3
REDES NEURAIS E SVM

Nesse capítulo serão introduzidos conceitos de Redes Neurais e Máquina de Vetores Suporte (SVM) necessários para compreensão da metodologia desenvolvida.

3.1
Redes Neurais

As redes neurais são modelos matemáticos inspirados no cérebro humano, ou seja, são construídas a partir de arquiteturas de neurônios conectados. Cada neurônio é representado por uma equação, e cada ligação entre os neurônios é representada por pesos ou funções que geram sinais de propagação simulando a sinapse. A Figura 3.1 exemplifica uma arquitetura de uma rede neural de múltiplas camadas.

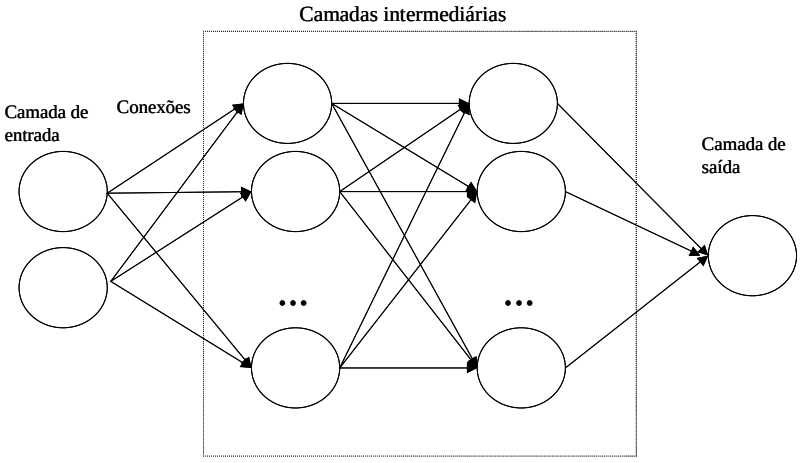

Figura 3.1: Exemplo de arquitetura de uma rede neural de múltiplas camadas

A camada de entrada representa os padrões, ou observações, que serão apresentados à rede. Essa camada é ligada à primeira camada intermediária por pesos que expressam o quanto um determinado padrão influenciará cada neurônio

da próxima camada. As camadas escondidas também estão ligadas por pesos sinápticos, o que pode ser interpretado também como suas contribuições para os próximos neurônios ligados. Esses pesos possuem o “conhecimento” da rede, pois são eles que são adaptados durante o treinamento. Nas camada escondidas, é efetuada a maior parte do processamento. A camada de saída fornece o resultado de todos os cálculos entre valores de entrada, pesos e funções de ativação.

Cada neurônio, tanto das camadas intermediárias como de saída, possui uma função de ativação. Dependo do sinal recebido pelos outros neurônios, esse neurônio fica mais ativo ou não. O resultado dessa ativação é utilizado para passar o sinal para outro neurônio ou para o resultado final da rede. Uma rede neural é especificada principalmente pela sua topologia, pelas características dos neurônios e pelas regras de treinamento.

A maioria das redes neurais é treinada por algum algoritmo representado por uma equação ou várias equações combinadas com o objetivo de produzir, a partir da saída da rede, um valor numérico que servirá para ajustar o valor das conexões de pesos de cada unidade de processamento. Esse algoritmo determina como os pesos de ligação da rede serão ajustados, e com isso alterará a ativação dos neurônios diante de um conjunto específico de estímulos de entrada, o que pode ser chamado de aprendizado.

Existem dois tipos de aprendizado para as redes neurais: o aprendizado supervisionado e o não-supervisionado. No primeiro, também chamada de aprendizado por correção de erro, os dados possuem exemplos que são comparados com a saída da rede, ou seja, esses algoritmos visam a minimizar a diferença entre a saída da rede e os exemplos. Já no aprendizado não-supervisionado, também chamada de aprendizado baseado em memória, não existem exemplos a serem seguidos. Esses algoritmos visam agrupar dados que possam possuir similaridades e simultaneamente maximizar a diferença entre as características de agrupamentos diferentes.

No presente estudo serão utilizadas as redes neurais *feedforwards* (supervisionadas) com aprendizado pela retropropagação do erro. Sua topologia

será definida através de testes de sensibilidade para definir quantas camadas escondidas e quantos neurônios em cada camada serão utilizados..

