Università degli Studi di Padova
Dipartimento di Scienze Statistiche
Corso di Laurea in

Statistica, Economia e Finanza

RELAZIONE FINALE

Studio dell'effetto di criteri alternativi per la selezione del parametro di lisciamento nei metodi di raggruppamento non parametrici

Relatore: dott.ssa Giovanna Menardi

Dipartimento di Scienze Statistiche

Laureanda: Martina Dossi

Matricola n. 1051342

Anno Accademico 2014/2015

Indice

Introduzione

Nella sua accezione più generale, l'analisi di raggruppamento (*cluster analysis*) ha lo scopo di far emergere da un insieme di dati gruppi di osservazioni che siano tra loro simili e il più possibile distinte dalle osservazioni appartenenti agli altri gruppi. Cormack (1971) definisce queste caratteristiche in termini di coesione intra-gruppo e isolamento tra i gruppi.

Un problema di *clustering* non ha natura strettamente statistica e per questo varie discipline hanno offerto il loro apporto sviluppando tecniche e approcci differenti. In questa tesi si circoscrive l'argomento ad un contesto statistico, dal momento che si considera l'insieme dei dati a disposizione come un campione casuale generato da una certa distribuzione di probabilità. Nell'ambito di queste tecniche, cui si fa riferimento come *metodi basati sulla densità*, si distinguono due categorie principali.

Nella prima confluiscono i metodi parametrici, basati sull'assunzione che la distribuzione generatrice dei dati sia una mistura di funzioni di densità, dove ciascuna componente è scelta in una famiglia parametrica nota (a meno di parametri) ed è associata ad un *cluster*. Per una trattazione completa, si rimanda, per esempio, a McLachlan e Peel (2004).

La seconda categoria, invece, che rappresenta l'oggetto di studio di questa tesi, è composta dai metodi non parametrici. L'idea, che si attribuisce a Carmichael *et al.* (1968) e successivamente a Wishart (1969), è che i gruppi sorgano naturalmente in corrispondenza di regioni dello spazio campionario caratterizzate da elevata densità e siano separati tra loro da avvallamenti in cui la densità è inferiore. Pertanto, i *cluster* identificati coincidono con i domini di attrazione delle mode della funzione di densità, diventandone una proprietà intrinseca; da qui il riferimento a questa classe di metodi come *clustering modale*. Il principale vantaggio deriva dalla possibilità di individuare gruppi aventi forma arbitraria associati alle regioni più densamente popolate. Inoltre, non è richiesta alcuna conoscenza a priori o ipotesi circa il numero di *cluster* esistenti, che emerge in modo spontaneo dal processo di stima della densità.

Per quanto diversi, tutti gli approcci inclusi nel *clustering modale* prevedono una stima della funzione di densità con un metodo non parametrico. Delineare la

struttura e l'andamento dei dati con uno stimatore che non necessita l'imposizione di alcun modello matematico permette di associare i *cluster* alle caratteristiche geometriche della funzione stimata. Quale che sia lo stimatore scelto, inoltre, la sua varianza risulterà governata da un parametro di lisciamento che ne renderà più o meno frastagliato l'aspetto e la cui determinazione sarà cruciale per la corretta identificazione delle mode e, quindi, dei gruppi. In questa tesi si approfondirà il metodo del nucleo, probabilmente il più diffuso.

Nella *cluster analysis* il punto di arrivo è definire se esistono, ed eventualmente quanti sono, i possibili gruppi all'interno del campione di dati in esame. Per questo motivo la stima della densità si inserisce in una fase intermedia ed è plausibile che un *range* di parametri di lisciamento, e non uno soltanto, sia ottimale per raggiungere lo stesso obiettivo. Capire che entità abbia la scelta del grado di lisciamento nella stima della funzione di densità rispetto ai risultati dell'analisi di *clustering* rappresenta il *focus* della tesi, sviluppato nell'ordine che segue.

Il primo capitolo tratta la stima non parametrica della funzione di densità, definendo, nello specifico, il metodo del nucleo dal suo sviluppo storico alle proprietà che lo caratterizzano. A seguire, prende spazio una rassegna dei principali criteri per la selezione del parametro di lisciamento.

Il secondo capitolo approfondisce due diversi approcci entro cui si possono distinguere i metodi di raggruppamento non parametrici, analizzando, per ciascuno, alcune procedure.

Il terzo capitolo riporta un'analisi empirica per confrontare i due approcci al *clustering* modale e valutare l'impatto di criteri alternativi di selezione del parametro di lisciamento nella stima della densità. L'analisi è svolta sia mediante uno studio di simulazione, sia attraverso un'applicazione a dati climatici.

Si conclude, infine, con valutazioni di ordine generale che raccolgono sinteticamente i risultati emersi, lasciando nota per ulteriori tracce di approfondimento.

Capitolo 1

Stima non parametrica della densità

Specificare la funzione di densità $f(x) : \mathbb{R} \to \mathbb{R}^+ \cup \{0\}$ di una variabile casuale continua $X \in \mathbb{R}$ rappresenta un concetto fondamentale in statistica, in quanto equivale ad esprimerne la conoscenza completa. Infatti, si ricava qualsiasi probabilità associata ad X secondo la relazione:

$$P(X \in A) = \int_A f(x)dx \qquad \forall A \subset \mathbb{R}. \tag{1.1}$$

Spesso, tuttavia, la distribuzione di probabilità di una variabile casuale non è nota ed è necessario stimarla sulla base delle realizzazioni campionarie a disposizione. Si assume, quindi, di osservare un campione di n copie, tipicamente indipendenti, della variabile casuale X, la cui funzione di densità $f(\cdot)$ non è nota. Per stimarla si possono intraprendere due strade:

- approccio parametrico: si presuppone che i dati siano generati da una famiglia parametrica di distribuzioni nota a meno di uno o più parametri; di conseguenza, stimare $f(\cdot)$ equivale a stimare i parametri per quella specifica distribuzione;
- approccio non parametrico: consiste nel ricavare dalle osservazioni ogni possibile informazione sulla variabile di interesse, evitando di porre restrizioni in merito alla distribuzione sottostante.

In seguito verrà approfondito il secondo approccio, rivolgendo in particolare l'attenzione al *metodo del nucleo*.

1.1 Il metodo del nucleo

1.1.1 Sviluppo storico

Il metodo del nucleo consente di definire uno stimatore non parametrico per la funzione di densità, tale da delineare una struttura nei dati senza le restrizioni derivanti dalla scelta di un particolare modello parametrico. Proposto nel 1956 da Rosenblatt e Al. e successivamente affinato da Parzen (1962), esso rappresenta una generalizzazione del concetto di istogramma, che è uno strumento grafico comunemente utilizzato per l'analisi esplorativa dei dati quando lo spazio è unidimensionale. Per realizzarlo, la retta dei numeri reali viene suddivisa in un numero arbitrario di intervalli aventi ampiezza pari ad h, spesso (ma non necessariamente) assunta costante, e su ciascuno è costruito un rettangolo la cui altezza è direttamente proporzionale al numero di unità che vi appartengono. Formalmente, indicati con $S = \{x_1, x_2, \ldots, x_n\}$ l'insieme dei dati campionari e con $\tilde{x}_i$ il punto centrale dell'intervallo in cui cade x_i, l'istogramma è definito nel modo seguente:

$$\tilde{f}(x) = \frac{1}{h} \sum_{i=1}^{n} I(x - \tilde{x}_i; h), \tag{1.2}$$

dove $I(x - \tilde{x}_i; h)$ è la funzione indicatrice dell'intervallo $[-h/2, h/2]$ e $1/h$ è una costante di normalizzazione posta per garantire che $\int \tilde{f}(x)dx = 1$. Tuttavia, limitarsi ad usare l'istogramma per stimare la densità presenta varie criticità:

1. l'inevitabile perdita di informazione, derivante dal fatto di usare il punto centrale di ogni intervallo come rappresentante dell'intervallo stesso in cui cade in punto di cui si vuole calcolare la densità;

2. $\tilde{f}(x)$ non è una funzione liscia come sarebbe auspicabile, ma è discontinua con un andamento a gradini: ha salti in corrispondenza di ogni $(\tilde{x}_i \pm h/2)$ e derivata nulla in ogni altro punto;

3. il comportamento di $\tilde{f}(x)$ è vincolato all'ampiezza fissata per gli intervalli.

Il primo problema è stato risolto con l'introduzione dello stimatore *naive*, chiamato anche istogramma mobile perchè considera intervalli centrati su ciascun punto di cui si stima la densità piuttosto che su punti fissi.

Data la variabile aleatoria X, si può esprimere la sua funzione di densità $f(x)$ in un punto x come il limite del rapporto tra la probabilità che la variabile casuale assuma valori in un certo intervallo $(x - h, x + h)$ e l'ampiezza di tale intervallo, al

tendere dell'ampiezza a 0:

$$f(x) = \lim_{h \to 0} \frac{P(x - h < X < x + h)}{2h}. \tag{1.3}$$

Quindi, fissato h arbitrariamente piccolo, si ricava una stima della quantità al numeratore attraverso la proporzione di unità campionarie appartenenti all'intervallo $(x - h, x + h)$. Lo stimatore *naive* è definito come:

$$\hat{f}(x) = \frac{1}{2h} \sum_{i=1}^{n} \frac{1}{n} w\left(\frac{x - X_i}{h}\right), \tag{1.4}$$

dove $w(z)$ è una funzione che assegna peso unitario alle osservazioni con distanza inferiore ad h rispetto al punto di interesse e nullo in caso contrario:

$$w(z) = \begin{cases} 1 & se \ \ |z| < 1 \\ 0 & altrimenti. \end{cases} \tag{1.5}$$

Di conseguenza, la stima della densità in un punto x si costruisce posizionando un rettangolo (la stessa forma utilizzata nell'istogramma usuale) con base $2h$ e altezza $(2nh)^{-1}$ su ogni osservazione e sommando il numero di rettangoli in corrispondenza di x. É evidente che la stima ottenuta sarà ancora una funzione a gradini e che, di conseguenza, la quantità $\hat{f}(x)$ rimanga non derivabile (oltre che strettamente legata ad h).

Come anticipato, è a partire dal 1956 che si sviluppa un nuovo approccio, il metodo del nucleo, in grado di rendere lo stimatore derivabile. Intuitivamente, l'idea è quella di collocare su ogni osservazione, in luogo del rettangolo, una funzione liscia detta nucleo (o *kernel*).

1.1.2 Definizione dello stimatore

Alla luce delle considerazioni avanzate, è possibile ora definire formalmente il metodo del nucleo per la stima della densità.

Sia $S = \{x_1, \ldots, x_n\}$ un campione di n osservazioni tratte da una variabile casuale X, con funzione di densità $f(x) : \mathbb{R} \to \mathbb{R}$ non nota, lo stimatore *kernel* di $f(x)$ risulta così definito:

$$\hat{f}(x) = \frac{1}{n} \sum_{i=1}^{n} K_h(x - X_i), \tag{1.6}$$

dove h è detto parametro di lisciamento o ampiezza della finestra, mentre $K_h(u) = [K(u/h)]/h$ è il nucleo, tipicamente scelto nella famiglia delle funzioni simmetriche, continue e derivabili in modo da trasferire queste proprietà a $\hat{f}(x)$ e tale che $\int K(x)dx = 1$. Inoltre, si farà riferimento ad una funzione *kernel* per cui valgono le seguenti condizioni:

$$K(x) \geqslant 0, \quad \int xK(x)dx = 0 \;\; e \;\; \int x^2K(x)dx = h^2 \neq 0 \tag{1.7}$$

La scelta di $K(x)$ determina la forma della funzione, mentre quella di h ne stabilisce l'ampiezza, governando la varianza dello stimatore.

Per quanto riguarda la forma della funzione *kernel*, è stato verificato che essa non modifica in modo sostanziale il comportamento dello stimatore e che, anzi, la preferenza di una funzione piuttosto che di un'altra è spesso dettata da motivi computazionali o di semplice convenienza. Nella Figura 1.1 se ne illustrano tre particolarmente significative (per una trattazione esaustiva si rimanda a Silverman (1986), Capitolo 3). Le tre forme, in ordine da sinistra, sono:

- *Kernel* uniforme: $K(x) = 1/2$ per $|x| < 1$, 0 altrimenti; è la forma utilizzata nei metodi precedenti (istogramma e istogramma mobile). Con questa scelta lo stimatore *kernel* si riduce allo stimatore *naive*. Come si è già notato, esso produce una stima della densità discontinua.
- *Kernel* gaussiano: $K(x) = \frac{1}{\sqrt{2\pi}} exp\left\{-\frac{x^2}{2}\right\}$; questa scelta è probabilmente la più diffusa e sarà quella adottata anche in seguito.

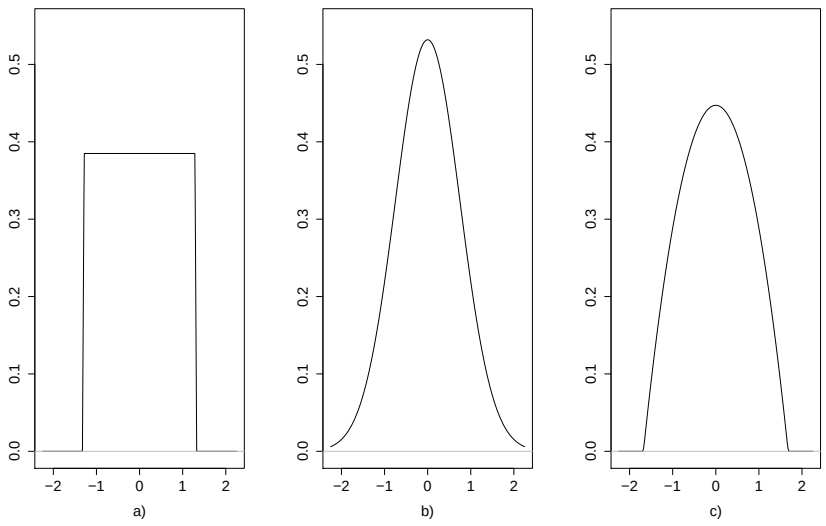


Figura 1.1: Diverse forme di *kernel*: a) uniforme, b) gaussiano, c) di Epanechnikov.

- *Kernel* di Epanechnikov: $K(x) = \frac{3}{4}\left(1 - \frac{1}{5}x^2\right)/\sqrt{5}$, per $|x| < \sqrt{5}$, 0 altrimenti; si è dimostrato essere il *kernel* con maggiore efficienza.

Al contrario, il valore dell'ampiezza della finestra h incide fortemente sul comportamento dello stimatore $\hat{f}(x)$. Per suggerire un'idea di quanto la stima della densità sia variabile in funzione di tale parametro si confrontano due diversi casi in Figura 1.2. Supponendo di conoscere la vera struttura della densità di una certa variabile X, se ne offre una rappresentazione nel primo grafico, dal quale si osservano chiaramente due mode. Nel secondo e nel terzo grafico la densità viene stimata: prima scegliendo un valore di h molto piccolo, che conduce ad una stima frastagliata, e poi con un h più elevato, tanto da oscurare la vera natura della densità e renderla unimodale.

Va considerato inoltre che questo esempio non rappresenta un caso isolato, ma piuttosto una caratteristica generale dello stimatore *kernel*. La sua sensibilità rispetto al parametro di lisciamento ne costituisce un limite intrinseco, più o meno marcato in relazione all'insieme dei dati oggetto di studio e agli obiettivi dell'analisi. In risposta, quindi, sono state avanzate in letteratura varie procedure alternative per la sua selezione, alcune delle quali saranno illustrate nei paragrafi successivi.

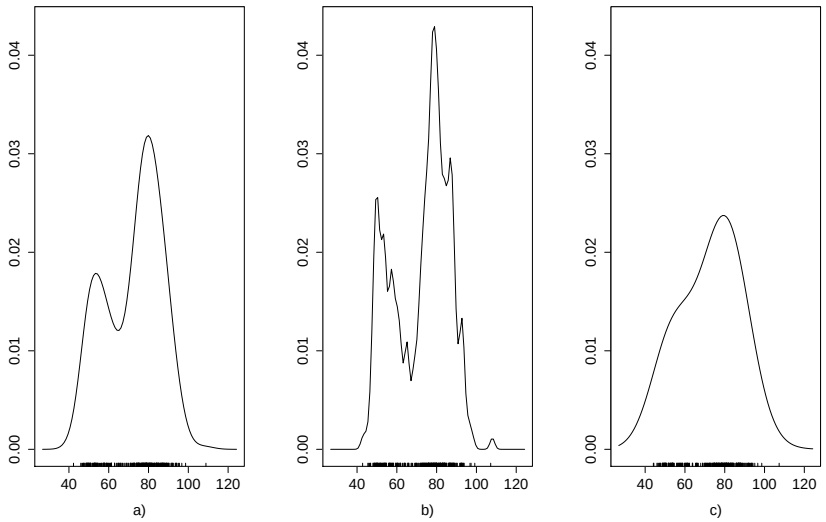


Figura 1.2: Funzione di densità: a) reale, b) sotto-lisciata, c) sovra-lisciata.

1.1.3 Proprietà

Si consideri lo stimatore definito in (1.6), basato su una funzione nucleo simmetrica e che soddisfi le proprietà in (1.7). Il valore atteso di $\hat{f}(x)$ è definito come:

$$E\{\hat{f}(x)\} = \int \frac{1}{h} K\left(\frac{x-y}{h}\right) f(y)dy, \tag{1.8}$$

ovvero come prodotto di convoluzione[1] tra la densità f e la funzione nucleo K. Posto $z = (x-y)/h$ e ricordando che l'integrale della funzione *kernel* ha valore unitario, l'espressione (1.8) risulta equivalente a:

$$E\{\hat{f}(x)\} = \int K(z)f(x-zh)dz. \tag{1.9}$$

Indicate con f' e f'' rispettivamente la derivata prima e la derivata seconda di f, attraverso uno sviluppo in serie di Taylor troncato al termine di secondo grado:

$$f(x-hz) \simeq f(x) + (-hz)f'(x) + \frac{1}{2}(-hz)^2 f''(x), \tag{1.10}$$

si ottiene l'espressione asintotica:

$$E\{\hat{f}(x)\} \simeq f(x) + \frac{h^4}{2} f''(x). \tag{1.11}$$

Per quanto riguarda la varianza, si dimostra (Silverman, 1986) che:

$$Var\{\hat{f}(x)\} = \sigma_k^2 = \frac{1}{n}\int \frac{1}{h^2} K\left(\frac{x-y}{h}\right)^2 f(y)dy - \left\{\frac{1}{h}\int K\left(\frac{x-y}{h}\right) f(y)dy\right\}^2; \tag{1.12}$$

mentre, asintoticamente, risulta:

$$Var\{\hat{f}(x)\} \simeq \frac{1}{nh}\int K(z)^2 dz. \tag{1.13}$$

La distorsione dello stimatore (*bias*), vale a dire lo scarto tra il suo valore atteso e la quantità non nota di interesse, non dipende direttamente dall'ampiezza campionaria n, ma soltanto dal valore assunto da h. Concettualmente, questo significa che per rendere la stima più accurata è inutile incrementare il numero di unità osservate, ma è necessario considerare metodi diversi per determinare il valore di h. Da quanto

[1] Si definisce prodotto di convoluzione tra due funzioni $f : \mathbb{R} \to \mathbb{R}$ e $g : \mathbb{R} \to \mathbb{R}$ l'operazione $(f * g)(x) = \int f(y)g(x-y)dy = \int f(x-y)g(y)dy$.

ricavato in (1.8) e in (1.11), è immediato notare che:

$$b(x) = \int \frac{1}{h} K\left(\frac{x-y}{h}\right) f(y)dy - f(x) \simeq \frac{h^2}{2}\sigma_k^2 f''(x) \neq 0, \tag{1.14}$$

e quindi lo stimatore risulta distorto per campioni finiti.

