

MODELO ADAPTATIVO BASEADO EM REGRAS NEBULOSAS APLICADO À PREVISÃO DE SÉRIES DE VAZÕES SEMANAIS

IVETTE LUNA*, SECUNDINO SOARES*, ROSANGELA BALLINI†

**Departamento de Engenharia de Sistemas – DENSIS
Faculdade de Engenharia Elétrica e de Computação – FEEC
Universidade Estadual de Campinas – UNICAMP
13081-970 Campinas, SP, Brasil*

†*Departamento de Teoria Econômica – DTE
Instituto de Economia – IE
Universidade Estadual de Campinas – UNICAMP
13083-857 Campinas, SP, Brasil*

Emails: iluna@cose.fee.unicamp.br, dino@cose.fee.unicamp.br,
ballini@eco.unicamp.br

Abstract— This paper suggests an online algorithm for generating Takagi-Sugeno fuzzy systems applied for time series prediction. The model proposed is built in two phases. First, the model structure is initialized via a constructive offline adjustment, considering only two initial fuzzy rules. In the second phase, the system is modified dynamically, applying a recursive learning algorithm based on the expectation maximization technique, as well as adding and pruning operators. The adaptive learning process reduces model complexity and defines automatically the model structure providing an efficient model. The proposed approach is applied to build a time series model for weekly streamflow forecasting and the results are compared with evolving Takagi-Sugeno model. Results show the suggested model is efficient for time series modeling, providing a simpler and parcimonious structure.

Keywords— Time series forecasting, EM algorithm, adaptive learning.

Resumo— Este artigo propõe um modelo de aprendizado online baseado em regras nebulosas do tipo Takagi-Sugeno, para modelos de séries temporais. A estrutura do modelo é construída em duas etapas. Na primeira etapa, o modelo é inicializado utilizando um algoritmo de aprendizado construtivo offline, composto por duas regras nebulosas. Na segunda etapa, o modelo é modificado de forma dinâmica, utilizando um algoritmo de aprendizado recursivo baseado no algoritmo de maximização de verossimilhança e em operadores de adição e punição de regras. O aprendizado adaptativo, reduz a complexidade do modelo, e define de forma automática a sua estrutura. A proposta é aplicada na construção de um modelo de previsão de vazões semanais e os resultados são comparados com o modelo evolutivo Takagi-Sugeno. Os resultados obtidos mostram que a proposta é uma alternativa eficiente para modelos de séries temporais, proporcionando uma estrutura simples e parcimoniosa.

Palavras-chave— Previsão de séries temporais, algoritmo EM, aprendizado adaptativo.

1 Introdução

Redes neurais e sistemas nebulosos têm se mostrado eficientes em diversas aplicações, destacando-se os modelos baseados em regras nebulosas, devido a sua simplicidade e eficiência (Angelov and Filev, 2004). Existem diversas abordagens para a construção de sistemas baseados em regras. Uma técnica bastante divulgada é a utilização de algoritmos de agrupamento, via a partição nebulosa do espaço de entrada em sub-regiões, como o algoritmo *Fuzzy C-Means* (Bezdek, 1981) e o algoritmo da montanha (Yager and Filev, 1994). Por outro lado, existem abordagens baseadas na alocação de modelos locais ou especialistas, as quais estão diretamente vinculadas ao princípio de dividir e conquistar, como é o caso dos modelos de mistura de especialistas (Jacobs et al., 1991).

Devido a complexidade dos problemas reais, os quais são, em geral, de natureza não linear, torna-se impossível saber com antecedência o número adequado de partições ou especialistas necessários para a construção dos modelos. Como consequência, abordagens de aprendizado online têm sido desenvolvidas, fornecendo assim, estratégias de aprendizado com alta capacidade de adaptação a baixo custo computacional.

Seguindo esta direção, diversas técnicas de aprendizado adaptativas e não supervisionadas de mode-

los de redes neurais e de sistemas baseados em regras nebulosas vêm sendo desenvolvidas nos últimos anos. Os trabalhos descritos em (Ramamurti and Ghosh, 1996), (Er and Wu, 2002), (Angelov and Filev, 2004) e (Leng et al., 2005) são apenas alguns exemplos deste tipo de aprendizado, sendo que, estas propostas consideram tanto a adaptação dos parâmetros como da estrutura do modelo.

Este artigo propõe um sistema adaptativo baseado em regras nebulosas (**A-FSM**), sendo o algoritmo de aprendizado baseado no algoritmo de maximização da verossimilhança (EM) (Jacobs et al., 1991). A vantagem de adotar técnicas de aprendizado baseadas no algoritmo EM, é a convergência rápida do algoritmo e a obtenção de ótimos locais, quando comparado ao algoritmo de retropropagação temporal (Haykin, 2001b).

