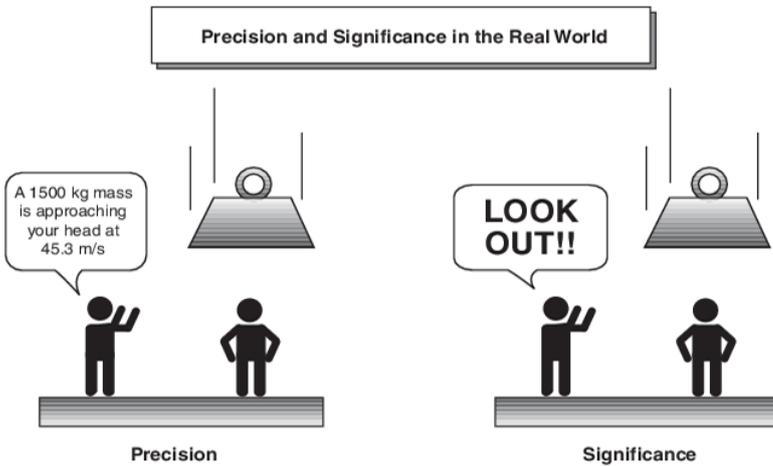




Nome: _____ Matrícula: _____

“As complexity rises, precise statements lose meaning and meaningful statements lose precision.” — Lotfi A. Zadeh
 “Rational thoughts never drive people's creativity the way emotions do.” — Neil deGrasse Tyson
 “Muitas vezes, problemas simples tem solução complexa e problemas complexos tem solução simples.”



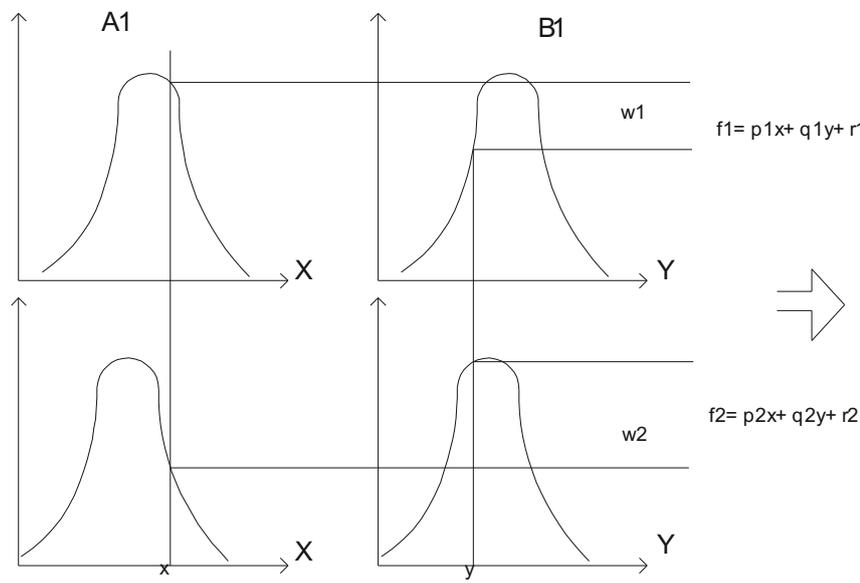
Google's DeepMind AI Just Taught Itself To Walk - ... youtube.com



mycomeup.com



R₁: Se x é A₁ e y é B₁, então f₁ = p₁x + q₁y + r₁
 R₂: Se x é A₂ e y é B₂, então f₂ = p₂x + q₂y + r₂



