

# Projeto de redes RBF baseado na estrutura dos dados e em informações de margem

Luiz C. B. Torres\*, André P. Lemos\*, Cristiano L. Castro†, Antônio P. Braga\*

\*Programa de Pós-Graduação em Engenharia Elétrica, Universidade Federal de Minas Gerais,  
Av. Antônio Carlos 6627, 31270-901, Belo Horizonte, MG, Brasil

Email: {luiztorres, andrepl, apbraga}@cpdee.ufmg.br

†Universidade Federal de Lavras, MG, Brasil

Email: ccastro@dcc.ufla.br

**Abstract**—The RBF network is commonly used for classification and function approximation. The center and radius of the activation function of neurons is an important parameter to be found before the network training. This paper presents a method based on computational geometry to find these coefficients without any parameters provided by the user. Experimental results showed that our approach is promising.

## I. INTRODUÇÃO

Através de uma função não linear  $\varphi(\cdot)$ , a rede neural *Radial Basis Function* (RBF) realiza o mapeamento dos dados do espaço de entrada  $\mathbb{R}^n$  para um espaço de alta dimensionalidade  $\mathbb{R}^m$ , tal que  $m \gg n$ . De acordo com o teorema de Cover [1], neste novo espaço, o problema pode ser separado linearmente de forma mais simples. Este princípio é compartilhado pelas *Support Vector Machines* (SVM) [2] que utilizam *kernels* previamente ajustados para a indução de um mapeamento não linear, seguido da estimação de um hiperplano de margem máxima.

Em contraste com a rede *Multilayer Perceptron* (MLP), que pode ser projetada com mais de uma camada escondida, a rede RBF, geralmente, é formulada com apenas uma camada escondida, como mostra a Figura 1. O primeiro passo para a construção da topologia da rede RBF é encontrar o número de neurônios da camada escondida que corresponde ao número de funções base  $\varphi(\cdot)$ . Normalmente,  $\varphi(\cdot)$  é descrita como uma função gaussiana  $\varphi(\mathbf{x}) = \exp\left(-\frac{\|\mathbf{x}-c_i\|^2}{2\sigma_i^2}\right)$ , sendo, portanto, necessário definir os valores do centro  $c_i$  e do raio  $\sigma_i$  para cada ( $i$ -ésimo) neurônio oculto. Estudos propostos na literatura para determinação dos centros e, conseqüentemente, do número de neurônios ocultos, são baseados nos métodos  $K$ -médias e suas variações [3]: *Fuzzy C-means* (FCM), redes *Self-Organizing Maps* (SOM) [4] e *winner-takes-all* (WTA). É importante ressaltar que, no uso dos métodos  $k$ -médias e FCM, é necessário definir pelo menos um parâmetro a priori: o número  $k$  de centros. Os raios  $\sigma_i$  das funções  $\varphi(\cdot)$ , são normalmente definidos de forma empírica ou com base nas distâncias euclidianas entre os centros. Outra forma de se encontrar os parâmetros  $c_i$  e  $\sigma_i$  é através do uso de *cross-validation* [5].

Para superar as dificuldades na determinação dos parâmetros, este artigo apresenta uma nova estratégia direcionada a problemas de classificação. Os valores ótimos de  $c_i$  e  $\sigma_i$  são encontrados sem a necessidade de qualquer informação a priori. Usando ferramentas da Geometria Computacional [6], o

método proposto busca pelos padrões pertencentes à margem de separação das classes, de forma que estes exemplos são assinalados como centros das funções base  $\varphi(\cdot)$  de uma rede RBF. Além de não necessitar de parâmetros adicionais para a obtenção dos centros, nossa abordagem conta com as coordenadas geométricas de cada exemplo da margem para encontrar o respectivo  $\sigma_i$  de cada neurônio. Um estudo experimental preliminar com *benchmarks* conhecidos na literatura mostra que o método proposto é promissor. Os resultados obtidos são comparados a uma SVM.

