UFMG

INFERÊNCIA NEBULOSA E SISTEMAS NEBULOSOS ADAPTATIVOS: ANFIS

Prof. Walmir Matos Caminhas - DELT/UFMG

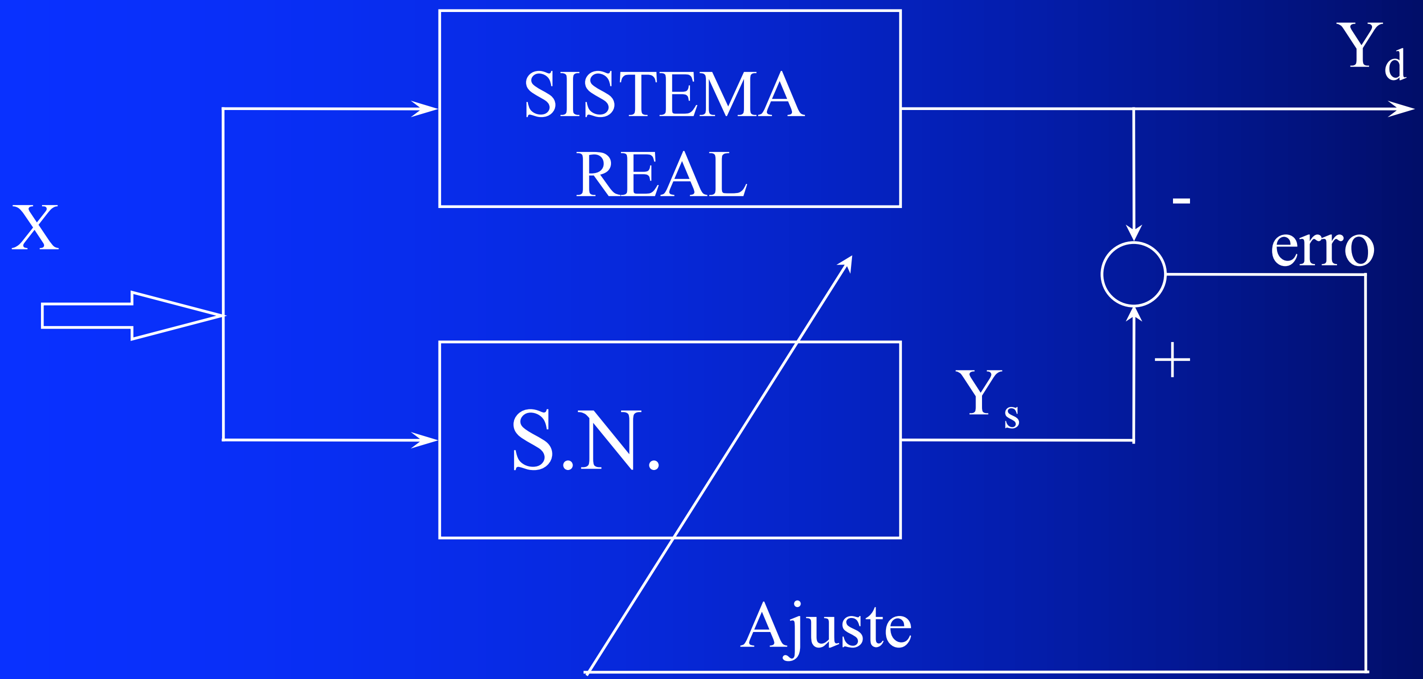
MODELAGEM VIA SISTEMAS NEBULOSOS

- PRINCIPAIS DIFICULDADE:
  - GERAÇÃO DAS REGRAS;
  - GERAÇÃO DAS FUNÇÕES DE PERTINÊNCIA;
  - ESCOLHA DOS OPERADORES.

- COMO SOLUCIONAR?
  - SISTEMAS ADAPTATIVOS

ANFIS: Sistema de Inferência Neuro-Fuzzy Adaptativo.

SISTEMAS NEBULOSOS ADAPTATIVOS



PARÂMETROS DE AJUSTE

- Funções de Pertinência
- Parâmetros dos Modelos de Saída
- Base de Regras (grau de certeza)

Regras implementadas

Se (x_1 é A_{1j}) e (x_2 é A_{2j})..e (x_i é A_{ij})..e (x_n é A_{nj})

Então $y_j = P_{1j}x_1 + P_{2j}x_2 + \ldots + P_{ij}x_i + P_{nj}x_n + q_j$

$$y_j = \sum_{i=1}^{n} P_{ij} . x_i + q_j$$

$$w_j = \prod_{i=1}^{n} \mu_{Aij}(x_i)$$

$$w_j = \mu_{A1j}(x_1).\ \mu_{A2j}(x_2).\ \ldots\ .\mu_{Aij}(xi).\ \mu_{Anj}(x_n)$$

$$\mu_{Aij}(x_i) = \exp\left[\frac{-1}{2}\left(\frac{x_i - c_{ij}}{\sigma_{ij}}\right)^2\right]$$

$$ys = \frac{\sum_{j=1}^{m} w_j \cdot y_j}{\sum_{j=1}^{m} w_j} = \frac{a}{b}$$

$$\min\ e = \frac{1}{2}(y_s - y_d)^2$$

$$e = f(x, c_{ij}, \sigma_{ij}, p_{ij}, q_j)$$

Método do Gradiente para atualização dos parâmetros

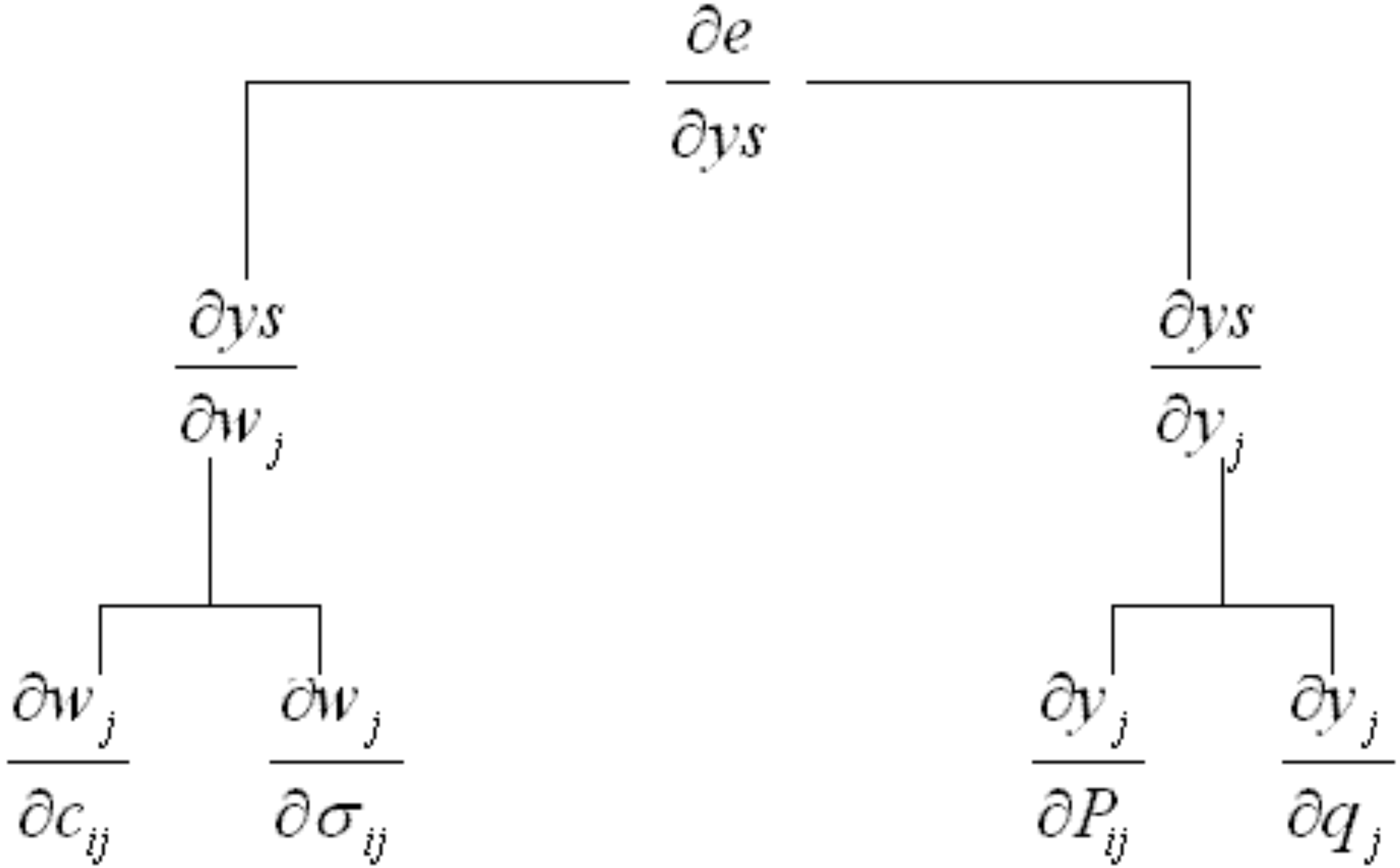
$$c_{ij}^{\,k+1} = c_{ij}^{\,k} - \alpha.\frac{\partial e}{\partial c_{ij}}\Big|k$$

