



# L'intelligenza biologica

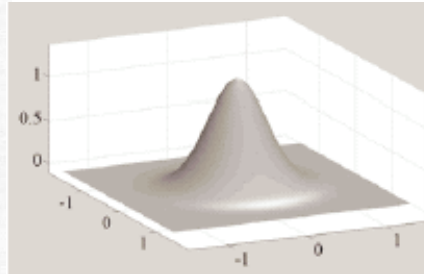
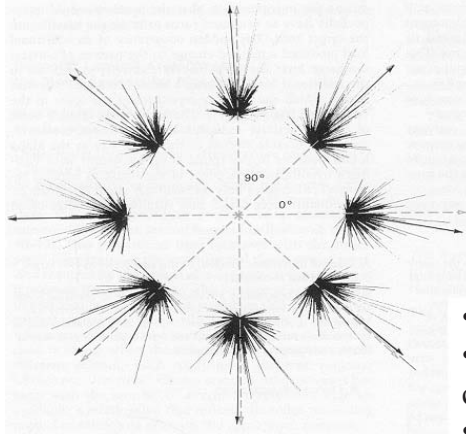
## Reti Neurali con funzione di attivazione a base radiale

Copyright N.A. Borghese Università di Milano 11/04/2003

<http://homes.dsi.unimi.it/~borghese> 1/45



## Funzione di attivazione radiale



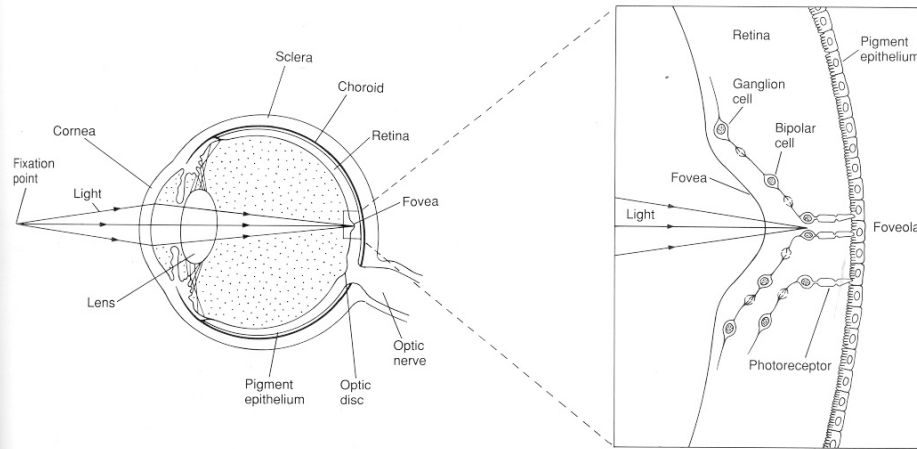
- Campo recettivo.
- Attivazione che cresce con la centralità dello stimolo.
- Attivazione che decresce esponenzialmente dal centro.
- Rapidità di diminuzione regolata dalla deviazione standard.

Copyright N.A. Borghese Università di Milano 11/04/2003

<http://homes.dsi.unimi.it/~borghese> 2/45



# Visione - l'occhio umano



Copyright N.A. Borghese Università di Milano 11/04/2003

<http://homes.dsi.unimi.it/~borghese> 3/45



# Neuroni gangliari

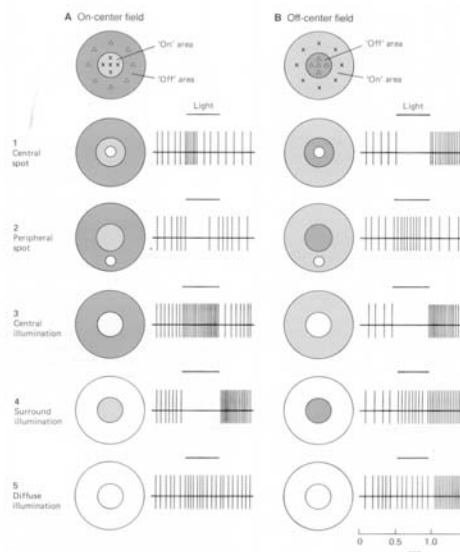


## Neuroni gangliari.

- Campo recettivo (rispondono a variazioni di luce in una regione localizzata dello spazio).
- Risposta a pattern spaziali e temporali.

## Campo recettivo.

- Frequenza di background (adattamento alla quantità di luce incidente).
- Circolare (dimensioni diverse).
- On-Off (elaborazione parallela).
- Trasmette informazione differenziale.



18-7

Copyright N.A. Borghese Università di Milano 11/04/2003

<http://homes.dsi.unimi.it/~borghese> 4/45



## Significato computazionale



- Trasmissione differenziale.
- Maggiore sensibilità a variazioni spaziali e temporali locali della luminosità.
- Robustezza verso il rumore nella catena di trasmissione.
- Riconoscimento delle variazioni locali di luminosità (bordi e movimento).

