EXTREME LEARNING MACHINES REGULARIZADAS DE FORMA AUTOMÁTICA A PARTIR DAS INFORMAÇÕES ESTRUTURAIS DA MATRIZ DE PROJEÇÃO

Lourenço R. G. Araújo* Luiz C. B. Torres* Leonardo J. Silvestre† Antônio P. Braga*

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

†Departamento de Computação e Eletrônica - Centro Universitário Norte do Espírito Santo - Universidade Federal do Espírito Santo
Rodovia Governador Mário Covas, Km 60, CEP. 29932-540, São Mateus, ES, Brasil

Email: Lrgaraujo@gmail.com luizlitc@gmail.com leonardo.silvestre@ufes.br apbraga@ufmg.br

Abstract— The following work presents a new approach to automatic selection of Tikhonov's regularization parameter, responsible for controlling the weight value of an ELM neural network. Two strategies are presented, which are based on measurements obtained from data projection: Silhouette (a cluster evaluation metric) and Fisher-score (a feature selection measure). Seven datasets are tested and results are compared to those obtained when the regularization parameter is selected through cross-validation. Both strategies show satisfactory classification performance (in terms of p-value), while presenting significant training time reduction.

Keywords— Extreme Learning Machines, ELM, Regularization, Fisher-Score, Silhouette

Resumo— Neste artigo apresenta-se uma nova abordagem para a seleção automática do parâmetro de regularização de Tikhonov, responsável por controlar o tamanho dos pesos de uma rede neural do tipo ELM. São apresentadas duas estratégias baseadas em medidas obtidas da projeção dos dados: Silhueta (uma medida de qualidade de agrupamentos) e Fisher-score (um critério para seleção de características). Sete bases de dados apresentando problemas de classificação binária são testadas e os resultados obtidos são comparados com o resultado obtido quando se seleciona o parâmetro de regularização por validação cruzada. As duas estratégias propostas mostraram um desempenho de classificação satisfatório (em termos de p-valor) e um ganho de tempo significativo, quando comparadas à estratégia iterativa de seleção do parâmetro de regularização.

Palavras-chave— Máquinas de Aprendizado Extremo, ELM, Regularização, Fisher-Score, Silhueta

1 Introdução

Embora eficientes para solução de problemas de aprendizado de máquinas, as Máquinas de Aprendizado Extremo (do inglês, ELMs) perdem a capacidade de generalização com relativa facilidade, problema conhecido como *overfitting* (Huang et al., 2004), e, portanto, beneficiam-se de estratégias de regularização. Por serem redes do tipo SLFN (*Single Layer Feedforward Network*)(Haykin, 1994), uma solução usual é a aplicação de uma regularização com norma L_2(Deng et al., 2009), também conhecida como regularização de Tikhonov (Tikhonov, 1963). A maioria das estratégias de regularização, no entanto, envolve o ajuste de parâmetros por métodos iterativos, o que acaba por comprometer a velocidade de execução do modelo. No trabalho de Huang et al. (2012), é apresentada uma metodologia de regularização que envolve a seleção, por validação cruzada, de um parâmetro de regularização com norma L_2. Já a estratégia conhecida como **OP-ELM**, proposta por Miche et al. (2010), envolve uma penalização com norma L_1, que tem como efeito a poda da rede. Finalmente, a estratégia conhecida como **TROP-ELM** (Miche et al., 2011), envolve a aplicação, inicialmente de uma penalização com norma L_1, seguida de uma penalização com norma L_2.

A introdução da etapa de validação cruzada para seleção de parâmetros de regularização na construção das ELMs acaba por aumentar o tempo necessário para o treinamento, o que mina uma das principais vantagens do modelo, que é justamente o tempo reduzido (Silvestre et al., 2015).

No trabalho de Silvestre et al. (2015), utiliza-se a informação espacial *a priori*, formalizada como uma matriz de afinidade, para regularização de ELMs. É provado que a utilização da matriz de afinidade é similiar à regularização de Tikhonov, e, quando utilizada uma matriz de afinidade independente de parâmetros, não é necessário o ajuste de nenhum parâmetro.