O restante do artigo encontra-se organizado da seguinte forma. A Seção II apresenta a formulação da rede neural RBF. A Seção III descreve os conceitos teóricos que fundamentam nossa abordagem. Na Seção IV, nossa estratégia para projeto de redes RBF é apresentada. Na Seção V, são descritos a metodologia adotada na condução dos experimentos, bem como os resultados obtidos com a aplicação do novo método. Finalmente, a Seção VI traz as conclusões.

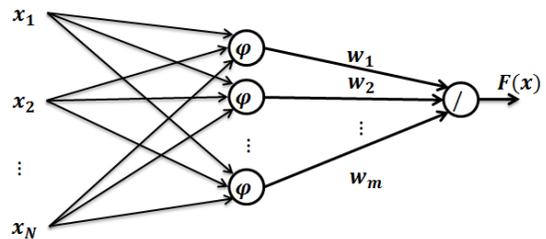


Figura 1. Topologia da Rede RBF.

## II. REDE NEURAL RBF

Uma rede neural RBF (*Radial Bases Function*) é considerada uma rede neural artificial que utiliza funções de bases radiais como funções de ativação dos neurônios da camada escondida. Diferentemente da rede MLP que pode ser construída com várias camadas intermediárias, a rede RBF geralmente é formulada com somente uma camada escondida. Formalmente, a rede RBF é construída utilizando somente três camadas como mostra a Figura 1. A primeira camada é responsável pela entrada dos padrões na rede. A segunda, também conhecida como camada escondida, realiza o mapeamento dos padrões de entrada para um espaço de alta dimensionalidade, utilizando funções de ativação de bases radiais não lineares. E a terceira, a camada de saída, é responsável pela resposta

da rede após os estímulos apresentados pelos padrões de entrada. De acordo com o teorema de Cover [1], o problema de classificação inicialmente formulado no espaço de entrada, que é posteriormente transformado em um espaço de alta dimensão, torna-se mais fácil de ser separado linearmente. Na formulação da rede RBF, o problema de classificação é transformado em um problema de aproximação de função em um espaço multi-dimensional. Neste novo espaço, também conhecido por espaço de características, o objetivo passa ser encontrar o modelo que gera a melhor aproximação dos dados de treinamento [7].

Para realizar o mapeamento dos dados para o espaço de características, a RBF faz uso de funções bases diferente das sigmóides utilizadas na rede MLP. Estas funções bases são utilizadas como funções de ativação dos neurônios da camada intermediária da rede (segunda camada), vide Figura 1. Dado o conjunto de dados  $V = \{\mathbf{x}_i, \mathbf{y}_i | i = 1 \dots N\}$  onde  $\mathbf{x}_i = \{\mathbf{x}_i \in \mathbb{R}^n | i = 1, 2 \dots N\}$  e  $\mathbf{y}_i = \{\mathbf{y}_i \in \mathbb{R}^1 | i = 1, 2 \dots N\}$ . O problema se traduz em minimizar a função

$$\arg \min_{\mathbf{w}} \sum_{i=1}^N \|\hat{F}(\mathbf{x}_i) - y_i\| \quad (1)$$

onde

$$\hat{F}(\mathbf{x}) = \sum_{i=1}^m w_i \varphi(\mathbf{x}, c_i) \quad (2)$$

$$\varphi(\mathbf{x}, c_i) = \exp\left(-\frac{\|\mathbf{x} - c_i\|^2}{2\sigma_i^2}\right) \quad (3)$$

$c_i$  é o centro da função de ativação do neurônio  $i$ , e  $\sigma_i$  é o raio de abrangência da função.

Para encontrar os parâmetros  $\sigma_i$  e  $c_i$ , utilizados na função de ativação gaussiana mostrada na Equação (3), são utilizadas heurísticas. Os métodos mais utilizados são: K-médias, FCM, rede SOM e WTA. Na maioria das vezes, esses métodos necessitam de mais parâmetros além do número de centros já definidos. Após encontrados estes parâmetros, o vetor de pesos  $\mathbf{w}$  que minimiza a Equação (1) é calculado através da Equação (4)

$$\mathbf{w} = \Psi^\dagger \mathbf{y} \quad (4)$$

onde

$$\Psi = \begin{pmatrix} \beta_{11} & \beta_{12} & \cdots & \beta_{1N} \\ \beta_{21} & \beta_{2,2} & \cdots & \beta_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ \beta_{m1} & \beta_{m2} & \cdots & \beta_{mN} \end{pmatrix} \quad (5)$$

$$\beta_{ij} = \mathbf{w}_i \varphi(\mathbf{x}_j, c_i)$$

e  $\Psi^\dagger$  é a pseudo-inversa linear *Moore-Penrose* [8] de  $\Psi$ .

