UNIVERSITÀ DEGLI STUDI DI BARI

Facoltà di Scienze Matematiche, Fisiche e Naturali

Corso di Laurea di I Livello in

FISICA GENERALE

TESI DI LAUREA IN FISICA

Algoritmo di Clustering basato sulla Meccanica Quantistica

Relatore:

Prof. Sebastiano Stramaglia

Laureando:

Giuseppe Colucci

ANNO ACCADEMICO 2006/2007

Indice

Introduzione

"Il criterio era rigoroso, e credo che sia
lo stesso seguito dai servizi segreti: non ci sono informazioni
migliori delle altre, il potere sta nello schedarle tutte,
e poi cercare le connessioni.
Le connessioni ci sono sempre, basta volerle trovare."
Il Pendolo di Foucault, U. Eco, *GEBURAH, 34*

Aristotele (384-322 a.C.) viene riconosciuto come il primo scienziato che ha assegnato un significato scientifico al concetto di classificazione. In particolare, egli propose l'uso delle *categorie*, inizialmente introdotte nella sua *metafisica* e successivamente estese alla *logica*, per descrivere il meccanismo della classificazione.

Nella logica di Aristotele, gli oggetti del discorso, ovvero i concetti, possono essere disposti entro una scala di maggiore o di minore universalità e classificati mediante un rapporto di *genere* e *specie*, in cui la specie rappresenta il contenuto, meno universale, e il genere rappresenta il contenente, quindi più universale. La scala complessiva dei concetti può quindi essere percorsa in due direzioni: dall'alto verso il basso (cioè nel senso genere-specie) si avrà un aumento di comprensione e una progressiva diminuzione di estensione fino a raggiungere la *specie infima*, ovvero l'individuo; dal basso verso l'alto (nel senso specie-genere), la piramide dei concetti offre invece un graduale au-

mento di estensione e una altrettanto graduale diminuzione di comprensione, fino ad arrivare ai generi sommi, ovvero le categorie.

Se si analizza il concetto moderno di classificazione gerarchica, si trova che essa corrisponde perfettamente alla nozione introdotta da Aristotele più di duemila anni fa'.

Si può dire dunque che la classificazione, indipendentemente dall'oggetto della stessa, sia una problematica che da sempre l'uomo cerca di affrontare per migliorare le sue capacità cognitive e raggiungere un livello di conoscenza sempre maggiore.

In questo lavoro di tesi si affronta il problema della classificazione non supervisionata, detta *clustering*, in particolare un algoritmo di clustering basato sulla meccanica quantistica.

Nel primo capitolo vengono illustrate le tecniche fondamentali di clustering, gerarchico e partizionale. Inizialmente vengono date delle definizioni relative alla rappresentazione dei dati e alla definizione di distanza nello spazio di rappresentazione dei dati con un formalismo matriciale. Successivamente si procede con l'analisi delle diverse tipologie di clustering, con l'analisi dei più comuni algoritmi di classificazione.

Nel secondo capitolo viene formalmente introdotto l'oggetto di discussione del presente lavoro: il Quantum Clustering, ovvero un algoritmo di clustering basato sulla meccanica quantistica. L'analogia di tale metodo con la meccanica quantistica si ritrova nella definizione dell'equazione di Schrödinger relativa ad una funzione d'onda definita a partire dai dati dell'insieme che si vuole classificare. Vengono illustrate, inoltre, diverse tecniche matematiche e statistiche alla base di questo algoritmo, con particolare riferimento al metodo di Parzen, all'analisi in componenti principali e al metodo di minimizzazione del gradiente (*gradient descent*).

Nel terzo capitolo sono illustrati i risultati del suddetto algoritmo relativi agli insiemi di dati di due sistemi biologici, *Iris* ed *E.Coli*. Dopo una breve descrizione degli insiemi, si procede con la visualizzazione bidimensionale dei dati dei due insiemi mediante l'analisi in componenti principali e con l'individuazione dei centri dei cluster, ovvero con l'individuazione dei gruppi dell'insieme i cui elementi sono tra loro più simili.

In appendice viene riportato il codice con la procedura di quantum clustering sviluppato con il software *Mathematica*.

Capitolo 1

Cenni sulla classificazione non supervisionata

Il cervello umano ha una certa tendenza nel trovare delle regolarità nei dati. Un modo per individuare tale regolarità consiste nel creare in un insieme di oggetti dei gruppi i cui elementi siano *simili* tra loro, ovvero effettuare una *classificazione*.

Una definizione di classificazione risale a J.S. Mill [18], il quale la definisce come *l'unione, reale o ideale, di ciò che è simile e la separazione di ciò che è differente.* Gli scopi primari di questa sistemazione sono[1]:

a. formare e acquisire la conoscenza,

b. analizzare le strutture del fenomeno in questione e

c. mettere in relazione gli aspetti del fenomeno con gli altri.

In questo capitolo vengono introdotte le nozioni di base relative alla classificazione non supervisionata (*clustering*). Si prosegue con una panoramica sulle principali metodologie di clustering.

[1]Mirkin B., *Mathematical Classification and Clustering*, J. Wiley & Sons (1996)

1.1 Clustering

Il clustering viene definito, in accordo con Jain [15], la classificazione o categorizzazione di oggetti, quali osservazioni o dati numerici, in gruppi (*cluster*). Più precisamente, il clustering ha come scopo la classificazione oggettiva e completamente non supervisionata dei dati in cluster significativi, questi ultimi derivanti solo da una struttura intrinseca ai dati stessi. La determinazione di questa struttura di raggruppamento insita nei dati ha come scopo l'agevolazione dell'interpretazione della realtà fenomenica che si sta analizzando.

In generale, il processo del clustering si articola nei seguenti passi:

1. rappresentazione dei dati;
2. definizione di un indice di prossimità dei dati, e
3. raggruppamento dei dati (clustering).

L'insieme dei dati di origine può essere descritto mediante l'introduzione di due differenti oggetti geometrici: la matrice dei pattern (*pattern matrix*) e la matrice di prossimità (*proximity matrix*).

Si consideri un insieme di n oggetti, ognuno dei quali è rappresentato da un vettore $\mathbf{x}$ d-dimensionale o *pattern*:

$$\mathbf{x} = (x_1, ..., x_d).$$

Ciascuna componente scalare x_i di $\mathbf{x}$ è detta *feature*, ovvero attributo, e corrisponde a una determinata proprietà o carattere numerico dell'oggetto. L'insieme dei dati può quindi essere rappresentato da una matrice $n \times d$, detta matrice dei pattern, in cui ogni riga individua un oggetto e ogni colonna definisce un attributo. Questa matrice viene definita anche *two mode* in quanto le righe e le colonne sono entità totalmente differenti.

Le d feature vengono spesso rappresentate come un sistema ortonormale

completo, con il quale è possibile definire uno spazio vettoriale d-dimensionale detto *spazio delle feature*. Considerando l'isomorfismo esistente tra vettori e punti dello spazio, ogni pattern di partenza si identifica con un punto dello spazio e le sue coordinate sono le feature.

La seconda struttura di rappresentazione dei dati prevede l'introduzione del concetto di *indice di prossimità* tra le coppie di pattern all'interno dello spazio delle features. Questi indici possono essere rappresentati mediante una matrice $n \times n$, la matrice di prossimità, che è detta anche *one mode*.

La scelta dell'indice di prossimità è di importanza cruciale, in quanto da esso dipende la tipologia di clustering e conseguentemente la sua validità.

Un indice di prossimità può essere di due tipi: somiglianza o dissomiglianza.

Formalmente, dato un insieme X di pattern si definisce indice di prossimità un'applicazione $d : X \times X \longrightarrow \mathbb{R}$ che soddisfa le seguenti proprietà:

1. **a.** $d(i,i) = 0$, (dissomiglianza)

   b. $d(i,i) \geq \max_k d(i,k)$, (somiglianza)

2. $d(i,j) = d(j,i)$

3. $d(i,j) \geq 0$.

Dalla definizione data si evince che l'indice di dissomiglianza corrisponde ad una distanza (o metrica) se soddisfa le due seguenti proprietà:

1. $d(i,j) = 0 \Leftrightarrow i = j$

2. $d(i,j) \leq d(i,k) + d(k,j)$.

In tal caso la coppia (X,d) è chiamata *spazio metrico*.

Una delle metriche più utilizzate nel clustering è la *distanza Euclidea*

$$d_2(\mathbf{x}_i, \mathbf{x}_j) = \sqrt{\sum_{k=1}^{d} |x_{i,k} - y_{j,k}|^2} = ||\mathbf{x}_i - \mathbf{x}_j||_2$$

caso particolare della *metrica di Minkowski*

$$d_p(\mathbf{x}_i, \mathbf{x}_j) = (\sum_{k=1}^{d} |x_{i,k} - y_{j,k}|^p)^{1/p} = ||\mathbf{x}_i - \mathbf{x}_j||_p.$$

Un'altra metrica molto utilizzata è la *city block* o *Manhattan*

$$d_M(\mathbf{x}_i, \mathbf{x}_j) = \sum_{k=1}^{d} |x_{i,k} - x_{j,k}| + |y_{i,k} - y_{j,k}|,$$

introdotta da Carmichael e Sneath nel 1969 e cosiddetta in riferimento alla minima distanza tra due punti in una città in cui le strade siano perpendicolari tra loro (quale è la città di Manhattan).

In base alle due tipologie di indice di prossimità possiamo distinguere le matrici di prossimità in *matrici di somiglianza* o di *dissomiglianza*.

Un indice di dissomiglianza è sempre positivo, è prossimo allo zero se gli elementi in questione sono simili tra loro e diventa molto grande quando gli oggetti sono molto differenti. L'indice di dissomiglianza tra un oggetto e sé stesso è quindi pari a 0.

Un esempio di indice di dissomiglianza è la metrica euclidea o di Manhattan, ma in generale un indice di prossimità non deve soddisfare la disuguaglianza triangolare, quindi non sempre corrisponde ad una metrica.

Si assume che la matrice di dissomiglianza sia una matrice simmetrica $n \times n$ definita positiva in cui l'elemento *(i,j)* è l'indice di dissomiglianza tra l'i-esimo e il j-esimo oggetto, e, per quanto detto prima, gli elementi della diagonale principale sono tutti nulli.

L'indice di somiglianza si definisce in modo analogo all'indice di dissomiglianza ed è tanto maggiore quanto più simili tra loro sono gli oggetti.

Spesso l'indice di somiglianza *s(i,j)* viene normalizzato tra 0 e 1, dove lo 0 indica che i e j non sono simili e 1 rappresenta la massima somiglianza tra i due elementi. Si assume quindi che vengano soddisfatte le seguenti condizioni:

1. $0 \leq s(i,j) \leq 1$
2. $s(i,i) = 1$
3. $s(i,j) = s(j,i)$

I valori *s(i,j)* sistemati in una matrice $n \times n$ danno luogo alla matrice di somiglianza, anch'essa simmetrica e con gli elementi della diagonale tutti uguali ad 1.

Fissata la misura di somiglianza, il clustering si riduce alla suddivisione di N punti in K cluster in modo che due punti che siano in un medesimo cluster siano più simili tra loro rispetto ad altri due punti che appartengono a cluster differenti.

Gli algoritmi di clustering possono essere divisi in due classi: metodi gerarchici e partizionali.

1.2 Algoritmi gerarchici

Gli algoritmi gerarchici sono spesso utilizzati nell'analisi dei dati per riassumerne le caratteristiche in modo rapido e completo. Essi infatti sono molto più versatili degli algoritmi partizionali, in quanto consentono l'analisi anche di insiemi di dati strutturati in cluster non isotropici. Il carattere chiaro ed evidente dei metodi gerarchici è messo in risalto nella rappresentazione grafica della struttura di raggruppamento tramite un *diagramma ad albero* o *dendrogramma*. Sezionando il dendrogramma in corrispondenza di

un certo livello di dissomiglianza si ottiene una partizione in gruppi disgiunti e omogenei dell'insieme di partenza.

Alla base di un algoritmo gerarchico ci sono i processi di *merging* (unione) e *splitting* (separazione). In base ai suddetti processi, gli algoritmi gerarchici si distinguono rispettivamente in *agglomerativi* e *divisivi*.

1.2.1 Formulazione matematica del clustering gerarchico

Sia X un insieme finito di elementi.

Definizione 1.1. Si definisce *partizione* di X un insieme $C = \{C_1, C_2, \ldots, C_m\}$ tale che

1. $C_i \bigcap C_j = \emptyset, \qquad i, j \in \{1, \ldots, m\}, \quad i \neq j,$
2. $C_1 \bigcup C_2 \bigcup \cdots \bigcup C_m = X.$

Sia P una successione ordinata di partizioni su X

$$p_0, p_1, \ldots, p_k, p_{k+1}, \ldots, p_m,$$

dove p_0 è la partizione le cui classi sono i singoli elementi di X, p_{k+1} è ottenuta da p_k per aggregazione di classi e p_m ha un unica classe contenente tutti gli elementi di X.

Definizione 1.2. Si definisce *gerarchia* un insieme H tale che

1. $X \in H,$
2. $x \in X \Rightarrow \{x\} \in H,$
3. $\forall h_1, h_2 \in H, \qquad h_1 \neq h_2 \Rightarrow h_1 \subset h_2 \quad \vee \quad h_2 \subset h_1.$

P è quindi una gerarchia in quanto soddisfa la definizione data; in particolare essa è detta *classificazione gerarchica su X*. Il dendrogramma risulta quindi una struttura ad albero con $m + 1$ livelli, ciascuno associato ad una partizione di P.

La formulazione matematica dei metodi gerarchici presuppone l'utilizzo di un formalismo matriciale nella definizione degli indici di prossimità.

Una definizione alternativa, ma altrettanto rigorosa dei metodi gerarchici si ottiene anche a partire dalla *teoria dei grafi* [2][15].

Dalla seconda metà degli anni '50, infatti, alcune delle tecniche di raggruppamento hanno ricevuto una più ampia trattazione teorico-metodologica grazie alla corrispondenza con la teoria dei grafi. Sono stati quindi sviluppati degli algoritmi di clustering in cui la misura di somiglianza è data dalla definizione di un *grafo di prossimità*. Esso è un grafo in cui ogni *arco* (*edge*) è *pesato* in base alla sua vicinanza tra due pattern. I cluster vengono quindi formati costruendo un *minimum spanning tree* (l'albero di connessione del grafo di peso minimo) e cancellandone gli spigoli iterativamente.

1.2.2 Algoritmi agglomerativi

Dato un insieme X di N oggetti da raggruppare, considero la matrice di somiglianza $N \times d$. L'algoritmo agglomerativo individua i cluster iniziali con gli N oggetti. Successivamente, trova la coppia di cluster più simili e li fonde (*merging*) in un unico cluster. Procede in questo modo finché tutti i pattern si trovano in un unico cluster. Una volta stabilito il livello di dissomiglianza, ciò che differenzia le diverse tecniche di clustering gerarchico è la distanza all'interno dello spazio delle features.

Metodo del legame singolo (Single-link method)

La distanza tra due gruppi è data dalla *minore* delle distanze tra gli elementi. Una possibile conseguenza negativa di questo metodo è l'unione di unità appartenenti a gruppi diversi.

Metodo del legame completo (Complete-link method)

La distanza tra due gruppi è data dalla *maggiore* delle distanze tra gli elementi, ovvero dal diametro della più piccola sfera che include il gruppo ottenuto aggregando i due gruppi.