No presente artigo, investiga-se a possibilidade de, a partir da estrutura dos dados, determinar, de forma automática, uma maneira de se regularizar a rede ELM construída. Diferentemente da metodologia apresentada por Silvestre et al. (2015), explora-se, neste trabalho, a estrutura dos dados projetados, valendo-se da ideia de separabilidade linear, uma das consequências do Teorema de Cover (Cover, 1965). Foi realizada uma pesquisa exploratória em que a metodologia apresentada por Deng et al. (2009) é utilizada como controle. Resultados preliminares com bases de dados reais mostraram que, para os métodos propostos, há um ganho de velocidade significativo na seleção

da estrutura de regularização, sem perdas consideráveis no desempenho de classificação.

O restante do artigo é apresentado da seguinte forma: na Seção 2, são apresentados os fundamentos das *Extreme Learning Machines*, bem como uma breve revisão das estratégias mais comuns de regularização. Em seguida na Seção 3, apresenta-se o método proposto e as justificativas para sua utilização. Na Seção 4, é descrita a metodologia adotada na condução dos experimentos e apresentam-se os resultados obtidos. Finalmente, a Seção 5 traz as discussões e conclusões.

2 Extreme Learning Machines

As *Extreme Learning Machines* são um tipo de SLFN (*Single Layer Feedforward Network*) que apresenta a propriedade de aproximação universal (Huang et al., 2004).

De modo geral, SLFNs são redes compostas por p neurônios em uma única camada oculta (Haykin, 1994). A i-ésima observação será modelada por:

$$\hat{y}_i = \sum_{j=1}^{p} w_j g(\mathbf{x_i}) = \sum_{j=1}^{p} w_j g(\mathbf{v_j x_i} + b_j) \quad (1)$$

em que p é o número de neurônios na camada escondida, $g(\cdot)$ é a função de ativação, o w_j é o peso ligando o j-ésimo neurônio à saída, $\mathbf{v_j}$ é o vetor de pesos ligando as entradas ao neurônio j e b_j é o termo de *bias* do j-ésimo neurônio.

Define-se, em seguida, a matriz H:

$$\mathbf{H}=\begin{bmatrix} g(\mathbf{v_1}\cdot\mathbf{x_1}+b_1) & \dots & g(\mathbf{v_p}\cdot\mathbf{x_1}+b_p) \\ \vdots & \dots & \vdots \\ g(\mathbf{v_1}\cdot\mathbf{x_N}+b_1) & \dots & g(\mathbf{v_p}\cdot\mathbf{x_N}+b_p) \end{bmatrix}. \quad (2)$$

A Equação 1 na forma matricial é reescrita como:

$$\mathbf{HW} = \hat{\mathbf{Y}}. \quad (3)$$

Para que a rede seja capaz de aproximar as N observações, é necessário que:

$$\mathbf{HW} = \hat{\mathbf{Y}} = \mathbf{Y}. \quad (4)$$

O algoritmo de *Extreme Learning Machine* propõe que os pesos $\mathbf{v}_j$ e os valores de bias b_j sejam inicializados de forma aleatória para $j = 1, ..., p$. Torna-se desnecessário o aprendizado dos pesos por propagação reversa dos erros, o que dispensa a utilização de algoritmos de otimização não linear, que implicam em um consumo elevado de tempo (Miche et al., 2011).

Para que a rede seja capaz de aproximar bem os dados, é necessário um número elevado de neurônios. Desde que atendidas algumas condições com relação à inicialização dos pesos e às funções de ativação utilizadas, conforme mostrado por Huang, as ELMs se comportam como aproximadores universais (Huang et al., 2006).

A matriz de pesos pode ser obtida, então, a partir da Equação 5:

$$\mathbf{W} = \mathbf{H}^{+}\mathbf{Y} \quad (5)$$

em que $\mathbf{H}^{+}$ é a pseudoinversa da matriz $\mathbf{H}$.

De acordo com Huang et al. (2004), a obtenção de $\mathbf{W}$ como apresentado na Equação 5 leva à minimização de erro de treinamento (trata-se de uma das soluções do problema de mínimos quadrados dado por $\|\mathbf{HW} - Y\|$).

Um algoritmo para o treinamento de ELMs é fornecido por Huang et al. (2004) e é descrito a seguir:

Algoritmo 1: Algoritmo para treinamento de ELMs

Entrada: $(\mathbf{x_i}, \mathbf{t_i})|\mathbf{x_i} \in \mathbf{R^n}, \mathbf{t_i} \in \mathbf{R^m}, i = 1, ..., N$; funções de ativação $g(x)$; p neurônios na camada oculta

Saída: Número esperado de nodos atingidos

início

Atribuir valores arbitrários a $\mathbf{v}_j$ e b_j para $j = 1, ..., p$;

Calcular o valor da matriz $\mathbf{H}$

Calcular o vetor W:

$$\mathbf{W} = \mathbf{H}^{+}\mathbf{Y}$$

em que $\mathbf{H}^{+}$ é a pseudoinversa de $\mathbf{H}$

fim

retorna $\sigma(S)$

Uma das principais vantagens do algoritmo proposto é a sua alta velocidade de treinamento.

