Classificação Fuzzy: Método de Treinamento Wang-Mendel

Classificação

Um classificador é um algoritmo (sistema) que associa um rótulo de classe a um objeto com base na descrição deste objeto.

IF $X_1 = v_1^1$ AND $X_2 \geq v_2^1$ THEN class is 1
IF $X_1 \leq v_1^2$ AND $X_2 = v_2^2$ THEN class is 2
IF $X_1 = v_1^3$ AND $X_2 = v_2^3$ THEN class is 2
IF $X_1 \neq v_1^4$ AND $X_2 \leq v_2^4$ THEN class is 3

- É comum dizer que o classificador prediz o rótulo da classe.

Classificação

Um classificador é um algoritmo (sistema) que associa um rótulo de classe a um objeto com base na descrição deste objeto.

IF $X_1 = v_1^1$ AND $X_2 \geq v_2^1$ THEN class is 1
IF $X_1 \leq v_1^2$ AND $X_2 = v_2^2$ THEN class is 2
IF $X_1 = v_1^3$ AND $X_2 = v_2^3$ THEN class is 2
IF $X_1 \neq v_1^4$ AND $X_2 \leq v_2^4$ THEN class is 3

- É comum dizer que o classificador prediz o rótulo da classe.

Classificação: Crisp x Fuzzy

- Classificador tradicional:

IF $X_1 = v_1^1$ AND $X_2 \geq v_2^1$ THEN class is 1
IF $X_1 \leq v_1^2$ AND $X_2 = v_2^2$ THEN class is 3
IF $X_1 = v_1^3$ AND $X_2 = v_2^3$ THEN class is 2

- Classificadores fuzzy:

IF X_1 is medium AND X_2 is small THEN class is 1
IF X_1 is medium AND X_2 is large THEN class is 2
IF X_1 is large AND X_2 is small THEN class is 3

Classificação

Considerando como entrada um objeto-desconhecido(v_1, v_2) a classe deste objeto seria, por exemplo:

- classif. tradicional: classe 2, se $v_1 = v_1^3$ e $v_2 = v_2^3$
- classif. fuzzy: classe 1 com grau μ_1 e classe 2 com grau μ_2

Classificadores: Aprendizado Supervisionado

A construção de um procedimento de classificação a partir de um conjunto de dados para os quais a classe é conhecida recebe o nome de

reconhecimento de padões,

discriminação,

aprendizado supervisionado de classificadores

(de forma a distinguir do processo de aprendizado não supervisionado ou agrupamento - clustering – casos para os quais a classe não é conhecida a priori e deve ser inferida a partir dos dados).

Classificador Fuzzy

A estrutura básica de um classificador fuzzy possui três componentes conceituais:

Base de dados (BD): Partição do Universo

Base de regras (BR): Conjunto de regras

Mecanismo de raciocínio: Inferência

Operadores (mais simples do que o SIF):

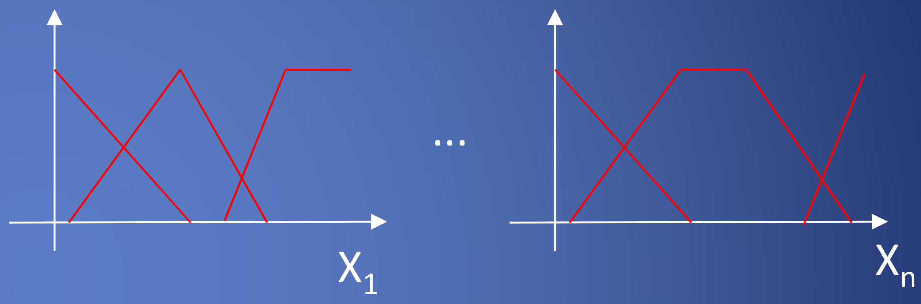
agregação de antecedentes

agregação das regras,

tomada de decisão (max ou max_soma).

Classificador Fuzzy: Aprendizado

Base de dados (BD): Partição do Universo

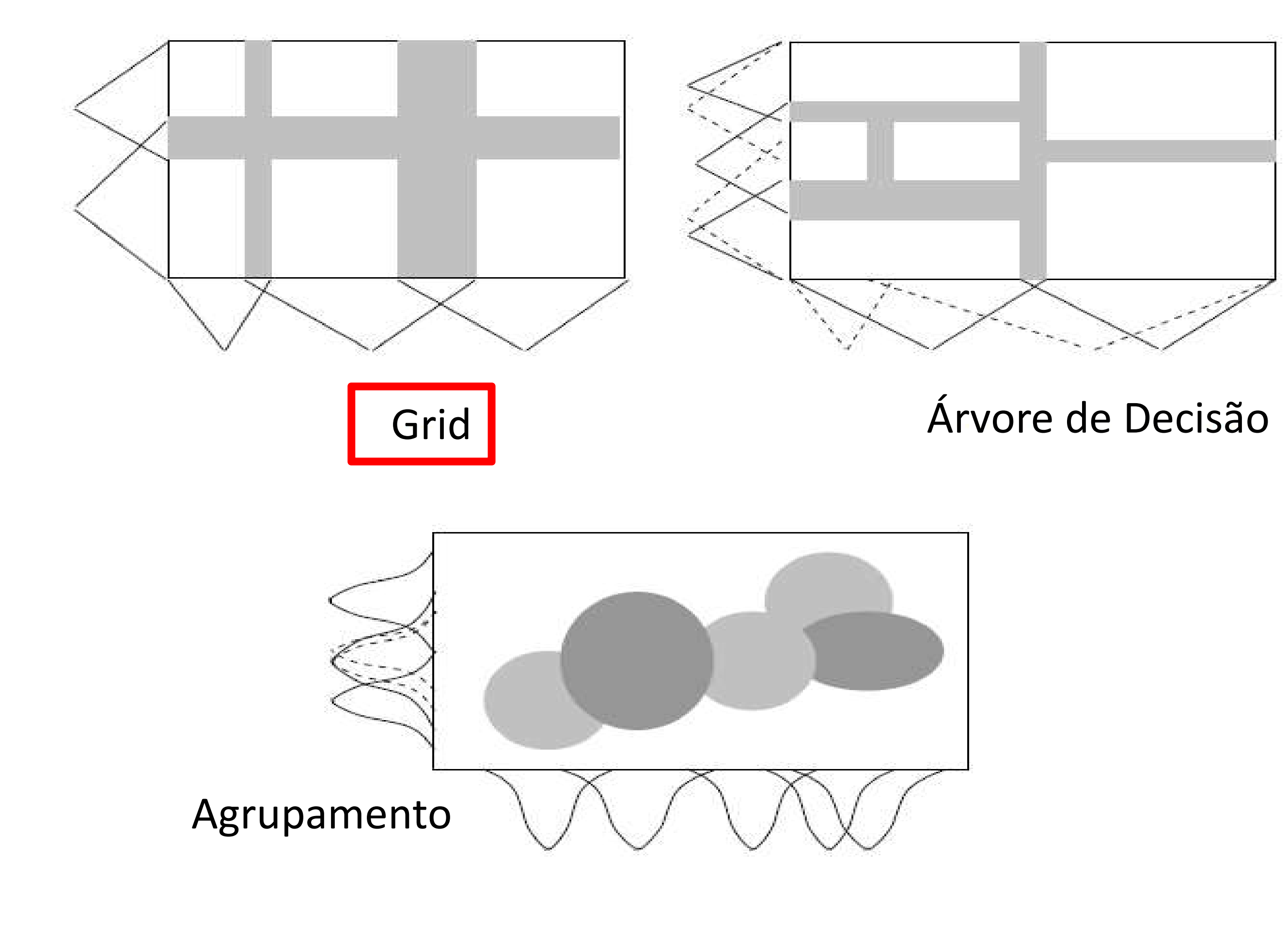


Base de regras (BR): Conjunto de regras

Mecanismo de raciocínio: Inferência

Partição Fuzzy do Universo

Diferentes métodos de partição

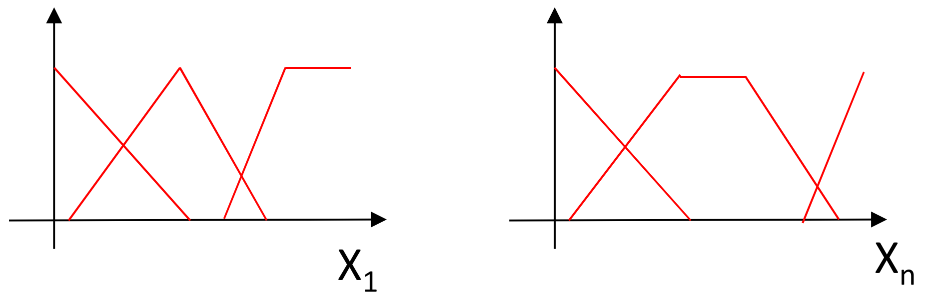


Partição por Grid

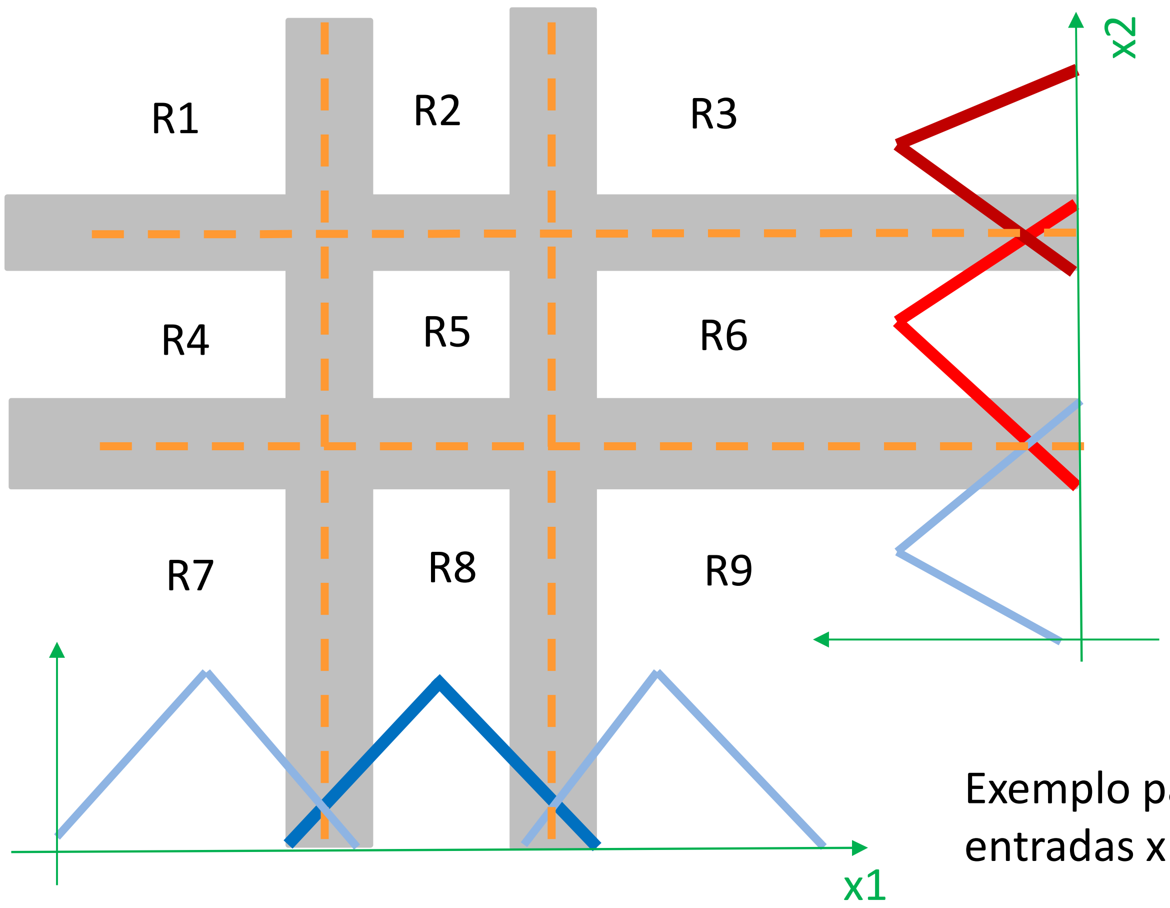
A maior vantagem desta estratégia é a simplicidade.

O especialista fixa o total, formato e localização dos conjuntos fuzzy.

As funções de pertinência formam então um grid no espaço de entrada e o total de regras é função do total de conjuntos fuzzy.



Partição por Grid



Exemplo para duas entradas x1 e x2

Partição por Grid

Partição por grid:

Uniforme: usada quando o especialista não possui conhecimento sobre regiões mais importanes do espaço de entrada.

Não uniforme: usada quando o especilista ou um método automático de ajuste foca o particionamento em regiões específicas do espaço de entrada.

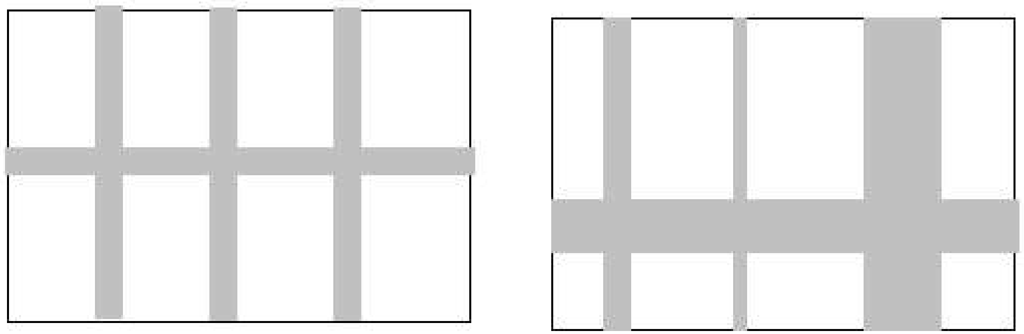
Partição por Grid

Partição por grid:

Uniforme: usada quando o especialista não possui conhecimento sobre regiões mais importanes do espaço de entrada.

Não uniforme: usada quando o especilista ou um método automático de ajuste foca o particionamento em regiões específicas do espaço de entrada.

Ajuste de partição:



Inicial(uniforme) Final (Não-uniforme)

Partição por Árvores de Decisão

Partição por árvores de decisão:

Um método de partição por árvores de decisão bastante flexível é o fuzzy k-d-trees. Este método resulta de uma série de cortes de guilhotinas (corte que é feito ao longo de todo espaço a ser particionado) e cada região produzida é então submetida ao corte novamente.

Partição por Árvores de Decisão

Partição por árvores de decisão:

Um método de partição por árvores de decisão bastante flexível é o fuzzy k-d-trees. Este método resulta de uma série de cortes de guilhotinas (corte que é feito ao longo de todo espaço a ser particionado) e cada região produzida é então submetida ao corte novamente.

Em cada passo do algoritmo, algumas decisões devem ser tomadas:

1)qual a dimensão a ser cortada ?

2) em que posição realizar o corte.

Partição por Árvores de Decisão

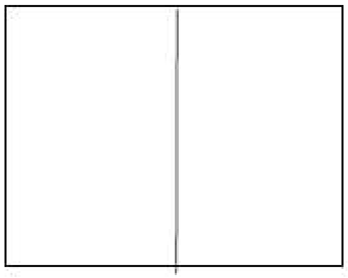
Partição por árvores de decisão:

Um método de partição por árvores de decisão bastante flexível é o fuzzy k-d-trees. Este método resulta de uma série de cortes de guilhotinas (corte que é feito ao longo de todo espaço a ser particionado) e cada região produzida é então submetida ao corte novamente.

Em cada passo do algoritmo, algumas decisões devem ser tomadas:

1)qual a dimensão a ser cortada ?

2) em que posição realizar o corte.



Partição por Árvores de Decisão

Partição por árvores de decisão:

Um método de partição por árvores de decisão bastante flexível é o fuzzy k-d-trees. Este método resulta de uma série de cortes de guilhotinas (corte que é feito ao longo de todo espaço a ser particionado) e cada região produzida é então submetida ao corte novamente.

Em cada passo do algoritmo, algumas decisões devem ser tomadas:

1)qual a dimensão a ser cortada ?

2) em que posição realizar o corte.

