3
Seleção de Variáveis Baseada em Informação Mútua sob Distribuição de Informação Uniforme (MIFS-U)

3.1
Seleção de Variáveis de Entrada para Problemas de Classificação

A seleção de variáveis de entrada desempenha um importante papel em sistemas de classificação tais como redes neurais artificiais (RNAs). Tais variáveis podem ser classificadas como pertinentes, irrelevantes ou redundantes, e, do ponto de vista do gerenciamento de um conjunto de dados, que pode ser gigantesco, reduzir o número de variáveis, selecionando somente aquelas pertinentes, é extremamente desejável. Pois, desse modo, melhor desempenho com menor esforço computacional é esperado (Kwak & Choi, 2002).

Hosmer & Lemeshow (1989) destacam a importância da seleção de variáveis, pois, com um menor número de variáveis, o modelo tende a ser mais generalizável e robusto.

Problemas de seleção de variáveis foram pesquisados por vários autores como Battiti (1994), Joliffe (1986) e Agrawal et al (1993). Um dos métodos mais populares para lidar com este problema é o de análise de componentes principais (*principal component analysis* – PCA) (Joliffe, 1986). Entretanto, quando se objetiva preservar os dados originais, este método não é adequado.

O algoritmo abordado nesta dissertação, a saber, o MIFS-U – Seletor de Variáveis sob Informação Mútua com Distribuição Uniforme de Informação – apresentado por Kwak & Choi (2002), objetivou superar a limitação do seletor de variáveis proposto por Battiti (1994), gerando melhor desempenho no processo de seleção de variáveis. Tal algoritmo, por sua simplicidade, pode ser usado em qualquer sistema de classificação, seja qual for o algoritmo de aprendizagem. No entanto, seu desempenho pode ser prejudicado como resultado de erros na estimação da informação mútua.

3.2
Problema FRn – k e o Algoritmo de Seleção Ideal

No processo de selecionar variáveis de entrada, é desejável reduzir o número de variáveis, excluindo aquelas que são irrelevantes ou redundantes. Este conceito é formalizado pela seleção das k variáveis mais relevantes de um conjunto de n variáveis, chamado de problema de "redução de variável" (*feature redution* – FR) (Battiti, 1994). Tal processo é descrito a seguir:

[FRn – k]: Dado um conjunto inicial de n variáveis, encontre o subconjunto com k < n variáveis que representa "a máxima informação" acerca do desfecho (variável explicada, de resultado ou de saída).

Como visto anteriormente, a informação mútua entre duas variáveis aleatórias diz respeito à quantidade de informação comum entre essas variáveis. O problema de selecionar variáveis de entrada pode ser resolvido calculando a informação mútua (IM) entre variáveis de entrada e o desfecho. Se a informação mútua entre variáveis de entrada e desfecho pudesse ser obtida com exatidão, o problema FRn – k poderia ser reformulado como segue:

[FRn – k]: Dados um conjunto inicial F com n variáveis e o desfecho D, encontre o subconjunto $S \subset F$ com k variáveis que minimiza $H(D|S)$, isto é, que maximiza a informação mútua $I(D;S)$. O método de seleção adotado aqui é conhecido como "seleção gulosa[2]". Neste método, a partir do conjunto vazio de variáveis selecionadas, adiciona-se a melhor variável de entrada viável ao conjunto anterior, uma a uma, até que o tamanho do conjunto atinja k. Esse algoritmo de seleção ideal que usa informação mútua é realizado como segue:

1) (Inicialização) conjunto $F \leftarrow$ "conjunto inicial com n variáveis", $S \leftarrow$ "conjunto vazio."

2) (Cálculo da IM com o desfecho), $\forall \phi_i \in F$, compute $I(D; \phi_i)$.

[2] O termo "gulosa" (em inglês, "greedy") vem do fato de que a seleção é feita passo a passo, não voltando atrás.

3) (Seleção da primeira variável) ache a variável que maximiza $I(D; \phi_i)$, faça $F \leftarrow F \setminus \{\phi_i\}$, $S \leftarrow \{\phi_i\}$.