$$\sigma_{ij}^{\,k+1} = \sigma_{ij}^{\,k} - \alpha.\frac{\partial e}{\partial \sigma_{ij}}\Big|k$$

$$P_{ij}^{\,k+1} = P_{ij}^{\,k} - \alpha.\frac{\partial e}{\partial P_{ij}}\Big|k$$

$$q_{j}^{\,k+1} = q_{j}^{\,k} - \alpha.\frac{\partial e}{\partial q_{j}}\Big|k$$

Obtenção das Derivadas



Derivadas Parciais

$$\frac{\partial e}{\partial c_{ij}} = \frac{\partial e}{\partial ys}\frac{\partial ys}{\partial w_j}\frac{\partial w_j}{\partial c_{ij}}$$

$$\frac{\partial e}{\partial ys} = (ys - yd)$$

$$\frac{\partial ys}{\partial w_j} = \left(\frac{y_j - ys}{b}\right)$$

$$\frac{\partial w_j}{\partial c_{ij}} = w_j\left[\frac{(x_i - c_{ij})}{\sigma_{ij^2}}\right]$$

Derivadas Parciais

$$\frac{\partial e}{\partial \sigma_{ij}} = \frac{\partial e}{\partial ys}\frac{\partial ys}{\partial w_j}\frac{\partial w_j}{\partial \sigma_{ij}}$$

$$\frac{\partial e}{\partial ys} = (ys - yd)$$

$$\frac{\partial ys}{\partial w_j} = \left(\frac{y_j - ys}{b}\right)$$

$$\frac{\partial w_j}{\partial \sigma_{ij}} = w_j\left[\frac{(x_i - c_{ij})^2}{\sigma_{ij^3}}\right]$$

Derivadas Parciais

$$\frac{\partial e}{\partial P_{ij}} = \frac{\partial e}{\partial ys}\frac{\partial ys}{\partial y_j}\frac{\partial y_j}{\partial P_{ij}}$$

$$\frac{\partial e}{\partial ys} = (ys - yd)$$

$$\frac{\partial ys}{\partial y_j} = \frac{w_j}{b}$$

$$\frac{\partial y_j}{\partial P_{ij}} = x_i$$

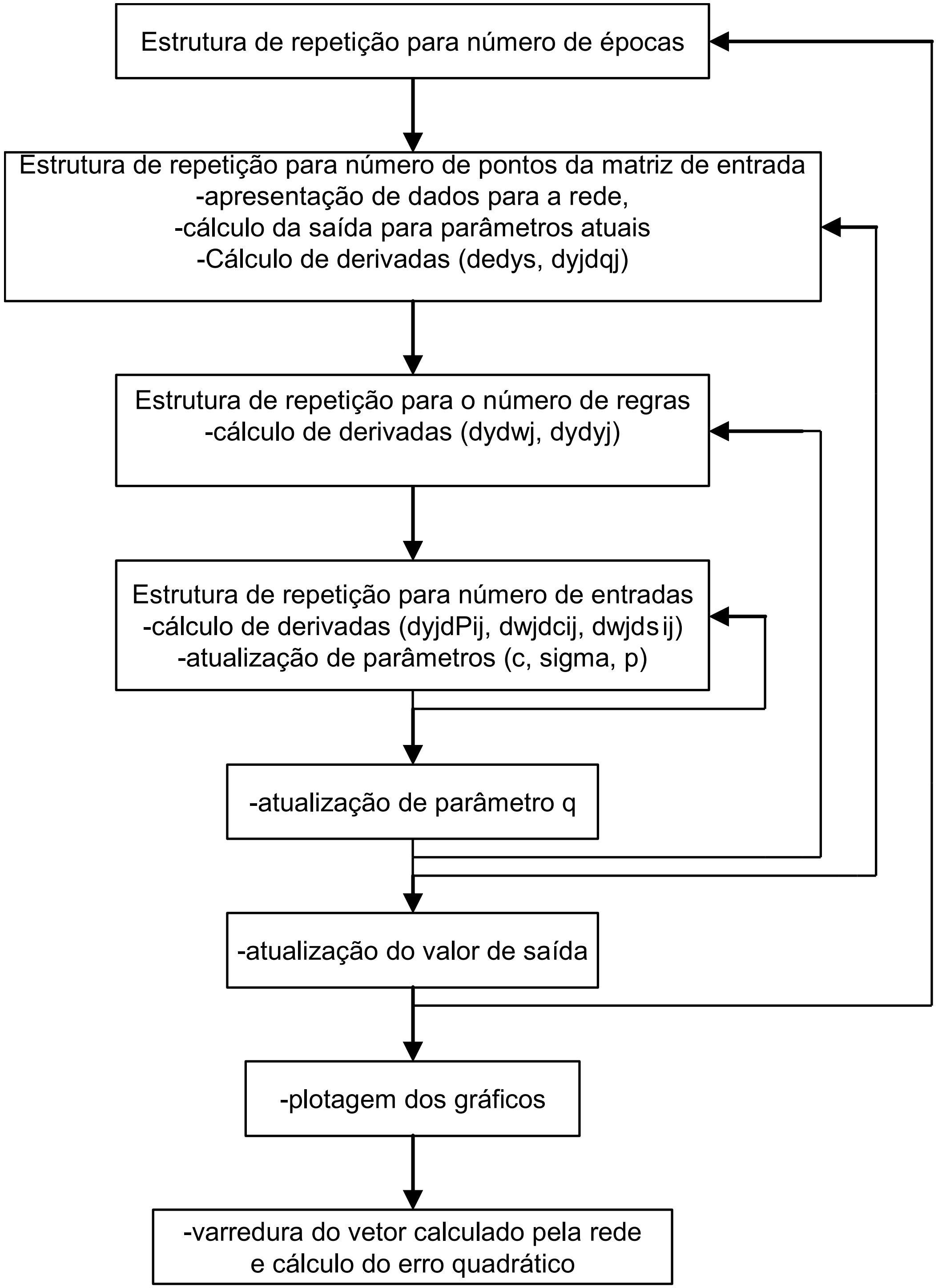
Derivadas Parciais

$$\frac{\partial e}{\partial q_j} = \frac{\partial e}{\partial ys}\frac{\partial ys}{\partial y_j}\frac{\partial y_j}{\partial q_j}$$

