

4

Modelo para Extração de Regras Fuzzy a partir de Máquinas de Vetor Suporte – FREx_SVM

4.1. Introdução

Como já mencionado, em máquinas de vetor suporte não se pode explicar a maneira como sua saída é obtida. No entanto, em inúmeras aplicações, é fundamental conhecer não somente a decisão de classificação, mas também saber de forma mais explícita como se chegou a essa decisão. Assim, é importante extrair conhecimento explícito das SVM. Para isso, define-se um modelo de extração de regras para revelar o conhecimento de uma SVM treinada.

Existem vários métodos de extração de regras a partir de redes neurais treinadas [44], [45], [46], [47], [48], [49]. No entanto, para o caso de SVM, o assunto ainda não foi suficientemente explorado [13], [14] e [15]. As regras extraídas pelos métodos existentes são simbólicas e, por isso, menos flexíveis.

Este trabalho tem como proposta a construção de um modelo de extração de regras fuzzy para a classificação em duas ou mais classes, denominado FREx_SVM, usando as projeções dos vetores suporte de uma SVM treinada.

Inicialmente é desenvolvido um modelo de extração de regras para o caso de classificação binária, exposto na seção 4.2 e que, em linhas gerais, consiste em:

- Treina-se uma SVM e obtêm-se os vetores suportes;
- Para cada vetor suporte acham-se suas projeções nos eixos coordenados;
- Define-se um número pré-determinado de conjuntos fuzzy para cada uma das coordenadas;
- A partir dos conjuntos fuzzy e dos vetores suporte geram-se as regras fuzzy.

Em seguida, na seção 4.3, o modelo é estendido para o caso de classificação em mais de duas classes, e que, em linhas gerais, consiste em:

- Escolhe-se um dos métodos de classificação em k classes;
- Treina-se uma SVM e obtêm-se os vetores suportes;
- Para cada vetor suporte, acham-se suas projeções nos eixos coordenados;
- Define-se um número pré-determinado de conjuntos fuzzy para cada uma das coordenadas;
- A partir dos conjuntos fuzzy e dos vetores suporte, geram-se as regras fuzzy.

Vale a pena mencionar que a opção pelo trabalho com vetores suporte foi motivada pelo fato de que, como já foi visto, o hiperplano ótimo de separação obtido pela SVM é definido em função dos vetores suporte. Assim, pode-se dizer que um vetor suporte representa bem a classe à qual pertence.

4.2. Extração de Regras – Caso Binário

O modelo de extração de regras de SVM binária é dividido em três etapas: obtenção das projeções dos vetores suporte, definição dos conjuntos fuzzy e extração das regras fuzzy. Essas etapas estão expostas detalhadamente nas seções 4.2.1, 4.2.2 e 4.2.3. A Figura 12 mostra um diagrama com um resumo do funcionamento do modelo, desde o treinamento da SVM até a obtenção das regras fuzzy.