Partição por Árvores de Decisão

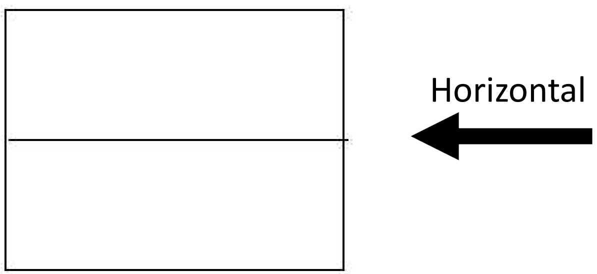
Partição por árvores de decisão:

Um método de partição por árvores de decisão bastante flexível é o fuzzy k-d-trees. Este método resulta de uma série de cortes de guilhotinas (corte que é feito ao longo de todo espaço a ser particionado) e cada região produzida é então submetida ao corte novamente.

Em cada passo do algoritmo, algumas decisões devem ser tomadas:

1)qual a dimensão a ser cortada

2) em que posição realizar o corte.



Partição por Árvores de Decisão

Partição por árvores de decisão:

Um método de partição por árvores de decisão bastante flexível é o fuzzy k-d-trees. Este método resulta de uma série de cortes de guilhotinas (corte que é feito ao longo de todo espaço a ser particionado) e cada região produzida é então submetida ao corte novamente.

Em cada passo do algoritmo, algumas decisões devem ser tomadas:

1)qual a dimensão a ser cortada

2) em que posição realizar o corte. ?

Partição por Árvores de Decisão

Partição por árvores de decisão:

Um método de partição por árvores de decisão bastante flexível é o fuzzy k-d-trees. Este método resulta de uma série de cortes de guilhotinas (corte que é feito ao longo de todo espaço a ser particionado) e cada região produzida é então submetida ao corte novamente.

Em cada passo do algoritmo, algumas decisões devem ser tomadas:

1)qual a dimensão a ser cortada

2) em que posição realizar o corte. ?

Partição por Árvores de Decisão

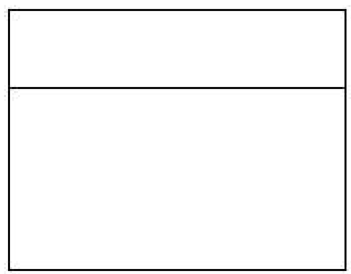
Partição por árvores de decisão:

Um método de partição por árvores de decisão bastante flexível é o fuzzy k-d-trees. Este método resulta de uma série de cortes de guilhotinas (corte que é feito ao longo de todo espaço a ser particionado) e cada região produzida é então submetida ao corte novamente.

Em cada passo do algoritmo, algumas decisões devem ser tomadas:

1)qual a dimensão a ser cortada

2) em que posição realizar o corte. ?



Partição por Árvores de Decisão

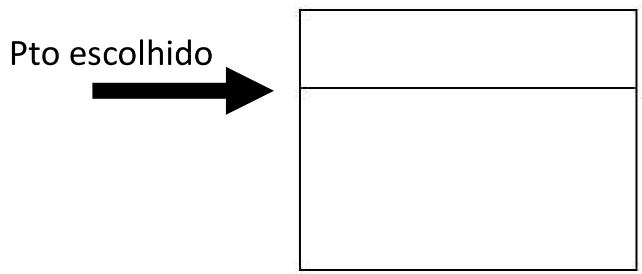
Partição por árvores de decisão:

Um método de partição por árvores de decisão bastante flexível é o fuzzy k-d-trees. Este método resulta de uma série de cortes de guilhotinas (corte que é feito ao longo de todo espaço a ser particionado) e cada região produzida é então submetida ao corte novamente.

Em cada passo do algoritmo, algumas decisões devem ser tomadas:

1)qual a dimensão a ser cortada

2) em que posição realizar o corte.



Partição por Árvores de Decisão

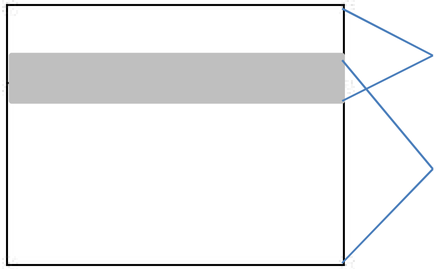
Partição por árvores de decisão:

Um método de partição por árvores de decisão bastante flexível é o fuzzy k-d-trees. Este método resulta de uma série de cortes de guilhotinas (corte que é feito ao longo de todo espaço a ser particionado) e cada região produzida é então submetida ao corte novamente.

Em cada passo do algoritmo, algumas decisões devem ser tomadas:

1)qual a dimensão a ser cortada

2) em que posição realizar o corte.

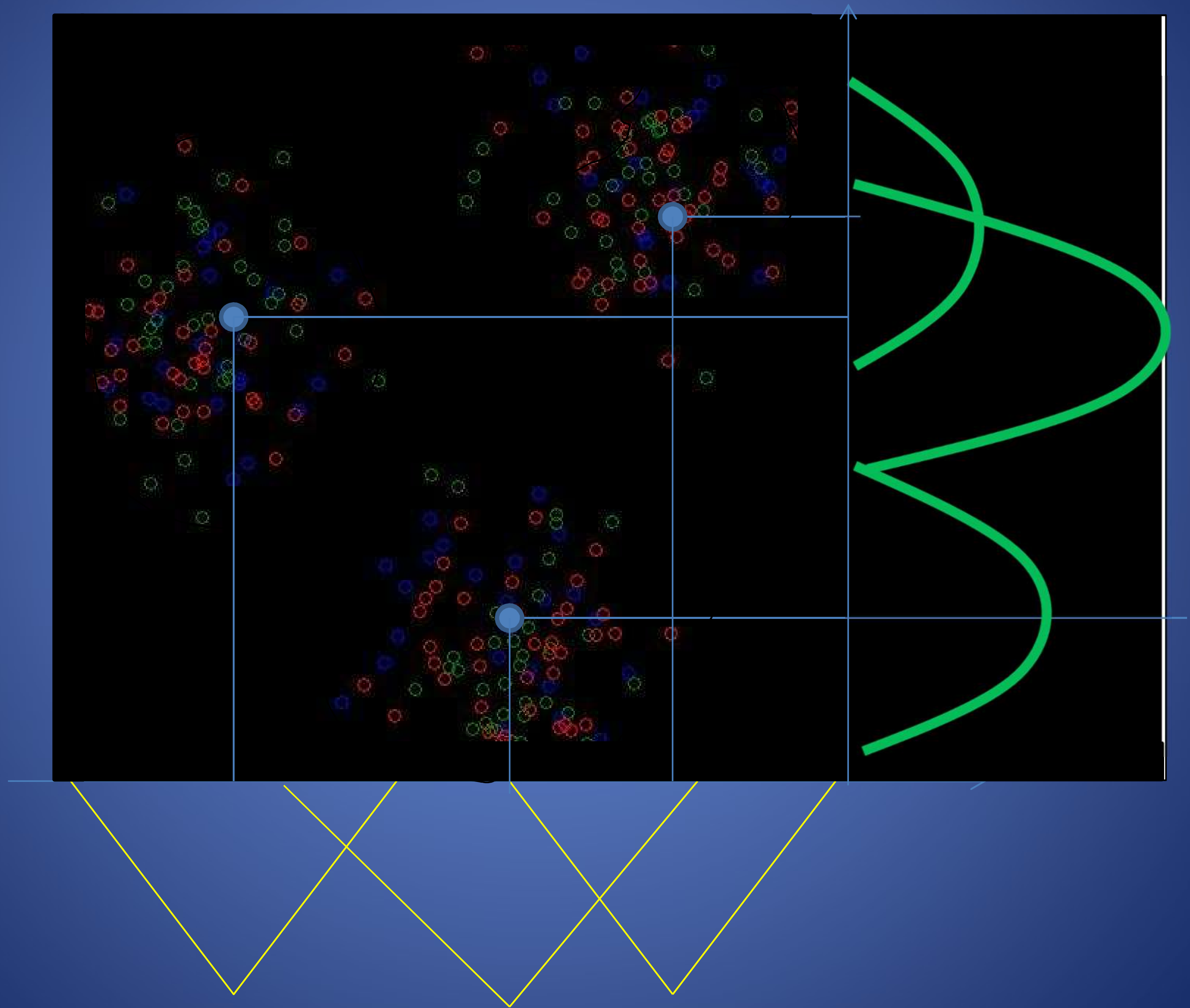


Partição por Agrupamento

Partição por agrupamento (clustering partition):

As partições por agrupamento dividem os dados em vários grupos (clusters) tal que a similaridade dentro de um grupo é maior do que entre os grupos.

Partição por Agrupamento



Partição por Agrupamento

Existem diferentes métodos de agrupamente, entre os quais se destacam:

- Agrupamento hard-c-means (ou k-means);
- Agrupamento fuzzy-c-means (ou c-means).

Partição crisp : clustering – k-means

Algoritmo k-means:

<u>Passo 1</u>: Inicializar o centro do cluster $\mathbf{c}_j$, j = *1* ... *k*. Tipicamente, selecionam-se *k* pontos aleatoriamente entre todos pontos disponíveis.

$$U = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 1 \\ 0 & 1 & \cdots & 0 \\ 1 & 0 & \cdots & 0 \end{bmatrix}$$

<u>Passo 2</u>: Determina-se a matriz de pertinência *U(n* x *k)* (eq. 2).

$$u_{ij} = \begin{cases} 1 & \text{se } \left\|\mathbf{x}_i - \mathbf{c}_i\right\|_2 \leq \left\|\mathbf{x}_j - \mathbf{c}_i\right\|_2, \text{para cada } 1 \neq \text{i} \\ 0 & \text{caso contrário} \end{cases} \qquad \text{(eq 2)}$$

<u>Passo 3</u>: Calcula-se a função custo *J* de acordo com (eq. 1). Parar se a função ficar abaixo de um certo limiar de tolerância ou se as melhorias em relação à iteração anterior ficarem abaixo de um certo limiar.

$$J = \sum_{i=1}^{n}\left(\sum_{j=1}^{k}\left\|\mathbf{x}_i - \mathbf{c}_j\right\|_2\right) \qquad \text{(eq. 1)}$$

<u>Passo 4</u>: Atualizar os centros dos clusters de acordo com a (eq. 3) e ir para o passo 2.

$$\mathbf{c}_i = \frac{1}{\left|G_i\right|}\sum_{j,\mathbf{x}_j \in G_i}\mathbf{x}_j \qquad \text{(eq. 3)}$$

Partição fuzzy : clustering - cmeans

Passo 1: Inicializar a matriz U com valores aleatórios em [0, 1] tal que as restrições em (2b) sejam atendidas.

Passo 2: Calcular os centros dos fuzzy clusters $\mathbf{c}_j$, j = *1* ... *c*. usando (eq 3b).

$$\mathbf{c}_j = \frac{\sum_{i=1}^{n} u_{ij}^{m} \mathbf{x}_i}{\sum_{i=1}^{n} u_{ij}^{m}}$$

Passo 3: Calcular a função de custo (eq 1b). Condição de Parada: idêntica à do k-means.

$$J = \sum_{i=1}^{n} \left(\sum_{j=1}^{c} u_{ij}^{m} \left\| \mathbf{x}_i - \mathbf{c}_j \right\|_2 \right), \qquad 1 < m < \infty$$

Passo 4: Calcular a nova matriz U usando a (eq. 4b). Ir para o passo 2.

$$u_{ij} = \frac{1}{\sum_{i=1}^{n} \left(\frac{d_{ij}}{d_{kj}} \right)^{2/(m-1)}}$$

A matriz U permite valores $u_{ij} \in [0, 1]$ com restrições:

$$\sum_{j=1}^{k} u_{ij} = 1, \forall i = 1, \cdots, n \quad \text{(pto pode} \in \text{ a mais de um cluster)}$$

$$0 < \sum_{j=1}^{k} \sum_{i=1}^{n} u_{ij} < n, \quad \text{(cada ponto } \in \text{ a pelo menos um cluster)} \qquad \text{(eq 2b)}$$

Classificador Fuzzy

Base de dados (BD): Partição do Universo (sobreposição entre os conjuntos)

Base de regras (BR): Conjunto de regras

Mecanismo de raciocínio: Inferência

TABELA 2. Classificação de conforto térmico em função das variáveis preditoras. **Thermal comfort rating according to the predicted variables.**

Regra	Temp. Superficial das Penas	Temp. Superficial da Pele	Empenamento	Conforto Térmico (peso)
1	Alta	Alta	Alta	Perigo (1,0)
2	Alta	Alta	Média	Perigo (0,75)
3	Alta	Alta	Baixa	Perigo (0,5)
4	Alta	Média	Alta	Perigo (0,5)
5	Alta	Média	Média	Alerta (1,0)
6	Alta	Média	Baixa	Alerta (0,75)
7	Alta	Baixa	Alta	Alerta (0,75)
8	Alta	Baixa	Média	Alerta (0,5)
9	Alta	Baixa	Baixa	Conforto (0,5)
10	Média	Alta	Alta	Perigo (1,0)
11	Média	Alta	Média	Perigo (0,5)
12	Média	Alta	Baixa	Alerta (1,0)
13	Média	Média	Alta	Alerta (0,75)
14	Média	Média	Média	Conforto (0,5)
15	Média	Média	Baixa	Conforto (0,75)
16	Média	Baixa	Alta	Alerta (0,5)
17	Média	Baixa	Média	Conforto (0,75)
18	Média	Baixa	Baixa	Conforto (1,0)
19	Baixa	Alta	Alta	Alerta (1,0)
20	Baixa	Alta	Média	Alerta (0,75)
21	Baixa	Alta	Baixa	Alerta (0,5)
22	Baixa	Média	Alta	Conforto (0,75)
23	Baixa	Média	Média	Conforto (1,0)
24	Baixa	Média	Baixa	Conforto (0,75)
25	Baixa	Baixa	Alta	Conforto (1,0)
26	Baixa	Baixa	Média	Conforto (0,75)
27	Baixa	Baixa	Baixa	Conforto (0,5)

Classificador Fuzzy

Base de dados (BD): Partição do Universo (sobreposição entre os conjuntos)

Base de regras (BR): Conjunto de regras

Especialista

Automática

WangMendel (mais simples)

Mecanismo de raciocínio: Inferência

Classificador Fuzzy

Base de dados (BD): Partição do Universo (grid ou c-means – há sobreposição)

Base de regras (BR): Conjunto de regras

Especialista

Automática

Mecanismo de raciocínio: Inferência

Operadores:

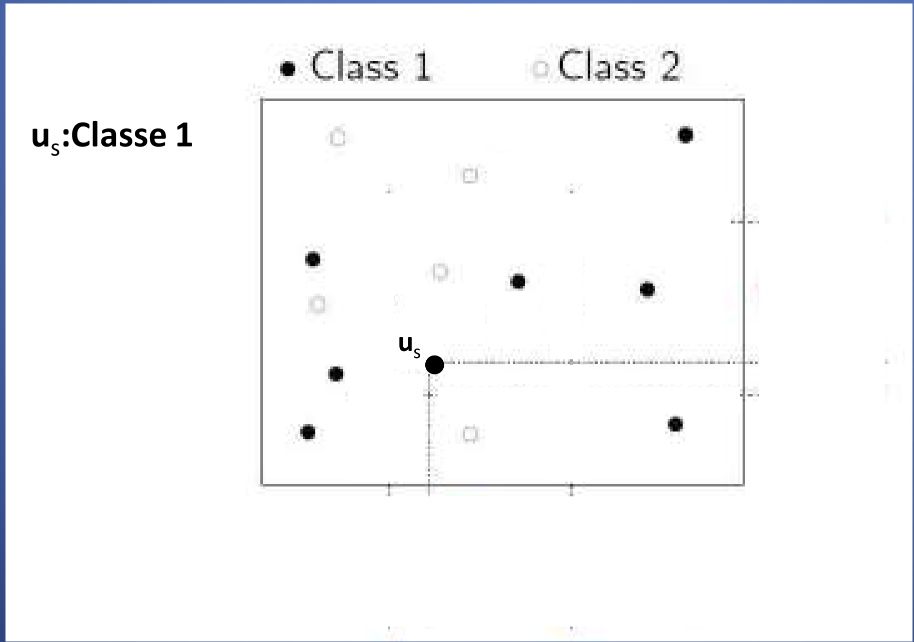
agregação de antecedentes (min ou produto)

agregação das regras (max),

tomada de decisão (max ou soma).

