Singular Value Decomposition e Principal Component Analysis

Una breve rassegna sulle tecniche di riduzione dimensionale

Giorgio Di Fusco

Università degli Studi di Napoli Federico II

15 maggio 2025

Argomenti

Riduzione della dimensionalità e trasformate

- Singular Value Decomposition
- Principal component analysis

Esempi applicativi

- Dati gaussiani affetti da rumore
- Il dataset `ovariancancer`

La maledizione della dimensionalità

La **maledizione della dimensionalità** è un fenomeno che si verifica quando si lavora con dati ad alta dimensione.

In molti domini, i sistemi complessi generano dati che sono organizzati in matrici di dimensione molto grandi.

La dimensionalità diventa quindi una sfida ricorrente nell'elaborazione dei dati.

SVD I

La **singular value decomposition** è una delle fattorizzazioni matriciali più importanti dell'era computazionale.

Fornisce un *algoritmo numericamente stabile* per decomporre una matrice A.

Definizione

Un algoritmo si dice numericamente stabile quando gli errori, una volta che sono stati introdotti, non si accumulano e amplificano durante il calcolo, mantenendo la precisione del risultato finale.

SVD II

SVD fornisce un metodo sistematico per determinare un'approssimazione a bassa dimensione di un generico dataset.

Questa tecnica è **data driven**: i modelli vengono costruiti esclusivamente a partire dai dati stessi, senza alcuna assunzione a priori.

Definizione della SVD I

Generalmente, siamo interessati nell'analisi di un dataset molto grande $\mathbf{X} \in \mathbb{C}^{n \times m}$:

$$\mathbf{X} = \begin{bmatrix} \vdots & \vdots & \dots & \vdots \\ x_1 & x_2 & \dots & x_m \\ \vdots & \vdots & \dots & \vdots \end{bmatrix}$$

Dove le colonne $x_k \in \mathbb{C}^n$ possono essere delle misure ottenute da qualche esperimento.

Definizione della SVD II

Definizione

La SVD è una decomposizione matriciale unica che esiste per ogni matrice complessa $\mathbf{X} \in \mathbb{C}^{n \times m}$*:*

$$X = U \Sigma V^{\dagger} \tag{1}$$

dove $U \in \mathbb{C}^{n \times n}$ *e* $V \in \mathbb{C}^{m \times m}$ *sono matrici unitarie con colonne ortonormali e* $\Sigma \in \mathbb{R}^{n \times m}$ *è una matrice reale, diagonale, con elementi non-negativi.*

$V^{\dagger}$ denota la matrice trasposta coniugata di V:

$$V^{\dagger} = (V^{T})^{*}$$

Ossia, la matrice ottenuta effettuando la trasposta e scambiando ogni valore con il suo complesso coniugato.

Calcolo della SVD

L'SVD è una pietra miliare della scienza e dell'ingegneria computazionale e la sua implementazione numerica è importante e matematicamente interessante.

La maggior parte delle implementazioni numeriche standard sono mature ed esiste un'interfaccia semplice in molti linguaggi informatici moderni, che ci permette di astrarre i dettagli alla base del calcolo della SVD.

Eseguire SVD in Matlab

```
1  >>X = randn(5,3);          % Creo una matrice 5x3 random
2  >>[U,S,V] = svd(X);           % Funzione svd
```

Mediante la funzione

`svd`

siamo in grado di ottenere la fattorizzazione di X come prodotto di U,S e V.

Interpretazione geometrica

- Le colonne della matrice U forniscono una base ortonormale per lo spazio delle colonne di X (spazio immagine).
- Le colonne di V forniscono una base ortonormale per lo spazio delle righe di X.
- Se X rappresenta misure spaziali nel tempo:
  - U codifica i pattern spaziali.
  - V codifica i pattern temporali.

