



# Aprendizaje Selectivo para Redes de Neuronas de Base Radial

**José María Valls Ferrán**  
Universidad Carlos III de Madrid

## *Índice*

- **Introducción**
- **Estado de la cuestión**
  - Redes de Neuronas de Base Radial
  - Métodos de aprendizaje retardado
- **Objetivos**
- **Método de Selección de Patrones por Vecindad Ponderada**
  - Ponderación Gaussiana
  - Ponderación Inversa
- **Conclusiones**
- .....

## Introducción

- **Introducción**
- Estado del arte
- Objetivos
- Método SPVP
  - P. Gaussiana
  - P. Inversa
- Conclusiones

- El **nivel de generalización** de una Red de Neuronas es la capacidad que ésta tiene para responder correctamente a entradas que no han sido utilizadas en el aprendizaje.
- **Las Redes de Neuronas de Base Radial (RNBR)** son aproximadores universales, como el perceptrón multicapa (también son ‘feedforward’).
  - **Ventaja:** Convergencia muy rápida
  - **Inconveniente:** Menor capacidad de generalización.

## Introducción

- **Introducción**
- Estado del arte
- Objetivos
- Método SPVP
  - P. Gaussiana
  - P. Inversa
- Conclusiones

- **Métodos de aprendizaje retardado** (o ‘perezoso’ o ‘lazy’).

Se construyen representaciones locales de la función objetivo cada vez que se recibe una muestra de test.

  - **Ventajas:** Buena capacidad de generalización.
  - **Inconvenientes:** Gran coste computacional, dependencia de varios parámetros
- **Objetivo:** Mejorar la capacidad de generalización de las RNBR utilizando técnicas de aprendizaje retardado.

## Redes de Neuronas de Base Radial (RNBR)

• Introducción  
 • Estado del arte  
   RNBR  
   A. Retardado  
 • Objetivos  
 • Método SPVP  
   P. Gaussiana  
   P. Inversa  
 • Conclusiones

- Las RNBR surgen como solución regularizada al problema de interpolación
- Interpolación exacta:

Dados un conjunto de N puntos  $\{\mathbf{x}_i \in R^p \mid i = 1, \dots, N\}$  y un conjunto de N números reales  $\{t_i \in R \mid i = 1, \dots, N\}$ , encontrar una función continua que cumpla  $y(\mathbf{x}_i) = t_i, \quad i = 1, \dots, N$

**Solución:** combinación lineal de N funciones de base radial

$$y(\mathbf{x}) = \sum_{i=1}^N w_i \phi(\|\mathbf{x} - \mathbf{x}_i\|)$$

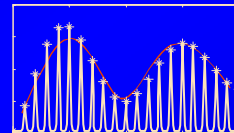
## Redes de Neuronas de Base Radial (RNBR)

• Introducción  
 • Estado del arte  
   RNBR  
   A. Retardado  
 • Objetivos  
 • Método SPVP  
   P. Gaussiana  
   P. Inversa  
 • Conclusiones

### • Aproximación de funciones

La interpolación exacta tiene inconvenientes:

- Deficiente capacidad de generalización
- Alto coste computacional



No se exigirá la interpolación exacta, sino sólo una aproximación a los puntos minimizando una función de coste

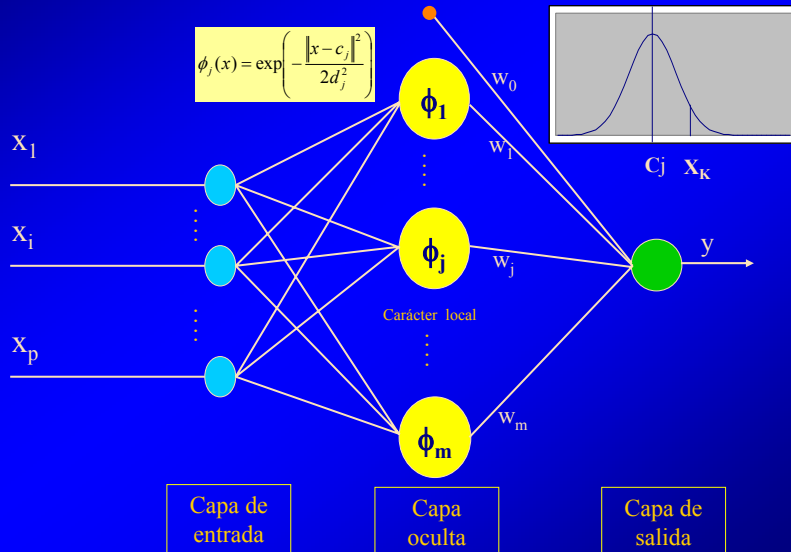
### Soluciones:

- Redes Regularizadas
- Reducción del número de funciones de base radial ( $m < N$ )

$$y(\mathbf{x}) = \sum_{i=1}^m w_i \phi_i(\mathbf{x}) + w_0$$

## Arquitectura de las RNBR

- Introducción
- Estado del arte RNBR
  - A. Retardado
- Objetivos
- Método SPVP
  - P. Gaussiana
  - P. Inversa
- Conclusiones



7

## Aprendizaje de las RNBR

- Introducción
- Estado del arte RNBR
  - A. Retardado
- Objetivos
- Método SPVP
  - P. Gaussiana
  - P. Inversa
- Conclusiones

- **Método totalmente supervisado**
  - Todos los parámetros se determinan de forma supervisada
- **Método Híbrido**
  - **Fase no supervisada**
    - Determinación de los centros de las neuronas ocultas
      - K-medias
      - SOM
    - Determinación de las desviaciones
  - **Fase supervisada**
    - Determinación de los pesos de las conexiones

Debido al carácter local de las RNBR y a la separación de las fases, el método híbrido es muy rápido

8

## Selección del modelo en RNBR

- Introducción
- Estado del arte  
RNBR
  - A. Retardado
- Objetivos
- Método SPVP
  - P. Gaussiana
  - P. Inversa
- Conclusiones

Problema de encontrar el número óptimo de neuronas ocultas para mejorar la capacidad de generalización.

- Métodos de crecimiento
- Métodos de poda
- Métodos mixtos

## RNBR ¿locales?

- Tradicionalmente se ha considerado que las RNBR son modelos **locales**, porque construyen aproximaciones centradas alrededor de cada neurona.
- Aunque las RNBR utilizan múltiples aproximaciones locales, son métodos 'eager' (aprendizaje temprano) porque construyen la hipótesis antes de que se conozca la query o patrón de test.
- Construyen una aproximación global basada solamente en los ejemplos de entrenamiento, que es una suma de múltiples aproximaciones locales.
- Las aproximaciones locales que se construyen no están centradas en la 'query' como pasa en los métodos lazy

## Métodos de aprendizaje retardado

- Introducción
- Estado del arte
  - RNBR
  - A. Retardado
- Objetivos
- Método SPVP
  - P. Gaussiana
  - P. Inversa
- Conclusiones

- **Métodos de aprendizaje retardado o 'perezoso'.** Se construyen representaciones locales de la función objetivo cada vez que se recibe una muestra de test.
  - Retrasan la generalización: cuando se dispone de patrón de test.
  - En lugar de estimar la función objetivo una sola vez para todo el espacio de instancias, se estima localmente y de manera diferente para cada nueva instancia.
  - Seleccionar datos relevantes, cada vez que se recibe una nueva instancia.
  - La relevancia de cada patrón se determina mediante una **función de la distancia**: puntos más cercanos, mayor importancia.

## Métodos de aprendizaje retardado

- Introducción
- Estado del arte
  - RNBR
  - A. Retardado
- Objetivos
- Método SPVP
  - P. Gaussiana
  - P. Inversa
- Conclusiones

### Algoritmo de k-vecinos.

Se seleccionan los k vecinos más próximos al patrón de test.

### Algoritmo de k-vecinos ponderado por distancia.

Se asigna un peso a cada vecino seleccionado en función de su distancia al patrón de test.

### Regresión local ponderada

Se construye una aproximación explícita a la función objetivo sobre una región local que rodea a la query.

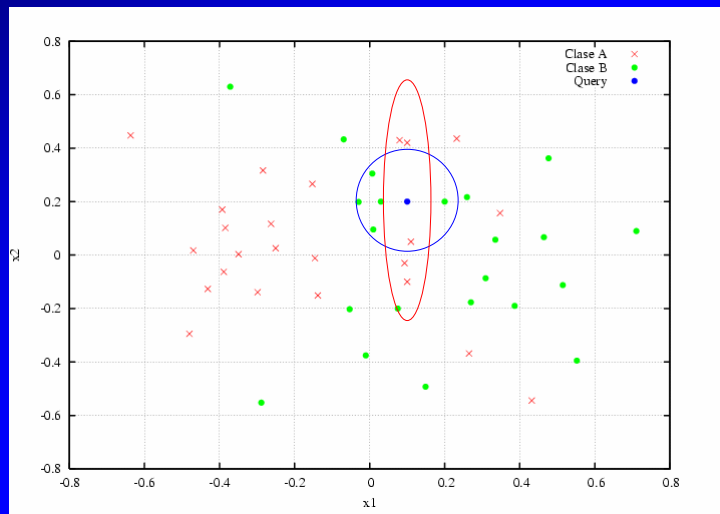
## Métodos de aprendizaje retardado

• Introducción  
 • Estado del arte  
   RNBR  
   A. Retardado  
 • Objetivos  
 • Método SPVP  
   P. Gaussiana  
   P. Inversa  
 • Conclusiones

- Buena capacidad de generalización
- Problemas con la función de distancia
- Problemas con la elección del parámetro  $k$
- Alto coste computacional

13

## K NN. Ejemplo con dos funciones de distancia



$K = 6$

Con función 1, Query clase B.