Método de Inferência

Dado um conjunto de teste S, consistindo de padrões $(\mathbf{u}_s, y_s)$, $s = 1, ..., |S|$ onde $\mathbf{u}_s = (u_1, ..., u_n)$ é o vetor de entradas e y_s é a saída.



Exemplo

n=2 (dimensões)

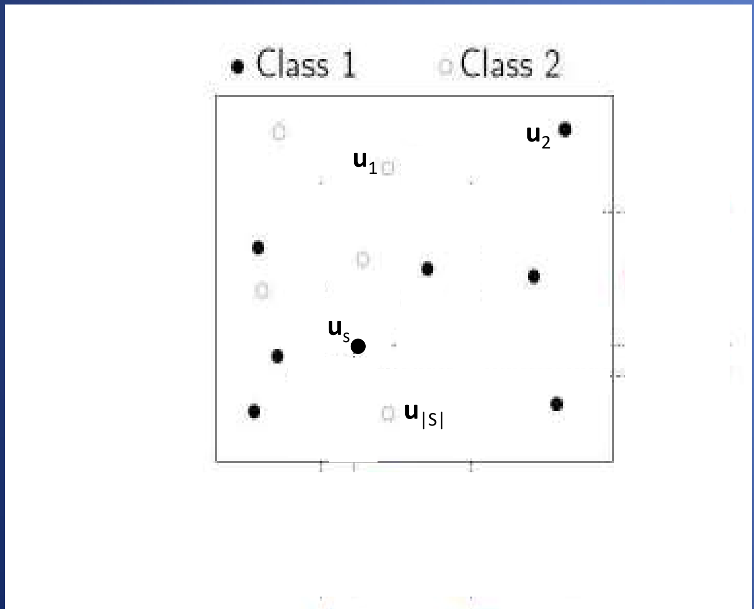
|S|=12 dados de teste

$(\mathbf{u}_s, y_s)$ dado de teste

y_s={classe1,class2}

Método de Inferência

Dado um conjunto de teste S, consistindo de padrões $(\mathbf{u}_s, y_s)$, $s = 1, ..., |S|$ onde $\mathbf{u}_s = (u_1, ..., u_n)$ é o vetor de entradas e y_s é a saída. Dada uma base de regras $RB = \{R^1, R^2, ..., R^j, ...R^m\}$ obtida pelo metodo WM ou pelo especialista, onde R^j : If X_1 is A_1^j and X_2 is A_2^j and ... and X_n is A_n^j then y is B^j. Cada padrão é apresentado à base RB de forma a se obter uma saída inferida $\hat{y}_s$.



Classificador Fuzzy

Base De Regras Fuzzy

$\hat{y}_1$

$\hat{y}_2$

.
.
.

$\hat{y}_s$

.
.
.

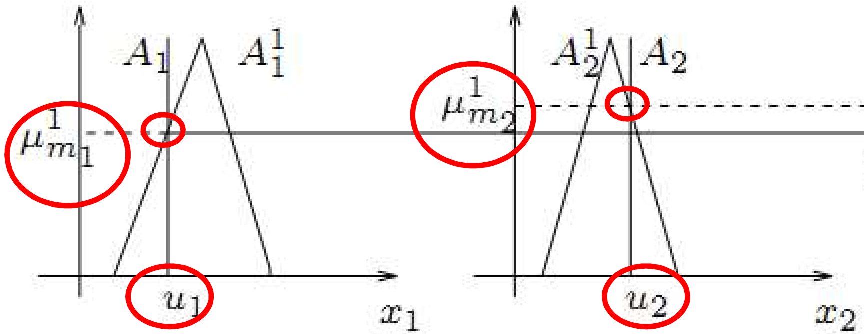
$\hat{y}_{|S|}$

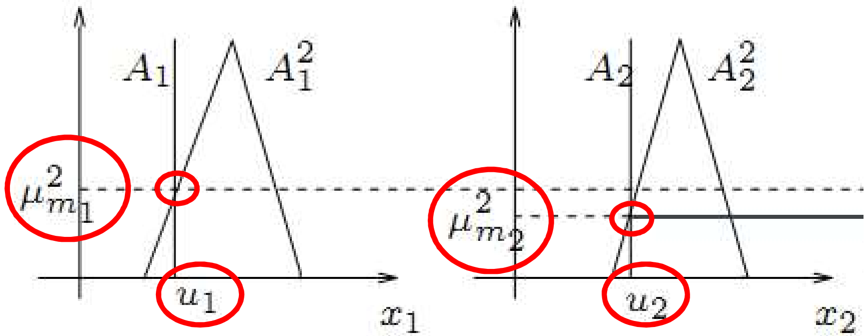
Método de Inferência

Dado um conjunto de teste S, consistindo de padrões $(\mathbf{u}_s, y_s)$, $s = 1, ..., |S|$ onde $\mathbf{u}_s = (u_1, ..., u_n)$ é o vetor de entradas e y_s é a saída. Dada uma base de regras $RB = \{R^1, R^2, ..., R^j, ...R^m\}$ obtida pelo metodo WM ou pelo especialista, onde R^j : If X_1 is A_1^j and X_2 is A_2^j and ... and X_n is A_n^j then y is B^j. Cada padrão é apresentado à base RB de forma a se obter uma saída inferida $\hat{y}_s$. O primeiro passo consiste em se obter o matching da entrada $\mathbf{u}_s = (u_1, ..., u_n)$ com a regra R^j. O matching da v-ésima variável do padrão de entrada s é dado por $\mu^j_{s_v} = \mu_{A_v^j}(u_v)$ onde $\mu_{A_v^j}(u_v)$ é o grau de pertinência para u_v na função de pertinência associada com o termo linguístico l_v definido para X_v na regra R^j. O segundo passo define o nível de disparo da regra a partir da apresentação do padrão de entrada $(\mu^j(\mathbf{u}_s) = \mu_{A_1^j}(u_1)\ \mathbf{t}\ \mu_{A_2^j}(u_2)\ \mathbf{t}\ ...\ \mathbf{t}\ \mu_{A_n^j}(u_n))$, onde $\mathbf{t}$ é uma t-norma (em geral produto ou mínimo). O terceiro passo consiste em se obter a saída inferida de cada regra (no caso do problema de classificação, consiste em se associar o nível de disparo com o label associado ao consequente (μ^j, B^j). O quarto passo consiste em se agregar as saídas inferidas por cada regra formando a saída do sistema $\hat{y}_s$ fuzzy baseado em RB. No caso do problema de classificação a classe inferida sera dada por $\hat{y}_s = \arg\max_{B^j}(\mu^1, \mu^2, ..., \mu^m)$, ou seja o label da regra que tem o maior nível de disparo. E este consiste no quinta e ultimo passo.

Inferência **Classificador Fuzzy** BD e BR Especialista

1) Matching das entradas com as regras





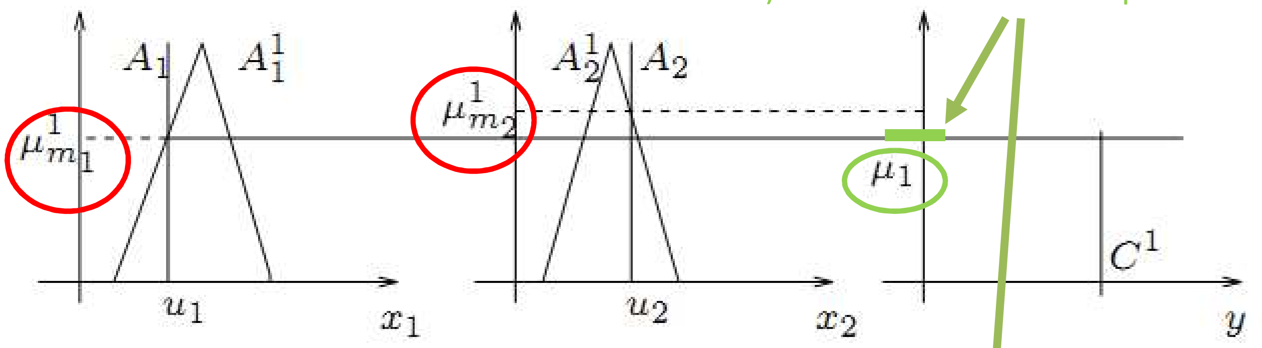
Método de Inferência

Dado um conjunto de teste S, consistindo de padrões $(\mathbf{u}_s, y_s)$, $s = 1, ..., |S|$ onde $\mathbf{u}_s = (u_1, ..., u_n)$ é o vetor de entradas e y_s é a saída. Dada uma base de regras $RB = \{R^1, R^2, ..., R^j, ...R^m\}$ obtida pelo metodo WM ou pelo especialista, onde R^j : If X_1 is A_1^j and X_2 is A_2^j and ... and X_n is A_n^j then y is B^j. Cada padrão é apresentado à base RB de forma a se obter uma saída inferida $\hat{y}_s$. O primeiro passo consiste em se obter o matching da entrada $\mathbf{u}_s = (u_1, ..., u_n)$ com a regra R^j. O matching da v-ésima variável do padrão de entrada s é dado por $\mu_{s_v}^j = \mu_{A_v^j}(u_v)$ onde $\mu_{A_v^j}(u_v)$ é o grau de pertinência para u_v na função de pertinência associada com o termo linguístico l_v definido para X_v na regra R^j. O segundo passo define o nível de disparo da regra a partir da apresentação do padrão de entrada $(\mu^j(\mathbf{u}_s) = \mu_{A_1^j}(u_1)$ **t** $\mu_{A_2^j}(u_2)$ **t** ... **t** $\mu_{A_n^j}(u_n))$, onde **t** é uma t-norma (em geral produto ou mínimo). O terceiro passo consiste em se obter a saída inferida de cada regra (no caso do problema de classificação, consiste em se associar o nível de disparo com o label associado ao consequente (μ^j, B^j). O quarto passo consiste em se agregar as saídas inferidas por cada regra formando a saída do sistema $\hat{y}_s$ fuzzy baseado em RB. No caso do problema de classificação a classe inferida sera dada por $\hat{y}_s = \arg\max_{B^j}(\mu^1, \mu^2, ..., \mu^m)$, ou seja o label da regra que tem o maior nível de disparo. E este consiste no quinta e ultimo passo.

Inferência **Classificador Fuzzy** BD e BR Especialista

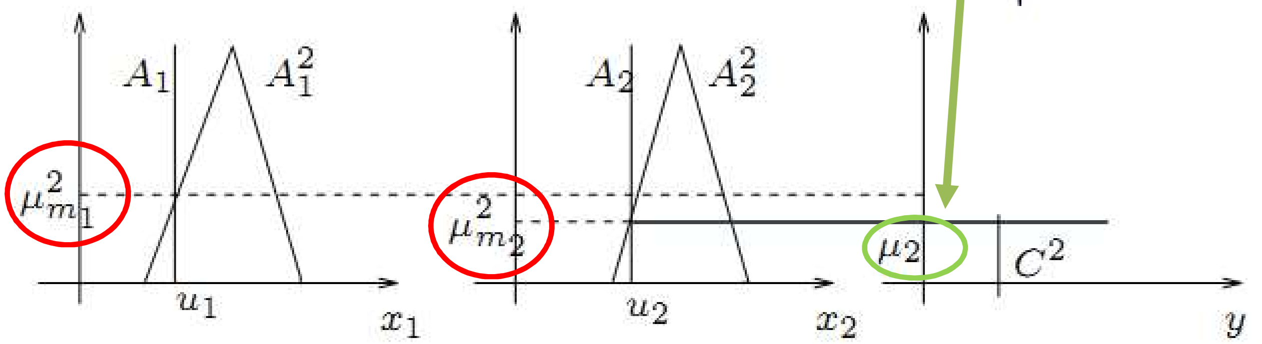
1) Matching das entradas com as regras

2) Cálculo do nível de disparo da regra



$$\mu_1 = \mu^1_{m_1} \wedge \mu^1_{m_2}$$

$$\mu_2 = \mu^2_{m_1} \wedge \mu^2_{m_2}$$



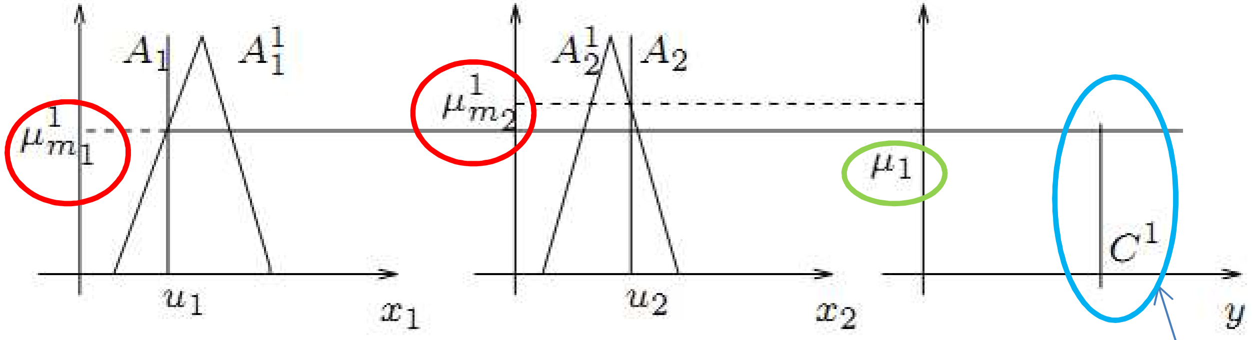
Método de Inferência

Dado um conjunto de teste S, consistindo de padrões $(\mathbf{u}_s, y_s)$, $s = 1, ..., |S|$ onde $\mathbf{u}_s = (u_1, ..., u_n)$ é o vetor de entradas e y_s é a saída. Dada uma base de regras $RB = \{R^1, R^2, ..., R^j, ...R^m\}$ obtida pelo metodo WM ou pelo especialista, onde R^j : If X_1 is A_1^j and X_2 is A_2^j and ... and X_n is A_n^j then y is B^j. Cada padrão é apresentado à base RB de forma a se obter uma saída inferida $\hat{y}_s$. O primeiro passo consiste em se obter o matching da entrada $\mathbf{u}_s = (u_1, ..., u_n)$ com a regra R^j. O matching da v-ésima variável do padrão de entrada s é dado por $\mu_{s_v}^j = \mu_{A_v^j}(u_v)$ onde $\mu_{A_v^j}(u_v)$ é o grau de pertinência para u_v na função de pertinência associada com o termo linguístico l_v definido para X_v na regra R^j. O segundo passo define o nível de disparo da regra a partir da apresentação do padrão de entrada $(\mu^j(\mathbf{u}_s) = \mu_{A_1^j}(u_1)$ **t** $\mu_{A_2^j}(u_2)$ **t** ... **t** $\mu_{A_n^j}(u_n))$, onde **t** é uma t-norma (em geral produto ou mínimo). O terceiro passo consiste em se obter a saída inferida de cada regra (no caso do problema de classificação, consiste em se associar o nível de disparo com o label associado ao consequente (μ^j, B^j). O quarto passo consiste em se agregar as saídas inferidas por cada regra formando a saída do sistema $\hat{y}_s$ fuzzy baseado em RB. No caso do problema de classificação a classe inferida sera dada por $\hat{y}_s = \arg\max_{B^j}(\mu^1, \mu^2, ..., \mu^m)$, ou seja o label da regra que tem o maior nível de disparo. E este consiste no quinta e ultimo passo.

