



**Università  
degli Studi  
di Palermo**



# Preparazione dei dati

CORSO DI BIG DATA  
a.a. 2022/2023

Prof. Roberto Pirrone

# Sommario

- Estrazione delle feature rilevanti
- Portabilità tra diverse tipologie di dati
- Data cleaning
- Riduzione della dimensionalità

# Processo di data preparation

- Estrazione delle feature e portabilità
  - Si ricercano le feature più significative in relazione al problema da risolvere e, in genere, tali feature vanno *convertite nelle tipologie di dati* più adatte per l'utilizzo con gli algoritmi di analisi
- Data cleaning
  - Eliminazione di dati erronei e/o inconsistenti
  - *Imputazione* dei dati mancanti attraverso un processo di stima
- Data reduction & transformation
  - Riduzione del volume dei dati tramite *campionamento* di un sotto insieme, *riduzione della dimensionalità* dei campioni ovvero *trasformazione* degli stessi secondo una diversa rappresentazione

# Portabilità tra tipologie di dati

Source data type	Destination data type	Methods
Numeric	Categorical	Discretization
Categorical	Numeric	Binarization
Text	Numeric	Latent semantic analysis ( <i>LSA</i> )
Time series	Discrete sequence	<i>SAX</i>
Time series	Numeric multidimensional	<i>DWT, DFT</i>
Discrete sequence	Numeric multidimensional	<i>DWT, DFT</i>
Spatial	Numeric multidimensional	2-d <i>DWT</i>
Graphs	Numeric multidimensional	<i>MDS</i> , spectral
Any type	Graphs	Similarity graph (Restricted applicability)

# Portabilità tra tipologie di dati

- Discretizzazione
  - Si creano degli intervalli discreti per rappresentare il range di variazione del dato numerico
    - Intervalli ad ampiezza costante
    - Intervalli logaritmici: ogni intervallo  $[a,b] \rightarrow \log(b) - \log(a) = \text{costante}$
    - Più in generale  $[a,b] \rightarrow f(b) - f(a) = \text{costante}$  per una certa  $f(.)$  che rappresenta la distribuzione dei dati
    - Intervalli a «profondità» costante: ogni intervallo contiene lo stesso numero di elementi

# Portabilità tra tipologie di dati

- Binarizzazione
  - Ogni categoria possibile induce la creazione di un *one hot vector* i cui componenti sono tutti nulli tranne quello in posizione corrispondente alla categoria

$$a \in \{c_1, c_2, \dots, c_\phi\} \rightarrow \mathbf{x} \in \mathbb{R}^\phi,$$

$$a = c_k \rightarrow \mathbf{x} = [0_1, 0_2, \dots, 1_k, \dots, 0_\phi]$$

# Portabilità tra tipologie di dati

- Latent Semantic Analysis (LSA)
  - Un corpus documentale può essere visto come una matrice binaria sparsa  $\mathcal{D}$  che ha come righe i termini e come colonne i documenti che li contengono.
  - La LSA è una forma di *Singular Value Decomposition* (SVD) di  $\mathcal{D}$  che mira a creare uno spazio euclideo in cui un corpus documentale viene trasformato in un insieme di vettori in  $\mathbb{R}^d$  dove è possibile giudicare la similarità tramite una misura di distanza.
  - Di norma gli elementi di  $\mathcal{D}$  sono pesati con una misura di tipo frequentista chiamata *Term Frequency – Inverse Document Frequency* TF-IDF

# Portabilità tra tipologie di dati

- TF-IDF

- Prodotto della frequenza  $f_{t,d}$  del termine  $t$  nel documento  $d$  appartenente a  $\mathcal{D}$ , avente dimensione  $N$ , per l'inverso della frequenza dei documenti che contengono il termine  $t$  e cioè  $N/n_t$
- Diversi schemi di pesatura:

$$\text{tf}(t, d) = f_{t,d} / \sum_{t' \in d} f_{t',d}$$

$$\text{idf}(t, \mathcal{D}) = \log \left( \frac{N}{n_t} \right)$$

$$\text{tf-idf}(t, d, \mathcal{D}) = \text{tf}(t, d) \times \text{idf}(t, \mathcal{D})$$

Si può mostrare che:

$$\frac{1}{N} \sum_{t,d} \text{tf}(t, d) \times \text{idf}(t, \mathcal{D}) \equiv MI(\mathcal{T}, \mathcal{D})$$



