DISSERTAÇÃO DE MESTRADO Nº 1105

REGULARIZAÇÃO DE CLASSIFICADORES GEOMÉTRICOS DE MARGEM LARGA BASEADOS NO GRAFO DE GABRIEL

Matheus Nogueira Salgado

DATA DA DEFESA: 14/02/2019

Universidade Federal de Minas Gerais

Escola de Engenharia

Programa de Pós-Graduação em Engenharia Elétrica

REGULARIZAÇÃO DE CLASSIFICADORES GEOMÉTRICOS DE MARGEM LARGA BASEADOS NO GRAFO DE GABRIEL

Matheus Nogueira Salgado

Dissertação de Mestrado submetida à Banca Examinadora designada pelo Colegiado do Programa de Pós-Graduação em Engenharia Elétrica da Escola de Engenharia da Universidade Federal de Minas Gerais, como requisito para obtenção do Título de Mestre em Engenharia Elétrica.

Orientador: Prof. Antônio de Pádua Braga

Belo Horizonte - MG

Fevereiro de 2019

Regularização De Classificadores Geométricos De Margem Larga Baseados No Grafo De Gabriel

Matheus Nogueira Salgado

Programa de Pós-Graduação em Engenharia Elétrica

Universidade Federal de Minas Gerais

Orientador: Prof. Antônio de Pádua Braga

Co-Orientador: Prof. Luiz Carlos Bambirra Torres

Dissertação de

Mestrado em Engenharia Elétrica

02/2019

Agradecimentos

Agradeço a Deus pela adoção (Gl 4:5).

Aos meus pais, José e Luciana, pelo cuidado ininterrupto e incentivo aos estudos. Aos meus irmãos, Isabella e Nathan, pela amizade e carinho. A minha parceira Clara, meu amor, tão presente que poderia construir seu próprio grafo de Gabriel.

Ao meu professor e orientador Braga e co-orientador Luiz pela paciência, incentivo e ensinamentos desde o final da graduação.

Lourenço, Murilo e demais colegas do LITC pela ajuda no dia a dia.

A CAPES do Brasil pelo apoio financeiro.

With four parameters I can fit an elephant, and with five I can make him wiggle his trunk.

John Von Neumann

Resumo

O presente trabalho destina-se ao estudo de novas formas de regularização baseada em informações extraídas do grafo de Gabriel. São duas principais contribuições: primeiro, um estudo preliminar avalia como o grafo de Gabriel pode ser utilizado na regularização de redes neurais RBF e como essa estrutura pode ser informativa. Características extraídas do grafo foram utilizadas para remoção de funções radiais estimadas através do método CG-RBF, que também utiliza o grafo em sua construção. A segunda contribuição é a proposta de uma nova abordagem de filtragem de ruído para um classificador construído com informações extraídas do grafo de Gabriel, o CHIP-CLASS. Esse classificador não utiliza algoritmos de otimização ou definição de parâmetros pelo usuário. Trabalhos anteriores mostraram que classificadores eficientes podem ser construídos assim. No entanto, ainda há muito o que avançar no controle da capacidade desses classificadores. Os resultados mostram que um conjunto especial de vértices do grafo de Gabriel é bastante informativo da região de separação entre classes e que a filtragem de amostras baseada em características do grafo pode ser utilizada para controlar a capacidade do modelo proposto.

Abstract

The present work is aimed at the study of new ways to build regularization based only on information extracted from the Gabriel graph. There are two main contributions: first, a preliminary study evaluates how the Gabriel graph can be used in the regularization of RBF neural networks and how this structure can be informative. Characteristics extracted from the graph were used to remove radial functions estimated by the CG-RBF algorithm, which also uses the graph in its construction. Second, a novel filtering approach is proposed for a classifier designed with information extracted from the Gabriel graph, the CHIP-CLASS. This classifier does not use neither optimization algorithms nor parameter definition by the user. Previous work has shown that efficient classifiers can be designed as such. However, there is still much to progress in regularization of these classifiers. The results show that a special set of Gabriel graph vertices is very informative of the classes separation region and that the filtering of samples based on characteristics of the graph can be used to control the capacity of the proposed model.

Sumário

Lista de Símbolos

D	Conjunto de dados
$\mathcal{L}(\cdot\|\cdot)$	Função de verossimilhança
$\mathcal{G}$	Grafo
$\mathcal{V}(\mathcal{G})$	Conjunto de vértices do grafo $\mathcal{G}$
$\mathcal{E}(\mathcal{G})$	Conjunto de arestas do grafo $\mathcal{G}$
$\psi_{\mathcal{G}}$	Função de incidência
v_i	Vértice i
e_i	Aresta i
M	Matriz de adjacência
G_G	Grafo de Gabriel
$\hat{y}$	Estimador de y
$\mathbf{x}$	Vetor
$\mathbf{y}$	Vetor
$\mathbf{w}$	Vetor de pesos
$\mathbf{m}$	Vetor de médias
$\mathbf{Z}$	Matriz
$\mathbf{\Sigma}$	Matriz de parâmetros
$\mathbf{H}$	Matriz de projeção dos padrões de entrada
$\mathbf{I}_p$	Matriz identidade de tamanho d
$sinal(\cdot)$	Função Sinal
h_i	Função radial associada a vértice v_i
λ	Parâmetro da regularização de tikhonov
$\mathcal{A}(v_i)$	Grau do vértice v_i
$\phi(\cdot,\cdot)$	Distância geodésica

Lista de Abreviaturas

AS	Conjunto de Arestas Suporte
D_C	Distância entre Classes
GG	Grafo de Gabriel
LOOCV	*Leave one out cross validation*
MAP	Estimador de máxima a posteriori verossimilhança
MLE	Estimador de máxima verossimilhança
MSE	Erro Quadrático Médio
PQ	Programação Quadrática
RBF	Radial Basis Function
SVM	Support Vector Machine
VS	Vetores de Suporte
VSE	Conjunto de Vetores Suporte Estruturais

Lista de Figuras

Lista de Tabelas

Capítulo 1

Introdução

Depois da formulação das máquinas de vetores suporte, as SVMs (Boser *et al.*, 1992), os classificadores de margem larga têm ganhado muita popularidade na literatura relacionada ao aprendizado de máquina. Porém, pouco foi investigado sob a perspectiva geométrica do problema (Smola, 2000), que propõe extrair informações apenas das relações espaciais dos dados. Uma maneira de fazer isso é utilizar a informação contida no grafo de Gabriel (GG) (Gabriel & Sokal, 1969), estrutura construída diretamente dessas relações. Trabalhos recentes têm mostrado que essa abordagem é promissora para classificadores de margem larga e também para redes neurais.

Um classificador de margem larga é definido pelas amostras que estão junto à margem de separação. Para isso é preciso identificar corretamente a região de separação entre classes. Estado da arte dos classificadores de margem larga, as SVM definem as amostras da região de separação como vetores de suporte (VS). Para identificar essas amostras é necessário resolver um problema de programação quadrática (PQ) que minimiza o erro e maximiza a margem de separação (Boser *et al.*, 1992). Diferentemente da SVM, que resolve o problema de identificação dessas amostras com otimização, a perspectiva geométrica é fundamentada no princípio de que a região de separação é intrínseca à disposição das amostras de entrada.

O CHIP-CLAS é um classificador que utiliza o grafo de Gabriel e trabalhos recentes têm mostrado a sua eficiência [(Torres *et al.*, 2015a),(Torres *et al.*,

2014),(Torres *et al.*, 2015b)], com resultados comparáveis à SVM. O classificador CG-RBF (Torres *et al.*, 2013) parte do mesmo princípio e utiliza o GG para estimar os parâmetros das funções radiais de uma rede neural *Radial Basis Function* (RBF). Essa estimativa não requer algoritmos de otimização ou escolha de parâmetros, uma vez que toda informação é obtida geometricamente. O objetivo dessa dissertação é avançar no projeto destes dois classificadores investigando e propondo novas formas de regularização que mantenham as suas características originais.

A regularização está relacionada a um dos dois fundamentais erros estatísticos em modelos científicos: *overfitting* e *underfitting*, como afirma McElreath em *Statistical Rethinking*, (McElreath, 2016, p. 166). Modelos de baixa capacidade estão sujeitos a *underfitting*, erro que ocorre quando o modelo não consegue aprender com os dados de entrada. O *overfitting*, por outro lado, ocorre quando o modelo tem alta capacidade e fica sujeito a aprender *demais* os dados e acaba modelando ruído. É difícil garantir que o modelo escolhido tenha a capacidade exata para se ajustar aos dados na medida correta. A regularização é uma ferramenta que insere novas informações ao modelo mais complexo para controlar sua capacidade. Dado um modelo, a regularização tem como objetivo evitar o *overfitting*. Outra forma de definir o papel da regularização é através do dilema viés *versus* variância: a regularização insere um viés para diminuir a variância de um modelo com o objetivo de encontrar o ponto de equilíbrio ótimo, que é o menor erro de teste.

As SVMs lidam com este problema utilizando variáveis de folga (Boser *et al.*, 1992), que são parâmetros de tolerância a ruído. Essa tolerância pode ser comparada à eliminação da sobreposição na região de separação, uma vez que essas amostras deixam de influenciar o valor da função objetivo. Esse parâmetro de regularização também é escolhido através de otimização. Em contraste, o método de regularização do CG-RBF e do CHIP-CLAS utiliza uma estratégia de filtragem dos dados para controlar a capacidade do modelo sem utilização de parâmetros.

Para o CG-RBF é feito um estudo da regularização através da retirada de funções radiais utilizando o grafo de Gabriel para identificação de margem. É feita uma comparação com a regularização de Tikhonov. O CHIP-CLAS, por outro lado, possui uma estratégia de regularização através de uma filtragem baseada

no grau do vértice (Torres, 2016). Apesar de seguir as diretrizes da perspectiva geométrica, a estratégia tem suas fraquezas. É definido um limiar arbitrário para retirada de ruídos e, devido a este limiar, amostras serão consideradas como ruído mesmo quando não houver.

Assim, este trabalho apresenta uma abordagem para filtragem das amostras a partir de propriedades intrínsecas ao grafo de Gabriel. O GG é um subgrafo da Triangulação de Delaunay (Zhang & King, 2002) cuja construção resulta em um grafo sempre conexo. Isso significa que há um caminho entre qualquer par de vértices e que a distância entre eles, também chamada de distância geodésica (Buckley & Harary, 1990), pode ser calculada. Sendo assim, a centralidade de qualquer vértice pode ser calculada em relação a qualquer grupo de vértices (Everett & Borgatti, 2005). A partir dessas propriedades, a distância média entre diferentes classes também pode ser calculada: o quanto uma amostra está distante das amostras de outras classes. As amostras da região de sobreposição estão mais próximas de outras classes do que as demais. Assim, é definida uma métrica de grafo não paramétrica que representa, indiretamente, a proximidade das amostras com a região de sobreposição. Depois da definição dessa métrica, o trabalho investiga suas propriedades e testa sua utilização na filtragem de ruído em bases de dados reais.

1.1 Publicações

- Salgado, M.N., Torres, L.C. & Braga, A.P. (2017). Modelo geométrico de margem larga baseado em propriedades estruturais de Grafos de Gabriel e em distância geodésica. Congresso Brasileiro de Inteligência Computacional.

- Salgado, M.N., Torres, L.C., Coelho, F. & Braga, A.P. (2018). Informação estrutural dada por grafos de gabriel aplicada a regularização de redes neurais RBF. Congresso Brasileiro de Automática.

1.2 Estrutura da Monografia

O trabalho está dividido em seis capítulos. Este capítulo apresenta uma introdução. O Capítulo 2 apresenta o referencial teórico que abrange todos os conceitos necessários para um melhor entendimento da dissertação. O Capítulo 3 propõe uma metodologia de regularização para o CG-RBF. O Capítulo 4 propõe uma metodologia de regularização para o CHIP-CLAS. O Capítulo 5 apresenta os resultados dos experimentos e testes realizados. Por fim, no Capítulo 6 tem-se a conclusão da dissertação e propostas de continuidade.

Capítulo 2

Referencial Teórico

2.1 Regularização: perspectiva Bayesiana

A regularização desempenha um papel fundamental no desenvolvimento de algoritmos de aprendizado supervisionado. A capacidade de generalização de um modelo é controlada ao inserir informação externa aos dados para diminuir a variância. A regularização aqui será considerada a partir de um prisma bayesiano em que a informação *a priori* é utilizada juntamente com os dados de entrada para o aprendizado de um modelo.

Considere como modelo, para simplificar as derivações que seguem, uma regressão linear para estimar $\hat{y}_i = \beta_0 + \beta_1 x_i$ dado um conjunto de dados $\{x_i, y_i\}$, onde $\{i = 1, 2, ..., n\}$. A partir de uma abordagem estatística frequentista, sabe-se que a melhor estimativa $\hat{\beta} = \{\hat{\beta}_0, \hat{\beta}_1\}$ para o parâmetro $\beta = \{\beta_0, \beta_1\}$ maximiza a função de verossimilhança (MLE), Eq. (2.1)

$$\begin{aligned} \mathcal{L}(\beta|\mathbf{y}) &= P(\mathbf{y}|\beta) \\ &= \prod_{i=1}^{n} P_y(y_i|\beta, \sigma^2) \\ &= \prod_{i=1}^{n} \frac{1}{\sigma\sqrt{2\pi}} \exp \frac{-(y_i - (\beta_0 + \beta_1 x_i))^2}{2\sigma^2} \end{aligned} \tag{2.1}$$

onde $\mathbf{y}$ é o vetor $\{y_i\}$ e σ é a variância inerente aos dados. A única suposição feita é que $\{x_i, y_i\}$ assume uma distribuição normal na reta de regressão.

A Eq. (2.2) mostra que maximizar a função de verossimilhança é equivalente à minimização do erro quadrático médio (MSE) para regressão linear.

$$\begin{aligned}
\hat{\beta}_{MLE} &= \arg\max_{\beta} \quad \mathcal{L}(\beta|\mathbf{y}) \\
&= \arg\max_{\beta} \quad \prod_{i=1}^{n} \frac{1}{\sigma\sqrt{2\pi}} \exp \frac{-(y_i - (\beta_0 + \beta_1 x_i))^2}{2\sigma^2} \\
&= \arg\max_{\beta} \quad \log \prod_{i=1}^{n} \frac{1}{\sigma\sqrt{2\pi}} \exp \frac{-(y_i - (\beta_0 + \beta_1 x_i))^2}{2\sigma^2} \\
&= \arg\max_{\beta} \quad -\sum_{i=1}^{n} \frac{(y_i - (\beta_0 + \beta_1 x_i))^2}{2\sigma^2} \\
&= \arg\min_{\beta} \quad \sum_{i=1}^{n} (y_i - (\beta_0 + \beta_1 x_i))^2
\end{aligned} \tag{2.2}$$

Abordando o mesmo problema a partir da estatística bayesiana, mostrada na Eq. (2.3)

$$P(\beta|y) = \frac{P(y|\beta)P(\beta)}{P(y)}, \tag{2.3}$$

podemos utilizar alguma informação *a priori* que não vem do conjunto de dados e maximizar a função de verossimilhança *a posteriori* (MAP). Essa informação *a priori* pode ser fraca ou forte, de acordo com o conhecimento prévio da distribuição de probabilidade dos parâmetros do modelo.