Metodo del legame medio (Group-average method)

La distanza tra due gruppi è data dalla *media* delle distanze tra gli elementi.

Metodo del centroide (Centroid method)

Il *centroide* di ciascun gruppo è definito come il punto che ha per coordinate la media delle coordinate degli elementi del gruppo. La distanza tra due gruppi è data dalla distanza euclidea tra i due centroidi corrispondenti. Ad ogni passo della procedura vengono aggregati i gruppi per i quali la distanza euclidea tra i centroidi risulta minima.

1.2.3 Algoritmi divisivi

Le tecniche e gli algoritmi di clustering gerarchico divisivi procedono in modo inverso rispetto al caso agglomerativo. Un algoritmo divisivo, infatti, considera inizialmente un unico cluster contenente tutti gli oggetti x del data set X di partenza. Successivamente procede per divisione (*splitting*),

sulla base di una metrica fissata nello spazio delle features. I metodi che ne derivano sono gli stessi descritti nella sezione precedente.

1.3 Algoritmi partizionali

Il clustering partizionale ha come obiettivo la creazione di un'unica partizione dei dati di partenza. Il maggiore vantaggio di queste tecniche rispetto al clustering gerarchico sta nella velocità di classificazione dei dati, anche nel caso di una massa ingente di dati.

Formalmente le metodologie di clustering partizionale si basano sulla suddivisione in due cluster del cluster di partenza che massimizza una funzione obiettivo (*criterion function*). Spesso questa funzione è identificata con la distanza tra i centroidi dei gruppi (*squared error*): dato un insieme X di elementi $\mathbf{x}_i$, la funzione relativa alla partizione P di X contenente K clusters è:

$$e^2(X,P) = \sum_{j=1}^{K} \sum_{i=1}^{n_j} ||\mathbf{x}_i^{(j)} - \mathbf{c}_j||^2$$

dove $\mathbf{x}_i^{(j)}$ è l'i-esimo elemento appartenente al j-esimo cluster, e $\mathbf{c}_j$ è il centroide del j-esimo cluster.

L'algoritmo partizionale più semplice ed al tempo stesso più utilizzato è il *k-means* (Mc Queen, 1967).

1.3.1 Algoritmo K-means

L'algoritmo *K-means* sceglie inizialmente k centroidi, coincidenti con k pattern dell'insieme di partenza scelti in maniera random. Assegna ciascun pattern al cluster con il centroide più vicino all'elemento stesso. Ricalco-

la i centroidi di tutti i cluster, e dopo aver fissato un criterio di convergenza (spesso dato dalla massimizzazione della funzione obiettivo) procede iterativamente fino al raggiungimento del suddetto criterio.

Il vantaggio maggiore di questo algoritmo è che essendo fissati il numero dei cluster ed il numero delle iterazioni, la complessità risulta lineare.

Il fatto che il numero dei cluster sia fissato a priori può risultare però uno svantaggio. Un algoritmo alternativo al *k-means* che però elimina il problema della conoscenza a priori del numero di cluster finali è l'ISODATA (Iterative Self-Organizing Data Analysis Technique A). Esso infatti è un algoritmo di clustering *dinamico* in quanto, nonostante proceda in modo identico al *k-means*, permette al termine del ciclo di iterazione la variazione del numero di cluster mediante i processi di *merging* e *splitting* degli stessi.

1.4 Altri algoritmi

Altri algoritmi di clustering sono stati sviluppati in base alle esigenze di diverse discipline.

Fuzzy clustering

Nelle tecniche di clustering precedentemente descritte ogni pattern appartiene ad uno ed un solo cluster. Ciò implica che i diversi cluster siano tra loro disgiunti e da ciò deriva l'appellativo di *hard clusters*. Gli algoritmi relativi all'hard clustering possono però essere variati in modo tale che essi permettano di considerare l'eventuale intersezione dei cluster. Un esempio è dato dall'algoritmo di *fuzzy clustering*, in cui ogni pattern è associato ad ogni cluster mediante una *funzione di appartenenza*; *fuzzy*, infatti, sta per *sfocato*, e si riferisce al diverso grado di appartenenza degli elementi in cluster, che

può idealmente essere rappresentato dalla variazione di *intensità* e quindi di *definizione* del cluster stesso (in contrapposizione ai netti contorni del *hard clustering*).

Algoritmi basati sulla fisica

Alcuni metodi di clustering si basano su analogie fenomenologiche e formali con fenomeni fisici. Un esempio è dato dall'algoritmo sviluppato da M. Blatt, S. Wiseman, E. Domany [4] in cui i patterns sono associati agli spin interagenti in un reticolo cristallino (modello di Potts) e la matrice di dissomiglianza viene costruita a partire dalle interazioni ferromagnetiche tra ciascuna coppia di spin adiacenti.

Un altro algoritmo basato sulla fisica, in particolare sulla meccanica quantistica, è stato sviluppato dai fisici David Horn e Assaf Gottlieb, che nel loro articolo [13] [14] hanno illustrato le applicazioni relative a diversi data set.

Nel prossimo capitolo saranno discusse la formulazione e le applicazioni di questo algoritmo.

Capitolo 2

Quantum Clustering (QC)

In questo capitolo viene formalmente introdotto l'algoritmo di clustering di D. Horn e A. Gottlieb basato sulla meccanica quantistica.

Dato un insieme di pattern, l'algoritmo inizialmente definisce una funzione data da una sovrapposizione di gaussiane costruita a partire direttamente dai dati dell'insieme. Tale funzione approssima la funzione di distribuzione di probabilità dei punti nello spazio delle features. Successivamente la identifica con il *ground state* dell'equazione di Schrödinger e da quest'ultima viene ottenuto il potenziale, i cui minimi individuano i centri dei cluster.

2.1 Formulazione generale

2.1.1 Metodo di Parzen

Per ottenere una stima della funzione di distribuzione dei dati di un insieme generico vengono spesso utilizzati dei metodi non parametrici di stima della densità di probabilità.

Sia $\mathbf{X}$ un insieme di n elementi $\mathbf{x}_1, \ldots, \mathbf{x}_i, \ldots, \mathbf{x}_n$ d-dimensionali, la cui

funzione di distribuzione è $p(\mathbf{x})$. La probabilità che un vettore $\mathbf{x}_i$ si trovi nella regione $\mathcal{R}$ è data da

$$P = \int_{\mathcal{R}} p(\mathbf{x}')d\mathbf{x}'. \tag{2.1}$$

Di conseguenza, la probabilità che k degli n elementi si trovino nella regione $\mathcal{R}$ è data dalla legge binomiale

$$P_k = \begin{pmatrix} n \\ k \end{pmatrix} P^k(1-P)^{n-k}, \tag{2.2}$$

il cui valore di aspettazione per k è

$$E[k] = nP. \tag{2.3}$$

Inoltre, per n sufficientemente grande, si trova che una buona stima di P è data da k/n. Ora, supponendo che $p(\mathbf{x})$ sia continua e che $\mathcal{R}$ sia abbastanza piccola da poter considerare $p(\mathbf{x})$ costante al suo interno, si può dire che

$$\int_{\mathcal{R}} p(\mathbf{x}')d\mathbf{x}' \simeq p(\mathbf{x})V, \tag{2.4}$$

dove $\mathbf{x}$ è un punto e V è il volume racchiuso in $\mathcal{R}$.

Combinando la (2.1),(2.3) e (2.4), si ottiene la stima di densità

$$p(\mathbf{x}) = \frac{k/n}{V}. \tag{2.5}$$

Ora, si supponga che il volume V sia fissato e che n sia sufficientemente grande, allora, facendo tendere V a zero,

$$\frac{P}{V} = \frac{\int_{\mathcal{R}} p(\mathbf{x}')d\mathbf{x}'}{\int_{\mathcal{R}} d\mathbf{x}'} \tag{2.6}$$

rappresenta una media di $p(\mathbf{x})$.

Tuttavia, per n fissato, $\mathcal{R}$ diventerebbe talmente piccola tanto da rendere $k = 0$, e quindi $p(\mathbf{x}) \simeq 0$. Se infatti k fosse maggiore di zero, la stima divergerebbe ($p(\mathbf{x}) \to \infty$).

Si supponga di avere un numero illimitato di campioni. Per valutare $p(\mathbf{x})$ si consideri una sequenza di regioni $\mathcal{R}_1, \mathcal{R}_2, \ldots, \mathcal{R}_n$ contenenti $\mathbf{x}$, dove la regione $\mathcal{R}_s$ viene utilizzata nel caso in cui $n = s$. Se V_n è il volume di $\mathcal{R}_n$, k_n il numero di campioni che cadono in $\mathcal{R}_n$ e $p_n(\mathbf{x})$ è l n-esima stima di $p(\mathbf{x})$, si può scrivere

$$p_n(\mathbf{x}) = \frac{k_n/n}{V_n}. \tag{2.7}$$

Per $n \to \infty$, si ottiene una serie di stime di densità $p_1(\mathbf{x}), \ldots, p_n(\mathbf{x}), \ldots$ e si dimostra che la serie converge a $p(\mathbf{x})$ se e solo se sono soddisfatte le seguenti condizioni:

- $\lim_{n\to\infty} V_n = 0$
- $\lim_{n\to\infty} k_n = \infty$
- $\lim_{n\to\infty} \frac{k_n}{n} = 0$

Un metodo utilizzato per ottenere una sequenza di regioni che soddisfano le precedenti condizioni è il *metodo della finestra di Parzen*. Tale metodo consiste nell'assumere che la regione $\mathcal{R}_n$ sia un ipercubo d-dimensionale. Detta h_n la lunghezza del lato dell'ipercubo, il volume di $\mathcal{R}_n$ risulta

$$V_n = h_n^d. \tag{2.8}$$

Per ottenere un espressione analitica del numero di elementi dell'insieme di partenza all'interno dell'ipercubo, k_n, si definisce la funzione (*window function*)

$$\phi(\mathbf{u}) = \begin{cases} 1 & |u_j| \leq 1/2 \quad j = 1, \ldots, d \\ 0 & \text{altrimenti} \end{cases} \tag{2.9}$$

Quindi, $\phi(\mathbf{u})$ definisce un ipercubo unitario centrato nell'origine e $\phi((\mathbf{x} - \mathbf{x}_j)/h_n)$ è uguale a 1 se $\mathbf{x}_j$ si trova nell'ipercubo, altrimenti è uguale a zero.

Il numero totale di elementi nell'ipercubo $\mathcal{R}_n$ è quindi

$$k_n = \sum_{i=1}^{n} \phi\left(\frac{\mathbf{x} - \mathbf{x}_j}{h_n}\right), \tag{2.10}$$

e sostituendo nella (2.7) si ottiene

$$p_n(\mathbf{x}) = \frac{1}{n} \sum_{i=1}^{n} \frac{1}{V_n} \phi\left(\frac{\mathbf{x} - \mathbf{x}_j}{h_n}\right). \tag{2.11}$$

In questo modo, $p_n(\mathbf{x})$ stima $p(\mathbf{x})$ come la media di funzioni di $\mathbf{x}$ e dei campioni $\mathbf{x}_i$, con $i = 1, \ldots, n$. La funzione $\phi(\mathbf{x})$ può essere di forma generale purché siano verificate le condizioni

$$\phi(\mathbf{u}) \geq 0 \qquad \text{e} \qquad \int \phi(\mathbf{u}) d\mathbf{u} = 1, \tag{2.12}$$

le quali assicurano che $p_n(\mathbf{x})$ sia una densità di probabilità, ovvero che $p_n(\mathbf{x})$ sia definita positiva e che sia normalizzata ad 1. La funzione $\phi(\mathbf{u})$ è spesso chiamata *kernel* e rappresenta il contributo di ogni punto alla stima. La stima complessiva è quindi ottenuta sommando i contributi di tutti i punti.

In generale, dato l'insieme $\mathbf{X}$ di n elementi $\mathbf{x}_1, \ldots, \mathbf{x}_i, \ldots, \mathbf{x}_n$ d-dimensionali, si cerca una funzione del tipo

$$p(\mathbf{x}) = \sum_{i=1}^{n} C_i \phi_i(\mathbf{x}). \tag{2.13}$$

Con il metodo della finestra di Parzen, però, è possibile considerare i pesi C_i indipendenti dalle loro posizioni, quindi la funzione da cercare diviene

$$p(\mathbf{x}) = C \sum_{i=1}^{n} \phi_i(\mathbf{x}). \tag{2.14}$$

Se si utilizzano come kernel delle gaussiane e si indica la stima di densità di probabilità con $\psi(\mathbf{x})$ si ottiene

$$\psi(\mathbf{x}) = C \sum_{i=1}^{n} e^{-\frac{(\mathbf{x}-\mathbf{x}_i)^2}{2\sigma^2}}. \tag{2.15}$$

2.1.2 L'equazione di Schrödinger

Si assume che ψ sia un autostato dell'equazione di Schrödinger

$$H\psi \equiv \left(-\frac{\sigma^2}{2}\nabla^2 + V(\mathbf{x})\right)\psi = E\psi, \tag{2.16}$$

dove σ è l'unico parametro libero dell'equazione che contiene tutte le costanti ed E è l'autovalore del *ground state*.

In meccanica quantistica il modulo quadro di una funzione d'onda $|\psi|^2$ viene interpretato come la distribuzione di probabilità al variare di $\mathbf{x}$. Nell'algoritmo descritto, la ψ rappresenta proprio la distribuzione dei punti dell'insieme $\mathbf{X}$, quindi anche il modulo quadro è associato alla distribuzione degli elementi dell'insieme, creando in tal modo una stretta analogia con il caso fisico.

Per qualsiasi insieme di dati, con σ fissata, ψ è data dalla (2.15) e il potenziale V è dato da

$$V(\mathbf{x}) = E + \frac{\frac{\sigma^2}{2}\nabla^2\psi}{\psi}, \tag{2.17}$$

che in coordinate cartesiane risulta

$$V(\mathbf{x}) = E - \frac{d}{2} + \frac{1}{2\sigma^2\psi}\sum_i (\mathbf{x}-\mathbf{x}_i)^2 e^{-\frac{(\mathbf{x}-\mathbf{x}_i)^2}{2\sigma^2}} = E - \frac{d}{2} + \sigma^2\frac{\partial}{\partial\sigma^2}\ln\psi. \tag{2.18}$$

Ponendo che il $\min(V) = 0$, il valore di E diventa

$$E = -min(\frac{\frac{\sigma^2}{2}\nabla^2\psi}{\psi}), \tag{2.19}$$

da cui segue il seguente

Teorema 2.1.1.

$$0 < E = \frac{d}{2} - min(\sigma^2\frac{\partial}{\partial\sigma^2}ln\psi) \leq \frac{d}{2} \tag{2.20}$$

Dim. Per il limite inferiore, segue direttamente dall'Hamiltoniana H che il suo autovalore E deve essere positivo in quanto V è una funzione non negativa per definizione e tale è anche l'operatore di energia cinetica (il Laplaciano).

Per il limite superiore, segue dalla considerazione che l'ultimo termine nell'equazione (2.18) è definito positivo.$\diamondsuit$

Se si considera il caso di un solo punto $\mathbf{x}_1$, la soluzione dell'equazione corrisponde al ground state dell'oscillatore armonico in meccanica quantistica con potenziale

$$V(\mathbf{x}) = \frac{1}{2\sigma^2\psi}\sum_i(\mathbf{x}-\mathbf{x}_1)^2$$

e

$$E = \frac{d}{2}.$$

Osservazione 1. L'energia del ground state nel caso di n punti è limitata da quella nel caso di un punto solo.