# Portabilità tra tipologie di dati

- Symbolic Aggregate Approximation (SAX)
  - Si effettua la media dei valori della serie temporale su finestre di osservazione contigue di data ampiezza  $w$
  - Si discretizza la serie risultante con intervalli a profondità costante
  - Il risultato è una serie discreta di simboli
- Come ottengo gli intervalli a profondità costante?
  - Si assume una distribuzione gaussiana dei valori delle serie temporali mediate che viene stimata in termini di media e varianza ed i cui *quantili* forniscono gli estremi degli intervalli

# Portabilità tra tipologie di dati

- Discrete Wavelet Transform (DWT) e Discrete Fourier Transform (DFT)
  - Usate per trasformare serie temporali in vettori multidimensionali di coefficienti che sono meno interdipendenti dei dati originali
  - Riduzione di dimensionalità
- La DWT si applica anche alla trasformazione di dati spaziali in dati numerici multidimensionali

# Portabilità tra tipologie di dati

- Discrete Wavelet Transform (DWT) e Discrete Fourier Transform (DFT)
  - Una sequenza discreta di simboli può essere trasformata in un insieme di sequenze binarie che descrivono la presenza di un solo simbolo per volta e poi questi ultimi sono trasformati con la DWT

```
ACACACTGTGACTG
10101000001000
01010100000100
00000010100010
00000001010001
```

# Portabilità tra tipologie di dati

- DWT (Haar Wavelets)
  - Si assuma una sequenza temporale  $(t_i; x_i)$  avente lunghezza  $q$  che è potenza di 2 e suddivisa ricorsivamente in due metà  $S_k^i$  ( $i$ -esimo segmento a profondità di suddivisione  $k$ ) fino ad arrivare a segmenti di lunghezza unitaria
    - $k = 1, \dots, \log_2(q) + 1$
    - $S_k^i = [(i - 1) \cdot q / 2^{k-1} + 1, i \cdot q / 2^{k-1}]$

# Portabilità tra tipologie di dati

- DWT (Haar Wavelets)
  - Il coefficiente  $i$ -esimo della DWT è, per ogni livello di decomposizione  $k$ , la semi-differenza tra i valori medi di due sotto-sequenze adiacenti  $S_{k+1}^{2i}$  e  $S_{k+1}^{2i-1}$  al livello di suddivisione  $k+1$

$$\begin{aligned}\psi_k^i &= \left( \Phi_{k+1}^{2 \cdot i - 1} - \Phi_{k+1}^{2 \cdot i} \right) / 2 \\ \Phi_k^i &= \left( \Phi_{k+1}^{2 \cdot i - 1} + \Phi_{k+1}^{2 \cdot i} \right) / 2 \\ \Phi_{\log_2(q)+1}^i &\equiv x^i\end{aligned}$$

# Portabilità tra tipologie di dati

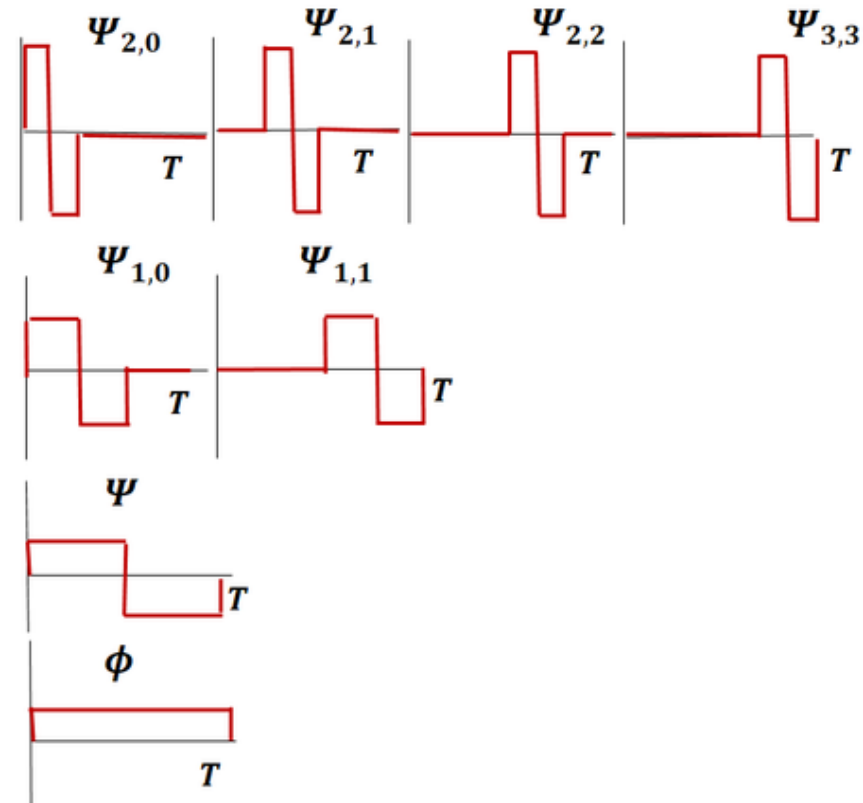
- DWT (Haar Wavelets)
  - Lo schema di calcolo è ricorsivo e parte calcolando le medie tra coppie di elementi successivi nella sequenza e poi andando a riaggregare le medie ad ogni livello fino a  $k = 1$ .