### III. REFERENCIAL TEÓRICO

#### A. Diagrama de Voronoi

O Diagrama de *Voronoi* representa a decomposição de um espaço em regiões de acordo com a distância entre determinados pontos [9]. Dados dois pontos  $p, q \in S$ , onde  $S$  representa um conjunto de pontos em um plano, a bissetriz  $B(p, q)$  corresponde a uma reta perpendicular que atravessa o centro do segmento de reta  $\overline{pq}$ ;  $\delta(\cdot)$  é o operador que fornece a distância euclidiana entre dois pontos (vetores). O diagrama de *Voronoi* ( $Vor(S)$ ) pode ser considerado como a divisão do plano em  $m$  polígonos convexos  $P$  [10], [6]. A Figura 2(a) ilustra um exemplo. Um polígono  $P(x_i)$  é chamado de polígono de *Voronoi* relativo a  $x_i$  e é formado através da interseção do conjunto das bissetrizes  $B(x_i, P(x_i))$ . Um ponto  $p \in S$  pertence a  $P(x_i)$  se, e somente se, a seguinte desigualdade for satisfeita,

$$\delta(p, x_i) \leq \delta(p, x_j), x_i, x_j \in S, j \neq i \quad (6)$$

#### B. Triangulação de Delaunay

No diagrama de *Voronoi*  $Vor(S)$ , cada elemento  $v \in S$  está associado a um polígono de  $Vor(S)$ . O grafo dual de  $Vor(S)$  tem por vértices os elementos de  $S$  e por arestas os pares de elementos de  $S$ , cujos polígonos de  $Vor(S)$  são vizinhos [11], [12]. Tal diagrama resultante é chamado de Diagrama de *Delaunay* [10], [6], conforme ilustrado na Figura 2(b). A Triangulação de *Delaunay* (*TD*) é uma subdivisão de um objeto geométrico em triângulos. A *TD* pode ser modelada através de um grafo planar  $G_p$ , composto por um conjunto de vértices  $V$  e um conjunto de arestas  $E$  não ordenadas, como pode ser visto na Figura 2(c). Uma aresta  $(v_i, v_j) \in G_p$  é definida se, e somente se, existir um círculo contendo o par  $(v_i, v_j)$  e se todos os outros vértices de  $V$  forem exteriores a este círculo. Essa característica é ilustrada na Figura 2(d).

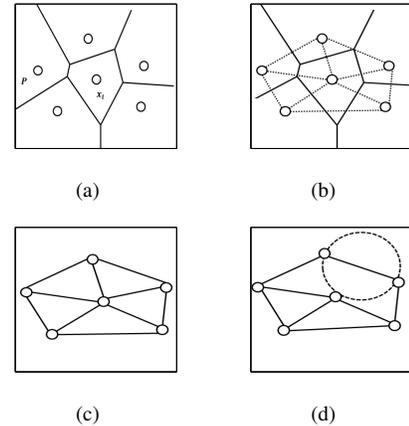


Figura 2. (a) Diagrama de *Voronoi*. (b) Grafo Dual do Diagrama de *Voronoi*. (c) Triangulação de *Delaunay* resultante. (d) Par de pontos de *Voronoi* que possuem uma aresta em comum representado em uma Triangulação de *Delaunay*.

