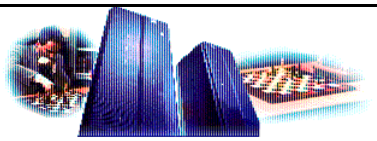
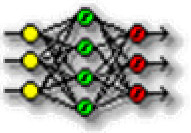
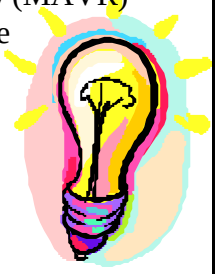




L'intelligenza biologica

Reti Neurali con funzione di attivazione a base radiale

Alberto Borghese
Università degli Studi di Milano
Laboratorio di Motion Analysis and Virtual Reality (MAVR)
Dipartimento di Scienze dell'Informazione
borghese@dsi.unimi.it



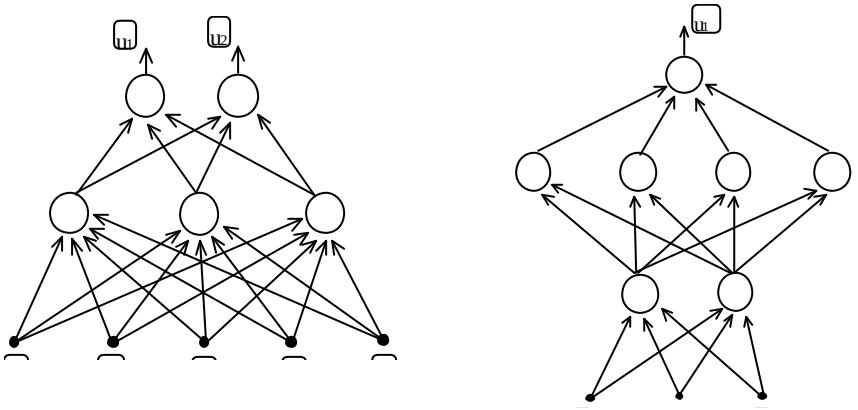
Sommario

.

RBF: reti neurali con neuroni a base radiale.

- Struttura della rete.
- Apprendimento ibrido.
- Teoria della regolarizzazione.
- Stima Bayesiana.
- Teoria del filtraggio.
- Approccio gerarchico.
- Approccio gerarchico locale.

Problema con MLP



Difficult to make it learn!

Approssimazione e RBF networks: ipotesi

Apprendimento da esempi è in molti casi equivalente ad approssimare una funzione multi-variabile.

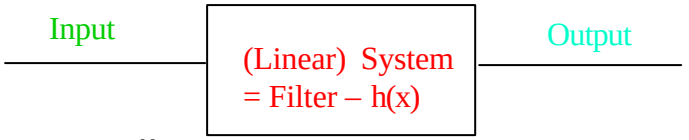
Dati $\{P(x_1, x_2, x_3 ....., x_M, x_{M+1})\}$ posso scrivere il mio set di esempi come: $\{P(\mathbf{x}, y(\mathbf{x}))\}$ con $y = x_{M+1}$ e $\mathbf{x} \in R^M$.

Ad esempio nello spazio 3D la mia funzione rappresenterà l'altezza della superficie $z = f(x,y)$.

E' una ricostruzione 2½ D.

Alcuni Richiami sulla Teoria del Filtraggio

Prodotto di convoluzione (visualizzazione grafica)



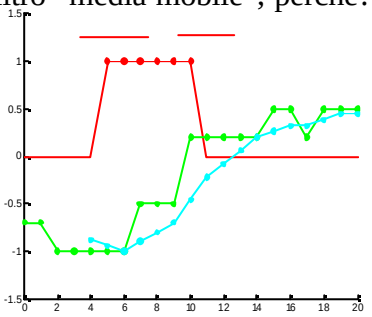
$$out(x_k) = 1/M \sum_{n=0}^{M} h(x_n) input(x_n - x_k) = h(x) * input(x)$$

Filtro "media mobile", perchè?

Il segnale "scorre" verso destra e sinistra rispetto al filtro al variare del valore di x_n".

In ogni posizione viene effettuata la somma dei prodotti dei campioni del segnale e del fiotro.

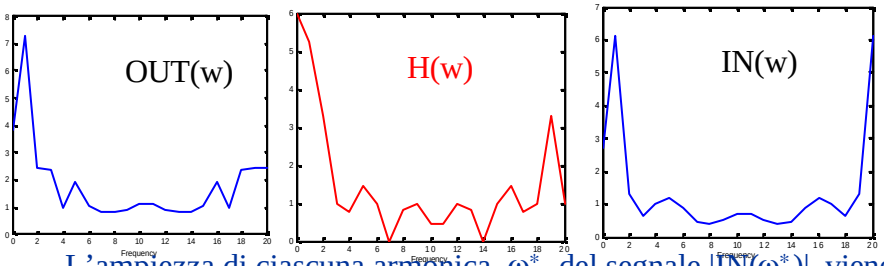
Il primo valore di uscita si ha per $x_n = 5$ pari all'ampiezza del filtro.



Rappresentazione nel dominio delle trasformate

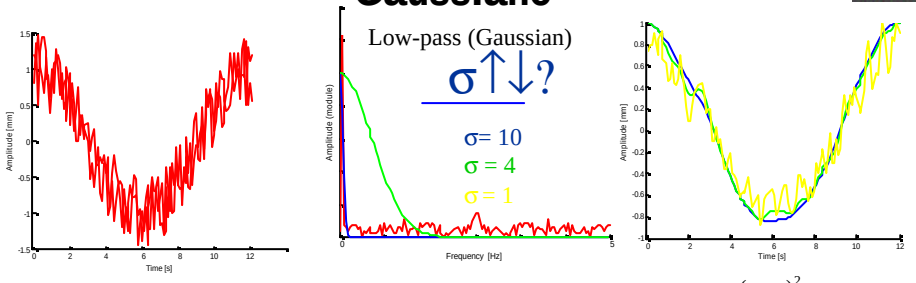
Input → (Linear) System = Filter – H(w) → Output

La convoluzione nel dominio dello spazio equivale ad un prodotto nel dominio delle frequenze $out(x) = \int h(\tilde{x}) input(x - \tilde{x}) d\tilde{x}$ $\quad OUT(w) = H(w) * IN(w)$



L'ampiezza di ciascuna armonica, ω^*, del segnale $|IN(\omega^*)|$, viene modificata selettivamente a seconda dell'ampiezza di $|H(\omega^*)|$.

Filtraggio reale passa-basso Gaussiano



Consideriamo il filtro Gaussiano: $h(x; x_c) = g(x - x_c; s) = e^{-\frac{(x-x_c)^2}{s^2}}$

La posizione della Gaussiana è definita da x_c, la sua ampiezza da σ.

La sua Trasformata di Fourier è ancora una Gaussiana: $H(w) = e^{-1/4 s^2 w^2} e^{-2\pi j v x_c}$

$|H(w)| = e^{-1/4 s^2 w^2}$ A parità di ω, l'ampiezza cresce con il decresce di σ. σ regola l'ampiezza della banda passante del filtro.

La Gaussiana attenua in modo progressivamente maggiore le armoniche con frequenze più elevate (filtraggio passa-basso)

se

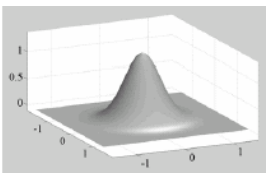
Struttura delle RBF networks

$$S(P) = \sum_{k=1}^{M} w_k G(P - P_k \mid s_k)$$

Funzioni quasi-locali

Ispirazione biologica. Campo recettivo di un neurone sensoriale (e.g. I neuroni della coclea sono sensibili ad una banda di frequenza, i neuroni della corteccia somatosensoriale primaria ad una regione del corpo, alcuni neuroni della corteccia visiva sono sensibili all'orientamento dei contorni...).

Ciascuna unità risponde ad input in una regione limitata, ovvero ciascun input attiva un certo numero di unità.