4) (Seleção gulosa) repita até que seja alcançado o número desejado de variáveis selecionadas:

a) (Cálculo da IM conjunta entre variáveis), $\forall \phi_i \in F$, compute $I(D; \phi_i,S)$.

b) (Seleção da próxima variável) escolha a variável $\phi_i \in F$ que maximiza $I(D; \phi_i, S)$ e faça $F \leftarrow F \setminus \{\phi_i\}$, $S \leftarrow \{\phi_i\}$.

5) Finalize o conjunto S contendo as variáveis selecionadas.

Na prática, a realização deste algoritmo torna-se inviável, face à alta dimensão do vetor de variáveis no cálculo de $I(D; \phi_i,S)$, visto que, tendo como objetivo a seleção de k (k<n) variáveis, o vetor S (composto pela variáveis já selecionadas), atinge dimensão igual a (k – 1). Para superar esse óbice, Batiti (1994) propôs um método alternativo, apresentado a seguir.

3.3
Seleção de Variáveis Baseada em Informação Mútua (MIFS)

O algoritmo MIFS é igual ao algoritmo anterior, com exceção do passo 4. Em vez de calcular $I(D; \phi_i,S)$, Batiti (1994) usou somente $I(D; \phi_i)$ e $I(\phi_i; \phi_s)$. Ou seja, para ser selecionada, uma variável não pode ser previsível a partir daquelas já selecionadas em S, e deve ser informativa em relação ao desfecho. No MIFS, o passo 4 no algoritmo de seleção anterior foi substituído como segue:

4) (Seleção gulosa) repita até que seja alcançado o número desejado de variáveis selecionadas:

a) (Cálculo da IM entre variáveis), para todos os pares de variáveis (ϕ_i, ϕ_s) com $\phi_i \in F$ e $\phi_s \in S$, compute $I(\phi_i; \phi_s)$, se ainda não foi avaliada.

b) (Seleção da próxima variável) escolha a variável $\phi_i \in F$ que maximiza $I(D;\phi_i) - \beta \sum_{\phi_s \in S} I(\phi_i;\phi_s)$ e faça $F \leftarrow F \setminus \{\phi_i\}$, $S \leftarrow \{\phi_i\}$.

Onde β é o parâmetro de redundância em relação às variáveis de entrada. Se $\beta = 0$, o algoritmo seleciona variáveis na ordem da informação mútua entre variáveis de entrada e o desfecho, e a redundância entre variáveis de entrada, portanto, nunca é refletida. Quando $\beta > 0$, a redundância se reduz.

A relação entre variáveis de entrada e desfecho pode ser representada na Figura 3.1. O primeiro algoritmo de seleção usa a informação mútua para escolher a variável ϕ_i que maximiza a informação mútua conjunta $I(D;\phi_i,\phi_s)$ que corresponde às áreas II, III, e IV. Como $I(D;\phi_s)$ (áreas II e IV) é comum a todas as variáveis não selecionadas ϕ_i no cálculo da informação mútua conjunta $I(D; \phi_i, \phi_s)$, o algoritmo seleciona a variável ϕ_i que maximiza a área III. Por outro lado, o algoritmo MIFS seleciona a variável que maximiza $I(D;\phi_i) - \beta \sum_{\phi_s \in S} I(\phi_i;\phi_s)$. Para β = 1, isso corresponde a área III subtraída da área I.