Misure di discrepanza

Per valutare l'accuratezza dello stimatore $\hat{f}$ rispetto alla vera densità f si utilizzano generalmente le seguenti misure di discrepanza:

- l'errore quadratico medio *(MSE)*, per stabilire la bontà dello stimatore in un singolo punto:

$$MSE_x(\hat{f}) = E\{\hat{f}(x) - f(x)\}^2 = \{E\{\hat{f}(x)\} - f(x)\}^2 + Var\{\hat{f}(x)\}, \tag{1.15}$$

  pari alla somma tra distorsione al quadrato e varianza dello stimatore in tale punto.

- L'errore quadratico integrato *(ISE)*, come misura globale della distanza tra $\hat{f}$ e f:

$$ISE(\hat{f}) = \int \{\hat{f}(x) - f(x)\}^2 dx, \tag{1.16}$$

  che risulta appropriato soltanto quando l'interesse è rivolto all'unico campione disponibile, e non, in termini di valore atteso, a tutti quelli che sarebbe possibile trarre da f.

- L'errore quadratico medio integrato *(MISE)*, ovvero il valore atteso di (1.16):

$$MISE(\hat{f}) = E\int \{\hat{f}(x) - f(x)\}^2 dx, \tag{1.17}$$

  che indica la bontà dello stimatore in tutto lo spazio campionario. Si ottiene:

$$\begin{aligned} MISE(\hat{f}) &= \int E\{\hat{f}(x) - f(x)\}^2 dx = \int MSE_x(\hat{f})dx = \\ &= \int \{E\{\hat{f}(x)\} - f(x)\}^2 dx + \int Var\{\hat{f}(x)\}dx, \end{aligned} \tag{1.18}$$

  per cui risulta che il *MISE* coincide con la somma dell'integrale dell'errore quadratico dello stimatore e dell'integrale della sua varianza.

Sostituendo ora, nella formula del *MISE* fornita in (1.18), le espressioni asintotiche ricavate per la varianza (1.13) e per la distorsione (1.14), si ricava l'espressione

asintotica del *MISE*:

$$AMISE(\hat{f}) = \frac{h^4}{4}\sigma_k^4 \int f''(x)^2 dx + \frac{1}{nh} \int K(z)^2 dz, \tag{1.19}$$

più semplice da trattare rispetto a quella del *MISE*. Si osserva che, attraverso il primo addendo, l'*AMISE* risulta direttamente proporzionale a h^4, mentre, considerando il secondo addendo, si ha un rapporto di proporzionalità inversa rispetto a (nh). Questo significa che un valore di h relativamente basso indurrà una diminuzione nell'errore sistematico dello stimatore sulle spese della varianza, e viceversa. Il *trade-off* tra varianza e *bias* suggerisce la criticità del ruolo ricoperto dal parametro di lisciamento h, in accordo con l'idea intuitiva presentata nel Paragrafo 1.1.2 in riferimento alla Figura 1.2.

1.2 Il parametro di lisciamento

La scelta del parametro di lisciamento è guidata dall'obiettivo per cui viene stimata la densità ed è vincolante per la performance dello stimatore. Se l'intento è esplorare i dati per coglierne l'andamento, anche una scelta soggettiva di h può rivelarsi efficace. Quando invece la stima della densità rappresenta l'obiettivo primario dell'analisi, individuare il valore ottimale di h diventa, come è naturale attendersi, il cuore del problema. Nell'analisi dei *cluster* non parametrica, oggetto di questa tesi, la stima della densità non costituisce il punto di arrivo dell'analisi, ma è, essenzialmente, una fase intermedia; tuttavia, poiché si assume che i *cluster* coincidano i domini di attrazione delle mode, è comunque molto importante che la funzione stimata sia quanto più possibile vicina alla vera densità, senza essere sovra-lisciata o sotto-lisciata, il che condurrebbe a risultati conclusivi fuorvianti.

Al problema della scelta di h non esiste un approccio universale, per questo in letteratura emergono diversi criteri di selezione. In generale, di fronte ad ogni specifico insieme di dati, si ricerca quel valore di h che consente di definire una stima della densità tale da riprodurre l'andamento sottostante dei dati oscurando i rumori causati da variazioni campionarie. Alcuni metodi mirano a fornire un valore di h che sia il più possibile ragionevole rispetto ad un'ampia gamma di situazioni, senza, tuttavia, garanzia matematica che sia anche quello ottimale. Come si vedrà, il primo metodo descritto rientra in questa classe. Una seconda categoria di procedure, invece, comprende quelle aventi una base matematica più solida e che richiedono uno sforzo computazionale maggiore, ma in grado di fornire un valore di h ottimale secondo un certo criterio. Come naturale proseguimento di quanto sviluppato finora,

il criterio perseguito sarà la minimizzazione del *MISE*, sfruttando la sua espressione asintotica *(AMISE)*, in (1.19). In questo modo, attraverso sostituzioni e calcoli, si ricava:

$$h_{ott} \simeq (\sigma_k^2)^{-2/5} \left(\frac{\int K(z)^2 dz}{n \int f''(x)^2 dx} \right)^{1/5}, \tag{1.20}$$

che, purtroppo, costituisce un'espressione puramente teorica, dal momento che presuppone la conoscenza di $f(x)$. Si possono comunque derivare alcune interessanti osservazioni:

- n e h sono inversamente proporzionali e il tasso con cui l'uno converge a 0 all'aumentare dell'altro è molto lento,
- $\int f''(x)^2 dx$ è una misura della rapidità delle fluttuazioni della densità, quindi si preferisce un valore piccolo di h quando le fluttuazioni sono più rapide.

Di seguito si presenta una rassegna dei principali approcci introdotti per la selezione del parametro di lisciamento. Per una trattazione completa si veda Wand e Jones (1994).

Riferimento alla distribuzione normale

Il problema emerso può essere affrontato utilizzando una famiglia standard di distribuzioni per assegnare un valore al termine $\int f''(x)^2 dx$ nell'espressione (1.20) per la scelta di h ottimale. Per convenienza, spesso si assume come riferimento la distribuzione normale, con media nulla e varianza σ^2, in modo da avere:

$$\int f''(x)^2 dx = \frac{3}{8\sqrt{\pi}} \sigma^{-5}. \tag{1.21}$$

Quindi, scegliendo un kernel di forma gaussiana, si ha:

$$h_N = (4\pi)^{-1/10} \frac{3}{8\sqrt{\pi}} \sigma n^{-1/5} = (\frac{4}{3n})^{1/5} \sigma \approx 1.06 \sigma n^{-1/5}. \tag{1.22}$$

Può sembrare contraddittoria l'idea di fare riferimento ad una distribuzione sottostante dei dati nell'ambito della stima non parametrica della densità. Infatti, se si sapesse che X ha distribuzione normale, allora sarebbe ragionevole stimarne la densità attraverso una stima dei parametri μ e σ^2, ottenuta sulla base delle realizzazioni campionarie di X, da inserire successivamente nell'usale formula della densità normale. La differenza, con questo procedimento, è che non si considerano vere e proprie ipotesi preliminari sulla distribuzione di X, ma si sta soltanto proponendo una tecnica per selezionare h che risulterà tanto più adeguata quanto più i dati si

avvicinano all'ipotesi di normalità. Pertanto, il metodo si adatta in modo soddisfacente a tutti quei casi in cui la vera densità è unimodale e liscia, mentre, se così non fosse, rischierà di produrne un sovra-lisciamento. D'altra parte, essendo la distribuzione normale tra le più lisce, il valore di h ricavato secondo questo approccio risulterà probabilmente più grande se confrontato con quello di altri metodi.

Per stimare σ una possibilità è usare la deviazione standard campionaria corretta, ma in letteratura si trovano anche stimatori più robusti, particolarmente indicati qualora siano presenti *outliers* o code lunghe e pesanti (si veda, ad esempio Silverman (1986), pp. 45-47, e Wasserman (2006)) .

Convalida incrociata dei minimi quadrati

Il criterio della convalida incrociata dei minimi quadrati è stato introdotto da Rudemo (1982) e Bowman (1984). Si consideri l'errore quadratico integrato definito in (1.16). Espandendo la formula si ricava:

$$ISE(\hat{f}) = \int \hat{f}(x)^2 dx - \int \hat{f}(x) f(x) dx + \int f(x)^2 dx. \tag{1.23}$$

Dal momento che l'ultimo addendo non dipende da h, si può, equivalentemente, minimizzare la quantità:

$$R(\hat{f}) = ISE(\hat{f}) - \int f(x)^2 dx = \int \hat{f}(x)^2 dx - \int \hat{f}(x) f(x) dx. \tag{1.24}$$

Il principio consiste nel costruire una stima per R sulla base dei dati a disposizione e individuare il valore di h per cui è minima. Sia $\hat{f}_{-i}(x)$ la stima (usualmente indicata con l'espressione *leave-one-out*) di $f(x)$ ottenuta escludendo dal campione l'i-esima osservazione, vale a dire:

$$\hat{f}_{-i}(x) = \frac{1}{h(n-1)} \sum_{i=1, i \neq j}^{n} K\left(\frac{x - X_j}{h}\right). \tag{1.25}$$

Uno stimatore non distorto di R è definito nel modo seguente:

$$M(h) = \int \hat{f}(x)^2 dx - \frac{2}{n} \sum_i \hat{f}_{-i}(X_i), \tag{1.26}$$

quantità che è possibile ricavare sulla base delle osservazioni campionarie. Infatti si dimostra (Silverman (1986), Paragrafo 3.4.3) che $R(\hat{f})$ e $M(h)$ hanno lo stesso valore atteso e, di conseguenza, minimizzare $M(h)$ equivale a minimizzare $R(\hat{f})$, quindi anche l'*ISE* e, di conseguenza, il *MISE*. In particolare, si ottiene una stima

migliore di h quando il valore che minimizza $M(h)$ è vicino a quello che minimizza il suo valore atteso $E\{M(h)\}$.

Vari studi hanno messo in luce un aspetto critico della *performance* di questo stimatore, sia da un punto di vista teorico che pratico. In particolare, è stata rilevata una forte variabilità, spingendo a cercare alternative più robuste.

Convalida incrociata della verosimiglianza

Questo metodo rappresenta lo sviluppo naturale dell'idea di utilizzare la verosimiglianza per valutare la bontà di adattamento di un modello statistico (Silverman, 1986). Si supponga di avere, oltre al campione di partenza $\{x_1, \ldots, x_n\}$, un'osservazione indipendente aggiuntiva y generata dalla stessa funzione di densità comune ai dati. Considerando $\hat{f}$ come una famiglia parametrica di densità che dipende dal parametro h, si determina la log-verosimiglianza $log\{\hat{f}(h; y)\}$ di h. Ma, dal momento che y in realtà non é disponibile, si omette dal campione un'osservazione x_i da usare in sua sostituzione. Si ottiene $\hat{f}_{-1}$ come definita in (1.25) e la log-verosimiglianza diventa $log\{\hat{f}_{-i}(h; x_i)\}$. Poichè niente motiva la scelta di un particolare x_i a favore di un altro, l'idea è quella di ricavare la log-verosimiglianza come una media rispetto a ciascun x_i escluso dal *dataset*; quindi:

$$CV(h) = \frac{1}{n} \sum_{i=1}^{n} log\hat{f}_{-i}(h; x_i) \tag{1.27}$$

costituisce la quantità da massimizzare per trovare il valore ideale di h. Secondo questo procedimento la stima ottenuta per la densità dovrebbe essere vicina a quella vera in termini di distanza di Kullback-Leibler (ma, come argomenta Silverman, solo sotto forti assunzioni per f):

$$d(f, \hat{f}) = \int f(x) log \left\{ \frac{f(x)}{\hat{f}(x)} \right\} dx, \tag{1.28}$$

il cui valore atteso coincide con quello di $-CV(h)$ a meno di una costante che non dipende da h per una stima calcolata con $(n-1)$ osservazioni.

Metodo *plug-in*

I metodi *plug-in* sono basati sull'idea di sostituire con una stima il termine incognito $f''(x)^2$ nella formula (1.20). Si tratta, in altri termini, di una generalizzazione del primo metodo proposto, in cui si è presa di riferimento la distribuzione normale, che prevede la sostituzione del termine incognito con una stima non parametrica di

tipo *kernel* (chiamata funzione *pilota*). Per poter approfondire le caratteristiche di questo metodo, si espongono ora brevemente alcuni concetti alla base.

Sia $f^{(r)}(x)$ la derivata *r-esima* di $f(x)$, per la quale si considera il seguente stimatore:

$$\hat{f}^{(r)}(x) = \frac{1}{nh^{(r+1)}} \sum_{i=1}^{n} K^{(r)} \left(\frac{x - X_i}{h} \right). \tag{1.29}$$

Si indichi con $R(f^{(s)})$ l'integrale della derivata della densità al quadrato, cioè:

$$R(f^{(s)}) = \int f^{(s)}(x)^2 dx \qquad \forall s. \tag{1.30}$$

Ora, posto $r = 2s$ e integrando per parti, la stessa quantità può essere espressa come:

$$\psi_r = \int f^{(r)}(x) f(x) dx \qquad \forall r. \tag{1.31}$$

Si osservi che ψ_r è anche il valore atteso della derivata r-esima di f, $\psi_r = E\{f^{(r)}(x)\}$. Questo permette di definire lo stimatore come:

$$\hat{\psi}_r(g) = \frac{1}{n} \sum_{i=1}^{n} \hat{f}^{(r)}(X_i; g), \tag{1.32}$$

dove g è un'ampiezza della finestra possibilmente diversa da h.

In linea con la notazione introdotta, si farà allora riferimento alla quantità $\int f''(x)^2 dx$ con ψ_4. Secondo la regola *plug-in* si sostituisce ψ_4 con lo stimatore *kernel* $\hat{\psi}_4(g)$. Il limite risiede nel fatto che non si tratta di una regola completamente automatica, data la dipendenza dalla scelta del parametro g. Si può pensare di definire g attraverso la formula per l'ampiezza della finestra ottimale ottenuta minimizzando il *MSE* asintotico di $\hat{\psi}_4(g)$ (Wand e Jones (1994), Paragrafo 3.5), ma questa quantità dipende da ψ_{r+2}. Immaginando di procedere iterativamente, anche per ψ_6 sarà possibile ricavare una stima *kernel*, la cui finestra ottimale però dipenderà da ψ_8. Il problema così non ha soluzione. Allora, in generale, una procedura *plug-in* consiste in l passi, cioè l successive stime *kernel*, con parametro di lisciamento iniziale ottenuto tramite un metodo semplice come il primo che è stato presentato. Una questione rilevante concerne chiaramente la scelta del valore l. Al crescere di l lo stimatore di h diventerà meno distorto perchè si allontana progressivamente dal primo parametro inserito per dare inizio alla procedura; ma subirà un incremento in termini di varianza. Si veda Wand e Jones (1994) per una spiegazione approfondita di questo metodo.

Convalida incrociata lisciata

Questo metodo è simile all'ultimo spiegato in quanto si affida ad uno stimatore *kernel* con ampiezza della finestra pilota g per stimare la componente ignota nella formula di h ottimale. La differenza risiede nel fatto che questo metodo si basa sull'espressione esatta del *MISE* e non sulla sua approssimazione asintotica. Risulta pertanto più complicato da analizzare e si rimanda a Wand e Jones (1994) per vedere nel dettaglio i passi che consentono di ricavarlo. È stato verificato (Wand e Jones, 1994) che questo metodo rappresenta una versione corretta della convalida incrociata dei minimi quadrati.

1.3 Il metodo del nucleo nel caso multivariato

In questo paragrafo si illustra una diretta estensione del metodo del nucleo da una a più dimensioni. Si presenta ora la notazione che verrà mantenuta in seguito. Sia X una variabile casuale continua, definita su $\mathbb{R}^d$ e avente funzione di densità $f : \mathbb{R}^d \to \mathbb{R}^+ \cup \{0\}$ non nota. Per la stima di f si osserva un campione S che assume la forma di una matrice di dati:

$$S_{n\times d} = \begin{pmatrix} x_{11} & \dots & x_{1j} & \dots & x_{1d} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ x_{i1} & \dots & x_{ij} & \dots & x_{id} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ x_{n1} & \dots & x_{nj} & \dots & x_{nd} \end{pmatrix},$$

dove ogni riga rappresenta un'unità statistica e ogni colonna una variabile. Una generica osservazione i-esima è rappresentata quindi da un vettore d-dimensionale $x_i = (x_{i1}, \dots, x_{id})$. Inoltre, $\int$ sarà un'abbreviazione di $\int \dots \int_{\mathbb{R}^d}$ e dx di $dx_1 \dots dx_d$.

Nella sua forma più generale lo stimatore *kernel* è definito come:

$$\hat{f}(x) = \frac{1}{n} \sum_{i=1}^{n} K_H(x - X_i), \tag{1.33}$$

dove H è una matrice $d \times d$ simmetrica e definita positiva chiamata matrice di lisciamento e, in analogia al caso univariato:

$$K_H(x) = |H|^{-1/2} K(H^{-1/2}x),$$

e K è una funzione *kernel* d-variata tale che $\int K(x)dx = 1$. Tipicamente si considera

per $K(x)$ una funzione di densità simmetrica, radiale e unimodale. Una scelta comune consiste nella densità della normale standard d-variata:

$$K(x) = \left(\frac{1}{\sqrt{2\pi}}\right)^d exp\left\{-\frac{x^\top x}{2}\right\}, \tag{1.34}$$

ma la generalizzazione a più dimensioni è ricavabile per ogni altra forma.

In generale, nella matrice H si contano $d(d+1)/2$ entrate indipendenti, che corrispondono ai parametri da stimare. Quindi, se da un lato una matrice completa consente più flessibilità, dall'altro introduce un grado maggiore di complessità, richiedendo un numero sempre crescente di parametri da stimare. Una semplificazione di (1.33) può essere ottenuta restringendo la scelta di H alla classe D delle matrici $d \times d$ diagonali e definite positive, cioè $H = Diag\{h_1^2, \ldots, h_d^2\}$ e un'ulteriore restrizione porta a considerare $H \in S$, dove $S = \{h^2 I : h > 0\}$, con I matrice identica di ordine d. In questo caso, che è il più semplice, il grado di lisciamento è lo stesso per ogni coordinata; pertanto, tipicamente, per limitare la perdita di informazione si opera una preventiva standardizzazione della matrice dei dati S.

Nell'ambito multivariato il problema relativo al lisciamento è duplice: si tratta prima di fissare una parametrizzazione per H e poi di scegliere un criterio per stimarla, per il quale si possono valutare i metodi discussi nel Paragrafo 1.2 (estesi a questo contesto) e altri diffusi in letteratura.

Considerando il riferimento alla distribuzione normale, la matrice di lisciamento può assumere la forma $h^2 S$, con S matrice corretta di varianze e covarianze campionarie di S e h^2 tale che:

$$h^2 = \left(\frac{4}{n(d+2r+2)}\right)^{\frac{2}{d+2r+4}}, \tag{1.35}$$

dove n è la numerosità campionaria, d la dimensione dello spazio considerato e r l'ordine di derivazione. Per una trattazione esaustiva del modo in cui è stata ottenuta questa formula si veda Chacón e Wand (2009).

Relativamente al metodo della convalida incrociata dei minimi quadrati, si ha:

$$\hat{H}_{LSCV} = \min_{H \in F}\left\{\int \hat{f}(x;H)^2 dx - \frac{2}{n}\sum_{i=1}^{n} \hat{f}_{-i}(X_i)\right\}, \tag{1.36}$$

dove $\hat{f}_{-i}(\cdot;H)$ è lo stimatore *kernel* basato sulle unità campionarie con esclusione di X_i e F rappresenta la classe delle matrici $d \times d$ simmetriche e definite positive. A seconda delle esigenze, F può essere sostituita da D o S, le sottoclassi entro cui

scegliere le altre parametrizzazioni per H. Questo stimatore conserva i limiti cui si è accennato per una dimensione, ma il tasso relativo di convergenza migliora per dimensioni elevate.