#### C. Grafo de Gabriel

O grafo de Gabriel ( $G_G$ ) é um subconjunto de pontos do Diagrama de *Voronoi* e também um subgrafo da Triangulação

de *Delaunay* [13], ou seja,  $G_G \subseteq T_D \subseteq Vor$ . Segundo [6], o Grafo de Gabriel  $G_G$  de um conjunto de pontos  $S$ , é um grafo cujo conjunto de vértices  $V = S$  e seu conjunto de arestas  $E$  deve obedecer à seguinte definição,

$$(v_i, v_j) \in E \leftrightarrow \delta^2(v_i, v_j) \leq [\delta^2(v_i, z) + \delta^2(v_j, z)] \quad \forall z \in V, v_i, v_j \neq z \quad (7)$$

o que implica que, para  $(v_i, v_j)$  constituir uma aresta de  $G_G$ , não pode haver nenhum outro vértice dentro do círculo cujo o diâmetro é a distância euclidiana entre  $v_i$  e  $v_j$ .

O KNN (*K-Nearest Neighbors*) é um método de classificação cuja idéia principal é classificar um elemento no espaço  $n$ -dimensional de acordo com a classe dominante de seus  $K$  vizinhos mais próximos, obtidos no conjunto de treinamento. Um dos problemas encontrados no método é encontrar o melhor valor para o parâmetro  $K$ . O Grafo de Gabriel e o Diagrama de *Voronoi* também podem ser usados para esse propósito, mas diferentemente do KNN, não necessitam do parâmetro  $K$ . Como mencionado na Seção III-A, um ponto no Diagrama de *Voronoi* representa um polígono convexo, cujas fronteiras representam seus vizinhos mais próximos.

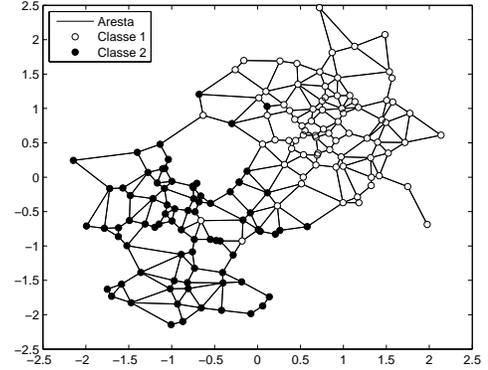
#### IV. METODOLOGIA PROPOSTA

A metodologia proposta nesse artigo busca pelas amostras encontradas na margem de separação entre as classes. Seu algoritmo pode ser formulado de acordo com os seguintes passos:

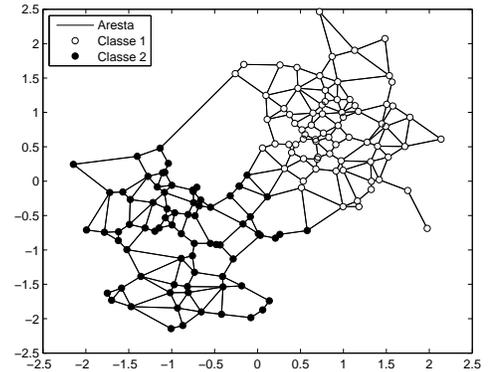
- 1) Obtém-se o Grafo de Gabriel ( $G_G$ ) com o conjunto de vértices sendo formado por todos os padrões de entrada, i.e.,  $V = \{\mathbf{x}_i \mid i = 1 \dots N\}$ , e o conjunto de arestas  $E$  satisfazendo a condição estabelecida na Equação (7). Veja exemplo na Figura 3(a).
- 2) Este passo do algoritmo é responsável por detectar e remover o ruído dos dados. Para todo  $\mathbf{x}_i \in V$ , analisa-se o subgrafo induzido pelo vértice  $\mathbf{x}_i$ , ou seja, o subgrafo formado pelas arestas que possuem  $\mathbf{x}_i$  como uma das extremidades. Se a maioria dos vizinhos (vértices adjacentes de  $\mathbf{x}_i$ ) possui rótulo diferente de  $\mathbf{y}_i$ , então  $\mathbf{x}_i$  é considerado como ruído e eliminado de  $V$ . Esse passo deve ser repetido até que não haja mais exclusão. A Figura 3(b) ilustra o conjunto de dados após a eliminação dos dados ruidosos.
- 3) A borda  $B_r$  das classes é encontrada da seguinte forma: seja  $\mathbf{x}_i \in V$  e,  $\forall \mathbf{x}_j \in V$  com  $j \neq i$ , caso a aresta  $(\mathbf{x}_i, \mathbf{x}_j)$  for formada por vértices de classes distintas (com rótulos diferentes), então ela é incluída no conjunto  $B_r$ , conforme ilustrado pela Figura 3(c).
- 4) Neste passo são calculados os pontos médios entre os exemplos que compõem as bordas das classes. Para cada aresta  $(\mathbf{x}_i, \mathbf{x}_j)$  pertencente ao conjunto  $B_r$ , calcula-se o ponto médio entre os vértices  $\mathbf{x}_i$  e  $\mathbf{x}_j$  de acordo com a seguinte expressão,