$$\begin{aligned}\psi_k^i &= (\Phi_{k+1}^{2 \cdot i - 1} - \Phi_{k+1}^{2 \cdot i}) / 2 \\ \Phi_k^i &= (\Phi_{k+1}^{2 \cdot i - 1} + \Phi_{k+1}^{2 \cdot i}) / 2 \\ \Phi_{\log_2(q)+1}^i &\equiv x^i\end{aligned}$$

# Portabilità tra tipologie di dati

- DWT (Haar Wavelets)

$$\begin{pmatrix} 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 \\ 1 & 1 & -1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & -1 & -1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}$$



*Queste funzioni di base vanno bene per una sequenza di otto elementi*

# Portabilità tra tipologie di dati

- DWT (Haar Wavelets)  $\psi = \left[ \psi_{\log_2(q)+1}^1, \dots, \psi_{\log_2(q)+1}^{q/2}, \dots, \psi_1^1, \phi_1^1 \right]$ 
  - Il vettore  $\psi = \text{DWT}(\mathbf{x})$  e ha lunghezza  $q$
  - La ricostruzione avviene attraverso la matrice dei vettori di base wavelet  $W$

$$W = \begin{bmatrix} \overline{W_1} \\ \vdots \\ \overline{W_q} \end{bmatrix}$$

$$\mathbf{x} = \sum_{i=1}^q \psi_i \cdot \overline{W_i}$$



# Portabilità tra tipologie di dati

- Multidimensional Scaling (MDS)

- Funziona per grafi pesati in cui il peso  $\delta_{ij}$  tra due nodi abbia un significato di distanza o similarità tra essi.
- Si assuma di voler rappresentare ogni nodo  $i = 1, \dots, n$  con un embedding k-dimensionale  $X_i \in \mathbb{R}^k$

- Assumendo di conoscere tutte le  $\delta_{ij} = \delta_{ji}$  che sono  $\binom{n}{2}$  si minimizzi il seguente funzionale

$$O = \sum_{i,j:i < j} (\|\overline{X_i} - \overline{X_j}\| - \delta_{ij})^2$$

# Portabilità tra tipologie di dati

- Generazione di grafi di similarità
  - Può essere conveniente cercare similarità a coppie per determinate applicazioni e alcuni algoritmi di analisi se ne avvantaggiano
  - Ogni oggetto  $O_i$  del data set è considerato un nodo multidimensionale
  - Se  $d(O_i, O_j)$  è minore di una certa soglia  $\epsilon$  si genera un arco  $(i, j)$  che viene pesato con un peso  $w_{ij}$  generato attraverso l'applicazione di un opportuno kernel ed esprime *similarità*

$$w_{ij} = e^{-\frac{d(O_i, O_j)^2}{t^2}} \quad \text{Heat kernel}$$

# Data cleaning

- Gestione dei dati mancanti
  - Eliminazione dei record incompleti
  - Stima o imputazione dei dati mancanti
    - Può inficiare le prestazioni dell'algoritmo di analisi
  - Uso di algoritmi di analisi robusti rispetto ai dati mancanti
    - Classificazione come metodo di imputazione della feature mancante come «special feature» del dato rispetto alla quale appunto si classifica
    - Utility matrix imputation nei recommender systems