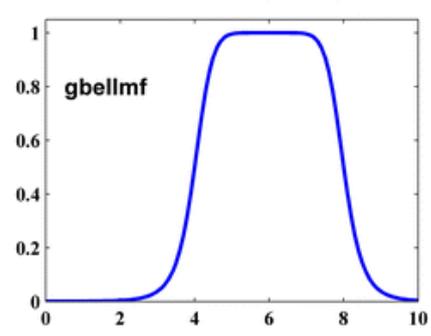
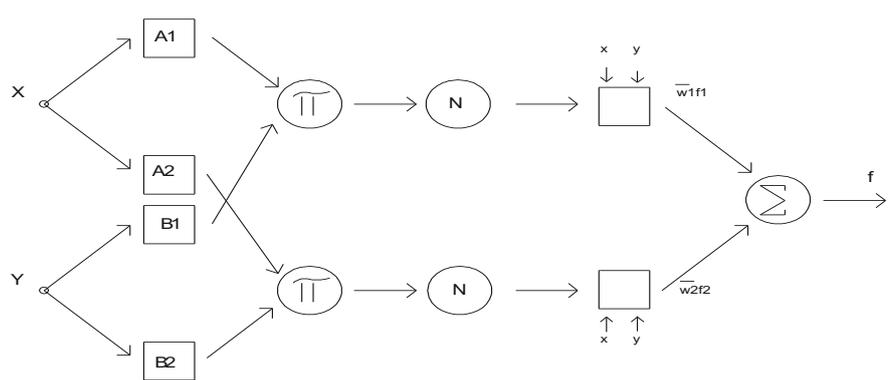
$$f_1 = p_1x + q_1y + r_1$$

$$f_2 = p_2x + q_2y + r_2$$

$$f = \frac{w_1f_1 + w_2f_2}{w_1 + w_2}$$

$$= \overline{w_1f_1} + \overline{w_2f_2}$$

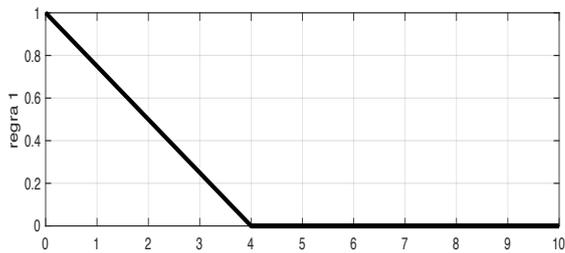
$$f(x; a, b, c) = \frac{1}{1 + \left| \frac{x-c}{a} \right|^{2b}}$$



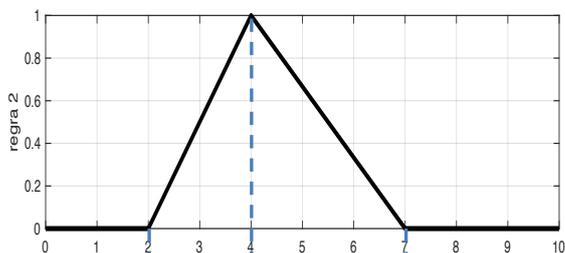
ANFIS □ → bloco adaptativo O → bloco fixo

2ª PROVA ICIN 1º/2018

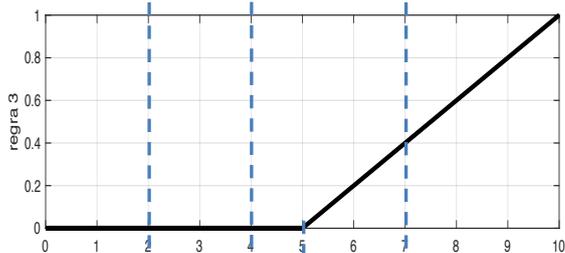
1ª Questão (2,0): Sistemas Fuzzy são uma forma de sintetizar uma superfície de controle não linear. Obtenha, para o caso unidimensional a seguir, a curva de controle do sistema de inferência *fuzzy* de Sugeno, $y = f(u)$.