**Le cellule gangliari codificano anche altri aspetti delle immagini: colore.**

Copyright N.A. Borghese Università di Milano 11/04/2003

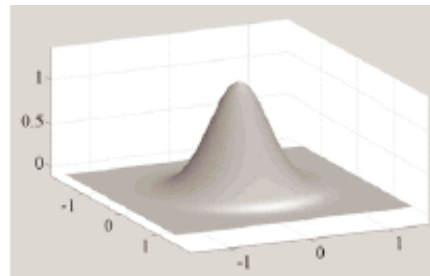
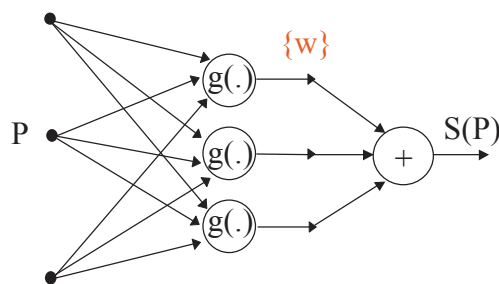
<http://homes.dsi.unimi.it/~borghese> 5/45



## Reti con elementi di attivazione radiali (RBF)



Perceptrone con unità quasi-locali. 
$$S(P) = \sum_{k=1}^M w_k g(P - P_k | \sigma_k)$$



$g(\cdot)$  Gaussiana.

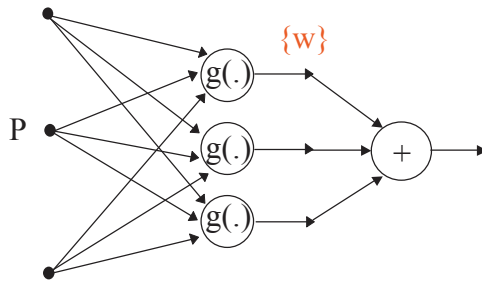
Distribuiamo le Gaussiane nell'insieme di definizione e calcoliamo la superficie come somma pesata.

Copyright N.A. Borghese Università di Milano 11/04/2003

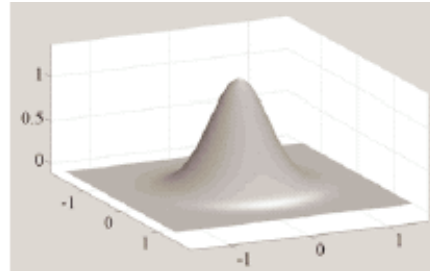
<http://homes.dsi.unimi.it/~borghese> 6/45



## Parametri delle RBF



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



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

- Per ogni Gaussiana si può definire un campo recettivo (regione di influenza). L'ampiezza è determinata da  $\sigma_k$ .
- Le regioni di influenza non sono disgiunte per le diverse Gaussiane.

Copyright N.A. Borghese Università di Milano 11/04/2003

<http://homes.dsi.unimi.it/~borghese> 7/45



## Determinazione dei parametri (supervised learning)



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

$$\min_{\{\alpha\}} (S[(P | \alpha) - y]^2)$$

$$\frac{\partial S}{\partial w_i} = -2 * \left( \sum_i w_i g(P; P_i | \sigma_i) \right) * g(P; P_i | \sigma_i)$$

$$\frac{\partial S}{\partial P_i} = -2 * \left( \sum_i w_i g(P; P_i | \sigma_i) \right) * \frac{\partial g(P; P_i | \sigma_i)}{\partial P_i}$$

$$\frac{\partial S}{\partial \sigma_i} = -2 * \left( \sum_i w_i g(P; P_i | \sigma_i) \right) * \frac{\partial g(P; P_i | \sigma_i)}{\partial \sigma_i}$$

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

Copyright N.A. Borghese Università di Milano 11/04/2003

<http://homes.dsi.unimi.it/~borghese> 8/45



## Learning ibrido



- 1) Determinazione di posizione e standard deviation delle unità.
- 2) Determinazione dei pesi sinaptici (perceptrone con unità di attivazione Gaussiane).

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



## Determinazione di posizione ed ampiezza delle unità



### *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,...).

### *Per determinarne l'ampiezza:*

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

Primo ordine:  $\sigma = \langle \Delta X_{\alpha\beta} \rangle$

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



## Approssimazione e RBF networks



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

L'approssimazione di una funzione da campioni sparsi e rumorosi può essere riformulato come un problema di controllo ottimo (problema di regolarizzazione).

La soluzione è una rete RBF sotto determinate ipotesi.



## RBF come soluzione di un problema di regolarizzazione



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

Il problema è *mal-posto*.

Regolarizzazione -> Problema variazionale:

$$H(f) = \min_{\{f(\cdot)\}} \left( \sum_i (f(\mathbf{x}_i) - y_i)^2 + \lambda \Phi(f) \right)$$

$\lambda$  scelto ad esempio con il criterio di cross-validation.



## RBF come soluzione di un problema di regolarizzazione



Scrivo il funzionale come:

$$\Phi(f) = \int_{R_D} \frac{|\tilde{f}(s)|^2}{|\tilde{G}(s)|^2} ds$$

La soluzione generale è rappresentata da:

$$f(\mathbf{x}) = \sum_i c_i G(\mathbf{x} - \mathbf{x}_i) + \sum_k d_k Q(\mathbf{x})$$

Funzioni simmetriche  
(a base radiale)

Polinomi

$$\{c_i\} \quad \{d_k\} \rightarrow \sum_i c_i Q(\mathbf{x}_i) = 0$$



## Soluzione del problema di regolarizzazione



$$H(f) = \min_{\{f(\cdot)\}} \left( \sum_i (f(\mathbf{x}_i) - y_i)^2 + \lambda \Phi(f) \right) \quad f(\mathbf{x}) = \sum_j c_j G(\mathbf{x} - \mathbf{x}_j) + \sum_k d_k Q(\mathbf{x})$$

Solitamente si trascurano i  $\{d_k\}$ ; si devono determinare i  $\{c_i\}$ .

$$\mathbf{c} = \mathbf{G}^{-1} \mathbf{y}$$

$$\mathbf{c} = (\mathbf{G} + \lambda \mathbf{I})^{-1} \mathbf{y}$$

- Determinazione dei pesi di un perceptrone.
- Le unità sono posizionate sui punti.
- Nulla è detto sull'ampiezza.