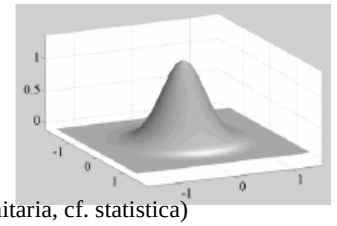
- Output della Gaussiana va velocemente a 0.
- Spline.
- In generale, funzioni a base radiale, in modulo decrescenti.

Funzioni di attivazione a simmetria radiale

Supponiamo che vale: $f(x) = f(Rx)$, con R matrice di rotazione. Allora, si dice che f(x) è invariante a rotazione, ovvero sia ha simmetria radiale.

Per le funzioni di attivazione RBF, questo vuol dire passare dalla generica formulazione:

$$\mathbf{z} = f(\mathbf{x}) = \sum_{k=1}^{M} \mathbf{w}_k e^{-(\mathbf{x}-\mathbf{x})\Sigma_k(\mathbf{x}-\mathbf{x})^T}$$



Alla formulazione a simmetria radiale:

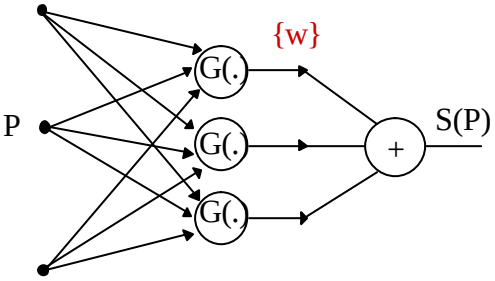
$$z = f(\mathbf{x}) = \frac{1}{\sqrt{ps}^M} e^{\frac{(\mathbf{x}-\mathbf{x}_k)^2}{s^2}}$$

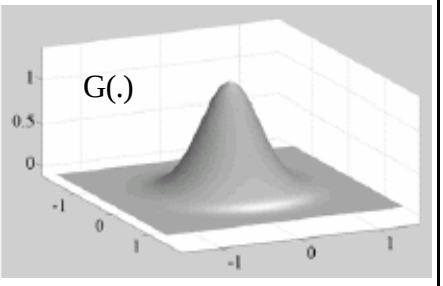
(ha norma L^1 unitaria, cf. statistica)

Questo riflette il fatto che non si hanno assunzioni a-priori sull'importanza delle variabili: tutte le direzioni hanno la stessa importanza.

Reti con elementi di attivazione radiali (RBF)

Perceptrone con unità quasi-locali. $S(P) = \sum_{k=1}^{M} w_k G(P - P_k \mid s_k)$

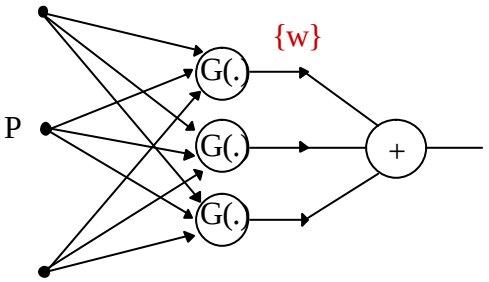




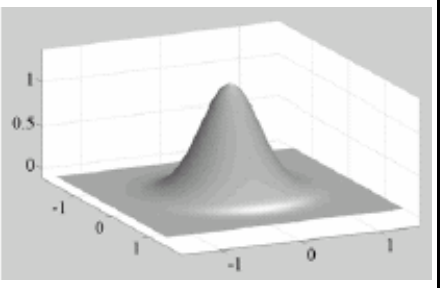
Distribuiamo le Gaussiane nell'insieme di definizione e calcoliamo la superficie come somma pesata.

$P, P_k \in R^m$ $S(P), w_k, \sigma_k$ (simmetria radiale) $\in R$

Parametri delle RBF



$$S(P \mid a) = \sum_{k=1}^{M} w_k G(P - P_k \mid s_k)$$



$\{P_k\}$, M, $\{\sigma_k\}$ – parametri strutturali.
$\{w_k\}$ – pesi sinaptici.

Per ogni Gaussiana si può definire un campo recettivo (regione di influenza). L'ampiezza è determinata da σ_k.

Al variare di α, varia la superficie ricostruita $S(P \mid \alpha)$.

Determinazione dei parametri (supervised learning)

Definisco una funzione costo J(.), funzione dei parametri α.

$$S(P \mid a) = \sum_{k=1}^{M} w_k G(P - P_k \mid s_k) \qquad \min_{\{a\}} J(a) = \min_{\{a\}} \left(S\left[(P \mid a) - y \right]^2 \right)$$

Il minimo di J(.) si ha quando le derivate rispetto ai parametri si annullano:

$$\frac{\partial J(.)}{\partial w_i} = -2 * \left(\sum_k w_k g(P; P_k \mid s_k) - y \right) * g(P; P_i \mid s_i) = 0$$

$$\frac{\partial J(.)}{\partial P_i} = -2 * \left(\sum_k w_k g(P; P_k \mid s_k) - y \right) * w_i \frac{\partial g(P; P_i \mid s_i)}{\partial P_i} = 0$$

$$\frac{\partial J(.)}{\partial s_i} = -2 * \left(\sum_k w_k g(P; P_k \mid s_k) - y \right) * w_i \frac{\partial g(P; P_i \mid s_i)}{\partial s_i} = 0$$

Non ci sono vincoli sui parametri. I valori di σ tendono a crescere ed i centri a concentrarsi al centro o a respingersi fuori dal campo recettivo.
Minimi locali.

Sommario

.

RBF: reti neurali con neuroni a base radiale.

- Struttura della rete.
- Apprendimento ibrido.
- Teoria della regolarizzazione.
- Stima Bayesiana.
- Teoria del filtraggio.
- Approccio gerarchico.
- Approccio gerarchico locale.

Learning ibrido

1) Determinazione di posizione e standard deviation delle unità.

2) Determinazione dei pesi sinaptici (perceptrone con unità di attivazione Gaussiane).

$$S(P \mid a) = \sum_{k=1}^{M} w_k G(P - P_k \mid s_k)$$

Determinazione di posizione ed ampiezza delle unità

$$S(P \mid \alpha) = \sum_{k=1}^{M} w_k g(P - P_k \mid \sigma_k)$$

Per determinare la posizione delle unità:

- Definizione del numero di unità.
- Posizionamento delle unità mediante clustering: minimizzazione di un errore di rappresentazione (k-means, fuzzy-clustering,…).

Vado ad inserire unità dove ci sono dati.

Per determinarne l'ampiezza:

- Si utilizzano euristiche di tipo P-nearest-neighbour.

  E.g. primo ordine: $\sigma = < \Delta x_{\alpha\beta} >$

  Dove α,β sono tutte le coppie di centri ottenute associando ad α il centro più vicino.

  σ controlla il grado di sovrapposizione tra due Gaussiane vicine.

La media può essere calcolata localmente: P-loc-nearest-neighbor.

Determinazione dei pesi sinaptici

$$S(P \mid a) = \sum_{k=1}^{M} w_k G(P - P_k \mid s_k)$$

Utilizzo la tecnica utilizzata per il perceptrone. Quale?

Ricordiamo che le reti RBF sono chiamate anche perceptroni non lineari o Gaussiani.

NB rilevanza dal punto di vista biologico e computazionale. Ho una struttura fissa: funzioni con campo recettivo locale, con queste unità posso realizzare funzioni complesse semplicemente agendo sui pesi. Le unità mi forniscono già una pre-elaborazione dell'input locale in spazio ma anche in frequenza.

Determinazione dei pesi sinaptici

$$S(P \mid a) = \sum_{k=1}^{M} w_k G(P - P_k \mid s_k)$$

Definisco una funzione costo e calcolo derivate rispetto ai **pesi:**

$$J(\{w_k\}) = \left(S(P_m) - S(P_m \mid a)\right)^2 = \left(S(P_m) - \sum_{k=1}^{M} w_k G(P_m - P_k \mid s_k)\right)^2$$

Indico i nostri esempi come $\{P_m, S(P_m)\}$

$$\frac{\partial J(.)}{\partial w_i} = -2 * \left(\sum_k w_k G\left((P - P_k) \mid s_k\right) - S(P)\right) * G\left((P - P_i) \mid s_i\right) = 0$$

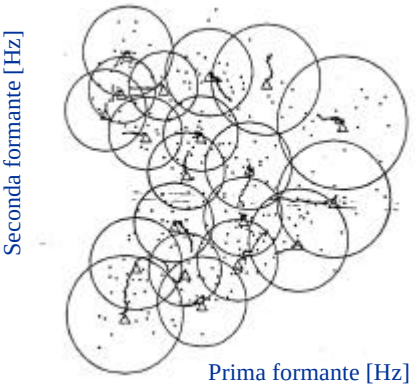
Scrivo l'equazione di cui sopra per ogni punto misurato (esempio) e avrò un sistema lineare nei pesi => "facilmente" risolvibile.

