B-Boost: Uma Extensão do Método de Boosting para Conjuntos de Treinamento Desbalanceados

Joseane P. Rodrigues, Ricardo B. C. Prudêncio, Flávia A. Barros

[1]Centro de Informática, Universidade Federal de Pernambuco
CEP 50732-970 - Recife (PE) - Brasil

josyrp@hotmail.com, rbcp@cin.ufpe.br, fab@cin.ufpe.br

Abstract. *Boosting methods have been well succeeded on a broad range of classification problems, being one of the most investigated approaches in the literature for ensembles of classifiers. Despite its potential performance gain, Boosting presents limitations when dealing with unbalanced training sets, i.e. sets presenting majority classes with size much higher than the others. In this context, we propose in this work the B-Boost method, an extension of Boosting for unbalanced training sets. Different from standard Boosting, the B-Boost performs, at each iteration, a sampling of training examples separately per class. The sampling in B-Boost is performed in such a way to generate a balanced training set containing the instances of each class which, at the iteration, are hard to be correctly classified. Experiments were performed comparing the proposed method to the standard Boosting. The results revealed that B-Boost can improve the classification performance for minority classes, which is an important aspect in different contexts of application.*

Resumo. *Métodos de Boosting têm se destacado em uma ampla quantidade de problemas de classificação, sendo uma das abordagens de combinação de classificadores mais investigadas na literatura. Apesar do potencial ganho de desempenho, Boosting apresenta limitações quando lidam com conjuntos de treinamento desbalanceados, i.e., com classes majoritárias de tamanho muito superior ao das outras classes. Dentro desse contexto, propomos o método B-Boost, uma extensão de Boosting para conjuntos de treinamento desbalanceados. Diferente de Boosting padrão, o método B-Boost realiza, a cada iteração, uma amostragem de exemplos separadamente por classe. A amostragem no B-Boost é feita de forma a gerar um conjunto de treinamento balanceado contendo as instâncias de cada classe que, na iteração, são difíceis de serem corretamente classificadas. Experimentos foram realizados para comparar o método proposto com o Boosting padrão. Os resultados revelaram que o B-Boost foi capaz de aumentar o desempenho de classificação para as classes minoritárias, o que é um aspecto importante em diferentes contextos de aplicação.*

1. Introdução

Métodos de combinação de classificadores têm sido usados com sucesso para aumentar o desempenho do processo de aprendizado [Kuncheva 2004]. Nesses métodos, as respostas individuais de um conjunto de classificadores diversos são combinadas para classificar novos exemplos. Espera-se que métodos de combinação sejam capazes de superar as

limitações individuais de cada classificador e explorar o bom desempenho que cada um possa apresentar em regiões específicas do espaço de exemplos [Dietterich 2000].

Boosting [Freund and Schapire 1997] é um método iterativo de combinação de classificadores, dentre os mais bem sucedidos da literatura. Em Boosting, a cada iteração, um classificador é induzido a partir de uma amostra de instâncias selecionadas do conjunto de treinamento. A seleção de instâncias é feita com base em probabilidades que são redefinidas a cada iteração de forma que instâncias mais difíceis de serem classificadas terão maior probabilidade de serem selecionadas nas iterações posteriores. A atualização das probabilidades leva em consideração o erro obtido pelo classificador no conjunto de treinamento. Após a construção de um número de classificadores, novos exemplos serão classificados através de combinação ponderada das respostas obtidas individuamente pelos classificadores.

Nesse trabalho, propomos uma variação do método de Boosting para tratamento de conjuntos de exemplos de treinamento desbalanceados. Conjuntos desbalanceados, i.e., onde o número de instâncias de uma dada classe é significativamente superior ao das classes minoritárias, em geral, introduzem dificuldades para os algoritmos de aprendizado. Comumente, o aprendizado em conjuntos desbalanceados geram classificadores tendenciosos, com uma alta taxa de acertos para a classe majoritária e um desempenho muito ruim para a classe minoritária. Assim como muitos algoritmos de aprendizado, Boosting têm limitações para lidar com esse tipo de situação, uma vez que as amostras geradas a cada iteração tendem também a ser desbalanceadas [Chawla et al. 2003].

