

Classificador Incremental de Margem Larga com Memória Parcial via Grafo de Gabriel

Marcus V. F. Diadelmo, Luiz C. B. Torres, Victor Bittencourt, Frederico G. Guimarães, Cristiano L. Castro e Antônio P. Braga

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

Email: {marcusdiadelmo,luizlitc,frederico.g.guimaraes,crislcastro,adpbraga}@gmail.com e vlbitten@hotmail.com

Resumo—Devido a gama de problemas que trabalham com grandes volumes de dados ou que trabalham com dados obtidos de forma temporal, surge a necessidade de utilizar Algoritmos de Aprendizado Incrementais. Dessa forma, este trabalho apresenta uma técnica de aprendizado incremental baseado em memória parcial e janelas temporais para construção de um classificador de margem larga. A cada janela temporal são obtidos os vetores geométricos (vetores na região de separação entre as classes) a partir da estrutura de um Grafo de Gabriel. Foram realizados testes estatísticos pareados entre a metodologia incremental e a metodologia clássica, a qual utiliza todo o conjunto para gerar o modelo preditor. Resultados preliminares mostraram a equivalência dos resultados das duas metodologias, mostrando que a técnica de aprendizado incremental proposta é promissora.

Palavras Chave: Aprendizado Incremental, Grafo de Gabriel, Memória Parcial, Janelas Temporais.

I. INTRODUÇÃO

Em problemas reais um desafio para a aprendizagem de máquina está na capacidade de manter permanentemente um modelo preditor com uma boa taxa de acerto, dado que as distribuições de dados que representam os problemas podem sofrer mudanças em suas propriedades estatísticas, podendo tornar o modelo preditor obsoleto. Problemas deste tipo requerem algoritmos de aprendizagem que possam modificar o modelo atual sempre que novos conjuntos de dados estão disponíveis. Além de atualizar o modelo, estes algoritmos devem esquecer determinadas informações antigas [1].

Esse cenário exige um processamento contínuo dos dados de entrada, bem como o acompanhamento das tendências e detecção de alterações. Os sistemas tradicionais, baseados em memória, geram modelos estáticos treinados a partir de um conjunto de treinamento fixo e não estão preparados para processar dados com alto nível de complexidade, nem mesmo são capazes de manter continuamente um modelo preditivo consistente com o estado real da natureza, muito menos são capazes de reagir rapidamente às mudanças [2].

Algoritmos que processam fluxos de dados contínuos geralmente entregam soluções aproximadas, proporcionando uma resposta rápida usando poucos recursos de memória. É necessário ter o compromisso entre recursos computacionais (memória disponível, tempo treinamento) e precisão do modelo, reduzindo recursos mas obtendo um modelo, mesmo

que não muito preciso, capaz de resolver o problema de classificação.

No aprendizado incremental o algoritmo deve ser capaz de atualizar o modelo quando existem novos dados disponíveis. Este conceito se assemelha ao aprendizado de agentes inteligentes, já que estes trabalham em ambientes dinâmicos, que mudam com o decorrer do tempo. O aprendizado incremental possui, ainda, a capacidade de armazenar somente informações relevantes e necessárias para uma futura adaptação. Assim estes algoritmos devem possuir a capacidade de melhorar sua base de conhecimento.

De acordo com [1] pode-se destacar algumas metodologias utilizadas para treinamento incremental, como *Single Example*, onde a cada novo dado o algoritmo é re-treinado, alguns autores denominam esta técnica de treinamento online. Existe a técnica de *Multiple Examples*, onde é utilizado um conjunto de dados recentes, janelas, para re-treinar o modelo, podendo este conjunto ser de tamanho fixo ou não. A técnica de memória parcial consiste em armazenar dados representativos e utilizá-los posteriormente para gerar um novo modelo. Estruturas Geométricas estão sendo muito utilizadas para obter dados relevantes, dados que representem a margem de separação, no intuito de obter classificadores de margem larga, como por exemplo a utilização de fecho convexo [3] e do Grafo de Gabriel [4].