La versione multivariata della classe dei metodi *plug-in* si sviluppa come naturale estensione del caso unidimensionale, quindi la quantità ignota sarà rimpiazzata da uno stimatore *kernel* del tipo:

$$\hat{\psi}_r(G) = \frac{1}{n}\sum_{i=1}^{n} \hat{f}^{(r)}(X_i; G), \tag{1.37}$$

dove G è un'altra matrice di lisciamento.

Anche per i restanti metodi, la convalida incrociata della verosimiglianza e la convalida incrociata lisciata, la generalizzazione a più dimensioni non modifica le considerazioni valide in una dimensione.

1.4 Estensioni

In questo paragrafo si richiamano alcuni metodi non parametrici di stima della densità, alternativi a quello del nucleo. Il metodo del *vicino più prossimo* e quello del *nucleo variabile* muovono dalla considerazione che h non vada fissato sull'intero campione, ma debba essere lasciato libero di variare, così da adattare l'entità del lisciamento alla densità locale dei dati. Per esempio, un caso frequente in cui spesso risulta preferibile ricorrere a questi metodi si presenta con insiemi di dati provenienti da distribuzioni aventi code lunghe. Stimare la densità con un parametro di lisciamento piccolo in modo da coglierne i dettagli centrali può provocare l'insorgenza di irregolarità lungo le code; d'altra parte, aumentarne il valore si accompagna al rischio di sovra-lisciamento. L'idea sarà pertanto quella di adottare un parametro h maggiore dove i dati sono più radi e inferiore dove sono più concentrati.

Nel metodo del *k-esimo* vicino più prossimo l'ampiezza della finestra per costruire la stima in un punto x è definita in modo da essere direttamente proporzionale alla distanza di x dalla *k-esima* osservazione più vicina. Un parametro k governa il grado di lisciamento, scelto in modo da essere considerevolmente più piccolo della numerosità campionaria, tipicamente $k \simeq \sqrt{n}$. Sia $d(y,t) = |y-t|$ la distanza euclidea tra due punti lungo la retta dei reali, allora: $d_1(x) \leq d_2(x) \leq \cdots \leq d_n(x)\ \forall x$ rappresentano le distanze di x da ogni unità osservata, ordinate in senso crescente. Lo stimatore del *k-esimo* vicino più prossimo è definito come segue:

$$\hat{f}(x) = \frac{k-1}{2nd_k(x)}. \tag{1.38}$$

Mentre lo stimatore *naive* è basato sul numero di osservazioni che cadono in un intervallo di ampiezza fissata centrato su ogni punto di interesse, quello del vicino più prossimo è inversamente proporzionale all'ampiezza del rettangolo necessario a contenere un dato numero di osservazioni. Così, nelle code della distribuzione, la distanza $d_k(x)$ sarà maggiore che nella parte centrale della distribuzione, riducendo il problema del sotto-lisciamento. Come lo stimatore *naive*, anche questo non fornisce una funzione liscia ed esiste una versione generalizzata che lo avvicina allo stimatore *kernel*. Definita con $K(x)$ la funzione *kernel* che integra ad uno, la forma generalizzata dello stimatore del *k-esimo* vicino più prossimo é:

$$\hat{f}(x) = \frac{1}{nd_k(x)} \sum_{i=1}^{n} K\left(\frac{x - X_i}{d_k(x)}\right). \tag{1.39}$$

Si tratta esattamente dello stimatore *kernel* valutato in un punto x con parametro di lisciamento $h = d_k(x)$. Il lisciamento complessivo risulta governato da k, ma la finestra usata dipende dalla densità di osservazioni nelle vicinanze di ogni punto.

Il metodo del nucleo variabile, invece, è costruito a partire dallo stimatore *kernel*, con la differenza che, come suggerisce il nome, il parametro di lisciamento per ogni nucleo posto sulle osservazioni non é costante ma varia dall'una all'altra. Siano K una funzione *kernel*, k un intero positivo e $d_{j,k}$ la distanza di X_j dalla *k-esima* osservazione più vicina tra le altre $(n-1)$. Il metodo del nucleo variabile con parametro di lisciamento h è definito come:

$$\hat{f}(x) = \frac{1}{n} \sum_{j=1}^{n} \frac{1}{hd_{j,k}} K\left(\frac{x - X_j}{hd_{j,k}}\right). \tag{1.40}$$

L'ampiezza della finestra di ogni nucleo piazzato sulle osservazioni è proporzionale a $d_{j,k}$, così da associare *kernel* più appiattiti alle unità campionarie situate in regioni scarsamente popolate e il contrario per quelle appartenenti a zone più dense. Per ogni valore fissato di k, il lisciamento globale dipenderà, in questo caso, da h. A differenza dello stimatore del vicino più prossimo, dove il parametro di lisciamento usato per costruire la stima della densità in un punto x dipende dalla sua distanza dall'osservazione più vicina, con lo stimatore del nucleo variabile la stima in un punto è indipendente da quel punto, basandosi unicamente sulle distante tra le unità campionarie.

Sia in relazione a questi metodi che ad altri, come quello delle serie ortogonali, che offrono approcci ancora differenti, si rimanda a Silverman (1986) per una trattazione completa.

Per quanto concerne il contesto multidimensionale, le procedure descritte posso-

no esservi estese allo stesso modo di quanto si è visto per il metodo *kernel*. Nel caso del metodo del vicino più prossimo, la generalizzazione a d dimensioni avviene considerando un'ipersfera d-dimensionale centrata nel punto x di interesse e con raggio pari alla distanza euclidea tra x e la k-*esima* osservazione più vicina. Valgono, in generale, le stesse considerazioni esposte per il caso univariato. Anche il metodo del nucleo variabile si applica a più dimensioni preservando la stessa idea di fondo. Tuttavia, una complicazione aggiuntiva sorge nel momento in cui serve stabile se un'osservazione cade in una regione densamente popolata oppure il contrario. Per questa ragione, la procedura si divide in due passi e nel primo si cerca, tramite un metodo alternativo che può essere semplicemente il più conveniente da applicare, una stima grezza della densità, chiamata stima *pilota*, positiva in ogni osservazione. Questo permette di avere un'idea di come siano collocate le unità nello spazio, e di conseguenza consente di ricavare i parametri di lisciamento da usare per implementare il metodo. Per approfondire questi metodi e altri si veda Silverman (1986) (Capitolo 5).

Capitolo 2

Metodi di raggruppamento non parametrici

L'idea su cui si fonda l'analisi di *clustering* non parametrico risale a Carmichael *et al.* (1968), con la definizione di *gruppi distinti* come regioni contigue e densamente popolate dello spazio campionario circondate da regioni pressoché vuote. Wishart (1969) riprende questo concetto sviluppando una procedura di *clustering* coerente con la nozione di *cluster* associati alle regioni intorno alle mode della funzione di densità sottostante i dati, indipendentemente dalla loro forma e varianza. Emerge, chiaro, il vantaggio legato all'analisi di raggruppamento modale: il concetto di *cluster* è definito in modo preciso come dominio di attrazione delle mode della distribuzione da cui provengono i dati. Il numero di regioni così caratterizzate è allora ben definito e parte integrante del processo di stima della densità. Inoltre, l'utilizzo di metodi non parametrici per la stima della densità consente l'identificazione di gruppi di forma arbitraria.

Nell'ambito dei metodi di *clustering* non parametrico si stagliano due approcci. Uno si basa sulla ricerca delle componenti connesse associate a regioni ad alta densità dello spazio campionario, attraverso l'individuazione delle curve di livello della funzione di densità. L'altro, invece, è finalizzato ad una ricerca diretta delle mode della densità, e definisce quindi i *cluster* a partire dai punti di massimo locale della funzione. Nei prossimi paragrafi questi due orientamenti saranno illustrati approfonditamente.

A livello di notazione, nel seguito si farà riferimento ad un campione casuale $S = \{x_1, \ldots, x_n\} \in \mathbb{R}^d$ tratto da una funzione di densità $f(x)$ non nota, rispetto alla quale si indica con $\hat{f}(x)$ la stima ottenuta con un metodo non parametrico. Comunemente, la scelta ricade sul metodo del nucleo, ma non si tratta di una scelta vincolante.

2.1 Metodi basati sulle curve di livello della densità

In questa categoria si inseriscono quei metodi che associano i *cluster* alle regioni più densamente popolate dello spazio campionario, sulla scia di quanto anticipato da Hartigan (1975) con l'introduzione del concetto di *cluster* quali insiemi di livello connessi della funzione di densità.

Fissata una soglia λ non negativa, si definisce:

$$L(\lambda; f) = L(\lambda) = \{x \in \mathbb{R}^d : f(x) > \lambda\} \qquad (0 \leqslant \lambda \leqslant \max f) \tag{2.1}$$

come l'insieme di tutte le osservazioni per cui la densità è superiore al livello fissato. Quando $f(x)$ è unimodale, $L(\lambda)$ è una regione connessa per qualsiasi valore della soglia; altrimenti, se sconnessa, $L(\lambda)$ sarà composta da due o più componenti connesse, che corrispondono alle regioni intorno alle mode di f identificate sezionando la funzione di densità in corrispondenza di λ. Al variare di λ si delinea una struttura gerarchica nota come *albero dei cluster*.

Attraverso la Figura 2.1 si illustra il procedimento per ricavare la struttura ad albero dei *cluster* in un semplice caso unidimensionale. Per la densità in questione non esiste un valore univoco di λ che permetta di visualizzare contemporaneamente i tre *cluster*. Casi come questo mettono in luce l'impossibilità di fare affidamento ad un unico λ, motivando la necessità di considerare interamente l'intervallo in cui varia.

Partendo dal punto di massimo assoluto della funzione di densità, si lascia scorrere verso il basso la soglia λ fino a che non interseca il primo massimo locale, in corrispondenza di $\lambda = \lambda_1$. Fino a questa soglia gli insiemi di livello individuati sono connessi. Diminuendo la soglia, per ogni valore di λ compreso tra λ_1 e λ_2, le curve di livello sono invece caratterizzate da due componenti connesse, chiamate $G1$ e $G2$. La situazione descritta è illustrata nel primo grafico. Procedendo fino a λ_3 (grafico in alto a destra) le regioni intorno alle due mode più alte si fondono e le corrispondenti curve di livello risultano formate da un'unica componente connessa, tornando di nuovo ad essere due da λ_3 a λ_4. La nuova componente connessa è indicata con $G3$ e rappresenta la moda meno prominente. Infine, in corrispondenza di $\lambda = 0$ si ottiene una regione che coincide con l'intero asse dei numeri reali $\mathbb{R}$, il supporto dello spazio campionario. Gli insiemi $G1$, $G2$ e $G3$ definiscono i *nuclei* dei cluster, mentre le osservazioni che non vi appartengono saranno allocate tra i nuclei in fase successiva. Il partizionamento finale della popolazione, comprensivo della fase di classificazione delle unità esterne ai nuclei, è riportato nel grafico in basso a sinistra, dove risulta che $\mathbb{R} = R1 \cup R2 \cup R3$. Come è evidente, il punto in cui ogni regione

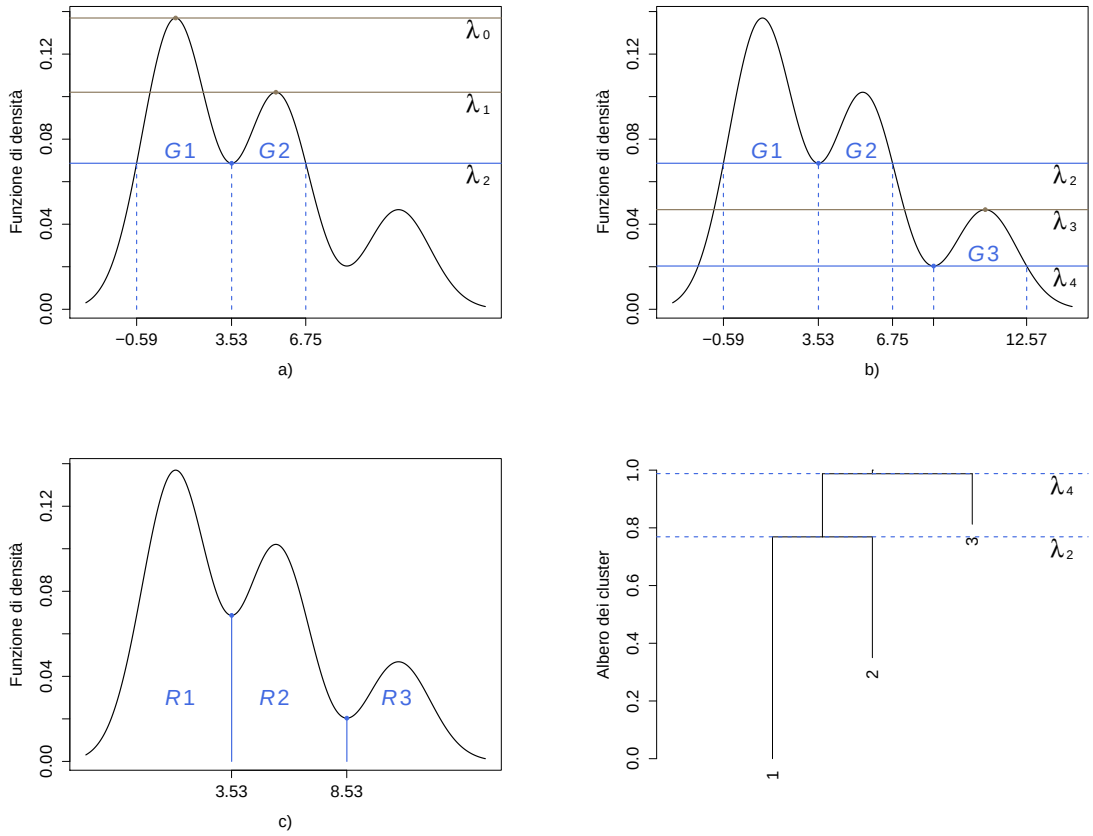


Figura 2.1: Identificazione dei *cluster* e struttura ad albero per d=1.

dà luogo a due componenti connesse costituisce un minimo locale per la funzione di densità.

Nell'ultimo grafico, in basso a destra, è riportato l'albero dei *cluster* rispetto ad una parametrizzazione alternativa di λ, ma equivalente. Infatti, l'asse delle ordinate non si riferisce al livello della densità, ma alla proporzione di unità con densità superiore a λ. La radice dell'albero, ossia il nodo in cima, corrisponde al livello $\lambda = 0$. I numeri $1, 2$ e 3 posti in prossimità delle foglie dell'albero indicano le tre mode emerse al variare di λ, che corrispondono, rispettivamente, ai nuclei $G1$, $G2$ e $G3$.

Sia ora definita con $p_\lambda = \int_{L(\lambda)} f(x)dx$ la probabilità associata alla regione $L(\lambda; f)$ e con λ_p la funzione inversa che consente di ricavare il livello λ tale che $\mathbb{P}\{L(\lambda_p)\} = p$. Osservando che l'insieme $L(\lambda)$, per ogni valore di λ, comprende un certo numero g di componenti connesse e che ad esso è associata una probabilità p, si può definire una corrispondenza tra le quantità p e g. Azzalini e Torelli (2007) presentano una funzione $g(p)$, alternativa all'albero, che fornisce il numero di componenti connesse di $L(\lambda_p)$ al variare di p in $[0, 1]$. Per valori estremi dell'intervallo vale $g(p) = 0$,

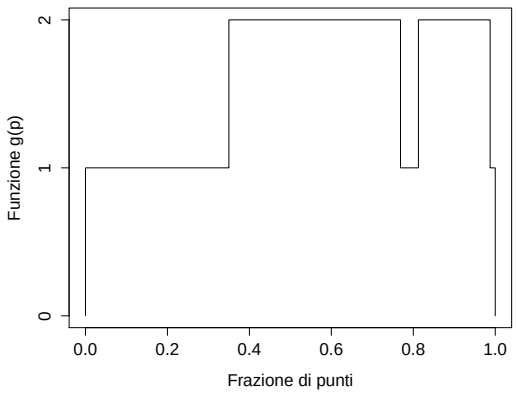


Figura 2.2: Funzione $g(p)$ associata alla densità rappresentata in Figura 2.1.

mentre per gli altri si ha $g(p) \geq 1$ (Figura 2.2). Quando la funzione di densità è unimodale, $g(p) \equiv 1$ per ogni $p \in (0, 1)$. Ogni incremento di $g(p)$ coincide con l'apparizione di una o più mode; un decremento, con la loro fusione. In particolare, un valore di p associato ad un incremento nella funzione denota la comparsa di tanti *cluster* quanto è il valore dell'incremento.

Si osserva che sia l'albero dei *cluster* sia la funzione $g(p)$ non solo indicano il numero di gruppi, ma permettono anche di evitare la necessità di selezionare λ.

2.1.1 Individuazione delle componenti connesse

Per quanto sia immediato stabilire se un punto appartiene ad un certo insieme di livello $L(\lambda)$, diventa sempre più complicato definire da quante e quali componenti connesse sia costituito l'insieme al crescere della dimensione dello spazio, già a partire da $d = 2$. Nei paragrafi successivi verranno descritti due metodi per affrontare questo problema. La teoria che entrambi sfruttano per rendere più trattabile il problema in uno spazio continuo e multidimensionale è quella dei grafi. I dati campionari allora ricoprono un duplice ruolo, impiegati prima per stimare la densità e poi per costruire il grafo.

Concettualmente, un grafo G è un modo di rappresentare relazioni esistenti tra coppie di oggetti mediante un insieme V di vertici e una collezione E di archi, che congiungono coppie di vertici appartenenti a V. In questo contesto, i vertici saranno associati alle osservazioni. In generale, un grafo è connesso se, per ogni coppia di vertici, si individua un percorso che li collega; se non è connesso, ogni sottografo connesso si chiama componente connessa. Sia indicato con $G(\lambda)$ il sottografo di G per una certa soglia λ, cioè l'insieme dei vertici con densità superiore a λ collegati da archi lungo i quali è soddisfatta una certa condizione in relazione a λ.

Allora, a meno dell'errore di stima, le componenti connesse di $G(\lambda)$ costituiscono un'approssimazione di quelle di $L(\lambda; f)$.

Ricorrere ai grafi consente di determinare la struttura ad albero dei *cluster*, considerando le componenti connesse di $G(\lambda)$ al variare della soglia λ. L'albero così ottenuto rappresenta un'approssimazione dell'albero che si otterrebbe a partire dalla stima della densità $\hat{f}$.

2.1.2 Il metodo di Azzalini e Torelli (2007)

In questo paragrafo si descrive il metodo proposto da Azzalini e Torelli (2007) per stabilire se $L(\lambda)$ sia connesso o, in caso contrario, da quali componenti connesse sia costituito.

Tassellatura di Voronoi e triangolazione di Delaunay

Dato un insieme S di n punti $\{x_1, \ldots, x_n\} \in \mathbb{R}^d$, si definisce *tassellatura di Voronoi* un partizionamento dello spazio in n poliedri $V(x_1), \ldots, V(x_n)$, eventualmente non limitati e tali che ogni punto del poliedro $V(x_i)$ sia più vicino a x_i che a tutte le altre $(n-1)$ osservazioni. Perché questo avvenga è necessario specificare una qualche misura di distanza; tipicamente, come per questo metodo, si considera quella euclidea. Le facce dei poliedri sono poligoni in $\mathbb{R}^{(d-1)}$, i loro spigoli linee in $\mathbb{R}^{(d-2)}$.