Dato un insieme di punti d-dimensionali all'interno di una certa regione di spazio, ci si aspetta che il potenziale cresca in maniera quadratica al di fuori della regione, e che abbia uno o più minimi all'interno della regione scelta, identificati ognuno con il centro di un cluster.

Un'interpretazione fisica di quanto affermato può essere ottenuta osservando che dato un potenziale $V(\mathbf{x})$, esso attrae gli elementi descritti dalla distribuzione $\psi(\mathbf{x})$ verso i suoi minimi, mentre, d'altra parte, il termine relativo all'energia cinetica (ovvero il Laplaciano di ψ) tende a diffonderli nello spazio. Il bilanciamento di questi due effetti dà luogo al carattere diffuso della distribuzione degli elementi dell'insieme.

2.2 Analisi in componenti principali (PCA) nel QC

Uno dei problemi principali riscontrati nell'applicazione del metodo di clustering enunciato, è la visualizzazione dei dati.

Nel caso in cui gli elementi dell'insieme abbiano una dimensione pari a 2, si procede semplicemente con la rappresentazione dei dati nel piano in cui le coordinate sono proprio le variabili dell'insieme, *(x,y)* (le *features*). Il potenziale sarà una funzione delle due variabili, $V(x, y)$, rappresentabile quindi con un plot tridimensionale, o con un contour plot che mostra le curve di livello di $V(x, y)$.

Nel caso in cui la dimensione degli elementi sia maggiore di 2, si cerca un metodo per la riduzione di dimensionalità dell'insieme.

A questo proposito, si procede con la descrizione del metodo dell'analisi in componenti principali.

L'analisi delle componenti principali (PCA) è una tecnica utilizzata nell'ambito della statistica multivariata atta alla semplificazione e successiva rappresentazione di un insieme di dati descritti da un elevato numero di variabili tra loro correlate, e quindi non rappresentabile graficamente, mediante la riduzione di dimensionalità dello stesso.

L'analisi della distribuzione di un insieme di n elementi di dimensione d richiede generalmente il calcolo di d medie, d varianze e $d(d-1)/2$ covarianze, per un totale di $2d + d(d-1)/2$ parametri, che cresce in maniera parabolica con la dimensione degli elementi.

Una riduzione nel numero di parametri a $2k$ si otterrebbe se le variabili fossero incorrelate.

La PCA permette la definizione delle *componenti principali*, ovvero un

nuovo set di variabili non correlate, combinazioni lineari delle variabili iniziali, tali che solo alcune di loro contribuiscono alla maggiorparte della variazione presente in tutte le variabili iniziali. Mediante le componenti principali è quindi possibile rappresentare i dati in un uno spazio di dimensione inferiore, pari proprio al numero di componenti principali selezionate.

Il problema di ridurre la dimensionalità di un insieme di dati fu proposto inizialmente da F. Galton (1869) con lo scopo di classificare un insieme di soggetti criminali in base a 12 misure di altrettante caratteristiche antropometriche fortemente correlate tra loro. Successivamente K. Pearson (1901) fornì la prima formulazione teorica delle componenti principali basandosi su di un'interpretazione geometrica delle stesse, mentre nel 1933 H. Hotelling ha sviluppato la teoria di Pearson, fornendo la versione attuale della derivazione algebrica delle componenti principali.

2.2.1 Determinazione delle componenti principali

Si consideri un insieme di n elementi rappresentati dai vettori $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n$ d-dimensionali, in cui la dimensione i-esima corrisponde al carattere numerico X_i. Utilizzando il concetto di matrice dei pattern, è possibile rappresentare gli n elementi con una matrice $n \times d$

$$\mathbf{X} = \begin{pmatrix} x_{11} & \ldots & x_{1d} \\ x_{21} & \ldots & x_{2d} \\ \vdots & \ddots & \vdots \\ x_{n1} & \ldots & x_{nd} \end{pmatrix}$$

in cui l'elemento generico x_{ij} rappresenta la j-esima componente dell'i-esimo pattern.

Poiché gli attributi sono delle variabili numeriche, lo spazio delle features

è generalmente lo spazio d-dimensionale $\mathbb{R}^d$ in cui ogni elemento è rappresentato da un punto. La rappresentazione dell'insieme è quindi possibile solo nel caso in cui $d \leq 3$.

Definizione 2.1. Si dice *baricentro* di $\mathbf{X}$ dei dati il vettore $\overline{\mathbf{x}} = (\overline{x}_{\cdot 1}, \ldots, \overline{x}_{\cdot d}) \in \mathbb{R}^d$ le cui componenti sono le medie dei valori di ciascun carattere, ossia le medie lungo le colonne di $\mathbf{X}$

$$\overline{x}_{\cdot j} = \frac{1}{n} \sum_{i=1}^{n} x_{ij}, \qquad j \in 1, \ldots, d.$$

Definizione 2.2. Si dice *matrice di covarianza* la matrice $\mathbf{S}$ in cui l'elemento s_{jk} è dato dalla covarianza dei caratteri X_j e X_k, ossia delle colonne j-esima e k-esima di $\mathbf{X}$

$$s_{jk} = \frac{1}{n} \sum_{i=1}^{n} (x_{ij} - \overline{x}_{\cdot j})(x_{ik} - \overline{x}_{\cdot k}) = \frac{1}{n} \sum_{i=1}^{n} x_{ij} x_{ik} - \overline{x}_{\cdot j} \overline{x}_{\cdot k}, \qquad j, k \in 1, \ldots, d.$$

Analogamente si definisce la *matrice di correlazione* $\mathbf{C}$ i cui elementi sono i coefficienti di correlazione tra i diversi caratteri.

Osservazione 2. Le matrici di covarianza e di correlazione sono matrici simmetriche di dimensione d.

Definizione 2.3. Si chiama *dispersione totale* dei dati la quantità

$$\Delta = \frac{1}{n} \sum_{i=1}^{n} |\mathbf{x}_i - \overline{\mathbf{x}}|^2.$$

Data la matrice $\mathbf{X}$ $n \times d$ dell'insieme dei dati iniziali, si consideri una nuova matrice $\mathbf{Y}$ ottenuta mediante una trasformazione lineare di $\mathbf{X}$

$$\mathbf{Y} = \mathbf{A}\mathbf{X}^T, \tag{2.21}$$

con $\mathbf{A}$ matrice $d \times d$ della trasformazione lineare

$$\mathbf{A} = \begin{pmatrix} \alpha_{11} & \ldots & \alpha_{1d} \\ \vdots & \ddots & \vdots \\ \alpha_{d1} & \ldots & \alpha_{dd} \end{pmatrix}$$

Una generica componente di $\mathbf{Y}$, $\mathbf{y}_j$, si può quindi esprimere come

$$\mathbf{y}_j = \begin{pmatrix} y_{j1} \\ \vdots \\ y_{jn} \end{pmatrix} = \begin{pmatrix} \sum_{i=1}^{d} x_{1i}\alpha_{ij} \\ \vdots \\ \sum_{i=1}^{d} x_{ni}\alpha_{ij} \end{pmatrix} = \mathbf{X}\boldsymbol{\alpha}_j.$$

dove $\boldsymbol{\alpha}_j$ è $(\alpha_{j1}, \ldots, \alpha_{jd})^T$, ovvero la j-esima colonna della matrice $\mathbf{A}$.

Detta $\mathbf{S_X}$ la matrice di covarianza di $\mathbf{X}$, la matrice di covarianza di $\mathbf{Y}$ è la matrice $\mathbf{S_Y}$, in cui l'elemento s_{ij} è dato dalla covarianza degli elementi $\mathbf{y}_i$ e $\mathbf{y}_j$:

$$s_{ij} \equiv s(\mathbf{y}_i, \mathbf{y}_j) = \boldsymbol{\alpha}_j^T \mathbf{S_X} \boldsymbol{\alpha}_i$$

Se si impone che il vettore $\mathbf{y}_1 = \mathbf{X}\boldsymbol{\alpha}_1$ abbia varianza massima rispetto alle altre componenti della matrice $\mathbf{Y}$, allora $\mathbf{y}_1$ si dice *prima componente principale*. Inoltre, poiché la varianza $\boldsymbol{\alpha}_1^T \mathbf{S_X} \boldsymbol{\alpha}_1$ è una funzione crescente dei valori di $\boldsymbol{\alpha}_1$, si pone il seguente vincolo sulla dimensione di $\boldsymbol{\alpha}_1$

$$\boldsymbol{\alpha}_1^T \boldsymbol{\alpha}_1 = 1.$$

La prima componente principale può quindi essere determinata risolvendo il problema di ottimizzazione:

$$\begin{cases} max_{\boldsymbol{\alpha}_1} \boldsymbol{\alpha}_1^T \mathbf{S_X} \boldsymbol{\alpha}_1 \\ \boldsymbol{\alpha}_1^T \boldsymbol{\alpha}_1 = 1 \end{cases} \tag{2.22}$$

L'approccio tipico per la risoluzione del problema (2.22) è il metodo dei moltiplicatori di Lagrange.

La funzione Lagrangiana relativa al problema (2.22) è

$$\mathfrak{L}(\boldsymbol{\alpha}_1, \lambda) = \boldsymbol{\alpha}_1^T \mathbf{S_X} \boldsymbol{\alpha}_1 - \lambda(\boldsymbol{\alpha}_1^T \boldsymbol{\alpha}_1 - 1) \tag{2.23}$$

e quindi il sistema da risolvere è

$$\begin{cases} \frac{\partial}{\partial \boldsymbol{\alpha}_1} \mathfrak{L}(\boldsymbol{\alpha}_1, \lambda) = 2\mathbf{S_X}\boldsymbol{\alpha}_1 - 2\lambda\boldsymbol{\alpha}_1 = 0 \\ \frac{\partial}{\partial \lambda} \mathfrak{L}(\boldsymbol{\alpha}_1, \lambda) = \boldsymbol{\alpha}_1^T \boldsymbol{\alpha}_1 - 1 = 0 \end{cases} \tag{2.24}$$

È evidente che la prima delle (2.24) si può riscrivere come

$$(\mathbf{S_X} - \lambda \mathbf{I}_d)\boldsymbol{\alpha}_1 = 0, \tag{2.25}$$

dove $\mathbf{I}_d$ è la matrice identità $d \times d$. Ne segue che l'equazione (2.25) definisce il problema agli autovalori della matrice $\mathbf{S_X}$, in cui λ è un autovalore e $\boldsymbol{\alpha}_1$ il corrispondente autovettore. Inoltre, la varianza che si vuole massimizzare è $\boldsymbol{\alpha}_1^T \mathbf{S_X} \boldsymbol{\alpha}_1 = \lambda \boldsymbol{\alpha}_1^T \boldsymbol{\alpha}_1 = \lambda$.

Se ne deduce quindi che la prima componente principale $\mathbf{y}_1 = \mathbf{X}\boldsymbol{\alpha}_1$ risulta definita dall'autovettore $\boldsymbol{\alpha}_1$ corrispondente al più grande degli autovalori di $\mathbf{S_X}$, indicato con λ_1.

In generale,

Definizione 2.4. data la matrice di covarianza $\mathbf{S_X}$ associata all'insieme di dati $\mathbf{X}$, si definisce la k-esima componente principale

$$\mathbf{y}_k = \mathbf{X}\boldsymbol{\alpha}_k.$$

La varianza di $\mathbf{y}_k$ risulta essere

$$s(\mathbf{y}_k, \mathbf{y}_k) = \lambda_k,$$

dove λ_k è il k-esimo autovalore più grande di $\mathbf{S_X}$ e $\boldsymbol{\alpha}_k$ è il corrispondente autovettore.

Osservazione 3. Il numero massimo di componenti principali che si possono definire è pari al rango della matrice di covarianza $\mathbf{S_X}$.

Definizione 2.5. Date due generiche componenti principali $\mathbf{y}_j, \mathbf{y}_k$, il piano $(\mathbf{y}_j, \mathbf{y}_k)$ individuato da quest'ultime prende il nome di *piano principale*. In particolare, il piano individuato dalle prime due componenti principali $(\mathbf{y}_1, \mathbf{y}_2)$ prende il nome di *primo piano principale*.

Si conclude questa sezione con un teorema significativo che assicura la conservazione della varianza totale del campione di partenza, ossia che la varianza dei dati iniziali si ridistribuisca in quella delle componenti principali:

Teorema 2.2.1. *Dato un insieme di* n *elementi* d*-dimensionali, con matrice di covarianza* $\mathbf{S_X} = (s_{ij})_{1\leq i,j\leq d}$ *e componenti principali* $(\mathbf{y}_i)_{1\leq i\leq n}$ *risulta:*

$$\sum_{i=1}^{d} s_{ii} = \sum_{i=1}^{d} \lambda_i,$$

dove λ_i *è l'*i*-esimo autovalore della matrice* $\mathbf{S_X}$.

2.2.2 Proprietà algebriche e geometriche delle componenti principali

Si consideri l'equazione (2.21)

$$\mathbf{Y} = \mathbf{A}\mathbf{X}^T.$$

$\mathbf{A}$ è la matrice ortogonale la cui k-esima colonna, $\boldsymbol{\alpha}_k$, corrisponde al k-esimo autovettore di $\mathbf{S_X}$. Quindi le componenti principali possono essere definite da una trasformazione ortonormale lineare di $\mathbf{X}$.

Con un formalismo puramente matriciale, l'equazione (2.25) può essere riscritta come

$$\mathbf{S_X}\mathbf{A} = \mathbf{A}\boldsymbol{\Lambda}, \tag{2.26}$$

dove $\boldsymbol{\Lambda}$ è la matrice diagonale in cui l'elemento $\boldsymbol{\Lambda}_{kk}$ è il k-esimo autovalore della matrice $\mathbf{S_X}$, λ_k. Dall'ortogonalità di $\mathbf{A}$ si ottiene che

$$\mathbf{A}^T\mathbf{S_X}\mathbf{A} = \boldsymbol{\Lambda}, \tag{2.27}$$

e

$$\mathbf{S_X} = \mathbf{A}\boldsymbol{\Lambda}\mathbf{A}^T. \tag{2.28}$$

Vengono ora enunciate delle proprietà algebriche generali relative alle componenti principali.

Proprietà 1. *Per ogni $q \in \mathbb{N}$, $1 \leq q \leq d$, si consideri la trasformazione lineare ortonormale*

$$\mathbf{Y} = \mathbf{B}\mathbf{X}^T, \tag{2.29}$$

dove $\mathbf{Y}$ è un vettore q*-dimensionale e $\mathbf{B}$ è una matrice $q \times d$, e sia $\mathbf{S_Y} = \mathbf{B}\mathbf{S_X}\mathbf{B}^T$ la matrice di covarianza per $\boldsymbol{Y}$. Allora la traccia di $\mathbf{S_Y}$, denotata con $tr(\mathbf{S_Y})$, risulta massima se $\mathbf{B} = \mathbf{A}_q$, dove $\mathbf{A}_q$ è costituita dalle prime q colonne di $\mathbf{A}$.*

Proprietà 2. *Si consideri nuovamente la trasformazione ortonormale (2.29). Allora la* $\mathrm{tr}(\mathbf{S_Y})$ *risulta minima se $\mathbf{B} = \mathbf{A}_q^*$, dove $\mathbf{A}_q^*$ è costituita dalle ultime q colonne di $\mathbf{A}$.*

Le due proprietà enunciate confermano l'importanza del ruolo delle prime componenti principali rispetto alle ultime nella descrizione della variabilità di un insieme.