→ GRBF → Problema di ottimizzazione.



## RBF Statistiche



$$f(\mathbf{x}) = \frac{\sum_i c_i G(\mathbf{x} - \mathbf{x}_i)}{\sum_i G(\mathbf{x} - \mathbf{x}_i)} \quad \text{Funzioni Gaussianhe normalizzate.}$$

$$H[f(\cdot)] = \int P(\mathbf{x}, y) [y - f(\mathbf{x})]^2 d\mathbf{x}dy \quad \text{Expected risk}$$

Non conosco  $P(\mathbf{x}, y)$ , posso:

- Stimarla dai dati.
- Considerare l' "Empirical risk":

$$H[f(\cdot)] = \int [y - f(\mathbf{x})]^2 d\mathbf{x}dy$$



## Parzen window estimator



$$P(\mathbf{z}) = \frac{1}{Nh} \sum_{i=1}^N \Phi\left(\frac{\mathbf{z} - \mathbf{z}_i}{h}\right) = \frac{1}{N\sigma} \sum_{i=1}^N K(\|\mathbf{x} - \mathbf{x}_i\|) K(y - y_i)$$

Parzen windows

$$H[f(\cdot)] = \int \frac{1}{N\sigma} K(\|\mathbf{x} - \mathbf{x}_i\|) K(y - y_i) [y - f(\mathbf{x})]^2 d\mathbf{x}dy$$

$$f(\mathbf{x}) = \frac{\sum_i y_i K(\|\mathbf{x} - \mathbf{x}_i\|)}{\sum_i K(\|\mathbf{x} - \mathbf{x}_i\|)} \quad \begin{array}{l} \text{Kernel regression} \\ \text{Nadaraya-Watson estimator} \end{array}$$





## L'approccio dell'analisi funzionale



$$s(\mathbf{x}) = \sum_{k=1}^M w_k g(\mathbf{x}; \mathbf{c}_k, \sigma_k) = \sum_{k=1}^M w_k \frac{e^{-\frac{\|\mathbf{x}-\mathbf{c}_k\|^2}{\sigma_k^2}}}{\sqrt{\pi^D \sigma_k^D}}$$

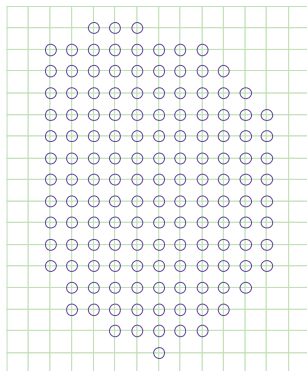
parameters to be configured:

$$M \in \mathbb{N}, \{w_k \in \mathbb{R}\}, \{\sigma_k \in \mathbb{R}\} \text{ e } \{\mathbf{c}_k \in \mathbb{R}^D\}$$

Ipotesi sottostante: unità uguali, equispaziate su un reticolo.



## RBF e Filtri



Gaussiane equispaziate su un grid.

$$|\mathbf{P}_{k+1} - \mathbf{P}_k| = \Delta \mathbf{P}_k$$

$$\sigma_k = \sigma \quad \forall k$$

Gaussiane normalizzate  $\|g(\cdot)\| = 1$

$$S(\mathbf{P}) = \sum_{k=1}^M w_k g(\mathbf{P} - \mathbf{P}_k | \sigma)$$

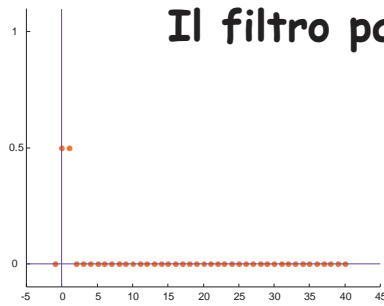
$g(\cdot)$  costituisce un filtro passa-basso.



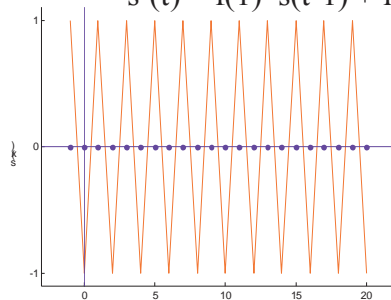
# Il filtro passa-basso



Moving average.