A partire dalla tassellatura di Voronoi è possibile determinare un'altra partizione dello spazio, definita *triangolazione di Delaunay*, che costituisce il grafo su cui si basa questo metodo. Si congiungono due elementi x_i e x_j di S se i corrispondenti poliedri $V(x_i)$ e $V(x_j)$ condividono almeno una porzione di una faccia. Il termine triangolazione trova origine nel fatto che, in due dimensioni, il partizionamento dà forma a dei triangoli. Si fornisce un'illustrazione di questi concetti nella Figura 2.3 per un insieme di punti in uno spazio bidimensionale. Operativamente, è possibile ottenere il diagramma di Delaunay senza dover prima definire la tassellatura di Voronoi.

Descrizione del metodo

Sia $\hat{f}(x)$ una stima non parametrica della funzione di densità $f(x)$ ottenuta a partire dal campione in esame S. Per implementare l'algoritmo non si richiede un metodo particolare di stima di f, ma piuttosto che $\hat{f}$ abbia valore finito e positivo almeno sui dati osservati e che goda, possibilmente, di buone proprietà statistiche.

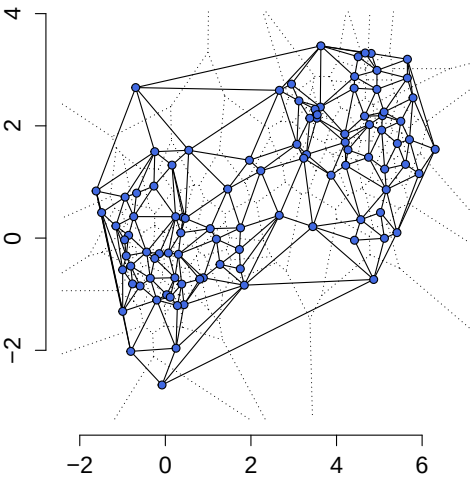


Figura 2.3: Le linee tratteggiate definiscono la tassellatura di Voronoi per un insieme di punti in due dimensioni, mentre le linee continue sovrapposte rappresentano la triangolazione di Delaunay.

Il corrispettivo empirico di $L(\lambda; f)$ è dato da:

$$L(\lambda; \hat{f}) = \hat{L}(\lambda) = \{x \in \mathbb{R}^d : \hat{f}(x) > \lambda\} \qquad (0 \leqslant \lambda \leqslant \max \hat{f}), \tag{2.2}$$

di cui si considera la restrizione:

$$S(\lambda; \hat{f}) = S(\lambda) = \{x_i \in S : \hat{f}(x_i) > \lambda\} \qquad (0 \leqslant \lambda \leqslant \max \hat{f}), \tag{2.3}$$

in linea con lo scopo dell'analisi che è partizionare il campione in esame S e non l'intero spazio $\mathbb{R}^d$. La frequenza relativa associata è $\hat{p}_\lambda = |S(\lambda)|/n$, dove $|\cdot|$ indica la cardinalità dell'insieme. È stato dimostrato che $\hat{L}(\lambda)$ è una stima consistente di $L(\lambda)$ per $n \to \infty$, a patto che $\hat{f}$ sia a sua volta una stima consistente di f (si rimanda a Wong e Lane (1981)).

Dopo aver costruito la triangolazione di Delaunay, per individuare le componenti connesse di $S(\lambda)$ si rimuovono dal grafo tutti i punti $x_i \notin S(\lambda)$ e gli archi con almeno un vertice tra questi punti. Si ottengono così gruppi di punti collegati dai restanti archi, che coincidono con le componenti connesse di $S(\lambda)$. Per n elevato, l'idea è che i poliedri emersi approssimino le componenti connesse di $\hat{L}(\lambda)$.

Questa procedura sarebbe idealmente da ripetere per ogni λ (o p) compreso tra 0 e 1; nella pratica, si considera una griglia finita di valori equamente spaziati entro quel *range*. Procedendo quindi in modo sequenziale si definisce la struttura ad albero in cui ciascuna foglia rappresenta il nucleo di un *cluster*.

Classificazione dei punti non allocati

Per completare la procedura, Azzalini e Torelli (2007) indicano un metodo per assegnare delle etichette ai punti che non appartengono ai nuclei dei *clusters*. Sia G il numero complessivo di gruppi identificati e $\hat{f}_j(x_0)$ la stima della densità in un punto x_0, privo di allocazione, basata sulle osservazioni campionarie che cadono nel nucleo dei gruppo j, per $j = 1, \ldots, G$. L'idea è quella di assegnare x_0 al gruppo per cui è maggiore il rapporto $r_j(x_0) = \hat{f}_j(x_0) / \max_{k \neq j} \hat{f}_k(x_0)$. Tra le varie possibili implementazioni di questa regola, una strategia intermedia tra costo computazionale e precisione consiste nell'ordinare i punti non allocati sulla base di $r_j(x)$, dividerli in blocchi di uguale ampiezza, allocare le osservazioni del blocco con maggiore densità ai gruppi esistenti, quindi procedere a delle nuove stime $\hat{f}_j$ per allocare il secondo gruppo e iterare l'algoritmo fino ad esaurire tutti i blocchi.

Aspetti computazionali

La procedura delineata incontra un limite computazionale nella fase di costruzione della triangolazione di Delaunay, per la quale il numero di operazioni richieste cresce con $n^{\lfloor d/2 \rfloor}$. In particolare, è stato verificato che per una dimensionalità $d \geq 7$ diventa complicato applicare la triangolazione già rispetto ad una numerosità di 200. Per superare questo ostacolo ed avere la possibilità di gestire dimensionalità più elevate, si cita la procedura alternativa con cui Menardi e Azzalini (2014) incorporano quest'algoritmo, in grado di traslare il problema ad un'unica dimensione qualsiasi sia quella di partenza, portando il costo computazionale proporzionale a $O(n^2)$.

2.1.3 Generalizzazione del metodo del legame singolo

Stuetzle e Nugent (2012) costruiscono una procedura in grado di fornire un'approssimazione della struttura ad albero formata dalle curve di livello della densità basata su grafi pesati.

Descrizione del metodo

Sia $\hat{f}_{ij}$ il valore minimo della densità stimata $\hat{f}$ lungo il segmento che collega le osservazioni x_i e x_j:

$$\hat{f}_{ij} = \min_{t \in [0,1]} \hat{f}((1-t)x_i + t x_j). \tag{2.4}$$

Indicato con G il grafo completo su tutte le osservazioni, i cui archi hanno peso $\hat{f}_{ij}$ e i vertici $\hat{f}_{ii}$, si definisce il sottografo di G per una certa soglia λ, $G(\lambda)$, come il

sottoinsieme di archi e vertici con peso $\hat{f}_{ij} > \lambda$. Per costruzione, i vertici di $G(\lambda)$ sono esattamente le osservazioni in $L(\lambda; \hat{f})$.

A questo punto, le componenti connesse di $G(\lambda)$ al variare del livello λ restituiscono una struttura ad albero, che rappresenta un'approssimazione di quella di $\hat{f}$. È possibile definire l'albero in modo ricorsivo, considerando che ciascun nodo N è associato ad un sottografo $\tilde{D}(N)$ di G per un certo livello della densità $\lambda(N)$. La radice dell'albero rappresenta l'intero grafo e il livello della densità associato è $\lambda(N) = 0$. Per determinare i discendenti di un nodo N si cerca il più basso livello λ_d per cui $G(\lambda) \cup \tilde{D}(N)$ ha due o più componenti connesse. Se non esiste una tale soglia, allora N sarà una foglia dell'albero, altrimenti si indicano con $C_1, \ldots, C_k$ le componenti connesse di $G(\lambda) \cup \tilde{D}(N)$ per il livello λ_d e si applica iterativamente la regola.

Sia ora denominato con T l'albero ricoprente massimo di G, vale a dire un grafo connesso e senza cicli contenente tutti i vertici di G e con il massimo peso totale tra tutti gli alberi possibili. Formalmente, l'albero T sarà tale da massimizzare la somma:

$$w(T) = \sum_{(x_i,x_j)\ in\ T} w(x_i, x_j),$$

dove $w(x_i, x_j)$ è il peso assegnato all'arco che collega x_i e x_j. È stato dimostrato che due vertici appartengono alla stessa componente connessa di $G(\lambda)$ se e soltanto se si trovano nella stessa componente connessa di $T(\lambda)$ (Stuetzle e Nugent (2012), Paragrafo 4). Questo implica che l'algoritmo possa essere implementato direttamente su T, piuttosto che su G.

Operativamente, poiché i pesi da attribuire agli archi non sono noti, vengono approssimati attraverso un problema di ottimizzazione che sfrutta una griglia di ricerca: $\hat{f}_{ij}$ sarà sostituito dal valore minimo di $\hat{f}$ su una griglia regolare di punti che connettono x_i e x_j. L'algoritmo calcola l'albero ricoprente massimo a partire dal campione a disposizione e, iterativamente, per ogni soglia λ considerata, rimuove archi e vertici con peso inferiore. I valori per λ a cui la procedura fa riferimento sono i pesi assegnati agli archi del grafo. Una volta terminata, si potranno contare un certo numero di componenti connesse, individuate nell'albero dalle foglie; le restanti osservazioni, quelle non allocate, saranno associate ai *cluster* secondo qualche criterio di classificazione.

Potatura dell'albero

Come diretta conseguenza dell'errore di cui necessariamente ogni metodo di stima della densità risulta affetto, è inevitabile che affiorino delle componenti connesse

spurie e che l'albero debba pertanto essere potato. Stuetzle e Nugent propongono alcune misure che consentono di conservare solo i *cluster* con una *prominenza* rilevante, da intendersi sia come estensione spaziale che come altezza raggiunta dalla moda rispetto all'avvallamento che la separa dalle altre mode. La prominenza di un *cluster* può essere definita tramite la sua massa in eccesso:

$$E(N) = \int_{D(N)} (f(x) - \lambda(N))dx, \tag{2.5}$$

il cui analogo campionario sarà dato da:

$$\tilde{E}(N) = \frac{1}{n}\sum_{i=1}^{n} I(x_i \in \tilde{D}(N)) \left(1 - \frac{\lambda(N)}{f(x_i)}\right). \tag{2.6}$$

Per potare l'albero si fissa una soglia γ rispetto alla quale rimuovere tutti i nodi aventi massa in eccesso inferiore.

Un'altra misura utile per valutare la prominenza di una moda è semplicemente fornita dalla sua massa:

$$S(N) = \int_{D(N)} f(x)dx, \tag{2.7}$$

stimata tramite:

$$\tilde{S}(N) = \frac{1}{n}\sum_{i=1}^{n} I(x_i \in \tilde{D}(N)). \tag{2.8}$$

Entrambe le misure sono monotone: se N_2 è un nodo discendente di N_1 allora $\tilde{E}(N_2) < \tilde{E}(N_1)$ ($\tilde{S}(N_2) < \tilde{S}(N_1)$). Si noti, tuttavia, che non esiste un metodo automatico per determinare il valore ideale di γ, la cui scelta resta soggettiva. La soluzione avanzata da Stuetzle e Nugent consiste nell'ordinare in senso decrescente la massa in eccesso per i nodi dell'albero dei *cluster*. Si osserva, tipicamente, che un esiguo numero di valori grandi è seguito da tanti valori più piccoli. Una massa in eccesso grande indica una separazione tra due mode prominenti, mentre, se piccola, le mode in questione sono spurie e probabilmente imputabili alla variabilità della stima della densità. Quindi, il suggerimento è quello di cercare lo scarto più evidente tra i valori della massa in eccesso e selezionare quello maggiore per la soglia γ da usare per potare l'albero.

Collegamento con il metodo del legame singolo

Per spiegare il motivo per cui l'algoritmo è stato definito dagli autori come una generalizzazione del metodo del legame singolo si richiama lo stimatore della densità ottenuto con il metodo del vicino più prossimo (1.38). Secondo questo metodo, la

densità stimata in un punto sarà tanto maggiore quanto più quel punto è vicino alle altre osservazioni. I vertici del grafo G avranno allora peso $\hat{f}_{ii} = \infty$.

Si dimostra che l'albero ottenibile da G è isomorfo al dendrogramma ricavato applicando il metodo del legame singolo, ovvero sono in corrispondenza biunivoca (Stuetzle e Nugent (2012), Capitolo 6). Questo implica che l'algoritmo di Stuetzle e Nugent implementato su una stima della densità calcolata con il metodo del vicino più prossimo produce lo stesso risultato del metodo del legame singolo. Tuttavia, una differenza sostanziale risiede nella modalità di estrazione dei *cluster*. Tipicamente, infatti, il dendrogramma viene tagliato in corrispondenza di un certo livello in modo da identificare i *cluster* sulla base delle osservazioni che si uniscono fino a quella soglia. Con questa tecnica tendono a delinearsi molti gruppi poveri di osservazioni e solo uno o pochi particolarmente numerosi; inoltre, come già è emerso, un'unica soglia spesso rischia di non rivelare tutte le mode della funzione di densità. Per rimediare a questi problemi, una valida soluzione è offerta dal metodo del legame singolo generalizzato, che prevede di potare l'albero sulla base di misure più appropriate che valutano la prominenza dei *cluster* (la massa e la massa in eccesso). Si noti infine che, con riferimento al metodo del vicino più prossimo, risulta:

$$\tilde{E}(N) = \frac{1}{n} \sum_{i=1}^{n} I(x_i \in \tilde{D}(N)) \left(1 - \frac{\lambda(N)}{f(x_i)}\right) = \tilde{S}(N), \tag{2.9}$$

essendo $\hat{f}(x_i) = \infty$.

L'idea di estrarre i *cluster* dal dendrogramma ottenuto con il metodo del legame singolo attraverso la rimozione di quelli con massa non rilevante era già stata trattata da Stuetzle nel 2003.

2.2 Metodi basati sulla ricerca delle mode

Questa classe di metodi individua i *cluster* attraverso una ricerca esplicita delle mode della funzione di densità. Si distingue dalle tecniche precedenti perchè non fa affidamento ai grafi né al concetto di insiemi di livello della funzione di densità; piuttosto, prevede una procedura (numerica) di individuazione dei punti di massimo locale della funzione. Le mode sono i rappresentanti dei gruppi e le altre osservazioni vengono allocate alla moda di pertinenza a seconda della procedura. In seguito si descrive l'algoritmo *mean-shift*.

2.2.1 Il metodo *mean-shift*

Il metodo presentato in questo paragrafo e noto come algoritmo *mean-shift* è stato introdotto da Fukunaga e Hostetler (1975).

Si tratta di una procedura iterativa che, ad ogni passo, sposta ciascuna osservazione secondo la direzione del gradiente della densità calcolato in quel punto di una quantità (detta *mean-shift*) proporzionale al modulo del gradiente stesso, producendo una sequenza convergente che trasporta ogni punto dalla posizione iniziale al massimo locale più vicino della funzione di densità. Due punti appartengono allo stesso *cluster* quando la sequenza realizzata a partire da essi converge alla stessa moda di $f(x)$, ossia quando fanno parte del dominio di attrazione dello stesso massimo locale della funzione. Nel grafico a sinistra della Figura 2.4 si illustra un'idea di come opera questo algoritmo in una dimensione; a destra, ne viene mostrata un'applicazione in due dimensioni.

Con riferimento alla notazione già introdotta, si denota ora con ∇f il vettore gradiente della funzione di densità $f(x)$:

$$\nabla f = \frac{\partial f}{\partial x} = \left(\frac{\partial f}{\partial x_1}, \dots, \frac{\partial f}{\partial x_d} \right)^\top . \tag{2.10}$$

Siano x_i^j, per $i = 1, \dots, n$, le coordinate relative alle osservazioni del campione al passo j lungo il percorso che le conduce alla moda di pertinenza; sia quindi $x_i^0 \equiv x_i$ la situazione iniziale, al passo 0. Per ciascun punto sarà costruita una sequenza $(x_i^0, x_i^1, x_i^2, \dots)$ in accordo con la seguente definizione:

$$x_i^{j+1} = x_i^j + a \nabla ln f(x_i^j) = x_i^j + a \frac{\nabla f(x_i^j)}{f(x_i^j)}, \tag{2.11}$$

dove a è una costante positiva scelta per garantire la convergenza del metodo. Si indica il secondo addendo come vettore del *mean-shift*. Nell'espressione, in luogo del gradiente regolare, viene inserito quello normalizzato, così da accelerare la convergenza per i punti situati in regioni a bassa densità, dove il gradiente assume valore inferiore. Infatti, la grandezza dello spostamento al passo j dipende solo dalla forma della densità sottostante. Quindi, quanto più il valore di $f(x)$ è piccolo (nelle code della densità o in prossimità di minimi locali) tanto più la quantità $\nabla ln f(x) = \nabla f(x)/f(x)$ risulterà maggiore di $\nabla f(x)$. Esiste un altro aspetto rilevante che giustifica la scelta del gradiente normalizzato. Considerando una densità

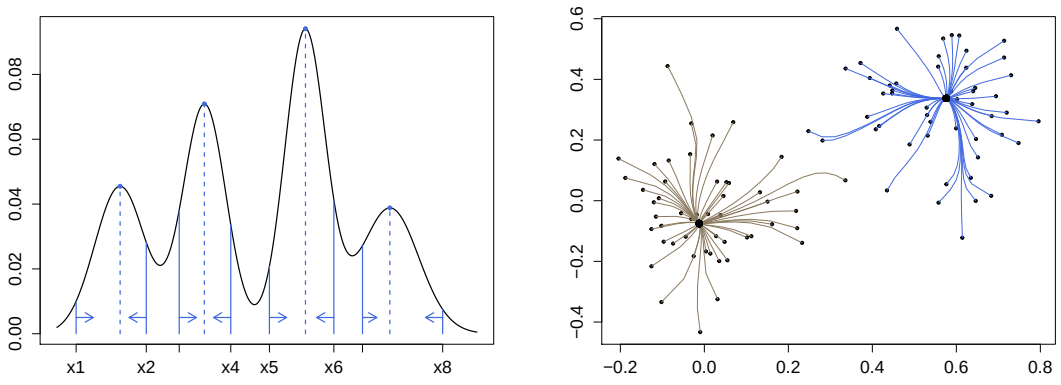


Figura 2.4: *Mean-shift clustering*: a sinistra per una dimensione, a destra per due dimensioni.

gaussiana, con matrice di covarianza identica e media μ:

$$f(x) = (2\pi)^{-n/2} exp\left\{-\frac{1}{2}(x-\mu)^\top (x-\mu)\right\}, \tag{2.12}$$

e assumendo $a = 1$, l'espressione (2.11) diventa:

$$x_i^1 = x_i + \nabla ln f(x_i) = x_i - (x_i - \mu) = \mu, \tag{2.13}$$

risultato che mostra come una sola iterazione sia sufficiente perchè i *cluster* si condensino in un unico punto. Si intuisce, allora, che l'algoritmo possa essere particolarmente efficace per popolazioni con gruppi di forma gaussiana.

Un modo semplice di derivare il *mean-shift* consiste nel calcolare prima una stima non parametrica della funzione di densità, e poi ricavarne il gradiente. Si considera quindi una funzione *kernel* che sia differenziabile, per poter stimare il gradiente della densità come il gradiente della densità stimata. Risulta:

$$\hat{\nabla} f(x) \equiv \nabla \hat{f}(x) = \frac{1}{nh^{(d+1)}} \sum_{i=1}^{n} \nabla K\left(\frac{x - x_i}{h}\right), \tag{2.14}$$

dove $K(\cdot)$ può assumere varie forme (Paragrafo 1.1.2). Si farà ora riferimento ad un *kernel* gaussiano di media nulla e varianza identica:

$$K(x) = (2\pi)^{-n/2} exp\left\{-\frac{1}{2}x^\top x\right\}, \tag{2.15}$$

il cui gradiente assume la forma:

$$\nabla K(x) = (2\pi)^{-n/2} \left(-x_1 exp\left\{ -\frac{1}{2} x^\top x \right\}, \ldots, -x_n exp\left\{ -\frac{1}{2} x^\top x \right\} \right)^\top . \qquad (2.16)$$

Sostituendo (2.16) in (2.14) al posto di $\nabla K(\cdot)$, si ottiene la stima del gradiente della densità:

$$\hat{\nabla} f(x) = \frac{1}{n} \sum_{i=1}^{n} \frac{(x_i - x)}{(2\pi)^{n/2} h^{(n+2)}} exp\left\{ -(x - x_i)^\top \left(\frac{x - x_i}{2h^2} \right) \right\}. \qquad (2.17)$$

Un'interpretazione intuitiva di quest'espressione è che, essenzialmente, si tratta di una misura pesata dello spostamento medio delle osservazioni rispetto al punto x, dove il peso è costituito dal fattore $(2\pi)^{-n/2} h^{-(n+2)} exp\{-(x - x_i)^\top (x - x_i)/2h^2\}$. La media campionaria di questi spostamenti pesati è quindi assunta come stima del gradiente.

