Università di Verona

A.A. 2020-21

Machine Learning & Artificial Intelligence

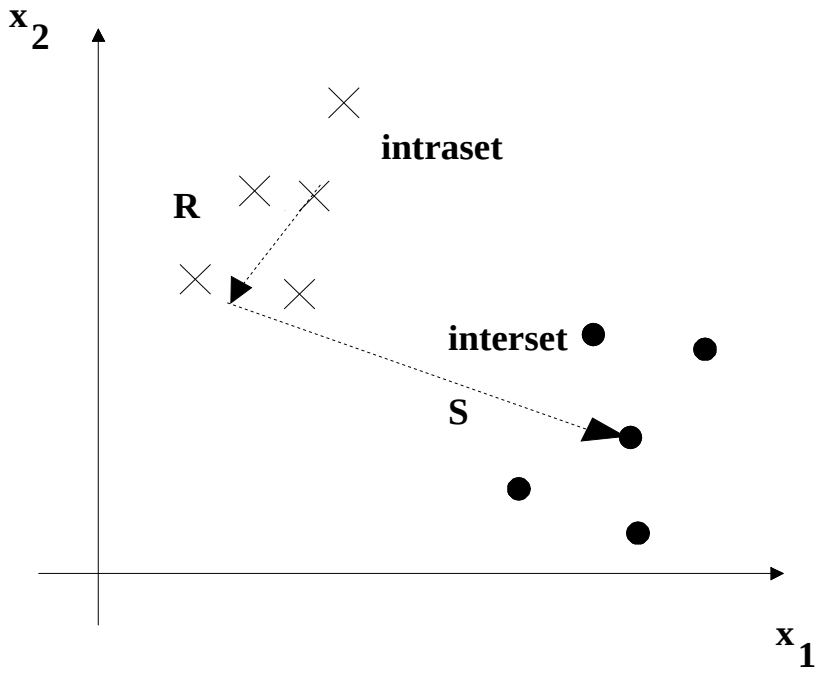
Trasformazioni lineari, metodo di Fisher
Metodi di estrazione e selezione delle feature,
Principal Component Analysis

Vittorio Murino

Ancora sulle Trasformazioni Lineari

- L'idea di base consiste nel cercare una trasformazione $\mathbf{W}$ che porta dai campioni $\mathbf{y}$ mal strutturati in un nuovo insieme $\mathbf{x}$ più semplice per la classificazione:

$$\mathbf{x} = \mathbf{W}\,\mathbf{y}$$



- Il problema è determinare la matrice di trasformazione $\mathbf{W}$.
- Si considerano due tipi di distanze:

$$\text{INTERSET} \Rightarrow S \qquad \text{INTRASET} \Rightarrow R$$

- Il parametro S è definito come la media di tutte le possibili distanze tra due campioni appartenenti a classi diverse.

$$S = \frac{1}{M_1 M_2} \sum_{i=1}^{M_1} \sum_{j=1}^{M_2} d^2\left[\mathbf{y}_i^{(1)}, \mathbf{y}_j^{(2)}\right]$$

- Il parametro R invece considera tutte le possibili distanze all'interno di una stessa classe.

$$R = \frac{1}{M(M-1)} \sum_{i=1}^{M} \sum_{j=1}^{M} d^2\left[\mathbf{y}_i, \mathbf{y}_j\right], \text{ con M pari a } \mathrm{M_1} \text{ o } \mathrm{M_2}$$

- Operando una trasformazione $\mathbf{W}$ si vuole rendere S massimo e R minimo per ottenere una buona separabilità delle classi.

- Si può quindi definire come obiettivo:

$$Q(\mathbf{x}) = \left.\frac{R_1 + R_2}{S}\right|_{\min}$$

  o criteri analoghi che tendano a massimizzare la distanza Interset e minimizzare quella Intraset.

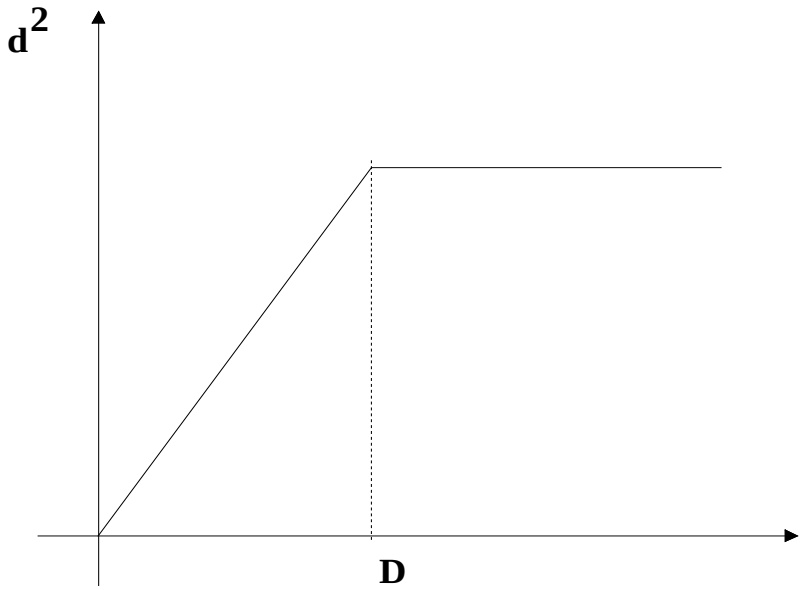
- La trasformazione per avere $Q(\mathbf{x})$ minimo diventa non lineare (e quindi è molto complicata).
- Esistono diversi tipi di distanze:
  - euclidea: non va molto bene perchè pesa molto i punti piú lontani (cosí si ricorre ad un tipo di distanza che tende a saturare);
  - saturazione: lo svantaggio è che con questo tipo di distanza si introducono discontinuità;
  - raccordi continui: questa è una delle soluzioni più semplici.

- Quindi, la scelta migliore per il tipo di distanza è quella di usare una funzione con derivata prima continua che approssimi la distanza euclidea fino ad una certa soglia e poi di considerare la distanza fissa, questo per non pesare troppo i campioni molto lontani.