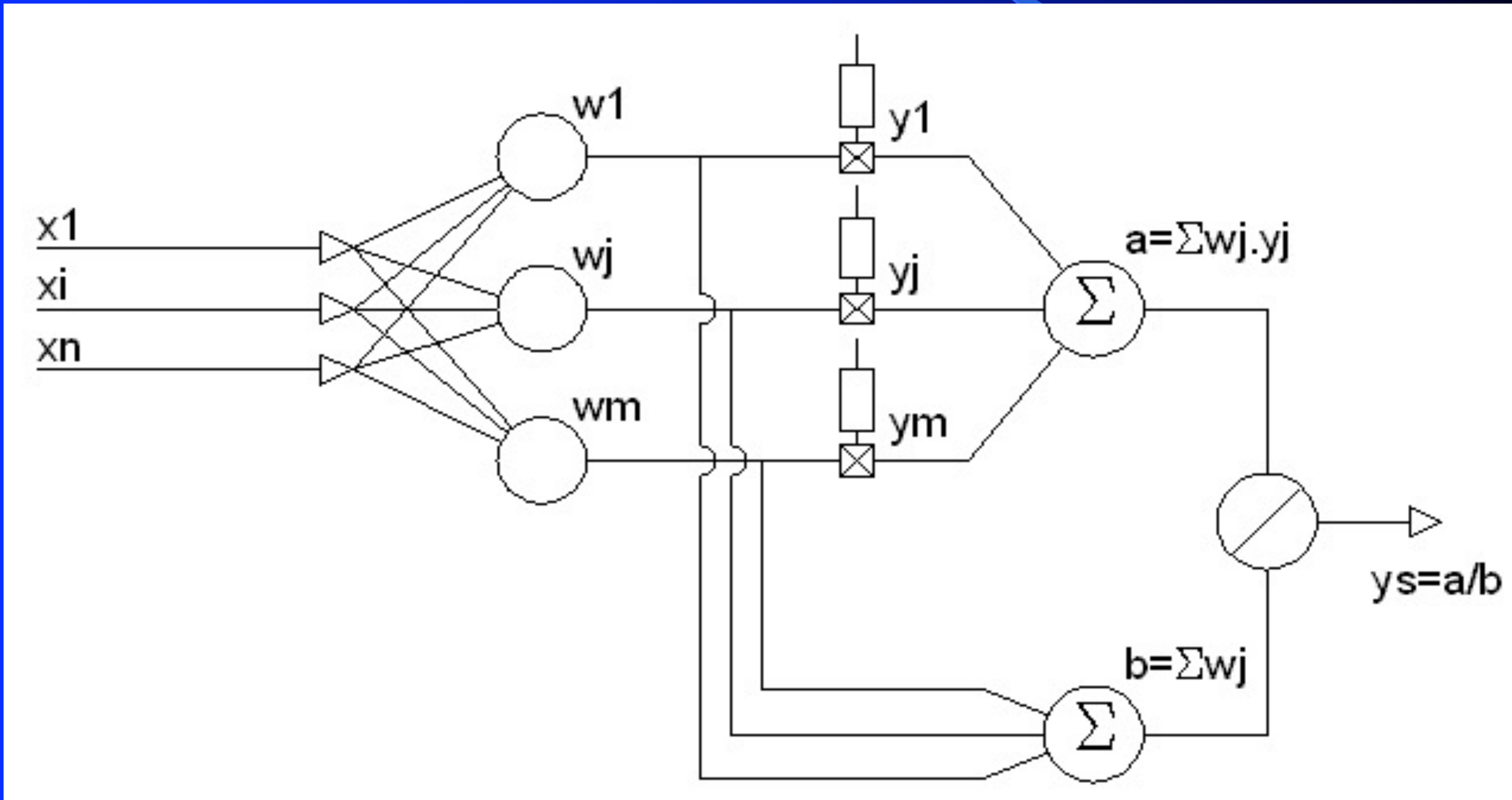
$$\frac{\partial e}{\partial ys} = (ys - yd)$$

$$\frac{\partial ys}{\partial y_j} = \frac{w_j}{b}$$

$$\frac{\partial y_j}{\partial q_j} = 1$$


Estrutura de repetição para número de épocas
Estrutura de repetição para número de pontos da matriz de entrada
-apresentação de dados para a rede,
-cálculo da saída para parâmetros atuais
-Cálculo de derivadas (dedys, dyjdqj)
Estrutura de repetição para o número de regras
-cálculo de derivadas (dydwj, dydyj)
Estrutura de repetição para número de entradas
-cálculo de derivadas (dyjdPij, dwjdcij, dwjds ij)
-atualização de parâmetros (c, sigma, p)
-atualização de parâmetro q
-atualização do valor de saída
-plotagem dos gráficos
-varredura do vetor calculado pela rede
e cálculo do erro quadrático