$$f_1 = 3$$

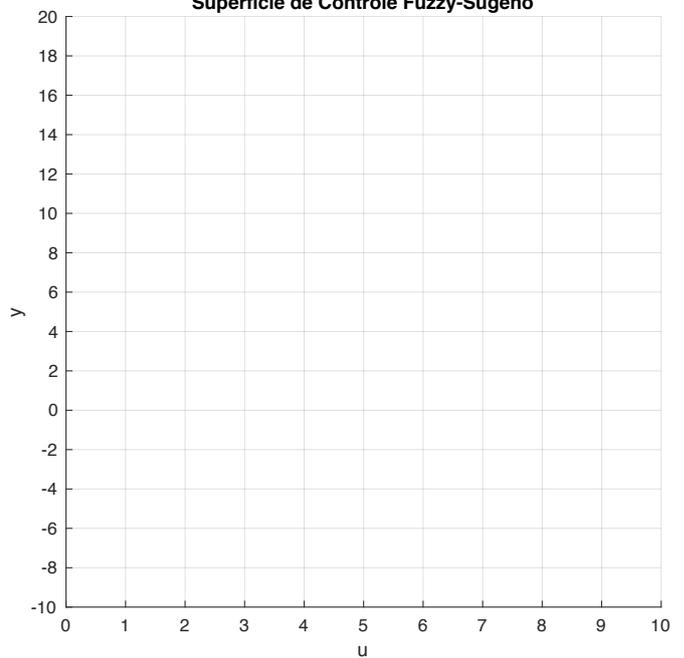


$$f_2 = -u - 1$$



$$f_3 = 2u$$

Superfície de Controle Fuzzy-Sugeno



A: $y = f_1 = 3$

B: $y = \frac{w_1 f_1 + w_2 f_2}{w_1 + w_2} = \frac{w_1(3) + w_2(-u-1)}{w_1 + w_2}$

$$w_1 = 1 - 0.25u; \quad w_2 = -1 + 0.5u;$$

$$y = \frac{3 - 0.75u + (-1 + 0.5u)(-u - 1)}{0.25u}$$

$$y = \frac{-0.5u^2 - 1.25u + 4}{0.25u}$$

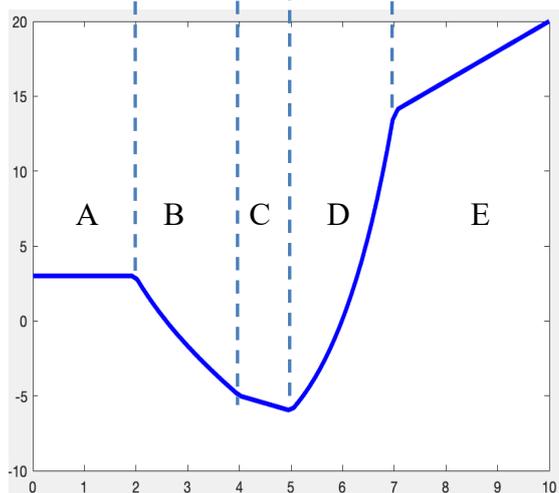
C: $y = f_2 = -u - 1$

D: $y = \frac{w_2 f_2 + w_3 f_3}{w_2 + w_3} = \frac{w_2(-u-1) + w_3(2u)}{w_2 + w_3}$

$$w_2 = 7/3 - u/3; \quad w_3 = -1 + 0.2u;$$

$$y = \frac{(7/3 - u/3)(-u - 1) + (-1 + 0.2u)(2u)}{7/3 - u/3 - 1 + 0.2u}$$

E: $y = f_3 = 2u$



2ª Questão (2,0) Assinale V – Verdadeiro ou F – Falso, justificando cada aspecto falso ou verdadeiro.