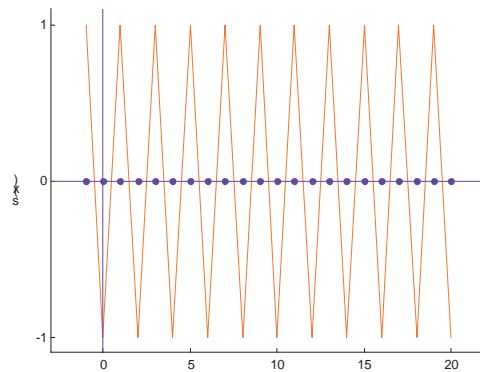
$$s'(t) = f(1)*s(t-1) + f(2) * s(t) = 0.5*s(t-1) + 0.5*s(t)$$



Estrae la componente continua (media mobile)



# Filtro passa-basso



$$s'(t) = f(1)*s(t-1) + f(2) * s(t) = 0.5*s(t-1) + 0.5*s(t) \Rightarrow$$

$$s'(t) = \sum_{k=0}^2 s(t-k) * f(k) \quad \boxed{s(t) * f(t)} \quad \text{Prodotto di convoluzione}$$



# Filtro-Gaussiano



$$s'(i) = \sum_{k=-N}^N s(k)g(i-k | \sigma) = s(.) * g(.)$$

E' un prodotto nel dominio delle frequenze:  $S'(\omega) = (S(\omega)*G(\omega))$

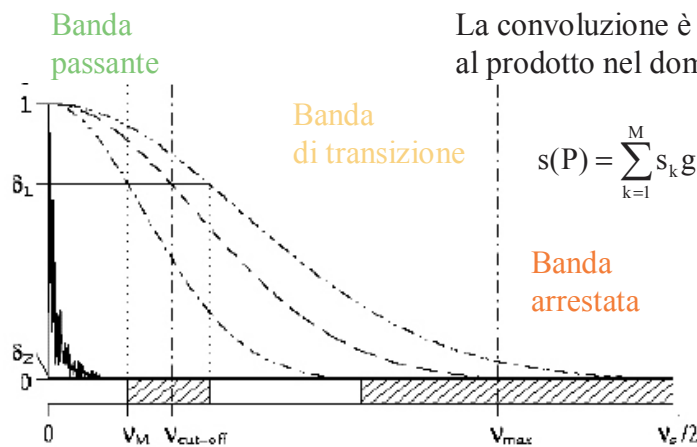
Cf. Kernel regression

Sostituisco ai pesi il valore della funzione campionata negli incroci della griglia di definizione.

$$w_k = s_k$$



# RBF come filtro -> criterio per $\sigma$



La convoluzione è equivalente al prodotto nel dominio delle

$$s(P) = \sum_{k=1}^M s_k g(P - P_k | \sigma)$$

Al diminuire di  $\sigma$ , aumenta la frequenza -> aumentare la frequenza di campionamento (vicinanza tra due Gaussiane): e.g.  $\sigma_1 = 1.465 \Delta P_1$  per  $|g(.)| < -40\text{dB}$  nella banda arrestata.

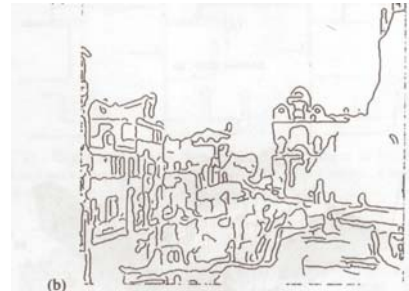


## Concetto di scala

Artificial Vision. Filter grids which operate at different scales.



(a) Small scale (high frequency)



(b) Large scale (low frequency)

Linear combination of Basis Functions:

$$s(P) = \sum_k^M s_k g(P - P_k | \sigma);$$

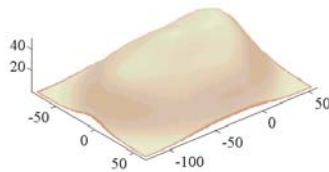


## Ricostruzione ad una certa scala



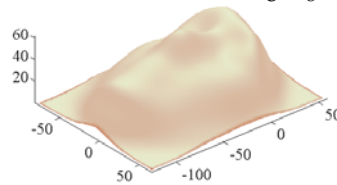
Approximation at layer #1

$\sigma = 16$



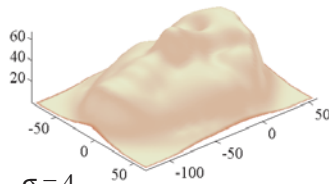
Approximation at layer #2

$\sigma = 8$



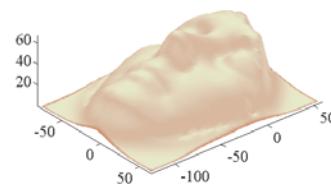
Approximation at layer #0

$\sigma = 4$



Approximation at layer #4

$\sigma = 2$





## RBF come filtro digitale: i problemi



$$s(P) = \sum_{k=1}^M s_k g(P - P_k | \sigma)$$

- 1) Non conosco  $\{S_k\}$ : i dati non sono equispaziati.
- 2) Non conosco il contenuto in frequenza dei dati (che guida la scelta di  $\sigma$ ).
- 3) Non ho un controllo dell'errore di ricostruzione.

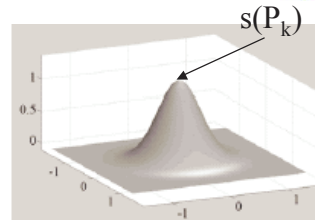


## 1) Dati non equispaziati



$$s(P) = \sum_{k=1}^M s_k g(P - P_k | \sigma)$$

Grigliatura dei dati.