Além disso, este artigo sugere uma versão adaptativa do algoritmo EM tradicional, sendo este aplicado no ajuste de modelos baseados em regras nebulosas, efetuando o ajuste dos parâmetros e da estrutura do modelo a medida que os dados estão sendo apresentados ao sistema definindo, de forma automática, o número de regras necessárias a cada iteração.

O sistema adaptativo é aplicado na previsão de uma série de vazões semanais do posto de Sobradinho.

O modelo **A-FSM** é comparado com o modelo evolutivo Takagi-Sugeno ((**eTS**), proposto em (Angelov and Filev, 2004). Os resultados mostram que a técnica de aprendizado sugerida é eficiente e mostra-se como uma alternativa para o ajuste dinâmico de modelos de séries temporais.

Este artigo está organizado como segue. A Seção 2 descreve a estrutura geral do modelo (**A-FSM**). A Seção 3 detalha o algoritmo de aprendizado. Os resultados de simulação são apresentados na Seção 4. Finalmente, as conclusões são descritas na Seção 5.

2 Estrutura do modelo

A estrutura do sistema proposto é composta por um conjunto de M regras nebulosas de primeira ordem, do tipo Takagi-Sugeno (TS) (Takagi and Sugeno, 1985).

Sejam $\mathbf{x}^k = [x_1^k, x_2^k, \dots, x_p^k] \in \mathbb{R}^p$ o vetor de entrada no instante k , $k \in \mathbb{Z}_0^+$, e $\hat{y}^k \in \mathbb{R}$ a saída do modelo, para uma determinada entrada \mathbf{x}^k .

A representação do sistema baseado em regras nebulosas, pode ser descrita por camadas. A partição do espaço de entrada é representada pelas primeiras duas camadas. Cada sub-região gerada, é definida por um centro $\mathbf{c}_i \in \mathbb{R}^p$ e por uma matriz de covariância \mathbf{V}_i de dimensão $p \times p$. Assim, cada vetor de entrada terá um grau de pertinência associado a cada uma das sub-regiões geradas pela base de regras.

A primeira camada é a responsável pelo cálculo das funções de pertinência a cada iteração k . Estas funções de pertinência são definidas pelo produto $\alpha_i P_i^k$, sendo α_i coeficientes positivos tal que $\sum_{i=1}^M \alpha_i = 1$ e $P_i^k = P[i | \mathbf{x}^k]$ definida como:

$$P_i^k = P[i | \mathbf{x}^k] = \left(\frac{1}{(2\pi)^{p/2} \det(\mathbf{V}_i)^{1/2}} \right) \times \left(\exp \left\{ -\frac{1}{2} (\mathbf{x}^k - \mathbf{c}_i) \mathbf{V}_i^{-1} (\mathbf{x}^k - \mathbf{c}_i)^T \right\} \right) \quad (1)$$

na qual \mathbf{V}_i e \mathbf{c}_i são a matriz de covariância e centro associados à i -ésima regra, respectivamente. A Eq.(1) representa a probabilidade de escolher a regra i , dado que apenas a entrada \mathbf{x}^k é conhecida, assim como os parâmetros \mathbf{V}_i e \mathbf{c}_i . Observa-se que $P[i | \mathbf{x}^k]$ é uma função de densidade de probabilidade condicional, que pode ser interpretada como a probabilidade do vetor de entrada \mathbf{x}^k pertencer à região ativa da i -ésima regra.

O grau de pertinência de cada padrão \mathbf{x}^k é representado por $g_i(\mathbf{x}^k) \in [0, 1]$, dado por:

$$g_i(\mathbf{x}^k) = g_i^k = \frac{\alpha_i \cdot P[i | \mathbf{x}^k]}{\sum_{q=1}^M \alpha_q \cdot P[q | \mathbf{x}^k]} \quad (2)$$

Estas funções de pertinência g_i^k são fornecidas pela segunda camada do modelo, tal que $\sum_{i=1}^M g_i^k = 1$.

Os coeficientes α_i , $i = 1, \dots, M$ podem ser interpretados como índices indiretos de relevância para cada regra gerada durante o treinamento, de tal forma que, quanto mais alto seja o valor de α_i , maior será a importância da i -ésima regra nebulosa.