Inferência **Classificador Fuzzy** BD e BR Especialista

1) Matching das entradas com as regras

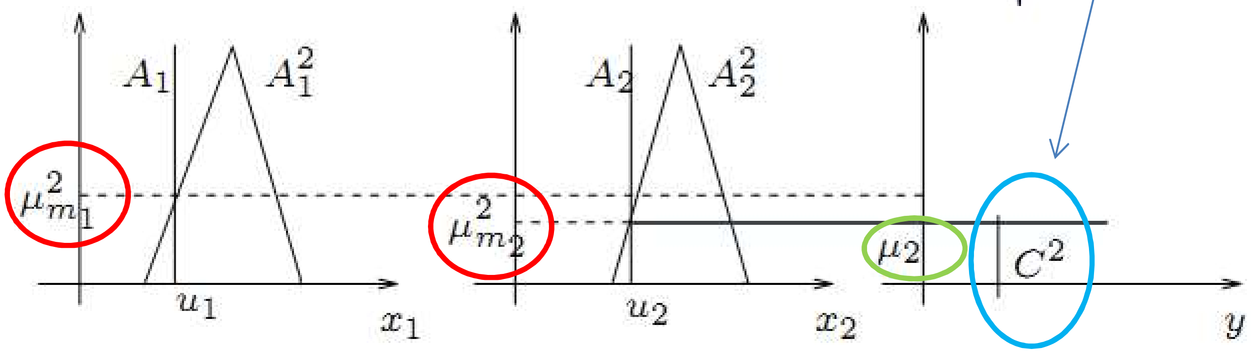
2) Cálculo do nível de disparo da regra



3) Saída Inferida pela regra

$$\mu_1 = \mu_{m_1}^1 \wedge \mu_{m_2}^1$$

$$\mu_2 = \mu_{m_1}^2 \wedge \mu_{m_2}^2$$



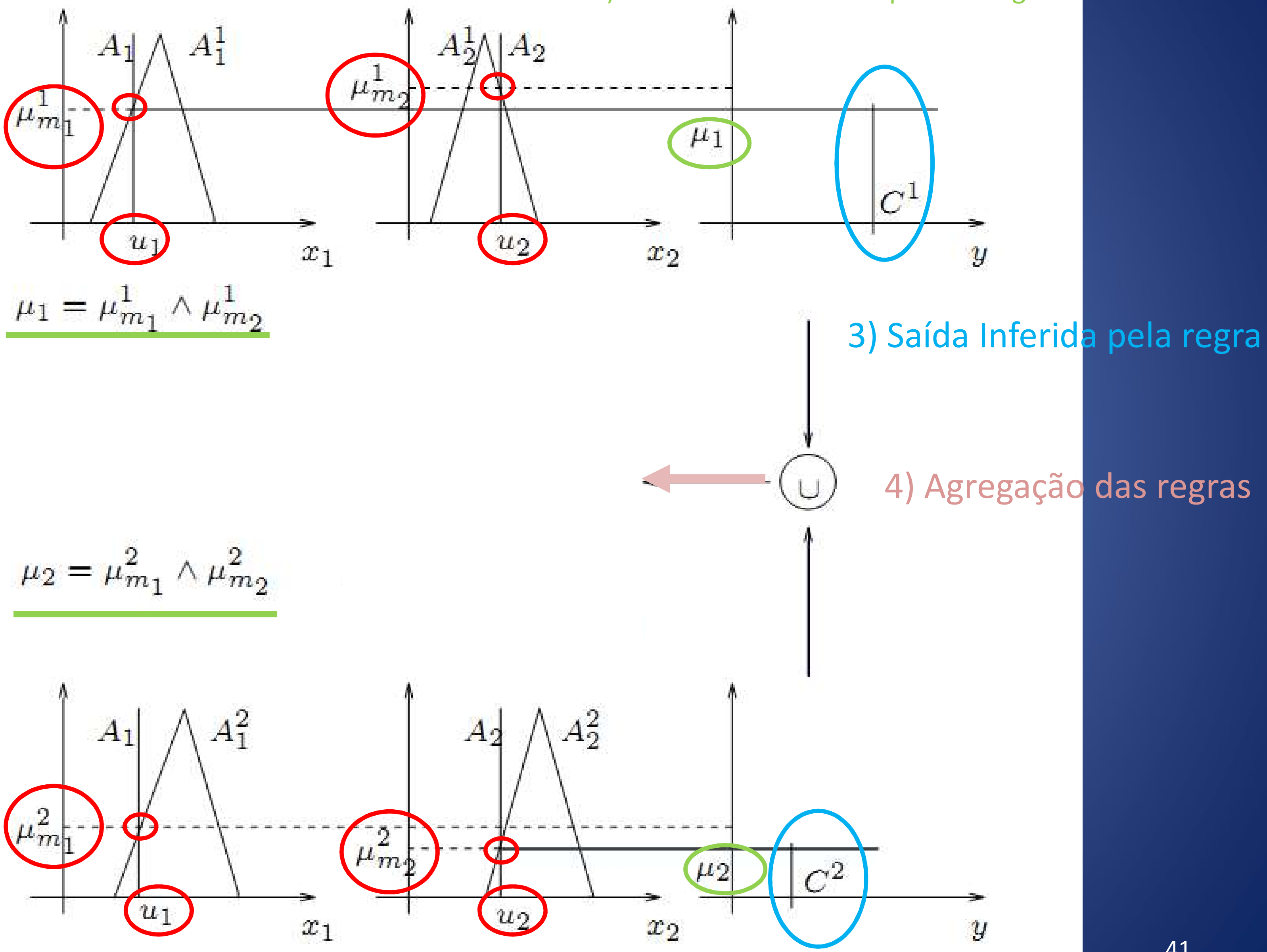
Método de Inferência

Dado um conjunto de teste S, consistindo de padrões $(\mathbf{u}_s, y_s)$, $s = 1, ..., |S|$ onde $\mathbf{u}_s = (u_1, ..., u_n)$ é o vetor de entradas e y_s é a saída. Dada uma base de regras $RB = \{R^1, R^2, ..., R^j, ...R^m\}$ obtida pelo metodo WM ou pelo especialista, onde R^j : If X_1 is A_1^j and X_2 is A_2^j and ... and X_n is A_n^j then y is B^j. Cada padrão é apresentado à base RB de forma a se obter uma saída inferida $\hat{y}_s$. O primeiro passo consiste em se obter o matching da entrada $\mathbf{u}_s = (u_1, ..., u_n)$ com a regra R^j. O matching da v-ésima variável do padrão de entrada s é dado por $\mu_{s_v}^j = \mu_{A_v^j}(u_v)$ onde $\mu_{A_v^j}(u_v)$ é o grau de pertinência para u_v na função de pertinência associada com o termo linguístico l_v definido para X_v na regra R^j. O segundo passo define o nível de disparo da regra a partir da apresentação do padrão de entrada $(\mu^j(\mathbf{u}_s) = \mu_{A_1^j}(u_1)\ \mathbf{t}\ \mu_{A_2^j}(u_2)\ \mathbf{t}\ ...\ \mathbf{t}\ \mu_{A_n^j}(u_n))$, onde $\mathbf{t}$ é uma t-norma (em geral produto ou mínimo). O terceiro passo consiste em se obter a saída inferida de cada regra (no caso do problema de classificação, consiste em se associar o nível de disparo com o label associado ao consequente (μ^j, B^j). O quarto passo consiste em se agregar as saídas inferidas por cada regra formando a saída do sistema $\hat{y}_s$ fuzzy baseado em RB. No caso do problema de classificação a classe inferida sera dada por $\hat{y}_s = \arg\max_{B^j}(\mu^1, \mu^2, ..., \mu^m)$, ou seja o label da regra que tem o maior nível de disparo. E este consiste no quinta e ultimo passo.

Inferência **Classificador Fuzzy** BD e BR Especialista

1) Matching das entradas com as regras

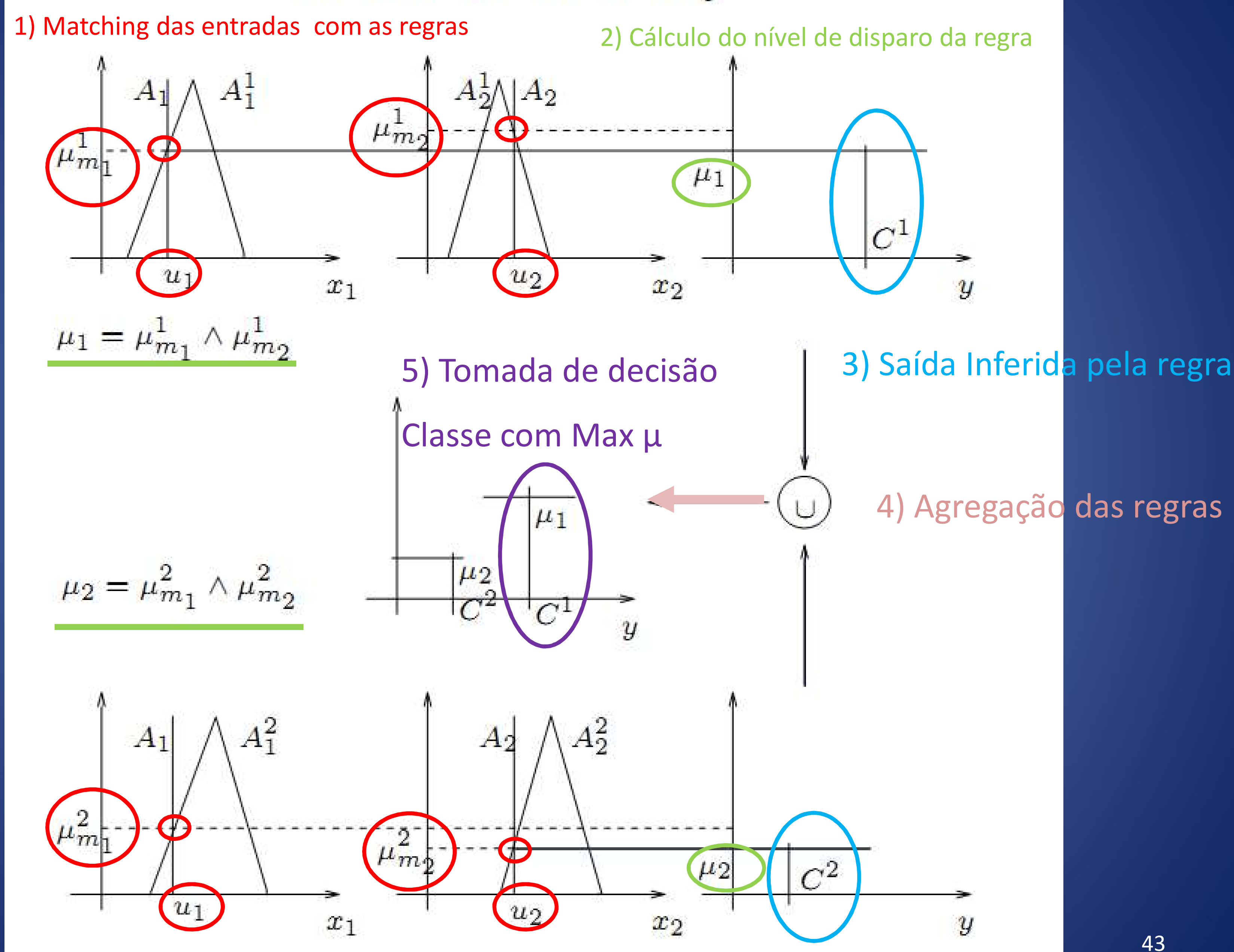
2) Cálculo do nível de disparo da regra



Método de Inferência

Dado um conjunto de teste S, consistindo de padrões $(\mathbf{u}_s, y_s)$, $s = 1, ..., |S|$ onde $\mathbf{u}_s = (u_1, ..., u_n)$ é o vetor de entradas e y_s é a saída. Dada uma base de regras $RB = \{R^1, R^2, ..., R^j, ...R^m\}$ obtida pelo metodo WM ou pelo especialista, onde R^j : If X_1 is A_1^j and X_2 is A_2^j and ... and X_n is A_n^j then y is B^j. Cada padrão é apresentado à base RB de forma a se obter uma saída inferida $\hat{y}_s$. O primeiro passo consiste em se obter o matching da entrada $\mathbf{u}_s = (u_1, ..., u_n)$ com a regra R^j. O matching da v-ésima variável do padrão de entrada s é dado por $\mu_{s_v}^j = \mu_{A_v^j}(u_v)$ onde $\mu_{A_v^j}(u_v)$ é o grau de pertinência para u_v na função de pertinência associada com o termo linguístico l_v definido para X_v na regra R^j. O segundo passo define o nível de disparo da regra a partir da apresentação do padrão de entrada $(\mu^j(\mathbf{u}_s) = \mu_{A_1^j}(u_1)\ \mathbf{t}\ \mu_{A_2^j}(u_2)\ \mathbf{t}\ ...\ \mathbf{t}\ \mu_{A_n^j}(u_n))$, onde $\mathbf{t}$ é uma t-norma (em geral produto ou mínimo). O terceiro passo consiste em se obter a saída inferida de cada regra (no caso do problema de classificação, consiste em se associar o nível de disparo com o label associado ao consequente (μ^j, B^j). O quarto passo consiste em se agregar as saídas inferidas por cada regra formando a saída do sistema $\hat{y}_s$ fuzzy baseado em RB. No caso do problema de classificação a classe inferida sera dada por $\hat{y}_s = \arg\max_{B^j}(\mu^1, \mu^2, ..., \mu^m)$, ou seja o label da regra que tem o maior nível de disparo. E este consiste no quinta e ultimo passo.

Inferência **Classificador Fuzzy** BD e BR Especialista



Classificador Fuzzy

Base de dados (BD): Partição do Universo (grid ou c-means)

Especialista

Base de regras (BR): Conjunto de regras

Automática

Mecanismo de raciocínio: Inferência

Operadores:

agregação de antecedentes

agregação das regras,

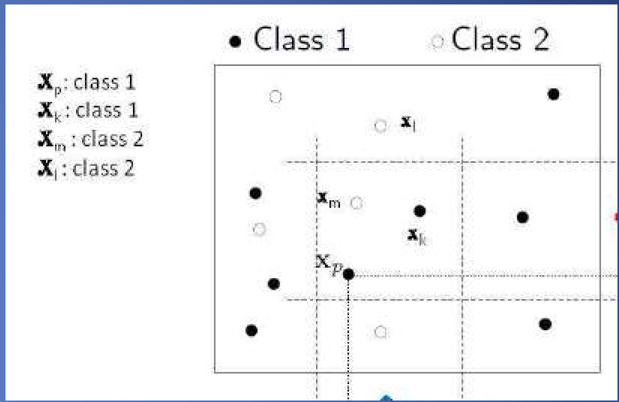
tomada de decisão (max ou soma).

Definição Automática da BR

- Dados de Treinamento

x1	x2	x3	y
15,04	-0,12	-0,48	13,47
20,97	0,29	1,10	-22,24
-25,54	0,30	-1,72	53,74
4,73	0,32	-1,17	-19,74
17,82	0,68	1,06	-83,21
-15,69	-0,36	1,01	47,35
-6,68	-0,54	-0,98	49,82
-3,98	0,58	-1,27	6,60
9,93	0,21	-2,53	33,42
8,41	0,74	0,43	-43,88

Problema de aproximação



Problema de classificação

x1	x2	Class
15,04	-0,12	1
20,97	0,29	1
-25,54	0,30	2
4,73	0,32	2
17,82	0,68	1
-15,69	-0,36	1
-6,68	-0,54	2
-3,98	0,58	2
9,93	0,21	2
8,41	0,74	1

•Método de Wang Mendel

(BR a partir dos dados)

Algoritmo de Wang Mendel

Método para geração de regras fuzzy a partir de dados de entrada-saída.