Neste trabalho é apresentada uma técnica de aprendizado incremental baseada em memória parcial e janelas temporais para construção de um classificador de margem larga. A cada janela temporal são obtidos os vetores geométricos (vetores na região de separação entre as classes) a partir da estrutura de um Grafo de Gabriel [4]. A técnica apresentada é inspirada em [5], onde é dito que a junção de diversos classificadores *fracos*, podem gerar um classificador *forte*. A técnica proposta foi testada em algumas bases de dados da UCI [6] e foi realizado um teste de comparação estatística com uma técnica de aprendizado tradicional, onde todo o conjunto de dados é dividido em treinamento e teste. A ideia é mostrar que o método proposto é promissor através da comparação com uma metodologia que utiliza toda a informação da base de dados para gerar um modelo, diferentemente do aprendizado incremental que utiliza janelas, informações temporais e descartes de informação consideradas como irrelevantes para o aprendizado.

O restante do artigo encontra-se organizado da seguinte forma. A Seção II apresenta os conceitos do aprendizado incremental. A Seção III descreve a formulação do grafo de Gabriel, e na Seção IV a formulação para construção do classificador de margem larga é descrita. Na Seção V, nossa abordagem de aprendizado incremental baseada em janelas é apresentada. Na Seção VI, são descritos a metodologia adotada na condução dos experimentos, e os resultados obtidos com a aplicação do novo método. Finalmente, a Seção VII traz as conclusões.

II. APRENDIZADO INCREMENTAL

Algoritmos incrementais são desejáveis em ambientes dinâmicos ou não estacionários (muitos casos reais), que mudam de forma inesperada. Estes algoritmos possuem a capacidade de incorporar novos dados e se adaptar à nova realidade do problema pois, diferentemente dos modelos tradicionais, os modelos incrementais aprendem ao longo do tempo com a apresentação de novos dados [7].

O aprendizado incremental, além de exigir a atualização do modelo a partir de novas informações, requer o descarte ou esquecimento de informações antigas, as formas de obtenção do modelo preditivo dependem de fatores como os recursos computacionais. De forma a exemplificar o funcionamento de um algoritmo de aprendizado incremental, consideremos um modelo preditor L e os dados de predição $y = L(X)$, a cada conjunto X_t um novo modelo pode ser obtido utilizando as informações mais recentes $L_{t+1} = \text{train}((X_t, y_t), L_t)$, ou um modelo pode ser obtido utilizando dados passados $L_{t+1} = \text{train}((X_i, y_i), \dots, (X_t, y_t), L_t)$. Existem diversas formas de lidar com dados incrementais, dentre elas podemos citar (i) a técnica de memória parcial, a qual armazena alguns dados e regularmente os utilizamos no treinamento, (ii) a técnica baseada em janelas, que recebe os dados temporais em janelas de tamanho fixo ou variável e (iii) técnica baseada em instância em que a cada novo dado apresentado é realizada a atualização do modelo [1]. Em [5] são descritas algumas técnicas existentes de algoritmos incrementais, além da proposta de uma nova técnica baseada na junção de diversos "classificadores fracos" para obtenção de um "classificador forte". A Figura 1 mostra um esquema genérico de um Algoritmo incremental. O fluxo 1 representa o fluxo natural do aprendizado, ou seja, receber os dados armazená-los e treiná-los. O fluxo 2 representa o processo incremental, uma vez que a partir da obtenção dos dados armazenados em memória e do resultado de predição do modelo é possível detectar mudança na distribuição dos dados e então realizar uma possível atualização no modelo.

Este trabalho é baseado na proposta descrita em [5], onde os dados são armazenados em janelas de tamanho fixo e dessas janelas são escolhidos alguns dados para compor o novo modelo, os demais são descartados. Dessa forma, este trabalho utiliza também o conceito de memória parcial e janelas. A metodologia proposta será detalhada um pouco mais adiante.