$s(P_k)$  viene stimato attraverso una media pesata (con una funzione inversa della distanza dal centro) dei punti appartenenti al campo recettivo della Gaussiana:

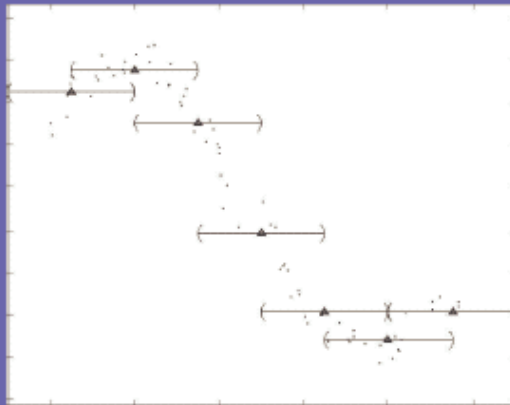
$$s(P_k) = \frac{\sum_m s(P_m) g(P_m - P_k | \sigma)}{\sum_m g(P_m - P_k | \sigma)} \quad P_m \in \text{RF}(P_k)$$

Operazioni quasi-locali => **Campo Recettivo**

Solo un sottoinsieme del data-set viene considerato.



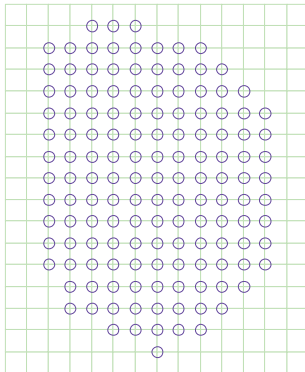
## Stima locale di $s(k)$



$$\bar{s}(c_k) = \frac{\sum_{x_r \in S_k} z_r \exp\left(-\frac{\|x_r - c_k\|^2}{\sigma_l^2}\right)}{\sum_{x_r \in S_k} \exp\left(-\frac{\|x_r - c_k\|^2}{\sigma_l^2}\right)}$$



## 2) Frequenza spaziale



$\Delta P$  controlla l'impaccamento delle gaussiane, e determina la massima frequenza ricostruibile (teorema di Shannon). Determina anche il valore di  $\sigma$ .

**Problemi se scelgo  $\sigma_{\min}$ :**

- Spreco di Gaussiane.
- Impaccamento troppo denso.
- Posso non avere punti all'interno del campo recettivo di una Gaussiana troppo stretta per stimare  $s_k$ .



## 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...).



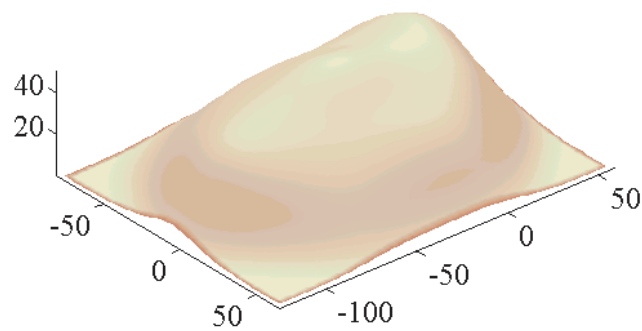
## Costruzione di un primo livello a scala ampia



$$\sigma_1 = 16$$

Approximation at layer #1

$$s(P) = \sum_{k=1}^M s_k g(P - P_k | \sigma_k)$$

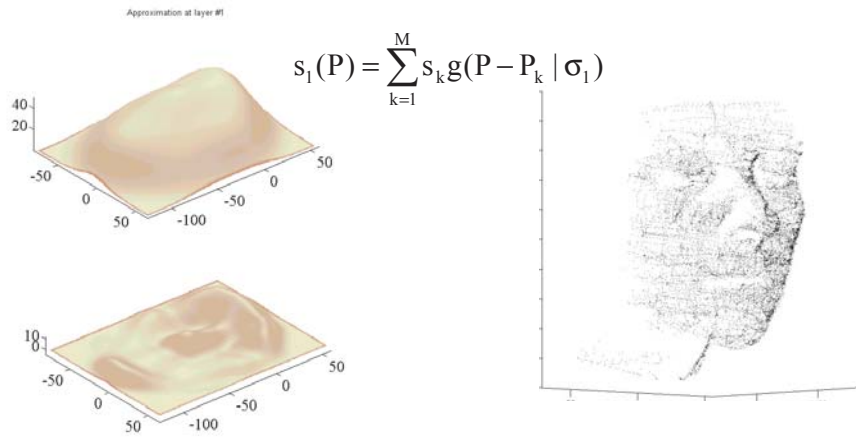




## Calcolo del residuo



Residuo della ricostruzione:  $r_1 = \{z_m - s_1(P)\} \forall P$  misurato,  $P_m$ .