Con función 2, Query clase A.

14

## Resumen

• Introducción  
 • Estado del arte  
 • Objetivos  
 • Método SPVP  
   P. Gaussiana  
   P. Inversa  
 • Conclusiones

- Redes de neuronas de base radial
  - Ventajas: Aproximadores universales, rapidez en el entrenamiento
  - Inconvenientes: Deficiente capacidad de generalización.
- Métodos *de aprendizaje retardado*
  - Ventaja: Buena capacidad de generalización
  - Inconvenientes:
    - » Dependencia del parámetro  $k$
    - » Dependencia de la función de distancia
    - » Alto coste computacional

## Objetivos del trabajo

• Introducción  
 • Estado del arte  
 • Objetivos  
 • Método SPVP  
   P. Gaussiana  
   P. Inversa  
 • Conclusiones

- Mejorar la capacidad de generalización de las RNBR con un planteamiento próximo al de los métodos de aprendizaje retardado, aprovechando la rapidez de las RNBR.
- Realización de un método general de aprendizaje retardado aplicable independientemente del modelo de redes de neuronas elegido.
- Selección de los elementos del conjunto de aprendizaje que mejor se adapten a un determinado patrón de test.



## Solución Propuesta

• Introducción  
 • Estado del arte  
 • Objetivos  
 • Método SPVP  
   P. Gaussiana  
   P. Inversa  
 • Conclusiones

- Se propone un método de entrenamiento de las RNBR utilizando una estrategia *retardada*
  - Restringir el número de patrones de entrenamiento utilizados
  - Selección de patrones mediante similitud respecto a test
  - Entrenar la red sólo cuando se quiere generalizar
  - Descartar redes entrenadas anteriormente
  - Si la selección está bien hecha se reduce el ruido por patrones irrelevantes

17

## Método de Selección de Patrones mediante Vecindad Ponderada

• Introducción  
 • Estado del arte  
 • Objetivos  
 • Método SPVP  
   P. Gaussiana  
   P. Inversa  
 • Conclusiones

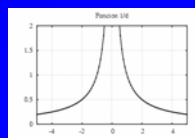
- Dos propuestas, utilizando diferentes funciones de ponderación

- Ponderación Gaussiana
- Ponderación Inversa

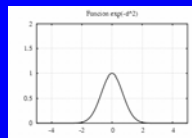
La función de ponderación asigna un peso a cada patrón de entrenamiento dependiendo de su distancia euclídea al patrón de test ( $q$ ).

- Requisitos de las funciones de ponderación

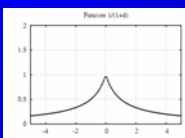
- Máximo de la función a distancia cero
- El valor de la función disminuye cuando la distancia aumenta.
- Valores no negativos



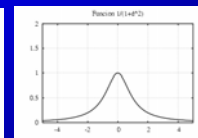
$$1/d$$



$$e^{-d^2}$$



$$1/(1+d)$$



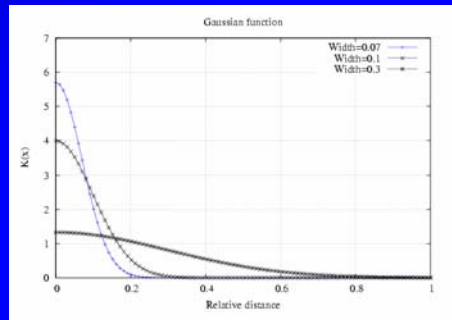
$$1/(1+d^2)$$

18

## Ponderación Gaussiana

- Introducción
- Estado del arte
- Objetivos
- Método SPVP
- P. Gaussiana
- P. Inversa
- Conclusiones

- La función de ponderación es la **función normal o Gaussiana**
- El valor de la función decrece con la distancia  $d(x_k, q)$
- Dependencia del parámetro sigma (desviación)



$$K(x_k) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{d(q, x_k)^2}{2\sigma^2}\right)$$

19

## PG. Descripción del método

- Introducción
- Estado del arte
- Objetivos
- Método SPVP
- P. Gaussiana
- P. Inversa
- Conclusiones

Sea  $q=(q_1, q_2, \dots, q_n)$  el patrón de test y  $X$  el conjunto de entrenamiento completo.