As ELMs, no entanto, apresentam tendência ao *overfitting* (Deng et al., 2009), o que pode ser explicado pela etapa de treinamento, que se dá pela minimização do risco empírico (Vapnik, 2013), sem considerar-se o risco estrutural, como pode ser visto na Equação 5.

No trabalho de Silvestre et al. (2015), é apresentada uma revisão das diferentes estratégias de regularização de ELMs, bem como uma proposta de regularização original dos autores.

2.1 Regularização de ELMs

A regularização de ELMs deve ser interpretada como uma etapa de minimização do risco estrutural (Vapnik, 2013), que impede o crescimento excessivo da rede (controlando ou a quantidade de neurônios, ou o módulo dos pesos).

As estratégias de regularização mais comuns envolvem tanto uma penalização por norma L_2 (regularização de Tikhonov), como apresentada por Deng et al. (2009) e Huang et al. (2012), quanto a seleção de neurônios mais importantes, graças a uma penalização com norma L_1, como os métodos **OP-ELM** (Miche et al., 2010) e **TROP-ELM** (Miche et al., 2011), sendo que o último envolve a aplicação de ambas as penalidades: inicialmente aplica-se uma penalidade com norma L_1, seguida de uma penalização com norma L_2.

Processos de regularização baseados em penalização com norma L_2 (também conhecidos como

regularização de Tikhonov) levam a uma diminuição no módulo dos pesos como um todo, o que proporciona maior suavidade à superfície de separação. Já a aplicação de penalidade com norma L_1 leva à seleção dos neurônios mais importantes, com os pesos dos neurônios menos importantes tendendo a zero (James et al., 2013).

Em comum, os métodos citados envolvem o ajuste de parâmetros de regularização, o que leva a um aumento no tempo computacional necessário para resolução dos problemas.

No trabalho de Silvestre et al. (2015), é proposto um algoritmo de treinamento de ELMs livre de parâmetros. Os autores propõem uma estratégia de regularização baseada em matrizes de afinidade, de modo que a matriz $\mathbf{H}$ é transformada para levar em consideração uma certa quantidade de informação *a priori* a respeito da estrutura dos dados. Os autores mostram que o efeito obtido é equivalente a uma regularização de Tikhonov.

3 O método proposto

Para o método proposto, utiliza-se a mesma equação apresentada por Deng por Deng et al. (2009) para cálculo da matriz de pesos, $\mathbf{W}$.

$$\mathbf{W} = (\mathbf{H}^T\mathbf{H} + \mathbf{\Lambda})^{-1}\mathbf{H}^T\mathbf{Y} \tag{6}$$

Como mostrado por Deng et al. (2009), a matriz $\mathbf{\Lambda}$ pondera o compromisso ótimo entre risco estrutural e risco empírico a ser selecionado. O caso particular que se considera para o presente trabalho consiste em $\Lambda = \lambda\mathbf{I}$. Logo, o valor dos pesos passa a ser dado pela Equação 7:

$$\mathbf{W} = (\lambda\mathbf{I} + \mathbf{H}^T\mathbf{H})^{-1}\mathbf{H}^T\mathbf{Y}. \tag{7}$$

O método proposto leva em consideração a importância da estrutura dos dados apresentada por Silvestre et al. (2015), combinada com o Teorema de Cover, e visa realizar uma seleção automática do parâmetro λ de regularização.

O ponto que se extrai do Teorema de Cover e que se explora neste trabalho diz respeito à possibilidade de separar linearmente um problema originalmente não separável linearmente desde que o espaço de entrada seja mapeado de forma não linear para um espaço de dimensão suficientemente grande (Cover, 1965).

Investiga-se, a partir da estrutura dos dados projetados (matriz $\mathbf{H}_{n\times p}$), se é possível extrair algum tipo de informação a respeito da separabilidade linear dos dados que possa ser utilizado como parâmetro de regularização λ. É esperado que, quanto mais separáveis sejam os dados projetados, maior deve ser o valor do parâmetro de regularização.