A saída do sistema \hat{y}^k , é calculada na última camada da estrutura do modelo, como uma combinação não linear dos consequentes y_i^k ponderados pelos seus respectivos graus de pertinência g_i^k , ou seja:

$$\hat{y}^k = \sum_{i=1}^M g_i^k \times y_i^k \quad (3)$$

sendo, neste caso, a saída y_i^k de cada regra R_i , definida como:

$$y_i^k = \phi^k \times \theta_i^T \quad (4)$$

na qual $\phi^k = [1 \ x_1^k \ x_2^k \ \dots \ x_p^k]$ é um vetor que contém os dados de entrada \mathbf{x}^k e o termo constante; e $\theta_i = [\theta_{i0} \ \theta_{i1} \ \dots \ \theta_{ip}]$ é o vetor de coeficientes dos modelos locais lineares, de dimensão $1 \times (p+1)$ para $i = 1, \dots, M$, que representam os consequentes das regras nebulosas. Embora as Equações de (2)-(4) considerem apenas uma única saída, o modelo pode ser estendido para múltiplas saídas.

Como pode ser observado, a dinâmica do modelo pode ser interpretada como um mecanismo de inferência nebulosa, sendo a base de regras composta por regras R_i , $i = 1, \dots, M$, do tipo

R_i^k : **Se** \mathbf{x}^k pertence a i -ésima região com grau de pertinência g_i^k **Então** $y_i^k = \phi^k \times \theta_i^T$.

3 Aprendizado Adaptativo

A identificação do modelo baseado em regras nebulosas é feita de forma que tanto a estrutura do modelo como os parâmetros sejam definidos simultaneamente durante o aprendizado. Assim, o objetivo do aprendizado adaptativo é:

- Definir o número de regras M que compõe a estrutura do modelo a cada iteração k ;
- Definir os parâmetros dos antecedentes das regras nebulosas, ou seja, os centros \mathbf{c}_i e as matrizes de covariância \mathbf{V}_i , $i = 1, \dots, M$;
- Definir os parâmetros dos consequentes ou modelos locais θ_i , $i = 2, \dots, M$

O processo de aprendizado pode ser dividido em duas etapas: inicialização do modelo e aprendizado online. Na primeira etapa, um método de aprendizado offline construtivo é usado para um ajuste inicial de uma base de regras composta por duas regras. Para o ajuste dessa base de regras é utilizada uma pequena parte do histórico da série temporal. Essa inicialização se faz necessária para que seja definido um ótimo local factível. Na segunda fase, a estrutura inicial e os parâmetros do modelo são ajustados de forma dinâmica conforme os dados são apresentados ao modelo via um método de aprendizado online. A seguir, cada uma dessas etapas é detalhada.

3.1 Inicialização

A inicialização do modelo utiliza um aprendizado construtivo baseado no algoritmo EM (Jacobs et al., 1991). Este algoritmo é composto por uma seqüência

iterativa de passos EM que maximiza a função de verossimilhança por meio do ajuste dos parâmetros do modelo. O algoritmo construtivo (Luna et al., 2007), é uma modificação do algoritmo EM tradicional, pois considera, durante o processo de ajuste dos parâmetros, operadores de adição de novas regras à estrutura, assim como, operadores de punição ou eliminação de regras existentes e pouco relevantes para o modelo de previsão. Este algoritmo é aplicado para inicializar o modelo adaptativo, por meio de um processo de ajuste offline, obtendo assim, uma inicialização adequada, que dará início à segunda etapa do processo de aprendizado.

Seja \mathbf{V}_i a matriz de covariância diagonal e definida positiva, de tal forma que soluções infactíveis sejam evitadas. De acordo com o algoritmo EM (Jacobs et al., 1991), e deduzindo as equações para a maximização da função de verossimilhança, uma solução ótima para o modelo proposto a cada passo M é definido por:

$$\alpha_i = \frac{1}{N} \sum_{k=1}^N h_i^k \quad (5)$$

$$\mathbf{c}_i = \left(\sum_{k=1}^N h_i^k \mathbf{x}^k \right) / \sum_{k=1}^N h_i^k \quad (6)$$

$$\mathbf{V}_i = \left(\sum_{k=1}^N h_i^k (\mathbf{x}^k - \mathbf{c}_i)' (\mathbf{x}^k - \mathbf{c}_i) \right) / \sum_{k=1}^N h_i^k \quad (7)$$

para $i = 1, \dots, M$, onde N é o número de dados utilizado na fase de inicialização e h_i^k , $i = 1, \dots, M$ é a probabilidade a posteriori estimada no passo E do processo de otimização. Uma solução ótima para θ_i surge a partir da solução via mínimos quadrados, para a seguinte equação:

$$\sum_{k=1}^N \frac{h_i^k}{\sigma_i^2} (y^k - \phi^k \times \theta_i) \cdot \phi^k = 0 \quad (8)$$

na qual y^k a saída desejada no instante k e σ_i o desvio padrão associado à saída local y_i , $i = 1, \dots, M$, sendo a variância σ_i^2 dada por:

$$\sigma_i^2 = \left(\sum_{k=1}^N h_i^k [y^k - y_i^k]^2 \right) / \sum_{k=1}^N h_i^k \quad (9)$$

Maiores detalhes referentes ao algoritmo EM podem ser encontrados em (Jacobs et al., 1991).