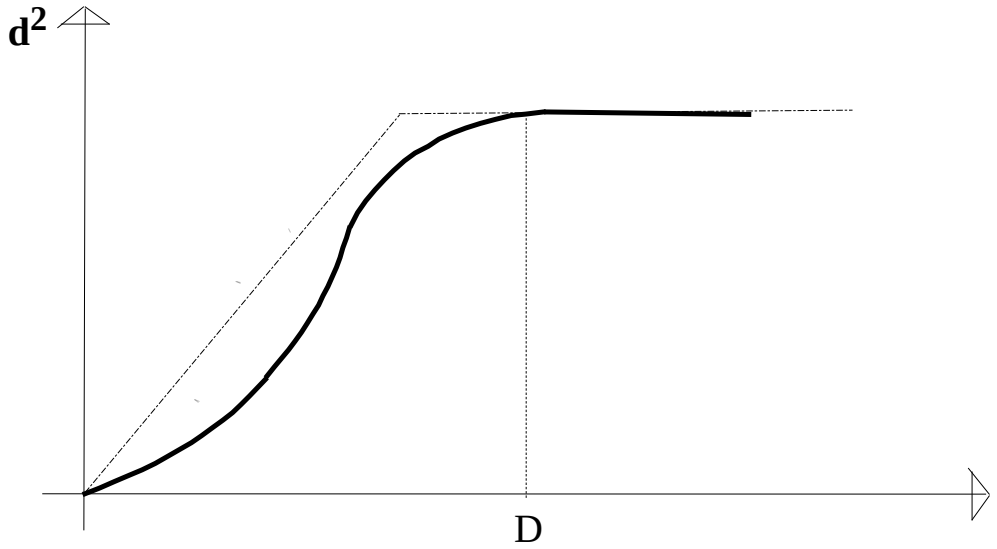
- Una forma di d^2 meno grossolana risulta essere una funzione tipo sigmoide:

$$d^*(\mathbf{y_1},\mathbf{y_2}) = -\exp\left\{-\frac{1}{2D^2}d^2(\mathbf{y_1},\mathbf{y_2})\right\}$$

dove d^2 é la distanza euclidea, che risulta continua e che tende a saturarsi.



- Queste considerazioni sul tipo di distanza valgono sia per R che per S.
- Tutte queste considerazioni vanno bene se i campioni sono distribuiti in maniera statistica (p.e., a nuvola) e non in maniera funzionale (p.e., lamellare).



- Si supponga di voler effettuare una trasformazione lineare $\mathbf{x} = \mathbf{W}\,\mathbf{y}$ tra due spazi in modo da verificare i criteri sopra esposti.
- In altre parole, si consideri $\mathbf{W}$ matrice, quindi si opera la trasformazione riducendo la dimensione dello spazio di partenza ($\mathbf{y}$).

$\Rightarrow \quad \dim(\mathbf{y}) = m$

$\dim(\mathbf{x}) = n$, allora $\dim(\mathbf{W}) = n \times m$

$$\mathbf{x} = \mathbf{W}\,\mathbf{y} \qquad \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} = \begin{bmatrix} w_{11} & \cdots\cdots & w_{1m} \\ \vdots & \ddots & \vdots \\ w_{n1} & \cdots\cdots & w_{nm} \end{bmatrix} \cdot \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{pmatrix}$$

- allora si ha che

$$S = \frac{1}{M_1 M_2} \sum_{q=1}^{M_1} \sum_{p=1}^{M_2} \left\{ \mathbf{W} \left(\mathbf{y}_q^{(1)} - \mathbf{y}_p^{(2)} \right) \right\}^2$$

e sviluppando:

$$S = \frac{1}{M_1 M_2} \sum_{q=1}^{M_1} \sum_{p=1}^{M_2} \sum_{i=1}^{n} \left\{ \sum_{j=1}^{m} w_{ij} \left(y_{q_j}^{(1)} - y_{p_j}^{(2)} \right) \right\}^2$$

$$R_1 = \frac{1}{M_1 (M_1 - 1)} \sum_{q=1}^{M_1} \sum_{p=1}^{M_1} \sum_{i=1}^{n} \left\{ \sum_{j=1}^{m} w_{ij} \left(y_{q_j}^{(1)} - y_{p_j}^{(1)} \right) \right\}^2$$

e analogamente per la seconda classe (ω_2).

- Le incognite sono ovviamente le w_{ij}
- La facilità o meno della soluzione dipende dalla funzione obiettivo che scelgo.
- Si considerino i seguenti casi:

1) $\mathbf{W}$ diagonale ($w_{ij} = 0$ se $i \neq j$ e $w_{ij} \neq 0$ se $i = j$)

Imponendo come obiettivo che $(R_1 + R_2)$ minimo

e ponendo il vincolo $\Sigma_k\, w_{kk} = 1$, che è la condizione di minimo Lagrangiano (altrimenti detto vincolo a perimetro costante), si arriva alla seguente soluzione (quest'ultimo vincolo serve per impedire allo spazio di arrivo di "esplodere", ossia, lo spazio $R_1 + R_2$ di arrivo deve essere contenuto in quello di partenza):

$$\sum_k w_{kk} = 1 \Rightarrow w_{kk} = \frac{1}{\sigma_k^2 \sum_{j=1}^{n} \left(\frac{1}{\sigma_j^2} \right)}$$

con $\sigma_k^2 = \frac{1}{N-1}\sum_{i=1}^{N}\left(y_{i_k}^{(l)} - \bar{y}_k^{(l)}\right)^2$

e $N = M_1 + M_2$ se $(R_1 + R_2)$ minima

$N = M_1$ o M_2 se R_1 o R_2 minimo, rispettivamente.

- Il termine σ_k^2 è definito come la varianza dei campioni lungo la componente k-esima (*feature* k-esima del campione).
- In questo modo, una piccola varianza implica che la misura k-esima é più affidabile, e viceversa, tale che la misura più affidabile sia pesata in maniera maggiore.
- Un'altro vincolo che è possibile imporre è $\prod w_{kk} = 1$ (vincolo a volume costante):

$$\prod_k w_{kk} = 1 \Rightarrow w_{kk} = \frac{1}{\sigma_k}\left(\prod_{j=1}^{n}\sigma_j\right)^{1/n}$$

che è inversamente proporzionale alla deviazione standard della k-esima misura.

2) **W** qualunque

L'obiettivo R_1+R_2 minimo deve essere raggiunto ora con il vincolo $R_1+R_2+S = costante$.

La procedura consiste nel definire due matrici i cui coefficienti possono essere calcolati tenendo conto dei parametri di intraset e interset.