A Eq. (2.4) representa o estimador MAP, onde $P(y|\beta)$ é a função de verossimilhança, $P(\beta)$ é a informação *a priori* da distribuição de probabilidade do parâmetro β e $P(y)$, distribuição de y que não depende de β. Percebe-se que o $\hat{\beta}_{MAP}$ difere de $\hat{\beta}_{MLE}$ na adição do termo $\log P(\beta)$ na função objetivo. A regularização depende da escolha da *prior* $P(\beta)$.

$$\begin{aligned}
\hat{\beta}_{MAP} &= \arg\max_{\beta} \quad P(\beta|\mathbf{y}) \\
&= \arg\max_{\beta} \quad \frac{P(y|\beta)P(\beta)}{P(y)} \\
&= \arg\max_{\beta} \quad P(y|\beta)P(\beta) \\
&= \arg\max_{\beta} \quad \log P(y|\beta) + \log P(\beta).
\end{aligned} \tag{2.4}$$

Considere $P(\beta) \sim \mathcal{N}(0, \tau^2)$, ou seja, a informação adicional é que os parâmetros β seguem uma distribuição normal centrada em zero e variância τ^2. Análogo ao estimador β_{MLE} (Eq. 2.2), a Eq. (2.4) é reescrita como

$$\begin{aligned}
\hat{\beta}_{L2} &= \arg\max_{\beta} \quad \log P(y|\beta) + \log P(\beta) \\
&= \arg\max_{\beta} \quad \log \prod_{i=1}^{n} \frac{1}{\sigma\sqrt{2\pi}} \exp \frac{-(y_i - (\beta_0 + \beta_1 x_i))^2}{2\sigma^2} + \log \prod_{j=0}^{p} \frac{1}{\tau\sqrt{2\pi}} \exp \frac{-\beta_j^2}{2\tau^2} \\
&= \arg\max_{\beta} \quad -\sum_{i=1}^{n} \frac{(y_i - (\beta_0 + \beta_1 x_i))^2}{2\sigma^2} - \sum_{j=0}^{p} \frac{\beta_j^2}{2\tau^2} \\
&= \arg\min_{\beta} \quad \sum_{i=1}^{n} (y_i - (\beta_0 + \beta_1 x_i))^2 + \frac{\sigma^2}{\tau^2} \sum_{j=0}^{p} \beta_j^2 \\
&= \arg\min_{\beta} \quad \sum_{i=1}^{n} (y_i - (\beta_0 + \beta_1 x_i))^2 + \lambda \sum_{j=0}^{p} \beta_j^2.
\end{aligned} \tag{2.5}$$

Estimar o parâmetro de um modelo através da máxima verossimilhança *a posteriori* utilizando uma *prior* gaussiana é equivalente a regularização de Tikhonov (Ng, 2004), ou *ridge regression*, para o caso específico da regressão linear. Também é conhecida como regularização de norma L2. Essa regularização tende a escolher valores de β mais próximos a zero do que uma alternativa sem regularização (James *et al.*, 2013, p. 203).

Outra regularização muito utilizada é a L1, ou LASSO (Ng, 2004) para o caso específico da regressão linear, que escolhe a *prior* como uma distribuição de Laplace,

$$\frac{1}{2b}\exp\frac{-\|x-\mu\|}{b} \tag{2.6}$$

onde μ é um parâmetro de localidade e b, de dispersão.

A Eq. (2.7) omite o desenvolvimento, análogo à Eq. (2.5), e apresenta o resultado do estimador $\hat{\beta}_{L1}$. Diferente da norma L2, a norma L1 tende a escolher valores de $\beta_j = 0$. Isso fica claro ao comparar as funções de densidade de probabilidade das distribuições gaussiana e Laplace em torno do valor zero. Por isso, o LASSO funciona como um seletor de variáveis na regressão linear (James *et al.*, 2013, p. 203).

$$\hat{\beta}_{L1} = \arg\min_{\beta} \quad \sum_{i=1}^{n}(y_i - (\beta_0 + \beta_1 x_i))^2 + \lambda\sum_{j=0}^{p}|\beta_j| \tag{2.7}$$

O objetivo desta seção é mostrar que a regularização não se vale apenas da informação contida no conjunto de dados, mas também considera um conhecimento prévio do desenvolvedor sobre o aprendizado do modelo para obter uma melhor generalização. Em outras palavras, significa adicionar viés para diminuir variância.

2.2 Teoria dos Grafos

A teoria dos grafos é um campo da matemática dedicado ao estudo dos grafos, que são estruturas utilizadas para modelar diversos problemas. O grafo é a representação matemática de quaisquer objetos, chamados de vértices, e se há relação entre eles, arestas. É possível que o primeiro grafo da história tenha sido elaborado por Leonhard Euller para resolver o problema das sete pontes de Konigsberg (Euler, 1736). O grafo é uma ferramenta importante para modelar problemas, como por exemplo, redes de computadores, perfis em redes sociais, ruas e esquinas de uma cidade (Bondy & Murty, 1976). A literatura está repleta de ferramentas que podem ser utilizadas nesses modelos.

Em Bondy & Murty (1976), o grafo $\mathcal{G}$ é definido por uma tripla ordenada de um conjunto não vazio de vértices $\mathcal{V}(\mathcal{G})$, um conjunto de arestas $\mathcal{E}(\mathcal{G})$ e uma

função de incidência $\psi_{\mathcal{G}}$ que associa cada aresta de $\mathcal{G}$ a um par de vértices. Por exemplo, $\mathcal{G} = (\mathcal{V}(\mathcal{G}), \mathcal{E}(\mathcal{G}), \psi_{\mathcal{G}})$, onde $\mathcal{V}(\mathcal{G}) = \{v_1, v_2, v_3\}$, $\mathcal{E}(\mathcal{G}) = \{e_1, e_2, e_3, e_4\}$ e $\psi_{\mathcal{G}}(e_1) = v_1v_2$, $\psi_{\mathcal{G}}(e_2) = v_2v_3$, $\psi_{\mathcal{G}}(e_3) = v_1v_3$ e $\psi_{\mathcal{G}}(e_4) = v_3v_2$. Nesta dissertação será usada uma simplificação dessa definição, mas de igual representação, onde o conjunto $\mathcal{E}(\mathcal{G}) = \{\psi_{\mathcal{G}}(e_i)\}$, para toda aresta e_i do grafo. Dentre outras classificações, um grafo pode ser classificado como simples, conexo, planar e com pesos.

Um grafo simples é um grafo não direcionado, sem laços e sem arestas paralelas. Um grafo é considerado não direcionado quando uma aresta $e_i = v_a, v_b$ pode ser representada também como $e_i = v_b, v_a$. A ordem do par de vértices não importa, pois a conexão entre dois vértices não é direcionada. Um grafo é direcionado quando $e_i = v_a, v_b$ indica que a conexão parte do vértice v_a para o vértice v_b. O grafo direcionado também é chamado de dígrafo. Um grafo sem laços (ou *loop*) é um grafo onde não existe nenhuma aresta que conecta o mesmo vértice, $e_i \neq v_a, v_a$.

Um grafo é conexo quando é possível, através das arestas, encontrar o caminho entre quaisquer pares de vértices do grafo. Existe um conjunto de arestas que conecta o vértice v_a a v_b para todos os vértices $v_a \neq v_b$.

Um grafo é planar quando é possível representá-lo graficamente em um plano sem que haja qualquer intersecção entre as arestas.

Um grafo pode ter suas arestas associadas a pesos. Neste caso, o caminho entre um par de vértices é ponderado pelo peso de cada aresta. Em um grafo sem pesos, cada aresta vale uma unidade. Esse peso representa o custo da ligação entre os vértices.

A Figura 2.1 mostra um grafo simples, conexo e planar. A Eq. (2.8) mostra a representação matricial do grafo da Figura 2.1, chamada matriz de adjacência, onde $M[i][j] = 1$ indica que há conexão entre o vértice i e j e $M[i][j] = 0$ que não há conexão. A matriz de adjacência de um grafo simples tem que ser simétrica e de diagonal igual a zero. Um grafo é uma representação matemática de objetos e suas ligações e pode ser visualizado de infinitas formas, porém toda a informação que ele carrega está contida na matriz de adjacência, que é uma matriz $n_v \times n_v$, onde n_v é a quantidade de vértices do grafo.

$$\begin{bmatrix} 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 0 \end{bmatrix} \tag{2.8}$$

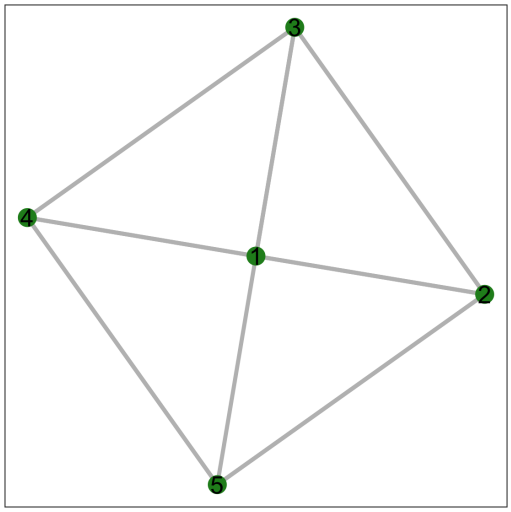

Figura 2.1: Exemplo de um grafo. Essa é uma representação visual de um grafo, um conjunto de vértices e arestas. A representação espacial de um grafo tem por objetivo apenas a visualização: as relações espaciais não são informações dos dados.

2.3 Grafo de Gabriel

O grafo de Gabriel (GG) é construído através das relações espaciais entre os dados de entrada em que a aresta entre dois vértices indica a proximidade entre eles. O GG é um grafo simples, planar e conexo construído a partir de uma regra simples. Considere que um vértice v_a de um grafo está conectado a todos os demais vértices. Exclua as arestas em que a hiperesfera de diâmetro igual a aresta contém algum outro vértice do grafo. A Figura 2.2 mostra essa regra visualmente. Segue agora a definição formal do grafo de Gabriel.

Considere um conjunto de dados $D = \{(x_i, y_i)|i = 1, ..., n\}$, em que $y_i \in \{+1, -1\}$ e $x_i \in \Re^d$, o grafo G_G de D com vértices $V = \{x_i \in D|i = 1, ..., n\}$ tem aresta E de vértices x_i e x_j se, e somente se, a inequação (2.9) é atendida, onde $\delta(\cdot, \cdot)$ é a distância euclidiana ao quadrado.

$$\delta(x_i, x_j)^2 \leq [\delta(x_i, x_k)^2 + \delta(x_j, x_k)^2], \forall x_k \in V \, e \, i \neq j \neq k \tag{2.9}$$

O grafo de Gabriel pode ser desenvolvido a partir da triangulação de Delaunay, que é construída a partir do diagrama de Voronoi. Essa outra maneira de entender a construção do grafo traz intuições sobre sua estrutura e propriedades. O diagrama de Voronoi decompõe o espaço de acordo com a disposição espacial dos dados de entrada (De Berg *et al.*, 2000). Essa decomposição é feita como se segue: dado o mesmo conjunto de pontos D, entre qualquer par de pontos é traçada a bissetriz, reta perpendicular que atravessa o ponto médio do segmento de reta entre esses dois pontos. O diagrama de Voronoi divide o hiperplano em polígonos convexos para cada observação de D formado pelo conjunto de bissetrizes mais próximas do ponto. A Figura 2.3 mostra um exemplo do diagrama de Voronoi. Por fim, cada elemento de D fica associado a um poliedro do diagrama. A triangulação de Delaunay (De Berg *et al.*, 2000), por sua vez, é o conjunto de pontos de D conectados entre si sempre que o poliedro associado é vizinho do outro. A Figura 2.3 mostra a triangulação resultante do diagrama de Voronoi. Por fim, o grafo de Gabriel é um subgrafo da triangulação de Delaunay, de acordo com Zhang & King (2002).

A ordem de complexidade da construção do grafo de Gabriel é $O(dn^3)$, onde n é a quantidade de dados e d a dimensão. Para avaliar cada par de vértices a complexidade é $O(n^2)$ e cada par precisa de $O(dn)$ operações.

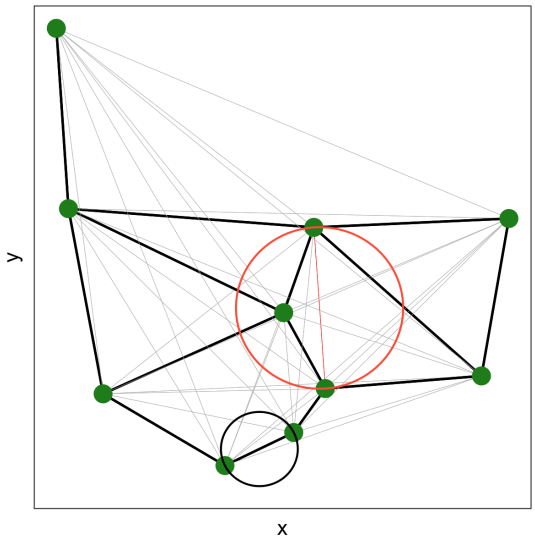

Figura 2.2: Exemplo da construção do grafo de Gabriel para um conjunto de dados: considere cada observação como um vértice do grafo e conecte todos os vértices entre si. Para cada aresta trace o círculo resultante da aresta como diâmetro e retire cada aresta em que há outro vértice contido no círculo traçado. Em contrapartida, mantenha as arestas que não atendem essa condição. A figura mostra duas possibilidades: em vermelho, uma aresta retirada pela presença de vértice no círculo. Em preto, uma aresta mantida pois não há vértice no círculo de diâmetro coincidente com a aresta. Todas as arestas em cinza também foram retiradas pela mesma razão da vermelha.

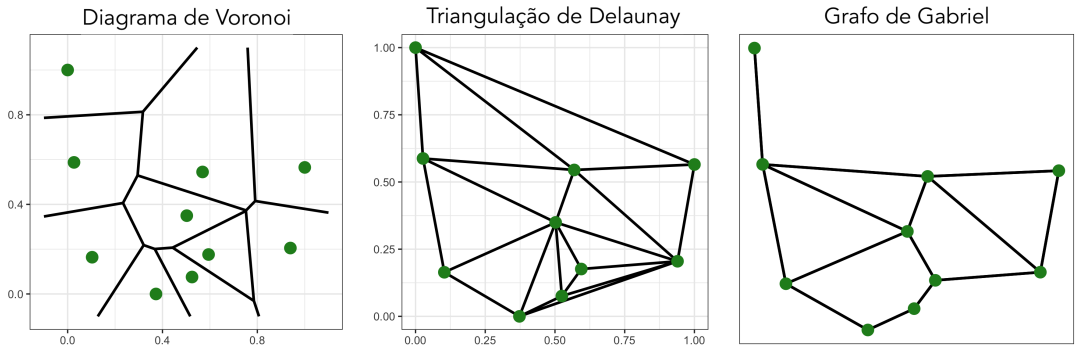

Figura 2.3: Esquerda: diagrama de Voronoi. Centro: triangulação de Delaunay. Direita: grafo de Gabriel. O gráfico do grafo de Gabriel não possui escala porque a representação espacial de um grafo é visual. O grafo é apenas um conjunto de vértices, conectados ou não.

2.4 Classificador baseado em grafo de Gabriel

Em sua tese de doutorado, Torres (2016) define um classificador baseado no grafo de Gabriel. Considere o grafo de Gabriel G_G construído do conjunto de dados $D = \{(x_i, y_i)|i = 1, ..., n\}$, em que $y_i \in \{+1, -1\}$ e $x_i \in \Re^d$. Defina como conjunto de Vetores de Suporte Estruturais (VSE) os vértices do grafo que tem conexão com vértices da classe oposta à sua, ou seja, $v_i \in$ VSE se existe uma aresta $e_k = v_i v_j$ que $y_i \times y_j = -1$. O conjunto de arestas $e_k = v_i v_j$ em que $y_i \times y_j = -1$ é definido como Arestas de Suporte (AS) e é a estrutura que definine o classificador. O classificador final é construído através do hiperplano formado pelo ponto médio mais próximo de uma amostra x que se deseja classificar; esse classificador é uma versão reduzida do classificador CHIP-CLAS proposto por Torres *et al.* (2015a). A Figura 2.4 mostra um exemplo dos hiperplanos que formam o classificador.