1. Se asocia un valor real  $d_k$  a cada patrón de entrenamiento  $x_k$

$$d_k = d(x_k, q) = \sqrt{\sum_{i=1}^n (x_{ki} - q_i)^2}$$

2. Se asocia a cada patrón  $x_k$  el valor resultante de aplicar al patrón la función de ponderación

$$K(x_k) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{d_k^2}{2\sigma^2}\right), \text{ para } k = 1, 2, \dots, N$$

3. Se calcula  $n_k$  : parte entera de  $K(x_k)$  que indicará cuántas veces el patrón  $x_k$  será repetido en el conjunto  $X_q$

20

## PG. Descripción del método

- Introducción
- Estado del arte
- Objetivos
- Método SPVP
- P. Gaussiana
- P. Inversa
- Conclusiones

4. Se construye el nuevo subconjunto  $X_q$  asociado al patrón de test  $q$ , dependiendo de los valores  $n_k$ .

Se entrena la RNBR para determinar:

- **Centros de las neuronas:**  
k-medias sobre el conjunto  $W_q$  (patrones sin repetir).
- **Desviaciones de las neuronas:**  
media geométrica de distancia a los dos centros más próximos.
- **Pesos:**  
de forma supervisada minimizando el error cuadrático medio sobre el conjunto de entrenamiento  $X_q$ .

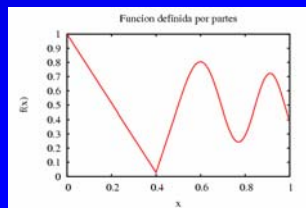
21

## PG. Dominios utilizados. Aproximación de funciones

### • Función definida por partes

$$f(x) = \begin{cases} -2.186x - 12.864 & \text{if } -10 \leq x < -2 \\ 4.246x & \text{if } -2 \leq x < 0 \\ 10e^{(-0.05x+0.5)} \sin[(0.03x+0.7)\pi] & \text{if } 0 \leq x \leq 10 \end{cases}$$

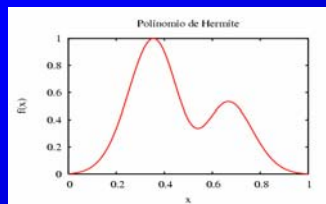
- Ciertos patrones son difíciles de generalizar



### • Polinomio de Hermite

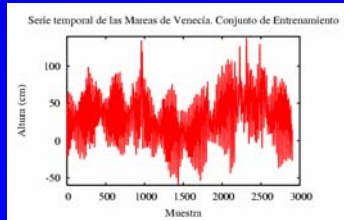
$$f(x) = 1.1 \left( 1 - x + 2x^2 \right) \exp \left( -\frac{1}{2} x^2 \right)$$

- Muy utilizado en la literatura

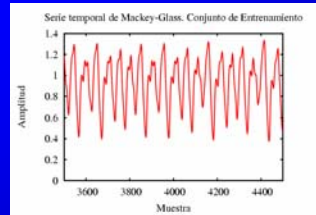


22

- **Serie temporal de las Mareas de Venecia**
  - Problema real
  - Situaciones inusuales de “agua alta”: Factores climáticos caóticos + Factores periódicos.
  - Pocos datos representan las situaciones de agua alta



- **Serie temporal de Mackey-Glass**
  - Serie temporal caótica generada artificialmente
  - Muy utilizada en la literatura

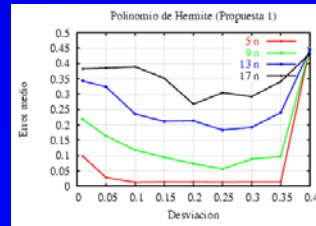


23

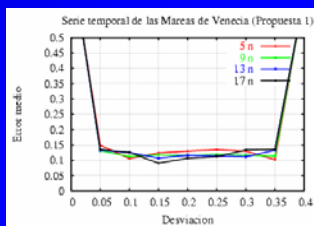
- **Función definida por partes**



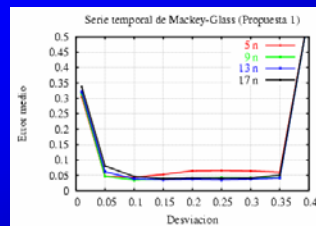
- **Polinomio de Hermite**



- **Serie de las Mareas de Venecia**



- **Serie de Mackey-Glass**



24

## PG. Resultados experimentales

- Introducción
- Estado del arte
- Objetivos
- Método SPVP
- P. Gaussiana
- P. Inversa
- Conclusiones

Comparación de los errores mínimos obtenidos con entrenamiento selectivo y con entrenamiento tradicional

	Ponderación Gaussiana	Entrenamiento Tradicional
Función definida por partes	0.0233 $\sigma = 0.25$ , 9 neuronas	0.04156 100 neuronas
Polinomio de Hermite	0.0121 $\sigma = 0.1$ , 5 neuronas	0.01904 40 neuronas
Serie de las Mareas de Venecia	0.0905 $\sigma = 0.15$ , 17 neuronas	0.09605 50 neuronas
Serie de Mackey-Glass	0.0348 $\sigma = 0.1$ , 9 neuronas	0.10273 110 neuronas

25

## Ponderación Gaussiana. Conclusiones

- Introducción
- Estado del arte
- Objetivos
- Método SPVP
- P. Gaussiana
- P. Inversa
- Conclusiones

- Se mejoran los resultados obtenidos con entrenamiento tradicional de las RNBR.
- El error depende mucho de la desviación.
  - Sería necesario determinar el valor adecuado de  $\sigma$ .
  - Puede ocurrir que con desviaciones muy pequeñas o muy grandes no se seleccione ningún patrón.