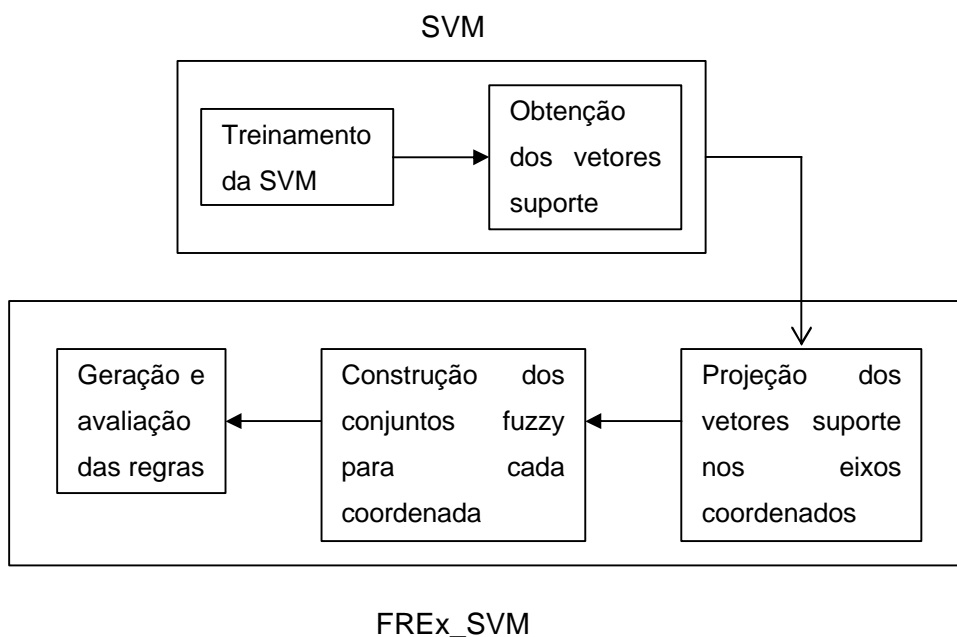


Figura 12 - Modelo de Extração de Conhecimento Explícito de SVM

4.2.1. Etapa 1 Obtenção das projeções dos vetores suporte

A etapa 1 do algoritmo consiste na projeção dos vetores suporte obtidos pela classificação com a SVM binária nas coordenadas cartesianas. O número de projeções é igual à dimensão do espaço de entradas (número de atributos de entrada). Considere o caso bidimensional e seja $s = (0,61, 0,88)$ um vetor suporte de uma certa classe. A projeção desse vetor suporte no eixo x é 0,61 e no eixo y é 0,88. A Figura 13 mostra essas projeções.

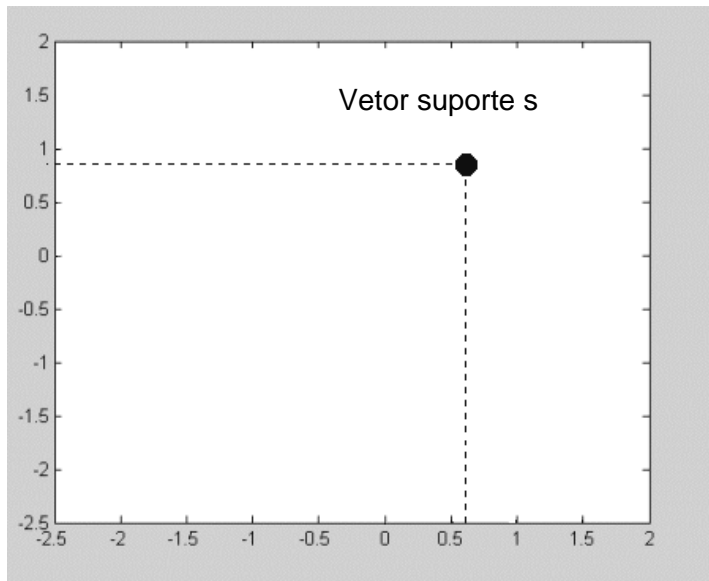


Figura 13 - Projeção do vetor suporte s

4.2.2. Etapa 2 Definição dos conjuntos fuzzy

A etapa 2 do algoritmo consiste na construção dos conjuntos fuzzy. Como já foi mostrado em [52] e [53], é preferível construir os conjuntos previamente a defini-los através dos dados. Assim, estabelece-se um número determinado de conjuntos fuzzy triangulares de tamanho fixo para todas coordenadas. Suponha-se que a dimensão do espaço de entrada seja n e que sejam definidos t conjuntos fuzzy para cada variável. O conjunto fuzzy C_{ij} é o j -ésimo conjunto definido na i -ésima coordenada, onde $i \in \{1, \dots, n\}$ e $j \in \{1, \dots, t\}$. A Figura 14 mostra a construção de 5 conjuntos fuzzy para $n = 2$.