3.2
Máquina de Vetores Suporte (SVM)

3.2.1
Classificador SVM

O classificador SVM (*Support Vector Machine*) é um classificador binário que procura um hiperplano ótimo como uma função de decisão em um espaço de dimensões maiores [Boser *et al.*,1992, Vapnik, 1998, Cristianini e Shawe-Taylor, 2000]. Ele possui um conjunto de treinamento definido, em que x_k são os exemplos de treinamento e y_k, os rótulos de classe. O método consiste em primeiramente mapear x em um espaço de dimensão maior por meio de uma função Φ, e então calcular a função de decisão da forma:

$$f(x) = \langle w, \Phi(x) \rangle + b$$

Este método visa a maximizar a distância entre o conjunto de pontos $\Phi(x_k)$ e o hiperplano parametrizado por (w,b); para isso utiliza-se um conjunto de treinamento para estimar os parâmetros. O sinal de $f(x)$ define a classe de x, sendo que, para o classificador SVM com aprendizado pela otimização quadrática, essa função pode ser escrita como:

$$\min_{w,\xi} \frac{1}{2}\|w\|^2 + C\sum_{k=1}^{m} \xi_k^2$$

sob a condição $\forall k$, $y_k f(x_k) \geq 1 - \xi_k^2$.

A solução desse problema é obtida pela teoria de Lagrange e pode-se provar [Rakotomamonjy, 2003] que o vetor w é da forma:

$$w = \sum_{k=1}^{m} \alpha_k^* y_k \Phi(x_k)$$

onde α_k^* é a solução do seguinte problema de otimização quadrática

$$\max_{\alpha} W(\alpha) = \sum_{k=1}^{m} \alpha_k - \frac{1}{2} \sum_{k,l}^{m} \alpha_k \alpha_l y_k y_l \left(K(x_k, x_l) + \frac{1}{C} \delta_{k,l} \right)$$

sujeito a $\sum_{k=1}^{m} y_k \alpha_k = 0$ e $\forall k$, $\alpha_K \geq 0$, onde $\delta_{K,J}$ é o símbolo de Kronecker e $K(x_K, x_l)$ é o núcleo do produto interno (kernel).

Os núcleos de produto interno mais utilizados são o polinomial, funções de base radial e um perceptron de duas camadas.

As SVMs são providas de tantas estatísticas que a melhor forma de avaliar seus desempenhos é pelo erro *leave-one-out* (L). Esse procedimento consiste em estimar a função de decisão de m-1 exemplos e depois testar o resultado com o exemplo que ficou de fora, repetindo este procedimento m vezes até que todos os exemplos tenham sido utilizados como teste uma vez. O erro *leave-one-out* é conhecido como sendo um estimador não viciado do desempenho do classificador. Um dos limites de erros (*L*) mais comuns para SVM foi definido por Vapnik (1998):

$$L \leq 4R^2 \|w\|^2$$

onde R é o raio da menor esfera que contenha todos os dados mapeados $\Phi(x_k)$.

Um limite mais estreito, denominado estimativa de distância ou *Span estimate*, também pode ser utilizado e está baseado na distância S_p entre o vetor suporte $\Phi(x_p)$ e a distância de todos os demais vetores suportes (Vapnik e Chapelle, 2000):

$$L \leq \sum_{p} \alpha_p^* S_p^2$$

onde S_2^p está relacionado com a matriz estendida do produto entre os vetores suportes:

$$\tilde{K}_{SV} = \begin{pmatrix} K & 1 \\ 1^T & 0 \end{pmatrix}$$

pela equação $S_p^2 = 1 \Big/ \left(\tilde{K}_{SV}^{-1}\right)_{pp}$ '

3.2.2
Seleção de Variáveis Utilizando SVM

O objetivo da seleção de variáveis é eliminar variáveis irrelevantes para melhorar o desempenho de um modelo a ser estimado. Esta seleção também é muito utilizada para reduzir custos computacionais e otimizar o tempo de processamento. Por último, mas não menos importante, a seleção de variáveis possibilita conhecer melhor os dados e as relações de causa e efeito, facilitando o entendimento do problema a ser considerado.

Atualmente existem diferentes tipos de modelos de seleção de variáveis, utilizando diversas medidas de otimização, incluindo pesos maiores para erros mais graves, como, por exemplo, em casos de medicina, onde um erro pode ser fatal. Esta parte da tese está baseada nas metodologias de seleção de variáveis utilizando SVM, que têm se mostrado bem eficientes. A teoria foi conceituada com base em Rakotomamonjy (2003).

