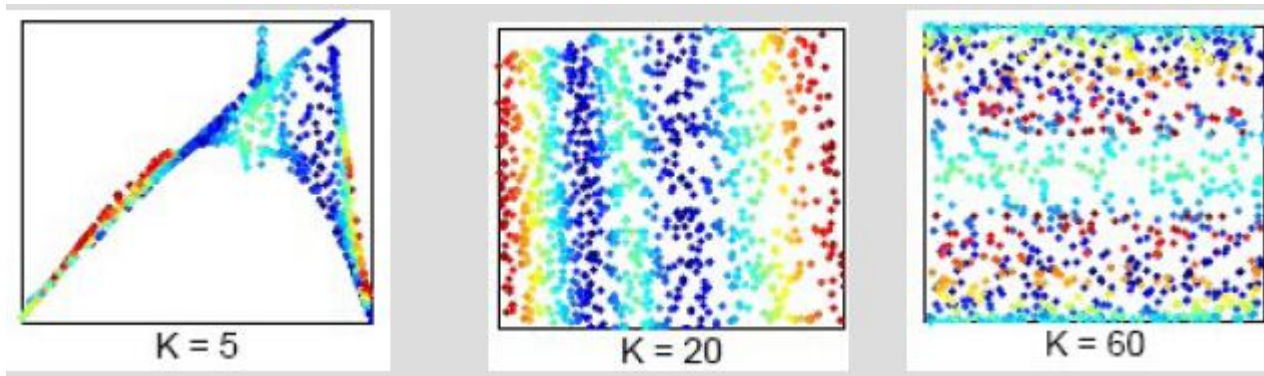
$$\sum_j W_{ij} = 1$$

$$W_{ij} = 0 \quad \text{si } x_j \text{ no pertenece al conjunto de vecinos de } x_i$$

Solución Cerrada: problema de mínimos cuadrados con restricciones.

Locally Linear Embedding

- Efecto de los vecinos próximos



Locally Linear Embedding

Para cada punto i :

1. Calcular la matriz de correlación de sus vecinos próximos C_i

$$C_{jk} = x_j^T x_k$$

2. Calcular la inversa de C_i

3. Calcular los multiplicadores de Lagrange

$$\alpha = 1 - \sum_j \sum_k C_{jk}^{-1} x_i^T x_k \quad \beta = \sum_{j,k} C_{jk}^{-1}$$

4. Valor de los coeficientes
$$W_{ij} = \sum_k C_{jk}^{-1} \left(x_i^T x_k + \frac{\alpha}{\beta} \right)$$

$$W_{ij} = 0 \quad \text{si } x_j \text{ no pertenece al conjunto de vecinos de } x_i$$