No método proposto, referido como B-Boost, realizamos a amostragem de instâncias separadamente por classe a cada iteração, ao contrário do Boosting padrão em que a informação sobre as classes das instâncias não é levada em consideração. Inicialmente, para cada classe, é definida uma distribuição de probabilidade de seleção restrita às instâncias da classe. Isso é feito normalizando as probabilidades originais de Boosting definidas para as instâncias na iteração atual. Em seguida, é realizada a amostragem separadamente por classe de forma que cada classe contenha o mesmo número de instâncias da classe majoritária no conjunto de exemplos de treinamento. Assim, o B-Boost gera uma amostra de treinamento balanceada a cada iteração, contendo as instâncias mais difíceis de cada classe no momento.

Para testar a viabilidade da proposta, realizamos experimentos em 20 conjuntos de dados do repositório UCI [Asuncion and Newman 2007] com diferentes níveis de desbalanceamento. Nos experimentos, o método B-Boost foi comparado com uma implementação padrão de Boosting, o método AdaBoost.M1 [Freund and Schapire 1997]. Os resultados apontaram um aumento do desempenho de classificação usando o B-Boost, considerando as classes minoritárias. Observamos ainda, que o ganho de desempenho teve uma tendência a ser maior para conjuntos de dados que apresentavam um maior grau de desbalanceamento.

O restante do trabalho é organizado como se segue. A seção 2 apresenta uma breve explanação sobre métodos de combinação de classificadores, seguida pela seção 3 onde apresentamos o trabalho proposto, os experimentos realizados e resultados obtidos. Na seção 4, apresentamos trabalhos relacionados e finalmente, na seção 5, concluímos o artigo com considerações finais e trabalhos futuros.

2. Combinação de Classificadores

Combinação de classificadores tem como objetivo aumentar a precisão da classificação alcançada por apenas um classificador isoladamente [Kuncheva 2004]. Classificadores com comportamentos diferentes são combinados, de modo que o ponto fraco de um seja compensado pelo bom desempenho de outro. O uso de métodos de combinação foi introduzido ainda na década de 60 (e.g., [Nilsson 1965]), e desde então tem sido tema de muitas pesquisas na área de Aprendizagem de Máquina.

Diferentes estratégias podem ser adotadas para combinar as respostas de classificadores uma vez construídos, como exposto em [Kuncheva 2004]. Dentre as mais freqüentemente usadas se incluem Votação Majoritária, em que uma instância é classificada com a classe que recebe mais votos entre os classificadores sendo combinados, e a Combinação Linear Ponderada, em que os votos de cada classificador são ponderados por pesos que indicam a contribuição dos classificadores na resposta final.

Trabalhos em combinação de classificadores podem ser diferenciados ainda pela forma como os componentes da combinação são gerados. De uma forma geral, um comitê pode ser definido com: (1) classificadores homogêneos, quando foram gerados por um único algoritmo de aprendizado, como em Bagging [Breiman 1996] e Boosting [Freund and Schapire 1997]; e (2) classificadores heterogêneos, em que os classificadores combinados foram gerados por mais de um algoritmo de aprendizado, como ocorre comumente em Stacking [Wolpert 1995] e Meta Decision Trees [Todorovski and Džeroski 2003]. Métodos de combinação homogêneos apresentam algumas vantagens sobre os métodos heterogêneos, como uma maior simplicidade de aplicação, uma vez que apenas uma classe de algoritmos é utilizada para gerar os componentes da combinação. Além disso, o uso de métodos homogêneos apresenta motivações teóricas, como prova de convergência do erro de treinamento [Freund and Schapire 1997]. Assim, métodos homogêneos serão o foco do nosso trabalho.