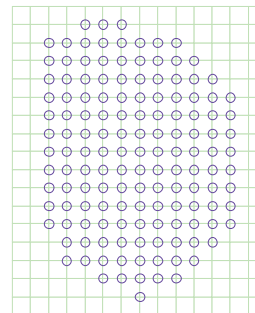
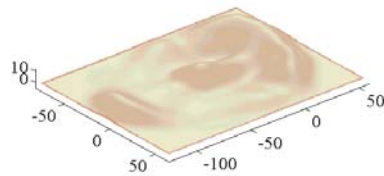
Residuo:  $\{z_m - s_1(P_m)\}$

Copyright N.A. Borghese Università di Milano 11/04/2003

<http://homes.dsi.unimi.it/~borghese> 31/45

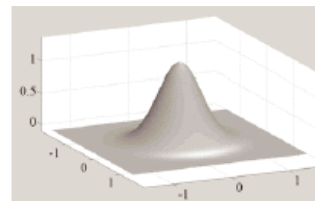


## Valutazione del residuo



Valutiamo il residuo  $\{r_1\}$  all'interno del campo recettivo di ogni Gaussiana dello strato come:

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



Copyright N.A. Borghese Università di Milano 11/04/2003

<http://homes.dsi.unimi.it/~borghese> 32/45





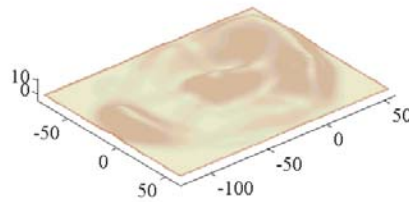
## Costruzione di un secondo livello gerarchico



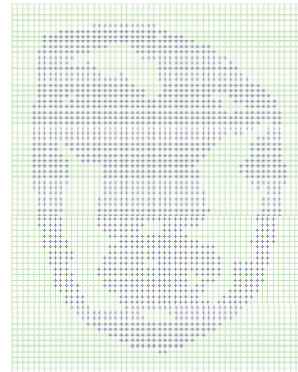
$$\text{Error}(P_k) = \frac{\sum_m |r_1(P_m) - S_1(P_m)|}{N_k} > \text{Criterio\_convergenza}$$

$\sigma = 8$

Output of layer #2



Layer #2

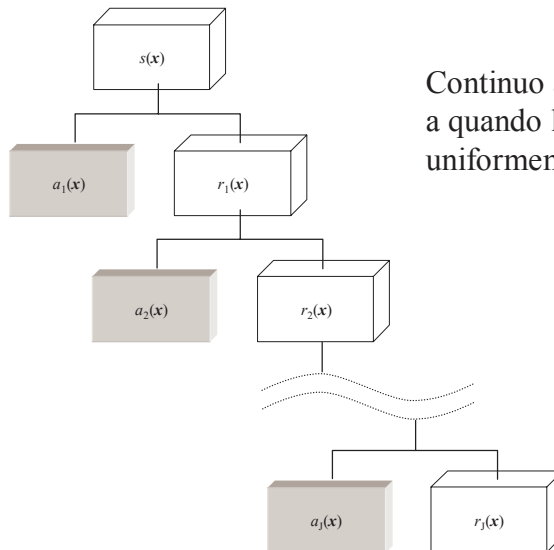


Copyright N.A. Borghese Università di Milano 11/04/2003

<http://homes.dsi.unimi.it/~borghese> 33/45



## Costruzione della gerarchia



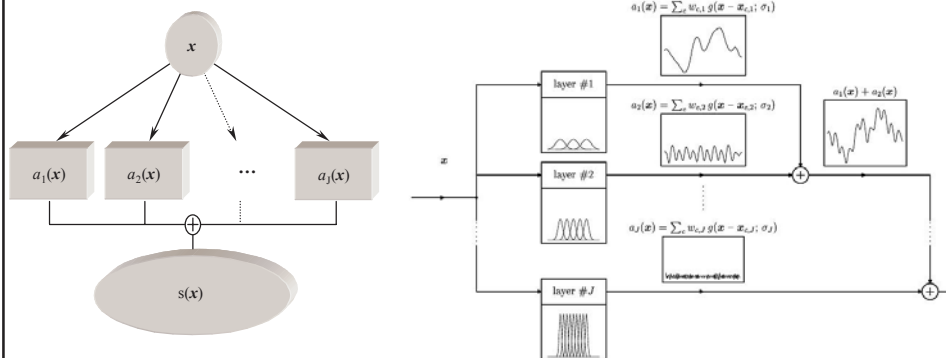
Continuo a costruire livelli fino a quando l'errore non è uniformemente sotto soglia.

Copyright N.A. Borghese Università di Milano 11/04/2003

<http://homes.dsi.unimi.it/~borghese> 34/45



## Sintesi della superficie



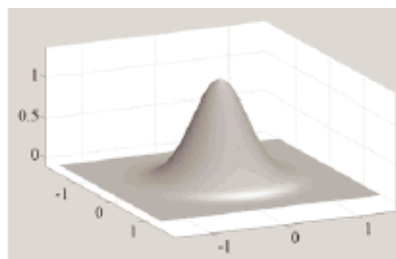
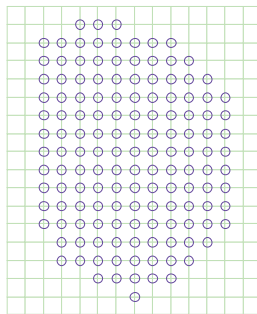
- Diversi strati (reti RBF).
- Ciascuno strato opera ad una certa scala.
- Gli strati non sono completi ma allocano Gaussiane solo dove l'errore locale è sopra-soglia.