III. GRAFO DE GABRIEL

Para a obtenção do modelo preditor, será utilizado um subconjunto de dados provenientes dos dados de treinamento, este subconjunto é formado pelos chamados Vetores Geométricos (VG) [4], os quais são determinados a partir do Grafo de

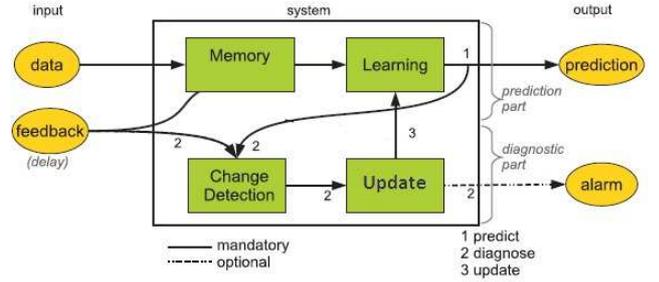


Figura 1: Esquema genérico de um algoritmo incremental.

Gabriel (GG) [8]. O GG é um subconjunto do diagrama de Voronoi e da Triangulação de Delaunay [9].

A. Definição

Seja o conjunto de dados $K = \{(\mathbf{x}_i, \mathbf{y}_i) \mid i = 1 \dots N\}$, onde $\mathbf{y}_i \in \{+1, -1\}$ e $\mathbf{x}_i \in \mathbb{R}^n$, o grafo de Gabriel GG de K possui um conjunto de vértices $V = \{\mathbf{x}_i \in K \mid i = 1, 2, \dots, N\}$ e um conjunto de arestas A , que são pares de vértices não orientados pertencentes a V . Uma aresta será formada entre os vértices \mathbf{x}_i e \mathbf{x}_j se, e somente se,

$$\delta^2(\mathbf{x}_i, \mathbf{x}_j) \leq [\delta^2(\mathbf{x}_i, \mathbf{x}_z) + \delta^2(\mathbf{x}_j, \mathbf{x}_z)], \quad (1)$$

$\forall \mathbf{x}_z \in V$ e $i \neq j \neq z$, onde $\delta(\cdot)$ é o operador que calcula a distância Euclidiana entre os vértices.

A Figura 2 mostra o Grafo de Gabriel para um determinado conjunto de dados. Os VGs são os pares de vértices de classes distintas que compartilham a mesma aresta.

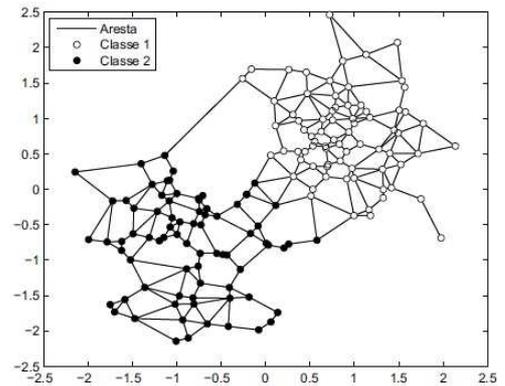


Figura 2: Grafo de Gabriel.

A obtenção do Grafo de Gabriel fornece uma importante informação sobre a margem de separação de classes, uma vez que amostras de classes distintas interligadas por um vértice do GG são amostras pertencentes às bordas de separação, tais amostras são chamadas de Vetores Geométricos. A utilização dos Vetores Geométricos proporciona informações importantes para obtenção de um classificador de Margem Larga. Uma forma de obter um classificador de margem larga utilizando os Vetores Geométricos será descrita na seção IV.

IV. CLASSIFICADOR

O classificador utilizado neste trabalho é baseado na combinação de classificadores de margem máxima definidos por todos pares de Vetores Geométricos [10]. Nesta abordagem, a combinação é ponderada por uma função de nível de compatibilidade baseada na distância Euclidiana entre uma amostra a ser classificada e cada hiperplano.

Seja H_i um hiperplano local gerado entre duas amostras $(\mathbf{v}_1, \mathbf{v}_2)$ onde o rótulo de $(\mathbf{v}_1) = -1$ e $(\mathbf{v}_2) = +1$, como ilustrado na Figura 3. O hiperplano passa exatamente pelo ponto mediano (ponto médio) entre \mathbf{v}_1 e \mathbf{v}_2 . Para cada hiperplano H_i existira um ponto médio \mathcal{PM}_i [10].