$$\bar{\mathbf{x}}_{ij} = \sum_{t=1}^n \mu(\mathbf{x}_i(t), \mathbf{x}_j(t)) \quad (8)$$

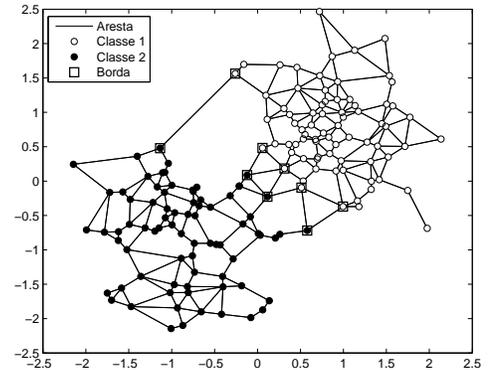
onde  $n$  é número de atributos (características) dos padrões de entrada e  $\mu(\cdot)$  é o operador que calcula



(a)



(b)



(c)

Figura 3. (a) Conjunto de padrões de entrada modelado com o Grafo de Gabriel. (b) Conjunto resultante após a eliminação dos dados ruidosos. (c) Detecção da borda das classes.

a média para o  $t$ -ésimo atributo. Após o cálculo dos pontos médios para todas as arestas de  $B_r$ , obtém-se um conjunto de ponto médios ( $P_M$ ) relativo às bordas das classes, conforme ilustrado pela Figura 4.

- 5) Dado o conjunto de dados  $V$  (conjunto resultante do passo 2), as Equações (9) e (10) a seguir fornecem os dados de entrada e saída para o treinamento da rede RBF,

$$X = [V \cup P_M], V \cap P_M = \emptyset \quad (9)$$

$$Y = \begin{cases} -1 & \text{se } \mathbf{x}_i \in V \text{ e } \mathbf{y}_i = -1 \\ 1 & \text{se } \mathbf{x}_i \in V \text{ e } \mathbf{y}_i = 1 \\ 0 & \text{se } \mathbf{x}_i \in P_M \end{cases} \quad (10)$$

onde  $X$  é o conjunto de todos os exemplos de  $V$ , acrescido do conjunto  $P_M$  encontrado no passo 4. Por sua vez,  $Y$  é o conjunto de rótulos, onde  $\mathbf{y}_i \in Y$ ,  $\mathbf{y}_i = 0$  se  $\mathbf{x}_i \in X$  for um ponto médio.

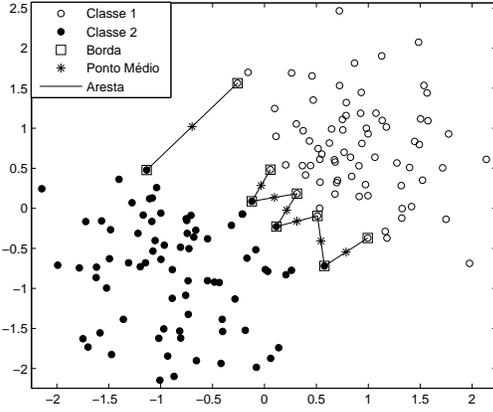


Figura 4. Pontos médios relativo às bordas das classes.