Foram investigadas, em um primeiro momento, duas medidas que poderiam, a princípio, ser indicativas da separabilidade linear dos dados: silhueta (Kaufman and Rousseeuw, 2009) e *Fisher score* (Duda et al., 1973).

3.1 Silhueta

O conceito de Silhueta (Kaufman and Rousseeuw, 2009) foi desenvolvido para técnicas de *clustering* e fornece uma medida da capacidade que os agrupamentos propostos teriam para separar de forma coerente os dados.

O valor de Silhueta para um dado grupo (classe) i pode ser obtido a partir da Equação (8):

$$s(i) = \frac{b(i) - a(i)}{max(b(i), a(i))} \tag{8}$$

em que i indica o índice dos grupos $a(i)$ é a distância média de intragrupo e $b(i)$ é a mínima distância média intergrupos.

Nota-se que o valor de Silhueta pode assumir valores entre $-1 \leq s(i) \leq 1$. É esperado que, quanto maior a separabilidade linear, maior será o valor de Silhueta, e vice-versa.

3.2 Fisher Score

A ideia motivadora do *Fisher score* (Duda et al., 1973) é encontrar, em um problema de seleção de características, a combinação de características que maximiza a distância interclasse e minimiza a distância intraclasse. Uma heurística comum consiste em analisar cada variável de forma independente: para cada uma das dimensões do problema, calcula-se a distância entre as médias, ponderada pela dispersão dos dados (variância).

$$F(x^j) = \frac{dist(\mu_+^j, \mu_-^j)}{(\sigma_+^j)^2 + (\sigma_-^j)^2} \tag{9}$$

em que μ_k^j e $(\sigma_k^j)^2$ são respectivamente a média e a variância da classe k ao longo da característica j. Quanto maior o valor de $F(x^j)$, maior a importância da dimensão avaliada.

3.3 Cálculo de λ

A partir dos valores calculados para a silhueta e para o critério de Fisher, foi necessário estabelecer algum tipo de relação entre os mesmos e um valor de λ adequado a ser utilizado na Equação 7 para o problema de classificação. Utilizou-se a média aritmética, tanto para o valor da silhueta quanto para o critério de Fisher.

Como a silhueta pode apresentar valores negativos, é proposta neste trabalho a seguinte função, obtida empiricamente após a realização de diversos testes em bases de dados sintéticas:

$$\lambda(\mathcal{S}) = \begin{cases} 0, \; se\, \mathcal{S} \leq 0 \\ \frac{-1}{10log(\mathcal{S})}, \text{caso contrário.} \end{cases} \tag{10}$$

onde $\mathcal{S}$ é o valor médio de silhueta.

Já o valor de λ baseado no critério de Fisher ($\mathcal{F}$) foi definido como a média aritmética dos valores obtidos de $\mathcal{F}$ para cada uma das dimensões da matriz $\mathbf{H}$, conforme mostrado na Equação 11.

$$\lambda = mean(\mathcal{F}). \tag{11}$$

4 Resultados

A Figura 1 mostra o efeito da regularização aplicada utilizando os valores calculados sobre a superfície de separação obtida com uma ELM de 1000 neurônios na camada escondida.

Os dados de duas classes foram gerados sinteticamente a partir de duas distribuições normais com mesma variância.

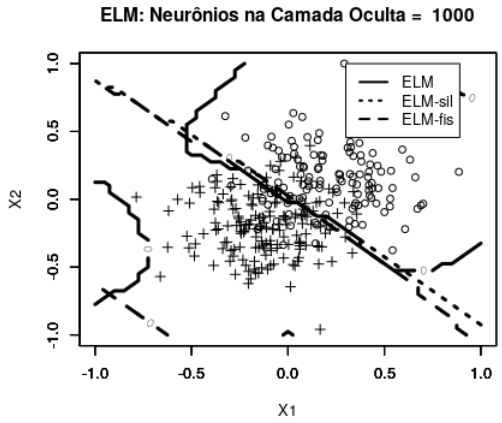

Figura 1: Superfícies de separação obtidas com ELM de 1000 neurônios na camada escondida: sem regularização (ELM) e com as regularizações baseadas no Critério de Fisher (ELM-fis) e na Silhueta (ELM-sil)

Observa-se na Figura 1 que os valores propostos de λ levam a uma suavização das superfícies de decisão. Sabe-se que, para duas distribuições normais, com mesma variância, a superfície ótima de separação é dada por um hiperplano entre as classes. As duas ELMs regularizadas, com os valores propostos de λ foram capazes de se aproximar melhor da superfície ótima.