Bagging é um método de combinação homogêneo que explora a instabilidade observada em alguns algoritmos de aprendizado, isto é, classificadores com comportamentos bastante distintos podem ser gerados a partir de pequenas variações do mesmo conjunto de treinamento. Classificadores gerados a partir de diferentes amostras de dados podem captar diferentes regularidades do problema de aprendizado sendo tratado. Neste caso, combinar tais classificadores poderia trazer um ganho na precisão na classificação. Em Bagging, T amostras são selecionadas aleatoriamente do conjunto de treinamento disponível. Em seguida, para cada amostra, o algoritmo em questão é usado para aprender um classificador diferente. Assim, T classificadores são construídos a partir da aplicação do mesmo algoritmo. Finalmente, um classificador final é construído através da combinação dos T classificadores, comumente por Votação Majoritária.

Boosting é um método homogêneo que, assim como Bagging, combina classificadores gerados usando amostras diferentes dos dados de treinamento. Entretanto, ele é um método iterativo, onde a amostra de uma iteração é selecionada considerando pesos (ou probabilidades) associados às instâncias, que variam conforme a precisão do classificador anterior para cada instância. A idéia principal de Boosting é que a cada iteração sejam selecionadas com maior probabilidade as instâncias que foram classificadas incorretamente nas etapas anteriores. Assim, espera-se aumentar a diversidade dos classificadores gerados.

Existem diversos algoritmos que implementam a idéia básica de Boosting, como exemplo o *AdaBoost.M1* [Freund and Schapire 1997], amplamente usado. Neste método, inicialmente é atribuído o mesmo peso a todas as instâncias no conjunto de treinamento (i.e., cada instância x_i tem a mesma probabilidade $D_t(i)$ de ser selecionada na primeira iteração). A partir da amostra de dados gerada em uma iteração t, um classificador h_t é construído. Em seguida, o classificador é usado para classificar todas as instâncias de treinamento e, posteriormente, o peso das instâncias é modificado, conforme a estimativa do *erro global* ε_t obtido pelo classificador (ver equação 1, onde $h_t(x_i)$ é a resposta obtida para a instância x_i e y_i é a classe correta de x_i). Se $\varepsilon_t \leq 1/2$, então a atualização das probabilidades D_{t+1} para a próxima iteração, é feita conforme as equações 2 e 3 (onde Z_t é um fator de normalização de forma que D_{t+1} seja uma distribuição). Se $\varepsilon_t > 1/2$, então a iteração é abortada e um novo passo é iniciado. Esse procedimento continua até a construção de um dado número T de classificadores.

$$\varepsilon_t = \sum_{i:h_t(x_i)\neq y_i} D_t(i) \tag{1}$$

$$\alpha_t = \frac{\varepsilon_t}{1-\varepsilon_t} \tag{2}$$

$$D_{t+1}(i) = \frac{D_t(i)}{Z_t} \times \begin{cases} \alpha_t, & se\ h_t(x_i) = y_i \\ 1, & caso\ contrario \end{cases} \tag{3}$$

A combinação final de Boosting é feita de forma ponderada, com a resposta de cada classificador h_t sendo ponderada pelo peso $log(1/\alpha_t)$. Desta maneira, um peso maior é dado a classificadores com baixo erro.

$$h_{final}(x) = argmax_{y\in Y} \sum_{t:h_t(x)=y} \log\left(\frac{1}{\alpha_t}\right) \tag{4}$$

3. Trabalho Realizado

Nesse trabalho, propomos o B-Boost, uma variação de Boosting para lidar com conjuntos de treinamento desbalanceados. Esses conjuntos se caracterizam pela existência de classes de tamanhos desproporcionais, com classes majoritárias com freqüência de exemplos muito superiores a das outras classes. Conjuntos de dados desbalanceados são um desafio para algoritmos de aprendizado, que comumente são baseados na diminuição de uma medida de erro médio, ou de outra medida de avaliação de predição, calculada de forma global para o conjunto de exemplos. Desta forma, o aprendizado é direcionado a classificar corretamente exemplos de classes majoritárias em detrimento dos exemplos de classes minoritárias que são menos significativas para o cálculo da medida de erro global [Visa and Ralescu 2005]. Assim, o classificador gerado pode ser de pouca utilidade, em especial, em contextos onde o custo de um erro para exemplos da classe minoritária é muito maior que o custo de um erro para exemplos majoritários [Chawla et al. 2004] (e.g., diagnóstico médico onde a classe minoritária é a presença de uma doença, ou detecção de fraudes onde a classe minoritária é relacionada à existência da fraude).