Proprietà 3. *(Decomposizione spettrale di $\mathbf{S_X}$)*

$$\mathbf{S_X} = \lambda_1 \boldsymbol{\alpha}_1 \boldsymbol{\alpha}_1^T + \lambda_2 \boldsymbol{\alpha}_2 \boldsymbol{\alpha}_2^T + \cdots + \lambda_d \boldsymbol{\alpha}_d \boldsymbol{\alpha}_d^T. \tag{2.30}$$

Proprietà 4. *Si consideri la trasformazione (2.29)*

$$\mathbf{Y} = \mathbf{B}\mathbf{X}^T.$$

Se si indica con $det(\mathbf{S_Y})$ il determinante della matrice di covarianza $\mathbf{S_Y}$, allora il $det(\mathbf{S_Y})$ è massimo se $\mathbf{B} = \mathbf{A}_q$.

Questo risultato è di fondamentale importanza statistica, in quanto il determinante della matrice di covarianza, detto *varianza generalizzata*, viene usato come misura dello spread per un qualsiasi vettore multivariato.

Proprietà 5. *Si supponga che si voglia determinare* $\mathbf{x}_j$ *in* $\mathbf{X}$ *mediante una funzione lineare di* $\mathbf{Y}$*, dove* $\mathbf{Y} = \mathbf{B}\mathbf{X}^T$*. Se* σ_j^2 *è la varianza nella determinazione di* $\mathbf{x}_j$ *da* $\mathbf{Y}$*, allora la* $\sum_{j=1}^{d} \sigma_j^2$ *è minima se* $\mathbf{B} = \mathbf{A}_q$*.*

Questo risultato dà anche una interpretazione geometrica delle componenti principali. Infatti, si supponga di voler ottenere la migliore determinazione di $\mathbf{X}$ in un sottospazio q-dimensionale, nel senso di minimizzare la somma della varianza su tutti gli elementi. Allora questo *sottospazio ottimale* è definito dalle prime q componenti principali.

La proprietà (5) ha dunque una duplice interpretazione, ovvero sia puramente algebrica che geometrica.

Viene enunciata ora un'importante proprietà geometrica che dà luogo ad una chiara interpretazione geometrica delle componenti principali.

Proprietà 6. *Si consideri la famiglia di ellissoidi* d*-dimensionali*

$$\mathbf{X}^T \mathbf{S}_\mathbf{X} \mathbf{X} = cost \tag{2.31}$$

Le componenti principali definiscono gli assi principali di questi ellissoidi.

La proprietà (6) risulta particolarmente interessante nel caso in cui la distribuzione dell'insieme $\mathbf{X}$ sia gaussiana. In questo caso, infatti, gli ellissoidi corrispondono a delle ipersuperfici di *probabilità costante* della distribuzione di $\mathbf{X}$.

2.2.3 Scelta del numero delle componenti principali

Nonostante l'osservazione (3) ci assicuri che è possibile trovare tante componenti principali quale è il rango di $\mathbf{S}_\mathbf{X}$, al fine di ottenere una riduzione di dimensionalità dell'insieme di partenza è necessario operare una scelta

delle componenti principali, che d'altra parte assicuri una minima perdita di informazione.

A tal fine,

Definizione 2.6. si definisce *fedeltà* della proiezione dei dati sul piano principale $(\mathbf{y}_j, \mathbf{y}_k)$ il rapporto

$$\frac{\lambda_j + \lambda_k}{\lambda_1 + \cdots + \lambda_d}.$$

Osservazione 4. La massima fedeltà si ottiene proiettando i dati sul primo piano principale.

Nella pratica, esistono tre *criteri euristici* principali adoperati per ridurre il numero delle componenti principali:

1. prendere solo quelle componenti che rappresentano l'80-90% della variabilità complessiva, calcolata con la fedeltà di più componenti.

2. Seguire la *regola di Kaiser*: prendere solo quelle componenti che hanno varianza maggiore di quella media (ottenuta come media dei λ_i).

3. Scegliere il numero di componenti attraverso il *grafico degli autovalori* o *Screen Plot*[1]. All'interno del grafico si sceglie il numero di componenti corrispondente al punto di gomito della spezzata.

La PCA in questo modo consente di controllare egregiamente il *trade off* tra la perdita di informazione e la semplificazione del problema.

[1]Lo *Screen Plot* è costruito ponendo sull'asse delle ascisse i numeri d'ordine degli autovalori e in ordinata gli autovalori ad essi corrispondenti. Un punto generico è quindi individuato dalla coppia (j, λ_j), e l'unione dei punti definisce una spezzata. Il numero di componenti principali sarà dato dal più piccolo numero d'ordine k tale che a sinistra di k l'andamento di λ_k sia fortemente decrescente, mentre a destra l'andamento di λ_{k+1} deve essere pressochè costante o comunque debolmente crescente.

2.2.4 Rappresentazione scale-free

Standardizzazione delle variabili d'origine

Spesso può accadere che i dati iniziali siano caratterizzati da unità di misura non paragonabili tra loro oppure da ampiezze molto diverse. In questi casi è necessario *standardizzare* le variabili aleatorie. A partire dai dati di origine, rappresentati dalla matrice $\mathbf{X}$, si calcola il valore medio di ogni variabile X_j

$$\overline{X}_{\cdot j} = \frac{1}{n}\sum_{i=1}^{n} X_{i,j}$$

e la deviazione standard

$$std_{\cdot j} = \sqrt{\sum_{i=1}^{n} \frac{1}{n-1}(X_{i,j} - \overline{X}_{\cdot j})^2}.$$

Si definiscono quindi le misure standardizzate

$$Z_{ij} = \frac{X_{ij} - \overline{X}_{\cdot j}}{std_{\cdot j}}$$

che possono essere rappresentate dalla matrice $\mathbf{Z}$. I dati così ottenuti hanno media nulla e varianza uguale ad 1, per cui sono tutti rappresentabili con numeri di grandezza comparabile.

Per il nuovo campione $\mathbf{Z}$ si trova che la matrice di covarianza coincide con la matrice di correlazione di $\mathbf{X}$, che è insensibile ai cambiamenti di scala. Pertanto, è opportuno effettuare sempre l'analisi delle componenti principali basandosi sulla matrice di correlazione invece che su quella di covarianza, in modo da evitare in ogni caso la standardizzazione delle variabili.

Whitened PCA

Un'alternativa alla standardizzazione delle variabili di origine è la procedura nota come *whitening* delle componenti principali. Essa consiste nel

normalizzare le componenti principali, cioè gli autovettori della matrice di covarianza (o di correlazione), dividendole per la radice quadrata dei rispettivi autovalori e le componenti principali così ottenute vengono dette *whitened*. Conseguentemente anche le matrici di covarianza o di correlazione risultano normalizzate ad 1.

2.3 Completamento dell'algoritmo

2.3.1 Visualizzazione dei cluster in due dimensioni

Dato l'insieme $\mathbf{X}$ di n elementi d-dimensionali, $\mathbf{x}_1, \ldots, \mathbf{x}_n$, la rappresentazione bidimensionale dell'insieme può essere ottenuta mediante l'analisi in componenti principali di $\mathbf{X}$ e la scelta di due delle componenti ottenute, in generale le prime due, indicate con $PC1$ e $PC2$, in quanto presentano la maggiore varianza.

Come precedentemente affermato, l'algoritmo di clustering analizzato permette la determinazione del potenziale $V(\mathbf{x})$ dell'equazione di Scrhödinger (2.16) e l'individuazione dei centri dei cluster mediante il calcolo dei minimi di $V(\mathbf{x})$, i quali dipendono dall'unico parametro nell'equazione (2.16), σ. Si osserva infatti in diversi insiemi di dati che la variazione di σ comporti una variazione nel numero di cluster individuati, in particolare al diminuire del parametro il numero di minimi di $V(\mathbf{x})$, e quindi di centri, aumenta fino a raggiungere il numero totale di elementi dell'insieme.

La scelta di σ è quindi di importanza cruciale affinché sia determinato il corretto numero di cluster, evitando allo stesso tempo che ci siano delle classificazioni errate.

Un metodo per la scelta della σ più adatta consiste nel variare il parametro con continuità fino a raggiungere la stabilità tra numero di cluster e classifi-

cazioni errate. D'altra parte, spesso si parte dall'analisi dell'insieme con una σ relativamente grande, cioè da un unico grande cluster, e la si fa decrescere fino a che il criterio precedentemente affermato è soddisfatto.

Fissata σ, è possibile ottenere una visualizzazione quantitativa della classificazione mediante la rappresentazione di $V(\mathbf{x})$ con curve di livello o con un grafico tridimensionale.

2.3.2 Complessità computazionale

Si consideri la funzione d'onda $\psi(\mathbf{x})$ data dall'equazione (2.15). Se si calcola questa funzione in un punto generico dello spazio, $\mathbf{x}$, saranno effettuate n operazioni, quindi se si vuole mappare la funzione su di una griglia $m \times m$ nello spazio saranno necessarie $n \times m^2$ operazioni. Per valori elevati di m si riscontra quindi un notevole incremento di complessità computazionale del problema e una forte limitazione dell'applicabilità del calcolo numerico. La risoluzione dei valori della funzione nello spazio infatti dipende solo da m.

Un problema analogo si riscontra con il calcolo del potenziale $V(\mathbf{x})$, o di qualsiasi altra funzione definita nella procedura. Inoltre, spesso è necessario, per una maggiore accuratezza, introdurre ulteriori componenti principali nel calcolo del potenziale.

Una soluzione a tale inconveniente si ottiene calcolando il potenziale solo nei punti dell'insieme in questione. In seguito, il calcolo dei minimi del potenziale ottenuto sarà effettuato partendo dai punti dell'insieme stesso. In questo modo, detta d la dimensione degli elementi dell'insieme, la complessità computazionale è fissata a $n^2 \times d$ operazioni.

Un'ulteriore riduzione della complessità computazionale può essere ottenuta avendo a disposizione i dati nella rappresentazione di una matrice di prossimità. Detto, infatti, $V_i \equiv V(\mathbf{x}_i)$ e $D_{ij} = d(\mathbf{x}_i, \mathbf{x}_j)$ la distanza tra

l'i-esimo e il j-esimo punto dell'insieme, risulta

$$V_i = E - \frac{d}{2} + \frac{1}{2\sigma^2}\frac{\sum_i D_{ij}^2 e^{-\frac{D_{ij}^2}{2\sigma^2}}}{\sum_i e^{-\frac{D_{ij}^2}{2\sigma^2}}}, \tag{2.32}$$

dove E è determinata dalla condizione $\min V_i = 0$.

2.4 Algoritmo di classificazione

Segue una breve descrizione del metodo di ottimizzazione del gradiente, utilizzato per la ricerca dei minimi del potenziale dell'equazione (2.16).

2.4.1 Metodo del gradiente

Sia $F(\mathbf{x})$ una funzione definita in un aperto Ω di $\mathbb{R}^n$ e di classe $\mathbf{C}^2$. Il gradiente di F è definito come

$$\nabla F(\mathbf{x}) = (\partial_1 F(\mathbf{x}), \ldots, \partial_n F(\mathbf{x})) \tag{2.33}$$

Fissato un punto $\mathbf{x} \in \Omega$, la retta passante per $\mathbf{x}$ e avente la direzione del gradiente in $\mathbf{x}$ è data in forma parametrica da

$$y = \mathbf{x} - \gamma(\nabla F)_{\mathbf{x}}, \tag{2.34}$$

con $\gamma \in \mathbb{R}$.

Se si considera $F(\mathbf{x} - \gamma(\nabla F)_{\mathbf{x}})$, per γ sufficientemente piccolo è possible effettuare lo sviluppo in serie di F in $\mathbf{x}$, per cui risulta

$$F(\mathbf{x} - \gamma(\nabla F)_{\mathbf{x}}) = F(\mathbf{x}) - \gamma(\nabla F)_{\mathbf{x}}^2 + O(\gamma^2), \tag{2.35}$$

e conseguentemente

$$F(\mathbf{x} - \gamma(\nabla F)_{\mathbf{x}}) \leq F(\mathbf{x}). \tag{2.36}$$

La variazione di $F(\mathbf{x})$ lungo una direzione individuata dal vettore unitario $\mathbf{t}$, $|\mathbf{t}|^2 = 1$, è

$$\Delta F = \mathbf{t} \cdot \nabla F, \tag{2.37}$$

ma il prodotto scalare $\mathbf{t} \cdot \nabla F$ è massimo in valore assoluto quando $\mathbf{t}$ è parallelo a ∇F. Pertanto, la direzione individuata da $-\nabla F$ si dice della *discesa più ripida.*

Si consideri ora un'iterazione del tipo

$$\mathbf{x}_{n+1} = \mathbf{x}_n - \gamma(\nabla F)_{\mathbf{x}_n} \tag{2.38}$$

Se per $m \in \mathbb{N}$ lo spostamento di $\mathbf{x}_n$, $\gamma(\nabla F)_{\mathbf{x}_n}$, diventa inferiore ad una soglia fissata, si può assumere che il valore $\mathbf{x}_m$ sia un punto di minimo per la funzione F.

Verrà ora illustrato l'algoritmo di quantum clustering utilizzato per la realizzazione della procedura in appendice.

2.4.2 L'algoritmo

1. Dato l'insieme $\mathbf{X}$ e fissata σ, si determini la funzione del potenziale $V(x)$ come in (2.18).

2. Si calcoli per ogni punto di $\mathbf{X}$ il gradiente di V, ∇V.

3. Mediante il metodo del gradiente, si faccia scivolare ogni punto verso la buca del potenziale, di una quantità pari a $\gamma(\nabla V)_{\mathbf{x}}$.

4. Si ripetano gli step 2 e 3 fino a che non vi sia alcuna variazione di x o la variazione sia inferiore ad una soglia data.

Capitolo 3

Applicazioni del QC

3.1 Iris

3.1.1 Descrizione dell'insieme

L'*Iris data set* (o insieme degli iris di Fisher) è un insieme di vettori multivariati introdotto da Ronald Aylmer Fisher [9] nel 1936 come un esempio di analisi discriminante. Esso è spesso chiamato anche insieme degli iris di Anderson in quanto Edgar Anderson nel 1935 raccolse i dati relativi a queste piante per quantificarne la variazione geografica nel Gaspé, una penisola ad est del Quebec, in Canada [1].

I pattern del presente data set descrivono fiori di iris, in particolare vengono specificate la *lunghezza* e la *larghezza* dei *sepali*, ovvero i costituenti del calice del fiore, e dei *petali*, per un totale di quattro attributi.

Il data set contiene 150 esempi, ciascuno descritto dal vettore delle suddette quattro feature.

Sono presenti tre categorie, *Iris Setosa*, *Iris Versicolor* e *Iris Virginica*, ciascuna rappresentata nel data set da 50 pattern.

Nel presente lavoro è stato utilizzato sia il data set con i pattern etichettati, per la visualizzazione dei dati di partenza, sia l'insieme i cui pattern non sono etichettati, in modo da effettuare una classificazione *ex novo*.

3.1.2 Rappresentazione dell'insieme

La rappresentazione dell'insieme di *Iris* è stata ottenuta mediante la visualizzazione dei dati nelle prime due componenti principali.