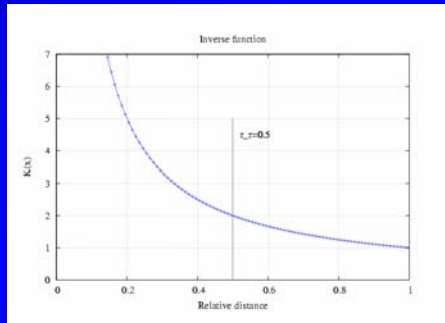
26

## Ponderación Inversa

- Introducción
- Estado del arte
- Objetivos
- Método SPVP
  - P. Gaussiana
  - P. Inversa**
- Conclusiones

- **Idea principal:** Modificar el método utilizando otra función de ponderación que no dependa de un parámetro:  
**Función Inversa**

$$K(x_k) = \frac{1}{d(q, x_k)}$$



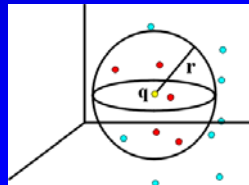
Los patrones más cercanos se van a incluir siempre en el conjunto  $X_q$

27

## Ponderación Inversa

- Introducción
- Estado del arte
- Objetivos
- Método SPVP
  - P. Gaussiana
  - P. Inversa**
  - PI-1
  - PI-2
- Conclusiones

- Se introduce un parámetro llamado **corte (r)**.



- Se pondera la influencia de los patrones seleccionados en K-medias
- Para evitar que queden clases vacías, se modifica la inicialización de los centros de K-medias.

28

## PI-1. Descripción del método

•Introducción  
•Estado del arte  
•Objetivos  
•Método SPVP  
  P. Gaussiana  
  **P. Inversa**  
    **PI-1**  
    PI-2  
•Conclusiones

1. Se establece el parámetro corte relativo ( $r_r \in [0,1]$ )
2. Se asocia un valor real  $d_k$  a cada patrón de entrenamiento  $x_k$
3. Para cada patrón  $x_k$  se calcula la distancia relativa  $d_{rk}$

$$d_{rk} = \frac{d_k}{d_{\max}}, \quad d_{\max} = \text{Max}(d_1, d_2, \dots, d_N)$$

4. Se asocia a cada patrón  $x_k$  el valor de la función inversa de la distancia  $d_k$

$$K(x_k) = \frac{1}{d_{rk}}, \text{ para } k = 1, 2, \dots, N$$

5. Se introduce un factor de normalización  $V$  para obtener los valores  $f_{nk}$

$$V = \frac{N}{\sum_{k=1}^N K(x_k)}, \text{ así } \sum_{k=1}^N f_{nk} = N$$

29

## PI-1. Descripción del método

•Introducción  
•Estado del arte  
•Objetivos  
•Método SPVP  
  P. Gaussiana  
  **P. Inversa**  
    **PI-1**  
    PI-2  
•Conclusiones

6. Los valores  $f_{nk}$  se utilizarán para calcular los valores  $n_k$ , que indicarán cuántas veces se incluirá  $x_k$  en  $X_q$

$$\begin{aligned} \text{si } d_{rk} < r_r \text{ entonces} \\ n_k &= \text{int}(f_{nk}) + 1 \\ \text{si no} \\ n_k &= 0 \end{aligned}$$

7. Se construye el nuevo subconjunto  $X_q$  asociado al patrón de test  $q$ .

30

## PI-1. Descripción del método

•Introducción  
•Estado del arte  
•Objetivos  
•Método SPVP  
  P. Gaussiana  
  P. Inversa  
    PI-1  
    PI-2  
•Conclusiones

Se entrena la RNBR. Para determinar los centros de las neuronas aplicando K-medias, se hace lo siguiente:

- El espacio de entrada al que se aplicará K-medias será el formado por los patrones del conjunto  $X_q$ .
- Se modifica la inicialización aleatoria de los centros, del siguiente modo
  1. Se calcula  $M_q$ , el centroide del conjunto de entrenamiento  $X_q$
  2. Se generan aleatoriamente  $k$  centros ( $c_{1q}, c_{2q}, \dots, c_{kq}$ ) tales que:

$$\|c_{jq} - M_q\| < \varepsilon, \text{ para } j = 1, 2, \dots, k \quad \text{siendo } \varepsilon \text{ un valor real muy pequeño.}$$

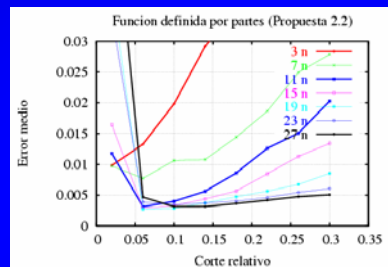
La determinación de las desviaciones y de los pesos se hace del mismo modo que en las propuestas anteriores.