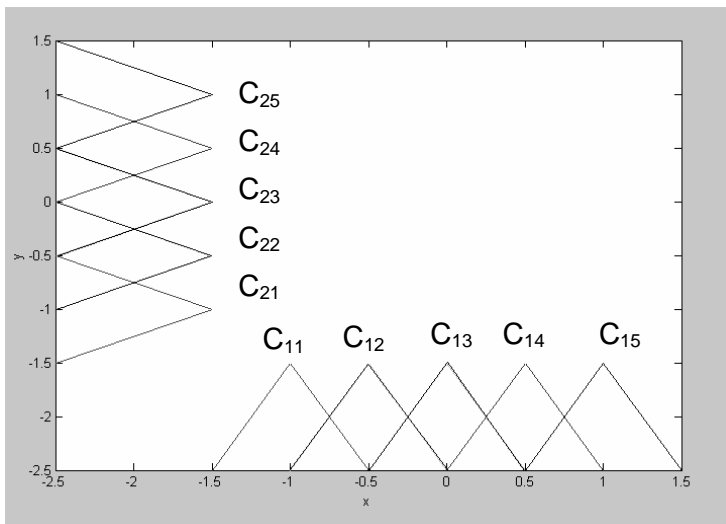


Figura 14 - Definição dos conjuntos fuzzy

Em seguida, para a projeção de cada vetor suporte em uma dada coordenada, são encontrados os graus de pertinência a cada um dos conjuntos fuzzy e é identificado o conjunto com o maior grau de pertinência. Isso é efetuado da seguinte forma: sejam x_i a projeção do vetor suporte x na i -ésima coordenada e $m_{C_{ij}}(x_i)$ o grau de pertinência de x_i ao conjunto C_{ij} . Calcula-se, então, $\max_{j \in \{1, \dots, t\}} \{m_{C_{ij}}(x_i)\}$ e determina-se, assim, o conjunto C_{ij} em que o grau de pertinência é máximo. Conforme indicado na Figura 15, $m_{C_{14}}(x_1) = 0,78$, $m_{C_{15}}(x_1) = 0,22$, $m_{C_{24}}(x_2) = 0,24$ e $m_{C_{25}}(x_2) = 0,76$, onde $x_1 = 0,61$ é a projeção do vetor suporte s no eixo x e onde $x_2 = 0,88$ é a projeção no eixo y . Os conjuntos em que os graus de pertinência são máximos, C_{14} e C_{25} , estão destacados com um quadrado ao redor. Para uma melhor visualização, os conjuntos C_{ij} em que o grau de pertinência de x_i é igual a 0 não aparecem na Figura 15.

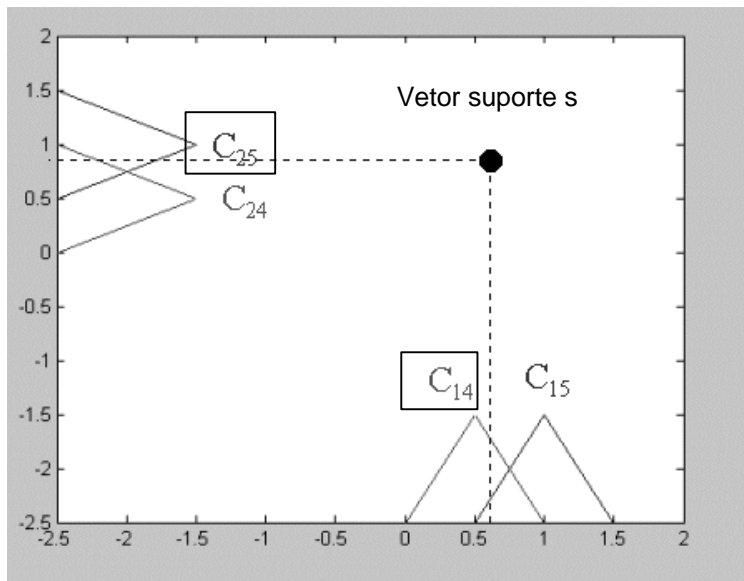


Figura 15 - Obtenção dos conjuntos fuzzy com maior grau de pertinência

4.2.3. Etapa 3 Extração de Regras Fuzzy

Nessa etapa, cada vetor suporte gera uma regra fuzzy da forma explicada a seguir.