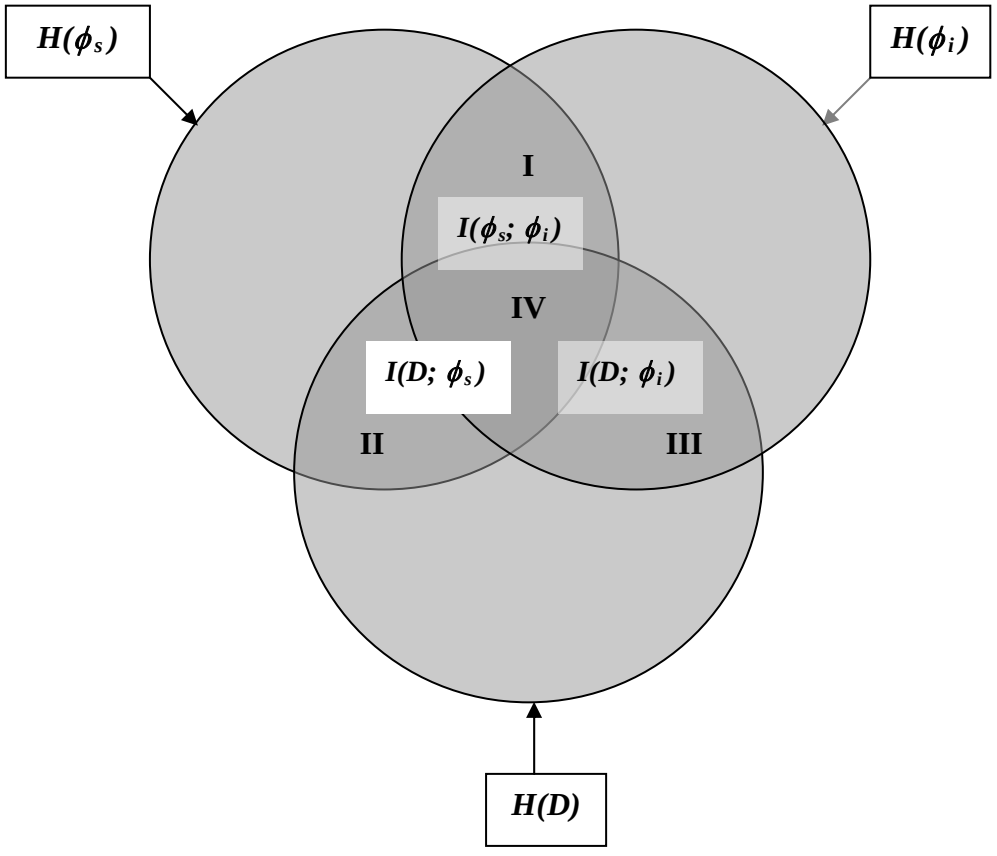

Figura 3.1 – Relação entre variáveis de entrada e desfecho

Portanto, se uma variável a ser selecionada é fortemente relacionada à alguma variável já selecionada, a área I é grande, podendo, pois, degradar o desempenho do algoritmo. Por essa razão, o MIFS não lida muito bem com problemas não-lineares.

3.4 O Seletor de Variáveis MIFS-U

Este algoritmo de seleção é mais próximo do algoritmo ideal (veja seção 3.2) do que o MIFS. O algoritmo ideal tenta maximizar $I(D;\phi_i,\phi_s)$ (áreas II, III e IV na Figura 3.1), e isto pode ser reescrito como

$$I(D;\phi_i,\phi_s) = I(D;\phi_s) + I(D;\phi_i \mid \phi_s). \quad (3.1)$$

Onde $I(D;\phi_i \mid \phi_s)$ representa a informação mútua restante entre o desfecho D e a variável ϕ_i para uma dada variável ϕ_s, o que corresponde à área III na Figura 3.1, enquanto que a área II mais a área IV representam $I(D;\phi_s)$. Visto que $I(D;\phi_s)$ é comum a todas as variáveis candidatas a serem selecionadas no algoritmo ideal, não há nenhuma necessidade nesse cálculo. Assim, o algoritmo ideal tenta achar a variável que maximiza $I(D;\phi_i \mid \phi_s)$ (área III). No entanto, calcular $I(D;\phi_i \mid \phi_s)$ requer tanto esforço quanto calcular $I(D; \phi_i; \phi_s)$.