Inspirado na SVM, o classificador baseado no grafo de Gabriel também utiliza um subconjunto dos dados como parâmetros do classificador, ao invés de todo o conjunto dos dados de entrada, como o algoritmo *k-nearest neighbors* (KNN). No caso da SVM, esses são os vetores de suporte, dados que estão na região de separação dos dados. A construção desse classificador está baseada na intuição de que um vértice que pertence ao conjunto VSE habita na região de separação dos dados (Zhang & King, 2002).

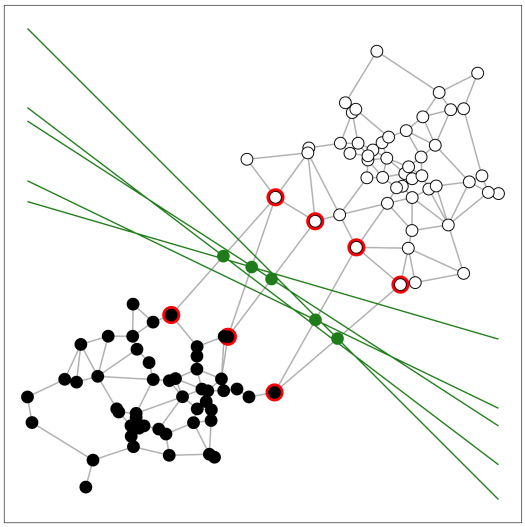

Figura 2.4: Classificador CHIP-CLAS reduzido. As classes são representadas pelos vértices de cores preta e branca. Os vértices marcados em vermelho são os VSE. Os vértices verdes são os pontos médios de cada AS. O hiperplano que passa por cada ponto médio é representado pelas retas verdes.

2.5 SVM

Os classificadores de margem larga têm ganhado espaço na literatura de aprendizado de máquina nas últimas duas décadas. Apesar de métodos conhecidos como *Boosting* (Freund *et al.*, 1996), *Gaussian Mixture Models* (Sha & Saul, 2006) e *Direct Parallel Perceptrons* (Fernandez-Delgado *et al.*, 2011) serem baseados na maximização de margem, sua popularidade se deve às SVM (Boser *et al.*, 1992), estado da arte em se tratando de classificadores de margem larga.

Considere novamente o conjunto de dados $D = \{(x_i, y_i)|i = 1, ..., n\}$, em que $y_i \in \{+1, -1\}$ e $x_i \in \Re^d$. Considere o mapeamento dos dados de entrada através de p funções $\psi_j(x_i, z_j)$, onde z_j são parâmetros dessa função de mapeamento. A classificação $\hat{y}_i$ do modelo de SVM é dada pela função sinal de uma combinação linear das funções de mapeamento,

$$\hat{y}_i = sinal\left[\sum w_j \psi_j(x_i, z_j) + b\right] \tag{2.10}$$

onde w_j representa os pesos dessa combinação linear.

Para que todos os padrões de entrada sejam classificados corretamente, a desigualdade $\hat{y}_i y_i \geq 1$ deve ser satisfeita para todo i. Além disso, para que a separação seja de margem máxima é necessário encontrar a norma mínima do vetor de pesos da combinação linear $\mathbf{w}$. Sendo assim, o problema de aprendizado pode ser descrito como um problema de otimização (Eq. 2.11).

$$\begin{aligned} \min_{\mathbf{w}} \quad & \frac{1}{2}\mathbf{w}^T\mathbf{w} \\ \text{s.t.} \quad & y_i(\mathbf{w}^T\psi(x_i, \mathbf{Z}) + b) \geq 1 \quad ,\forall i \end{aligned} \tag{2.11}$$

Uma vez que a matriz $\mathbf{Z}$ é conhecida, o problema de otimização da Eq. (2.11) tem um mínimo global, já que as restrições são lineares e a função de custo é convexa. O problema dual pode ser obtido ponderando as restrições através dos multiplicadores de Lagrange (Boser *et al.*, 1992), como é mostrado nas Eq. (2.12) - (2.15).

$$J(\mathbf{w}, b, \alpha) = \frac{1}{2}\mathbf{w}^T\mathbf{w} - \sum_{i=1}^{n} \alpha_i \left[y_i(\mathbf{w}^T\psi(x_i, \mathbf{Z}) + b) - 1 \right] \tag{2.12}$$

$$\frac{\partial J}{\partial \mathbf{w}} = \mathbf{w} - \sum_{i=1}^{n} \alpha_i y_i \psi(x_i, \mathbf{Z}) = 0 \tag{2.13}$$

$$\mathbf{w} = \sum_{i=1}^{n} \alpha_i y_i \psi(x_i, \mathbf{Z}) \tag{2.14}$$

$$\frac{\partial J}{\partial b} = \sum_{i=1}^{n} \alpha_i y_i = 0 \tag{2.15}$$

Por fim, o problema de otimização dual que representa o aprendizado da SVM com $\mathbf{Z}$ conhecido é da seguinte forma, como mostra a Eq. (2.16).

$$\begin{aligned} \min_{\alpha} \quad & \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i\alpha_j y_i y_j \psi(x_i, \mathbf{Z})^T\psi(x_i, \mathbf{Z}) \\ \text{s.t.} \quad & \sum_{i=1}^{n} \alpha_i y_i = 0 \\ & \alpha_i \geq 0 \end{aligned} \tag{2.16}$$

As SVM regulam a capacidade do modelo na presença de sobreposição de classes através de suas variáveis de folga (Boser *et al.*, 1992). As variáveis de folga das SVM agem como parâmetros de flexibilidade que permitem que algumas observações estejam dentro da margem ou do lado errado. Sendo assim, elas são ignoradas pela função-objetivo na otimização. Na prática, a utilização das variáveis de folga resulta na eliminação da sobreposição na região de separação, já que essas amostras deixam de influenciar a função-objetivo.

A formulação das SVM com variáveis de folga incorpora o termo ξ_i na desigualdade $\hat{y}_i y_i \geq 1$, agora $\hat{y}_i y_i + \xi_i \geq 1$. O problema primal, Eq. (2.11), pode ser reescrito agora com a variável de folga ξ_i e um parâmetro $C \geq 0$ de regularização, como mostrado na Eq. (2.17).

$$\begin{aligned} \min_{\mathbf{w}} \quad & \frac{1}{2}\mathbf{w}^T\mathbf{w} + C\sum_{i=1}^{n}\xi_i \\ \text{s.t.} \quad & y_i(\mathbf{w}^T\psi(x_i, \mathbf{Z}) + b) + \xi_i \geq 1 \quad , \forall i \end{aligned} \tag{2.17}$$

A formulação com variáveis de folga tem o seu problema dual obtido também através dos multiplicadores de Lagrange, como mostrado nas Eq. (2.18) - (2.22).

$$J_\xi(\mathbf{w}, b, \alpha, \xi) = \frac{1}{2}\mathbf{w}^T\mathbf{w} - \sum_{i=1}^{n}\alpha_i\left[y_i(\mathbf{w}^T\psi(x_i, \mathbf{Z}) + b) + \xi_i - 1\right] + C\sum_{i=1}^{n}\xi_i \tag{2.18}$$

$$\frac{\partial J_\xi}{\partial \mathbf{w}} = \mathbf{w} - \sum_{i=1}^{n}\alpha_i y_i \psi(x_i, \mathbf{Z}) = 0 \tag{2.19}$$

$$\mathbf{w} = \sum_{i=1}^{n}\alpha_i y_i \psi(x_i, \mathbf{Z}) \tag{2.20}$$

$$\frac{\partial J_\xi}{\partial b} = \sum_{i=1}^{n}\alpha_i y_i = 0 \tag{2.21}$$

$$\frac{\partial J_\xi}{\partial \xi_i} = C - \alpha_i = 0 \tag{2.22}$$

Por fim, o problema de otimização dual que representa o aprendizado das SVM com folga é da seguinte forma, como mostra a Eq. (2.23).

$$\begin{aligned} \min_{\alpha} \quad & \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j \psi(x_i, \mathbf{Z})^T \psi(x_i, \mathbf{Z}) \\ \text{s.t.} \quad & \sum_{i=1}^{n} \alpha_i y_i = 0 \\ & 0 \leq \alpha_i \leq C \end{aligned} \tag{2.23}$$

As SVM utilizam um algoritmo de otimização para encontrar o decisor que separa as classes com margem máxima, como é mostrado na Figura 2.5. A medida da margem é a distância entre os vetores de suporte e o decisor.

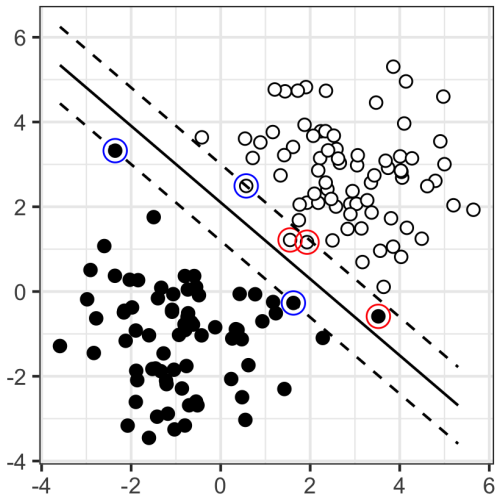

Figura 2.5: As diferentes classes dos dados são representadas pelas cores branco e preto. A reta contínua representa o decisor gerado pelas SVM e as retas pontilhadas, a sua margem. Os vetores suporte escolhidos pelas SVM estão circulados em azul e, em vermelho, temos as observações ignoradas de acordo com a variável de folga (regularização). As SVM maximizam a margem do classificador.

2.6 Redes Neurais RBF

As redes neurais *radial basis function* (RBF) possuem apenas uma camada escondida onde os padrões de entrada são mapeados por funções radiais. As funções radiais são caracterizadas por serem monotônicas em relação a um ponto central.

Considere novamente o conjunto de dados $D = \{(x_i, y_i)|i = 1, ..., n\}$, em que $y_i \in \{+1, -1\}$ e $x_i \in \Re^d$. A formulação de uma rede neural RBF está apresentada na Eq. (2.24), onde $h_j(x, z_j)|j = 1, ..., p$ é a função radial de parâmetro z_j, p é o número de funções radiais, $\mathbf{Z}$ é a matriz de parâmetros z_j, $\mathbf{x}$ é a matriz de entrada $[x_1, x_2, ..., x_n]^T$ e $\mathbf{w} = w_j|j = 0, ..., p$ são os pesos da camada de saída da rede.

$$f(\mathbf{x}, \mathbf{Z}) = \sum_{j=1}^{p} w_j h_j(\mathbf{x}, z_j) + w_0 \tag{2.24}$$

De forma geral, as funções radiais são definidas por um vetor de médias $\mathbf{m}$ e uma matriz de parâmetros $\mathbf{\Sigma}$, onde $h(\mathbf{x}, [\mathbf{m}, \mathbf{\Sigma}]) = \phi((\mathbf{x} - \mathbf{m})^T \mathbf{\Sigma}^{-1} (\mathbf{x} - \mathbf{m}))$. A forma mais utilizada para a função $\phi(\cdot)$ é a função gaussiana, que resulta na Eq. (2.25).

$$h(\mathbf{x}, [\mathbf{m}, \mathbf{\Sigma}]) = \exp(-\frac{1}{2}(\mathbf{x} - \mathbf{m})^T \mathbf{\Sigma}^{-1} (\mathbf{x} - \mathbf{m})) \tag{2.25}$$

Se considerada a independência das variáveis de entrada $\mathbf{x}$, a matriz $\mathbf{\Sigma}$ será diagonal, reduzindo a quantidade de parâmetros a serem encontrados no treinamento da rede de p^2 para p. Outra simplificação adicional comumente utilizada é considerar todos esses parâmetros iguais a uma constante r^2. Portanto, $\mathbf{\Sigma} = r^2\mathbf{I}$, onde $\mathbf{I}$ é a matriz identidade.

Considere que $\mathbf{H}$, Eq. (2.26), é a projeção dos padrões de entrada pelas funções radiais $h_k(x_k, z_k)$, e $z_k = \{c_k, r_k\}\,|k = 1, ..., p$ são os parâmetros estimados de centro e raio, e p é o número de funções radiais.

$$\boldsymbol{H} = \begin{bmatrix} h_1(x_1, z_1) & h_2(x_1, z_2) & ... & h_p(x_1, z_p) \\ h_1(x_2, z_1) & h_2(x_2, z_2) & ... & h_p(x_2, z_p) \\ ... & ... & ... & ... \\ h_1(x_N, z_1) & h_2(x_N, z_2) & ... & h_p(x_N, z_p) \end{bmatrix} \tag{2.26}$$

O treinamento da rede é feito através da minimização da Eq. (2.27), que representa o erro entre os rótulos com penalização para valores altos de w_j. A minimização segue o desenvolvimento mostrado nas Eq. (2.28) - (2.30).

$$J = \sum_{i=1}^{N} (y_i - \hat{y}_i)^2 + \sum_{j=1}^{p} \lambda_j w_j^2 \tag{2.27}$$

$$\frac{\partial J}{w_j} = \frac{\partial J}{w_j} \sum_{i=1}^{N} (y_i - \hat{y}_i)^2 + \frac{\partial J}{w_j} \sum_{j=1}^{p} \lambda_j w_j^2 = 0 \tag{2.28}$$

$$2 \sum_{i=1}^{N} (\hat{y}_i - y_i)\, h(x_i, z_j) + 2\lambda_j w_j = 0 \tag{2.29}$$

$$\sum_{i=1}^{N} \hat{y}_i h(x_i, z_j) + \lambda_j w_j = \sum_{i=1}^{N} y_i h(x_i, z_j) \tag{2.30}$$

Em sua forma matricial, a Eq. (2.30) é dada pela Eq. (2.31), onde $\mathbf{H}$ é a matriz de projeção de entrada, $\hat{\mathbf{y}} = \mathbf{Hw}$ é a saída da rede, λ é um parâmetro de regularização, $\mathbf{I_p}$ é a matriz identidade de dimensão p e $\mathbf{y}$ é o vetor de rótulos.

$$\mathbf{H}^T \hat{\mathbf{y}} + \lambda \mathbf{I}_p \mathbf{w} = \mathbf{H}^T \mathbf{y} \tag{2.31}$$

O vetor de pesos $\mathbf{w}$ da camada de saída da RBF será obtido através da Equação 2.32.

$$\mathbf{w} = \left(\mathbf{H}^T \mathbf{H} + \lambda \mathbf{I}_p\right)^{-1} \mathbf{H}^T \mathbf{y} \tag{2.32}$$

A classificação final é obtida através da simplificação mostrada na Eq. (2.33), em que $sinal(\cdot)$ representa a função sinal.

$$\hat{\mathbf{d}}^* = sinal(\mathbf{Hw}) \tag{2.33}$$

O erro de classificação e_i de um dado x_i é obtido conforme a Eq. (2.34), com valores possíveis de $\{0, 1\}$, devido à simplificação realizada.

$$e_i = \frac{(y_i - \hat{y}_i^*)^2}{4} \tag{2.34}$$

2.7 Resumo do Capítulo

Este capítulo apresenta um breve referencial teórico para o desenvolvimento do restante do trabalho. É discutido aqui o conceito de regularização, teoria dos grafos, o grafo de Gabriel, SVM e redes neurais RBF.