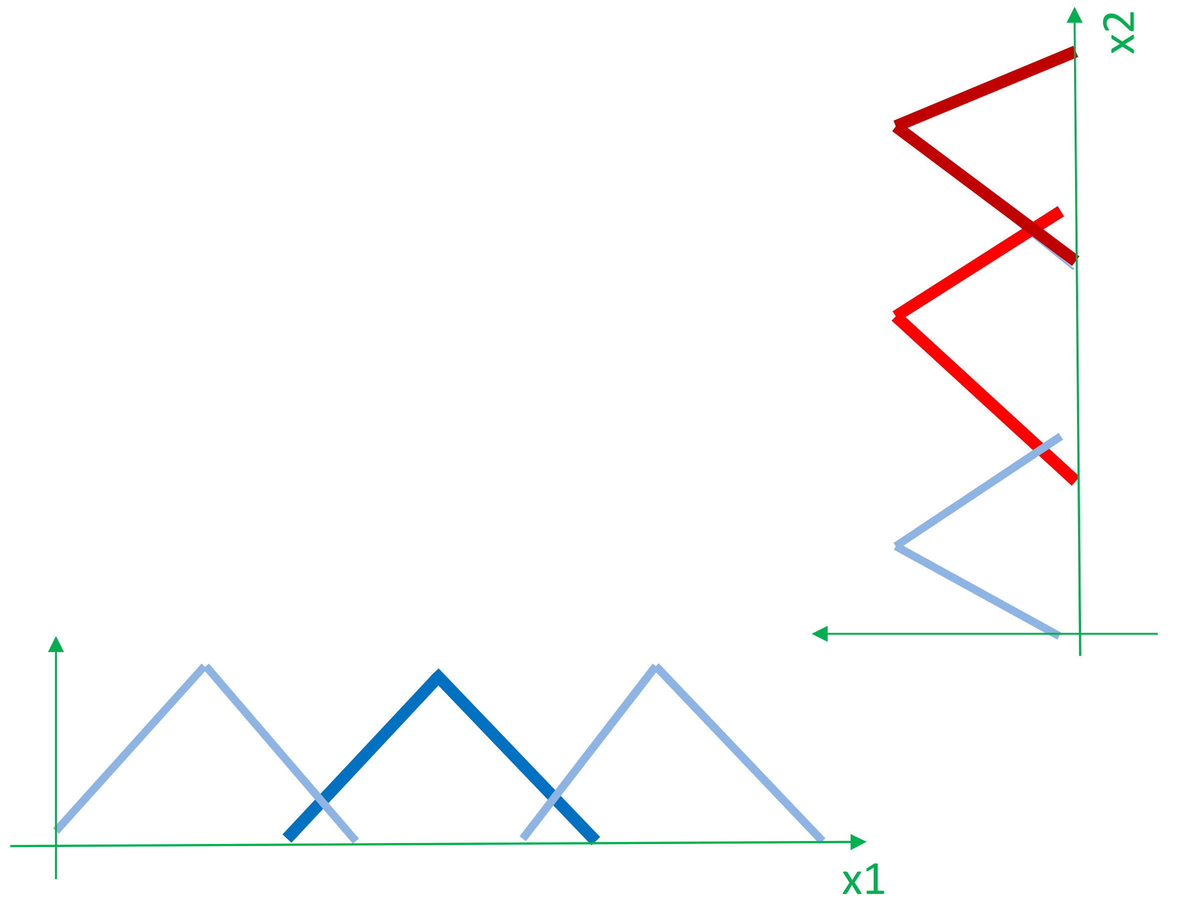
Passo1: Definir o número de termos linguísticos e particionar os universos de todas as variáveis de entrada.

Passo2: Criar uma regra fuzzy para cada elemento do conjunto de treino – para cada variável (atributo) selecionar a MF com maior grau de pertinência.

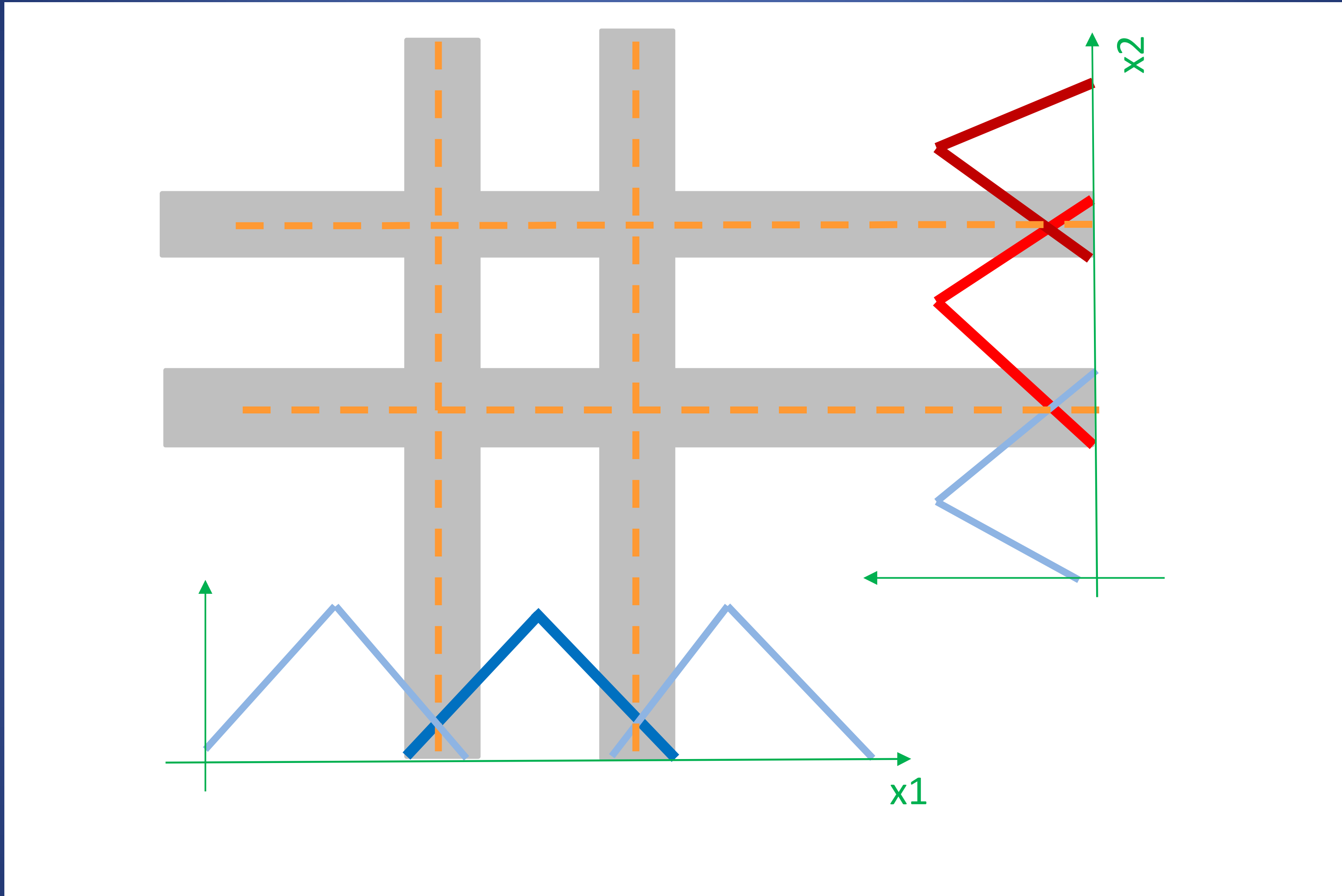
Passo 3: calcular o nível de disparo de todas as regras
(função da norma-t dos níveis de ativação de cada antecedente).

Passo 4: Eliminar as redundâncias e inconsistências (regras com mesmo antecedente e consequente diferente) deixando apenas as regras com maior nível de disparo.

WM: Passo 1 – Partição Fuzzy

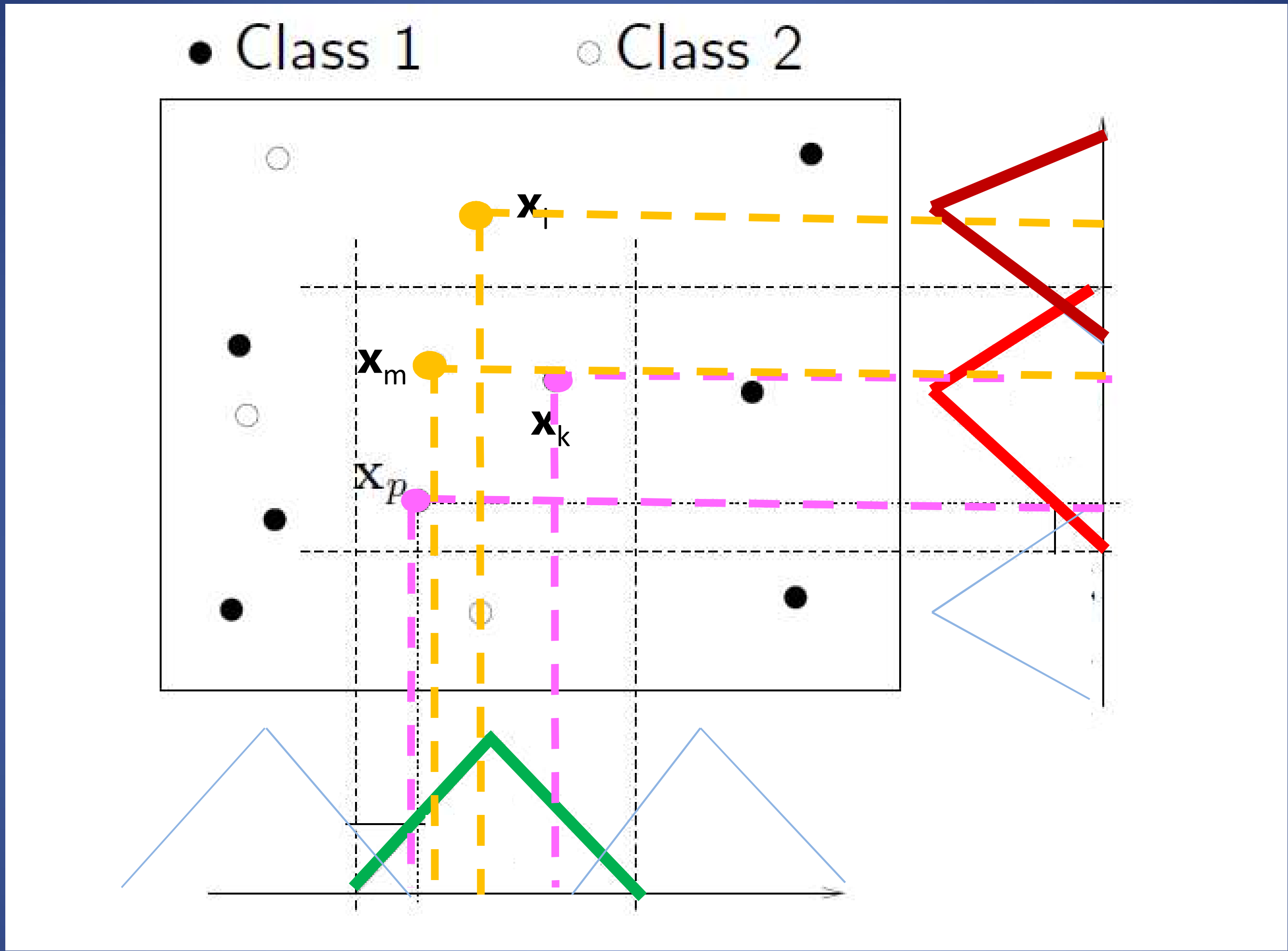


WM: Passo 1 – Partição Fuzzy



WM: Passo 2 – 1 regra p/ cada entrada de treino

X_p: class 1
X_k : class 1
X_m : class 2
X_l : class 2



Regra p: Se x1 é A^p_1 e x2 é A^p_2 então classe é **class1**

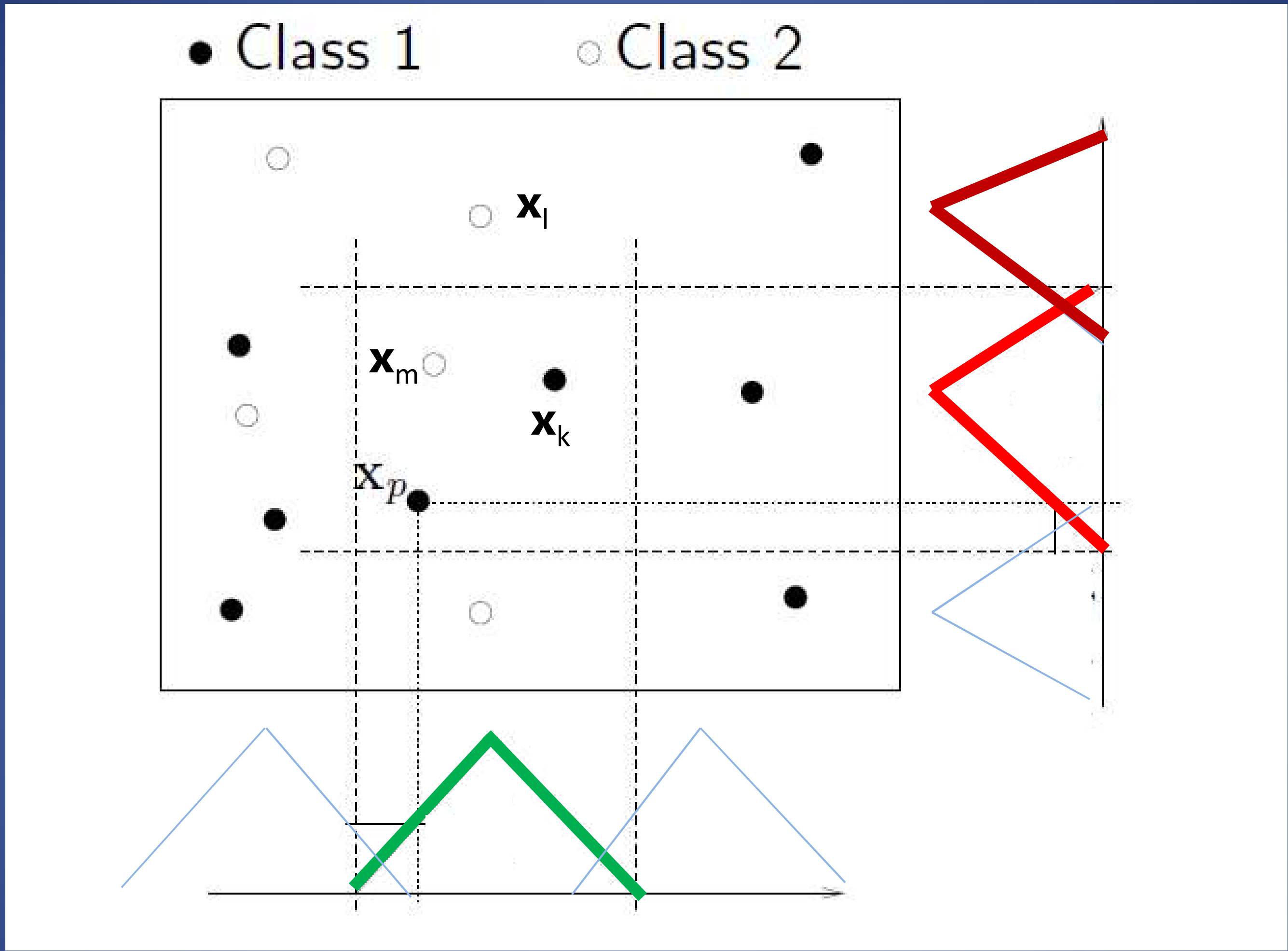
Regra k: Se x1 é A^k_1 e x2 é A^k_2 então classe é **class1**

Regra m: Se x1 é A^m_1 e x2 é A^m_2 então classe é **class2**

x_l não gera Regra na Região A^j_1 x A^j_2

WM: Passo 2 – 1 regra p/ cada entrada de treino

$\mathbf{X}_p$: class 1
$\mathbf{X}_k$: class 1
$\mathbf{X}_m$: class 2
$\mathbf{X}_l$: class 2