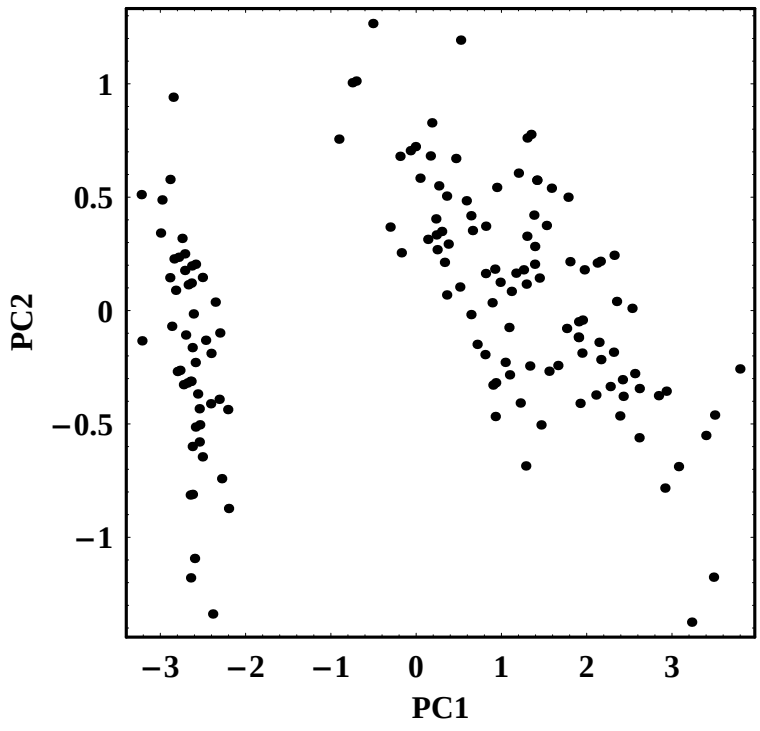

Figura 3.1: Rappresentazione del data set *Iris* nelle prime due componenti principali, *PC1* e *PC2*.

Si osservano in questa rappresentazione due gruppi di punti ben distinti, di cui il secondo ha una varianza nettamente superiore rispetto al primo. Ciò è dovuto al fatto che due delle tre famiglie di iris sono piuttosto simili tra

loro. Per verificare quanto detto è stato rappresentato, sempre nelle prime due componenti principali, l'insieme con i dati etichettati, in modo tale da poter distinguere le tre famiglie con colori diversi.

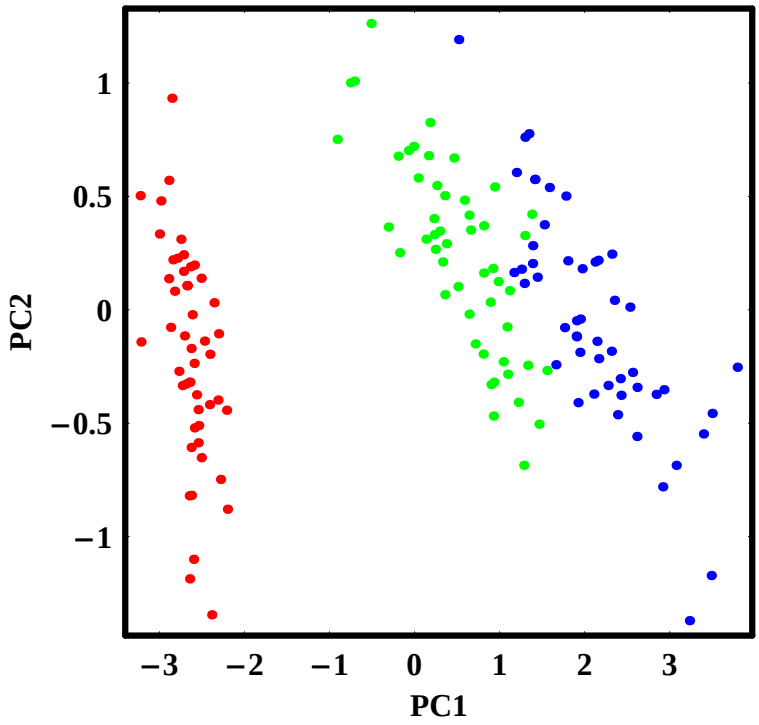

Figura 3.2: Rappresentazione delle tre famiglie di iris presenti nel data set. L'*iris setosa* corrisponde al colore rosso, ben distinta dalle famiglie *versicolor* e *virginica*, rispettivamente individuate dai colori verde e blu.

In questa figura è evidente la compenetrazione visiva delle due serie di dati.

3.1.3 Determinazione del potenziale

Per la determinazione del potenziale dell'equazione di Schrödinger (2.16) è stato in primo luogo determinato il valore di σ ottimale per l'individuazione

dei cluster. Per questo motivo è stato rappresentato il potenziale nel primo piano principale con delle curve di livello per diversi valori di σ ed è stato scelto come valore migliore del parametro $\sigma = 0.5$, corrispondente ad un contour plot con delle curve di livello ben definite intorno a tre minimi.

In figura (3.3) si osserva accanto ad ogni contour plot un grafico di V/E in funzione del numero di identificazione di ciascun pattern. I pattern con valori elevati di V/E corrispondono agli elementi dell'insieme più esterni nella rappresentazione in componenti principali, e quindi gli elementi che provocano maggior *noise* nella ricerca dei centri dei cluster. Per questo nella procedura di minimizzazione del potenziale viene introdotta una variabile di taglio, t, che indica il valore massimo di V/E che deve essere assunto da un elemento per essere considerato nella procedura.

Si introducono, per completezza, in figura (3.4) i risultati relativi all'insieme dei dati di *Iris* etichettati.

Fissata σ, il potenziale ottenuto è quello rappresentato in figura (3.5).

3.1.4 Risultati della classificazione

Utilizzando la procedura con il metodo del gradiente sulla prima e seconda componente principale, si ottengono cinque cluster, in disaccordo con la classificazione vera dell'insieme. Ciò è dovuto alla presenza di minimi locali dovuti a punti o gruppi di punti isolati, o distanti dai centri dei cluster.

Per ovviare a questo problema, come già accenato, si può inserire nella procedura la possibilità di effettuare un taglio sul potenziale, in particolare su V/E.

In questo modo si osserva infatti che con un taglio di $V/E = 0.43$, ovvero considerando solo i punti con un potenziale inferiore a $0.43E$, si ottengono i tre cluster attesi, rappresentati insieme al potenziale in figura (3.9).

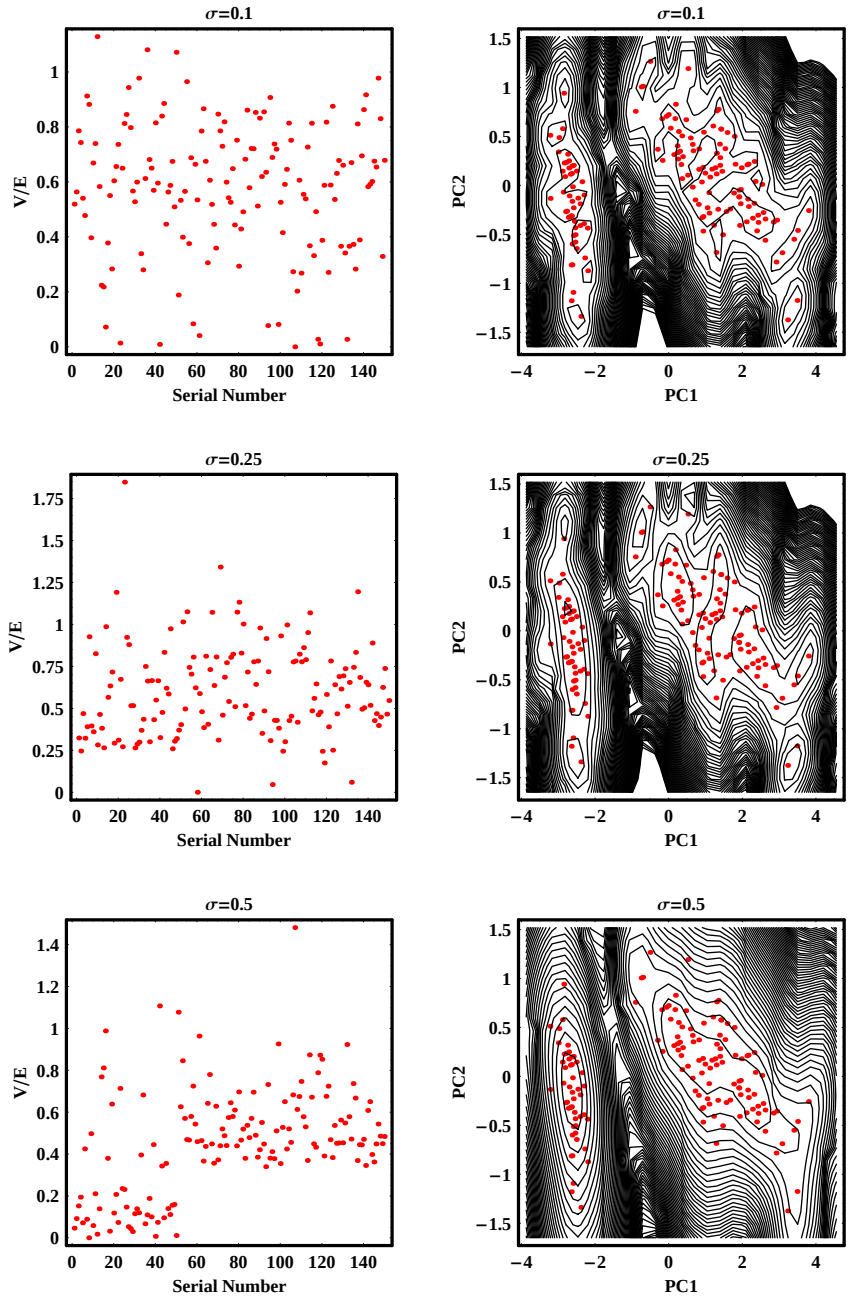

Figura 3.3: Valori di $V(\mathbf{x})/E$ in funzione del *serial number* di *Iris* e rappresentazione del potenziale per tre diversi valori di σ, $\sigma = 0.1, 0.25, 0.5$.

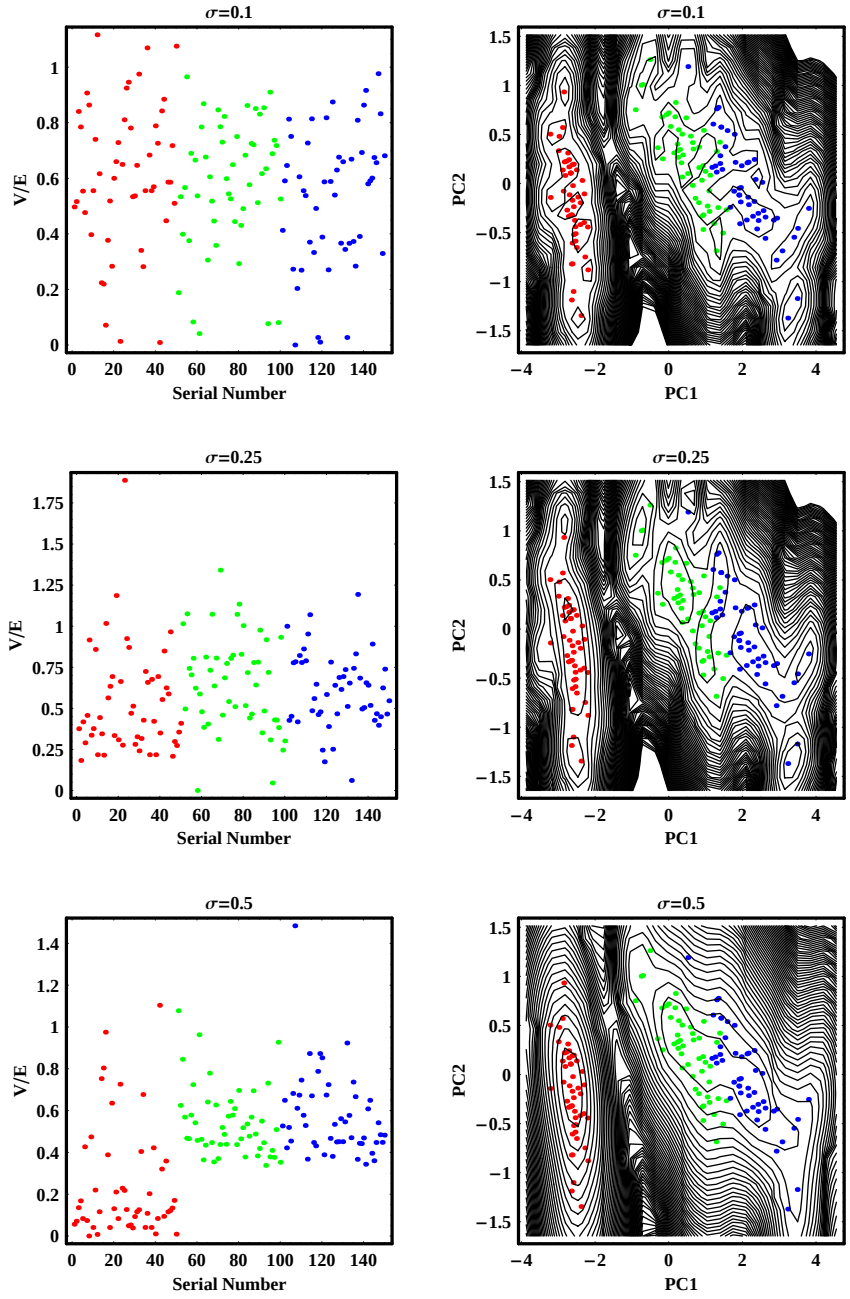

Figura 3.4: Valori di $V(\mathbf{x})/E$ in funzione del *serial number* di *Iris* con classificazione nota e rappresentazione del potenziale per tre diversi valori di σ, $\sigma = 0.1, 0.25, 0.5$.

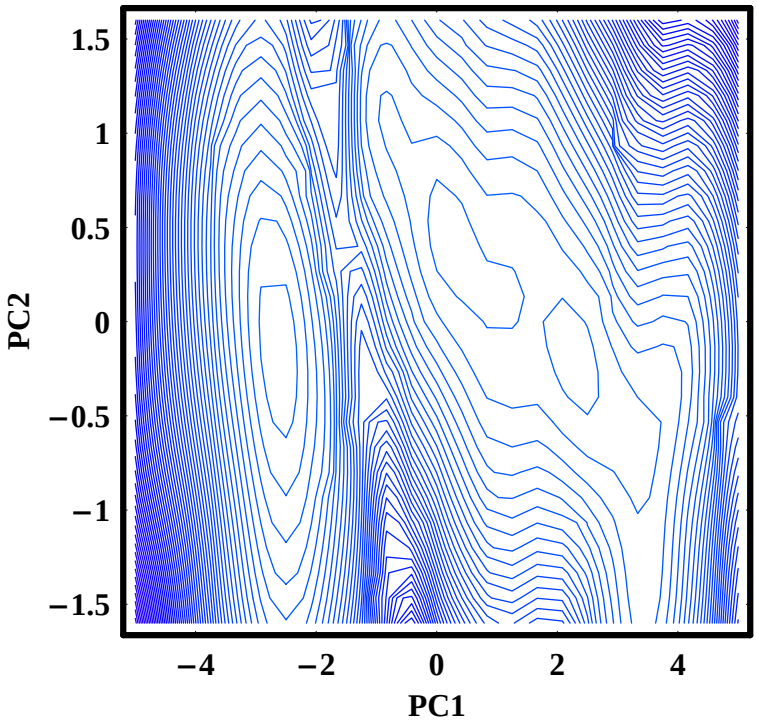

Figura 3.5: Curve di livello del potenziale $V(\mathbf{x})$ ottenuto per $\sigma = 0.5$.