- 6) Cada elemento do conjunto  $B_r$  é utilizado como centro da função de ativação dos neurônios da rede RBF, ou seja, cada exemplo que compõe a borda é um centro  $c_i \in B_r$  para função de ativação  $\varphi_i$  de um neurônio da rede. O número de exemplos no conjunto  $B_r$  é igual ao número  $m$  de neurônios da camada escondida da rede RBF.
- 7) O raio  $\sigma_i$  da  $i$ -ésimo neurônio da rede é encontrado através da distância entre o centro  $c_i$  e o ponto médio  $j$  mais próximo, onde  $j \in P_M$ . A equação (11) mostra como é encontrado o  $\sigma_i$  de cada neurônio

$$\sigma_i = 2(\min \delta(c_i, P_M(j))), \quad \forall j \in P_M \quad (11)$$

onde  $\delta(\cdot)$  é o operador que calcula a distância euclidiana entre os dois pontos. A Figura 5 mostra os respectivos centros  $c_i$  e raios  $\sigma_i$  de cada neurônio, onde ( $d_i = \sigma_i$ ), e  $j$  é o ponto médio mais próximo de  $c_i$  para o problema mostrado na Figura 3(a). A Figura 6 mostra o resultado do método aplicado ao *toy problem* das duas luas.

## V. RESULTADOS

A maioria das bases de dados foi obtida através do repositório público *UCI benchmark repository* [14]. A base de

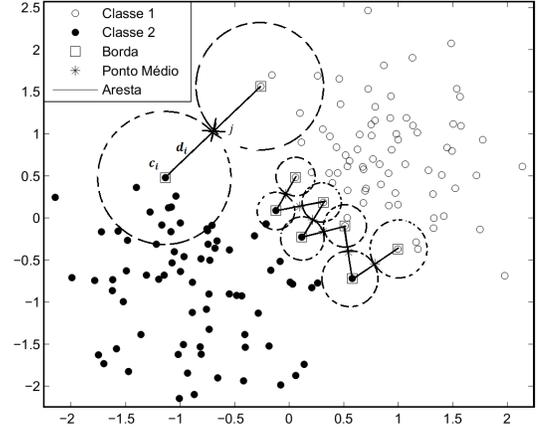


Figura 5. Centros e raios das funções de ativação dos neurônios da camada intermediária.

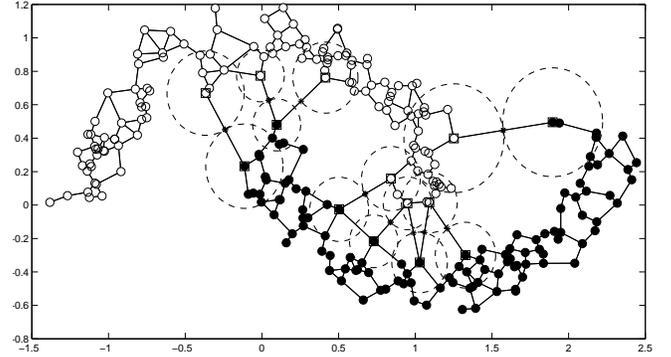


Figura 6. Metodologia proposta aplicada ao *toy problem* das duas luas.

dados *Appendicitis data set* foi retirada do sítio *Keel Data Set Repository*. Estas bases de dados são referenciadas em inúmeros trabalhos na literatura, o que as torna um ótimo *benchmark* para este trabalho. Todas as bases utilizadas no trabalho são binárias, são elas: *Stalog Australian Credit* (acr), *Bupa liver disorders* (bld), *The Statlog Heart Disease* (hea), *The Johns Hopkins University ionosphere* (ion), *The Pima Indians Diabetes* (pid), *The Wisconsin Breast cancer* (wbc) e *Appendicitis data set* (adc). Todas as bases foram pré-processadas. Os exemplos contendo atributos faltantes foram retirados e o restante dos dados foram normalizados com média  $\bar{x}_j = 0$  e desvio padrão igual a 1. A Tabela I mostra as características das bases de dados, onde o número de atributos categóricos e numéricos são denotados por  $n_{num}$  e  $n_{cat}$  respectivamente, e  $N$  é o número total de exemplos. Nossa metodologia foi confrontada com o classificador *Support Vector Machine* (SVM). A implementação utilizada do método SVM foi a LIBSVM [15].