Para cada vetor suporte x , seja C_{ij} o conjunto fuzzy da i -ésima coordenada no qual o grau de pertinência de x_i é máximo, como definido na etapa anterior, onde $j \in \{1, \dots, t\}$. Seja $p = (p_1, \dots, p_n)$ um ponto de \mathbb{R}^n . A regra gerada pelo vetor suporte x será:

r : Se $\{p_1 \text{ é } C_{1j_1}, \dots, p_k \text{ é } C_{kj_n}\}$, então p é da classe do vetor suporte x .

Na Figura anterior a regra gerada é:

Se $\{p_1 \text{ é } C_{14} \text{ e } p_2 \text{ é } C_{25}\}$, então p é da classe do vetor suporte s .

Como já foi dito, são geradas tantas regras quanto forem os vetores suporte.

Pode ocorrer que dois vetores suporte de classes diferentes gerem regras conflitantes (mesmo antecedente e conseqüentes diferentes). Para resolver esse problema são definidas duas métricas (seções 4.2.3.1 e 4.2.3.2): abrangência fuzzy e acurácia fuzzy - extensões das definições apresentadas em [16]. Assim, quando são geradas regras conflitantes, a regra que possui maior acurácia fuzzy é incluída no conjunto de regras e as outras são excluídas.

Para ilustrar o funcionamento do algoritmo FREx_SVM, considera-se um exemplo simples, mostrado na Tabela 2, que consiste em seis padrões (P1, ..., P6) de dois atributos (peso e altura) definidos em duas classes -1 e 1. Após o treinamento da SVM, determinam-se P1 e P3, da classe -1, e P2, da classe 1, como vetores suportes (na Tabela 2, em negrito).

Tabela 2 - Exemplo para ilustrar o funcionamento do FREx_SVM

| Pontos | Peso | Altura | Classe |
|-----------|-----------|-------------|-----------|
| P1 | 60 | 1,75 | -1 |
| P2 | 65 | 1,60 | 1 |
| P3 | 70 | 1,80 | -1 |
| P4 | 76 | 1,85 | -1 |
| P5 | 80 | 1,65 | 1 |
| P6 | 85 | 1,68 | 1 |

Para cada coordenada, dois conjuntos fuzzy triangulares e complementares são construídos. Usando-se a notação previamente utilizada, para o atributo peso

tem-se os conjuntos C_{11} e C_{12} , e para o atributo altura, C_{21} e C_{22} . A Figura 16 mostra os conjuntos fuzzy para os atributos 1 e 2. Os graus de pertinência resultantes para cada atributo de entrada são mostrados nas Tabelas 3 e 4.

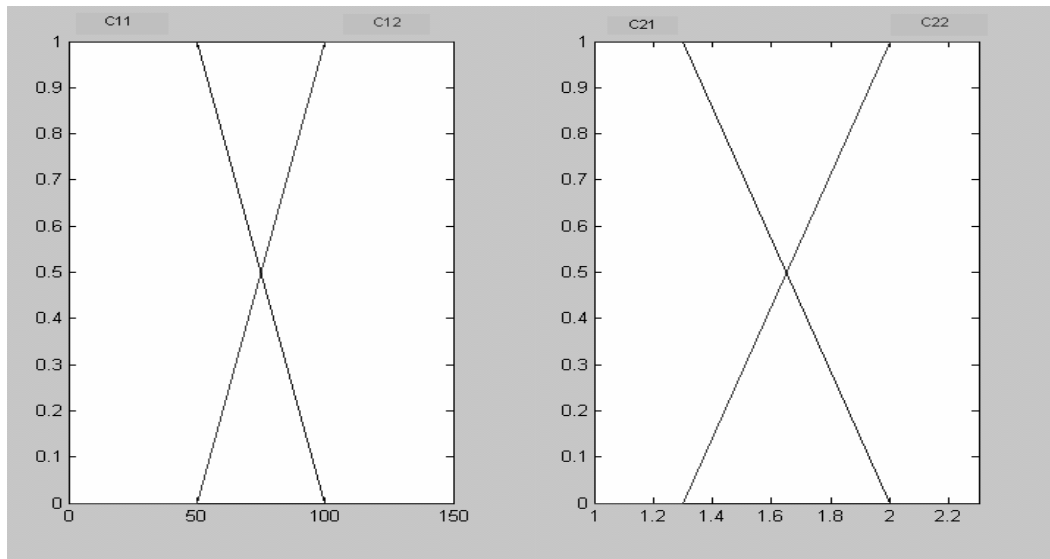


Figura 16 - Conjuntos fuzzy definidos para os atributos peso e altura

O conjunto com grau de pertinência máximo da projeção do vetor suporte P1 na primeira coordenada é C_{11} e na segunda coordenada é C_{22} . Assim, P1 gera a seguinte regra:

Regra 1: Se $\{x_1 \text{ é } C_{11} \text{ e } x_2 \text{ é } C_{22}\}$, então $x = (x_1, x_2)$ é da classe -1.

O vetor suporte P2 tem maior grau de pertinência para os conjuntos C_{11} e C_{21} . Assim, P2 gera a regra:

Regra 2: Se $\{x_1 \text{ é } C_{11} \text{ e } x_2 \text{ é } C_{21}\}$, então $x = (x_1, x_2)$ é da classe 1.

Para P3, os conjuntos com os graus de pertinência máxima das suas projeções são C_{11} e C_{22} . A regra gerada é

Regra 3: Se $\{x_1 \text{ é } C_{11} \text{ e } x_2 \text{ é } C_{22}\}$, então $x = (x_1, x_2)$ é da classe -1.

Tabela 3 - Grau de pertinência para o 1º atributo

| C_{11} | μ | C_{12} | μ |
|----------|-------|----------|-------|
| 60 | 0,8 | 60 | 0,2 |
| 65 | 0,7 | 65 | 0,3 |
| 70 | 0,6 | 70 | 0,4 |
| 76 | 0,48 | 76 | 0,52 |
| 80 | 0,4 | 80 | 0,6 |
| 85 | 0,3 | 85 | 0,7 |

Como pode ser visto, os vetores suporte P1 e P3 da classe -1 geram a mesma regra. Além disso, apesar de existirem seis padrões de entrada, que poderiam produzir seis regras, apenas três delas são geradas.

Tabela 4 - Grau de pertinência para o 2^o atributo

| C ₂₁ | μ | C ₂₂ | μ |
|-----------------|-------|-----------------|-------|
| 1,75 | 0,42 | 1,75 | 0,58 |
| 1,60 | 0,67 | 1,60 | 0,33 |
| 1,80 | 0,33 | 1,80 | 0,67 |
| 1,85 | 0,25 | 1,85 | 0,75 |
| 1,65 | 0,58 | 1,65 | 0,42 |
| 1,68 | 0,46 | 1,68 | 0,54 |

A seguir são apresentadas as definições de acurácia e abrangência fuzzy, que são as métricas mais utilizadas para avaliação de regras, propostas para o modelo FReX_SVM. Como anteriormente mencionado, essas definições são adaptações da definição de Lanas [16].

4.2.3.1. Acurácia Fuzzy

A acurácia de uma regra fuzzy mede quão bem ela descreve os dados. Para o problema de classificação em k classes, a acurácia fuzzy de uma regra r associada à classe c , $c \in \{1, 2, \dots, k\}$, é definida por

$$FA_c^r = \frac{\sum_{i=1}^{m_c} m_r(x_i)}{\sum_{j=1}^m m_r(x_j)}, \quad (36)$$

onde m_r é o produto dos graus de pertinência dos padrões a cada conjunto presente no antecedente da regra r , m_c é o número de padrões na classe c e m é o número total de padrões.

Para cada regra r , a soma das acurácias FA_c^r , $c = \{1, \dots, k\}$, é igual a 1.