Copyright N.A. Borghese Università di Milano 11/04/2003

<http://homes.dsi.unimi.it/~borghese> 35/45



## 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.

Copyright N.A. Borghese Università di Milano 11/04/2003

<http://homes.dsi.unimi.it/~borghese> 36/45



## Algoritmo di configurazione



A. Initialization:  $\{r_0(p_j) = z_j\}$  (residual = height of sampled points)

For each layer,  $l$

B. Configuration:

For each neuron,  $i$ , positioned in  $c_{i,l}$

B.1. retrieval of the points which belong to the receptive field of the neuron,  $S_{i,l}$

B.2. if the residual is larger than the measurement error, insert the neuron in  $c_{i,l}$

B.3. estimate of the surface height in the neuron center,  $\bar{x}_{i,l}$  and set to that value the weight  $w_{i,l}$

C. Compute the output of the layer,  $\{a_l(p_j)\}$

D. Compute the residual,  $\{r_l(p_j) = r_{l-1}(p_j) - a_l(p_j)\}$

Copyright N.A. Borghese Università di Milano 11/04/2003

<http://homes.dsi.unimi.it/~borghese> 37/45



## Implementazione veloce dell'algoritmo di configurazione



points array  
Input space partitioning data structure



- circular receptive field approximated by bounding square

- space partitioned in *cells* ( $\Delta x_l \times \Delta x_l$  squares)
- in-place partial sorting: points belonging to the same cell are adjacent in the points array
- store the position of the first point and the number of the points belonging to each cell
- points retrieval without distance calculation

Copyright N.A. Borghese Università di Milano 11/04/2003

<http://homes.dsi.unimi.it/~borghese> 38/45



## Basi funzionali



Base di Riesz:  $\left\| s(P_j) - \sum_k s(P_k) g(P_k - P_j; \sigma) \right\| < \varepsilon$

Tutte le funzioni appartenenti ad un sottospazio funzionale di L2 possono essere approssimate



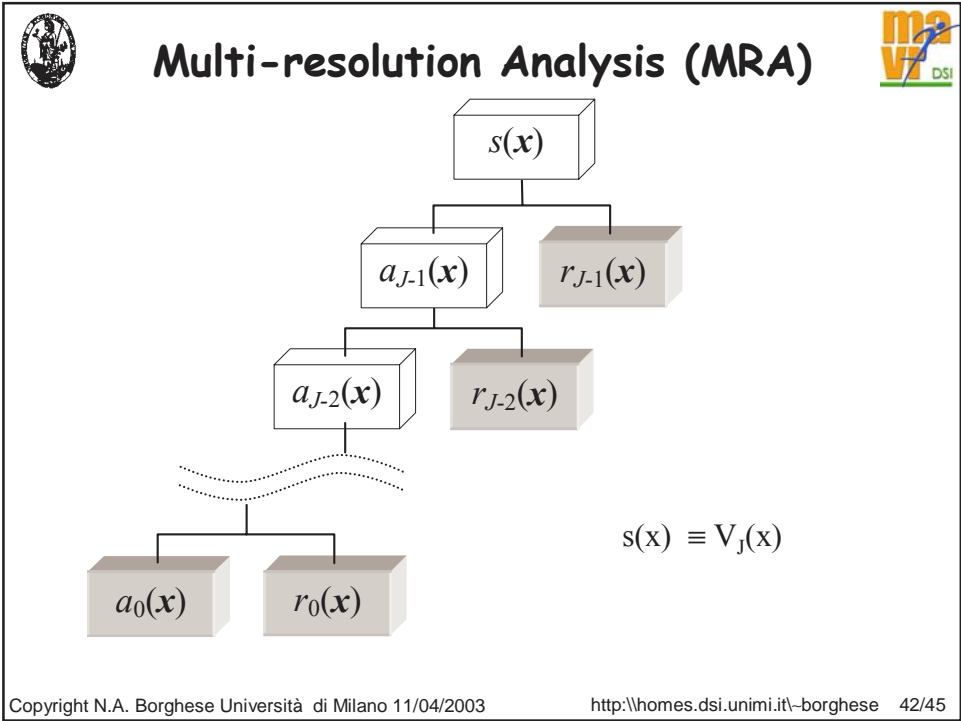
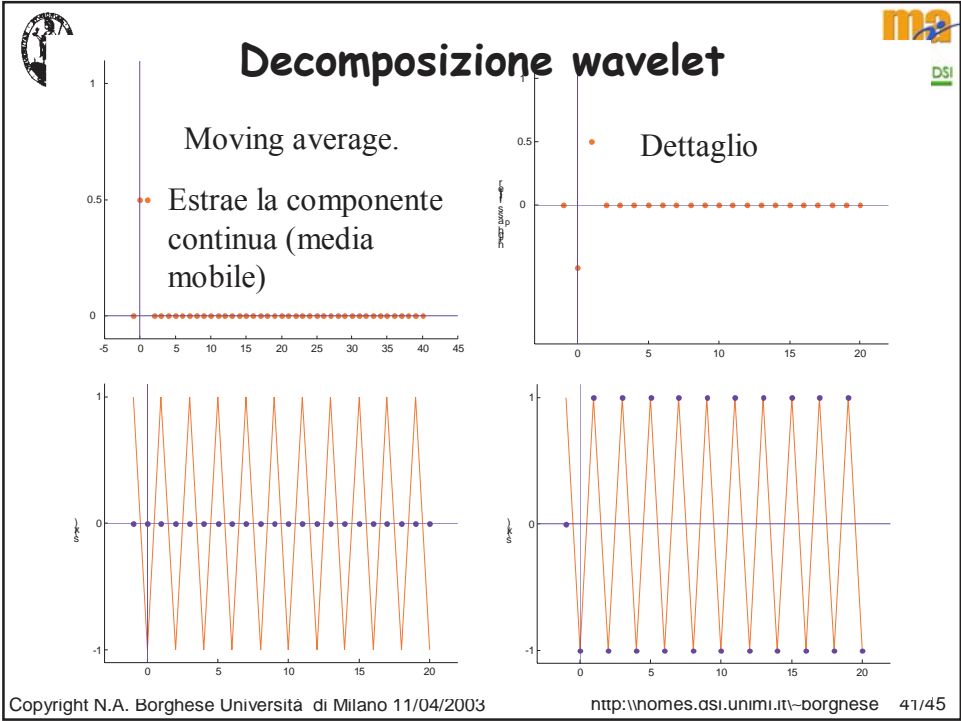
## Multi-resolution Analysis (MRA)