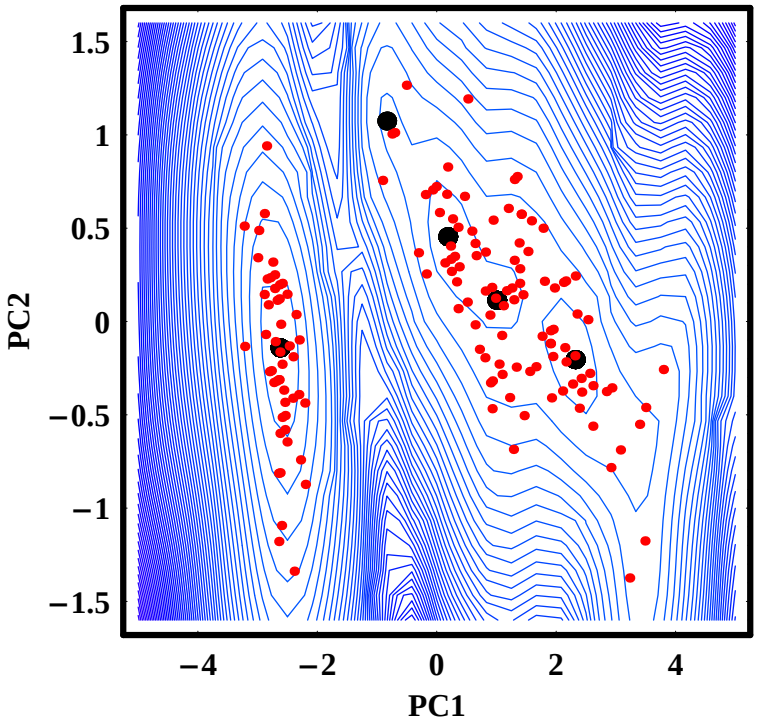

Figura 3.6: Punti di *Iris* soggetti al potenziale $V(\mathbf{x})$. I punti neri rappresentano i minimi di $V(\mathbf{x})$ e dunque i centri dei cluster.

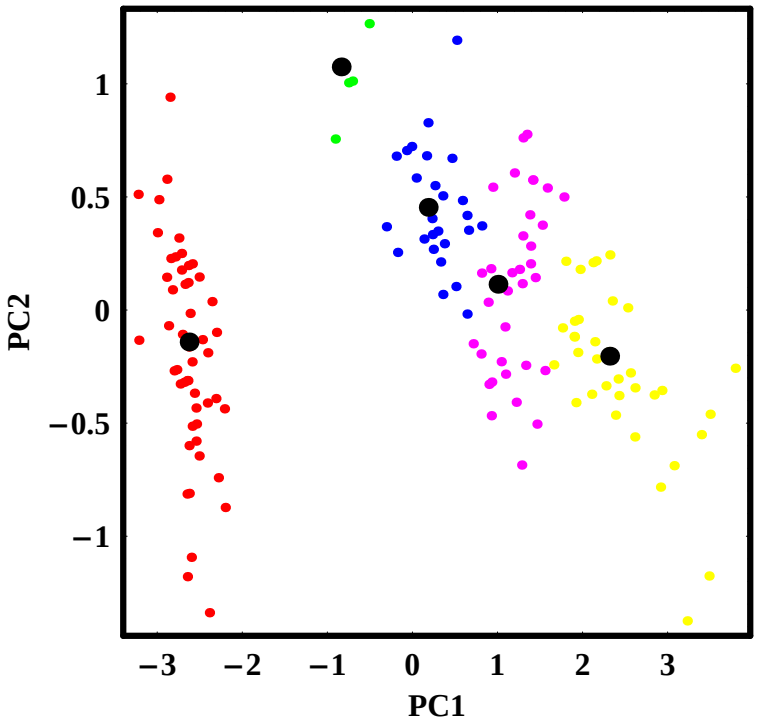

Figura 3.7: Cluster di *Iris* ottenuti con $\sigma = 0.5$.

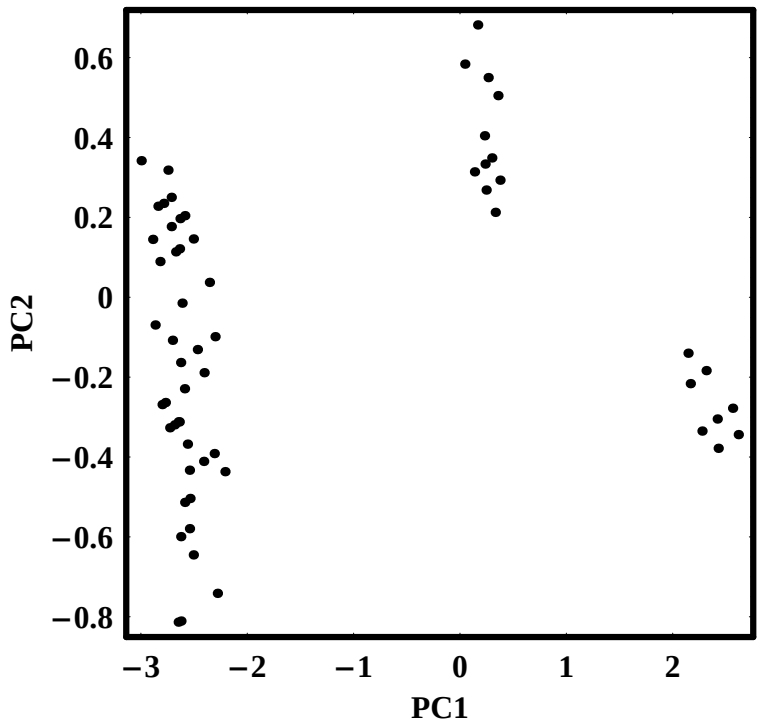

Figura 3.8: Visualizzazione in componenti principali di *Iris* a cui è stato applicato un taglio di $V/E = 0.43$.

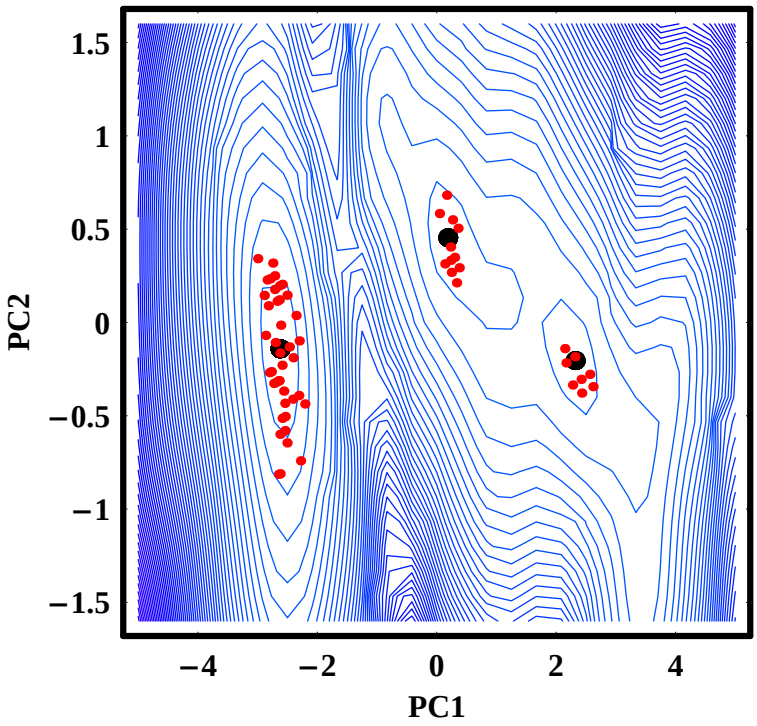

Figura 3.9: Potenziale di *Iris* (taglio di $V/E = 0.43$) con rappresentazione dei punti e dei tre centri dei cluster trovati.

Per osservare la classificazione di tutti i punti dell'insieme, è stata definita un ulteriore funzione all'interno della procedura che associa ogni punto con $V/E > 0.43$ al cluster con il centro più vicino. In questo caso è stata utilizzata, per semplicità e rapidità di calcolo, la distanza euclidea.

Si giunge così ad una classificazione piuttosto corretta, con 12 classificazioni errate, 6 nel secondo e 6 nel terzo cluster. L'errore percentuale della classificazione è dunque pari all'8%.

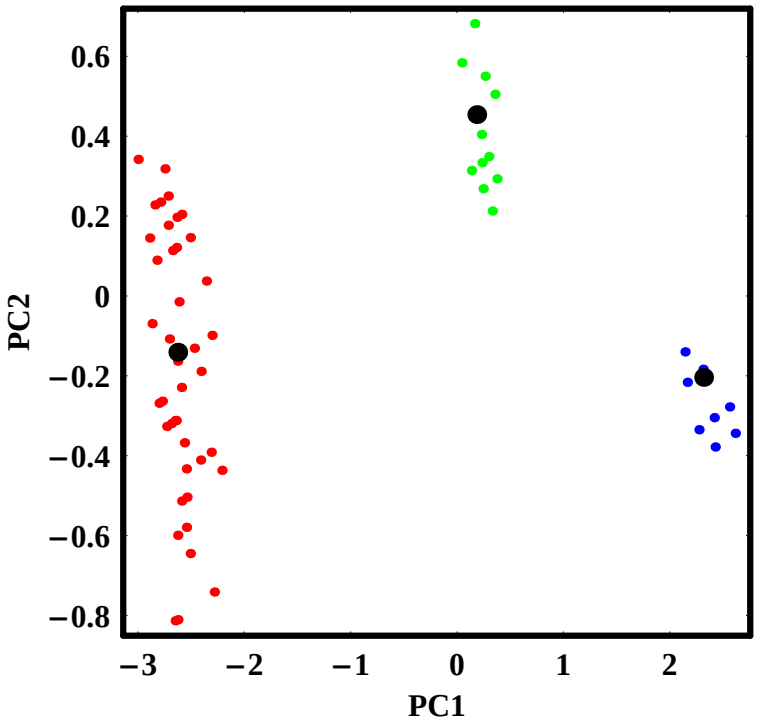

Figura 3.10: Cluster di *Iris* (taglio di $V/E = 0.43$).

Se si considerano, invece, tutti i punti si trova che è possibile ottenere tre cluster per $\sigma = 0.62$. In questo caso, però, si nota la presenza di 18 classificazioni errate, quindi un errore del 12%, superiore a quello trovato con il metodo precedente, per $\sigma = 0.5$.

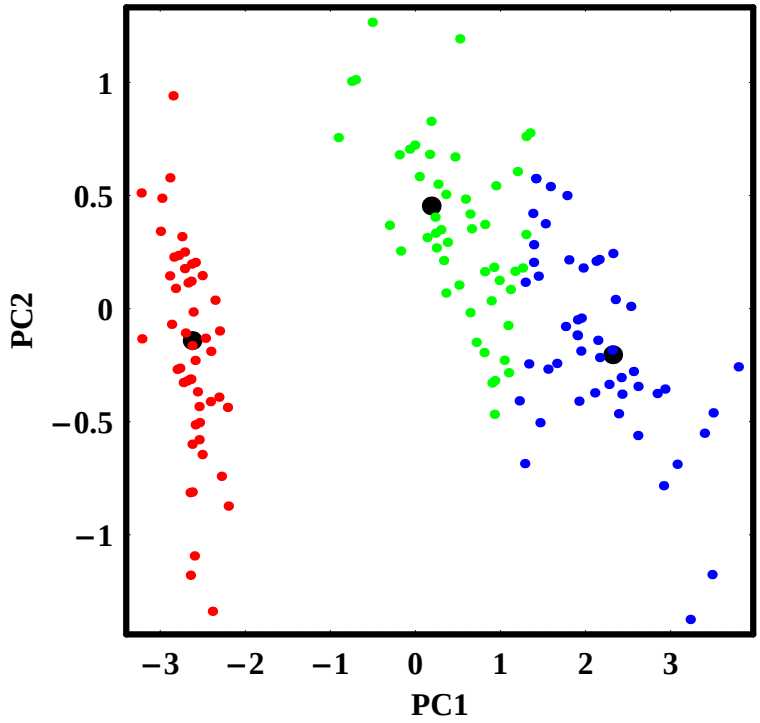

Figura 3.11: Cluster con tutti i punti di *Iris* (taglio di $V/E = 0.43$). I punti con $V/E \geq 0.43$ vengono associati ai cluster in cui la cui distanza euclidea dai centri e minima.

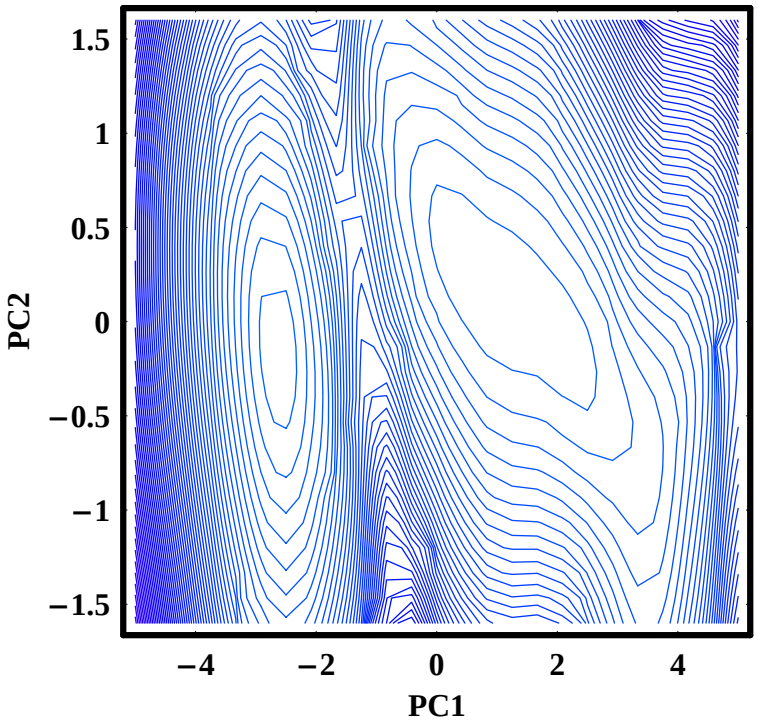

Figura 3.12: Curve di livello del potenziale $V(\mathbf{x})$ ottenuto per $\sigma = 0.62$.