31

## PI-1. Resultados experimentales

### • Función definida por partes

- Con un número grande de neuronas el error se mantiene casi constante con el corte, a partir de  $r_c=0.06$ .
- Cuando el número de neuronas es pequeño, el error crece rápidamente con el corte.
- Con cortes muy pequeños, el error siempre es grande.



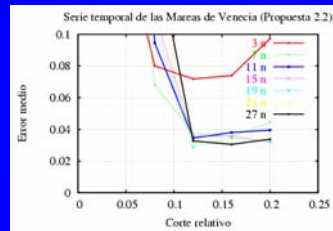
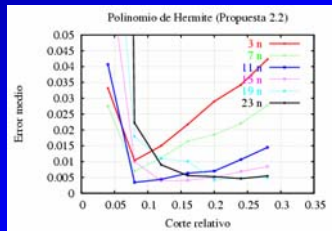
Neuronas Ocultas							
Corte	3	7	11	15	19	23	27
0.02	0.00984	0.00975	0.01178	0.01643	0.03824	0.03424	0.05668
0.06	0.01328	0.00774	0.00315	0.00289	0.00273	0.00389	0.00469
0.1	0.01983	0.01063	0.00401	0.00332	0.00285	0.00348	0.00315
0.14	0.02918	0.01079	0.00563	0.00440	0.00374	0.00373	0.00316
0.18	0.03532	0.01441	0.00861	0.00567	0.00474	0.00405	0.00366
0.22	0.04075	0.01865	0.01262	0.00848	0.00565	0.00461	0.00420
0.26	0.04602	0.02492	0.01506	0.01130	0.00675	0.00545	0.00477
0.3	0.05072	0.02787	0.02028	0.01340	0.00855	0.00607	0.00508

32

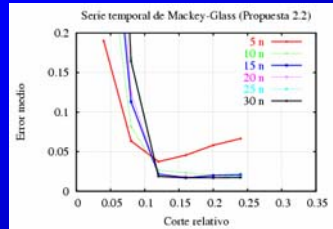


## PI-1. Resultados experimentales

- Polinomio de Hermite, series de Mareas de Venecia y Mackey-Glass



- Se observa una tendencia similar
- Errores grandes con cortes pequeños
- Con suficiente número de neuronas el error casi no depende del corte



33

## PI-1. Resultados experimentales

Comparación de los errores mínimos obtenidos

	Ponderación Inversa - 1	Ponderación Gaussiana	Entr. Tradic.
Función definida por partes	<b>0.00273</b> $r_r = 0.06, 19 \text{ n}$	0.0233 $\sigma = 0.25, 9 \text{ n}$	0.04156 100 n
Polinomio de Hermite	<b>0.00345</b> $r_r = 0.08, 9 \text{ n}$	0.0121 $\sigma = 0.1, 5 \text{ n}$	0.01904 40 n
Serie de las Mareas de Venecia	<b>0.02888</b> $r_r = 0.12, 19 \text{ n}$	0.0905 $\sigma = 0.15, 17 \text{ n}$	0.09605 50 n
Serie de Mackey-Glass	<b>0.01651</b> $r_r = 0.15, 25 \text{ n}$	0.0348 $\sigma = 0.1, 9 \text{ n}$	0.10273 110 n

34

## PI-1. Conclusiones

•Introducción  
•Estado del arte  
•Objetivos  
•Método SPVP  
  P. Gaussiana  
  P. Inversa  
    PI-1  
    PI-2  
•Conclusiones

- En todos los dominios se mejoran los resultados anteriores.
- El corte no es un parámetro crítico. Tampoco lo es el número de neuronas.

A partir de un valor suficiente del corte para que se seleccione un número mínimo de patrones, el error no varía sustancialmente, siempre que la red tenga un número de neuronas suficiente.