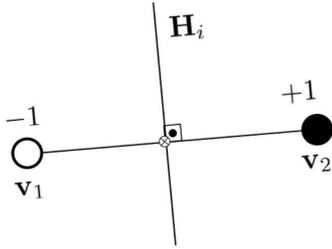


Figura 3: Definição do hiperplano local entre o par $(\mathbf{v}_1, \mathbf{v}_2)$ do grafo de Gabriel

A expressão que define H_i é dada por [3], e descrita como

$$H_i(\mathbf{x}, \mathbf{w}_i, b_i) = \mathbf{x}^T \mathbf{w}_i - b_i \quad (2)$$

sendo $\mathbf{w}_i = \mathbf{v}_1 - \mathbf{v}_2$ o vetor normal do hiperplano onde sua orientação é descrita através das seguintes inequações

$$\begin{aligned} \mathbf{x}^T \mathbf{w}_i - b_i &\geq 0, \\ \mathbf{x}^T \mathbf{w}_i - b_i &< 0 \end{aligned}$$

onde o *bias* b_i é calculado como

$$b_i = \left(\frac{\mathbf{v}_1 + \mathbf{v}_2}{2} \right) \mathbf{w}_i^T. \quad (3)$$

A. Classificação

Dado um conjunto de m hiperplanos locais $\{H_1, H_2, \dots, H_m\}$, a classificação de uma amostra arbitraria \mathbf{x}_j é dada pelos seguintes passos

- 1) Calcular a saída $\tilde{y}_{i,j}$ para todos H_i

$$\tilde{y}_{i,j} = H_i(\mathbf{x}_j, \mathbf{w}_i, b_i), \quad \forall i = 1, \dots, m. \quad (4)$$

e distância

$$d_{i,j} = \delta^2(\mathbf{x}_j, \mathcal{PM}_i) \text{sgn}(\tilde{y}_{i,j}), \quad \forall i = 1, \dots, m \quad (5)$$

onde $d_{i,j}$ é a distância euclidiana entre \mathbf{x}_j e todos os pontos médios.

- 2) Obter o nível de compatibilidade da amostra \mathbf{x}_j para o i -ésimo hiperplano H_i

$$\omega_{i,j} = \exp \left[\frac{- \left| \left(\frac{1}{d_{i,j}} \right) \right|}{\sigma^2} \right] \quad \forall i = 1, \dots, m \quad (6)$$

onde

$$\sigma = \frac{1}{\max(d_{i,j})}. \quad (7)$$

- 3) Dado que $\omega_{i,j}$ é inversamente proporcional a distância de \mathbf{x}_j até H_i . Uma normalização é imposta, tal que o somatório dos níveis de compatibilidade da amostra \mathbf{x}_j seja igual a uma unidade

$$\sum_{i=1}^m \omega_{i,j} = 1 \quad \forall \mathbf{x}_j. \quad (8)$$

- 4) Por fim, a função de decisão $f(\mathbf{x}_j)$ é uma combinação linear do valor de saída de cada hiperplano ponderado pelos níveis de compatibilidade correspondente a \mathbf{x}_j . Esta função pode ser escrita como

$$f(\mathbf{x}_j) = \begin{cases} +1 & \text{se } \sum_{i=1}^m \omega_{i,j} \text{sgn}(\tilde{y}_{i,j}) \geq 0 \\ -1 & \text{caso contrário} \end{cases} \quad (9)$$

V. ALGORITMO PROPOSTO

Os dados são apresentados de forma sequencial e armazenados em um Janela Temporária (JT) de tamanho fixo. Do instante 0 até $k_0 = 0 + \Delta t_0$ a JT recebe um conjunto de dados e, através do Grafo de Gabriel, os Vetores Geométricos (VG's) são determinados e armazenados em outra janela, chamada de Janela Geométrica (JG), os demais dados são descartados, estando, assim JT livre para receber novos dados. Sendo que Δt_0 é o tempo necessário para realizar este processo. Do instante k_0 até $k_1 = k_0 + \Delta t_1$ o processo se repete e assim acontece sucessivamente até k_n . Dessa forma, em JG estarão contido os Vetores Geométricos das n repetições do processo de armazenamento em JT e determinação dos VG's. O modelo de classificador é então gerado utilizando os dados contidos em JG. Um esquema dessa metodologia é mostrado na Figura 4.