Anche per questo metodo l'efficacia nell'identificazione dei *cluster* risulta fortemente influenzata dalla scelta del parametro di lisciamento (sia esso uno scalare, un vettore o una matrice), da scegliere in modo che sia ottimale per la stima del gradiente della densità.

2.2.2 Altri metodi

Sebbene il metodo *mean-shift* sia probabilmente il più diffuso, in letteratura se ne trovano molti altri che si inseriscono nello stesso approccio.

Un'alternativa è offerta, per esempio, dal metodo di Li *et al.* (2007). Si parte dalla considerazione che, per sua natura, la stima della densità ottenuta con il metodo *kernel* riveli una struttura riconducibile ad una mistura di densità, seppur formata da tante componenti quante sono le osservazioni, con centri che sono quindi noti. Questo consente di sviluppare un approccio che unisce i vantaggi del *clustering* parametrico a quelli derivanti dal *clustering* non parametrico. Data quindi la struttura a mistura della stima *kernel*, gli autori discutono l'utilizzo di un algoritmo denominato *Modal Expectation Maximization, (MEM)*, variante del più noto *EM* impiegato nei metodi parametrici per la massimizzazione della verosimiglianza e, invece, finalizzato alla ricerca dei massimi locali della funzione di densità stimata; segue l'evidente analogia con il metodo *mean-shift*. Gli autori propongono, inoltre, un'estensione gerarchica del *clustering*. Al livello l della gerarchia viene costruita una stima *kernel* con ampiezza della finestra $h_l > h_{l-1}$ e l'implementazione dell'algoritmo *MEM* consente di ottenere le mode collegate a quelle individuate al livello

$l-1$. Quindi, per costruzione, i due corrispondenti partizionamenti ai livelli $l-1$ e l risultano nidificati.

Capitolo 3

Un'analisi empirica

In questo capitolo si valuta, da un punto di vista empirico, il comportamento dell'approccio non parametrico all'analisi di raggruppamento sia mediante uno studio di simulazione, sia mediante un'applicazione a dati reali.

3.1 Studio di simulazione

Preso nota del fatto che le tecniche di *clustering* non parametrico dipendono dalla fase di stima della densità e che questa, a sua volta, risulta influenzata dalla scelta del parametro di lisciamento, l'obiettivo centrale dello studio sarà proprio quello di valutare il comportamento dell'approccio in esame in relazione a diverse procedure di selezione del vettore di lisciamento. Infatti, per un dato insieme di osservazioni, la corretta identificazione dei *cluster* non può prescindere dalla ricerca del grado ottimale di lisciamento per la stima della densità.

Per consentire una maggiore flessibilità, senza tuttavia imbattersi nell'eccessivo sforzo computazionale che richiederebbe l'uso di una matrice completa di lisciamento, si è scelto di utilizzare una parametrizzazione diagonale, ovvero $H = Diag\{h_1^2, \ldots, h_d^2\}$. Saranno considerati i seguenti criteri per la scelta del vettore di lisciamento:

- vettore asintoticamente ottimale in riferimento alla distribuzione normale *(Hns)*;
- il metodo della convalida incrociata dei minimi quadrati *(Hlscv)*;
- il metodo *plug-in (Hpi)*;
- il metodo della convalida incrociata lisciata *(Hscv)*.

Il secondo obiettivo dello studio consiste nel confrontare i due approcci descritti nel capitolo precedente per implementare la formulazione non parametrica dell'ana-

lisi di raggruppamento, ovvero l'approccio basato sulla determinazione delle componenti connesse associate alle curve di livello e l'approccio basato sulla ricerca delle mode. In particolare, si prendono in considerazione il metodo proposto da Azzalini e Torelli (che sarà indicato con *Pdf Cluster*, dal nome del pacchetto in R [1]) e l'algoritmo *mean-shift* (a cui si farà riferimento con la sigla *Ms Cluster*). Nello studio non è stato incluso il metodo di Stuetzle e Nugent (2012) per un motivo di natura strettamente computazionale; infatti, il relativo pacchetto (*gslclust*) in R accetta unicamente uno scalare come valore per il parametro di lisciamento da usare per la stima della densità, obbligando ad una preventiva standardizzazione dei dati. Per lo stesso motivo, non si è considerato nell'analisi il metodo di Li *et al.* (2007), precedentemente descritto.

Da un punto di vista operativo, i due metodi vengono resi confrontabili ricavando il grado di lisciamento ottimale per la stima della funzione di densità per il metodo di Azzalini e Torelli e per la stima del gradiente della densità per l'algoritmo *mean-shift*. In ogni contesto, si adotta un *kernel* di forma gaussiana, con media nulla e matrice di varianza pari alla matrice identità.

I risultati verranno valutati in termini di qualità rispetto all'identificazione di gruppi caratterizzati da forme, livelli di separazione e variabilità diversi. Per ciascuna configurazione dei gruppi si mantiene fisso il loro numero, imposto pari a due. Tutti i risultati saranno poi raccolti per diverse numerosità e dimensioni campionarie.

Di seguito vengono elencate le distribuzioni scelte per estrarre i campioni analizzati nello studio:

- gruppi sferici: ogni gruppo viene generato da una distribuzione normale $N(\mu_g, kI_d)$, $g = 1, 2$, con $\mu_1 = 0 \cdot 1_d$ e $\mu_2 = 5 \cdot 1_d$ (1_d è un vettore unitario di dimensione d). La matrice di varianza e covarianza è la matrice identica I_d di ordine d moltiplicata per uno scalare k. Si tratta di gruppi sferici la cui sovrapposizione è, eventualmente, legata al valore di k:
  - si fissa $k = 1$ perchè i due gruppi siano ben separati;
  - si fissa $k = 5$ affinché la maggiore dispersione delle unità possa indurre una parziale sovrapposizione nei due gruppi;
- gruppi concavi: ogni gruppo viene generato simulando uniformemente 3 ipercubi di lato unitario aventi una faccia in comune. Il vertice più esterno dei gruppi è posizionato in $0 \cdot 1_d$ e in $4 \cdot 1_d$.

[1] R (R Development Core Team, 2013) è il linguaggio di programmazione utilizzato per implementare lo studio.

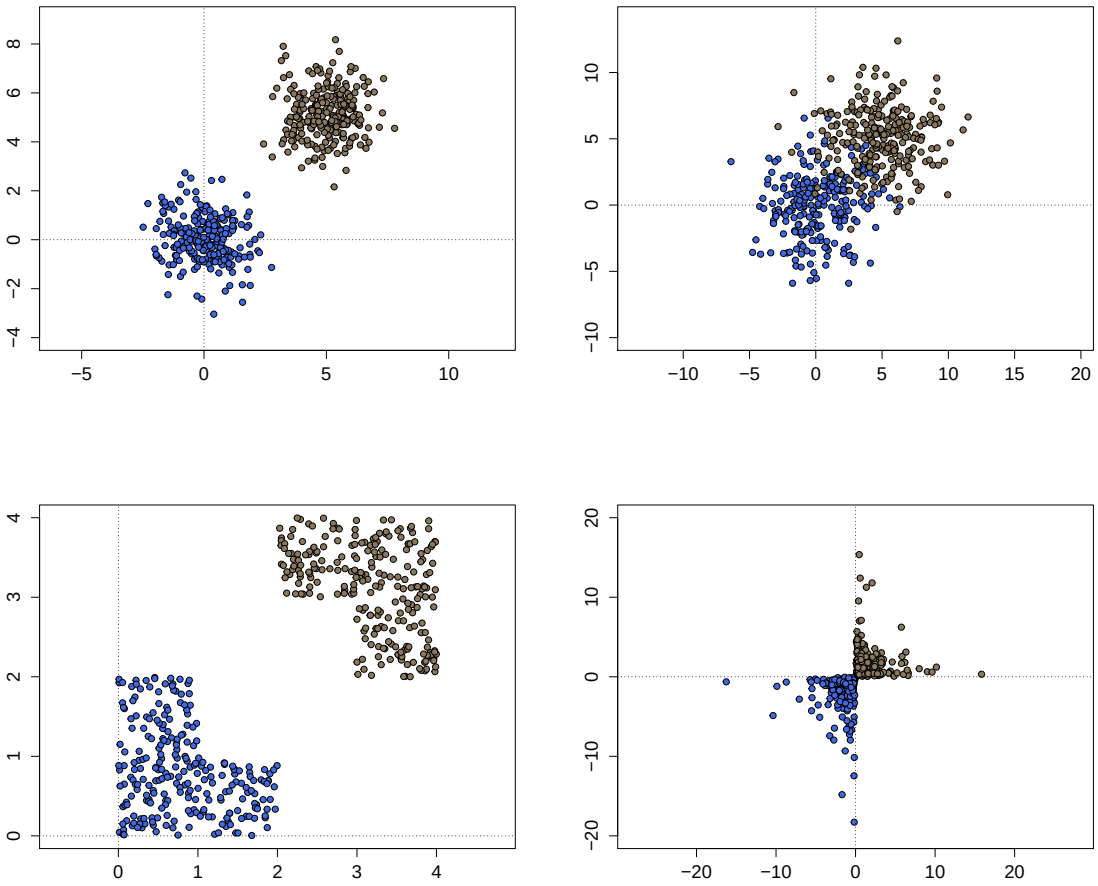


Figura 3.1: Un esempio di campioni generati dalle distribuzioni bi-variate utilizzate nello studio di simulazione. In alto a sinistra: gruppi sferici e ben separati; a destra: gruppi sferici e parzialmente sovrapposti. In basso a sinistra: gruppi concavi; a destra: gruppi asimmetrici.

- gruppi asimmetrici: ogni gruppo viene generato dalla funzione di densità di una variabile y_g tale che $y_1 = e^x$ e $y_2 = -e^x$ con x avente distribuzione normale standard.

Per avere un'idea della forma di queste distribuzioni si veda la Figura 3.1, in cui sono rappresentati in uno spazio bidimensionale quattro campioni ciascuno di ampiezza $n = 500$.

Le numerosità campionarie valutate in ciascun caso sono: $n \in \{250, 500, 1000\}$, mentre per la dimensione dello spazio si considera: $d \in \{2, 3, 6\}$.

Per ciascuno scenario si eseguono 500 replicazioni. Al termine di ogni iterazione viene calcolato l'indice *ARI (Adjusted Rand Index)* (Hubert e Arabie, 1985), che fornisce una misura di accordo tra due partizioni dello stesso insieme di dati, basandosi su come coppie di elementi sono classificate in una tabella di contingenza.

In questa analisi le due partizioni oggetto di confronto saranno da un lato quella individuata dall'algoritmo di *clustering* e dall'altro quella che si assume nota (cioè fissata da un criterio esterno: nel processo di generazione delle unità si include l'assegnazione automatica, per ognuna, di una specifica classe). L'indice ARI è stato definito a partire da una misura R (*Rand Index*, Rand (1971)), rispetto alla quale rappresenta una versione corretta. Si ricava ora l'espressione dell'indice R. Siano $Z = \{Z_i, \ldots, Z_r\}$ e $Y = \{Y_1 \ldots, Y_q\}$ due diverse partizioni (rispettivamente in r e in q sottogruppi) dello stesso insieme $S = \{x_1, \ldots, x_n\}$ composto di n unità, tali che $\bigcup_{i=1}^{r} Z_i = \bigcup_{j=1}^{q} Y_j = S$ e $Z_i \cap Z_{i'} = Y_j \cap Y_{j'} = \emptyset \; \forall \; i \neq i', \; j \neq j'$. Si può assumere che Z sia il partizionamento ricavato per via del criterio esterno, mentre che Y sia il risultato della procedura di *clustering*. Siano allora:

- a = numero di coppie di osservazioni appartenenti allo stesso gruppo sia di Z che di Y;
- b = numero di coppie di osservazioni appartenenti a gruppi distinti sia di Z che di Y;
- c = numero di coppie di osservazioni appartenenti allo stesso gruppo di Z e a gruppi diversi di Y;
- d = numero di coppie di osservazioni appartenenti a gruppi diversi di Z e allo stesso gruppo di Y.

Si definisce l'indice R come:

$$R = \frac{a+b}{a+b+c+d} \qquad \in [0,1], \tag{3.1}$$

dove, intuitivamente, la quantità al numeratore rappresenta il numero di accordi tra i due raggruppamenti, mentre $(c+d)$ il numero di disaccordi. L'indice vale 1 quando le due partizioni coincidono esattamente. L'ARI rappresenta un aggiustamento dell'indice R definito per garantire che il valore atteso sia nullo quando Z e Y sono due partizioni casuali. Il massimo che può assumere è ancora 1, in caso di raggruppamenti uguali, ma può anche avere valore negativo.

Per quanto concerne l'aspetto strettamente computazionale, in R sono stati utilizzati i pacchetti: *mvtnorm* (Genz *et al.*, 2014), per generare dati da distribuzioni normali multivariate, *ks* (Duong, 2015), per ottenere i vettori di lisciamento, *pdfCluster* (Azzalini e Menardi, 2014), per implementare il metodo di Azzalini e Torelli (2007) e calcolare l'indice ARI, *LPCM* (Einbeck e Evers, 2013), per l'algoritmo *mean-shift*.

Si rimanda all'Appendice per avere visione dei codici che sono stati sviluppati per generare i gruppi.

3.1.1 Risultati

Si prendono ora in esame i risultati dello studio di simulazioni, riportati nelle tabelle da 3.2 a 3.13.

È immediato notare quanto la forma dei gruppi influenzi la capacità di entrambi i metodi nel fornire un partizionamento corretto delle unità.

Rispetto ai gruppi di forma sferica (Tab. $3.1 - 3.6$), l'aumento del valore atteso dell'ARI al crescere di d sembra, apparentemente, in contrasto con quel fenomeno, noto come *maledizione della dimensionalità*, che consiste nel deterioramento dei risultati di una procedura statistica quando la dimensione dello spazio, e quindi la dispersione dei dati, aumentano. In realtà, per come sono stati definiti i vettori delle medie dei due gruppi, con la dimensione dello spazio da un lato cresce la dispersione tra i dati e dall'altro si genera uno spazio vuoto sempre meglio definito tra i due *cluster*, con l'effetto di isolarli e compensare i limiti altrimenti causati dalla *maledizione della dimensionalità*. Dunque, con gruppi ben separati si osserva un comportamento ottimale di entrambi i metodi rispetto ad ogni combinazione di n e di d, per (quasi) ogni scelta del vettore di lisciamento. Risalta infatti un unico caso anomalo, in due dimensioni, con *Ms Cluster* e *Hlscv*. L'ARI, molto basso, diminuisce al crescere di n; in particolare, il metodo trova nella maggior parte dei casi un solo gruppo e, in un'esigua minoranza, arriva a identificarne anche più di 40. Questa estrema variabilità rappresenta un limite del metodo della convalida incrociata dei minimi quadrati cui già si era accennato nel Paragrafo 1.2 e probabilmente dipende dalla stima *leave-one-out* della densità usata nel processo di stima della quantità da minimizzare (1.24). Infatti, se viene escluso dalla stima della densità un valore anomalo, allora questa può subire forti variazioni.

Quando si introduce tra i gruppi una parziale sovrapposizione (Tab. $3.4 - 3.6$), necessariamente, i metodi di raggruppamento incontrano una difficoltà crescente nell'allocazione di quelle osservazioni che cadono nella zona di sovrapposizione. L'ARI assume in generale un valore medio inferiore e una deviazione standard superiore rispetto al caso analizzato precedentemente, ma mantiene comunque uno standard medio-alto, con la tendenza a crescere con la dimensione dello spazio per i motivi già delineati. Si osserva che, con la massima dimensione considerata, l'algoritmo *Ms Cluster*, sebbene identifichi quasi perfettamente i due gruppi, alloca presumibilmente tutti gli *outliers* ad altri gruppi, con il risultato di sovrastimare eccessivamente il numero reale di quelli esistenti. Di fatto, in uno spazio con più coordinate, di fronte

ad una maggiore dispersione dei dati, quanto più il vettore di lisciamento è piccolo e sotto-liscia la stima della densità, tanto più è probabile che sorgano molte mode spurie, che l'algoritmo *mean-shift*, avendo noto come procede, associa erroneamente a *cluster*. Invece, il metodo *Pdf Cluster*, oltre ad avere valori molto alti per l'ARI, individua anche correttamente il numero di gruppi.

Tenendo presente che tutti i risultati ottenuti per gruppi di forma gaussiana, con e senza sovrapposizione, sono molto soddisfacenti, risultano ancora più accurati i raggruppamenti ricavati con i vettori di lisciamento *Hns* e *Hscv* attraverso entrambi i metodi di raggruppamento e sotto un punto di vista globale, cioè per ciascun valore esaminato di n e d.

Si analizza ora il caso di gruppi concavi (Tab. $3.7-3.9$). A dispetto di un'accresciuta complessità del problema di *clustering*, dovuta alla forma non standard dei gruppi, il comportamento di entrambi i metodi risulta comunque positivo, seppur con un evidente peggioramento nella capacità di identificazione dei *cluster* al crescere di d. In due dimensioni operano particolarmente bene, soprattutto il metodo *Ms Cluster*, per il quale l'ARI è massimo in corrispondenza di ogni n sia con *Hns* che con *Hlscv*. Inoltre, sempre in relazione a questo metodo, la scelta di *Hscv* fornisce i risultati migliori rispetto ad ogni contesto valutato. In linea di massima, *Ms Cluster* risulta preferibile a *Pdf Cluster* per ogni vettore di lisciamento considerato, con la sola esclusione di *Hpi*.

Per quanto riguarda l'ultima forma in esame, quella di gruppi caratterizzati da forte asimmetria (Tab. $3.10-3.12$), è possibile ottenere buoni risultati, ma il criterio di selezione del parametro di lisciamento assume maggior peso. Al crescere di d il calo della *performance* dei due metodi non mostra una rilevanza significativa. Anzi, il comportamento peggiore si registra con *Ms Cluster* in due dimensioni, rispetto ai vettori *Hns* e *Hlscv*. Si nota che, sebbene il valore medio dell'ARI sia complessivamente più alto con *Ms Cluster*, questo metodo sovrastima notevolmente il numero di gruppi. Infatti, i campioni generati da quest'ultima distribuzione presentano un elevato numero di *outliers* che evidentemente rendono critica la corretta identificazione dei gruppi (anche, in pochi casi, da parte di *Pdf Cluster*). Di conseguenza, la tendenza è quella di definire tanti piccoli gruppi oltre agli unici due realmente esistenti. Questo caratterizza il comportamento dell'algoritmo *Ms Cluster* per ogni dimensionalità in questione, peggiorando notevolmente per $d = 6$, dove si ha un'esplosione di gruppi con $n = 1000$ e, in particolare, rispetto a *Hpi* e *Hscv*.

Il metodo *Pdf Cluster* si caratterizza per una maggiore robustezza in presenza di *outliers*, anche quando $d = 6$. Al crescere di n e di d i risultati tendono ad uguagliarsi per ogni vettore di lisciamento considerato. Nel complesso non si ottengono risultati

ottimali, ma sicuramente accettabili. Si rileva che *Hscv* può essere la scelta migliore per quasi ogni combinazione di d e di n.