4.1 Bases de dados reais

Foram realizados testes envolvendo sete das dez bases de dados testadas por Silvestre et al. (2015), e compararam-se os resultados obtidos quando o parâmetro de regularização λ foi selecionado por validação cruzada, **ELM-reg**, com os resultados obtidos quando λ foi selecionado baseado na silhueta, **ELM-sil**, e quando λ foi selecionado baseado no critério de Fisher, **ELM-fis**. Não foram testadas todas as bases utilizadas por Silvestre et al. (2015) pois três dentre as dez bases são originariamente não binárias e optou-se por não realizar a binarização das bases.

Para o método **ELM-reg**, o parâmetro λ foi selecionado por validação cruzada do tipo $10 - fold$, conforme a metodologia apresentada por Huang et al. (2012).

Para cada uma das sete bases de dados foram realizadas 30 repetições, com divisão entre conjunto de teste e treinamento na proporção 70% para treinamento e 30% para teste. Para cada uma das bases, foram treinadas ELMs com 10, 30, 100, 500 e 1000 neurônios na camada escondida.

As bases de dados utilizadas na condução dos experimentos foram obtidas no repositório UCI (Dheeru and Karra Taniskidou, 2017), são elas: *Australin Credit* (acr), *QSAR biodegradation* (bio), *BUPA Liver Disorders* (bld), Statlog Heart (hea), *Pima Indian Diabetes* (pid), *Congressional Voting Records Data Set* (vot) e *Wisconsin Breast Cancer* (wbc).

As características de cada uma das bases são apresentadas na Tabela 1.

Tabela 1: Bases de dados Estudadas

Base de Dados	Amostras	Atributos
acr	690	14
bio	1055	41
bld	345	6
hea	270	13
pid	768	8
vot	435	16
wbc	683	10

Todos os testes apresentados neste trabalho foram executados em um *notebook* Dell Inspiron, com sistema operacional Debian, processador Intel Core i7 4510U, com 8GB de memória RAM. Para o método **ELM-reg**, os valores de λ foram selecionados em $\{2^{-25}, 2^{-24}, ..., 2^{23}, 2^{24}\}$. Todos os testes foram realizados apenas com o *Desktop Environment* e o software RStudio abertos. Foi descartada a pior medida de tempo para cada uma das abordagens.

Os testes foram todos realizados utilizando-se a linguagem de programação R (R Core Team, 2015).

As Tabelas 2 e 3 apresentam os resultados obtidos para as duas estratégias testadas de obtenção de λ comparados aos resultados obtidos quando se seleciona o valor de λ por validação cruzada (**ELM-reg**). A Tabela 2 apresenta os resultados com relação à acurácia de classificação e a Tabela 3 apresenta os resultados obtidos para o tempo de execução do processo de treinamento. Os maiores valores de acurácia e os menores valores de tempo estão destacados em negrito.

Foram realizados testes estatísticos comparando o desempenho (acurácia de teste e tempo de execução) dos métodos propostos (**ELM-sil** e

Tabela 2: Acurácia de teste ($\%media \pm \sigma$)