3.2.2.1
Algoritmo SVM-RFE

O Algoritmo SMV-RFE foi proposto por Guyon *et al.* (2000) para selecionar genes que são relevantes em diagnósticos de câncer – um problema de classificação. O objetivo é encontrar o subconjunto de *r* variáveis, entre as *d* variáveis existentes (*r*<*d*), que maximizem a desempenho do classificador. O

método se inicia com todas as variáveis e retira uma a uma até restarem *r* variáveis. A variável a ser removida é aquela que minimiza a variação de $\|w\|^2$ após sua retirada.

O critério R_C para ordenar a importância das variáveis pode ser descrito como:

$$\left| \|w\|^2 - \|w^{(i)}\|^2 \right| = \frac{1}{2} \left| \sum_{k,j} \alpha_k^* \alpha_j^* y_k y_j K(x_k, x_j) - \sum_{k,j} \alpha_k^{*(i)} \alpha_j^{*(i)} y_k y_j K^{(i)}(x_k, x_j) \right|$$

onde $K^{(i)}$ é a matriz que representa o núcleo do produto interno dos dados de treinamento quando a variável *i* é removida ($K_{k,j}^{(i)} = \left\langle \Phi(x_k^{(i)}), \Phi(x_j^{(i)}) \right\rangle$) e $\alpha_k^{*(i)}$ é a solução correspondente do problema de otimização quadrático do SVM.

Com o intuito de simplificar e reduzir a complexidade computacional deste algoritmo, supõe-se que $\alpha_k^{*(i)}$ é igual a α_k^*, mesmo que a variável tenha sido removida. Pode-se considerar que a variável removida é aquela que possui menor influência no peso da norma do vetor.

3.2.2.2
Algoritmo para Ordenamento de variáveis com SVM

Os algoritmos de seleção de variáveis necessitam de um critério para ordenar a importância das variáveis. Em vários trabalhos os limites de erro (*L*) têm sido utilizados para seleção de variáveis [Duan *el al*., 2002], e recentemente Weston *et al.* (2001) empregaram os limites dos raios das esferas para seleção de variáveis utilizando o algoritmo do gradiente decrescente. Essa idéia pode ser estendia para outros limites da generalização do erro. Em Rakotomamonjy (2003) são investigados três critérios: C_t , que são o peso do vetor $\|w\|^2$; os limites dos raios das esferas $R^2\|w\|^2$; e a estimativa da distância. Além disso, são testadas as duas formas abaixo de utilização para cada critério.

Método de ordem zero: nesse caso, o critério C_t é utilizado diretamente para ordenar a importância das variáveis; identifica-se a variável que produz o menor valor de C_t quando removida. O critério de ordenação se torna $R_C(i) = C_t^{(i)}$, sendo $C_t^{(i)}$ o critério de valor quando a variável *i* é removida. No caso $\|w\|^2$:

$$R_{C(i)} = \left\|w^{(i)}\right\|^2 = \sum_{k,j} \alpha_k^{*(i)} \alpha_j^{*(i)} y_k y_j K^{(i)}\left(x_k, x_j\right)$$

onde $K^{(i)}$ é a matriz do núcleo do produto interno dos dados de treinamento quando a variável *i* é removida

- ✓ Método de ordem zero: nesse caso, o critério C_t é utilizado diretamente para ordenar a importância das variáveis; identifica-se a variável que produz o menor valor de C_t quando removida. O critério de ordenação se torna $R_C(i) = C_t^{(i)}$, sendo $C_t^{(i)}$ o critério de valor quando a variável *i* é removida. No caso $\|w\|^2$:

  $$R_{C(i)} = \left\|w^{(i)}\right\|^2 = \sum_{k,j} \alpha_k^{*(i)} \alpha_j^{*(i)} y_k y_j K^{(i)}\left(x_k, x_j\right)$$

  onde $K^{(i)}$ é a matriz do núcleo do produto interno dos dados de treinamento quando a variável *i* é removida

- ✓ Método de primeira ordem: utiliza derivadas do critério C_t em relação à variável. Este método ordena a importância das variáveis por sua influência no valor do erro absoluto da derivada. Nesse caso o critério de

ordenação é $R_C(i) = |\nabla C_t|$. Considerando $\|w\|^2$, o critério passaria a ser $R_C(i) = \left|\nabla \|w\|^2\right|$, e o algoritmo seria:

1. inicialização: *Ordem* =[]; *Var*=[1,...,N]
2. repita
    a. Treine o Classificador SVM com todos os dados e variáveis de *Var*
    b. Para todas as variáveis em *Var*, faça: avalie o critério de ordenação $R_C(i)$ da variável *i*.
    c. Selecione a variável *j* que minimizar R_C : *Ordem*=[j]
    d. Remova a variável j de *Var*
3. Pare quando todas as variáveis estiverem ordenadas, ou seja, *Var*=[]

Similarmente ao algoritmo SVM-RFE, o problema de selecionar as *r* melhores variáveis é solucionado com base na seleção *backwards* (Kohavi e John, 1997). Essa solução é utilizada pela sua simplificação e otimização do processamento computacional quando comparado a outros métodos. Portanto o algoritmo começa com todas as variáveis e remove uma a uma até restarem as *r* variáveis mais importantes.

3.2.2.3
Cálculo do gradiente em função de um fator escalar

Para o critério de primeira ordem, o objetivo é medir a sensibilidade de um dado critério em relação à variável. Uma possibilidade é introduzir um fator escalar virtual para calcular o gradiente de um critério em relação ao fator escalar ν. Esse último funciona como um termo multiplicativo nas variáveis de entrada e, portanto, $k(x, x')$ torna-se:

$$k(\nu \cdot x, \nu \cdot x')$$

onde · representa o produto vetorial. Conseqüentemente, obtêm-se as derivadas do núcleo Gaussiano $k(\nu \cdot x, \nu \cdot x') = e^{-\frac{\|\nu \cdot x - \nu \cdot x'\|^2}{2\sigma^2}}$:

$$\frac{\partial k}{\partial \nu_i} = -\frac{1}{\sigma^2}(\nu \cdot x, \nu \cdot x')^2 k(x, x') = -\frac{1}{\sigma^2}\left(x_i - x_i'\right)^2 k(x, x')$$

onde se usa $\nu_i = 1$. Então, precisa-se calcular o gradiente dos limites relativos à variável ν_i e, dado um critério de ordenamento C_t, o termo fica:

$$R_C(i) = \left|\frac{\partial C_t(\alpha, b)}{\partial \nu_i}\right|$$

onde C_t é tanto $\|x\|^2$, $R^2 w^2$ ou $\sum_p \alpha_p^* S_p^2$ e depende da solução da otimização quadrática do SVM e do termo constante *b*.

Detalhes dos cálculos das derivadas podem ser encontrados em Rakotomamonjy (2002); são obtidos a partir de Bengio (2000) e Chapelle *et al.* (2002). Os resultados encontrados foram:

Gradiente do peso do vetor

$$R_C(i) = \left| \sum_{k,j} \alpha_k^* \alpha_j^* y_k y_j \frac{\partial k(\nu \cdot x_k, \nu \cdot x_j)}{\partial \nu_i} \right|$$

Gradiente do limite dos raios das esferas

$$R_C(i) = \left| \|w\|^2 \sum_{k,j} (\beta_k \beta_j - \beta_k \delta_{k,j}) \frac{\partial k(\nu \cdot x_k, \nu \cdot x_j)}{\partial \nu_i} + R^2 \sum_{k,j} \alpha_k^* \alpha_j^* y_k y_j \frac{\partial k(\nu \cdot x_k, \nu \cdot x_j)}{\partial \nu_i} \right|$$

onde R^2 é a função objetiva ótima do seguinte problema:

$$\max_\beta \quad \sum_k \beta_k k(\nu \cdot x_k, \nu \cdot x_k) - \sum_{k,j} \beta_k \beta_j k(\nu \cdot x_k, \nu \cdot x_j)$$

sujeito as restrições: $\sum_k \beta_k$ e $\beta_k \geq 0 \; \forall k$

Gradiente da estimativa de distância

$$R_C(i) = \left| \sum_{p=1}^{l} 2 \left(-H^{-1} \frac{\partial H}{\partial \nu_i} \alpha^* \right)_{pp} S_p^2 + \alpha_p^* S_p^4 \left(\tilde{K}_{SV}^{-1} \frac{\partial \tilde{K}_{SV}}{\partial \nu_i} \tilde{K}_{SV}^{-1} \right)_{pp} \right|$$

onde H é a seguinte matriz $H = \begin{pmatrix} K^Y & Y \\ Y^T & 0 \end{pmatrix}$ e $K_{kj}^Y = y_k y_j k(\nu \cdot x_k, \nu \cdot x_j)$