O algoritmo de aprendizado construtivo é utilizado para inicializar o modelo e definir de forma automática a base de regras inicial do aprendizado online. Como foi mencionado no início da seção, este algoritmo utiliza operadores de adição e eliminação de regras nebulosas. Estes operadores também serão utilizados como base para a adaptação da estrutura do modelo no aprendizado online.

3.1.1 Adição offline de regras nebulosas

O critério proposto para determinar uma situação na qual é necessário gerar uma nova regra nebulosa é chamado de critério do antecedente. Este critério verifica

se a base de regras atual cobre o espaço de entrada de forma adequada. Assim, cada dado apresentado ao modelo deve pertencer a uma região ativa de pelo menos uma das regras nebulosas já existentes na estrutura do modelo, com uma probabilidade mínima superior a um limiar previamente definido.

Supondo que os dados tenham uma distribuição normal, para um dado nível de confiança de $\gamma\%$, os dados pertencentes à região ativa da i -ésima regra nebulosa estarão contidos no intervalo $[\mathbf{c}_i - z_\gamma \sqrt{\text{diag}(\mathbf{V}_i)}, \mathbf{c}_i + z_\gamma \sqrt{\text{diag}(\mathbf{V}_i)}]$, onde $\text{diag}(\mathbf{V}_i)$ é a diagonal principal de \mathbf{V}_i . Neste artigo foi adotado um nível de confiança $\gamma = 72,86\%$, resultando em um $z_\gamma = 1,1$ (valor encontrado na tabela de distribuição normal) e um limiar igual a 0,1357. Esta condição implica que a máxima probabilidade $P[i | \mathbf{x}^k]$ de \mathbf{x}^k ativar pelo menos uma das regras nebulosas tem que ser maior que 0,1357. Ou seja:

$$\max (P[i | \mathbf{x}^k])_{i=1, \dots, M} > 0,1357 \quad (10)$$

Se a condição definida pela Eq. (10) não é satisfeita por um ou mais dados do conjunto de treinamento, significa que estes dados possuem uma probabilidade de pertencer a alguma região ativa coberta pela base de regras, inferior ao desejado. Denotando por Ω o conjunto contendo todos os dados entrada-saída que não satisfazem a Eq. (10), uma nova regra será inserida à estrutura do sistema. Assim,

$$M = M + 1 \quad (11)$$

e a estimativa inicial do centro desta nova regra \mathbf{c}_M é dada por:

$$\mathbf{c}_M = \frac{1}{N_\Omega} \sum_{t \in \Omega} \mathbf{x}^t \quad (12)$$

sendo N_Ω o número de dados $[\mathbf{x}^t \ y^t]$ em Ω . Além disso, os demais parâmetros associados a essa nova regra são inicializados da seguinte forma:

- $\sigma_M = 1,0$;
- $\theta_M = [\bar{y} \ 0 \ \dots \ 0]_{1 \times (p+1)}$, sendo \bar{y} o valor médio das saídas desejadas pertencentes ao conjunto Ω .

Os parâmetros α_i e \mathbf{V}_i , para $i = 1, \dots, M$, devem ser re-inicializados de acordo com:

- $\mathbf{V}_i = 10^{-4} \mathbf{I}$, onde $\mathbf{I}_{p \times p}$ é a matriz identidade;
- $\alpha_i = 1/(M)$.

Como α_i indica a relevância de cada regra nebulosa, a sua re-inicialização é necessária para que todas as regras tenham o mesmo peso, fazendo com que o ajuste dos parâmetros seja não-tendencioso, obtendo assim uma nova partição do espaço de entrada. A re-inicialização de \mathbf{V}_i é necessária já que o processo de re-ajuste definirá novos valores de dispersão para cada regra nebulosa, de acordo com a nova partição gerada, ou seja, considerando neste processo a nova regra inserida na estrutura do modelo.