p	ELM-reg	ELM-sil	ELM-fis
acr			
10	$\mathbf{84.7 \pm 2.7}$	83.7 ± 2.7	84.4 ± 1.8
30	85.7 ± 2.0	$\mathbf{86.4 \pm 2.2}$	$\mathbf{86.4 \pm 2.0}$
100	85.5 ± 2.7	86.0 ± 2.2	$\mathbf{86.6 \pm 2.2}$
500	86.2 ± 2.1	85.6 ± 2.1	$\mathbf{86.5 \pm 1.9}$
1000	$\mathbf{86.8 \pm 1.9}$	85.4 ± 1.5	85.5 ± 1.9
bio			
10	74.9 ± 3.6	74.8 ± 4.1	$\mathbf{75.8 \pm 4.4}$
30	$\mathbf{84.0 \pm 2.0}$	83.5 ± 2.1	83.7 ± 2.6
100	84.6 ± 1.8	85.7 ± 1.7	$\mathbf{85.7 \pm 1.6}$
500	85.7 ± 1.5	85.4 ± 2.4	$\mathbf{87.1 \pm 1.4}$
1000	*	84.9 ± 3.8	$\mathbf{86.7 \pm 1.9}$
bld			
10	$\mathbf{71.7 \pm 4.6}$	67.6 ± 3.5	67.4 ± 4.3
30	$\mathbf{71.3 \pm 3.9}$	68.8 ± 4.6	68.4 ± 3.6
100	$\mathbf{72.3 \pm 3.7}$	71.8 ± 5.5	70.3 ± 3.2
500	72.4 ± 4.1	$\mathbf{73.8 \pm 3.6}$	73.7 ± 3.7
1000	$\mathbf{72.4 \pm 3.9}$	70.8 ± 3.9	71.4 ± 3.3
hea			
10	$\mathbf{81.7 \pm 4.0}$	79.2 ± 5.8	77.9 ± 4.6
30	$\mathbf{83.7 \pm 3.4}$	82.5 ± 2.5	83.3 ± 2.9
100	$\mathbf{83.3 \pm 3.9}$	82.6 ± 4.0	82.4 ± 3.1
500	$\mathbf{83.7 \pm 3.0}$	81.0 ± 3.8	82.7 ± 3.6
1000	$\mathbf{83.7 \pm 3.9}$	81.1 ± 4.5	83.3 ± 3.3
pid			
10	$\mathbf{76.9 \pm 2.5}$	76.4 ± 2.4	75.9 ± 2.7
30	76.8 ± 2.4	76.9 ± 2.0	$\mathbf{77.7 \pm 2.2}$
100	75.6 ± 2.4	76.1 ± 2.1	$\mathbf{77.3 \pm 2.3}$
500	*	76.7 ± 2.0	$\mathbf{77.2 \pm 2.4}$
1000	76.5 ± 2.5	$\mathbf{76.7 \pm 2.6}$	76.6 ± 2.2
vot			
10	89.5 ± 3.2	88.3 ± 3.2	$\mathbf{90.6 \pm 3.2}$
30	$\mathbf{95.2 \pm 2.0}$	94.5 ± 1.5	94.4 ± 1.5
100	93.9 ± 1.9	94.4 ± 1.8	$\mathbf{95.2 \pm 1.5}$
500	$\mathbf{95.7 \pm 1.5}$	92.7 ± 2.2	94.9 ± 1.3
1000	$\mathbf{95.4 \pm 1.6}$	93.5 ± 1.6	94.1 ± 1.9
wbc			
10	$\mathbf{95.7 \pm 1.5}$	95.3 ± 1.4	94.6 ± 1.9
30	$\mathbf{96.6 \pm 1.2}$	96.3 ± 1.0	96.0 ± 1.2
100	96.3 ± 1.1	$\mathbf{96.5 \pm 1.0}$	96.3 ± 1.1
500	96.4 ± 0.9	96.0 ± 1.0	$\mathbf{96.5 \pm 1.1}$
1000	96.4 ± 1.1	96.2 ± 1.0	$\mathbf{96.5 \pm 1.2}$

Tabela 3: Tempo de treinamento (em segundos)