# Data cleaning

- Gestione dei dati mancanti

	GLADIATOR	GODFATHER	BEN-HUR	GOODFELLAS	SCARFACE	SPARTACUS
U <sub>1</sub>	1			5		2
U <sub>2</sub>		5			4	
U <sub>3</sub>	5	3		1		
U <sub>4</sub>			3			4
U <sub>5</sub>				3	5	
U <sub>6</sub>	5		4			

(a) Ratings-based utility

	GLADIATOR	GODFATHER	BEN-HUR	GOODFELLAS	SCARFACE	SPARTACUS
U <sub>1</sub>	1			1		1
U <sub>2</sub>		1			1	
U <sub>3</sub>	1	1		1		
U <sub>4</sub>			1			1
U <sub>5</sub>				1	1	
U <sub>6</sub>	1		1			

(b) Positive-preference utility

## Collaborative filtering

Tecniche di stima dei dati mancanti (preferenze di un utente ovvero item simili per preferenze degli utenti) attraverso il calcolo di *funzioni di similarità* tra le righe o le colonne della utility matrix

I  $k$  utenti/item più simili vengono usati per imputare i valori

# Data cleaning

- Eliminazione dei dati inconsistenti o erronei
  - Analisi di dati inconsistenti tra flussi in arrivo da sorgenti diverse
    - Es. «John Fitzgerald Kennedy» vs. «JFK»
  - Uso della conoscenza di dominio per eliminare le inconsistenze
  - Analisi statistica dei dati per individuare il trend ovvero esplicito utilizzo di algoritmi di outlier detection

# Data cleaning

- Scalatura e normalizzazione
  - Scalatura min-max

$$y_i^j = \frac{x_i^j - \min_j}{\max_j - \min_j}$$

- Z-scaling

$$z_i^j = \frac{x_i^j - \mu_j}{\sigma_j}$$

# Riduzione della dimensionalità dei dati

- Campionamento
  - Campionamento statistico per ridurre la dimensionalità di un data set
  - Campionamento di sequenze
  - Campionamento «ottimo» da una distribuzione per la stima di una certa quantità desiderata
- Selezione di sottoinsiemi di feature rilevanti
- Riduzione della dimensionalità per *rotazione degli assi*
- Riduzione della dimensionalità per trasformazione della tipologia dei dati

# Riduzione della dimensionalità dei dati

- Campionamento
  - Il campionamento può essere «stratificato» nel senso che si preferisce ricoprire il dominio della variabile con una serie di intervalli o *strati* e garantire la presenza di campioni in ogni strato
  - L'altra tecnica di campionamento è quella «per importanza» nella quale alcune parti dei dati sono certamente più rilevanti di altre e quindi esiste una distribuzione  $p(x)$  che descrive la probabilità di campionare un certo valore  $x$ .
  - La probabilità  $p(x)$  ottima per eccellenza è la stessa distribuzione di probabilità dei dati (anche se non la conosciamo)



# Riduzione della dimensionalità dei dati

- Campionamento
  - Le sequenze sono campionate attraverso la creazione di un «magazzino» di campioni di dimensione  $k$
  - Il campione  $n$ -esimo viene inserito nel magazzino con probabilità  $p(x_n)=k/n$  (si ricordi che  $n$  cresce via via che arrivano nuovi campioni)
  - Si elimina casualmente uno dei vecchi dati per far posto al nuovo campione
  - Si può dimostrare che la probabilità di essere inseriti nel magazzino dopo che sono arrivati  $n$  campioni è sempre  $k/n$

# Riduzione della dimensionalità dei dati

- Campionamento
  - Vedremo successivamente che la dimensionalità di un data set  $\mathcal{D}$ , già curato, sul quale addestriamo un algoritmo di apprendimento, può essere variata per *sottocampionamento* o anche per *sovracampionamento*
  - Queste tecniche si applicano ai classificatori in cui gli esempi appartenenti alle varie classi sono fortemente *sbilanciati* ovvero una o più classi sono fortemente più numerose delle altre
  - Sottocampionamento → riduco i campioni delle classi più popolate
  - Sovracampionamento → accresco «artificialmente» i campioni delle classi meno popolate

# Riduzione della dimensionalità dei dati

- Selezione di sottoinsiemi di feature rilevanti
  - tramite approccio non supervisionato → clustering
  - tramite approccio supervisionato → classificazione
- Riduzione della dimensionalità per rotazione degli assi
  - Dati altamente correlati si aggregano lungo *poche direzioni preferenziali* che possono diventare gli *assi di un apposito sistema di riferimento* ottenuto per rotazione