Capítulo 3

Regularização de Redes Neurais RBF

A primeira contribuição deste trabalho investiga a utilização de propriedades extraídas do grafo de Gabriel para a regularização de redes neurais RBF. Aqui as funções radiais foram definidas como uma função gaussiana, Eq. (2.25), e seus parâmetros estimados através do método CG-RBF (Torres *et al.*, 2013), que utiliza o grafo de Gabriel nessa estimativa. Este capítulo investiga a regularização através da retirada de funções radiais e a representatividade do conjunto de vértices VSE, que são os vértices do grafo que fazem conexão entre classes distintas.

Centros e raios das funções radiais de uma RBF são estimados na literatura através de heurísticas como K-médias e suas variações (Sing *et al.*, 2003): *Fuzzy C-means* (FCM), redes *Self-Organizing Maps* (SOM) (Bouchired *et al.*, 1998) e winner-takes-all (WTA). O método CG-RBF (Torres *et al.*, 2013), mencionado anteriormente, não utiliza parâmetros para essa estimativa, uma vez que toda a informação é extraída diretamente das relações espaciais dos dados representada pelo grafo de Gabriel.

O controle da capacidade de uma rede RBF pode ser realizado através da regularização de Tikhonov, também chamada de norma L2, representada pelo termo de penalização $\sum_{j=1}^{p} \lambda_j w_j^2$, Eq. (2.27). É mostrado que a magnitude do vetor de pesos da camada de saída indica generalização (Haykin, 1994). Outro

indicativo é a quantidade de funções radiais que projetam os dados de entrada na saída da camada escondida.

Em sua abordagem, o CG-RBF realiza a regularização através da filtragem de ruído e escolha de funções radiais. Essa escolha é feita considerando, após a filtragem, o conjunto VSE (Torres *et al.*, 2015a). Essa estratégia parte do pressuposto de que os vértices desse conjunto estão na margem de separação entre classes. Para investigar o comportamento dos VSE na determinação dos parâmetros das funções radiais, é proposta uma comparação entre duas possíveis interpretações para os VSE: indicadores de ruído, portanto as funções radiais associadas a estes vértices são eliminadas para realizarmos a regularização; indicadores de margem, assim as funções radiais associadas aos VSE são mantidas e as demais eliminadas. O objetivo aqui é então inferir a sua importância e função na regularização do modelo. O grafo extraí informação diretamente dos dados para regularização de redes RBF.

3.1 Estimativa de Parâmetros das funções radiais da RBF

As propriedades do grafo de Gabriel podem ser utilizadas para estimar os centros e raios das funções radiais da RBF. As funções radiais são definidas aqui como uma função gaussiana $h(x, z) = \exp\left(\frac{\|x-c\|^2}{2\sigma^2}\right)$, onde $z = \{c, \sigma\}$.

Considere o conjunto de dados $D = \{(x_i, y_i)|i = 1, ..., N\}$, em que $y_i \in \{+1, -1\}$ e $x_i \in \Re^d$. $GG_D(V, E)$ é o Grafo de Gabriel resultante do conjunto de dados D. O centro c_i, parâmetro de $z_i = \{c_i, r_i\}$ da função radial $h_i(x_i, z_i)$, é estimado como o vértice $v_i \in GG_D(V, E)$. O raio σ_i é estimado como

$$\sigma_i = \frac{1}{N_{e_j}} \sum_{e_j} \frac{\eta(e_j)}{2}, \tag{3.1}$$

onde $\eta(e_j)$ é o comprimento da aresta e_j, e_j é toda aresta $\in GG_D(V, E)$ associada ao vértice v_i e N_{e_j} é o número de arestas. Ou seja, a média da metade do comprimento das arestas de v_i. Um exemplo é mostrado na Figura 3.1.

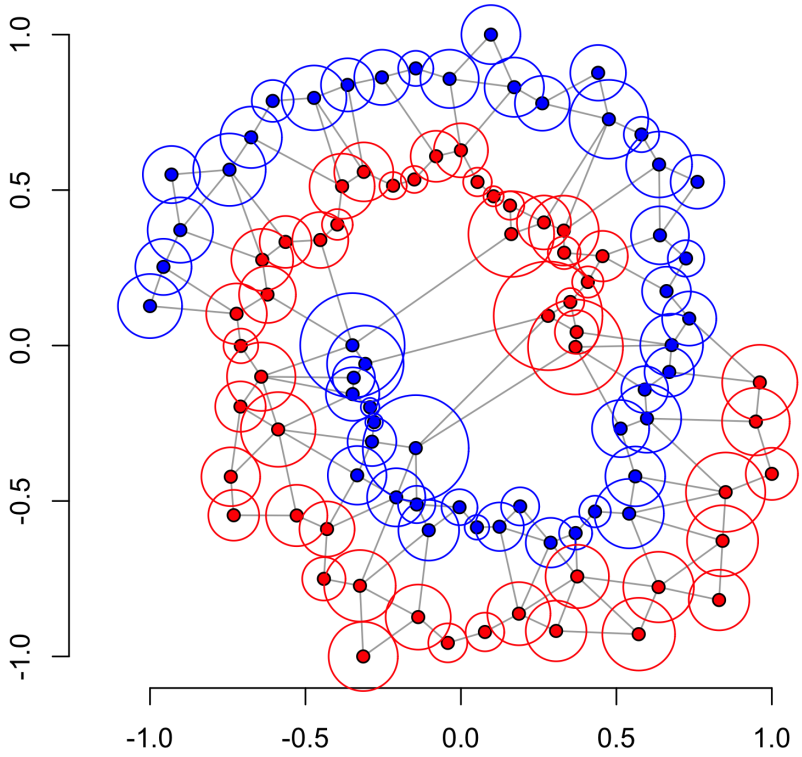

Figura 3.1: Centros (pontos) e raios (círculos) da RBF estimados através das propriedades do Grafo de Gabriel.

A avaliação dos padrões de entrada pelas funções radiais resultam na matriz de projeção H, Eq. (2.26).

3.2 Regularização de Tikhonov

A escolha do parâmetro de regularização λ, Eq. (2.27), é feita através da minimização do erro de validação. A estimativa do erro, por sua vez, é feita através de *Leave One Out Cross Validation* (LOOCV), conforme a Eq. (3.2)

$$LOOCV = \frac{1}{N}\sum_{k=1}^{N}\left(\left[Hw^{(k)}\right]_k - y_k\right)^2, \tag{3.2}$$

onde $[Hw]_k$ representa a estimativa $\hat{y}_k$ e $w^{(k)}$ corresponde ao vetor de pesos calculado através da Eq. (2.32), retirando-se a linha k de H. Uma versão simplificada de LOOCV, mostrada na Eq. (3.3), retorna o erro de validação cruzada em unidade de amostras rotuladas erroneamente por amostra. A Eq. (3.3) foi obtida da Eq. (3.2) utilizando a função $\mathbf{L}(\hat{d}_k, d_k)$, onde $\mathbf{L}(\hat{d}_k, d_k) = 0$ se $\hat{d}_k = d_k$, $\mathbf{L}(\hat{d}_k, d_k) = 1$ se $\hat{d}_k \neq d_k$ e $\hat{d}_k = \text{sinal}([Hw]_k)$. Portanto, $LOOCV^*$ pode ser entendida como $\frac{Ne}{N}$, onde Ne é número de amostras rotuladas erroneamente.

$$LOOCV^* = \frac{1}{N}\sum_{k=1}^{N}\mathbf{L}(\hat{d}_k, d_k) \tag{3.3}$$

A regularização é feita, então, penalizando os valores altos de w_i.

3.3 Regularização por eliminação de funções radiais - Proposta I

Para o conjunto de dados D, de N amostras, foram definidas N funções radiais através do grafo $GG_D(V, E)$. Cada uma dessas funções radiais está centrada em um vértice do grafo. Como mostrado em outros trabalhos (Torres *et al.*, 2015b), o conjunto de VSE está na margem de separação dos dados. Aqui, duas intuições são possíveis: o conjunto VSE representa a margem de separação ou pode ser

considerado como ruído. A proposta I avalia o efeito de regularização causado pela eliminação das funções radiais associadas a vértices desse conjunto.

A estratégia é retirar toda função radial h_i associada a um vértice v_i que tem conexão com a classe oposta, ou seja, um vértice $v_i \in \mathcal{S}$, sendo $\mathcal{S}$ o conjunto de vértices VSE. Foram desenvolvidos testes para investigar esse comportamento.

Os testes foram realizados na base de dados *spiral* para valores de desvio padrão dp entre $[1 \times 10^{-2}, 2 \times 10^{-2}; \text{por } 1 \times 10^{-2}]$, como são mostrados alguns exemplos na Figura 3.2. Quanto maior o valor de dp, maior a sobreposição das espirais e, consequentemente, maior o ruído nos dados. Para cada base de dados foi calculado o $LOOCV^*$, Eq. 3.3, para 4 diferentes configurações: primeiro, sem regularização, w é calculado como na Eq. (2.32) para $\lambda = 0$. Na segunda configuração foi escolhido o valor de λ que minimiza a Eq. (3.3). Para a terceira e quarta configurações foram repetidas as escolhas de λ do primeiro e do segundo, porém com a eliminação das funções radiais h_i associadas a vértices v_i que têm aresta com vértices de classes diferentes. A Tabela 3.1 resume as configurações descritas (Config 1 - Config 6), e os resultados estão apresentados no Capítulo 5, Figura 5.1.

A minimização da Eq. (3.3) para obtenção de λ ótimo nas configurações 2 e 4, Tabela 3.1, foi feita através do método *Brent* (Brent, 1971). Além dos resultados de $LOOCV^*$ para cada configuração, também foi registrado o valor de λ ótimo para cada passo do teste.

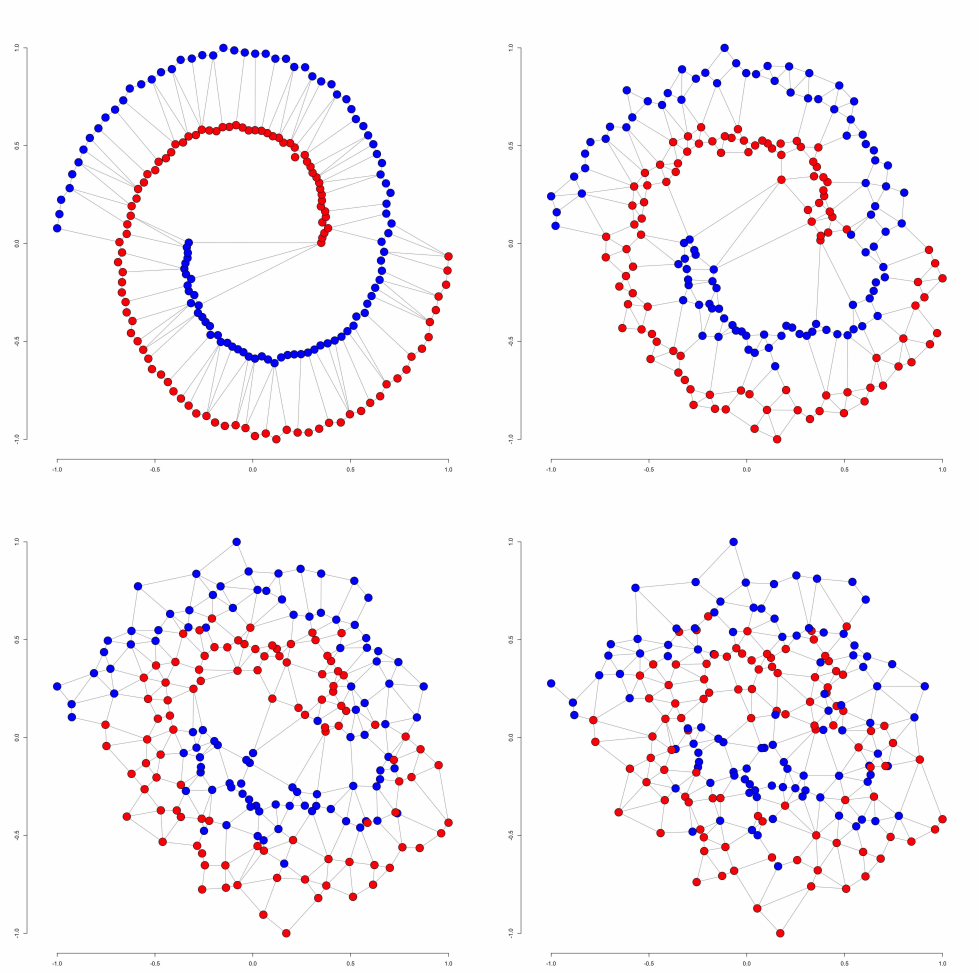

Figura 3.2: Base de dados *spiral* gerada com diferentes valores de desvio padrão: [0.01,0.06,0.13,0.2].

Tabela 3.1: Retirada de funções radiais h_i e λ para cada configuração. As configurações 3 e 4 são referentes à proposta I, enquanto 5 e 6 são referentes à proposta II.

Configuração	Retirada de Funções Radiais h_i	λ
Config 1	Nenhuma	0
Config 2	Nenhuma	ótimo
Config 3	associadas aos VSE	0
Config 4	associadas aos VSE	ótimo
Config 5	não associadas aos VSE	0
Config 6	não associadas aos VSE	ótimo

3.4 Regularização por eliminação de funções radiais - Proposta II

Na primeira proposta foi investigado o comportamento da tentativa de regularização através da eliminação de funções radiais associadas ao VSE. Na proposta II a intuição oposta é desenvolvida: os VSE como indicadores de margem. As funções radiais h_i associadas a vértices que fazem conexão com a classe oposta são mantidas e todas as demais são eliminadas.

No grafo de Gabriel, um vértice que faz conexão com a classe oposta é crucial para a classificação (Torres *et al.*, 2015b): em problemas sem sobreposição de classes eles estão na margem de separação. Já em problemas com sobreposição de classes, são possíveis candidatos a ruído. Assim, intuitivamente, espera-se que a eliminação de funções radiais feita na proposta I tenha melhor desempenho para base de dados com sobreposição, e que a proposta II tenha melhor desempenho em bases de dados não sobrepostas.

O experimento descrito anteriormente foi repetido para o cenário atual acrescentando duas novas configurações para comparação: configuração 5, h_i associadas aos VSE são mantidas e as demais eliminadas, e $\lambda = 0$. A configuração 6 representa a mesma configuração anterior, porém para λ escolhido através da minimização da Eq. (3.2). A Tabela 3.1 também resume as configurações 5 e 6.

3.5 Resumo do capítulo

Este capítulo investiga a utilização de propriedades extraídas do grafo de Gabriel para a regularização de redes neurais RBF. Uma metodologia é apresentada através da estratégia de retirada de funções radiais do CG-RBF, classificador que utiliza o grafo de Gabriel para estimar os parâmetros de suas funções radiais.

Capítulo 4

Regularização do CHIP-CLASS

A segunda contribuição deste trabalho é a investigação acerca de como a capacidade do classificador baseado em grafo de Gabriel, o CHIP-CLASS (Torres, 2016), pode ser controlada utilizando apenas informações extraídas do próprio grafo. O classificador proposto é sensível à ocorrência de sobreposição dos dados, como qualquer classificador de margem larga. Na sobreposição, o classificador tende a sobreajustar os dados, resultando em baixa generalização do modelo. Estado da arte, as SVMs lidam com este problema utilizando variáveis de folga (Boser *et al.*, 1992), que são parâmetros de tolerância a ruído, Seção 2.5. Essa tolerância pode ser comparada à eliminação da sobreposição na região de separação, uma vez que essas amostras deixam de influenciar o valor da função objetivo. Esse parâmetro de regularização também é escolhido através de otimização. Em contraste, o CHIP-CLASS propõe uma estratégia de filtragem dos dados para controlar a capacidade do modelo sem utilização de parâmetros.