- a) A proposta da Lógica Fuzzy, proposta originalmente por Mamdani, é emular a psicologia humana, em que graus de pertinência substituem as verdades absolutas. Um paradoxo é representado pelo grau de pertinência 0,5. Um valor *fuzzy* de 0,4 seria “quase um paradoxo”. Algumas incongruências teóricas em relação à lógica de conjuntos clássica, no entanto, ainda precisam ser corrigidas.
- b) Uma máquina de lavar roupas *fuzzy* é mais cara do que uma máquina de lavar roupas convencional pois precisa mais sensores e de processamento adicional, reduz, porém, o consumo de energia (só lava até estar limpo, e não um tempo pré-programado). Pode-se afirmar que a mesma estratégia de economia de energia seria possível numa máquina de lavar comercial equipada com um sensor de sujeira (foto-emissor com foto-detector na água mediriam a turbidez, “grau-de-sujeira”) e uma “*look-up table*”.
- c) Os bonecos dinâmicos da empresa Deep Mind (Google) que aprendem a andar em labirintos são exemplos impressionantes de aprendizagem por reforço profundo. Um agente interage com o ambiente (segunda uma certa política) e obtêm recompensas ou punições imediatas e considera expectativas de recompensas acumuladas (futuras) para escolher a próxima ação. Tem-se uma sequência de “experimentos”: Estados → Ações → Recompensas → Estados, ..., que vão atualizando os valores de cada ação em cada estado. Pela Programação Dinâmica (otimização de processos decisórios discretos) de Bellman sabe-se que a solução ótima será encontrada.
- d) Aprendizado por reforço não pode, a rigor, ser considerado um treinamento supervisionado, pois a resposta desejada não é conhecida a priori. Para o problema do “swing-up” de um pêndulo sobre carro via aprendizado por reforço, por exemplo, são necessárias, muitas tentativas até que se obtenha sucesso. Como temos quatro variáveis de estado (posição e velocidade do carro, posição e velocidade angular da haste) é necessário usar aprendizagem por reforço aproximado (Rede Neural Artificial em lugar da tabela de valores).

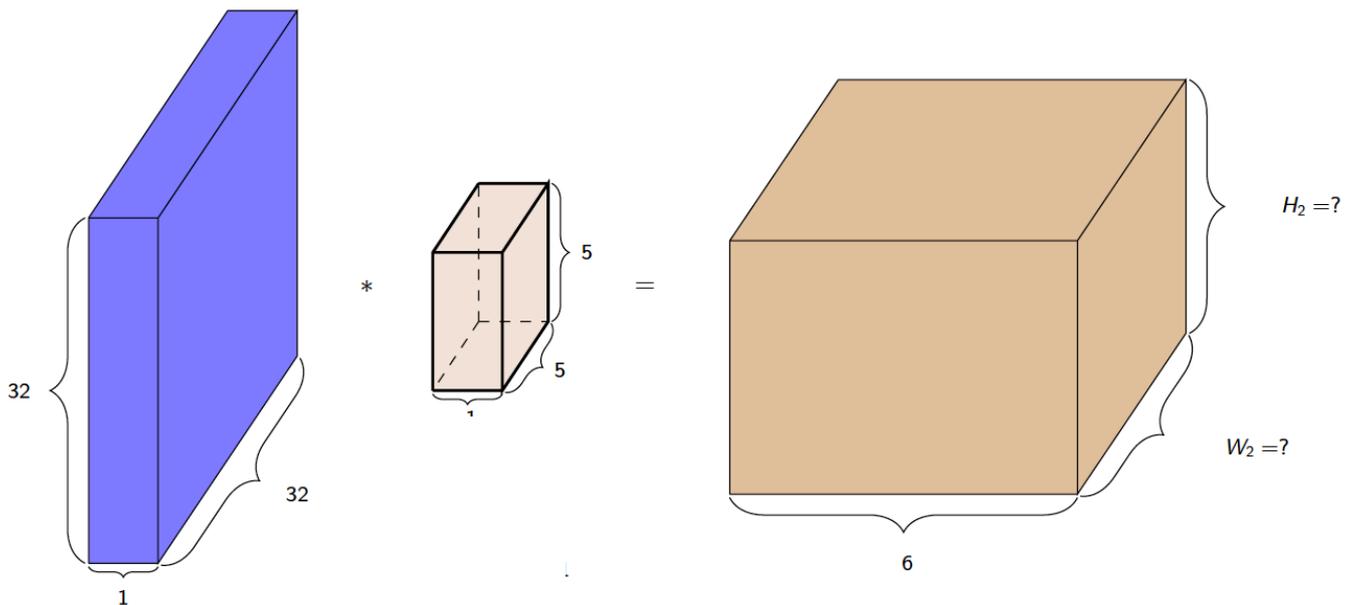
- a) F – Lógica fuzzy foi proposta originalmente por Zadeh. As “incongruências” em relação à lógica clássica são aceitas como compromisso de uma lógica multi-valorada e não precisam ser corrigidas.
- b) V – a implementação da lógica fuzzy pode ser feita em uma tabela “*look-up-table*”.
- c) F – não existe garantia de que “o ótimo” será encontrado. O princípio de Bellman é utilizado para reduzir o espaço de busca.
- d) F, V – Aprendizado por reforço aproximado “pode” ser utilizado, mas ainda seria possível utilizar uma tabela para armazenar os valores associados a cada estado discretizado.