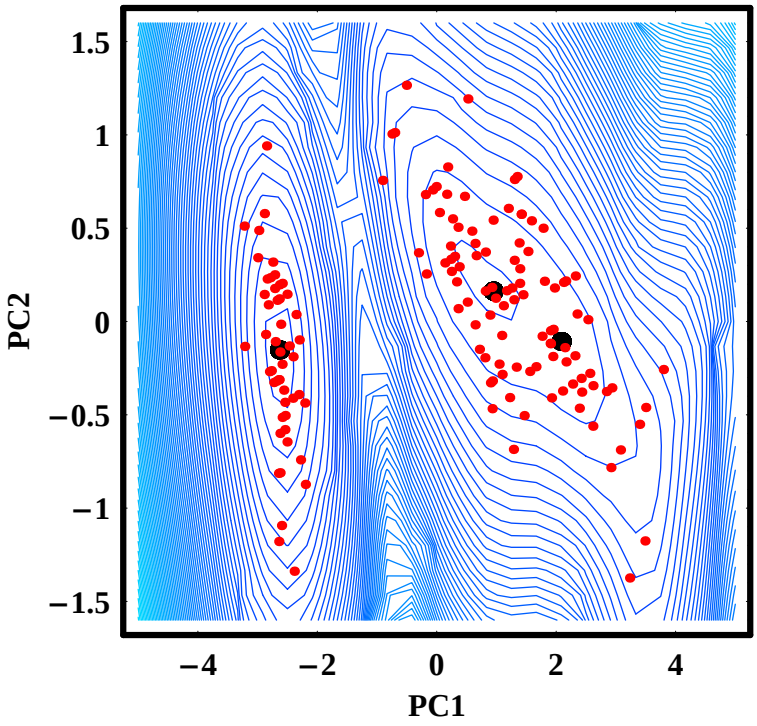

Figura 3.13: Punti di *Iris* soggetti al potenziale $V(\mathbf{x})$. I punti neri rappresentano i minimi di $V(\mathbf{x})$ e dunque i centri dei cluster.

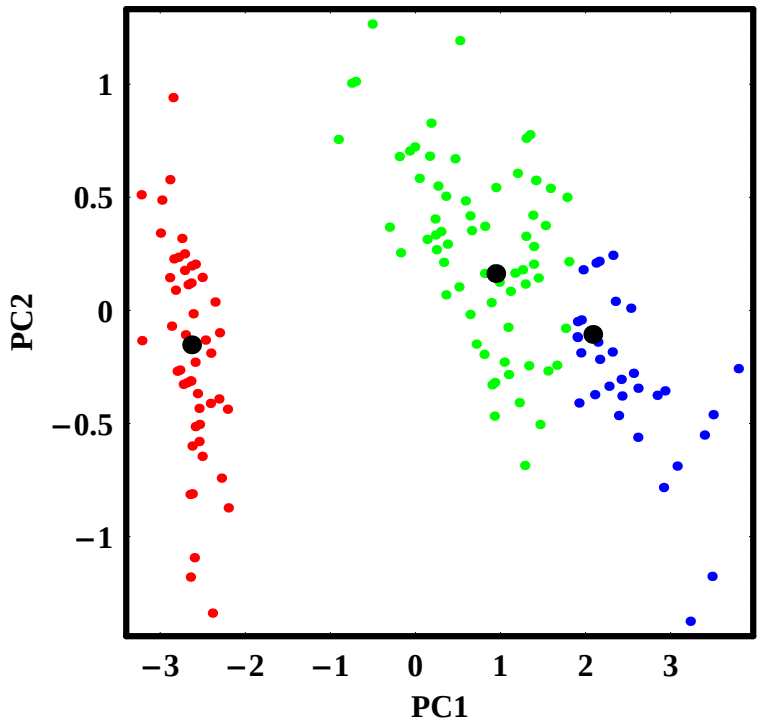

Figura 3.14: Cluster di *Iris* con $\sigma = 0.62$. Le 18 classificazioni errate si trovano nei cluster individuati dai colori verde e blu.

3.2 Escherichia Coli

3.2.1 Descrizione dell'insieme

L'*E. Coli* data set è stato introdotto nel 1996 da K. Nakai e P. Horton. Nel loro articolo [11] [12] viene presentato un programma, *PSORT*, atto alla predizione della localizzazione subcellulare di proteine. In particolare, è stato analizzato l'insieme di *E.Coli*, ottenuto scegliendo 336 sequenze note di siti di localizzazione subcellulare nel batterio *Escherichia Coli* dal *PROSITE* database.

Il data set analizzato è un insieme di 336 esempi descritti da sette feature.

I primi tre attributi di ciascun pattern sono delle applicazioni di metodi per il riconoscimento dei segnali di ciascuna sequenza, di cui due sono variabili reali ed una binaria. Il quarto attributo è una variabile binaria relativa alla presenza di carica sui siti delle lipoproteine predette. Il quinto è il risultato dell'analisi discriminante degli amminoacidi delle proteine della membrana esterna. Gli ultimi due attributi sono i risultati di due diverse applicazioni di uno stesso programma ($ALOM$) alle sequenze dei siti di localizzazione.

3.2.2 Rappresentazione dell'insieme e determinazione del potenziale

Analogamente al caso di *Iris*, anche nel caso di *E.Coli* la rappresentazione è stata ottenuta con l'individuazione dei punti nel piano principale. In questo caso sono ben distinti tre agglomerati di punti di dimensione differente.

Per la scelta di sigma sono stati utilizzati i valori $\sigma = 0.01, 0.1, 0.2$, osservando che le curve di livello sono ben definite e approssimano la suddetta classificazione in tre agglomerati per $\sigma = 0.2$.

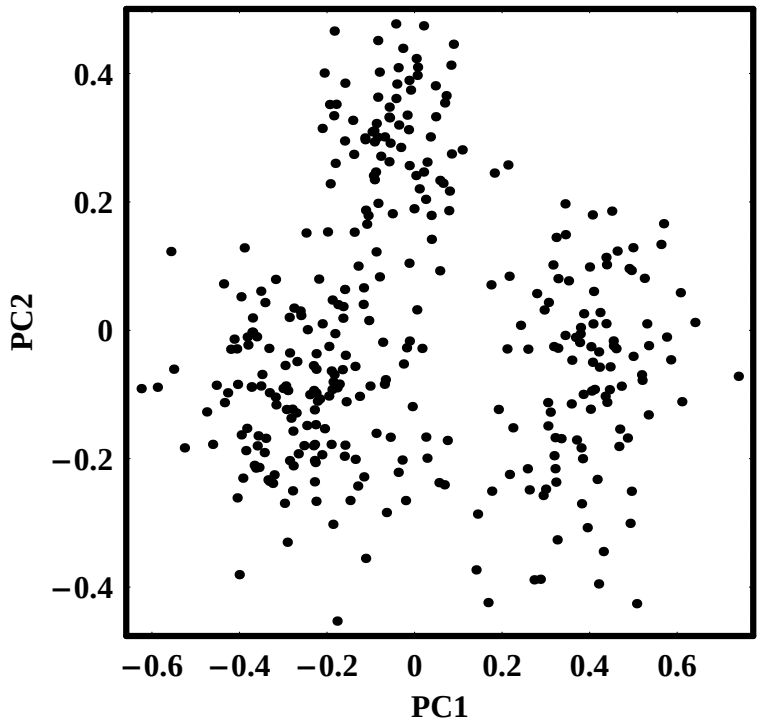

Figura 3.15: Rappresentazione del data set *E.Coli* nelle prime due componenti principali, *PC1* e *PC2*.

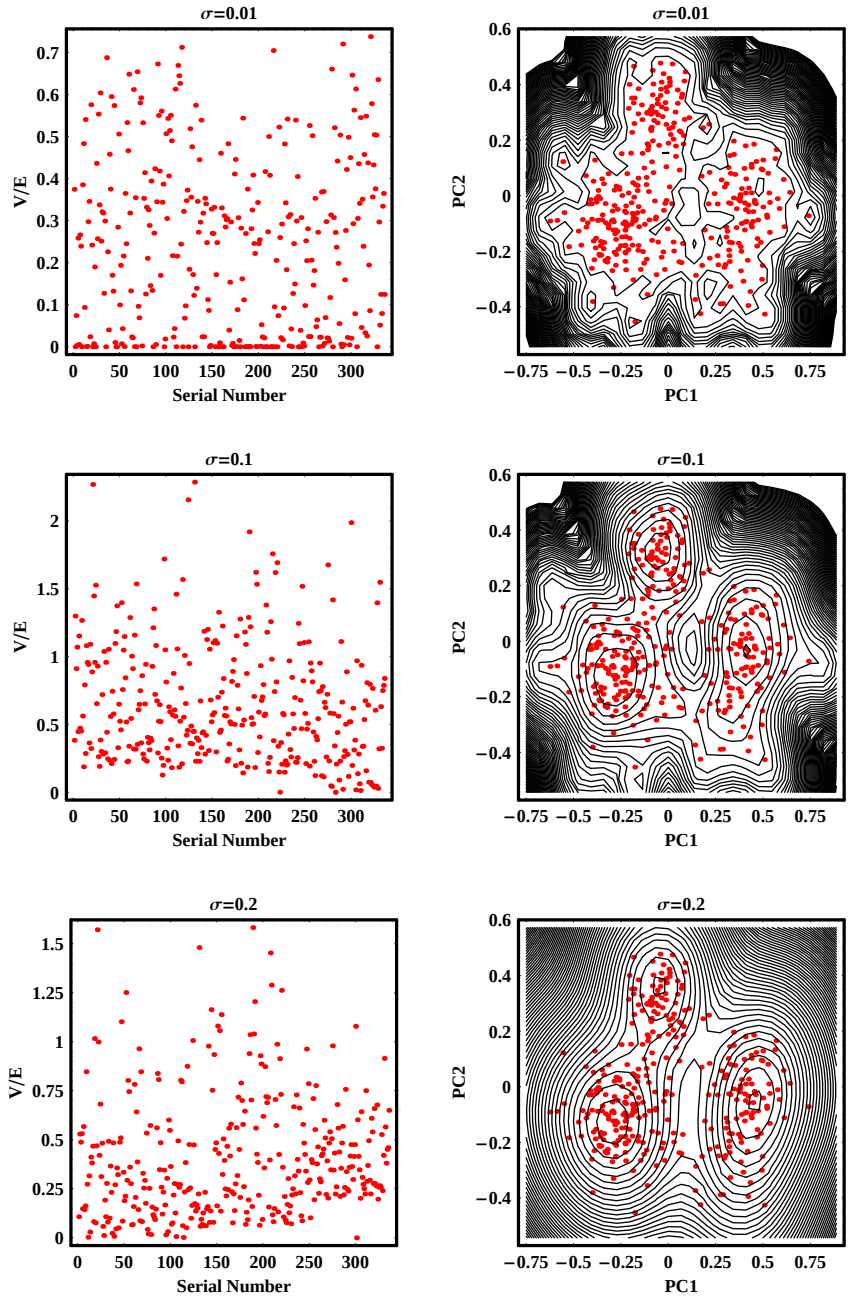

Figura 3.16: Valori di $V(\mathbf{x})/E$ in funzione del *serial number* di *E.Coli* e rappresentazione del potenziale per tre diversi valori di σ, $\sigma = 0.01, 0.1, 0.2$.

Fissata $\sigma = 0.2$, il potenziale ottenuto per l'insieme *E.Coli* è quello rappresentato dalle curve di livello in figura (3.17).

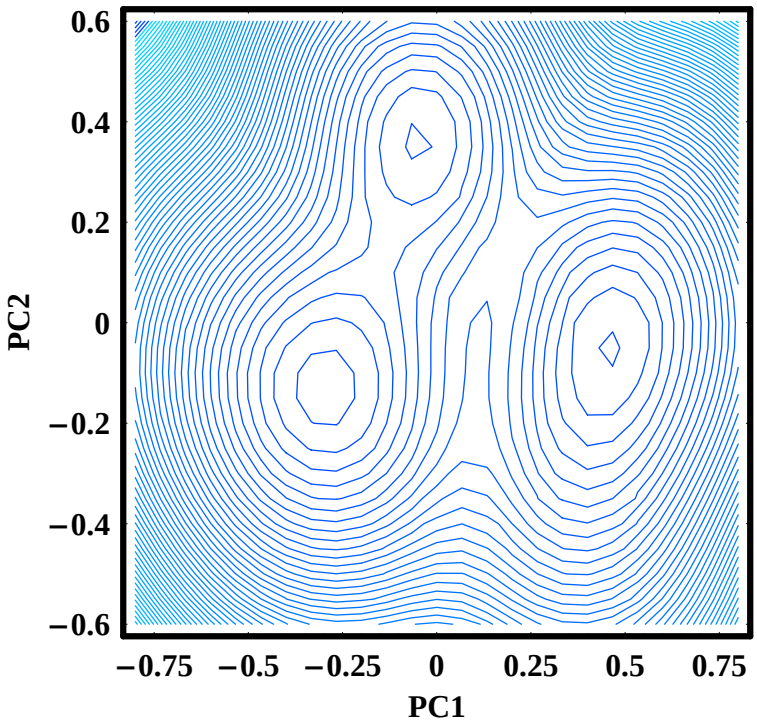

Figura 3.17: Curve di livello del potenziale $V(\mathbf{x})$ ottenuto per $\sigma = 0.2$.

3.2.3 Risultati della classificazione

Associando ogni punto dell'insieme al centro verso cui il punto stesso tende con il metodo del gradiente, sono stati determinati i tre cluster definitivi dell'insieme. In questo caso una verifica della classificazione non è possibile in quanto i dati di *E.Coli* non sono etichettati.

La procedura permette la visualizzazione dei cluster con colori diversi, riportata in figura (3.19).

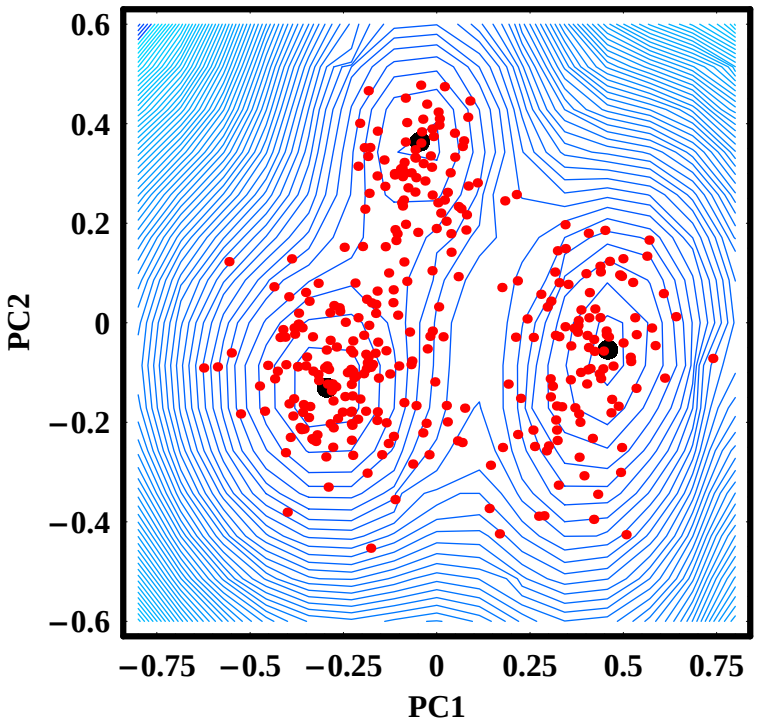

Figura 3.18: Punti di *E.Coli* soggetti al potenziale $V(\mathbf{x})$. I punti neri rappresentano i minimi di $V(\mathbf{x})$, ovvero i centri dei cluster.