Calcolo gli autovalori di $\mathbf{BC}^{-1}$ dove

$$\mathbf{B} \quad \text{interset} \rightarrow b_{jk} = \frac{1}{M_1 M_2} \sum_{q=1}^{M_1} \sum_{p=1}^{M_2} \left(y_{q_j}^{(1)} - y_{p_j}^{(2)} \right) \cdot \left(y_{q_k}^{(1)} - y_{p_k}^{(2)} \right)$$

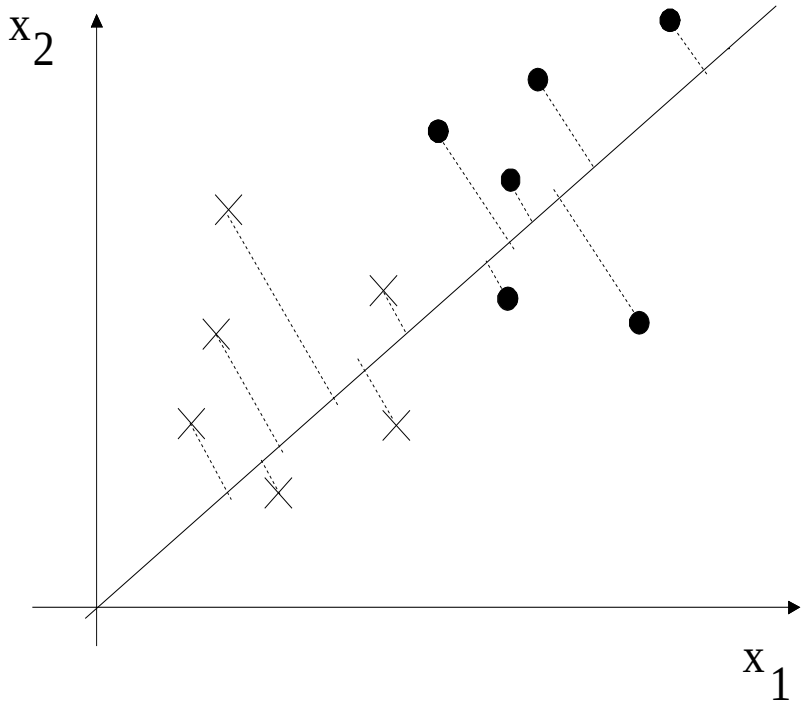
$$\mathbf{C} \quad \text{intraset} \rightarrow c_{jk} = \frac{2}{M(M-1)} \sum_{q=1}^{M} \sum_{p=1}^{M} \left(y_{q_j}^{(l)} - y_{p_j}^{(l)} \right) \cdot \left(y_{q_k}^{(l)} - y_{p_k}^{(l)} \right)$$

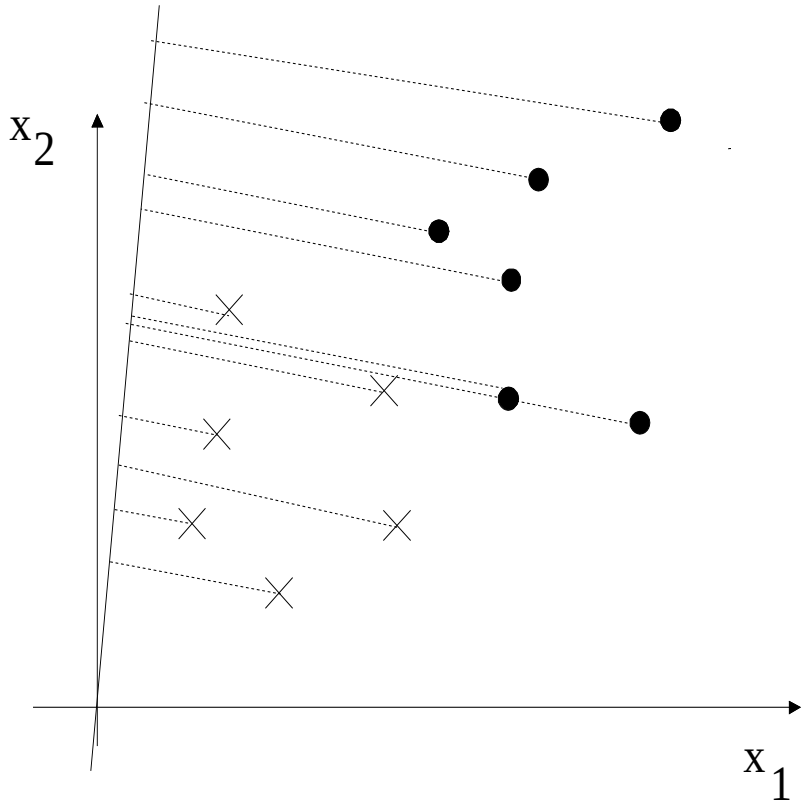
dove $M = M_1 + M_2$ ed $l = 1,2$

- Metto gli autovalori in ordine $\lambda_1 > \lambda_2 > \ldots$
- Calcolo gli autovettori corrispondenti $\mathbf{v}_1$, $\mathbf{v}_2$, ..., quindi: $\mathbf{W} = \begin{bmatrix} \mathbf{v}_1 \\ \mathbf{v}_2 \\ \vdots \\ \mathbf{v}_n \end{bmatrix}$
- Per ridurre lo spazio, si possono considerare meno autovettori.
- L'autovettore $\mathbf{v}_1$ è chiamato *direzione di Fisher* e permette di avere la proiezione lungo un asse con la massima distanza interset e minima intraset.
- Si poteva anche scegliere la funzione obiettivo come $\max\{S\}$ con il vincolo $R_1 + R_2 + S = costante$.
- In pratica però la trasformazione comporta dei calcoli così onerosi per vincoli significativi che non viene usata.

La trasformata di Fisher

- Il problema è quello di ridurre la dimensionalità dello spazio delle features in modo da rendere il problema computazionalmente trattabile.
- È in sostanza la proiezione delle *feature* caratterizzanti un campione su una retta, cioè su una direzione (da un problema *d-dimensionale* ad un problema *1-dimensionale*).
- Ovviamente, se le classi erano ben separate nello spazio a d dimensioni, tipicamente non lo saranno in quello a 1 dimensione (perchè avranno elementi sovrapposti), per cui il problema è cercare l'orientazione della retta per la quale la separazione delle classi è migliore.