3ª Questão (2,0) Considere uma camada convolucional de uma rede neural profunda. Observe que, sem preenchimento, alguns pixels não são mapeados pela máscara na imagem de saída.

a) Obtenha, em função de W_1 , W_2 , F , P e S , equações para as dimensões de saída (H_2 , W_2).

- Imagem de entrada: $H_1 \times W_1$
- Máscara (Filtro): $F \times F$
- Quantidade de Filtros: K
- Preenchimento (Padding): P
- Passo (Stride): S

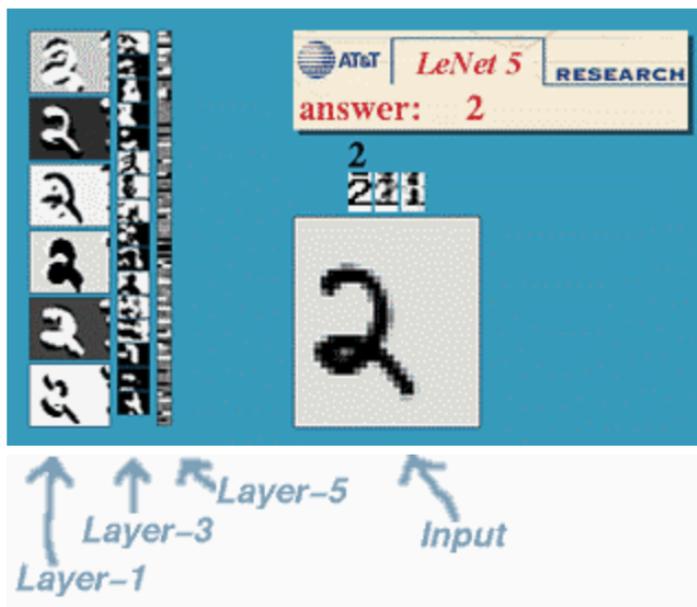
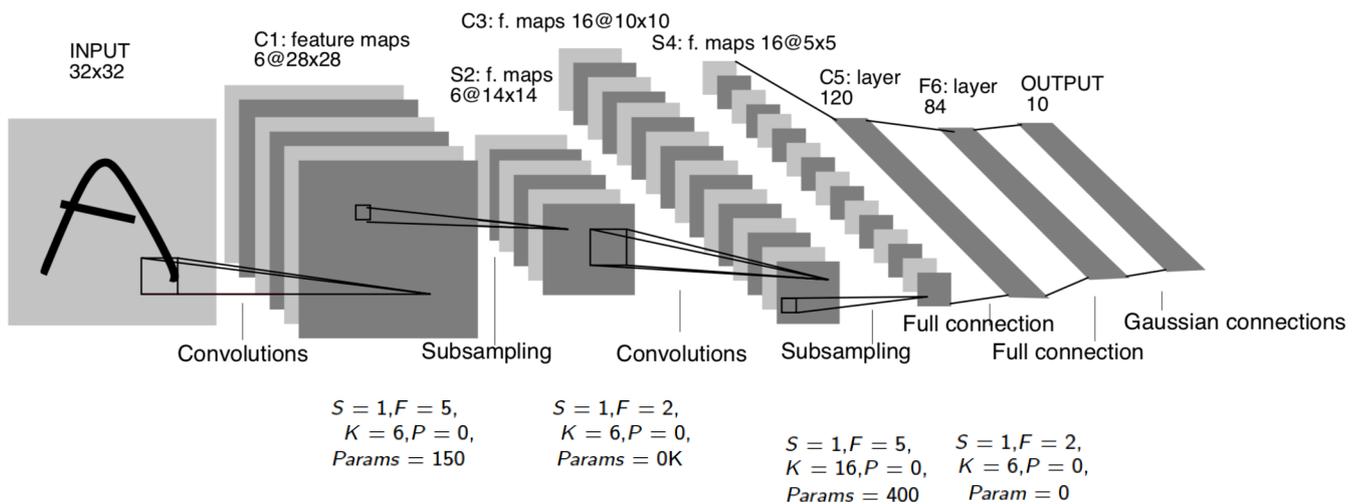
b) Obtenha agora (H_2 , W_2) para uma imagem de entrada de 32×32 , $F = 5$, $K = 6$, $P = 0$ e $S = 1$.