Ciò che rende particolarmente utile la SVD è il fatto che sia U che V sono matrici unitarie, cioè: $UU^* = U^*U = I_{n\times n}$ e $VV^* = V^*V = I_{m\times m}$. Questo significa che risolvere un sistema di equazioni che coinvolgono U o V è computazionalmente efficiente.

PCA

La Principal component analysis è uno degli utilizzi principali della SVD, in quanto fornisce un sistema di riferimento rispetto al quale è possibile quantificar la significatibilità dei dati attraverso la **varianza**.

Questa tecnica procede a rilevare correlazioni, ridondanze e rumore attraverso la **matrice di covarianza**, che rappresenta la misura della dipendenza lineare tra le variabili di un dataset.

La matrice di covarianza

Definizione

Data una matrice dei dati $X \in \mathbb{R}^{m \times n}$, *dove:*

- m *è il numero di osservazioni (campioni)*
- n *è il numero di variabili (features)*

la ***matrice di covarianza*** *si presenta nella forma:*

$$C = C_{ij} \tag{2}$$

dove C_{ii} *rappresenta la varianza della variabile* i, *mentre* $C_{i,j} = Cov(X_i, X_j)$ *rappresenta la covarianza tra le variabili* i *e* j*:*

$$C = \begin{bmatrix} \sigma^2(X_1) & Cov(X_1, X_2) & \dots \\ Cov(X_1, X_2) & \sigma^2(X_2) & \dots \\ \vdots & \vdots & \ddots \end{bmatrix}$$

Dalla diagonalizzazione classica alla SVD

Mediante la **fattorizzazione** della matrice di covarianza siamo in grado di trovare delle direzioni principali di variazione (detti anche **componenti principali**).

La diagonalizzazione classica richiede però che la matrice sia simmetrica e diagonalizzabile. Numericamente, matrici vicine alla singolarità o con autovalori molto piccoli creano problemi. Inoltre sorge il problema delle matrici rettangolari che non possono essere diagonalizzate nel senso classico.

Computazione della PCA

Supponiamo di aver condotto diversi esperimenti e ogni vettore di misurazione viene disposto come riga in una matrice X.

Calcoliamo il vettore delle medie (la media di tutte le righe) e lo sottraiamo ad X. Il vettore media $\overline{x}$ di ciascun esperimento è ottenuto come:

$$\overline{x_j} = \frac{1}{n} \sum_{i=1}^{n} X_{ij}$$

Otteniamo quindi la matrice delle medie:

$$\overline{X} = \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix} \overline{x}$$

Sottraendo $\overline{X}$ da X otteniamo la matrice B:

$$B = X - \overline{X}$$

La normalizzazione della matrice di covarianza

La matrice di covarianza normalizzata delle righe di B è data da:

$$C = \frac{1}{n} B^* B$$

Questo procedimento prende il nome di **normalizzazione**. La normalizzazione garantisce che i valori ottenuti rappresentino una stima imparziale della varianza e della covarianza tra le variabili, indipendentemente dal numero di osservazioni disponibili.

La matrice di covarianza normalizzata sarà poi il punto di partenza per la computazione della **PCA**, attraverso la sua fattorizzazione, per individuare le direzioni principali di variazione dei dati.

Il comando pca

In Matlab, ci sono comandi addizionali come `pca` e `princomp` per eseguire la PCA:

```
>>[V,score,s2]= pca(X);
```

- La matrice V è equivalente alla matrice V ottenuta dalla funzione `svd`;
- Il vettore $s2$ contiene gli autovalori della matrice di covarianza di X, anche noti come **varianze delle componenti principali**;
- La variabile *score* contiene le coordinate di ogni riga di B nelle direzioni delle componenti principali.

Dati gaussiani rumorosi I

Si consideri la nuvola di dati affetta da rumore della Figura 1.

La nuvola è stata generata a partire da un punto centrale `xC` e delle componenti `sig` note a priori.

Successivamente abbiamo ruotato la nuvola di un angolo pari a $\pi/3$.