Assim, $I(D;\phi_i \mid \phi_s)$ é calculada de forma aproximada através de $I(\phi_i;\phi_s)$ e $I(D;\phi_i)$, que são relativamente fáceis de calcular. A informação mútua condicional pode ser representada como

$$I(D;\phi_i \mid \phi_s) = I(D;\phi_i) - \{I(\phi_i;\phi_s) - I(\phi_i;\phi_s \mid D)\} \quad (3.2)$$

Onde $I(\phi_i;\phi_s)$ corresponde às áreas I e IV, e $I(\phi_i;\phi_s \mid D)$, à área I. De modo que o termo $I(\phi_i;\phi_s) - I(\phi_i;\phi_s \mid D)$ corresponde à área IV. O termo $I(\phi_i;\phi_s \mid D)$ significa a informação mútua entre a variável já selecionada ϕ_s e a

variável candidata ϕ_i para um dado desfecho *D*. Se o condicionamento pelo desfecho *D* não altera a razão entre a entropia de ϕ_s e a informação mútua entre ϕ_i e ϕ_s, ou seja, se vale a seguinte relação (condição do algoritmo):

$$\frac{H(\phi_s \mid D)}{H(\phi_s)} = \frac{I(\phi_i;\phi_s \mid D)}{I(\phi_i;\phi_s)} \tag{3.3}$$

$I(\phi_i;\phi_s \mid D)$ pode ser representada por

$$I(\phi_i;\phi_S \mid D) = \frac{H(\phi_s \mid D)}{H(\phi_s)}\, I(\phi_i;\phi_s) \tag{3.4}$$

Usando a equação acima e a Eq. (3.2), obtém-se o seguinte:

$$\begin{aligned} I(D;\phi_i \mid \phi_s) &= I(D;\phi_i) - \left(1 - \frac{H(\phi_s \mid D)}{H(\phi_s)}\right) I(\phi_i;\phi_s) = \\ &= I(D;\phi_i) - \left(\frac{H(\phi_s) - H(\phi_s \mid D)}{H(\phi_s)}\right) I(\phi_i;\phi_s) = \\ &= I(D;\phi_i) - \frac{I(D;\phi_s)}{H(\phi_s)}\, I(\phi_i;\phi_s). \end{aligned} \tag{3.5}$$

Considerando que cada região na Figura 3.1 corresponda a sua respectiva informação, a condição apresentada na Eq. (3.3) é mais difícil de ser satisfeita quando a informação está concentrada em uma das seguintes regiões: $H(\phi_s \mid \phi_i; D)$, $I(\phi_s;\phi_i \mid D)$, $I(D;\phi_s \mid \phi_i)$ ou $I(D;\phi_s;\phi_i)$. É mais provável que a condição (3.3) seja válida quando a informação está distribuída uniformemente ao longo da região de $H(\phi_s)$ na Figura 3.1. Razão pela qual, o algoritmo é referido, simplesmente, por MIFS-U.

Com base no ora exposto, o passo 4 revisado do algoritmo de seleção ideal toma, assim, a seguinte forma:

4) (Seleção gulosa) repita até que seja alcançado o número desejado de variáveis selecionadas:

a) (Cálculo da entropia) $\forall \phi_s \in S$, compute $H(\phi_s)$, se ainda não foi avaliada.

b) (Cálculo da IM entre variáveis), para todos os pares de variáveis (ϕ_i, ϕ_s) com $\phi_i \in F$ e $\phi_s \in S$, compute $I(\phi_i; \phi_s)$, se ainda não foi avaliada.

c) (Seleção da próxima variável) escolha a variável $\phi_i \in F$ que maximiza $I(D;\phi_i) - \beta \sum_{\phi_s \in S} \left(I(D;\phi_s) / H(\phi_s)\right) I(\phi_i;\phi_s)$ e faça $F \leftarrow F \setminus \{\phi_i\}$, $S \leftarrow \{\phi_i\}$.

O parâmetro β oferece flexibilidade ao algoritmo tal como no MIFS. Caso se adote β igual a zero, a informação mútua entre as variáveis de entrada não é levada em consideração e o algoritmo seleciona tais variáveis na ordem da informação mútua de cada uma delas com o desfecho. A redundância entre as variáveis de entrada não é, portanto, refletida. Quando β aumenta, as variáveis redundantes são excluídas mais eficientemente. Em geral, pode-se tomar β = 1 (Breiman et al.,1984). Caso este em que há um equilíbrio, em termos de peso, entre a redundância da variável candidata e a informação mútua desta com o desfecho. Assim, para todas as experiências desta dissertação, fixou-se β = 1.

Kwak & Choi (2002) salientam que o algoritmo MIFS-U pode ser aplicado a problemas complexos sem excessivo esforço computacional.