Fazendo uso da LIBSVM, o objetivo é minimizar a Equação (12) abaixo

$$\begin{aligned} \min_{\mathbf{w}, b, \xi} \quad & \frac{1}{2} \mathbf{w}^T \mathbf{w} + \lambda \sum_{i=1}^N \xi_i \\ \text{S.t.} \quad & g(x) \leq 0 \\ & y_i (\mathbf{w}^T \vartheta(\mathbf{x}_i) + b) \geq 1 - \xi_i, \quad \xi_i \geq 0 \end{aligned} \quad (12)$$

onde

$$\vartheta(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\gamma \|\mathbf{x}_i - \mathbf{x}_j\|), \quad \gamma > 0 \quad (13)$$

Para encontrar os coeficientes  $\gamma$  e  $\lambda$ , foi utilizado o método *10-fold cross-validation* e um *grid* de busca. O objetivo desta metodologia é encontrar o par  $(\gamma, \lambda)$  que resulta na maior acurácia do método *cross-validation*. No experimento realizado, os parâmetros foram variados da seguinte forma

$$\begin{aligned} \gamma &= \{2^{-n}, 2^{-(n-2)}, \dots, 2^{15}\}, \quad n = 5 \\ \lambda &= \{2^{-n}, 2^{-(n-2)}, \dots, 2^2\}, \quad n = 15 \end{aligned}$$

Em contrapartida, para a rede RBF, os centros  $c_i$  e o raio  $\sigma_i$  da função de ativação do  $i$ -ésimo neurônio  $\varphi_i$  foram encontrados através da metodologia proposta neste trabalho.

A Tabela II apresenta os resultados da nossa proposta (GC-RBF) e da SVM. Para cada base de dados foram realizadas 10 repetições, sempre aleatorizando os dados. São mostrados a média da acurácia e o desvio padrão respectivamente.

Tabela I. CARACTERÍSTICA DAS BASES DE DADOS

Base	$n_{num}$	$n_{cat}$	$n$	N
acr	6	8	14	690
bld	6	0	6	345
hea	7	6	13	270
ion	33	0	33	351
pid	8	0	8	768
snr	60	0	60	208
wbc	9	0	9	683
adc	7	0	7	106

Tabela II. RESULTADOS: MÉDIA DA ACURÁCIA E DESVIO PADRÃO.

Base	CG-RBF	SVM
acr	0.8586 (0.0214)	0.8551 (0.0094)
bld	0.6672 (0.0334)	0.6920 (0.0166)
hea	0.8280 (0.0259)	0.8427 (0.0295)
ion	0.9270 (0.0206)	0.9270 (0.0236)
pid	0.7494 (0.0196)	0.7545 (0.0161)
snr	0.7588 (0.0629)	0.8485 (0.0326)
wbc	0.9627 (0.0084)	0.9680 (0.0104)
adc	0.6058 (0.0566)	0.6044 (0.0477)

Para comparar os resultados, foi utilizado o teste estatístico *Wilcoxon signed rank test* [16]. Este teste é bastante utilizado na literatura para comparar dois classificadores de acordo com o trabalho de [17] com mais de 1530 citações segundo o *Google Scholar*. O teste de *Wilcoxon* é considerado um método não-paramétrico para comparação de duas amostras não pareadas.

As hipóteses para o teste estatístico foram formuladas como se segue:

- $H_0$ : Os dois classificadores são iguais. (Hipótese nula)
- $H_1$ : Os classificadores são diferentes. (Hipótese alternativa)

Aplicando o teste *Wilcoxon signed rank test*, obteve-se o *p-value* igual a 0.078125. Como o *p-value* é maior que o nível de significância de 0.05. Conclui-se com um nível de significância de 5% que a hipótese nula  $H_0$  não deve ser rejeitada, ou seja, para classificar as bases de dados descritas na Tabela I, o classificador CG-RBF mostrou-se equivalente à SVM.