3.1.2 Eliminação offline de regras nebulosas

Como pode ser observado na Eq. (5), α_i pode ser interpretado como uma medida da relevância que cada regra nebulosa tem para a estrutura atual do modelo, quando comparadas entre si. Isto acontece porque α_i é proporcional a soma de todas as probabilidades h_i^k , as quais também podem ser consideradas como estimativas a posteriori das funções de pertinência g_i^k , sob o conjunto de dados de treinamento. Assim, quanto mais vezes a i -ésima regra nebulosa for ativada, maior será o valor de α_i .

Logo, α_i é o parâmetro que defini se existe ou não alguma regra a ser punida e eliminada da estrutura do modelo. Para tal, a regra com menor α_i será uma boa candidata e, para um dado limiar α_{min} , toda regra com $\alpha_i < \alpha_{min}$ a cada iteração, será eliminada da estrutura do sistema.

Após a inserção ou eliminação de alguma regra, o modelo será reajustado durante algumas iterações EM por meio das equações definidas na Seção 2 e Subseção 3.1. O processo global de aprendizado é finalizado após atingir a convergência e quando não existir mais mudanças na estrutura do modelo. Estes dois operadores de regras nebulosas também serão utilizados no aprendizado online, considerando algumas modificações que serão descritas a seguir.

3.2 Aprendizado online

O aprendizado online de modelos baseados em regras nebulosas, tem se tornado um método promissor, com diversos trabalhos propondo esta abordagem. Alguns exemplos de modelos com aprendizado online, que além de considerar o ajuste dos parâmetros, também consideram a adaptação da estrutura são os trabalhos desenvolvidos por J. Er (Er and Wu, 2002) e os modelos produzidos por P. Angelov (Angelov and Filev, 2004).

O maior atrativo do aprendizado online, no caso da construção de modelos de séries temporais, é que não é necessário re-ajustar o modelo com todo o histórico da série, toda vez que um novo dado é disponibilizado. Além disso, a estrutura do modelo é modificada de acordo com a necessidade ao longo do tempo. Este artigo propõe um aprendizado online de modelos de séries temporais baseados em regras nebulosas, transformando as Eqs. (5)-(9) para uma versão recursiva, de tal forma que o ajuste da estrutura e dos parâmetros do modelo sejam feitos dinâmica e simultaneamente.

O aprendizado adaptativo é realizado a cada instante de tempo k , $k = 1, 2, \dots$. Neste tipo de aprendizado as informações mais recentes têm uma maior influência no ajuste dos parâmetros e na estrutura do modelo. Observando a Eq. (5), e considerando uma janela no tempo $T = N$, esta equação pode ser re-escrita como segue:

$$\begin{aligned}\alpha_i^N &= \frac{1}{N} \sum_{k=1}^N h_i^k = \frac{1}{N} \left(\sum_{k=1}^{N-1} h_i^k + h_i^N \right) \\ &= \frac{1}{N} [(N-1)\alpha_i^{N-1} + h_i^N] \\ &= \alpha_i^{N-1} + \frac{1}{N} [h_i^N - \alpha_i^{N-1}] \quad (13)\end{aligned}$$

onde h_i^N é a estimação a posteriori de g_i^k e α_i^{N-1} é o valor estimado para α_i considerando apenas os $N-1$ dados iniciais. Como pode ser observado, a Eq. (13) é uma estimação recursiva de (5). Seguindo este mesmo procedimento para um número genérico de iterações k e com uma janela no tempo T , as Eqs. (5)-(9) podem ser re-escritas recursivamente, obtendo as seguintes aproximações:

$$\alpha_i^{k+1} = \alpha_i^k + \frac{1}{T} [h_i^k - \alpha_i^k] \quad (14)$$

$$\mathbf{c}_i^{k+1} = \mathbf{c}_i^k + \frac{1}{\gamma_i^{k+1}} [\mathbf{x}^k - \mathbf{c}_i^k] \quad (15)$$

$$\mathbf{V}_i^{k+1} = \mathbf{V}_i^k + \frac{1}{\gamma_i^{k+1}} [(\mathbf{x}^k - \mathbf{c}_i^k)(\mathbf{x}^k - \mathbf{c}_i^k)' - \mathbf{V}_i^k] \quad (16)$$

$$(\sigma_i^2)^{k+1} = (\sigma_i^2)^k + \frac{1}{\gamma_i^{k+1}} [(y^k - y_i^k)^2 - (\sigma_i^2)^k] \quad (17)$$

na qual:

$$\frac{1}{\gamma_i^{k+1}} = \frac{h_i^{k+1}}{\sum_{t=1}^{k+1} h_i^t} \quad (18)$$