# Riduzione della dimensionalità dei dati

- Riduzione della dimensionalità per rotazione degli assi - PCA
  - La matrice di covarianza di una data set  $D$  di  $n$  record aventi dimensione  $d$  è

$$C = \frac{D^T D}{n} - \bar{\mu}^T \bar{\mu}$$

- I suoi elementi si possono esprimere come

$$c_{ij} = \frac{\sum_{k=1}^n x_k^i x_k^j}{n} - \mu_i \mu_j \quad \forall i, j \in \{1 \dots d\}$$

# Riduzione della dimensionalità dei dati

- Riduzione della dimensionalità per rotazione degli assi - PCA
  - Per qualunque vettore  $v \in \mathbb{R}^d$  è possibile calcolare la varianza unidimensionale della proiezione dei dati su  $v$ ,  $Dv$  come:

$$\bar{v}^T C \bar{v} = \frac{(D\bar{v})^T D\bar{v}}{n} - (\bar{\mu}\bar{v})^2$$

- Cerchiamo la base ortonormale di vettori su cui proiettare i dati per *massimizzare* la loro varianza

# Riduzione della dimensionalità dei dati

- Riduzione della dimensionalità per rotazione degli assi - PCA
  - Il problema di ottimizzazione può essere affrontato con i moltiplicatori di Lagrange

$$\bar{v}^T C \bar{v} + \lambda (1 - \|\bar{v}\|^2)$$

- Se poniamo il gradiente del funzionale pari a 0, otteniamo che la base di vettori è costituita dagli autovettori di C e le varianze sono i corrispondenti autovalori

$$C \bar{v} - \lambda \bar{v} = 0 \qquad \bar{v}^T C \bar{v} = \bar{v}^T \lambda \bar{v} = \lambda$$

# Riduzione della dimensionalità dei dati

- Riduzione della dimensionalità per rotazione degli assi - PCA
  - $C$  è semidefinita positiva cioè  $\mathbf{v}^T C \mathbf{v} \geq 0 \quad \forall \mathbf{v} \neq \mathbf{0}$  e può essere diagonalizzata

$$C = P \Lambda P^T$$

- $P$  è la matrice degli autovettori di  $C$ ,  $P^T P = I$ , mentre  $\Lambda$  è la matrice diagonale degli autovalori di  $C$
- Possiamo riordinare le righe di  $P$  in senso decrescente dal massimo al minimo autovalore (varianza dei dati lungo il corrispondente autovettore) e considerare le  $k$  **componenti principali** cioè quelle per cui la **varianza totale** (cioè la somma di tutte le varianze per le componenti selezionate) ha magnitudine superiore ad una certa soglia

# Riduzione della dimensionalità dei dati

- Riduzione della dimensionalità per rotazione degli assi - PCA
  - Si può mostrare che i dati trasformati  $D' = DP$ , con media  $\bar{\mu}P$ , hanno una matrice di covarianza  $\Lambda$  :

$$\frac{(DP)^T DP}{n} - (\bar{\mu}P)^T (\bar{\mu}P) =$$

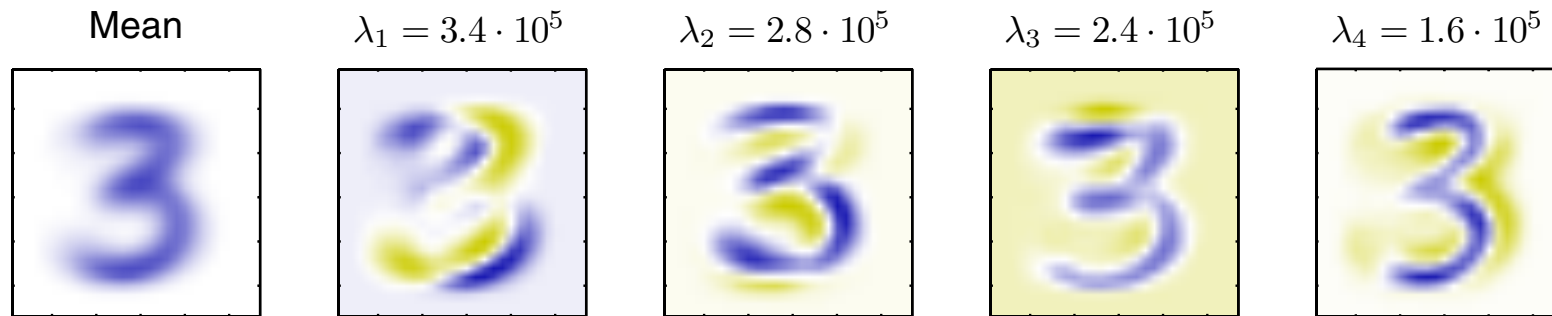
$$\frac{P^T D^T DP}{n} - P^T \bar{\mu}^T \bar{\mu} P =$$