In conclusione, si possono trarre le seguenti valutazioni. Rispetto all'identificazione di gruppi di forma sferica i risultati dell'algoritmo *Pdf Cluster* sono leggermente più accurati di quelli derivanti dall'altro metodo. Il comportamento di entrambi è comunque ottimale per ogni criterio di selezione di h ma, nel dettaglio, la performance migliore si rileva con *Hns* e con *Hscv*. In presenza di sovrapposizione fra i gruppi il valore medio dell'ARI risulta generalmente inferiore rispetto a quando è assente. Per i gruppi concavi è emerso che la combinazione in grado di adattarsi ad ogni d e ad ogni n è data dall'algoritmo *Ms Cluster* e dal vettore *Hscv*, altrimenti si può notare che comunque questo metodo di raggruppamento è più soddisfacente di *Pdf Cluster*. Quando, infine, i gruppi sono segnati da una forte asimmetria, i due metodi operano in modo equiparabile fino a che la dimensionalità è ridotta, ma si dimostra più efficace *Ms Cluster* per 6 dimensioni. Quanto alla scelta del vettore di lisciamento, ancora *Hscv* sembra essere il più opportuno.

Tra i due metodi di raggruppamento si osserva, inoltre, un diverso comportamento rispetto alla presenza di *outliers*. *Pdf Cluster* è più robusto e anche quando la dispersione dei dati è elevata tende a identificare correttamente il numero dei gruppi; invece, l'algoritmo *Ms Cluster*, che risulta essere più sensibile, alloca gli *outliers* ad altri gruppi.

	Pdf Cluster			*Ms Cluster*		
	n=250	n=500	n=1000	n=250	n=500	n=1000
Hns	0.999 (0.003)	0.999 (0.003)	0.999 (0.002)	0.999 (0.003)	0.999 (0.003)	0.999 (0.002)
	2.000 (0.000)	2.000 (0.000)	2.000 (0.000)	2.000 (0.000)	2.000 (0.000)	2.000 (0.000)
Hlscv	0.912 (0.164)	0.926 (0.151)	0.930 (0.155)	0.450 (0.488)	0.257 (0.430)	0.089 (0.278)
	2.466 (1.005)	2.384 (0.937)	2.386 (1.041)	2.406 (4.406)	2.672 (7.145)	2.314 (8.529)
Hpi	0.984 (0.062)	0.982 (0.062)	0.984 (0.059)	0.999 (0.004)	0.999 (0.003)	0.999 (0.002)
	2.066 (0.256)	2.070 (0.255)	2.066 (0.249)	2.050 (0.227)	2.102 (0.316)	2.178 (0.455)
Hscv	0.993 (0.038)	0.991 (0.042)	0.989 (0.047)	0.999 (0.003)	0.999 (0.003)	0.999 (0.002)
	2.026 (0.159)	2.032 (0.176)	2.040 (0.196)	2.000 (0.000)	2.012 (0.109)	2.028 (0.177)

Tabella 3.1: Risultati delle simulazioni per gruppi sferici e separati, con $d = 2$. La tabella di sinistra è relativa al metodo *Pdf Cluster*, quella di destra all'algoritmo *Ms Cluster*. Per entrambe, in corrispondenza di ogni vettore di lisciamento h, la prima riga riporta il valore medio dell'ARI e la seconda il numero medio di gruppi identificati al variare della numerosità n. Tra parentesi è indicata la corrispondente deviazione standard.

	Pdf Cluster			*Ms Cluster*		
	n=250	n=500	n=1000	n=250	n=500	n=1000
Hns	1.000 (0.000)	1.000 (0.001)	1.000 (0.000)	1.000 (0.001)	1.000 (0.001)	1.000 (0.000)
	2.000 (0.000)	2.000 (0.000)	2.000 (0.000)	2.000 (0.000)	2.000 (0.000)	2.000 (0.000)
Hlscv	0.958 (0.104)	0.954 (0.117)	0.966 (0.090)	1.000 (0.001)	1.000 (0.001)	1.000 (0.000)
	2.202 (0.508)	2.228 (0.633)	2.160 (0.446)	2.000 (0.000)	2.000 (0.000)	2.000 (0.000)
Hpi	0.997 (0.047)	0.998 (0.019)	0.999 (0.018)	0.999 (0.004)	0.998 (0.002)	0.999 (0.002)
	2.006 (0.077)	2.006 (0.077)	2.006 (0.077)	2.182 (0.449)	2.378 (0.597)	2.706 (0.810)
Hscv	0.999 (0.011)	1.000 (0.001)	1.000 (0.000)	1.000 (0.001)	1.000 (0.001)	1.000 (0.001)
	2.002 (0.045)	2.000 (0.000)	2.000 (0.000)	2.006 (0.077)	2.028 (0.165)	2.110 (0.320)

Tabella 3.2: Risultati per gruppi sferici e separati, con d=3. Cfr. Tabella 3.1.

	Pdf Cluster			*Ms Cluster*		
	n=250	n=500	n=1000	n=250	n=500	n=1000
Hns	1.000 (0.000)	1.000 (0.000)	1.000 (0.000)	1.000 (0.000)	1.000 (0.000)	1.000 (0.000)
	2.000 (0.000)	2.000 (0.000)	2.000 (0.000)	2.000 (0.000)	2.000 (0.000)	2.000 (0.000)
Hlscv	1.000 (0.000)	1.000 (0.000)	1.000 (0.000)	0.999 (0.002)	1.000 (0.001)	1.000 (0.001)
	2.000 (0.000)	2.000 (0.000)	2.000 (0.000)	2.092 (0.303)	2.122 (0.340)	2.192 (0.433)
Hpi	1.000 (0.000)	1.000 (0.000)	1.000 (0.000)	0.985 (0.011)	0.985 (0.008)	0.984 (0.006)
	2.000 (0.000)	2.000 (0.000)	2.000 (0.000)	3.892 (1.427)	5.838 (1.967)	10.168 (2.864)
Hscv	1.000 (0.000)	1.000 (0.000)	1.000 (0.000)	1.000 (0.001)	1.000 (0.001)	0.999 (0.001)
	2.000 (0.000)	2.000 (0.000)	2.000 (0.000)	2.018 (0.133)	2.064 (0.245)	2.350 (0.562)

Tabella 3.3: Risultati per gruppi sferici e separati, con d=6. Cfr. Tabella 3.1.

	Pdf Cluster			*Ms Cluster*		
	n=250	n=500	n=1000	n=250	n=500	n=1000
Hns	0.741 (0.129)	0.763 (0.062)	0.775 (0.035)	0.729 (0.156)	0.765 (0.065)	0.779 (0.030)
	2.008 (0.228)	2.014 (0.134)	2.012 (0.109)	2.140 (1.512)	2.142 (1.500)	2.078 (0.555)
Hlscv	0.664 (0.167)	0.685 (0.140)	0.697 (0.142)	0.598 (0.300)	0.668 (0.257)	0.714 (0.216)
	2.502 (1.024)	2.536 (1.230)	2.552 (1.266)	2.644 (2.953)	2.410 (3.441)	2.128 (2.501)
Hpi	0.721 (0.112)	0.737 (0.084)	0.750 (0.069)	0.731 (0.122)	0.761 (0.063)	0.771 (0.045)
	2.172 (0.418)	2.144 (0.363)	2.136 (0.360)	2.436 (1.628)	2.370 (0.903)	2.758 (4.179)
Hscv	0.731 (0.123)	0.750 (0.076)	0.761 (0.057)	0.709 (0.195)	0.762 (0.073)	0.779 (0.030)
	2.064 (0.303)	2.072 (0.266)	2.078 (0.283)	2.006 (0.756)	2.244 (2.780)	2.096 (0.378)

Tabella 3.4: Risultati per gruppi sferici e parzialmente sovrapposti, con d=2. Cfr. Tabella 3.1.

	Pdf Cluster			*Ms Cluster*		
	n=250	n=500	n=1000	n=250	n=500	n=1000
Hns	0.877 (0.045)	0.886 (0.032)	0.890 (0.021)	0.887 (0.040)	0.892 (0.028)	0.894 (0.020)
	2.004 (0.063)	2.002 (0.045)	2.000 (0.000)	2.118 (0.341)	2.164 (0.402)	2.198 (0.423)
Hlscv	0.820 (0.118)	0.830 (0.115)	0.849 (0.093)	0.748 (0.315)	0.808 (0.257)	0.860 (0.170)
	2.248 (0.586)	2.256 (0.599)	2.188 (0.487)	2.964 (6.570)	2.722 (3.901)	2.376 (1.016)
Hpi	0.855 (0.073)	0.861 (0.074)	0.870 (0.062)	0.861 (0.061)	0.873 (0.049)	0.884 (0.033)
	2.082 (0.275)	2.104 (0.318)	2.078 (0.276)	3.382 (1.231)	3.892 (1.862)	4.458 (1.801)
Hscv	0.876 (0.046)	0.884 (0.033)	0.888 (0.024)	0.886 (0.041)	0.891 (0.029)	0.894 (0.020)
	2.006 (0.077)	2.002 (0.045)	2.002 (0.045)	2.078 (0.276)	2.184 (0.422)	2.318 (0.534)

Tabella 3.5: Risultati per gruppi sferici e parzialmente sovrapposti, con d=3. Cfr. Tabella 3.1.

	Pdf Cluster			*Ms Cluster*		
	n=250	n=500	n=1000	n=250	n=500	n=1000
Hns	0.969 (0.100)	0.982 (0.014)	0.984 (0.008)	0.959 (0.029)	0.972 (0.012)	0.977 (0.008)
	1.990 (0.100)	2.000 (0.000)	2.000 (0.000)	5.174 (1.769)	5.864 (1.916)	7.210 (2.178)
Hlscv	0.967 (0.101)	0.981 (0.020)	0.984 (0.014)	0.854 (0.071)	0.912 (0.042)	0.940 (0.018)
	1.992 (0.109)	2.002 (0.045)	2.002 (0.045)	16.016 (3.357)	19.344 (3.761)	25.170 (4.864)
Hpi	0.961 (0.121)	0.977 (0.039)	0.982 (0.023)	0.506 (0.114)	0.670 (0.104)	0.785 (0.081)
	1.986 (0.118)	2.014 (0.118)	2.008 (0.089)	47.732 (6.919)	62.262 (8.134)	86.350 (10.589)
Hscv	0.968 (0.101)	0.982 (0.014)	0.984 (0.008)	0.977 (0.016)	0.977 (0.012)	0.976 (0.008)
	1.990 (0.100)	2.000 (0.000)	2.000 (0.000)	3.238 (1.133)	4.526 (1.591)	7.364 (2.295)

Tabella 3.6: Risultati per gruppi sferici e parzialmente sovrapposti, con d=6. Cfr. Tabella 3.1.

	Pdf Cluster			*Ms Cluster*		
	n=250	n=500	n=1000	n=250	n=500	n=1000
Hns	0.998 (0.023)	0.996 (0.033)	0.985 (0.058)	1.000 (0.000)	1.000 (0.000)	1.000 (0.000)
	2.008 (0.089)	2.018 (0.133)	2.060 (0.238)	2.000 (0.000)	2.000 (0.000)	2.000 (0.000)
Hlscv	0.309 (0.120)	0.202 (0.069)	0.125 (0.035)	1.000 (0.000)	1.000 (0.000)	1.000 (0.000)
	8.326 (2.693)	12.474 (3.924)	19.382 (5.096)	2.000 (0.000)	2.000 (0.000)	2.000 (0.000)
Hpi	0.549 (0.157)	0.423 (0.104)	0.324 (0.082)	0.805 (0.171)	0.691 (0.175)	0.558 (0.148)
	4.400 (1.011)	5.564 (1.072)	7.206 (1.396)	3.148 (1.479)	4.290 (2.750)	6.352 (4.534)
Hscv	0.613 (0.167)	0.474 (0.123)	0.357 (0.093)	0.982 (0.064)	0.947 (0.106)	0.834 (0.166)
	3.904 (0.919)	4.996 (1.040)	6.556 (1.292)	2.112 (0.580)	2.454 (1.485)	3.922 (3.999)

Tabella 3.7: Risultati per gruppi concavi, con d=2. Cfr. Tabella 3.1.

	Pdf Cluster			*Ms Cluster*		
	n=250	n=500	n=1000	n=250	n=500	n=1000
Hns	0.774 (0.176)	0.726 (0.178)	0.656 (0.163)	0.805 (0.172)	0.775 (0.176)	0.739 (0.162)
	2.940 (0.733)	3.160 (0.779)	3.504 (0.777)	3.016 (1.203)	3.362 (1.515)	4.016 (2.153)
Hlscv	0.413 (0.157)	0.255 (0.083)	0.149 (0.041)	0.954 (0.205)	0.964 (0.186)	0.960 (0.196)
	6.428 (1.912)	9.918 (2.399)	16.340 (3.405)	2.008 (0.966)	1.966 (0.192)	1.960 (0.196)
Hpi	0.516 (0.148)	0.370 (0.112)	0.266 (0.074)	0.527 (0.139)	0.453 (0.126)	0.366 (0.099)
	4.752 (1.225)	6.648 (1.530)	9.096 (1.796)	5.310 (1.762)	7.142 (2.955)	10.488 (5.129)
Hscv	0.790 (0.183)	0.656 (0.184)	0.471 (0.151)	0.942 (0.120)	0.883 (0.153)	0.779 (0.180)
	2.924 (0.846)	3.712 (1.060)	5.330 (1.371)	2.294 (0.837)	2.726 (1.318)	4.206 (3.440)

Tabella 3.8: Risultati per gruppi concavi, con d=3. Cfr. Tabella 3.1.

	Pdf Cluster			*Ms Cluster*		
	n=250	n=500	n=1000	n=250	n=500	n=1000
Hns	0.435 (0.421)	0.571 (0.364)	0.627 (0.262)	0.223 (0.076)	0.236 (0.078)	0.232 (0.076)
	1.726 (0.672)	2.302 (0.899)	3.174 (1.100)	21.336 (3.434)	19.110 (3.924)	20.330 (5.758)
Hlscv	0.343 (0.422)	0.472 (0.386)	0.598 (0.285)	0.731 (0.401)	0.691 (0.431)	0.514 (0.489)
	1.622 (0.645)	2.178 (0.885)	3.212 (1.137)	8.080 (22.763)	9.038 (44.217)	11.432 (76.727)
Hpi	0.450 (0.392)	0.488 (0.293)	0.458 (0.186)	0.033 (0.008)	0.038 (0.008)	0.042 (0.008)
	2.058 (0.756)	2.956 (1.022)	4.818 (1.465)	113.674 (7.188)	131.444 (8.766)	129.664 (9.321)
Hscv	0.279 (0.425)	0.523 (0.472)	0.774 (0.351)	0.899 (0.138)	0.879 (0.150)	0.856 (0.164)
	1.334 (0.493)	1.624 (0.572)	2.076 (0.629)	2.644 (1.066)	2.960 (1.602)	3.604 (2.593)

Tabella 3.9: Risultati per gruppi concavi, con d=6. Cfr. Tabella 3.1.

	Pdf Cluster			*Ms Cluster*		
	n=250	n=500	n=1000	n=250	n=500	n=1000
Hns	0.200 (0.363)	0.363 (0.399)	0.664 (0.284)	0.024 (0.137)	0.070 (0.241)	0.209 (0.384)
	1.450 (0.604)	1.878 (0.748)	2.698 (0.787)	5.916 (3.983)	8.378 (5.358)	12.768 (14.347)
Hlscv	0.375 (0.135)	0.261 (0.087)	0.198 (0.059)	0.002 (0.038)	0.002 (0.042)	0.001 (0.022)
	8.724 (2.348)	15.302 (3.470)	27.156 (4.716)	3.784 (1.236)	5.144 (1.509)	7.200 (2.967)
Hpi	0.832 (0.136)	0.716 (0.139)	0.593 (0.108)	0.808 (0.082)	0.817 (0.044)	0.818 (0.039)
	3.150 (0.972)	4.686 (1.341)	7.738 (1.910)	14.506 (2.443)	22.990 (3.248)	36.390 (4.186)
Hscv	0.841 (0.142)	0.731 (0.139)	0.594 (0.107)	0.593 (0.413)	0.885 (0.065)	0.889 (0.026)
	3.010 (0.925)	4.496 (1.312)	7.704 (1.880)	9.892 (3.964)	15.580 (2.713)	24.942 (3.363)

Tabella 3.10: Risultati per gruppi asimmetrici, con d=2. Cfr. Tabella 3.1.

	Pdf Cluster			*Ms Cluster*		
	n=250	n=500	n=1000	n=250	n=500	n=1000
Hns	0.680 (0.392)	0.787 (0.255)	0.832 (0.161)	0.611 (0.428)	0.834 (0.279)	0.940 (0.044)
	1.856 (0.404)	2.070 (0.305)	2.222 (0.470)	9.402 (2.006)	13.558 (2.599)	19.412 (2.941)
Hlscv	0.873 (0.199)	0.791 (0.183)	0.588 (0.258)	0.206 (0.385)	0.366 (0.459)	0.721 (0.409)
	2.298 (0.747)	3.268 (1.829)	7.838 (4.998)	7.222 (2.923)	10.382 (7.027)	15.656 (14.211)
Hpi	0.877 (0.145)	0.806 (0.132)	0.683 (0.136)	0.731 (0.051)	0.738 (0.042)	0.743 (0.036)
	2.402 (0.591)	3.282 (0.998)	5.242 (1.480)	24.948 (3.385)	41.748 (4.699)	71.958 (6.467)
Hscv	0.909 (0.150)	0.862 (0.123)	0.779 (0.132)	0.843 (0.064)	0.845 (0.029)	0.841 (0.026)
	2.126 (0.338)	2.612 (0.750)	3.992 (1.236)	15.338 (2.458)	26.316 (3.637)	45.944 (4.791)

Tabella 3.11: Risultati per gruppi asimmetrici, con d=3. Cfr. Tabella 3.1.

	Pdf Cluster			*Ms Cluster*		
	n=250	n=500	n=1000	n=250	n=500	n=1000
Hns	0.303 (0.417)	0.576 (0.338)	0.651 (0.245)	0.819 (0.030)	0.849 (0.023)	0.876 (0.016)
	1.370 (0.483)	1.852 (0.355)	1.996 (0.063)	22.378 (3.198)	34.004 (4.181)	52.868 (5.615)
Hlscv	0.327 (0.438)	0.611 (0.357)	0.662 (0.239)	0.814 (0.032)	0.848 (0.024)	0.876 (0.017)
	1.376 (0.485)	1.842 (0.365)	1.998 (0.045)	22.884 (3.478)	34.200 (4.568)	52.452 (5.932)
Hpi	0.266 (0.432)	0.469 (0.412)	0.624 (0.304)	0.515 (0.045)	0.531 (0.035)	0.546 (0.028)
	1.280 (0.449)	1.630 (0.492)	1.956 (0.338)	61.694 (5.934)	113.380 (8.612)	210.794 (12.444)
Hscv	0.317 (0.450)	0.574 (0.389)	0.642 (0.273)	0.737 (0.037)	0.733 (0.030)	0.732 (0.024)
	1.342 (0.475)	1.770 (0.421)	1.986 (0.195)	31.996 (4.154)	60.562 (6.215)	115.734 (9.165)

Tabella 3.12: Risultati per gruppi asimmetrici, con d=6. Cfr. Tabella 3.1.

3.2 Applicazione a dati reali

Nel seguente paragrafo vengono considerati gli stessi algoritmi, *Ms Cluster* e *Pdf Cluster*, in un contesto reale, in un'applicazione a dati climatici. Il *dataset* analizzato è stato costruito a partire dai dati resi disponibili *online* dalla Banca Dati Agrometeorologica Nazionale [2] *(BDAN)* del Ministero delle Politiche Agricole e Forestali *(MiPAF)*. Sono state considerate 6 variabili, misurate nelle stazioni meteorologiche di 62 comuni sparsi sul territorio italiano. L'intento è quello di esplorare come operano i due metodi di *clustering* in relazione ai diversi vettori di lisciamento, cercando di valutare la sensatezza dei raggruppamenti forniti.