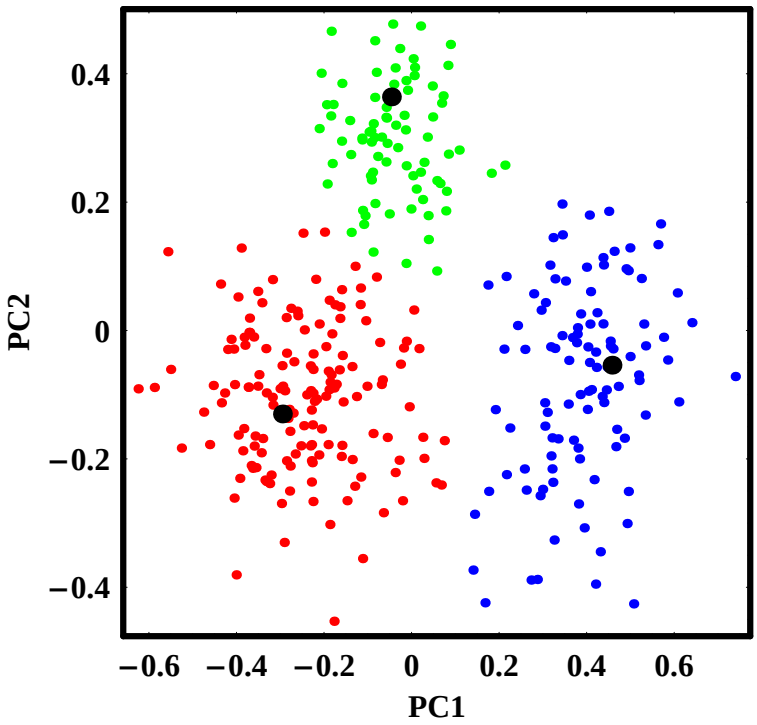

Figura 3.19: Cluster dell'insieme *E.coli*. I punti neri corrispondono ai centri dei cluster.

Conclusioni

Dall'analisi condotta sull'algoritmo di Horn e Gottlieb basato sulla meccanica quantistica si evince la versatilità di tale algoritmo. Infatti esso può essere applicato a qualsiasi insieme di dati considerando la funzione di distribuzione come la funzione d'onda di un sistema fisico, e quindi descritto dall'equazione di Schrödinger. Quest'ultima, come si è visto, contiene un solo parametro, la σ, quindi è possibile, solo variando quest'ultima, ottenere le diverse soluzioni del potenziale, ovvero le possibili classificazioni.

Rispetto alle tecniche di clustering gerarchico, questo algoritmo consente una più rapida individuazione dei cluster, grazie alla ridotta complessità computazionale. In questo caso, infatti, non è necessario calcolare tutte le possibili classificazioni al variare del livello di dissomiglianza, ma si ha per una fissata σ la corrispondente classificazione naturale dei dati.

D'altra parte, in analogia con il concetto di dendrogramma, in questo caso è possibile effettuare un plot del numero di cluster in funzione della σ, ottenendo una struttura ad albero in cui però sono possibili variazioni di appartenenza dei diversi punti con il variare del parametro, ovvero è possibile che un punto cambi cluster. Questo si ripercuoterebbe nel diagramma con la comparsa, ad una certa σ, di linee trasversali rispetto alla struttura piramidale.

Rispetto alle più comuni tecniche di clustering partizionale, come il k-

means, l'algoritmo risulta anche in questo caso più comodo in quanto non è necessario fissare a priori il numero di cluster che si vogliono ottenere; ciò però a discapito di una minore velocità di esecuzione e quindi di una maggiore complessità computazionale.

L'algoritmo del quantum clustering dando luogo a *hard clusters*, non è adatto per la descrizione di insiemi che prevedono un grado di appartenenza per i punti dei diversi cluster, ovvero quando per la descrizione dell'insieme è necessaria la logica fuzzy.

L'originalità di questo algoritmo sta principalmente nell'astrazione della funzione di distribuzione a ground state dell'equazione di Schrödinger, quindi qualsiasi insieme di dati può essere trattato come se fosse un sistema fisico descritto sempre dalla stessa equazione, con l'unica variazione nel parametro legato alla scala del problema.

Non si riscontrano, ancora, al giorno d'oggi delle applicazioni rilevanti dell'algoritmo studiato, ma, secondo l'autore del presente lavoro di tesi, il contributo maggiore sta proprio nell'aver proposto una nuova metodologia di clustering che esce dagli schemi della pura programmazione informatica e permette un sincretismo interdisciplinare, alla base dello sviluppo della conoscenza umana.

Appendice A

Procedura di Quantum Clustering sviluppata in Mathematica

```
Off[General::spell1];
Off[General::spell];
Needs[Graphics`Colors`];
Needs[Graphics`Graphics`];
Needs[Statistics`MultiDescriptiveStatistics`];
Clear[Global`*];

dir = C:\\Clustering\\;

H[x_List, ξ_, σ_]:=
Module[{n, d},
{n, d} = Dimensions[ξ];
-d/2 + Σ_{i=1}^{n} (x-ξ[[i]]).(x-ξ[[i]]) e^{-(x-ξ[[i]]).(x-ξ[[i]])/(2σ^2)} / (2σ^2 Σ_{i=1}^{n} e^{-(x-ξ[[i]]).(x-ξ[[i]])/(2σ^2)})
];
```

```
En[ξ_, σ_]:=En[ξ, σ] =
Module[{k, Ht},
Ht = Table[H[ξ[[j]], ξ, σ], {j, Length[ξ]}];
k = Ordering[Ht, 1][[1]];
{−Ht[[k]], ξ[[k]], Ht}
];

ReadDataSet[file_]:=
Module[{m, n, v, c, ξ},
{m, n, v, v, v} = Read[file, Table[Number, {5}]];
ξ = Read[file, Table[Number, {n}, {m + 1}]];
Close[file];
{Round[Transpose[ξ][[m + 1]]], Transpose[Transpose[ξ][[Range[m]]]]}
];

V[x_List, ξ_, σ_]:=En[ξ, σ][[1]] + H[x, ξ, σ]/;
NumberQ[x/.List → Plus];

Colors = {Red, Green, Blue, Magenta, Yellow, Orange, Black};

ShowDataSet[ξ_, c_, σ_]:=
Module[{fig = {}, m, n, k = Length[σ], PC, x11, x12, x21, x22,
Vt, xmin, Ht, i, f0, f1},
{n, m} = Dimensions[ξ];
PC = If[m > 2, Transpose[Transpose[PrincipalComponents[ξ]][[{1, 2}]]],
ξ];
{x11, x12} = {Min[Transpose[PC][[1]]], Max[Transpose[PC][[1]]]};
```

```
{x11, x12} = {If[x11 < 0, 1.2x11, 0.8x11], If[x12 < 0, 0.8x12, 1.2x12]};
{x21, x22} = {Min[Transpose[PC][[2]]], Max[Transpose[PC][[2]]]};
{x21, x22} = {If[x21 < 0, 1.2x21, 0.8x21], If[x22 < 0, 0.8x22, 1.2x22]};
Do[
{Et, xmin, Ht} = En[PC, σ[[i]]];
Vt = Transpose[{Range[Length[PC]], 1 + Ht/Et}];
AppendTo[fig,
Show[Graphics[{PointSize[0.015],
Flatten[Transpose[{Colors[[c]], Map[Point, Vt]}]]}],
Frame → True, PlotLabel → σ= <> ToString[σ[[i]]],
FrameLabel → {Serial Number, V/E}, DisplayFunction → Identity,
PlotRange → All, AspectRatio → 1]];
f0 = Graphics[{PointSize[0.015],
Flatten[Transpose[{Colors[[c]], Map[Point, PC]}]]}];
f1 = ContourPlot[H[{x1, x2}, PC, σ[[i]]], {x1, x11, x12},
{x2, x21, x22}, ContourShading → False, Contours → 60,
DisplayFunction → Identity];
AppendTo[fig, Show[f0, f1, Frame → True,
PlotLabel → σ= <> ToString[σ[[i]]],
FrameLabel → {PC1, PC2}, AspectRatio → 1,
DisplayFunction → Identity]],
{i, Length[σ]}
];
Show[GraphicsArray[Partition[fig, 2]],
DisplayFunction → $DisplayFunction, ImageSize → 300{2, Length[σ]}]
];
```

```
ClusterCenter[ξ_, σ_, c_, t_:0.4]:=
Module[{Et, xmin, Ht, Vt, Y, ind, x, var, sol, j, PC, CC},
{n, d} = Dimensions[ξ];
{Et, xmin, Ht} = En[ξ, σ];
Vt = 1 + Ht/Et;
Y = Select[Vt, # < t&];
ind = Ordering[Vt, Length[Y]];
x = ξ[[ind]];
PC = If[d > 2, Transpose[Transpose[PrincipalComponents[ξ]][[{1, 2}]]],
ξ];
Show[
Graphics[{PointSize[0.015],
Flatten[Transpose[{Colors[[c[[ind]]]], Map[Point, PC[[ind]]]}]]},
Axes → True];
var = Table[a[i], {i, d}];
CC =
Table[sol = FindMinimum[V[var, ξ, σ], Transpose[{var, x[[j]]}],
Method → Gradient]; var/.sol[[2]], {j, Length[x]}];
Union [CC, SameTest → ((#1 − #2).(#1 − #2) < 10^−6&)]
];

Moto[ξ_, σ_, γ_:0.1]:=
Module[{i, j, n, d, ΔV, test, x1, x2, MyList, OurList, m, M, l, c, k},
{n, d} = Dimensions[ξ];
ΔV[x1_, x2_] = {D[H[{x1, x2}, ξ, σ], x1], D[H[{x1, x2}, ξ, σ], x2]};
OurList = Table[
test = True; i = 1; MyList = {ξ[[j]]};
```

```
While[test, i = i + 1;
{x1, x2} = MyList[[i - 1]];
AppendTo[MyList, {x1, x2} - γΔV[x1, x2]];
test =
√((MyList[[i]] - MyList[[i - 1]]).(MyList[[i]] - MyList[[i - 1]])) >
10^-6;
];
Print[j = , j, " # iterazioni = ", i, " ", MyList[[i]]];
MyList, {j, n}
];
M = Max[Map[Length, OurList]];
MyList = Table[
l = OurList[[k]];
m = Length[l];
c = l[[m]];
Join[l, Table[c, {M - m}]],
{k, Length[OurList]}
];
Table[MyList[[Range[Length[MyList]], i]], {i, M}]
];

Clusters[ξ_, LTT_, t_:0.43, σ_, etic_]:=
Module[{LL, CC, ind, clu, LT = LTT[[1]], vv = LTT[[2]], kk, j0, a},
LL = Ceiling [LT[[-1]]10^3] 10^ - 3;
CC = Union[LL];
posizioni = Table[Flatten[Position[LL, CC[[i]]]], {i, Length[CC]}];
{n, d} = Dimensions[ξ];
```

```
{Et, xmin, Ht} = En[ξ, σ];
Vt = 1 + Ht/Et;
Do[
kk = vv[[posizioni[[i]]]];
Do[
clu[kk[[j]]] = i, {j, 1, Length[kk]}
], {i, 1, Length[posizioni]}
];
Y = Select[Vt, # > t&];
ind = Reverse[Ordering[Vt]];
Do[
distmin = 1000;
Do[
dist = √((ξ[[ind[[i]]]] − CC[[j]]).(ξ[[ind[[i]]]] − CC[[j]]));
If[dist < distmin, distmin = dist; j0 = j],
{j, 1, Length[CC]}
];
clu[ind[[i]]] = j0,
{i, 1, Length[Y]}
];
Table[clu[i], {i, n}];
misclass = Select[Table[clu[i], {i, n}] − etic, # ≠ 0&];
Print[Le classificazioni errate sono , Length[misclass],
 e l'errore percentuale è , Length[misclass]/n100//N, %];
Table[clu[i], {i, n}]
]
```

```
ClusterPlot[LT_, ξ_, CC_, fig_]:=
Module[{Lista, LL, centri, figcentri, All},
LL = Ceiling [LT[[−1]]10^3] 10^ − 3;
centri = Union[LL];
Lista = Table[{PointSize[0.015], Colors[[j]],
Map[Point, Table[ξ[[CC[[j, i]]]], {i, 1, Length[CC[[j]]]}]]},
{j, 1, Length[CC]}];
figcentri = {PointSize[0.03],
Map[Point, Table[centri[[i]], {i, 1, Length[centri]}]]};
All = Graphics[Flatten[{Lista, figcentri}, 1], AspectRatio → 1,
Frame → True, FrameLabel → {PC1, PC2}, AspectRatio → 1,
DisplayFunction → Identity, FrameStyle → Thickness[0.01],
TextStyle → {FontFamily->Times New Roman, FontWeight → Bold,
FontSize → 12}];
Show[{Graphics[Lista], Graphics[figcentri], fig}, AspectRatio → 1,
Frame → True];
Export[dir <> Cluster.eps, All];
];
```

Bibliografia

[1] Anderson E., *The irises of the Gaspe Peninsula*, Bullettin of the American Society 59: 2-5 (1935)

[2] Barzilai H., Kheyfits A. e Andrews K., *A Gentle Introduction to Mathematical Cluster Analysis*, DIMACS Educational Module Series (2003)

[3] Bishop C. M., *Neural Network for Pattern Recognition*, Oxford University Press (1995) pp. 310-319

[4] Blatt M., Wiseman S. e Domany E., *Data clustering using a model granular magnet*, Phys. Rev. Lett. 76 (1996) 3251

[5] Cufaro Petroni N., *Statistica con elementi di probabilità*, Università degli Studi di Bari, a.a. 2004-05 pp. 23-36

[6] D'Agostini G., *Teaching statistics in the physics curriculum: Unifying and clarifying role of subjective probability* (1999)

[7] De Palma F., *Algoritmi di clustering per ricerche di sorgenti puntiformi per telescopi di neutrini*, Tesi di Laurea in Fisica Astroparticellare, Università degli Studi di Bari, A.A. 2005/06.

[8] Duda R. O., Hart P. E. e Stork. D.G., *Pattern Classification*, Wiley-Interscience, 2nd ed. (2001) chapter 4

[9] Fisher R. A., *The use of multiple measurements in taxonomic problems*, Annals of Eugenics 7: 179-188 (1936)

[10] Geymonat L., *Storia della filosofia - Volume I*, Garzanti (1973) pp. 212-227

[11] Nakai K. e Horton P., *A Probablistic Classification System for Predicting the Cellular Localization Sites of Proteins*, Intelligent Systems in Molecular Biology, 109-115 St. Louis, USA 1996.

[12] Nakai K. e Horton P., *A probabilistic inference system for the prediction of subcellular localization sites of protein: application to E. coli data set*, Proceedings of the *Intelligent systems in molecular biology*, pp. 368-383, Hawaii, USA 1996.

[13] Horn D. e Gottlieb A., *The method of quantum clustering*, Proceedings of the Neural Information Processing Systems: NIPS'01 (2001) 769-776

[14] Horn D. e Gottlieb A., *Algorithm for Data Clustering in Pattern Recognition Problems Based on Quantum Mechanics*, Phys. Rev. Lett. 88 (2002) 018702

[15] Jain A. K. e Dubes R. C., *Alghoritms for Clustering Data*, Prentice-Hall (1988) pp. 1-132, 266-272

[16] Jolliffe I. T., *Principal Component Analysis*, Springer Verlag, 2nd ed. (2002) pp. 1-38

[17] Kaufman L. e Rousseeuw P. J., *Finding Groups in Data, An Introduction to Cluster Analysis*, J. Wiley & Sons (1990) pp. 1-50

[18] Mill J. S., *A system of logic, ratiocinative and inductive*, Harper & Brothers, 6th ed. (1882)

[19] Mirkin B., *Mathematical Classification and Custering*, J. Wiley & Sons (1996) pp.1-5

[20] Nardulli G.,*Meccanica Quantistica I, Principi*, FrancoAngeli (2001) pp. 155-195

[21] Roberts S. J., *Parametric and nonparametric unsupervised cluster analysis*, Pattern Recognition, 30 261 (1997)