A seguir, apresenta-se um esquema do algoritmo MIFS-U, ilustrando a sua dinâmica.

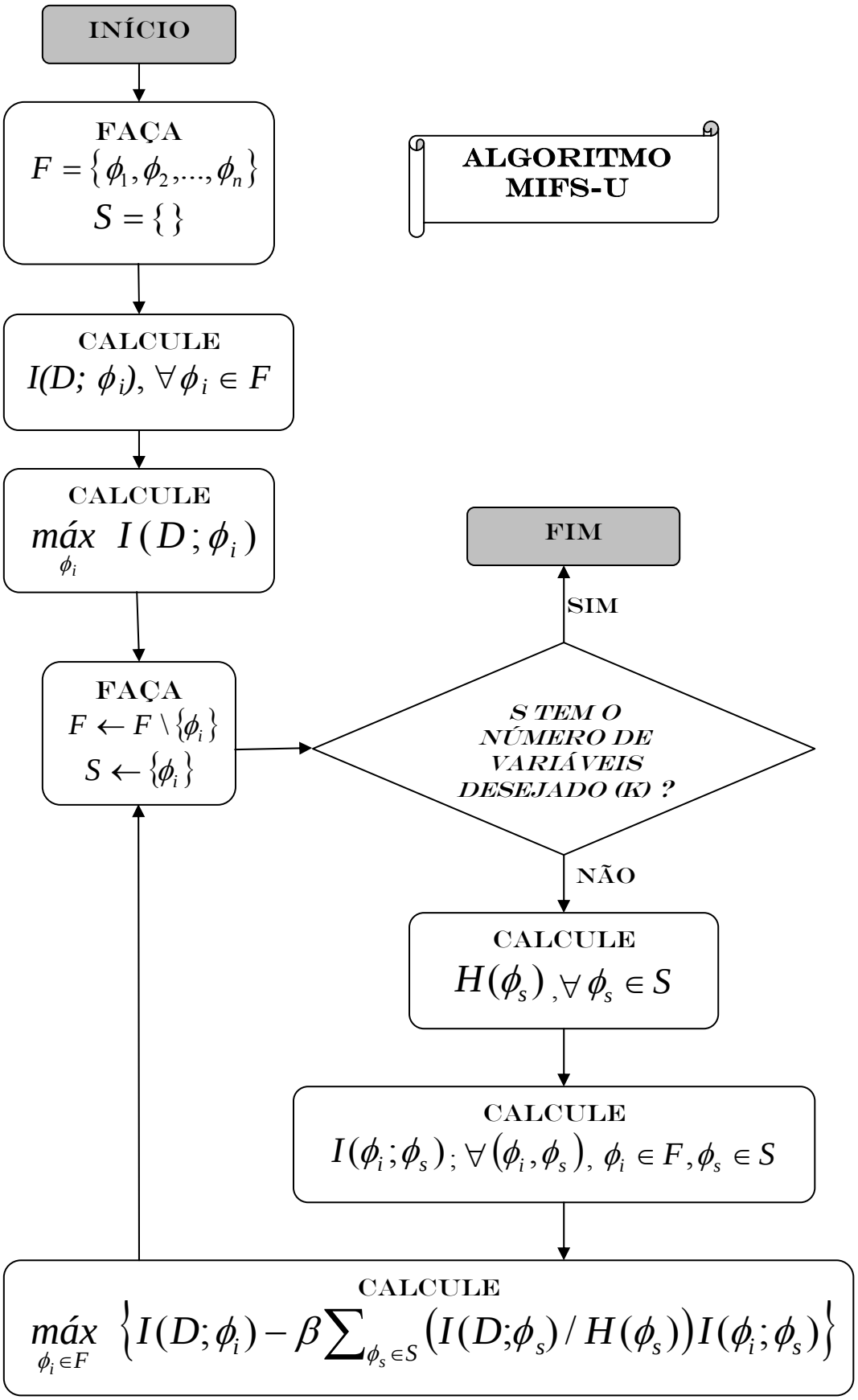

Figura 3.2 – Esquema do algoritmo MIFS-U