Obiettivo dell'esperimento è applicare PCA alla nuvola come se non conoscessimo le sue componenti principali.

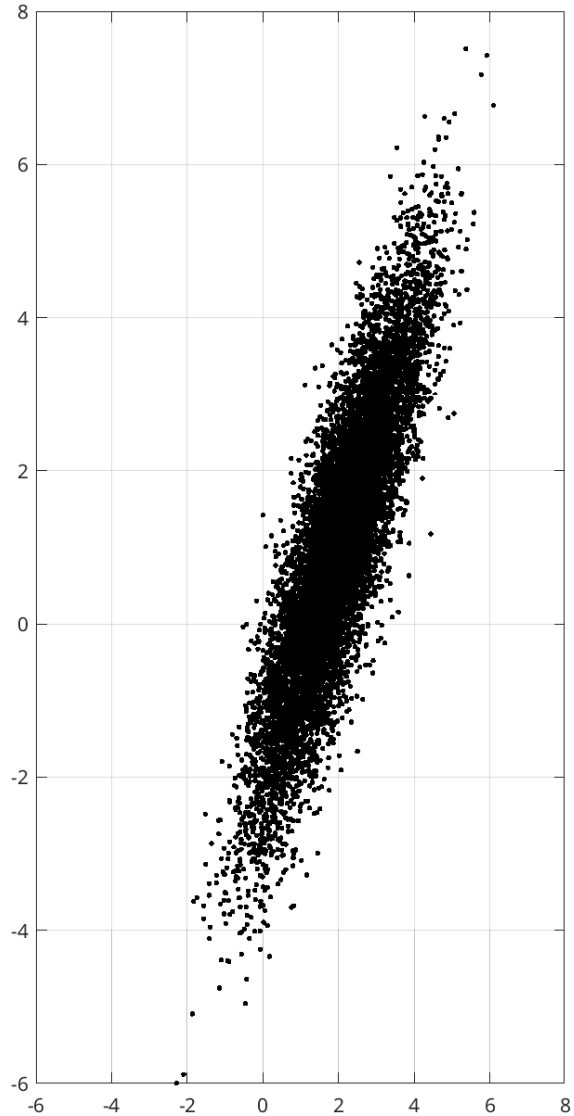


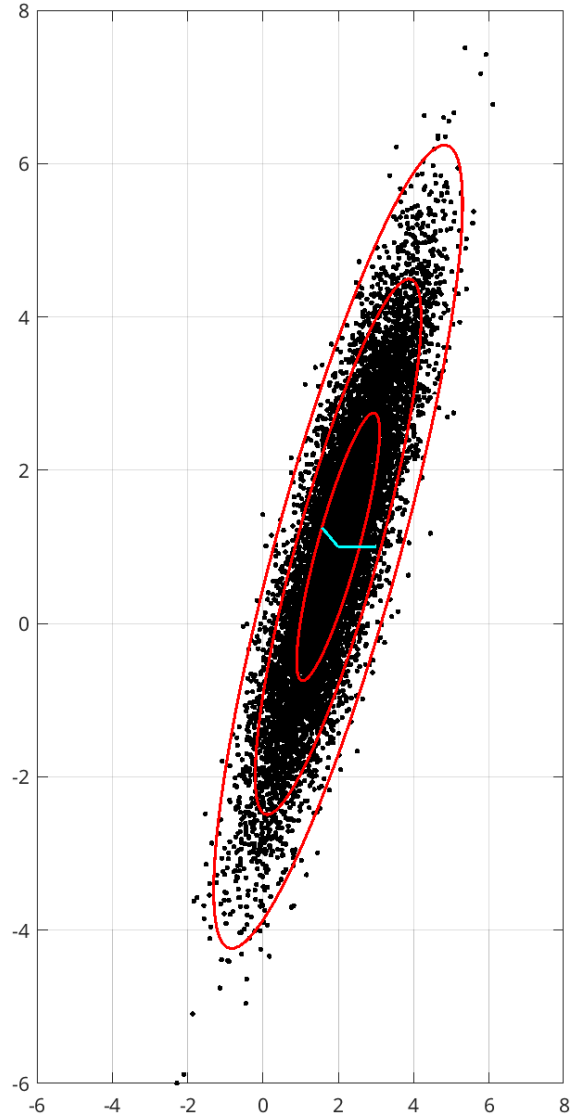
Figura: Nuvola di dati

Dati gaussiani rumorosi II

Attraverso lo script Matlab `pca_random.m` si utilizza la PCA per ottenere le matrici `U`, `S` e `V`.