$$P^T CP \qquad P^T CP = P^T P \Lambda P^T P = I \Lambda I = \Lambda$$

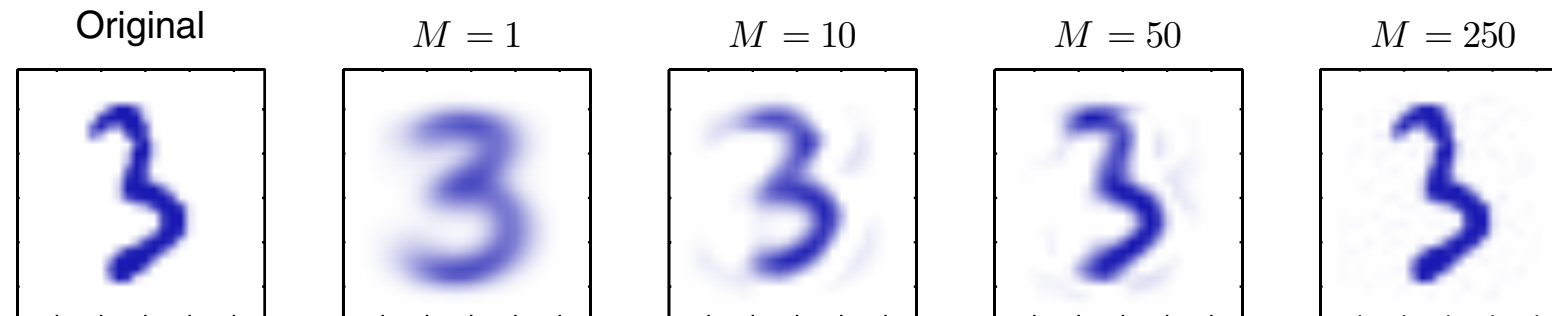


# Riduzione della dimensionalità dei dati

- Riduzione della dimensionalità per rotazione degli assi - PCA



Ricostruzione del valor medio della cifra «3» per i primi 4 autovalori



Ricostruzione di un campione al variare delle componenti principali; dimensionalità originale: 784 (28 x 28)

# Riduzione della dimensionalità dei dati

- Riduzione della dimensionalità per rotazione degli assi – SVD
  - È una fattorizzazione della matrice dei dati di dimensione  $n \times d$

$$D = Q\Sigma P^T$$

- $Q$  ha dimensione  $n \times n$  e le sue colonne formano una base ortonormale denominata *left singular vectors*  $\rightarrow Q^T Q = I$
- $\Sigma$  è una matrice diagonale di dimensione  $n \times d$  e contiene i *singular values*, non negativi; gli elementi delle righe oltre  $n$  sono tutti nulli. I singular values sono le «variabili latenti» di questa rappresentazione dei dati
- $P$  ha dimensione  $d \times d$  e le sue colonne formano una base ortonormale denominata *right singular vectors*  $\rightarrow P^T P = I$

# Riduzione della dimensionalità dei dati

- Riduzione della dimensionalità per rotazione degli assi – SVD

$$DD^T = Q\Sigma(P^T P)\Sigma^T Q^T = Q(\Sigma\Sigma^T)Q^T$$

$Q \rightarrow$  autovettori di  $DD^T$   
 $\Sigma\Sigma^T \rightarrow$  autovalori di  $DD^T$

$$D^T D = P\Sigma^T(Q^T Q)\Sigma P^T = P(\Sigma^T \Sigma)P^T$$

$P \rightarrow$  autovettori di  $D^T D$   
 $\Sigma^T \Sigma \rightarrow$  autovalori di  $D^T D$