- Avrò comunque una perdita, ma tra le possibili trasformazioni quella di Fisher è la migliore.

- Si supponga di avere un insieme di N campioni d-dimensionali $\mathbf{x}_1$, .., $\mathbf{x}_N$, di cui N_1 classificati come ω_1 ed N_2 classificati come ω_2.
- Si vuole cercare una trasformazione $\mathbf{w}$, ossia una combinazione lineare delle componenti di $\mathbf{x}$ tale da generare i corrispondenti campioni (scalari) y_1,..., y_N:

  $\mathbf{w}^t \mathbf{x} = y$
- Geometricamente, se la norma di $\mathbf{w}$ è pari a 1 (ovvero un grado di libertà corrispondente ad una rette di direzione generica), allora ogni y_i è la proiezione del campione $\mathbf{x}_i$ sulla retta di direzione $\mathbf{w}$.
- L'aspetto importante è la direzione di $\mathbf{w}$ non l'ampiezza (in quanto essa comprende solo uno scalamento).

- Siccome si vuole separare le due classi anche nel nuovo spazio monodimensionale allora si considera come misura di separazione la differenza delle medie dei campioni. Quindi:

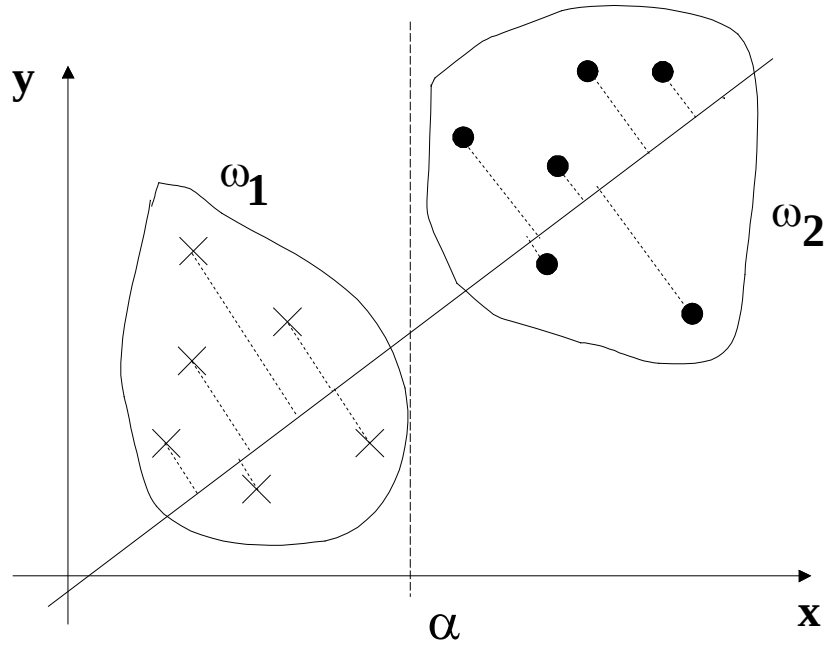
$$\tilde{m}_1 = \mathbf{w}^t \cdot \mathbf{m}_1$$

$$\tilde{m}_2 = \mathbf{w}^t \cdot \mathbf{m}_2$$

dove

$$\left.\begin{aligned} \mathbf{m}_1 &= \frac{1}{N_1} \cdot \sum_{i=1}^{N_1} \mathbf{x}_i^{(1)} \\ \mathbf{m}_2 &= \frac{1}{N_2} \cdot \sum_{i=1}^{N_2} \mathbf{x}_i^{(2)} \end{aligned}\right\} \text{ medie prima della trasformazione}$$

$$\left.\begin{aligned} \tilde{m}_1 &= \frac{1}{N_1} \cdot \sum_{i=1}^{N_1} \mathrm{y}_i^{(1)} \\ \tilde{m}_2 &= \frac{1}{N_2} \cdot \sum_{i=1}^{N_2} \mathrm{y}_i^{(2)} \end{aligned}\right\} \text{ medie dopo la trasformazione}$$



- Si vuole ottenere che la differenza tra le medie delle due classi (trasformate) sia grande rispetto alla deviazione standard di ogni classe.
- Allora, si definisce il discriminante lineare di Fisher come la funzione lineare $\mathbf{w}^t\mathbf{x}$ per la quale la funzione J è massima:

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

dove $\tilde{s}_1$ e $\tilde{s}_2$ sono le dispersioni (*scatter*) dei campioni classificati ω_1 ed ω_2, rispettivamente, definite come:

$$\tilde{s}_i^2 = \sum_{j=1}^{N_i} \left(y_j^{(i)} - \tilde{m}_i \right)^2$$

- Si vuole che le dispersioni siano abbastanza piccole, ossia, che i campioni di una classe siano abbastanza concentrati intorno al valore medio.
- Per ottenere J come una funzione esplicita di $\mathbf{w}$ si definiscono le matrici di dispersione (*scatter matrices*) S_i ed S_w:

$$S_i = \sum_{j=1}^{N_i} \left(\mathbf{x}_j^{(i)} - \mathbf{m}_i \right) \left(\mathbf{x}_j^{(i)} - \mathbf{m}_i \right)^t \qquad S_w = S_1 + S_2$$

- Analogamente:

$$\tilde{s}_i^2 = \sum_{j=1}^{N_i} \left(\mathrm{y}_j^{(i)} - \tilde{\mathrm{m}}_i \right)^2 = \sum_{j=1}^{N_i} \left(\mathbf{w}^t \mathbf{x}_j^{(i)} - \mathbf{w}^t \mathbf{m}_i \right)^2 = \mathbf{w}^t S_i \mathbf{w}$$

- In tal modo:

$$\tilde{s}_1^2 + \tilde{s}_2^2 = \mathbf{w}^t S_w \mathbf{w} \qquad \left(\tilde{m}_1 - \tilde{m}_2 \right)^2 = \mathbf{w}^t S_B \mathbf{w}$$

dove

$S_B = (\mathbf{m}_1 - \mathbf{m}_2) \cdot (\mathbf{m}_1 - \mathbf{m}_2)^{\mathrm{t}}$

- Quindi per ottenere $J(\mathbf{w})$ massimo, si deve esplicitare J in funzione diretta di $\mathbf{w}$ e quindi derivarlo rispetto a $\mathbf{w}$ ed eguagliare a 0.