p	ELM-reg	ELM-sil	ELM-fis
acr			
10	2.06 ± 0.17	0.006 ± 0.002	$\mathbf{0.001 \pm 0.000}$
30	3.03 ± 0.08	0.011 ± 0.002	$\mathbf{0.004 \pm 0.000}$
100	12.39 ± 0.65	0.046 ± 0.004	$\mathbf{0.023 \pm 0.002}$
500	365.28 ± 12.64	0.820 ± 0.016	$\mathbf{0.68 \pm 0.016}$
1000	2515.78 ± 33.50	4.957 ± 0.061	$\mathbf{4.69 \pm 0.040}$
bio			
10	2.55 ± 0.39	0.015 ± 0.006	$\mathbf{0.002 \pm 0.000}$
30	4.11 ± 0.33	0.023 ± 0.005	$\mathbf{0.007 \pm 0.000}$
100	16.44 ± 0.73	0.086 ± 0.005	$\mathbf{0.035 \pm 0.002}$
500	431.44 ± 3.07	1.091 ± 0.019	$\mathbf{0.805 \pm 0.020}$
1000	*	7.861 ± 0.043	$\mathbf{5.385 \pm 0.026}$
bld			
10	1.83 ± 0.35	0.002 ± 0.000	$\mathbf{0.001 \pm 0.000}$
30	2.61 ± 0.42	$\mathbf{0.003 \pm 0.000}$	0.004 ± 0.002
100	8.89 ± 1.11	$\mathbf{0.017 \pm 0.001}$	0.019 ± 0.004
500	277.67 ± 4.41	0.549 ± 0.017	$\mathbf{0.546 \pm 0.011}$
1000	2135.44 ± 10.03	$\mathbf{4.110 \pm 0.261}$	4.223 ± 0.039
hea			
10	1.78 ± 0.26	$\mathbf{0.001 \pm 0.000}$	$\mathbf{0.001 \pm 0.000}$
30	2.28 ± 0.36	$\mathbf{0.003 \pm 0.000}$	$\mathbf{0.003 \pm 0.001}$
100	8.06 ± 0.39	$\mathbf{0.015 \pm 0.001}$	0.017 ± 0.001
500	269.89 ± 4.85	$\mathbf{0.512 \pm 0.007}$	0.518 ± 0.013
1000	2119.78 ± 32.10	$\mathbf{4.075 \pm 0.013}$	4.084 ± 0.024
pid			
10	2.03 ± 0.08	0.005 ± 0.000	$\mathbf{0.002 \pm 0.000}$
30	2.89 ± 0.22	0.011 ± 0.001	$\mathbf{0.005 \pm 0.000}$
100	12.89 ± 0.70	0.055 ± 0.008	$\mathbf{0.025 \pm 0.001}$
500	*	0.820 ± 0.010	$\mathbf{0.688 \pm 0.018}$
1000	2545.00 ± 27.96	5.319 ± 0.063	$\mathbf{4.804 \pm 0.019}$
vot			
10	1.94 ± 0.17	0.004 ± 0.004	$\mathbf{0.001 \pm 0.000}$
30	2.67 ± 0.35	0.006 ± 0.001	$\mathbf{0.005 \pm 0.001}$
100	9.89 ± 0.55	0.024 ± 0.002	$\mathbf{0.021 \pm 0.003}$
500	303.67 ± 3.83	0.622 ± 0.002	$\mathbf{0.567 \pm 0.005}$
1000	2265.17 ± 43.49	4.346 ± 0.034	$\mathbf{4.258 \pm 0.027}$
wbc			
10	2.11 ± 0.33	0.008 ± 0.005	$\mathbf{0.001 \pm 0.000}$
30	3.00 ± 0.43	0.010 ± 0.001	$\mathbf{0.004 \pm 0.000}$
100	11.61 ± 0.49	0.050 ± 0.013	$\mathbf{0.023 \pm 0.001}$
500	344.89 ± 2.19	0.778 ± 0.016	$\mathbf{0.737 \pm 0.132}$
1000	2431.72 ± 9.55	4.987 ± 0.053	$\mathbf{4.660 \pm 0.043}$

ELM-fis) com o **ELM-reg**. Para cada uma das medidas de desempenho, foram realizados dois *t-testes* subsequentes (Montgomery, 2017), com correção de Bonferroni (McDonald, 2009) para o valor de $\alpha = 0.05$, o que leva a $\alpha_{bonferroni} = 0.025$.

Para todos os testes, a hipótese nula consistiu em: $H_0 : \mu_{ELM-reg} = \mu_{ELM-sil} = \mu_{ELM-fis}$.

Já a hipótese alternativa, consistiu em:

$$\begin{cases} H_{A1} : \mu_{ELM-reg} > \mu_{ELM-sil} \\ H_{A2} : \mu_{ELM-reg} > \mu_{ELM-fis} \end{cases} \quad (12)$$

Os *p-valores* encontrados indicam que, para o nível de significância adotado, em termos de tempo de execução da etapa de treinamento, rejeita-se a hipótese nula de que o tempo computacional gasto para seleção do parâmetro de regularização por validação cruzada é o mesmo tempo gasto para seleção do parâmetro com as técnicas propostas (mais rápidas).

Em termos de acurácia de teste, a técnica **ELM-sil** foi diferente (inferior) da técnica **ELM-reg** com significância estatística. Já para a técnica **ELM-fis**, não foram observados indícios suficientes para a rejeição da hipótese nula de igualdade. Os resultados (média das diferenças e *p-valor*) podem ser observados na Tabela 4.

Tabela 4: Resultados do teste t para a acurácia de teste

Estratégia	Média	*p-valor*
(ELM-reg - ELM-fis)	0.0020	0.234
(ELM-reg - ELM-sil)	0.0065	0.004

Uma justificativa possível, no entanto, para a utilização do método ELM-sil ao invés do método ELM-reg, consiste no ganho em termos de velocidade (especialmente para um maior número de neurônios) ao custo de uma perda pequena em termos de acurácia ($mean(Acc_{ELM-reg} - Acc_{ELM-reg}) = 0.65\%$).