Na literatura, encontramos diferentes técnicas para balancear dados, sendo as mais comuns [Visa and Ralescu 2005]: (1) under-sampling - reduzir o tamanho dos dados das classes majoritárias até que elas fiquem com aproximadamente o mesmo tamanho das classes minoritárias (e.g., [Kubat and Matwin 1997]); over-sampling - aumentar o tamanho dos dados das classes minoritárias replicando exemplos (e.g., [Ling and Li 1998]) e; (3) modificação dos métodos de classificação - os métodos de classificação são modificados diretamente para diminuir a importância das classes majoritárias (e.g., [Akbani et al. 2004]).

No nosso trabalho, escolhemos usar o conceito de over-sampling no B-Boost, seguindo algumas considerações. Algoritmos de classificação adaptados para dados desbalanceados têm apresentado resultados promissores, e poderiam ser usados para geração dos classificadores a serem combinados em Boosting. No entanto, a aplicação de Boosting ficaria restrita a um número reduzido de algoritmos já adaptados. Dentre as técnicas de amostragem, under-sampling envolve perda de informação, por descartar potenciais dados úteis nas classes majoritárias. Assim, optamos por adotar over-sampling, tanto para tornar o B-Boost flexível em relação ao algoritmo-base usado para geração dos classificadores como para evitar perda significativa de informação.

A idéia básica do B-Boost é relativamente simples e consiste em modificar a fase de amostragem de instâncias do Boosting, realizando a seleção de instâncias separadamente por classe de instâncias. Para cada classe, é selecionado um dado número de instâncias da classe, de forma que as instâncias mais difíceis teriam maior probabilidade de serem selecionadas. Em Boosting padrão, as instâncias mais difíceis também são selecionadas com maior probabilidade, no entanto, a informação da classe das instâncias não é levada em consideração. O número de instâncias selecionadas para cada classe no B-Boost é igual ao tamanho da classe majoritária no conjunto de treinamento. Assim, em cada iteração, são geradas amostras com classes balanceadas, contendo as instâncias de cada classe mais difíceis no momento.

Formalmente, seja $D_t(i)$ a probabilidade de seleção da i-ésima instância de treinamento na iteração t. Na amostragem do B-Boost, para cada valor de classe $y \in Y$, são calculadas probabilidades intermediárias de seleção $D_t^{[y]}(i)$ para as instâncias associadas à classe (i.e., as instâncias x_i tal que $y_i = y$). Isso é feito normalizando as probabilidades atuais $D_t(i)$ das instâncias da classe, através da equação:

$$D_t^{[y]}(i) = \frac{D_t(i)}{\sum_{x_j : y_j = y} D_t(j)} \tag{5}$$

A normalização é feita para garantir que $D_t^{[y]}$ seja uma distribuição. Finalmente, seja n_{max} o tamanho da classe majoritária. Para cada $y \in Y$, são selecionadas n_{max} instâncias da classe y, considerando a distribuição $D_t^{[y]}$. A partir da amostra gerada, o método B-Boost procede como o método AdaBoost.M1 apresentado na seção 2. Um classificador é gerado a partir da amostra, o erro do classificador é calculado através da equação 1 e o cálculo das probabilidades $D_{t+1}(i)$ para a próxima iteração é feito através das equações 2 e 3. A classificação final do B-Boost é realizada através da equação 4.

Diferentes experimentos foram realizados para avaliar a viabilidade do método

proposto. Nas próximas seções, são descritos os experimentos realizados com o B-Boost, incluindo a descrição dos conjuntos de dados utilizados nos experimentos, a metodologia aplicada nos experimentos, bem como os resultados obtidos.