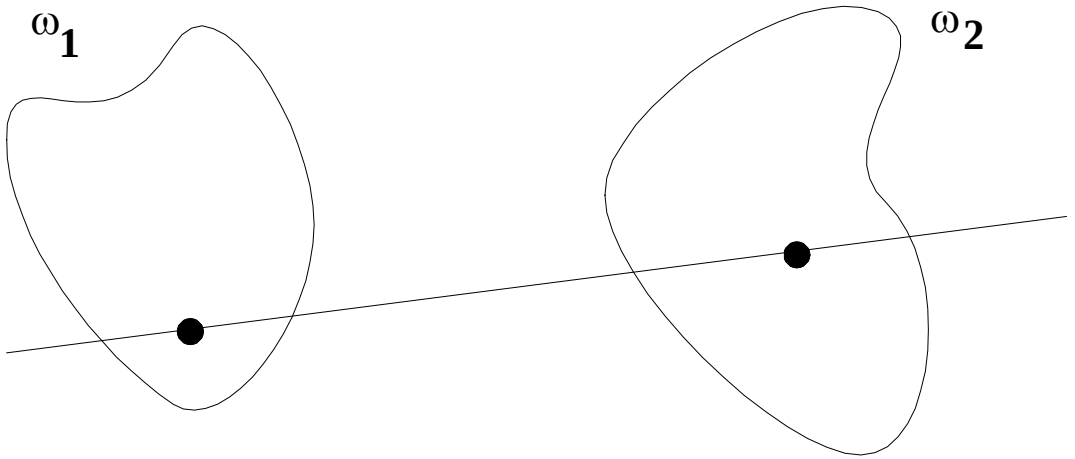
$$J(\mathbf{w}) = \frac{\left| \tilde{m}_1 - \tilde{m}_2 \right|}{\tilde{s}_1^2 + \tilde{s}_2^2} = \frac{\mathbf{w}^t S_B \mathbf{w}}{\mathbf{w}^t S_w \mathbf{w}}$$

- Derivando ottengo che:

$$\frac{\partial J}{\partial \mathbf{w}} = 0 \Rightarrow \mathbf{w} = S_w^{-1}\left(\mathbf{m}_1 - \mathbf{m}_2\right)$$

  che è la trasformata di Fisher.

- La dimostrazione parte con l'assunto che $S_B\mathbf{w}$ è sempre lungo la direzione che congiunge le medie delle due classi.



Il problema della (maledizione della) dimensionalità
Curse of dimensionality

- In problemi pratici si possono trovare anche *training set* di varia natura: pochi campioni, poche/molte feature, feature non selezionabili, feature non indipendenti

- Ci sono 2 problemi importanti per il progetto di un classificatore
  - la complessità computazionale del sistema,
  - l'influenza della dimensionalità (e la cardinalità) del training set sull'accuratezza della classificazione.

- Se le feature sono statisticamente indipendenti allora si può dimostrare che si raggiungono prestazioni ottime
- Esempio: pb. a 2 classi, normale, multivariato, covarianza uguale: $p(\mathbf{x}|\omega_j) = N(\boldsymbol{\mu}_j, \boldsymbol{\Sigma})$
- Si può dimostrare che, con *prior* uguali, la probabilità d'errore è data da

$$P(e) = \frac{1}{\sqrt{2\pi}} \int_{r/2}^{\infty} e^{-\frac{u^2}{2}} du$$

dove r^2 è il quadrato della distanza di Mahalanobis

$$r^2 = (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^t \, \boldsymbol{\Sigma}^{-1} \, (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)$$

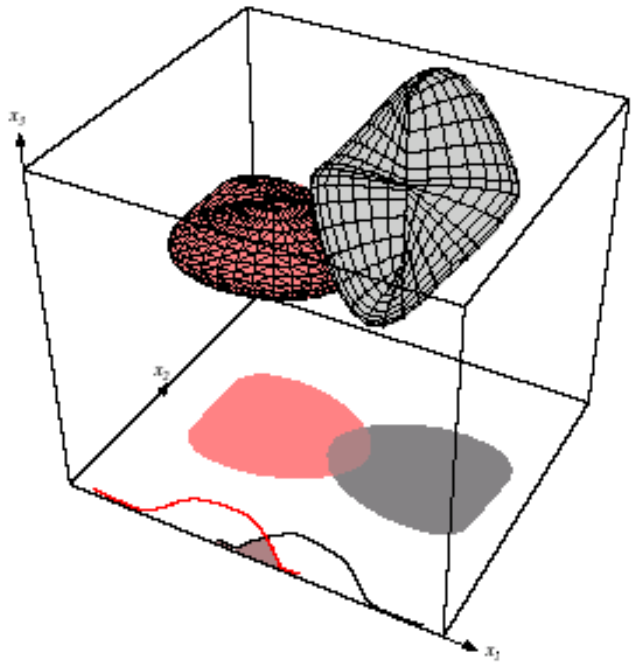
- Si noti che la probabilità di errore decresce quando *r* cresce, tendendo a 0 quando *r* tende ad infinito

- Nel caso $\Sigma = \mathrm{diag}(\sigma_1^2, \ldots, \sigma_d^2)$ (diagonale) allora

$$r^2 = \sum_{k=1}^{d} \left(\frac{\mu_{i1} - \mu_{i2}}{\sigma_i} \right)^2$$

- Qui si vede che ogni feature (indipendente) contribuisce a diminuire la probabilità d'errore, fino a renderla arbitrariamente piccola
- In generale, se le prestazioni sono inadeguate, si possono aggiungere feature, ma al costo di complicare il classificatore e l'estrattore di feature
- Tuttavia se la struttura probabilistica è nota, il rischio di Bayes può non variare anche aggiungendo feature