I valori singolari sono quasi simili alla varianza della nuvola come mostrato in tabella:

	σ_1	σ_2
Data	2	0.5
SVD	1.974	0.503

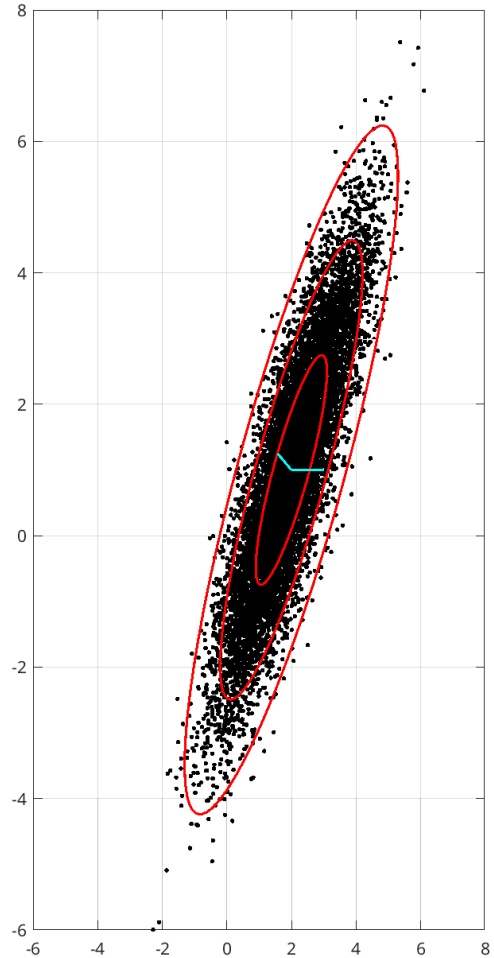


Dati gaussiani rumorosi III

Anche la matrice U ottenuta mediante SVD si avvicina di molto alla matrice di rotazione $R_{\frac{\pi}{3}}$:

$$R_{\frac{\pi}{3}} = \begin{bmatrix} 0.5 & -0.8660 \\ 0.8660 & 0.5 \end{bmatrix},$$