a) $W_2 = \frac{W_1 - F + 2P}{S} + 1$; $H_2 = \frac{H_1 - F + 2P}{S} + 1$

b) 28×28 (também disponível na questão 4)

4ª Questão (2,0) Considere a rede neural convolucional (CNN) LeNet-5 (LeCunn et al, Gradient-based learning applied to document recognition, Proceedings IEEE, Nov. 1998).



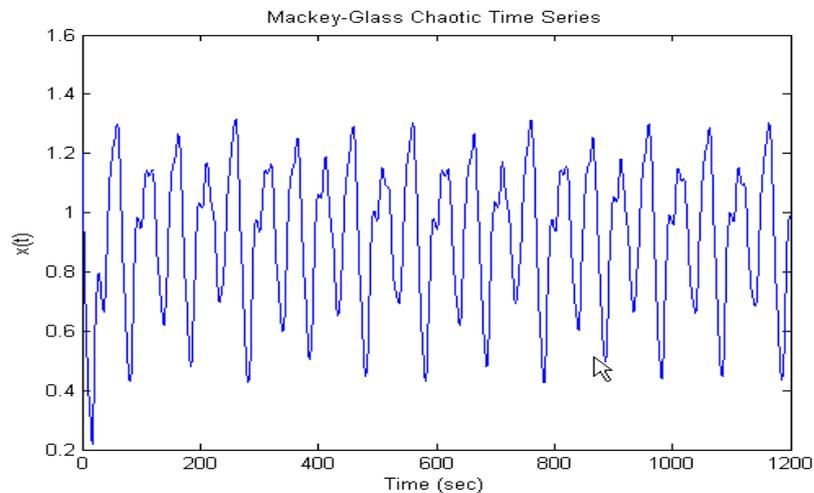
- (1,0) Uma forma de se reduzir o número de parâmetros a serem treinados em camadas CNN é conhecido por compartilhamento de pesos (“weight sharing”). O que significa isto e por quê funciona (em comparação a uma camada densa).
- (0,5) Qual a finalidade dos mapas de características (feature maps)?
- (0,5) Apresente os parâmetros utilizados em “sua” rede CNN do exercício 3 de ICIN, justificando a escolha dos parâmetros.

- Numa camada de convolução o mesmo filtro (com os mesmos pesos) é aplicado ao longo de toda a imagem de entrada. Se o objetivo é detectar linhas horizontais na imagem (“feature” – padrão na imagem), quer se detectar linha horizontais em qualquer posição na imagem. Não é necessário um filtro diferente para diferentes regiões na imagem.

Uma camada densa tem pesos individualizados, é como se a máscara (filtro) tivesse a dimensão da imagem de entrada. Uma camada convolucional utiliza vários filtro, cada um para uma característica que se queira extrair.
- É mais fácil reconhecer objetos a partir de suas características. Se, por exemplo, tiver rodas pode ser um carro. Se tiver nariz, pode ser uma pessoa. Se tiver asas, pode ser avião. A maioria dos carros/pessoas/aviões compartilha características comuns, a Rede Neural Convolucional (“Profunda”) procura estas características em imagens com “invariância a posição”, invariância a tamanho, invariância a cor, invariância a rotação etc. As “features” são mais fáceis de reconhecer e os objetos são classificados pelas “features” que tem em comum.
-