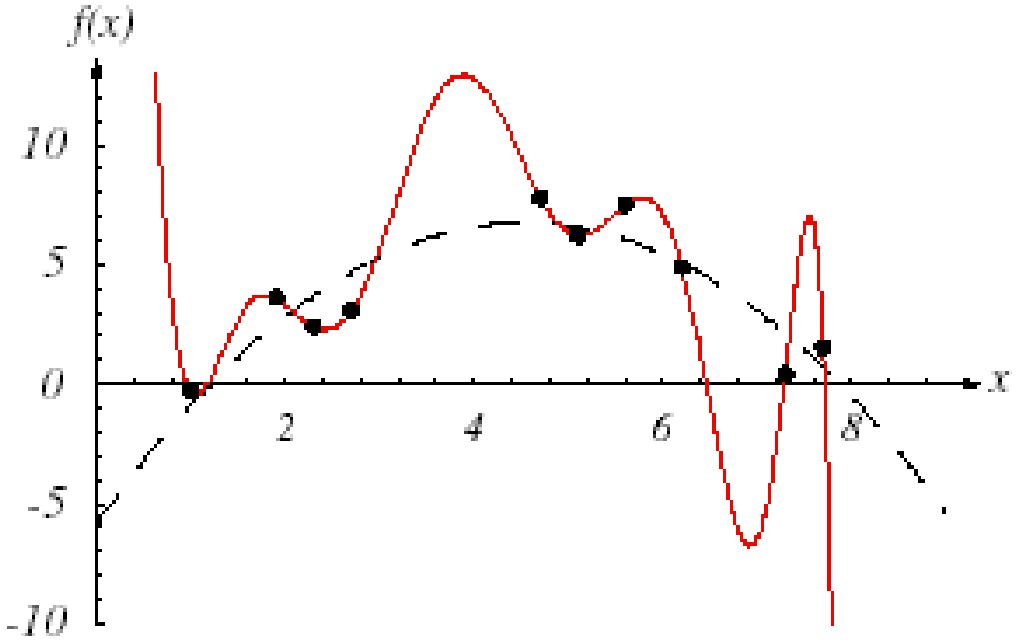
- In pratica, aggiungere feature porta a peggiorare le prestazioni. Questo è dovuto a:
  - modello errato (e.g., ipotesi Gaussiana o ipotesi di probabilità condizionale);
  - numero di campioni insufficiente, e quindi le distribuzioni non sono stimate con accuratezza.



Overfitting

- Quando il numero di campioni è insufficiente allora
  - si cerca di ridurre la dimensionalità delle feature
  - combinare le feature in qualche modo (***feature extraction***)
  - si cerca una stima migliore della matrice di covarianza partendo da una stima nota
  - si soglia la matrice di covarianza o si impone una matrice diagonale
- Facendo questo però si impone l'indipendenza delle feature, ma in realtà queste potrebbero NON esserlo!
- Quindi le prestazioni potrebbero non essere solo sub-ottime e la motivazione si può addurre ancora ai dati insufficienti

- Il problema è analogo a quello del fit di dati (interpolazione vs approssimazione)
- Se si interpola correttamente si perde in capacità di generalizzazione: la parabola è la funzione che approssima meglio i dati



Metodo delle componenti principali (PCA) (trasformata di Hotelling o di Karhunen-Loève)

- Trasformazione lineare originariamente introdotta da Hotelling per decorrelare gli elementi di un vettore casuale
- Karhunen & Loève svilupparono in seguito un'analoga trasformazione per segnali continui
- Viene anche chiamata metodo delle *componenti principali* o degli *autovettori*.
- Data una popolazione di vettori di variabili casuali $\mathbf{x}_i$ reali, i vettori base della trasformata KL sono dati dagli autovettori ortonormalizzati della propria matrice di covarianza (autocorrelazione) $\mathbf{C}$

- Sia data una popolazione di vettori di v.a. (*n* x 1) del tipo

$$\mathbf{x} = \{\mathbf{x}_1^{\mathrm{t}}, \mathbf{x}_2^{\mathrm{t}}, \ldots, \mathbf{x}_{\mathrm{M}}^{\mathrm{t}}\}$$

  Si definisce la media come

$$\mathbf{m}_{\mathbf{x}} = E\{\mathbf{x}\}$$

  dove $E\{.\}$ è l'operatore di Valore Atteso

- La matrice di covarianza della popolazione è definita come

$$\mathbf{C}_{\mathbf{x}} = E\{(\mathbf{x}-\mathbf{m}_{\mathbf{x}})\,(\mathbf{x}-\mathbf{m}_{\mathbf{x}})^{\mathrm{t}}\}$$

- Caratteristiche:
  - Siccome $\mathbf{x}$ è di dimensionalità n, allora $\mathbf{C}_{\mathbf{x}}$ è $n \times n$;
  - gli elementi c_{ii} sono la varianza dell'i-esimo componente dell'insieme di $\mathbf{x}$;
  - c_{ij} sono le relative covarianze tra componenti;
  - $\mathbf{C}_{\mathbf{x}}$ è reale e simmetrica; se x_i e x_j sono scorrelati, allora $c_{ij} = c_{ji} = 0$.
- Se la popolazione è di dimensione finita M, allora

$$\mathbf{m}_{\mathbf{x}} = \frac{1}{M}\sum_{k=1}^{M}\mathbf{x}_k \qquad \mathbf{C}_{\mathbf{x}} = \frac{1}{M}\sum_{k=1}^{M}(\mathbf{x}_k - \mathbf{m}_{\mathbf{x}})(\mathbf{x}_k - \mathbf{m}_{\mathbf{x}})^t = \frac{1}{M}\sum_{k=1}^{M}\mathbf{x}_k\mathbf{x}_k^t - \mathbf{m}_{\mathbf{x}}\mathbf{m}_{\mathbf{x}}^t$$

- Siccome $\mathbf{C_x}$ è reale e simmetrica, si può sempre trovare un insieme di *n* autovettori ortonormali
- Siano $\mathbf{e_i}$ e λ_i, $i=1,2,\ldots,n$ gli autovettori e i relativi autovalori di $\mathbf{C_x}$ ordinati in ordine decrescente, ie.,

  $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n$
- Sia $\mathbf{A}$ una matrice le cui righe sono formate dagli autovettori di $\mathbf{C_x}$ ordinati come sopra, e la si usi per trasformare i vettori $\mathbf{x}$ come segue

$$\mathbf{y} = \mathbf{A}\,(\mathbf{x} - \mathbf{m_x})$$

- La media dei vettori $\mathbf{y}$ è zero e la matrice di covarianza:

$$\mathbf{m_y} = \mathbf{0} \qquad\qquad \mathbf{C_y} = \mathbf{A}\,\mathbf{C_x}\,\mathbf{A}^{\mathrm{T}}$$

- Inoltre, $\mathbf{C_y}$ è una matrice diagonale con gli autovalori di $\mathbf{C_x}$ nella diagonale.

$$\mathbf{C_y} = \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \cdots & 0 & \lambda_n \end{bmatrix}$$

- Gli elementi di $\mathbf{y}$ sono quindi scorrelati
- Siccome le righe di $\mathbf{A}$ sono vettori ortonormali, allora $\mathbf{A}^{-1} = \mathbf{A}^{\mathrm{T}}$ e ogni vettore $\mathbf{x}$ può essere calcolato da $\mathbf{y}$:

$$\mathbf{x} = \mathbf{A}^{\mathrm{T}}\mathbf{y} + \mathbf{m_x}$$

- Si supponga di costruire $\mathbf{A}$ mediante i primi k (più grandi) autovettori.