Neste capítulo serão apresentadas a sensibilidade do classificador à sobreposição de dados, a forma como essa filtragem é realizada no CHIP-CLASS através do grau do vértice e uma abordagem alternativa baseada em métricas de distância em grafos.

4.1 O problema da sobreposição de amostras

A sobreposição de classes é um problema geral em aprendizado de máquina. Uma vez que um classificador é treinado para minimização do erro, lidar com a

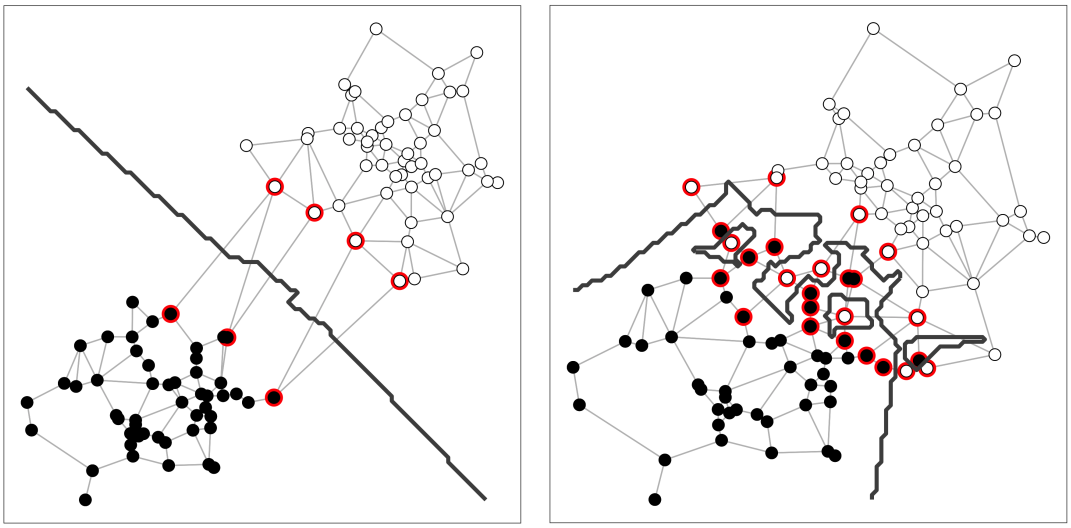

Figura 4.1: Classificador baseado em grafo de Gabriel separando duas classes. As classes são representadas pelas cores dos vértices, branco e preto. Os vértices marcados em vermelho representam aqueles escolhidos como VSE (Vetores Suporte Estruturais). Eles definem o classificador final gerado pelo hiperplano que passa pelo ponto médio de cada aresta entre os vetores suporte. No lado esquerdo temos as classes sobrepostas e, do lado direito, classes não sobrepostas.

sobreposição faz parte do dilema fundamental da regularização da capacidade do modelo (James *et al.*, 2013). Esta seção expõe o problema da sobreposição de classes para o classificador baseado no grafo de Gabriel CHIP-CLASS.

Os dados foram gerados a partir de uma distribuição gaussiana bivariada de mesma variância, porém de média $\mu_1 \neq \mu_2$ para as diferentes classes. O que determina a superfície final de decisão é o conjunto VSE. Quando as classes estão distantes uma da outra o VSE descreve bem a fronteira de separação entre as classes, como mostrado na Figura 4.1 (a). No entanto, quando há sobreposição de classes, o decisor final passa a contornar cada observação da região de separação resultando em uma superfície de decisão grosseira, o que causa *overfitting*, como mostrado na Figura 4.1 (b). O resultado do *overfitting* é a baixa generalização do modelo e, portanto, baixa eficiência.

A filtragem de ruído age então como regularização para o classificador baseado em grafo de Gabriel. A retirada de amostras da região de sobreposição aumen-

tará a capacidade de generalização do classificador, diminuindo o *overfitting* e melhorando sua performance para dados de teste.

4.2 Filtragem via Grau do Vértice

Além das soluções estado da arte para o problema descrito, já foi sugerida uma solução baseada no grafo de Gabriel. Em Torres (2016), o autor propõe uma abordagem baseada no grau do vértice para determinar ruídos. O grau de um vértice x_i indica a quantidade de vértices conectados a x_i.

Tomemos como exemplo um grafo de Gabriel G_G formado pelo conjunto de pontos $x_i|i = 1, ..., N$ e $x \in \Re^n$, com uma classe $y_i = \{-1, +1\}$ atribuída para cada x_i. É calculada uma medida de qualidade $q(x_i)$ para cada vértice de acordo com a Eq. (4.1), em que $\mathcal{A}(x_i)$ representa o grau do vértice x_i e $\hat{\mathcal{A}}(x_i)$ o grau do vértice x_i considerando apenas vértices de mesma classe. É retirado então todo vértice x_i que valida a inequação (4.2), sendo t_{y_j} a média dos valores de $q(x)$ para todo x pertencente à classe y_j.

$$q(x_i) = \frac{\hat{\mathcal{A}}(x_i)}{\mathcal{A}(x_i)} \tag{4.1}$$

$$q(x_j) < t_{y_j} \tag{4.2}$$

A filtragem pelo grau do vértice considera como ruído todos os vértices que tenham uma relação aresta-mesma-classe e total-arestas abaixo de um limiar. Limiar este escolhido através da média dos valores encontrados. É importante ressaltar que essa estratégia garante que, sempre que utilizada, uma ou mais amostras serão consideradas ruído, mesmo que não haja sobreposição dos dados. Isso acontece porque sempre teremos medidas de $q(x_i)$ menores que a média de seus valores para um conjunto de vértices x (exceto em um caso raro em que a medida de qualidade $q(x_i)$ é igual para todos os vértices). A Figura 4.2 mostra visualmente como essa filtragem é calculada.

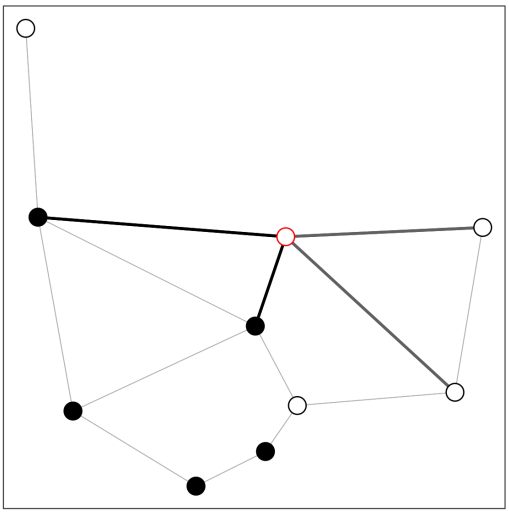

Figura 4.2: Filtragem baseada no grau do vértice. O grafo é a representação via grafo de Gabriel de um conjunto de dados dividido em duas classes representadas pelas cores preto e branco. Seja qv o vértice marcado com borda vermelha, pertencente à classe branca. $q(qv) = \frac{2}{4}$, pois o vértice em questão possui um total de 4 arestas, sendo 2 com vértices de mesma classe. No exemplo dado, o limiar $t_{classe=branco} = \frac{17}{30}$, calculado como a média dos valores de $q(\cdot)$ para todos os vértices da classe branca, indica que o vértice qv seria retirado na filtragem, uma vez que $q(qv) < t_{classe=branco}$

.

4.3 Medidas de Centralidade Em grafos

4.3.1 Centralidade de Vértices

Medidas de centralidade são amplamente utilizadas em problemas práticos modelados por grafos. Essas medidas procuram explicar o quão central ao grafo um determinado vértice é. Na análise de redes sociais, por exemplo, a centralidade é uma medida que procura encontrar os agentes mais importantes da rede (Everett & Borgatti, 2005). Outro exemplo é a utilização no cálculo das palavras mais influentes de um texto, quando ele é modelado como um grafo, no método CRA (Corman *et al.*, 2002).

Uma variedade de medidas específicas de centralidade foi proposta desde os anos 50. No entanto, em 1979, Freeman separou as medidas de centralidade em três grupos - *degree*, *closeness* e *betweenness* - e apresentou a medida canônica de cada grupo (Freeman *et al.*, 1979). Desde então, essas são as medidas de centralidade em grafo mais utilizadas, juntamente com a medida baseada em autovetores, *Eigenvector Centrality*, proposta por Bonacich (Bonacich, 1972). As medidas são descritas como:

- *Degree Centrality* mede o grau de cada vértice do grafo. O grau indica, simplesmente, a quantidade de ligações que um determinado vértice tem na rede.
- *Closeness Centrality* mede o inverso da soma da *distância geodésica* (Buckley & Harary, 1990) entre um vértice e todos os demais. Distância geodésica é a distância entre dois vértices calculada no grafo. Ela equivale ao comprimento do caminho mais curto entre esses dois vértices.
- *Betweennes Centrality* mede o número de caminhos mais curtos entre qualquer par de vértices que passam por um determinado vértice.
- *Eigenvector Centrality* mede a centralidade de um vértice a patir dos autovalores e autovetores da matriz de adjacência do grafo (Bonacich, 1972).

A Figura 4.3 mostra as quatro medidas de centralidade citadas avaliadas para um mesmo grafo de Gabriel, construído a partir de um conjunto de dados aleatório

oriundo de uma distribuição uniforme bidimensional. A métrica de grafo que será definida mais adiante é baseada em *closeness cetrality*. Dentre as quatro medidas apresentadas, ela é a que melhor descreve a centralidade de um vértice em relação a sua posição espacial no grafo. Intuitivamente, o vértice mais central está no centro do grafo, e a medida vai diminuindo radialmente.

4.3.2 Centralidade de Grupos

O conceito de centralidade de um vértice pode ser estendido para centralidade de um grupo de vértices dentro do grafo (Everett & Borgatti, 2005). Vamos aqui estender a medida *closeness centrality* para *closeness group centrality*.

Considere o grafo $G(V, E)$ e $\mathcal{C}$, subconjunto de V formando um grupo de vértices. A *closeness centrality* do grupo $\mathcal{C}$ é calculada como o inverso da soma das distâncias entre todo vértice x fora do grupo, ao grupo $\mathcal{C}$, definido na Eq. (4.3) como $\mathcal{D}_f(x, \mathcal{C})$. Seja o conjunto $\mathcal{D}_x$ formado pela distância entre um vértice x fora do grupo $\mathcal{C}$ e todos os vértices pertencentes a esse grupo. A distância $\mathcal{D}_f(x, \mathcal{C})$ pode ser calculada como a média de $\mathcal{D}_x$, por exemplo. A distância entre uma observação e um grupo de observações pode ser calculada de várias outras maneiras (máximo, mínimo, mediana, moda, etc.), como já conhecido pelos processos de agrupamento hierárquico (Johnson, 1967).

$$
\begin{aligned}
&\mathcal{D}_x = \{\phi(x, c), c \in \mathcal{C}\}\, x \in \{V - \mathcal{C}\} \\
&\text{Onde } \phi(\cdot, \cdot) \text{ é a distância geodésica.} \\
&\mathcal{D}_f(x, \mathcal{C}) = f(\mathcal{D}_x) \\
&\text{Onde } f(\cdot) = \text{mínimo, máximo, média ou mediana.} \\
&\textit{Group Closeness} = \sum_{x \in V - \mathcal{C}} \mathcal{D}_f(x, \mathcal{C}) \\
&\textit{Normalized Group Closeness} = \frac{|V - \mathcal{C}|}{\textit{Group Closeness}} \\
&\text{Onde } |\cdot| \text{ representa o tamanho do conjunto .}
\end{aligned}
\tag{4.3}
$$

A métrica de grafo definida na seção seguinte é baseada na distância entre vértice-grupo, ou seja, $\mathcal{D}_f(x, \mathcal{C})$. Observe que para o cálculo dessa medida é preciso definir o grupo $\mathcal{C}$ e a função $f(\mathcal{D}_x)$.

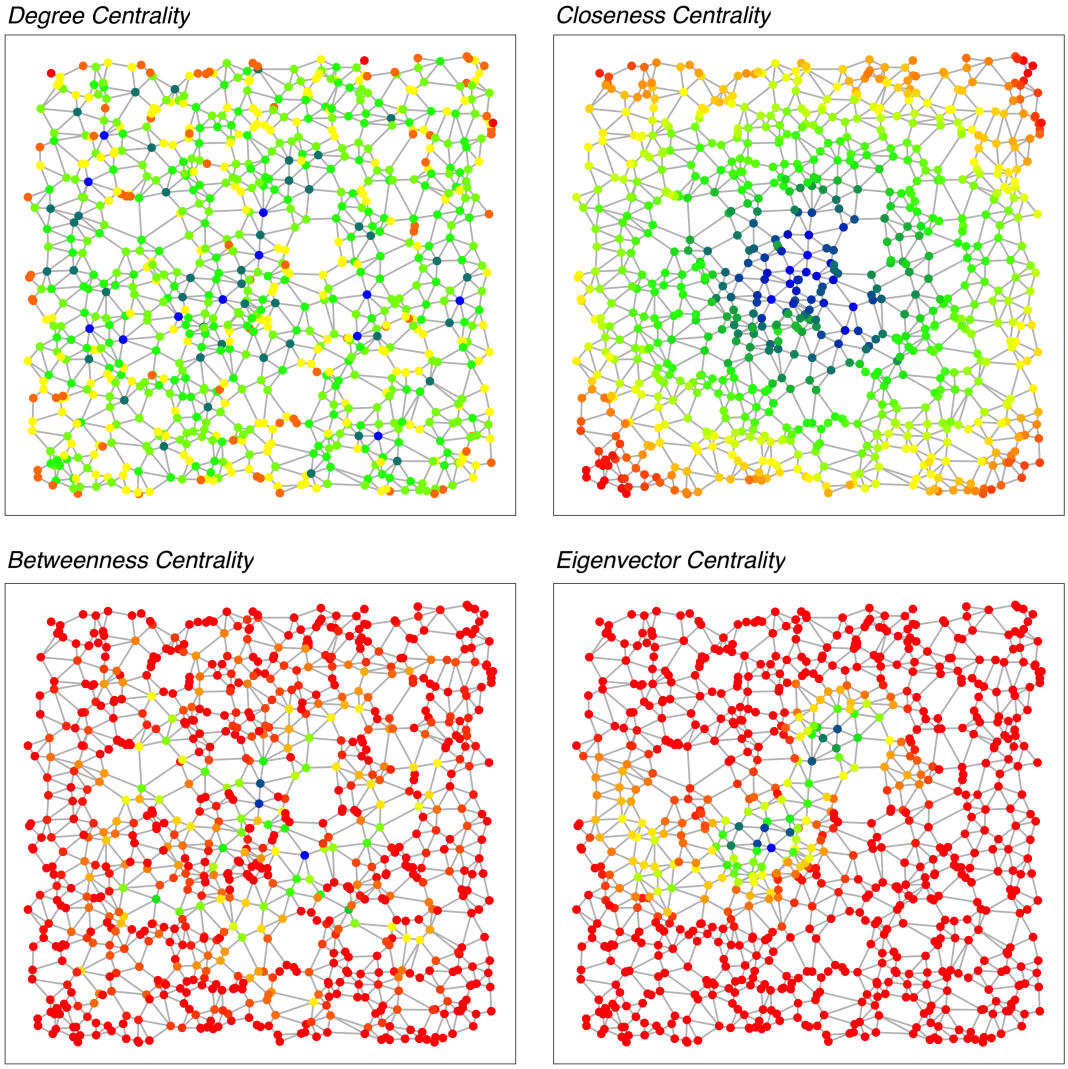

Figura 4.3: Medidas de centralidade em grafos. O grafo de Gabriel foi construído para um conjunto de pontos de uma distribuição de probabilidade uniforme bidimensional. A respectiva medida de centralidade foi calculada no grafo e as cores representam o valor da medida de centralidade. Vermelho = valores mais baixos e Azul = valores mais altos.