Risultati da Moody & Darken

Esperimento di riconoscimento automatico del parlato.

Analizzo in tempo reale lo spettro per estrarre le formanti.

Dalle formanti voglio riconoscere le vocali (sillabe).



Posizione e standard deviation delle unità della rete (20 unità). Queste unità mi ricostruiscono una superficie (pre-processing) su cui poi andrò ad applciare tecniche di clustering per individuare le diverse vocali.

Sommario

.

RBF: reti neurali con neuroni a base radiale.

- Struttura della rete.
- Apprendimento ibrido.
- Teoria della regolarizzazione.
- Stima Bayesiana.
- Teoria del filtraggio.
- Approccio gerarchico.
- Approccio gerarchico locale.

Approssimazione e RBF networks: ipotesi

Apprendimento da esempi è in molti casi equivalente ad approssimare una funzione multi-variabile.

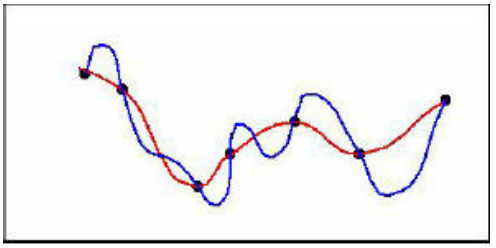
Dati $\{P(x_1, x_2, x_3 ....., x_M, x_{M+1})\}$ posso scrivere il mio set di esempi come: $\{P(\mathbf{x}, y(\mathbf{x}))\}$ con $y = x_{M+1}$ e $\mathbf{x} \in R^M$.

Ad esempio nello spazio 3D la mia funzione rappresenterà l'altezza della superficie z = f(x,y).

E' una ricostruzione 2½ D.

Interpolazione dei dati

Dato un insieme $\{y_i = f(\mathbf{x_i})\}$ => trovare $f(\mathbf{x})$.



Esistono infinite soluzioni.
E' un problema *mal-posto*.

Occorre introdurre delle ipotesi sull'andamento della funzione per "scegliere" quale funzione meglio rappresenta i dati.

Difficilmente siamo in grado di ricostruire andamenti complicati tra i punti, privilegeremo soluzioni "smooth".

RBF come soluzione di un problema di regolarizzazione

Dato un insieme $\{y_i = f(\mathbf{x_i})\}$ => trovare $f(\mathbf{x})$ non è sufficiente, si aggiungono ipotesi sulla funzione (stabilizzatori della soluzione).

Regolarizzazione -> Problema variazionale:

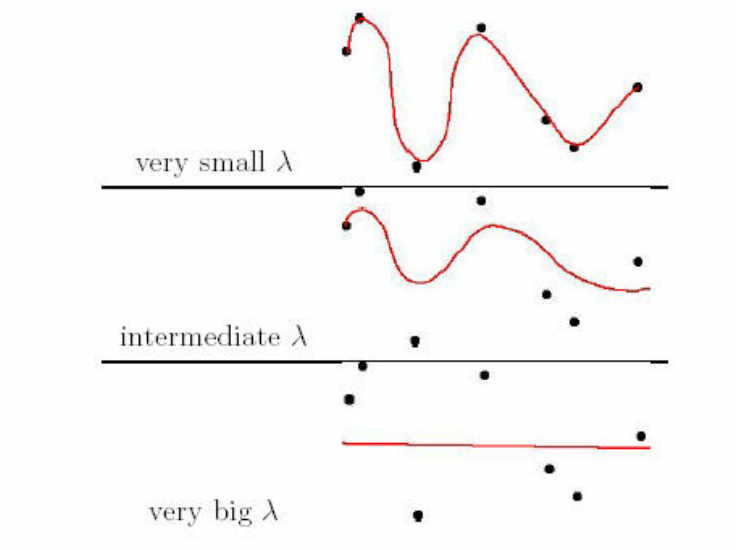
$$H[f] = \min_{\{f(.)\}} \left(\sum_i \left(f(\mathbf{x}_i) - y_i \right)^2 + l\,\Phi[f] \right)$$

Misura la fedeltà della soluzione f(.) ai campioni, grado di interpolazione.

Stabilizzatore (e.g. Penalizza le variazioni brusche), grado di smoothness.

λ è un parametro scelto dall'utente.

Ruolo del parametro l



Soluzione di un problema di regolarizzazione

$$H[f] = \min_{\{f(.)\}}\left(\frac{1}{2}\sum_i \left(f(\mathbf{x}_i) - y_i\right)^2 + l\,\Phi[f]\right) \quad ==> \quad \frac{dH[f]}{df} = 0$$

Trovo f(.) tale per cui H[f(.)] è minimo. E' una minimizzazione funzionale. Gestibile in pochi casi in forma analitica.

Esempio (interpolazione mediante spline cubica).

Utilizzo come stabilizzatore: $f[f(.)] = \int \frac{d^2 f(.)}{dx^2} dt$

$$\frac{dH[f]}{df} = 0 \qquad \frac{d}{df}\frac{1}{2}\sum_i \left(f(\mathbf{x}_i) - y_i\right)^2 + l\,\frac{d}{df}\int \frac{d^2 f(.)}{dx^2} dt = 0$$

RBF come soluzione di un problema di regolarizzazione

$$H[f] = \min_{\{f(.)\}} \left(\frac{1}{2} \sum_i \left(f(\mathbf{x}_i) - y_i \right)^2 + l \Phi[f] \right) \qquad \frac{dH[f]}{df} = 0$$

Scrivo il funzionale come: $\Phi[f] = \int_{R_D} \frac{\left| \tilde{f}(w) \right|^2}{\tilde{G}(w; s)} dw$

$\tilde{G}(w; s)$ è la trasformata di Fourier di una funzione a base radiale decrescente a 0 (ad esempio, una Gaussiana). In generale σ descrive il potere filtrante di G(.)

L'ampiezza di ogni armonica $\overline{w}$ di f(x), $\tilde{f}(\overline{w})$ viene divisa per $\tilde{G}(\overline{w}; s)$ l'ampiezza dell'armonica corrispondente del filtro. Ne risulta che armoniche ad alta frequenza producono un costo elevato.

La soluzione f(x) ha uno spazio nullo. Sono le funzioni p(x), che non alterano l'integrale, e si annullano nei punti x_i.

Sviluppo della soluzione (I)

Trasformiamo secondo Fourier anche la funzione f(x):

$$f(x) = C \int_{R^D} dw \, \tilde{f}(w) e^{ixw}$$

Il funzionale diventa:

$$H[\tilde{f}] = 1/2 \sum_{i=1}^{N} \left(y_i - C \int_{R^D} dw \, \tilde{f}(w) e^{ixw} \right)^2 + l \int_{R^D} dw \frac{\left| \tilde{f}(w) \right|^2}{\tilde{G}(w; s)}$$

espresso nel dominio delle frequenze.

La norma di $\tilde{f}(w)$ coincide con la norma di f(t) negli spazi di Hilbert. La norma della trasformata coincide con la norma della funzione.

Sviluppo della soluzione (II)

$$H[\tilde{f}]=1/2\sum_{i=1}^{N}\left(y_i - C\int_{R^D} dw\ \tilde{f}(w)e^{ixw}\right)^2 + l\int_{R^D} dw\ \frac{\left|\tilde{f}(w)\right|^2}{\tilde{G}(w;s)}$$

Minimizzo rispetto alla trasformata di f(x):

$$\frac{dH[\tilde{f}(w)]}{d\tilde{f}(.)}=\sum_{i=1}^{N}\left(y_i - C\int_{R^D} dw\ \tilde{f}(w)e^{ixw}\right)^2 + l\int_{R^D} dw\ \frac{\left|\tilde{f}(w)\right|^2}{\tilde{G}(w;s)}$$

Dato che f(x) è reale, la sua trasformata sarà pari (simmetrica) e quindi: $\tilde{f}^*(w)=\tilde{f}(-w)$

$\tilde{G}(w;s)$ Viene supposta simmetrica (cf. Gaussiana), per cui la sua (anti)trasformata sarà reale.