3.1. Conjuntos de Dados

Nos experimentos, utilizados 20 conjuntos de dados coletados do repositório UCI (ver tabela 1). Todos os conjuntos utilizados são referentes a problemas de classificação binários, embora o uso de B-Boost não seja restrito a esse tipo de problema. Isso foi feito, nesse primeiro momento de experimentos, para facilitar a medição do grau de desbalanceamento de cada conjunto. Na nossa análise, o grau de desbalanceamento de um conjunto foi medido como o valor da freqüência da classe majoritária do conjunto.

Nos conjuntos utilizados, consideramos tanto conjuntos relativamente balanceados (com freqüência da classe majoritária próximo de 50%), assim como conjuntos com maior grau de desbalanceamento (ver tabela 1, coluna 2). Nos 20 conjuntos, o grau de desbalanceamento foi em média de 64.80%, com valor mínimo de 53.36% e valor máximo de 93.87%.

3.2. Experimentos

Nos experimentos realizados, o método B-Boost foi comparado com o AdaBoost.M1 implementado no ambiente WEKA [Witten and Frank 2005]. Tanto no B-Boost como no AdaBoost.M1, utilizamos Decision Stumps (árvores de decisão com um nível) como algoritmo-base a ser combinado. Esse algoritmo-base é utilizado comumente em Boosting por apresentar um alto viés indutivo [Sun et al. 2007], o que é importante para gerar classificadores diversos. No nosso trabalho, foi utilizada a implementação de Decision Stumps do próprio ambiente WEKA. O número de iterações dos métodos de combinação, que corresponde ao número de classificadores sendo combinados, foi de 100 iterações.

Para cada conjunto de dados, foi realizada validação cruzada 10-fold. Utilizamos três critérios para avaliação dos resultados da classificação. Inicialmente, calculamos a precisão (taxa de acertos global) da combinação de classificadores, com teste-t para comparação estatística entre os métodos. Avaliamos ainda a taxa de verdadeiros positivos (True Positive rate) separadamente para a classe majoritária e para a classe minoritária do conjunto. Os resultados obtidos nos experimentos são apresentados e analisados na próxima sub-seção.

3.3. Resultados

A tabela 1 (colunas 3 e 6) traz as taxas de precisão global obtidas respectivamente por AdaBoost.M1 e B-Boost para os 20 conjuntos utilizados nos experimentos. De uma forma geral, a precisão global com B-Boost é menor que a precisão obtida com AdaBoost.M1. De fato, para os 20 conjuntos de dados considerados, B-Boost obteve em números absolutos uma perda de precisão em 15 conjuntos (embora, a diferença estatística tenha sido confirmada em apenas 3 conjuntos, considerando o teste-t, com 95% de confiança).

Um ganho de desempenho para o B-Boost é observado, no entanto, considerando a taxa de verdadeiros positivos para a classe minoritária (tabela 1, colunas 4 e 6), em especial, para conjuntos de dados que estão mais desbalanceados. Na figura 1, podemos observar que, de uma forma geral, existe uma tendência crescente do ganho obtido por