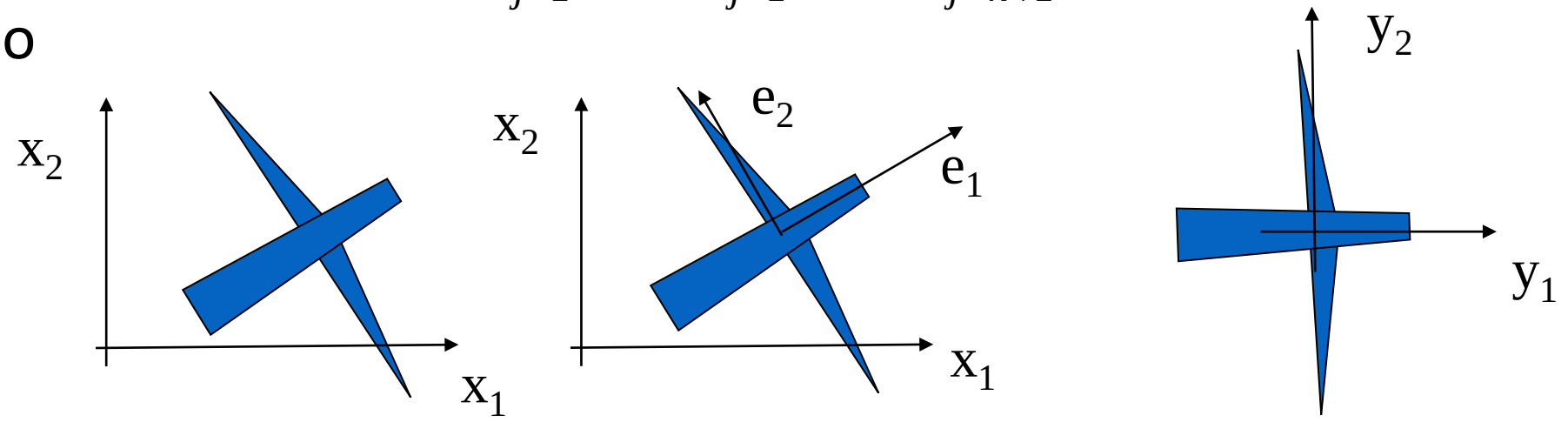
- Allora $\mathbf{A}_k$ è una matrice di dimensione k *x* n e i vettori $\mathbf{y}$ hanno dimensione k.
- I vettori ricostruiti $\mathbf{x}$ non saranno più esatti e sono dati da:

$$\hat{\mathbf{x}} = \mathbf{A}_k^T \mathbf{y} + \mathbf{m}_\mathbf{x}$$

- Si dimostra che la trasformata di Hotelling minimizza l'errore quadratico medio dei vettori $\mathbf{x}$

$$e_{ms} = \sum_{j=1}^{n} \lambda_j - \sum_{j=1}^{k} \lambda_j = \sum_{j=k+1}^{n} \lambda_j$$