Sviluppo della soluzione (III)

Consideriamo il primo termine: $\frac{d}{d\tilde{f}(v)}\left\{1/2\sum_{i=1}^{N}\left(y_i - C\int_{R^D} dw\ \tilde{f}(w)e^{ixw}\right)^2\right\}$

$$= \left\{\sum_{i=1}^{N}\left(y_i - C\int_{R^D} dw\ \tilde{f}(w)e^{ixw}\right)\int_{R^D} dw\ \frac{d\tilde{f}(w)}{d\tilde{f}(v)}e^{ixw}\right\}$$

Antitrasformo

$$= \left\{\sum_{i=1}^{N}\left(y_i - f(x_i)\right)\int_{R^D} d(w-v)e^{ixw}dw\right\}$$

$$= \left\{\sum_{i=1}^{N}\left(y_i - f(x_i)\right)e^{ixv}\right\}$$

Sviluppo della soluzione (IV)

Consideriamo il secondo termine: $\frac{d}{d\tilde{f}(v)}\left\{\int_{R^D}\frac{\tilde{f}(-w)\tilde{f}(w)}{\tilde{G}(w;s)}dw\right\}$

$$=\int_{R^D}\frac{\tilde{f}(-w)}{\tilde{G}(w;s)}\frac{d\tilde{f}(w)}{d\tilde{f}(v)}dw$$

$$=2\int_{R^D}\frac{\tilde{f}(-w)}{\tilde{G}(w;s)}d(w-v)dw$$

$$=2\frac{\tilde{f}(-v)}{\tilde{G}(v;s)}$$

Sviluppo della soluzione (V)

Ricapitolando, nel dominio delle frequenze:

$$\frac{dH[\tilde{f}(.)]}{d\tilde{f}(v)}=0 \quad ? \quad \left\{\sum_{i=1}^{N}(y_i-f(x_i))e^{ixv}\right\} + \lambda\frac{\tilde{f}(-v)}{\tilde{G}(v;s)} = 0$$

Sostituiamo ν a –ν (le trasformate non cambiano perchè sono simmetriche per ipotesi). Moltiplicando entrambi i membri per $\tilde{G}(w)$:

$$\tilde{f}(v)=\tilde{G}(-v;s)\frac{\sum_{i=1}^{N}\left(y_i-f(x_i)e^{ixv}\right)}{l}e^{ix_iv}$$

Sviluppo della soluzione (VI)

$$\tilde{f}(v) = \frac{\sum_{i=1}^{N} (y_i - f(x_i))}{l} \tilde{G}(-v; s) e^{ix_i v}$$

Non dipende da w

Prodotto nel dominio delle frequenze = convoluzione nel dominio dello spazio.

Definisco i coefficienti: $w_i = \frac{y_i - f(x_i)}{l}$

Termini ortogonali al funzionale (polonomi), L dimensione dello spazio nullo.

Da cui antitrasformando si ottiene:

$$f(x) = \sum_{i=1}^{N} w_i d(x - x_i) * G(x; s) = \sum_{i=1}^{N} w_i G(x - x_i; s) + \sum_{l=1}^{L} d_l Q_l(\mathbf{x})$$

Analisi della soluzione

$$f(x) = \sum_{i=1}^{N} c_i d(x - x_i) * G(x; s) = \sum_{i=1}^{N} w_i G(x - x_i; s) + \sum_{l=1}^{L} d_l Q_l(\mathbf{x})$$

$$w_i = \frac{y_i - f(x_i)}{l} \qquad d_l = 0$$

Tante unità (N) quanti sono il numero di punti.

Se G(.) è una Gaussiana otteniamo RBF Gaussiane.

Due elementi che regolano la smoothness della soluzione

- Standard deviation, σ, di G(.), uguale per tutte le unità.
- Grado di fedeltà ai dati λ.

Come determino il valore dei pesi {w_i}

$$f(x)=\sum_{i=1}^{N} w_i G(x-x_i;s) \qquad w_i=\frac{y_i-f(x_i)}{l}$$

f(.) dipende dai w_i

Ma $f(x_k)$ è una combinazione lineare di funzioni radiali.

$$f(x_j)=\sum_{i} w_i G(x_j-x_i;s)+\sum_{l=1}^{L} d_l Q_l(x_j)$$

Per ogni punto m misurato, possiamo scrivere:

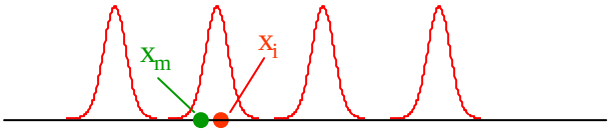
$$y_m = f(\mathbf{x_m}) + \lambda\, w_m$$

Abbiamo tanti w_m quanti sono gli esempi ([$\mathbf{x_m}$, y_m])

Determinazione dei {w_i} (I)

$$f(x)=\sum_{i=1}^{N} w_i G(x-x_i;s) \qquad w_i=\frac{y_i-f(x_i)}{l}$$

f(.) dipende dai w_i



Se le Gaussiane hanno domini separati, preso un punto, x_m, in un intorno del centro della i-esima Gaussiana, x_i,

$f(x)=\sum_{i=1}^{N} w_i G(x-x_i;s)$ avrà come componente ? 0 solamente: $w_i G(x-x_i;s)$

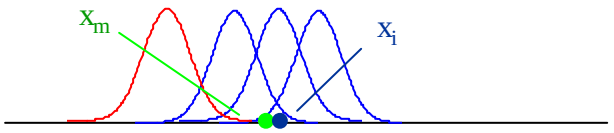
➔Ciascun peso, w_i, può essere determinato solamente guardando alla coppia di valori [y_m, $w_i G(x_m-x_i;\sigma)$], per tutti i punti che stanno nel campo recettivo della Gaussiana i-esima. Problemi disaccoppiati.

G(x-xi;σ) non estende la sua influenza sugli altri punti (è utile?).

Determinazione dei {w_i} (II)

$$f(x)=\sum_{i=1}^{N} w_i G(x-x_i;s) \qquad w_i=\frac{y_i-f(x_i)}{l}$$

f(.) dipende dai w_i



Se le Gaussiane hanno domini parzialmente sovrapposti, preso un punto, x_m, in un intorno del centro della i-esima Gaussiana, x_i, un certo numero di Gaussiane daranno un contributo non nullo a $f(x)=\sum_{i=1}^{N} w_i G(x-x_i)$ quali?

➔Ciascun peso non può essere determinato solamente guardando alla coppia di valori [y_m, $w_i G(x_m-x_i;\sigma)$] per quei punti che cadono nel campo recettivo della Gaussiana i-esima.

G(x-xi; σ) estende l'influenza sugli altri punti; occorre tenerne conto. Come?