O Filtro GG tem a função de selecionar os Vetores Geométricos da JT atual e descartar os demais dados.

O agente decisor é responsável por informar o momento de atualização do modelo, estudos detalhados estão sendo realizados para definir a metodologia de decisão, podendo ser desde medição de acurácia até a detecção de mudança [1]. Neste trabalho o agente decisor decide realizar o treinamento assim que a Janela Geométrica esteja cheia ou não exista mais dados de entrada.

Vale salientar que o classificador pode ser gerado a qualquer momento, desde que JG não esteja vazia. A escolha do momento de obter um novo classificador pode ser feita de diversas formas, neste trabalho são consideradas taxas de erro de classificação. Ao utilizar a JG para gerar o classificador, são encontrados os VG's dessa janela, dessa forma a JG pode ser atualizada mantendo somente os VG's e descartando os demais dados, reduzindo assim a quantidade de dados na JG.

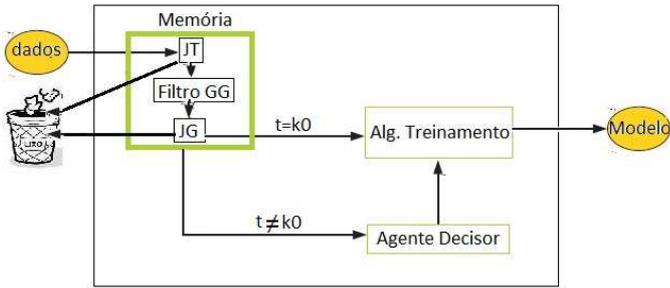


Figura 4: Esquema da metodologia Aprendizado Incremental em janelas. Sendo JT janela temporária e JG janela geométrica.

É importante destacar que este tipo de aprendizado realiza o descarte dos dados devido às restrições de memória. Em problemas com grande volume de dados e pouca memória, a utilização de técnicas tradicionais, onde todo conjunto de dados é utilizado para gerar o modelo, se torna impossível. Assim o modelo incremental pode ser avaliado através de uma base menor, onde é possível realizar o aprendizado tradicional, e comparado ao modelo tradicional, sendo que a equivalência entre os modelos é suficiente para validar o modelo incremental.

Na Figura 5 é possível observar um resultado obtido pelo janelamento. Nesta Figura, os dados em azul e vermelho são os dados presentes na Janela Geométrica, os dados em verde representam os dados descartados durante o processo, sendo que, a Janela Temporária tem uma capacidade de armazenar 20 dados. Nesta Figura são mostrados também os vértices do Grafo de Gabriel que interligam os Vetores Geométricos bem como os pontos médios que são utilizados na geração do classificador. No caso do aprendizado clássico, todos os dados, verdes, vermelho e azul, seriam utilizados para gerar o modelo.

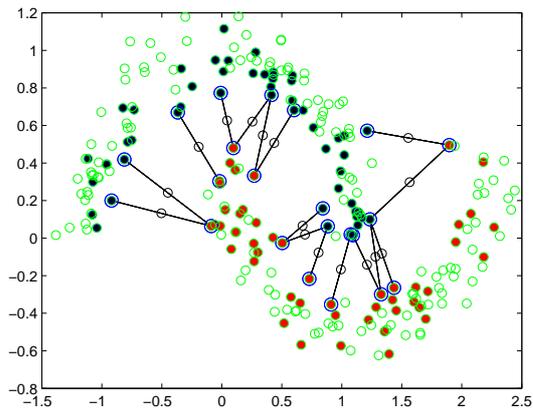


Figura 5: Aprendizado Incremental em janelas.

VI. RESULTADOS

Para analisar o desempenho da metodologia proposta, serão utilizados três bases de dados muito utilizadas para avaliação de classificadores [6]. Os dados foram distribuídos

em conjuntos de treinamento e teste e avaliados utilizando duas metodologias. Uma delas consiste em treinar um modelo utilizando todos os dados de treinamento e avaliar o modelo com os dados de teste, ou seja, a metodologia tradicional. A segunda metodologia utilizada foi a proposta neste trabalho, da mesma forma, o conjunto foi avaliado através dos dados de teste.