Para a estratégia baseada no critério de Fisher, é possível que resultados melhores sejam obtidos com a ortogonalização da matriz **H**. Já para a abordagem baseada em Silhueta, será in-

teressante buscar uma função $\lambda(sil)$ melhor que aquela apresentada na Equação 10, que, como mencionado, foi obtida de forma empírica e mostrou resultados promissores, mas ainda não equivalentes àqueles obtidos por validação cruzada.

5 Conclusão

Este trabalho é um ponto de partida para a investigação da importância de medidas de separabilidade dos dados na seleção automática do parâmetro de regularização de Tikhonov.

Para as duas metodologias propostas, foram obtidos resultados capazes de rivalizar com os resultados obtidos quando o parâmetro de regularização foi selecionado por validação cruzada, com ganhos significativos de tempo.

O trabalho abre a perspectiva para a seleção de parâmetro de regularização de redes ELM sem a necessidade do procedimento de validação cruzada, a partir da estrutura dos dados projetados, de uma forma que os autores acreditam ser inédita.

Restam a explorar alguns pontos, em particular, a busca por relações menos empíricas e mais teóricas entre os valores de Silhueta e Critério de Fisher e o valor de λ para regularização.

6 Agradecimentos

Os autores gostariam de agradecer o apoio do CNPQ, da CAPES-Brasil e da FAPEMIG.

Referências

Cover, T. M. (1965). Geometrical and statistical properties of systems of linear inequalities with applications in pattern recognition, *IEEE transactions on electronic computers* (3): 326–334.

Deng, W., Zheng, Q. and Chen, L. (2009). Regularized extreme learning machine, *Computational Intelligence and Data Mining, 2009. CIDM'09. IEEE Symposium on*, IEEE, pp. 389–395.

Dheeru, D. and Karra Taniskidou, E. (2017). UCI machine learning repository.

Duda, R. O., Hart, P. E. and Stork, D. G. (1973). *Pattern classification*, Wiley, New York.

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

Huang, G.-B., Chen, L., Siew, C. K. et al. (2006). Universal approximation using incremental constructive feedforward networks with random hidden nodes, *IEEE Trans. Neural Networks* **17**(4): 879–892.

Huang, G.-B., Zhou, H., Ding, X. and Zhang, R. (2012). Extreme learning machine for regression and multiclass classification, *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* **42**(2): 513–529.

Huang, G.-B., Zhu, Q.-Y. and Siew, C.-K. (2004). Extreme learning machine: a new learning scheme of feedforward neural networks, *Neural Networks, 2004. Proceedings. 2004 IEEE International Joint Conference on*, Vol. 2, IEEE, pp. 985–990.

James, G., Witten, D., Hastie, T. and Tibshirani, R. (2013). *An introduction to statistical learning*, Vol. 112, Springer.

Kaufman, L. and Rousseeuw, P. J. (2009). *Finding groups in data: an introduction to cluster analysis*, Vol. 344, John Wiley & Sons.

McDonald, J. H. (2009). *Handbook of biological statistics*, Vol. 2, Sparky House Publishing Baltimore, MD.

Miche, Y., Sorjamaa, A., Bas, P., Simula, O., Jutten, C. and Lendasse, A. (2010). Op-elm: optimally pruned extreme learning machine, *IEEE transactions on neural networks* **21**(1): 158–162.

Miche, Y., Van Heeswijk, M., Bas, P., Simula, O. and Lendasse, A. (2011). Trop-elm: a double-regularized elm using lars and tikhonov regularization, *Neurocomputing* **74**(16): 2413–2421.

Montgomery, D. C. (2017). *Design and analysis of experiments*, John wiley & sons.

R Core Team (2015). *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria.

Silvestre, L. J., Lemos, A. P., Braga, J. P. and Braga, A. P. (2015). Dataset structure as prior information for parameter-free regularization of extreme learning machines, *Neurocomputing* **169**: 288–294.

Tikhonov, A. (1963). Solution of incorrectly formulated problems and the regularization method, *Soviet Meth. Dokl.* **4**: 1035–1038.

Vapnik, V. (2013). *The nature of statistical learning theory*, Springer science & business media.