A forma como se define a acurácia fuzzy depende do número de padrões por classe. Assim, se houver um banco de dados em que a diferença entre o número de padrões por classe seja grande, a acurácia fuzzy pode conduzir a falsos resultados. Logo, se o banco de dados tem um número diferente de padrões por classe, é necessário definir um fator de correção CF para compensar a distribuição não uniforme dos padrões:

$$CF_c^r = \frac{1}{m_c \left(\sum_{j=1}^k \frac{FA_j^r}{m_j} \right)}. \quad (37)$$

A nova acurácia fuzzy é, então, definida através da equação (38).

$$(FA_c^r)^* = FA_c^r \cdot CF_c^r. \quad (38)$$

Para cada regra r , a soma das novas acurácias $(FA_c^r)^*$, $c = \{1, \dots, k\}$, é

$$\sum_{i=1}^k (FA_i^r)^* = \sum_{i=1}^k (FA_i^r)(CF_i^r) = \sum_{i=1}^k FA_i^r \left(\frac{1}{m_i \sum_{j=1}^k \frac{FA_j^r}{m_j}} \right) = \left(\sum_{i=1}^k \frac{FA_i^r}{m_i} \right) \left(\frac{1}{\sum_{j=1}^k \frac{FA_j^r}{m_j}} \right) = 1$$

Para as regras geradas no exemplo apresentado na Tabela 2, as acurácias fuzzy são²:

Regra 1

$$FA_{-1}^1 = \frac{0,8 * 0,58 + 0,6 * 0,67 + 0,48 * 0,75}{0,8 * 0,58 + 0,7 * 0,33 + 0,6 * 0,67 + 0,48 * 0,75 + 0,4 * 0,42 + 0,3 * 0,54}$$

$$= \frac{1,23}{1,79} = 0,69,$$

$$FA_1^1 = \frac{0,7 * 0,33 + 0,4 * 0,42 + 0,3 * 0,54}{0,8 * 0,58 + 0,7 * 0,33 + 0,6 * 0,67 + 0,48 * 0,75 + 0,4 * 0,42 + 0,3 * 0,54}$$

$$= \frac{0,56}{1,79} = 0,31.$$

Regra 2

$$FA_{-1}^2 = \frac{0,8 * 0,42 + 0,6 * 0,33 + 0,48 * 0,25}{0,8 * 0,42 + 0,7 * 0,67 + 0,6 * 0,33 + 0,48 * 0,25 + 0,4 * 0,58 + 0,3 * 0,46}$$

$$= \frac{0,65}{1,49} = 0,44,$$

$$FA_1^2 = \frac{0,7 * 0,67 + 0,4 * 0,58 + 0,3 * 0,46}{0,8 * 0,42 + 0,7 * 0,67 + 0,6 * 0,33 + 0,48 * 0,25 + 0,4 * 0,58 + 0,3 * 0,46}$$

$$= \frac{0,84}{1,49} = 0,56.$$

Os dados no exemplo são distribuídos uniformemente nas duas classes; assim o fator de correção é 1 para ambas as classes.

² Como a regra 1 e a regra 3 representam a mesma regra, só se calcula a acurácia fuzzy das regras 1 e 2.

4.2.3.2. Abrangência Fuzzy

A abrangência fuzzy mede o número de padrões afetados pela regra. A definição da abrangência fuzzy da regra r é

$$FC^r = \frac{\sum_{j=1}^m \mathbf{m}_r(x_j)}{m}, \quad (39)$$

onde $\mathbf{m}(x_j)$ e m foram definidos na seção anterior.

Usando o mesmo exemplo, a abrangência fuzzy para as regras 1 e 2 é

$$\begin{aligned} FC^1 &= \frac{0,8 * 0,58 + 0,7 * 0,33 + 0,6 * 0,67 + 0,48 * 0,75 + 0,4 * 0,42 + 0,3 * 0,54}{6} \\ &= \frac{1,79}{6} = 0,30, \end{aligned}$$

$$\begin{aligned} FC^2 &= \frac{0,8 * 0,42 + 0,7 * 0,67 + 0,6 * 0,33 + 0,48 * 0,25 + 0,4 * 0,58 + 0,3 * 0,46}{6} \\ &= \frac{1,49}{6} = 0,25. \end{aligned}$$

4.3. Extração de Regras – Caso de k classes, $k > 2$

Com o objetivo de extrair regras para a classificação em k classes com SVM, deve-se primeiro escolher um entre os métodos de classificação apresentados na seção 2.3 e, em seguida, treinar a SVM com os dados de treinamento pré-processados. Deste modo, obtêm-se os vetores suporte associados a cada classe. A Figura 17 mostra um diagrama do modelo de extração de regras para a classificação em múltiplas classes.