Para cada base de dados foi realizado um teste de hipótese estatística t pareado a fim de verificar a equivalência dos métodos incremental e tradicional, para isso foram realizado 30 execuções de cada algoritmo para cada base de dados. O teste consiste em analisar se as médias dos modelos são equivalentes, para isso utilizou-se um nível de significância de 0.05 e uma potência de 80%. Abaixo temos o teste de hipótese realizado.

$$\begin{cases} H_0 : \mu_1 - \mu_2 = 0 \\ H_1 : \mu_1 - \mu_2 \neq 0 \end{cases}$$

A. Sonar

Esta base de dados foi utilizada por [11] no estudo sobre classificação de dados de Sonar na diferenciação de rochas cilíndricas e um metal cilíndrico. A base possui 208 amostras, cada um com um conjunto de 60 características e uma característica indicando a qual classe o dado pertence. A Figura 6 mostra um boxplot do desempenho das duas metodologias para um total de 30 execuções para cada uma delas.

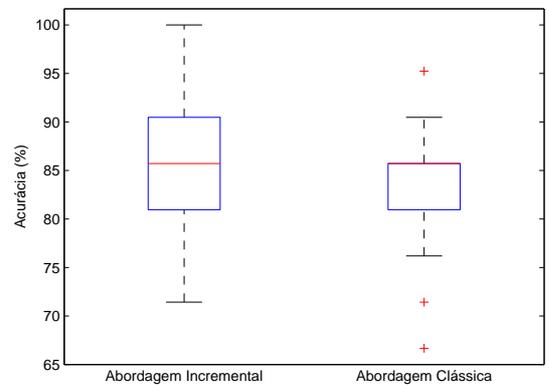


Figura 6: Desempenho dos classificadores para a base Sonar.

A Figura 7 mostra a comparação das médias das metodologias com um intervalo de confiança de 95%. Nela, é possível verificar a equivalência das médias, uma vez que a diferença zero está dentro do intervalo de confiança, dessa forma não podemos rejeitar a hipótese de igualdade das médias

B. Parkinson

Esta base de dados foi utilizada por [12] no estudo de distúrbios de voz. A base possui 195 amostras, cada um com um conjunto de 23 características e uma característica indicando a qual classe o dado pertence. A Figura 8 mostra um boxplot do desempenho das duas metodologias para um total de 30 execuções para cada uma delas.

95% family-wise confidence level

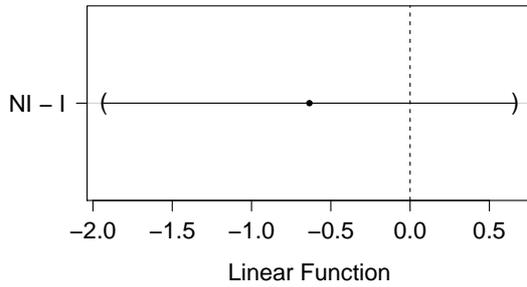


Figura 7: Intervalo de Confiança para comparação de desempenho médio. NI - Não Incremental, I - Incremental.

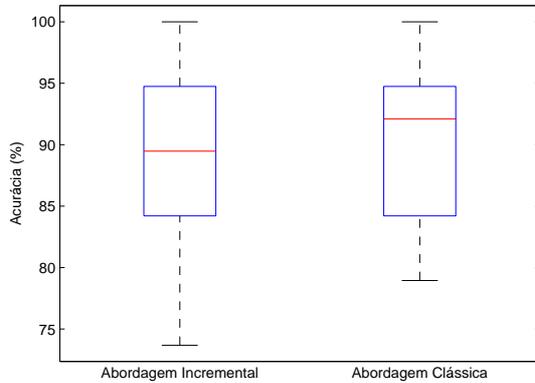


Figura 8: Desempenho dos classificadores para a base Parkinson.