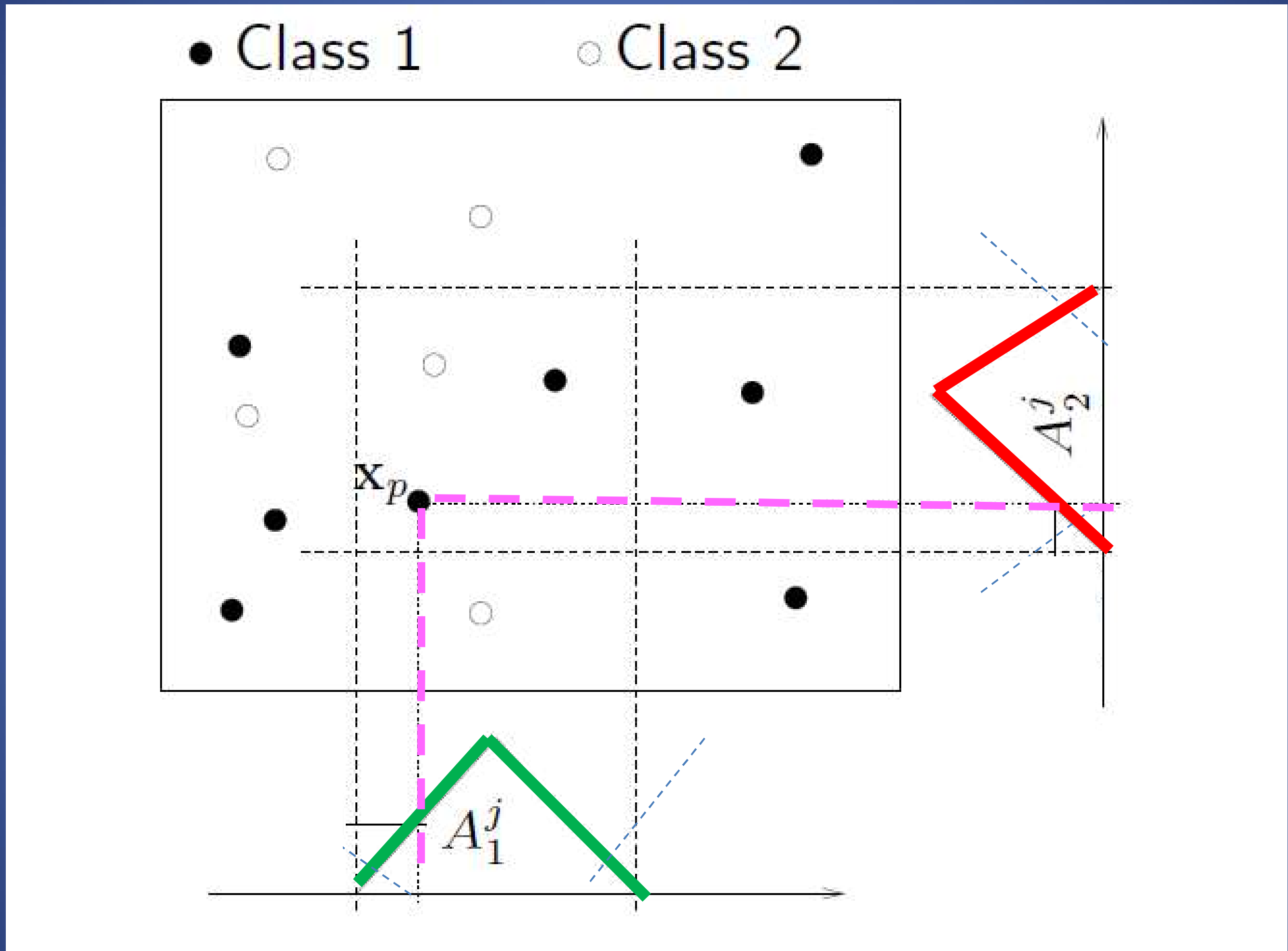
Regra p: Se x1 é A^j_1 e x2 é A^j_2 então classe é **class1**

Regra k: Se x1 é A^j_1 e x2 é A^j_2 então classe é **class1**

Regra m: Se x1 é A^j_1 e x2 é A^j_2 então classe é **class2**

x_l não gera Regra na Região A^j_1 x A^j_2

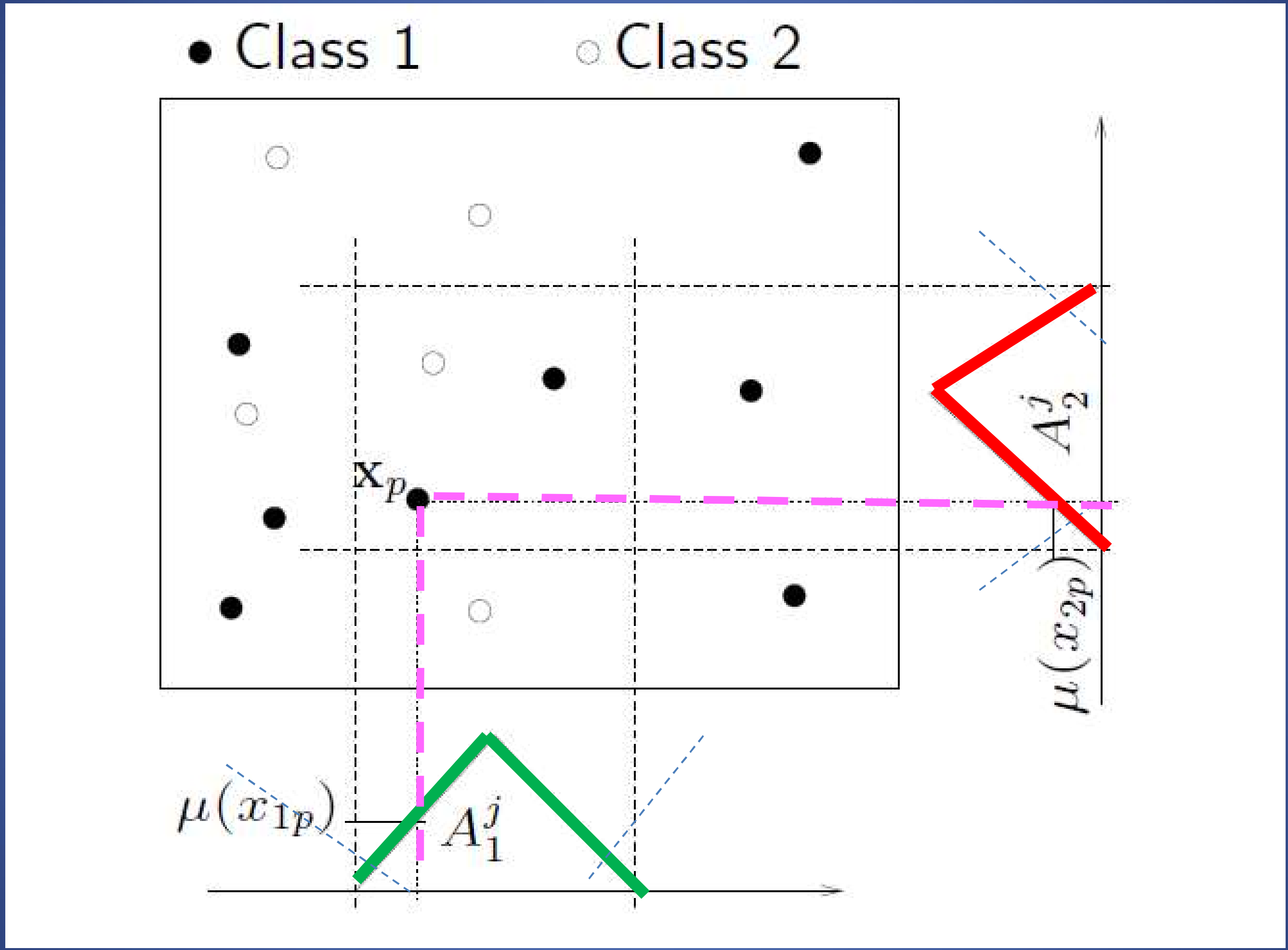
WM: Passo 3 – Nível de disparo da regra



Regra p: Se x1 é A^j_1 e x2 é A^j_2 então classe é **class1**

Nível de disparo da regra p: $\mu(X_{1p})$ **t** $\mu(X_{2p})$

WM: Passo 3 – Nível de disparo da regra



Regra p: Se x1 é A^j_1 e x2 é A^j_2 então classe é **class1**

Nível de disparo da regra p: $\mu(\mathbf{X}_{1p})$ **t** $\mu(\mathbf{X}_{2p})$

WM: Passo 4 – elimina redundância e inconsistência nas regras (nível de disparo)

Regra p: Se x1 é A^j_1 e x2 é A^j_2 então classe é **class1**

Nível de disparo da regra: $FS_{Rp} = \mu(\mathbf{X}_{1p})\ t\ \mu(\mathbf{X}_{2p})$

Regra k: Se x1 é A^j_1 e x2 é A^j_2 então classe é **class1**

Nível de disparo da regra k: $FS_{Rk} = \mu(\mathbf{X}_{1k})\ t\ \mu(\mathbf{X}_{2k})$

Regra m: Se x1 é A^j_1 e x2 é A^j_2 então classe é **class2**

Nível de disparo da regra m: $FS_{Rm} = \mu(\mathbf{X}_{1m})\ t\ \mu(\mathbf{X}_{2m})$

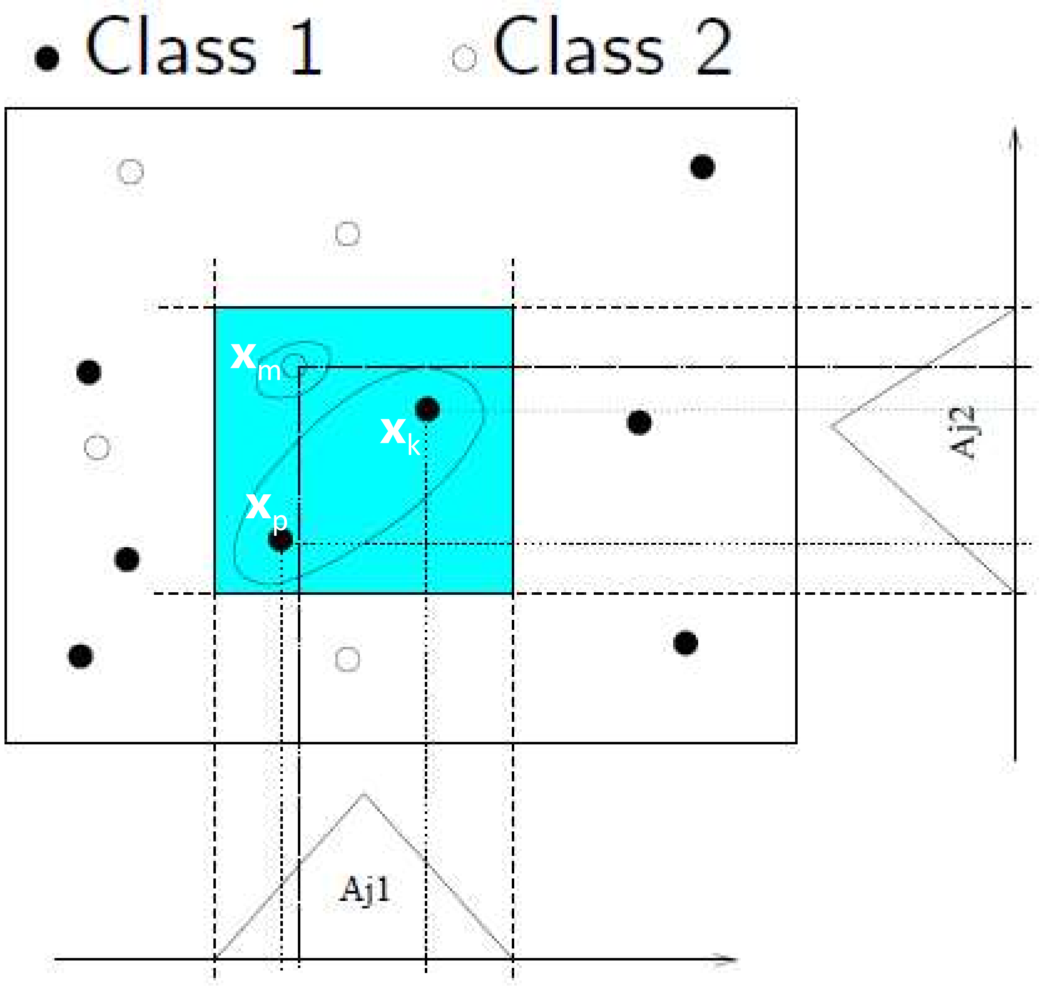
Regras p e k são reduntantes (iguais) e ambas são inconsistentes com a regra m

Se $FS_{Rp} < FS_{Rk} < FS_{Rm}$ **Apenas a regra m permanece**

Regra m: Se x1 é A^j_1 e x2 é A^j_2 então classe é **class2**

Nível de disparo da regra m: $\mu(\mathbf{X}_{1m})\ t\ \mu(\mathbf{X}_{2m})$

WM: Passo 4 – elimina redundância e inconsistência nas regras (maioria)



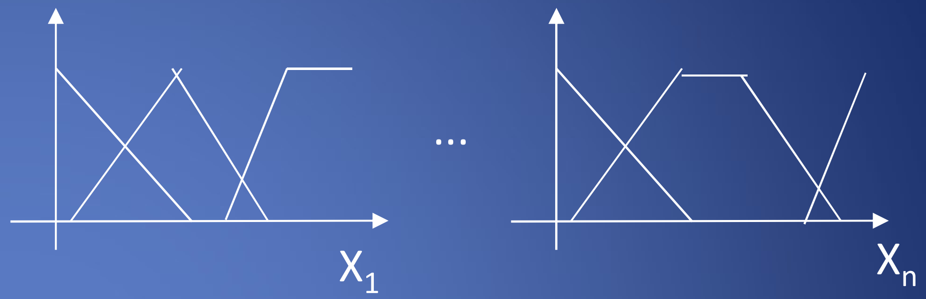
Sumarizando:
Método de Treinamento Wang-Mendel

1 Método de Wang-Mendel

Dado um conjunto de treinamento T, consistindo de pares $(\mathbf{x}_t, y_t)$, $t = 1, ..., |T|$ onde $\mathbf{x}_t = (x_1, ..., x_n)$ é o vetor de entradas y_t é a saída, uma regra R^t é criada para cada par $(\mathbf{x}, y)$, considerando a partição fuzzy para cada variável de entrada X_v, $v = 1, ..., n$, onde L_v é o total de termos linguísticos na partição da variável X_v. O termo linguístico l_v^t escolhido para cada variável X_v na regra R^t é l_v^t = arg max $(\mu_{l_v}(x_v), l_v = 1, ..., L_v)$, i.e, l_v^t é associado com a função de pertinênca que tem o maor grau de pertinência para o ponto com coordenada x_v. Um base de regras inicial (chamada base completa) é gerada ($RB_{complete} = \{R^1, R^2, ..., R^t, ...R^{|T|}\}$) agrupando todas as regras geradas pelos dados de treinamento. $RB_{complete}$ pode conter regras redundantes (iguais) e inconsistentes (mesmo antecedente e consequentes diferentes). Um base de regras final (base reduzida) RB_{reduc} é criada a partir de um processo de refinamento de $RB_{complete}$. Neste caso, apenas uma única regra é mantida para cada antecedente possível (premissa da regra) e seu consequente é escolhido tendo por base o nível de disparo (FS do inglês Firing Strength) ($FS^t = \mu_{l_1^t}(x_1)$ $\mathbf{t}$ $\mu_{l_2^t}(x_2)$ $\mathbf{t}$ ... $\mathbf{t}$ $\mu_{l_n^t}(x_n)$), onde $\mathbf{t}$ é uma t-norma (em geral produto ou mínimo) e $\mu_{l_v^t}(x_v)$ é o grau de pertinência para x_v na função de pertinência associada com o termo linguístico l_v escolhido para X_v na regra R^t. O consequente é tal que seu nível de disparo é o maior dentre todos os consequentes com mesmo antecedente (mesma premissa da regra).