Estrutura Neurofuzzy Implementada em sala

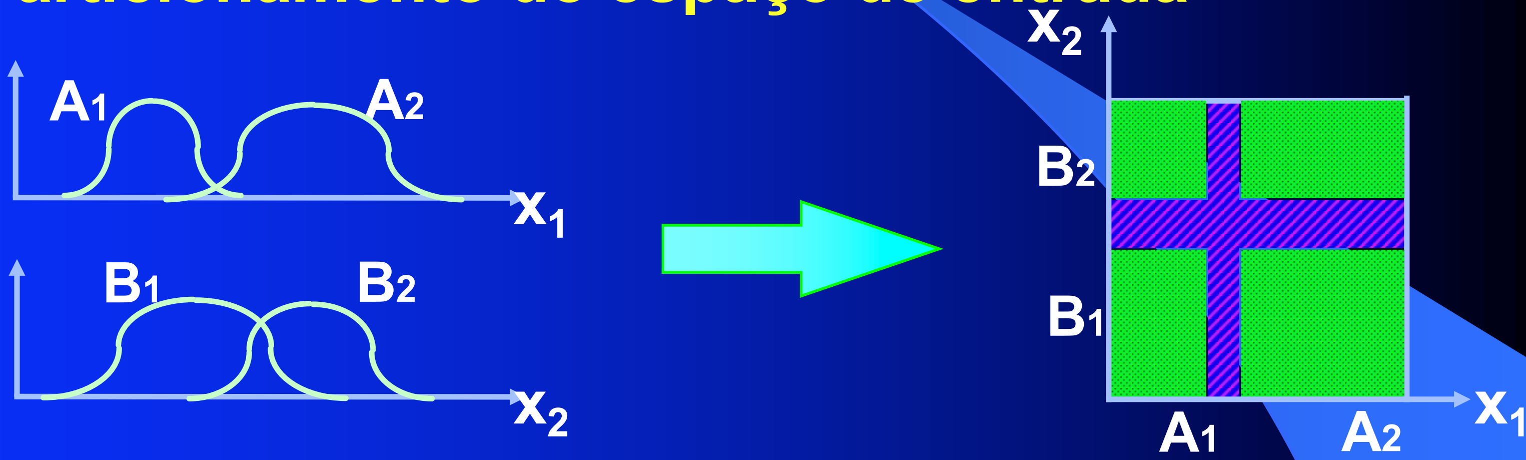


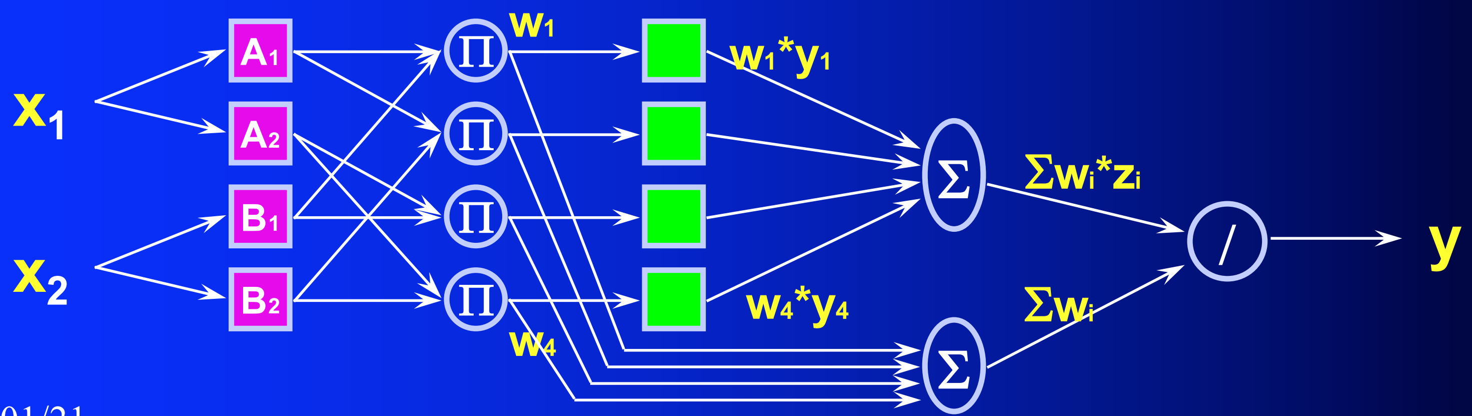
Exercício Computacional 4

- Implemente e estrutura descrita.

GENFIS1 - MatLab

- **Particionamento do espaço de entrada**

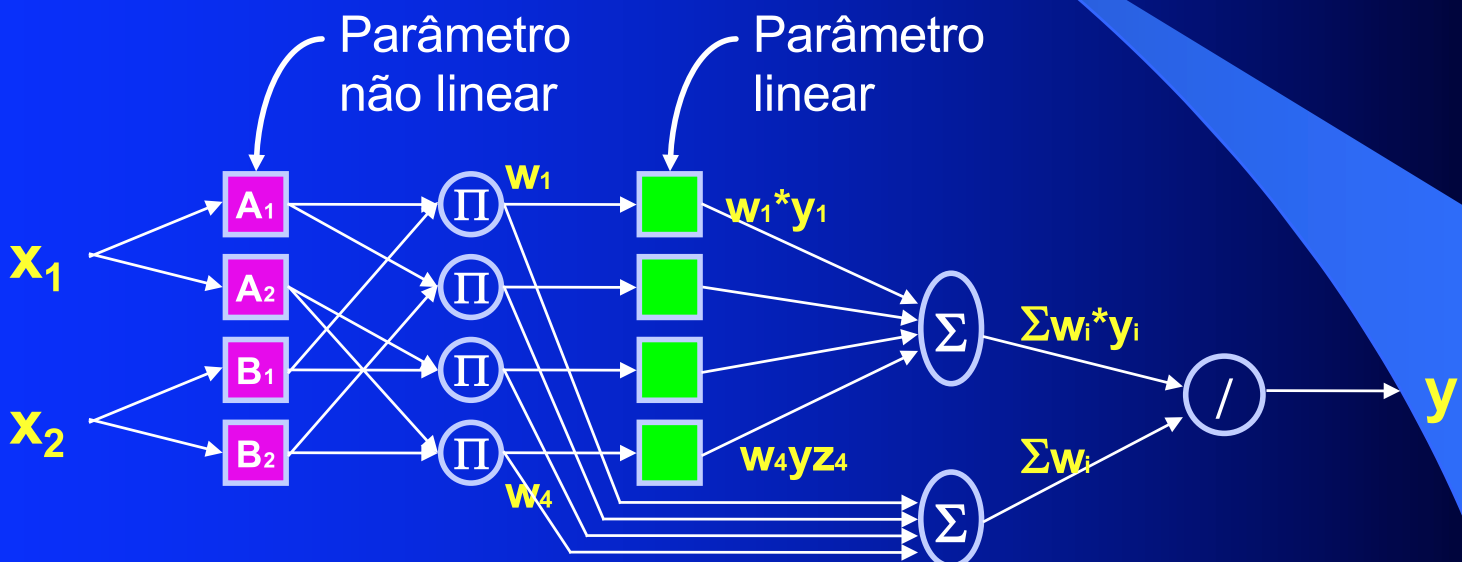




APRENDIZADO DO ANFIS

- Ajuste de parâmetros das funções f_i;
  - mínimos quadrados, métodos de otimização
- Ajuste dos parâmetros das funções de pertinência
  - métodos de otimização (gradiente e outros)

ANFIS: Método de Treinamento Híbrido



	forward pass	backward pass
Parâmetros das MF's	fixo	Gradiente
Coef. param. (linear)	Mínimos quadrados	fixo

ANFIS: APROXIMADOR UNIVERSAL

Um modelo de Sugeno de *ordem zero* é um aproximador universal:

Para qualquer dado $e > 0$ e qualquer função g de valores reais, existe um modelo de *ordem zero* de Sugeno S tal que $|g(x) - S(x)| < e$.

% Implementação do MatLab

```
close all; clear all;clc;
load xt; load ydt; load xv; load ydv

options = genfisOptions('GridPartition');
options.NumMembershipFunctions = 5;
in_fis  = genfis(xt,ydt,options);

options = anfisOptions;
options.InitialFIS = in_fis;
options.EpochNumber = 20;
out_fis = anfis([xt ydt],options);
ys=evalfis(out_fis,xv)
plot(xv,ydv,xx,ys);
legend('Training Data','anfis Output');
```

Exemplo 1: Modelagem de função Sinc com duas entradas

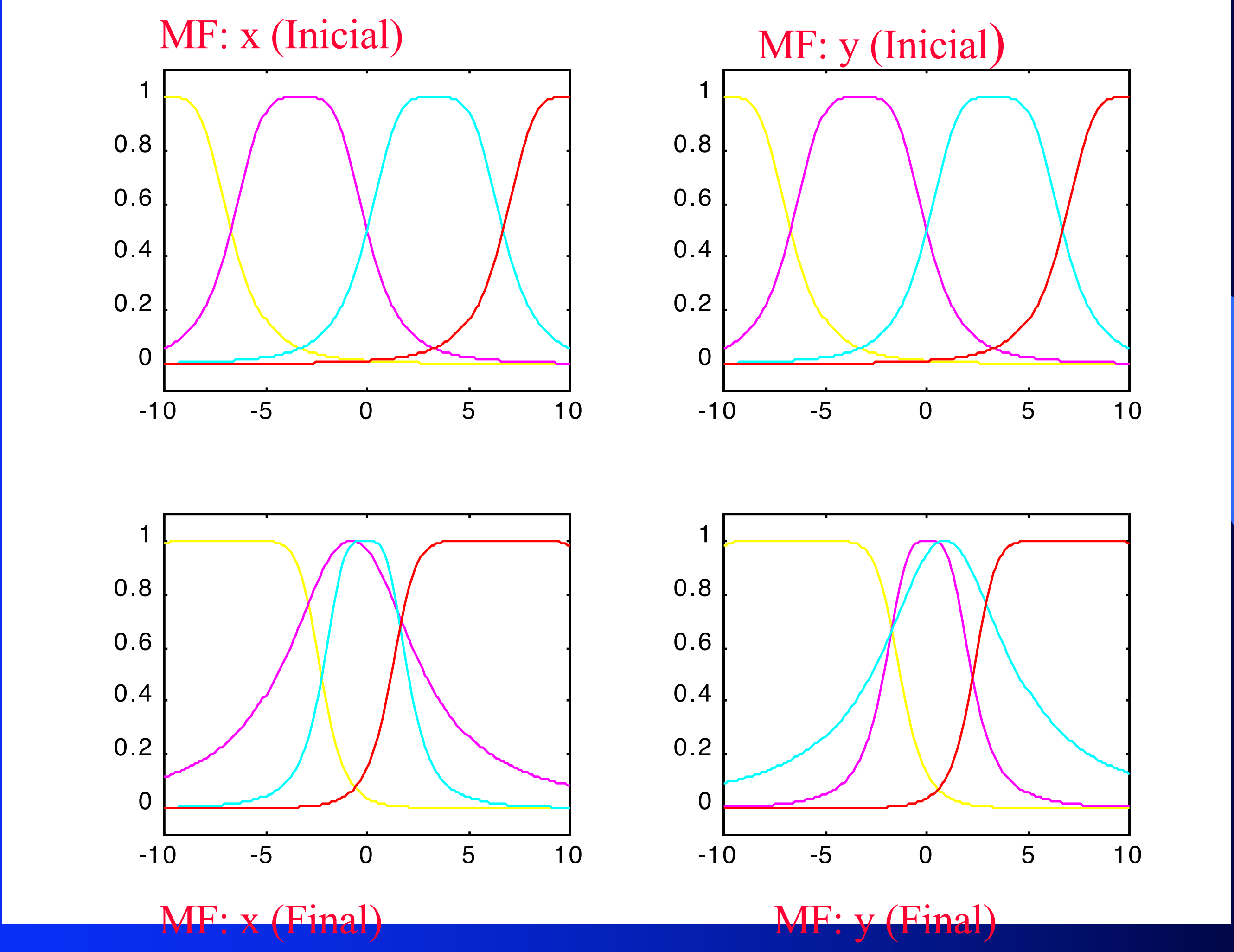
$$z = \operatorname{sinc}(x,y) = \frac{\sin(x).\sin(y)}{x.y}$$