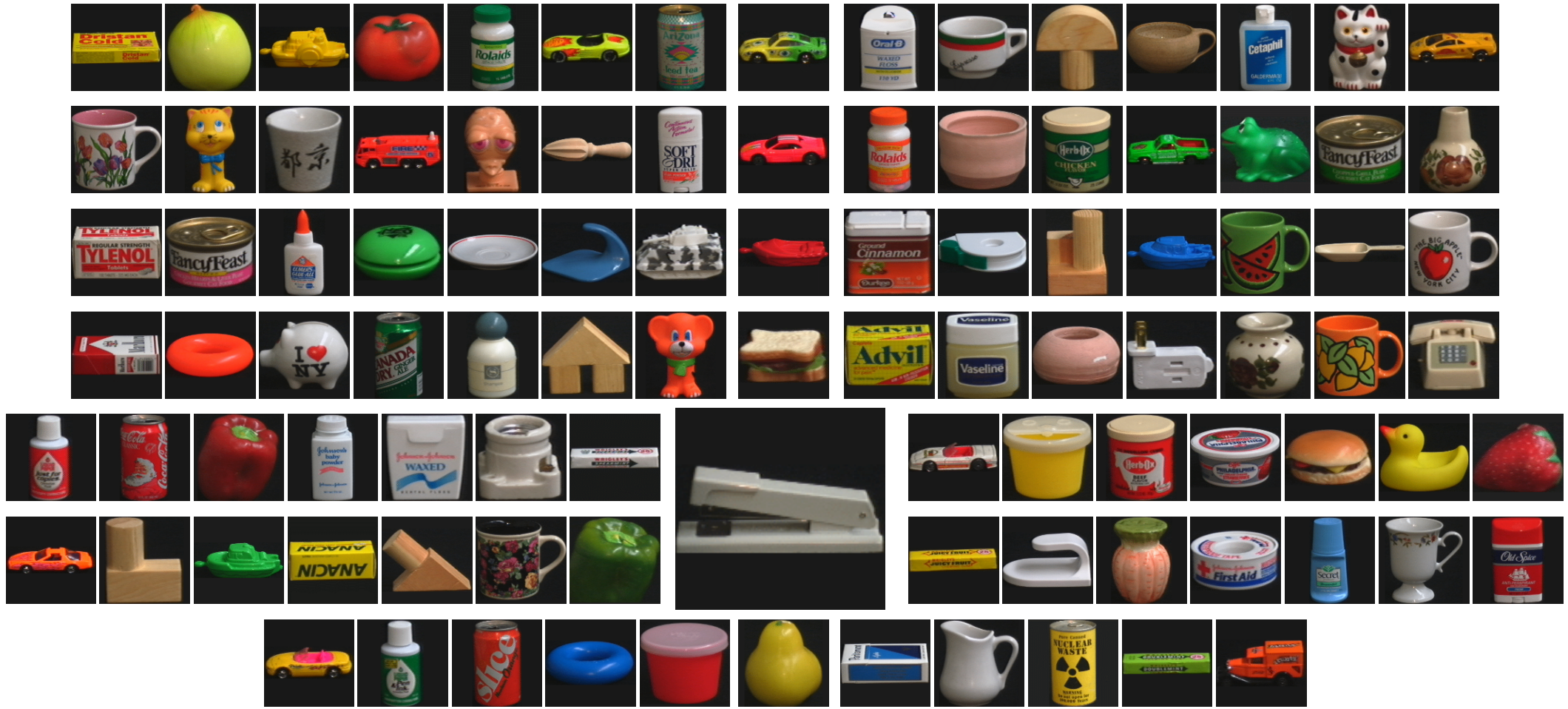
- Esempio



Esempio: Riconoscimento *appearance-based*

- Il riconoscimento è basato sulla vista o apparenza.
- Si utilizzano le img come componenti base dei modelli invece delle feature.
- Ogni oggetto è rappresentato da un insieme di viste, teoricamente prese da tutti i possibili punti di vista in tutte le condizioni di illuminazione
- L'identificazione dell'oggetto significa trovare l'insieme che contiene l'img più simile all'obj da riconoscere.
- Permette di confrontare direttamente i modelli con i dati di ingresso.
- Il database di modelli può diventare troppo grande!

Esempio: Riconoscimento *appearance-based*



Autospazi delle immagini (*image eigenspace*)

- Metodo basato sulle viste
- Ipotesi:
  - ogni img contiene un solo obj
  - gli obj sono acquisiti da una telecamera fissa sotto condizioni di prospettiva debole
  - img normalizzate nelle dimensioni, ie., la dimensione dell'img è il minimo rettangolo che contiene la vista più grande dell'obj
  - l'energia di ogni img è normalizzata a 1: $\sum_{i=1}^{N}\sum_{j=1}^{N} I(i,j)^2 = 1$
  - l'obj è completamente visibile (non occluso)
- Il confronto tra img viene eseguito mediante l'operazione di *correlazione*:

$$c = I_1 \otimes I_2 = \frac{1}{K}\sum_{i=1}^{N}\sum_{j=1}^{N} I_1(i,j) I_2(i,j), \quad K \text{ costante di normalizzazione}$$

- Algoritmo
  - Dati O obj, P punti di vista, L direzioni di illuminazione, si hanno OPL img nel database
  - Si costruisce la matrice di covarianza Q, e si rappresenta ogni img $\mathbf{x}_{pl^o}$ con il relativo vettore dell'autospazio di coordinate $\mathbf{g}_{pl^o}$.
  - Si possono utilizzare solo le componenti associate agli autovalori più grandi per rappresentare le img:

$$\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n, \quad \lambda_i \cong 0 \quad \text{per} \quad i > k$$

$$\mathbf{x}_j \cong \mathbf{x}_m + \sum_{i=1}^{k} g_{ji}\mathbf{e}_i$$

    con $\mathbf{e}_i$ autovettori della matrice Q e g_{ij} coefficienti associate all'img $\mathbf{g}$
  - Se $k << n$ la rappresentazione è notevolmente ridotta.
  - L'insieme delle viste di un obj (variando posa e direzione di illuminazione) fanno variare in maniera continua il punto $\mathbf{g}_{pl^o}$ nell'autospazio, creando una varietà (*manifold*).

- La correlazione tra le img si effettua mediante il calcolo della distanza tra le img nell'autospazio.
- Nelle ipotesi di normalizzazione delle img $\|\mathbf{x}_1\|^2 = \|\mathbf{x}_2\|^2 = 1$, si ha $\|\mathbf{x}_1 - \mathbf{x}_2\|^2 = 2(1 - \|\mathbf{x}_1{}^T\mathbf{x}_2\|)$, ie., massimizzare la correlazione significa minimizzare la distanza:

$$\|\mathbf{x}_1 - \mathbf{x}_2\|^2 = \left\|\sum_{i=1}^{n} g_{1i}\mathbf{e}_i - \sum_{i=1}^{n} g_{2i}\mathbf{e}_i\right\|^2 \cong \left\|\sum_{i=1}^{k} g_{1i}\mathbf{e}_i - \sum_{i=1}^{k} g_{2i}\mathbf{e}_i\right\|^2 =$$

$$= \left\|\sum_{i=1}^{k} (g_{1i} - g_{2i})\mathbf{e}_i\right\|^2 = \sum_{i=1}^{k} (g_{1i} - g_{2i})^2 = \|\mathbf{g}_1 - \mathbf{g}_2\|^2$$

- Il costo computazionale è $O(k)$ invece di $O(n)$.
- Tale trattazione suggerisce come effettuare la fase di identificazione di un oggetto:
  - date le img di un modello, si calcolano i punti nell'autospazio e la relativa curva interpolante

- per identificare un obj da una nuova img $\mathbf{y}$, si proietta $\mathbf{y}$ nell'autospazio (usando gli autovettori della matrice di covarianza di tutte le OPL img nel database), ottenendo un punto $\mathbf{g}_y$;
- si deve cercare la curva $\mathbf{g}^o(\mathbf{p},\mathbf{l})$ ($= \mathbf{g}_{pl}{}^o$) più vicina a $\mathbf{g}_y$.

- Come si stimano $\mathbf{g}_p{}^o$:

$$\mathbf{g}_{pl}^{o} = \left[\mathbf{e}_1\ \mathbf{e}_2\ \cdots \mathbf{e}_k\right]\left(\mathbf{x}_{pl}^{o} - \mathbf{x}_m\right)$$

- Come si stima la proiezione di $\mathbf{y}$ nell'autospazio, $\mathbf{g}_y$:

$$\mathbf{g}_y = \left[\mathbf{e}_1\ \mathbf{e}_2\ \cdots \mathbf{e}_k\right]\left(\mathbf{y} - \mathbf{x}_m\right)$$

- Considerazioni e commenti:
  - trovare il punto di una curva più vicino ad un dato punto non è sempre banale;
  - non è sempre vero che $k << n$;
  - trovare gli autovalori di grosse matrici è computazionalmente costoso;
  - la segmentazione tra obj e sfondo non è sempre una semplice operazione.