- Se detectan dos problemas
  - Los resultados dependen de las inicializaciones aleatorias de K-medias.
  - En espacios con dimensionalidad alta y baja densidad de datos puede ocurrir que no se seleccione ningún patrón

## Ponderación Inversa con inicialización determinista de K-medias y tratamiento de patrones nulos (PI-2)

•Introducción  
•Estado del arte  
•Objetivos  
•Método SPVP  
  P. Gaussiana  
  P. Inversa  
    PI-1  
    PI-2  
•Conclusiones

- Se propone una **inicialización determinista** de los centros:
  - Sean  $(x_1, x_2, \dots, x_l)$  los patrones seleccionados para responder al patrón  $q$ , ordenados de forma decreciente por frecuencias. Sea  $m$  el número de neuronas.
  - Si  $m \leq l$ , entonces el centro de la neurona  $i$  se inicializará a la posición de  $x_i$  para  $i=1, 2, \dots, m$
  - Si  $m > l$ , las  $l$  primeras neuronas se inicializarán según el punto anterior, y las restantes según el método aleatorio descrito en la propuesta anterior.

## Tratamiento de patrones nulos

Si para un patrón de test  $q$  el conjunto de patrones seleccionados  $X_q$  está vacío, se proponen las siguientes alternativas:

- Introducción
- Estado del arte
- Objetivos
- Método SPVP
  - P. Gaussiana
  - P. Inversa
  - PI-1
  - PI-2
- Conclusiones

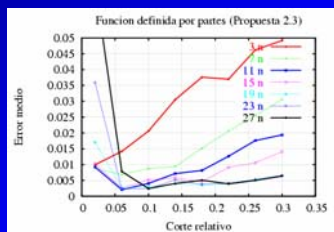
- **Alternativa 1:** Se aplica el método de selección de patrones al patrón de entrenamiento más cercano, como si fuera el patrón de test.
- **Alternativa 2:** Se entrena la red por el método tradicional, utilizando todos los patrones sin replicar.

37

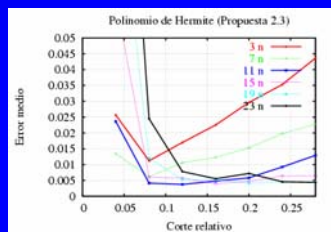
## PI-2. Resultados experimentales

- Introducción
- Estado del arte
- Objetivos
- Método SPVP
  - P. Gaussiana
  - P. Inversa
  - PI-1
  - PI-2
- Conclusiones

### • Función definida por partes



### • Polinomio de Hermite



- Los resultados son similares a los de la propuesta anterior sin depender de inicializaciones aleatorias.
- Con todos los cortes estudiados se seleccionan patrones de entrenamiento. No hace falta utilizar las alternativas de tratamiento de patrones nulos.

38

## PI-2. Resultados experimentales

- Introducción
- Estado del arte
- Objetivos
- Método SPVP
  - P. Gaussiana
  - P. Inversa
  - PI-1
  - PI-2
- Conclusiones

### •Mareas de Venecia

### •Mackey-Glass

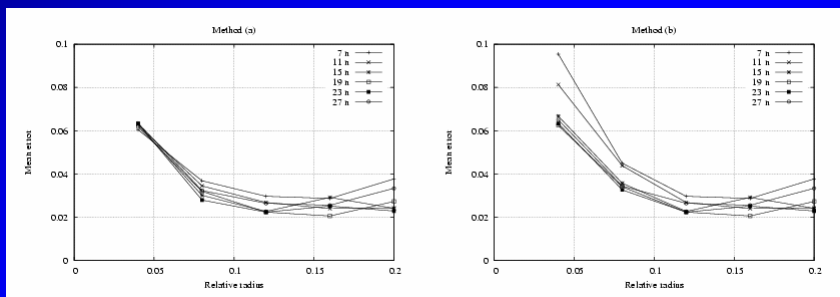
•Dominios de alta dimensionalidad:  
aparecen “patrones nulos”

•Los resultados, en zonas sin “nulos”  
son similares a los del método anterior,  
pero sin necesidad de inicializaciones  
aleatorias.

Corte	Nulos Venecia	Nulos MacKey
0.04	14	45
0.08	2	0

## PI-2. Resultados experimentales

### •Mareas de Venecia

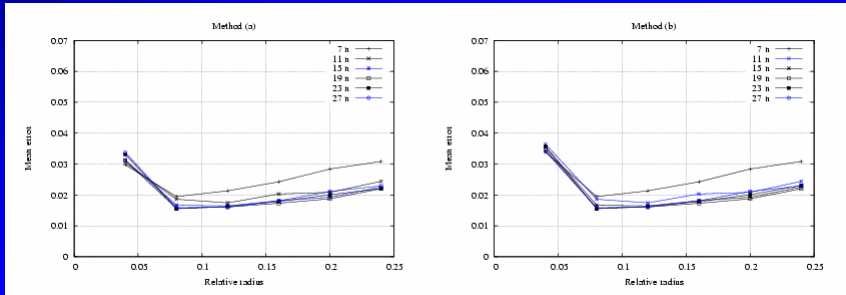


Alternativa 1  
Local

Alternativa 2  
Global

## PI-2. Resultados experimentales

### •Mackey-Glass



Alternativa 1  
Local

Alternativa 2  
Global

41

## PI-2. Resultados experimentales

### Zona de patrones nulos

### •Mareas de Venecia

Mean errors with deterministic initialization of centers and null patterns processing. Venice Lagoon time series