Determinazione dei {w_i}: soluzione matriciale (I)

$$f(x)=\sum_{i=1}^{N} w_i G(x-x_i;s) \qquad y_i=f(x_i)+l\,w_i$$

f(.) dipende da tutti i w_i

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \dots\dots \\ y_N \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^{i=N} w_k G(x_1-x_i;s) \\ \sum_{i=1}^{i=N} w_k G(x_2-x_i;s) \\ \sum_{i=1}^{i=N} w_k G(x_3-x_i;s) \\ \dots\dots \\ \sum_{i=1}^{i=N} w_k G(x_N-x_i;s) \end{bmatrix} + \lambda \begin{bmatrix} w_1 \\ w_2 \\ w_3 \\ \dots\dots \\ w_N \end{bmatrix}$$

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \dots\dots \\ y_N \end{bmatrix} = \begin{bmatrix} G(x_1-x_1;s_1) & G(x_1-x_2;s_2) & G(x_1-x_3;s_3) & G(x_1-x_N;s_N) \\ G(x_2-x_1;s_1) & G(x_2-x_2;s_2) & G(x_2-x_3;s_3) & G(x_2-x_N;s_N) \\ G(x_3-x_1;s_1) & G(x_3-x_2;s_2) & G(x_3-x_3;s_3) & G(x_3-x_N;s_N) \\ & & & \\ G(x_N-x_1;s_1) & G(x_N-x_2;s_2) & G(x_N-x_k;s_3) & G(x_5-x_N;s_N) \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ w_3 \\ \dots\dots \\ w_N \end{bmatrix} + \lambda \begin{bmatrix} w_1 \\ w_2 \\ w_3 \\ \dots\dots \\ w_N \end{bmatrix}$$

Determinazione dei {w_i}: soluzione matriciale (I I)

$$f(x)=\sum_{i=1}^{N} w_i G(x-x_i;s) \qquad y_i = f(x_i) + l\, w_i$$

f(.) dipende da tutti ia w_i

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \ldots\ldots \\ y_N \end{bmatrix} = \begin{bmatrix} G(x_1-x_1;s_1) & G(x_1-x_2;s_2) & G(x_1-x_3;s_3) & G(x_1-x_N;s_N) \\ G(x_2-x_1;s_1) & G(x_2-x_2;s_2) & G(x_2-x_3;s_3) & G(x_2-x_N;s_N) \\ G(x_3-x_1;s_1) & G(x_3-x_2;s_2) & G(x_3-x_3;s_3) & G(x_3-x_N;s_N) \\ & & & \\ G(x_N-x_1;s_1) & G(x_N-x_2;s_2) & G(x_N-x_k;s_3) & G(x_5-x_N;s_N) \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ w_3 \\ \ldots\ldots \\ w_N \end{bmatrix} + \lambda \begin{bmatrix} w_1 \\ w_2 \\ w_3 \\ \ldots\ldots \\ w_N \end{bmatrix}$$

$\mathbf{y} = (\mathbf{G} + \lambda\, \mathbf{I})\mathbf{w}$ $\quad$ $\mathbf{y},\mathbf{w}$: N x 1; λ scalare, $\mathbf{G}$, $\mathbf{I}$: N x N

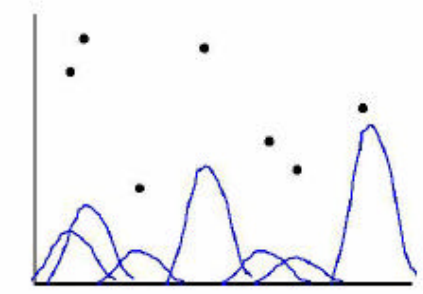
➔ $w = (G + \lambda\, I)^{-1}\, y$

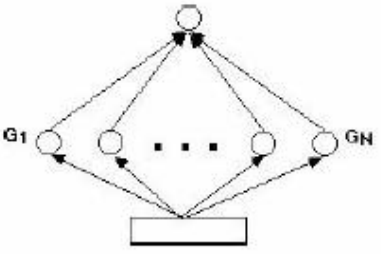
• λ -> ? ∞ il peso i-esimo è una frazione (che tende a 0) dell'altezza misurata per il campione i-esimo.

• λ -> ? 0 il peso i-esimo dipende dalla forma di G. Se s è molto piccola coinciderà con ym, altrimenti, se più campioni cadono sotto la stessa Gausiana, sarà una versione filtrata degli stessi.

Come agisce la regolarizzazione?

Ciascuna funzione radiale viene posta in corrispondenza di un campione e cerca quindi di interpolare in modo smooth tra gli stessi.





La struttura della rete è data dal numero di punti e dalla loro posizione, i pesi vengono calcolati tramite un sistema lineare che tiene conto dei vincoli di interpolazione e smoothness.

Sommario

.

RBF: reti neurali con neuroni a base radiale.

- Struttura della rete.
- Apprendimento ibrido.
- Teoria della regolarizzazione.
- Stima Bayesiana.
- Teoria del filtraggio.
- Approccio gerarchico.
- Approccio gerarchico locale.

Approccio Bayesiano

Supponiamo che i nostri esempi: $\{\mathbf{x_i}; y_i\}$ siano ottenuti campionando in modo randomico una funzione f(.) non nota, in presenza di rumore: $y_i = f(\mathbf{x_i}) + \varepsilon_i$ (i=1, ...N). Gli ε_i sono indipendenti, sono funzioni randomiche che rappresentano il rumore con una sua distribuzione statistica.

Supponiamo di considerare la funzione f (.) come una realizzazione di un campo randomico con una certa distribuzione a-priori, che indichiamo con P[f]. Questo rappresenta la conoscenza a-priori sul campo di funzioni da cui è estratta e si può utilizzare per dare un vincolo soft alla soluzione, favorendo quelle funzioni di probabilità che meglio soddisfano le ipotesi a prori sul campo di funzioni.

Approccio Bayesiano: formalizzazione

Definisco f(.) la mia funzione probabilità da determinare e g(.) = {**xi**; yi} l'insieme dei dati.

- P[f|g] – Probabilità **condizionata** della funzione f, dati i campioni g.
- P[g|f] – Probabilità **condizionata** di ottenere i dati g da un modello f: probabilità di ottenere i valiri $\{y_i\}$ campionando la funzione nei $\{\mathbf{x_i}\}$: rappresenta un modello dell'errore.
- P[f] – Probabilità **a-priori** sul campo randomico f.

Dal teorema di Bayes si ottiene: P[f|g] = P[g|f] P[f]

Approccio Bayesiano: le ipotesi

- P[g|f] – Assumiamo che il rumore di misura (ε_i) sia normalmente distribuito con varianza σ.

$$P[g \mid f] = e^{-\frac{1}{2s^2}\sum_{k=1}^{N}(y_i - f(\mathbf{x_i}))^2}$$

• P[f] – Viene scelto in modo analogo al termine di regolarizzazione: diamo probabilità alta a funzioni che hanno costo basso:

$$P(f(.)) = e^{-af[f]}$$

Tramite la regola di Bayes, otteniamo:

$$P(f \mid g) = e^{-\frac{1}{2s^2}\sum_{k=1}^{N}(y_i - f(\mathbf{x_i}))^2} e^{-af(f)} = e^{-\frac{1}{2s^2}\left[\sum_{k=1}^{N}(y_i - f(\mathbf{x_i}))^2 + as^2 f[f]\right]}$$

Un modo classico di stimare f è mediante la MAP, f(.) che minimizza l'esponente ➔ problema di regolarizzazione con $\lambda = 2\sigma^2\alpha$.

se

Sommario

.

RBF: reti neurali con neuroni a base radiale.

- Struttura della rete.
- Apprendimento ibrido.
- Teoria della regolarizzazione.
- Stima Bayesiana.
- Teoria del filtraggio.
- Approccio gerarchico.
- Approccio gerarchico locale.

Autoscan

$\{z_i = S(\mathbf{x_i}, y_i)\}$

- Pair of video-cameras + standard laser pointer.
- The range data are obtained by "painting" the surface manually.
- Set of range data, which is denser where required.
- High precision in spot localization (cross-correlation, bright image).

Drawback: High scanning time.

RBF networks e filtri: ipotesi

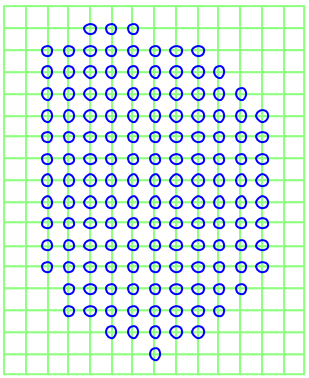
Apprendimento da esempi è in molti casi equivalente ad approssimare una funzione multi-variabile.

Dati $\{P(x_1, x_2, x_3 ....., x_M, x_{M+1})\}$ posso scrivere il mio set di esempi come: $\{P(\mathbf{x}, y(\mathbf{x}))\}$ con $y = x_{M+1}$ e $\mathbf{x} \in R^M$.