Método de Wang Mendel

Dada um partição Fuzzy:

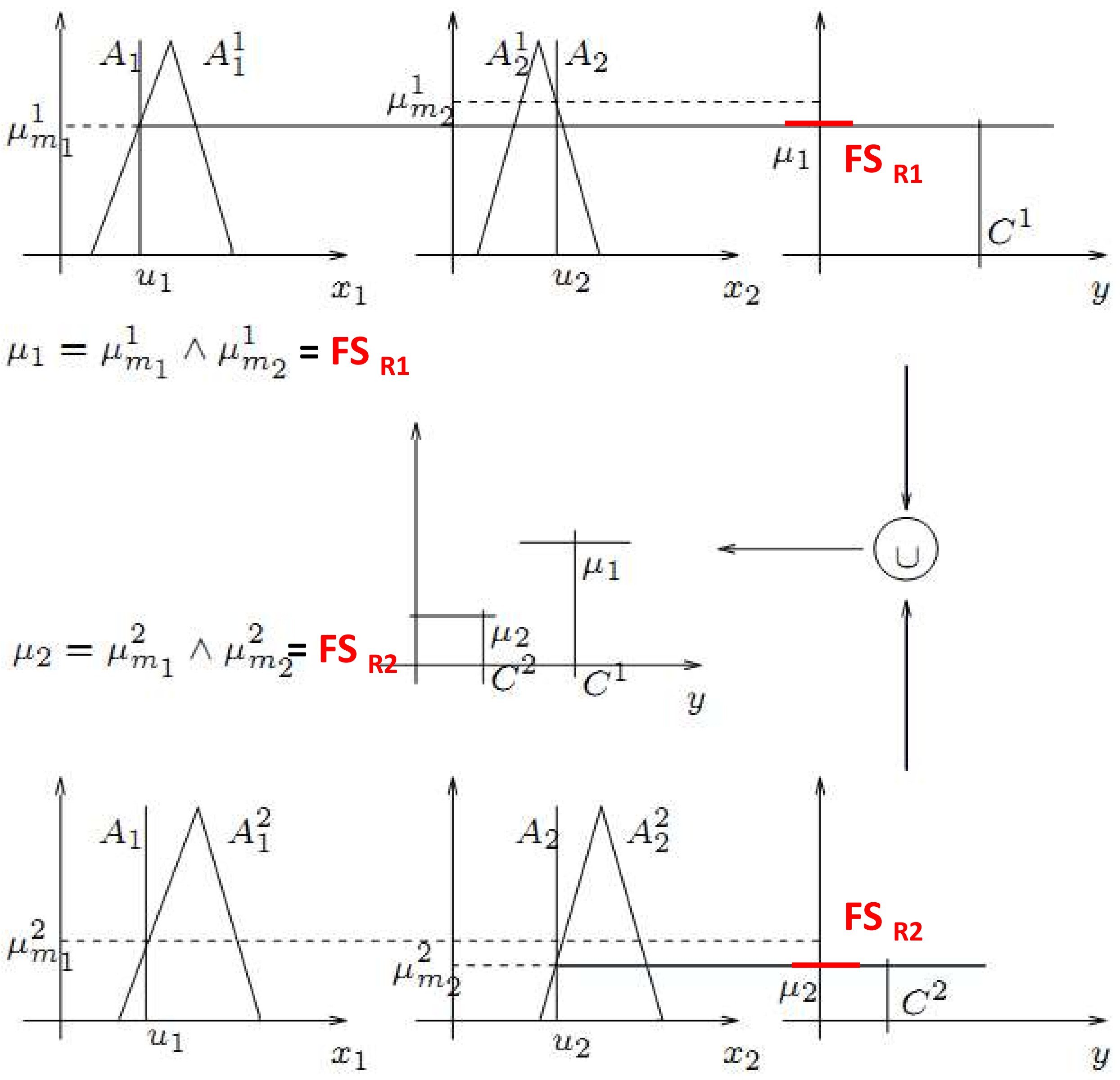


Para cada $(\mathbf{x}_t, y_t)$, com $\mathbf{x}_t = (x_{t1}, \ldots, x_{tn}) \in T$,
criar uma regra fuzzy com antec e conseq dados por:
(antec = max_i *MFi* (X_k), $k=1,\ldots n$, *conseq*$=y_t$);

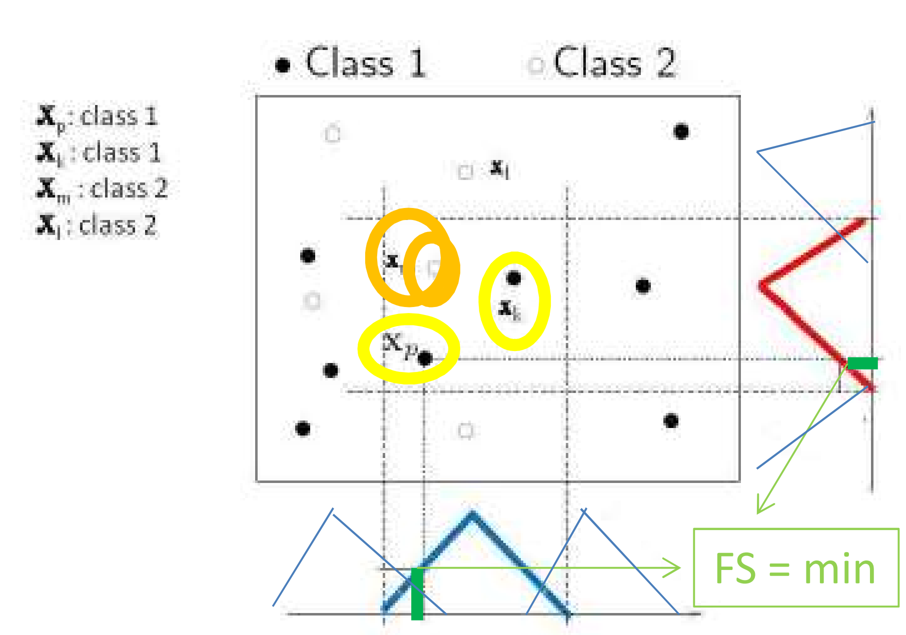
Calcular nível de disparo (FS_{Rt}) para cada regra fuzzy criada;

Eliminar redundâncias e inconsistências
(Regra Final = max FS_{Rt} para toda regra de mesmo antec.

Classificador Fuzzy



Wang Mendel Method



Problema de Classificação

1) Para cada $\mathbf{x}_t \in T$, criar uma regra fuzzy (antec = max *MF*);

2) Calcular nível de disparo (*FS)* para cada regra fuzzy criada;

3) Eliminar redundâncias e inconsist (Regra Final = max *FS*).

R_p: If X_1 is $A^p{}_1$ and X_2 is $A^p{}_2$ then Y is Class 1

R_k: If X_1 is $A^k{}_1$ and X_2 is $A^k{}_2$ then Y is Class 1

R_m: If X_1 is $A^m{}_1$ and X_2 is $A^m{}_2$ then Y is Class 2

redundâncias

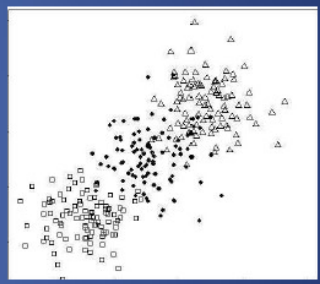
inconsistências

R_m: If X_1 is $A^m{}_1$ and X_2 is $A^m{}_2$ then Y is Class 2 regra final

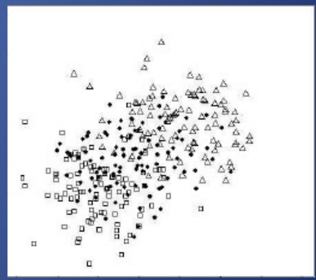
Aplicação do Método WM

- Problema de Classificação

•Dados artificiais representando 3 classes



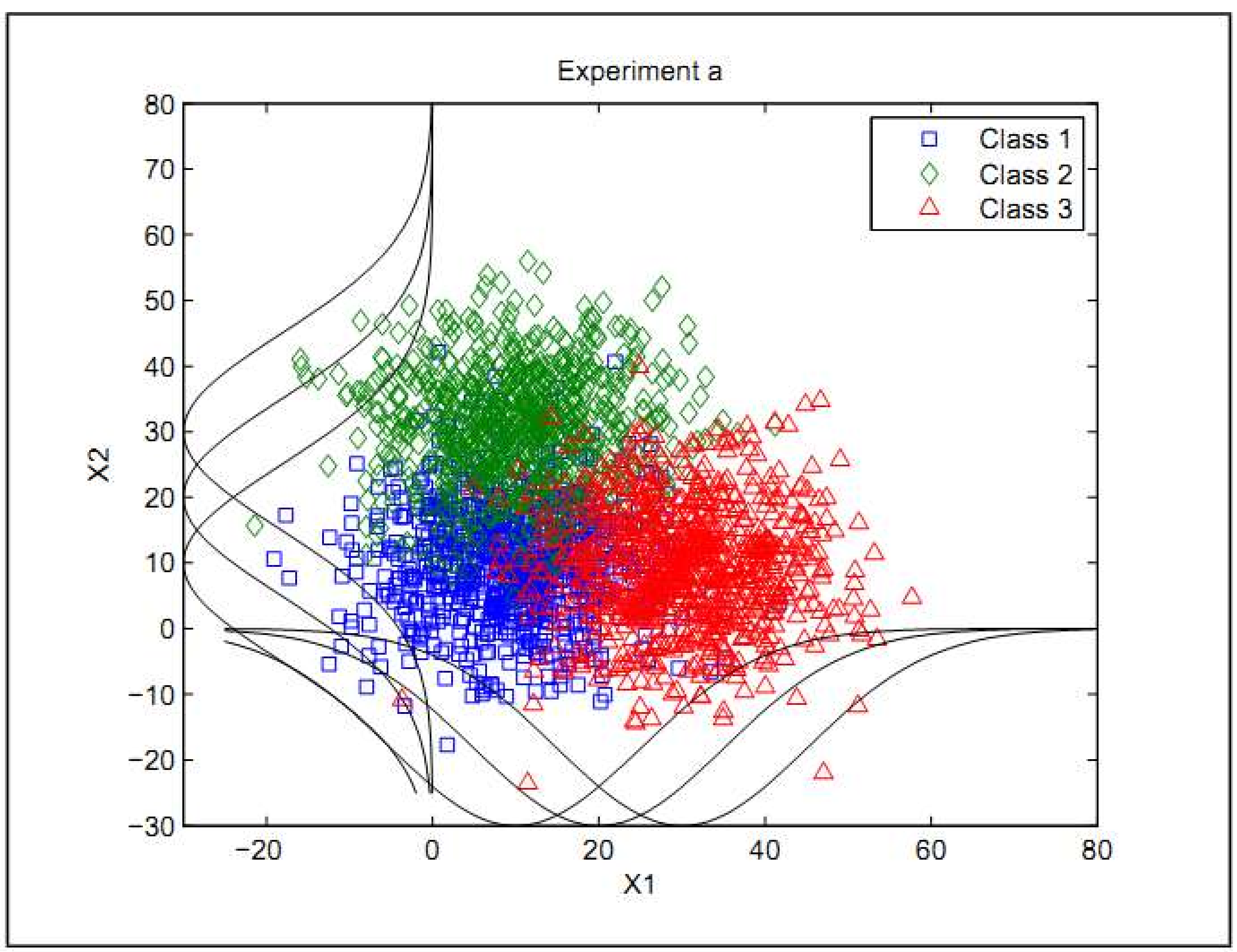
2 variáveis de entrada



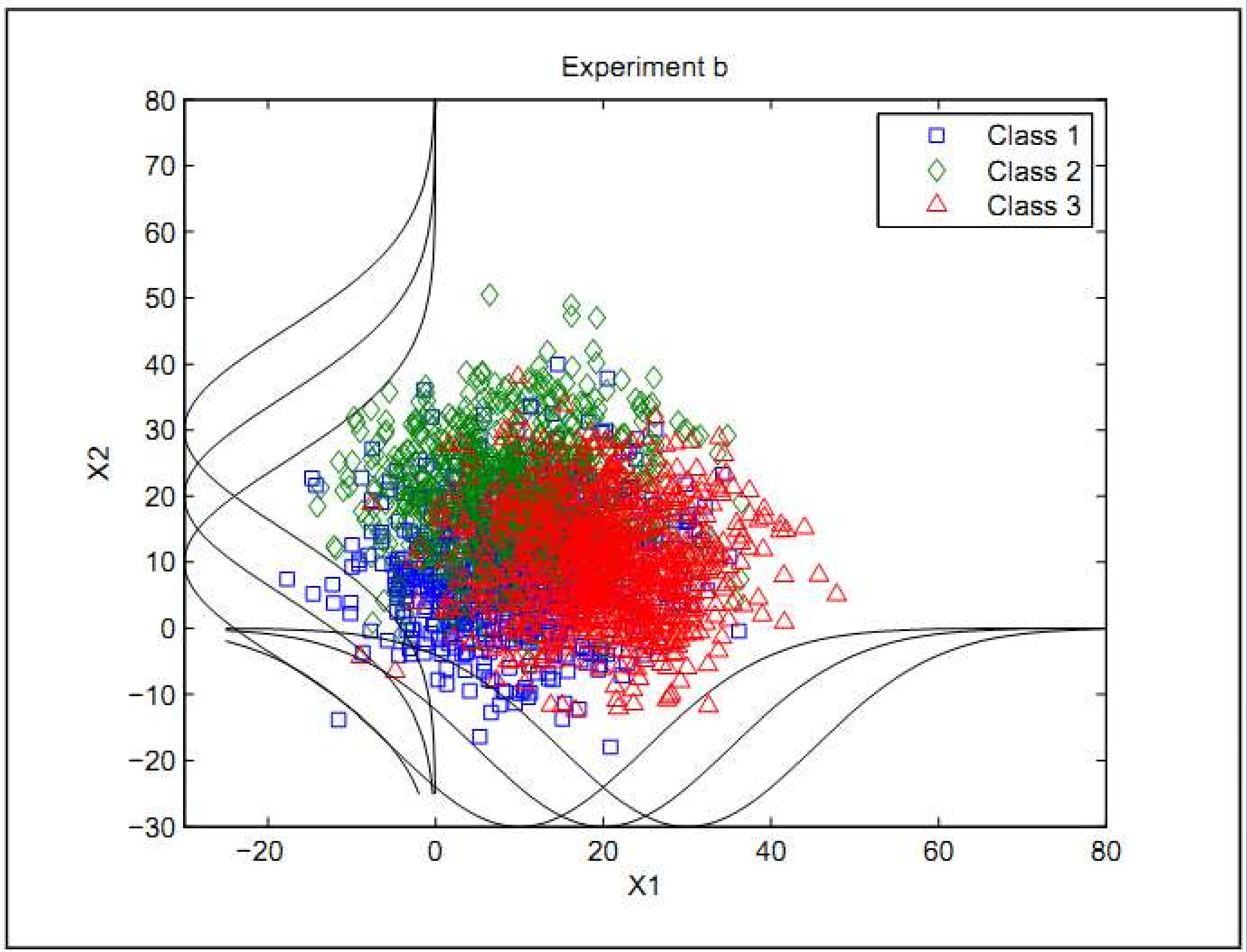
•Instâncias mais separadas

•Instâncias mais próximas

(instância mais separada)

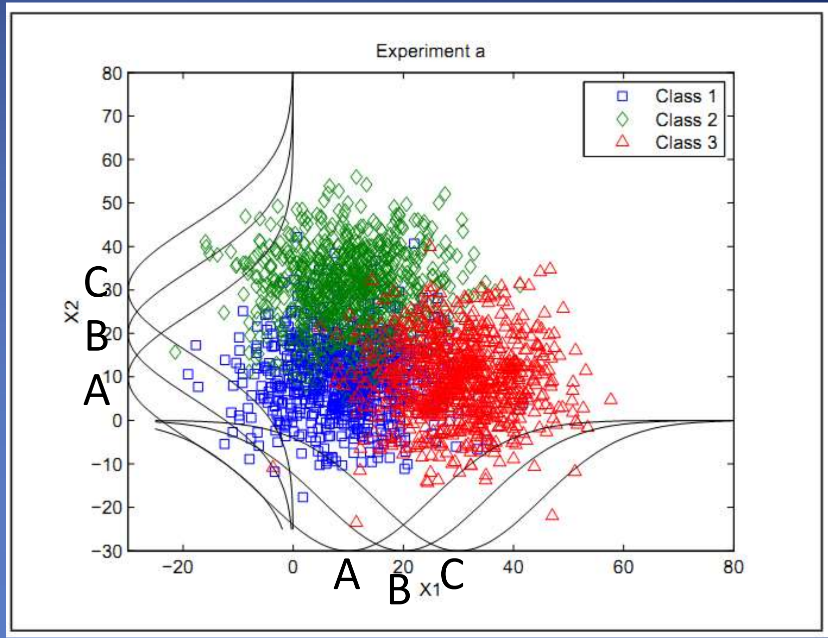


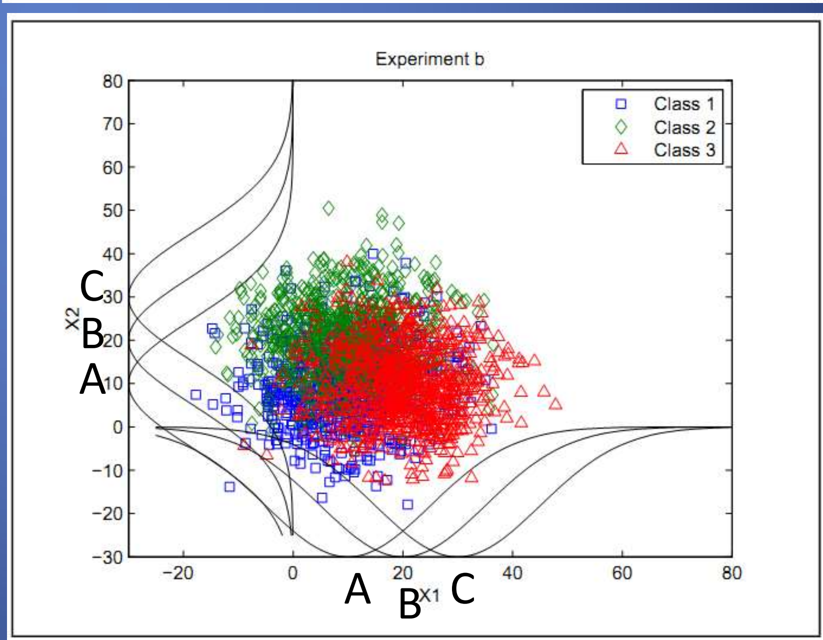
(instância mais próxima)