$$U = \begin{bmatrix} 0.4998 & -0.8662 \\ -0.8662 & 0.4998 \end{bmatrix}$$



```
clear all, close all,  clc

% Centro dei dati (media)
xC = [2;1;];
% Assi principali
sig = [2; .5];

% Rotazione della nuvola di pi/3
theta = pi/3;
% Costruzione della matrice di rotazione
R = [cos(theta) -sin(theta);
    sin(theta) cos(theta)];

% Creazione di 10000 punti
nPoints = 10000;
X = R * diag(sig)*randn(2,nPoints) + diag(xC)*ones(2,nPoints);

% Plot della nuvola di dati
subplot(1,2,1)
scatter(X(1,:),X(2,:),'k.','LineWidth',2)
hold on, box on, grid on
axis([-6 8 -6 8])

% Calcolo media
Xavg = mean(X,2);
% Calcolo dati sottratti della media
```

```
B = X - Xavg*ones(1,nPoints);
%Ricerca delle PCA con svd
[U,S,V]=svd(B/sqrt(nPoints),'econ');

subplot(1,2,2)
% Plot dei dati per evidenziare PCA
scatter(X(1,:),X(2,:),'k.','LineWidth',2)
hold on, box on, grid on
axis([-6 8 -6 8])

theta = (0:.01:1)*2*pi;
[Xstd] = U*S*[cos(theta); sin(theta)];
% Plot degli intervalli di confidenza
plot(Xavg(1)+Xstd(1,:),Xavg(2)+Xstd(2,:),'r-','LineWidth',1.5)
plot(Xavg(1)+2*Xstd(1,:),Xavg(2)+2*Xstd(2,:),'r-','LineWidth',1.5)
plot(Xavg(1)+3*Xstd(1,:),Xavg(2)+3*Xstd(2,:),'r-','LineWidth',1.5)

% Plot delle componenti principali
plot([Xavg(1) Xavg(1)+U(1,1)*S(1,1)], ...
    [Xavg(2) Xavg(2)+U(2,1)*S(2,1)],'c-','LineWidth',1.5)
plot([Xavg(1) Xavg(1)+U(1,2)*S(2,2)], ...
    [Xavg(2) Xavg(2)+U(2,2)*S(2,2)],'c-','LineWidth',1.5)

set(gcf,'Position',[1400 100 3*600 3*300])
```

Calcolo della PCA per il dataset `ovariancancer` I

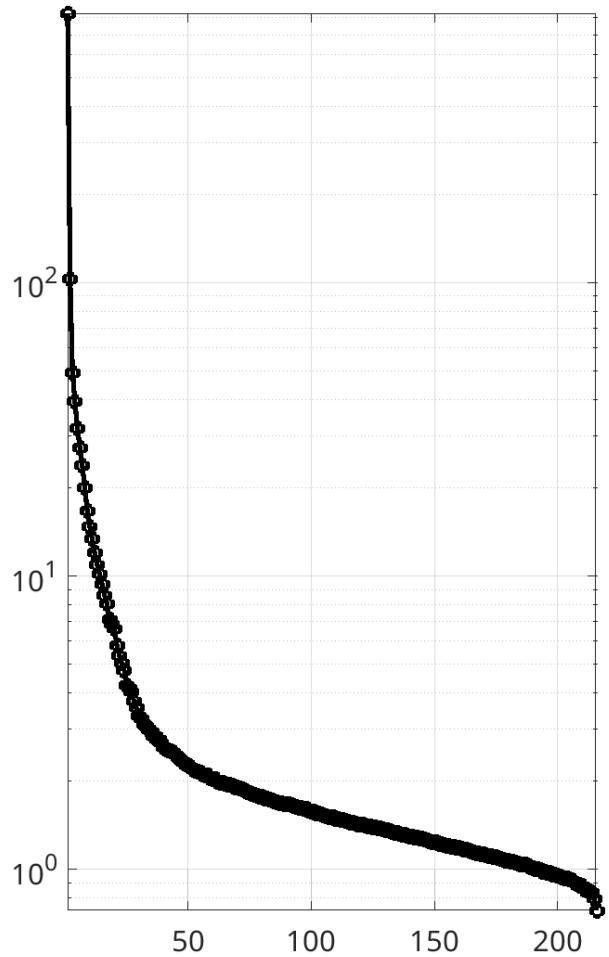
In questo esperimento utilizziamo la Principal Component Analysis (PCA) tramite Singular Value Decomposition (SVD) per analizzare dati di spettrometria di massa relativi a campioni di pazienti con tumore ovarico. Gli obiettivi sono:

- Visualizzare l'importanza di ciascuna componente principale (PC).
- Capire quanta varianza è spiegata dalle prime PC.
- Rappresentare i dati in 3D usando le prime tre componenti per distinguere i campioni Cancer da Normal.