4.4 Distância Entre Classes

A eliminação da sobreposição de amostras na margem de separação é formulada aqui a partir de propriedades intrínsecas ao grafo de Gabriel, representação estrutural dos dados. A relação de distância relativa entre classes ao longo do grafo é calculada através da métrica de grafo $D_C(\cdot)$.

Considere $G_G = (V, E)$ um grafo de Gabriel rotulado de ordem N. Cada rótulo define a classe de cada vértice $v \in V$. Cada aresta $e \in E$ tem um peso associado w igual à distância euclidiana entre as amostras que geraram aquela aresta. A distância entre classes $D_C : V \to \Re \geq 0$ de um vértice $v \in V$ é definida como a média da distância entre v e todos os vértices de classes distintas à sua própria, Eq. (4.4).

Definição $D_C(\cdot)$

$$
\begin{aligned}
&\mathcal{D}_v = \{\phi(v, c), c \in \mathcal{C}\}\, v \in \{V - \mathcal{C}\} \\
&\text{Onde } \phi(\cdot,\cdot) \text{ é a distância geodésica e} \\
&\mathcal{C} \text{ é o conjunto vértices de } G_G \text{ de classe diferente de v.} \\
&\mathcal{D}_f(v, \mathcal{C}) = f(\mathcal{D}_v) \\
&\text{Onde } f(\cdot) = \text{função média}(\cdot) \\
&D'_C(v) = \mathcal{D}_f(v, \mathcal{C}) \\
&\mathbf{D_C(v)} = \frac{D'_C(v)}{max\,\{D'_C(u)\}}, \forall u \in V \ .
\end{aligned}
\tag{4.4}
$$

Em outras palavras, a métrica $D_C(\cdot)$ para um vértice qualquer x equivale ao valor de $\mathcal{D}_f(x, \mathcal{C})$, Eq. (4.3), em que $\mathcal{C}$ representa o conjunto de vértices de classe diferente da classe de x e $f = \text{média}(\mathcal{D}_x)$.

A métrica admite valores entre 0 e 1: $D_C(u_1) = 0$ quando os vértices da outra classe estão no mesmo ponto que u_1 e $D_C(u_1) = 1$ quando u_1 é o vértice mais distante.

As amostras da região da margem de separação podem ser identificadas através de $D_C(\cdot)$. A Figura 4.4 representa um grafo sintético dividido em duas classes com pontos dispostos como dois hexágonos regulares. O valor especificado em cada vértice indica a distância média entre aquele ponto e os pontos da outra

classe, como definido na Eq. (4.4). Assim, percebe-se que os menores valores dessa métrica estão na fronteira entre as classes e aumentam à medida em que se afastam dela.

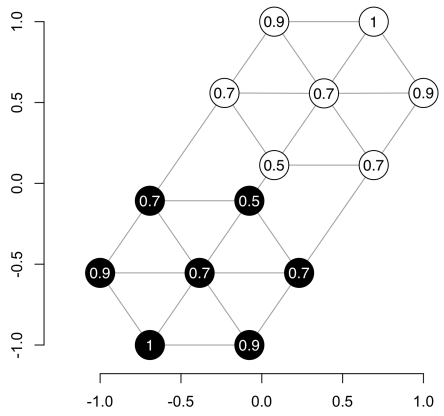

Figura 4.4: $D_C(\cdot)$ calculado para cada vértice do grafo sintético. A figura mostra que os menores valores da métrica estão no encontro entre as duas classes e aumentam à medida em que se afastam dela.

4.5 Propriedades Dc

4.5.1 Separação não-linear

A métrica de grafo $D_C(\cdot)$, definida na Eq. (4.4), pode ser calculada para qualquer conjunto de amostras modeladas via grafo de Gabriel e o seu resultado avaliado para diferentes formatos de margem.

A base de dados sintética *benchmark Fullmoon* tem margem de separação não linear, além de alguns dados sobrepostos, como mostrado no lado esquerdo da Figura 4.5. Seja $\mathcal{W}$ o conjunto de vértices do grafo de Gabriel construído através das amostras de *Fullmoon*, mostrado no lado direito da Figura 4.5. A interpolação dos dados de $D_C(\mathcal{W})$ gera uma superfície que sugere que a métrica identifica amostras de uma margem de separação não linear, resultando em valores menores

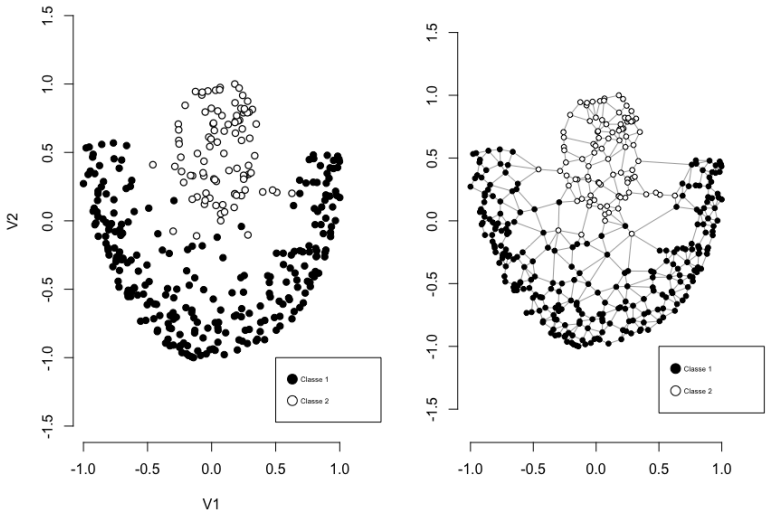

Figura 4.5: Esquerda: Base de dados sintética *benchmark Fullmoon*. Direita: Base de dados sintética *benchmark Fullmoon* modelada pelo grafo de Gabriel.

para essas amostras. O gráfico resultante dessa interpolação está apresentado na Figura 4.6 e mostra a distribuição aproximada da métrica para qualquer ponto entre as amostras. O vale formado pela superfície acompanha a margem de separação não linear entre as classes. Outra evidência desse resultado está nas curvas de nível e no mapa de calor da superfície aproximada, Figuras 4.7(a) e 4.7(b), respectivamente.

O valor de $D_C(\cdot)$ de uma amostra pode assumir qualquer valor entre 0 e 1, dependendo apenas da sua localização na representação estrutural. A distribuição da métrica para todas as observações é mostrada pelo histograma da Figura 4.8, juntamente com uma curva gaussiana de média e desvio padrão estimados dos valores de $D_C(\mathcal{W})$.

4.5.2 Normalidade

O comportamento normal de $D_C(\cdot)$ para a base *Fullmoon* foi confirmado através do teste de Kolmogorov-Smirnov (Wilcox, 2005) que resultou em um p-valor de 0.7492, que implica na falha em rejeitar a hipótese nula de que as amostras seguem

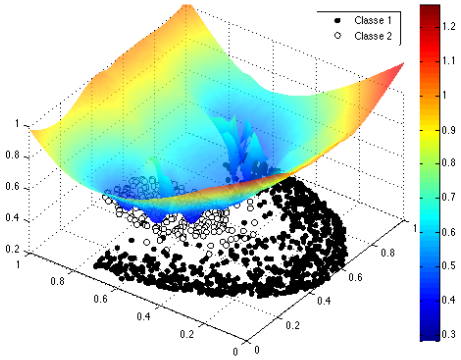

Figura 4.6: Gráfico da Superfície de $D_C(\mathcal{W})$, base sintética *benchmark Fullmoon*.

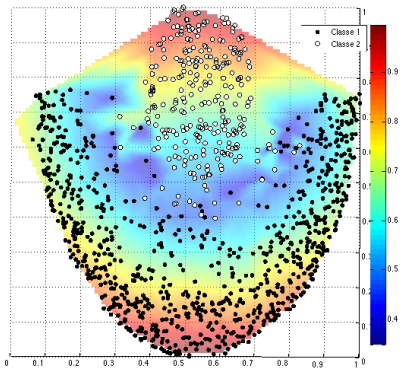

Figura 4.7: Resultado da métrica de grafo $D_C(\mathcal{W})$, base sintética *benchmark Fullmoon*.

uma distribuição normal, para um intervalo de confiança de 95%. Essa é uma característica interessante que será explorada em bases de dados reais no próximo capítulo.

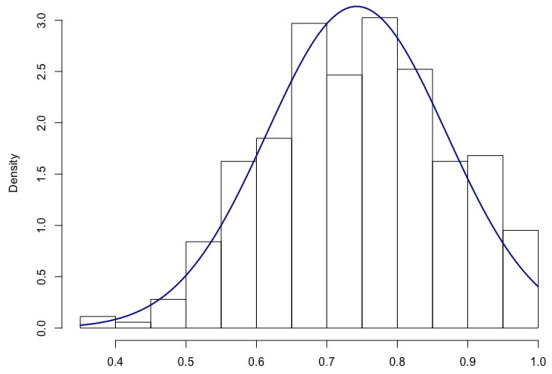

Figura 4.8: Histograma de $D_C(\mathcal{W})$ normalizado, base sintética *benchmark Full-moon*.

4.5.3 Limiar de Filtragem

Uma das fraquezas do método de filtragem baseado no grau do vértice é o seu limiar, como descrito na seção 4.2. Uma vez que são retiradas amostras com índice menor que a média de todos os índices, sempre teremos amostras retiradas. Nesse caso, o método retira observações mesmo quando não há sobreposição de dados. Este trabalho propõe um limiar de filtragem único para cada vértice que não partilha dessa mesma fraqueza.

O limiar de filtragem escolhido é uma variação da própria métrica $D_C(\cdot)$. $D_C(\cdot)$ é calculada como $\mathcal{D}_f(x, \mathcal{C})$, Eq. (4.4), em que $\mathcal{C}$ representa o conjunto de vértices de classe diferente da classe de x. Considere o complemento de $D_C(\cdot)$, indicado por $\bar{D_C}(\cdot)$, como a mesma medida, porém utilizando $\{V - \mathcal{C}\}$ como grupo, o conjunto de vértices de **mesma classe** de x.

$$\bar{D_C}(\cdot) = \mathcal{D}_f(x, \{V - \mathcal{C}\}) \tag{4.5}$$

A filtragem se dá então retirando todo vértice $v \in V$ que atende a desigualdade na Eq. (4.6). A intuição por trás desse princípio é de que uma amostra pode ser

considerada ruído se estiver mais próxima da classe oposta à sua do que da sua própria classe.

$$\begin{aligned} D_C(v) &< D_{\bar{C}}(v) \\ \frac{D_C(v)}{D_{\bar{C}}(v)} &< 1 \ . \end{aligned} \tag{4.6}$$

4.6 Resumo do capítulo

Este capítulo apresenta uma metodologia para regularização do classificador CHIP-CLAS. A regularização é feita através da definição de uma métrica de grafo para filtragem de amostras da região de separação entre classes.

Capítulo 5

Resultados

Este capítulo apresenta os resultados da utilização de informações extraídas do grafo de Gabriel na regularização de modelos. Na primeira seção, as propostas do capítulo 3 são utilizadas para regularização de redes RBF através da retirada de funções radiais. Na segunda seção, a métrica de grafo elaborada no capítulo 4 é utilizada para regularização do CHIP-CLASS e comparada com o método original.

5.1 Resultado da regularização de redes neurais RBF

O grafo de Gabriel foi utilizado para estimar os parâmetros das funções radiais de uma rede neural RBF, como explicado na Seção 3.1. O experimento foi projetado para entender se o conjunto especial de vértices do grafo, o VSE, pode ser utilizado para regularizar a rede através da retirada de funções radiais. Esse experimento utiliza uma base de dados sintética onde o desvio padrão dos dados é controlado para variar a sobreposição de classes, como mostrado na Figura 3.2. Por fim, são realizados experimentos em bases de dados reais.

Para melhor compreensão, os resultados estão organizados em três subseções: (1) Os resultados da proposta I (eliminação de funções radiais) são comparados com o *base line* e com a regularização de Tikhonov; (2) São adicionados à discussão os resultados da proposta II comparados aos demais resultados obtidos; e

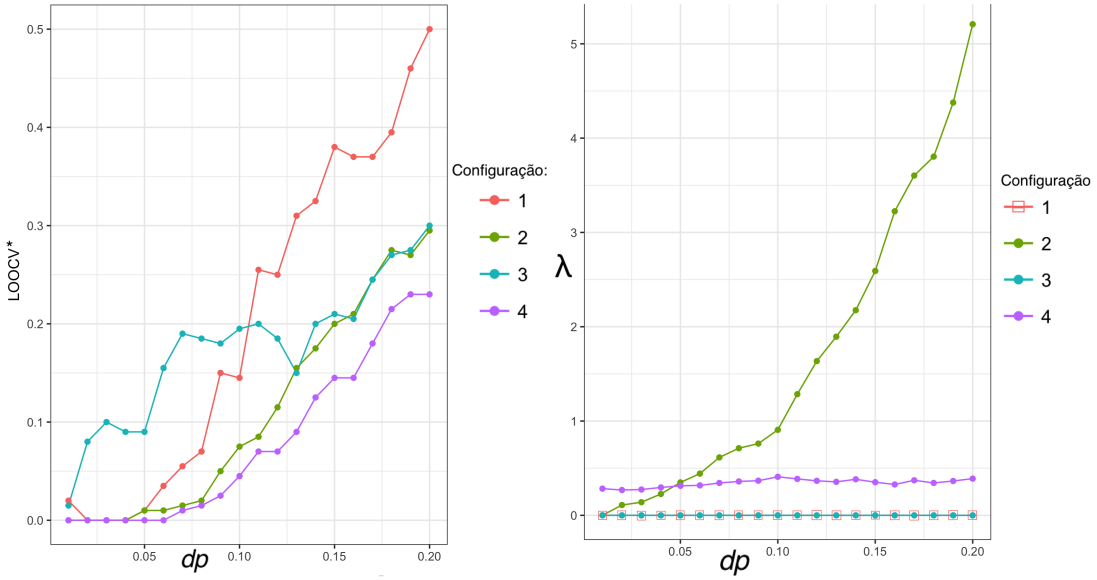

Figura 5.1: Resultados de testes feitos na base *spiral*. 1) sem regularização; 2) regularização de Tikhonov; 3) regularização via proposta I; 4) regularização de Tikhonov + proposta I.

(3) Experimentos em bases de dados reais são realizados e discutidos.

5.1.1 Resultados - Proposta I

Para cada valor de desvio padrão *dp* associado à distribuição que gerou a base *spiral* de teste, foi calculado o valor de LOOCV, para as 4 diferentes configurações (Tabela 3.1). Este resultado está apresentado na Figura 5.1.

O efeito da retirada de funções radiais associadas ao conjunto VSE pode ser observado comparando o resultado entre as configurações 1 e 3. Observa-se que para bases sobrepostas, a retirada dessas funções age como reguladora, diminuindo o erro. No entanto o desempenho é muito pior para base de dados não sobrepostos. Isso indica que funções radiais que representam bem os dados estão sendo retiradas e isso prejudica o modelo.

Em comparação com a regularização de Tikhonov observa-se que para base de dados sobrepostos as duas regularizações têm desempenho semelhante (configurações 2 e 3). O lado direito da Figura 5.1, que mostra o λ escolhido na regularização de Tikhonov, ajuda a entender a relação entre as duas alternativas

de regularização: naturalmente, quanto maior a sobreposição, maior o valor de λ para a penalização da norma de $\mathbf{w}$. No entanto, a retirada de funções radiais faz com que o valor de λ continue estável. Isso é um indício de que a maior penalização de $\mathbf{w}$ e a retirada de funções radiais são equivalentes para a base sintética testada.