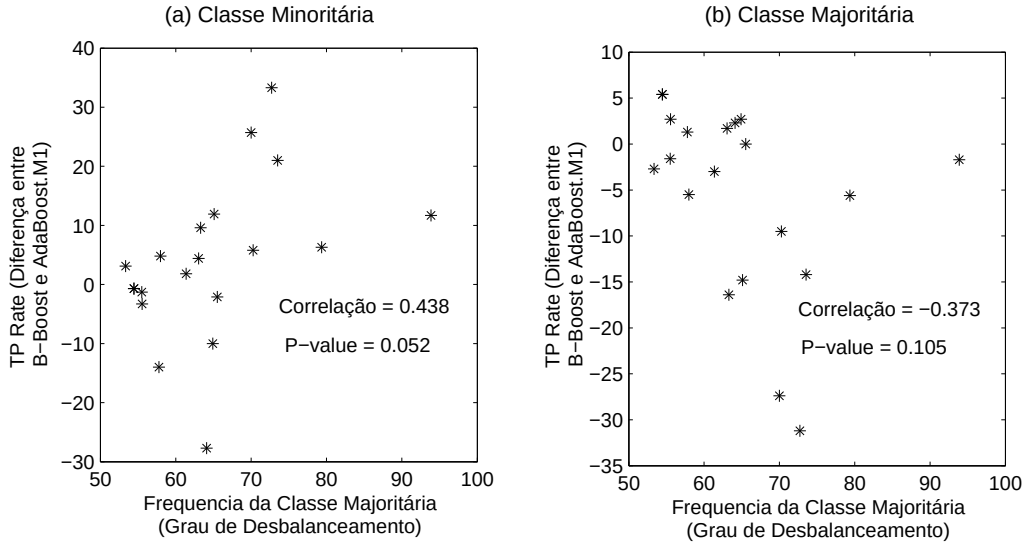

Figura 1. Diferença de TP Rate entre os métodos B-Boost e AdaBoost.M1 considerando o grau de desbalanceamento dos dados para: (a) classe minoritária e (b) classe majoritária.

B-Boost em relação ao grau de desbalanceamento dos dados. A correlação nessa figura foi de fato positiva (0.438), embora não possa ser confirmada estatisticamente (p-value para correlação foi um pouco maior que 0.05, se considerarmos um grau de confiança de 95%).

Para a classe majoritária, observamos que a taxa de verdadeiros positivos cai com o uso de B-Boost em comparação ao AdaBoost.M1 (tabela 1, colunas 5 e 7). No gráfico da figura 2, observamos que essa queda de desempenho se relaciona de forma negativa com o grau de desbalanceamento dos dados, ou seja, quanto maior o grau de desbalanceamento menor o desempenho de B-Boost para a classe originalmente majoritária. O valor absoluto da correlação nesse caso foi de 0.373 que é menor que a correlação observada para o ganho de B-Boost com a classe minoritária. Isso indicaria que proporcionalmente, o ganho obtido com B-Boost para as classes minoritárias é maior que a perda obtida para as classes majoritárias, levando em conta o grau de desbalanceamento dos dados. Experimentos em mais conjuntos de dados seriam necessários para confirmar essa suposição de forma estatística.

Considerando os resultados obtidos, B-Boost não seria indicado em situações onde o custo de um erro de classificação fosse equivalente para as classes majoritária e minoritária, uma vez que de uma forma geral, a precisão global do método foi pior que a precisão observada com o AdaBoost.M1. Por outro lado, o B-Boost seria indicado para dados com maior grau de desbalanceamento e onde o custo de um erro na classe minoritária fosse maior que o custo de um erro na classe majoritária. Desta forma, o ganho de precisão obtido pelo B-Boost para a classe minoritária seria importante para diminuir o custo global da classificação no contexto onde fosse aplicado.

Tabela 1. Resultados obtidos para 20 conjuntos de dados do repositório UCI.

Conjunto de Dados	Freqüência Majoritária	AdaBoost.M1			B-Boost		
		Precisão Global	TP Rate (Minor.)	TP Rate (Major.)	Precisão Global	TP Rate (Minor.)	TP Rate (Major.)
Breast-cancer	70.27	72.02	42.4	84.6	67.13	48.2	75.1
Breast-w	65.52	95.27	92.1	96.9	94.56	90	96.9
Cilinder-bands	57.77	75.47	63.6	84.6	70.55	49.6	85.9
Colic	63.04	80.7	72.1	85.8	83.42	76.5	87.5
Colic.Orig	63.30	83.42	70.2	90.2	85.47	79.8	73.8
Credit-a	55.50	86.66	85.7	87.5	85.21	84.4	85.9
Credit-g	70.00	73.9	43.3	87	62.4	69	59.6
Diabetes	65.10	75.52	57.5	85.2	70.05	69.4	70.4
Haberman	73.52	75.16	37	88.9	70.26	58	74.7
Heart-statlog	55.55	80.37	78.3	82	80.37	75	84.7
Heart-c	54.45	82.83	79	86.1	70.55	78.3	91.5
Heart-h	54.45	82.83	79	86.1	85.47	78.3	91.5
Hepatitis	79.35	82.48	53.1	90.2	79.35	59.4	84.6
Ionosphere	64.10	92.3	83.3	97.3	83.76	55.6	99.6
Labor	64.91	89.47	80	94.6	87.71	70	97.3
Liver-disorders	57.97	68.11	52.4	79.5	66.95	57.2	74
Postoperative	72.72	67.04	4.2	90.6	53.4	37.5	59.4
Sick	93.87	97.37	77.50	98.7	96.55	89.2	97
Sonar	53.36	82.21	80.40	83.8	83.76	83.5	81.1
Vote	61.37	97.01	96.40	97.4	95.86	98.2	94.4