Nel caso dello scanner, i punti 3D rappresentano dei punti nello spazio geometrico 3D (e la ricostruzione è 2½ D), ma possono essere punti in uno spazio di qualsiasi dimensionalità: possono essere punti in uno spazio cognitivo, di controllo.....

The RBF model

$$S(P) = \sum_k^M w_k G(P - P_k \mid s_k)$$



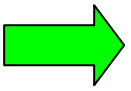
Structural parameters to be set:

- M – Number of units.
- $\{P_k\}$ – Position of the units.
- σ_k – Gaussian width.

Linear parameters:

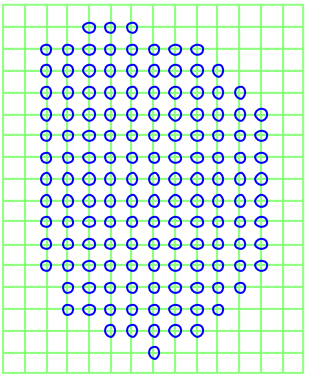
- $\{w_k\}$ – "Weights".

Grid support



Simple computation

RBF e Filtri



$$S(P)=\sum_k^M w_k G(P-P_k \mid s_k)$$

Gaussiane equispaziate su un grid.

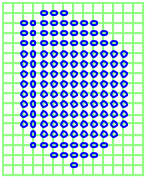
$|P_{k+1} - P_k| = \Delta P_k$
$\sigma_k = \sigma \ \forall \kappa$
$w_k = S(P_k)$

Gaussiane normalizzate $||g(.)|| = 1$

$$S(P)=\sum_{k=1}^{M} S_{m_k} G(P-P_k \mid s) =$$
$$S_{m_1} G(P-P_1 \mid s)+S_{m_2} G(P-P_2 \mid s)+\ldots =$$
$$S_m(.) * G(P-. \mid s)$$

G(.) costituisce un filtro passa-basso: $S(P)=S_m(.)*G(P-.\mid s)$

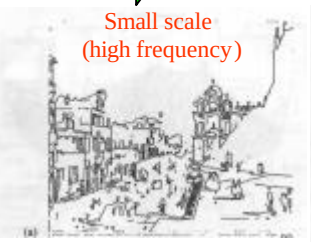
Role of the scale s

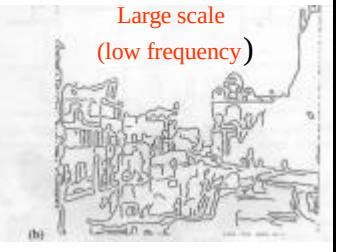


Linear filter (low-pass)

$$S(P)=\sum_k^M S_k G(P-P_k \mid s)$$

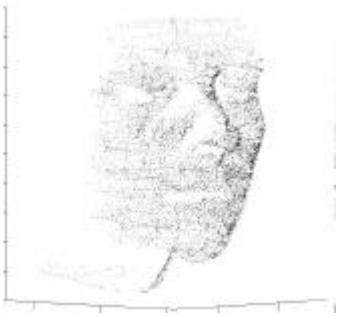
σ – Scale of the filter (bandwidth)

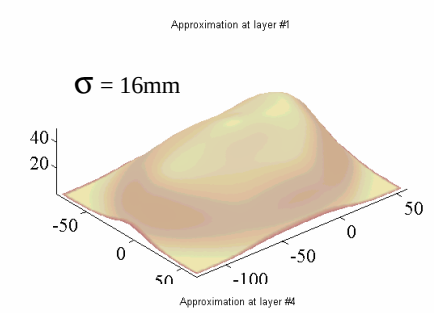




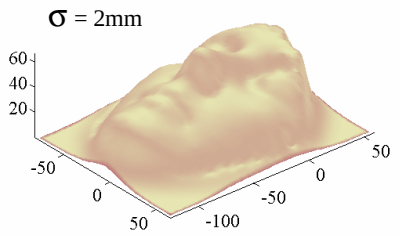
Setting s, we set the spatial frequency of the reconstruction.

Ricostruzione da renge data a scala diversa





$$S(P)=\sum_k^M S_k G(P-P_k \mid s)$$



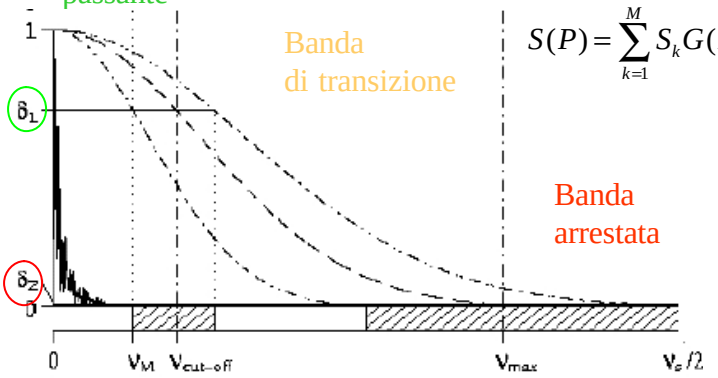
RBF come filtro -> criterio per s

Banda passante

Banda di transizione

Banda arrestata

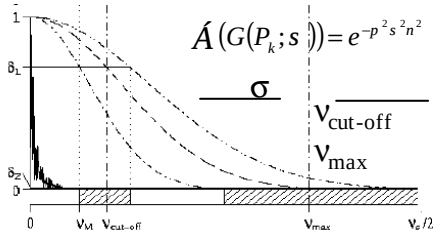
$$S(P)=\sum_{k=1}^{M} S_k G(P-P_k \mid s)$$



ν_M Massima frequenza fatta passare inalterata (e.g. attenuazione: -3dB).
ν_{max} Massima frequenza del filtro (e.g. ampiezza filtro: -40dB).

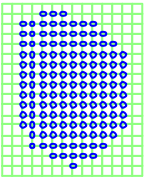
Al diminuire di σ, aumenta la frequenza -> aumenta la frequenza di campionamento (vicinanza tra due Gaussiane.σ → ν_{max} → ν_S→ ΔP_k

Parameters computation



Quasi-finite support

Con δ_1 = -3dB e δ_2 = -40dB
➔ $s_1 = 1.465\ DP_l$ equivalente ad un 75% di overlap tra Gaussiane adiacenti.



Conservative with respect to Parzen window estimators (65% of overlap)

$$S(P) = \sum_k^M S_k G(P - P_k \mid s)$$

σ ⇒ ΔP {Pk} ⇒ M

All the structural parameters have been set.
With {Sk} the surface could be reconstructed. But we do not have them

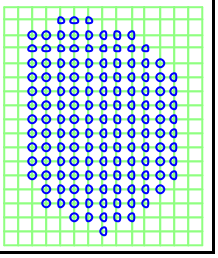
Computation of the $\{S_k\}$

$$S(P) = \sum_k^M S_k G(P - P_k \mid s)$$

We do not have the height of the surface in the grid crossing, **$\{S_k\}$**, and we must estimate it.

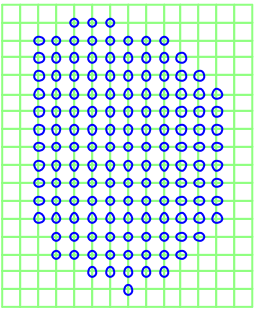
$$S_k = \frac{\sum_m S_m(P_m) f(|P_m - P_k|)}{\sum_m f(|P_m - P_k|)}$$

We can apply weighted mean estimate where each point is estimated with a function, f(|.|), of its distance from P_k.



Riassunto sulla configurazione delle RBF network

$$S(P)=\sum_k^M S_k G(P-P_k \mid s)$$



- Scelta di un valore di σ adeguato.
- Calcolo dell'"impaccamento" delle Gaussiane: Gaussiane. $\sigma \rightarrow \nu_S \rightarrow \Delta P_k$ (teoria del filtraggio).
- Stima dell'altezza della superficie nei grid crossings.

$$S_k = \frac{\sum_m S_m(P_m) f(|P_m - P_k|)}{\sum_m f(|P_m - P_k|)}$$

Problems

$$S(P) = \sum_{k=1}^{M} S_k g(P - P_k \mid \sigma) \qquad S_k = \frac{\sum_m S_m(P_m) f(|P_m - P_k|)}{\sum_m f(|P_m - P_k|)}$$

Related to the choice of σ:

• No information on spatial frequency content.