5ª Questão (2,0): Um sistema ANFIS – Sugeno de 1ª ordem (linear) deve ser utilizado para extrapolar uma série temporal caótica.

$$\frac{dx(t)}{dt} = \frac{0,2x(t-\tau)}{1+x(t-\tau)^{10}} - 0,1x(t) \quad , \text{ com } x(0)=1,2 \text{ e } \tau = 17.$$



Isto é, a função f deve ser aprendida a partir de certas amostras passadas: $x(t+6) = f(x(t), x(t-5), x(t-10), x(t-20))$

Cada sinal de entrada dos ANFIS pode ser representado por duas funções de pertinência do tipo “gbellmf”.

- Quantos parâmetros das premissas e quantos das conseqüências precisam ser adaptados?
- Descreva de forma sucinta o treinamento ANFIS (antecedentes e consequentes).

a) O sistema Sugeno de 1ª ordem tem 4 entradas e uma saída.

Como cada entrada é descrita por duas funções de pertinência “gbellmf”, temos $3 \times 2 \times 4 = 24$ parâmetros das premissas (3 parâmetros gbell x 2 funções x 4 entradas)

As regras são do tipo:

R1: se x_t é A1 e x_{t5} é B1 e x_{t10} é C1 e x_{t20} é D1 então $x_{t+6} = p1 \cdot x_t + q1 \cdot x_{t5} + r1 \cdot x_{t10} + s1 \cdot x_{t20} + o1$

R2: se x_t é A2 e x_{t5} é B1 e x_{t10} é C1 e x_{t20} é D1 então $x_{t+6} = p2 \cdot x_t + q2 \cdot x_{t5} + r2 \cdot x_{t10} + s2 \cdot x_{t20} + o2$

R3: se x_t é A1 e x_{t5} é B2 e x_{t10} é C1 e x_{t20} é D1 então $x_{t+6} = p3 \cdot x_t + q3 \cdot x_{t5} + r3 \cdot x_{t10} + s3 \cdot x_{t20} + o3$

...

R16: se x_t é A2 e x_{t5} é B2 e x_{t10} é C2 e x_{t20} é D2 então $x_{t+6} = p16 \cdot x_t + q16 \cdot x_{t5} + r16 \cdot x_{t10} + s16 \cdot x_{t20} + o16$

Existem portanto 16 regras a 5 parâmetros das conseqüências → 80 parâmetros das conseqüências.

b) O treinamento é feito em duas etapas: Feedforward e Backward

- No passo feedforward fixam-se os parâmetros das premissas e obtêm-se a saída da rede em função de (p_i, q_i, r_i, s_i) . A saída é combinação linear destes parâmetros. É feito um ajuste destes parâmetros pelo método dos mínimos quadrados.

- No passo backward é feita a retropropagação do erro para ajustar pelo método do gradiente descendente, os parâmetros das premissas. Nesta etapa os parâmetros das conseqüências são mantidos fixos.

Obs: O cálculo do sinal u no modelo de Sugeno é feito por

Grau de compatibilidade de cada regra: $h_i(x_t, x_{t5}, x_{t10}, x_{t20}) = \mu_{F_{x_t}}^i(x_t) \wedge \mu_{F_{x_{t5}}}^i(x_{t5}) \wedge \mu_{F_{x_{t10}}}^i(x_{t10}) \wedge \mu_{F_{x_{t20}}}^i(x_{t20})$

Normalização da contribuição de cada regra ativa: $w_i(x_t, x_{t5}, x_{t10}, x_{t20}) = \frac{h_i(x_t, x_{t5}, x_{t10}, x_{t20})}{\sum_{i=1}^{16} h_i(x_t, x_{t5}, x_{t10}, x_{t20})}$

Média ponderada das regras ativas normalizadas:

$$x_{t+6} = \sum_{i=1}^{16} w_i(x_t, x_{t5}, x_{t10}, x_{t20}) (p_i x_t + q_i x_{t5} + r_i x_{t10} + s_i x_{t20} + o_i)$$