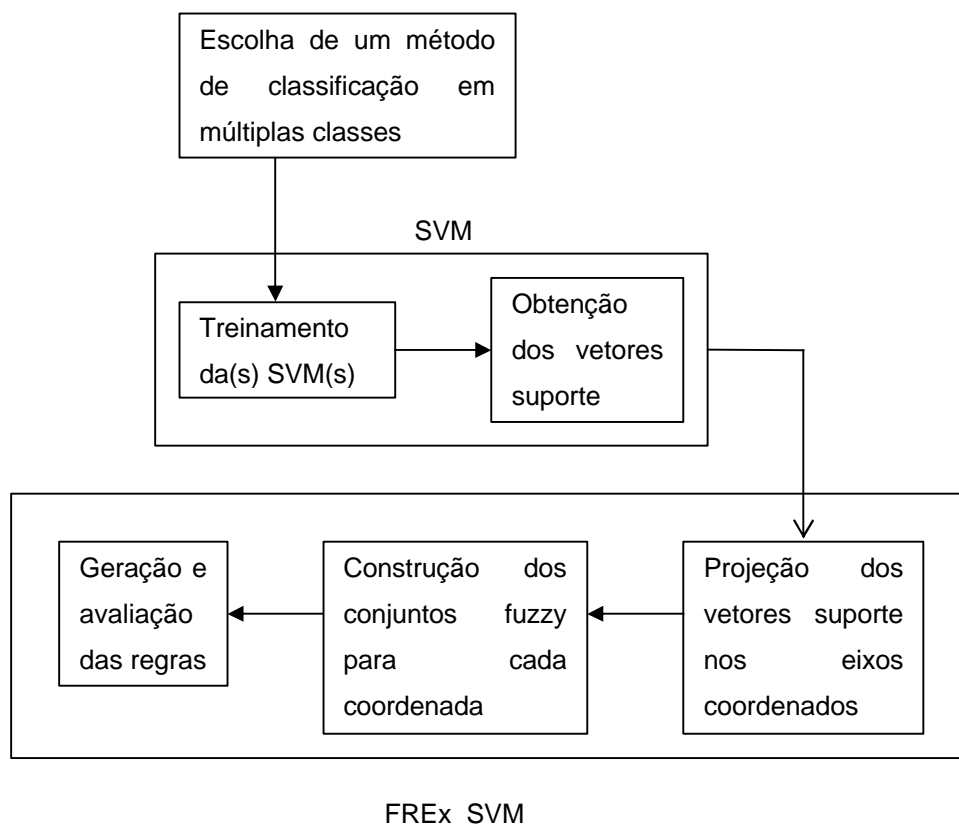


Figura 17 - Modelo de Extração de Regras para Classificação em Múltiplas Classes

Cada método de classificação gera, a princípio, um conjunto de regras. Esses conjuntos de regras são avaliados em relação à acurácia e à abrangência fuzzy. Assim, pode-se verificar qual método de classificação em múltiplas classes gera melhores conjuntos de regras quanto a essas medidas.

Como mencionado, o método de decomposição um por classe constrói k SVMs para separar cada classe de todas as outras. Seja SVM_i a que separa a classe i das outras, $i = \{1, \dots, k\}$. Para essa SVM, somente os vetores suporte da classe i são considerados para a construção das regras, já que esses vetores suporte auxiliam na definição da classe i . A informação contida em um vetor

suporte da classe j , $j \neq i$, não contribui para definir a classe j ; apenas serve para excluir a classe i .

No método de separação das classes duas a duas, é construída uma SVM para separar as classes i e j , $j \neq i$. Nesse caso, todos os vetores suporte obtidos ajudam a definir as classes i e j e, por isso, são usados na geração das regras fuzzy.

Também para o método de Crammer e Singer, todos os vetores suporte obtidos são usados já que eles, nesses casos, ajudam a definir as classes.