Una collezione di  $\{V_{j \in \mathbb{Z}}\}$  di sottospazi di L2 è una MRA sse:

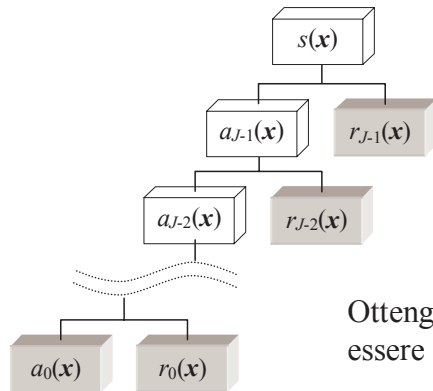
- $V_j \subset V_{j+1} \quad \forall j \in \mathbb{Z}$ .
- $\bigcap V_j = \emptyset$  e  $\bigcup V_j$  è densa in L2.
- Per tutti i j,  $s(\cdot) \in V_j$  sse  $s(2 \cdot) \in V_{j+1}$ .
- Esiste una *scaling function*,  $\phi(\cdot) \in V_0$  tale per cui  $\phi(\cdot - k)$  è una base di Riesz per  $V_0$ .

Gli spazi sono caratterizzati dalla scala di  $\phi(\cdot)$ . Scegliendo uno spazio con j sufficientemente elevato, si può ricostruire una funzione di L2 con l'approssimazione desiderata.





## Decomposizione



Proietto  $s^J$  su  $V_{J-1}$  coperto da:

$$\{2^{J-1} \phi(2^{J-1} \bullet -k)\}_k$$

Otengo un residuo. Il residuo può essere calcolato direttamente come:

$$r_j(\cdot) = \sum_k d_{k,j} 2^{j/2} \psi(2^j \bullet -k)$$

↑  
wavelet



## Decomposizione Wavelet



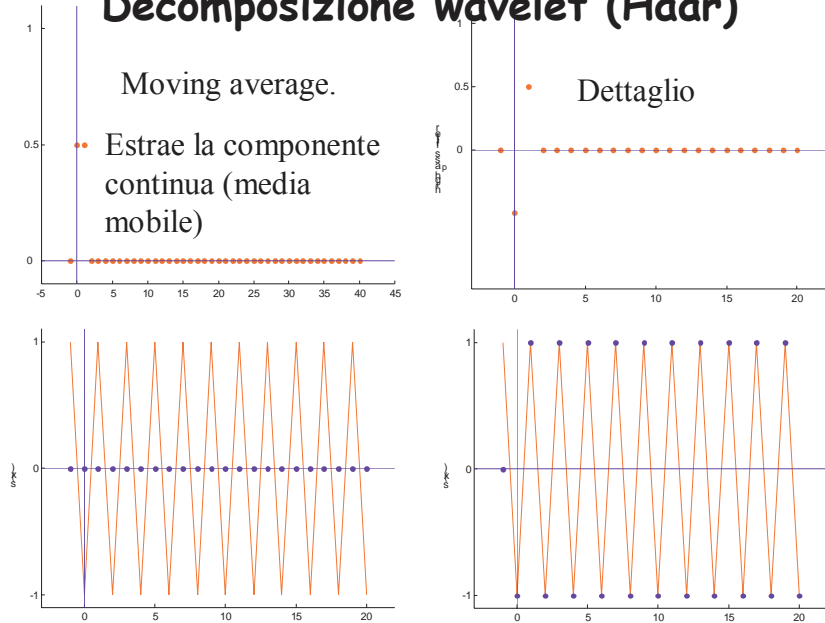
Decomposizione degli spazi funzionali:  $V_j = V_{j-1} \oplus W_{j-1}$ .

↑  
Sottospazio coperto dalle Wavelet

I coefficienti  $\{c\}$  e  $\{d\}$  si possono ottenere direttamente proiettando le funzioni del livello superiore su opportune funzioni di base.



# Decomposizione wavelet (Haar)



Copyright N.A. Borghese Università di Milano 11/04/2003

<http://wonomes.asi.unimi.it/~borgnese> 45/45