As duas formas de regularização, quando combinadas, obtêm o melhor resultado. Isso é um indício de que as duas formas, apesar de terem desempenhos semelhantes na sobreposição, podem ter caráteres de regularização diferentes. Um estudo mais aprofundado é necessário para tirar conclusões mais sólidas desses indícios.

Ao avaliar esses resultados buscando um entendimento maior de qual informação pode ser extraída do grafo de Gabriel, algumas conclusões interessantes podem ser apontadas. Os resultados indicam que a intuição de que o conjunto de vértices VSE representa ruído é verdadeira na sobreposição. Por outro lado, em base de dados não sobrepostos, a retirada dessas funções radiais prejudica o desempenho da rede. Na próxima seção será possível avaliar o impacto da intuição inversa: a retirada de funções radiais não associadas ao conjunto VSE.

5.1.2 Resultados - Proposta II

A proposta II procurou manter as funções radiais associadas aos VSE e eliminar as demais, considerando a intuição de que os vértices do conjunto VSE são indicadores de margem. Esse resultado está apresentado na Figura 5.2.

Primeiro vamos comparar a variação do erro de teste entre as configurações 1 e 5 que representam, respectivamente, nenhuma regularização e regularização através da escolha de funções radiais associadas ao conjunto VSE. Observa-se que o desempenho é melhor tanto para sobreposição quanto para classes separadas. Já a comparação entre a escolha ou retirada dessas funções radiais, configurações 5 e 3, respectivamente, observa-se que o desempenho da escolha (configuração 5) é superior para dados não sobrepostos. No caso da sobreposição, no entanto, o comportamento da configuração 3 é mais estável. Isso é um indício de que as funções radiais associadas ao conjunto VSE representam bem os dados quando

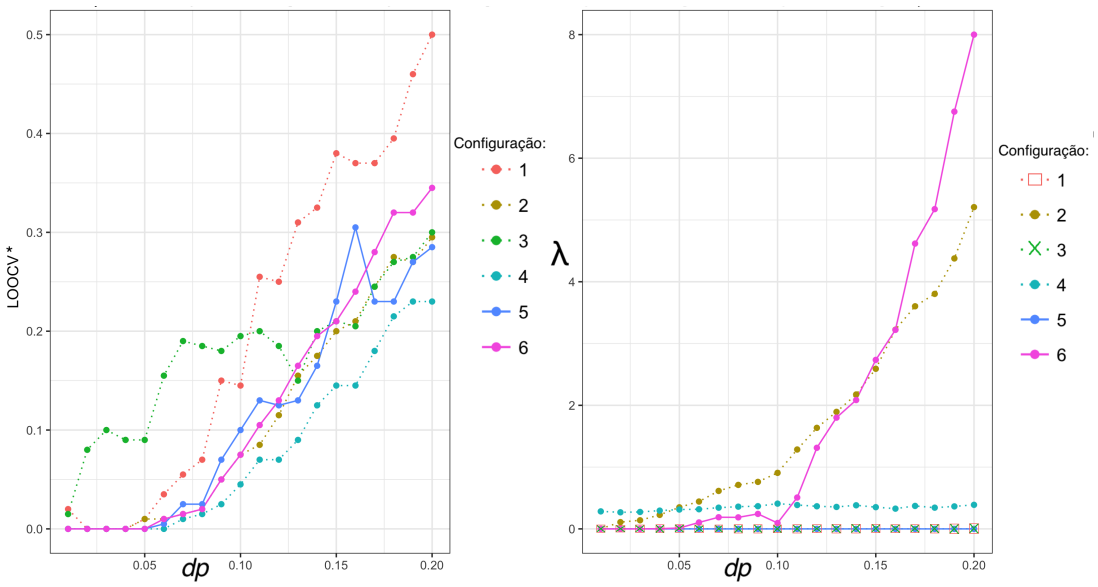

Figura 5.2: Resultados de testes feitos na base *spiral*. 1) sem regularização; 2) regularização de Tikhonov; 3) regularização via proposta I; 4) regularização de Tikhonov + proposta I. 5) regularização via proposta II. 6) regularização de Tikhonov + proposta II.

não há sobreposição. Quando a sobreposição aumenta, essa característica vai ficando menos evidente.

O lado direito da Figura 5.2 traz novas evidências de que o VSE passa a representar ruído quando a sobreposição aumenta. Quando apenas essas funções são mantidas, o valor de λ na combinação com a regularização de Tikhonov aumenta ainda mais com a sobreposição. O efeito em λ observado para a configuração 4 fica ainda mais evidente.

A conclusão deste experimento é que o conjunto de vértices especiais chamado VSE do grafo de Gabriel são estruturas que representam bem as funções radiais que devem ser escolhidas para regularizar uma rede RBF. No entanto, quando a sobreposição de classes aumenta, essa representação vai ficando cada vez mais fraca.

Essa é uma informação importante no estudo do classificador baseado no grafo de Gabriel. Sabe-se que a melhor escolha do conjunto VSE é sensível a sobreposição das classes. Essa estrutura parece ser bem representativa dos dados no geral. Porém, ainda é necessário avanço nos estudos para dar mais robustez ao classificador.

5.1.3 Bases de dados reais

Experimentos em base de dados reais também foram realizados para investigar o desempenho das metodologias propostas. 14 bases de dados reais binárias extraídas do *UCI* (Lichman, 2013) foram utilizadas, exceto por *breastHess*, um problema de expressão genética (Hess *et al.*, 2006) e *Appendicitis*, extraído do *KEEL* (Alcalá-Fdez *et al.*, 2011). Mais informações sobre as bases de dados utilizadas são apresentadas na Tabela 5.1. Os dados passaram por um pré-processamento de remoção de valores faltantes e reescala entre $\{-1, 1\}$. As bases de dados *segmentation* e *glass* não são originalmente binárias, mas foram reduzidas como mostrado em (Castro & Braga, 2013). A estimativa do erro de teste é feita da mesma forma que os experimentos anteriores: LOOCV*, Eq. (3.3). Os resultados são mostrados na Tabela 5.2 e avaliam a performance das 6 diferentes configurações propostas (Tabela 3.1) por LOOCV*.

Tabela 5.1: Informações sobre as 14 bases de dados reais em que os experimentos foram realizados.

Base de dados	abv	Amostras	Atributos
Segmentation	seg	210	18
Glass	gla	214	9
Appendicitis	app	106	7
Ionosphere	ion	351	33
Breastcancer	bre	683	9
Australian	aus	690	14
Diabetes	dia	768	8
BreastHess	bhe	133	30
Bupa	bup	345	6
Haberman	hab	306	3
Banknote	ban	1372	4
Fertility	fer	100	9
Parkinons	par	195	22
ILPD	ilp	579	10

Tabela 5.2: Resultado de LOOCV* das seis configurações para as 14 bases de dados reais.

abv	Config1	Config2	Config3	Config4	Config5	Config6
seg	0.114	0.014	0.224	0.100	0.033	**0.005**
gla	0.061	**0.033**	0.107	0.037	**0.033**	**0.033**
app	0.255	0.142	0.226	**0.113**	0.217	0.151
ion	0.453	**0.046**	0.148	0.131	0.245	0.048
bre	0.095	**0.031**	0.063	0.038	0.082	0.034
aus	0.529	0.139	0.184	0.228	0.454	**0.138**
dia	0.431	**0.236**	0.271	0.286	0.392	**0.236**
bhe	0.241	**0.173**	0.241	0.226	0.308	**0.173**
bup	0.400	0.310	0.472	0.478	0.522	**0.307**
hab	0.402	0.291	0.278	**0.265**	0.301	0.284
ban	0.259	0.001	0.153	**0.000**	0.026	0.023
fer	0.350	**0.120**	**0.120**	**0.120**	0.390	**0.120**
par	0.056	0.046	0.210	0.169	0.174	**0.041**
ilp	0.420	0.297	0.314	**0.287**	0.356	0.297
Média $Rank(\mathcal{L})$	5.2500	2.0000	4.2143	2.8929	4.7143	**1.9286**

Para comparação entre a performance das diferentes configurações foi utilizado o teste de *Friedman*. Quanto menor o valor da média do $Rank(\mathcal{L})$, melhor a performance da configuração, como pode ser verificado na Tabela 5.2.

Os experimentos mostram que, para as bases de dados reais analisadas, a regularização de Tikhonov domina sobre a regularização através da eliminação de funções radiais pois os melhores resultados são para as configurações 2, 4 e 6, nos quais ela está presente. Por outro lado, tendo em vista o custo de um algoritmo de otimização, a a regularização porposta poderia ser uma alternativa à regularização de Tikhonov. Porém, essa estratégia ainda carece de mais estudo.

5.2 Resultados da Regularização do CHIP-CLAS

Como explicado na Seção 2.4, o classificador CHIP-CLASS é definido através das arestas formadas pelo conjunto de vértices VSE (Torres, 2016), construído através do hiperplano formado pelo ponto médio mais próximo da amostra que se deseja classificar. A métrica de grafo definida na Seção 4.4 é utilizada para regularização do CHIP-CLAS. Para isso, dois experimentos foram projetados para investigar as características da métrica proposta. No primeiro, é verificada a distribuição de $D_C(\cdot)$ quando aplicada em bases reais. Em bases sintéticas foi verificado que essa distribuição é normal. No segundo experimento é verificada a eficiência do classificador para as duas formas de regularização: pela abordagem original, via grau do vértice; e pela métrica proposta neste trabalho.

Ambos experimentos foram realizados em 19 bases de dados reais extraídas do *UCI* (Lichman, 2013), exceto por *bupa*, extraída do *KEEL* (Alcalá-Fdez *et al.*, 2011), e por dois problemas de expressão gênica: *golub* (Golub *et al.*, 1999) e *breastCancerHess* (Hess *et al.*, 2006). Os dados passaram por um pré-processamento de remoção de valores faltantes e reescala entre $\{-1, 1\}$. Todas as bases de dados são originalmente binárias, exceto *segmentation* e *glass*, que foram reduzidas a binárias como mostrado em (Castro & Braga, 2013).

5.2.1 Verificação empírica da normalidade de $D_C(\cdot)$ em bases reais

O primeiro experimento mostrou que a distribuição da métrica de grafo D_C quando aplicada a bases reais pode ser aproximada a uma distribuição normal para algumas bases de dados. O experimento seguiu os seguintes passos:

1. Construção do Grafo de Gabriel;
2. Avaliação da métrica $D_C(\cdot)$;
3. Teste de normalidade.

Um grafo de Gabriel foi construído para cada uma das 19 bases de dados. A métrica de grafo $D_C(\cdot)$, Equação 4.4, foi avaliada para cada estrutura de dados. O teste de Kolmogorov-Smirnov (Wilcox, 2005) foi utilizado para verificar se as medidas de $D_C(\cdot)$ vêm de uma distribuição normal. A Tabela 5.3 mostra os resultados do teste: p-valor maior que 0.05 indica falha em rejeitar a hipótese nula de que as amostras seguem uma distribuição normal, para um nível de confiança 95%. 11 das 19 bases têm distribuição normal.

5.2.2 Filtragem baseada em $D_C(\cdot)$

O CHIP-CLAS foi utilizado para classificar os dados das 19 bases mencionadas. O experimento foi realizado utilizando validação cruzada com 10 *folds*. A performance média AUC é mostrada na Tabela 5.4. O objetivo é medir o efeito da filtragem na performance AUC do classificador. Para cada combinação de *folds* da validação cruzada, o experimento seguiu os seguintes passos:

1. Construção do grafo de Gabriel;
2. Filtragem;
3. Reconstrução do grafo de Gabriel (sem ruído);
4. Classificação;
5. Avaliação da performance AUC.

Tabela 5.3: Resultado do teste Kolmogorov-Smirnov de normalidade para $D_C(\cdot)$ aplicado a bases reais. Os valores maiores que 0.05 estão em negrito e indicam que as bases que têm distribuição normal.

Base	Id	D	pvalue
segmentation	1	0.08	**1.20E-01**
glass	2	0.03	**9.84E-01**
appendicitis	3	0.08	**4.43E-01**
ionosphere	4	0.09	5.72E-03
sonar	5	0.04	**8.80E-01**
breastcancer	6	0.11	1.93E-07
australian	7	0.08	1.45E-04
diabetes	8	0.06	9.17E-03
breastHess	9	0.07	**5.40E-01**
bupa	10	0.07	**5.47E-02**
haberman	11	0.06	**2.57E-01**
banknote	12	0.07	1.17E-06
fertility	13	0.13	**6.96E-02**
parkinsons	14	0.12	9.05E-03
climate	15	0.05	**1.96E-01**
ILPD	16	0.06	3.06E-02
german	17	0.03	**2.02E-01**
heart	18	0.10	1.36E-02
golub	19	0.08	**7.44E-01**

O classificador foi testado para três diferentes modelos: no primeiro, sem filtragem, elimina-se os passos 2 e 3. No segundo modelo foi utilizada a filtragem dos dados baseada no grau do vértice (Torres, 2016). Por fim, no terceiro modelo, foi utilizada a filtragem baseada na métrica de grafo, $D_C(\cdot)$, apresentada neste trabalho, e explicada a seguir.

A filtragem proposta baseada em $D_C(\cdot)$ é feita da seguinte forma: seja $\mathcal{V}_i$ o conjunto de vértices da base de dados de $id = i$ modelada via grafo de Gabriel, sendo $i = 1, 2...19$, Tabela 5.3. $D_C(\mathcal{V}_i)$ é o vetor resultante da avaliação de $D_C(v)$ para todo $v \in \mathcal{V}_i$ e retira-se todo $u \in \mathcal{V}_i$ que satisfaz a Eq. (4.6) em cada iteração da validação cruzada. A performance média AUC para os três modelos é mostrada na Tabela 5.4.

Os três modelos de classificação foram comparados através do teste de *Friedman*, indicado em Demšar (2006) para comparação de múltiplos classificadores. O teste ranqueia os três modelos, como mostrado na Tabela 5.4. Quanto menor o valor da Média do Rank $R(\mathcal{L})$, melhor classificado é o modelo. Para verificar a significância das diferenças entre os valores de $R(\mathcal{L})$, encontrados pelo teste de *Friedman*, foi utilizado o teste post-hoc com correção de *Bonferroni* (Demšar, 2006), que utiliza a distribuição *t-Student* ajustada para múltiplas comparações. O resultado dessa comparação é mostrado na Figura 5.3, que indica que o classificador com filtragem baseada em $D_C(\cdot)$ é estatisticamente diferente e, portanto, mais eficiente do que o classificador sem filtragem. O mesmo não pode ser verificado para o classificador com filtragem baseada no grau do vértice.

O principal resultado desse trabalho é a comparação entre a filtragem baseada no grau do vértice e $D_C(\cdot)$. Os testes foram realizados em 19 bases reais do *UCI Repository*. Foi realizado um teste estatístico não-paramétrico para testar a hipótese nula de que não há diferença na performance média entre os classificadores e um teste post-hoc de múltiplas comparações para encontrar a configuração de melhor desempenho. Como mostrado na Figura 5.3, o classificador com filtragem baseada em $D_C(\cdot)$ tem melhor desempenho médio que o classificador sem filtragem. Isso não ocorre na filtragem baseada no grau do vértice, que não tem desempenho superior ao classificador sem filtragem. Esse resultado mostra que o novo método de filtragem é promissor.