Per ciascuna variabile è stata calcolata una media dei valori giornalieri registrati durante il mese di giugno 2015. Esse sono le seguenti:

1. Velocità e direzione del vento. Si fa riferimento al vento filato, che indica la distanza percorsa dal vento in un determinato periodo di tempo e viene espresso in chilometri. La rilevazione avviene a 10 metri dal suolo.

2. Temperatura minima dell'aria a 2 metri dal suolo; espressa in gradi *Celsius*.

3. Temperatura massima dell'aria a 2 metri dal suolo; espressa in gradi *Celsius*.

4. Umidità relativa diurna dell'aria, rilevata a 2 metri dal suolo ed espressa in percentuale. Indica il rapporto percentuale tra la quantità di vapore contenuta in una massa d'aria e la quantità massima che lo stesso volume d'aria può contenere, a parità di temperatura e pressione.

5. Precipitazione; espressa in millimetri.

6. Pressione atmosferica media. Per convenzione, si definisce come il peso dell'aria che grava su una superficie di 1 m^2. La sua unità di misura è l'*ettoPascal (hPa)*.

Prima di procedere con la descrizione dei risultati, per la quale si rimanda al paragrafo successivo, si spiega brevemente la composizione climatica dell'Italia (Figura 3.2), con la finalità di interpretare meglio le partizioni individuate.

La penisola italiana, situata a Sud dell'Europa, si caratterizza per una notevole estensione in latitudine, da $47°N$ a $36°N$. Possiede una complessa orografia, essendo attraversata da Alpi e Appennini, e risente dell'azione mitigatrice del Mar Mediterraneo. Questi fattori, insieme a molti altri, determinano le variazioni climatiche esistenti da una zona all'altra. In generale, l'Italia è compresa nella fascia

[2] *http://cma.entecra.it/Banca_dati_agrometeo/index3.htm*

temperata con clima mediterraneo, ma risulta divisa in almeno sei macro aree che si caratterizzano per aspetti climatici diversi. Il passaggio dall'una all'altra è graduale e anche in una stessa zona i valori termici e pluviometrici assumono una certa variabilità. Con riferimento alla Figura 3.2 si descrivono ora le sei aree principali. La regione Alpina (colorata in grigio) comprende Valle d'Aosta, Tentino Alto Adige, ed i settori montuosi di Piemonte, Veneto, Lombardia e Friuli Venezia Giulia. Il clima alpino è molto condizionato dall'altitudine e si riconduce al tipo temperato freddo. Alcune caratteristiche di questo clima sono i forti sbalzi di temperatura tra giorno e notte, l'elevata frequenza di giorni di gelo e l'abbondanza di precipitazioni, che arrivano a un massimo in estate e ad un minimo in inverno, periodo in cui assumono anche carattere nevoso. La Pianura Padana e i settori pianeggianti dell'Alto Adriatico (dal Piemonte fino alle alte Marche, in verde) si inseriscono in una fascia climatica temperata continentale, dove una ridotta influenza del mare accentua le escursioni termiche sia giornaliere che stagionali. Le precipitazioni non sono molto abbondanti e hanno una frequenza maggiore nelle stagioni intermedie. Abruzzo, Molise e Marche centro meridionali confluiscono in un'area indicata come *Medio Adriatico* (in rosa), che presenta un comportamento termico e pluviometrico con caratteri mediterranei, ma con elementi di continentalità dettati dall'influenza mitigatrice limitata del Mar Adriatico e dall'esposizione favorevole alle correnti da Nord e da Est. C'è poi il versante Ligure Tirrenico, dalla Toscana alla Campania (colorato in giallo). Queste regioni risentono dell'azione mitigante del Tirreno. A parità di latitudine, il clima è molto più mite e umido rispetto al settore adriatico, sia per la presenza degli Appennini che riducono gli effetti delle correnti artiche da Nord e da Est, sia perchè il Mar Tirreno è un bacino profondo, le cui acque superficiali sono sufficientemente calde da mitigare in modo sensibile il clima, rendendolo quasi mediterraneo. Le estati sono calde e secche; i periodi di freddo intenso limitati. La regione Appenninica (in azzurro) comprende i settori montuosi del Centro Italia. Il clima risulta influenzato dalla quota, con temperature in diminuzione al suo aumentare. Gli inverni sono freddi e nevosi, mentre le estati calde nelle conche, progressivamente più fresche in quota. Le precipitazioni risultano molto abbondanti, ben distribuite durante l'anno con massimi in autunno e in primavera. Dalla Campania verso Sud si estende infine la regione Mediterranea (color arancio), dove il clima risulta fortemente condizionato dal mare. Le estati sono secche con temperature molto elevate durante le espansioni dell'anticiclone Africano, mentre gli inverni sono molto piovosi. Per quanto riguarda la zona interna delle isole, il clima registrato è di tipo continentale appenninico.

Figura 3.2: Zone climatiche in Italia.

3.2.1 Risultati

I grafici che illustrano i risultati ottenuti seguono la numerazione da 3.3 a 3.7.

Il vettore di lisciamento ottenuto con riferimento alla distribuzione normale *(Hns)* tende, rispetto agli altri, a sovra-lisciare la funzione di densità. Per questa ragione, nell'analisi condotta viene inserito anche in forma scalata, moltiplicato per un fattore pari a 0.75 secondo quanto suggerito da Azzalini e Torelli (2007). Con l'algoritmo *PdfCluster*, l'uso di *Hns* conduce all'identificazione di un unico gruppo; mentre, con la versione scalata, ne affiorano due (Figura 3.3(a)). I comuni appartenenti al gruppo minore (in rosso) si collocano in generale nelle coste dell'Italia meridionale. Sono comuni caratterizzati in media da una velocità del vento molto forte, una maggiore umidità relativa e uno sbalzo non significativo tra temperatura minima e massima. Questo partizionamento, seppur ragionevole, è poco informativo. Si ricercano le cause, quindi, andando ad esplorare i passi iniziali dell'algoritmo. I nuclei dei *cluster* individuati sono molto ridotti, composti da 2 e 3 comuni. Quelli neri hanno caratteristiche meteorologiche molto simili e sono relativamente vicini fra loro; i rossi sono distanti e hanno un'altitudine molto diversa, anche se si tratta di località costiere le cui condizioni climatiche risultano quasi identiche, salvo per le precipitazioni. Al passo successivo viene poi aggiunto un discreto numero di comuni ad entrambi i gruppi. Si può intuire che, procedendo, considerata la distanza tra i comuni rossi e la loro collocazione in zone piuttosto estreme e isolate, per tutti

quelli ancora da allocare nel Centro e Nord Italia sarà maggiore la densità calcolata in relazione al gruppo nero (al quale, infatti, saranno allocati tutti con un'unica eccezione).

Nelle Figure 3.3(b) e 3.4 viene illustrato il raggruppamento fornito dal metodo *Ms Cluster* prima rispetto ad *Hns*, poi con lo stesso vettore scalato per 0.75. Nel primo caso (Figura 3.3(b)) si ottiene un partizionamento molto simile a quello emerso con *Pdf Cluster* rispetto al vettore *Hns* scalato. Vengono tuttavia identificati 13 gruppi, tra i quali 11 sono formati ciascuno da uno o due comuni, mentre gli altri due gruppi, in azzurro e in blu, coincidono in larga misura con quelli identificati da *Pdf Cluster*.

Con *Hns* moltiplicato per 0.75 il numero di gruppi identificati è più che raddoppiato. Tuttavia, tra i 30 *cluster* emersi, soltanto 5 contano almeno 4 comuni, mentre tutti gli altri sono formati al più da 2 comuni. Per agevolare l'interpretazione di questo risultato, in Figura 3.4(b) vengono indicati soltanto i comuni dei 5 gruppi più numerosi insieme ai 4 che comprendono 2 comuni ciascuno, trascurando i gruppi di un'unità. Si può notare a questo punto qualche corrispondenza con le aree climatiche note in Italia illustrate nella Figura 3.2. I comuni rossi si trovano nelle vicinanze della regione alpina e quelli azzurri presso la zona appenninica. I posti segnati in verde sono tutti situati sulla costa, facendo così presumere che la vicinanza di una località al mare abbia comunque delle implicazioni sul clima che lo rendono simile. I comuni in blu e in nero si trovano invece per la prevalenza nell'entroterra, in zone pianeggianti o leggermente montuose. Anche i *cluster* formati da due comuni mostrano condizioni climatiche molto simili. Dunque, considerati i limiti apportati dalla struttura del *dataset* studiato in generale alle tecniche di *clustering* non parametrico, la performance di quest'ultimo metodo con vettore di lisciamento *Hns* scalato si rivela, nel complesso, soddisfacente.

Si prende ora in analisi il criterio *plug-in* per la selezione del parametro di lisciamento, con riferimento alle Figure 3.5 e 3.6(a). Per quanto riguarda il metodo *Pdf Cluster* i due gruppi risultanti si dividono quasi equamente il numero totale dei comuni. Esplorandone la composizione si scopre che, in media, i due gruppi si distinguono soprattutto per la velocità del vento, coerentemente con il fatto che, dove è superiore, i comuni presentano anche una maggiore altitudine. Di norma, infatti, la velocità del vento cresce con la quota. L'interpretazione, tuttavia, è resa complicata dal fatto che comuni segnati in rosso si trovino sia nella regione mediterranea che in parte di quella appenninica, mentre quelli neri sono sparsi un po' a Nord e lungo le coste del Centro Italia. Provando a tornare ai primi passi dell'algoritmo emerge un comportamento molto simile a quello che già si era delineato con *Hns* scalato, tanto che i nuclei dei *cluster* quasi coincidono, ma, procedendo con l'allocazione dei

comuni, con *Hpi* ne vengono inclusi un numero consistente anche nel gruppo rosso.

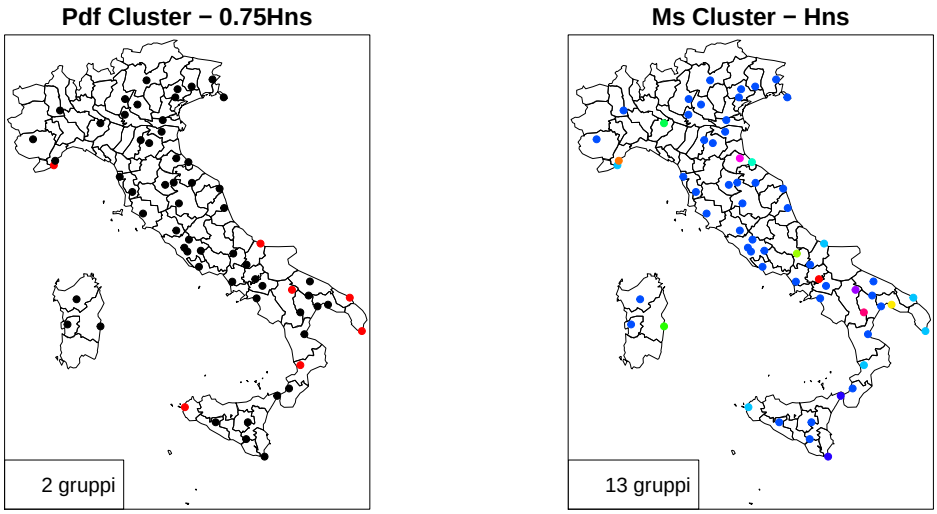
Per cogliere in modo più chiaro il comportamento di *Ms Cluster*, in relazione allo stesso vettore di lisciamento, si riporta come prima un grafico aggiuntivo (Figura 3.6(a)) per rappresentare i gruppi di maggiore numerosità, perchè, in analogia al caso con *Hns* scalato, si considerano i comuni che singolarmente creano un gruppo degli *outliers*. Confrontando le Figure 3.4(b) e 3.6(a) è immediato notare che i risultati sono quasi perfettamente sovrapponibili.

Restano da valutare i vettori *Hscv* e *Hlscv*. Il metodo *Pdf Cluster*, con entrambi, trova soltanto un gruppo, mentre l'algoritmo *Ms Cluster* si comporta in modo diverso. Scegliendo *Hscv* (Figura 3.6(b)) emergono 10 gruppi: uno molto ampio composto dal Centro e dall'Italia Nord-orientale (in azzurro), un altro più ridotto costituito solo da località costiere (in verde) e i restanti 8 gruppi formati ciascuno da un solo comune. Da questo punto di vista il partizionamento è molto vicino a quello ottenuto con *Pdf Cluster* e *Hns* scalato, fatta eccezione per gli 8 gruppi singoli. La stessa analogia emerge con *Hlscv* (Figura 3.7), tra i 17 gruppi identificati, ne emergono tre con numerosità maggiore di 1. È assente una distinzione tra regioni climatiche nel Centro e nel Nord Italia, ma viene correttamente individuata con i comuni in rosso e verde la regione Mediterranea: i comuni rossi per le aree sulla costa e quelli in verde per l'entroterra. In media il gruppo rosso si differenzia dagli altri per un'escursione termica giornaliera inferiore, in linea con l'effetto mitigante provocato dal mare, una velocità del vento e una percentuale di umidità relativa superiori.

In conclusione, il metodo *Ms Cluster* si è dimostrato più efficace di *Pdf Cluster*, individuando, per ogni scelta del vettore di lisciamento ad eccezione di *Hscv*, raggruppamenti interpretabili sulla base delle informazioni già disponibili per le regioni climatiche in Italia. È opportuno sottolineare, in ogni caso, che al di là dell'individuazione di alcuni gruppi di dimensione molto limitata, entrambi i metodi identificano, per diverse scelte di h, una partizione pressoché stabile, con due gruppi che inglobano al loro interno quelli noti della classificazione italiana in aree climatiche.

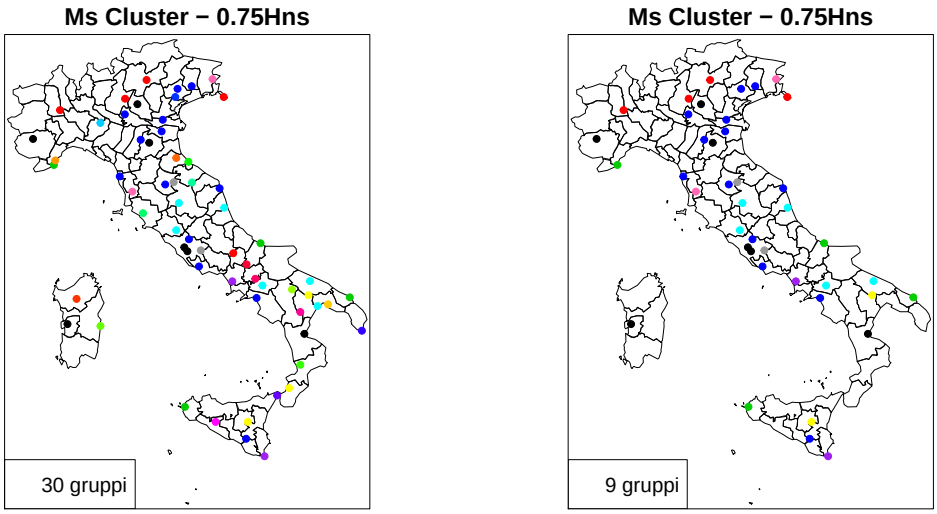
Questo risultato è presumibilmente imputabile a due motivazioni: da un lato il fatto di aver preso in esame i soli dati climatici relativi al mese di giugno, mentre le aree climatiche di riferimento, delineate in Figura 3.2, forniscono un'informazione globale che tiene conto della variabilità stagionale; dall'altro lato, per tenere sotto controllo la complessità computazionale, si è scelto di limitare il numero di variabili a 6 (trascurando quindi altre informazioni potenzialmente utili, quali, ad esempio, eliofania, radiazione solare e temperatura del terreno) e non si è tenuto nota della posizione geografica dei comuni di interesse.

Da un punto di vista strettamente statistico, invece, i metodi di *clustering* non parametrico sono inadeguati rispetto a dimensioni particolarmente elevate dello spazio campionario. Inoltre, al fine di distinguere correttamente i gruppi, è necessario riscontrare delle differenze rimarcabili in termini di densità; mentre è possibile che questo non si verifichi nel caso esaminato, trattandosi di un'area geografica piuttosto ristretta.



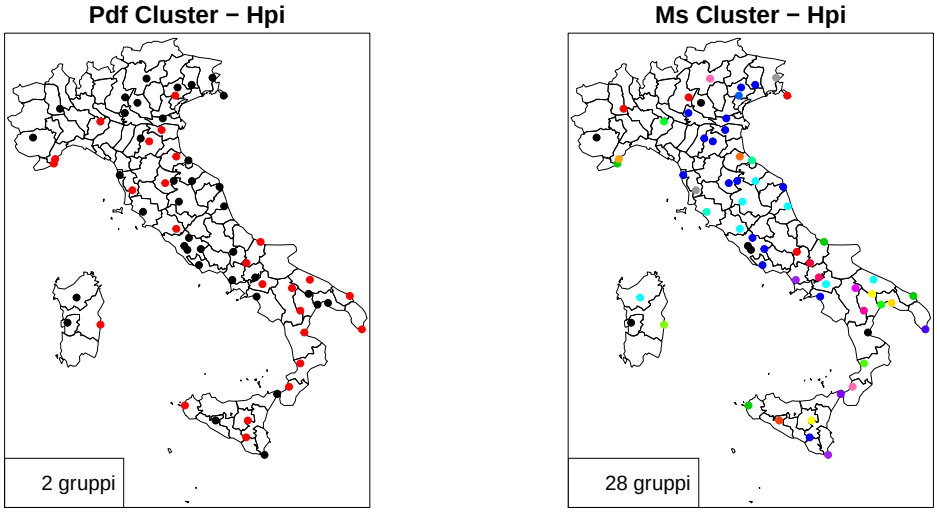
(a) (b)

Figura 3.3: A sinistra: algoritmo *Pdf Cluster* con vettore di lisciamento *Hns* moltiplicato per 0.75; a destra: algoritmo *Ms Cluster* con vettore di lisciamento *Hns*.



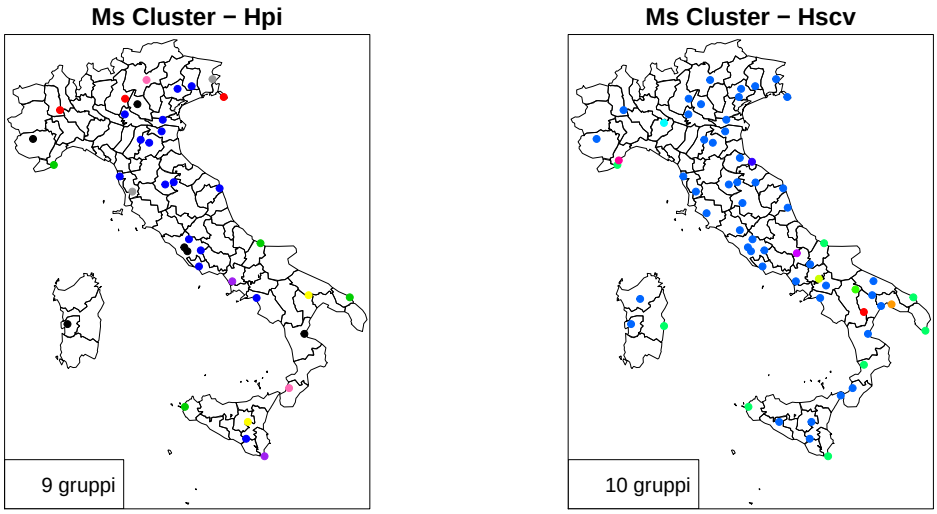
(a) (b)

Figura 3.4: A sinistra: algoritmo *Ms Cluster* con vettore di lisciamento *Hns* moltiplicato per 0.75; a destra lo stesso con esclusione dei gruppi con un solo comune.