Denotando por $S_i^{k+1} = \sum_{t=1}^{k+1} h_i^t$, e $S_i(\mathbf{x}^{k+1}) = h_i^{k+1}$, uma aproximação para $\sum_{t=1}^{k+1} h_i^t$ pode ser obtida como:

$$S_i^{k+1} \approx S_i(\mathbf{x}^{k+1}) + \frac{T-1}{T} S_i^k \quad (19)$$

Re-escrivendo S_i^{k+1} tem-se:

$$S_i^{k+1} \approx S_i^k + [S_i(\mathbf{x}^{k+1}) - \frac{S_i^k}{T}] \quad (20)$$

A Eq. (20) expõe alguns aspectos interessantes referentes às estimativas adotadas. S_i^k/T pode ser interpretada como uma estimativa do valor médio de α_i no intervalo de tempo T , prévio a $k+1$. Assim, quanto maior for S_i^k/T , maior será a importância da i -ésima regra para a próxima iteração. Se $S_i(\mathbf{x}^{k+1})$ obtém um valor baixo em T , então S_i^{k+1} diminuirá e a i -ésima regra terá um risco maior de ser eliminada.

Para estimar θ_i , aplica-se o algoritmo recursivo dos quadrados mínimos ponderados (wRLS), o qual considera um fator de esquecimento ao longo do tempo (f_{forget}). Maiores detalhes concernentes a este algoritmo de otimização podem ser encontrados em (Haykin, 2001a) e em (Ramamurti and Ghosh, 1996). Assim, as equações do algoritmo wRLS adaptadas ao problema de otimização de θ_i são definidas como:

$$\theta_i^{k+1} = \theta_i^k + \mathbf{C}_i^{k+1} \phi^k \times h_i^k (y^k - y_i^k) \quad (21)$$

na qual:

$$\mathbf{C}_i^{k+1} = \frac{\mathbf{C}_i^k}{f_{forget}^k + h_i^k (\phi^k)^T \mathbf{C}_i^k \phi^k} \quad (22)$$

é a matriz de covariância para cada θ_i durante o ajuste online. Neste artigo, o fator de esquecimento $f_{forget} \in (0, 1]$ foi inicializado para $k=0$ como $f_{forget}^0 = 0,85$ e suavemente incrementado, tal que $f_{forget}^\infty \rightarrow 1,0$.

As condições iniciais para θ_i^0 , $i = 1, \dots, M$ foram dadas pelos valores obtidos na fase de inicialização.

Desta forma, e após a fase de inicialização, o aprendizado online é aplicado. Enquanto os parâmetros dos modelos são ajustados utilizando as Eqs. (14)-(22), para ajustar a estrutura do modelo a cada iteração é necessário adaptar os operadores de adição e eliminação utilizados na fase de inicialização.

3.2.1 Geração online de regras nebulosas

Similar ao operador de geração de regras novas utilizado na fase offline de inicialização, a geração online de regras nebulosas será aplicada verificando a condição definida na Eq. (10) para o dado atual de entrada \mathbf{x}^k . Caso esta condição não seja satisfeita, uma nova regra é gerada, sendo os seus parâmetros inicializados como segue:

- $\mathbf{c}_{M+1}^{k+1} = \mathbf{x}^k$
- $\sigma_{M+1}^{k+1} = 1.0$;
- $\theta_{M+1}^{k+1} = [y^k \ 0 \ \dots \ 0]_{1 \times p+1}$
- $\mathbf{V}_{M+1}^{k+1} = 10^{-4} \mathbf{I}$, onde \mathbf{I} é uma matriz identidade de ordem $p \times p$;
- $\alpha_{M+1}^{k+1} = 10^{-5}$. Embora este valor inicial seja pequeno o suficiente para interferir na dinâmica da estrutura atual, todos os parâmetros α_i , $i = 1, \dots, M+1$ são re-normalizados tal que a soma de todos estes coeficientes seja sempre igual a unidade.