$$\mu_A(x) = gbell(x;a,b,c) = \frac{1}{1+\left|\frac{x-c}{a}\right|^{2b}}$$

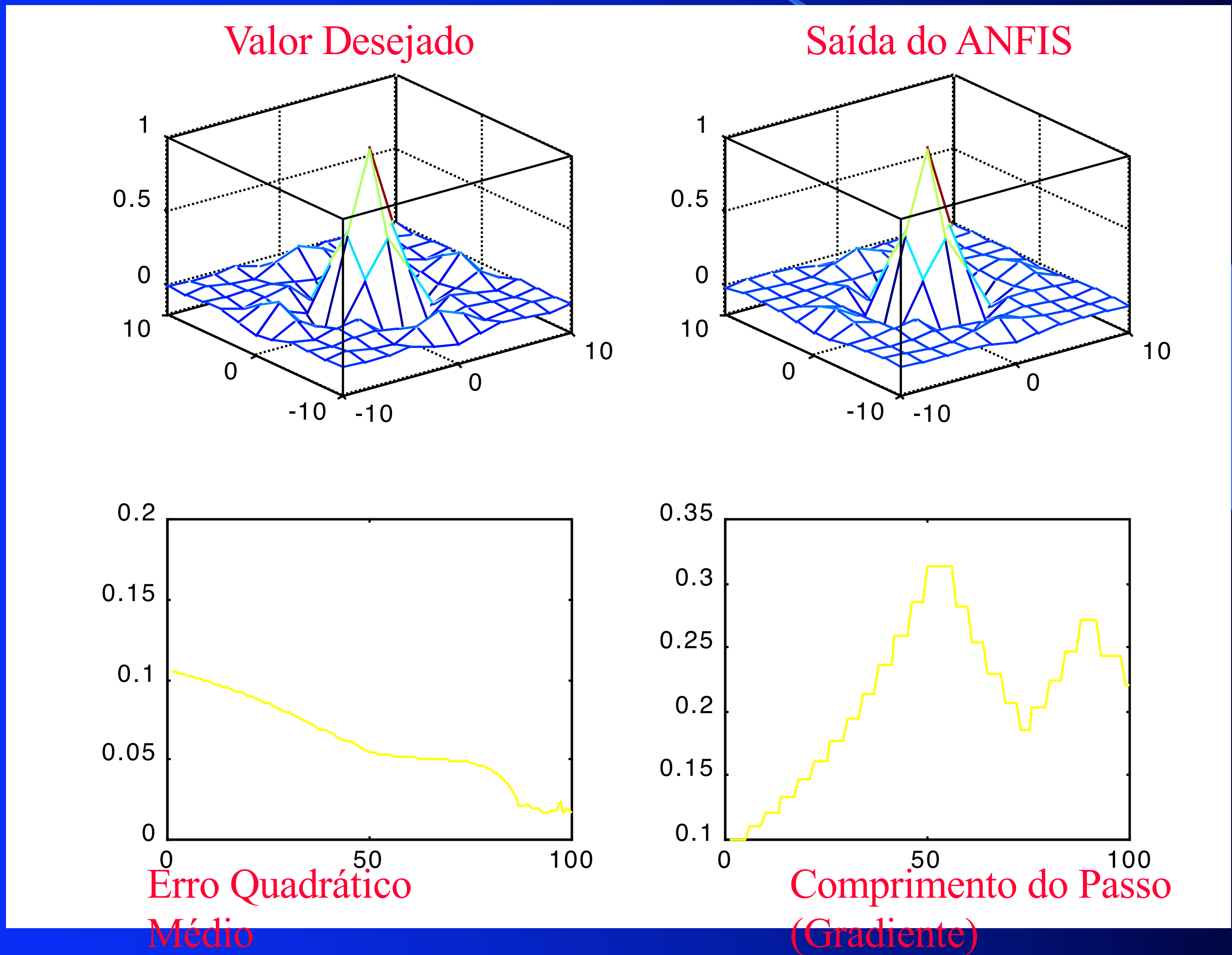
Exemplo 1: Arquitetura ANFIS

- Entrada: distribuição uniforme dos dados de entrada num intervalo de [-10, 10]x[-10, 10];
- Número de padrões para o treinamento: 121 pares de dados de treinamento;
- Topologia do ANFIS:
  - 16 regras, com quatro funções de pertinência para cada variável de entrada;
  - número total de parâmetros a serem ajustados é 72, incluindo 24 parâmetros antecedentes (não lineares) e 48 parâmetros consequentes (lineares).

Exemplo 1: Funções de Pertinência



Exemplo 1: Resultados



Exemplo 1: Comparação com MLP (Quick-prop)