$$\mu = \mathbf{0} \Rightarrow C = \frac{D^T D}{n}$$

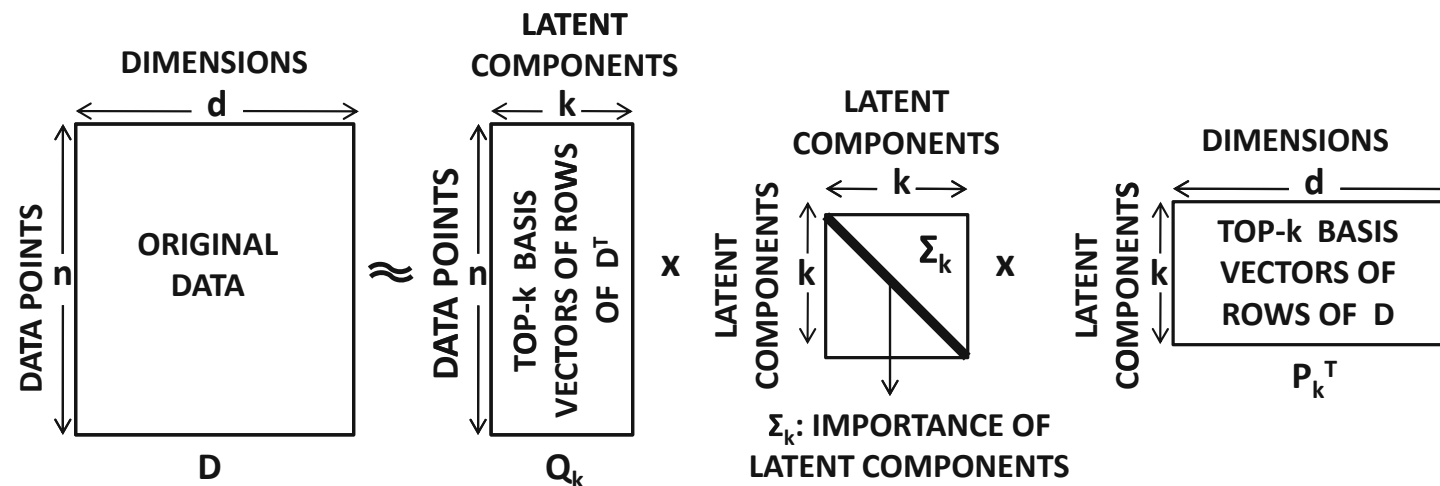
$P \rightarrow$  autovettori di  $C$   
 $\Sigma^T \Sigma \rightarrow$  autovalori di  $C$  moltiplicati per  $n$

SVD e PCA sono la stessa rotazione degli assi per dati centrati rispetto alla propria media

# Riduzione della dimensionalità dei dati

- Riduzione della dimensionalità per rotazione degli assi – SVD
  - La SVD troncata rappresenta il data set con i primi  $k$  valori singolari

$$D \approx Q_k \Sigma_k P_k^T$$



- $k \ll n, d$
- Le  $d - k$  colonne restanti dei dati ricostruiti  $D' = DP$  sono effettivamente nulle

# Riduzione della dimensionalità dei dati

- Riduzione della dimensionalità per rotazione degli assi – SVD
  - La SVD troncata rappresenta il data set con i primi  $k$  valori singolari

$$D \approx Q_k \Sigma_k P_k^T$$

- Richiamando le considerazioni sulla PCA che massimizza la varianza (somma dei quadrati delle distanze euclidee dalla media) la *SVD massimizza l'energia associata ai dati* in termini di somma dei quadrati delle distanze dall'origine che si ottiene da  $D^T D$
- Tale energia, nel caso troncato, è data dalla *somma dei quadrati dei primi  $k$  valori singolari*; dato un right singular vector  $v$  e il corrispondente valore singolare  $\sigma$ :

$$(D\bar{v})^T (D\bar{v}) = \bar{v}^T (D^T D) \bar{v} \equiv \bar{v}^T \bar{v} \sigma^2 \bar{v}^T \bar{v} = \sigma^2$$

# Riduzione della dimensionalità dei dati

- Riduzione della dimensionalità per rotazione degli assi – LSA
  - SVD troncata applicata ad una matrice  $D$  di valori TF-IDF relativi all'occorrenza di  $n$  termini globalmente in  $d$  documenti
  - $D$  è molto sparsa e di elevatissime dimensioni
  - Non viene fatta la normalizzazione rispetto alla media perché la sparsità garantisce una matrice di covarianza comunque approssimativamente proporzionale a  $D^T D$