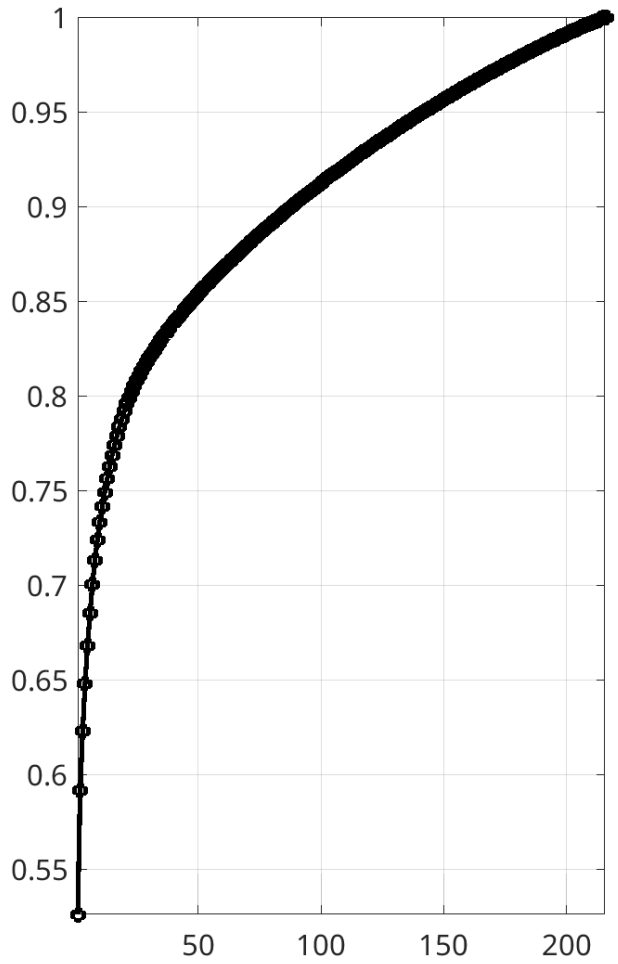
Risultati ottenuti

Eseguendo la PCA tramite decomposizione ai valori singolari (SVD), osserviamo che una quota rilevante della varianza complessiva è concentrata nelle prime componenti principali. Questo significa che i dati genetici dei pazienti mostrano una forte ridondanza e correlazione, ovvero molte variabili (geni) variano insieme.

La decrescita rapida dei valori singolari indica che già le prime poche componenti principali catturano la maggior parte dell'informazione utile, mentre le componenti successive rappresentano solo dettagli marginali o rumore.



La curva cumulativa della varianza spiegata conferma questo comportamento, mostrando che è possibile rappresentare i dati in uno spazio a bassa dimensionalità senza perdite significative di informazione. In termini pratici, questo suggerisce che l'espressione genica dei pazienti tende a seguire pattern comuni, con sovrapposizioni evidenti tra individui, soprattutto tra quelli con caratteristiche biologiche simili.



```
clear all, close all, clc

load ovariancancer.mat;

[U,S,V] = svd(obs,'econ');

figure
subplot(1,2,1)
semilogy(diag(S),'k-o','LineWidth',2.5)
set(gca,'FontSize',15),axis tight, grid on
subplot(1,2,2)
plot(cumsum(diag(S))./sum(diag(S)),'k-o','LineWidth',2.5)
set(gca,'FontSize',15),axis tight, grid on
set(gcf,'Position',[1400 100 3*600 3*250])

figure, hold on
for i=1:size(obs,1)
    x = V(:,1)'*obs(i,:)';
    y = V(:,2)'*obs(i,:)';
    z = V(:,3)'*obs(i,:)';
    if(grp{i}=='Cancer')
        plot3(x,y,z,'rx','LineWidth',3);
    else
        plot3(x,y,z,'bo','LineWidth',3);
    end
```

```
end
xlabel('PC1'),ylabel('PC2'),zlabel('PC3'),
legend('Cancer','Normal')
view(85,25),grid on, set(gca,'FontSize',15)
```