• Frequency variability. σ should be chosen small enough to reconstruct the finest details (high packing). Few points close to P_k to estimate $S_k = S(P_k)$.

No control on the reconstruction error.

$\{S_k\}$ are computed using all the data points. No locality assumption.

The approach as it is has little adaptivity and little hope to be real-time.

Sommario

.

RBF: reti neurali con neuroni a base radiale.

- Struttura della rete.
- Apprendimento ibrido.
- Teoria della regolarizzazione.
- Stima Bayesiana.
- Teoria del filtraggio.
- Approccio gerarchico.
- Approccio gerarchico locale.

Strategia costruttiva di s(P)

Inizio con una scala molto ampia e valuto il residuo.

Approssimazione del residuo, dove presente, ad una scala più piccola.

Calcolo di un secondo residuo

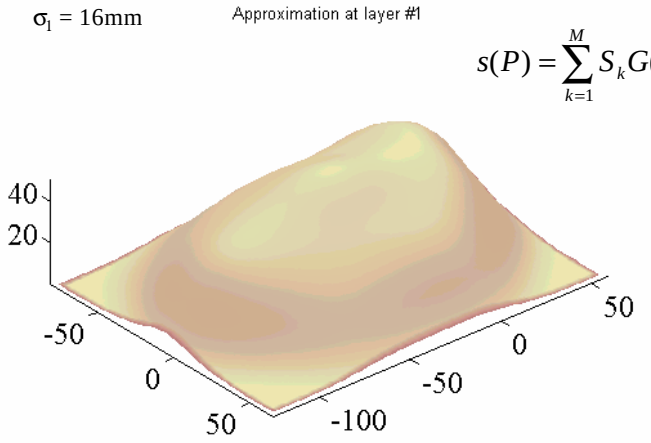
…..

Fino a che l'errore di approssimazione soddisfa il mio criterio di approssimazione (cross-validation, contenuto in frequenza del rumore, rischio empirico o strutturale…).

NB Le Gaussiane vengono inserite negli incroci della griglia **solamente** dove l'errore (residuo) è sopra soglia (*calcolo del residuo locale*).

Costruzione di un primo livello a scala ampia

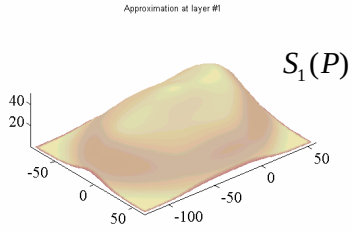
σ_1 = 16mm



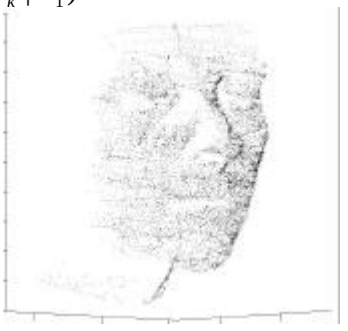
$$s(P) = \sum_{k=1}^{M} S_k G(P - P_k \mid s_1)$$

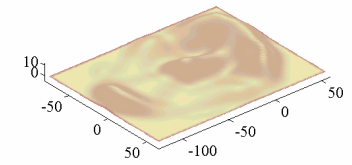
Calcolo del residuo

Residuo della ricostruzione: $r_1 = \{z_m - S_1(P)\}$ $\forall$P misurato, P_m.



$$S_1(P) = \sum_{k=1}^{M} S_k G(P - P_k \mid s_1)$$

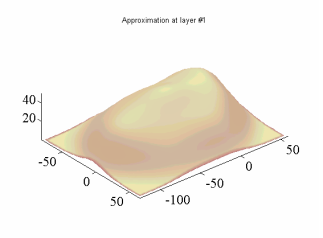




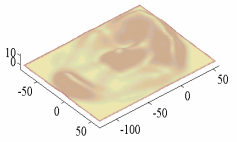
Residuo: $\{z_m - S_1(P_m)\}$

Error evaluation

Key element is the residual: $r(P_m) = z_m - S_1(P_m)$.



Continuos surface at a low scale (σ = 16mm)



Residual at the sampled points

The residual is defined only in the surface sampled points (range data).

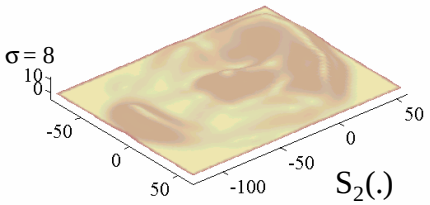
The residual is evaluated using an integral metric to avoid outliers (spike reproduction). The error can be computed as: $e = \frac{\sum_m |r_1(P_m)|}{N}$

Costruzione di un secondo livello gerarchico

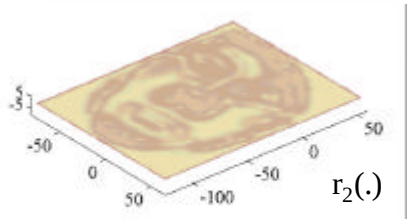
Al layer 1:

$$\text{Error}(P_k) = \frac{\sum_m |r_1(P_m) - S_1(P_m)|}{N_k}$$

Output of layer #2



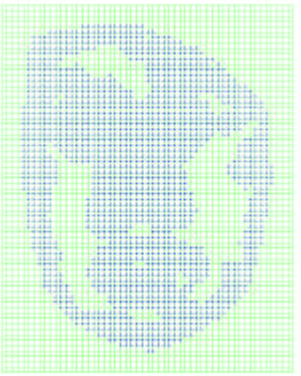
$S_2(.)$



$r_2(.)$

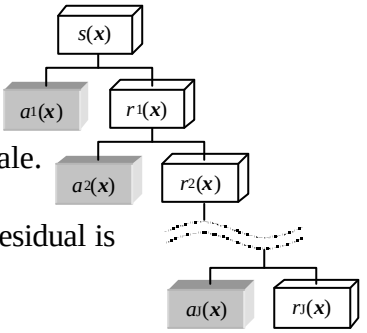
NB nei livelli successivi si approssima il residuo = si aggiunge dettaglio.

Layer #2



The HRBF configuration algorithm

1) Start with a large scale and reconstruct the surface.
2) Compute the residual.



Do:

2) Create a denser grid (lower scale).
3) Compute the residual at this lower scale.
4) Evaluate the residual.
5) Insert the Gaussians only where the residual is under-threshold.

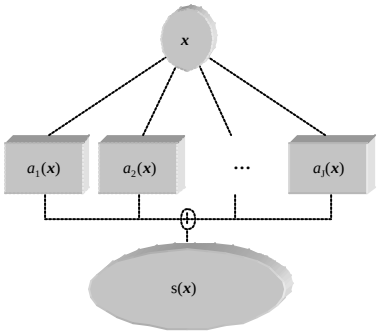
Until:

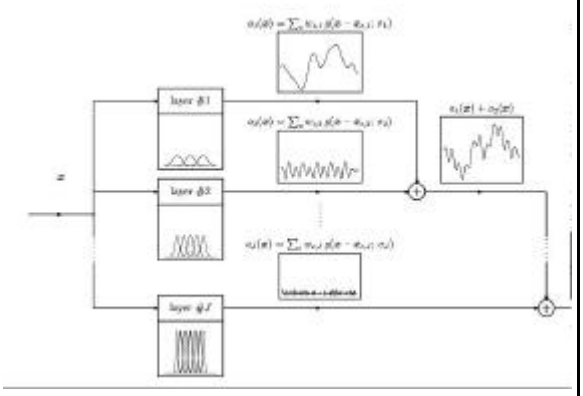
6) Until the residual goes under threshold.