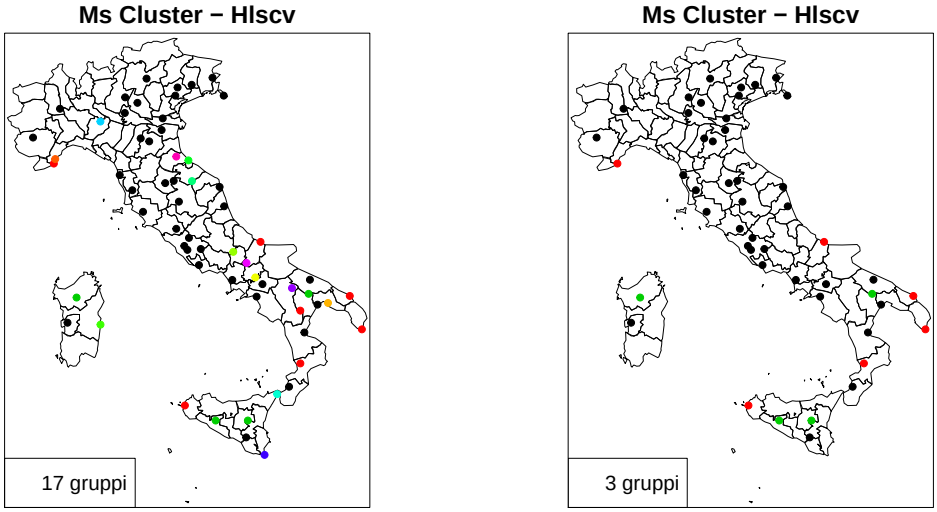
(a) (b)

Figura 3.5: A sinistra: algoritmo *Pdf Cluster* con vettore di lisciamento *Hpi*; a destra: algoritmo *Ms Cluster* con lo stesso vettore di lisciamento.



(a) (b)

Figura 3.6: A sinistra: algoritmo *Ms Cluster* con vettore di lisciamento *Hpi* e gruppi di almeno 2 comuni. A destra: algoritmo *Ms Cluster* con vettore di lisciamento *Hscv*.



(a) (b)

Figura 3.7: A sinistra: algoritmo *Ms Cluster* con vettore di lisciamento *Hlscv*; a destra lo stesso con esclusione dei gruppi con un solo comune.

Conclusione

L'obiettivo che si è perseguito in questa tesi è stato quello di approfondire lo studio della formulazione non parametrica del problema di raggruppamento, nelle sue due varianti, e capire in che misura la sua efficacia fosse vincolata alla scelta del parametro di lisciamento usato per la stima della densità. Si è quindi cercata una risposta attraverso l'analisi dei risultati provenienti sia da uno studio di simulazione che da un'applicazione a dati climatici.

Attraverso lo studio di simulazione sono emersi risultati più soddisfacenti che dall'applicazione ai dati reali, da una parte per una possibile scelta troppo poco realistica, o troppo semplicistica, degli scenari di simulazione, dall'altra a causa dell'informazione ridotta contenuta nei dati reali considerati. Questa parziale inadeguatezza dei risultati è inoltre imputabile a dei limiti strutturali sia del *dataset* studiato, caratterizzato da una quantità di variabili elevata in relazione al numero di osservazioni, sia dei metodi di *clustering* modale che, in generale, basandosi su una stima non parametrica della densità, sono inappropriati rispetto a dimensioni elevate dello spazio campionario. Si ricorda, in aggiunta, il fenomeno della *maledizione della dimensionalità*, che rende più complicato per i metodi di *clustering* identificare i gruppi al crescere della dimensionalità.

La forma dei gruppi incide sulle tecniche di raggruppamento: alcune difficoltà, che non si rilevano per l'individuazione di gruppi sferici (specialmente se ben separati), emergono invece per gruppi di forma più irregolare. Tuttavia, si deve tenere presente che i risultati sono globalmente soddisfacenti, soprattutto se relazionati al fatto che, in generale, altri metodi di raggruppamento, sia basati sulla distanza, sia parametrici, tendono a favorire l'individuazione di gruppi di forma sferica.

Un comportamento interessante si è osservato per l'allocazione degli *outliers*. Da entrambe le analisi condotte, infatti, si conferma una diversa sensibilità da parte dei due metodi, a prescindere dal vettore di lisciamento selezionato. Quello di Azzalini e Torelli, in generale, non li riconosce, tendendo comunque ad associarli ai *cluster* di numerosità maggiore; l'algoritmo *mean-shift*, invece, crea in corrispondenza degli *outliers* gruppi molto ridotti. Probabilmente, per avvicinare i risultati, servirebbe un

parametro di lisciamento più piccolo per il primo metodo, più grande per il secondo.

Complessivamente, l'approccio basato sulla ricerca delle mode ha rivelato una *performance* migliore rispetto a quello basato sulle curve di livello, definendo raggruppamenti generalmente più coerenti e interpretabili, anche se il metodo di Azzalini e Torelli risulta più robusto in presenza di *outliers*.

La scelta del parametro di lisciamento diventa determinante sulla *performance* dei metodi di *clustering* quanto più la forma dei gruppi è irregolare e distante da quella sferica (caso in cui, invece, non si nota una differenza significativa tra i vettori considerati). Anche quando la separazione fra i gruppi o la prominenza delle mode è limitata, il problema di definire l'ampiezza della finestra ottimale assume maggiore rilievo.

Non è possibile, a priori, stabilire un criterio per la selezione del grado di lisciamento che garantisca un buon adattamento rispetto ad ogni situazione; in generale, la scelta dipende dagli obiettivi dell'analisi di raggruppamento. Con il metodo della convalida incrociata lisciata si possono ottenere, in linea di massima, risultati più apprezzabili che con gli altri metodi. Il criterio *plug-in*, che non si è distinto per aver fornito raggruppamenti ottimali, ha dimostrato, in ogni caso, di essere una scelta soddisfacente. Il vettore di lisciamento asintoticamente ottimale con riferimento alla distribuzione normale, invece, tende a sovra-lisciare la stima della funzione di densità e, se i gruppi non sono ben separati, i metodi di *clustering* possono riscontrare più difficoltà nella loro identificazione. Per far fronte a questo limite e migliorare l'efficacia del metodo, Azzalini e Torelli propongono di scalare il vettore per un fattore pari a 0.75. Il metodo della convalida incrociata dei minimi quadrati, infine, ha rivelato di essere molto variabile, sovra-lisciando o sotto-lisciando spesso eccessivamente la stima della funzione di densità.

Volendo generalizzare ulteriormente queste considerazioni, sarebbe utile studiare anche altri scenari. Ad esempio, valutare numerosità campionarie più ampie e verificare quanto effettivamente diventi più complicato raccogliere risultati in dimensioni superiori. Si potrebbe tentare di rimediare all'incremento della dimensionalità sfruttando tecniche di riduzione, come, ad esempio, le componenti principali, e confrontare i risultati.

Sarebbe interessante, anche, considerare gradi di sovrapposizione tra i *cluster* più marcati di quello introdotto per i gruppi sferici, e replicare lo stesso con le altre forme. Si potrebbe estendere l'analisi anche a configurazioni diverse di *cluster*; ad esempio, in due dimensioni si può pensare di disporre le osservazioni in modo che formino una circonferenza e generalizzare poi un *cluster* così definito a più dimensioni.

Infine, si ricorda che i criteri presenti in letteratura per la scelta del parametro di

lisciamento sono molteplici, e in questa tesi si è fatto riferimento solo ai principali. Una possibilità per nuove analisi è quindi quella di stimare la densità adottando criteri di lisciamento alternativi a quelli considerati, ed eventualmente sviluppati *ad hoc* per il problema di individuazione delle regioni ad alta densità.

Appendice

Si riportano in seguito le funzioni *R* sviluppate per le simulazioni del Capitolo 3.

Codice 1: generazione di gruppi sferici.
I valori richiesti in input dalla funzione sono: n, la numerosità campionaria, p, un vettore di due elementi per le probabilità dei due gruppi, *mu1* e *sigma1*, *mu2* e *sigma2*, media e matrice di varianza rispettivamente del primo e secondo gruppo.

```
mist.norm<-function(n,p,mu1,mu2,sigma1,sigma2) {
d<-length(mu1)
prob<-sample((1:2),n,rep=T,prob=c(p,1-p))
a<-table(prob)[[1]]
b<-table(prob)[[2]]
mat1<-matrix(rmvnorm(a,mu1,sigma1),ncol=d)
mat2<-matrix(rmvnorm(b,mu2,sigma2),ncol=d)
list(X=rbind(mat1,mat2),gr=c(rep(1,a),rep(2,b))) }
```

Codice 2: generazione di gruppi asimmetrici.
I valori richiesti in input dalla funzione sono: n, la numerosità campionaria, d, la dimensione dello spazio, p, un vettore di due elementi per le probabilità dei due gruppi.

```
esp<-function(n,d,p) {
prob<-sample((1:2),n,rep=T,prob=c(p,1-p))
a<-table(prob)[[1]]
b<-table(prob)[[2]]
mat1<-matrix(exp(rmvnorm(a,rep(0,d),diag(1,d))),ncol=d)
mat2<-matrix(-exp(rmvnorm(b,rep(0,d),diag(1,d))),ncol=d)
list(X=rbind(mat1,mat2),gr=c(rep(1,a),rep(2,b))) }
```

Codice 3: generazione di gruppi concavi.
I valori richiesti in input dalla funzione sono: n, la numerosità campionaria, d, la dimensione dello spazio, p, un vettore di due elementi per le probabilità dei due

gruppi, *pos1* e *pos2*, le coordinate dell'angolo più esterno dei due gruppi. Ad esempio, per il caso specifico rappresentato nella Figura 3.1: *pos1=c(0,0)* e *pos2=c(4,4)*.

```
unif<-function(n,dim,p,pos1,pos2) {
prob<-sample((1:6),n,rep=T,p)
a<-table(prob)[[1]]
b<-table(prob)[[2]]
c<-table(prob)[[3]]
d<-table(prob)[[4]]
e<-table(prob)[[5]]
f<-table(prob)[[6]]
mat<-matrix(NA,n,d)
if (d==2) {
mat[1:a,]<-matrix(c(runif(a,0,1),runif(a,0,1)),ncol=2)
mat[(a+1):(a+b),]<-matrix(c(runif(b,1,2),runif(b,0,1)),ncol=2)
mat[(a+b+1):(a+b+c),]<-matrix(c(runif(c,0,1),runif(c,1,2)),ncol
    =2)
mat[(a+b+c+1):(a+b+c+d),]<-matrix(c(runif(d,-1,0),runif(d,-1,0))
    ,ncol=2)
mat[(a+b+c+d+1):(a+b+c+d+e),]<-matrix(c(runif(e,-2,-1),runif(e
    ,-1,0)),ncol=2)
mat[(a+b+c+d+e+1):n,]<-matrix(c(runif(f,-1,0),runif(f,-2,-1)),
    ncol=2) }
if (d==3) {
mat[1:a,]<-matrix(c(runif(a,0,1),runif(a,0,1),runif(a,0,1)),ncol
    =3)
mat[(a+1):(a+b),]<-matrix(c(runif(b,0,1),runif(b,1,2),runif(b
    ,0,1)),ncol=3)
mat[(a+b+1):(a+b+c),]<-matrix(c(runif(c,1,2),runif(c,0,1),runif(
    c,0,1)),ncol=3)
mat[(a+b+c+1):(a+b+c+d),]<-matrix(c(runif(d,-1,0),runif(d,-1,0),
    runif(d,0,1)),ncol=3)
mat[(a+b+c+d+1):(a+b+c+d+e),]<-matrix(c(runif(e,-1,0),runif(e
    ,-2,-1),runif(e,0,1)),ncol=3)
mat[(a+b+c+d+e+1):n,]<-matrix(c(runif(f,-2,-1),runif(f,-1,0),
    runif(f,0,1)),ncol=3) }
if (d==4) {
mat[1:a,]<-matrix(c(runif(a,0,1),runif(a,0,1),runif(a,0,1),runif
    (a,0,1)),ncol=4)
mat[(a+1):(a+b),]<-matrix(c(runif(b,0,1),runif(b,1,2),runif(b
    ,0,1),runif(b,0,1)),ncol=4)
mat[(a+b+1):(a+b+c),]<-matrix(c(runif(c,1,2),runif(c,0,1),runif(
    c,0,1),runif(c,0,1)),ncol=4)
mat[(a+b+c+1):(a+b+c+d),]<-matrix(c(runif(d,-1,0),runif(d,-1,0),
    runif(d,0,1),runif(d,0,1)),ncol=4)
```

```
mat[(a+b+c+d+1):(a+b+c+d+e),]<-matrix(c(runif(e,-1,0),runif(e
    ,-2,-1),runif(e,0,1),runif(e,0,1)),ncol=4)
mat[(a+b+c+d+e+1):n,]<-matrix(c(runif(f,-2,-1),runif(f,-1,0),
    runif(f,0,1),runif(f,0,1)),ncol=4) }
if (d==5) {
mat[1:a,]<-matrix(c(runif(a,0,1),runif(a,0,1),runif(a,0,1),runif
    (a,0,1),runif(a,0,1)),ncol=5)
mat[(a+1):(a+b),]<-matrix(c(runif(b,0,1),runif(b,1,2),runif(b
    ,0,1),runif(b,0,1),runif(b,0,1)),ncol=5)
mat[(a+b+1):(a+b+c),]<-matrix(c(runif(c,1,2),runif(c,0,1),runif(
    c,0,1),runif(c,0,1),runif(c,0,1)),ncol=5)
mat[(a+b+c+1):(a+b+c+d),]<-matrix(c(runif(d,-1,0),runif(d,-1,0),
    runif(d,0,1),runif(d,0,1),runif(d,0,1)),ncol=5)
mat[(a+b+c+d+1):(a+b+c+d+e),]<-matrix(c(runif(e,-1,0),runif(e
    ,-2,-1),runif(e,0,1),runif(e,0,1),runif(e,0,1)),ncol=5)
mat[(a+b+c+d+e+1):n,]<-matrix(c(runif(f,-2,-1),runif(f,-1,0),
    runif(f,0,1),runif(f,0,1),runif(f,0,1)),ncol=5) }
if (d==6) {
mat[1:a,]<-matrix(c(runif(a,0,1),runif(a,0,1),runif(a,0,1),runif
    (a,0,1),runif(a,0,1),runif(a,0,1)),ncol=6)
mat[(a+1):(a+b),]<-matrix(c(runif(b,0,1),runif(b,1,2),runif(b
    ,0,1),runif(b,0,1),runif(b,0,1),runif(b,0,1)),ncol=6)
mat[(a+b+1):(a+b+c),]<-matrix(c(runif(c,1,2),runif(c,0,1),runif(
    c,0,1),runif(c,0,1),runif(c,0,1),runif(c,0,1)),ncol=6)
mat[(a+b+c+1):(a+b+c+d),]<-matrix(c(runif(d,-1,0),runif(d,-1,0),
    runif(d,0,1),runif(d,0,1),runif(d,0,1),runif(d,0,1)),ncol=6)
mat[(a+b+c+d+1):(a+b+c+d+e),]<-matrix(c(runif(e,-1,0),runif(e
    ,-2,-1),runif(e,0,1),runif(e,0,1),runif(e,0,1),runif(e,0,1)),
    ncol=6)
mat[(a+b+c+d+e+1):n,]<-matrix(c(runif(f,-2,-1),runif(f,-1,0),
    runif(f,0,1),runif(f,0,1),runif(f,0,1),runif(f,0,1)),ncol=6)
    }

gr<-c(rep(1,(a+b+c)),rep(2,(d+e+f)))
X<-rbind(mat[1:(a+b+c),]+rep(1,(a+b+c))%*%t(pos1),mat[(a+b+c+1):
    n,]+rep(1,(d+e+f))%*%t(pos2))
list(X=X,gr=gr) }
```

Bibliografia

Azzalini A.; Menardi G. (2014). pdfcluster: Cluster analysis via nonparametric density estimation. r package version 1.0-2. *See https://cran.r-project.org/package=pdfCluster.*

Azzalini A.; Torelli N. (2007). Clustering via nonparametric density estimation. *Statistics and Computing*, **17**(1), 71–80.

Bowman A. W. (1984). An alternative method of cross-validation for the smoothing of density estimates. *Biometrika*, **71**(2), 353–360.

Carmichael J.; George J. A.; Julius R. (1968). Finding natural clusters. *Systematic Zoology*, pp. 144–150.

Chacón, José E e Duon T.; Wand M. (2009). Asymptotics for general multivariate kernel density derivative estimators.

Cormack R. M. (1971). A review of classification. *Journal of the Royal Statistical Society. Series A (General).*

Duong T. (2015). ks: Kernel smoothing. r package version 1.9.4. *See https://cran.r-project.org/package=ks.*

Einbeck J.; Evers L. (2013). Lpcm: Local principal curve methods. r package version 0.44-8. *See https://cran.r-project.org/package=LPCM.*

Fukunaga K.; Hostetler L. D. (1975). The estimation of the gradient of a density function, with applications in pattern recognition. *Information Theory, IEEE Transactions on*, **21**(1), 32–40.

Genz A.; Bretz F.; Miwa T.; Mi X.; Leisch F.; Scheipl F.; Bornkamp B.; Hothorn T. (2014). mvtnorm: Multivariate normal and t distributions. r package version 1.0-2. *See https://cran.r-project.org/package=mvtnorm.*

Hartigan J. A. (1975). *Clustering algorithms.* John Wiley & Sons, Inc.

Hubert L.; Arabie P. (1985). Comparing partitions. *Journal of classification*, **2**(1), 193–218.

Li J.; Ray S.; Lindsay B. G. (2007). A nonparametric statistical approach to clustering via mode identification. *Journal of Machine Learning Research*, **8**(8), 1687–1723.

McLachlan G.; Peel D. (2004). *Finite mixture models*. John Wiley & Sons.

Menardi G.; Azzalini A. (2014). An advancement in clustering via nonparametric density estimation. *Statistics and Computing*, **24**(5), 753–767.

Parzen E. (1962). On estimation of a probability density function and mode. *The annals of mathematical statistics*, pp. 1065–1076.

R Development Core Team (2013). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

Rand W. M. (1971). Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical association*, **66**(336), 846–850.

Rosenblatt M.; Al. (1956). Remarks on some nonparametric estimates of a density function. *The Annals of Mathematical Statistics*, **27**(3), 832–837.

Rudemo M. (1982). Empirical choice of histograms and kernel density estimators. *Scandinavian Journal of Statistics*.

Silverman B. W. (1986). *Density estimation for statistics and data analysis*, volume 26. CRC press.

Stuetzle W. (2003). Estimating the cluster tree of a density by analyzing the minimal spanning tree of a sample. *Journal of classification*, **20**(1), 025–047.

Stuetzle W.; Nugent R. (2012). A generalized single linkage method for estimating the cluster tree of a density. *Journal of Computational and Graphical Statistics*.

Wand M. P.; Jones M. C. (1994). *Kernel smoothing*. Crc Press.

Wasserman L. (2006). *All of nonparametric statistics*. Springer Science & Business Media.

Wishart D. (1969). Mode analysis: A generalization of nearest neighbor which reduces chaining effects. *Numerical taxonomy*, **76**(282-311), 17.

Wong M. A.; Lane T. (1981). A kth nearest neighbour clustering procedure. In *Computer Science and Statistics: Proceedings of the 13th Symposium on the Interface*, pp. 308–311. Springer.

Ringraziamenti

A Giovanna Menardi, per il tempo che mi ha dedicato.
Ad Ainhoa e Francesca, per ogni giorno iniziato insieme.
Alla mia famiglia, da cui non avrei potuto ricevere maggior sostegno e fiducia.