# Riduzione della dimensionalità dei dati

- Utilizzo della Haar Wavelet Transform
  - La ricostruzione di una serie temporale si può riscrivere

$$T = \sum_{i=1}^q a_i \overline{W}_i = \sum_{i=1}^q \left( a_i ||\overline{W}_i|| \right) \frac{\overline{W}_i}{||\overline{W}_i||}$$

- I vettori di base normalizzati sono ortonormali
- Si può dimostrare che troncando la HWT con i primi  $k$  coefficienti normalizzati si minimizza l'errore di ricostruzione

# Riduzione della dimensionalità dei dati

- Multi Dimensional Scaling (MDS)

$$O = \sum_{i,j:i < j} \left( \|\overline{X_i} - \overline{X_j}\| - \delta_{ij} \right)^2$$

- Piuttosto che effettuare una minimizzazione si affronta il problema come decomposizione SVD
- Nota la matrice  $\Delta = [\delta_{ij}^2]_{n \times n}$ , cerchiamo la matrice  $D = [X_i]_{n \times k}$  dei dati che abbia  $\Delta$  come insieme delle distanze e tale che  $k \ll n$
- $X_i \rightarrow \text{embedding}$  k-dimensionale del nodo  $i$



# Riduzione della dimensionalità dei dati

- Multi Dimensional Scaling (MDS)
  - Dalla legge del coseno (  $a^2 = b^2 + c^2 - 2bc \cos(\alpha)$  ) estesa ai vettori n-dimensionali possiamo ricavare la matrice dei prodotti scalari  $S = [S_{ij} \triangleq X_i \cdot X_j]$  a partire da  $\Delta$

$$\overline{X_i} \cdot \overline{X_j} = -\frac{1}{2} \left[ \|\overline{X_i} - \overline{X_j}\|^2 - \left( \|\overline{X_i}\|^2 + \|\overline{X_j}\|^2 \right) \right] \quad \forall i, j \in \{1 \dots n\}$$

$$S = -\frac{1}{2} (I - U/n) \Delta (I - U/n) \equiv DD^T$$

- $U$  è la matrice di tutti 1 di ordine  $n$
- I vettori  $X_i$  si intendono a media nulla

# Riduzione della dimensionalità dei dati

- Multi Dimensional Scaling (MDS)
  - La SVD troncata di  $S \approx Q_k \Sigma_k^2 Q_k^T$  fornisce una sua fattorizzazione per ricavare  $D$

$$S \approx Q_k \Sigma_k^2 Q_k^T = (Q_k \Sigma_k) (Q_k \Sigma_k)^T$$

$$D = Q_k \Sigma_k$$

# Riduzione della dimensionalità dei dati

- Trasformazione spettrale per embedding di grafi
  - Un grafo non orientato  $G=(N,A)$  è caratterizzato da una matrice di pesi  $W=[w_{ij}]$  che esprimono *similarità* (non distanze) tra i nodi  $\rightarrow$  etichette degli archi
  - Ricerchiamo l'embedding di nodi  $D = [X_i]_{n \times k}$  che minimizzi

$$O = \sum_{i=1}^n \sum_{j=1}^n w_{ij} \left\| \bar{X}_i - \bar{X}_j \right\|^2$$

- Useremo la matrice laplaciana  $L$  di  $G$

# Riduzione della dimensionalità dei dati

- Trasformazione spettrale per embedding di grafi
  - Vediamo il caso monodimensionale cioè embedding scalare per ogni nodo

$$O = \sum_{i=1}^n \sum_{j=1}^n w_{ij} (y_i - y_j)^2$$

*Matrice Laplaciana:  
Grado – Adiacenza*

$$L = \Lambda - W, \quad \Lambda_{ii} = \sum_{j=1}^n w_{ij}$$

$$O = 2\bar{y}^T L \bar{y}, \quad \bar{y}^T \Lambda \bar{y} = 1, \quad \bar{y} = (y_1, \dots, y_n)^T$$

$$\Lambda^{-1} L \bar{y} = \lambda \bar{y}$$

Scalatura per evitare la soluzione banale  $\mathbf{y} = \mathbf{0}$

- La soluzione è data dai k più piccoli autovettori della matrice  $\Lambda^{-1}L$