4. Trabalhos Relacionados

Nessa seção, apresentamos alguns trabalhos que propuseram outras variações de Boosting para tratamento de conjuntos desbalanceados. Assim como o B-Boost, esses trabalhos têm em comum o fato de realizarem over-sampling dos exemplos da classe minoritária durante a geração de amostras de treinamento a cada iteração de Boosting.

Em [Guo and Viktor 2004], os autores propuseram o método DataBoost-IM que como o B-Boost realiza amostragem por classe. Nesse trabalho, a cada iteração o método ordena as instâncias de cada classe usando as probabilidades de seleção. Isso é feito para identificar as instâncias mais difíceis de cada classe (i.e., com maior probabilidade de seleção). Em seguida, para cada classe, é separado um determinado número de instâncias que são adicionadas diretamente na amostra de treinamento. Finalmente, a amostra é complementada de forma probabilística como no Boosting padrão. O método DataBoost-IM força que cada amostra de treinamento contenha tanto dados da classe majoritária como da classe minoritária. A limitação dessa proposta, no entanto, é definir quantas instâncias dentre as mais difíceis devem ser adicionadas na amostra de cada iteração. Além disso, eles adicionam instâncias na ordem de suas probabilidades, independentemente do valor das probabilidades em si. Assim, o método pode replicar exemplos com probabilidade relativamente baixa e que, de fato, não seriam exemplos difíceis.

Em [Sun et al. 2007], os autores propuseram um método de Boosting onde custos são introduzidos no cálculo das probabilidades de seleção. Nesse trabalho, cada exemplo recebe um valor de custo associado à classe a que pertence, de forma que classes minoritárias são associadas a valores de custo mais altos. O valor da probabilidade de

seleção de cada instância é ponderado pelo custo associado. Assim, a cada iteração, o método seleciona com maior probabilidade as instâncias mais difíceis de serem classificadas, levando em consideração também as classes das instâncias. Três equações diferentes foram propostas em [Sun et al. 2007] para considerar custos no cálculo das probabilidades de seleção. Idéias similares a esse trabalho podem ser utilizadas no futuro para para modificar a quantidade de exemplos minoritários replicados no B-Boost, visando controlar o conflito entre ganho para a classe minoritária e perda para a classe majoritária, observado nos experimentos descritos na seção anterior.

Finalmente, em [Chawla et al. 2003], os autores apresentam o método SMOTE-Boost em que exemplos sintéticos da classe minoritária são replicados em cada iteração do Boosting. Nesse trabalho, exemplos sintéticos são produzidos a partir de exemplos reais da classe minoritária, visando evitar um eventual risco de overfitting que pode ser gerado quando os mesmos exemplos da classe minoritária são replicados de forma excessiva para gerar um conjunto de treinamento balanceado.

5. Conclusões

Nesse artigo, apresentamos o B-Boost, que combina Boosting com over-sampling para balanceamento de conjuntos de treinamento. Experimentos foram realizados em 20 problemas de classificação com diferentes níveis de desbalanceamento de dados, comparando o B-Boost com o método de AdaBoost.M1. Os experimentos revelaram um ganho na taxa de verdadeiros positivos para a classe minoritária quando o B-Boost foi utilizado. Esse ganho, no entanto, foi obtido, em muitos casos, com uma perda de desempenho da classificação para a classe majoritária.