A Figura 9 mostra a comparação das médias das metodologias com um intervalo de confiança de 95%. Nela, é possível verificar a equivalência das médias, uma vez que a diferença zero está dentro do intervalo de confiança, dessa forma não podemos rejeitar a hipótese de igualdade das médias

95% family-wise confidence level

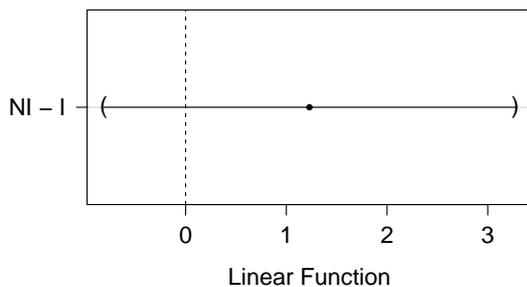


Figura 9: Intervalo de Confiança para comparação de desempenho médio. NI - Não Incremental, I - Incremental.

C. ILPD

Esta base possui 583 amostras, cada um com um conjunto de 10 características e uma característica indicando a qual classe o dado pertence [13]. A Figura 10 mostra um boxplot do desempenho das duas metodologias para um total de 30 execuções para cada uma delas.

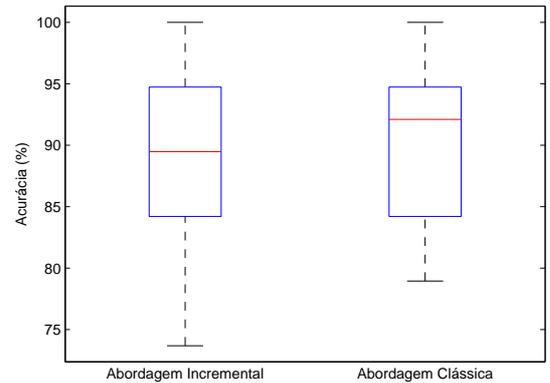


Figura 10: Desempenho dos classificadores para a base ILPD.

A Figura 11 mostra a comparação das médias das metodologias com um intervalo de confiança de 95%. Nela, é possível verificar a equivalência das médias, uma vez que a diferença zero está dentro do intervalo de confiança, dessa forma não podemos rejeitar a hipótese de igualdade das médias

95% family-wise confidence level

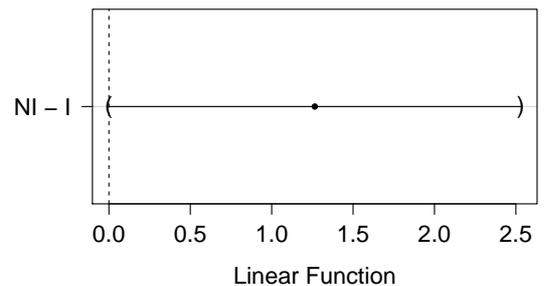


Figura 11: Intervalo de Confiança para comparação de desempenho médio. NI - Não Incremental, I - Incremental.

A Tabela I mostra também a comparação de desempenho, medido através da acurácia, das metodologias clássica e da proposta incremental através de um teste t pareado com 95% de nível de confiança e 80% de potência. Com os testes realizados, não podemos rejeitar a hipótese nula de igualdade de desempenho médio das metodologias.

Para os testes realizados, a execução de 30 rodadas de cada abordagem foi suficiente para garantir a potência requerida de 80% para os teste estatísticos, tornando forte e válida a não rejeição da hipótese nula de que as abordagens são equivalentes.

Tabela I: Comparação de Desempenho entre a Metodologia Clássica e a Metodologia Incremental proposta

Base de Dados	Intervalo de Confiança		
	Lim. Inferior	Estimativa	Lim. Superior
Sonar	-1,9332000	-0,6350000	0,6632000
Parkinson	-0,8149000	1,2290000	3,2729000
ILPD	-0,0000147	1,2650000	2,5290000