Reconstruction is obtained by adding the approximations: $s(\mathbf{x}) = \sum_j a_j(\mathbf{x})$

Each approximation is obtained as: $S(\mathbf{x}) = \sum_k^M S_k G(\mathbf{x} - \mathbf{x_k} \mid s)$

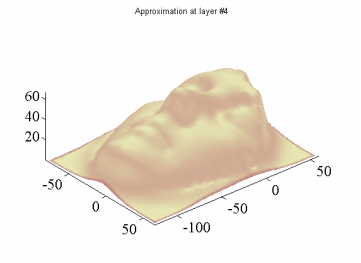
Sintesi della superficie



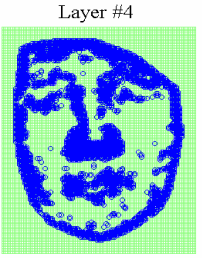


•Diversi strati (reti RBF).
•Ciascuno strato opera ad una certa scala.
•Gli strati non sono completi ma allocano Gaussiane solo dove l'errore locale è sopra-soglia.

HRBF Networks



Noise



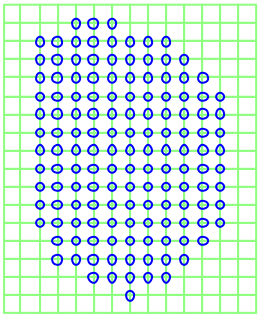
Sommario

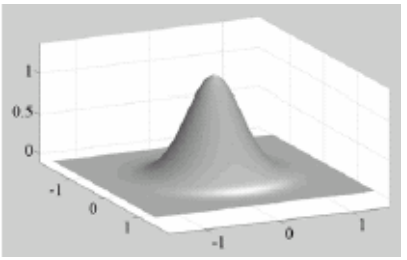
.

RBF: reti neurali con neuroni a base radiale.

- Struttura della rete.
- Apprendimento ibrido.
- Teoria della regolarizzazione.
- Stima Bayesiana.
- Teoria del filtraggio.
- Approccio gerarchico.
- Approccio gerarchico locale.

Sfruttamento della quasi-localita'



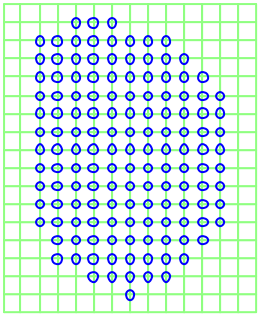


I campi recettivi sono parzialmente sovrapposti.

Operazioni quasi-locali sui punti all'interno dei campi recettivi:

- Calcolo dell'altezza della funzione nell'incrocio.
- Calcolo dell'errore locale.

Where locality is interesting?



$$S(P) = \sum_{k=1}^{M} S_k g(P - P_k \mid \sigma)$$

Estimate of the Surface in the grid crossings:

$$S_k = \frac{\sum_m S_m(P_m) f(|P_m - P_k|)}{\sum_m f(|P_m - P_k|)}$$

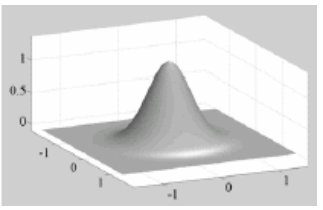
Computation of the error: $e = \frac{\sum_m |r(P_m)|}{N}$

and its association to the grid crossings [*"5) Insert the Gaussians only where the residual is under-threshold."*]

Costly operation because f(.) has to be computated for all the data points and there is no criterion on how to evaluate locally e.

Receptive field

$$S(P)=\sum_{k=1}^{M} S_k G(P-P_k \mid s)$$



- $S(P_k)$ is estimated as a weighted (with the distance) mean of the points which lie inside the receptive field of the Gaussian:

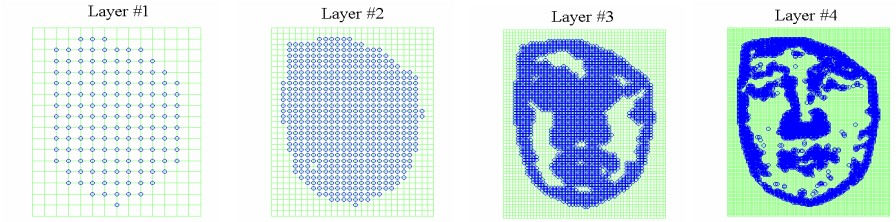
$$S_k = S(P_k)=\frac{\sum_m S(P_m) g(P_m - P_k \mid s)}{\sum_m g(P_m - P_k \mid s)} \qquad P_m \in RF(P_k)$$

- An errror measure can be associated to each Gaussian as the residual computed to all the points inside the receptive field.

$$e(P_k)=\frac{\sum_m |r(P_m)|}{N(P_k)} \qquad P_m \in RF(P_k)$$

Sparse approximation

Grids do not need to be complete, but Gaussians can be inserted only in those crossings where the error is over threshold.

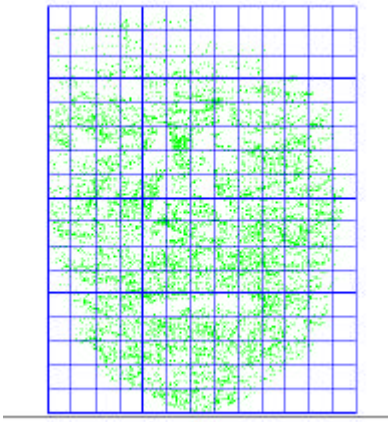


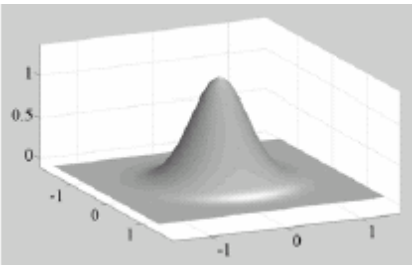
We obtain a sparse approximation (less dense units, where less dense sampling and larger details).

Local control of the reconstruction error. Error globally under threshold.

Representation of data locality

Approximate the Receptive field with squares (Manhattan distance).

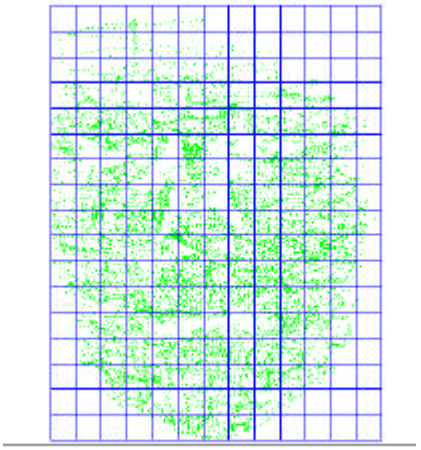




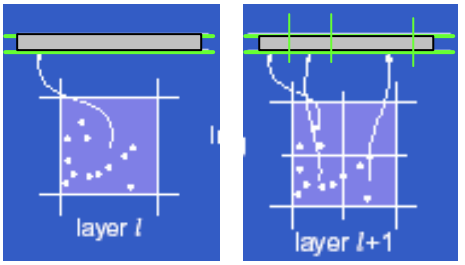
Quad-tree subdivision

Data are partitioned into parallelepipeds supported by the grid at the lowest (largest) scale.

Efficient data storage



Quad-tree subdivision by means of in-place sorting.

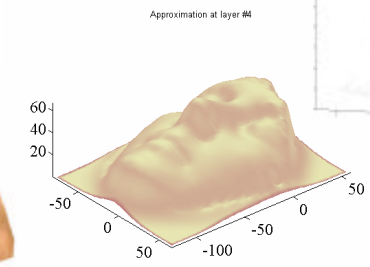


Block of range data associated to the grid square

Data can be retrieved directly.

At every layer, only data inside the receptive field have to be sorted.

Quantitative results



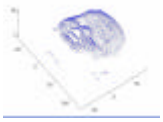
#Layers	Mean error [mm]	Error std [mm]	# of gaussians	σ [mm]
1	2.33	4.38	141/270	16
2	0.03	2.06	573/980	8
3	0.04	1.02	1,778/3,795	4
4	0.06	0.77	3,078/14,933	2

Data set cardinality: 12,471. Processing time on a Pentium III, 800Mhz, was < 1s.

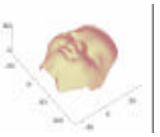
Summary of HRBF

- Incremental Reconstruction (multi-scale).
- Local computation.
- Error-driven (local adaptation of the scale).
- Local control on the error.

Real object



Range data



3D surface