- Convergência mais rápida;
- Erro quadrático médio menor
- Figura 12.7 pag. 347

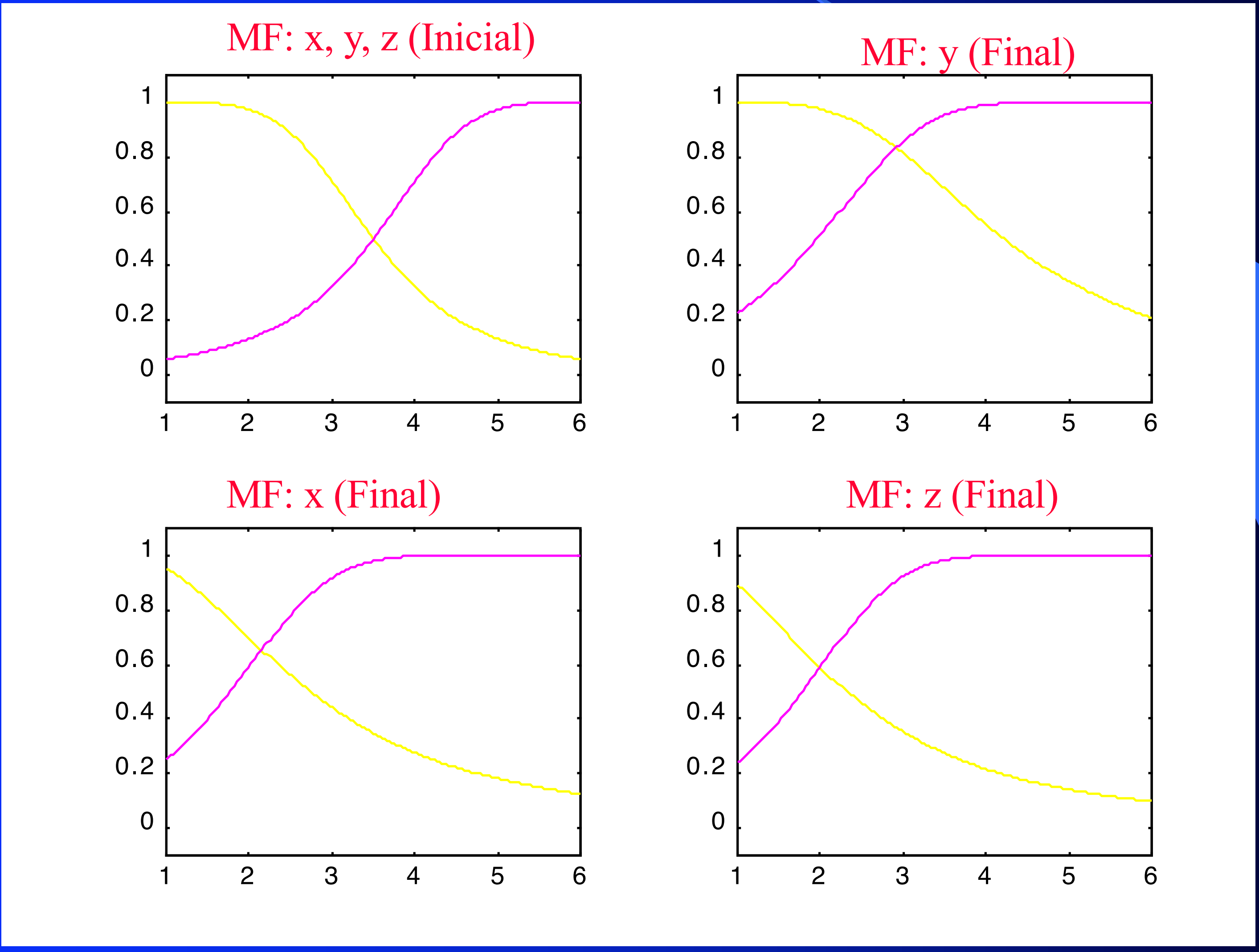
Exemplo 2: Modelagem de função Não-Linear com três entradas

$$ydt = (1 + x^{0.5} + y^{-1} + z^{-1.5})^2$$

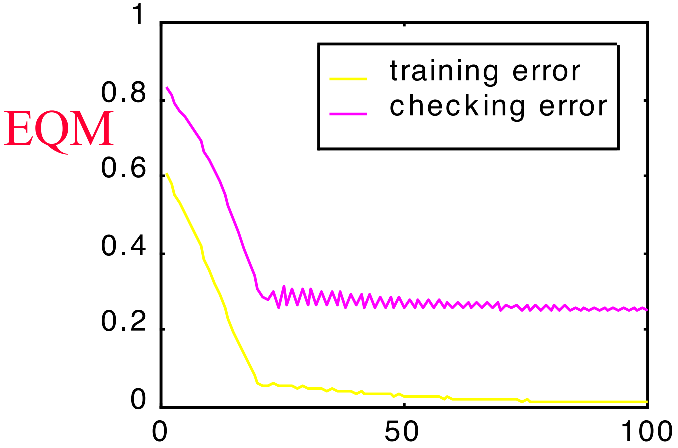
Exemplo 2: Arquitetura ANFIS

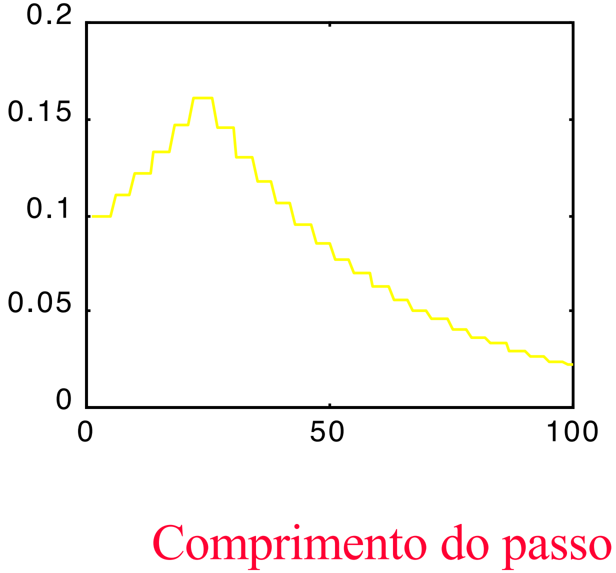
- Número de padrões para o treinamento: 216 distribuição uniforme no intervalo [1,6]x[1,6]x[1,6]
- Número de padrões para validação: 125 distribuição uniforme [1.5,5.5]x[1.5,5.5]x[1.5,5.5];
- Topologia do ANFIS:
  - 8 regras, com duas funções de pertinência para cada variável de entrada;
  - número total de parâmetros a serem ajustados é 50, incluindo 18 parâmetros antecedentes (não lineares) e 32 parâmetros consequentes (lineares).

Exemplo 2: Funções de Pertinência



Exemplo 2: Resultados





Exemplo 2: Comparação com MLP (GMDH)

- ANFIS:
  - EPE Treinamento: 0,043%
  - EPE Validação: 1,006%
- RNA (GMDH):
  - EPE Treinamento: 4,7%
  - EPE Validação: 5,7%

EPE (erro percentual médio)