VII. CONCLUSÃO

A investigação da técnica de aprendizado incremental descrita neste artigo se mostrou promissora, dado que em uma comparação feita com a técnica de aprendizado tradicional, a qual utiliza toda informação do conjunto de treinamento para gerar um modelo, o método apresentado se mostrou equivalente, para as bases de dados testadas, em um teste estatístico com 95% de confiança. O fato de a metodologia proposta ser equivalente a metodologia tradicional mostra que o algoritmo proposto realiza o descarte dos dados e a geração de um modelo preditor incremental de forma eficiente, uma vez que, mesmo trabalhando com um conjunto de dados menor, consegue ser equivalente ao método tradicional que utiliza toda informação dos dados. Dessa forma, nota-se que a utilização das informações do Grafo de Gabriel unidas a memória parcial e janelamento pode ser uma estratégia eficiente para a obtenção de um classificador incremental.

Como trabalhos futuros, deseja-se estudar técnicas para definir o momento ideal de atualizar o modelo, respeitando restrições de memória e acurácia de classificação. Além de trabalhar com base de dados com grande volume de dados, onde os métodos tradicionais não podem ser aplicados.

AGRADECIMENTOS

O presente trabalho foi realizado com o apoio financeiro da CAPES - Brasil, CNPq e Fapemig.

REFERÊNCIAS

- [1] J. Gama, I. Zliobaité, A. Bifet, M. Pechenizkiy, and A. Bouchachia. *A survey on concept drift adaptation*. ACM Comput. Surv. 46, 4, Article 44 (March 2014), 37 pages, March 2014.
- [2] J. Gama. *A survey on learning from data streams: current and future trends*. Prog Artif Intell (2012) 1:45–55 DOI 10.1007/s13748-011-0002-6.
- [3] Bennett, K. P., Bredensteiner, E. J.: ‘Duality and geometry in SVM classifiers’, *ICML*, 2000, pp. 57-64
- [4] TORRES, L. C. B. ; COELHO, Frederico ; CASTRO, C. L. ; BRAGA, A. P. *A Graph of Gabriel Approach for Large Margin Classifiers*. In: LA-CCI - The Latin American Congress on Computational Intelligence Co-located with ARGENCON, 2014, San Carlos de Bariloche. Proceedings LA-CCI 2014, 2014. v. 1. p. 25-29.
- [5] Robi Polikar, Lalita Udpa, Satish S. Udpa. *Learn++: An Incremental Learning Algorithm for Supervised Neural Networks*. IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART C: APPLICATIONS AND REVIEWS, VOL. 31, NO. 4, NOVEMBER 2001

- [6] Hettich, S., Blake, C. L., Merz, C. J., 1998. *UCI repository of Machine learning Databases*, Department of Information and Computer Science, University of California, Irvine, CA, <http://www.ics.uci.edu/mllearn/MLRepository.html>.
- [7] C. Giraud-Carrier. 2000. *A note on the utility of incremental learning*. AI Commun. 13, 4 (Dec. 2000), 215–223.
- [8] Mark de Berg, Marc van Kreveld, Mark Overmars, and Otfried Schwarzkopf. *Computational Geometry: Algorithms and Applications*. Springer-Verlag, second edition, 2000.
- [9] ZHANG, W.; KING, I. *A study of the relationship between support vector machine and gabriel graph*. Neural Networks, IJCNN, vol. 1, p. 239–244, 2002. Proceedings of the 2002 International Joint Conference.
- [10] TORRES, L. C. B., CASTRO, C. L., BRAGA, A. P.; *A Parameterless Mixture Model for Large Margin Classification* In: International Joint Conference on Neural Networks, 2015, Killarney. IJCNN. , 2015
- [11] Gorman, R. P., and Sejnowski, T. J. (1988). *Analysis of Hidden Units in a Layered Network Trained to Classify Sonar Targets*: In Neural Networks, Vol. 1, pp. 75-89.
- [12] Little M.A., McSharry P.E., Roberts S.J., Costello D.A.E., Moroz I.M. *Exploiting Nonlinear Recurrence and Fractal Scaling Properties for Voice Disorder Detection*. BioMedical Engineering OnLine 2007.
- [13] Lichman, M. (2013). *UCI Machine Learning Repository* [<http://archive.ics.uci.edu/ml>]. Irvine, CA: University of California, School of Information and Computer Science.