3.2.2 Eliminação online de regras nebulosas

O operador de punição ou eliminação de regras nebulosas no aprendizado online, baseia-se no operador utilizado no aprendizado construtivo da fase de inicialização. No entanto, uma condição a mais deve ser verificada. Após uma regra ser gerada e inserida na estrutura do modelo, o seu respectivo α_i terá um valor pequeno. Caso o operador de punição seja aplicado diretamente, a regra recém gerada será eliminada, sem ter um tempo mínimo para o seu ajuste e posterior verificação da sua relevância para as futuras iterações. Esta situação é resolvida definindo um índice chamado de índice de permanência τ . Cada vez que uma nova regra é gerada, o seu respectivo índice de permanência será inicializado e incrementado a cada ativação no futuro. Ou seja:

$$\tau_i^{k+1} = \tau_i^k + 1 \quad (23)$$

Assim, uma regra poderá ser eliminada apenas se o α_i associado a essa regra for inferior ao limiar pré-definido e $\tau_i^k > \epsilon T$, onde $\epsilon > 0$ e T é a mesma janela no tempo utilizada durante o ajuste online. Esta condição garante que nenhuma regra seja eliminada logo após a sua inserção na estrutura do modelo, permitindo o seu ajuste durante um mínimo de tempo, eliminando assim, oscilações bruscas e pouco necessárias na estrutura do modelo.

4 Previsão de séries temporais

O modelo proposto é utilizado na construção de um modelo de série temporal para a série de vazões semanais do posto de Sobradinho, localizado na bacia do rio São Francisco, região nordeste do Brasil. Estes dados abrangem o período da primeira semana de Janeiro de 1931 até a última semana de Dezembro de 2000.

Os dados são separados em dois conjuntos. O primeiro, composto apenas pelos primeiros cinco anos é utilizado na fase de inicialização do modelo. Este mesmo conjunto de dados é analisado para a seleção adequada de entradas para o modelo de previsão. Para tal, utilizou-se o critério de Informação Mútua Parcial (PMI), aplicando a metodologia descrita em (Luna et al., 2006). De acordo com este critério, a partir de um conjunto inicial de possíveis entradas igual a 15, foram selecionados os atrasos 1, 2, 3 e 6, para a construção do modelo de série temporal.

Assim, os vetores de entrada-saída são construídos, e procede-se a inicialização do modelo, usando o algoritmo offline. O parâmetro α_{min} para a fase de inicialização foi igual a 0,001 obtendo uma estrutura inicial composta por 18 regras nebulosas.

A fase de ajuste online foi inicializada a partir da primeira semana de Janeiro de 1936. O parâmetro α_{min} para esta fase de ajuste dinâmico foi igual a 0,0025, $T = 104$, o qual representa uma janela de tempo equivalente aos dois últimos anos, e $\epsilon = 0.50$. A Figura 1 ilustra a variação do número de regras durante o aprendizado online. Em geral, o número de regras durante o ajuste online da estrutura do modelo de série temporal é sempre menor ou igual ao número inicial de regras obtidas na inicialização offline. Isto ocorre pois o aprendizado online permite que a estrutura do modelo mantenha apenas as regras nebulosas de maior relevância no período T imediatamente anterior ao instante atual $k+1$, mesmo definindo um α_{min} maior que o utilizado na fase de inicialização. Para um instante $k > 500$, nota-se que a estrutura do modelo oscilou entre um total de 4 a 12 regras nebulosas.

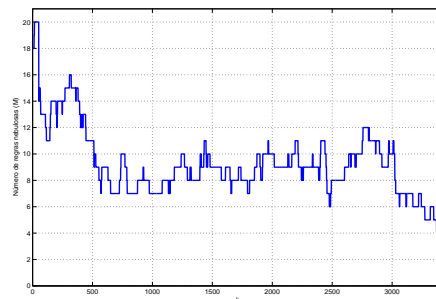


Figura 1: Variação do número de regras durante o aprendizado online.

A Figura 2 ilustra a análise dos resíduos durante o aprendizado online para verificar a adequação do modelo. A Figura 2-(a) ilustra a função de autocorrelação parcial estimada e a Figura 2-(b) ilustra a fun-

ção de correlação cruzada estimada a cada iteração. Em ambas figuras, as linhas tracejadas representam os limites $\pm 2/\sqrt{n}$, os quais fornecem uma indicação geral de possível quebra do comportamento de ruído branco, com n igual ao número de graus de liberdade do modelo obtido. Observa-se que, a medida que os dados são apresentados ao modelo, os resíduos estimados estão sendo menos correlacionados, localizando-se dentro dos limites permitidos.

Para comparação dos resultados, neste artigo o modelo eTS foi ajustado para a série de vazões semanais. Como o modelo pode ser inicializado sem nenhuma informação *a priori*, ou seja, sem nenhuma regra e com somente um dado, o modelo eTS foi aplicado as séries de vazões para previsão um passo à frente, a partir do ano de 1936.