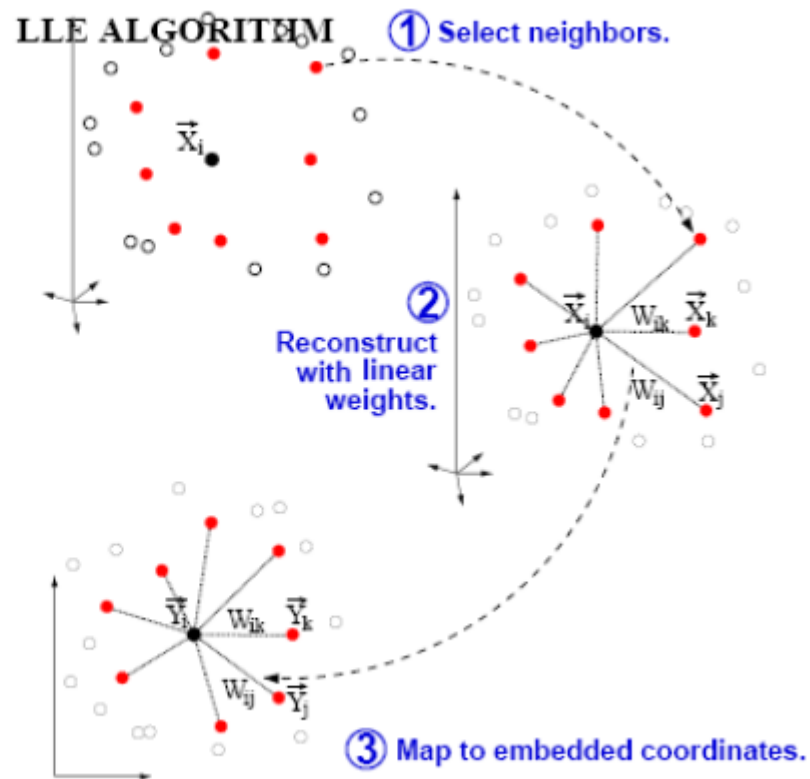
Cálculo
de los
coeficientes
 W

Locally Linear Embedding

- W obedecen una importante simetría:
 - Invariantes rotaciones
 - Invariantes reescalado $\rightarrow \sum_j W_{ij} = 1$
 - Invariantes translaciones de los datos y su vecindario
- LLE construye vecindarios en un nuevo espacio d -dimensional $d \ll D$ basándose en propiedades de simetría.
 $f: x_i \rightarrow y_i$

Locally Linear Embedding

- LLE ALGORITHM**
1. Compute the neighbors of each data point, \vec{X}_i .
 2. Compute the weights W_{ij} that best reconstruct each data point \vec{X}_i from its neighbors, minimizing the cost in Equation (1) by constrained linear fits.
 3. Compute the vectors \vec{Y}_i best reconstructed by the weights W_{ij} , minimizing the quadratic form in Equation (2) by its bottom nonzero eigenvectors.



Locally Linear Embedding

- En el nuevo espacio de dimensión $d \ll D$

1. Reconstrucción de un punto y_i con los puntos de su vecindario:

$$\hat{y}_i = \sum_{j=1}^K W_{ij} y_j \quad W_{ij} \text{ Coeficientes encontrados anteriormente}$$

2. Minimizar la nueva función de coste

$$\theta(Y) = \sum_{i=1}^N \left| y_i - \sum_{j=1}^K W_{ij} y_j \right|^2 \quad \text{Con las restricciones:}$$

$$\sum_i y_i = 0 \quad \text{Eliminar desplazamientos constantes centrando en cero}$$

$$\text{Covarianza} = I_{d \times d} \quad \text{Evitar soluciones degeneradas}$$

Locally Linear Embedding

$$\min(\theta(Y)) \quad \theta(Y) = \sum_{i=1}^N \left| y_i - \sum_{j=1}^K w_{ij} y_j \right|^2$$

$$\theta(Y) = \sum_{i,j} M_{ij} y_i^T y_j \quad \text{donde} \quad M_{ij} = \delta_{ij} - W_{ij} - W_{ji} + \sum_k W_{ki} W_{kj}$$