$r_r$	Meth	Hidden Neurons						NP	%PP
		7	11	15	19	23	27		
0.04	(a)	0.06042	0.06276	0.06292	0.06186	0.06330	0.06352	14	100
0.04	(b)	0.09542	0.08128	0.06672	0.06239	0.06333	0.06500	14	100
0.08	(a)	0.03685	0.03447	0.03011	0.03197	0.02792	0.03231	2	100
0.08	(b)	0.04497	0.04382	0.03572	0.03407	0.03266	0.03441	2	100

### •Mackey-Glass

Mean errors with deterministic initialization of centers and null patterns processing ( $r_r = 0.04$ ). Mackey-Glass time series.

	Hidden Neurons						NP	%PP
	7	11	15	19	23	27		
Method (a)	0.02974	0.03043	0.03132	0.03114	0.03309	0.03373	45	100
Method (b)	0.03385	0.03641	0.03545	0.03464	0.03568	0.03408	45	100

42

## PI-2. Resultados experimentales

Comparación de los errores mínimos obtenidos

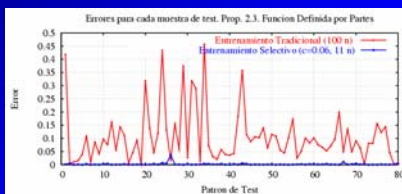
	P. Inversa – 2	P. Inversa – 1	P. Gauss.	Entr. Tradic.
<b>Función definida por partes</b>	<b>0.00208</b> $r_r=0.06, 11\ n$	0.00273 $r_r=0.06, 19\ n$	0.0233 $\sigma = 0.25, 9\ n$	0.04156 100 n
<b>Polinomio de Hermite</b>	<b>0.0038</b> $r_r=0.12, 11\ n$	0.00345 $r_r=0.08, 9\ n$	0.0121 $\sigma = 0.1, 5\ n$	0.01904 40 n
<b>Serie de las Mareas de Venecia</b>	<b>0.02059</b> $r_r=0.16, 19\ n$	0.02888 $r_r=0.12, 19\ n$	0.0905 $\sigma = 0.15, 17\ n$	0.09605 50 n
<b>Serie de Mackey-Glass</b>	<b>0.01554</b> $r_r=0.08, 30\ n$	0.01651 $r_r=0.15, 25\ n$	0.0348 $\sigma = 0.1, 9\ n$	0.10273 110 n

43

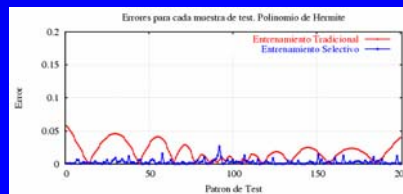
## Comparación de los errores por cada patrón de test

Comparación de los errores para cada patrón de test

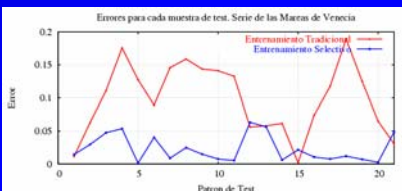
- Función definida por partes



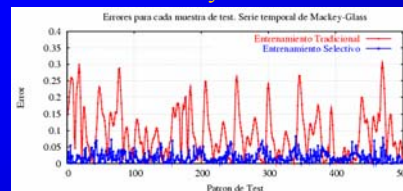
- Polinomio de Hermite



- Mareas de Venecia



- Serie de Mackey-Glass



44

## Conclusiones

•Introducción  
•Estado del arte  
•Objetivos  
•Método SPVP  
  P. Gaussiana  
  P. Inversa  
•Conclusiones y  
trabajos futuros

### Resumen de las conclusiones expuestas

- La capacidad de generalización de las RNBR aumenta significativamente respecto al entrenamiento tradicional.
- La función inversa utilizada logra buenos resultados sin ser el corte un parámetro crítico: Método robusto.
  - El método no depende de inicializaciones aleatorias
  - Se garantiza la predicción del 100% de los patrones
- El método puede aplicarse a otros modelos de redes de neuronas.

## Líneas futuras de investigación

•Introducción  
•Estado del arte  
•Objetivos  
•Método SPVP  
  P. Gaussiana  
  P. Inversa  
•Conclusiones y  
trabajos futuros

- Estudio de diferentes funciones de ponderación
- Estudio de diferentes funciones de distancia  
(La distancia se usa tanto en la activación de las neuronas como en la selección de los patrones)
- Validación del método propuesto en otros modelos de redes de neuronas, especialmente el perceptron multicapa.
- Búsqueda automática de la función óptima para un cierto problema