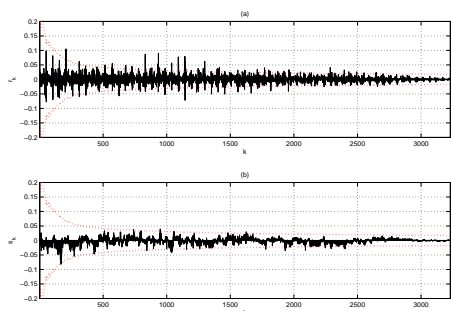


Figura 2: Análise dos resíduos.

A Figura 3 apresenta as previsões da série de vazões para os modelos A-FSM e eTS, na previsão um passo à frente, a partir da primeira semana de Janeiro de 1991 até a última semana de Dezembro de 2000. A Tabela 1 apresenta os erros relativo médio (ERM (%)), raiz do erro quadrático médio (REQM (m^3/s)), absoluto médio (EAM (m^3/s)) e coeficiente de correlação (ρ) para os modelos A-FSM e eTS, no período de Janeiro de 1936 a Dezembro de 2000.

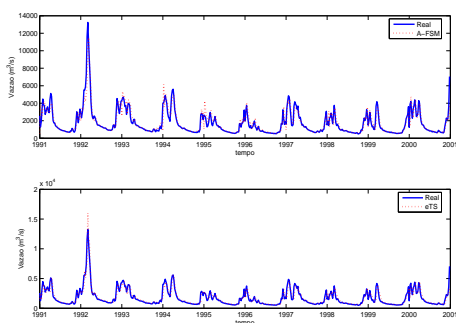


Figura 3: Previsão um passo à frente.

Tabela 1: Erros de previsão.

Modelo	M (rules)	ERM (%)	REQM (m^3/s)	EAM (m^3/s)	ρ
A-FSM	4-12	15,74	614,04	335,75	0,91
eTS	13	17,52	570,77	341,88	0,92

5 Conclusões

Este artigo apresenta um modelo baseado em regras nebulosas com aprendizado adaptativo para previsão de séries temporais. A estratégia de aprendizado é baseada no algoritmo de otimização EM e em operadores de adição e eliminação de regras nebulosas da estrutura do modelo. Estes operadores são aplicados durante o aprendizado definindo de forma automática a estrutura do modelo. O modelo A-FSM foi aplicado para a geração de um modelo de previsão de vazões semanais do posto de Sobradinho, e aplicado na previsão um passo à frente e os resultados foram comparados com o modelo eTS. A análise apresentada mostrou que o sistema nebuloso proposto fornece modelos de previsão adequados e parcimoniosos.

Agradecimentos

Os autores agradecem à Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP), ao Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), pelo auxílio.

Referências

- Angelov, P. P. and Filev, D. P. (2004). An Approach to Online Identification of Takagi-Sugeno Fuzzy Models, *IEEE Transactions on Systems, Man and Cybernetics-part B* **34**(1): 484–498.
- Bezdek, J. (1981). *Pattern Recognition with Fuzzy Objective Function Algorithms*, New York, USA: Plenum Press.
- Er, M. J. and Wu, S. (2002). A fast learning algorithm for parsimonious fuzzy neural systems, *Fuzzy Sets and Systems* **126**: 337–351.
- Haykin, S. (2001a). *Kalman Filtering and Neural Networks*, John Wiley & Sons, Inc. .
- Haykin, S. (2001b). *Redes Neurais, Princípios e Prática*, 2 edn, Bookman.
- Jacobs, R., Jordan, M., Nowlan, S. and Hinton, G. (1991). Adaptive Mixture of Local Experts, *Neural Computation* **3**(1): 79–87.
- Leng, G., McGinnity, T. and Prasad, G. (2005). An approach for on-line extraction of fuzzy rules using a self-organising fuzzy neural network, *Fuzzy Sets and Systems* **150**(2): 211–243.
- Luna, I., Ballini, R. and Soares, S. (2006). Técnica de identificação de modelos lineares e não-lineares de séries temporais, *Sba: Controle & Automação Sociedade Brasileira de Automatica* **17**: 245 – 256.
- Luna, I., Soares, S. and Ballini, R. (2007). A Constructive-Fuzzy Rule Based System for Time Series Forecasting, *International Joint Conference on Neural Networks - IJCNN'07* .
- Ramamurti, V. and Ghosh, J. (1996). Structural Adaptation in Mixture of Experts, *13th International Conference on Pattern Recognition* **4**: 704–708.
- Takagi, T. and Sugeno, M. (1985). Fuzzy Identification of Systems and Its Applications to Modeling and Control, *IEEE Transactions on Systems, Man and Cybernetics* (1): 116–132.
- Yager, R. and Filev, D. (1994). *Essentials of Fuzzy Modeling and Control*, John Wiley and Sons. Inc.