Exemplo 3: Identificação on-line em sistemas de controle

$$y(k+1) = 0.3y(k)+0.6y(k-1)+f(u(k))$$

$$f(u) = 0.6\sin(\Pi u) + 0.3\sin(3\Pi u) + 0.1\sin(5\Pi u)$$

$$y'(k+1) = 0.3\,y'(k)+0.6y'(k-1)+F(u(k))$$

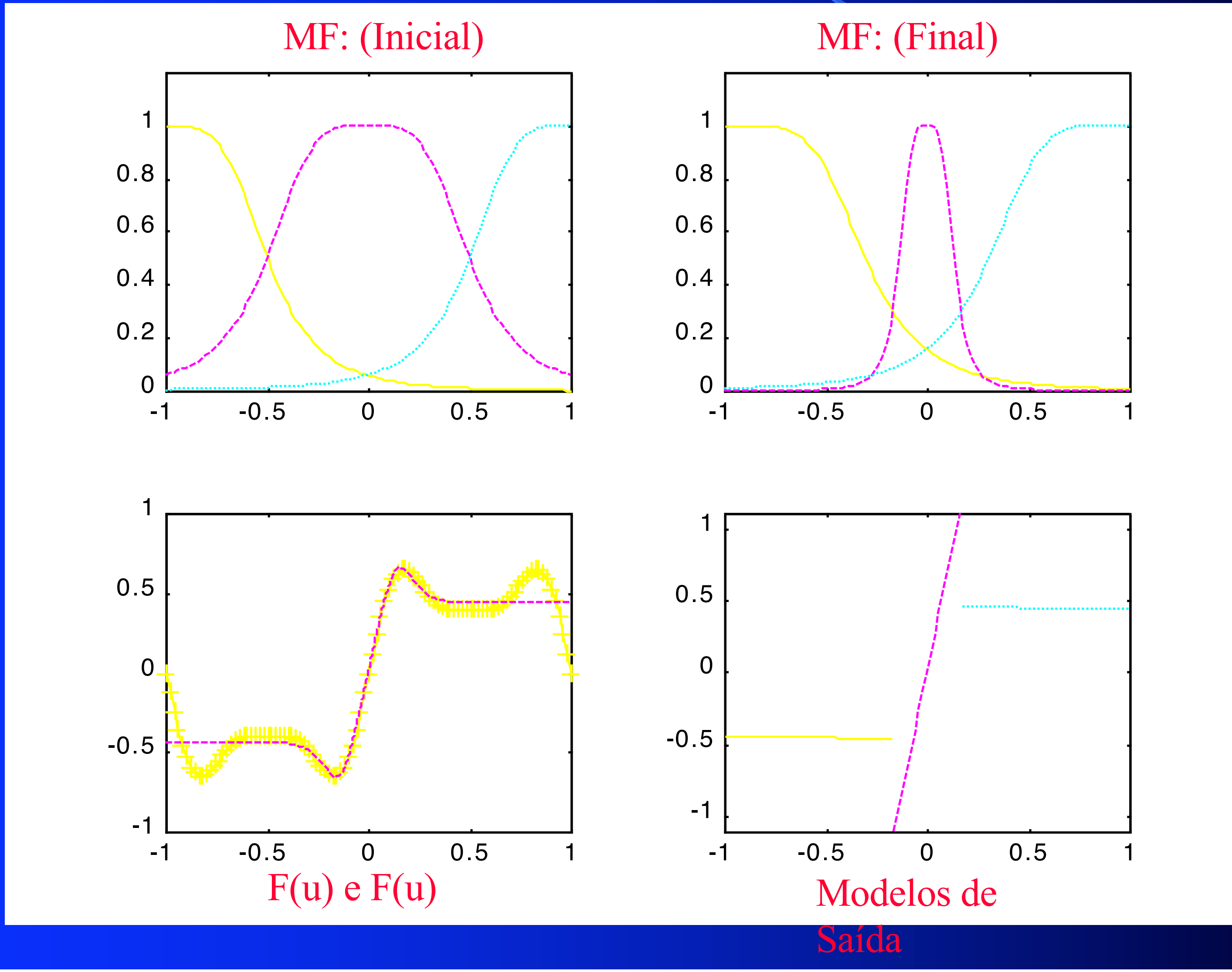
$$u(k) = \sin(2\Pi k/250)$$

F(.) é a função implementada pelo ANFIS

Exemplo 3: Arquitetura ANFIS (1)

- Número de parâmetros lineares: 6
- Número de funções de pertinência= 3
- Parâmetros não lineares: 9
- Total de parâmetros: 15
- Número de padrões de treinamento: 101
- Número de regras nebulosas (no. de modelos de saída): 3

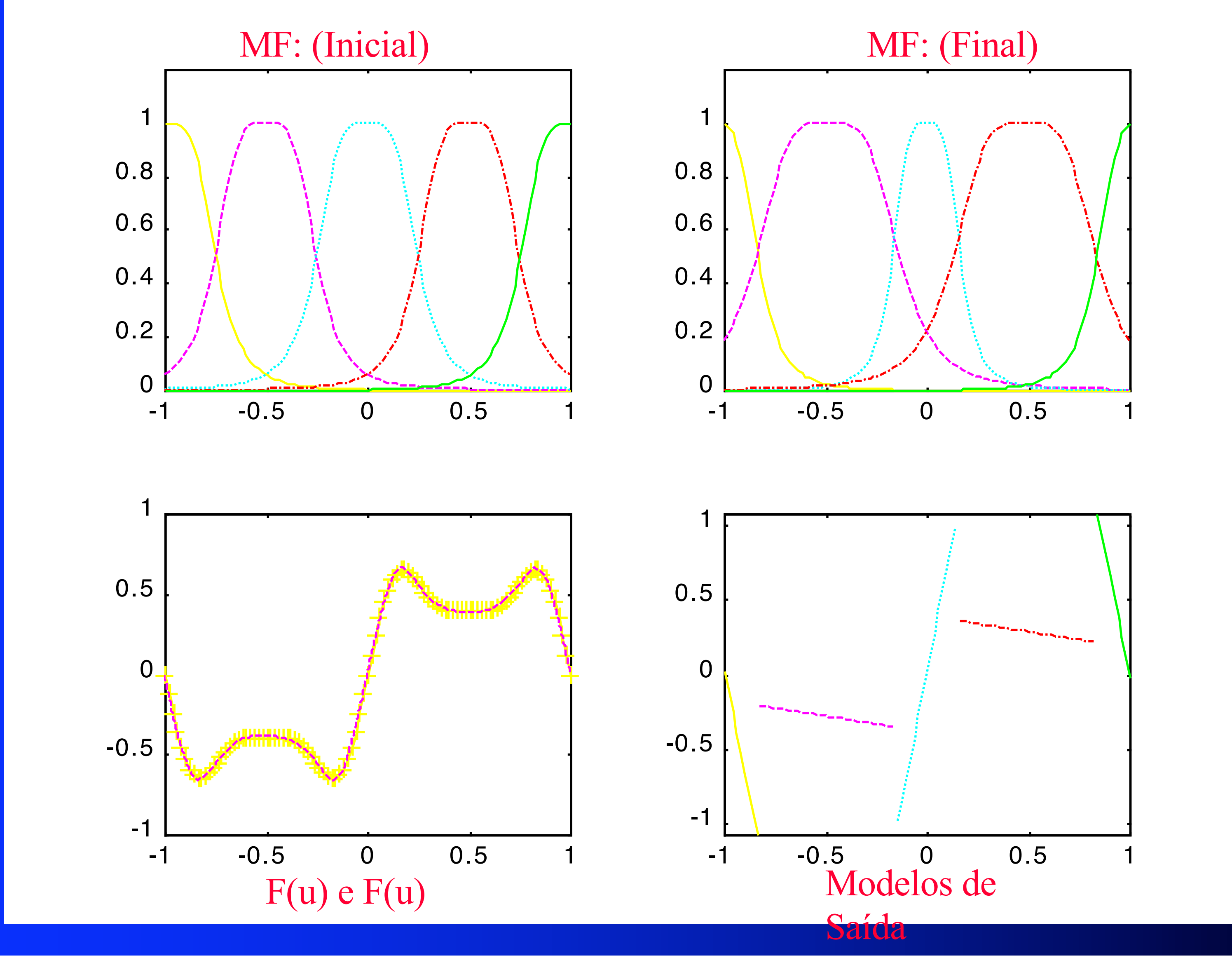
Exemplo 3: Resultados com 3 MF's



Exemplo 3: Arquitetura ANFIS (2)

- Número de parâmetros lineares: 10
- Número de funções de pertinência= 5
- Parâmetros não lineares: 15
- Total de parâmetros: 25
- Número de padrões de treinamento: 101
- Número de regras nebulosas (no. de modelos de saída): 5

Exemplo 3: Resultados com 5 MF's



Exemplo 3: Comparação com MLP

- ANFIS:
  - Número de Parâmetros: 35
  - Períodos de amostragem para adaptação: 250
- RNA (GMDH):
  - Número de Parâmetros: 261
  - Períodos de amostragem para adaptação: 50.000
- Figura 12.13 do livro, página 352