Diferentes trabalhos futuros podem ser apontados a partir do estado atual da pesquisa. Inicialmente, apesar do B-Boost ter sido comparado com o Boosting padrão, o método proposto deve ser comparado no futuro com outras propostas que trataram o problema de desbalanceamento de dados em Boosting (ver seção 4). Outro aspecto a considerar é que, na proposta atual do B-Boost, o tamanho da classe minoritária em cada amostra de treinamento foi definido de forma a se igualar ao tamanho da classe majoritária. Seria necessário investigar se esse critério, em certos problemas, não levaria a um over-sampling excessivo da classe minoritária, gerando problemas de overfitting (como mencionado em [Chawla et al. 2002]). Dentro desse contexto, pretendemos introduzir valores de custo associados às classes de instâncias (como em [Sun et al. 2007]), visando definir, conforme a tarefa de classificação, o montante de over-sampling que deve ser realizado para a classe minoritária.

Agradecimentos: Os autores agradecem pelo financiamento do CNPq e FAPESB.

Referências

Akbani, R., Kwek, S., and Japkowicz, N. (2004). Applying support vector machines to imbalanced datasets. In *Proceedings of the 15th European Conference on Machine Learning*, pages 39–50.

Asuncion, A. and Newman, D. (2007). Uci machine learning repository. University of California, Irvine, School of Information and Computer Sciences. http://www.ics.uci.edu/∼mlearn/MLRepository.html.

Breiman, L. (1996). Bagging predictors. *Machine Learning*, 24(2):123–140.

Chawla, N., Bowyer, K., Hall, L., and Kegelmeyer, W. (2002). Smote: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16:321–357.

Chawla, N., Japkowicz, N., and Kotcz, A. (2004). Editorial: special issue on learning from imbalanced data sets. *SIGKDD Explor. Newsl.*, 6(1):1–6.

Chawla, N., Lazarevic, A., Hall, L., and Bowyer, K. (2003). Smoteboost: Improving prediction of the minority class in boosting. *Lecture Notes in Computer Science*, 2838:107–119.

Dietterich, T. (2000). Ensemble methods in machine learning. *Lecture Notes in Computer Science*, 1857:1–15.

Freund, Y. and Schapire, R. E. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1):119–139.

Guo, H. and Viktor, H. (2004). Learning from imbalanced data sets with boosting and data generation: the databoost-im approach. *SIGKDD Explor. Newsl.*, 6(1):30–39.

Kubat, M. and Matwin, S. (1997). Addressing the curse of imbalanced data set: One sided sampling. In *Proceedings of the Fourteenth International Conference on Machine Learning*, pages 179–186.

Kuncheva, L. (2004). *Combining Pattern Classifiers - Methods and Algorithms*. John Wiley and Sons, New Jersey.

Ling, C. and Li, C. (1998). Data mining for marketing: Problems and solutions. In *Proceedings of the 4th International Conference on Knowledge Discovery and Data Mining*, pages 73–79.

Nilsson, N. J. (1965). *Learning Machines: Foundations of Trainable Pattern-Classifying Systems*. McGraw Hill, New York, EUA.

Sun, Y., Kamel, M., Wong, A., and Wang, Y. (2007). Cost-sensitive boosting for classification of imbalanced data. *Pattern Recognition*, 40(12):3358–3378.

Todorovski, L. and Džeroski, S. (2003). Combining classifiers with meta decision trees. *Machine Learning*, 50(3):223–249.

Visa, S. and Ralescu, A. (2005). Issues in mining imbalanced data sets - a review paper. In *Proceedings of the Sixteen Midwest Artificial Intelligence and Cognitive Science Conference*, pages 67–73.

Witten, I. H. and Frank, E. (2005). *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, San Francisco, 2 edition.

Wolpert, D. (1995). Stacked generalization. *Neural Networks*, 5:241–259.