## VI. CONCLUSÃO

Este trabalho apresentou resultados preliminares de uma nova abordagem para construir redes neurais RBF. Classificadores baseados em SVMs (*Support Vectors Machines*) e redes de regularização são normalmente projetados utilizando métodos exaustivos e *cross-validation* para obtenção de seus parâmetros “ótimos”, exigindo muito tempo e experiência do usuário. O método proposto neste artigo, por sua vez, não é sensível a parâmetros, tal que todos os parâmetros necessários para o projeto do classificador são obtidos geometricamente e diretamente da base de dados, sem a necessidade de informação a priori. No caso particular deste trabalho, partimos do princípio de que os parâmetros “ótimos” encontram-se na margem de separação das classes e vetores de entrada gerados a partir deles trazem informação valiosa para o problema. Resultados com *benchmarks* multivariados mostram que nossa abordagem foi equivalente ao classificador SVM dentro do domínio experimental deste trabalho, sem a utilização de nenhum método para selecionar parâmetros ou de ajustes conduzidos pelo usuário. Futuramente, pretende-se aplicar a metodologia em mais bases de dados e estendê-la para problemas de regressão.

## REFERÊNCIAS

- [1] T. M. Cover, “Geometrical and statistical properties of systems of linear inequalities with applications in pattern recognition,” *Electronic Computers, IEEE Transactions on*, no. 3, pp. 326–334, 1965.
- [2] C. Cortes and V. Vapnik, “Support-vector networks,” *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.
- [3] J. Sing, D. Basu, M. Nasipuri, and M. Kundu, “Improved k-means algorithm in the design of rbf neural networks,” in *TENCON 2003. Conference on Convergent Technologies for Asia-Pacific Region*, vol. 2. IEEE, 2003, pp. 841–845.
- [4] S. Bouchired, M. Ibnkahla, D. Roviras, and F. Castanie, “Equalization of satellite mobile communication channels using combined self-organizing maps and rbf networks,” in *Acoustics, Speech and Signal Processing, 1998. Proceedings of the 1998 IEEE International Conference on*, vol. 6. IEEE, 1998, pp. 3377–3379.
- [5] R. Kohavi *et al.*, “A study of cross-validation and bootstrap for accuracy estimation and model selection,” in *International joint Conference on artificial intelligence*, vol. 14. Lawrence Erlbaum Associates Ltd, 1995, pp. 1137–1145.
- [6] M. de Berg, M. van Kreveld, M. Overmars, and O. Schwarzkopf, *Computational Geometry: Algorithms and Applications*, 2nd ed. Springer-Verlag, 2000.
- [7] S. Haykin and N. Network, *A comprehensive foundation*, 2004, vol. 2.
- [8] A. Albert, *Regression and the Moore-Penrose pseudoinverse*. Academic Press New York, 1972, vol. 3.
- [9] F. Aurenhammer and R. Klein, “Voronoi diagrams,” in *Handbook of Computational Geometry*. Elsevier, 1990, pp. 152–159.

- [10] L. H. Figueiredo, *Introdução a Geometria Computacional*. Rio de Janeiro: Impa, 1991.
- [11] J. Li and C.-C. Kuo, "A dual graph approach to 3d triangular mesh compression," in *Image Processing, 1998. ICIP 98. Proceedings. 1998 International Conference on*, vol. 2, oct 1998, pp. 891–894 vol.2.
- [12] H. Zhang and X. He, "On simultaneous straight-line grid embedding of a planar graph and its dual," *Information Processing Letters*, vol. 99, no. 1, pp. 1 – 6, 2006.
- [13] W. Zhang and I. King, "A study of the relationship between support vector machine and gabriel graph," in *Neural Networks, 2002. IJCNN '02. Proceedings of the 2002 International Joint Conference on*, vol. 1, 2002, pp. 239–244.
- [14] K. Bache and M. Lichman, "UCI machine learning repository," 2013. [Online]. Available: <http://archive.ics.uci.edu/ml>
- [15] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, pp. 27:1–27:27, 2011, software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm>.
- [16] F. Wilcoxon, "Individual comparisons by ranking methods," *Biometrics bulletin*, vol. 1, no. 6, pp. 80–83, 1945.
- [17] J. Demšar, "Statistical comparisons of classifiers over multiple data sets," *The Journal of Machine Learning Research*, vol. 7, pp. 1–30, 2006.