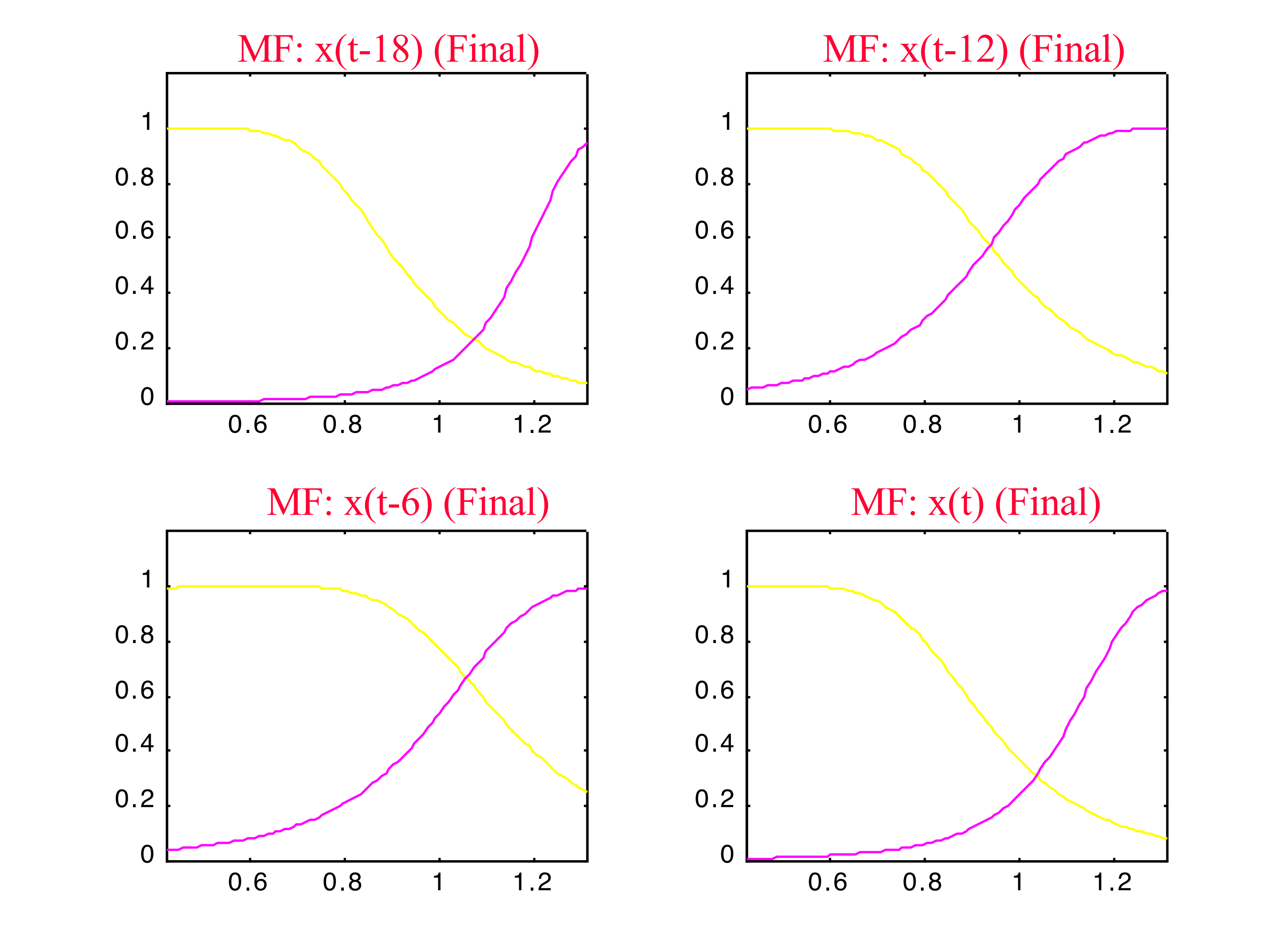
Exemplo 4: Predição de Séries Temporais Caóticas

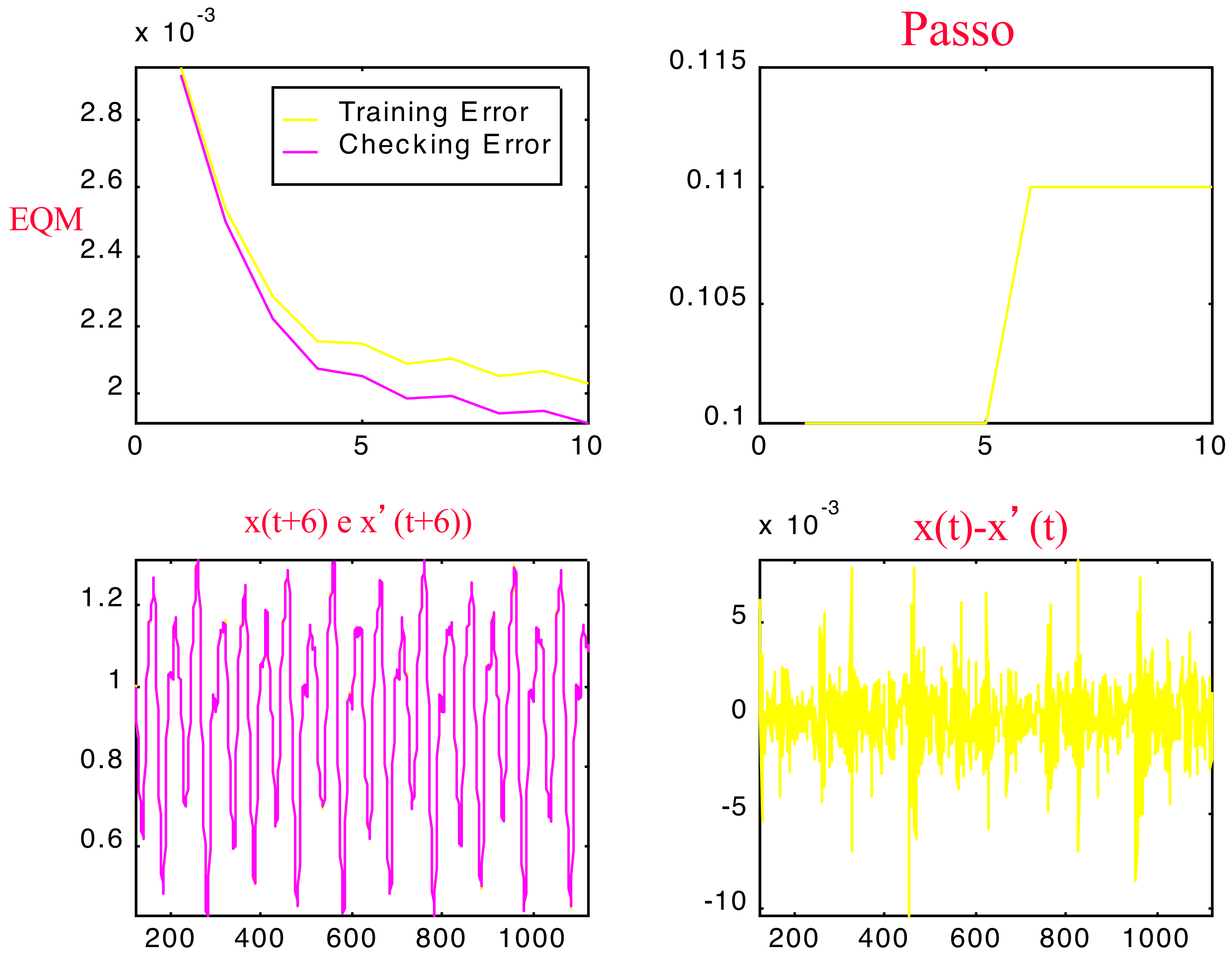
$$\dot{x}(t) = \frac{0.2\,x(t-\tau)}{1+x^{10}(t-\tau)} - 0.1\,x(t)$$

Exemplo 4: Arquitetura ANFIS

- Número de parâmetros lineares: 80
- Número de funções de pertinência= 2
- Parâmetros não lineares: 24 (2x3x4)
- Total de parâmetros: 104
- Número de padrões de treinamento: 500
- Número de padrões de validação: 500
- Número de regras nebulosas (no. de modelos de saída): 16

Exemplo 4: Resultados




x 10^-3
EQM
Training Error
Checking Error
Passo
x(t+6) e x' (t+6))
x 10^-3
x(t)-x' (t)

Exemplo 3: Comparação com Modelos AR

- ANFIS:
  - $EQM_{treinamento} = 0.0016$
  - $EQM_{validação} = 0.0015$

- Modelo AR:
  - $EQM_{treinamento} = 0.0050$
  - $EQM_{validação} = 0.078$