Tabela 5.4: Resultados: média de AUC e desvio padrão para três modelos de classificadores. Os melhores resultados estão apresentados em negrito. N_t/N_a indicam, respectivamente, o número total de amostras da base e o número de atributos.

Base	Classificador Sem Filtro	Classificador com Filtragem baseada no Grau do Vértice	Classificador com Filtragem baseada em $Dc(\cdot)$	N_t/N_a
segmentation	**0.790 ± 0.193**	0.785 ± 0.154	0.744 ± 0.083	210/18
glass	0.768 ± 0.178	0.921 ± 0.106	**0.933 ± 0.082**	214/9
appendicitis	0.404 ± 0.210	0.762 ± 0.173	**0.775 ± 0.173**	106/7
ionosphere	0.838 ± 0.058	**0.877 ± 0.083**	0.813 ± 0.078	351/33
sonar	**0.824 ± 0.065**	0.728 ± 0.084	0.720 ± 0.086	208/60
breastcancer	0.972 ± 0.017	0.972 ± 0.015	**0.976 ± 0.015**	683/9
australian	0.769 ± 0.033	0.801 ± 0.019	**0.833 ± 0.051**	690/14
diabetes	0.630 ± 0.067	**0.727 ± 0.037**	0.711 ± 0.05	768/8
breastHess	0.686 ± 0.149	0.827 ± 0.120	**0.845 ± 0.115**	133/30
bupa	**0.616 ± 0.094**	0.561 ± 0.066	0.579 ± 0.107	345/6
haberman	0.562 ± 0.078	0.593 ± 0.095	**0.635 ± 0.082**	306/3
banknote	0.875 ± 0.032	0.932 ± 0.026	**0.934 ± 0.027**	1372/4
fertility	0.534 ± 0.235	0.556 ± 0.226	**0.612 ± 0.332**	100/9
parkinsons	0.790 ± 0.084	0.782 ± 0.101	**0.831 ± 0.066**	195/22
climate	0.598 ± 0.098	**0.731 ± 0.114**	0.730 ± 0.097	540/18
ILPD	0.573 ± 0.057	0.600 ± 0.063	**0.660 ± 0.04**	579/10
german	0.604 ± 0.048	**0.658 ± 0.068**	0.647 ± 0.057	1000/24
heart	0.748 ± 0.136	0.800 ± 0.090	**0.806 ± 0.068**	270/13
golub	**0.769 ± 0.156**	0.747 ± 0.121	0.738 ± 0.141	72/50
Média do Rank $R(\mathcal{L})$	2.4737	1.8947	**1.6316**	

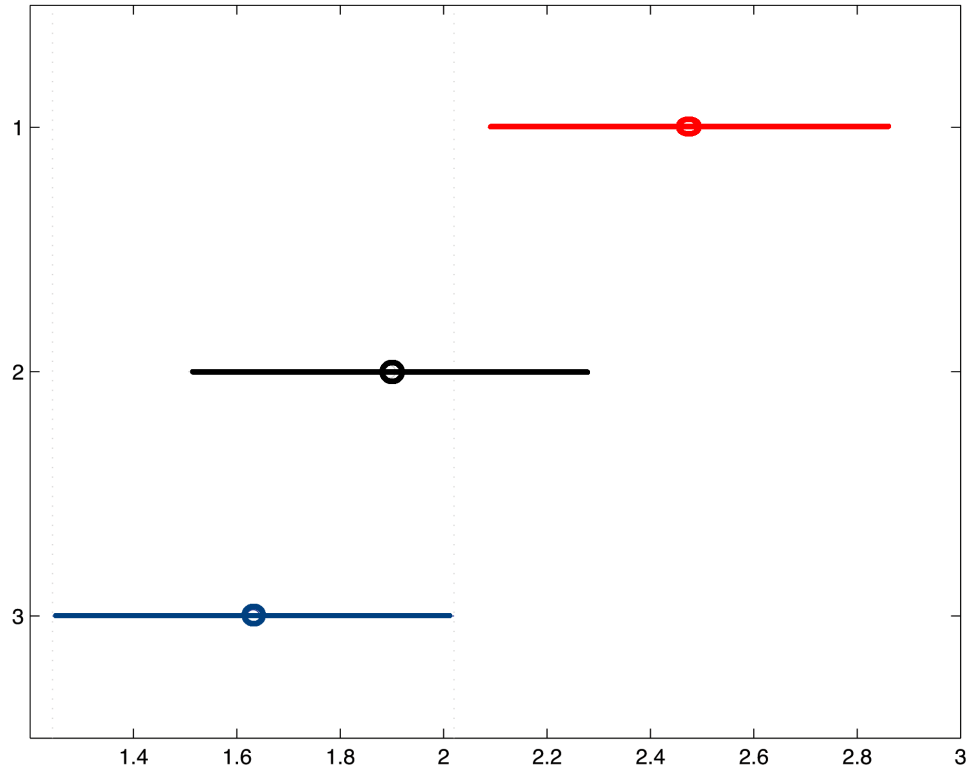

Figura 5.3: Comparação entre os três modelos de classificadores através do ranqueamento do teste de *Friedman* e comparação pelo método *Bonferroni*. Em vermelho: classificador sem filtragem. Em preto: classificador com filtragem baseada no grau do vértice. Em azul: classificador com filtragem baseada em $D_C(\cdot)$.

Capítulo 6

Conclusões e Trabalhos Futuros

O trabalho aqui apresentado investiga como as informações extraídas do grafo de Gabriel podem ser utilizadas para a regularização de classificadores. Trabalhos anteriores mostram que a utilização do grafo de Gabriel é promissora para estimar os parâmetros da primeira camada de uma rede RBF (Torres *et al.*, 2013) e para o desenvolvimento de um classificador (Torres *et al.*, 2014). Estes modelos são chamados, respectivamente, de CG-RBF e CHIP-CLASS. São apresentadas técnicas para utilizar o grafo de Gabriel para regularização de ambos.

Primeiro, é feita uma investigação das características que levam à generalização em redes neurais RBF usando informações extraídas do grafo de Gabriel. Um conjunto especial de amostras é extraído do grafo e chamado de Vetores de Suporte Estruturais (VSE). A representação é utilizada não apenas para estimar os parâmetros das funções radiais da RBF, como também para eliminar funções radiais, promovendo regularização. São investigadas tanto a capacidade dos VSE de indicação de ruído como a intuição inversa: de que os VSE indicam margem de separação. Além da regularização via eliminação de funções radiais específicas, a estratégia sugerida é comparada à regularização de Tikhonov, que penaliza a magnitude do vetor de pesos da camada de saída.

Era esperado que o conjunto VSE indicasse ruído para classes sobrepostas e margem para classes separadas. Isso, de fato, ocorre, uma vez que a retirada de funções radiais associadas ao VSE tem melhor desempenho em base de dados separadas e pior em base de dados sobrepostas. No entanto, a representatividade de margem do VSE é dominante: a regularização que considera essa premissa

é mais bem comportada. Seu desempenho é muito superior para classes bem separadas, como esperado, mas para classes sobrepostas seu desempenho está próximo da intuição oposta.

A comparação com a regularização de Tikhonov revelou uma característica interessante do VSE. Para classes sobrepostas, o conjunto de vértices pode ser considerado ruído pois a penalização dos pesos da camada de saída, o parâmetro λ, permanece estável. Em contraste, λ cresce rapidamente com a sobreposição quando os vértices do VSE não são retirados.

A investigação do conjunto especial de vértices VSE da primeira parte do trabalho serve de base para os estudos da segunda parte. Apesar da estrutura ser um excelente candidato para extrair informações da margem de separação dos dados, a melhor escolha desse conjunto é sensível à sobreposicão de dados. Para maior robustez do classificador baseado no grafo de Gabriel, o CHIP-CLASS, é preciso uma técnica de filtragem em casos de sobreposição.

Na segunda parte, o trabalho apresenta uma abordagem para regularização do CHIP-CLASS através de uma métrica de grafo capaz de detectar e eliminar ruído, respeitando as características do classificador: não utilização de parâmetros ou algoritmos de otimização.

A métrica de grafo é definida a partir da distância geodésica relativa entre diferentes classes. Essa métrica é utilizada aqui para extrair informações do conjunto de dados para identificação de amostras da margem de separação e, assim, identificar e eliminar amostras que causam a sobreposição.

Os experimentos mostram duas características dessa métrica de grafo. A primeira é que ela pode apresentar distribuição normal para base de dados reais. Das 19 diferentes bases de dados testadas, 11 apresentaram essa característica, o que representa uma quantidade expressiva. A segunda é que a métrica de grafo sugerida, Eq. (4.4), realmente identifica amostras pertencentes à margem de separação. Como é mostrado no trabalho, essa métrica pode ser utilizada na filtragem, retirando as amostras que apresentam os menores valores da métrica, sendo o único filtro estatisticamente mais eficiente que o classificador sem filtro, dentre os testados.

O trabalho contribui para o avanço dos estudos da utilização do grafo de Gabriel no aprendizado supervisionado.

6.1 Propostas de Continuidade

Seguem algumas propostas de continuidade para este trabalho:

- Formalizar a filtragem proposta como regularização do classificador baseado no grafo de Gabriel;
- Relacionar as características de bases reais - como número de amostras, atributos, desbalanceamento, natureza - com o desempenho do classificador para investigar suas propriedades;
- Explorar diferentes configurações de $\mathcal{C}$ e $f(\cdot)$ para métrica $D_C(\cdot)$.

Referências

Alcalá-Fdez, J., Fernández, A., Luengo, J., Derrac, J., García, S., Sánchez, L. & Herrera, F. (2011). Keel data-mining software tool: data set repository, integration of algorithms and experimental analysis framework. *Journal of Multiple-Valued Logic & Soft Computing*, **17**. 46, 49

Bonacich, P. (1972). Factoring and weighting approaches to status scores and clique identification. *Journal of Mathematical Sociology*, **2**, 113–120. 33

Bondy, J.A. & Murty, U.S.R. (1976). *Graph theory with applications*, vol. 290. Macmillan London. 8

Boser, B.E., Guyon, I.M. & Vapnik, V.N. (1992). A training algorithm for optimal margin classifiers. In *Proceedings of the fifth annual workshop on Computational learning theory*, 144–152, ACM. 1, 2, 14, 15, 16, 29

Bouchired, S., Ibnkahla, M., Roviras, D. & Castanie, F. (1998). 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, 3377–3379, IEEE. 21

Brent, R.P. (1971). An algorithm with guaranteed convergence for finding a zero of a function. *The Computer Journal*, **14**, 422–425. 25

Buckley, F. & Harary, F. (1990). *Distance in graphs*. Addison-Wesley. 3, 33

Castro, C.L. & Braga, A.P. (2013). Novel cost-sensitive approach to improve the multilayer perceptron performance on imbalanced data. *IEEE transactions on neural networks and learning systems*, **24**, 888–899. 46, 49

Corman, S.R., Kuhn, T., McPhee, R.D. & Dooley, K.J. (2002). Studying complex discursive systems. *Human communication research*, **28**, 157–206. 33

De Berg, M., van Kreveld, M., Overmars, M. & Schwarzkopf, O. (2000). Computational geometry: Algorithms and applications springer-verlag. 11

Demšar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *Journal of Machine learning research*, **7**, 1–30. 52

Euler, L. (1736). Solutio problematis ad geometriam situs pertinentis. *Commentarii Academiae Scientiarum Imperialis Petropolitanae*, **8**, 128–140. 8

Everett, M.G. & Borgatti, S.P. (2005). Extending centrality. *Models and methods in social network analysis*, **35**, 57–76. 3, 33, 34

Fernandez-Delgado, M., Ribeiro, J., Cernadas, E. & Ameneiro, S.B. (2011). Direct parallel perceptrons (dpps): fast analytical calculation of the parallel perceptrons weights with margin control for classification tasks. *IEEE transactions on neural networks*, **22**, 1837–1848. 14

Freeman, L.C., Roeder, D. & Mulholland, R.R. (1979). Centrality in social networks: Ii. experimental results. *Social networks*, **2**, 119–141. 33

Freund, Y., Schapire, R.E. *et al.* (1996). Experiments with a new boosting algorithm. In *Icml*, vol. 96, 148–156. 14

Gabriel, K.R. & Sokal, R.R. (1969). A new statistical approach to geographic variation analysis. *Systematic zoology*, **18**, 259–278. 1

Golub, T.R., Slonim, D.K., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J.P., Coller, H., Loh, M.L., Downing, J.R., Caligiuri, M.A. *et al.* (1999). Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *science*, **286**, 531–537. 49

Haykin, S. (1994). *Neural networks: a comprehensive foundation*. Prentice Hall PTR. 21

Hess, K.R., Anderson, K., Symmans, W.F., Valero, V., Ibrahim, N., Mejia, J.A., Booser, D., Theriault, R.L., Buzdar, A.U., Dempsey, P.J. *et al.* (2006). Pharmacogenomic predictor of sensitivity to preoperative chemotherapy with paclitaxel and fluorouracil, doxorubicin, and cyclophosphamide in breast cancer. *Journal of clinical oncology*, **24**, 4236–4244. 46, 49

James, G., Witten, D., Hastie, T. & Tibshirani, R. (2013). *An introduction to statistical learning*, vol. 112. Springer. 7, 8, 30

Johnson, S.C. (1967). Hierarchical clustering schemes. *Psychometrika*, **32**, 241–254. 34

Lichman, M. (2013). UCI machine learning repository. 46, 49

McElreath, R. (2016). *Statistical Rethinking: A Bayesian Course with Examples in R and Stan*, vol. 122. CRC Press. 2

Ng, A.Y. (2004). Feature selection, l 1 vs. l 2 regularization, and rotational invariance. In *Proceedings of the twenty-first international conference on Machine learning*, 78, ACM. 7

Sha, F. & Saul, L.K. (2006). Large margin gaussian mixture modeling for phonetic classification and recognition. In *Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings. 2006 IEEE International Conference on*, vol. 1, I–I, IEEE. 14

Sing, J., Basu, D., Nasipuri, M. & Kundu, M. (2003). Improved k-means algorithm in the design of rbf neural networks. In *TENCON 2003. Conference on Convergent Technologies for the Asia-Pacific Region*, vol. 2, 841–845, IEEE. 21

Smola, A.J. (2000). *Advances in large margin classifiers*. MIT press. 1

Torres, L. (2016). *Classificador por arestas de suporte(CLAS): Métodos de aprendizado baseados em grafos de Gabriel*. Tese de doutorado, UFMG. 3, 13, 29, 31, 49, 52

Torres, L., Castro, C., Coelho, F., Torres, F.S. & Braga, A. (2015a). Distance-based large margin classifier suitable for integrated circuit implementation. *Electronics Letters*, **51**, 1967–1969. 1, 13, 22

Torres, L.C., Lemos, A.P., Castro, C.L. & Braga, A.P. (2013). Projeto de redes rbf baseado na estrutura dos dados e em informações de margem. In *Computational Intelligence, 2013 1st BRICS Countries Congress(BRICS-CCI) and 11th Brazilian Congress (CBIC) on*, 1–7. 2, 21, 55

Torres, L.C., Coelho, F., Castro, C.L. & Braga, A.P. (2014). A graph of gabriel approach for large margin classifiers. LA-CCI-The Latin American Congress on Computational Intelligence Co-located with ARGENCON. 1, 55

Torres, L.C., Castro, C.L. & Braga, A.P. (2015b). A parameterless mixture model for large margin classification. In *Neural Networks (IJCNN), 2015 International Joint Conference on*, 1–6, IEEE. 2, 24, 27

Wilcox, R. (2005). Kolmogorov–smirnov test. *Encyclopedia of biostatistics*. 38, 50

Zhang, W. & King, I. (2002). 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, 239–244, IEEE. 3, 11, 13