$$M = (I - W)^T (I - W)$$

M es una matriz simétrica semidefinida positiva

M es una matriz dispersa, pues W también lo es

$$\min(\theta(Y)) \approx \min(M)$$

Locally Linear Embedding

$$\min(\theta(Y)) \approx \min(M)$$

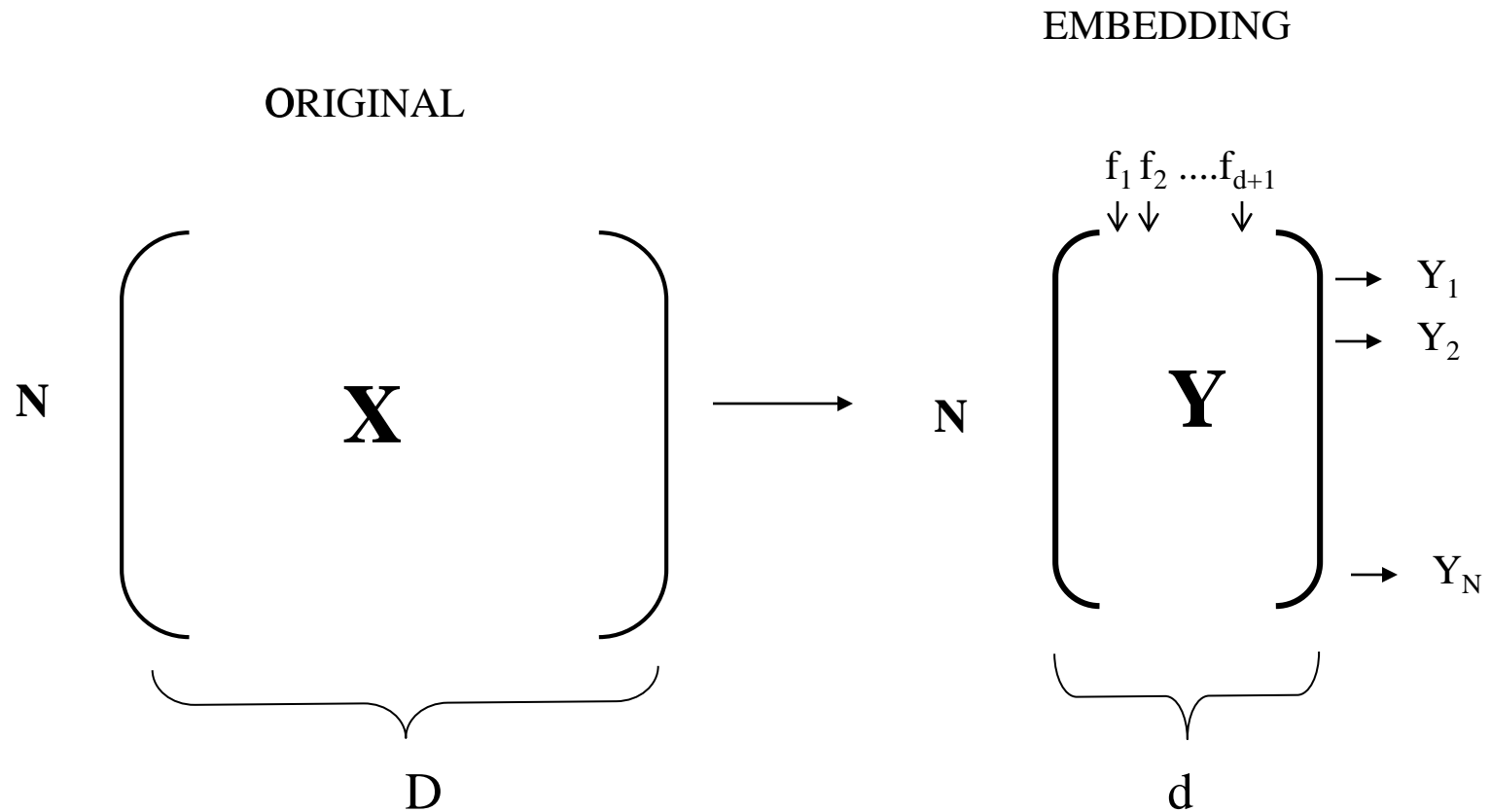
Problema de descomposición espectral:

$$0 = \lambda_1 \leq \lambda_2 \leq \lambda_3 \leq \lambda_4 \cdots \leq \lambda_N$$

$\lambda_1 = 0 \rightarrow$ Autovector unitario (restricción a que los “embeddings” tengan media cero). $\sum_i y_i = 0$

El resto de los d autovectores (autovalores más pequeños) nos van a dar los nuevos y : $x_i \rightarrow y_i = (f_2(i), \dots, f_{d+1}(i))$

Locally Linear Embedding



$f_i \rightarrow$ Autovectores asociados a los menores autovalores de la matriz

$$M = (I - W)^T (I - W)$$

Locally Linear Embedding

- Nuevo dato x

$$y_k(x) = \sum_{i=1}^n y_k(x_i) W(x, x_i)$$

$$k = 1 \dots d$$

i = realmente en **los vecinos de x** , no vecinos
 $W(x, x_i) = 0$

donde $W(x, x_i)$ es el coeficiente de x_i en la reconstrucción de x por sus vecinos más próximos en el conjunto de entrenamiento (volver a la transparencia del cálculo de W , número 76).

$$x = \sum_{j=1}^K W(x, x_j) x_j$$

$$W(x, x_i) = \sum_k C_{ik}^{-1} \left(x^T x_k + \frac{\alpha}{\beta} \right)$$

$$\alpha = 1 - \sum_j \sum_k C_{jk}^{-1} x^T x_k$$

$$\beta = \sum_{j,k} C_{jk}^{-1}$$