Resultado do Método de WM

If X1 is	and X2 is	WM then Class is: Instance a	WM then Class is: Instance b
1	1	A	A
1	2	A	B
1	3	A	C
2	1	B	A
2	2	B	B
2	3	B	C
3	1	C	A
3	2	C	B
3	3	C	C





Aplicação do Método WM

- Problemas de Classificação (4 classes)

5 variáveis de entrada

STG (The degree of study time for goal object materials)

SCG (The degree of repetition number of user for goal object materials)

STR (The degree of study time of user for related objects with goal object)

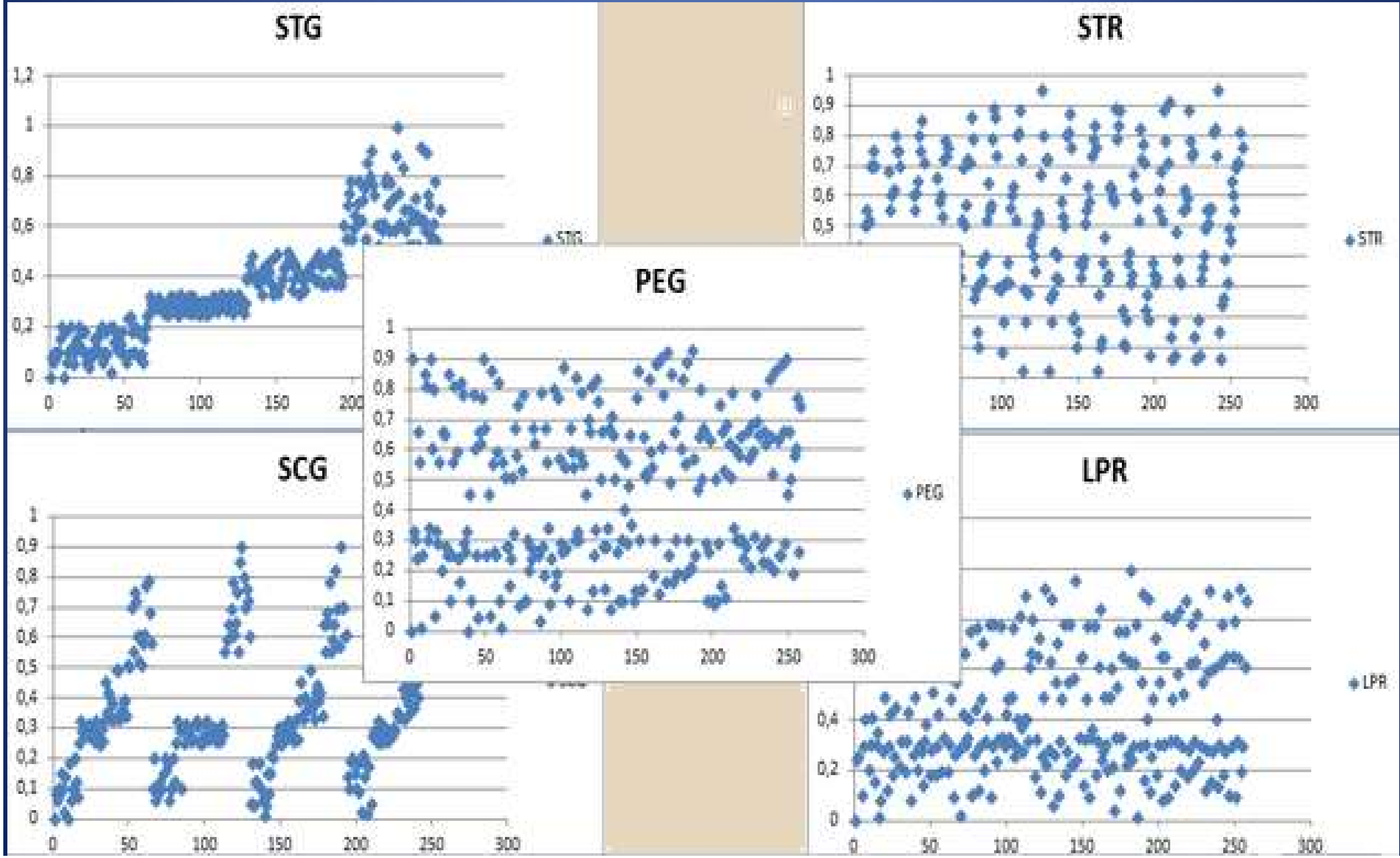
LPR (The exam performance of user for related objects with goal object)

PEG (The exam performance of user for goal objects)

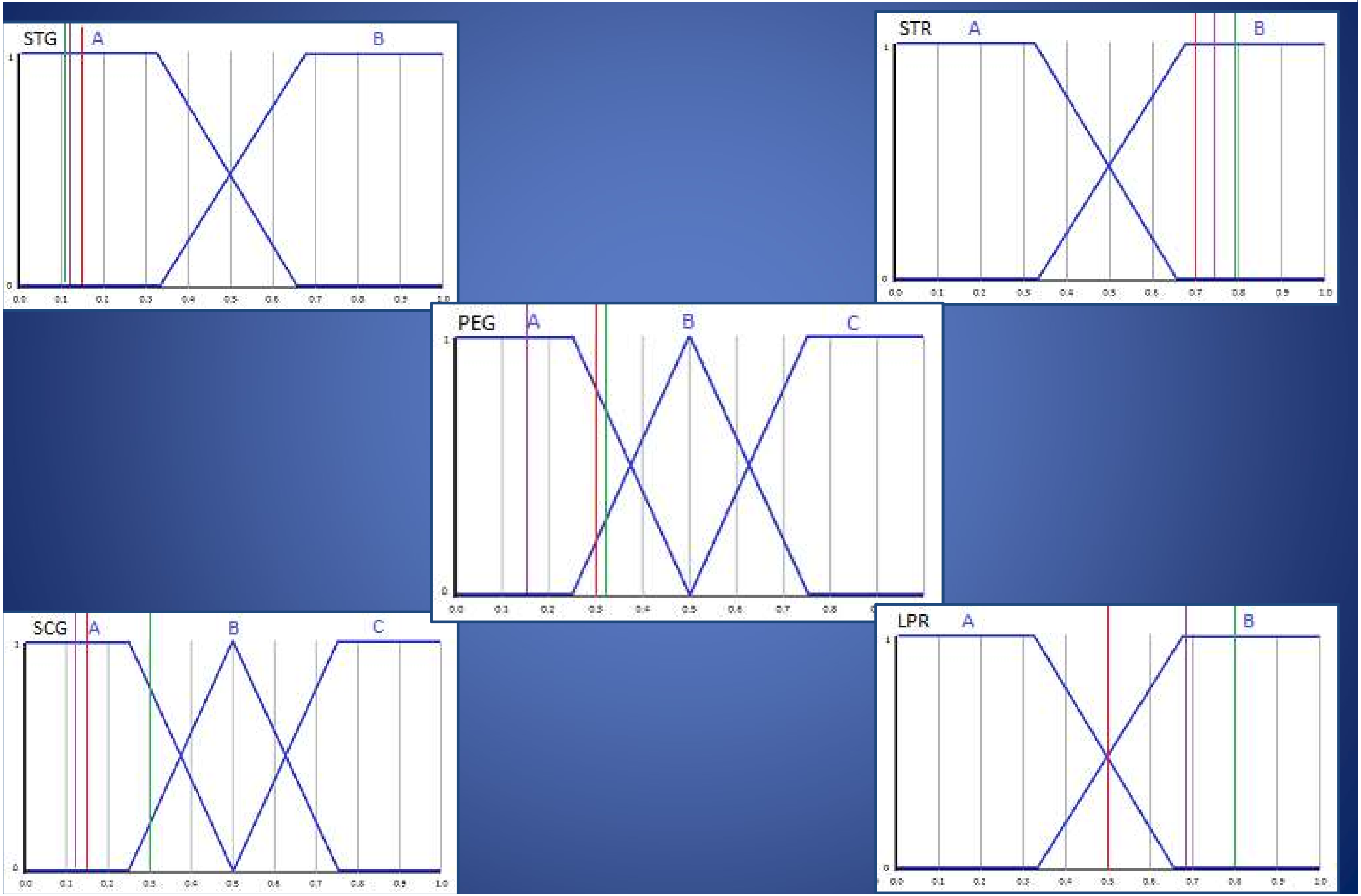
1 variável de saída

Nível de conhecimento do aluno: {(L)ow, (M)edium, (H)igh, (V)ery (H)igh}

Classificação do conhecimento do Usuário

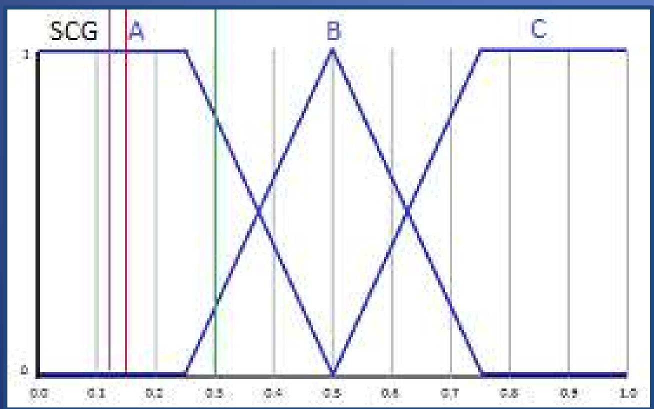


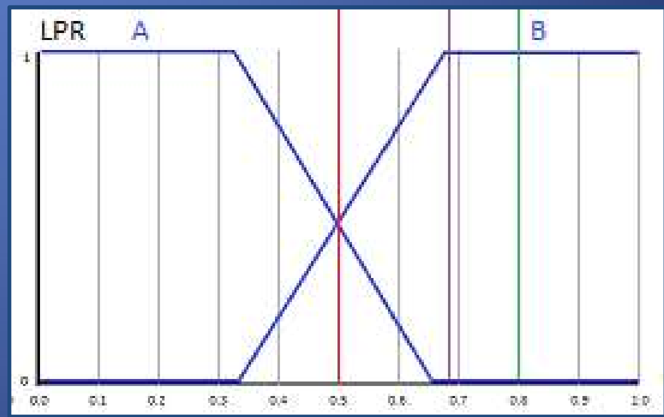
WM: Matching (violeta, vermelho e verde) com a Partição



WM: Tabela de Matching: Ponto x Partição

	STG		SCG			STR		LPR		PEG		
	A	B	A	B	C	A	B	A	B	A	B	C
x' (verde) M	1.0	0.0	0.8	0.2	0.0	0.0	1.0	0.0	1.0	0.7	0.3	0.0
x'' (vermelho) L	1.0	0.0	1.0	0.0	0.0	0.0	1.0	0.5	0.5	0.8	0.2	0.0
x''' (violeta) L	1.0	0.0	1.0	0.0	0.0	0.0	1.0	0.0	1.0	1.0	0.0	0.0





WM: tabela de Matching: Ponto x Partição

	STG		SCG			STR		LPR		PEG		
	A	B	A	B	C	A	B	A	B	A	B	C
x' (verde) M	<u>1.0</u>	0.0	<u>0.8</u>	0.2	0.0	0.0	<u>1.0</u>	0.0	<u>1.0</u>	<u>0.7</u>	0.3	0.0
x'' (vermelho) L	<u>1.0</u>	0.0	<u>1.0</u>	0.0	0.0	0.0	<u>1.0</u>	0.5	<u>0.5</u>	<u>0.8</u>	0.2	0.0
x''' (violeta) L	<u>1.0</u>	0.0	<u>1.0</u>	0.0	0.0	0.0	<u>1.0</u>	0.0	<u>1.0</u>	<u>1.0</u>	0.0	0.0

Valores <u>**destacados**</u> indicam a <u>**maior compatibilidade**</u> (maior Matching) dentro da partição de cada variáveil

WM: Gerando Uma Regra Para Cada Ponto

	STG		SCG			STR		LPR		PEG		
	A	B	A	B	C	A	B	A	B	A	B	C
x' (verde) M	1.0	0.0	0.8	0.2	0.0	0.0	1.0	0.0	1.0	0.7	0.3	0.0
x'' (vermelho) L	1.0	0.0	1.0	0.0	0.0	0.0	1.0	0.5	0.5	0.8	0.2	0.0
x''' (violeta) L	1.0	0.0	1.0	0.0	0.0	0.0	1.0	0.0	1.0	1.0	0.0	0.0

Para x’: SE STG=A E SCG=A E STR=B E LPR=B E PEG=A ENTAO Classe=Middle

Para x”: SE STG=A E SCG=A E STR=B E LPR=B E PEG=A ENTAO Classe=Low

Para x’”: SE STG=A E SCG=A E STR=B E LPR=B E PEG=A ENTAO Classe=Low

WM: Definindo nível de ativação (μ) da regra

	STG		SCG			STR		LPR		PEG		
	A	B	A	B	C	A	B	A	B	A	B	C
x' (verde) M	1.0		0.8				1.0		1.0	0.7		
x'' (vermelho) L	1.0		1.0				1.0		0.5	0.8		
x''' (violeta) L	1.0		1.0				1.0		1.0	1.0		

E=min

$\mu(x')=\min(1,0.8,1,1,0.7)=0.7$

$\mu(x'')=\min(1,1,1,0.5,0.7)=0.5$

$\mu(x''')=\min(1,1,1,1,1)=1.0$

Para x': SE STG=A E SCG=A E STR=B E LPR=B E PEG=A ENTAO Classe=Middle ($\mu(x')=0.7$)

Para x'': SE STG=A E SCG=A E STR=B E LPR=B E PEG=A ENTAO Classe=Low ($\mu(x'')=0.5$)

Para x''': SE STG=A E SCG=A E STR=B E LPR=B E PEG=A ENTAO Classe=Low ($\mu(x''')=1.0$)

WM: Definindo a regra vencedora (maior μ)

	STG		SCG			STR		LPR		PEG		
	A	B	A	B	C	A	B	A	B	A	B	C
x' (verde) M	1.0		0.8				1.0		1.0	0.7		
x'' (vermelho) L	1.0		1.0				1.0		0.5	0.8		
x''' (violeta) L	1.0		1.0				1.0		1.0	1.0		

E=min

$\mu(x')=\min(1,0.8,1,1,0.7)=0.7$

$\mu(x'')=\min(1,1,1,0.5,0.7)=0.5$

$\mu(x''')=\min(1,1,1,1,1)=1.0$

SE STG=A E SCG=A E STR=B E LPR=B E PEG=A ENTAO Classe=Middle
$(\mu(x')=0.7)$

SE STG=A E SCG=A E STR=B E LPR=B E PEG=A ENTAO Classe=Low
$(\mu(x'')=0.5)$

SE STG=A E SCG=A E STR=B E LPR=B E PEG=A ENTAO Classe=Low
$(\mu(x''')=1.0)$

WM: Repetir o processo para outros pontos

		STG		SCG			STR		LPR		PEG		
		A	B	A	B	C	A	B	A	B	A	B	C
Xa	VH	0.8				0.8		1.0		1.0	0.8		
Xb	L	0.7				0.9		0.9		0.5	0.7		
Xc	L	0.8				0.8		0.7		1.0	0.8		

E=min

μ(xa)=min(.8,.8,1,1,0.8)=0.8

μ(xb)=min(.7,.9,.9,.5,0.7)=0.5

μ(xc)=min(.8,.8,.7,1,.8)=0.7

Para xa: SE STG=A E SCG=C E STR=B E LPR=B E PEG=A ENTAO
Classe=Middle (μ(x')=0.8)

Para xb: SE STG=A E SCG=C E STR=B E LPR=B E PEG=A ENTAO
Classe=Low (μ(x'')=0.5)

Para x_c: SE STG=A E SCG=C E STR=B E LPR=B E PEG=A ENTAO
Classe=Low (μ(x''')=0.7)